

## Documentation

# EL1262, EL1262-0050

**Digital Input Terminal with oversampling** 

Version: 2.4 Date: 2018-11-13



# Table of contents

1	Forev	word	5
	1.1	Notes on the documentation	5
	1.2	Safety instructions	6
	1.3	Documentation issue status	7
	1.4	Version identification of EtherCAT devices	7
2	Prod	uct overview	12
	2.1	EL1262 - Introduction	12
	2.2	EL1262 - Technical data	13
	2.3	Technology	14
	2.4	Start	14
3	Basio	es communication	15
•	3.1	EtherCAT basics	
	3.2	EtherCAT cabling – wire-bound	
	3.3	General notes for setting the watchdog	
	3.4	EtherCAT State Machine	
	3.5	CoE Interface	-
	3.6	Distributed Clock	
4		nting and wiring	
·	4.1	Instructions for ESD protection	
	4.2	Installation on mounting rails	
	4.3	Installation instructions for enhanced mechanical load capacity	
	4.4	Connection system	
	4.5	Installation positions	
	4.6	Positioning of passive Terminals	
	4.7	ATEX - Special conditions (standard temperature range)	
	4.8	ATEX Documentation	
	4.9	UL notice	
	4.10	EL1262-xxxx - LEDs and pin assignment	41
5	Com	missioning	43
	5.1	TwinCAT Quick Start	43
		5.1.1 TwinCAT 2	45
		5.1.2 TwinCAT 3	55
	5.2	TwinCAT Development Environment	67
		5.2.1 Installation of the TwinCAT real-time driver	67
		5.2.2 Notes regarding ESI device description	73
		5.2.3 TwinCAT ESI Updater	77
		5.2.4 Distinction between Online and Offline	77
		5.2.5 OFFLINE configuration creation	78
		5.2.6 ONLINE configuration creation	83
		5.2.7 EtherCAT subscriber configuration	91
	5.3	General Notes - EtherCAT Slave Application	. 100
	5.4	Oversampling terminals and TwinCAT Scope	. 108
		5.4.1 TwinCAT 3 procedure	109

		5.4.2	TwinCAT 2 procedure	117
	5.5	Basic fur	nction principles	. 126
	5.6	Switchin	g characteristics	. 133
	5.7	Applicati	on of the SENT protocol with EL1262-0050	. 134
	5.8	Access t	o TEDS with EL1262-0050 and EL2262	. 135
	5.9	Example	programs	. 135
6	Appe	ndix		. 143
	6.1	EtherCA	T AL Status Codes	. 143
	6.2 Firmware compatibility		e compatibility	. 143
	6.3	Firmware	e Update EL/ES/EM/EPxxxx	. 143
		6.3.1	Device description ESI file/XML	144
		6.3.2	Firmware explanation	147
		6.3.3	Updating controller firmware *.efw	148
		6.3.4	FPGA firmware *.rbf	149
		6.3.5	Simultaneous updating of several EtherCAT devices	153
	6.4	Restoring	g the delivery state	. 154
	6.5	Support	and Service	. 155

# 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<sup>®</sup>, TwinCAT<sup>®</sup>, EtherCAT<sup>®</sup>, EtherCAT P<sup>®</sup>, Safety over EtherCAT<sup>®</sup>, TwinSAFE<sup>®</sup>, XFC<sup>®</sup> and XTS<sup>®</sup> 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<sup>®</sup> is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

#### Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

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

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

# 1.2 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.

#### **Risk of injury!**

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

#### **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
2.4	Chapter Commissioning/Example Programs: "Example 3: reading and writing TEDS data"     amended
2.3	"Application of the SENT protocol with EL1262-0050" section incl. example program (2) added within chapter "Commissioning"
2.2	Update chapter "Notes on the documentation"
	Update of Technical data
	Addenda chapter "Installation instructions for enhanced mechanical load capacity"
	<ul> <li>Update chapter "TwinCAT 2.1x" -&gt; "TwinCAT Development Environment" and "TwinCAT Quick Start"</li> </ul>
2.1	"Oversampling terminals and TwinCAT Scope" section added
2.0	Migration
1.5	Structural update
	"Technical data" section updated
	"LEDs and pin assignment" section updated
1.4	Structural update
	• EL1262-0050 amended
1.3	Technical data amended
1.2	Notes on device description update amended; note on trademarks inserted
1.1	Example program amended
1.0	Technical description added, first publication
0.1	Provisional documentation for EL1262

# 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	Туре	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

- y naroware 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: <u>EtherCAT Box</u>
- Safety: <u>TwinSafe</u>
- · Terminals with factory calibration certificate and other measuring terminals

#### Examples of markings



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



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



Fig. 3: CU2016 switch with serial/ batch number



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



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



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



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



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

# 2 **Product overview**

# 2.1 EL1262 - Introduction

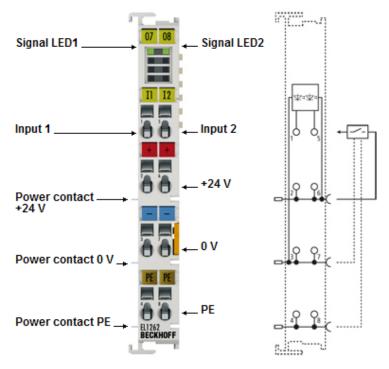


Fig. 9: EL1262-0000

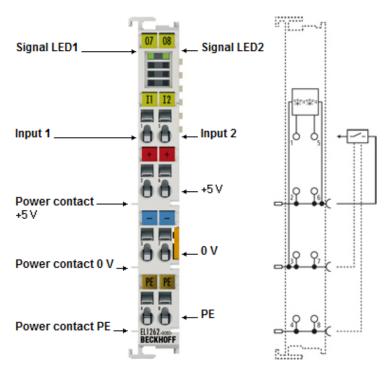


Fig. 10: EL1262-0050

#### 2-channel digital input terminal with oversampling

The EL1262 digital input terminal acquires the fast binary control signals from the process level and transmits them, in an electrically isolated form, to the controller. The signals are sampled with a configurable, integer multiple (oversampling factor: n) of the bus cycle time (n microcycles per bus cycle). For each

microcycle, the EtherCAT Terminal generates a process data block that is transferred collectively during the next bus cycle. The time base of the terminal can be synchronized precisely with other EtherCAT devices via distributed clocks. This procedure enables the temporal resolution of the acquisition of the digital input signals to be increased to n times the bus cycle time.

The EL1262-0050 offers a version with 5 V input voltage (TTL level) and 5 V supply voltage.

#### Quick links

- EtherCAT function principles
- LEDs and pin assignment [▶ 41]
- <u>Commissioning [} 43]</u>
- Basic function principles [▶ 126]

### 2.2 EL1262 - Technical data

Technical data	EL1262	EL1262-0050
Digital inputs	2	
Nominal voltage of the inputs	24 V <sub>DC</sub> (-15%/+20%)	5 V <sub>DC</sub> (-15%/+20%)
Signal voltage "0"	-3 V +5 V (based on EN 61131-2, type 3)	< 0.8 V
Signal voltage "1"	+11 V +30 V (based on EN 61131-2, type 3)	>2.4 V
Input current	typ. 3 mA (based on EN 61131-2, type 3)	typ. 50 μA
Input filter delay	typ. < 1 μs	
Oversampling factor	n = integer multiple of the cycle time,	11000
Distributed clock (DC) precision	<<1 µs	
Sampling rate	max. 1 Msamples/s	
Supply voltage for electronics	via the power contacts (24 V)	via the power contacts (5 V) (Read the notes $[\blacktriangleright 42]$ !)
Current consumption via E-bus	typ. 70 mA	
Electrical isolation	500 V (E-bus/field voltage)	
Bit width in process image	max. 254 bytes	
Configuration	via TwinCAT System Manager	
Weight	approx. 55 g	
Permissible ambient temperature range during operation	0°C + 55°C	
Permissible ambient temperature range during storage	-25°C + 85°C	
Permissible relative humidity	95%, no condensation	
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (v	width aligned: 12 mm)
Mounting [ > 27]	on 35 mm mounting rail conforms to	EN 60715
Vibration/shock resistance	conforms to EN 60068-2-6 / EN 6006 see also installation instructions for t	
	mechanical load capacity [▶ 30]	
EMC immunity/emission	conforms to EN 61000-6-2 / EN 6100	10-6-4
Protection class	IP20	
Installation position	variable	
Approval	CE <u>ATEX [] 38]</u>	
	<u>cULus [▶ 40]</u>	

#### Electrical supply

The fast input circuits of the EL1262-xxxx are supplied via the power contacts. Please note that the EL1262-0050 can only be operated with a 5 V supply voltage! If necessary, use a <u>EL9505</u> power supply terminal for supplying the EL1262-0050.

### 2.3 Technology

#### Functioning

The EL1262 is a digital input terminal with two channels. It can read the voltage level not only cyclically with the EtherCAT cycle, but also several times in between based on distributed clock support of the EL1262. In the EL1262 the ESC (EtherCAT slave controller) handles the data communication to the EtherCAT fieldbus and supports the distributed clock functionality. This enables the ESC to read the inputs of the EL1262 cyclically and equidistantly with high precision and store the values in the memory. When the EtherCAT frame fetches the data from the EL1262, a whole set of process data is ready for transfer. The inputs can be sampled with significantly higher frequency than the fieldbus cycle. Hence the term oversampling.

Further information on commissioning can be found in section "Basic function principles [> 126]"

### 2.4 Start

For commissioning:

- mount the EL12xx as described in the section Mounting and wiring [▶ 26]
- configure the EL12xx in TwinCAT as described in the section Commissioning [▶ 43].

# 3 Basics communication

### 3.1 EtherCAT basics

Please refer to the chapter <u>EtherCAT System Documentation</u> for the EtherCAT fieldbus basics.

### 3.2 EtherCAT cabling – wire-bound

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

#### Cables and connectors

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

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

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

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

#### Recommended cables

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

#### **E-Bus supply**

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

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

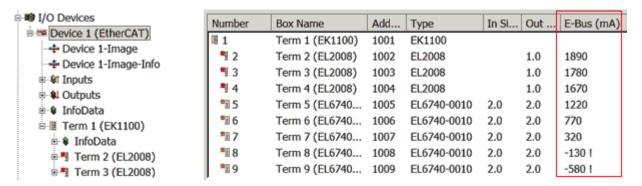


Fig. 11: System manager current calculation

#### NOTE

#### Malfunction possible!

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

### **3.3 General notes for setting the watchdog**

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

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

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

#### SM watchdog (SyncManager Watchdog)

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

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

#### PDI watchdog (Process Data Watchdog)

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

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

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

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

BECKHI

Advanced Settings		×
General     Behavior     Timeout Settings     FMMU / SM     Thit Commands     Oistributed Clock     ESC Access	Behavior Startup Checking Check Vendor Id Check Product Code Check Revision Number Check Serial Number	State Machine Auto Restore States Relnit after Communication Error Log Communication Changes Final State OP O SAFEOP in Config Mode O SAFEOP O PREOP O INIT
	Process Data         □ Use LRD/LWR instead of LRW         ✓ Include WC State Bit(s)         General         □ No AutoInc - Use 2. Address         Watchdog         □ Set Multiplier (Reg. 400h):         □ Set PDI Watchdog (Reg. 410h):         ✓ Set SM Watchdog (Reg. 420h):	Info Data         ✓ Include State         □ Include Ads Address         □ Include AoE NetId         □ Include Drive Channels         2498         1000       ms:         1000       ms:         1000       ms:         1000       ms:         0K       Cancel

Fig. 12: 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  $\rightarrow$  watchdog base time = 1 / 25 MHz \* (2498 + 2) = 0.0001 seconds = 100 µs SM watchdog = 10000  $\rightarrow$  10000 \* 100 µs = 1 second watchdog monitoring time

#### 

#### 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.

#### 

#### Damage of devices and undefined state possible!

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

### 3.4 EtherCAT State Machine

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

A distinction is made between the following states:

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

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

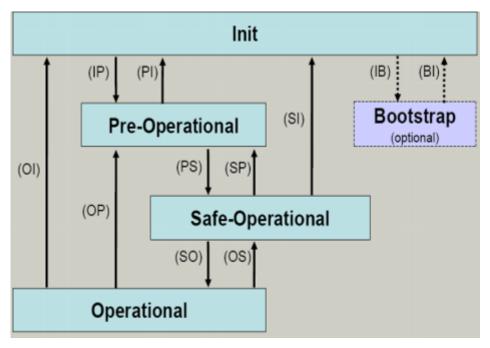


Fig. 13: States of the EtherCAT State Machine

#### 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 <u>watchdog</u> [▶ <u>16</u>] monitoring sets the outputs of the module in a safe state - depending on the settings in SAFEOP and OP - e.g. in OFF state. If this is prevented by deactivation of the watchdog monitoring in the module, the outputs can be switched or set also in the SAFEOP state.

#### Operational (Op)

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

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

#### Boot

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

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

## 3.5 CoE Interface

#### **General description**

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

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

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

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

- Index: 0x0000 ...0xFFFF (0...65535<sub>dez</sub>)
- SubIndex: 0x00...0xFF (0...255<sub>dez</sub>)

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:

neral EtherCAT Process Data Startup CoE - Online Online			
Update Lis	st 📃 🗖 Auto Update	🔽 Single	Update 🔽 Show Offline Data
Advanced.			
Add to Startu	Ip 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)
连 - 10F0:0	Backup parameter handling	RO	>1<
主 ··· 1400:0	PWM RxPDO-Par Ch.1	RO	>6<
i±… 1401:0	PWM RxPDO-Par Ch.2	RO	> 6 <
	PWM RxPDO-Par h.1 Ch.1	RO	>6<
· <b>⊡</b> 1403:0	PWM RxPDO-Par h.1 Ch.2	RO	>6<
主 ··· 1600:0	PWM RxPDO-Map Ch.1	RO	>1<

#### Fig. 14: "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.

<b>•</b>			Data	Comment
C <ps></ps>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
C <ps></ps>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
C <ps></ps>	CoE	0x1C12:01	0x1600 (5632)	download pdo 0x1C12:01 i.
C <ps></ps>	CoE	0x1C12:02	0x1601 (5633)	download pdo 0x1C12:02 i
C <ps></ps>	CoE	0x1C12:00	0x02 (2)	download pdo 0x1C12 cou

Fig. 15: Startup list in the TwinCAT System Manager

Edit...

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.

eneral 🛛 EtherCAT	Process Data Startup Co	oE - Online	Online
Update Lis	t 🗖 Auto Update	🔽 Single	Update 🔽 Show Offline Data
Advanced.			
Add to Startu	p 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	
主 1011:0	Restore default parameters	RO	>1<
🖻 – 1018:0	Identity	RO	> 4 <
1018:01	Vendor ID	RO	0x0000002 (2)
1018:02	Product code	RO	0x09C63052 (163983442)
1018:03	Revision	RO	0x00130000 (1245184)
1018:04	Serial number	RO	0x00000000 (0)
吏 10F0:0	Backup parameter handling	RO	>1<
主 ··· 1400:0	PWM RxPDO-Par Ch.1	RO	>6<
±1401:0	PWM RxPDO-Par Ch.2	RO	> 6 <
主 1402:0	PWM RxPDO-Par h.1 Ch.1	RO	>6<
连 ··· 1403:0	PWM RxPDO-Par h.1 Ch.2	RO	> 6 <
	PWM RxPDO-Map Ch.1	RO	>1<

Fig. 16: 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.

G	eneral   EtherCAT	Process Data Startup	pE - Online 🛛 (	Online
	Update Li:	st 📃 🗖 Auto Update	🔽 Single L	Jpdate 🔲 Show Offline Data
	Advanced			
	Add to Start	Jonline Data	- Mod	lule 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 <
	. <b>.</b> 1400:0	PWM RxPDO-Par Ch.1	RO	>6<

Fig. 17: 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 <u>EtherCAT system documentation</u> on the Beckhoff website.

# 3.6 Distributed Clock

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

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

For detailed information please refer to the EtherCAT system description.

# 4 Mounting and wiring

## 4.1 Instructions for ESD protection

#### NOTE

#### Destruction of the devices by electrostatic discharge possible!

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

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

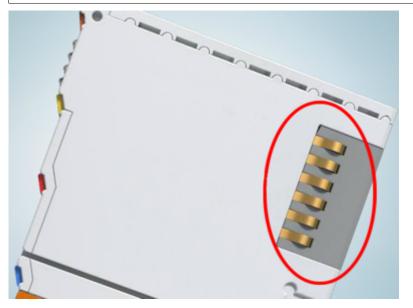


Fig. 18: Spring contacts of the Beckhoff I/O components

### 4.2 Installation on mounting rails

#### 

#### Risk of electric shock and damage of device!

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

#### Assembly

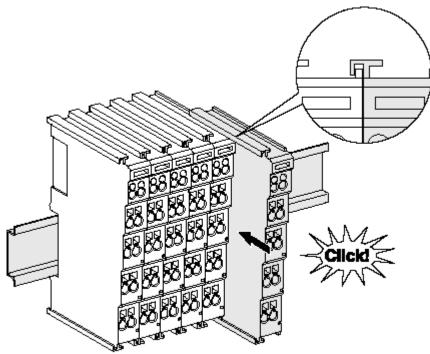


Fig. 19: Attaching on mounting rail

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

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

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

#### Fixing of mounting rails

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

#### Disassembly

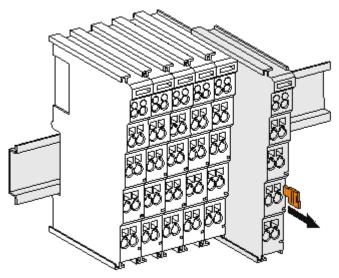


Fig. 20: Disassembling of terminal

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

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

#### Connections within a bus terminal block

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

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

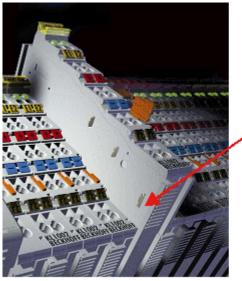


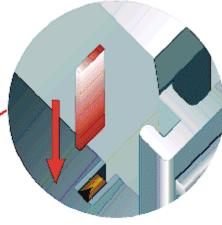
#### Power Contacts

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

#### **PE power contact**

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





#### Fig. 21: Power contact on left side

NOTE

#### Possible damage of the device

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

#### **A WARNING**

#### **Risk of electric shock!**

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

# 4.3 Installation instructions for enhanced mechanical load capacity

#### **WARNING**

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

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

#### Additional checks

The terminals have undergone the following additional tests:

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

#### Additional installation instructions

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

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

### 4.4 Connection system

#### 

#### Risk of electric shock and damage of device!

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

#### Overview

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

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

#### Standard wiring



Fig. 22: Standard wiring

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

#### Pluggable wiring



Fig. 23: Pluggable wiring

The terminals of KSxxxx and ESxxxx series feature a pluggable connection level.

The assembly and wiring procedure for the KS series is the same as for the KLxxxx and ELxxxx series. The KS/ES series terminals enable the complete wiring to be removed as a plug connector from the top of the housing for servicing.

The lower section can be removed from the terminal block by pulling the unlocking tab.

Insert the new component and plug in the connector with the wiring. This reduces the installation time and eliminates the risk of wires being mixed up.

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

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

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

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

#### High Density Terminals (HD Terminals)



Fig. 24: High Density Terminals

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

# i

#### Wiring HD Terminals

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

#### Ultrasonically "bonded" (ultrasonically welded) conductors

 Ultrasonically "bonded" conductors
 It is also possible to connect the Standard and High Density Terminals with ultrasonically "bonded" (ultrasonically welded) conductors. In this case, please note the tables concerning the wire-size width [> 33] below!

#### Wiring

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

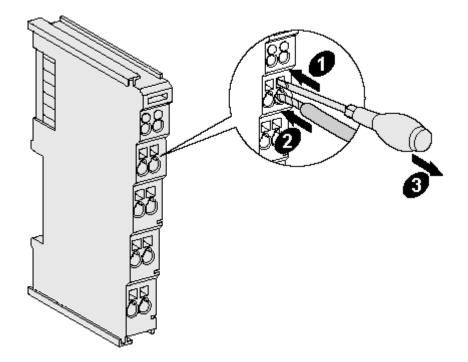


Fig. 25: Mounting a cable on a terminal connection

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

- 1. Open a spring-loaded terminal by slightly pushing with a screwdriver or a rod into the square opening above the terminal.
- 2. The wire can now be inserted into the round terminal opening without any force.
- 3. The terminal closes automatically when the pressure is released, holding the wire securely and permanently.

Terminal housing	ELxxxx, KLxxxx	ESxxxx, KSxxxx
Wire size width	0.08 2,5 mm <sup>2</sup>	0.08 2.5 mm <sup>2</sup>
Wire stripping length	8 9 mm	9 10 mm

#### High Density Terminals ELx8xx, KLx8xx (HD)

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

Terminal housing	High Density Housing
Wire size width (conductors with a wire end sleeve)	0.14 0.75 mm <sup>2</sup>
Wire size width (single core wires)	0.08 1.5 mm <sup>2</sup>
Wire size width (fine-wire conductors)	0.25 1.5 mm <sup>2</sup>
Wire size width (ultrasonically "bonded" conductors)	only 1.5 mm <sup>2</sup> (see <u>notice [▶ 32]</u> !)
Wire stripping length	8 9 mm

#### Shielding



Shielding

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

### 4.5 Installation positions

#### NOTE

#### Constraints regarding installation position and operating temperature range

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

#### **Optimum installation position (standard)**

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

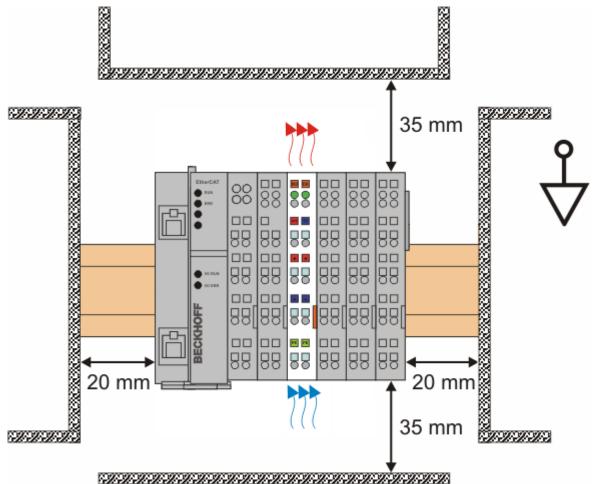


Fig. 26: Recommended distances for standard installation position

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

#### Other installation positions

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

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

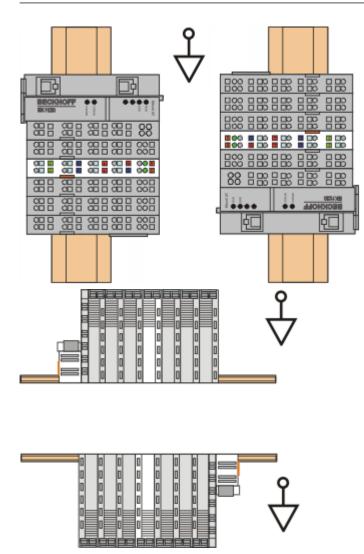


Fig. 27: Other installation positions

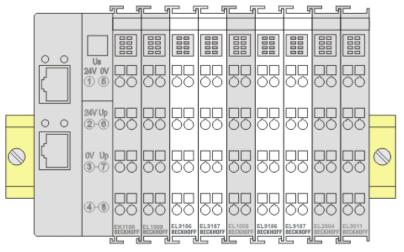
### 4.6 Positioning of passive Terminals

#### Hint for positioning of passive terminals in the bus terminal block

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

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

#### Examples for positioning of passive terminals (highlighted)



#### Fig. 28: Correct positioning

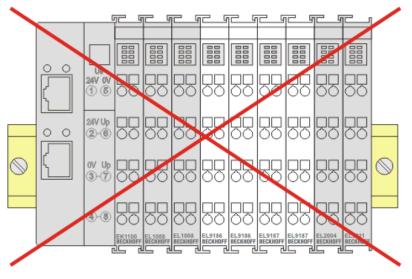


Fig. 29: Incorrect positioning

# 4.7 ATEX - Special conditions (standard temperature range)

#### **WARNING**

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

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

#### Standards

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

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

#### Marking

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

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

or



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

### 4.8 **ATEX Documentation**

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

Pay also attention to the continuative documentation

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

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

### 4.9 UL notice

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

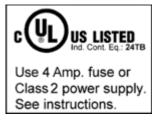
#### Basic principles

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

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



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



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

#### Application

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

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

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

### 4.10 EL1262-xxxx - LEDs and pin assignment

EL1262-0000

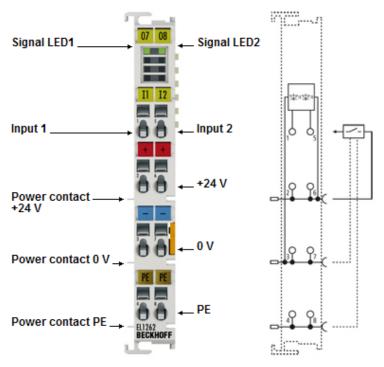


Fig. 30: EL1262-0000

#### EL1262-0000 pin assignment

Terminal point		Description	
Name	No.		
Input 1	1	Input 1	
+ 24 V	2	+24 V (internally connected to terminal point 6 and positive power contact)	
0 V	3	0 V (internally connected to terminal point 7 and negative power contact)	
PE	4	PE contact	
Input 2	5	Input 2	
+ 24 V	6	+24 V (internally connected to terminal point 2 and positive power contact)	
0 V	7	0 V (internally connected to terminal point 3 and negative power contact)	
PE	8	PE contact	

#### LEDs

LED	Color	Meaning	
INPUT 1	green	off	There is no input signal at the respective input
INPUT 2		on	+24 V input signal at the respective input

#### EL1262-0050

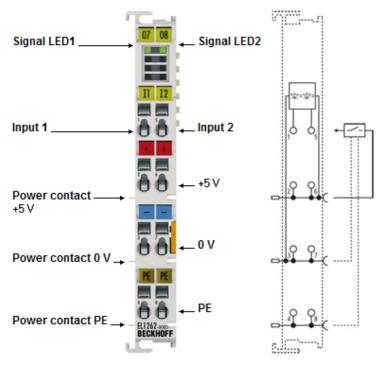


Fig. 31: EL1262-0050

#### NOTE

#### Be sure the correct supply voltage is used!

The EL1262-0050 can only be operated with a 5 V supply voltage! The terminal will not work in a terminal network with a 24 V supply voltage at the power contacts! If necessary, use a <u>EL9505</u> power supply terminal for supplying the EL1262-0050.

Terminal point		Description
Name	No.	
Input 1	1	Input 1
+ 5 V	2	+5 V (internally connected to terminal point 6 and positive power contact)
0 V	3	0 V (internally connected to terminal point 7 and negative power contact)
PE	4	PE contact
Input 2	5	Input 2
+ 5 V	6	+ 5 V (internally connected to terminal point 2 and positive power contact)
0 V	7	0 V (internally connected to terminal point 3 and negative power contact)
PE	8	PE contact

#### LEDs

LED	Color	Meaning	
INPUT 1	green	off	There is no input signal at the respective input
INPUT 2		on	+5 V input signal at the respective input

# 5 Commissioning

### 5.1 TwinCAT Quick Start

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

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

- EtherCAT Systemmanual: Fieldbus Components → EtherCAT Terminals → EtherCAT System Documentation → Setup in the TwinCAT System Manager
- **TwinCAT 2**  $\rightarrow$  TwinCAT System Manager  $\rightarrow$  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 <u>http://infosys.beckhoff.com</u>: TwinCAT 2 → TwinCAT System Manager → IO - Configuration → Adding an I/O Device
- "online": The existing hardware configuration is read
  - See also <u>http://infosys.beckhoff.com</u>:
     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:

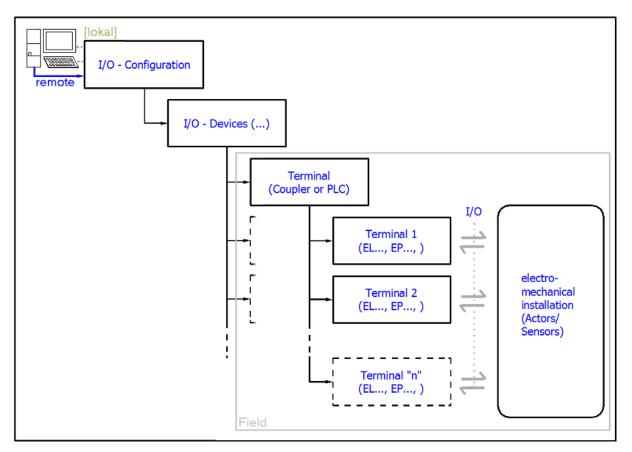


Fig. 32: Relationship between user side (commissioning) and installation

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)

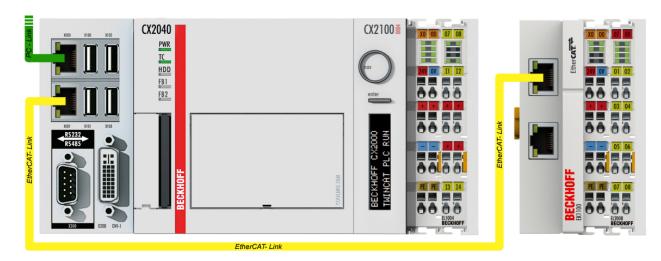


Fig. 33: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

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

### 5.1.1 TwinCAT 2

#### Startup

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

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

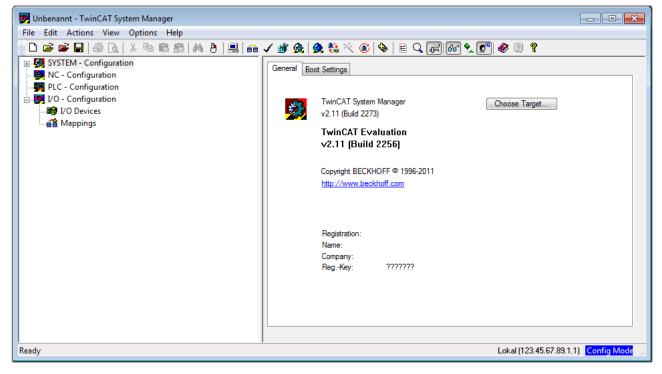


Fig. 34: Initial TwinCAT 2 user interface

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 [▶ 47]".

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" $\rightarrow$ "Choose Target System",	via the symbol "
Choose Target System	
⊕ 🚱Local (123.45.67.89.1.1)	OK Cancel
	Search (Ethernet)
	Set as Default
Connection Timeout (s): 5	

Fig. 35: 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				23
Enter Host Name / IP:			Refresh Status	Broadcast Search
Host Name C	onnected Address A	MS NetId	TwinCAT OS Ve	rsion Comment
Enter destination computer name				
& activate '	Enter Host Name	/ IP"		
L				
Route Name (Target):			Route Name (Remote):	MY-PC
AmsNetId:			Target Route	Remote Route
Transport Type:	TCP/IP 🔻		Project	None
Address Info:			<ul> <li>Static</li> <li>Temporary</li> </ul>	<ul> <li>Static</li> <li>Temporary</li> </ul>
Host Name      IP Address				
Connection Timeout (s):	5			
			Add Route	Close

Fig. 36: Specify the PLC for access by the TwinCAT System Manager: selection of the target system

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

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

#### Adding devices

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

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

🖃 🌆 SYSTEM - Configura	ation
MC - Configuration	
□	😭 I <u>m</u> port Device
Appings	Scan Devices
	Paste Ctrl+V
	Paste with Links Alt+ Ctrl+V

Fig. 37: Select "Scan Devices..."

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

4 new I/O devices found	<b>X</b>
Device 1 (EtherCAT) Device 3 (EtherCAT) [Local Area Connection (TwinCAT-Intel PCI Ethernet A] Device 2 (USB) Device 4 (NOV/DP-RAM)	OK Cancel Select All Unselect All

Fig. 38: 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 <u>sample configuration [} 44]</u> described at the beginning of this section, the result is as follows:

□
I/O Devices
Device 1 (EtherCAT)
Device 1-Image
Device 1-Image-Info
🖃 🕸 Outputs
🖽 🔒 InfoData
🖃 📕 Term 1 (EK1200)
🛨 📲 Term 2 (EL1004)
🖃 🗮 Device 3 (EtherCAT)
Device 3-Image
🗄 🛛 😂 İnputs
🗄 🛛 🜲 Outputs
🏥 💀 😫 InfoData
🗄 📲 Term 4 (EK1100)
🛓 💀 😫 InfoData
📺 📲 Term 5 (EL2008)
Term 3 (EL9011)

Fig. 39: 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:

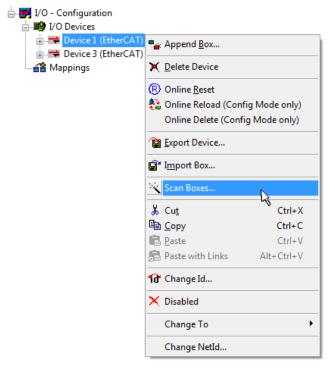


Fig. 40: Reading of individual terminals connected to a device

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	
`` <b>`` ```````````````````````````````</b>	
POUs Interior MAIN (PRG)	0001       PROGRAM MAIN         0002       VAR         0003       END_VAP         0006       0007         0008       0009         0001       •         0001       •         0001       •         0001       •         0001       •         0002       •         0003       •         0004       •         0005       •         0004       •         0005       •         •       •
POUs 📲 Data types 📮 Visualizations 🌄 Resources	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. 41: 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 Or	nline Window Help	_ 8 ×
12 <b></b>		
POUs interpretation (PRG)	0001         PROGRAM MAIN           0002         VAR           0003         nSwitchCtrl         :         BOOL:= TRUE;           0004         nRotateUpper         :         WORD :=16#8000;           0005         nRotateLower         :         WORD :=16#01;           0006         END_VAR         .         WORD :=16#01;           0007         VAR_INPUT         .         BOOL;           0008         bEL1004_Ch4         AT%I*         :         BOOL;           0009         END_VAR         .         :         BOOL;           0010         VAR_OUTPUT         .         .         :         BOOL;           0010         VAR_OUTPUT         .         .         :         BYTE;           0011         nEL2008_value         AT%Q*         :         BYTE;           0013         CAR         .         .         .	
	•         •           0001 (* Program example *)         •           0002 IF bEL1004_Ch4 THEN         •           0003 IF nSwitchCtrl THEN         •	4 • • •
	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).	- <u> </u>
POUs 📴 Data 💭 Visu 💭 Res		4
	Target: Local (123.45.67.89.1.1), Run Time: 1 TwinCAT Config Mode Lin.: 13, Col.: 7	ONLINE OV READ

Fig. 42: Sample program with variables after a compile process (without variable integration)

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

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

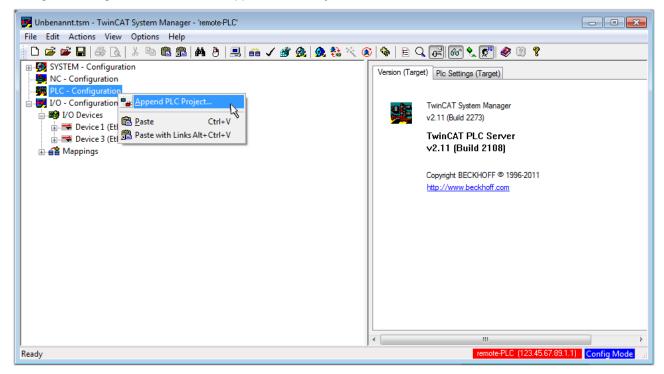


Fig. 43: Appending the TwinCAT PLC Control project

Select the PLC configuration "PLC\_example.tpy" in the browser window that opens. The project including the two variables identified with "AT" are then integrated in the configuration tree of the System Manager:

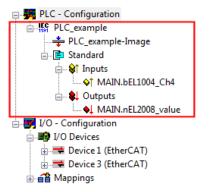


Fig. 44: PLC project integrated in the PLC configuration of the System Manager

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":

🗾 Unbenannt.tsm - TwinCAT System Ma	nager - 'remote-PLC'				- • •
File Edit Actions View Options	Help				
📄 🗅 📂 🖼 🚽 🎒 🔂 🕹 🖿 🖻	i 📾   🏘 👌   🔜   🖴 🗸 🎯 👧   👧 🗞 🎋	🔇 🌒 🖗 🛛 🖹 🔍	P 60 😒 🔊 🤌	8 🛛 📍	
🕀 🐼 SYSTEM - Configuration		Variable Flags	Online		*
- 🔀 NC - Configuration		Variable Flags	Uniine		
📄 🚔 PLC - Configuration	<u> Change Link</u>	Name:	MAIN.bEL1004_Ch4		
	Clear Link(s)		BOOL		
PLC_example-Image	Goto Link Variable	Type:	BOOL		
Standard	Take Name Over From Linked Variable	Group:	Inputs	Size:	0.1
i Inputs		Address:	0.0	User ID:	0
MAIN.bEL1004_Ch4	🚆 Insert Variable			0001101	
MAIN.nEL2008 valu	🔆 <u>D</u> elete	Linked to	]		
□	Move Address	Comment:	Variable of IEC1131 pr	niect "PLC example"	Lodated with Tas
in the configuration	Move Address	Comment.	valiable of IEC 1151 pr	oject i Lo_example .	E
Device 1 (EtherCAT)	→3 Online <u>W</u> rite				
Device 3 (EtherCAT)	→3 Online Force				
	- Release Force				
	🔍 Add To Watch				
	🕅 Remove From Watch				
	Pag Remover roll water				
		ADS Info:	Port: 801, IGrp: 0xF02	1, IOffs: 0x0, Len: 1	
					Ψ.
1		•	III		۱.
			remote-P	LC (123.45.67.89.1.1)	Config Mode

Fig. 45: 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 Term 2 (EL1004) Imput > IX 26.0, BIT [0.1] Imput > IX 26.2, BIT [0.1] Imput > IX 26.3, BIT [0.1] Imput > IX 26.3, BIT [0.1] Imput > IX 26.3, BIT [0.1] Imput - Channel 4. Term 2 Device 3 [EtherCAT] Device 3 [EtherCAT] Imput 5 [EL2008] Imput 0.1522.0, BIT [0.1]	Show Variables Unused Used and unused Exclude disabled Exclude other Devices Exclude same Image Show Tooltips 2 (EL1004) . Device 1 (EtherCAT) . I/O Device Matching Type Matching Size All Types Array Mode Offsets Continuous Show Dialog Variable Name Hand over Take over Cancel OK

Fig. 46: Selecting PDO of type BOOL

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

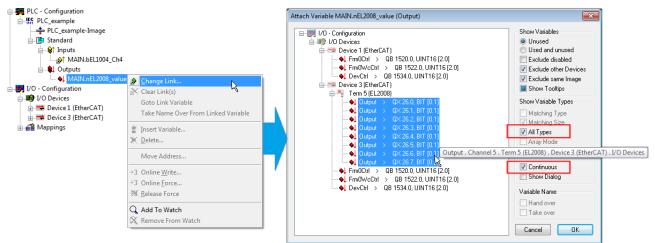


Fig. 47: Selecting several PDOs simultaneously: activate "Continuous" and "All types"

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 ( $\blacksquare$ ) 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:

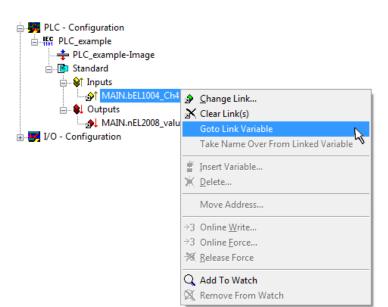


Fig. 48: Application of a "Goto Link" variable, using "MAIN.bEL1004\_Ch4" as a sample

The process of assigning variables to the PDO is completed via the menu selection "Actions"  $\rightarrow$  "Generate

Mappings", key Ctrl+M or by clicking on the symbol in the menu.

This can be visualized in the configuration:

⊡∵ <b>≦</b> ≌ Ma	ppings
- 60	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  $\checkmark$  (or via "Actions"  $\rightarrow$  "Check Configuration"). If no error is present, the configuration can be

activated via (or via "Actions"  $\rightarrow$  "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"  $\rightarrow$  "Choose Run-Time System...":

ne				
Login	F11			
Logout	F12			
Download				
Run	F5			
Stop	Shift+F8			
Reset				
Reset All		Channes Burg Times	Q.,	ĺ
Toggle Breakpoint	F9	Choose Run-Time		l
Breakpoint Dialog			(149.35.17.99.1.1)	OK
Step over	F10		ult>= (255.255.255.255.255.255) e-PLC (123.45.67.89.1.1)	Cancel
Step in	F8	III La	aufzeitsystem 1 (Port 801)	
Single Cycle	Ctrl+F5		4	
Write Values	Ctrl+F7			Version Inf
Force Values	F7			Version min
Release Force	Shift+F7			
Write/Force-Dialog	Ctrl+Shift+F7			
Show Call Stack	/			
Display Flow Control	Ctrl+F11			_
Simulation Mode		,		
Communication Parameters				
Sourcecode download	-/-			
Choose Run-Time System	4			
Create Bootproject				
Create Bootproject (offline)				
Delete Bootproject				

Fig. 49: 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"  $\rightarrow$  "Login", the F11 key or by clicking on the symbol  $\square$ . 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* File Edit Project Insert Extras Or			
	X 🗈 🖻 🙀 🙀		
	Image: State State         Image: State<		
POUs PUs Res	0001 (* Program example *)           0002 IF bEL1004_Ch4 THEN           0003 IF nSwitchCrt1 THEN           0005 nRotateLower := ROL(nRotateLower, 2);           0006 nRotateUpper := ROP(nRotateUpper, 2);           0007 nEL2008_value := WORD_TO_BYTE(nRotate           0009 ELSE           0011 IF NOT nSwitchCtrl := TRUE;           0012 END_IF           0013 END_IF           0014 0015	nSwitchCtrl = TRUE nSwitchCtrl = TRUE	nRotateLower = 16#0100
L	Target: remote-PLC (123.45.67.89.1.1), Run Time: 1	Lin.: 1, Col.: 18 UNLINE:	SIM RUN BP FORCE OV READ

Fig. 50: PLC Control logged in, ready for program startup

The PLC can now be started via "Online"  $\rightarrow$  "Run", F5 key or

### 5.1.2 TwinCAT 3

#### Startup

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

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

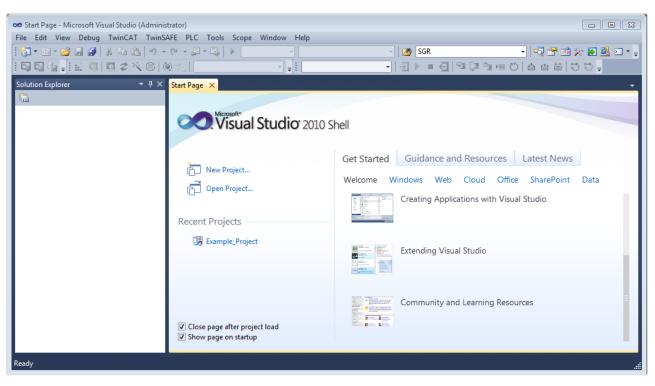


Fig. 51: Initial TwinCAT 3 user interface

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

New Project			? <mark>×</mark>
Recent Templates		.NET Framework 4   Sort by: Default	🔹 🔝 📰 Search Installed Temp 🔎
Installed Templates		TwinCAT XAE Project (XML format)	Type: TwinCAT Projects
<ul> <li>Other Project Type</li> <li>TwinCAT Measure TwinCAT Projects</li> <li>Online Templates</li> </ul>	ment		TwinCAT XAE System Manager Configuration
Name:	Example_Project		
Location:	C:\my_tc3_proje	cts\ •	Browse
Solution:	Create new solut	ion 🔹	]
Solution name:	Example_Project		Create directory for solution
			Add to Source Control
			OK Cancel

Fig. 52: Create new TwinCAT project

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

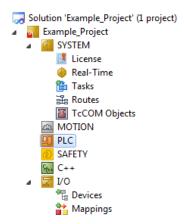


Fig. 53: New TwinCAT3 project in the project folder explorer

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 [>58]".

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:

	• •	- T				(Administrat TwinCAT	or) TwinSAFE	PLC	Tools	Scope	Window	Help	
-			-				I • 🖳   ▶				inCAT RT ()		•
: 🖓		I = 1 I	è 🖪	2	🔨 🎯	🔕 🐾 🛛	<local></local>		Ţ				•
Solut	ion Exp	lorer			<b>▼</b> ₽3	<u>&lt;</u>			Cho	se Targ	et System		

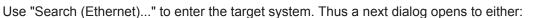
expand the pull-down menu:

<local></local>	-	
<local></local>		
Choose Target System	1	
	-	_

and open the following window:

Choose Target System	23
⊡ <mark>-</mark> <local> (123.45.67.89.1.1)</local>	ОК
	Cancel
	Search (Ethernet)
	Search (Fieldbus)
	🔲 Set as Default
Connection Timeout (s): 5	

Fig. 54: Selection dialog: Choose the target system

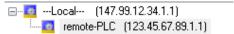


- enter the known computer name after "Enter Host Name / IP:" (as shown in red)
- perform a "Broadcast Search" (if the exact computer name is not known)
- enter the known computer IP or AmsNetID.

Add Route Dialog						23
Enter Host Name / IP:				Refresh Status		Broadcast Search
Host Name C	onnected	Address	AMS NetId	TwinCAT	OS Ve	rsion Comment
Enter destir	nation (	computer	name			
& activate '	'Enter l	Host Name	e / IP"			
	_					
Route Name (Target):				Route Name (Remo	te):	MY-PC
AmsNetId:				Target Route		Remote Route
Transport Type:	TCP/IP	•		Project		○ None
Address Info:			-	Static		Static
O Host Name O IP.	Address			Temporary		Temporary
Connection Timeout (s):	5					

Fig. 55: Specify the PLC for access by the TwinCAT System Manager: selection of the target system

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



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

#### Adding devices

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

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

in the

menu bar. The TwinCAT System Manager may first have to be set to "Config mode" via  $\overset{4}{\square}$  or via the menu "TwinCAT"  $\rightarrow$  "Restart TwinCAT (Config mode)".

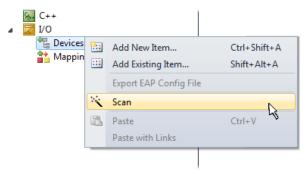


Fig. 56: Select "Scan"

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



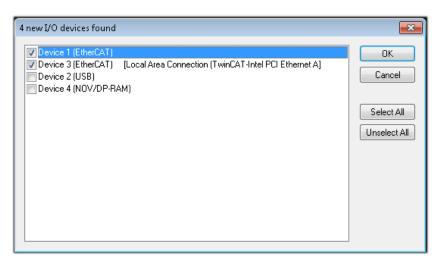


Fig. 57: Automatic detection of I/O devices: selection the devices to be integrated

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 <u>sample configuration [} 44]</u> described at the beginning of this section, the result is as follows:

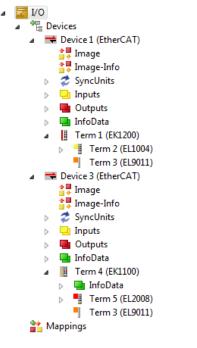


Fig. 58: Mapping of the configuration in VS shell of the TwinCAT3 environment

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:

🗧 🔁 Device 1 (EtherCAT)		Add New Item	Ctrl+Shift+A
> 📑 Device 2 (EtherCAT)		Add Existing Item	Shift+Alt+A
Mappings	ĸ	Remove	Del
		Change NetId	
		Save Device 1 (EtherCAT) As	
		Append EtherCAT Cmd	
		Append Dynamic Container	
		Online Reset	
		Online Reload	
		Online Delete	
	Ś	Scan	Ν
		Change Id	4
		Change To	
E)	à	Сору	Ctrl+C
Ж	6	Cut	Ctrl+X
	3	Paste	Ctrl+V
		Paste with Links	
E Contraction of the second		Independent Project File	

Fig. 59: 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....":

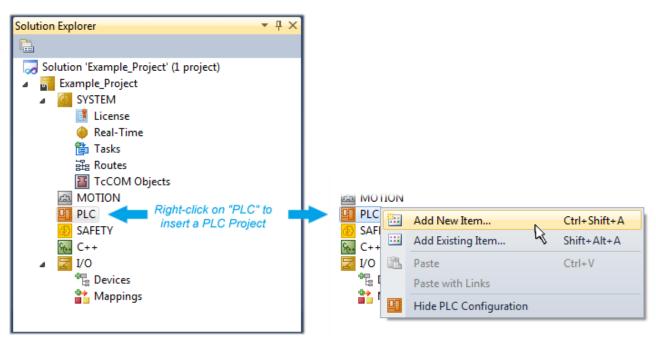


Fig. 60: Adding the programming environment in "PLC"

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 - Exampl	le_Project				ES 8
Installed Templates		Sort by:	Default		Search Installed Templates
Plc Templates Online Templates			Standard PLC Project	Plc Templates	Type: Plc Templates Creates a new TwinCAT PLC project
		1	Empty PLC Project	Plc Templates	containing a task and a program.
Name:	PLC_example				
Location:	C:\my_tc3_proje	cts\Exam	ole_Project\Example_Projec	t\ •	Browse
					Add Cancel

Fig. 61: 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 doubleclicking on "PLC\_example\_project" in "POUs". The following user interface is shown for an initial project:

Example_Project - Microsoft Visual Studio (Admin File Edit View Project Build Debug Twin)	nistrator) CAT TwinSAFE PLC Tools Scope Window Help	
		•   <del> </del>
Solution Explorer 🔹 🖣 🗙	MAIN ×	-
	1 PROGRAM MAIN	
Solution 'Example_Project' (1 project)	2 VAR 3 END VAR	
Example_Project	4	
MOTION		
PLC_example		
PLC_example Project		
External Types References		
References DUTs		
GVLs	1	
A 🗁 POUs		
MAIN (PRG)		
VISUs		
✓ Interpretation ✓		
MAIN		
PLC_example Instance		
SAFETY		
≥ <b>1</b> /O		
Ready	🖪 Ln 1 Col 1 Ch 1	INSi

Fig. 62: Initial "Main" program of the standard PLC project

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

	CAT TwinSAFE PLC Tools Scope Window Help
] • ⊡ • 🚰 🛃 🗿   X 🖬 🖺   9 • (°	
9 Fil 🖫 🚽 🔛 🧧 🗖 🌮 🕄 🎯 🚫 🕈	🖕   remote-PLC 🔹 🚽 PLC_example 🔹   日 🕨 = 日   知 🗐 🗐 管理 🗇
ution Explorer 🔹	<sup>₽</sup> × MAIN ×
Solution 'Example_Project' (1 project) Example_Project SYSTEM MOTION PLC PLC_example PLC_example Project PLC_example Project External Types External Types References DUTs GVLs POUs MAIN (PRG) VISUs PLC_example.tmc PICTask (PIcTask) MAIN PLC_example Instance PICTask Outputs MAIN.bEL1004_Ch4 PICTask Outputs MAIN.nEL2008_value	<pre>     I PROGRAM MAIN     I</pre>

Fig. 63: 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:

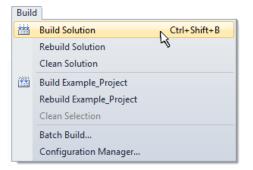
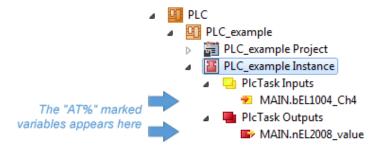


Fig. 64: Start program compilation

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:

-

<ul> <li>PLC</li> <li>PLC_example</li> <li>PLC_example Project</li> <li>PLC_example Instance</li> <li>PLC_example Instance</li> </ul>		
MAIN.bEL1004_Ch4	я	Change Link
PIcTask Outputs MAIN.nEL2008 value	X	Clear Link(s)
SAFETY		Goto Link Variable
₩ C++		Take Name Over from linked Variable
⊳ <mark>⊠</mark> I/O		Move Address
		Online Write '0'
		Online Write '1'
	<b>→3</b>	Online Write
	÷3	Online Force
	->>	Release Force
	2	Add to Watch
	×	Remove from Watch

Fig. 65: 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:

Search: Show Variables	Attach Variable MAIN.bEL1004_Ch4 (Input)	×
□       □	<pre>I/0 Devices Device 1 (EtherCAT)  Comparison Compar</pre>	<ul> <li>Unused</li> <li>Used and unused</li> <li>Exclude disabled</li> <li>Exclude other Devices</li> <li>Exclude same Image</li> <li>Show Tooltips</li> <li>Sort by Address</li> <li>Show Variable Types</li> <li>Matching Type</li> <li>Matching Size</li> <li>All Types</li> <li>Array Mode</li> <li>Offsets</li> <li>Continuous</li> <li>Show Dialog</li> <li>Variable Name</li> <li>Hand over</li> <li>Take over</li> </ul>

Fig. 66: 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:

<ul> <li>PLC</li> <li>PLC_example</li> <li>PLC_example Project</li> <li>PLC_example Instance</li> <li>PLCTask Inputs</li> <li>MAIN.bEL1004_Ch4</li> <li>PICTask Outputs</li> </ul>			Se	Attach Variable MAIN.nEL2008_value (Output) earch:	Show Variables © Urused Used and unused Exclude disabled V Exclude other Devices V Exclude same Image Show Toolips
MAIN.nEL2008_value	a.	Change Link		└─ <b>ड&gt;</b> DevCtrl > QB 1534.0, UINT [2.0] ⊡- <del>™</del> Device 3 (EtherCAT)	Sort by Address
%. C++	K	Clear Link(s)			Show Variable Types
⊳ 🚰 I/O		Take Name Over from linked Variable			Matching Type Matching Size
		Display Mode  Move Address			Array Mode
	→3	Online Write			Continuous
	→3	Online Force			Show Dialog
	->>	Release Force		Output > QX 26.7, BIT [0.1]	Variable Name
	о Я	Add to Watch Remove from Watch		~	<ul> <li>Hand over</li> <li>Take over</li> </ul>
L	_				Cancel OK

Fig. 67: 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:

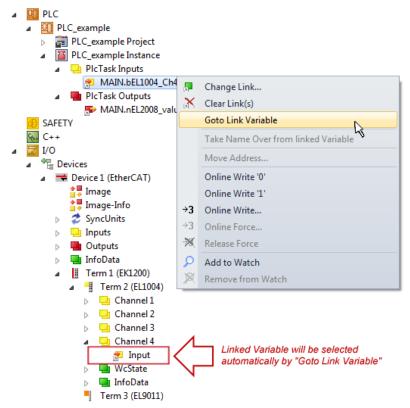


Fig. 68: Application of a "Goto Link" variable, using "MAIN.bEL1004\_Ch4" as a sample

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

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

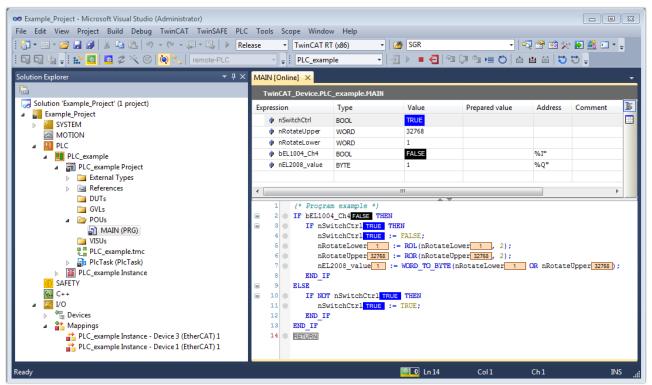


Fig. 69: TwinCAT development environment (VS shell): logged-in, after program startup

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

### 5.2 TwinCAT Development Environment

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

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

#### Details:

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

#### 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
  - <u>More...</u>

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

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

### 5.2.1 Installation of the TwinCAT real-time driver

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

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

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



 File
 Edit
 Actions
 View
 Options
 Help

 Image: I

Fig. 70: System Manager "Options" (TwinCAT 2)

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

🚥 Example_Project - Microsoft Visual Studio (	Administrator)
File Edit View Project Build Debug	TwinCAT TwinSAFE PLC Tools Scope Window Help
: 🛅 • 🖽 • 💕 🛃 🥥   🔏 🛍 🛍 🔊	Activate Configuration
i 🖸 🖓 🖕 i 🔐 🧧 🗖 🎓 🖄 🎯	Restart TwinCAT System
	Restart TwinCA
	Opulate Firmware/EEPROM
	Show Realtime Ethernet Compatible Devices
	File Handling
	EtherCAT Devices
	About TwinCAT

Fig. 71: Call up under VS Shell (TwinCAT 3)

The following dialog appears:

Installation of TwinCAT RT-Ethernet Adapters	
Ethernet Adapters	Update List
Installed and ready to use devices     LAN3 - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)	install
IOOM - TwinCAT-Intel PCI Ethernet Adapter     IG - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)	Bind
Compatible devices	Unbind
Disabled devices	Enable
	Disable
	Show Bindings

Fig. 72: Overview of network interfaces

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

A Windows warning regarding the unsigned driver can be ignored.

Alternatively an EtherCAT-device can be inserted first of all as described in chapter <u>Offline configuration</u> <u>creation, section "Creating the EtherCAT device"</u> [> 78] in order to view the compatible ethernet ports via its EtherCAT properties (tab "Adapter", button "Compatible Devices…"):

SYSTEM - Configuration   General Adapter Et	herCAT Online CoE - Online		
⊡-₩ I/O - Configuration □-₩ I/O Devices		💿 OS (NDIS) 💫 🔘 PCI	O DPRAM
⊕	Description:	1G (Intel(R) PR0/1000 PM Netw	ork Connection - Packet Sched
	Device Name:	\DEVICE\{2E55A7C2-AF68-48A	2-A9B8-7C0DE2A44BF0}
	PCI Bus/Slot:		Search
	MAC Address:	00 01 05 05 f9 54	Compatible Devices
	IP Address:	169.254.1.1 (255.255.0.0)	

Fig. 73: *EtherCAT device properties(TwinCAT 2): click on "Compatible Devices..." of tab "Adapter"* 

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

⊿	2	I/O	1
	۸	°C	Devices
		$\triangleright$	\Rightarrow Device 1 (EtherCAT)

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

🕹 1G Properties 🔹 😢 🛛
General Authentication Advanced
Connect using:
TwinCAT-Intel PCI Ethernet Adapter (
This connection uses the following items:
Client for Microsoft Networks Client for Microsoft Networks QoS Packet Scheduler TwinCAT Ethernet Protocol
I <u>n</u> stall <u>U</u> ninstall P <u>r</u> operties
Allows your computer to access resources on a Microsoft network.
<ul> <li>✓ Show icon in notification area when connected</li> <li>✓ Notify me when this connection has limited or no connectivity</li> </ul>
OK Cancel

Fig. 74: Windows properties of the network interface

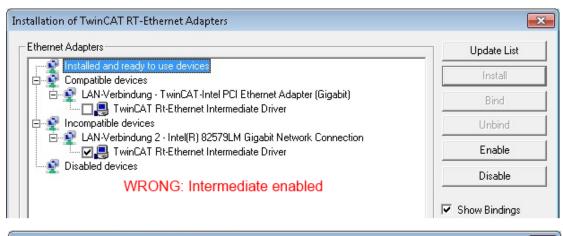
A correct setting of the driver could be:

nernet Adapters	Update List
Installed and ready to use devices □ LAN-Verbindung - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)	Install
Compatible devices	Bind
	Unbind
└──️ LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection ──♀ Disabled devices	Enable
Driver OK	Disable

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

Other possible settings have to be avoided:

nernet Adapters	Update List
Installed and ready to use devices  Installed and ready to use devices  LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection	Install
🖉 🔺 TwinCAT Ethernet Protocol for all Network Adapters	Bind
🔄 🔽 🚚 TwinCAT Rt-Ethernet Intermediate Driver	
LAN-Verbindung - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)	Unbind
▼ →- TwinCAT Ethernet Protocol for all Network Adapters 	Enable
	Disable
Disabled devices	



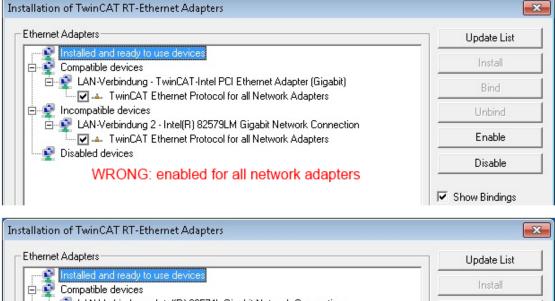




Fig. 76: Incorrect driver settings for the Ethernet port

Bind

Enable

Disable

#### 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 ( Configure
This connection uses the following items:
🗹 💂 QoS Packet Scheduler 🔗
TwinCAT Ethernet Protocol
Internet Protocol (TCP/IP)
Install Uninstall Properties
Internet Protocol (TCP/IP) Properties
General
You can get IP settings assigned automatically if your network suppo this capability. Otherwise, you need to ask your network administrator the appropriate IP settings.
You can get IP settings assigned automatically if your network suppo this capability. Otherwise, you need to ask your network administrator
You can get IP settings assigned automatically if your network suppo this capability. Otherwise, you need to ask your network administrator the appropriate IP settings.

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

### 5.2.2 Notes regarding ESI device description

### Installation of the latest ESI device description

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

The ESI files for Beckhoff EtherCAT devices are available on the <u>Beckhoff website</u>.

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 <u>TwinCAT ESI Updater</u> [▶ <u>77</u>] is available for this purpose.



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

### **Device differentiation**

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

- · family key "EL"
- name "2521"
- type "0025"
- and revision "1018"

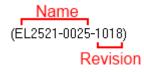


Fig. 78: Identifier structure

The order identifier consisting of name + type (here: EL2521-0010) 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 <u>further notes [ $\blacktriangleright$  7].</u>

### **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								
Apply to all	Yes No							

Fig. 79: OnlineDescription information window (TwinCAT 2)

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

TwinCAT XAE									
New device type found (EL2521-0024 - 'EL2521-0024 1K. Pulse Train 24V DC Ausgang'). ProductRevision EL2521-0024-1016									
Use available online description instead (YES) or try to load appropriate descriptions from the web									
Apply to all	Yes No Online ESI Update (Web access required)								

Fig. 80: Information window OnlineDescription (TwinCAT 3)

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

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 al- lows 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 '<u>General notes on the use of Beckhoff EtherCAT IO components</u>' and for manual configuration to the chapter '<u>Offline configuration creation'</u> [**>** 78].

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

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

### OnlineDescriptionCache00000002.xml

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

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

Add Ether	CAT device at port B (E-Bus)	of Term 1				8
Search:	el2	Name:	Term 2	Multiple:	1	ОК
Туре:	EL2002 2CH EL2004 4CH EL2004 4CH				~	Cancel Port B (E-Bus) C (Ethernet) X2 OUT'
	Extended Information	📃 Show Hidde	en Devices	📝 Show Sul	b Groups	

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

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

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

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

### OnlineDescription for TwinCAT 3.x

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

C:\User\[USERNAME]\AppData\Roaming\Beckhoff\TwinCAT3\Components\Base\EtherCATCache.xml (Please note the language settings of the OS!) You have to delete this file, too.

### Faulty ESI file

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

TwinCAT	l System Manager	Microsoft Visual Studio
<u>^</u>	Error parsing EtherCAT device description! File 'C:\TwinCAT\Io\EtherCAT\Beckhoff EL9xx.xml' Device 'EL9999' PD0 'Status Us' is assigned to a not existing Sync Manager instance (0) Description will be ignored.	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.
	ОК	ОК

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

Reasons may include:

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

### 5.2.3 TwinCAT ESI Updater

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

File Edit Actions View	Options Help
📔 D 📽 📽 日   🚳 🖪	Update EtherCAT Device Descriptions

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

The call up takes place under:

"Options" → "Update EtherCAT Device Descriptions"

### Selection under TwinCAT 3:

👓 Exampl	e_Project - Microsoft	Visual Studio (	(Administrator)						
File Edit	View Project B	uild Debug	TwinCAT Tw	inSAFE PLC Too	ls Scope Window	Hel	p		
i 🛅 = 🖻	3 - 💕 🔙 🗿 🐇	<b>₽</b> ₿ 9	Activate	Configuration			- 🦄 SGR	-   2	2 😤 🥺
	🐚 🚽 i 🔝 🚨 🗖	2 🔨 🎯	🦉 Restart T	winCAT System	Jevices			j≣ 4≣ 0   <b>4</b> 4	1 🎽 🕴
			Restart T	winco		•			
			Selected	nem		•			
			EtherCAT	l Devices		•	Update Device Descripti	ons (via ETG Website	e)
			About Tr	vinCAT			Reload Device Description	ons	45
	EtherCAT Slave	Information (E	SI) Updater					23	
	Vendor		Loaded	JRL					
	KOFF Beckhoff Au	utomation GmbH	0	http://download.beckh	off.com/download/Con	fig/Ethe	erCAT/XML_Device_Description/Be	ckhoff_EtherC	
	Target Path:	C:\TwinCAT\;	3.1\Config\Io\Eth	erCAT			ОК	Cancel	

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

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

The call up takes place under:

"TwinCAT"  $\rightarrow$  "EtherCAT Devices"  $\rightarrow$  "Update Device Description (via ETG Website)...".

### 5.2.4 Distinction between Online and Offline

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

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

In any case, during each startup the EtherCAT master checks whether the slaves it finds match the configuration. This test can be parameterised in the extended slave settings. Refer to <u>note "Installation of the latest ESI-XML device description" [ $\triangleright$ \_73].</u>

### 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 [ > 83] (Ethernet port at the IPC)
- <u>detecting the connected EtherCAT devices [> 84]</u>. This step can be carried out independent of the preceding step
- troubleshooting [ 87]

The <u>scan with existing configuration [) 88]</u> can also be carried out for comparison.

### 5.2.5 **OFFLINE** configuration creation

### Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.

File Edit Actions View Options Help	Þ	>	SYSTEM	1	Add New Item	Ctrl+Shift+A
		8	MOTION		Add Existing Item	Shift+Alt+A
SYSTEM - Configuration     MR - Configuration		>	SAFETY		Export EAP Config File	
PLC - Configuration	а.		‰• C++	22	Scan	
I/O - Configuration     I/O Devices	11	1	I/O Bevices	12	Paste	Ctrl+V
Append Device	2	Þ	Mappings		Paste with Links	

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

Select type 'EtherCAT' for an EtherCAT I/O application with EtherCAT slaves. For the present publisher/ subscriber service in combination with an EL6601/EL6614 terminal select "EtherCAT Automation Protocol via EL6601".

Insert Dev	ice
Туре:	HIO Beckhoff Lightbus     Profibus DP     Beckhoff Lightbus     Bec

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

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

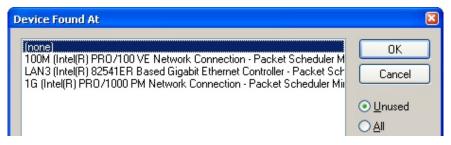


Fig. 88: Selecting the Ethernet port

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

SYSTEM - Configuration     NC - Configuration     NC - Configuration     PLC - Configuration     I/O - Configuration	General Adapter Ett	
I/O Devices		OS (NDIS) ○ PCI ○ DPRAM
Device 1 (EtherCAT)     Mappings	Description:	1G (Intel(R) PR0/1000 PM Network Connection - Packet Sched
	Device Name:	\DEVICE\{2E55A7C2-AF68-48A2-A9B8-7C0DE2A44BF0}
	PCI Bus/Slot:	Search
	MAC Address:	00 01 05 05 f9 54 Compatible Devices
	IP Address:	169.254.1.1 (255.255.0.0)
		Promiscuous Mode (use with Netmon/Wireshark only)
		Virtual Device Names
	O Adapter Referen	nce
	Adapter:	×
	Freerun Cycle (ms):	4

Fig. 89: EtherCAT device properties (TwinCAT 2)

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

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



### 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 [ $\blacktriangleright$  67].

### **Defining EtherCAT slaves**

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

🗄 🖅 I/O - Configuration	4	7	I/C	)			
🗐 🏘 I/O Devices	1	⊿		Devices			
Device 1 (EtherCAT)	<u> </u>  -		Þ	Device 1 (EtherCAT)	-	Add New Item	Ctrl+Shift+A
Mappings	<u> </u> -		Ľ.	Mappings		Add Existing Item	CLIFF AIFLA
	Ь.,				$\times$	Remove	

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

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

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

Overview of physical layer

• "Ethernet": cable-based 100BASE-TX: 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 Ether	CAT Device		×
Search:	Name: Term 1 Multiple: 1	×	ОК
Type:	<ul> <li>Beckhoff Automation GmbH &amp; Co. KG</li> <li>KTS</li> <li>EtherCAT Infrastructure components</li> <li>Ethernet Port Multiplier(CU25xx)</li> <li>Communication Terminals (EL6xxx)</li> <li>System Couplers</li> <li>EK1100 EtherCAT Coupler (2A E-Bus)</li> <li>EK1101 EtherCAT Coupler (2A E-Bus)</li> <li>EK1101 EtherCAT Coupler (2A E-Bus)</li> <li>EK1541 EtherCAT Coupler (2A E-Bus, POF, ID switch)</li> <li>EK1814 EtherCAT IO-Coupler (1A E-Bus, 4 Ch. Dig. In, 3ms, 4 Ch. Dig. Out 24V, 0.5A</li> <li>EK1818 EtherCAT IO-Coupler (1A E-Bus, 8 Ch. Dig. In, 3ms, 4 Ch. Dig. Out 24V, 0.5A</li> <li>EK1828 Otho EtherCAT IO-Coupler (1A E-Bus, 8 Ch. Dig. Out 24V, 0.5A)</li> <li>EK1828 Otho EtherCAT IO-Coupler (1A E-Bus, 8 Ch. Dig. Out 24V, 0.5A)</li> <li>EK1828 Otho EtherCAT IO-Coupler (1A E-Bus, 8 Ch. Dig. Out 24V, 0.5A)</li> <li>EK1828 Otho EtherCAT IO-Coupler (1A E-Bus, 8 Ch. Dig. Out 24V, 0.5A)</li> <li>Extended Information</li> </ul>	)	Cancel  Port A D B (Ethernet) C

Fig. 91: Selection dialog for new EtherCAT device

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

Add Ether	CAT device at port B (E-Bus)	of Term 1 (EK1100)			23
Search:	el2521	Name:	Term 2	Multiple: 1	ОК
Туре:	EL2521 1CH EL2521-002 EL2521-002 EL2521-002 EL2521-012	n GmbH & Co. KG erminals (EL2xxx) n. Pulse Train Output (EL252 4 1Ch. Pulse Train 24V DC C 5 1Ch. Pulse Train 24V DC C 4 1Ch. Pulse Train 24V DC C 11 1Ch. Pulse Train Output (	lutpuk (EL2521-0024-10 lutput negative (EL252 lutput Capture/Compare	1-0025-1021)	Cancel Port B (E-Bus) C (Ethernet) X2 OUT'
	Extended Information	🔲 Show Hidder	Devices	📝 Show Sub Groups	-

Fig. 92: Display of device revision

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

Add Ether	CAT device at port B (E-Bus) of Terr	m 1 (EK1100)				23
Search:	el2521	Name:	Term 2	Multiple:	1 🚔	ОК
Туре:	EL2521 1Ch. Pu EL2521 1Ch. Pu EL2521 1Ch. Pu EL2521 1Ch. Pu EL2521 1Ch. Pu EL2521-0024 Ch. F EL2521-0024 Ch. F	(EL2xxx) Train Output NEL2 Ilse Train Output (E Ilse Train Output (E Ilse Train Output (E Ilse Train Output (E Pulse Train Output (E Pulse Train 24V DC Ch. Pulse Train 24V	EL2521-0000-0000) EL2521-0000-1016) EL2521-0000-1017) EL2521-0000-1020) EL2521-0000-1021) Output (EL2521-0024-1 DC Output (EL2521-00 DC Output (EL2521-00	24-1016)	b Groups	Cancel Port  B (E-Bus)  C (Ethernet) X2 OUT'

Fig. 93: Display of previous revisions

### Device selection based on revision, compatibility

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

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

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

### Example:

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

# EL2521-0025-1018)

### Fig. 94: Name/revision of the terminal

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

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

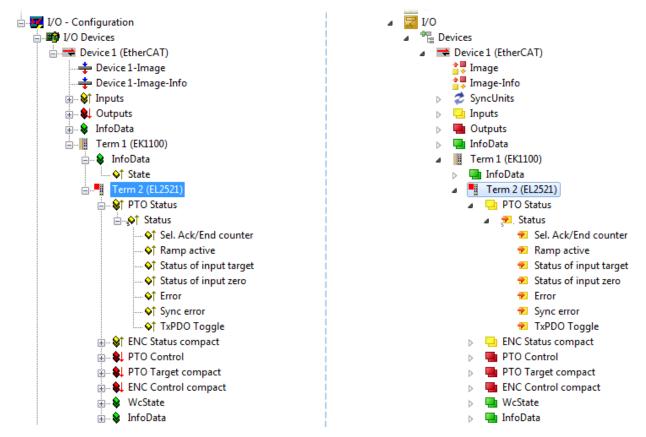


Fig. 95: EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)

### 5.2.6 **ONLINE** configuration creation

### Detecting/scanning of the EtherCAT device

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

- on TwinCAT 2 by a blue display "Config Mode" within the System Manager window: Config Mode .
- on TwinCAT 3 within the user interface of the development environment by a symbol 4.

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 ( 2) or TwinCAT 3 icon ( 2) 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.

TwinCAT 2.x Systemmanager	TwinCAT target system mode_	Τw	vinCAT	3.x GUI	_
Local (192.168.0.20.1.1) Config Mode				•(	*
	← Windows-Taskbar →	•• <b>_</b>	<b>)</b>	12:37 05.02.2015	
	TwinCAT local system mode				

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

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

👜 🐼 SYSTEM - Configuration	4	2	I/O			
→ 📴 NC - Configura → 🙀 PLC - Configura			📲 Devices		Add New Item	Ctrl+Shift+A
I/O - Configura 🔐 Import Device				:::	Add Existing Item	Shift+Alt+A
I/O Devices					Export EAP Config File	
📲 Mappings 📉 Scan Devices				***		
	i –			1	Scan	
🔁 <u>P</u> aste Ctrl+V					Paste	Ctrl+V
Paste with Links Alt+Ctrl+V					Paste with Links	

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

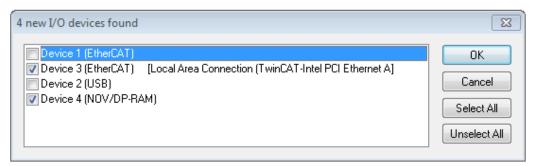
This scan mode attempts to find not only EtherCAT devices (or Ethernet ports that are usable as such), but also NOVRAM, fieldbus cards, SMB etc. However, not all devices can be found automatically.

TwinCAT System Manager	Microsoft Visual Studio
HINT: Not all types of devices can be found automatically	HINT: Not all types of devices can be found automatically
OK Cancel	OK Cancel

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



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



### Fig. 99: Detected Ethernet devices

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



### Selecting the Ethernet port

Ethernet ports can only be selected for EtherCAT devices for which the TwinCAT real-time driver is installed. This has to be done separately for each port. Please refer to the respective installation page [ $\blacktriangleright$  67].

### Detecting/Scanning the EtherCAT devices



### Online scan functionality

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



Fig. 100: Example default state

### 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 <u>comparison</u>

 $[\blacktriangleright \underline{88}]$  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:

General	EtherCAT	DC	Proces	s Data	Startup	CoE - Online	Online
Type:		EL252	1-0025 1	1Ch. Pu	lse Train 2	4V DC Output	negative
Product	/Revision:	EL252	1-0025-	1018 (09	9d93052 /	03fa0019)	

Fig. 101: Installing EthetCAT terminal with revision -1018

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

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

After some time Beckhoff extends the EL2521-0025 by a new feature C. Therefore the FW is changed, outwardly recognizable by a higher FW version and **a new revision -1019**. Nevertheless the new device naturally supports functions and interfaces of the predecessor version(s); an adaptation of 'B.tsm' or even 'B.pro' is therefore unnecessary. The series-produced machines can continue to be built with 'B.tsm' and 'B.pro'; it makes sense to perform a <u>comparative scan [> 88]</u> 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:

General	EtherCAT	DC	Proce	ss Data	a Startup	CoE - Online	
Type:		EL2521-0025 1Ch. Pulse Train 24V DC Output r					
Product	/Revision:	EL252	1-0025	1019 (	09d93052 /	03fb0019)	

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

This is usually not noticed by the commissioning engineers. TwinCAT cannot signal anything either, since 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.b 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.

TwinCAT System Manager 🛛 🕅	Microsoft Visual Studio
<b>Can for boxes</b>	Can for boxes
Ves No	Yes No

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

. ✔ I/O - Configuration ⊟. ∰ I/O Devices			4	Z I/O ₽	Devices			
Device 1 (EtherCAT)     Device 3 (EtherCAT)	∎ <mark>⊯</mark> Append <u>B</u> ox			. (	Device 1 (EtherCAT) Device 2 (EtherCAT)		Add New Item	Ctrl+Shift+A
	Montas Desire Interport poxili			**	Mappings	$\times$	Add Existing Item. Remove	Shift+Alt+A Del
-	🔨 Scan Boxes					***	Online Delete	
	00	rl+X					Change Id	
-	Change NetId					•	Disable	

Fig. 104: 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.

Scanning remote-PLC (123.45.67.89.1.1) Config Mode
--

Fig. 105: Scan progressexemplary by TwinCAT 2

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

TwinCAT System Manager 🛛 😵	Microsoft Visual Studio
Activate Free Run	2 Activate Free Run
Yes No	Yes No

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

In Config/FreeRun mode the System Manager display alternates between blue and red, and the EtherCAT device continues to operate with the idling cycle time of 4 ms (default setting), even without active task (NC, PLC).



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

🏨 🙊 💱 🔨 🚳 🖹 🔍 🖓 🚳 👷	: 🔝 🔤 🥩 🔨 🎯 🐾 🛛 <local> 🔹 🚽</local>
General EtherCA Toggle Free Run State (Ctrl-F5)	Toggle Free Run State

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

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

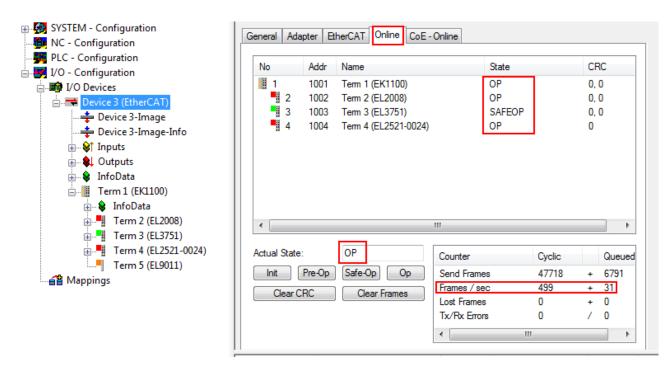


Fig. 109: Online display example

Please note:

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

The configuration is now complete. It can be modified as described under manual procedure [> 78].

### Troubleshooting

Various effects may occur during scanning.

- An unknown device is detected, i.e. an EtherCAT slave for which no ESI XML description is available. In this case the System Manager offers to read any ESI that may be stored in the device. This case is described in the chapter "Notes regarding ESI device description".
- Device are not detected properly

Possible reasons include:

- faulty data links, resulting in data loss during the scan
- slave has invalid device description

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

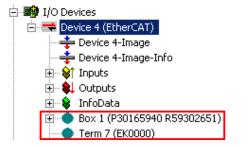


Fig. 110: Faulty identification

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

### Scan over existing Configuration

### NOTE

### Change of the configuration after comparison

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

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



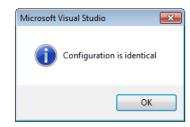


Fig. 111: Identical configuration (left: TwinCAT 2; right: TwinCAT 3)

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

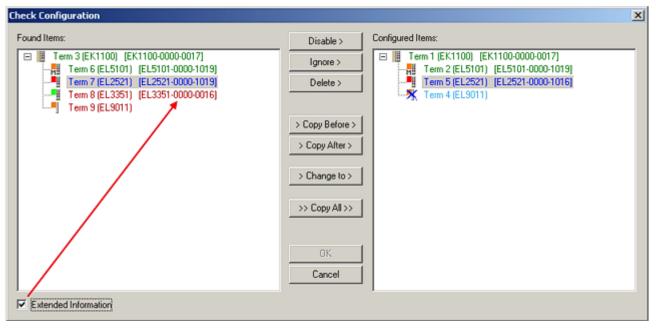


Fig. 112: Correction dialog

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.
	<ul> <li>It is present, but in a different revision, which also differs in its properties from the one specified.</li> <li>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.</li> </ul>

### 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.

# EL2521-0025-1018) (EL2521-0025-1018)

Fig. 113: Name/revision of the terminal

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

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

Found Items:	Disable >	Configured Items:
□ If 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)	Ignore > Delete > Copy Before > > Copy After > > Change to > > Copy All >> OK	□       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	OK Cancel	

Fig. 114: Correction dialog with modifications

Once all modifications have been saved or accepted, click "OK" to transfer them to the real \*.tsm configuration.

### Change to Compatible Type

TwinCAT offers a function "Change to Compatible Type..." for the exchange of a device whilst retaining the links in the task.

🗄 🔫 Device 1 (EtherCAT)	4	:	➡ Device 1 (EtherCAT) ፤		
' ⊟- <b>∦</b> Box1 (AX5101-0000-0011)	Þ	<b>■</b> ⊳	L Drive 2 (AX5101-0000-0011)	<b>.</b>	Add New Item
	I		WDT WcState	•	Uisable
WcState		$\triangleright$	🖣 🛄 InfoData		Change to Compatible Type
🗄 🗣 InfoData Change to Compatible Type					Add to HotConnect group
Add to Hot Connect Groups					Delete from HotConnect group

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

This function is preferably to be used on AX5000 devices.

### Change to Alternative Type

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

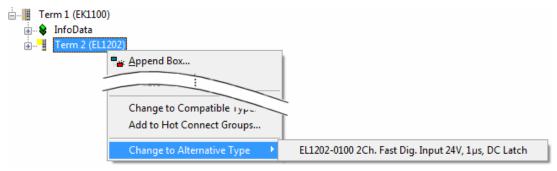


Fig. 116: TwinCAT 2 Dialog Change to Alternative Type

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

### 5.2.7 EtherCAT subscriber configuration

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

TwinCAT 2:		winCAT 3:	
🖃 📲 Term 3 (EL3751) <	4	📲 Term 3 (EL3751) 🛛 🔫	<ul> <li>doubleclick on the terminals element opens properties with several tabs</li> </ul>
🚋 🛛 😂 PAI Status		> 🄁 PAI Status	
🗄 😂 PAI Samples 1		> 📮 PAI Samples 1	*
🚋 🛛 😂 PAI Timestamp		> 🛄 PAI Timestamp	
🖶 象 WcState		> 🛄 WcState	General EtherCAT Settings DC Process Data Startup CoE - Online Diag History Online
🛓 💀 😫 InfoData		🔉 🔚 InfoData	

Fig. 117: Branch element as terminal EL3751

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

### "General" tab

Allgemein Ethe	rCAT   Prozessdaten   Startup   CoE - Online   Onlin	e
<u>N</u> ame:	Klemme 6 (EL5001)	ld: 6
Тур:	EL5001 1K. SSI Encoder	
<u>K</u> ommentar:		×
	Disabled	Symbole erzeugen 🗖

Fig. 118: "General" tab

Name	Name of the EtherCAT device
ld	Number of the EtherCAT device
Туре	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

Allgemein	EtherCAT	Prozessdaten Startup	CoE - Online
Тур:		EL5001 1K. SSI Encode	
Produkt / R	evision:	EL5001-0000-0000	
Auto-Inc-A	dresse:	FFFB	
EtherCAT-A	Adresse: 🗖	1006 🚊	Weitere Einstellungen
Vorgänger-	Port:	Klemme 5 (EL5001) - B	<b>_</b>
http://ww	vw.beckhoff.	de/german/default.htm?Eth	erCAT/EL5001.htm

### Fig. 119: "EtherCAT" tab

Type Product/Revision Auto Inc Addr.	EtherCAT device type Product and revision number of the EtherCAT device 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 <sub>hex</sub> . For each further slave the address is decremented by 1 (FFFF <sub>hex</sub> , FFFE <sub>hex</sub> 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.

Allgemein EtherCAT Prozessdaten	Startup CoE - Online Online
Sync-Manager:	PDO-Liste:
SMSizeTypeFlags0246MbxOut1246MbxIn20Outputs35Inputs	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	Lade PDO-Info aus dem Gerät Sync-Unit-Zuordnung

### Fig. 120: "Process Data" tab

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

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

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

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

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

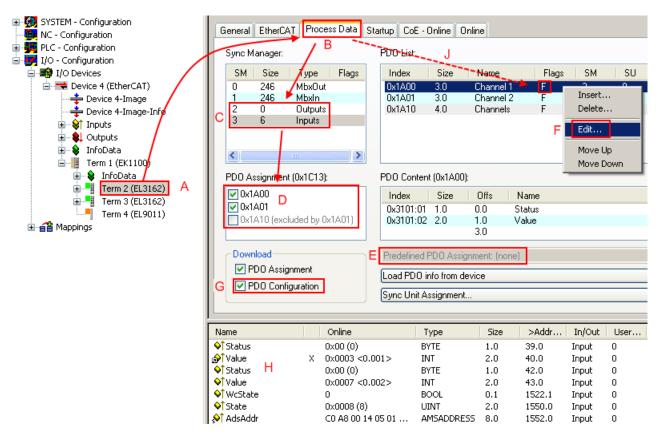


Fig. 121: Configuring the process data

### Manual modification of the process data

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

A <u>detailed description [ $\blacktriangleright$  99]</u> 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.

RECKHO

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

### Fig. 122: "Startup" tab

Column	Description
Transition	Transition to which the request is sent. This can either be
	<ul> <li>the transition from pre-operational to safe-operational (PS), or</li> </ul>
	<ul> <li>the transition from safe-operational to operational (SO).</li> </ul>
	If the transition is enclosed in "<>" (e.g. <ps>), the mailbox request is fixed and cannot be modified or deleted by the user.</ps>
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

# Move UpThis button moves the selected request up by one<br/>position in the list.Move DownThis button moves the selected request down by one<br/>position in the list.NewThis button adds a new mailbox download request to<br/>be sent during startup.DeleteThis button deletes the selected entry.EditThis 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.

gemein   EtherC	AT Prozessdaten Startu	ID CoE	- Online Online
Update L	ist 📃 🗖 Auto Upd	ate	
Advanced	d All Objects		
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
Ė∽ 1011:0	Restore default parame	BW	>1<
1011:01	Restore all	BW	0
Ė∽ 1018:0	Identity object	RO	> 4 <
1018:01		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
Ė∽ 1C00:0	SM type	RO	> 4 <
1C00:01	Subindex 001	RO	0x01 (1)
1C00:02	Subindex 002	RO	0x02 (2)
1C00:03	00:03 Subindex 003		0x03 (3)
1C00:04	Subindex 004	RO	0x04 (4)
🖻 - 1C13:0	SM 3 PDO assign (inputs)	BW	>1<
1C13:01	Subindex 001	BW	0x1A00 (6656)
<u>⊟</u> 3101:0	Inputs	RO P	>2<
3101:01	Status	RO P	0x41 (65)
3101:02	Value	RO P	0x00000000 (0)
Ė∽ 4061:0	Feature bits	BW	> 4 <
4061:01	disable frame error	BW	FALSE
4061:02	enbale power failure Bit	BW	FALSE
4061:03	enable inhibit time	BW	FALSE
4061:04	enable test mode	BW	FALSE
4066	SSI-coding	BW	Gray code (1)
4067	SSI-baudrate	BW	500 kBaud (3)
4068	SSI-frame type	BW	Multiturn 25 bit (0)
4069	SSI-frame size	BW	0x0019 (25)
406A	Data length	BW	0x0018 (24)
406B	Min. inhibit time[µs]	BW	0x0000 (0)

### Fig. 123: "CoE – Online" tab

### Object list display

Column	Description			
Index	Index	Index and sub-index of the object		
Name	Name	Name of the object		
Flags	Flags RW The object can be read, and data can be written to the object (read/write			
RO The object can be rea		The object can be read, but no data can be written to the object (read only)		
	Ρ	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.		

Advanced Settings		×
Dictionary Backup	Online - via SDO Information     All Objects     Mappable Objects (RxPDO)     Mappable Objects (TxPDO)     Backup Objects     Settings Objects	
	© Offline - via EDS File Browse OK Abbrechen	

Fig. 124: Dialog "Advanced settings"

Online - via SDO InformationIf this option button is selected, the list of the objects included in the object<br/>list of the slave is uploaded from the slave via SDO information. The list<br/>below can be used to specify which object types are to be uploaded.Offline - via EDS FileIf this option button is selected, the list of the objects included in the object<br/>list is read from an EDS file provided by the user.

### "Online" tab

Allgemein Eth	erCAT   Prozessdaten   Sta	artup CoE - Online Online				
Status-Maschine						
Init	Bootstrap	aktueller Status: OP				
Pre-Op	Safe-Op	angeforderter Status: OP				
Op	Fehler löschen	angerordener status, jor				
∟ ⊏DLL-Status		]				
Port A:	Carrier / Open					
Port B:	Carrier / Open					
Port C:	No Carrier / Closed					
Port D:	No Carrier / Open					
File access	over EtherCAT					

Fig. 125: "Online" tab



### State Machine

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

General EtherCAT Settings DC	Process Data Startup CoE - Online Diag History Online
Operation Mode:	DC-Synchron (input based)
	Advanced Settings

Fig. 126: "DC" tab (Distributed Clocks)

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

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

 $\label{eq:Fieldbus Components} \rightarrow \mbox{EtherCAT Terminals} \rightarrow \mbox{EtherCAT System documentation} \rightarrow \mbox{EtherCAT basics} \rightarrow \mbox{Distributed Clocks}$ 

### 5.2.7.1 Detailed description of Process Data tab

### Sync Manager

Lists the configuration of the Sync Manager (SM).

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

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

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

### **PDO Assignment**

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

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

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

### Activation of PDO assignment

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

a) the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see <u>Online tab [> 97]</u>),

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

### PDO Content

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

### Download

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

### **PDO Assignment**

If this check box is selected, the PDO assignment that is configured in the PDO Assignment list is downloaded to the device on startup. The required commands to be sent to the device can be viewed in the <u>Startup [94]</u> tab.

### **PDO Configuration**

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

## 5.3 General Notes - EtherCAT Slave Application

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

### Diagnosis in real time: WorkingCounter, EtherCAT State and Status

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

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

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

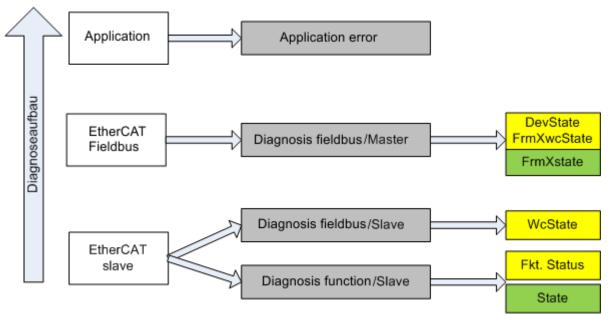


Fig. 127: Selection of the diagnostic information of an EtherCAT Slave

In general, an EtherCAT Slave offers

 communication diagnosis typical for a slave (diagnosis of successful participation in the exchange of process data, and correct operating mode) This diagnosis is the same for all slaves.

### as well as

• function diagnosis typical for a channel (device-dependent) See the corresponding device documentation

The colors in Fig. "Selection of the diagnostic information of an EtherCAT Slave" also correspond to the variable colors in the System Manager, see Fig. "Basic EtherCAT Slave Diagnosis in the PLC".

Colour	Meaning	
yellow	Input variables from the Slave to the EtherCAT Master, updated in every cycle	
red	Dutput variables from the Slave to the EtherCAT Master, updated in every cycle	
green	Information variables for the EtherCAT Master that are updated acyclically. This means that it is possible that in any particular cycle they do not represent the latest possible status. It is therefore useful to read such variables through ADS.	

Fig. *"Basic EtherCAT Slave Diagnosis in the PLC"* shows an example of an implementation of basic EtherCAT Slave Diagnosis. A Beckhoff EL3102 (2-channel analogue input terminal) is used here, as it offers both the communication diagnosis typical of a slave and the functional diagnosis that is specific to a channel. Structures are created as input variables in the PLC, each corresponding to the process image.

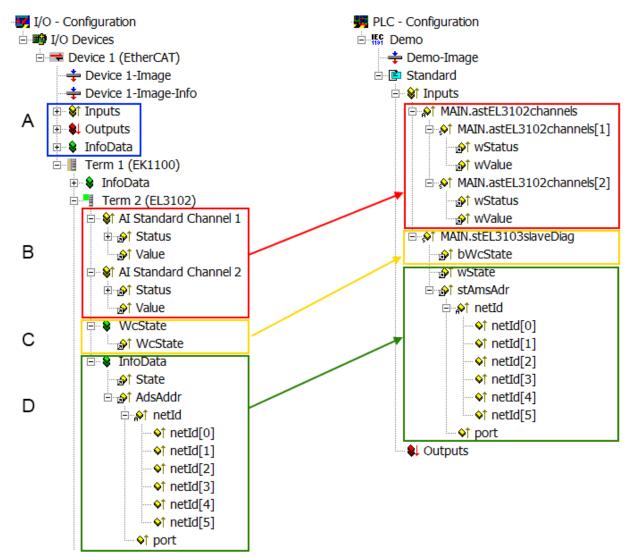


Fig. 128: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

Code	Function	Implementation	Application/evaluation
A	The EtherCAT Master's diagnostic infor- mation		At least the DevState is to be evaluated for the most recent cycle in the PLC.
	updated acyclically (yellow) or provided acyclically (green).		The EtherCAT Master's diagnostic informa- tion offers many more possibilities than are treated in the EtherCAT System Documenta- tion. A few keywords:
			CoE in the Master for communication with/through the Slaves
			Functions from <i>TcEtherCAT.lib</i>
			Perform an OnlineScan
В	In the example chosen (EL3102) the EL3102 comprises two analogue input channels that transmit a single function status for the most recent cycle.	<ul> <li>Status</li> <li>the bit significations may be found in the device documentation</li> <li>other devices may supply more information, or none that is typical of a slave</li> </ul>	In order for the higher-level PLC task (or cor- responding 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.
С	For every EtherCAT Slave that has cyclic process data, the Master displays, using what is known as a WorkingCounter, whether the slave is participating success- fully and without error in the cyclic ex- change of process data. This important, el- ementary information is therefore provided for the most recent cycle in the System Manager	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 Syn- cUnit	In order for the higher-level PLC task (or cor- responding control applications) to be able to rely on correct data, the communication sta- tus of the EtherCAT Slave must be evaluated there. Such information is therefore provided with the process data for the most recent cy- cle.
	1. at the EtherCAT Slave, and, with identical contents		
	2. as a collective variable at the EtherCAT Master (see Point A)		
	for linking.		
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	State current Status (INITOP) of the Slave. The Slave must be in OP (=8) when operating normally. <i>AdsAddr</i>	Information variables for the EtherCAT Mas- ter that are updated acyclically. This means that it is possible that in any particular cycle they do not represent the latest possible sta- tus. It is therefore possible to read such vari- ables through ADS.
	<ul> <li>is only rarely/never changed, except when the system starts up</li> <li>is itself determined acyclically (e.g. EtherCAT Status)</li> </ul>	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 corre- sponds to the AMS-NetID of the EtherCAT Master; communication with the individual Slave is possible via the <i>port</i> (= EtherCAT address).	

### 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"*:

General EtherCA	T DC Process Data St	artup CoE	Online Online
Update	List 📃 🗖 Auto Upo	late 🔽 🤇	Single Update 🔽
Advanc	ed		
Add to Sta	artup Offline Data		Module OD (Aol
Index	Name	Flags	Value
<u>.</u>	Al Inputs Ch.2	RO	> 17 <
<b>⊕</b> 6401:0	Channels	RO	>2<
Ė <sup></sup> 8000:0	Al 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
A00008	Enable user calibration	RW	FALSE
8000:0B	Enable vendor calibration	RW	TRUE

### Fig. 129: EL3102, CoE directory

### EtherCAT System Documentation

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

A few brief extracts:

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

### Commissioning aid in the TwinCAT System Manager

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

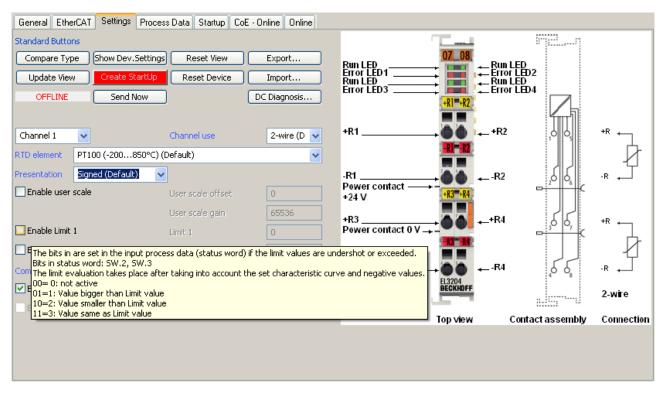


Fig. 130: Example of commissioning aid for a EL3204

This commissioning process simultaneously manages

- CoE Parameter Directory
- DC/FreeRun mode
- the available process data records (PDO)

Although the "Process Data", "DC", "Startup" and "CoE-Online" that used to be necessary for this are still displayed, it is recommended that, if the commissioning aid is used, the automatically generated settings are not changed by it.

The commissioning tool does not cover every possible application of an EL/EP device. If the available setting options are not adequate, the user can make the DC, PDO and CoE settings manually, as in the past.

### EtherCAT State: automatic default behaviour of the TwinCAT System Manager and manual operation

After the operating power is switched on, an EtherCAT Slave must go through the following statuses

- INIT
- PREOP
- SAFEOP
- OP

to ensure sound operation. The EtherCAT Master directs these statuses in accordance with the initialization routines that are defined for commissioning the device by the ES/XML and user settings (Distributed Clocks (DC), PDO, CoE). See also the section on "Principles of <u>Communication, EtherCAT State Machine [18]</u>" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

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

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

### Standard setting

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

- EtherCAT Master: OP
- Slaves: OP

This setting applies equally to all Slaves.

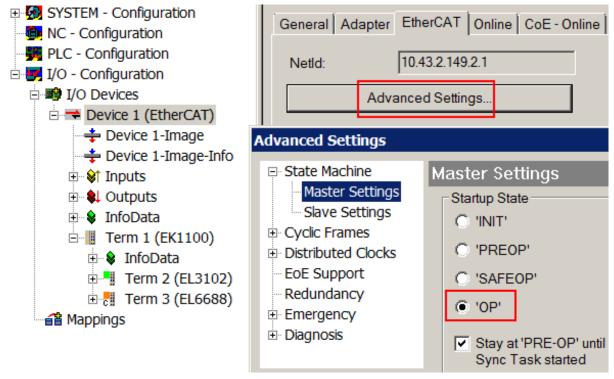


Fig. 131: Default behaviour of the System Manager

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

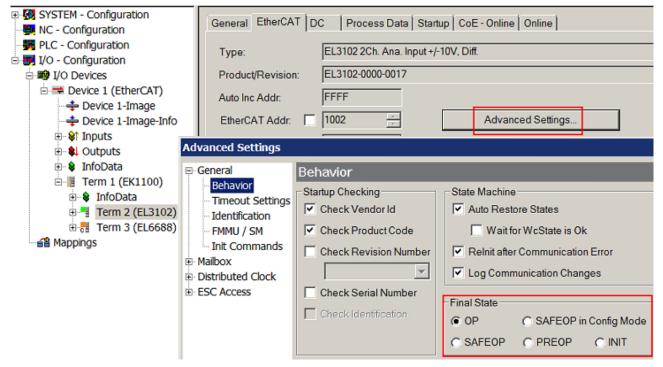


Fig. 132: Default target state in the Slave

### **Manual Control**

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

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

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

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

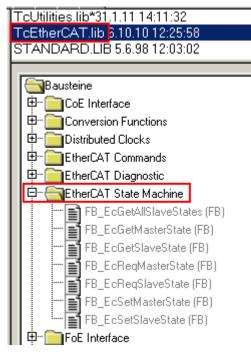


Fig. 133: PLC function blocks

### Note regarding E-Bus current

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

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

General Ada	apter EtherCAT Online	CoE - On	line			
Netld:	10.43.2.149.2.1		A	dvanced S	ettings	
	1					
Number	Box Name	Address	Туре	In Size	Out S	E-Bus (
1	Term 1 (EK1100)	1001	EK1100			
2	Term 2 (EL3102)	1002	EL3102	8.0		1830
3	Term 4 (EL2004)	1003	EL2004		0.4	1730
4	Term 5 (EL2004)	1004	EL2004		0.4	1630
<b>5</b>	Term 6 (EL7031)	1005	EL7031	8.0	8.0	1510
<b>-</b> 6	Term 7 (EL2808)	1006	EL2808		1.0	1400
1 7	Term 8 (EL3602)	1007	EL3602	12.0		1210
8	Term 9 (EL3602)	1008	EL3602	12.0		1020
9	Term 10 (EL3602)	1009	EL3602	12.0		830
10	Term 11 (EL3602)	1010	EL3602	12.0		640
11	Term 12 (EL3602)	1011	EL3602	12.0		450
12	Term 13 (EL3602)	1012	EL3602	12.0		260
13	Term 14 (EL3602)	1013	EL3602	12.0		70
c 14	Term 3 (EL6688)	1014	EL6688	22.0		-240 !

Fig. 134: Illegally exceeding the E-Bus current

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

### Message

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

Fig. 135: Warning message for exceeding E-Bus current

### NOTE

### Caution! Malfunction possible!

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

# 5.4 Oversampling terminals and TwinCAT Scope

Generally input data of a terminal could be achieved by the scope either directly (via the activated ADS server) or by creation of a PLC variable which is linked to the PDO of a terminal for recording them. Both procedures will be explained for TwinCAT 3 (TC3) at first and for TwinCAT 2 (TC2) respectively.

Oversampling means that an analog or digital input device supplies not only one measured value for each process data cycle/EtherCAT cycle (duration T), but several, which are determined at a constant interval t < T. The ratio T/t is the oversampling factor n.

A channel thus offers not only one PDO for linking in the process data, as in the example here with the EL3102, but n PDOs as in the case of the EL3702 and other oversampling terminals.

The definition of "oversampling" by the Beckhoff's point of view shouldn't be mixed up with the oversampling process of a deltaSigma ADC:

- **deltaSigma ADC**: the frequency used by the ADC to sample the analogue signal is faster than a multiple times than the frequency of the provided digital data (typically in kHz range). This is called oversampling resulting by the functional principle of this converter type and serve amongst others for anti-aliasing.
- **Beckhoff**: the device/ the terminal read of the used ADC (could be a deltaSigma ADC also) digital sample data n-times more than the PLC/ bus cycle time is set and transfers every sample to the control bundled as an oversampling PDO package.

For example these both procedures are arranged sequentially by their technical implementation within the EL3751 and can also be present simultaneously.

EL3	31	02
-----	----	----

Name	Туре	Size
🔊 Status	Status_4099	2.0
<b>♦</b> ↑ Value	INT	2.0
∮ ↑ Status	Status_4099	2.0
<b>♦</b> † Value	INT	2.0

EL3702	
--------	--

Name	Туре	Size
♦↑ Ch1 CycleCount	UINT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
�† Ch1 Value	INT	2.0
♦↑ Ch2 CycleCount	UINT	2.0
♦↑ Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0
<b>♦</b> † Ch2 Value	INT	2.0
♦↑ Ch2 Value	INT	2.0

Fig. 136: Oversampling PDO of the EL37xx series and in the comparison with EL31xx

Accordingly the Scope2 (TC2) or ScopeView (TC3) can read in and display several PDOs per cycle in correct time.

### 5.4.1 TwinCAT 3 procedure

From TwinCAT 3.1 build 4012 and using the revision as below specified in the configuration, the integrated ScopeView recognizes in its variable browser that the oversampling data is an array package and activates ForceOversampling automatically. The array as a whole must be selected using *AddSymbol* (see description in the next section). The extended PDO name provides the basis for this. Since a specific revision of the respective terminal ScopeView is able to detect the array type of a set of variables autonomous.

Terminal	Revision
EL4732	all
EL4712	all
EL3783	EL3783-0000-0017
EL3773	EL3773-0000-0019
EL3751	all
EL3742	all
EL3702	all
EL3632	all
EL2262	all
EL1262-0050	all
EL1262	all

#### Recording a PLC Variable with the TwinCAT 3 – ScopeView

By a precondition of an already created TwinCAT 3 – project and a connected PLC with an oversampling able terminal within the configuration it will be illustrated how an oversampling variable can be represented by the Scope (as a standard part of the TwinCAT 3 environment). This will be explained by means of several steps based on an example project "SCOPE\_with\_Oversampling" as a standard PLC project.

#### Step 1: Adding a project "Scope YT"

The example project "SCOPE\_with\_Oversampling" have to be added a TwinCAT Measurement – project "Scope YT project" (C) by right click (A) and selection (B) "Add"  $\rightarrow$  "New Project..". Then "Scope for OS" will be entered as name. The new project just appears within the solution explorer (D).

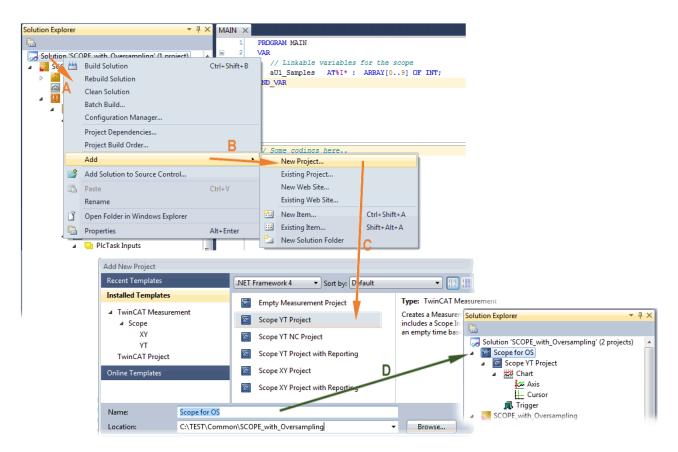


Fig. 137: Adding a Scope project into an already existing project

#### Step 2a: Creation of a PLC variable within a POU

Within the TwinCAT 3 development environment an input variable as an array with respective amount than is given by the oversampling factor have to be defined at first how it's illustrated in an example for the POU "MAIN" and an oversampling factor 10 with structured text (ST) as follows:

```
PROGRAM MAIN
VAR
aU1_Samples AT%I* : ARRAY[0..9] OF INT;
END_VAR
```

The identification "AT%I\*" stands for swapping out this array variable to link it with the process data objects (PDOs) of a terminal later. Notice that at least the number of elements have to be the same as the oversampling factor so that the indices can be set from 0 to 9 also. As soon as the compiling procedure was started and ended successful (in doing so no program code may be present) the array appears into the solution explorer of the TwinCAT 3 development environment within the section PLC under "...Instance".

The following illustration shows extracts of the solution explorer on the right. As an example that linking of an array variable to a set of oversampling process data is represented herewith:

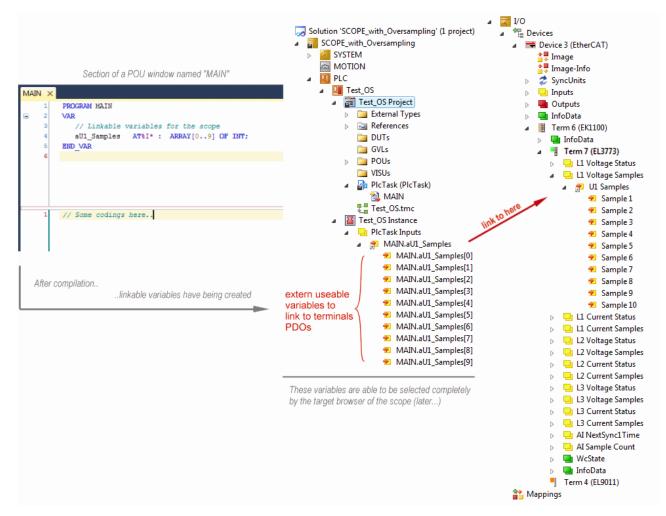


Fig. 138: Representation of a created PLC array variable ("aUI\_Samples") to link with oversampling PDOs of EL3773

#### Step 2b: Creation of a PLC variable via a free task

When a POU is not needed onto the particular system, a referenced variable could be applied via a free task also. If a free task is not existing still yet, it can be created by a right-click to "Task" of the project within SYSTEM with "Add New Item...".

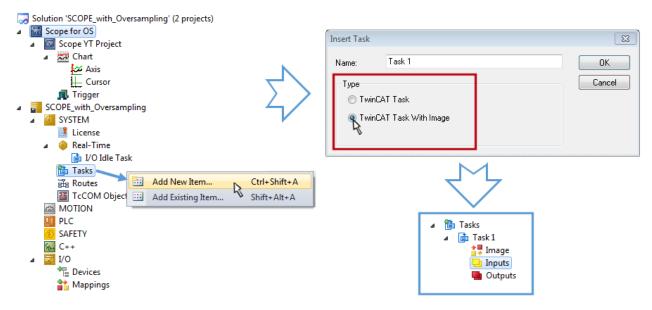


Fig. 139: Insertion of a free task

The Task have to be inserted as "TwinCAT Task With Image" and also creates an "Inputs" and "Outputs" folder therefore. The properties of the new (or as the case may be already existing) task must have activated the attribute "Create symbols" to make them selectable by the "Target Browser" of the Scope later on. The task cycle time have to be changed if so. Then, with10 x Oversampling 1 ms at 100 µs base time, resulting 10 ticks will be set by the usage of the EL3751 for example:

Task Online Paramete	er (Online)		
Name: Task 1	1	Port: 301	
Auto start		Object Id: 0x02010010	
🔲 Auto Priority Manage	ement	Options	
Priority: 1	<b>•</b>	Disable	
Cycle ticks: 10	🗧 1.000 ms	☑ Create symbols	
Start tick (modulo):	0 ≑	Include external symbols	
🔲 Separate input u	pdate		
Pre ticks:	0	Edem sync	
Warning by exceeding	ng		
Message box		✓ Floating point exceptions	
Watchdog Cycles:	0		

Fig. 140: Task property "Create symbols" must be activated

There's a default value given for the Port number (301) that should be changed, if necessary. This number have to make acquainted for the Scope, if applicable, later on. By a right click on "Inputs" that oversampling based variable can now be appended with the fitting datatype of an array. "ARRAY [0..9] OF DINT" referred to as "Var 1" in this case:

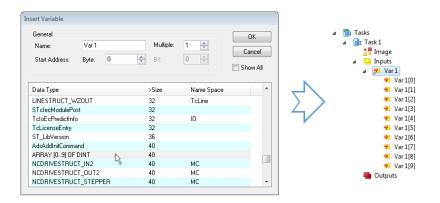


Fig. 141: Insertion of variable "Var 1" fitting to the oversampling (-factor)

#### Step 3: Linking an array variable with an oversampling PDO

By right click on "MAIN.aUI\_Samples" (according to the last preceding paragraph Step 2a) or rather "Var 1" (according to the last preceding paragraph Step 2b) within the Solution Explorer a window opens to select the process data:



⊿ 🛃 MAIN.aU1_Samples			4	💼 Task 1		
MAIN.aU1_Sam	121	Change Link		🛟 Image		
🔁 MAIN.aU1_Sam		Clear Link(s)		a 🛄 Inputs a 😥 Var 1		
🐔 MAIN.aU1_Sam		Goto Link Variable		✓ Var 1 ✓ Var 1[0]		Change Link
🔁 MAIN.aU1_Sam		Take Name Over from linked Variable		🔁 Var 1[0]	X	Clear Link(s)
MAIN.aU1_Sam		Take Name Over from linked variable		✓ Var 1[2]	۶N	
🔁 MAIN.aU1_Sam		Add New Item Ctr	Ctrl+Shi	₩ Var 1[2]		Goto Link Variable
🔁 MAIN.aU1_Sam		Move Address		✓ Var 1[5]		Take Name Over from linked Variable
🔁 MAIN.aU1_Sam		more Addressin				
🔁 MAIN.aU1_Sam	→3	Online Write		✓ Var 1[5]	100	Add New Item Ctrl+Sh
🐔 MAIN.aU1_Sam	<b>→3</b>	Online Force		🔁 Var1[6] 🔁 Var1[7]		Move Address
	₩	Release Force		₩ Var 1[8]	→3	Online Write
	0	A		🔁 Var 1[9]	<b>→3</b>	Online Force

Fig. 142: Set up the link of the PLC array variable (left: for the last preceding paragraph Step 2a, right: for the last preceding paragraph Step 2b)

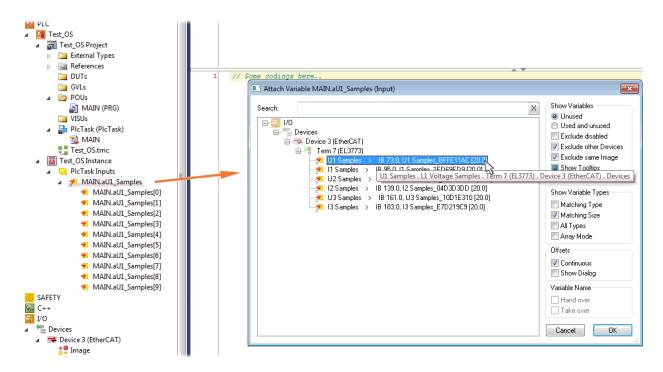


Fig. 143: Select the EL3773 PDO "L1 Voltage Samples" to create a link to the PLC array variable "aUI\_Samples"

The selection of PDO "U1 Samples" of the EL3773 for "MAIN.aUI\_Samples" based by the last preceding paragraph Step 2a as illustrated above have to be done in the same way for "Var 1" accordingly.

#### Step 4: Selection of the PLC array variable for the Y-axis of the scope

Now the configuration will be activated ( ) and logged in the PLC ( ), so the array variable will be visible for the target browser of the scope for being selected.

Thereby the drop down menu will be opened by right clicking on "Axis" (A) for selection of the scope features (B):

SCOPE_with_Oversampling - Micro	soft V	isual Studio		
File Edit View Project Build	Debug	TwinCAT F	PLC Tools Scope Window Help	
🛅 = 🖽 - 📂 🔙 🍠   🐰 🖬 🕻	8 4	) - (* - 🚚	- 🖳 🕨 Debug 🔸 TwinCAT RT (x86) 🔸	
i 🖈 🖪   🖪 🛷 🔨 🎯   🔐 🛼	An	y PC (CX2040)		
Solution Explorer		<b>▼</b> ₽ 3	Scope YT Project 🗙 MAIN [Online]	
			Chart	
Solution 'SCOPE_with_Oversam Scope for OS Scope YT Project	pling'	(2 projects)	Stat: 00:00:00 End: 00:00:00 Pos: 00:00:00 Time: 00     D0 0.00:00:10.000 № 4 ▶ ▶ □	
Chart			0,67	
Cursor A		Target Browser Cursor Window		
SCOPE_with_Oversampling SYSTEM		New Empty Ch		
MOTION	✓	Visible	Target Browser	* 🗆 ×
PLC Image: A state of the state of	9	Apply Defaults Send Project B	ROUTES Port-Name Port-Nr. Port-Type Queue-Size Fixed Server Client	Attribute
<ul> <li>Image: Test_OS Project</li> <li>Image: Test_OS Project</li> <li>Image: Test_OS Project</li> </ul>	2	Clear Error List	PIcTask 350 R0TaskPortITE 80 True True False	0000010 0000010
References		Change Ads S		0000010
DUTs 🔂 GVLs		Change Index	TC	
⊿ 🦢 POUs ज MAIN (PR	×	Copy Delete	Dei	
VISUs		Properties	Alt+Enter	

Fig. 144: Selection of the oversampling variable with the target browser

By addressing the corresponding system that represents the PLC containing the array variable ("Any PC (CX2040)" in this case) navigation up to the variable "aUI\_Samples" (C) have to be done.

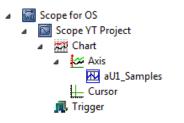
ROUTES       Name       Type       Index-Group       Index-Offset       Size       Full-Name         My-Computer       Any PC (CX2040)       Index-Offset       Size       Full-Name         PICTask (350)       Port 851 (851)       Index-Offset       Size       Add Symbol         Image: Constants.       Image: MIN.       Image: Add Subsymbols       Image: Add Subsymbols       Image: Add Subsymbols         Image: Image	Target Browser							<b>*</b> □ X
And Y (0.25) On the second secon		Name	Туре	Index-Group	Index-Offset	Size	Full-Name	
4 III	Any PC (CX2040)     PIcTask (350)     Port 851 (851)     Gonstants.     MAIN.     MAIN.	I aU1_Samples		🔒 🛛 Add Sym	ibol N	20	MAIN.aUI_Samples	

Fig. 145: Appending the variable "aUI\_Samples" below "axis" within the scope project of the solution explorer

Variable don't appears into the target browser
If "ROUTES" don't offer a possibility for selection of the provided variables, the corresponding port should be declared for the target browser:

	Ta	arge	t Bro	wsei	,						
	-	Ξà,	\$	D,		ž		畲	Ê		
"E	"Enable Server Ports"										

Using "Add symbol" displays the variable "aUI\_Samples" below "axis" within the scope project of the solution explorer directly.



Now the program start have to be done with

formally although there's no program still yet. Using "Start

Recording" Like process data value of the oversampling PDO "L1 Voltage Samples " via the linked PLC array variable can be recorded time dependent now.

As an example a sine wave input measurement value (204.5 Hz) will be illustrated below:

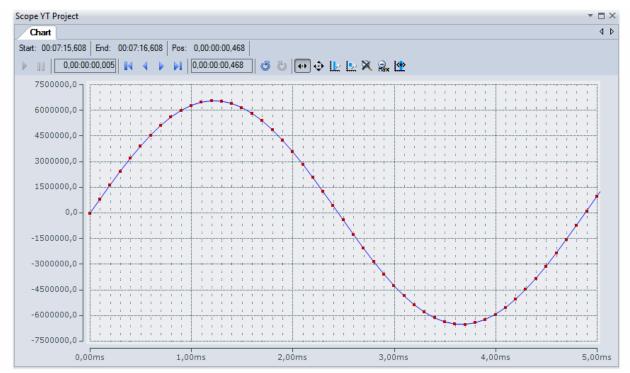


Fig. 146: Example of recording a sine signal with 10 x oversampling at 1 ms measurement cycle time

The X-axis view was fitted properly by using "Panning X" et after the recording was stopped . Following the "Chart" property "Use X-Axis SubGrid" was set to true with 10 divisions as well as the "ChannelNodeProperties" attribute "Marks" was set to "On" with the colors "Line Color" blue and "Mark Color" red. Therefore the latter indicates that 10 oversampling measurement points by the red marks.

#### Proceeding with / via ADS alternatively

In former TwinCAT 3 versions (or a lower revision as specified in the <u>table [> 109]</u> above) the oversampling PDO of the respective oversampling able terminal can be made visible for the ScopeView by activation of the ADS server.

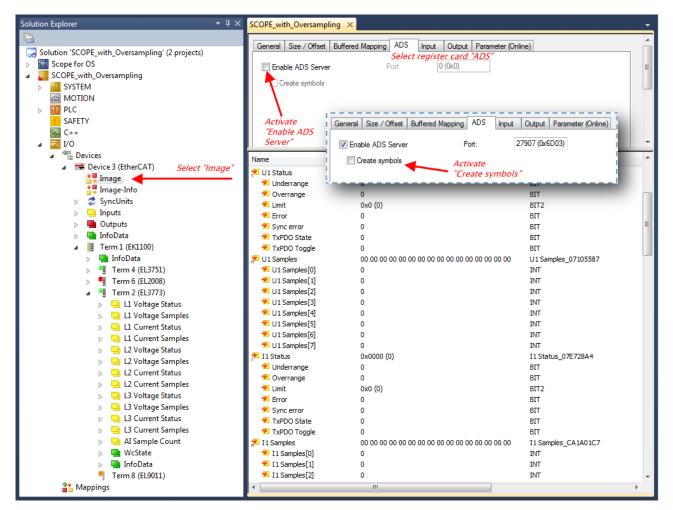


Fig. 147: Activation of the ADS server of the EtherCAT device (TwinCAT 3)

The activation of the server can be carried out by selection of "Image" within the left sided solution explorer: "I/O  $\rightarrow$  Devices  $\rightarrow$  Device .. (EtherCAT)  $\rightarrow$  Image".

Next the register card "ADS" have to be selected to activate each checkbox "Enable ADS Server" and "Create symbols" then (the port entry is done automatically).

Thereby it is possible to access process data without an embedded POU and accordingly without a linked variable:

Name BR AI Sample Count	Туре	Index	Indau			
AI Sample Count			index	Size	Full-Name	Co
	AI Sample	0xF030	0x00D0	2	Term 2 (EL3773).AI Sample Count	
BIG L1 Current Samples	L1 Current	0xF030	0x0078	16	Term 2 (EL3773).L1 Current Samples	
BIG L1 Current Status	L1 Current	0xF030	0x0076	2	Term 2 (EL3773).L1 Current Status	
BIG L1 Voltage Samples	L1 Voltage	0xF030	0x0066	16	Term 2 (EL3773).L1 Voltage Samples	
BIG L1 Voltage Status	L1 Voltage	0xF030	0x0064	2	Term 2 (EL3773).L1 Voltage Status	
BIG L2 Current Samples	L2 Current	0xF030	0x009C	16	Term 2 (EL3773).L2 Current Samples	
BIG L2 Current Status	L2 Current	0xF030	0x009A	2	Term 2 (EL3773).L2 Current Status	
BIG L2 Voltage Samples	L2 Voltage	0xF030	0x008A	16	Term 2 (EL3773).L2 Voltage Samples	
BIG L2 Voltage Status	L2 Voltage	0xF030	0x0088	2	Term 2 (EL3773).L2 Voltage Status	
BIG L3 Current Samples	L3 Current	0xF030	0x00C0	16	Term 2 (EL3773).L3 Current Samples	
BIG L3 Current Status	L3 Current	0xF030	0x00BE	2	Term 2 (EL3773).L3 Current Status	
BIG L3 Voltage Samples	L3 Voltage	0xF030	0x00AE	16	Term 2 (EL3773).L3 Voltage Samples	
BIG L3 Voltage Status	L3 Voltage	0xF030	0x00	2	Term 2 (EL3773).L3 Voltage Status	
<b>BII</b> InputToggle	BOOL	0xF031	0x2FA4	1	Term 2 (EL3773).InputToggle	
BIT) WcState	BOOL	0xF031	0x2F94	1	Term 2 (EL3773).WcState	
	<ul> <li>11 Voltage Samples</li> <li>11 Voltage Status</li> <li>12 Current Samples</li> <li>12 Current Status</li> <li>12 Voltage Samples</li> <li>12 Voltage Status</li> <li>13 Current Samples</li> <li>13 Current Status</li> <li>13 Current Status</li> <li>13 Voltage Samples</li> <li>13 Voltage Status</li> <li>13 Voltage Status</li> <li>13 Voltage Status</li> <li>11 InputToggle</li> </ul>	Image Suppose       L1 Voltage Samples       L1 Voltage         Image Status       L1 Voltage Status       L1 Voltage         Image Status       L2 Current Samples       L2 Current         Image Status       L2 Current Status       L2 Current         Image Status       L2 Voltage Samples       L2 Voltage         Image Status       L2 Voltage       L2 Voltage         Image L2 Voltage Status       L2 Voltage       L3 Current Samples         Image L3 Current Status       L3 Current       L3 Voltage Samples         Image L3 Voltage Status       L3 Voltage       L3 Voltage         Image L3 Voltage Status       L3 Voltage       BOOL         Image Voltage       BOOL       BOOL	III Voltage SamplesL1 Voltage0xF030IIII Voltage StatusL1 Voltage0xF030IIII Voltage StatusL2 Current0xF030IIII L2 Current StatusL2 Current0xF030IIII L2 Voltage SamplesL2 Voltage0xF030IIII L2 Voltage StatusL2 Voltage0xF030IIII L2 Voltage StatusL2 Voltage0xF030IIII L2 Voltage StatusL3 Current0xF030IIII L3 Voltage SamplesL3 Voltage0xF030IIII L3 Voltage StatusL3 Voltage0xF030IIII L3 Voltage StatusL3 Voltage0xF030IIII InputToggleBOOL0xF031IIII VoltateBOOL0xF031	Image Support         L1 Voltage Samples         L1 Voltage         0xF030         0x0066           Image Status         L1 Voltage         0xF030         0x0064           Image Status         L2 Voltage         0xF030         0x009C           Image Status         L2 Current         0xF030         0x009C           Image Status         L2 Current         0xF030         0x009A           Image Status         L2 Voltage         0xF030         0x008A           Image Status         L2 Voltage         0xF030         0x0088           Image Status         L2 Voltage         0xF030         0x0008           Image Status         L2 Voltage         0xF030         0x0008           Image Status         L3 Current         0xF030         0x0008           Image Status         L3 Voltage Samples         L3 Voltage         0xF030         0x0008           Image Status         L3 Voltage Status         L3 Voltage         0xF030         0x0004           Image Status         L3 Voltage         0xF030         0x00         0xF030         0x00           Image Status         L3 Voltage         0xF030         0x02FA4         0xF031         0x2FA4           Image St	III Voltage Samples       L1 Voltage       0xF030       0x0066       16         III Voltage Status       L1 Voltage       0xF030       0x0064       2         III Voltage Status       L1 Voltage       0xF030       0x0094       2         III Voltage Status       L2 Current       0xF030       0x009A       2         III Voltage Samples       L2 Current       0xF030       0x009A       2         III Voltage Samples       L2 Voltage       0xF030       0x008A       16         III Voltage Status       L2 Voltage       0xF030       0x008A       2         III Voltage Status       L2 Voltage       0xF030       0x0008       2         III Voltage Status       L3 Current       0xF030       0x0008       2         III Voltage Samples       L3 Current       0xF030       0x0008       2         III Voltage Samples       L3 Voltage       0xF030       0x0008E       2         III Voltage Status       L3 Voltage       0xF030       0x00AE       16         III Voltage Status       L3 Voltage       0xF030       0x00AE       16         III NputToggle       BOOL       0xF031       0x2FA4       1         III	III         Voltage Samples         L1         Voltage         0xF030         0x0066         16         Term 2 (EL3773).L1         Voltage Samples           III         Voltage Status         L1         Voltage         0xF030         0x0064         2         Term 2 (EL3773).L1         Voltage Samples           III         Voltage Status         L1         Voltage         0xF030         0x0064         2         Term 2 (EL3773).L1         Voltage Samples           III         Voltage         0xF030         0x0092         16         Term 2 (EL3773).L2         Current Samples           IIII         Voltage Samples         L2         Current         0xF030         0x009A         2         Term 2 (EL3773).L2         Current Samples           IIII         Voltage Samples         L2         Voltage         0xF030         0x008A         16         Term 2 (EL3773).L2         Voltage Samples           IIIII         Voltage Samples         L2         Voltage         0xF030         0x0088         2         Term 2 (EL3773).L2         Voltage Samples           IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII

Fig. 148: Direct access to PDOs of the terminal by ScopeView

BECKHOFF



#### Data type not valid

It may happen that the target browser is unable to determine the data type after insertion of the oversampling PDO (according to an array variable usually). In this case it can be changed by the channel properties:

Scope for OS	Properties	▼ 🗖 X
<ul> <li>Scope YT Project</li> <li>Kit right-click:</li> </ul>	I1 Samples TwinCAT.Measurement.ChannelNodeProp	erties 🔹
Axis then select	2. 2↓ □	
11 Samples	▲ Acquisition	
	Data-Type	INT16
🚚 Trigger	Enabled	True

#### TwinCAT 3: Activate the ADS Server of an EtherCAT device

Also see Beckhoff Information System:

🦝 TwinCAT 3
Notes on the documentation
🕞 Overview
Ticensing
📁 TExxxx   TC3 Engineering
TE1000 TC3 System
TC3 Workbench
🖀 System
PLC
¡= C/C++
🗃 Matlab/Simulink
🖀 Safety
🕞 VO
Connectivity
TwinCAT 3 Source Control
TE1111   TC3 EtherCAT Simulation
TE1120   TC3 XCAD Interface
re TE13xx   TC3 ScopeView
Foreword
Proverview
📻 Installation
📻 Configuration
im .NET API
🗃 Samples
Appendix
FAQ - frequently asked questions and answers
TwinCAT Scope Glossary
🥦 How To's
Oversampling recordings with the TwinCAT 3 Scope
Opening .svd files from a network directory

### 5.4.2 TwinCAT 2 procedure

The TwinCAT Scope2 supports the import and display of oversampling process data such as is used by oversampling-able terminals.



#### System requirements

- A TwinCAT Scope2 must be installed on the system.
- An oversampling-able terminal must be present in the configuration.

The data type of the variables is also conveyed to the TwinCAT Scope2 via the ADS data. Therefore the array variable must be created

• in the PLC, see <u>step 1a [▶ 118]</u>

• or directly in the System Manager if only one free task is present, see step 1b [ 118]

The same settings are to be made in the Scope2 for both cases, see step 2 [> 120]

#### Recording of a PLC variable with the TwinCAT 2 – Scope2

#### Step 1a: TwinCAT 2 PLC

Since the channel data are to be used in the PLC, a linkable ARRAY variable must be created there, as shown in the following example:

```
VAR
aiEL3773_Ch1_DataIn AT%I*: ARRAY[1..10] OF INT;
END_VAR
```

Fig. 149: PLC declaration

This then appears in the list in the System Manager; as a rule it can also be reached via ADS without further measures since PLC variables are always created as ADS symbols in the background.

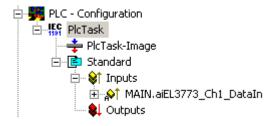


Fig. 150: PLC in the System Manager

Note: the Scope2 can only "see" such variables in the variable browser if TwinCAT and the PLC are in RUN mode.

#### Step 1b: TwinCAT 2 - free task

So that the linking works, an array variable with the channel data must be present in the system manager; i.e. each oversampling data package must be present in an array. This array variable must be defined and created manually in the System Manager.

File Edit Actions View	Options Help
🗋 D 🖻 📽 🖬 🏼 🚳 🖪	<u>L</u> anguage
E SYSTEM - Configurati	Add <u>V</u> ariable Type
NC - Configuration	Delete Variable Type

Fig. 151: Add Variable Type

An ARRAY variable of the type as known by the PLC must be created in the syntax as known from the PLC. In this example an array of 0..9 of type INT, i.e. with 10 fields.

Define Variable Ty	pe		X
Type ARRAY [09] o	fINT		OK Cancel
Alias	Base Type:	T16 🗸	Cancer
Array		Array Definition	
C Struct			
String	80		
Struct Members	Define Variable	Array	<b>x</b>
	Define Variable Array Dim.:	Array	ОК
		1	
			ОК
	Array Dim.:	1	ОК

Fig. 152: Definition of the variable type

If this variable is known to the System Manager, an instance of it can be assigned to an additional task with a right-click. It appears in the overview, sorted according to bit size.

In	sert ¥ariable						
	- General Name:	Var 152		M	fultiple:	1	÷
	Comment:						
	Start Address:	Byte:	0	÷	Bit:	0	* *
	Variable Type						
		LINESTRUCT_VARTYPE_FS0			<u>م</u>		6.0 🔺 9.0
		Array[0] of int			,	2	0.0
		U1 Samples_40 VARTYPE_FS0	EMESS/			_	0.0 3.0
		VARTYPE_FS0 LINESTRUCT	EMESS/ SERCEN	AGE_27 ICIN	,	-	1.0 🔜
			WZOUT			3	2.0

Fig. 153: Overview of declared types

In this example the variable *Var152* is created. It can now be linked with the PDO-Array of the respective channel of the terminal.

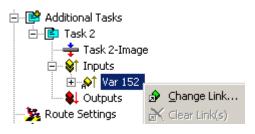


Fig. 154: Linking

If MatchingSize is activated in the dialog, the individual channels are offered directly.

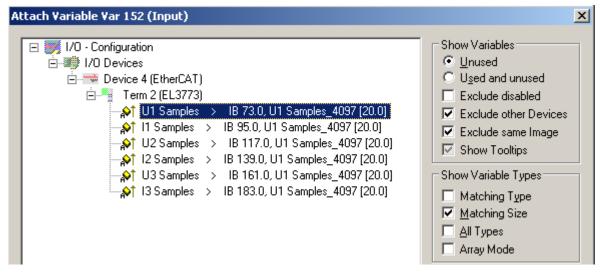


Fig. 155: Array variables of an oversampling terminal

So that the variables can also be found via ADS in the Scope2, the ADS symbols must be activated as well as the Enable Auto-Start, otherwise the task will not run automatically. ADS symbol tables are then created for all variables that have this task in their process data images.

🖻 📴 Additional Tasks	Name: Task 2	<u>P</u> ort: 301 🚍
🛶 Task 2-Image	Auto start	Options
🖻 🖓 Inputs	🔲 Auto Priority Management	🔲 <u>1</u> /0 at task begin
⊡ ∯† Var 152 •••• <b>\$↓</b> Outputs	Priority: 1	Disable
	Cycle ticks: 10 😴 10.000 r	ms Create symbols
NC - Configuration	🔲 Start tick (modulo): 🛛 🚆	Include external symbols

Fig. 156: Settings in the additional task

#### Step 2: Configuration in the Scope2

So that the linking works, an array variable with the channel data of the respective terminal must be present in the system manager; i.e. each oversampling data package must be present in an array. This array variable must be defined and created manually; see above [> 118].

You can now browse to the variable concerned in the Scope2.

Target Browser						
📇 💽 🕞 🙃 💼						
ROUTES	Name	Туре	Index-Group	Index-Offset	Size	Full-Name
🚊 🔜 601276-001	816 VAR 152	ARRAY[0] OF INT	61472	0	20	TASK 2.INPUTS.VAR 152
📄 📋 TCIO.IoTask (301)						
🖻 – 🔁 TASK 2.						
🖻 🖳 🧰 🔁 INPUTS.						
<b>BIG</b> VAR 152						
👩 TCIO.IoTask (302)						
📕 🔤 TCIO.IoTask (303)						

Fig. 157: Variable browser up to the array VAR152

The array is then not to be opened; instead the array symbol is to be selected by right-clicking on *AddSymbol*.

Name	Туре	Index-Grou
86 VAR 152_	ABBAYIO 1 OF INT	61472
É	🖀 🛛 Add Symbol	
É	😤 🛛 Add Subsymbols	-

Fig. 158: AddSymbol on the array

*ForceOversampling* and *DataType* INT16 must be set in the channel which has now been created. If necessary *SymbolBased* must be temporarily deactivated in addition.

Channel Acquisition	<del>т х</del>
VAR 152	▼
Target       Connection         Target System (NetId)       Task Sample Time         601276-001 (172:17:40:39:1.1)       Free Sample [ms] 10,000         Target Port       Use Local Server         301 : unknown       Symbol based         Symbol Connect       Index Group 0xF020	Array Settings  ArrayLenght: 10  Force Oversampling  ArrayLenght: Comment
Index Offset 0x0000 Size 2 Bytes	

Fig. 159: Channel settings

In order to check that individual oversampling values are really being logged, the *Marks* can be activated in the Scope2. Please observe the interrelationships between task cycle time, sampling time of the Scope2 channel and oversampling factor.

	4 Þ 🗙	Channel Settings 🛛 📮 🗙
	4 ک	VAR 152
		Common
		Visible
		Name VAR 152
		Comment:
· · · · · · · · · · · · · · · · · · ·		L
		🗹 Antialias Color
↓ //		Width 1 🗮
		Marks
		💿 On Color
		C Auto C Off Size 3 ∓
		O Off Size 3 🛨
		.∟ _−Modify
		Offset 0
		Scale 1
	Y Y	Bit Mask OxFFFFFFFFFFFFFFFFF
		Time Shift [µs] 0 📫
80,00ms 90,00ms	100,00ms	
53,00113	100,0000	

An additional example illustrates the following image by representation of an oversampling – variable from the EL3751 with 10 x oversampling:

Fig. 160: Activation of the marks

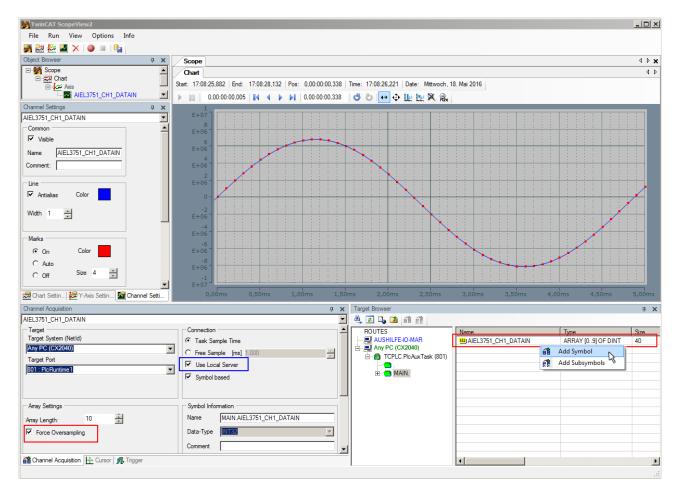


Fig. 161: Illustration of a 10 x oversampling variable of the EL3751 by the Scope2

Within the image was marked subsequently that the oversampling variable originated by the PLC was just added to the Y-axis (observe selection of the PLC-POU name "MAIN" within the "ROUTES" tree). Herewith "Force Oversampling" was activated due to the oversampling variable is not provided by the terminal.

#### Proceeding with TwinCAT 2/ alternatively via ADS

In former TwinCAT 2 versions (or a lower revision as specified in the <u>table [ $\blacktriangleright$  109]</u> above) the oversampling PDO of the respective oversampling able terminal can be made visible for the Scope2 by activation of the ADS server.

So the creation of a PLC variable can be disclaimed as well. Therefore the ADS server of the EtherCAT Device where the oversampling able terminal is connected with have to be activated.

🖐 test_scope2.tsm - TwinCAT System Manager - 'CXXfcSorte	er'				_	
File Edit Actions View Options Help						
D 📽 📽 🖬 🎒 🖪   X 🖻 🖻 🔗 🔺 8   🖳 6	ia 🗸 💣 👧 👧 💱 i	* 💽 🍫 🖹 🔍 🖓 🚳 🔧 🕵 🎯 🖇				
Image: SYSTEM - Configuration         Image: NC - Configuration	Enable ADS Server					
🗈 – 😂 L1 Voltage Status						
L1 Voltage Samples	Name	Create symbols		Size	>Address	•
E L1 Current Status	😣 U1 Status	Activate	tatus_4096	2.0	148.0	
El Current Samples El Samples	U1 Samples	"Create symbols"		20.0	150.0	
E voltage Status	♦ I1 Status	0x8000 (32768)	U1 Status_4096	2.0	170.0	
	A↑ I1 Samples	FD FF FA FF FD FF FD FF 02 00 00 00 FA FF FD FF 00 00 00 00	ARRAY [09] OF INT	20.0	172.0	
E Starter States	♦ U2 Status	0x8000 (32768)	U1 Status_4096	2.0	192.0	
E Status	♦ U2 Samples	CA 00 C4 00 CA 00 CC 00 C7 00 CC 00 C7 00 CC 00 CA 00 C4 00	ARRAY [09] OF INT	20.0	194.0	
+ & L3 Voltage Samples	♦ I2 Status	0x8000 (32768)	U1 Status_4096	2.0	214.0	
L3 Current Status	A↑ I2 Samples	F5 FF FD FF F7 FF EF FF FA FF F5 FF EF FF F2 FF FA FF F5 FF	ARRAY [09] OF INT	20.0	216.0	
	♦ U3 Status	0x8000 (32768)	U1 Status 4096	2.0	236.0	
庄 😽 AI NextSync1Time	A↑ U3 Samples	D4 00 D7 00 D7 00 DA 00 DA 00 D4 00 D7 00 D7 00 D4 00 DA 00	ARRAY [09] OF INT	20.0	238.0	
庄 😂 AI Sample Count	♦13 Status	0x8000 (32768)	U1 Status_4096	2.0	258.0	
🕀 😵 WcState	A↑ I3 Samples	08 00 FD FF FA FF 08 00 05 00 FA FF 05 00 02 00 FD FF FD FF	ARRAY [09] OF INT	20.0	260.0	
庄 😵 InfoData	♦↑ StartTimeNextLatch	DC 4C 4F 3E F6 9C 2C 07	StartTimeNextLatch 4098	8.0	280.0	
Term 5 (EL9011)	♦ Sample Count	0xF230 (62000)	UINT	2.0	288.0	
	♦ Frm0State	0x0000 (0)	UINT	2.0	1520.0	
Scope2_Test (Standard) - Device 3 (EtherCAT)	♦↑ Frm0WcState	0x0000 (0)	UINT	2.0	1522.0	
	<b>♦</b> ↑WcState	0	BOOL	0.1	1522.3	
	<b>♦</b> <sup>↑</sup> WcState	0	BOOL	0.1	1522.4	
	<b>♦</b> ↑ WcState	0	BOOL	0.1	1522.5	
	♦↑ Frm0InputToggle	0x0028 (40)	UINT	2.0	1524.0	
						•

Fig. 162: Activation of the ADS server of the EtherCAT Device (TwinCAT 2)

The activation of the ADS server have to be carried out by selection of the "Device – Image" on the left sided configuration tree:

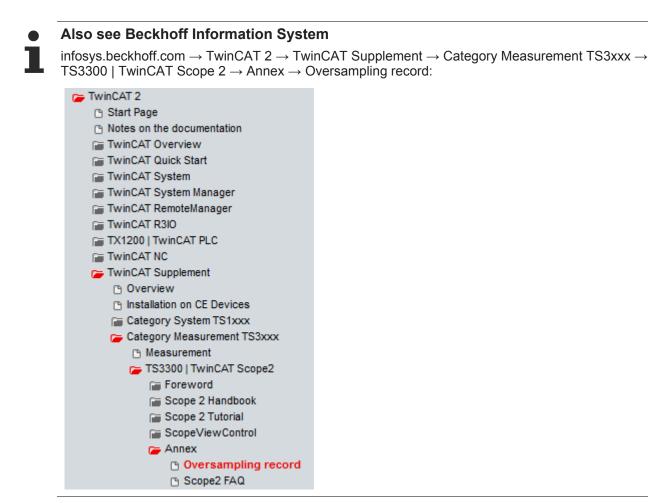
"I/O − Configuration → I/O Devices → Device .. (EtherCAT) → Device .. – Image".

Next the register card "ADS" have to be selected to activate each checkbox "Enable ADS Server" and "Create symbols" then (the port entry is done automatically).

Thus with the Scope2 process data can be accessed via the target browser without an embedded POU and without a variable reference respectively.

🛛 🔁 📫 🟥						
ROUTES	Name	Туре	Index	Index	Size	Full-Name
	<b>BIG</b> AI NEXTSYNC1TIME	AI NEXTSY	0xF030	0x0118	8	TERM 2 (EL3773).AI NEXTSYNC1TIME
Any PC (CX2040)	816 AI SAMPLE COUNT	AI SAMPLE	0xF030	0x0120	2	TERM 2 (EL3773).AI SAMPLE COUNT
AdsPort of Image 4 (27908)	8 L1 CURRENT SAMPLES	L1 CURRE	0xF030	0x00AC	20	TERM 2 (EL3773).L1 CURRENT SAMPLES
	8 L1 CURRENT STATUS	L1 CURRE	0xF030	0x00AA	2	TERM 2 (EL3773).L1 CURRENT STATUS
	<b>BIS</b> L1 VOLTAGE SAMPLES	L1 VOLTAG	0xF030	0x0096	20	TERM 2 (EL3773).L1 VOLTAGE SAMPLES
	8 L1 VOLTAGE STATUS	L1 VOLTAG	0xF030	0x0094	2	TERM 2 (EL3773).L1 VOLTAGE STATUS
	L2 CURRENT SAMPLES	L2 CURRE	0xF030	0x00D8	20	TERM 2 (EL3773).L2 CURRENT SAMPLES
	800 L2 CURRENT STATUS	L2 CURRE	0xF030	0x00D6	2	TERM 2 (EL3773).L2 CURRENT STATUS
	<b>BIG</b> L2 VOLTAGE SAMPLES	L2 VOLTAG	0xF030	0x00C2	20	TERM 2 (EL3773).L2 VOLTAGE SAMPLES
	8 L2 VOLTAGE STATUS	L2 VOLTAG	0xF030	0x00C0	2	TERM 2 (EL3773).L2 VOLTAGE STATUS
	<b>BIG</b> L3 CURRENT SAMPLES	L3 CURRE	0xF030	0x0104	20	TERM 2 (EL3773).L3 CURRENT SAMPLES
	8 L3 CURRENT STATUS	L3 CURRE	0xF030	0x0102	2	TERM 2 (EL3773).L3 CURRENT STATUS
	<b>BIG</b> L3 VOLTAGE SAMPLES	L3 VOLTAG	0xF030	0x00EE	20	TERM 2 (EL3773).L3 VOLTAGE SAMPLES
	816 L3 VOLTAGE STATUS	L3 VOLTAG	0xF030	0x00EC	2	TERM 2 (EL3773).L3 VOLTAGE STATUS
	<b>BII INPUTTOGGLE</b>	BIT	0xF031	0x2FA5	1	TERM 2 (EL3773).INPUTTOGGLE
	BIT WCSTATE	BIT	0xF031	0x2F95	1	TERM 2 (EL3773).WCSTATE

Fig. 163: Direct access of the Scope2 to the terminal's PDOs



Beckhoff TwinCAT supports the Scope2 with some oversampling devices in a special way by automatically calculating a special ADS array symbol in the background, which appears in the Scope2 in the variable browser. This can be then linked as a variable and automatically brings along the array information.

🚊 🔄 TERM 2 (EL3702).	•	Name	Туре	Index	Index	Size	Full-Name
816) CH1 CYCLECOUNT		EG CH1 SAMPLE 0[0]	CH1 SAMPLE 0_TYPE	61488	73	2	TERM 2 (EL3702).CH1 SAMPLE 0[0]
E-BIS CH1 SAMPLE 0		BIG CH1 SAMPLE 0[1]	CH1 SAMPLE 0_TYPE	61488	75	2	TERM 2 (EL3702).CH1 SAMPLE 0[1]
EIG CH1 SAMPLE O(0)		EG CH1 SAMPLE 0[2]	CH1 SAMPLE 0_TYPE	61488	77	2	TERM 2 (EL3702).CH1 SAMPLE 0[2]
BIG CH1 SAMPLE 0[1]		EIG CH1 SAMPLE 0[3]	CH1 SAMPLE 0 TYPE	61488	79	2	TERM 2 (EL3702).CH1 SAMPLE 0[3]
BIG CH1 SAMPLE 0[2]		BIG CH1 SAMPLE 0[4]	CH1 SAMPLE 0 TYPE	61488	81	2	TERM 2 (EL3702).CH1 SAMPLE 0[4]
BIG CH1 SAMPLE 0[3]		EG CH1 SAMPLE 0[5]	CH1 SAMPLE 0_TYPE	61488	83	2	TERM 2 (EL3702).CH1 SAMPLE 0[5]
		EIG CH1 SAMPLE 0[6]	CH1 SAMPLE 0 TYPE	61488	85	2	TERM 2 (EL3702).CH1 SAMPLE 0[6]
		BIG CH1 SAMPLE 0[7]	CH1 SAMPLE 0 TYPE	61488	87	2	TERM 2 (EL3702).CH1 SAMPLE 0[7]
		EIG CH1 SAMPLE 0[8]	CH1 SAMPLE 0_TYPE	61488	89	2	TERM 2 (EL3702).CH1 SAMPLE 0[8]
EIG CH1 SAMPLE 0[6]		BIG CH1 SAMPLE 0[9]	CH1 SAMPLE 0 TYPE	61488	91	2	TERM 2 (EL3702).CH1 SAMPLE 0(9)
		TS CH1 SAMPLE 0[T10]	CH1 SAMPLE 0_TYPE	61488	73	2	TERM 2 (EL3702).CH1 SAMPLE 0[T10]
CH1 SAMPLE 0[9]							
🖻 🖳 TERM 2 (EL3702).	▲	Name	Type	Index	Index	Size	Full-Name
🖻 – 🛑 TERM 2 (EL3702).	-	Name 🕈 119 CH1 VALUE	Type INT16	Index 61488	Index 73	Size 2	Full-Name TERM 2 (EL3702).CH1 SAMPLE 0(T10).CF
	<b>_</b>						
En BIS CH1 CYCLECOUNT	-						
EIII CYCLECOUNT EIIII  CYCLECOUNT EIIII CYCLE	•						
CH1 CYCLECOUNT CH1 SAMPLE 0 CH1 SAMPLE 0 CH1 SAMPLE 0[0] SMD CH1 SAMPLE 0[1]	•						
	•						
	•						
	•						

Fig. 164: Automatically calculated array variable (red) in the Scope2

es muss eine ARRAY-Variable vorliegen die über ADS erreichbar ist. Dies kann eine PLC-Variable oder eine im System Manager definierte ARRAY-Variable sein. Diese wird dann vom Scope2 erkannt.

Summary: an array variable have to be provided which is reachable via ADS. This can be a PLC variable of a POU or a defined array variable by the system manager or alternatively the ADS server of the terminals device is just activated.

### 5.5 Basic function principles

#### Table of contents

- <u>Delivery state [▶ 126]</u>
- Distributed Clock [▶ 126]
- Input characteristics [▶ 127]
- <u>Start-up behavior [▶ 128]</u>
- Process data [ 128]
- <u>Tips for operation [▶ 130]</u>

#### **Delivery state**

It is not necessary to make any particular settings when operation of the EL1262 is first started. The EL1262 operates as a normal 2-channel digital input terminal.



#### XML Device Description

If the XML description of the EL1262 is not available in your system you can download the latest XML file from the download area of the <u>Beckhoff website</u> and install it according to the installation instructions.

#### **Distributed Clock**

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

The distributed clock represents a local clock in the 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.</li>

The EL1262 only offers 32-bit support.



#### EtherCAT and Distributed Clocks

A basic <u>introduction into EtherCAT and distributed clocks</u> is available for download from the Beckhoff website.

#### Example:

The fieldbus/EtherCAT master is operated with a cycle time of 1 ms to match the higher-level PLC cycle time of 1 ms, for example. This means that every 1 ms an Ethernet frame is sent to collect the process data from the EL1262. The local terminal clock therefore triggers an interrupt in the ESC every 1 ms (1 kHz), in order to make the process data available in time for collection by the EtherCAT frame. This first interrupt is called SYNC1.

Let's assume the EL1262 is set to an oversampling rate n = 1000 in the TwinCAT System Manager. See selection dialog for EL1262 oversampling factors in the TwinCAT System Manager. This causes the ESC to generate a second interrupt in the terminal with an n-times higher frequency, in this case 1 MHz or 1  $\mu$ s period. This interrupt is called SYNC0. With each SYNC0 signal the voltage is sampled as a digital value (0/1) and the corresponding values are sequentially stored in a buffer.

Generation of the SYNC0 pulse from the local synchronized clock within the distributed clock network ensures that the input values are sampled at highly equidistant intervals with the period of the SYNC1 pulse.

The maximum oversampling factor depends on the memory size of the used ESC. In the KKYY0200 version of the EL1262 it is n = 1000.

The values accumulated in the buffer are sent as a packet to the higher-level controller. With two channels and n = 1000, 2 x 1000 = 2000 bits = 250 bytes of process data are transferred during each EtherCAT cycle.

The oversampling factor for the EL1262 can be set to predefined values between 1 and 1000.

<ul> <li>2 Ch 1000 times oversampling</li> <li>2 Channels - No oversampling</li> <li>2 Ch 2 times oversampling</li> <li>2 Ch 3 times oversampling</li> <li>2 Ch 4 times oversampling</li> <li>2 Ch 5 times oversampling</li> <li>2 Ch 10 times oversampling</li> <li>2 Ch 10 times oversampling</li> <li>2 Ch 10 times oversampling</li> <li>2 Ch 20 times oversampling</li> <li>2 Ch 40 times oversampling</li> <li>2 Ch 200 times oversampling</li> <li>2 Ch 200 times oversampling</li> <li>2 Ch 200 times oversampling</li> <li>2 Ch 400 times oversampling</li> <li>2 Ch 400 times oversampling</li> <li>2 Ch 400 times oversampling</li> </ul>		
2 Ch 2 times oversampling 2 Ch 3 times oversampling 2 Ch 4 times oversampling 2 Ch 5 times oversampling 2 Ch 8 times oversampling 2 Ch 10 times oversampling 2 Ch 10 times oversampling 2 Ch 20 times oversampling 2 Ch 40 times oversampling 2 Ch 50 times oversampling 2 Ch 100 times oversampling 2 Ch 200 times oversampling 2 Ch 200 times oversampling 2 Ch 400 times oversampling 2 Ch 400 times oversampling	2 Ch 1000 times oversamp	ling 🔄 💌
2 Ch 40 times oversampling 2 Ch 50 times oversampling 2 Ch 100 times oversampling 2 Ch 200 times oversampling 2 Ch 400 times oversampling	2 Channels - No oversampling 2 Ch 2 times oversampling 2 Ch 3 times oversampling 2 Ch 4 times oversampling 2 Ch 5 times oversampling 2 Ch 8 times oversampling 2 Ch 10 times oversampling 2 Ch 16 times oversampling	g ,
	2 Ch 40 times oversampling 2 Ch 50 times oversampling 2 Ch 100 times oversamplir 2 Ch 200 times oversamplir	) ) )g
2 Ch 1000 times oversampling	2 Ch 1000 times oversampl	

Fig. 165: *EL1262* oversampling factor selection dialog in the TwinCAT System Manager

Please note that the EL1262 process image characteristics change depending on the oversampling factor, see <u>Process data [] 128]</u> description.

#### Oversampling factor

Regarding the calculation of SYNC0 from the SYNC1 pulse based on manual specification of an oversampling factor, please note that for SYNC0 only integer values are calculated at nanosecond intervals. Example: 187,500 µs is permitted, 333.3 µs is not! Values other than those offered in the dialog are not possible. If implausible values are used, the terminal will reach the OP state, but its behavior will correspond to an oversampling factor of 1, and only the first bit will contain valid data. Example: For SYNC1/EtherCAT cycle = 1 ms oversampling factors such as 1, 2, 5 or 100 are permitted, but not 3.

#### Input characteristics

The input characteristics of the EL1262 meet the requirements of EN61131-2:2003 Type 1.

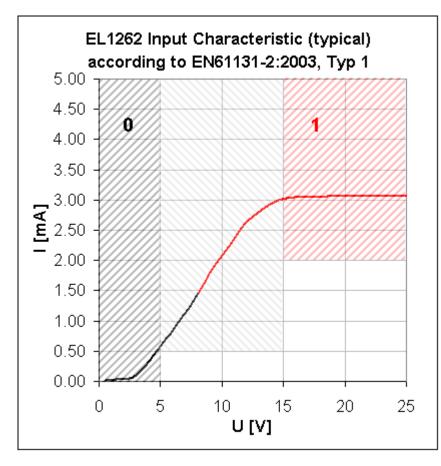


Fig. 166: Typical EL1262 input characteristics. Beckhoff reserves the right to make unannounced changes.

The input circuit of the EL12xx is optimized for fast signal changes and for the fastest possible signal acquisition. The duration required by a signal change (a rising or falling edge) to propagate from the clamping point at the front of the terminal through to the logic of the central evaluation unit (ESC) is specified for the EL12xx series as  $T_{on}/T_{off} < 1 \mu$ s, for both rising ( $T_{on}$ ) and falling edges ( $T_{off}$ ). The low absolute magnitude of this propagation time means that the temperature drift of the propagation time is also very small.

It should be borne in mind that the input circuit does not include *any* filtering. It has been optimized for the fastest possible signal transmission from the input to the evaluation unit. Fast level changes or pulses in the µs range therefore reach the evaluation unit unfiltered or unattenuated. It may be necessary to use screened cables in order to eliminate interference from the surroundings.

The sensor/signal transducer must be able to generate sufficiently steep signal edges. The power supply unit used should have sufficient buffer reserves to ensure that the signal reaches the terminal with a sufficiently steep edge in spite of capacitive or inductive cable losses.

#### Start-up behavior

From the time the EtherCAT fieldbus starts up the EL1262 takes around 60 bus cycles until it delivers process data in OP state for the first time and continuously.

#### **Process data**

The EL1262 offers a range of process data for transfer. In the default state the terminal appears in the System Manager as shown in *EL1262 default state*.

<b>Wunbenannt - TwinCAT System Manag</b> File Edit Actions View Options Help	jer						
	General     EtherCAT     D       Type:     F       Product/Revision:     F       Auto Inc Addr:     F       EtherCAT Addr:     F	EL1262 2Ch. Dig EL1262-0000-00 FFFF 1002 = Term 1 (EK1100 UINT BOOL UINT BOOL UDINT BOOL UDINT BOOL BOOL	Data Oni g. Input 24 002 0) - B <u>Size</u> 2.0 0.1 2.0 0.1 4.0 0.1 0.1 0.1	iine /, 1μs, DC Ονε Ad Ad >Address 26.0 28.0 29.0 31.0 32.0 1522.0 1522.0	In/Out Input Input Input Input Input Input Input Input Input	ttings User ID 0 0 0 0 0 0 0	
Mappings	<ul> <li>VecState2</li> <li>♦ State</li> </ul>	BOOL UINT	0.1 2.0	1522.0 1550.0	Input Input	0	
Ready				Loca	l (172.16.6	5.49.1.1)	Free Run 🥢

Fig. 167: EL1262 default state

Chx Cycle Count

EL1262 cycle counter: during each cycle the EL1262 increments this 16-bit counter by 1. The counter can be used to check the EL1262 for lost frames or data repetitions. The cycle counters for both channels show the same value.

Chx Input 0

Depending on the selected oversampling factor the digital input values are listed here, from 1 bit to 125 bytes per channel.

• Gap

This variable is only used as a placeholder and does not represent a usable process data

NextSync1Time

As mentioned above the SYNC1 interrupt triggers provision of the accumulated process data in the EL1262 in synchrony with the fieldbus. The time of the SYNC1 interrupt is the same as the first SYNC0 interrupt, which determines sampling of the inputs. The NextSync1Time value transferred by the EL1262 during an EtherCAT cycle is the start value for the *next* SYNC1 interrupt with a resolution of 32 bit (see <u>Distributed Clocks [> 126]</u>). The NextSync1Time process data can be deactivated in the ProcessData tab. NextSync1Time can be used to specify the read time for each individual sample within the distributed clock accuracy.



#### Chx Input presentation

The Chx Input process data must cover a large range of values from 1 to 1000 bits. In order to maintain a clear display of the configuration tree and the task variable links, the Chx Input variables are shown either as bit or byte. Oversampling factor <= 100: individual bits are displayed. Oversampling factor > 100: bits are consolidated as bytes. The task receiving the EL1262 process data therefore has to offer bit or byte arrays as appropriate.

#### Tips for operation

#### Distributed Clocks settings

In the EL1262 Distributed Clock options under Advanced Settings the time of the SYNC1 interrupt can be shifted forward slightly, see *EL1262 Advanced Settings, Distributed Clock*. By activating the "Based on Input Reference" checkbox the SYNC1 interrupt is shifted forward by a few µs. For further information please refer to the <u>Distributed Clock system description [▶ 126]</u>.

Advanced Settings				×
Advanced Settings	Distributed Clock Cyclic Mode Operation Mode:  ✓ Enable SYNC 0 Cycle Time (µs):  ✓ Sync Unit Cycle 20 ✓ User Defined 20 ✓ Enable SYNC 0 SYNC 1 ✓ Sync Unit Cycle x1 ✓ SYNC 0 Cycle ✓ 5 Uk of Wood	F Based on Input f + = Cycle Time (μs):	ampling	×
	Enable SYNC 1		Abbrechen	

Fig. 168: EL1262 Advanced Settings, Distributed Clocks

#### Linking large variables

The option "Change Multi Link" can be used to link larger memory areas with continuous variables. Proceed as follows:

General EtherCAT	DC F	Process Dat	a Online		
Operation Mode:	2 Ch 200	times overs	ampling	•	
Advanced Settings					
	^	savancea o	retarigs		
Name	Туре	Size	>Address	In/Out	
♦↑Ch1 CycleCount	UINT	2.0	26.0	Input	
📢 Chi Input	BYTE	1.0	28.0	Input	
👏 Chi Input	BYTE	1.0	29.0	Input	
👏 Chi Input	BYTE	1.0	30.0	Input	
👏 Chi Input	BYTE	1.0	31.0	Input	
👏 Chi Input	BYTE	1.0	32.0	Input	
👏 Chi Input	BYTE	1.0	33.0	Input	
♦îCh1 Input	BYTE	1.0	34.0	Input	
♦îCh1 Input	BYTE	1.0	35.0	Input	
👏 Chi Input	BYTE	1.0	36.0	Input	
♦îCh1 Input	BYTE	1.0	37.0	Input	
♦îCh1 Input	BYTE	1.0	38.0	Input	
👏 Chi Input	BYTE	1.0	39.0	Input	
👏 Chi Input	BYTE	1.0	40.0	Input	
👏 Chi Input	BYTE	1.0	41.0	Input	
👏 Chi Input	BYTE	1.0	42.0	Input	
👏 Chi Input	BYTE	1.0	43.0	Input	
♦îCh1 Input	BYTE	1.0	44.0	Input	
♦îCh1 Input	BYTE	1.0	45.0	Input	
♦îCh1 Input	BYTE	1.0	46.0	Input	
👏 Chi Input	BYTE	1.0	47.0	Input	
♦î Ch1 Input	BYTE	1.0	48.0	Input	
♦î Ch1 Input	BYTE	1.0	49.0	Input	
👏 Chi Input	BYTE	1.0	50.0	Input	
♦îCh1 Input	BYTE	1.0	51.0	Input	
♦î Ch1 Input	BYTE	1.0	52.0	Input	
♦↑ Ch2 CycleCount	UINT	2.0	53.0	Input	
📢 Ch2 Input	BYTE	1.0	55.0	Input	

Fig. 169: Select the variables in the terminal with the mouse

General EtherCAT	DC F	Process Data	a Online					
Operation Mode: 2 Ch 200 times oversampling								
	,							
	Advanced Settings							
Name	Type	Size	>Address	In/Out				
♦↑Ch1 CycleCount	UINT	2.0	26.0	Input				
Ch1 Input	BVTE	1.0	28.0	Input				
- 🕸 🚯 🚯 🕺 🕪	ange <u>S</u> ing	le Link:	29.0	Input				
♦↑ Ch1 Input Ch.	ange <u>M</u> ulti Li	nk 🔊	30.0	Input				
🔍 Chi Input	ar Links		31.0	Input				
• Currinboo	ar Links	-	32.0	Input				
♦1 Chi Input	oto		33.0	Input				
Cut tubud as a line	CUC		34.0	Input				
Chi Input Mo	ve Address.		35.0	Input				
	VE AUUIESS.		36.0	Input				
Ch1 Input     →3 On     →	line <u>W</u> rite		37.0	Input				
♦Ferr Tubud	_		38.0	Input				
	line <u>F</u> orce,,,		39.0	Input				
Chi Input 💥 Re	ease Force		40.0	Input				
📢 Chi Input			41.0	Input				
♦↑ Ch1 Input Me	rge Variable	s	42.0	Input				
<pre> Ch1 Input </pre>			43.0	Input				
🔄 🔶 Chi Input 📿 Ad	d To Watch		44.0	Input				
Chi Input	move From <sup>v</sup>	Natch	45.0	Input				
♦I CUT TUDA			46.0	Input				
📢 Chi Input	nt List	Ctrl+P	47.0	Input				
♦Î Chi Input <sup>I</sup> Chi Input <sup>I</sup> Chi Input <sup>I</sup>		Ctrl+C	48.0	Input				
	by List	Cutte	49.0 50.0	Input				
♦1 Ch1 Input ♦1 Ch1 Input (Ch1 Input	port List		50.0 51.0	Input				
		1.11		Input				
♦][Ch1 Input ♦] Ch2 CycleCount	BYIE	1.U 2.0	52.0 53.0	Input Input				
♦↑ Ch2 CycleCounc	BYTE	2.0	55.0	Input				
VI CHZ INDUC	DYIC	1.0	55.0	TUDUC				

Fig. 170: Right-click, Change Multi Link

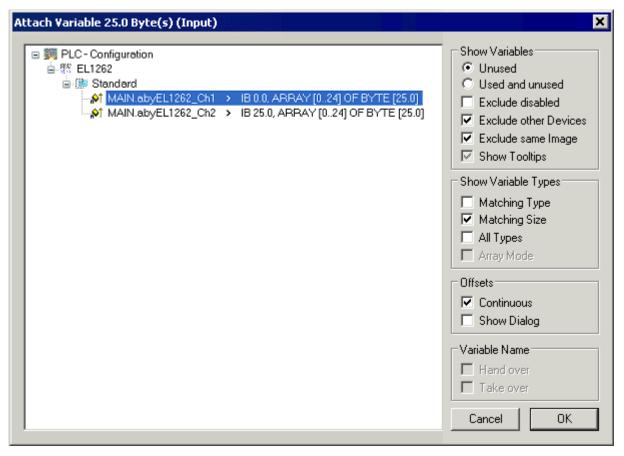


Fig. 171: Select the variable range from the task

### 5.6 Switching characteristics

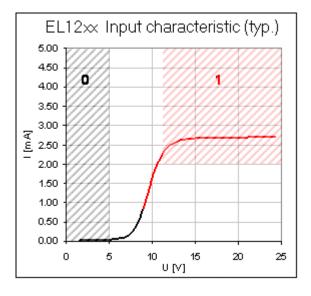


Fig. 172: Input curve based on EN 61131-2, type 3; typical measurement (Beckhoff reserves the right to make changes without prior notice!)

The input circuit of the EL12xx is optimized for fast signal changes and for the fastest possible signal acquisition. The duration required by a signal change (a rising or falling edge) to propagate from the clamping point at the front of the terminal through to the logic of the central evaluation unit (ESC) is specified for the EL12xx series as  $T_{on}/T_{off} < 1 \mu$ s, for both rising ( $T_{on}$ ) and falling edges ( $T_{off}$ ). The low absolute magnitude of this propagation time means that the temperature drift of the propagation time is also very small.

It should be borne in mind that the input circuit does not include *any* filtering. It has been optimized for the fastest possible signal transmission from the input to the evaluation unit. Fast level changes or pulses in the µs range therefore reach the evaluation unit unfiltered or unattenuated. It may be necessary to use screened cables in order to eliminate interference from the surroundings.

The sensor/signal transducer must be able to generate sufficiently steep signal edges. The power supply unit used should have sufficient buffer reserves to ensure that the signal reaches the terminal with a sufficiently steep edge in spite of capacitive or inductive cable losses.

### 5.7 Application of the SENT protocol with EL1262-0050

#### SAE J2716 SENT (Single Edge Nibble Transmission)

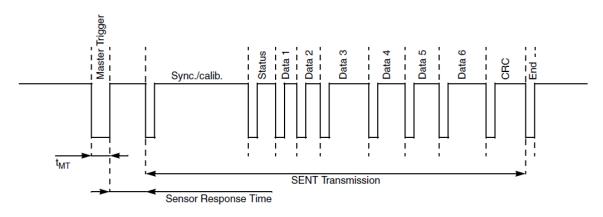
The SENT protocol is a point-to-point scheme for transmitting signal values from a sensor to a controller. It is intended to allow high resolution data transmission with a lower system cost than available serial data solutions.<sup>(1)</sup>

#### Hardware

The SENT protocol is a one-way, synchronous voltage interface which requires three wires: a signal line (low state < 0.5V, high state > 4.1V), a supply voltage line (5 V) and a ground line.

#### Protocol

Data is transmitted in units of 4 bits (1 Nibble). The interval between two falling edges (single edge) of the modulated signal with a constant amplitude voltage identifies the beginning of a frame and have to be evaluated on the receiver side.



#### Fig. 173: SENT Protokoll

A SENT message is 32 bits long (8 Nibbles) and consists of the following components:

- 24 bits of signal data (6 Nibbles) that represents 2 measurement channels of 3 Nibbles each (such as pressure and temperature)
- 4 bits (1 Nibble) for CRC error detection
- 4 bits (1 Nibble) of status/communication information

(1) SAE J2716 standard, sae.org, accessed 2011-09-13

#### Implementation of the Control

```
⊕_fb_SENT
wEI1262_State (%IB128)
                                                                                                                                                   = 16#0008
    bEL1262_WC (%IX130.0) = EAUSE
⊞-byInput (%IB0)
    ⊞…fbFlanke
                iNewData = 16#2F45
byStatus = 16#0C
iValue_1 = 16#00CD
iValue_2 = 16#0034
                LogicCheckOK = 16#00002F45
LogicChecknOK = 16#00000000
  IF wEL1262_State.3 AND NOT bEL1262_WC THEN
                         fb_SENT(byInput:=byInput); (* Call Sent functionblock *)
                         fbFlanke(CLK:=fb_SENT.bNewData ); (* if new Data are available *)
                       fbFlanke(CLK:=fb_SENT.bNewData ); (* if new Data are available *)
IF fbFlanke(CLK:=fb_SENT.bNewData ); (* if new Data are available *)
iNewData:=iNewData+1; (* Counter: Counts up ++1 if a sent protocol rece:
byStatus:=fb_SENT.arSentData[0];
iValue_1:=BYTE_TO_WORD(fb_SENT.arSentData[3])+ROL(BYTE_TO_WORD(fb_SENT.arSentData[1]);
iValue_2:=fb_SENT.arSentData[5]+fb_SENT.arSentData[4]*16#F;
(* Only for Slave which mirror the Data nibbles *)
IF (fb_SENT.arSentData[1]+fb_SENT.arSentData[6])=15 THEN
LoisCharbOW(Lasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CCLasi_CC
                                                                     LogicCheckOK:=LogicCheckOK+1;
                                                ELSE
                                                                     LogicChecknOK:=LogicChecknOK+1;
                                               END IF
                         END IF
   END_IF
```

Fig. 174: Extract of the ST implementation within TwinCAT

#### Also see about this

Example programs [> 136]

### 5.8 Access to TEDS with EL1262-0050 and EL2262

A simple communication with TEDS modules as components of sensors and actuators can be applied with the terminals EL1262-0050 and EL2262.

See example: reading and writing TEDS data [ 139].

The URN can be read out with the example to use further functions of it.

### 5.9 Example programs

#### Using the sample programs

This document contains sample applications of our products for certain areas of application. The application notes provided here are based on typical features of our products and only serve as examples. The notes contained in this document explicitly do not refer to specific applications. The customer is therefore responsible for assessing and deciding whether the product is suitable for a particular application. We accept no responsibility for the completeness and correctness of the source code contained in this document. We reserve the right to modify the content of this document at any time and accept no responsibility for errors and missing information.

#### Example 1: Frequency measurement with inductive sensor

Bownload: (https://infosys.beckhoff.com/content/1033/el1262/Resources/zip/1675495563.zip)

Data:

- 1 ms cycle time
- · 1000x oversampling on both channels

Connection diagram:

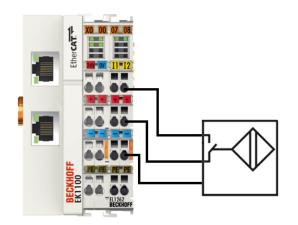


Fig. 175: Wiring for example program 1

#### Example 2: SENT protocol

Bownload: (https://infosys.beckhoff.com/content/1033/el1262/Resources/zip/4241239179.zip)

- Hardware requirements:
  - EL1262-0050
  - EL9505
  - A control that allows at least 1 ms task cycle time (IPC/ CX)

The EL1262 is able to work with an oversampling of 1 µs and also 1000 Bit per 1 ms are available.

A Sent telegram has a resolution of 3  $\mu$ s per digit. So, if 1  $\mu$  resolution will be used, it is enough to interpret the received data signal.

Connection diagram:

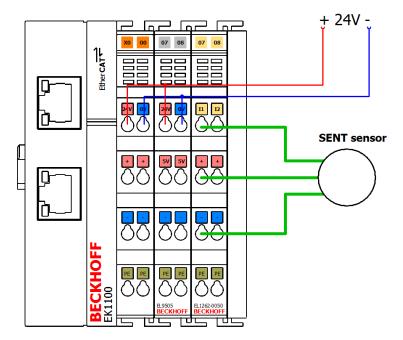


Fig. 176: Wiring for example program 2

#### Starting the example program

The application examples have been tested with a test configuration and are described accordingly. Certain deviations when setting up actual applications are possible.

The following hardware and software were used for the test configuration:

- TwinCAT master PC with Windows XP Professional SP 3, TwinCAT version 2.10 (Build 1330) and INTEL PRO/100 VE Ethernet adapter
- Beckhoff EtherCAT Coupler EK1100, EL1262 and EL9011 Terminals
- Inductive proximity limit switch, switching to positive potential, with 3-wire connection, max. switching frequency: 3000 Hz

#### Procedure for starting the program

- After clicking the Download button, save the zip file locally on your hard disk, and unzip the \*.TSM (configuration) and the \*.PRO (PLC program) files into a temporary working folder
- Run the \*.TSM file and the \*.PRO file; the TwinCAT System Manager and TwinCAT PLC will open
- Connect the hardware in accordance with <u>Wiring for example program 1 [> 135]</u> (other electrical or mechanical switching elements configured for a normally open function can be used instead of the inductive proximity limit switch), and connect the Ethernet adapter of your PC to the EtherCAT coupler (you will find further instructions on this in the corresponding coupler manuals)
- Select the local Ethernet adapter (with real-time driver, if applicable) under System configuration, I/O configuration, I/O devices, Device (EtherCAT); then on the "Adapter" tab choose "Search...", select the appropriate adapter and confirm (see Fig. Searching the Ethernet adapter + Selection and confirmation of the Ethernet adapter).

FL1262 example frequency measurement.ts	m - TwinCAT System	Manager	
File Edit Actions View Options Help			
] 🗅 🖆 🗳 🖬   🍜 🖪   X 🖻 🖻 🛱   4	4 8 🖳 🙃 🗸	🎯 🙆 🧶 🗞 🔨 🎯 🗎	Q 🖓 66' 🗙 🕵 🧶 🕽 '
SYSTEM - Configuration NC - Configuration Cam - Configuration Superiode Cam - Configuration Sup	General Adapter E Description: Device Name: MAC Address: IP Address: Freerun Cycle (ms):	therCAT   Online   CoE - Online   FC9004_4 (Intel(R) PRO/100 S Deskto \DEVICE\(E012AEDA-299C-4EDB-8CS 00 01 05 01 05 81 172.16.8.10 (255.255.255.0) 4 • Promiscuous Mode (use with Netmo Virtual Device Names (TwinCAT v2.	51-03EC890CBE25}
		ILUCA	(172.10.0.45.1.1) Conny Mode //

Fig. 177: Searching the Ethernet adapter



Fig. 178: Selection and confirmation of the Ethernet adapter

Activate and confirm the configuration (Fig. Activation of the configuration + Confirming the activation of the configuration)



Fig. 179: Activation of the configuration

TwinCAT System Manager 🛛 🗙						
2	Activate Configuration (Old Configurations will be overwritten!)					
	OK Cancel					

Fig. 180: Confirming the activation of the configuration

• Confirm new variable mapping, restart in RUN mode (Fig. *Generate variable mapping* + *Restarting TwinCAT in RUN mode*)



Fig. 181: Generating variable mapping



Fig. 182: Restarting TwinCAT in RUN mode

• In TwinCAT PLC, under the "Project" menu, select "Rebuild all" to compile the project (Fig. *Compile project*)

👺 TwinCAT PLC Control - EL1262 example frequency measurement.pro								
File	Edit	Project	Insert	Extras	Online	Window	Help	
1	اھا	Build				Ctrl+F8		
		Rebui	id all					
	POU: " 🗐 📘	Load		d informa	<b>s</b> ition			
		Objec	:t ct datab	ase			) )	
		Optio	ns					
		Trans	late into	other lar	nguages		•	
		Docur	ment					
		Expor	t					

Fig. 183: Compile project

• In TwinCAT PLC: log in with the "F11" button, confirm loading the program (Fig. *Confirming program start*), run the program with the "F5" button

TwinCAT	PLC Control			×
?	No program on	the controller! D	ownload the new	program?
	Yes	No	Cancel	

Fig. 184: Confirming program start

#### Example 3: reading and writing TEDS data

#### Program description / function

This sample program illustrates how to read/write the data of a separate TEDS module (TEDS = Transducer Electronic Data Sheet). Such TEDS modules are available on the market for retrofitting sensors or actuators, in order to identify the device after installation or to read out specific data (calibration, manufacturer etc.). The device used in this example was an HBM TEDS 1-TEDS-BOARD-L, version 2018.

This sample program is expressly intended as a feasibility demonstration. Specifically, there is no claim to interoperability with any other TEDS modules. It is the responsibility of users to transfer the methods formulated here to their own implementations.

This demonstration does not cover TEDS modules that are integrated in the sensor and communicate on the sensor lines. This is common for IEPE (vibration) or strain gauges/measuring bridges. It is possible to connect an IEPE sensor equipped with TEDS to Beckhoff ELM3602/ELM3604 terminals.

The following configuration is required:

#### [EK1100] + [EL2262] + [EL9505] + [EL1262-0050] + EL9011

The configuration can control 2 TEDS modules. Only single-channel operation is shown in the example.

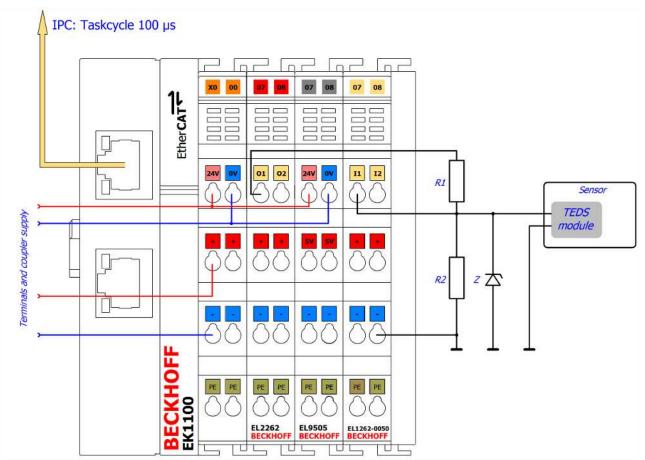


Fig. 185: Wiring for sample program 3

The voltage divider can be dimensioned with R1 = 2180  $\Omega$  (e.g. 680  $\Omega$  + 1500  $\Omega$ ), R2 = 680  $\Omega$  and Z = 5,1 V for example.

#### Notes on the program (visualization)

First the URN has to be read (A). Only then are further functions available.

The program determines the URN for each bit by reinitializing the module, since the terminal for the input causes a time offset that is too large (see "Bit repeat count" at the top right).

Data can be written either by entering hexadecimal values (B) or a text string (ASCII) (C); hexadecimal values must be separated by spaces in the text field. Which of the two inputs is to be used for writing can be specified with the checkbox "*Write ASCII data*" (E):

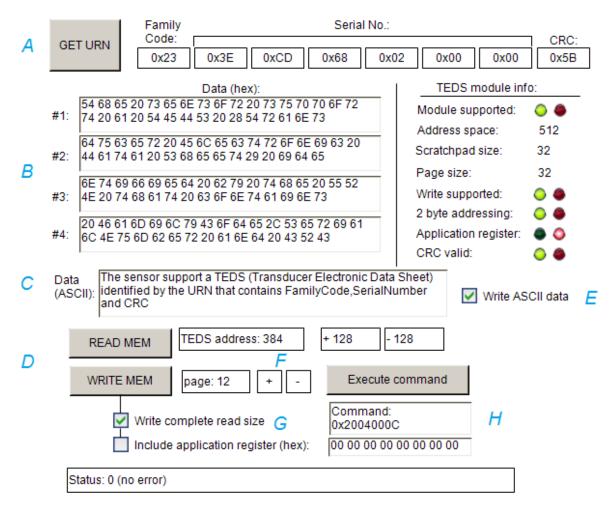


Fig. 186: Visualization of the sample program for TEDS with EL1262-0050 and EL2262

The basic function after the identified URN is (D) reading (READ MEM) and writing (WRITE MEM) TEDS data. By issuing such a command, the associated command statement is generated in the text field (H) and can also be changed and then executed with "Execute command". Via +/- the TEDS address or page can be changed (F). Both the start address and "page" can be entered directly for read / write accesses.

The hexadecimal data (B) of *text field* #1 to #4 each represent 32 bytes of the total read/write buffer size of 128 bytes, as configured in the sample program. If the checkbox "*Complete read size*" (G) is unchecked, only *text field* #1 will be used for writing usually (except the module supports page sizes > 32 byte). Accordingly, only the first characters of the ASCII data text will be written. In any case, the number of bytes as a page of the module is configured will be used. Note, that the module usually supports write access to addresses of a multiple value of the page size only (so, e.g. assuming a page size of 32 bytes and an address 234 is input, an error 0x35 'writing fail' will occur when WRITE MEM is commanded and confirmed; but 352 is valid and results no error).

Selection of "Include application register" provides whether the application register shall be written or read additionally (G).

#### Download:

https://infosys.beckhoff.com/content/1033/el1262/Resources/zip/5750275595.zip

#### Preparations for starting the sample programs (tnzip file / TwinCAT 3)

• Click on the download button to save the Zip archive locally on your hard disk, then unzip the \*.tnzip archive file in a temporary folder.

File	Edit View Project Build Debug 1	TwinCAT TwinSAF	FE	PLC Tools	Scope Window He	lp
	New	•		Release	▪ TwinCAT RT (x64)	- 🎽
	Open	<u> </u>	đ	Project/Solut	ion	Ctrl+Shift+O
	Add	• 1	2	Web Site		Shift+Alt+O
	Close	2	2	File		Ctrl+O
ി	Close Solution		2	Open Project	from Target	
	Save TwinCAT Project1.sln	Ctrl+S		Open Solutio	n from Archive	B
	Save TwinCAT Project1.sln As					~

Fig. 187: Opening the \*. tnzip archive

- Select the .tnzip file (sample program).
- · A further selection window opens. Select the destination directory for storing the project.
- For a description of the general PLC commissioning procedure and starting the program please refer to the terminal documentation or the EtherCAT system documentation.
- The EtherCAT device of the example should usually be declared your present system. After selection of the EtherCAT device in the "Solutionexplorer" select the "Adapter" tab and click on "Search...":

eneral Adapter Et	herCAT Online CoE - Online	
Network Adapte	er	
	OS (NDIS) OPCI	DPRAM
Description:		
Device Name:		
PCI Bus/Slot:		Search
MAC Address:		Compatible Devices
IP Address:		
	Promiscuous Mode (use with Wir	reshark only)
	Virtual Device Names	
Adapter Refere	nce	
Adapter:		
reerun Cycle (ms):	4	

Fig. 188: Search of the existing HW configuration for the EtherCAT configuration of the example

• Checking NetId: the "EtherCAT" tab of the EtherCAT device shows the configured NetId:

General Adapte	EtherCAT	Online C	oE - Online	
NetId:	127.0.0.1.4.1			Advanced Settings

The first 4 numbers have to be identical with the project Netld of the target system. The project Netld can be viewed within the TwinCAT environment above, where a pull down menu can be opened to choose a target system (by clicking right in the text field). The number blocks are placed in brackets there next to each computer name of a target system.

- Modify the NetId: By right clicking on "EtherCAT device" within the solution explorer a context menu opens where "Change NetId..." have to be selected. The first 4 numbers of the NetId of the target computer have to be entered; the both last values are 4.1 usually. Example:
  - NetId of project: myComputer (123.45.67.89.1.1)
  - Entry via "Change NetId…": 123.45.67.89.4.1

## 6 Appendix

### 6.1 EtherCAT AL Status Codes

For detailed information please refer to the EtherCAT system description.

### 6.2 Firmware compatibility

The terminals of the EL12xx series have no firmware.

### 6.3 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 (<u>https://www.beckhoff.de</u>). 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.

### 6.3.1 Device description ESI file/XML

# **NOTE**Attention regarding update of the ESI description/EEPROM

Some slaves have stored calibration and configuration data from the production in the EEPROM. These are irretrievably overwritten during an update.

The ESI device description is stored locally on the slave and loaded on start-up. Each device description has a unique identifier consisting of slave name (9 characters/digits) and a revision number (4 digits). Each slave configured in the System Manager shows its identifier in the EtherCAT tab:

SYSTEM - Configuration	General EtherCAT	Process Data Startup	CoE - Online Online	
SYSTEM - Configuration     NC - Configuration     PLC - Configuration     I/O - Configuration     I/O - Configuration     I/O Devices     I/O Device 2 (EtherCAT)     Device 2-Image     Device 2-Imag	General EtherCAT Type: Product/Revision: Auto Inc Addr: EtherCAT Addr: Previous Port:	Process Data Startup EL3204 4Ch. Ana. Inp EL3204-0000-0016 FFFF 1002	0 1	ettings

Fig. 189: 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:

SYSTEM - Configuration     NC - Configuration     PLC - Configuration     J/O - Configuration     J/O - Configuration     J/O Devices     J/O Devices	General Adapter Et
→      →      →      →      →      →      →      →      →      →      →      →      →      →      →      →      ↓      ○utpt     ⊕      →      ↓      Outpt     ⊕      ↔      ↓      Outpt     ⊕      ↔      ↓      Outpt     ⊕      ↔      ↓      Outpt     ⊕      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↓      ↔      ↔      ↓      ↔      ↔      ↓      ↔	e
TI TI Scan Boxes.	

Fig. 190: 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 🛛 🔯				
٩	Configuration is identical			
	ОК			

Fig. 191: Configuration is identical

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

Check Configuration		🖂
Found Items:	Disable > Ignore > Delete > Copy Before > Copy After > Copy After > Copy After > Copy After > OK Cancel	Configured Items:
<ul> <li>Extended Information</li> </ul>		

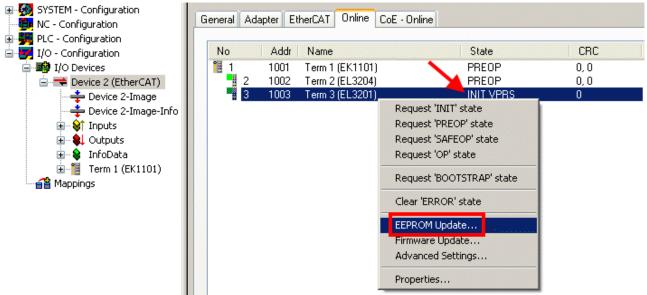
Fig. 192: Change dialog

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

#### Changing the ESI slave identifier

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

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



#### Fig. 193: 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.

Write EEPROM	🖂
Available EEPROM Descriptions:	ОК
EL3162 2Ch. Ana. Input 0-10V (EL3162-0000-0000)	
EL3201 1Ch. Ana. Input PT100 (RTD) (EL3201-0000-0016)	Cancel
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)	
B 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. 194: 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 Ether-CAT slave therefore has to be switched off briefly in order for the change to take effect.

### 6.3.2 Firmware explanation

#### Determining the firmware version

#### Determining the version on laser inscription

Beckhoff EtherCAT slaves feature serial numbers applied by laser. The serial number has the following structure: **KK YY FF HH** 

KK - week of production (CW, calendar week) YY - year of production FF - firmware version

HH - hardware version

Example with ser. no.: 12 10 03 02:

12 - week of production 12

10 - year of production 2010

03 - firmware version 03

02 - hardware version 02

#### Determining the version via the System Manager

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

#### CoE Online and Offline CoE

Two CoE directories are available:

online: This is offered in the EtherCAT slave by the controller, if the EtherCAT slave supports this. This CoE directory can only be displayed if a slave is connected and operational.
offline: The EtherCAT Slave Information ESI/XML may contain the default content of the CoE. This CoE directory can only be displayed if it is included in the ESI (e.g. "Beckhoff EL5xxx.xml").

The Advanced button must be used for switching between the two views.

In Fig. *Display of EL3204 firmware version* the firmware version of the selected EL3204 is shown as 03 in CoE entry 0x100A.

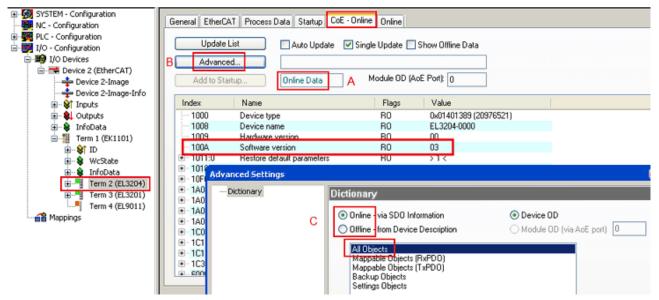


Fig. 195: Display of EL3204 firmware version

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

### 6.3.3 Updating controller firmware \*.efw

#### CoE directory

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

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

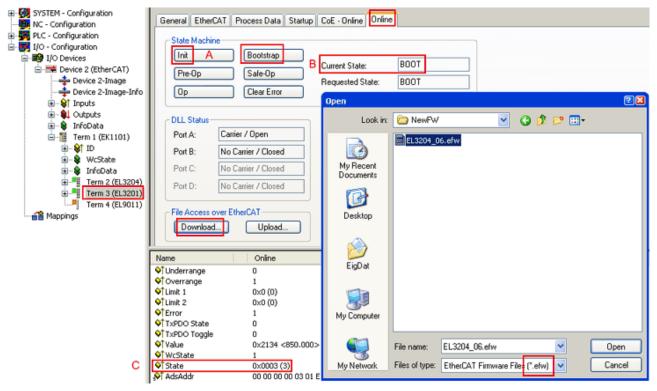


Fig. 196: Firmware Update

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	Microsoft Visual Studio
Coad I/O Devices	Activate Free Run
Yes No	Yes No

• Switch EtherCAT Master to PreOP

Solution Explorer	<b>-</b> ₽ ×	.⇒ × MAIN		
© ⊖ ☆ 'o - ₫ / <b>/ -</b>		General Adapter EtherCA Online Co E - Online		
Search Solution Explorer (Ctrl+ü)	<u>- م</u>			
SAFETY		No Addr Name State		CRC
% C++		1 1001 Term 5 (EL1004) PREC	P	0, 0
∡ <u>∠</u> I/O		2 1002 Term 6 (EL2004) PREC	P	0, 0
▲ 📲 Devices		cii 3 1003 Term 7 (EL6688)	P	0
✓ ➡ Device 2 (EtherCAT)				
Tmage-Info				
SyncUnits		Actual State: PREOP Counter	Cyclic	Queued
Inputs		Init P-Op Safe-Op Op Send Frames	17167	+ 5289
🔁 Frm0State		Clear RC Clear Frames Frames / sec	499	+ 43
🔁 Frm0WcState		Lost Frames	0	+ 0
🔁 Frm0InputToggle		Tx/Rx Errors	0	/ 0
✤ SlaveCount				
🔁 DevState				

- 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!	
ОК	

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

### 6.3.4 FPGA firmware \*.rbf

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

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

The firmware version number included in the terminal serial number contains both firmware components. If one of these firmware components is modified this version number is updated.

#### Determining the version via the System Manager

The TwinCAT System Manager indicates the FPGA firmware version. Click on the Ethernet card of your EtherCAT strand (Device 2 in the example) and select the *Online* tab.

The *Reg:0002* column indicates the firmware version of the individual EtherCAT devices in hexadecimal and decimal representation.

😎 TwinCAT System Manager	
<u>File Edit Actions View Options Help</u>	
] 🗅 📂 📽 🔚 🎒 强 👗 🛍 💼	💼 🛤 ð 🖳 📾 🗸 🎯 👧 👧 🎨 🔨 🚳 🖹
SYSTEM - Configuration     CNC - Configuration	General Adapter EtherCAT Online
NC - Configuration	No Addr Name State CRC Reg:0002 📐
PLC - Configuration	1 1001 Term 1 (EK1100) OP 0 0x0002 (11)
🖻 🐺 I/O - Configuration	2 1002 Term 2 (EL1012) OP 0 0x0002 (10)
🖻 🏘 I/O Devices	📕 3 1003 Term 3 (EL2004) OP 0 0x0002 (11)
🖃 📲 Device 2 (EtherCAT)	4 1004 Term 4 (EL3102) OP 0 0x0002 (10)
Device 2-Image	5 1005 Term 5 (EL4102) OP 0 0x000B (11)
🛁 Device 2-Image-Info	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         0x0008 (11)           6         1006         Term 6 (EL5001)         OP         0         0x0002 (11)           7         1007         Term 7 (EL6751)         OP         0         0x000C (12)
🗄 🕸 💱 Inputs	7 1007 Term 7 (EL6751) OP 0 0x000C (12)
Dutputs	Actual State: OP Send Frames: 74237
⊕- 😵 InfoData ⊕- 🛄 Term 1 (EK1100)	Init Pre-Op Safe-Op Op Frames / sec: 329
Mappings	Clear CRC Clear Frames 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
	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 ▼
I I	,
Ready	Local () Free Run

Fig. 197: FPGA firmware version definition

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

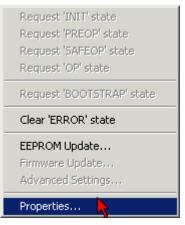


Fig. 198: Context menu Properties

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.

1	Advanced Settings		×
	⊡ - Diagnosis <b>Online View</b> ⊡ - Emergency Scan	Online View □ 0000 'ET1xxxx Rev/Type'  □ 0002 'ET1xxxx Build' □ 0004 'SM/FMMU Cnt' □ 0006 'DPRAM Size' □ 0008 'Features' □ 0010 'Phys Addr' □ 0012 'Phys Addr 2nd'	0000 Add
	,		OK Abbrechen

Fig. 199: 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):

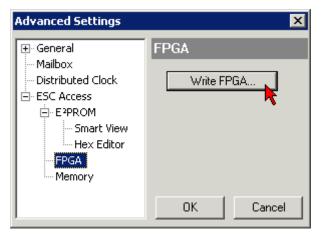
• Switch TwinCAT system to ConfigMode/FreeRun with cycle time >= 1 ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.

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

click the Advanced Settings button in the EtherCAT tab:

🛒 TwinCAT System Manager				- 🗆 ×
File Edit Actions View Options Help				
] 🗅 🚅 📽 🔚 🎒 🖪 🖁 📾 📾	B 🗛 👌	🔍 📾 🗸 💣 👧 🖉	<b>)</b> 😫 🔨 🎯	💊   🖹
SYSTEM - Configuration CNC - Configuration NC - Configuration NC - Configuration I/O - Configuration I/O - Configuration I/O Devices I/O Devices CHARP Gerät 2 (EtherCAT) CHARP Device 2-Image	General Ethe Type: Product / Revisi Auto Inc Addre EtherCAT Addre	ss: FFFC	ncoder	
Device 2-Image Device 2-Image Device 2-Image-Info Device 2-Image-Info Device 2-Image Inputs Device 2-Image InfoData Device 2-Image InfoData Term 1 (EL2004) Device 2-Image InfoData Term 3 (EL2004) Device 2-Image InfoData InfoData InfoData InfoData InfoData InfoData InfoData InfoData InfoData InfoData InfoData InfoData InfoData InfoData	Previous Port:	Term 4 (EL5001) -	B	<u>_</u>
🗄 😵 Channel 1	Name	Online	Туре	Size
	<ul> <li>♦↑ Status</li> <li>♦↑ Value</li> <li>♦↑ WcState</li> <li>♦↑ State</li> <li>♦↑ AdsAddr</li> </ul>	0x41 (65) 0x00000000 (0) 0 0x0008 (8) AC 10 03 F3 03 01 ED 03	BYTE UDINT BOOL UINT AMSADDRESS	1.0 4.0 0.1 2.0 8.0
Ready			Local () Con	fig Mode //

• The *Advanced Settings* dialog appears. Under *ESC Access/E<sup>2</sup>PROM*/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 💌	G 🤌 📂 🖽-
SocCOM_T1_EBUS_BGA_LVTTL_F2_S	54_BLD12.rbf
File name: A_LVTL_F2_S4_BLD12.rbf	Open
File type: FPGA File (*.rbf)	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!

### 6.3.5 Simultaneous updating of several EtherCAT devices

The firmware and ESI descriptions of several devices can be updated simultaneously, provided the devices have the same firmware file/ESI.

General Adapter EtherCAT Online CoE - Online		
General         Adapter         EtherCAT         Or           No         Addr         Name           1         1001         Term 5 (EK)           2         1002         Term 6 (EL)           3         1003         Term 7 (EL)           4         1004         Term 8 (EL)           5         1005         Term 9 (EL)	State           (1101)         INIT           .3102)         INIT           .3102)         INIT           .3102)         INIT           .3102)         INIT	
	Firmware Update	

Fig. 200: Multiple selection and firmware update

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

### 6.4 Restoring the delivery state

To restore the delivery state for backup objects in 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*)

General EtherCA	T DC Proc	cess Data 🛛 Start	up CoE · Or	nline   Onl	ine		
Update L	.ist 🛛 🗆	Auto Update	🗐 Single Up	odate 🔽 S	Show Offline	e Data	
Advance	d						
Add to Star	tup	etting objects					
Index	Name		Fk	ags	Value		<b>_</b>
1000	Device type		R	)	0x00001	389 (5001)	
1008	Device name		B	0	EL5101		
1009	Hardware vers	sion	B	כ	09		
100A	Software versi	on	B	0	10		
⊡ 1011:0	Restore defau	lt parameters	R	_	>1<		
	SubIndex 001		۳		0x00000	000 (0)	
	Identity	1	RI		> 4 <		
Name	Туре	Size	>Addr	In/Out	User ID	Linked to	
<b>♦</b> ↑ Status	USINT	1.0	26.0	Input	0		
<b>\$</b> † Value	UINT	2.0	27.0	Input	0		
<b>\</b> ↓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	AMSADI	DRESS 8.0	1552.0	Input	0		
of petId	ADDAV	F0 60	1552.0	Toput	n		

Fig. 201: 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 Dia	log	×
Dec:	1684107116	ОК
Hex:	0x64616F6C	Abbruch
Float:	1684107116	
Bool:	0 1	Hex Edit
Binär:	6C 6F 61 64	4
Bitgröße	○1 ○8 ○16 ⊙32	○ 64 ○ ?

Fig. 202: 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.

### 6.5 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

#### Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for <u>local support and service</u> 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: Fax: e-mail: +49(0)5246/963-0 +49(0)5246/963-198 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
 +49(0)5246/963-479

 Fax:
 +49(0)5246/963-479
 service@beckhoff.com

# List of Illustrations

Fig.	1	EL5021 EL terminal, standard IP20 IO device with serial/ batch number and revision ID (since 2014/01)
Fig.	2	EK1100 EtherCAT coupler, standard IP20 IO device with serial/ batch number
Fig.	3	CU2016 switch with serial/ batch number
Fig.	4	EL3202-0020 with serial/ batch number 26131006 and unique ID-number 204418 10
Fig.	5	EP1258-00001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique se- rial 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 num- ber 0033170110
Fig.	8	ELM3604-0002 terminal with unique ID number (QR code) 100001051 and serial/ batch number 44160201
Fig.	9	EL1262-0000
Fig.	10	EL1262-0050
Fig.	11	System manager current calculation
Fig.	12	EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog
Fig.	13	States of the EtherCAT State Machine
Fig.	14	"CoE Online " tab
Fig.	15	Startup list in the TwinCAT System Manager
Fig.	16	Offline list
Fig.	17	Online list 23
Fig.	18	Spring contacts of the Beckhoff I/O components
Fig.	19	Attaching on mounting rail
Fig.	20	Disassembling of terminal
Fig.	21	Power contact on left side
Fig.	22	Standard wiring
Fig.	23	Pluggable wiring
Fig.	24	High Density Terminals
Fig.	25	Mounting a cable on a terminal connection
Fig.	26	Recommended distances for standard installation position
Fig.	27	Other installation positions
Fig.	28	Correct positioning
Fig.	29	Incorrect positioning
Fig.	30	EL1262-0000
Fig.	31	EL1262-0050
Fig.	32	Relationship between user side (commissioning) and installation
Fig.	33	Control configuration with Embedded PC, input (EL1004) and output (EL2008)
Fig.	34	Initial TwinCAT 2 user interface
Fig.	35	Selection of the target system
Fig.	36	Specify the PLC for access by the TwinCAT System Manager: selection of the target system 46
Fig.	37	Select "Scan Devices"
Fig.	38	Automatic detection of I/O devices: selection the devices to be integrated
Fig.	39	Mapping of the configuration in the TwinCAT 2 System Manager
Fig.	40	Reading of individual terminals connected to a device
Fig.	41	TwinCAT PLC Control after startup

Fig. 42	Sample program with variables after a compile process (without variable integration)	50
Fig. 43	Appending the TwinCAT PLC Control project	50
Fig. 44	PLC project integrated in the PLC configuration of the System Manager	51
Fig. 45	Creating the links between PLC variables and process objects	51
Fig. 46	Selecting PDO of type BOOL	52
Fig. 47	Selecting several PDOs simultaneously: activate "Continuous" and "All types"	52
Fig. 48	Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample	53
Fig. 49	Choose target system (remote)	54
Fig. 50	PLC Control logged in, ready for program startup	55
Fig. 51	Initial TwinCAT 3 user interface	56
Fig. 52	Create new TwinCAT project	56
Fig. 53	New TwinCAT3 project in the project folder explorer	57
Fig. 54	Selection dialog: Choose the target system	57
Fig. 55	Specify the PLC for access by the TwinCAT System Manager: selection of the target system	58
Fig. 56	Select "Scan"	58
Fig. 57	Automatic detection of I/O devices: selection the devices to be integrated	59
Fig. 58	Mapping of the configuration in VS shell of the TwinCAT3 environment	59
Fig. 59	Reading of individual terminals connected to a device	60
Fig. 60	Adding the programming environment in "PLC"	61
Fig. 61	Specifying the name and directory for the PLC programming environment	61
Fig. 62	Initial "Main" program of the standard PLC project	62
Fig. 63	Sample program with variables after a compile process (without variable integration)	63
Fig. 64	Start program compilation	63
Fig. 65	Creating the links between PLC variables and process objects	64
Fig. 66	Selecting PDO of type BOOL	64
Fig. 67	Selecting several PDOs simultaneously: activate "Continuous" and "All types"	65
Fig. 68	Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample	65
Fig. 69	TwinCAT development environment (VS shell): logged-in, after program startup	66
Fig. 70	System Manager "Options" (TwinCAT 2)	68
Fig. 71	Call up under VS Shell (TwinCAT 3)	68
Fig. 72	Overview of network interfaces	68
Fig. 73	EtherCAT device properties(TwinCAT 2): click on "Compatible Devices…" of tab "Adapter"	69
Fig. 74	Windows properties of the network interface	69
Fig. 75	Exemplary correct driver setting for the Ethernet port	70
Fig. 76	Incorrect driver settings for the Ethernet port	71
Fig. 77	TCP/IP setting for the Ethernet port	72
Fig. 78	Identifier structure	73
Fig. 79	OnlineDescription information window (TwinCAT 2)	74
Fig. 80	Information window OnlineDescription (TwinCAT 3)	74
Fig. 81	File OnlineDescription.xml created by the System Manager	75
Fig. 82	Indication of an online recorded ESI of EL2521 as an example	75
Fig. 83	Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)	75
Fig. 84	Using the ESI Updater (>= TwinCAT 2.11)	77
Fig. 85	Using the ESI Updater (TwinCAT 3)	77
Fig. 86	Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)	78
Fig. 87	Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)	78

Fig.	88	Selecting the Ethernet port	78
Fig.	89	EtherCAT device properties (TwinCAT 2)	79
Fig.	90	Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)	79
Fig.	91	Selection dialog for new EtherCAT device	80
Fig.	92	Display of device revision	80
Fig.	93	Display of previous revisions	81
Fig.	94	Name/revision of the terminal	81
Fig.	95	EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)	82
Fig.	96	Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)	83
Fig.	97	Scan Devices (left: TwinCAT 2; right: TwinCAT 3)	83
Fig.	98	Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)	83
Fig.	99	Detected Ethernet devices	84
Fig.	100	Example default state	84
Fig.	101	Installing EthetCAT terminal with revision -1018	85
Fig.	102	Detection of EtherCAT terminal with revision -1019	85
Fig.	103	Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: Twin-CAT 3)	85
Fig.	104	Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)	86
Fig.	105	Scan progressexemplary by TwinCAT 2	86
Fig.	106	Config/FreeRun query (left: TwinCAT 2; right: TwinCAT 3)	86
Fig.	107	Displaying of "Free Run" and "Config Mode" toggling right below in the status bar	86
Fig.	108	TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: Twin-CAT 3)	86
Fig.	109	Online display example	87
Fig.	110	Faulty identification	87
		Identical configuration (left: TwinCAT 2; right: TwinCAT 3)	88
-		Correction dialog	88
-		Name/revision of the terminal	89
Fig.	114	Correction dialog with modifications	90
		Dialog "Change to Compatible Type" (left: TwinCAT 2; right: TwinCAT 3)	90
-		TwinCAT 2 Dialog Change to Alternative Type	90
		Branch element as terminal EL3751	91
-		"General" tab	91
-		"EtherCAT" tab	92
-		" " " " " " " " " " " " " " " " " " "	93
-		Configuring the process data	94
-		"Startup" tab	95
-		"CoE – Online" tab	96
-		Dialog "Advanced settings"	97
-		"Online" tab	97
-		"DC" tab (Distributed Clocks)	98
-		Selection of the diagnostic information of an EtherCAT Slave	100
-		Basic EtherCAT Slave Diagnosis in the PLC.	101
-		EL3102, CoE directory	103
-		Example of commissioning aid for a EL3204	103
-		Default behaviour of the System Manager	
9.			

Fig.	132	Default target state in the Slave	105
Fig.	133	PLC function blocks	106
Fig.	134	Illegally exceeding the E-Bus current	107
Fig.	135	Warning message for exceeding E-Bus current	107
Fig.	136	Oversampling PDO of the EL37xx series and in the comparison with EL31xx	108
Fig.	137	Adding a Scope project into an already existing project	110
Fig.	138	Representation of a created PLC array variable ("aUI_Samples") to link with oversampling PDOs of EL3773	111
Fig.	139	Insertion of a free task	111
Fig.	140	Task property "Create symbols" must be activated	112
Fig.	141	Insertion of variable "Var 1" fitting to the oversampling (-factor)	112
Fig.	142	Set up the link of the PLC array variable (left: for the last preceding paragraph Step 2a, right: for the last preceding paragraph Step 2b)	113
Fig.	143	Select the EL3773 PDO "L1 Voltage Samples" to create a link to the PLC array variable "aUI_Samples"	113
Fig.	144	Selection of the oversampling variable with the target browser	114
Fig.	145	Appending the variable "aUI_Samples" below "axis" within the scope project of the solution explorer	114
Fig.	146	Example of recording a sine signal with 10 x oversampling at 1 ms measurement cycle time	115
Fig.	147	Activation of the ADS server of the EtherCAT device (TwinCAT 3)	116
Fig.	148	Direct access to PDOs of the terminal by ScopeView	116
Fig.	149	PLC declaration	118
Fig.	150	PLC in the System Manager	118
Fig.	151	Add Variable Type	118
Fig.	152	Definition of the variable type	119
Fig.	153	Overview of declared types	119
Fig.	154	Linking	120
Fig.	155	Array variables of an oversampling terminal	120
Fig.	156	Settings in the additional task	120
Fig.	157	Variable browser up to the array VAR152	121
Fig.	158	AddSymbol on the array	121
Fig.	159	Channel settings	121
Fig.	160	Activation of the marks	122
Fig.	161	Illustration of a 10 x oversampling variable of the EL3751 by the Scope2	123
Fig.	162	Activation of the ADS server of the EtherCAT Device (TwinCAT 2)	124
Fig.	163	Direct access of the Scope2 to the terminal's PDOs	124
Fig.	164	Automatically calculated array variable (red) in the Scope2	125
Fig.	165	EL1262 oversampling factor selection dialog in the TwinCAT System Manager	127
Fig.	166	Typical EL1262 input characteristics. Beckhoff reserves the right to make unannounced changes.	128
Fig.	167	EL1262 default state	129
Fig.	168	EL1262 Advanced Settings, Distributed Clocks	130
Fig.	169	Select the variables in the terminal with the mouse	131
Fig.	170	Right-click, Change Multi Link	132
Fig.	171	Select the variable range from the task	133
Fig.	172	Input curve based on EN 61131-2, type 3; typical measurement (Beckhoff reserves the right to make changes without prior notice!)	133

Fig.	173	SENT Protokoll	134
Fig.	174	Extract of the ST implementation within TwinCAT	135
Fig.	175	Wiring for example program 1	136
Fig.	176	Wiring for example program 2	136
Fig.	177	Searching the Ethernet adapter	137
Fig.	178	Selection and confirmation of the Ethernet adapter	138
Fig.	179	Activation of the configuration	138
Fig.	180	Confirming the activation of the configuration	138
Fig.	181	Generating variable mapping	138
Fig.	182	Restarting TwinCAT in RUN mode	139
Fig.	183	Compile project	139
Fig.	184	Confirming program start	139
Fig.	185	Wiring for sample program 3	140
Fig.	186	Visualization of the sample program for TEDS with EL1262-0050 and EL2262	141
Fig.	187	Opening the *. tnzip archive	142
Fig.	188	Search of the existing HW configuration for the EtherCAT configuration of the example	142
Fig.	189	Device identifier consisting of name EL3204-0000 and revision -0016	144
Fig.	190	Scan the subordinate field by right-clicking on the EtherCAT device	145
Fig.	191	Configuration is identical	145
Fig.	192	Change dialog	145
Fig.	193	EEPROM Update	146
Fig.	194	Selecting the new ESI	146
Fig.	195	Display of EL3204 firmware version	147
Fig.	196	Firmware Update	148
Fig.	197	FPGA firmware version definition	150
Fig.	198	Context menu Properties	150
Fig.	199	Dialog Advanced Settings	151
Fig.	200	Multiple selection and firmware update	153
Fig.	201	Selecting the "Restore default parameters" PDO	154
Fig.	202	Entering a restore value in the Set Value dialog	154