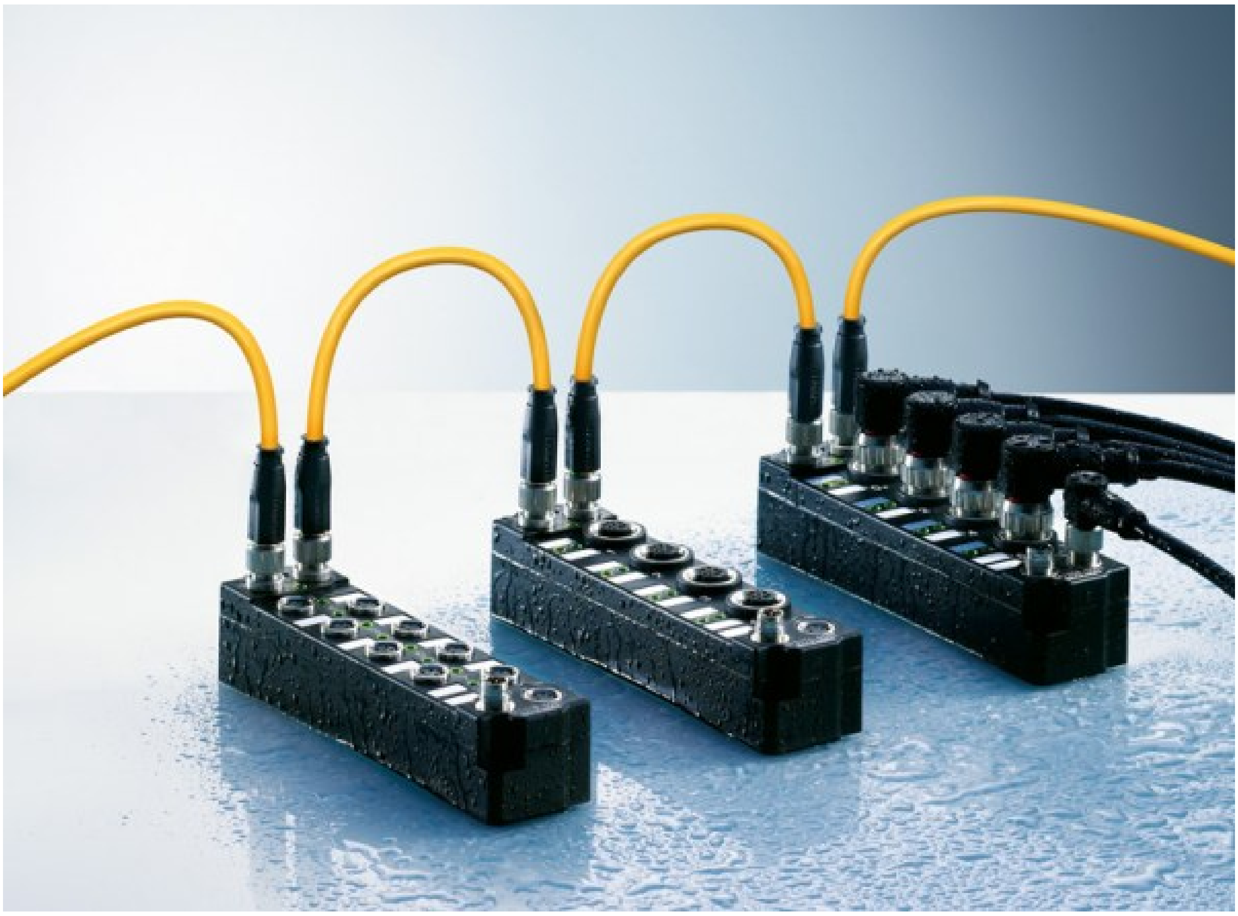


Documentation for

EP3632

2-channel interface for Condition Monitoring (IEPE)

Version: 1.1.0
Date: 2019-02-08

BECKHOFF

Table of contents

1 Foreword

1.1 Notes on the documentation

Intended audience

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

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

Disclaimer

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

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

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

Trademarks

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

Patent Pending

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

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

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

Copyright

1.2 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Description of instructions

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

⚠ DANGER

Serious risk of injury!

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

⚠ WARNING

Risk of injury!

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

⚠ CAUTION

Personal injuries!

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

NOTE

Damage to environment/equipment or data loss

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

Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3 Documentation issue status

Version	Comment
1.1.0	• Complements Technical data • Correction chapter *EP3632-0001 - Introduction* • Corrections chapter *CoE object description*
1.0.0	• Complements • 1st public issue
0.3	• Update chapter *Mounting and cabling* • Update chapter *Commissioning/Configuration*
0.2	• Update chapter "Basic function principles"
0.1	• first preliminary version

Firm and hardware version

The documentation refers to the firm and hardware status that was valid at the time it was prepared.

The properties of the modules are subject to continuous development and improvement. Modules having earlier production statuses cannot have the same properties as modules with the latest status. Existing properties, however, are always retained and are not changed, so that these modules can always be replaced by new ones.

The firmware and hardware version (delivery state) can be found in the batch number (D number) printed at the side of the EtherCAT Box.

Syntax of the batch number (D number)

D: WW YY FF HH

WW - week of production (calendar week)
YY - year of production
FF - firmware version
HH - hardware version

Example with D No. 29 10 02 01:

29 - week of production 29
10 - year of production 2010
02 - firmware version 02
01 - hardware version 01

2 Product overview

2.1 EtherCAT Box - Introduction

The EtherCAT system has been extended with EtherCAT Box modules with protection class IP 67. Through the integrated EtherCAT interface the modules can be connected directly to an EtherCAT network without an additional Coupler Box. The high-performance of EtherCAT is thus maintained into each module.

The extremely low dimensions of only 126 x 30 x 26.5 mm (h x w x d) are identical to those of the Fieldbus Box extension modules. They are thus particularly suitable for use where space is at a premium. The small mass of the EtherCAT modules facilitates applications with mobile I/O interface (e.g. on a robot arm). The EtherCAT connection is established via screened M8 connectors.

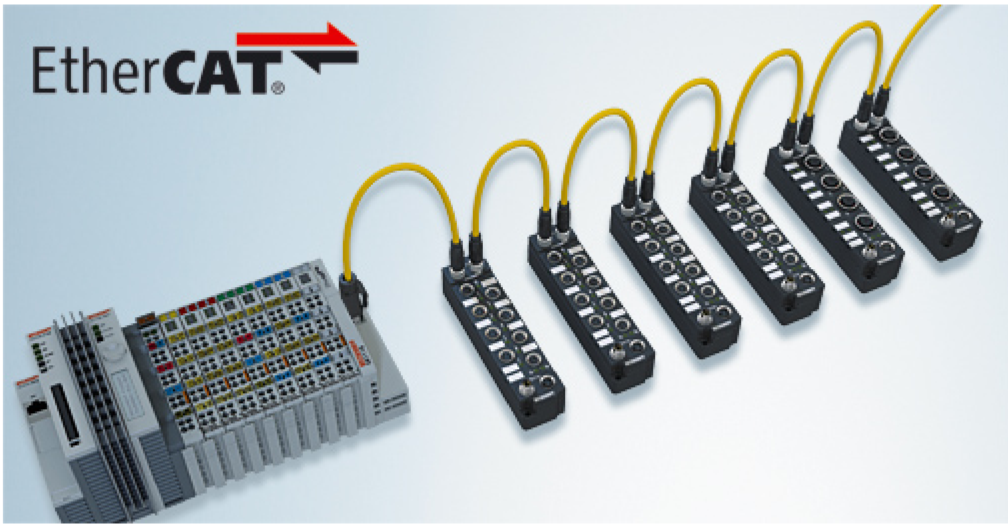


Fig. 1: EtherCAT Box Modules within an EtherCAT network

The robust design of the EtherCAT Box modules enables them to be used directly at the machine. Control cabinets and terminal boxes are now no longer required. The modules are fully sealed and therefore ideally prepared for wet, dirty or dusty conditions.

Pre-assembled cables significantly simplify EtherCAT and signal wiring. Very few wiring errors are made, so that commissioning is optimized. In addition to pre-assembled EtherCAT, power and sensor cables, field-configurable connectors and cables are available for maximum flexibility. Depending on the application, the sensors and actuators are connected through M8 or M12 connectors.

The EtherCAT modules cover the typical range of requirements for I/O signals with protection class IP67:

- digital inputs with different filters (3.0 ms or 10 μs)
- digital outputs with 0.5 or 2 A output current
- analog inputs and outputs with 16 bit resolution
- Thermocouple and RTD inputs
- Stepper motor modules

XFC (eXtreme Fast Control Technology) modules, including inputs with time stamp, are also available.

Fig. 2: EtherCAT Box with M8 connections for sensors/actuators

Fig. 3: EtherCAT Box with M12 connections for sensors/actuators

Basic EtherCAT documentation

You will find a detailed description of the EtherCAT system in the Basic System Documentation for EtherCAT, which is available for download from our website (www.beckhoff.com) under Downloads.

EtherCAT XML Device Description

You will find XML files (XML Device Description Files) for Beckhoff EtherCAT modules on our website (www.beckhoff.com) under Downloads, in the Configuration Files area.

2.2 EP3632-0001 - Introduction

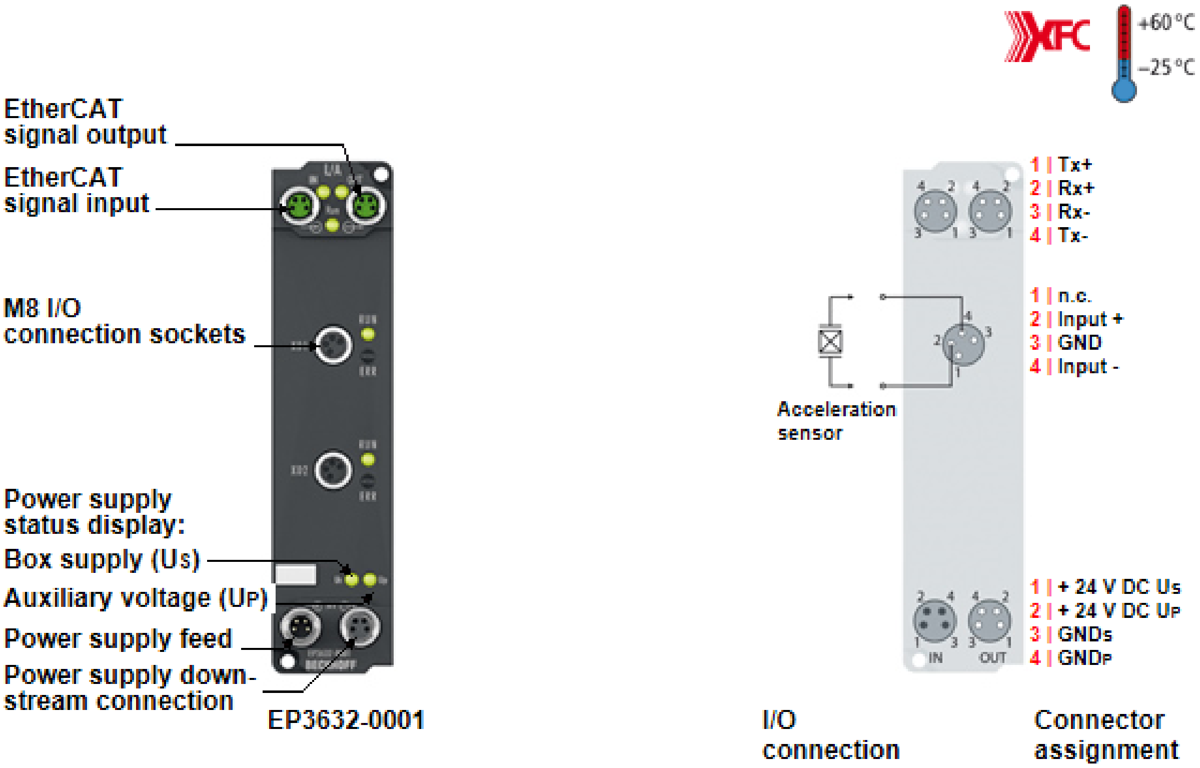


Fig. 4: EP3632-0001

The EP3632 EtherCAT Box is a 2-channel oversampling interface for up to two IEPE sensors with 2-wire connection. The current of the integrated constant current source can be set to 2 mA, 4 mA or 8 mA separately for each channel depending on the sensor and cable length.

The input signal is sampled according to the oversampling principle with up to 50 ksamples per second per channel. The measured values can be correlated to other parts of the system through distributed clocks. Except for filtering, the EP3632 does not pre-process the vibration amplitude values. This is done by the retrieving controller. The interface can be adapted to application-specific requirements by means of adjustable filters and supply currents.

The TwinCAT 3 Condition Monitoring library offers extensive algorithms for signal evaluation by the controller, enabling full utilization of the performance and flexibility benefits of the PC platform.

2.3 Technical data

Technical data	EP3632-0001
Technology	Condition Monitoring/IEPE
Feldbus	EtheCAT
Fieldbus connection	2 x M8 socket, (green)
Number of inputs	2
Connection of inputs [▸ 28]	2 x M8 socket
Signal voltage	IEPE constant current supply and recording of modulated AC voltage
Sensor state monitoring	yes, through monitoring of the bias voltage
Measuring range	default ±5 V up to 25 kHz, ±250 mV up to 10 Hz
Input filter limit frequency	analog parameterisable 5th order low-pass filter up to 25 kHz, typically 0.05 Hz high-pass filter
Measuring error	<± 0.5 % (DC; relative to full scale value)
Resolution	16 bit (incl. sign)
Nominal voltage	24 V_{DC} (-15 %/+20 %)
Conversion time	20 µs (max. 50 kSamples/s)
Supply current I_{EXCITE}	typ. 2/4/8 mA (separately configurable for both channels)
Supply of module electronics	from control voltage Us
Current consumption of module electronics	typ. 120 mA
Sensor supply	from control voltage U_S
Connection for power supply	Feeding: 1 x M8 plug, 4-pin Onward connection: 1 x M8 socket, 4-pin
Distributed clocks	yes
Special features	automatic anti-aliasing function, wire breakage detection
Electrical isolation	Control voltage / fieldbus: yes
Weight	app. 165 g
Permissible ambient temperature range during operation	-25°C ... +60°C
Permissible ambient temperature range during storage	-40°C ... +85°C
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP65, IP66, IP67 (conforms to EN 60529)
Installation position	any
Approval	CE

2.4 Basic function principles

2.4.1 Vibration analysis

Vibration analysis refers to deriving of information from existing (mechanical) vibrations e.g. in machines or certain production processes. It can take place during Condition Monitoring, e.g. for drives, punching and pressing tools, in production processes such as balancing of rotating parts, or during other movements such as torsion of towers/wind turbines.

The table below illustrates typical acceleration values for natural and technical processes.

Typical acceleration values

Machine or event	Typical g-value
Commercial aircraft (take-off)	≈ 0,5
Formula 1 car (start)	≈ 1 – 1,5
Commercial aircraft (turning, max.)	≈ 2,5
Pendulum at 90° amplitude	≈ 2
Space Shuttle during journey into orbit	max. 3 (exact)
Space Shuttle during re-entry into the Earth's atmosphere	max. 1.6
Typical roller coaster ride (max.)	4 (6)
Formula 1 car (cornering, max.)	≈ 4 – 5
Circular looping (base)	≥ 6
Combat aircraft/aerobatics (max.)	≈ 10 (13,8)
Ejector seat	15 – 20
Car back-rest breaks at	≈ 20
Head-on car collision	up to ≈ 50
Car passenger compartment during crash	max. 120
Survived by a human	≈ 180
Hard fist stroke	up to ≈ 100
Raindrop hitting the eye	up to ≈ 150
Ball pen hitting hard floor from 1 m height	order of magnitude 1000
Hard disk falling on hard floor from 1 m height (without deformation of the floor)	10,000 or more
Laboratory centrifuge	≈ 10.000
Ultra centrifuge	≈ 100.000
Rifle bullet during firing	≈ 100.000
Spike during ejection from a nettle cell	5.410.000
Nuclear bomb explosion (bomb case)	up to ≈ 10^{11}
Neutron star surface	≈ $2 \cdot 10^{11}$

2.4.2 Application of condition monitoring

Condition monitoring can be used to glean information on the state of rotating/moving parts through measurement of vibrations at machines/drives/gears and subsequent analysis with suitable mathematical tools (e.g. TwinCAT library, FFT, custom user programs).

The existing vibration is recorded continuously or at longer, regular time intervals and finally compared with a setpoint value/initial value (Fig. *Sample of ball bearing damage and subsequent analysis*).
In this way any damage can be detected at an early stage. Instead of changing components preventively on a regular basis or waiting for sudden damage and subsequent expensive downtime and possible consequential damage, repairs and downtimes become plannable. Needless failures, consequential damage or prematurely and costly replacement of intact parts can be avoided.

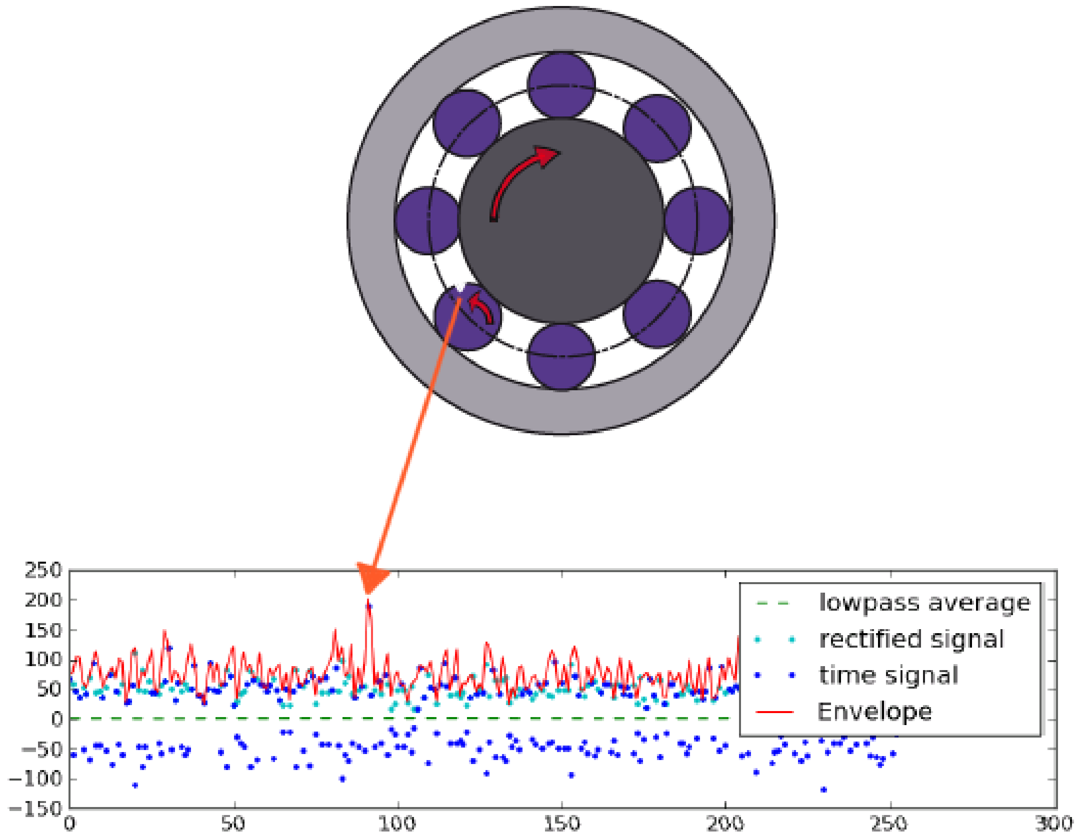


Fig. 5: Example of ball bearing damage and subsequent analysis

In the example above the so-called envelope enables analysis of shock pulses resulting from unevenness in the roller bearing. The defective element can be identified based on the periodicity (envelope spectrum).

2.4.3 Output signals of IEPE sensors

Vibrations can be recorded with IEPE (Integrated Electronics Piezo Electric) sensors, for example. The advantage of this technology is an integrated amplifier as impedance transformer, so that only a simple two-wire connection (coax) is required for the low-impedance output signal. IEPE sensors are typically supplied with 2…20 mA constant current. In inactive state they produce a constant DC bias voltage (zero voltage/ U_{bias}) typically 7…14 V. Depending on the acceleration of the sensor, an analog AC voltage generated proportionally to the movement is added to the sensor's U_{bias} ; e.g. a 50 Hz sinusoidal deflection with an amplitude of 1 g (= 9.81 m/s^2) produces a 50 Hz sinusoidal output voltage of AC +/-50 mV + U_{bias} in the case of a sensor with a sensitivity of 50 mV/g (Fig. *Output signal of an IEPE sensor (sample)*). The maximum output signal of a sensor is usually AC +/-5 V (+ U_{bias}).

Increasing cable length results in increasing cable capacitance (typically 100 pF/m), so that the control capability of the integrated amplifier drops with increasing signal frequency. This can be partly compensated by increasing the supply current (Fig. *Control capability of the IEPE impedance transformer depending on cable capacitance and supply current*).

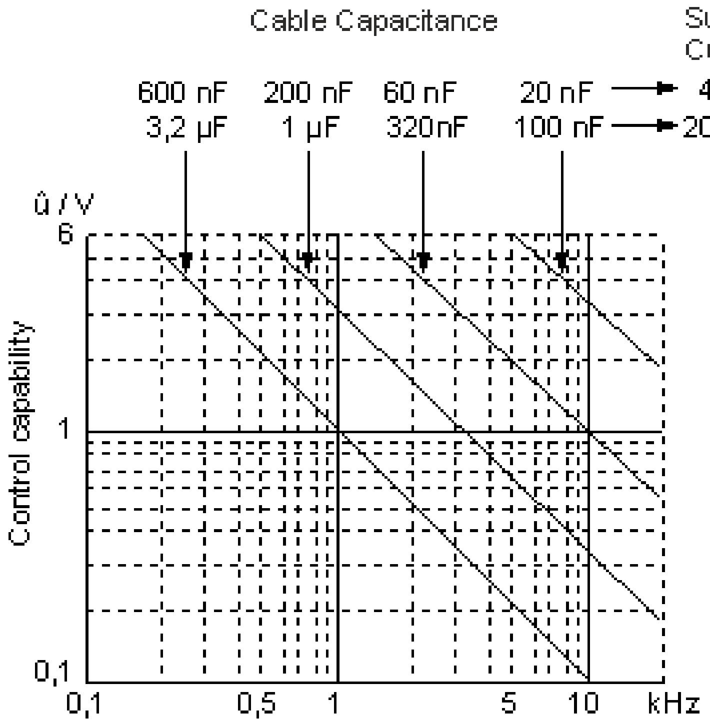


Fig. 6: Control capability of the IEPE impedance transformer depending on cable capacitance and supply current.

The basic properties of IEPE acceleration sensors are described by various parameters such as sensitivity (e.g. 50 mV/g), measuring range (e.g. +/-100 g), +/-3 dB frequency range (under 1 Hz to several kHz), current consumption (2…20 mA), bias voltage etc. The figure *Frequency response of an acceleration sensor* shows an example of a frequency response (amplitude of the output signal in relation to the frequency).

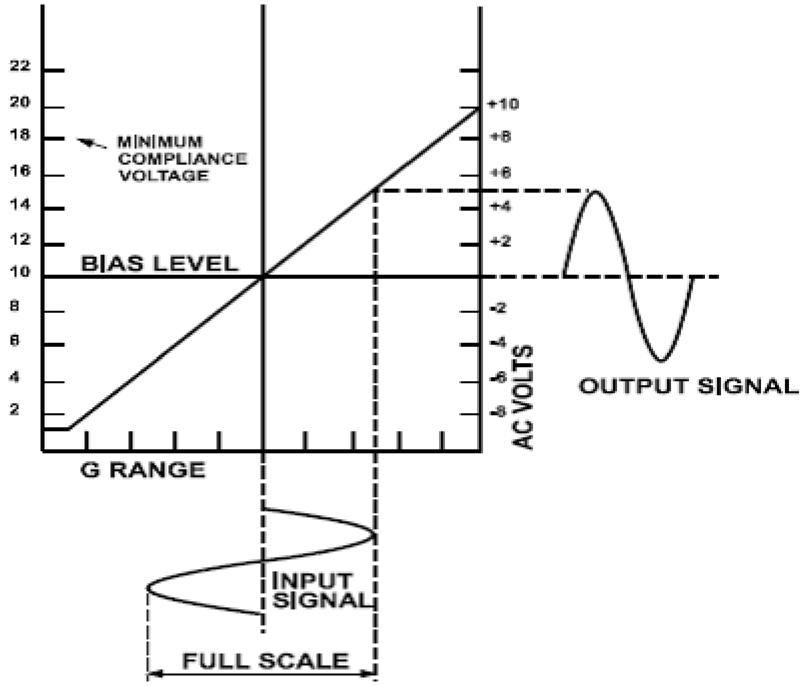


Fig. 7: Output signal of an IEPE sensor (sample)

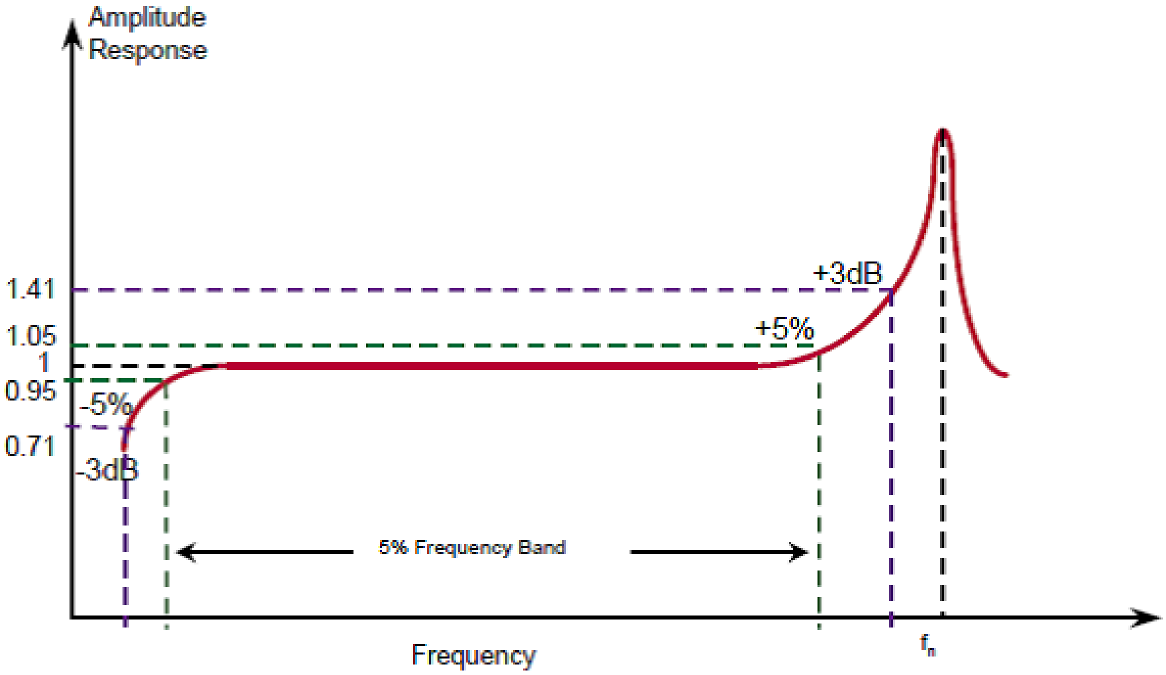


Fig. 8: Frequency response of an acceleration sensor

Other proprietary names for this electrical technique include DeltaTron®, Isotron™, ICP™, Piezotron™ or CCLD

Note: At their core IEPE sensors usually have quartz crystals, which experience small charge changes due to the motion. Measuring and transporting these over several meters requires complex cable installations and measuring instruments for charge amplification. For cost-effective and robust application in industrial environments the solution is to integrate a simple charge amplifier as impedance transformer in the sensor. This enables it to generate the above-mentioned voltage signal and transport the vibration information over larger distance (several tens of meters).

2.4.4 Basic principles of IEPE technology

IEPE ("Integrated Electronics Piezo-Electric") is the standardized name for an analog electrical interface between piezoelectric sensors and electronic analysis equipment. Different manufacturers have developed their own brand names, such as ICP®, CCLD®, Isotron®, DeltaTron®, Piezotron®.

Application

Piezoelectric sensors are usually based on a quartz, in which an electrical charge is shifted under mechanical load. The charge can be detected as a voltage if the measurement is made with sufficiently high impedance. This is preferably a static process << 10 seconds, since otherwise the charge difference is dissipated through external or internal derivative. Such a sensor is therefore less suitable for static long-term loads, such as weighing a silo. Such sensors tend to be used for all kinds of vibration measurements (unbalance detection, sound signals via microphones up to ultrasound, mechanical vibrations, foundation monitoring, etc.).

Over the decades, two electrical forms have developed:

- Direct charge output
- IEPE output

Charge output

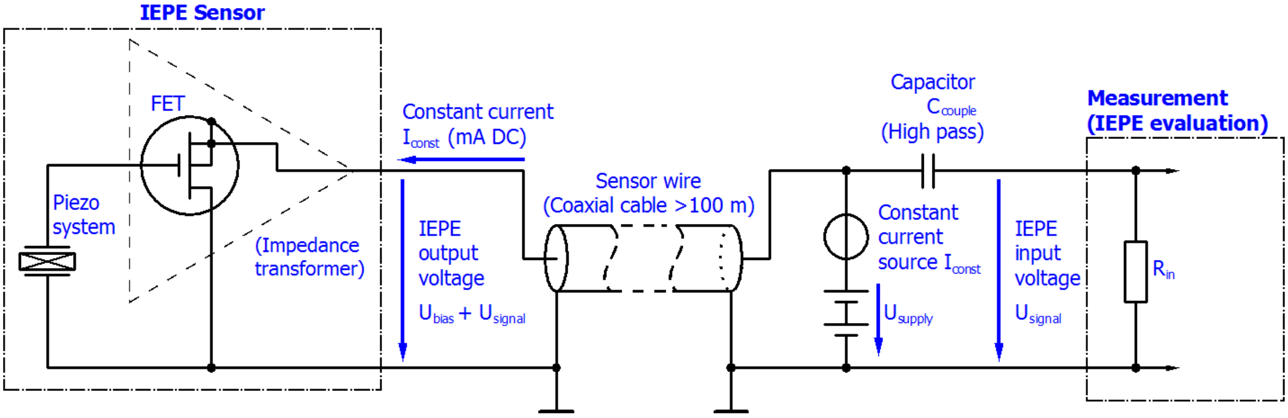
If the voltage information is obtained directly via a (short) two-pole cable, the electronic measuring system must be a so-called charge amplifier.

Advantage: the sensor can be exposed to high temperatures > 150 °C; no power supply is required.

Disadvantage: very sensitive to external influences on the line; elaborate receiving electronics and cable due to high source impedance.

IEPE output

Since the load interface is not generally accepted for industrial applications, a more robust transfer method was sought at an early stage. To this end, IEPE integrates a FET component directly in the sensor.



If this sensor is supplied with constant current between 2 and 8 mA (DC) via the two-pole cable, the resulting bias voltage is approx. 8 to 15 V. If the sensor is now subjected to an accelerating force, the FET changes its internal resistance by the present voltage on its gate. Since the outer constant current source tries to maintain its current, the bias voltage changes within a range of several volts, according to the mechanical load. Although the evaluation unit has to supply the DC current, it can derive the vibration acceleration from the back-measured voltage, even over longer distances.

Advantage: robust system, which is suitable for installation in industrial conditions.

Disadvantage: upper temperature limit for the sensor 150 to 200 °C, smaller dynamic range.

Notes on constant current

- The higher the feed current, the more the vibration sensor heats up. This can be disadvantageous. Note the information provided by the sensor manufacturer.
- The higher the feed current, the higher the maximum transferable signal frequency, since the charge supply/discharge can be handled more quickly on the cable.
- The higher the supply current, the higher the resulting bias voltage. As a result, the transfer may become more robust against electromagnetic influence, but on the other hand large positive amplitudes may enter the upper saturation.

Notes on the IEPE measuring device

- In some IEPE measuring devices the supply current can be switched off (0 mA), so that they can be used for voltage measurements. Cf. for example Beckhoff ELM3604
- Since normally only AC signals are of interest in the vibration range, IEPE evaluations have an electrical high-pass > 10 Hz on the input side. Depending on the application e.g. slow tower vibrations, the limit of this high-pass may be relevant. Cf. for example the configurable high-pass of the ELM3604, which can be switched off.
- The bias voltage is suitable for detection of wire breakage/short circuit. E.g. see also ELM3604 diagnostics options.

3 Mounting and cabling

3.1 Mounting

3.1.1 Dimensions

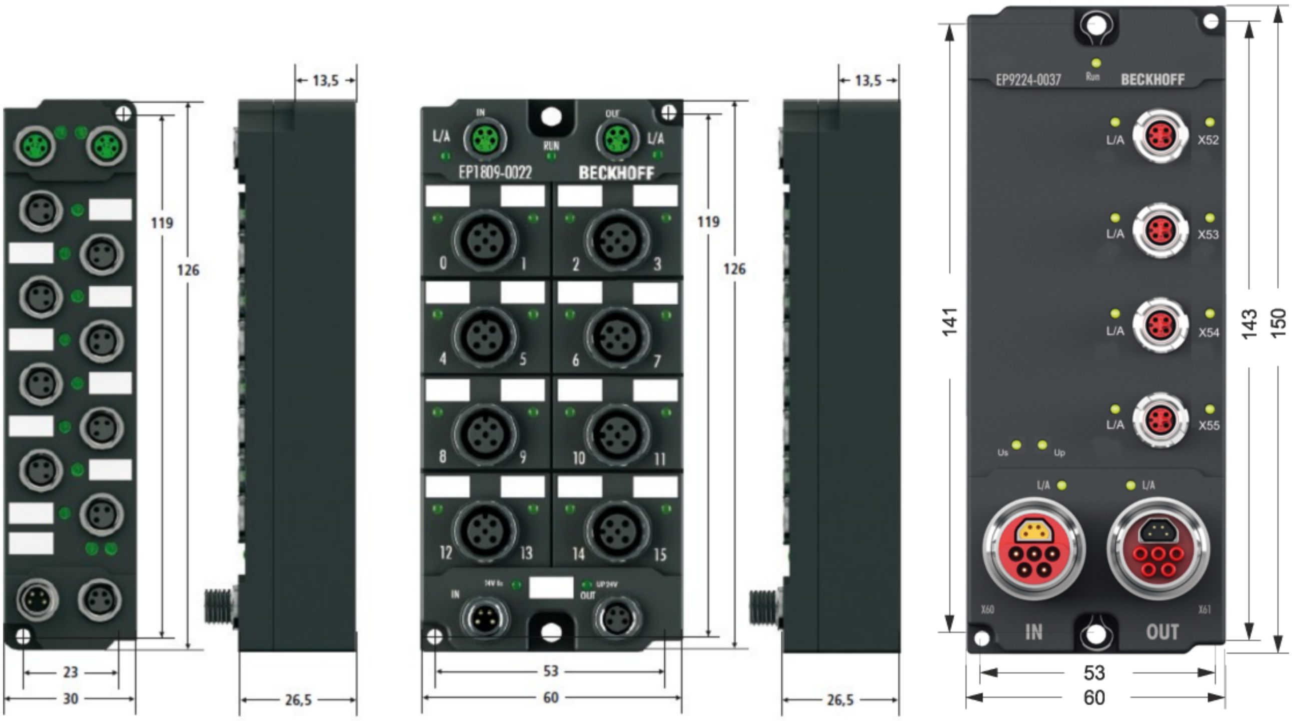


Fig. 9: Dimensions of the EtherCAT Box Modules

All dimensions are given in millimeters.

Housing properties

EtherCAT Box	lean body	wide bodies
Housing material	PA6 (polyamide)	
Casting compound	Polyurethane	
Mounting	two fastening holes Ø 3 mm for M3	two fastening holes Ø 3 mm for M3 two fastening holes Ø 4.5 mm for M4
Metal parts	Brass, nickel-plated	
Contacts	CuZn, gold-plated	
Power feed through	max. 4 A (M8) max. 16 A (7/8“) max. 15.5 A (B17 5G 1.5 mm^2)	
Installation position	variable	
Protection class	IP65, IP66, IP67 (conforms to EN 60529) when screwed together	
Dimensions (H x W x D)	app. 126 x 30 x 26.5 mm	app. 126 x 60 x 26,5 mm app. 150 x 60 x 26.5 mm (without 7/8", B17)

3.1.2 Nut torque for connectors

M8 connectors

It is recommended to pull the M8 connectors tight with a nut torque of **0.4 Nm**. When using the torque control screwdriver ZB8800 is also a max. torque of **0.5 Nm** permissible.

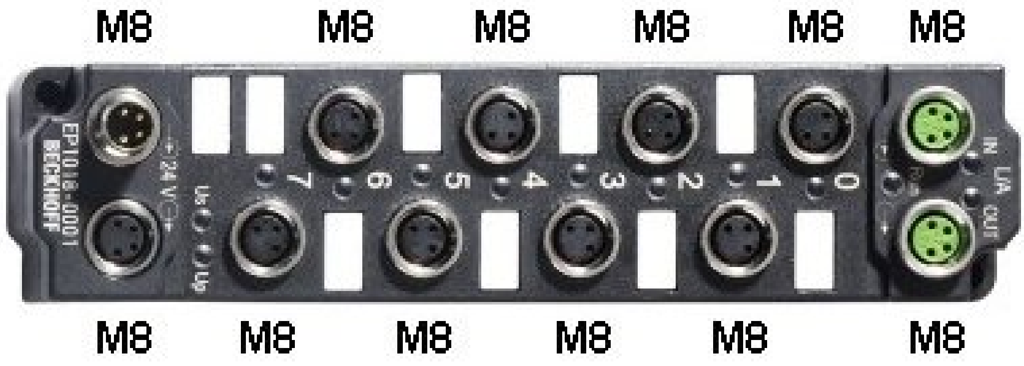


Fig. 10: EtherCAT Box with M8 connectors

M12 connectors

It is recommended to pull the M12 connectors tight with a nut torque of **0.6 Nm**.

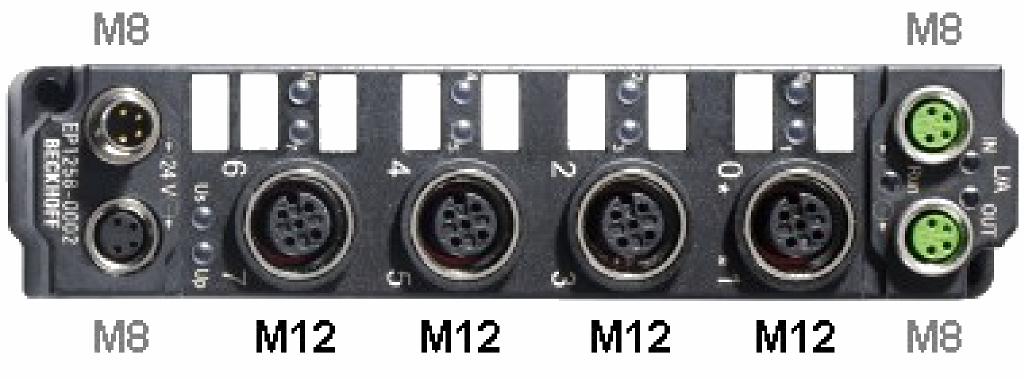


Fig. 11: EtherCAT Box with M8 and M12 connectors

7/8" plug connectors

We recommend fastening the 7/8" plug connectors with a torque of **1.5 Nm**.

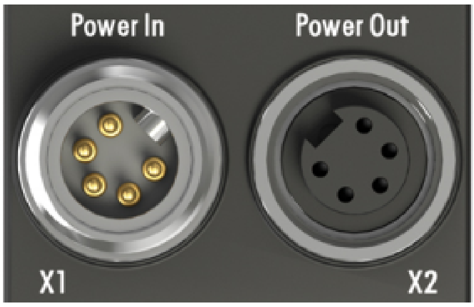


Fig. 12: 7/8" plug connectors

Torque socket wrenches

Fig. 13: ZB8801 torque socket wrench

> **Ensure the right torque**
> Use the torque socket wrenches available by Beckhoff to pull the connectors tight (ZB8800, ZB8801-0000)!

3.2 Cabling

3.2.1 EtherCAT connection

For the incoming and ongoing EtherCAT connection,

- the EtherCAT Box (EPxxxx) has two M8 sockets, marked in **green**
- the Coupler Box (FBB-x110) has two M12 sockets

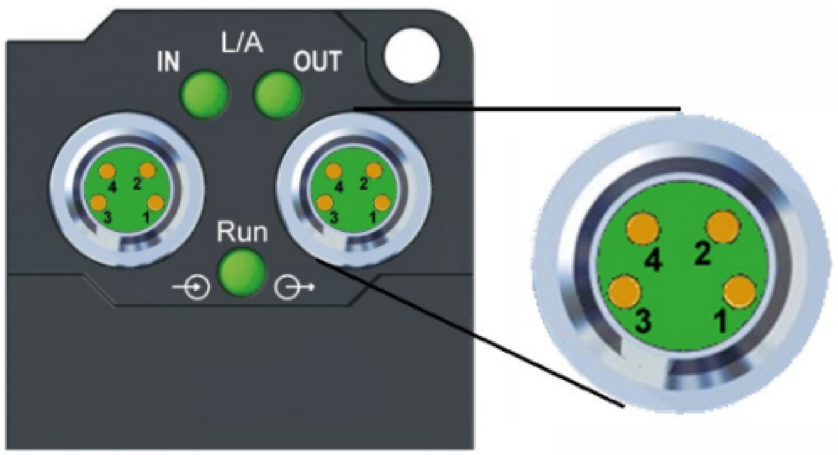


Fig. 14: EtherCAT Box: M8, 30 mm housing

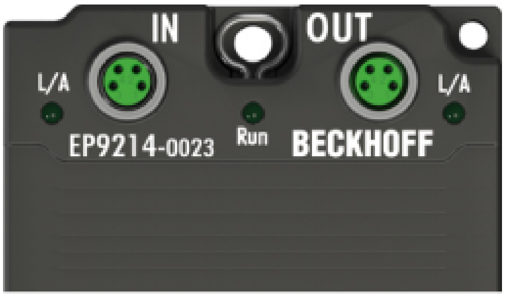


Fig. 15: EtherCAT Box: M860 mm housing (example: EP9214)

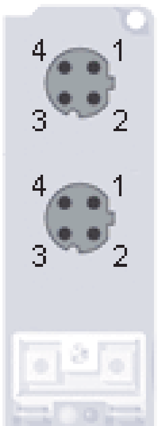


Fig. 16: Coupler Box: M12

Assignment

There are various different standards for the assignment and colors of connectors and cables for Ethernet/ EtherCAT.

Ethernet/EtherCAT		Plug connector			Cable		Standard
Signal	**Description**	**M8**	**M12**	**RJ45**[1]	**ZB9010, ZB9020, ZB9030, ZB9032, ZK1090-6292, ZK1090-3xxx-xxxx**	**ZB9031 and old versions of ZB9030, ZB9032, ZK1090-3xxx-xxxx**	**TIA-568B**
Tx +	Transmit Data+	Pin 1	Pin 1	Pin 1	yellow[2]	orange/white[3]	white/orange
Tx -	Transmit Data-	Pin 4	Pin 3	Pin 2	orange[2]	orange[3]	orange
Rx +	Receive Data+	Pin 2	Pin 2	Pin 3	white[2]	blue/white[3]	white/green
Rx -	Receive Data-	Pin 3	Pin 4	Pin 6	blue[2]	blue[3]	green
Shield	Shield	Housing		Shroud	Screen	Screen	Screen

[1]) colored markings according to EN 61918 in the four-pin RJ45 connector ZS1090-0003
[2]) wire colors according to EN 61918
[3]) wire colors

Assimilation of color coding for cable ZB9030, ZB9032 and ZK1090-3xxxx-xxxx (with M8 connectors)

For unification the prevalent cables ZB9030, ZB9032 and ZK1090-3xxx-xxxx this means the pre assembled cables with M8 connectors were changed to the colors of EN61918 (yellow, orange, white, blue).So different color coding exists. But the electrical properties are absolutely identical.

EtherCAT connector

The following connectors can be supplied for use in Beckhoff EtherCAT systems.

Name	Connector	Comment
ZS1090-0003	RJ45	four-pole, IP20, field-configurable
ZS1090-0004	M12, male	four-pin, IP67, for field assembly
ZS1090-0005	RJ45	eight-pole, IP20, field-configurable, suitable for gigabit Ethernet
ZS1090-0006	M8 plug connector	four-pole, IP67, field-configurable, for cable type ZB903x
ZS1090-0007	M8 socket	four-pole, IP67, field-configurable, for cable type ZB903x
ZS1090-1006	M8 plug connector	four-pole, IP67, field-configurable up to OD = 6.5 mm
ZS1090-1007	M8 socket	four-pole, IP67, field-configurable up to OD = 6.5 mm

3.2.2 EtherCAT - Fieldbus LEDs

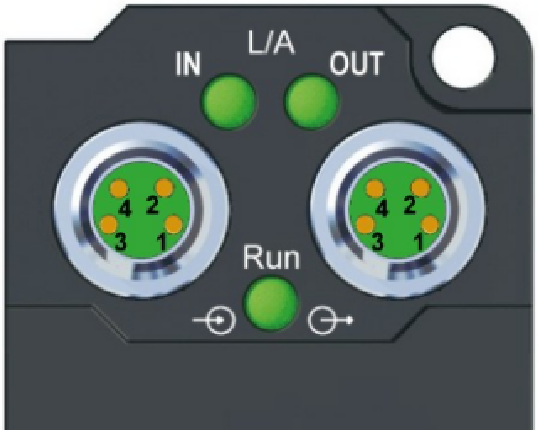


Fig. 17: EtherCAT-LEDs

LED display

LED	Display	Meaning
IN L/A	off	no connection to the preceding EtherCAT module
	Lit	LINK: connection to the preceding EtherCAT module
	flashing	ACT: Communication with the preceding EtherCAT module
OUT L/A	off	no connection to the following EtherCAT module
	Lit	LINK: connection to the following EtherCAT module
	flashing	ACT: Communication with the following EtherCAT module
Run	off	Status of the EtherCAT module is Init
	flashes quickly	Status of the EtherCAT module is pre-operational
	flashes slowly	Status of the EtherCAT module is safe-operational
	Lit	Status of the EtherCAT module is operational

EtherCAT statuses

The various statuses in which an EtherCAT module may be found are described in the Basic System Documentation for EtherCAT, which is available for download from our website (www.beckhoff.com) under Downloads.

3.2.3 Power Connection

The feeding and forwarding of supply voltages is done via two M8 connectors at the bottom end of the modules:

- IN: left M8 connector for feeding the supply voltages
- OUT: right M8 connector for forwarding the supply voltages

Fig. 18: EtherCAT Box, Connectors for power supply

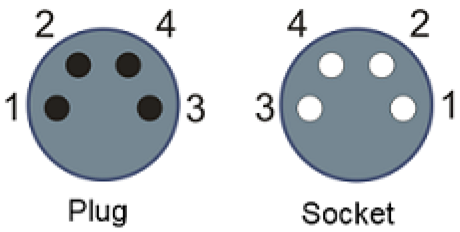


Fig. 19: Pin assignment M8, Power In and Power Out

PIN assignment

Pin	Voltage	
1	Control voltage Us, +24 V_{DC}	
2	Auxiliary voltage Up, +24 V_{DC}	
3	GNDs*	*) may be connected internally to each other depending on the module: see specific module descriptions
4	GNDp*	

The pins M8 connectors carry a maximum current of 4 A.

Two LEDs display the status of the supply voltages.

NOTE

Don't confuse the power connectors with the EtherCAT connectors!

Never connect the power cables (M8, 24 V_{DC}) with the green marked EtherCAT sockets of the EtherCAT Box Modules! This can damage the modules!

Control voltage Us: 24 V_{DC}

Power is supplied to the fieldbus, the processor logic, the inputs and the sensors from the 24 V_{DC} control voltage Us. The control voltage is electrically isolated from the fieldbus circuitry.

Auxiliary voltage Up 24 V_{DC}

The Auxiliary voltage Up supplies the digital outputs; it can be brought in separately. If the load voltage is switched off, the fieldbus functions and the power supply and functionality of the inputs are retained.

Redirection of the supply voltages

The IN and OUT power connections are bridged in the module (not IP204x-Bxxx and IE204x). The supply voltages Us and Up can thus easily be transferred from EtherCAT Box to EtherCAT Box.

NOTE

Pay attention to the maximum permissible current!

Pay attention also for the redirection of the supply voltages Us and Up, the maximum permissible current for M8 connectors of 4 A must not be exceeded!

Supply via EP92x4-0023 PowerBox modules

If the machine requires higher current or if the EtherCAT Box Modules are installed far away from the control cabinet with included power supply, the usage of four cannel power distribution modules EP9214 or EP9224 (with integrated data logging, see www.beckhoff.com/EP9224) is recommended.

With these modules intelligent power distribution concepts with up to 2 x 16 A and a maximum of 2.5 mm² cable cross-section can be realized.

Fig. 20: EP92x4-0023, Connectors for Power In and Power Out

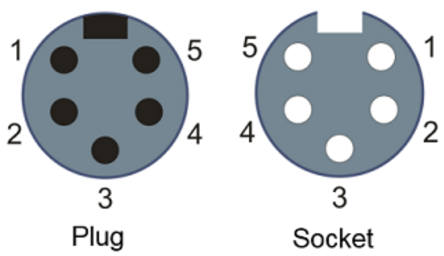


Fig. 21: Pin assignment 7/8”, Power In and Power Out

Electrical isolation

Digital modules

In the digital input/output modules, the grounds of the control voltage (GNDs) and the auxiliary voltage (GNDp) are connected to each other!

Check this at the documentation of each used EtherCAT Box.

Analog modules

In the analog input/output modules the grounds of the control voltage (GNDs) and the auxiliary voltage (GNDp) are separated from each other in order to ensure electrical isolation of the analog signals from the control voltage.

In some of the analog modules the sensors or actuators are supplied by Up - this means, for instance, that in the case of 0...10 V inputs, any reference voltage (0...30 V) may be connected to Up; this is then available to the sensors (e.g. smoothed 10 V for measuring potentiometers).

Details of the power supply may be taken from the specific module descriptions.

NOTE

Electrical isolation may be cancelled!

If digital and analog fieldbus boxes are connected directly via four-core power leads, the analog signals in the fieldbus boxes may be no longer electrically isolated from the control voltage!

3.2.4 Status LEDs for power supply

Fig. 22: Status LEDs for power supply

LED display

LED	Display	Meaning
Us (Control voltage)	off	The power supply voltage Us is not present
	green illuminated	The power supply voltage Us is present
	red illuminated	Because of overload (current > 0.5 A) the sensor supply generated from power supply voltage Us was switched off for all sensors fed from this.
Up (Auxiliary voltage)	off	The power supply voltage Up is not present
	green illuminated	The power supply voltage Up is present

3.2.5 Power cable conductor losses M8

The ZK2020-xxxx-yyyy power cables should not exceed the total length of 15 m at 4 A (with continuation). When planning the cabling, note that at 24 V nominal voltage, the functionality of the module can no longer be assured if the voltage drop reaches 6 V. Variations in the output voltage from the power supply unit must also be taken into account.

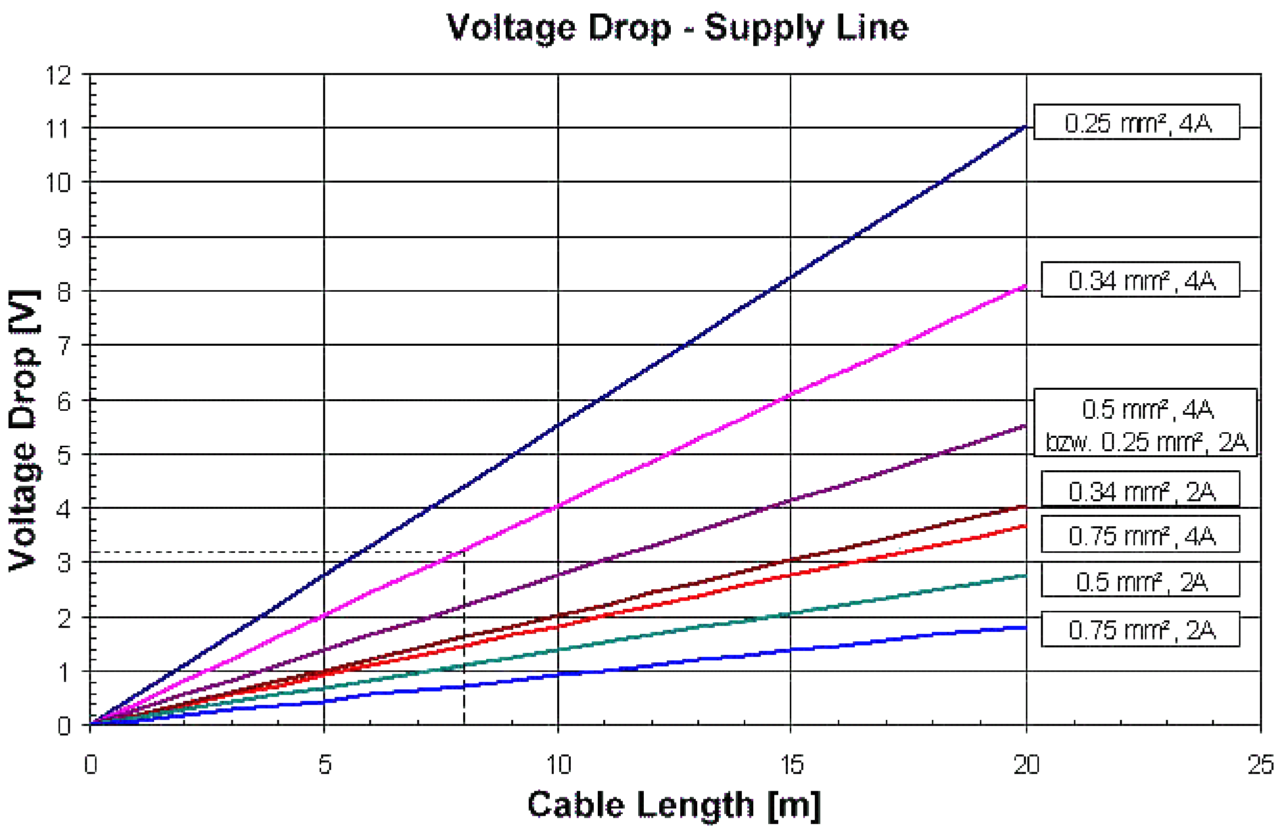


Fig. 23: Power cable conductor losses

Example

8 m power cable with 0.34 mm² cross-section has a voltage drop of 3.2 V at 4 A.

EP92x4 Power Distribution Modules

With EP9214 and EP9224 Power Distribution Modules intelligent concepts for voltage supply are available. Further information may be found under www.beckhoff.com/EP9224.

3.2.6 Signal connection, LEDs

Signal connection IEPE sensors

The X01 and X02 inputs enable connection of up to two IEPE sensors with 2-wire connection.

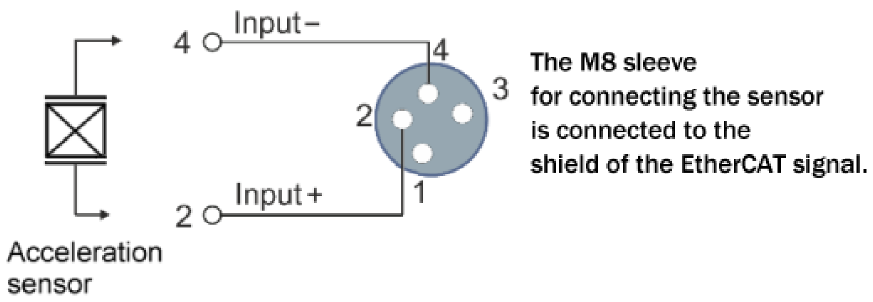


Fig. 24: EP3632 - Signal connection for the X01 and X02 inputs, M8, 4-pin

Meanings on the LEDs

Fig. 25: EP3632 - LEDs

LED	Color	Meaning	
RUN	green	This LED indicates the terminal`s operating state	
		offf	State of the EtherCAT State Machine [▶ 33]: **INIT** = initialization of the terminal
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox-communication and different standard-settings set
		Single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync-Managers [▶ 33] channels and the distributed Clocks. Outputs remain in safe state.
		on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox- and process data communication is possible
Error 1/2	red	Open circuit error	

4 Commissioning/Configuration

4.1 Inserting into the EtherCAT network

Installation of the latest XML device description

Please ensure that you have installed the latest XML device description in TwinCAT. This can be downloaded from the Beckhoff website (http://www.beckhoff.de/english/download/elconfg.htm?id=1983920606140) and installed according to the installation instructions.

At the Beckhoff TwinCAT System Manager the configuration tree can be build in two different ways:

- by scanning [▸ 29] for existing hardware (called "online") and
- by manual inserting/appending [▸ 29] of fieldbus devices, couplers and slaves.

Automatic scanning in of the box

- The EtherCAT system must be in a safe, de-energized state before the EtherCAT modules are connected to the EtherCAT network!
- Switch on the operating voltage, open the TwinCAT System Manager [▸ 32] (Config mode), and scan in the devices (see Fig. 1). Acknowledge all dialogs with "*OK*", so that the configuration is in "FreeRun" mode.

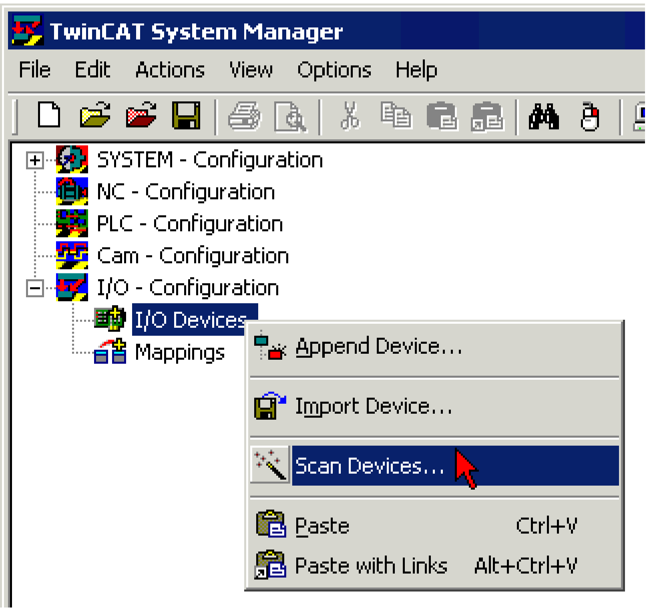


Fig. 26: Scanning in the configuration (I/O Devices -> right-click -> Scan Devices...)

Appending a module manually

- The EtherCAT system must be in a safe, de-energized state before the EtherCAT modules are connected to the EtherCAT network!
- Switch on the operating voltage, open the TwinCAT System Manager [▸ 32] (Config mode)
- Append a new I/O device. In the dialog that appears select the device *EtherCAT (Direct Mode),* and confirm with *OK*.

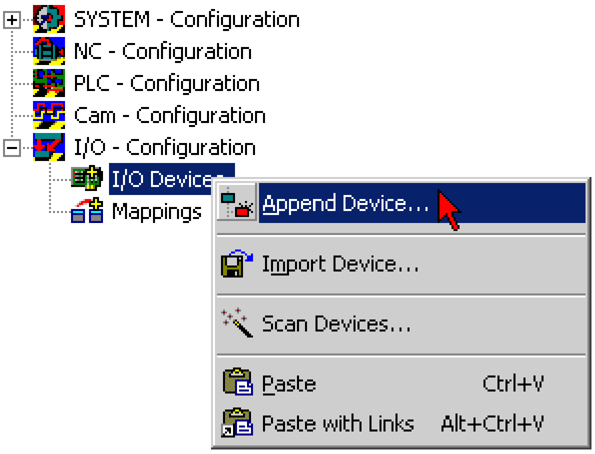


Fig. 27: Appending a new I/O device (I/O Devices -> right-click -> Append Device...)

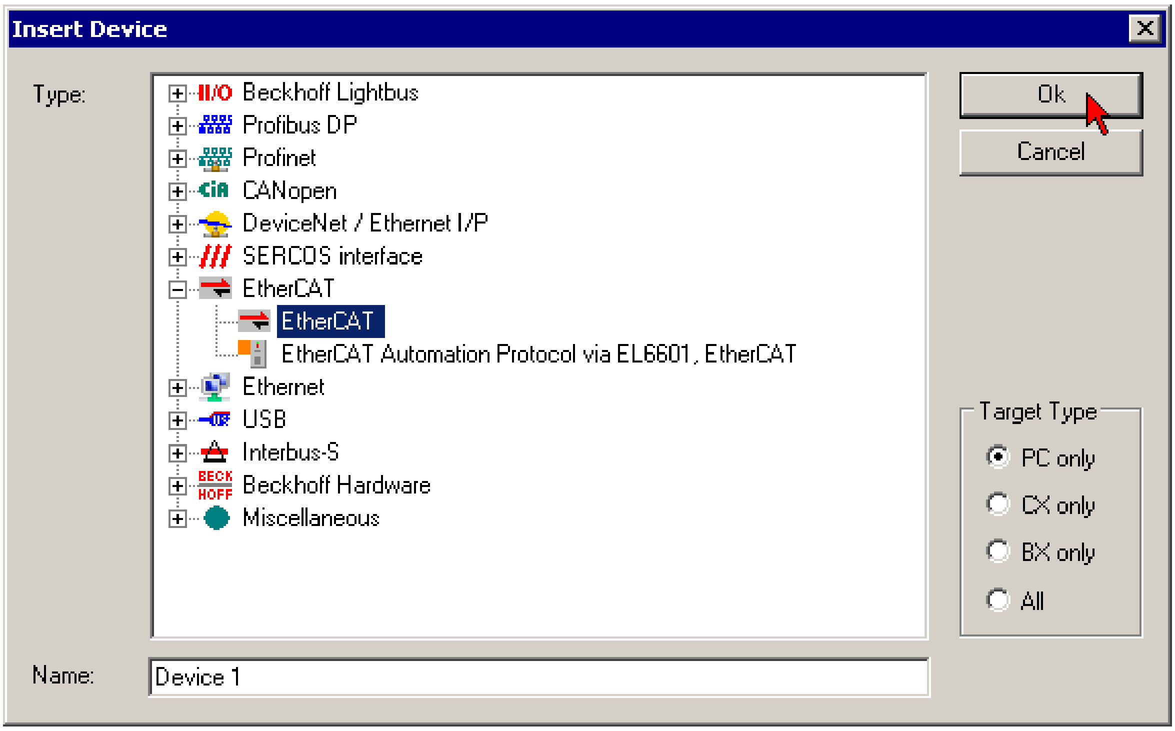


Fig. 28: Selecting the device EtherCAT

- Append a new box.

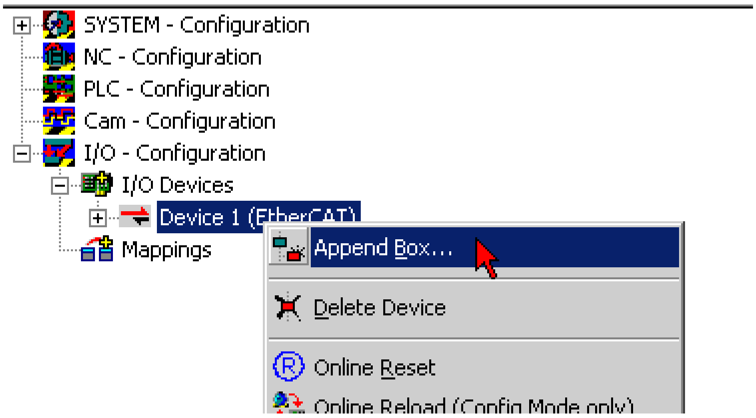


Fig. 29: Appending a new box (Device -> right-click -> Append Box...)

- In the dialog that appears select the desired box (e.g. EP2816-0008), and confirm with *OK*.

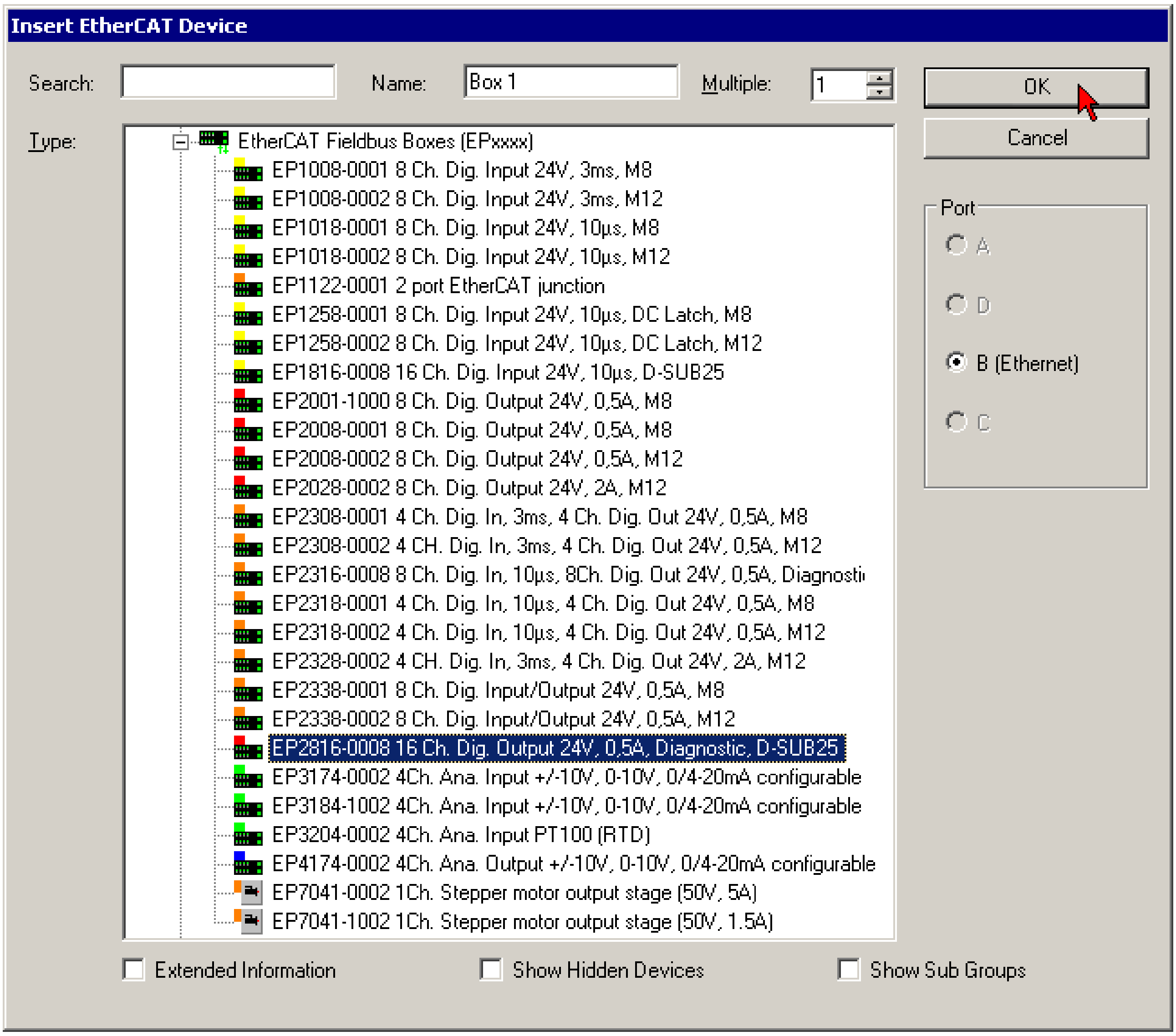


Fig. 30: Selecting a Box (e.g. EP2816-0008)

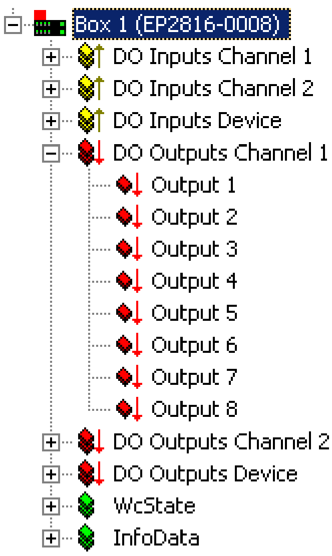


Fig. 31: Appended Box in the TwinCAT tree

4.2 Configuration via TwinCAT

In the left-hand window of the TwinCAT System Manager, click on the branch of the EtherCAT Box you wish to configure (EP2816-0008 in this example).

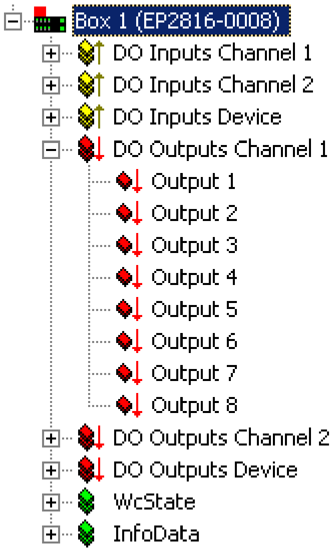


Fig. 32: Branch of the EtherCAT box to be configured

In the right-hand window of the TwinCAT System manager, various tabs are now available for configuring the EtherCAT Box.

General tab

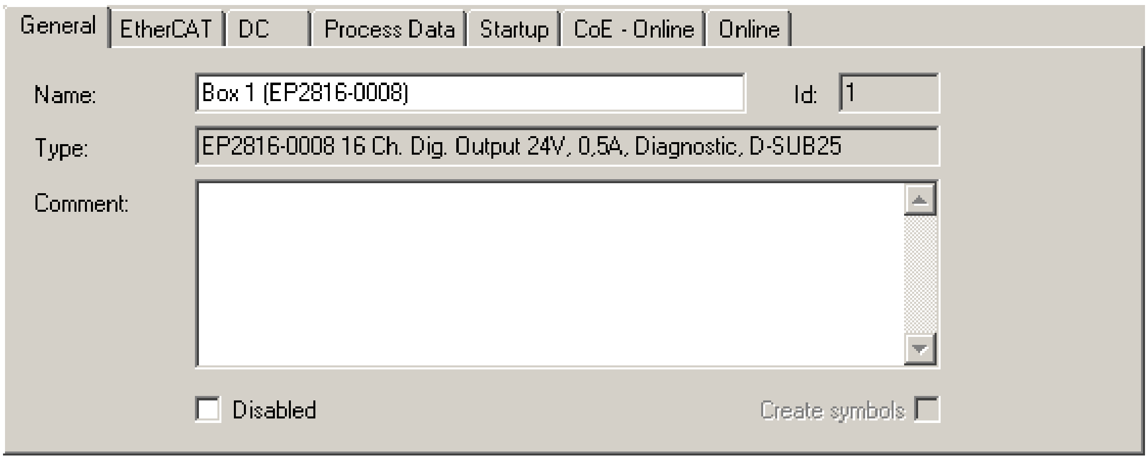


Fig. 33: General tab

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

EtherCAT tab

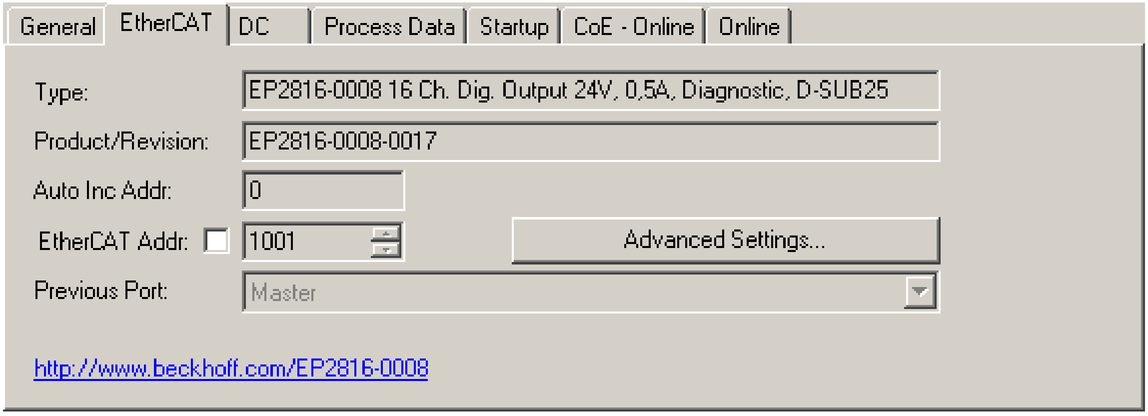


Fig. 34: EtherCAT tab

Type	EtherCAT device type
Product/Revision	Product and revision number of the EtherCAT device
Auto Inc Addr.	Auto increment address of the EtherCAT device. The auto increment address can be used for addressing each EtherCAT device in the communication ring through its physical position. Auto increment addressing is used during the start-up phase when the EtherCAT master allocates addresses to the EtherCAT devices. With auto increment addressing the first EtherCAT slave in the ring has the address 0000_{hex}. For each further slave the address is decremented by 1 ($FFFF_{hex}$, $FFFE_{hex}$ etc.).
EtherCAT Addr.	Fixed address of an EtherCAT slave. This address is allocated by the EtherCAT master during the start-up phase. Tick the checkbox 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 combobox 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 (PDO). The user can select a PDO via PDO assignment and modify the content of the individual PDO via this dialog, if the EtherCAT slave supports this function.

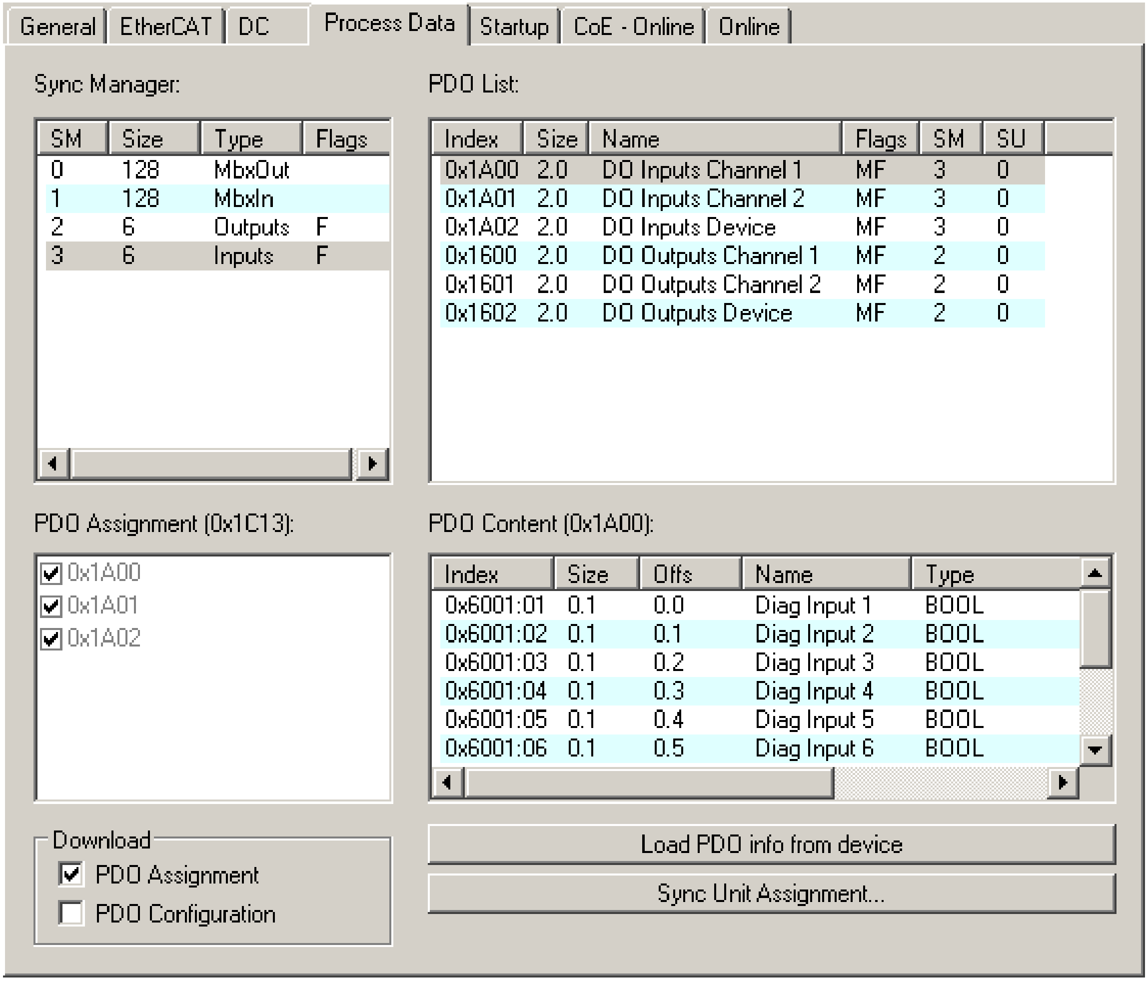


Fig. 35: Process Data tab

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 do select a greyed out PDO, the currently selected PDO has to be deselected first.

Activation of PDO assignment

- the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see Online tab [▸ 38]),
- and the System Manager has to reload the EtherCAT slaves ( button)

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.

<table>
<tr><th>Column</th><th colspan="2">Description</th></tr>
<tr><td>Index</td><td colspan="2">PDO index.</td></tr>
<tr><td>Size</td><td colspan="2">Size of the PDO in bytes.</td></tr>
<tr><td>Name</td><td colspan="2">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.</td></tr>
<tr><td rowspan="2">Flags</td><td>F</td><td>Fixed content: The content of this PDO is fixed and cannot be changed by the System Manager.</td></tr>
<tr><td>M</td><td>Mandatory PDO. This PDO is mandatory and must therefore be assigned to a Sync Manager! Consequently, this PDO cannot be deleted from the PDO Assignment list</td></tr>
<tr><td>SM</td><td colspan="2">Sync Manager to which this PDO is assigned. If this entry is empty, this PDO does not take part in the process data traffic.</td></tr>
<tr><td>SU</td><td colspan="2">Sync unit to which this PDO is assigned.</td></tr>
</table>

PDO Content

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

Download

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

PDO Assignment

If this check box is selected, the PDO assignment that is configured in the PDO Assignment list is downloaded to the device on startup. The required commands to be sent to the device can be viewed in the Startup [▸ 35] 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.

Startup tab

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

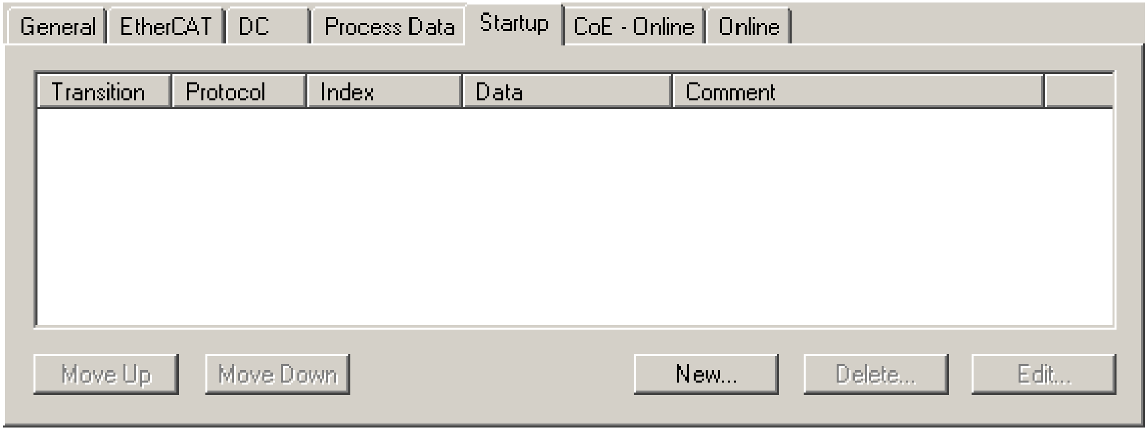


Fig. 36: Startup tab

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

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

CoE - Online tab

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

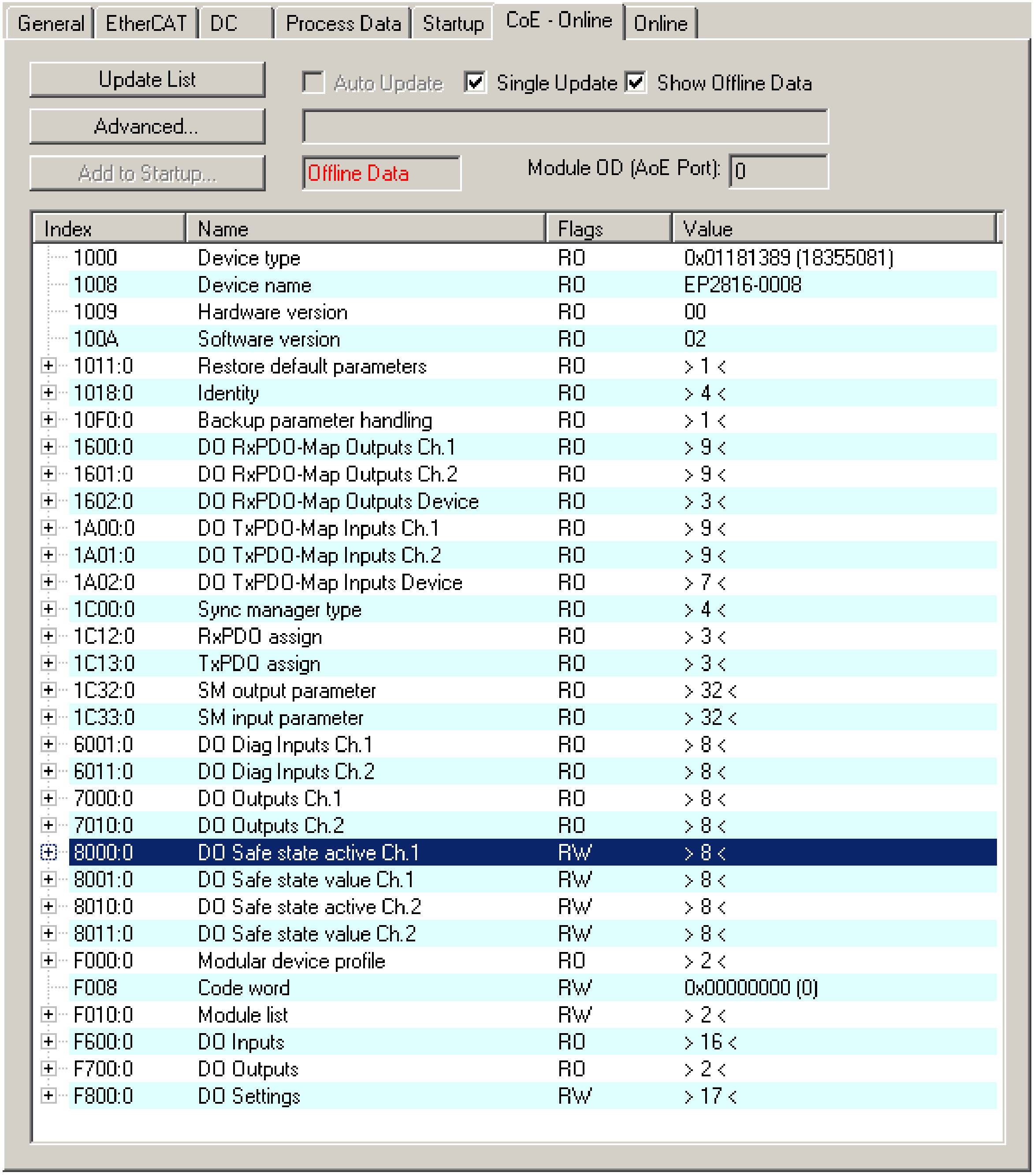


Fig. 37: CoE - Online tab

Object list display

Column	Description	
Index	Index and subindex of the object	
Name	Name of the object	
Flags	RW	The object can be read, and data can be written to the object (read/write)
	RO	The object can be read, but no data can be written to the object (read only)
	P	An additional P identifies the object as a process data object.
Value	Value of the object	

Update List The *Update list* button updates all objects in the displayed list

Auto Update If this check box is selected, the content of the objects is updated automatically.

Advanced The *Advanced* button opens the *Advanced Settings* dialog. Here you can specify which objects are displayed in the list.

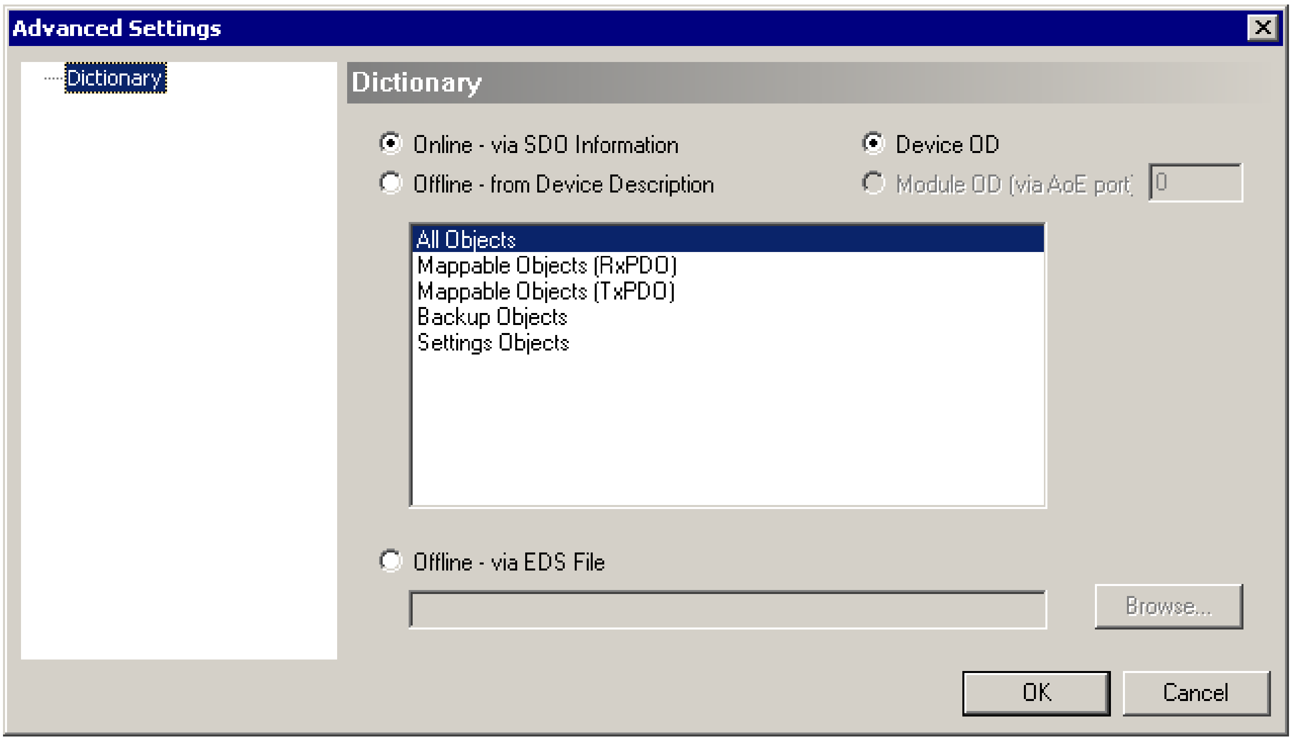


Fig. 38: Advanced settings

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

Online tab

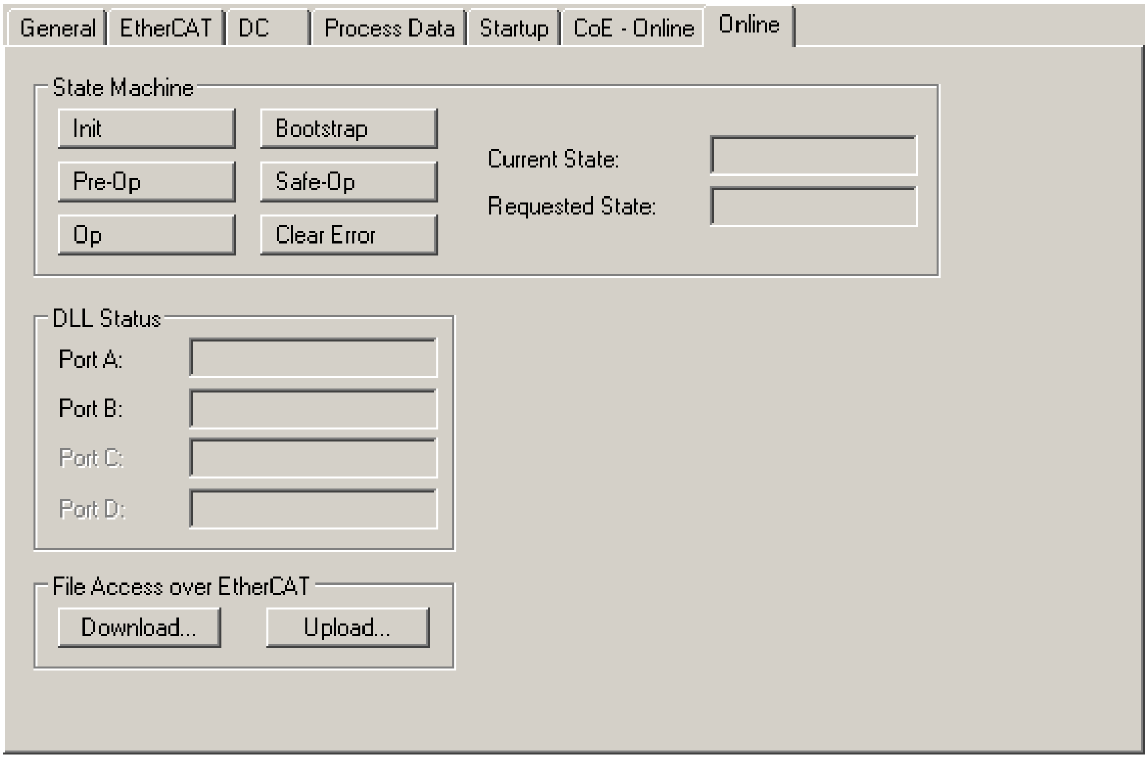


Fig. 39: Online tab

State Machine

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

DLL Status

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

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

File Access over EtherCAT

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

4.3 Oversampling terminals/boxes and TwinCAT Scope

Generally input data of a terminal/box 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/box 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/boxes.

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/box 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.

EL3102

Name	Type	Size
Status	Status_4099	2.0
Value	INT	2.0
Status	Status_4099	2.0
Value	INT	2.0

EL3702

Name	Type	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
Ch2 Value	INT	2.0
Ch2 Value	INT	2.0

Fig. 40: 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.

4.3.1 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
EP3632-0001	all
EPP3632-0001	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/box 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“ has to be added a TwinCAT Measurement – project “Scope YT project” (C) by right click (A) and selection (B) “Add” → “New Project..”. Then “Scope for OS” will be entered as name. The new project just appears within the solution explorer (D).

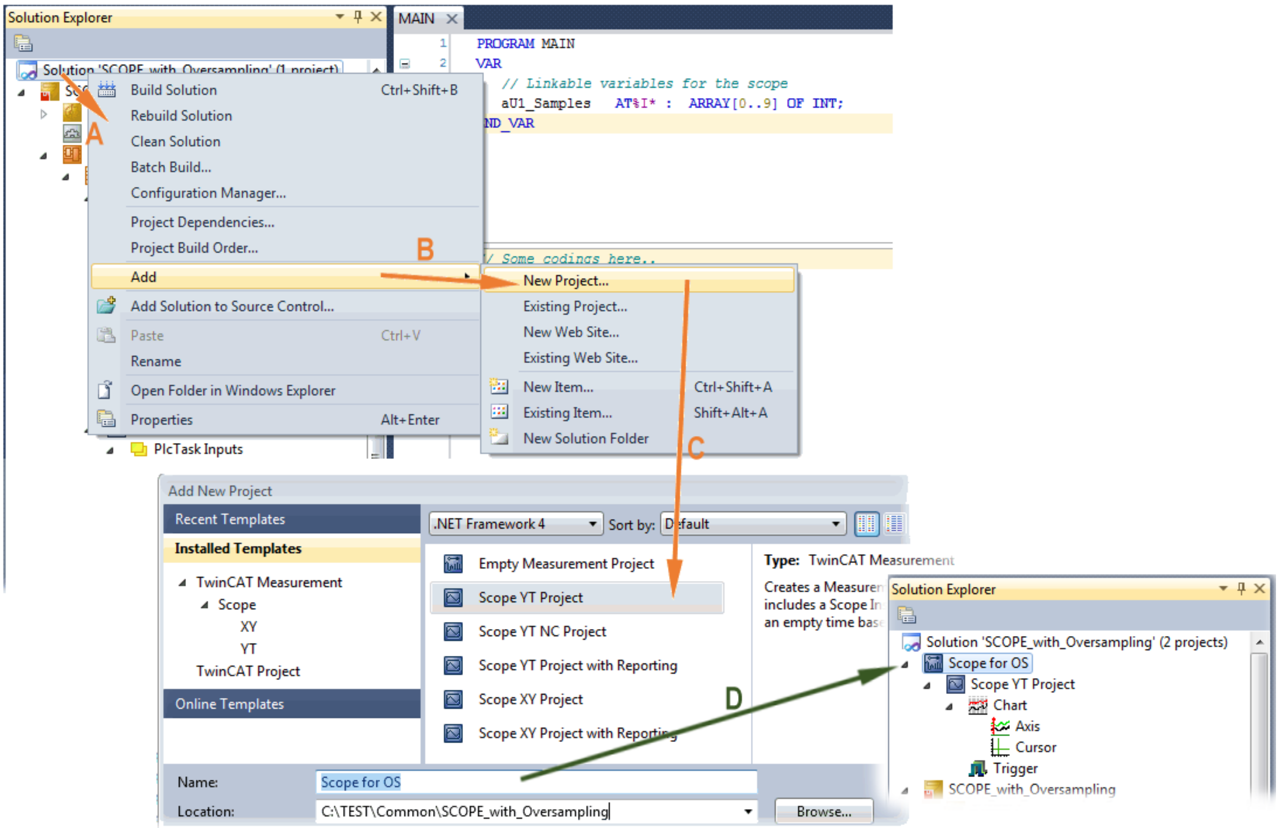


Fig. 41: Adding a Scope project into an already existing project

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/box later. Notice that at least the number of elements has 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:

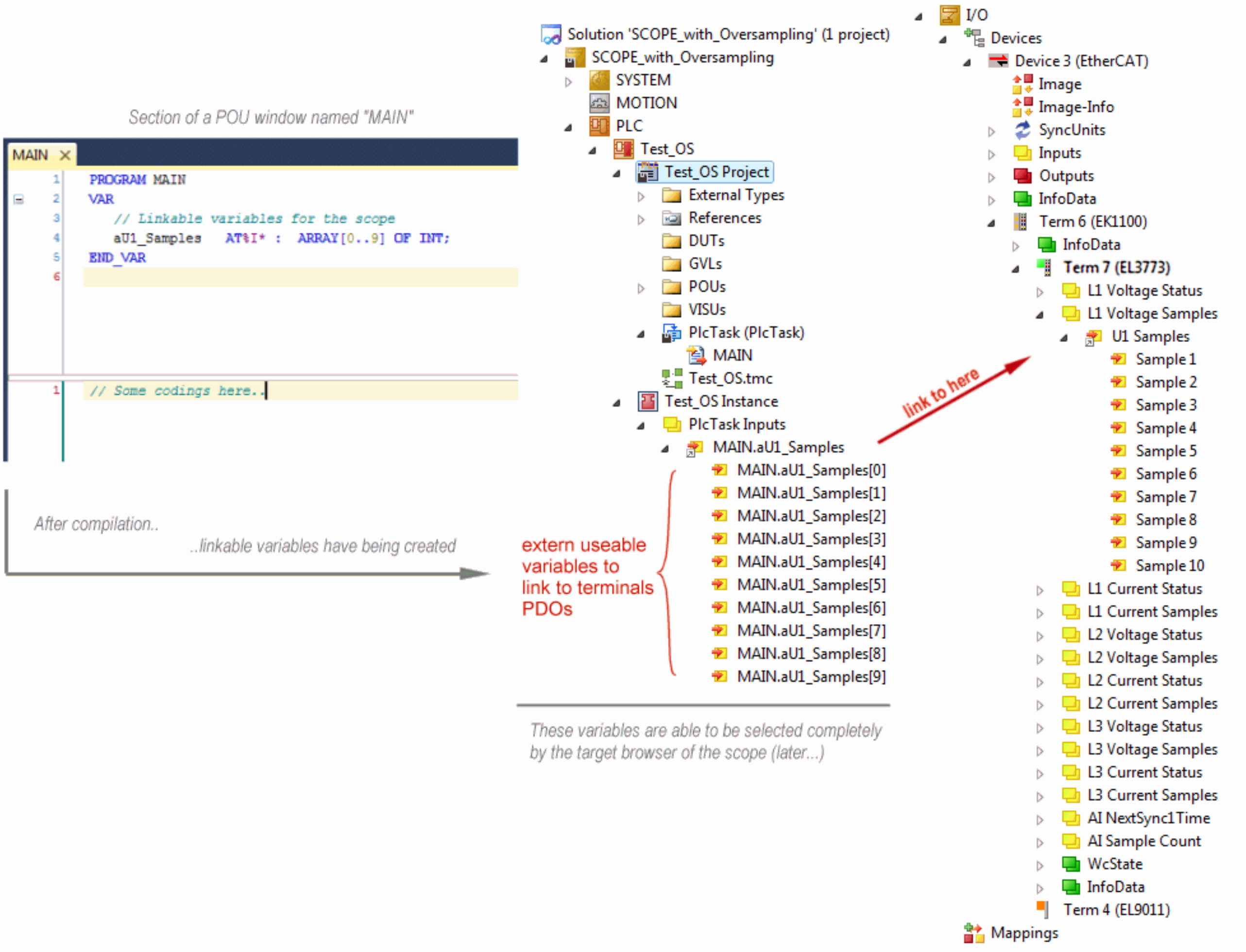


Fig. 42: Representation of a created PLC array variable („aUI_Samples") to link with oversampling PDOs of EL3773

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…".

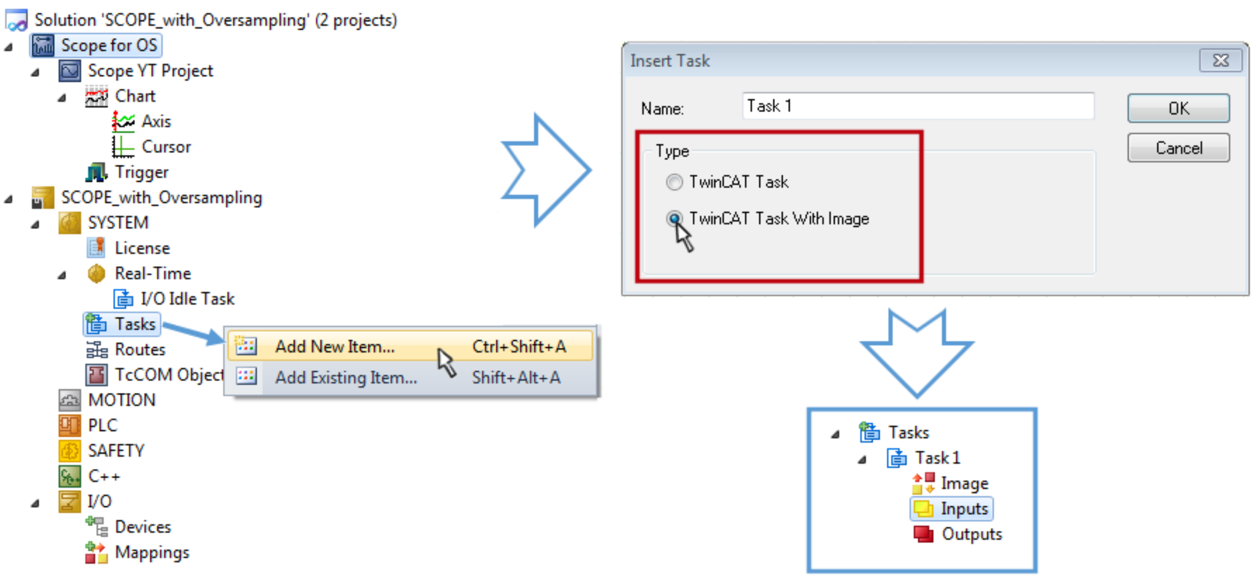


Fig. 43: Insertion of a free task

The Task has 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 has 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:

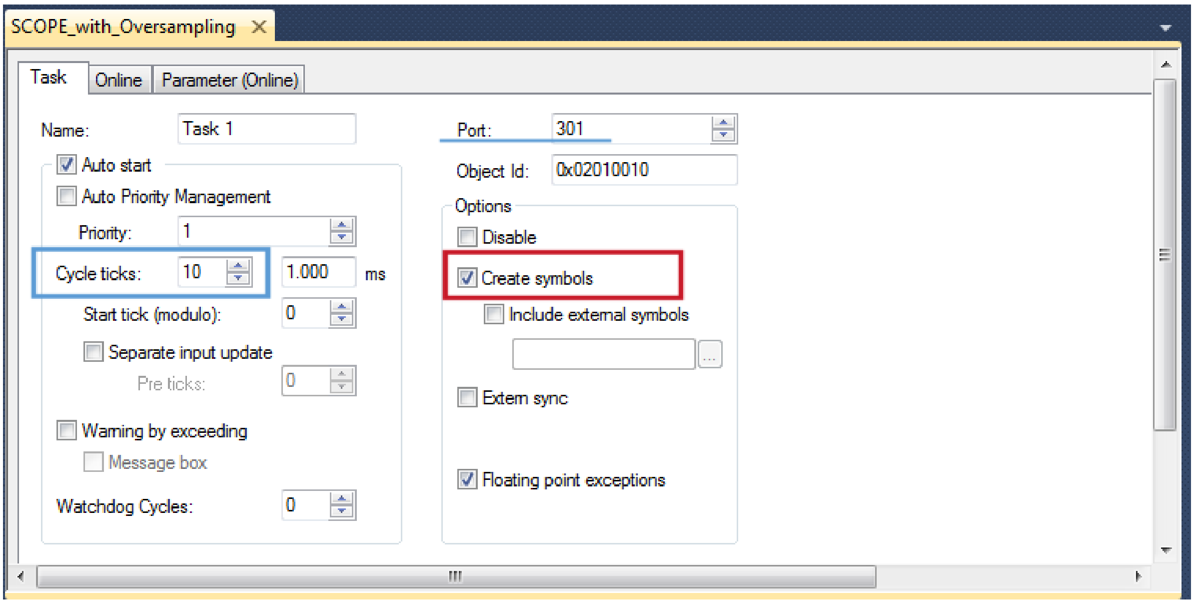


Fig. 44: Task property "Create symbols" must be activated

There’s a default value given for the Port number (301) that should be changed, if necessary. This number has 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:

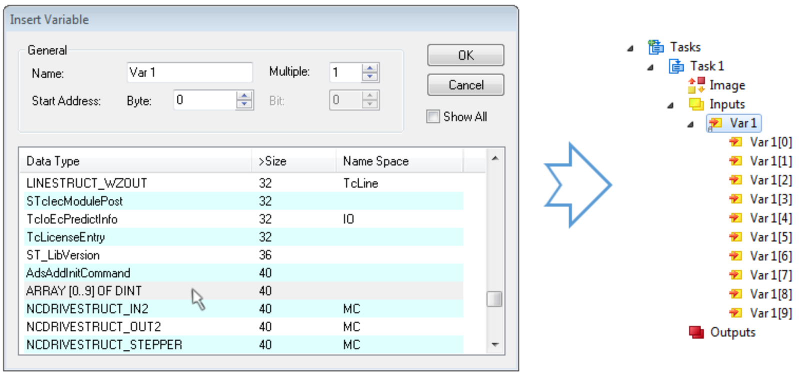


Fig. 45: Insertion of variable "Var 1" fitting to the oversampling (-factor)

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:

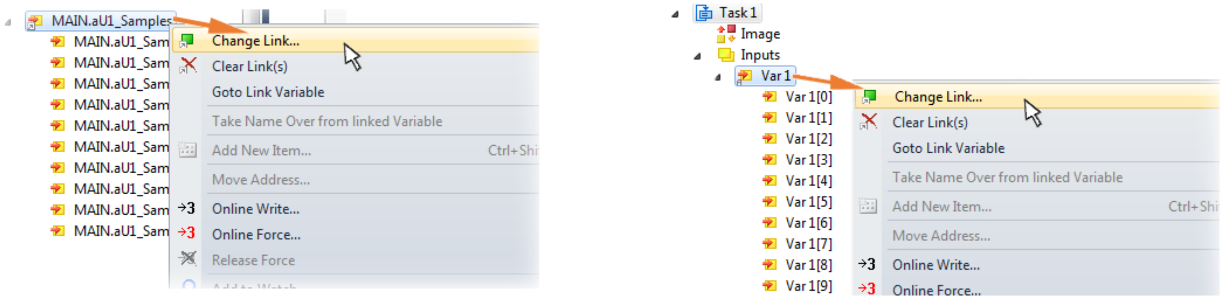


Fig. 46: Set up the link of the PLC array variable (left: for the last preceding paragraph Step 2a, right: for the last preceding paragraph Step 2b)

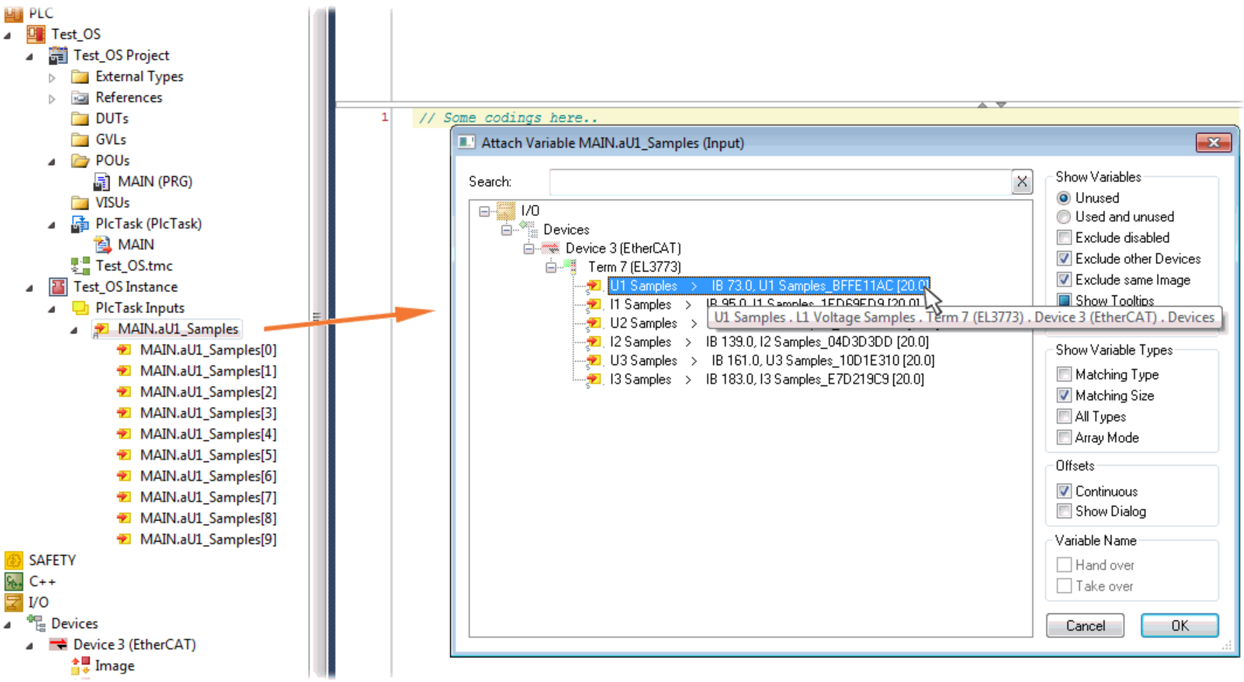


Fig. 47: Select the EL3773 PDO "L1 Voltage Samples" to create a link to the PLC array variable „aUI_Samples“

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):

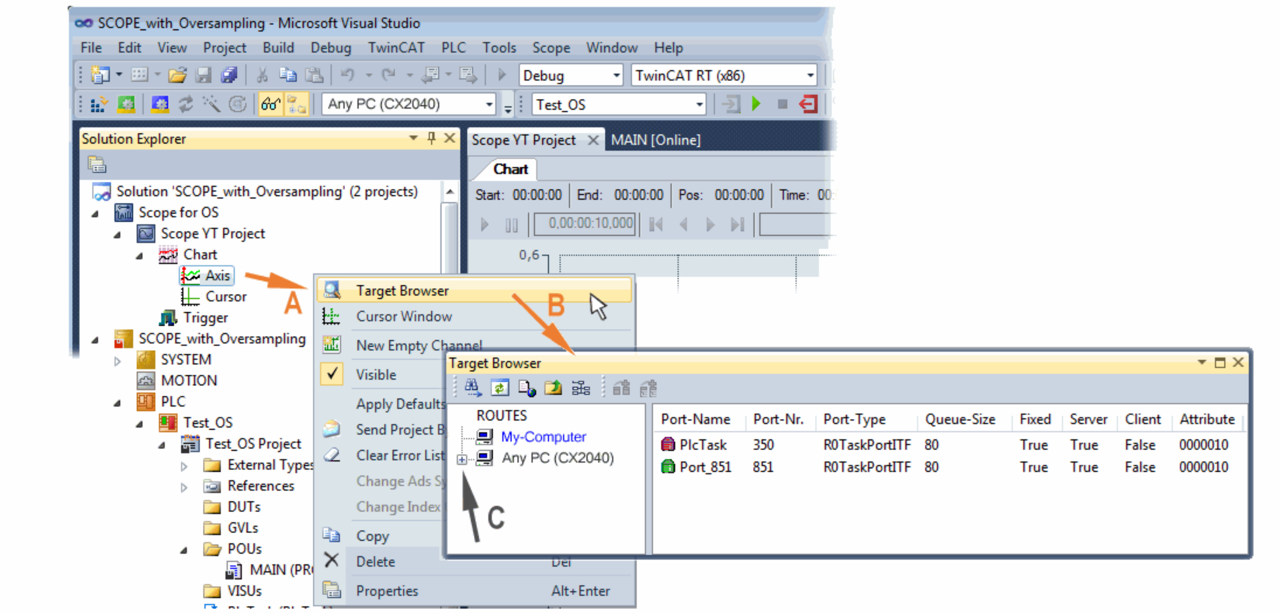


Fig. 48: Selection of the oversampling variable with the target browser

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.

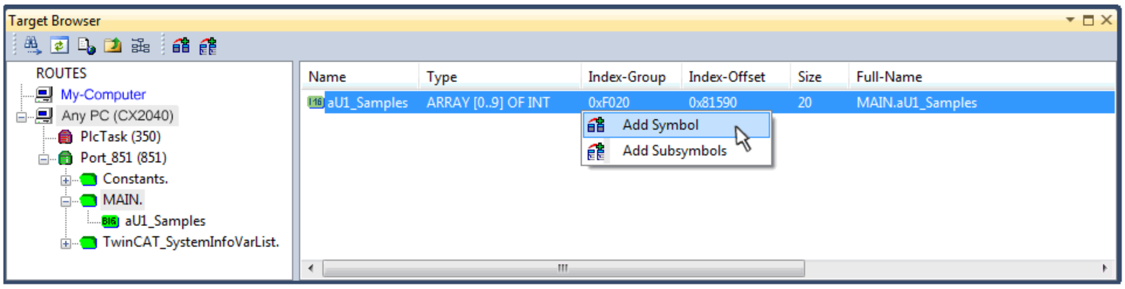
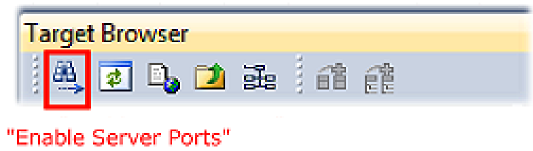


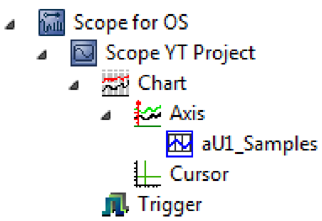
Fig. 49: Appending the variable "aUI_Samples“ below “axis” within the scope project of the solution explorer

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:



Using “Add symbol” displays the variable "aUI_Samples“ below “axis” within the scope project of the solution explorer directly.



Now the program start has to be done with  formally although there's no program still yet. Using "Start Recording"  the 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:

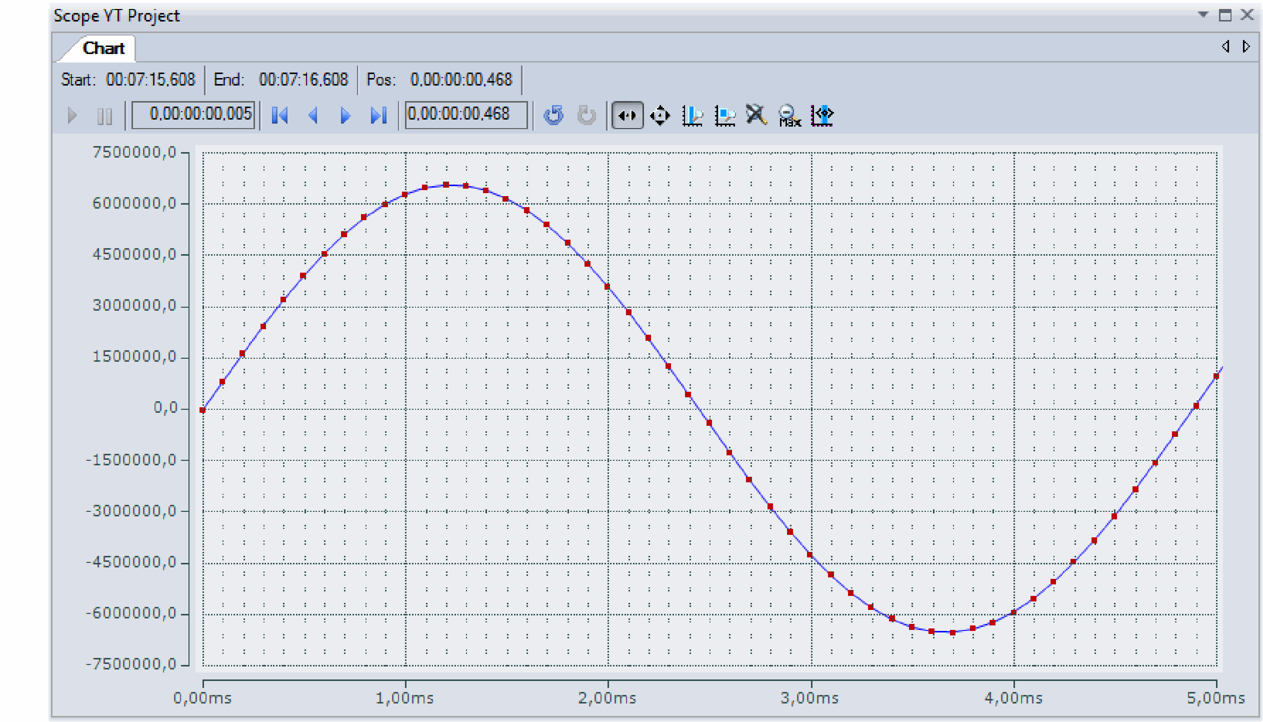


Fig. 50: Example of recording a sine signal with 10 x oversampling at 1 ms measurement cycle time

The X-axis view was fitted properly by using "Panning X"  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 table [▶ 41] above) the oversampling PDO of the respective oversampling able terminal/box can be made visible for the ScopeView by activation of the ADS server.

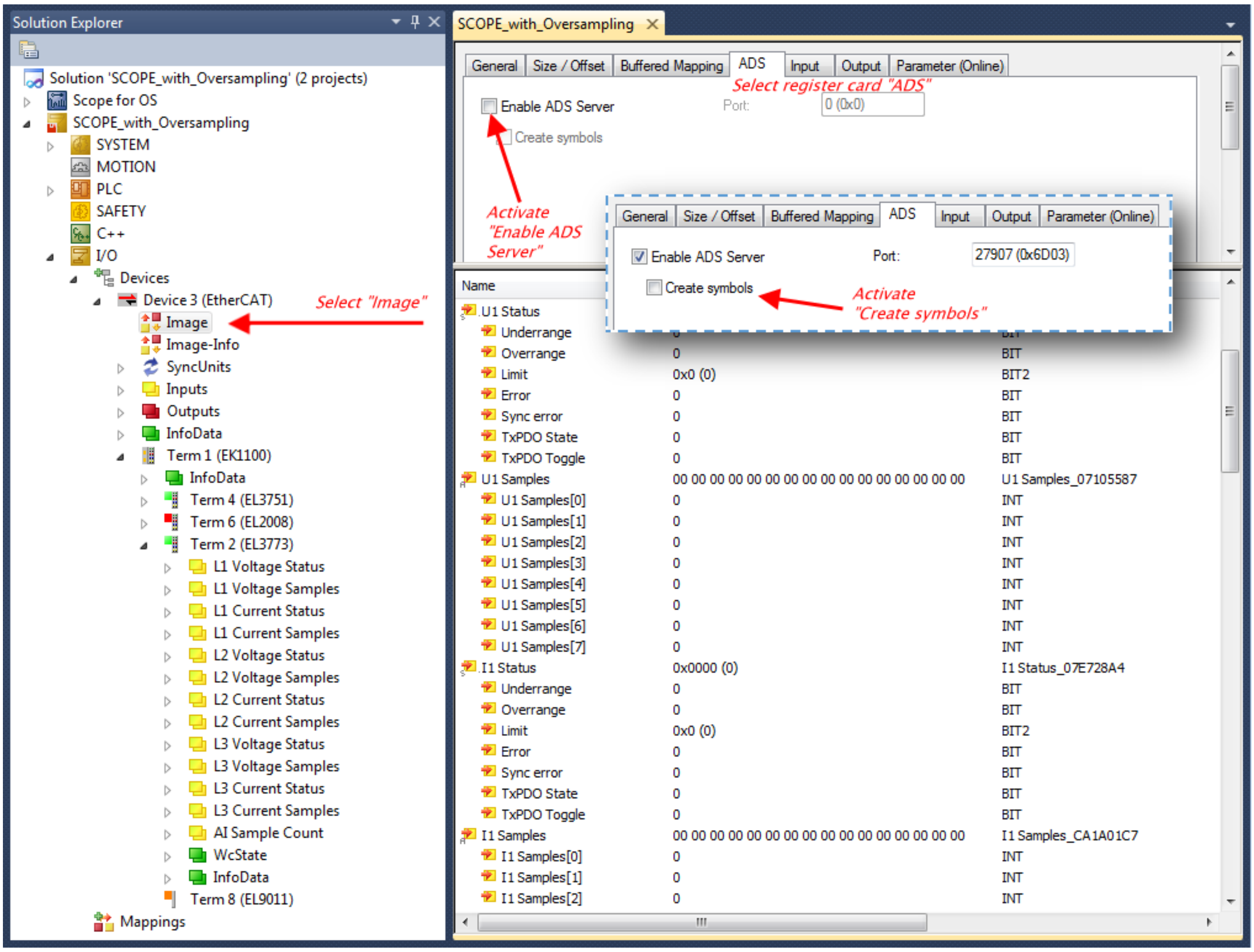


Fig. 51: Activation of the ADS server of the EtherCAT device (TwinCAT 3)

The activation of the server can be carried out by selection of “Image” within the left sided solution explorer: „I/O → Devices → Device .. (EtherCAT) → 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:

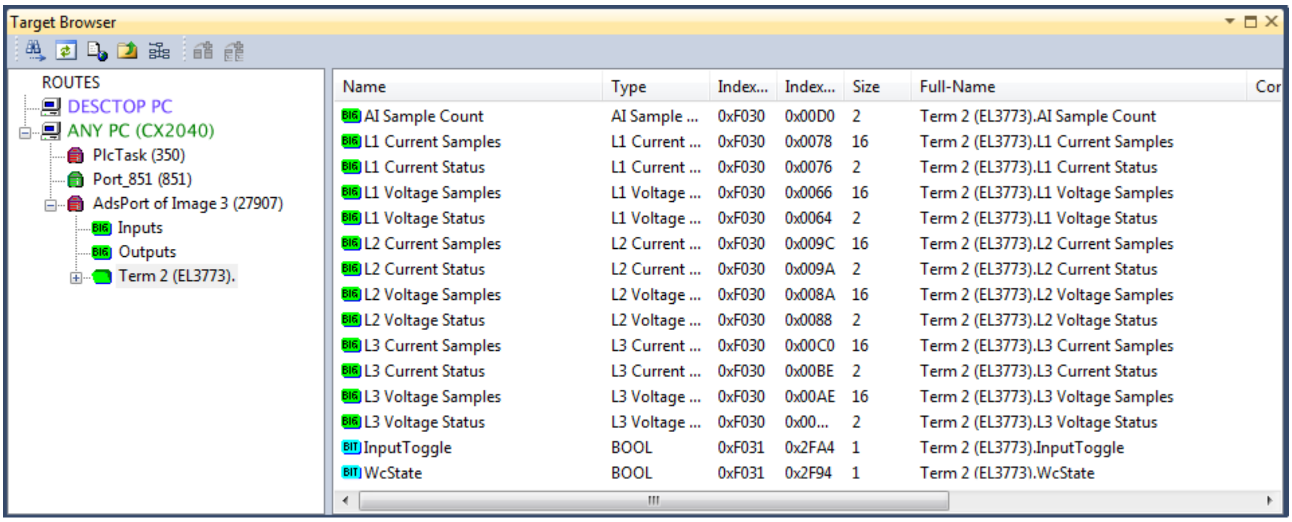
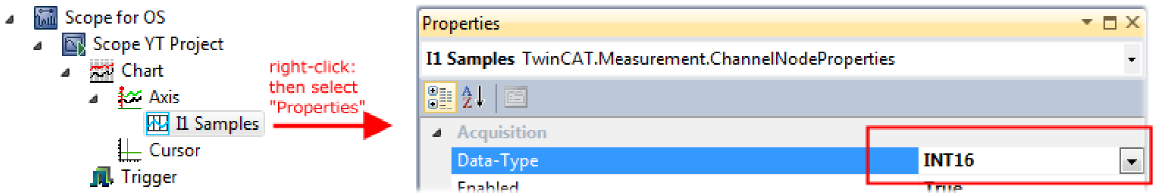


Fig. 52: Direct access to PDOs of the terminal by ScopeView

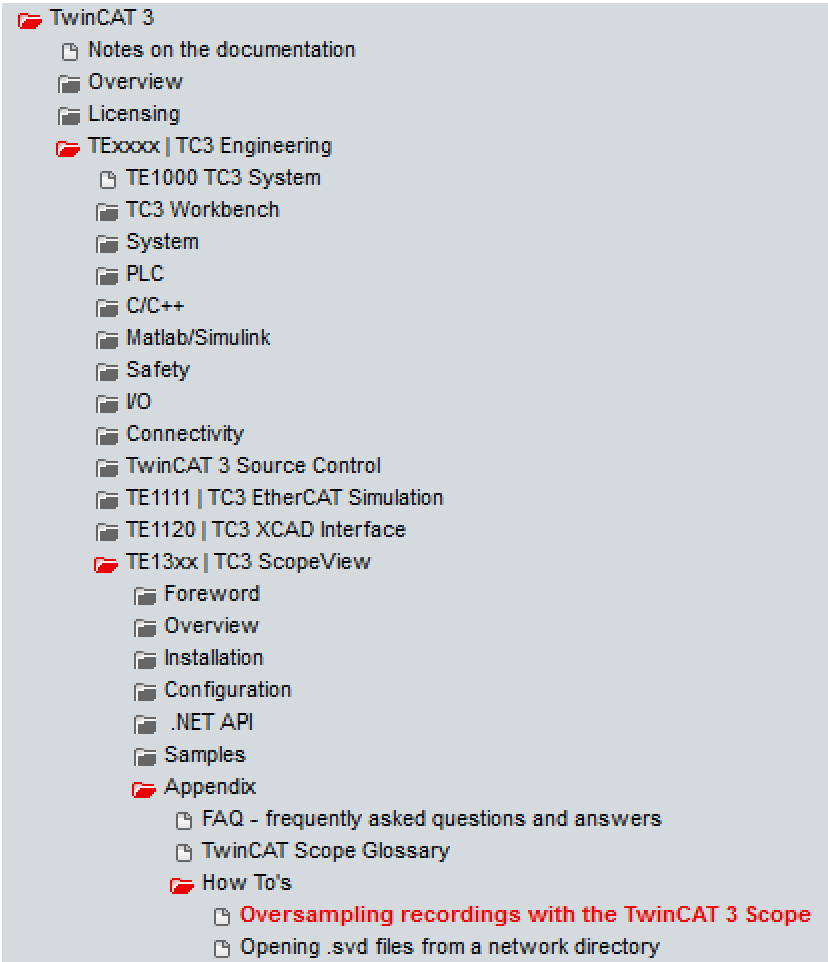
Data type not valid

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:



TwinCAT 3: Activate the ADS Server of an EtherCAT device

Also see Beckhoff Information System:



4.3.2 TwinCAT 2 procedure

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

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 step 1a [▶ 50]

- or directly in the System Manager if only one free task is present, see step 1b [▸ 50]

The same settings are to be made in the Scope2 for both cases, see step 2 [▸ 52]

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. 53: 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. 54: 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.

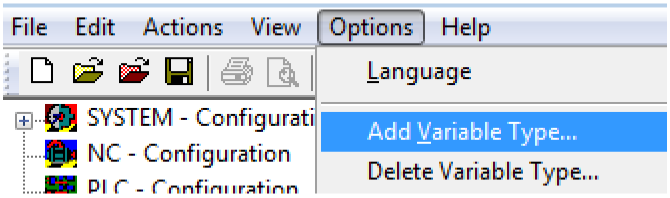


Fig. 55: Add Variable Type

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.

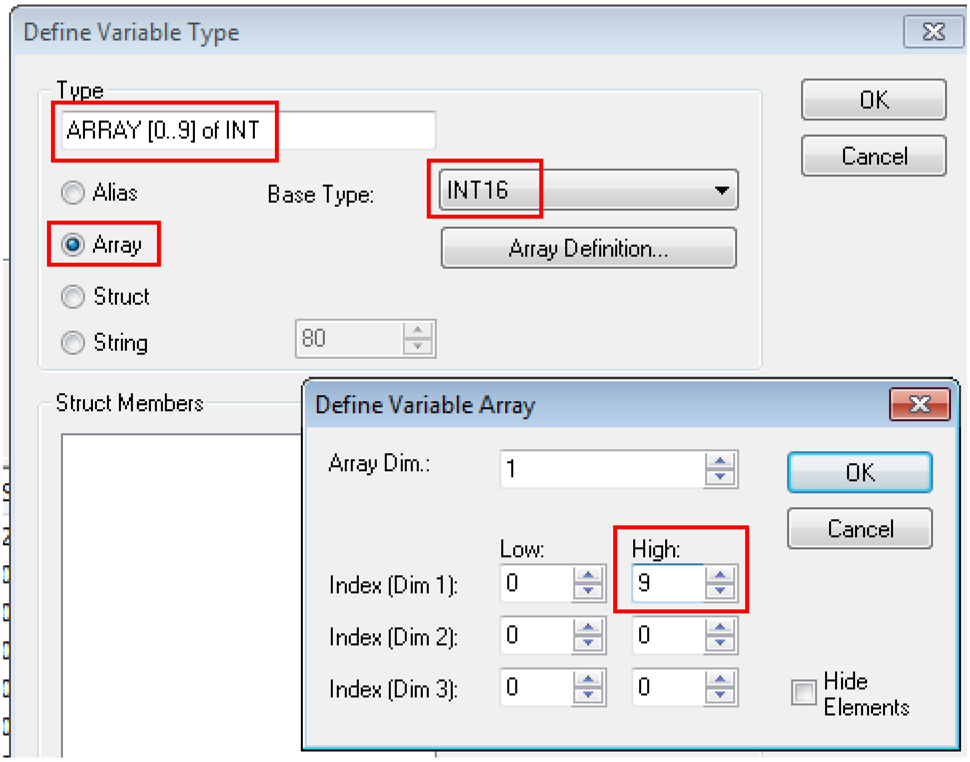


Fig. 56: Definition of the variable type

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.

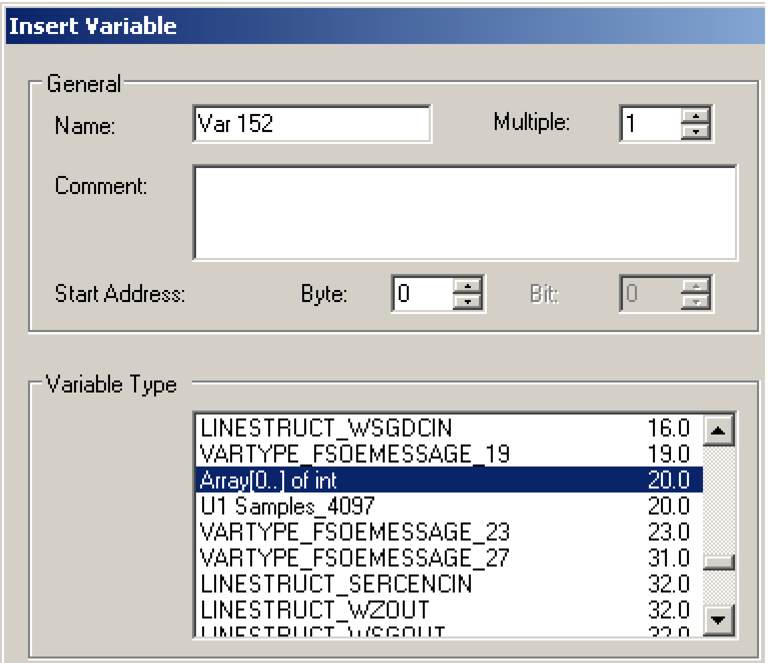


Fig. 57: Overview of declared types

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

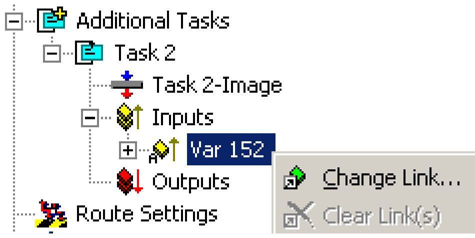


Fig. 58: Linking

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

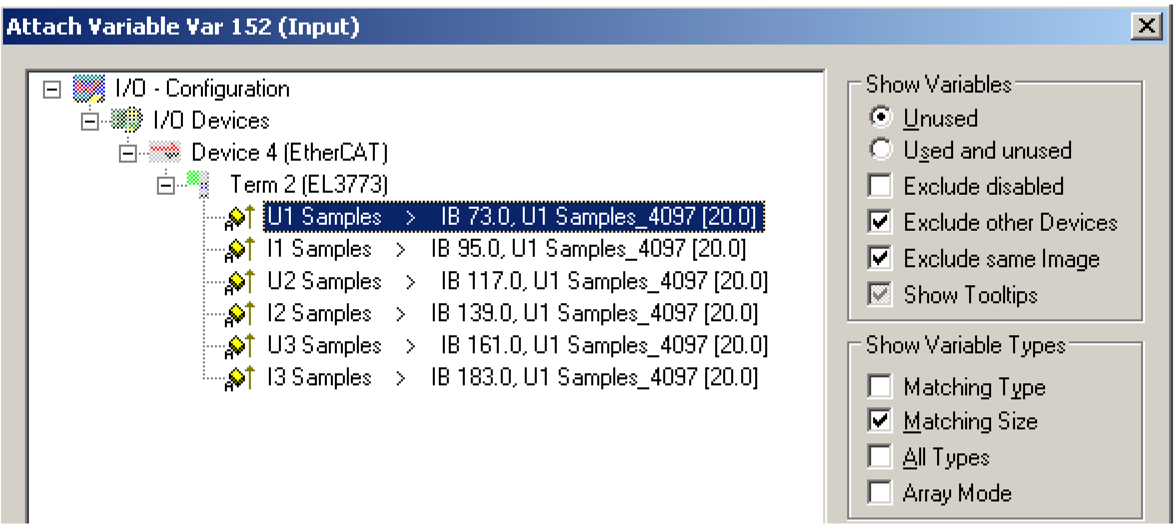


Fig. 59: Array variables of an oversampling terminal

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.

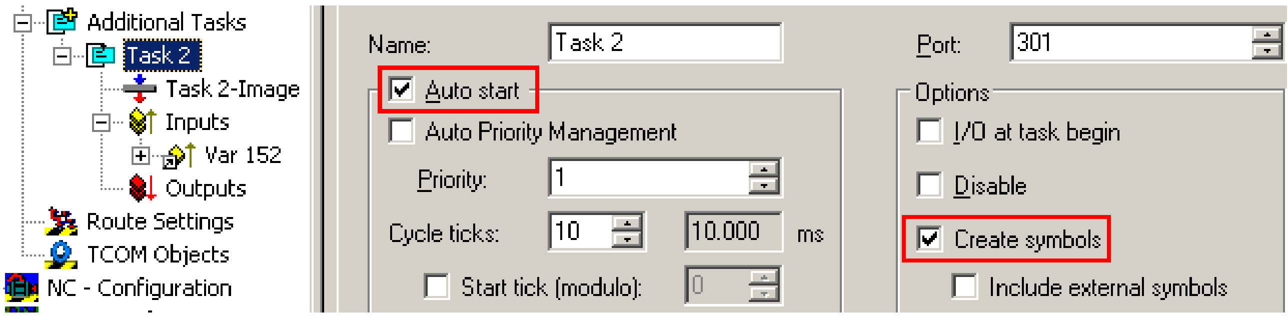


Fig. 60: Settings in the additional task

Step 2: Configuration in the Scope2

So that the linking works, an array variable with the channel data of the respective terminal/box 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 [▸ 50].

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

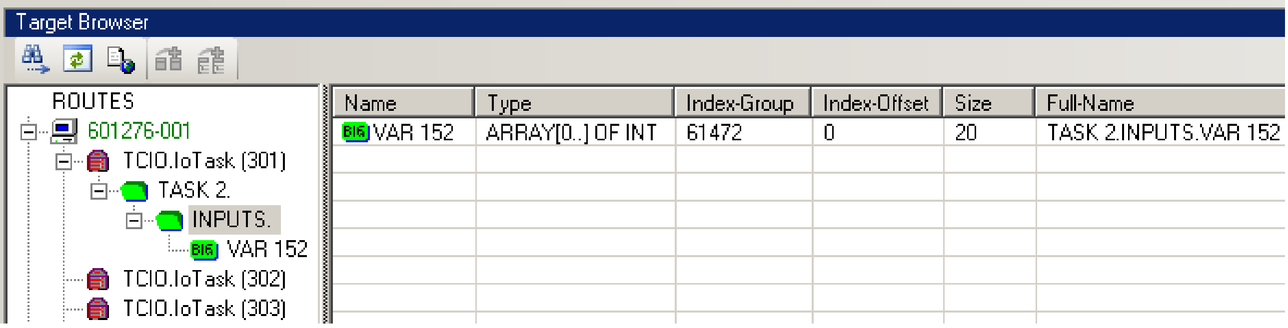


Fig. 61: Variable browser up to the array VAR152

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

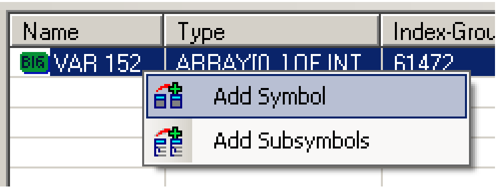


Fig. 62: AddSymbol on the array

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

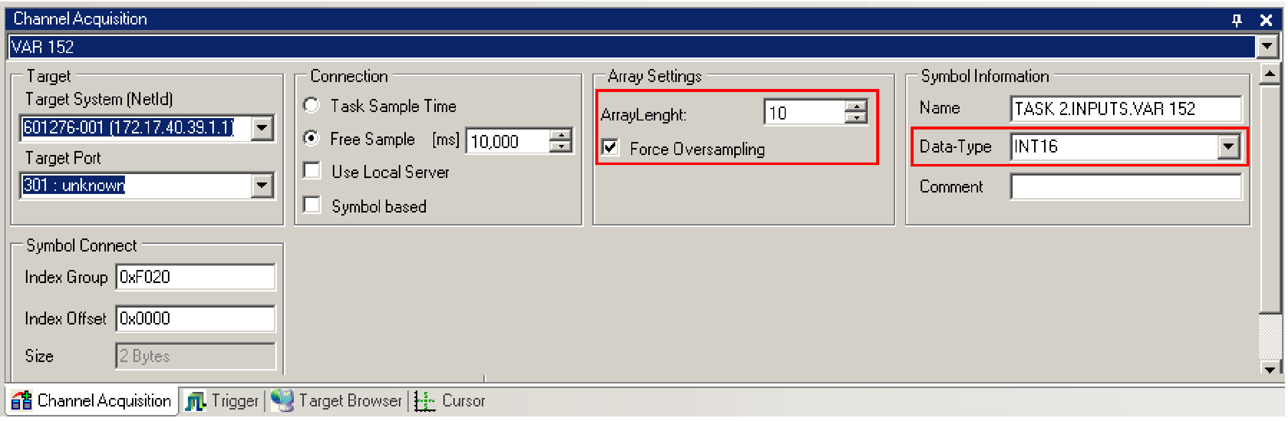


Fig. 63: Channel settings

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.

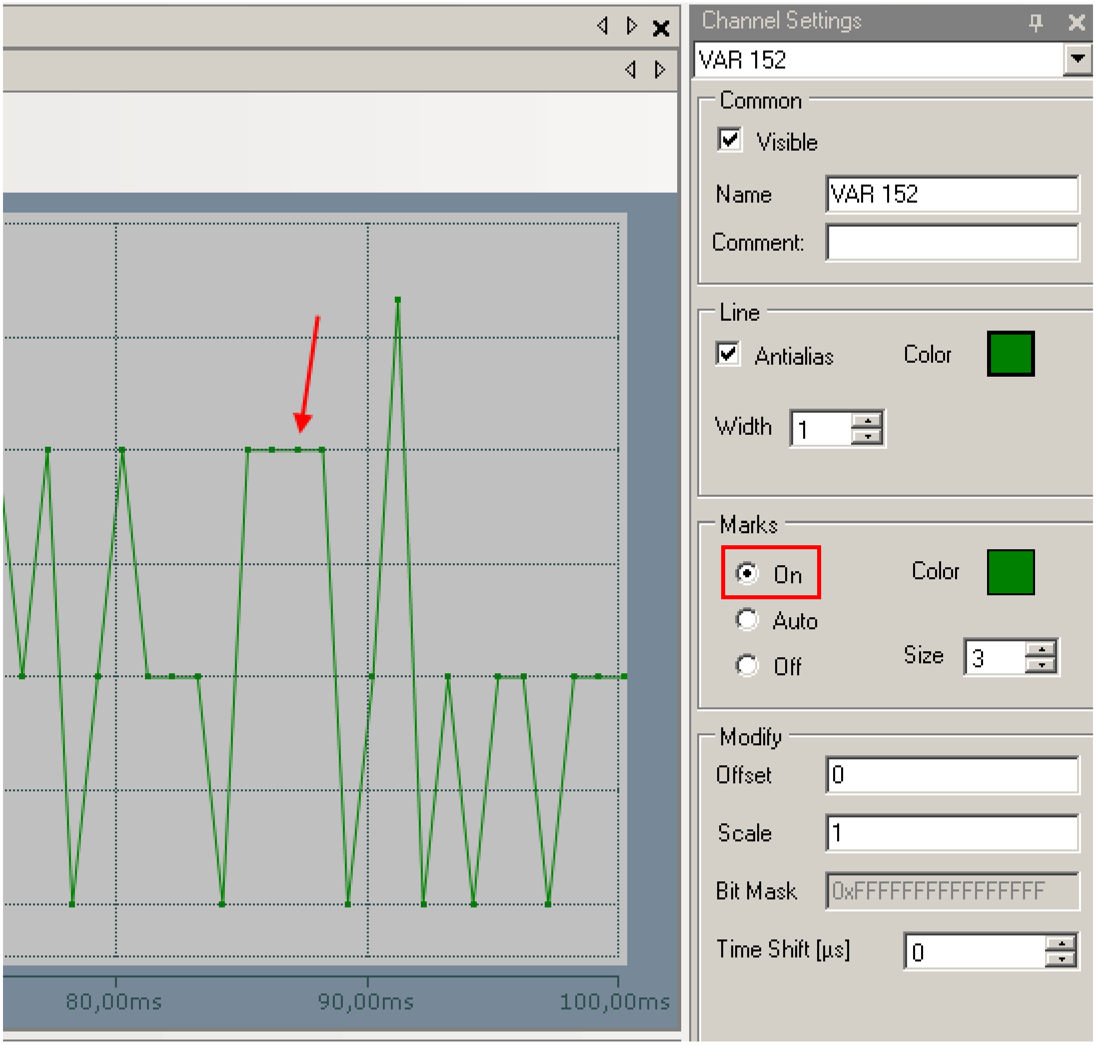


Fig. 64: Activation of the marks

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

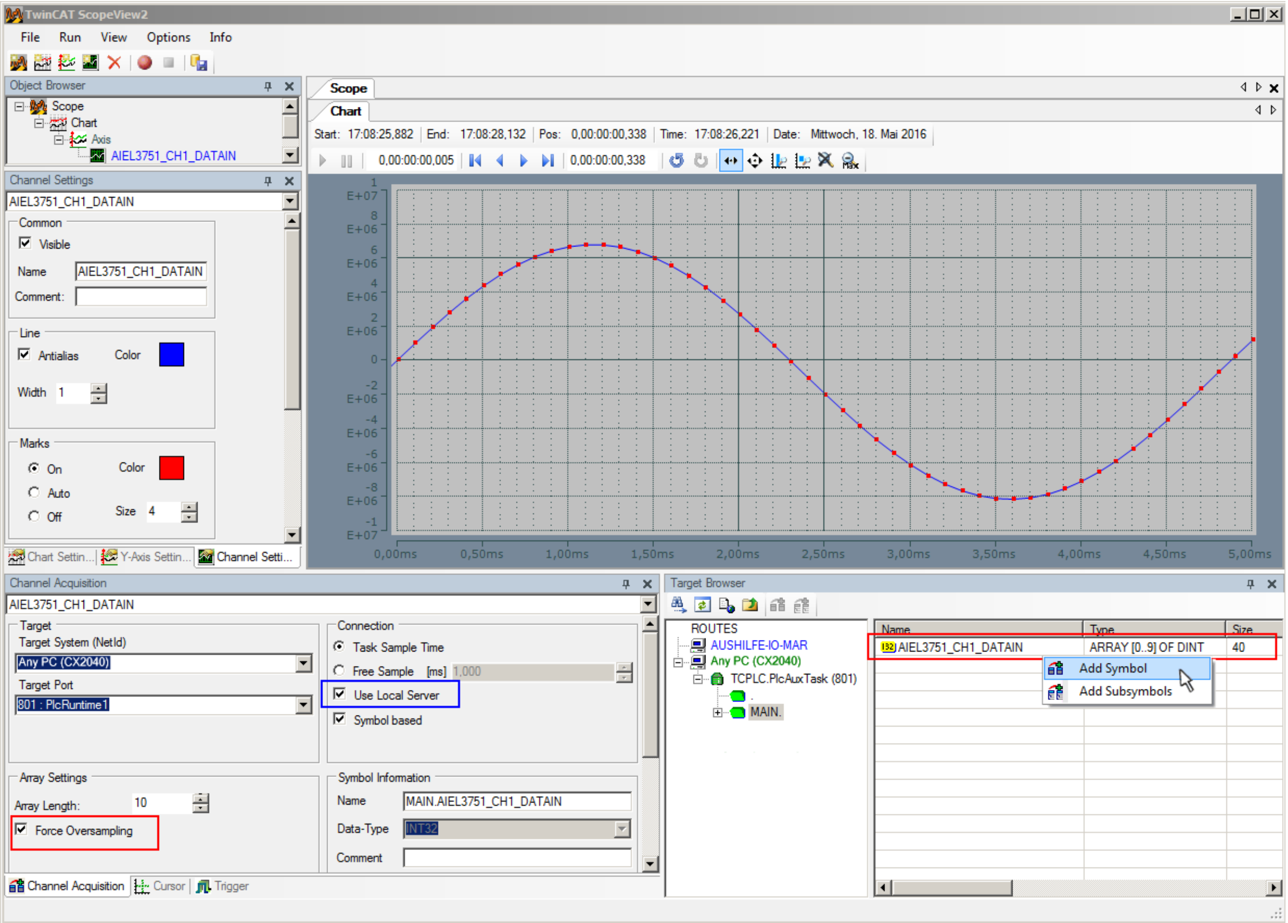


Fig. 65: Illustration of a 10 x oversampling variable of the EL3751 by the Scope2

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/box.

Proceeding with TwinCAT 2/ alternatively via ADS

In former TwinCAT 2 versions (or a lower revision as specified in the table [▶ 41] above) the oversampling PDO of the respective oversampling able terminal/box 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/box is connected with have to be activated.

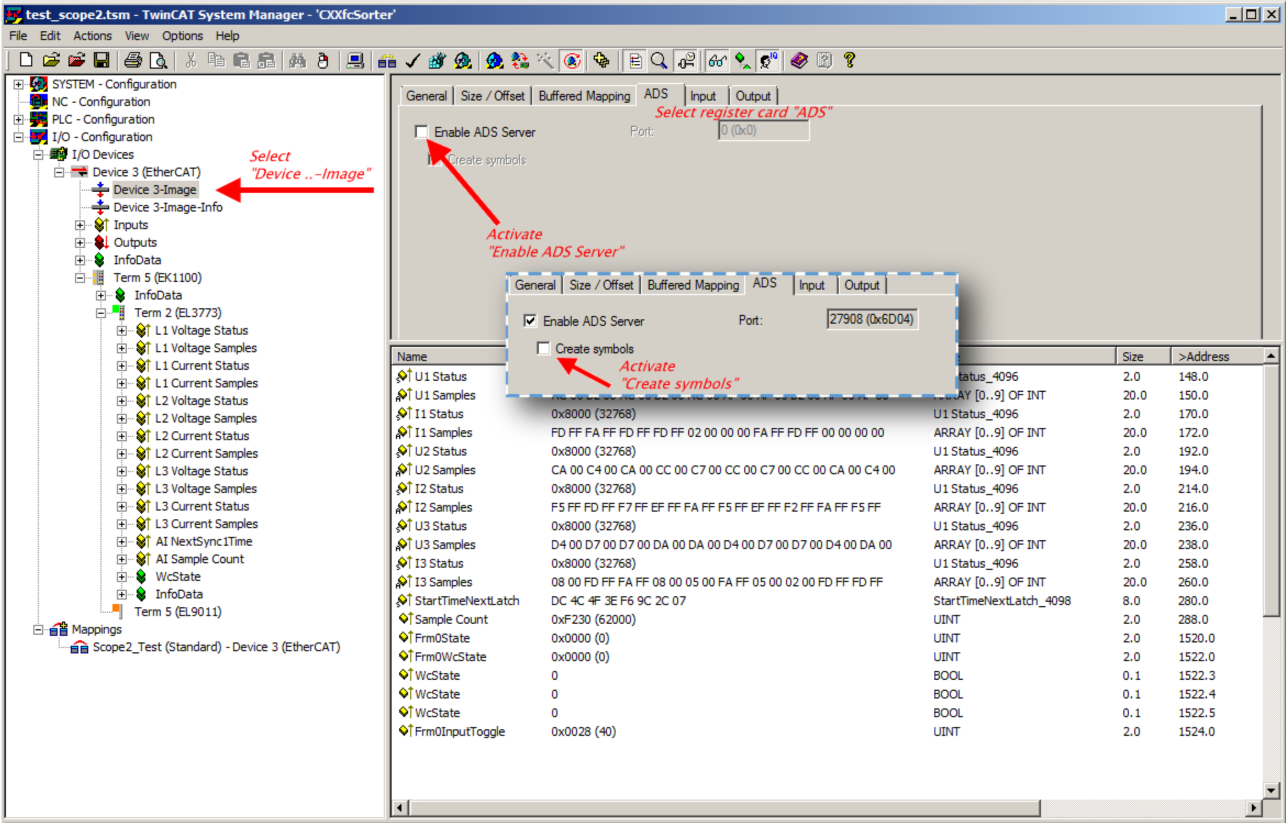


Fig. 66: Activation of the ADS server of the EtherCAT Device (TwinCAT 2)

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.

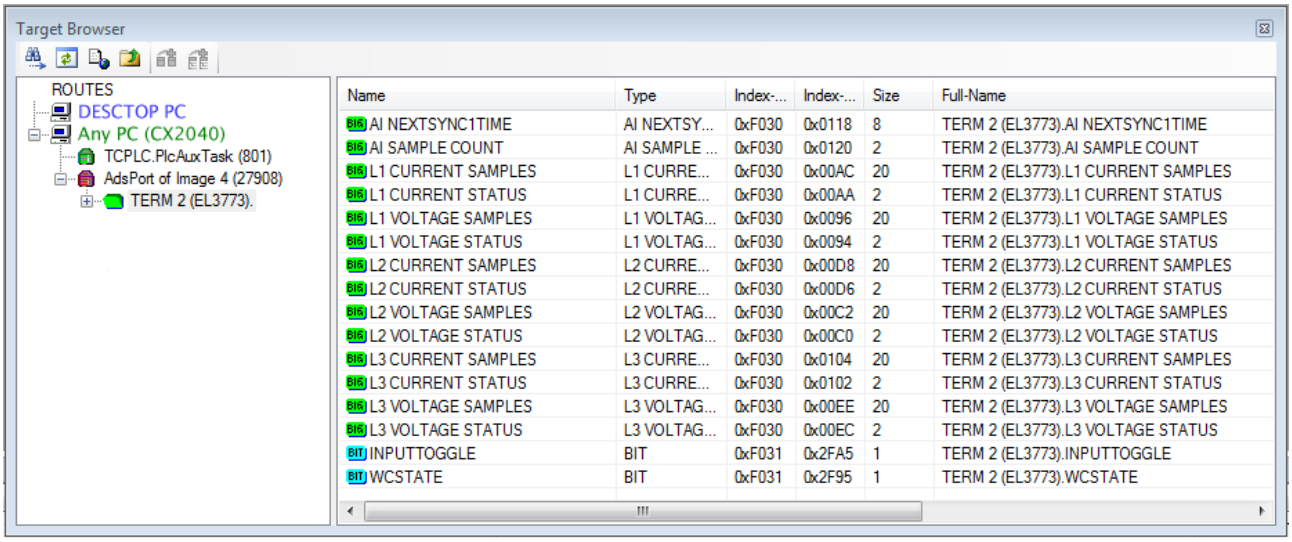
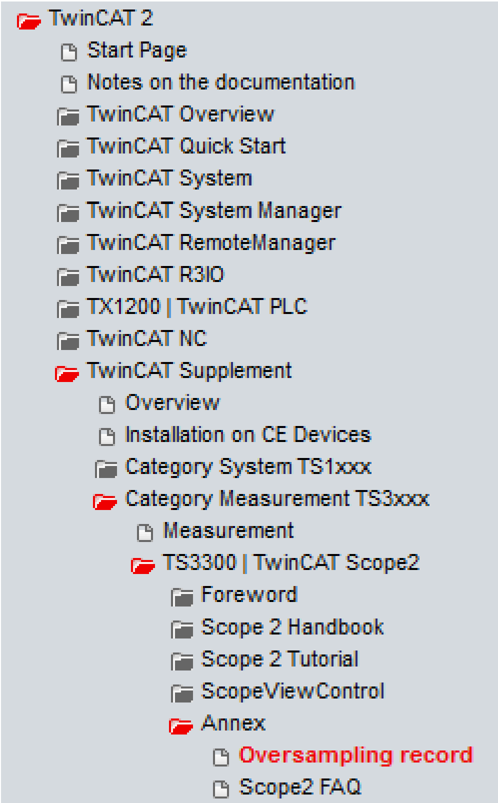


Fig. 67: Direct access of the Scope2 to the terminal's PDOs

Also see Beckhoff Information System

infosys.beckhoff.com → TwinCAT 2 → TwinCAT Supplement → Category Measurement TS3xxx → TS3300 | TwinCAT Scope 2 → Annex → Oversampling record:



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.

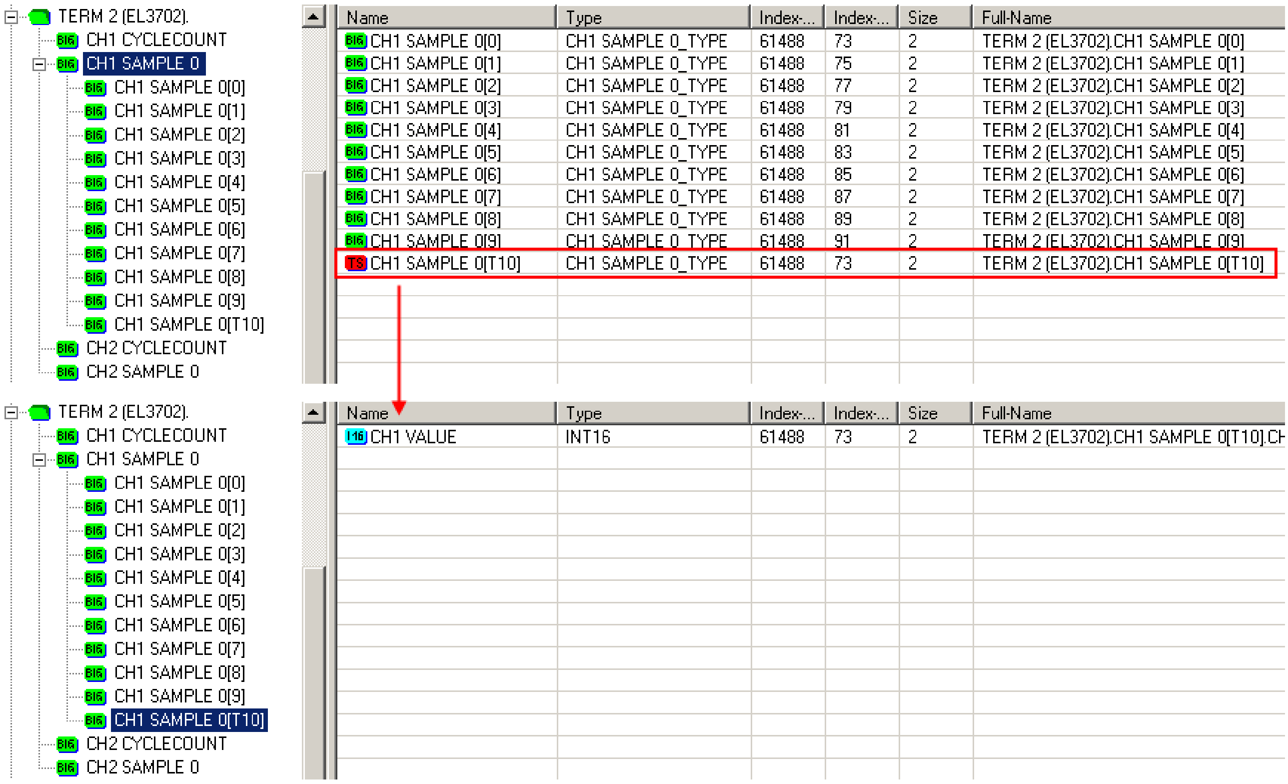


Fig. 68: Automatically calculated array variable (red) in the Scope2

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/ boxes device is just activated. This is then detected by Scope2.

4.4 Commissioning

Determination of the desired sampling rate

The necessary/desired sampling rate results from the selected cycle time and the set oversampling factor. Automatic setting of the terminal/box by the selection of the sampling rate alone is not possible.

Maximum values: 50-fold oversampling, 50 kSP/s, cycle time 10 ms

Sampling Rate	Oversampling factor	Cycle time / µs 100	200	250	500	1000	2000
	1	10 kSps	5 kSps	4 kSps	2 kSps	1 kSps	0,5 kSps
	2	20 kSps	10 kSps	8 kSps	4 kSps	2 kSps	1 kSps
	4	40 kSps	20 kSps	16 kSps	8 kSps	4 kSps	2 kSps
	5	50 kSps	25 kSps	20 kSps	10 kSps	5 kSps	2,5 kSps
	8		40 kSps		16 kSps	8 kSps	4 kSps
	10		50 kSps	40 kSps	20 kSps	10 kSps	5 kSps
	16					16 kSps	8 kSps
	20				40 kSps	20 kSps	10 kSps
	25				50 kSps	25 kSps	12,5 kSps
	32						16 kSps
	40					40 kSps	20 kSps
	50					50 kSps	25 kSps

Fig. 69: Sampling rates in relation to cycle time and oversampling

Sampling Time	Oversampling factor	Cycle time / µs 100	200	250	500	1000	2000
	1	100 µs	200 µs	250 µs	500 µs	1000 µs	2000 µs
	2	50 µs	100 µs	125 µs	250 µs	500 µs	1000 µs
	4	25 µs	50 µs	62,5 µs	125 µs	250 µs	500 µs
	5	20 µs	40 µs	50 µs	100 µs	200 µs	400 µs
	8		25 µs		62,5 µs	125 µs	250 µs
	10		20 µs	25 µs	50 µs	100 µs	200 µs
	16					62,5 µs	125 µs
	20				25 µs	50 µs	100 µs
	25				20 µs	40 µs	80 µs
	32						62,5 µs
	40					25 µs	50 µs
	50					20 µs	40 µs

Fig. 70: Sampling times in relation to cycle times and oversampling

Configurations that demand sampling times not divisible by 500 ns are not supported.

Setting the sampling rate

1. Select the terminal/box in the TwinCAT tree
2. Select the “DC/Oversampling” tab
3. Select the operating mode (1/2-channel)
4. “Sync Unit Cycle Time” is indicated; on the basis of the above table…
5. Select the oversampling factor. The “Sample Cycle Time (µs)” indicates the reciprocal value of the sampling rate. The SM automatically activates all process data entries thereafter.

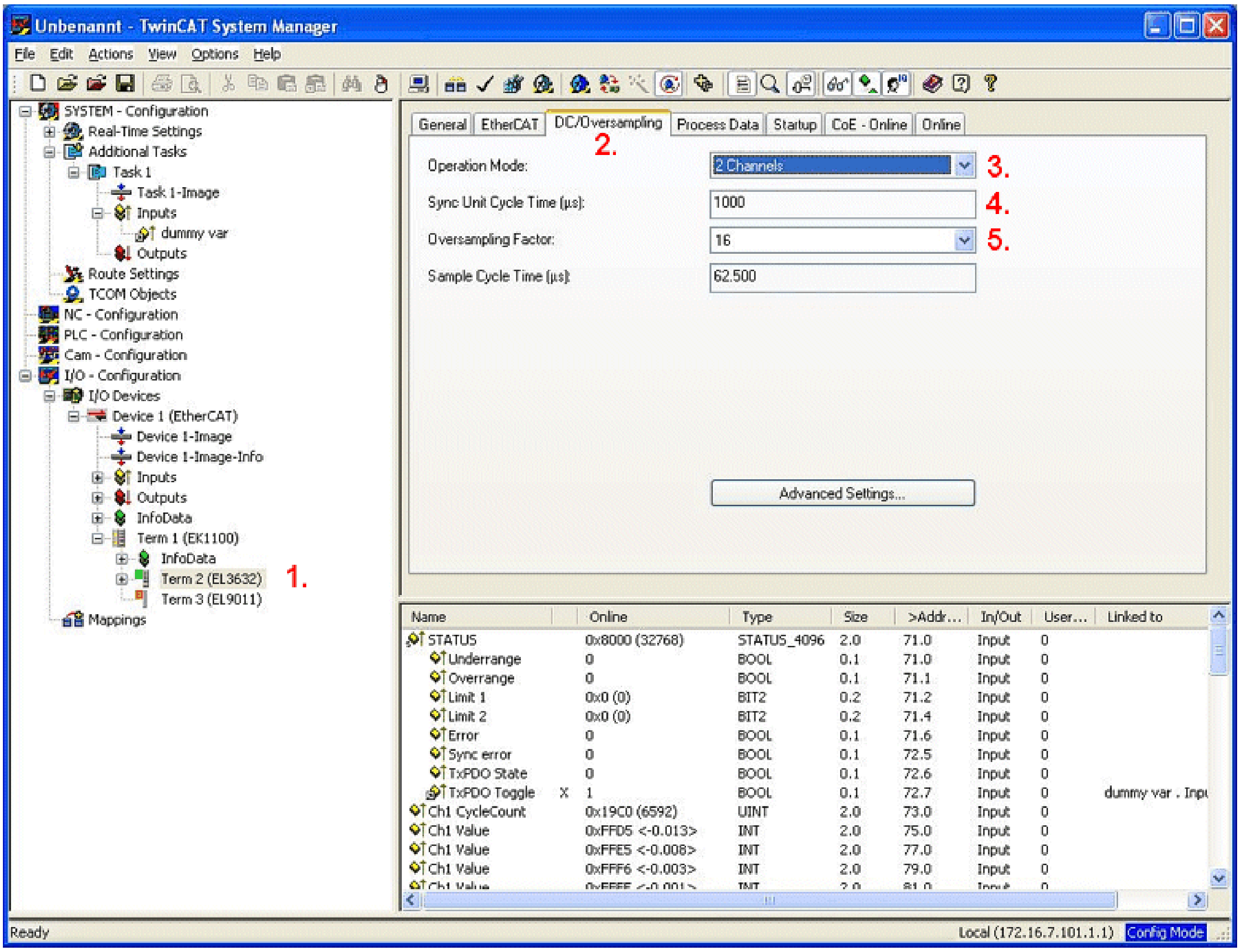


Fig. 71: Setting the sampling rate in TwinCAT

Loading the configuration data (ESI) from the terminal

If the online description is used, the DC/Oversampling dialog is not displayed in the TwinCAT System Manager.
To use the dialog, copy the XML Description into the designated TwinCAT/Io/EtherCAT folder before the TwinCAT System Manager is started.

Application with external masters

The oversampling function can also be activated manually: The oversampling factor should be specified depending on the required sampling rate and cycle time.

For each required channel the status word and the corresponding number of samples have to be entered in object 0x1C13. Activate PDOs "Next Sync1 Time" and/or "Sample Counter", if necessary. To this end, initially set subindex 0 to 0 and at the end to the number of entered values.

The sync interrupts should be parameterized as follows: Sync0: CycleTime/Oversampling Factor, set Enable; Sync1 Cycle Unit Cycle, set Enable.

The master must support Distributed Clocks.

Selection of the process data

No longer necessary with TwinCAT.

Filter

Each channel has a parameterizable 5th order filter with Butterworth characteristic with upstream and downstream anti-aliasing filters that are parameterized automatically. The whole filter stage is based on hardware. There are no software filters (apart from the active offset setting). When the limit frequency is set to 10 Hz (0), an additional amplification factor 20 is activated automatically.

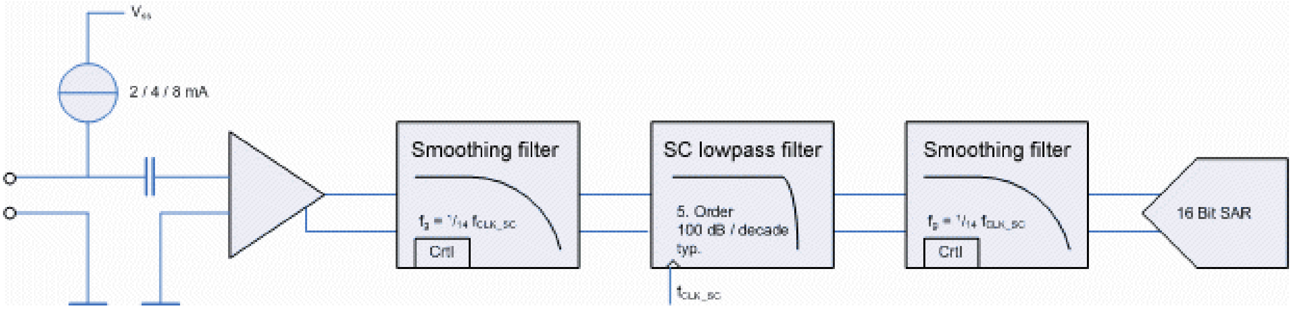


Fig. 72: Filter structure

The analog input filters can be set via the CoE objects 0x80n0:15 (channel 1/2). It is not possible to switch off the filters.

The calculation of the average value is configurable:
0: 10 Hz, Gain 20
1: 100 Hz
2: 500 Hz
3: 1000 Hz
4: 5000 Hz
5: 10000 Hz
6: 25000 Hz
7: 2000 Hz (from firmware 11)

Setting the filters

The filters must be configured separately for both channels.
Setting the filters will interrupt the conversion of new data for a short time.

Active offset setting

The function "Active offset adjustment" calculates the long-term average of the values. The calculated offset value is used instead of the set "user" and "vendor calibration offset" entries. At least one function, "user" or "vendor calibration", must be activated

The calculation of the average value is configurable:
Level 1: b = 1/4096
Level 2: b = 1/8192
Level 3: b = 1/16384
Level 4: b = 1/32768
Level 5: b = 1/65536
Level 6: b = 1/131072 (128 k)
Level 7: b = 1/262144 (256 k)
Level 8: b = 1/524288 (512 k).

Calibration

The input values can be calibrated by means of manufacturer or user values:

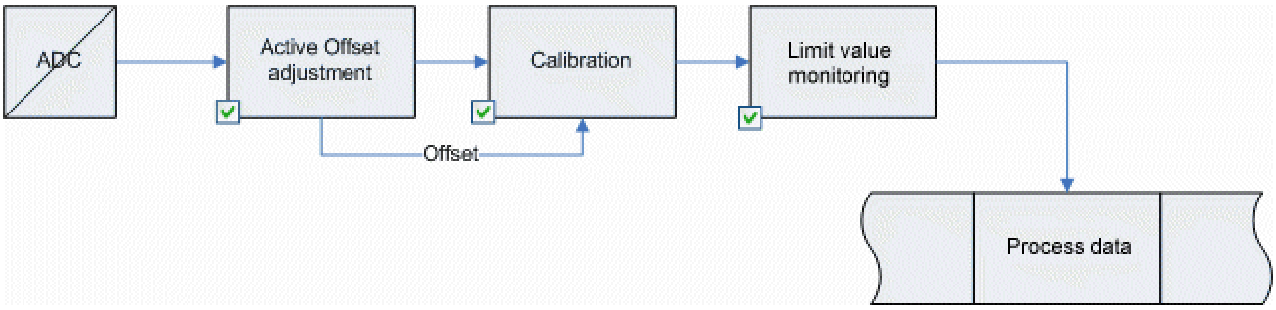


Fig. 5: Data flow

Vendor calibration, index 0x80n0:0B

The vendor calibration is enabled via index 0x80n0:0B. Parameterization takes place via the indices

- 0x80nF:01 offset (manufacturer compensation)
- 0x80nF:02 gain (manufacturer compensation)

$Y_H = (X_{ADC} - B_K) * A_K$	Measured value following manufacturer calibration (corresponds to X_{ADC} if index 0x80n0:0B is inactive)

User calibration, index 0x80n0:0A

The user calibration is enabled via index 0x80n0:0A. Parameterization takes place via the indices

- 0x80n0:17 User calibration offset
- 0x80n0:18 User gain compensation

$Y_A = (Y_H - B_W) * A_W$	Measured value following user calibration (corresponds to Y_H if index 0x80n0:0A is inactive)

Active offset adjustment

If the function "Active offset adjustment" is active the offset values are not used. Instead, a dynamically calculated offset is subtracted.

Example interpretation

Sample: A sensor with a sensitivity S of 100 mV / g (10.2 mV/(m/s^2)) is connected to a synchronized EL3632/EP3632 (15-bit resolution + sign, +/- 5 V). In the process data an amplitude of 1507 is measured.

$a = Y_A * 5\,V / (2^{15} * S)$ $a = 1507 * 5\,V / (2^{15} * 0.1\,V/g)$ $a = 2.3\,g$ $a = 2.3\,g * 9.81\,(m/s^2) / g$ $a = 22.5\,m/s^2$	Conversion of process data value Y_A to acceleration a.
$Y_A = 2^{15} / 5\,V * S * a$ $Y_A = 2^{15} / 5\,V * 0.1\,V/g * 2.3\,g$ $Y_A = 1507$	Conversion of acceleration g to process data value Y_A.

Sensor connection

The supply current for the sensors is configurable. With 8 mA the vertical (standard) installation position of the terminal must be ensured. The smallest possible supply current should be set, depending on the sensor and cable length.

After switching on the 24 V supply voltage or connecting the sensor, a leakage current forms due to the input capacity on the high-pass filter. This current is based on the physical properties of electrolytic capacitors and is technically impossible to prevent. This current stabilizes at a constant value within a few minutes, and during the measurement it generates a constant offset of typically a few mV within the specified tolerance range. If this offset should prove to be disturbing when analyzing the measurement, it can be permanently and automatically subtracted out by activating the "Active offset adjustment" (object 0x80n0:21).

A shielded (simple or multiple) sensor cable must be used. The shield should be connected directly at the shield connections of the terminal.

The red error LED comes on and the error bit is set in the event of an open circuit or if no sensor is connected. If only the first channel is activated, the red LED for the second channel is disabled in SAFEOP and OP state.

Unused inputs

Unused inputs must not be short-circuited.

Measuring error

Measuring error < ±0,5% (DC; relative to full scale value), taking into account the Butterworth characteristic.

4.4.1 CoE object description

4.4.1.1 Profile-specific and parameterization objects

EtherCAT XML Device Description

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

Parameterization via the CoE list (CAN over EtherCAT)

The EtherCAT device is parameterized via the CoE-Online tab (double-click on the respective object) or via the Process Data tab (allocation of PDOs). Please note the following general CoE notes when using/manipulating the CoE parameters:

- Keep a startup list if components have to be replaced
- Differentiation between online/offline dictionary, existence of current XML description
- use “CoE reload” for resetting changes

4.4.1.1.1 Restore object

Index 1011 Restore default parameters

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

4.4.1.1.2 Configuration data

Index 80n0 AI settings (for n = 0: channel 1; n = 1: channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	AI Settings	Maximum subindex	UINT8	RO	0x20 (32_{dec})
80n0:07	Enable limit	Activate limit evaluation (deviates from other EL terminals!)	BOOLEAN	RW	0x0 (FALSE)
80n0:0A	Enable user calibration	Enabling of the user calibration	BOOLEAN	RW	0x0 (FALSE
80n0:0B	Enable vendor calibration	Enabling of the vendor calibration	BOOLEAN	RW	0x1 (TRUE)
80n0:13	Limit 1	Upper limit value	INT16	RW	0x0000 (0_{dec})
80n0:14	Limit 2	Lower limit value	INT16	RW	0x0000 (0_{dec})
80n0:15	Filter settings	Filter [▸ 60] settings 0: 10 Hz, Gain 20 1: 100 Hz 2: 500 Hz 3: 1000 Hz 4: 5000 Hz 5: 10000 Hz 6: 25000 Hz 7: 2000 Hz (from firmware 11)	ENUM	RW	10000 Hz (5)
80n0:17	User calibration offset	User calibration offset	INT16	RW	0
80n0:18	User calibration gain	User calibration gain factor	INT16	RW	16384
80n0:20	Supply current	Sensor current setting 0: 2 mA 1: 4 mA 2: 8 mA	ENUM	RW	2 mA (0)
80n0:21	Active offset adjustment	Automatic offset calculation [▸ 60] 0: Disabled 1: Level 1 2: Level 2 3: Level 3 4: Level 4 5: Level 5 6: Level 6 7: Level 7 8: Level 8	ENUM	RW	Disabled (0)

Index 80n8 AI advanced settings (for n = 0: channel 1; n = 1: channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n8:0	AI Advanced Settings	Maximum subindex	UINT8	RO	0x06 (6_{dec})
80n8:06	Input Amplifier	permitted values: 0: Preset (setting via object 0x80n0:15 [▸ 63]) 1: ON (switch-on of the analog amplifier x20 irrespective of filter settings; gain factor of 20 is not changeable) 2: OFF (switch-off of the analog amplifier irrespective of the filter settings)	BIT2	RW	0x00 (0_{dec})

Index 8012 AI Device settings

Index (hex)	Name	Meaning	Data type	Flags	Default
8012:0	AI Device Settings	Maximum subindex	UINT8	RO	0x11 (17_{dec})
8012:11	DC Timestamp Shift		UINT32	RW	0x00000000 (0_{dec})

Index 80nF AI Vendor data (for n = 0: channel 1; n = 1: channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80nF:0	AI vendor data	Maximum subindex	UINT8	RO	0x04 (4_{dec})
80nF:01	Calibration offset (gain 1)	Offset (vendor calibration), gain 1	INT16	RW	0x0000 (0_{dec})
80nF:02	Calibration gain (gain 1)	Gain (vendor calibration), gain 1	INT16	RW	0x0000 (0_{dec})
80nF:03	Calibration offset (gain 20)	Offset (vendor calibration), gain 20	INT16	RW	0x0000 (0_{dec})
80nF:04	Calibration gain (gain 20)	Gain (vendor calibration), gain 20	INT16	RW	0x0000 (0_{dec})

4.4.1.1.3 Command object

Index FB00 command

Index (hex)	Name	Meaning	Data type	Flags	Default
FB00:0	DCM Command	Command interface	UINT8	RO	0x03 (3_{dec})
FB00:01	Request	0x8000: Software reset, hardware is re-initialized with the current CoE configuration (this otherwise happens only during the transition to INIT)	OCTET-STRING[2]	RW	{0}
FB00:02	Status	0x8000: 0x01 if a reset has taken place	UINT8	RO	0x00 (0_{dec})
FB00:03	Response	0x8000: not used	OCTET-STRING[4]	RO	{0}

4.4.1.1.4 Input data

Index 60n0 Status (for n = 0: channel 1; n = 1: channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	SAI Inputs	Maximum subindex	UINT8	RO	0x10 (16_{dec})
60n0:01	Underrange	Indicates that the electrical measuring range is undershot	BOOLEAN	RO	0x00 (0_{dec})
60n0:02	Overrange	Indicates that the electrical measuring range is exceeded	BOOLEAN	RO	0x00 (0_{dec})
60n0:03	Limit 1	1: one or more values smaller than or equal to Limit 2 2: one or more values greater than or equal to Limit 1 3: 1 and 2 both true	BIT2	RO	0x00 (0_{dec})
60n0:07	Error	Measuring error	BOOLEAN	RO	0x00 (0_{dec})
60n0:0E	Sync error	Synchronization error	BOOLEAN	RO	0x00 (0_{dec})
60n0:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0_{dec})
60n0:10	TxPDO Toggle	A new measured value is available (if the toggle bit was changed). Status bits may be changed independent from the toggle bit.	BOOLEAN	RO	0x00 (0_{dec})

Index 60n1 Samples (for n = 0: channel 1; n = 1: channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
60n1:0	Samples	Maximum subindex	UINT8	RO	0x32 (50_{dec})
60n1:01	Subindex 001	1. Sample	UINT16	RO P	0x0000 (0_{dec})
...	...	...	...	...	...
60n1:32	Subindex 050	50. Sample	UINT16	RO P	0x0000 (0_{dec})

Index 6020 Next Sync 1 Time

Index (hex)	Name	Meaning	Data type	Flags	Default
6020:0	Next Sync 1 Time	Maximum subindex	UINT8	RO	0x01 (1_{dec})
6020:01	Start time next latch	DC timestamp of the next data set	UINT64	RO P	0x00 00 00 00 00 00 00 00 (0_{dec})

Index 6021 Sample Count

Index (hex)	Name	Meaning	Data type	Flags	Default
6021:0	Sample Count	Maximum subindex	UINT8	RO	0x01 (1_{dec})
6021:01	Sample Count	Sample counter (incremented with each ADC value)	UINT16	RO P	0x0000 (0_{dec})

4.4.1.1.5 Information / diagnosis data (device-specific)

Index 10F3 Diagnosis History

Index (hex)	Name	Meaning	Data type	Flags	Default
10F3:0	Diagnosis History	Maximum subindex	UINT8	RO	0x15 (21_{dec})
10F3:01	Maximum Messages	Maximum number of stored messages A maximum of 16 messages can be stored	UINT8	RO	0x00 (0_{dec})
10F3:02	Newest Message	Subindex of the latest message	UINT8	RO	0x00 (0_{dec})
10F3:03	Newest Acknowledged Message	Subindex of the last confirmed message	UINT8	RW	0x00 (0_{dec})
10F3:04	New Messages Available	Indicates that a new message is available	BOOLEAN	RO	0x00 (0_{dec})
10F3:05	Flags	not used	UINT16	RW	0x0000 (0_{dec})
10F3:06	Diagnosis Message 001	Message 1	OCTET-STRING[20]	RO	{0}
...	...	..	...	...	...
10F3:15	Diagnosis Message 016	Message 16	OCTET-STRING[20]	RO	{0}

Index 10F8 Actual Time Stamp

Index (hex)	Name	Meaning	Data type	Flags	Default
10F8:0	Actual Time Stamp	Time stamp	UINT64	RO	

4.4.1.2 Standard objects and PDO mapping

EtherCAT XML Device Description

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

Parameterization via the CoE list (CAN over EtherCAT)

The EtherCAT device is parameterized via the CoE-Online tab (double-click on the respective object) or via the Process Data tab (allocation of PDOs). Please note the following general CoE notes when using/manipulating the CoE parameters:

- Keep a startup list if components have to be replaced
- Differentiation between online/offline dictionary, existence of current XML description
- use “CoE reload” for resetting changes

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: The Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x012C1389 (19665801_{dec})

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EP3632

Index 1009 Hardware version

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

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	05

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x0E303052 (238039122_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00110000 (1114112_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter handling

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

Index 1A00 Analog Input TxPDO-Map Status Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	Analog Input TxPDO-MapStatus Ch.1	PDO Mapping TxPDO 1	UINT8	RO	0x09 (9_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Status), entry 0x01 (Underrange))	UINT32	RO	0x6000:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Status), entry 0x02 (Overrange))	UINT32	RO	0x6000:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Status), entry 0x03 (Limit 1))	UINT32	RO	0x6000:03, 2
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Status), entry 0x05 (Limit 2))	UINT32	RO	0x6000:05, 2
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (Status), entry 0x07 (Error))	UINT32	RO	0x6000:07, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (Status), entry 0x0E (Sync error))	UINT32	RO	0x6000:0E, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (Status), entry 0x0F (TxPDO State))	UINT32	RO	0x6000:0F, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (Status), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6000:10, 1

Index 1A01 Analog Input TxPDO-MapSamples 1 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	Analog Input TxPDO-MapSamples 1 Ch.1	PDO Mapping TxPDO 2	UINT8	RO	0x01 (1_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x01 ())	UINT32	RO	0x6001:01, 16

Index 1A02 Analog Input TxPDO-MapSamples 2 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	Analog Input TxPDO-MapSamples 2 Ch.1	PDO Mapping TxPDO 3	UINT8	RO	0x01 (1_{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x02 ())	UINT32	RO	0x6001:02, 16

Index 1A03 Analog Input TxPDO-MapSamples 3 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A03:0	Analog Input TxPDO-MapSamples 3 Ch.1	PDO Mapping TxPDO 4	UINT8	RO	0x01 (1_{dec})
1A03:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x03 ())	UINT32	RO	0x6001:03, 16

Index 1A04 Analog Input TxPDO-MapSamples 4 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A04:0	Analog Input TxPDO-MapSamples 4 Ch.1	PDO Mapping TxPDO 5	UINT8	RO	0x01 (1_{dec})
1A04:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x04 ())	UINT32	RO	0x6001:04, 16

Index 1A05 Analog Input TxPDO-MapSamples 5 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A05:0	Analog Input TxPDO-MapSamples 5 Ch.1	PDO Mapping TxPDO 6	UINT8	RO	0x01 (1_{dec})
1A05:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x05 ()))	UINT32	RO	0x6001:05, 16

Index 1A06 Analog Input TxPDO-MapSamples 6 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A06:0	Analog Input TxPDO-MapSamples 6 Ch.1	PDO Mapping TxPDO 7	UINT8	RO	0x01 (1_{dec})
1A06:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x06 ())	UINT32	RO	0x6001:06, 16

Index 1A07 Analog Input TxPDO-MapSamples 7 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A07:0	Analog Input TxPDO-MapSamples 7 Ch.1	PDO Mapping TxPDO 8	UINT8	RO	0x01 (1_{dec})
1A07:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x07 ())	UINT32	RO	0x6001:07, 16

Index 1A08 Analog Input TxPDO-MapSamples 8 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A08:0	Analog Input TxPDO-MapSamples 8 Ch.1	PDO Mapping TxPDO 9	UINT8	RO	0x01 (1_{dec})
1A08:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x08 ())	UINT32	RO	0x6001:08, 16

Index 1A09 Analog Input TxPDO-MapSamples 9 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A09:0	Analog Input TxPDO-MapSamples 9 Ch.1	PDO Mapping TxPDO 10	UINT8	RO	0x01 (1_{dec})
1A09:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x09 ())	UINT32	RO	0x6001:09, 16

Index 1A0A Analog Input TxPDO-MapSamples 10 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0A:0	Analog Input TxPDO-MapSamples 10 Ch.1	PDO Mapping TxPDO 11	UINT8	RO	0x01 (1_{dec})
1A0A:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x0A ())	UINT32	RO	0x6001:0A, 16

Index 1A0B Analog Input TxPDO-MapSamples 11 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0B:0	Analog Input TxPDO-MapSamples 11 Ch.1	PDO Mapping TxPDO 12	UINT8	RO	0x01 (1_{dec})
1A0B:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x0B ())	UINT32	RO	0x6001:0B, 16

Index 1A0C Analog Input TxPDO-MapSamples 12 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0C:0	Analog Input TxPDO-MapSamples 12 Ch.1	PDO Mapping TxPDO 13	UINT8	RO	0x01 (1_{dec})
1A0C:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x0C ())	UINT32	RO	0x6001:0C, 16

Index 1A0D Analog Input TxPDO-MapSamples 13 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0D:0	Analog Input TxPDO-MapSamples 13 Ch.1	PDO Mapping TxPDO 14	UINT8	RO	0x01 (1_{dec})
1A0D:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x0D ())	UINT32	RO	0x6001:0D, 16

Index 1A0E Analog Input TxPDO-MapSamples 14 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0E:0	Analog Input TxPDO-MapSamples 14 Ch.1	PDO Mapping TxPDO 15	UINT8	RO	0x01 (1_{dec})
1A0E:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x0E ())	UINT32	RO	0x6001:0E, 16

Index 1A0F Analog Input TxPDO-MapSamples 15 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0F:0	Analog Input TxPDO-MapSamples 15 Ch.1	PDO Mapping TxPDO 16	UINT8	RO	0x01 (1_{dec})
1A0F:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x0F ())	UINT32	RO	0x6001:0F, 16

Index 1A10 Analog Input TxPDO-MapSamples 16 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A10:0	Analog Input TxPDO-MapSamples 16 Ch.1	PDO Mapping TxPDO 17	UINT8	RO	0x01 (1_{dec})
1A10:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x10 ())	UINT32	RO	0x6001:10, 16

Index 1A11 Analog Input TxPDO-MapSamples 17 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A11:0	Analog Input TxPDO-MapSamples 17 Ch.1	PDO Mapping TxPDO 18	UINT8	RO	0x01 (1_{dec})
1A11:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x11 ())	UINT32	RO	0x6001:11, 16

Index 1A12 Analog Input TxPDO-MapSamples 18 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A12:0	Analog Input TxPDO-MapSamples 18 Ch.1	PDO Mapping TxPDO 19	UINT8	RO	0x01 (1_{dec})
1A12:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x12 ())	UINT32	RO	0x6001:12, 16

Index 1A13 Analog Input TxPDO-MapSamples 19 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A13:0	Analog Input TxPDO-MapSamples 19 Ch.1	PDO Mapping TxPDO 20	UINT8	RO	0x01 (1_{dec})
1A13:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x13 ())	UINT32	RO	0x6001:13, 16

Index 1A14 Analog Input TxPDO-MapSamples 20 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A14:0	Analog Input TxPDO-MapSamples 20 Ch.1	PDO Mapping TxPDO 21	UINT8	RO	0x01 (1_{dec})
1A14:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x14 ())	UINT32	RO	0x6001:14, 16

Index 1A15 Analog Input TxPDO-MapSamples 21 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A15:0	Analog Input TxPDO-MapSamples 21 Ch.1	PDO Mapping TxPDO 22	UINT8	RO	0x01 (1_{dec})
1A15:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x15 ())	UINT32	RO	0x6001:15, 16

Index 1A16 Analog Input TxPDO-MapSamples 22 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A16:0	Analog Input TxPDO-MapSamples 22 Ch.1	PDO Mapping TxPDO 23	UINT8	RO	0x01 (1_{dec})
1A16:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x16 ())	UINT32	RO	0x6001:16, 16

Index 1A17 Analog Input TxPDO-MapSamples 23 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A17:0	Analog Input TxPDO-MapSamples 23 Ch.1	PDO Mapping TxPDO 24	UINT8	RO	0x01 (1_{dec})
1A17:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x17 ())	UINT32	RO	0x6001:17, 16

Index 1A18 Analog Input TxPDO-MapSamples 24 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A18:0	Analog Input TxPDO-MapSamples 24 Ch.1	PDO Mapping TxPDO 25	UINT8	RO	0x01 (1_{dec})
1A18:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x18 ())	UINT32	RO	0x6001:18, 16

Index 1A19 Analog Input TxPDO-MapSamples 25 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A19:0	Analog Input TxPDO-MapSamples 25 Ch.1	PDO Mapping TxPDO 26	UINT8	RO	0x01 (1_{dec})
1A19:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x19 ())	UINT32	RO	0x6001:19, 16

Index 1A1A Analog Input TxPDO-MapSamples 26 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1A:0	Analog Input TxPDO-MapSamples 26 Ch.1	PDO Mapping TxPDO 27	UINT8	RO	0x01 (1_{dec})
1A1A:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x1A ())	UINT32	RO	0x6001:1A, 16

Index 1A1B Analog Input TxPDO-MapSamples 27 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1B:0	Analog Input TxPDO-MapSamples 27 Ch.1	PDO Mapping TxPDO 28	UINT8	RO	0x01 (1_{dec})
1A1B:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x1B ())	UINT32	RO	0x6001:1B, 16

Index 1A1C Analog Input TxPDO-MapSamples 28 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1C:0	Analog Input TxPDO-MapSamples 28 Ch.1	PDO Mapping TxPDO 29	UINT8	RO	0x01 (1_{dec})
1A1C:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x1C ())	UINT32	RO	0x6001:1C, 16

Index 1A1D Analog Input TxPDO-MapSamples 29 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1D:0	Analog Input TxPDO-MapSamples 29 Ch.1	PDO Mapping TxPDO 30	UINT8	RO	0x01 (1_{dec})
1A1D:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x1D ())	UINT32	RO	0x6001:1D, 16

Index 1A1E Analog Input TxPDO-MapSamples 30 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1E:0	Analog Input TxPDO-MapSamples 30 Ch.1	PDO Mapping TxPDO 31	UINT8	RO	0x01 (1_{dec})
1A1E:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x1E ())	UINT32	RO	0x6001:1E, 16

Index 1A1F Analog Input TxPDO-MapSamples 31 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1F:0	Analog Input TxPDO-MapSamples 31 Ch.1	PDO Mapping TxPDO 32	UINT8	RO	0x01 (1_{dec})
1A1F:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x1F ())	UINT32	RO	0x6001:1F, 16

Index 1A20 Analog Input TxPDO-MapSamples 32 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A20:0	Analog Input TxPDO-MapSamples 32 Ch.1	PDO Mapping TxPDO 33	UINT8	RO	0x01 (1_{dec})
1A20:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x20 ())	UINT32	RO	0x6001:20, 16

Index 1A21 Analog Input TxPDO-MapSamples 33 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A21:0	Analog Input TxPDO-MapSamples 33 Ch.1	PDO Mapping TxPDO 34	UINT8	RO	0x01 (1_{dec})
1A21:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x21 ())	UINT32	RO	0x6001:21, 16

Index 1A22 Analog Input TxPDO-MapSamples 34 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A22:0	Analog Input TxPDO-MapSamples 34 Ch.1	PDO Mapping TxPDO 35	UINT8	RO	0x01 (1_{dec})
1A22:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x22 ())	UINT32	RO	0x6001:22, 16

Index 1A23 Analog Input TxPDO-MapSamples 35 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A23:0	Analog Input TxPDO-MapSamples 35 Ch.1	PDO Mapping TxPDO 36	UINT8	RO	0x01 (1_{dec})
1A23:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x23 ())	UINT32	RO	0x6001:23, 16

Index 1A24 Analog Input TxPDO-MapSamples 36 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A24:0	Analog Input TxPDO-MapSamples 36 Ch.1	PDO Mapping TxPDO 37	UINT8	RO	0x01 (1_{dec})
1A24:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x24 ())	UINT32	RO	0x6001:24, 16

Index 1A25 Analog Input TxPDO-MapSamples 37 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A25:0	Analog Input TxPDO-MapSamples 37 Ch.1	PDO Mapping TxPDO 38	UINT8	RO	0x01 (1_{dec})
1A25:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x25 ())	UINT32	RO	0x6001:25, 16

Index 1A26 Analog Input TxPDO-MapSamples 38 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A26:0	Analog Input TxPDO-MapSamples 38 Ch.1	PDO Mapping TxPDO 39	UINT8	RO	0x01 (1_{dec})
1A26:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x26 ())	UINT32	RO	0x6001:26, 16

Index 1A27 Analog Input TxPDO-MapSamples 39 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A27:0	Analog Input TxPDO-MapSamples 39 Ch.1	PDO Mapping TxPDO 40	UINT8	RO	0x01 (1_{dec})
1A27:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x27 ())	UINT32	RO	0x6001:27, 16

Index 1A28 Analog Input TxPDO-MapSamples 40 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A28:0	Analog Input TxPDO-MapSamples 40 Ch.1	PDO Mapping TxPDO 41	UINT8	RO	0x01 (1_{dec})
1A28:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x28 ())	UINT32	RO	0x6001:28, 16

Index 1A29 Analog Input TxPDO-MapSamples 41 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A29:0	Analog Input TxPDO-MapSamples 41 Ch.1	PDO Mapping TxPDO 42	UINT8	RO	0x01 (1_{dec})
1A29:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x29 ())	UINT32	RO	0x6001:29, 16

Index 1A2A Analog Input TxPDO-MapSamples 42 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A2A:0	Analog Input TxPDO-MapSamples 42 Ch.1	PDO Mapping TxPDO 43	UINT8	RO	0x01 (1_{dec})
1A2A:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x2A ())	UINT32	RO	0x6001:2A, 16

Index 1A2B Analog Input TxPDO-MapSamples 43 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A2B:0	Analog Input TxPDO-MapSamples 43 Ch.1	PDO Mapping TxPDO 44	UINT8	RO	0x01 (1_{dec})
1A2B:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x2B ())	UINT32	RO	0x6001:2B, 16

Index 1A2C Analog Input TxPDO-MapSamples 44 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A2C:0	Analog Input TxPDO-MapSamples 44 Ch.1	PDO Mapping TxPDO 45	UINT8	RO	0x01 (1_{dec})
1A2C:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x2C ())	UINT32	RO	0x6001:2C, 16

Index 1A2D Analog Input TxPDO-MapSamples 45 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A2D:0	Analog Input TxPDO-MapSamples 45 Ch.1	PDO Mapping TxPDO 46	UINT8	RO	0x01 (1_{dec})
1A2D:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x2D ())	UINT32	RO	0x6001:2D, 16

Index 1A2E Analog Input TxPDO-MapSamples 46 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A2E:0	Analog Input TxPDO-MapSamples 46 Ch.1	PDO Mapping TxPDO 47	UINT8	RO	0x01 (1_{dec})
1A2E:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x2E ())	UINT32	RO	0x6001:2E, 16

Index 1A2F Analog Input TxPDO-MapSamples 47 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A2F:0	Analog Input TxPDO-MapSamples 47 Ch.1	PDO Mapping TxPDO 48	UINT8	RO	0x01 (1_{dec})
1A2F:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x2F ())	UINT32	RO	0x6001:2F, 16

Index 1A30 Analog Input TxPDO-MapSamples 48 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A30:0	Analog Input TxPDO-MapSamples 48 Ch.1	PDO Mapping TxPDO 49	UINT8	RO	0x01 (1_{dec})
1A30:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x30 ())	UINT32	RO	0x6001:30, 16

Index 1A31 Analog Input TxPDO-MapSamples 49 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A31:0	Analog Input TxPDO-MapSamples 49 Ch.1	PDO Mapping TxPDO 50	UINT8	RO	0x01 (1_{dec})
1A31:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x31 ())	UINT32	RO	0x6001:31, 16

Index 1A32 Analog Input TxPDO-MapSamples 50 Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A32:0	Analog Input TxPDO-MapSamples 50 Ch.1	PDO Mapping TxPDO 51	UINT8	RO	0x01 (1_{dec})
1A32:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (Samples), entry 0x32 ())	UINT32	RO	0x6001:32, 16

Index 1A40 Analog Input TxPDO-MapStatus Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A40:0	Analog Input TxPDO-MapStatus Ch.2	PDO Mapping TxPDO 65	UINT8	RO	0x09 (9_{dec})
1A40:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (Status), entry 0x01 (Underrange))	UINT32	RO	0x6010:01, 1
1A40:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (Status), entry 0x02 (Overrange))	UINT32	RO	0x6010:02, 1
1A40:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (Status), entry 0x03 (Limit 1))	UINT32	RO	0x6010:03, 2
1A40:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (Status), entry 0x05 (Limit 2))	UINT32	RO	0x6010:05, 2
1A40:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (Status), entry 0x07 (Error))	UINT32	RO	0x6010:07, 1
1A40:06	SubIndex 006	6. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A40:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (Status), entry 0x0E (Sync error))	UINT32	RO	0x6010:0E, 1
1A40:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (Status), entry 0x0F (TxPDO State))	UINT32	RO	0x6010:0F, 1
1A40:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (Status), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6010:10, 1

Index 1A41 Analog Input TxPDO-MapSamples 1 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A41:0	Analog Input TxPDO-MapSamples 1 Ch.2	PDO Mapping TxPDO 66	UINT8	RO	0x01 (1_{dec})
1A41:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x01 ())	UINT32	RO	0x6011:01, 16

Index 1A42 Analog Input TxPDO-MapSamples 2 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A42:0	Analog Input TxPDO-MapSamples 2 Ch.2	PDO Mapping TxPDO 67	UINT8	RO	0x01 (1_{dec})
1A42:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x02 ())	UINT32	RO	0x6011:02, 16

Index 1A43 Analog Input TxPDO-MapSamples 3 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A43:0	Analog Input TxPDO-MapSamples 3 Ch.2	PDO Mapping TxPDO 68	UINT8	RO	0x01 (1_{dec})
1A43:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x03 ())	UINT32	RO	0x6011:03, 16

Index 1A44 Analog Input TxPDO-MapSamples 4 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A44:0	Analog Input TxPDO-MapSamples 4 Ch.2	PDO Mapping TxPDO 69	UINT8	RO	0x01 (1_{dec})
1A44:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x04 ())	UINT32	RO	0x6011:04, 16

Index 1A45 Analog Input TxPDO-MapSamples 5 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A45:0	Analog Input TxPDO-MapSamples 5 Ch.2	PDO Mapping TxPDO 70	UINT8	RO	0x01 (1_{dec})
1A45:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x05 ())	UINT32	RO	0x6011:05, 16

Index 1A46 Analog Input TxPDO-MapSamples 6 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A46:0	Analog Input TxPDO-MapSamples 6 Ch.2	PDO Mapping TxPDO 71	UINT8	RO	0x01 (1_{dec})
1A46:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x06 ())	UINT32	RO	0x6011:06, 16

Index 1A47 Analog Input TxPDO-MapSamples 7 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A47:0	Analog Input TxPDO-MapSamples 7 Ch.2	PDO Mapping TxPDO 72	UINT8	RO	0x01 (1_{dec})
1A47:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x07 ())	UINT32	RO	0x6011:07, 16

Index 1A48 Analog Input TxPDO-MapSamples 8 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A48:0	Analog Input TxPDO-MapSamples 8 Ch.2	PDO Mapping TxPDO 73	UINT8	RO	0x01 (1_{dec})
1A48:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples),), entry 0x08 ())	UINT32	RO	0x6011:08, 16

Index 1A49 Analog Input TxPDO-MapSamples 9 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A49:0	Analog Input TxPDO-MapSamples 9 Ch.2	PDO Mapping TxPDO 74	UINT8	RO	0x01 (1_{dec})
1A49:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x09 ())	UINT32	RO	0x6011:09, 16

Index 1A4A Analog Input TxPDO-MapSamples 10 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A4A:0	Analog Input TxPDO-MapSamples 10 Ch.2	PDO Mapping TxPDO 75	UINT8	RO	0x01 (1_{dec})
1A4A:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x0A ())	UINT32	RO	0x6011:0A, 16

Index 1A4B Analog Input TxPDO-MapSamples 11 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A4B:0	Analog Input TxPDO-MapSamples 11 Ch.2	PDO Mapping TxPDO 76	UINT8	RO	0x01 (1_{dec})
1A4B:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x0B ())	UINT32	RO	0x6011:0B, 16

Index 1A4C Analog Input TxPDO-MapSamples 12 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A4C:0	Analog Input TxPDO-MapSamples 12 Ch.2	PDO Mapping TxPDO 77	UINT8	RO	0x01 (1_{dec})
1A4C:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x0C ())	UINT32	RO	0x6011:0C, 16

Index 1A4D Analog Input TxPDO-MapSamples 13 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A4D:0	Analog Input TxPDO-MapSamples 13 Ch.2	PDO Mapping TxPDO 78	UINT8	RO	0x01 (1_{dec})
1A4D:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x0D ())	UINT32	RO	0x6011:0D, 16

Index 1A4E Analog Input TxPDO-MapSamples 14 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A4E:0	Analog Input TxPDO-MapSamples 14 Ch.2	PDO Mapping TxPDO 79	UINT8	RO	0x01 (1_{dec})
1A4E:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x0E ())	UINT32	RO	0x6011:0E, 16

Index 1A4F Analog Input TxPDO-MapSamples 15 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A4F:0	Analog Input TxPDO-MapSamples 15 Ch.2	PDO Mapping TxPDO 80	UINT8	RO	0x01 (1_{dec})
1A4F:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x0F ())	UINT32	RO	0x6011:0F, 16

Index 1A50 Analog Input TxPDO-MapSamples 16 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A50:0	Analog Input TxPDO-MapSamples 16 Ch.2	PDO Mapping TxPDO 81	UINT8	RO	0x01 (1_{dec})
1A50:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x10 ())	UINT32	RO	0x6011:10, 16

Index 1A51 Analog Input TxPDO-MapSamples 17 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A51:0	Analog Input TxPDO-MapSamples 17 Ch.2	PDO Mapping TxPDO 82	UINT8	RO	0x01 (1_{dec})
1A51:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x11 ())	UINT32	RO	0x6011:11, 16

Index 1A52 Analog Input TxPDO-MapSamples 18 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A52:0	Analog Input TxPDO-MapSamples 18 Ch.2	PDO Mapping TxPDO 83	UINT8	RO	0x01 (1_{dec})
1A52:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x12 ())	UINT32	RO	0x6011:12, 16

Index 1A53 Analog Input TxPDO-MapSamples 19 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A53:0	Analog Input TxPDO-MapSamples 19 Ch.2	PDO Mapping TxPDO 84	UINT8	RO	0x01 (1_{dec})
1A53:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x13 ())	UINT32	RO	0x6011:13, 16

Index 1A54 Analog Input TxPDO-MapSamples 20 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A54:0	Analog Input TxPDO-MapSamples 20 Ch.2	PDO Mapping TxPDO 85	UINT8	RO	0x01 (1_{dec})
1A54:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x14 ())	UINT32	RO	0x6011:14, 16

Index 1A55 Analog Input TxPDO-MapSamples 21 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A55:0	Analog Input TxPDO-MapSamples 21 Ch.2	PDO Mapping TxPDO 86	UINT8	RO	0x01 (1_{dec})
1A55:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x15 ())	UINT32	RO	0x6011:15, 16

Index 1A56 Analog Input TxPDO-MapSamples 22 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A56:0	Analog Input TxPDO-MapSamples 22 Ch.2	PDO Mapping TxPDO 87	UINT8	RO	0x01 (1_{dec})
1A56:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x16 ())	UINT32	RO	0x6011:16, 16

Index 1A57 Analog Input TxPDO-MapSamples 23 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A57:0	Analog Input TxPDO-MapSamples 23 Ch.2	PDO Mapping TxPDO 88	UINT8	RO	0x01 (1_{dec})
1A57:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x17 ())	UINT32	RO	0x6011:17, 16

Index 1A58 Analog Input TxPDO-MapSamples 24 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A58:0	Analog Input TxPDO-MapSamples 24 Ch.2	PDO Mapping TxPDO 89	UINT8	RO	0x01 (1_{dec})
1A58:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x18 ())	UINT32	RO	0x6011:18, 16

Index 1A59 Analog Input TxPDO-MapSamples 25 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A59:0	Analog Input TxPDO-MapSamples 25 Ch.2	PDO Mapping TxPDO 90	UINT8	RO	0x01 (1_{dec})
1A59:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x19 ())	UINT32	RO	0x6011:19, 16

Index 1A5A Analog Input TxPDO-MapSamples 26 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A5A:0	Analog Input TxPDO-MapSamples 26 Ch.2	PDO Mapping TxPDO 91	UINT8	RO	0x01 (1_{dec})
1A5A:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x1A ())	UINT32	RO	0x6011:1A, 16

Index 1A5B Analog Input TxPDO-MapSamples 27 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A5B:0	Analog Input TxPDO-MapSamples 27 Ch.2	PDO Mapping TxPDO 92	UINT8	RO	0x01 (1_{dec})
1A5B:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x1B ())	UINT32	RO	0x6011:1B, 16

Index 1A5C Analog Input TxPDO-MapSamples 28 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A5C:0	Analog Input TxPDO-MapSamples 28 Ch.2	PDO Mapping TxPDO 93	UINT8	RO	0x01 (1_{dec})
1A5C:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x1C ())	UINT32	RO	0x6011:1C, 16

Index 1A5D Analog Input TxPDO-MapSamples 29 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A5D:0	Analog Input TxPDO-MapSamples 29 Ch.2	PDO Mapping TxPDO 94	UINT8	RO	0x01 (1_{dec})
1A5D:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x1D ())	UINT32	RO	0x6011:1D, 16

Index 1A5E Analog Input TxPDO-MapSamples 30 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A5E:0	Analog Input TxPDO-MapSamples 30 Ch.2	PDO Mapping TxPDO 95	UINT8	RO	0x01 (1_{dec})
1A5E:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x1E ())	UINT32	RO	0x6011:1E, 16

Index 1A5F Analog Input TxPDO-MapSamples 31 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A5F:0	Analog Input TxPDO-MapSamples 31 Ch.2	PDO Mapping TxPDO 96	UINT8	RO	0x01 (1_{dec})
1A5F:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x1F ())	UINT32	RO	0x6011:1F, 16

Index 1A60 Analog Input TxPDO-MapSamples 32 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A60:0	Analog Input TxPDO-MapSamples 32 Ch.2	PDO Mapping TxPDO 97	UINT8	RO	0x01 (1_{dec})
1A60:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x20 ())	UINT32	RO	0x6011:20, 16

Index 1A61 Analog Input TxPDO-MapSamples 33 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A61:0	Analog Input TxPDO-MapSamples 33 Ch.2	PDO Mapping TxPDO 98	UINT8	RO	0x01 (1_{dec})
1A61:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x21 ())	UINT32	RO	0x6011:21, 16

Index 1A62 Analog Input TxPDO-MapSamples 34 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A62:0	Analog Input TxPDO-MapSamples 34 Ch.2	PDO Mapping TxPDO 99	UINT8	RO	0x01 (1_{dec})
1A62:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x22 ())	UINT32	RO	0x6011:22, 16

Index 1A63 Analog Input TxPDO-MapSamples 35 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A63:0	Analog Input TxPDO-MapSamples 35 Ch.2	PDO Mapping TxPDO 100	UINT8	RO	0x01 (1_{dec})
1A63:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x23 ())	UINT32	RO	0x6011:23, 16

Index 1A64 Analog Input TxPDO-MapSamples 36 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A64:0	Analog Input TxPDO-MapSamples 36 Ch.2	PDO Mapping TxPDO 101	UINT8	RO	0x01 (1_{dec})
1A64:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x24 ())	UINT32	RO	0x6011:24, 16

Index 1A65 Analog Input TxPDO-MapSamples 37 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A65:0	Analog Input TxPDO-MapSamples 37 Ch.2	PDO Mapping TxPDO 102	UINT8	RO	0x01 (1_{dec})
1A65:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x25 ())	UINT32	RO	0x6011:25, 16

Index 1A66 Analog Input TxPDO-MapSamples 38 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A66:0	Analog Input TxPDO-MapSamples 38 Ch.2	PDO Mapping TxPDO 103	UINT8	RO	0x01 (1_{dec})
1A66:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x26 ())	UINT32	RO	0x6011:26, 16

Index 1A67 Analog Input TxPDO-MapSamples 39 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A67:0	Analog Input TxPDO-MapSamples 39 Ch.2	PDO Mapping TxPDO 104	UINT8	RO	0x01 (1_{dec})
1A67:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x27 ())	UINT32	RO	0x6011:27, 16

Index 1A68 Analog Input TxPDO-MapSamples 40 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A68:0	Analog Input TxPDO-MapSamples 40 Ch.2	PDO Mapping TxPDO 105	UINT8	RO	0x01 (1_{dec})
1A68:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x28 ())	UINT32	RO	0x6011:28, 16

Index 1A69 Analog Input TxPDO-MapSamples 41 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A69:0	Analog Input TxPDO-MapSamples 41 Ch.2	PDO Mapping TxPDO 106	UINT8	RO	0x01 (1_{dec})
1A69:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x29 ())	UINT32	RO	0x6011:29, 16

Index 1A6A Analog Input TxPDO-MapSamples 42 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A6A:0	Analog Input TxPDO-MapSamples 42 Ch.2	PDO Mapping TxPDO 107	UINT8	RO	0x01 (1_{dec})
1A6A:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x2A ())	UINT32	RO	0x6011:2A, 16

Index 1A6B Analog Input TxPDO-MapSamples 43 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A6B:0	Analog Input TxPDO-MapSamples 43 Ch.2	PDO Mapping TxPDO 108	UINT8	RO	0x01 (1_{dec})
1A6B:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x2B ())	UINT32	RO	0x6011:2B, 16

Index 1A6C Analog Input TxPDO-MapSamples 44 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A6C:0	Analog Input TxPDO-MapSamples 44 Ch.2	PDO Mapping TxPDO 109	UINT8	RO	0x01 (1_{dec})
1A6C:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x2C ())	UINT32	RO	0x6011:2C, 16

Index 1A6D Analog Input TxPDO-MapSamples 45 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A6D:0	Analog Input TxPDO-MapSamples 45 Ch.2	PDO Mapping TxPDO 110	UINT8	RO	0x01 (1_{dec})
1A6D:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x2D ())	UINT32	RO	0x6011:2D, 16

Index 1A6E Analog Input TxPDO-MapSamples 46 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A6E:0	Analog Input TxPDO-MapSamples 46 Ch.2	PDO Mapping TxPDO 111	UINT8	RO	0x01 (1_{dec})
1A6E:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x2E ())	UINT32	RO	0x6011:2E, 16

Index 1A6F Analog Input TxPDO-MapSamples 47 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A6F:0	Analog Input TxPDO-MapSamples 47 Ch.2	PDO Mapping TxPDO 112	UINT8	RO	0x01 (1_{dec})
1A6F:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x2F ())	UINT32	RO	0x6011:2F, 16

Index 1A70 Analog Input TxPDO-MapSamples 48 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A70:0	Analog Input TxPDO-MapSamples 48 Ch.2	PDO Mapping TxPDO 113	UINT8	RO	0x01 (1_{dec})
1A70:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x30 ())	UINT32	RO	0x6011:30, 16

Index 1A71 Analog Input TxPDO-MapSamples 49 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A71:0	Analog Input TxPDO-MapSamples 49 Ch.2	PDO Mapping TxPDO 114	UINT8	RO	0x01 (1_{dec})
1A71:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x31 ())	UINT32	RO	0x6011:31, 16

Index 1A72 Analog Input TxPDO-MapSamples 50 Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A72:0	Analog Input TxPDO-MapSamples 50 Ch.2	PDO Mapping TxPDO 115	UINT8	RO	0x01 (1_{dec})
1A72:01	SubIndex 001	1. PDO Mapping entry (object 0x6011 (Samples), entry 0x32 ())	UINT32	RO	0x6011:32, 16

Index 1A80 Analog Input Timestamp TxPDO-Map NextSync1Time

Index (hex)	Name	Meaning	Data type	Flags	Default
1A80:0	Analog Input Time-stamp TxPDO-Map NextSync1Time	PDO Mapping TxPDO 129	UINT8	RO	0x01 (1_{dec})
1A80:01	SubIndex 001	1. PDO Mapping entry (object 0x6020 (NextSync1Time), entry 0x01 ())	UINT32	RO	0x6020:01, 64

Index 1A81 Analog Input Timestamp TxPDO-Map Sample Count

Index (hex)	Name	Meaning	Data type	Flags	Default
1A81:0	Analog Input Time-stamp TxPDO-Map Sample Count	PDO Mapping TxPDO 130	UINT8	RO	0x01 (1_{dec})
1A81:01	SubIndex 001	1. PDO Mapping entry (object 0x6021 (SampleCount), entry 0x01 (Sample Count))	UINT32	RO	0x6021:01, 16

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x00 (0_{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x06 (6_{dec})
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	Subindex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A01 (6657_{dec})
1C13:03	Subindex 003	3. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A40 (6720_{dec})
1C13:04	Subindex 004	4. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A41 (6721_{dec})
1C13:05	Subindex 005	5. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A80 (6784_{dec})
1C13:06	Subindex 006	6. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A81 (6785_{dec})
1C13:07	Subindex 007	7. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})
1C13:69	Subindex 105	105. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 Event (no outputs available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 Event (outputs available)	UINT16	RW	0x0022 (34_{dec})
1C33:02	Cycle time	as 0x1C32:02	UINT32	RW	0x000F4240 (1000000_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: Synchronous with SM 2 Event is supported (outputs available) • Bit 1: Synchronous with SM 3 Event is supported (no outputs available) • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 01: input shift through local event (outputs available) • Bit 4-5 = 10: input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08 or 0x1C33:08)	UINT16	RO	0x0C06 (3078_{dec})
1C33:05	Minimum cycle time	as 0x1C32:05	UINT32	RO	0x00002710 (10000_{dec})
1C33:06	Calc and copy time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:07	Minimum delay time		UINT32	RO	0x00000000 (0_{dec})
1C33:08	Command	as 0x1C32:08	UINT16	RW	0x0000 (0_{dec})
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:0B	SM event missed counter	as 0x1C32:11	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	as 0x1C32:13	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	as 0x1C32:32	BOOLEAN	RO	0x00 (0_{dec})

Index F000 Modular device profile

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

Index F008 Code word

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

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Maximum subindex	UINT8	RW	0x03 (3_{dec})
F010:01	SubIndex 001	Profile AI	UINT32	RW	0x0000012C (300_{dec})
F010:02	SubIndex 002	Profile AI	UINT32	RW	0x0000012C (300_{dec})
F010:03	SubIndex 003	Profile AI	UINT32	RW	0x0000012C (300_{dec})

4.5 Notices on analog specifications

Beckhoff I/O devices (terminals, boxes, modules) with analog inputs are characterized by a number of technical characteristic data; refer to the technical data in the respective documents.

Some explanations are given below for the correct interpretation of these characteristic data.

4.5.1 Full scale value (FSV)

An I/O device with an analog input measures over a nominal measuring range that is limited by an upper and a lower limit (initial value and end value); these can usually be taken from the device designation.
The range between the two limits is called the measuring span and corresponds to the equation (end value - initial value). Analogous to pointing devices this is the measuring scale (see IEC 61131) or also the dynamic range.

For analog I/O devices from Beckhoff the rule is that the limit with the largest value is chosen as the full scale value of the respective product (also called the reference value) and is given a positive sign. This applies to both symmetrical and asymmetrical measuring spans.

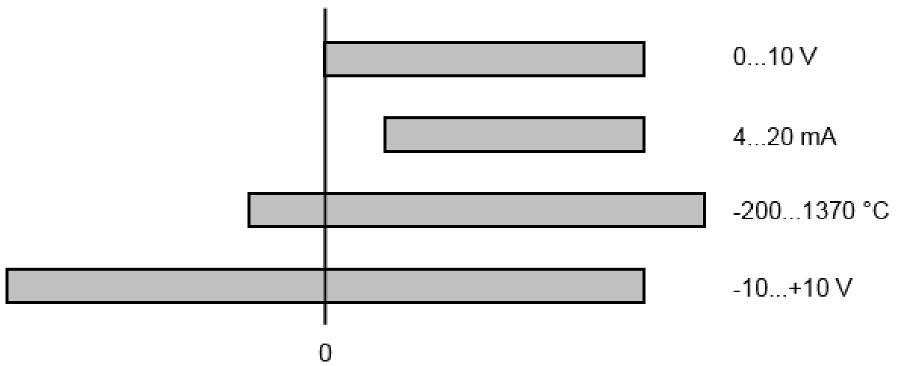


Fig. 73: Full scale value, measuring span

For the above **examples** this means:

- Measuring range 0...10 V: asymmetric unipolar, full scale value = 10 V, measuring span = 10 V
- Measuring range 4...20 mA: asymmetric unipolar, full scale value = 20 mA, measuring span = 16 mA
- Measuring range -200...1370°C: asymmetric bipolar, full scale value = 1370°C, measuring span = 1570°C
- Measuring range -10...+10 V: symmetric bipolar, full scale value = 10 V, measuring span = 20 V

This applies to analog output terminals/ boxes (and related Beckhoff product groups).

4.5.2 Measuring error/ measurement deviation

The relative measuring error (% of the full scale value) is referenced to the full scale value and is calculated as the quotient of the largest numerical deviation from the true value (‘measuring error’) referenced to the full scale value.

$$\text{Measuring error} = \frac{|\text{max. deviation}|}{\text{full scale value}}$$

The measuring error is generally valid for the entire permitted operating temperature range, also called the ‘usage error limit’ and contains random and systematic portions of the referred device (i.e. ‘all’ influences such as temperature, inherent noise, aging, etc.).

It is always to be regarded as a positive/negative span with ±, even if it is specified without ± in some cases.

The maximum deviation can also be specified directly.

Example: Measuring range 0...10 V and measuring error < ± 0.3 % full scale value → maximum deviation ± 30 mV in the permissible operating temperature range.

Lower measuring error

Since this specification also includes the temperature drift, a significantly lower measuring error can usually be assumed in case of a constant ambient temperature of the device and thermal stabilization after a user calibration.

This applies to analog output devices.

4.5.3 Temperature coefficient tK [ppm/K]

An electronic circuit is usually temperature dependent to a greater or lesser degree. In analog measurement technology this means that when a measured value is determined by means of an electronic circuit, its deviation from the "true" value is reproducibly dependent on the ambient/operating temperature.

A manufacturer can alleviate this by using components of a higher quality or by software means.

The temperature coefficient, when indicated, specified by Beckhoff allows the user to calculate the expected measuring error outside the basic accuracy at 23 °C.

Due to the extensive uncertainty considerations that are incorporated in the determination of the basic accuracy (at 23 °C), Beckhoff recommends a quadratic summation.

Example: Let the basic accuracy at 23 °C be ±0.01% typ. (full scale value), tK = 20 ppm/K typ.; the accuracy A35 at 35 °C is wanted, hence ΔT = 12 K

$$G35 = \sqrt{(0.01\%)^2 + (12K \cdot 20\,\frac{ppm}{K})^2} = 0.026\%\ \text{full scale value, typ}$$

Remarks: ppm $\triangleq 10^{-6}$ % $\triangleq 10^{-2}$

4.5.4 Single-ended/differential typification

For analog inputs Beckhoff makes a basic distinction between two types: *single-ended* (SE) and *differential (DIFF)*, referring to the difference in electrical connection with regard to the potential difference.

The diagram shows two-channel versions of an SE module and a DIFF module as examples for all multi-channel versions.

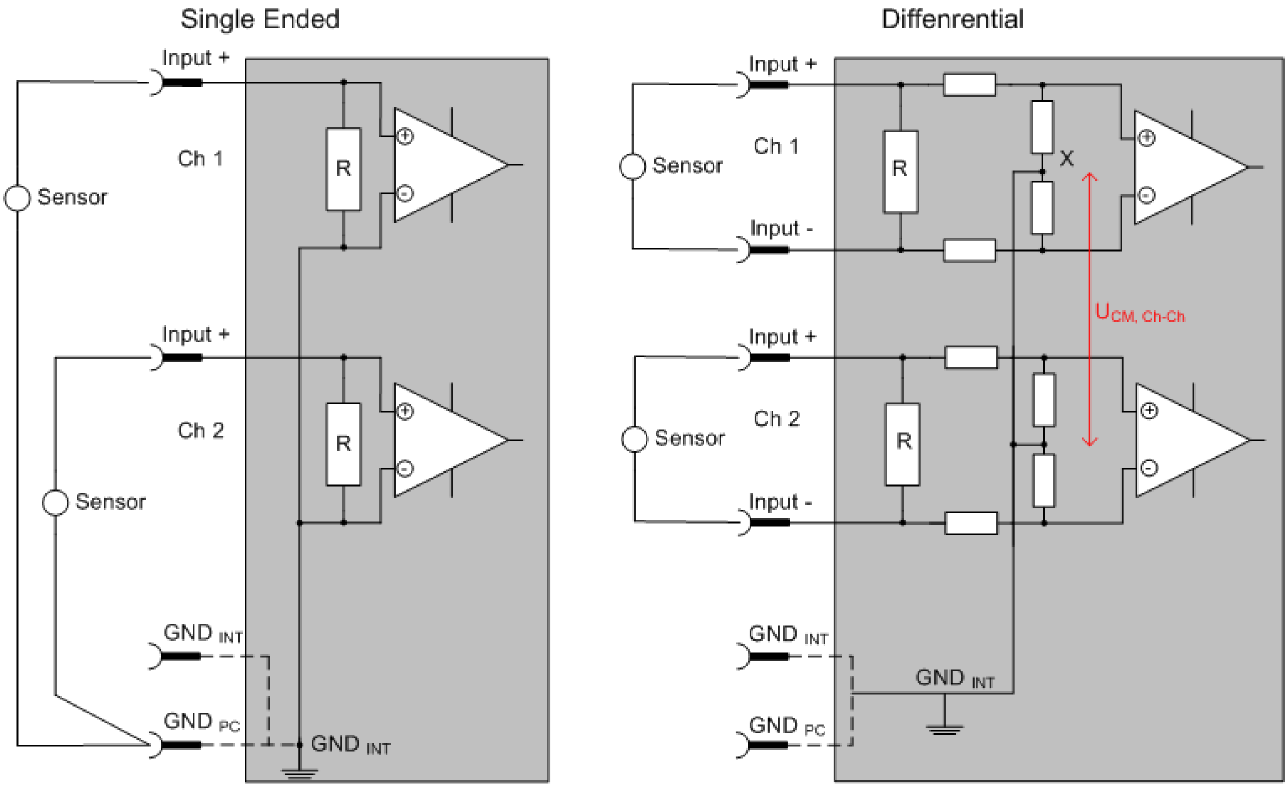


Fig. 74: SE and DIFF module as 2-channel version

Note: Dashed lines indicate that the respective connection may not necessarily be present in each SE or DIFF module. Electrical isolated channels are operating as differential type in general, hence there is no direct relation (voltaic) to ground within the module established at all. Indeed, specified information to recommended and maximum voltage levels have to be taken into account.

The basic rule:

- Analog measurements always take the form of voltage measurements between two potential points. For voltage measurements a large R is used, in order to ensure a high impedance. For current measurements a small R is used as shunt. If the purpose is resistance measurement, corresponding considerations are applied.
  - Beckhoff generally refers to these two points as input+/signal potential and input-/reference potential.
  - For measurements between two potential points two potentials have to be supplied.
  - Regarding the terms "single-wire connection" or "three-wire connection", please note the following for pure analog measurements: three- or four-wire connections can be used for sensor supply, but are not involved in the actual analog measurement, which always takes place between two potentials/wires.
    In particular this also applies to SE, even though the term suggest that only one wire is required.
- The term "electrical isolation" should be clarified in advance.
  Beckhoff IO modules feature 1..8 or more analog channels; with regard to the channel connection a distinction is made in terms of:
  - how the channels WITHIN a module relate to each other, or
  - how the channels of SEVERAL modules relate to each other.

The property of electrical isolation indicates whether the channels are directly connected to each other.

  - Beckhoff terminals/ boxes (and related product groups) always feature electrical isolation between the field/analog side and the bus/EtherCAT side. In other words, if two analog terminals/ boxes are not connected via the power contacts (cable), the modules are effectively electrically isolated.
  - If channels within a module are electrically isolated, or if a single-channel module has no power contacts, the channels are effectively always differential. See also explanatory notes below. Differential channels are not necessarily electrically isolated.
- Analog measuring channels are subject to technical limits, both in terms of the recommended operating range (continuous operation) and the destruction limit. Please refer to the respective terminal/ box documentation for further details.

Explanation

- **differential (DIFF)**
  - Differential measurement is the most flexible concept. The user can freely choose both connection points, input+/signal potential and input-/reference potential, within the framework of the technical specification.
  - A differential channel can also be operated as SE, if the reference potential of several sensors is linked. This interconnection may take place via the system GND.
  - Since a differential channel is configured symmetrically internally (cf. Fig. SE and DIFF module as 2-channel variant), there will be a mid-potential (X) between the two supplied potentials that is the same as the internal ground/reference ground for this channel. If several DIFF channels are used in a module without electrical isolation, the technical property V_{CM} (common-mode voltage) indicates the degree to which the mean voltage of the channels may differ.
  - The internal reference ground may be accessible as connection point at the terminal/ box, in order to stabilize a defined GND potential in the terminal/ box. In this case it is particularly important to pay attention to the quality of this potential (noiselessness, voltage stability). At this GND point a wire may be connected to make sure that $V_{CM,max}$ is not exceeded in the differential sensor cable. If differential channels are not electrically isolated, usually only one $V_{CM, max}$ is permitted. If the channels are electrically isolated this limit should not apply, and the channels voltages may differ up to the specified separation limit.
  - Differential measurement in combination with correct sensor wiring has the special advantage that any interference affecting the sensor cable (ideally the feed and return line are arranged side by side, so that interference signals have the same effect on both wires) has very little effect on the measurement, since the potential of both lines varies jointly (hence the term common mode). In simple terms: Common-mode interference has the same effect on both wires in terms of amplitude and phasing.
  - Nevertheless, the suppression of common-mode interference within a channel or between channels is subject to technical limits, which are specified in the technical data.
  - Further helpfully information on this topic can be found on the documentation page *Configuration of 0/4..20 mA differential inputs* (see documentation for the EL30xx terminals, for example).
- **Single Ended (SE)**
  - If the analog circuit is designed as SE, the input/reference wire is internally fixed to a certain potential that cannot be changed. This potential must be accessible from outside on at least one point for connecting the reference potential, e.g. via the power contacts (cable).
  - In other words, in situations with several channels SE offers users the option to avoid returning at least one of the two sensor cables to the terminal/ box (in contrast to DIFF). Instead, the reference wire can be consolidated at the sensors, e.g. in the system GND.
  - A disadvantage of this approach is that the separate feed and return line can result in voltage/ current variations, which a SE channel may no longer be able to handle. See common-mode interference. A V_{CM} effect cannot occur, since the module channels are internally always 'hard-wired' through the input/reference potential.

Typification of the 2/3/4-wire connection of current sensors

Current transducers/sensors/field devices (referred to in the following simply as ‘sensor’) with the industrial 0/4-20 mA interface typically have internal transformation electronics for the physical measured variable (temperature, current, etc.) at the current control output. These internal electronics must be supplied with energy (voltage, current). The type of cable for this supply thus separates the sensors into *self-supplied* or *externally supplied* sensors:

Self-supplied sensors

- The sensor draws the energy for its own operation via the sensor/signal cable + and -.
  So that enough energy is always available for the sensor’s own operation and open-circuit detection is possible, a lower limit of 4 mA has been specified for the 4-20 mA interface; i.e. the sensor allows a minimum current of 4 mA and a maximum current of 20 mA to pass.
- 2-wire connection see Fig. *2-wire connection*, cf. IEC60381-1
- Such current transducers generally represent a current sink and thus like to sit between + and – as a ‘variable load’. Refer also to the sensor manufacturer’s information.

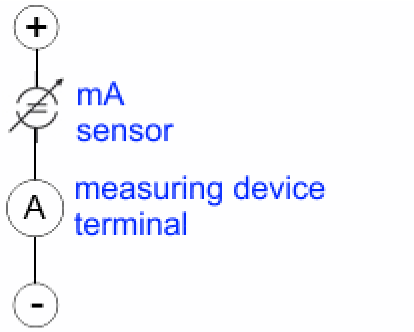


Fig. 75: 2-wire connection

Therefore, they are to be connected according to the Beckhoff terminology as follows:

preferably to **‘single-ended’ inputs** if the +Supply connections of the terminal/ box are also to be used - connect to +Supply and Signal

they can, however, also be connected to **‘differential’ inputs**, if the termination to GND is then manufactured on the application side – to be connected with the right polarity to +Signal and –Signal
It is important to refer to the information page *Configuration of 0/4..20 mA differential inputs* (see documentation for the EL30xx terminals, for example)!

Externally supplied sensors

- 3- and 4-wire connection see Fig. *Connection of externally supplied sensors*, cf. IEC60381-1
- the sensor draws the energy/operating voltage for its own operation from 2 supply cables of its own. One or two further sensor cables are used for the signal transmission of the current loop:
  - 1 sensor cable: according to the Beckhoff terminology such sensors are to be connected to **‘single-ended’ inputs** in 3 cables with +/-/Signal lines and if necessary FE/shield
  - 2 sensor cables: for sensors with 4-wire connection based on +supply/-supply/+signal/-signal, check whether +signal can be connected to +supply or –signal to –supply.

    - Yes: then you can connect accordingly to a Beckhoff **‘single-ended’ input**.

    - No: the Beckhoff **‘differential’ input** for +Signal and –Signal is to be selected; +Supply and – Supply are to be connected via additional cables.
    It is important to refer to the information page *Configuration of 0/4..20 mA differential inputs* (see documentation for the EL30xx terminals, for example)!

Note: expert organizations such as NAMUR demand a usable measuring range <4 mA/>20 mA for error detection and adjustment, see also NAMUR NE043.
The Beckhoff device documentation must be consulted in order to see whether the respective device supports such an extended signal range.
Usually there is an internal diode existing within unipolar terminals/ boxes (and related product groups), in this case the polarity/direction of current have to be observed.

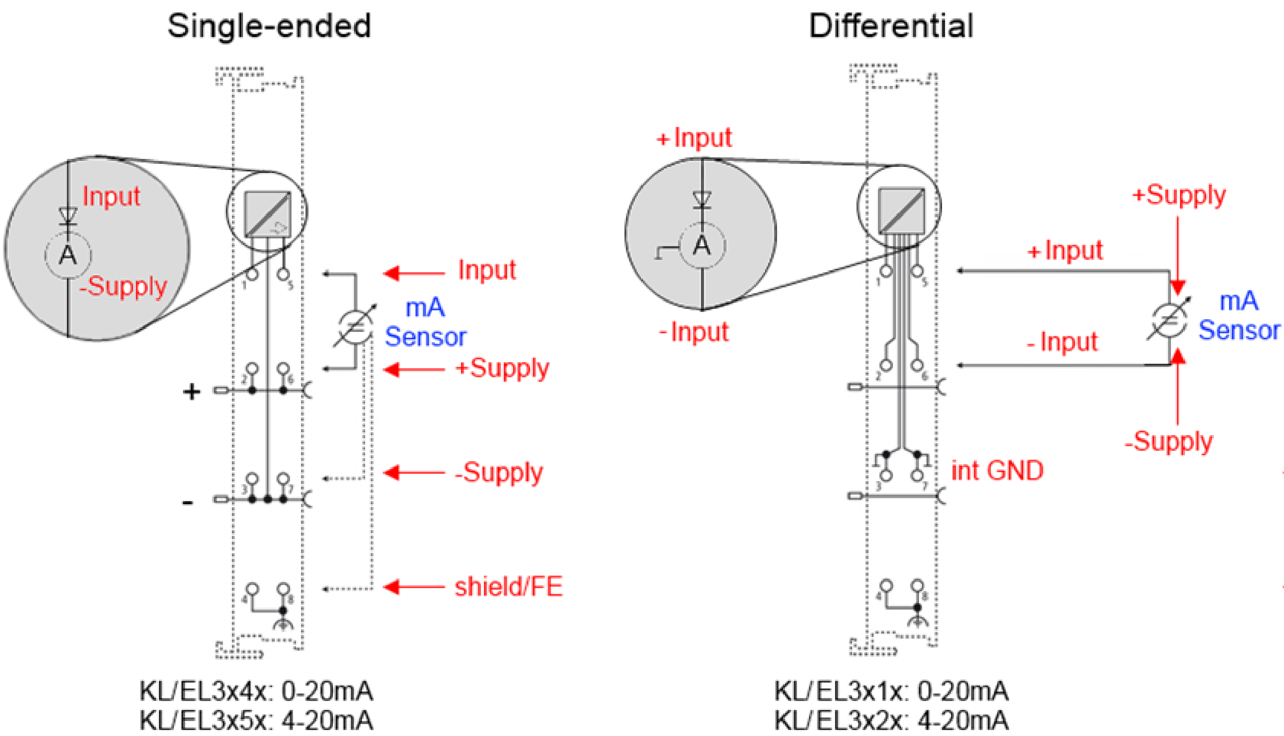


Fig. 76: Connection of externally supplied sensors

Classification of the Beckhoff terminals/ boxes - Beckhoff 0/4-20 mA terminals/ boxes (and related product groups) are available as **differential** and **single-ended** terminals/ boxes (and related product groups):

Single-ended	Differential
EL3x4x: 0-20 mA, EL3x5x: 4-20 mA; KL and related product groups exactly the same	EL3x1x: 0-20 mA, EL3x2x: 4-20 mA; KL and related product groups exactly the same
Preferred current direction because of internal diode	Preferred current direction because of internal diode
Designed for the connection of externally-supplied sensors with a 3/4-wire connection	The terminal/ box is a passive differential current measuring device; passive means that the sensor is not supplied with power.
Designed for the connection of self-supplied sensors with a 2-wire connection	

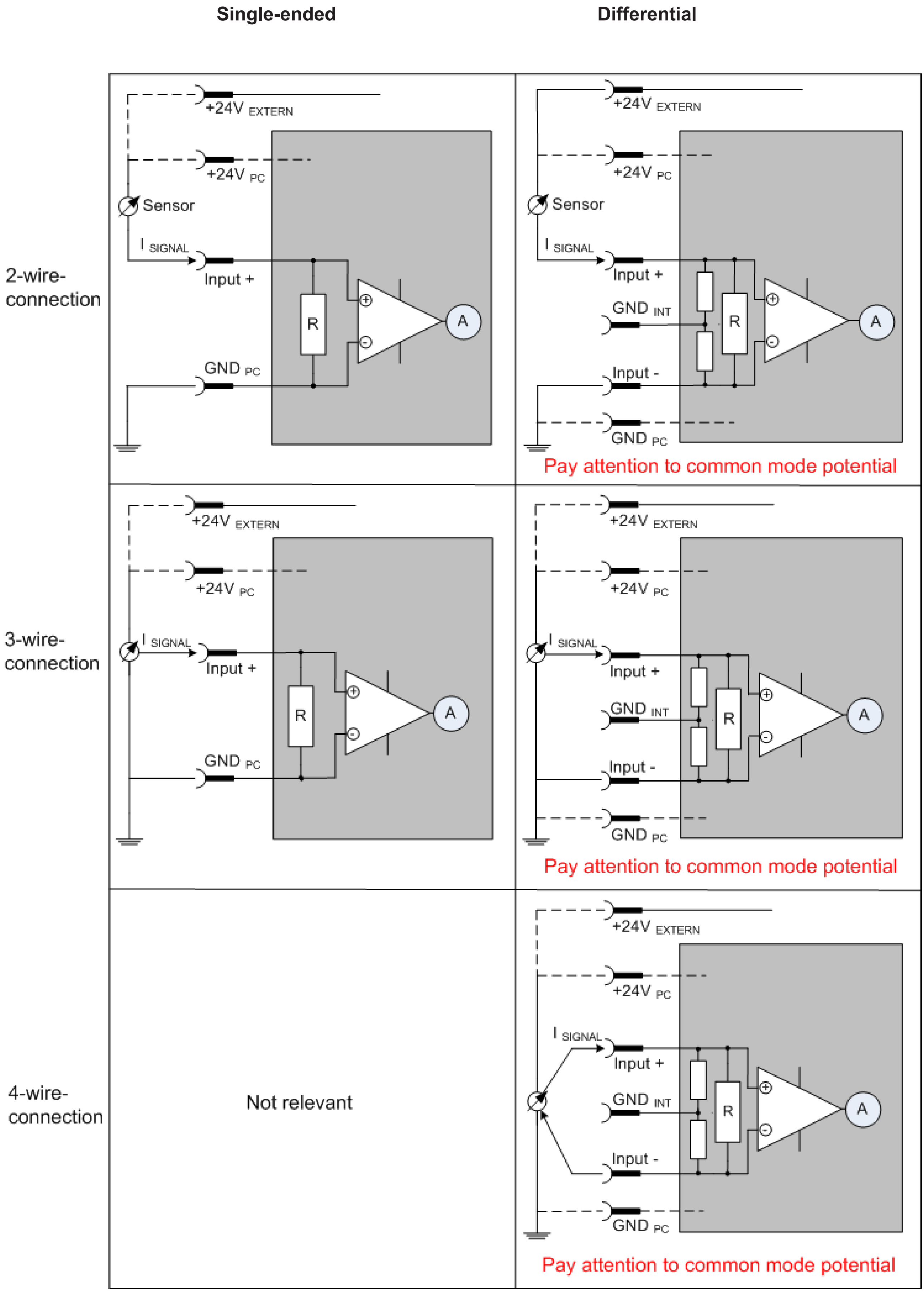


Fig. 77: 2-, 3- and 4-wire connection at single-ended and differential inputs

4.5.5 Common-mode voltage and reference ground (based on differential inputs)

Common-mode voltage (V_{cm}) is defined as the average value of the voltages of the individual connections/ inputs and is measured/specified against reference ground.

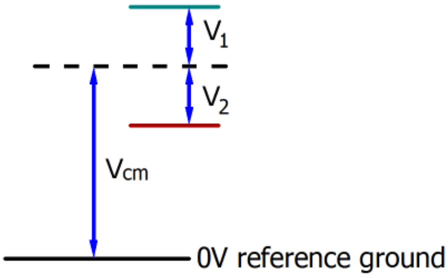


Fig. 78: Common-mode voltage (V_{cm})

The definition of the reference ground is important for the definition of the permitted common-mode voltage range and for measurement of the common-mode rejection ratio (CMRR) for differential inputs.

The reference ground is also the potential against which the input resistance and the input impedance for single-ended inputs or the common-mode resistance and the common-mode impedance for differential inputs is measured.

The reference ground is usually accessible at or near the terminal/ box, e.g. at the terminal contacts, power contacts (cable) or a mounting rail. Please refer to the documentation regarding positioning. The reference ground should be specified for the device under consideration.

For multi-channel terminals/ boxes with resistive (=direct, ohmic, galvanic) or capacitive connection between the channels, the reference ground should preferably be the symmetry point of all channels, taking into account the connection resistances.

Reference ground samples for Beckhoff IO devices:

1. Internal AGND fed out: EL3102/EL3112, resistive connection between the channels
2. 0V power contact: EL3104/EL3114, resistive connection between the channels and AGND; AGND connected to 0V power contact with low-resistance
3. Earth or SGND (shield GND):
    - EL3174-0002: Channels have no resistive connection between each other, although they are capacitively coupled to SGND via leakage capacitors
    - EL3314: No internal ground fed out to the terminal points, although capacitive coupling to SGND

4.5.6 Dielectric strength

A distinction should be made between:

- Dielectric strength (destruction limit): Exceedance can result in irreversible changes to the electronics
    - Against a specified reference ground
    - Differential
- Recommended operating voltage range: If the range is exceeded, it can no longer be assumed that the system operates as specified
    - Against a specified reference ground
    - Differential

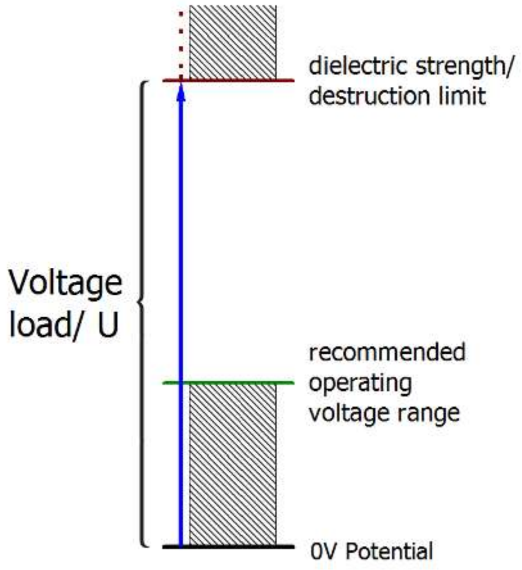


Fig. 79: Recommended operating voltage range

The device documentation may contain particular specifications and timings, taking into account:

- Self-heating
- Rated voltage
- Insulating strength
- Edge steepness of the applied voltage or holding periods
- Normative environment (e.g. PELV)

4.5.7 Temporal aspects of analog/digital conversion

The conversion of the constant electrical input signal to a value-discrete digital and machine-readable form takes place in the analog Beckhoff EL/KL/EP input modules with ADC (analog digital converter). Although different ADC technologies are in use, from a user perspective they all have a common characteristic: after the conversion a certain digital value is available in the controller for further processing. This digital value, the so-called analog process data, has a fixed temporal relationship with the “original parameter”, i.e. the electrical input value. Therefore, corresponding temporal characteristic data can be determined and specified for Beckhoff analogue input devices.

This process involves several functional components, which act more or less strongly in every AI (analog input) module:

- the electrical input circuit
- the analog/digital conversion
- the digital further processing
- the final provision of the process and diagnostic data for collection at the fieldbus (EtherCAT, K-bus, etc.)

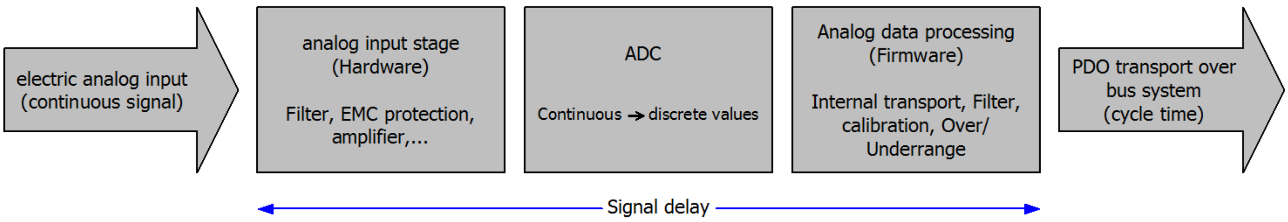


Fig. 80: Signal processing analog input

Two aspects are crucial from a user perspective:

- “How often do I receive new values?”, i.e. a sampling rate in terms of speed with regard to the device/ channel
- What delay does the (whole) AD conversion of the device/channel cause?
  - i.e. the hardware and firmware components in its entirety. For technological reasons, the signal characteristics must be taken into account when determining this information: the run times through the system differ, depending on the signal frequency.

This is the “external” view of the “Beckhoff AI channel” system – internally the signal delay in particular is composed of different components: hardware, amplifier, conversion itself, data transport and processing. Internally a higher sampling rate may be used (e.g. in the deltaSigma converters) than is offered “externally” from the user perspective. From a user perspective of the “Beckhoff AI channel” component this is usually irrelevant or is specified accordingly, if it is relevant for the function.

For Beckhoff AI devices the following specification parameters for the AI channel are available for the user from a temporal perspective:

1. Minimum conversion time [ms, µs]
= the reciprocal value of the maximum **sampling rate** [sps, samples per second]:
Indicates how often the analog channel makes a newly detected process data value available for collection by the fieldbus. Whether the fieldbus (EtherCAT, K-bus) fetches the value with the same speed (i.e. synchronous), or more quickly (if the AI channel operates in slow FreeRun mode) or more slowly (e.g. with oversampling), is then a question of the fieldbus setting and which modes the AI device supports.
For EtherCAT devices the so-called toggle bit indicates (by toggling) for the diagnostic PDOs when a newly determined analog value is available.
Accordingly, a maximum conversion time, i.e. a smallest sampling rate supported by the AI device, can be specified.
Corresponds to IEC 61131-2, section 7.10.2 2, “Sampling repeat time”

2. Typical signal delay
Corresponds to IEC 61131-2, section 7.10.2 1, “Sampling duration”. From this perspective it includes all internal hardware and firmware components, but not “external” delay components from the fieldbus or the controller (TwinCAT).
This delay is particularly relevant for absolute time considerations, if AI channels also provide a time stamp that corresponds to the amplitude value – which can be assumed to match the physically prevailing amplitude value at the time.
Due to the frequency-dependent signal delay time, a dedicated value can only be specified for a given signal. The value also depends on potentially variable filter settings of the channel.
A typical characterization in the device documentation may be:

2.1 Signal delay (step response)
Keywords: Settling time
The square wave signal can be generated externally with a frequency generator (note impedance!)
The 90 % limit is used as detection threshold.
The signal delay [ms, µs] is then the time interval between the (ideal) electrical square wave signal and the time at which the analog process value has reached the 90 % amplitude.

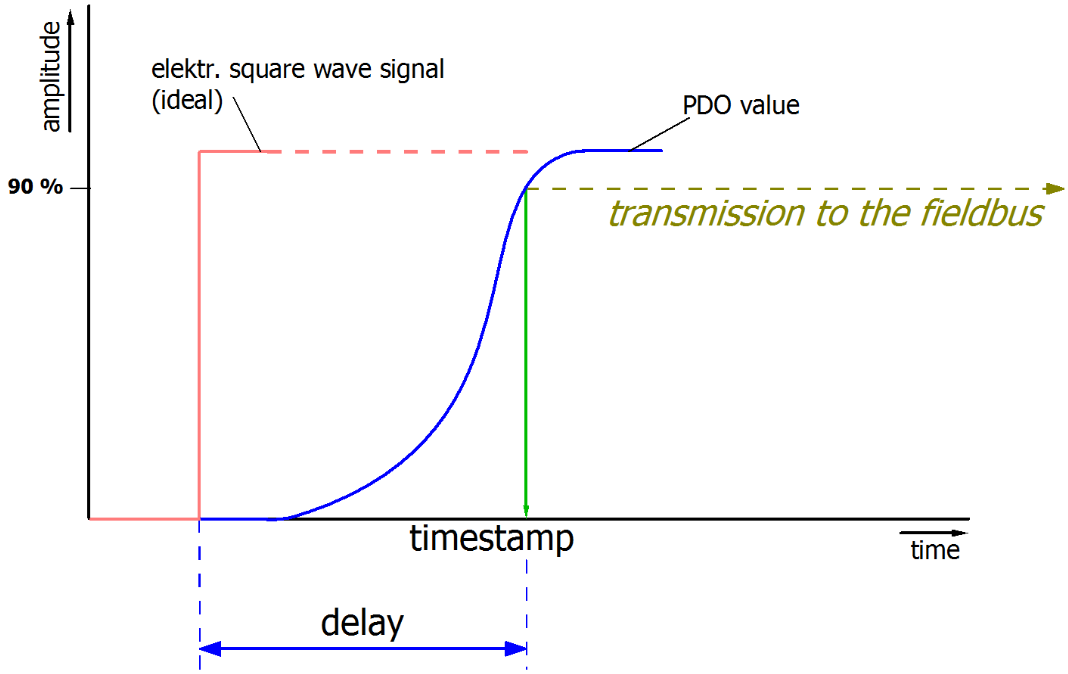


Fig. 81: Diagram signal delay (step response)

2.2 Signal delay (linear)
Keyword: Group delay
Describes the delay of a signal with constant frequency
A test signal can be generated externally with a frequency generator, e.g. as sawtooth or sine. A simultaneous square wave signal would be used as reference.
The signal delay [ms, µs] is then the interval between the applied electrical signal with a particular amplitude and the moment at which the analog process value reaches the same value.
A meaningful range must be selected for the test frequency, e.g. 1/20 of the maximum sampling rate.

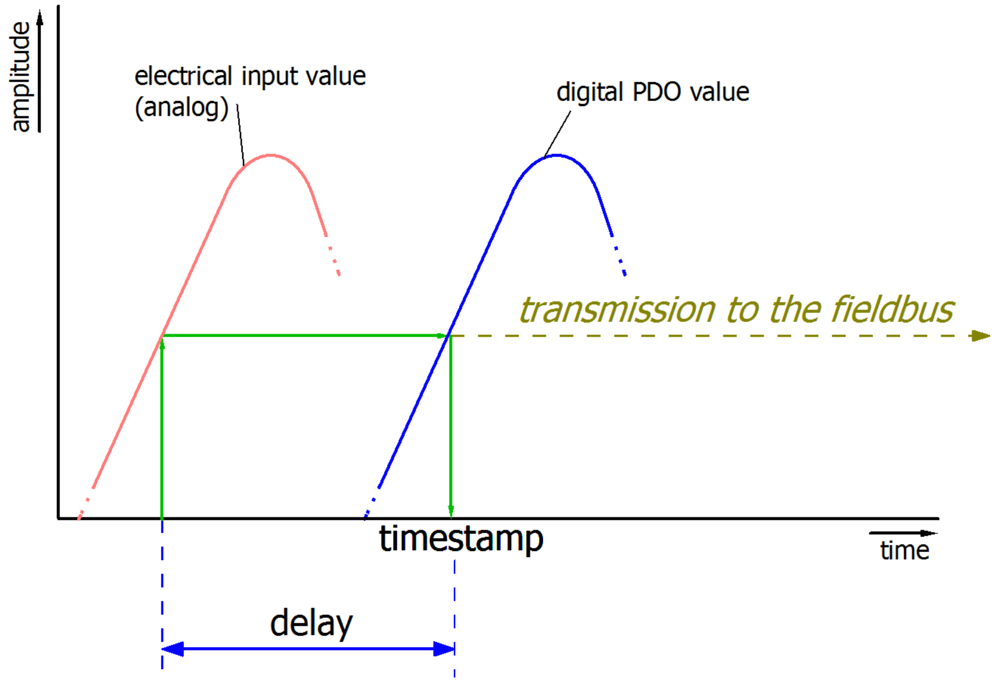


Fig. 82: Diagram signal delay (linear)

3. Additional information:
may be provided in the specification, e.g.
3.1 Actual sampling rate of the ADC (if different from the channel sampling rate)
3.2 Time correction values for run times with different filter settings
…

4.6 Application example

The application example described below for the EtherCAT Terminals can also be used for the EP3632 and EPP3632 EtherCAT Boxes.

Use of ScopeView2 with oversampling

Preparation

- Use TwinCAT 2.11, build 1549 or higher
- Use current ScopeView2

Activation of oversampling

The following steps must be performed (fig. Activation of ScopeView2):

1. Select the process image of the EtherCAT master (in this case: “Device 1 - process image”)
2. Select “Enable ADS Server” in the “ADS” tab
3. Select “Create symbols”
4. Note the “Port” (usually 27905)
5. Adopt configuration (note: PLC restart is necessary!)

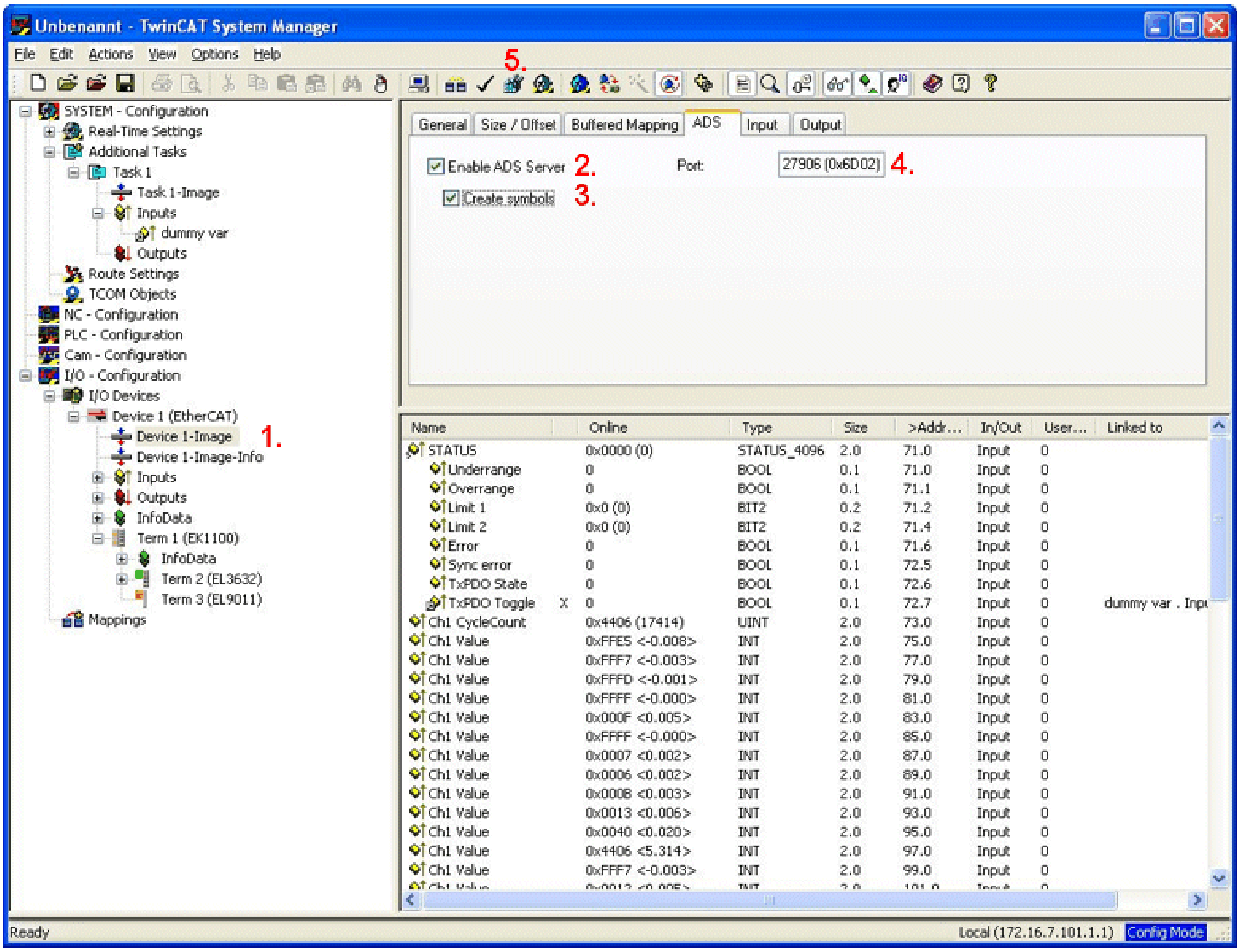


Fig. 83: Activation of ScopeView2

Use of the oversampling variable

The following steps must be performed (fig. *Target Browser*):

- Select ScopeView2 “View” - > “Target Browser”
- Select “Enable Server Ports” (button at top left)
- Enter the noted port (see fig. *Activation of ScopeView2*) in the field at the bottom left
- Confirm “Add” and “OK”
- A new entry, “AdsPort of Image 1”, appears in the Target Browser; select it and display the tree with [+]
- Select the variable “CH1 SAMPLE 0” or “CH2 SAMPLE 0” for the terminal used and display the tree with [+]

The last entry in the tree contains the oversampling variable (can be recognized by the index T +Oversampling factor)
A double-click on this variable displays its characteristics. A double-click on the variable with the blue box inserts it into the object browser.

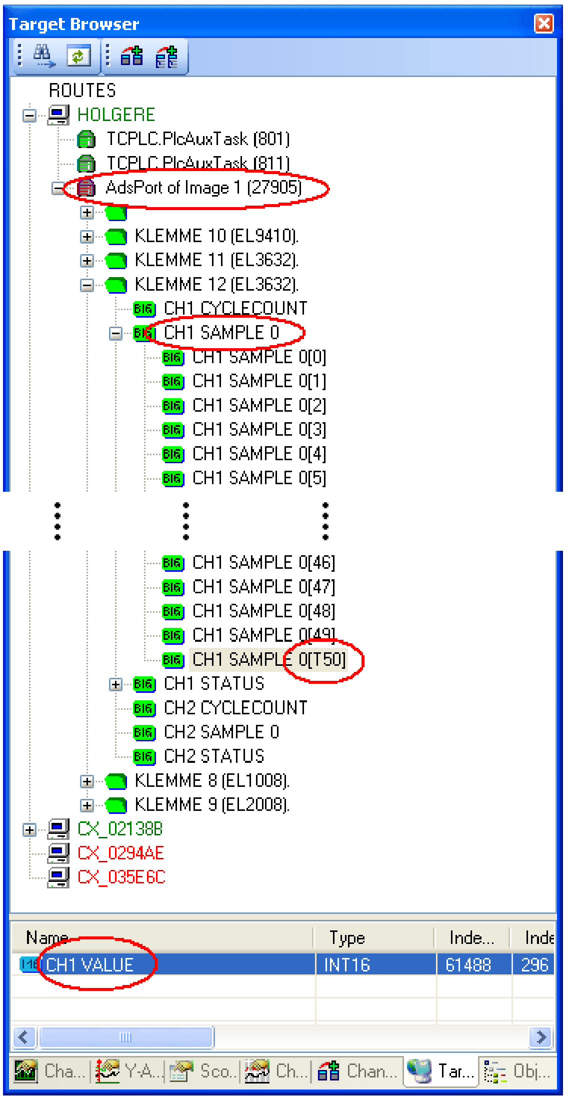


Fig. 84: Target Browser

Creating a task

An additional task must be used if no PLC is active: TwinCAT System Manager: “SYSTEM - Configuration” - > additional task - > rights mouse button “Add task” - > rights mouse button “Insert variable”. Create variable(s) and link with a terminal. The meaning and size of the variables are not relevant.

4.7 Error descriptions and troubleshooting

Error Codes

Error Index 0x6000:07	Underange Index 0x6000:01	Overrange Index 0x6000:02	TxPDO State Index 0x6000:0F	Sync Error Index 0x6000:0E	Error description	Remedy
1	1				Measurement is below range	Reduce the input level, change the gain (filter settings)
1		1			Measuring range exceeded	Reduce the input level, change the gain (filter settings)
1					General measuring error	e.g. open circuit
				1	Synchronization error	Jitter of master too high, distributed clocks switched off

The error LED lights up only in case of open circuit.

Troubleshooting

The dialog for setting the sampling rate is missing

The dialog for setting the sampling rate is missing. The TwinCAT System Manager uses the “online description” of the terminal. The note to use the dialog for the parameterization is only available in the XML description.
Solution: Make sure that the latest XML description is used.

The amplitude is too small

Filter settings incorrect.
Solution: Correct the settings in the CoE in the entries 0x80n0:15.

The terminal switches to SAFEOP

The terminal switches to SAFEOP. The real-time-settings are not accurate enough.
Solution: Use a PC without mobile chipset/CPU.

4.8 Restoring the delivery state

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

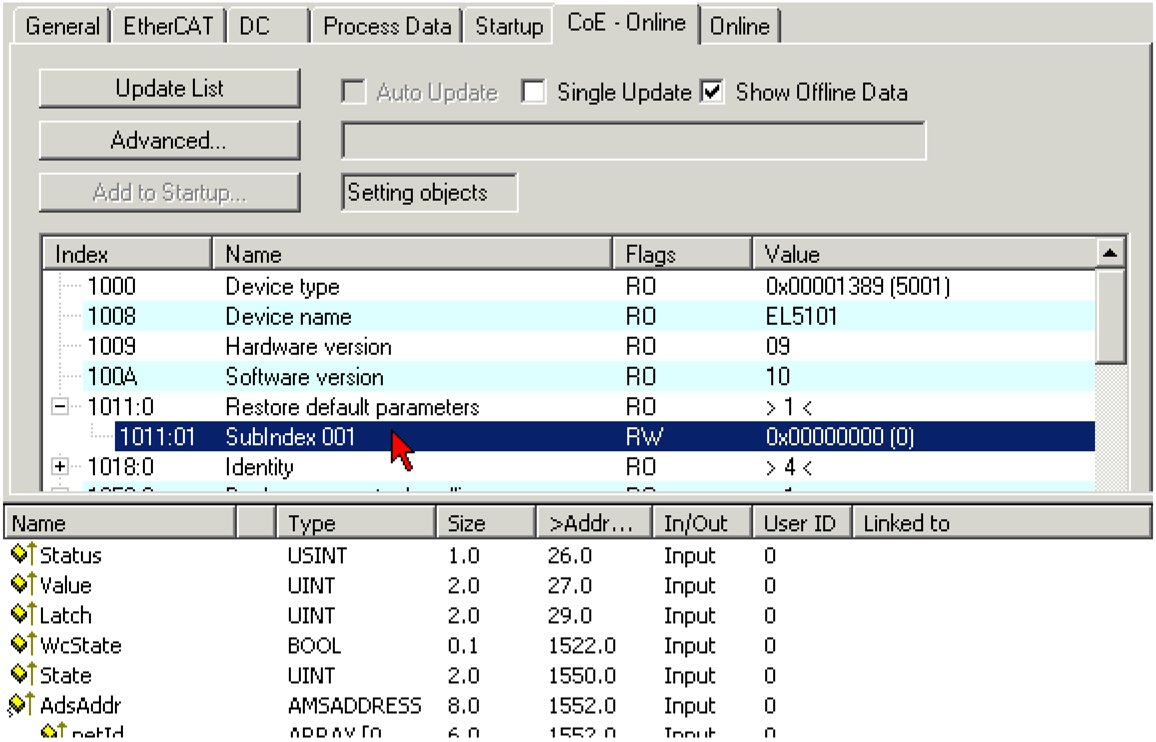


Fig. 85: Selecting the Restore default parameters PDO

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.

All backup objects are reset to the delivery state.

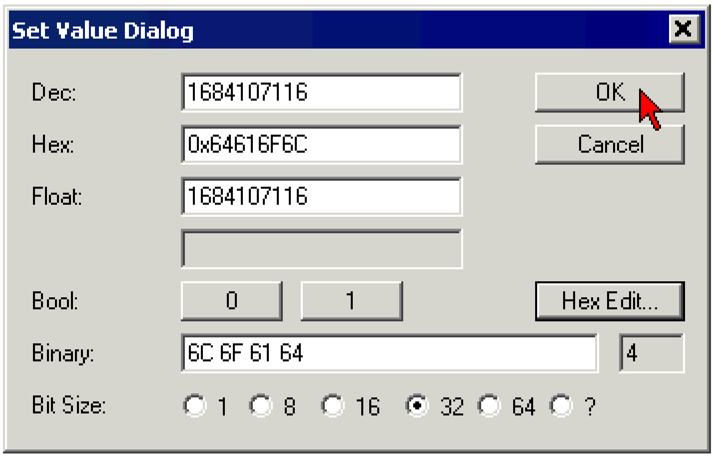


Fig. 86: Entering a restore value in the Set Value dialog

Alternative restore value

In some older terminals / boxes the backup objects can be switched with an alternative restore value:
Decimal value: 1819238756
Hexadecimal value: 0x6C6F6164

An incorrect entry for the restore value has no effect.

4.9 Firmware Update EL/ES/EM/EPxxxx

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

Storage locations

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

- Depending on functionality and performance EtherCAT slaves have one or several local controllers for processing I/O data. The corresponding program is the so-called **firmware** in *.efw format.
- In some EtherCAT slaves the EtherCAT communication may also be integrated in these controllers. In this case the controller is usually a so-called **FPGA** chip with *.rbf firmware.
- In addition, each EtherCAT slave has a memory chip, a so-called **ESI-EEPROM**, for storing its own device description (ESI: EtherCAT Slave Information). On power-up this description is loaded and the EtherCAT communication is set up accordingly. The device description is available from the download area of the Beckhoff website at (http://www.beckhoff.de). All ESI files are accessible there as zip files.

Customers can access the data via the EtherCAT fieldbus and its communication mechanisms. Acyclic mailbox communication or register access to the ESC is used for updating or reading of these data.

The TwinCAT System Manager offers mechanisms for programming all 3 parts with new data, if the slave is set up for this purpose. Generally the slave does not check whether the new data are suitable, i.e. it may no longer be able to operate if the data are unsuitable.

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 re-quire re-commissioning by the manufacturer.

Device description ESI file/XML

NOTE

Notice regarding update of the ESI description/EEPROM

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

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

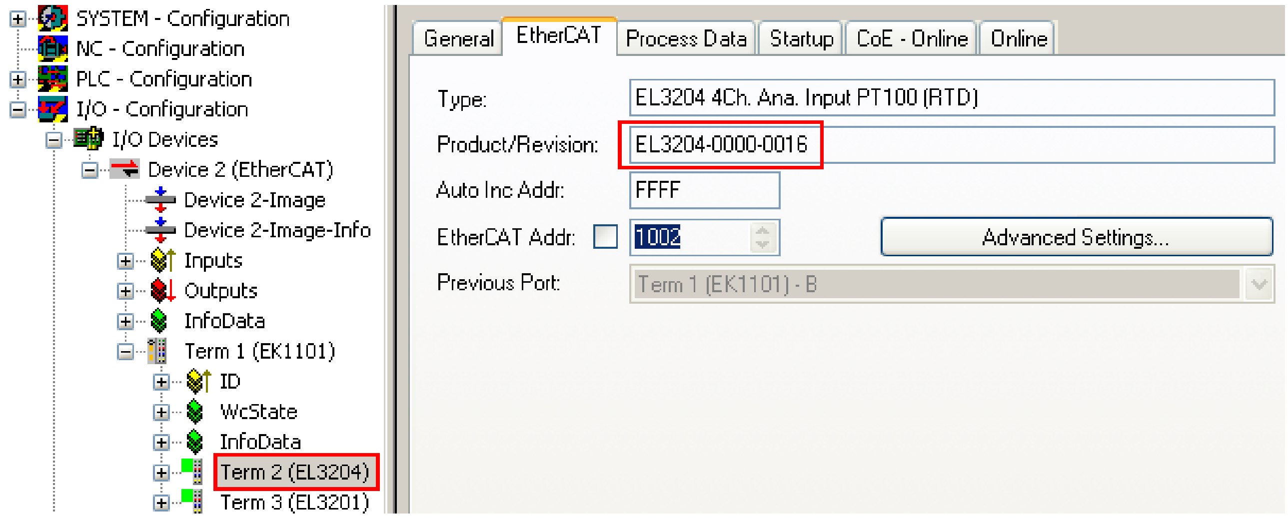


Fig. 87: Device identifier consisting of name EL3204-0000 and revision -0016

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

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

Update of XML/ESI description

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

Display of ESI slave identifier

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

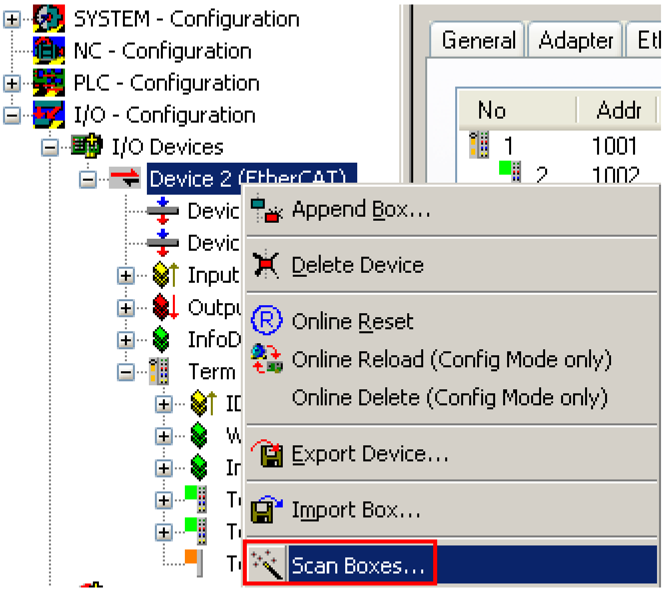


Fig. 88: Scan Boxes

Scan the subordinate field by right-clicking on the EtherCAT device in Config/FreeRun mode

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

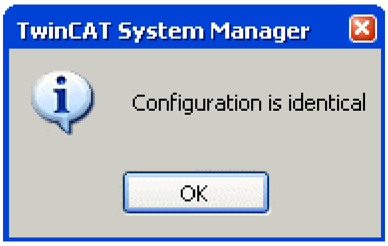


Fig. 89: Configuration is identical

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

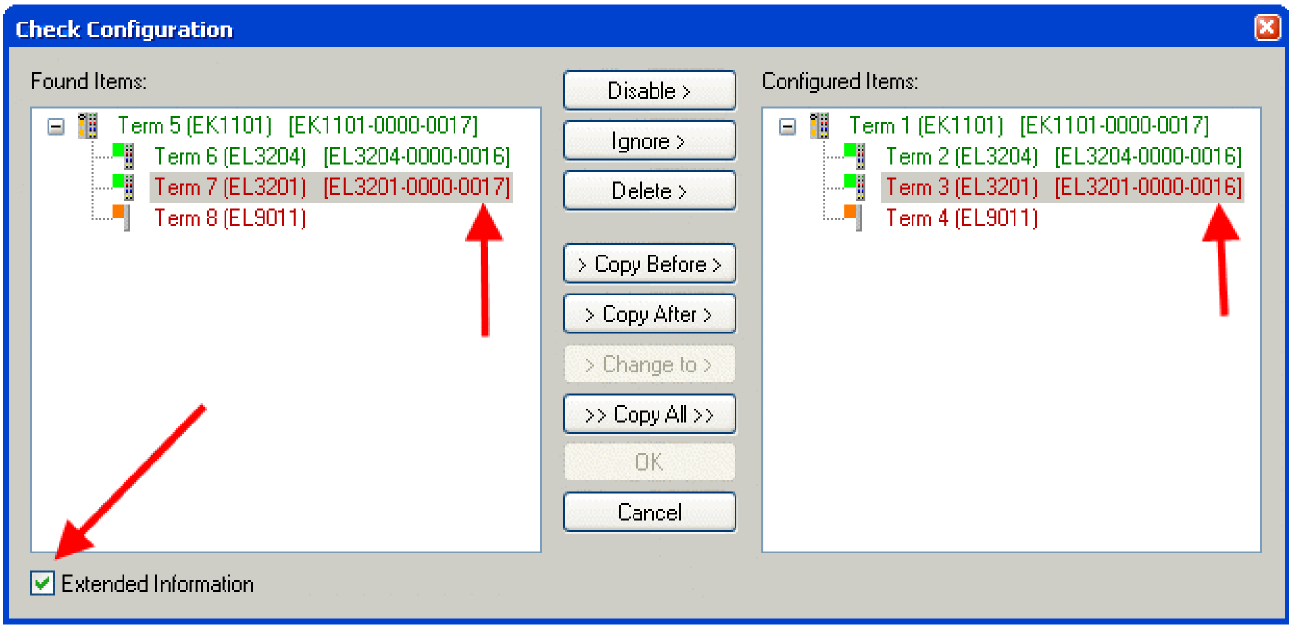


Fig. 90: Change dialog

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*

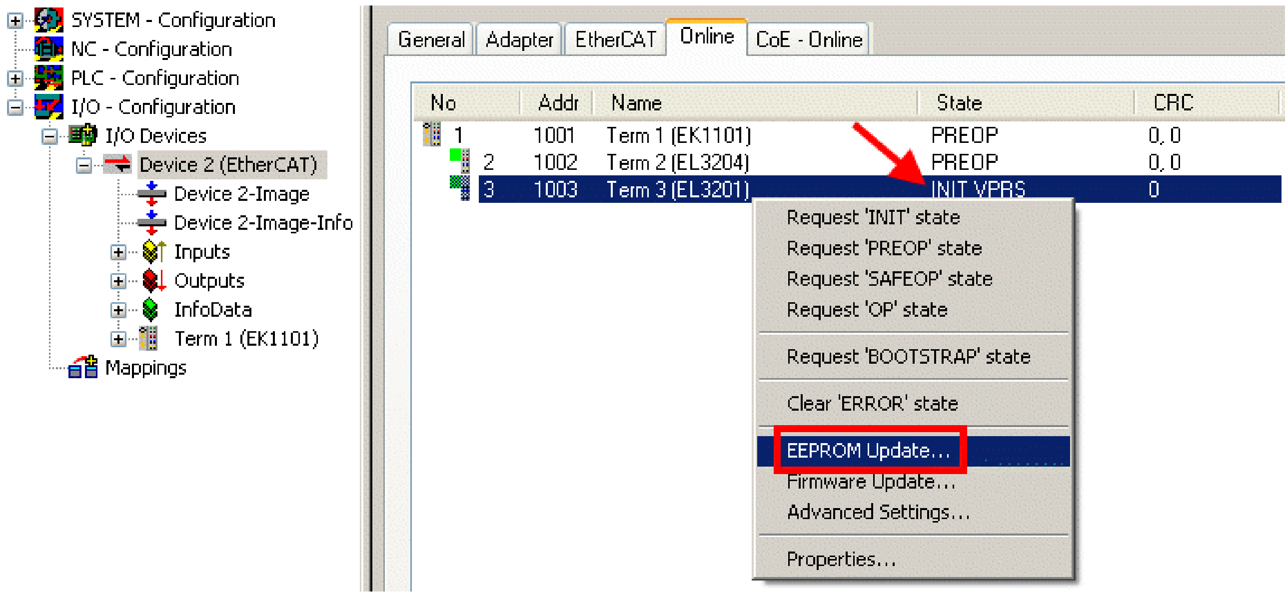


Fig. 91: EEPROM Update

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

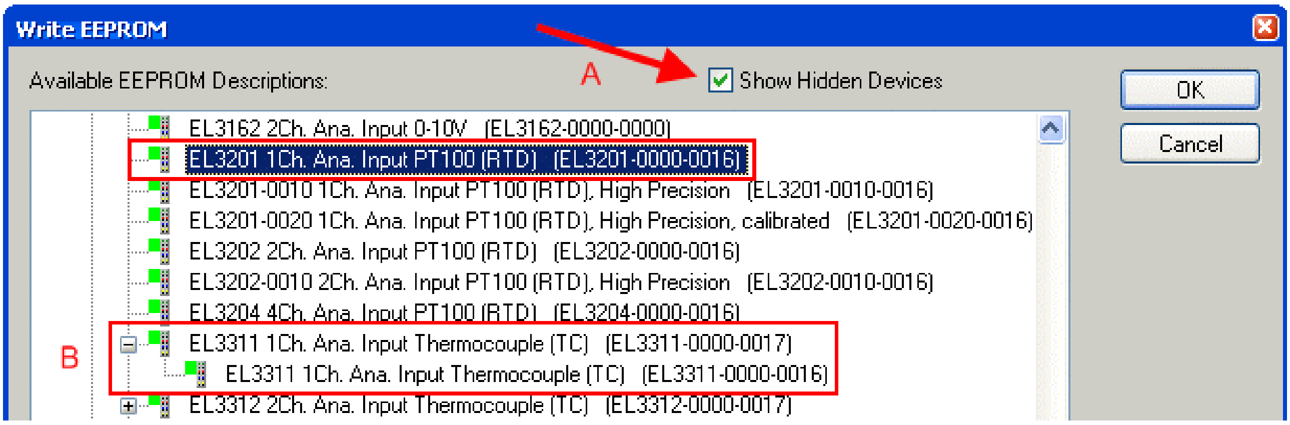


Fig. 92: Selecting the new ESI

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

The change only takes effect after a restart.

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

Determining the firmware version

Determining the version on laser inscription

Beckhoff EtherCAT Box feature batch numbers (D number) applied by laser. The D-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 D-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 does supported it. 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.

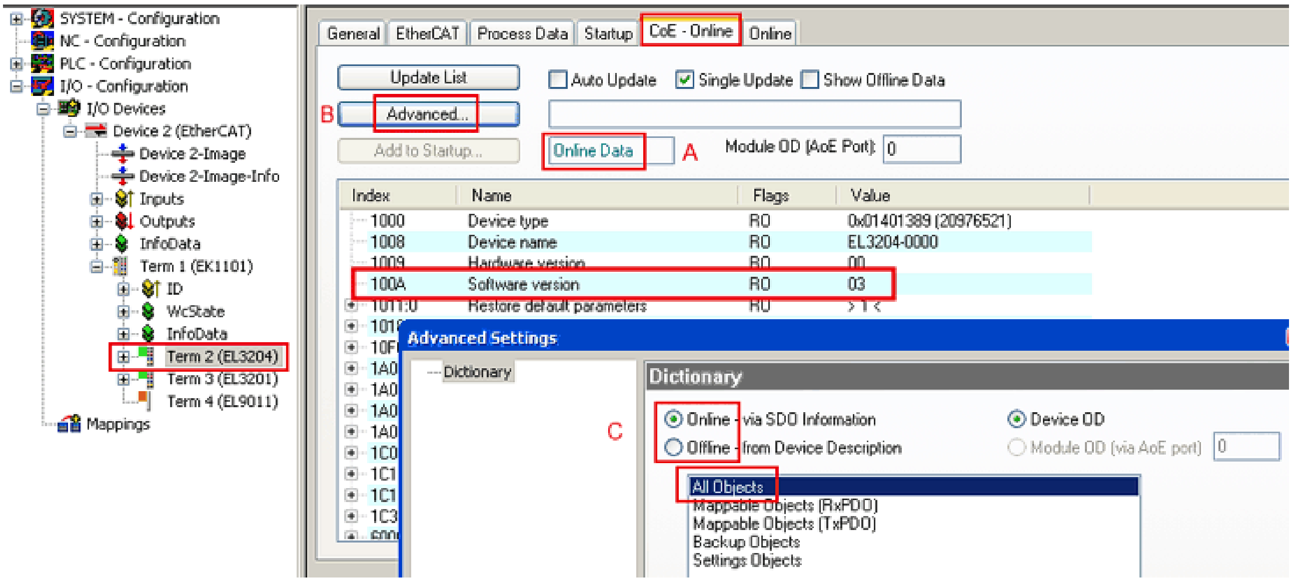


Fig. 93: Display of EL3204 firmware version

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*.

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.*

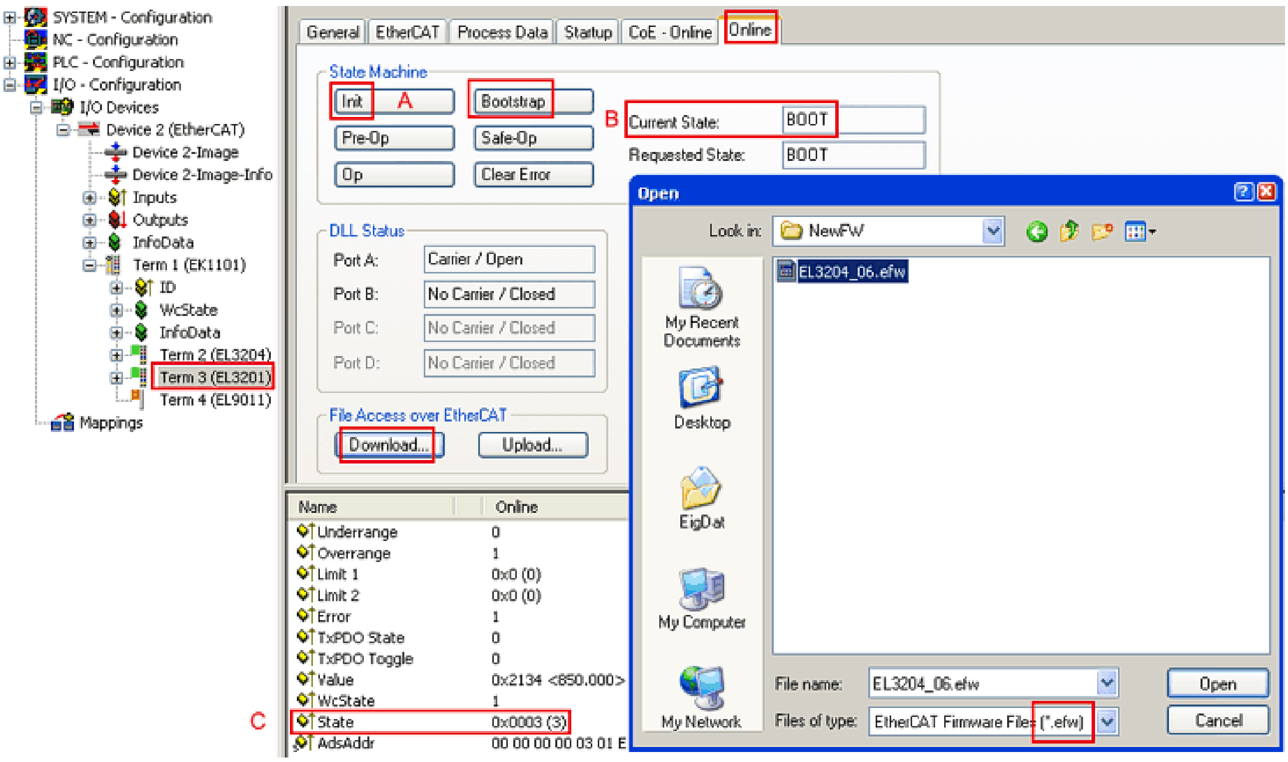


Fig. 94: Firmware Update

Proceed as follows, unless instructed otherwise by Beckhoff support.

- Switch slave to INIT (A)

- Switch slave to BOOTSTRAP
- Check the current status (B, C)
- Download the new *efw file
- After the download switch to INIT, then OP
- Switch off the slave briefly

FPGA firmware *.rbf

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

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

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

Determining the version via the System Manager

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

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

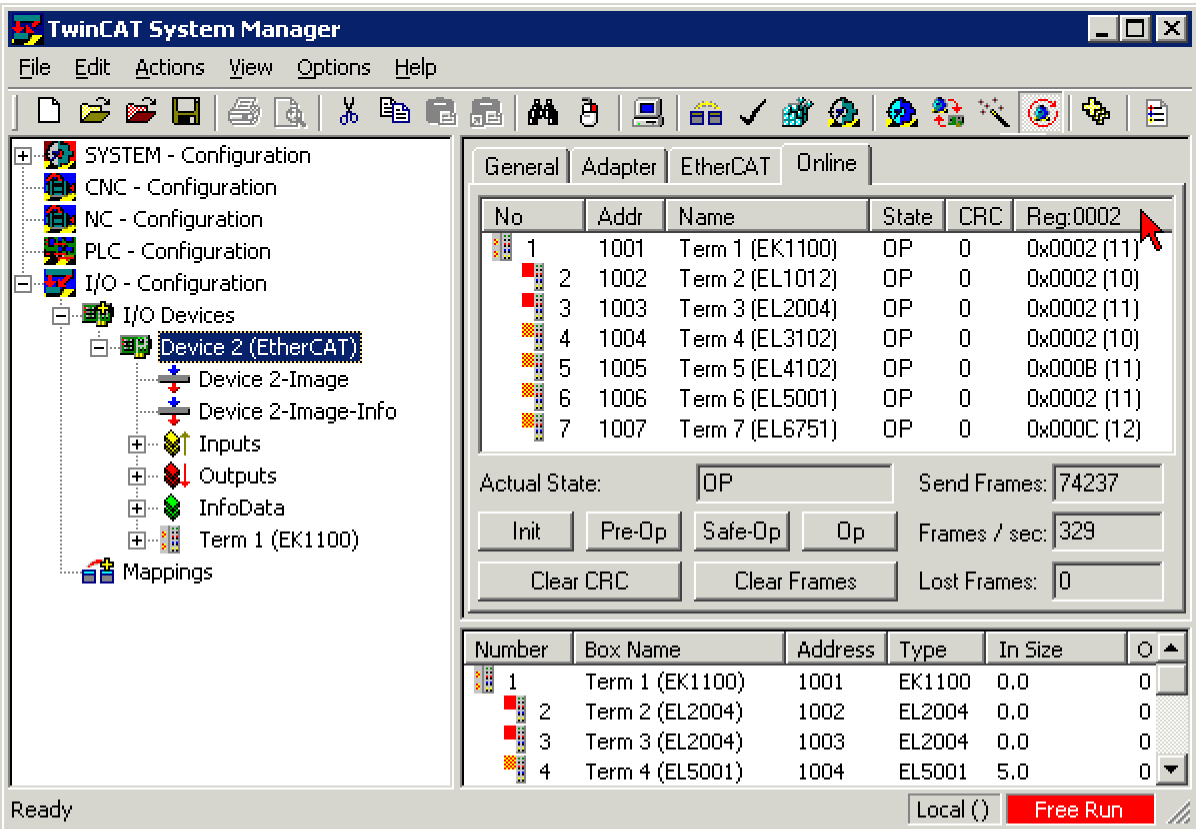


Fig. 95: FPGA firmware version definition

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

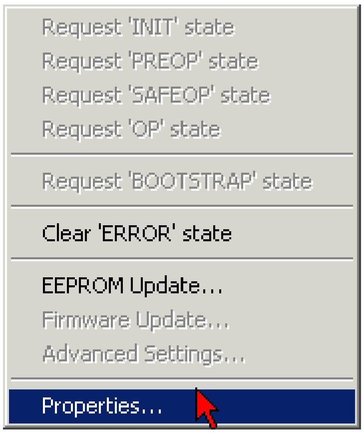


Fig. 96: Context menu *Properties*

The *Advanced Settings* dialog appears where the columns to be displayed can be selected. Under *Diagnosis/***Online View** select the *'0002 ETxxxx Build'* check box in order to activate the FPGA firmware version display.

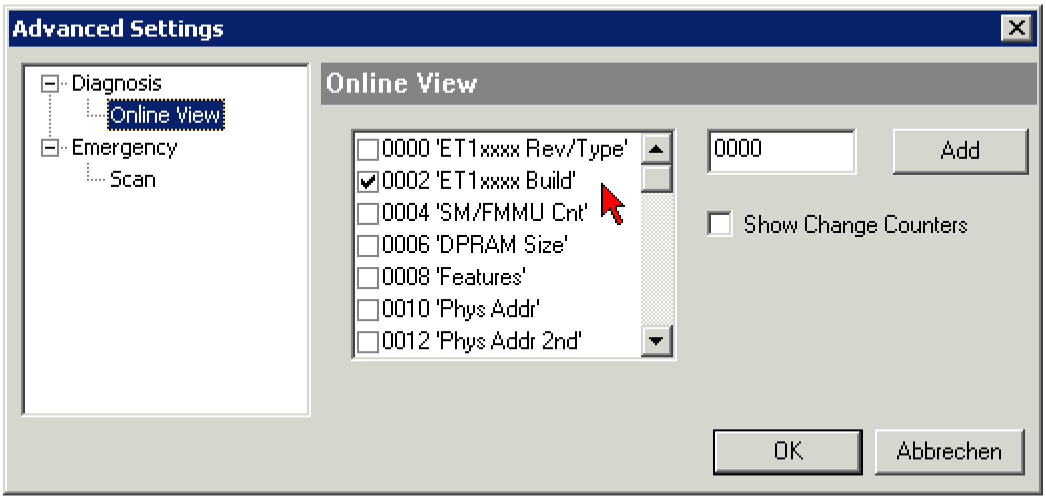


Fig. 97: Dialog *Advanced Settings*

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

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

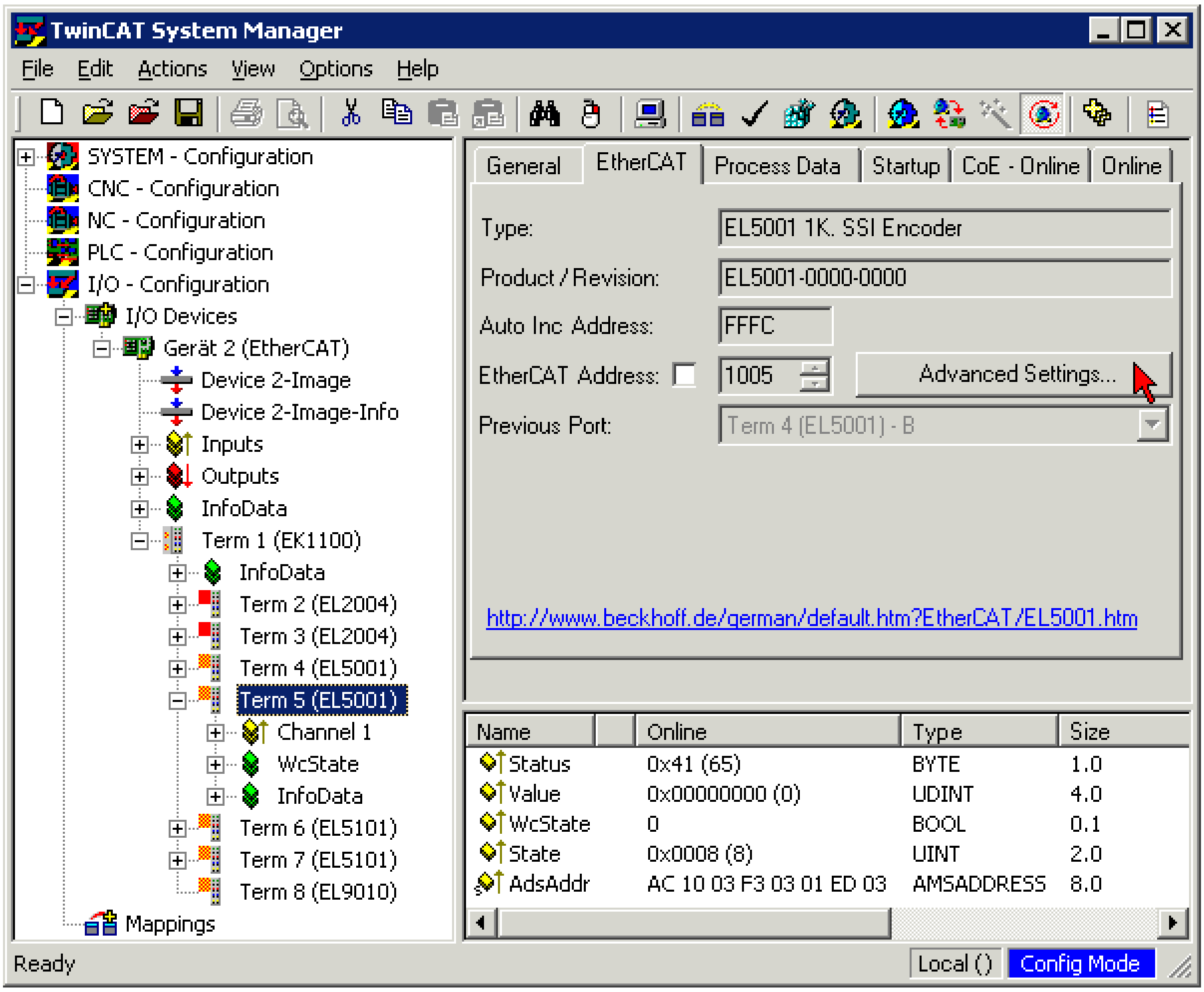


Fig. 98: Select dialog *Advanced Settings*

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

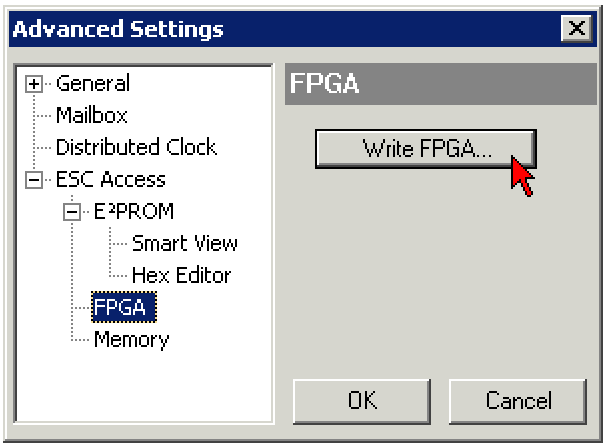


Fig. 99: Select dialog *Write FPGA*

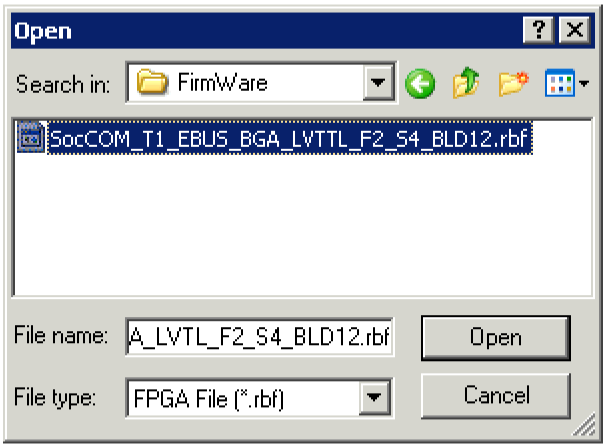


Fig. 100: Select file

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

NOTE

Risk of damage to the device!

A firmware download to an EtherCAT device must never be interrupted! If this process is cancelled, the supply voltage switched off or the Ethernet connection interrupted, the EtherCAT device can only be recommissioned by the manufacturer!

In order to activate the new FPGA firmware a restart (switching the power supply off and on again) of the EtherCAT device is required.

Simultaneous updating of several EtherCAT devices

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

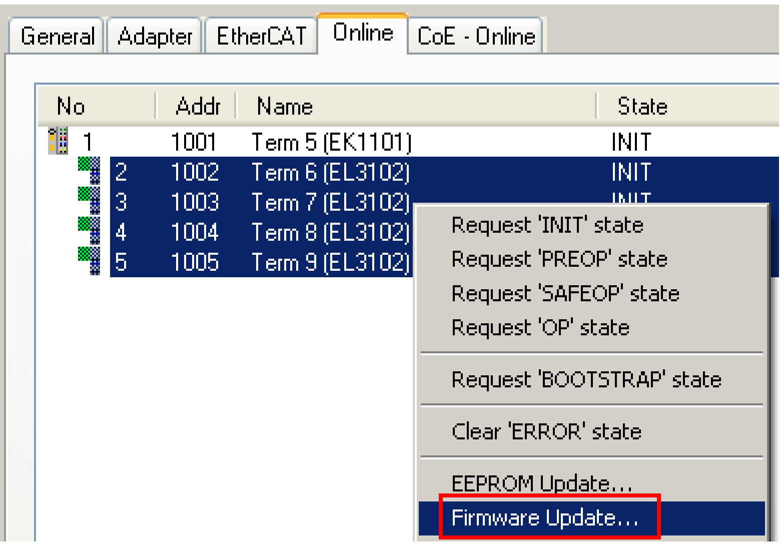


Fig. 101: Multiple selection and firmware update

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

5 Appendix

5.1 General operating conditions

Protection degrees (IP-Code)

The standard IEC 60529 (DIN EN 60529) defines the degrees of protection in different classes.

1. Number: dust protection and touch guard	Definition
0	Non-protected
1	Protected against access to hazardous parts with the back of a hand. Protected against solid foreign objects of Ø 50 mm
2	Protected against access to hazardous parts with a finger. Protected against solid foreign objects of Ø 12.5 mm.
3	Protected against access to hazardous parts with a tool. Protected against solid foreign objects Ø 2.5 mm.
4	Protected against access to hazardous parts with a wire. Protected against solid foreign objects Ø 1 mm.
5	Protected against access to hazardous parts with a wire. Dust-protected. Intrusion of dust is not totally prevented, but dust shall not penetrate in a quantity to interfere with satisfactory operation of the device or to impair safety.
6	Protected against access to hazardous parts with a wire. Dust-tight. No intrusion of dust.

2. Number: water* protection	Definition
0	Non-protected
1	Protected against water drops
2	Protected against water drops when enclosure tilted up to 15°.
3	Protected against spraying water. Water sprayed at an angle up to 60° on either side of the vertical shall have no harmful effects.
4	Protected against splashing water. Water splashed against the disclosure from any direction shall have no harmful effects
5	Protected against water jets
6	Protected against powerful water jets
7	Protected against the effects of temporary immersion in water. Intrusion of water in quantities causing harmful effects shall not be possible when the enclosure is temporarily immersed in water for 30 min. in 1 m depth.

*) These protection classes define only protection against water!

Chemical Resistance

The Resistance relates to the Housing of the Fieldbus/EtherCAT Box and the used metal parts. In the table below you will find some typical resistance.

Character	Resistance
Steam	at temperatures >100°C: not resistant
Sodium base liquor (ph-Value > 12)	at room temperature: resistant > 40°C: not resistant
Acetic acid	not resistant
Argon (technical clean)	resistant

Key

- resistant: Lifetime several months
- non inherently resistant: Lifetime several weeks
- not resistant: Lifetime several hours resp. early decomposition

5.2 EtherCAT Box- / EtherCAT P Box - Accessories

Fixing

Ordering information	Description
ZS5300-0001	Mounting rail (500 mm x 129 mm)

Marking material, plugs

Ordering information	Description
ZS5000-0000	Fieldbus Box set M8 (contact labels, plugs)
ZS5000-0002	Fieldbus Box set M12 (contact labels, plugs)
ZS5000-0010	plugs M8, IP67 (50 pieces)
ZS5000-0020	plugs M12, IP67 (50 pieces)
ZS5100-0000	marking labels, not printed, 4 stripes at 10 pieces
ZS5100-xxxx	printed marking labels, on request

Tools

Ordering information	Description
ZB8800	torque wrench for M8 cables with knurl, incl. ratchet
ZB8800-0001	M12 ratchet for torque wrench ZB8800
ZB8800-0002	M8 ratchet (field assembly) for torque wrench ZB8800
ZB8801-0000	torque wrench for hexagonal plugs, adjustable
ZB8801-0001	torque cable key, M8/wrench size 9, for torque wrench ZB8801-0000
ZB8801-0002	torque cable key, M12/wrench size 13, for torque wrench ZB8801-0000
ZB8801-0003	torque cable key, M12 field assembly/wrench size 13, for torque wrench ZB8801-0000

Further accessories

Further accessories may be found at the price list for Beckhoff fieldbus components and at the internet under https://www.beckhoff.com

5.3 Support and Service

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

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages:
http://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49(0)5246/963-0
Fax:	+49(0)5246/963-198
e-mail:	info@beckhoff.com

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49(0)5246/963-157
Fax:	+49(0)5246/963-9157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49(0)5246/963-460
Fax:	+49(0)5246/963-479
e-mail:	service@beckhoff.com

Table of figures