BECKHOFF New Automation Technology

Documentation | EN

EPP6228-0022

IO-Link master

2023-02-01 | Version: 1.4

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®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® 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, EP1456722, EP2137893, DE102015105702 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
1.4	• EtherCAT P status LEDs updated • Chapter "PLC library: Tc3_IoLink" added
1.3	• Product image updated • Technical data updated • Chapter "Commissioning and configuration" updated • Structure update
1.2	• Dimensions updated • UL requirements updated
1.1	• Front page updated
1.0	• First release
0.1	• First preliminary version

Firmware and hardware versions

This documentation refers to the firmware and hardware version that was applicable at the time the documentation was written.

The module features are continuously improved and developed further. Modules having earlier production statuses cannot have the same properties as modules with the latest status. However, existing properties are retained and are not changed, so that older modules can always be replaced with new ones.

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

Syntax of the batch number (D-number)

D: WW YY FF HH	Example with D no. 29 10 02 01:
WW - week of production (calendar week) YY - year of production FF - firmware version HH - hardware version	29 - week of production 29 10 - year of production 2010 02 - firmware version 02 01 - hardware version 01

Further information on this topic: Version identification of EtherCAT devices [▶ 73].

2 Product group: EtherCAT P Box modules

EtherCAT P

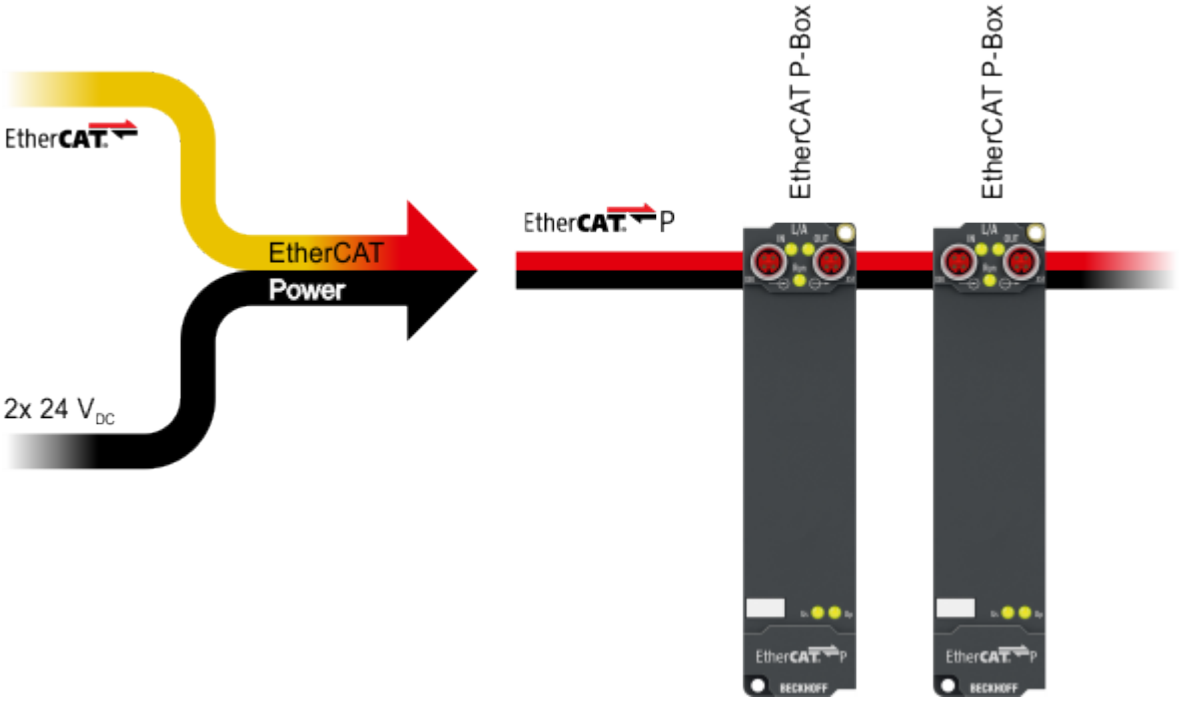
EtherCAT P supplements the EtherCAT technology with a process in which communication and supply voltages are transmitted on a common line. All EtherCAT properties are retained with this process.

Two supply voltages are transmitted per EtherCAT P line. The supply voltages are electrically isolated from each other and can therefore be switched individually. The nominal supply voltage for both is 24 V_{DC}.

EtherCAT P uses the same cable structure as EtherCAT: a 4-core Ethernet cable with M8 connectors. The connectors are mechanically coded so that EtherCAT connectors and EtherCAT P connectors cannot be interchanged.

EtherCAT P Box modules

EtherCAT P Box modules are EtherCAT P slaves with degree of protection IP67. They are designed for operation in wet, dirty or dusty industrial environments.

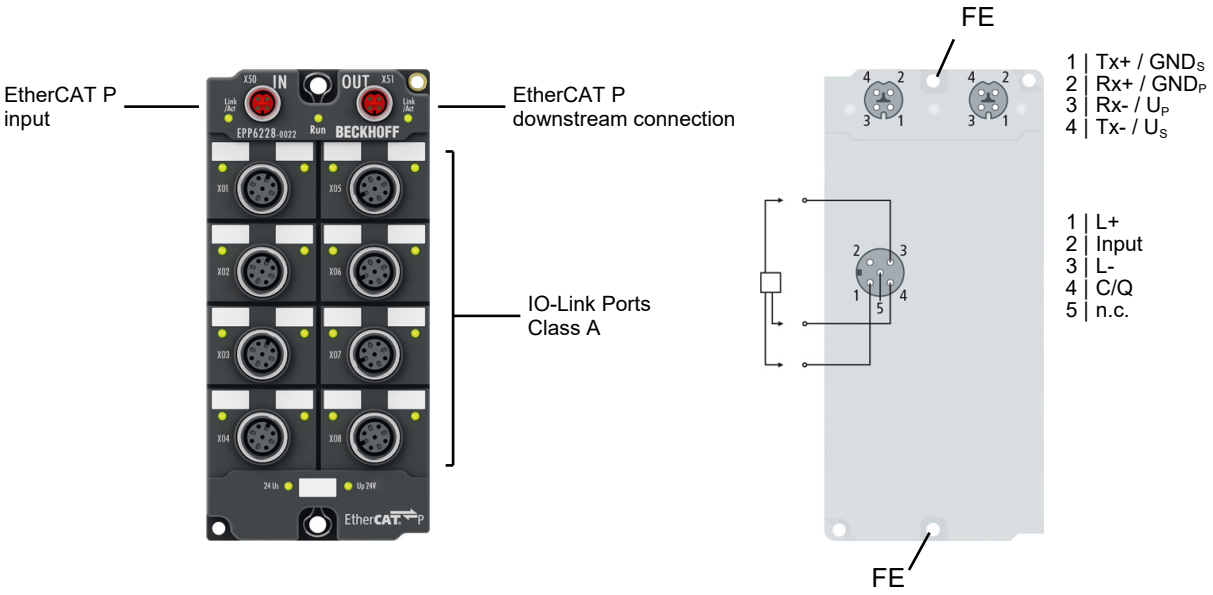


EtherCAT basics

A detailed description of the EtherCAT system can be found in the EtherCAT system documentation.

3 Product overview

3.1 Introduction



IO-Link-Master

The EPP6228 IO-Link module enables connection of up to eight IO-Link devices, e.g. IO-Link box modules, actuators, sensors or combinations thereof. A point-to-point connection is used between the module and the device. The EtherCAT P Box is parameterised via the EtherCAT master. IO-Link is designed as an intelligent link between the fieldbus level and the sensor, so that parameterisation information can be exchanged bidirectionally via the IO-Link connection. The parameterisation of the IO-Link devices with service data can be done from TwinCAT via ADS or very conveniently via the integrated IO-Link configuration tool. In addition to the IO-Link channels, the EPP6228 features eight digital inputs on pin 2 of the respective M12 socket. In the standard setting, the IO-Link channels C/Qx of the EPP6228 accept both IO-Link devices and standard sensors with 24 V DC.

Quick links

Technical data [▶ 10]
Process image [▶ 12]
IO-Link connection [▶ 24]

3.2 Technical data

All values are typical values over the entire temperature range, unless stated otherwise.

EtherCAT P	
Connection	2 x M8 socket, 4-pin, P-coded, red
Distributed Clocks	yes

Supply voltages	
Connection	See EtherCAT P connection
U_S nominal voltage	24 V_{DC} (-15 % / +20 %)
U_S sum current: $I_{S,sum}$	max. 3 A
Current consumption from U_S	100 mA + Current consumption of the connected IO-Link devices
Rated voltage U_P	24 V_{DC} (-15 % / +20 %)
U_P sum current: $I_{P,sum}$	max. 3 A
Current consumption from U_P	None. U_P is only forwarded.

IO-Link Ports	
Number	8x Class A
Connection	8x M12 socket
Cable length	max. 20 m
Specification	IO-Link V1.1
Data rate	COM1: 4.8 kbit/s COM2: 38.4 kbit/s COM3: 230.4 kbit/s
Device power supply	24 V_{DC} from the control voltage U_S max. 1.4 A per port, short-circuit proof max. 3.0 A in total
Digital input (pin 2): characteristic	Type 3 according to EN 61131-2, compatible with type 1
Digital input (pin 2): input filter	10 µs
Digital input (pin 2): input current	3.0 mA

Housing data	
Dimensions W x H x D	60 mm x 126 mm x 26.5 mm (without plug connectors)
Weight	approx. 250 g
Installation position	variable
Material	PA6 (polyamide)

Environmental conditions	
Ambient temperature during operation	-25 … +60 °C -25 … +55 °C according to cULus
Ambient temperature during storage	-40 … +85 °C
Vibration resistance, shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27 Additional checks [▶ 11]
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP65, IP66, IP67 (conforms to EN 60529)

Approvals / markings	
Approvals / markings *)	CE, cULus [▶ 26]

*) Real applicable approvals/markings see type plate on the side (product marking).

Additional tests

The devices have undergone the following additional tests:

Test	Explanation
Vibration	10 frequency sweeps in 3 axes
	5 Hz < f < 60 Hz displacement 0.35 mm, constant amplitude
	60.1 Hz < f < 500 Hz acceleration 5 g, constant amplitude
Shocks	1000 shocks in each direction, in 3 axes
	35 g, 11 ms

3.3 Scope of supply

Make sure that the following components are included in the scope of delivery:

- 1 x EPP6228-0022 EtherCAT P Box
- 2x protective cap for EtherCAT P socket, M8, red (pre-assembled)
- 10x labels, blank (1 strip of 10)

Pre-assembled protective caps do not ensure IP67 protection

Protective caps are pre-assembled at the factory to protect connectors during transport. They may not be tight enough to ensure IP67 protection.

Ensure that the protective caps are correctly seated to ensure IP67 protection.

3.4 Process image

Assignment of IO-Link ports to process data

Process image in TwinCAT	IO-Link port	Status variable	Process data of the IO-Link device	Digital input
Box 1 (EP6228-0022) Module 1 (DeviceState Inputs Device) DeviceState Inputs Device Device Diag Device State Module 2 (DeviceState Inputs) DeviceState Inputs State Ch1 State Ch2 State Ch3 State Ch4 State Ch5 State Ch6 State Ch7 State Ch8 Module 3 (IO-Link Slave) Module 4 (IO-Link Slave) Module 5 (IO-Link Slave) Module 6 (IO-Link Slave) Module 7 (IO-Link Slave) Module 8 (IO-Link Slave) Module 9 (IO-Link Slave) Module 10 (IO-Link Slave) WcState InfoData	1	Module 2 DeviceState Inputs State Ch1	Module 3 [1)]	Module 11 Digital Inputs Pin2 Ch1
	2	Module 2 DeviceState Inputs State Ch2	Module 4 [1)]	Module 11 Digital Inputs Pin2 Ch2
	3	Module 2 DeviceState Inputs State Ch3	Module 5 [1)]	Module 11 Digital Inputs Pin2 Ch3
	4	Module 2 DeviceState Inputs State Ch4	Module 6 [1)]	Module 11 Digital Inputs Pin2 Ch4
	5	Module 2 DeviceState Inputs State Ch5	Module 7 [1)]	Module 11 Digital Inputs Pin2 Ch5
	6	Module 2 DeviceState Inputs State Ch6	Module 8 [1)]	Module 11 Digital Inputs Pin2 Ch6
	7	Module 2 DeviceState Inputs State Ch7	Module 9 [1)]	Module 11 Digital Inputs Pin2 Ch7
	8	Module 2 DeviceState Inputs State Ch8	Module 10 [1)]	Module 11 Digital Inputs Pin2 Ch8

[1)] The modules "Module 3" to "Module 10" only exist in the process data if the corresponding IO-Link ports have been configured.

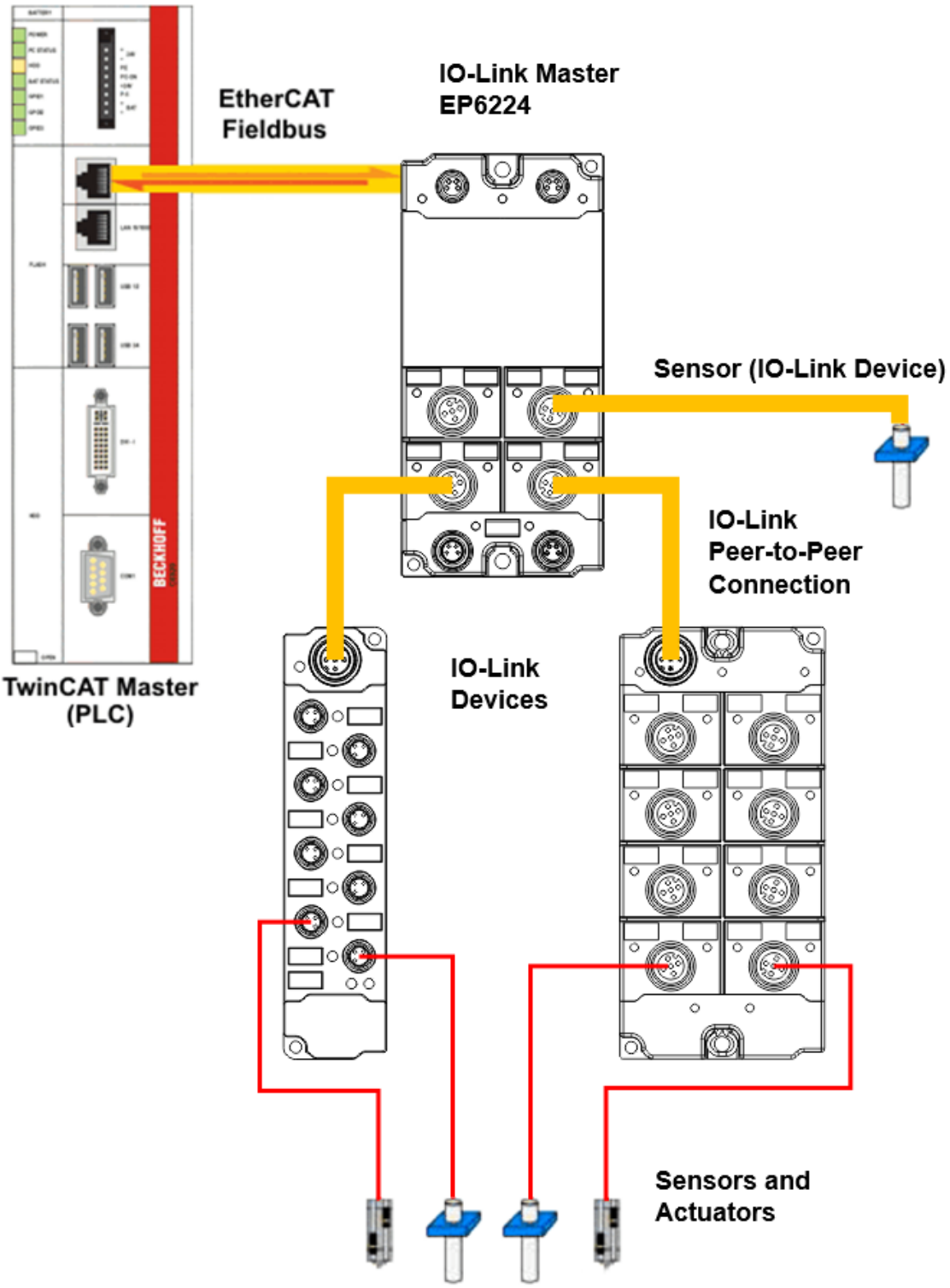
4 IO-Link basics

IO-Link is a communication system for connecting intelligent sensors and actuators to an automation system. The IEC 61131-9 standard specifies IO-Link under the designation "Single-drop digital communication interface for small sensors and actuators" (SDCI).

Both the electrical connection data and the communication protocol are standardized and summarized in the IO-Link Spec.

4.1 IO-Link system layout

An IO-Link system consists of an IO-Link master and one or more IO-Link devices, i.e. sensors or actuators. The IO-Link master provides the interface to the higher-level controller and controls communication with the connected IO-Link devices.
The IO-Link masters from Beckhoff have several IO-Link ports, to each of which one IO-Link device can be connected. IO-Link is not a fieldbus, but rather a point-to-point connection.



⚠ CAUTION

Risk of device damage

The IO-Link devices must be supplied from the 24 V power supply of the IO-Link master provided for this purpose. Otherwise, damage to the IO-Link port is possible.

4.2 Establishment of IO Link communication

The establishment of the IO-Link communication is illustrated in Fig. *Establishment of IO-Link communication.* This illustrates in particular the sequence when automatically scanning [▶ 35] the IO-Link port.

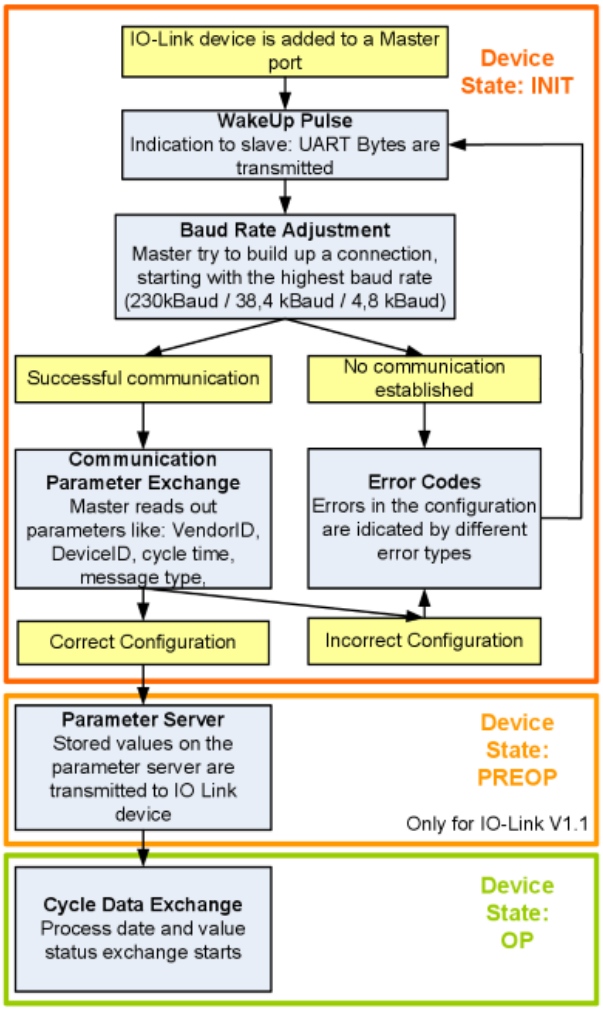


Fig. 1: Establishment of IO Link communication

- If an IO-Link device is connected to a master port, the master attempts to establish communication. A defined signal level, the **wake-up pulse**, signals to the device that UART bytes are to be sent from now on.
  From this point on, all data will be interpreted by the IO-Link device as UART bytes.
- The master runs through all baud rates [▶ 16], starting with the fastest baud rate (COM3 = 230 kbaud). A successful connection has been established when the slave responds to the wake-up pulse.
- First of all the master reads the **basic parameters** (Vendor ID, Device ID, process data length, telegram type and cycle time) and compares them with the existing configuration.
- If no connection could be established to the device, or if the saved parameters differ from those read, the corresponding error is output.
- If the saved parameters differ from those read, the IO-Link device changes to the PREOP state.
  If the IO-Link device specification is V1.1, the parameter server [▶ 16] is now executed. If the IO-Link device specification is V1.0, this step is omitted and the device changes directly to OP.
- Finally the cycle time is written and the device changes to OP. After that the master cyclically exchanges data with the device.

4.3 Device description IODD

IO-Link devices possess individual system information in the form of an IO device description (IODD), which contains:

- Communication features
- Device parameters with value range and default values
- Identification, process and diagnostic data
- Device data
- Text description
- Picture of the device
- Vendor's logo

If the IODD is imported, then the device data are automatically detected during automatic scanning [▶ 35] with TwinCAT and adopted in the System Manager.

4.4 Parameter server

In order to be able to use the functionality of the parameter server, both the IO-Link master and the IO-Link device must be specified to V1.1. The IO-Link revision of the device can be read for the individual port under Settings [▶ 42]. All IO-Link masters from Beckhoff with current firmware support the IO-Link specification V1.1.

- The parameter server in the IO-Link master contains parameter data that are saved in the IO-Link device. The memory capacity is max. 2 kbyte (including header).
  If the IO-Link device is exchanged, then the data are loaded from the parameter server into the new device. The requirement for this is that the device is of the same type (VendorID and DeviceID must be the same).
- If a new IO-Link device is configured, then the IO-Link master loads the parameters from the IO-Link device into the parameter server when starting for the first time.
  Data from other IO-Link devices that are already configured (VendorID and DeviceID do not correspond to the configured device) are overwritten.
- At each further start the IO-Link master uses a checksum to check whether the data in the parameter server correspond to those on the IO-Link device and if necessary downloads them to the device.
- If the parameters change during the device runtime, this can be reported to the Master via the store button [▶ 50] (ParamDownloadStore [▶ 51]). The master then starts the parameter server with an upload.
- By default the event is not set each time the parameters are written, therefore the end of the parameterization procedure has to be reported to the IO-Link device via the store button [▶ 50] (ParamDownloadStore [▶ 51]).
  The IO-Link device then sends the corresponding event to the master. The data are loaded into the parameter server.
- In the case of a pre-programmed IO-Link device, no download takes place from the parameter server to the device.

4.5 Data transfer rate

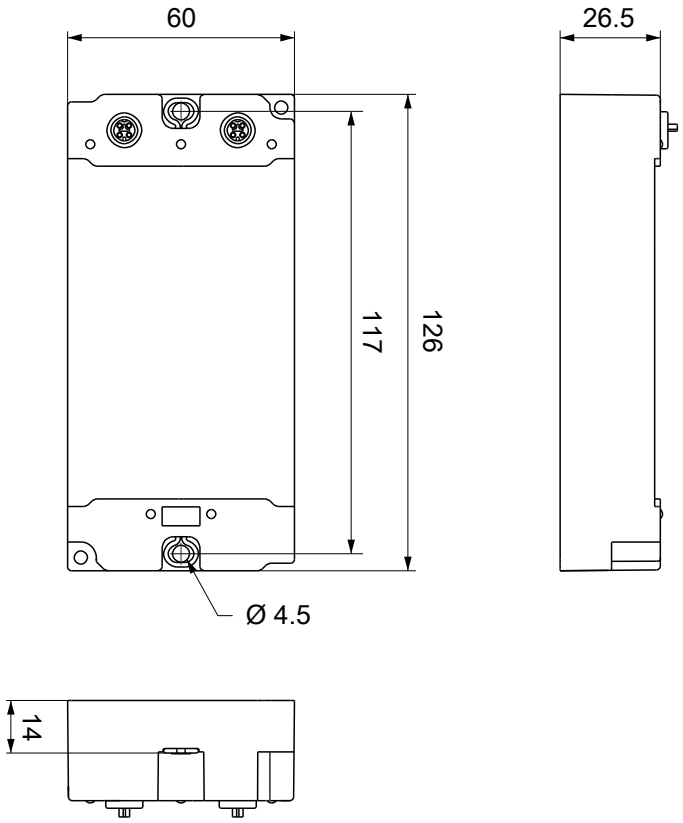
An IO-Link master according to specification V1.1 supports all three transmission methods and automatically adjusts the baud rate to that of the IO-Link device.
An IO-Link device usually supports only one baud rate. IO-Link devices with different baud rates can be connected to the various ports of the master.

- COM1 = 4.8 kbaud
- COM2 = 38.4 kbaud
- COM3 = 230.4 kbaud

5 Mounting and cabling

5.1 Mounting

5.1.1 Dimensions



All dimensions are given in millimeters.
The drawing is not true to scale.

Housing features

Housing material	PA6 (polyamide)
Sealing compound	polyurethane
Mounting	two mounting holes Ø 4.5 mm for M4
Metal parts	brass, nickel-plated
Contacts	CuZn, gold-plated
Installation position	variable
Protection class	IP65, IP66, IP67 (conforms to EN 60529) when screwed together
Dimensions (H x W x D)	approx. 126 x 60 x 26.5 mm (without connectors)

5.1.2 Fixing

NOTE

Dirt during assembly

Dirty connectors can lead to malfunctions. Protection class IP67 can only be guaranteed if all cables and connectors are connected.

- Protect the plug connectors against dirt during the assembly.

Mount the module with two M4 screws in the centrally located mounting holes.

5.2 Functional earth (FE)

The mounting holes also serve as connections for the functional earth (FE).

Make sure that the box is grounded to low impedance via the functional earth (FE) connections. You can achieve this, for example, by mounting the box on a grounded machine bed.

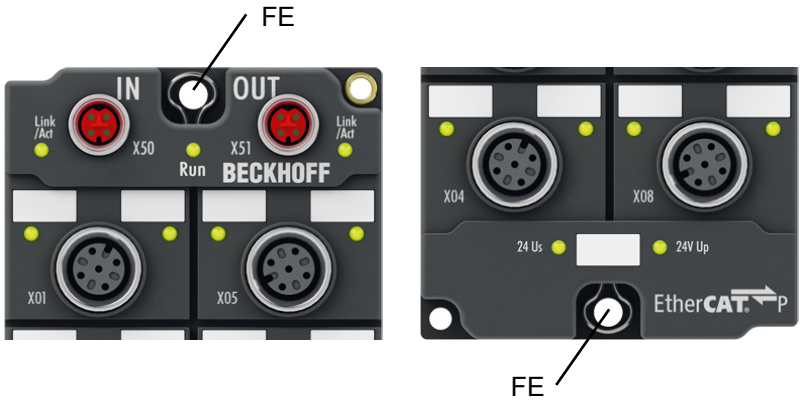


Fig. 2: Connection for functional earth (FE)

5.3 Cabling

Tightening torques for plug connectors

Screw connectors tight with a torque wrench. (e.g. ZB8801 from Beckhoff)

Connector diameter	Tightening torque
M8	0.4 Nm
M12	0.6 Nm

5.3.1 EtherCAT P

⚠ WARNING

Power supply from SELV/PELV power supply unit!

SELV/PELV circuits (Safety Extra Low Voltage, Protective Extra Low Voltage) according to IEC 61010-2-201 must be used to supply the EtherCAT P Power Sourcing Device (PSD).

Notes:

- SELV/PELV circuits may give rise to further requirements from standards such as IEC 60204-1 et al, for example with regard to cable spacing and insulation.
- A SELV (Safety Extra Low Voltage) supply provides safe electrical isolation and limitation of the voltage without a connection to the protective conductor,
  a PELV (Protective Extra Low Voltage) supply also requires a safe connection to the protective conductor.

⚠ CAUTION

Observe the UL requirements

- When operating under UL conditions, observe the warnings in the chapter UL Requirements [▶ 26].

EtherCAT P transmits two supply voltages:

- **Control voltage U_S**
  The following sub-functions are supplied from the control voltage U_S:
  - the fieldbus
  - the processor logic
  - typically the inputs and the sensors if the EtherCAT P Box has inputs.
- **Peripheral voltage U_P**
  The digital outputs are typically supplied from the peripheral voltage U_P for EtherCAT P Box modules with digital outputs. U_P can be supplied separately. If U_P is switched off, the fieldbus function, the function of the inputs and the supply of the sensors are maintained.

The exact assignment of U_S and U_P can be found in the pin assignment of the I/O connections.

Redirection of the supply voltages

The supply voltages are passed on internally from the "IN" connection to the "OUT" connection. Hence, the supply voltages U_S and U_P can be passed from one EtherCAT P Box to the next EtherCAT P Box in a simple manner.

NOTE

Note the maximum current.

Ensure that the maximum permitted current of 3 A for the M8 connectors is not exceeded when redirecting EtherCAT P.

5.3.1.1 Connectors

NOTE

Risk of damage to the device!

Bring the EtherCAT/EtherCAT P system into a safe, powered down state before starting installation, disassembly or wiring of the modules!

Two M8 sockets at the upper end of the modules are provided for supply and downstream connection of EtherCAT P:

- IN: left M8 socket for EtherCAT P supply
- OUT: right M8 socket for downstream connection of EtherCAT P

The metal threads of the M8 EtherCAT P sockets are internally linked to the FE connection via high impedance RC combination. See chapter Functional earth (FE) [▶ 18].

Fig. 3: Connectors for EtherCAT P

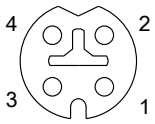


Fig. 4: M8 socket, p-coded

Contact	Signal	Voltage	Core color [1)]
1	Tx +	GND_S	yellow
2	Rx +	GND_P	white
3	Rx -	U_P: peripheral voltage, +24 V_{DC}	blue
4	Tx -	U_S: control voltage, +24 V_{DC}	orange
Housing	Shield	Shield	Shield

[1)] The core colors apply to EtherCAT P cables and ECP cables from Beckhoff.

5.3.1.2 Status LEDs

5.3.1.2.1 Supply voltages

EtherCAT P Box modules indicate the status of the supply voltages via two status LEDs. The status LEDs are labeled with the designations of the supply voltages: Us and Up.

LED	Display	Meaning
U_S (control voltage)	off	The supply voltage U_S is not available.
	green illuminated	The supply voltage U_S is available.
U_P (peripheral voltage)	off	The supply voltage U_P is not available.
	green illuminated	The supply voltage U_P is available.

5.3.1.2.2 EtherCAT

L/A (Link/Act)

A green LED labeled "L/A" or "Link/Act" is located next to each EtherCAT/EtherCAT P socket. The LED indicates the communication state of the respective socket:

LED	Meaning
off	no connection to the connected EtherCAT device
lit	LINK: connection to the connected EtherCAT device
flashes	ACT: communication with the connected EtherCAT device

Run

Each EtherCAT slave has a green LED labelled "Run". The LED signals the status of the slave in the EtherCAT network:

LED	Meaning
off	Slave is in "Init" state
flashes uniformly	Slave is in "Pre-Operational“ state
flashes sporadically	Slave is in "Safe-Operational" state
lit	Slave is in "Operational" state

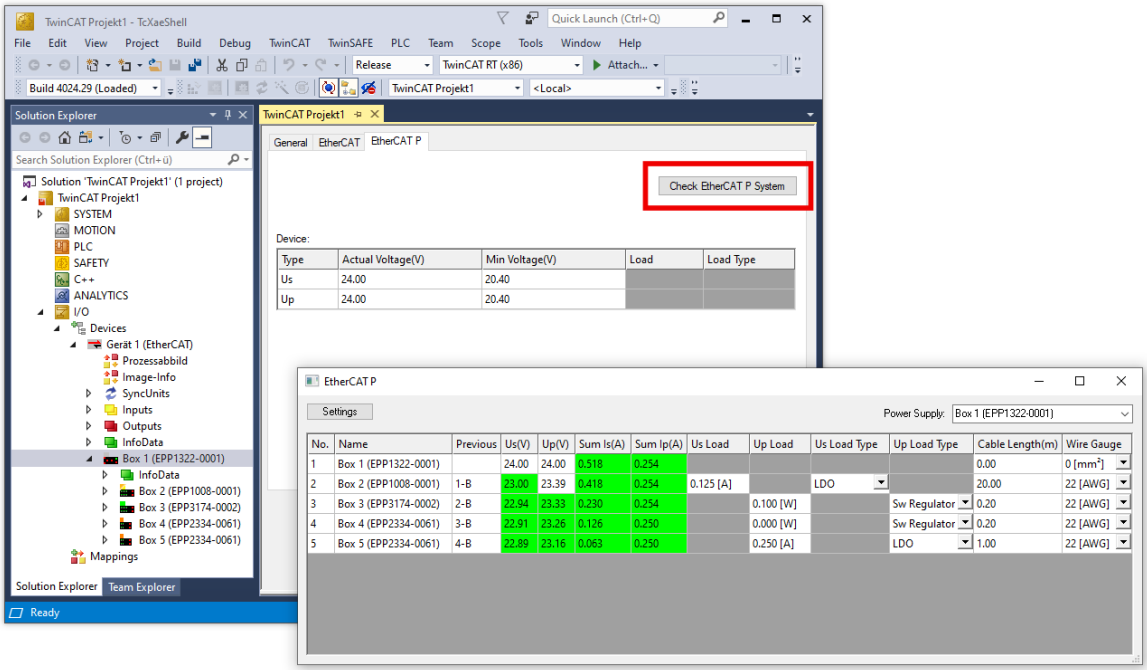
Description of the EtherCAT slave states

5.3.1.3 Conductor losses

Take into account the voltage drop on the supply line when planning a system. Avoid the voltage drop being so high that the supply voltage at the box lies below the minimum nominal voltage.
Variations in the voltage of the power supply unit must also be taken into account.

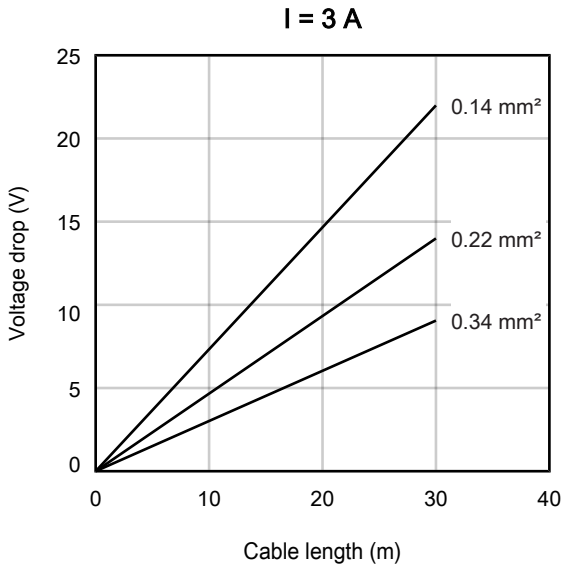
Planning tool for EtherCAT P

You can plan cable lengths, voltages and currents of your EtherCAT P system using TwinCAT 3. The requirement for this is TwinCAT 3 Build 4020 or higher.



Further information can be found in the quick start guide IO configuration in TwinCAT in chapter "Configuration of EtherCAT P via TwinCAT".

Voltage drop on the supply line



5.3.2 IO-Link

5.3.2.1 Connectors

The IO-Link ports are implemented as M12 sockets.

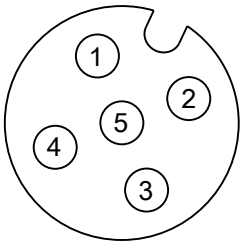


Fig. 5: M12 socket

Contact	Function	Description	Core color [1]
1	L+	Supply voltage (U_{S1})	brown
2	DI	Digital input	white
3	L-	GND	blue
4	C/Q	IO-Link data cable	black
5	-	-	grey

[1] The core colors apply to M12 sensor cables from Beckhoff:

- ZK2000-5xxx
- ZK2000-6xxx
- ZK2000-7xxx.

5.3.2.2 Status LEDs

Fig. 6: Status LEDs of an IO-Link port

1 - IO-Link

LED signal	Meaning
off	Possibilities: • Port not configured • Logic level low [1)]
red illuminated flashes sporadically green	Possibilities: • IO-Link connection attempt • No IO-Link device connected • Incorrect IO-Link device connected • IO-Link device defective
flashing green	IO-Link communication active
green illuminated	Logic level high [1)]

[1)] Port configured as digital input or output

2 - Digital input DI

The LED lights up when a high level is present on the digital input DI.

5.4 UL Requirements

The installation of the EtherCAT Box Modules certified by UL has to meet the following requirements.

Supply voltage

⚠ CAUTION

CAUTION!

This UL requirements are valid for all supply voltages of all marked EtherCAT Box Modules!
For the compliance of the UL requirements the EtherCAT Box Modules should only be supplied

- by a 24 V_{DC} supply voltage, supplied by an isolating source and protected by means of a fuse (in accordance with UL248), rated maximum 4 Amp, or
- by a 24 V_{DC} power source, that has to satisfy *NEC class 2*.
  A *NEC class 2* power supply shall not be connected in series or parallel with another (class 2) power source!

⚠ CAUTION

CAUTION!

To meet the UL requirements, the EtherCAT Box Modules must not be connected to unlimited power sources!

Networks

⚠ CAUTION

CAUTION!

To meet the UL requirements, EtherCAT Box Modules must not be connected to telecommunication networks!

Ambient temperature range

⚠ CAUTION

CAUTION!

To meet the UL requirements, EtherCAT Box Modules has to be operated only at an ambient temperature range of -25 °C to +55 °C!

Marking for UL

All EtherCAT Box Modules certified by UL (Underwriters Laboratories) are marked with the following label.

Fig. 7: UL label

5.5 Disposal

Products marked with a crossed-out wheeled bin shall not be discarded with the normal waste stream. The device is considered as waste electrical and electronic equipment. The national regulations for the disposal of waste electrical and electronic equipment must be observed.

6 Commissioning and configuration

6.1 Integrating into a TwinCAT project

The procedure for integration in a TwinCAT project is described in these Quick start guide.

6.2 Configuration of the IO link master

EtherCAT XML device description and configuration files

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.

When adding the IO-Link master (see chapter Integrating into a TwinCAT project [▶ 28]) in the TwinCAT System Manager, an additional tab called "IO-Link" is created (fig. *IO-Link tab*). A detailed description can be found in chapter Configuration of the IO-Link devices [▶ 30]

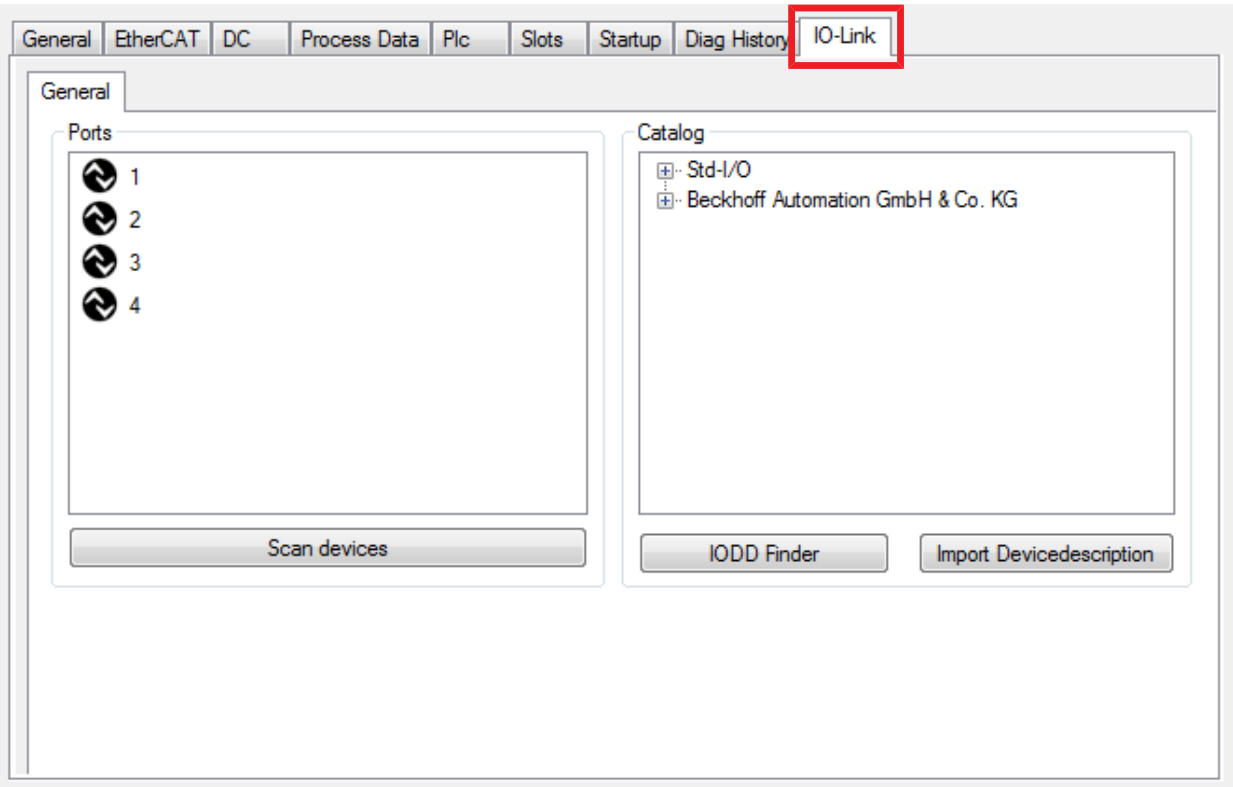


Fig. 8: "IO Link" tab

IO-Link Extension

If the tab “IO-Link” is not displayed, the associated System Manager extension is missing. The System Manager extension is required for TwinCAT version 2.10, build 1325 to 1330.

- If your System Manager version or TwinCAT3 does not yet provide this support, it can be installed later if necessary. Please contact our Support [▶ 79].

6.3 Configuration of the IO-Link devices

The configuration of the IO link devices is carried out in the IO link configuration tool. Configure the IO link device as described below.

✓ Requirement: an IO-Link master has been added in the Solution Explorer under the "I/O" entry.

1. Open the IO link configuration tool [▶ 30].
2. Import the IODD file of the IO link device [▶ 34].
3. Assign devices to ports.
   ⇨ Assign a device to a port [▶ 31].
   ⇨ Configure a port as digital in- or output [▶ 34].
4. Remove a device from a port [▶ 40].
5. Activate the IO link configuration [▶ 41], so that changes become effective.

6.3.1 Open the IO link configuration tool

✓ Requirement: an IO-Link master has been added in the Solution Explorer under the "I/O" entry.

1. Double-click on the IO-Link master.
   ⇨ A device editor for the IO-Link master opens.
2. Click on the "IO-Link" tab.
   ⇨ The IO-Link configuration tool opens. The configuration tool contains two fields:

- „Ports“
  The left-hand field "Ports" shows a list of the ports of the IO-Link master. If a device has been assigned to a port, the device designation is shown next to the port.
- „Catalog“
  The right-hand field "Catalog" shows the device catalog.
  The device catalog contains an alphabetically sorted list of the IO-Link devices for which a device description (IODD) exists in the local TwinCAT installation.
  The IODDs for the EPIxxxx, ERIxxxx IO-Link Box modules from Beckhoff can be downloaded via the Download finder. The downloaded zip file contains the IODD device description files for the Beckhoff EPIxxxx, ERIxxxx IO-Link Box modules.

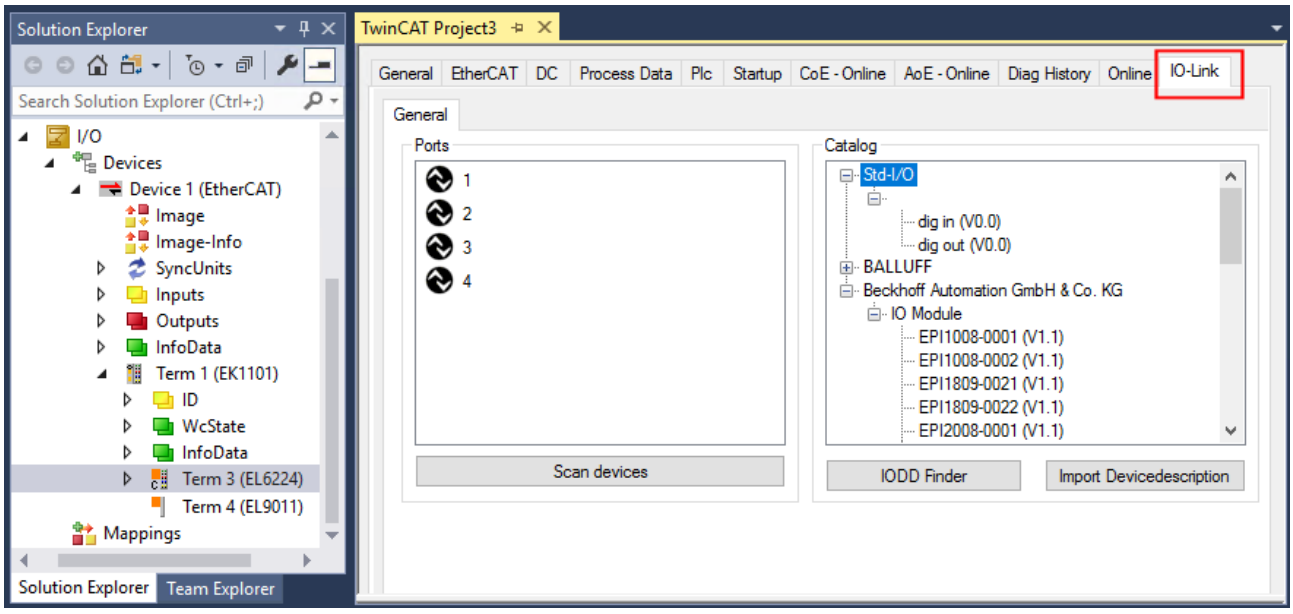


Fig. 9: IO-Link configuration tool

6.3.2 Integrating IO-Link devices

The integration of the IODD file should always be the first step, since this enables the breakdown of the individual process data of the IO-Link devices as well as the display of the parameters.

There are several ways of integrating an IO-Link device:

1. Importing the IODD file (offline and online) via
   - ⇨ button Import Devicedescription [▶ 32] (A) or
   - ⇨ button IODD Finder [▶ 32] (B)
2. Select the device in the "Catalog" field and assign it to a port [▶ 34].
3. Automatic scanning of the IO-Link ports (online) via
   - ⇨ button Scan devices [▶ 35] (C)
4. Manual insertion (offline and online) via
   - ⇨ menu Create Device [▶ 39] (D)

Application note

- If the IODD is not available,
  the IO-Link device should be integrated online by scanning.
- Manual integration of the IO-Link devices via “Create Device” should only be carried out if the IODD of the vendor and the IO-Link device are not available at the time of project creation.

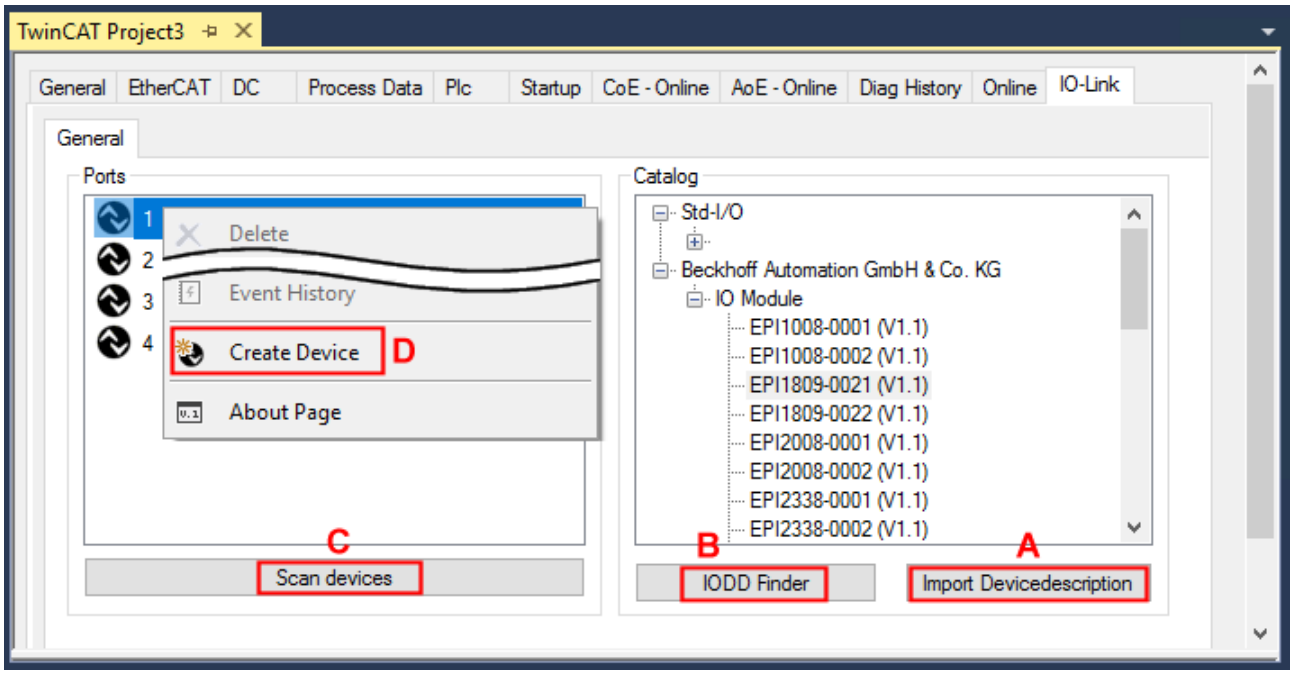


Fig. 10: Creating IO-Link devices

6.3.2.1 1. Importing the device description IODD

Importing the device description simplifies the integration of the IO-Link devices. The individual process data are broken down, enabling simple parameterization of the sensor. The IODD only needs to be imported during the initial commissioning of a new IO-Link device. The import is port-independent. Proceed as follows to import the IODD:

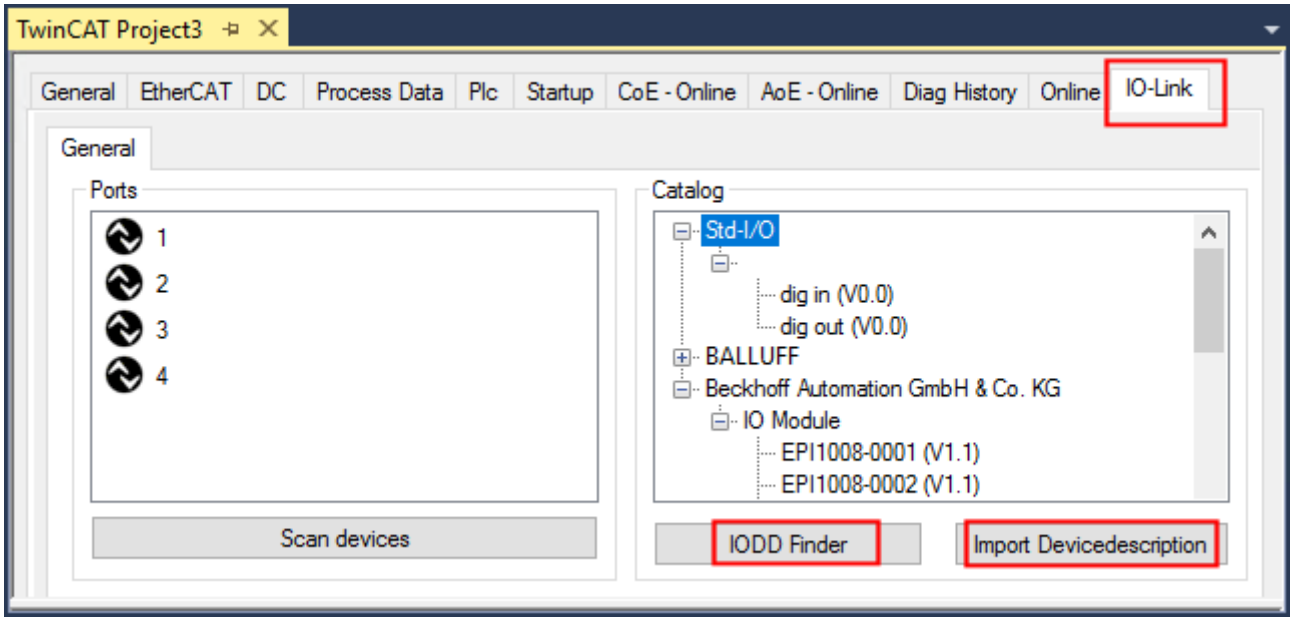


Fig. 11: Import of the IODD device description via “IODD Finder” or “Import Devicedescription”

Button "Import Devicedescription“

1. Press the “Import Devicedescription” button in the “IO-Link” tab
2. Select the .xml file of the desired sensor.
3. After pressing the Open button, the imported files are stored in the following folder:
   - for TwinCAT 2.x: \TwinCAT\IO\IOLink
   - for TwinCAT 3.x: \TwinCAT\3.X\Config\IO\IOLink.

⇨ The imported device descriptions are listed in a tree structure in the “Catalog” field, sorted by vendor.

No manual copying of the XML files

Do not copy the files directly into the folder; read them in via *Import Devicedescription* instead! Important checks will otherwise be bypassed!

Button "IODD Finder“

1. Press the “IODD Finder” button in the “IO-Link” tab
2. Searching for the desired IO-Link sensor/device by entering them in the search mask; see the figure below (1)
3. Selecting the desired IO-Link sensor/device. Move the mouse pointer over the figure of the desired IO-Link sensor/device. A blue download icon appears, see the following figure (2).

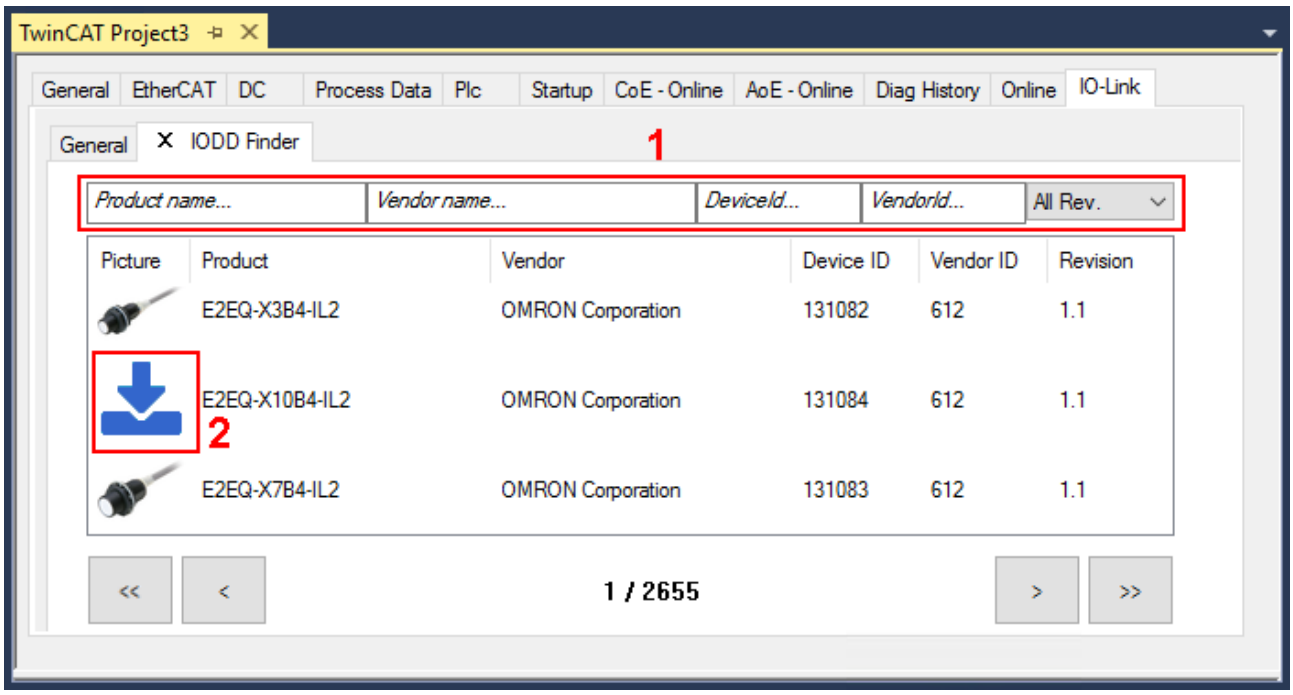


Fig. 12: IODD Finder, selection and import of the .xml-file

4. After clicking the download symbol, the .xml file of the selected IO-Link sensor/device is imported and stored in the following folder:
   - for TwinCAT 2.x: \TwinCAT\IO\IOLink
   - for TwinCAT 3.x: \TwinCAT\3.X\Config\IO\IOLink
5. When moving the mouse pointer over the IO-Link sensor/device, a green icon now indicates (see the following figure (3)) that the .xml file already exists.

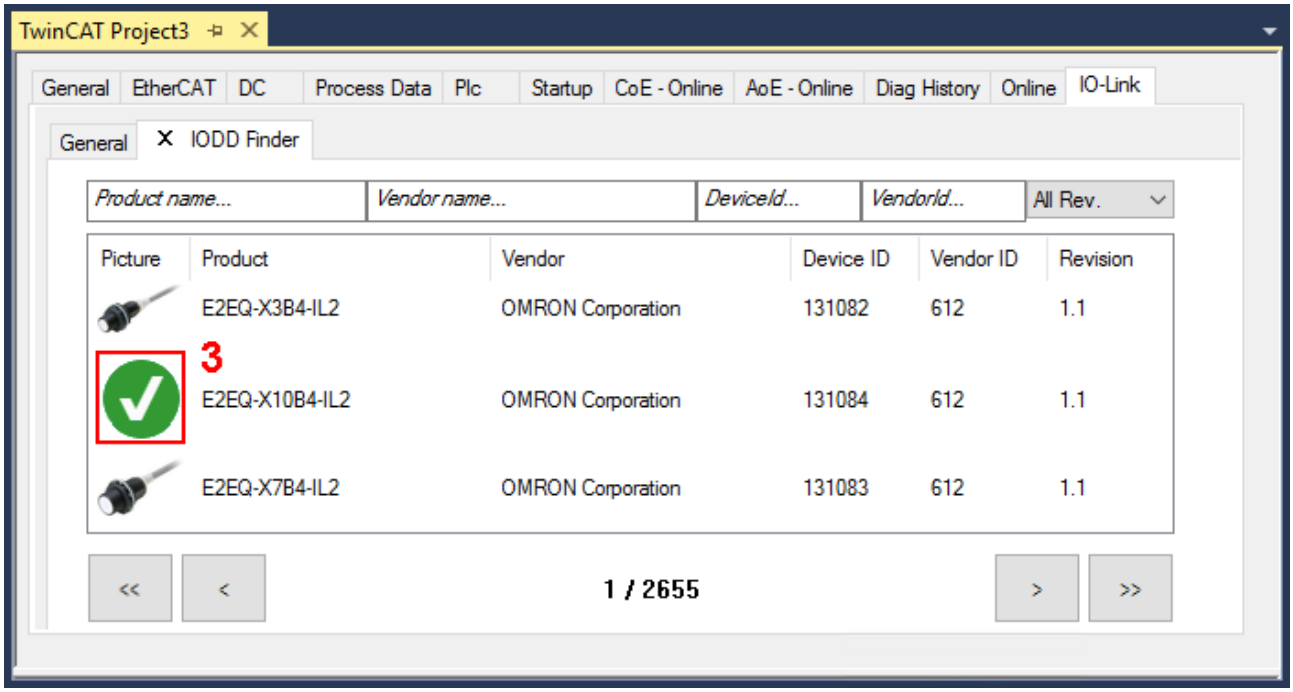


Fig. 13: IODD Finder, display of an already imported device description

⇨ The imported device descriptions are listed in a tree structure in the “Catalog” field of the IO-Link tab, sorted by vendor.

6.3.2.2 2. Assigning IO-Link device to port n

Online configuration

✓ Requirement: The IO-Link device is connected.

1. Press the button Scan devices (see chapter Automatic scanning [▶ 35])

⇨ The device is automatically detected and created with the corresponding parameters. If several devices are stored in the IODD file, the first entry is always selected here. Grouping in the IODD is usually carried out by the vendor if the process data are the same and there are only mechanical differences (e.g. other material).

Offline configuration

The *Catalog* field shows the IO-Link device catalog, which lists the already imported device descriptions in a tree structure, sorted by vendor.

1. Select the desired IO-Link device from the *Catalog* field
   - via drag and drop or
   - by right-clicking on the product with "Add to Port n".

Activating the configuration

2. Activate the IO link configuration [▶ 41], so that changes become effective.

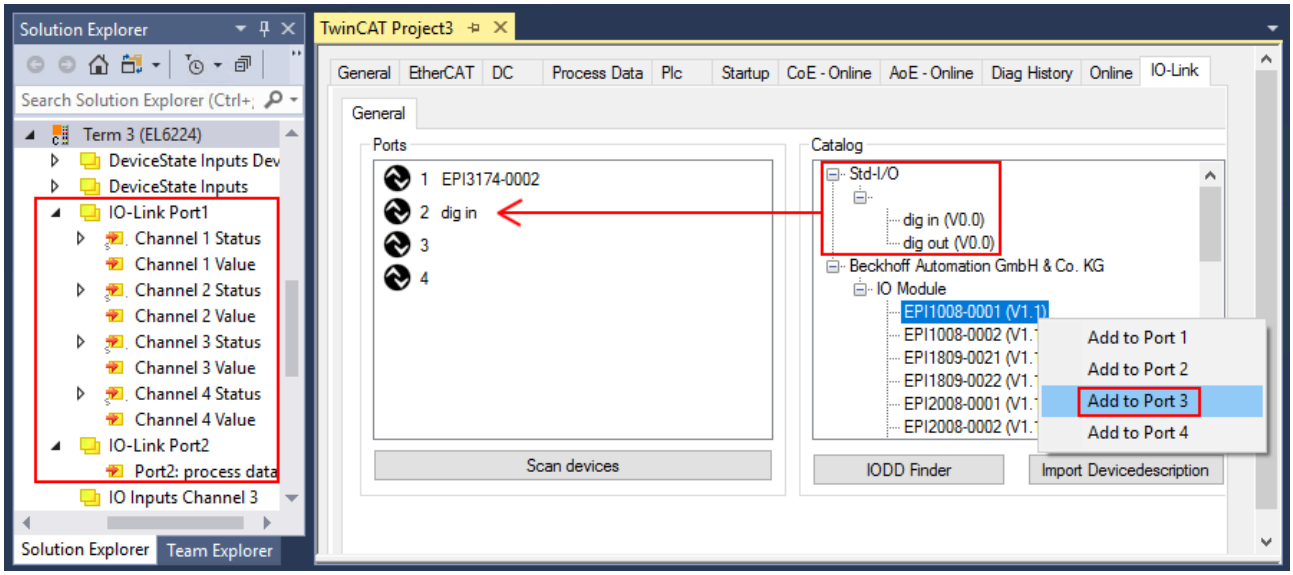
⇨ The IO-Link devices are displayed, and the process data are created. If an error is found when integrating the IO-Link device, e.g. wrong VendorID or no device connected, then this is indicated via the status of the port (object state Ch.n 0xF100:0n).

Configuration of the IO-Link ports as digital in- or output

IO-Link ports can also be configured as digital inputs or digital outputs. This allows digital sensors and actuators having no IO-Link functionality to be connected to IO-Link ports.

1. Expand the "Std-I/O" tree node in the "Catalog" field.
   ⇨ The operating modes "dig in" and "dig out" appear.
2. Configure the desired port. There are two ways to do this:
   - Drag-and-drop: pull "dig in" or "dig out" onto the port in the "Ports" field or
   - Right-click on "dig in" or "dig out" and click on "Add to Port n“.

Example of port assignment on the IO link master EL6224



Port1:
EPI3174-0002 is assigned

Port2:
is configured as digital input

Port3:
EPI1008-0001 will be assigned

Process data of Port1 and Port2 are displayed in the Solution Explorer.

6.3.2.3 3. Automatic scanning of the IO-Link ports

This part of the documentation describes the configuration of the physically available IO-Link devices in TwinCAT.

During automatic scanning of the IO-Link ports, the steps “WakeUp pulse”, “Baud rate setting”, “Reading of the communication parameters”, plus “Parameter server” and “Cyclic data exchange”, if applicable, are performed, see Establishing the IO-Link communication [▶ 15]. The corresponding IO-Link device must be connected to the IO-Link port for this.

The connected devices are automatically detected, configured and a search is performed for the associated IODD.

Finding connected IO-Link devices

✓ Requirement: the master and the devices are cabled and supplied with voltage.

1. Click on the “Scan devices” button (see the following figure).

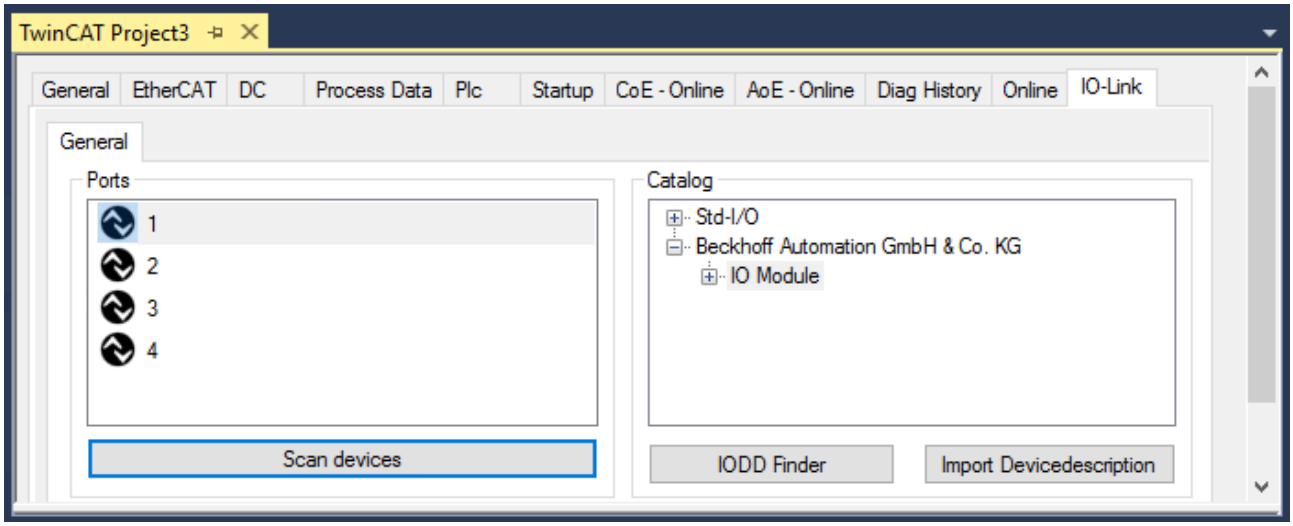


Fig. 14: Scan devices

⇨ The connected IO-Link devices can be found.

⇨ The information window lists the connected device for each of the four ports. Only port2 of the master is assigned an IO-Link device.

⇨ Confirm with the *OK* button.

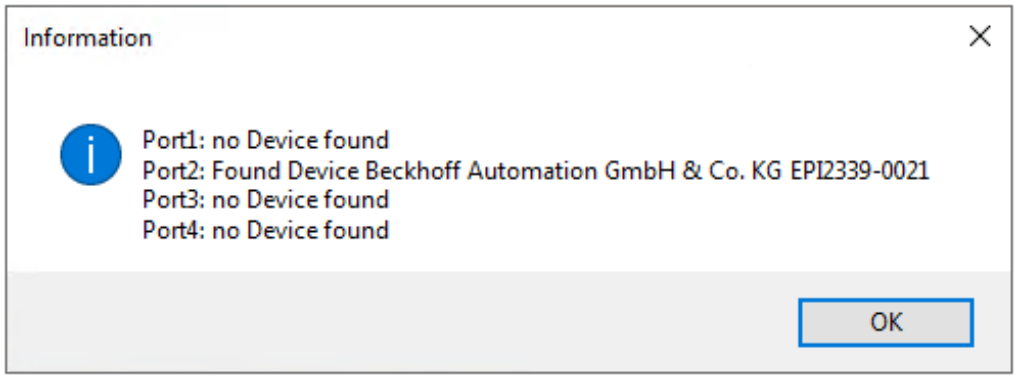


Fig. 15: Information “Scan devices”

2. To be able to work with the devices, the button “Reload Devices” must be clicked. 

The IO-Link devices are now entered in the *General* display. The Port2 “Details” field displays information about the connected device. Additionally the tabs Settings [▶ 37] and Parameter [▶ 38] can be opened.

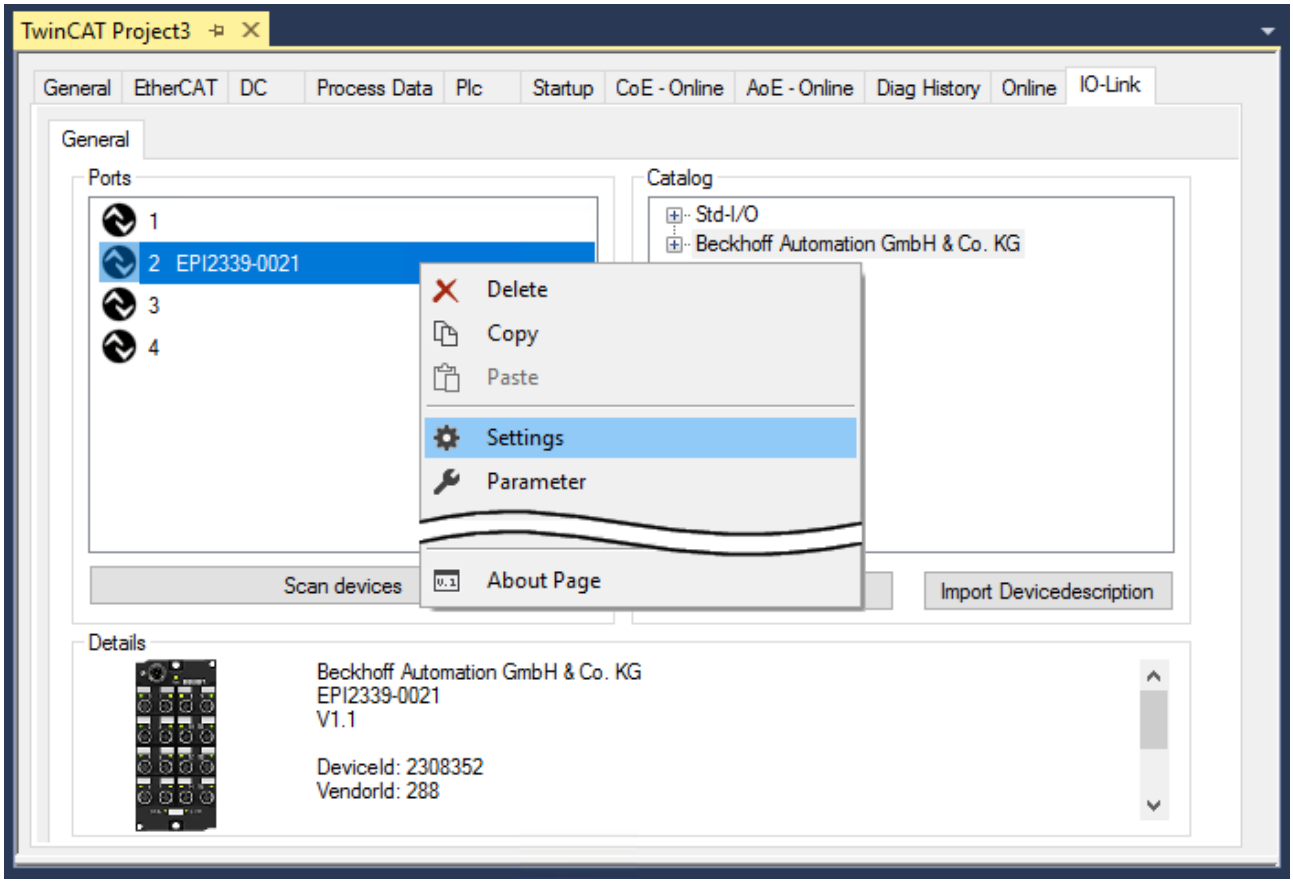


Fig. 16: Device at Port2, Display “Details”, open tabs “Settings” and “Parameter”

Show settings of the device

3. Right-click on port2, to display more details in dialog “Settings”.
4. If necessary, change the settings as described in chapter Settings of the IO-Link devices [▶ 42].

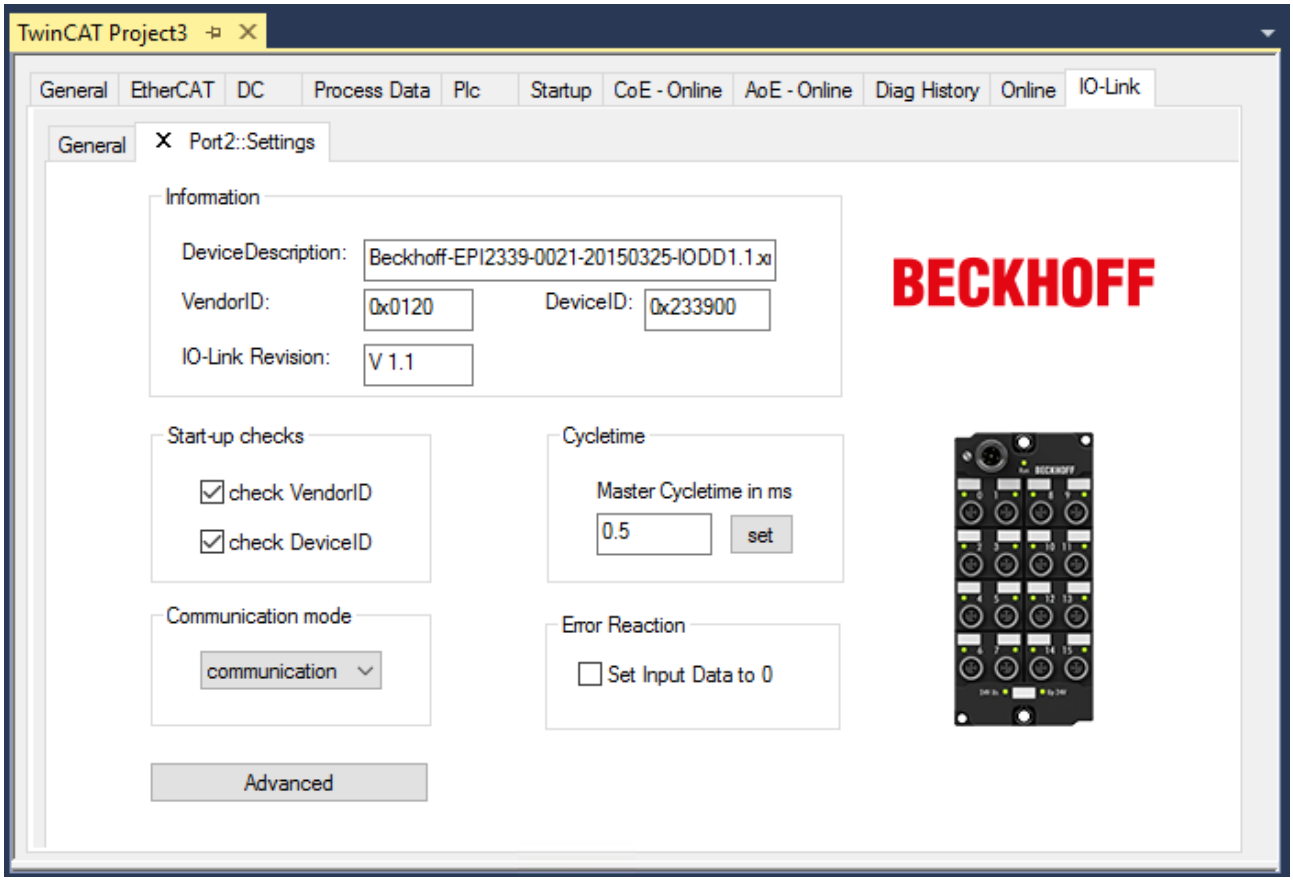


Fig. 17: Settings of the device assigned to port2

Show parameters of the device

5. Open the Parameter tab via
   - double-click on Port2 or
   - richt-click on Port2 and select “Parameter” in the menu.

   ⇨ The Parameters of of the respective IO link device are listed.
6. Parameterize the device as described in chapter EPIxxxx, ERIxxxx - Setting of the IO-Link device parameters [▶ 44].

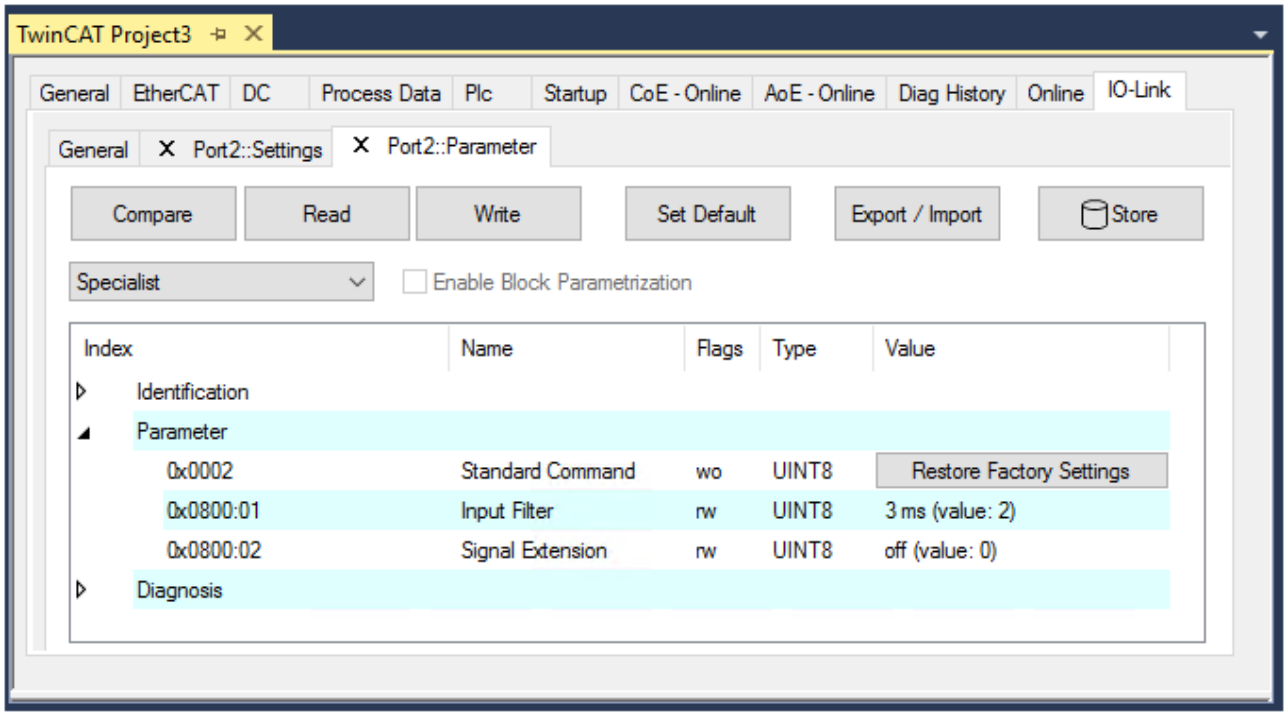


Fig. 18: Parameter of the device assigned to port2

6.3.2.4 4. Manual insertion via Create Device

This part of the documentation describes the manual configuration of the IO-Link devices in TwinCAT.

The manual insertion of the IO-Link device should only be carried out if the IODD from the vendor and the IO-Link device are not available. By saving the project, the settings for the individual ports are saved. The devices that were created are **not** stored in the “Catalog” (see the figure below (A)). To insert the IO-Link devices manually via “Create Device”, proceed as follows:

1. The IODD of the IO link device is already available:
   Select the respective device from the “Catalog” field sorted by manufacturer (see following figure (A)).
2. No IODD is available:
   Add the device can be manually via “Create Device”. These data are **not** saved in the “Catalog” field and must be manually entered for each port.
3. Right-click on the port to open the context menu (see the figure below (B)) and select “Create Device”.
4. In the “Create Device” dialog an IO-Link device with the basic communication parameters can be created. The mandatory fields here are: For Vendor ID, Device ID and process data length see the figure below (C). The values VendorID and DeviceID can be entered both in hexadecimal notation (input format: 0xnnnn) and as decimal numbers (nnnn).
   The communication parameters to be entered can be found in the information provided by the device vendor.
5. If the IO-Link device version is 1.1, then the parameter server is activated by the selection of the check box “Revision V1.1” (see following figure (D)).
6. Activate the IO link configuration [▶ 41], so that changes become effective.

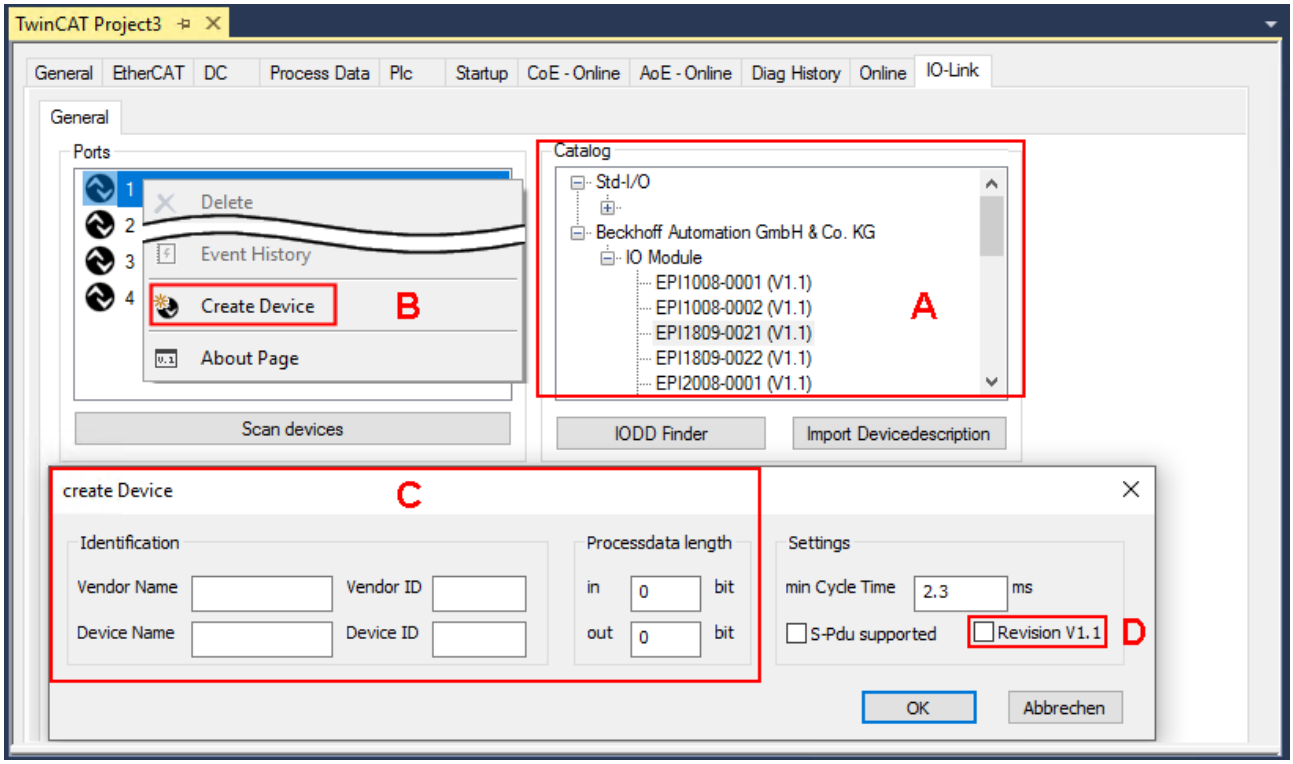


Fig. 19: Manual creation of an IO-Link device via the "Create Device" dialog (C)

Reading the IODD

Even when manually creating and scanning, the IODD should always be read in as well in order to display further sensor-specific information.

7. In the “Settings” tab of the IO link devices further settings can be made as described in chapter Settings of the IO-Link devices [▶ 42].

6.3.3 Removal of IO-Link devices

To remove a device that has already been inserted, proceed as follows.

1. Right-click on the port to open the context menu and select “Delete”.

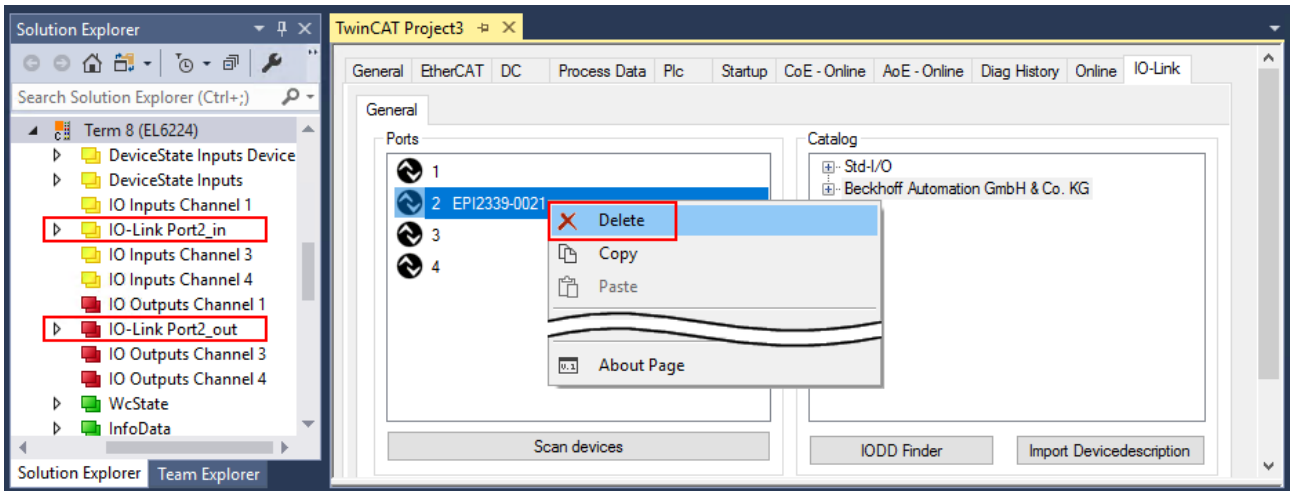


Fig. 20: Remove the device from port2

2. Activate the IO link configuration [▶ 41], so that changes become effective.

⇨ The already create process data are removed.

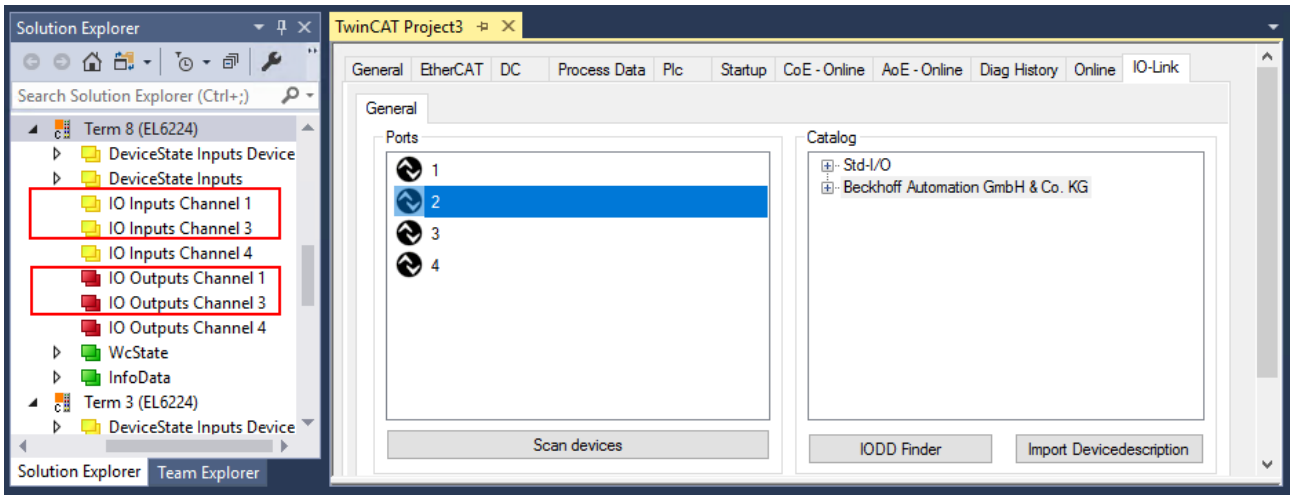


Fig. 21: The device was removed from the port2, the process data no longer displayed in the tree.

6.3.4 Activating the configuration

Changes in the IO-Link configuration tool only become effective when you activate the IO-Link configuration.

There are two ways to activate the IO-Link configuration:

- Click on the "Reload Devices" button

- Activate the TwinCAT configuration:
  Click on the "Activate Configuration" button

6.4 Settings of the IO-Link devices

To find the basic settings of the devices for each port, proceed as follows.

1. right-click on the port to open the context menu and select “Settings”.

⇨ A new tab “Portx:: Settings” opens where the settings described below can be made.

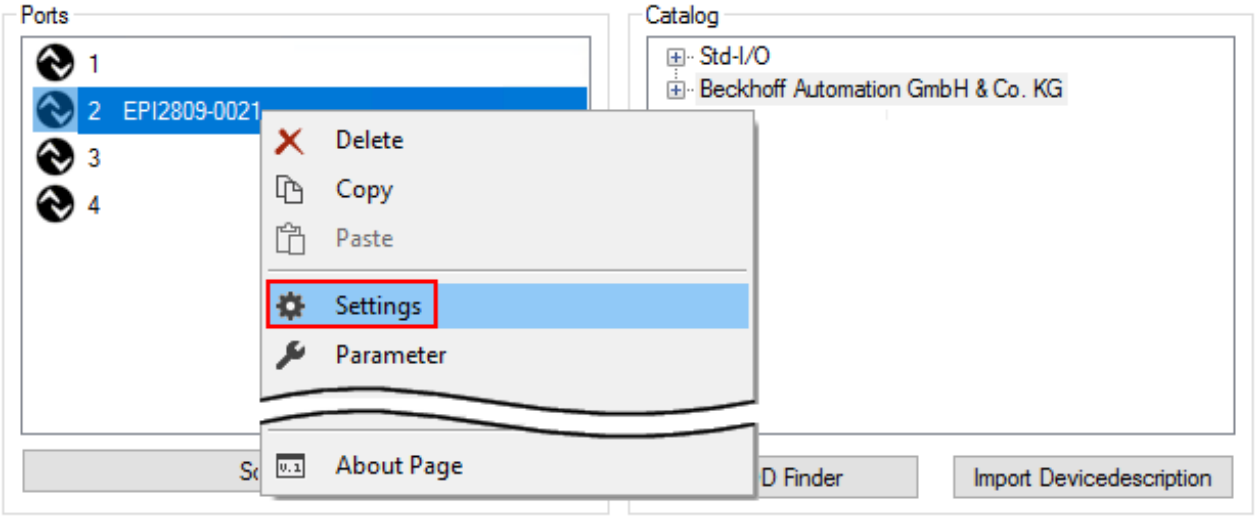


Fig. 22: Context menu - Settings

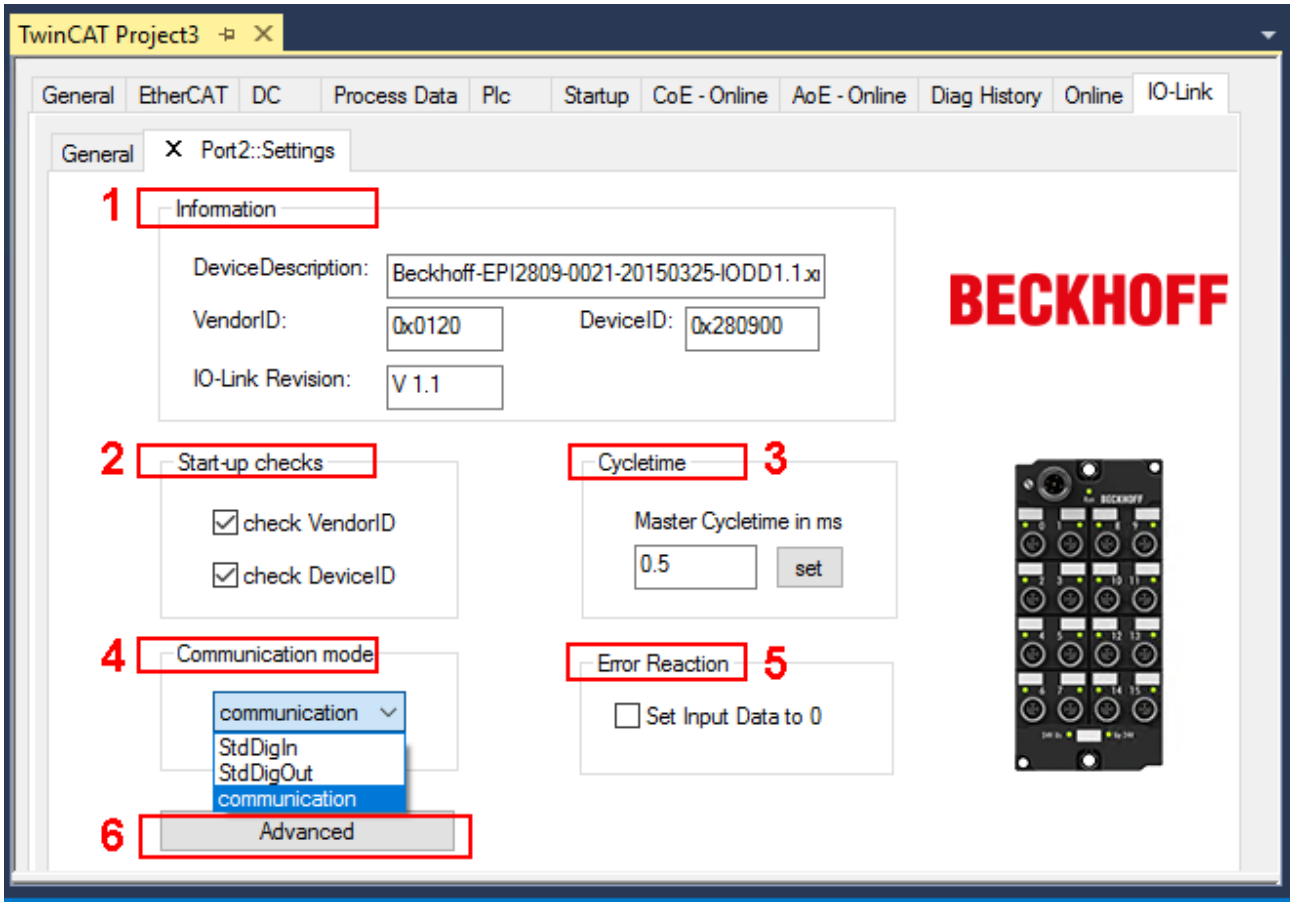
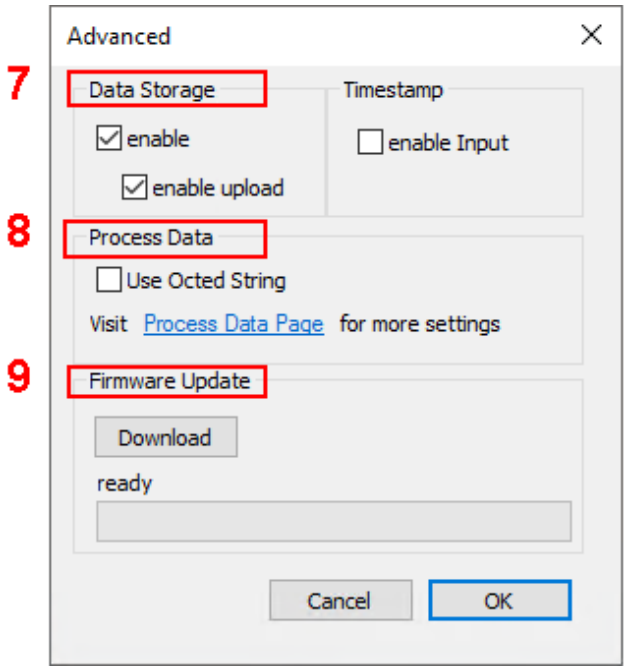


Fig. 23: Settings of the IO-Link devices

1. **Information**
   This field is for information only; the IODD that was read in is displayed under Device Description. Furthermore, the VendorID, DeviceID and the IO-Link revision (V1.0 or V1.1) of the IO-Link devices are displayed. If the device is an IO-Link device V1.1, then the parameter server [▶ 16] function is supported.

The following settings can be made in the settings for the IO-Link devices (see figure above):

2. **Start-up checks**
   This parameter can be used to specify that the Vendor ID and Device ID should be checked when the IO-Link device starts up.
   ⇨ This avoids errors when exchanging IO-Link devices.
3. **CycleTime**
   Specifies the cycle time for the IO-Link master
4. **Communication mode**
   Selection of the mode in which the IO-Link port is to be operated.
   ⇨ “Communication”: Default mode for IO-Link devices
   ⇨ “StdDigIn / StdDigOut”: Mode for non-IO-Link devices, automatically selected if the port is configured as a digital input or output [▶ 34].
5. **Error Reaction**
   If the “Set Input Data to 0” field is activated:
   ⇨ input data are set to 0 in case of error
   ⇨ Status display: “Error”
6. **Button "Advanced"**
7. **Data Storage**
   Pay attention to the sensor version:
   ⇨ V1.0 -> data storage is not supported
   ⇨ V1.1 -> data are stored in the parameter server (preset)
8. **Process Data Format**
   Adaptation of the process data format
   If Field “Use Octet String” is selected
   ⇨ complex data types (process data) are created as octet strings.
   Advantage: simple further processing in the PLC
9. **Firmware Update of the Beckhoff IO-Link devices**
   For a firmware update use the “Download” button.
   Observe the description in the documentation of EPIxxxx boxes in chapter Firmware Update des IO-Link Devices.



6.5 EPIxxxx, ERIxxxx - Setting of the IO-Link device parameters

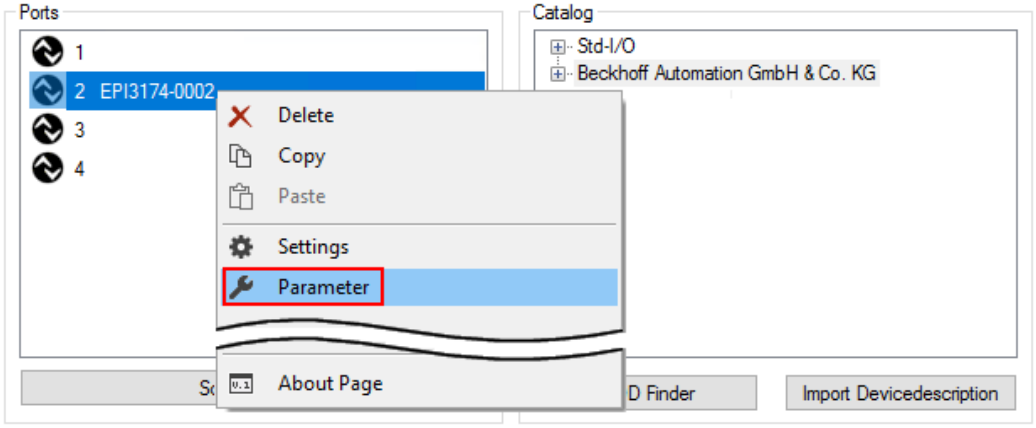
This chapter explains how to read out and set the IO-Link device parameters.

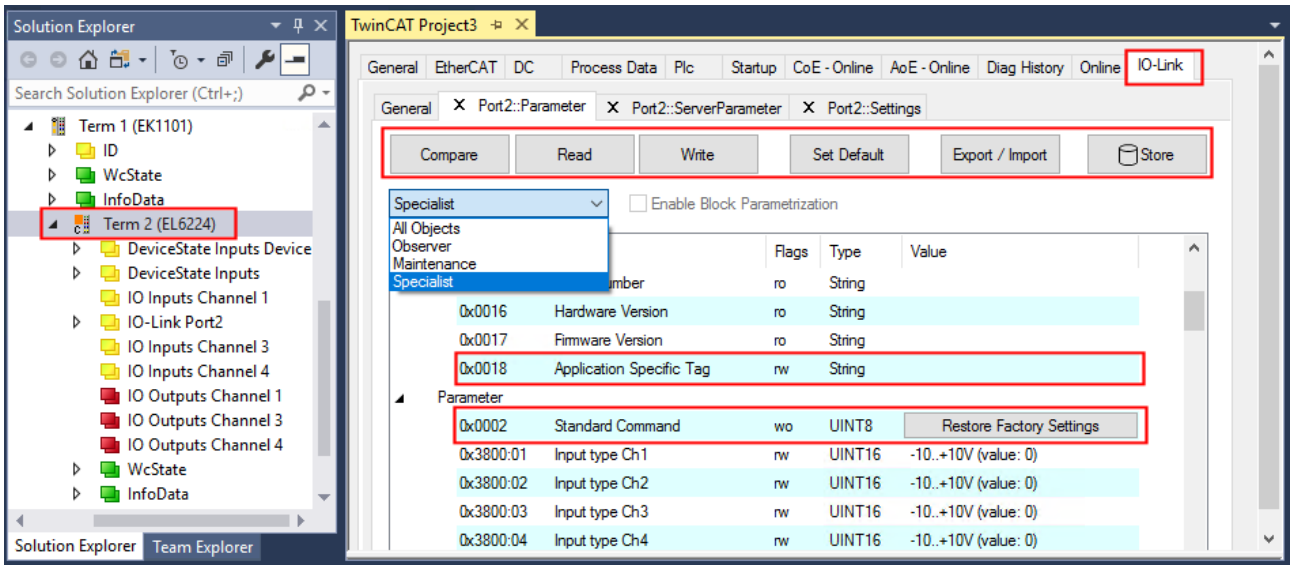
The number and type of the objects shown on the “Parameters” tab vary according to the type of sensor. The default settings as stored in the IODD can initially be seen.

To open the “Parameter” tab

1. Click the IO-Link master in the TwinCAT tree structure.
2. Click the “IO-Link” tab.
3. Select the port to which the IO-Link device is connected,
4. Double-click or by right-click to the port and select “Parameter”.

⇨ The “Parameter” tab is opened.





The device parameters are listed in the tab. The buttons Compare [▶ 45], Read, Write [▶ 47], Set Default [▶ 48], Export/Import [▶ 49] and Store [▶ 50] are located at the top of the tab. The “Read”, “Write” and “Store” buttons are used to read out the parameters stored in the IO-Link device, load them and store them in the parameter server of the master.

Different user roles can be selected from the drop-down menu. The default user role is “Specialist”. The parameters are displayed in different representations and scopes.

Restarting the IO link device or restoring of the application parameters is possible via the parameter Standard Command [▶ 53].

Application specific information can be specified in parameter (0x0018) Application Specific Tag [▶ 54].

“Compare” button

1. Press the “Compare” button.
   ⇨ the parameter data of the configuration are compared with the parameter sets in the sensor.

⇨ The result is displayed in the “Parameter” tab see following figures.

Conformity of configuration and sensor data

The match is confirmed by a green tick in front of the index. Matching values are displayed in the “Value” field (see index 0x0018 “Application Specific Tag”).

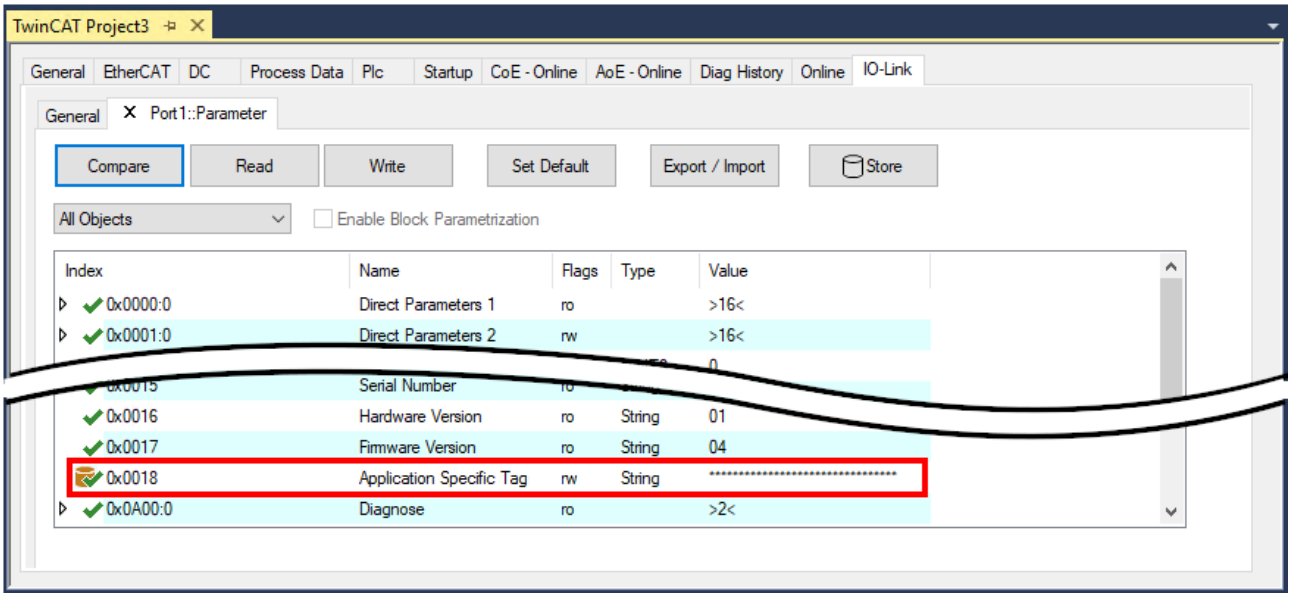


Fig. 24: Display of matching data in the “Parameter” tab

Deviations between configuration and sensor data

Deviations are indicated by a pen-symbol in front of the index. If there are different values in the “Value” field, the value “Compare” is displayed (see Index 0x0018 “Application Specific Tag”).

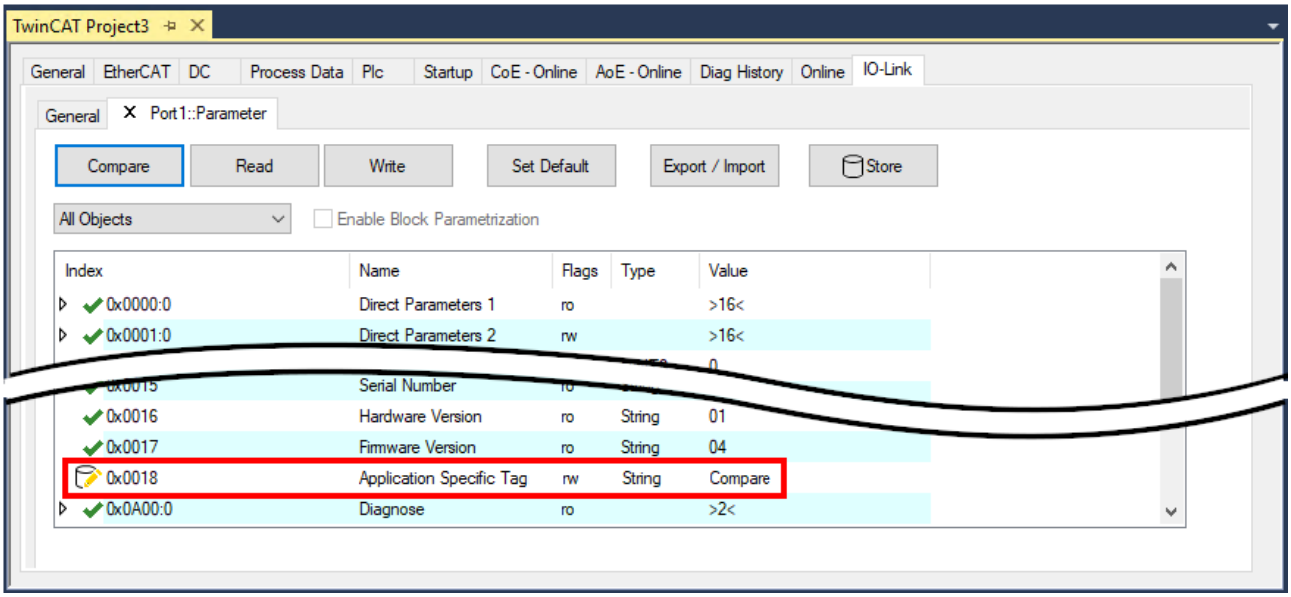


Fig. 25: Display of deviating data in the “Parameter” tab

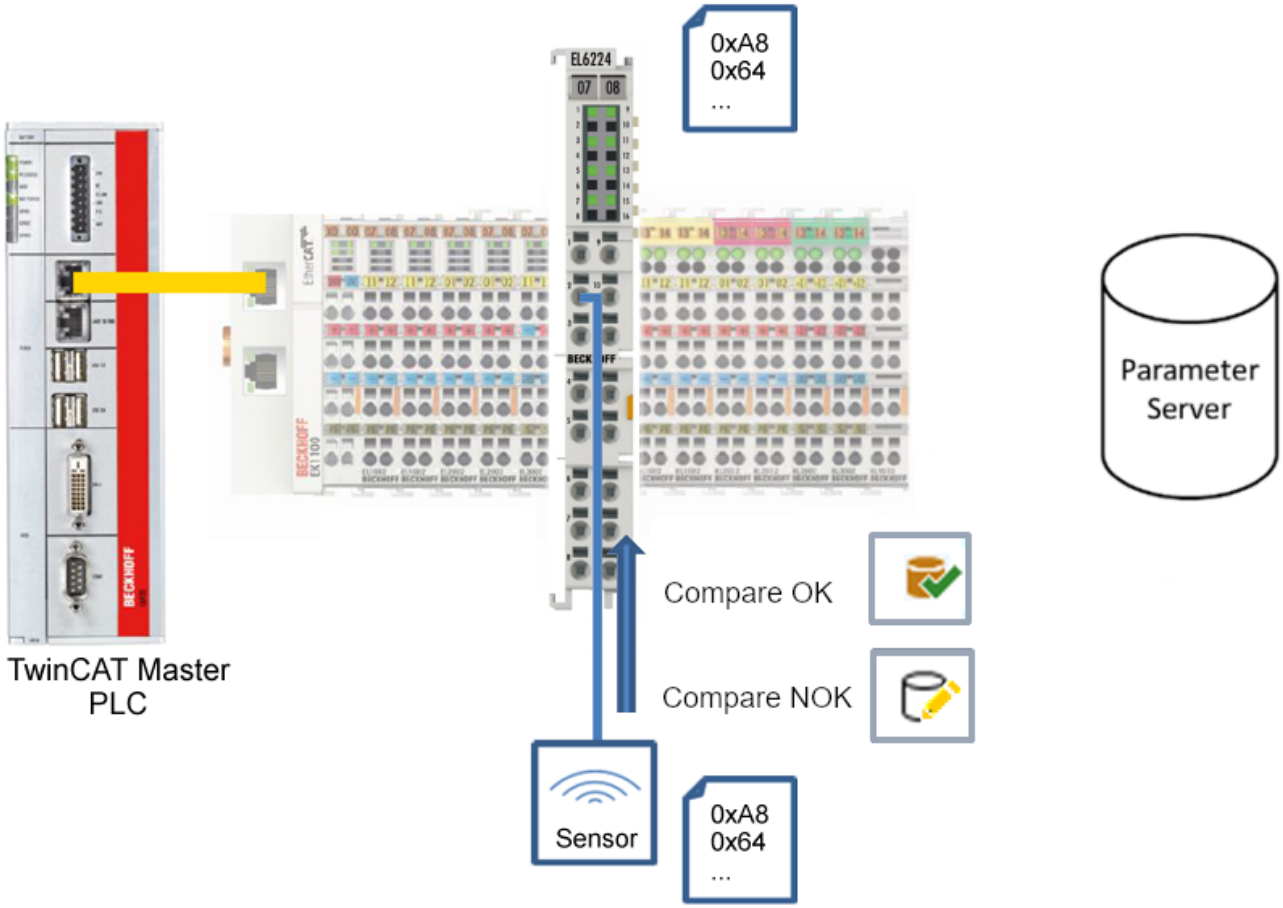


Fig. 26: Compare configuration and sensor data

“Read” button

The default values from the IODD file are always preset

1. Press the “Read” button

⇨ The current parameter values of the sensor are read. The successful reading of the data is confirmed with a green tick in front of the index.

“Write” button

The default values from the IODD file are always preset

1. Enter the desired value under “Value”.
2. Press the Enter key.
   ⇨ The values are accepted.
3. Press the “Write” button.

⇨ The data is written to the device (offline configuration is possible). The successful writing process is confirmed via a storing symbol in front of the index.

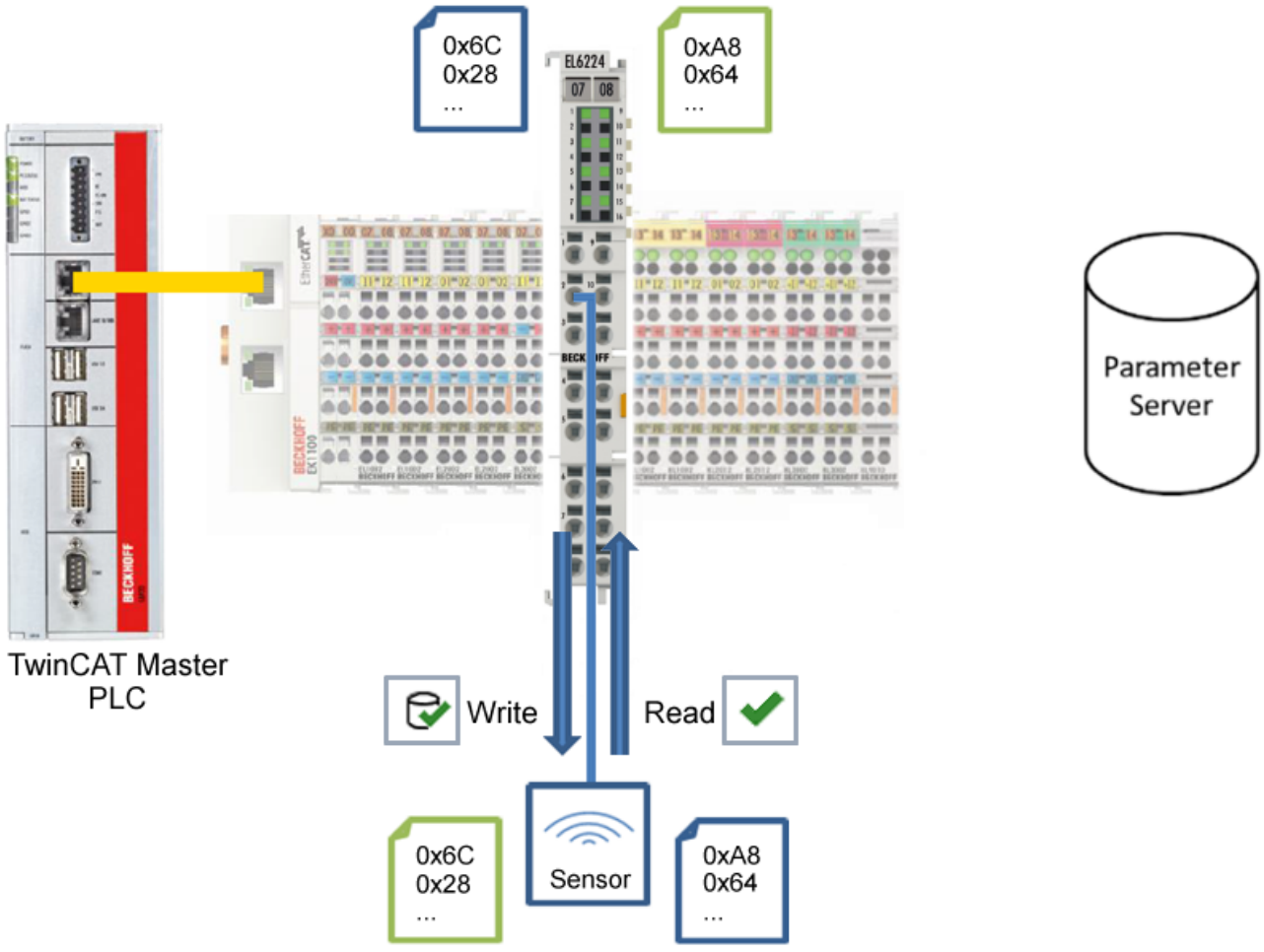


Fig. 27: Write parameter data to the sensor, read parameter data from the sensor.

“Set Default” button

1. Press the “Set Default” button.

⇨ All parameter values are set to the default settings.

Write default-values to the sensor

Note that the default-values must also be written to the device via the “Write” button.

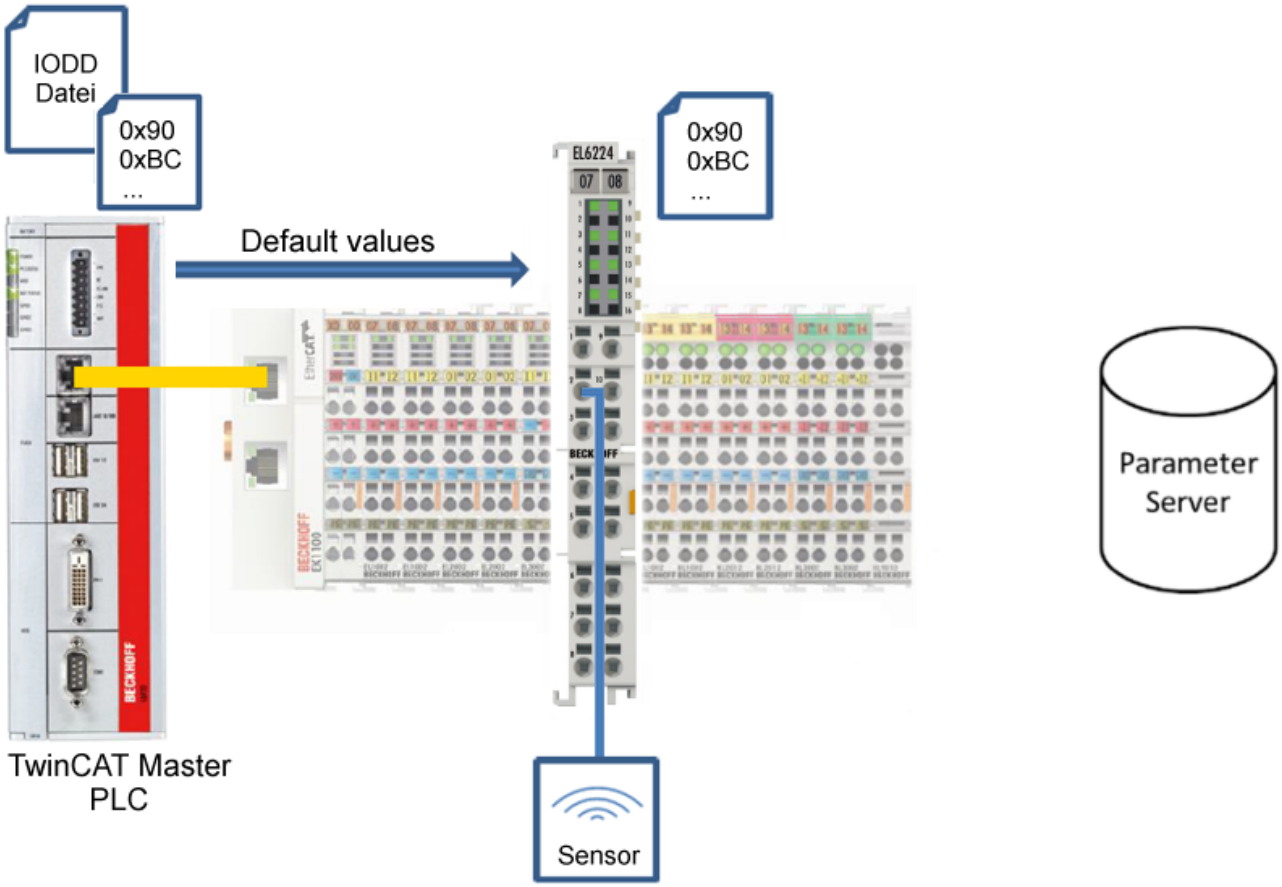


Fig. 28: Reset parameter values to default

“Export / Import” button

The set parameter values can be exported as a .vbs file and restored later via Import.

1. Press the “Export / Import” button (see the diagram below (1)).
   ⇨ The Import / Export dialog is opened
2. Specify the path under which you want to export or import the .vbs file, see fig. (2) below and confirm with the “Open” button, see fig. (4) below.
3. In addition, the export options “Attach Store Command” and “Enable Block Parametrization” can be selected as shown in fig. (3) below.
   ⇨ “Attach Store Command”: The parameters are loaded into the parameter server after the script has written all values.
   ⇨ “Enable Block Parametrization”: Block parameterization is enabled. For some sensors, writing is only possible when block parameterization is enabled.
4. Press the “Export” or “Import” button
   ⇨ The parameters are adopted from the imported file. The change of parameters is marked with a pencil symbol.
5. Write the new parameter values to the sensor via “Write” button.

⇨ The data is written to the device (offline configuration is possible). The successful writing process is confirmed via a storing symbol in front of the index.

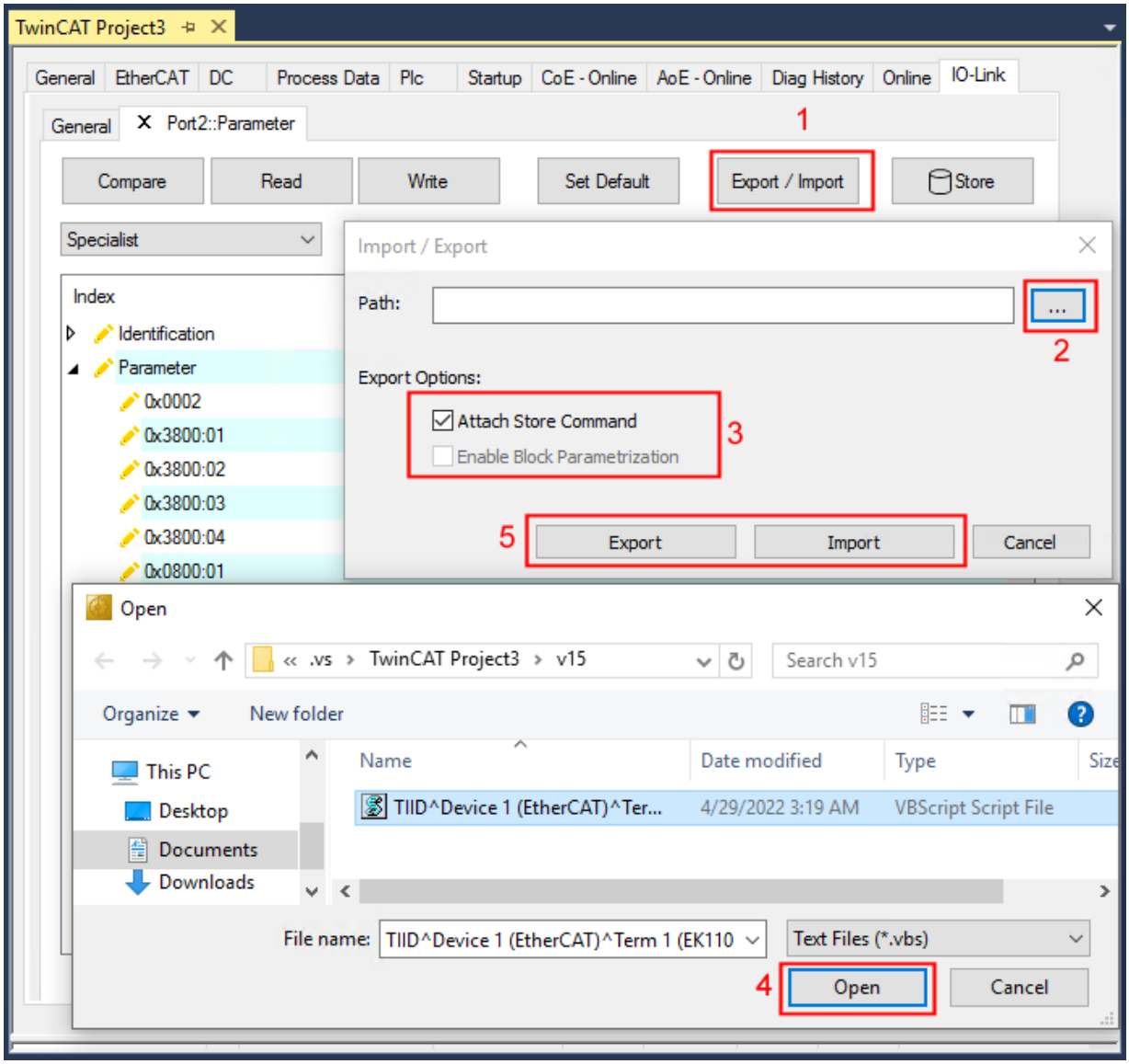


Fig. 29: Parameterization IO-Link device - Export / Import

“Store” button

1. Click “Store” (data storage):
   ⇨ The Beckhoff IO-Link master stores sensor-dependent-data, e. g. the following parameters
   (0x0018) “Application-Specific Tag”,
   (0x08n0) “Settings” and
   0x3800 “Range Settings”.
   The success of storing process is marked with the storing symbol.
   ⇨ If the IO-Link device is exchanged for a similar module, the device can be restored.

The stored values are displayed in the “ServerParameter” tab

2. Right-click on the device and select “Parameter Server” from the menu.

⇨ The stored values are displayed.

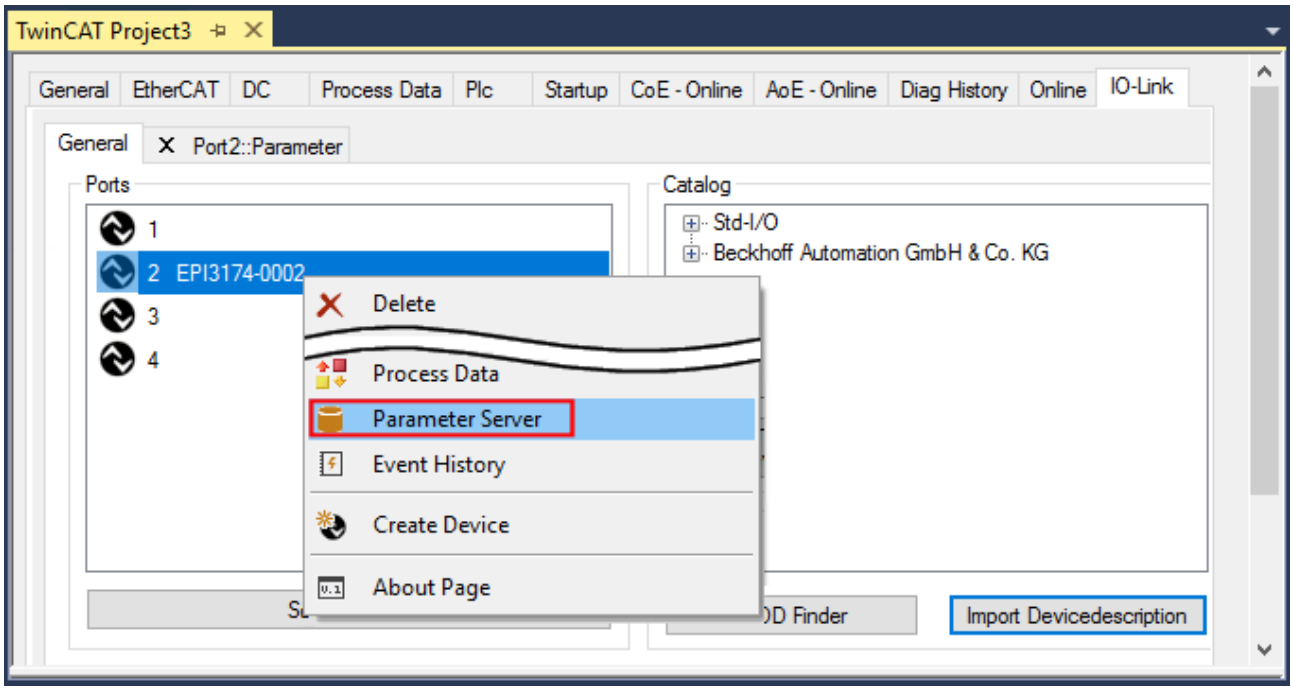


Fig. 30: Open the “ServerParameter” tab

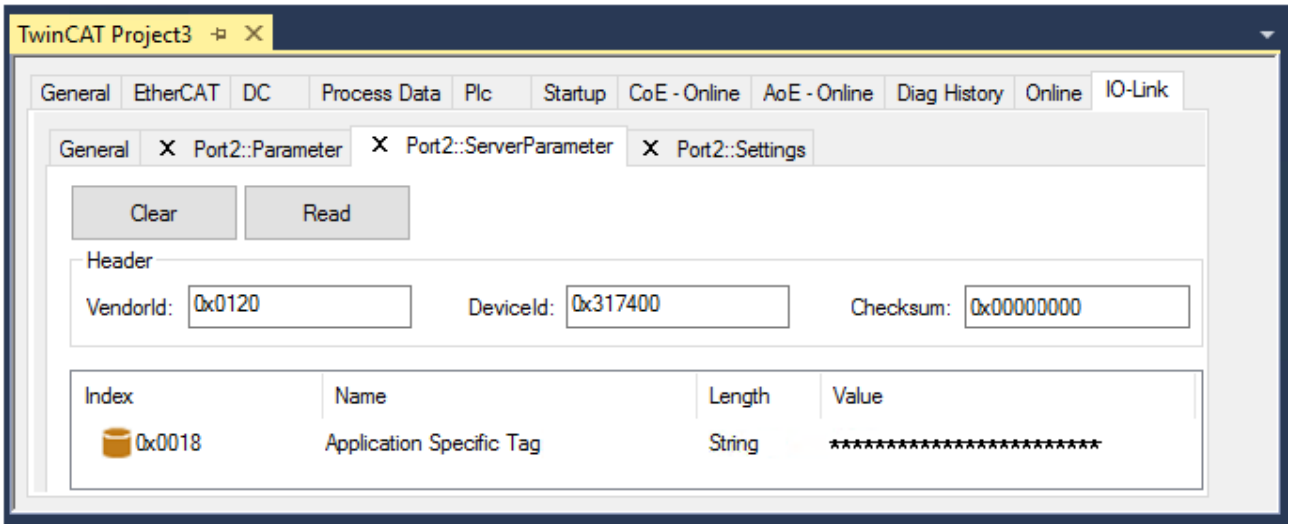


Fig. 31: „ServerParameter“ tab

Activate store button via PLC

As for CoE, the Indexgroup of an ADS command is specified as **0xF302** for the IO link data channel.

According to the IO-Link specification devices with ISDU support shall use index **0x0002** to receive the SystemCommand. The following list displays coding examples for system commands (ISDU), the complete table “Coding of SystemCommand (ISDU)” can be found in the IO-Link specification.

Command (hex)	Command (dec)	Name of the command	Definition
….			
0x01	1	ParamUploadStart	Start Parameter Upload
0x02	2	ParamUploadEnd	Stop Parameter Upload
0x03	3	ParamDownloadStart	Start Parameter Download
0x04	4	ParamDownloadEnd	Stop Parameter Download
0x05	**5**	**ParamDownloadStore**	**Finalize parameterization and start Data Storage**
0x06	6	ParamBreak	Cancel all Param commands
….			

Use an ADS Write function block for activating the store-function via the plc. The following figure shows a sample code for activation of the store-Button (command 0x05 “ParamDownloadStore”)

```
Case_Write:
    AdsWrite_EL6224( WRITE := FALSE );
    AdsWrite_EL6224.IDXGRP   := EL6224_Ch_iGrp;
    AdsWrite_EL6224.IDXOFFS  := EL6224_Ch_iOffWri;
    AdsWrite_EL6224.LEN      := SIZEOF(EL6224_bywrite);
    AdsWrite_EL6224.SRCADDR  := ADR(EL6224_bywrite);
    AdsWrite_EL6224( Write := TRUE);
    eSwitch1 := Case_WriBu;

EL6224_AoePortCh   : UINT  := 16#1001;
EL6224_Ch_iGrp     : UDINT := 16#F302;
EL6224_Ch_iOffManu : UDINT := 16#00100000;
EL6224_Ch_iOffPro  : UDINT := 16#00140000;
EL6224_Ch_iOffWri  : UDINT := 16#00020000;
EL6224_sManu       : STRING;
EL6224_sPro        : STRING;
EL6224_bywrite      : BYTE := 16#5;
```

Fig. 32: Sample code for activation of the store-function via the plc

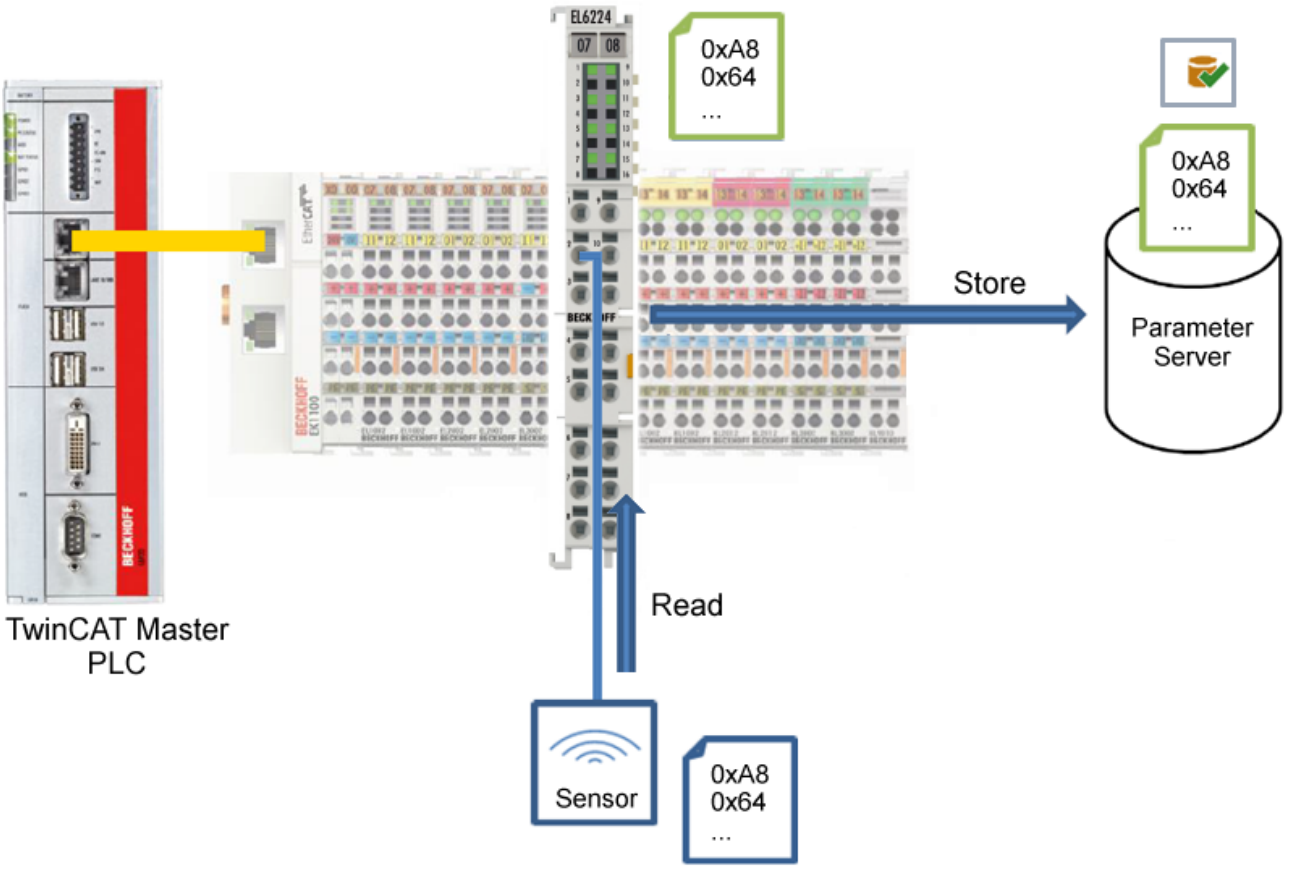


Fig. 33: Store parameters

Standard Command (Index 0x0002)

The IO-Link master writes various IO-Link-specific commands to the “Standard Command” during startup. Some of these commands are available in the TwinCAT interface (see figure below).

1. Click “Standard Command” in the parameter list of the “All Objects” user role, then double-click “Standard Command” in the right-hand field.
2. Select the desired value from the list of different options and
   - “Device Reset”: Restarts the IO-Link device.
   - “Application Reset”: No function.
   - “Restore Factory Settings”: Restoring the application parameters, i.e. the Settings parameter (0x0800).
3. Use the “Write” button (as described above [▶ 47]).

⇨ The data is written to the device (offline configuration is possible). The successful writing process is confirmed via a storing symbol in front of the index.

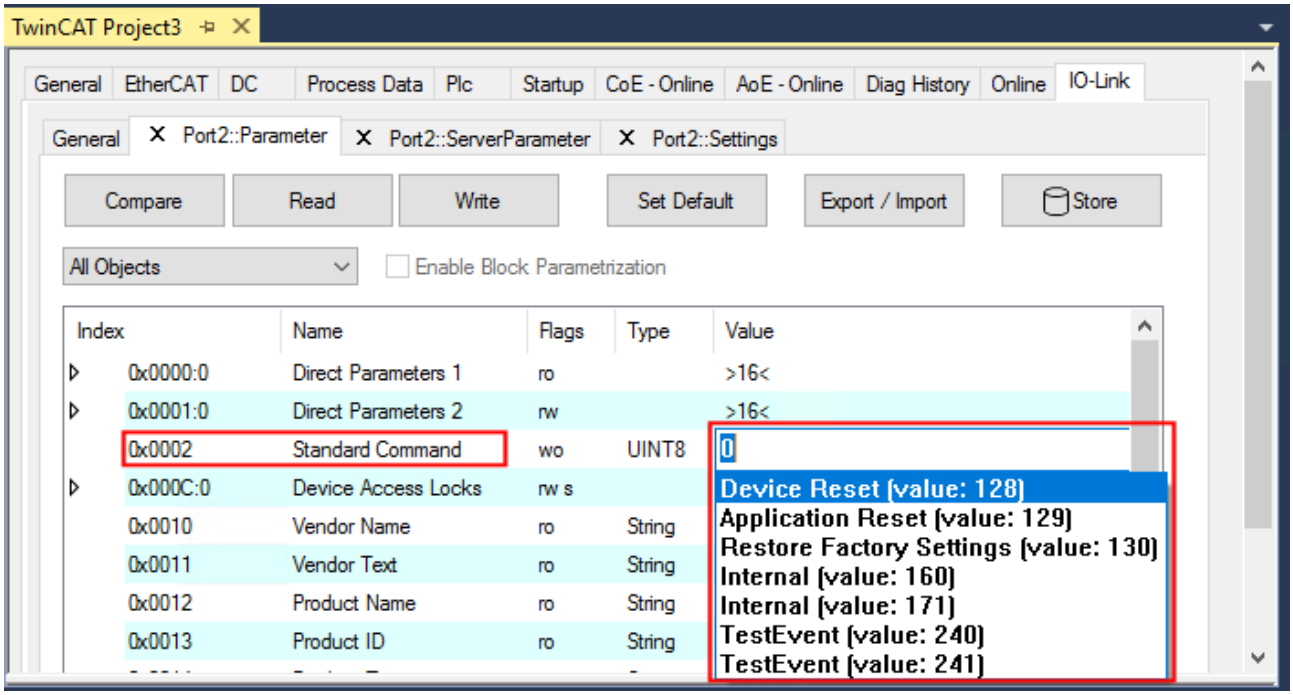


Fig. 34: Parameters IO-Link device “Standard Command”

“Application Specific Tag” (Index 0x0018)

Application-specific information can be entered and stored here.

1. Click “Application-Specific Tag” in the parameter list, then double-click “Application-Specific Tag” in the right-hand field.
2. Enter application-specific information and confirm with the Enter key.
3. Use the Write [▶ 47] button and the Store [▶ 50] button, if required (as described above).

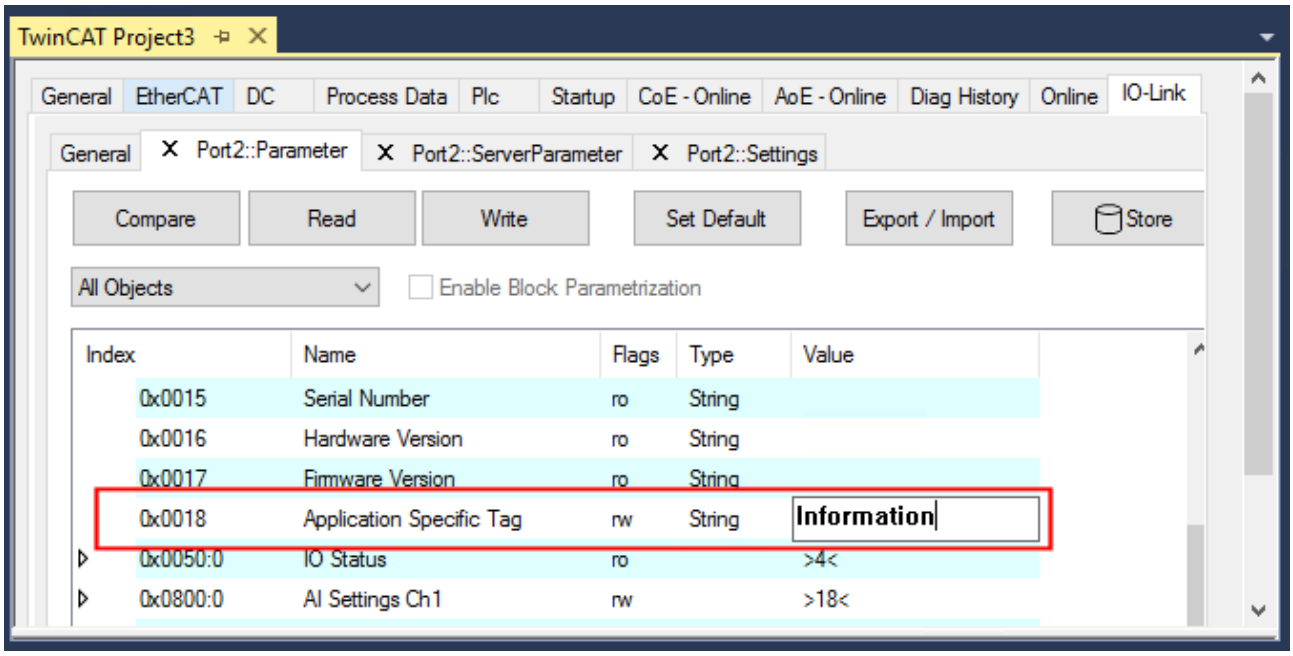


Fig. 35: Parameters IO-Link device: “Application Specific Tag”

6.6 Access to IO-Link data

6.6.1 IO-Link system communication

The EP622x IO-Link master is divided into two services. On the one hand it represents an IO-Link master in relation to the connected IO-Link devices, while on the other it is an EtherCAT slave in relation to the PLC TwinCAT master. The system communication is illustrated in Fig. *Illustration of the system communication of an EtherCAT master.*

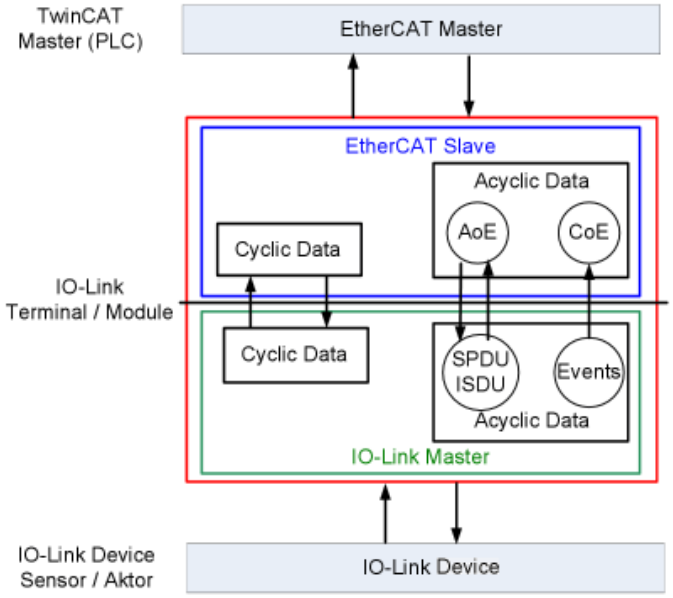


Fig. 36: Illustration of the system communication of an EtherCAT master

In principle cyclic and acyclic data are exchanged. The cyclic process data can be accessed via the PDOs, the acyclic data via AoE [▶ 58]. The events are additionally displayed under Diag History [▶ 61] in the System Manager.

- **Cyclic data:**
  - Process data
  - Value status
- **Acyclic data:**
  - Device data
  - Events

6.6.2 PDO Assignment

The scope of the process data offered varies depending on the configured IO-Link ports.
DeviceState Inputs Device and *DeviceState Inputs* are selected by default. Device-specific PDOs (*0x1A0n Port (n-1) Process Data*) are only displayed after configuring on the respective port and restarting the EtherCAT system or reloading the configuration in Config mode (F4); see Configuration of the IO-Link master.

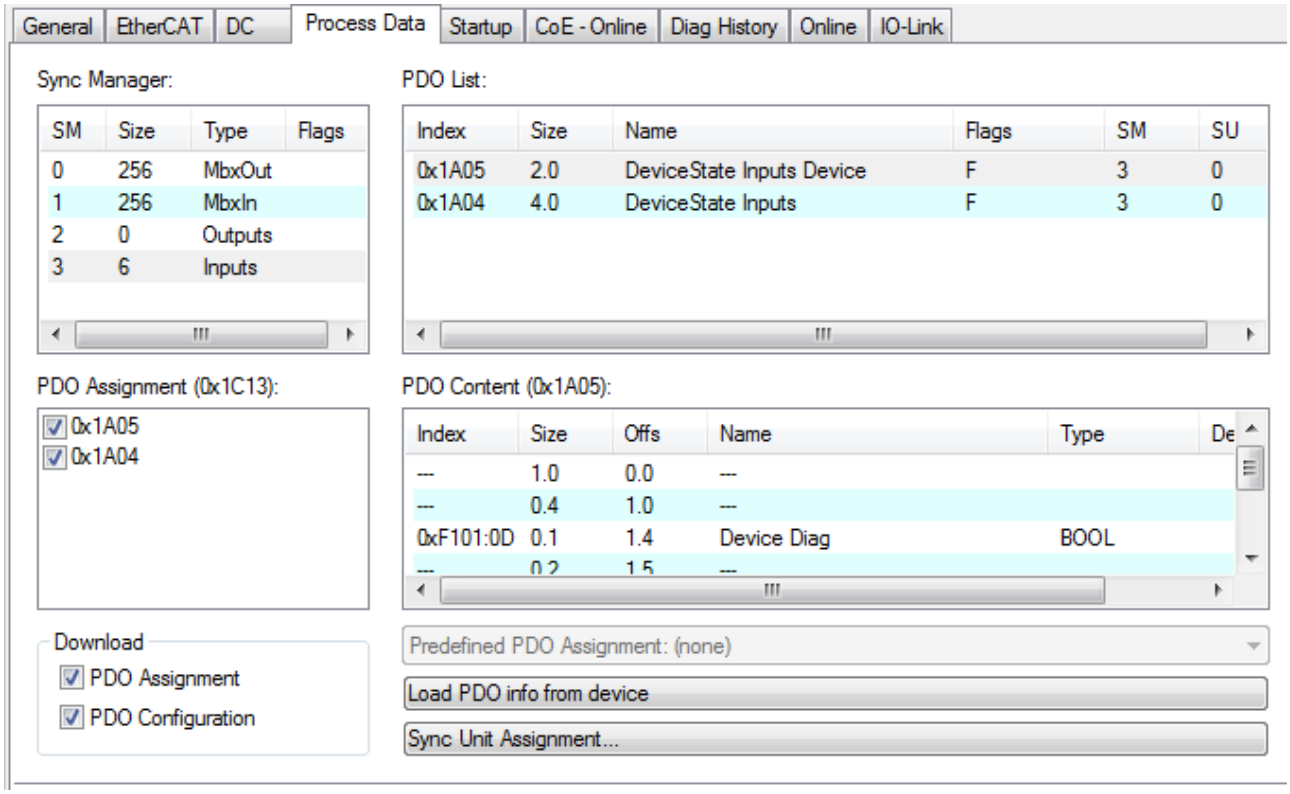


Fig. 37: Illustration of the EP6224-2022 process data allocation, SM3 inputs

SM3, PDO Assignment 0x1C13			
Index	**Size (byte.bit)**	**Name**	**PDO Content**
0x1A05	2.0	DeviceState Inputs Device	Index 0xF101:0D - Device Diag Index 0xF101:10 - Device State
0x1A04	4.0	DeviceState Inputs	Index 0xF100:01 - State Ch1 Index 0xF100:02 - State Ch2 Index 0xF100:03 - State Ch3 Index 0xF100:04 - State Ch4
0x1A00	0.0 - 32.0	Port 1 Process Data	IO-Link device-specific / only active after configuration
0x1A01	0.0 - 32.0	Port 2 Process Data	IO-Link device-specific / only active after configuration
0x1A02	0.0 - 32.0	Port 3 Process Data	IO-Link device-specific / only active after configuration
0x1A03	0.0 - 32.0	Port 4 Process Data	IO-Link device-specific / only active after configuration

Process data representation

If data types are used that don't conform to IEC61131-3, they are represented as octed strings.

The state of the IO-Link ports 1 to 4 is displayed via index 0xF100:0n.
The indices 0xF101:xx provide general diagnosis data.

Index	Size (byte.bit)	Name	Meaning
0xF101:0D	0.1	Device Diag	Occurrence of events (on the slave side) is reported via a status bit
0xF101:10	0.1	Device State	Interruption of communication with one of the slaves is reported via a status bit
0xF100:01	1.0	State Ch.1	0x_0 = Port disabled 0x_1 = Port in std dig in 0x_2 = Port in std dig out 0x_3 = Port in communication OP 0x_4 = Port in communication COMSTOP / dig in Bit (only in the std. IO Mode) 0x_8 = Process Data Invalid Bit 0x1_ = Watchdog detected 0x2_ = internal Error 0x3_ = invalid Device ID 0x4_ = invalid Vendor ID 0x5_ = invalid IO-Link Version 0x6_ = invalid Frame Capability 0x7_ = invalid Cycle Time 0x8_ = invalid PD in length 0x9_ = invalid PD out length 0xA_ = no Device detected 0xB_ = error PreOP/Data storage (error in IO-Link State PreOP)
0xF100:02	1.0	State Ch.2	
0xF100:03	1.0	State Ch.3	
0xF100:04	1.0	State Ch.4	

6.6.3 Accessing IO-Link parameters

The exchange of the acyclic data takes place via a specified index and subindex range that is device-specific and can be read about in the corresponding vendor documentation.

6.6.4 Parameter data exchange

An intelligent IO-Link sensor/actuator can support parameterization by ISDUs (Indexed Service Data Unit). The PLC must explicitly query or, when marked as such, send these acyclic service data.

ISDU access

TwinCAT supports access via ADS and via the CoE directory.

The respective parameter is addressed via the so-called ISDU index. The following ranges are available:

Designation	Index range
System	0x00..0x0F
Identification	0x10..0x1F
Diagnostic	0x20..0x2F
Communication	0x30..0x3F
Preferred Index	0x40..0xFE
Extended Index	0x0100..0x3FFF
	The range 0x4000 to 0xFFFF is reserved

The use and implementation of these ranges is the responsibility of the sensor/actuator manufacturer. For clarification, just a few of the possible indices are listed here. Please take a look at the relevant chapter “Object description and parameterization”.

Index	Name
0010	Vendor Name
0011	Vendor Text
0012	Product Name
0013	Product ID
0015	Serial Number
0016	Hardware Revision
0017	Firmware Revision
...	...

IO-Link operating modes

The IO-Link ports on the IO-Link master can be operated in the following nine modes (see Object description and parameterization - IO-Link State, Index 0xA0n0:01):

- INACTIVE: Statemachine is inaktive
- DIGINPUT: The port behaves like a digital input
- DIGOUTPUT: The port behaves like a digital output
- ESTABLISHCOMM: The IO-Link wakeup sequence is running
- INITMASTER: Readout the IO-Link device and check the communication parameters
- INITDEVICE: Initialization of the IO-Link device
- PREOPERATE: Parameter server is running
- OPERATE: The port is used for IO-Link communication
- STOP: Communication is stopped (COM-stop)

6.6.5 ADS

Communication relating to IO link demand data is initiated via an ADS command. An ADS address always consists of a NetID and PortNo. TwinCAT relays ADS commands to the EP622x via AoE (*A*DS *o*ver *E*therCAT), from where the command is relayed to the IO link master section and therefore the data channel.

AoE NetID

The EP622x is assigned a dedicated AoE-NetID for communication with the IO-Link master section. This is assigned by the configuration tool (Fig. *Assignment AoE-NetID*).
NetID at *EP622x -> "EtherCAT"-> "Advanced Settings" -> "Mailbox" -> "AoE"*

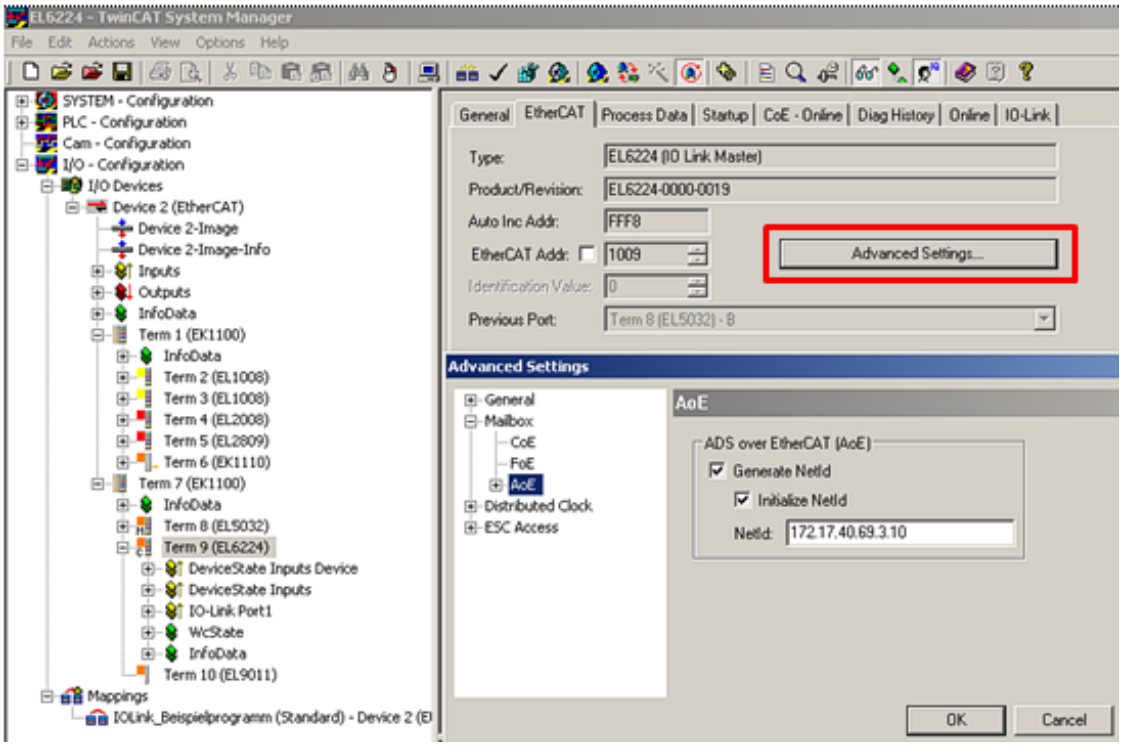


Fig. 38: AoE-NetID allocation

PortNo

The individual IO-Link ports for the master are allocated via the port number. The port numbers are allocated in ascending order from 0x1000. I.e. IO-Link Port1 corresponds to PortNo 0x1000 and IO-Link Portn corresponds to PortNo 0x1000 + n-1.
The following specification applies for the EP6224 (4-port IO-Link master):

- IO-Link Port1 corresponds to PortNo 0x1000
- IO-Link Port2 corresponds to PortNo 0x1001
- IO-Link Port3 corresponds to PortNo 0x1002
- IO-Link Port4 corresponds to PortNo 0x1003

ADS Indexgroup

As for CoE, the Indexgroup of an ADS command is specified as 0xF302 for the IO link data channel.

ADS Indexoffset

The IO link addressing with index and subindex is coded in the Indexoffset. The Indexoffset has a size of 4 bytes and is subdivided as follows: 2-byte index, 1-byte reserve, 1-byte subindex.

- Example: Indexoffset 0x12340056 corresponds to index 0x1234 and subindex 56

6.6.6 Access to events

Some of the IO-Link sensors forward events that occur to the master. These events may be items of information, warnings or error messages, e.g. short circuit or overheating.
The IO-Link master reports these events by setting the Device Diag bit. Further information on the events can be read via the CoE directory or the DiagHistory tab.

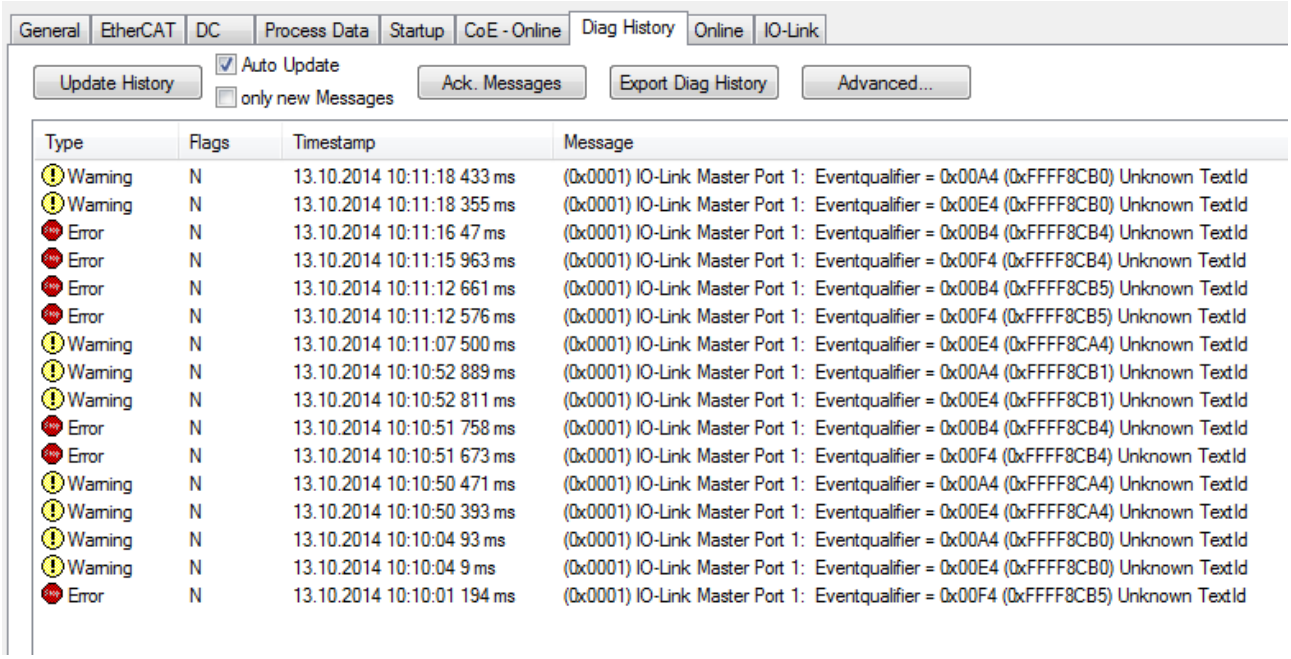


Fig. 39: DiagHistory tab

The events are arranged according to type (information, warning, error), flag, occurrence of the event (time stamp) and message (port number & event code).
The meaning of the individual messages can be taken from the vendor documentation. The IO-Link device can be directly allocated on the basis of the port number. The events occurring can be managed using the various buttons.

- **Update History:** if the "Auto Update" field is not selected, then the current events can be displayed via the "Update History" button
- **Auto Update:** if this field is selected, then the list of events occurring is automatically updated
- **Only new Messages:** if this field is selected, then only those messages that have not yet been confirmed are displayed.
- **Ack. Messages:** an event that occurs is reported via the Device Diag. Confirming the message will reset the bit to 0.
- **Export Diag History:** the events that have occurred can be exported as a "txt" file and thus archived.
- **Advanced:** this field currently (as at 3rd quarter 2015) has no function.

6.6.7 PLC library: Tc3_IoLink

The PLC library “Tc3_IoLink” is used for communication with IoLink devices.

Function blocks are available for this purpose that support the "Common Profile" and "Smart Sensor Profile" and enable parameters to be read and written.

See software documentation in the Beckhoff Information System:

TwinCAT 3 | PLC Library: Tc3_IoLink

6.7 Restoring the delivery state

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

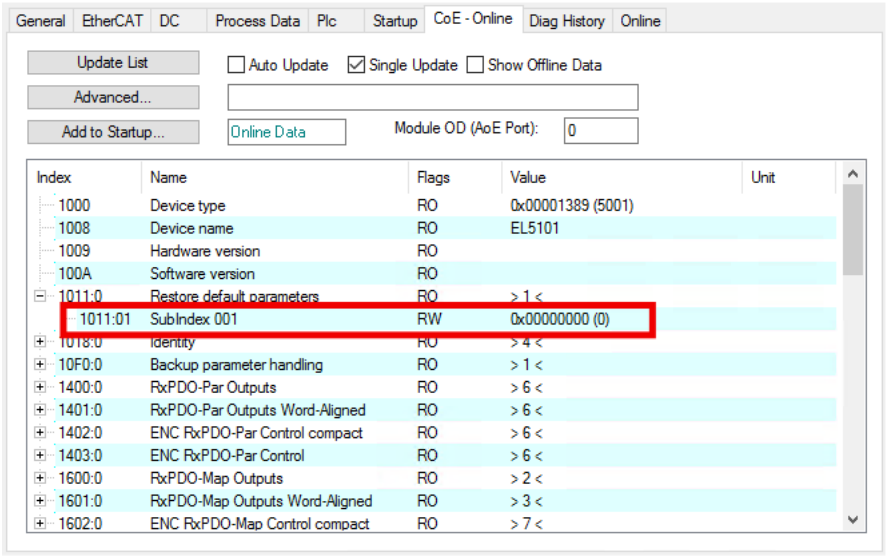


Fig. 40: Selecting the Restore default parameters PDO

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.

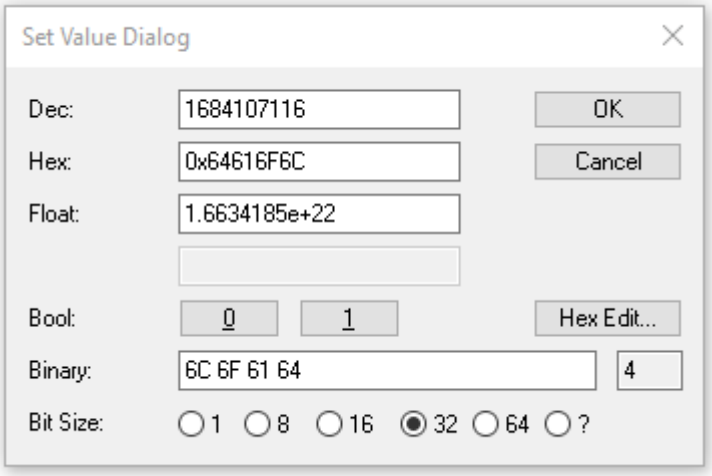


Fig. 41: Entering a restore value in the Set Value dialog

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.

6.8 Decommissioning

⚠ WARNING

Risk of electric shock!

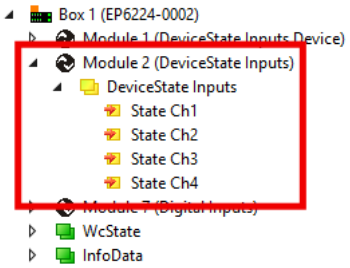
Bring the bus system into a safe, de-energized state before starting disassembly of the devices!

7 Diagnosis

7.1 Status of the IO-Link ports

There is a status byte for each IO-Link port. You can find the status bytes at the following locations:

- In the process data object "Module 2 (DeviceState Inputs)"



(example for EP6224-0002)

- In the CoE object F100

7.1.1 Interpretation of the status bytes

The status bytes are divided into two nibbles.

Low nibble

Value	Meaning
0x_0	Port disabled
0x_1	Port in std dig in
0x_2	Port in std dig out
0x_3	Port in communication OP
0x_4	Port in communication COMSTOP / dig in Bit (only in std. IO Mode)
0x_5	not defined
0x_6	not defined
0x_7	not defined
0x_8	Process Data Invalid Bit

Combinations are possible and are output as an addition of the values (see note below).

Higher nibble

Value	Meaning
0x1_	Watchdog detected
0x2_	internal Error
0x3_	invalid Device ID
0x4_	invalid Vendor ID
0x5_	invalid IO-Link version
0x6_	invalid Frame Capability
0x7_	invalid Cycle Time
0x8_	invalid PD in length
0x9_	invalid PD out length
0xA_	no Device detected
0xB_	error PreOP/Data storage

Combinations are possible and are output as an addition of the values (see note below).

Addition of the values in case of simultaneously occurring diagnosis messages

If diagnosis messages occur simultaneously, the value is output as a sum in the status byte of the relevant channel.

- Often 0x03 "Port in communication OP" and 0x08 "Process Data Invalid Bit" occur simultaneously:
  0x03 + 0x08 = 0x0B (11_{dec})

⇨ The value 0x0B (11_{dec}) is output in the status byte.

7.2 ADS Error Codes

Error codes are generated in the event of an error during ADS access to an IO-Link device.

The possible error codes are listed in tables C.1 and C.2.

Example of an AdsReturnCode

AdsReturnCode 0x**80110700**

- **80**: Device Application Error (IO-Link Spec),
- **11**: Index not Available (IO-Link Spec),
- **0700**: General ADS Error

ErrorTypes (IO-Link Spec)

Incident	Error Code	Additional Code	Name	Definition
Device application error – no details	0x80	0x00	APP_DEV	This ErrorType shall be used if the requested service has been refused by the Device application and no detailed
Index not available	0x80	0x11	IDX_NOTAVAIL	This ErrorType shall be used whenever a read or write access occurs to a not existing Index.
Subindex not available	0x80	0x12	SUBIDX_NOTAVAIL	This ErrorType shall be used whenever a read or write access occurs to a not existing Subindex.
Service temporarily not available	0x80	0x20	SERV_NOTAVAIL	This ErrorType shall be used if a parameter is not accessible for a read or write service due to the current state of the Device application.
Service temporarily not available – local control	0x80	0x21	SERV_NOTAVAIL_LOCCTRL	This ErrorType shall be used if a parameter is not accessible for a read or write service due to an ongoing local operation at the Device (for example operation or parameterization via an on-board Device control panel).
Service temporarily not available – Device control	0x80	0x22	SERV_NOTAVAIL_DEVCTRL	This ErrorType shall be used if a read or write service is not accessible due to a remote triggered state of the device application (for example parameterization during a remote triggered teach-in operation or calibration).
Access denied	0x80	0x23	IDX_NOT_WRITEABLE	This ErrorType shall be used if a write service tries to access a read-only parameter.
Parameter value out of range	0x80	0x30	PAR_VALOUTOFRNG	This ErrorType shall be used for a write service to a parameter outside its permitted range of values.
Parameter value above limit	0x80	0x31	PAR_VALGTLIM	This ErrorType shall be used for a write service to a parameter above its specified value range.
Parameter value below limit	0x80	0x32	PAR_VALLTLIM	This ErrorType shall be used for a write service to a parameter below its specified value range.
Parameter length overrun	0x80	0x33	VAL_LENOVRRUN	This ErrorType shall be used when the content of a write service to a parameter is greater than the parameter specified length. This ErrorType shall also be used, if a data object is too large to be processed by the Device application (for example ISDU buffer restriction).
Parameter length underrun	0x80	0x34	VAL_LENUNDRUN	This ErrorType shall be used when the content of a write service to a parameter is less than the parameter specified length (for example write access of an Unsigned16 value to an Unsigned32 parameter).
Function not available	0x80	0x35	FUNC_NOTAVAIL	This ErrorType shall be used for a write service with a command value not supported by the Device application (for example a SystemCommand with a value not implemented).
Function temporarily unavailable	0x80	0x36	FUNC_UNAVAILTEMP	This ErrorType shall be used for a write service with a command value calling a Device function not available due to the current state of the Device application (for example a SystemCommand).
Invalid parameter set	0x80	0x40	PAR_SETINVALID	This ErrorType shall be used if values sent via single parameter transfer are not consistent with other actual parameter settings (for example overlapping set points for a binary data setting
Inconsistent parameter set	0x80	0x41	PAR_SETINCONSIST	This ErrorType shall be used at the termination of a block parameter transfer with ParamDownloadEnd or ParamDownloadStore if the plausibility check shows inconsistencies
Application not ready	0x80	0x82	APP_DEVNOTRDY	This ErrorType shall be used if a read or write service is refused due to a temporarily unavailable application (for example peripheral controllers during startup).
Vendor specific	0x81	0x00	UNSPECIFIC	This ErrorType will be propagated directly to higher level processing elements as an error (no warning) by the Master.
Vendor specific	0x81	0x01 to 0xFF	VENDOR_SPECIFIC	

Table C.1 ErrorTypes, IO-Link Spec

Derived ErrorTypes (IO-Link Spec)

Incident	Error Code	Additional Code	Name	Definition
Master – Communication error	0x10	0x00	COM_ERR	The Master generates a negative service response with this ErrorType if a communication error occurred during a read or write service, for example the SDCI connection is interrupted.
Master – ISDU timeout	0x11	0x00	I-SERVICE_TIMEOUT	The Master generates a negative service response with this ErrorType, if a Read or Write service is pending longer than the specified I-Service timeout in the Master.
Device Event – ISDU error (DL, Error, single shot, 0x5600)	0x11	0x00	I-SERVICE_TIMEOUT	If the Master received an Event with the EventQualifier and the EventCode 0x5600, a negative service response indicating a service timeout is generated and returned to the requester (Master – ISDU timeout).
Device Event – ISDU illegal service primitive (AL, Error, single shot, 0x5800)	0x11	0x00	I-SERVICE_TIMEOUT	If the Master received an Event with the EventQualifier and the EventCode 0x5800, a negative service response indicating a service timeout is generated and returned to the requester (Master – ISDU timeout).
Master – ISDU checksum error	0x56	0x00	M_ ISDU_CHECKSUM	The Master generates a negative service response with this ErrorType, if its data link layer detects an ISDU checksum error.
Master – ISDU illegal service primitive	0x57	0x00	M_ ISDU_ILLEGAL	The Master generates a negative service response with this ErrorType, if its data link layer detects an ISDU illegal service primitive.
Device Event – ISDU buffer overflow (DL, Error, single shot, 0x5200)	0x80	0x33	VAL_LENOVRRUN	If the Master received an Event with the EventQualifier and the EventCode 0x5200, a negative service response indicating a parameter length overrun is generated and returned to the requester (see parameter length overrun) Events from legacy Devices shall be redirected in compatibility mode to this derived ErrorType

Table C.2 Derived ErrorTypes, IO-Link Spec

7.3 Further error diagnosis

Device State Inputs Device (0x1A05)

It is indicated in the PDO "Device Diag" (0xF101:0D) that at least one event has occurred in the "Diag History".
"Device State" is the standard status bit for EtherCAT slaves and shows, for example, that communication with one of the slaves has been interrupted.

DeviceState Inputs (0x1A04)

The status of the IO-Link ports is mapped as a status byte in the "DeviceState Inputs" process data object.
See chapter Status of the IO-Link ports [▶ 65].

Nominal/actual comparison of the parameter objects

The indices 0x90n0 (Info data) can be referred to for validation of the configuration indices 0x80n0 of the connected IO-Link device.
In case of error these objects can be used to compare the configuration with the actual state.

Lost Frame Counter

The Lost Frame counter in object 0xA0n0:02 is for the diagnosis of the transmission quality. TwinCAT provides the possibility here to diagnose problems, e. g. with the wiring, EMC or power supply.

8 Appendix

8.1 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 IP67 modules 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

8.2 Accessories

Mounting

Ordering information	Description	Link
ZS5300-0011	Mounting rail	Website

Cables

A complete overview of pre-assembled cables for fieldbus components can be found here.

Ordering information	Description	Link
ZK2000-6xxx-xxxx	Sensor cable M12, 4-pin	Website
ZK2000-7xxx-0xxx	Sensor cable M12, 4-pin + shield	Website
ZK700x-xxxx-xxxx	EtherCAT P cable M8	Website

Labeling material, protective caps

Ordering information	Description
ZS5000-0012	Protective cap for M8 sockets, P-coded, IP67 (50 pieces)
ZS5000-0020	Protective cap for M12 sockets, IP67 (50 pcs.)
ZS5100-0000	Inscription labels, unprinted, 4 strips of 10
ZS5000-xxxx	Printed inscription labels on enquiry

Tools

Ordering information	Description
ZB8801-0000	Torque wrench for plugs, 0.4…1.0 Nm
ZB8801-0001	Torque cable key for M8 / wrench size 9 for ZB8801-0000
ZB8801-0002	Torque cable key for M12 / wrench size 13 for ZB8801-0000
ZB8801-0003	Torque cable key for M12 field assembly / wrench size 18 for ZB8801-0000

Further accessories

Further accessories can be found in the price list for fieldbus components from Beckhoff and online at https://www.beckhoff.com.

8.3 Version identification of EtherCAT devices

8.3.1 General notes on marking

Designation

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

- family key
- type
- version
- revision

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

Notes

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

8.3.2 Version identification of EP/EPI/EPP/ER/ERI boxes

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

Structure of the serial number: **KK YY FF HH**	Example with serial number 12 06 3A 02:
KK - week of production (CW, calendar week)	12 - production week 12
YY - year of production	06 - production year 2006
FF - firmware version	3A - firmware version 3A
HH - hardware version	02 - hardware version 02

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

Syntax: D ww yy x y z u

D - prefix designation
ww - calendar week
yy - year
x - firmware version of the bus PCB
y - hardware version of the bus PCB
z - firmware version of the I/O PCB
u - hardware version of the I/O PCB

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

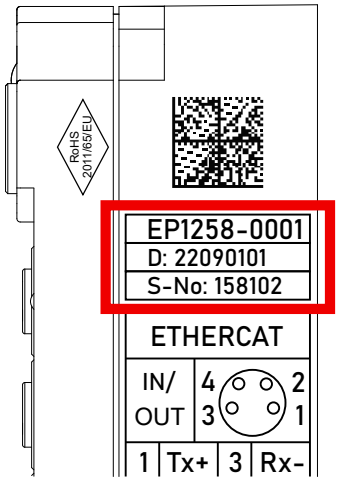


Fig. 42: EP1258-00001 IP67 EtherCAT Box with batch number/DateCode 22090101 and unique serial number 158102

8.3.3 Beckhoff Identification Code (BIC)

The Beckhoff Identification Code (BIC) is increasingly being applied to Beckhoff products to uniquely identify the product. The BIC is represented as a Data Matrix Code (DMC, code scheme ECC200), the content is based on the ANSI standard MH10.8.2-2016.

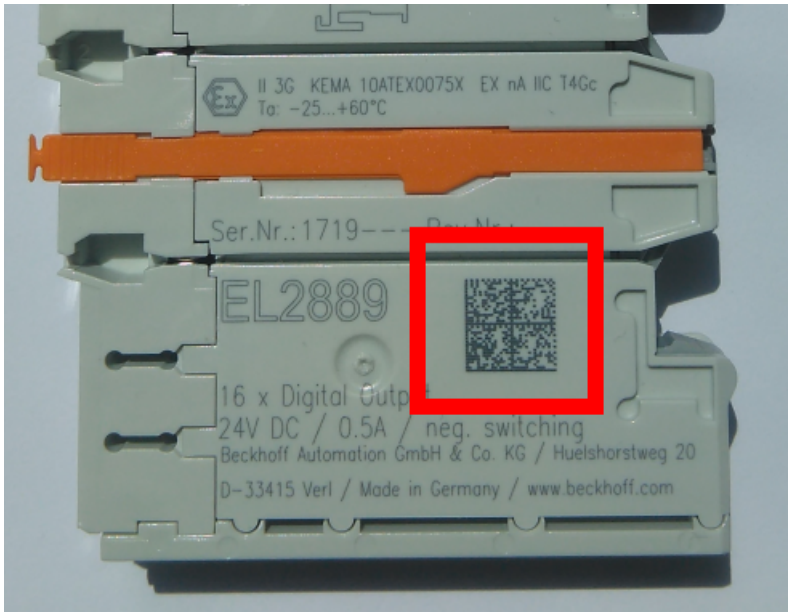


Fig. 43: BIC as data matrix code (DMC, code scheme ECC200)

The BIC will be introduced step by step across all product groups.

Depending on the product, it can be found in the following places:

- on the packaging unit
- directly on the product (if space suffices)
- on the packaging unit and the product

The BIC is machine-readable and contains information that can also be used by the customer for handling and product management.

Each piece of information can be uniquely identified using the so-called data identifier (ANSI MH10.8.2-2016). The data identifier is followed by a character string. Both together have a maximum length according to the table below. If the information is shorter, spaces are added to it.

Following information is possible, positions 1 to 4 are always present, the other according to need of production:

Position	Type of information	Explanation	Data identifier	Number of digits incl. data identifier	Example
1	Beckhoff order number	**Beckhoff order number**	1P	8	1P072222
2	Beckhoff Traceability Number (BTN)	**Unique serial number, see note below**	SBTN	12	SBTNk4p562d7
3	Article description	**Beckhoff article description, e.g. EL1008**	1K	32	1KEL1809
4	Quantity	**Quantity in packaging unit, e.g. 1, 10, etc.**	Q	6	Q1
5	Batch number	Optional: Year and week of production	2P	14	2P401503180016
6	ID/serial number	Optional: Present-day serial number system, e.g. with safety products	51S	12	51S678294
7	Variant number	Optional: Product variant number on the basis of standard products	30P	32	30PF971, 2*K183
...					

Further types of information and data identifiers are used by Beckhoff and serve internal processes.

Structure of the BIC

Example of composite information from positions 1 to 4 and with the above given example value on position 6. The data identifiers are highlighted in bold font:

1P072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

Accordingly as DMC:

Fig. 44: Example DMC **1P**072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

BTN

An important component of the BIC is the Beckhoff Traceability Number (BTN, position 2). The BTN is a unique serial number consisting of eight characters that will replace all other serial number systems at Beckhoff in the long term (e.g. batch designations on IO components, previous serial number range for safety products, etc.). The BTN will also be introduced step by step, so it may happen that the BTN is not yet coded in the BIC.

NOTE

This information has been carefully prepared. However, the procedure described is constantly being further developed. We reserve the right to revise and change procedures and documentation at any time and without prior notice. No claims for changes can be made from the information, illustrations and descriptions in this information.

8.3.4 Electronic access to the BIC (eBIC)

Electronic BIC (eBIC)

The Beckhoff Identification Code (BIC) is applied to the outside of Beckhoff products in a visible place. If possible, it should also be electronically readable.

Decisive for the electronic readout is the interface via which the product can be electronically addressed.

K-bus devices (IP20, IP67)

Currently, no electronic storage and readout is planned for these devices.

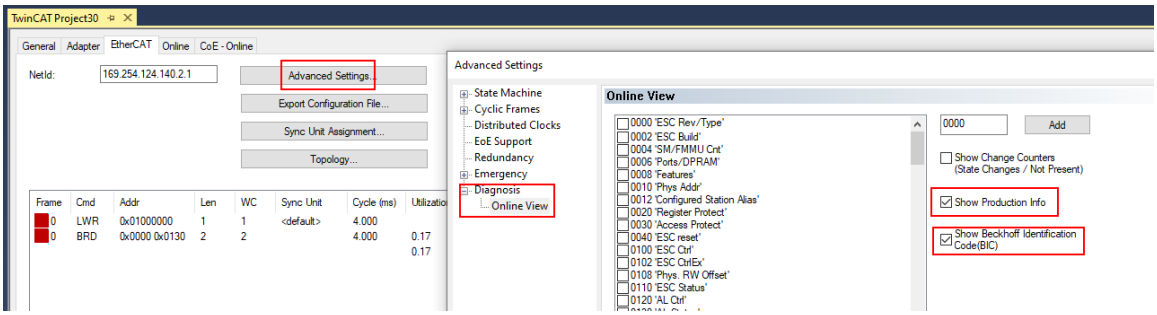
EtherCAT devices (IP20, IP67)

All Beckhoff EtherCAT devices have a so-called ESI-EEPROM, which contains the EtherCAT identity with the revision number. Stored in it is the EtherCAT slave information, also colloquially known as ESI/XML configuration file for the EtherCAT master. See the corresponding chapter in the EtherCAT system manual (Link) for the relationships.

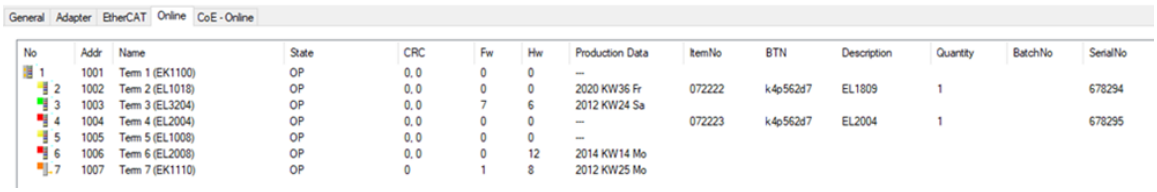
The eBIC is also stored in the ESI-EEPROM. The eBIC was introduced into the Beckhoff I/O production (terminals, box modules) from 2020; widespread implementation is expected in 2021.

The user can electronically access the eBIC (if existent) as follows:

- With all EtherCAT devices, the EtherCAT master (TwinCAT) can read the eBIC from the ESI-EEPROM
  - From TwinCAT 3.1 build 4024.11, the eBIC can be displayed in the online view.
  - To do this,
    check the checkbox "Show Beckhoff Identification Code (BIC)" under
    EtherCAT → Advanced Settings → Diagnostics:



  - The BTN and its contents are then displayed:



  - Note: as can be seen in the illustration, the production data HW version, FW version and production date, which have been programmed since 2012, can also be displayed with "Show Production Info".
  - From TwinCAT 3.1. build 4024.24 the functions *FB_EcReadBIC* and *FB_EcReadBTN* for reading into the PLC and further eBIC auxiliary functions are available in the Tc2_EtherCAT Library from v3.3.19.0.
- In the case of EtherCAT devices with CoE directory, the object 0x10E2:01 can additionally by used to display the device's own eBIC; the PLC can also simply access the information here:

- The device must be in PREOP/SAFEOP/OP for access:

Index	Name	Flags	Value
1000	Device type	RO	0x015E1389 (22942601)
1008	Device name	RO	ELM3704-0000
1009	Hardware version	RO	00
100A	Software version	RO	01
100B	Bootloader version	RO	J0.1.27.0
1011:0	Restore default parameters	RO	> 1 <
1018:0	Identity	RO	> 4 <
10E2:0	Manufacturer-specific Identification C...	RO	> 1 <
10E2:01	SubIndex 001	RO	1P158442SBTN0008jekp1KELM3704 Q1 2P482001000016
10F0:0	Backup parameter handling	RO	> 1 <
10F3:0	Diagnosis History	RO	> 21 <
10F8	Actual Time Stamp	RO	0x170bfb277e

- the object 0x10E2 will be introduced into stock products in the course of a necessary firmware revision.
- From TwinCAT 3.1. build 4024.24 the functions *FB_EcCoEReadBIC* and *FB_EcCoEReadBTN* for reading into the PLC and further eBIC auxiliary functions are available in the Tc2_EtherCAT Library from v3.3.19.0.

- Note: in the case of electronic further processing, the BTN is to be handled as a string(8); the identifier "SBTN" is not part of the BTN.
- Technical background
  The new BIC information is additionally written as a category in the ESI-EEPROM during the device production. The structure of the ESI content is largely dictated by the ETG specifications, therefore the additional vendor-specific content is stored with the help of a category according to ETG.2010. ID 03 indicates to all EtherCAT masters that they must not overwrite these data in case of an update or restore the data after an ESI update.
  The structure follows the content of the BIC, see there. This results in a memory requirement of approx. 50..200 bytes in the EEPROM.
- Special cases
  - If multiple, hierarchically arranged ESCs are installed in a device, only the top-level ESC carries the eBIC Information.
  - If multiple, non-hierarchically arranged ESCs are installed in a device, all ESCs carry the eBIC Information.
  - If the device consists of several sub-devices with their own identity, but only the top-level device is accessible via EtherCAT, the eBIC of the top-level device is located in the CoE object directory 0x10E2:01 and the eBICs of the sub-devices follow in 0x10E2:nn.

Profibus/Profinet/DeviceNet... Devices

Currently, no electronic storage and readout is planned for these devices.

8.4 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: https://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

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 5246 963 157
Fax:	+49 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 5246 963 460
Fax:	+49 5246 963 479
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963 0
Fax:	+49 5246 963 198
e-mail:	info@beckhoff.com
web:	https://www.beckhoff.com

More Information:
www.beckhoff.com/epp6228-0022

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com