BECKHOFF New Automation Technology

Manual | EN

CX8180

Embedded PC for RS232/RS485

3/2/2020 | Version: 1.3

EUROTEC
TECNOLOGIE IMPIANTISTICHE · AUTOMAZIONE · TELECONTROLLO

Table of contents

1 Notes on the documentation ... 5
1.1 Representation and structure of warnings ... 6
1.2 Documentation Issue Status ... 7
2 For your safety ... 8
2.1 Intended use ... 8
2.2 Staff qualification ... 8
2.3 Safety instructions ... 9
3 Transport and storage ... 10
4 Product overview ... 11
4.1 Structure ... 13
4.2 Name plate ... 14
4.3 Ethernet interfaces ... 15
4.4 RS232/RS485 interface ... 17
4.5 MicroSD card ... 18
4.6 Reset button ... 18
5 Commissioning ... 19
5.1 Mounting ... 19
5.1.1 Dimensions ... 19
5.1.2 Note the permissible installation positions ... 20
5.1.3 Securing on mounting rail ... 22
5.2 Connecting the power supply ... 23
6 Configuration ... 25
6.1 Operating system ... 25
6.1.1 Features included ... 26
6.1.2 Update image ... 27
6.1.3 FTP Server ... 28
6.2 IP address ... 30
6.2.1 Setting in the operating system ... 30
6.3 Web service ... 31
6.3.1 Starting the Beckhoff Device Manager ... 31
6.3.2 Enabling a remote display ... 32
6.3.3 Starting a remote connection ... 33
6.4 TwinCAT ... 34
6.4.1 Connecting to the CX81xx ... 34
6.4.2 Scanning for devices ... 36
6.4.3 Creating a PLC project ... 37
6.4.4 Linking variables ... 39
6.4.5 Using the KS8000 protocol ... 40
6.4.6 Activating KL6xx1 mode ... 41
7 Programming ... 43
7.1 Seconds UPS ... 43
7.1.1 Function block ... 45
7.1.2 Data types ... 47

7.1.3 PlcAppSystemInfo ... 48
7.2 Control diagnostic LEDs ... 49
7.2.1 Function F_CX81x0_LED_WD ... 49
7.2.2 Function F_CX81x0_LED_ERR ... 49
7.2.3 Data type E_CX81x0_LED ... 50
7.3 Function F_CX81xx_ADDRESS ... 50
7.4 Real Time Clock (RTC) ... 50

8 Ethernet X001 Interface ... 52
8.1 Ethernet ... 52
8.2 Topology example ... 54
8.3 ADS-Communication ... 55

9 Error handling and diagnosis ... 57
9.1 Diagnostic LEDs ... 57
9.2 Power supply terminal LEDs in K-bus mode ... 57
9.3 Power supply terminal LEDs in E-bus mode ... 60

10 Care and maintenance ... 61
10.1 Replace the battery ... 61

11 Technical data ... 62

12 Appendix ... 64
12.1 Certification ... 64
12.1.1 FCC ... 64
12.2 Support and Service ... 65

1 Notes on the documentation

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 the components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

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

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without prior announcement.
No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, und 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®

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

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.
The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.
Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

1.1 Representation and structure of warnings

The following warnings are used in the documentation. Read and follow the warnings.

Warnings relating to personal injury:

⚠ DANGER

Serious risk of injury

Hazard with high risk of death or serious injury.

⚠ WARNING

Risk of injury

Hazard with medium risk of death or serious injury.

⚠ CAUTION

Slight risk of injury

There is a low-risk hazard that can result in minor injury.

Warnings relating to damage to property or the environment:

NOTE

Damage to the environment or devices

There is a potential hazard to the environment and equipment.

Notes showing further information or tips:

Tip or pointer

This notice provides important information that will be of assistance in dealing with the product or software. There is no immediate danger to product, people or environment.

1.2 Documentation Issue Status

Version	Comment
1.0	First version
1.1	Chapter “Connecting the power supply” adjusted
1.2	Chapter “Technical data” revised and Chapter “FTP Server” added.
1.3	Chapter “Device Manager” adjusted.

2 For your safety

Read the chapter on safety and follow the instructions in order to protect from personal injury and damage to equipment.

Limitation of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Unauthorized modifications and changes to the hardware or software configuration, which go beyond the documented options, are prohibited and nullify the liability of Beckhoff Automation GmbH & Co. KG.

In addition, the following actions are excluded from the liability of Beckhoff Automation GmbH & Co. KG:

- Failure to comply with this documentation.
- Improper use.
- Untrained personnel.
- Use of unauthorized replacement parts.

2.1 Intended use

The CX81xx Embedded PC is a control system and is intended for mounting on a DIN rail in a control cabinet or terminal box.

The Embedded PC series is used in conjunction with Bus Terminals for recording digital or analog signals from sensors and transferring them to actuators or higher-level controllers.

The Embedded PC is designed for a working environment that meets the requirements of protection class IP20. This involves finger protection and protection against solid foreign objects up to 12.5 mm, but not protection against water. Operation of the devices in wet and dusty environments is not permitted, unless specified otherwise. The specified limits for electrical and technical data must be adhered to.

Improper use

The Embedded PC is not suitable for operation in the following areas:

- Potentially explosive atmospheres.
- Areas with an aggressive environment, e.g. aggressive gases or chemicals.
- Living areas. In living areas, the relevant standards and guidelines for interference emissions must be adhered to, and the devices must be installed in housings or control boxes with suitable attenuation of shielding.

2.2 Staff qualification

All operations involving Beckhoff software and hardware may only be carried out by qualified personnel with knowledge of control and automation engineering. The qualified personnel must have knowledge of the administration of the Embedded PC and the associated network.

All interventions must be carried out with knowledge of control programming, and the qualified personnel must be familiar with the current standards and guidelines for the automation environment.

2.3 Safety instructions

The following safety instructions must be followed during installation and working with networks and the software.

Mounting

- Never work on live equipment. Always switch off the power supply for the device before installation, troubleshooting or maintenance. Protect the device against unintentional switching on.
- Observe the relevant accident prevention regulations for your machine (e.g. the BGV A 3, electrical systems and equipment).
- Ensure standard-compliant connection and avoid risks to personnel. Ensure that data and supply cables are laid in a standard-compliant manner and ensure correct pin assignment.
- Observe the relevant EMC guidelines for your application.
- Avoid polarity reversal of the data and supply cables, as this may cause damage to the equipment.
- The devices contain electronic components, which may be destroyed by electrostatic discharge when touched. Observe the safety precautions against electrostatic discharge according to DIN EN 61340-5-1/-3.

Working with networks

- Limit physical and electronic access to all devices to an authorized group of persons.
- Change the default passwords to reduce the risk of unauthorized access. Regularly change the passwords.
- Install the devices behind a firewall.
- Apply the IT security precautions according to IEC 62443, in order to limit access to and control of devices and networks.

Working with the software

- Use up-to-date security software. The safe function of the Embedded PC can be compromised by malicious software such as viruses or Trojans.
- The sensitivity of an Embedded PC against malicious software increases with the number of installed and active software.
- Uninstall or disable unnecessary software.

Further information about the safe handling of networks and software can be found in the Beckhoff Information System:
http://infosys.beckhoff.com

Document name
Documentation about IPC Security

3 Transport and storage

Transport

NOTE

Short circuit due to moisture

Moisture can form during transport in cold weather or in the event of large temperature fluctuations.

Avoid moisture formation (condensation) in the Embedded PC, and leave it to adjust to room temperature slowly. If condensation has occurred, wait at least 12 hours before switching on the Embedded PC.

Despite the robust design of the unit, the components are sensitive to strong vibrations and impacts. During transport the Embedded PC must be protected from

- mechanical stress and
- use the original packaging.

Table 1: Dimensions and weight of the CX8180 Embedded PC.

	CX8180
Dimensions (W x H x D)	71 mm x 100 mm x 73 mm
Weight	230 g

Storage

- The battery should be removed if the Embedded PC is stored at temperatures above 60 °C. The battery should be stored separate from the Embedded PC in a dry environment at a temperature between 0 °C and 30 °C.
  The preset date and time are lost if the battery is removed.
- Store the Embedded PC in the original packaging.

4 Product overview

CX8100 designates a product family of Embedded PCs based on a 32-bit ARM CPU. The CX8100 Embedded PC is programmable and is able to execute its own control program. In addition to that the Embedded PC acts as a slave device of a higher-level fieldbus system.

The CX8100 Embedded PC has the following basic configuration:

- a 512 MB MicroSD card,
- an Ethernet interface
- as well as two switched Ethernet interfaces (2 x RJ45, switched).

You can use the CX8100 Embedded PCs as decentralized controllers and in this way ensure that the local program continues to be executed on the CX8100 in the event of an interruption in the higher-level fieldbus system.

The operating system is Microsoft Windows Embedded Compact 7. Because there is no monitor port, the operating system and its "virtual" display can only be accessed via the network. Beckhoff Device Manager and Remote Display (Cerhost)

The Embedded PC features an internal 1-second UPS as persistent data memory. The 1-second UPS enables persistent data to be saved to the MicroSD card in the event of a power failure.

Power supply terminal

The power supply terminal for the Embedded PC is located on the right-hand side. Bus Terminals (K-bus) or EtherCAT Terminals (E-bus) can be attached on the right-hand side of the power supply terminal. The power supply terminal automatically recognizes the respective bus system (K-bus or E-bus).

The use of EtherCAT Terminals (E-bus) enables further options, such as the implementation of different topologies, the integration of further bus systems such as CANopen, PROFIBUS and PROFINET and – with the EtherCAT Box Modules – connection to the IP67 world.

Fieldbus interface

CX8100 devices are being prepared for further fieldbus systems such as EtherCAT (slave), PROFINET, EtherNet/IP, CANopen, PROFIBUS and other communication systems.

Available fieldbus systems:

- CX8110: EtherCAT
- CX8180: RS232/485
- CX8190: Ethernet (Realtime Ethernet, ADS UDP, ADS TCP, EAP, Web Services)
- CX8191: BACnet (client and server)

Programming

The CX8100 Embedded PCs are programmed according to the high-performance IEC 61131-3 standard. The TwinCAT 3 automation software forms the basis for the programming of the Embedded PC.

Configuration

The CX8100 Embedded PC is commissioned via the Ethernet interface. The fieldbus interface and all connected devices such as EtherCAT Terminals or Bus Terminals are then read out via TwinCAT 3. The configuration is stored on the Embedded PC after the parameterization. The configuration thus created can be accessed again later.

The shortest usable task time is 500 µs, although this is only achievable with a very small system load. A task time of 1 to 50 ms is recommended for the I/O data. Other tasks can also be set slower. When using shorter cycle times, the total system load is to be observed.

If too short a cycle time is selected, the Web visualization and Remote Display may operate very slowly or cause timeouts. The user is responsible for configuring his system such that it is not overloaded.

4.1 Structure

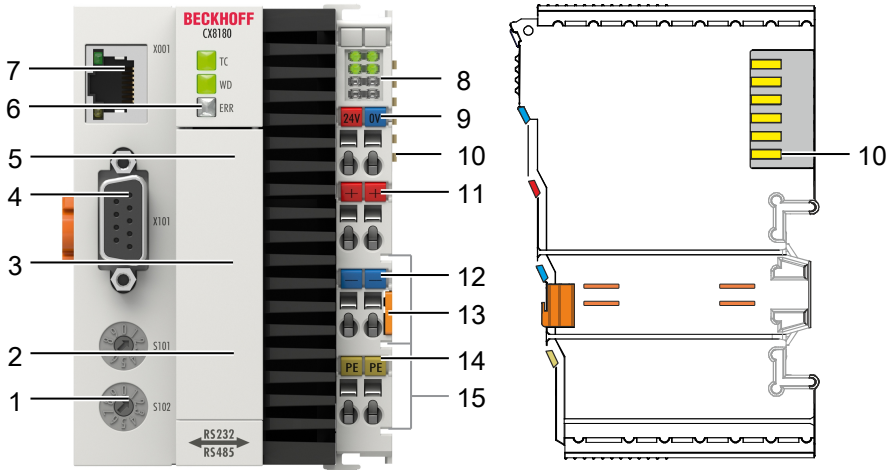


Fig. 1: Sample configuration of a CX8180 Embedded PC.

Table 2: Legend for the configuration.

No.	Component	Description
1	Address selection switch (S101, S102)	With address selection switch
2	Battery compartment [▶ 61] (under the front flap).	Power supply for the battery-backed clock for time and date.
3	MicroSD card slot [▶ 18] (under the front flap).	Slot for industrial MicroSD cards.
4	D-Sub socket [▶ 17] (X101).	Interface for connecting the RS232/RS485 serial interfaces.
5	Reset button [▶ 18]	This switches the Embedded PC to Config mode.
6	Diagnostic LEDs [▶ 57].	Diagnostic LEDs. You can create your own diagnostic messages for the WD and ERR diagnostic LEDs (see: Control diagnostic LEDs [▶ 49]).
7	Ethernet interface [▶ 15] (X001).	Interface for commissioning and programming the Embedded PC.
8	Diagnostic LEDs, power supply terminal [▶ 57].	Diagnosis of the power supply for the Embedded PC and the terminal bus. Status of the E-bus and K-bus communication.
9	Spring-loaded terminals, +24 V and 0 V	Power supply for Embedded PC.
10	Terminal bus (K-bus or E-bus)	Interface for EtherCAT Terminals or Bus Terminals. Data exchange and supply.
11	Spring-loaded terminal, +24 V	Power supply for Bus Terminals via power contact.
12	Spring-loaded terminal, 0 V	Power supply for Bus Terminals via power contact.
13	Terminal release	Releases the power supply terminal and therefore the Embedded PC from the mounting rail.
14	Spring-loaded terminal, PE	Spring-loaded terminal for power contact PE.
15	Power contacts, +24 V, 0 V, PE	Power contacts for Bus Terminals.

4.2 Name plate

The CX8180 Embedded PC features a name plate on the left-hand side of the housing.

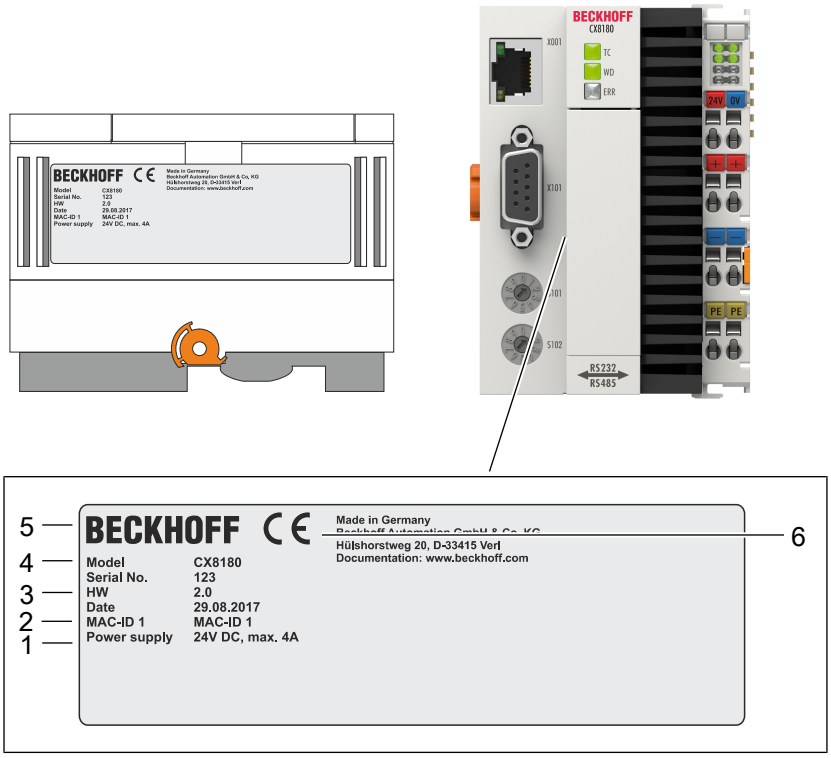


Fig. 2: CX8180 name plate.

Table 3: Legend for the name plate.

No.	Description
1	Information on the power supply unit. 24 V DC, 4 A max.
2	MAC address of the Ethernet interface X001. By default, the host name is formed from CX plus the last 3 bytes of the MAC address: for example, the MAC address: 00-01-05-aa-bb-cc results in the host name **CX-aabbcc**.
3	Information on: • serial number, • hardware version • and date of manufacture.
4	Information on the model. The last two numbers code the version of the Embedded PC.
5	Vendor data including address.
6	CE conformity.

4.3 Ethernet interfaces

You can program and commission the CX8180 Embedded PC via the X001 Ethernet interface. The Ethernet interface achieves speeds of 10 / 100 Mbit.

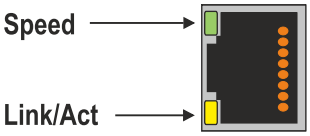


Fig. 3: Ethernet interface X001.

The LEDs on the left of the interfaces indicate the connection status. The lower LED (LINK/ACT) indicates whether the interface is connected to a network. If this is the case the LED is yellow. The LED flashes when data transfer on the interface is in progress.

The upper LED (SPEED) indicates the connection speed. The LED is not lit if the speed is 10 Mbit. At 100 Mbit the LED is green.

Fig. 4: Ethernet interface, pin numbering.

Ethernet interface X001

Table 4: Ethernet interface X001, pin assignment.

PIN	Signal	Description
1	TD +	Transmit +
2	TD -	Transmit -
3	RD +	Receive +
4	connected	reserved
5		
6	RD -	Receive -
7	connected	reserved
8		

Transmission standards

10Base5

The transmission medium for 10Base5 consists of a thick coaxial cable ("yellow cable") with a max. transmission speed of 10 Mbaud arranged in a line topology with branches (drops) each of which is connected to one network device. Because all the devices are in this case connected to a common transmission medium, it is inevitable that collisions occur often in 10Base5.

10Base2

10Base2 (Cheaper net) is a further development of 10Base5, and has the advantage that the coaxial cable is cheaper and, being more flexible, is easier to lay. It is possible for several devices to be connected to one 10Base2 cable. It is frequent for branches from a 10Base5 backbone to be implemented in 10Base2.

10BaseT

Describes a twisted pair cable for 10 Mbaud. The network here is constructed as a star. It is no longer the case that every device is attached to the same medium. This means that a broken cable no longer results in failure of the entire network. The use of switches as star couplers enables collisions to be reduced. Using full-duplex connections they can even be entirely avoided.

100BaseT

Twisted pair cable for 100 MBaud. It is necessary to use a higher cable quality and to employ appropriate hubs or switches in order to achieve the higher data rate.

10BaseF

The 10BaseF standard describes several optical fiber versions.

Short description of the 10BaseT and 100BaseT cable types

Twisted pair copper cable for star topologies, where the distance between two devices may not exceed 100 meters.

UTP

Unshielded twisted pair
This type of cable belongs to category 3, and is not recommended for use in an industrial environment.

S/UTP

Screened/unshielded twisted pair (screened with copper braid)
Has a general screen of copper braid to reduce influence of external interference. This cable is recommended for use with Bus Couplers.

FTP

Foiled shielded twisted pair (screened with aluminum foil)
This cable has an outer screen of laminated aluminum and plastic foil.

S/FTP

Screened/foiled-shielded twisted pair (screened with copper braid and aluminum foil)
Has a laminated aluminum screen with a copper braid on top. Such cables can provide up to 70 dB reduction in interference power.

STP

Shielded twisted pair
Describes a cable with an outer screen, without defining the nature of the screen any more closely.

S/STP

Screened/shielded twisted pair (wires are individually screened)
This identification refers to a cable with a screen for each of the two wires as well as an outer shield.

ITP

Industrial Twisted-Pair
The structure is similar to that of S/STP, but, in contrast to S/STP, it has only one pair of conductors.

4.4 RS232/RS485 interface

Two serial interfaces are wired on the D-Sub socket: one with RS232 and one with RS485 physics. Both interfaces have the same potential and are electrically isolated from the CPU. The interfaces can be used simultaneously and work independently of each other.

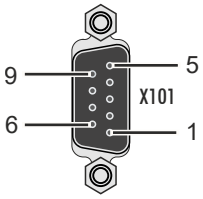


Fig. 5: X101 RS232/RS485 interface

Observe the PIN assignment of the D-Sub socket when assembling cables for wiring. The ZB3180 connector with two cable outlets can be used for wiring (see: D-sub connector ZB3180).

Table 5: X101 RS232/RS485 interface, PIN assignment.

PIN	Meaning	Description	Signal
1	**RS485**	**(+)**	**A**
2	RxD (RS232)	Signal in	Receive Data
3	TxD (RS232)	Signal out	Transmit Data
4	+ 5 V	+	Vcc
5	GND	Ground	Ground
6	**RS485**	**(-)**	**B**
7	RTS (RS232)	Signal out	Request to Send
8	CTS (RS232)	Signal in	Clear to Send
9	GND	Ground	Ground

Termination resistor

Usually, no termination resistors are required for low baud rates. A termination resistor can lead to an improved transmission level in cases where long lines, a high line resistance, a large number of RS485 devices and/or high baud rates are used. RS485 is implemented in different ways, and it is not necessarily clear whether a termination resistor improves or impairs the result.

Our recommendation is to use an active termination resistor (PIN4 - 390 Ohm, PIN1 - 220 Ohm, PIN6 - 390 Ohm, PIN5/9). PIN4 is assigned 5 V_{DC} and PIN5/9 GND for the active termination resistor (RS485). PIN4 should only be used for the termination resistor. Do not operate any other electrical components on PIN4, otherwise the CX8180 may be destroyed.

4.5 MicroSD card

In the basic configuration the CX81xx contains a MicroSD card with 512 MB. You can order it as an option with larger cards (up to 8 GB).

The cards employed are SLC memory with extended temperature range for industrial applications. Use exclusively MicroSD cards approved by Beckhoff.

Example of a MicroSD card:

Fig. 6: MicroSD card for the CX81xx.

Order designation	Capacity	Description
CX1900-0123	1 GB	MicroSD card (SLC memory) with extended temperature range for industrial applications instead of the 512 MB card (ordering option)
CX1900-0125	2 GB	
CX1900-0127	4 GB	
CX1900-0129	8 GB	

Order designation	Capacity	Description
CX1900-0122	512 MB	MicroSD card (SLC memory) with extended temperature range for industrial applications as spare part.
CX1900-0124	1 GB	
CX1900-0126	2 GB	
CX1900-0128	4 GB	
CX1900-0130	8 GB	

4.6 Reset button

Use the Reset button to activate Config mode. You can use this function if the PLC program unexpectedly causes an error. To do this, keep the Reset button pressed during the restart for an extended period.

Activate Config mode as follows:

1. Open the front flap.
2. Switch off the Embedded PC.
3. Start the Embedded PC and keep the Reset button pressed until the ERR LED turns red and then yellow.

⇨ The CX81xx Embedded PC is put into Config mode.

5 Commissioning

5.1 Mounting

5.1.1 Dimensions

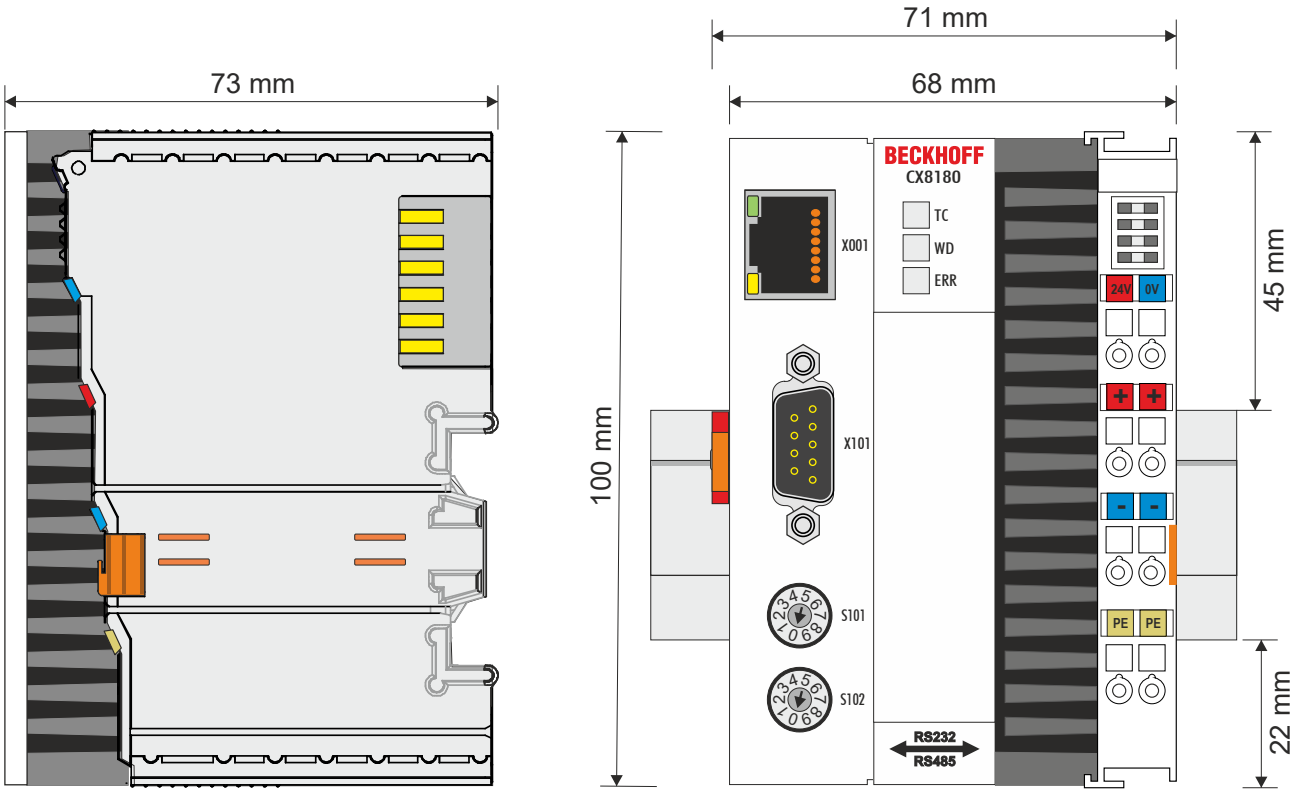


Fig. 7: Dimensions of the CX81xx Embedded PC.

Technical drawings in DWG and STP formats can be found at:

http://www.beckhoff.com

5.1.2 Note the permissible installation positions

Increased heat generation

The Embedded PC may overheat if the installation position is incorrect or the minimum distances are not adhered to.

Ensure adequate ventilation. A horizontal installation position is ideal. Leave at least 30 mm clearance above and below the Embedded PC.

Note the following specifications for the control cabinet:

- Keep to the prescribed ambient temperature. Measure the temperature below the Embedded PC at a distance of 30 mm to the cooling fins, in order to determine the ambient temperature correctly.
- Adhere to the minimum distances of 30 mm above and below the Embedded PCs.
- Additional electrical equipment affects the heat generation in the control cabinet. Select a suitable control cabinet enclosure depending on the application, or ensure that excess heat is dissipated from the control cabinet.

Prescribed installation position for temperatures up to 60 °C

Install the Embedded PC horizontally in the control cabinet on a mounting rail, in order to ensure optimum heat dissipation.

Ventilation openings are located at the top and bottom of the housing. This ensures an optimum airflow through the Embedded PC in vertical direction. In addition, a minimum clearance of 30 mm above and below the Embedded PCs is required, in order to ensure adequate ventilation.

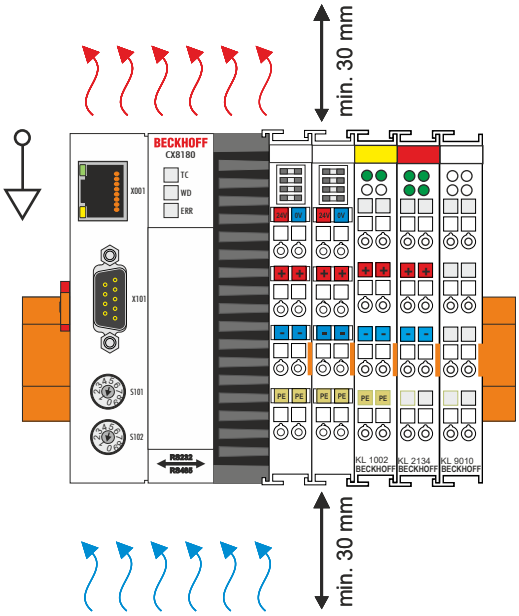


Fig. 8: Embedded PC CX8180, horizontal installation position.

Installation positions with reduced temperature range up to 50 °C

You can also mount the Embedded PC vertically or horizontally on the mounting rail. Note that you can then only operate the Embedded PC up to an ambient temperature of 50 °C.

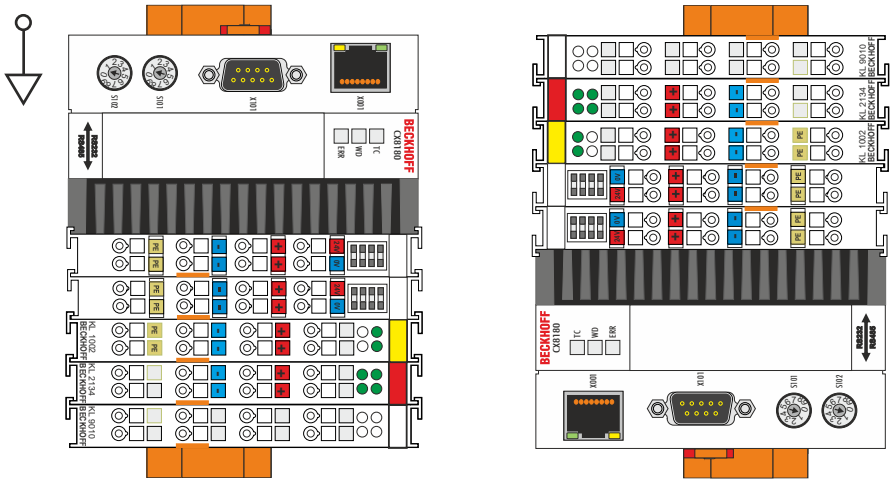


Fig. 9: Embedded PC CX8180, vertical installation position.

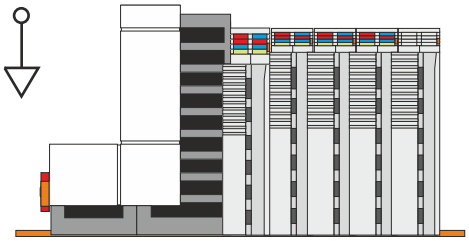


Fig. 10: Embedded PC CX8180, horizontal installation position.

Ensure that Bus Terminals that are connected to the Embedded PC are designed for operation in vertical or horizontal position.

5.1.3 Securing on mounting rail

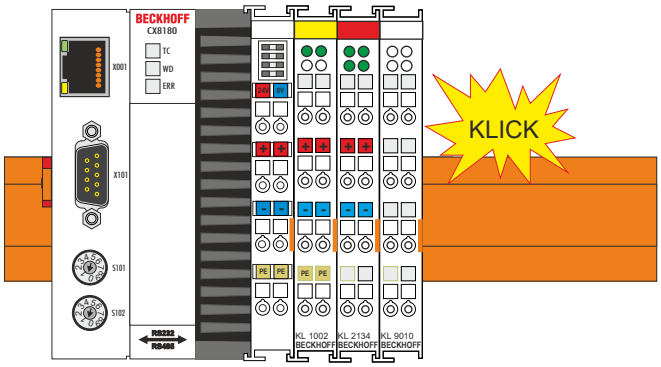
The housing is designed such that the Embedded PC can be pushed against the mounting rail and latched onto it. The Embedded PC is fastened to the DIN rail by means of a catch on the left side of the Embedded PC.

Requirements:

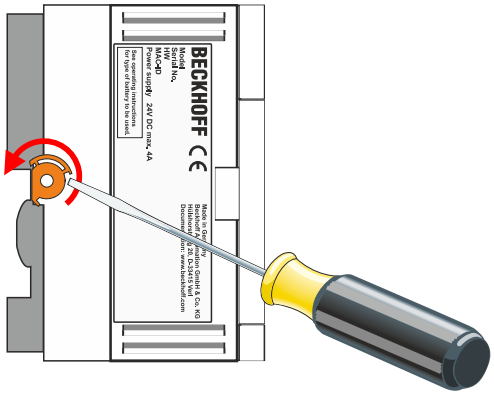
- Mounting rail of type TS35/7.5 or TS35/15 according to DIN EN 60715.

Secure the Embedded PC on the mounting rail as follows:

1. Place the Embedded PC at the front of the mounting rail. Slightly press the Embedded PC onto the mounting rail until a soft click can be heard and the Embedded PC has latched.



2. Subsequently, lock the catch on the left side of the Embedded PC. Use a screwdriver to do this.



⇨ Double-check the correct installation and latching of the Embedded PC on the mounting rail.

5.2 Connecting the power supply

NOTE

Damage to the Embedded PCs

The Embedded PCs may be damaged during wiring.

- The cables for the power supply should only be connected in de-energized state.

The power supply terminals require an external voltage source, which provides 24 V DC (-15% / +20%). The power supply terminal must provide 4 A at 24 V, in order to ensure the operation of the Embedded PCs in all situations.

The cabling of the Embedded PC in the control cabinet must be done in accordance with the standard EN 60204-1:2006 PELV = Protective Extra Low Voltage:

- The "PE" and "0 V" conductors of the voltage source for a basic CPU module must be on the same potential (connected in the control cabinet).
- Standard EN 60204-1:2006, section 6.4.1:b stipulates that one side of the circuit, or a point of the energy source for this circuit must be connected to the protective earth conductor system.

Connection example

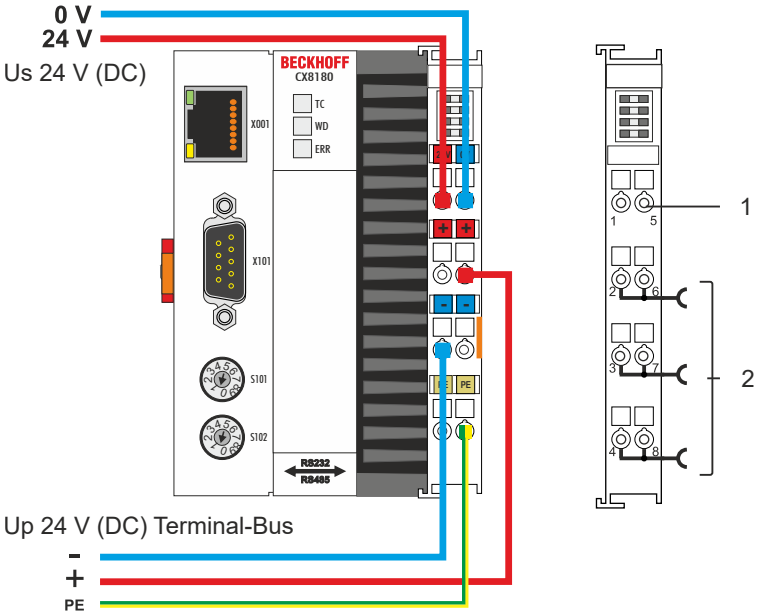


Table 6: Legend for the connection example

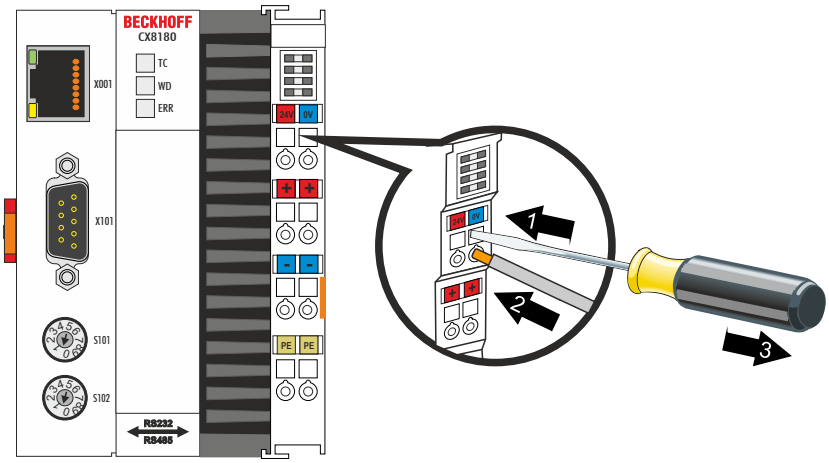
No.	Description
1	The upper spring-loaded terminals identified with "24 V" and "0 V" supply the Embedded PC and the terminal bus (data transfer via K-bus or E-bus).
2	The spring-loaded terminals identified as "+", "-" and "PE" supply the Bus Terminals via the power contacts and the sensors or actuators connected to the Bus Terminals.

Opening and closing spring-loaded terminals:

The cables of an external voltage source are connected to the power supply unit with spring-loaded terminals. Connect the cables as follows:

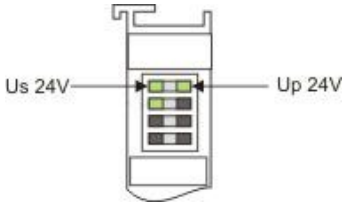
Table 7: Required wire cross-sections and strip lengths

Wire cross-section	0.5 ... 2.5 mm^2	AWG 20 .. AWG 14
Strip length	8 ... 9 mm	0.33 inch



The voltage source has been connected to the power supply unit successfully when the two upper power supply terminal LEDs light up in green.

- The left LED (Us) indicates the supply of the basic CPU module and terminal bus.
- The red LED (Up) indicates the Bus Terminal supply via the power contacts.



NOTE

Interrupting / switching off the power supply

To switch off the Embedded PC, do not disconnect the ground (0 V), because otherwise current may continue to flow via the shielding, depending on the device, and damage the Embedded PC or peripheral devices.

- Always disconnect the 24 V line. Devices connected to the Embedded PC, which have their own power supply (e.g. a Panel) must have the same potential for "PE" and "0 V" as the Embedded PC have (no potential difference).

Observe the UL requirements

The CX8180 Embedded PCs are UL certified. The corresponding UL label can be found on the type plate.

The CX8180 Embedded PCs can thus be used in areas in which special UL requirements have to be met. These requirements apply to the system voltage (Us) and to the power contacts (Up). Application areas without special UL requirements are not affected by UL regulations.

UL requirements

- The Embedded PCs must not be connected to unlimited voltage sources.
- Embedded PCs may only be supplied from a 24 V DC voltage source. The voltage source must be insulated and protected with a fuse of maximum 4 A (corresponding to UL248).
- Or the power supply must originate from a voltage source that corresponds to NEC class 2. An NEC class 2 voltage source must not be connected in series or parallel with another NEC class 2 voltage source.

6 Configuration

6.1 Operating system

The Microsoft Windows Embedded Compact 7 operating system is used on the CX8180 Embedded PC. This operating system is optimized for the CX8180 Embedded PC. This means that not all features of Windows Embedded Compact 7 are available.

Security

For reasons of security the CERHOST and TELNET services are deactivated in the delivery state. To reactivate these services, you need a MicroSD card reader.

CERHOST

CERHOST is deactivated by current images on first start-up via the registry file *CeRemoteDisplay_Disable.reg*, which is located in the folder *RegFiles*.

To reactivate CERHOST you have to delete the file *CeRemoteDisplay_Disable.reg* from the folder *RegFiles* and also the folder *Documents and Settings*

Then reinsert the MicroSD card in the Embedded PC and reboot. The Embedded PC creates a new *Document and Settings* directory and then reboots automatically.

The Embedded PC is then accessible again via CERHOST.

TELNET

TELNET is deactivated by current images on first start-up via the registry file *Telnet_Disable.reg*, which is located in the folder *RegFiles*.

To reactivate TELNET you have to delete the file *Telnet_Disable.reg* from the folder *RegFiles* and also the folder *Documents and Settings*.

Then reinsert the MicroSD card in the Embedded PC and reboot. The Embedded PC creates a new *Document and Settings* directory and then reboots automatically.

6.1.1 Features included

Features	CX8180
ATL	X
MFC	X
XML DOM	X
XML Minimal Parser	X
COM	X
DCOM	X
COM Storage	X
Winsock	X
TCP/IP	X
TCP/IPv6	X
Firewall	X
Network Utilities (IpConfig, Ping, Route)	X
Object Exchange Protocol OBEX	-
Message Queuing MSMQ	-
UPnP	
Control Point	-
Device Host	X
SOAP	
Client	-
Server	-
Server	-
File Server (SMB/CIFS)	X
FTP Server	X
Print Server (SMB/CIFS)	-
RAS Server / PPTP Server	X
Simple Network Time Protocol (SNTP)	X
SNTP Client Service	X
Simple Network Management Protocol (SNMP)	X
Telnet Server	X
Web Server (HTTPD) / Active Server Pages (ASP) Support / JScript 5.8 / VBScript 5.8	X
Internet Explorer 7.0	-
NET Compact Framework	v3.5
RDP Client (Remote Desktop protocol)	-
CAB File Installer/Uninstaller	X

6.1.2 Update image

NOTE

Loss of data

All data on the MicroSD card will be deleted. Backup any data that you may have on the MicroSD card before proceeding.

The new image will be copied directly to the MicroSD card in order to update the image of the Embedded PC.

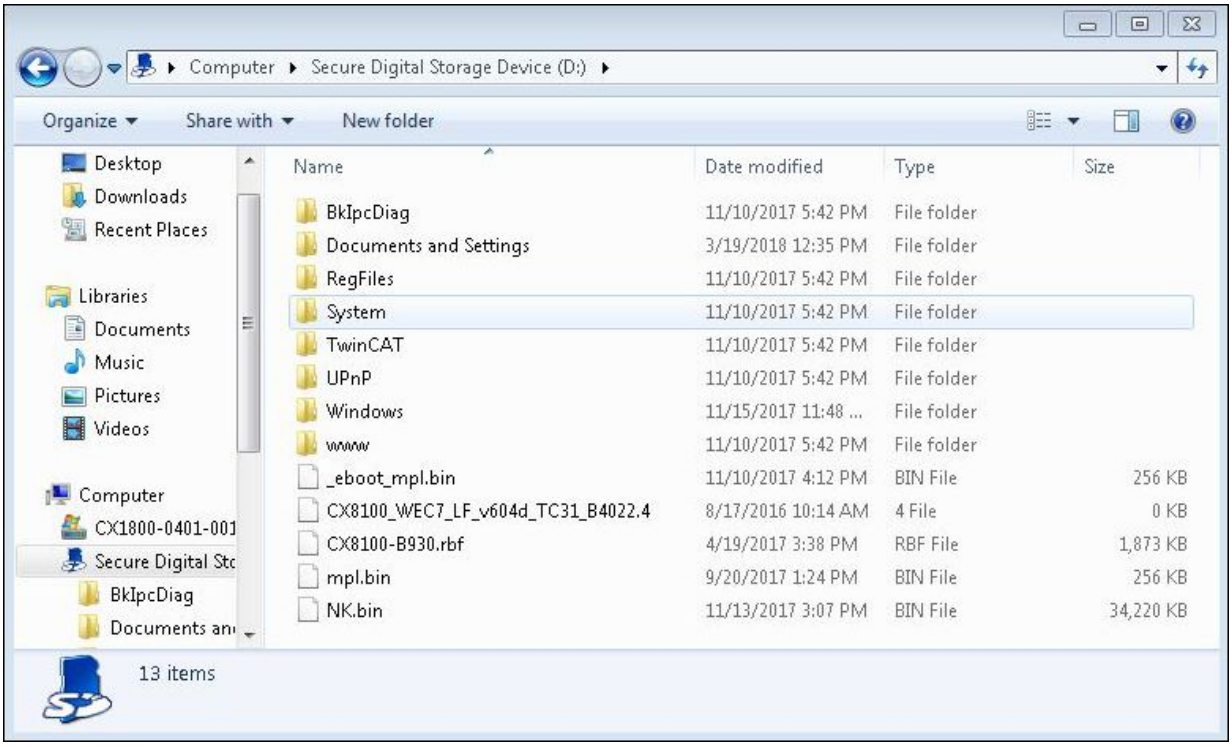
The new image is made available by Beckhoff Service. Perform the update only after consulting with Beckhoff Service.

Requirements:

- Card reader for MicroSD cards.

Update the image as follows:

1. Switch the Embedded PC off and remove the MicroSD card from the Embedded PC.
2. Insert the MicroSD card into an external card reader and open the MicroSD card's folder tree.
3. Delete all files and folders on the MicroSD card.



4. Copy all files and folders of the new image to the empty MicroSD card.
5. Install the MicroSD card in the Embedded PC again.
6. Start the Embedded PC.

⇨ The Embedded PC is started and saves the current hardware configuration in the folder **Documents and Settings**. The image has now been successfully updated.

6.1.3 FTP Server

Restricted access through firewall

From image version "CX8100_WEC7_LF_v604h_TC31_B4022.20", the firewall for the CX8180 is enabled by default. This means that a passive FTP connection (as used by Microsoft, for example) cannot be established. We therefore recommend using active FTP access. Enter TCP ports 20 and 21 in the firewall.

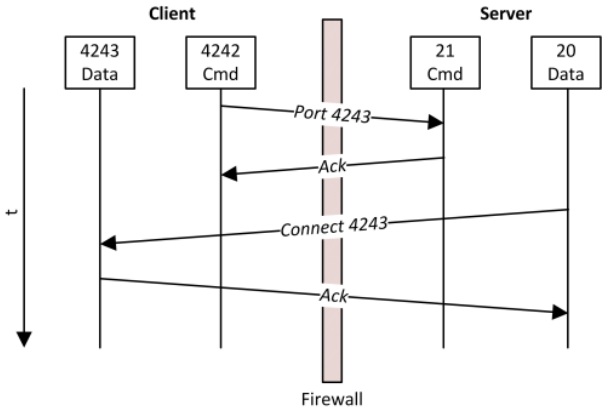
The File Transfer Protocol (FTP) is based exclusively on TCP-based communication connections. FTP specifies two TCP ports, which are important for data transfer:

- Port 20/tcp: This port is also referred to as **data port** and is used to send/receive files and directory lists.
- Port 21/tcp: This port is generally referred to as **command port** and is used to exchange status information between the client and server.

Separate TCP connections are used for sending and receiving files (data port) and for transmitting commands (command port). With FTP, two connection modes are available for setting up such connections: "**Active FTP**" and "**Passive FTP**". Depending on the connection mode, the two ports mentioned above are initiated differently, which is described in more detail below.

Active FTP

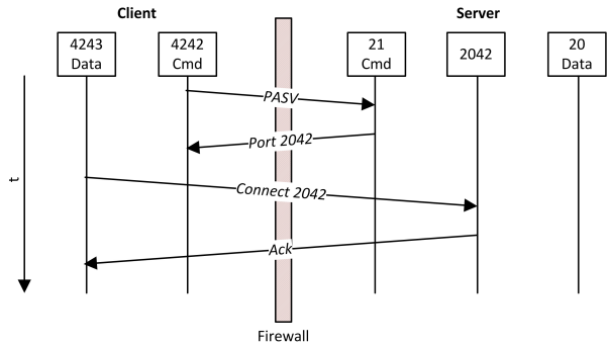
With active FTP, the client connects to the command port of the FTP server. The client uses a random port N, e.g. 4242/tcp, as source port. The client then listens on port N+1 and notifies the server of this port. The server then connects to the client on port N+1 and uses its data port as the source port.



A problem with active FTP is that the client itself does not establish a connection to the server's data port, but communicates a port (N+1) to the server, which then connects to the client via its data port. In the case of firewalls or NAT devices that are located upstream of the client, this could involve additional configuration effort on the client side, since the data port of the client behind the firewall must be accessible to the server (see figure "Connect 4243").

Passive FTP

This method is used when the client is not directly accessible by the server. This is the case, for example, if the client is behind a firewall that uses NAT to rewrite the client's address. With passive FTP, the FTP client initiates a connection via two random TCP ports N (command port) and N+1 (data port). The first port is used to connect to the server's command port. However, instead of the client communicating its port N+1 to the server so that the server can open a connection to it (see active FTP), the client first transmits a so-called PASV command. The server now knows that the connection is via passive FTP. As a result, the server opens a (random) port P as data port and transmits it to the client. The client then initiates a connection with port P and uses port N+1 (data port) as the source port. This connection is then used to transfer the data.



On closer examination it becomes apparent that the firewall problem of active FTP is reversed with passive FTP. On the server side, the firewall should be configured such that the data port of the server can be reached by the client. Many FTP servers offer the option to configure the data ports to be used.

6.2 IP address

6.2.1 Setting in the operating system

Under Windows Embedded Compact 7, the X001 Ethernet interface is displayed as EMAC1.

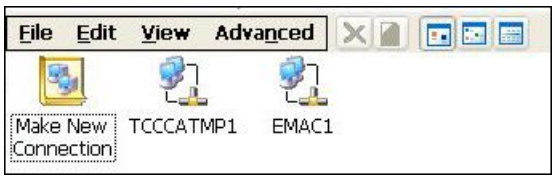


Fig. 11: Ethernet interface with Windows Embedded Compact 7.

EMAC1 (X001)

As standard, DHCP is active and the IP address is assigned automatically. You can deactivate DHCP and assign a static IP address.

6.3 Web service

6.3.1 Starting the Beckhoff Device Manager

Using the Beckhoff Device Manager, an Industrial PC can be configured by remote access with the aid of a web browser. Depending on the image version, access takes place via different protocols and requires different open ports. For older image versions access takes place via the HTTP protocol and Port 80 (TCP). More up-to-date image versions use HTTPS and Port 443 (TCP).

Requirements:

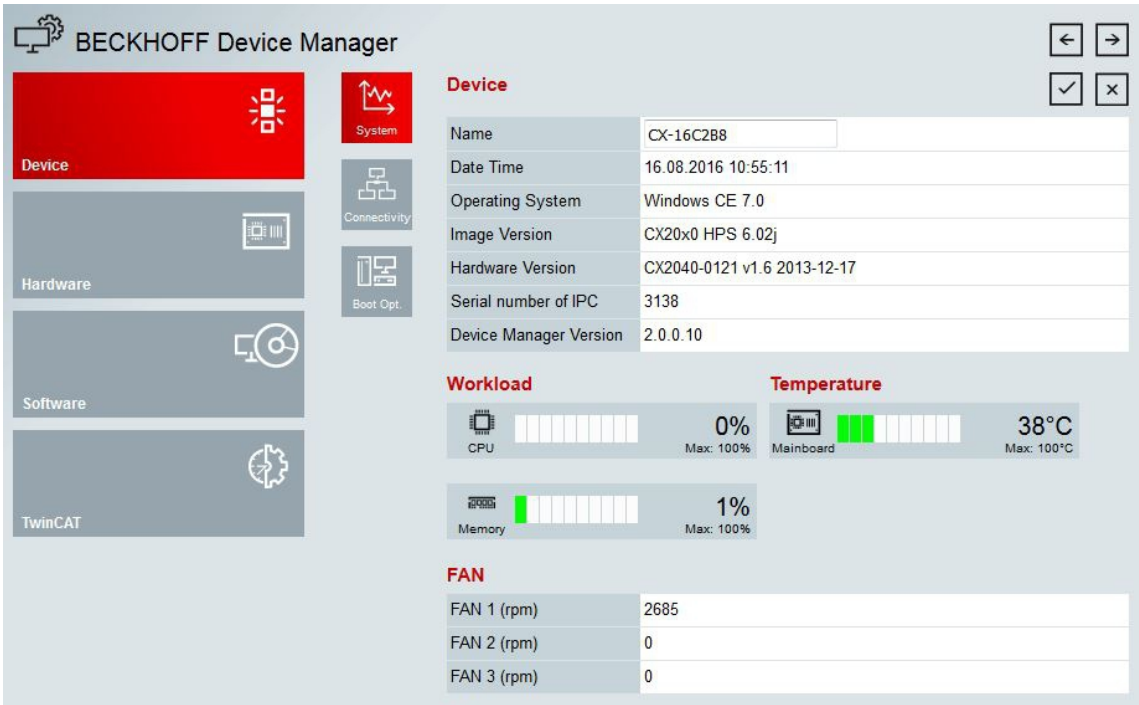
- Host PC and Embedded PC must be located in the same network. Depending on the operating system version, the network firewall must allow access via port 80 (HTTP) or port 443 (HTTPS).
- IP address or host name of the Embedded PC.

Table 8: Access data for the Beckhoff Device Manager on delivery.

Operating system	Access data
Windows Embedded Compact 7	User name: Administrator Password: 1

Start the Beckhoff Device Manager as follows:

1. Open a web browser on the host PC.
2. Enter the IP address or the host name of the Industrial PC in the web browser to start the Beckhoff Device Manager.
   - Example with IP address: https://169.254.136.237/config
   - Example with host name: https://CX-16C2B8/config
3. Enter the user name and password. The start page appears:



⇨ Navigate forward in the menu and configure the Industrial PC. Note that modifications only become active once they have been confirmed. It may be necessary to restart the Industrial PC.

6.3.2 Enabling a remote display

So that you can remotely access an Industrial PC with CE operating system, you must first activate Remote Display in the Beckhoff Device Manager. The remote display is disabled by default.

Requirements:

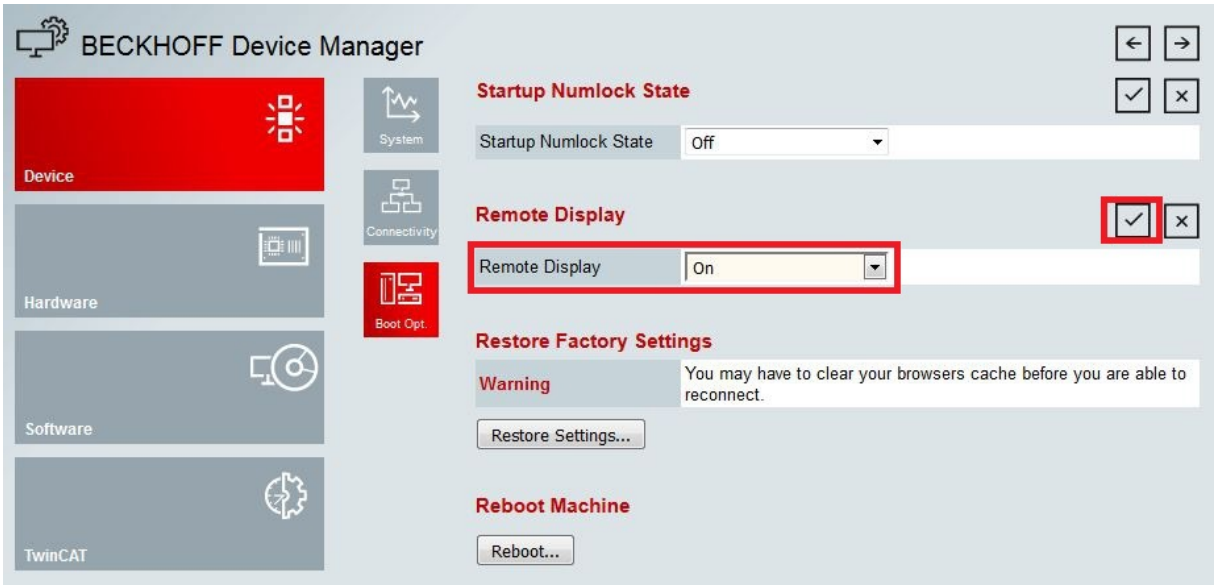
- Host PC and Embedded PC must be located in the same network. Depending on the operating system version, the network firewall must allow access via port 80 (HTTP) or port 443 (HTTPS).
- The IP address or the host name of the Embedded PC must be known.

Table 9: Access data for the Beckhoff Device Manager on delivery.

Operating system	Access data
Windows Embedded Compact 7	User name: Administrator Password: 1

Enable the remote display as follows:

1. Open a web browser on the host PC.
2. Enter the IP address or the host name of the Industrial PC in the web browser to start the Beckhoff Device Manager.
   - Example with IP address: https://169.254.136.237/config
   - Example with host name: https://CX-16C2B8/config
3. Enter the user name and password. The start page appears.
4. In the menu under **Device** click on **Boot Opt**.



5. Under **Remote Display** select the option **On** and confirm the settings.
6. In the information window click **OK** to accept the settings.

⇨ You have successfully activated Remote Display on the Industrial PC. After restarting, you can remotely access your Industrial PC.

6.3.3 Starting a remote connection

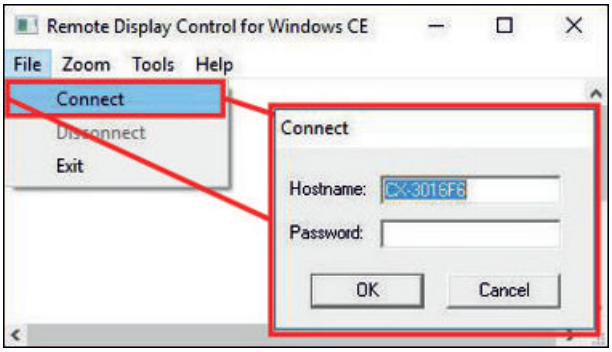
With the aid of the Remote Display Control program (CERHOST), a remote connection can be established and an Industrial PC with CE operating system can be remotely controlled from a host PC.

Requirements:

- Remote Display is active. See: Enabling a remote display.
- Host name of the Embedded PC.
- Remote Display Control (CERHOST). Download under: https://infosys.beckhoff.com/content/1033/CX8180_HW/Resources/zip/5047075211.zip

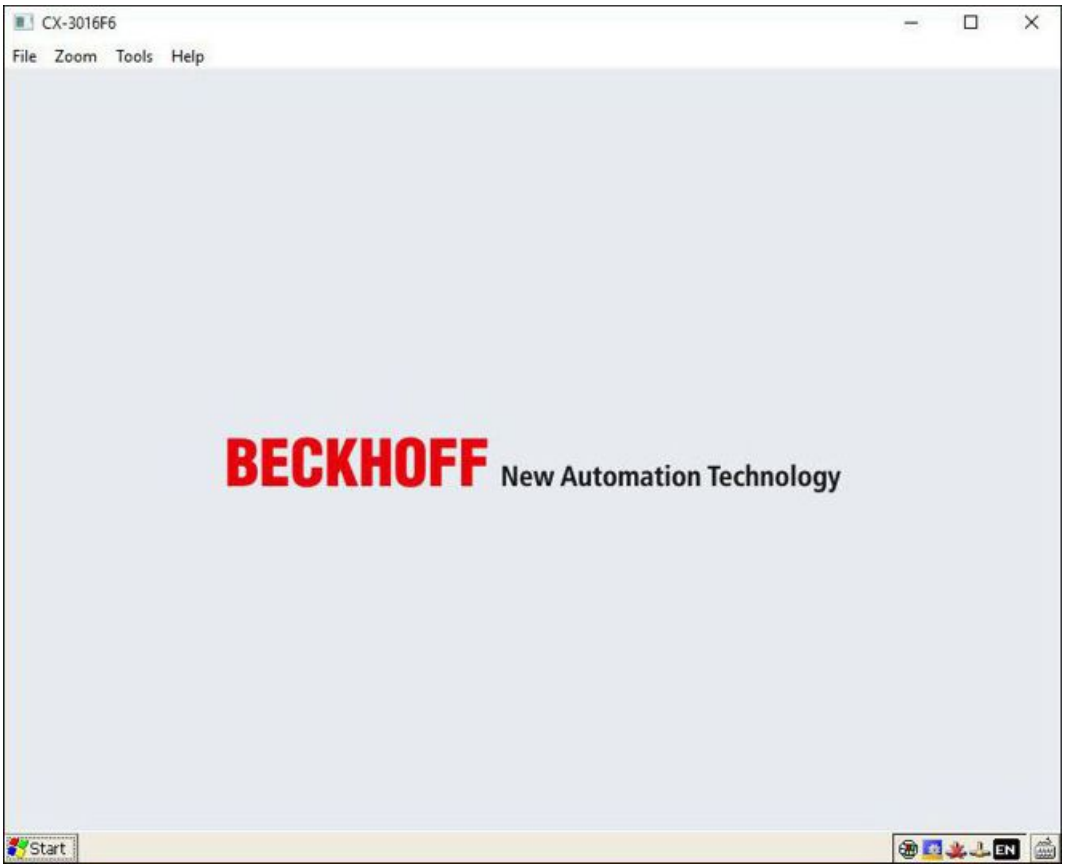
Start the remote connection as follows:

1. Unpack the zip file on the host PC and run *cerhost.exe*.
2. Click on **File** in the menu bar and then on **Connect**.



3. Enter the host name of the Embedded PC in the **Hostname** field.

⇨ The remote connection is started and the Windows Embedded CE 7 start screen appears.



6.4 TwinCAT

6.4.1 Connecting to the CX81xx

Before you can work with the CX81xx you must connect your local computer to the CX81xx (target system). Then you can search for devices such as EtherCAT terminals with the help of the IP address or the host name.

The local PC and the target system must be connected to the same network or directly to each other via an Ethernet cable. In TwinCAT a search can be performed for all devices in this way and project planning subsequently carried out.

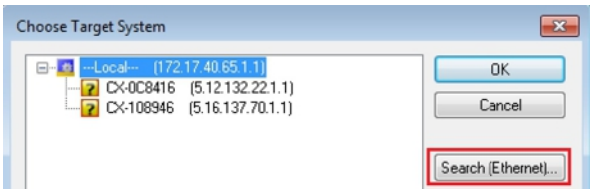
Requirements for this step:

- TwinCAT 3 must be in Config mode.
- IP address or host name of the Embedded PC.

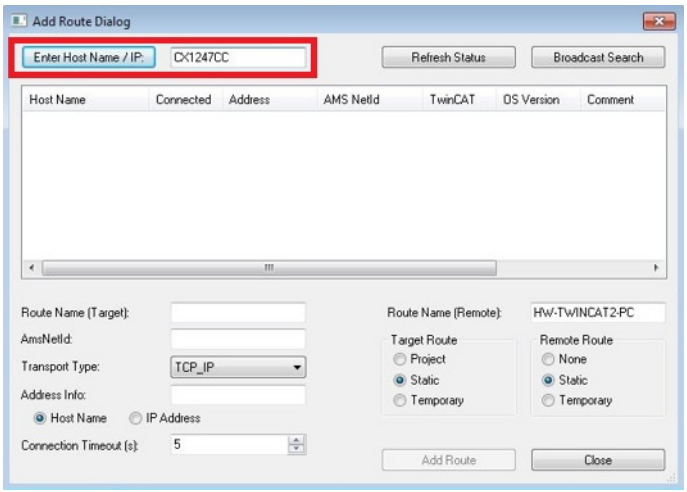
Establish a connection as follows:

1. In the menu at the top click on **File > New > Project** and create a new TwinCAT XAE project.
2. In the tree view on the left click on **SYSTEM**, and then **Choose Target**.

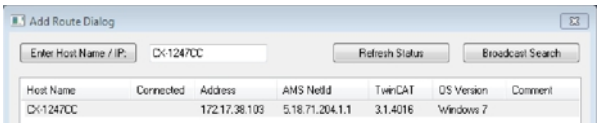
3. Click on **Search (Ethernet)**.



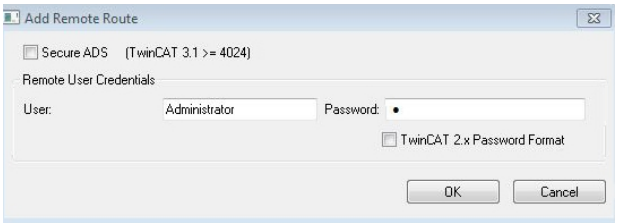
4. Type the host name or the IP address of the device into the **Enter Host Name / IP** box and press **[Enter]**.



5. Mark the device found and click on **Add Route**.



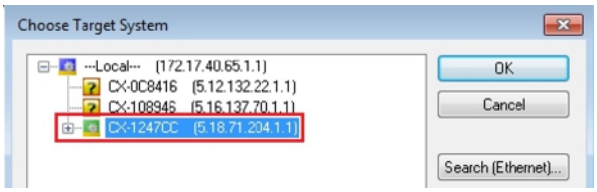
The **Logon Information** window appears.
Enter the user name and password for the CX in the **User Name** and **Password** fields and click **OK**.



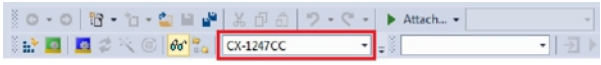
The following information is set as standard in CX devices:
User name: Administrator **Password:** 1

6. If you do not wish to search for any further devices, click on **Close** to close the **Add Route window**.
The new device is displayed in the **Choose Target System** window.

7. Select the device you want to specify as target system and click **OK**.



⇨ You have successfully searched for a device in TwinCAT and inserted the device as the target system.
The new target system and the host name are displayed in the menu bar.



Using this procedure you can search for all available devices and also switch between the target systems at any time. Next, you can append the device to the tree view in TwinCAT.

6.4.2 Scanning for devices

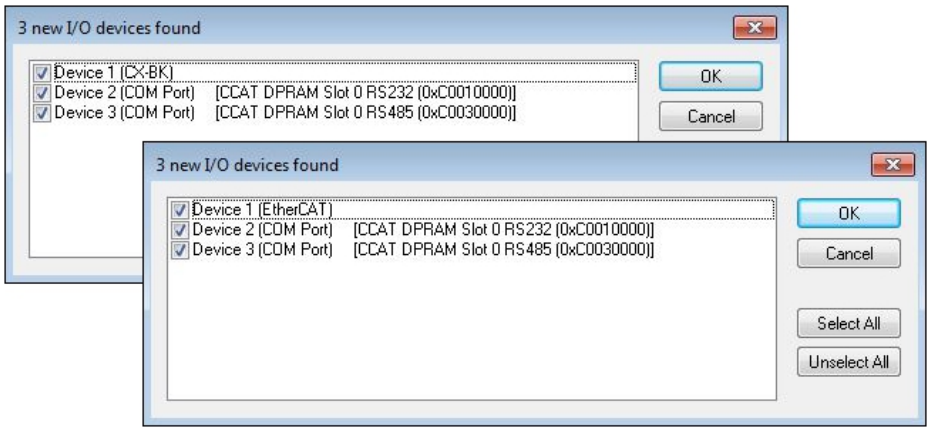
As soon as the CX81xx has been inserted as the target system in TwinCAT you can scan for further devices and in this way, for example, insert all the EtherCAT Terminals or Bus Terminals connected to the CX81xx into the TwinCAT tree view.

Requirements for this step:

- The CX81xx is connected as the target system to TwinCAT (see: Connecting to the CX81xx [▶ 34]).
- TwinCAT 3 is in ConfigMode.

Scan for devices as follows:

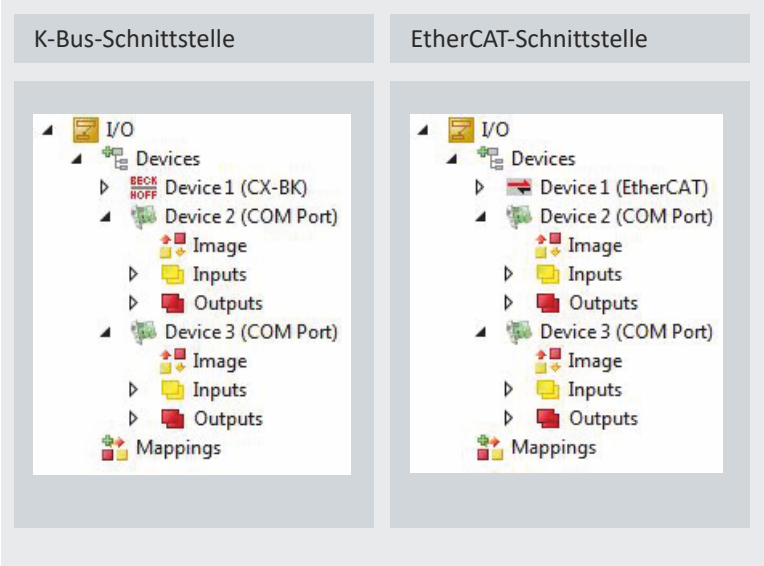
1. In the tree view on the left, right-click on **Devices** under **I/O**.
2. In the context menu click on **Scan**.
3. Select the devices you want to use and confirm the selection with **OK**.



Depending on whether EtherCAT terminals or Bus Terminals are connected to the CX81xx, the K-bus interface (Bus Terminals) or the EtherCAT interface (EtherCAT terminals) will be found.

4. Confirm the request with **Yes**, in order to look for boxes.
5. Confirm the request whether to enable FreeRun with **Yes**.

⇨ The devices are created in the tree view. Depending on the connected terminals, either a Bus Coupler or an EtherCAT coupler with the associated terminals will be displayed.



In the next step you can create a small program.

6.4.3 Creating a PLC project

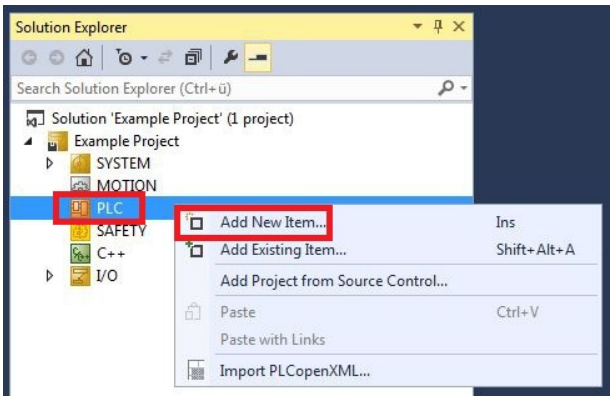
The next steps describe how to create a PLC project in TwinCAT and add it in the tree view.

Requirements for this step:

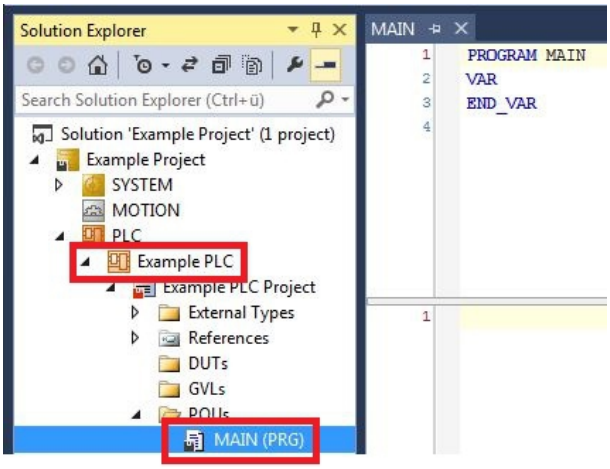
- A newly created TwinCAT XAE project.

Create a PLC project as follows:

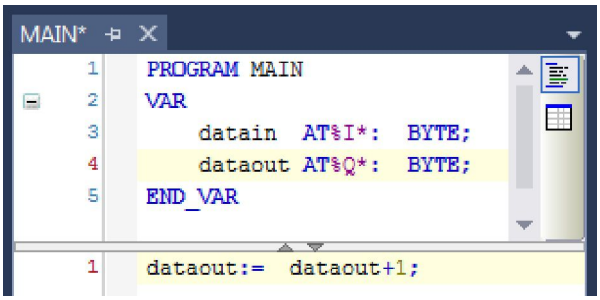
1. Right-click on **PLC** in the tree view.
2. In the context menu click on **Add New Item** and select the **Standard PLC Project**.



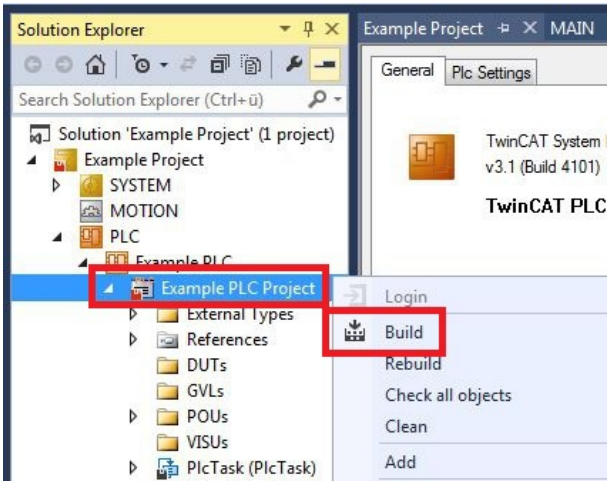
3. In the tree view click on the newly created PLC project, then double-click on **MAIN (PRG)** under **POUs**.



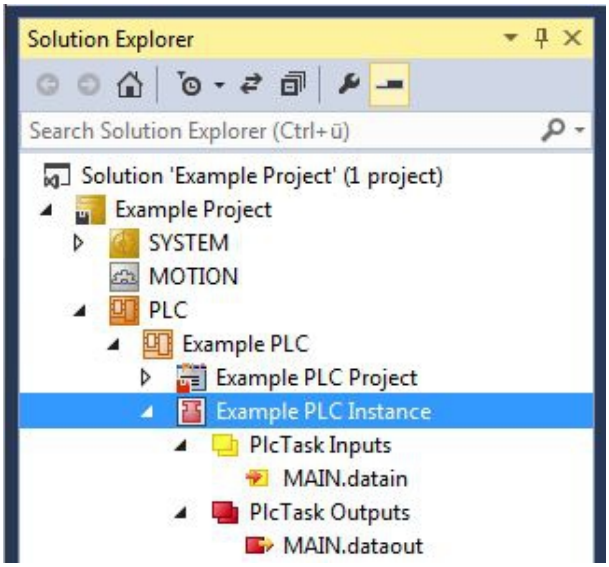
4. Write a small program, as shown in the diagram below.



5. In the tree view right-click on the PLC project, then click on **Build** in the context menu.



⇨ You have successfully created a PLC project and added the project in TwinCAT. A PLC instance is created with the variables for the inputs and outputs from the PLC project.



In the next step you can link the variables with the hardware.

6.4.4 Linking variables

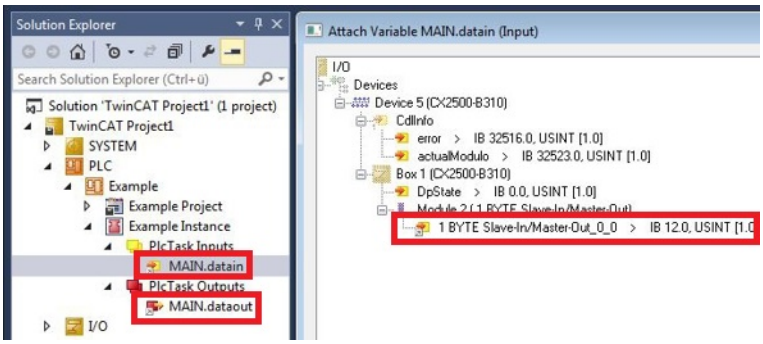
Once the PLC project has been successfully appended in TwinCAT you can link the newly created input and output variables from the PLC project with the inputs and outputs of your hardware.

Requirements for this step:

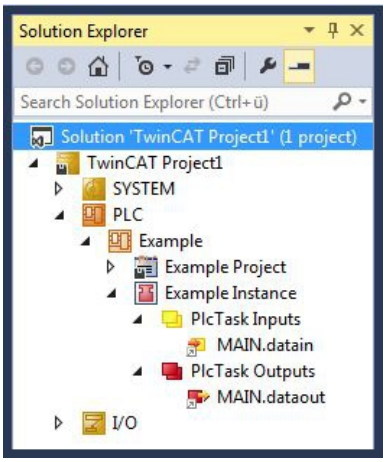
- An appended PLC project in TwinCAT.

Link the variables as follows:

1. Double-click on the input or output variables in the tree view under **PLC**.
   The **Attach Variable** window appears and shows which inputs or outputs can be linked with the variables from the PLC project.

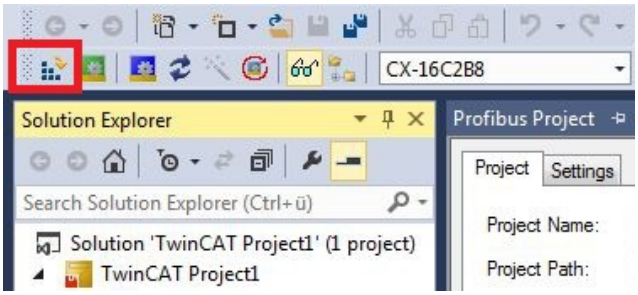


2. Double-click on the inputs or outputs of the hardware in the **Attach Variable** window.
   Link the input variables with the inputs and the output variables with the outputs of the hardware.



   Variables that are already linked are indicated with a small arrow icon in TwinCAT.

3. In the toolbar click on **Activate Configuration**.



4. Confirm the request whether TwinCAT is to start in Free Run mode with **Yes**.

⇨ You have successfully linked variables with the hardware. Use Activate Configuration to save and activate the current configuration.

Next, the configuration can be loaded to the Embedded PC in order to start TwinCAT automatically in Run mode and then start the PLC project.

6.4.5 Using the KS8000 protocol

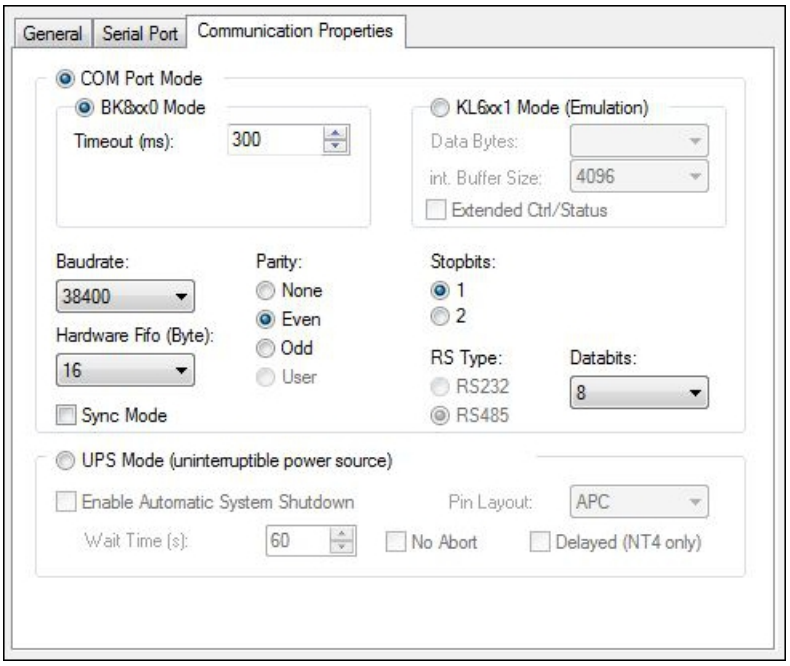
The KS8000 protocol is an open protocol from Beckhoff that can be used to address the BK8100 (RS232) or BK8000 (RS485) Bus Couplers. The protocol is already implemented and ready for use. The two serial RS232/RS485 interfaces of the Embedded PC CX8180 can be configured as BK8xx0 masters for connection to the Bus Couplers.

Requirements:

- The CX8180 Embedded PC has been selected as the target system.

Proceed as follows:

1. Click **Device 2** (RS232) or **Device 3** (RS485) in the tree view on the left.
2. Click the **Communication Properties** tab.



3. Click the BK8xx0 Mode option to activate the KS8000 protocol.

⇨ You have successfully activated the KS8000 protocol. Follow the same steps to deactivate the protocol.

6.4.6 Activating KL6xx1 mode

The KL6xx1 mode can be used for any serial protocols. Custom protocols or other open protocols can also be used.

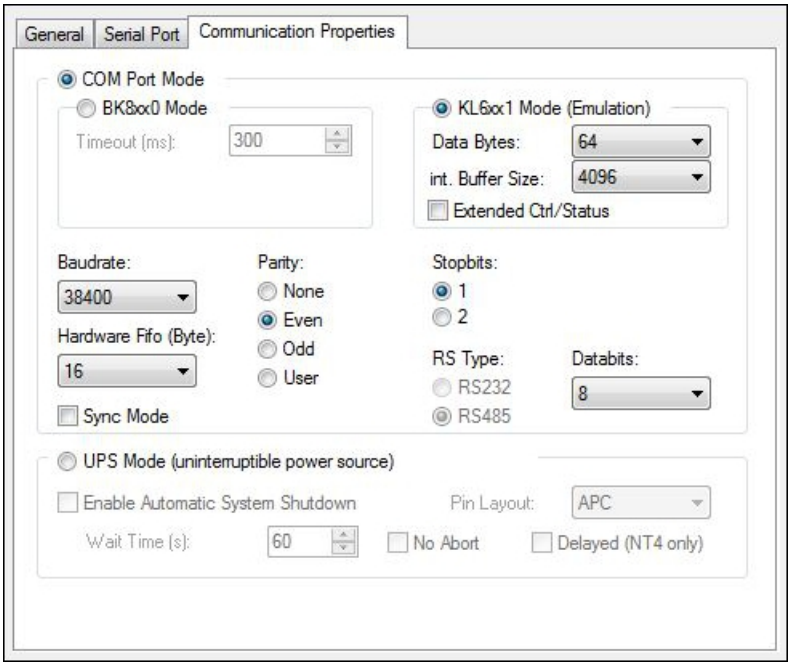
This mode behaves like other Beckhoff terminals with serial interface. It means that the programming and the interface is similar to a serial interface. The Beckhoff supplement blocks (such as ModbusRTU, COMLIB, ...) of the serial interface usually offer the 64-byte mode for the PC interface.

Requirements:

- The CX8180 Embedded PC has been selected as the target system.

Proceed as follows:

1. Click **Device 2** (RS232) or **Device 3** (RS485) in the tree view on the left.
2. Click the **Communication Properties** tab.



3. Click the option **KL6xx1 Mode (Emulation)** to use the CX8180 for all other protocols.

⇨ You have successfully activated the KL6xx1 mode. Follow the same steps to deactivate the KL6xx1 mode.

Programming

Alternatively, you can operate and program the interface yourself. The data structure consists of a control and status word (2 bytes) and a data array. The control byte is written by the PLC program and acknowledges the sending of data with the TR bit.

The data (bits 4 to 15) are transmitted when an edge change (toggle) is encountered. The RA bit acknowledges that the receive data have been read. This enables the interface to recognize that it can copy new data into the data array. The status word indicates how many data are valid. When a positive edge is encountered, the IR bit (Reset) clears the buffers and resets the interface. The interface acknowledges the command in the status word with the IA bit. Finally, the IR bit is set to "False" again.

Table 10: Control word structure.

Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Name	Length of data to be written (OL bits)												Reserve	IR bit	RA bit	TR bit

Table 11: Legend for the control word.

Bit	Name	Comment	
CW.3	Reserved	-	-
CW.2	IR (InitRequest)	1_{bin}	The controller requests initialization. The send and receive functions are blocked, the FIFO indicators are reset, and the interface is again initialized with the value. The interface acknowledges completion of the initialization via bit SW.2 (IA).
		0_{bin}	The controller once again requests the interface to prepare for serial data exchange.
CW.1	RA (ReceiveAccepted)	toggle	The controller acknowledges receipt of data by changing the state of this bit. Only then new data can be transferred from the interface to the controller.
CW.0	TR (TransmitRequest)	toggle	Via a change of state of this bit the controller notifies the interface that the DataOut bytes contain the number of bytes indicated via the OL bits. The interface acknowledges receipt of the data in the status word via a change of state of bit SW.0 (TA). Only now new data can be transferred from the controller to the interface.

Table 12: Status word structure.

Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Name	Length of data in the buffer ((IL bits)												BUF_F	IA bit	RR bit	TA bit

Table 13: Legend for status word

Bit	Name	Comment	
SW.3	BUF_F (buffer full)	1_{bin}	The receive FIFO is full. All further incoming data will be lost!
SW.2	IA (InitAccepted bit)	1_{bin}	Initialization was completed by the interface.
		0_{bin}	The interface is ready again for serial data exchange.
SW.1	RR (ReceiveRequest)	toggle	Via a change of state of this bit the interface notifies the controller that the DataIn bytes contain the number of bytes indicated via the IL bits. The controller has to acknowledge receipt of the data in the control word via a change of state of bit CW.1 (RA). Only then new data can be transferred from the interface to the controller.
SW.0	TA (TransmitAccepted)	toggle	The interface acknowledges receipt of data by changing the state of this bit. Only now new data can be transferred from the controller to the interface.

Sample

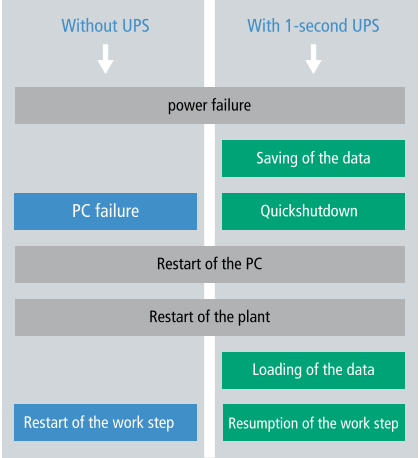
In this sample program, the operating principle is explained using the RS232 interface. Data is sent via the RS232 interface and received via a bridge (PIN 2 to 3).
Download: https://infosys.beckhoff.com/content/1033/CX8180_HW/Resources/zip/2495112075.zip

7 Programming

7.1 Seconds UPS

Loss of data

The use of the 1-second UPS outside of the documented possibilities can lead to loss or corruption of data.
Use only TwinCAT to control the 1-second UPS and save only persistent data with a maximum size of 1 MB.



The 1-second UPS is an UltraCap capacitor that continues to supply the processor with power in the event of a power failure.

During this period persistent data can be saved, which are available on switching on again.

Since the 1-second UPS is designed for the entire service life, the holding time is considerably longer with new devices. The capacitors age over the course of time and the holding time decreases. Therefore a maximum of 1 MB persistent data can be reliably saved over the entire service life.

Do not save any other data and do not use any other applications to control the 1-second UPS.

Please note that the 1-second UPS does not supply power to the K-bus or the E-bus and that their data may already be invalid when the 1-second UPS is activated. Also, the fieldbus system (or Ethernet) may not work or not work properly once the 1-second UPS was activated.

Storage location and names of the files:
The persistent data are saved by default in the TwinCAT boot directory:

Development environment	File path	File name
TwinCAT 3	\\TwinCAT\3.1\Boot\Plc	Port_85x.bootdata Port_85x.bootdata-old (backup) The x in the file name stands for the number of the runtime system.

Configure the 1-second UPS as follows in order to save persistent data:

- Declare important data such as counter values in the PLC as VAR PERSISTENT. Then call the function block FB_S_UPS_CX81xx cyclically in TwinCAT with the fastest task in order to control the 1-second UPS (see: Function block [▶ 45]).
- Select the mode in the function block in order to specify what should happen in the case of a power failure. Specify, for example, whether persistent data are saved and a quick shutdown is executed (see: Data types [▶ 47]).
- You can then check the validity of the variables and monitor whether the persistent variables are loaded without error (see: PlcAppSystemInfo).

Components	Version
TwinCAT on the development PC and on the control system	TwinCAT 3.1 Build 4020.16 or higher

Saving and loading persistent data

The persistent data are saved in the Port_85x.bootdata file on the MicroSD card. On starting the PLC the Port_85x.bootdata file is loaded from the MicroSD card, backed up there as Port_85x.bootdata_old (backup) and then deleted.

Another current Port_85x.bootdata file is not written until the system is shut down or the 1-second UPS is activated.

If no Port_85x.bootdata file exists when starting the Embedded PC, the persistent data are invalid and will be deleted (standard setting). The reason for this is that the 1-second UPS was activated before the TwinCAT PLC was started during startup of the Embedded PC. In this case no persistent data were saved, since the system was unable to ensure sufficient buffer time for saving the data.

Always call the function block from the PLC and always use the fastest task to do so. In the case of a power failure Beckhoff recommends not calling the rest of the application in order to ensure that sufficient time remains for writing the data.

```
IF NOT FB_S_UPS_CX81xx.bPowerFailDetect THEN
    ;//Call programs and function blocks
END_IF
```

The rest of the application influences the CPU load and the CPU load in turn affects the period during which the persistent data are written.

Loading a backup of the persistent data

A registry setting can be used to determine whether the backup file is deleted or used. The backup file is used by default (setting 0):

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Beckhoff\TwinCAT\Plc]"ClearInvalidPersistentData"= 0
```

If the backup file is to be deleted, the value of "ClearInvalidPersistentData" in the registry must be set to 1.

It is also possible in TwinCAT to specify on the left in the tree view under **PLC > CX8190** whether the backup file is to be used or not.

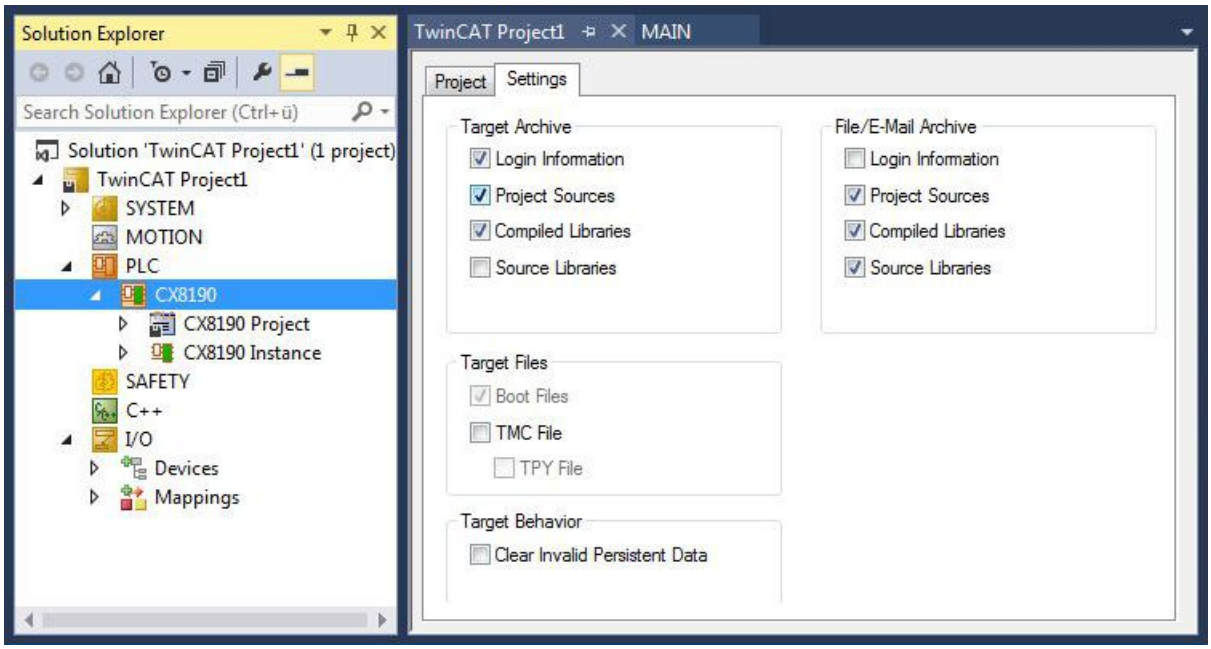


Fig. 12: Loading a backup of the persistent data. Settings in TwinCAT 3.

The backup files will be deleted if the option **Clear Invalid Persistent Data** is activated. Corresponds to registry entry 1.

7.1.1 Function block

FUNCTION_BLOCK FB_S_UPS_CX81xx

The function block FB_S_UPS_CX81xx can be used on CX81xx devices with second UPS, in order to control the second UPS from the PLC. This enables the persistent data to be saved according to the selected mode in the event of a power failure. The default input values of the FB_S_UPS_CX81xx should be retained.

The second UPS does not have sufficient capacity for bridging power failures. Saving can take place only on MicroSD cards.

The 1-second UPS can be used only for a few seconds in the event of a power failure in order, to save persistent data. The data must be saved in the fast “persistent mode” “SPDM_2PASS”, even though this can lead to real-time violations. Make sure you configure adequate router memory for saving the persistent data.

Regardless of the mode and therefore irrespective of whether data were saved or a quick shutdown was performed, the UPS switches off the mainboard after the capacitors have been discharged, even if the voltage has returned in the meantime.

NOTE

Loss of data

If other applications or the PLC keep further files open or write to them, file errors may occur if the 1-second UPS switches off the controller.

Function block modes

A QuickShutdown is performed automatically in the eSUPS_WrPersistData_Shutdown mode (standard setting) after the storage of the persistent data.

In the eSUPS_WrPersistData_NoShutdown mode only the persistent data are saved, no QuickShutdown is performed.

In eSUPS_ImmediateShutdown mode a quick shutdown is executed immediately, without saving data.

In the eSUPS_CheckPowerStatus mode only a check is performed as to whether a power failure has occurred. If this is the case, the function block only switches back to the PowerOK state after the expiry of tRecoverTime (10s).

VAR_INPUT

```
VAR_INPUT
    sNetID          : T_AmsNetId:= ''; (* '' = local netid *)
    iPLCPort        : UINT := 0; (* PLC Runtime System for writing persistent data *)
    tTimeout        : TIME := DEFAULT_ADS_TIMEOUT; (* ADS Timeout *)
    eUpsMode        : E_S_UPS_Mode := eSUPS_WrPersistData_Shutdown; (* UPS mode (w/
wo writing persistent data, w/wo shutdown) *)
    ePersistentMode : E_PersistentMode := SPDM_2PASS; (* mode for writing persistent data *)
    tRecoverTime    : TIME := T#10s; (* ON time to recover from short power failure in mode eSUPS_Wr
PersistData_NoShutdown/eSUPS_CheckPowerStatus *)
END_VAR
```

sNetID: AmsNetID of the controller (type: T_AmsNetID)

iPLCPort: Port number of the PLC runtime system (851 for the first PLC runtime system, 852 for the second PLC runtime system, …). If no port number is specified, iPLCPort is 0. The function block then automatically determines the port of the PLC runtime system.

tTimeout: Timeout for writing of the persistent data or the quick shutdown.

eUpsMode: Defines whether persistent data are to be written and whether a quick shutdown is to be executed. The default value is eSUPS_WrPersistData_Shutdown, i.e. a quick shutdown is executed automatically once the persistent data have been saved. (Type: E_S_UPS_Mode)

ePersistentMode: Mode for the writing of the persistent data. Default value is SPDM_2PASS.

tRecoverTime: Time after which the UPS returns to PowerOK state in UPS modes without quick shutdown. The tRecoverTime must be greater than the maximum charging time of the UPS, otherwise the UPS may discharge too much in the event of short, consecutive power failures, which could result in the charge being insufficient for storing the persistent data.

VAR_OUTPUT

```
VAR_OUTPUT
    bPowerFailDetect  : BOOL; (* TRUE while powerfailure is detected *)
    eState            : E_S_UPS_State := eSUPS_PowerOK; (* current ups state *)
END_VAR
```

bPowerFailDetect: TRUE during power failure. FALSE if the supply voltage is present.

eState: Internal state of the function block (type: E_S_UPS_State)

VAR_GLOBAL

```
VAR_GLOBAL
   eGlobalSUpsState  :  E_S_UPS_State; (*current ups state*)
END_VAR
```

eGlobalSUpsState: Internal state of the function block as global copy of VAR_OUTPUT

eState: For values see E_S_UPS_State

Requirements

Development environment	Target platform	Hardware	PLC libraries to be linked
TwinCAT v3.1	CX81xx	Seconds UPS	Tc2_SUPS

7.1.2 Data types

E_S_UPS_Mode

```
eSUPS_WrPersistData_Shutdown: Schreiben der persistenten Daten und dann QuickShutdown

eSUPS_WrPersistData_NoShutdown: Nur Schreiben der persistenten Daten (kein QuickShutdown)

eSUPS_ImmediateShutdown: Nur QuickShutdown (kein Schreiben der persistenten Daten)

eSUPS_CheckPowerStatus: Nur Status ermitteln (weder Schreiben der Persistenten Daten noch QuickShutd
own)
```

E_S_UPS_State

```
eSUPS_PowerOK:
in allen Modi: Versorgungsspannung ist OK

eSUPS_PowerFailure:
in allen Modi: Versorgungsspannung fehlerhaft (steht nur einen Zyklus an)

eSUPS_WritePersistentData:
im Modus eSUPS_WrPersistData_Shutdown: Schreiben der persistenten Daten ist aktiv
im Modus eSUPS_WrPersistData_NoShutdown: Schreiben der persistenten Daten ist aktiv

eSUPS_QuickShutdown:
im Modus eSUPS_WrPersistData_Shutdown: QuickShutdown ist aktiv
im Modus eSUPS_ImmediateShutdown: QuickShutdown ist aktiv

eSUPS_WaitForRecover:
im Modus eSUPS_WrPersistData_NoShutdown: Warten auf Wiederkehr der Spannung
im Modus eSUPS_CheckPowerStatus: Warten auf Wiederkehr der Spannung

eSUPS_WaitForPowerOFF:
im Modus eSUPS_WrPersistData_Shutdown: Warten auf das Abschalten durch die USV
im Modus eSUPS_ImmediateShutdown: Warten auf das Abschalten durch die USV
```

Requirements

Development environment	Target platform	Hardware	PLC libraries to be linked
TwinCAT v3.1	CX81xx	Seconds UPS	Tc2_SUPS

7.1.3 PlcAppSystemInfo

Each PLC contains an instance of type 'PlcAppSystemInfo' with the name '_AppInfo'.

The corresponding namespace is 'TwinCAT_SystemInfoVarList'. This must be specified for use in a library, for example.

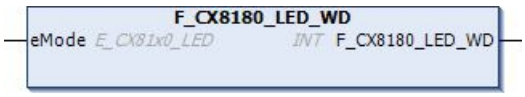
```
TYPE PlcAppSystemInfo
STRUCT
     ObjId              : OTCID;
     TaskCnt            : UDINT;
     OnlineChangeCnt    : UDINT;
     Flags              : DWORD;
     AdsPort            : UINT;
     BootDataLoaded     : BOOL;
     OldBootData        : BOOL;
     AppTimestamp       : DT;
     KeepOutputsOnBP    : BOOL;
     ShutdownInProgress : BOOL;
     LicensesPending    : BOOL;
     BSODOccured        : BOOL;

     TComSrvPtr         : ITComObjectServer;

     AppName            : STRING(63);
     ProjectName        : STRING(63);
END_STRUCT
END_TYPE
```

ObjId	Object ID of the PLC project instance
TaskCnt	Number of tasks in the runtime system
OnlineChangeCnt	Number of online changes since the last complete download
Flags	Reserved
AdsPort	ADS port of the PLC application
BootDataLoaded	PERSISTENT variables: LOADED (without error)
OldBootData	PERSISTENT variables: INVALID (the back-up copy was loaded, since no valid file was present)
AppTimestamp	Time at which the PLC application was compiled
KeepOutputsOnBP	The flag can be set and prevents that the outputs are zeroed when a breakpoint is reached. In this case the task continues to run. Only the execution of the PLC code is interrupted.
ShutdownInProgress	This variable has the value TRUE if a shutdown of the TwinCAT system is in progress. Some parts of the TwinCAT system may already have been shut down.
LicensesPending	This variable has the value TRUE if not all licenses that are provided by license dongles have been validated yet.
BSODOccured	This variable has the value TRUE if Windows is in a BSOD.
TComSrvPtr	Pointer to the TcCOM object server
AppName	Name generated by TwinCAT, which contains the port.
ProjectName	Name of the project

7.2 Control diagnostic LEDs

7.2.1 Function F_CX81x0_LED_WD

Since the CX8180 supports different protocols, the two LEDs WD and ERR on the CX8180 are not used by the firmware. This allows the user to create his own diagnosis messages. The LEDs can be used to indicate, for example, whether the CX8180 has received an IP address from the DHCP server or whether devices are exchanging data.



The function F_CX8180_LED_WD controls the WD LED on the CX8180. Various color and flashing modes can be used here. The possible LED colors are red and green.

VAR_INPUT

```
VAR_INPUT
    eMode       : E_CX81x0_LED;
END_VAR
```

eMode [▶ 50]: way in which the LED lights up.

VAR_OUTPUT

```
F_CX8180_LED_WD       : INT;
```

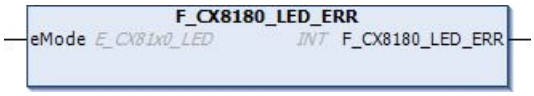
F_CX8180**_LED_WD**: -1, non-implemented flash code, 0 OK

Requirements

Development environment	Target platform	Hardware	PLC libraries to include
TwinCAT v3.1 Build 4022.30	ARM	CX8180	Tc2_SystemCX

7.2.2 Function F_CX81x0_LED_ERR

Since the CX8180 supports different protocols, the two LEDs WD and ERR on the CX8180 are not used by the firmware. This allows the user to create his own diagnosis messages. The LEDs can be used to indicate, for example, whether the CX8180 has received an IP address from the DHCP server or whether devices are exchanging data.



The function F_CX8180 CX8180 _LED_ERR controls the ERR LED on the CX8180. Various color and flashing modes can be used here. The possible LED colors are red and green.

VAR_INPUT

```
VAR_INPUT
    eMode       : E_CX81x0_LED;
END_VAR
```

eMode [▶ 50]: way in which the LED lights up.

VAR_OUTPUT

```
F_CX8180_LED_ERR      : INT;
```

F_CX8180**_LED_ERR**: -1, non-implemented flash code, 0 OK

Requirements

Development environment	Target platform	Hardware	PLC libraries to include
TwinCAT v3.1 Build 4022.30	ARM	CX8180	Tc2_SystemCX

7.2.3 Data type E_CX81x0_LED

```
TYPE E_CX81x0_LED : (
    eCX81x0_LED_GREEN_OFF,
    eCX81x0_LED_GREEN_ON,
    eCX81x0_LED_GREEN_FLASHING_Quick,
    eCX81x0_LED_GREEN_FLASHING_200ms,
    eCX81x0_LED_GREEN_FLASHING_500ms,
    eCX81x0_LED_GREEN_FLASHING_Pulse,
    eCX81x0_LED_RED_OFF:=20,
    eCX81x0_LED_RED_ON,
    eCX81x0_LED_RED_FLASHING_Quick,
    eCX81x0_LED_RED_FLASHING_200ms,
    eCX81x0_LED_RED_FLASHING_500ms,
    eCX81x0_LED_RED_FLASHING_Pulse,
    eCX81x0_LED_GREEN_RED_OFF:=100,
    eCX81x0_LED_GREEN_RED_FLASHING_200ms,
    eCX81x0_LED_GREEN_RED_FLASHING_500ms
);
END_TYPE
```

7.3 Function F_CX81xx_ADDRESS

This function reads the position of the address selection switch of the CX8180. One possible application is that you can activate different program parts in the PLC depending on the switch position.

VAR_INPUT

```
VAR_INPUT
    iCX_Typ       : INT;          (* Use product code without 'CX' e.g.: CX8180 -> 8180 *)
END_VAR
```

VAR_OUTPUT

```
F_CX80xx_ADDRESS      : INT;
```

F_CX80xx_ADDRESS : -1, non-implemented CX, address of the switch

Requirements

Development environment	Target platform	Hardware	PLC libraries to include
TwinCAT v3.1 Build 4022.30	ARM	CX8180	Tc2_SystemCX

7.4 Real Time Clock (RTC)

The real-time clock (RTC) is read out via the function blocks FB_LocalSystemTime and can be set with the function block NT_SetLocalTime (see TcUtilities.lib). The RTC is supplied by the battery and can thus continue to run in the power-off state.

Real-time clock (RTC) running too slow

TwinCAT uses its own real-time driver. This has the advantage that the quality of the real-time has been much improved, and the jitter of the task has been reduced to a minimum. This calls the operating system from TwinCAT. The RTC on the operating system is controlled via TwinCAT and must be called at certain times.

If the task cycle time on the TwinCAT side is very long and the CPU load is high, the operating system is no longer called with sufficient frequency. As a result, the RTC on the operating system is slow. If you have noticed that the RTC is slow and the time on the CX8180 is also slow, you can apply the following troubleshooting procedure.

Remedy

The call of the RTC is always linked to a reading of the hardware RTC. This has the disadvantage of a slightly higher CPU load, but since the CPU load is already high due to your application, the additional CPU load is negligible. Adjust the settings in the registry:

```
[HKEY_LOCAL_MACHINE\Platform]
  “SoftRTC"=dword:0
```

Table 14: Description of the SoftRTC registry key

Value	Description
0	The hardware RTC is always read out when the Windows time is requested.
1	The hardware RTC is read out once at startup. From then on, the Windows clock continues to run via the internal system tick.

Standard setting: "SoftRTC"=dword:1

8 Ethernet X001 Interface

8.1 Ethernet

Ethernet was originally developed by DEC, Intel and XEROX (as the "DIX" standard) for passing data between office devices. The term nowadays generally refers to the *IEEE 802.3 CSMA/CD* specification, published in 1985. Because of the high acceptance around the world this technology is available everywhere and is very economical. This means that it is easy to make connections to existing networks.

There are now a number of quite different transmission media: coaxial cable (10Base5), optical fiber (10BaseF) or twisted pairs (10BaseT) with shield (STP) or without shield (UTP). Using Ethernet, different topologies can be built such as ring, line or star.

Ethernet transmits Ethernet packets from a sender to one or more receivers. This transmission takes place without acknowledgement, and without the repetition of lost packets. To achieve reliable data communication, there are protocols, such as TCP/IP, that can run on top of Ethernet.

MAC-ID

The sender and receiver of Ethernet packets are addressed by means of the MAC-ID. The MAC-ID is a 6-byte identification code unique to every Ethernet device in the world. The MAC-ID consists of two parts. The first part (i.e. the first 3 bytes) is a manufacturer identifier. The identifier for Beckhoff is 00 01 05. The next 3 bytes are assigned by the manufacturer and implement a unique serial number. The MAC-ID can, for example, be used for the BootP protocol in order to set the TCP/IP number. This involves sending a telegram containing the information such as the name or the TCP/IP number to the corresponding node. You can read the MAC-ID with the KS2000 configuration software.

The Internet Protocol (IP)

The internet protocol (IP) forms the basis of this data communication. IP transports data packets from one device to another; the devices can be in the same network, or in different networks. IP here looks after the address management (finding and assigning MAC-IDs), segmentation and routing. Like the Ethernet protocol, IP does not guarantee that the data is transported - data packets can be lost, or their sequence can be changed.

TCP/IP was developed to provide standardized, reliable data exchange between any numbers of different networks. TCP/IP was developed to provide standardized, reliable data exchange between any numbers of different networks. Although the term is often used as a single concept, a number of protocols are layered together here: e.g. IP, TCP, UDP, ARP and ICMP.

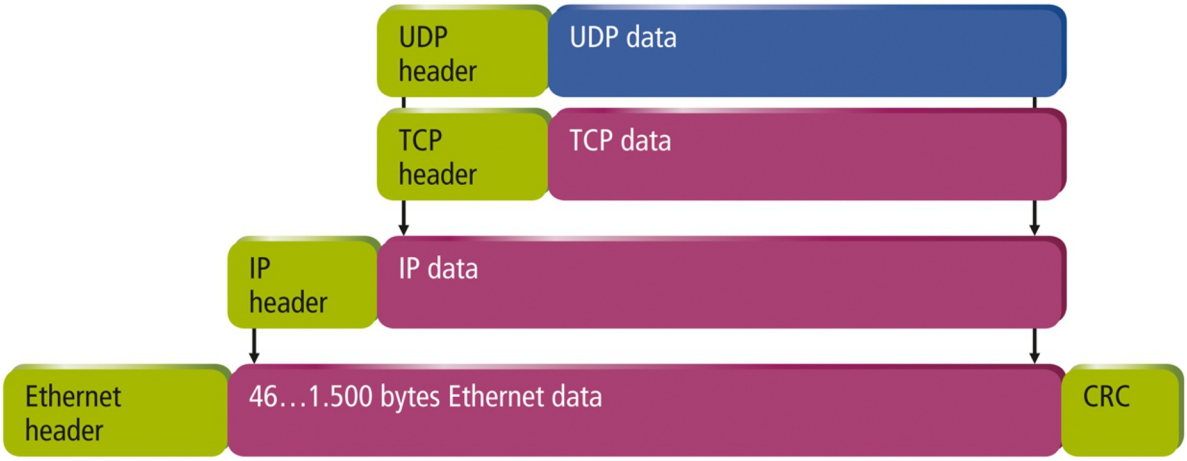


Fig. 13: Structure of the Ethernet protocol.

Transmission Control Protocol (TCP)

The Transmission Control Protocol (TCP) which runs on top of IP is a connection-oriented transport protocol. It includes error detection and handling mechanisms. Lost telegrams are repeated.

User Datagram Protocol (UDP)

UDP is connectionless transport protocol. It provides no control mechanism when exchanging data between sender and receiver. This results in a higher processing speed than, for example, TCP. Checking whether or not the telegram has arrived must be carried out by the higher-level protocol.

Protocols running on top of TCP/IP and UDP/IP

The following protocols can run on top of TCP/IP or UDP:

- ADS
- ModbusTCP

Both of these protocols are implemented in parallel on the Bus Coupler, so that no configuration is needed to activate the protocols.

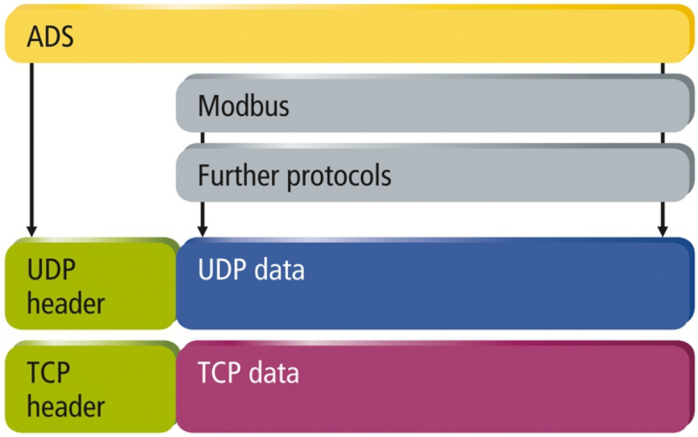


Fig. 14: Protocols running on top of TCP/IP and UDP/IP.

ADS can be used on top of either TCP or UDP, but ModbusTCP is always based on TCP/IP.

8.2 Topology example

With RS232, a direct connection (peer-to-peer) is only possible between two devices. With RS485 a line topology is possible. Limit the number of nodes to four devices.

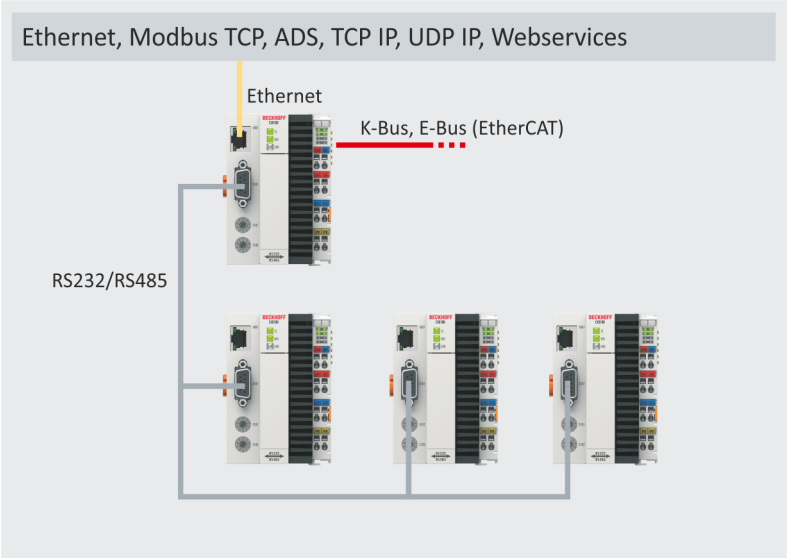


Fig. 15: Topology example: CX8180 connected to other CX8180 devices via RS485.

8.3 ADS-Communication

The ADS protocol (ADS: Automation Device Specification) is a transport layer within the TwinCAT system. It was developed for data exchange between the different software modules, for instance the communication between the NC and the PLC. This protocol enables communication with other tools from any point within the TwinCAT. If it is necessary to communicate with another PC or device, the ADS protocol is used on top of TCP/IP. Within a networked system it is thus possible to reach all data from any point.

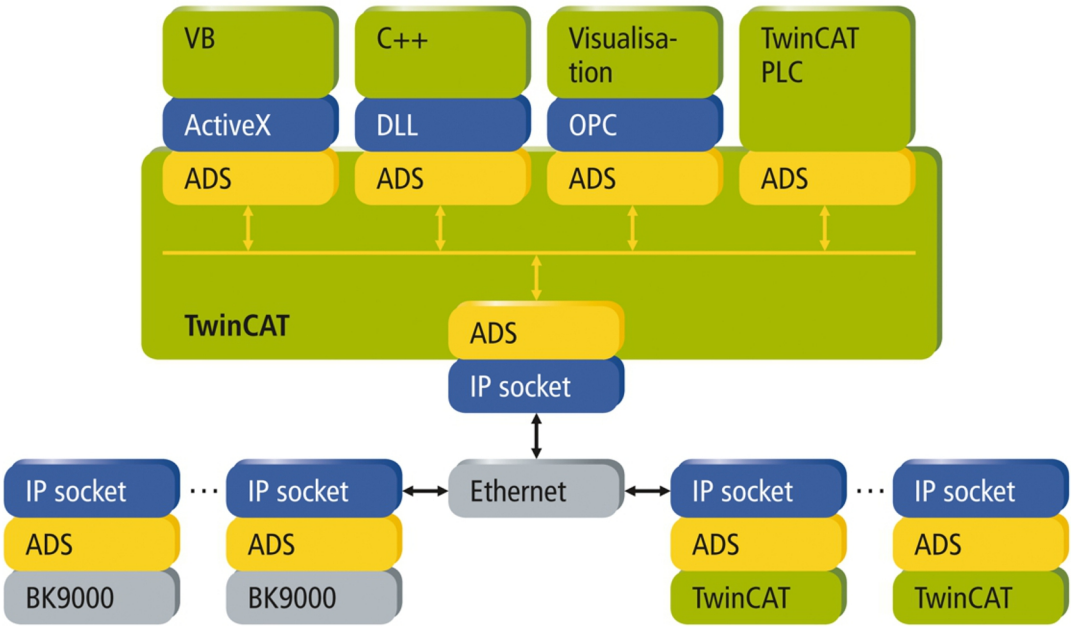


Fig. 16: The ADS protocol as a transport layer within TwinCAT.

The ADS protocol runs on top of the TCP/IP or UDP/IP protocols. It allows the user within the Beckhoff system to use almost any connecting route to communicate with all the connected devices and to parameterize them. Outside the Beckhoff system a variety of methods are available to exchange data with other software tools.

Software interfaces

ADS-OCX
The ADS-OCX is an Active-X component. It offers a standard interface to, for instance, Visual Basic, Delphi, etc.

ADS-DLL
You can link the ADS-DLL (DLL: Dynamic Link Library) into your C program.

OPC
The OPC interface is a standardized interface for communication used in automation technology. Beckhoff offer an OPC server for this purpose.

Protocol

The ADS functions provide a method for accessing the Bus Coupler information directly from the PC. ADS function blocks can be used in TwinCAT for this. The function blocks are contained in the *Tc2_System.lib* library. It is also equally possible to call the ADS functions from AdsOCX, ADSDLL or OPC.

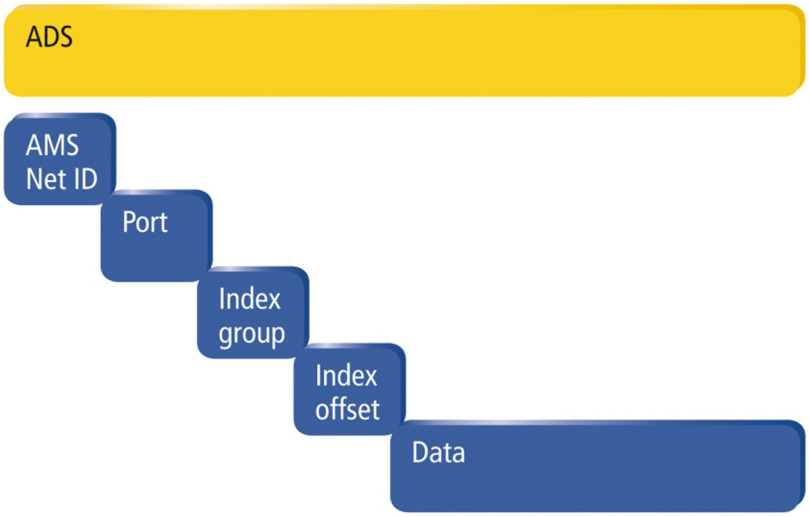


Fig. 17: Structure of the ADS communication.

AMSNetID
The AMSNetID provides a reference to the device that is to be addressed. This is taken from the MAC address of the first Ethernet port (X001) and is printed on the side of the CX80xx. For the AMSNetID the bytes 3..6 plus ".1.1" are typically used.
Example:
MAC address 00-01-**05-01-02-03**
AMSNetID 5.1.2.3.1.1

Port number
The port number distinguishes sub-elements in the connected device.
Port 851: local process data PLC runtime 1

Index group
The index group distinguishes different data within a port.

Index offset
Indicates the offset, from which reading or writing the byte is to start.

Len
Gives the length of the data, in bytes, that is to be read or written.

TCP port number
The TCP port number for the ADS protocol is 48898 or 0xBF02.

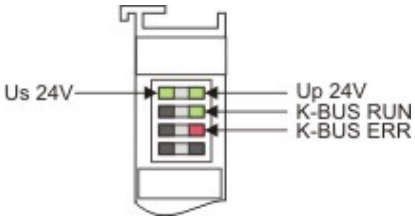
9 Error handling and diagnosis

9.1 Diagnostic LEDs

Display	LED	Meaning
	TC	TwinCAT Status LED: TwinCAT is in Run mode (green). TwinCAT is in Stop mode (red). TwinCAT is in Config mode (blue).
	WD	No function ex factory. The LED can be parameterized for user-specific diagnostic messages (see: F_CX8190_LED_WD function).
	ERR	Lights up red when switching on. Software is being loaded. Goes off if everything is OK. The LED can be parameterized for user-specific diagnosis messages (see: F_CX8190_LED_ERR function).

9.2 Power supply terminal LEDs in K-bus mode

The power supply unit checks the connected Bus Terminals for errors. The red LED "K-bus ERR" is off if no error is present. The red LED "K-bus ERR" flashes if Bus Terminal errors are present.

Display	LED	Meaning
	Us 24 V	Power supply for basic CPU module. The LED lights green if the power supply is correct.
	Up 24V	Power supply for terminal bus. The LED lights green if the power supply is correct.
	K-BUS RUN	Diagnostic K-bus. The green LED lights up in order to indicate fault-free operation. "Error-free" means that the communication with the fieldbus system is also running.
	K-BUS ERR	Diagnostic K-bus. The red LED flashes to indicate an error. The red LED blinks with two different frequencies.

The frequency and number of the flashes can be used to determine the error code and the error argument. An error is indicated by the "K-bus ERR" LED in a particular order.

Table 15: K-bus ERR LED, fault indication sequence through the LED.

Order	Meaning
Fast blinking	Starting the sequence
First slow sequence	**Error code**
No display	Pause, the LED is off
Second slow sequence	**Error code argument**

Count how often the red LED K-bus ERR flashes, in order to determine the error code and the error argument. In the error argument the number of pulses shows the position of the last Bus Terminal before the error. Passive Bus Terminals, such as a power feed terminal, are not included in the count.

Table 16: K-BUS ERR LED, fault description and troubleshooting.

Error code	Error code argument	Description	Remedy
Persistent, continuous flashing		EMC problems.	• Check power supply for undervoltage or overvoltage peaks. • Implement EMC measures. • If a K-bus error is present, it can be localized by a restart of the power supply (by switching it off and then on again)
3 pulses	0	K-bus command error.	• No Bus Terminal inserted. • One of the Bus Terminals is defective; halve the number of Bus Terminals attached and check whether the error is still present with the remaining Bus Terminals. Repeat this procedure until the faulty Bus Terminal has been found.
4 pulses	0	K-bus data error, break behind the power supply unit.	Check whether the Bus End Terminal 9010 is connected.
	n	Break behind Bus Terminal n	Check whether Bus Terminal n+1 after the power supply unit is connected correctly; replace if necessary.
5 pulses	n	K-bus error in register communication with Bus Terminal n.	Replace Bus Terminal at location n.
6 pulses	0	Error at initialization.	Replace Embedded PC.
	1	Internal data error.	Hardware reset of the Embedded PC (switch off and back on again).
	8	Internal data error.	Hardware reset of the Embedded PC (switch off and back on again).
7 pulses	0	Process data lengths of the set and actual configurations do not correspond.	Check the configuration and the Bus Terminals for consistency.

For some error the LED "K-BUS ERR" does not go out, even if the error was rectified. Switch the power supply for the power supply unit off and back on again to switch off the LED after the error has been rectified.

State variable

In TwinCAT there is a State variable under the Bus Coupler for K-bus diagnostics.

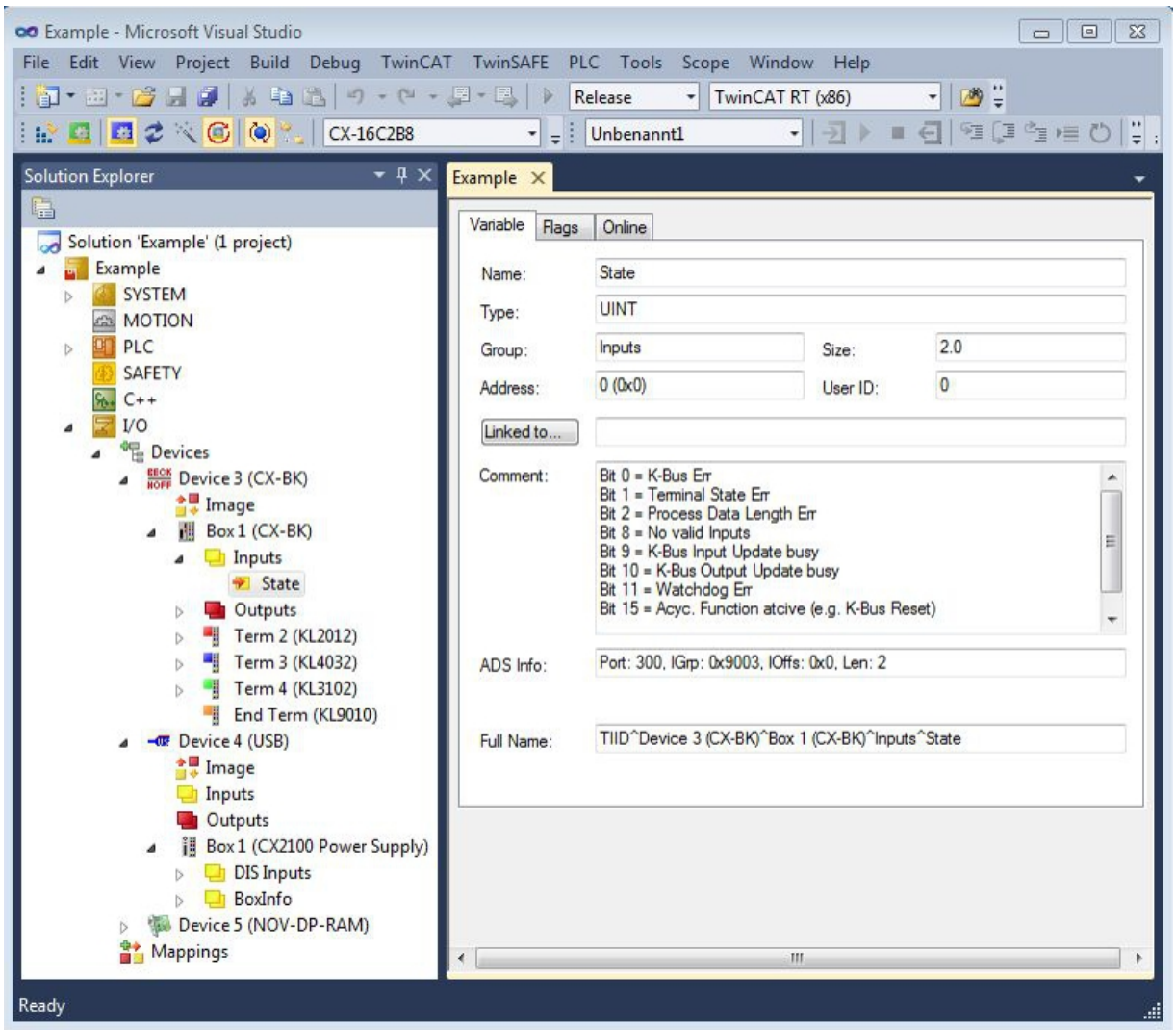


Fig. 18: Status variable for error handling and diagnostics under TwinCAT.

If the value is "0", the K-bus operates synchronous and without error. If the value is <> "0" there may be a fault, or it may only be an indication that the K-bus cycle is longer than the task. In which case it would no longer be synchronous with the task. The task time should be faster than 100 ms. We recommend a task time of less than 50 ms. The K-bus update time typically lies between one and five ms.

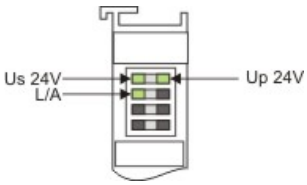
Table 17: Description of the State variable values.

Bit	Description
Bit 0	K-bus error.
Bit 1	Terminal configuration has changed since the start.
Bit 2	Process image lengths do not match.
Bit 8	(still) no valid inputs.
Bit 9	K-bus input update not yet complete.
Bit 10	K-bus output update not yet complete.
Bit 11	Watchdog.
Bit 15	Acyclic K-bus function active (e.g. K-bus reset).

If there is a K-bus error, this can be reset via the IOF_DeviceReset function block (in the TcIoFunctions.lib).

9.3 Power supply terminal LEDs in E-bus mode

The power supply unit checks the connected EtherCAT Terminals. The "L/A" LED is lit in E-bus mode. The "L/A" LED flashes during data transfer.

Display	LED		Meaning
	Us 24 V		Power supply for basic CPU module. The LED lights green if the power supply is correct.
	Up 24 V		Power supply for terminal bus. The LED lights green if the power supply is correct.
	L / A	off	E-bus not connected.
		on	E-bus connected / no data traffic.
		flashes	E-bus connected / data traffic on the E-bus.

10 Care and maintenance

10.1 Replace the battery

NOTE

Risk of explosion

An incorrectly inserted battery may explode and damage the Embedded PC.

Only use original batteries and ensure that the positive and negative poles are inserted correctly.

The battery must be replaced every 5 years. Spare batteries can be ordered from Beckhoff Service. A battery of type CR2032 is used for the Embedded PC.

Table 18: Technical data of the battery.

Battery type	Electrical properties (at 20 °C)		Standard load	Dimensions	
	nominal voltage	nominal capacity	continuous load	Diameter	Height
CR2032	3.0 V	225 mAh	0.20 mA	20.0 mm	3.20 mm

The battery compartment is below the front flap. The battery stores the time and date. The time and date are reset if the battery is removed.

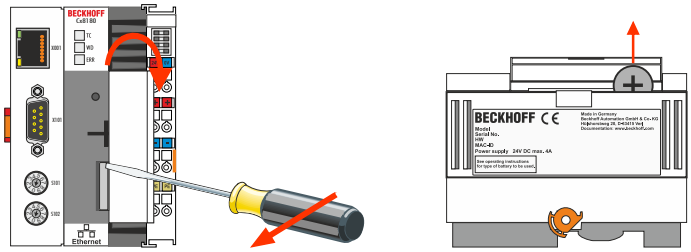
Be aware of this behavior for your hardware and software configuration and reset the time and date after a battery change.

Requirements:

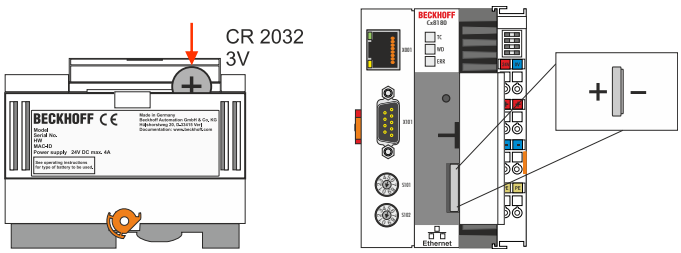
- The Embedded PC is switched off.

Replace the battery as follows:

1. Open the front flap.
2. Apply the screwdriver below or above the battery and prize the battery carefully out of the battery compartment.



3. Push the new battery into the battery compartment. The plus pole points to the left towards the Ethernet interfaces.



⇨ The battery change is complete. Close the front flap and reset the date and time.

11 Technical data

Table 19: Technical data, dimensions and weights.

	CX8180
Dimensions (W x H x D)	71 mm x 100 mm x 73 mm
Weight	230 g

Table 20: Technical data, general data.

Technical data	CX8180
Processor	ARM Cortex™-A9, 800 MHz 32-bit
Main memory	512 MB DDR3-RAM
Flash memory	MicroSD card (ATP) 512 MB (optionally 1, 2, 4, 8 GB)
Interfaces	1 x RJ45 10/100 MBit/s
Bus interface	D-Sub socket, 9-pin, 1 x RS232, 1 x RS485
Protocol	Serial communication
Persistent memory	1-second UPS integrated
Power supply	24 V_{DC} (-15 %/+20 %)
Max. power consumption	4 W
Max. power consumption (with UPS charging)	9 W
Dielectric strength	500 V (supply / internal electronics)
Operating system	Microsoft Windows Embedded Compact 7
Control software	TwinCAT 3, licenses not included
Diagnostic LED	1 x TC Status, 1 x WD LED, 1 x ERR LED
Clock	internal battery-backed clock (RTC) for time and date (battery exchangeable)
Approvals	CE, UL

Table 21: Technical data, I/O terminals.

Technical data	CX8180
I/O connection	via power supply terminal (E-bus or K-bus, automatic recognition)
Power supply for I/O terminals	max. 2 A
Power contacts current loading	max. 10 A
Process data on the K-bus	max. 2 kB in and 2 kB out
max. number of terminals (K-bus)	64 (255 with K-bus extension)
max. number of terminals (E-bus)	up to 65534 terminals.

Table 22: Technical data, environmental conditions.

Technical data	CX8180
Ambient temperature during operation	-25 °C to +60 °C
Ambient temperature during storage	-40 °C ... +85 °C see notes under: Transport and storage [▶ 10]
Relative humidity	95% no condensation
Vibration resistance	conforms to EN 60068-2-6
Shock resistance	conforms to EN 60068-2-27
EMC immunity	conforms to EN 61000-6-2
EMC emission	conforms to EN 61000-6-4
Protection class	IP20

Table 23: Technical data, Ethernet interface X001.

Technical data	Description
Data transfer medium	4 x 2 twisted pair copper cables category 5 (100 MBit/s)
Cable length	100 m from switch to CX8180
Data transfer rate	10/100 MBit/s
Topology	star wiring
Protocols	all non-real-time-capable protocols that are based on TCP or UDP and require no real-time extension

Table 24: Technical data, X101 RS232/RS485 interface.

Technical data	RS232
Number of I/O modules	1
Data transfer medium	shielded copper cable, 3 x 0.25 mm^2
Cable length	15 m
Data transfer rate	300 Baud ... 115 kBaud
Topology	Direct connection (peer-to-peer)

Technical data	RS485
Number of I/O modules	4
Data transfer medium	shielded copper cable, 3 x 0.25 mm^2
Cable length	200 m
Data transfer rate	300 Baud ... 115 kBaud
Topology	Line topology

12 Appendix

12.1 Certification

12.1.1 FCC

FCC Approvals for the United States of America

FCC: Federal Communications Commission Radio Frequency Interference Statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

FCC Approval for Canada

FCC: Canadian Notice

This equipment does not exceed the Class A limits for radiated emissions as described in the Radio Interference Regulations of the Canadian Department of Communications.

12.2 Support and Service

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

Beckhoff's branch offices and representatives

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

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

You will also find further documentation for Beckhoff components there.

Castel Maggiore (BO) 40013
Via Lirone 60/i

e-mail: info@eurotecsrl.com
Tel: +39 051/6321595
Website: www.eurotecsrl.com

C.F.: 04103780377
P.I.: 01596581205