

Documentation | EN

# BC3150

Bus Terminal Controller for PROFIBUS DP





# Table of contents

<b>1</b>	<b>Foreword .....</b>	<b>5</b>
1.1	Notes on the documentation.....	5
1.2	Safety instructions .....	6
1.3	Documentation issue status .....	7
<b>2</b>	<b>Product overview.....</b>	<b>8</b>
2.1	Bus Terminal Controllers of the BCxx50 series .....	8
2.2	BC3150 - Introduction.....	9
2.3	Technical data .....	10
2.3.1	Technical data - BC3150 .....	10
2.3.2	Technical data - PROFIBUS-DP.....	11
2.3.3	Technical data - PLC .....	11
2.4	The principle of the Bus Terminal.....	12
2.5	The Beckhoff Bus Terminal system .....	12
<b>3</b>	<b>Mounting and wiring.....</b>	<b>15</b>
3.1	Instructions for ESD protection .....	15
3.2	Mounting.....	16
3.2.1	Dimensions .....	16
3.2.2	Installation on mounting rails .....	17
3.2.3	Disposal .....	17
3.3	Wiring.....	18
3.3.1	Potential groups, insulation testing and PE .....	18
3.3.2	Power supply .....	20
3.3.3	PROFIBUS Connection .....	22
3.3.4	PROFIBUS cabling .....	23
3.4	ATEX - Special conditions (extended temperature range) .....	26
3.5	Continuative documentation for ATEX and IECEx .....	27
<b>4</b>	<b>Parameterization and Commissioning .....</b>	<b>28</b>
4.1	Start-up behavior of the Bus Terminal Controller .....	28
4.2	Configuration .....	29
4.2.1	Overview.....	29
4.2.2	Creating a TwinCAT configuration.....	31
4.2.3	Downloading a TwinCAT configuration.....	32
4.2.4	Uploading a TwinCAT configuration .....	33
4.2.5	Resources in the Bus Terminal Controller .....	35
4.2.6	ADS connection via serial interface .....	37
4.2.7	PROFIBUS .....	39
4.2.8	K-bus .....	48
4.2.9	PLC.....	50
4.2.10	Setting the slave address .....	53
4.3	Configuration software KS2000.....	54
4.4	Programming cable .....	55
<b>5</b>	<b>Programming .....</b>	<b>56</b>
5.1	BCxx50 PLC features.....	56

5.2	TwinCAT PLC .....	56
5.3	TwinCAT PLC - Error codes .....	57
5.4	Remanent data .....	60
5.5	Allocated flags .....	61
5.6	Local process image in delivery state .....	61
5.7	Mapping the Bus Terminals .....	63
5.8	Local process image in the TwinCAT configuration .....	63
5.9	Creating a boot project .....	64
5.10	Communication between TwinCAT and BX/BCxx50 .....	65
5.11	Up- and downloading of programs .....	66
5.12	Libraries .....	70
5.12.1	Libraries overview .....	70
5.12.2	TcBaseBX .....	71
5.13	Program transfer .....	75
5.13.1	Program transfer via the serial interface .....	75
5.13.2	Program transfer via PROFIBUS .....	77
5.14	PROFIBUS process image .....	78
<b>6</b>	<b>PROFIBUS .....</b>	<b>80</b>
6.1	System introduction .....	80
6.1.1	Fieldbus Overview .....	80
6.1.2	PROFIBUS DP .....	83
6.1.3	PROFIBUS DPV1 .....	85
6.1.4	Plug connectors, cables and switches .....	86
6.1.5	Topology .....	88
<b>7</b>	<b>Error handling and diagnosis .....</b>	<b>90</b>
7.1	Diagnostics .....	90
7.2	Diagnostic LEDs .....	91
<b>8</b>	<b>Appendix .....</b>	<b>94</b>
8.1	First steps with the BC3150 .....	94
8.2	Switching between controllers .....	96
8.3	Firmware Update .....	98
8.4	Test standards for device testing .....	101
8.5	Bibliography .....	101
8.6	List of Abbreviations .....	102
8.7	Support and Service .....	103

# 1 Foreword

## 1.1 Notes on the documentation

### Intended audience

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

It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.

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

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

### Disclaimer

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

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

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

### Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH. Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

### Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents: EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702 with corresponding applications or registrations in various other countries.



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

### Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

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

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

## 1.2 Safety instructions

### Safety regulations

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

### Exclusion of liability

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

### Personnel qualification

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

### Description of instructions

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

#### **DANGER**

##### **Serious risk of injury!**

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

#### **WARNING**

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

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

#### **CAUTION**

##### **Personal injuries!**

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

#### **NOTE**

##### **Damage to environment/equipment or data loss**

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



##### **Tip or pointer**

This symbol indicates information that contributes to better understanding.

### 1.3 Documentation issue status

Version	Comment
2.2.0	<ul style="list-style-type: none"> <li>• Technical data updated</li> <li>• Ex marking added to technical data</li> <li>• Chapter <i>Instructions for ESD protection</i> added</li> <li>• Chapter <i>Disposal</i> added</li> <li>• New title page</li> </ul>
2.1.0	<ul style="list-style-type: none"> <li>• PROFIBUS connection corrected</li> </ul>
2.0.0	<ul style="list-style-type: none"> <li>• Migration</li> </ul>
1.1.0	<ul style="list-style-type: none"> <li>• Notes to meet the UL requirements added.</li> </ul>
1.0.1	<ul style="list-style-type: none"> <li>• Chapter <i>Creating a boot project</i> corrected</li> <li>• Minor routine corrections (typing errors, spelling, etc.)</li> <li>• English version available</li> </ul>
1.0.0	<ul style="list-style-type: none"> <li>• First public issue (only in German)</li> </ul>

#### Firmware BC3150

For updating your firmware you need a [serial cable \[► 55\]](#), the KS2000 configuration software, or the firmware update program.

Firmware	Comment
0xB3	Firmware-Version 0xB3

The firmware and hardware versions (delivery state) can be found on the sticker on the underside of the Bus Terminal Controller.

## 2 Product overview

### 2.1 Bus Terminal Controllers of the BCxx50 series

Bus Terminal Controllers are Bus Couplers with integrated PLC functionality. The BCxx50 Bus Terminal Controllers have a fieldbus interface, are intelligent slaves and can be used as decentralized intelligence within the system. They are located in a cost-optimized and compact housing. In contrast to the BCxx00 range, the BCxx50 range supports up to 255 Bus Terminals via the K-Bus extension.

The Bus Terminal Controller is programmed using the TwinCAT programming system according to IEC 61131-3. The BCxx50 configuration/programming interface is used for loading the PLC program. If the TwinCAT software PLC is in use, the PLC program can also be loaded via the fieldbus.

The inputs and outputs of the connected Bus Terminals are assigned in the default setting of the mini-PLC. Each individual Bus Terminal can be configured in such a way that it exchanges data directly through the fieldbus with the higher-level automation device. Similarly, pre-processed data can be exchanged between the Bus Terminal Controller and the higher-level controller via the fieldbus.

#### Fieldbus interface

The variants of the BCxx50 series Bus Terminal Controllers differ in terms of their fieldbus interfaces. Various versions cover the most important fieldbus systems:

- [BC3150](#): PROFIBUS DP
- [BC5150](#): CANopen
- [BC5250](#): DeviceNet
- [BC8050](#): RS485, various protocols
- [BC8150](#): RS232, various protocols

#### Programming

The BCxx50 devices are programmed according to the powerful IEC 61131-3 standard. Like for all other BECKHOFF controllers, the TwinCAT automation software is the basis for parameterization and programming. Users therefore have the familiar TwinCAT tools available, e.g. PLC programming interface, System Manager and TwinCAT Scope. Data is exchanged optionally via the serial interface (COM1) or via the fieldbus through Beckhoff PC FCxxx fieldbus cards.

#### Configuration

The configuration is also carried out using TwinCAT. The fieldbus interface can be configured and parameterized via the System Manager. The System Manager can read all connected devices and Bus Terminals. After the parameterization, the configuration is saved on the BCxx50 via the serial interface. The configuration thus created can be accessed again later.



## 2.2 BC3150 - Introduction

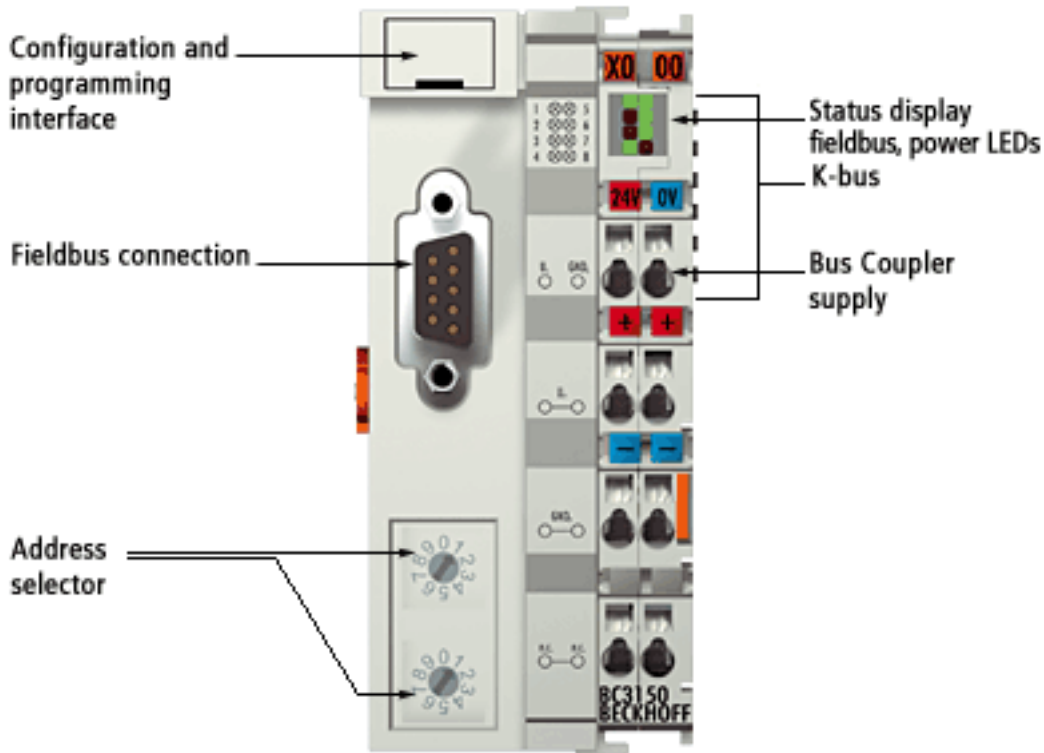


Fig. 1: BC3150 - Bus Terminal Controller for PROFIBUS

The BC3150 Bus Terminal Controller in a compact housing is a Bus Coupler with integrated PLC functionality and has a fieldbus interface for PROFIBUS. It is an intelligent slave and can be used as distributed intelligence in the PROFIBUS system. In the BC3150, one unit consists of the controller, between 1 and 64 terminals, and a bus end terminal. In addition, the BC3150 supports up to 255 Bus Terminals via the K-bus extension.

The Bus Terminal Controller offers automatic baud rate detection up to 12 Mbaud and two address selection switches for address assignment.

## 2.3 Technical data

### 2.3.1 Technical data - BC3150

Technical data	BC3150
Processor	16 bit micro-controller
Diagnostic LEDs	2 x power supply, 2 x K-Bus
Configuration and programming software	TwinCAT PLC
Fieldbus interface	PROFIBUS DP
Serial interface	COM1 (RS232 for configuration and programming, automatic baud rate detection 9600/19200/38400 baud)
Number of bus terminals	64 (255 with K-bus extension)
Digital peripheral signals	2040 inputs/outputs
Analog peripheral signals	1024 inputs/outputs
Configuration possibility	via TwinCAT or the controller
Maximum fieldbus byte number	depending on fieldbus
Maximum number of bytes - PLC	2048 bytes of input data, 2048 bytes of output data
Bus connection	D-sub, 9-pin

Versorgung	BC3150
Power supply (Us)	24 V <sub>DC</sub> (-15% /+20%)
Input current (Us)	60 mA + (total K-bus current)/4
Starting current	approx. 2.5 x continuous current
K-bus current (5 V)	up to hardware version 04: maximum 1000 mA from hardware version 05: maximum 1750 mA
Power contact voltage (Up)	maximum 24 V <sub>DC</sub>
Power contact current load (Up)	maximum 10 A
Recommended back-up fuse (Up)	≤10 A
Dielectric strength	500 V (power contact/supply voltage/Ethernet/fieldbus)

#### ⚠ CAUTION



#### UL-Requirements

For power supply of the Bus Terminal Controller (Us) use a 4 A fuse or an *NEC Class 2* power supply to meet the UL requirements!

Technical data	BC3150
Weight	approx. 100 g
Dimensions (W x H x D)	approx. 44 mm x 100 mm x 68 mm
Permissible ambient temperature range during operation	-25°C ... +60°C
Permissible ambient temperature range during storage	-40°C ... +85°C
Relative humidity	95% no condensation
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4
Installation position	variable
Protection class	IP20
Approvals/markings*	CE, UKCA, cULus, EAC, ATEX [▶ 26]

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

**Ex marking**

Standard	Marking
ATEX	II 3 G Ex nA IIC T4 Gc

**2.3.2 Technical data - PROFIBUS-DP**

System data	PROFIBUS (BC3150)
Number of I/O modules	126 (BC3150 max. 99 devices)
Number of I/O points	depending on controller
Data transfer medium	shielded copper cable 2 x 0.25 mm <sup>2</sup> , cable type A according to EN 50 170
Segment length	up to 1200 m
Number of segments	3
Data transfer rate	up to 12 MBaud
Topology	RS485 line, optical fiber ring
Transmission time	approx. 0.5 ms with 10 stations for 32 bits input/output at 12 MBaud

**2.3.3 Technical data - PLC**

PLC data	BCxx5x
Programmability	via serial programming interface or via the fieldbus
Program memory	48 kbyte
Source code memory	128 kbyte
Data memory	32 kbyte
Remanent flags	2 kbyte
PLC cycle time	Approx. 3.0 ms for 1000 IL commands (without I/O cycle)
Programming languages	IEC 6-1131-3 (IL, LD, FBD, ST, SFC)
Runtime	1 PLC task
Online Change	Yes
Up/Down Load Code	Yes/Yes

## 2.4 The principle of the Bus Terminal

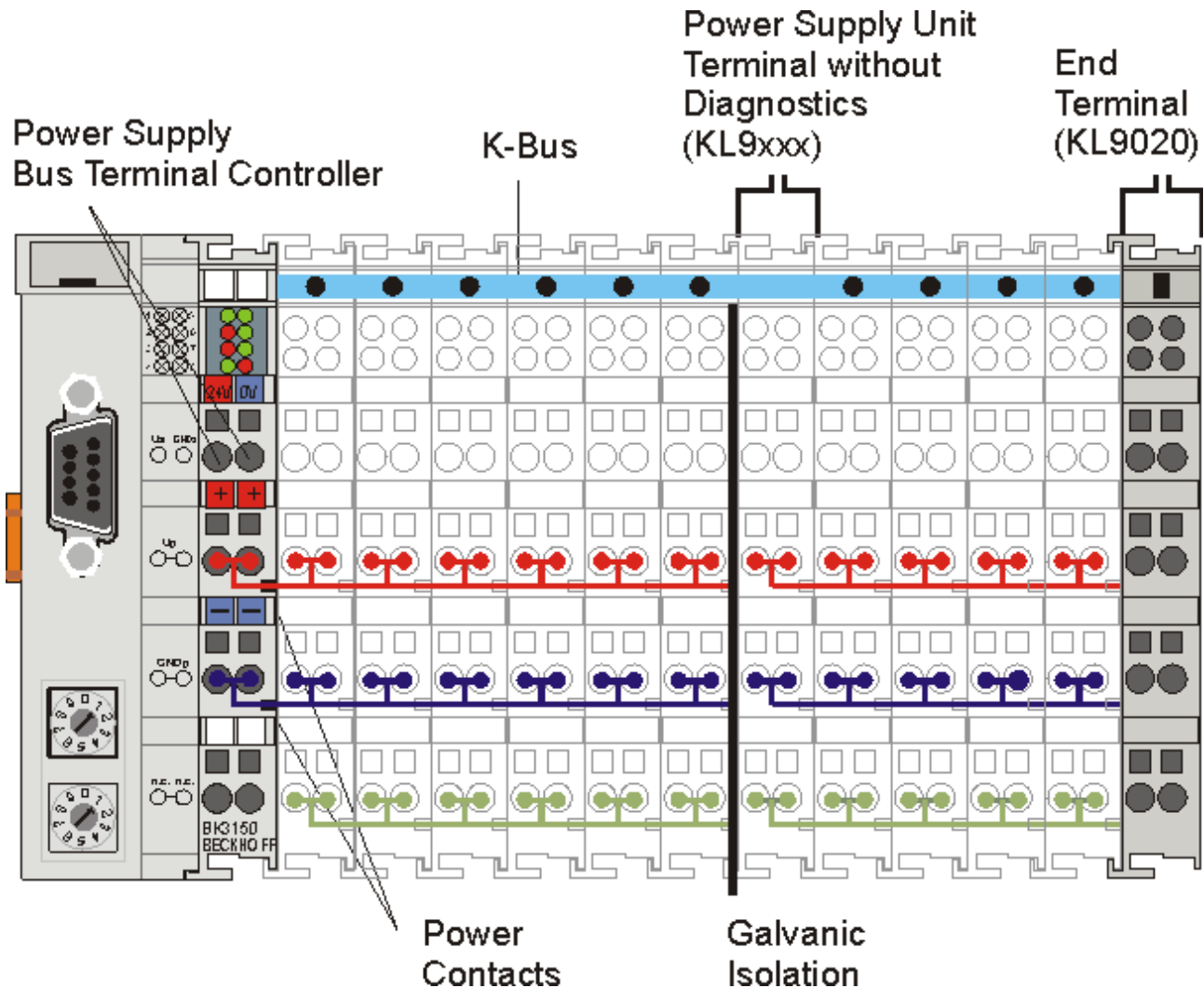


Fig. 2: The principle of the Bus Terminal

## 2.5 The Beckhoff Bus Terminal system

### Up to 256 Bus Terminals, with 1 to 16 I/O channels per signal form

The Bus Terminal system is the universal interface between a fieldbus system and the sensor / actuator level. A unit consists of a Bus Coupler as the head station, and up to 64 electronic series terminals, the last one being an end terminal. Up to 255 Bus Terminals can be connected via the K-Bus extension. For each technical signal form, terminals are available with one, two, four or eight I/O channels, which can be mixed as required. All the terminal types have the same mechanical construction, so that difficulties of planning and design are minimized. The height and depth match the dimensions of compact terminal boxes.

### Decentralized wiring of each I/O level

Fieldbus technology allows more compact forms of controller to be used. The I/O level does not have to be brought to the controller. The sensors and actuators can be wired decentrally, using minimum cable lengths. The controller can be installed at any location within the plant.

### Industrial PCs as controllers

The use of an Industrial PC as the controller means that the operating and observing element can be implemented in the controller's hardware. The controller can therefore be located at an operating panel, in a control room, or at some similar place. The Bus Terminals form the decentralized input/output level of the

controller in the control cabinet and the subsidiary terminal boxes. The power sector of the plant is also controlled over the bus system in addition to the sensor/actuator level. The Bus Terminal replaces the conventional series terminal as the wiring level in the control cabinet. The control cabinet can have smaller dimensions.

### **Bus Couplers for all usual bus systems**

The Beckhoff Bus Terminal system unites the advantages of a bus system with the possibilities of the compact series terminal. Bus Terminals can be driven within all the usual bus systems, thus reducing the controller parts count. The Bus Terminals then behave like conventional connections for that bus system. All the performance features of the particular bus system are supported.

### **Mounting on standardized mounting rails**

The installation is standardized thanks to the simple and space-saving mounting on a standardized mounting rail (EN 60715, 35 mm) and the direct wiring of actuators and sensors, without cross connections between the terminals. The consistent labelling scheme also contributes.

The small physical size and the great flexibility of the Bus Terminal system allow it to be used wherever a series terminal is also used. Every type of connection, such as analog, digital, serial or the direct connection of sensors can be implemented.

### **Modularity**

The modular assembly of the terminal strip with Bus Terminals of various functions limits the number of unused channels to a maximum of one per function. The presence of two channels in one terminal is the optimum compromise of unused channels and the cost of each channel. The possibility of electrical isolation through potential feed terminals also helps to keep the number of unused channels low.

### **Display of the channel state**

The integrated LEDs show the state of the channel at a location close to the sensors and actuators.

### **K-Bus**

The K-Bus is the data path within a terminal strip. The K-Bus is led through from the Bus Coupler through all the terminals via six contacts on the terminals' side walls. The end terminal terminates the K-Bus. The user does not have to learn anything about the function of the K-Bus or about the internal workings of the terminals and the Bus Coupler. Many software tools that can be supplied make project planning, configuration and operation easy.

### **Potential feed terminals for isolated groups**

The operating voltage is passed on to following terminals via three power contacts. You can divide the terminal strip into arbitrary isolated groups by means of potential feed terminals. The potential feed terminals play no part in the control of the terminals, and can be inserted at any locations within the terminal strip.

Up to 64 Bus Terminals can be used in a terminal block, with optional K-Bus extension for up to 256 Bus Terminals. This count does include potential feed terminals, but not the end terminal.

### **Bus Couplers for various fieldbus systems**

Various Bus Couplers can be used to couple the electronic terminal strip quickly and easily to different fieldbus systems. It is also possible to convert to another fieldbus system at a later time. The Bus Coupler performs all the monitoring and control tasks that are necessary for operation of the connected Bus Terminals. The operation and configuration of the Bus Terminals is carried out exclusively by the Bus Coupler. Nevertheless, the parameters that have been set are stored in each Bus Terminal, and are retained in the event of voltage drop-out. Fieldbus, K-Bus and I/O level are electrically isolated.

If the exchange of data over the fieldbus is prone to errors or fails for a period of time, register contents (such as counter states) are retained, digital outputs are cleared, and analog outputs take a value that can be configured for each output when commissioning. The default setting for analog outputs is 0 V or 0 mA. Digital

outputs return in the inactive state. The timeout periods for the Bus Couplers correspond to the usual settings for the fieldbus system. When converting to a different bus system it is necessary to bear in mind the need to change the timeout periods if the bus cycle time is longer.

**The interfaces**

A Bus Coupler has six different methods of connection. These interfaces are designed as plug connectors and as spring-loaded terminals.

## 3 Mounting and wiring

### 3.1 Instructions for ESD protection

#### NOTE

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

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

- Please ensure you are electrostatically discharged and avoid touching the contacts of the device directly.
- Avoid contact with highly insulating materials (synthetic fibers, plastic film etc.).
- Surroundings (working place, packaging and personnel) should be grounded probably, when handling with the devices.
- Each assembly must be terminated at the right hand end with a KL9010 bus end terminal, to ensure the protection class and ESD protection.

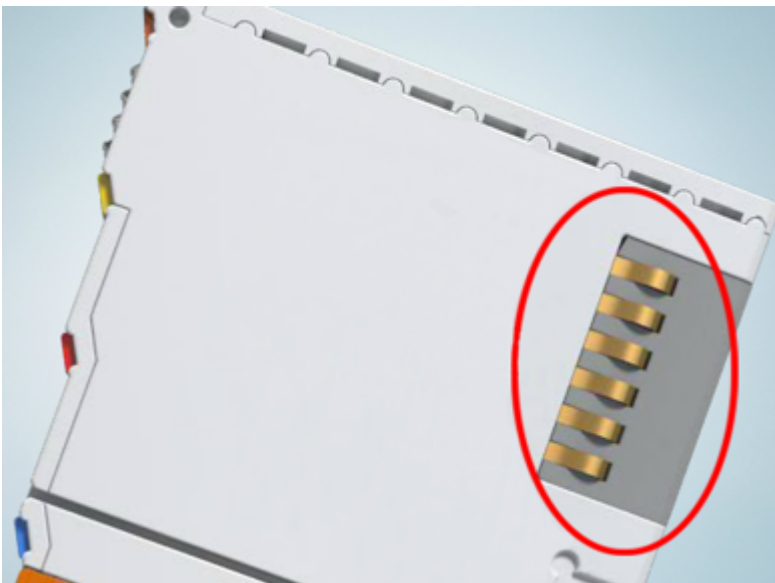


Fig. 3: Spring contacts of the Beckhoff I/O components

## 3.2 Mounting

### ⚠ WARNING

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

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

### 3.2.1 Dimensions

The Beckhoff Bus Terminal system is characterized by low physical volume and high modularity. When planning a project it must be assumed that at least one Bus Coupler and a number of Bus Terminals will be used. The mechanical dimensions of the Bus Couplers are independent of the fieldbus system.

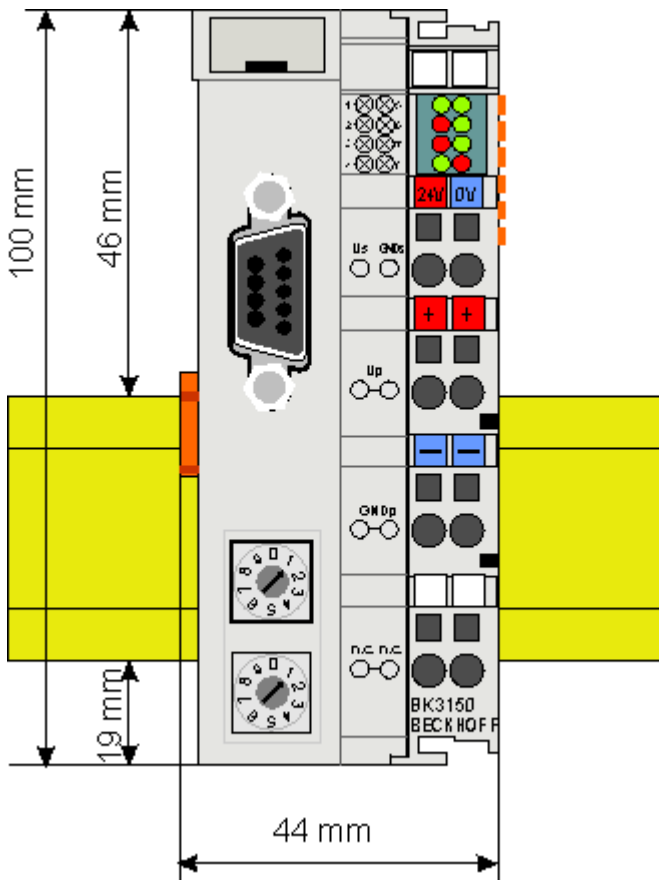


Fig. 4: BCxx50 - Dimensions

The total width of the fieldbus station is the width of the Bus Coupler/Bus Terminal Controller plus the width of the Bus Terminals being used (incl. KL9010 bus end terminal). Depending on design, the Bus Terminals are 12 mm or 24 mm wide. The height is 100 mm.

The BCxx50 series Bus Terminal Controllers are 68 mm deep.



### 3.2.2 Installation on mounting rails

The Bus Coupler and all the Bus Terminals can be clipped, with a light press, onto a 35 mm mounting rail. A locking mechanism prevents the individual housings from being pulled off again. For removal from the mounting rail the orange colored tension strap releases the latching mechanism, allowing the housing to be pulled off the rail without any force.

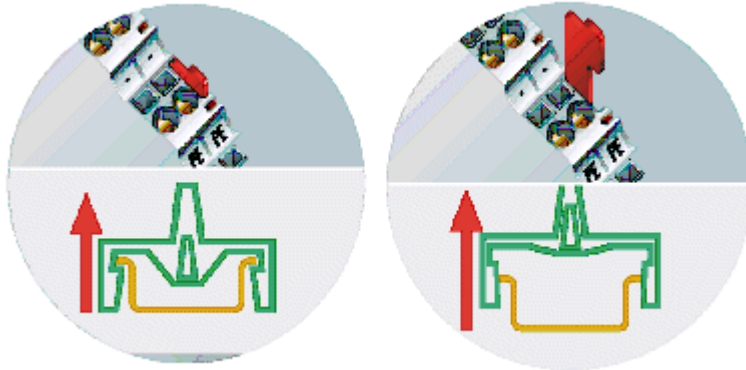


Fig. 5: Release the locking mechanism by pulling the orange tab

Up to 64 Bus Terminals can be attached to the Bus Coupler on the right hand side. When plugging the components together, be sure to assemble the housings with groove and tongue against each other. A properly working connection cannot be made by pushing the housings together on the mounting rail. When correctly assembled, no significant gap can be seen between the attached housings.

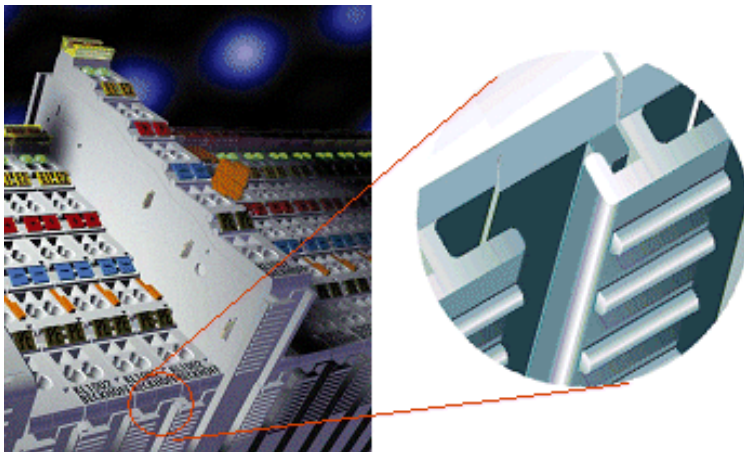


Fig. 6: Groove and tongue of the housings

**NOTE**

**Bus Terminals should only be pulled or plugged in switched-off state.**

Insertion and removal of Bus Terminals is only permitted when switched off. The electronics in the Bus Terminals and in the Bus Coupler are protected to a large measure against damage, but incorrect function and damage cannot be ruled out if they are plugged in under power.

### 3.2.3 Disposal



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

## 3.3 Wiring

### ⚠ WARNING

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

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

### 3.3.1 Potential groups, insulation testing and PE

#### Potential groups

A Beckhoff Bus Terminal block usually has three different potential groups:

- The fieldbus interface is electrically isolated (except for individual Low Cost couplers) and forms the first potential group.
- Bus Coupler / Bus Terminal Controller logic, K-bus and terminal logic form a second electrically isolated potential group.
- The inputs and outputs are supplied via the power contacts and form further potential groups.

Groups of I/O terminals can be consolidated to further potential groups via potential supply terminals or separation terminals.

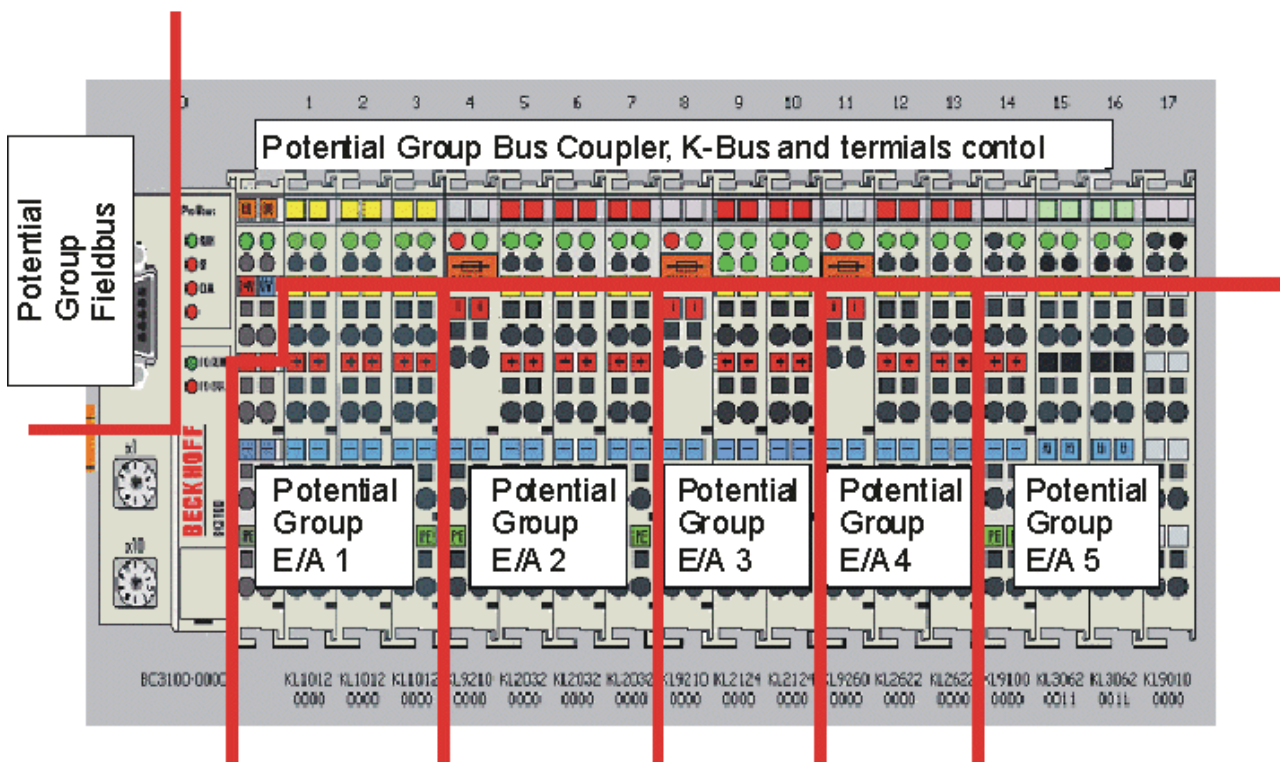


Fig. 7: Potential groups of a Bus Terminal block

#### Insulation testing

The connection between Bus Coupler / Bus Terminal Controller and Bus Terminals is realized automatically by latching the components. The transfer of the data and the supply voltage for the intelligent electronics in the Bus Terminals is performed by the K-bus. The supply of the field electronics is performed through the power contacts. Plugging together the power contacts creates a supply rail. Since some Bus Terminals (e.g. analog Bus Terminals or 4-channel digital Bus Terminals) are not looped through these power contacts or not completely the Bus Terminal contact assignments must be considered.

The potential feed terminals interrupt the power contacts, and represent the start of a new supply rail. The Bus Coupler / Bus Terminal Controller can also be used for supplying the power contacts.

### PE power contacts

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

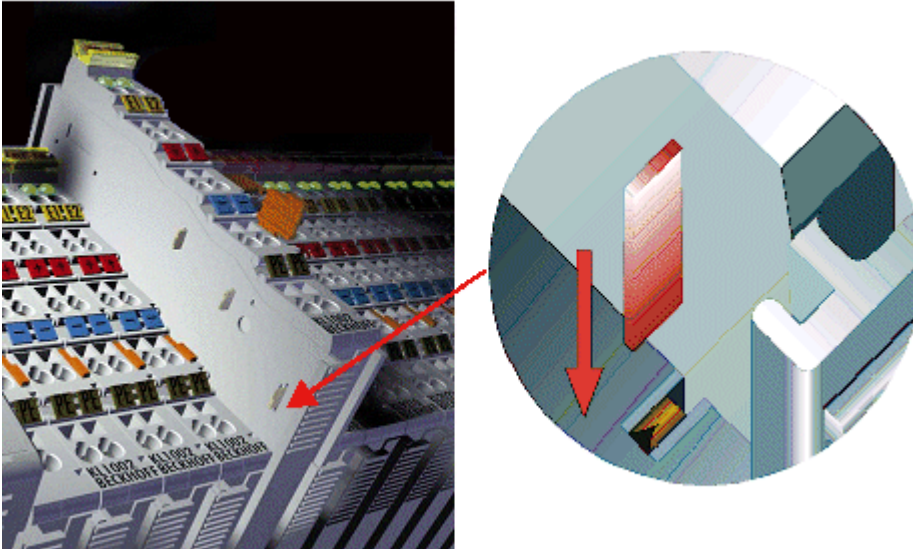


Fig. 8: Power contact on the left

It should be noted that, for reasons of electromagnetic compatibility, the PE contacts are capacitively coupled to the mounting rail. This can both lead to misleading results and to damaging the terminal during insulation testing (e.g. breakdown of the insulation from a 230 V power consuming device to the PE conductor). The PE supply line at the Bus Coupler / Bus Terminal Controller must be disconnected for an insulation test. In order to uncouple further feed locations for the purposes of testing, the feed terminals can be pulled at least 10 mm out from the connected group of other terminals. In that case, the PE conductors do not have to be disconnected.

The power contact with the label PE must not be used for other potentials.

### 3.3.2 Power supply

**⚠ WARNING**

**Risk of injury through electric shock and damage to the device!**  
 Bring the Bus Terminals system into a safe, de-energized state before starting mounting, disassembly or wiring of the components!

#### Supply of Bus Terminal Controller and Bus Terminals (Us)

The Bus Terminal Controller requires a supply voltage of 24 V<sub>DC</sub>.

The connection is made by means of the upper spring-loaded terminals labelled 24 V and 0 V. This supply voltage is used for the electronic components of the Bus Coupler / Bus Terminal Controllers and (via the K-bus) the electronic components of the Bus Terminals. It is galvanically separated from the field level voltage.

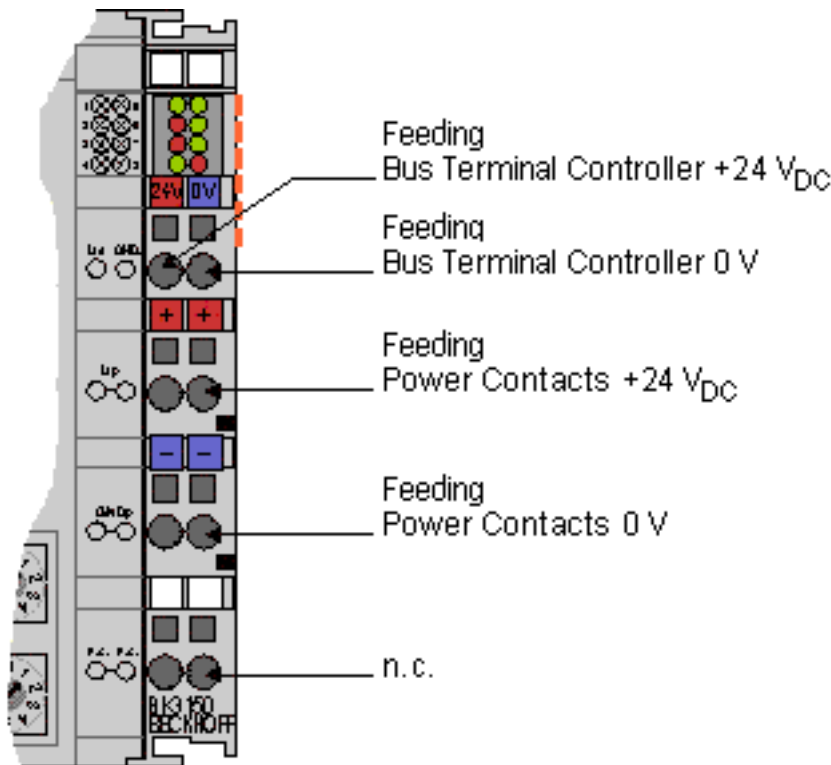


Fig. 9: Terminal points for the Bus Terminal Controller supply

**UL US LISTED**  
 Ind. Cont. Eq.: 24TB  
 For Us and GNDs:  
 Use 4 Amp. fuse or  
 Class 2 power supply.  
 See instructions.

Fig. 10: UL identification

**⚠ CAUTION**

**Note the UL requirements for the power supply.**

To comply with the UL requirements, the 24 V<sub>DC</sub> supply voltage for Us must originate

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

**⚠ CAUTION****No unlimited voltage sources!**

To comply with the UL requirements, Us must not be connected with unlimited voltage sources.

**Power contacts supply (Up)**

The bottom six connections with spring-loaded terminals can be used to feed the supply for the peripherals. The spring-loaded terminals are joined in pairs to a power contact. The feed for the power contacts has no connection to the voltage supply for the BC electronics.

The spring-loaded terminals are designed for wires with cross-sections between 0.08 mm<sup>2</sup> and 2,5 mm<sup>2</sup>.

The assignment in pairs and the electrical connection between feed terminal contacts allows the connection wires to be looped through to various terminal points. The current load from the power contact must not exceed 10 A for long periods. The current carrying capacity between two spring-loaded terminals is identical to that of the connecting wires.

**Power contacts**

On the right hand face of the Bus Terminal Controller there are three spring contacts for the power contact connections. The spring contacts are hidden in slots so that they cannot be accidentally touched. By attaching a Bus Terminal the blade contacts on the left hand side of the Bus Terminal are connected to the spring contacts. The tongue and groove guides on the top and bottom of the Bus Terminal Controllers and of the Bus Terminals guarantees that the power contacts mate securely.

### 3.3.3 PROFIBUS Connection

#### M12 round connector

The M12 socket is inverse-coded and has five pins. Pin 1 transfers 5 V<sub>DC</sub>, pin 3 transfers GND for the active termination resistor. These must never be misused for other functions, as this can lead to destruction of the device.

Pins 2 and 4 transfer the PROFIBUS signals. These must never be swapped over, as this will prevent communication. Pin 5 transfers the shield, which is capacitively connected to the base of the Fieldbus Box.

#### Pin assignment M12 socket (-B310)

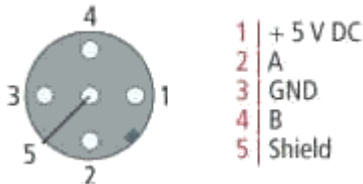


Fig. 11: Pin assignment M12 socket (-B310)

#### Pin assignment M12 socket/plug connector (-B318)

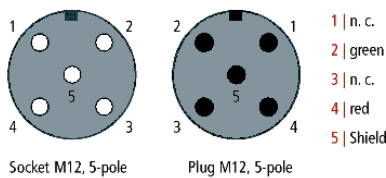


Fig. 12: Pin assignment M12 socket/plug connector (-B318)

#### Nine-pin D-Sub

Pin 6 transfers 5 V<sub>DC</sub>, pin 5 transfers GND for the active termination resistor. These must never be misused for other functions, as this can lead to destruction of the device.

Pins 3 and 8 transfer the PROFIBUS signals. These must never be swapped over, as this will prevent communication.

#### Pin assignment of the PROFIBUS D-sub socket

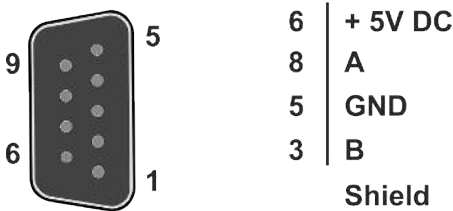


Fig. 13: Pin assignment of the PROFIBUS D-sub socket

#### PROFIBUS wire colors

PROFIBUS line	M12	D-Sub
B red	Pin 4	Pin 3
A green	Pin 2	Pin 8

#### Connection of the Fieldbus Box modules

The Fieldbus Box modules are connected either directly or via a T-piece (or Y-piece).

The B318 series features a socket and a plug connector, i.e. this is where the PROFIBUS is routed in the module. The supply voltage (+5 V<sub>DC</sub>) for the termination resistor is only present at the socket. The termination resistor ZS1000-1610 is only available as a plug connector.

The incoming PROFIBUS line should always end with a socket.

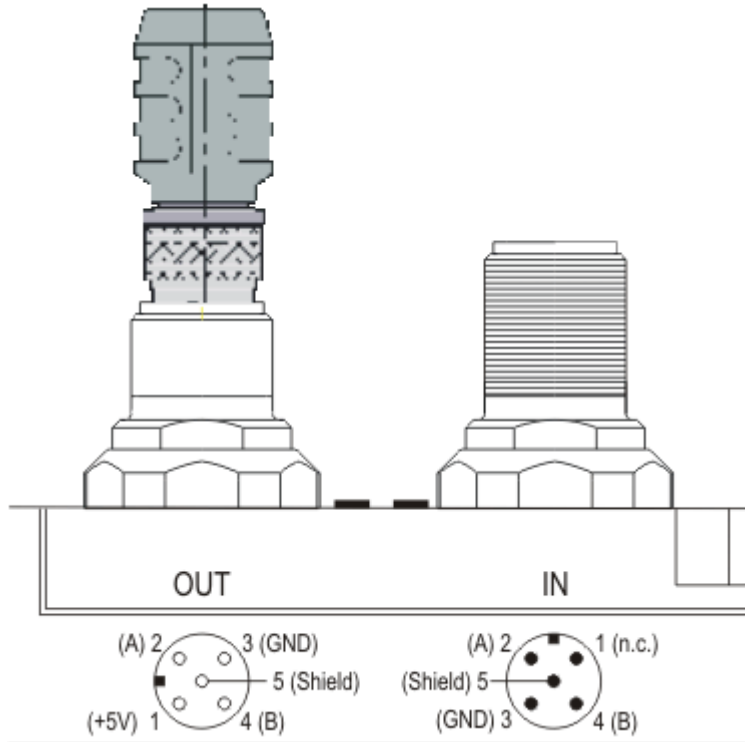


Fig. 14: Pin assignment socket/plug connector Fieldbus Box modules

Two T-pieces are available:

- ZS1031-2600 with +5 V<sub>DC</sub> transfer for supplying the termination resistor
- ZS1031-2610 without +5 V<sub>DC</sub> transfer

### 3.3.4 PROFIBUS cabling

Physical aspects of the data transmission are defined in the PROFIBUS standard (see PROFIBUS layer 1: Physical Layer).

The types of area where a fieldbus system can be used is largely determined by the choice of the transmission medium and the physical bus interface. In addition to the requirements for transmission security, the expense and work involved in acquiring and installing the bus cable is of crucial significance. The PROFIBUS standard therefore allows for a variety of implementations of the transmission technology while retaining a uniform bus protocol.

#### Cable-based transmission

This version, which accords with the American EIA RS-485 standard, was specified as a basic version for applications in production engineering, building management and drive technology. A twisted copper cable with one pair of conductors is used. Depending on the intended application area (EMC aspects should be considered) the screening may be omitted.

Two types of conductor are available, with differing maximum conductor lengths (see the RS-485 table).

## RS485 - Fundamental properties

RS-485 transmission according to the PROFIBUS standard	
Network topology	Linear bus, active bus terminator at both ends, stubs are possible.
Medium	Screened twisted cable, screening may be omitted, depending upon the environmental conditions (EMC).
Number of stations	32 stations in each segment with no repeater. Can be extended to 127 stations with repeater
Max. bus length without repeater	100 m at 12 Mbit/s 200 m at 1500 Kbit/s, up to 1.2 km at 93.75 kbit/s
Max. bus length with repeater	Line amplifiers, or repeaters, can increase the bus length up to 10 km. The number of repeaters possible is at least 3, and, depending on the manufacturer, may be up to 10.
Transmission speed (adjustable in steps)	9.6 kbit/s; 19.2 kbit/s; 93.75 kbit/s; 187.5 kbit/s; 500 kbit/s; 1500 kbit/s; 12 Mbit/s
Connector	9-pin D-Sub connector for IP20 M12 round connector for IP65/67

### Cabling for PROFIBUS DP and PROFIBUS FMS

Note the special requirements on the data cable for baud rates greater than 1.5 Mbaud. The correct cable is a basic requirement for correct operation of the bus system. If a simple 1.5 Mbaud cable is used, reflections and excessive attenuation can lead to some surprising phenomena. It is possible, for instance, for a connected PROFIBUS station not to achieve a connection, but for it to be included again when the neighboring station is disconnected. Or there may be transmission errors when a specific bit pattern is transmitted. The result of this can be that when the equipment is not operating, PROFIBUS works without faults, but that there are apparently random bus errors after start-up. Reducing the baud rate (< 93.75 kbaud) corrects this faulty behavior.

If reducing the baud rate does not correct the error, then in many cases this can indicate a wiring fault. The two data lines may be crossed over at one or more connectors, or the termination resistors may not be active, or they may be active at the wrong locations.

#### ● Recommended cables

**i** Installation is made a great deal more straightforward if preassembled cables from Beckhoff are used! Wiring errors are avoided, and commissioning is more rapidly completed. The Beckhoff range includes fieldbus cables, power supply cables, sensor cables and accessories such as termination resistors and T-pieces. Connectors and cables for field assembly are nevertheless also available.

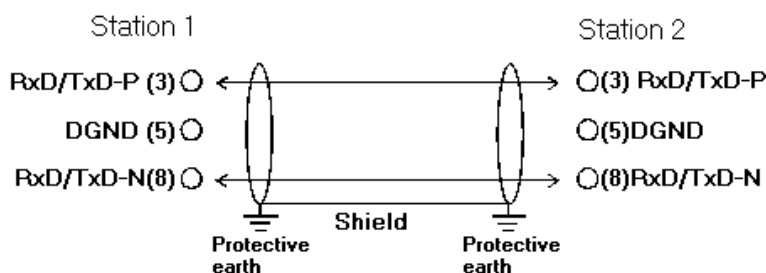


Fig. 15: PROFIBUS cable assignment

#### ● Termination resistors

**i** In systems with more than two stations all devices are wired in parallel. The PROFIBUS cable must be terminated with resistances at both ends, in order to avoid reflections and associated transfer problems.

### Distances

The bus cable is specified in EN 50170. This yields the following lengths for a bus segment.



Baud rate in kbits/sec	9.6	19.2	93.75	187.5	500	1500	12000
Cable length in m	1200	1200	1200	1000	400	200	100

Stubs up to 1500 kbaud <6.6 m; at 12 Mbaud stub segments should not be used.

**Bus segments**

A bus segment consists of at most 32 devices. 126 devices are permitted in a PROFIBUS network. Repeaters are required to refresh the signal in order to achieve this number. Each repeater is counted as one device.

IP-Link is the subsidiary bus system for Fieldbus Boxes, whose topology is a ring structure. There is an IP master in the coupler modules (IP230x-Bxxx or IP230x-Cxxx) to which up to 120 extension modules (IExxxx) may be connected. The distance between two modules may not exceed 5 m. When planning and installing the modules, remember that because of the ring structure the IP-Link master must be connected again to the last module.

**Installation guidelines**

When assembling the modules and laying the cables, observe the technical guidelines provided by the PROFIBUS User Organization (PROFIBUS Nutzerorganisation e.V.) for PROFIBUS DP/FMS (see <https://www.profibus.com>).

**Checking the PROFIBUS wiring**

A PROFIBUS cable (or a cable segment when using repeaters) can be checked with a few simple resistance measurements. The cable should meanwhile be removed from all stations:

1. Resistance between A and B at the start of the lead: approx. 110 Ohm
2. Resistance between A and B at the end of the lead: approx. 110 Ohm
3. Resistance between A at the start and A at the end of the lead: approx. 0 Ohm
4. Resistance between B at the start and B at the end of the lead: approx. 0 Ohm
5. Resistance between screen at the start and screen at the end of the lead: approx. 0 Ohm

If these measurements are successful, the cable is okay. If, in spite of this, bus malfunctions still occur, this is usually a result of EMC interference. Observe the installation notes from the PROFIBUS User Organization (<https://www.profibus.com>).

### 3.4 ATEX - Special conditions (extended temperature range)

#### ⚠ WARNING

**Observe the special conditions for the intended use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas (directive 2014/34/EU)!**

- The certified components are to be installed in a suitable housing that guarantees a protection class of at least IP54 in accordance with EN 60079-15! The environmental conditions during use are thereby to be taken into account!
- For dust (only the fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9): The equipment shall be installed in a suitable enclosure providing a degree of protection of IP54 according to EN 60079-31 for group IIIA or IIIB and IP6X for group IIIC, taking into account the environmental conditions under which the equipment is used!
- If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values!
- Observe the permissible ambient temperature range of -25 to 60°C for the use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas!
- Measures must be taken to protect against the rated operating voltage being exceeded by more than 40% due to short-term interference voltages!
- The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- The fuses of the KL92xx/EL92xx power feed terminals may only be exchanged if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

#### Standards

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

- EN 60079-0:2012+A11:2013
- EN 60079-15:2010
- EN 60079-31:2013 (only for certificate no. KEMA 10ATEX0075 X Issue 9)

#### Marking

The Beckhoff fieldbus components with extended temperature range (ET) certified according to the ATEX directive for potentially explosive areas bear the following marking:



**II 3G KEMA 10ATEX0075 X Ex nA IIC T4 Gc Ta: -25 ... +60°C**

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: -25 ... +60°C  
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

or



**II 3G KEMA 10ATEX0075 X Ex nA nC IIC T4 Gc Ta: -25 ... +60°C**

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: -25 ... +60°C  
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

### 3.5 Continulative documentation for ATEX and IECEx

#### NOTE



#### **Continulative documentation about explosion protection according to ATEX and IECEx**

Pay also attention to the continuative documentation

#### **Ex. Protection for Terminal Systems**

Notes on the use of the Beckhoff terminal systems in hazardous areas according to ATEX and IECEx,

that is available for [download](#) within the download area of your product on the Beckhoff homepage [www.beckhoff.com](http://www.beckhoff.com)!

## 4 Parameterization and Commissioning

### 4.1 Start-up behavior of the Bus Terminal Controller

When the Bus Terminal Controller is switched on it checks its state, configures the K-bus, creates a configuration list based on the connected Bus Terminals and starts its local PLC.

The I/O LEDs flash when the Bus Terminal Controller starts up. If the system is in an error-free state, the I/O LEDs should stop flashing after approx. 2-3 seconds. In the event of a fault the error type determines which LED flashes (see chapter *Diagnostic LEDs*).

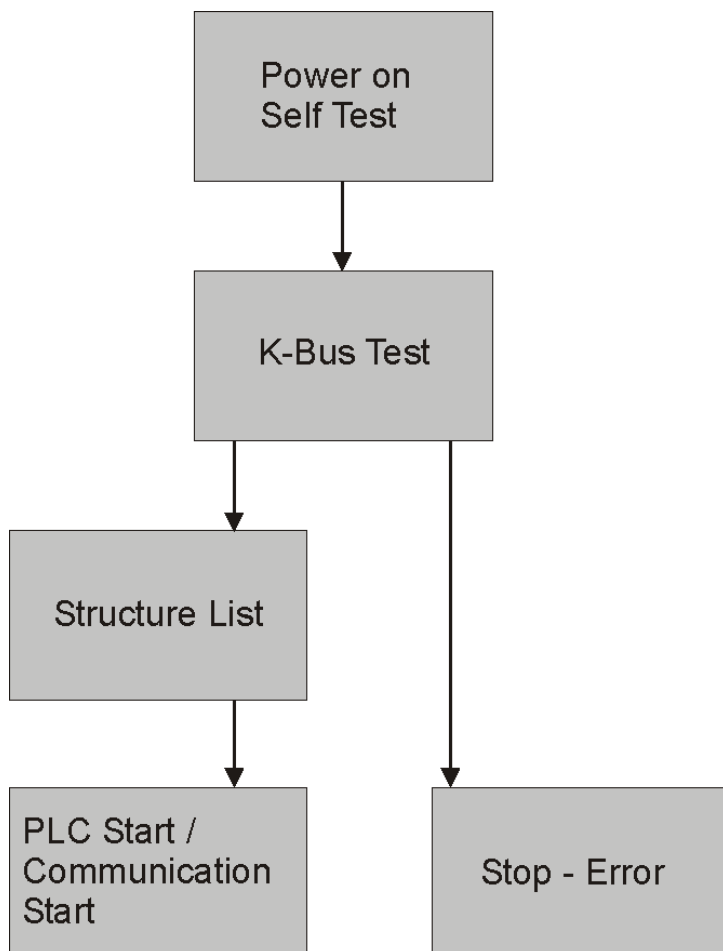


Fig. 16: Start-up behavior of the Bus Terminal Controller

## 4.2 Configuration

### 4.2.1 Overview

#### Configuration types

The Bus Terminal controllers of the BCxx50, BCxx20 and BXxx00 series can be configured in two different ways: DEFAULT CONFIG or TwinCAT CONFIG.

#### DEFAULT-CONFIG

Bus Terminals are mapped in the order they are inserted, i.e. first the complex Bus Terminals followed by the digital Bus Terminals.

The complex Bus Terminals are mapped as follows:

- Word Alignment
- complex representation

#### ⚠ CAUTION

#### The process image depends on the connected terminals!

The process image changes when a terminal is added or removed!

The data of the fieldbus slaves interface are referred to as PLC variables. The PLC variables have addresses from %QB1000 and %IB1000

The DEFAULT CONFIG (without PLC program) can also be used for writing and testing of the Connected Bus Terminals. To this end, the Bus Terminal Controller must be scanned in the System Manager, and FreeRun mode must be enabled (to use this function, no PLC program may be active on the Bus Terminal Controller).

#### TWINCAT-CONFIG

In the TwinCAT CONFIG the Bus Terminals and PLC variables can be freely linked as required (TwinCAT System Manager file required). The configuration is transferred to the coupler via the System Manager and ADS.

The following is required for the TwinCAT configuration (TC file):

- Via the fieldbus (PROFIBUS, CANopen, Ethernet)
  - PROFIBUS: (BC3150, BX3100)
    - PC with FC310x from version 2.0 and TwinCAT 2.9 build 1000
    - BX3100 with CIF60 or CP5412
    - TwinCAT 2.9 build 946  
(NOTE: with PROFIBUS cards from Hilscher only one ADS communication is permitted, i.e. either System Manager or PLC Control)
    - CANopen: (BC5150, BX5100)
      - PC with FC510x from version 1.76 TwinCAT build 1030
      - DeviceNet: (BC5250, BX5200)
    - on request
    - Ethernet: (BC9050, BC9020, BC9120, BX9000)
    - PC with TwinCAT 2.10 build 1322
  - Via the serial ADS TwinCAT 2.9 build 1010
    - BX3100 version 1.00
    - BX5100 version 1.00
    - BX5200 version 1.10
    - BX8000 version 1.00

- BC3150, BC5150, BC5250, BC9050, BC9020, BC9120 from firmware B0
- For BC8150 from TwinCAT 2.10 build 1243

BCxx50 and BXxx00 can be parameterized via the System Manager of the TwinCAT program.

- Variable I/O mapping
- Type-specific PROFIBUS data (BC3150 and BX3100 only)
- RTC (real-time clock) (BX series only)
- SSB (Smart System Bus) (BX series only)
- PLC settings
- K-Bus settings

The configuration can be transferred to the BCxx50 or BXxx00 via fieldbus ADS protocol or serial ADS protocol.

The TwinCAT configuration can be used to link variables, I/Os and data. The following is possible:

- PLC - K-BUS
- PLC fieldbus (e.g. PROFIBUS slave interface to PLC)
- K-bus fieldbus (only for BX controllers)
- Support for TwinSAFE terminals (only BX controllers from firmware 1.17)

In addition, the TwinCAT configuration can be used to parameterize special behavior, for example whether data are preserved or set to "0" in the event of a fieldbus error.

The real-time clock can be set via a tab in the system manager.

### **Work steps**

1. Setting the fieldbus address
2. Open the System Manager and create a TC file
3. Configure fieldbus data in the TC file
4. Save the TC file
5. Opening a new system manager, creating a PC file and reading in saved TX file
6. Creating a link to a PLC task
7. Saving the configuration
8. Starting the TwinCAT system
9. Open the TC file in the System Manager, complete the configuration and transfer it to the BCxx50, BCxx20 or BXxx00
10. Transfer the program to BCxx50, BCxx20 or BXxx00
11. Creating a boot project

## 4.2.2 Creating a TwinCAT configuration

In order to configure a Bus Terminal Controller of the BCxx50, BCxx20, BXxx00 or BC9191 series, create a BX file in the System Manager. To simplify matters, files for the basic units have already been prepared. Open the corresponding Bus Terminal Controller with *New from Template*.

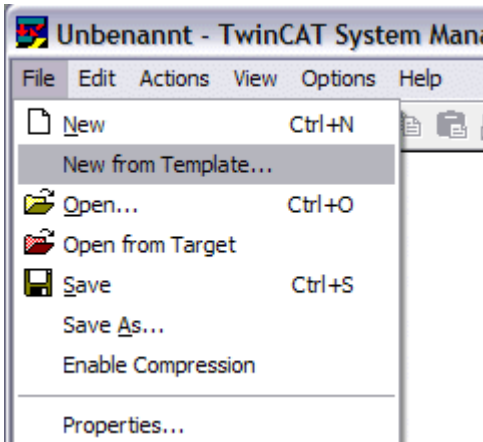


Fig. 17: Creating a TwinCAT configuration

Select the corresponding Bus Terminal Controller.

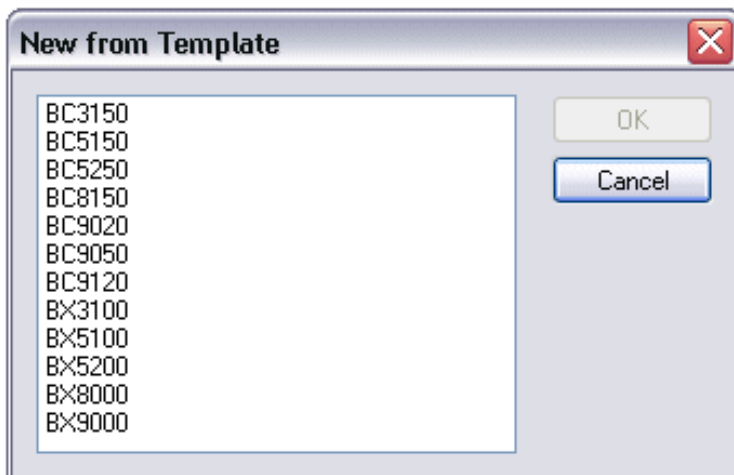


Fig. 18: Selecting the Bus Terminal Controller

All Bus Terminal Controller components are now available:

- Fieldbus interface
- [K-bus interface \[► 48\]](#)
- [PLC program \[► 50\]](#)
- SSB (only Bus Terminal Controllers of the BX series)

Please refer to the relevant chapter for device configuration.

### 4.2.3 Downloading a TwinCAT configuration

The TwinCAT configuration is loaded into the Bus Terminal Controller via ADS protocol.

#### Serial ADS protocol

(all Bus Terminal Controllers of the BXxx00 and BCxx50 series)

Enter the serial ADS connection, as described in the chapter [Serial ADS \[▶ 37\]](#).

#### ADS protocol via the fieldbus

(BC3150, BC5150, BC9x20, BC9050, BX3100, BX5100, BX9000, BC9191 only)

A prerequisite is that TwinCAT operates as master and is engaged in data exchange, i.e. the physical and fieldbus configuration must be complete, and data exchange must take place between the master (e.g. fieldbus master card) and the Bus Terminal Controller.

#### Choose Target System

Select the Bus Terminal Controller onto which the configuration is to be loaded. Use the function key F8 to open the dialog for downloading your file to the corresponding device.

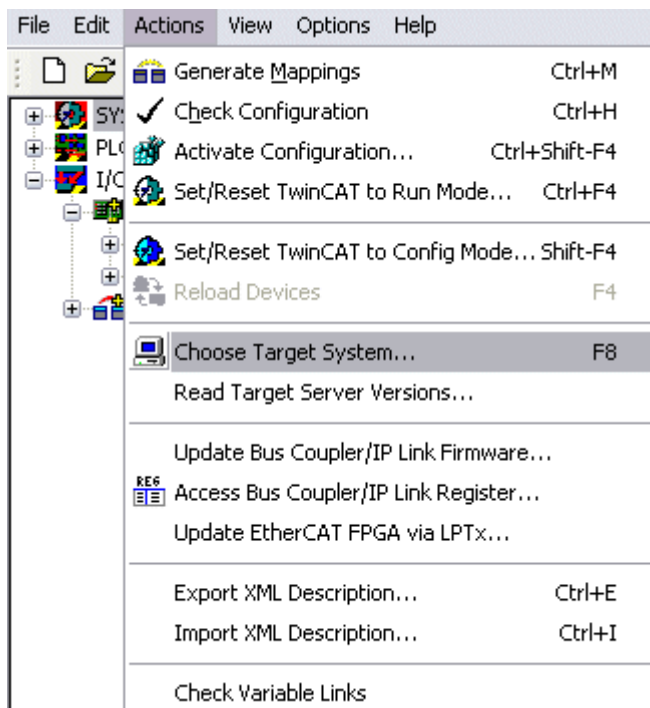


Fig. 19: Downloading a TwinCAT configuration

Select the corresponding Bus Terminal Controller.



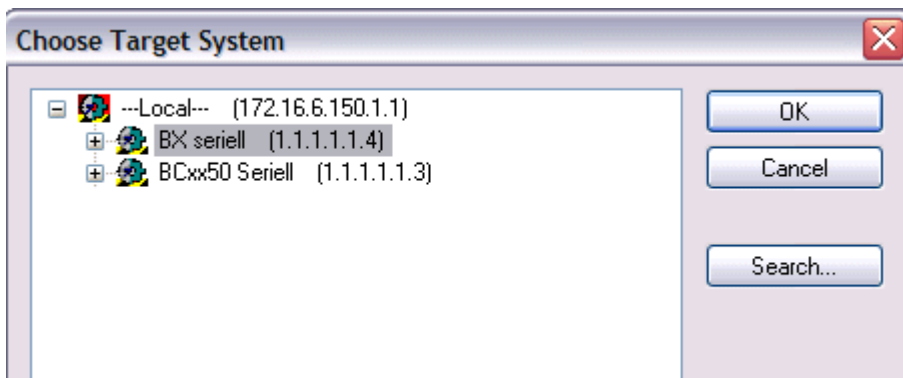


Fig. 20: Selecting the Bus Terminal Controller

The state of the Bus Terminal Controller is shown at the bottom right of the System Manager.

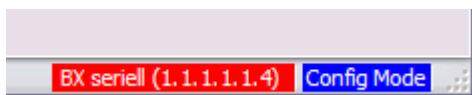


Fig. 21: State of the Bus Terminal Controller

In *Config mode / FreeRun* the configuration can now be downloaded to Bus Terminal Controller. If the Bus Terminal Controller is in *Stop mode*, ADS communication is not yet activated. In this case, it is not possible to download the configuration.

To activate the TwinCAT configuration select **Ctrl+Shift+F4** or *Activate Configuration*.

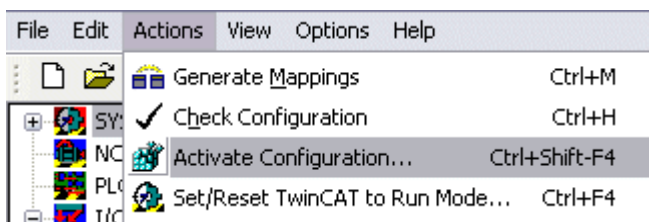


Fig. 22: Activating the TwinCAT configuration

The current configuration is loaded onto the Bus Terminal Controller. The display will show *Store Config*, and the BUS and I/O LED will flash. Once the configuration is successfully loaded onto Bus Terminal Controller, *TwinCAT Config* should appear in the display of a BXxx00. The corresponding program can now be transferred to the Bus Terminal Controller (program-download via the fieldbus).

## 4.2.4 Uploading a TwinCAT configuration

The TwinCAT configuration is loaded into the Bus Terminal Controller via ADS protocol.

### Serial ADS protocol

(all Bus Terminal Controllers of the BCxx50, BCxx20 and BXxx00 series)

Enter the serial ADS connection, as described in the chapter [Serial ADS \[▶ 37\]](#).

### ADS protocol via the fieldbus

(BC3150, BC5150, BC9x20, BC9050, BX3100, BX5100, BX9000, BC9191 only)

A prerequisite is that TwinCAT operates as master and is engaged in data exchange, i.e. the physical and fieldbus configuration must be complete, and data exchange must take place between the master (e.g. fieldbus card) and the Bus Terminal Controller.

## Choose Target System

Select the Bus Terminal Controller onto which the configuration is to be loaded. Use the function key [F8] to open the dialog for downloading your file to the corresponding device.

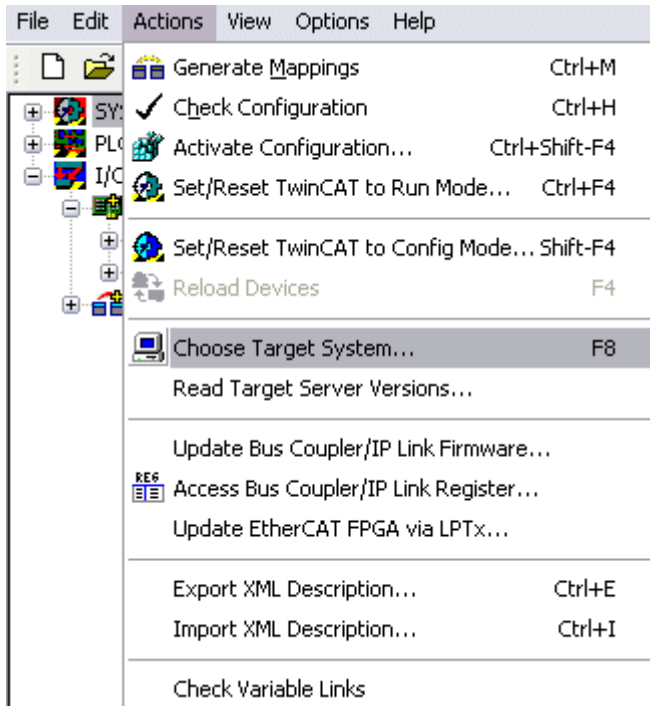


Fig. 23: Choose Target System

Select the corresponding Bus Terminal Controller.

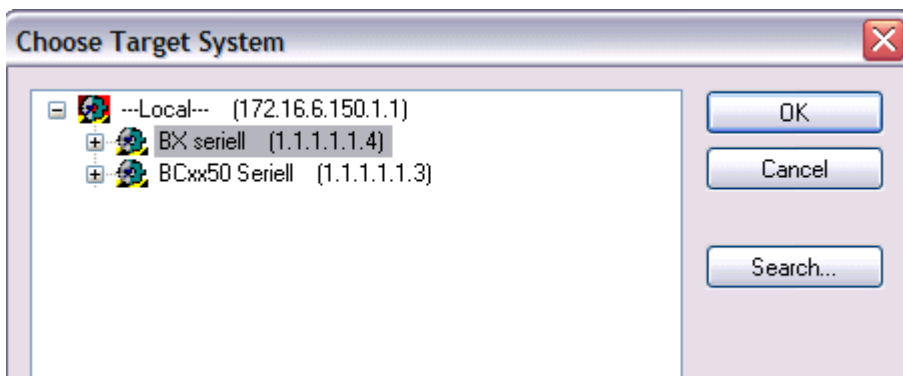


Fig. 24: Selecting the Bus Terminal Controller

The state of the Bus Terminal Controller is shown at the bottom right of the System Manager.

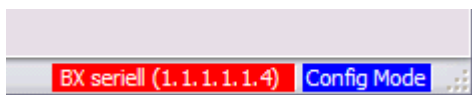


Fig. 25: State of the Bus Terminal Controller

Click on the red folder. The TwinCAT configuration will now be uploaded.

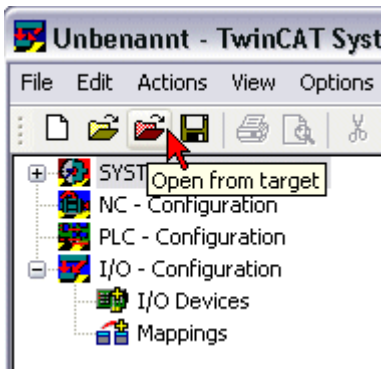


Fig. 26: Uploading the TwinCAT configuration

### 4.2.5 Resources in the Bus Terminal Controller

The memory resources assigned in the Bus Terminal Controller are shown in the System Manager in the *Resources* tab of the Bus Terminal Controller.

#### Mapping code

The mapping code is required for calculating the TwinCAT configuration (see Figure *Memory for the code mapping*). The percentages are added here. In the example from Fig. *Memory for code mapping*, 8% of the memory is allocated to the mapping calculation.

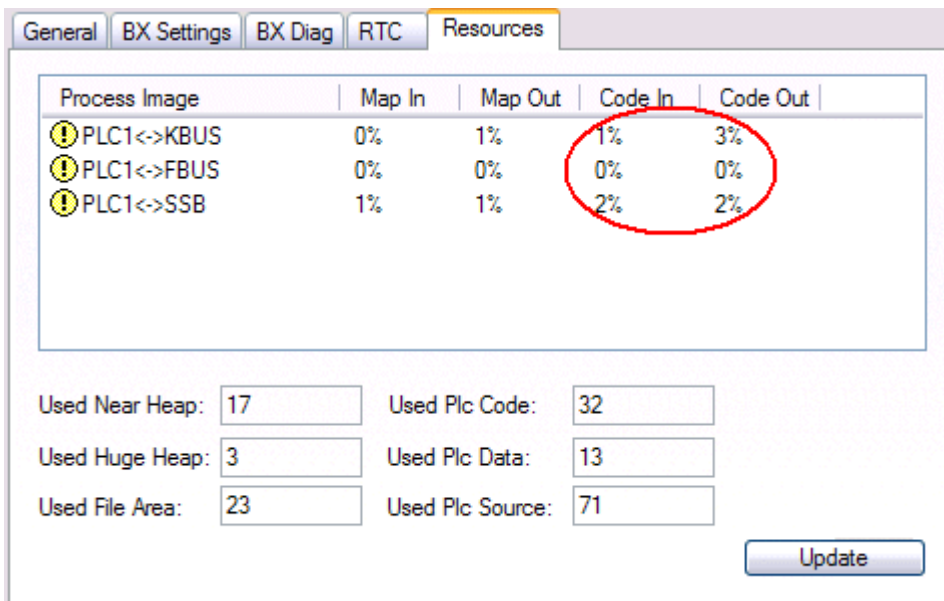


Fig. 27: Memory for code mapping

#### Data memory mapping

Data memory for mapping. The values are to be considered individually, i.e. each value can be up to 100%.

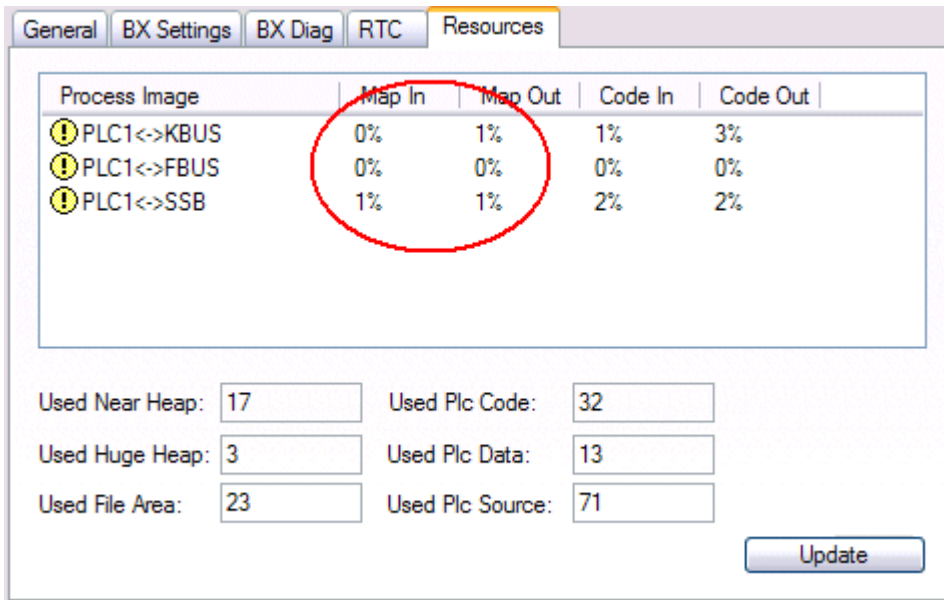


Fig. 28: Data memory mapping

**Used code and data memory**

Fig. Code and data memory (1) "Used PLC code" in %.

Fig. Code and data memory (2) "Used PLC data" in %.

Fig. Code and data memory (3) "Used PLC source" in %.

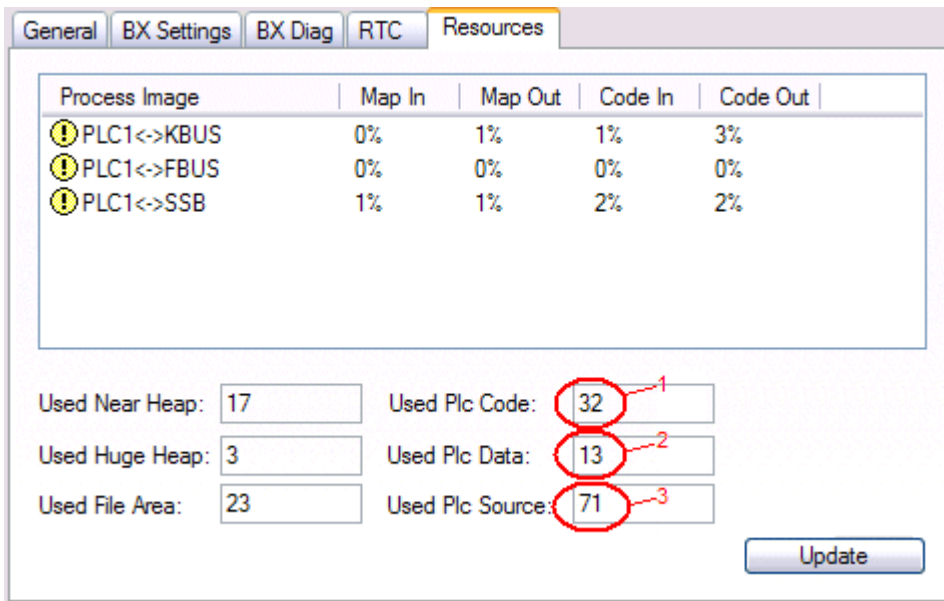


Fig. 29: Code and data memory

**Other memory**

Fig. Other Memory (1) "Used Near Heap" is required for the COM interface and SSB. % values.

Fig. Other Memory (2) "Used Huge Heap" is required for the ADS communication. % values. This value should be less than 30 %.

Fig. Other Memory (3) "Used File Area" is required for the TwinCAT configuration, the TSM file and the 16 kbyte flash access. % values.

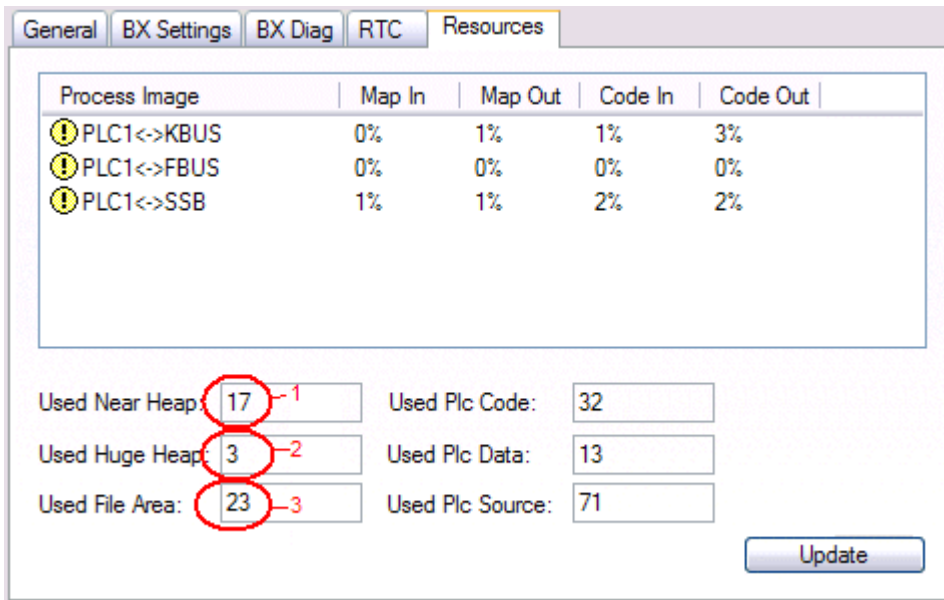


Fig. 30: Other memory

## 4.2.6 ADS connection via serial interface

(from firmware version 1.xx or 0.99x, Bus Terminal Controllers of the BX series and for all BCxx50)

From TwinCAT 2.9 build 1020 (TwinCAT level PLC, NC or NCI)

### ● Use only a serial connection

**i** To ensure trouble-free operation of the ADS link via the serial interface, only a serial connection to the BX controller is allowed. After successful configuration via the System Manager, close the System Manager before starting programming.

### ● AMS Net ID in delivery state (default)

#### **i** For BX9000

The default AMS Net ID is 172.16.21.20.1.1. If the IP address of the BX9000 is changed, the AMS Net ID of the BX9000 also changes. There is a menu option for displaying the current AMS Net ID. Example: If you change the IP address to 10.2.3.7, the AMS Net ID changes to 10.2.3.7.1.1.

#### **For BC9050, BC9020, BC9120**

The default AMS Net ID is 172.16.xxx.[DIP switch].1.1. If the IP address of the BX9xxx is changed, the AMS Net ID of the BX9xxx also changes.

Example: If you change the IP address to 10.2.3.7, the AMS Net ID changes to 10.2.3.7.1.1.

BC9050: DEFAULT 172.16.21.[DIP-Switch].1.1

BC9020: DEFAULT 172.16.22.[DIP-Switch].1.1

BC9120: DEFAULT 172.16.23.[DIP-Switch].1.1

### Initializing the ADS connection

Enter the Bus Terminal Controller in the remote connection under TwinCAT. Click on the TwinCAT icon and open the features menu. The following settings can be made under the >AMS Remote< tab.

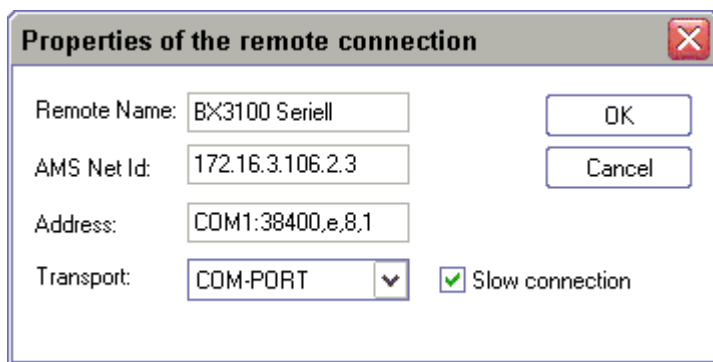


Fig. 31: Properties of the remote connection

Remote Name: Any  
 AMS-Net-ID: 1.1.1.1.1.1 (Default)  
 Address: COM Port: Baud rate, parity, data bits, stop bits  
 Transport: Select "COM port"

When the Bus Terminal Controller is switched on, the default AMS Net ID is always "1.1.1.1.1" (except all Ethernet Controllers).

The AMS Net ID can be changed as required. Please note that the new AMS Net ID cannot be changed again in this way.

If you need to change the new AMS Net ID again, you have to restart the Bus Terminal Controller, so that the AMS Net ID is reset to the default AMS Net ID, "1.1.1.1.1".  
 You can now change the AMS Net ID again.

### ● Strings can only be entered at the second call

**i** No strings can be entered under address when the dialog is first called (see above). Enter the name, AMS Net ID and transport type and close the dialog. With the second call you can enter your COM port.

The communication starts when TwinCAT is in Config mode (TwinCAT icon is blue) or RUN mode (TwinCAT icon is green). The COM interface remains open until a TwinCAT stop occurs (TwinCAT icon is red). It is then available again for other programs. No error message is issued if the COM interface is used by another program during a TwinCAT restart (e.g. by the KS2000 configuration software).

### ● AMS Net ID after ADS connection via the fieldbus

**i** If you have addressed the Bus Terminal Controller with an ADS connection via the fieldbus before the serial ADS was used, the AMS Net ID was automatically changed by the System Manager. In this case a new serial ADS connection is only possible, if the AMS Net ID is adjusted.

## **BX series: reading the AMS Net ID**

The current AMS Net ID can be read from the menu via the display of BX series Bus Terminal Controller.

AMS	AMS Net ID
1.1.1.1.1.1	

## 4.2.7 PROFIBUS

### 4.2.7.1 PROFIBUS settings

The PROFIBUS-relevant settings are made in the DP slaves tab.

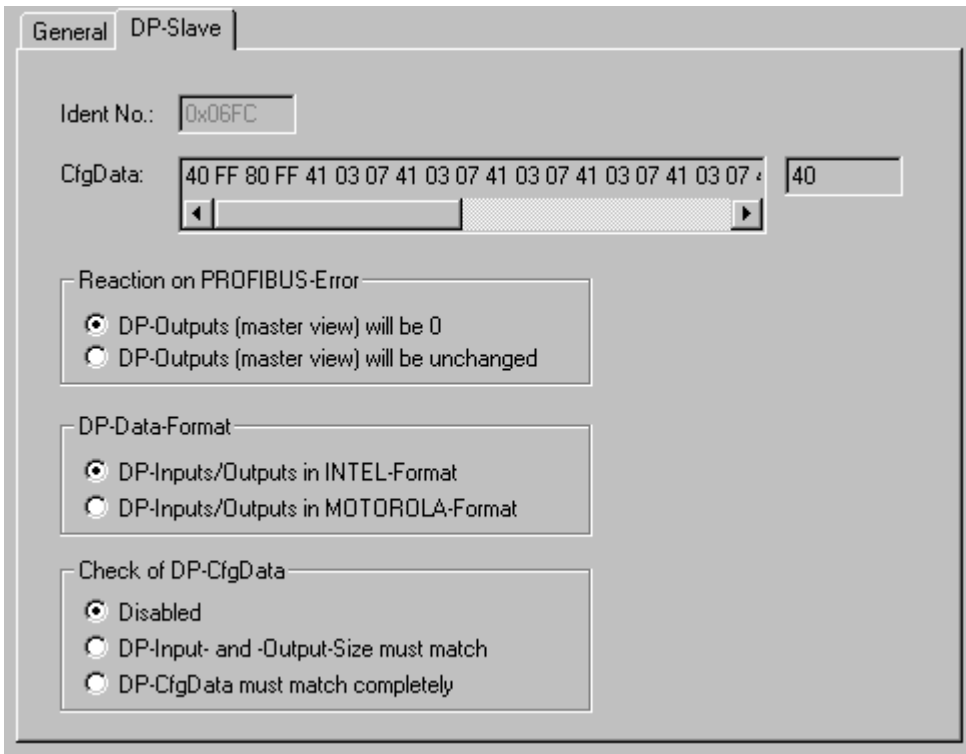


Fig. 32: DP slaves tab

#### Reaction to PROFIBUS Error

In the event of a PROFIBUS error (e.g. PROFIBUS cable is pulled), the output data of the PROFIBUS master are retained unchanged or set to 0 (depending on the parameterization). From the perspective of the Bus Terminal controller, these are input data which the PROFIBUS master transfers to the Bus Terminal Controller.

#### DP data format

The PROFIBUS data can be transferred in Intel or Motorola format.

#### Checking the DP configuration data


The configuration data can


- not be checked at all;
- checked for correct length; or
- checked for length and content

## 4.2.7.2 Master configuration

### 4.2.7.2.1 Basic Device File (GSD)

All field devices with PROFIBUS slave interface are described via the GSD file. The BX3100 is a PROFIBUS DP slave. For data exchange, the GSD file has to be inserted in the master configuration tool. The maximum length of input and output data 244 bytes in each direction.

 Download GSD file (German) (<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3238674315.zip>)

 Download GSD file (English) (<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3238672139.zip>)

These files are also available from the download area of the BX9000 at the Beckhoff website:  
[www.beckhoff.com/BX3100](http://www.beckhoff.com/BX3100)

The GSD file can describe the following data types:

Variable	Length
Integer 8	1 bytes
UnInteger 8	1 bytes
Integer 16	2 bytes
UnInteger 16	2 bytes
Integer 32	4 bytes
UnInteger 32	4 bytes
Float 32	4 bytes

Data types that are not listed, e.g. BOOL variables, can be described via the length. For this purpose, the GSD file includes configuration data starting with a byte and ending with 64 words.

### 4.2.7.2.2 Creating a TwinCAT PC file

#### DEFAULT CONFIG

DEFAULT-CONFIG contains the PROFIBUS data from the address %IB1000 input and %QB1000. The data length depends on the number of configured PROFIBUS data. A maximum of 244 bytes of input data and 244 bytes of output data can be parameterized. No further settings are required for this configuration type.



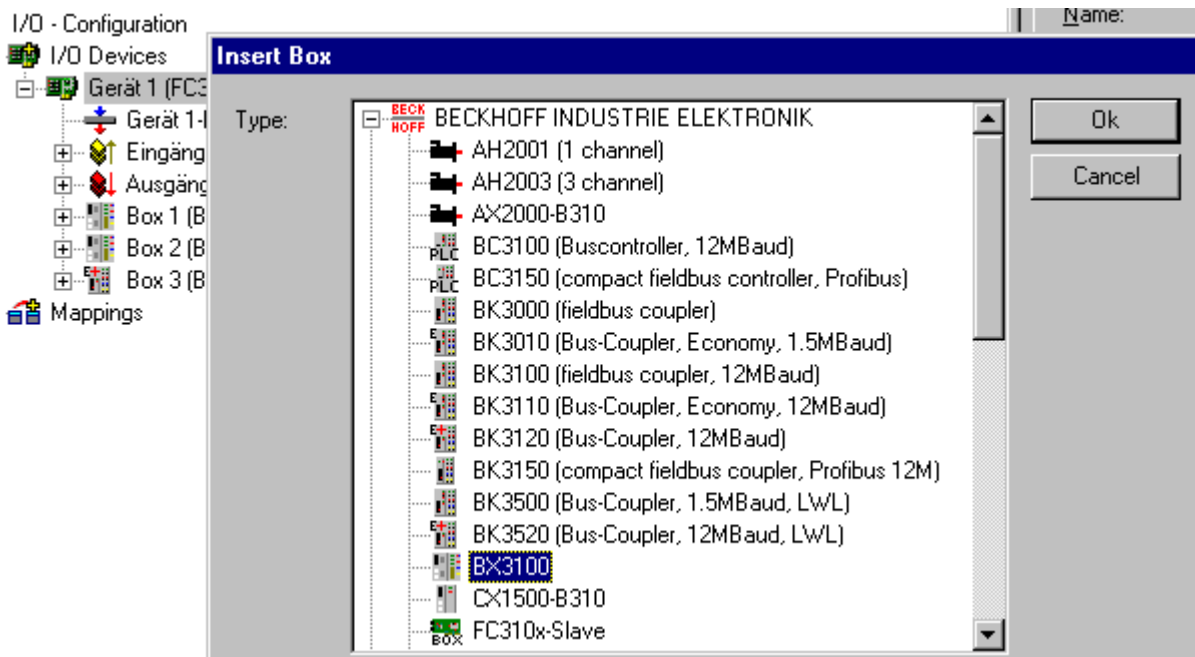


Fig. 33: Inserting BX3100

**Inserting the BX3100 in a PROFIBUS master via the System Manager**

Select your PROFIBUS master PC card in the System Manager and right-click on your PROFIBUS master card. The PROFIBUS devices will then be offered for selection, among them a BX3100. Select this and confirm with OK.

**Opening an existing BX file**

If you wish to work with the DEFAULT-CONFIG, interrupt the process at this point (see Creating PROFIBUS data in DEFAULT-CONFIG). Otherwise select your configured BX file. If no such file exists, you have to create one (Creating a BX file |> 31). Once you have selected your BX file, all required PROFIBUS data are copied automatically into your project from this file. You now have to create a link to your task and set the PROFIBUS address. The configuration can then be saved and the TwinCAT system started.

**Creating PROFIBUS data in DEFAULT-CONFIG**

- The PROFIBUS data can now be created. Various variables are available:
  - Integer 8 bit
  - Integer 16 bit
  - Integer 32 bit
  - Unsigned 8 bit
  - Unsigned 16 bit
  - Unsigned 32 bit
  - FLOAT 32 bit

They have to match the variable types currently projected in the BX. Data sizes from 1 byte to 64 words are available.

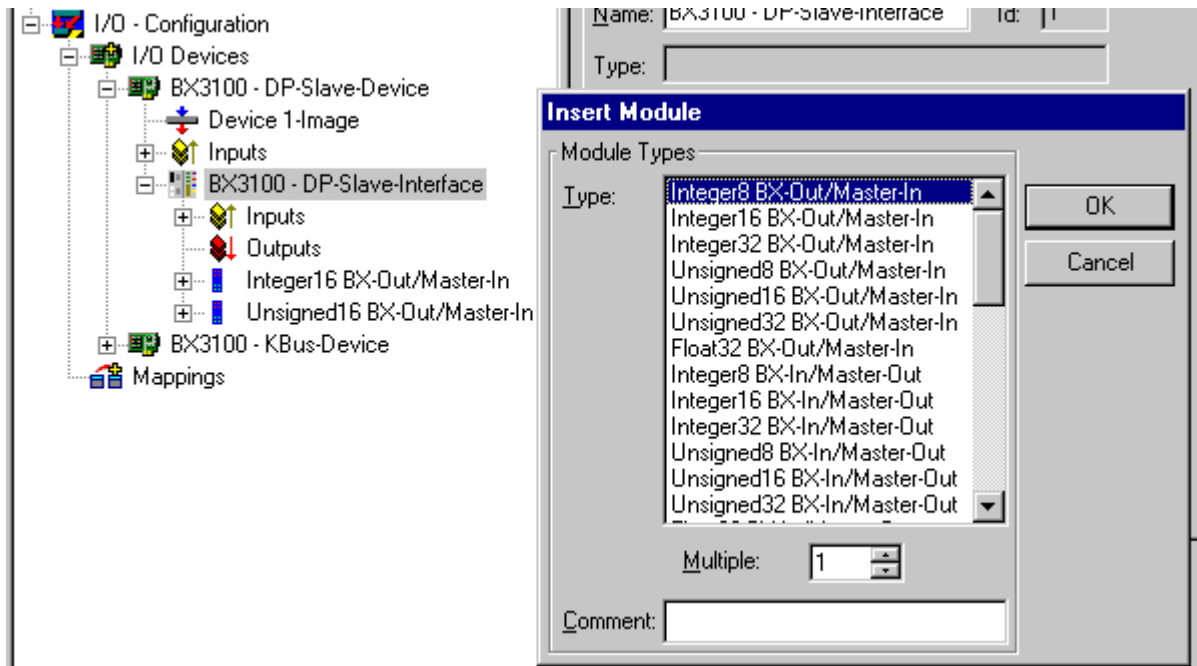


Fig. 34: Creating PROFIBUS data

ADS Communication has to be selected for preparing the configuration. If the configuration data are to be transferred via the fieldbus, data communication has to be active, i.e. the BX must be in active data exchange with the higher-level controller.

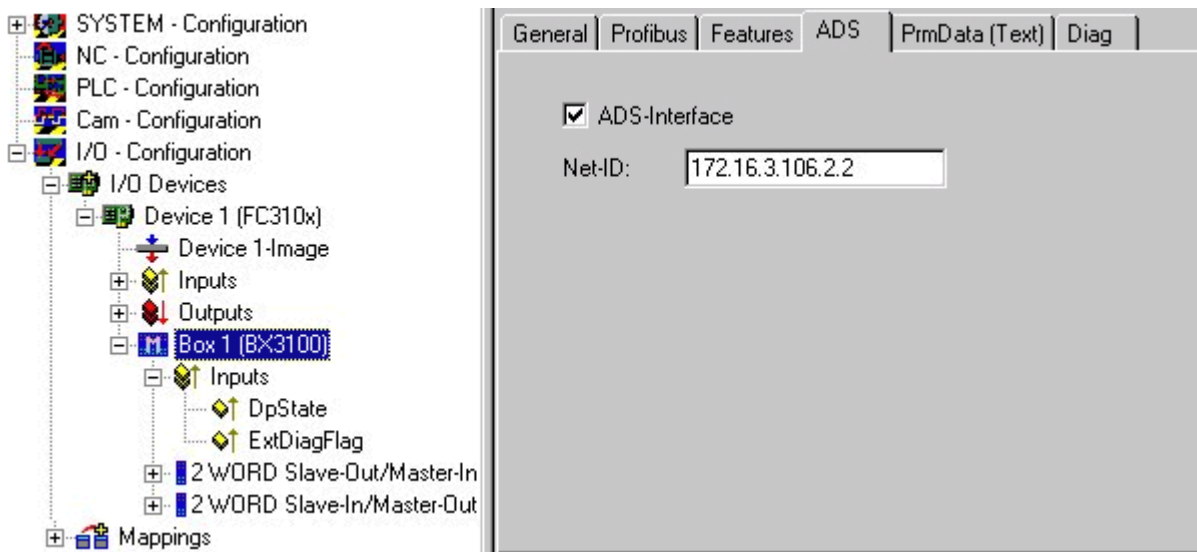


Fig. 35: ADS Interface

### 4.2.7.2.3 Variable mapping - PROFIBUS DP

Creating the PROFIBUS data.

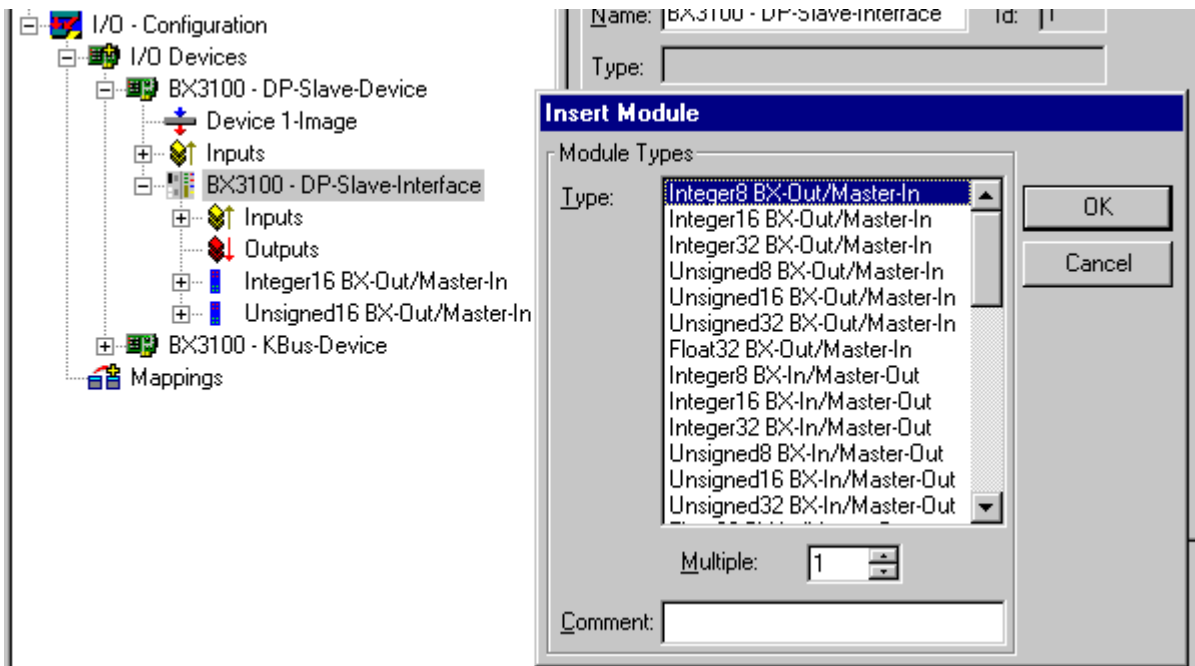


Fig. 36: Creating PROFIBUS data

For linking with the PLC project, the PLC program must be read in. Add your PLC program under PLC Configuration. The PLC variables can now be linked with the fieldbus variables.

**i Permitted number of PROFIBUS data**

- Ensure that the maximum number of PROFIBUS data is not exceeded.
- BX3100: max. 244 byte inputs and 244 byte outputs
- BC3150: max. 128 byte inputs and 128 byte outputs

#### 4.2.7.2.4 BX3100 at Siemens S7 controllers

### Configuration - Siemens S7 controller

#### Inserting the images

In order to assign an image to the devices in the Siemens software, the following graphics must be copied into the *Step7\S7Data\ncbmp* directory.

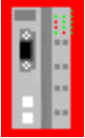


Fig. 37: <https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3733330315.zip>

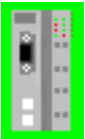


Fig. 38: <https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3733334667.zip>



Fig. 39: <https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3733337227.zip>

#### Inserting the GSD files

- Go to *Extras\Install new GSD* in the hardware catalog for your Step7.
- Select the directory in which the BECKHOFF GSD is located, and import the files.
- You will then find them in the hardware catalog under *Profibus DP\Other field devices\I/O*.

## Configuration: Siemens S7 controller with BX3100

### BX3100 parameter data

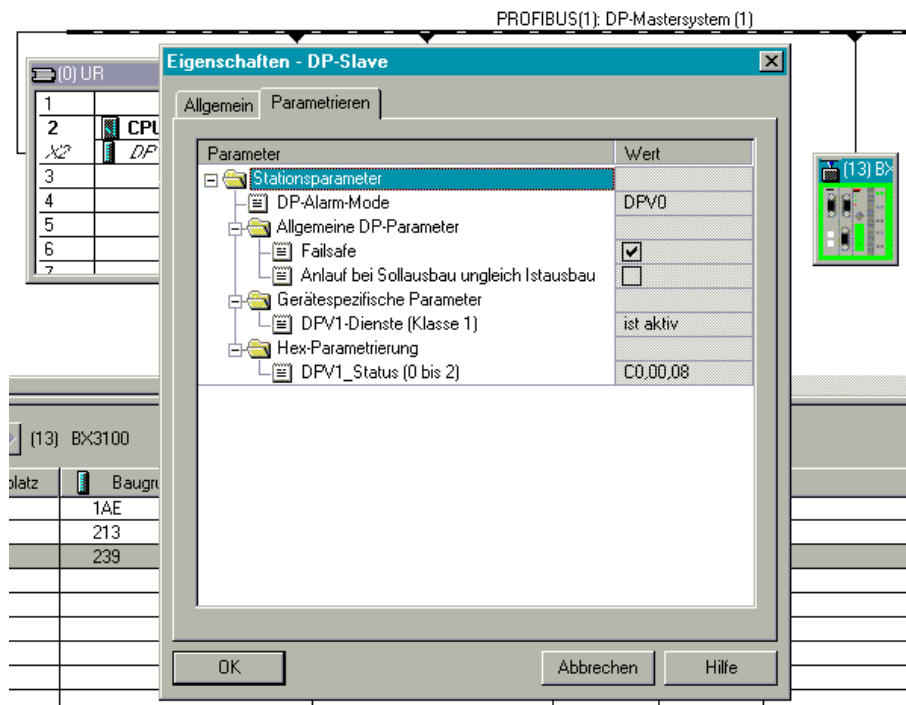


Fig. 40: Settings

### BX3100 module configuration

#### Sample 1

1 x BX3100

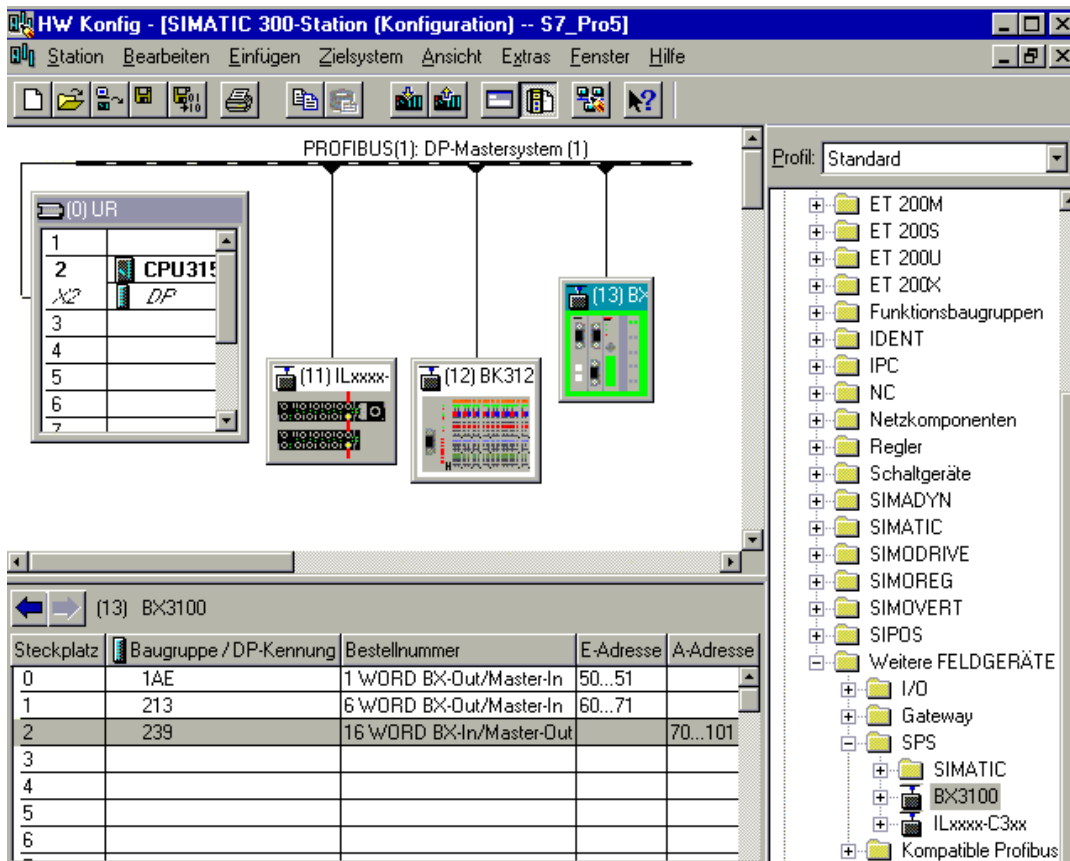


Fig. 41: Sample for entering individual bytes.

**i** **byte for byte**  
Each individual byte requires one byte of ConfigData.

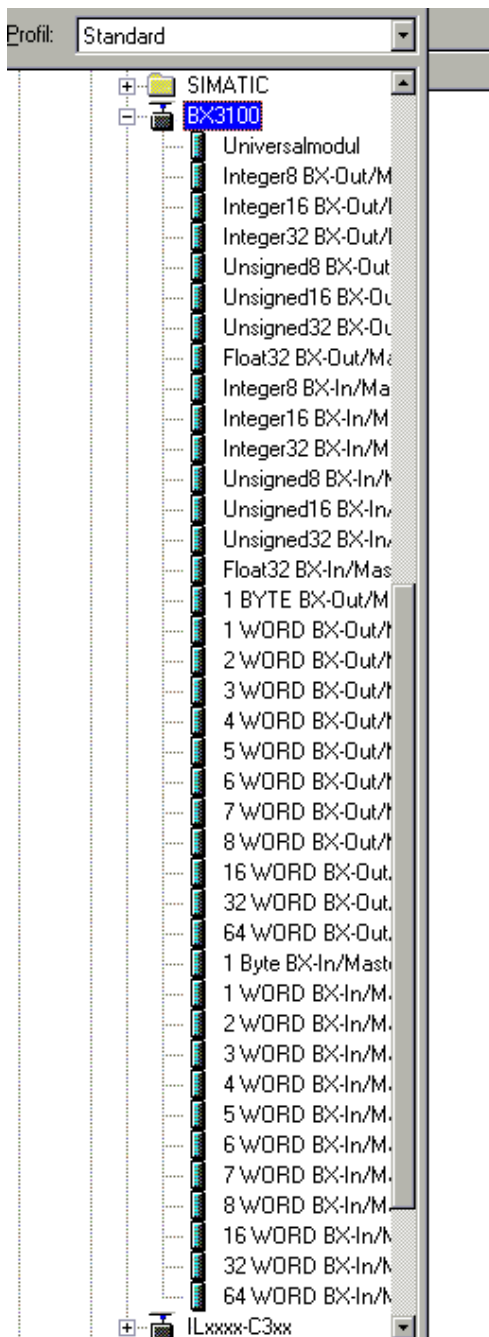


Fig. 42: Sample for entering associated bytes.

## 4.2.8 K-bus

### **i** Bus Terminal and end terminal required

To operate a Bus Terminal Controller of the BC or BX series, at least one Bus Terminal with process image and the end terminal must be connected to the K-bus.

#### BX Settings tab

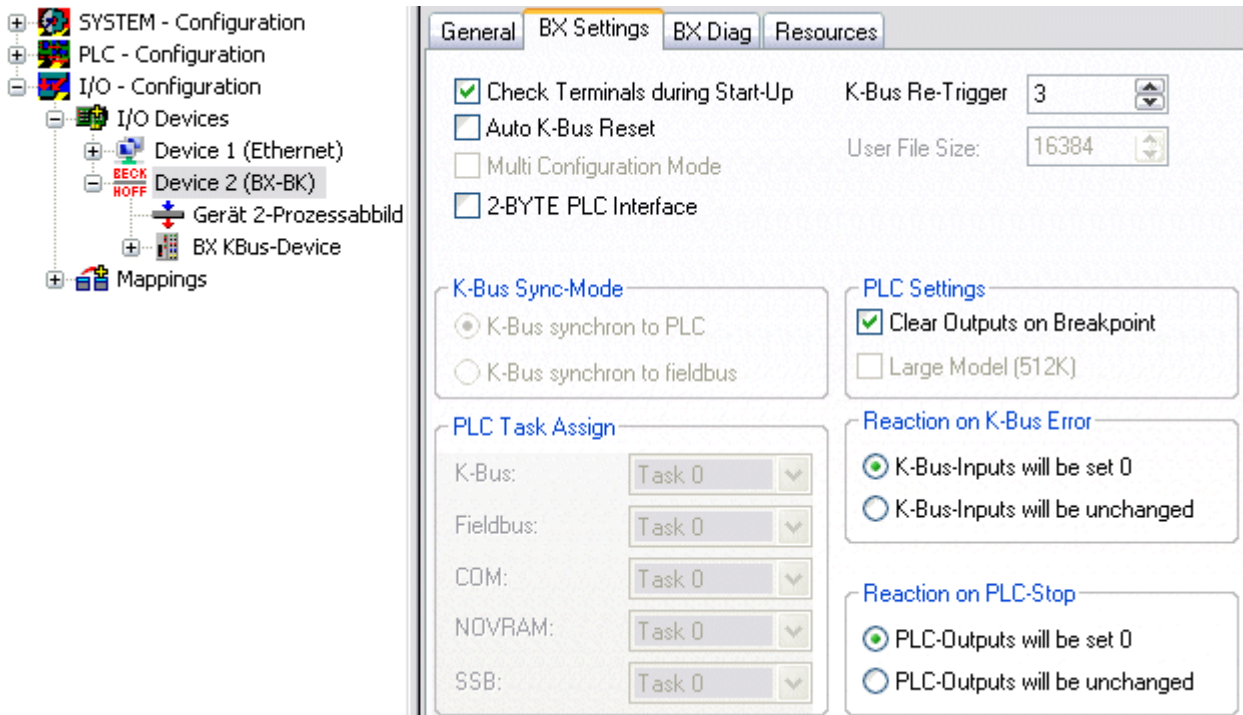


Fig. 43: BX Settings tab

#### Check Terminals during Start-up

When a boot project is created, the current Bus Terminal configuration is stored. The connected Bus Terminals are checked when the Bus Terminal Controller restarts. If this option is selected, the Bus Terminal Controller does not enter into data exchange. The PLC project will not be started.

#### Auto K-Bus Reset

Once a K-bus error has been rectified, the Bus Terminal Controller automatically resumes the data exchange.

#### **⚠ CAUTION**

#### **Once a K-Bus error has been rectified, the outputs become active again immediately!**

Ensure that the outputs are reactivated immediately and that analog outputs retain their programmed value, if this is not dealt with in your PLC project.

#### Clear Outputs on Breakpoint

If breakpoints are set in PLC Control, the K-Bus is no longer processed, i.e. the outputs are set to a safe state (zero).

#### K-Bus Sync Mode

Writing and reading of the Bus Terminals can take place synchronously with task 1 or the fieldbus.



### K-Bus Re-Trigger

If the processor is busy dealing with the PLC project or the SSB, the K-Bus cannot be processed for a certain amount of time. This leads to triggering of the Bus Terminal watchdog and dropping of the outputs. The Bus Terminal Controller is set such that the K-bus watchdog is re-triggered 3 times after 85 ms. The K-Bus watchdog would then be activated.

K-Bus Re-Trigger 0: 100 ms

K-Bus Re-Trigger 1:  $2 \times 85 \text{ ms} = 170 \text{ ms}$

K-Bus Re-Trigger 2:  $3 \times 85 \text{ ms} = 255 \text{ ms}$

K-Bus Re-Trigger 3:  $4 \times 85 \text{ ms} = 340 \text{ ms}$

### Reaction on K-Bus Error

In the event of a K-Bus error, the K-Bus inputs are set to "0" or retain their last state.

### Response on PLC-Stop

The user can set the behavior of the fieldbus output data in the event of the PLC project being stopped. The master will use these data as input data. In the event of a PLC stop, the data can be set to "0" or remain unchanged.

### BX Diag tab

Display of the cycle time for task 1, K-bus, fieldbus processing and the SSB load.

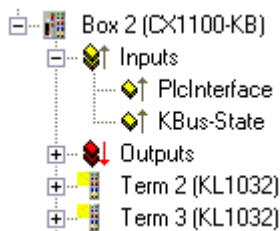
	Actual Value	Maximum Value
PLC-Task 1 (μs):	72	144
PLC-Task 2 (μs):		
PLC-Task 3 (μs):		
PLC-Task 4 (μs):		
K-Bus (μs):	246	303
Fieldbus (μs):	21	31
SSB (μs):		
SSB-Overhead (%):		
Display 1:	TWINCAT-CONFIG	
Display 2:	BC9020PROJEKT	

Fig. 44: BX Diag tab

*Factory Settings:* the Bus Terminal Controller is set to its delivery. These settings are reactivated via Restart System or by switching the system off and on again (display shows DEFAULT-CONFIG).

*Reset Maximum Values:* resets the maximum values

## K-Bus variables



**PLC interface:** Not supported (only included for moving CX or BX projects)

**K-bus state:** see Diagnostics

## 4.2.9 PLC

### 4.2.9.1 Inserting a PLC project

For variable mapping, configuration has to be specified in the system manager. This is where the link between PLC and hardware is specified. The variables can process and link bit, byte, word or data structures. Automatic addressing via the System Manager is possible, but should be checked for offset.

#### **i** Word alignment, byte orientation

With data structures, ensure that the Bus Terminal Controller saves the data in word alignment and the System Manager operates byte-oriented (see [Data structures](#) [► 65])

A valid project has to be compiled and saved in PLC Control. These data are saved as a \*.tpy file. For inserting a PLC project, right-click on *PLC - Configuration*. Select your current PLC project.

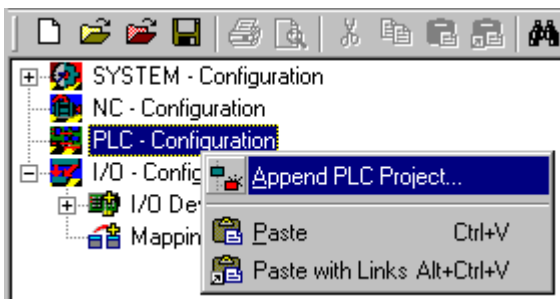


Fig. 45: Selecting the PLC project

Link the PLC variable with the hardware (e.g. digital Bus Terminal).

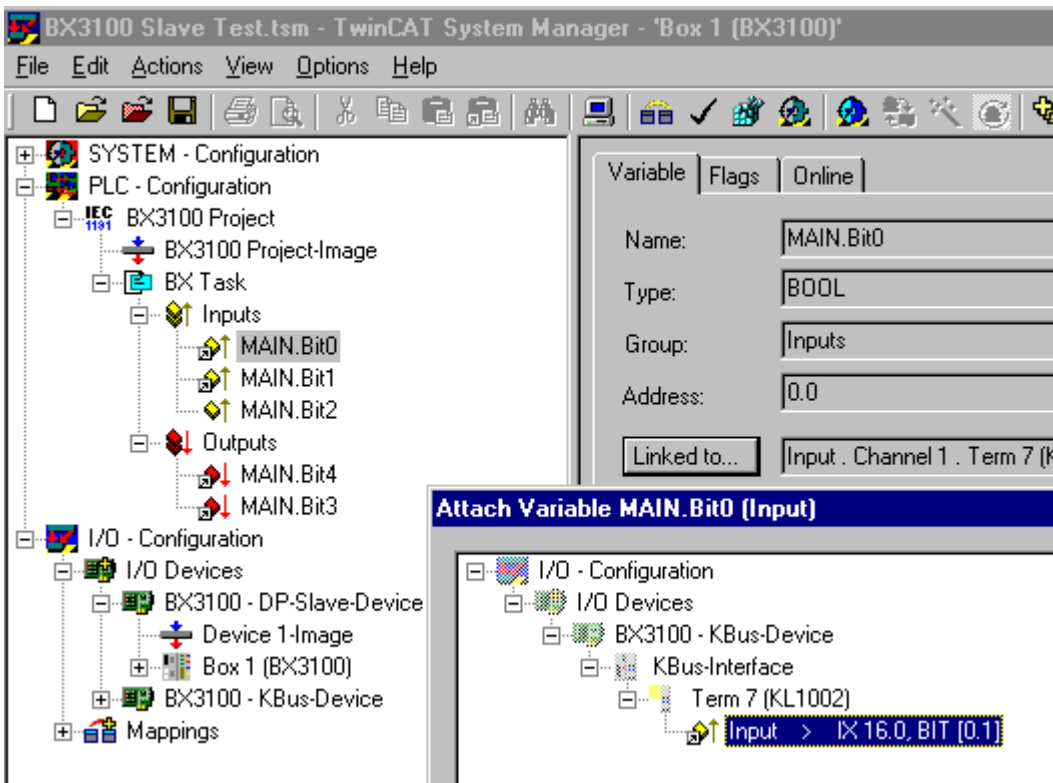


Fig. 46: Connecting PLC variable and hardware

Once all links have been created, activate the configuration *Actions/Activate Configuration (Ctrl+Shift+F4)* and start TwinCAT *Set/Reset TwinCAT to Run Mode*. Ensure that you have selected the correct target system (bottom right in the System Manager window).

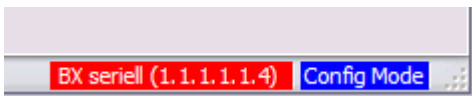


Fig. 47: Target system display

### 4.2.9.2 Measuring the PLC cycle time

The task time is set in PLC Control. The default setting is 20 ms.

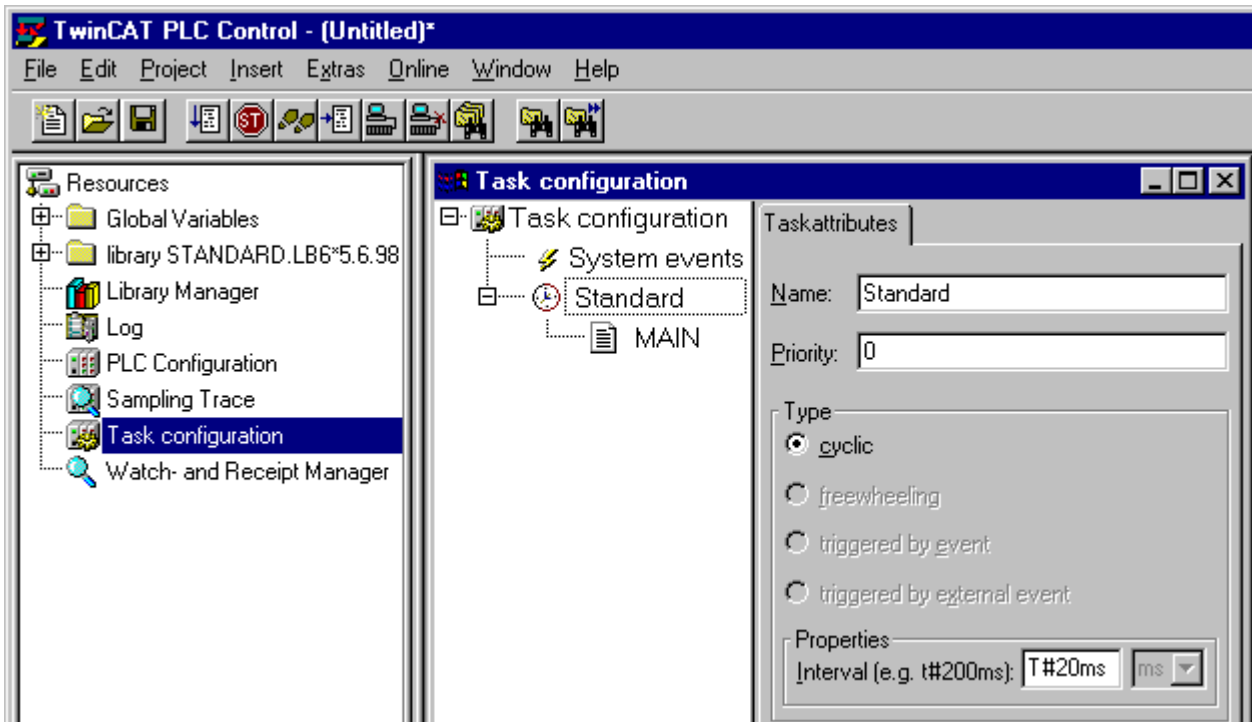


Fig. 48: Setting the task time

In the default setting, the PLC program is called every 20 ms, as long as the general cycle time is less than 20 ms. To determine the load of your system, the PLC cycle time can be measured in the System Manager. In order to ensure trouble-free operation, the set task time should be 20-30 % higher than the measured total cycle time. A precise cycle time breakdown can be found under [K-Bus tab \[► 48\]](#) description. The total cycle time is displayed with the TcBase library (see TcBase.lbx or TcBaseBCxx50.lbx).

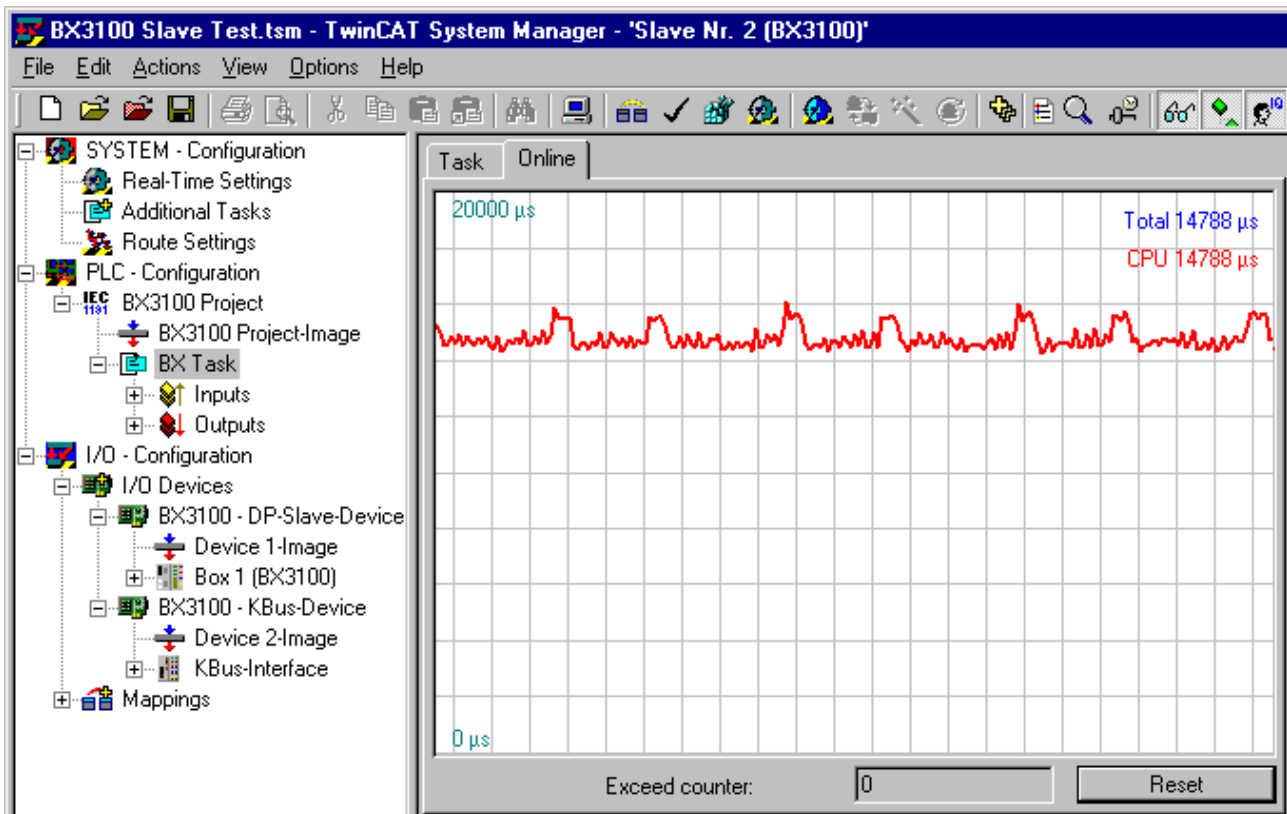


Fig. 49: Displaying the PLC cycle time

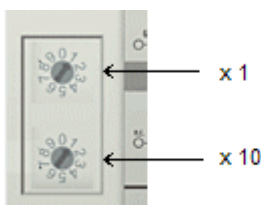
### 4.2.10 Setting the slave address

The address must be set via the two rotary selection switches. The default setting is 11. All addresses are permitted, although each address may only occur once within the network. For changing an address the Bus Coupler must be switched off. The switches can be set to the required position using a screwdriver. Ensure that the switches engage correctly. The lower switch is the ten-multiplier, the upper switch is the one-multiplier. The change in address is active as soon as the module is switched on.

#### Example

You want to set address 34:

- Lower rotary selection switch Sx11: 3
- Upper rotary selection switch Sx10: 4



#### Special functions using the address switch

In addition, the address switch can be used for following functions. To this end the associated address has to be set (as long as the coupler is switched off), only the end terminal may be connected (KL9010), and the coupler must then has to be connected to the supply voltage.

- Address 99: Manufacturer's setting
- Address 98: Delete boot project
- Address 97: Delete TwinCAT configuration

## 4.3 Configuration software KS2000

Bus Terminal controllers of the BCxx50, BXxx20 and BXxx00 series cannot be parameterized and configured with the KS2000 configuration software. These devices must be configured with the TwinCAT System Manager.

The KS2000 configuration software offers configuration and diagnostic support for the Bus Terminals attached to the Bus Terminal Controller.

It is advisable to set the baud rate in the KS2000 configuration software and the BCxx50 BCxx20 and BXxx00 to 38400 baud (8 data bits, even, 1 stop bit).

---

### ● COM1 - automatic baud rate detection



The COM 1 interface of the BXxx00 features automatic baud rate detection between 9.6 kbaud and 56.4 kbaud.

---

### ● Required KS2000 version



Configuration or diagnostics of Bus Terminals at BXxx00 is supported from KS2000 version 4.3.14.

---

In some Bus Terminals (e.g. KL25xx, KL6811, KL6201, KL6401) the following parameters must be set in order to be able to use the configuration dialogs:

- A PLC project or boot project must be deactivated.
- The BX controller must be in its default configuration. Set the manufacturer's setting or switch to Config Mode in the TwinCAT System Manager (blue TwinCAT icon).
- The BX controller must be in FreeRun mode. Activate it with the TwinCAT System Manager.

You can now log in with the KS2000 configuration software via ADS (port 100) or the serial cable and use the KS2000 dialogs in the Bus Terminals.

## 4.4 Programming cable

Use the KS2000-Z2 programming cable for serial programming of the Bus Terminal Controller. This cable is included in the scope of supply of the KS2000 software, or it can be ordered separately (order identifier KS2000-Z2).



Fig. 50: Programming cable KS2000-Z2

### KS2000-Z2

The programming cable offers the option of programming the BCxx50 via the serial interface.

#### **NOTE**

#### **Do not interrupt the ground connection for the supply voltage**

When the programming cable (between BCxx50 and PC) is connected, the ground connection of the Bus Terminal controller must not be interrupted or disconnected, since this may destroy the programming cable.

## 5 Programming

### 5.1 BCxx50 PLC features

Description	Value
Data memory	32 kbyte
Program memory	48 kbyte minus task-configuration minus POU's during online change
Source code memory	128 kbyte
RETAIN	2 kbyte
INPUT	2 kbyte
OUTPUT	2 kbyte
FLAG	4 kbyte
Max. variable size	16 kbyte
Max. POU's	Limited by memory

### 5.2 TwinCAT PLC

The Beckhoff TwinCAT Software System turns any compatible PC into a real-time controller with a multi-PLC system, NC axis control, programming environment and operating station. The TwinCAT programming environment is also used for programming the BC/BX. If you have TwinCAT PLC (Windows NT4/2000/XP) installed, you can use the fieldbus connection or the serial port for downloading and debugging software.

TwinCAT I/O or TwinCAT PLC can also be used as the Ethernet Master (host), in order to exchange process data with the Bus Terminal Controller. TwinCAT provides you with the System Manager as a configuration tool, as well as the drivers and the ADS protocol.

#### Bus Terminal Controllers of the BCxx50, BCxx20 and BXxx00 series

These 2nd-generation Bus Terminal Controllers are configured with the TwinCAT System Manager and programmed with TwinCAT PLC Control. TwinCAT PLC must be installed for these couplers (Windows NT4, Windows 2000, Windows XP).

#### Programming and program transfer

- [via the serial interface \[► 75\]](#)
- via the fieldbus interface (only for Bus Terminal controllers for PROFIBUS, CANopen and Ethernet)

#### Online change

The Bus Terminal Controllers of the BX series and the BCxx50 support online change. This means that the PLC program is replaced with a new program without interrupting the program. The switch-over to the new program occurs after the task is completed. This means that two versions of the PLC program have to be stored. 512 kbyte are available, which therefore have to be divided by two, leaving 256 kbyte for the actual PLC program. In addition, several kbyte are required for task configuration etc. During an online change, dynamic data are stored in memory. Should a program approach the memory limit (program size greater than 240 kbyte), the online change may no longer work, even though the program may still be written to the BX after "Rebuild all".

#### When is online change not available?

Online change is not available under certain conditions,.

- Inserting of a new library
- Changing the task setting



- "Rebuild all"
- Controller memory limit is almost reached (PLC program greater than 90%)

### 5.3 TwinCAT PLC - Error codes

Error type	Description
PLC compiler error	Maximum number of POU's (...) exceeded
PLC compiler error	Out of global data memory ...

#### Error POU's

For each function block one POU (process object unit) is created. 256 function blocks are available by default.

**Error 3612: Maximum number of POU's (100) exceeded! Compile is aborted.**

Data allocation

1 Error(s), 0 Warning(s).

Fig. 51: Maximum number of POU's exceeded

If libraries are integrated this value may be insufficient. In this case, the number of POU's should be increased.

To this end, open in PLC Control under Projects/Options...

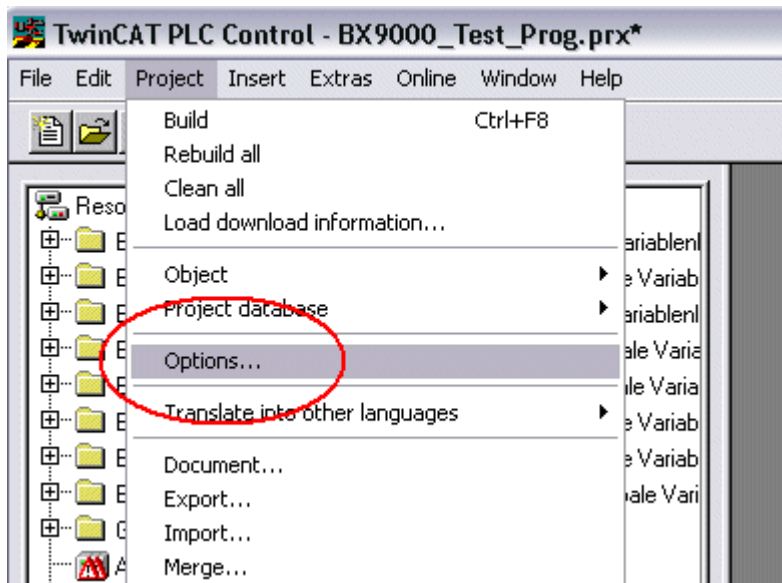


Fig. 52: Menu path Projects / Options / Controller Settings

...the controller settings.

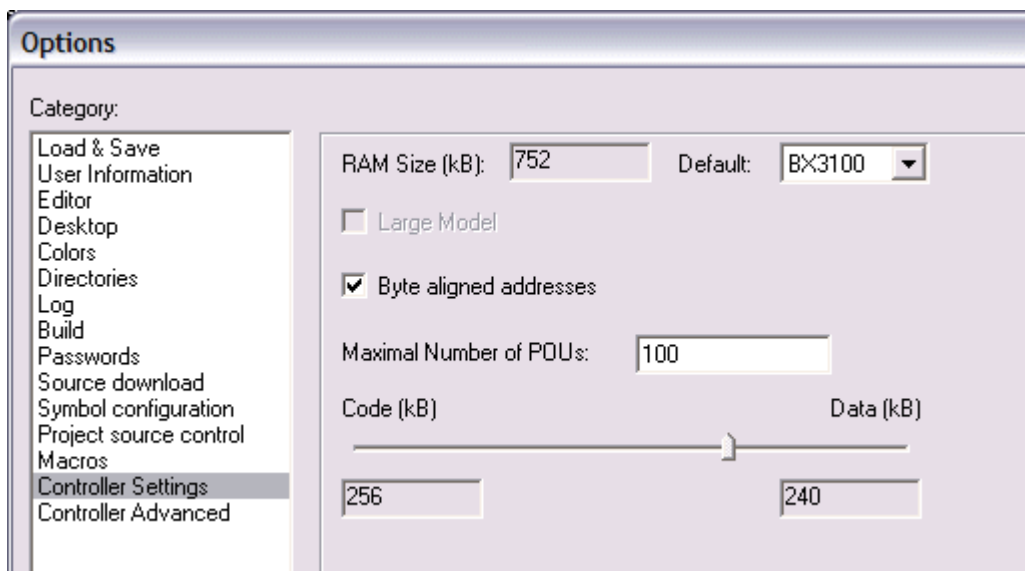


Fig. 53: Controller settings

Changing these settings will deactivate online changes.

**Global memory error**

```
Interface of POU 'MAIN'
Data allocation
Error 3803: MAIN (7): Out of global data memory. Variable 'Test_', 16002 bytes.
1 Error(s), 0 Warning(s).
```

Fig. 54: Global memory insufficient

2 x 16 kbyte of data are available by default. If large data quantities are to be used, this range should be increased. A maximum of 14 data segments are possible for the BX.

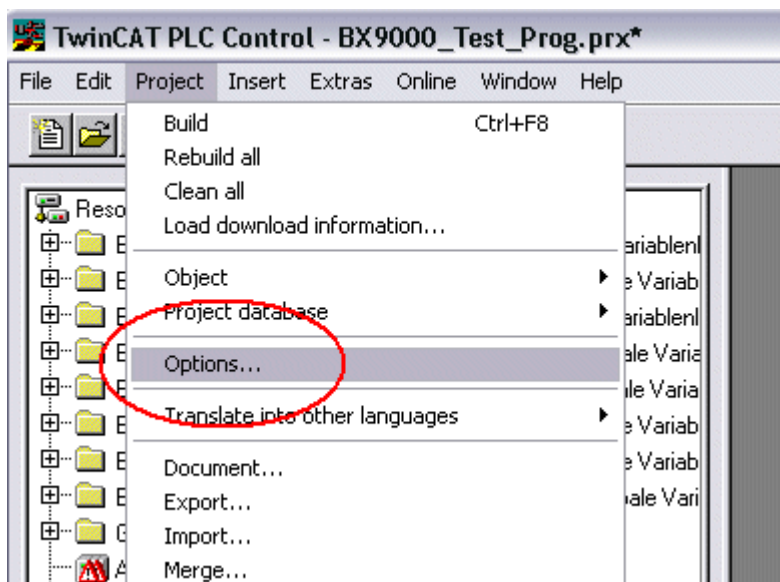


Fig. 55: Menu path Projects / Options / Build

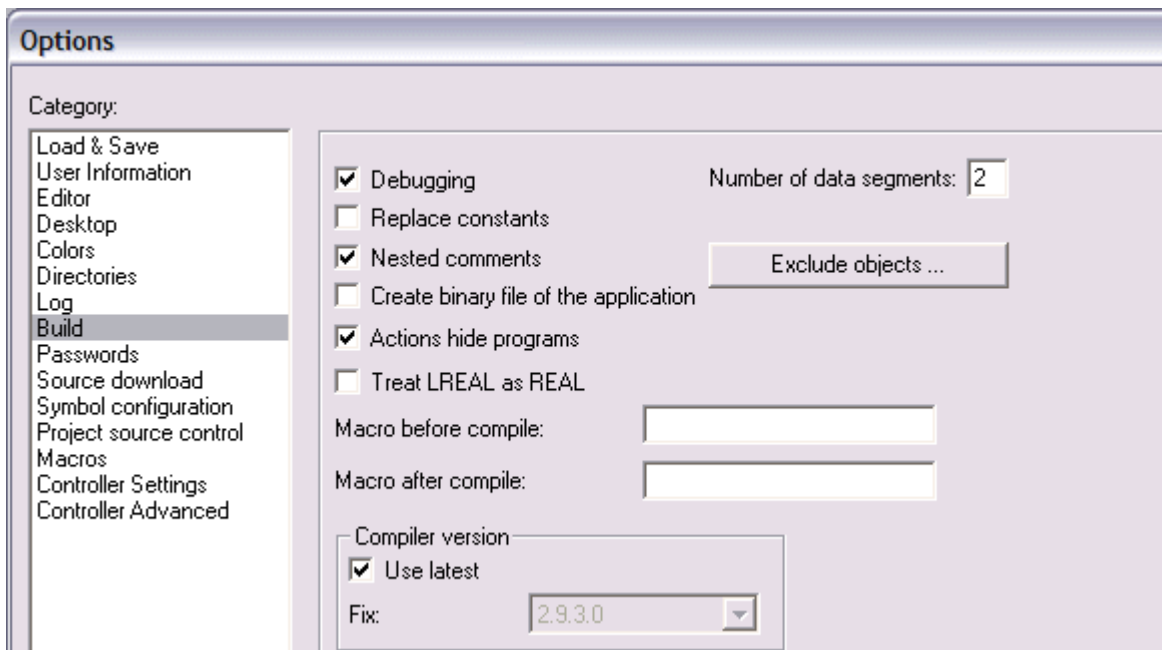


Fig. 56: Build

## 5.4 Remanent data

2000 bytes of remanent data are available on the BC9191 and the BX controller. These data are declared as VAR RETAIN in PLC Control.

### Example

```
VAR RETAIN
  Test      :BOOL;
  Count     :INT;
END_VAR
```

Retain data are located between VAR RETAIN and END\_VAR. These data are stored in a NOVRAM and are consistent across the whole 2 kbyte range. The RETAIN data are stored in the NOVRAM after each cycle. For 2 kbyte approx. 2 ms are required (for 1 kbyte approx. 1 ms). The variables can be configured locally or globally. Allocated variables (%MB, %QB, %IB) cannot be used as remanent data.

#### ● Do not use VAR\_RETAIN in function blocks

**i** VAR\_RETAIN should not be used in function blocks. All FB data are copied into the retain memory. This leads to an unnecessary increase in cycle time, and the retain memory is filled with unnecessary data.

#### ● Do not use variables with address as remanent data

**i** Variables that have been assigned an address (%MB, %QB, %IB) must not be used as remanent data.

### Example for remanent data in the function block

This should be avoided, if possible, since all the data of a function block, in which even just a single remanent bit is found, are stored by default. A program sample can be found below.

#### Function block test (no program code required - in ST semicolon is sufficient)

```
FUNCTION_BLOCK Test
VAR_INPUT
END_VAR
VAR_OUTPUT
END_VAR
VAR
END_VAR
VAR_IN_OUT
  Counter   :INT;
END_VAR
```

#### MAIN program

```
PROGRAM MAIN
VAR
  fb_Test:Test;
END_VAR
VAR_RETAIN
  iCounter1:INT;
END_VAR
fb_Test(Counter:=iCounter1);
```

## 5.5 Allocated flags

4 kbyte of allocated flags are available. They can be used to assign different variable types to the same address, e.g. for converting strings to bytes. Data can also be placed here that can be read or written via ADS by the controller.

**● Allocated variables are not remanent data**



For the Bus Terminal Controllers of the BX series and the BCxx50 the allocated variables are **not** saved as remanent data.

### Reading/writing of allocated flags via ADS

The flags may also be read via the controller and ADS. In PROFIBUS, the DPV-1 services are used for this purpose, in CANopen SDO communication is used.

The AmsNetID can be obtained from the System Manager, or it can be displayed via the Bus Terminal Controller menu.

The PLC port number is 800.

Index group	Meaning	Index offset (value range)
0x4020	Flag (only BXxxx0)	0..4096

### Example

#### BX program

```
VAR
    Flag_01 AT %MB0: WORD;
END_VAR
```

#### TwinCAT PC/CX master program

```
VAR
    fbADRSREAD: ADSREAD;
    Flag_M: WORD;
END_VAR

fbADRSREAD (
    NETID:='172.16.3.0.2.3' , (* AMSNetId BX *)
    PORT:=800 , (* 800 - PLC *)
    IDXGRP:=16#4020 , (* 0x4020hex falgs *)
    IDXOFFS:=0 , (* byte offset *)
    LEN:=2 , (* Lenght byte *)
    DESTADDR:=ADR(Merker) ,
    READ:=TRUE ,
    TMOUT:=t#1s );
IF NOT fbADRSREAD.BUSY THEN
    fbADRSREAD(READ:=FALSE);
END_IF
```

## 5.6 Local process image in delivery state

The process image of the Bus Terminal Controller consists of input, output and flag area. In addition, there are unallocated data without fixed address. They are created without specifying an address. For these variable types the memory allocation is as follows:

- BCxx50 48 kbyte,
- BC9x20 128 kbyte,
- BXxx00 256 kbyte.

The maximum size of a variable or structure (array) is 16 kbyte. For the allocated data 2048 bytes of input data and 2048 bytes of output data are available. The Bus Terminal Controller has 4 kbyte of memory allocated for the flag area.

In the delivery state (default configuration) of the BX/BCxx50, fixed addresses are allocated for all connected Bus Terminals. The data for PROFIBUS communication start from address offset 1000<sub>dec</sub>. The length of the PROFIBUS data depends on the number of configured data. The maximum is 244 bytes for the BX3100 and 128 bytes for the BC3150.

Inputs	Outputs
Bus Terminal %IB0 ...	Bus Terminal %QB0 ...
PROFIBUS DATA (PLC variable) %IB1000 ...	PROFIBUS DATA (PLC variables) %QB1000 ...
... %IB2047 Maximal	... %QB2047 Maximal

### Addressing of the connected Bus Terminals

The default setting is for all the connected Bus Terminals to be assigned to the local process image. Mapping within the Bus Terminal Controller is carried out according to the following rule: First all the complex Bus Terminals, in the sequence they are physically inserted, followed by the digital Bus Terminals which are filled to a byte. The default mapping of the complex Bus Terminals is:

- complete evaluation
- Intel format
- Word alignment

### Example structure

Bus Terminal Controllers: 1 x BX/BCxx50  
 Position 1: 1 x KL1012  
 Position 2: 1 x KL1104  
 Position 3: 1 x KL2012  
 Position 4: 1 x KL2034  
 Position 5: 1 x KL1501  
 Position 6: 1 x KL3002  
 Position 7: 1 x KL4002  
 Position 8: 1 x KL6001  
 Position 9: 1 x KL9010

### Process image

Bus Terminal	Position	Input image	Output image
KL1501	5	%IB0...%IB5	%QB0...%QB5
KL3002	6	%IB6...%IB13	%QB6...%QB13
KL4002	7	%IB14...%IB21	%QB14...%QB21
KL6001	8	%IB22...%IB29	%QB22...%QB29
KL1012	1	%IX30.0..30.1	-
KL1104	2	%IX30.1..30.5	-
KL2012	3	-	%QX30.0..30.1
KL2034	4	-	%QX30.2..30.5
KL9010	9	-	-

## 5.7 Mapping the Bus Terminals

The precise assignment of the byte-oriented Bus Terminals may be found in the configuration guide for the particular bus terminal. This documentation can be found on the Internet at <http://www.beckhoff.de>.

Byte oriented Bus Terminals	Bit oriented Bus Terminals
KL15x1	KL10xx, KL11xx, KL12xx, KL17xx, KM1xxx
KL25xx	KL20xx, KL21xx, KL22xx, KL26xx, KL27xx, KM2xxx
KL3xxx	
KL4xxx	
KL5xxx	
KL6xxx	
KL7xxx	
KL8xxx	
	KL9110, KL9160, KL9210, KL9260

## 5.8 Local process image in the TwinCAT configuration

The TwinCAT configuration (TwinCAT CONFIG) enables free mapping between fieldbus, K-bus and PLC variables. Variables can be linked independent of their address via the System Manager.

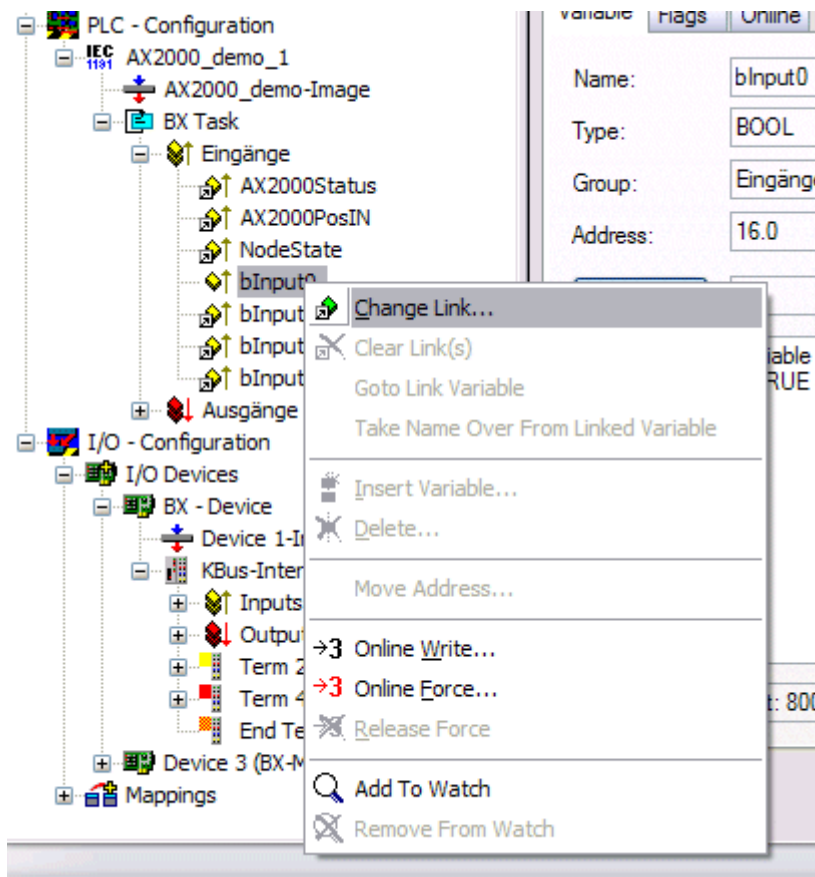


Fig. 57: Changing variable links

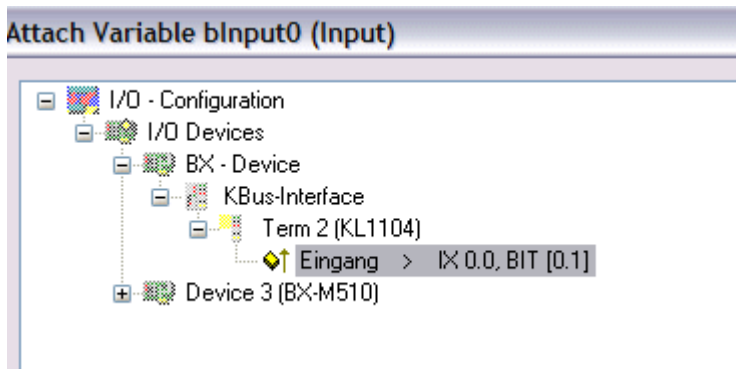


Fig. 58: Linking a variable with an input

In the default configuration all Bus Terminals are assigned fixed addresses. If a Bus Terminal is inserted, the whole address range may be shifted. The TwinCAT configuration enables allocated variables to be linked to a Bus Terminal, as required. This is parameterized in the System Manager, and the configuration is then downloaded to the Bus Terminal Controller (see [TwinCAT configuration](#) [▶ 29]). It is also possible to upload an existing TwinCAT configuration.

## 5.9 Creating a boot project

The following memory resources are available for generating the boot project

- approx. 250 kbyte flash on the Bus Terminal controllers of the BX series;
- approx. 48 kbyte flash on the Bus Terminal controllers of the BCxx50 series.

### PLC Control

After logging into TwinCAT PLC Control, a boot project can be created.

- Opening a PLC project
- Selecting the target system (or selection the serial interface)
- Logging into the BX/BCxx50
- Creating a boot project (Online>Create boot project)

The PLC LED lights up green once a valid boot project is available on the BX/BCxx50.

In the Bus Terminal controllers of the BX series, the PLC LED flashes orange while boot project is created. The PLC LED lights up orange if no boot project is available on the BX.

### Deleting a boot project

The boot project can be deleted from the Bus Terminal Controller. The following steps must be followed:

- Opening the project
- Logging into the Bus Terminal Controller
- Deleting the boot project (Online>Delete boot project)

The PLC LED lights up orange when the boot project is deleted.

### ● Using the current project as boot project

**I** After an online change the old project is still shown as boot project. To use the current project (after the online change) as the boot project, the boot project has to be recreated.

### Bypassing the start of the boot project\*

With the Bus Terminal controllers of the BX series, starting of the boot project during booting can be prevented by pressing the Navi button. This does not delete the boot project. The project is reloaded when the Bus Terminal Controller is rebooted.



\* from version 0.85

## 5.10 Communication between TwinCAT and BX/BCxx50

For transferring data from TwinCAT to the Bus Terminal Controller, it makes sense to organize the data in a structure. Please note the following to account for the differences in data management on the two systems.

- If two different data types are sent in sequence (e.g. byte and INT), the following variable is set to the next even address offset
- Boolean variables should never be allocated individually within a structure, since they would invariably occupy 1 byte. Boolean expressions should always be masked in a byte or word.

### Example 1: A structure on the BX/BCxx50 and on the PC

Variable	BX/BCxx50 memory	PC memory (TwinCAT)
Byte	%..B0	%..B0
INT (1)	%..B2	%..B1
INT (2)	%..B4	%..B3

Due to the fact that another variable type (INT) follows the first byte, in the BX/BCxx50 it was assigned the next free even address. In order to achieve the same data structure on both systems, a dummy byte has to be inserted in the PC project (see example 2).

### Example 2: A structure on the BX/BCxx50 and on the PC with the same memory allocation

Variable	BX/BCxx50 memory	PC memory (TwinCAT)
Byte	%..B0	%..B0
Byte (dummy)	%..B1 (not necessarily required, since the system deals with this itself if the variable does not exist)	%..B1
INT (1)	%..B2	%..B2
INT (2)	%..B4	%..B4

### Data structure

```
Type PB_Data
STRUCT
  wVar_1:WORD;
  iValue_1:INT;
  iValue_2:INT;
  iValue_3:INT;
END_STRUCT
END_TYPE
```

### Creating a variable structure

```
VAR_Global
  strData_Out AT %QB1000:PB_Data; (*PLC Variables *)
  bInput_01 AT %IX0.0:BOOL; (* Input from a terminal *)
END_VAR
```

### Small programming example

```
strData_Out.wVar_1.0:=bInput_01;
```

#### **i** Do not use real values in a mixed data structure

A mixed data structure should not contain real values. If this is nevertheless the case, the high and low words must be swapped in the BX/BCxx50 or in the TwinCAT master project. It is better to use an array of Real values or to transfer the Real values individually.

### **i** Larger fieldbus data blocks

You can transfer larger fieldbus data blocks, in order to have a reserve for your structure. Disadvantage: These reserves are then transferred with each fieldbus telegram, resulting in overload of the fieldbus communication.

## 5.11 Up- and downloading of programs

The Bus Terminal Controller has a memory for the source code. It can be used for storing the program, the task configuration, and the libraries. Should the memory be insufficient, the source code may be stored without task configuration and libraries. This takes up significant less memory space!

### General settings

The timing of the source code download to the target system can be specified via Edit/Options. Open the options menu.

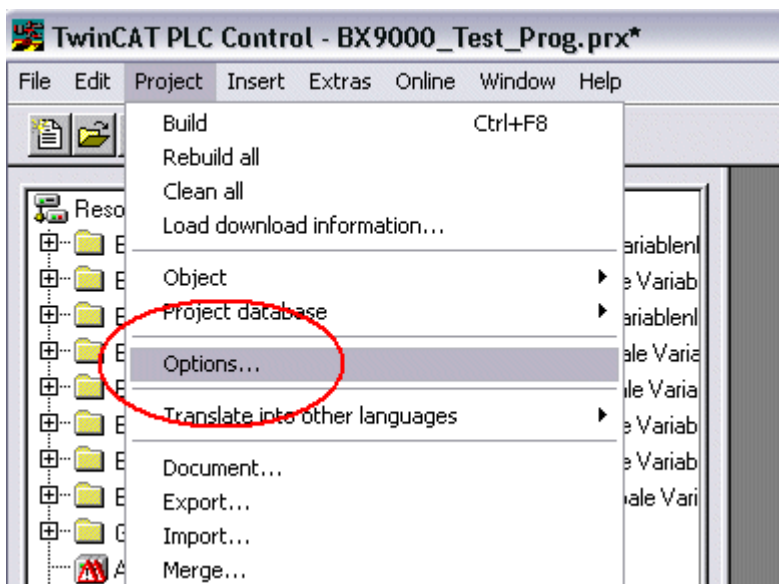


Fig. 59: Opening the options menu

Select Source Download.

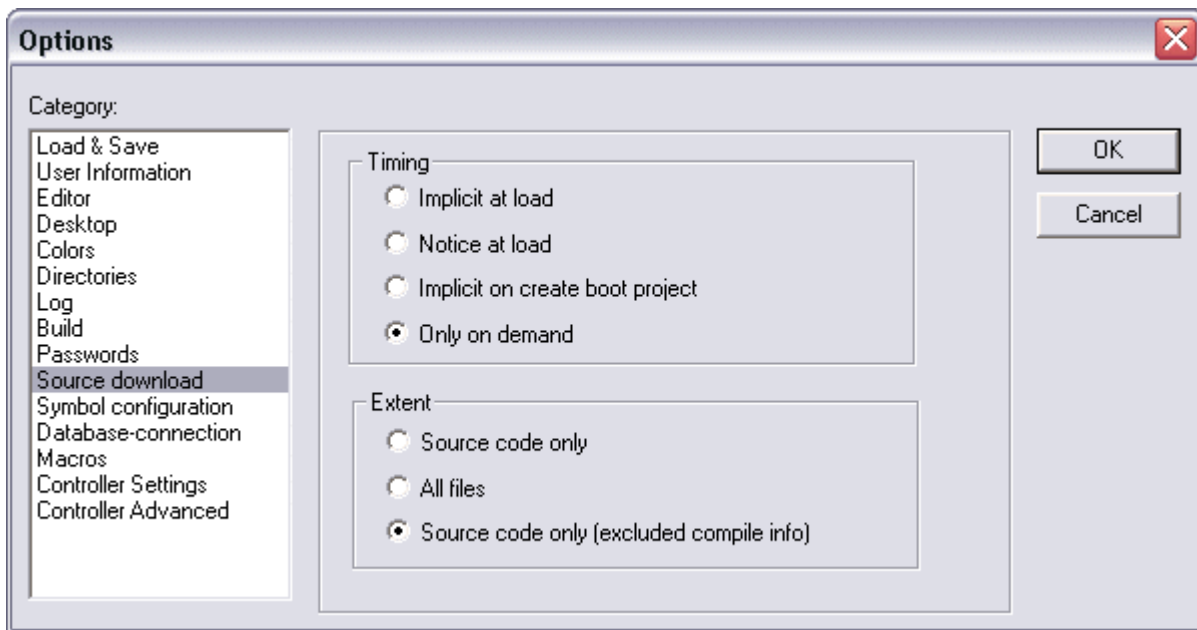


Fig. 60: Selecting Source Download

Here you can set which parts of the source code are to be downloaded to the Bus Terminal Controller, and when.

**Source code only:** the prx file with information on the online change is transferred. Login via online change is possible (the PLC does not stop).

**All files:** as *Source code only*, plus all required libraries.

**Source code only (compile info excluded):** only the prx file is transferred. Login is only possible when the PLC stops.

Which option you can use depends on the size of your projects.

### Downloading a program

The source code can be transferred to the target system on request. This requires the user to be logged in with his program. Under Online/Source code download the program code can now be transferred to the Bus Terminal Controller.

Online	Window	Help
Login		F11
Logout		F12
Download		
Run		F5
Stop		Shift+F8
Reset		
Reset All		
Toggle Breakpoint		
Breakpoint Dialog		F9
Step over		F10
Step in		F8
Single Cycle		Ctrl+F5
Write Values		
Force Values		Ctrl+F7
Release Force		F7
Write/Force-Dialog		Shift+F7
Write/Force-Dialog		
Write/Force-Dialog		Ctrl+Shift+F7
Show Call Stack...		
Display Flow Control		Ctrl+F11
Simulation Mode		
Communication Parameters...		
Sourcecode download		
Choose Run-Time System...		
Create Bootproject		
Create Bootproject (offline)		
Delete Bootproject		

Fig. 61: Downloading the program code

After a short delay, a window will open that indicates the download progress.

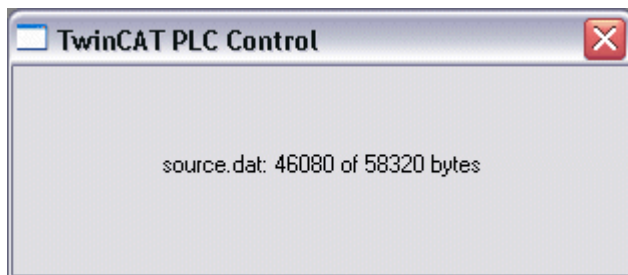


Fig. 62: Download progress

### Uploading a program

For uploading the program code again, open a new file in PLC Control. Then click on the PLC button.

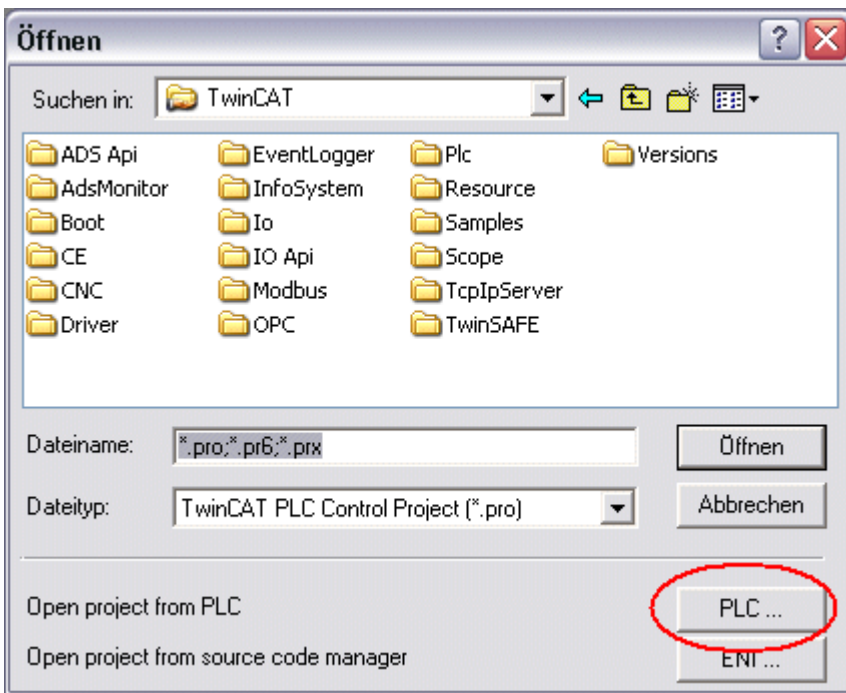


Fig. 63: Uploading a program

Select the data transfer route:

- *BCxx50 or BX via AMS*, if you are connected to the Bus Terminal Controller via the fieldbus, or
- *BCxx50 or BX via serial*, if you are connected to the Bus Terminal Controller via the serial interface.

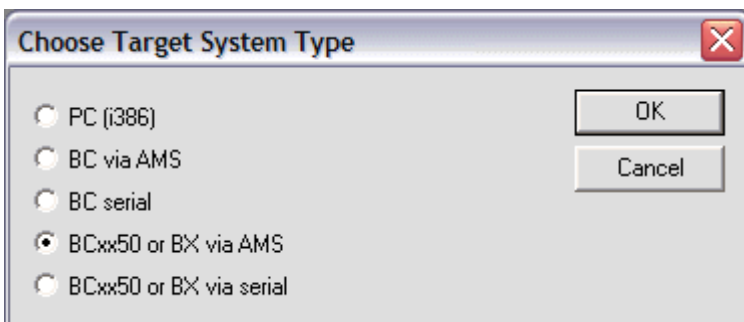


Fig. 64: Selecting the data transfer route

Then select the device and confirm with OK.

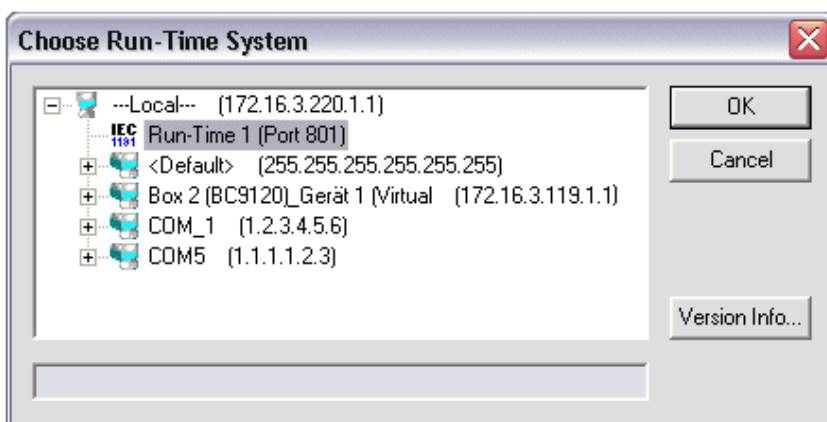


Fig. 65: Selecting the device

The source code will now be uploaded.

**Password**

You can protect your project with a password (in PLC Control Project/Options/Passwords).





## 5.12 Libraries

### 5.12.1 Libraries overview

The TwinCAT Automation Software offers various libraries for the BCxx50 series Bus Terminal Controllers (Bus Coupler with PLC functionality). See [Beckhoff Information System](#).

**Download**

The libraries are also included in this documentation. To extract the libraries, left-click on the link and copy the libraries to directory TwinCAT\PLC\LIB.

-  Standard  
(<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3730726795.zip>)
-  TcSystemBCxx50. TcSystemBCxx50 requires the TcBaseBCxx50 library.  
(<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3730728971.zip>)
-  TcBaseBCxx50  
(<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3730731147.zip>)
-  ChrAscBX.lbx  
(<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3730733323.zip>)

**● Use the library that matches the firmware**

**i** The latest firmware requires the latest library. If you update your Bus Terminal Controller, please also update the libraries.  
Copy these libraries to the LIB directory, remove these libraries from your project and add them again.

**TcSystemBCxx50**

ADS	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
ADSREAD		B0	B0	B1	B0	-
ADSWRITE		B0	B0	B1	B0	-
ADSRDWRT		B0	B0	B1	B0	-
ADSWRTCTL		B0	B0	B1	B0	-
ADSRDSTATE		B0	B0	B1	B0	-
ADSRDDEVINFO		B0	B0	B1	B0	-

Bit Functions	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
CLEARBIT32		B0	B0	B1	B0	-
CSETBIT32		B0	B0	B1	B0	-
GETBIT32		B0	B0	B1	B0	-
SETBIT32		B0	B0	B1	B0	-

Controller	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
FB_BasicPID	-	B0	B0	B1	B0	-
-	-	-	-	-	-	-

File Access	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
FB_ReadFromFile		-	-	-	-	-
FB_WriteToFile		-	-	-	-	-
FB_ReadWriteFile		-	-	-	-	-

Memory Functions	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
MEMCMP		B0	B0	B1	B0	-
MEMCYP		B0	B0	B1	B0	-
MEMMOVE		B0	B0	B1	B0	-
MEMSET		B0	B0	B1	B0	-

NOVRAM Functions	Version	Firmware				
		BX3100	BX5100	BX5200	BX8000	-
-	-	-	-	-	-	-

SFC	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
AnalyzeExpression		-	-	-	-	-
AppendErrorString		-	-	-	-	-
SFCActionControl		-	-	-	-	-

System / Time / TBus	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
DRAND		B0	B0	B1	B0	-
SYSTEMTIME_TO_DT		B0	B0	B1	B0	-
DT_TO_SYSTEMTIME		B0	B0	B1	B0	-
GetSysTick		B0	B0	B1	B0	-
PresetSysTick		B0	B0	B1	B0	-
Reboot		B0	B0	B1	B0	-

Debug	Version	Firmware				
		BC3150	BC5150	BC5250	BC8150	-
F_ReadDebugTimer		B0	B0	B1	B0	-
F_StartDebugTimer		B0	B0	B1	B0	-

## 5.12.2 TcBaseBX

### 5.12.2.1 BX debugging function

These functions can be used for measuring command execution times in a PLC project. The unit is a tick. One tick corresponds to 5.12 µs.

### Start Debug Timer function

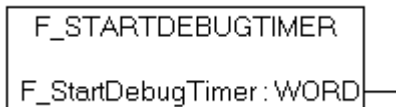


Fig. 66: Function block F\_STARTDEBUGTIMER

Calling this function starts the timer. The return value is "0".

### Read Debug Timer function

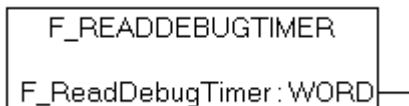


Fig. 67: Function block F\_READDEDEBUGTIMER

This function reads the timer value. The return value has to be multiplied with 5.12  $\mu$ s.

### Example

```
VAR  
    Timer_BX      :WORD;  
    i              :INT;  
END_VAR
```

### Program

```
F_STARTDEBUGTIMER();  
For i:=0 to 1000 do  
    ;  
END_FOR  
Timer_BX:=F_READDEDEBUGTIMER();
```



### 5.12.2.2 System task information

```
VAR_GLOBAL
    SystemTaskInfo : SYSTEMTASKINFOTYPE;
END_VAR
```

System flags are implicitly declared variables. Using the Input Assistant, a variable SystemTaskInfoArr can be found under system variables. This variable is a field with four structures of type SYSTEMTASKINFOTYPE [▶ 73]. The structure definition can be found in the system library. The index in this field is the task ID.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

### 5.12.2.3 System Task Info Type

```
TYPE SYSTEMTASKINFOTYPE
STRUCT
    active          :      BOOL;
    taskName        :      STRING(16);
    firstCycle      :      BOOL;
    cycleTimeExceeded :    BOOL;
    cycleTime       :      UDINT;
    lastExecTime    :      UDINT;
    priority        :      BYTE;
    cycleCount      :      UDINT;
END_STRUCT
END_TYPE
```

#### Key

- active: This variable indicates whether the task is active.
- taskName: the task name.
- firstCycle: During the first PLC task cycle, this variable has the value: TRUE.
- cycleTimeExceeded: this variable indicates whether the set task cycle time was exceeded.
- cycleTime: set task cycle time in multiples of 100 ns.
- lastExecTime: cycle time required for the last cycle in multiples of 100 ns.
- priority: set task priority.
- cycleCount: cycle counter.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

### 5.12.2.4 System info

```
VAR_GLOBAL
    SystemInfo : SYSTEMINFOTYPE;
END_VAR
```

System flags are implicitly declared variables. Using the Input Assistant, a variable Systeminfo can be found under system variables. The type SYSTEMINFOTYPE [▶ 74] is declared in the system library. For accessing the variable, the system library has to be integrated in the project.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

### 5.12.2.5 System information type

```

TYPE SYSTEMINFOTYPE
STRUCT
    runTimeNo      :      BYTE;
    projectName    :      STRING(32);
    numberOfTasks  :      BYTE;
    onlineChangeCount :      UINT;
    bootDataFlags  :      BYTE;
    systemStateFlags :      WORD;
END_STRUCT
END_TYPE
    
```

**Key**

runTimeNo: indicates the number of the runtime system (1).  
 projectName: project name as STRING.  
 numberOfTasks: number of tasks contained in the runtime system (max. 1).  
 onlineChangeCount: number of online changes since the last complete download.  
 bootDataFlags: Reserved  
 systemStateFlags: Reserved.

Development environment	Target platform	PLC libraries to be linked
TwinCAT v2.9.0	BCxx50, BC9x20 Controller	TcBaseBCxx50.lbx

### 5.12.2.6 ADS

#### 5.12.2.6.1 Local ADS Port Numbers

Port number	Description
100 [▶ 74] dec	Reading and writing of registers and tables from the coupler and the complex Bus Terminals
801 [▶ 74] dec	Local PLC process image

#### 5.12.2.6.2 ADS services

**Local process image task 1 port 800 or 801**

Data can be read from and written to the local process image. If it is necessary for outputs to be written, it is important to ensure that they are not used by the local PLC, because the local controller will overwrite these values. The data are not associated with a watchdog, and therefore must not be used for outputs that would have to be switched off in the event of a fault.

Index group	Meaning	Index offset (value range)
0xF020	Inputs	0...2047
0xF030	Outputs	0...2047
0x4020	Flags	0...4095

**ADS services**

**AdsServerAdsState**

Data type (read only)	Meaning
String	Start - the local PLC is running Start - the local PLC is stopped

**AdsServerDeviceState**

Data type (read only)	Meaning
INT	0: Start - the local PLC is running 1: Start - the local PLC is stopped

**AdsServerType**

Data type (read only)	Meaning
String	BX PLC Server

**Register Port 100**

On the Bus Terminal Controllers of the BX series, and on the BCxx50, the ADS port number for register communication is fixed at 100.

Index Group	Index offset (value range)		Meaning
	Hi-Word	Lo-Word	
0 [READ ONLY]	0...127	0..255	Registers in the Bus Coupler Hi-Word, table number of the Bus Coupler Lo-Word, register number of the table
1...255	0...3	1-255	Register of the Bus Terminals Hi-Word, channel number Lo-Word, register number of the Bus Terminal



**Minimum timeout**

When reading the register, the time out of the ADS block has to be set to a time longer than 1 second.



**Setting the password**

When writing to the registers, the password has to be set (see the documentation for the particular Bus Terminal).

## 5.13 Program transfer

### 5.13.1 Program transfer via the serial interface

Every Bus Terminal Controller can be programmed via the PC's RS232 interface.

Select the serial interface in TwinCAT PLC Control.

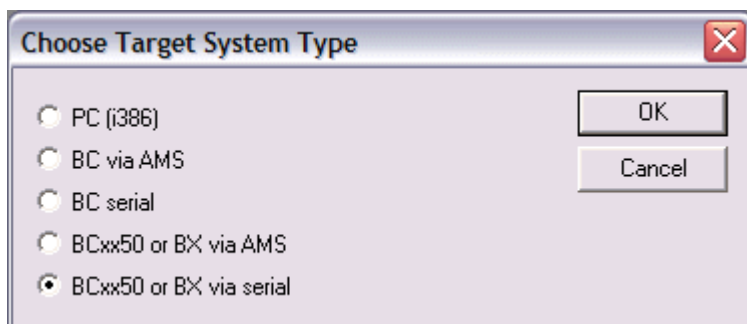


Fig. 68: Selecting the data transfer route - serial interface

The settings for the serial interface, port number, baud rate etc. are found under Online/Communication parameters in PLC Control.

The Bus Terminal Controller requires the following setting:

- Baud Rate: 9600/19200/38400/57600 baud (automatic baud rate detection)
- Stop bits: 1
- Parity: Straight line

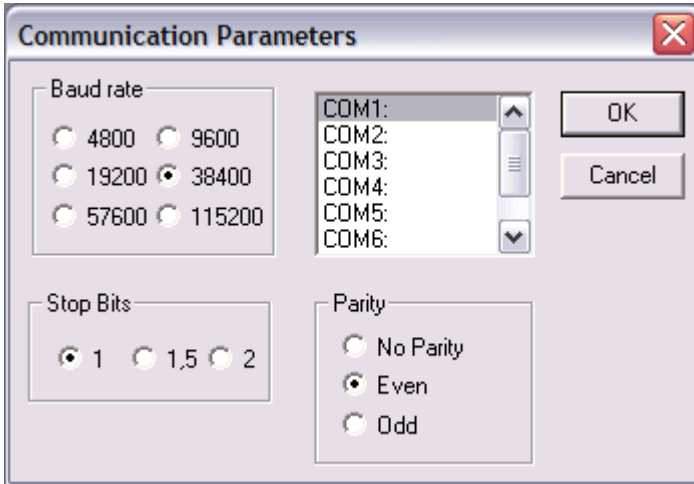


Fig. 69: Parameterization of the serial interface

**Program transfer via the serial interface and ADS**

The Bus Terminal Controller can be programmed via the PC's RS232 interface. Before you can work with the Bus Terminal Controller, TwinCAT must be notified of it (see serial ADS [▶ 37]).

Select the ADS connection in TwinCAT PLC Control.

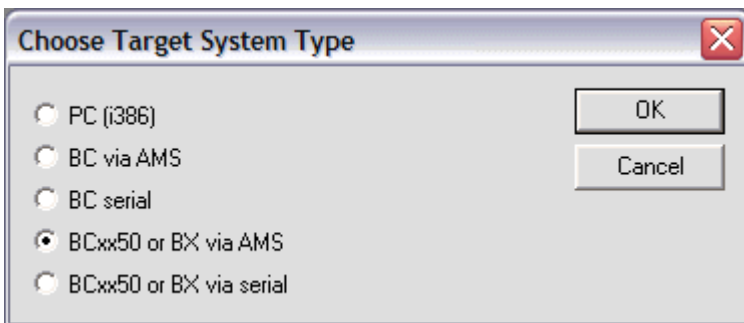


Fig. 70: Selecting the data transfer route - AMS

PLC Control can be accessed via *Online/Communication Parameters...*

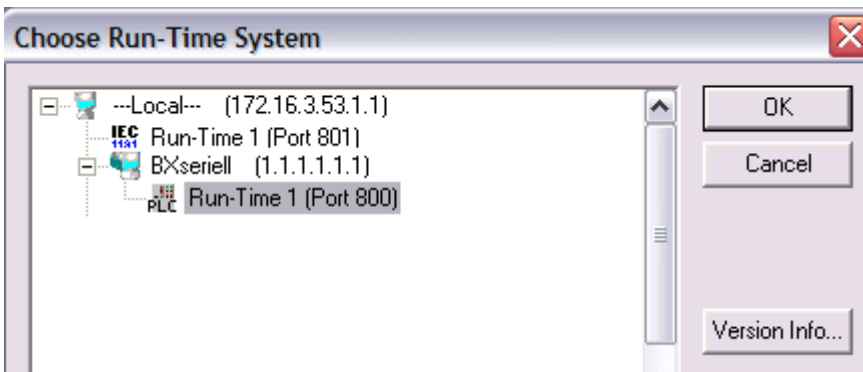


Fig. 71: Selecting the device

## 5.13.2 Program transfer via PROFIBUS

TwinCAT offers a facility for transferring the user program to the Bus Terminal Controller over the fieldbus. The BC/BX can be selected as the target system in PLC Control, after saving in the registry and restarting the TwinCAT system. The TwinCAT-level TwinCAT PLC is necessary.

Minimum requirements:

- FC310x with firmware from 2.55
- TwinCAT 2.9 Build 1025

### Initializing the Bus Terminal Controller

The coupler must first be made known to the system before it can be selected in PLC Control. Enter the Bus Terminal Controller in the System Manager, specify type, quantity and size of the fieldbus variables and link them with a task. For the subsequent program download via the PROFIBUS, the ADS interface must be enabled for the Bus Terminal Controller in the ADS tab. Save your settings and activate the configuration. Then start the TwinCAT system and the cyclic task.

### TwinCAT System Manager

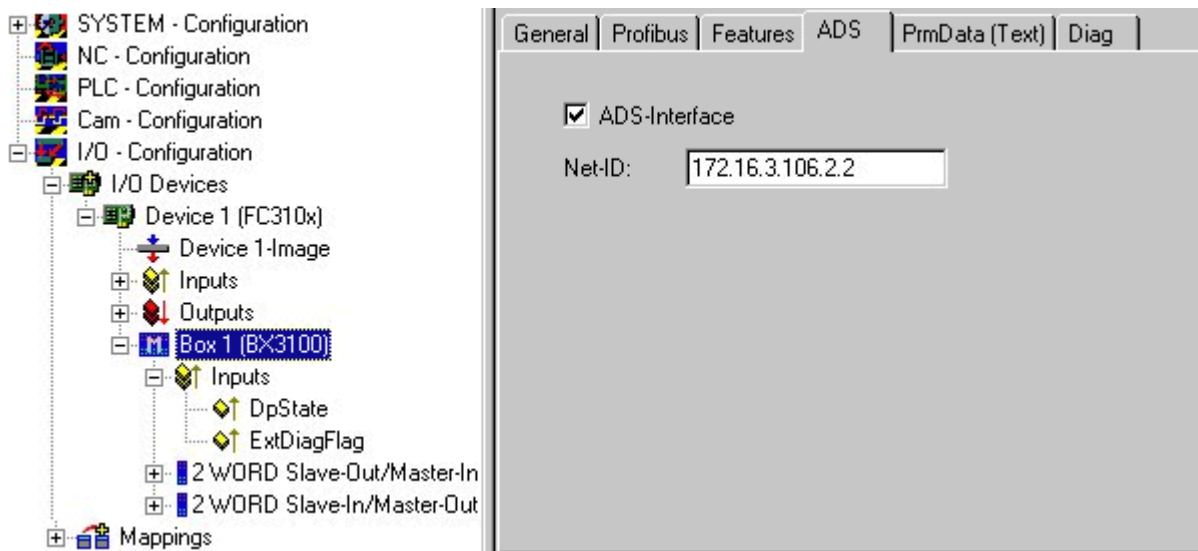


Fig. 72: ADS Interface

### PLC Control

When TwinCAT PLC Control is restarted, TwinCAT asks for the target platform, i.e. the device on which the user program is later to run. TwinCAT offers two target platforms as controller, the PC or the Bus Terminal Controller.

Two options are available to you for transmission to the Bus Terminal Controller:

- AMS for BCxx00 (Bus Terminal Controller without online change, one task)
- AMS for BCxx50 and BX (Bus Terminal Controller with online change, two tasks)
- BC serial – the serial cable for communication via the RS232 interface [▶ 75] of the PC and the programming interface of the Bus Terminal Controller

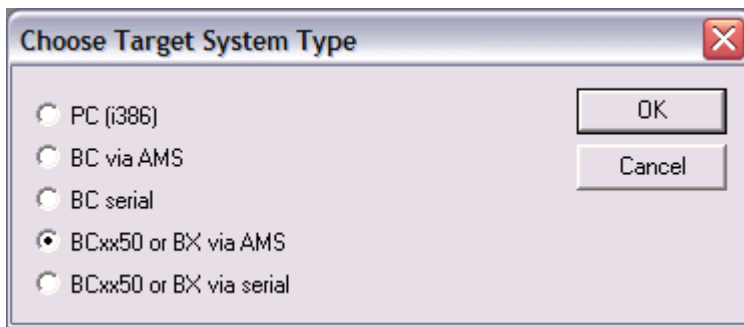


Fig. 73: Selection of the target platform

After your program has been created, select the target system under the *Online* toolbar. TwinCAT must be running to do this. In the sample, this is the Ethernet card with Box 1 and the Run-Time 1 of the Bus Terminal Controller.

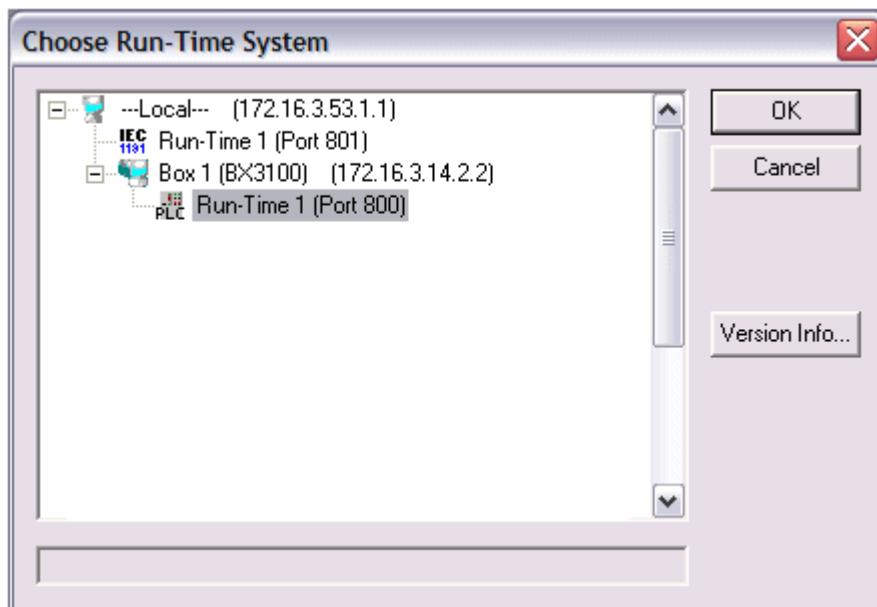


Fig. 74: selection of the target system

## 5.14 PROFIBUS process image

The BC3150 is a PROFIBUS slave device. A basic device file (GSD) is available for the configuration on the master side, which has to be integrated in the respective master configuration software. The type and quantity of the data to be transferred to the PROFIBUS master is specified on the slave side, i.e. in the BC3150. The BC3150 can process a maximum of 128 bytes of input data and 128 bytes of output data. Two configuration types are available.

No special configuration is required for the DEFAULT CONFIG. The PROFIBUS data start at a certain address offset. The type and quantity of data is configured in the master.

The TwinCAT System Manager is required for the TwinCAT CONFIG. It can be used for byte-wise linking of PROFIBUS data with PLC data, or for linking data from the K-Bus to the PROFIBUS directly.

The input data of the BC3150 are output data of the master, and output data of the BC3150 are input data of the slave.

### DEFAULT CONFIG

In DEFAULT CONFIG, all allocated variables with address 1000 or greater are transferred to the master. The type and number of variables is configured on the master side.

**Example**

Var\_1 AT %IB1000 :INT; a 2-byte output variable has to be configured on the master side.

Var\_2 AT %IB1002 :DINT; a 4-byte output variable has to be configured on the master side.

The order of the variables on the master side must be adhered to. Address gaps are only permitted if they are configured on the master side. Otherwise, addressing must be continuous from address 1000.

**TWINCAT CONFIG**

For the TwinCAT CONFIG, a configuration must be created via the System Manager. The System Manager is used for specifying the type and number of variables and for linking the variables. The PROFIBUS data may therefore be located anywhere in the permissible allocated process image.

Permissible process image:

INPUT %IB0 - %IB2047

OUTPUT %QB0 - %QB2047

## 6 PROFIBUS

### 6.1 System introduction

#### 6.1.1 Fieldbus Overview

PROFIBUS is a manufacturer-independent, open fieldbus standard with a wide range of applications in manufacturing and process automation. Manufacturer-independence and openness are guaranteed by the international standards EN 50170 and EN 50254. PROFIBUS allows devices from different manufacturers to communicate without the need for specially adapted interfaces. PROFIBUS is suitable both for fast, time-critical applications and for complex communication tasks.

PROFIBUS offers communication protocols of different functional levels: DP and FMS. According to the application, RS-485, IEC 1158-2 or optical fiber are available as transmission techniques.

PROFIBUS specifies the technical features of a serial fieldbus system with which distributed digital automation devices can be networked together, from the field level up to the cell level. PROFIBUS is a multi-master system, and therefore permits common operation of a number of automation, engineering or visualization systems with their decentralized peripheral devices on one bus.

#### PROFIBUS device types

PROFIBUS distinguishes the following device types:

**Master devices** determine the data traffic on the bus. A master may transmit messages without having received an external request when it is in possession of the bus access authorization (token). Masters are also referred to as active devices.

**Slave devices** are peripheral devices such as input/output devices, valves, drives, measuring transducers and the Beckhoff PROFIBUS slaves from the BK3xx0, BC3xx0, IPxxx-B310, IL230x-B310 and IL230x-C310 series. They do not receive any bus access authorization, so that they are only allowed to acknowledge messages that have been received, or to send messages in response to a request from master. Slaves are referred to as passive devices. They only require a small proportion of the bus protocol, which means that they can be implemented with little effort.

#### PROFIBUS DP

PROFIBUS DP is designed for efficient data exchange at the field level. The central automation devices, e.g. PLC/PC or process control systems, communicate with decentralized field devices such as I/O, drives, valves etc. via a fast serial link. The data exchange with the decentralized devices is predominantly cyclic. The communication functions required for this are specified by the basic DP functions conforms to EN 50170.

In addition to these basic functions, PROFIBUS DP also offers extended acyclic communication services for such purposes as parameterization and other operations. These are also supported by the Beckhoff PROFIBUS slaves of the IPxxx-B310, IL230x-B310 and IL230x-C310 series. A central controller (master) cyclically reads the input information from the slaves, and writes the output information cyclically to the slaves. The bus cycle time here should be shorter than the central automation system's program cycle time, which lies around 10 ms in many applications.

A high data throughput is not in itself sufficient for successful use of a bus system. Ease of handling, good diagnostic facilities and secure transmission technology are also of the utmost importance if the user's demands are to be satisfied. These properties are ideally combined in PROFIBUS DP.

#### System configuration and device types

PROFIBUS DP allows single master or multi-master systems to be implemented. This permits a high level of flexibility in system configuration. A maximum of 126 devices (master or slaves) can be connected to one bus. A station address between 0 and 99 can be chosen for the Beckhoff PROFIBUS slaves from the IPxxx-B310, IL230x-B310 and IL230x-C310 series. The specifications for the system configuration



contain the number of stations, the assignment of the station addresses to the I/O addresses, data consistency of the I/O data and the format of the diagnostic messages and the bus parameters being used. Every PROFIBUS DP system consists of different device types. Three types of device are distinguished:

Class	Description
<b>DP master class 1 (DPM1)</b> e.g. Beckhoff FC310x: PROFIBUS master card for PCs	This involves a central controller that exchanges information cyclically with the decentralized stations (slaves) in a specified message cycle. Typical devices include, for instance, programmable logic controllers (PLCs) or PCs.
<b>DP master class 2 (DPM2)</b>	Devices of this type are engineering, project design or operating devices. They are used for commissioning, for servicing and diagnosis in order to configure the connected devices, to evaluate measured values and parameters and to interrogate the status of devices.
<b>DP slave</b> e.g. Beckhoff IPxxx-B310: Fieldbus Box for PROFIBUS	A PROFIBUS DP slave is a peripheral device (I/O, drive, measuring transducer, etc.) that reads input information and passes output information on to the peripherals. It is also possible to have devices that only handle either input or output information. The quantity of input and output information is device-dependent, and may not exceed 246 bytes of input data and 246 bytes of output data.

**Mono master systems**

In single master systems only one master is active on the bus in the operating phase of the bus system. The PLC controller is the central control component. The decentralized slaves are coupled to the PLC controller via the transmission medium. The shortest bus cycle time is achieved with this system configuration.

**Multi-Master Operation**

In a multi-master mode there is more than one master on the bus. They either form subsystems that are independent of one another, each consisting of one DPM1 and the associated slaves, or additional project design and diagnostic devices. All the DP masters can read the input and output images of the slaves. Writing the outputs is only possible for one DP master (the one assigned as DPM1 during the project planning). Multi-master systems achieve a medium bus cycle time. In time-critical applications, the increase in bus cycle time should be observed by adding a diagnostic tool.

**Basic device files (GSD)**

In PROFIBUS DP, the performance characteristics of the devices are documented by the manufacturers and made available to users in the form of a device data sheet and of a basic device file. The structure, content and coding of these basic device files (GSD) is standardized. They make it easy to plan a project with any PROFIBUS DP slaves using project planning devices from various manufacturers. The PROFIBUS User Organization (Profibus Nutzer Organisation - PNO) archives this information for all manufacturers, and will provide information about the GSD from any manufacturer on request. The GSD files are read by a PROFIBUS master configuration software, and appropriate adjustments are transferred to the PROFIBUS master. Please see the appropriate software manual from the master manufacturer for a description.

The Beckhoff GSD files may be obtained from the internet under <http://www.beckhoff.de>.

**Diagnostic functions**

The extensive diagnostic functions of PROFIBUS DP allow rapid fault localization. Diagnosis of the Beckhoff Bus Coupler is not switched on in the default setting of the type file or the GSD file. The diagnostic messages are transmitted over the bus and collated by the master.

They are divided into three levels:

Diagnosis type	Description
<b>Station-related</b>	Messages relating to the general operational readiness of a device such as overtemperature or undervoltage
<b>Module-related</b>	These messages indicate that diagnostic signals are pending within a specific I/O sub range of the device (e.g. an 8 bit output module)
<b>Channel-related</b>	Here the cause of an error is related to a single input/output bit (channel), such as a short circuit on output 2

The Beckhoff PROFIBUS slaves from the IPxxxx-B310, IL230x-B310 and IL230x-C310 series support the PROFIBUS DP diagnostic functions. Assessment of the diagnostic data by means of the controller depends on the support for the PROFIBUS master. Please refer to the device manuals for the master interfaces for details of how to handle the diagnosis.

### Sync and Freeze Mode

In addition to the user data traffic related to the device, which is automatically dealt with by DPM1, a DP master has the option of sending control commands to one DP slave, to a group of them or to all of them at the same time. These control commands are transmitted as Multicasts. These control commands can be used to specify the sync and freeze operating modes, in order to synchronize the DP slave. They permit event-controlled synchronization of the DP slaves.

The DP slaves start **Sync mode** when they receive a sync control command from the assigned DP master. In this operating state, the outputs of all the addressed DP slaves are frozen at their current values. In the following user data transmissions, the DP slaves store the output data, but the output states themselves nevertheless remain unchanged. Only when the next sync control command is received from the master the stored output data is switched through to the outputs. Sync operation is ended with an Unsync control command.

A freeze control command similarly causes the addressed DP slaves to enter **Freeze mode**. In this operation mode the states of the inputs are frozen at their current value. The input data is only updated again when the DP master has sent the next freeze control command to the devices concerned. Freeze operation is ended with an unfreeze command.

### System behavior

The system behavior is also standardized in PROFIBUS DP, so that devices can to a large extent be interchanged. It is largely determined by the operating state of the DPM1. This can either be controlled locally, or over the bus by the project design device.

The following three major states are distinguished:

Operation mode	Description
<b>Stop</b>	There is no data traffic between the DPM1 and the DP slaves. The Bus Coupler only addresses the Bus Terminals once after the power has been switched on (none of the I/O LEDs are lit).
<b>Clear</b>	The DPM1 reads the input information from the DP slaves, and maintains the outputs of the DP slaves in a safe state (depending on the reaction to fieldbus errors, the green I/O LED is lit and the outputs are set).
<b>Operate</b>	The DPM1 is in a data transfer phase. In the course of cyclic data traffic the inputs of the DP slaves are read and the output information is transmitted to the DP slaves (the green I/O LED is lit).

The DPM1 sends its local status at a configurable time interval using a multicast command cyclically to all the DP slaves that have been assigned to it. The reaction that the system has to the occurrence of an error during the DPM1's data transfer phase, such as the failure of a DP slave, is specified in the *Auto-Clear* operating parameter. If this parameter is set to *True*, then the DPM1 switches the outputs of all the associated DP slaves into a safe state as soon as one DP slave is no longer ready for the transfer of user data. The DPM1 then switches into the Clear state. If the parameter is *False* then the DPM1 remains in the operating state even after a fault, and the user can himself specify the system's reaction.

**Data traffic between the DPM1 and the DP slaves**

The data traffic between the DPM1 and the DP slaves that have been assigned to it is automatically executed by the DPM1 in a specified, continuously repeated sequence. The user specifies the assignment of a DP slave to the DPM1 when the bus system's project is being planned. Those DP slaves that are included in or excluded from the cyclic user data traffic are also defined.

The data traffic between the DPM1 and the DP slaves is divided into the parameterization, configuration and data transfer phases.

Before a DP slave is included in the data transfer phase, the DPM1 checks, in the parameterization and configuration phase, whether the theoretical configuration that has been planned agrees with the actual configuration of devices. The check requires the device type, the format and length information, as well as the number of inputs and outputs, to be in agreement. The user is thus provided with reliable protection against errors in parameterization. In addition to the transfer of user data, which is automatically carried out by the DPM1, it is possible to send new parameterization data to the DP slaves at the user's request.

**Protection mechanisms**

In the context of decentralized peripherals it is necessary, for reasons of safety and reliability, for the system to be given extremely effective functions to protect against incorrect parameterization or the failure of the transmission mechanisms. PROFIBUS DP uses monitoring mechanisms in the DP Master and in the DP Slaves. They are implemented in the form of time monitors. The monitoring interval is specified in when the DP system project is planned.

Protection mechanisms	Description
<b>At the DP Master</b>	The DPM1 monitors the slave's transfer of user data with the <code>Data_Control_Timer</code> . An individual monitoring timer is used for each assigned slave. The time monitor triggers if a proper transfer of user data does not take place within the monitoring interval. In this case the user is informed. If automatic error reaction is enabled ( <code>Auto_Clear = True</code> ) then the DPM1 leaves the <i>Operate</i> state, switches the outputs of the assigned slaves into a safe state, and then goes into the <i>Clear</i> operating state.
<b>At the DP Slave</b>	The slave uses communication monitoring in order to detect errors of the master or in the transmission link. If data is not transferred with the assigned master within the communication monitoring interval the slave switches the outputs into the safe state itself. The slave inputs and outputs further require access protection in multi-master systems, to ensure that direct access is only made from the authorized master. The slaves will make an image of the inputs and outputs available to other masters, and this can be read by any other master even if it does not have access authorization.

**Ident number**

Every DP slave and every DPM1 must have an individual identification number. This is required so that a DP master can identify the types of the connected devices without any significant protocol overhead. The master compares the identification numbers of the connected DP devices with the identification numbers in the project planning data specified by DPM2. The transfer of user data only starts if the correct device types are connected to the bus at the correct station addresses. This provides protection from project planning errors. Manufacturer-specific identification numbers are issued by the PROFIBUS User Organization (PNO). The PNO administers the identification numbers along with the basic device data (GSD).

**6.1.2 PROFIBUS DP**

In PROFIBUS DP systems a master (PLC, PC, etc.) usually communicates with many slaves (I/Os, drives, etc.); only the master actively accesses the bus (by sending unsolicited telegrams), while a DP slave only sends telegrams when requested by the master.

## DP StartUp

Before the master and slave can cyclically exchange data, the parameter and configuration data is transmitted from the master to the slaves during the DP StartUp phase. After the parameter and configuration data has been sent, the master interrogates the slave's diagnostic data until the slave indicates that it is ready for data exchange. Depending on the extent of the calculations that the slave must carry out after receiving the parameter and configuration data, it can take up to a few seconds before it is ready for data exchange. For this reason the slave possesses the following states:

### Parameter data

The parameter data is sent from the master to the slave in the SetPrmLock request telegram. The SetPrmLock response telegram does not contain any data, and therefore consists of a single byte, the short acknowledgement. The parameter data consists of DP parameters (e.g. the setting of the DP watchdog or checking the IdentNumber (unique to each DP device)), of DPV1-/DPV2 parameters and of application-specific parameters that only have to be transmitted once during the StartUp. If an error is found in the parameter data, this is indicated in the diagnostic data, and the slave either remains in or enters the WAIT-PRM state.

### Configuration data

The configuration data is sent from the master to the slave in the ChkCfg request telegram. The ChkCfg response telegram does not contain any data, and therefore consists of a single byte, the short acknowledgement. The configuration data describes the assignment of the DP modules to the cyclic I/O data that is to be exchanged between the master and slave via the Data\_Exchange telegram in the cyclic data exchange phase. The sequence of the DP modules added to a slave in the DP configuration tool determines the sequence of the associated I/O data in the Data\_Exchange telegram.

### Diagnostic data

The diagnostic data is requested by the master using a SlaveDiag request telegram without any data. The slave replies with the diagnostic data in a SlaveDiag response telegram. The diagnostic data consists of the standard DP diagnostics (e.g. the state of the slave, the IdentNumber) and of application-specific diagnostic data.

### Cyclic data exchange

At the core of the PROFIBUS DP protocol is the cyclic data exchange, during which the master exchanges I/O data with each slave within a PROFIBUS DP cycle. This involves the master sending the outputs to each slave with a DataExchange request telegram, while the slave replies with the inputs in a DataExchange response telegram. This means that all the output and/or input data is transmitted in one telegram, in which the DP configuration (the sequence of DP modules) specifies the assignment of the output and/or input data to the slave's actual process data.

### Diagnosis during cyclic data exchange

A slave can send a diagnostics signal to the master during cyclic data exchange. In this case, the slave sets a flag in the DataExchange response telegram, whereby the master recognizes that there is new diagnostic data in the slave. It then fetches that data in the SlaveDiag telegram. The diagnostic data is therefore not available in real-time as the cyclic I/O data in the controller, but always delayed by at least one DP cycle.

### Synchronization with Sync and Freeze

The Sync and Freeze commands in the GlobalControl request telegram (broadcast telegram) allow the master to synchronize the activation of the outputs (Sync) or the reading of the inputs (Freeze) in a number of slaves. When the Sync command is used, the slaves are first switched into Sync mode (a process that is acknowledged in the diagnostic data). The I/O data is then exchanged sequentially with the slaves in the DataExchange telegram. Transmitting the Sync command in the GlobalControl telegram then has the effect of causing the slaves to generate the most recently received outputs. In Freeze operation a Freeze command is first sent in the GlobalControl telegram, in response to which all the slaves latch their inputs. These are then fetched sequentially by the master in the DataExchange telegram.

### States in the master

The master distinguishes between the CLEAR state (all outputs are set to the Fail\_Safe value) and the OPERATE state (all outputs have the process value). The Master is usually switched into the CLEAR mode when, for instance, the PLC enters STOP.

### Class 1 and Class 2 DP Masters

The Class 1 master refers to the controller that carries out cyclic I/O data exchange with the slaves, while a Class 2 master is a B&B device that generally only has read access to the slaves' I/O data.

## 6.1.3 PROFIBUS DPV1

PROFIBUS DPV1 refers primarily to the acyclic read and write telegrams, with which data sets in the slave are acyclically accessed. A distinction between a Class 1 and a Class 2 master is also made for DPV1. The difference between acyclic Class 1 (C1) and Class 2 (C2) connections is that the acyclic C1 connection is established during the DP StartUp phase of cyclic DP operation. Once the slave has reached the WAIT-CFG state it is possible for acyclic DPV1-C1 read and write telegrams to be sent from the master to the slave, whereas the C2 connection is established separately, independently of the cyclic DP connection. This is usually carried out by a second (C2) master so that, for instance, a manufacturer-specific project configuration and diagnostic tool can access the slave's data.

When two masters are used, however, it must always be borne in mind that these share bus access (a token is exchanged), so that time relationships are less favorable than in the case of a single master system.

## 6.1.4 Plug connectors, cables and switches

### The physics of the transmission

Physical aspects of the data transmission are defined in the PROFIBUS standard. See PROFIBUS layer 1 (physical layer).

The types of area where a fieldbus system can be used is largely determined by the choice of the transmission medium and the physical bus interface. In addition to the requirements for transmission security, the expense and work involved in acquiring and installing the bus cable is of crucial significance. The PROFIBUS standard therefore allows for a variety of implementations of the transmission technology while retaining a uniform bus protocol.

Cable-based transmission: This version, which accords with the American EIA RS-485 standard, was specified as a basic version for applications in production engineering, building management and drive technology. A twisted copper cable with one pair of conductors is used. Depending on the intended application area (EMC aspects should be considered) the shielding may be omitted.

### Cable-related faults

Two types of conductor are available, with differing maximum conductor lengths; see the "RS485" table. The plug connector assignment and the wiring are shown in the diagram below. Note the special requirements on the data cable for baud rates greater than 1.5 Mbaud. The correct cable is a basic requirement for correct operation of the bus system. If a "simple" 1.5 Mbaud cable is used, reflections and excessive attenuation can lead to some surprising phenomena. This could, for example be that some station is not connected, but when the neighboring station is unplugged the connection appears again. Or there may be transmission errors when a specific bit pattern is transmitted. The result of this can be that when the equipment is not operating, PROFIBUS works without faults, but that there are apparently random bus errors after start-up. Reducing the baud rate (< 93.75 kbaud) corrects this faulty behavior.

If reducing the baud rate does not correct the error, then in many cases this can indicate a wiring fault. The two data lines maybe crossed over at one or more connectors, the termination resistors may not be switched on, or they may be active at the wrong locations.

#### **i** Preassembled cable from Beckhoff

Installation is made a great deal more straightforward if preassembled cables from Beckhoff are used. Wiring errors are avoided, and commissioning is more rapidly completed. The range includes fieldbus cables, power supply cables, sensor cables and accessories such as termination resistors and T-pieces. Connectors and cables for field assembly are nevertheless also available.

### PROFIBUS connection of the Fieldbus Box modules

The M12 socket is inverse-coded and has 5 contact pins. Pin 1 transfers 5 V<sub>DC</sub>, pin 3 transfers GND for the active termination resistor. These must never be misused for other functions, as this can lead to destruction of the device. Pins 2 and 4 transfer the PROFIBUS signals. These must never be swapped over, as this will prevent communication. Pin 5 transfers the shield, which is capacitively connected to the base of the Fieldbus Box.

### PROFIBUS socket pin assignment

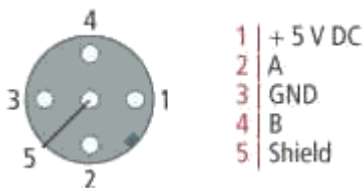


Fig. 75: Pin assignment M12 socket

**PROFIBUS wire colors**

PROFIBUS line	M12	D sub
B red	Pin 4	Pin 3
A green	Pin 2	Pin 8

**RS485 - Fundamental properties**

RS-485 transmission according to the PROFIBUS standard	
Network topology	Linear bus, active bus terminator at both ends, stubs are possible.
Medium	Screened twisted cable, shielding may be omitted, depending upon the environmental conditions (EMC).
Number of stations	32 stations in each segment with no repeater. Can be extended to 127 stations with repeater
Max. bus length without repeater	100 m at 12 Mbit/s 200 m at 1500 kbit/s, up to 1.2 km at 93.75 kbit/s
Max. bus length with repeater	Line amplifiers (repeaters) can increase the bus length to the order of 10 km. The number of repeaters possible is at least 3, and, depending on the manufacturer, may be up to 10
Data transfer rate	9.6, 19.2, 93.75, 187.5, 500, 1500 kbit/s, up to 12 Mbit/s, infinitely variable
Connector	9-pin D sub connector for IP20 M12 round connector for IP65/67

**Cabling for PROFIBUS-DP and PROFIBUS\_FMS**

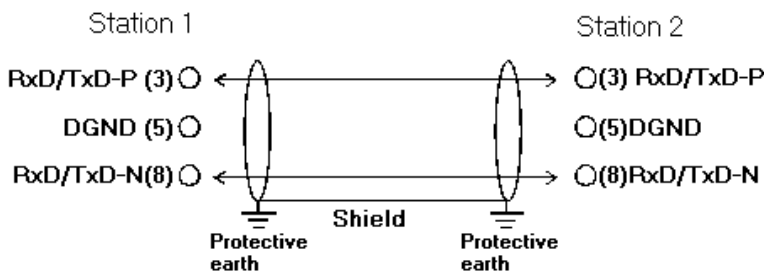


Fig. 76: PROFIBUS cable assignment

**Termination resistors**

**i** In systems with more than two stations all devices are wired in parallel. The PROFIBUS cable must be terminated with resistances at both ends, in order to avoid reflections and associated transfer problems.

**Addressing**

**Setting of station addresses**

The PROFIBUS address must be set using the two rotary selection switches behind the transparent cover. The default setting is 11. Any address is permitted, but each address may only be used once within the network. The address can be modified when the Fieldbus Box (Bus Coupler) is switched off. Release the cover (Fieldbus Box only) and set the switches to the required position using a screwdriver. Make sure that the switches engage properly. The change in address is active as soon as the device is switched on.

**Fieldbus Box address**

The switch on the left represents the tens, while that on the right represents the units.



Fig. 77: Address selection switch

### Bus Coupler address

The lower switch S311 represents the tens, while the upper switch S310 represents the unit.

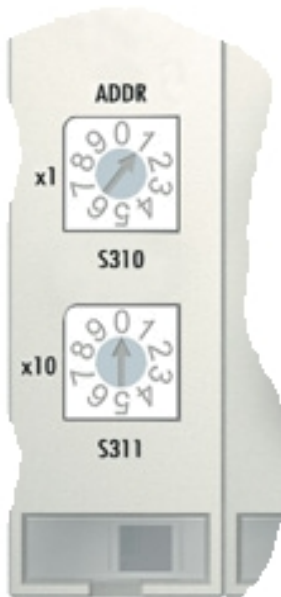


Fig. 78: Address selection switch

## 6.1.5 Topology

- A bus segment may consist of a maximum of 32 devices (including the repeaters).
- The maximum cable length of a segment depends on the data transfer rate in use and on the quality of the bus cables being used.
- No more than 9 repeaters may be installed between two devices.
- Stubs are to be avoided, and are not permitted above 1.5 Mbaud.
- The maximum number of devices is 127
- Interrupting the supply voltages from cable ends by switching off the repeater/slave, or by pulling out the plug, is not permitted.



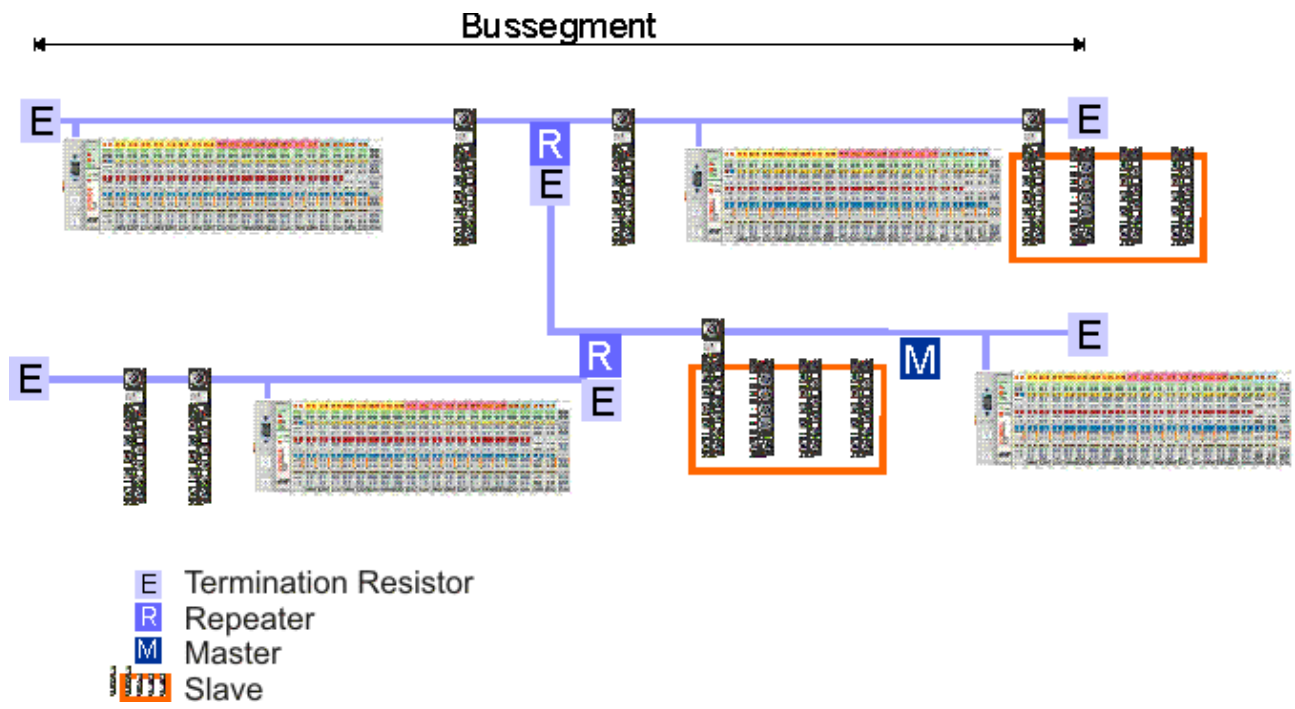


Fig. 79: RS485 topology with 3 segments and 2 repeaters

## 7 Error handling and diagnosis

### 7.1 Diagnostics

#### PROFIBUS state

In many cases it is important to know whether the communication with the higher-level master is still OK. To this end, link the *DpState* variable with your PLC program.

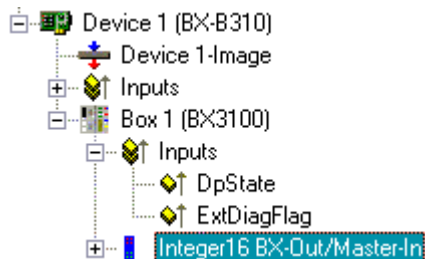


Fig. 80: PROFIBUS diagnostic byte in the System Manager

Error number	Description	Remedy
0	No error	-
129	Waiting for configuration data	Start PROFIBUS
130	Waiting for parameter data	Start PROFIBUS
131	No master available	Check PROFIBUS cables and connectors

#### Example

If the PROFIBUS is interrupted, e.g. if the cable is pulled or the PLC is switched, the Bus Terminal Controller indicates this by reporting 130 in DP state. That is, the Bus Terminal Controller is waiting for parameter data from the master.

*Slave boot sequence after a timeout or starting of the master:*  
parameter data - configuration data - data exchange

#### Reading fieldbus state by ADS

In default or TwinCAT configuration the fieldbus state can be read via ADSREAD.

Parameter ADSREAD function block	Description
NetID	local – empty string
Port	1
IndexGroup	16#0006
IndexOffset	BX3100: 16#000C_A0F4 BC3150: 16#000C_A080
LEN	1

#### State of the K-bus

An internal bus or Bus Terminal error is indicated in the K-Bus state. A more precise fault description can be obtained via a function block (in preparation). To this end, link the *K-bus state* variable with your PLC program.

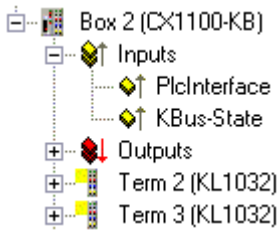


Fig. 81: Display of the K-bus status in the variable *K-bus state*

Error bit	Description	Error type
0	No error	No ERROR.
Bit 0	K-bus error	ERROR
Bit 2	K-Bus is re-triggered	NOTE

**Reading K-bus state by ADS**

In default or TwinCAT configuration the fieldbus status can be read via ADSREAD.

Parameter ADSREAD function block	Description
NetID	local – empty string
Port	1
IndexGroup	16#0006
IndexOffset	16#000C_9000
LEN	1

**7.2 Diagnostic LEDs**

The Bus Coupler features status indicator LEDs. The row of LEDs on the left describes the status of the fieldbus and of the PLC. The row of LEDs on the right indicates the supply voltage and the K-Bus state.

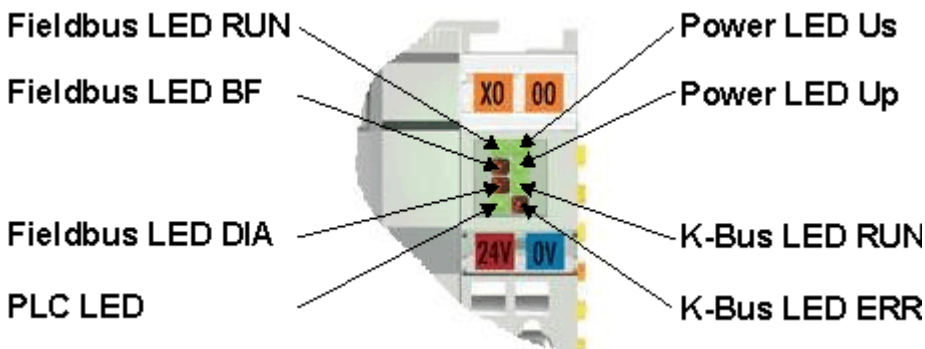


Fig. 82: LEDs

**LEDs for power supply diagnostics**

LED (Power LEDs)	Meaning
LED Us	LED off: Bus Coupler has no voltage 24 V <sub>DC</sub>
LED Up	LED off: No 24 V <sub>DC</sub> power supply connected to the power contacts

**LEDs for K-Bus diagnostics**

LED (Power LEDs)	Meaning
LED RUN	LED off: no K-Bus update, LED on, flashing: K-bus running
LED ERR	LED off: no error, LED flashing: see K-Bus error code

## K-Bus error code diagnostics

Error code	Error argument	Description	Remedy
0	-	EMC problems	<ul style="list-style-type: none"> <li>• Check power supply for undervoltage or overvoltage peaks</li> <li>• Implement EMC measures</li> <li>• If a K-bus error is present, it can be localized by a restart of the coupler (by switching it off and then on again)</li> </ul>
1	0	EEPROM checksum error	Enter factory settings with the KS2000 configuration software
	1	Code buffer overflow	Insert fewer Bus Terminals. Too many entries in the table for the programmed configuration
	2	Unknown data type	Software update required for the Bus Coupler
2	-	Reserve	-
3	0	K-bus command error	<ul style="list-style-type: none"> <li>• No Bus Terminal inserted</li> <li>• 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 until the defective Bus Terminal is located.</li> </ul>
	n	Break behind Bus Terminal n	Check whether the KL9010 Bus End Terminal is connected
4	0	K-bus data error, break behind the Bus Coupler	Check whether the n+1 Bus Terminal is correctly connected; replace if necessary.
	n	Break behind Bus Terminal n	Check whether the KL9010 Bus End Terminal is connected
5	n	K-bus error in register communication with Bus Terminal n	Exchange the nth Bus Terminal
6	0	Error at initialization	Exchange Bus Coupler
	1	Internal data error	Perform a hardware reset on the Bus Coupler (switch off and on again)
	2	DIP switch changed after a software reset	Perform a hardware reset on the Bus Coupler (switch off and on again)
7	0	Note: cycle time was exceeded	Warning: the set cycle time was exceeded. This indication (flashing LEDs) can only be cleared by booting the Bus Coupler again. Remedy: increase the cycle time
9	0	Checksum error in Flash program	Re-transfer the program to the Bus Terminal Controller
	1	Incorrect or faulty library implemented	Remove the faulty library
10	n	Bus Terminal n is not consistent with the configuration that existed when the boot project was created	Check the nth Bus Terminal. The boot project must be deleted if the insertion of an nth Bus Terminal is intentional.
14	n	nth Bus Terminal has the wrong format	Start the Bus Coupler again, and if the error occurs again then exchange the Bus Terminal
15	n	Number of Bus Terminals is no longer correct	Start the Bus Coupler again. If the error occurs again, restore the manufacturers setting using the KS2000 configuration software
16	n	Length of the K-bus data is no longer correct	Start the Bus Coupler again. If the error occurs again, restore the manufacturers setting using the KS2000 configuration software

**LED bus - fieldbus diagnostics**

LED	Meaning
LED RUN	no fieldbus connected, Bus Coupler searches for baud rate
LED BF	error flashing - error type - display
LED DIA	Bus Coupler has found baud rate, waiting for config and parameter data

LED RUN	LED BF	LED DIA	Meaning
off	off	off	no fieldbus connected, Bus Coupler searches for baud rate
on	on	on	
on	on	off	Bus Coupler has found baud rate, waiting for config and parameter data
on	off	off	no error, coupler in data exchange
on	off	flashing	error, see error code

**LED PLC - PLC diagnostics**

LED	Meaning
PLC LED	LED on: PLC running, LED off: PLC stopped

## 8 Appendix

### 8.1 First steps with the BC3150

For the following sample, the following hardware and software components are required:

#### Hardware

- FC310x from firmware 2.0
- BC3150
- KL10x4
- KL20x4
- KL9010
- PROFIBUS cable + cabling material (such as 24 V<sub>DC</sub> power supply unit etc.)

#### Software

- TwinCAT 2.9 build 1020 (minimum TwinCAT PLC level)

#### Example 1: Default Configuration

a.) Open the following link



(<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3730735499.zip>)

This link contains the System Manager file for the master configuration with the FC310x PROFIBUS card and the BC3150 slaves. Select the correct PCI address for the FC3101

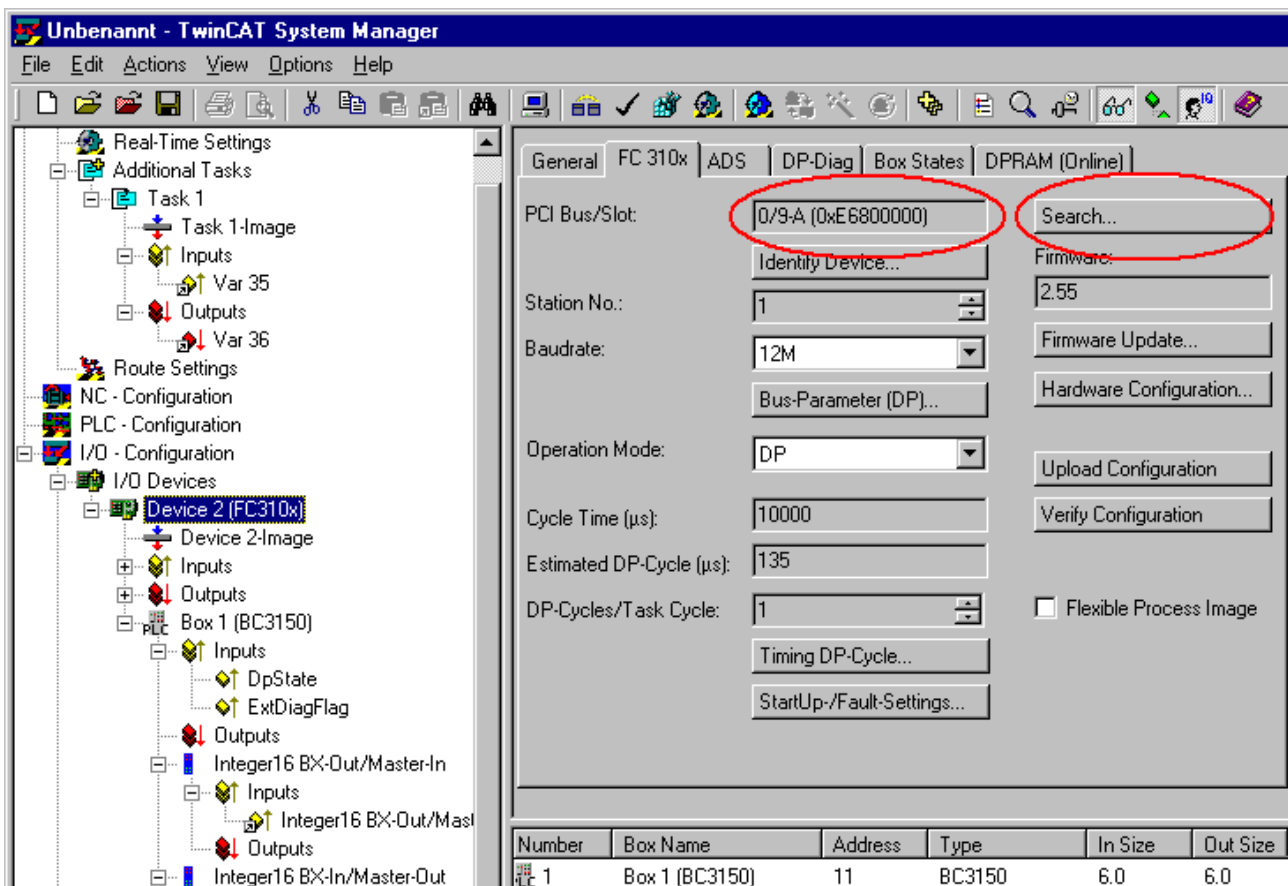


Fig. 83: Selecting the PCI address for the FC3101 PROFIBUS card

and set PROFIBUS address [► 53] 11 on the BC3150.

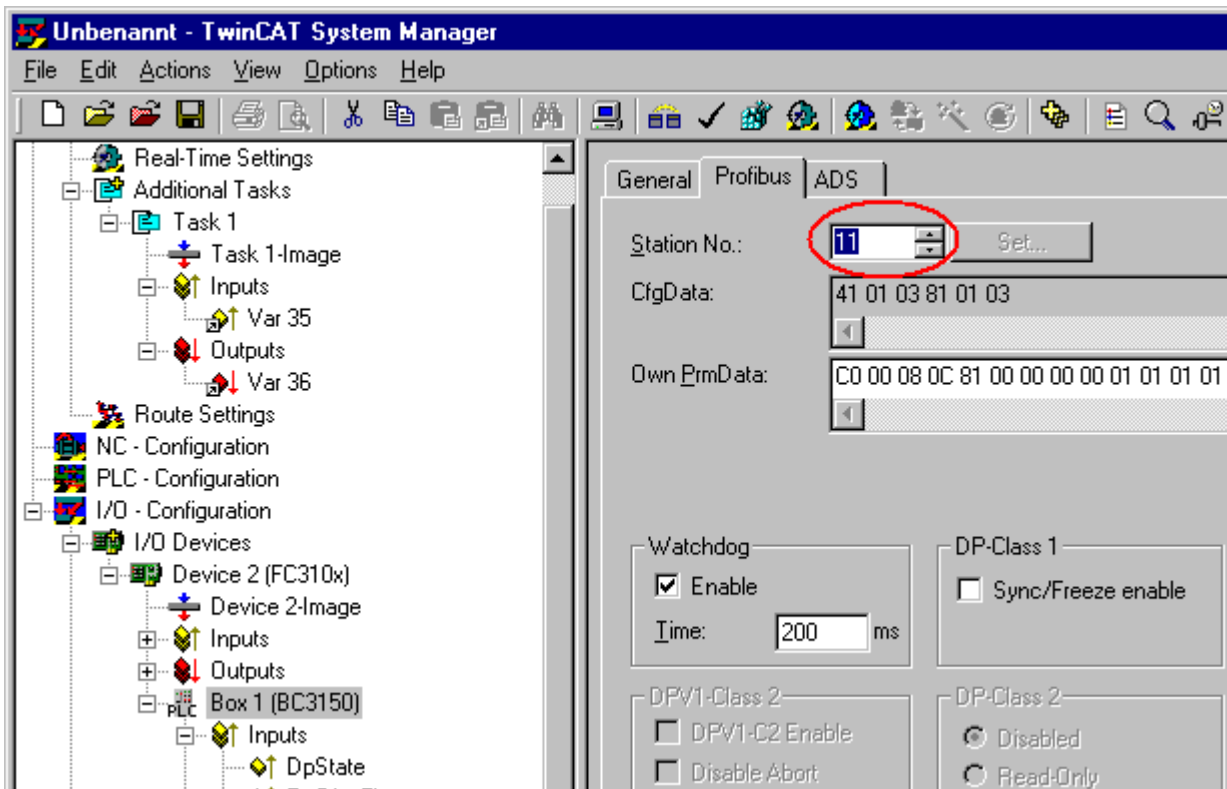


Fig. 84: Setting the PROFIBUS address

Activate the configuration and start the system. The TwinCAT icon (bottom right on the desktop) should be green. The bus LED of the BC3150 must light up green. Should this not be the case, check the BC3150 address and the physical connection to the PROFIBUS master (terminating resistors etc.).

a.) Open the following link

 (<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3730737675.zip>)

This link is the PLC project for the BC3150. Select the BC3150 under >Online\Selecting the target system<.

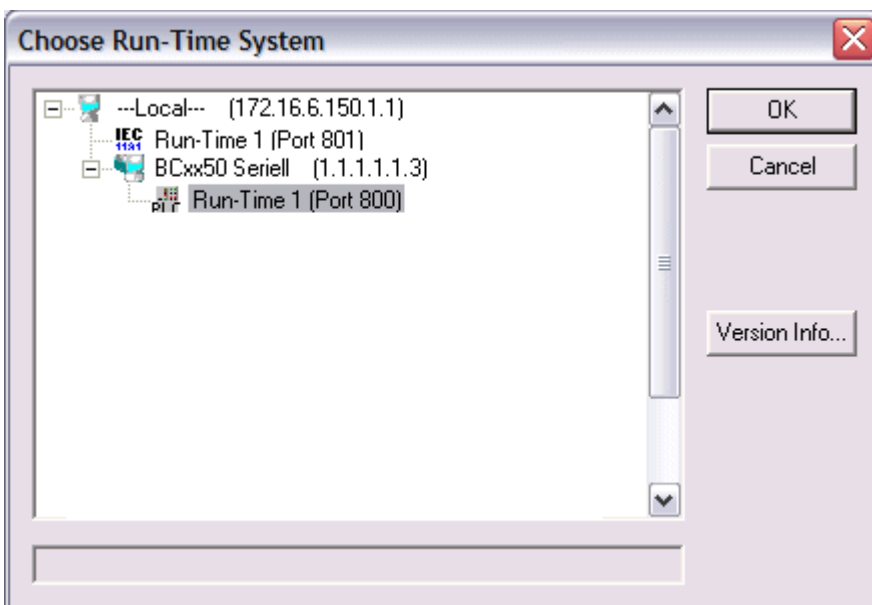


Fig. 85: Selecting the BC3150 as target system

Load the project into the controller (>Online\Login<) and Start the program (>Online\Start<). The PLC LED on the BC3150 should now be orange.

## Program description

The program increments a value >Profibus\_Out\_1< as long as the variable Profibus\_Input\_1 contains the value 0<sub>bin</sub>. At the same time, the first digital output cycles with approx. 2 Hz. In the System Manager, the variable >Var 36< can be forced to a value unequal zero. This stops the counter, and the first digital output is set to a fixed value of 1<sub>bin</sub>.

## 8.2 Switching between controllers

### Switching from BCxx00 to BCxx50/BCxx20

#### File names

In the Bus Terminal controllers of the BCxx50 and BCxx20 series, libraries have the extension \*.lbx, programs have the extension \*.prx.

#### Flag variables

The allocated flag variables

- of the BCxx00 are assigned %MB0...%MB511 (except BC9000/BC9100: %MB0...%B4095).
- of the BCxx20 are assigned %MB0...%MB4095
- of the BCxx50 are assigned %MB0...%MB4095

Status information such as K-bus/fieldbus status and cycle tick is not copied to the BCxx50/BCxx20. This information is available in TcSystemBCxx50.lbx as a function for the BCxx50/BCxx20.

The allocated flags do **not** act as retain variables.

#### Retain data

The retain data have to be declared as VAR\_RETAIN [► 60]. Up to 2 kbyte are available.

#### PLC Variables

In the Default-Config the PLC variables start from %IB1000 and %QB1000.

#### Large model

Not applicable for BCxx50 and BCxx20.

Max. memory:

- BCxx50: 48 kbyte
- BCxx20: 128 kbyte

#### Task time

The task time is specified in the PLC Control. It should be set to a realistic value (measuring of PLC cycle time and K-Bus). The background time is not used.

#### Task configuration

A maximum of one task is available. This task must be configured.

#### PLC and fieldbus terminals

For the standard Bus Terminal Controllers (BCxx00) it was possible to select whether a Bus Terminal is assigned to the fieldbus or the local PLC.



In the default configuration of the BCxx50/BCxx20 all Bus Terminals are assigned to the local PLC. An assignment to the fieldbus is not possible in this case.

### Switching from BCxx00 to BXxx00

#### File names

In the Bus Terminal controllers of the BCxx00, libraries have the extension \*.lbx, programs have the extension \*.prx.

#### Flag variables

The allocated flag variables

- of the BCxx00 are assigned %MB0...%MB511 (except BC9000/BC9100: %MB0...%B4095).
- of the BXxx00 are assigned %MB0...%MB4095

Status information such as K-bus/fieldbus status and cycle tick is not copied to the BXxx20. This information is available in TcSystemBCxx00.lbx as a function for the BXxx50.

The allocated flags do **not** act as retain variables.

#### Retain data

The retain data have to be declared as `VAR RETAIN [► 60]`. Up to 2 kbyte are available.

#### PLC Variables

In the Default-Config the PLC variables start from %IB1000 and %QB1000.

#### Large model

Not applicable for BXxx00. Max. memory: 256 kbyte.

#### Task time

The task time is specified in the PLC Control. It should be set to a realistic value (measuring of PLC cycle time and K-Bus). The background time is not used.

#### Task configuration

A maximum of one task is available. This task must be configured.

#### PLC and fieldbus terminals

For the standard Bus Terminal Controllers (BCxx00) it was possible to select whether a Bus Terminal is assigned to the fieldbus or the local PLC.

In the default configuration of the BXxx00 all Bus Terminals are assigned to the local PLC. An assignment to the fieldbus is not possible in this case.

### Switching from PC to BCxx50/BCxx20/BXxx00

#### File names

In the Bus Terminal controllers of the BCxx50/BCxx20 and BXxx00 series, libraries have the extension \*.lbx, programs have the extension \*.prx.

#### Allocated variables

For the Bus Terminal controllers of the BCxx50/BCxx20 and BXxx00 series, a limited number of allocated data are available:

- inputs 2 kbyte, %IB0...2048
- outputs 2 kbyte, %QB0...2048
- flags 4 kbyte, %MB0...4095

### Task configuration

A maximum of one task is available. A sensible task time should be selected. Adjust the task time to your application by measuring the required system time (PLC + K-Bus + fieldbus + other).

### Retain data

For the Bus Terminal controllers of the BCxx50, BCxx20 and BXxx00 series, up to 2 kbyte of retain data are available. Ensure that no (or only very few) retain data are used in function blocks (see [RETAIN data \[► 60\]](#)).

## 8.3 Firmware Update

### Firmware update program

The firmware update program is required for loading a new firmware to the Bus Coupler. The program is transferred via the serial interface.

Note for BX3100:

Updates are not available with BX3100 firmware 0.64 (or lower). If these devices need updating, send the BX3100 to the manufacturer with a corresponding note.

Beckhoff Automation GmbH & Co. KG  
Service Department  
Stahlstr. 31  
33415 Verl, Germany

 Firmware update program 241 (<https://infosys.beckhoff.com/content/1033/bc3150/Resources/zip/3238791819.zip>) 71 kbytes (for Windows NT4.0 SP6, 2000, XP).

The program *FirmwareUpdate.exe* and the file *TcRouterHelper.dll* have to be in the same directory. Open the program by double-clicking on *FirmwareUpdate.exe*.

### Update for Bus Terminal Controllers

#### BX series

Select the appropriate device of - in this example "Serial interface (BX)".

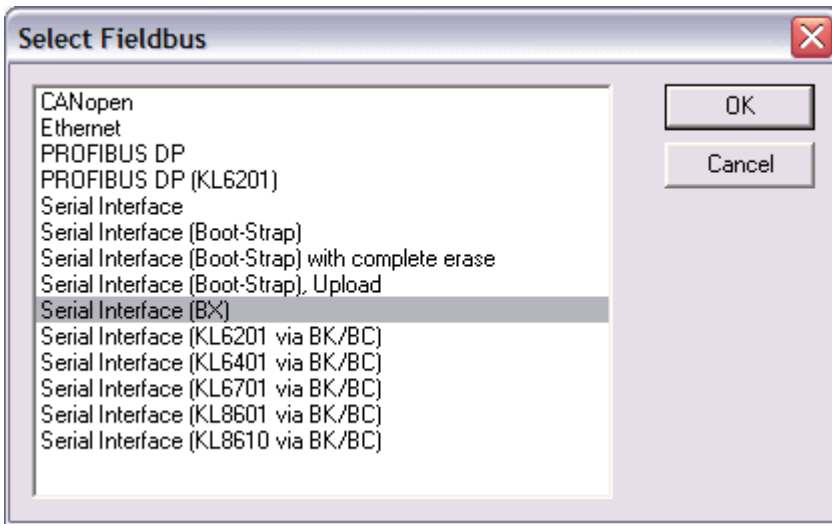


Fig. 86: Selecting a BX series Bus Terminal Controller

**BCxx50 series**

Select the corresponding device, in this case "Serial Interface".

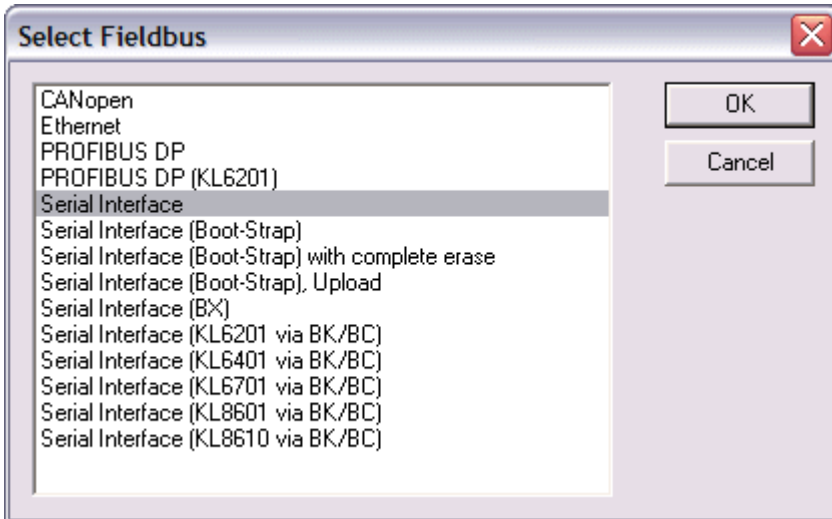


Fig. 87: Selecting a BC series Bus Terminal Controller

**BX and BCxx50 series**

Then select the COM port.

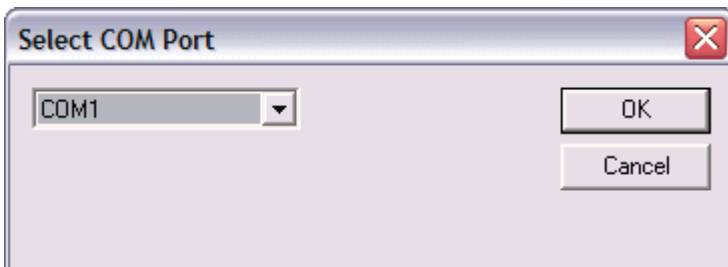


Fig. 88: Select the COM port

Open the file you wish to download.

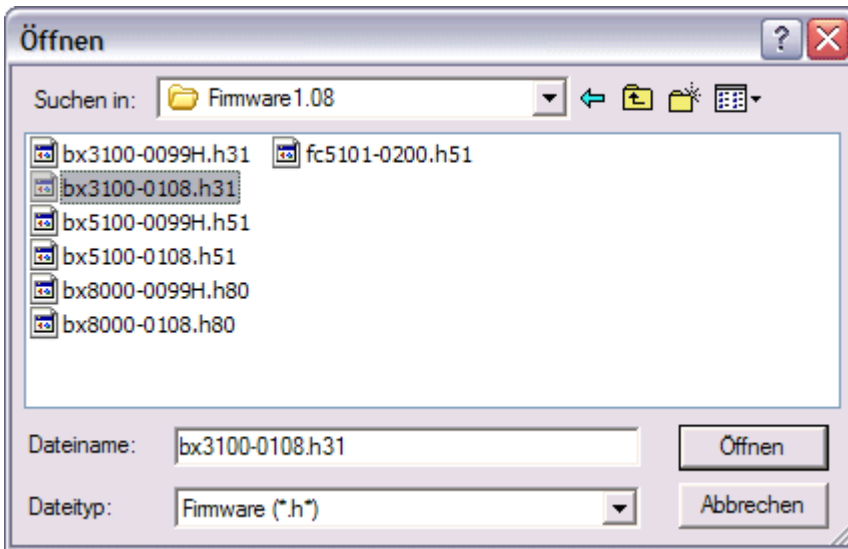


Fig. 89: Open the firmware file

Start the download via the green 'traffic light'. The download begins after about a minute, and is then also shown on the BX's display. After successful download (approx. 2 to 3 minutes) the Bus Terminal Controller reboots automatically.

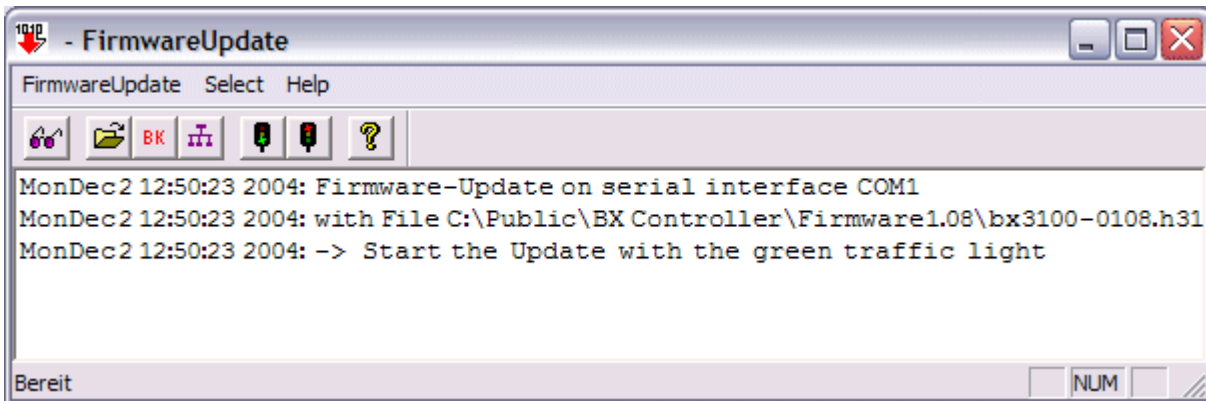


Fig. 90: Status messages relating to the firmware update

## 8.4 Test standards for device testing

### EMC

#### EMC immunity

EN 61000-6-2

#### Electromagnetic emission

EN 61000-6-4

### Vibration / shock resistance

#### Vibration resistance

EN 60068-2-6

#### Shock resistance

EN 60068-2-27

## 8.5 Bibliography

### German books

#### PROFIBUS

- PROFIBUS DP/DPV1  
Basic principles, tips and tricks for users  
by Manfred Popp  
ISBN: 3778527819

#### General fieldbus technology

- Gerhard Gruhler (Pub.): **Fieldbuses and device communication systems**  
Practical knowledge with comparison options  
Published by Franzis Verlag, 2001  
244 pages  
ISBN 3-7723-5745-8

### English books

(in preparation)

#### PROFIBUS-DP standards

- IEC 61158 and IEC 61784
- DIN 19245, Part 3
- Euronorm EN 50 170

### Web sites

- <http://www.profibus.de>

## 8.6 List of Abbreviations

### **DP**

Decentralized periphery. PROFIBUS protocol for fast cyclic data exchange

### **FMS**

PROFIBUS transfer protocol (Fieldbus Message Specification).

### **Freeze mode**

This command makes the slave freeze its inputs

### **GSD file**

German device master file

### **GSE file**

English device master file

### **IP20, IP65, IP66, IP67**

Protection class (contact, water, dust)

### **K-bus**

Terminal Bus: Internal bus for communication between Bus Coupler and Bus Terminals

### **PNO**

PROFIBUS User Organization (see [www.profibus.de](http://www.profibus.de))

### **Repeater**

Provides signal conditioning, connecting individual bus segments

### **PLC**

Programmable logic controller

### **Sync-Mode**

This command makes the slave hold its outputs unchanged until it receives the Sync telegram.

## 8.7 Support and Service

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

### Beckhoff's branch offices and representatives

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

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

You will also find further documentation for Beckhoff components there.

### Beckhoff Support

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

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

Hotline: +49 5246 963 157  
Fax: +49 5246 963 9157  
e-mail: [support@beckhoff.com](mailto:support@beckhoff.com)

### Beckhoff Service

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

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

Hotline: +49 5246 963 460  
Fax: +49 5246 963 479  
e-mail: [service@beckhoff.com](mailto:service@beckhoff.com)

### Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20  
33415 Verl  
Germany

Phone: +49 5246 963 0  
Fax: +49 5246 963 198  
e-mail: [info@beckhoff.com](mailto:info@beckhoff.com)  
web: <https://www.beckhoff.com>

## Table of figures

Fig. 1	BC3150 - Bus Terminal Controller for PROFIBUS .....	9
Fig. 2	The principle of the Bus Terminal .....	12
Fig. 3	Spring contacts of the Beckhoff I/O components.....	15
Fig. 4	BCxx50 - Dimensions .....	16
Fig. 5	Release the locking mechanism by pulling the orange tab.....	17
Fig. 6	Groove and tongue of the housings.....	17
Fig. 7	Potential groups of a Bus Terminal block .....	18
Fig. 8	Power contact on the left .....	19
Fig. 9	Terminal points for the Bus Terminal Controller supply.....	20
Fig. 10	UL identification .....	20
Fig. 11	Pin assignment M12 socket (-B310).....	22
Fig. 12	Pin assignment M12 socket/plug connector (-B318).....	22
Fig. 13	Pin assignment of the PROFIBUS D-sub socket.....	22
Fig. 14	Pin assignment socket/plug connector Fieldbus Box modules.....	23
Fig. 15	PROFIBUS cable assignment .....	24
Fig. 16	Start-up behavior of the Bus Terminal Controller .....	28
Fig. 17	Creating a TwinCAT configuration.....	31
Fig. 18	Selecting the Bus Terminal Controller .....	31
Fig. 19	Downloading a TwinCAT configuration.....	32
Fig. 20	Selecting the Bus Terminal Controller .....	33
Fig. 21	State of the Bus Terminal Controller.....	33
Fig. 22	Activating the TwinCAT configuration.....	33
Fig. 23	Choose Target System .....	34
Fig. 24	Selecting the Bus Terminal Controller .....	34
Fig. 25	State of the Bus Terminal Controller.....	34
Fig. 26	Uploading the TwinCAT configuration .....	35
Fig. 27	Memory for code mapping .....	35
Fig. 28	Data memory mapping .....	36
Fig. 29	Code and data memory .....	36
Fig. 30	Other memory .....	37
Fig. 31	Properties of the remote connection.....	38
Fig. 32	DP slaves tab.....	39
Fig. 33	Inserting BX3100 .....	41
Fig. 34	Creating PROFIBUS data.....	42
Fig. 35	ADS Interface .....	42
Fig. 36	Creating PROFIBUS data.....	43
Fig. 37	BC3150d.....	44
Fig. 38	BC3150n.....	44
Fig. 39	BC3150s .....	44
Fig. 40	Settings.....	45
Fig. 41	Sample for entering individual bytes. ....	46
Fig. 42	Sample for entering associated bytes.....	47
Fig. 43	BX Settings tab .....	48
Fig. 44	BX Diag tab.....	49



Fig. 45	Selecting the PLC project .....	50
Fig. 46	Connecting PLC variable and hardware .....	51
Fig. 47	Target system display .....	51
Fig. 48	Setting the task time .....	52
Fig. 49	Displaying the PLC cycle time .....	53
Fig. 50	Programming cable KS2000-Z2 .....	55
Fig. 51	Maximum number of POU's exceeded .....	57
Fig. 52	Menu path Projects / Options / Controller Settings .....	57
Fig. 53	Controller settings .....	58
Fig. 54	Global memory insufficient .....	58
Fig. 55	Menu path Projects / Options / Build .....	58
Fig. 56	Build .....	59
Fig. 57	Changing variable links .....	63
Fig. 58	Linking a variable with an input .....	64
Fig. 59	Opening the options menu .....	66
Fig. 60	Selecting Source Download .....	67
Fig. 61	Downloading the program code .....	68
Fig. 62	Download progress .....	68
Fig. 63	Uploading a program .....	69
Fig. 64	Selecting the data transfer route .....	69
Fig. 65	Selecting the device .....	69
Fig. 66	Function block F_STARTDEBUGTIMER .....	72
Fig. 67	Function block F_READDEBUGTIMER .....	72
Fig. 68	Selecting the data transfer route - serial interface .....	75
Fig. 69	Parameterization of the serial interface .....	76
Fig. 70	Selecting the data transfer route - AMS .....	76
Fig. 71	Selecting the device .....	76
Fig. 72	ADS Interface .....	77
Fig. 73	Selection of the target platform .....	78
Fig. 74	selection of the target system .....	78
Fig. 75	Pin assignment M12 socket .....	86
Fig. 76	PROFIBUS cable assignment .....	87
Fig. 77	Address selection switch .....	88
Fig. 78	Address selection switch .....	88
Fig. 79	RS485 topology with 3 segments and 2 repeaters .....	89
Fig. 80	PROFIBUS diagnostic byte in the System Manager .....	90
Fig. 81	Display of the K-bus status in the variable K-bus state .....	91
Fig. 82	LEDs .....	91
Fig. 83	Selecting the PCI address for the FC3101 PROFIBUS card .....	94
Fig. 84	Setting the PROFIBUS address .....	95
Fig. 85	Selecting the BC3150 as target system .....	95
Fig. 86	Selecting a BX series Bus Terminal Controller .....	99
Fig. 87	Selecting a BC series Bus Terminal Controller .....	99
Fig. 88	Select the COM port .....	99
Fig. 89	Open the firmware file .....	100
Fig. 90	Status messages relating to the firmware update .....	100



More Information:  
[www.beckhoff.en/BC3150](http://www.beckhoff.en/BC3150)

Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Germany  
Phone: +49 5246 9630  
[info@beckhoff.com](mailto:info@beckhoff.com)  
[www.beckhoff.com](http://www.beckhoff.com)

