BECKHOFF New Automation Technology

Documentation | EN

Fieldbus Box Modules for Ethernet

2009-10-29 | Version: 1.2.1

Table of contents

1 Foreword

1.1 Notes on the documentation

Intended audience

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

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

Disclaimer

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

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

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

Trademarks

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

Patent Pending

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

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

Copyright

1.2 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Description of instructions

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

⚠ DANGER

Serious risk of injury!

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

⚠ WARNING

Risk of injury!

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

⚠ CAUTION

Personal injuries!

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

NOTE

Damage to environment/equipment or data loss

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

Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3 Documentation issue status

Version	Comment
1.2.1	System overview updated
1.2	Pictures of Field Bus Box Modules with M12 Ethernet connector (B901) updated
1.1	Chapter *Check of the IP-Link connection* added
1.0	First version

1.4 Bus Coupler as a general term

Parts of this manual give general information about Ethernet implementation in Beckhoff products.
Thus in the following often the term *Bus Coupler* is used, that describes not only the IP20 products, but also means the IP67 modules.

2 Product overview

2.1 The Fieldbus Box System

Fieldbus box modules are robust fieldbus stations for a large number of different fieldbus systems. They offer a wide range of I/O functionality. All relevant industrial signals are supported. As well as digital and analog inputs and outputs including thermocouple and RTD inputs, there are also incremental encoder interfaces available for displacement and angle measurement as well as serial interfaces to solve a large number of communications tasks.

Three varieties of signal connection

The digital inputs and outputs can be connected with snap-on 8 mm diameter plugs, screw-in M8 connectors, or with screw-in M12 pendants. The M12 version is provided for analog signals.

All important signal types

Special input and output channels on the combination I/O modules can be used for either input or output. It is not necessary to configure them, since the fieldbus interface is available for every combination channel as well as for input and output data. The combination modules give the user all of the advantages of fine signal granularity.

The processor logic, the input circuitry and the power supply for the sensor are all fed from the control voltage. The load voltage for the outputs can be supplied separately. In those Fieldbus Boxes in which only inputs are available, the load power supply, UP, can optionally be connected in order to pass it on downstream.

The states of the Fieldbus Box, the fieldbus connection, the power supplies and of the signals are indicated by LEDs.

The label strips can be machine printed elsewhere, and then inserted.

Fieldbus Boxes can be combined for greater flexibility

In addition to the Compact Box, the Fieldbus Box series also includes extendable devices, namely the Coupler Box and the Extension Box, as well as intelligent devices, the PLC Boxes.

Compact Box

The Compact Box makes the I/O data from the connected digital and analog sensors and actuators available to the fieldbus.

Coupler Box

The Coupler Box also collects I/O data from the Extension Boxes via an interference-proof optical fiber connection (IP-Link). Up to 120 Extension Boxes can be connected to a Coupler Box. In this way a distributed IP67 I/O network is formed with only one fieldbus interface.

The Coupler Box is capable of automatically recognizing the extension modules connected to it during start-up, and maps the I/O data automatically into the fieldbus process image – a configuration is not necessary. The Coupler Box appears, from the fieldbus point of view, along with all of the networked Extension Boxes, as a single participating bus device with a corresponding number of I/O signals.

The Coupler Box corresponds to the Bus Coupler in the BECKHOFF Bus Terminal system. BECKHOFF fieldbus devices made to protection class IP 20 (Bus Terminals) and IP 67 (Fieldbus Box) can be combined without difficulty – the data is handled in the same way in either case.

IP-Link

The IP-Link is an optical fiber connection with a transmission rate of 2 MBits/s which is capable of transmitting 1000 items of binary I/O data in approx. 1 ms, rapidly and securely. Smaller configurations are correspondingly faster. Because of the high usable data rate, the coupling via IP-Link does not reduce the performance of the fieldbus at all.

Low-priced plug connectors made according to Protection Class IP 67 can be used for the rapid and simple preparation of the IP-Link cable, in situ. The connection does not require special tools, and can be performed quickly and simply. The IP-Link cables can also be obtained with prepared plugs if required.

The separate supply of the output voltage allows output groups to be switched off individually. Differing potentials can also be created within an extension ring without difficulty, since the IP-Link naturally has optimum electrical isolation.

Extension box

Like the Compact Boxes, the Extension Boxes cover the full spectrum of I/O signals, and may be up to 15 m apart. They are remarkably small in size, and lead to particularly economical I/O solutions with high levels of protection. Here again, the digital inputs and outputs may optionally be connected via snap-on 8 mm connectors, or via screw-in connectors (M8 and M12). Analog signal types are provided with the M12 version. The snap-on connectors lock in place positively, forming a shake-proof connection, while the screw-in connectors offer the advantage of high resistance to being pulled out.

PLC Box

The PLC Box is an intelligent Fieldbus Box with PLC functionality for distributed pre-processing of the I/O signals. This allows parts of the application to be farmed out from the central controller. This reduces the load on the CPU and the fieldbus. Distributed counting, controlling and switching are typical applications for the PLC Box. The reaction times are independent of the bus communication and of the higher-level controller.

In the event of a bus or controller failure, maintenance of function (e.g. bringing the process to a safe state in an orderly manner) is possible.

Programming is carried out with TwinCAT in accordance with IEC 61131-3. Five different programming languages are available:

- Instruction List (IL)
- Function Block Diagram (FBD)
- Ladder Diagram (LD)
- Sequential Function Chart (SFC)
- Structured Text (ST)

The program download occurs either via the fieldbus or via the programming interface.

Extensive debugging functions (breakpoint, single step, monitoring, etc) are also available. The PLC Box contains a powerful 16 bit controller, 32/96 kByte program memory and 32/64 kByte data memory. A further 512 bytes of non-volatile memory are available for remanent flags.

PLC Box with IP-Link

The programmable PLC Box with IP-Link provides almost unlimited I/O possibilities. Up to 120 extension modules, with more than 2000 I/Os, can be directly addressed from the PLC program. The PLC Box is thus also suitable for use as a small, autonomous controller for the operation of parts of equipment or small machines.

2.2 Fieldbus Box - Naming conventions

The identifications of the Fieldbus Box modules are to be understood as follows:
IXxxxy-zyyy

IX describes the design:

"IP" stands for the Compact Box design [▶ 12]
"IL" stands for the Coupler Box design (with IP-Link) [▶ 12]
"IE" stands for the Extension Box design [▶ 12]

xxxy describes the I/O connection:

xxx describes the I/O property:
"10x" - 8 x digital inputs
"15x" - counter module
"20x" - 8 x digital outputs
"25x" - PWM module
"23x" - 4 x digital inputs and 4 x digital outputs
"24x" - 8 x digital inputs and 8 x digital outputs
"3xx" - 4 x analog inputs
"4xx" - 4 x analog outputs
"5xx" - incremental encoder or SSI transducer
"6xx" - Gateway module for RS232, RS422, RS485, TTY

y represents the mechanical connection:
"0" stands for 8mm snap-on connection,
"1" stands for M8 bolted connection
"2" stands for M12 bolted connection and
"9" stands for M23 bolted connection

zyyy describes the programmability and the fieldbus system

z distinguishes whether the device is a slave or is a programmable slave:

"B" - not programmable
"C" - programmable (PLC Box)

"yyy" stands for the fieldbus system and the bus connection:
"110" - EtherCAT
"200" - Lightbus
"310" - PROFIBUS
"318" - PROFIBUS with integrated tee-connector
"400" - Interbus
"510" - CANopen
"518" - CANopen with integrated tee-connector
"520" - DeviceNet
"528" - DeviceNet with integrated tee-connector
"730" - Modbus
"800" - RS485
"810" - RS232
"900" - Ethernet TCP/IP with RJ45 for the bus connection
"901" - Ethernet TCP/IP with M12 for the bus connection
"903" - PROFINET
"905" - EtherNet/IP

Compact Box

Compact Box

The Compact Box modules offer a wide range of I/O functionality. All relevant industrial signals are supported. The digital inputs and outputs can be connected either with snap-on 8 mm diameter plugs, screw-in M8 connectors, or screw-in M12 connectors. The M12 version is made available for analog signals.

Depending on the module, the I/O section and the power supply section can differ.

Coupler Box

Coupler Box

There are three versions of the coupler box named IL230x-Bxxx. It differs from the compact box in that this module offers an interface to what are known as extension boxes. This interface is a subsidiary bus system based on the optical fiber what is known as IP Link. This powerful subsidiary bus system can handle up to 120 extension boxes at one coupler box.

Extension Box

Extension Box

Extension Modules, that are independent of the fieldbus and that can only be operated together with a coupler box via IP Link.

PLC Box

PLC Box

A PLC Box differ from the Coupler Box in that this module can be programmed in IEC 61131-3. This means that this slave is also capable of working autonomously, without a master, for instance for control or regulation tasks.

Also see about this

- Fieldbus Box - Naming conventions [▶ 12]

2.3 Firmware and hardware issue status

The documentation refers to the hardware and software status that was valid at the time it was prepared. The properties are subject to continuous development and improvement. Modules having earlier production statuses cannot have the same properties as modules with the latest status. Existing properties, however, are always retained and are not changed, so that these modules can always be replaced by new ones. The number beginning with a *D* allows you to recognize the firmware and hardware status of a module.

Syntax:

D . ww yy x y z u

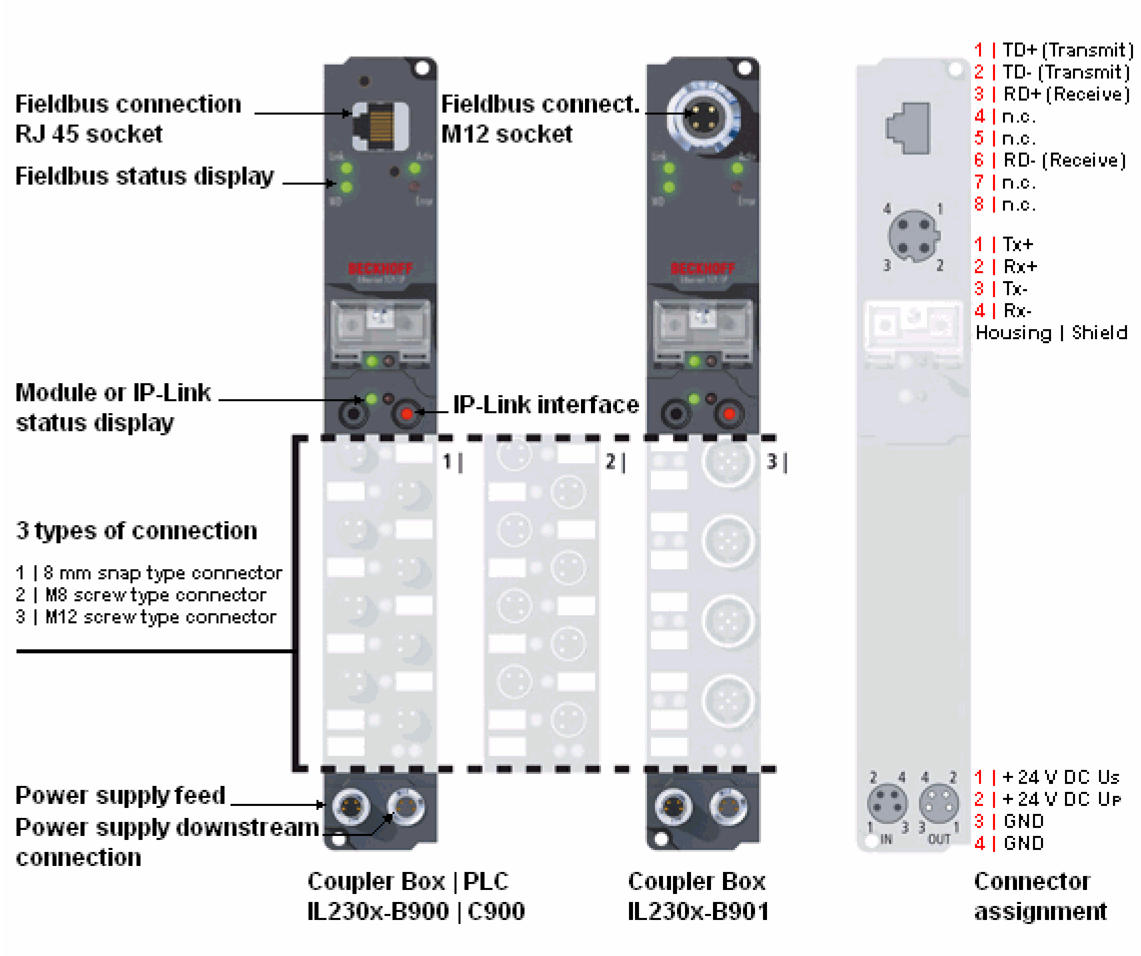
ww - calendar week
yy - year
x - bus board firmware status
y - bus board hardware status
z - I/O board firmware status
u - I/O board hardware status

Example:

D.22081501

- Calendar week 22
- in the year 2008
- bus board firmware status: 1
- bus board firmware hardware status: 5
- I/O board firmware status: 0 (no firmware is necessary for this board)
- I/O board hardware status: 1

2.4 Technical data



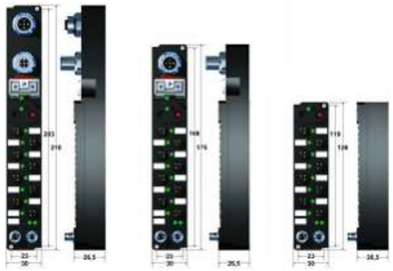
Technical data	IL230x-B900, IL230x-C900	IL230x-B901
Extension modules (IL....)	max. 120	
Digital peripheral signals (IL....)	max. 960 In- and Outputs	
Analog peripheral signals (IL....)	max. 480 In- and Outputs	
Transmission medium	4 x 2 twisted pair copper cable; category 3 (10 MBaud), category 5 (100 MBaud)	
Transfer rate	10/100 MBaud	
Topology	star shaped cabling	
Distance between modules	100 meters (Hub/Switch to Fieldbus Box)	
Configuration	via KS2000 or via controller	
Protocols	UDP- ADS, TCP-ADS, Fast-ADS, ModbusTCP, Real time Ethernet	
Power supply	Control voltage: $24V_{DC}$ (-15%/+20%); load voltage: according to I/O type	
Control voltage current consumption	according to I/O type + current consumption of sensors, max. 0.5 A	
Load voltage current consumption	according to I/O type	
Power supply connection	Feed: 1 x M8 connector 4-pin downstream connection: 1 x M8 socket 4-polig (except IP/IE204x)	
Connection Fieldbus	1 x RJ45 socket	1 x M12 socket, d-coded
Electrical isolation	Channels / control voltage: no between the channels: no control voltage / fieldbus: yes	
Permissible ambient temperature range during operation	0°C ... +55°C	
Permissible ambient temperature range during storage	-25°C ... +85°C	
Vibration / shock resistance	according to EN 60068-2-6 / EN 60068-2-27, EN 60068-2-29	
EMC resistance burst / ESD	according to EN 61000-6-2 / EN 61000-6-4	
Protection class	IP 65/66/67 (according to EN 60529)	
Installation position	variable	
Approvals	CE, UL E172151	

Note

Detailed technical data about all available I/O types can be found in the document Signal Types, Installation, Configuration of the I/O modules, that may be found on the Beckhoff CD Products & Solutions or in the internet (http://www.beckhoff.com) under Download/Fieldbus Box.

3 Mounting and wiring

3.1 Dimensions



All dimensions are given in millimeters.

General

Technical data	Fieldbus Box
Material	PA6 (polyamide), casting compound: polyurethane
Assembly	2 x fixing holes for M3
Metal parts	Brass, nickel-plated
Contacts	CuZn, gold-plated
Vibration / shock resistance	according to EN 60068-2-6 / EN 60068-2-27, EN 60068-2-29
EMC resistance burst / ESD	according to EN 61000-6-2 (EN 50082) / EN 61000-6-4 (EN 50081)
Permissible ambient temperature during operation	0 ... 55°C
Permissible ambient temperature during storage	-25 ... + 85°C
Installation position	any
Type of protection	IP65/66/67 when screwed together
Approvals	CE, UL E172151

IPxxxx-Bxx8, IL230x-Bxx8, IL230x-B110, IXxxxx-B400, IXxxxx-B90x, IXxxxx-C900

Technical data	Compact and Coupler Box with integrated tee connector
Dimensions (H x W x D)	ca. 210 x 30 x 26,5 mm (height to upper edge of fieldbus socket: 30 mm)
Weight	ca. 260 g - 290 g, depending on module type

IPxxxx-Bxx0, IL230x-Bxx0, IL230x-Cxx0

Technical data	Compact and Coupler Box
Dimensions (H x W x D)	Approx. 175 x 30 x 26.5 mm (height to upper edge of fieldbus socket: 30 mm, with T- connector ZS1031-2600 height approx. 65 mm)
Weight	Approx. 250 g - 280 g, depending on module type

IExxxx

Technical data	Extension box
Dimensions (H x W x D)	Approx. 126 x 30 x 26.5 mm
Weight	Approx. 120 g - 200 g, depending on module type

3.2 Ethernet Connection

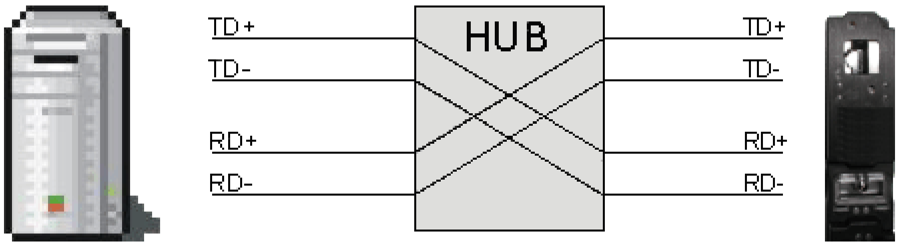
The connection to the Ethernet bus is made via an enclosed RJ45 connector (a Western plug).

The cable should be CAT5 or CAT5e cable.

Standard patch cable are also applicable. But this combination doesn't fulfill the corresponding class of protection.

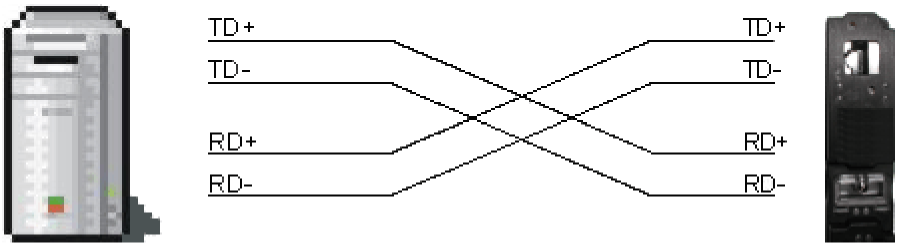
Cabling

Connection via hub or switch



Connect the PC's network card to the hub using a standard Ethernet cable, and connect the hub, again using a standard Ethernet cable, to the Fieldbus Box. Connection via a switch is done in the same way.

Direct connection between PC with Ethernet card and B900



To connect the PC directly to the Fieldbus Box, you must use an Ethernet cable in which the pairs of cores have been crossed (a crossover cable).

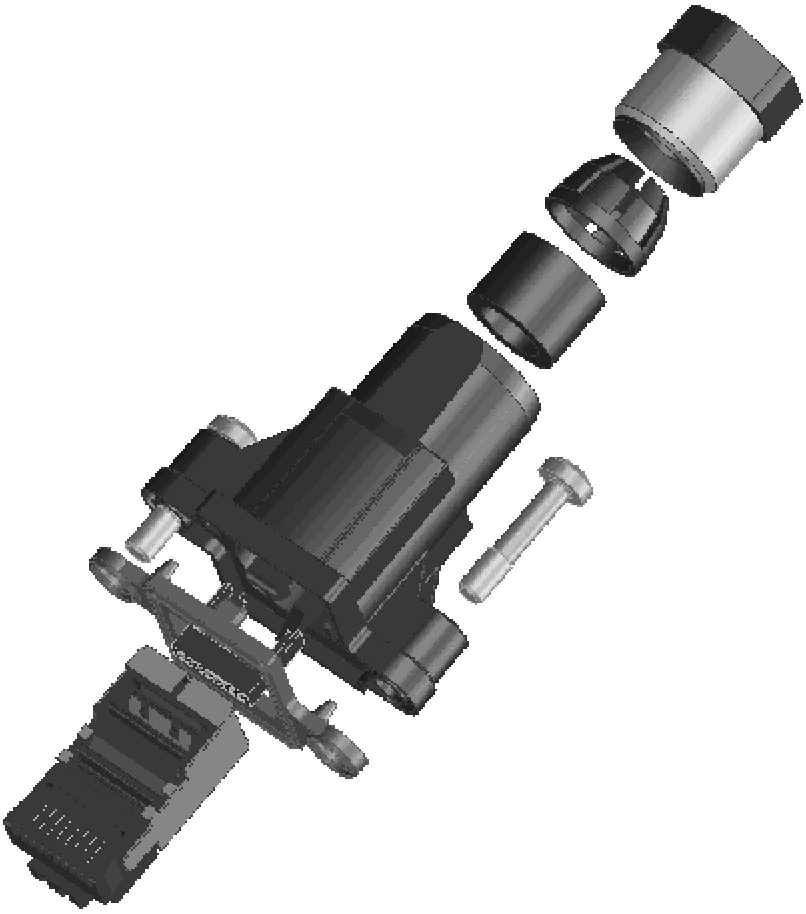
Pin assignment of the RJ45 connector

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

Note: There is no standardized color setting for the wires!

3.3 Ethernet connector: RJ45

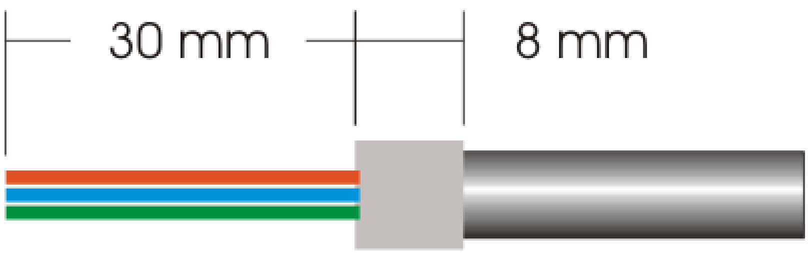
The connection to the Ethernet is done with an encapsulated IP67 RJ45 connector (Western plug).



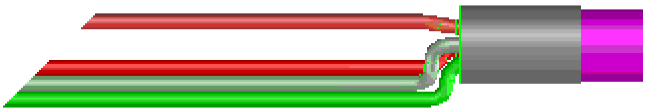
Assembly

Preparation of the cable

- push the cable gland and the connector case onto the cable.
- strip 30mm of the insulation from the cable.
- cut the shield mesh to 8mm and pull it back over the cable mantle.
- remove all other filling materials, shield strands, plastic wires etc..

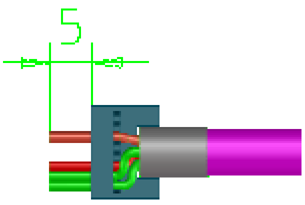


- align the single wires acc. to the pin assignment next to each other.
- cut the wires angular to ease the feed into the inlet guide(ca.45°).

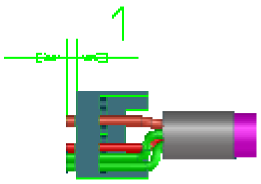


Assembly of the connector

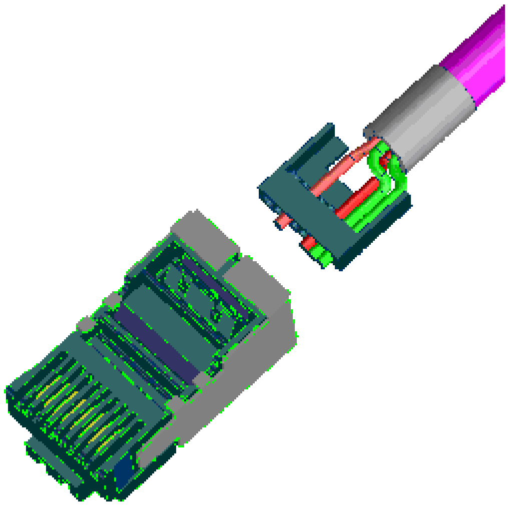
- feed the wires into the inlet guide according the pinning.
- pull the guide over the wires until the cut-out reaches the cable mantle.
- check if all wires are in the correct position.
- cut the wires rectangular to 5mm.



- pull the guide back to the end of the cable until only 1mm stays over.



- push the cable from the backside into the RJ45 connector with the guide . If the cable will not fit into the RJ45 due to its thickness, you can use an edgeless tool to flatten it to feed it into the connector.
- Crimp the cable, when the guide is in its stop position.



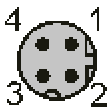
Pin assignment of the RJ45 connector

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

Note: There is no standardized color setting for the wires!

3.4 Ethernet connector: M12

The connection to the Ethernet is done with an d-coded M12 socket (IP67).



1	TD +
2	RD +
3	TD -
4	RD -
Housing	Shield

4 Ethernet

4.1 Overview

4.1.1 Ethernet

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

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

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

MAC-ID

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

The Internet Protocol (IP)

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

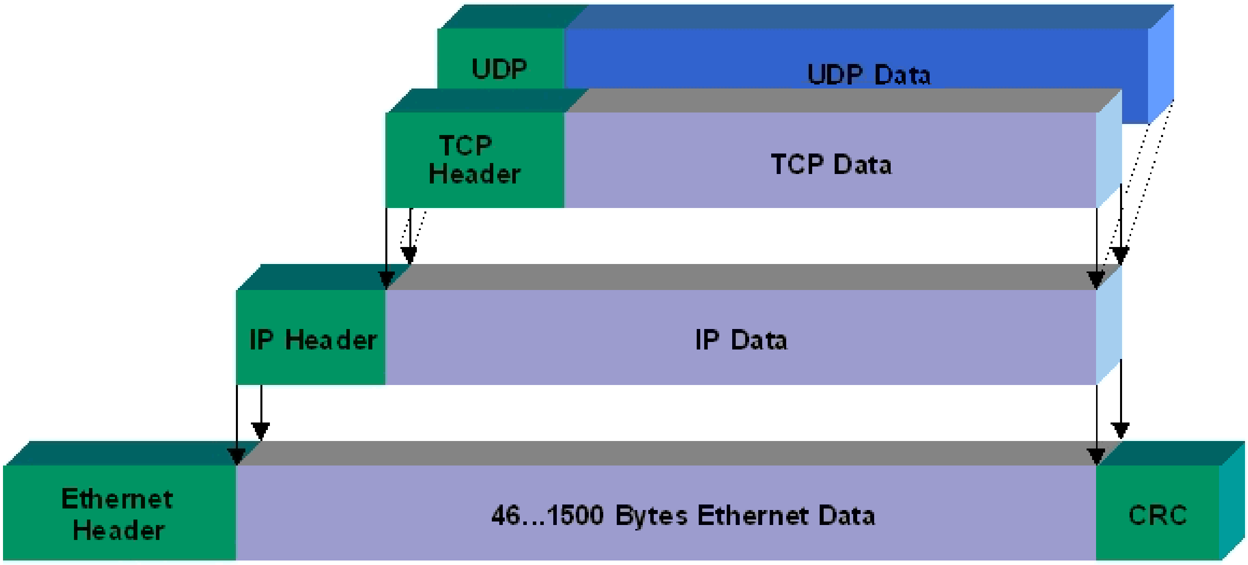
TCP/IP was developed to provide standardized, reliable data exchange between any number of different networks. TCP/IP is thus substantially independent of the hardware or software being used. Although the term is often used as if it were a single concept, a number of protocols are layered together: e.g. IP, TCP, UDP, ARP and ICMP.

Transmission Control Protocol (TCP)

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

User Datagram Protocol (UDP)

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

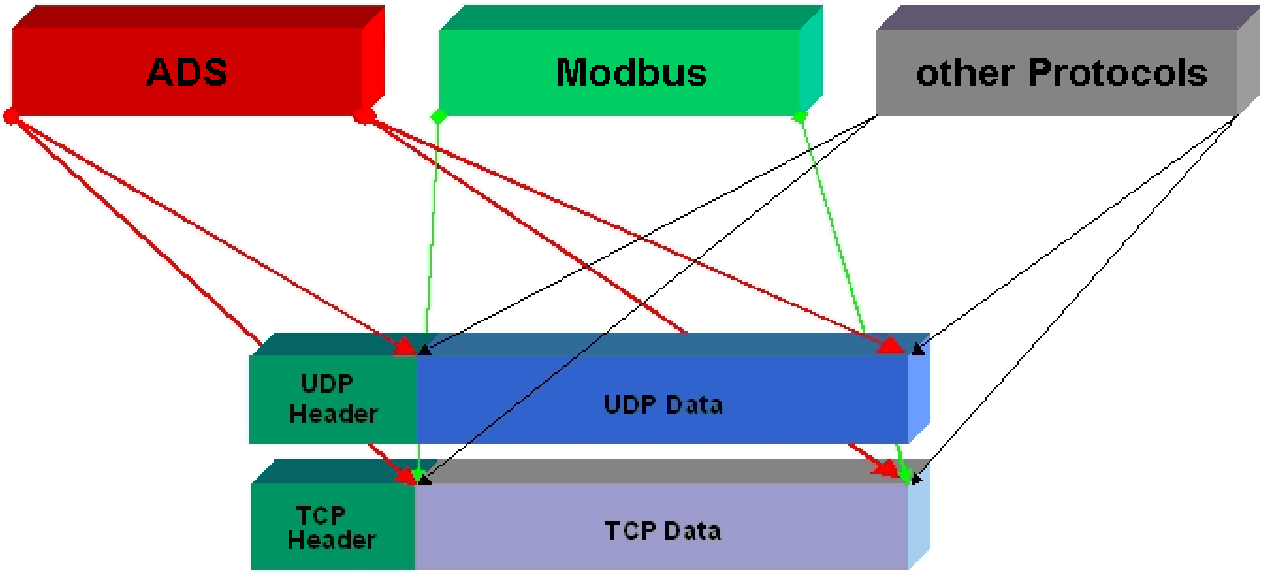


Protocols running on top of TCP/IP and UDP/IP

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

- ADS
- ModbusTCP

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



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

4.1.2 Topology

In 10BaseT and 100BaseT a number of stations are star connected according to the Ethernet standard.

Star topology

The simplest form of a star LAN consists of a single point-to-point connection. All messages pass via a central node (the hub or switch), which then passes the information to the desired device according to the destination address.

Tree topology

A tree topology consists of a number of connected star networks. As soon as the network contains a number of hubs or switches, the topology is classified as a tree. Ideally the connections between the star couplers have a particularly wide bandwidth, since these transport the most traffic. When constructing tree topologies, the repeater rule must be observed. This is also known as the 5-4-3 repeater rule. There must be no more than two pairs of repeaters (or of hubs) in the transmission path between any two stations, unless they are separated by bridges, switches or routers. A transmission path may consist of at most five segments and four repeater sets (two repeater pairs). Up to three of these segments may be coaxial segments to which the stations are connected. The remaining segments must consist of point-to-point connections; these are also known as IRL (inter repeater link) connections.

Cabling guidelines

Structured cabling provides general guidelines for constructing the cabling for a LAN. It specifies maximum permitted cable lengths for the wiring within the grounds, building or floor. Standardized in EN 50173, ISO 11801 and TIA 568-A, *structured cabling* provides the basis for an advanced, application-independent and economical network infrastructure. The wiring standards are applicable to a range defined as having a geographical extent of up to 3 km and an office area of up to one million sq meters, with between 50 and 50,000 end devices. Recommendations for the structure of a cabling system are also given. The figures can vary, depending on the topology selected, the transmission media and coupling modules used under industrial conditions, and on the use of components from various manufacturers in one network. The given figures should therefore only be considered as recommendations.

4.1.3 Ethernet Cable

Transmission standards

10Base5

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

10Base2

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

10BaseT

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

100BaseT

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

10BaseF

The 10BaseF standard describes several optical fiber versions.

Short description of the 10BaseT and 100BaseT cable types

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

UTP

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

S/UTP

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

FTP

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

S/FTP

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

STP

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

S/STP

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

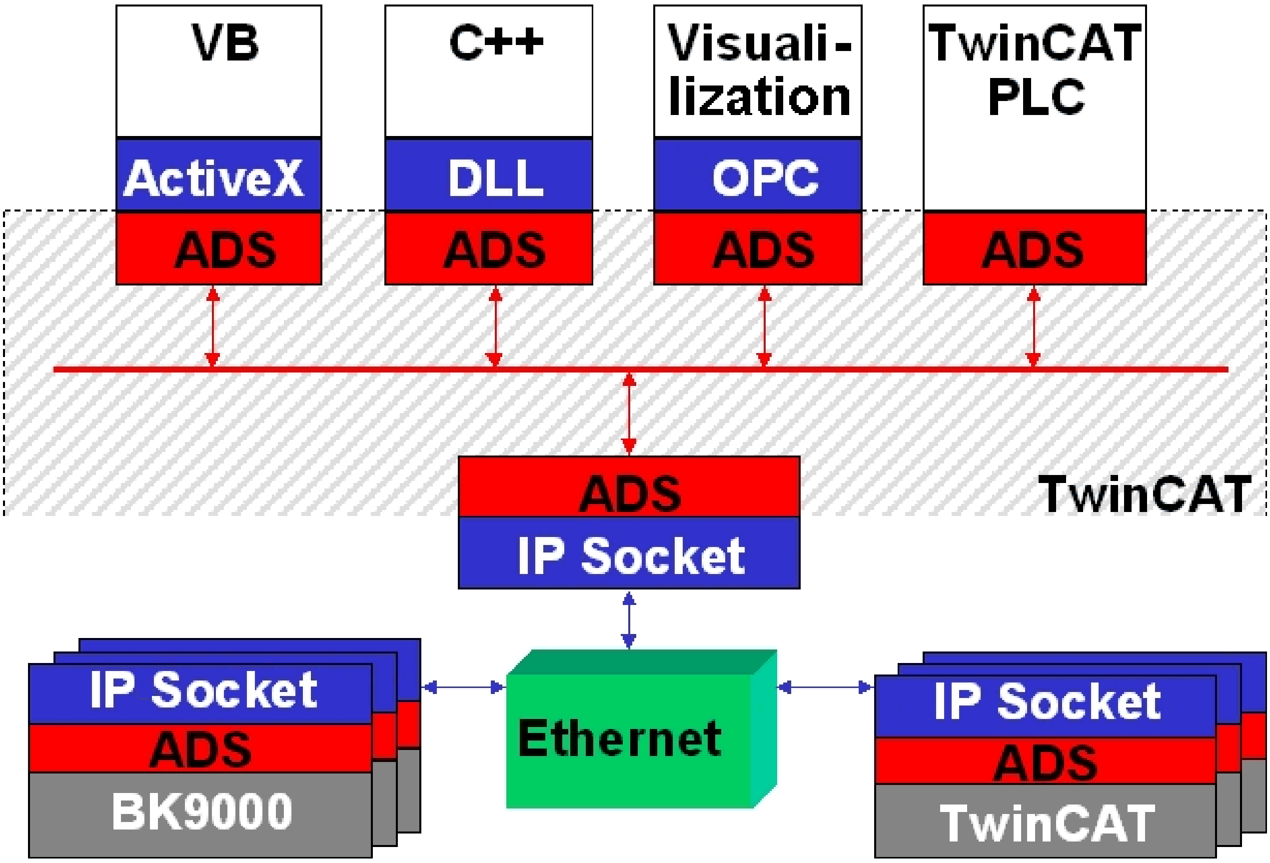
ITP

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

4.2 ADS communication

4.2.1 ADS Communication

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



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

Software interfaces

ADS-OCX

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

ADS-DLL

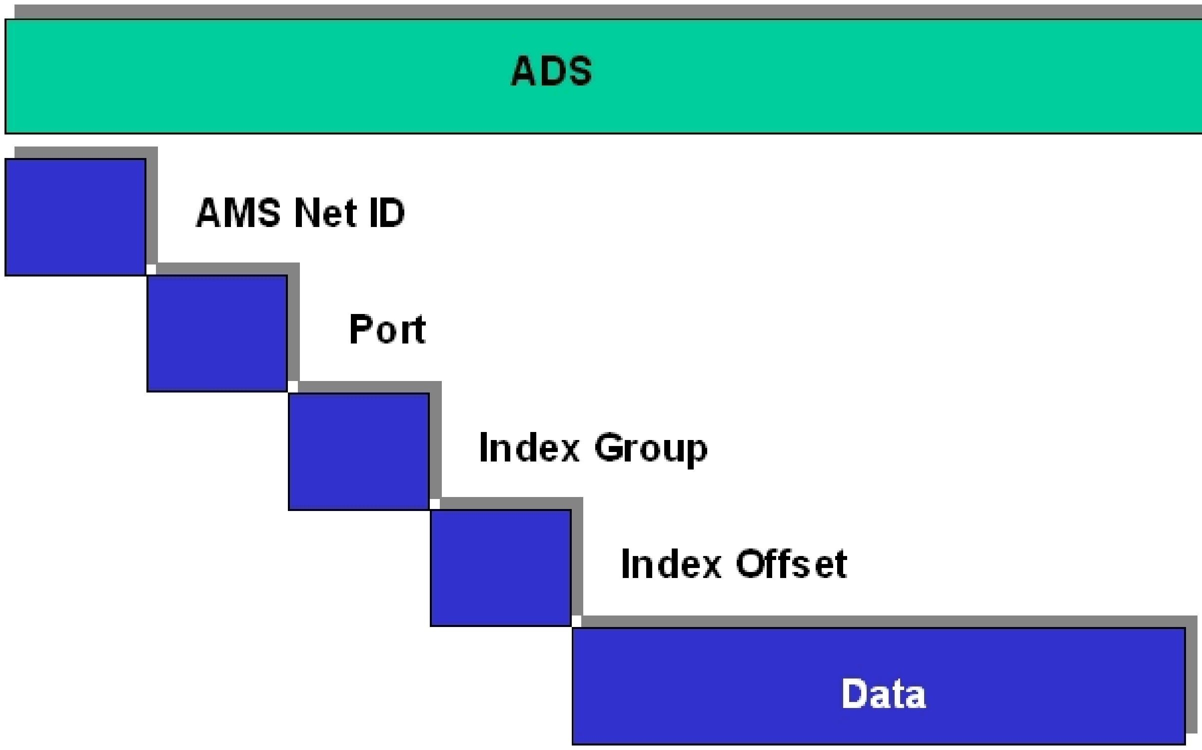
You can link the ADS-DLL (DLL: Dynamic Link Library) into your C program.

OPC

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

4.2.2 ADS Protocol

The ADS functions provide a method for accessing the Bus Coupler information directly from the PC. ADS function blocks can be used in TwinCAT PLC Control for this. The function blocks are contained in the *TcSystem.lib* library. It is also equally possible to call the ADS functions from AdsOCX, ADSDLL or OPC. It is possible to access all the data through ADS port number 300, and to access the registers of the Bus Coupler and Bus Terminals through ADS port number 100.



AMSNetID

The AMSNetID provides a reference to the device that is to be addressed. This is created from the set TCP/IP address and an additional 2 bytes. These additional 2 bytes consist of "1.1", and can not be altered.
Example:
IP address 172.16.17.128
AMSNetID 172.16.17.128.1.1

Port number

The port number distinguishes sub-elements in the connected device.
Port 100: Register access
Port 300: Fieldbus process data
Port 800: Local process data (BC90x0 only)

Index group

The index group distinguishes different data within a port.

Index offset

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

Len

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

TCP port number

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

4.2.3 ADS Services

Note
The user data of an ADS Telegram is maximum 1900 byte.

Process data port 300

The fieldbus data is accessed via ADS port number 300. The data is monitored by a watchdog. If no further telegram arrives within 1000 ms the outputs will be switched to the safe state.

Index group	Meaning	Index offset (value range)
0xF020	Inputs	0...511
0xF030	Outputs	0...511

Local process image port 800 (BC9000 only)

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 is 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...511
0xF030	Outputs	0...511
0x4020	Flags (BC 9000 only)	0...4096

ADS services

Table 1: AdsServerAdsStateB7

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

Table 2: AdsServerDeviceStateB7

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

Table 3: AdsServerTypeB7

Data type (read only)	Meaning
String	Coupler_PLC

Table 4: ADSWriteControlBA

Data type (write only)	Meaning
NetID	Net ID of the BC9000, C900
Port	800
ADSSTATE	5 - RUN / 6 - STOP
DEVSTATE	0
LEN	0
SRCADDR	0
WRITE	positive edge starts the block
TMOUT	for example: t#1000ms

Note

Note for the acyclic data communication, that the establishment of a TCP/IP connection takes app. 20 - 30 milliseconds. After successful establishment of the connection, the ADS data is to be read or written. An ADS read of about 1000 Byte takes app. 50 ms. If no data is transferred for 10 seconds, the TCP/IP connection is cut by the BC/BK9000, B/C900.

Register port 100

The ADS port number in the BK/BC9000 for register communication is fixed, being set at 100.

Index group	Index offset (value range)		Meaning
	Hi-Word	Low Word	
0	0...127	0...255	Registers in the Bus Coupler High word, table number of the Bus Coupler Low word, register number of the table
1...64	0...3	1...64	Register of the Bus Terminal High word, channel number Low word, register number of the Bus Terminal

Note

Note when reading the register that the time out for the ADS block is set to a time longer than 1 second.

Note

Note when writing to the registers that the password is set (see the documentation for the particular Bus Terminal).

Access control and association between IP and AMS-Net-ID[B6]

The AMS *Net-Id* table permits access control to the BC 9000 via AMS. As soon as this table has entries, only those AMS devices that have been entered will be able to access the BC 9000. An explicit association of the AMS-Net-ID and the IP address of the node is also made here.
The *AMS Net-Id* table can be filled with ADS write commands:
A maximum of 10 entries is possible.

Table 5: The structure

AMS Net ID	Size
AMS Net ID	6 bytes
IP address	4 bytes
Reserve	2 bytes
Reserve	4 bytes
Reserve	4 bytes

Access takes place via port number: 10.000

Index group: 700

Index Offset (write)	Comment	Data
0	Add an entry	Data structure, 20 bytes
1	Delete an entry	-
2	Delete all entries	-
10	Save the table in Flash memory	-

Index Offset (read)	Comment	Data
0	Number of entries	2 bytes
1...10	Entry n (1...10)	Data structure, 20 bytes

Note

The first entry must be that for the device that his writing into the table, because the settings have immediate effect. Make sure that all the settings are correct. The table can also be deleted if the end terminal only is inserted, and the DIP switches 1 to 7 are set to ON.

)[B6]: as from firmware B6
)[B7]: as from firmware B7
)[BA] : as from firmware BA

See the example [▸ 34]

4.2.4 AMS routing table

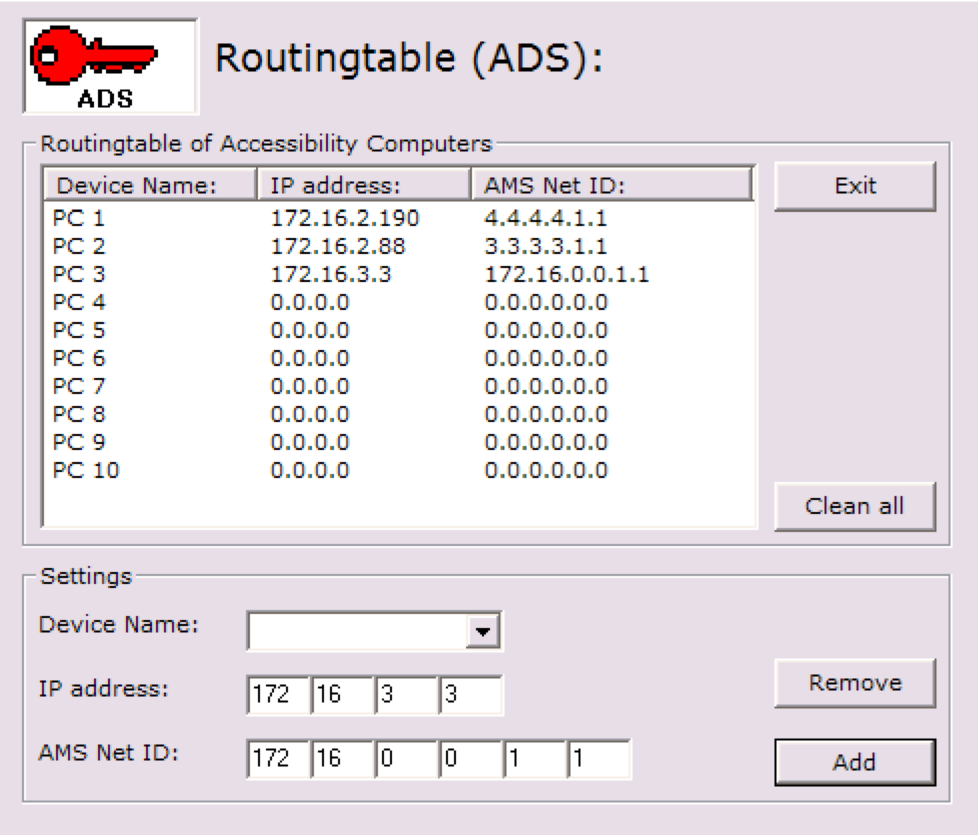
The AMS routing table can be used for two functions:

1. If the AMS Net ID does not match the TCP/IP address, the link between the two non-matching numbers can be entered here.
   The AMS Net ID matches the TCP/IP address if the first 4 bytes are identical and the AMS Net ID ends with ".1.1".
2. Only AMS Net IDs stored in the table can access the BC9000 via ADS.

Please ensure that all ADS devices with which communication is required are entered in the table. If the table is empty (default), all ADS devices can access the BC9000.

KS2000 dialog

From KS2000 version 4.3.0.39 entries in the AMS routing table can be made via dialog.



Entering values in the routing table via ADS

The AMS routing table can be loaded into the BC9000 via ADS. Ensure that the first entry is that of the sender, since this setting becomes effective immediately and may otherwise prevent further entries in the AMS routing table.

Example ST program: routingtable.pro (Resources/pro/9165521547.pro)

4.2.5 ADS Process Image

The process image is sub-divided into input image and output image. The Coupler Box first maps all complex (byte-oriented) Extension Boxes (IE3xxx, IE4xxx, IE5xxx, IE6xxx), into the process image in the order in which they are inserted in IP-Link. Afterwards all digital Extension Boxes (IE1xxx, IE2xxx) are added to the process image.

The complex (byte-oriented) Extension Boxes are always mapped with the following settings:

- complete evaluation
- Word alignment
- Intel format

Example 1:

IL2301-B900 (4DI, 4DO)
2 x IE1001 (8DI)
2 x IE2002 (8DO)
1 x IE3312 (4 AI)
1 x IE4132 (4AO)

Position	Module	Input %I..Byte Offset	Output %Q..Byte Offset
1	IE3312	0...7	0..7
2	IE4132	8...15	8...15
3	IL2301	16.0...16.3	16.0...16.3
5	IE1001	16.4...17.3	-
6	IE2002	-	16.4...17.3

Note

Detailed description about the mappings of the Fieldbus Box Modules can be found in the document Signal Types, Installation, Configuration of the I/O modules, that may be found on the BECKHOFF CD Products & Solutions or in the internet (http://www.beckhoff.com) under Download/Field-bus Box.

4.2.6 Real Time Ethernet with Fast ADS

The Fast ADS protocol is based on Ethernet. In combination with TwinCAT, this bus system has real-time capability and reaches cycle times of < 1ms. The real-time Ethernet driver was developed for Windows 2000, Windows XP and Intel Ethernet chips.

Further information can be found at http://www.pc-control.net/pdf/022002/pcc_ethernet_e.pdf.

BK9000

On the BK9000 from firmware version B8 the Fast ADS protocol is implemented in parallel to the existing protocols. Simultaneous utilization is only possible, if the function ADS was selected in the System Manager (*IP Address* tab). If the function *NoRealTimeFlag* is not selected, communication is exclusively via Fast ADS.

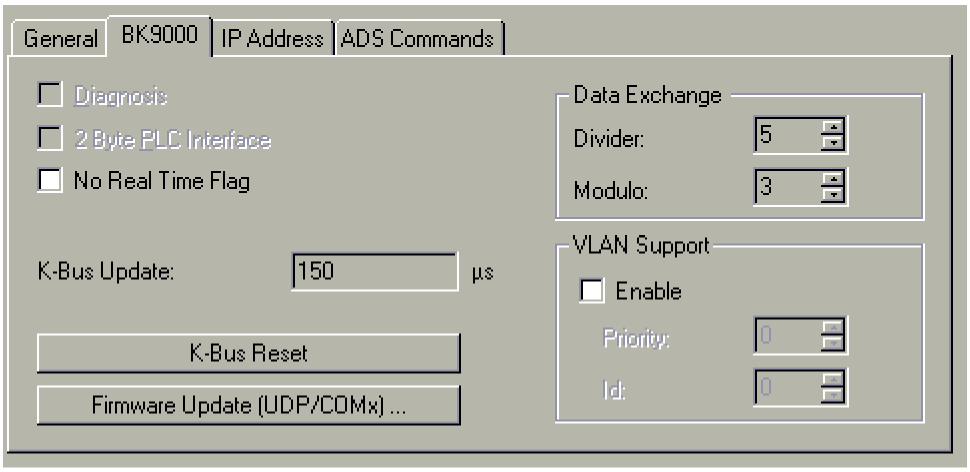
With Fast ADS, the watchdog is set to 100 ms. If it is triggered (e.g. if TwinCAT stops), all other Ethernet services are usable again. The K-Bus works synchronous with the Fast ADS.

B900

RT-Ethernet respectively Fast ADS is implemented on all B900 modules. The task cycle time of the TwinCAT has to be smaller than 10ms. Otherwise the missing telegram will be interpreted as error and the outputs will be set to zero!

Settings at the System Manager

BK9000 tab



No Real Time Flag

With activated flag TCP/IP or UDP telegrams are also able to be received. But the Jitter of the real time telegrams increases!

K-Bus Reset

On the K-Bus a reset is released.

Data Exchange

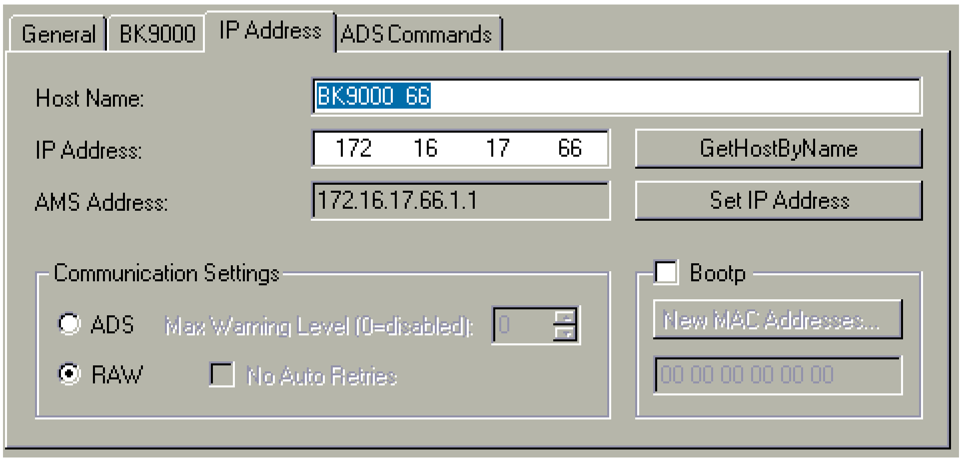
This defines, after how many task cycles a telegram is to be sent to the Fieldbus Station (Bus Coupler / Fieldbus Box)
Modulo sets, in witch Task the telegram is to be sent. This enables to optimize the system performance.

Example: Divider 5 and Modulo 3 - after each 5th Task (in fact the 3rd) a telegram is sent.

VLAN

VLAN (virtual LANs) supporting Switches are able to prioritize incoming telegrams by this.

IP Address tab



Host Name

The name of the fieldbus station (Bus Coupler / Fieldbus Box) can be edited here.

IP Address

The IP-Address of the fieldbus station (Bus Coupler / Fieldbus Box) is to be set here.

GetHostByName

With this button, an IP address can (BK9000: from Firmware-Version B2) be taken from a DHCP-Server (Dynamic Host Configuration Protocol Server).

Set IP Address

With this button, the IP address is set via ARP.

AMS Address

Shows the AMS-Net ID of the fieldbus station (Bus Coupler / Fieldbus Box). It is automatically generated, by adding two extra byte values (".1.1") to the IP address.

BootP

This Checkbox is selectable, if the Beckhoff BootP Server is installed and started (from TwinCAT Version 2.8).

New MAC Addresses

With started Beckhoff BootP Server, the new MAC addresses (*Media Access Controller* addresses) can be displayed by clicking this button.
The new addresses are displayed in the field under the button.

If only one new fieldbus station was connected, accordingly only one new MAC address is displayed.
To this new MAC address a desired IP-address can be allocated in the field *IP Address* mentioned above.

If further fieldbus stations are connected, you can repeat this procedure on the tabs *IP Address* of this stations.

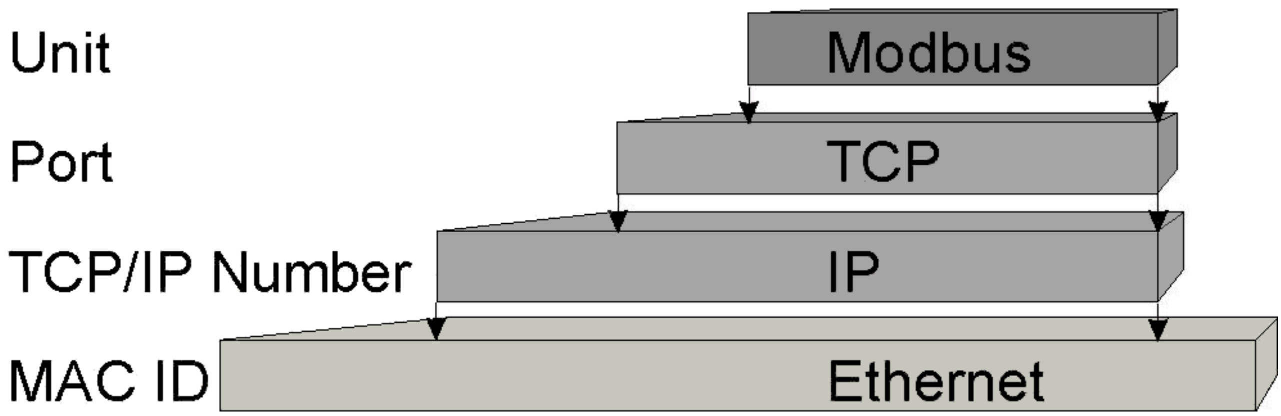
Communication Settings

Settings about communication via ADS or RAW.

4.3 Modbus TCP

4.3.1 ModbusTCP Protocol

The Ethernet protocol is addressed by means of the MAC-ID. The user does not normally need to be concerned about this address. The IP number has a length of 4 bytes, and must be parameterized by the user on the Bus Coupler and in the application. In ModbusTCP, the TCP port is set to 502. The UNIT can be freely selected under ModbusTCP, and does not have to be configured by the user.



TCP port number

The TCP port number for ModbusTCP has been standardized to 502.

Modbus-Unit

The unit is returned by the slave.

ModbusTCP Protocol

Byte	Name	Description
0	Transaction identifier	Is returned by the slave
1	Transaction identifier	Is returned by the slave
2	Protocol identifier	Always 0
3	Protocol identifier	Always 0
4	Length field	0 (if the message is less than 256 bytes in length)
5	Length field	Number of following bytes
6	UNIT identifier	Is returned by the slave
7	Modbus	The Modbus protocol with the function follows

4.3.2 Modbus TCP interface

Address		Description		
0x0000 0x00FF		Process data interface Inputs		
0x0800 0x08FF		Process data interface Outputs		
0x1000 0x1006	Read only	Bus Coupler identification		
0x100A		2 byte PLC interface		
0x100B		Bus terminal diagnosis		
0x100C		Bus Coupler status		
0x1010		Process image length in bits, analog outputs (without PLC variables)		
0x1011		Process image length in bits, analog inputs (without PLC variables)		
0x1012		Process image length in bits, digital outputs		
0x1013		Process image length in bits, digital inputs		
0x1020		Watchdog, current time in [ms]		
0x110A	Read/Write	2 byte PLC interface		
0x110B		Bus terminal diagnosis		
0x1120		Watchdog, pre-defined time in [ms] (Default value: 1000)		
0x1121		Watchdog Reset Register		
0x1122		Type of watchdog	1	Telegram watchdog (default)
			0	Write telegram watchdog
0x1123**		ModbusTCP mode**	1	Fast Modbus
			0	Normal Modbus (default)
0x4000* 0x47FF		Flags area (%MB..)*		

* Busterminal Controller BC9xx0 and BX9000
** for BC9x00 with firmware B7 or higher and BK9000 with firmware B5 or higher andallBK9xxxandBC/BX9xxxnotlisted

Watchdog

The watchdog is active under the factory settings. After the first write telegram the watchdog timer is initiated, and is triggered each time a telegram is received from this device. Other devices have no effect on the watchdog. A second approach, which represents a more sensitive condition for the watchdog, is for the watchdog only to be re-triggered after each write telegram. To do this, write a zero into register 0x1122 (default value "1").
The watchdog can be deactivated by writing a zero to offset 0x1120. The watchdog register can only be written if the watchdog is not active. The data in this register is retained.

Watchdog register

If the watchdog timer on your slave has elapsed it can be reset by writing twice to register 0x1121. The following must be written to the register: 0xBECF 0xAFFE. This can be done either with function 6 or with function 16.

The Bus Coupler's status register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FB	-	-	-	-	-	-	-	-	-	-	-	-	-	CNF	KB

Table 6: Key

Bit	Name	Value	Description
15	FB	1_{bin}	Fieldbus error, watchdog time elapsed
14...2	-	-	reserved
1	CNF	1_{bin}	Bus Coupler configuration error
0	KB	1_{bin}	Bus Terminal error

ModbusTCP mode

The fast Modbus mode should only be used in small local networks. The fast ModbusTCP is not active under the default settings. If problems are found to occur with this type of communication, the Bus Coupler should be switched to "normal" ModbusTCP communication. The mode is set in the Modbus interface, offset 0x1123. It is necessary to reset the coupler (e.g. using ModbusTCP function 8) after the change. It is not permitted to send more than one Modbus service within one Ethernet frame in fast Modbus mode.

2 byte PLC interface

Registers in the complex terminals and Bus Terminal Controller registers can be both read and written using the 2 byte PLC interface. The complex terminal registers are described in the associated terminal documentation. The Bus Coupler registers can be used, for example, to read terminal bus diagnostics data, the terminal composition or the cycle times, and the programmed configuration can be written. It is also possible for a manual terminal bus reset to be carried out. The 2-byte PLC interface requires two bytes each of input and output data; a special protocol is processed via these 2 bytes. A description of the 2 byte PLC interface, the registers available in the Bus Couplers and of function blocks for various PLC systems that support the 2 byte PLC interface can be supplied on request.

2 byte diagnostic interface

The terminals' error messages can be sent over the 2-byte diagnostic interface. Terminal bus diagnostics must however be activated for this purpose. The 2-byte diagnostic interface occupies two bytes each of output and input data. A special protocol is processed via these two bytes. A description of the 2 byte-diagnostic interface can be supplied on request.

4.3.3 ModbusTCP slave error answer (BK9000, BC9000, IP/ILxxxx-B/C900)

When the user sends the slave either a request or information that the coupler does not understand, the slave responds with an error report. This answer contains the function and the error code. 0x80 is added to the value returned by the function.

Code	Name	Meaning
1	ILLEGAL FUNCTION	Modbus function not implemented
2	ILLEGAL DATA ADDRESS	Invalid address or length
3	ILLEGAL DATA VALUE	Invalid parameter - Diagnostic functions - Incorrect register
4	SLAVE DEVICE ERROR	Watchdog or K-Bus error
6	SLAVE DEVICE BUSY	Output data is already been received from another IP device

4.3.4 ModbusTCP Process Image

The ModbusTCP process image makes a fundamental distinction between digital and byte-oriented (word-oriented) signals (Bus Terminals). Additionally, a distinction is made between inputs and outputs. The mapping table shows which Bus Terminals belong to which mapping.

At word-oriented access, first the analog and then the digital modules are listed in the address room.

Table 7: Byte-oriented access

Digital inputs	Digital outputs
Offset 0 ... 4095 Modbus function Reading 2 Writing: -	Offset 0 ... 4095 Modbus function Reading 1 Writing5, 15

Table 8: Word-oriented access

Analog/digital inputs	Analog/digital outputs
Offset 0x000 ... 0x0FF Modbus function Reading 3, 4, 23 Writing: -	Offset 0x800 ... 0x8FF Modbus function Reading 3, 23 Writing6, 16, 23

This will be clarified by example.

Example

IL2301-B900 (Remark: The IL2301-B900 has got 4 in- and 4 outputs on the module)
1 x IE1001
1 x IE2001

Inputs			Outputs		
Modbus function	**Modbus Address/Offset**	**Fieldbus Box Module**	**Modbus function**	**Modbus Address/Offset**	**Fieldbus Box Module**
3 read	0x0000	4 Bit IL2301	3 read	0x0800	4 Bit IL2301
4 read	0x0000	8 Bit IE1001	1 read	0x0000	8 Bit IE2001
2 read	0x0000 - 0x0001		5 write	0x0000-0x000B	
			15 write	0x0000-0x000B	
			6 write	0x800	
			16 write	0x800	
			23 write	0x800	

4.3.5 ModbusTCP functions

4.3.5.1 ModbusTCP Functions

In the Modbus protocol, the functions determine whether data is to be read or written, and what kind of data is involved.

Function	Code	Description
Read coil status [▸ 43]	1	Read digital outputs
Read input status [▸ 44]	2	Read digital inputs
Read holding register [▸ 44]	3	Read analog outputs and inputs / GPR
Read input register [▸ 45]	4	Read analog inputs / GPR
Force single coil [▸ 45]	5	Writing a digital output
Preset single register [▸ 46]	6	Write one analog output / GPR
Diagnostic [▸ 46]	8	Diagnostic
Force multiple coils [▸ 48]	15	Write a number of digital outputs
Preset multiple register [▸ 49]	16	Write a number of analog outputs / GPRs
Read / Write Registers [▸ 49]	23	Write and read a number of process data outputs / GPRs

GPR (General Preset Register) - register structure of the Modbus interface (see Appendix)

4.3.5.2 Read Coil Status (Function 1)

The *Read Coil Status* function can be used to read the digital outputs that have been set.

The first 10 digital outputs are read in this example. The start address is zero. An offset can be entered in the *Start address* field.

Query

Byte Name	Example
Function code	1
Start address high	0
Start address low	0
Count high	0
Count low	10

The fieldbus coupler answers with *byte count* 2, i.e. 2 bytes of data are returned. The query was for 10 bits, and these are now distributed over 2 bytes. The third bit in the output process image of the BK9000 is set, and the fieldbus coupler returns the value 4 in the first data byte.

Response

Byte Name	Example
Function code	1
Byte Count	2
Data bits 0...7	4
Data bits 8...18	0

4.3.5.3 Read Input Status (Function 2)

The *Read Input Status* function can be used to read the digital input data. The first 10 digital inputs are read in this example. The start address is zero. An offset can be entered in the *Start address* field

Query

Byte Name	Example
Function code	2
Start address high	0
Start address low	0
Count high	0
Count low	10

The fieldbus coupler answers with *Byte count* 2, i.e. two bytes of data are returned. The query was for 10 bits, and these are now distributed over 2 bytes. The third bit in the output process image of the BK9000 is set, and the fieldbus coupler returns the value 4 in the first data byte.

Response

Byte Name	Example
Function code	2
Byte Count	2
Data bits 0...7	1
Data bits 8...18	0

4.3.5.4 Read Holding Register (Function 3)

The *Read Holding Register* function can be used to read the input and output words and the registers. The inputs have offsets 0 - 0xFF while the outputs have offsets 0x800 - 0x8FF.

The first two analog outputs are read in this example. The analog outputs begin at offset 0x800. The length indicates the number of channels to be read.

Query

Byte Name	Example
Function code	3
Start address high	8
Start address low	0
Count high	0
Count low	2

The fieldbus coupler answers with byte count 4, i.e. 4 bytes of data are returned. The query was for two analog channels, and these are distributed over two words. In the analog output process image, the first channel has the value 0x3FFF, while the second channel has the value 0x0.

Response

Byte Name	Example
Function code	3
Byte Count	4
Data 1 high byte	63
Data 1 low byte	255
Data 2 high byte	0
Data 2 low byte	0

4.3.5.5 Read Input Register (Function 4)

The *Read Input Register* function reads the analog inputs.

In this example the first two analog inputs of slave number 11 are read. The analog outputs start at an offset of 0x0000. The length indicates the number of words to be read. A KL3002 has two words of input data, which is why the value to be entered in *Count low* is two.

Query

Byte Name	Example
Function code	4
Start address high	0
Start address low	0
Count high	0
Count low	2

The fieldbus coupler answers with byte count 4, i.e. four bytes of data are returned. The query was for two analog channels, and these are now distributed over 2 words. In the analog input process image, the first channel has the value 0x0038, while the second channel has the value 0x3F1B.

Response

Byte Name	Example
Function code	4
Byte Count	4
Data 1 high byte	0
Data 1 low byte	56
Data 2 high byte	63
Data 2 low byte	11

4.3.5.6 Force Single Coil (Function 5)

The *Force Single Coil* function can be used to write a digital output. The third digital output is written in this example. The digital outputs start at address 0x0000. The digital value is located in *Data high*. To switch the output on, *Data high* must contain the value 0xFF, while 0x00 is used to switch the output off again. *Data low* must contain the value 0x00.

Query

Byte Name	Example
Function code	5
Start address high	0
Start address low	2
Data high	255
Data low	0

The coupler answers with the same telegram.

Response

Byte Name	Example
Function code	5
Start address high	0
Start address low	2
Data high	255
Data low	0

4.3.5.7 Preset Single Register (Function 6)

The *Preset Single Register* function can be used to access the output process image or the flag process image (only at the controller) and the Modbus TCP interface [▶ 40].

The first analog output is written with function 6. The analog outputs start at an offset of 0x0800. Here again the offset always describes a word. This means offset 0x0803 refers to the fourth word in the output process image.

Query

Byte Name	Example
Function code	6
Start address high	8
Start address low	0
Data high	63
Data low	255

The Fieldbus Coupler replies with the same telegram and confirmation of the received value.

Response

Byte Name	Example
Function code	6
Start address high	8
Start address low	0
Data high	63
Data low	255

4.3.5.8 Diagnosis (Function 8)

The diagnosis function provides a series of tests for checking the communication system between the master and the slave and for examining a variety of internal error states within the slave. A broadcast telegram is not supported.

The function uses two bytes in the query to specify a sub function code defining the test that is to be carried out. The slave returns the function code and the sub function code in the response.

The diagnostic queries use a two-byte data field to send diagnostics data or control information to the slave.

Query

Byte Name	Example
Function code	8
Sub function high	0
Sub function low	0
Data high	2
Data low	3

Response

Byte Name	Example
Function code	8
Sub function high	0
Sub function low	0
Data high	2
Data low	3

Echo a request (Sub function 0)

Sub function 0 causes the data that is sent to the slave by the master to be returned.

Coupler reset (Sub function 1)

Sub function 1 re-initializes the BC9000/BK9000. Error counters are reset, and the Bus Terminal Controller executes a self-test. No telegrams are either received or sent while the Bus Terminal Controller is being reset. The IP socket is closed.

Note

Before the Bus Terminal Controller restarts it sends one more answer with sub function 1, after which the IP socket is closed.

Sub function	Data field (query)	Data field (response)
0x0001	0x0000	0x0000

Delete All Counter Contents (Sub function 10)

Calling this sub function deletes the contents of all error counters in the Bus Terminal Controller.

Sub function	Data field (query)	Data field (response)
0x000A	0x0000	Echo query data

Bus Communication Answer Counter (Sub function 11)

Returns the number of communication answers.

Sub function	Data field (query)	Data field (response)
0x000B	0x0000	Counter value

Error Answer Counter (Sub function 13)

This counter contains the number of error answer telegrams that the coupler has sent.

Sub function	Data field (query)	Data field (response)
0x000D	0x0000	Counter value

The following functions contain the counter states for various units. This means that the Modbus telegrams can be distinguished through their units, if, for example, a Bus Coupler is accessed by different masters.

Slave Answers (Sub function 14)

Contains the number of answers that the slave has sent.

Sub function	Data field (query)	Data field (response)
0x000E	0x0000	Counter value

Number of unanswered telegrams (Sub function 15)

Contains the number of answers that the slave has not sent.

Sub function	Data field (query)	Data field (response)
0x000F	0x0000	Counter value

Number of Error Answers (Sub function 16)

Contains the number of error answers that the slave has sent.

Sub function	Data field (query)	Data field (response)
0x0010	0x0000	Counter value

4.3.5.9 Force Multiple Coils (Function 15)

The *Force multiple coils* function can be used to set or reset a number of digital outputs at the same time.

The first 20 digital outputs are written in this example. The digital outputs start at an offset of 0x0000. Here the offset always describes a bit. Offset 0x0003 writes to the fourth bit in the output process image. The length indicates the number of bits, and the *Byte count* is formed from the combination all the bytes that are to be written.

Example: 20 bits yield a byte count of 3 (rounded up to a byte boundary).

The data bytes contain the values for the individual bits. In this example, bits 0 to 15 are set to TRUE, while bits 16 to 23 are FALSE.

Query

Byte Name	Example
Function code	15
Start address high	0
Start address low	0
Length high	0
Length low	20
Byte Count	3
Data 1 bit 0...7	255
Data 2 bit 8...15	255
Data 3 bit 16...23	0

Response

The Bus Coupler answers with the same telegram.

Byte Name	Example
Function code	15
Start address high	0
Start address low	0
Length high	0
Length low	20

4.3.5.10 Preset multiple register (Function 16)

The *Preset Multiple Register* function can be used to write a number of analog outputs. The first two analog output words are written in this example. The analog outputs start at an offset of 0x0800. Here the offset always describes a word. Offset 0x0003 writes to the fourth word in the output process image. The length indicates the number of words, and the *Byte count* is formed from the combination of all the bytes that are to be written.

Example: 4 words – correspond to a byte count of 8

The data bytes contain the values for the analog outputs. In this example, two words are to be written. The first word is to receive the value 0x7FFF, and the second word is to receive the value 0x3FFF.

Query

Byte Name	Example
Function code	16
Start address high	8
Start address low	0
Length high	0
Length low	2
Byte Count	4
Data 1 byte 1	127
Data 1 byte 2	255
Data 2 byte 1	63
Data 2 byte 2	255

Response

The coupler replies with the start address and the length of the transmitted words.

Byte Name	Example
Function code	16
Start address high	8
Start address low	0
Length high	0
Length low	2

4.3.5.11 Read or Write Registers (Function 23)

A number of analog outputs can be written and a number of analog inputs read with one telegram using the *Read / Wrrite Registers* function. In this example the first two analog output words are written, and the first two analog inputs are read. The analog outputs start at offset 0x0800, while the inputs start at offset 0x0000. Here the offset always describes a word. Offset 0x0003 writes to the fourth word in the output process image. The length indicates the number of words, and the *Byte count* is formed from the combination of all the bytes that are to be written. Example: 4 words – correspond to a byte count of 8

The data bytes contain the values for the analog outputs. In this example, two words are to be written. The first word is to receive the value 0x3FFF, and the second word is to receive the value 0x7FFF.

Query

Byte Name	Example
Function code	23
Read start address high	0
Read start address low	0
Read length high	0
Read length low	2
Write start address high	8
Write start address low	0
Write length high	0
Write length low	2
Byte Count	4
Data 1 high	63
Data 1 low	255
Data 2 high	127
Data 2 low	255

Response

The coupler replies with the start address and the length of the bytes to be transferred in *Byte count*. The data information follows. In this example the first word contains 0x0038 while the second word contains 0x3F0B.

Byte Name	Example
Function code	23
Byte Count	4
Data 1 high	0
Data 1 low	56
Data 2 high	63
Data 2 low	11

4.3.5.12 Examples for ModbusTCP

Examples for Modbus TCP are contained in this help text.

Simple example: ModbusTCP via VB6.0modbustcp.zip (Resources/zip/9165522955.zip)

Example: ModbusTCP via VB6.0, set and reset of the watchdog:bk9000modbus.zip (Resources/zip/9165524363.zip)

5 Parameterisation and Commissioning

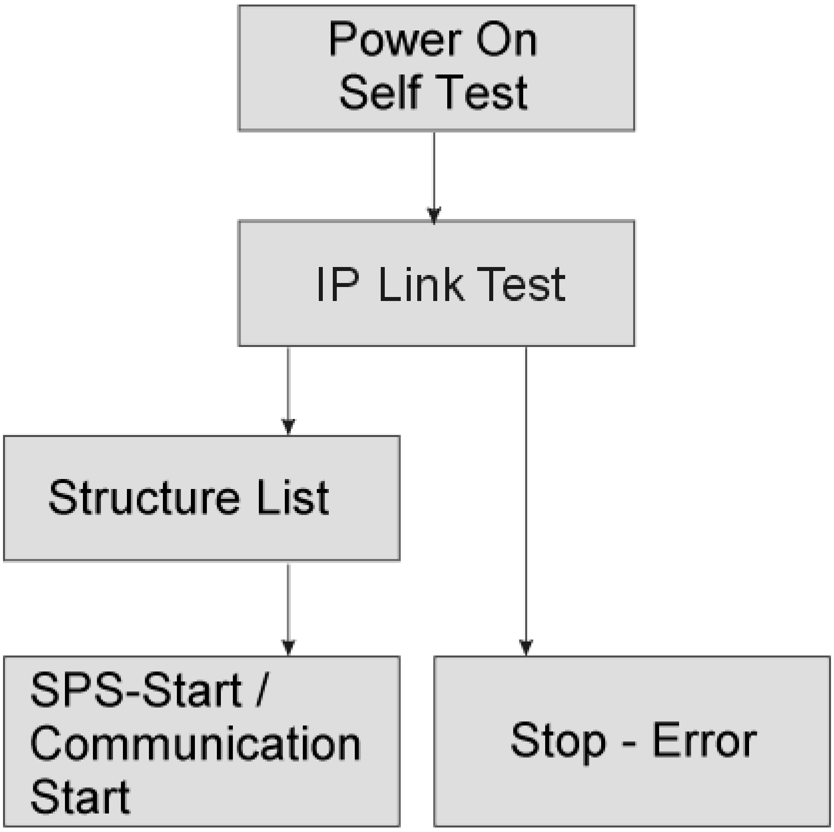
5.1 Note about parameterization

Note

Changes, e.g. the MAC-ID, that were done with the KS2000 configuration software are only stored in the volatile memory (RAM) of the Fieldbus Box.After the changes a software reset is required. By this, the changes will be copied into the flash memory and are permanent.A Cold-Start (Power-ON/OFF) is not enough. It has to be a software reset!

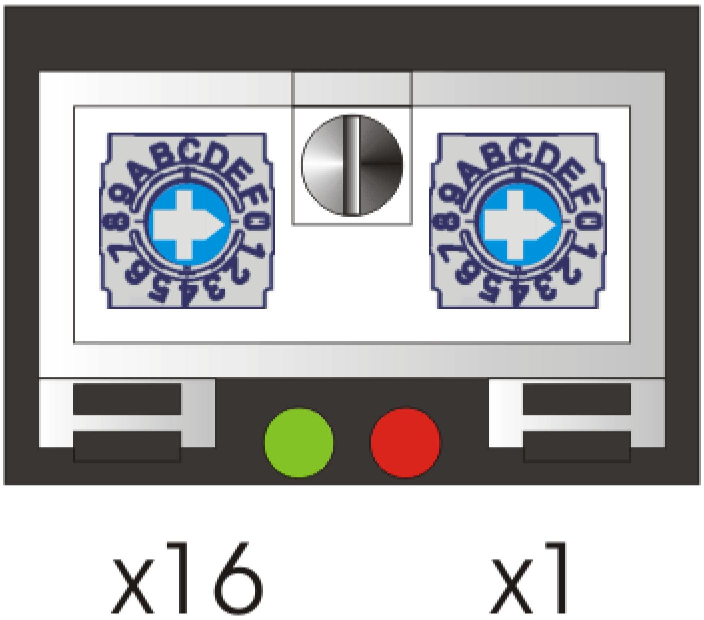
5.2 Start-up behavior of the Fieldbus Box

After power up, the Fieldbus Box checks its state, configures the IP-Link (if present) and refers to the extension modules to create a structure list. If the Fieldbus Box contains a decentralized controller (IL230x-C310) the local PLC is started once the structure list has successfully been created.
The I/O LEDs illuminate and flash as the module starts up. If there are no errors, the I/O LEDs should stop flashing within about 2-3 seconds. If there is an error, then the LED that flashes will depend on the type of that error (see Diagnostic LEDs [▶ 74]).



5.3 Parameterizing via Rotary Switches

By using the rotary switches, the following parameterizations are possible without software changes.



Hex switch x10	Hex switch x1	Meaning
0x0 - D	0x0 - F	IP address direct via switch
0xF	0x0	IP address via DHCP server
0xF	0x1	IP address via BootP server
0xF	0x2	BootP save address
Hex switch x10	**Hex switch x1**	**Meaning (Power ON/OFF)**
0xF	0x3	Fast ModbusTCP active (only B900, B901)
0xF	0x4	Fast ModbusTCP inactive (only B900, B901)
0xF	0x5	BOOTP / DHCP Server 2.3 (Rockwell) (only B905)
0xF	0x6	reserved
0xF	0x7	reserved
0xF	0x8	reserved
0xF	0x9	reserved
0xF	0xA	reserved
0xF	0xB	reserved
0xF	0xC	deletes AMS NetID table
0xF	0xD	sets fieldbus specific default values
0xF	0xE	deletes boot project (only for Controller Box IL230x-C900)
0xF	0xF	Sets module back to manufacturer settings

Restoring the manufacturer settings

- Switch of the module.
- Set the rotary switches to 0xFF and switch on the module.
- After setting the default parameter successfully, the LEDs *I/O RUN* and *I/O ERR* are flashing simultaneously.

- Now you can switch of the module again, to set the IP address.

Delete the boot project (only for Controller Box IL230x-C900)

- Switch of the module.
- Set the rotary switches to 0xFE and switch on the module.
- After deleting the boot project successfully, the LEDs *I/O RUN* and *I/O ERR* are flashing simultanously.
- Now you can switch of the module again, to set the IP address.

Setting the Ethernet parameters

- Switch of the module.
- Set the rotary switches to 0xFD and switch on the module.
- The LEDs *I/O RUN* and *I/O ERR* are flashing.
- The default values of table 100 are set.
- Now you can switch of the module again, to set the IP address.

5.4 Sample for KS2000 view IP address 172.16.200.xxx

The registers can be set through a dialog using the KS2000 configuration software, or it is possible to write into the registers directly.

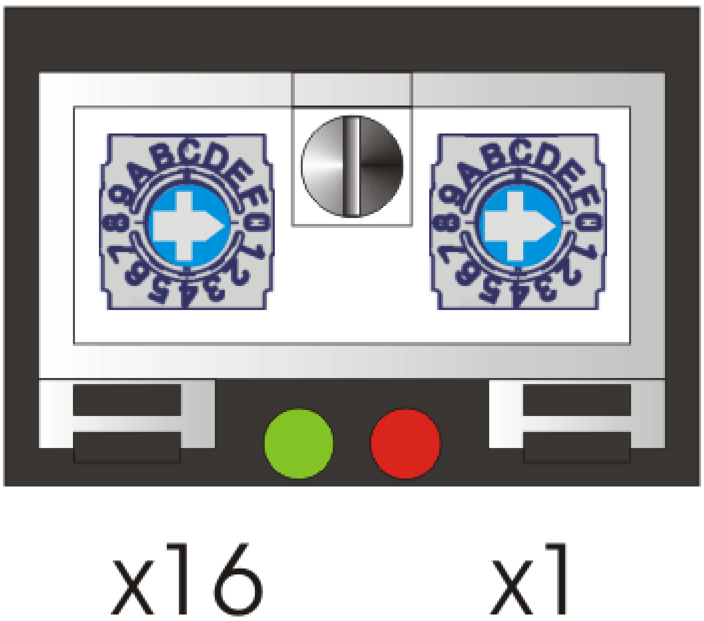
Register	Description		Default
0-1	IP address		0xAC, 0x10, 0x12, 0x00
2-3	IP mask		0xFF, 0xFF, 0x00, 0x00
4-13	Device Name		IL-B900 (ASCII coded)
14	Watchdog AMS/ADS		1000 ms
15.0	0_{bin}	enable ModbusTCP	0_{bin}
	1_{bin}	disable ModbusTCP	
15.1	0_{bin}	enable AMS/ADS	0_{bin}
	1_{bin}	disable AMS/ADS	
16.8	0_{bin}	Ethernet Modus halbduplex	1_{bin}
	1_{bin}	Ethernet Modus vollduplex	
16.12	0_{bin}	Auto baud off	1_{bin}
	1_{bin}	Auto baud on	
16.13	0_{bin}	10 MBaud	1_{bin}
	1_{bin}	100 MBaud	
17-18	Default Gateway		0x00, 0x00, 0x00, 0x00
24	Watchdog ModbusTCP		1000 ms
26	ModbusTCP Port No.		502
27	ADS connection duration		10 s
28	Modbus connection duration		10 s
29.0	ModbusTCP mode		0_{bin}
	1_{bin}	Fast ModbusTCP	

5.5 IP Address

5.5.1 IP-address

The IP address can be set using four different procedures, and these will be described in more detail below.

Procedure	Explanation	Necessary components
KS2000	Addressing using the KS2000 configuration software and DIP switches [▶ 57]	KS2000 configuration software and KS2000 cable
ARP	Addressing via the ARP table [▶ 58]	PC with network
BootP	Addressing via BootP server [▶ 59]	BootP-Server
DHCP	Addressing via DHCP server [▶ 60]	DHCP-Server



Hex switch x10	Hex switch x1	Meaning
0x0 - D	0x0 - F	IP address direct via switch
0xF	0x0	IP address via DHCP-Server
0xF	0x1	IP address via BootP-Server
0xF	0x2	BootP save Address
0xF	0x3	Fast ModbusTCP active
0xF	0x4	Fast ModbusTCP inactive
0xF	0x5	reserved
0xF	0x6	reserved
0xF	0x7	reserved
0xF	0x8	reserved
0xF	0x9	reserved
0xF	0xA	reserved
0xF	0xB	reserved
0xF	0xC	delete AMS NetID table
0xF	0xD	set fieldbus specific default values
0xF	0xE	delete Boot project (only for IL230x-C900)
0xF	0xF	manufacturer setting

5.5.2 Setting of the address using KS2000 and switch

The setting of the IP address can be done via the KS2000 and the 2 rotary switches.

Only the last byte will be changed by the switch! All the other settings will be read directly from table 100. These values can only be changed with the KS2000 software.

Before power on, the rotary switch x10 has to have a value < 0xF.

To change the IP address it is necessary to reset the write protection in the KS2000. After the setting a Reset of the box is necessary to store and activate the changes.

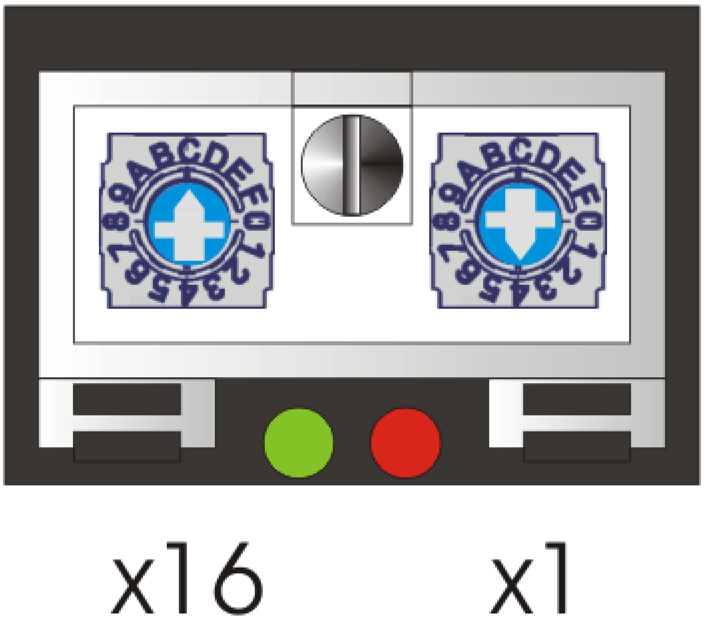
Table 9: Table 100

Register	High-Byte	Low-Byte
0	IP-Byte 2	IP-Byte 1
1	not used	IP-Byte 3

Table 10: Default

IP-Byte	Default value (hex)	Default value (dec)
1	0xAC	172_{dec}
2	0x10	16_{dec}
3	0x11	17_{dec}
4	(rotary switches)	(0 to 239_{dec})

Example:



Switch	x16	x1	
multiplicator	16	1	
this example	**Switch position**	$C_{hex} = 12_{dec}$	4_{hex}
value	12_{dec} x 16_{dec} = 192_{dec}	4_{dec}	
IP address	172.16.17.**196**		

5.5.3 Setting of the IP address via the ARP table

An easy method to change the addressing is using the DOS window. Only addresses in the same network class can be changed though! The new address will be stored and kept even when the module is powered down.

Procedure:

- switch to any IP address. DHCP or BootP shall not be activated.
- open the DOS box on your PC
- generate an entry in your ARP table using "ping >IP address<
- read the ARP tabel with "ARP -a"
- delet the module from the table with "ARP-d >IP-Address OLD<
- carry out a manula entry in the table "ARS -s >IP Address NEW< >MAC-ID [▸ 61]<"
- check with "ping -l 123 >IP-Address NEW<" if the new IP-Address is valid

Temporary flashing of the ERROR LED during the initializing phase shows, that the module is addressed via ARP. The switches have no information about the actual module addresss!

Note

When changing the IP address, all dynamic ARP entries should be deleted. Only a ping with a length of 123 Byte for changing the address is accepted (>ping -l "IP-Address<).

Example

1. C:>ping 172.16.17.255
2. C:>ARP -a
   172.16.17.255   00-01-05-00-11-22
3. C:>arp -d 172.16.17.255
4. C:>arp -s 172.16.44.44 00-01-05-00-11-22
5. C:>ping -l 123 172.16.44.44

5.5.4 Setting the IP-address via the Beckhoff BootP-server

Turn the rotary switch to 0xF1 resp. 0xF2 for using the Beckhoff BootP-server. The LED *TCP/IP ERROR* will flash during the address allocation.

Rotary switch in position 0xF2

The address assigned by the BootP server is stored, and the BootP service will not be restarted after the next cold start.

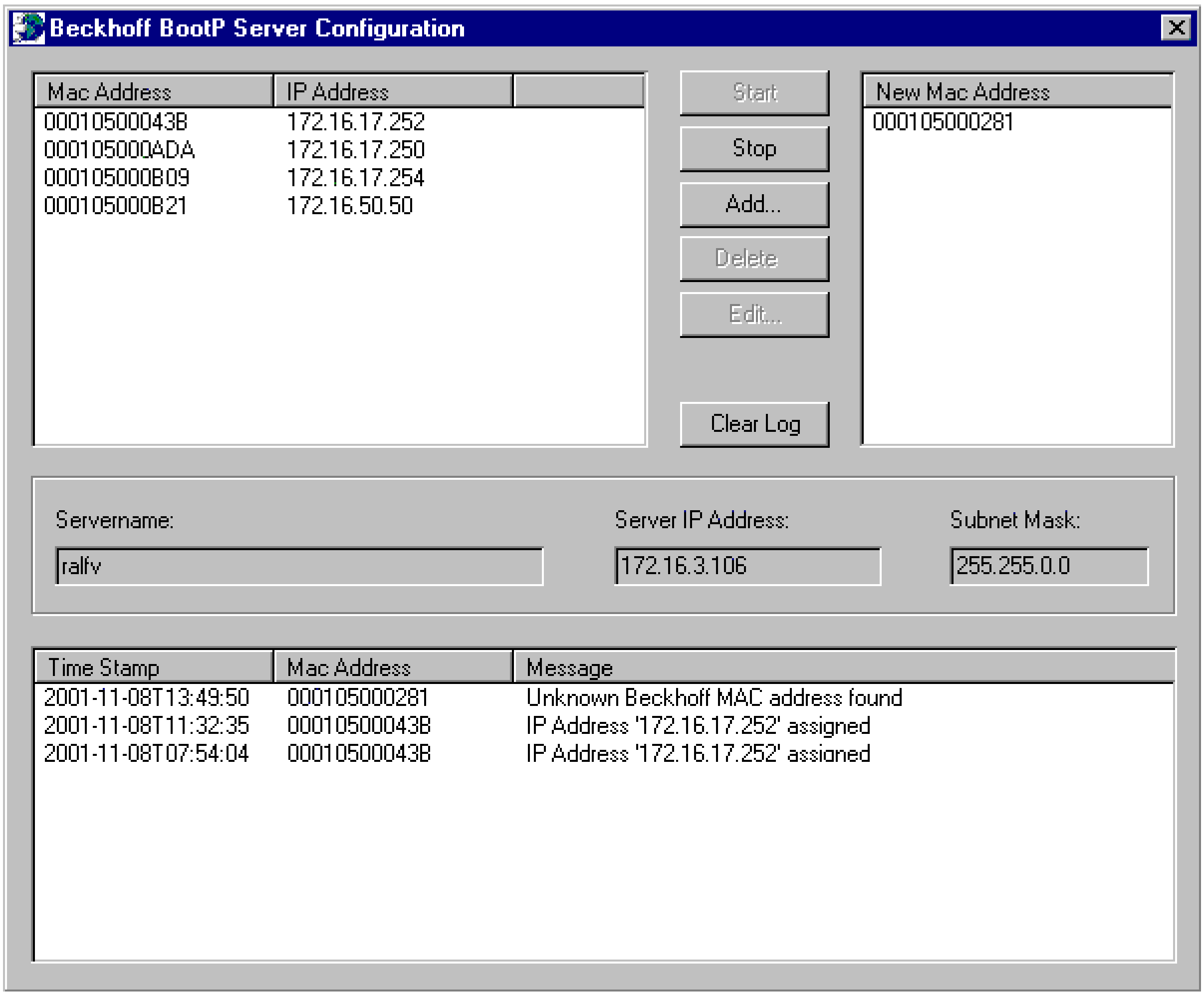
The address can be deleted by switching to manufacturer setting with the KS2000 or via the rotary switch (0xFF).

Rotary switch in position 0xF1

The IP address assigned by the BootP server is only valid until the module is switched off. At the next cold start, the BootP server must assign a new IP address to the module.
The address is, however, retained through a software reset of the module.

Beckhoff BootP server

Beckhoff supplies a BootP server for Windows 98, ME, NT4.0, NT2000 and XP.
You find the installation version on the Beckhoff TwinCAT CD in folder *Unsupported Utilities* or from http://download.beckhoff.com/download/Software/TwinCAT/TwinCAT2/Unsupported Utilities/TcBootP Server.



As soon as the BootP server has started, the *New MAC Address* window shows all the Beckhoff nodes that are working in BootP mode and still have not received an IP address. The assignment of the MAC-ID [▶ 61] to IP address is made with the "<<" button. Successful assignment is displayed in the log window.
To start the BootP server automatically when your PC boots, it is only necessary to provide a shortcut in the Windows autostart folder. Include the */Start* parameter in the shortcut (.../TcBootPDlg.exe/start).

Also see about this

- Parameterizing via Rotary Switches [▶ 53]

5.5.5 Address Configuration via DHCP Server

To set the address by means of a DHCP server, set the rotary switches to 0xF0. In this state, the DHCP service is switched on, and the module is automatically assigned an IP number by the DHCP server. For this purpose the DHCP server must know the module's MAC-ID [▶ 61]. The IP address should be set statically. The *TCP/IP Error* LED flashes while the address is being allocated.

5.5.6 Subnet mask

The subnet mask is subject to the control of the network administrator, and specifies the structure of the subnet.

Small networks without a router do not require a subnet mask. The same is true if you do not use registered IP numbers. A subnet mask can be used to subdivide the network with the aid of the mask instead of using a large number of network numbers.

The subnet mask is a 32-bit number.

- Ones in the mask indicate the subnet part of an address region.
- Zeros indicate that part of the address region which is available for the host IDs.

Description	Binary representation	Decimal representation
IP address	10101100.00010000.00010001.11 001000	172.16.17.200
Subnet mask	11111111.11111111.00010100.00 000000	255.255.20.0
Network ID	10101100.00010000.00010000.00 000000	172.16.16.0
Host ID	00000000.00000000.00000001.11 001000	0.0.1.200

Standard subnet mask

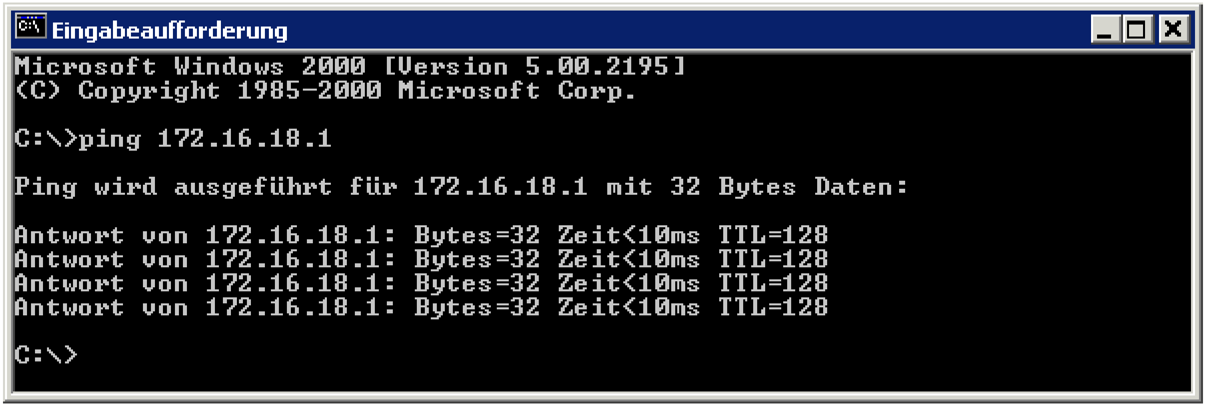
Address class	Standard subnet mask (decimal)	Standard subnet mask (hex)
A	255.0.0.0	FF.00.00.00
B	255.255.0.0	FF.FF.00.00
C	255.255.255.0	FF.FF.FF.00

Note

Neither subnet 0 nor the subnet consisting only of ones may be used. Neither host number 0 nor the host number consisting only of ones may be used! If the IP address is set using the KS2000 configuration software, it is necessary for the subnet mask also to be changed with the KS2000 configuration software. If ARP addressing is used, the associated standard subnet mask, based on the IP address, is entered. Under BootP or DHCP the subnet mask is entered by the server.

5.5.7 Testing the IP Address

Use the *Ping* command to test the IP address.



```
Eingabeaufforderung
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\>ping 172.16.18.1

Ping wird ausgeführt für 172.16.18.1 mit 32 Bytes Daten:

Antwort von 172.16.18.1: Bytes=32 Zeit<10ms TTL=128
Antwort von 172.16.18.1: Bytes=32 Zeit<10ms TTL=128
Antwort von 172.16.18.1: Bytes=32 Zeit<10ms TTL=128
Antwort von 172.16.18.1: Bytes=32 Zeit<10ms TTL=128

C:\>
```

5.5.8 Reading the MAC-ID

Proceed as follows to read the MAC-ID:

- Change the IP address of your PC to 172.16.x.x and the Subnet mask to 255.255.0.0
  The default IP address of the Ethernet Fieldbus Boxes is 172.16.18.1 (rotary switch setting: 0, 1).
- Start the DOS Window
- Send a Ping to IP address 172.16.17.1
- Read the MAC-ID with *arp -a*.

5.5.9 Network Classes

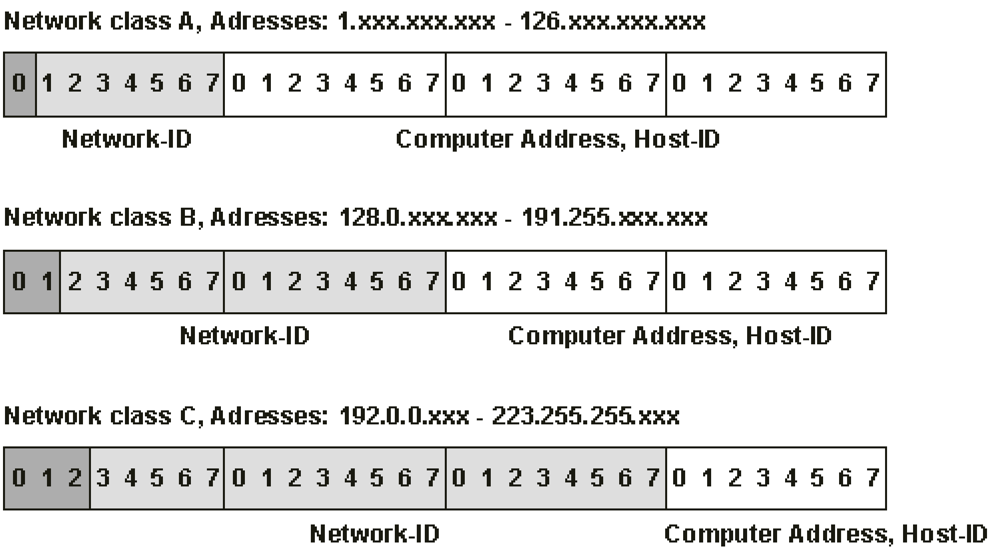
Three different network classes are distinguished. They specify how many address bits are reserved for the Network -ID and how many for the computer number (or node number). The difference is located in the first 3 bits of the IP address.

Network-class	Number of bits for the Network ID	Possible number of networks	Number of bits for the node address	Possible number of nodes per net-work
A	7	126	24	16 777 214
B	14	16 382	16	65 536
C	21	2 097 150	8	254

NOTE

Warning

An IP address must be unique within the entire connected network!



Note

In a communication with another Ethernet devices, the IP address set must have the same network class. Example: Your PC has address 172.16.17.55, which means that the Bus Coupler must have address 172.16.xxx.xxx (each xxx stands for a number between 0...255. The 0 is normally used for routers/switches, and should therefore be reserved).

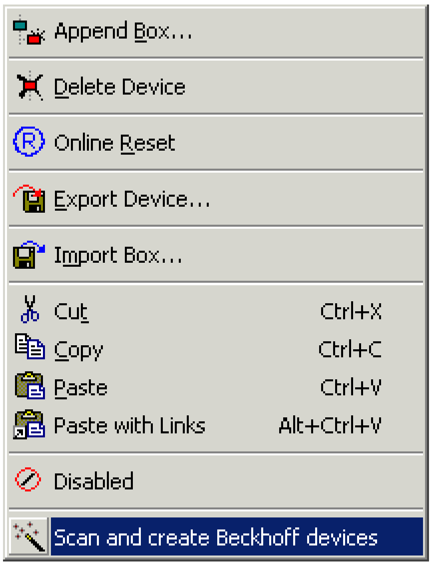
In order to see the PC's own address, the command *ipconfig* can be entered into a DOS window under Windows NT/2000/XP.

6 Configuration

6.1 TwinCAT System Manager

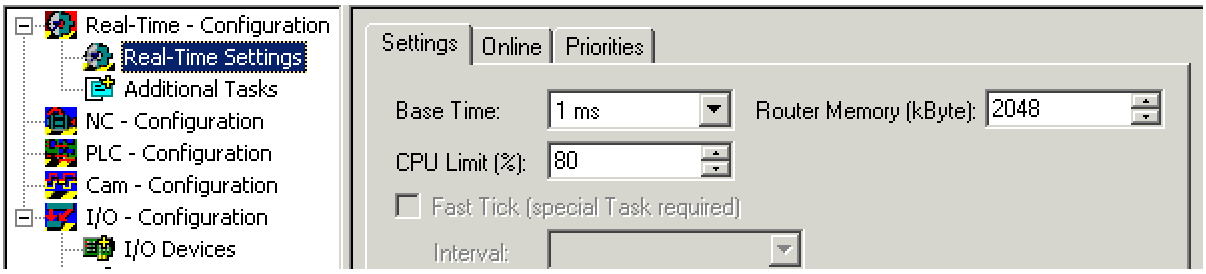
6.1.1 Configuration using the System Manager

Enter a general Ethernet card in the TwinCAT System Manager under devices. If the Bus Couplers are already connected to the network and have IP addresses, you can also read these. This will cause all the Bus Couplers with Bus Terminals and their configurations to be loaded. You can then adapt these to meet your requirements.



Commissioning with 6 or more Ethernet nodes

If you enter more than five nodes in the System Manager, you will have to increase the router memory. You will find this setting in the System Manager under *Real Time Settings*. Enter 350 kbyte there for each Bus Coupler. This will give, for instance, 3.5 MB (10 x 350 kbyte) for 10 Bus Couplers.

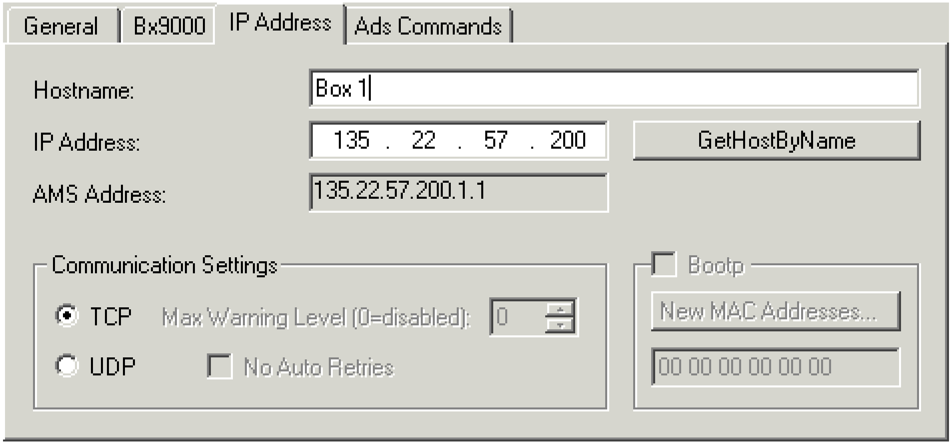


Note: The router memory depends on your PC's working memory, and there is a limit to how high it can be set.
Calculation of the maximum possible router memory:
Max. router memory [MB] = (1 MB +((RAM [MB] - 4 MB) x 0.4)

Example:
Suppose your PC has 128 MB of RAM:
1MB +(128MB - 4MB) x 0,4 = 50,6 MB
So the maximum size of the router memory must not exceed 50.6 MB if there is 128 MB of RAM.

6.1.2 IP Address Tab

Ethernet components require an unambiguous IP address in the network. The setting dialog necessary for configuration of the fieldbus nodes in the TwinCAT system is described below. The tab illustrated appears after you have selected a BK9000, BC9000 or other Ethernet fieldbus device with the right mouse button (Add box) under your Ethernet cards in the System Manager.



Host name
The name of the Bus Coupler station can be edited here.

IP address
Enter the IP address of the Bus Coupler here.

GetHostByName
By clicking this button you can (as from BK9000 firmware version *B2*) obtain an IP address from a Windows 2000 DHCP (Dynamic Host Configuration Protocol) server.

AMS address
Reports the Bus Coupler's AMS Net ID. It is automatically generated from the IP address by adding two additional bytes (e.g. ".1.1").

BootP

This checkbox can be activated if the Beckhoff BootP server has been installed and started or (as from TwinCAT Version 2.8).

New MAC Addresses
If the Beckhoff BootP server has been started, the MAC addresses (Media Access Controller) of Ethernet fieldbus devices that have newly been connected can be displayed by clicking this button. If only one new Ethernet fieldbus device has been connected, then only one new MAC address will accordingly be displayed. You can then assign it to the desired IP address in the *IP address field* described above. Each time another Ethernet fieldbus component is added you can repeat this process on the tab described here for the new device.

Communication Settings

Settings for IP (Internet Protocol) communication.

TCP
If this option is active (which is the default) the communication is handled by TCP (*Transmission Control Protocol*). This means that telegram packets to and from Bus Couplers are exchanged using a reliable (acknowledged) mechanism.

UDP
If this option is selected, communication is handled by UDP (*User Datagram Protocol*). In contrast to TCP, the individual telegram packets are not exchanged using a reliable mechanism. They are, in other words, not individually acknowledged by the receiver after they have arrived. Telegram packets that are damaged or

whose sequence has become disturbed are neither re-sent nor sorted. The advantage of UDP is that the telegram transfer times can be calculated, because it is not necessary to wait for an answer from the receiver. It is therefore possible to talk of UDP in terms of a limited real-time capability.

Max. Warning Level
This setting option is only active with *UDP*. The maximum value of the error counter can be entered here. Waiting for response telegrams from the fieldbus node stops when the set maximum value has been reached. Instead, read telegrams only, based on a higher cycle time, are sent to the fieldbus nodes.

No Auto Retries
This checkbox also can only be selected when the *UDP* option is active. If this checkbox is active, then when the level set under *Max. Warning Level* has been reached it will be necessary to execute the *IO / Reset* function.

6.2 Process Image

6.2.1 ModbusTCP Process Image

The ModbusTCP process image makes a fundamental distinction between digital and byte-oriented (word-oriented) signals (Bus Terminals). Additionally, a distinction is made between inputs and outputs. The mapping table shows which Bus Terminals belong to which mapping.

At word-oriented access, first the analog and then the digital modules are listed in the address room.

Table 11: Byte-oriented access

Digital inputs	Digital outputs
Offset 0 ... 4095 Modbus function Reading 2 Writing: -	Offset 0 ... 4095 Modbus function Reading 1 Writing5, 15

Table 12: Word-oriented access

Analog/digital inputs	Analog/digital outputs
Offset 0x000 ... 0x0FF Modbus function Reading 3, 4, 23 Writing: -	Offset 0x800 ... 0x8FF Modbus function Reading 3, 23 Writing6, 16, 23

This will be clarified by example.

Example

IL2301-B900 (Remark: The IL2301-B900 has got 4 in- and 4 outputs on the module)
1 x IE1001
1 x IE2001

Inputs			Outputs		
Modbus function	**Modbus Address/Offset**	**Fieldbus Box Module**	**Modbus function**	**Modbus Address/Offset**	**Fieldbus Box Module**
3 read	0x0000	4 Bit IL2301	3 read	0x0800	4 Bit IL2301
4 read	0x0000	8 Bit IE1001	1 read	0x0000	8 Bit IE2001
2 read	0x0000 - 0x0001		5 write	0x0000-0x000B	
			15 write	0x0000-0x000B	
			6 write	0x800	
			16 write	0x800	
			23 write	0x800	

6.2.2 ADS Process Image

The process image is sub-divided into input image and output image. The Coupler Box first maps all complex (byte-oriented) Extension Boxes (IE3xxx, IE4xxx, IE5xxx, IE6xxx), into the process image in the order in which they are inserted in IP-Link. Afterwards all digital Extension Boxes (IE1xxx, IE2xxx) are added to the process image.

The complex (byte-oriented) Extension Boxes are always mapped with the following settings:

- complete evaluation
- Word alignment
- Intel format

Example 1:

IL2301-B900 (4DI, 4DO)
2 x IE1001 (8DI)
2 x IE2002 (8DO)
1 x IE3312 (4 AI)
1 x IE4132 (4AO)

Position	Module	Input %I..Byte Offset	Output %Q..Byte Offset
1	IE3312	0...7	0..7
2	IE4132	8...15	8...15
3	IL2301	16.0...16.3	16.0...16.3
5	IE1001	16.4...17.3	-
6	IE2002	-	16.4...17.3

Note

Detailed description about the mappings of the Fieldbus Box Modules can be found in the document Signal Types, Installation, Configuration of the I/O modules, that may be found on the BECKHOFF CD Products & Solutions or in the internet (http://www.beckhoff.com) under Download/Field-bus Box.

6.3 Configuration of the complex modules

6.3.1 Register Communication

6.3.1.1 General Register Description

Different operating modes or functionalities may be set for the complex modules. The *General Description of Registers* explains those register contents that are the same for all complex modules. The module-specific registers are explained in the following section.
Access to the module's internal registers is described in the section on *Register Communication*.

General Description of Registers

Complex modules that possess a processor are able to exchange data bi-directionally with the higher-level controller. These modules are referred to below as intelligent modules. These include the analog inputs (0-10 V, -10-10 V, 0-20 mA, 4-20 mA), the analog outputs (0-10 V, -10-10 V, 0-20 mA, 4-20 mA), the serial interface terminals (RS485, RS232, TTY, data exchange terminals), counter terminals, encoder interface and SSI interface terminals, PWM terminals and all the modules that can be parameterized.

The main features of the internal data structure are the same for all the intelligent modules. This data area is organized as words, and includes 64 memory locations. The important data and the parameters of the module can be read and set through this structure. It is also possible for functions to be called by means of corresponding parameters. Each logical channel in an intelligent module has such a structure (so a 4-channel analog module has 4 sets of registers).

This structure is divided into the following areas:

Range	Address
Process variables	0-7
Type register	8-15
Manufacturer parameters	16-30
User parameters	31-47
Extended user region	48-63

Registers R0-R7 (in the terminal's internal RAM)

The process variables can be used in addition to the actual process image. Their function is specific to the terminal.

R0-R5

The function of these registers depends on the type of terminal.

R6

Diagnostic register. The diagnostic register can contain additional diagnostic information. Parity errors, for instance, that occur in serial interface terminals during data transmission are indicated here.

R7

Command register
- High-Byte_Write = function parameter
- Low-Byte_Write = function number
- High-Byte_Read = function result
- Low-Byte_Read = function number

Registers R8-R15 (in the terminal's internal ROM)

The type and system parameters are hard programmed by the manufacturer, and the user can read them but cannot change them.

R8

Fieldbus Box type: The Fieldbus Box type in register R8 is needed to identify the Fieldbus Box.

R9

Software version x.y.: The software version can be read as a string of ASCII characters.

R10

Data length: R10 contains the number of multiplexed shift registers and their length in bits. The Bus Coupler sees this structure.

R11

Signal channels: Related to R10, this contains the number of channels that are logically present. Thus for example a shift register that is physically present can perfectly well consist of several signal channels.

R12

Minimum data length: The particular byte contains the minimum data length for a channel that is to be transferred. If the MSB is set, the control/status byte is not absolutely necessary for the terminal's function, and if the Bus Coupler is appropriately configured it is not transferred to the controller. The information is located
- in the high byte of an output module
- in the low byte of an input module

R13

Data type register

Data type register	Description
0x00	Terminal with no valid data type
0x01	Byte array
0x02	Structure 1 byte n bytes
0x03	Word array
0x04	Structure 1 byte n words
0x05	Double word array
0x06	Structure 1 byte n double words
0x07	Structure 1 byte 1 word
0x08	Structure 1 byte 1 double word
0x11	Byte array with variable logical channel length
0x12	Structure 1 byte n bytes with variable logical channel length (e.g. 60xx)
0x13	Word array with variable logical channel length
0x14	Structure 1 byte n words with variable logical channel length
0x15	Double word array with variable logical channel length
0x16	Structure 1 byte n double words with variable logical channel length

R14

reserved

R15

Alignment bits (RAM): The analog terminal is placed on a byte boundary in the K-Bus with the alignment bits.

Registers R16-R30 (manufacturer's parameters, serial EEPROM)

The manufacturer parameters are specific for each type of terminal. They are programmed by the manufacturer, but can also be modified by the controller. The manufacturer parameters are stored in a serial EEPROM in the terminal, and are retained in the event of voltage drop-out. These registers can only be altered after a code-word has been set in R31.

Registers R31-R47 (application parameters, serial EEPROM)

The application parameters are specific for each type of terminal. They can be modified by the programmer. The application parameters are stored in a serial EEPROM in the terminal, and are retained in the event of voltage drop-out. The application region is write-protected by a code-word.

R31

Code-word register in RAM: The code-word 0x1235 must be entered here so that parameters in the user area can be modified. If any other value is entered into this register, the write-protection is active. If write protection is inactive, the code-word is returned when the register is read, but if write protection is active, then the register contains a null value.

R32

Feature register: This register specifies the terminal's operating modes. Thus, for instance, a user-specific scaling can be activated for the analog I/O modules.

R33-R47

Terminal-specific Registers: These registers depend on the type of terminal.

Registers R47-R63 (Register extension for additional functions)

These registers are provided for additional functions.

6.3.1.2 Example for Register Communication

Control Byte

The Control Byte is located in the output image, and can be read or written.

Bit	7	6	5	4	3	2	1	0
Name	REG	R/W	register number					

Bit	Name	Description	
7	REG	1_{bin}	Register Communication switched on: The first two Data Bytes are not used for process data exchange, but are written to the register set of the Fieldbus Box or are read from there.
6	R/W	0_{bin}	Read: the register should be read without changing it.
		1_{bin}	Write: the register should be written.
5-0	register number	Number of the Registers, that should be read or written. 64 Registers are addressable.	

Status Byte

The Status-Byte is located in the output image und can only be read.

Bit	7	6	5	4	3	2	1	0
Name	REG	R	register number					

Bit	Name	Description	
7	REG	1_{bin}	receipt register number
6	R	0_{bin}	Read
5-0	register number	Number of the Registers, that was read or written.	

Example 1

Table 13: Reading register 8 of KL3204 or IP/IE3202

Byte 0 (control byte)	Byte 1 (data out, high byte)	Byte 2 (data out, low byte)
0x88 (1000 1000_{bin})	0xXX	0xXX

Bit 0.7 set indicates register communication active
Bit 0.6 not set indicates reading the register.
Bit 0.5 to Bit 0.0 indicates with 001000_{bin} the register number 8.
The output data word (Byte 1 and Byte 2) has no function at the reading access. If you want to change a register, you have to write the desired value into the output data word.

Byte 0 (status byte)	Byte 1 (data in, high byte)	Byte 2 (data in, low byte)
0x88	0x0C	0x84

The terminal/box returns the type name 0x0C84 (equivalent unsigned integer 3204) in the input data word (Byte 1 and Byte 2).

Special feature in the naming of Fieldbus Boxes:
The last figure of the delivered unsigned Integer (3204) is not the same like the last character of the Fieldbus Box name (3202), witch stands for the connector type (0 for S8, 1 for M8 and 2 for M12). It returns instead of that the number of channels (IE3204 owns 4 channels).

Note

In order to write into registers, you have to write the password (0x1235) into register 31, so that write protection is deactivated. It is activated again by writing any value other than 0x1235. Note that some of the settings that can be made in registers only become active after the next power restart (power-off/power-on) of the module.

Example 2

Process of register communication for writing into register.

Table 14: 1. Write register 31 (set code word)

Byte 0 (control byte)	Byte 1 (data out, high byte)	Byte 2 (data out, low byte)
0xDF	0x12	0x35

Answer of the module/terminal

Byte 0 (status byte)	Byte 1 (data in, high byte)	Byte 2 (data in, low byte)
0x9F	0xXX	0xXX

Table 15: 2. Read register 31 (verify, if code word is set)

Byte 0 (control byte)	Byte 1 (data out, high byte)	Byte 2 (data out, low byte)
0x9F	0xXX	0xXX

Answer of the module/terminal

Byte 0 (status byte)	Byte 1 (data in, high byte)	Byte 2 (data in, low byte)
0x9F	0x12	0x35

Table 16: 3. Write Register 32 (change register)

Byte 0 (control byte)	Byte 1 (data out, high byte)	Byte 2 (data out, low byte)
0xE0	0x00	0x02

Answer of the module/terminal

Byte 0 (status byte)	Byte 1 (data in, high byte)	Byte 2 (data in, low byte)
0xA0	0xXX	0xXX

Table 17: 4. Read Register 32 (verify changed register)

Byte 0 (control byte)	Byte 1 (data out, high byte)	Byte 2 (data out, low byte)
0xA0	0xXX	0xXX

Answer of the module/terminal

Byte 0 (status byte)	Byte 1 (data in, high byte)	Byte 2 (data in, low byte)
0xA0	0x00	0x02

Table 18: 5. Write Register 31 (set code word back)

Byte 0 (control byte)	Byte 1 (data out, high byte)	Byte 2 (data out, low byte)
0xDF	0x00	0x00

Answer of the module/terminal

Byte 0 (status byte)	Byte 1 (data in, high byte)	Byte 2 (data in, low byte)
0x9F	0xXX	0xXX

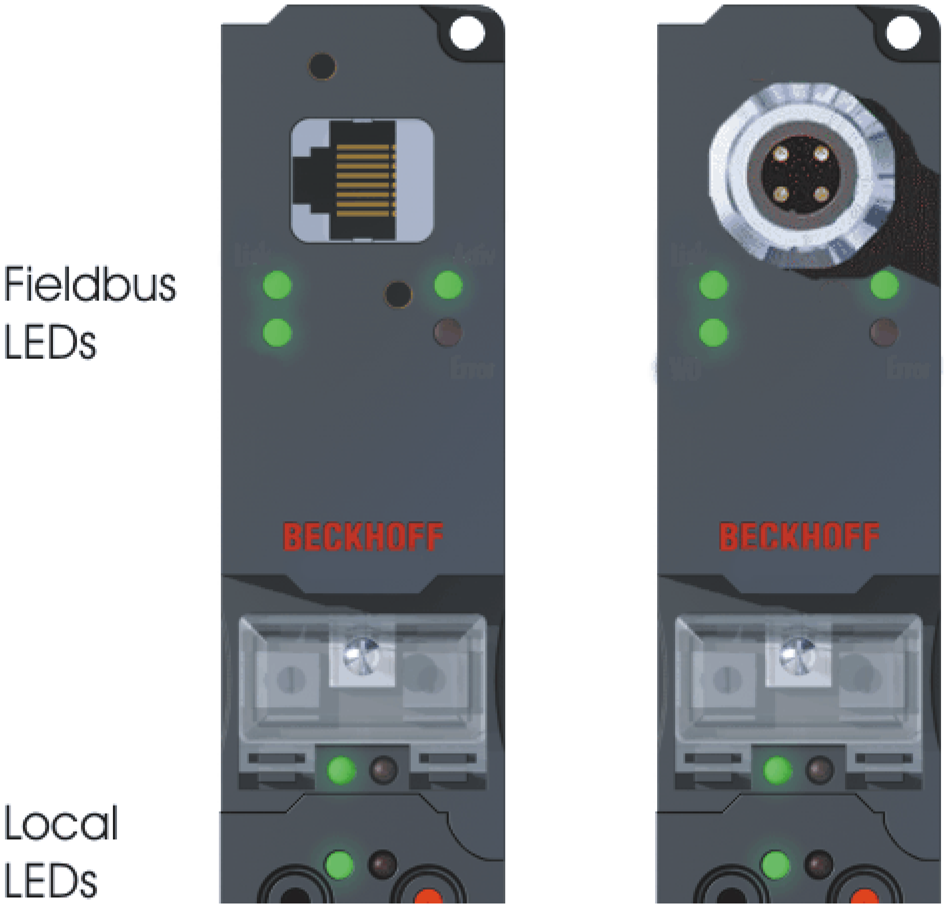
7 Error Handling and Diagnosis

7.1 Diagnostic LEDs - Overview

Error diagnosis

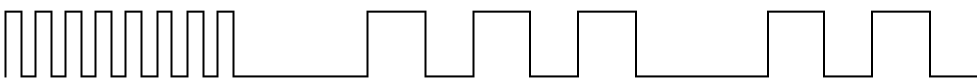
There are 2 sorts of errors:

- Fieldbus Errors [▸ 75]
- Local Errors [▸ 77] on Compact Box or Coupler Box



Blink Codes

Blink sequence	Meaning
Fast flashing	Beginning of the Blink Code
first slow sequence	Error code
second slow sequence	Error argument
third slow sequence (optional)	Error argument



Beginning Error code Error argument

7.2 Diagnostic LEDs for Ethernet

After switching on, the Fieldbus Box immediately checks the connected configuration. Error-free start-up is indicated when the red *I/O ERR* LED goes out. If the *I/O ERR* LED blinks, an error in the area of the Inputs/ Outputs is indicated. The error code can be determined from the frequency and number of blinks. This permits rapid rectification of the error.

The module has two groups of LEDs for the display of status.
The upper group with four LEDs indicates the status of the respective fieldbus. The significance of the fieldbus status LEDs is explained in the appropriate sections of this manual. It corresponds to the usual fieldbus display.

At the lower end of the Module are two more green LEDs that indicate the supply voltage. The left hand LED indicates the presence of the 24 V_{DC} supply for the Fieldbus Box. The right hand LED indicates the presence of the supply to the power contacts.

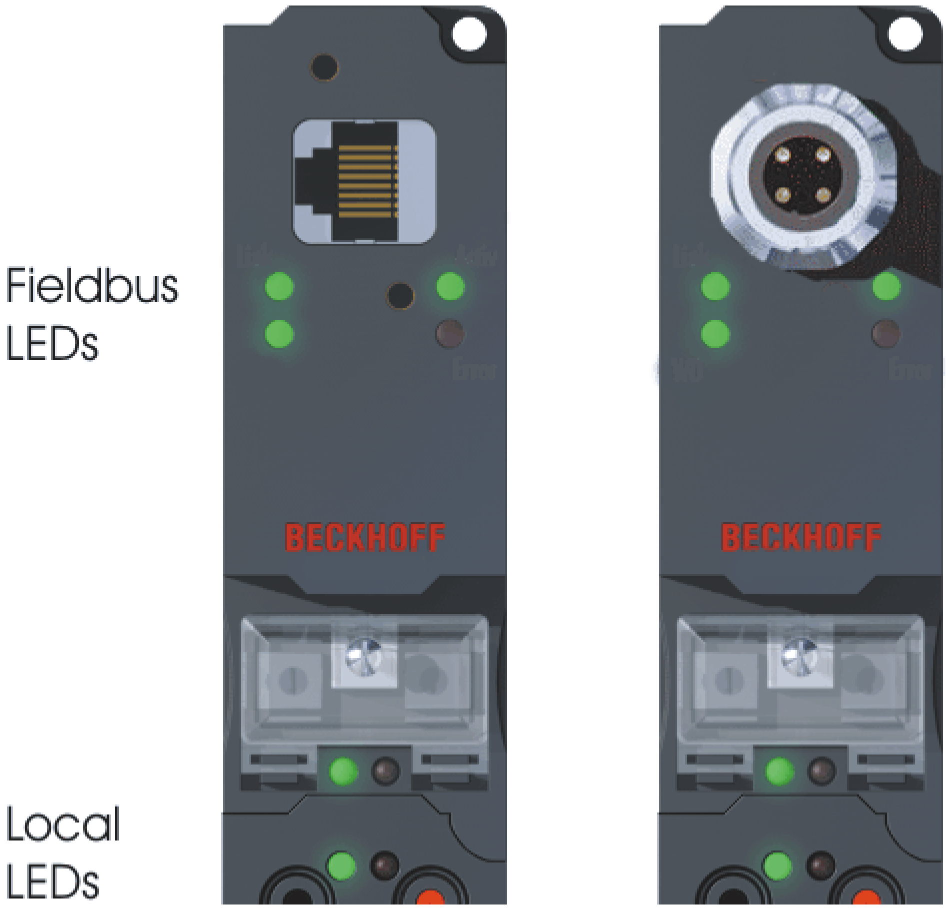


Fig. 1: B90x_DiaLED

LEDs for power supply diagnosis

LED	In	Out
LINK	Physical connection present	No physical connection present
ACTIV	Flashing: Bus traffic present	No bus traffic (bus idle)
ERROR	The LED flashes slowly if DHCP or BootP is active but the Bus Coupler has not yet received an IP address The LED flashes rapidly (5 times, only when switching on); the Bus Coupler is addressed with ARP. The settings on the rotary switches are not valid.	No error.
	ON, when the box receives more than one Ethernet frame per module cycle. The master sends to fast!	
WD	Out - Watchdog error or no communication	- Start communication - reset WD error
	On - Watchdog works	No error

Fig. 2: FBB_power_LED

LEDs for power supply diagnosis

LED	Meaning
Left LED off	Module has no power
Left LED red	Short circuit detection for sensor supply (< 500mA) is active. Sensors/Inputs are not supplied anymore
Right LED off	No 24 V_{DC} power for the outputs connected

7.3 Diagnostic LEDs for local errors

Local error in a Coupler Box (IL230x-Bxxx/Cxxx)

The term *local error* means that an error has occurred in the Fieldbus Box or the IP-Link. IP-Link errors most often turn out to be a result of inappropriate use of the optical fiber.

LED green	LED red			Description	Remedy
off	off			No data exchange	Module in synchronous mode or - activate PROFIBUS cyclic data
off	1	0		EEPROM checksum error	Set manufacturer’s setting with the KS2000 software
off	2			Reserved	-
off	3			Break location has been recognized	interruption before the master's receiver
	3	n		Break location has been recognized	n-th module before the master's receiver
	3	n	m	Break location has been recognized	(n*10)+m-th module before the master's receiver
off	4	n		Too many faulty telegrams have been detected (more than 25%)	The optical fiber wiring in front of the nth extension module should be checked
off	5	n		Register access to complex modules has failed	Check the nth module
off	11	n		Complex module working incorrectly	Exchange the nth module
off	12	n		More than 120 modules in the ring	Connect fewer modules
off	13	n		nth module unknown	Firmware update required
on	off			Module is exchanging data	no error

Local errors in an Extension Box

LED green	LED red	Description
off	on	No data is being received over the IP-Link
off	blinks, flickers	Faulty IP-Link protocols are being received (very poor data connection)
blinks, flickers	blinks, flickers	Faulty IP-Link protocols are being received (poor data connection), does not necessarily lead to an error
on	off	IP-Link protocols are being received, no error

Faulty protocols can occur, because of:

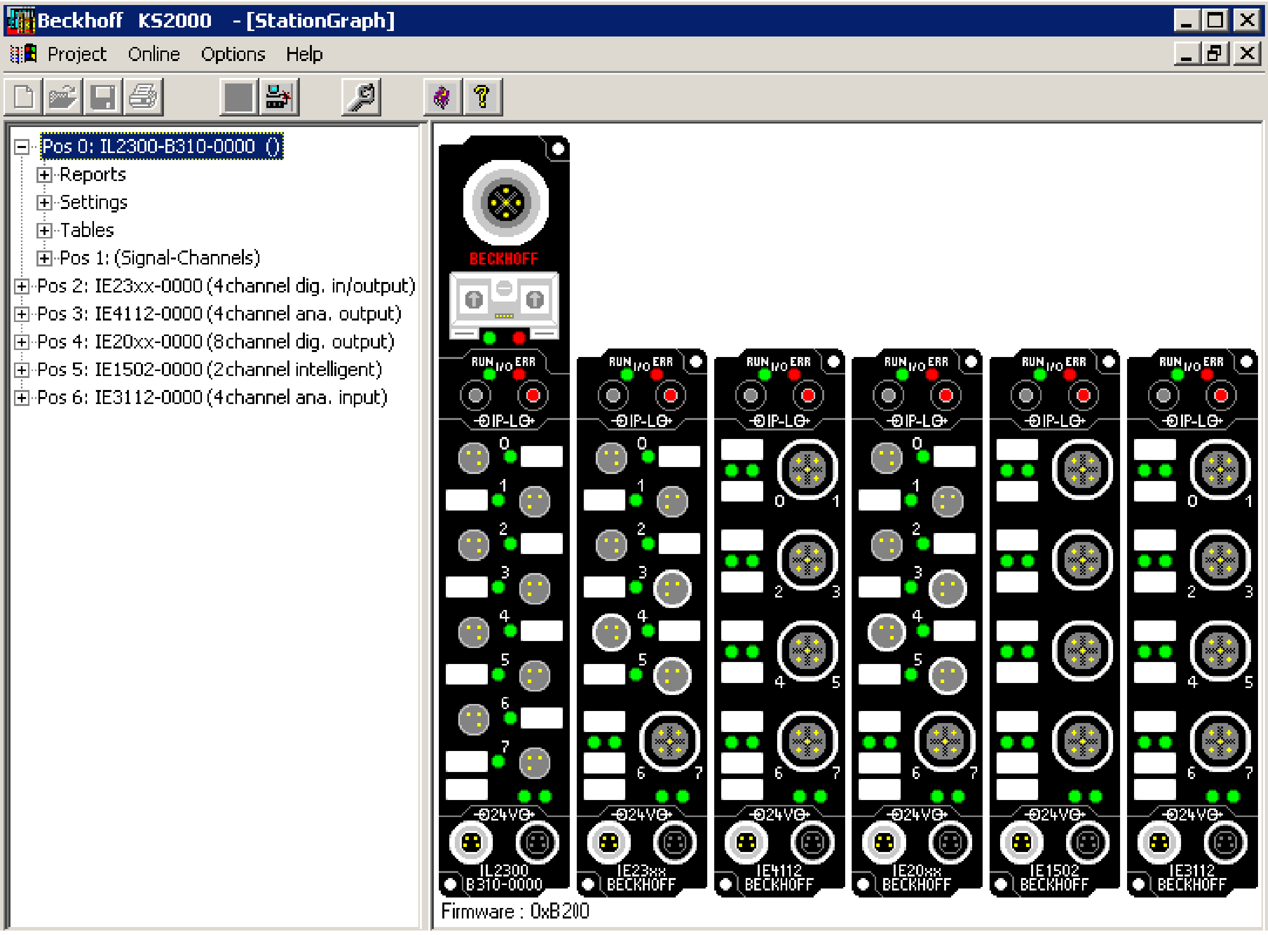
- bad configured IP-Link connectors
- IP-Link cable with higher dampening, e.g. because of a sharp curve
- contaminated sender LED (module before the faulty one)
- contaminated receiver

The internal IP-Link error counter [▸ 79] of the Coupler Box can be read with the KS2000 software.

7.4 Check of the IP-Link connection

A correct assembled IP-Link cable will assure an error free transmission.

An additional testing of the transmission quality and error diagnostics is possible with the KS2000 configuration software.



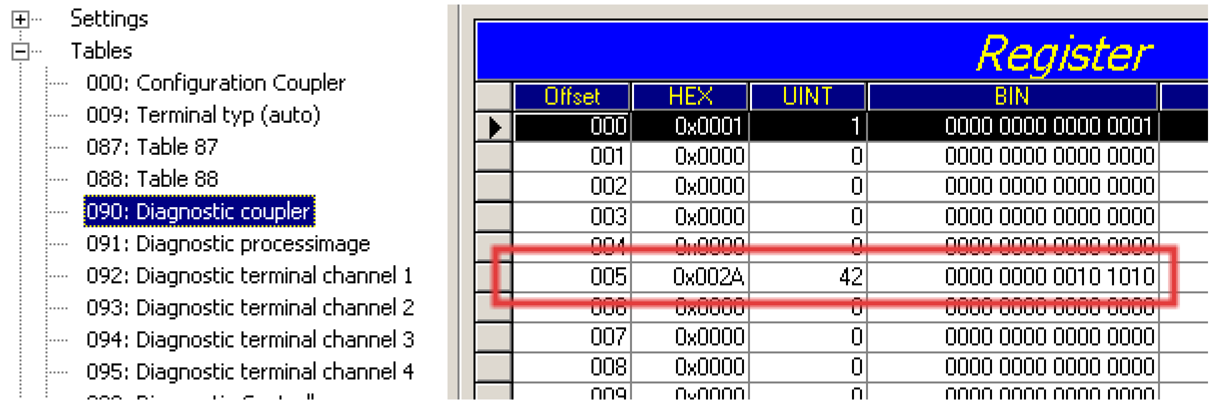
For this test, the fieldbus master (e.g. a PROFIBUS PC Card) should be on the bus and it should transmit data cyclical. Another way to generate cyclic data is, to switch the coupler to *free running* via the KS2000 software.

The result should be, that the I/O RUN LED flashes in a bright green. This shows, that a data exchange with the connected extension boxes takes place. A red blinking I/O ERR LED shows faulty IP-Link telegrams. These faulty telegrams will be repeated automatically like in any other fieldbus system. This way a transmission of the data is guaranteed.

Error counter

Table 90, offset 005 shows possible IP-Link errors. Sporadic appearing errors do not mean any problem for the communication, as long as they do not reach a critical limit.

This error counter is only reset by the Power ON/OFF.



If lots of errors occur in a very short time, this will be interpreted as a heavy disturbance of the communication and the coupler box will report this error. This can be seen at offset 006 and 007. Both values will show a value > 200 and the I/O ERR LEDs of the coupler box will blink the according error code.

Note

The KS2000 Configuration Software communicates with the Coupler Box via the serial channel. The content of the registers will not be refreshed automatically.

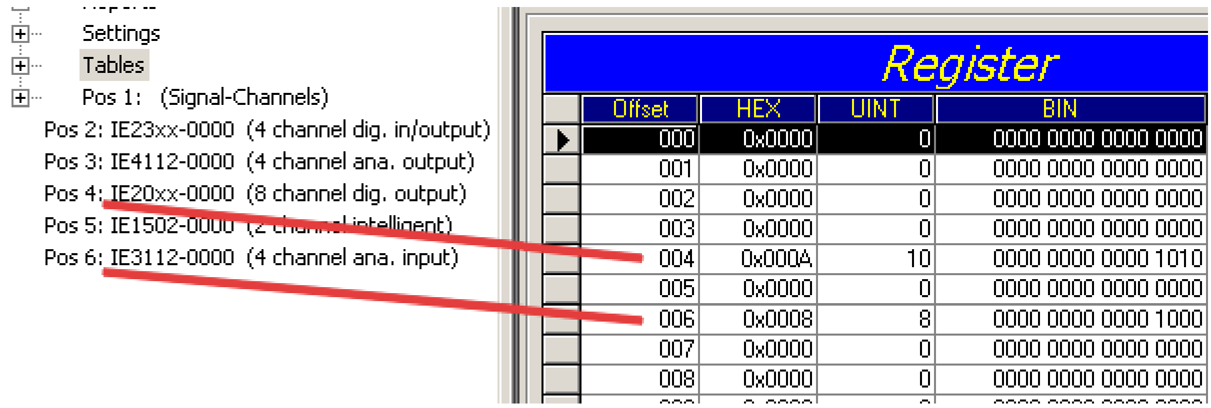
Position of the error

In case of an IP-Link error, the Coupler Box tries to read the error location from the register of the Extension Box. If the fiber optic ring is interrupted or the communication is heavily disturbed, this is not possible. Only the position of the last functioning Extension Box before the receiver of the Coupler Box can be recognized. The box will then flash this error code via the I/O ERR LED.

If the communication via IP-Link is still running, table 87 shows the error counter of each Extension Box.

The offset register corresponds to the position of the Extension Box in the KS2000 tree (left side of graphic). This example shows errors at offset 004 and 006.

In the "real" world the faulty IP-Link telegram was reported from the IE20xx and the IE3112, that means the problem has to looked for before these modules.



The error can be up to:

- the sending module
- the receiving module
- the IP-Link cable
- the connectors

If there is an error in table 90 and none in table 87, the faulty transmission is between the last Extension Box and the Coupler Box.

In most cases the transmission errors can be traced back to bad configured IP-Link connectors or a too high attenuation of the cable due to sharp bending.

The values of table 87 directly come from the extension boxes. In case of an IP-Link interruption these values will be set to zero and only table 90 can be used.

Note

If you want to operate a Coupler Box (e.g. IL2300-Bxxx, IL2301-Bxxx or IL2302-Bxxx) totally without Extension Box Modules (IExxxx), you have to connect the send and receive socket of this Coupler Box directly by using an IP Link Cable! For this the IP Link Jumper ZK1020-0101-1000 fits perfect.

7.5 General Errors

No data exchange after replacing a bus coupler

You have exchanged the Ethernet Bus Coupler and set the same TCP/IP number, but data is not exchanged.

Every Ethernet device has its own, unique MAC-ID. This number is saved when connecting to an Ethernet node, and stored in a table. This table contains the correspondences between the MAC-ID and the TCP/IP number. You must delete this table. Do this in a DOS window, by entering the command "arp -d" and the TCP/IP number of the Bus Coupler.
Example: >arp -d 172.16.17.203<
If the DHCP protocol or the BootP protocol is active it is necessary to set the MAC-ID of the new coupler in the DHCP server or BootP server after changing the Bus Coupler.

Communication errors when online *

After a period in the online state (logged in via Ethernet/AMS) the message *Communication error - logging out* always occurs.

The data traffic through the Ethernet interface is jamming. Remedy:

- Reduce the level of data communication.
- Stop the cyclical data traffic, or lengthen the task time.
- Reduce the number of windows open in the online display.
- Log in via the serial interface.

* only BC9000

7.6 ADS Diagnostic

Status inputs

It is possible to monitor the BK/BC9000 Bus Coupler's communication in the system manager. Each Bus Coupler has status inputs that can be found in the hardware tree.

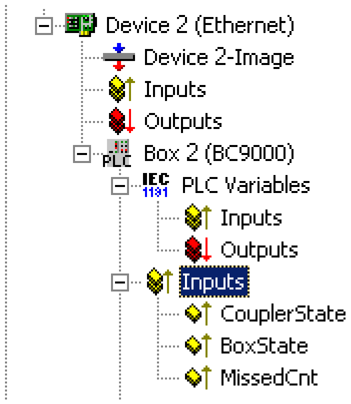
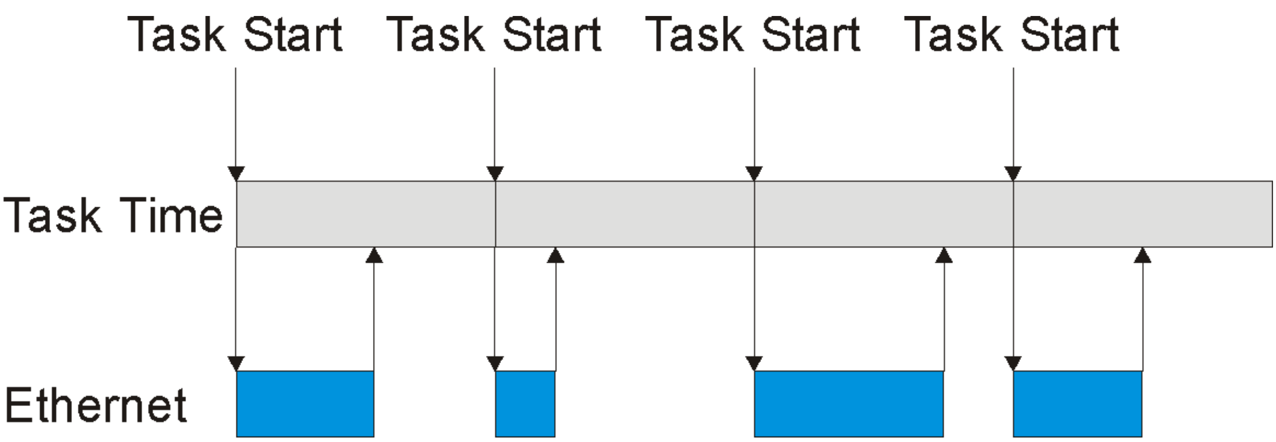


Table 19: Coupler state

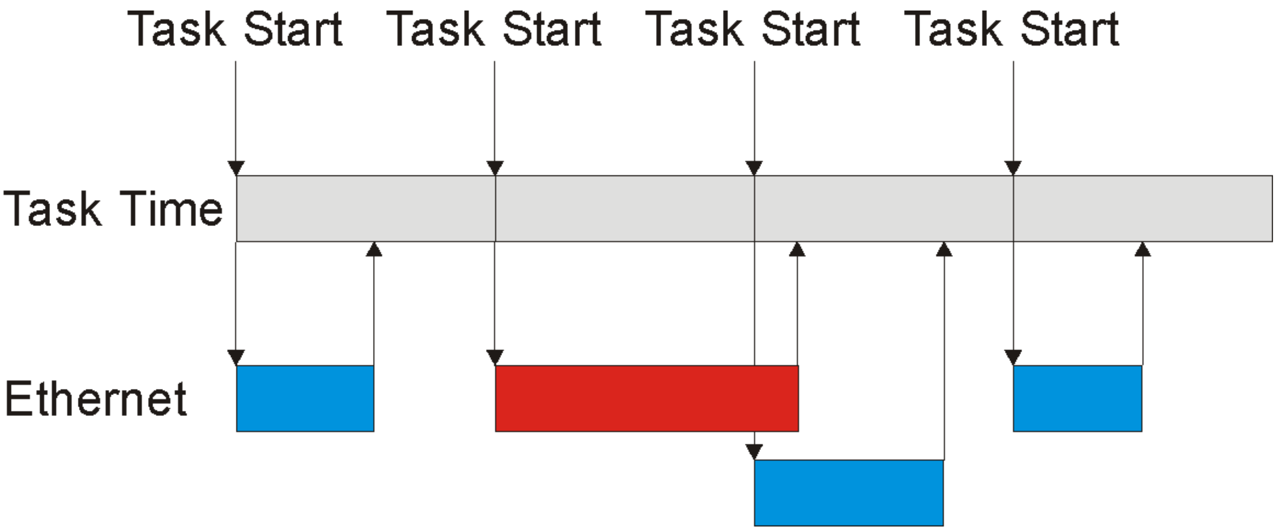
Value	Meaning
0x0000	No error
0x0001	Bus Terminal error, there is a K-Bus error
0x0002	Configuration error; the parameterized configuration does not match the actual configuration

MissedCnt

Ideal configuration:
The task time is always longer than the Ethernet transmission time. An Ethernet telegram is transmitted at the beginning of the task, and it returns it to the PC again after a period of time, t_{Eth}. If the time t_{Eth} is always smaller than the task time that has been set, the value in the *MissedCnt* counter remains constant.



If, after the task time has elapsed, an Ethernet telegram has still not arrived at the PC, only reaching it after the next task has started, then TwinCAT will continue to work with the old input data. In addition, the *MissedCnt* counter is incremented.



This can have the following causes:

- The chosen task time is too short. Set
  - 100 ms or more for TCP
  - 20 ms or more for UDP
- Too many collisions in the network: use switches instead of hubs!
- The bus loading is too high: change to 100 MBaud!
- You are logged in to the BC9000: this consumes additional processing capacity in the Bus Terminal Controller, which lengthens the response time.

The two following diagnostic inputs have different meanings, depending on the transmission type.

TCP/IP diagnosis

Table 20: BoxState

Value	Meaning
0x0000	No error
0x0001	No current inputs

Table 21: MissedCnt

Value	Meaning
0xyyyy	Number of telegrams that have not returned in time for the start of the task. This value should remain almost constant. If the value keeps rising, the task time should be lengthened.

UDP/IP diagnosis

Table 22: BoxState

Value	Meaning
0x0000	No error
0x0001	No current inputs
0x0002	Outputs are written as zero
0xxxzz	xx - warning level. The value here is incremented by one each time the data is not received in time. When data is exchanged correctly, the value is decremented by one. When the maximum warning level (default value 100) is reached, zero is written to the output data, and it is only enabled again for the normal process image when the warning level has reached zero.

Table 23: MissedCnt

Value	Meaning
0xyyyy	Number of telegrams that have not returned in time for the start of the task. This value should remain almost constant. If the value keeps rising, the task time should be lengthened.

7.7 ModbusTCP Diagnostic

See Modbus diagnostic function [▶ 46]

See ModbusTCP error answers [▶ 41]

8 Accessories

8.1 Fieldbus Box accessories

The necessary accessories for the Fieldbus Box Modules are also available from Beckhoff in protection class IP67. You may get an overview from the Beckhoff catalog or from our internet pages (http://www.beckhoff.com).

Fieldbus Accessories

- Pre-assembled cable
- Plug
- Distributor

Power supply

- Pre-assembled cable
- Plug
- Distributor

Sensor power supply

- Pre-assembled cable
- Plug
- Distributor

IP-Link

- Pre-assembled cable
- Plug

8.2 Power cables

Ordering data

Order designation	Power lead	Screw-in connector	Contacts	Cross-section	Length
ZK2020-3200-0020	Straight socket, open end	M8	4-pin	0.34 mm²	2.00 m
ZK2020-3200-0050					5.00 m
ZK2020-3200-0100					10.00 m
ZK2020-3400-0020	Angled socket, open end				2.00 m
ZK2020-3400-0050					5.00 m
ZK2020-3400-0100					10.00 m
ZK2020-3132-0001	Straight socket, straight socket				0.15 m
ZK2020-3132-0005					0.50 m
ZK2020-3132-0010					1.00 m
ZK2020-3132-0020					2.00 m
ZK2020-3132-0050					5.00 m
ZK2020-3334-0001	Angled socket, angled socket				0.15 m
ZK2020-3334-0005					0.50 m
ZK2020-3334-0010					1.00 m
ZK2020-3334-0020					2.00 m
ZK2020-3334-0050					5.00 m

Further available power cables may be found in the Beckhoff catalog or on our internet pages (http://www.beckhoff.com).

Technical data

Technical data	
Rated voltage according to IEC60 664-1	60 V_{AC} / 75 V_{DC}
Contamination level according to IEC 60 664-1	3/2
Insulation resistance IEC 60 512-2	$>10^9$W
Current carrying capacity according to IEC 60512-3	4 A
Volume resistance according to IEC 60512-2	< 5 mW
Protection class according to IEC 60529	IP65/66/67, when screwed together
Ambient temperature	-30°C to +80°C

9 Appendix

9.1 General operating conditions

Protection degrees (IP-Code)

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

1. Number: dust protection and touch guard	Definition
0	Non-protected
1	Protected against access to hazardous parts with the back of a hand. Protected against solid foreign objects of Ø50 mm
2	Protected against access to hazardous parts with a finger. Protected against solid foreign objects of Ø12,5 mm.
3	Protected against access to hazardous parts with a tool. Protected against solid foreign objects Ø2,5 mm.
4	Protected against access to hazardous parts with a wire. Protected against solid foreign objects Ø1 mm.
5	Protected against access to hazardous parts with a wire. Dust-protected. Intrusion of dust is not totally prevented, but dust shall not penetrate in a quantity to interfere with satisfactory operation of the device or to impair safety.
6	Protected against access to hazardous parts with a wire. Dust-tight. No intrusion of dust.

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

*) These protection classes define only protection against water!

Chemical Resistance

The Resistance relates to the Housing of the Fieldbus Box and the used metal parts.

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

Key

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

9.2 Approvals

Approvals

UL E172151

Conformity mark

CE

Type of protection

IP65/66/67 in accordance with EN60529

9.3 Test standards for device testing

EMC

Resistance: EN 61000-6-2

Emission: EN 61000-6-4

Resistance to Vibration

EN 60068-2-2 Vibration test, Amplitude 2 g (Standard 1 g)

EN 60068-2-27 Shock Test, Shock count 1000 (Standard 2)

9.4 Support and Service

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

Beckhoff's branch offices and representatives

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

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

You will also find further documentation for Beckhoff components there.

Beckhoff Support

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

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

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

Beckhoff Service

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

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

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

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

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

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