

BECKHOFF New Automation Technology

Documentation | EN

Fieldbus Box for Modbus



Table of contents

1 Foreword	5
1.1 Notes on the documentation.....	5
1.2 Safety instructions	6
1.3 Documentation Issue Status.....	7
2 Product Overview	8
2.1 The Fieldbus Box System.....	8
2.2 Fieldbus Box - Naming conventions	10
2.3 Firmware and hardware issue status.....	12
2.4 Modules - Technical Data	13
3 Modbus	14
3.1 Description of the Fieldbus System	14
3.2 Topology of the RS485.....	15
3.3 Modbus Cabling.....	16
3.3.1 Modbus RS485 connection.....	16
3.3.2 Modbus installation (RS485).....	16
3.3.3 Wiring the IP-Link	17
3.4 Communication.....	20
3.4.1 Modbus Protocol.....	20
3.4.2 Modbus Interface	20
3.4.3 Data frame	22
3.4.4 Modbus slave error answer	22
3.4.5 Functions	22
3.4.6 Parameters of the Fieldbus.....	32
4 Parameterisation and Commissioning	35
4.1 Start-up behavior of the Fieldbus Box	35
4.2 Modbus Address.....	36
4.3 Baud Rate.....	37
4.4 Process Data	38
4.4.1 Mapping of the FieldbusBox Modules.....	38
4.5 Configuration	42
4.5.1 Register Communication	42
5 Diagnosis and Error Handling	47
5.1 Diagnostic LEDs	47
5.2 Diagnostic LEDs for Modbus	48
5.3 Diagnostic LEDs for local errors	49
5.4 Check of the IP-Link connection.....	51
6 Fieldbus Box accessories	54
6.1 PROFIBUS Accessories	54
6.2 Power cables	60
6.3 KS2000 Configuration software.....	61
7 Appendix	62
7.1 General operating conditions.....	62
7.2 Approvals.....	64

7.3	Test standards for device testing.....	65
7.4	Support and Service	66

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.

The logo for EtherCAT, featuring the word "EtherCAT" in a bold, black, sans-serif font. A red arrow points from the top of the "A" towards the right, ending above the "T". A registered trademark symbol (®) is located to the right of the "T".

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

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

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

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

1.2 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Description of instructions

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

DANGER

Serious risk of injury!

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

WARNING

Risk of injury!

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

CAUTION

Personal injuries!

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

NOTE

Damage to environment/equipment or data loss

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



Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3 Documentation Issue Status

Version	Modification
1.0.2	<ul style="list-style-type: none">• System overview updated
1.0.1	<ul style="list-style-type: none">• Test standards for device testing updated• Minor routine corrections (typing errors, orthography etc.)
1.0	<ul style="list-style-type: none">• Modbus specific data changed• Signal types and the signals' connection assignments have been placed in the fieldbus-neutral documentation covering signal types (Fieldbus Box I/O Modules). You can find this on the internet in the Download area at http://www.beckhoff.com.
0.1	<ul style="list-style-type: none">• First version (only german language)

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:
IXxxxz-zyyy

IX describes the design:

"IP" stands for the [Compact Box design \[► 11\]](#)

"IL" stands for the [Coupler Box design \(with IP-Link\) \[► 11\]](#)

"IE" stands for the [Extension Box design \[► 11\]](#)

xxxz 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 \[► 11\]](#)

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 Modules - Technical Data

Technical data	IPxxxx-B730	IL230x-B730
Extension modules	-	Max. 120 with altogether 512 bytes input and 512 bytes output
Digital peripheral signals	according to I/O type	max. 960 inputs and outputs
Analog peripheral signals	according to I/O type	max. 255 inputs and outputs
Configuration facilities	via KS2000 or address switch	
Bus nodes	69	
Topology	Line	
Transmission Physics	RS 485 (like Profibus) shielded, twisted pairs	
Max. cable length	1200 m	
Protocol	RTU (ASCII)	
Baud rate	150 to 38400 Baud (default: 9,6 kBaud)	
Power supply connection	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	
Load voltage current consumption	Feed: 1 x M8 connector, 4-pin downstream connection: 1 x M8 socket, 4-pin (except IP/IE204x)	
Fieldbus connection	1 x M12 socket, 5-pin, (inverse coding)	
Electrical isolation	Channels/control voltage: no between the channels: no control voltage/fieldbus: yes	
Operating temperature	0°C ... +55°C	
Storage temperature	-25°C ... +85°C	
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, EN 60068-2-29	
EMC resistance burst / ESD	conforms to EN 61000-6-2 / EN 61000-6-4	
Protection class	IP 65/66/67 (according EN 60529)	
Installation position	any	
Approval	CE, UL E172151	

3 Modbus

3.1 Description of the Fieldbus System

Modbus is an open serial communication protocol, which is based on a Master/Slave architecture. Since it is quite easy to install on any serial interface, it has found a wide usage. Originally it was developed for the connection of controllers. But due to its spread of use, it is also used for I/O communication. The relatively low transmission rate of 38.4 kBaud admits Modbus for applications with a low number of stations or slow real time requirements.

The systems consists of a Master and several Slave stations, where the communication is purely controlled by the Master. Modbus uses two basic communication mechanisms:

- Polling - the Master sends a telegram to any station and waits for an answer
- Broadcast - the Master sends a telegram to all stations without waiting for an answer

The telegrams allows the reading and writing of process data (in / output data) either individually or group wise. The data can be coded in ASCII or in RTU format.

Modbus is used on different physical layers. The most common one is the RS485 physics, a (shielded) twisted pair of wires with an end termination like PROFIBUS.

Configuration

The station address can be defined with two decimal rotary switches from 1 to 69. Transmission rate and other features can be set with the KS2000 configuration software via the serial interface or with the address switch 69 -99.

Diagnostics

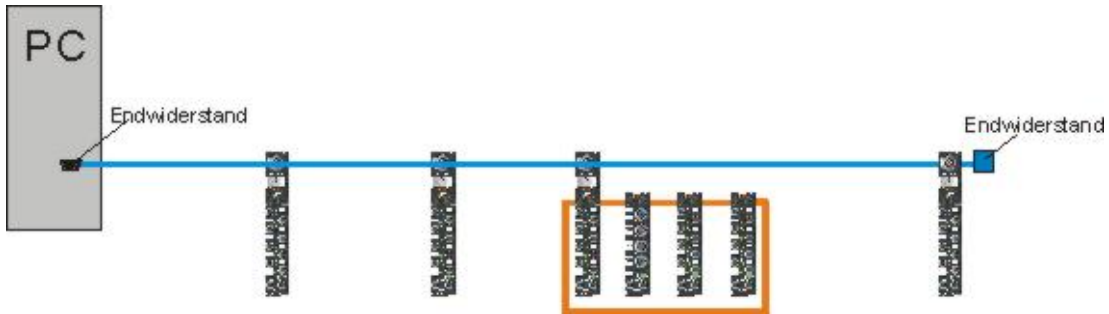
All BECKHOFF Modbus stations support the diagnostic functions of the Modbus protocol. All diagnostic messages are transmitted via the bus and are collected in the Master. The status of the network, the status of the device and the status of the in and outputs and the power are displayed with LEDs.

Cord sets

Since Modbus uses the RS485 transmission like PROFIBUS, all PROFIBUS cord sets can be used. Using the premoulded BECKHOFF cord sets simplifies the installation and helps avoiding mistakes. The delivery program covers fieldbus cord sets, power supply cord sets and sensor cord sets as well as termination resistors and field wire able connectors.

3.2 Topology of the RS485

RS 485 Modbus-Master with several IP/ILxxx-B730 Fieldbus Box modules:



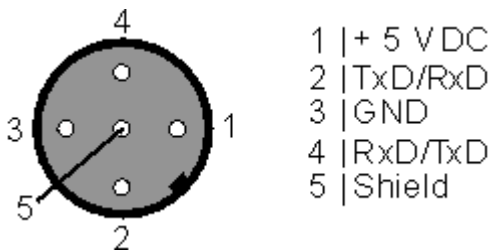
3.3 Modbus Cabling

3.3.1 Modbus RS485 connection

M12 circular connector

The M12 socket is inverse coded, and has five pins. Pin 1 is 5 V_{DC} and 3 is GND for the active termination resistor. These must never be misused for other functions, as this can lead to destruction of the device. Pin 2 and pin 4 are the Modbus signals. These must never be swapped over, as this will prevent communication. Pin 5 is the shield, and this is capacitatively coupled to the Fieldbus Box chassis.

M12 socket pin assignment



Type	M12	9 pin D-Sub
+5 V DC	Pin 1	Pin 6
TxD/RxD	Pin 2	Pin 3
GND	Pin 3	Pin 5
RxD/TxD	Pin 4	Pin 8
Shield	Pin 5	D-Sub female connector (case)

3.3.2 Modbus installation (RS485)

All PROFIBUS cord sets and connectors can be used for the Modbus wiring.

Distances

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

Baud-Rate in kBits/sec	9.6	19.2	38.4
distance in m	1200	1200	1200

Bus segments

A bus segment consists of at most 32 devices. 69 devices are permitted in a Modbus network. Repeaters are required to refresh the signal in order to achieve this number. PROFIBUS repeaters can be used. Each repeater is counted as one device.

IP-Link

IP-Link is the sub system of the modular Fieldbus Box system. The topology is a ring structure with an IP-Link Master IP230x-B730. A maximum of 120 extension boxes can be connected to one IP-Link Master (coupler box). The distance between two IP-Link boxes can be 15m maximum. Check the IP-Link [installation \[► 17\]](#) for more information.

3.3.3 Wiring the IP-Link

IP-Link is the subsidiary bus system for Fieldbus Boxes. Topologically, it is a ring. The IP-Link master is located in the Coupler Box (IL230x-Bxxx or IL230x-Cxxx). The extension boxes (IExxxx) are slaves. A maximum of 120 extension boxes may be connected. Additionally the available address range of the according Fieldbus Box coupler has to be considered. The distance between two extension boxes may not exceed 15 meter. When planning and installing the extension boxes, note that the last extension box in the optical fiber ring must be connected to the coupler box.

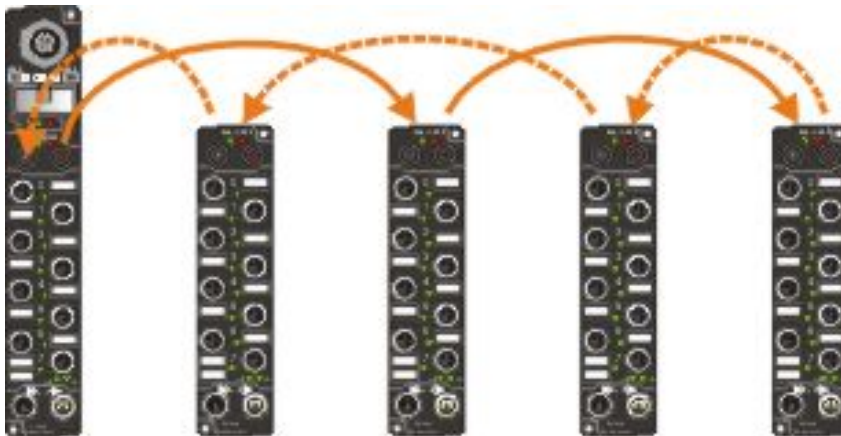
Structure/Topology

Line structure

In this case only every second extension boxes is connected. Note here that the maximum distance of the last extension box is halved.

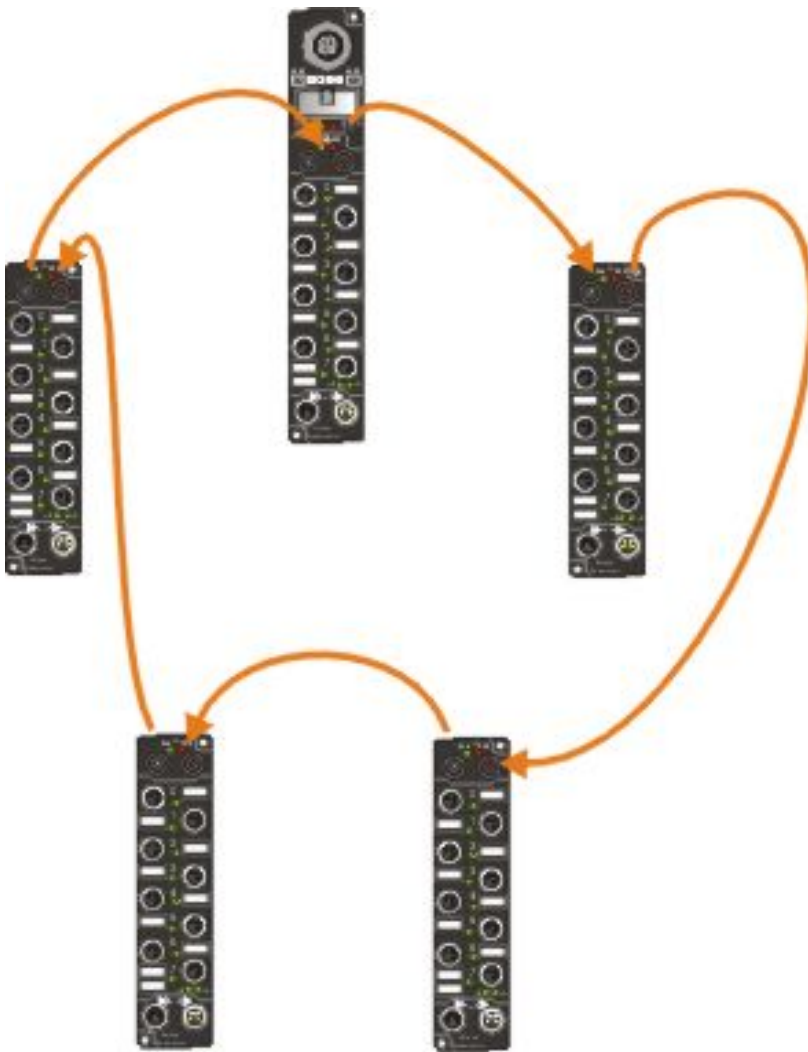
Example

You have 4 extension boxes (4 x 15 m = 60 m). Because only every second extension box is connected to the outward line, the maximum distance for the last extension box becomes 30 meter.



Ring structure

In this case the first and last extension box must each be no more than 15 meter from the Coupler Box.



Technical Data

Table 1: IP-Link

	IP-Link
Baud Rate	2 Mbaud
Medium	Light
Number of devices	max. 120
Distance between two stations	15 m
Cable	Plastic optical fiber 1000 um - single fiber core, PUR cladding with Kevlar fiber, diameter 5.5 mm
IP link connector	only ZS1020-0010 is permitted
Extraction force	20 N - 30 N

Table 2: IP-Link Z1101 Cable

	Z1101
Nominal external diameter	5.5 mm
Cladding thickness	1.4 mm
Fiber	Plastic fiber optic 980/1000 μm, PMMA
Core	PE cladding, 2.2 mm nominal diameter
Temperature	-20...+70°C
External cladding material	PUR
Nominal weight kg/km	25
Tensile strength (DIN VDE 0888 Part 100 Procedure 501)	Continuous 100 N Momentary 400 N
Bending radius	Radius minimum 50 mm
Alternating bending strength (DIN VDE 0888 Part 100 Procedure 509)	30,000 cycles
drag chain approval	Radius 10 x diameter, 2 Mio. cycles
Attenuation at 650 nm	< 200 dB/km

3.4 Communication

3.4.1 Modbus Protocol

RTU

In RTU mode the protocol starts with a pause of 3.5 character times, and finishes in the same way (illustrated in the diagram with T1-T2-T3-T4). The characters permitted for transmission in all fields are hexadecimal 0... 9, A..., F.

Start	Slave Address	Function	Data	CRC	END
t1-t2-t3-t4	1 Byte	1 Byte	n Bytes	2 Byte	t1-t2-t3-t4

ASCII

In ASCII mode the telegram starts with a colon (":", character 0x3A), and ends with a *carriage return and a line feed* ("CRLF", character 0x0D and 0x0A). The characters transferred are represented in the ASCII code.

Start	Slave Address	Function	Data	LRC	END
1 Byte 0x3A	2 Byte	2 Byte	n Bytes	2 Byte	2 Byte 0x0D, 0x0A

3.4.2 Modbus Interface

Address	Description		
0x0000 0x00FF	Process data interface inputs		
0x0800 0x08FF	Process data interface outputs		
0x1000 0x1006	Read only	Coupler Fieldbus Box code	
0x100A		2 Byte PLC interface	
0x100B		terminal diagnostics	
0x100C		coupler status	
0x1010		length of process image in Bit, analog outputs (without PLC variables)	
0x1011		length of process image in Bit, analog inputs (without PLC variables)	
0x1012		length of process image in Bit, discrete outputs	
0x1013		length of process image in Bit, discrete inputs	
0x1020		Watchdog, current time in [ms]	
0x110A		Read/Write	2 Byte PLC Interface
0x110B	terminal diagnostics		
0x1120	Watchdog predefined time in [ms] (Default: 1000)		
0x1121	Watchdog Reset Register		
0x1122	1 _{bin}		telegram watchdog (default)
	0 _{bin}		write telegram watchdog
0x4000 0x40FF*	Flag area (%MB..)*		

* only for Controller (BC7300, BC8x50, IL230x-C730)

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

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

FB: Fieldbus error, Watchdog timed out

KB: Bus terminal error/IP-Link error

CNF: Bus Coupler Configuration error

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

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.

* is not implemented in the BC8150

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.

3.4.3 Data frame

Description	Value
Data bits	8
Stop bits	1
Baud rate	9600 Baud
Parity	none

3.4.4 Modbus slave error answer

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

3.4.5 Functions

3.4.5.1 Modbus 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 [▶ 22]	1	Read digital outputs
Read input status [▶ 23]	2	Read digital inputs
Read holding register [▶ 24]	3	Read analog outputs and inputs / GPR
Read input register [▶ 25]	4	Read analog inputs / GPR
Force single coil [▶ 26]	5	Writing a digital output
Preset single register [▶ 27]	6	Write one analog output / GPR
Diagnostics [▶ 27]	8	Diagnostic
Force multiple coils [▶ 29]	15	Write a number of digital outputs
Preset multiple register [▶ 30]	16	Write a number of analog outputs / GPRs
Read / write registers [▶ 31]	23	Write and read a number of process data outputs / GPRs

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

3.4.5.2 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	1	0x01	"01"	0x30, 0x31
Start address high	0	0x00	"00"	0x30, 0x30
Start address low	0	0x00	"00"	0x30, 0x30
Count high	0	0x00	"00"	0x30, 0x30
Count low	10	0x10	"10"	0x31, 0x30
Error Check, LRC/ CRC		0xBC, 0xA7	"E4"	0x45, 0x34
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

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 Coupler is set, and the fieldbus coupler returns the value 4 in the first data byte.

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function code	1	0x01	"01"	0x30, 0x31
Byte Count	2	0x02	"02"	0x30, 0x32
Data Bit 0...7	4	0x04	"04"	0x30, 0x34
Data Bit 8...18	0	0x00	"00"	0x30, 0x30
Error Check, LRC/ CRC		0x23, 0x3D	"EE"	0x45, 0x45
End -Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.3 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	2	0x02	"02"	0x30, 0x32
Start address high	0	0x00	"00"	0x30, 0x30
Start address low	0	0x00	"00"	0x30, 0x30
Count high	0	0x00	"00"	0x30, 0x30
Count low	10	0x0A	"10"	0x31, 0x30
Error Check, LRC/ CRC		0xF8, 0xA7	"E3"	0x45, 0x33
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

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 Coupler is set, and the fieldbus coupler returns the value 4 in the first data byte.

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	2	0x02	"02"	0x32, 0x30
Byte Count	2	0x02	"02"	0x32, 0x30
Data Bits 0..7	1	0x01	"01"	0x31, 0x30
Data Bits 8..18	0	0x00	"00"	0x30, 0x30
Error Check, LRC/ CRC		0x20, 0x29	"F0"	0x46, 0x30
End -Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.4 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	Sample			
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	3	0x03	"03"	0x30, 0x33
Start address high	8	0x08	"08"	0x30, 0x38
Start address low	0	0x00	"00"	0x30, 0x30
Count high	0	0x00	"00"	0x30, 0x30
Count low	2	0x02	"02"	0x30, 0x32
Error Check, LRC/ CRC		0xC6, 0xC1	"E8"	0x45, 0x38
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

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	Sample			
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	3	0x03	"03"	0x30, 0x33
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 High-Byte	63	0x3F	"3F"	0x33, 0x46
Data 1 Low-Byte	255	0xFF	"FF"	0x46, 0x46
Data 2 High-Byte	0	0x00	"00"	0x30, 0x30
Data 2 Low-Byte	0	0x00	"00"	0x30, 0x30
Error Check, LRC/ CRC		0xC6, 0xC1	"B0"	0x42, 0x30
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.5 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	4	0x04	"04"	0x30, 0x34
Start-Address high	0	0x00	"00"	0x30, 0x30
Start-Address low	0	0x00	"00"	0x30, 0x30
Count high	0	0x00	"00"	0x30, 0x30
Count low	2	0x02	"02"	0x30, 0x32
Error Check, LRC/ CRC		0x71, 0x61	"EF"	0x45, 0x46
End -Frame		t1-t2-t3	CRLF	0x0D, 0x0A

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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	4	0x04	"04"	0x30, 0x34
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 High-Byte	0	0x00	"00"	0x30, 0x30
Data 1 Low-Byte	56	0x38	"38"	0x33, 0x38
Data 2 High-Byte	63	0x3F	"3F"	0x33, 0x46
Data 2 Low-Byte	11	0x0B	"0B"	0x30, 0x42
Error Check, LRC/ CRC		0x80, 0x7E	"6A"	0x36, 0x41
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.6 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	5	0x05	"05"	0x30, 0x35
Start-Address high	0	0x00	"00"	0x30, 0x30
Start-Address low	2	0x02	"02"	0x30, 0x32
Data high	255	0xFF	"FF"	0x46, 0x46
Data low	0	0x00	"00"	0x30, 0x30
Error Check, LRC/ CRC		0x2D, 0x50	"EF"	0x45, 0x46
End -Frame		t1-t2-t3	CRLF	0x0D, 0x0A

The coupler answers with the same telegram.

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	5	0x05	"05"	0x30, 0x35
Start-Address high	0	0x00	"00"	0x30, 0x30
Start-Address low	2	0x02	"02"	0x30, 0x32
Data high	255	0xFF	"FF"	0x46, 0x46
Data low	0	0x00	"00"	0x30, 0x30
Error Check, LRC/ CRC		0x2D, 0x50	"EF"	0x45, 0x46
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.7 Preset Single Register (Function 6)

The *Preset Single Register* function can be used to access the output process image and the interface.

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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	6	0x06	"06"	0x30, 0x36
Start-Address high	8	0x06	"06"	0x30, 0x36
Start-Address low	0	0x00	"00"	0x30, 0x30
Data high	63	0x3F	"3F"	0x33, 0x46
Data low	255	0xFF	"FF"	0x46, 0x46
Error Check, LRC/ CRC		0xDA, 0xB0	"A9"	0x41, 0x39
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

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

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	6	0x06	"06"	0x30, 0x36
Start-Address high	8	0x06	"06"	0x30, 0x36
Start-Address low	0	0x00	"00"	0x30, 0x30
Data high	63	0x3F	"3F"	0x33, 0x46
Data low	255	0xFF	"FF"	0x46, 0x46
Error Check, LRC/ CRC		0xDA, 0xB0	"A9"	0x41, 0x39
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.8 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	8	0x08	"08"	0x30, 0x38
Sub function high	0	0x00	"00"	0x30, 0x30
Sub function low	0	0x00	"00"	0x30, 0x30
Data high	2	0x02	"02"	0x30, 0x32
Data low	3	0x03	"03"	0x30, 0x33
Error Check, LRC/ CRC		0xA1, 0xC0	"E8"	0x45, 0x38
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

Response

Byte Name	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	8	0x08	"08"	0x30, 0x38
Sub function high	0	0x00	"00"	0x30, 0x30
Sub function low	0	0x00	"00"	0x30, 0x30
Data high	2	0x02	"02"	0x30, 0x32
Data low	3	0x03	"03"	0x30, 0x33
Error Check, LRC/ CRC		0xA1, 0xC0	"E8"	0x45, 0x38
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

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 Controller will restart, it will send a response with Sub function 1.

**Note**

The BC8150 support the sub function 0 and 1. All other sub function answer with the value "0".

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)
0x000B	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)
0x000C	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

3.4.5.9 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	15	0x0F	"0F"	0x30, 0x46
Start-Address high	0	0x00	"00"	0x30, 0x30
Start-Address low	0	0x00	"00"	0x30, 0x30
Length high	0	0x00	"00"	0x30, 0x30
Length low	20	0x14	"14"	0x31, 0x34
Byte Count	3	0x03	"03"	0x30, 0x33
Data 1 Bit 0..7	255	0xFF	"FF"	0x46, 0x46
Data 2 Bit 8..15	255	0xFF	"FF"	0x46, 0x46
Data 3 Bit 16..23	0	0x00	"00"	0x30, 0x30
Error Check, LRC/ CRC		0x01, 0x95	"D1"	0x44, 0x31
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

Response

The Bus Coupler answers with the same telegram.

Byte Name	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	15	0x0F	"0F"	0x30, 0x46
Start-Address high	0	0x00	"00"	0x30, 0x30
Start-Address low	0	0x00	"00"	0x30, 0x30
Length high	0	0x00	"00"	0x30, 0x30
Length low	20	0x14	"14"	0x31, 0x34
Error Check, LRC/ CRC		0xDA, 0xB0	"D2"	0x44, 0x32
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.10 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	16	0x10	"10"	0x31, 0x30
Start-Address high	8	0x08	"08"	0x30, 0x38
Start-Address low	0	0x00	"00"	0x30, 0x30
Length high	0	0x00	"00"	0x30, 0x30
Length low	2	0x02	"02"	0x30, 0x32
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 Byte 1	127	0x7F	"7F"	0x37, 0x46
Data 1 Byte 2	255	0xFF	"FF"	0x46, 0x46
Data 2 Byte 1	63	0x3F	"3F"	0x33, 0x46
Data 2 Byte 2	255	0xFF	"FF"	0x46, 0x46
Error Check, LRC/ CRC		0xCD, 0xE3	"1B"	0x31, 0x42
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

Response

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

Byte Name	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	16	0x10	"10"	0x31, 0x30
Start-Address high	8	0x08	"08"	0x30, 0x38
Start-Address low	0	0x00	"00"	0x30, 0x30
Length high	0	0x00	"00"	0x30, 0x30
Length low	2	0x02	"02"	0x30, 0x32
Error Check, LRC/ CRC		0x43, 0x02	"DB"	0x44, 0x42
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.5.11 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 / Write 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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	23	0x17	"17"	0x31, 0x17
Read Start-Address high	0	0x00	"00"	0x30, 0x30
Read Start-Address low	0	0x00	"00"	0x30, 0x30
Read Länge high	0	0x00	"00"	0x30, 0x30
Read Länge low	2	0x02	"02"	0x30, 0x32
Write Start-Address high	8	0x08	"08"	0x30, 0x38
Write Start-Address low	0	0x00	"00"	0x30, 0x30
Write Länge high	0	0x00	"00"	0x30, 0x30
v Länge low	2	0x02	"02"	0x30, 0x32
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 high	63	0x3F	"3F"	0x33, 0x46
Data 1 low	255	0xFF	"FF"	0x46, 0x46
Data 2 high	127	0x7F	"7F"	0x37, 0x46
Data 2 low	255	0xFF	"FF"	0x46, 0x46
Error Check, LRC/ CRC		0x76, 0xD3	"12"	0x31, 0x32
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

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	Modbus RTU		Modbus ASCII	
Start - Frame	-	-	":"	0x3A
Address	11	0x0B	"0B"	0x30, 0x42
Function-Code	23	0x17	"17"	0x31, 0x17
Byte Count	4	0x04	"04"	0x30, 0x34
Data 1 high	0	0x00	"00"	0x30, 0x30
Data 1 low	56	0x38	"38"	0x33, 0x38
Data 2 high	63	0x3F	"3F"	0x33, 0x46
Data 2 low	11	0x0B	"0B"	0x30, 0x42
Error Check, LRC/ CRC		0xF8, 0xA7	"E3"	0x45, 0x33
End - Frame		t1-t2-t3	CRLF	0x0D, 0x0A

3.4.6 Parameters of the Fieldbus**3.4.6.1 Parameterization via address switch**

Some features of the coupler can be set by the address switch. After the setting, the power has to be switched off and on again.

Repeat the procedure for each parameter you want to change.

Address	Type	Value	
0	Function	0	KS2000 communication (parameter: 19200 Baud, 8 data, even parity, 1 Stop bit)
71-77	Data - frame	71	8 data bits no Parity
		72	8 data bits even Parity
		73	8 data bits odd Parity
		74	7 data bits even Parity
		75	7 data bits odd Parity
		76	1 Stop bit
		77	2 Stop bits
81-89	Baud rate	81	38400 Baud
		82	19200 Baud
		83	9600 Baud
		84	4800 Baud
		85	2400 Baud
		86	1200 Baud
		87	600 Baud
		88	300 Baud
		89	150 Baud
90-95	Watchdog	90	Disable
		91	1000 ms
		92	2000 ms
		93	4000 ms
		94	8000 ms
		95	max. ca. 65 sec
96-97	Modbus Mode	96	RTU
		97	ASCII
98	Function	98	factory setting
99	Function	99	reserved

3.4.6.2 Table 100

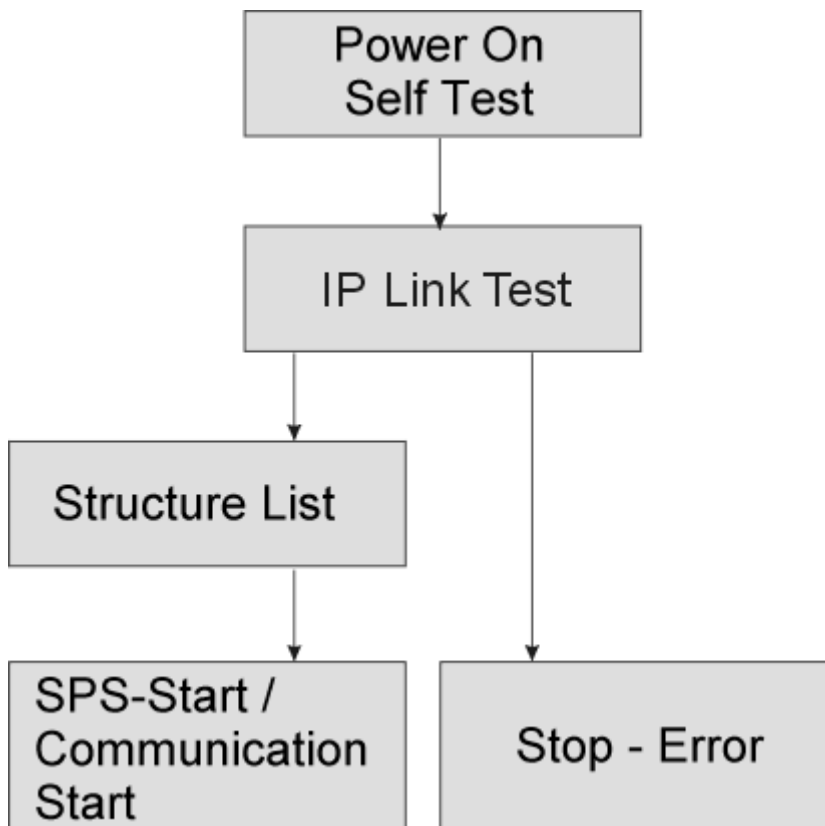
Table 3: Table 100

Register	Type	Value		Default
0	Watchdog	0...65767	0 ... Disable	1000 [ms]
1	Protocol	0	RTU	0
		1	ASCII	
2	Data bits Bit 0..3	1	8 Data bytes	1
		3	7 Data bytes & Parity	
		7	8 Data bytes & Parity	
	Stop bits Bit 4	0	one Stop bit	0
		1	two Stop bits	
	Parity Bit 12	0	Even	0
1		Odd		
3	Baud rate	0x8000	38400 Baud	0x8002
		0x8001	19200 Baud	
		0x8002	9600 Baud	
		0x8003	4800 Baud	
		0x8004	2400 Baud	
		0x8005	1200 Baud	
		0x8006	600 Baud	
		0x8007	300 Baud	
		0x8008	150 Baud	

4 Parameterisation and Commissioning

4.1 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).



4.2 Modbus Address

The address of the serial Fieldbus Box has to be set via two rotary switches. The default setting is 11. Addresses from 01 to 69 are allowed. Every address has to be unique in the Modbus network. The address has to be changed in the power off mode of the box. Be careful, that the switch snaps into the desired position.

The left switch value is times 10, the right switch times 1. The new address is valid after a power on.

Example

You want to set address 34:

- left switch: 3
- right switch: 4



4.3 Baud Rate

The default baud rate is 9600 baud. This can be changed via the [address switch](#) or the [KS2000 software](#) [► 32].

**Note**

Address "0" is reserved for KS2000 and TwinCAT communication. Default baud rate for this is 192000 baud. In KS2000 the Multipoint address 0 has to be set.

The default setting for the Fieldbus Box is

Modbus	Frame	Watchdog	End of Frame
RTU	8 Data, 1 Stop, 9600 baud, no parity	1000 ms	3 ms

4.4 Process Data

4.4.1 Mapping of the FieldbusBox Modules

4.4.1.1 Mapping of the Fieldbus Box modules

The Modbus process image makes a fundamental distinction between digital and byte-oriented (word-oriented) signals. Additionally, a distinction is made between inputs and outputs. The Coupler Box has a modular mapping, which consists of the different extensions.

Bit oriented Fieldbus Box modules

Bit oriented Fieldbus Box modules have a fixed number of input and/or output data. These can not be changed.

Module	Inputs [Bit]	Modbus Function	Outputs [Bit]	Modbus Function
IP10xx-B8x0 IE10xx	8	read 2, 3, 4, 23	-	-
IP20xx-B8x0 IE20xx	-	-	8	read 1, 3, 4, 23 write 5, 6, 15, 16, 23
IP23xx-B8x0 IE23xx	8 (High-nibble is not used, last 4 Bits)	read 2, 3, 4, 23	8 (High-Nibbel is not used, last 4 Bits)	read 1, 3, 4, 23 write 5, 6, 15, 16, 23
IP240x-B8x0 IE240x	8	read 2, 3, 4, 23	8	read 1, 3, 4, 23 write 5, 6, 15, 16, 23

Byte oriented Fieldbus Box modules

Byte oriented Fieldbus Box modules can be parameterized differently. The table shows the default setting.

Module	Inputs [Bit]	Modbus Function	Outputs [Bit]	Modbus Function
IP15xx-B730 IE15xx	6	read 3, 4, 23	6	read 3, 4, 23 write 6, 16, 23
IP25xx-B730 IE25xx	6	read 3, 4, 23	6	read 3, 4, 23 write 6, 16, 23
IP3xxx-B730 IE3xxx	8	read 3, 4, 23	-	-
IP41xx-B730 IE41xx	-	-	8	read 3, 4, 23 write 6, 16, 23
IP5xxx-B730 IE5xxx	6	read 3, 4, 23	6	read 3, 4, 23 write 6, 16, 23
IP6xxx-B730 IE6xxx	4	read 3, 4, 23	4	read 3, 4, 23 write 6, 16, 23

Memory partitioning

For the digital functions there is an input and an output process image is available. For all complex signals (all Bus Terminals or modules that are connected with a byte or more into the Insert process image) a process image is available. Anhand der Offset Adresse wird unterschieden zwischen Eingang (Offset 0x0000) und Ausgang (0x0800) - [see Modbus Interface \[► 20\]](#).

Digital
Output
KL2xxx
IP/IL/IE 2xxx

Digital
Input
KL1xxx
IP/IL/IE 1/2xxx

All
Signals

Modbus function
1, 2, 5, 15

Modbus function
3, 4, 6, 16, 23

4.4.1.2 Mapping of an IL230x-B730

An IP-Link system always starts with a IL230x-B730 and some (max. 120) extension modules. The mapping depends on the number and type of the connected extension modules.

Rule: First all byte oriented extensions appear in the mapping, in the sequence of their appearance in the IP-Link ring. After that, all bit oriented (digital) extensions in their sequence, rounded to 8 Bit.

Sample 1

System:

1. IL2301-B730
2. IE1001
3. IE1002
4. IE2042

Module	Position	Input			Output		
		Size	Function	Offset	Size	Function	Offset
IL2301-B730	1	4 Bit	2 read	0..3	4 Bit	1 read	0..3
			3, 4 read			0x0000 (Bit 0..3)	3 read
						5 write	0..3
						6 write	0x0800 (Bit 0..3)
						15 write	0..3
						16 write	0x0800 (Bit 0..3)
IE1001	2	8 Bit	2 read	4..11	-		
			3, 4 read	0x0000 (Bit 4..11)			
IE1002	3	8 Bit	2 read	12..19	-		
			3, 4 read	0x0000 (Bit 12..15) 0x0001 (Bit 0..3)			
IE2042	4	-			8 Bit	1 read	4..11
						3 read	0x0800 (Bit 4..11)
						5 write	4..11
						6 write	0x0800 (Bit 4..11)
						15 write	4..11
						16 write	0x0800 (Bit 4..11)

Modbus always work with a size 16 Bit or one word. In this sample a word is transmitted, but the last bits are filled with "0".

Sample 2

System:

1. IL2301-B730
2. IE1001
3. IE3312
4. IE3112
5. IE4112
6. IE2302
7. IE2042

Modul	Position	Input			Output		
		Size	Function	Offset	Size	Function	Offset
IE3312	3	4 Words	3, 4 read	0x0000..0x0003	-		
IE3112	4	4 Words	3, 4 read	0x0004..0x0007	-		
IE4112	5	-			4 words	3 read	0x0800...0x0803
						6 write	0x0800...0x0803
						16 write	0x0800...0x0803
IL2301-B730	1	4 Bit	2 read	0..3	-		
			3, 4 read	0x0008 (Bit 0..3)			
IE1001	2	8 Bit	2 read	4..11	-		
			3, 4 read	0x0008 (Bit 4..11)			
IE2302	6	4 Bit	2 read	12..15	4 Bit	1 read	0..3
						3 read	0x0804 (Bit 0..3)
						5 write	0..3
			3, 4 read	0x0008 (Bit 12..15)		6 write	0x0804 (Bit 0..3)
			15 write	0..3			
IE2042	7	-			8 Bit	1 read	4..11
						3 read	0x0804 (Bit 4..11)
						5 write	4..11
						6 write	0x0804 (Bit 4..11)
						15 write	4..11
						16 write	0x0804 (Bit 4..11)

4.5 Configuration

4.5.1 Register Communication

4.5.1.1 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.

4.5.1.2 Activating of the complete mapping

To use the register communication, the complete mapping (with Control-/Status Byte) has to be activated in the Fieldbus Box.

The mapping can be activated with the KS2000 configuration software.

4.5.1.3 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 4: 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 5: 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	0xFF	0xFF

Table 6: 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	0xFF	0xFF

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 7: 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	0xFF	0xFF

Table 8: 4. Read Register 32 (verify changed register)

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

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 9: 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	0xFF	0xFF

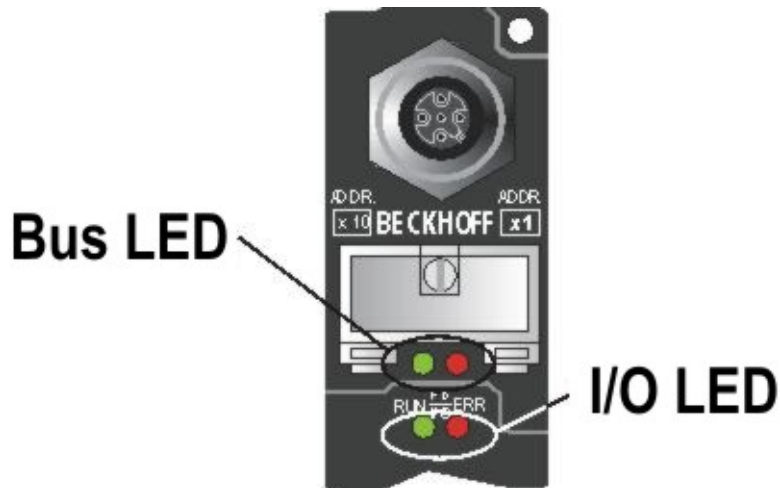
5 Diagnosis and Error Handling

5.1 Diagnostic LEDs

Error Diagnosis

There are 2 kinds of error:

- Fieldbus error
- Local errors [▶ 49] in a Compact Box or Coupler Box



Blink Codes

Blinking sequence	Meaning
Fast blinking	Start of the error code
First slow sequence	Error code
Second slow sequence	Error argument
Third slow sequence (optional)	Error argument if more than 20 extensions



Start of the error code

Error code

Error argument

5.2 Diagnostic LEDs for Modbus

The two upper LEDs describe the Modbus status

LED green	LED red	E/A RUN	Description	Help
off	off	no meaning	- communication loss - Watchdog time expired	- start communication new - Watchdog reset or new start
off	on or flashing	no meaning	- error in data transmission	this can be caused by: - Parity error - Check sum error - too many stop bits - wrong baud rate
on	off	an	- Module in operation	no error

5.3 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	(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:

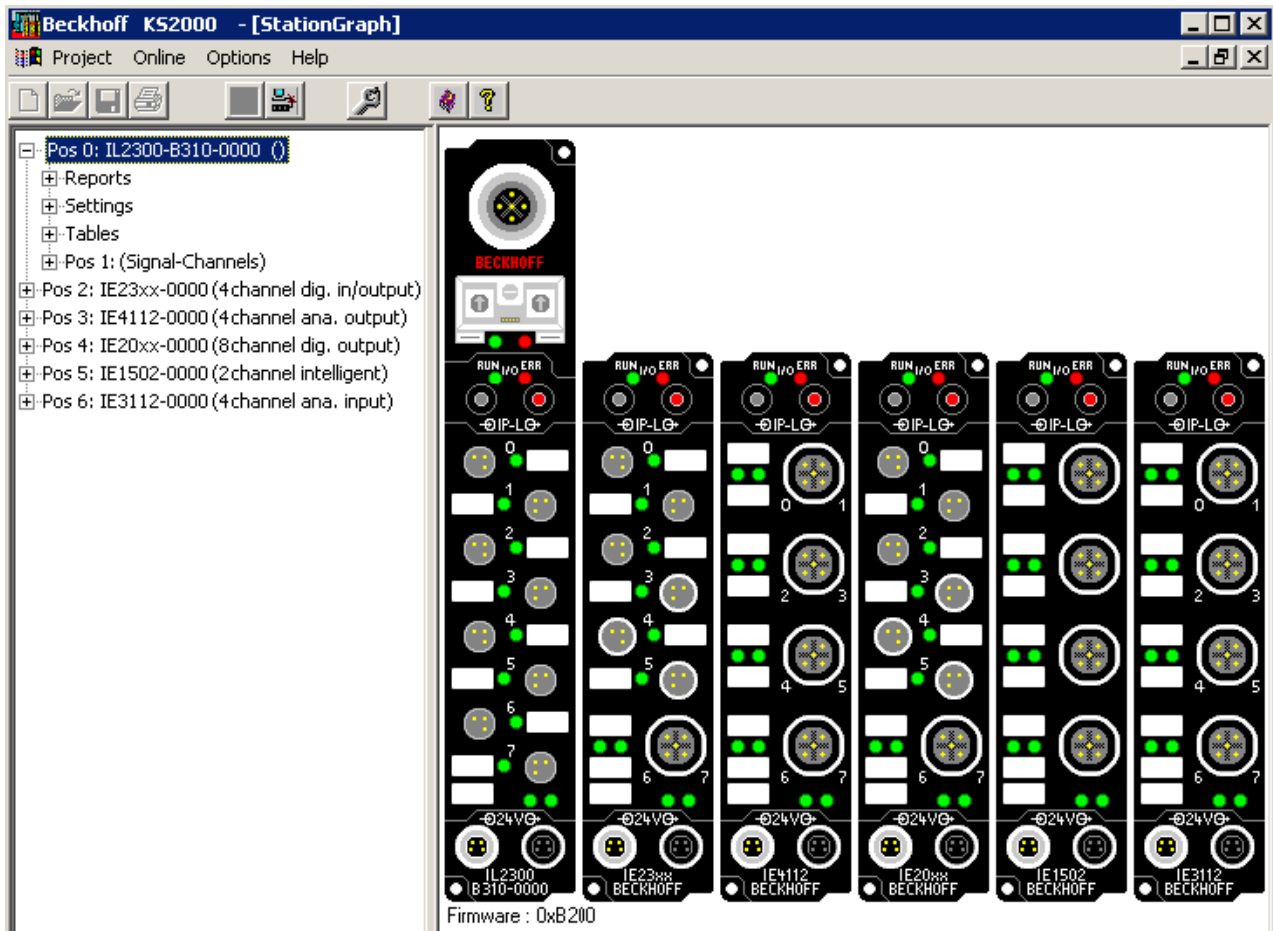
- bad configured IP-Link connectors
- IP-Link cable with higher dampening, e.g. because of a sharp curve
- contaminated sender LED (module before the faulty one)
- contaminated receiver

The internal IP-Link error counter [▶ 51] of the Coupler Box can be read with the KS2000 software.

5.4 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.

- Settings
- Tables
 - 000: Configuration Coupler
 - 009: Terminal typ (auto)
 - 087: Table 87
 - 088: Table 88
 - 090: Diagnostic coupler**
 - 091: Diagnostic processimage
 - 092: Diagnostic terminal channel 1
 - 093: Diagnostic terminal channel 2
 - 094: Diagnostic terminal channel 3
 - 095: Diagnostic terminal channel 4

<i>Register</i>				
Offset	HEX	UINT	BIN	
000	0x0001	1	0000	0000 0000 0001
001	0x0000	0	0000	0000 0000 0000
002	0x0000	0	0000	0000 0000 0000
003	0x0000	0	0000	0000 0000 0000
004	0x0000	0	0000	0000 0000 0000
005	0x002A	42	0000	0000 0010 1010
006	0x0000	0	0000	0000 0000 0000
007	0x0000	0	0000	0000 0000 0000
008	0x0000	0	0000	0000 0000 0000
009	0x0000	0	0000	0000 0000 0000

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 be looked for before these modules.

- Settings
- Tables
 - Pos 1: (Signal-Channels)
 - Pos 2: IE23xx-0000 (4 channel dig. in/output)
 - Pos 3: IE4112-0000 (4 channel ana. output)
 - Pos 4: IE20xx-0000 (8 channel dig. output)
 - Pos 5: IE1502-0000 (2 channel intelligent)
 - Pos 6: IE3112-0000 (4 channel ana. input)

<i>Register</i>				
Offset	HEX	UINT	BIN	
000	0x0000	0	0000	0000 0000 0000
001	0x0000	0	0000	0000 0000 0000
002	0x0000	0	0000	0000 0000 0000
003	0x0000	0	0000	0000 0000 0000
004	0x000A	10	0000	0000 0000 0000 1010
005	0x0000	0	0000	0000 0000 0000 0000
006	0x0008	8	0000	0000 0000 0000 1000
007	0x0000	0	0000	0000 0000 0000 0000
008	0x0000	0	0000	0000 0000 0000 0000

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 (IExxx), 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.

6 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

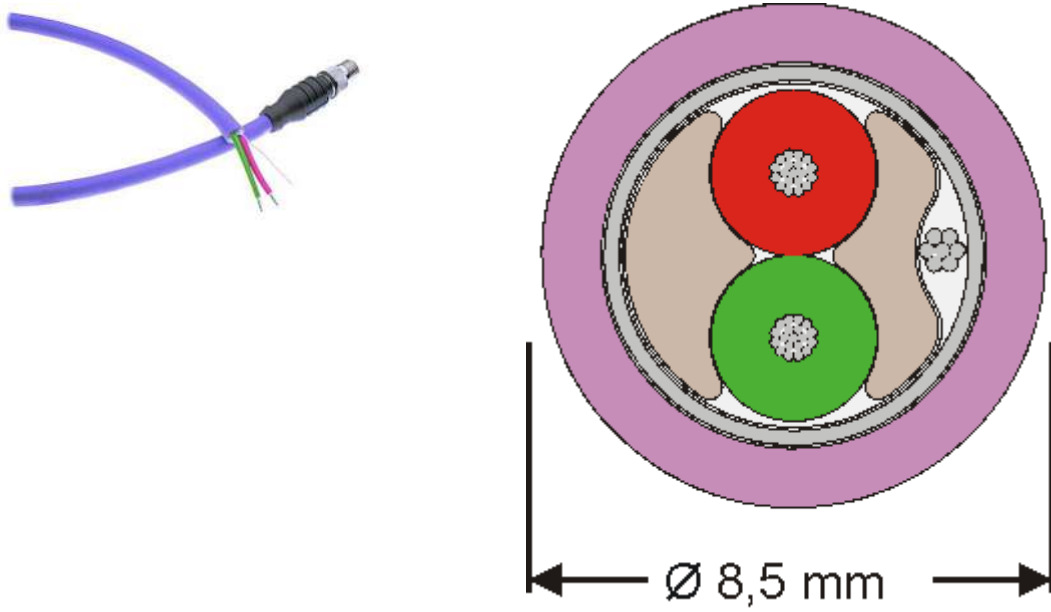
- Pre-assembled cable
- Plug

6.1 PROFIBUS Accessories

These accessories can also be used for other RS485-based fieldbus systems (Modbus, RS485)

PROFIBUS cables
Ordering data

Order designation	Fieldbus cable	Connector	Contacts	Cross-section	Length	Fig.
ZK1031-6200-1020	Straight socket, open end	M12 screw-in	8-pin	0.32	2.00	A
ZK1031-6200-1050	Straight socket, open end	M12 screw-in	8-pin	0.32	5.00	A
ZK1031-6200-1100	Straight socket, open end	M12 screw-in	8-pin	0.32	10.00	A
ZK1031-6200-1150	Straight socket, open end	M12 screw-in	8-pin	0.32	15.00	A
ZK1031-6100-1020	Straight plug, open end	M12 screw-in	8-pin	0.32	2.00	B
ZK1031-6100-1050	Straight plug, open end	M12 screw-in	8-pin	0.32	5.00	B
ZK1031-6100-1100	Straight plug, open end	M12 screw-in	8-pin	0.32	10.00	B
ZK1031-6100-1150	Straight plug, open end	M12 screw-in	8-pin	0.32	15.00	B
ZK1031-6251-1003	Straight plug, straight socket	M12 screw-in	8-pin	0.32	0.30	A and B
ZK1031-6251-1005	Straight plug, straight socket	M12 screw-in	8-pin	0.32	0.50	A and B
ZK1031-6251-1010	Straight plug, straight socket	M12 screw-in	8-pin	0.32	1.00	A and B
ZK1031-6251-1020	Straight plug, straight socket	M12 screw-in	8-pin	0.32	2.00	A and B
ZK1031-6251-1050	Straight plug, straight socket	M12 screw-in	8-pin	0.32	5.00	A and B
ZK1031-6251-1100	Straight plug, straight socket	M12 screw-in	8-pin	0.32	10.00	A and B
ZK1031-6251-1150	Straight plug, straight socket	M12 screw-in	8-pin	0.32	15.00	A and B



Plug

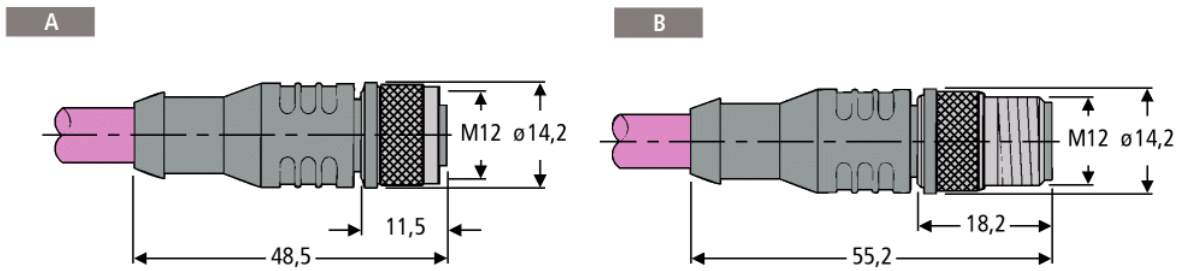
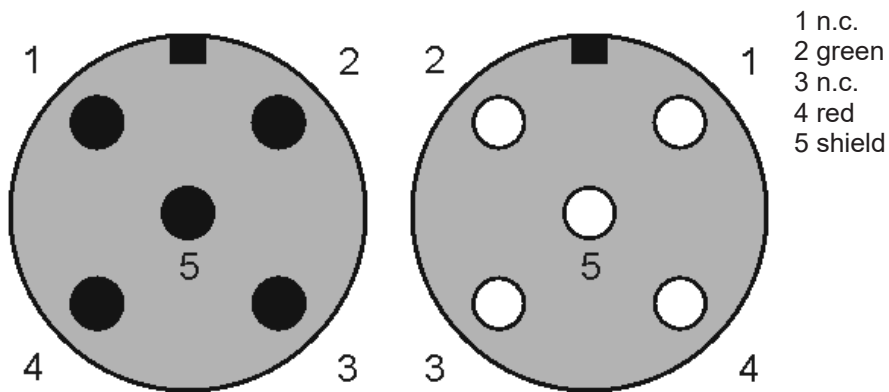


Table 10: Pin assignment



Technical data

Electrical data	
Ratings	300 V, 80°C
Material	Outer cladding PUR, inner insulation polyethylene
Cross-section	0.32 mm ²
DC resistance	54.12 Ohm/km
Line capacitance	26.9 pF/m
Nominal impedance	150 Ohm at 1 MHz
Approval	UL approval, AWM, Type 20233, 80°C, 300V; CSA AWM, I/II A/B, 80°C, 300V FT1

PROFIBUS Accessories

Ordering data

Order designation	Description
ZS1031-2600	T-piece, 12 Mbaud
ZS1000-2600	Y-piece, 12 Mbaud
ZS1000-1610	Terminating resistor (plug)
ZS1000-0620	Coupling for field assembly
ZS1000-0610	Plug for field assembly
ZS1031-6610	Control cabinet lead-through M12, plug connector

Table 11: ZS1031-2600

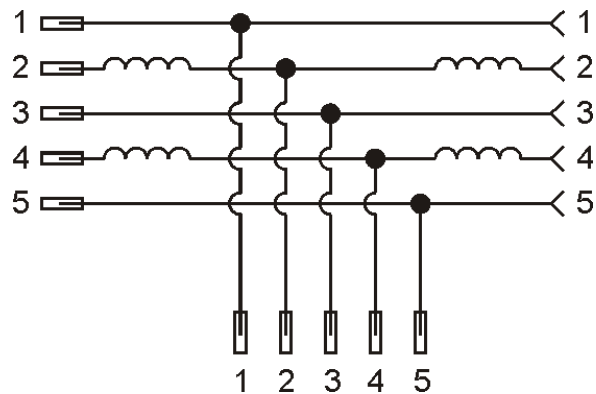
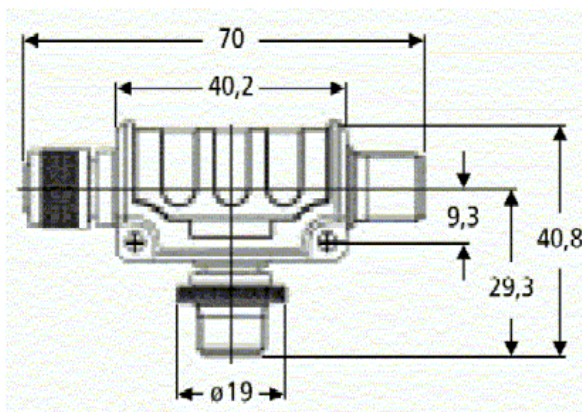
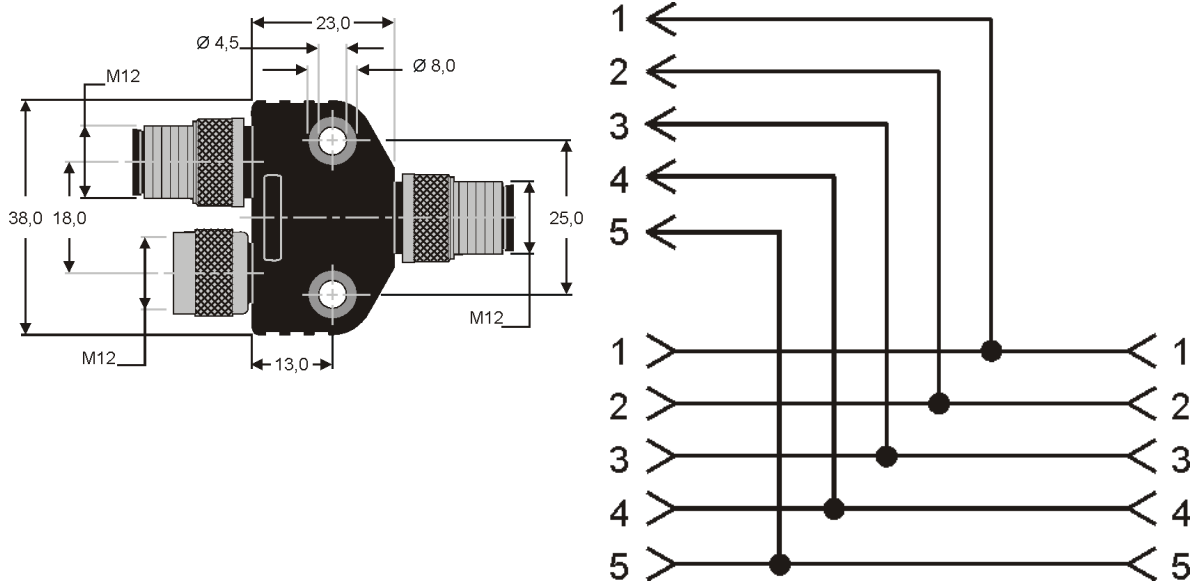


Table 12: ZS1000-2600



ZS1000-1610

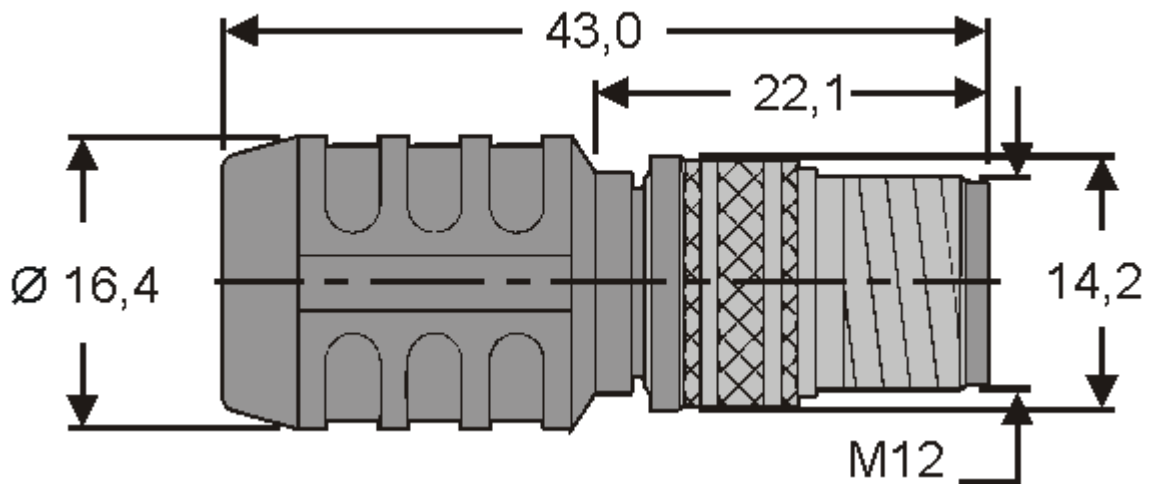
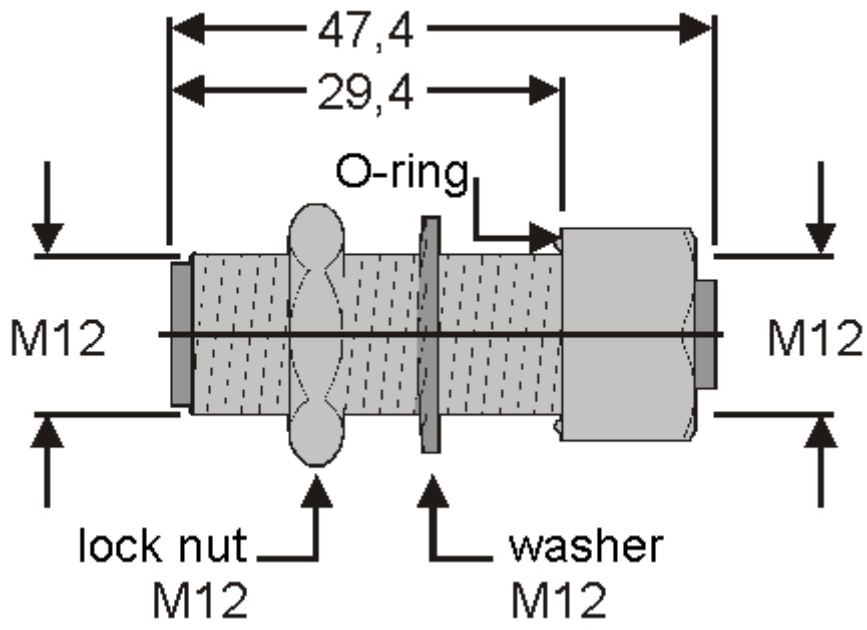


Fig. 1: Add Profibus End.gif (7936 Byte)

ZS1031-6610



6.2 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 mm ²	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

6.3 KS2000 Configuration software

The KS2000 configuration software allows the configuration, commissioning and diagnostics of the coupler and extensions. The connection is established with a serial communication cable via the RS232 port.

i **Note**

Take care to use the right baud rate in the KS2000. If you have switched the Modbus to e.g. 38.4 kBaud, you have to change the KS2000 settings accordingly.

7 Appendix

7.1 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

7.2 Approvals

Approvals

UL E172151

Conformity mark

CE

Type of protection

IP65/66/67 in accordance with EN60529

7.3 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)

7.4 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