Manual

SSM-BG41/BG42/ BG43

Order-No.: VIPA SSM-HB29E
Rev. 99/49

The information in this manual is supplied without liability. Its contents may be amended at any stage without prior notice. The software described in this manual is subject to the conditions of a general licensing agreeement. The user has a copy licence, which entitles one to use the software within their company.

Contravention will result in liability for damages.

Ohmstraße 4, D-91074 Herzogenaurach

Phone.: +49 (9132) 744-0

Fax.: +49 (9132) 744-144

EMail: info@vipa.de

http://www.vipa.de

Hotline: +49 (9132) 744-114

About this Manual

This manual describes the operation of the communication processors SSM-BG41/42/43. The modular communication processor SSM is an intelligent carrier building block, upon which a series of input, output and communication modules can be equipped.

Depending on the building block, up to 3 expansion boards can be optionally combined.

It can be inserted into automation equipment 115U (without fan) up to 188U, as well as into central or add-on equipment.

Overview

Chapter 1: Introduction

In this chapter you will find general safety information on the handling of building blocks liable to electrostatic danger. An overview of the modules which can be used is also provided.

Chapter 2: Interface building blocks

The building blocks BG41, BG42 and BG43 are introduced in this chapter, as well as the block diagram firmware and details on memory distribution firmware.

Chapter 3: Interface modules

Chapter 3 attends to the usable modules. You will find a description to the structure, mode of operation and allocation of pins. The use with or without data handling blocks will be described for the module SSI, counter, analogue input/output and DCF77.

Chapter 4: Software

In this chapter you will find instructions on how to parameterize the buidling blocks. For parameterizing you can use the VIPA data handling blocks or you can parameterize the building block via interface channel. The large part of the chapter attends to the use of building blocks with procedures e.g. STX/ETX and 3964(R) with or without RK512.

Chapter 5: Installation

The configuration of a building block is described in this chapter. Instructions are given on how to adjust the address, on the function of the plug connector and on the voltage supply of modules. Within this chapter you are also provided with information on the start-up response of a configured building block, as well as help in fault diagnosis. Apart from construction guidelines, which contain information on the interference immune construction of stored programmable controls, program examples of different transmission protocols are also detailed.

Chapter 6: Technical Data

In this chapter technical data on building blocks and modules is provided.

Contents

1 Introduction

1 Introduction

1.1 Safety and handling precautions for the user

1.1.1 Handling electrostatically sensitive modules

VIPA building blocks are equipped with highly integrated components in MOS-technology. These components are highly sensitive to excess voltage, which can for example result from electrostatic discharge.

Building blocks at risk are characterized by the following symbol:

The symbol is to be found on building blocks, building block carriers or on the packaging and indicate building blocks liable to electrostatic danger.

Building blocks liable to electrostatic danger can be destroyed by energies or voltage, which lie far below the human's point of perception. If a person, who is not electrically discharged, handles building blocks liable to electrostatic danger, such voltages may occur, leading to damage of components and therefore reducing the functionability of the building blocks or rendering them completely useless. Only a very small percentage of such damaged building blocks will be immediately recognized as faulty. The fault may first become apparent after being in operation for a longer period of time.

Due to static discharge, damaged components can develop a temporary fault by changes in temperature, vibrations or a change in load.

Functional interferences and failures can only be effectively prevented by a consistent employment of protective devices and a responsible compliance with the handling regulations.

1.1.2 Shipping electrostatically sensitive modules

Always use the original packaging for dispatch. The prepared building blocks could additionally be wrapped with a conductive packaging. Conductive packaging is antistatic foil or metallozed plastic-film containers..

A battery is to be found on the interface building block as a compensating charge for the clock. Insure that the battery's connections are not touched or short-circuited, when dispatched in conductive packaging

1.1.3 Tests and modifications to electrostatically sensitive modules

The following aspects are to be observed, when measuring building blocks liable to electrostatic danger:

- Floating measuring equipment is to be temporarily discharged.
- Measuring equipment to be used is to be earthed.

When altering building blocks liable to electrostatic danger, one is to ensure that an earthed soldering iron is used.

When working with or on building blocks liable to electrostatic danger, one is to ensure that both the people and the working material are sufficiently earthed.

1.2 General

The building blocks are differentiated by the number of existing interfaces, which can be adapted by means of VIPA interface modules to different physical modes of transmission and can be used for particular functions. They can be linked with any peripheral device, which uses the corresponding transmission technology.

Every VIPA interface building block (BG41, BG42, BG43) is additionally equipped with a battery-backed real-time clock and a diagnostic interface, which can be connected to a diagnostic adapter (UPI-FOX).

1.3 Area of employment

The VIPA interface building block is used in PLC systems as an interface for the input and output of data. It is equipped with transmitting and receiving buffers (each with 256 bytes), which are run by the Z80 processor situated on the module. The module's 25-pole SubD sockets are pin-compatible with CP 525 when using serial communication modules.

The interface building block can, for example, be used to connect a keyboard, a terminal, a printer or to communicate with a computer. Data for output or input can be filed in data handling blocks in the programmable controller (PLC).

If no personal parameters are set, the following values for each interface are preset by serial transmission:

- No protocol
- Rate of transmission 9600 baud
- Start bit 1
- Data bit 8
- Parity check even parity
- Stop bit 2

The interfaces can be adapted to certain physical modes of transmission by modules, which are inserted in the designed slots of the mother circuit board.

The following VIPA modules can be presently used:

- 20mA Current Loop Module (Order No.: VIPA SSM-MD25)
- 20mA Current Loop/ RS232C Combination module (Order No.: VIPA SSM-MD26)
- RS232C Module (Order No.: VIPA SSM-MD22)
- RS422 / RS485 Module (non-floating) (Order No.: VIPA SSM-MD21)
- RS422P Module (floating) (Order No.: VIPA SSM-MD33)
- RS485P Module (floating) (Order No.: VIPA SSM-MD34)
- CENTRONICS Module (Order No.: VIPA SSM-MD24)
- 5V Counter Module (Order No.: VIPA SSM-MD19)
- 24V Counter Module (Order No.: VIPA SSM-MD18)
- Analogue Input Module (Order No.: VIPA SSM-MD40 to -MD44)
- Analague Output Module (Order No.: VIPA SSM-MD45 to -MD49)
- SSI Module (Order No.: VIPA SSM-MD39)

1.4 Deliverable interface building blocks

- 1 Module Slot (Order No.: VIPA SSM-BG41)
- 2 Module Slots (Order No.: VIPA SSM-BG42)
- 3 Module Slots (Order No.: VIPA SSM-BG43)

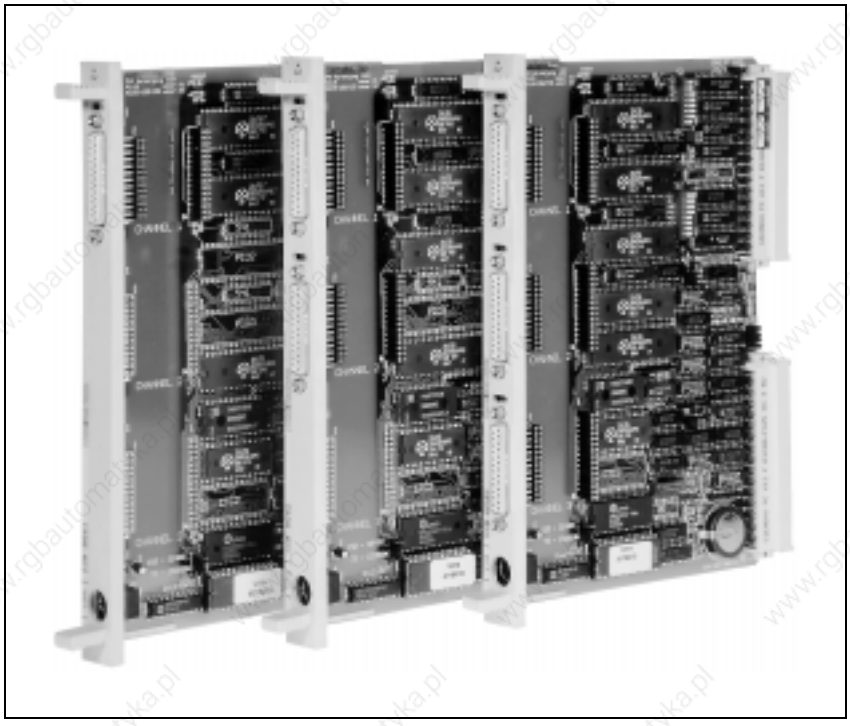


Fig. 1-1: Deliverable interface building blocks

2 Interface modules

2 Interface building blocks

2.1 Principle of data transmission

Data, which is written into the corresponding data channel from the PLC, is recorded by the building block in the appropriate transmission buffer (256 Byte) and from there is sent through the interface. In the other direction (Interface → PLC) the data received is recorded in the appropriate receiving buffer and can be read by the PLC over the data channel.

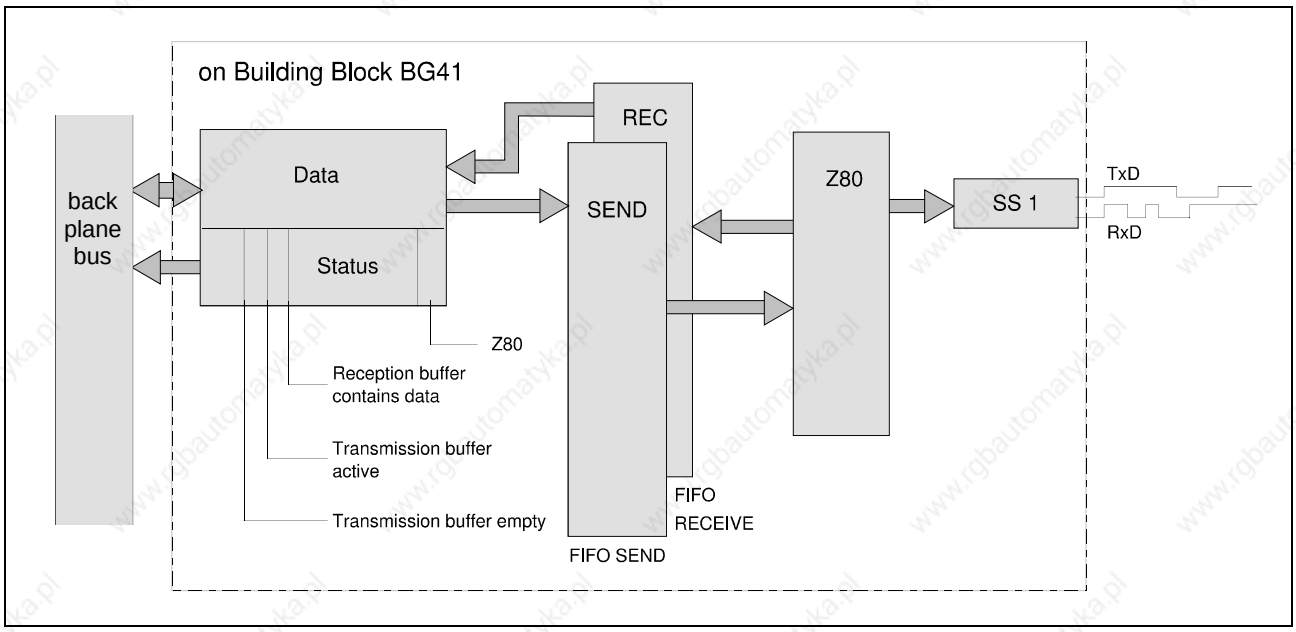
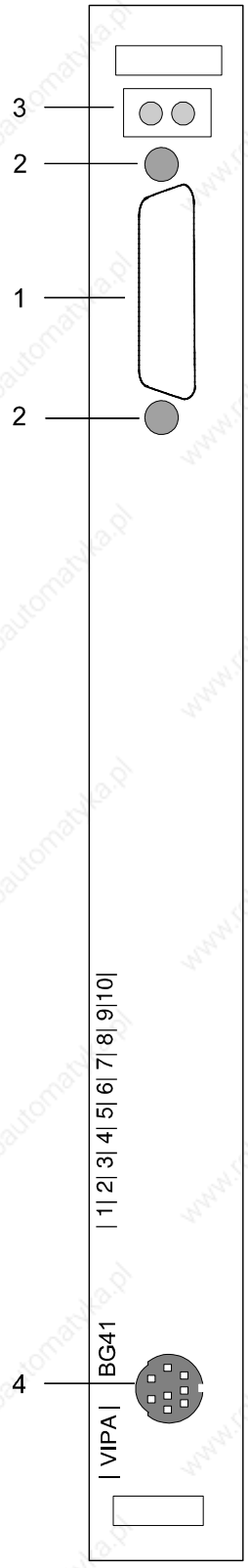


Fig. 2-1: Data transfer via interface (Example Module BG41)

2.2 Building block structure

2.2.1 Building block BG41



The interface building block BG41 is a flat building block in double Euro-format with two 48-pole base plugs in the construction system ES 902. It has a width of 1 1/3 mounting places.

The building block BG41 has 1 interface, which can be adapted to certain physical modes of transmission by means of interface modules.

The front panel of the module consists of:

(1) A 25-pole SubD socket for the interface with

(2) hexagonal threaded bolts for fastening the plugs

(3) Two or four LEDs above the 25-pole SubD socket, which provide information on the operational state and on each connected module (see chapter 3).

(4) An 8-pole mini DIN socket for the service interface.

Fig. 2-2: Front panel of building block BG41

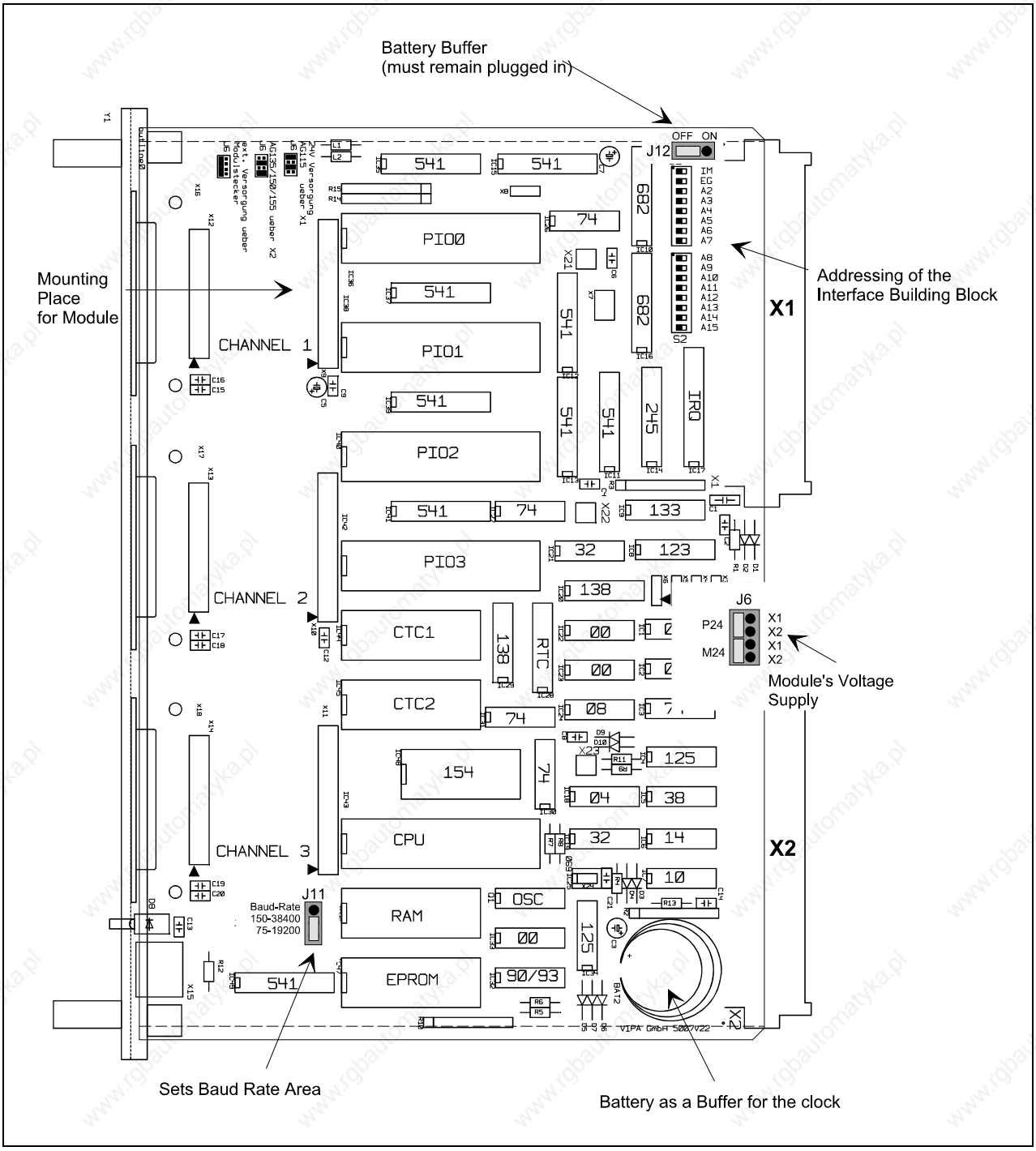
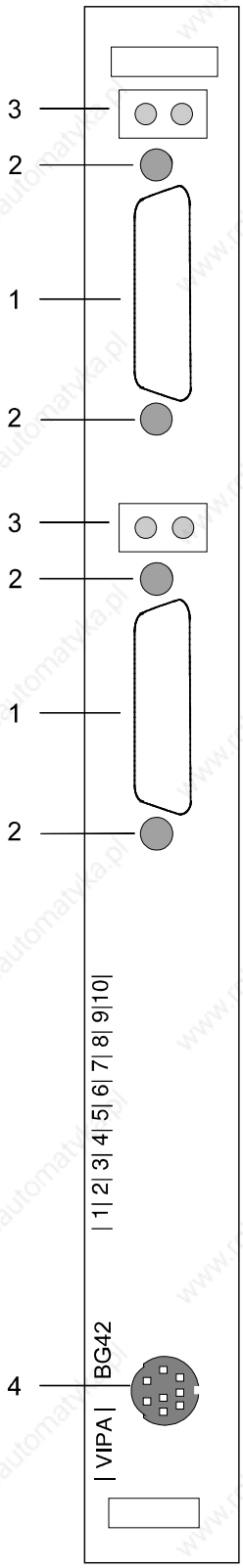


Fig. 2-3: Position of jumpers and DIL switches on BG41

2.2.2 Building block BG42



The interface building block BG41 is a flat building block in double Euro-format with two 48-pole base plugs in the construction system ES 902. It has a width of 1 1/3 mounting places.

The building block BG42 has 2 interfaces, which could be adapted to certain physical modes of transmission by means of interface modules.

The front panel of the module consists of:

(1) Two 25-pole SubD socket for the two interfaces with

(2) hexagonal threaded bolts for fastening the plugs

(3) Two or four LEDs above the 25-pole SubD sockets, which provide information on the operational state and on each connected module (see chapter 3).

(4) An 8-pole mini DIN socket for the service interface.

Fig. 2-4: Front panel of building block BG42

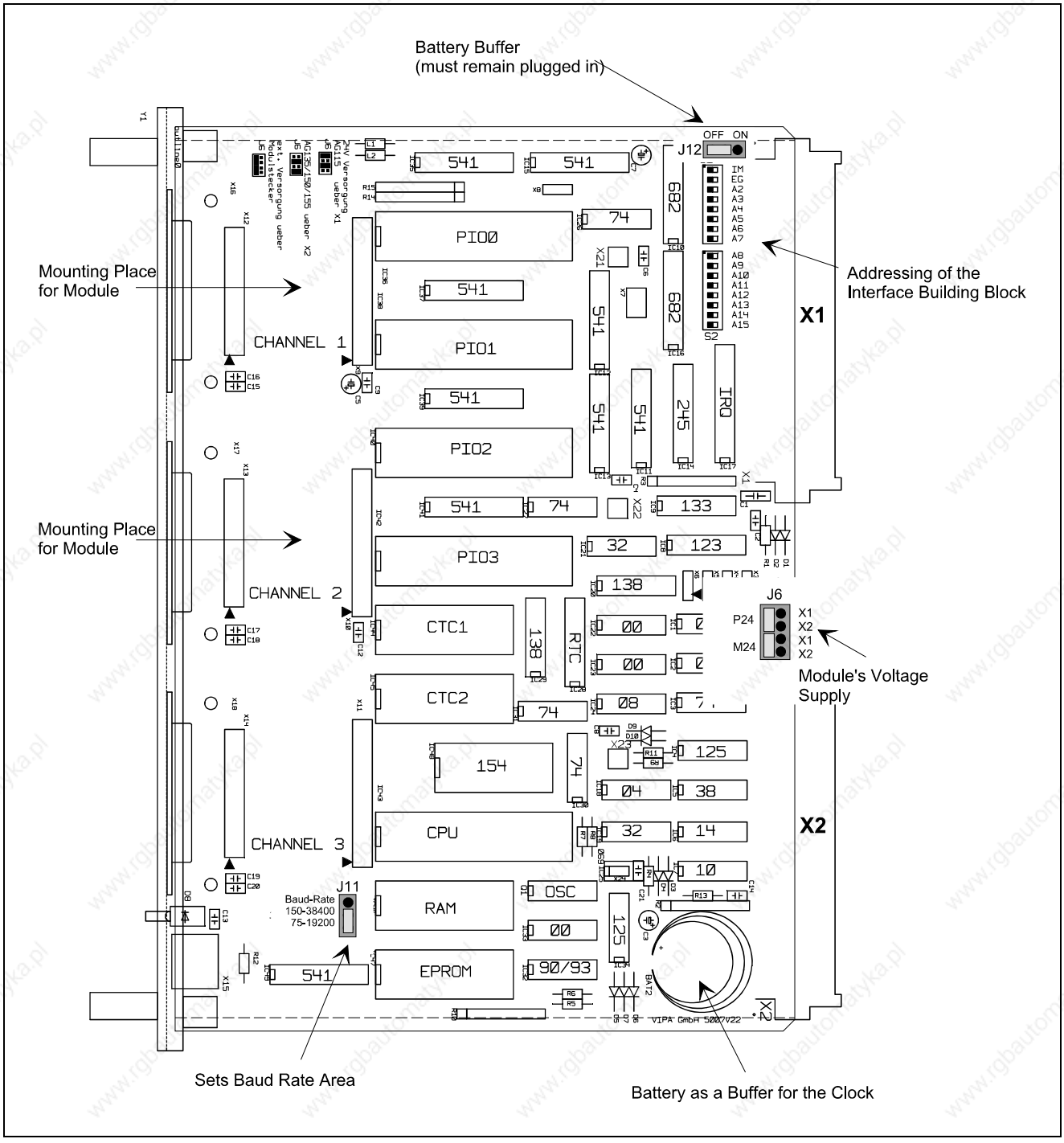
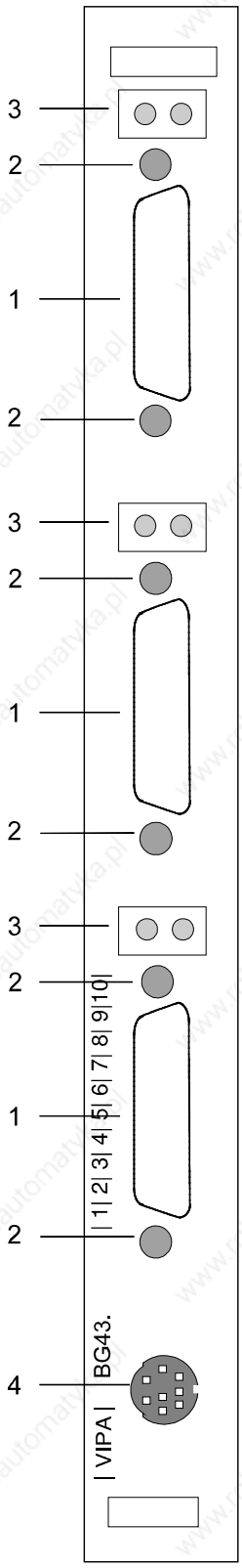


Fig. 2-5: Position of Jumpers and DIL switches on BG42

2.2.3 Building block BG43



The building block module BG41 is a flat building block in double Euro-format with two 48-pole base plugs in the construction system ES 902. It has a width of 1 1/3 mounting places.

The building block BG42 has 3 interfaces, which could be adapted to certain physical modes of transmission by means of interface modules.

The front panel of the module consists of:

(1) Three 25-pole SubD socket for the three interfaces with

(2) hexagonal threaded bolts for fastening the plugs

(3) Two or four LEDs above the 25-pole SubD sockets, which provide information on the operational state and on each connected module (see chapter 3).

(4) An 8-pole mini DIN-socket for the service interface.

Fig. 2-6: Front panel of building block BG43

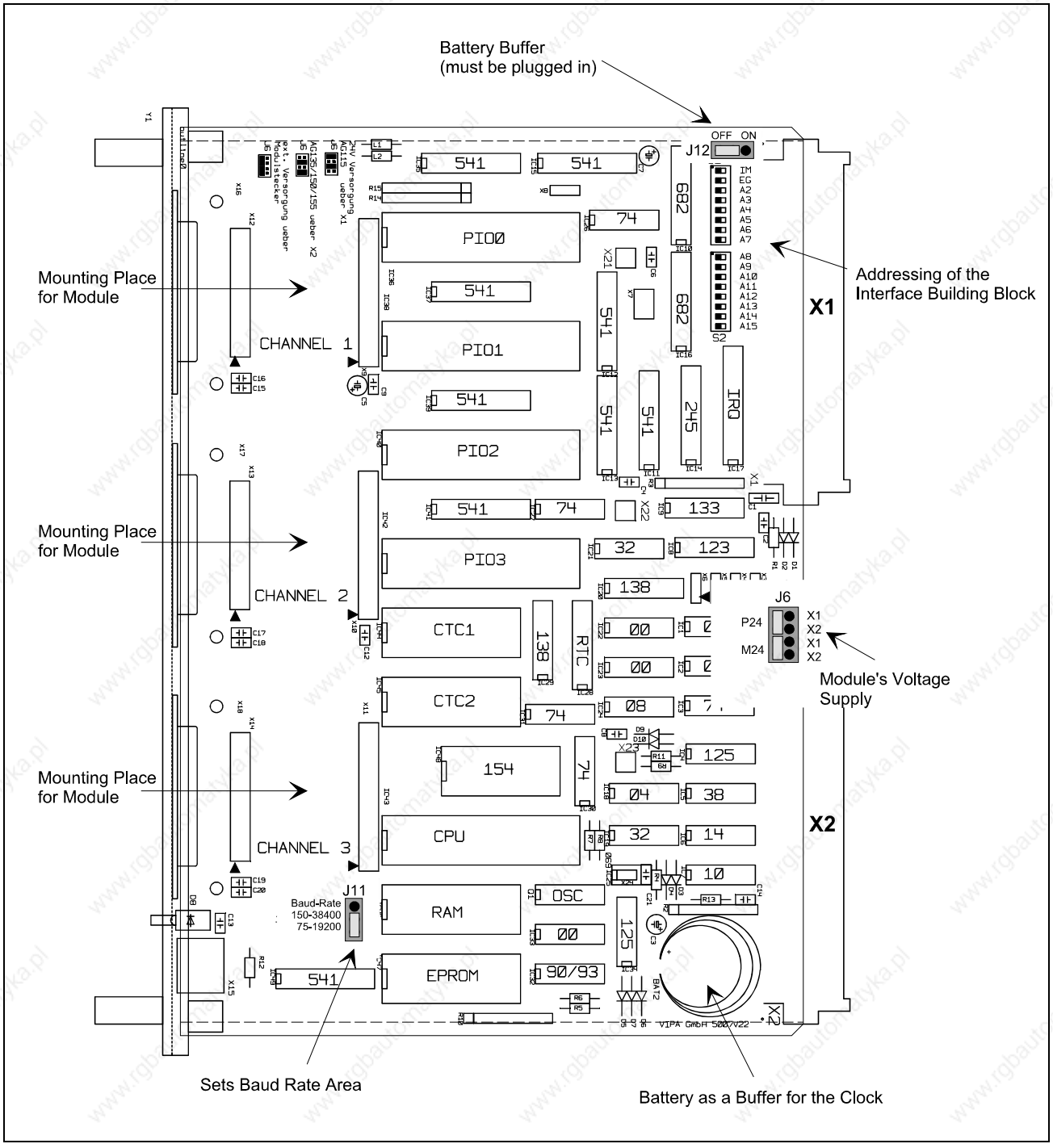


Fig. 2-7: Position of jumpers and DIL switches on BG43

2.3 Allocation of the backplane connector

2.3.1 Backplane connector X1

	d	b	z
2	n.c.	M	+5V
4	UBAT	PESP	n.c.
6	ADB12	ADB00	/CPKL
8	ADB13	ADB01	/MEMR
10	ADB14	ADB02	/MEMW
12	ADB15	ADB03	/RDY
14	n.c.	ADB04	DB0
16	n.c.	ADB05	DB1
18	n.c.	ADB06	DB2
20	n.c.	ADB07	DB3
22	n.c.	ADB08	DB4
24	n.c.	ADB09	DB5
26	n.c.	ADB10	DB6
28	n.c.	ADB11	DB7
30	+24V	BASP	M24V
32	/BASPA	M	n.c.

Tab. 2-1: Allocation of the backplane connector X1

2.3.2 Backplane connector X2

	d	b	z
2	n.c.	M	+5V
4	n.c.	n.c.	n.c.
6	n.c.	n.c.	n.c.
8	n.c.	n.c.	n.c.
10	n.c.	n.c.	n.c.
12	n.c.	n.c.	n.c.
14	n.c.	n.c.	n.c.
16	n.c.	n.c.	n.c.
18	n.c.	n.c.	n.c.
20	n.c.	n.c.	n.c.
22	n.c.	n.c.	n.c.
24	n.c.	n.c.	n.c.
26	n.c.	n.c.	n.c.
28	n.c.	n.c.	n.c.
30	n.c.	n.c.	M24V
32	n.c.	M	+24V

Tab. 2-2: Allocation of the backplane connector X2

2.4 Block diagram of firmware

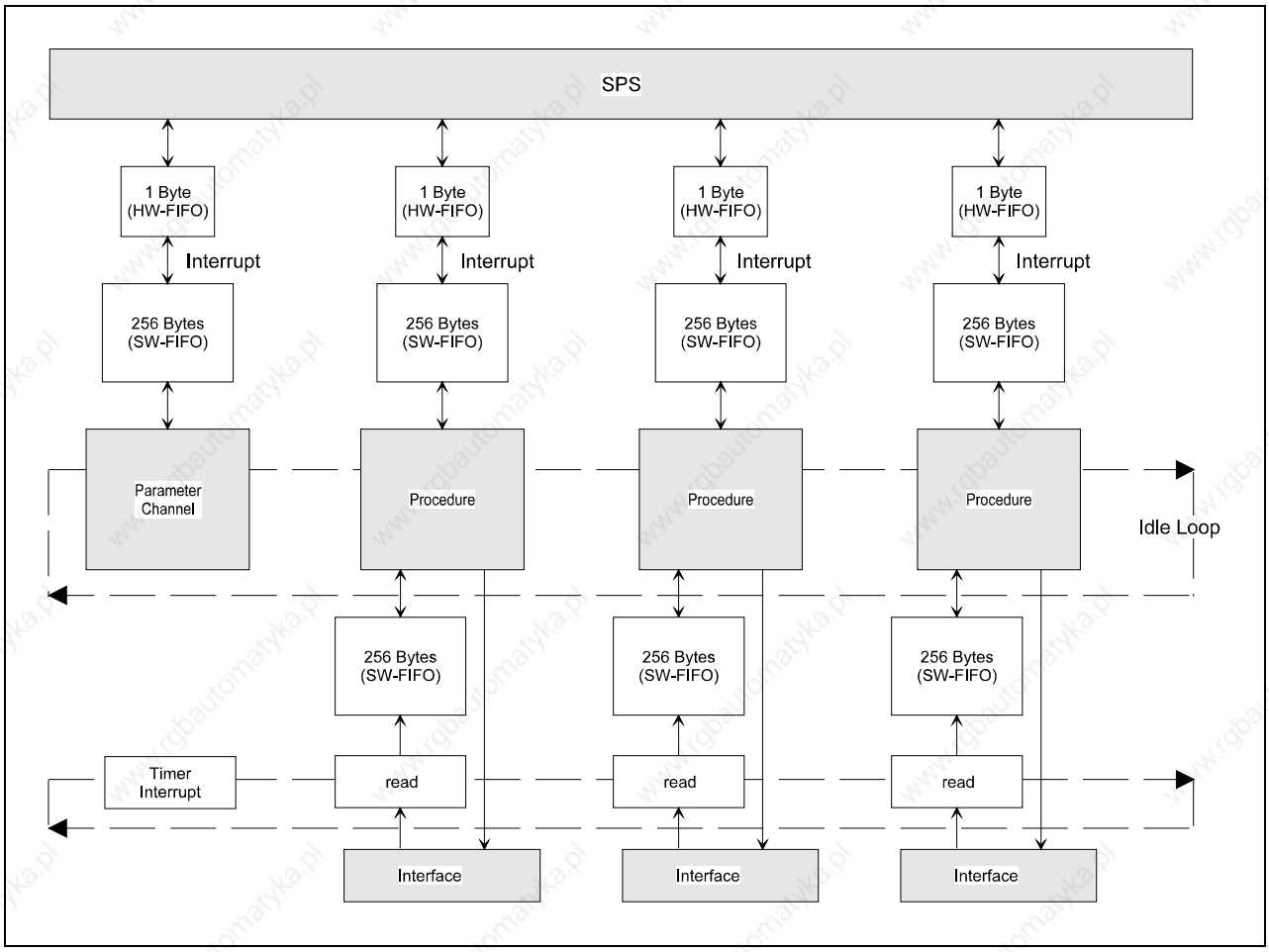


Fig. 2-8: Firmware block diagram BG43

2.5 Firmware's memory distribution

The Z80 processor on communication processors has an address range of 64 KByte. This is distributed as follows:

0000 : : 7FFF	Firmware-EPROM (32 KByte)
8000 : : FFFF	RAM (32 KByte)

Fig. 2-9: Firmware's memory distribution

3 Interface module

3 Interface modules

3.1 General

The modular communication processor SSM is an intelligent carrier module, upon which a series of input, output and communication modules can be equipped.

Depending on the module, up to 3 interface modules can be optionally combined.

The communication processor SSM is suitable for:

- Communication with automation equipment and computers
- Connection with keyboards, terminals or printers
- Time and clock acquisition
- Recording the time standard of PTB-Braunschweig
- Recording and output of analogue process quantities.

3.1.1 Voltage supply 5V and 24V

The 5V, which are supplied by the back plane bus, are generally sufficient for the power supply of the modules used.

The 20mA current loop module and/or the 20mA current loop combination module require a voltage supply of 24 V if they are actively used in a 20mA current loop with internal power supply. The SSI module's encoder can also be supplied with 24 V. Guidelines on voltage supply can be found in chapter 5.

3.1.2 LED indicators on the modules

The interface modules are equipped with LEDs. It is possible to determine the type of module or inserted building block being used on the interface by the colour of the LED.

In this chapter the significance of the LED indicators are described in connection with the respective modules.

Type of Module	Order No.	LED 1	LED 2	LED 3	LED 4
20mA Current Loop	MD25	red (receive)	red (send)		
20mA Current Loop/RS232C Combination	MD26	red (receive)	red (send)		
RS232C	MD22	green (receive)	green (send)		
RS422 / RS485	MD21	yellow (receive)	yellow (send)		
RS422P	MD33	yellow (receive)	yellow (send)		
RS485P	MD34	yellow (receive)	yellow (send)		
CENTRONICS	MD24	red (error)	green (error-free)		
SSI	MD39				
5V/24V Counter	MD18	green (power on)	yellow (Z3:CY/BW)	yellow (Z2:CY/BW)	yellow (Z1:CY/BW)
Analogue Input	MD40-MD44				
Analogue Output	MD45-MD49				
DCF77 Antenna	MD36	green (receive)	red (status)		

Tab. 3-1: LED-indicators on the modules

3.1.3 Selection table: function - module - function blocks

Functions	Required or possible module	Initialization (OB20, OB21, OB22)	Cycle (OB1)
Serial functions			
No procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB100, FB101, FB5, FB15	FB3, FB4
STX/ETX procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB100, FB101, FB5, FB15, FB40	FB43, FB44
3964(R) procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB30, FB37	FB31-33
3964(R) with RK 512 procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB20, FB27	FB21-26
[1]Mettler Balance	MD22, MD25, MD26		FB17, FB18, FB7, FB8
Hohner Encoder	MD34	FB5	FB19
Parallel functions			
	MD24	FB100, FB101, FB5	FB3
Additional functions			
SSI module	MD39	FB100, FB101	FB45
Counter	MD18, MD19	FB100, FB101, FB12	FB10, FB11
Analogue input	MD40, MD41, MD42, MD43,	FB50, FB51, FB54	FB52, FB55, FB56, FB57, FB58
Analogue output	MD45, MD46, MD47, MD48, MD49	FB50, FB51, FB54	FB53, FB56
Measurement of temperature (PT100)	MD44	FB50, FB51, FB54	FB52, FB55, FB56, FB57, FB58
Clock			FB102
DCF77 antenna	MD36	FB100, FB101	FB102

Tab. 3-2: Selection table: function - module - function blocks

[1] The use of the Mettler Balance is described in manual HB30.

3.2 Functional description and allocation of terminal pins

3.2.1 20mA current loop-modul (MD25)

3.2.1.1 General

The RS232C interface can no longer be sufficiently relied upon in an environment where high interference levels are expected. The 20mA current loop interface was therefore developed. With this interface a logical "1" is implemented through a current of 20mA and a "0" through 0mA. The respective currents are generated in an active suscriber and are led through a loop back to the sender. Only two circuit pairs are therefore required.

Point-to-point or bus connections can be realised with the 20mA current loop module. The module contains two line current loops for transmission and reception. In bus operations the module in a voltage-free state connects all loops, in order to prevent the entire system from being impaired.

In its delivery state the interface module 20mA current loop is set to the internal voltage supply through the back plane bus. If the external voltage supply is switched to the 20mA current source, the jumpers X3 and X5 must be replugged.

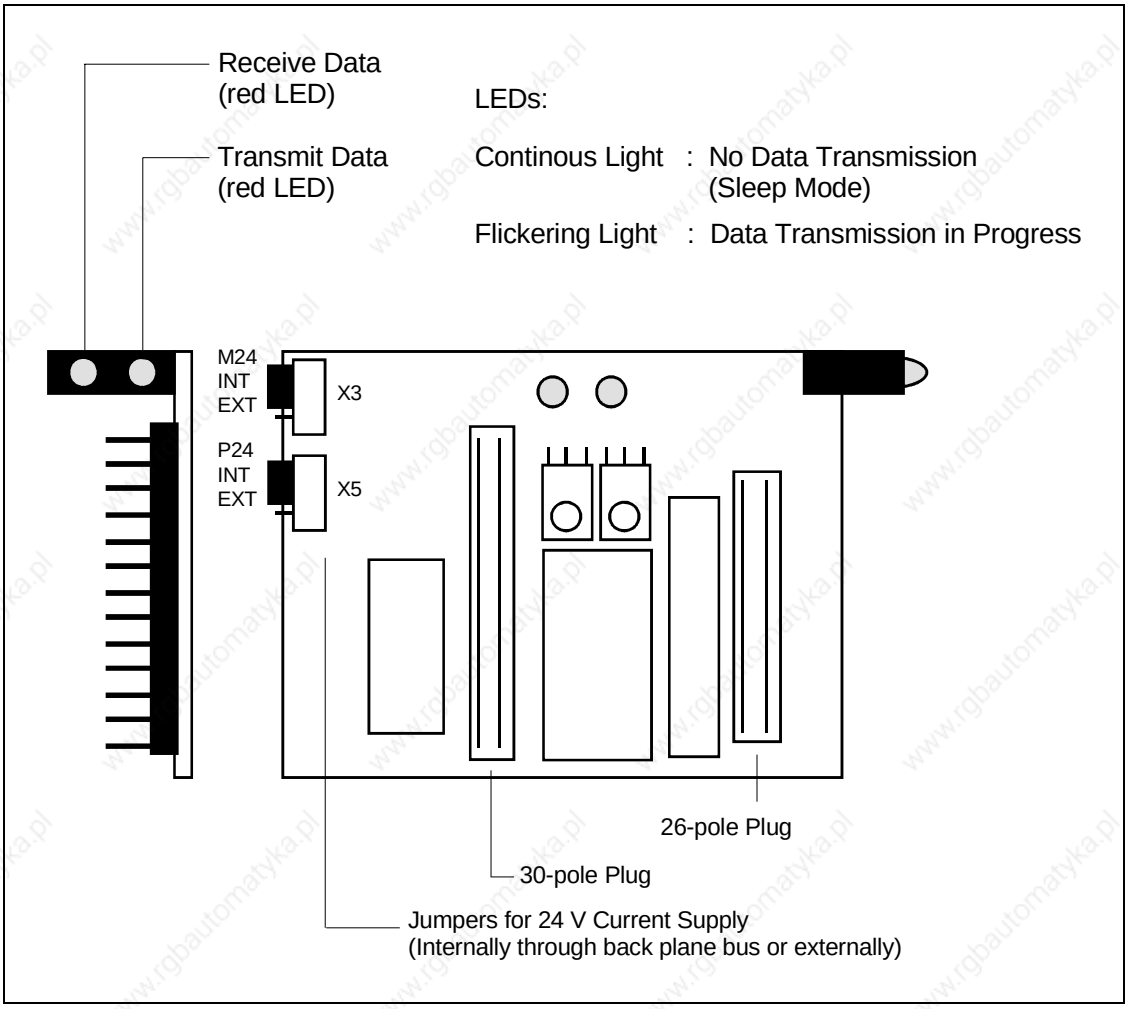


Fig. 3-1: Structure of the 20mA current loop-moduls

3.2.1.2 Allocation of the 20mA current loop module's terminal pins

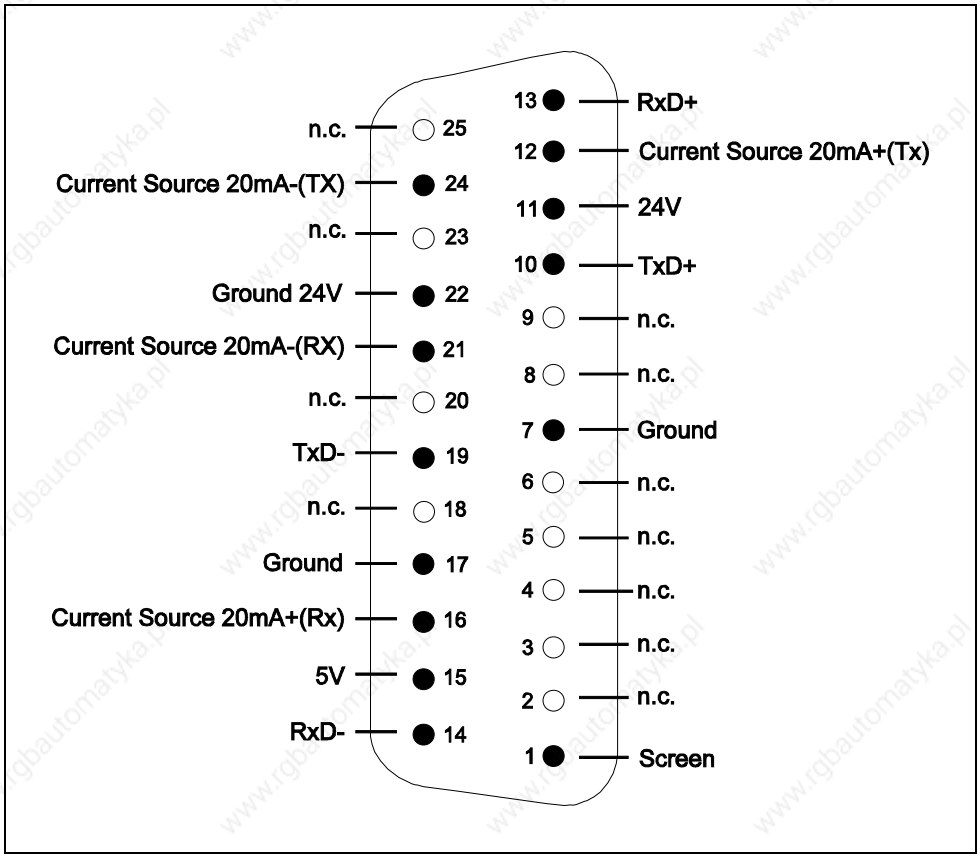


Fig. 3-2: Allocation of terminal pins when operating 20mA current loop modules

If the 20mA current loop interface module or the combination module are used as a 20mA current loop module, the interface can be operated as a passive or active part of the transmission link. The type of contacts for peripheral equipment varies accordingly.

It is possible to obtain 24 V from pin 11 (+24 V) and pin 22 (ground), if this voltage is available in the connected PLC.

When using the 20mA current loop module at a baud rate of 38,4 Kbaud, a Twisted Pair cable is required from a cable length of more than 50m, in order to avoid the other wire pair from being influenced.

3.2.1.3 Voltage supply for a 20mA current loop

If the 20mA current loop module or the 20mA current loop combination module are operated as the active part of a current loop, they can be supplied internally or externally from the 20mA current sources found on the module:

- internally through the back plane bus (base plug X1 or X2)
- externally through the 25-pole SubD socket

The current sources are to be supplied with 10 to 36 V (typically 24 V) and produce the following characteristic curves:

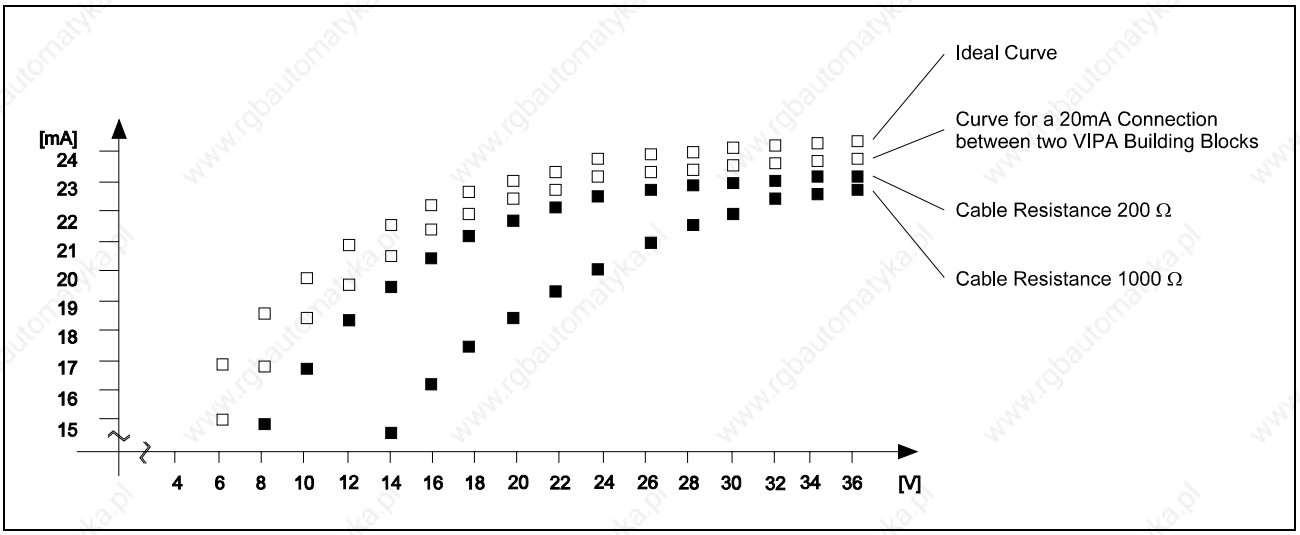


Fig. 3-3: Characteristic curves of the current sources in relation to the applied voltage

3.2.1.4 An active 20mA current loop interface

The serial interface functions as the active part of the transmission link, the peripheral device as the passive part.

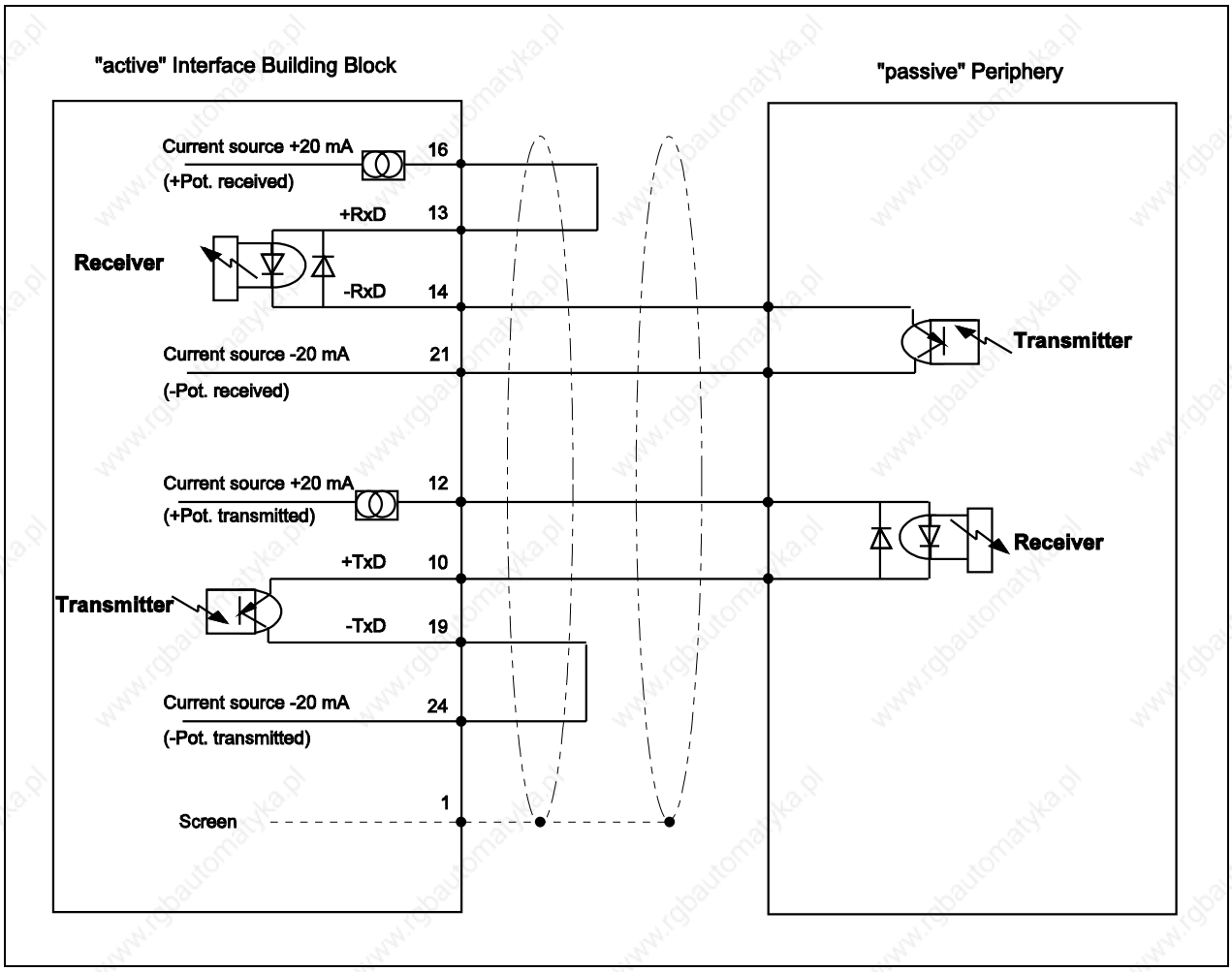


Fig. 3-4: An active 20mA current loop interface

Observe shielding! See construction guidelines in chapter 5!

The jumpers J6 on the mother board and X3 and X5 on the module enable the current sources for the 20mA current loop to be supplied with 24 V from the back plane bus.

3.2.1.5 A passive 20mA current loop interface

Peripheral equipment (e.g. the printer PT 88), as the active part, are in the position to take over the supply of the 20mA-connection. In this manner, the serial interface can be operated as the passive part. A 24V-supply will therefore not be needed for the operation of the module.

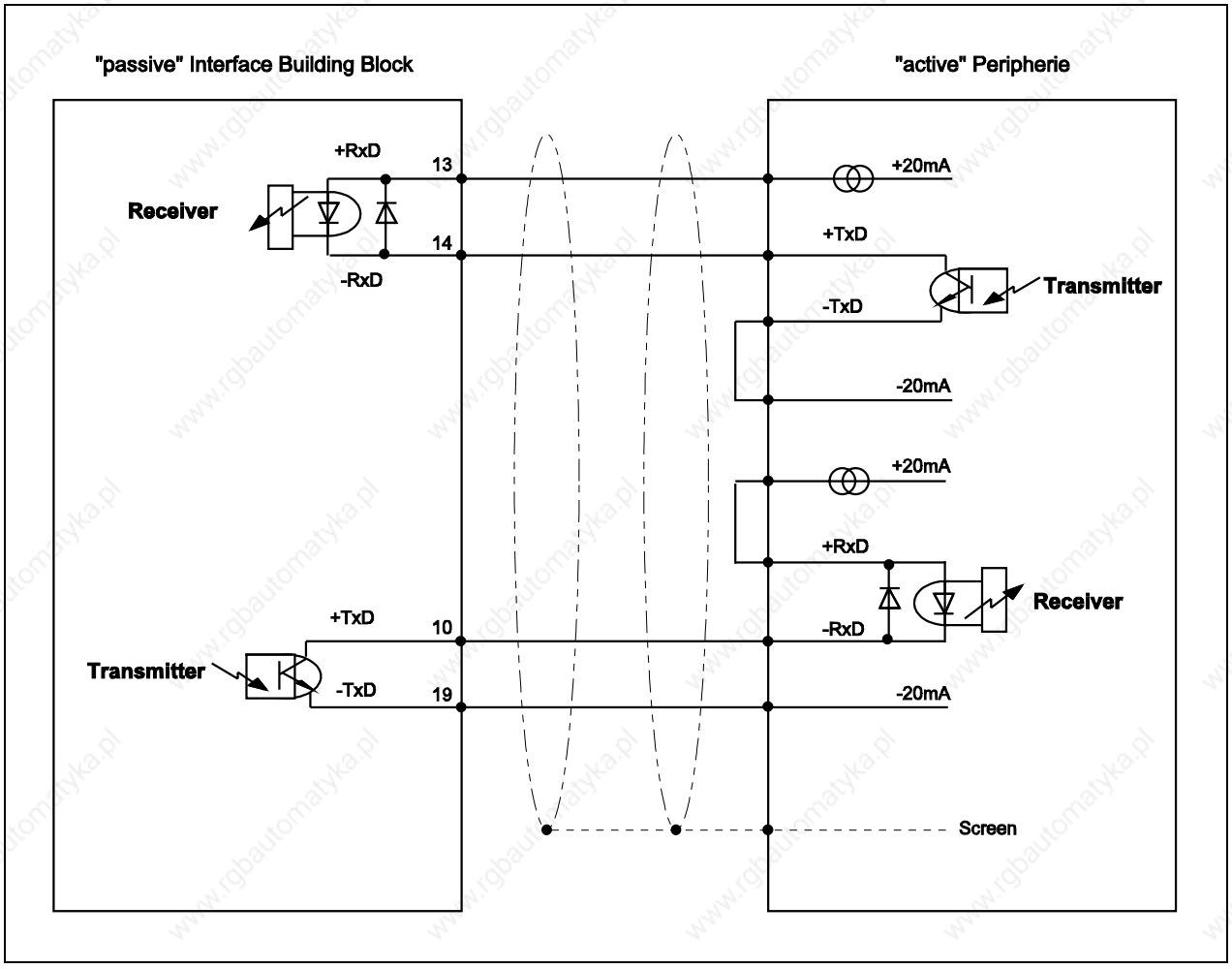


Fig. 3-5: A passive 20mA current loop interface

3.2.2 RS232C module (MD22)

The RS232C is defined for data transmission for a maximum of 15m up to 38,4baud. The communication occurs by data, signal and control lines .

The RS232C module serves the point-to-point connection on the basis of the RS232C standards. No presetting has to be observed when operating the RS232C interface module.

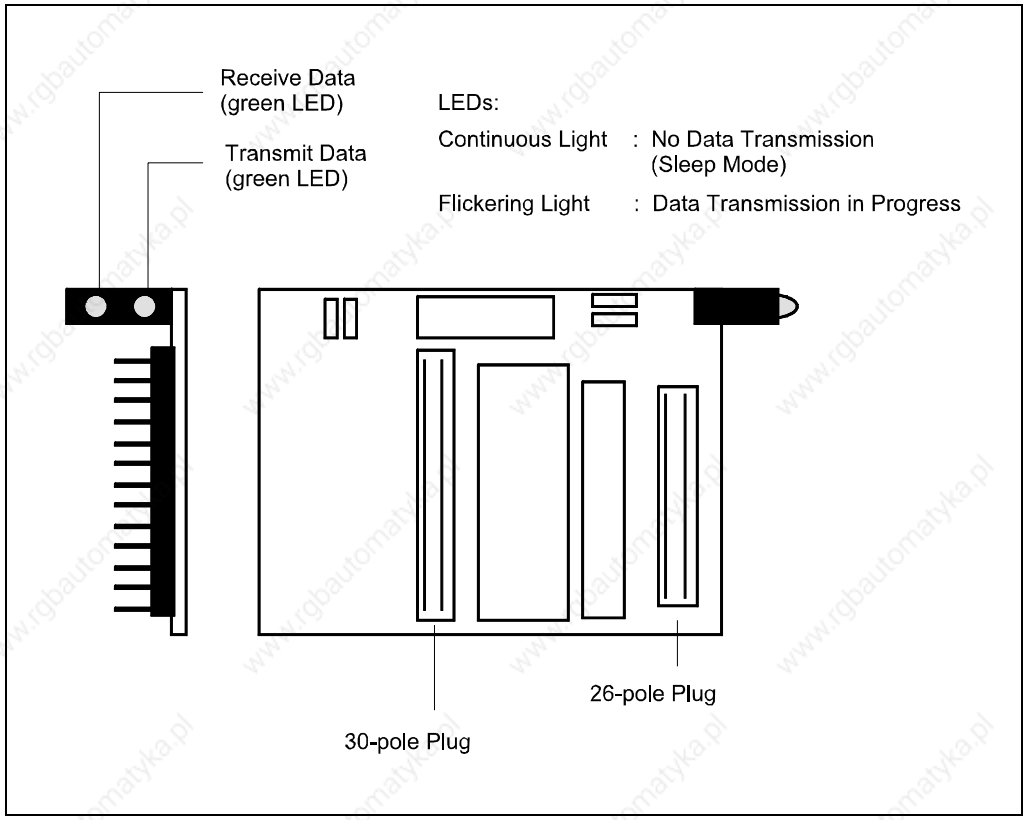


Fig. 3-6: The structure of the RS232C modules

3.2.2.1 Allocation of the RS232C module's terminal pins

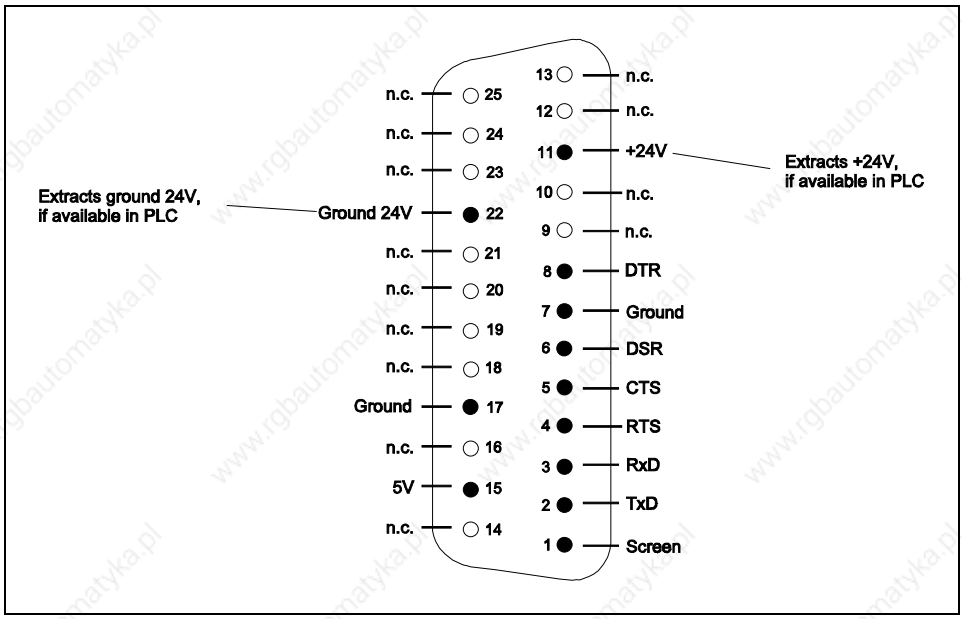


Fig. 3-7: Allocation of terminal pins when operating RS232C modules

3.2.2.2 Connection with a RTS/CTS handshake

It is possible to establish a connection through the RS232C interface with or without a handshake.

When operating an RS232C interface, a hardware handshake is used through the signals RTS and CTS. These handshake signals must be operated by peripheral equipment.

If the employed peripheral equipment is operated with a DTR/DSR handshake instead of with the RTS/CTS handshake as is here the case, these signals must be used to operate the serial interface's RTS/CTS pins.

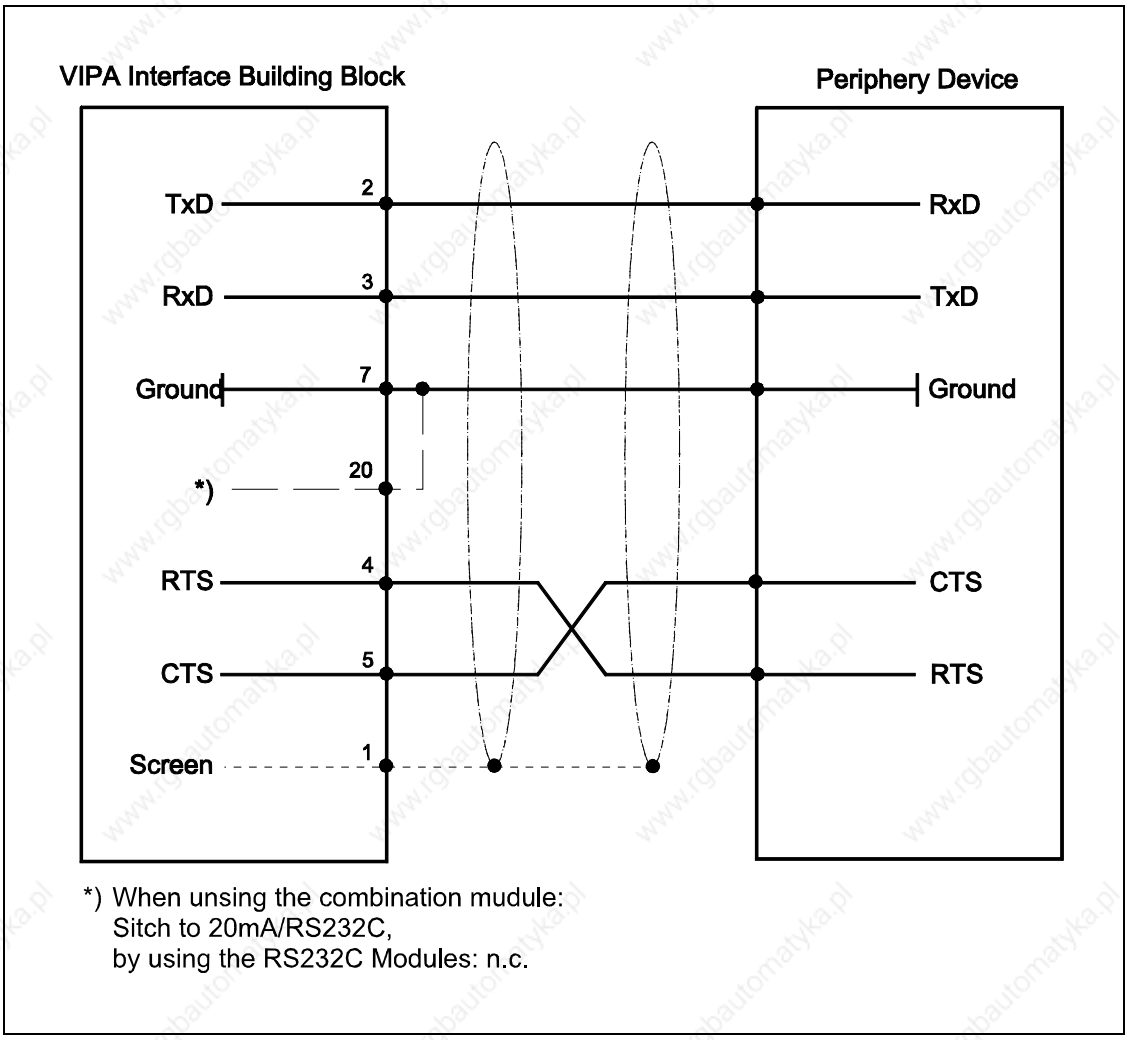


Fig. 3-8: RS232C interface : connection with a RTS/CTS handshake

Observe screening! See construction guidelines!

3.2.2.3 Connection without a handshake

If the peripheral equipment does not have a handshake signal at its disposal, it is possible to operate without a hardware handshake. In this case according to VIPA, the RTS/CTS should always be bridged!

With this type of connection it must be insured, that no character is lost. This can be achieved by observing the following points:

- a low baud rate or a quick recording of characters by the peripheral equipment
- a quick disposal of reception buffers on the PLC side

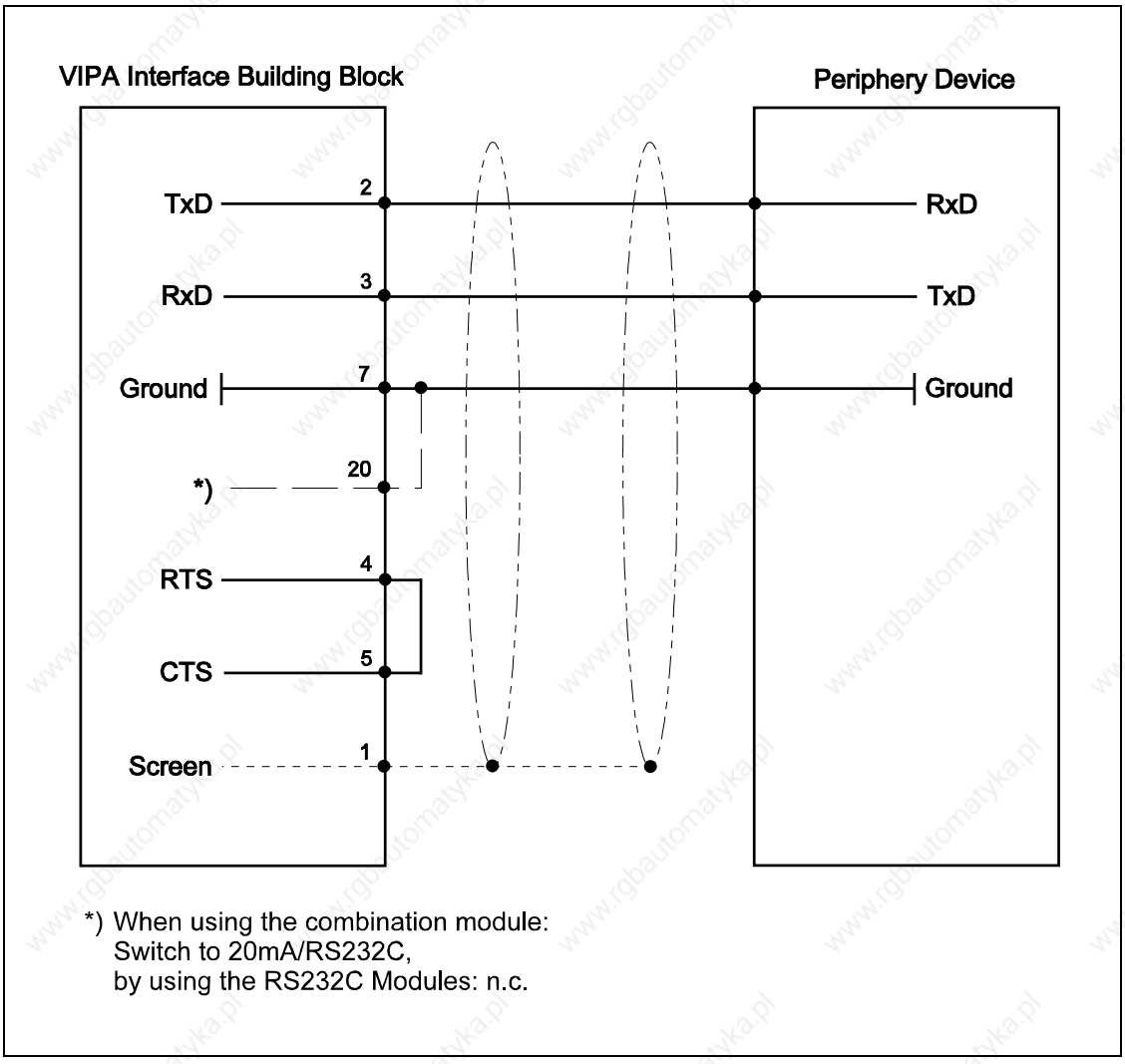


Fig. 3-9: RS232C interface: connection without a handshake

Observe screening! See construction guidelines!

3.2.3 20mA current loop/RS232C combination module (MD26)

3.2.3.1 General

The combination module serves as a point-to-point or a bus connection and has two line current loops for transmission and reception at its disposal. In a bus operation the module, in a voltage-free state, connects all loops in order to prevent the entire system from being impaired.

When operating a RS232C, only a point-to-point connection is possible.

The 20mA current loop/RS232C combination module in its delivery state is so adjusted, that when operating as a 20mA current loop module the internal voltage supply is carried out by the back plane bus. If the external voltage supply (25-pole SubD sockets) is switched to the 20mA current source, the plug connectors X3 and X5 must be replugged.

The LED indicator is only effective, when the combination module is used as a 20mA current loop module.

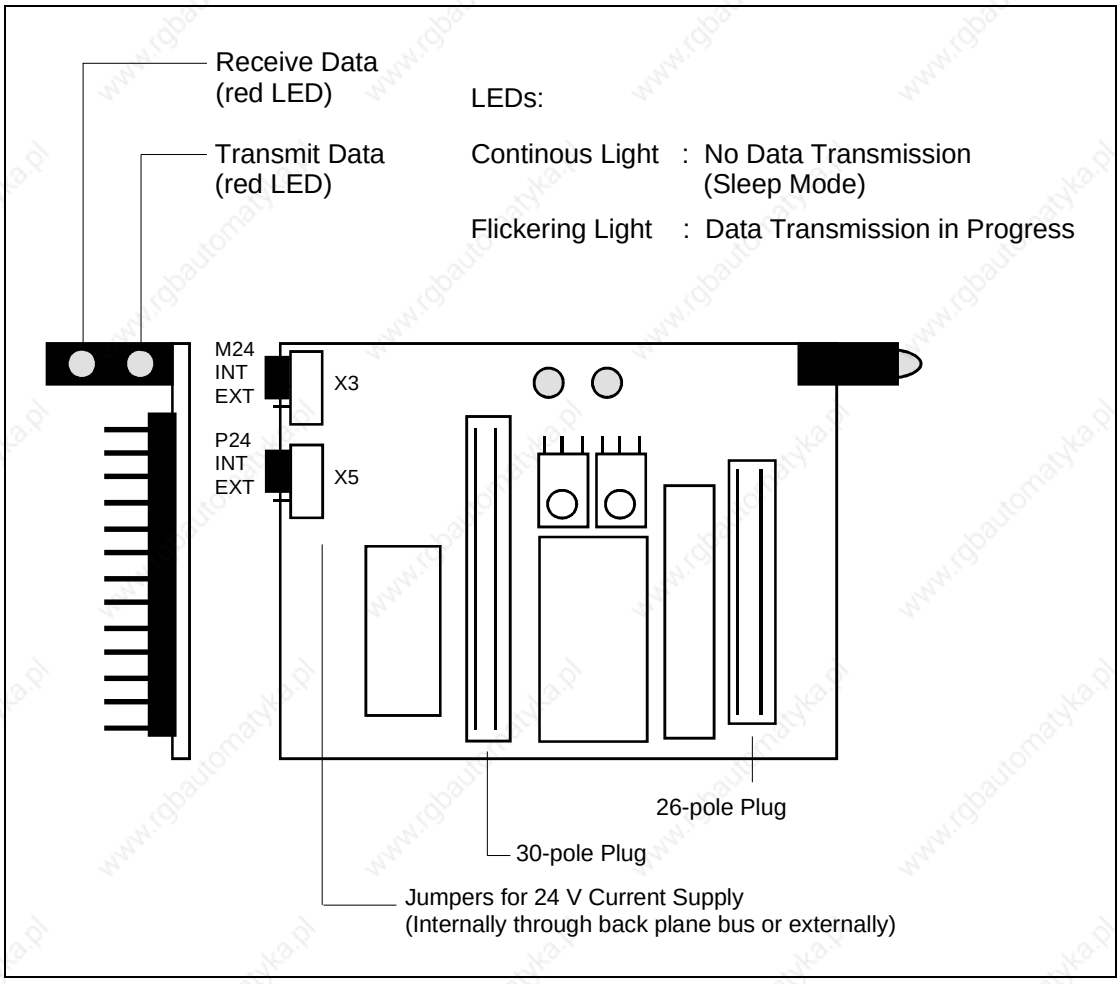


Fig. 3-10: Structure of a 20mA current loop/RS232C combination module

When using a 20mA current loop module at a baud rate of 38,4 Kbaud, a twisted pair cable is required from a cable length of more than 50m, in order to avoid the other wire pair from being influenced.

3.2.3.2 Allocation of terminal pins when using a 20mA current loop module

Details on the 20mA current loop module can be found in chapter 3.2.1.

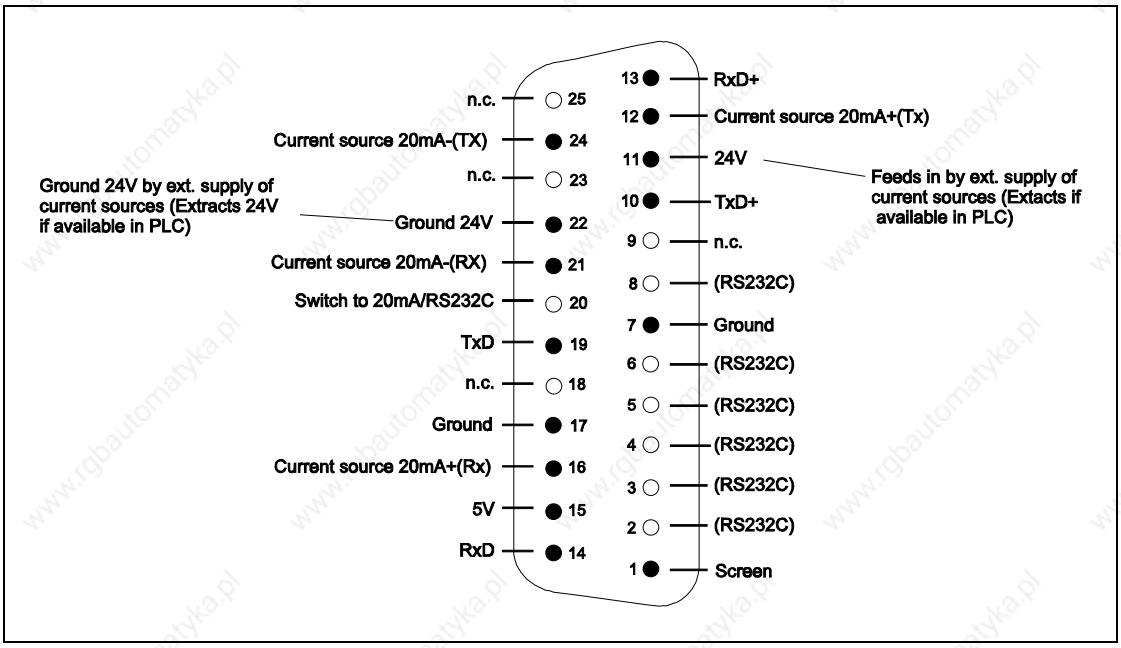


Fig. 3-11: Allocation of terminal pins when using a 20mA Current Loop Module

3.2.3.3 Allocation of terminal pins when using a RS232C-module

Details on the RS232C module are to be found in chapter 3.2.2..

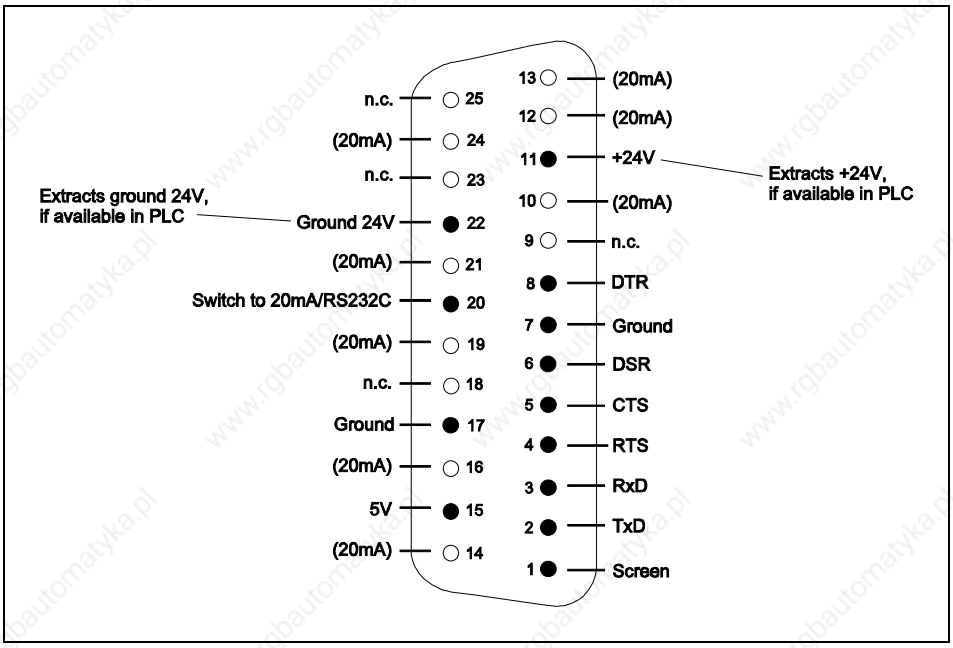


Fig. 3-12: Allocation of terminal pins when using a RS232C-module

The **switchover** between both forms of operation occurs through pin 20 on the 25-pole SubD socket:

- Pin 20 on ground → Operation as a RS232C module
- Pin 20 open → Operation as a 20mA Current Loop module

3.2.4 RS422/RS485 module (MD21)

3.2.4.1 General

The RS422 interface is suitable for transmission lengths of up to 1200m. In using a four-wire line, data can be simultaneously sent and received. As a result of the symmetrical transmission a higher interference immunity is achieved. Coupling on the grounding conductor does not influence the signal. One advantage in comparison to the RS232C interface is its bus capabilities. It can be connected to up to 16 suscribers.

The RS485 interface was developed, in order to enable a higher number of bus suscribers to be connected to the bus operation. Up to 32 suscribors can in this case be connected. Similar to the RS422, the transmission is symmetrically constructed, but here one has a higher interference immunity. By using repeaters (signal amplifiers) the distance can be doubled.

The RS422/RS485 module serves the point-to-point or the multipoint connection on the basis of the RS422/RS485 norm. A load resistance of 100Ω can additionally be connected. The module is non-floating.

When using the RS422/RS485 module in its delivery state the jumper for the 100Ω resistance is plugged into the OFF position.

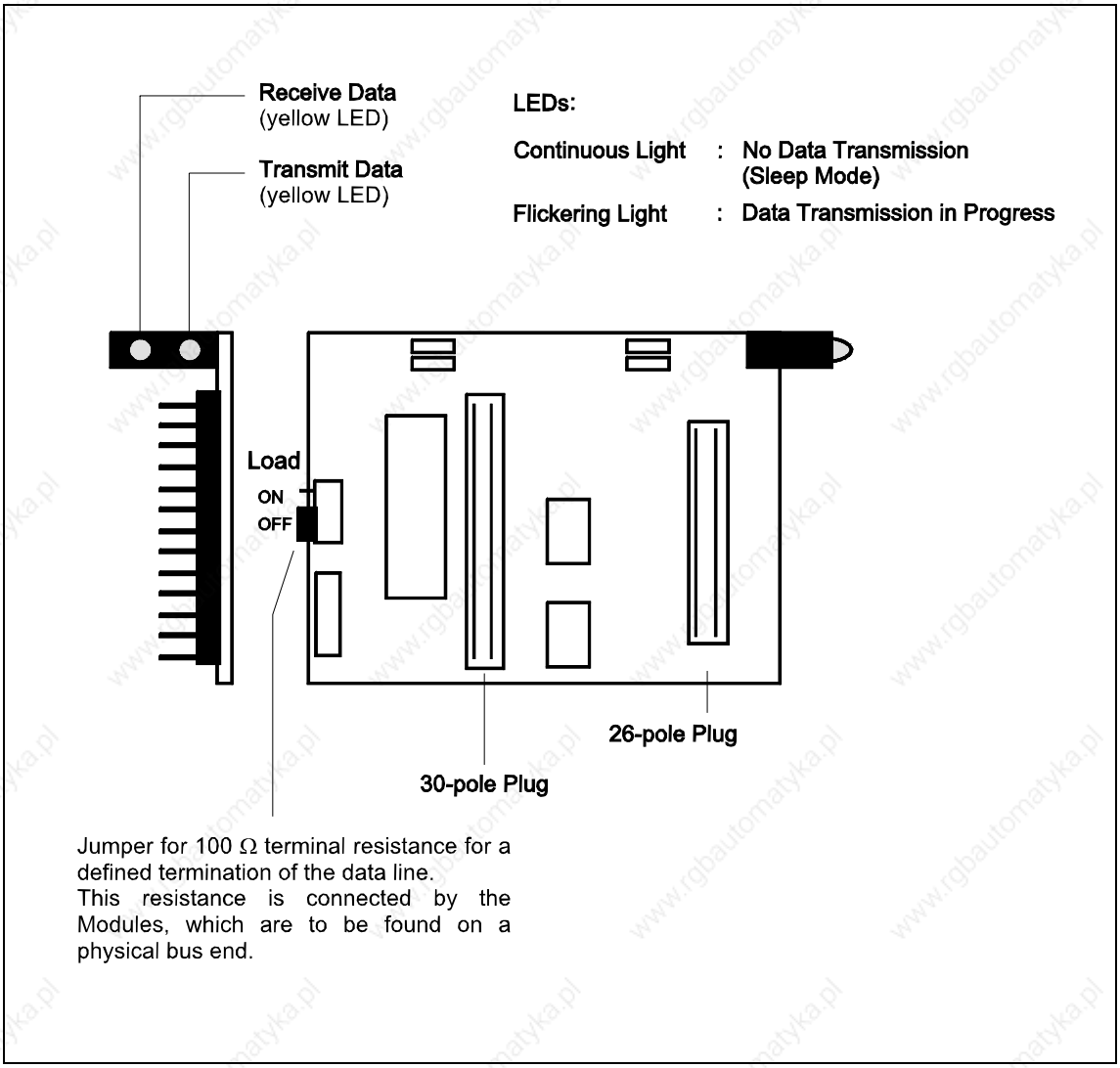


Fig. 3-13: Structure of the RS422/RS485 module

3.2.4.2 Allocation of the RS422/RS485 module's terminal pins

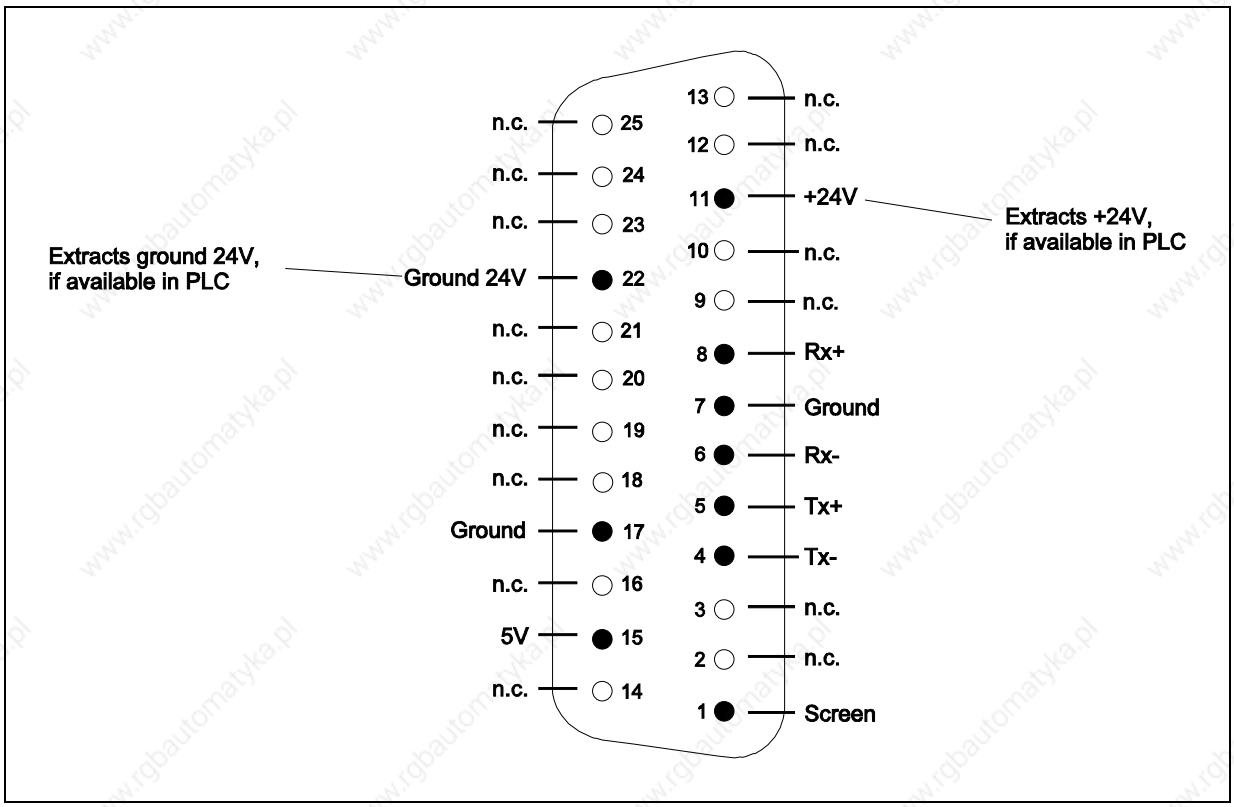


Fig. 3-14: Allocation of terminal pins when using a RS422/RS485-moduls

It is possible to obtain 24 V from pin 11 (+24 V) and pin 22 (ground), if this voltage is available in the connected PLC.

3.2.4.3 Point to point connection

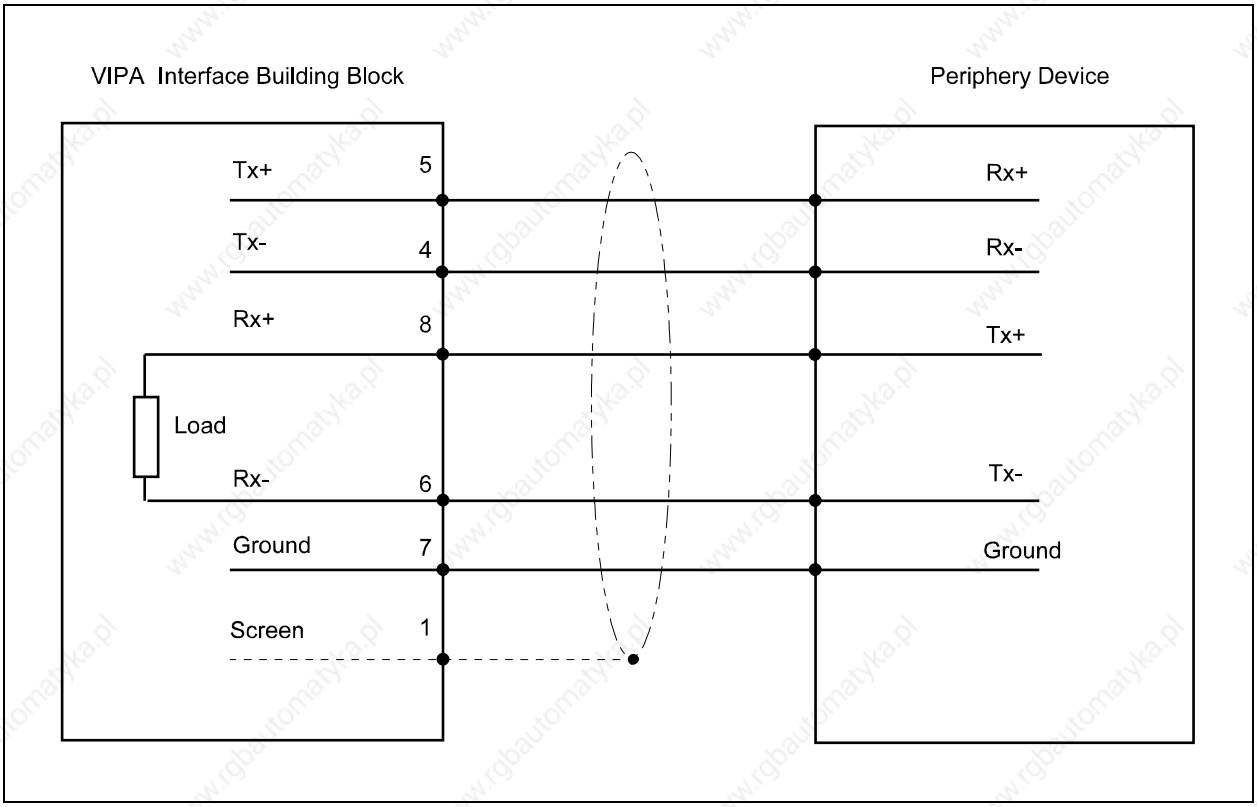


Fig. 3-15: Point to point connection with a RS422/RS485-module

Two twisted-pair wires, which have to be particularly shielded, are to be used for the TX and RX links.

A load resistance of 100 Ω can be switched between the Rx links by means of the plug connector J1 on the RS422/RS485 module. This is necessary for large cable lengths or high transmission speeds.

3.2.4.4 Two-wire connection

By linking Rx+ and Tx+ or Rx- and Tx- it is possible to produce a two-wire connection. The RS485 interface uses the two-wire connection.

Due to the links, the sent data is immediately recoupled. The recoupled data is to be filtered by using software. It is advisable to use the RS485P module for the operation of the RS485, as in this module the receiver is switched off at transmission.

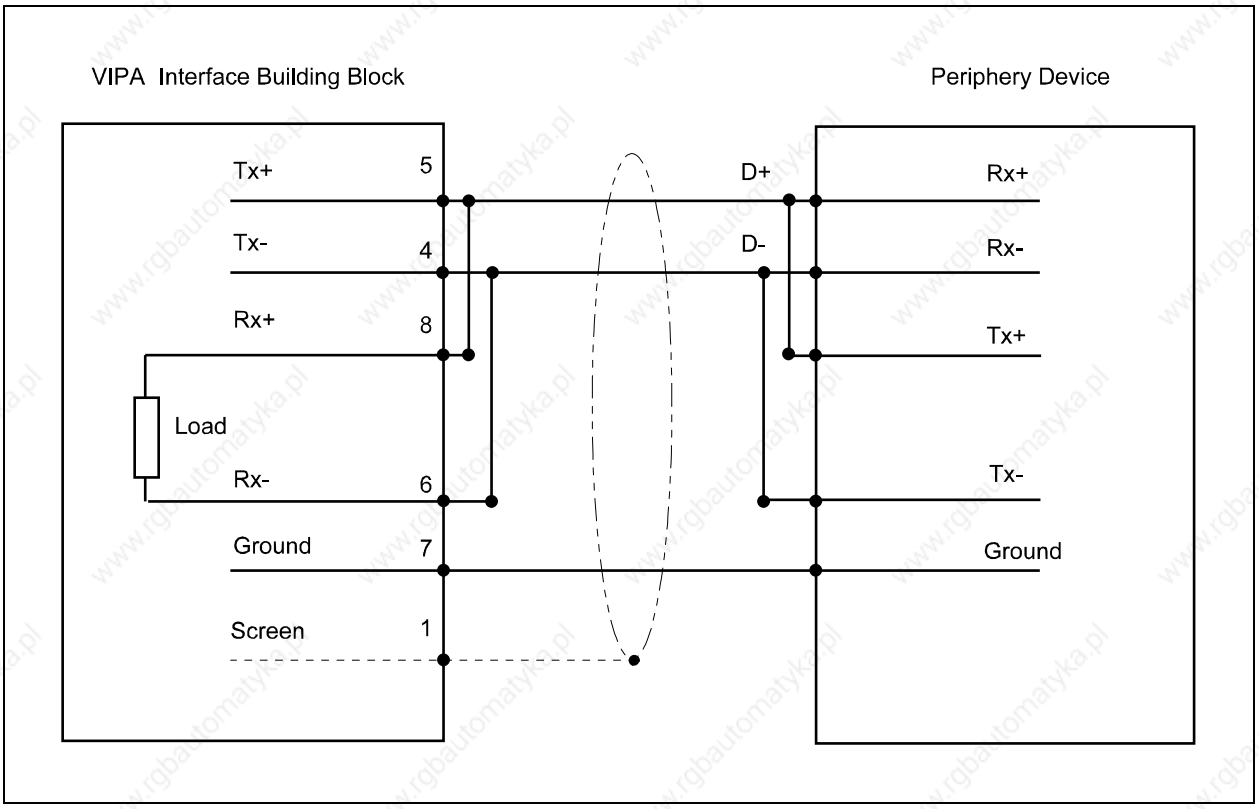


Fig. 3-16: Two-wire connection with a RS422/RS485 module

3.2.4.5 Multidrop connection (Multipoint connection)

The multidrop connection should be selected, if several external pieces of equipment are to be connected to the interface as receivers.

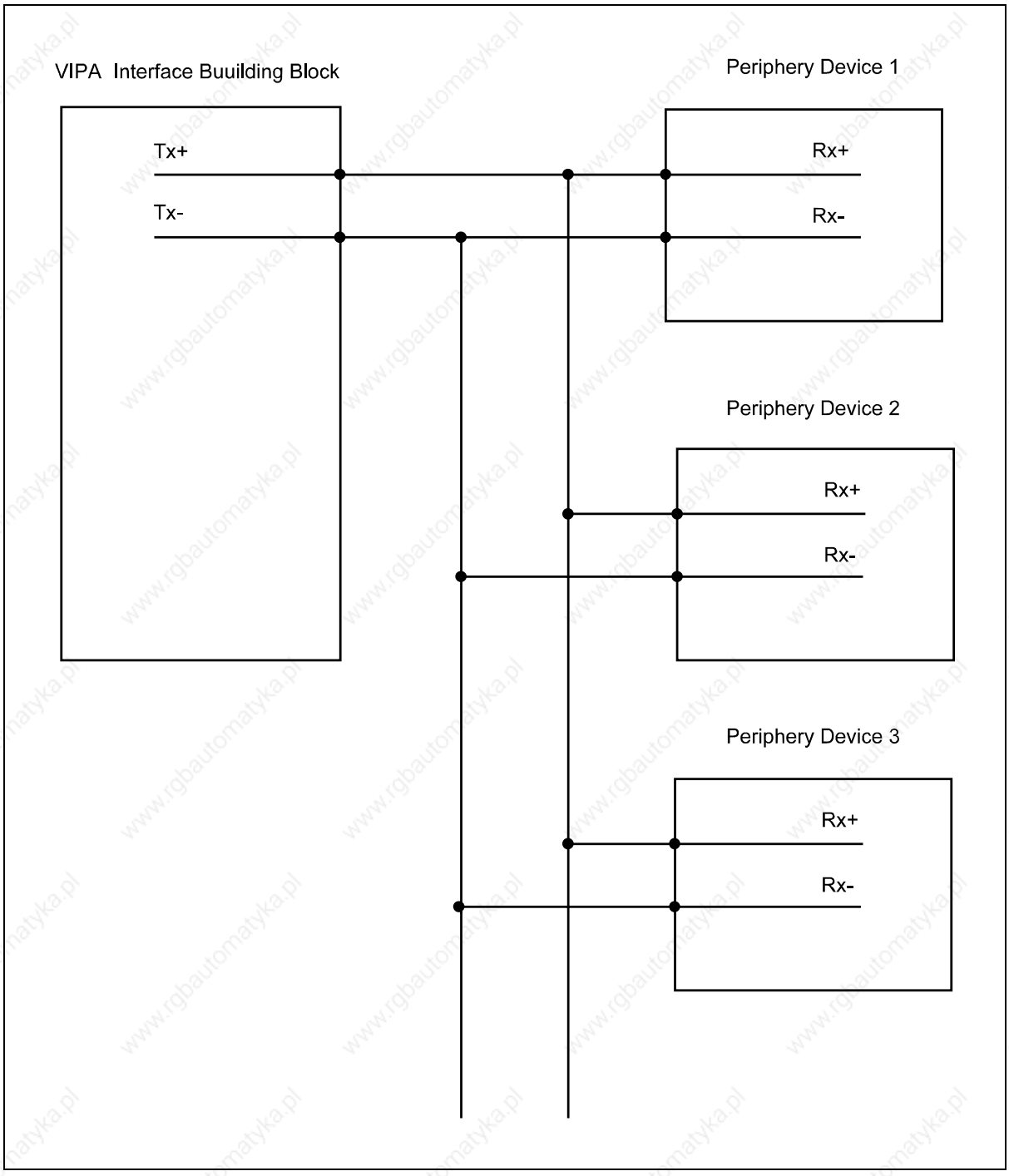


Fig. 3-17: Multidrop connection with a RS422/RS485 module

The number of connection possibilities for external equipment depends on the type of interface.

16 external pieces of equipment can be connected with RS422 or
32 external pieces of equipment with RS485.

3.2.4.6 Four-wire bus connection

The four-wire connection should be selected, if several external pieces of equipment are to be connected to the bus.

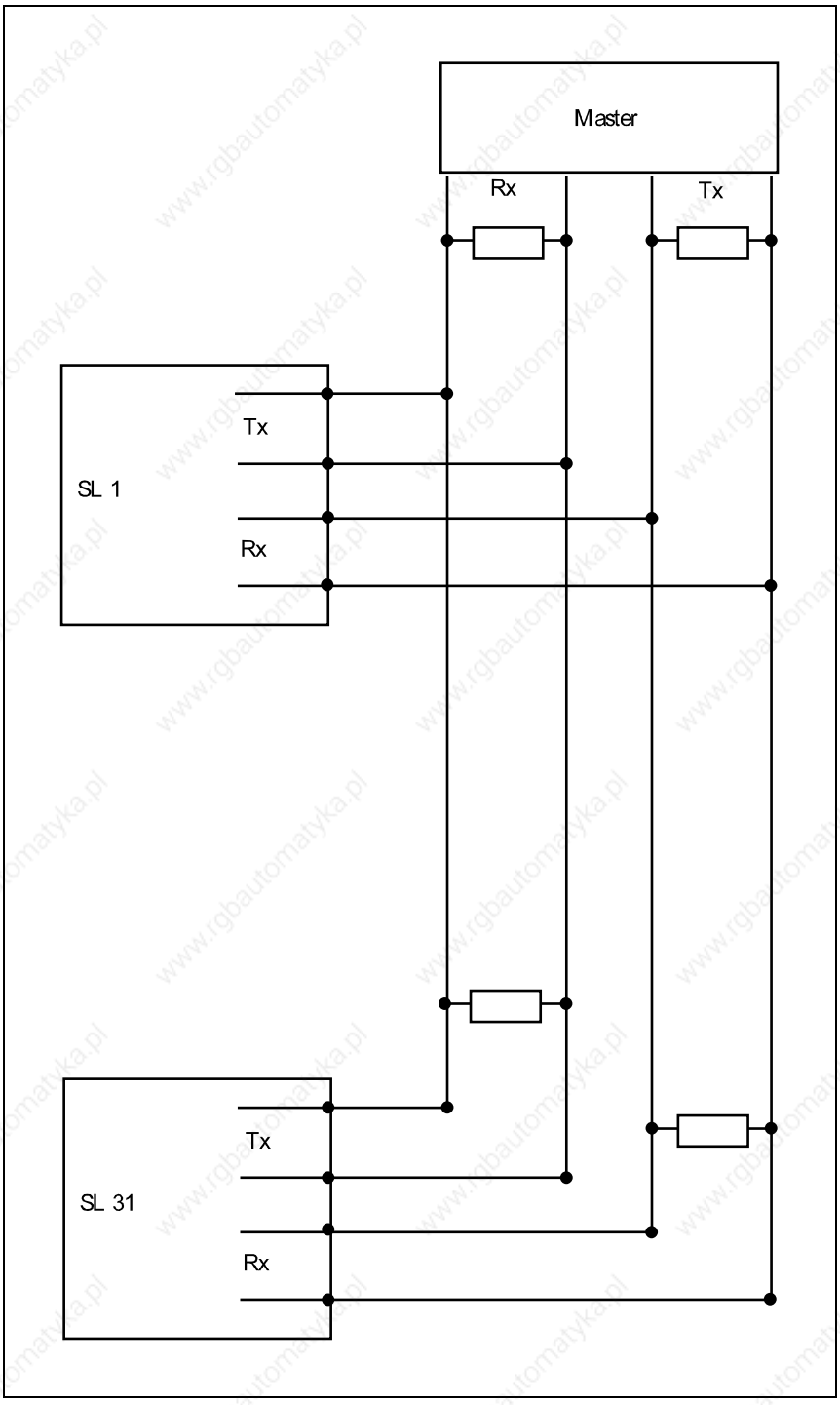


Fig. 3-18: Four-wire connection with a RS422/RS485 module

The number of slave connections, depends on the type of interface:

15 slaves can be connected with RS422 or
31 slaves with RS485

3.2.5 RS422P (MD33) and RS485P (MD34) module floating

3.2.5.1 General

The module serves the point-to-point or the multidrop connection on the basis of the RS422/RS485 norm. The module possesses a potential separation.

When operating RS422P or RS485P modules on the bus ends, a 100Ω terminating resistor is additionally connected by means of the existing switch on the module.

The terminating resistor is connected, when the switch is in the ON position.

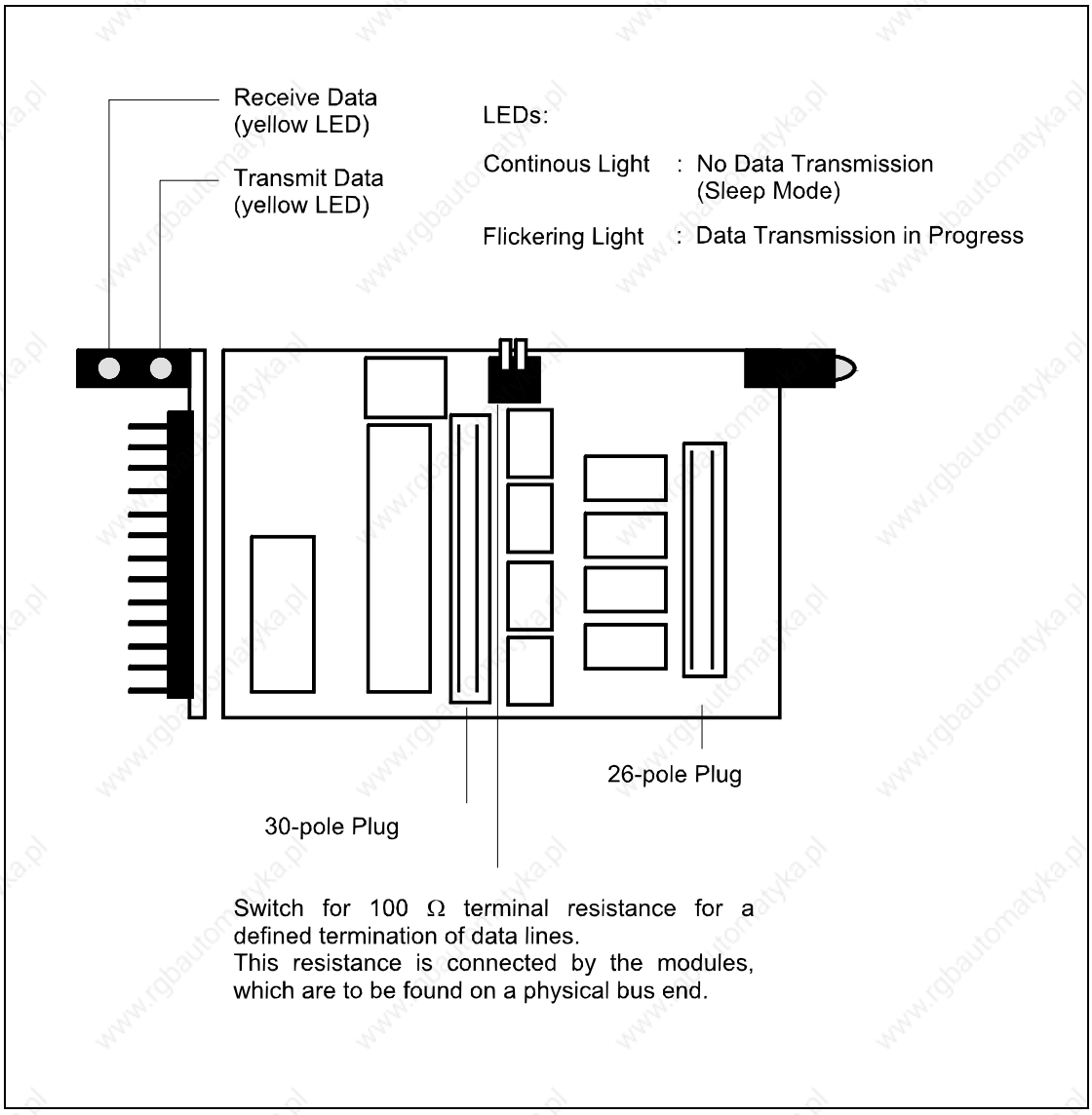


Fig. 3-19: Structure of RS422P and RS485P modules

Please notice

Transmitter and receiver are always active on the module MD33. This module can send a "Break".

If using the MD34 the receiver only receives data when it does not send itself, that means one does not receive the even sent data. MD34 is not Break-able.

3.2.5.2 Allocation of RS422P and RS485P terminal pins

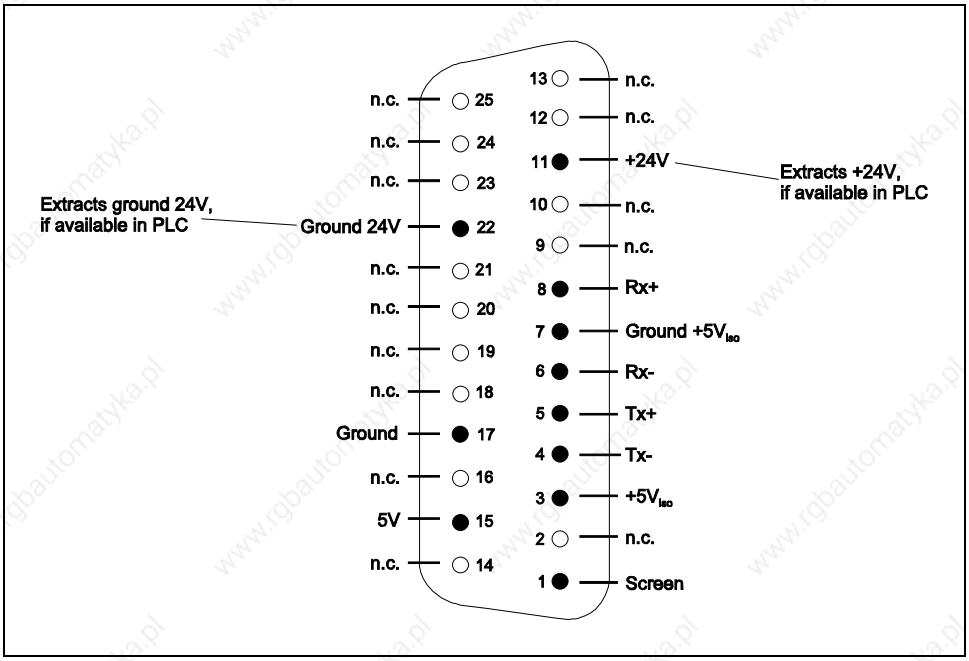


Fig. 3-20: Allocation of terminal pins when using RS422P and RS485P modules

3.2.5.3 Two-wire Bus Connection

By linking Rx+ and Tx+ or Rx- and Tx- it is possible to produce a two-wire bus connection. The RS485 interface uses the two-wire connection.

Due to the links, the sent data is immediately recoupled. The recoupled data is to be filtered by using software. With the RS485P module the receiver is switched off at transmission and thereby disables the recoupling.

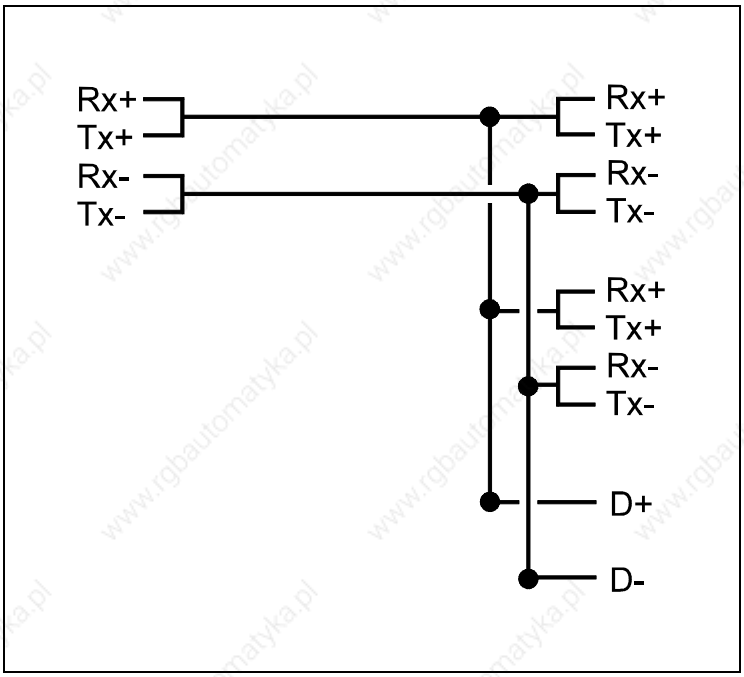


Fig. 3-21: Two-wire bus connection

3.2.6 CENTRONICS module (MD24)

3.2.6.1 General

The module is used for controlling equipment with CENTRONICS interfaces. The module is preferably used for controlling printers.

When using the CENTRONICS module, no presettings need to be observed.

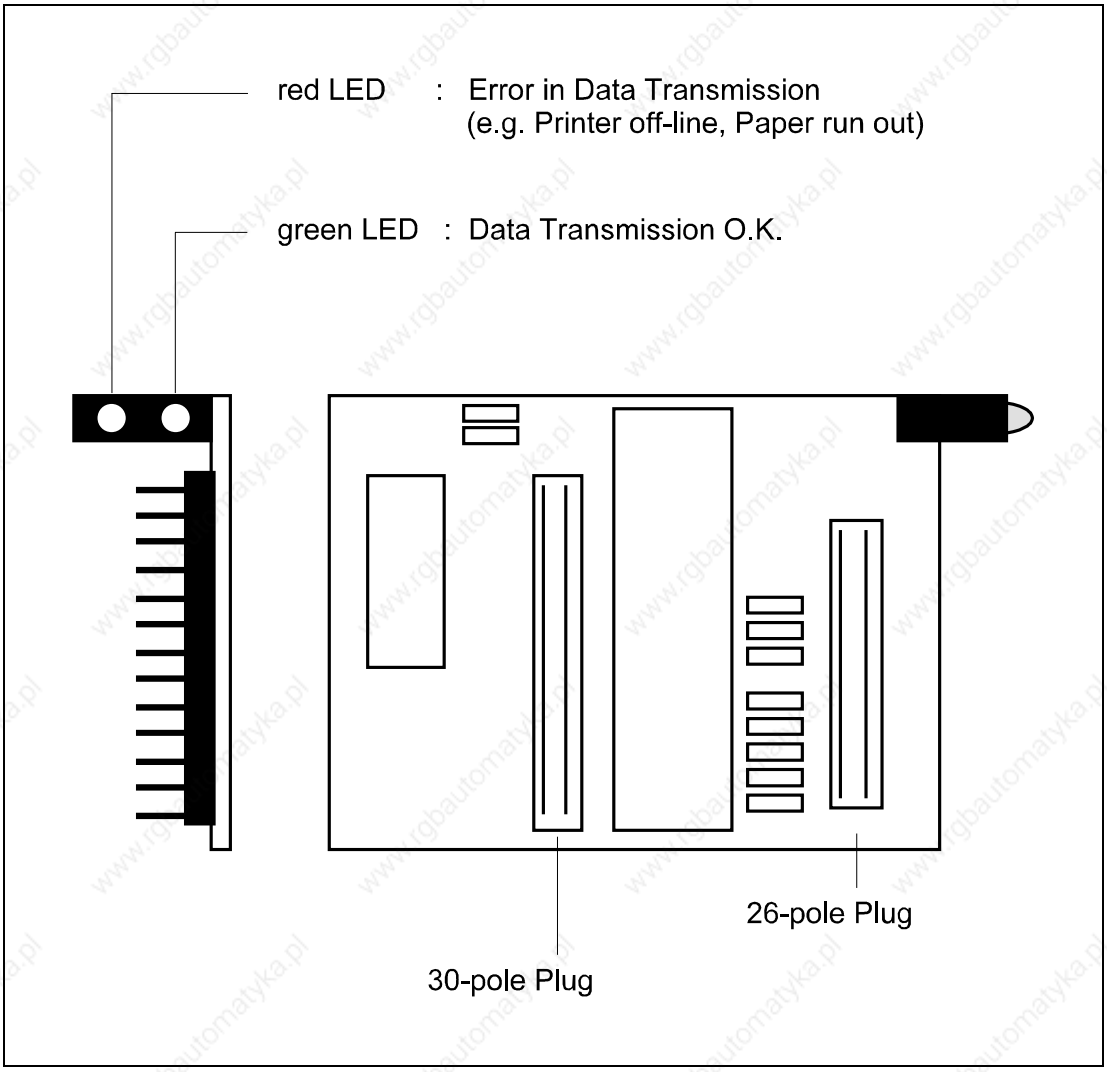


Fig. 3-22: Structure of the CENTRONICS module

3.2.6.2 Allocation of the CENTRONICS module's terminal pins

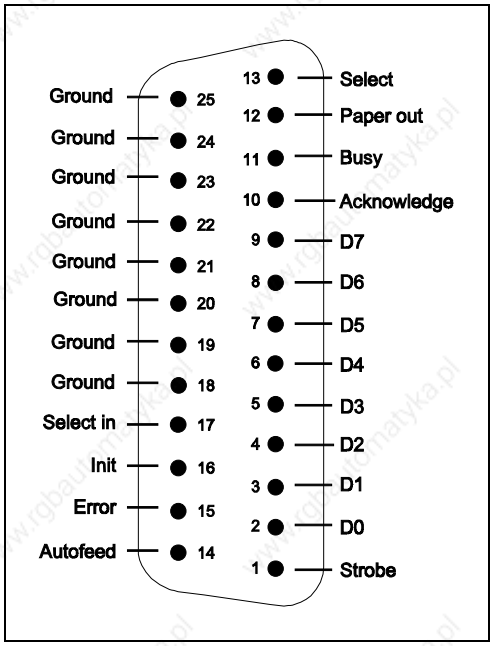


Fig. 3-23: Allocation of terminal pins when using a CENTRONICS module

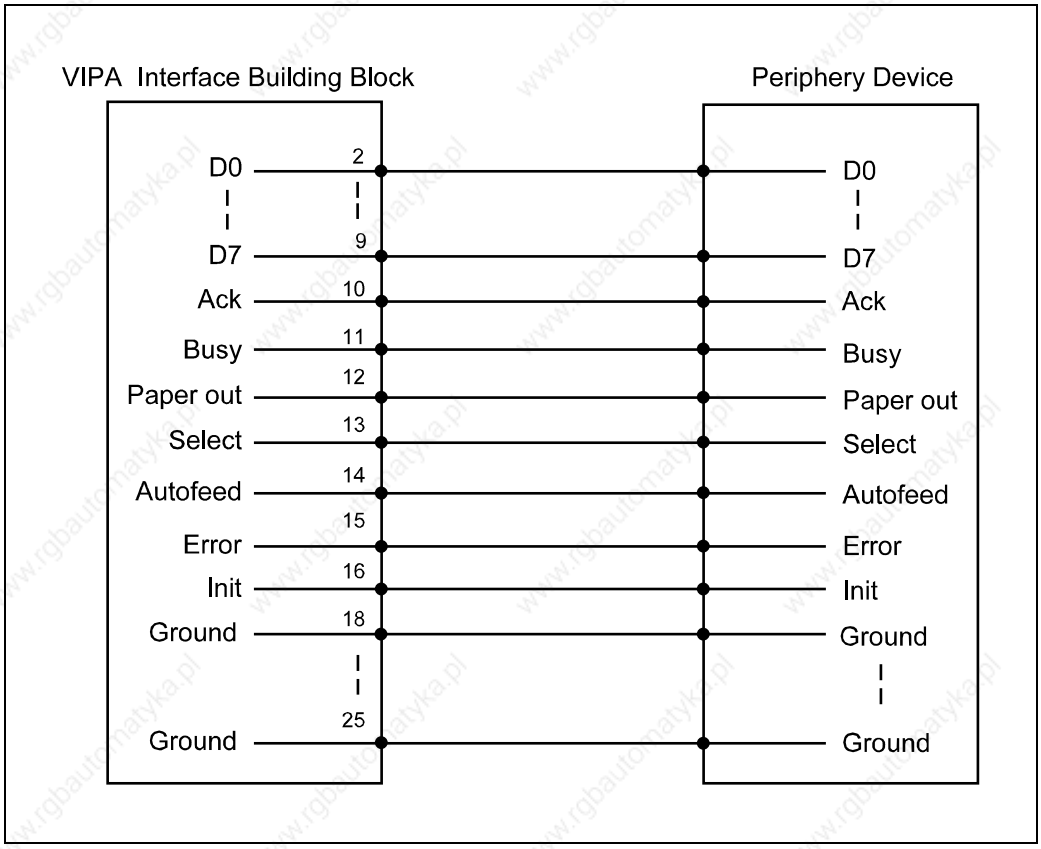


Fig. 3-24: Point-to-point connection when using a CENTRONICS module

The allocation of the 25-pole SubD sockets corresponds to the allocation selected by IBM for the connection of a parallel printer in accordance with CENTRONICS. Thus any printer cable, suitable for a PC (personal computer), can be used.

3.2.7 SSI module (MD39)

3.2.7.1 General

An SSI interface is a synchronous pulsing serial interface. SSI is the abbreviation for **S**ynchronous **S**erial **I**nterface. The clock frequency is preset by the user. The SSI module enables the connection of completely coded transducers with an SSI interface.

The module converts the transducers' serial information into parallel information and makes these available to the interface module and consequently the control.

Connections for 2 test probes are available on the module. The connected test probes can be supplied directly by the module and therefore do not require external voltage supply.

It is possible to transmit the data in Gray or binary code.

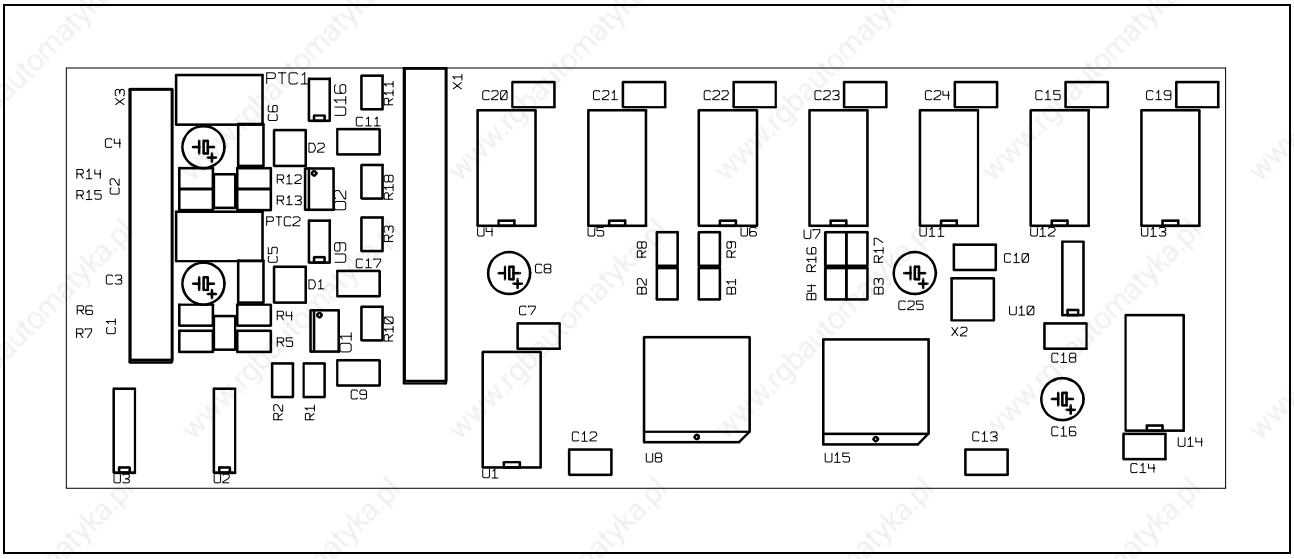


Fig. 3-25: Structure of an SSI module

Characteristics of the patented SSI protocol

- The cabling requirement is not dependant upon the length of the data words.
  Only 4 links are used.
- Maximum interference immunity by using symmetrical timing and data signals.
- Safe data acquisition by using the cyclic Gray code (eligible).
- Electrical decoupling of the receiver and the coder by means of the optical coupling device.

3.2.7.2 Data flow

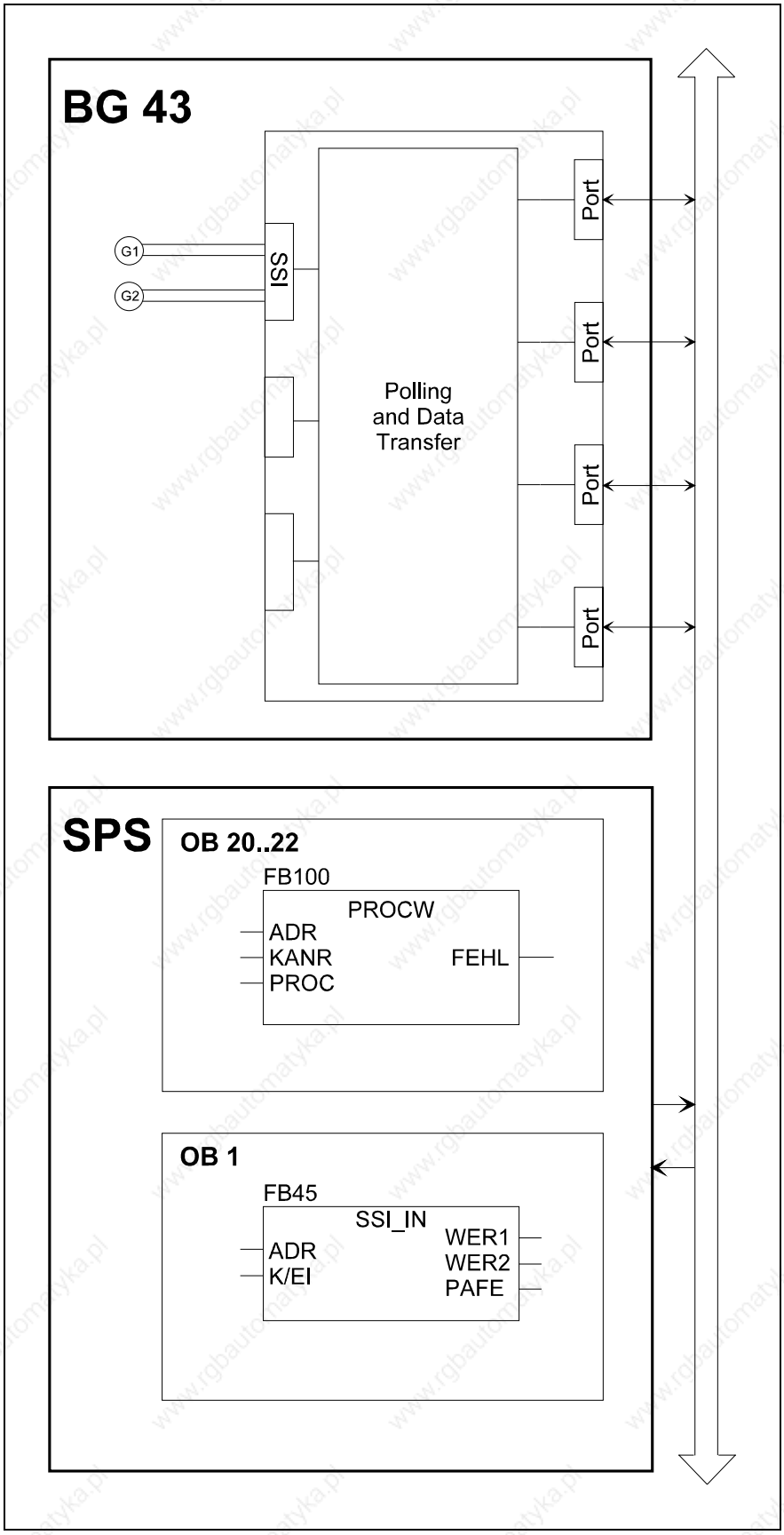


Fig. 3-26: Data flow

3.2.7.3 Allocation of the SSI module's terminal pins

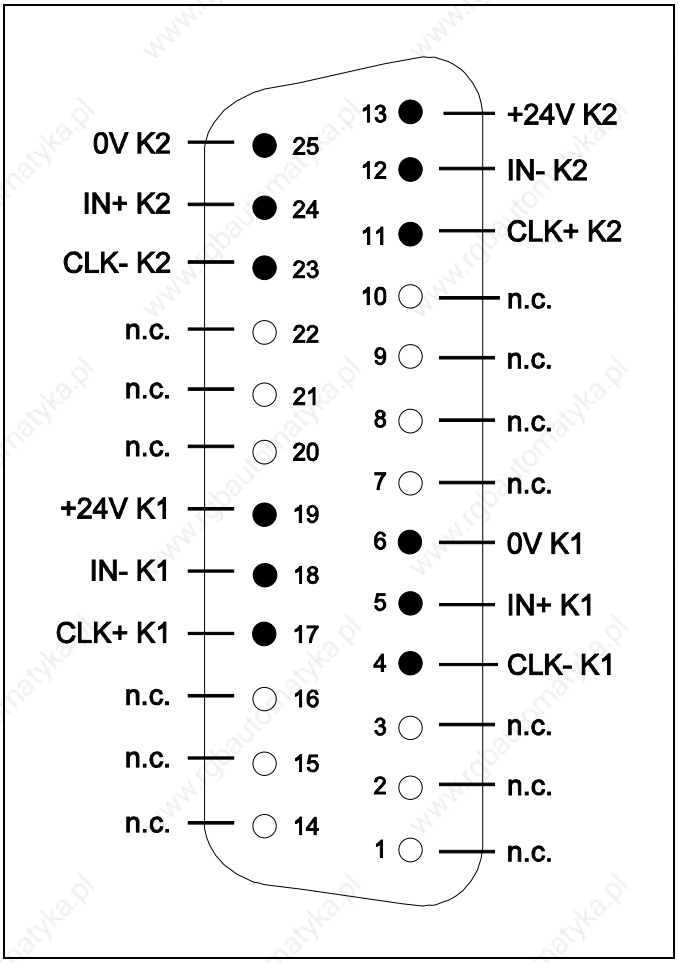


Fig. 3-27: Allocation of the terminal pins when using an SSI module

Most suitable cable length

The clock frequency was increased to 10 MHz with the release 4. The cable length depends on the clock frequency and the transfer rate.

Cable length for **10 MHz** Interface modules:

max. cable length	at 150 kBit	at 300 kBit
	90 m	25 m

Older Interface modules til release 3 have a internal frequency of **6 MHz**.

Cable length for **6 MHz** Interface modules:

max. cable length	at 150 kBit	at 300 kBit
	150 m	60 m

3.2.7.4 Functional mode

The SSI module contains two SSI receivers, which can simultaneously read into the data of two transducers. The data can be read by writing an address into the SSI module's register area. In principle, 25 clock pulses are generated, which then read the transducer's data into the internal shift register. This shift register can be read in parallel.

The Gray-Code is converted into the binary code within the module. This adjustment can be determined per channel by the soldered areas of B2 and B4.

The pulse is given to the encoder by means of the RS485 driver software. In principle, 24 bit data is received. If an encoder with a higher data width is used, as for example 25 bits, the least significant bit (LSB) is therefore cut off.

The test probes can be supplied with 24 V directly from the printed circuit board. The 24V current supply for two encoders is safeguarded with 1A.

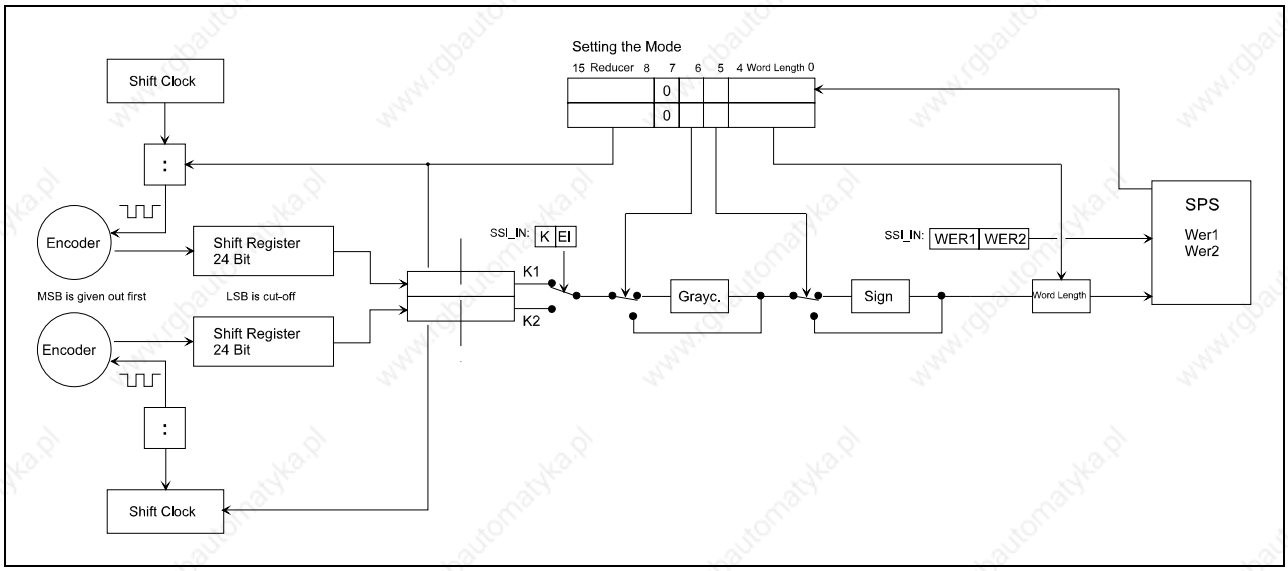


Fig. 3-28: Block diagram of an SSI

3.2.7.5 Soldered areas

Soldered areas are to be found on the module and are responsible for presetting the data coding for channels 1 and 2. Please observe the following table for the significance of the individual soldered areas.

Channel	Funktion	Bridge B1	Bridge B2	Bridge B3	Bridge B4
1	150 kBit			open	
	300 kBit			closed	
	Binary coding				open
	Gray coding				closed
2	150 kBit	open			
	300 kBit	closed			
	Binary coding		open		
	Gray coding		closed		

Tab. 3-3: SSI module's soldered areas

3.2.7.6 Application of SSI modules with VIPA data handling blocks

Possible data handling blocks

Initialization	FB100, FB101
Cycle	FB45

For the application of an SSI module, it is necessary to enter the SSI application into the FB100 or the FB101.

The function blocks FB100 or FB101 are described in chapter 4.1.4 as well as parametrizing examples.

The code to be given for the identifier PROC is 12 (0Ch) for the SSI transmission.

3.2.7.7 FB45 (SSI_IN), reading SSI data

The data handling block FB45 (SSI_IN) offers the possibility to output data from a SSI module. This function block can be used in all CPUs. The SSM is to be addressed in the peripheral area PY128-PY248.

The following parameters are to be transferred when loading FB45:

Des.	Format	Explanation
ADR	KF	Building block's periphery address
K/EI	KY	Number of channels / Number of encoders
WER1	W	Output value low word
WER2	W	Output value high word
PAFE	BY	Error byte

Tab. 3-4: List of parameters for the loading of FB45

The following details are necessary for the abovementioned paramenters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

K/EI Number of channels onto which the module is plugged. The channel number to be used depends on the interface:

Building block BG41	Channel number 1
Building block BG42	Channel number 1, 2
Building block BG43	Channel number 1, 2, 3

Number of input channels, whose contents are to be read.

Range of values 1, 2.

WER1 The low word of the read values is to be filed in this flag word.

WER2 The high word of the read values is to be filed in this flag word.

PAFE Error byte. This byte has a value of 0 when functioning correctly. If no data is received, this byte is set at 1.

3.2.7.8 Example

The SSI module should be used with channel 1 of the building block. The building block is to be found at the address F080h. This corresponds to the peripheral address PY128. A device which provides measured values is connected to the input channel of the SSI module. The building block's error messages should be filed in the flag byte MB 199 and the module's error messages should be filed in the flag byte MB35.

Start OBs

```
OB21

         SPA FB 101
NAME     #PROCW
ADR      =KF +128      Peripheral address of the block PY128, in accordance
                       with F080h.
KANR     =KF +1        Channel 1 on the interface building block
PROC     =KF +12       SSI procedure, code 12
FEHL     =MB 199       Error code in flag byte 199
```

```
OB22

         SPA FB 101
NAME     #PROCW
ADR      =KF +128      Peripheral address of the block PY128, in accordance
                       with F080h.
KANR     =KF +1        Channel 1 on the interface building block
PROC     =KF +12       SSI procedure, code 12
FEHL     =MB 199       Error code in flag byte 199
```

Cycle OB

```
OB1

         SPA FB 45
         NAME    #SSI_IN
ADR      =KF +128      Peripheral address of the block PY128, in accordance
                       with F080h.
K/EI     =KY 1,1       Channel 1 on the building block, input channel on
                       module
WER1     =MW 12        Low word of the read values in MW12
WER2     =MW 10        High word of the read values in MW10
PAFE     =MB 35        Error byte filed in flag byte 35
```

3.2.7.9 Application of the SSI-module without data handling blocks

3.2.7.10 Parameter channel

Further information for the application of the interface without Data Handling Blocks can be found in chapter 4.2.

Select the SSI function for the corresponding interface channel via the parameter channel with the command 5.

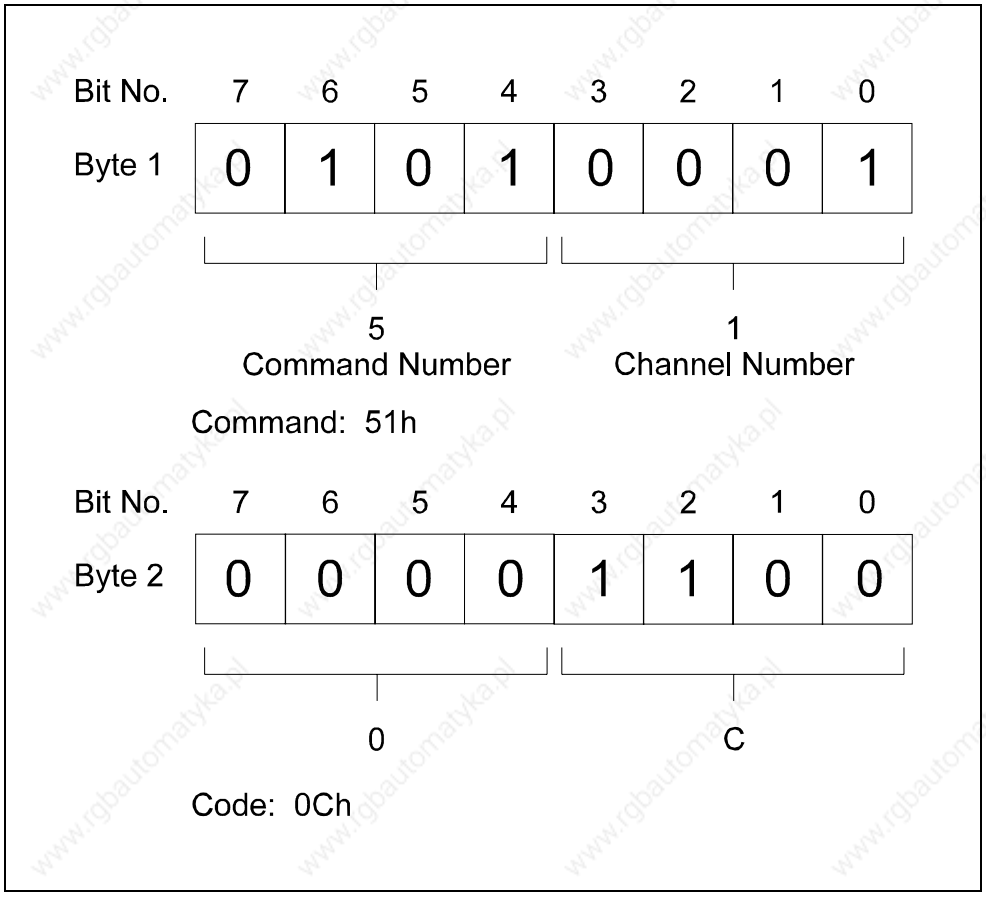


Fig. 3-29: Function selection SSI with command 5

3.2.7.11 Commands for the data channel

Use the data channel for handing over the commands.

Command overview

1	Reading of an SSI value	1 byte command
3	Setting of the mode	2 byte command

Command 1 - Reading of an SSI encoder value

An SSI encoder value can be read via the data channel with command 1. One has the possibility of selecting 2 SSI encoders.

0 Encoder 1

1 Encoder 2

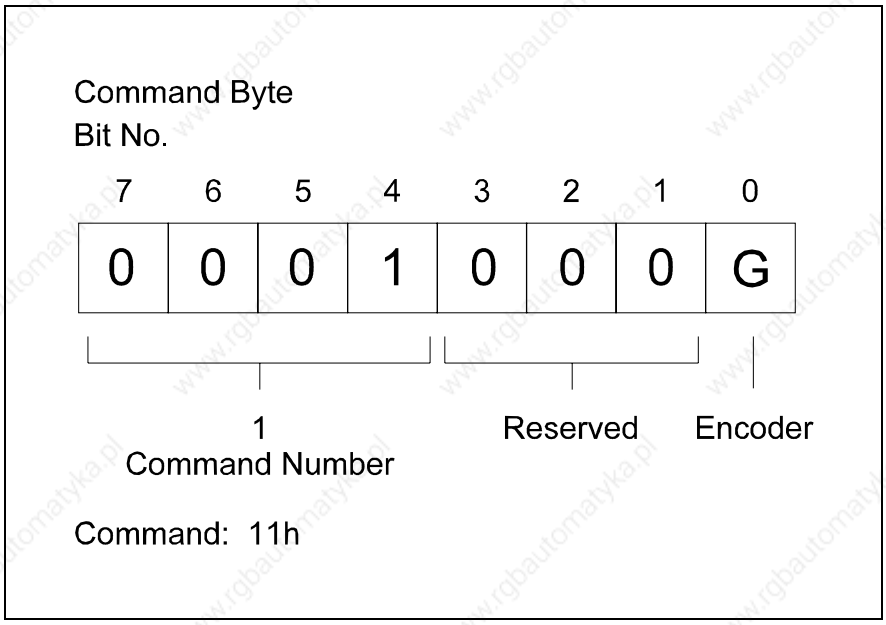


Fig. 3-30: Reading of an SSI encoder with command 1

The following bytes are returned to the data channel as follows:

Bit. No.	7	6	5	4	3	2	1	0
LSB	D7	D6	D5	D4	D3	D2	D1	D0

Bit. No.	7	6	5	4	3	2	1	0
	D15	D14	D13	D12	D11	D10	D9	D8

Bit. No.	7	6	5	4	3	2	1	0
	D23	D22	D21	D20	D19	D18	D17	D16

Bit. No.	7	6	5	4	3	2	1	0
MSB	D31	D30	D29	D28	D27	D26	D25	D24

Command 3 - Setting the mode

When setting the mode, two additional bytes are transferred as argument.

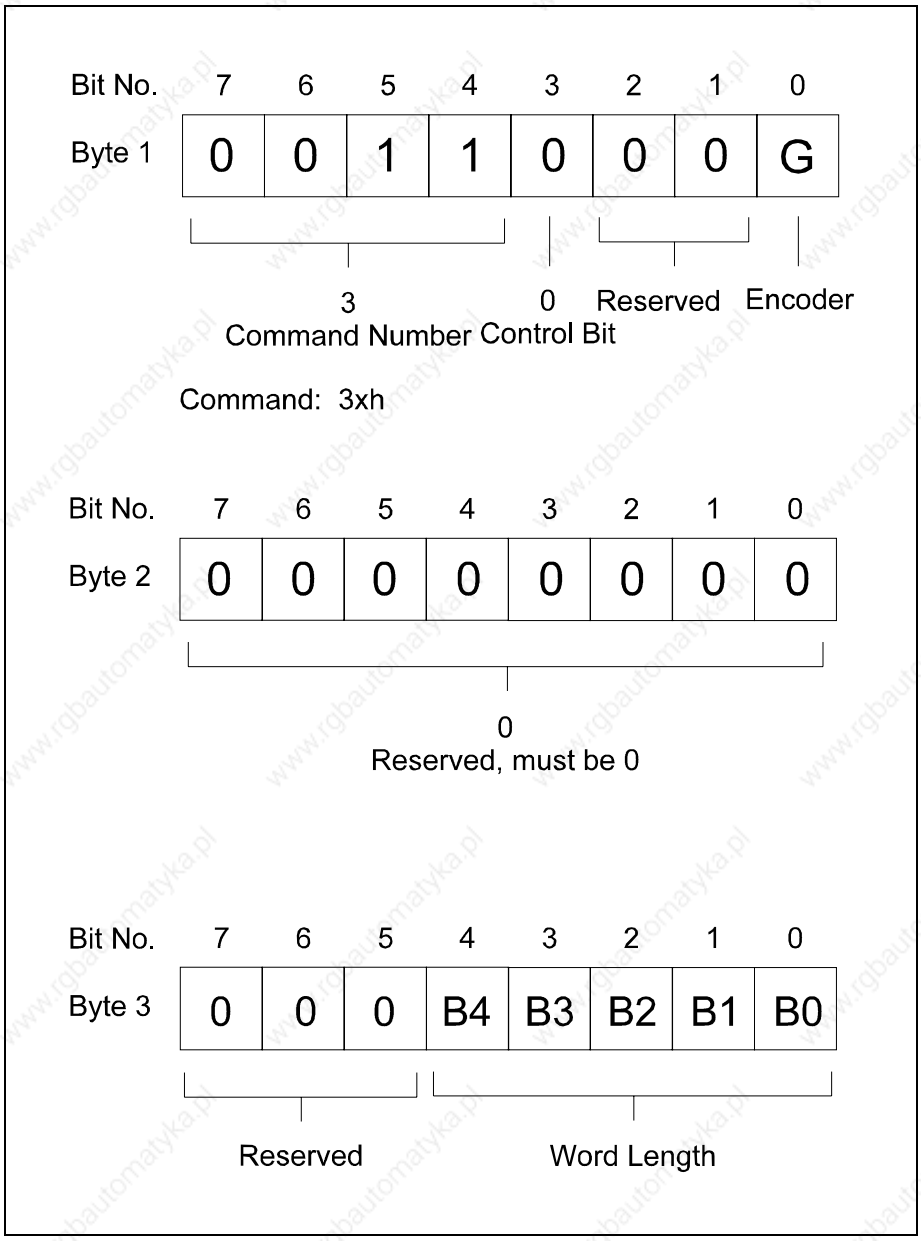


Fig. 3-31: Setting of the mode SSI with command 3

Bit 3 in byte 1 is used as the control bit for the block commands. If this bit is set at 1, the command applies for all connected sonsors of the channel.

The mode bits are occupied as follows:

Bit 0 to 4 (W0...W4)	The bit length of the encoder telegram is adjusted here. Standard value: 24 0 not permissible 1...24 area, which can be given. 1 corresponds to the bit length 1, 2 corresponds to the bit length 2 etc.
Bit 5 to 7	reserved and must be set at 0.

3.2.8 Counter module (MD18,19)

3.2.8.1 General

The counter module is available in the configurations 24V (MD18) and 5V (MD19).

Three independant counters are to be found on the module, 2 of which can be cascaded.

The counter modules in their delivery state are so adjusted, that 3 independant 16-bit counters are available. (See the positioning of the plug connectors X1, X2 and X3, as drawn below)

If counter 1 and counter 2 are to be cascaded, the jumpers X1 and X2 are to be replugged into position "32".

With the plug connectors from X4 to X6, one can alternatively place Carry" or "Borrow" with each counter on the yellow LEDs and the respective switch outputs.

- Carry Change from the negative to the positive counter area.
- Borrow Change from the positive to the negative counter area.

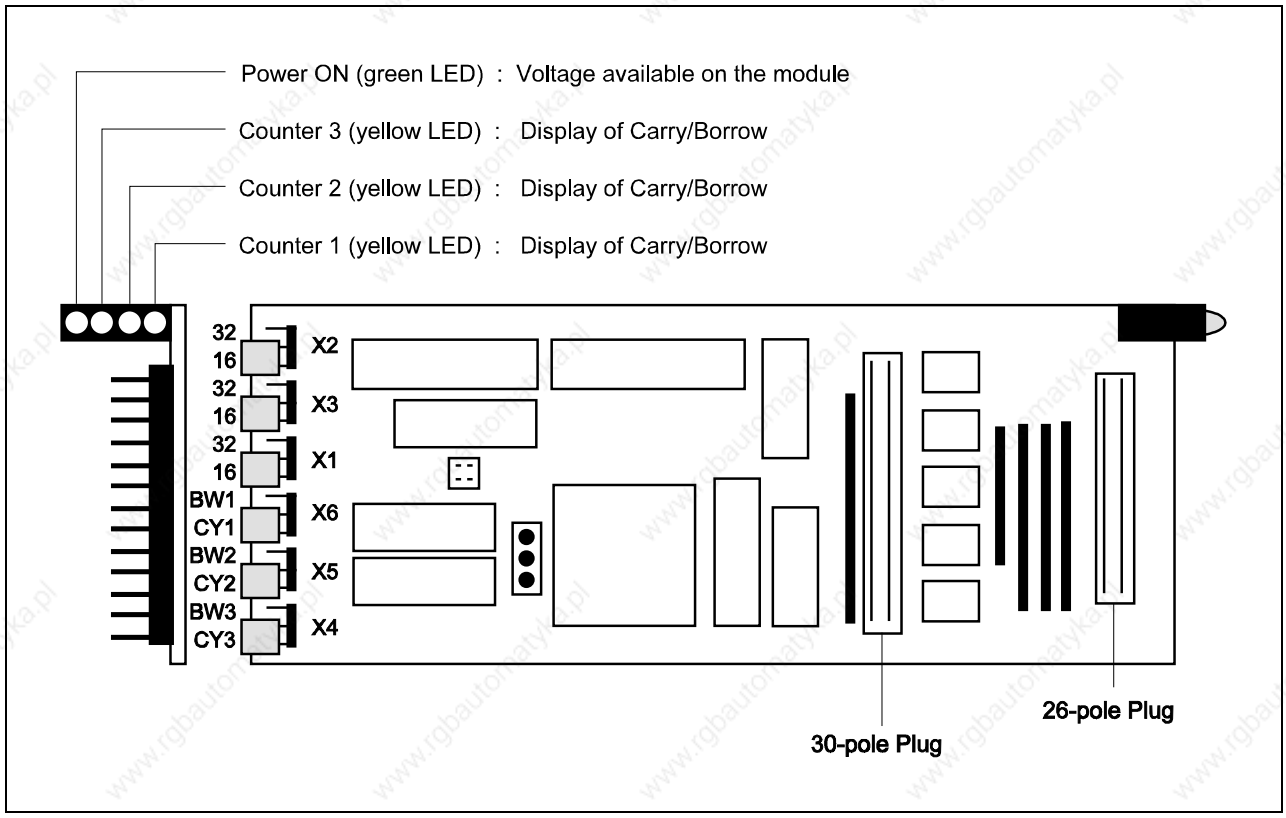


Fig. 3-32: Structure of the 5V counter module

Eight different operating modes are available for connecting incremental or path encoders:

- Normal up and down counter
- Measurement of the pulse-width
- Measurement of the frequency
- 5 different operating modes for out-of-phase impulses

3.2.8.2 Cascading of counter 1 and 2 to a 32 bit counter

Counter 1 and 2 can be interconnected by the three jumpers X1, X2 and X3. The following settings are possible:

Jumper X1, X2 and X3 in position 16
Standard position, both counters are operated independantly from each other.

Jumper X1, X2 and X3 in position 32
Counter 1 and 2 are cascaded by means of hardware. The inputs of counter 2 are not to be connected and counter 2 is to be operated in mode 0. The 32 bit counter value can be obtained from the counter value of counter 1 (low word) and the counter value of counter 2 (high word).

3.2.8.3 Carry and borrow handling

The counters provide borrow or carry information for each counter channel. Any carry or borrow flag which appears is saved in the status register. The flags can be deleted by describing the counter with a new value. A carry which is set is deleted by a borrow and vice-versa. The carry or borrow flag can be alternatively placed on the yellow light emitting diodes on the front of the building block and on the 25-pole SubD socket's switch outputs. The choice is determined by means of the jumper X4, X5 and X6:

- Channel 1 over Jumper X6 CY1 (Carry Channel 1) or BW1 (Borrow Channel 1)
- Channel 2 over Jumper X5 CY2 (Carry Channel 2) or BW2 (Borrow Channel 2)
- Channel 3 over Jumper X4 CY3 (Carry Channel 3) or BW3 (Borrow Channel 3)

With the modules it is possible to connect 5V or 24V to the outputs. By active borrow or carry the respective voltage is connected to the output. The output is non-floating and is **not short-circuit proof**. It can drive a maximum of 200mA. Ensure to have a joint mass point.

The 5V module receives its voltage from the internal 5V. In the case of the 24V module, the 24V voltage supply must be separately fed. This can occur over the rear cover bus or over the 25-pole SubD socket.

One is to note, that in the case of external supply the rear cover bus is disconnected by the plug connector J6. For further details see chapter 5.2.9.

3.2.8.4 Allocation of 5V counter's terminal pins

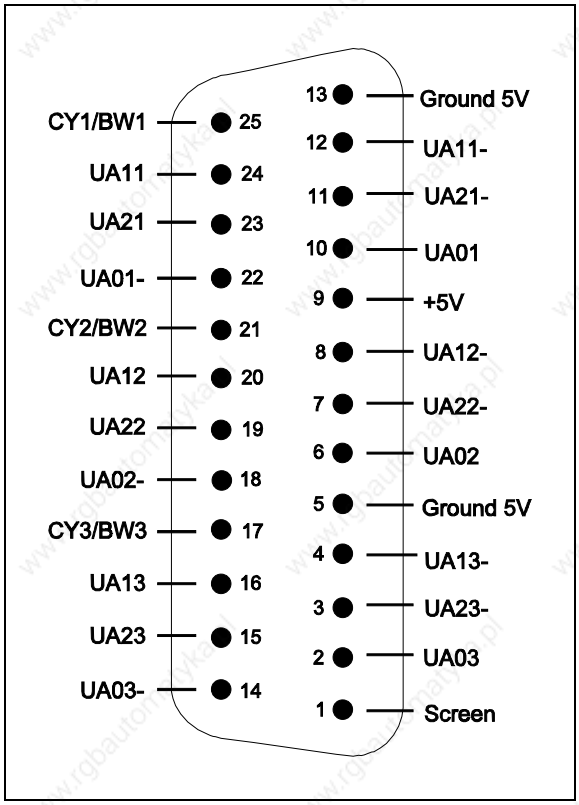


Fig. 3-33: Allocation of terminal pins when using a 5V counter module

Description of Signals:

Earth	:	Supply voltage output
+5V	:	Supply voltage output
CYa/BWa	:	Carry or borrow output of counters 1, 2 or 3 active: high - max. 5V, 100mA inactive: high-impedance
UAxa-	:	Counter input 1 or 2 for counter 1, 2 or 3 active: low - TTL or RS 422 level
UAxa	:	Counter input 1 or 2 for counter 1, 2 or 3 active: high - only connect by differential drives (RS 422), otherwise leave open
UA0a	:	Reset counters 1, 2 or 3 active: high - TTL or RS 422 level
UA0a-	:	Reset counters 1, 2 or 3 active: low - only connect by differential drives (RS 422), otherwise leave open

If UA01(pin 10), UA02 (pin 6) and UA03 (pin 2) are not used, they must be earthed!

3.2.8.5 Allocation of 24V counter's terminal pins

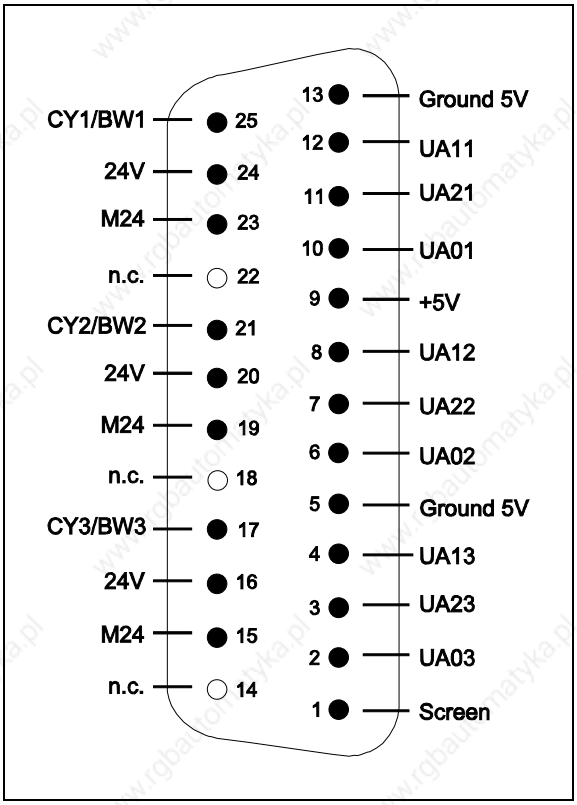


Fig. 3-34: Allocation of terminal pins when using a 24V counter module

Description of Signals:

Earth	:	Supply voltage output
+5V	:	Supply voltage output
24V	:	Supply voltage output (for external voltage supply or extraction
M24	:	Supply voltage output by adjustment to internal voltage supply))
CYa/BWa	:	Carry or borrow output of counter 1, 2 or 3 active: high - max. 24V, 50mA inactive: high-impedance
UAxa	:	Counter input 1 or 2 for counter 1, 2 or 3 active: high 24V - level
UA0a	:	Reset counter 1, 2 or 3 active: high 24V - level

If UA01(pin 10), UA02 (pin 6) and UA03 (pin 2) are not used, they can be left open.

3.2.8.6 Application of counter modules with VIPA data handling blocks

Possible data handling blocks

Initialization	FB100, FB101, FB12
Cycle	FB10, FB11

3.2.8.7 General

The interfaces of the building blocks BG41, BG42 and BG43 can be used as counters. For this the procedure "ZÄHLER" must be selected with the FB 100 or the FB101. Note, that a counter module is connected to the corresponding interface.

With the counter module impulses can be counted or frequencies or pulse widths can be measured. Connect the input pulse at the 25-pole sub D socket. Carry or Borrow are available as outputs for each counter.

The module contains as a counter block the triple counter TMCT 12316 from Texas-Instruments.

For each of the three counters, 8 different modes can be set independently from each other on a 16-bit register.

Counter 1 and Counter 2 can be cascaded to a 32-bit counter.

3.2.8.8 Programming the counter modules

If the counter module is to be used for one or more interfaces of the building blocks BG41, BG42 or BG43 the counter procedure is to be set with FB 100 or FB 101.

By selecting the procedure "Counter", the following parameters are passed to the building block:

- Building block’s addresses
- Channel number on which the counter module is to be operated (depending on building block 1, 2 or 3).
- Code for the counter procedure (Code: 05)

Together with the FB100 or the FB101, FB12 is to be loaded into the start OB. The counter mode is selected with the FB12.

If the procedure driver has been activated for the counter module, further parameters have to be transferred to the module.

On transferring the parameters it is determined how the counter module is to be operated and which information you wish to receive or transfer to it:

You can

- read the counter status
- read the counter reading
- give the counter a default value
- allow the counter to operate in a certain mode

The parameters can be transferred by means of VIPA data handling blocks (HTB) or directly transferred by the use of commands.

3.2.8.9 FB10 (LESEN), reading of the counter reading

With FB10 (LESEN) the value of a particular counter can be read.

The following parameters are to be transferred when loading FB10:

Des.	Format	Explanation
ADR	KH	Building block's abssolut base address
KNR	KF	Number of channels
ZNR	KF	Counter number
WERT	W	read value (16 Bit)
NEUS	M	New start bit

Tab. 3-5: List of parameters for the loading of FB10

Due to time reasons this function block does not examine the reliability of the formal operands KNR and the validity of the counter number. If the read numerical value is filed in a data word, the corresponding data block must be opened before loading FB 10 with A DBxx.

The following details are necessary for the abovementioned parameters:

ADR Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

KNR Number of channels, to which the counter module is connected. For the building blocks the following inputs are possible:

BG41 - 1

BG42 - 1 or 2

BG43 - 1, 2 or 3

ZNR Number of counters on the module, which are to be read. Each counter module has three counters.

The inputs1, 2 or 3 are therefore permitted.

WERT Giving the number of the data or flag words, in which the value read from the counter is to be entered.

NEUS As long as the bit is at the signal stand is at 1, the processing of the function block is discontinued.

3.2.8.10 FB11 (SCHREIBEN), give counter a default value

With FB11 (SCHREIBEN) a certain counter can be given a default value.

The following parameters are to be transferred when loading FB11:

Des.	Format	Explanation
ADR	KH	building block's absolute base address
KNR	KF	Number of channels
ZNR	KF	Counter number
WERT	W	value to be entered (16 Bit)
NEUS	M	New start bit

Tab. 3-6: List of parameters for the loading of FB11

Due to time reasons this function block does not examine the reliability of the formal operands KNR and the validity of the counter number.

If the write numerical value is taken out of a data word, the correspponding data block must be opened before loading FB 11 with A DBxx.

The following details are necessary for the abovementioned parameters:

ADR Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

KNR Number of channels, to which the counter module is connected. For the building blocks the following inputs are possible:

BG41 - 1

BG42 - 1 or 2

BG43 - 1, 2 or 3

ZNR Number of counters on the module, which are to be given a default value. Each counter module has three counters.

The inputs 1, 2 or 3 are therefore permitted.

WERT Gives the number of data or flag words, in which the value to be transferred to the counter is filed.

NEUS As long as the bit is at the signal stand is at 1, the processing of the function block is discontinued.

3.2.8.11 FB12 (MODE), mode selection for the counter

Use the FB12 (MODE) for selecting the mode of the three counters of the counter module. Each counter, independent from the other counters, can be set to one of eight different modes. Load the FB12 in start OB.
The following parameters are to be transferred when loading FB12:

Des.	Format	Explanation
ADR	KH	building block's absolute base address
KNR	KF	Number of channels
MOD1	KF	Mode for counter 1 (0 ... 7)
MOD2	KF	Mode for counter 2 (0 ... 7)
MOD3	KF	Mode for counter 3 (0 ... 7)

Tab. 3-7: List of parameters for the loading of FB12

The following details are necessary for the abovementioned parameters:

ADR Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

KNR Number of channels, to which the counter module is connected. For the building blocks the following inputs are possible:

BG41 - 1

BG42 - 1 or 2

BG43 - 1, 2 or 3

MOD1
MOD2
MOD3 Mode to be set for the counters 1, 2 und 3. Eight modes can be set independantly from each other for each counter on the module. These modes are described in detail on the following pages.

Counter modes

Eight modes are available for the counters.

Mode 0

Depending on which input the pulse is situated, it is counted upwards or downwards. The functions of the 5V counter module differ in this respect to those of the 24V counter module:

5V Counter module	24V Counter module
UA1x pulse → counts downwards UA2x +5V	UA1x ground → counts upwards UA2x pulse
UA1x pulse → counts upwards UA2x ground	UA1x pulse → counts downwards UA2x ground

Mode 1-5 **Direction discriminator**

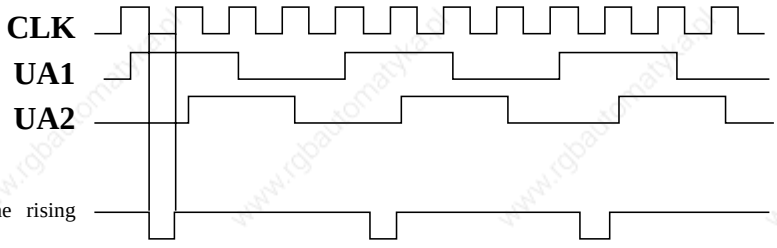
With this mode, out of phase input pulses e.g. from a rotary pulse generator, can be evaluated.

Depending on the phase position of UA1 to UA2, counting is done upwards or downwards.

UA1 before UA2 → counts upwards

UA1 after UA2 → counts downwards

In mode 1 and 2 a rising edge from UA1 or UA2 results in a count impulse.



Mode 1 Evaluation of the rising edge UA2 only

Mode 2 Ealuation of the rising edge UA2 only

Mode 3 Evaluation of the rising and falling edge of UA1

Mode 4 Evaluation of the rising and falling edge of UA2

Mode 5 Evaluation of the rising and falling edges of both UA1 and UA2

Mode 6 **Measurement of pulse width**

In this mode the impulse, whose width is to be measured, is to be positioned at input UA1. A counter frequency of 10 Mhz is used.

The count direction can be determined by the UA2.

UA2 = 1 → counts downwards

UA2 = 0 → counts upwards

The counting process is started with the falling edge of the UA1 and is terminated with the rising edge. The corresponding count is recorded in a register and can be read from there.

Mode 7 **Measurement of frequency**

With this mode an unknown frequency can be measured. The unknown frequency is to be positioned at input UA1 and a known frequency e.g. created by the clock generator, at UA2.

If UA2 goes from 1 to 0, the counting process starts with the frequency of UA1. If UA2 goes from 0 to 1, the counter process is stopped, the count is loaded into the output register and the counter is reset.

Example

The following example shows how a counter module is to be parametrized. You will find here among others the parametrizing of the functional blocks for reading and the ? of counters.

Startup-OBs

```
#OB20

00002      :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                          Base address
      KANR =KF +1                            Channel number
      PROC =KF +5                            Procedure counter
      FEHL =MB 199                           Error byte
0000E      :
00010      :SPA FB 12
      NAME #MODE
      ADR  =KH F080                          Base address
      KNR  =KF +1                            Channel number
      MOD1 =KF +1                            Counter 1 in mode 1
      MOD2 =KF +3                            Counter 2 in mode 3
      MOD3 =KF +0                            Counter 3 in mode 0
0001E      :
00020      :BE
```

```
#OB21

00002      :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                          Base address
      KANR =KF +1                            Channel number
      PROC =KF +5                            Procedure counter
      FEHL =MB 199                           Erorr byte
00010      :SPA FB 12
      NAME #MODE
      ADR  =KH F080                          Base address
      KNR  =KF +1                            Channel number
      MOD1 =KF +1                            Counter 1 in mode 1
      MOD2 =KF +3                            Counter 2 in mode 3
      MOD3 =KF +0                            Counter 3 in mode 0
00020      :BE
```

```
#OB22

00002      :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                          Base address
      KANR =KF +1                            Channel number
      PROC =KF +5                            Procedure counter
      FEHL =MB 199                           Error byte
00010      :SPA FB 12
      NAME #MODE
      ADR  =KH F080                          Base address
      KNR  =KF +1                            Channel number
      MOD1 =KF +1                            Counter 1 in mode 1
      MOD2 =KF +3                            Counter 2 in mode 3
      MOD3 =KF +0                            Counter 3 in mode 0
00020      :BE
```

Cycle OB

```
#OB1

00002      :SPA FB 10                         Read counter value
      NAME #LESEN
      ADR  =KH F080                          Base address
      KNR  =KF +1                            Channel number
      ZNR  =KF +1                            Counter number 1
      WERT =MW 20                            File count value in MW 20
      NEUS =M 0.0                            Power on flag
00012      :A   DB 10
00014      :SPA FB 10                         Read counter value
      NAME #LESEN
      ADR  =KH F080                          Base address
      KNR  =KF +1                            Channel number
      ZNR  =KF +1                            Counter number 1
      WERT =DW 1                             File count value in DW 1 DB 10
      NEUS =M 0.0                            Power on flag
00024      :A   DB 10
00026      :U   M 1.0                         Flag to preset counter
00028      :R   M 1.0
0002A      :SPB FB 11                         Preset counter
      NAME #SCHREIBE
      ADR  =KH F080                          Base address
      KNR  =KF +1                            Channel number
      ZNR  =KF +2                            Counter number 2
      WERT =DW 2                             Value is in DW 2 DB 10
      NEUS =M 0.0                            Power on flag
0003A      :BE
```

3.2.8.12 Application of counter modules without data handling blocks

Further information for the application of the interface without Data Handling Blocks can be found in chapter 4.2.

Command: 80 - Setting the counter mode

With the command 80 the building block is informed, that the counter's mode should be set. The following two bytes transfer the mode which is to be set.

xx occupies the high-order byte

yy occupies the low-order byte

The two bytes are structured as follows:

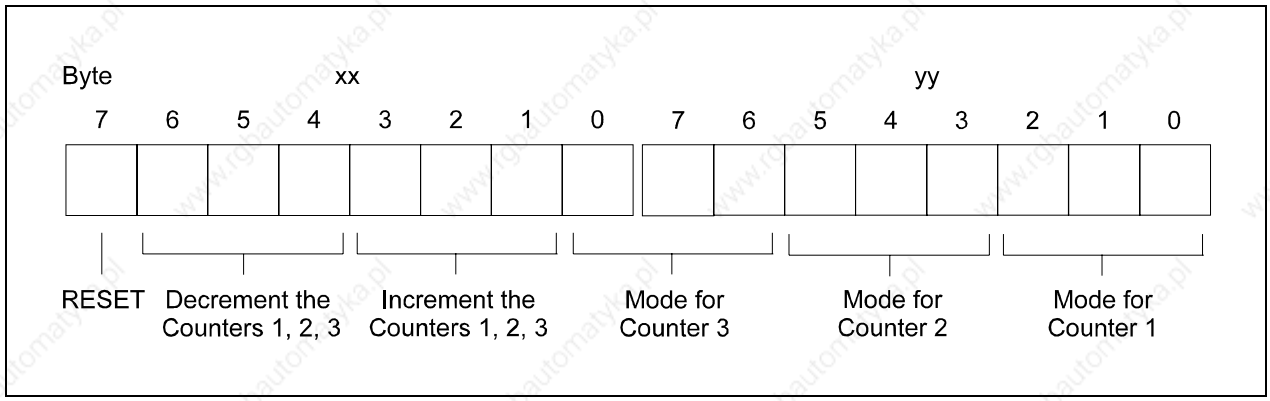


Fig. 3-35: Counter`s sequence bytes from command 80

Byte xx	Bit 7	RESET	0-all counters activated 1-all counter reset
	Bit 6	DOWN3	Bits are normally at 1.
	Bit 5 Bit 4	DOWN2 DOWN1	If a bit is set by means of software from 1 to 0 and back to 1, the corresponding counter is decremented.
	Bit 3	UP3	Bits are normally at 1.
	Bit 2 Bit 1	UP2 UP1	If a bit is set by means of software from 1 to 0 and back to 1, the corresponding counter is incremented.
Byte yy	Bit 0	Mode	
	Bit 7	0-7	
	Bit 6	for counter 3	
	Bit 5	Mode	
	Bit 4	0-7	
	Bit 3	for counter 2	
	Bit 2	Mode	
	Bit 1	0-7	
	Bit 0	for counter 1	

Reading the counter status

Command 00

The counter module's status is read by the interface building block from a register. In this register a counter overflow which may arise is saved upwards (carry) or downwards (borrow).

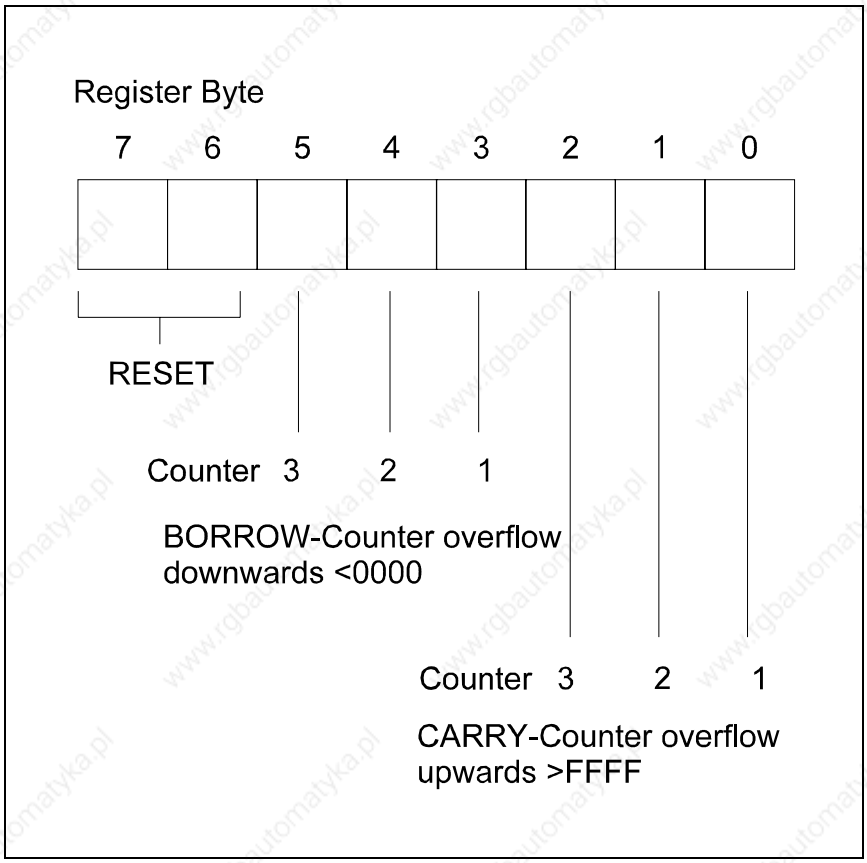


Fig. 3-36: Reading the counter status with command 00

The status byte is returned over the respective channel and can therefore be processed by the PLC.

Reading out of the counter state

Command 01/02/03

With the command 01 the counter state of the counter 1 is returned via the channel, to which the counter module is connected. The high-order byte of the counter value comes first and then the lower-order byte. Analogues apply for counter 2 and 3.

Describing a counter with a value

Command: 81 xx yy / 82 xx yy / 83 xx yy

In order to pre-occupy one of the 3 counters with a value, the command 81 (or 82/83), followed by the value chosen, is given via the corresponding channel. In this case xx is the high-order byte, yy is the low-order byte of the chosen 16-bit value.

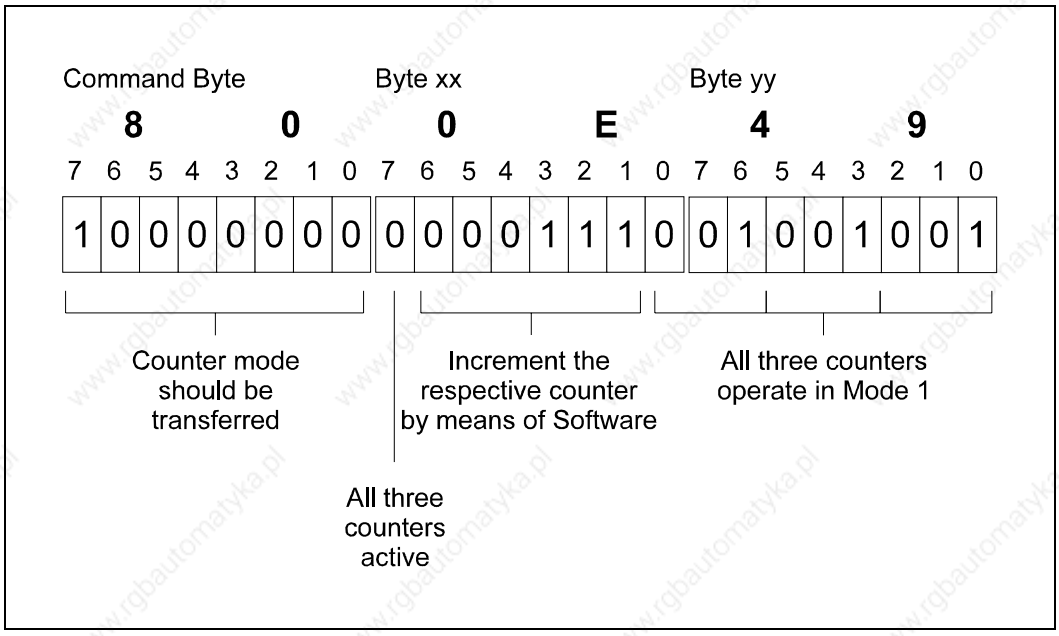
Example:

a) All three counters should be operated with mode 1.

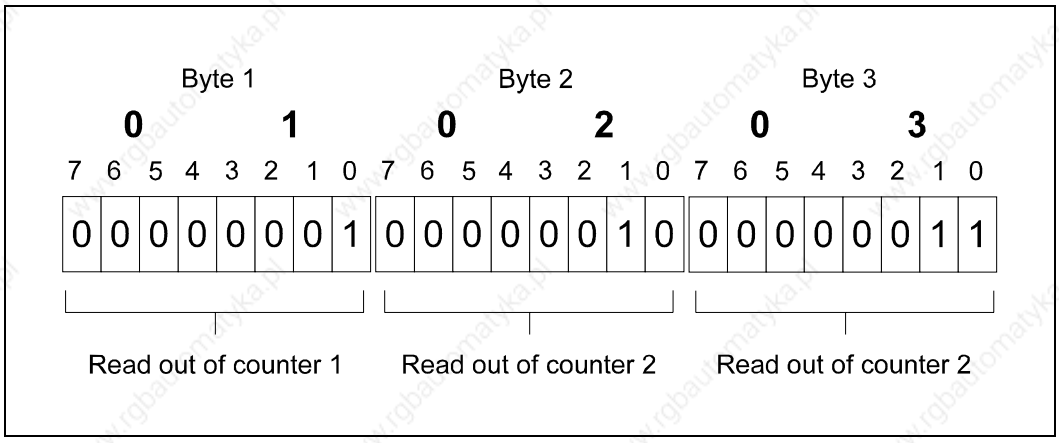
b) If the mode is programmed, the momentary counter reading should be read.

The following bytes are to be transferred:

a)



b)



From the building block 6 bytes can be read through the corresponding channel, whereby the counter's high-order byte is always in first place.

xx	high-order byte
	Counter state Counter 1
yy	low-order byte
xx	high-order byte
	Counter state Counter 2
yy	low order byte
xx	high-order byte
	Counter state Counter 3
yy	low-order byte

3.2.9 Analogue input (MD40-44) and output module (MD45-49)

3.2.9.1 General

The module for analogue input or analogue output enables the connection of analogue process interface equipment to the AG-115U, -135U and -155U. They convert the analogue signals with a resolution of 12 bits into a SPS-CPU digital signal and vice versa. The individual channels are polled and when requested by the SPS are directly transfered without delay. The interface's digital control is electrically decoupled.

The use of analogue modules will be supported from interface building block firmware number 4118V14 on.

The analogue input module possesses 8 inputs, the analogue output module 4 outputs. When using the module on a SSM-BG43, up to 12 output channels or 24 input channels are available. With a scanning period of 1 ms per channel, a maximum scanning period of

8 ms is produced by 8 inputs, if all channels are polled.

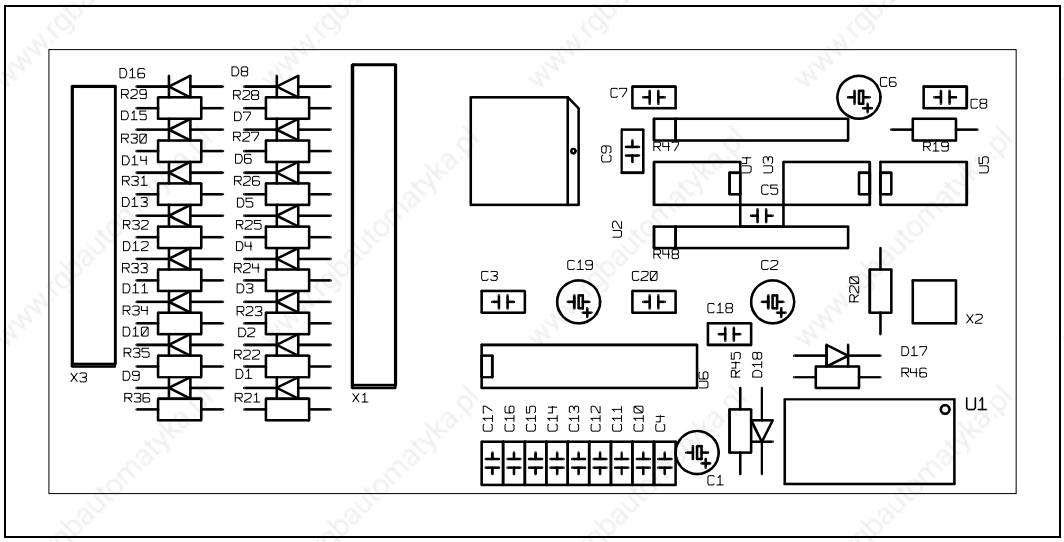


Fig. 3-37: Structure of the analogue input module

With the analogue output module, an undefined current is issued during the running up!

The following modules are presently available:

- MD40 Voltage input 8 channels, single ended, 12 bit, ±10V
- MD41 Voltage input 8 channels, single ended, 12 bit, ±5V or 0...10V
- MD42 Voltage input 8 channels, single ended, 12 bit, 0...5V
- MD43 Voltage input 8 channels, single ended, 12 bit, 0...20 or 4...20mA
- MD44 Connection from PT100, 4 channel temperature area: -200°C...850°C
- MD45 Voltage output 4 channels, 12 bit, ±10V
- MD46 Voltage output 4 channels, 12 bit, ±5V
- MD47 Voltage output 4 channels, 12 bit, 0...10V
- MD48 Voltage output 4 channels, 12 bit, 0...5V
- MD49 Current output 4 channels, 12 bit 0...20 or 4...20 mA

3.2.9.2 Data flow

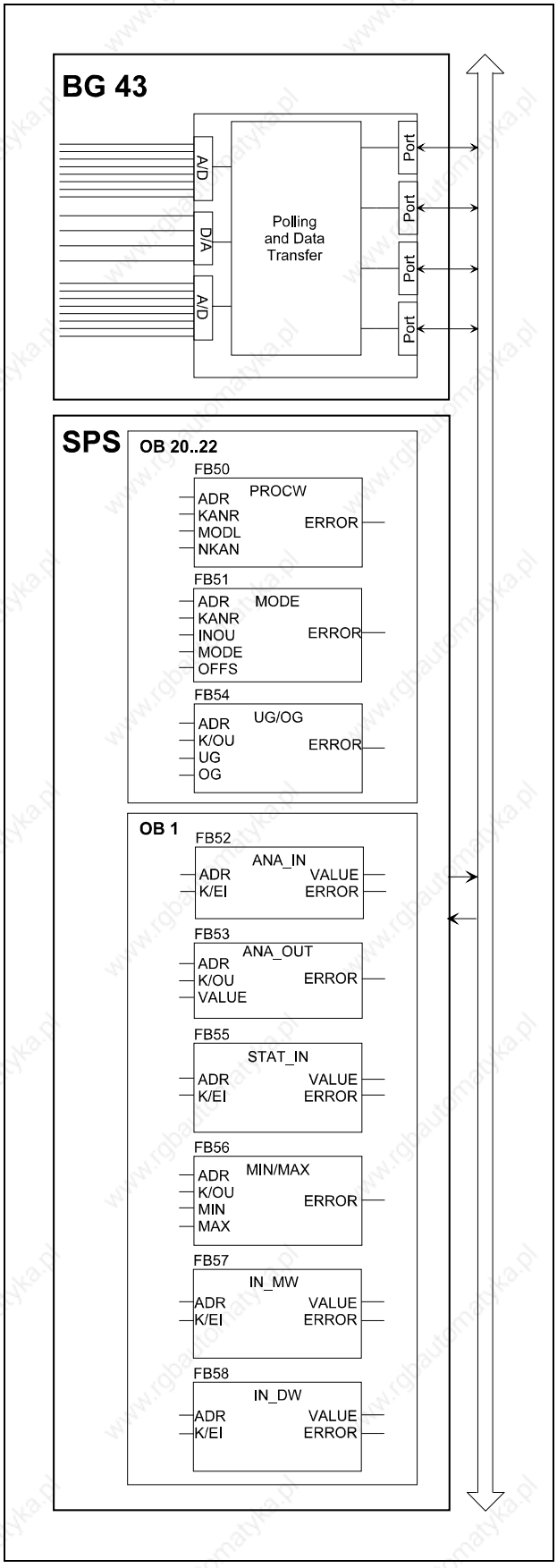


Fig. 3-38: Data flow of analogue input or analogue output module

3.2.9.3 Allocation of analogue module's terminal pins

Analogue input module

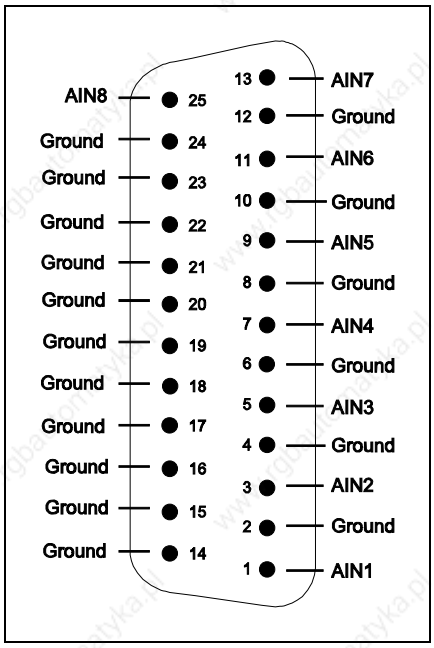


Fig. 3-39: Allocation of terminal pins when using an analogue input module

Analogue output module

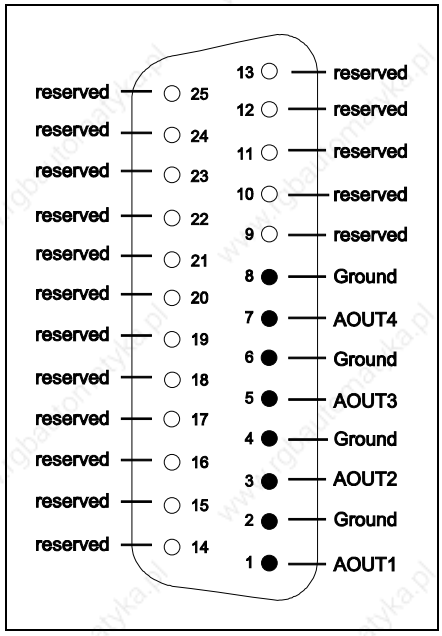


Fig. 3-40: Allocation of terminal pins when using an analogue output module

The reserved pins are not allowed to be connected!

3.2.9.4 Functional mode

Analogue input module

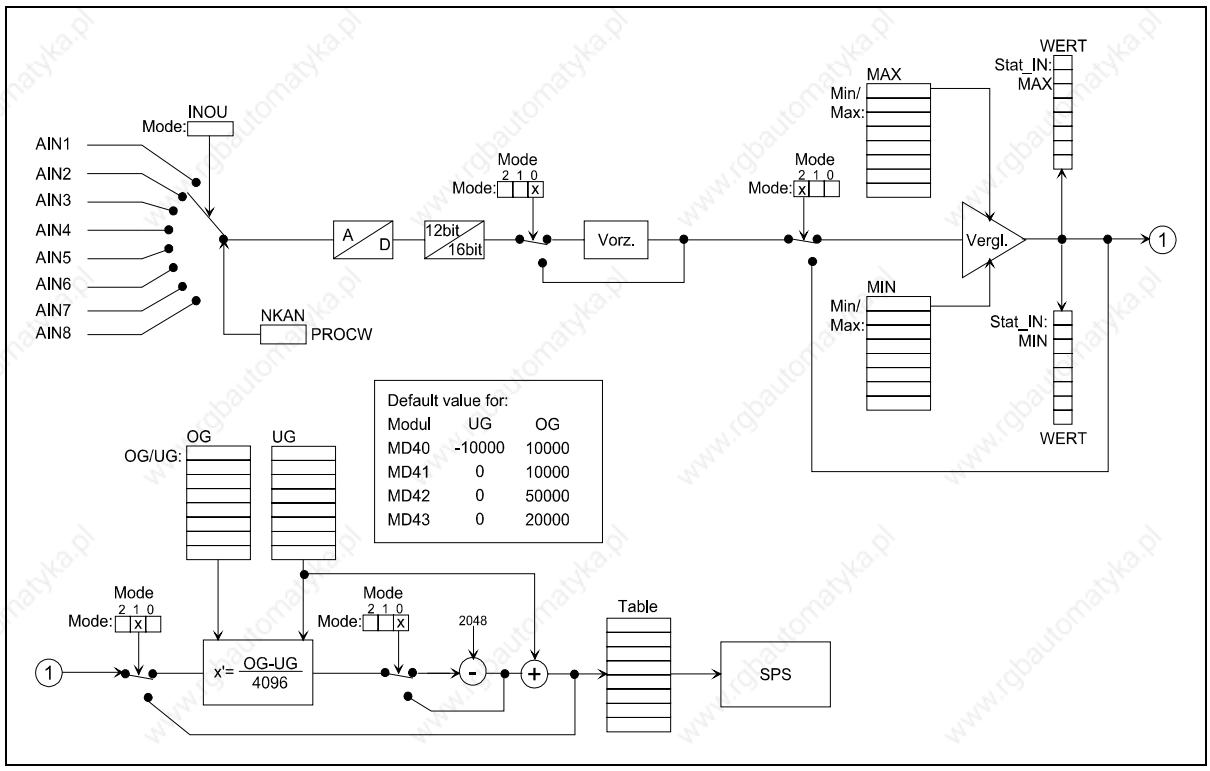


Fig. 3-41: Block diagram of an analogue input module

Analogue output module

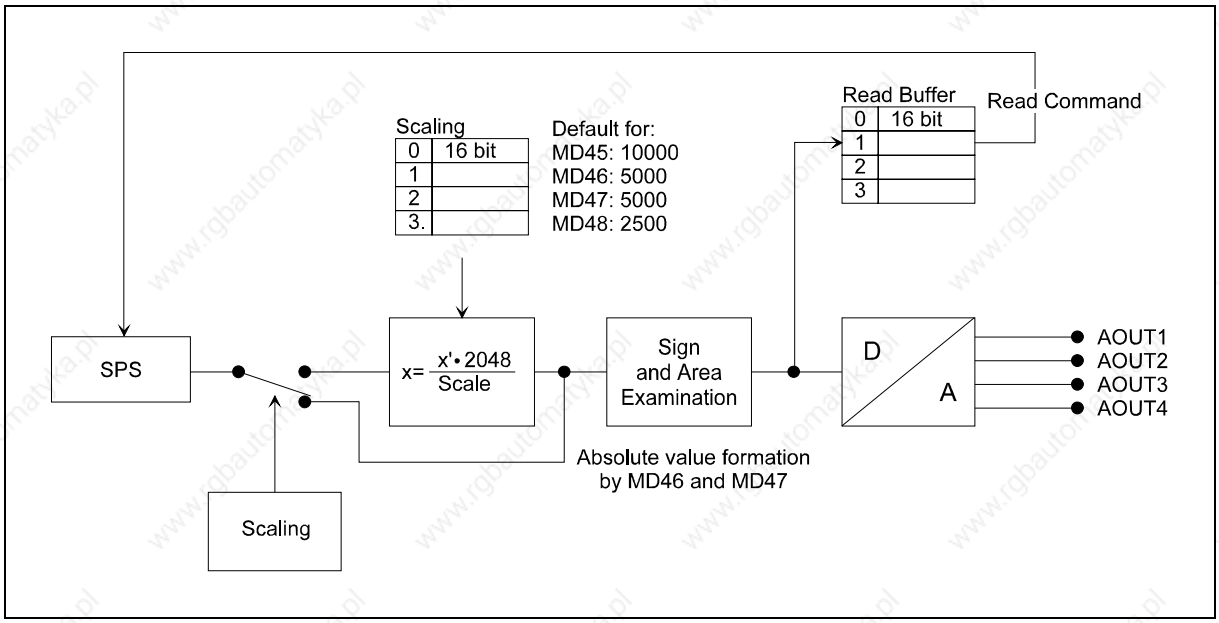


Fig. 3-42: Block diagram of an analogue output module

3.2.9.5 Application of analogue input/output with data handling blocks

Possible data handling blocks for analogue input

Initialization	FB50, FB51, FB54
Cycle	FB52, FB55, FB56, FB57, FB58

Possible data handling blocks for analogue output

Initialization	FB50, FB51, FB54
Cycle	FB53, FB56

3.2.9.5.1 FB50 (PROCW), procedure selection

The procedure and the module number are set with the FB50 (PROCW). The number of analogue channels which should be polled by the firmware, can be additionally set. This enables an increase in the polling rate of the used channels.

FB50 (PROCW) is to be load at the new start or restart (OB 20 ... 22) for each module.

The following parameters are to be transferred when loading FB 50:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
KANR	KF	Number of channels
MODL	KF	Module number
NKAN	KF	Maximum inputs
FEHL	BY	Error byte

Tab. 3-8: List of parameters fot the loading of FB50

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral base address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

KANR Number of channels, to which the module is connected. The used channel number is dependant upon the interface:

Building block BG41	Channel number 1
Building block BG42	Channel number 1, 2
Building block BG43	Channel number 1, 2, 3

MODL Number of modules, see order number or module imprint. Valid number are from 40 to 49.

NKAN Number of channels, which should be polled by the firmware. The channels must be occupied in ascending order. Valid numbers are 1-8. This parameter gives the end value of the channels to be polled. The channels 1 up to this value are always polled.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.

3.2.9.5.2 FB51 (MODE), mode selection

The data handling block FB51 (MODE) offers the possibility of inputting an offset and for selecting a mode for an input/output or for all inputs/outputs.

It is useful, to load the FB 51 (MODE) at the new start or restart (OB 20 ... 22). Fb51 is to be loaded for each channel.

The following parameters are to be transferred when loading FB51:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
KANR	KF	Number of channels
INOU	KF	Input/output number 1-8, 9
MODE	KM	Mode
OFFS	KF	Offset of outputs/inputs
FEHL	BY	Error byte

Tab. 3-9: List of parameters for the loading of FB51

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

KANR Number of channels, to which the module is connected. The used channel number is dependant upon the interface:

Building block BG41	Channel number 1
Building block BG42	Channel number 1, 2
Building block BG43	Channel number 1, 2, 3

INOU Analogue input or output (1-8). If 9 is entered here, then the values apply for all inputs or outputs of the module.

MODE Bit number 0 = 0 signed
1 unsigned
1 = 0 binary representation from Wandler -2048...2047 (0...4095)
1 normalised representation in mV or mA
2 = 0 no bottom limit for current measurement
1 bottom limit 4 mA

OFFS Offset to the zero point of the input/output

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.
1 Building block not ready
2 Building block does not answer

3.2.9.5.3 FB52 (ANA_IN), reading of analogue value

With this data handling block FB52 (ANA_IN) an analogue input can be read.

The following paramters are to be transferred when loading FB52:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
K/EI	KY	Number of channels / analogue input
WERT	W	Output value
FEHL	BY	Error byte

Tab. 3-10: List of parameters for the loading of FB52

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

K/EI K: Number of channels to which the module is connected (1-3).
EI: Number of analogue inputs (1-8)

WERT Flag word, in which the analogue value is filed.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.
1 Building block not ready
2 Building block does not answer

3.2.9.5.4 FB53 (ANA_OUT), giving the analogue value

With this data handling block FB53 (ANA_OUT) analogue values can be given out.

The following parameters are to be transferred when loading FB53:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
K/OU	KY	Number of the channels/ analogue output
WERT	W	Output value
FEHL	BY	Error byte

Tab. 3-11: List of parameters for the loading of FB53

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

K/OU

K: Number of channels to which the module is connected (1-3).

OU: Number of analogue outputs (1-4, 9)

By 9, the VALUE is given out to all 4 outputs of the module.

WERT Flag word, in which the analogue value is filed.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.

1 Building block not ready

There has to be an interval of at least 10 ms between access to the same channel or else there will be a misoperation!

3.2.9.5.5 FB54 (UG/OG), establishing the upper and lower levels

With this data handling block FB54 (UG/OG) the lower and upper levels for one input/output or for all inputs/outputs can be given. These values are used as a scaling.

The following parameters are to be transferred when loading FB54:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
K/OU	KY	Number of the channels / analogue outputs or inputs
UG	KF	Lower level for scaling
OG	KF	Upper level for scaling
FEHL	BY	Error byte

Tab. 3-12: List of parameters for the loading of the FB54

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

K/OU

K: Number of channels to which the module is connected (1-3).

OU: Number of analogue outputs or inputs (1-9)

At 9 the VALUE is set for all of the outputs or inputs of the module.

UG Lower level, which is indicated at the smallest input/output value.

OG Upper level, which is indicated at the highest input/output value.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.

1 Building block not ready

For a faultless functioning of the FB 54 it is required to set Bit 2 (lower limit 4mA) in FB51 under mode!

Transformating the analogue range to the scaled range

A sensor gives an analogue value. The firmware shows this value in the nominal range. The nominal range is hardware specific and is in the range of -2048 to 2047. You don't want to work with this values, but you want to predefine your own range, which is directy combined with the sensor values. The nominal range is shown to a normalized range. With setting the upper and lower limits the normalized range can be determined. Possible are numbers with a width of maximal 2 Byte (2^{15}=32768).

To preserve the sensor resolution always choose the possibly greatest numerical range for normalizing.

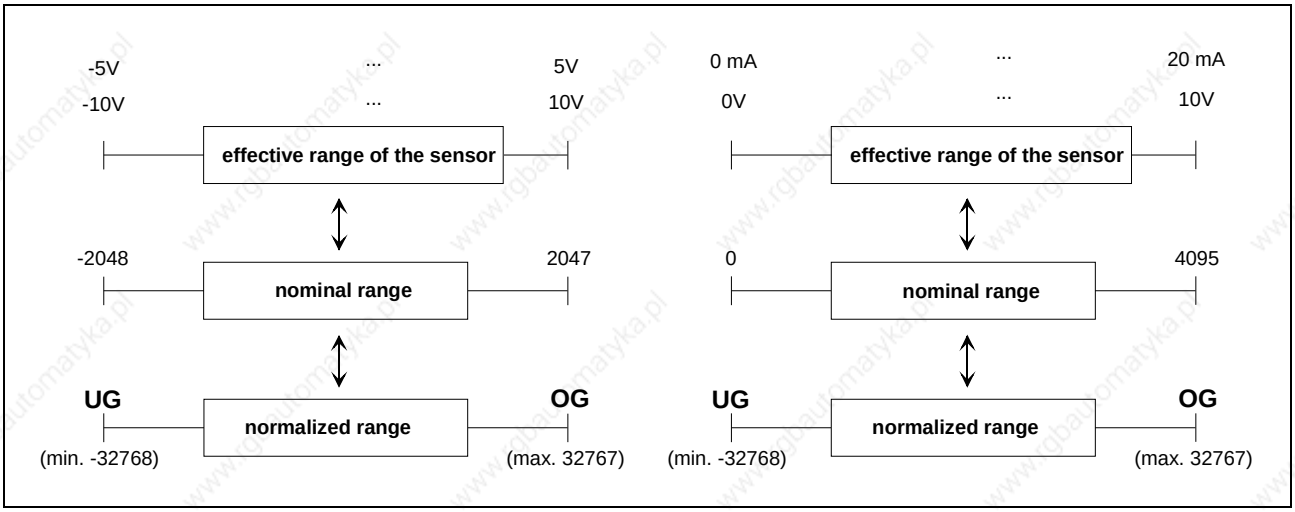


Fig. 3-43: Transformating the analogue range to the scaled range

Example

You set in a sensor (4...20mA), which shows the level of a container holding 30l.
How to define the upper and lower limits?

At 4mA 0l shall be shown. At 20mA 30l shall be shown.
Set 0 for lower limit and 30000 for upper limit (always choose the greatest number).

In case of wire breakage the lower limit will be undershoot and the data handling block will show an error message.

3.2.9.5.6 FB55 (STAT_IN), reading of the status bit

The status bit can be read with the data handling block FB55 (STAT_IN). This function is presently only available with the 4-20mA module (MD43).

The following parameters are to be transferred when loading FB55:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
K/EI	KY	Number of channels / analogue inputs
WERT	W	Status bits
FEHL	BY	Error byte

Tab. 3-13: List of parameters for the loading of FB55

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

K/EI
K: Number of channels to which the module is connected (1-3).
EI: Number of analogue inputs (1-8)

WERT Flag word, in which the status bits are filed.
LB MAX-Status, RB MIN-Status.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.
1 Building block not ready
2 Building block does not answer

The following bytes of the inputs E1 to E8 are returned:

Bit. No.	7	6	5	4	3	2	1	0
MIN Status	E8	E7	E6	E5	E4	E3	E2	E1

Bit. No.	7	6	5	4	3	2	1	0
MAX Status	E8	E7	E6	E5	E4	E3	E2	E1

3.2.9.5.7 FB56 (MIN/MAX), determinig the min. and max. values

With this data handling block FB56 (MIN/MAX) a minimum and a maximum value can be determined for one input/output or for all inputs/outputs.

The following parameters are to be transferred when loading FB56:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
K/OU	KY	Number of channels / analogue input
MIN	W	Minimum value
MAX	W	Maximum value
FEHL	BY	Error byte

Tab. 3-14: List of parameters for the loading of FB56

The following details are necessary for the abovementioned parameters:

ADR (absolut) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2.
Further details can be found in chapter 5.

K/OU
K: Number of channels to which the module is connected (1-3).
EI: Number of analogue outputs or inputs (1-9)
At 9 the VALUE for all of the module's outputs or inputs are set.

MIN Flag word, in which the minimum value is filed.

MAX Flag word, in which the maximum value is filed.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.
1 Building block not ready

3.2.9.5.8 FB57 (IN_MW), reading of analogue value in the flag word

With this data handling block FB57 (IN_MW) one or all analogue inputs can be read.

The following parameters are to be transferred when loading FB57:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
K/EI	KY	Number of channels / analogue input
WERT	W	Output value
FEHL	BY	Error byte

Tab. 3-15: List of parameters for the loading of FB57

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

K/EI K: Number of channels to which the module is connected (1-3).
EI: Number of analogue inputs (1-9)
At 9 all of the module's analogue inputs are read.

WERT Flag word, in which the analogue value is filed.
If it is entered at EI 9, all input values are filed from the flag word on, which has been entered in WERT (VALUE).

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.

1 Building block not ready

2 Building block does not answer

3.2.9.5.9 FB58 (IN_DW), reading of analogue values in the data word

With this data handling block FB58 (IN_DW) one or all analogue inputs can be read. The data block is to be opened before loading the FB.

The following parameters are to be transferred when loading the FB58:

Des.	Format	Explanation
ADR	KF	Peripheral base address of the building block
K/EI	KY	Number of channels / Analogue input
WERT	W	Output value
FEHL	BY	Error byte

Tab. 3-16: List of parameters for the loading of FB58

The following details are necessary for the abovementioned parameters:

ADR (peripher) Input of the peripheral address, under which the building block can be controlled by the PLC. This address can be set with the DIL switches S1 and S2. Further details can be found in chapter 5.

K/EI

K: Number of channels to which the module is connected (1-3).

EI: Number of analogue inputs (1-9)

At 9 all of the module's analogue inputs are read.

WERT Data word, in which the analogue value is to be filed.

If it is entered at EI 9, all input values from the data word given in WERT are filed in the opened DB.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.

1 Building block not ready

2 Building block does not answer

3.2.9.5.10 Example

Start OB

```
OB21

         SPA FB50
NAME     #PROCW
ADR      :KF 128             Building block's peripheral address
KANR     :KF 1               Channel number
MODL     :KF 40              Module number
NKAN     :KF 3               Maximum of inputs to be polled
FEHL     :MB127              Error code in flag byte 127
         :
         :SPA FB 59
         #Timer
         :
         :SPB FB 51
NAME     #MODE
ADR      =KF +128            Base address of the SSM, PY128
KANR     =KF +1              Channel number of the module
INOU     =KF +1              Analogue input number 1
MODE     =KM 00000000 00000010 Signed, normalised depiction in mV, no limit
OFFS     =KF +0              Offset of I/O
FEHL     =MB 55              Error code to be filed in MB 55
         :
         :SPA FB 59
         #Timer
         :
         SPB FB 54
NAME     #UG/OG
ADR      =KF +128            Base address of the SSM, PY128
K/O      =KY 1,3             Channel number / input, output number
UG       =KF -20             Lowest level for scaling
OG       =KF +2000           Highest level for scaling
FEHL     =MB 19              Error byte

         :SPA FB 59
         #Timer
         :
```

There has to be a timer with 50 ms between each function call of FB FB50, FB51 and FB54 for correct funktion of the module.

The timer has to be programmed by the user.

Cycle OB

```
OB1

         :SPA FB 52
NAME     #ANA_IN
ADR      =KF +128            Base address of the SSM PY128
K/EI     =KY 1,1             Module's channel number, number of the input
                             channel
WERT     =MW 20              Flag word MW20 for the analogue input value
FEHL     =MB 55              Error code to be filed in MB55

         :SPA FB 57
NAME     #IN_MW
ADR      =KF +128            Base address of the SSM PY128
K/EI     =KY 1,1             Module's channel number, number of the input
                             channel
WERT     =MW 20              Flag word MW20 for the analogue input value
FEHL     =MB 55              Error code to be filed in MB55
         :A DB20             Open data block DB20
         :SPA FB 58
NAME     #IN_DW
ADR      =KF +128            Base address of the SSM PY128
K/EI     =KY 1,1             Module's channel number, number of the input
                             channel
WERT     =DW 20              Data word DW20 for the analogue input value
FEHL     =MB 55              Error code to be filed in MB55

         :SPB FB 53
NAME     #ANA_OUT
ADR      =KF +128            Base address of the SSM, PY128
K/OU     =KY 1,1             Module's channel number, number of the output
                             channel
WERT     =MW 52              Flag word MW52 for the analogue output value
FEHL     =MB 54              Error code to be filed in MB54
```

```
          SPA FB 55
NAME      #STAT_IN
ADR       =KF +128            Base address of the SSM, PY128
K/EI      =KY 1,2             Channel number / input, output number
WERT      =MW 30              Returning of the status bits
FEHL      =MB 18              Error byte
If the min. or max. value falls down or up, a corresponding status bit is set.

          SPB FB 56
NAME      #MIN/MAX
ADR       =KF +128            Base address of the SSM, PY128
K/OU      =KY 1,1             Channel number / input, output number
MIN       =MW 32              Smallest valid value
MAX       =MW 34              Largest valid value
FEHL      =MB 17              Error byte
The FB can be loaded in start OB or bit-controlled in OB1.
```

3.2.9.6 Application of analogue input/output modules without data handling blocks

3.2.9.6.1 Parameter channel

Further information for the application of the interface without data handling blocks can be found in chapter 4.2.

Select the procedure with command 5.

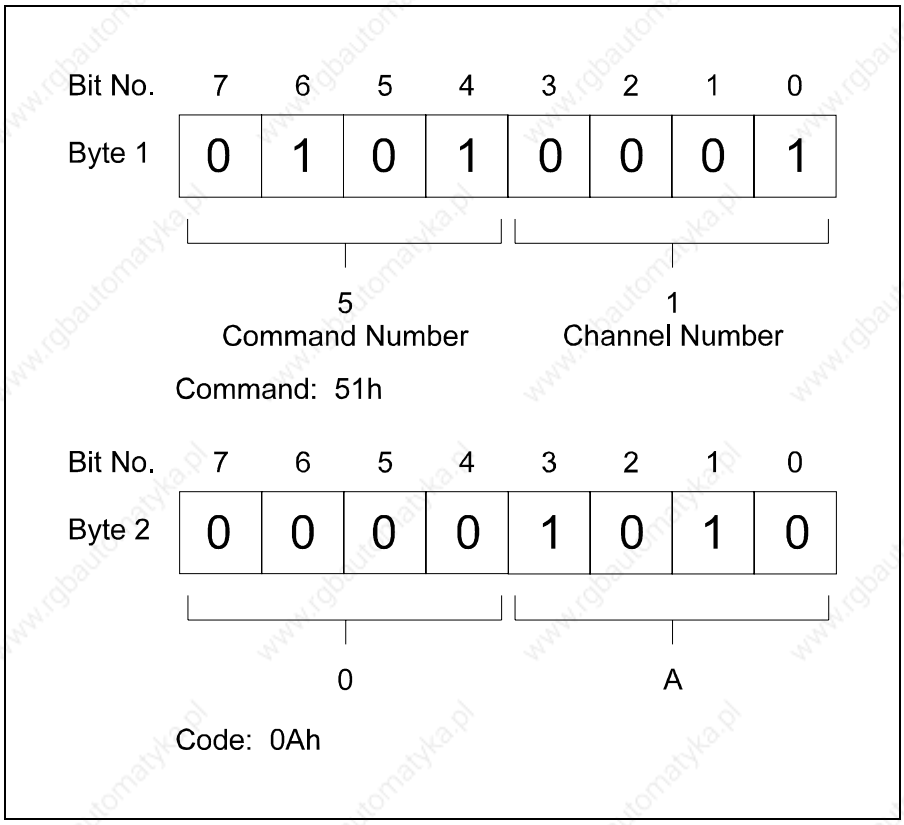


Fig. 3-44: Function selection of analogue input/analogue output with command 5

3.2.9.6.2 Commands for the data channel

Use the data channel for transferring the commands.

Command overview

1	Reading of the analogue value	1 byte command
2	Writing of the analogue value	3 byte command
3	Setting of the Mode	3 byte command
4	Selection of Mode	2 byte command
5	Scaling	5 byte command
6	Reading the status	1 byte command
7	Writing min/max	5 byte command

Command 1 - Reading the analogue value

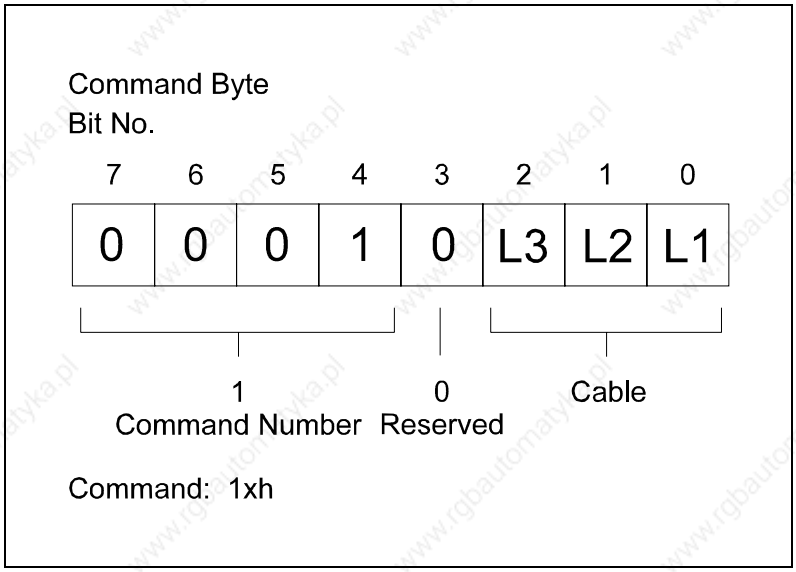


Fig. 3-45: Reading of the analogue value of the analogue input/output with command 1

The line 0 .. 7 is assigned to the inputs or outputs.

The following bytes are returned as follows:

Bit No.	7	6	5	4	3	2	1	0
LSB	D7	D6	D5	D4	D3	D2	D1	D0

Bit No.	7	6	5	4	3	2	1	0
MSB	D15	D14	D13	D12	D11	D10	D9	D8

If this command is used on the output module, one receives the output's set value in binary format.

Command 2 - Writing the analogue value

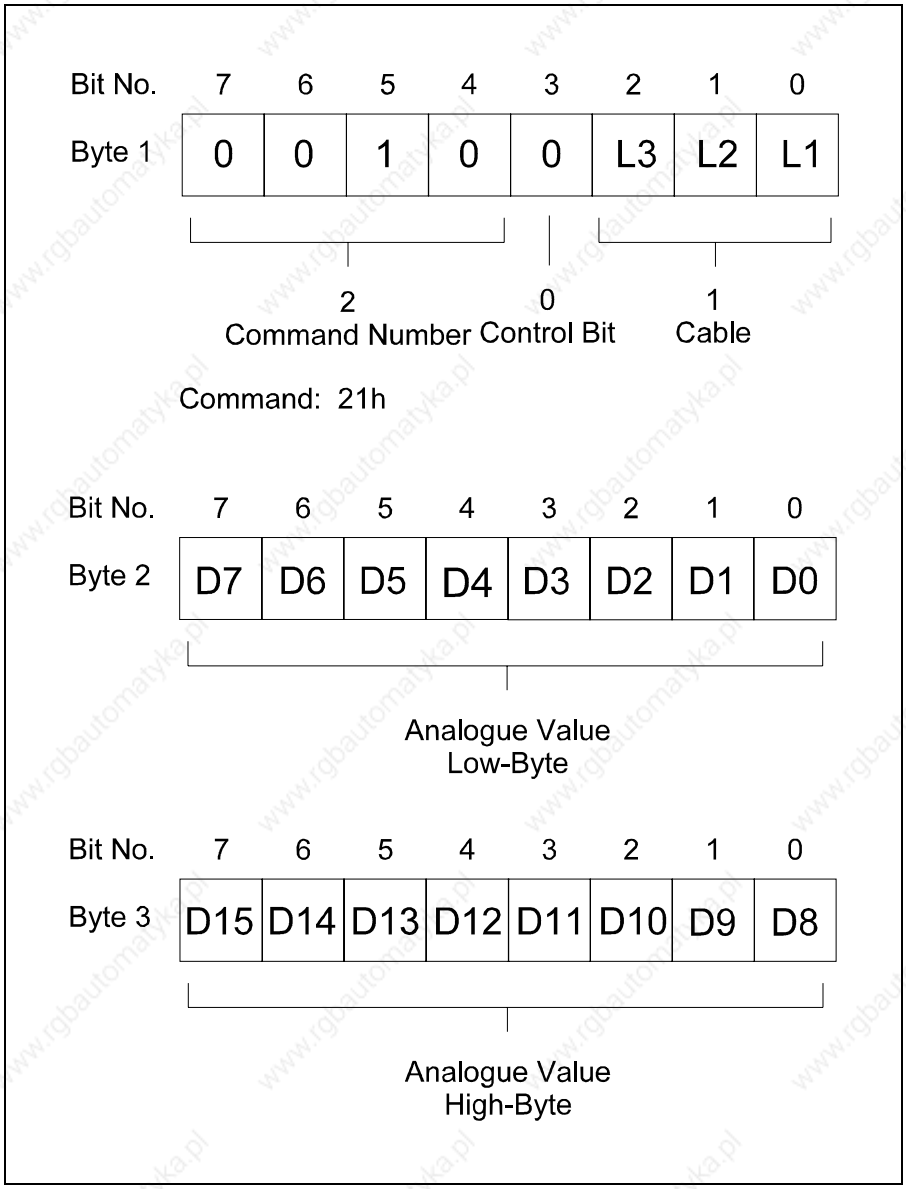


Fig. 3-46: Writing the analogue value of the analogue input/output with command 2

Bit 3 is used as the control bit for block commands. If this bit is set at 1, the command applies for all lines.

Command 3 - Setting of the mode

When setting the mode two further bytes are transferred as an argument.

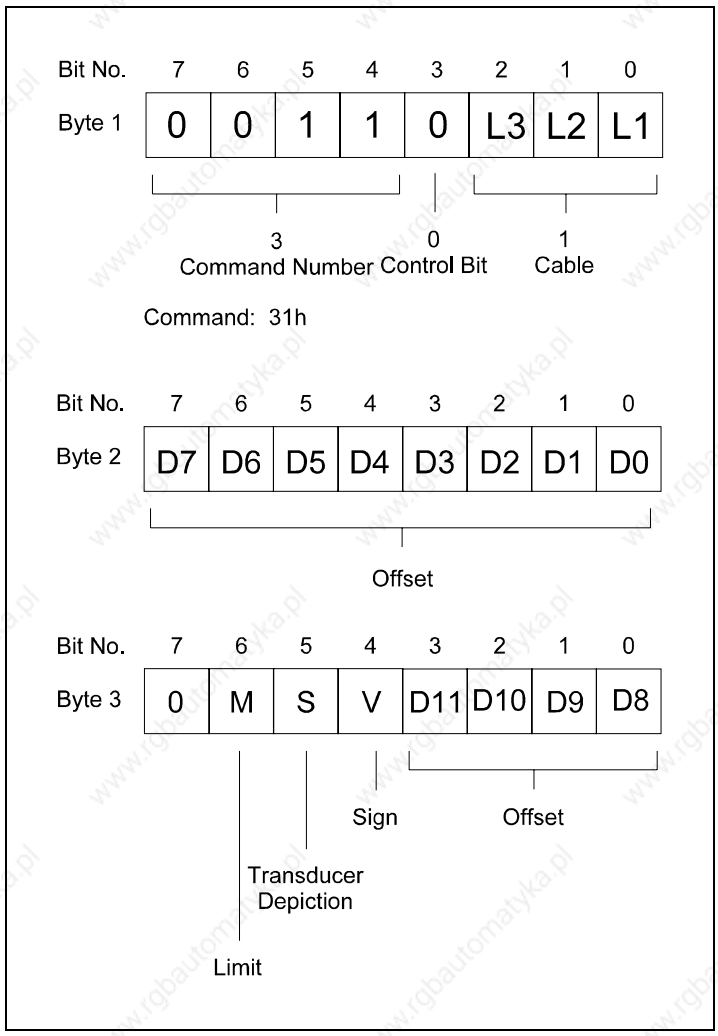


Fig. 3-47: Setting of the mode for the analogue input/output with command 3

Bit 3 is used as the control bit for block commands.
If this bit is set at one, this command applies to all lines.
Offset: The offset carries out a zero balance (11 bits signed). D11= Sign
The mode bits are occupied as follows:

Bit 7 is not defined and should be set to 0.

Bit 6 (M)	0	no bottom limit for current measurement
	1	bottom limit 4 mA
Bit 5 (S)	0	binary depiction from Wandler (Range from -2048...+2047 or 0...4095)
	1	normalised depiction in [mV] or [µA] (dependant on the module and how it is set)
Bit 4 (V)	0	signed
	1	unsigned

Command 4 - Module selection command

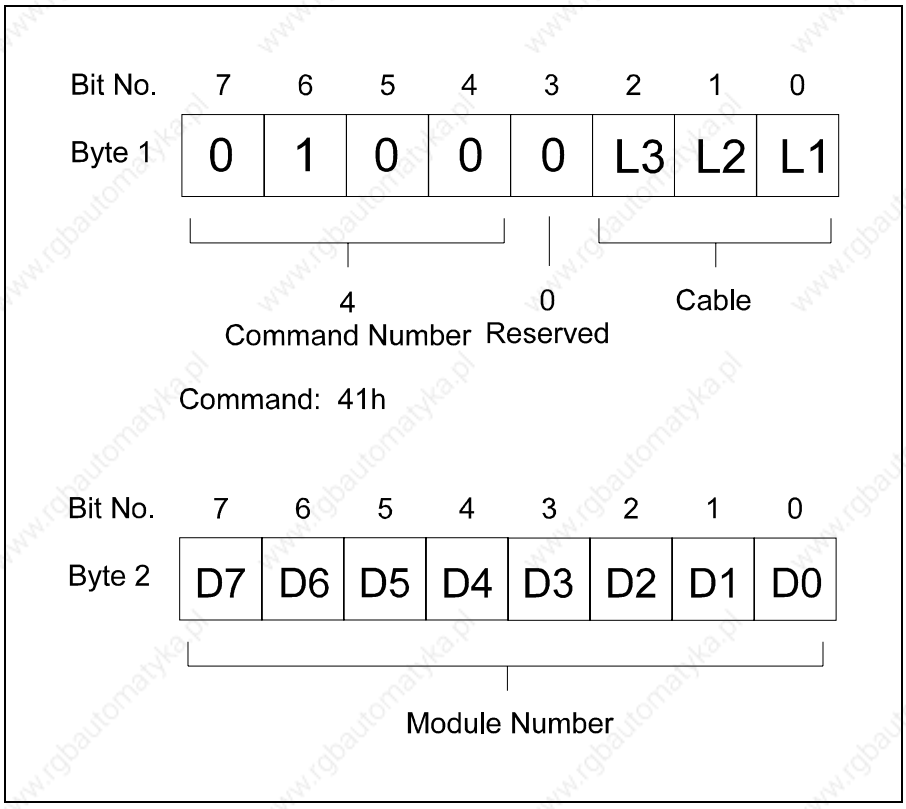


Fig. 3-48: Module selection of the analogue input/output with command 4

Line 0...2 Determines the number of lines to be polled.

Module Number The module number corresponds to the last numeral of the module's order number.
Example: The 10V input module has the order number VIPA SSM-MD40. Enter the module number 40dez.

Command 5 - Scaling

With scaling four extra bytes are transferred as an argument.

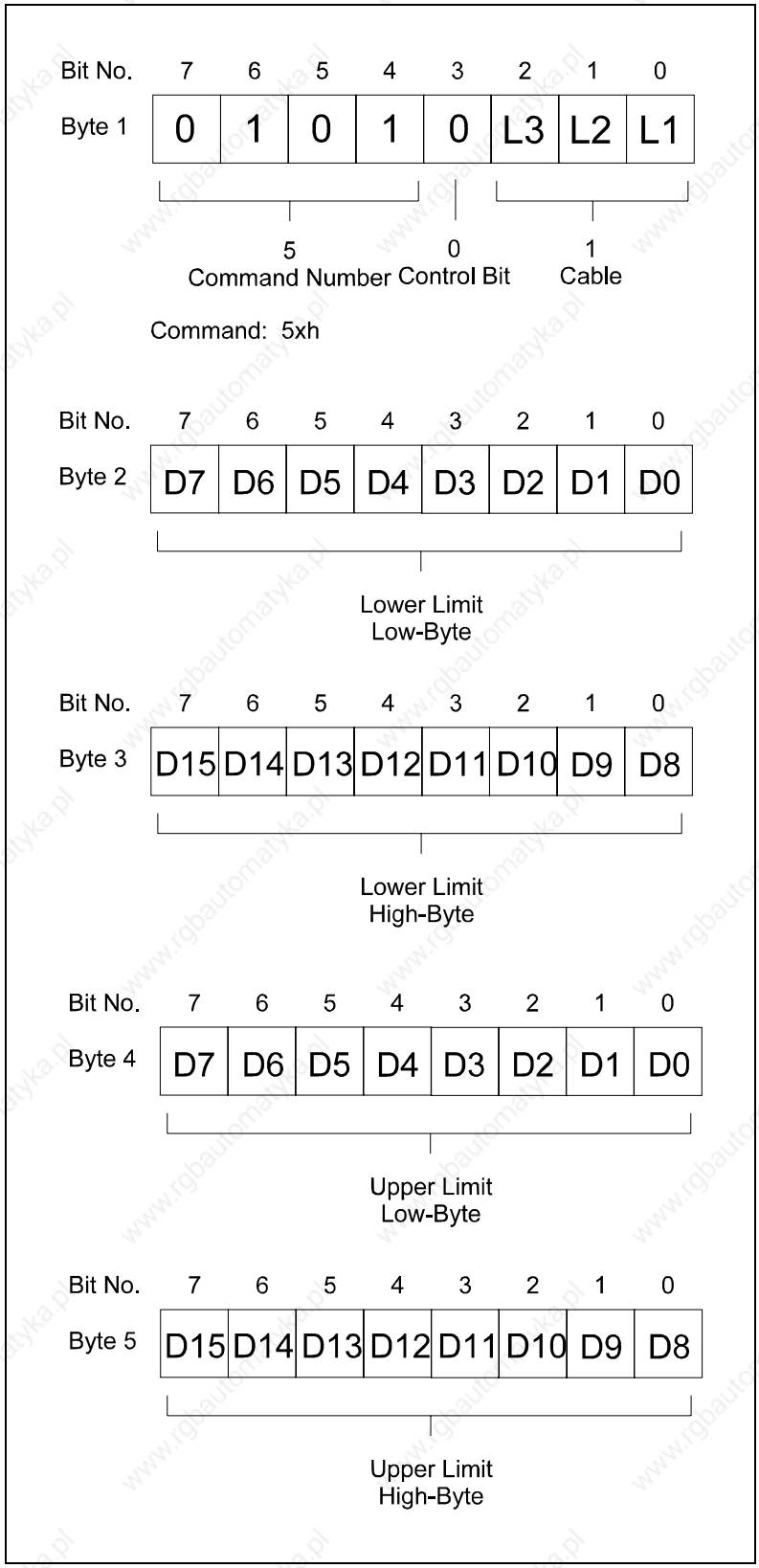


Fig. 3-49: Scaling of an analogue input/output with command 5

Command 6 - Reading the status

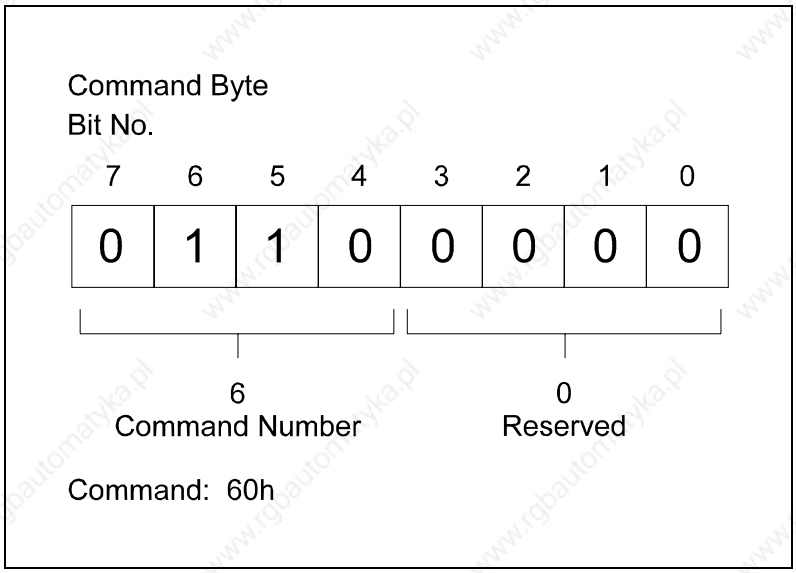


Fig. 3-50: Reading the status of the analogue input/output with command 6

The following bytes are sent back in the following way:

Bit No.	7	6	5	4	3	2	1	0
MIN Status	E7	E6	E5	E4	E3	E2	E1	E0

Bit No.	7	6	5	4	3	2	1	0
MAX-Status	E15	E14	E13	E12	E11	E10	E9	E8

Command 7 - Writing of the MIN/MAX limits

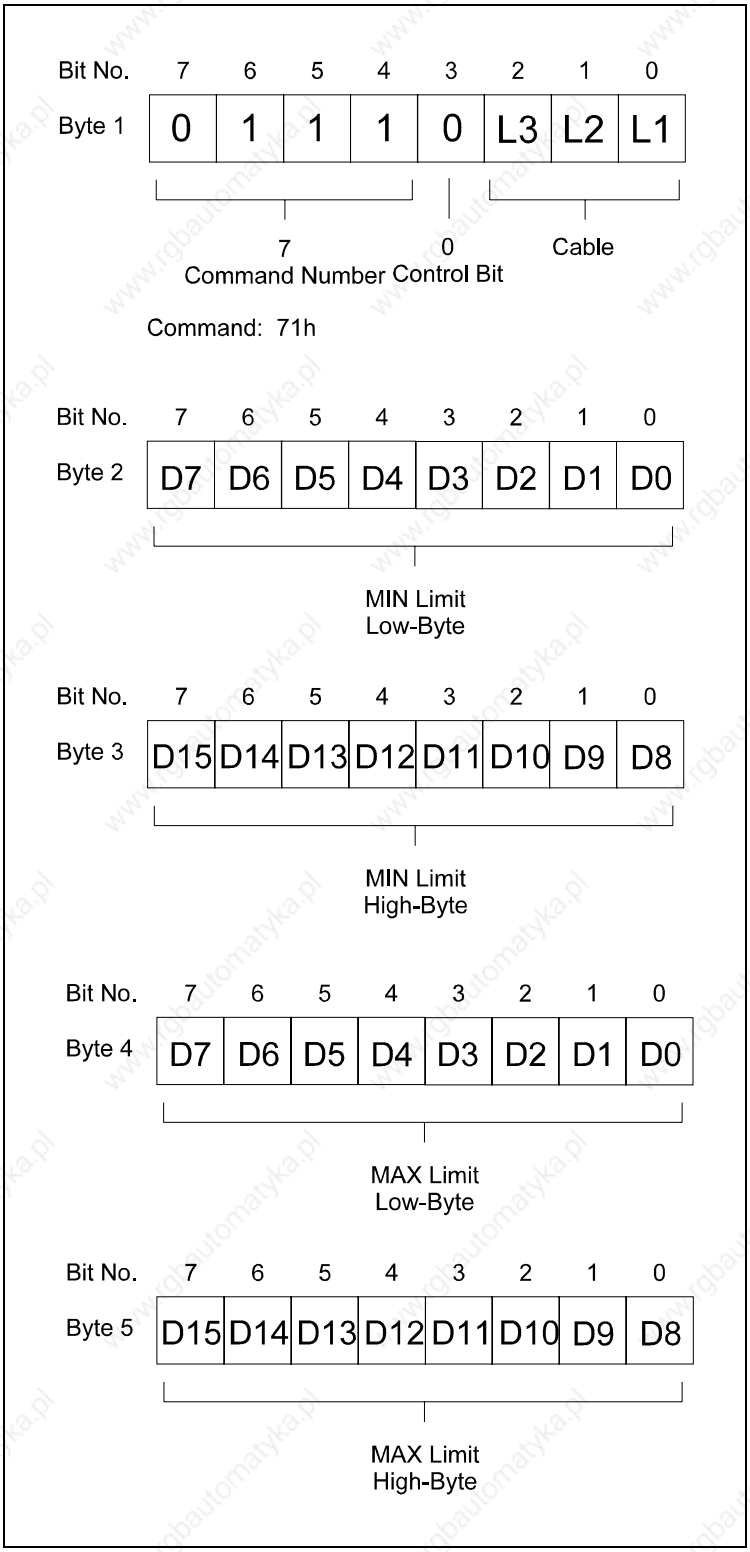


Fig. 3-51: MIN/MAX levels of the analogue input/output with command 6

Bit 3 is used as the control bit for block commands.
If this bit is set at one, the command applies for all lines.

3.2.9.7 Temperature module for PT100 (MD44)

3.2.9.7.1 General

The temperature module PT100 enables the connection of different types of PT100 temperature sensors with the AG-115U,-135U and-155U. The modul delivers a 16 Bit integer value proportional to the temperature sensor's resistor. This value is definitely assigned to the sensor's temperature.

When using this module on the SSM-BG43, up to 12 PT100 channels are available.

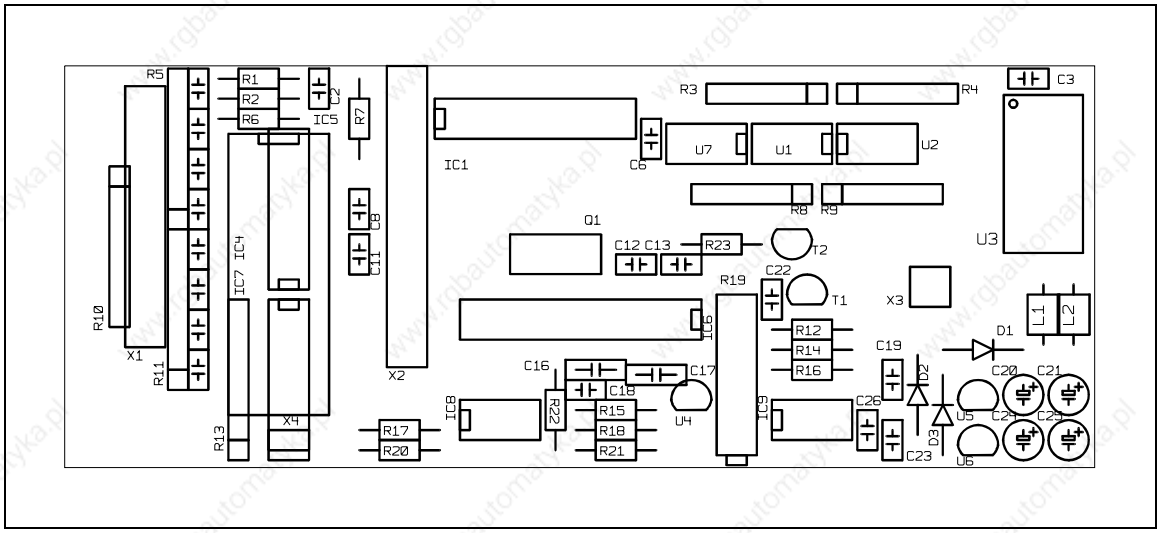


Fig. 3-52: Structure of the temperature module for PT100

3.2.9.7.2 Allocation of the PT100 module's terminal pins

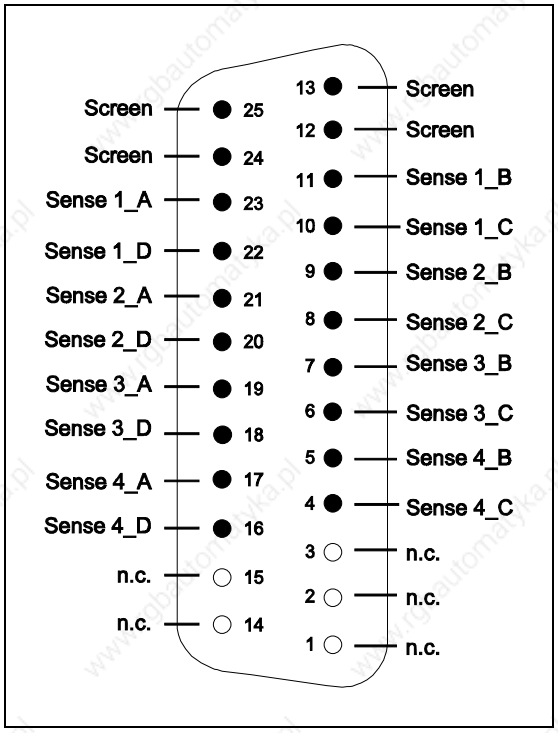


Fig. 3-53: Allocation of the temperature module's terminal pins for PT100

3.2.9.7.3 Connection example

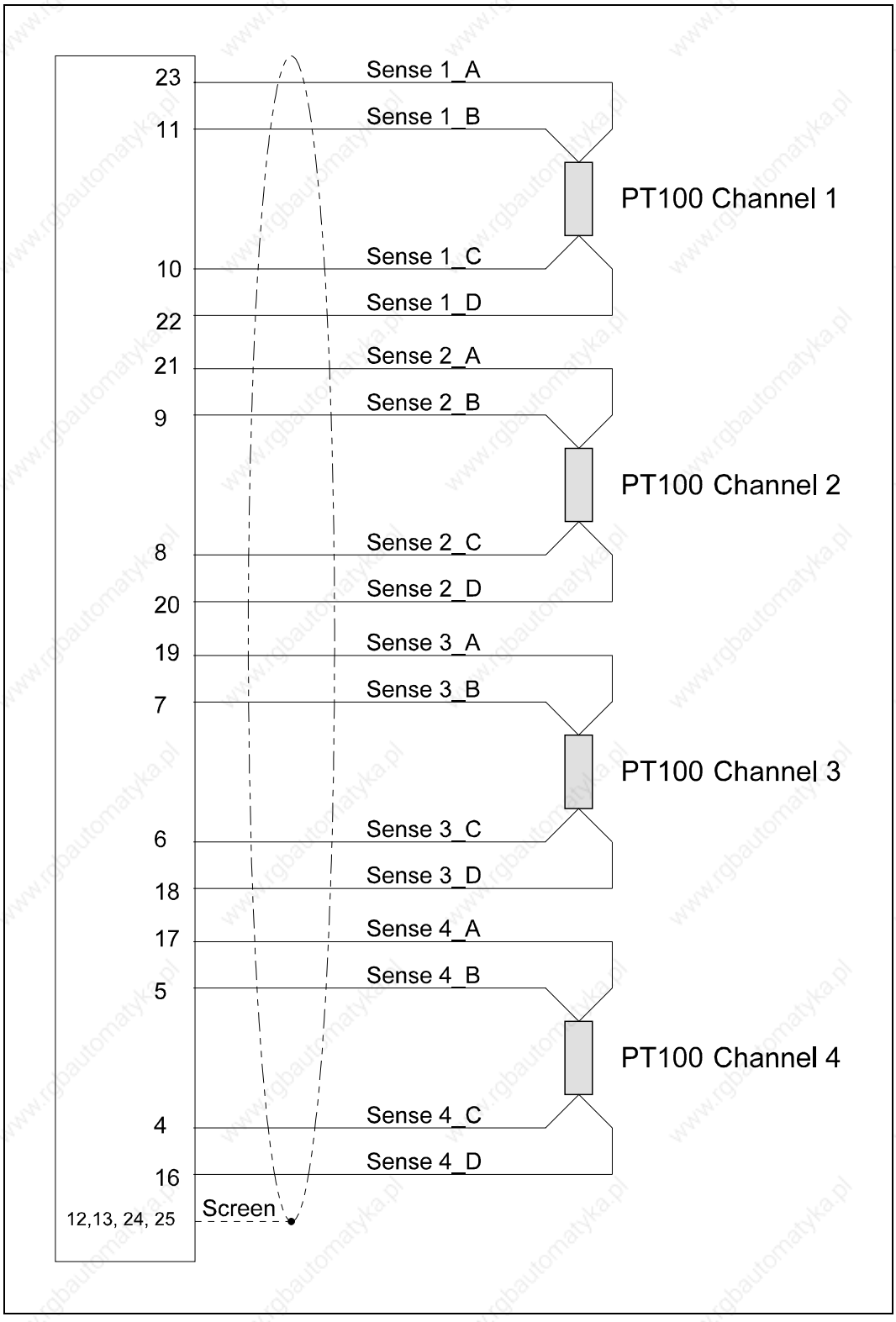


Fig. 3-54: An application example of the temperature module

3.2.10 DCF77 antenna module (MD36)

3.2.10.1 General

The Federal Institute of Physical Technology in Braunschweig operates a cesium atomic frequency standard with an arithmetical response deviation of 1 second in 1 million years. This clock time is coded (DCF77) and is irradiated from an LF transmitter in Mainflingen near Frankfurt.

The time signal irradiated from the transmitter in Mainflingen can be picked up by the intelligent antenna DCF77.

The antenna possesses its own processor for transmitting the decoded time signal to a 20mA interface. Every 20mA interface can be used as a communication module, for example, the SSM-MD25 and the SSM-MD26 modules.

Every full hour the antenna module synchronizes with the quartz clock on the interface building block, in order to provide the officially established time and not a manipulated one.

The antenna module is not an independant clock. It merely serves as the synchronisation of existing real-time clocks.

By means of the standard interface and the open time telegram, it is possible to connect this module to any other system with a 20mA interface. Due to the receiver evacuation, this module possesses a greater signal-to-interference ratio. It is also possible to supply several suscribors in a chain parameter matrix form.

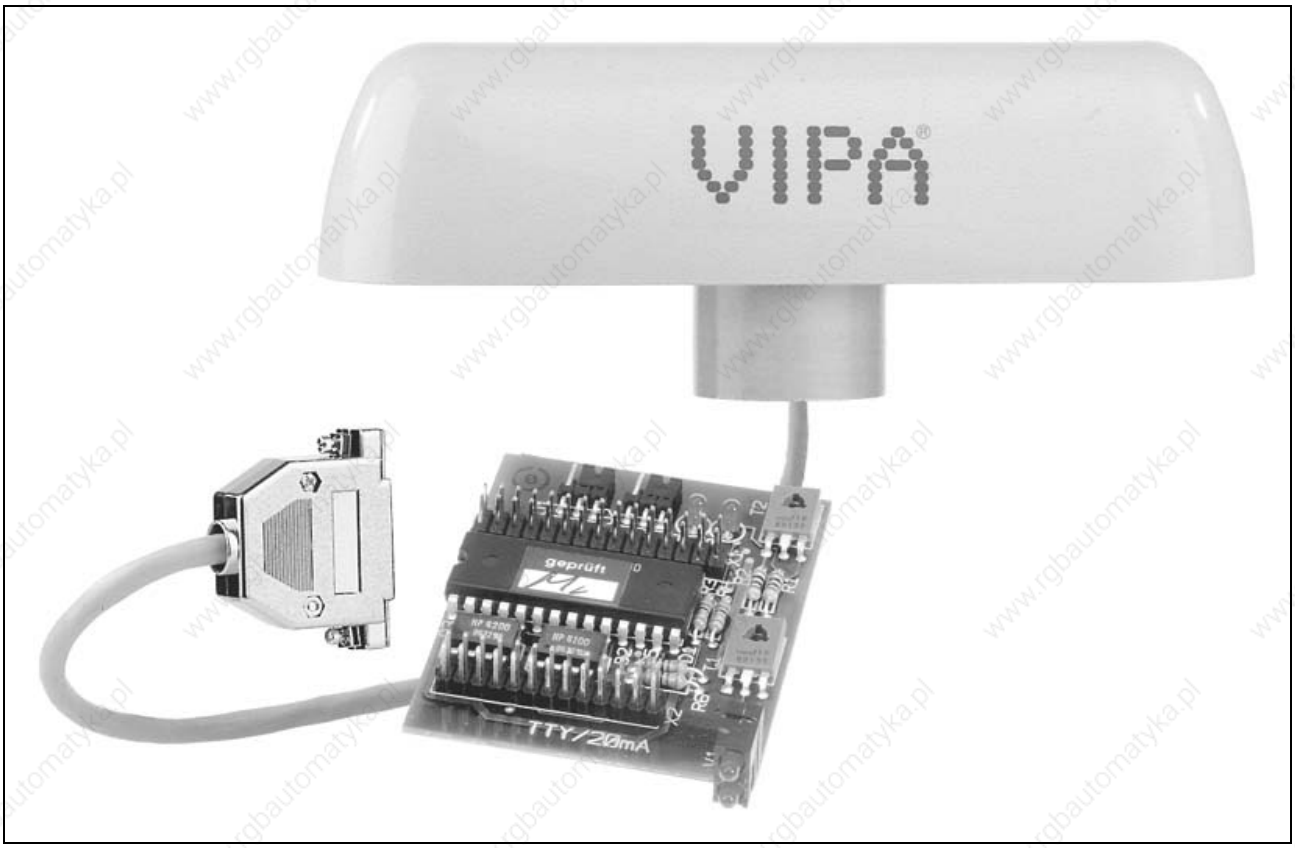


Fig. 3-55: Combination DCF77 antenna with 20ma current loop module

3.2.10.2 Transmitter's location

The transmitter which belongs to the Federal Institute for Physical Technology, Braunschweig is located in Mainflingen near Frankfurt. Like a radio, the reception is greatly dependent upon place and location. However, within a radius of 1500 km around Frankfurt it is generally possible to get a reception without any difficulties.

Fig. 3-56: Location of transmitter for the DCF77 signal

3.2.10.3 The module's dimensions

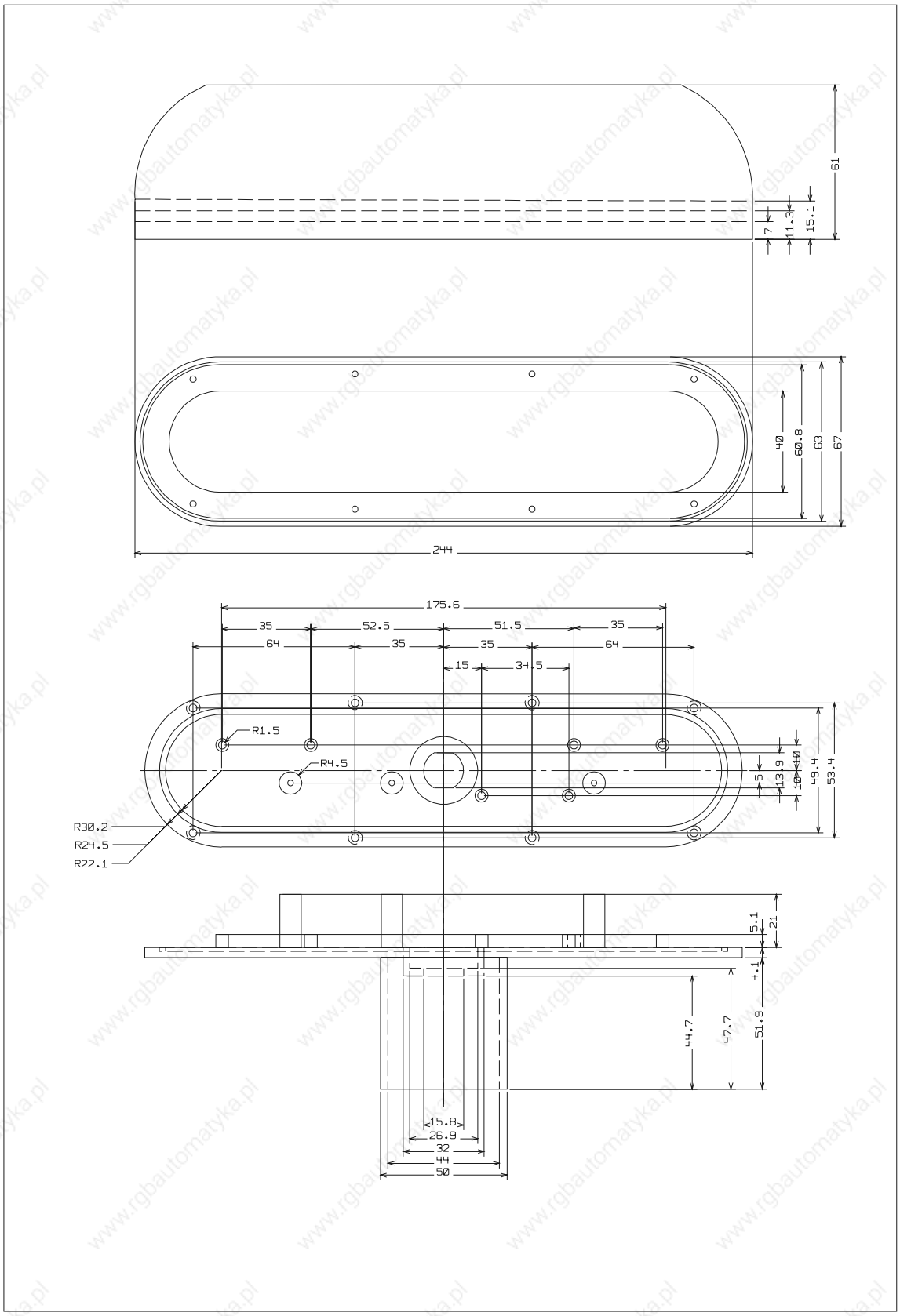


Fig. 3-57: The dimensions of the DCF77 antenna module

3.2.10.4 Allocation of DCF77 antenna module's terminal pins

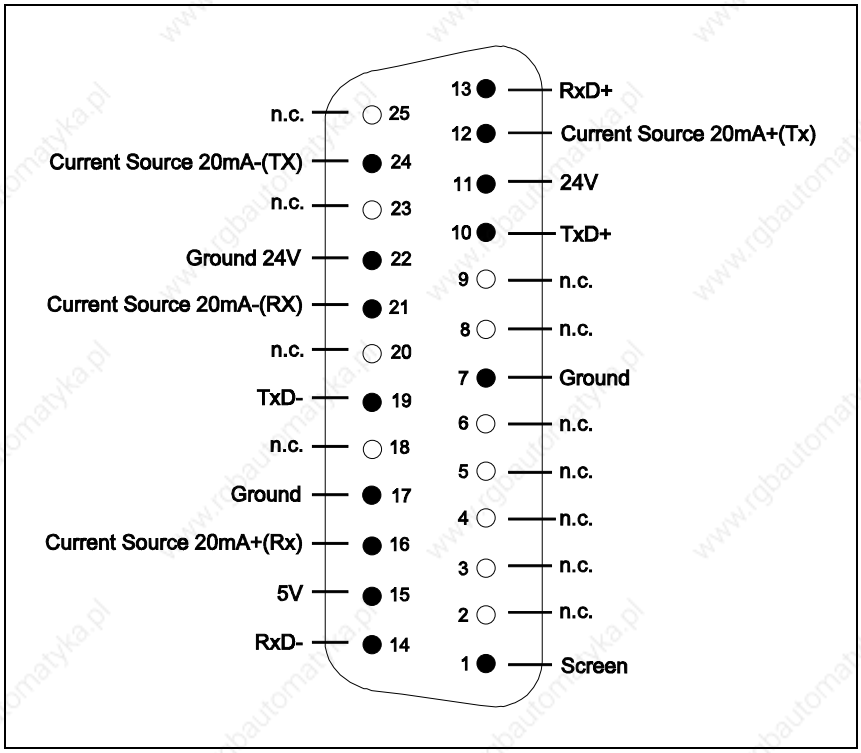


Fig. 3-58: Allocation of the DCF77 antenna module's terminal pins

3.2.10.5 The time plan

The following figure shows the temporal course of transmtting the coded time information from the German time signal transmitter.

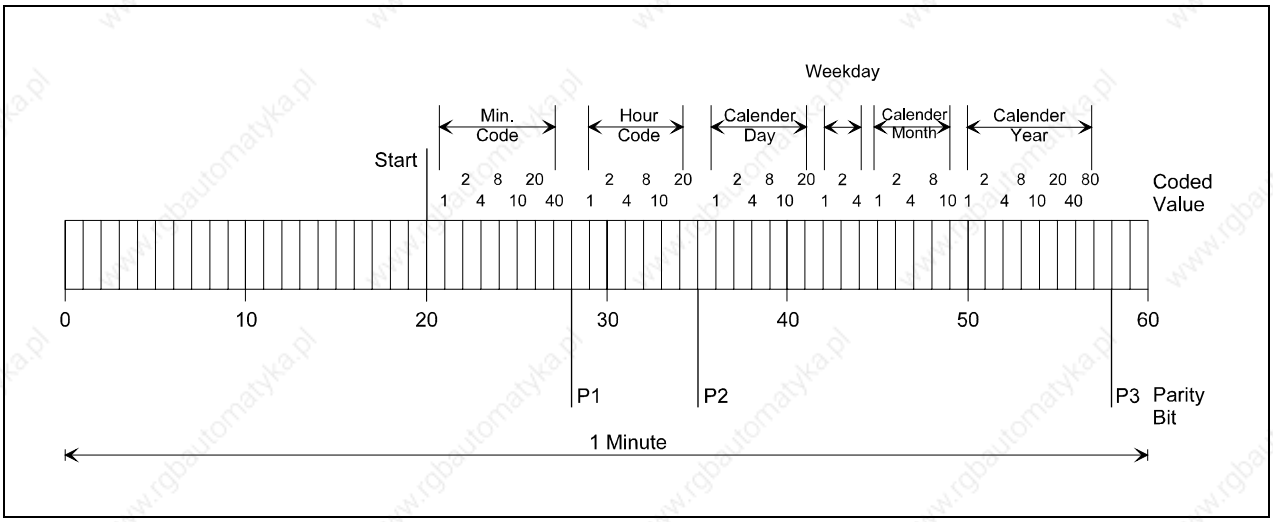


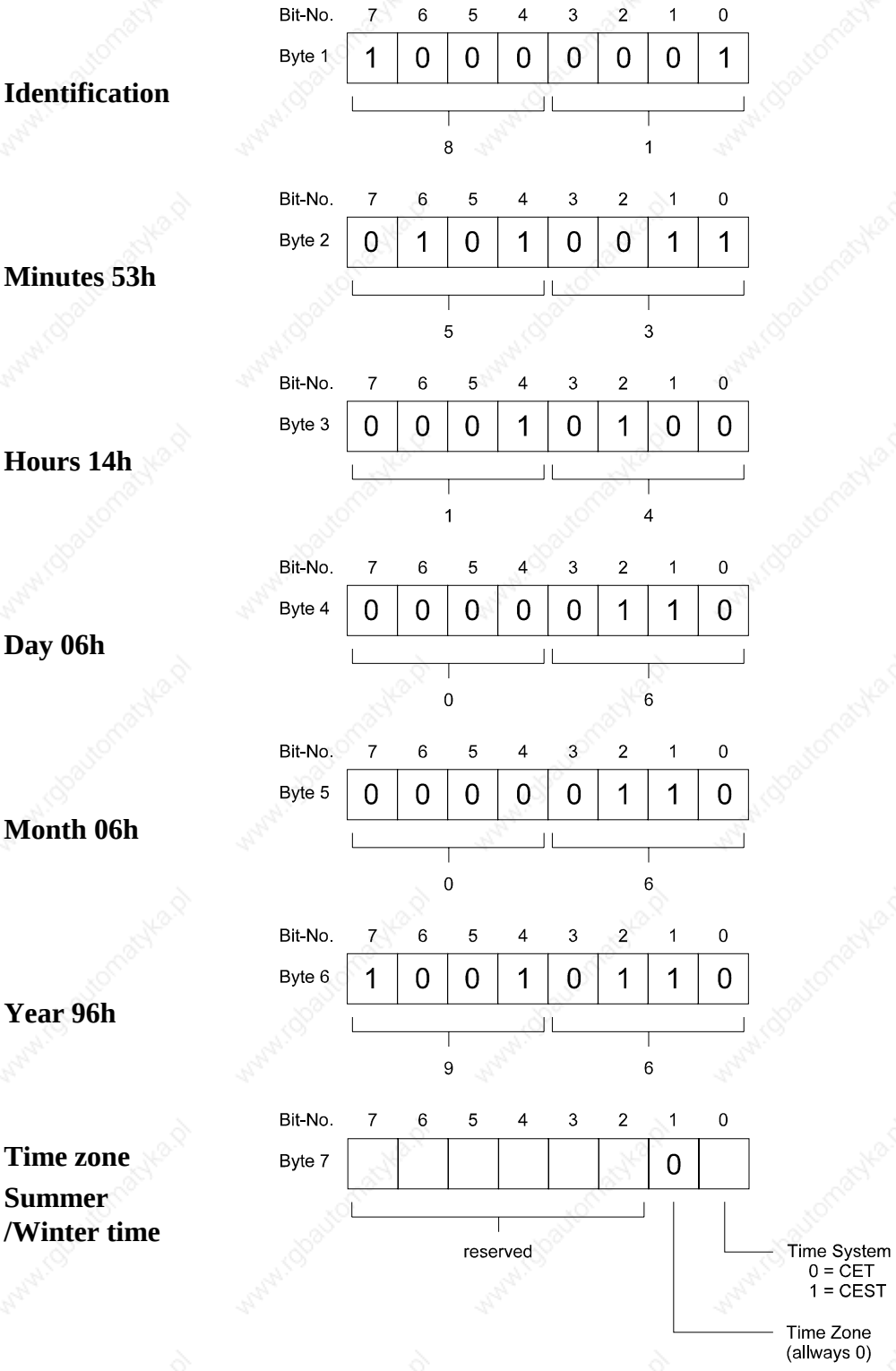
Fig. 3-59: Plan of the coded time information

The carrier frequency is 77,5 kHz. The coding of the time information is carried out in a BCD code by reducing the carrier frequency by 25% at the start of a second for 100 ms as binary 0 or for 200 ms as binary 1.

3.2.10.6 Transmission protocol

The antenna module sends 7 bytes through the 20mA Current Loop interface. The bytes are not acknowledged.

The following is, for example, transferred: 06.06.95 14:53:00



3.2.10.7 Alignment of the antenna

As shown in Fig. 3-60 the antenna must be aligned, so that it is **tangential** to the transmitter.

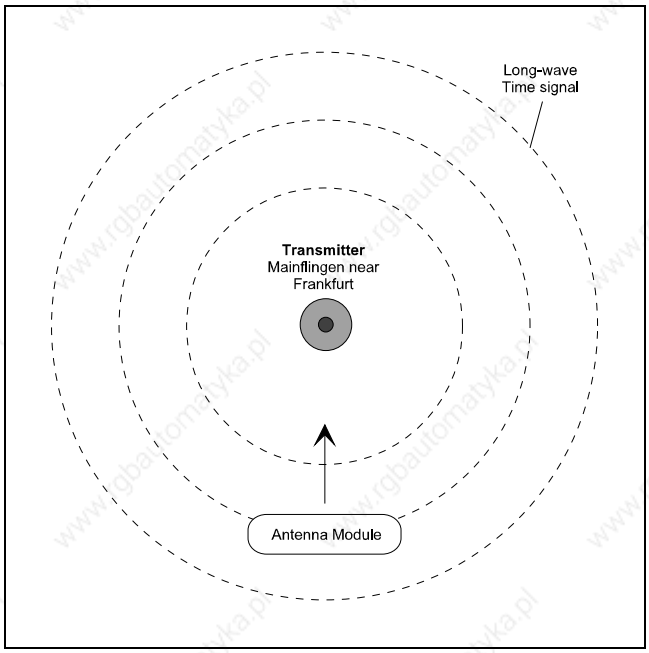


Fig. 3-60: Alignment of the DCF77 antenna module

3.2.10.8 Connection example

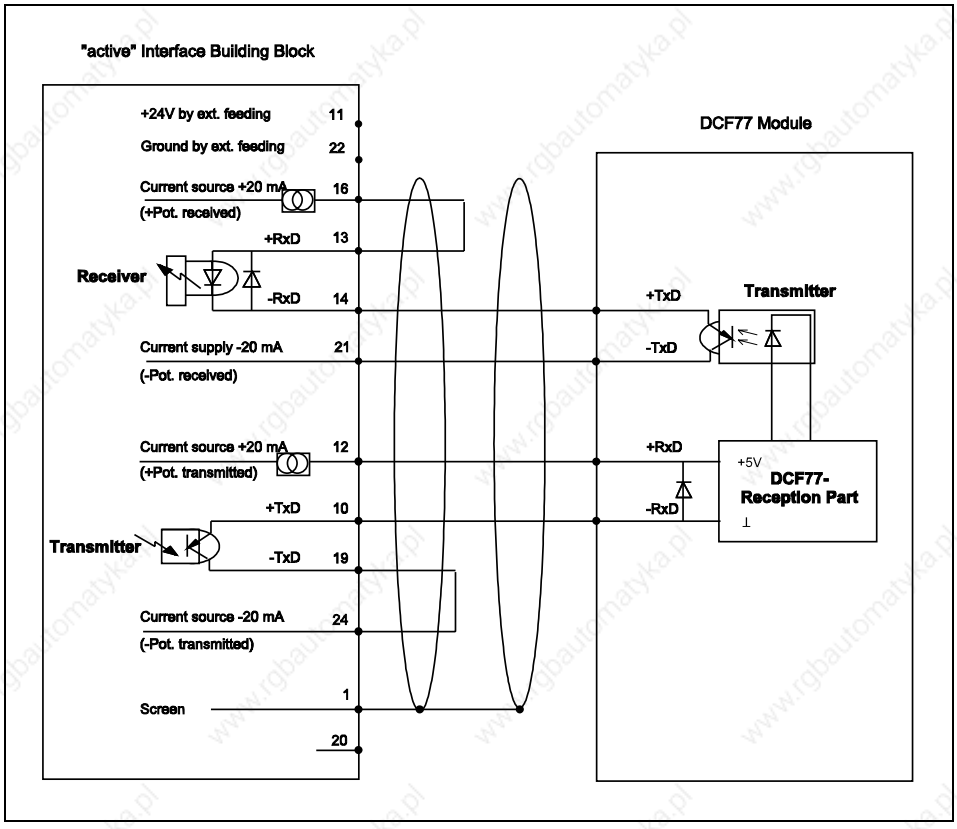


Fig. 3-61: Connection standards for the DCF77 antenna module

3.2.10.9 Start-up response

After being switched on or reset, both LEDs light up on the antenna module after a short transient period. By correct alignment the green LED starts to blink a pulse a second. After approximately one minute, the red light also starts to blink and hereby indicates the synchronisation phase.

Error-free recieved time telegrams are always transmitted through the 20mA interface. The plausibility check of the date and time is carried out in the antenna module.

If an error-free time signal has been received and transmitted to the interface building block, both LEDs go out. The antenna module goes into a current economy mode through the sleep mode.

From now on the clock chip on the interface building block is synchronised every full hour with the actual time.

The absolute difference in propogation time between the DCF77 time and the internal time is at least 60 ms at the point of synchronisation with optimal reception conditions. This difference is larger by poorer reception conditions.

3.2.10.10 Resetting of the antenna module

Antenna modules are reset by removing the 20 mA plug. In the current economy mode the modules can not be directly reset. The voltage is held by a capacitor. A reset procedure can only be activated after the completion of the discharge time (aprrox. 1 minute). The following procedure is to be observed:

- Remove the antenna module's 20mA plug from the interface building block.
- Keep to the 1 minute waiting period.
- Reconnect the antenna module's 20mA plug.

3.2.10.11 Description of the light emitting diodes

Two light emitting diodes are to be found directly on the antenna module, which provide the following information necessary for installation:

LED 1 (green):		Indicates the modulation on the DCF77 carrier signal (by good reception blinks a pulse a second)
LED 2 (red):	ON	After switching on or resetting the building block
	BLINKS	Building block finds itself in the synchronisation phase
	OFF	The time telegram was received in its entirety and error-free and the time was transmitted once to the connected building block.

Piece of advice:

The module does not possess a field strength indicator. The quality of the reception can be determined by the blinking rhythm of the green LED.

Flickering means a bad reception; the antenna must be aligned better.

3.2.10.12 Installation of an interface building block

A 20mA module (SSM-MD25 or SSM-MD26) is required for operating an interface building block (from firmware 4118V12 on). The module must be separately ordered.

From firmware 4118V12 and by selecting the DCF77 function, the following transmission parameters are automatically set, as long as jumper J11 is plugged into position “75 to 19200“ (FB5 is not required).

4800 baud, 8 bit/character, 1 stop bit, even parity

The operation of an interface module will be described below step by step:

- Connect the 20mA module to the interface building block
- Jumper J11 on the interface building block in position 75...19200Baud
- Plug the 24V supply for the front socket onto the building block. The adjustment of the jumper depends on the type of PLC used.
- Connect the antenna module with the interface building block and switch the PLC on. Both LEDs on the interface building block light up and consequently indicate the voltage supply.

  After a short build-up time, both LEDs light up on the antenna module.
- Adjust code 7 with PROCW (FB100) under PROC.
- Align the antenna module, as shown in Fig. 3-60.
- With the clock (FB 102), the interface building block’s internal clock can be read.

3.2.10.13 Application of the DCF77 antenna module with data handling blocks

Possible data handling blocks

Initialization	FB100, FB101
Cycle	FB102

The DCF77 antenna module consists of a DCF77 reception unit, which provides a 20mA current loop interface with the received decode time signal. One needs a 20mA current loop module for the transmission of the time signal. The function blocks FB100 or FB101 and FB5 are required for the application of an antenna module. From the firmware 4118V12 the transmission parameters are automatically set when the DCF77 function is selected, as long as the jumper J11 is plugged into position 75 to 19200 (FB5 not required). The synchronisation of the clock situated on the building block is carried out by the firmware every full hour. See chapter 4.1.3.2 for details on the operation of the clock on the interface building block.

The parameters for the data handling blocks are explained in the following chapters:

FB100 see chapter 4.1.4.1

FB5 see chapter 4.1.3.1

Example:

The following example shows how an interface building block is to be parametrized, so that one can read the time signal of a DCF77 antenna module with a 20mA current loop module.

FB100 (PROCW)

Channel 1 of the building block should be set for the operation of the antenna module. The building block's base address is F0A0. Any error which may arise should be filed in the flag byte MB127.

```
        SPA FB100

NAME    :PROCW
ADR     :KH F0A0      Building block's absolute base address
KANR    :KF+1         Channel 1
PROC    :KF+7         Code for DCF77 antenna module
FEHL    :MB127        Error code in flag byte 127
```

FB5 (PARAMETR)

In this example the building block's DIL switches are to be set to the absolute address F0A0h. The plug connector J11 is in the position 150-38400baud. Any error code which may arise is to be filed in MB127. Channel 1is to be set to the following values:

4800baud, 8bit/character, 1 stop bit, with even parity

```
        SPA FB5
NAME    :PARAMETER
ADR     :KH F0A0      Building block's absolute base address
K/0     :KY1,0        Channel
BAUD    :KF+8         Baud rate = 4800 (according to the table)
ZEI     :KF+8         8 bit/character
PARI    :KY1,1        with parity, even parity
STOP    :KF+1         1 Stop bit
FEHL    :MB127        Error code in flag byte 127
```

3.2.10.14 FB102 (UHR), operation of the internal clock

The clock situated on the interface building block can be set with the FB 102 (UHR) and sent to the automation device. It is however possible to set the time automatically with the help of a DCF77 antenna module, which is available from VIPA.

For peripheral addressing please use a FB 103.

The following parameters are to be transferred with the loading of FB 102:

Des.	Format	Explanation
ADR	KH	Building blocks absolute base address
DB	B	Details of data block
FRG	BI	Release bit
NEUS	M	Block the building block

Tab. 3-17: List of parameters for the loading of FB102

ADR (absolut) Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2.

In the case of peripheral addressing enter the address, which you have set with the DIL switch S1 on the building block, into FB103. The following addresses can be loaded: BG41: PY128-PY252, BG42 and BG43: PY128-PY248.

DB Entry of the data blocks, in which the data record for setting the clock or the time and date are filed (in delivery state DB 99). The following functions are carried out in this data block:

Transfer time and date to PLC (BCD depiction)

Transfer time and date to PLC (ASCII character)

FRG If this bit is placed at 1, the date and time are read once or they are sent in order to set the clock. After transmission the bit is returned to 0. If this bit is placed at 0 when loading FB 102, the building block will again leave immediately!

NEUS If the data handling block runs through with a set new start bit, the module will be blocked for all SPS accesses. By loading the FB28 with a set new start bit, the building block will be released again.

Allocation DB99

The data words of the data block DB99 are allocated as follows:

DW0	DL0:	40h	=	building block's aknowledgement byte
	DR0:	Error byte		
		00h	=	no error
		01h	=	building block wrongly addressed or defect
		04h	=	data block not available
		0Ah	=	wrong clock command
DW1	DL1:	40h	=	clock should be addressed
	DR1:	Sub-command for operating clock		
		01h	=	Transfer time and date to PLC (BCD depiction)
		02h	=	irrelevant if using the DCF77
		03h	=	irrelevant if using the DCF77
		04h	=	irrelevant if using the DCF77
		05h	=	Transfer time and date to PLC (ASCII character)

DW1 is to be previously occupied by the user prior to the loading of FBs.

DW2	DL2	00h…59h	seconds (BCD)
	DR2	00h…59h	minutes (BCD)
DW3	DL3	00h…23h	hours (BCD)
	DR3	01h…31h	day (BCD)
DW4	DL4	01h…12h	month (BCD)
	DR4	00h…99h	year (BCD)
DW5	DL5	00h…06h	weekday, Sunday = 00h (BCD)
	DR5	00h	

The allocation of the data words from DW2 up to DW5 is only of significance for the sub-command 01h (= transfer time and date in BCD format to PLC).

For the sub-command 05h (= transfer time and date in ASCII format to the PLC) the string to be transferred is filed in the data words DW2…DW11.

The following character string is, for example, transmitted: Fr 02.06.89 22:55:58

Allocation of the Data Blocks

DW2	DW3	DW4	DW5	DW6	DW7	DW8	DW9	DW10	DW11
FR	07.	06	.96		22	:55		:58	

Example FB102 (UHR)

Function "set clock": the clock is to be set to:

Tuesday 06.06.89 14:53:00

DB 99 is to be allocated the following values:

	DL	DR		
DW0	40h	00h	Acknow. byte	Error byte
DW1	40h	02h	Address clock	Set clock
DW2	00h	53h	Seconds	Minutes
DW3	14h	06h	Hours	Day
DW4	06h	89h	Month	Year
DW5	02h	00h	Weekday	

Tab. 3-18: Example FB102 (UHR)

OB Cycle

```
BAUSTEIN#OB1
      BIB     #14026
00002       :
00004       :UN  M 0.0
00006       :S   M 0.0                    Time in every cycle
0000C       :
0000E       :SPA FB 102
      NAME #UHR
      ADR  =KH F080                       Building block's base address:
                                            F080h
      DB   =DB 99                         Clock DB-Nr:99
      FRG  =M 0.0                         Enable bit
      NEUS =M 0.1                         Restart flag, which is to be
0001A       :                             placed in OB22
0001E       :U   M 0.1                    Erase restart flag
00020       :R   M 0.1
00022       :BE
```

4 Software

4 Software

The interface building blocks can be parameterized in two ways:

1. Using the VIPA handling blocks which can be ordered by VIPA.
2. Parameterizing the interface building blocks directy over channel 0.

The needed function blocks for the wanted function can be seen in Tab. 4-1.

Functions	**Required or possible module**	**Initialization (OB20, OB21, OB22)**	**Cycle (OB1)**
Serial functions			
No procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB100, FB101, FB5, FB15	FB3, FB4
STX/ETX procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB100, FB101, FB5, FB15, FB40	FB43, FB44
3964(R) procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB30, FB37	FB31-33
3964(R) with RK 512 procedure	MD21, MD22, MD25, MD26, MD33, MD34	FB20, FB27	FB21-26
[1]Mettler Balance	MD22, MD25, MD26		FB17, FB18, FB7, FB8
Hohner Encoder	MD34	FB5	FB19
Parallel functions			
	MD24	FB100, FB101, FB5	FB3
Additional functions			
SSI module	MD39	FB100, FB101	FB45
Counter	MD18, MD19	FB100, FB101, FB12	FB10, FB11
Analogue input	MD40, MD41, MD42, MD43,	FB50, FB51, FB54	FB52, FB55, FB56, FB57, FB58
Analogue output	MD45, MD46, MD47, MD48, MD49	FB50, FB51, FB54	FB53, FB56
Measurement of temperature (PT100)	MD44	FB50, FB51, FB54	FB52, FB55, FB56, FB57, FB58
Clock			FB102
DCF77 antenna	MD36	FB100, FB101	FB102

Tab. 4-1: Selection table: function - module - function blocks

[1] The use of the Mettler Balance is described in manual HB30.

4.1 Handling with VIPA data handling blocks

VIPA delivers functional blocks for changing standard parameters, for functional and channel selection and for operating the internal clock as well as the necessary functional blocks for individual modules.

4.1.1 Important advice

With some data handling blocks it is possible to address the building blocks peripherally. This method of addressing is for example necessary, when one wishes to use the building blocks in an expansion unit.

Absolute addressing offers you the possibility to use the entire address space of an automation device as you wish.

In the following, the functional blocks necessary for absolute addressing will be described and the blocks with the same function for peripheral addressing will be listed.

Further information on the methods of addressing can be found in chapter 5.

4.1.2 Overview of parameterizing possibilities

- **Parameterizing with VIPA data handling blocks**

Changing of the standard parameter

FB5, PARAMETR, Baud rate, length of signals, parity, number of stop bits with absolute addressing.

FB15 PARAMPER, Baud rate, length of signal, parity, number of stop bits with peripheral addressing.

FB102 UHR, Handling of the clock on the module with absolute addressing.

FB103 UHRP, Handling of the clock on the module with peripheral addressing.

Function Selection

FB100, PROCW, Protocols, procedures for a channel with absolute addressing.

FB101 PROCWP, Protocols, procedures for a channel with peripheral addressing.

Data Handling without a procedure

FB3, SEND, Data output without a procedure but with absolute addressing.

FB13 SEND, Data output without a procedure but with peripheral addressing.

FB4, RECEIVE, Data received without a procedure but with absolute addressing.

FB14 RECEIVE, Data received without a procedure but with peripheral addressing.

Data Handling Blocks dependant on procedures

STX/ETX procedure

FB40 PARA_STX, Parametrizing the STX/ETX interface
FB43 SEND, Data output
FB44 RECEIVE, Data received

3964(R)-procedure

FB30 SYNCHRON (up to Firmware level 4.1), Select 3964R procedure
FB37 SYNCHRON (from Firmware level 5.0 on), Select 3964R procedure
FB28 RELEASE BUILDING BLOCK, the building block blocked out with SYNCHRON, is released again.
FB33 SEND, Data output
FB31 RECEIVE, Data received

3964(R) with RK512 procedure

FB20 SYNCHRON (up to Firmware level 4.1), Select 3964R with RK512 procedure
FB27 SYNCHRON (from Firmware level 5.0 on) Select 3964R with RK512 procedure
FB28 RELEASE BUILDING BLOCK, the building block blocked out with SYNCHRON, is released again.
FB23 SEND, Data output
FB21 RECEIVE, Data received
FB25 FETCH, Data requested
FB24 RECEIVE-ALL, Data passively received
FB22 SEND-ALL, Requested data sent

Data Handling Blocks dependant on the Module

CENTRONICS module

FB3 SEND, Data output without a procedure but with absolute addressing.
FB13 SEND, Data output without a procedure but with peripheral addressing.

Counter module

FB12 MODE, Mode selection for the counter
FB10 READ, Read the counter-reading
FB11 WRITE, Preset counter with a numerical value

Analogue input/analogue output module

FB50 PROCW, Adjust channel and module number
FB51 MODE, Mode selection for input and outputs
FB52 ANA_IN, Read the analogue value
FB53 ANA_OUT, Send the analogue value
FB54 UG/OG, Predefine the highest and lowest levels for scaling
FB55 STAT_IN, Send out the status bit
FB56 MIN/MAX, Set the min./max. values for input and outputs
FB57 IN_MW, File the analogue values in the flag word
FB58 IN_DW, File the analogue values in the data word

SSI module

FB45 SSI_IN, Read SSI data

- **Parametrizing with user software**

  The standard parameter can be changed directly over the module's channel 0.
  See chapter 4.2. for further details.

4.1.3 Changing the standard parameters

The interface module is automatically parametrized after the PLC has been switched on for the first time. The following standard values are set:

- Standard function (without a procedure/protocol)
- Baud rate of 9600 baud
- Even parity
- Start bit
- 8 Data bits
- 2 Stop bits
- All buffers are reset
- Error-LED is switched off

The FB5 or the FB15 are required for changing the standard values. The FB102 or the FB103 are required, if one wishes to set the internal clock..

4.1.3.1 FB5 (PARAMETR), parametrizing

With the FB 5 (PARAMETR) it is possible to alter the standard parameter.

It is advisable to load the FB 5 (PARAMETR) when restarting (OB 20 ... 22). FB 5 should be reloaded once for each channel.

Please use FB 15 for peripheral addressing.

The FB is not required, if standard parametrizing should be used.

The following parameters are to be transferred when loading FB 5:

Des.	Format	Explanation
ADR	KH	Building block's absol. base address
K/0	KY	Number of channels
BAUD	KF	Baud rate code
ZEI	KF	Bit / Character
PARI	KY	Parity code
STOP	KF	Number of stop bits
FEHL	BY	Error byte

Tab. 4-2: List of parameters for the loading of FB 5

The following details are necessary for the abovementioned parameters:

ADR (absolute)

Input of the physical start address, under which the module can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2.

In the case of peripheral addressing, enter the address, which you have set with the S1 switch on the building block, into the FB 15. You can load the following addresses: BG41: PY128-PY252, BG42 and BG43: PY128-PY248

K/0 K= Number of channels, through which the data should be transmitted.

The second byte must be 0. The channel number used is dependent upon the interface:

Building block BG41	channel number	1
Building block BG42	channel number	1, 2
Building block BG43	channel number	1, 2, 3

BAUD Different baud rates can be set by entering the respective codes. Optional baud rates from 150 to 38400 baud can be set with the plug connector J11 on the building block. To enable this, J11, which is inserted in its delivery state at a baud rate of 75 to 19200 baud, must be reinserted.

Code (decimal)	Plug Connector J11 in Position 38400 / 19200 J11	 38400 / 19200 J11
01	75	150
02	100	(200)
03	110	(220)
04	150	300
05	300	600
06	600	1200
07	1200	2400
08	2400	4800
09	4800	9600
10	9600	19200
11	19200	38400

ZEI Character lengths for transmission. The length of characters can be from 5 to 8 bits.

PARI Parity Setting:

left byte:	1	=	even parity
	0	=	uneven parity
right byte	1	=	with parity
	0	=	without parity

STOP Setting the number of stop bits

Input		Stop bits
1	=	1
2	=	1 1/2
3	=	2

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction, an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0, once the cause of the error has been eliminated.

An overview of possible error codes can be found in chapter 4.1.5.3.

Example: FB5 (PARAMETR)

Channel 1 of the building block should be set as follows:

- 1200 baud
- 7 bit/character
- 2 stop bit
- without parity

The building block's DIL switch is set to the absolute address F0A0H. The plug connector J 11 is positioned at 75-19200 baud. Any error codes, which may possibly arise, should be filed in MB127.

```
         SPA FB5

NAME     :PARAMETR
ADR      :KH F0A0      Building block's absol. base address
K/0      :KY1,0        Channel
BAUD     :KF+7         Baud rate = 1200 (according to the table)
ZEI      :KF+7         7 bit/character
PARI     :KY0,0        Without parity
STOP     :KF+3         2 Stop bits
FEHL     :MB127        Error code in flag byte 127
```

If the plug connector is plugged into the position 150-38400 baud, the same transmission format can be achieved by using the following parametrizing:

```
         SPA FB5

NAME     :PARAMETR
ADR      :KH F0A0      Building blocks's absol. base address
K/0      :KY1,0        Channel 1
BAUD     :KF+6         Baud rate = 1200 (according to the table)
ZEI      :KF+7         7 bit/character
PARI     :KY0,0        Without parity
STOP     :KF+3         2 stop bits
FEHL     :MB127        Error code in flag byte 127
```

4.1.3.2 FB102 (UHR), Operation of the Internal Clock

The clock situated on the interface building block can be set with the FB 102 (UHR) and sent to the automation device. It is however possible to set the time automatically with the help of a DCF77 antenna module, which is available from VIPA.

For peripheral addressing please use a FB 103.

The following parameters are to be transferred with the loading of FB 102:

Des.	Format	Explanation
ADR	KH	Building blocks absolute base address
DB	B	Details of data block
FRG	BI	Release bit
NEUS	M	Block the building block

Tab. 4-3: List of Parameters for the Loading of FB102

ADR (absolut) Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2.

In the case of peripheral addressing enter the address, which you have set with the DIL switch S1 on the building block, into FB103. The following addresses can be loaded: BG41: PY128-PY252, BG42 and BG43: PY128-PY248.

DB Entry of the data blocks, in which the data record for setting the clock or the time and date are filed (in delivery state DB 99). The following functions are carried out in this data block:

Transfer time and date to PLC (BCD depiction)

Set the clock *

Set the clock one hour forward *
(adjusts from winter to summer time)

Set the clock one hour back *
(adjusts from summer to winter time)

Transfer time and date to PLC (ASCII character)

* These functions are not available,
if the DCF77 antenna module is being operated.

FRG If this bit is placed at 1, the date and time are read once or they are sent in order to set the clock. After transmission the bit is returned to 0. If this bit is placed at 0 when loading FB 102, the building block will again leave immediately!

NEUS If the data handling block runs through with a set new start bit, the module will be blocked for all SPS accesses. By loading the FB28 with a set new start bit, the building block will be released again.

Allocation DB99

The data words of the data block DB99 are allocated as follows:

DW0

DL0: 40h = building block's aknowledgement byte

DR0: Error byte

- 00h = no error
- 01h = building block wrongly addressed or defect
- 04h = data block not available
- 0Ah = wrong clock command

DW1

DL1: 40h = clock should be addressed

DR1: Sub-command for operating clock

- 01h = Transfer time and date to PLC (BCD depiction)
- 02h * = Set clock (Sub-command in DW2 up to DW5)
- 03h * = Set clock one hour forward (adjusts from winter to summer time)
- 04h * = Set clock one hour back (adjusts from summer to winter time)
- 05h = Transfer time and date to PLC (ASCII character)

* These functions are not available, if the DCF77 antenna module is being operated.

DW1 is to be previously occupied by the user prior to the loading of FBs.

DW2

DL2	00h…59h	seconds (BCD)
DR2	00h…59h	minutes (BCD)

DW3

DL3	00h…23h	hours (BCD)
DR3	01h…31h	day (BCD)

DW4

DL4	01h…12h	month (BCD)
DR4	00h…99h	year (BCD)

DW5

DL5	00h…06h	weekday, Sunday = 00h (BCD)
DR5	00h	

The allocation of the data words from DW2 up to DW5 is only of significance for the sub-command 01h (= transfer time and date in BCD format to PLC) and 02h (= set clock).

The data words are irrelevant for the sub-commands 03h (= set clock one hour forward) and 04h (= set clock one hour back).

For the sub-command 05h (= transfer time and date in ASCII format to the PLC) the string to be transferred is filed in the data words DW2…DW11.

The following character string is, for example, transmitted: Fr 02.06.89 22:55:58

Allocation of the Data Blocks

DW2	DW3	DW4	DW5	DW6	DW7	DW8	DW9	DW10	DW11
FR	0	7.	06	.9	6	22	:5	5:	58

Example FB102 (UHR)

Function "set clock": the clock is to be set to:

Tuesday 06.06.89 14:53:00

DB 99 is to be allocated the following values:

	DL	DR		
DW0	40h	00h	Acknow. byte	Error byte
DW1	40h	02h	Address clock	Set clock
DW2	00h	53h	Seconds	Minutes
DW3	14h	06h	Hours	Day
DW4	06h	89h	Month	Year
DW5	02h	00h	Weekday	

Tab. 4-4: Example FB102 (UHR)

OB Cycle

```
BAUSTEIN#OB1
      BIB     #14026
00002       :
00004       :UN  M 0.0
00006       :S   M 0.0                    Time in every cycle
0000C       :
0000E       :SPA FB 102
      NAME #UHR
      ADR  =KH F080                       Building block's base address:
                                            F080h
      DB   =DB 99                         Clock DB-Nr:99
      FRG  =M 0.0                         Enable bit
      NEUS =M 0.1                         Restart flag, which is to be
0001A       :                             placed in OB22
0001E       :U   M 0.1                    Erase restart flag
00020       :R   M 0.1
00022       :BE
```

4.1.4 Selection of function

4.1.4.1 FB100 (PROCW), protocol/procedure selection

The function of the interface is selected with the FB 100 (PROCW). The channel number and protocols and procedures to be used can be determined here.

For peripheral addressing please use FB101

The following parameters are to be transferred when loading FB 100:

Des	Format	Explanation
ADR	KH	Building block's absolute base address
KANR	KF	Number of channels
PROC	KF	Protocol, procedure
FEHL	BY	Error byte

Tab. 4-5: List of parameters for the loading of FB100

The following details are necessary for the abovementioned parameters:

ADR (absolut)

Input of the physical start address, under which the module can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2.

In the case of peripheral addressing enter the address, which can be set with the DIL switch S1 onto the building block, into FB101. The following addresses can be loaded: BG41: PY128-PY252, BG42 and BG43: PY128-PY248

KANR

KANR= Number of channels, through which the data is to be transmitted. The channel number used is dependent upon the interface:

Building block BG41	Channel number 1
Building block BG42	Channel number 1, 2
Building block BG43	Channel number 1, 2, 3

PROC

The function selection occurs by entering the respective codes. The following code should be given:

Code 00:	without a protocol/procedure
Code 01:	STX/ETX - protocol
Code 02:	Meldefunktion (nur für Meldebaugruppen)
Code 05:	Counter (only possible with counter modules)
Code 06:	Clock pulse generator (only possible with clock pulse generator modules)
Code 07:	DCF77 (only possible with DCF77 antenna modules)
Code 12:	SSI Module (only possible with SSI modules)

FEHL

This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.

An overview of possible error codes are to be found in chapter 4.1.5.3.

Examples: FB100 (PROCW)

It should be operated without a protocol or a procedure through channel 1 of the building block. The building block's base address is F0A0. Any error which may possibly arise, should be filed in the flag byte MB 127.

```
          SPA FB100

NAME      :PROCW
ADR       :KH F0A0      Building block's absolute base address
KANR      :KF+1         Channel 1
PROC      :KF+0         Code for without protocol/procedure
FEHL      :MB127        Error code in flag byte 127
```

Procedure STX/ETX should be used through channel 3 of the building block. The building block's base address is F0A0. Any errors which may possibly occur, should be filed in the flag byte MB 107.

```
          SPA FB100

NAME      :PROCW
ADR       :KH F0A0      Building block's absolute base address
KANR      :KF+3         Channel 3
PROC      :KF+1         Code for procedure STX/ETX
FEHL      :MB127        Error code in flag byte 127
```

4.1.5 Data handling without a protocol and procedure

It is possible to transmit data to a peripheral piece of equipment without a procedure and protocols. One only has to define the respective channel with FB 100 (Code 00) as the general input/output interface. This is in accordance with the fundamental philosophy of interface building blocks.

4.1.5.1 FB3 (SEND), data output without a protocol and procedure

This function block is used for transmitting data to a peripheral piece of equipment without a protocol and procedure. For peripheral addressing please use FB13.

When loading FB3, the following parameters are to be transferred:

Des.	Format	Explanation
ADR	KH	Building block's absolute base address
K/DB	KY	Number of channels/Number of data blocks
ANF	W	Start of transmssion buffer
ANZ	BY	Number of bytes
FRG	BI	Release strobe bit
FEHL	BY	Error byte
NEUS	M	New start flag

Tab. 4-6: List of parameters for the loading of FB3

The following details are necessary for the abovementioned parameters:

ADR Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2.

In the case of peripheral addressing enter the address, which you have set with the DIL switch S1 on the building block, into FB13. The following addresses can be loaded: BG41: PY128-PY252, BG42 and BG43: PY128-PY248

K/DB K= Number of channels, through which the data is to be sent. This channel must be defined with the FB 100 as the general input/output interface (without a procedure/protocol).

DB= Number of DBs, which are contained in the data to be sent.

ANF Word variable, which contains the number of data words from the point where the characters to be sent are filed.

ANZ Byte variable, which contains the number of bytes, which are to be transmitted.

FRG If this bit is set at 1, the amount of data given in ANZ is transferred once. After transmission the bit is set back to 0. If the bit is set at 0 when loading FB3, the building block is exited immediately.

FEHL All bits of these bytes are set to 0, if functioning correctly. In the case of a malfunction, an error code is entered. The error is acknowledged automatically, i.e. the byte is reset to 0 once the cause of the error has been eliminated.

An overview of the possible error codes are to be found in chapter 4.1.5.3.

NEUS As long as the bit is at signal state 1, the handling of the function block will be discontinued.

Example FB3 (SEND)

31 bytes should be transmitted through channel 1, which are filed from data word DW5 in DB 23. The DIL switches are set to the absolute address F1A0h and the M4.7 is used as the enable bit. Any error codes, which may possibly arise, should be filed in MB200.

```
        :
        :
        :L  KB31
        :T  MB40        The flag byte contains the number
        :               of bytes to be transmitted
        :L  KB5
        :T  MW58        The flag word contains the number
        :               of data words, from where the data
        :               is to be filed
        :
        :S  M 4.7       Set release flag
        :SPA FB3
NAME    :SEND
ADR     :KHF1A0         Building block's absolute base address
K/DB    :KY1,23
ANF     :MW58
ANZ     :MB40
FRG     :M 4.7
FEHL    :MB200          Error code in flag byte 200
NEUS    :M 1.0          New start flag
```

4.1.5.2 FB4 (RECEIVE), data reception without a protocol and procedure

This data block is used for receiving data without a protocol and procedure from a peripheral piece of equipment. For peripheral addressing please use FB 14.

The following parameters are to be transmitted when loading FB4:

Des.	Format	Explanation
ADR	KH	Building block's absolute base address
K/DB	KY	Number of channels/data block number
SZ	BY	Write pointer
LZ	BY	Read pointer
ANF	W	Start of receiving buffer
ANZ	BI	Number of bytes
FEHL	BY	Error byte
NEUS	M	New start flag

Tab. 4-7: List of parameters for the loading of FB4

The following details are necessary for the abovementioned parameters:

ADR Input the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2.

In the case of peripheral addressing enter the address, which you have set with the DIL switch S1 on the building block, into FB14. The following addresses can be loaded: BG41: PY128-PY252, BG42 and BG43: PY128-PY248

K/DB K= Number of channels, through which the data should be received. This channel must be defined with the FB 100 as a general input/output interface (without a procedure/protocol).

DB= Number of DBs, in which the data to be received is to be filed.

SZ Write pointer: is raised by the handling block into the reception DB every time a byte is registered. The write pointer represents an index for the reception buffer.

LZ Read pointer: can be processed by user, if the data was processed from the reception DB. The number of bytes received can be calculated from the difference between the write and read pointer. Write and read pointers can be evaluated differently depending on the requirements (see example) and must be stated by the user. If the read pointer is not updated (or the write pointer not reset) an error message 5 will result (buffer overflow).

ANF Word variable, which contains the number of the first data words in the reception DB, from the point when the received data has been filed. The data is filed in the order of date left/date right in the receive buffer.

ANZ Number of the bytes, reserved for the buffer, in the reception DB. The buffer is organised as a loop buffer, i.e. if the buffer is completely full, any further bytes will be recorded once again at the start of the buffer.

FEHL The bits of these bytes are set at 0, if functioning correctly. In the case of a malfunction, an errror code will be entered. The error is acknowledged immediately, i.e. the byte will be reset at 0 once the cause of the error has been eliminated. Error 5 (buffer overflow) is to be acknowledged by the user.

An overview of possible error codes can be found in chapter 4.1.5.3.

NEUS As long as the bit is at the signal state 1, the handling of the function block is discontinued.

Management of write and read pointers

The handling of write and read pointers can be clarified by using two examples. The following is required for this: the receive buffer begins from the data word DW 5 (parameter ANF) and the size of the receive buffer (loop buffer) is 12 bytes (parameter ANZ).

Before data is received the write pointer (SZ) and the read pointer (LZ) point to DL5, the difference is therefore 0. If, for example, 5 bytes are received from a peripheral device, they are recorded alternately in data word left (DL) and data word right (DR).

The write pointer is increased by 1 for every byte received. After receiving 5 bytes, it therefore has a value of 5 and points to the next free space in reception DB. Both evaluation possiblities are shown in the following examples:

Example 1: Write and read pointer

The read pointer is placed at the position of the write pointer, if the data received is processed by the user program. If data is newly received after this, the write pointer is again increased by one for every byte (calculated from the new starting point) and the read pointer is updated accordingly.

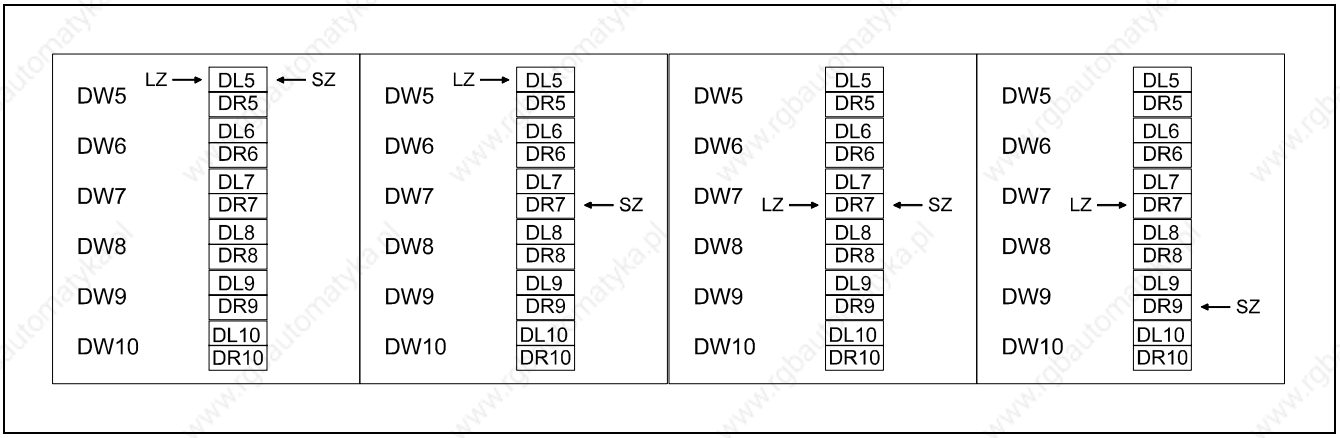


Fig. 4-1: Evaluation possibility 1 with the help of write and read pointers

Example 2: Write and read pointers

The second evaluation possibility is to always leave the read pointer at 0 and to reset the write pointer to 0, if the received data is processed by the user program. The read pointer always points directly to the number of bytes received.

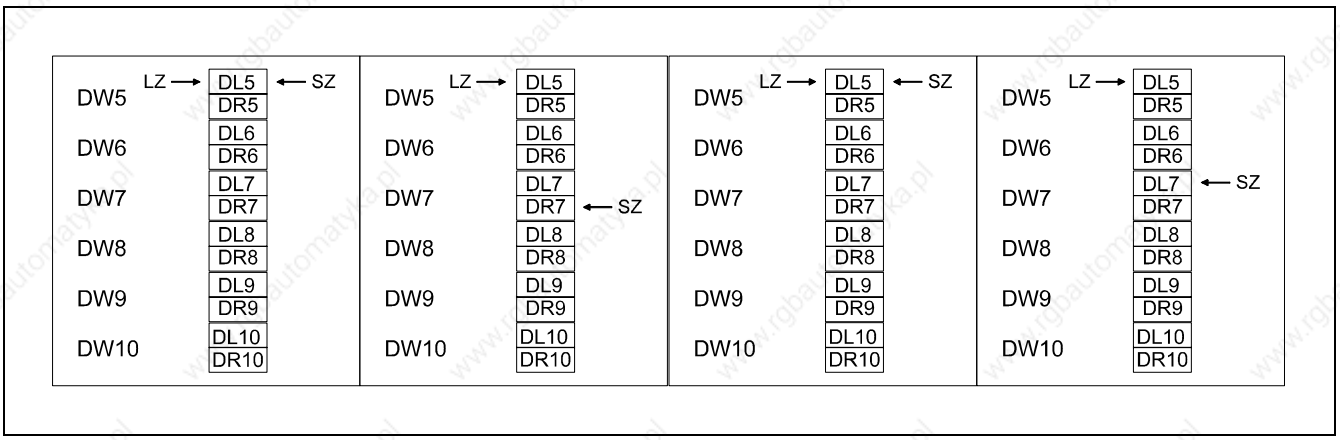


Fig. 4-2: Evaluation possibility 2 with the help of write and read pointers

Example FB4 (RECEIVE)

Data should be received on channel 1 and filed in DB 100. The receive buffer begins in DB 100 at DW5 and contains 40 bytes. The DIL switches are set to the absolute address F180h and any error which may arise is to be filed in MB20.

```
        :
        :
        :L  KB40
        :T  MB14        The flag byte contains the size
        :               of the receive buffer
        :L  KB5
        :T  MW12        The flag word conains the number
        :               of data words, from the point where the data
        :               is to be filed
        :
        :
        :SPA FB 4
NAME    :RECEIVE
ADR     :KH F180        Absolute base address
K/DB    :KY1,100        Channel 1, data block 100
SZ      :MB10
LZ      :MB11
ANF     :MW12
ANZ     :MB14
FEHL    :MB20           Error code in flag byte 20
NEUS    :M 1.0          New start flag
```

4.1.5.3 Error code, which FEHL can comprise of

Error No.	Error
1	Building block not plugged in, wrongly addressed or defect.
2	Channel no. <1 or >3
3	DB0 used (should not be used)
4	Data block given does not exist
5	Buffer overflow reception
6	Parameter error in indentifier STOP
7	Parameter error in identifier PARI
8	Parameter error in identifier ZEI
9	Parameter error in identifier BAUD

Tab. 4-8: Error code, which FEHL can comprise of

4.1.6 Data handling with procedure STX/ETX

The interfaces of the building blocks BG41, BG42 and BG43 can be operated with the procedure STX/ETX, if this function is selected in FB100 or FB101. The STX/ETX procedure is used for the transmission of ASCII characters (20h…7Fh). It operates without a block check (BCC). If data is to be read in from the periphery, the start character STX (02h) must be available. The characters to be transferred follow. The end character must be ETX (03h). The useful data, i.e. all characters between STX and ETX, is transferred to the automation device after the end character ETX has been received.

When sending the data from the PLC to a peripheral device, the useful data is transferred to the building block and from there transferred to the external device with STX as the start character and ETX as the end character.

The data handling blocks FB 43 and FB 44 come together with the PLC.

4.1.6.1 FB40 (PARA_STX), parametrizing of the STX/ETX procedure

With FB40, the function for reparametrizing the STX/ETX procedure is selected.

The following parameters are to be transferred when loading FB40:

Des.	Format	Explanation
ADR	KY	Building block's absolute base address
K/M	KY	Number of channels/ FIFO-Mode
STX	KY	Start character
E1/2	KF	End character 1/end character 2
ZVZ	BY	Character delay time
FEHL	KH	Error byte

Tab. 4-9: List of parameters for the loading of FB40

The following details are necessary for the abovementioned parameters:

ADR (absolut) Input the physical start address, under which the building block can be accessed by the PLC. This start address can be set with DIL swiches S1 and S2 (see chapter 5).

K/M K = Number of channels, whose STXC/ETX protocol should be parametrized.

M = FIFO Mode.

- 0 This mode only saves the actual telegram. An old telegram, which has not yet been collected, will be erased.
- 1 With this mode all telegrams up to the maximum buffer size are saved. Telegrams, which can no longer be written into the buffer, will be chopped.

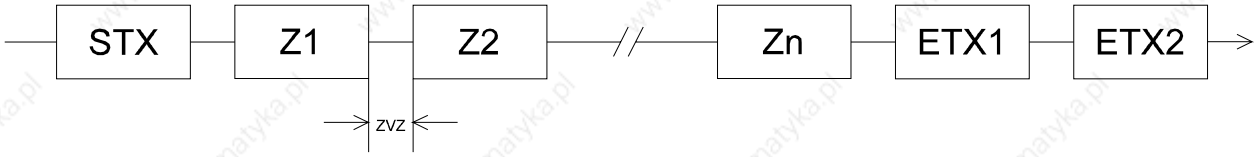
STX ASCII character of the new start identifier. Enter in right byte.

E1/2 The ASCII character of the end identifier
Left Byte: 1. end character
Right Byte: 2. end character
is used as the 2. end character 0h. In this way only one end character is evaluated.

ZVZ Character delay time in 50ms units.

FEHL This byte has a value of 0, if functioning correctly. In the case of a malfunction an error code is entered. The error is acknowledged immediately, i.e. the byte is reset at 0 once the cause of the error has been eliminated.
An overview of the possible error codes can be found in chapter 4.1.5.3.

Telegram structure:



4.1.6.2 FB43 (SEND), data output with STX/ETX procedure

This data handling block replaces the FB3. The list of parameters is identical to those of the FB3. The description of FB3 can be found in chapter 4.1.5.1.

4.1.6.3 FB44 (RECEIVE), data reception with STX/ETX procedure

This data handling block replaces the FB4. The list of parameters is identical to those of the FB4. A description of FB4 can be found in chapter 4.1.5.2.

Example of STX/ETX:

Call the FB100 before the FB40 and select the procedure STX/ETX.

```
BAUSTEIN#OB20/OB21/OB22
00002       :SPA FB 100
      NAME #PROCW
      ADR  =KH F080              Building block's address
      KANR =KF +1                Channel number
      PROC =KF +1                Procedure number(1 = STX/ETX)
      FEHL =MB 199               Error byte
0000E       :
00010       :SPA FB 40
      NAME #PARA-STX
      ADR  =KH F080              Buildling block's address
      K/M  =KY 1,0               Channel number, without FIFO
      STX  =KY 0,2               Start character is 02H
      E1/2 =KY 3,0               1. End character 03H, no 2. end character
      ZVZ  =KF +5                Character delay time is 250ms
      FEHL =MB 198               Error byte
00022       :BE
```

4.1.7 Handling with procedure 3964(R)

4.1.7.1 General

The building blocks BG41, BG42 and BG43 can be operated with the procedure 3964(R). It is not necessary to select the procedure with the FB100 or the FB101, as this is carried out by the FB37 (SYNCHRON).

A 3964 (R) operation is not possible on channel 3 of the BG43!

If the 3964 (R) operation is needed on all three channels, the firmware 4017v5x is to be used.

The procedure 3964 differs from 3964R in only two aspects of the procedure:

- The BCC byte is dropped in the procedure 3964. DLE/ETX applies for the end of the telegram.
- Acknowledgement delay time (standard requirements)

  In procedure 3964: QVZ = 550 ms.

  In procedure 3964R: QVZ = 2000 ms

In order to be able to work with procedure 3964 or 3964R, the following VIPA data handling blocks have been made available:

SYNCHRON FB37 (from firmware level 401xV50 and all 4118)

With this function block the building block is set to operate with the 3964(R) procedure, the baud rate and format of data transmission is selected and the size of the block and priority are set.

Difference to FB30: Input of a character delay time (ZVZ)
Input of an acknowledgement delay time (QVZ)

SYNCHRON FB30 (up to firmware level 401xV41)

This function block is to be used instead of FB37, if the firmware level 4.1 or older are to be used.

RECEIVE FB31

With this function block, telegrams are received from partner equipment.

SEND FB33

This function block is used to send an output telegram to partner equipment.

RELEASE BUILDING BLOCK FB28

This function block resets the new start bit and releases the building block for the SPS-access. This bit is to be given by the individual data handling blocks.

4.1.7.2 Procedure

The following describes the structure of the procedure and the telegrams:

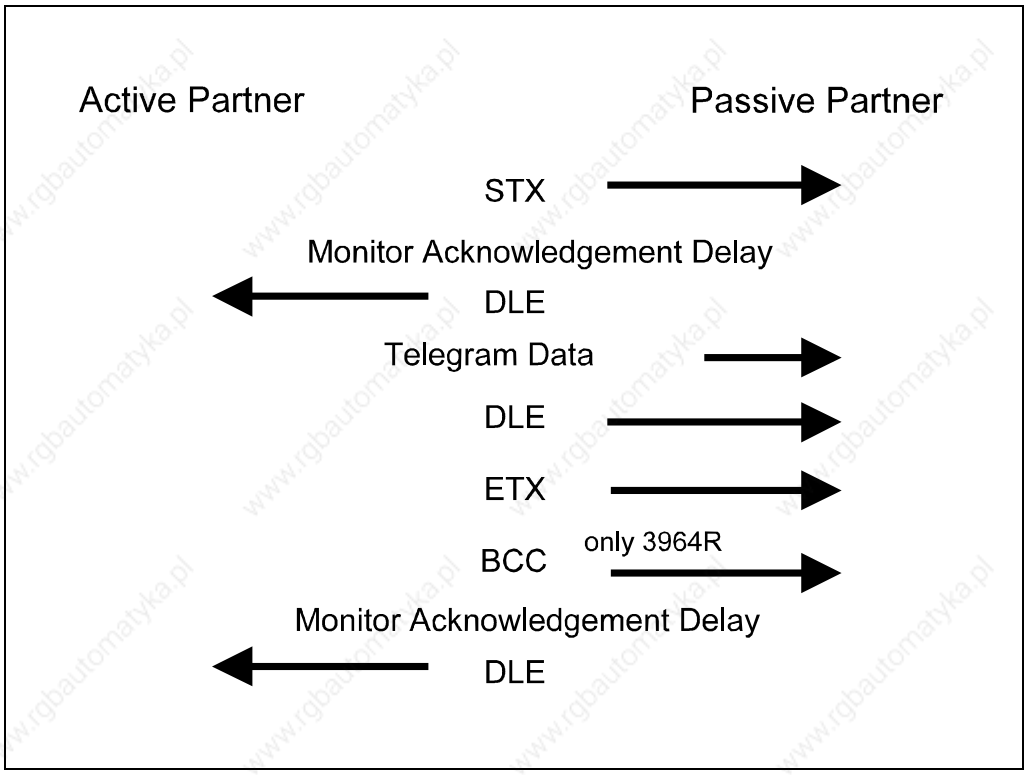


Fig. 4-3: Procedure 3964(R)

You can transmit max. 250 bytes per telegram.

4.1.7.3 FB37 (SYNCHRON) from firmware 401xV50 and all of 4118 for 3964(R)

This data handling block prepares the interface building block for the operation of the procedure 3964(R).

The data handling block must be loaded in all start branches (OB20,21,22) for each interface channel to be used. If 2 channels are used with the procedure 3964(R), this data handling block must be loaded twice.

The data handling block has formal operands. Several operating parameters are to be set through these formal operands. They are only directly addressable.

The following parameters are to be transmitted when loading FB37:

Des.	Format	Explanation
ADRE	KH	Address set with the DIL switches
SSNR	KF	Interface number 1,2,3
BAUD	KF	Baud rate code 1...11 (10 = 9600Baud)
FORM	KM	Transmission format:8bit,even parity,2 stop
PRIO	KF	0=low, 1=high priority
BLOC	KF	Block size in bytes (always even-numbered!)
ZVZ	KF	Character delay time (value*10ms)
QVZ	KF	Acknowledgement delay time (value*10ms)
PAFE	MB	Byte for the output of a wrong number
NEUS	M	Blocks the building block

Tab. 4-10: List of parameters for the loading of FB37

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number1, 2 or 3 is to be given.

BAUD Different baud rates can be set by inputting the respective codes. With the plug connector J11 on the building block, an optional baud rate of 150 to 38400 baud can be set. In order to achieve this, J11, which has a baud rate of 75 to 19200 baud in its delivery state, must be replugged.

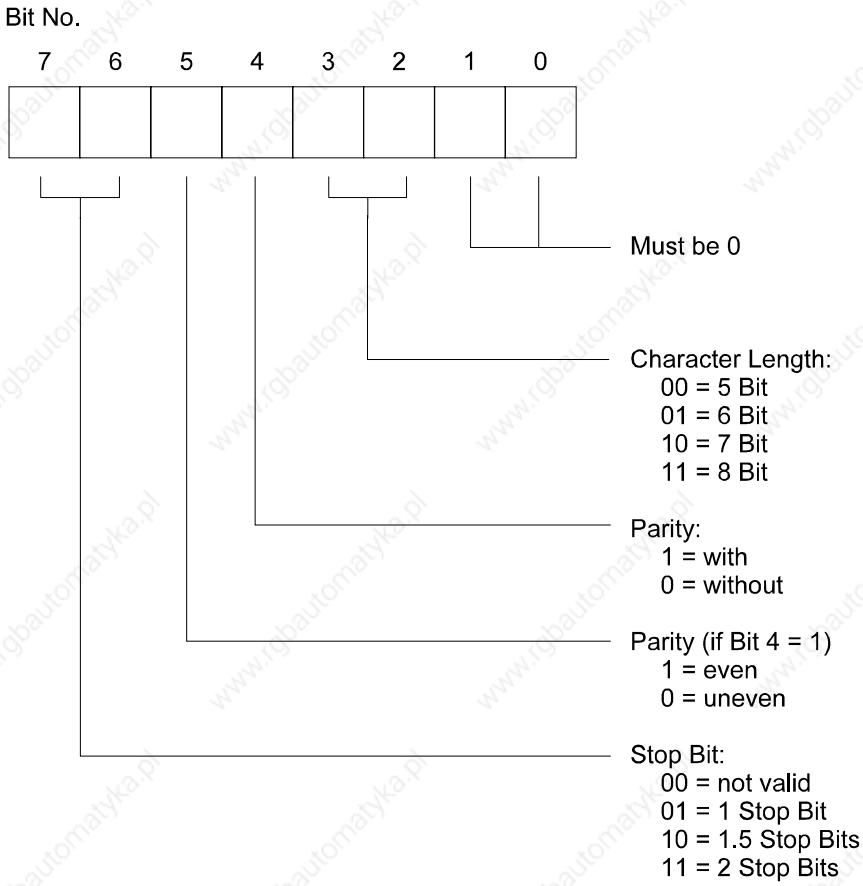
Code (decimal)	Plug Connector J11 in Position: 38400 / 19200 J11	Plug Connector J11 in Position: 38400 / 19200 J11
01	75	150
02	100	(200)
03	110	(220)
04	150	300
05	300	600
06	600	1200
07	1200	2400
08	2400	4800
09	4800	9600
10	9600	19200
11	19200	38400

FORM Transmission format: The command is a two-byte-command.

High-Byte 0 = Operation with a procedure 3964

1 = Operation with a procedure 3964(R)

Low-Byte Transmission format of the serial interface.



PRIO Priority: Controls the telegram procedure by initialization conflict

0 = low priority

1 = high priority

BLOC The block size indicates how many bytes are to be exchanged between the PLC and the interface building block every time the data handling blocks SEND and RECEIVE are loaded. The greater the number of bytes to be exchanged, the higher the cycle time load.

The block size has no influence on the data transmission! The building block only begins with the output of telegrams, when all data to be output has been transferred from the PLC to the building block.

The standard setting is a block size of 128 bytes. Only for "slow" PLCs e.g. CPU 941/942, can it be necessary to set the block size at a smaller value, so that the cycle time load is not too large.

ZVZ Character delay time in 10ms steps

QVZ Acknowledgement delay time in 10ms steps

PAFE A byte contains a wrong number for output. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

A value unequal to 0 represents an error, the given interface is not initialized.

PAFE Error codes:		**registered by:**
0	No error	Data handling block
1	Wrong command	Interface building block
2	Wrong channel number	Interface building block
3	Wrong procedure number	Interface building block
4	Initialization wrong	Interface building block
5	Wrong clock command	Interface building block
6	Invalid clock data	Interface building block
7	Parametrizing wrong	Interface building block
8	Wrong number of parameters	Interface building block
9	Not used	
10	Not used	
11	Invalid value on the operand ADRE	Data handling block
12	Invalid value on the operand SSNR	Data handling block
13	Invalid value on the operand BAUD	Data handling block
14	Invalid value on the operand PRIO	Data handling block
15	Invalid value on the operand BLOC	Data handling block
16	Undefined answer from interface building block	Data handling block
17	Invalid value in high byte FORM PROZ <> 3964 or 3964R	Data handling block
18	Character delay time is 0	
19	Acknowledgement delay time is 0	

NEUS If the data handling block is run with a set new start bit, the building block is blocked for all SPS accesses. The building block is again released by loading the FB28 with a set new start bit.

The data handling block examines the parametrized values by the following limits:

ADRE	F080...F1FC
SSNR	1...3
BAUD	1...11
FORM	Data handling block examines High-Byte: 0 = 3964 1 = 3964(R) Low Byte: Transmission format of the serial interface.
PRIO	0/1
BLOC	16...128, always even-numbered
ZVZ	1...65535
QVZ	1...65535

Example: FB37 (SYNCHRON)

```
         SPA    FB37
NAME     :SYNCHRON
ADRE     :KH F080                  Building block's address as set by DIL
                                   switches
SSNR     :KF +1                    Interface number 1,2,3
BAUD     :KF +10                   Baud rate code 1...11 (10 = 9600baud)
FORM     :KM 00000001 11111100     Transmission format:8bit,even parity,2 stop
PRIO     :KF +0                    0=low, 1=high priority
BLOC     :KF +128                  Block size in bytes (always even-numbered!)
ZVZ      :KF +20                   200ms Character delay time(Value*10ms)
QVZ      :KF +500                  5000ms Acknowledgement delay time
                                   (Value*10ms)
PAFE     :MB 5                     Byte for the output of a wrong number
NEUS     :M  1.0                   Blocking of the building block
```

4.1.7.4 FB30 (SYNCHRON) up to firmware 4.1 for 3964(R)

This data handling block prepares the interface building block for operation of the procedure 3964(R).

The data handling block must be loaded in all start branches (OB20,21,22) for each interface to be used. If two channels are used with the procedure 3964(R), this data handling block must be loaded twice.

The data handling block possesses formal operands.

Several working parameters are to be set through these formal operands. They are only directly addressable.

The following parameters are to be transmitted when loading FB30:

Des.	Format	Explanation
ADRE	KH	Building block's address set by DIL switches
SSNR	KF	Interface number 1,2,3
BAUD	KF	Baud rate code 1...11 (10 = 9600baud)
FORM	KM	Transmission format:8bit,even parity,2 stop
PRIO	KF	0=low, 1=high priority
BLOC	KF	Block size in bytes (always even-numbered!)
PAFE	MB	Byte for the output of a wrong number

Tab. 4-11: List of parameters for the loading of FB30

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1, 2 or 3 is to be given.

BAUD Different baud rates can be set by inputting the respective codes. With the plug connector J11 on the building block, optional baud rates from 150 to 38400 baud can be set. In order to achieve this, J11 which has a baud rate of 75 to 19200 baud in its delivery state, must be replugged.

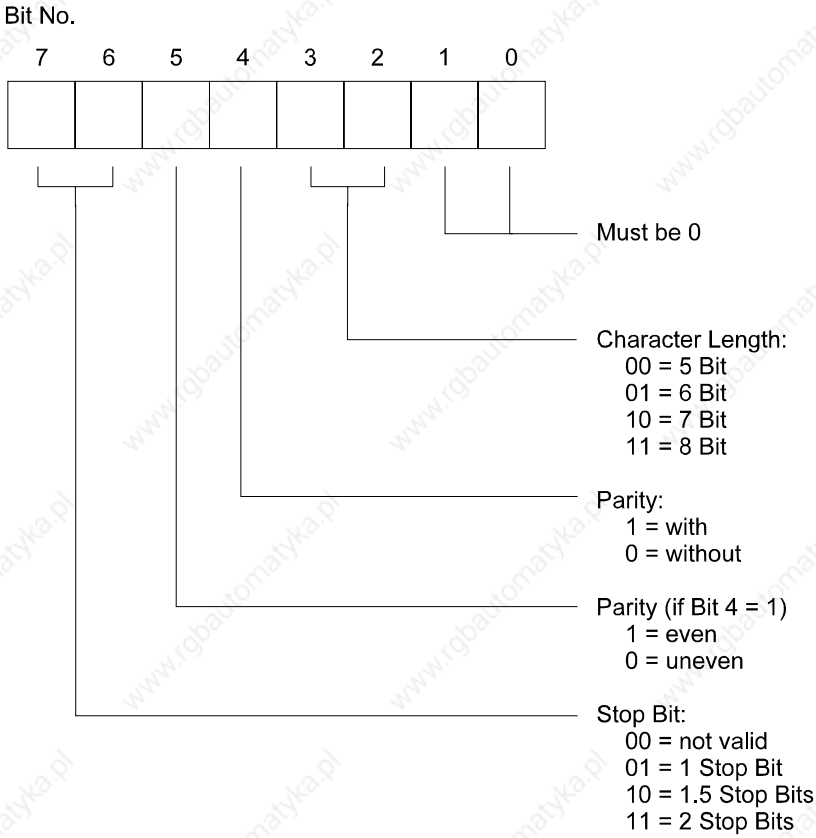
Code (decimal)	Plug Connector J11 in Position 38400 / 19200 J11	 38400 / 19200 J11
01	75	150
02	100	(200)
03	110	(220)
04	150	300
05	300	600
06	600	1200
07	1200	2400
08	2400	4800
09	4800	9600
10	9600	19200
11	19200	38400

FORM Transmission format: The comand is a two-byte-command.

High-Byte 0 = Operation with the procedure 3964

1 = Operation with the procedure 3964(R)

Low-Byte Transmission format of the serial interface.



PRIO Priority: Controls the telegram procedure by initialization conflict

0 = low priority

1 = high priority

BLOC The block size indicates how many bytes are to be exchanged between the PLC and the interface building block every time the data handling blocks SEND and RECEIVE are loaded. The greater the number of bytes to be exchanged, the greater the cycle time load.

The block size has no influence on the data transmission! The building block only starts to output telegrams, when all data to be transmitted has been transferred from the PLC to the building block.

The standard setting is a block size of 128 bytes. Only for "slow" PLCs e.g. CPU 941/942 is it sometimes necessary to set the block size to a smaller value, so that the cycle time load is not too large.

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

A value uneven to 0 represents an error, the given interface is not initialized.

	PAFE Error Codes	**registered by:**
0	No Error	Data handling block
1	Wrong Command	Interface building block
2	Wrong channel number	Interface building block
3	Wrong procedure number	Interface building block
4	Initialization wrong	Interface building block
5	Wrong clock command	Interface building block
6	Invalid clock data	Interface building block
7	Parametrizing wrong	Interface building block
8	Wrong number of parameters	Interface building block
9	Not used	
10	Not used	
11	Invalid value on operands ADRE	Data handling block
12	Invalid value on operands SSNR	Data handling block
13	Invalid value on operands BAUD	Data handling block
14	Invalid value on operands PRIO	Data handling block
15	Invalid value on operands BLOC	Data handling block
16	Undefined answer from interface building block	Data handling block
17	Invalid value in high byte FORM PROZ <> 3964 or 3964R	Data handling block

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...3
BAUD	1...11
FORM	Data handling block examines
	High-Byte: 0 = 3964
	1 = 3964(R)
	Low Byte: Transmission format of the serial interface
PRIO	0/1
BLOC	16...128, always even-numbered

Example: FB30 (SYNCHRON)

```
          SPA     FB30
NAME      :SYNCHRON
ADRE      :KH F080                  Building block's address as set by DIL
                                    switches
SSNR      :KF +1                    Interface number 1,2,3
BAUD      :KF +10                   Baud rate code 1...11 (10 = 9600baud)
FORM      :KM 00000001 11111100     Transmission format:8bit,even parity,2 stop
PRIO      :KF +0                    0=low, 1=high priority
BLOC      :KF +128                  Block size in bytes (always even-numbered!)
PAFE      :MB 5                     Byte for the output of a wrong number
```

4.1.7.5 FB31 (RECEIVE), data reception for 3964(R)

This data handling block is used for the importing of a telegram from the interfaces BG41, BG42 and BG43 when using the procedure 3964(R).

In FB31 lubricating flags are used. These must be saved when loading the FB31 from a time or alarm protected OB.

The data handling block posseses formal operands. Several working parameters are to be set through these formal operands. They can be directly as well as indirectly parametrized.

The following parameters are to be transmitted when loading FB31:

Des.	Format	Explanation
ADRE	KH	The address set with DIL switches
SSNR	KF	Interface number 1,2,3
A-NR	KF	Order number(1...255)
ZTYP	KC	Details of data type
DBNR	KY	Number of data blocks detailed under target type of given function block
BWNR	KY	Byte/word number from which an incoming telegram should be filed.
MXBW	KF	Maximum word length of incoming telegram
ANZW	MW	The display word is filed under this word
PAFE	MB	Error messages in relation to parametrizing are filed in this byte
NEUS	M	New start bit

Tab. 4-12: List of parameters for the loading of FB31

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1, 2 or 3 is to be given.

A-NR Order Number, a number in the area of 1...255. This number may not be given to any other data handling block.

ZTYP The telegram's target type is to be entered into the PLC.
Possible target types (data types) are:

DB	for data block
EB	for the process image of the inputs
AB	for the process image of the outputs
MB	for flag bytes
Z	for counter words
T	for timer words

Other details result in the parametrizing error message (PAFE) 21.

DBNR If the target type DB is selected, the DBNR is to be entered in the right byte. Valid area: 2...255. If this area is violated, it results in PAFE: 17. If the given DB does not exist, the data handling block will therefore register PAFE: 19.

BWNR Byte/word number. By means of these operands it is determined, from which word, e.g. data word of the given DB, or from which byte, e.g. AB, a received telegram should be filed.

MXBW Maximum number of bytes/words. With these operands it is determined, what the maximum length of a telegram should be. By means of the operands BWNR and MXBW, the data handling block examines if the target area, e.g. a DB, is long enough. If this is not the case, the data handling block registers PAFE: 20.

ANZW Display word. The display word occupies a word. In the right byte status bits are filed. In the left byte a wrong number is filed, when the identifier “finished with error“ appears in the right byte.

The status identifiers in the right byte are structured as follows:

Bit 0: Not used
Bit 1: Job is running
Bit 2: Finished without an error
Bit 3: Finished with an error, → Error number in the left byte
Bit 4: Interface momentarily occupied
Bit 5-7: Not used

The data handling block imports at the polling of the interface building block only as many bytes, as were parametrized as block size on the data handling block SYNCHRON.

Due to these reasons the data handling block RECEIVE must often be loaded several times, in order to import a complete telegram from the interface building block. If the data handling block has one or more data blocks but has not yet imported the last data block from the interface building block, it places as an identifier for “job running“ bit 1 in the right byte of the display word.

If all data blocks have been correctly imported from the interface building block, the data handling block places as identifcation for "finished without an error" the bit 2 in the right byte of the display word.

If the telegram could not be imported from the interface building block, e.g. the telegram is too long, the data handling block places as identifier for "finished with error" the bit 3 in the right byte of the display word. In this case an error number is then in the left byte of the display word.

Error Code, delivered by ANZW:

05	Telegram header contains an invalid target type
09	Reciever does not answer with DLE
14	Wrong telegram header (the first 2 bytes)
15	Transmission disconnected by receiver
16	Wrong command code 2. letter (invalid target area)
20	DB error (DB0 used, DB too short, DB not available)
21	Receiver does not answer within QVZ (Timeout on SSM building block)
22	Wrong command code 1. letter (invalid source area)
23	QVZ in connection set-up
50	Coordination flag set
52	Data length greater than 128 bytes

PAFE Contains a byte for the issuing of an error number. If PAFE provides a value of 0, the parameter details are correct. A value unequal to 0 represents an error.

PAFE Error codes

0	No error
11	Invalid address ADRE
12	Invalid interface number SSNR
16	Undefined answer received from interface building block
17	Invalid DB no. DBNR
18	Invalid byte/word number BWNR
19	DB not available DBNR
20	DB too short (by indirect parametrizing)
21	Invalid target type ZTYP
23	Target area too short BWNR+MXBW
26	Invalid order number
31	Invalid number of valid data at reception MXBW

NEUS As long as the bit is at the signal state 1, the processing of the function block is discontinued.

The data handling block examines the parametrized values by the folllowing limits:

ADRE	F080...F1FC
SSNR	1...3
A-NR	1...255
ZTYP	DB,EB,AB,MB,T ,Z .
DBNR	2...255
BWNR	depending on target area
MXBW	depending on targert area (1...)

The target area must be at least as long as was given by the operands BWNR and MXBW.

ANZW	No examination
PAFE	No examination
NEUS	No examination

Example: FB31 (RECEIVE) by direct parametrizing:

```
     :SPA FB31
NAME :RECEIVE
ADRE : KH F080   The address set by the DIL switches
SSNR : KF +1     Interface number 1,2,3
A-NR : KF +3     Order number (1...255)
ZTYP : KC DB     Target type
DBNR : KY 0,5    DB no. by target type DB
BWNR : KY 0,20   Telegram filed from DW 20
MXBW : KF +100   Incoming telegrams may have a maximum length of 100 words
ANZW : MW 30     In this word the display word is filed
PAFE : MB 32     In this byte error messages in relation to parametrizing are
                 filed
NEUS : M  1.0    New start bit
```

Example: FB31 (RECEIVE) by indirect parametrizing:

If the operands are to be indirectly transferred, it is possible in the following form. XX is to be parametrized on the operand ZTYP. This is the identifier for indirect parametrizing. The DB number, in which the operands lie, are to given on the operand DBNR. The data word number, from where the operands lie, is to be given on the operand BWNR. The operands ADRE, SSNR, A-NR, MXBW have in this case no meaning. The operands ANZW, NEUS and PAFE can not be indirectly parametrized. They must still be given directly to the formal operands.

```
     :SPA FB31
NAME :RECEIVE
ADRE : KH 0000    Irrelevant
SSNR : KF +0      Irrelevant
A-NR : KF +0      Irrelevant
ZTYP : KC XX      Identifier: indirect parametrizing
DBNR : KY 0,6     Parameter lies in DB 6
BWNR : KY 0,50    from DW 50
MXBW : KF +0      Irrelevant
ANZW : MW 30      The display word is filed in this word
PAFE : MB 32      Error messages in relation to parametrizing are filed under
                  this byte
NEUS : M  1.0     New start bit
```

Allocation of the given DBs:

```
DB 6:

DW 50 KH ADRE
DW 51 KF SSNR
DW 52 KF A-NR
DW 53 KC ZTYP
DW 54 KY DBNR
DW 55 KY BWNR
DW 56 KF MXBW
```

4.1.7.6 FB33 (SEND), data output for 3964(R)

This data handling block is used to actively transmit a telegram to the interface building blocks BG41, BG42 and BG43 when using the procedure 3964(R).

Lubricating flags are used in FB33. These must be saved before loading FB31 from a time or alarm protected OB.

The data handling block possesses formal operands.

Several working parameters are to be set by these formal operands. They can be both directly and indirectly parametrized.

The following parameters are to be transferred when loading FB33:

Des.	Format	Explanation
ADRE	KH	The address set with DIL switches
SSNR	KF	Interface number 1,2,3
A-NR	KF	Order number (1...255)
QTYP	KC	Source type
QDBN	KY	DB no. by source type DB or DX
QBWN	KY	Byte/word number form where data is to be sent
BWAN	KF	Amount of data to be sent
ANZW	MW	The display word is filed in this word
PAFE	MB	Error messages or parametrizing are filed in this byte
NEUS	M	New start bit

Tab. 4-13: List of parameters for the loading of FB33

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1, 2 or 3 is to be given.

A-NR Order number, a number in the area of 1...255. This number may not be given to any other data handling block.

QTYP The source type is to be given in its own PLC.
Possible source types are:

DB	for data blocks
DX	for data block development
EB	for the process image of inputs
AB	for the process image of outputs
MB	for flag bytes
Z	for counter words
T	for timer words

Other details result in a parametrizing error message PAFE: 27.

QDBN If the source type DB or DX are selected, the DB no. is to be given into the right byte. Valid area: 2...255. If this area is violated a PAFE: 28 is registered.
If the given DB or DX do not exist, the data handling block registers a PAFE: 29.

QBWN Byte/word number. By means of these operands it is determined, from which, e.g. data word of the given DB, or from which byte, e.g. AB, the data is to be sent.

BWAN Gives the amount of data in bytes/words, which are to be sent. At least 1 byte/word must always be sent, otherwise the data handling block registers a PAFE: 31

ANZW Display word. The display word contains one word. Status bits are filed in the right byte. An error number is filed in the left byte, if the idenfication “finished with an error“ appears in the right byte.

The status identifiers in the right byte are structured as follows:

Bit 0: Not used
Bit 1: Job running
Bit 2: Finished without an error
Bit 3: Finished with an error, → Error number in the left byte
Bit 4: Interface momentarily occupied
Bit 5-7:Not used

The data handling block hands over at the loading of the interface building block only enough bytes, as were parametrized as block size on the data handling block SYNCHRON.

For these reasons the data handling block SEND may have to be loaded several times, in order to transmit a complete telegram to the interface building block. If the data handling block has one or more data blocks, but has not yet transmitted the last data block to the interface building block, bit 1 is placed in the right byte of the display word as identifier for “job running.“

If all data blocks have correctly been received by the interface building block and the receiver does not report an error, the data handling block SEND places the Bit 2 in the right byte of the display word as identifier for “finished without an error.“

If the telegram from the interface building block has not been received, e.g. receiver does not answer, or when the receiver reports an error, the data handling block SEND places the bit 3 in the right byte of the display word as identifier for “finished with an error.“ In this case an error number is in the left byte of the display word.

Error code, delivered by ANZW:

05	Telegram header contains invalid target type
09	Receiver does not answer with DLE
14	Wrong telegram header (the first 2 bytes)
15	Transmission terminated by receiver
16	Wrong command code 2. letter (invalid target area)
20	DB error (DB0 used, DB too long, DB not available)
21	Receiver does not answer within QVZ (Timeout on SSM building block)
22	Wrong command code 1. letter (invalid source area)
23	QVZ in connection set-up
50	Coordination flag set
52	Data length greater than 128 bytes

PAFE Contains a byte for the output of an errror number. If PAFE has a value of 0, the parameter details are correct. A value uneven to 0 represents an error.

PAFE Error codes

0	No error
11	Invalid address ADRE
12	Invalid interface number SSNR
16	Undefined answer received from interface building block
17	Invalid DB no. by indirect parametrizing DBNR
18	Invalid byte/word number by indirect parametrizing BWNR
19	DB not available by indirect parametrizing DBNR
20	DB too short by indirect parametrizing
26	Invalid order number A-NR
27	Invalid source type QTYP
28	Invalid source DB number QDBN
29	Source DB not available
31	Invalid amount of valid data at reception MXBW
40	When transmitting from a wide byte area, e.g. MB into a wide word area e.g. DB, the amount of data BWAN must be even.

NEUS As long as the bit is at the signal state 1, the processing of the function block is discontinued.

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...3
A-NR	1...255
QTYP	DB,DX,EB,AB,MB,T,Z
QDBN	2...255
QBWN	depending on source type (0...)
BWAN	depending on source type (1...)
ANZW	not examined
PAFE	not examined
NEUS	not examined

Example: FB33, (SEND) by direct parametrizing

```
     :SPA FB33
NAME :SEND
ADRE : KH F080    The address set with DIL switches
SSNR : KF +1      Interface number 1,2,3
A-NR : KF +4      Order number(1...255)
QTYP : KC MB      Source type
QDBN : KY 0,0     DB no. by source type DB
QBWN : KY 0,100   From data in the source type
BWAN : KF 10      Amount of data to be sent (Bytes/words depending on source
                  type)
ANZW : MW 30      The display word is filed in this word
PAFE : MB 32      Error messages or parametrizing are filed in this byte
NEUS : M  1.0     New start bit
```

Example: FB33 (SEND) by indirect parametrizing

If the operands are to be transferred indirectly, it is possible in the following form. XX is to be parametrized on the operand QTYP. This is the identifier for indirect parametrizing. The DB number, in which the operands lie, is to be given on the operands QDBN. The DB's data word number, from where the operands lie, must be given on the operands QBWN. The remaining operands are in this case of no meaning. The operands ANZW, NEUS and PAFE can not be indirectly parametrized. They must still be directly given to the formal operands.

```
     :SPA FB33
NAME :SEND
ADRE : KH 0000    Irrelevant
SSNR : KF +0      Irrelevant
A-NR : KF +0      Irrelevant
QTYP : KC XX      Identifier: indirect parametrizing
QDBN : KY 0,6     Parameter lies in DB 6
QBWN : KY 0,50    From DW 50
BWAN : KF +0      Irrelevant
ANZW : MW 30      The display word is filed in this word
PAFE : MB 32      Error messages or parametrizing are filed in this byte
NEUS : M  1.0     New start bit
```

Allocation of the given DBs:

```
DB 6:
DW 50 KH ADRE
DW 51 KF SSNR
DW 52 KF A-NR
DW 53 KC QTYP
DW 54 KY QDBN
DW 55 KY QBWN
DW 56 KF BWAN
```

4.1.7.7 FB28 (FREIGABE), Building block release

This data handling block resets the new start bit and releases the building block for SPS access. The building block is to be loaded at the end of the OB1. The new start bit is set in the OB22 by the user, if an automatic new start identifier is wanted. This bit is to be given by the individual data handling blocks with the identifier “NEUS.“

The data handling block possesses formal operands. The working parameters are to be set by these formal operands.

The following parameters are to be transferred when loading FB28:

Des.	Format	Explanation
ADRE	KH	The address set with DIL switches
NEUS	BI	New start bit
PAFE	BY	Error messages or parametrizing are filed in this byte.

Tab. 4-14: List of parameters for loading FB28

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

NEUS Release of building block when NEUS is set at 1.

PAFE Contains one byte for the issuing of an error number. If PAFE has the value 0 the parameter details are correct. A value unequal to 0 represents an error.

PAFE Error codes

0	No error
11	Invalid address ADRE. An address smaller than F080h was given.

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
NEUS	no examination
PAFE	no examination

4.1.8 Handling with prozedure 3964(R) with RK512

4.1.8.1 General

The building blocks BG41, BG42 and BG43 can be operated with the procedure 3964(R) in connection with the RK512 protocol. It is not necessary to select the procedure with the FB100 or FB101 as this is carried out by the FB27 (SYNCHRON).

A 3964(R) operation is not possible on channel 3 of BG43.
If the 3964(R) operation is required on all three channels, the firmware 4017v5x is to be used.

The procedure 3964 differs from the 3964R in two points of the procedure:

- the BCC byte is dropped in the procedure 3964. DLE/ETX applies for the end of the telegram.
- Acknowledgement delay time

  In procedure 3964: QVZ = 550 ms.
  In procedure 3964R: QVZ = 2000 ms

In order to be able to work with the procedure 3964 or 3964R (called 3964(R) in the following) with the RK512 protocol, the following VIPA data handling blocks have been made available:

SYNCHRON FB27 (from firmware level 401xV50 and 4118)

With this function block the building block is set to operate with the procedure 3964(R) and RK512 protocol, the baud rate and the data transmission format is selected, and the block size and priority are set.

Difference to FB20: Input of a character delay time (ZVZ)
Input of an acknowledgement delay time (QVZ)

SYNCHRON FB20 (up to firmware level 401xV41)

This function block is to be used instead of FB27, if firmware level 4.1 or older are to be used.

RECEIVE FB21

With this function block telegrams are received from partner equipment, whereby the target information from the telegram header are not used, but are replaced by the target definitions given in the formal operands.

SEND FB23

This function block is used for sending an outgoing telegram to a partner equipment.

FETCH FB25

With this function block data is requested from partner equipment.

RECEIVE-ALL FB24

With this function block telegrams are received from partner equipment.

SEND-ALL FB22

From this function block requested telegrams from the partner equipment are answered.

Depending on the case of application only one function block is always required from the group receive (FB21 RECEIVE or FB24 RECEIVE-ALL).

RELEASE BUILDING BLOCK FB28

This function block resets the new start bit and releases the building block for the SPS access. This bit is to be given by the individual data handling blocks.

4.1.8.2 Procedure

The following describes the structure of the procedure and the telegrams:

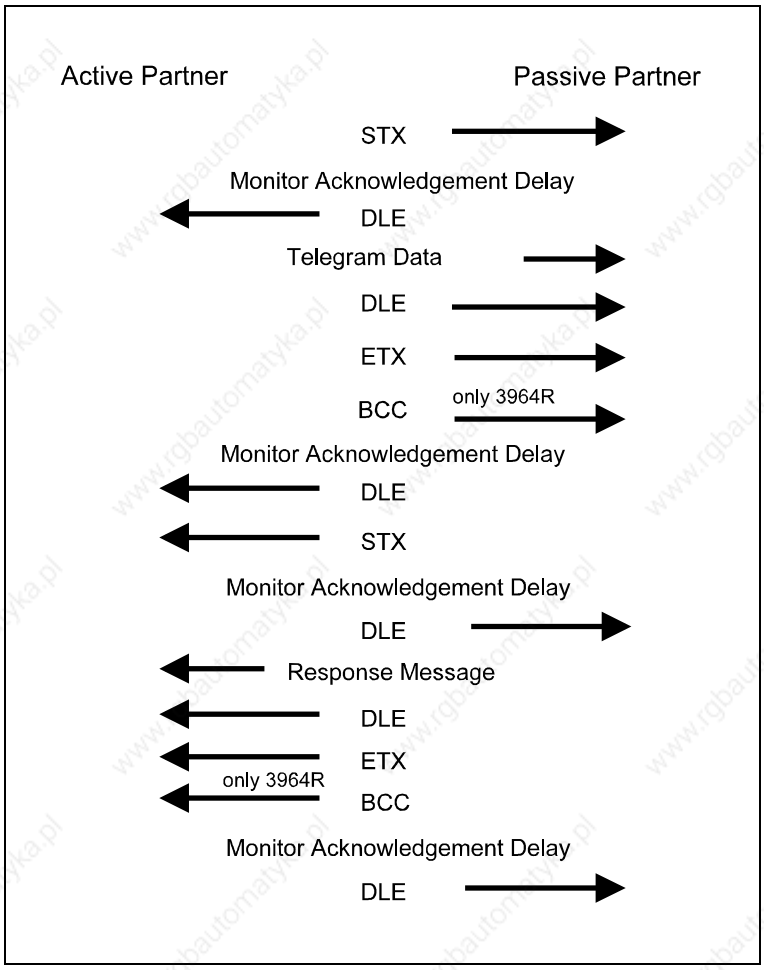


Fig. 4-4: Structure of the procedure 3964(R) with RK512

Time-outs:

The following times are regarded as time-out:

Acknowledgement delay time:	(QVZ) =	2000 ms
Character delay time:	(ZVZ) =	220 ms

QVZ is monitored between STX and DLE as well as between BCC and DLE. ZVZ is monitored during the entire reception of the telegrams.

STX is transmitted again when QVZ is expired to STX. After 5 attempts a NAK is sent and the connection set-up is terminated. The same happens when an NTX or any character has been received after an STX.

When the QVZ is expired after the telegram (after BCC byte) or when a character unequal to DLE is received, the connection set-up and the telegram are repeated. Five attempts are also undertaken, after which a NAK is transmitted and the transmission interrupted.

Passive operation:

When the procedure driver is waiting for the connection set-up and receives a character unequal to STX, it transmits NAK. If it receives an NAK character, the procedure driver does not transmit an answer.

If the ZVZ is transgressed at reception, an NAK is transmitted and a new connection set-up is awaited.

If the procedure driver is not ready when receiving the STX, it transmits an NAK.

The Block Check Character (BCC byte):

In the procedure 3964R a block check character is attached to the end of the telegram for better data protection. The BCC byte is formed by an XOR linkage through the data of the entire telegram including the DLE/ETX.

When a BBC byte is received, which differs from the self ascertains one, an NAK instead of a DLE is transmitted.

Initialisation conflict:

If both partners simultaneously attempt a connection set-up within the QVZ, the partner with the lowest priority transmits a DLE and switches to receive.

DLE:

If the character DLE appears in a telegram, it is doubled by the procedure driver, i.e. DLE/DLE is transmitted. Upon reception, double DLEs are filed as one DLE in the buffer. The combinations DLE/ETX/BCC (only by 3964R) is regarded as the end of the telegram.

The control codes : 02h = STX
03h = ETX
10h = DLE
15h = NAK

4.1.8.3 Logical telegram sequence

SEND (Transmission of data)

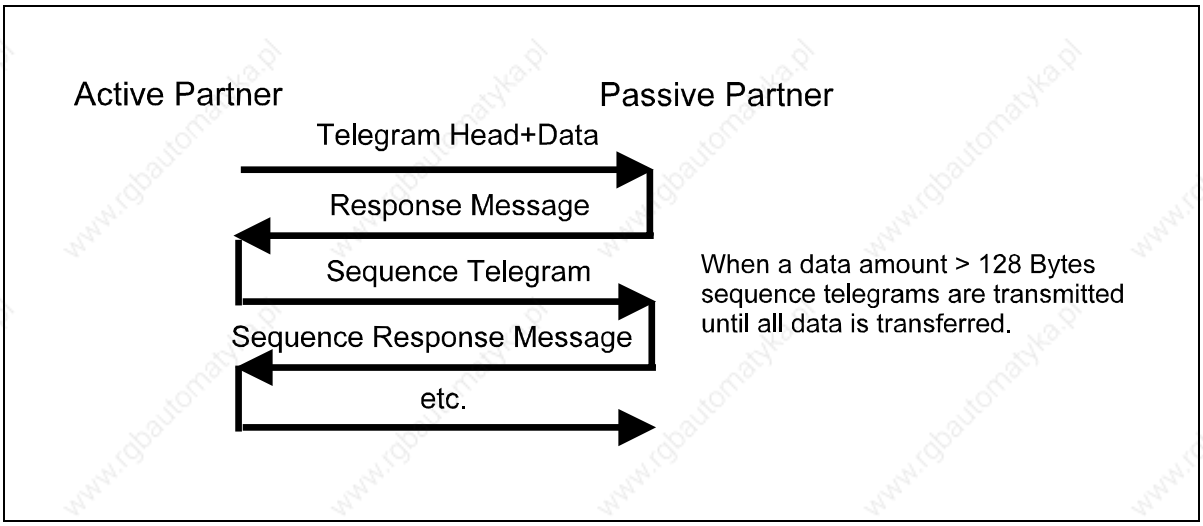


Fig. 4-5: SEND of the procedure 3964(R) with RK512

FETCH (Fetching data)

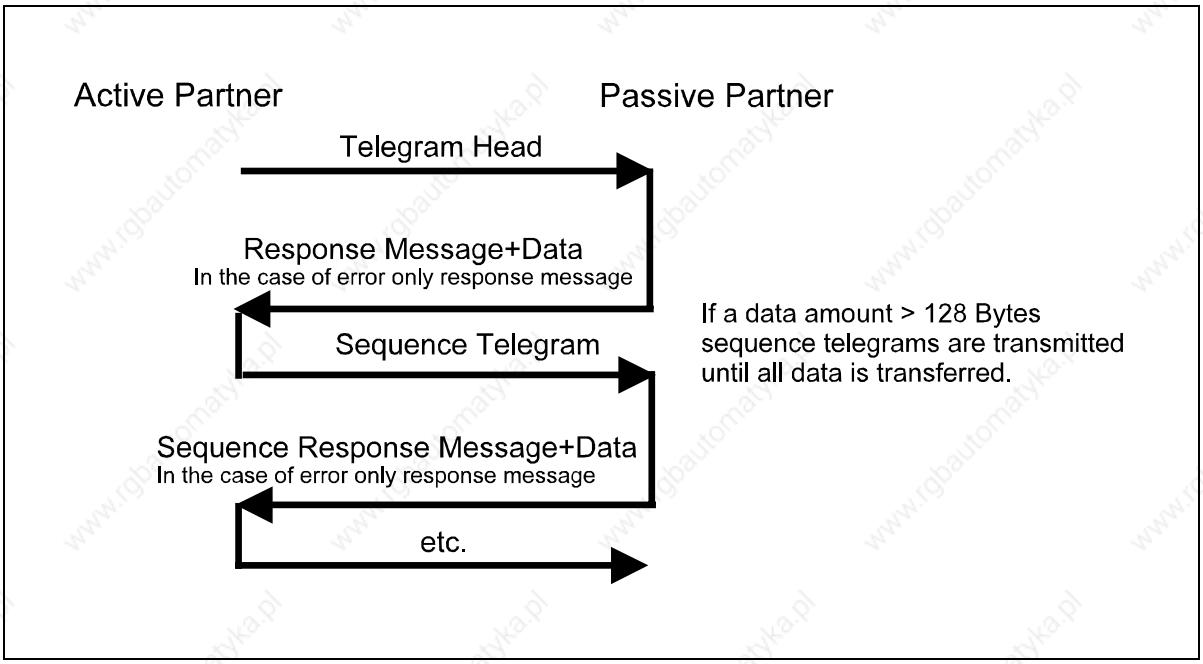


Fig. 4-6: FETCH of the procedure 3964 (R) with RK512

The procedure waits in both cases a maximum of 5s for the response message, after which it stops receiving.

4.1.8.4 Telegram contents

Structure of an output telegram

Each telegram has a header. Depending on the history of the telegram traffic, the header contains all necessary information.

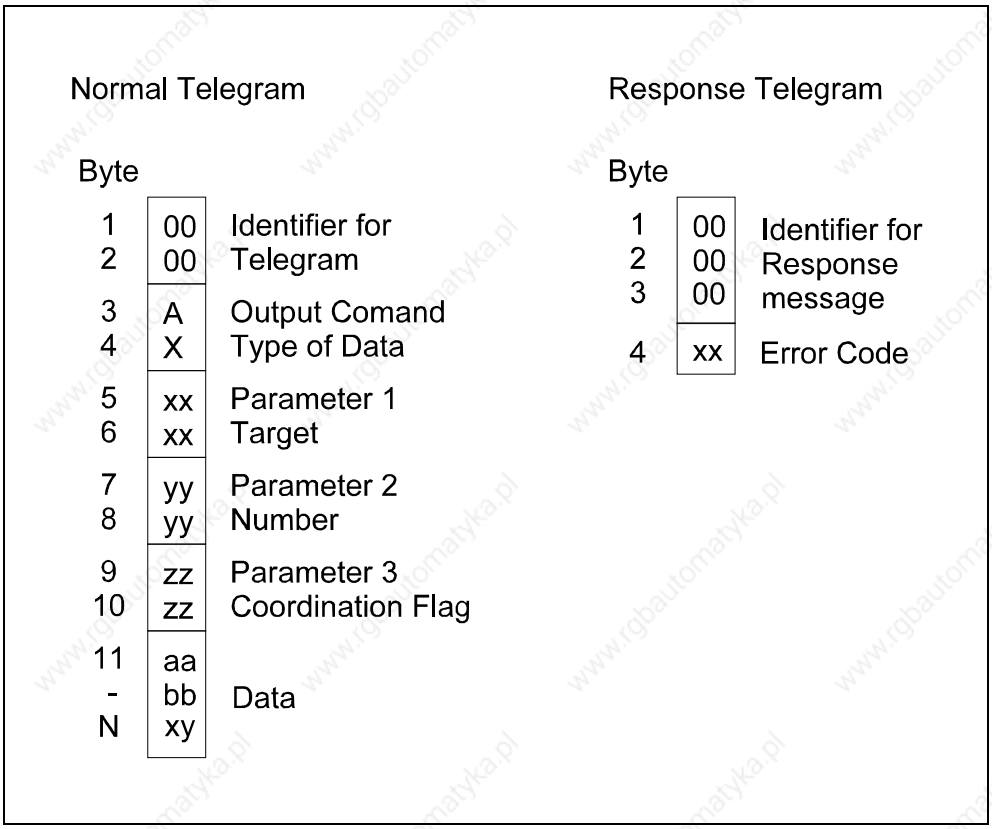


Fig. 4-7: Structure of an output telegram from 3964 (R) with RK512

By data amounts >128 bytes, sequence telegrams are transmitted.

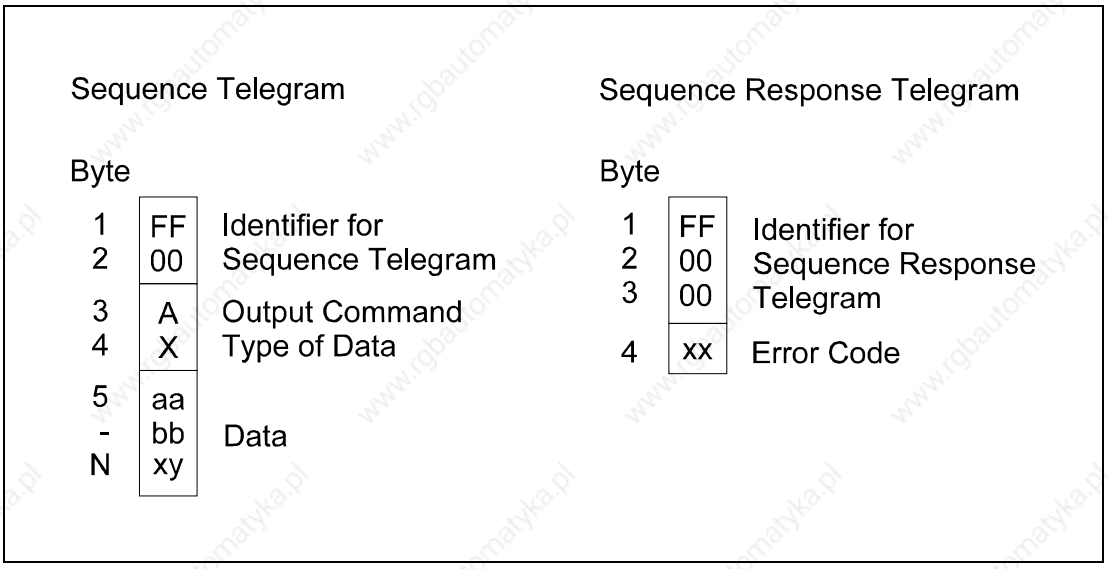


Fig. 4-8: Structure of output sequence telegrams from 3964 (R) with

Example: Output telegram

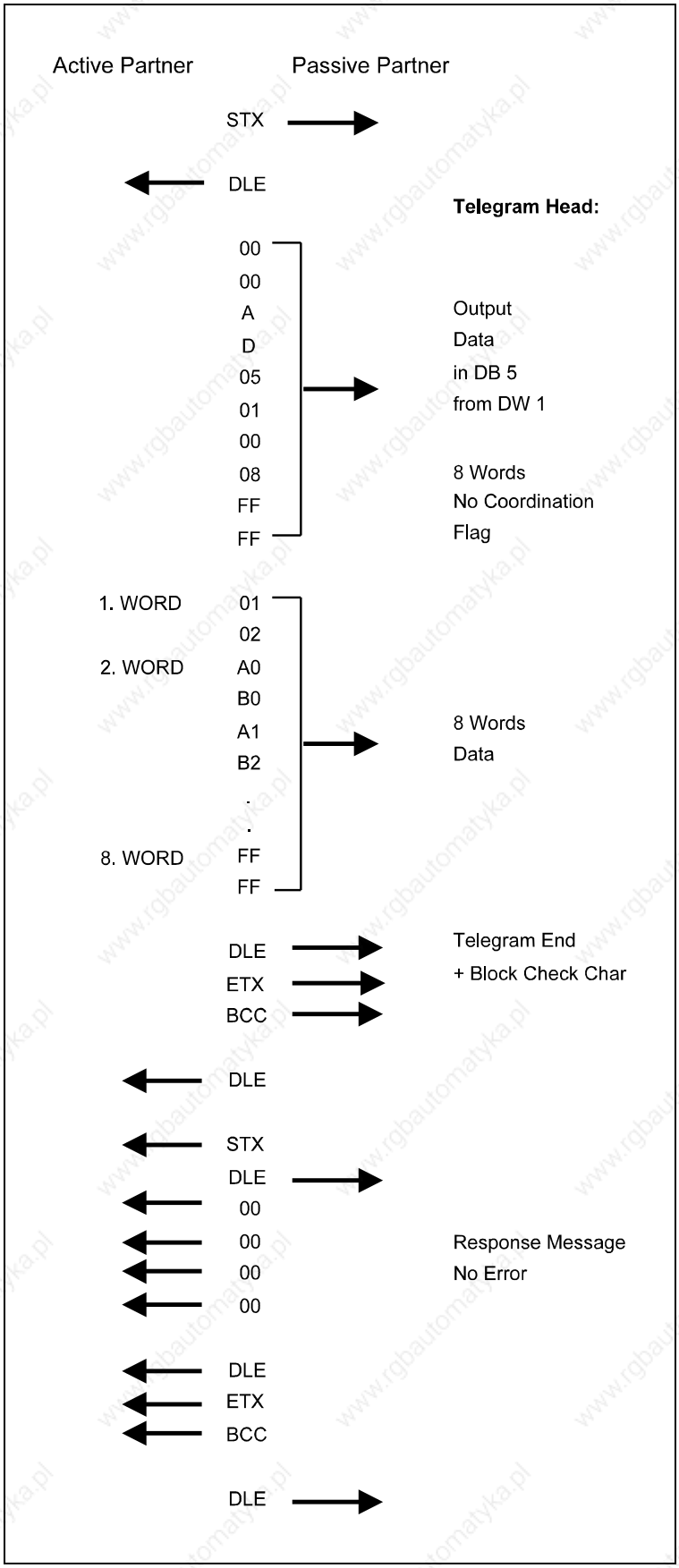


Fig. 4-9: Example of an output

Structure of an input telegram

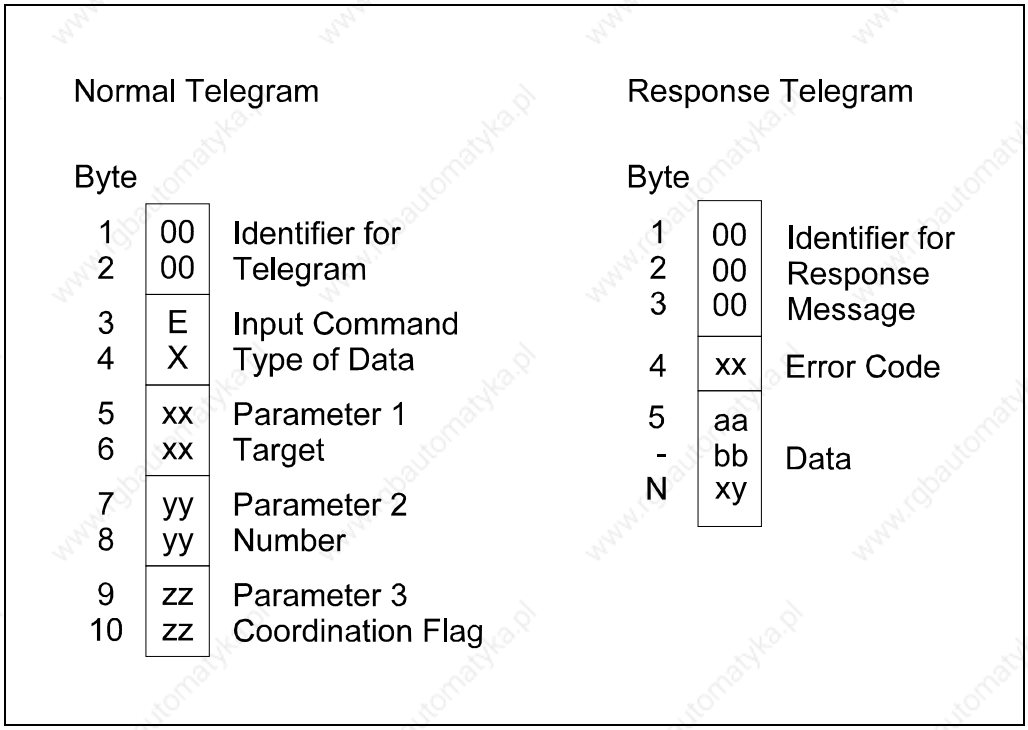


Fig. 4-10: Structure of a normal input telegram

By data amounts >128 Bytes, sequence telegrams are transmitted.

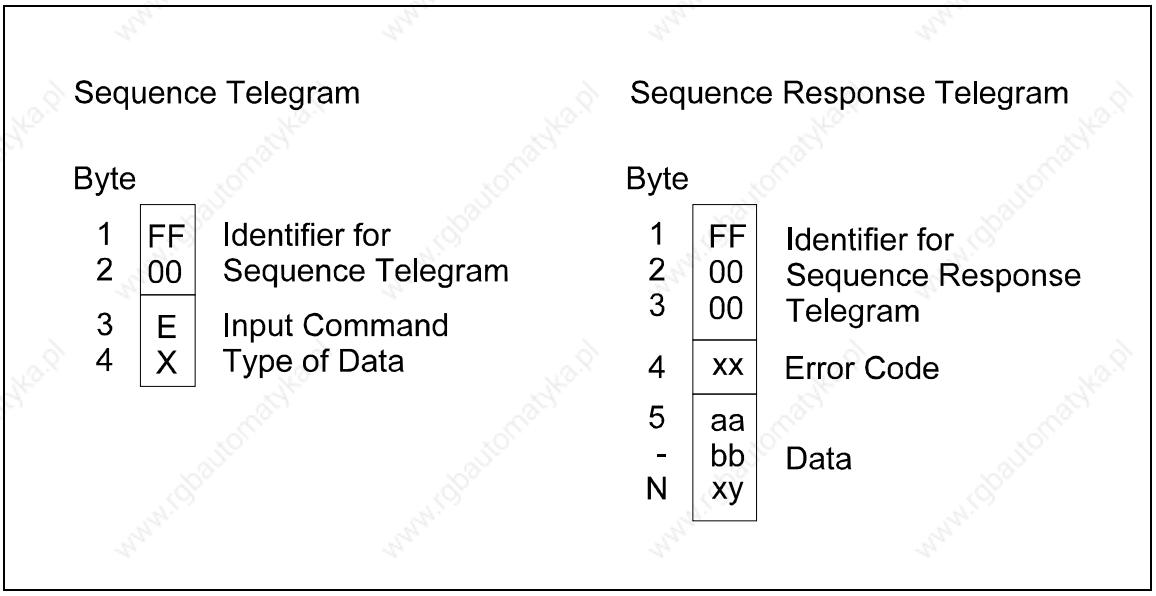


Fig. 4-11: Structure of a sequence input telegramm

Example: Input telegram

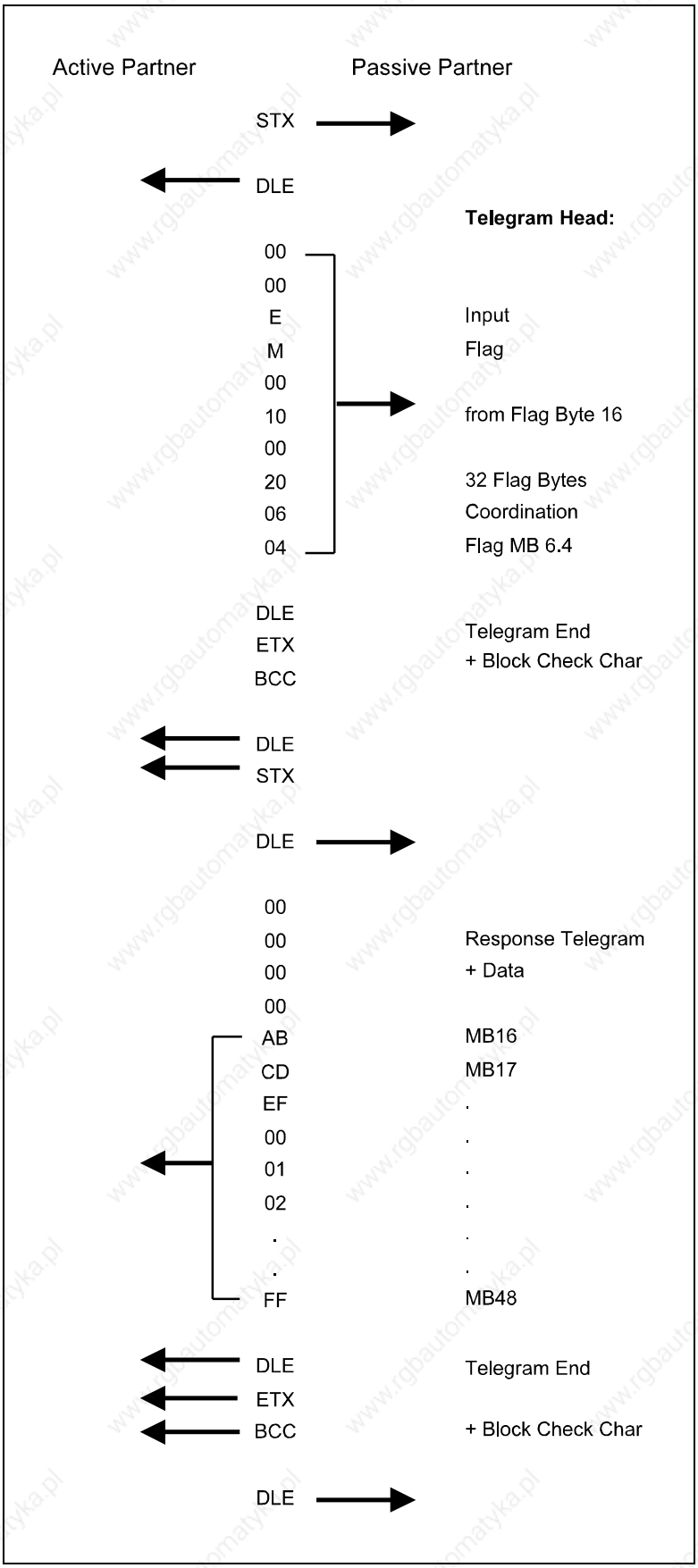


Fig. 4-12: Example of an input telegram

Coordination flag:

The coordination flag is set in active operation in the partner PLC when a telegram is received. This occurs both by input and output commands. If the coordination flag is set and a telegram with this flag has been received, the data is not accepted (or passed on), but an error response message is transmitted (error code 32h). In this case the coordination flag is to be reset by the user in the partner PLC. If a telegram does not contain a coordination flag, FFFFh must be entered.

4.1.8.5 FB27 (SYNCHRON) of firmware 401xV50 and 4118 for 3964(R) with RK512

This data handling block prepares the interface building block for operation of the procedure 3964(R). The function block supports the procedure 3964(R) with the RK512 protocol.

The data handling block must be loaded in all start branches (OB20, 21, 22) for each interface to be used. If two channels are used with the procedure 3964(R), this data handling block must be loaded twice.

The data handling block possesses formal operands. Several working parameters are to be set through these formal operands. They are only directly addressable.

The following parameters are to be transmitted when loading FB27:

Des.	Format	Explanation
ADRE	KH	The address set with DIL switches
SSNR	KF	Interface number 1,2 (3)
BAUD	KF	Baud rate code 1...11 (10 = 9600baud)
FORM	KM	Transmission format:8bit,even parity,2 stop
PRIO	KF	0=low, 1=high Priority
BLOC	KF	Block size in bytes (always even-numbered!)
ZVZ	KF	Character delay time (Value*10ms)
QVZ	KF	Acknowledgement delay time (Value*10ms)
PAFE	MB	Byte for the output of an error number
NEUS	M	Blocking of the building block

Tab. 4-15:List of parameters for loading FB27

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1 or 2 is to be given. By Firmware 4017, the number 3 can also be given.

BAUD Different baud rates can be set by inputting the respective codes. With the plug connector J11 on the building block , optional baud rates from 150 to 38400 baud can be set. In order to achieve this, J11 which has a baud rate of 75 to 19200 baud in its delivery state, must be replugged.

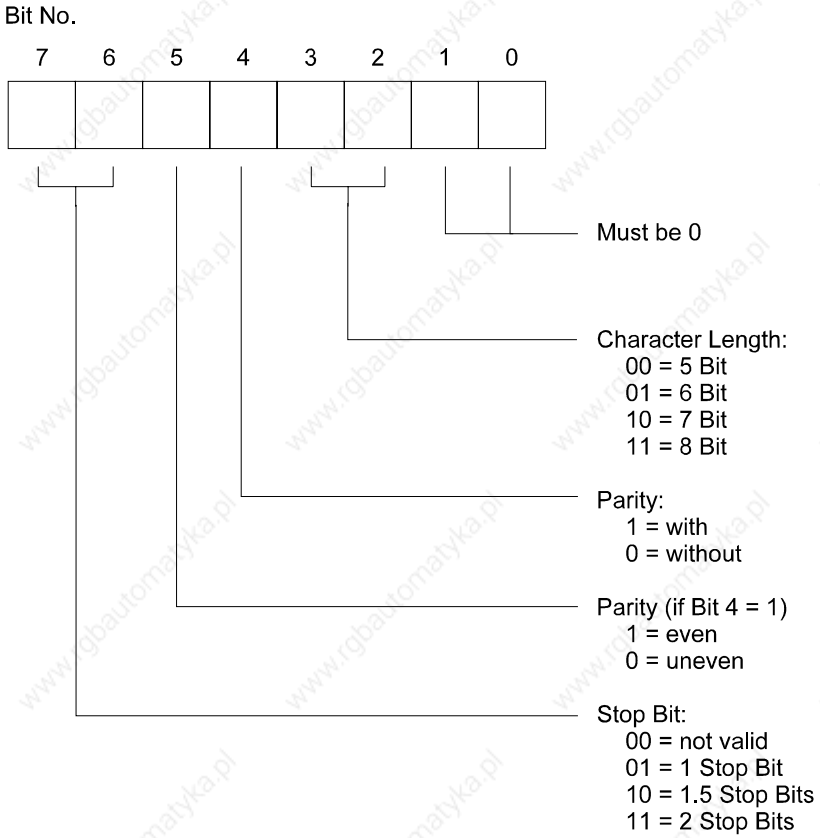
Code (decimal)	Plug Connector J11 in Position: 38400 / 19200 J11	Plug Connector J11 in Position: 38400 / 19200 J11
01	75	150
02	100	(200)
03	110	(220)
04	150	300
05	300	600
06	600	1200
07	1200	2400
08	2400	4800
09	4800	9600
10	9600	19200
11	19200	38400

FORM Transmission format: The command is a two byte command.

High-Byte 0 = Operation with the procedure 3964

1 = Operation with the procedure 3964(R)

Low-Byte Transmission format of the serial interface.



PRIO Priority: Controls the telegram procedure by initialization conflict

0 = low priority

1 = high priority

BLOC The block size indicates how many bytes are to be exchanged between the PLC and the interface building block every time the data handling blocks SEND/SEND-ALL, RECEIVE/RECEIVE-ALL, FETCH are loaded. The greater the number of bytes to be exchanged, the greater the cycle time load. The block size has no influence on the data transmission! The building block only starts to output telegrams, when all data to be transmitted has been transferred from the PLC to the building block.

The standard setting is a block size of 128 bytes. Only for "slow" PLCs e.g. CPU 941/942 is it sometimes necessary to set the block size to a smaller value, so that the cycle time load is not too large.

ZVZ Character delay time in 10ms steps.

QVZ Acknowledgement delay time in 10ms steps.

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

A value unequal to 0 represents an error, the given interface is not initialised.

PAFE Error codes		registered by:
0	No error	Data handling block
1	Wrong command	Interface building block
2	Wrong channel number	Interface building block
3	Wrong procedure number	Interface building block
4	Wrong initialisation	Interface building block
5	Wrong clock command	Interface building block
6	Invalid clock data	Interface building block
7	Wrong parametrizing	Interface building block
8	Wrong number of parameters	Interface building block
9	Not used	
10	Not used	
11	Invalid value on the operands ADRE	Data handling block
12	Invalid value on the operands SSNR	Data handling block
13	Invalid value on the operands BAUD	Data handling block
14	Invalid value on the operands PRIO	Data handling block
15	Invalid value on the operands BLOC	Data handling block
16	Undefined answer from the interface building block	Data handling block
17	Invalid value in high byte FORM PROZ <> 3964 or 3964R	Data handling block
18	Character delay time is 0	Data handling block
19	Acknowledgement delay time is 0	Data handling block

NEUS If the data handling block is run with a set new star bit, the building block is blocked for all SPS accesses. By loading the FB28 with a set new start bit, the building block is once again released.

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...2 (3)
BAUD	1...11
FORM	Data handling block examines High-Byte: 0 = 3964 1 = 3964R Low Byte: Transmission format of the serial interface
PRIO	0/1
BLOC	16...128, always even-numbered
ZVZ	1...65535
QVZ	1...65535

Example: FB27 (SYNCHRON)

```
        SPA     FB27
NAME    :SYNCHRON
ADRE    :KH F080                    The address set with DIL switches
SSNR    :KF +1                      Interface number 1,2 (3)
BAUD    :KF +10                     Baud rate code 1...11 (10 = 9600baud)
FORM    :KM 00000001 11111100       Transmission format:8bit,even parity,2 stop
PRIO    :KF +0                      0=low, 1=high priority
BLOC    :KF +128                    Block size in bytes (always even-numbered!)
ZVZ     :KF +20                     200ms Character delay time (Value*10ms)
QVZ     :KF +500                    5000ms Acknowledgement delay time
                                    (Value*10ms)
PAFE    :MB 5                       Byte for the output of an error number
NEUS    :M  1.0                     Blocking of the building block
```

4.1.8.6 FB20 (SYNCHRON) up to firmware 4.1 for 3964(R) with RK512

This data handling block prepares the interface building block for the operation of the procedure 3964(R).

The block supports the procedure 3964(R) with the RK512 protocol. The data handling block must be loaded in all start branches (OB20,21,22) for each interface to be used. If two channels are used with the procedure 3964(R), this data handling block must be loaded twice.

The data handling block possesses formal operands.

Several working parameters are to be set through these formal operands. They can only be directly addressed.

The following parameters are to be transmitted when loading FB20:

Bez.	Format	Erklärung
ADRE	KH	The address set with DIL switches
SSNR	KF	Interface number 1,2 (3)
BAUD	KF	Baud rate code 1...11 (10 = 9600baud)
FORM	KM	Transmission format:8bit,even parity,2 stop
PRIO	KF	0=low, 1=high priority
BLOC	KF	Block size in bytes (always even-numbered!)
PAFE	MB	Byte for the output of an error number

Tab. 4-16: List of parameters for the loading of FB20

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1 or 2 are to be given. By the firmware 4017, number 3 can also be given.

BAUD Different baud rates can be set by inputting the respective codes. With the plug connector J11 on the building block, optional baud rates from 150 to 38400 baud can be set. In order to achieve this, J11 which has a baud rate of 75 to 19200 baud in its delivery state, must be replugged.

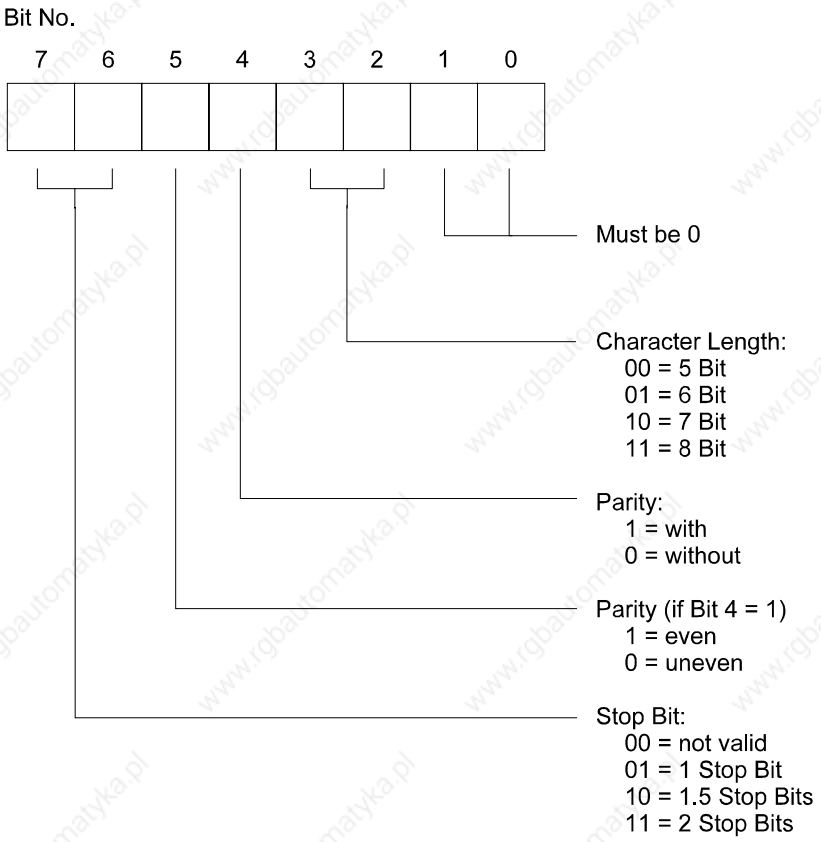
Code (decimal)	Plug Connector J11 in Position: 38400 / 19200 J11	Plug Connector J11 in Position: 38400 / 19200 J11
01	75	150
02	100	(200)
03	110	(220)
04	150	300
05	300	600
06	600	1200
07	1200	2400
08	2400	4800
09	4800	9600
10	9600	19200
11	19200	38400

FORM Transmission format: The command is a two byte command.

High-Byte 0 = Operation with the procedure 3964

1 = Operation with the procedure 3964(R)

Low-Byte Transmission format of the serial interface.



PRIO Priority: Controls the telegram procedure by initialization conflict

0 = low priority

1 = high priority

BLOC The block size indicates how many bytes are to be exchanged between the PLC and the interface building block every time the data handling blocks SEND/SEND-ALL, RECEIVE/RECEIVE-ALL, FETCH are loaded. The greater the number of bytes to be exchanged, the greater the cycle time load.

The block size has no influence on the data transmission! The building block only starts to output telegrams, when all data to be transmitted has been transferred from the PLC to the building block.

The standard setting is a block size of 128 bytes. Only for "slow" PLCs e.g. CPU 941/942 is it sometimes necessary to set the block size to a smaller value, so that the cycle time load is not too large.

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

A value unequal to 0 represents an error, the given interface is not initialised.

PAFE	Error codes	registered by:
0	No error	Data handling block
1	Wrong command	Interface building block
2	Wrong channel number	Interface building block
3	Wrong procdure number	Interface building block
4	Wrong initialisation	Interface building block
5	Wrong clock command	Interface building block
6	invalid clock data	Interface building block
7	Wrong parametrizing	Interface building block
8	Wrong number of parameters	Interface building block
9	Not used	
10	Not used	
11	Invalid value on the operands ADRE	Data handling block
12	Invalid value on the operands SSNR	Data handling block
13	Invalid value on the operands BAUD	Data handling block
14	Invalid value on the operands PRIO	Data handling block
15	Invalid value on the operands BLOC	Data handling block
16	Undefined answer from interface building block	Data handling block
17	Invalid value in high byte FORM PROZ <> 3964 or 3964R	Data handling block

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...2 (3)
BAUD	1...11
FORM	Data handling block examines
	High-Byte: 0 = 3964
	1 = 3964(R)
	Low Byte: Transmission format of the serial interface
PRIO	0/1
BLOC	16...128, always even-numbered

Beispiel: FB20 (SYNCHRON)

```
         SPA    FB20
NAME     :SYNCHRON
ADRE     :KH F080                  Building block's address as set with DIL
                                   switches
SSNR     :KF +1                    Interface number 1,2 (3)
BAUD     :KF +10                   Baud rate code 1...11 (10 = 9600baud)
FORM     :KM 00000001 11111100     Transmission format:8bit,even parity,2 stop
PRIO     :KF +0                    0=low, 1=high priority
BLOC     :KF +128                  Block size in bytes (always even-numbered!)
PAFE     :MB 5                     Byte for the output of an error number
```

4.1.8.7 FB21 (RECEIVE), data reception for 3964(R) with RK512

This data handling block is used to import a telegram from the interface building blocks BG41, BG42 and BG43 when using the procedure 3964(R).

Lubricating flags are used in FB21. These must be saved before loading FB21 from a time or alarm protected OB.

The data handling block posseses formal operands. Several working parameters are to be set through these formal operands. They can be directly as well as indirectly parametrized.

The following parameters are to be transmitted when loading FB21

Des.	Format	Explanation
ADRE	KH	The address set with DIL switches
SSNR	KF	Interface number 1,2 (3)
A-NR	KF	Order number (1...255)
ZTYP	KC	Function block details
DBNR	KY	Number of handling blocks, detailed under the target type of the given function block
BWNR	KY	Byte/word number, from where an incoming telegram is to be filed
MXBW	KF	Maximum length of words of incoming telegrams
ANZW	MW	The display word is filed under this word
PAFE	MB	Error messages or parametrizing are filed in this byte.
NEUS	M	New start bit

Tab. 4-17:List of parameters for the loading of FB21

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1 or 3 is to be given. By the firmware 4017, the number 3 can also be given..

A-NR Order Number, a number in the area of 1...255. This number may not be given to any other data handling block.

ZTYP The telegram's target type is to be entered into the PLC.
Possible target types (data types) are:

DB	for data handling block
DX	for data handling block development
EB	for process image of inputs
AB	for process image of oputputs
MB	for flag bytes
Z	for counter words
T	for timer words

Other details result in a parametrizing error message (PAFE) 21.

DBNR If the target type DB or DX is selected, the DBNR is to be entered in the right byte. Valid area: 2...255. If this area is violated, it results in PAFE: 17. If the given DB does not exist, the data handling block will therefore register a PAFE: 19.

BWNR Byte/word number. By means of these operands it is determined, from which word, e.g. data word of the given DB, or from which byte, e.g. AB, a received telegram should be filed.

MXBW Maximum number of bytes/words. The maximum length of a telegram is determined by these operands. By means of the operands BWNR and MXBW, the data handling block examines if the target area, e.g. a DB, is long enough. If this is not the case, the data handling block PAFE: 20 is registered.

ANZW Display word. The display word occupies a word. In the right byte status bits are filed. In the left byte a wrong number is filed, when the identifier “finished with error“ appears in the right byte.

The status identifiers in the right byte are structured as follows:

Bit 0: Not used
Bit 1: Job is running
Bit 2: Finished without an error
Bit 3: Finished with an error, → Error number in the left byte
Bit 4: Interface momentarily occupied
Bit 5-7:Not used

The data handling block imports at the polling of the interface building block only as many bytes, as were parametrized as block size on the data handling block SYNCHRON.

Due to these reasons the data handling block RECEIVE must often be loaded several times, in order to import a complete telegram from the interface building block. If the data handling block has one or more data blocks but has not yet imported the last data block from the interface building block, it places as an identifier for “job running“ bit 1 in the right byte of the display word.

If all data blocks have been correctly imported from the interface building block, the data handling block places as identifcation for "finished without an error" the bit 2 in the right byte of the display word.

If the telegram could not be imported from the interface building block, e.g. the telegram is too long, the data handling block places as identifier for "finished with error" the bit 3 in the right byte of the display word. In this case an error number is then in the left byte of the display word.

ANZW Error code

05	Telegram header contains invalid target type
09	Receiver does not answer with DLE
14	Wrong telegram header (the first 2 bytes)
15	Transmission terminated by receiver
16	Wrong command code 2. letter (invalid target area)
20	DB error (DB0 used, DB too short, DB not available)
21	Receiver does not answer within QVZ (Timeout on SSM building block)
22	Wrong command code 1. letter (invalid source area)
23	QVZ in connection set-up
50	Coordination flags set
52	Data length greater than 128 bytes

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

PAFE Error codes

0	No error
11	Invalid address ADRE
12	Invalid interface number SSNR
16	Undefined answer received from the interface building block
17	Invalid DB no. DBNR
18	Invalid byte/word number BWNR
19	DB not available DBNR
20	DB too short (by indirect parametrizing)
21	Invalid target type ZTYP
23	Target area too short BWNR+MXBW
26	Invalid order number
31	Invalid amount of valid data at reception MXBW

NEUS As long as the bit is at the signal stand 1, the processing of the function block is discontinued.

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...2 (3)
A-NR	1...255
ZTYP	DB,EB,AB,MB,T ,Z .
DBNR	2...255
BWNR	depending on the target area
MXBW	depending on the target area (1...)

The target area must be at least as long as that given by the operands BWNR and MXBW.

ANZW	no examination
PAFE	no examination
NEUS	no examination

Example: FB21 (RECEIVE) by direct parametrizing:

```
     :SPA FB21
NAME :RECEIVE
ADRE : KH F080   The address set with the DIL switches
SSNR : KF +1     Interface number 1,2 (3)
A-NR : KF +3     Order number (1...255)
ZTYP : KC DB     Target type
DBNR : KY 0,5    DB no. by target type DB
BWNR : KY 0,20   Telegram filed from DW 20
MXBW : KF +100   Incoming telegrams may have a maximum length of 100 words
ANZW : MW 30     The display word is filed in this word
PAFE : MB 32     Error messages in relation to parametrizing are filed in this
                 byte
NEUS : M  1.0    New start bit
```

Example: FB21 (RECEIVE) by indirect parametrizing:

If the operands are to be indirectly transferred, it is possible in the following form. XX is to be parametrized on the operand ZTYP. This is the identifier for indirect parametrizing. The DB number, in which the operands lie, are to given on the operand DBNR. The data word number, from where the operands lie, is to be given on the operand BWNR. The operands ADRE, SSNR, A-NR, MXBW have in this case no meaning. The operands ANZW, NEUS and PAFE can not be indirectly parametrized. They must still be given directly to the formal operands.

```
     :SPA FB21
NAME :RECEIVE
ADRE : KH 0000   Irrelevant
SSNR : KF +0     Irrelevant
A-NR : KF +0     Irrelevant
ZTYP : KC XX     Identifier: indirect parametrizing
DBNR : KY 0,6    Parameter lies in DB 6
BWNR : KY 0,50   from DW 50
MXBW : KF +0     Irrelevant
ANZW : MW 30     The display word is filed in this word
PAFE : MB 32     Error messages in relation to parametrizing are filed in this
                 byte
NEUS : M  1.0    New start bit
```

Allocation of given DBs:

```
DB 6:

DW 50 KH ADRE
DW 51 KF SSNR
DW 52 KF A-NR
DW 53 KC ZTYP
DW 54 KY DBNR
DW 55 KY BWNR
DW 56 KF MXBW
```

4.1.8.8 FB23 (SEND), data output for 3964(R) with RK512

This data handling block is used to actively transmit a telegram to the interface building blocks BG41, BG42 und BG43 when using the procedure 3964(R) with the RK512 protocol.

Lubricating flags are used in FB23. These must be saved before loading the FB23 from a time or alarm protected OB.

The data handling block posseses formal operands.

Several working parameters are to be set through these formal operands. They can be directly as well as indirectly parametrized.

The following parameters are to be transmitted when loading FB23:

Des.	Format	Explanation
ADRE	KH	The address set with the DIL switches
SSNR	KF	Interface number 1,2 (3)
A-NR	KF	Order number (1...255)
QTYP	KC	Source type
QDBN	KY	DB no. by source type DB or DX
QBWN	KY	Byte/word number from which data is to be sent.
BWAN	KF	Amount of data to be sent
ZTYP	KC	Target type for the telegram in the PLC
ZDBN	KY	DB no. by target type DB
ZBWN	KY	Byte/word from where data is to be written
KOOR	KY	Coordination flags
ANZW	MW	The display word is filed in this word.
PAFE	MB	Error messages in relation to parametrizing are filed in this byte.
NEUS	M	New start bit

Tab. 4-18:List of parameters for the loading of FB23

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1 or 2 is to be given. By the firmware 4017, the number 3 may also be given.

A-NR Order Number, a number in the area of 1...255. This number may not be given to any other data handling block.

QTYP The source type is to be entered into its own PLC.
Possible source types are:

DB	for data blocks
DX	for data block development
EB	for the process image of inputs
AB	for the process image of outputs
MB	for flag bytes
Z	for counter words
T	for timer words

Other details result in the parametrizing error message PAFE: 27.

QDBN If the source type DB or DX are selected, the DB no. is to be given into the right byte. Valid area: 2...255. If this area is violated a PAFE: 28 is registered.
If the given DB or DX do not exist, the data handling block registers a PAFE: 29.

QBWN Byte/word number. By means of these operands it is determined, from which, e.g. data word of the given DB, or from which byte, e.g. AB, the data is to be sent.

BWAN Gives the amount of data in bytes/words, which are to be sent. At least 1 byte/word must always be sent, otherwise the data handling block registers a PAFE: 31

ZTYP The telegram's target type is to be entered into the PLC.
Possible target types (data types) are:

DB	for data block
EB	for the process image of the inputs
AB	for the process image of the outputs
MB	for flag bytes
Z	for counter words
T	for timer words

Other details result in the parametrizing error message (PAFE) 21.

ZDBN If the target type DB is selected, the DBNR is to be entered into the right byte. Valid area: 2...255. The receiver examines the parameters, if DB, DX are available with a valid DB, DX number.

ZBWN Byte/word number. By means of these operands it is determined, from which word, e.g. data word of the given DB, or from which byte, e.g. AB, a received telegram should be filed. The receiver of the telegram undertakes this examination.

KOOR The usage of coordination flags is parametrized. In a high byte the byte number of the coordination flag must be parametrized and in a low byte its bit number must be parametrized. If no coordination flags are used, a value of 255 in high and low byte is to be entered.

Access to the target area is controlled with a coordination flag: If the receiver of the data recognizes that the given coordination flag is set at 1, then the data from the data handling block SEND will not be accepted. The error 50 (target area blocked) is sent back instead.

If the receiver of the data recognizes that the area is free to write into (coordination flag at 0) he accepts the data and subsequently sets the coordination flag. The flag remains set for as long as the user program has analysed the incoming data and has reset the flag.

ANZW Display word. The display word contains one word. Status bits are filed in the right byte. An error number is filed in the left byte, if the idenfication “finished with an error“ appears in the right byte.

The status identifiers in the right byte are structured as follows:

Bit 0: Not used
Bit 1: Job running
Bit 2: Finished without an error
Bit 3: Finished with an error, → Error number in the left byte
Bit 4: Interface momentarily occupied
Bit 5-7:Not used

At the loading of the interface building block, the data handling block hands over only enough bytes, as were parametrized as block size on the data handling block SYNCHRON.

For these reasons the data handling block SEND may have to be loaded several times, in order to transmit a complete telegram to the interface building block. If the data handling block has one or more data blocks, but has not yet transmitted the last data block to the interface building block, bit 1 is placed in the right byte of the display word as identifier for “job running.“

If all data blocks have been correctly received by the interface building block and the receiver does not report an error, the data handling block SEND places the bit 2 in the right byte of the display word as identifier for “finished without an error.“

If the telegram from the interface building block has not been received, e.g. receiver does not answer, or when the receiver reports an error, the data handling block SEND places the bit 3 in the right byte of the display word as identifier for “finished with an error.“ In this case an error number is in the left byte of the display word.

Error code, delivered by ANZW:

05	Telegram header contains invalid target type
09	Receiver does not answer with DLE
14	Wrong telegram header (the first 2 bytes)
15	Transmission terminated by receiver
16	Wrong command code 2. letter (invalid target area)
20	DB error (DB0 used, DB too long, DB not available)
21	Receiver does not answer within QVZ (Timeout on SSM building block)
22	Wrong command code 1. letter (invalid source area)
23	QVZ in connection set-up
50	Coordination flag set
52	Data length greater than 128 bytes

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

PAFE Error codes

0	No error
11	Invalid address ADRE
12	Invalid interface number SSNR
16	Undefined answer received from the interface building block
17	Invalid DB no. by indirect parametrizing DBNR
18	Invalid byte/word number by indirect parametrizing BWNR
19	DB not available by indirect parametrizing DBNR
20	DB too short by indirect parametrizing
21	Invalid targat type (ZTYP)
26	Invalid order number A-NR
27	Invalid source type QTYP
28	Invalid source DB number QDBN
29	Source DB not available
31	Invalid amount of valid data at reception MXBW
32	Invalid bit number in coordination flags
33	Invalid byte number in coordination flags
40	When transmitting from a wide byte area e.g. MB to a wide word area e.g. DB, the amount of data BWAN must be even.

NEUS As long as the bit is at the signal stand is at 1, the processing of the function block is discontinued.

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...2 (3)
A-NR	1...255
QTYP	DB,DX,EB,AB,MB,T,Z
QDBN	2...255
QBWN	depending on source type (0...)
BWAN	depending on source type (1...)
ZTYP	DB,DX,EB,AB,MB,T ,Z .
ZDBN	2...255
ZBWN	depending on target area (0...)

The target area must be at least as long as that given by means of the operands BWAN and ZBWN.

ANZW	no examination
PAFE	no examination
NEUS	no examination

Example: FB23, (SEND) by direct parametrizing

```
:
     :SPA FB23
NAME :SEND
ADRE : KH F080   Address set with DIL switches
SSNR : KF +1     Interface number 1,2 (3)
A-NR : KF +4     Order number (1...255)
QTYP : KC MB     Source type
QDBN : KY 0,0    DB no. by source type DB
QBWN : KY 0,100  From Data in the source type
BWAN : KF 10     Amount of data to be sent (bytes/words depending on source
                 type)
ZTYP : KC DB     Target type
ZDBN : KY 0,5    DB no. by target type DB
ZBWN : KY 0,20   Telegram to be filed from DB 20 in target
KOOR : KY 20.3   Coordination flag in target PLC
ANZW : MW 30     The display word is filed in this word
PAFE : MB 32     Error messages in relation to parametrizing are filed in this
                 byte
NEUS : M  1.0    New start bit
```

Example: FB23, (SEND) by indirect parametrizing

If the operands are to be indirectly transferred, it is possible in the following form. XX is to be parametrized on the operand QTYP. This is the identifier for indirect parametrizing. The DB number, in which the operands lie, are to given on the operand QDBN. The DB's data word number, from where the operands lie, is to be given on the operand QBWN. The rest of the operands are in this case of no significance. The operands ANZW, NEUS and PAFE can not be indirectly parametrized. They must still be given directly to the formal operands.

```
     :SPA FB23
NAME :SEND
ADRE : KH 0000   Irrelevant
SSNR : KF +0     Irrelevant
A-NR : KF +0     Irrelevant
QTYP : KC XX     Identifier: indirect parametrizing
QDBN : KY 0,6    Parameter lies in DB 6
QBWN : KY 0,50   from DW 50
BWAN : KF +0     Irrelevant
ZTYP : KC        Irrelevant
ZDBN : KY 0,0    Irrelevant
ZBWN : KF 0,0    Irrelevant
KOOR : KY 0,0    Irrelevant
ANZW : MW 30     The display word is filed in this word
PAFE : MB 32     Error messages in relation to parametrizing are filed in this
                 byte
NEUS : M  1.0    New start bit
```

Allocation of the given DBs:

```
DB 6:

DW 50 KH ADRE
DW 51 KF SSNR
DW 52 KF A-NR
DW 53 KC QTYP
DW 54 KY QDBN
DW 55 KY QBWN
DW 56 KF BWAN
DW 57 KC ZTYP
DW 58 KY ZDBN
DW 59 KF ZBWN
DW 60 KY KOOR
```

4.1.8.9 FB25 (FETCH), data request for 3964(R) with RK512

This data handling block is used to fetch data from another PLC through the interface building blocks BG41, BG42 and BG43 when using the procedure 3964(R) with the RK512 protocol.

Source and target information is transferred when using this data handling block.

Lubricating flags are used in FB25. These must be saved before loading thr FB25 from a time or alarm protected OB.

The data handling block posseses formal operands.

Several working parameters are to be set through these formal operands. They can be directly as well as indirectly parametrized.

The following parameters are to be transmitted when loading FB 25 :

Des.	Format	Explanation
ADRE	KH	Address set with DIL switches
SSNR	KF	Interface number 1,2 (3)
A-NR	KF	Order number (1...255)
QTYP	KC	Source type
QDBN	KY	DB no. by the source type DB or DX
QBWN	KY	Byte/word number from where the data is to be sent
BWAN	KF	Amount of data to be sent
ZTYP	KC	Target type for the telegram in the PLC
ZDBN	KY	DB no. by target type DB
ZBWN	KY	Byte/word number from where the data is to be written
KOOR	KY	Coordination flag
ANZW	MW	The display word is filed in this word
PAFE	MB	Error messages in relation to parametrizing are filed in this byte
NEUS	M	New start bit

Tab. 4-19: List of parameters for the loading of FB25

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1 or 2 is to be given. By the firmware 4017 the number 3 can also be given.

A-NR Order Number, a number in the area of 1...255. This number may not be given to any other data handling block.

QTYP The source type is to be entered into its own PLC.
Possible source types are:

DB	for data blocks
DX	for data block development
EB	for the process image of inputs
AB	for the process image of outputs
MB	for flag bytes
Z	for counter words
T	for timer words

Other details result in a parametrizing error message PAFE: 27.

QDBN If the source type DB or DX are selected, the DB-No. is to be entered into the right byte. Valid area: 2...255.
The receiver examines, if DB, DX are available with a valid DB, DX number.

QBWN Byte/word number. By means of these operands it is determined, from which, e.g. data word of the given DB, or from which byte, e.g. AB, the data is to be sent.

BWAN Gives the amount of data in bytes/words, which are to be fetched. At least 1 byte/word must always be sent, otherwise the data handling block registers a PAFE: 31

ZTYP The telegram's target type is to be entered into the PLC.
Possible target types (data types) are:

DB	for data block
DX	for data block development
EB	for the process image of the inputs
AB	for the process image of the outputs
MB	for flag bytes
Z	for counter words
T	for timer words

Other details result in the parametrizing error message (PAFE) 21.

ZDBN If the target type DB is selected, the DBNR is to be entered into the right byte. Valid area: 2...255. If this area is violated, it results in PAFE: 28. If the given DB or DX do not exist, the data handling block PAFE: 29 will therefore be registered.

ZBWN Byte/word number. By means of these operands it is determined, from which word, e.g. data word of the given DB, or from which byte, e.g. AB, a received telegram should be filed.

KOOR The usage of coordination flags is parametrized. In a high byte the byte number of the coordination flag must be parametrized and in a low byte its bit number must be parametrized. If no coordination flags are used, a value of 255 in high and low byte is to be entered.

Access to the target area is controlled with a coordination flag:

If the flag is not set in the foreign PLC, this means that the data is not valid in the foreign PLC. The foreign PLC sends the error number 50 decimal back to ones own PLC. The data handling block FETCH registers with the display word:

“Finished with error“ error number: 50

ANZW Display word. The display word contains one word. Status bits are filed in the right byte. An error number is filed in the left byte, if the idenfication “finished with an error“ appears in the right byte.

The status identifiers in the right byte are structured as follows:

Bit 0: Not used
Bit 1: Job running
Bit 2: Finished without an error
Bit 3: Finished with an error, → Error number in the left byte
Bit 4: Interface momentarily occupied
Bit 5-7:Not used

At the loading of the interface building block the data handling block hands over only enough bytes, as were parametrized as block size on the data handling block SYNCHRON.

For these reasons the data handling block FETCH may have to be loaded several times, in order to receive a complete telegram from the interface building block. If the data handling block has one or more data blocks, but has not yet received the last data block from the interface building block, bit 1 is placed in the right byte of the display word as identifier for “job running.“

If all data blocks have correctly been received by the interface building block the data handling block FETCH places the bit 2 in the right byte of the display word as identifier for “finished without an error.“

If the telegram from the interface building block has not been received, e.g. receiver does not answer, or when the receiver reports an error and therefore can not provide the data, the data handling block FETCH places the bit 3 in the right byte of the display word as identifier for “finished with an error.“

Error code, delivered by ANZW:

05	Telegram header contains invalid target type
09	Receiver does not answer with DLE
14	Wrong telegram header (the first 2 bytes)
15	Transmission terminated by receiver
16	Wrong command code 2. letter (invalid target area)
20	DB error (DB0 used, DB too long, DB not available)
21	Receiver does not answer within QVZ (Timeout on SSM building block)
22	Wrong command code 1. letter (invalid source area)
23	QVZ in connection set-up
50	Coordination flag set
52	Data length greater than 128 bytes

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

PAFE Error codes

0	No error
11	Invalid address ADRE
12	Invalid interface number SSNR
16	Undefined answer received ffrom the interface building block
17	Invalid DB no. by indirect parametrizing DBNR
18	Invalid byte/word number by indirect parametrizing BWNR
19	DB not available by indirect parametrizing DBNR
20	DB too short by indirect parametrizing
21	Invalid targat type (ZTYP)
26	Invalid order number A-NR
27	Invalid source type QTYP
28	Invalid source DB number QDBN
29	Source DB not available
31	Invalid amount of valid data at reception MXBW
32	Invalid bit number in coordination flags
33	Invalid byte number in coordination flags
40	When transmitting from a wide byte area e.g. MB to a wide word area e.g. DB, the amount of data BWAN must be even.

NEUS As long as the bit is at the signal stand is at 1, the processing of the function block is discontinued.

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...2 (3)
A-NR	1...255
QTYP	DB,DX,EB,AB,MB,T,Z
QDBN	2...255
QBWN	depending on source type (0...) The source area must be at least as long as that given
BWAN	depending on source type (1...) by the operands QBWN and BWAN.
ZTYP	DB,DX,EB,AB,MB,T ,Z
ZDBN	2...255 The target area must be at least as long as that given
ZBWN	depending on the target area (0...) by the operands ZDBN and ZBWN.
ANZW	not examined
PAFE	not examined
NEUS	not examined

Example: FB25, (FETCH) by direct parametrizing

```
:
     :SPA FB25
NAME :FETCH
ADRE : KH F080   Address set with DIL switches
SSNR : KF +1     Interface number 1,2 (3)
A-NR : KF +4     Order number (1...255)
QTYP : KC MB     Source type in foreign PLC
QDBN : KY 0,0    DB no. by source type DB in foreign PLC
QBWN : KY 0,100  From data in source type in foreign PLC
BWAN : KF 10     Amount of data to be sent (Bytes/words depending on source
type)
ZTYP : KC DB     Target type in own PLC
ZDBN : KY 0,5    DB no. by target area DB in own PLC
ZBWN : KY 0,20   Telegram filed from DB 20 in target in own PLC
KOOR : KY 20.3   Coordination flag in foreign PLC
ANZW : MW 30     The display word is filed in this word
PAFE : MB 32     Error messages in relation to parametrizing are filed in this
                 byte
NEUS : M  1.0    New start bit
```

Example: FB25, (FETCH) by indirect parametrizing

If the operands are to be indirectly transferred, it is possible in the following form. XX is to be parametrized on the operand QTYP. This is the identifier for indirect parametrizing. The DB number, in which the operands lie, are to given on the operand QDBN. The DB's data word number, from where the operands lie, is to be given on the operand QBWN. The rest of the operands are in this case of no significance. The operands ANZW, NEUS and PAFE can not be indirectly parametrized. They must still be given directly to the formal operands.

```
     :SPA FB25
NAME :FETCH
ADRE : KH 0000    Irrelevant
SSNR : KF +0      Irrelevant
A-NR : KF +0      Irrelevant
QTYP : KC XX      Identifier: indirect parametrizing
QDBN : KY 0,6     Parameter lies in DB 6
QBWN : KY 0,50    from DW 50
BWAN : KF +0      Irrelevant
ZTYP : KC         Irrelevant
ZDBN : KY 0,0     Irrelevant
ZBWN : KF 0,0     Irrelevant
KOOR : KY 0,0     Irrelevant
ANZW : MW 30      The display word is filed in this word
PAFE : MB 32      Error messages in relation to parametrizing are filed in this
                  byte
NEUS : M  1.0     New start bit
```

Allocation of the given DBs:

```
DB 6:

DW 50 KH ADRE
DW 51 KF SSNR
DW 52 KF A-NR
DW 53 KC QTYP
DW 54 KY QDBN
DW 55 KY QBWN
DW 56 KF BWAN
DW 57 KC ZTYP
DW 58 KY ZDBN
DW 59 KF ZBWN
DW 60 KY KOOR
```

4.1.8.10 FB24 (RECEIVE-ALL), passive data reception for 3964(R) with RK512

This data handling block is used for importing a telegram from the interface building blocks BG41, BG42 and BG43 when using the procedure 3964(R) with a RK512 protocol.

When using this data handling block a received telegram is filed in the target area, as is described in the telegram header. The data handling block must be loaded cyclically (in OB).

The data handling block posseses formal operands. Several working parameters are to be set through these formal operands. They can only be directly parametrized.

The following parameters are to be transmitted when loading FB24 :

Des.	Format	Explanation
ADRE	KH	Address set with DIL switches
SSNR	KF	Interface number 1,2 (3)
ANZW	MW	The display word is filed in this word
PAFE	MB	Error messages in relation to parametrizing is filed in this byte
NEUS	M	New start bit

Tab. 4-20:List of parameteres for the loading of FB24

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1 or 2 is to be given. By the firmware 4017 the number 3 can also be given.

ANZW Display Word. The display word occupies a word. In the right byte status bits are filed. In the left byte a wrong number is filed, when the identifier “finished with error“ appears in the right byte.

The status identifiers in the right byte are structured as follows:

Bit 0: Not used
Bit 1: Job is running
Bit 2: Finished without an error
Bit 3: Finished with an error, → Error number in the left byte
Bit 4: Interface momentarily occupied
Bit 5-7:Not used

At the polling of the interface building block the data handling block imports only as many bytes, as were parametrized as block size on the data handling block SYNCHRON.

Due to these reasons the data handling block RECEIVE-ALL must often be loaded several times, in order to import a complete telegram from the interface building block. If the data handling block has one or more data blocks but has not yet imported the last data block from the interface building block, it places as an identifier for “job running“ bit 1 in the right byte of the display word.

If all data blocks have been correctly imported from the interface building block, the data handling block places as identifcation for "finished without an error" the bit 2 in the right byte of the display word.

If the telegram could not be imported from the interface building block, e.g. the telegram is too long, the data handling block places as identifier for "finished with error" the bit 3 in the right byte of the display word. In this case an error number is then in the left byte of the display word.

Error code, delivered by ANZW:

05	Telegram header contains invalid target type
09	Receiver does not answer with DLE
14	Wrong telegram header (the first 2 bytes)
15	Transmission terminated by receiver
16	Wrong command code 2. letter (invalid target area)
20	DB error (DB0 used, DB too long, DB not available)
21	Receiver does not answer within QVZ (Timeout on SSM building block)
22	Wrong command code 1. letter (invalid source area)
23	QVZ in connection set-up
50	Coordination flag set
52	Data length greater than 128 bytes

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

PAFE Error codes

0	No Error
11	Invalid address ADRE
12	Invalid interface number SSNR
16	Undefined answer received from the interface building block

NEUS As long as the bit is at the signal stand is at 1, the processing of the function block is discontinued.

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...2 (3)
ANZW	Not examined
PAFE	Not examined
NEUS	Not examined

Example: FB24 (RECEIVE-ALL):

```
     :SPA FB24
NAME :RECEIVE-ALL
ADRE : KH F080   Address set with DIL switches
SSNR : KF +1     Interface number 1,2 (3)
ANZW : MW 30     The display word is filed in this word
PAFE : MB 32     Error messages in relation to parametrizing are filed in this
                 byte
NEUS : M  1.0    New start bit
```

4.1.8.11 FB22 (SEND-ALL), requested data output for 3964(R) with RK512

This data handling block is used for transmitting telegrams to the interface building blocks BG41, BG42 and BG43, when using the procedure 3964(R) with the RK512 protocol.

The building block transfers the requested data in a telegram on the interface building blocks. The area, in which the data to be sent is, is described in the request telegram. A request can be triggered by the data handling block FETCH.

The data handling block SEND-ALL must be cyclically polled (in OB1).

The data handling block posseses formal operands. Several working parameters are to be set through these formal operands. They can only be directly parametrized.

The following parameters are to be transmitted with the loading of FB22 :

Des.	Format	Explanation
ADRE	KH	Address set with DIL switches
SSNR	KF	Interface number 1,2 (3)
PAFE	MB	Error messages in relation to parametrizing are filed in this byte.
NEUS	M	New start bit

Tab. 4-21:List of parameters for the loading of FB22

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

SSNR The interface number 1 or 2 is to be given. By the firmware 4017 the number 3 can also be given.

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

PAFE Error codes

0	No error
11	Invalid address ADRE
12	Invalid interface number SSNR
16	Undefined answer received by interface building block

NEUS As long as the bit is at the signal stand is at 1, the processing of the function block is discontinued.

Coordination flag:

The protocol 3964(R) supports the usage of a coordination flag. This flag is used for blocking access to the source area or to allow the reading of the source area. The coordination flag is given by the sender of the request telegram in the telegram header. On receiving a new request telegram the data handling block SEND-ALL examines, if the parametrized flag in the telegram header is set. If this is the case, reading access is denied. The sender receives a response telegram with the information, that the area for the job was blocked.

If the given flag is at 0, the requested data is sent. If it is at 1, the reading access is denied.

If, for some reason, the data can not be sent to the receiver, he receives a response telegram with a corresponding error number. The user is supplied with this error number in the display word.

Error codes of the response telegrams:

12	Invalid bit number in the coordination flags given (allowed 0...7) or invalid byte number in the coordination flags given (allowed 0...199)
16	Invalid area type, from where the data is requested (allowed DB,DX,AB,EB,MB,T,Z)
20	Invalid area, from where the data is to be read (i.e. e.g. Block is not available or DB too short or access at MD255)
50	Area is blocked by coordination flag (Flag = 1)

The data handling block examines the parametrized values at the following limits:

ADRE	F080...F1FC
SSNR	1...2 (3)
PAFE	not examined
NEUS	not examined

Example: FB22, (SEND-ALL)

```
     :
     :SPA FB22
NAME :SEND-ALL
ADRE : KH F080    Address set with DIL switches
SSNR : KF +1      Interface number 1,2 (3)
PAFE : MB 32      Error messages in relation to parametrizing are filed in this
                  byte
NEUS : M  1.0     New start bit
```

4.1.8.12 FB28 (FREIGABE), release of building block

This data handling block resets the new start bit and releases the building block for SPS access. The block is to be loaded at the end of the OB1.

The new start bit is set by the user in the OB22, if an automatic new start identifier is requested. This bit is to be given by the individual data handling blocks as the identifier "NEUS".

The data handling block posseses formal operands. The working parameters are to be set through these formal operands.

The following parameters are to be transmitted when loading FB28 :

Des.	Format	Explanation
ADRE	KH	Address set with DIL switches
NEUS	BI	New start bit
PAFE	BY	Error messages in relation to parametrizing are filed in this byte.

Tab. 4-22: List of parameters for the loading of FB28

The following details are necessary for the abovementioned parameters:

ADRE Input of the physical start address, under which the building block can be accessed by the PLC. This start address can be set with the DIL switches S1 and S2 (see chapter 5).

NEUS Release of building block, when NEUS is set at 1.

PAFE Contains a byte for issuing an error number. PAFE provides the value 0, the parameter details are therefore correct. A value unequal to 0 represents an error.

PAFE Error codes

0 No error

11 Invalid address ADRE. An address smaller than F080h was entered.

The data handling block examines the parametrized values at the following limits:

ADRE F080...F1FC

NEUS not examined

PAFE not examined

4.2 Application of interfaces without data handling blocks

4.2.1 Address allocation

The building blocks occupy four (BG41) or eight (BG42, BG43) successive addresses in the PLC's address area. The setting of the base address, under which the building block should respond, is described in chapter 5.1.

Each of the building block's channels occupies 2 addresses, the data and the status register, whereby channel 0 as the parameter channel is of particular significance.

Commands are transferred through channel 0, which modify the building block's internal parameters, e.g.:

- Set clock,
- Select procedure
- Select baud rate
- Set data format

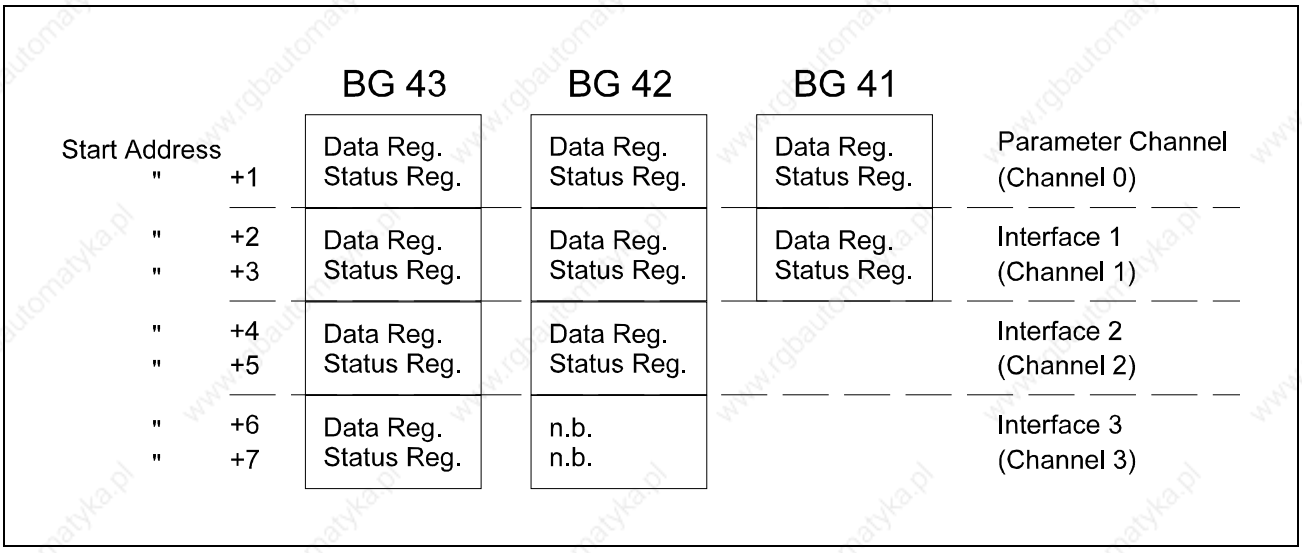


Fig. 4-13: Data and status register of the BG41/2/3

Data/Status register

Each register has a register, through which data is exchanged bidirectionally between the PLC and the building block, and a register, which communicates the status of the channel to the PLC. This status register can only be read by the PLC.

4.2.2 Interface channel

Status register interface channel

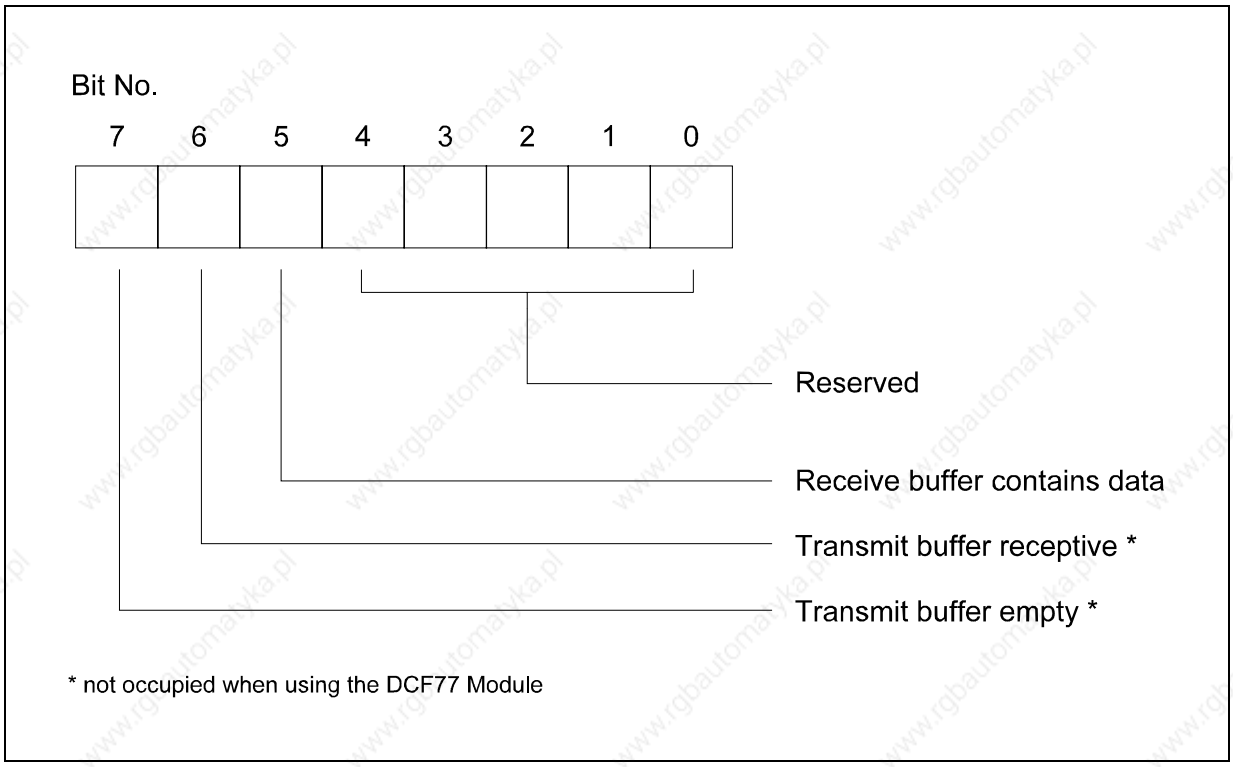


Fig. 4-14: status register interface channel

If the bits to be evaluated are set, the interfaces have the following status:

BIT 7 = 1:
indicates, that the transmission buffer of the respective channel is empty i.e. without further status query, 256 bytes can be written in succession into the data channel. Once a byte has been entered, the repeated access from the PLC to data channel is immediately possible. However, the building block only acknowledges this access after 20 µs with READY. This time is required in order to process the data in the building block. After that, all data in the transmission buffer is automatically sent from the corresponding interface.

BIT 6 = 1:
indicates that the transmission buffer is ready. The bit should be queried before a single access. If the transmission buffer is not ready, the building block does not acknowledge anything and it results in an acknowledgement delay error (QVZ) in the PLC's program.

*Due to temporal reasons a **buffer overflow can not be intercepted** by setting this bit. Control of the buffer allocation is carried by either the standard software or by the user software.*

BIT 5 = 1:

indicates, that the building block has received data, which is to be found in the reception buffer. If the corresponding channel also functions as the receiver, this bit should be cyclically polled, so that the data can be read where necessary. Otherwise an overflow can occur in the reception buffer.

If bit 5 = 0, reading access is not allowed, as in this case the building block does not acknowledge this. It also takes 20 µs before another character is made available in the reception channel.

Data register interface channel

If a channel is used for the operation of a DCF77 antenna module, information on the DCF77 system can be read on a continuous basis.

The bits of the corresponding periphery bytes have the following significance:

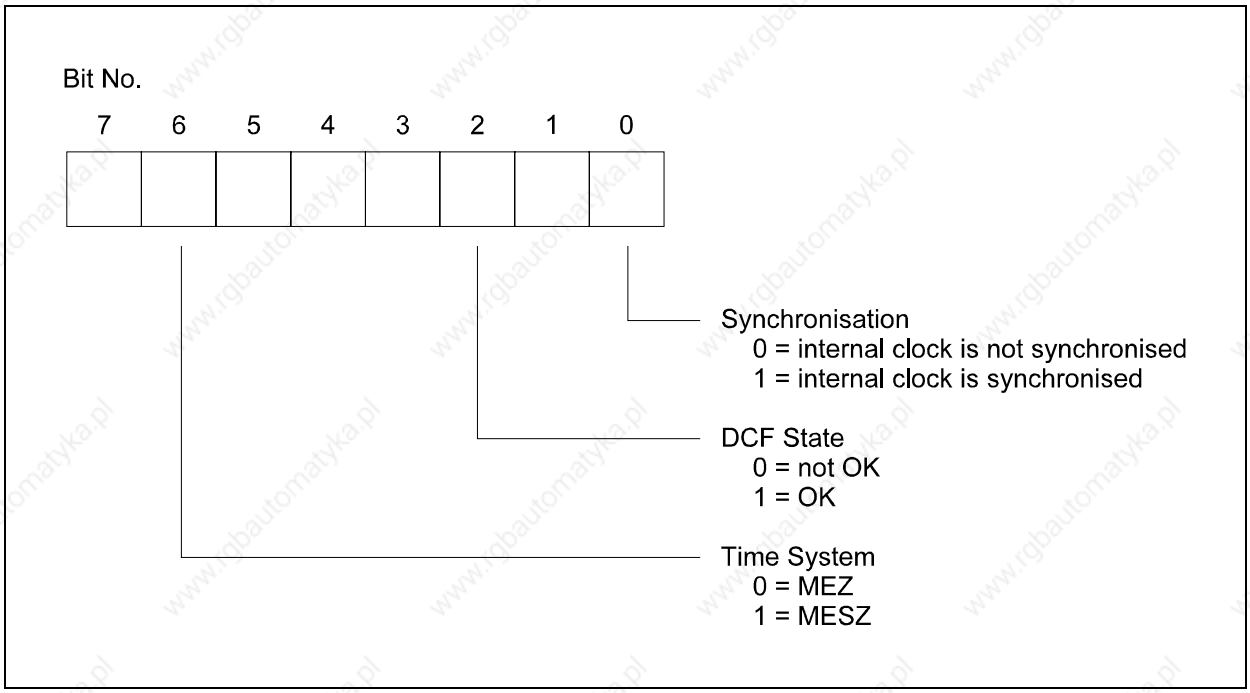


Fig. 4-15: Data register interface channel

4.2.3 Parameter channel

Status register parameter channel

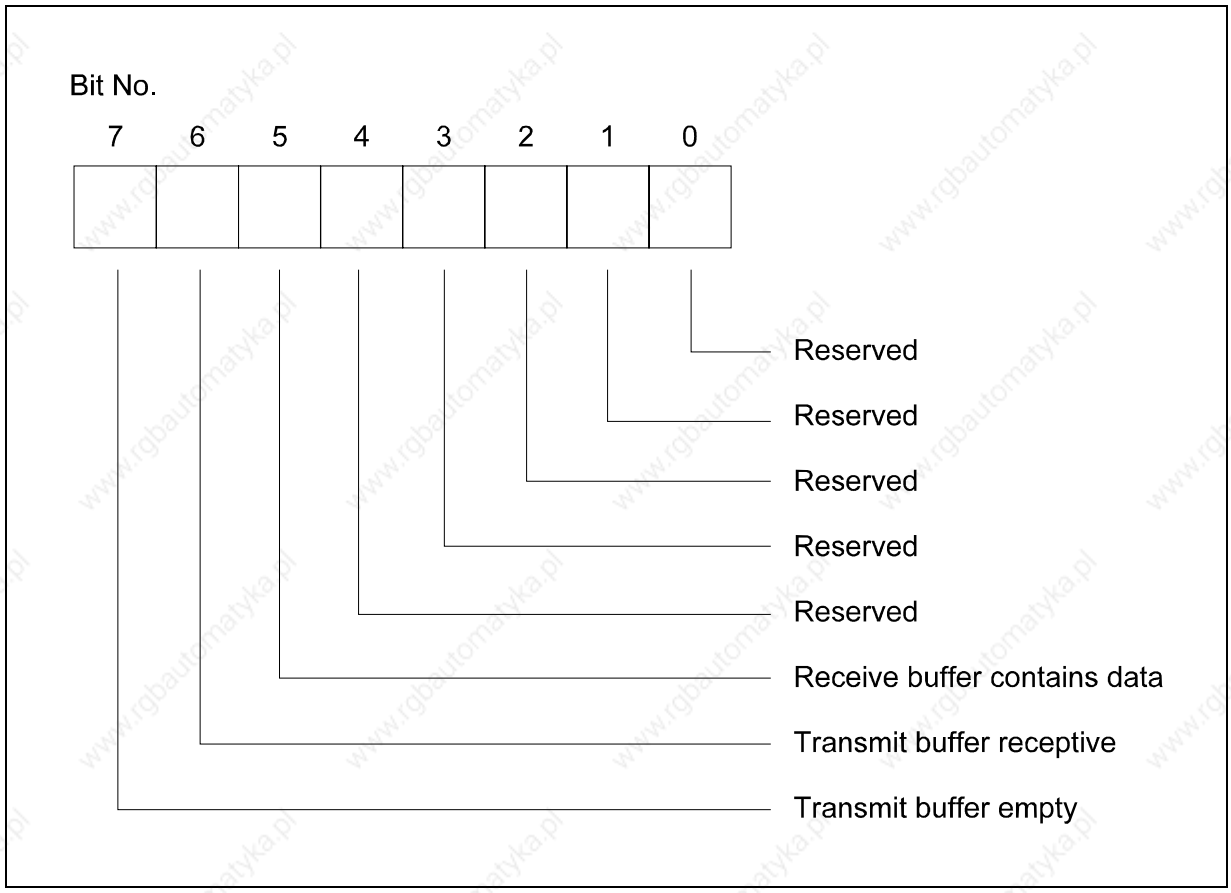


Fig. 4-16: Status register parameter channel

If the bits to be evaluated are set, the interface has the following status:

BIT 7 = 1 as status interface channel
BIT 6 = 1 as status interface channel
BIT 5 = 1 as status interface channel

4.2.4 Acknowledgement and error code

Acknowledgement

The commands 4 and 5 are confirmed by an acknowledgement, which must be received over the data register of channel 0. The acknowledgement consists of two bytes. In the first byte the command and the channel number are repeated.

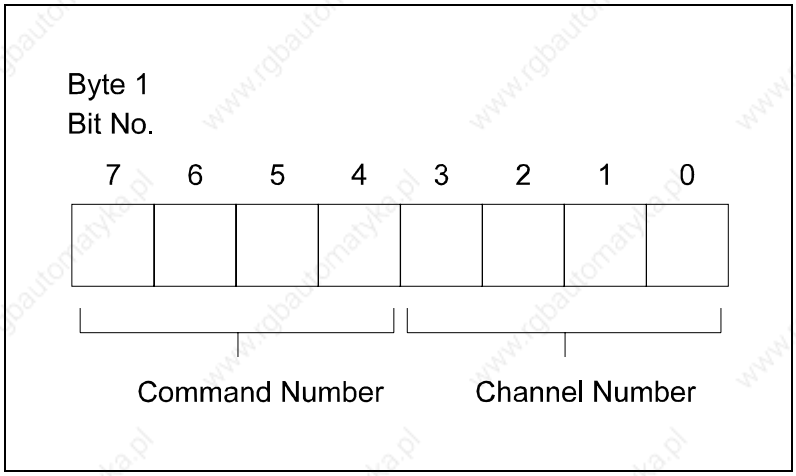


Fig. 4-17: Acknowledgement byte 1

Error code

The second byte contains an error code, which enables an exact error analysis. If the sescond byte is 0, no error has arisen.

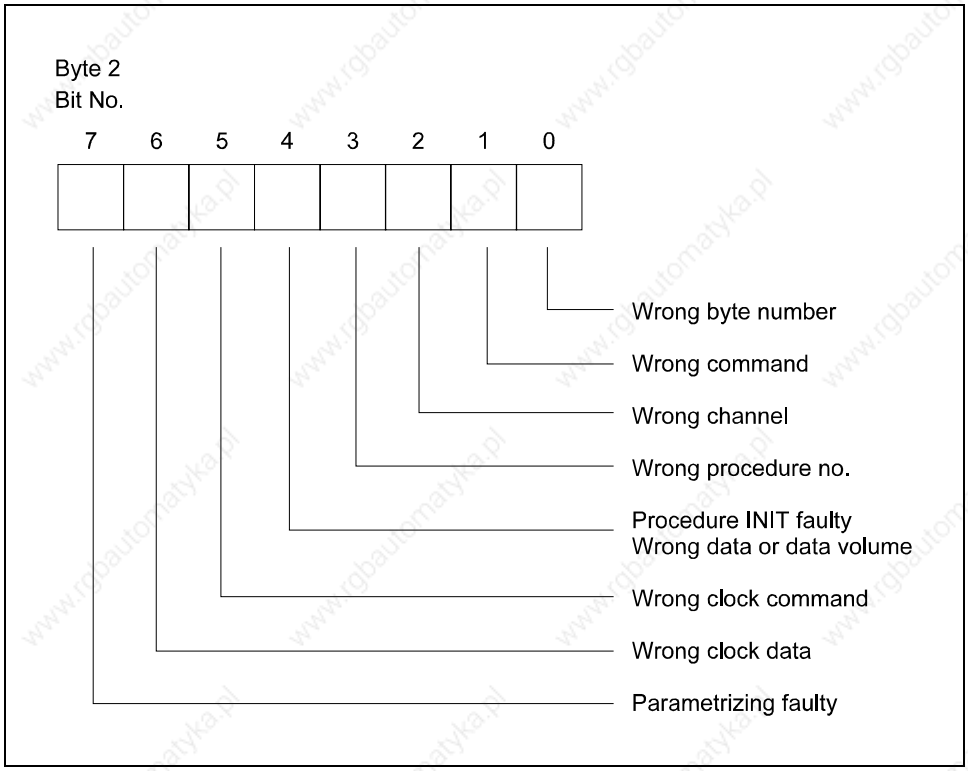


Fig. 4-18: Acknowledgement byte 2

4.2.5 PLC interface parameter channel

By means of the parameter channel (channel 0), the building block can be given different commands, which are significant for the input/output channel.

With software it is possible to

- reset or re-parametrize the building block
- read or set the date/time
- determine the protocol for the interface
- set the baud rate and the data format for the serial interface module
- issue the firmware's version number.

The commands are given to the command channel in the form of one or multiple instructions. The first byte is the so-called command byte and contains the number of the command in the left (high) nibble and the number of the channel for which the command is for in the right (low) nibble.

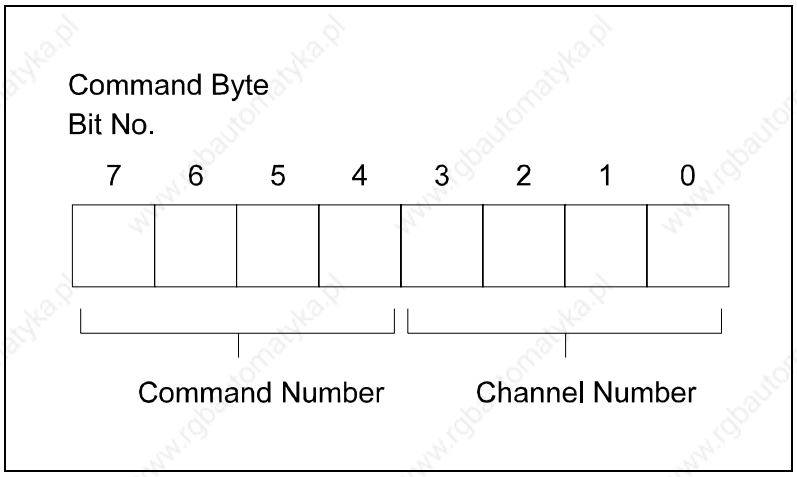


Fig. 4-19: Command byte parameter channel

The following commands are available:

0	Reset the entire building block
1	Reset the transmission and reception buffer for one channel
2	Change the baud rate for one channel
3	Change the data format for one channel
4	Operate the clock
5	Select a protocol

4.2.5.1 Reset the entire building block

Command 0

This byte is a one-byte command and does not require a channel number, as it is effective for the whole building block. It is automatically triggered by a new start in PLC, when the building block is addressed in the periphery area.

This command sets the building block to a state, which corresponds to the new start. All characters are transferred without a protocol, i.e. as they come from the control or periphery device. In addition to this, all buffers are deleted and the following values are set:

- Baud rate 9600 baud
- Even parity
- Start bit
- 8 data bits
- 2 stop bits

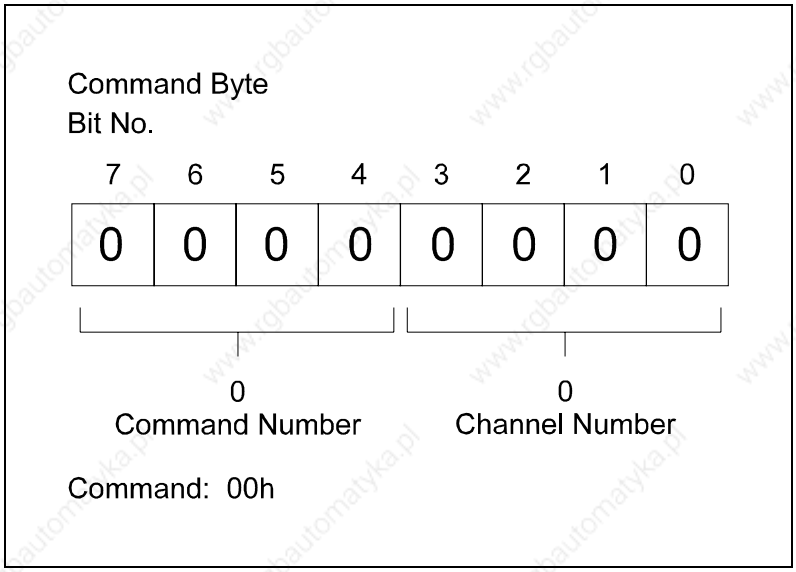


Fig. 4-20: Command byte command 0

4.2.5.2 Resetting the transmission and reception buffers for the interface channel

Command 1

This command deals with a one-byte command. The channel number, whose transmission and reception buffers are to be deleted, is to be entered into the right nibble.

Example:

The transmission and reception buffer of channel 1 are to be deleted.

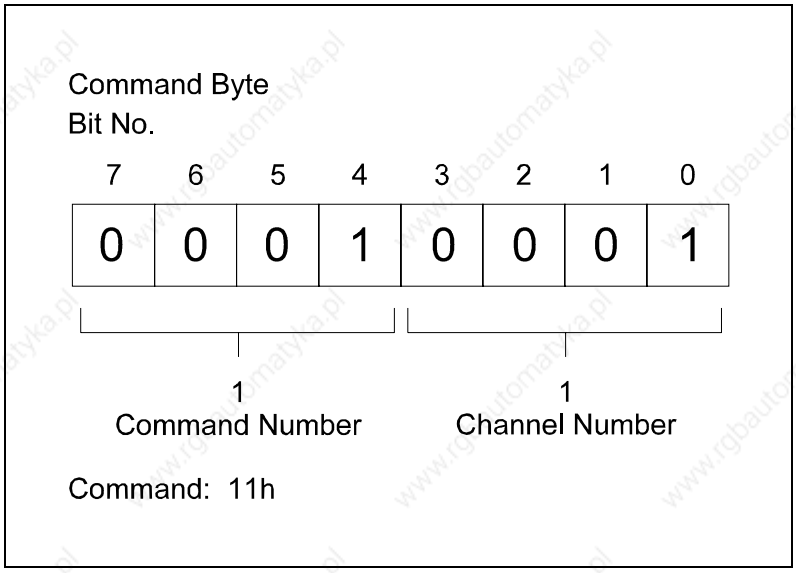


Fig. 4-21: Command byte command 1

4.2.5.3 Change the channel's baud rate

Command 2

This command is a two-byte command and requires the transfer of the code for the baud rate into the second byte.

> ! *If a baud rate of 38400 baud is used, the plug connector J11 on the base circuit board is to be placed in position 38.4.*

Assignment list (code/baud rate)

Code (decimal)	Plug Connector J11 in Position 38400 / 19200 J11	 38400 / 19200 J11
01	75	150
02	100	(200)
03	110	(220)
04	150	300
05	300	600
06	600	1200
07	1200	2400
08	2400	4800
09	4800	9600
10	9600	19200
11	19200	38400

Fig. 4-22: Code/baud rate

Example 1:

Channel 1 should be set to 4800 baud.

J11 is plugged into position 19200

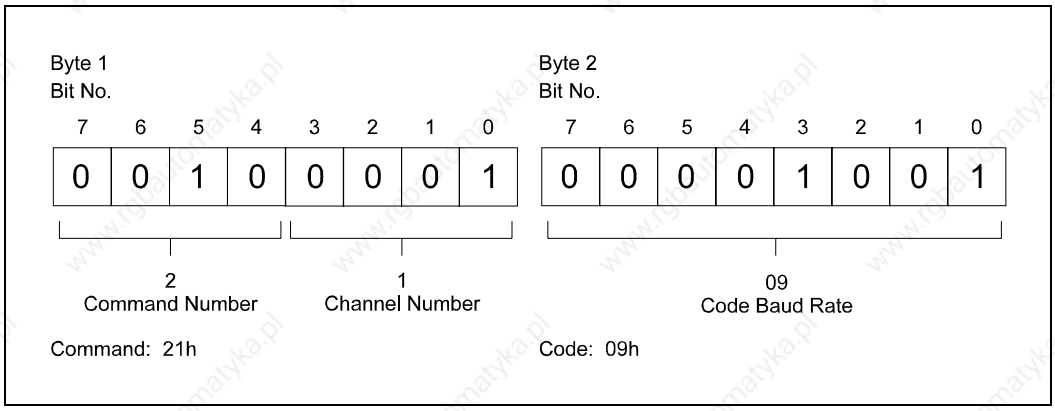


Fig. 4-23: Example 1, changing the baud rate

Example 2:

Channel 2 should be set to 300 baud.

J11 is plugged into position 38400

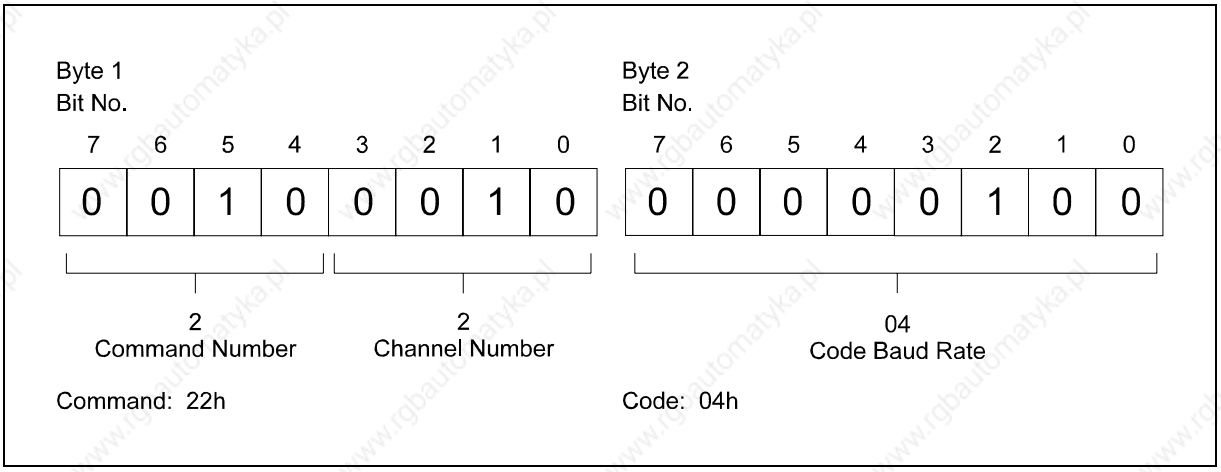


Fig. 4-24: Example 2, changing the baud rate

4.2.5.4 Changing the channel's data format

Command 3

This command is a two-byte command and requires the mode, which is to be used, to be entered into the second byte.

This byte is structured as follows:

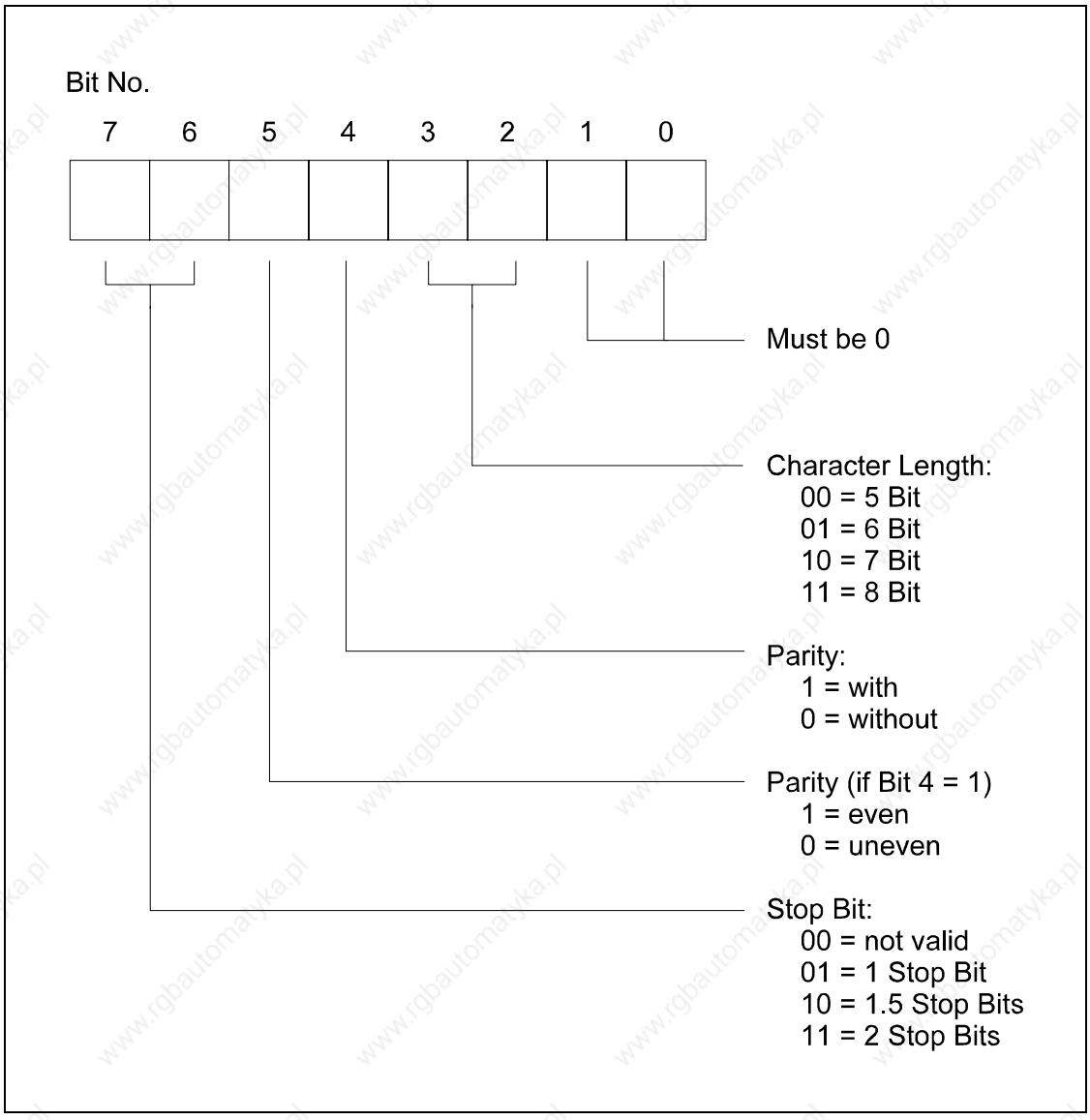


Fig. 4-25: Changing the data format with command 3

Example:

Channel 1 should be set with the following parameters:

- 7 Data bits
- 1 Stop bits
- Uneven priority

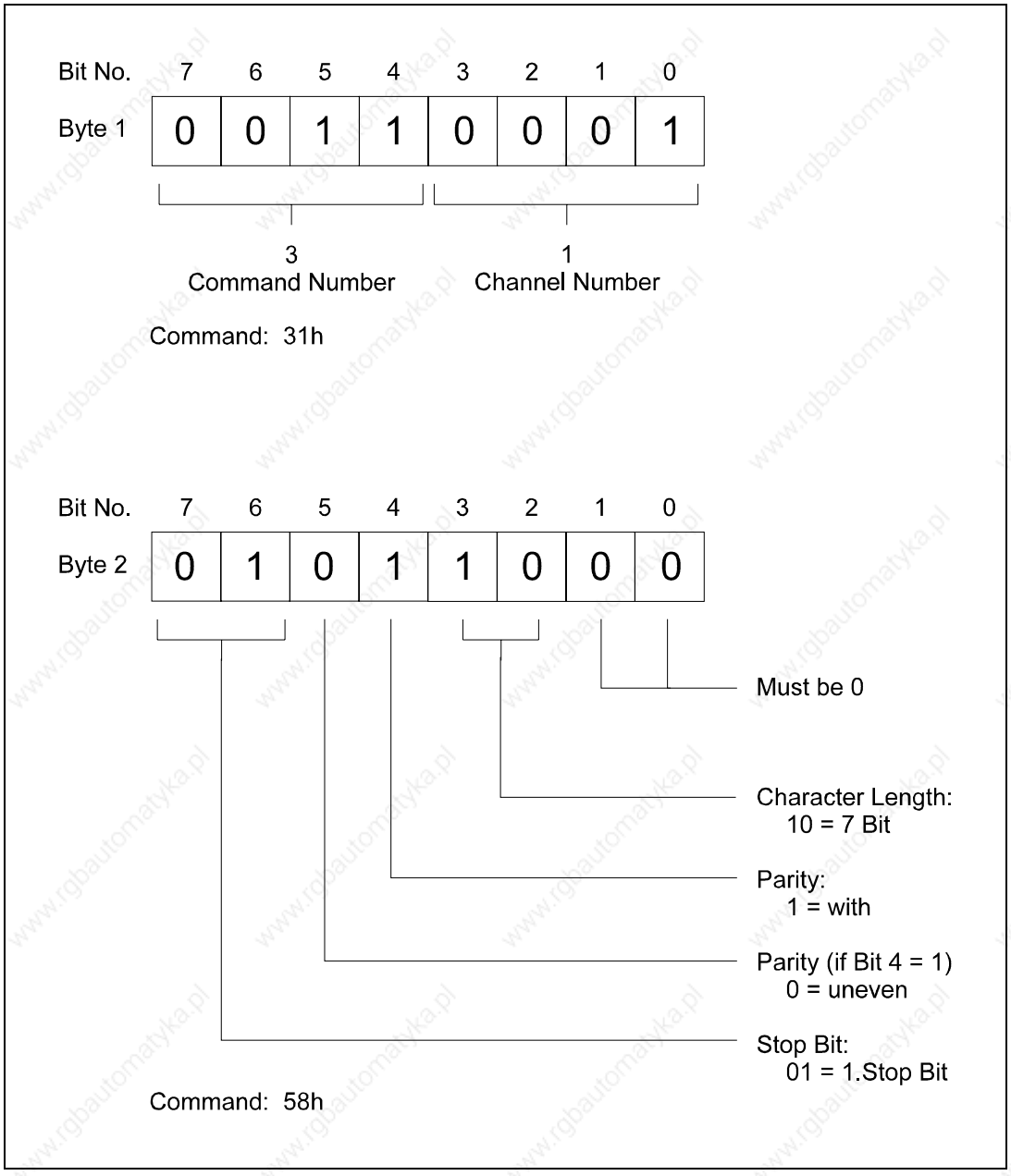


Fig. 4-26: Example of how to change the data format of a channel

4.2.5.5 Operating the clock

Command 4

With this command it is necessary to enter the channel number 0, as it applies for the entire building block and is not assigned to any interface.

It is a command, which can contain 2 to 22 Bytes. The second byte further specifies the command 4 by means of the instructions 01 to 05.

Instruction 01: Read date and time in a packed BCD format from PLC (7 bytes, after 2 acknowledgement bytes)

Date and time are received through the data channel 0 in the following order:

- first acknowledgement byte
- second acknowledgement byte
- seconds
- minutes
- hours
- day
- month
- year
- weekday

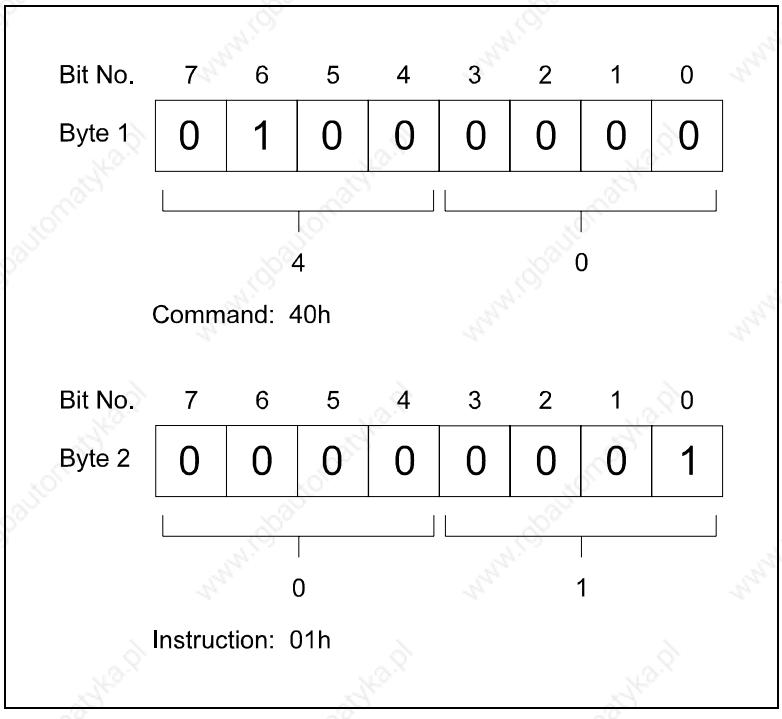


Fig. 4-27: Operating the clock, command 4, instruction 01

Instruction 02: Set date and time

Instruction 02 is not available when using the DCF77 module.

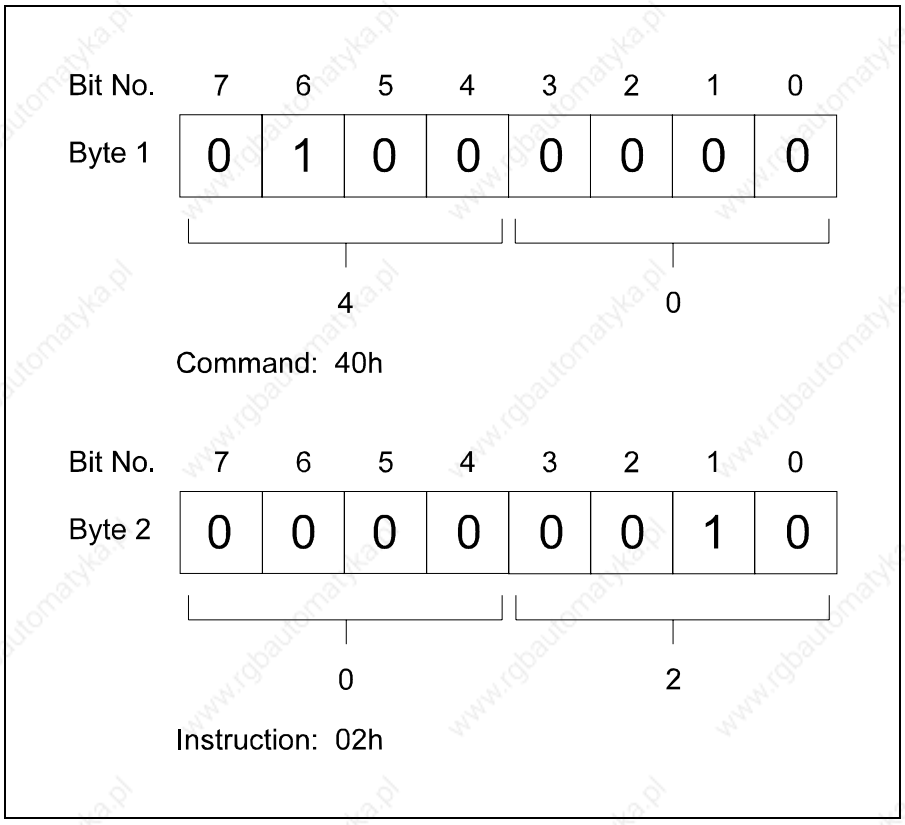


Fig. 4-28: Operating the clock, command 4, instruction 02

After instruction 02 (set time and date) further information is required, which is to be entered in 7 successive bytes. The following order is to be maintained:

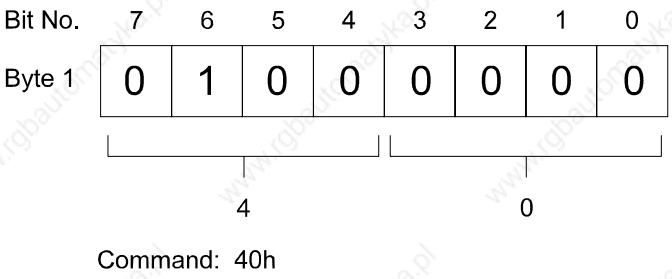
Byte 3:	Seconds	Value Range 00 ... 59
Byte 4:	Minutes	Value Range 00 ... 59
Byte 5:	Hours	Value Range 00 ... 23
Byte 6:	Day	Value Range 01 ... 31
Byte 7:	Month	Value Range 01 ... 12
Byte 8:	Year	Value Range 00 ... 99
Byte 9:	Weekday	Value Range 00 ... 06 (00=Sunday)

Two acknowledgement bytes are returned.

Example :

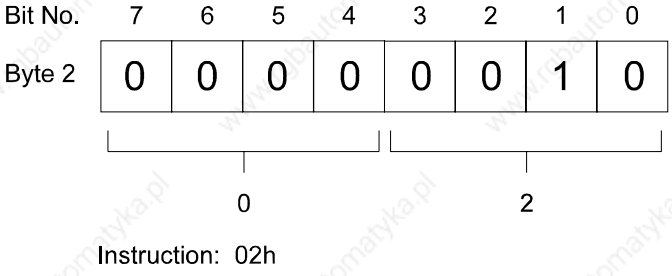
To be set: Tuesday 06.06.95 14:53:00

Command:
Operating the clock



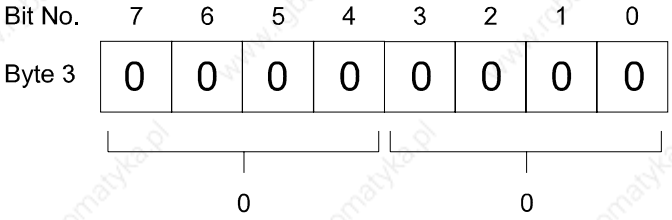
Command: 40h

Instruction:
Set clock

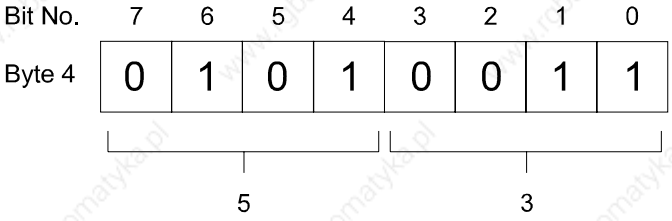


Instruction: 02h

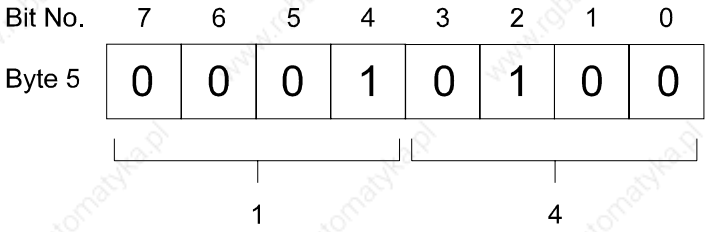
Seconds 00h



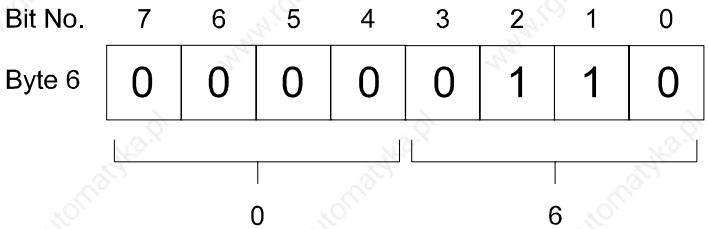
Minutes 53h



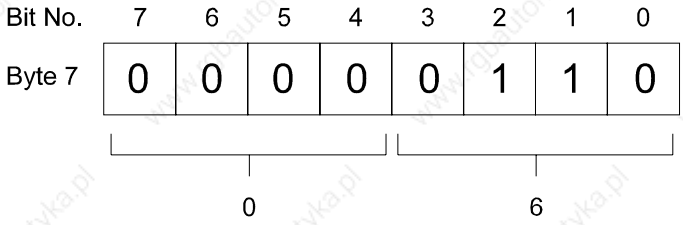
Hours 14h



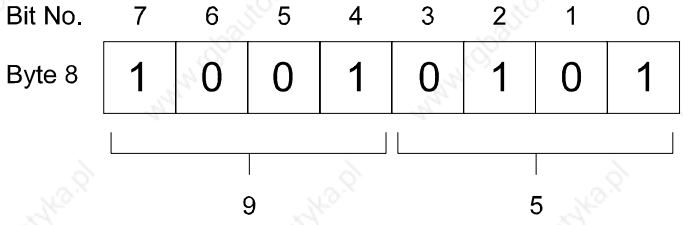
Day 06h



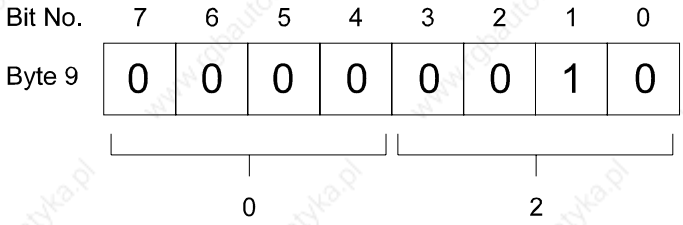
Month 06h



Year 95h



Weekday
Tuesday 02h



Instruction 03: Put clock 1 hour forward (Changing from winter to summer time)

Instruction 03 is not available when using a DCF77 module.
Two acknowledgement bytes are returned.

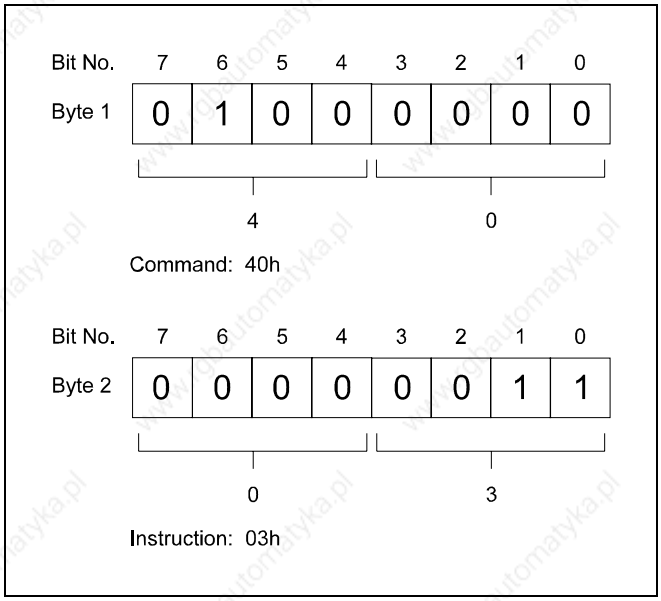


Fig. 4-29: Operating the clock, command 4, instruction 03

Instruction 04: Put clock back 1 hour.
(Changing from summer to winter time)

Instruction 04 is not available when using a DCF77 module.
Two acknowledgement bytes are returned.

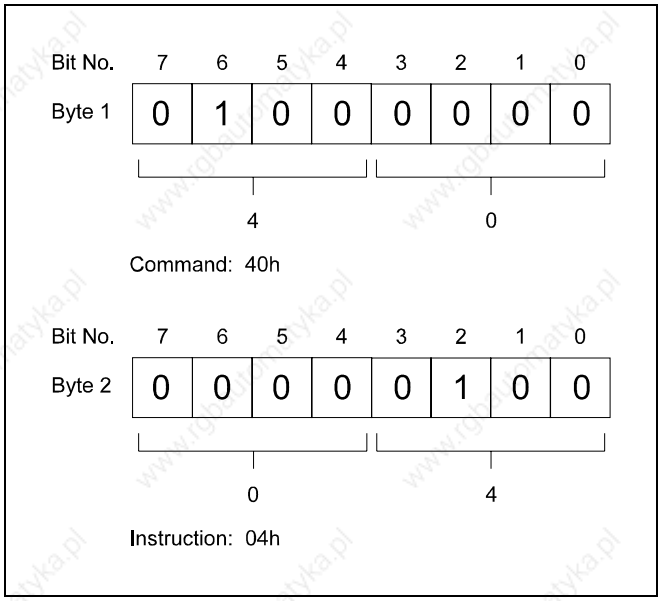


Fig. 4-30: Operating the clock, command 4, instuction 04

Instruction 05: Date and time are to be transferred to the PLC as ASCII code.
(20 bytes, after 2 acknowledgement bytes)

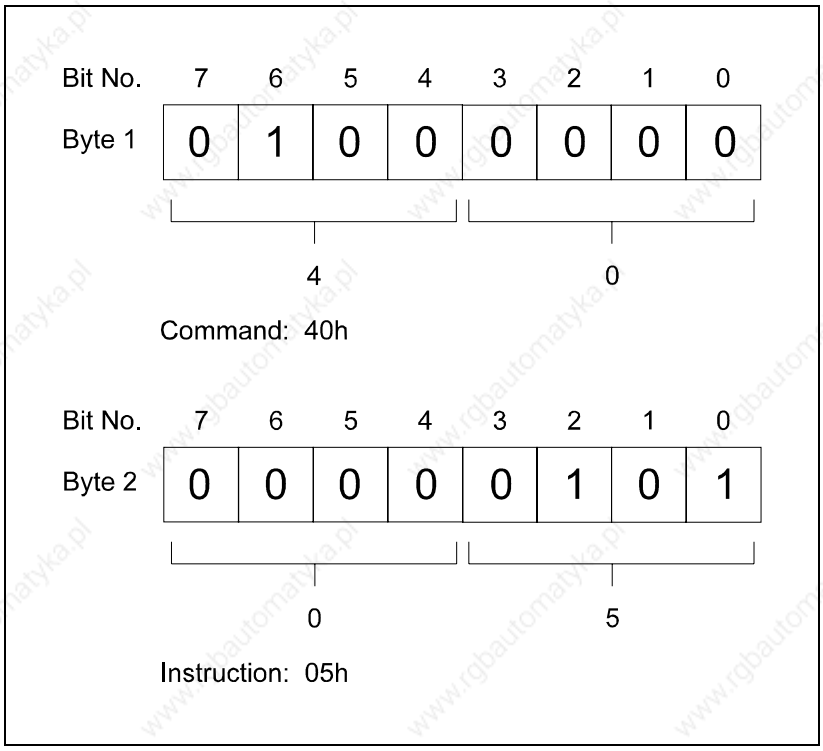


Fig. 4-31: Operating of the clock, command 4, instruction 05

After the two acknowledgement bytes, the following character string, for example, is transferred to the PLC:

Tuesday 06.06.1995 14:53:00

The character string is received through the data channel of channel 0. Including the two acknowledgement bytes, 22 bytes altogether are transferred.

4.2.5.6 Selection of function (protocols and procedures)

Command 5

This command is a two-byte command. The first byte contains the command and the channel number, for which the function is valid. The function is selected with the second byte with the aid of a code.

The following code is to be entered:

Code 00:	without a protocol/procedure
Code 01:	STX/ETX protocol
Code 02:	*Signal function (only for signalling module)*
Code 03:	Procedure 3964 with RK512
Code 04:	Procedure 3964R with RK512
Code 05:	Counter (only possible with a counter module)
Code 06:	Clock generators (only possible with clock generators)
Code 07:	DCF77 (only possible with a DCF77 antenna module)
Code 08	Procedure 3964
Code 09	Procedure 3964R
Code 10	Analogue input /analogue output
Code 12:	SSI module (only possible with an SSI module)

An acknowledgement is given after every two-byte command.

Example: Procedure 3964R with RK512 is to be operated through channel 1:

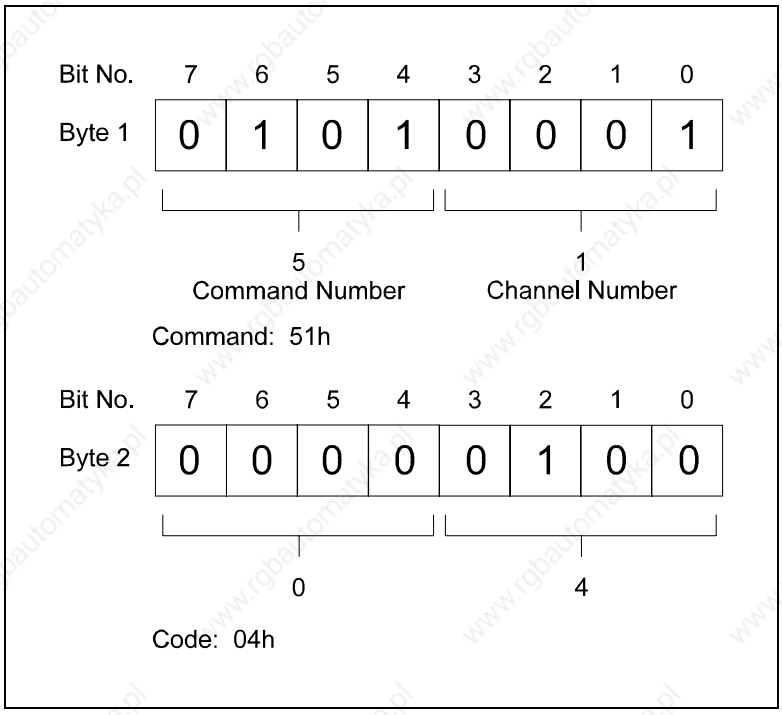


Fig. 4-32: Function selection (protocols and procedures) with command 5, example 1

Two acknowledgement bytes are returned.

Example 2: Channel 2 should be operated without a protocol/procedure.

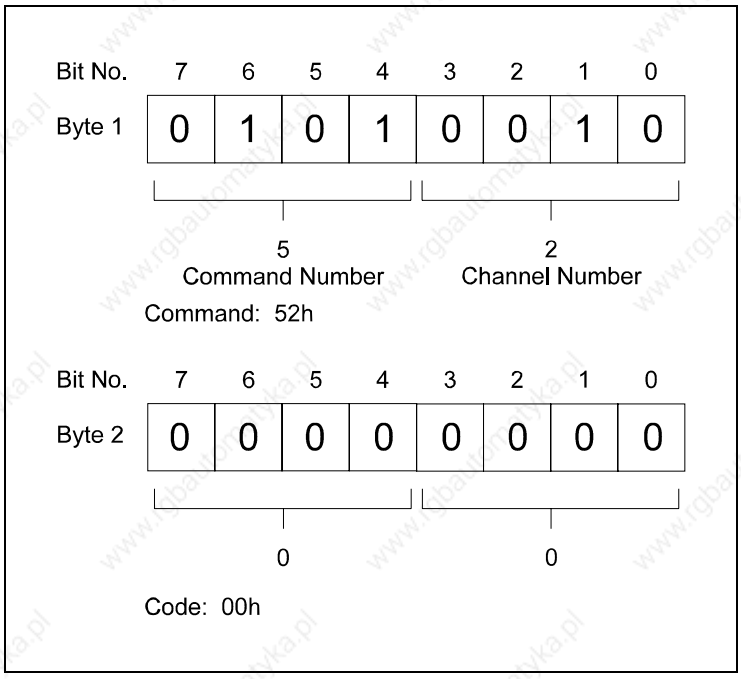


Fig. 4-33: Function selection (protocols and procedures) with command 5, example 2

Two acknowledgement bytes are returned.

Example 3: Channel 3 should be operated with an STX/ETX protocol.

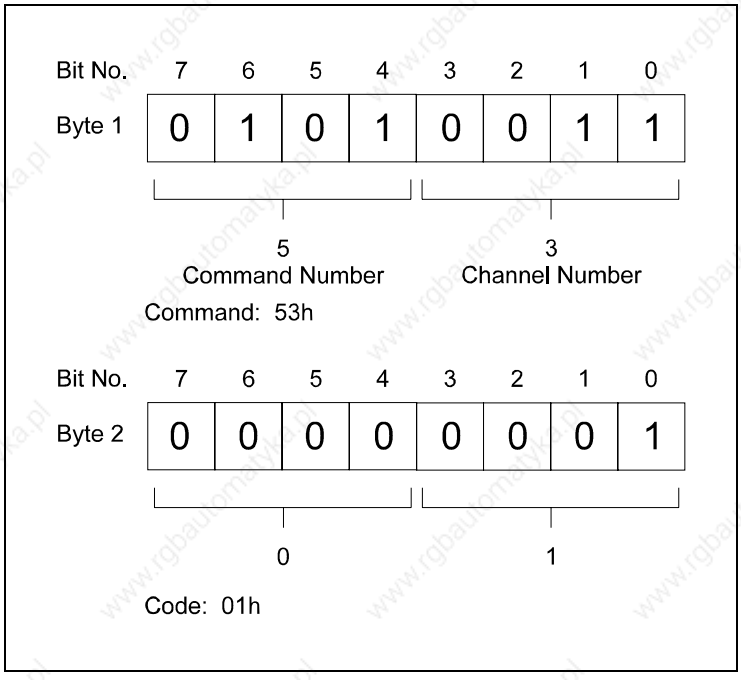


Fig. 4-34: Function selection (protocols and procedures) with command 5, example3

Two acknowledgement bytes are returned.

4.2.5.7 Parameter description of STX/ETX

Command 8

Start and end characters and the delay time of the STX/ETX protocol can be altered with this command.

Standard defaults:

STX	02h
ETX1	03h
ETX2	00h
ZVZ	00h
Mode	00h

This command effects the protocol type STX/ETX (Code 01), which must be activated by command 5. A re-parametrizing during operation is possible.

The command consists of 6 bytes altogether, which are to be sent through the parameter channel 0. The adjustment is to be carried out separately for each channel.

The ASCII character NUL (00h) is not allowed for the start or end identifier, as this has a special function. If the parameter ETX2 is loaded with the value 00h, only an end character is evaluated.

ZVZ (Character Delay Time)

The character delay time determines which maximum delay is permitted between the protocol's individual characters. The parameter is given in time units to 50 ms. A delay time of 50 ms to 12,75 is therefore adjustable.

Mode

The parameter "Mode" determines the method of the input buffer. With the value 00h, a FIFO with a depth of 2 telegrams is established. The value 01h establishes a FIFO with a depth of n-telegrams.

As only valid telegrams are transferred to the SPS, these must be saved in between. The telegram has a maximum length of 255 bytes.

The telegrams on the SPS must be terminated with the ASCII character NUL (00h). All incoming telegrams, concluded with the ASCII character NUL are transferred to the SPS.

Example

The following parameters are to be entered via channel 2:
STX character: 08h, first ETX character: 07h, second ETX character: 06h, ZVZ: 100ms,
Mode: FIFO with a depth of 2 telegrams.

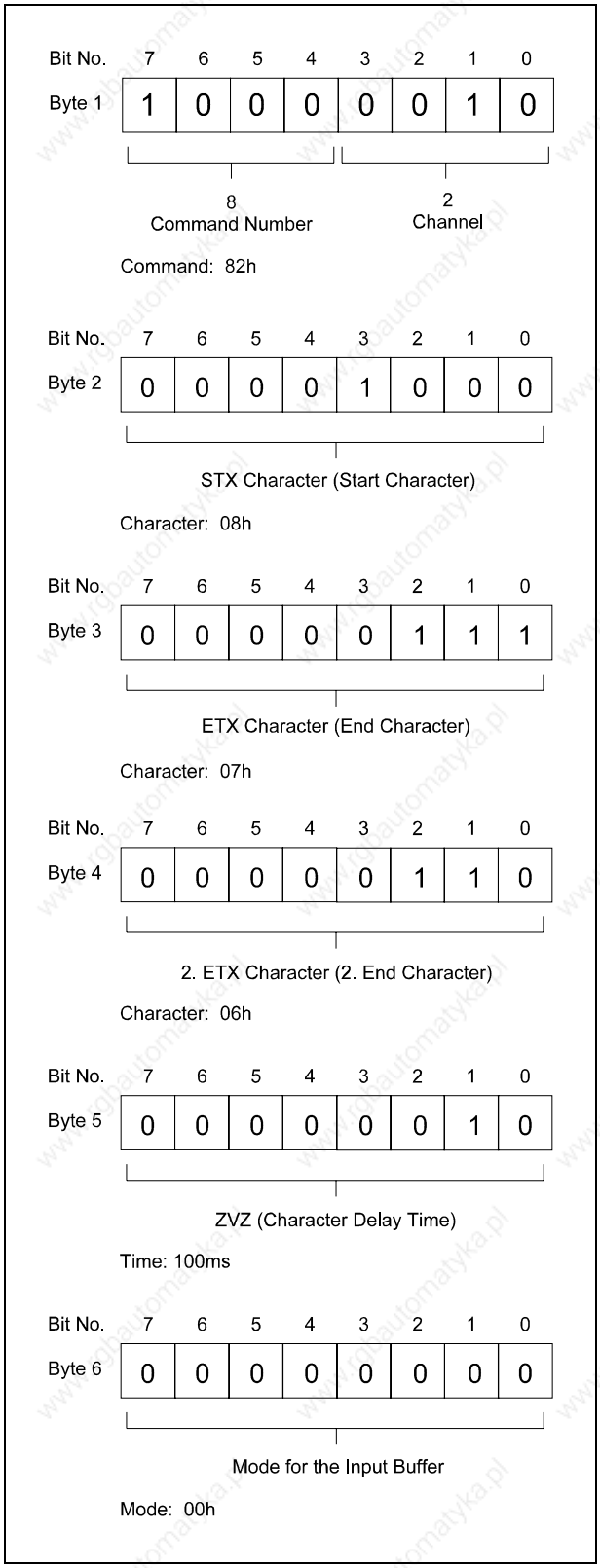


Fig. 4-35: Example of STX/ETX parametriezed with command 8

5 Startup

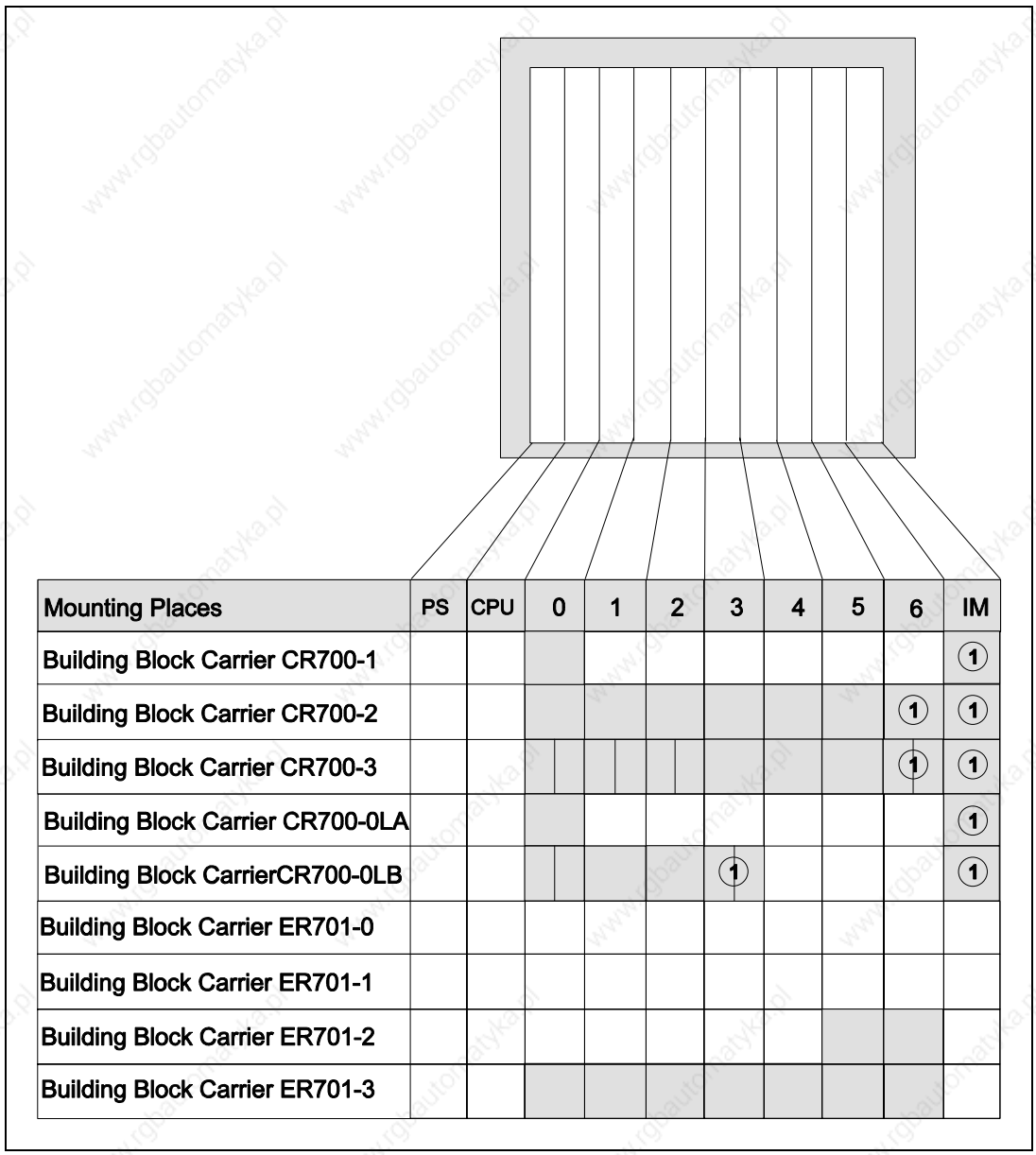
5 Startup

5.1 Startup procedure

- Connect the corresponding module to the building block.
- If a 24V supply is necessary on the modules, equip them with the respective jumpers (see chapter 5.2.9).
- Address the building block with the respective DIL switches (see chapter 5.2.7).
- Switch off the current supply at the PLC and plug the building block according to the instructions into your PLC (see chapter 5.2).
- Connect the building block's 25-pole SubD socket with the corresponding peripheral.
- Switch the voltage supply at the PLC on again.
- Observe the reaction of the possibly available LEDs on the building block. See chapter 3.1.2.

5.2 Presetting

5.2.1 Mounting place in PLC-115U



Mounting Places	PS	CPU	0	1	2	3	4	5	6	IM
Building Block Carrier CR700-1										①
Building Block Carrier CR700-2									①	①
Building Block Carrier CR700-3									①	①
Building Block Carrier CR700-0LA										①
Building Block CarrierCR700-0LB						①				①
Building Block Carrier ER701-0										
Building Block Carrier ER701-1										
Building Block Carrier ER701-2										
Building Block Carrier ER701-3										

Fig. 5-1: Mounting place in PLC-115U

The grey areas show where the cards may be plugged in.

① In these mounting places the 24V are <u>not</u> available.

> ⚠ *Please observe the DIL switch positions when using the IM mounting places.*

5.2.2 Mounting places in PLC-135U

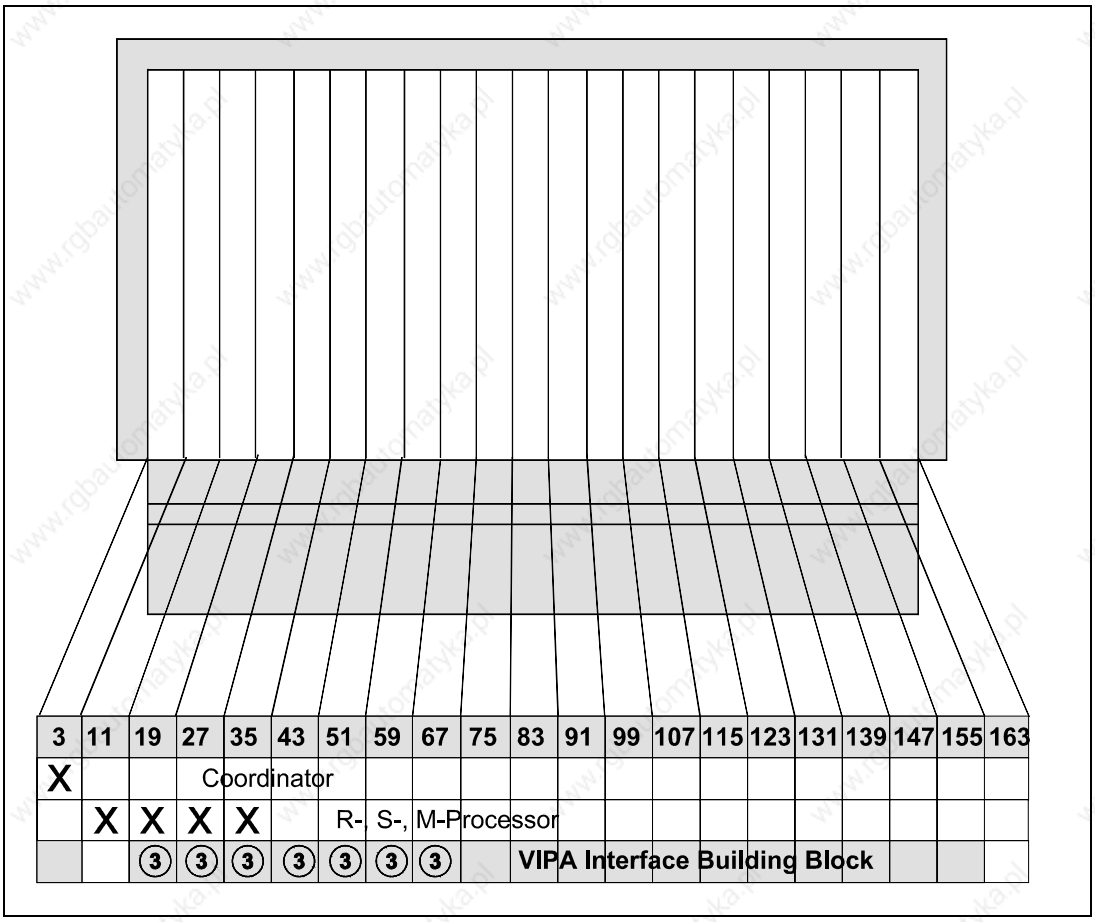


Fig. 5-2: Mounting places in PLC-135U

The grey areas show where the cards may be plugged into.

③ In these mounting places the 24V are available for the active TTY interface

5.2.3 Mounting place in PLC-150U

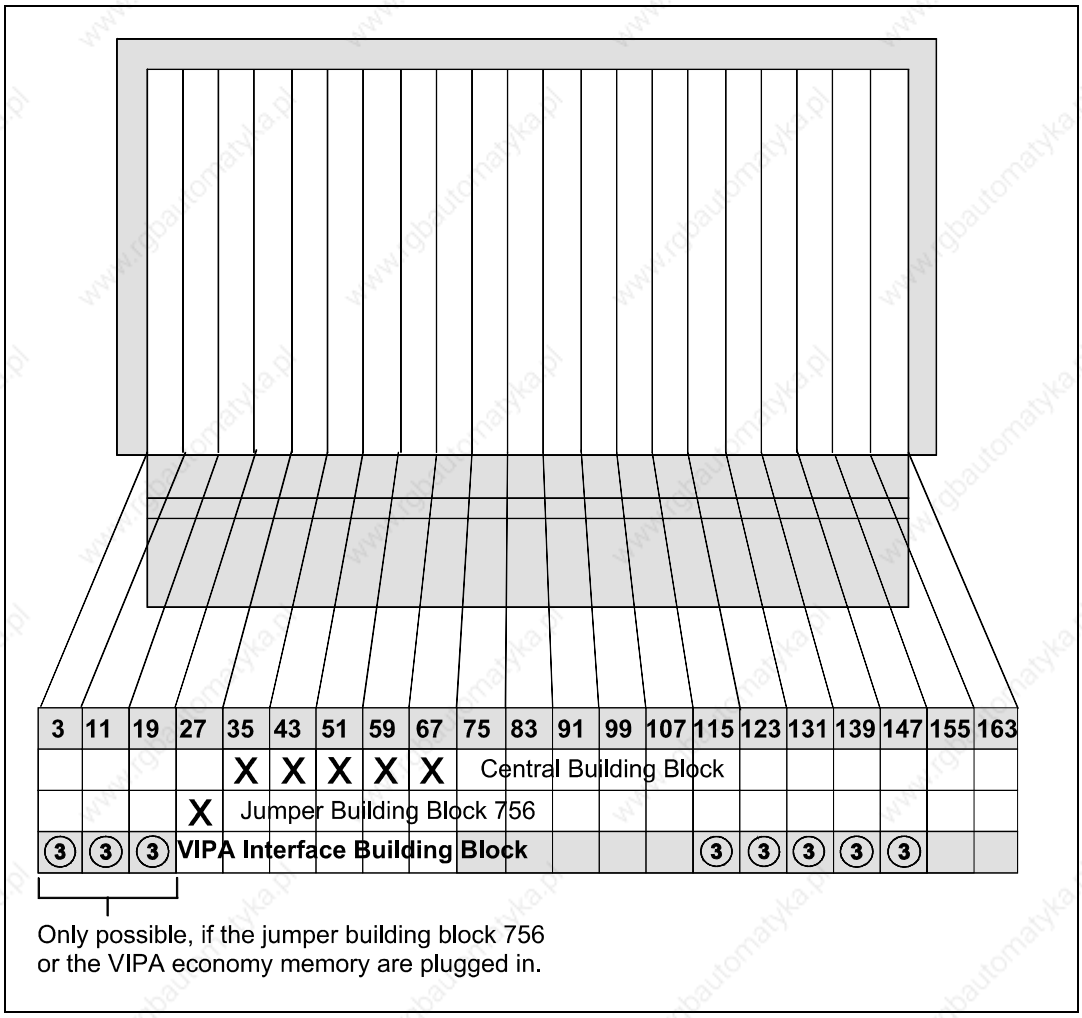


Fig. 5-3: Mounting place in PLC-150U

The grey areas show where the cards may be plugged into.

③ In these mounting places the 24V are available for the active TTY interface.

5.2.4 Mounting place in PLC-155U

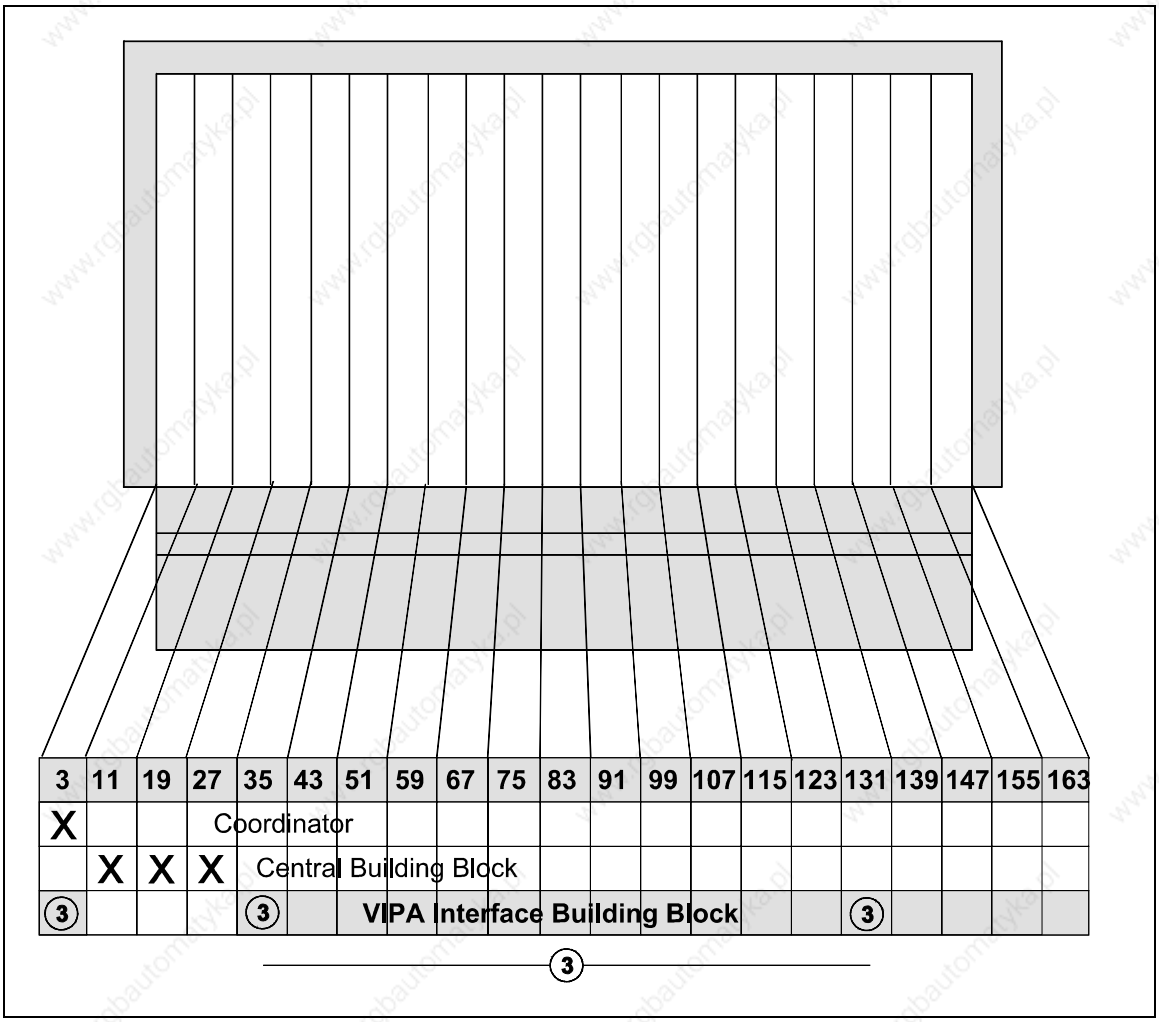


Fig. 5-4: Mounting place in PLC-155U

The grey areas show where the cards may be plugged into.

③ In these mounting places the 24V are available for the active TTY interface.

5.2.5 Mounting place in EG-185U

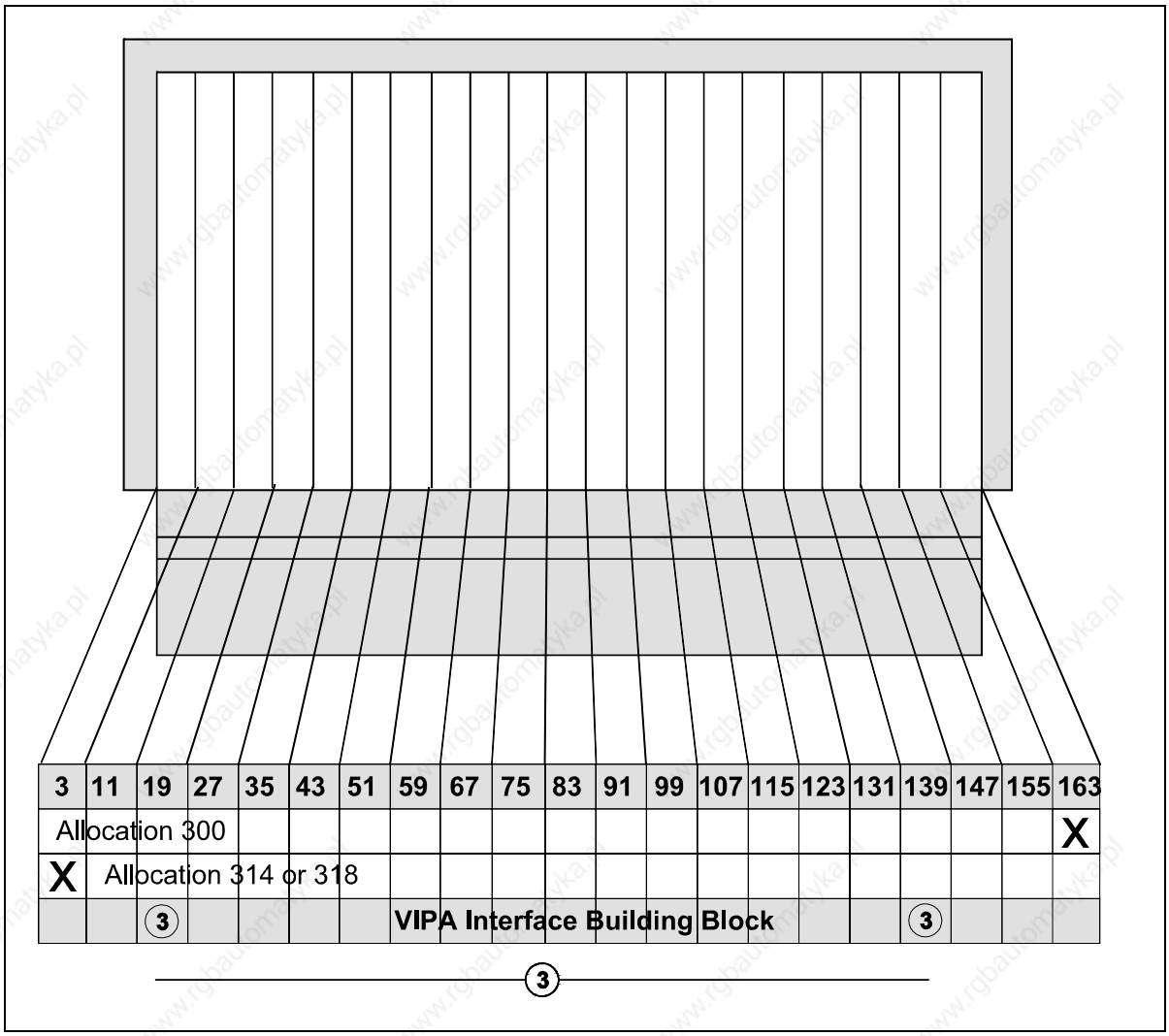


Fig. 5-5: Mounting place in EG-185U

The grey areas show where the cards may be plugged into.

③ In these mounting places the 24V are available for the active TTY interface.

In all other expanded equipment 24V are not available. The interface card can be operated in all mounting places.

5.2.6 Mounting places in PLC-188U

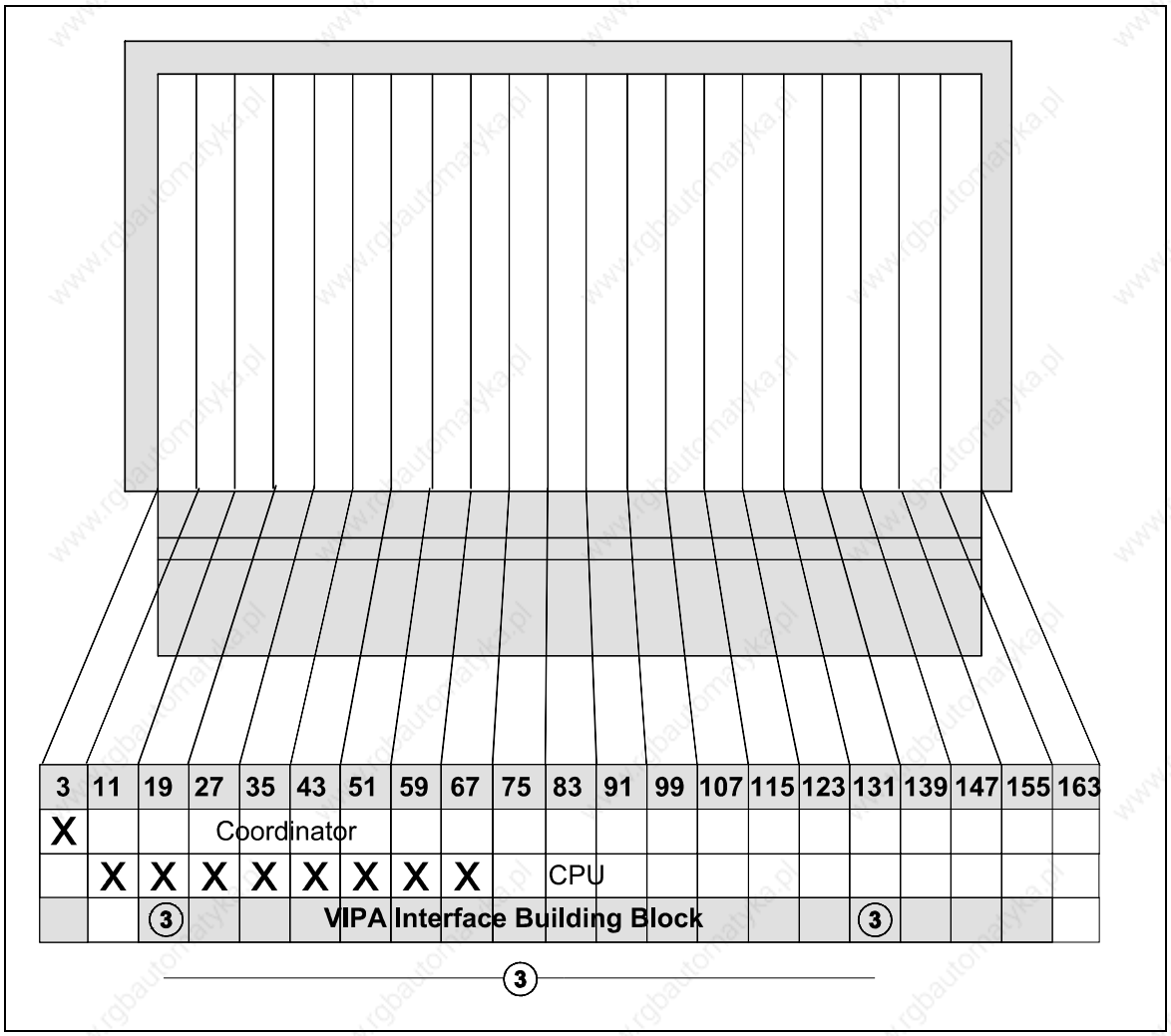


Fig. 5-6: Mounting places in PLC-188U

The grey areas show where the cards may be plugged into.

③ In these mounting places the 24V are available for the active TTY interface.

5.2.7 Setting the address with DIL switches S1 and S2

The start address, under which the interface building block can be operated by the automation equipment, is set with the DIL switches S1 and S2. The addressing of the interface building block is not dependant upon the mounting place! The address, under which it is controlled, is only dependant upon the setting of the DIL switches and **not** from the mounting place in the SPS, i.e. it can occupy any mounting place.

It is possible to set the interface building block in two modes.

Absolute address

With the absolute addressing it is possible to use the entire address space of the automation equipment as you like. For the interface building block BG41, it must be divisible by 4 (without a remainder) and for the interface building blocks BG42 and BG43 divisible by 8 (without a remainder). It is required, that the selected address area is not already occupied by RAM or other building blocks.

The interface building block BG42 occupies eight addresses and uses six addresses, whereas the interface building block BG43 occupies and uses eight addresses.

Peripheral address

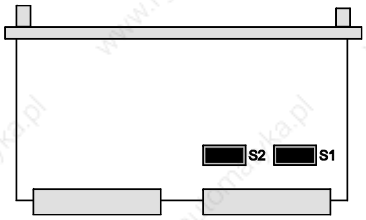
With peripheral addressing the building block can only be addressed in a particular area. In order to do this, bring the switch EG situated on the DIL switch S1 into the ON position. The interface building block can therefore be addressed in the peripheral area. In the peripheral area addresses can only be used which do not run over the process image, i.e. the peripheral bytes PYO and PY127 can not be used.

The interface building block can also be used in an expanded piece of equipment. In this case one should observe, that the building block can still only be addressed in the peripheral area.

If an IM building block is not used in the PLC 115U, the interface building block may be used and by simulating the final plug a mounting place can be saved. Both switches IM and EG on the DIL switch S1 must therefore be switched to ON.

*If the building block is not used on the IM mounting place, the switch IM on the DIL switch S1 must be set to **OFF**!*

5.2.7.1 Examples



abs. Adr.	periph. Adr.	A_7 A_6 A_5 A_4 A_3 A_2
F080	PY128	1 0 0 0 0 0
F084	PY132	1 0 0 0 0 1
F088	PY136	1 0 0 0 1 0
F08C	PY140	1 0 0 0 1 1
F090	PY144	1 0 0 1 0 0
F094	PY148	1 0 0 1 0 1
F098	PY152	1 0 0 1 1 0
F09C	PY156	1 0 0 1 1 1
F0A0	PY160	1 0 1 0 0 0
F0A4	PY164	1 0 1 0 0 1
F0A8	PY168	1 0 1 0 1 0
F0AC	PY172	1 0 1 0 1 1
F0B0	PY176	1 0 1 1 0 0
F0B4	PY180	1 0 1 1 0 1
F0B8	PY184	1 0 1 1 1 0
F0BC	PY188	1 0 1 1 1 1
F0C0	PY192	1 1 0 0 0 0
F0C4	PY196	1 1 0 0 0 1
F0C8	PY200	1 1 0 0 1 0
F0CC	PY204	1 1 0 0 1 1
F0D0	PY208	1 1 0 1 0 0
F0D4	PY212	1 1 0 1 0 1
F0D8	PY216	1 1 0 1 1 0
F0DC	PY220	1 1 0 1 1 1
F0E0	PY224	1 1 1 0 0 0
F0E4	PY228	1 1 1 0 0 1
F0E8	PY232	1 1 1 0 1 0
F0EC	PY236	1 1 1 0 1 1
F0F0	PY240	1 1 1 1 0 0
F0F4	PY244	1 1 1 1 0 1
F0F8	PY248	1 1 1 1 1 0
F0FC	PY252	1 1 1 1 1 1

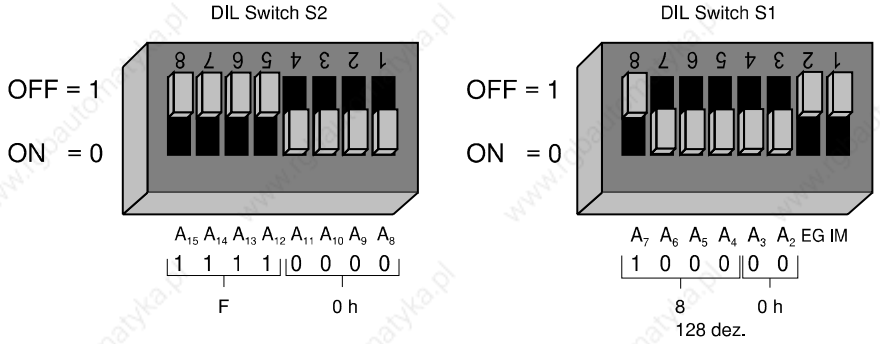
This classification table is only relevant for the building block BG41.

With regards to the building blocks BG42 and BG43 the position of the switch A_2 is not evaluated and therefore the addresses ending in 4 or C are not permitted.

Three examples on addressing are shown in the following. With the exclusion of the switches EG and IM (DIL switch S1) the addresses are depicted in reverse (log. 1 == OFF, log. 0 == ON).

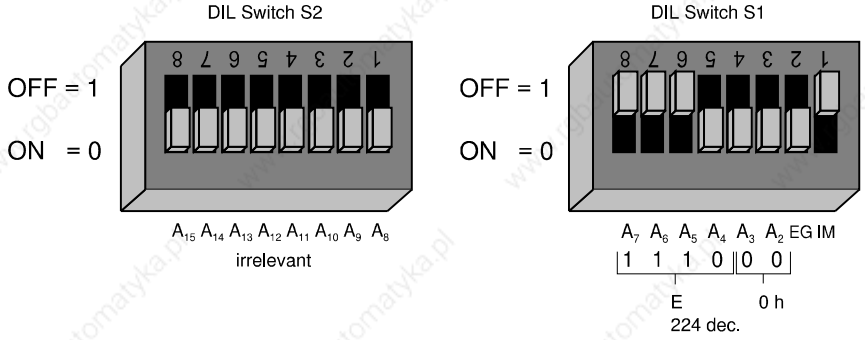
Example 1

The building block is be used in a central device and is controlled under the start address PY128 (=F080h).



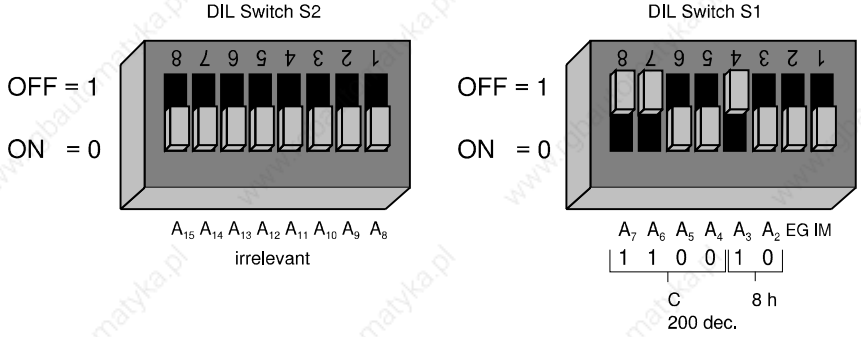
Example 2

The building block is used in an expanded piece of equipment and is controlled under the start address PY224 (=F0E0).



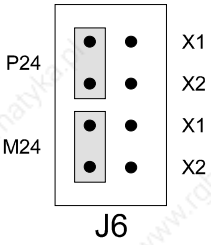
Example 3

The building block is used on the IM position (PLC115-U only) and is controlled under the start address PY200 (=F0C8h).



5.2.8 Plug connectors J6, J11 and J12

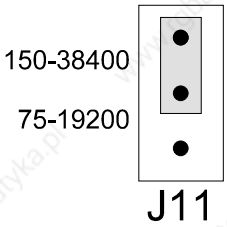
Plug connector J6



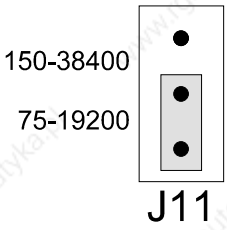
With the plug connector J6 the voltage supply is switched (internally via X1 and X2, externally via the 25-pole SubD socket(s)).
In the position shown here, the plug connector is in the parking position (state at delivery).

Plug connector J11

With the plug connector J11 the area of the eligible baud rate is set.

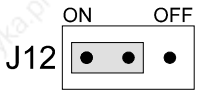


In this setting the baud rates of 150 baud to 38.400 baud are possible.



In this setting (state at delivery) the baud rates of 75 baud to 19.200 baud are possible.

Plug connector J12



With the plug connector J12 the battery's floating charge is set, which is necessary for particular applications.

The building blocks and modules described in this handbook do not require a float charging of the battery. Leave J12 plugged into the position shown (state at delivery).

Position of the DIL switches and plug connectors

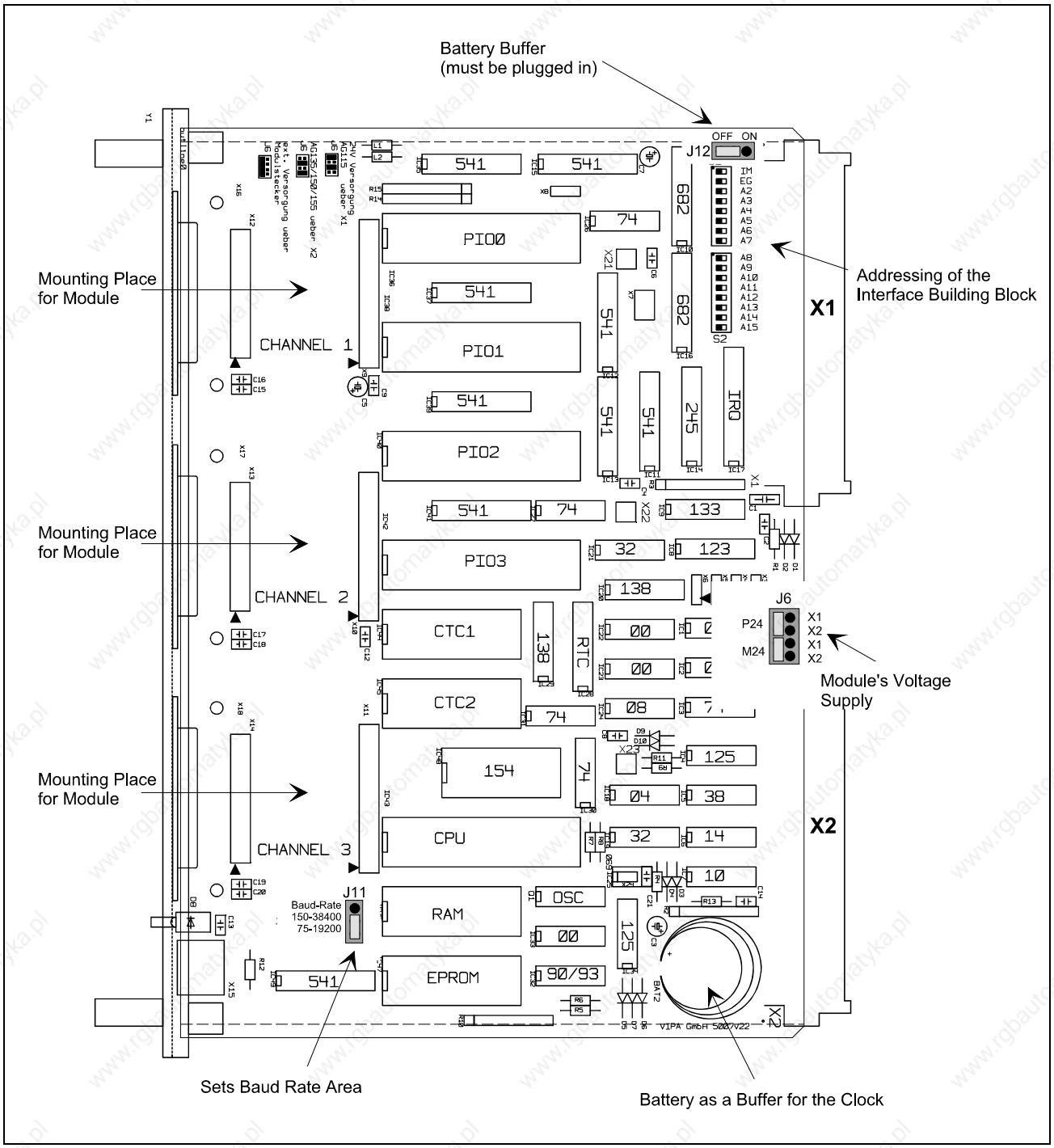


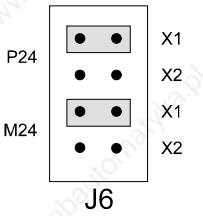
Fig. 5-7: Position of the jumpers and the DIL switches on the BG43

5.2.9 Voltage supply of the module

Internal voltage supply via the back plane bus

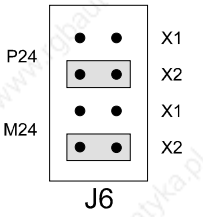
If the PLC on the bus provides 24V, the supply of the current sources can take place via the bus. By means of the connector field J6 on the base circuit board, one can choose if the 24V should be taken from the top or the bottom base plug. If the top base plug (X1) carries 24V, the jumpers are to be plugged as follows (applies for PLC–115U):

Voltage supply via X1 for PLC-115U



If the 24V are only available in the bottom base plug (X2) the plug connectors are to be plugged as follows (applies for PLC-135U, PLC-150U, PLC-155U):

Voltage supply via X2 for PLC-135U, PLC-150U, PLC-155U



The plug connector J1 and J2 on the 20mA module (or on the combination module) are to be brought into the position for internal voltage supply "INT".

5.2.9.1 Circuit diagram 24V supply BG43

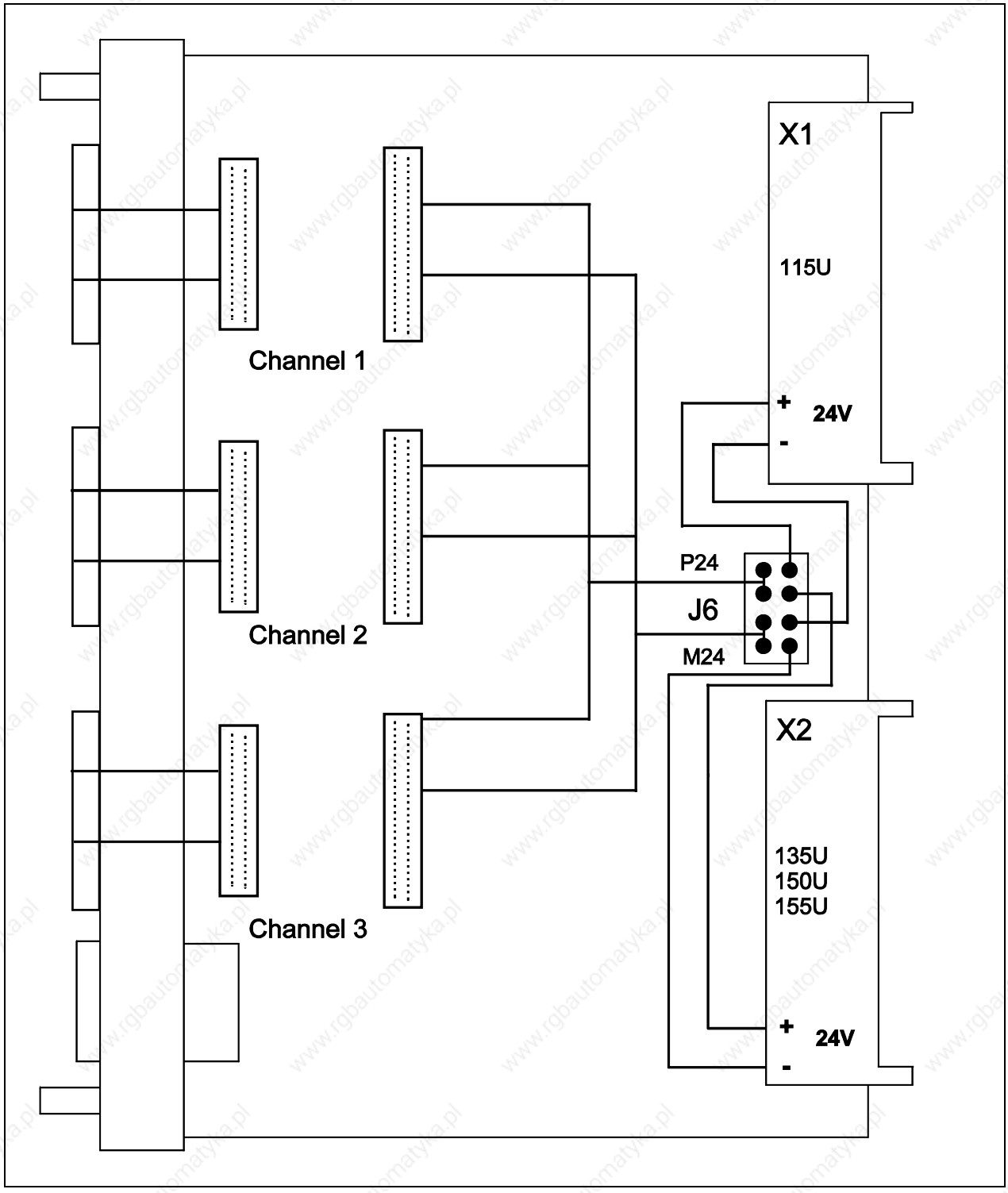


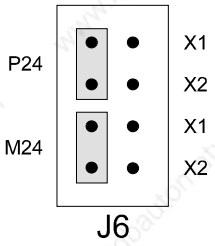
Fig. 5-8: Circuit diagram of the building block BG43, for example

External voltage supply via the 25-pole SubD socket

The plug connectors J1 and J2 are to be found on the 20mA module and the combination module and can be switched to external and internal voltage supply. The plug bridge M24 (J1) switches the grounding conductor, the plug bridge P24 (J2) switches the 24 V. For external voltage supply of the building block, both must be plugged into the position "EXT" (external voltage supply).

On the interface building block the plug connector J6's plug bridges must be plugged into the "parking position" (state at delivery) when receiving its voltage supply externally.

Parking position of the plug connector J6



The connector field J6 is responsible for the feeding through of the 24V supply from the back plane bus. If the building block is externally supplied, it is imperative that the bridges are switched into the position as shown. The external voltage must be applied to the pins 11 (+24 V) and 22 (earth) of the 25-pole sub-D plug.

5.3 Software installation of VIPA's data handling blocks

The data handling blocks which can be acquired from VIPA are found in a packed form on the disk (Order-No.VIPA SSM-SW433).

To install these data handling blocks, please follow the following steps, which may also be found on the disk in the file "Read.me".

- Draw up an index on your data carrier (e.g. on drive C:) with the command
  `MKDIR C:\SW433`
- Change in that index C:\SW433 with the command
  `CD C:\SW433`
- Copy the files from your disk drive (e.g. from drive A:) into this index with the command
  `COPY A:\*.* C:`
- Unpack the files with the command
  `SW433V55`

The files can now be found in unpacked form in the index SW433 on your drive C. All available data handling blocks will be descibed briefly below.

Name of File	Description
130wb@st.s5d	Data handling blocks for a clock, ASCII protocol, counter and STX/ETX protocol for 130wb.
135r@@st.s5d	Data handling blocks for a clock, ASCII protocol, counter, 3964(R) with RK512 and STX/ETX protocol for R processor.
135s@@st.s5d	Data handling blocks for a clock, ASCII protocol, counter and STX/ETX protocol for S processor.
150ak@st.s5d	Data handling blocks for a clock, ASCII protocol, counter and STX/ETX protocol for 150ak.
150u@@st.s5d	Data handling blocks for a clock, ASCII protocol, counter and STX/ETX protocol for CPU 150U.
948946st.s5d	Data handling blocks for a clock, ASCII protocol, counter, 3964(R), 3964(R) with RK512, recording function and STX/ETX protocol for CPU 948, CPU 946.
ag115@st.s5d	Data handling blocks for a clock, ASCII protocol, counter, 3964(R), 3964(R) with RK512, recording function and STX/ETX protocol for CPU 941/B - 944/B.
cpu928st.s5d	Data handling blocks for a clock, ASCII protocol, counter, 3964(R), 3964(R) with RK512, recording function and STX/ETX protocol for CPU 928, CPU 928B.
cpu945st.s5d	Data handling blocks for a clock, ASCII protocol, counter, 3964(R), 3964(R) with RK512, recording function and STX/ETX protocol for CPU 945.

analogst.s5d	Data handling blocks for the analogue module MD40 - MD49 for all CPUs.
ssi@@@st.s5d	Data handling blocks for a synchronized serial interface for all CPUs.
hohzusst.s5d	Data handling blocks for Hohner absolute value generator for all CPUs.
met115st.s5d	Data handling blocks to be connected to the Mettler balance for CPU 941/B to 944/B.
met928st.s5d	Data handling blocks to be connected to the Mettler balance for CPU 928/B.
met945st.s5d	Data handling blocks to be connected to the Mettler balance for CPU 945.
met948st.s5d	Data handling blocks to be connected to the Mettler balance for CPU 948 and CPU 946.

5.4 Startup response

Network-ON
- after plugging in the building block for the first time
  - the standard parameters are set:
    Standard function (without procedure/protocol)
    Baud rate 9600 baud
    Even parity
    1 Start bit
    8 Data bits
    2 Stop bits
  - all buffers are reset

STOP-START
- when restarting
  - the selected function (protocol/procedure) and the changed standard parameter stay the same
  - all buffers are reset

5.5 Error diagnosis

5.5.1 Errors in serial communication

- **The PLC goes into the operation state STOP once the data handling blocks are loaded**

Reason:

The building block address (DIL switch) do not match the address of the handling blocks (parameter ADR).

Remedial action:

Examine the DIL switch position.

By means of the programming equipment on the PLC print out the addresses used by the interface building block with the function "PLC INFO". If values are uneven to FFh (e.g. C0h), the building block is recognized by the SPS-CPU. If the PG reads the value Ffh on these addresses, this implies that the building block is not properly parametized by the DIL switches.

Examine the handling block's address (parameter ADR).

Reason:

The wrong data handling block is used.

Remedial action:

Use the data handling block of the corresponding CPU.

- **No communication is taking place**

Reason:

The plug for the 25-pole SubD socket is wired wrongly.

Remedial action:

Examine the correctness of the interface parameter.

In the case of the module RS232C, examine the bridges between pin 4 and 5

In the case of the combination module used with the RS232C, examine the bridges between pin 7 and 20.

Carry out a loopback test, where you directly reconnect the output with the input.

Check if RECEIVE is called.

5.5.2 Application of the diagnostic interface

As it is becoming more important to monitor the data exchange between the PLC and the peripheral, VIPA building blocks have a separate interface. On this interface the time telegram is available when using the DCF77 antenna module.

When using serial interface modules, the interface signals (Rx and Tx) are made available as a TTL-level on the diagnostic interface.

VIPA offers a diagnostic adaptor for connecting it to your PC. The adaptor cable is connected to both of the serial interfaces on your PC. The TTL signal is converted to the RS232C signal on the diagnostic adaptor.

With suitable diagnostic software one can analyse the signal flow on one's own PC and thereby examine the function of your serial interface.

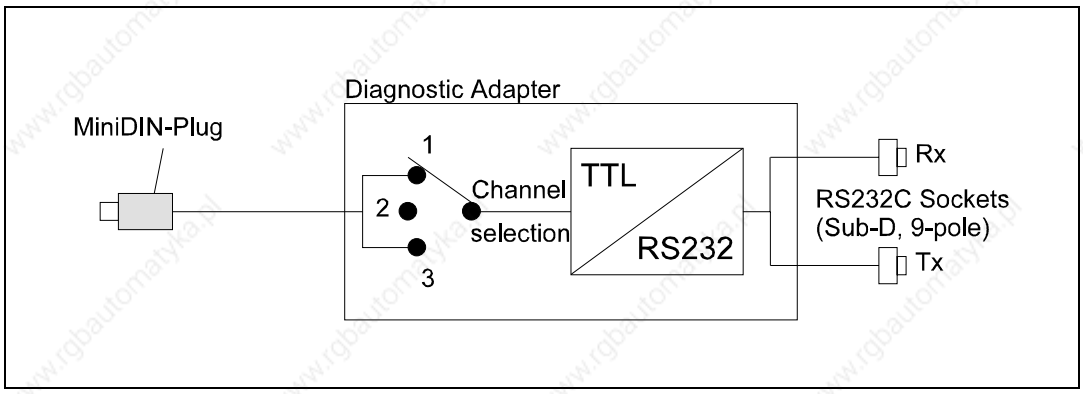


Fig. 5-9: How the diagnostic adaptor works

Using the RS485 or RS485/422P module the diagnostic interface can only be used restricted.

This modules are't able to run in full duplex operation, that means if the receive channel (RXD) isn't active the send channel (TXD) is operating!.

5.5.2.1 Pin allocation of the diagnostic socket

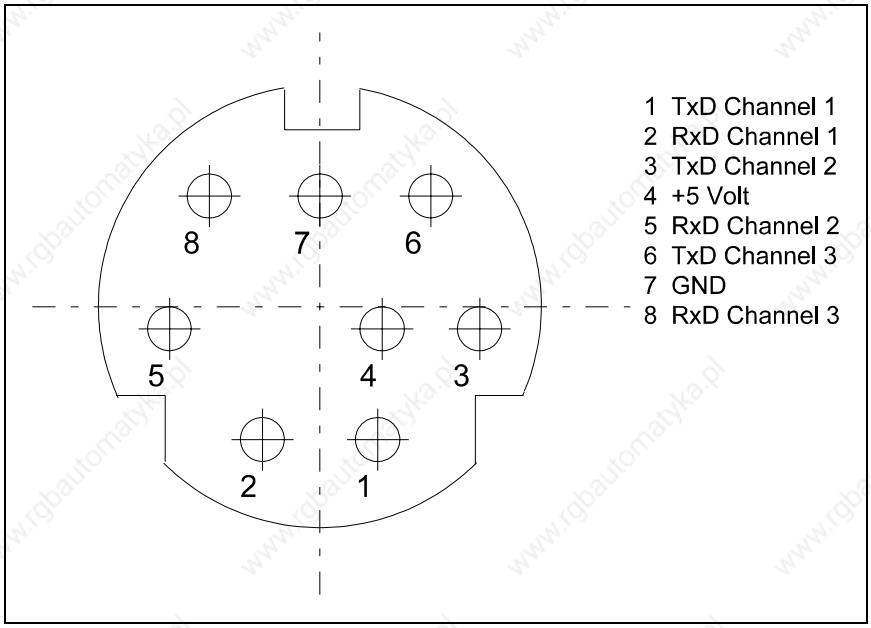


Fig. 5-10: Pin allocation of the diagnostic socket

5.6 Examples

5.6.1 STX/ETX

With this example a SEND order from interface 1 to interface 2 of an SSM-BG42 or an SSM-BG43 is realised.

With the FB ‘PARA-STX’ the start character is restricted to STX (02H) and the end characterisation is restricted to the end character ETX (03h) and EOT (04H). In the FIFO mode on interface 1 all incoming telegrams are filed one behind the other in the card’s 254 byte buffer. With the FB ‘STX-REC’ one can read these telegrams one after another.

FIFO is not operated on interface 2. Only the last telegram is always in the buffer, all previous telegrams which have not been disposed of are deleted.

The SEND order is initiated by setting the M13.0. After the order has been completed the M13.0 is reset.

Cold Restart OB

```
OB20
      BIB     #5095
00000       :
00002       :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                 Base address
      KANR =KF +1                   Interface number
      PROC =KF +1                   Procedure STX/ETX
      FEHL =MB 199                  Error byte
0000E       :
00010       :SPA FB 40
      NAME #PARA-STX
      ADR  =KH F080                 Base address
      K/M  =KY 1,1                  Interface 1/with FIFO
      STX  =KY 0,2                  Start classification is 02H
      E1/2 =KY 3,4                  End classification is 03H04H
      ZVZ  =KF +50                  2500ms(Value * 50ms)
      FEHL =MB 198                  Error byte
00020       :
00022       :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                 Base address
      KANR =KF +2                   Interface number
      PROC =KF +1                   Procedure STX/ETX
      FEHL =MB 199                  Error byte
0002E       :
00030       :SPA FB 40
      NAME #PARA-STX
      ADR  =KH F080                 Base address
      K/M  =KY 2,0                  Interface 2/without FIFO
      STX  =KY 0,2                  Start classiication is 02H
      E1/2 =KY 3,4                  End classification is 03H04H
      ZVZ  =KF +50                  2500ms(Value * 50ms)
      FEHL =MB 198                  Error byte
00040       :BE
```

Restart OB

```
OB21
      BIB     #5095
00000       :
00002       :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                    Base address
      KANR =KF +1                      Interface number
      PROC =KF +1                      Procedure STX/ETX
      FEHL =MB 199                     Error byte
0000E       :
00010       :SPA FB 40
      NAME #PARA-STX
      ADR  =KH F080                    Base address
      K/M  =KY 1,1                     Interface 1/with FIFO
      STX  =KY 0,2                     Start classification is 02H
      E1/2 =KY 3,4                     End classification is 03H04H
      ZVZ  =KF +50                     2500ms(Value * 50ms)
      FEHL =MB 198                     Error byte
00020       :
00022       :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                    Base address
      KANR =KF +2                      Interface number
      PROC =KF +1                      Procedure STX/ETX
      FEHL =MB 199                     Error byte
0002E       :
00030       :SPA FB 40
      NAME #PARA-STX
      ADR  =KH F080                    Base address
      K/M  =KY 2,0                     Interface 2/without FIFO
      STX  =KY 0,2                     Start classification is 02H
      E1/2 =KY 3,4                     End classification is 03H04H
      ZVZ  =KF +50                     2500ms(Value * 50ms)
      FEHL =MB 198                     Error byte
00040       :BE
```

Power On OB

```
OB22
      BIB     #5095
00000       :
00002       :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                 Base address
      KANR =KF +1                   Interface number
      PROC =KF +1                   Procedure STX/ETX
      FEHL =MB 199                  Error byte
0000E       :
00010       :SPA FB 40
      NAME #PARA-STX
      ADR  =KH F080                 Base address
      K/M  =KY 1,1                  Interface 1/with FIFO
      STX  =KY 0,2                  Start classification is 02H
      E1/2 =KY 3,4                  End classification is 03H04H
      ZVZ  =KF +50                  2500ms(Value * 50ms)
      FEHL =MB 198                  Error byte
00020       :
00022       :SPA FB 100
      NAME #PROCW
      ADR  =KH F080                 Base address
      KANR =KF +2                   Interface number
      PROC =KF +1                   Procedure STX/ETX
      FEHL =MB 199                  Error byte
0002E       :
00030       :SPA FB 40
      NAME #PARA-STX
      ADR  =KH F080                 Base address
      K/M  =KY 2,0                  Interface 2/ without FIFO
      STX  =KY 0,2                  Start classification is 02H
      E1/2 =KY 3,4                  End classification is 03H04H
      ZVZ  =KF +50                  2500ms(Value * 50ms)
      FEHL =MB 198                  Error byte
00040       :
00042       :UN  M 1.0              Set new start flag
00044       :S   M 1.0
00046       :BE
```

Cycle OB

```
OB1
      BIB     #5104
00000       :
00002       :L   KF +8                Send data from DW8
00006       :T   MW 10
00008       :L   KF +16               Send 16 byte
0000C       :T   MB 12
0000E       :
00010       :U   M 1.0                Send initiation
00012       :S   M 13.0               Set transmission access
00014       :SPA FB 43
      NAME #STX-SEND
      ADR  =KH F080                   Base address
      K/DB =KY 1,10                   Interface 1/ from DB 10
      ANF  =MW 10                     Flag word with 1.DW
      ANZ  =MB 12                     Flag byte with length
      FRG  =M 13.0                    Transmission access
      FEHL =MB 14                     Error byte
      NEUS =M 0.0                     New start flag
00026       :
00028       :L   KF +0                1. DW of the reception buffer
0002C       :T   MW 22
0002E       :L   KF +50               Length of the reception buffer in by
00032       :T   MB 24
00034       :SPA FB 44
      NAME #STX-REC
      ADR  =KH F080                   Base address
      K/DB =KY 2,11                   Interface 2/reception DB 11
      SZ   =MB 20                     Writing indicator
      LZ   =MB 21                     Reading indicator
      ANF  =MW 22                     Flag word with buffer start
      ANZ  =MB 24                     Flag byte with buffer length
      FEHL =MB 25                     Error byte
      NEUS =M 0.0                     New start flag
00048       :
0004A       :SPA FB 28
      NAME #RB_NEUS
      ADRE =KH F080                   Base address
      NEUS =M 0.0                     New start flag
      PAFE =MB 9                      Error byte
00054       :
00056       :L   MB 20                Writing indicator
00058       :L   KB 0
0005A       :!=F                      No telegram received
0005C       :BEB
0005E       :
00060       :                         Analyse data
00062       :
00064       :L   KB 0
00066       :T   MB 20                Writing indicator at start
00068       :BE
```

5.6.2 3964R

With this example a SEND order from interface 1 to interface 2 of the SSM-BG42 or the SSM-BG43 is realised.

The SEND order is initiated by setting the M1.0. After the order is completed the M1.0 is reset. As described in this example, the FB ‘SEND’ must run until "order completed" is reported back.

In the following the building blocks necessary for the two orders are given:

Cold Restart OB

```
OB20
      BIB     #6075
00000       :SPA FB 37
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +1                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even,2 Stop bits,8bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and the card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0000C       :
0000D       :SPA FB 37
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +2                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even,2 Stop bits,8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and the card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
00019       :
0001A       :BE
```

Restart OB

```
OB21
      BIB     #6075
00000       :SPA FB 37
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +1                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even,2 Stop bits,8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and the card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0000C       :
0000D       :SPA FB 37
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +2                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even,2 Stop bits,8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and the card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
00019       :
0001A       :BE
```

Power On OB

```
OB22
      BIB     #6075
00000       :SPA FB 37
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +1                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even,2 Stop bits,8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and the card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0000C       :
0000D       :UN  M 0.0                Set the new start flag in order to
0000E       :S   M 0.0                block the building block in OB1
0000F       :SPA FB 37
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +2                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even,2 Stop bits,8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and the card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0001B       :
0001C       :BE
```

Cycle OB

```
OB1
      BIB     #5104
00000       :
00002       :U   M 1.0
00004       :SPB FB 33
      NAME #SEND
      ADRE =KH F080                  Base address
      SSNR =KF +1                    Interface number
      A-NR =KF +1                    Order number
      QTYP =KC MB                    Source area is flag
      QDBN =KY 0,0                   Irrelevant
      QBWN =KY 0,5                   From MB 5
      BWAN =KF +40                   Send 40 flag bytes
      ANZW =MW 2                     Indicator word
      PAFE =MB 4                     Error byte
      NEUS =M 0.0                    New start flag
0001C       :O   M 3.2                Order completed without an error
0001E       :O   M 3.3                Order completed with an error
00020       :R   M 1.0                Reset initiation
00022       :
00024       :SPA FB 31
      NAME #RECEIVE
      ADRE =KH F080                  Base address
      SSNR =KF +2                    Interface 2
      A-NR =KF +2                    Order number
      ZTYP =KC DB                    Aim is the data block
      DBNR =KY 0,11                  DB no. is 11
      BWNR =KY 0,0                   File data from DW 0
      MXBW =KF +80                   Max. buffer size 80 words
      ANZW =MW 6                     Indicator word
      PAFE =MB 8                     Error byte
      NEUS =M 0.0                    New start flag
0003C       :
0003E       :SPA FB 28
      NAME #RB_NEUS
      ADRE =KH F080                  Base address
      NEUS =M 0.0                    New start flag
      PAFE =MB 9                     Error byte
00048       :BE
```

5.6.3 3964R with RK512

With this example the SEND order between interface 1 and 2 and a FETCH order between interface 2 and 3 of the building block BG42 and BG43 are realised.

The SEND order is initiated by setting the flag M1.0. After the SEND order is completed M1.0 is reset. As described in the example, the FB 'SEND' must run until "order completed" has been reported back.

The FETCH order is initiated by setting the flag M1.1. After the FETCH order has been completed M1.1 is reset. As described in the example, the FB 'FETCH' must run until "order completed" has been reported back.

The building blocks necessary for the two orders are provided in the following:

Cold Restart OB

```
OB20
      BIB     #6075
00000       :SPA FB 27
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +1                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even, 2 Stop bits, 8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0000C       :
0000D       :SPA FB 27
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +2                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even, 2 Stop bits, 8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
00019       :
0001A       :BE
```

Restart OB

```
OB21
      BIB     #6075
00000       :SPA FB 27
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +1                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even, 2 Stop bits, 8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0000C       :
0000D       :SPA FB 27
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +2                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even, 2 Stop bits, 8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
00019       :
0001A       :BE
```

Power On OB

```
OB22
      BIB     #6075
00000       :SPA FB 27
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +1                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even, 2 Stop bits, 8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0000C       :
0000D       :UN  M 0.0                Set the new start flag in order to
0000E       :S   M 0.0                block the building block in OB 1
0000F       :SPA FB 27
      NAME #SYNCHRON
      ADRE =KH F080                  Base address
      SSNR =KF +2                    Interface number
      BAUD =KF +10                   Baud rate (9600)
      FORM =KM 00000001 11111100     Parity even, 2 Stop bits, 8 bits per byte
      PRIO =KF +0                    Low transmission priority
      BLOC =KF +64                   Block size between SPS and card
      ZVZ  =KF +20                   Character delay time 200 ms
      QVZ  =KF +220                  Acknowledgement delay time 2200 ms
      PAFE =MB 199                   Error byte
      NEUS =M 0.0                    New start flag
0001B       :
0001C       :BE
```

Cycle OB

```
OB1
00001      :U   M 1.0                Send access
00002      :SPB FB 23
      NAME #SEND
      ADRE =KH F080                 Base address
      SSNR =KF +1                   Interface number
      A-NR =KF +1                   Order number
      QTYP =KC DB                   Source is DB
      QDBN =KY 0,10                 Number of data blocks
      QBWN =KY 0,0                  1. data word to be sent
      BWAN =KF +50                  Number of DW to be sent
      ZTYP =KC DB                   Aim is DB
      ZDBN =KY 0,11                 Data block 11
      ZBWN =KF +0                   1. Word of the reception buffer
      KOOR =KY 255,255              Without a coordination flag
      ANZW =MW 2                    Indicator word in MW 2
      PAFE =MB 197                  Error byte
      NEUS =M 0.0                   New start flag
00012      :O   M 3.2                Order completed without an error
00013      :O   M 3.3                or order completed with an error
00014      :R   M 1.0                then SEND is not to be run any longer
00015      :
00016      :U   M 1.1                Access Fetch
00017      :SPB FB 25
      NAME #FETCH
      ADRE =KH F080                 Base address
      SSNR =KF +2                   Interface number
      A-NR =KF +2                   Order number r
      QTYP =KC MB                   Source is flag area
      QDBN =KY 0,0                  Irrelevant
      QBWN =KY 0,5                  From flag byte 5
      BWAN =KF +46                  46 flag bytes
      ZTYP =KC MB                   Aim is flag area
      ZDBN =KY 0,0                  Irrelevant
      ZBWN =KF +10                  Record from MB 10
      KOOR =KY 255,255              Without coordination flag
      ANZW =MW 4                    Indicator word
      PAFE =MB 195                  Error byte
      NEUS =M 0.0                   New start flag
00027      :O   M 5.2                Order completed without an error
00028      :O   M 5.3                or order completed with an error
00029      :R   M 1.1                then FETCH is not to be run any longer
0002A      :
0002B      :SPA FB 24                Building block to receive the
      NAME #REC ALL                 Data of SEND
      ADRE =KH F080                 Base address
      SSNR =KF +2                   Interface number
      ANZW =MW 4                    Indicator word
      PAFE =MB 196                  Error byte
      NEUS =M 0.0                   New start flag
00032      :
00033      :SPA FB 22                Building block to transfer the data
      NAME #SEND-ALL                for FETCh
      ADRE =KH F080                 Base address
      SSNR =KF +1                   Interface number
      PAFE =MB 194                  Error byte
      NEUS =M 0.0                   New start flag
00039      :
0003A      :SPA FB 28                Building block access after OB1 has
      NAME #RB_NEUS                 run through network ON
      ADRE =KH F080                 Base address
      NEUS =M 0.0                   New start flag
      PAFE =MB 195                  Error byte
00040      :BE
```

5.7 Structure guidelines

The structure guidelines contain information on the interference immune structure of stored-program controls. It is described how interferences in automation equipment can occur, how electromagnetic compatibility (EMC) can be saved and how to act by a screening.

5.7.1 What does EMC mean?

One understands by electromagnetic compatibility (EMC) the capability of an electric piece of equipment to function without an error in a pre-given electromagnetic environment, without being influenced by its environment or itself influence its environment in an undue manner.

All products are developed for operation in tough industrial environments and fulfil high standards with regard to EMC. However, you should carry out an EMC planning before installing the control and include possible sources of interference in the examination.

5.7.2 Overview of the possible effects of interferences

Elektromagnetic interferences can launch into the automation equipment in different ways.

- Fields
- E/A signal lines
- Bus system
- Current supply
- Ground wire

Depending on the expansion medium (conducted or not) and the distance from the source of interference, interferences succeed in entering the automation equipment in different coupling mechanisms.

They are differentiated as follows:

- galvanic coupling
- capacitive coupling
- inductive coupling
- radiation coupling

5.7.3 The most important basic rules to guarantee EMC

In order to guarantee EMC, it often suffices when one keeps to a few elementary rules. Therefore observe the following guidelines when constructing the controls:

- Ensure that you use a good configured two-dimensional earth of inactive metal parts when setting up an automation piece of equipment.
  - Establish a central connection between the ground and the earth electrode/protective earth conductor system.
  - Connect all inactive metal parts on a large area and with little impedance.
  - Use no aluminium parts if possible. Aluminium oxidises easily and is therefore less suitable for the earth.

- Ensure to have a proper cable tracing when wiring.
  - Divide the cabling into cable groups.
    (Heavy current, current supply cables and signal and data cables).
  - Always lay the heavy current cables and the signal and data cables in separate channels or bundles.
  - Lead the signal and data cables as tight as possible on the ground area (e.g. supporting bars, metal rails, thin metal barriers).

- Ensure to fasten the cable screens perfectly.
  - Data cables are to be laid shielded. The screen is to be applied on both sides.
  - Analogue cables are to be laid shielded. The laying of screens on one side can be advantageous when transmitting signals with small amplitudes.
  - Lay the cable screens largely on a screen/protective earth conductor rail after the barrier entry and secure the screens with cable clamps.
  - Ensure that the screen/protective earth conductor rail is connected with little impedance to the barrier.
  - Use metallic or metallized connector housing for screened data cables.

- Use special EMC measures in particular application cases.
  - Wire all inductances with capacities, which are not controlled by VIPA building blocks.
  - Use incandescent lamps to illuminate the barriers and avoid fluorescent lamps.

- Create a uniform reference potential and earth all resources where possible.
  - Be aware of the intended input of earthing measures. The earthing of the control serves as a protection and functional measure.
  - Connect the plant parts and barriers with the central and expanded equipment in stelliform with the earth/protection earth conductor system. The formation of earth loops is therefore prevented.
  - Lay in potential differences enough dimensioned potential equalization cables between plant parts and barriers.

5.7.4 Screening of cables

Electric, magnetic or electromagnetic interference fields are weakened by a screening; here one talks about damping.

Interfering currents are conducted to earth on cable screens through screen rails conductively connected to the case. Here one must insure, that the connection to the protective earth conductor is impedance weak, or else the interference currents would become interference sources.

When screening cables the following is to be observed:

- Use where possible only cables with a braided shield. .
- The congruent thickness of the screens should be more than 80%.
- Avoid cables with membrane screens, as the foil can be easily damaged by tensile and compression loads: the result is a reduction in the screen effect.
- Usually the screens should always be put on both sides. A good interference suppression in a high frequency area can only be achieved by a connection of the screens on both sides.

  In the case of an exception the screen can be put on one side. Thereby one only achieves a damping of the lower frequencies. A one-sided screen connection can be better, if:
  - the cable laying of a equipotential bonding conductor can not be carried out
  - analogue signals (a few mV or µA) are transmitted.
  - membrane screens (static screens) are used.
- Always use metallic or metallized plugs by data cables for serial couplings. Fasten the data cable's screen to the plugs case. Do **not** put the screen onto pin 1 of the connector strip!
- It is advisable by stationary operation to strip the isolation of the screened cable continuosly and to put it on the screen/protective earth conductor rail.

By potential differences between the earthing points, a compensatory current can flow over the screen connected on both sides.

Remedial action: potential compensation cable

- Use metal cable clamps to secure the braided shields. The clamps must surround the screen extensively and have good contact.
- Place the screen in the cabinet on a screen rail directly after the cable has entered. Direct the screen to the building block , do **not** put it on again!

6 Technical data

6 Technical data

6.1 Interface building blocks

Voltage supply	+5 V ± 5 %
Current consumption	600 mA
Internal processor clock frequency	10 MHz
Rate of transmission asynchronous	75 ... 38.400 baud
Handshake by RS232C	RTS/CTS
Dimensions	
- Height	233,4 mm
- Depth	160,0 mm
Weight	ca. 400 g
Environmental conditions	
- operating temperature	0 °C ... 55 °C
- storage and transport temp.	-20 °C ... 70 °C
- relative air humidity (no dewfall)	95 % at 25 °C
- no forced ventilation	
Space requirement	1 mounting place
Can be used on the IM mounting place	yes

6.2 Module

6.2.1 20mA current loop module

Current supply	+5 V (via building block)
Current consumption	12 mA
Rate of transmission asynchronous	75 ... 38.400 baud
Handshake	-
Potential separation	yes
20 mA current sources	2
Load voltage	DC 24 V
- valid area	10 V ... 36 V DC
- internally via back plane bus (Base plug)	
- externally via the 25-pole socket	
Constant current at 24 V	ca. 22 mA
Dimensions	
- Height	50 mm
- Depth	70 mm with LED

6.2.2 20mA-/RS232C combination module

Voltage supply	+5 V (via building block)
Current consumption	30 mA
Rate of transmission asynchronous	75 ... 38.400 baud
Handshake	RTS/CTS
Potential separation:	
- 20 mA Current Loop	yes
- RS232C	no
20 mA current sources	2
Load voltage	typ. DC 24 V
- valid area	10 ... 36 V DC
- internally via back plane bus (Base plug)	
- externally via the 25-pole socket	
Constant current at 24 V	ca. 22 mA
Dimensions	
- Height	50 mm
- Depth	70 mm with LED

6.2.3 RS232C module

Voltage supply	+5 V (via building block)
Current consumption	55 mA
Rate of transmission asynchronous	75 ... 38.400 baud
Handshake	RTS/CTS
Potential separation	no
Dimensions	
- Height	50 mm
- Depth	70 mm with LED

6.2.4 RS422/RS485 module

Voltage supply	+5 V (via building block)
Current consumption	85 mA
Potential separation	no
Rate of transmission asynchronous	75 ... 38.400 baud
Terminal resistance	100 Ω (connectable through socket link)
Dimensions	
- Height	50 mm
- Depth	70 mm with LED

6.2.5 RS422P/RS485P module

Voltage supply	+5 V (via building block)
Current consumption	100 mA
Potential separation	yes
Rate of transmission asynchronous	75 ... 38.400 baud
Terminal resistance (send and receiving lines)	100 Ω (connectable by switch)
Dimensions	
- Height	50 mm
- Depth	90 mm with LED

6.2.6 CENTRONICS module

Voltage supply	+5 V (via building block)
Current consumption	40 mA
Potential separation	no
Dimensions	
- Height	50 mm
- Depth	70 mm with LED

6.2.7 SSI module

Width of data word	24 bit hardwired
Type of data	Binary or Gray code can be selected by hardware adjustment
Number of channels	2
Type of transmission	RS485
Type of cable	Twisted pair
Pulse rate	625 kHz
Voltage supply	+5 V
(via building block)	+24 V
Current consumption	
+5 V (without load)	typ. 100 mA
+24 V load	max. 0,5 A (electr. protected)
cable length	60 m
Dimesions	
- Height	ca. 50,8 mm
- Depth	ca. 139,7 mm

6.2.8 Counter module (5V/24V)

Voltage supply	+5 V (via building block)
Current consumption	390 mA
Number of counters	3
Frequency	max. 250 kHz
Outputs for Carry-Borrow	1 per counter
Types of operation	8
Voltage input MD18 (24V counter)	
- 0 level	0 ... 12 V
- 1 level	14 ... 25 V
Voltage input MD19 (5V counter) TTL level or differential drive	
- 0 level	0 ... 2 V
- 1 level	3 ... 5 V
Output voltage	
- Carry-Borrow	200 mA at 5 V
Dimensions	
- Height	50 mm
- Depth	145 mm with LED

6.2.9 Analogue input module

MD40, MD41, MD42, MD43

Number of inputs	8	
Potential separation	yes (Inputs versus earth point and not inputs against each other)	
Input areas (nominal values)		
- MD40	±10 V or 0 ... 20 V	
- MD41	±5 V or 0 ... 10 V	
- MD42	±2,5 V or 0...5 V	
- MD43	±10 mA or 0...20 mA or 4...20 mA	
Input resistance		
- MD40	min.	250 KOhm
- MD41	min.	350 KOhm
- MD42	min.	1 MOhm
- MD43	max.	500 Ohm
Connection type of the signaling transmitter	Double wire connection voltage and current measurement on a joint reference potential	
Digital depiction of the input signal	11 bit + sign 12 bit complement to two	
Depiction of measurement value	complement to two	
Measurement principle	successive approximation	
Conversion time per measurement value	10 µs	
Cycle time per input	max. 1 ms	
Processing time for the output on the SPS		
1 value read	normalized	0,4-1,2 ms
	hex	0,3-0,7 ms
8 values read	normalized	0,7-1,2 ms
	hex	0,6-0,8 ms
3 modules(24 inputs)	normalized	3,2-4,2 ms
	hex	3 ms
6 modules(48 inputs)	normalized	6,4-8,4 ms
	hex	5,5-6,4 ms
Valid input voltage (destruction level)	max. 200 V DC	
Measurement tolerance	max. ± 3 %	

Temperature error	±0,1 %
Rating of the isolation	according to VDE 0160
Line length screened	max. 200 m
Voltage supply	+5 V (via building block)
current consumption	typ. 133 mA
Dimensions	
- Height	ca. 50,8 mm
- Depth	ca. 118 mm

MD44

Number of inputs	4
Input areas	Pt100 (-200 ... +850 °C)
Connection type of the signaling transmitter	Four wire connection
Resolution	0,1 °C
Resolution internally	19 bit
Digital depiction of the input signal	16 bit complement to two
Depiction of measurement value	Complement to two
Measurement principle	Multi Slope
Conversion time	typ. 1,5 s / 4 channels
Cycle time per input	max. 1 ms
Meausurement tolerance	max. ± 0,3 °C
Line length screened	max. 200 m
Voltage supply	+5 V (via building block)
Current consumption	typ. 100 mA
Dimensions	
- Height	ca. 50,8 mm
- Depth	ca. 130 mm

6.2.10 Analogue output module

Number of Outputs	4
Potential separation	yes (Outputs versus earth points, not outputs against each other)
Output areas (nominal values)	
- MD45	±10 V
- MD46	±5 V
- MD47	0...10 V
- MD48	0...5 V
- MD49	0...20 mA or 4...20 mA
Load resistance	
- MD45	min. 3 KOhm
- MD46	min. 3 KOhm
- MD47	min. 3 KOhm
- MD48	min. 3 KOhm
Load connection typee	Two wire connection, Outputs with joint reference potential
Digital depiction of the output signal	11 bit + sign or 12 bit complement to two
Conversion time	1 ms
Output tolerance	max. ± 3 %
Temperature error	max. ±0,1 %
Short-circuit protection	yes
Short-circuit current	ca. 30 mA
Line length screened	max. 200 m
Voltage supply	+5 V (via building block)
Current consumption	typ. 300 mA
Dimensions	
- Height	50,8 mm
- Depth	127 mm

6.2.11 DCF77 antenna module

Voltage supply	+24 V (via interface module)
Current consumption	max. 20 mA
Reception frequency	77,5 kHz
Reception area	ca.1.500 km radius around Frankfurt/Main
Width of band	ca. 30 Hz
Standard perimeter	ca. 40 dB
Plug connector	25pol. Sub-D plug
Antenna connection cable	ca. 5 m max. 1.000 m
Operation temperature	0 °C ... +55 °C
Time error of the radio clock	practically zero
Time error of the quartz clock	< 1:20.000
Dimensions	
- Height	113 mm
- Width	67 mm
- Length	235 mm
Case	weather proof, ABS synthetics, IP65
Weight	ca. 300 g

6.3 FB's memory space requirement

FB-No.	Size in Bytes								
	150AK	**150U**	**941-944**	**130WB**	**135R**	**135S**	**928/B**	**948**	**945**
FB2			168				154	178	172
FB3	206	226	210	196	198	204	212	238	280
FB4	210	224	218	200	238	244	256	402	284
FB5	252	328	244	252	244	244	244	228	252
FB7		346	434		346	346	410	494	466
FB10	116	102	174	116	102	102	128	194	154
FB11	132	112	162	132	112	112	144	176	174
FB12	152	136	152	150	136	136	136	168	164
FB13	212	232	216	202	204	210	218	244	288
FB14	216	250	224	206	244	250	262	308	292
FB15	262	338	258	262	254	258	254	238	266
FB16			440		352		416	494	474
FB17		352	1098			352	1178	1284	1142
FB18			554				552	586	584
FB20			716		850		846	742	756
FB21			1636		1512		1642	1568	1936
FB22			1054		912		1106	990	1288
FB23			1594		1432		1618	1488	1890
FB24			1294		1146		1344	1260	1580
FB25			1592		1442		1594	1472	1854
FB26			1180		1088		1248	1152	1272
FB27			1008		716		1154	998	1068
FB28			132				162	178	148
FB30			716				846	728	758
FB31			1510				1520	1464	1634
FB33			1392				1430	1312	1534
FB37			1008				1154	996	1064
FB40	224	188	224	224	224	224	224	194	236
FB43	218	232	232	208	210	216	228	254	298
FB44	228	258	238	218	276	274	286	310	306
FB45	178	178	178	178	178	178	178	178	178
FB50	148	148	148	148	148	148	148	148	148
FB51	180	180	180	180	180	180	180	180	180
FB52	180	180	180	180	180	180	180	180	180
FB53	144	144	144	144	144	144	144	144	144
FB54	144	144	144	144	144	144	144	144	144
FB55	182	182	182	182	182	182	182	182	182
FB56	168	168	168	168	168	168	168	168	168
FB100	128	178	128	128	178	128	178	134	140
FB101	134	180	134	134	180	134	180	138	146
FB102	494	1606	624	494	1606	450	1942	2482	720
FB103	395	1612	578	394	1612	404	1954	2494	604

Tab. 6-1: Memory space requirements of the individual FBs

6.4 Overview cycle load

The tables contain the necessary time for the transmission from the SPS to the building block.

CPU928B

Action	Transfer Length in Byte	Cycle load in ms
SEND	10	4
	50	10
	100	18
	128	22
RECEIVE	10	4
	50	14
	100	19
	128	24

CPU943B

Action	Transfer Length in Byte	Cycle Load in ms
ASCII, STX/ETX	10	8
	50	35
	100	70
	200	90
3964R SEND	10	17
	50	44
	100	79
	128	99
3964R RECEIVE	10	20
	50	65
	100	120
	128	160

CPU944

Action	Transfer Length in Byte	Cyle Load in ms
SEND	10	3
	50	9
	100	15
	128	18
RECEIVE	10	3
	50	10
	100	18
	128	24

CPU945

Action	Transfer Length in Byte	Cycle Load in ms
ASCII, STX/ETX	10	0,15
	50	0,6
	100	1,25
	200	1,6
3964R SEND	10	3,1
	50	4,3
	100	5,7
	128	6,5
3964R RECEIVE	10	2
	50	3,2
	100	5
	128	5,8

CPU948

Action	Transfer Length in Byte	Cycle Load in ms
ASCII, STX/ETX	10	0,24
	50	1
	100	2,1
	200	4,1
3964R SEND	10	4,7
	50	5,8
	100	7
	128	7,8
3964R RECEIVE	10	0,8
	50	2
	100	3,5
	128	4,2

Appendix

Appendix

A List of figures

B Index of tables

C Index

M. Stich