CL400 / CL500

R500 Computer Interface Module
Module Description

Edition 102

CL400 / CL500

R500 Computer Interface Module
Module Description

1070 072 131-102 (96.08) GB

© 1992-1996

by Robert Bosch GmbH, Erbach / Germany
All rights reserved, including applications for protective rights.
Reproduction or distribution by any means subject to our prior written permission.

Discretionary charge 20,- DM

1 Safety Instructions

Before you start working with the module / software, we recommend that you thoroughly familiarize yourself with the contents of this manual. Keep this manual in a place where it is always accessible to all users.

1.1 Proper use

This instruction manual presents a comprehensive set of instructions and information required for the standard operation of the described products.

The products described hereunder

- were developed, manufactured, tested and documented in accordance with the relevant safety standards. In standard operation, and provided that the specifications and safety instructions relating to the project phase, installation and correct operation of the product are followed, there should arise no risk of danger to personnel or property.
- are certified to be in full compliance with the requirements of the
  - COUNCIL DIRECTIVE 89/336/EEC of May 3rd 1989 on the approximation of the laws of the Member States relating to electromagnetic compatibility, 93/68/EEC (amendments of Directives), and 93/44/EEC (relating to machinery)
  - COUNCIL DIRECTIVE 73/23/EEC (electrical equipment designed for use within certain voltage limits)
  - Harmonized standards EN 50081–2 and EN 50082–2
- are designed for operation in an industrial environment (Class A emissions). The following restrictions apply:
  - No direct connection to the public low–voltage power supply is permitted.
  - Connection to the medium and/or high–voltage system must be provided via transformer.

  The following applies for application within a personal residence, in business areas, on retail premises or in a small–industry setting:
  - Installation in a control cabinet or housing with high shield attenuation.
  - Cables that exit the screened area must be provided with filtering or screening measures.
  - The user will be required to obtain a single operating license issued by the appropriate national authority or approval body. In Germany, this is the Federal Institute for Posts and Telecommunications, and/or its local branch offices.

⇒ **This is a Class A device. In a residential area, this device may cause radio interference. In such case, the user may be required to introduce suitable countermeasures, and to bear the cost of the same.**

Proper transport, handling and storage, placement and installation of the product are indispensable prerequisites for its subsequent flawless service and safe operation.

1.2 Qualified personnel

This instruction manual is designed for specially trained personnel. The relevant requirements are based on the job specifications as outlined by the ZVEI and VDMA professional associations in Germany. Please refer to the following German–Language publication:
Weiterbildung in der Automatisierungstechnik
Publishers: ZVEI and VDMA Maschinenbau Verlag
Postfach 71 08 64
60498 Frankfurt/Germany

Interventions in the hardware and software of our products not described in this instruction manual may only be performed by our skilled personnel.

Unqualified interventions in the hardware or software or non–compliance with the warnings listed in this instruction manual or indicated on the product may result in serious personal injury or damage to property.

Installation and maintenance of the products described hereunder is the exclusive domain of trained electricians as per IEV 826–09–01 (modified) who are familiar with the contents of this manual.

Trained electricians are persons of whom the following is true:

- They are capable, due to their professional training, skills and expertise, and based upon their knowledge of and familiarity with applicable technical standards, of assessing the work to be carried out, and of recognizing possible dangers.
- They possess, subsequent to several years' experience in a comparable field of endeavour, a level of knowledge and skills that may be deemed commensurate with that attainable in the course of a formal professional education.

With regard to the foregoing, please read the information about our comprehensive training program. The professional staff at our training centre will be pleased to provide detailed information. You may contact the centre by telephone at (+49) 6062 78–258.

1.3 Safety markings on components

DANGER! High voltage!

DANGER! Corrosive battery acid!

CAUTION! Electrostatically sensitive components!

Disconnect mains power before opening!

Lug for connecting PE conductor only!

Functional earthing or low–noise earth only!

Screened conductor only!

1.4 Safety instructions in this manual

DANGEROUS ELECTRICAL VOLTAGE
This symbol warns of the presence of a **dangerous electrical voltage**. Insufficient of lacking compliance with this warning can result in **personal injury.**

DANGER
This symbol is used wherever insufficient or lacking observance of this instruction can result in **personal injury.**

CAUTION
This symbol is used wherever insufficient or lacking observance of instructions can result in **damage to equipment or data files.**

⟹ This symbol is used to alert the user to an item of special interest.

1.5 Safety instructions for the described product

DANGER
Fatal injury hazard through ineffective Emergency–OFF devices! Emergency–OFF safety devices must remain effective and accessible during all operating modes of the system. The release of functional locks imposed by Emergency–OFF devices must never be allowed to cause an uncontrolled system restart! Before restoring power to the system, test the Emergency–OFF sequence!

DANGER
Danger to persons and equipment!
Test every new program before operating the system!

DANGER
Retrofits or modifications may interfere with the safety of the products described hereunder!

The consequences may be severe personal injury or damage to equipment or the environment. Therefore, any system retrofitting or modification utilizing equipment components from other manufacturers will require express approval by Bosch.

DANGEROUS ELECTRICAL VOLTAGE
Unless described otherwise, maintenance procedures must always be carried out only while the system is isolated from the power supply. During this process, the system must be blocked to prevent an unauthorized or inadvertent restart.

If measuring or testing procedures must be carried out on the active system, these must be carried out by trained electricians.

CAUTION
Danger to the module!
Do not insert or remove the module while the controller is switched ON! This may destroy the module. Prior to inserting or removing the module, switch OFF or remove the power supply module of the controller, external power supply and signal voltage!

CAUTION
Only Bosch–approved spare parts may be used!

CAUTION
Danger to the module!
All ESD protection measures must be observed when using the module! Prevent electrostatic discharges!

Observe the following protective measures for electrostatically endangered modules (EEM)!

- The Employees responsible for storage, transport and handling must be trained in ESD protection.
- EEMs must be stored and transported in the protective packaging specified.
- Out of principle, EEMs may be handled only at special ESD work stations equipped for this particular purpose.
- Employees, work surfaces and all devices and tools that could come into contact with EEMs must be on the same potential (e.g. earthed).
- An approved earthing wrist strap must be worn. It must be connected to the work surface via a cable with integrated 1 MW resistor.
- EEMs may under no circumstances come into contact with objects susceptible to accumulating an electrostatic charge. Most items made of plastic belong to this category.
- When installing EEMs in or removing them from an electronic device, the power supply of the device must be switched OFF.

1.6 Trademarks

All trademarks referring to software that is installed on Bosch products when shipped from the factory represent the property of their respective owners.

At the time of shipment from the factory, all installed software is protected by copyright. Software may therefore be duplicated only with the prior permission of the respective manufacturer or copyright owner.

MS–DOS® and Windows™ are registered trademarks of Microsoft Corporation.

2 Hardware Description

The R500 Computer interface module has two equivalent V.24/20 mA serial interfaces which enable the CL400/CL500 controller to be connected to additional Bosch controllers or other communicating devices.

2.1 Power Supply

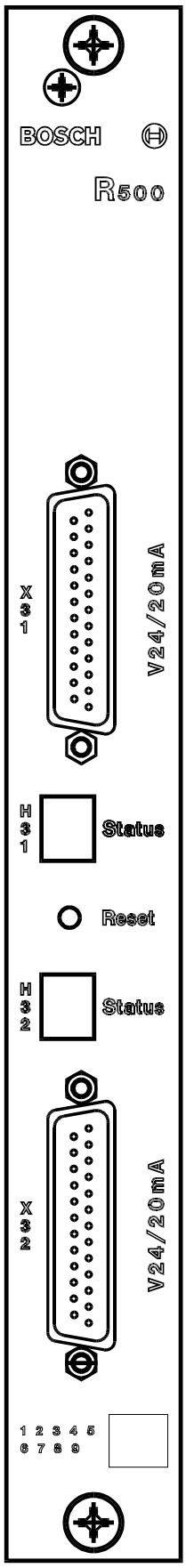
The R500 does NOT require an external power supply. All required operating voltages are taken from the power supply module of the CL400/CL500 controller.

2.2 Compatibility with R500P and R500M

The R500 hardware and software concepts enable the R500 to be operated in conjunction with the R500P and R500M-EN modules in the basic unit of the CL400/CL500.

In order to address the R500 Computer interface module from within the PLC program, special function modules must be integrated into the PLC program (please refer to Section **Fehler! Verweisquelle konnte nicht gefunden werden.**).

2.3 Front Panel



V.24/20 mA interface

Channel 1

Status display for X31 interface

Reset button

Status display for X32 interface

Channel 0

V.24/20 mA interface

Version ID / Labelling field

2.3.1 Status Display

There is a status display for each of the two channels on the R500 Computer interface module.

The H31 7-segment display is linked to Channel 1.

The H32 7-segment display is linked to Channel 0.

2.3.2 Interface Connectors

The X31 and X32 interface connectors comprise the V.24/20 mA interfaces (please refer to Page 2-5). These interfaces are used for point-to-point connection to an external device.

2.3.3 Reset Button

The R500 operating software is re-initialised by pressing the Reset button.

- All active jobs are cancelled irrespective of status.
- All jobs are deleted from the job queue.
- STOP and CLAB module requests are cancelled.

2.3.4 Version ID

The version identification is used to identify the individual release versions. Each version modification is identified by a white dot.

2.3.5 Labelling Field

The labelling field enables the user to mark the module exterior with a system-specific designation. This could be the block address of the module, for example.

2.4 CL400/CL500 Slot Assignments

A maximum of 5 R500 Computer interface modules can be operated in one PLC.

In GG2, slots 4 through 14 are assigned to the CL400.

In GG2/K, slots 4 through 8 are assigned.

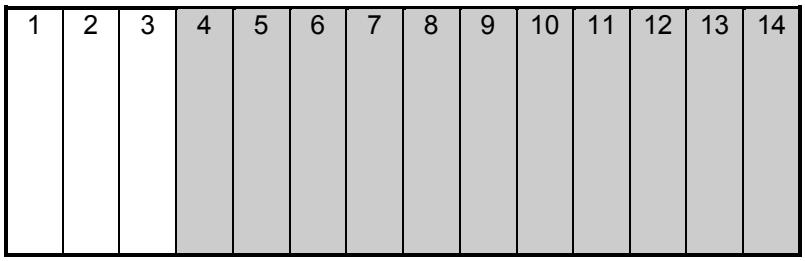


Fig. 2-1 CL400 — Slot assignment, GG2 Basic unit

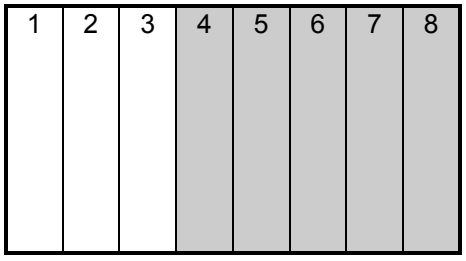


Fig. 2-2 CL400 — Slot assignment, GG2/K Basic unit

In GG2, slots 5 through 14 are assigned to the CL500.

In GG2/K, slots 5 through 8 are assigned.

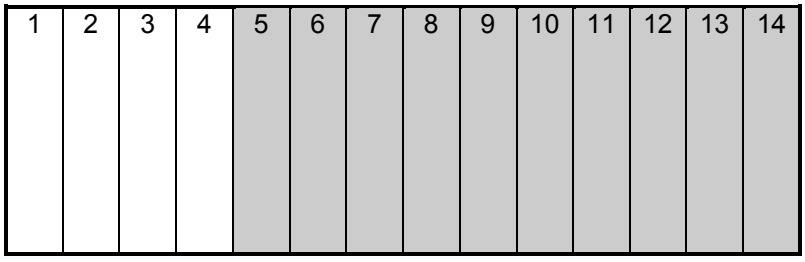


Fig. 2-3 CL500 — Slot assignment, GG2 Basic unit

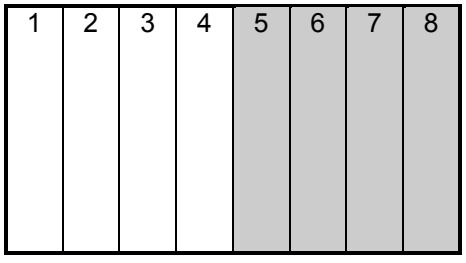


Fig. 2-4 CL500 — Slot assignment, GG2/K Basic unit

2.5 V.24/20 mA Interface

The V.24/20 mA interface is available to the user on Channel 0 and Channel 1. Devices which also have a V.24 or 20 mA interface can be connected to the V.24/20 mA interface.

The following protocols are available to the user:

- BÜP19E (Bosch standard protocol for secure data transmission)
- BÜP03E (Bosch transmission protocol for free configuration)
- BÜP64 (Siemens 3964R transmission protocol)

2.6 DIP Switch Settings on the R500 Computer Interface Module

All DIP switch settings must be completed prior to start-up of the R500 module.

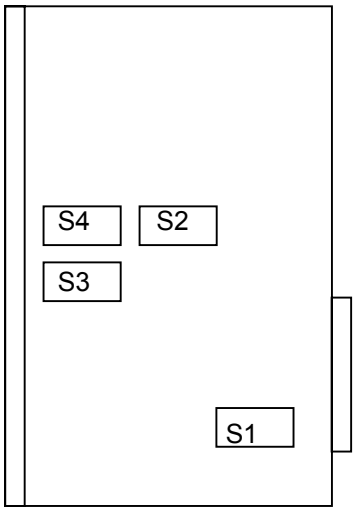
Warning:

When handling the R500 Computer interface module, all necessary precautions must be taken in order to protect against ESD.

Prevent electrostatic discharge.

DIP Switch	Function
S1	Setting the block address
S2	Protocol-specific settings
S3	Selecting transmission format, character length, transmission speed of the serial transmission, and control signals for the V.24/20 mA interface, Channel 0
S4	Selecting transmission format, character length, transmission speed of the serial transmission, and control signals for the V.24/20 mA interface, Channel 1

Fig. 2-5 R500 — DIP Switch settings



2.6.1 R500 Block Address

The R500 occupies a block on the CL400/CL500 system bus.

The R500 block address is selected by means of the S1 DIP switch.

The setting procedure is similar to that used with all other modules belonging to the CL400/CL500 PLC series.

2.6.2 Protocol-specific DIP Switch Settings

Protocol-specific settings are implemented on the S2 DIP switch. For this purpose, switch segments SS1 through SS4 have been assigned to Channel 0 and switches SS5 through SS8 to Channel 1.

- BÜP19E protocol, refer to Page 7-2
- BÜP64 protocol, refer to Page 9-2

2.6.3 V.24/20 mA Interface Control Signals

The control signals (DTR/DSR) for the V.24/20 mA interface can be queried or not queried by setting the SS8 switch segment SS8 of the S3 and S4 DIP switches.

NOTE –

SS8 to ON - Control signal query enabled

SS8 to OFF - Control signal query disabled

2.6.4 V.24/20 mA Interface Transmission Rate

The transmission rate for the V.24/20 mA interface is set with switch segments SS5 through SS7 on the **S3** and **S4** DIP switches.

SS5	SS6	SS7	Transmission rate
ON	ON	ON	19200 bps
OFF	ON	ON	9600 bps
ON	OFF	ON	4800 bps
OFF	OFF	ON	2400 bps
ON	ON	OFF	1200 bps
OFF	ON	OFF	600 bps
ON	OFF	OFF	300 bps
OFF	OFF	OFF	110 bps

Fig. 2-6 V.24/20 mA Interface — Transmission rate DIP switch settings

2.6.5 V.24/20 mA InterfaceTransmission Format

The transmission format for the V.24/20 mA interface is selected with switch segments SS1 through SS4 on the **S3** and **S4** DIP switches.

Note the following settings:

SS1:

ON - 1 Stop bit

OFF - 2 Stop bits

SS2:

ON - 7 Data bits

OFF - 8 Data bits

SS3:

ON - Parity EVEN

OFF - Parity ODD

SS4:

ON - Parity

OFF - No parity

2.6.5.1 V.24 Interface Signal Levels and Pin Assignments

The signal levels and connector pin assignments correspond to the VDE Guidelines 2880, Page 2, for programmable logic controllers, process and data interfaces.

Signal level (data line):

logical 1 --> -15 V to -3 V
logical 0 --> +3 V to +15 V

Signal level (signalling and control line):

active --> +3 V to +15 V
passive --> -15 V to -3 V

Function	Designation	Pin no.	Signal direction
Shield	Shield	*	
Transmit Data, PLC	TxD	2	-->
Receive-data, PLC	RxD	3	<--
Signal ground	Signal-Ground	7	
Data Set Ready	DSR	6	<--
Data Terminal Ready	DTR	20	-->

Fig. 2-7 V.24 Interface — Pin assignment

* The shield is connected to the metal shell of the D-SUB connectors.

2.6.5.2 20 mA Interface Signal Levels and Pin Assignments

The signal levels and connector pin assignments correspond to the VDE Guidelines 2880, Page 2 for programmable logic controllers, process and data interfaces (limit: max. 27 V blocking voltage). Dependent upon the pin assignment, 20 mA interfaces enable active or passive operation, i.e., with or without power supply.

Line states: logical 1 --> 20 mA
logical 0 --> no current

20 mA active

Function	Designation	Pin no.	Signal direction
Shield	Shield	*	
Receive-data + PLC	RxD +	12	<--
Receive-data - PLC	RxD -	24	
Transmit Data + PLC	TxD +	13	-->
Transmit Data - PLC	TxD -	25	
Data Set Ready +	DSR +	14	<--
Data Set Ready -	DSR -	18	
Reader Control +	RDRCTL +	16	-->
Reader Control -	RDRCTL -	21	

Fig. 2-8 20 mA Interface — Pin assignment, active operation

* The shield is connected to the metallic shell of the D-SUB connectors.

Pins 9 (12 V in) and 10 (12 V out) must be jumpered for active mode.

20 mA passive

Function	Designation	Pin no.	Signal direction
Shield	Shield	*	
Receive-data + PLC	RxD +	22	<--
Receive-data - PLC	RxD -	12	
Transmit Data + PLC	TxD +	23	-->
Transmit Data - PLC	TxD -	13	
Data Set Ready +	DSR +	11	<--
Data Set Ready -	DSR -	14	
Reader Control +	RDRCTL +	19	-->
Reader Control -	RDRCTL -	16	

Fig. 2-9 20 mA Interface — Pin assignment, passive operation

* The shield is connected to the metallic shell of the D-SUB connectors.

2.6.6 V.24/20 mA Interface Cable Length Specifications

The maximum permitted interface cable length depends on the baud rate The following length specifications apply to the twisted-pair, shielded 14 x 0.14 Bosch cable, Bosch no. 910152:

Transmission rate	V.24	20 mA
Baud rate (bps)	[m]	[m]
110...4800	300	350
9600	300	300
19200	100	150

Fig. 2-10 V.24/20 mA Interface — Cable lengths

It is assumed that there are only slight differences in potential (-2 V < V_{diff} < +2 V) between sending and receiving device (applies only to the V.24 interface).
It is also assumed that the interface modules of the partner device are of a quality that is at least equal to that of the R500 computer interface module.
Do NOT route data transmission lines parallel and adjacent to power lines.

2.7 Status Displays

There is a status display for each channel of the R500 computer interface module. Channel-specific statuses and fault messages are displayed.

The H31 seven-segment display is connected to the X31 interface and the H32 to the X32 interface.

2.7.1 Channel-specific Displays

These messages are directly allocated to the corresponding channel and consequently to this protocol.

Code	Explanation
.	Centralised job is being processed. The centralised job is displayed by activating the decimal point.
P	Protocol fault, - Fault while programming the protocol memory - no protocol loaded
0..F	Job number (hex.) of a defective job

2.7.2 Channel-independent Displays

These messages concern the entire module and appear either on the lower display or on both displays.

Code	Explanation
H	Hardware fault
P flashes	Flash memory is being loaded by the PG utility program

2.8 Protocol Loader

The transfer protocols are loaded onto the R500 or R500P by means of the PROFI software.

The protocol loader for the R500 or R500P module is located in the loader under the Configuration menu option.

The protocols are loaded into the R500 module by the SK500 or the ZS400.

The protocols must be filed in the BOSCH.BIB subdirectory.

2.9 R500 Specifications

Specifications	R500
Interfaces	V.24/20 mA according to VDE 2880 Page 2 both interfaces are isolated
Baud rates	110 to 19200 bps V.24/20 mA interface
Power consumption from 5 V voltage supply	430 mA typically 690 mA maximum
Power consumption from 5 V (Vp)	typically 7 mA (operation)
Power consumption from 3.5 V (Vp)	typically 2 mA (standby)
Power consumption from +12V- ISO voltage supply	140 mA + 20 mA per active current loop
Power consumption from -12V- ISO voltage supply	20 mA typically
Resistance to line transient interference of mains supply	as per DIN EN 61131-2
Max. number of modules per controller	5
Ambient temperature range	0 to +55 °C
Storage temperature range	-25 to +70 °C
Modular width	single
Weight	590 g

Fig. 2-11 R500 — Specifications

3 PLC Interface

The PLC interface provides access to the communications system and requires the following function modules:

- R5INIT
- R5REQ
- R5CON

The PLC interface can issue jobs to the R500 (client characteristics of the CL400/CL500).

The function modules do not monitor the jobs (time-out). Monitoring is implemented in the R500 firmware.

NOTE –

If the R500 is operated only as a server, function modules are NOT required.

4 R5INIT Initialisation Function Module

4.1 Characteristic Data

Characteristic data	R5INIT
Module name	R5INIT
Block length	871 words
Call-up length	7 words
Number of parameters	6
Used markers	M230 to M254

Fig. 4-1 R5INIT — Characteristic data

The **R5INIT initialisation** function module creates various tables in the ZS and on the computer interface module and fills them with information which is re-accessed when these modules are communicating.

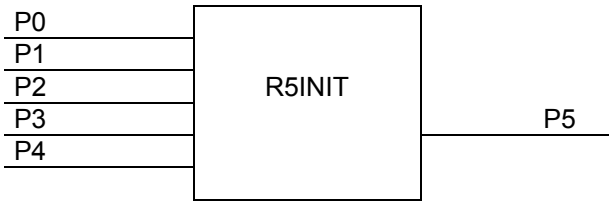
When the controller has been switched on, the program cycle should and may only run once on this module before communication can occur via the computer interface. The module is therefore usually called up in the OM5 and/or OM7 start-up organisation module.

NOTE –

If the R500 is operated only as a server, the R5INIT module is NOT required.

4.2 I/O Parameters

The R5INIT has 5 input parameters and one output parameter.



4.2.1 Input Parameters

Parameter	Att.	Description
P0	W	Module number of computer interface 0
P1	W	Module number of computer interface 1
P2	W	Module number of computer interface 2
P3	W	Module number of computer interface 3
P4	W	Module number of computer interface 4

Fig. 4-2 R5INIT — Input parameters

The module number corresponds to the line number of the entry in the system table (not the block address).

The assignment of the computer interface number (CXN) to the module number is freely selectable.

For example, if the module number is transferred to the P2 parameter, the CXN = 2.

If the module numbers for the computer interfaces are modified in the system configuration table (SC table), the R5INIT parameters must be modified accordingly. The value FFFFH must be transferred as a parameter for modules which are not in the system.

As this module may be called up only once in each ZS central processing unit, one call-up initialises **all** computer interface modules (R500, R500P and R500M-EN) which are meant to execute the data communication jobs of the relevant ZS.

4.2.2 Output Parameter

Parameter	Att.	Description
P5	W	Result

Fig. 4-3 R5INIT — Output parameter

This parameter provides the PLC program with the result of the initialisation and signals any faults.

Code	Explanation
0000H	Initialisation ended without fault
0001H	No CXN matching Parameter 0 available
0002H	No CXN matching Parameter 1 available
0004H	No CXN matching Parameter 2 available
0008H	No CXN matching Parameter 3 available
0010H	No CXN matching Parameter 4 available
0020H	No free memory for memory management list
0021H	No data-field address in the memory management list
0030H	Communications fault on the CXN
0031H	Communications fault on the ZS

Fig. 4-4 R5INIT — Error codes

5 R5REQ Function Module

5.1 Characteristic Data

Characteristic data	R5REQ
Module name	R5REQ
Block length	1051 words
Call-up length	7 words
Number of parameters	6
Used markers	M230 to M254

Fig. 5-1 R5REQ — Characteristic data

.

5.2 I/O Parameters

The R5REQ function module has four input and two output parameters.

5.2.1 Input Parameters

Parameter	Att.	Description
P0	W	Data type

This parameter specifies the type of data range. A data area of 16 words is reserved for each job for the protocol-specific parameters. Data modules (44H), the data buffer (64H) and the data field (43H) can be used as data ranges. The user must always reserve 16 words per job for the parameters which are specific to the job.

Parameter	Att.	Description
P1	W	Data type attribute

If P0 specifies a data module as a data area, P1 indicates the data-module number. In other cases P1 is meaningless and should be occupied with FFFFH.

Permitted values are 0 - 255 and FFFFH.

Parameter	Att.	Description
P2	W	Byte offset

The start address for the job parameters is specified in the selected data area.

Parameter	Att.	Description
P3	W	Job bit array JBA

Each of the 16 bits of this parameter refers to a block of job parameters in the data range specified by P0 through P2. Comprising, as it were, a binary control panel with 16 switches, it has been termed *job bit array* (JBA). Bit 0 to P3 therefore refers to the 0 job parameter block, and Bit 15 refers to the 15th job parameter block in the specified data area.
A bit which is set in P3 denotes the start of the job specified in the associated job parameter block.
If several bits are set at the same time, the associated jobs are transferred to the R500 in the same cycle.
When the program cycle has run once on the R5REQ with the set job bits, the job has completely started. The job bits must then again be reset via the respective application program.

5.2.2 Output Parameters

Parameter	Att.	Description
P4	W	Fault message

Basic status and error codes, which the R5REQ identifies when jobs are issued, are entered in this parameter:

- The LOW byte in the P4 contains the status or error code,
- The HIGH byte contains a coded recommendation concerning a reaction, or *response*, to the fault message.

Error code in the LOW byte::

Code	Explanation
00H	No fault, correct call-up
01H	The P3 parameter JBA (job bit array) is zero
02H	The specified data module does not exist
03H	The specified data area is too short
04H	The specified byte offset is invalid
06H	Wrong SFL, unsuitable ZS version
07H	The R500 has not been entered in the SC table
08H	R500 has wrong block address

PLC response in the HIGH byte::

Code	Explanation
00H	No fault, correct call-up
01H	Repeat call-up of the R5REQ
02H	Modify PLC program
03H	Modify protocol parameters
04H	The program cycle must be rerun on the R5INIT function module
05H	The program cycle must be rerun on the R5REQ function module
06H	The program cycle must be rerun on the R5CON function module
07H	Reload the protocol

If the R5REQ detects additional fault messages on job start-up, it writes them into the first word of the associated job block.

Parameter	Attr.	Description
P5	W	Result bit array RBA

This parameter indicates to the user whether any faults have been identified when the jobs were issued. A job is assigned to each result bit. The assignment corresponds to the P3 for the input parameter. A set result bit indicates that the job which is addressed with the same bit number in P3 could not be issued. In addition, a detailed error code is output in the first word of the associated job block, thereby completing the *result bit array*.

5.3 Explanations of Parameters in Data Range

A parameter field of 16 words must be reserved for each job in the data field DF, data buffer DB or data module DM.

The address and offset of this data range are specified with the P0 - P2 parameters for the R5REQ module.

A bit which starts with Bit 0 (zero) is assigned to each parameter field in the job bit array JBA and in the result bit array RBA.

Example of associated data area when the P2 parameter (offset) has the value 0:

Job bit of the JBA	Associated data area	Byte addresses
0	DW0 - DW15	0 - 31
1	DW16 - DW31	32 - 63
2	DW32 - DW47	64 - 95
3	DW48 - DW63	96 - 127
4	DW64 - DW79	128 - 159
5	DW80 - DW95	160 - 191
6	DW96 - DW111	192 - 223
7	DW112 - DW127	224 - 255
8	DW128 - DW143	256 - 287
9	DW144 - DW159	288 - 319
10	DW160 - DW175	320 - 351
11	DW176 - DW191	352 - 383
12	DW192 - DW207	384 - 415
13	DW208 - DW223	416 - 447
14	DW224 - DW239	448 - 479
15	DW240 - DW255	480 - 511

Fig. 5-2 R5REQ — List of parameters

The max. 16 job parameters are divided into two groups.

1. Protocol-independent parameters:

Their content is always the same, irrespective of the protocol and/or service which is actuated by the subsequent parameters.

2. Protocol-dependent parameters:

These parameters differ for all protocols and/or services.

5.3.1 Protocol-independent Parameters

Data word	Content	
	HIGH byte	LOW byte
DW0	PLC response	Error code
DW1	CXN	Channel number
DW2	Job number	

Fig. 5-3 R5REQ — Protocol-independent parameters

DW0

The 0 data word returns the result of the job. Whenever the R5REQ module is called up, the result of the job is transferred to the module.

The result word is divided into LOW byte (error codes), and HIGH byte (PLC response). The error code and PLC response tables are listed below.

Error code (LOW byte)

Code	Explanation
00H	Job transferred without fault to CXN
01H	Job is already being processed
02H	Job is in the queue
03H	Job acknowledged but not yet confirmed
04H	Wrong channel number
05H	Wrong CXN
06H	Wrong job number
07H	Job cannot be terminated
08H	No free job number on the CXN
10H	CXN 0 not initialised
11H	CXN 1 not initialised
12H	CXN 2 not initialised
13H	CXN 3 not initialised
14H	CXN 4 not initialised
20H	CXN 0 not entered in SC table
21H	CXN 1 not entered in SC table
22H	CXN 2 not entered in SC table
23H	CXN 3 not entered in SC table
24H	CXN 4 not entered in SC table
30H	Communications fault on the CXN
31H	Communications fault on the ZS
92H	Invalid command parameter
D0H	No protocol loaded for Channel 0 or Channel 1
D1H	Job queue is full

Fig. 5-4 R5REQ — Error codes

PLC response (HIGH byte)

Code	Explanation
00H	Job transferred without fault to CXN
01H	Call up R5REQ again
02H	Modify PLC program
03H	Modify protocol parameter
04H	Rerun program cycle on R5INIT
05H	Rerun program cycle on R5REQ
06H	Rerun program cycle on R5CON
07H	Reload V.24 protocol
08H	Check system configuration

Fig. 5-5 R5REQ — PLC response

DW1

This word indicates on which R500 and via which channel this job is to be processed.

The CXN is entered in the HIGH byte of this word. The CXN is specified by the R5INIT function module (see Page 4-2).

The channel number indicates via which module interface the job is to be processed.

DW2

Up to 16 jobs can be started in parallel with the computer interfaces of the CL400/CL500. These jobs can be differentiated by the job-number information. Another job with the same job number cannot be started until the previous one has ended. An attempt to run two jobs with the same job number will be rejected by the R5REQ with a fault message.

However, several centralised jobs which have different job numbers in one PLC cycle can be started.

The R500 can process up to 16 jobs in parallel. The R500 has 16 job channels for this purpose. The job numbers 0 to 15 are assigned directly to the job channels. Jobs with larger job numbers search for the next free job channel on the R500. Therefore, the job numbers from 0 to 15 should not be mixed with larger job numbers, otherwise jobs may be rejected despite free job channels.

The job numbers can accept a value from **0 to 7FFFH**.

Jobs which have been transferred to the computer interface and are waiting to be processed are deleted from the job queue by the 16th bit of the job number.

The job numbers 12 to 15 are reserved for high-priority jobs. The information on this parameter ensures that this job is given a high priority and is processed accordingly.

5.3.2 Protocol-dependent Parameters

The protocol-dependent parameters are described in Sections BÜP19E (Page **Fehler! Textmarke nicht definiert.**), BÜP03E (Page **Fehler! Textmarke nicht definiert.**) and BÜP64 (Page **Fehler! Textmarke nicht definiert.**).

6 R5CON Confirmation Module

6.1 Characteristic Data

Characteristic data	R5CON
Module name	R5CON
Block length	473 words
Call-up length	6 words
Number of parameters	5
Used markers	M230 to M254

Fig. 6-1 R5CON — Characteristic data

The **R5CON Confirmation** module monitors the centralised jobs. Its function is to return information to the user while the job is processing and after the job has ended. The module retrieves the status and the fault word from the job table and transfers this information to the address which the user indicated as a parameter.

The R5CON confirmation module can be used to acknowledge either only one job per call-up or all fault-free jobs and one defective job. The R5CON retrieves the status and the fault word from the job table and transfers them to the PLC program. However, if all fault-free jobs are acknowledged, the status and the fault word contain only the information which relates to the one permitted defective job.

The R5CON should acknowledge all jobs issued by the R5REQ; the results of the individual jobs are read out. A local acknowledgement is generated for unconfirmed services.

A new job cannot be started with this number until the acknowledgement has been read out.

6.2 I/O Parameters

The R5CON confirmation module has two input and three output parameters.

6.2.1 Input Parameters

Parameter	Att.	Description
P0	W	CXN and channel number

This parameter indicates the CXN and the number of the channel on which the corresponding job has been processed.

Parameter	Att.	Description
P1	W	Job number (0 to 7FFFH, FFFEH or FFFFH)

Permitted entries are:

- Job numbers from 0 to 7FFFH or
- the FFFFH and FFFEH wildcard characters

Using the information supplied by a job number 0 to 7FFFH, the user selects precisely that job which he would like to acknowledge. A specific job is therefore checked systematically.
The indicated job number is mirrored in the P2 output parameter.

If the FFFEH wildcard character is indicated as a job number, the R5CON function module acknowledges all the jobs (from the interface named by P0) which ended without fault up to this time.
If jobs also ended with faults, the R5CON also acknowledges the defective job which has the highest job number and returns its status and error codes in the output parameters. Any other defective jobs are not acknowledged by this R5CON call-up.
If the job has ended, the value FFFFH is signalled back to the P2 output parameter.
If only fault-free jobs have ended, these are all acknowledged and the value FFFEH is signalled back to the P2 output parameter.

If the FFFFH wildcard character is indicated in the P1 parameter, the R5CON acknowledges the job which has been processed by the R500, although this acknowledgement has still not been retrieved. If this applies to several jobs, the job with the highest job number is acknowledged. This job number is then entered in the P2 output parameter. All other jobs are not acknowledged.

6.2.2 Output Parameters

Parameter	Att.	Description
P2	W	Job number

In this parameter the R5CON writes the job number of the job whose status and error codes are in the P3 and P4 output parameters.
If an explicit job number was indicated in P1, this job number is mirrored in the P2.
If no job has ended with P1 = FFFFH according to the parameterisation, the value FFFFH is signalled back to P2 as a job number.
If there is no defective job according to parameterisation of the P1 = FFFEH, however one or more jobs ended without a fault message, the value FFFEH is signalled back to the P2.

Parameter	Att.	Description
P3	W	Status

The current status of the job is output to this word for every program cycle of the R5CON. The current status of the job is indicated in the HIGH byte of this word.

High byte	Explanation
00H	Job ended without fault (concluded)
01H	Job being processed on the CXN
02H	Job is in the CXN queue
03H	Job ended with fault(s)
04H	No job with this job number being processed

Fig. 6-2 R5CON — HIGH Byte status

The P3 LOW byte supplies a code which aids the user in implementing a suitable procedure should a fault occur (PLC response). This code is not valid until the job has ended.

LOW byte	Explanation
00H	Fault-free operation
01H	Repeat call-up of the R5CON function module
02H	Modify PLC program
03H	Check and modify job parameters
04H	Rerun R5INIT program cycle function module
05H	Rerun R5REQ program cycle function module
06H	Reserved
07H	Reload V.24 protocol
08H	Check project-phase data (configuration)
09H	Repeat job
0AH	Check call-up parameter for the R5CON function module
10H	Repeat job; job was terminated by the PLC program or via Reset button
20H	Job fault-free, however it had reduced data length when read
30H	Reconnect to Initiate Service
40H	Reset module, restart required
50H	Undefined error code, no response possible

Fig. 6-3 R5CON — LOW Byte status

Parameter	Att.	Description
P4	W	Fault message

If a fault has occurred, i.e., P3 HIGH byte = 03, the detailed error code can be found in P4. These codes require evaluation usually only during start-up or when service procedures are being implemented.

However, the fault message is deleted after one PLC cycle.

7 BUEP19E Protocol

7.1 Introduction

The BÜP19E transfer protocol *(BOSCH Übertragungs-Protokoll)* can be loaded into the R500 and R500P computer interface modules. It transfers data by means of a point-to-point connection.

As a result, data can be exchanged between two computer interface modules or one computer interface module and a central processing unit or an SK500 module.

Data can also be exchanged between a computer interface module and a computer.

The ‘PG Protocol Loader' loads the BÜP19E transfer protocol into Channel 0 of the R500P interface module. On the R500 Computer interface module, the protocol can either be loaded into Channel 0, Channel 1 or simultaneously into both channels. As a result, the user is presented with flexible options regarding the use of the BÜP19E protocol on the computer interface module.

The RST and PST abbreviations occur frequently in this description and are defined as follows:

RST - Requesting station

Specifies the communications partner which starts the protocol job.

PST - Peripheral station

Specifies the communications partner which responds to a received protocol job.

7.2 Protocol-specific Settings

The S2 DIP switch for the BÜP19E protocol has the following function:

Switch	Setting	Explanation	Channel
SS1	ON OFF	peripheral jobs have priority centralised jobs have priority	0
SS2	-	no function	0
SS3	-	no function	0
SS4	-	no function	0
SS5	ON OFF	peripheral jobs have priority centralised jobs have priority	1
SS6	-	no function	1
SS7	-	no function	1
SS8	-	no function	1

Fig. 7-1 BÜP19E — S2 DIP Switch settings

NOTE –

When installing the link, ensure that the low priority is assigned to one unit and the high priority is assigned to the other unit.

7.3 Protocol-specific Parameters

The BÜP19E uses protocol-specific parameters to parameterise the data traffic for RST and PST. Each parameter is explained below.

7.3.1 Command Type / Operand Type Parameters

Parameter word	**Contents**	
	HIGH byte	LOW byte
PW4	Type of command	Type of operand
Example	41H	01H
	Output	Word

Fig. 7-2 BÜP19E — Command type / Operand type

7.3.1.1 Command Type Parameter

This parameter indicates the types of commands for the job. There are two types of commands:

'E'(45H) Input into the RST

'A' (41H) Output from the RST

The direction of the jobs always refers to the RST, i.e., an output job (41H --> 'A') transfers data from the RST to the PST and an input job (45H --> 'E') transfers data from the PST to the RST.

If a job is started with a deviating type of command, the job is terminated and a corresponding fault message is returned.

7.3.1.2 Operand Type Parameter

The LOW byte of the parameter specifies the types of operands for the job. Depending on the RST command code, eight different types of operands can be used:

Code	Data type	Number in bytes
00H	Byte	1
01H	Word	2
02H	Double word	4
03H	Quad word	8
04H	Byte mask	2
05H	Word mask	4
06H	Double word mask	8
07H	Quad word mask	16

Fig. 7-3 BÜP19E — Operand type parameter

When selecting the operands, check whether they are also supported by the PST.

7.3.2 RST / PST Command Code Parameter

Parameter word	Contents	
	High byte	LOW byte
PW5	RST command code	PST command code
Example	44H	44H
	Data module	Data module

Fig. 7-4 BÜP19E — RST / PST Command code

In this parameter the RST command code (hex code of the command) is indicated in the HIGH byte and the PST command code is indicated in the LOW byte.

7.3.3 RST Command Attribute

Parameter word	Contents
PW6	RST command attribute
Example	17H
	Data module 23

Fig. 7-5 BÜP19E — RST Command attribute

7.3.4 RST Address — High Part

Parameter word	Contents
PW7	RST address, High part
Example	10H
	ZS 2

Fig. 7-6 BÜP19E — RST Address, High part

The block address of the module, to which the RST command refers, is indicated in the PW7 parameter word.

7.3.5 RST Address — Low Part

Parameter word	Contents
PW8	RST address, Low part
Example	32H
	50 bytes

Fig. 7-7 BÜP19E — RST Address, Low part

The byte address (offset) of the utilised command is indicated in the low part of the RST address.

7.3.6 PST Command Attribute

Parameter word	Contents
PW9	PST command attribute
Example	AH
	Data module 10

Fig. 7-8 BÜP19E — PST Command attribute

The field index (for field commands) or the control code (for special commands) of the PST command is indicated in the PST command attribute.

Refer also to RST command attribute.

7.3.7 PST Address — High Part

Parameter word	Contents
PW10	PST address, High part
Example	F0H
	ZS400

Fig. 7-9 BÜP19E — PST Address, High part

The block address of the module, to which the PST command refers, is indicated in the parameter.

7.3.8 PST Address — Low Part

Parameter word	Contents
PW11	PST address, Low part
Example	46H
	70 bytes

Fig. 7-10 BÜP19E — PST Address, Low part

The byte address (offset) of the utilised command is indicated in the low part of the PST address.

7.3.9 Data Volume Parameter

Parameter word	Contents
PW12	Volume of data (in data types) according to type of operand
Example	28H
	Volume = 40

Fig. 7-11 BÜP19E — Data volume parameter

The volume of data to be transferred, referring to the type of operand, is indicated in the parameter.

If the 'Mask' type of operand is used, the volume of data should be set to 1. The double length (set mask and reset mask) of the type (e.g. word mask) indicated in the mask identification is then always transferred in the sequence low part, high part.

Example:

Data type	PW11	Number in bytes
Byte	3	3
Word	7	14
Double word	2	8
Quad word	5	40
Byte mask	1	2
Word mask	1	4
Double word mask	1	8
Quad word mask	1	16

7.3.10 Coordination Parameter

Parameter word	Contents	
	High byte	LOW byte
PW13	Coordination sequence point	Field coordination marker
Example	02H	FFH
	I/O status	No coordination marker

Fig. 7-12 BÜP19E — Coordination parameter

The coordination parameter consists of a field coordination marker (LOW byte) and a coordination sequence point (HIGH byte).

The function of the coordination markers is to coordinate the implementation of a command on the PST. The command is not implemented until the coordination event is "true"; as far as the CL500 control type is concerned, the coordination sequence point must be accessed or the field coordination marker must be set to "1"; for the other specified control types the addressed coordination marker must be "1" before a coordinated command can be implemented.

CL500 controller acting as PST

The Central Processing Unit Control command in the CL500 system uses two types of coordination markers:

- Process coordination marker (PCM)

and

- from ZS501: Field coordination marker (FCM)

The PCMs are specified as follows:

00 H	System STOP status
01 H	System RUN status
02 H	I/O status
03 H	I/O status or STOP
04 H	EP
05 H	EP or STOP
06 H	OM1
07 H	OM1 or STOP
0F H	No process coordination

The FCMs are specified in the special marker area as follows:

- SM16.0 .. SM17.7 single FCM;
  these FCMs are reset by the system
  after the coordinated command
  has been implemented;
- SM18.0 .. SM19.7 permanent FCMs;
  are <u>not</u> reset by the system after the
  coordinated command has been
  implemented;

The FCMs are coded as follows:

SM16.0 .. SM 17.7 --------->	FCM	0 H ..	F H
SM18.0 .. SM 19.7 --------->	FCM	10 H ..	1F H
No field coordination ---->	FCM	FF H	

7.4 Command Description for Central Processing Units

The central processing units differentiate between field commands and special commands.

7.4.1 Field Commands

7.4.1.1 Data Field — 43H

Command	Data field
Command code	43H
Command attribute	FFFFH
Address, High part	Block address
Address, Low part	Byte address 0 ... 24k
Operand type	All

7.4.1.2 Data Module without Header — 44H

Command	Data module without header
Command code	44H
Command attribute	DM number 0 ... 255
Address, high part	Block address
Address low part	Byte address 0 ... 511
Type of operand	All

7.4.1.3 Marker — 4DH

Command	Marker
Command code	4DH
Command attribute	0
Address, high part	Block address
Address, low part	Byte address 0 ... 255
Type of operand	All

7.4.1.4 Data Buffer — 64H

Command	Data buffer
Command code	64H
Command attribute	0
Address, high part	Block address
Address, low part	Byte address 0 ... 511
Type of operand	All

7.4.2 Special Commands

7.4.2.1 Identification Command — 76H

Command	Identification command
Command code	76H
Command attribute	Control code
Address, high part	Block address
Address, low part	0
Type of operand	0

Description of control codes

HIGH byte								LOW byte
7	6	5	4	3	2	1	0	Identification number

Identification number of different FFH number

Bit 0 : 0 --> without identification code

1 --> with identification code

Bit 3 : 0 --> without protocol priority

1 --> with protocol priority

Bit 4 : 0 --> accept identification number

1 --> do not accept identification number

Bit 5 - 7 0 reserve

Description of identification code

Unit identifier (Word) - CL500 : 0000H

Type identifier (Word) - R500 : 2080H

SW version (Byte) - Binary code

HW version (Byte) - Binary code

Module block address (Word)

7.5 Fault Messages

The BÜP19E protocol generates fault information which is divided into three areas:

- PLC response (Byte)
- Fault class (Byte)
- Error code (Byte)

The fault information is filed in the R5CON check-back parameters in the following sequence:

P3 parameter

High byte	LOW byte
Status	PLC response

P4 parameter

High byte	LOW byte
Fault class	Error code

7.5.1 Job Status

The current status of the job is specified in the Status by the function module.

Status	Explanation
00H	Job ended without fault
01H	Job written to queue
02H	Job being processed
03H	Job ended with fault(s)

7.5.2 PLC Response

The PLC response should provide the user with information on further procedures.

The following conventions apply:

PLC response	Explanation
00H	No response, job fault-free
01H	Repeat FM job
02H	Program fault, change PLC program
03H	Change protocol parameter
04H	Rerun the program cycle on the R5INIT
05H	Rerun the program cycle on the R5REQ
06H	Rerun the program cycle on the R5CON
07H	Reload the protocol
10H	Active job terminated by the user
20H	Job fault-free, however with reduced data length

Fig. 7-13 BÜP19E — PLC Response

7.5.3 Error Class

The error class refers to the generation point of the fault information. The following conventions apply:

Error class	Explanation
00H	Fault-free
30H	System bus
31H	Operating system
32H	PST job implementation
33H	RST job implementation
34H	Protocol sequence
FFH	Fault messages from function modules

Fig. 7-14 BÜP19E — Error class

7.5.4 Error Catalogue

Fault information			Explanation
PLC response	Error class	Error code	
00	00	00	Fault-free job
01	34	05	Protocol cannot be implemented due to an internal protocol fault (RST) - No connection - Faulty connection - Interface hardware not ready - The incoming message cannot be analysed
01	30	02	Memory access fault (RST) Faults occurred when RST memory area was accessed. Data incorrectly read or incompletely written
01	30	03	Wrong memory type (RST). The type of access for the addressed RST memory is not permitted (e.g. writing to EPROM)
01	30	05	Internal communications fault (RST) Fault during the processing of a command
01	30	20	Unknown command code (RST)
01	30	21	Unknown protocol identifier (RST)
01	30	25	Wrong parameter identifier (RST)
01	30	26	Block-length fault (RST)
01	30	28	Unknown message type (RST)
01	30	29	Unknown command type (RST)
01	30	3A	Wrong address alignment (RST)
01	30	3B	Invalid address area (RST)
01	30	3C	Invalid parameter (RST)
01	30	3D	Invalid operand type (RST)
01	30	63	Buffer overflow The data length is larger than the available memory location of 800H bytes (memory area of BÜP19E).
02	32	01	Module not available (PST) The indicated block address addresses a system area which does not contain a module.
02	32	10	Module not addressable (PST) The indicated block address has no module number
02	32	20	STOP fault (PST)
02	32	21	RUN fault (PST)
02	32	22	Operating mode cannot be changed (PST)
02	32	23	Field access prohibited (PST), unauthorised access mode
02	32	24	Field protection active)
02	32	25	Timer disabled (PST)
02	32	26	Module number too high (PST)
02	32	27	No data module (PST)
02	32	28	Data module too small (PST)
02	32	29	Monitor disabled (PST)
02	32	2A	Reference list disabled (PST)
02	32	2B	System-clock fault (PST)
02	32	2C	Reference-list entry cannot be changed (PST)

Fig. 7-15 BÜP19E — Fault messages, Part 1

Fault information			Explanation
PLC response	Error class	Error code	
03	32	02	Memory access fault (PST) Faults occurred when RST memory area was accessed: Data incorrectly read or incomplete.
03	32	03	Wrong memory type (PST) The type of access for the addressed PST memory is not permitted (e.g. writing to EPROM)
03	32	05	Internal communications fault (PST) Fault during the processing of a command
03	32	20	Command code unknown (PST)
03	32	21	Protocol identifier unknown (PST)
03	32	23	Invalid coordination marker (PST)
03	32	25	Wrong parameterisation (PST)
03	32	26	Block-length fault (PST)
03	32	28	Message type unknown (PST)
03	32	29	Command type unknown (PST)
03	32	3A	Wrong address alignment (PST)
03	32	3B	Invalid address area (PST)
03	32	3C	Invalid parameter (PST)
03	32	3D	Invalid operand type (PST)
03	32	40	Identification still not implemented (PST)
03	32	63	Buffer overflow. The data length is larger than the available memory location of 800H bytes.
03	32	D2	Coordination marker disabled Occurs only for field coordination
03	30	01	No module (RST)
03	30	10	Module not addressable (RST) The indicated block address has no module number
03	30	21	RUN fault (RST)
03	30	22	Operating mode cannot be changed (RST)
03	30	23	Field access prohibited (RST)
03	30	24	Field protection active (RST)
03	30	25	Timer disabled (RST)
03	30	26	Module number too large (RST)
03	30	27	No data module (RST)
03	30	28	Data module too small (RST)
03	30	29	Monitor disabled (RST)
03	30	2B	System-clock fault (RST)
03	30	2C	Reference-list entry cannot be changed (RST)
03	33	10	No valid block addr. for the indicated module number
03	33	20	Wrong RST command code
03	33	23	Command for RST larger than 80H
03	33	29	Wrong command type
03	33	3C	No module type for indicated module number
03	33	3D	Wrong operand type
03	33	47	Wrong data length
10	31	60	Passive job terminated
10	31	61	Active job terminated
20	‘Length’		Job fault-free, requested data length <> recommended data length

Fig. 7-16 BÜP19E — Fault messages, Part 2

8 BUEP03E Protocol

8.1 Introduction

The BÜP03E transfer protocol *(BOSCH Übertragungs-Protokoll)* handles all communication tasks between the CL500 computer interface module and connected peripheral devices, such as code readers, measuring instruments, positioning controllers, intelligent control panels, printer, terminal, etc.

The BÜP03E protocol is loaded onto the R500/R500P module and receives the jobs in the form of commands from the central processing unit.

The BÜP03E protocol consists of several commands. The operator can use these commands to compile a "user-defined" protocol. Such protocols are generated in the form of command combinations. As a result, the flexibility of the BÜP03E with respect to a variety of peripheral devices is ascertained.

The BÜP03E protocol enables the communication with those peripherals that do not adhere to a specific protocol. This communication procedure is handled by the Receive command which receives all transmitted characters and subsequently presents them to the operator for further interpretation.

8.2 BÜP03E Protocol — Application Areas

The BÜP03E transfer protocol implements the following tasks:

- Receives data from peripherals (e.g. code reader, measuring instruments, etc.), and conveys it to the programmable logic controller (PLC).
- Transmits data from the PLC to the peripherals (e.g. printer, etc.).
- In order to reduce the burden on both PLC program and PLC programmer, it implements the printer formatting and code conversion tasks on the R500 Computer interface module.

  The BÜP03E protocol includes the Formatted Output command which provides the operator with a simple method of sending the texts, or texts combined with variables, to the printer.

  Using the command parameters, the operator specifies the data and data formats to be transmitted to, or received from, the peripheral devices. The resulting commands are then implemented on the R500 Computer interface module by means of the R5REQ function module. They enable access to the following data ranges in the PLC:

  - Data modules
  - Data fields
  - Data buffers

8.3 Explanation of Terms

UART Receive buffer: The R500 operating system reads all the characters into this 384-byte buffer via the serial interface. This receive buffer is located on the R500 module.

BÜP03E Receive buffer: The BÜP03E reads the characters from the UART receive buffer and writes them to its own BÜP03E receive buffer, the size of which can be selected from 10 to 256 bytes. This receive buffer is located on the R500 module.

PLC Receive buffer: This receive buffer, defined by the user on the PLC side, accepts the contents from the BÜP03E receive buffer. This buffer is therefore the same size. This receive buffer is located on the central processing unit.

Unrestricted Receive: The Receive process receives all characters and writes them to the designated Receive buffer; no conditions are taken into account. The Receive process can be concluded with the monitoring-time overflow or stopped via the Terminate Job command.

Conditional Receive: The receive process can take account of the conditions with regard to start and end identifiers, termination characters, Receive-data length and monitoring time. The characters are accepted only under the preset conditions. The Receive process can be ended by end identifier(s), Receive-data length or monitoring time, or terminated by termination character or aTerminate Job command.

Receive Mode: The receive features, e.g. start, end identifiers, cancel criteria, data length, number of BCC, etc.

Ring buffer: For the intermediate storage of data. The ring buffer is managed by RD and WR pointers. The WR pointer is counted up to the end of the ring buffer and is then placed at the start of the ring buffer. If the RD pointer stops counting due to data being read out previously, the RD pointer may be overtaken by the WR pointer.

RD pointer: For reading out the data from the ring buffer.

WR pointer: For writing the data to the ring buffer.

User pointer: Situated in the first word of the PLC Receive buffer. It indicates the relative position of the character which was received last.

Job parameter list: Specifies command parameters and is addressed by R5REQ parameters P0 - P2.

Command chain: Links several commands with the Start command via the command position number.

Identifier: Three words which are used for determining the address in the DM, DF or DB.

The words in detail:

1st word: Field type (44H = DM, 64H = DB, 43H = DF).

2nd word: Field index (DM = 0-255, DB = 0, DF = FFFFH).

3rd word: Field offset (DM = 0-510, DB = 0-510, DF = 0-24 kbyte).

Caution: Only the even-numbered byte addresses are permitted.

Control text: Text combined with format instructions.

8.4 Transmission Format

The non-displaying ASCII characters are used to control the protocols. These control characters, like the other data, are written to memory as hex code by the user. All the data is then output with the Transmit command via the serial interface. The data can be input into the PLC memory either via the PG editor or the PLC program commands. The PG editor only allows the ASCII and hex formats. In both the ASCII and hex formats, the addresses are zero-filled up to the word boundaries. A single control character can therefore not be input with the PG editor. The user must edit the required control characters, possibly combined with a text character, as a hex word.

No.	Symbol	Type	Sign	Data field	F
D0	Text	Word	N	5402	H
D2		Word	N		H
D4		ASCII	N	ESTE	

Fig: Inputting text with control characters

The first line is input as a hex word, and the second line as ASCII text in the PG editor.

The hex dump of the data in the controller:

02 54 45 53 54 45 ==> STX TEST

ASCII	Hex	Description
STX	02	Start of Text ASCII character
ETX	03	End of Text ASCII character
ETB	17	End of Block ASCII character
CAN	18	Cancel ASCII character
NUL	00	All bits = 0 ASCII character

Fig. 8-1 Control characters — Protocols

ASCII	Hex	Description
CR	0D	Carriage Return ASCII character
LF	0A	Line Feed ASCII character
FF	0C	Form Feed ASCII character
TAB	09	Tabulator ASCII character
ESC	1B	Escape ASCII character

Fig. 8-2 Control characters — General-purpose

The commands execute only the specified tasks and do not analyse the responses of the partner module. Any response to the fault statuses must occur on the PLC side. For example, a data request is transmitted to a peripheral, but is not understood by the latter. This case must be analysed on the PLC side, and an appropriate response must be issued.

8.5 Transmission Sequence

8.5.1 Receiving Characters

Two different receive modes can be specified by the Receive Mode command.

a) Unrestricted Receive: The user defines a PLC Receive buffer (max. 256 bytes) on the PLC side. When the Receive command activates communication via R5REQ, all characters are received and filed in the BÜP03E Receive buffer until

- the monitoring time overflows
- or the job is terminated
- or the Receive-data length has been reached.

Unrestricted Receive also facilitates communication with the peripherals which have variable protocol.

A practical case: BÜP03E should Receive-data from a code reader. However, the code reader transmits the data with a start identifier only during the first transmission. If the transmission has to be repeated, the start identifier would be missing. In practice, this means that the protocol sequences are not always uniform.

In such a case, the user can receive all the characters and react accordingly, as indicated on Page 8-30 - Example 1.

b) Conditional Receive: Receive Mode parameters, such as start and end identifiers, termination characters and Receive-data length, can also be specified by the Receive Mode command. If these identifiers are recognised in the data flow, the appropriate response is implemented. All received characters (even the control characters) are received in the BÜP03E Receive buffer.

The Data Copy command copies all data from the BÜP03E Receive buffer into the PLC Receive buffer for interpretation by the user.

8.5.2 Transmitting Characters

There are two modes available:

a) Unformatted output: The Transmit command transmits data from a PLC transmit buffer (DM, DF, DB) without conversion by the serial interface. Reformatting is not implemented, i.e., a 0FFH in the memory causes a 0FFH in the line.

Using this procedure, the user can easily output self-formulated request and acknowledgement information. For example, if the connected device at the start of the communication requires an STX control character, this can be edited as a hex word (STX = 02H) combined with a text character in the memory which is to be transmitted. This control character is then transmitted with the other data.

b) Formatted output: In some applications (e.g. printer..) texts or texts with variables must be output according to the following display types

- Hex
- Decimal
- ASCII characters
- Floating point

The Formatted Output command, described on Page 71, fulfils these requirements.

8.6 Receive Buffer

8.6.1 UART Receive Buffer

The data received via the serial interface is first filed in the UART Receive buffer. This Receive buffer is designed as a ring buffer and contains a total of 384 characters. This buffer is managed by the R500 operating system. The characters are read into this buffer via the interface, independent of the BÜP03E protocol. The characters are read out character by character from the BÜP03E log and filed in the BÜP03E Receive buffer. The oldest character is always read out first. The character, which indicates the internal RD pointer, is read out and the internal RD pointer is moved by one character. The UART Receive buffer can be reset (flushed) if the user issues the Reset Receive Buffer command. Information such as

- Buffer empty

- Character there

- Receive fault

is also available for processing. The receive fault is indicated as a centralised fault (buffer overflow, parity fault...). The BÜP03E protocol responds to this fault by issuing a Reset command to the UART Receive buffer and returning a fault message to the user.

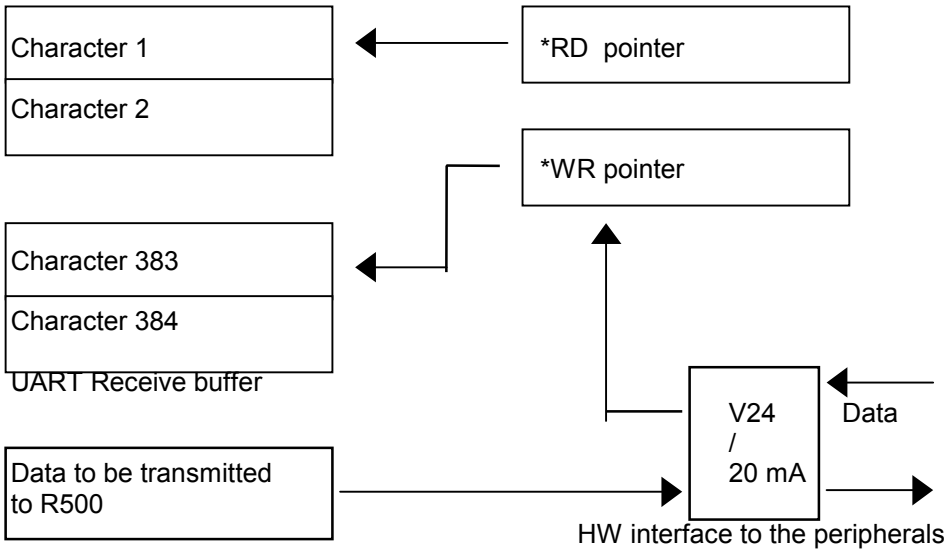
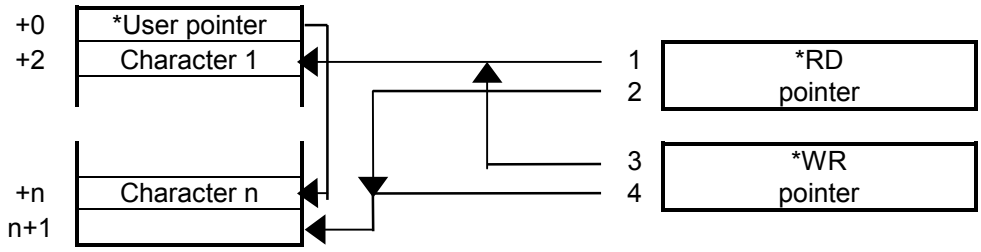


Fig. 8-3 UART Receive buffer

8.6.2 BÜP03E Receive Buffer

The Receive command continually copies data from the UART Receive buffer into the BÜP03E Receive buffer. The user can select the size of this BÜP03E Receive buffer by issuing the Receive command via a parameter, however the size must be within the range of 10 to 256 bytes. Otherwise, a fault message is returned and the BÜP03E Receive buffer is not applied.

The BÜP03E Receive buffer is managed as a ring buffer, as long as the user does not influence the RD and WR pointers by issuing the Reset Receive Buffer command. The first word in the defined BÜP03E Receive buffer is reserved for the user pointer. The user pointer points to the character which was last read in. A maximum of 254 bytes are available for the characters. The user pointer begins with 0 (zero), although "Character 1" is in the third position in the BÜP03E Receive buffer.



1+3 : RD and WR pointers after Reset Receive Buffer has been selected

4 : WR pointer after n-character has been read into the Receive buffer

2 : RD pointer after n-character has been read out of the Receive buffer

2+4 : Read in n-character and output to PLC

Fig. 8-4 BÜP03E — Receive buffer

The RD and WR pointers are required for internal management of the ring buffer. These pointers are always reset with the Receive command and can also be reset at any time by the user with the Reset Receive Buffer command. After being reset, both pointers point to the start of the BÜP03E Receive buffer. If n-characters are now received, the WR pointer is increased by n-characters and points to the next free location in the BÜP03E Receive buffer (n+1). The RD pointer points to the character which was first received until the user has read the data with the Data Copy command. RD and WR pointers are then equal, and (RD_Pointer-1) defined as user pointers.

If the characters are not retrieved in time by the user, the WR pointer may overtake the RD pointer, resulting in a loss of data. The user is informed of a data loss when he retrieves the data with the Data Copy command. If this fault occurs, the data is not copied into the PLC Receive buffer and the RD and WR pointers are reset by BÜP03E. When the next Data Copy process occurs, the characters are copied to the head of the PLC memory.

8.6.3 PLC Receive Buffer

The PLC Receive buffer is the same as the BÜP03E Receive buffer. The received data is collected in this buffer and is provided to the user for interpretation.

The Data Copy command writes all the data which is in the BÜP03E Receive buffer into the PLC Receive buffer in one cycle.

8.6.4 Data Exchange Between Receive Buffers

The previous section explained in detail the internal design and management of the three Receive buffers (UART, BÜP03E, and PLC Receive buffers). This section describes how data exchange and data flow has been organised.

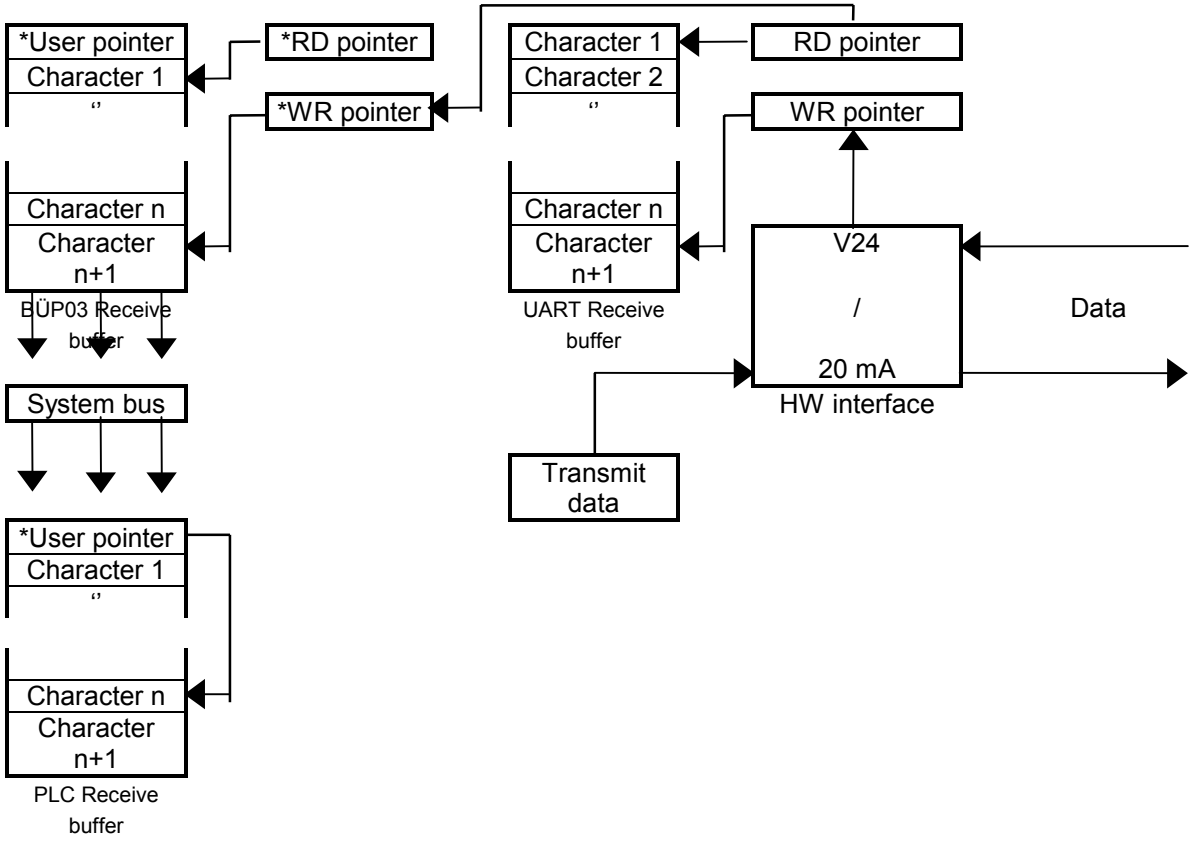


Fig. 8-5 BÜP03E — Data exchange between Receive buffers

The characters in the UART Receive buffer are read into the BÜP03E Receive buffer character by character. The user can use Data Copy to read all the data which is in the BÜP03E Receive buffer into the PLC Receive buffer.

The Transmit command defines a transmit area on the PLC side. BÜP03E reads this transmit area into a transmit buffer and transmits the data out of this buffer to the serial interface.

8.7 BÜP03E Commands

8.7.1 Command Activation

The BÜP03E protocol incorporates several commands which should be activated by the user for communication purposes. The commands are activated by the R5REQ function module.

The first parameters (P0 - P2) of the R5REQ function module point to a memory (DM, DF, DB) in which the protocol-specific job parameter list has been filed (see description of R5REQ, Pages **Fehler! Textmarke nicht definiert.** ff).

Protocol-specific parameters:

PW1 : Result of the job

PW2 : CXN and channel number

PW3 : Job number (0 - 32767)

The user assigns PW4 - PW15 with parameters according to the command.

When a job parameter list for a specific command has been described and Bit 15 has been set to Low (zero) in PW3, the command is conveyed to the BÜP03E protocol by the R5REQ function module. The command is then activated and entered in the R500 command queue. BÜP03E retrieves this command from the command queue and executes it.

A command can only be active once at any one time. If the user nevertheless attempts to start a command which is already active, BÜP03E terminates the command and issues a fault message. The BÜP03E protocol ensures that the command, e.g. Receive, cannot be active twice at the same time.

8.7.2 Operating Method

The BÜP03E protocol provides the user with a number of commands. These commands then execute the jobs which are required for communication.

The following commands have been defined:

- Start
- Receive
- Transmit
- Reset Receive Buffer
- Data Copy
- Formatted Output
- Receive Mode
- Terminate Job

Each command is indicated by a job parameter list. All these commands are activated by the R5REQ. The Terminate Job command is provided by the R500 system. The remaining commands form the BÜP03E protocol.

In order to be able to activate a peripheral, the user has to execute many of these elementary commands via a function call module. This is a major problem from the point of view of time and management. It would be easier to link several elementary commands toghether, and start them via a function call module. This method of combining commands is explained on Page 8-12.

8.7.2.1 Interlinking via BÜP03E START Command

The associated job parameter lists are filed contiguously one after the other in the PLC memory (DM, DB, DF). Each job parameter list specifies a command (==> job). The sequence of the commands in the PLC memory specify the positions of the commands. Application-specific combinations of the command positions can be stipulated with the Start command.

If the Start command is now started by means of the R5REQ function module, the command chain, indicated via identifiers, is retrieved from BÜP03E and executed according to the stipulated sequence.

R5REQ parameter PW1 - PW3	
Command identifier 'ST'	
Identifier for the command chain	
No. of parameter list: 4	
1	2
FFH	4

Start command

Command 1 Receive
Command 2 Receive mode
Command 3 Reset Receive Buffer
Command 4 Transmit

Job parameter lists are located directly behind each other in the job memory (DM,DF,DB)

Fig. 8-6 BÜP03E — Interlinking command

The jobs are processed in the following sequence: 2, 1, 4, End

With this method the maximum job memory is restricted to one data module (= 512 bytes). A maximum of 16 commands can be defined per job memory and, depending on the application, can be linked to each other.

This method unloads the system bus by transferring all the job parameter lists once. A command may occur several times in the command chain.

During the active phase of Receive, the following individual commands can be started parallel to the commenced command chain:

- Data Copy
- Transmit
- Reset Receive Buffer
- Receive Mode
- Formatted Output

Parallel operation is only possible with Receive in the command chain, not in single command mode.

If a fault occurs (e.g. invalid parameters, system bus fault...), the command chain is terminated immediately and the fault message is returned with the interlinking depth.

In order to establish a single communication with a peripheral, the interlinking of the following commands must be indicated with the Start command.

- Set Receive mode
- Request data
- Enable Receive
- Acknowledge Receive

The Data Copy command is not inserted into the command chain, but is activated parallel to Receive. The Start command starts the command chain.

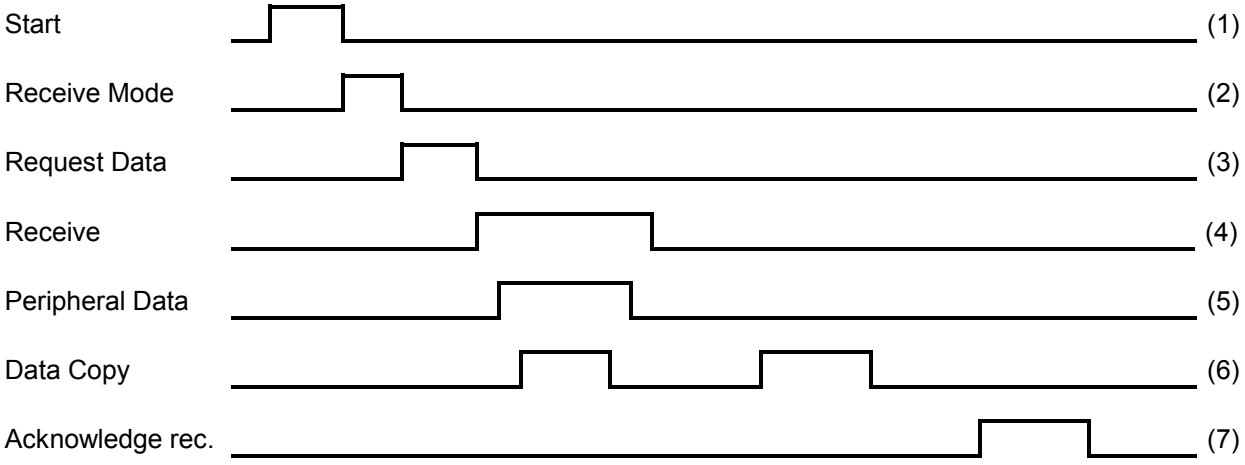


Fig. 8-7 BÜP03E — Chronological sequence of a single communication

(1) Start the command chain

(2) Set receive mode (end character, Receive-data length).

Do NOT specify a start character, otherwise a peripheral NAK is not recognised as BÜP03E would be waiting for the start character.

(3) Request data by "Transmitting" a control text.

(4) Switch ON the (unrestricted) receive.

(5) Peripheral responds to the data request by BÜP03E.

(6) At the same time as the receive process, copy the data into the PLC Receive buffer and analyse. There are two options available for this step:

- Peripheral transmits the data after the data request has been successfully received by BÜP03E . The data is retrieved until, e.g. an end identifier, is recognised.
- Peripheral transmits, e.g. an NAK, and requests a repeat of the data request. In this case either the communication can be terminated with a Terminate Job sequence or the step (3) can be repeated.

(7) Communication is terminated via the stipulated conditions. The received data is checked by the PLC program and positively acknowledged if correct. If a fault occurs, the received data is negatively acknowledged and communica tion can be restarted at Step (1).

8.7.3 Command Types

This section provides a detailed explanation of the BÜP03E commands and their parameters and lists possible application examples. These examples should be regarded as an introduction to the *BÜP03E Command Language*..

When using the commands, the user should NOT impose unnecessarily high traffic loads on the system bus. High loads can be prevented by skilful use of the commands in program loops and function modules. For example, if a job is complex, the commands should be linked to each other as only one R5REQ call is adequate for activation of the command chain. The data should only be retrieved with the Data Copy command when required, as there is no point in calling the data until it is available. The data receive times in an ideal case are indicated in the table below. A practical value would be 100 ms for 100 characters at a baud rate of 19200. The 100 characters to be received can therefore be retrieved within 100 ms when the serial interface is operating at a baud rate of 19200.

Baud rate (bps)	Time for one character	Time for 100 characters
19200	0.50 ms	50 ms
9600	1.00 ms	100 ms
4800	2.00 ms	200 ms
2400	4.00 ms	400 ms
1200	8.00 ms	800 ms
600	16.00 ms	1600 ms
300	32.00 ms	3200 ms
110	90.90 ms	9090 ms

Fig. 8-8 BÜP03E — Transfer times

8.7.3.1 START Command

Description: This command defines the execution sequence of a command chain in the PLC memory (DM, DB, DF) and starts this command chain.

Parameter word	Contents	
PW4	Command identifier - ST (Start)	
PW5	Field type	Identifier on the command chain
PW6	Field index	
PW7	Field offset	
PW8	Number (1-14) Number of commands	
PW9	2nd command	1st command
PW10	4th command	3rd command
PW11	6th command	5th command
PW12	8th command	7th command
PW13	10th command	9th command
PW14	12th command	11th command
PW15	14th command	13th command
PW16	reserved	reserved

Fig. 8-9 BÜP03E — Start parameters

Valid position numbers are 1 to 14 and FFH. An FFH as a position number denotes the end of the command chain. If 14 commands have been linked, an end identifier (0FFH) is NOT required.

NOTE –

Changes to the job memory (command memory) are not considered until the next command starts.

8.7.3.2 RECEIVE Command

Description: This command enables data to be received without restriction via the interface and written to the indicated buffer.

Parameter word	Contents
PW4	Command identifier - RX (Receive)

Parameter word	Contents
PW5	Monitoring time in 10 ms matrix

Maximum selectable time (10 ms * 65535) is approximately 10 min. When 0 (zero) is specified, the time monitor is switched OFF. The monitoring time indicates the maximum time between two characters.

NOTE –

The sequence of the monitoring time in the DW4 denotes the normal end of the receive process and not a termination with fault.

If a data copy is attached, fault 0305H is issued in the status word and 4003H in the fault word, provided a character has not already been received. (Receive buffer empty)

Parameter word	Contents
PW6	PLC Receive buffer size (10-256 bytes)

This parameter specifies the size of the Receive buffer on the R500 module and on the PLC side.

Parameter word	Contents
PW7	Receive-data length (max. 256 bytes)

This parameter can specify, e.g., the length of the data during the Receive process. BÜP03E normally concludes the Receive process when the specified number of characters has been reached. At the same time *all received characters* (including control characters) are counted. When the length is 0 (zero), the stringsize condition is ignored.

PW8 - PW16 are reserved.

8.7.3.3 TRANSMIT Command

Description: This command enables data to be transmitted from the memory to the peripherals via the interface; the data is output without being converted.

Parameter word	Contents	
PW4	Command identifier - TX (Transmit)	
PW5	Field type	Identifier on the command chain
PW6	Field index	
PW7	Field offset	
PW8	Length of the transmit data in bytes	

Fig. 8-10 BÜP03E — Transmit command parameters

Max. possible length is 512 bytes (= one DM size). For larger quantities of data the user should start several jobs.

PW9 - PW16 are reserved.

8.7.3.4 RESET RECEIVE BUFFER Command

Description: This command enables the internal RD and WR pointers of the BÜP03E and UART Receive buffer to be reset.

Syntax:

As on Page 8-15 in the Start Syntax section.

Parameter word	Contents
PW4	Command identifier - RS (Reset)
PW5	0: Reset UART Receive buffer 1: Reset UART and BÜP03E Receive buffer

Fig. 8-11 BÜP03E — Reset Receive Buffer command parameters

PW6 - PW16 are reserved.

8.7.3.5 RECEIVE MODE Command

Description: The following modes can be selected with this command:

- Start character
- End character
- Number of BCC
- Termination character
- Marker number for interlinking depth

The selected Receive modes are valid until they are redefined. The Receive modes can be modified within a command chain if the Receive process (Receive command) is inactive. If an attempt is made to modify a receive mode while Receive is active, the process is terminated and a fault message is issued.

The receive mode values are reset during

- Power ON and when the standby supply is defective or missing
- loading of the protocol

and remain unchanged during a

- R500 Reset procedure
- Stop/Run procedure

During a ZS Stop/Run procedure the R500 operating system terminates all inactive jobs, and activates a Terminate Job sequence for active jobs.

An R500 Reset procedure is not registered by the central processing unit.

Basic adjustment of the receive mode values:

- No start character
- No end character
- No BCC
- No termination character
- Interlinking depth not returned

Syntax:

As on Page 8-15 in the Start - Syntax section.

Job parameter list:

Parameter word	Contents	
PW5	No. of start characters	No. of end characters

End characters:

0 : No end condition

1 : One end character

2 : Two end characters

Start characters:

0 : No start condition

1 : One start character

2 : Two start characters

The start characters may occur in the data flow. If only one end character has been defined, it must NOT occur in the data flow; otherwise the Receive process is terminated. However, if two end characters are defined, any number of end characters may occur in the data flow, provided one end character is not directly behind the other. If one end character is directly behind the other, the Receive process is always terminated.

Parameter word	Contents	
	HIGH byte	LOW byte
PW6	2nd start character	1st start character

These start characters are only considered when the number has been specified in the HIGH byte of PW5, otherwise they are ignored.

Parameter word	Contents	
	High byte	LOW byte
PW7	2nd end character	1st end character

These end characters are only considered when the number has been specified in the LOW byte of PW5, otherwise they are ignored.

Parameter word	Contents		
	HIGH byte	LOW byte	
PW8	Number of BCC	Termination type	No. of terminations

No. of termination characters:

0 : No termination characters

1-4 : Number of termination characters

Termination type:

0 : Individual termination characters

1 : All termination characters

Number of BCC:

0 : No BCC characters

1 : One BCC character

2 : Two BCC characters

Termination type 1: The termination process requires that all the defined termination characters occur one behind the other in the data flow.

Termination type 0: One of the defined termination characters in the data flow causes the termination process.

BCC: The Block Check Count has a value of one byte or one word which is formed according to a specific method (e.g. LRC, CRC) via a data record. BCC is formed on the receiver side according to the same method and compared with the received BCC. This method enables any receive faults to be recognised.

BCC can only be expected during the end-character sequence and is in addition to the received data.

Parameter word	Contents	
	High byte	LOW byte
PW9	2nd termination character	1st termination character
PW10	4th termination character	3rd termination character

These termination characters are only considered when the PW7 number has been defined between 1 and 4, otherwise they are ignored.

PW11:

Specifies the marker byte (0 - 255) in which the user wants to have the interlinking depth if a fault or termination occurs. If values are larger than 255, an interlinking depth is not returned. Interlinking depth 1 refers to the first, 2 to the second command in the command chain.

PW12 - PW16 are reserved.

8.7.3.6 DATA COPY Command

Description: This command enables the data which has been read into the BÜP03E Receive buffer to be copied from the PLC program into the PLC Receive buffer during the Receive process. If the BÜP03E Receive buffer is not emptied quickly enough with this command, a data loss may occur which is signalled back. If no new characters have arrived after the last "Data copy", the user is informed by a fault message.

The BÜP03E Receive buffer is always copied 1:1 into the PLC Receive buffer. The old data in the PLC Receive buffer is always overwritten by this command.

Syntax:

As on Page 8-15 in the Start - Syntax section.

Parameter word	Contents	
PW4	Command identifier - DC (Data copy)	
PW5	Field type	Identifier on PLC Receive buffer
PW6	Field index	
PW7	Field offset	

Fig. 8-12 BÜP03E — Data-Copy command parameters

PW8 - PW16 are reserved.

8.7.3.7 FORMATTED OUTPUT Command

Description: The Formatted Output command enables text, variables or text combined with variables to be output to the printer. This command interprets the control text and associated variables indicated by the parameters (PW5 - PW7). The format instructions have been integrated in the control text and begin with a **%** character. The characters which do not begin with the **%** character appear unchanged in the output. The format instructions ensure the required representation method of the next argument.

The control text is separated from the arguments with a comma. The arguments (variables) are situated one behind the other after the control text (comma) and always begin at the word boundaries. The numerical values of the variables must be copied by the PLC program to the appropriate location after the comma.

The PG editor inserts a 0 (zero) into the control text if the text ends with an odd number. This 0 (zero) is ignored by BÜP03E.

The control text and the arguments may be in a data module, for example. The identifier is passed to this data area with the Formatted Output command.

Possible control text with two arguments (see also Page 8-15 in the Start section - Example 4):

Temperature of %3d degrees is too high by %3d degrees,<Arg1><Arg2>

The printer would print the text

Temperature of 100 degrees is too high by 10 degrees

if the value of Arg1 = 100 and Arg2 = 10 . There are as many arguments as there are format instructions. If this is not the case, a fault message is returned.

The Formatted Output command provides the following format instructions:

%+-mu	Decimal number (unsigned)	1 word (0..65535)
%+-md	Decimal number(signed)	1 word (32767..-32767)
%#x	Hex number	1 word
%c	Single ASCII character	1 word
%+-m.nf	Floating-point number	2 words

m : Total number of positions

n : Positions after decimal point

u, d, x, c, f : Identifiers for format instructions

\+ : Output with sign

\- : Left-justified output (normal: right-justified)

+- : Left-justified and with sign

\# : Output of hex numbers with leading zeros

A character variable is in the LOW byte of the word.

A floating-point number is positioned in memory according to the Low-High word principle, and is represented with single accuracy.

All numbers are displayed from the **%** character, right or left-justified (see above). A hex number can be displayed with (%#x) or without (%x) leading zeros.

If the actual "%", ",", and "\" characters are to be output, the following control sequences are used:

- **\%** prints a **%** character

- **\,** prints a **,** character

- \\ prints a \ character

Syntax:

As on Page 8-15 in the Start - Syntax section.

Parameter word	Contents	
PW4	Command identifier - FO (Formatted Output)	
PW5	Field type	Identifier on text with formats and variables
PW6	Field index	
PW7	Field offset	

Parameter word	Contents
PW8	Length (control text + arguments) in bytes

The length may be indicated as an estimate, however the estimate must be at least as large as the actual length, i.e., if the actual length of the data is 48 bytes, the estimated length may be indicated as 50 bytes. BÜP03E actually copies 50 bytes from the PLC memory (DM, DB, DF) into the R500 memory, however the actual data itself would be selected from the volume of data in the R500 memory. It is not necessary to count the text precisely to the byte. However, the estimated length should not be excessive (in the case of DM, for example, the max. length always = 512 bytes). On account of the system bus access times, the length should be indicated as accurately as possible.

PW9 - PW16 are reserved.

Floating-point numbers

The floating-point numbers are represented according to two methods. Internal representation specifies how floating-point numbers have been filed in memory. External representation specifies how floating-point numbers are printed or displayed. These methods of representation are explained below.

Examples of external representation:

Constants	Value
23.45e6	$23.45 * 10^{6}$
2.1e-5	$2.1 * 10^{-5}$
-1.23	-1.23

Internal representation according to Siemens:

VZ	23-bit mantissa	VZ	7-bit exponent

2^{31} mantissa * 10^{n} 2^{0}

Internal representation according to IEEE - floating-point format:

VZ	8-bit exponent	23-bit mantissa

2^{31} $2^{(n-127)}$ * mantissa 2^{0}

The floating-point numbers occupy four bytes with single accuracy.

BÜP03E provides the %m.nf format instructions for floating-point numbers.

The specified methods of representation are:

- internal : IEEE format

- external : 50.44e6, 50.44e-6, 1.23, -466.600 ....

The exponents "e6, e-6" are added if the internal number cannot be represented in the indicated format. The values of the above-mentioned numbers would then be: 50.44 * 10^6 and 50.44 * 10^-6.

8.7.3.8 Terminating Jobs

The user has the options, as described below, to naturally conclude or terminate a job.

Terminate via R5REQ function module

The parameter word (PW3) in the job parameter list has the following structure.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
a	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x = Job number (0 - 7FFFH)

a = 0 --> Start job

a = 1 --> Delete job from the queue (terminate)

It is clear from the above diagram that the job is started from Bit 15 when logic is 0 and terminateed from Bit 15 when logic is 1, i.e., the user can start or terminate the job with the same job parameter list. Only Bit 15 must be controlled accordingly. When a job is terminated, only the first three parameters in the job parameter list (PW1 - PW3) are relevant. If the user terminates a job, BÜP03E terminates the command chain and issues a fault message.

A command chain is started via Start with the start command and is terminated via Start with the termination command, as only this command is in the command queue.

Normal conclusion via monitoring time

The Receive command enables a monitoring time to be indicated up to a max. 10 min. If this time elapses during the Receive process, without a character arriving from the peripheral, the Receive process is concluded normally and is possibly stepped-on in the command chain. The time is started with the first character for the Unrestricted Receive process and with the start character for the Conditional Receive process. This time is then reset when the character has been received.

Termination via termination character

The termination characters are specified in the Receive Mode command. Depending on the setting, the Receive process is terminated with a fault message either when single termination characters occur or when all defined termination characters are positioned one behind the other. In this case a command chain is NOT stepped-on.

Normal conclusion via start and end characters

The start and end characters are specified with the Receive Mode command. If the end character(s) occur in the data flow, the Receive process is concluded normally and is possibly stepped-on in the command chain.

Normal conclusion via Receive-data length

The length of the data to be received in bytes is specified with the Receive command. In the case of Unconditional (unrestricted) Receive processes, all of the characters transmitted by the peripheral (including control characters) are considered. In the case of a Conditional Receive process, all characters (including control characters) with the start character are considered. The Receive process is concluded normally and is possibly stepped-on in the command chain if the specified number of characters has been received.

8.7.3.9 Summary of BÜP03E Commands

Param. word	Receive	Data Copy	Reset Receive Buffer
PW4	RX	DC	RS
PW5	Monitoring time 0 - 65535 in 10 ms Raster unit	PLC Receive buffer id. Field types 44H=DM, 43H=DF, 64H=DB	Reset Receive buffer 0 = UART 1 = UART + BÜP03E
PW6	Size of the PLC receive buffer 10 bytes - 256 bytes	Field index DM = 0-255 DB = 0, DF = FFFFH	
PW7	Receive-data length in bytes 1 - 255 0: no length condition	Field offset DM = 0-510, DB=0-510 DF=0-24 kbytes	

Fig. 8-13 BÜP03E — Summary of Commands, Part 1

Param. word	Receive Mode	Transmit	Formatted Output
PW4	MD	TX	FO
PW5	LOW byte: No. of end characters HIGH byte: No. of start characters	Transmit data id. Field types 44H=DM, 43H=DF, 64H=DB	Control text id. Field types 44H=DM, 43H=DF, 64H=DB
PW6	LOW byte: 1st start character HIGH byte: 2nd start character	Field index DM = 0-255 DB = 0, DF = FFFFH	Field index DM = 0-255 DB = 0, DF = FFFFH
PW7	LOW byte: 1st end character HIGH byte: 2nd end character	Field offset DM = 0-510, DB=0-510 DF=0-24 kbytes	Field offset DM = 0-510, DB=0-510 DF=0-24 kbytes
PW8	LOW byte r. nibble: No. of aborts LOW byte l. nibble: Abort type HIGH byte: No. of BCC	Transmit data length in bytes 1 - 512 bytes	Length of control text with variables in bytes 1 - 512 bytes
PW9	LOW byte: 1st abort character HIGH byte: 2nd abort character		
PW10	LOW byte: 3rd abort character HIGH byte: 4th abort character		
PW11	Marker byte for interlinking depth # 0-255: none		

Fig. 8-14 BÜP03E — Summary of Commands, Part 2

Param. word	Start		
PW4	ST		
PW5	Command chain id. Field types 44H=DM, 43H=DF, 64H=DB		
PW6	Field index DM = 0-255 DB = 0, DF = FFFFH		
PW7	Field offset DM = 0-510, DB=0-510 DF=0-24 kbytes		
PW8	Maximum utilised interlinked parameter list no. 1 - 16		
PW9	LOW byte: 1st command HIGH byte: 2nd command		
PW10	LOW byte: 3rd command HIGH byte: 4th command		
	"		

Fig. 8-15 BÜP03E — Summary of Commands, Part 3

8.7.4 Examples of BÜP03E Commands

The examples illustrate how the interlinked commands implement effective applications. In theory, any command can be interlinked with any other command. However, commands should be interlinked effectively. For example, the Data Copy command can be completely integrated into the command chain following the Receive command. However, the command is not activated during the Receive process. The Data Copy command should therefore be started, as required, parallel to the Receive command.

The parameter list (16 words each) stands for the examples listed below in the DM10, commencing from DW0, and is started with the FM call in the PLC program.

```
CM    -R5REQ,6
P0    0     ;Data in the data module (0=DM, 1=DB, 2=DF)
P1    10    ;Data module number 10
P2    0     ;Data word number      0
P3    1     ;Only one job parameter list
P4    M0    ;Fault message
P5    M2    ;Result
```

Example 1: Only Unrestricted Receive should be activated with time-monitoring and without interlinking. The parameters are entered in the DM10, commencing from DW0.

```
DW0 - DW4 = reserved for R5REQ
DW6     = 'RX'      Command identifier
DW8     = 10        Monitoring time = 100 ms
DW10    = 100       Size of the PLC Receive buffer = 100
DW12    = 0         No Receive-data length condition
```

The received characters are written to the BÜP03E Receive buffer and can be loaded into the PLC Receive buffer with the Data Copy command at any time.

Example 2: A data area should be transmitted to the interface. The parameters are entered in the DM10, beginning with DW0. The characters to be transmitted are in the DM11, commencing from DW0.

```
DW0 - DW4 = reserved for R5REQ
DW6     = 'TX'      Command identifier
DW8     = K44H      Identifier
DW10    = 11        for
DW12    = 0         Transmit buffer
DW14    = 150       Transmit 150 bytes
```

The 150 bytes written to the DM11, commencing from DW0, are transmitted unchanged.

Example 3: Data should always be received conditionally and then retrieved; the following conditions are defined with the Receive Mode command.

- Two start characters
- Two end characters
- Receive-data length
- Two BCC
- Single abort characters
- Four abort characters

Command 1: Select Mode

DW0 - DW4 = Reserved for R5REQ
DW6 = 'MD' Command identifier
DW8 = K0202H Two start, two end characters
DW10 = 'ss' Start characters
DW12 = 'ee' End characters
DW14 = K0214H Two BCC, single, 4 abort
DW16 = 'aa' Termination
DW18 = 'aa' character
DW20 = 100 Return interlinking depth in marker byte
100 if a fault occurs.

Command 2: Unrestricted Receive

DW32 - DW36, DW62 = Reserved for R5REQ
DW38 = 'RX' Command identifier
DW40 = 10 Monitoring time = 100 ms
DW42 = 100 Size of the PLC Receive buffer = 100
DW44 = 150 Receive-data length = 150 characters

Command 3: Data Copy

DW64 - DW68, DW94 = Reserved for R5REQ
DW70 = 'DC' Command identifier
DW72 = K44H Identifier
DW74 = 11 for
DW76 = 0 PLC Receive buffer

Command 4: Start

DW96 - DW100, DW126 = Reserved for R5REQ
DW102 = 'ST' Command identifier
DW104 = K44H Identifier
DW106 = 10 for
DW108 = 0 Command chain
DW110 = 2 No. of the job parameter lists which should be transferred
DW112 = K0201H Co-position number of commands
First Receive Mode, then Receive
DW114 = K00FFH End of command chain

Only the parameter list for the Start command is started with the R5REQ. The other commands in the command chain are started by BÜP03E according to the specified sequence. The "DW112 = K0201H" command sequence specifies that the command with position number 1 (Receive Mode) is executed before 2 (receive). The Data Copy command is started parallel to the command chain.

Example 4: A simple text combined with variables is sent.

DW0 - DW4 = Reserved for R5REQ

DW6 = 'FO' Command identifier

DW8 = K44H Identifier

DW10 = 11 on

DW12 = 0 control text with variables

DW14 = 50 Estimated length of control text + variables in bytes.

The control text and variables are in the DM11, commencing from DW0, e.g.

DW0 Temperature, required %2d degrees, actual %2d degrees,

DW38 20

DW40 25

The output then has the following format:

Temperature, nominal 20 degrees, actual 25 degrees

8.8 Checkback Signals and Fault Messages

The jobs can be started or terminated with the R5REQ function module.

The R5CON function module monitors the issued jobs. If the job has been terminated or terminated, BÜP03E enters the information into the job table on the ZS module via the processed jobs. R5CON reads this information which the user receives in his output parameters (P3, P4).

The R5CON function module has two input and three output parameters (see also the specification for the R5CON function module). These parameters are defined as follows.

P0 (word): CXN and channel number

P1 (word): Job number

P2 (word): Job number

P3 (word): Status

P4 (word): Fault message

P2 (job number) indicates to which job the words, P3 (status) and P4 (fault message), have been designated.

The BÜP03E protocol generates fault information which is divided into three areas:

- PLC response (byte)
- Error class (byte)
- Error code (byte)

If commands are interlinked, additional information is required to define the command in which the fault or job abort has occurred. This information, known as "Interlinking depth", is provided in a marker byte which is indicated in the Receive Mode command.

The existing error classes and those jointly used by BÜP03E are

- 30H System bus
- 31H Operating system

The error class

- 40H BÜP03E protocol

is defined for the BÜP03E itself.

The fault information is written to the R5CON checkback parameters in the following sequence:

P3 parameter

HIGH byte	LOW byte
Status	PLC response

P4 parameter

HIGH byte	LOW byte
Error class	Error code

8.8.1 Job Status

The current status of the job is specified in the Status by the function module:

Status	Explanation
00H	Job ended fault-free
01H	Job written to queue
02H	Job processing
03H	Job ended with fault(s)

Fig. 8-16 BÜP03E — Job status

8.8.2 PLC Response

The PLC response is intended to provide the user with advice and help on the faults within the fault information.

The following conventions apply:

PLC response	Explanation
00H	No response, job fault-free
01H	Repeat FM job
02H	Program fault, change PLC program
03H	Change protocol parameter
04H	Rerun the program cycle on the R5INIT
05H	Rerun the program cycle on the R5REQ
06H	Rerun the program cycle on the R5CON
07H	Reload the protocol
09H	Repeat the job
10H	Active job terminated by the user
20H	Job fault-free, however with reduced data length

Fig. 8-17 BÜP03E — PLC Response

8.8.3 Error Class

The error class refers to the generation point of the fault information. The following conventions apply:

Error class	Explanation
00H	Fault-free
30H	System bus
31H	Operating system
40H	BÜP03E fault messages
FFH	Fault message of the function modules

Fig. 8-18 BÜP03E — Error class

8.8.4 Error Catalogue

Fault information			Explanation
PLC response	Error class	Error code	
00	00	00	Fault-free job
00	30	00	Receive acknowledgement
00	30	01	Command executed fault-free
00	31	00	No fault
02	40	05	Command already active - no start
03	30	90	Field type undefined on receiving module
03	30	92	Invalid command parameter
03	30	93	Wrong address alignment
03	30	94	Address fault
03	30	D6	Module number too large
03	30	D7	Module not available
03	30	D8	Data module too small
03	31	56	Destination module unknown
03	40	01	Parameter fault in the job string
03	40	02	Data loss in the BÜP03E Receive buffer
03	40	08	FO - format fault
03	40	09	FO - no or only separator
03	40	0A	FO - too many variables
05	40	03	Receive buffer empty
05	40	06	UART transmit/receive fault
09	31	50	System bus not allocated
09	31	52	Partner command buffer full
09	31	53	Command execution interval elapsed
09	40	07	Receive active - no mode change
10	31	60	Passive job terminated
10	31	61	Active job terminated
10	40	04	Abort via abort character

Fig. 8-19 BÜP03E — Fault messages

9 BUEP64 Protocol

9.1 Introduction

A frequent application occurring in automated systems is the data exchange between complex or spatially distributed control systems.

This section discusses the BÜP64 protocol (Siemens 3964 / 3964R protocol) which can be loaded into the R500 and R500P computer modules. The BÜP64 protocol transfers data across a point-to-point-connection, as a result of which data can be exchanged between two modules with implemented "3964" or "3964R" protocol; the principal difference between the two protocols is that "3964R" for data security transmits a block checksum according to the control sequence DLE ETX or DLE ETB and uses an acknowledgement delay time of 2 s, whereas "3964" dispenses with a block checksum and uses an acknowledgement delay time of 550 ms.

The user selects the communication protocol, "3964" or "3964R", by setting the S2 DIP switch.

The user controls the "3964R / 3964" protocols by means of the standard function modules R5INIT, R5REQ and R5CON:

R5INIT: initialises the computer modules concerned with data traffic;

R5REQ: starts a protocol job;

R5CON: acknowledges started protocol jobs;

The abbreviations **RST** and **PST** occur frequently throughout this documentation and are defined as follows:

RST: Requesting station
specifies the communications partner which starts the protocol job;
PST: Peripheral station
specifies the communications partner which responds to a received protocol job;

9.2 Protocol-specific DIP Switch Settings

The S2 DIP switch can be defined for the BÜP64 protocol as follows:

Switch	Setting	Explanation	Channel
SS1	ON OFF	Peripheral jobs have priority Centralised jobs have priority	0
SS2	ON OFF	Protocol 3964 Protocol 3964R	0
SS3	ON OFF	Protocol end identifier incl. DLE ETB Protocol end id. always DLE ETX	0
SS4	OFF	No function	0
SS5	ON OFF	Peripheral jobs have priority Centralised jobs have priority	1
SS6	ON OFF	Protocol 3964 Protocol 3964R	1
SS7	ON OFF	Protocol end identifier incl. DLE ETB Protocol end id. always DLE ETX	1
SS8	OFF	No function	1

Fig. 9-1 BÜP64 — S2 DIP switch settings

NOTE –

When installing the link, ensure that the low priority is allocated to one device and the high priority to the other device.

9.2.1.1 Protocol End Identifier DIP Switch

The introduction of a selecting option for the protocol end identifier (DLE ETX or DLE ETB) for the R500/R500P is necessitated by the varying end identifiers used by several Siemens-made components:

- the 3964R data transfer controller for the Siemens process computer uses the DLE ETB identifier for transfer completion sequences.
- the CP524 and CP525 communication processors (interfaced with a Siemens Simatic controller) use the DLE ETX identifier for both completion sequences and transfer response message

The BÜP64 protocol, which has been implemented on the BOSCH CL300 and PC600 controllers, transmits the DLE ETX identifier, although it accepts both end identifiers when messages are being received.

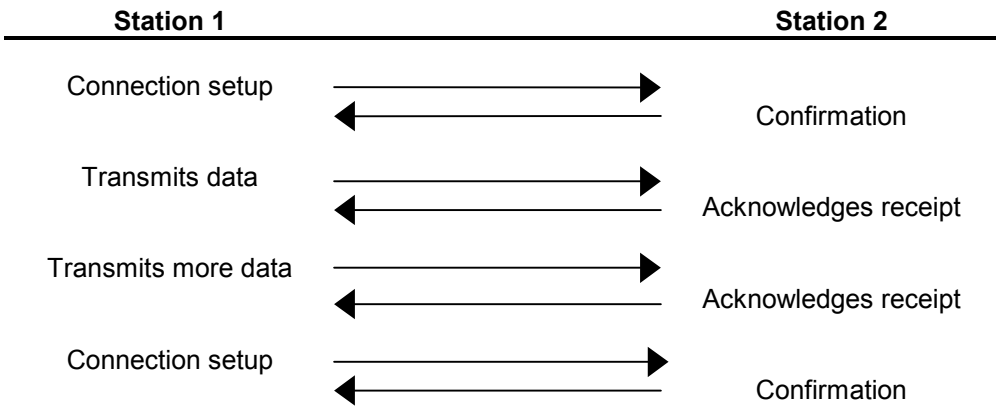
Unlike on any other interface modules, the R500/R500P interface modules enable DIP switch selection of the appropriate end identifier for each intended mode of communication; the R500/R500P is therefore able to communicate with all modules!

To enable interfacing with BOSCH controllers, the end identifier DIP switch (SS3, SS7) is set to OFF (DLE ETX)!

9.3 BÜP64 Transfer Completion Sequence Handling

Information is usually transferred according to the following sequence:

- The connection is established.
- Data blocks, max. 128 data bytes, are transferred.
- Each data block is acknowledged by a subsequent transfer response message.
- The individual blocks are repeated if transfer faults occur.
- After transfer of the final data block, the connection enters an idle state.



Data is requested according to the same procedure.

The 3964R protocol is best divided into two levels.

Interpreter

The Interpreter manages the data exchange, and processes and monitors the centralised or decentralised jobs. It establishes a connection with the PLC, processes the data and ensures that data is transferred in blocks. The Interpreter recognises faults and signals a defective or fault-free data exchange to the PLC.

The Interpreter calls up the Transfer Routine with all data required for its functioning.

Transfer Routine

The Transfer Routine handles the actual acknowledgement interchange on the line. Once initiated by the Interpreter, the Transfer Routine completes the transfer automatically

If transfer faults occur, the defective block is repeated. The Transfer Routine signals a fault-free and/or defective data exchange to the Interpreter.

To facilitate understanding of the description of the following sequences, the abbreviations used therein are briefly explained:

RST	=	Requesting station
PST	=	Peripheral station
STX	=	Start of Text, control character for initiating the data transfer.
ETB	=	End of Block, control character
ETX	=	End of Text, control character
BCC	=	Block Check Character" block check character, checksum of the transferred data.
DLE	=	Data Link Escape, control character. NOTE – If this identifier occurs as useful data, it is transmitted a second time ==> DLE doubling. DLE doubling is included in the checksum.
CDT	=	Character Delay Time, maximum time between two characters. This time is monitored by the receive device (220 ms).
ADT	=	Acknowledgement Delay Time, maximum time up to confirmation of a control character. ADT depends on the protocol: 3964 -> 550 ms 3964R -> 2000 ms
RDT	=	Response Delay Time, maximum time up to the start of a response message or transfer completion sequence (5 - 20 seconds, depending on the baud rate).

NAK = Negative Acknowledge, control character, negative response, e.g. if a transfer fault occurs.

BWT = Block Wait Time, max. 4 seconds, max. time until a defective block is repeated.

High / Low priority = Specifies which device has priority control when an access procedure is occurring.

Checksum = Secures the data and is transmitted at the end of each block. The block check character BCC is generated from the parity (*) of the information bits of all data bytes (incl. DLE doubling) in a transmitted or received block. The generation begins with the first user data byte and ends after the DLE ETX character. No restrictions have been specified for the information characters (code-transparent).

* Parity: XO (Exclusive OR) of all data bytes.

9.4 SEND Connection Sequence

The requesting station (RST) transmits the STX control character to establish the connection.

If the peripheral station (PST) responds before the acknowledgement delay time elapses with the DLE character, the RST switches over to Transmit Mode.

If the PST responds with the NAK character, with any other character, or if the acknowledgement delay time (ADT) elapses without any response, the connection setup fails.

After a total of three unsuccessful attempts, the procedure is terminated and a fault is signalled to the PLC.

If the connection setup is successful, the data is transmitted to the peripheral device. The peripheral monitors the intervals between the incoming characters.

The interval between two characters must not be more than the character delay time (CDT = 220 ms).

When the transmission is complete, the Transfer Routine adds the DLE, ETX and BCC characters as an end identifier and waits for an acknowledgement character. If the peripheral transmits the DLE character within the acknowledgement delay time ADT, the data block has been accepted fault-free. If the response was NAK, any other character, or if the time elapses without any response, the transmission is repeated. After a total of three unsuccessful attempts, the procedure is terminated and a fault is signalled to the PLC.

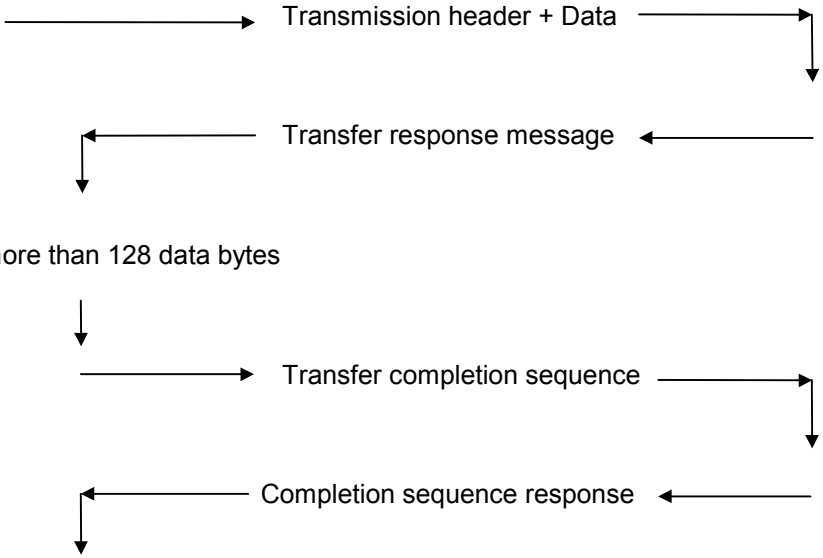
If the peripheral transmits the NAK character during a routine transfer, the current block is terminated and is repeated according to the abovementioned procedure

Up to 128 user data bytes can be transferred per message.

If a DLE occurs in the form of useful data, DLE is doubled.

Larger volumes of useful data are transferred in the form of one or more transfer completion sequences.

Transmission of data - Data is usually transmitted in the following sequence:



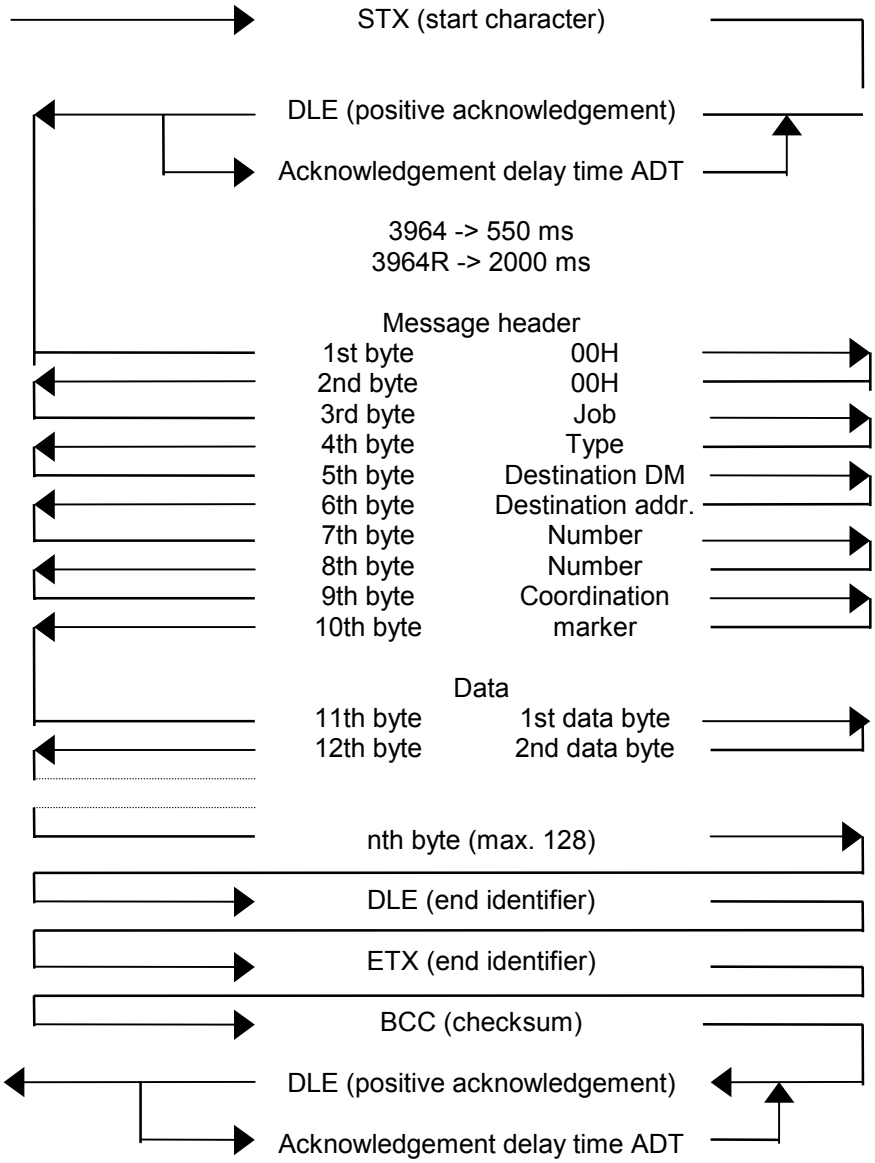
Source types	Transmission header Byte 3,4	Destination types
Data modules	Byte 3: 41H -> 'A' Byte 4: 44H -> 'DM'	Data modules

NOTE –

Only data modules are transferred, as the various operand formats of the individual controller types (CL400/500, CP525, CL300, PC600) cause misunderstandings.

9.4.1 Detailed SEND Sequence

9.4.1.1 Message Header and Data



9.4.1.2 Response Message

A response message is expected after a command message has been issued. Depending on the selected baud rate, the following times must be observed:

The monitoring time is dependent on the baud rate (bps).

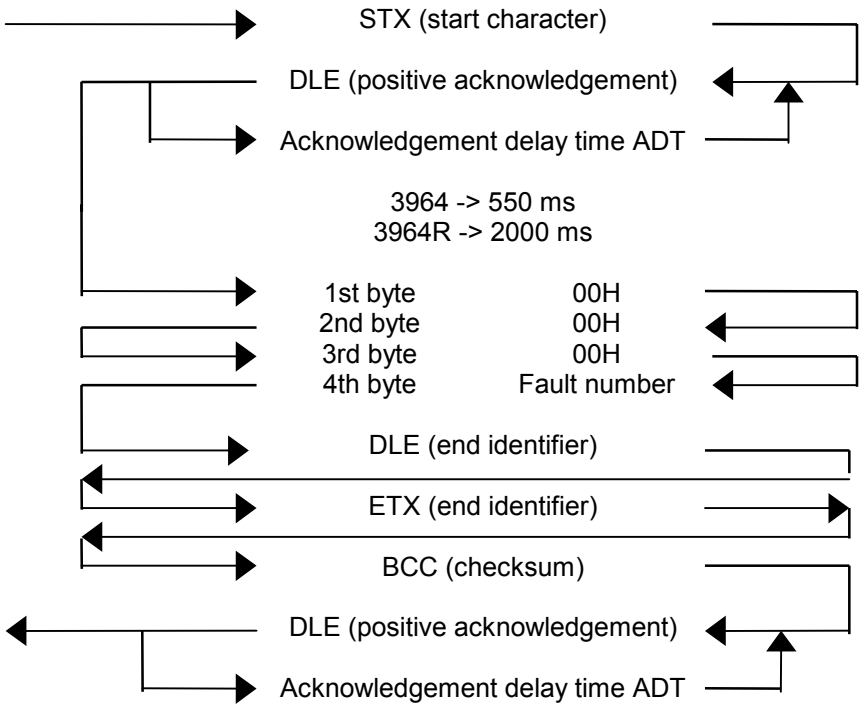
19200 - 1200 = 5 sec.

600 = 7 sec.

300 = 10 sec.

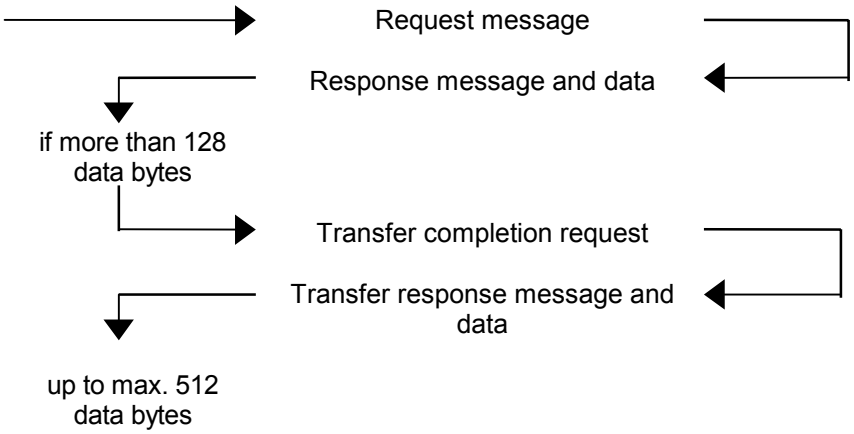
150 = 15 sec.

110 = 20 sec.



9.5 Requesting Data (FETCH)

Data is requested in the following sequence:



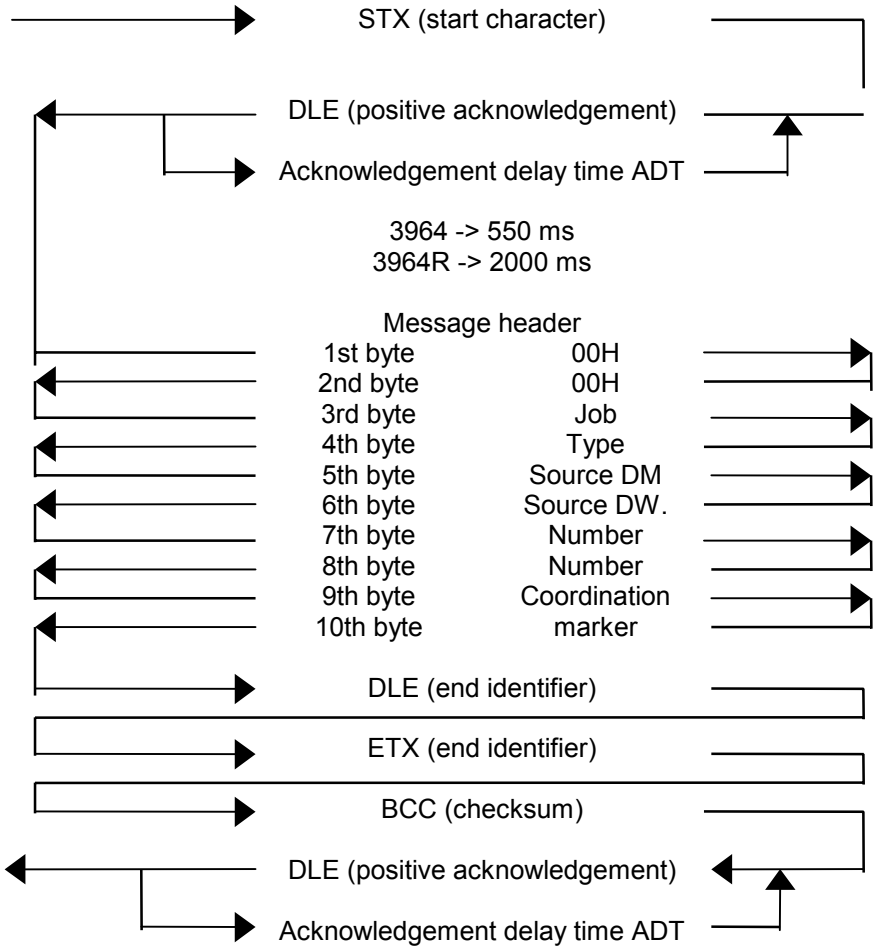
Source types	Message header Byte 3,4	Destination types
Data modules	Byte 3: 45H -> 'E' Byte 4: 44H -> 'DM'	Data modules

NOTE –

Only data modules are transferred !

9.5.1 Detailed REQUEST Sequence

9.5.1.1 Request Message



9.5.1.2 Response Message

A response message is expected after a command message has been issued. Depending on the selected baud rate, the following times must be observed:

The monitoring time is dependent on the baud rate (bps).

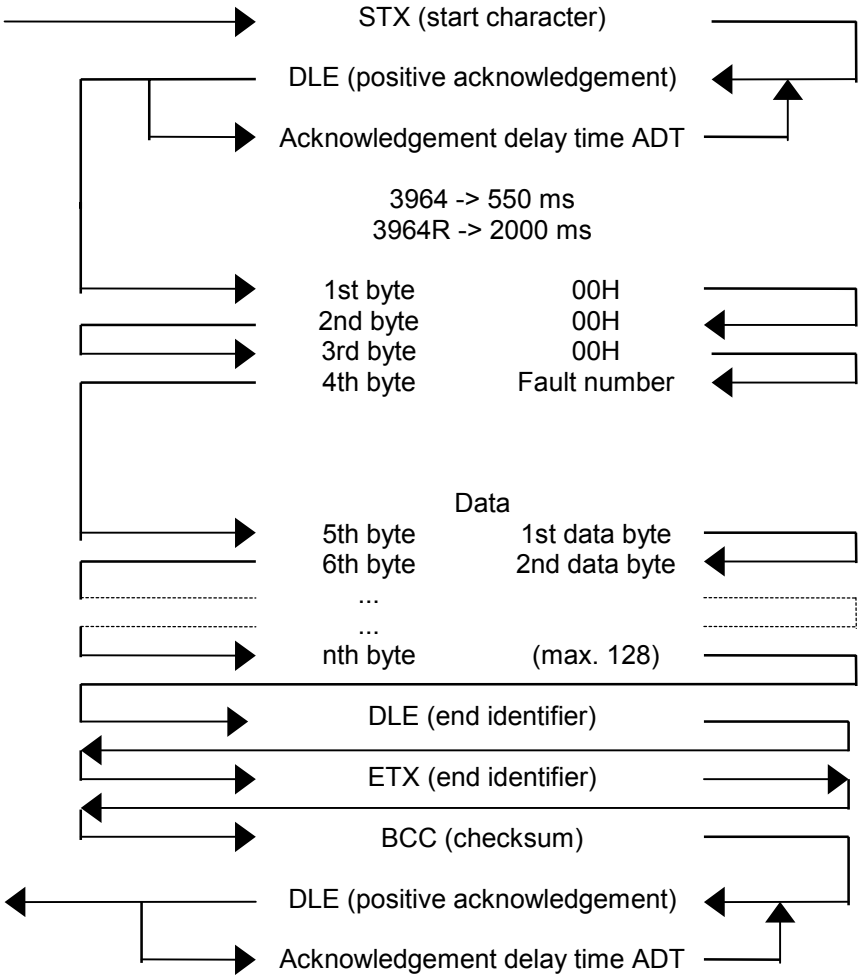
19200 - 1200 = 5 sec.

600 = 7 sec.

300 = 10 sec.

150 = 15 sec.

110 = 20 sec.



9.6 RECEIVE Connection Sequence

If there is no centralised job, the Transfer Routine remains in an idle state. If an STX is now received from the peripheral, a response can be implemented with DLE, i.e., ready-to-receive state is displayed if the Receive buffer is empty.

(All other characters have an NAK as a consequence and the Transfer Routine remains in an idle state).

If the buffer is not ready, another check is implemented after approx. 400 ms to determine whether the buffer is now empty (ADT = max. 550 ms).

If the buffer is still not ready, the connection setup fails. An NAK is transmitted.

The following receive characters are written to the buffer. The character delay time CDT is restarted after each received character. If the CDT elapses without another character being received, NAK is transmitted to the peripheral.

A DLE doubling is recognised as data and the block checksum is generated. A DLE doubling is simply accepted as user data.

When the character sequence DLE, ETX and BCC has been recognised, the Transfer Routine ends the Receive process and transmits DLE for a correctly received block or an NAK for an incorrectly received block.

If transmission faults occurred during the Receive process, the Receive process is continued until connection release occurs (DEL, ETX, BCC) and NAK is transmitted. A repeat of the block is expected.

If a new transfer is not started within a block wait time of 4 s, or the Receive process was not successful after a total of six attempts, the Transfer Routine aborts the Receive process and signals a fault to the PLC.

9.7 Initialisation Conflict

If a device responds to the start character (STX) by issuing the start character (STX) itself, an initialisation conflict occurs.

The device with the low priority resets its transmit job and signals its ready-to-receive state (DLE).

The low-priority device executes its transmit job following the receive and connection release procedures.

NOTE –

When installing the link, ensure that the low priority is allocated to one device and the high priority to the other device.

9.8 Quasi-Duplex Mode

Strict adherence to the request message - response message sequence is mandatory for the R500(P)!

Bidirectional data traffic in Quasi-duplex mode: if the R500(P) request messages are not directly answered with the response message from the PST (because the PST first transmits a request message), this R500(P) status is terminated with a fault message.

9.9 Transfer Message Formats

9.9.1 Transmission Header and Data

The message header consists of 10 bytes.

Byte		
1	00 (FF)	Identifier for message header (transfer completion sequence header)
2	00	
3	(ASCII)	Command
4	(ASCII)	Command type
5	(High)	Destination or source identifier
6	(Low)	
7	(High)	Data volume
8	(Low)	
9	(High)	Coordination marker
10	(Low)	
11 12 . . n		Data
n+1 n+2 n+3		DLE end identifier, ETX/ETB BCC

Byte designations:

-1- Message identifier 00 (or FF for transfer completion sequence)
-2- Message identifier (00)
-3- Command, SEND = 'A', FETCH = 'E'
-4- Command type, type of data to be transferred
'D' = Data module
-5,6- Destination address for SEND, source address for FETCH
-7,8- Number of user data to be transferred in words.
-9,10- Coordination marker

9.9.2 Response Message Structure

Byte	
1 00	Identifier for response message
2 00	
3 00	
4 00	Fault number (0 = no fault)
5	DLE end identifier,
6	ETX
7	BCC

9.9.3 Response Message and Data Structure

Byte	
1 00 (FF)	Identifier for response message (transfer response message)
2 00	
3 00	
4 00	Fault number (0 = no fault)
5	Data
.	
.	
.	
n	
n+1	DLE end identifier,
n+2	ETX
n+3	BCC

9.10 Checkback Signals

The BÜP64 protocol generates fault information which is divided into three areas:

PLC response (byte)

Error class (byte)

Error code (byte)

The fault information is written to the R5CON checkback parameters in the following sequence:

P3 parameter

HIGH byte	LOW byte
Status	PLC response

P4 parameter

HIGH byte	LOW byte
Error class	Error code

9.10.1 Job Status

The current status of the job is specified in the Status by the function module:

Status	Explanation
00H	Job ended fault-free
01H	Job written to queue
02H	Job processing
03H	Job ended with fault(s)

Fig. 9-2 BÜP64 — Job status

9.10.2 PLC Response

The PLC response is intended to provide the user with advice and help on the faults within the fault information.

The following conventions apply:

PLC response	Explanation
00H	No response, job fault-free
01H	Repeat FM job
02H	Program fault, change PLC program
03H	Change protocol parameter
04H	Rerun the program cycle on the R5INIT
05H	Rerun the program cycle on the R5REQ
06H	Rerun the program cycle on the R5CON
07H	Reload the protocol
10H	Active job terminated by the user
20H	Job fault-free, however with reduced data length

Fig. 9-3 BÜP64 — PLC response

9.10.3 Error Class

The error class refers to the generation point of the fault information.

The following conventions apply:

Error class	Explanation
00H	Fault-free
30H	System bus
31H	Operating system
32H	PST job execution
33H	RST job execution
34H	Protocol sequence
FFH	Fault message of the function modules

Fig. 9-4 BÜP64 — Error class

9.10.4 Fault Messages

Fault information			Explanation
PLC re-sponse	Error class	Error code	
00	00	00	Fault-free job
09	30	50	System bus not allocated; fault signal during transmission of the command
09	30	52	Full command buffer on system bus partner
09	30	53	Expired command execution interval
09	30	54	Full CXN acknowledgement wait buffer
03	30	56	Unknown destination module
10	31	60	Passive job terminated
10	31	61	Active job terminated
03	32	0A	Source/destination type illegal when R500/R500P acts as PST: - Field type not defined on receiver module - Memory not available or wrong type
03	32	0C	Data-type or range fault when R500/R500P acts as PST: - Wrong address alignment - Address fault: address access refused
03	32	10	Fault in command byte
09	32	12	Fault in system command when R500/R500P acts as PST - System bus not allocated - Partner command buffer full - Command execution interval expired - Full CXN acknowledgement wait buffer - Destination module unknown
03	32	14	Data module not permitted when R500/R500P acts as PST: - Data module number too large - Data module not available - Data module too small
09	32	16	Transfer completion sequence not equal to command messg.
09	32	2A	Command rec'd, but in STOP when R500/R500P acts as PST: - Data access via system bus blocked - Fault signal active during data access
02	32	32	Data module blocked by coordination marker when R500/R500P acts as PST: - Access type prohibited (read, write) - Field protection active; i.e., access to field type blocked by user command
03	32	34	Length fault when R500/R500P acts as PST: - Buffer overflow
09	32	36	Synchronisation fault; e.g. transfer completion sequence precedes command message
03	33	01	Module no. does NOT exist for indicated block address
03	33	20	Wrong command type (only DM)
03	33	29	Wrong command type (only I/O)
03	33	3D	Wrong operand type (only word)
03	33	91	Execute condition not defined; system bus command cannot be executed on addressed module
03	33	92	Invalid command parameter
03	33	93	Wrong address alignment
03	33	94	Address fault: address access refused
03	33	95	Memory not available or wrong type

Fig. 9-5 BÜP64 — Fault messages, Part 1

Fault information			Explanation
PLC response	Error class	Error code	
03	33	96	Buffer overflow
03	33	B0	Data access via system bus blocked
03	33	B1	Fault signal active during data access
03	33	D3	Access type prohibited (read, write, bit access)
03	33	D4	Field protection active
03	33	D6	Data-module number too large
03	33	D7	Data-module not available
03	33	D8	Data module too small
03	33	ED	Wrong PST-DM number
03	33	EC	Wrong PST-DM address
03	33	F1	Own source / destination address defective
03	33	F2	Address of data module/word wrong
03	33	F3	Volume of data defective
04	34	F7	Time-monitoring fault at protocol level
04	34	F9	Permitted number of attempts exceeded
09	34	FE	Fault while opening the line; received character not equal to STX
09	34	FF	No interpretable message received
20	Length		Reduced length was transferred

Fig. 9-6 BÜP64 — Fault messages, Part 2

9.11 Job Parameters

A job is specified by 16 parameters. Each parameter has a width of 2 bytes. Data ranges of 16 data words each are available for the job parameters; a maximum of 16 jobs can be parameterised for each call-up of the R5REQ.

The parameter words PW1...PW3 are protocol-independent and job-specific parameters; while the parameter words PW4...PW16 contain protocol-specific information.

The parameters can be written to the data module, the data field or data buffer; the block start address of 16 data words (PW1...PW16) is always an even byte address.

The entries for the 3964R can be explained as follows:

Parameter word	Explanation
PW1	Job result (protocol-independent)
PW2	High byte: CXN (protocol-independent) LOW byte: Channel number (protocol-independent)
PW3	Job number (protocol-independent)
PW4	High byte: Command (SEND ‘A’ / FETCH ‘E’) LOW byte: Operand type = word
PW5	High byte: Command type RST (‘DM’) LOW byte: Command type PST (‘DM’)
PW6	RST address: DM number
PW7	RST address: Block address
PW8	RST address: DM word number
PW9	PST address: DM number
PW10	Free
PW11	PST address: DM number
PW12	Volume of data (in words)
PW13	Coordination marker
PW14	Free
PW15	Free
PW16	RESERVED

Fig. 9-7 BÜP64 — Job parameters

Several examples illustrating parameterisation of various centralised jobs can be found on Page 9-31 ff.

9.11.1 Protocol-specific Parameters — PW4 through PW16

Data traffic is parameterised by "3964R" for RST **and** PST via the protocol-specific parameters. The individual parameters are explained below.

9.11.1.1 Command / Operand Type — PW4

The protocol provides the SEND and FETCH commands (PW4 HIGH byte):

Parameter word	Contents	
	High byte	LOW byte
PW4	Command type	Operand type
Example	41H	01H
	Send	Word

Fig. 9-8 BÜP64 — Operand type

- SEND --> "A" = 41 hex

  Data is transmitted from the RST to the PST

- FETCH --> "E" = 45 hex

  Data is fetched from the PST and written to the RST

Data exchange between data modules is only permitted when BÜP64 is implemented on the R500/R500P; the operand type (PW4 LOW byte) is fixed for this reason.

- Operand type = WORD (01 hex)

9.11.1.2 RST / PST Command Type — PW5

Parameter word	Contents	
	High byte	LOW byte
PW5	RST command code	PST command code
Example	44H	44H
	Data module	Data module

Fig. 9-9 BÜP64 — RST/PST Command type

Only the "Data module (DM = 44 hex)" command types exist for RST and PST when the "3964R" protocol is implemented; the following situations may therefore occur:

- SEND: Transmit data from the RST data module to the PST data module
- FETCH: Fetch data from the PST data module and write it to the RST data module

9.11.1.3 RST Address: DM Number — PW6

Parameter word	Contents
PW6	RST address
Example	AH
	Data module 10

Fig. 9-10 BÜP64 — RST Address: DM Number

The PW6 parameter word indicates the data-module number for the RST; data modules DM0 to DM255 can be addressed.

9.11.1.4 RST Address: Block Address of Central Processing Unit — PW7

Parameter word	Contents
PW7	RST address
Example	8H
	ZS1

Fig. 9-11 BÜP64 — RST Address: Block address

The block address of the respondent central processing unit is used to address the RST in the CL500 system;

NOTE –

In comparison, the central processing unit on the PST is addressed in the coordination marker (please refer to the description of the *Siemens 3964R/3964 Data Transfer Controller*).

9.11.1.5 RST Address: DM Word Number — PW8

Parameter word	Contents
PW8	RST address
Example	0H
	Word 0

Fig. 9-12 BÜP64 — RST Address: DM word number

Like the parameterisation of the PST DM word number (P10), the RST DM word number is specified in one byte (0,1,2...255) in order not to confuse the user with possibly varying parameter specifications; the word number is converted into the CL500 format (byte address 0, 2, 4 ,6 ....510) within the protocol;

9.11.1.6 PST Address: DM number — PW9

The PW9 parameter word indicates the data-module number for the PST; data modules DM0 to DM255 can be addressed.

Parameter word	Contents
PW9	PST address
Example	AH
	Module 10

Fig. 9-13 BÜP64 — PST Address: DM number

9.11.1.7 PST Address: DM Word Number — PW11

1 byte is available in the message for transfer of the DM offset; the offset is therefore specified as a word number (0..255); the word numbers have the following format for the various peripheral controllers:

- CL500 Word number 0, 1, 2, 3 ... 255
- PC600 Word number 0, 1, 2, 3 ... 255
- Siemens Word number 0, 1, 2, 3 ... 255
- CL300 Word number 0, 1, 2, 3 ... 127

Parameter word	Contents
PW11	PST address (word number)
Example	4H
	Word number 4

Fig. 9-14 BÜP64 — PST Address: DM word number

9.11.1.8 Data Volume — PW12

The PW12 parameter word indicates the volume of data which is to be transferred; the specification refers to the operand type (word) specified in PW4 (LOW byte); a maximum of 256 words can be transferred;

Parameter word	Contents
PW12	Volume of data
Example	20H
	32 words

Fig. 9-15 BÜP64 — Data volume

9.11.1.9 Coordination Markers — PW13

The function of the coordination markers is to coordinate the execution of a command on the PST. The command is only executed when the corresponding coordination event is "true"; the coordination sequence point or the field coordination marker "1" must therefore be accessed for the CL500 control type; as far as the other specified control types are concerned, the addressed coordination marker must be "1" for execution of a coordinated command.

CL500 controller acting as PST

The Central Processing Unit Control command in the CL500 system uses two types of coordination markers:

- Process coordination marker (PCM) (as of Version 1.2)

and

- as of ZS501: Field coordination marker (FCM)

The PCMs are specified as follows:

00 H	System STOP status
01 H	System RUN status
02 H	I / O status
03 H	I / O status or STOP
04 H	EP
05 H	EP or STOP
06 H	OM1
07 H	OM1 or STOP
0F H	Process coordination not permitted in conjunction with reset PCM disable flag

The FCMs are specified in the special marker area as follows:

- SM16.0 .. SM17.7 single FCM;
  these FCMs are reset by the system following execution of the coordinated command;

- SM18.0 .. SM19.7 permanent FCMs;
  are <u>not</u> reset by the system following execution of the coordinated command;

The FCMs are coded as follows:

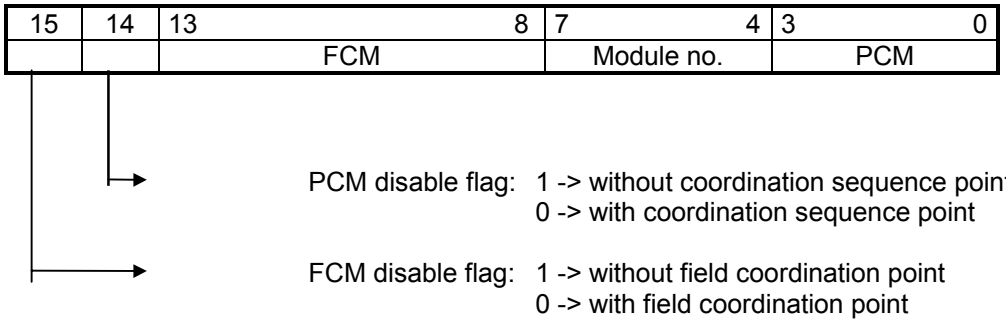
SM16.0 .. SM 17.7 ---------> FCM 0 H .. F H

SM18.0 .. SM 19.7 ---------> FCM 10 H .. 1F H

No field coordination --------> FCM FF H

The respondent ZS in the CL500 system is addressed by the ZS (1H-FH) module number which is indicated in the coordination marker. This addressing procedure is based on the *Siemens 3964R/3964 Data Transfer Controller* specification.

The process and field coordination markers, as well as the module number, are linked to the "3964R" coordination marker format according to the following schematic:



Module no.: Module number of the respondent central processing unit

NOTE –

FFFFH in the coordination marker denotes:
Command not coordinated to module no. 0.

This can be used effectively for the ZS400.

NOTE –

If a Siemens controller is used as the RST communications partner, the following restriction must be considered on account of the parameterisation options in the Siemens controller:

- Only the process coordination markers 0..7 can be parameterised;
- In order to address all the central processing units, they must be indicated with module numbers 1-4 in the SC system configuration table;

The following coordination marker bytes for the Siemens controller are proposed for the coordination of central processing units:

Coord. marker	Explanation
00 x 0 .. 7 . . 1F x 0 .. 7	Process and field coordination: - 00 x 0 --> SM16.0 and PCM 0 - 0A x 4 --> SM17.2 and PCM 4 - 1F x 7 --> SM19.7 and PCM 7
40 x 0 . 5F x 0	Only field coordination: - 40 x 0 --> Special marker SM16.0 - 5F x 0 --> Special marker SM19.7
80 x 0 .. 7	Only process coordination 0 .. 7
FF x F	No coordination

x denotes the specification of module numbers 1-4, described at the start of this Section.

CL300 controller acting as PST

The CL300 controller permits a central processing unit to have maximum configuration; it is therefore not necessary to specify the CPU number within the coordination marker.

When the Siemens format is observed, the coordination marker has the following appearance:

15	7 4	3 0
Word address	0	Bit number

Where: bit number - 0...15
word address - 0...254
word address 0, 2, 4 ... 254

PC600 controller acting as PST

The PC600 controller permits a central processing unit to have maximum configuration; it is therefore not necessary to specify the CPU number within the coordination marker.

When the Siemens format is observed, the coordination marker has the following appearance:

15	7 4	3 0
Word address	0	Bit number

Where: bit number - 0...15
word address - 0...255
word address 0, 1, 2, 3 ... 255

Siemens CP524/CP525 controller acting as PST

When these Siemens controllers are used as a PST, the coordination marker (CM) contains information on the CPU number, the CM byte and the CM bit;

- Byte number of the CM: If no CM is used, FFH must be in this location;

  the byte number must be within the range of 0-223 for the CP524 and CP525;

- Bit number of the CM: If no CM is used, FH is in this location;

- CPU number: If the number of the respondent CPU is indicated as a number between 1 and 4;

  If no CPU number, but a CM, is indicated, 0H is in this location;
  If no CPU number and no CM are indicated, FH is in this location;
  In both cases all CPUs are possible for the partner.

15	7 4	3 0
Byte number	CPU number	Bit number

9.12 Job Parameterisation Examples

9.12.1 SEND Job (CL500 -> CL300)

Nine (9) words, beginning with the byte address 26H (word number 13H), are written, without coordination, from the DM17 data module in the ZS500 with block address 10H to the DM35 data module in the ZE301 from byte address 64H (word number 32H). The transfer is processed by Channel 1 of the R500 with the number 2 (CXN2, see R5INIT parameterisation); the job receives job number 4.

The job is parameterised as follows:

Param. word	Contents	Explanation
PW1	---	No entry
PW2	0201H	CXN2, Channel 1
PW3	4H	Job number 4
PW4	4101H	Output job (SEND) / word
PW5	4444H	RST code: DM PST code: DM
PW6	11H	RST address: DM number 17
PW7	10H	RST address: Block addr. ZS2 = 10H
PW8	13H	RST address: Data-word no. 19
PW9	23H	PST address: Data-word no. 35
PW10	---	No entry
PW11	32H	PST address: Data-word no. 50
PW12	9H	Volume of data: 9 words
PW13	FFFFH	Without coordination
PW14	---	No entry
PW15	---	No entry
PW16	---	No entry

Fig. 9-16 BÜP64 — Example of SEND parameters (CL500 -> CL300)

9.12.2 FETCH Job (CL500 <- CL300)

Thirty-three (33) words, beginning with the byte address 1AH (word number DH, see Chap. 9.2), are read from the DM12 data module in the ZE301 and written, without coordination, to the DM27 data module in the ZS3 with block address 18H from byte address 50H (word number 28H, see Chap. 9.2).

The transfer is processed by Channel 0 of the R500 with the number 3 (CXN3 see R5INIT parameterisation).

The job receives job number 7.

The job is parameterised as follows:

Param. word	Contents	Explanation
PW1	---	No entry
PW2	0300H	CXN3, Channel 0
PW3	7H	Job number 7
PW4	4501H	Output job (FETCH) / word
PW5	4444H	RST code: DM PST code: DM
PW6	1BH	RST address: DM number 27
PW7	18H	RST address: Block addr. ZS3 = 18H
PW8	28H	RST address: Data word no. 40
PW9	CH	PST address: DM no. 12
PW10	---	No entry
PW11	DH	PST address: Data word no. 13
PW12	21H	Volume of data: 33 words
PW13	FFFFH	Without coordination
PW14	---	No entry
PW15	---	No entry
PW16	---	No entry

Fig. 9-17 BÜP64 — Example of FETCH parameters (CL500 <- CL300)

9.12.3 SEND Job (CL500 -> CL500)

Four (4) words, beginning with the byte address 12H (word number 9H), are written, without coordination, from the DM5 data module in the ZS500 with block address 10H to the DM9 data module in the ZS500 with module number 2 from byte address 24H (word number 12H).

The transfer is processed by Channel 1 of the R500 with the number 2 (CXN2 see R5INIT parameterisation).

The job receives job number 4.

The job is parameterised as follows:

Param. word	Contents	Explanation
PW1	---	No entry
PW2	0201H	CXN2, Channel 1
PW3	4H	Job number 4
PW4	4101H	Output job (SEND) / word
PW5	4444H	RST code: DM PST code: DM
PW6	5H	RST address: DM number 5
PW7	10H	RST address: Block addr. ZS2 = 10H
PW8	9H	RST address: Data word no. 9
PW9	9H	PST address: DM no. 9
PW10	---	No entry
PW11	12H	PST address: Data-word no. 18
PW12	9H	Volume of data: 9 words
PW13	FF2FH	Without coordination, module number 2
PW14	---	No entry
PW15	---	No entry
PW16	---	No entry

Fig. 9-18 BÜP64 — Example of SEND parameters (CL500 -> CL500)

10 Index of Figures

11 Index

1070 072 131-102 (96.08) GB • HB SP • AT/PLS • Printed in Germany