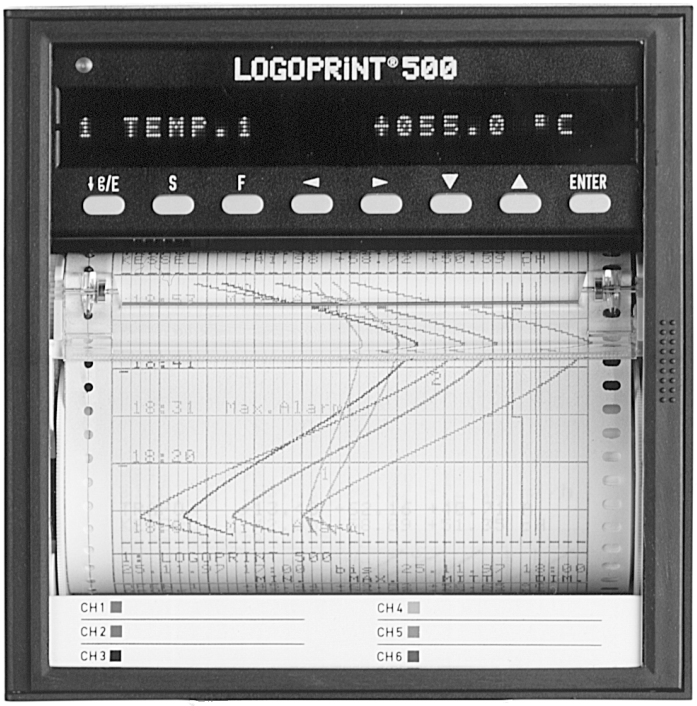


LOGOPRiNT® 500

Printing recorder with
text printing
and LED dot matrix display

B 95.4012.2

Interface description

4.98/00350647

Contents

Contents

1 Introduction

1.1 Preface

Please read these Operating Instructions before commissioning the interface. Keep the operating instructions in a place which is at all times accessible to all users.

Please assist us to improve these operating instructions where necessary.

Your suggestions will be most welcome.

Phone Germany (0661) 6003-725
abroad (int. +49) 661 6003-0
Fax Germany (0661) 6003-681
abroad (int. +49) 661 6003-607

All the necessary information for operating the interface is contained in these Operating Instructions. However, if any problems should arise during start-up, you are asked not to carry out any prohibited manipulations. You could endanger your rights under the warranty!

Please contact the nearest JUMO office or the main factory.

When returning chassis, modules or components, the rules of EN 100 015 “Protection of electrostatically sensitive devices” have to be observed. Use only the appropriate **ESD** pakkaging material for transport.

Please note that we can not be held liable for any damage caused by **ESD** (electrostatic discharges).

1.2 Typographical conventions

1.2.1 Warning signs

The signs for **Danger** and **Warning** are used in this manual under the following conditions:

Danger This symbol is used when there may be **danger to personnel** if the instructions are disregarded or not followed accurately!

Warning This symbol is used when there may be **danger to equipment or data** if the instructions are disregarded or not followed accurately!

Warning This symbol is used where special care is required when handling components liable to damage through electrostatic discharges.

1.2.2 Note signs

Note This symbol is used when your **special attention** is drawn to a remark.

Reference This sign refers to further information in other handbooks, chapters or sections.

abc[1] **Footnote** Footnotes are notes which refer to certain points in the text. Footnotes consist of two parts:

Marking in the text and the footnote text.

The markings in the text are arranged as continuous raised numbers.

The footnote text (in smaller typeface) is placed at the bottom of the text and starts with a number and a full stop.

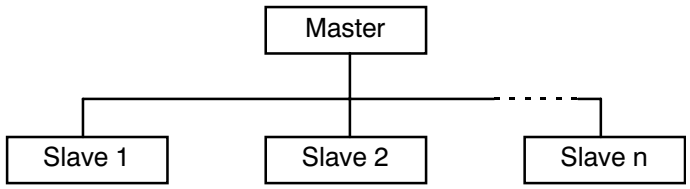
1.2.3 Presentation

0x0010 **Hexadecimal number** A hexadecimal number is identified by being preceded by a “0x” (here 16 decimal).

2 Protocol description

2.1 Master-slave principle

The communication between a PC (master) and an instrument (slave) using MODbus/Jbus takes place according to the master-slave principle in the form of data request/instruction - response.



The master controls the data exchange, the slaves only have a response function. They are identified by their instrument address. Up to 255 slaves can be accessed.

2.2 Transmission mode (RTU)

The transmission mode used is the RTU mode (Remote Terminal Unit). Data are transmitted in binary form (hexadecimal) with 8 bits, 16 bits for integers and 32 bits for float values.

Data format The data format describes the arrangement of a byte transmitted. The data format can be as follows:

Data word	Parity bit	Stop bit 1/2 bit	Bit number
8 bits	—	1	9
8 bits	—	2	10
8 bits	even	1	10
8 bits	odd	1	10
8 bits	always 0	1	10

2.3 Instrument address

The instrument address of the slaves can be set between 1 and 255 (decimal). The instrument address 0 is reserved.

☞ A maximum of 31 slaves can be addressed via the RS422/ RS485 interface.

The address is made in binary format (hexadecimal) in the transfer protocol.

2.4 Timing of the communication

The start and end of a datablock are identified by transmission pauses. The maximum permitted interval between two consecutive characters is three times the transmission time of a single character.

The character transmission time (the time taken to transmit one character) depends on the baud rate and the data format which is used.

For a data format of 8 data bits, no parity bit and one stop bit, this is:

character transmission time [msec] = 1000 * 9 bits / (baud rate)

For the other data formats it is:

character transmission time [msec] = 1000 * 10 bits / (baud rate)

Sequence

Data request from master transmission time = n characters * 1000 * x bits / (baud rate)
Marker for end of data request 3 characters * 1000 * x bits / (baud rate)
Processing of the data request by the slave (max. 250msec)
Response of the slave Transmission time = n characters * 1000 * x bits / (baud rate)
Marker for end of response 3 characters * 1000 * x bits / (baud rate)

Example

Marker for end of data request or response for 9 bits and 10 bits data format.

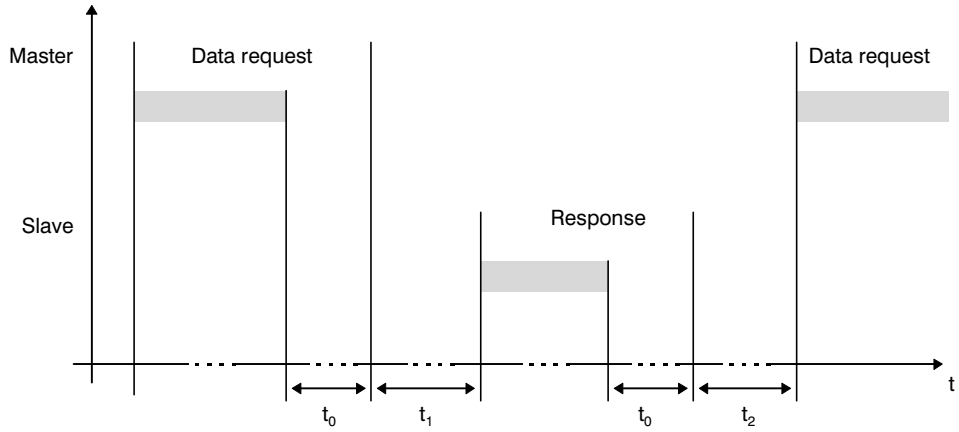
Waiting time = 3 characters * 1000 * 10 bits / (baud rate)

Baud rate [baud]	**Data format** [bit]	**Waiting time** [msec]
187k	10	0.160
	9	0.144
125k	10	0.240
	9	0.216
38400	10	0.781
	9	0.703
19200	10	1.563
	9	1.406
9600	10	3.125
	9	2.813
4800	10	6.250
	9	5.625
2400	10	12.500
	9	11.250
1200	10	25.000
	9	22.500
600	10	50.000
	9	45.000
300	10	100.000
	9	90.000
150	10	200.000
	9	180.000

2.4.1 Timing sequence of a data request

Timing scheme

A data request runs according to the following timing scheme:



t_0 End marker = 3 characters
(the time depends on the baud rate)

t_1 This time depends on the internal processing.
The maximum processing time is 250 msec.

☞ A minimum response time can be set in the instrument, under the menu item *Configuration level 3* ➔ *Interface.* This preset time is the minimum time which will be waited before a response is transmitted (0 — 500 msec). If a smaller value is set, then the response time may be longer than the preset value (because the internal processing time is longer) and the instrument answers as soon as the internal processing is completed. A preset time of 0 msec means that the instrument responds with the maximum possible speed.

The minimum response time which can be set is required by the RS485 interface in the master, in order to switch over the interface driver from receive to transmit. This parameter is not required by the RS422 interface.

t_2 This time is needed by the instrument to switch from transmit back to receive. The master must wait for this time before presenting a fresh data request. This time must always be observed, even when the data request is addressed to a different instrument.

RS422 interface: t_2 = 1msec
RS485 interface: t_2 = 10 msec

2.4.2 Communication during the internal processing time of the slave

No data requests from the master are permitted during the internal processing time of the slave. Any data requests which are made in this period will be ignored by the slave.

2.4.3 Communication during the response time of the slave

No data requests from the master are permitted during the response time of the slave. Any data requests which are made in this period will result in the invalidation of all data on the bus at that moment.

2.5 Structure of the data blocks

All data blocks have the same structure:

Data structure

Slave address	Function code	Data field	Checksum CRC16
1 byte	1 byte	x byte(s)	2 bytes

Each data block contains four fields:

Slave address instrument address of a specific slave

Function code function selection (read, write, bit, word)

Data field contains the information:

- bit address (word address)
- bit number (word number)
- bit value (word value)

Checksum recognition of transmission errors

2.6 Error handling

Error codes

There are five error codes:

1 invalid function

2 invalid parameter address

3 parameter value outside range of values

4 slave not ready

8 write access to parameter is denied

Response to error

Slave address	Function XX OR 80h	Error code	Checksum CRC16
1 byte	1 byte	1 byte	2 bytes

The function code is linked by OR with 0x80, i.e. the MSB (most significant bit) is set to 1.

Example

Data request:

01	02	00	00	00	00	CRC16

Response:

01	82	01	CRC16

Special cases

The slave does not respond to the following errors:

- the checksum (CRC16) is incorrect
- the instruction of the master is incomplete or over-defined
- the number of the words or bits to be read is zero
- the instrument is being configured from the keypad or the setup program

2.7 Distinction MODbus/Jbus

The MODbus protocol is compatible with the Jbus protocol. The structure of the data blocks is identical.

☞ MODbus differs from Jbus in the absolute addresses of the data. The addresses of the MODbus are shifted by one compared with those of the Jbus.

Absolute address	Jbus address	MODbus address
0	1	0
1	2	1
2	3	2
...	...	...

2.8 Checksum (CRC16)

The checksum (CRC16) serves to recognise transmission errors. If an error is identified in the evaluation, the corresponding instrument does not answer.

Calculation scheme

CRC = 0xFFFF			
	CRC = CRC XOR ByteOfMessage		
	For (1 to 8)		
		CRC = SHR(CRC)	
		if (flag shifted right = 1)	
		then CRC = CRC XOR 0xA001	else
while (not all ByteOfMessage processed);			

☞ The low byte of the checksum is transmitted first.

Example 1

Read out event counter 2 (current counter reading = 12345).

Data request: Read two words, starting at address 0x42 (CRC16 = 0xDA66)

14	03	0042	0002	66DA

Response: (CRC16 = 0x92BB)

14	03	04	E400	4640	BB92
			Word 1	Word 2	

Word 1 and Word 2 result in the answer 12345.0

Example 2

Request status of the open-collector outputs.

Instruction: Read one word, starting at address 0x3F (CRC16 = 0xC3B6)

14	03	003F	0001	B6C3

Response (CRC = 0x4774):

14	03	02	0001	7447
			Word 1	

Word 1 results in the setting of output 1 only

2.9 Configuration of the interface

Configuration of the interface is by the eight keys of the printing recorder or by the optional setup program.

Configuration via the printing recorder

First the configuration level 3 has to be called up and the parameter “Interface” selected. The parameters for configuring the interface are now available.

Configuration level 3 ➔ Interface		
Protocol:	J-BUS, MOD-BUS	Set protocol
Baud:	0.15, 0.3, 1.2, 2.4, 4.8, 9.6, 19.2, 38.4, 125, 187.5 kbaud	Select baud rate
Data format:	8/1/NONE, 8/1/ODD, 8/1/EVEN, 8/2/NONE, 8/1/ZERO	Select data format (Data bit/stop bit/parity)
Address:	1 — 255	Select address
Minimum response time:	0 — 500 msec	Select minimum response time

Configuration via the setup program

The menu item *Edit* ➔ *Interface (RS422/485)* is used for configuration through the setup program.

3 Functions

With the functions described below it is possible to:

- read out the current measurements
- monitor the operating status of the printer
- transmit values to the printer for output to paper
- transmit texts to the printer for output to paper
- send texts to the printer for display

Functions

The following functions are available for the instrument:

Function number	**Function**	
0x01/0x02	read n bits	(max. 256 bits)
0x03/0x04	read n words	(max. 80 words)
0x05	write one bit	
0x06	write one word	
0x0F	write n bits	(max. 256 bits)
0x10	write n words	(max. 80 words)

There are no separate areas for bit and word for the system variables. The bit and word areas overlap and can be read and written both as bit area or as word area.

Address calculation

The word address is calculated as follows:

word address = base address + variables address

The bit address is calculated as follows:

bit address = word address * 16 + bit number

Example: word address of the logic inputs:

word address = 0x002F + 0x000E = 0x003D

Example: bit address of the logic input no. 3

bit address = (0x002F + 0x000E) · 0x0010 + 0x0002 = 0x03D2

3.1 Read n bits

This function reads n bits, starting from a defined address.

Data request

Slave address	Function 0x01 or 0x02	Address first bit	Number of bits	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Response

Slave address	Function 0x01 or 0x02	Number of bytes read	Bit values	Checksum CRC16
1 byte	1 byte	1 byte	x byte(s)	2 bytes

Example

Read the status of the first 4 logic inputs (process data)
⇨ Section 6.1.2 “Process data”

Bit address = (base address + process data address) * 16 + bit number
= (0x002F + 0x000E) * 0x10 + 0x00 = 0x03D0

Data request:

0A	01	03D0	0004	CRC16

Response:

0A	01	01	0F	CRC16

In every case, at least 8 bits (1 byte) are read, irrespective of the number of bits to be read, since the response is made in bytes.

In the example above this means that the bits 0x03D0—0x03D7 are read.

0x03D7	0x03D6	0x03D5	0x03D4	0x03D3	0x03D2	0x03D1	0x03D0

8 bits = 1 byte

For all non-relevant bits (0x03D4—0x03D7) the response is the value 0.

3.2 Read n words

This function reads n words from a defined address.

Data request

Slave address	Function 0x03 or 0x04	Address first word	Number of words	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Response

Slave address	Function 0x03 or 0x04	Number of bytes read	Word value(s)	Checksum CRC16
1 byte	1 byte	1 byte	x byte(s)	2 bytes

Example

Read the first 3 measurement inputs
⇨ Section 6.1.2 “Process data”

Word address= base address + process data address
= 0x002F + 0x0002 = 0x0031

Data request:

14	03	0031	0006	CRC16

Response:

14	03	0C	1999	4348	4CCC	4348	2666	4396	CRC16
			Measurement 1 200.1		Measurement 2 200.3		Measurement 3 300.3		

3.3 Write one bit

In the function “Write bit”, the data blocks for instruction and response are identical.

Instruction

Slave address	Function 0x05	Bit address	Bit value XX 00	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Response

Slave address	Function 0x05	Bit address	Bit value XX 00	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

☞ The following applies for the bit value: FF00 = set bit, 0000 = cancel bit

Example

Set the status bit 0 of the data block “Text for printing”
⇨ Section 6.2.1 “Text for printing”

Bit address = (base address + address “Data structure status”) * 16 + bit number
= (0x007C + 0x0) * 0x10 + 0x0
= 0x07C0

Instruction:

14	05	07C0	FF00	CRC16

Response (as instruction):

14	05	07C0	FF00	CRC16

3.4 Writing one word

In the word writing function, the data blocks for instruction and response are identical.

Instruction

Slave address	Function 0x06	Word address	Word value	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Response

Slave address	Function 0x06	Word address	Word value	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Example

Write flag 1 for display control (= 0x0001)
⇨ Section 6.2.3 “Flags for display control”

Word address= base address + address flag 1

= 0x0126 + 0x0002 = 0x0128

Instruction:

14	06	0128	0001	CRC16

Response (as instruction):

14	06	0128	0001	CRC16

3.5 Write n bits

Instruction

Slave address	Function 0x0F	Address first bit	Bit number	Byte number	Bit value(s)	Check-sum CRC16
1 byte	1 byte	2 bytes	2 bytes	1 byte	x byte(s)	2 bytes

Response

Slave address	Function 0x0F	Address first bit	Number of bits	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Example

Set the status bit 0, bit 1 and bit 2 of the data block “Text for printing”

Status bit 0 = 1, status bit 1 = 0, status bit 2 = 1
⇨ Section 6.2.1 “Text for printing”

Bit address = (base address + address “Data structure status”) * 16 + bit number
= (0x007C + 0x0) * 0x10 + 0x0 = 0x07C0

Instruction:

14	0F	07C0	0003	01	05	CRC16

Response:

14	0F	07C0	0003	CRC16

3.6 Write n words

Instruction

Slave address	Function 0x10	Address first word	Number of words	Number of bytes	Word value(s)	Check-sum CRC16
1 byte	1 byte	2 bytes	2 bytes	1 byte	x byte(s)	2 bytes

Response

Slave address	Function 0x10	Address of first word	Number of words	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Example

Write “Text for printing”
(2 words: “ABC” = 0x4142, 0x4300)
⇨ Section 6.2.1 “Text for printing”

Word address= base address + process data address

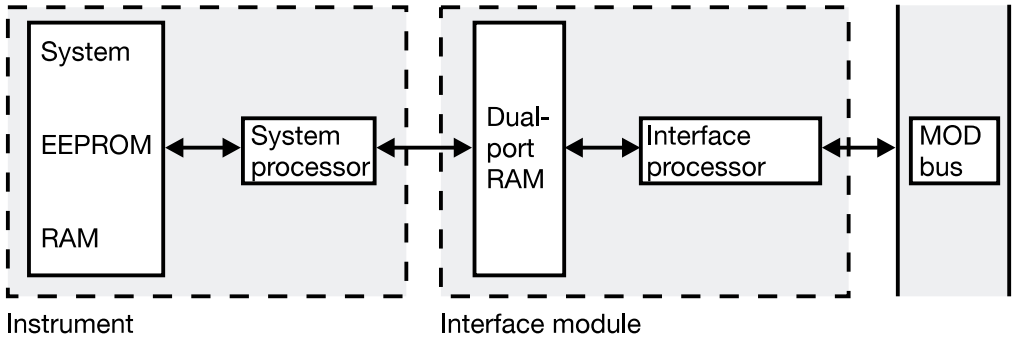
= 0x007C + 0x0002 = 0x007E

Instruction:

14	10	007E	0002	04	4142	4300	CRC16

Response:

14	10	007E	0002	CRC16



For the data transmission to the MODbus, the process values are placed in a dual-port RAM by the system processor. Not all the system variables present in the instrument are updated cyclically in the dual- port RAM. The dual-port RAM is divided into two areas:

System variables

These variables (cyclic data) can be read directly by the master, because they can be updated cyclically (within the sampling time) in the dual-port RAM.

Cyclic data can only be read (R/O - read only).

Data after data request

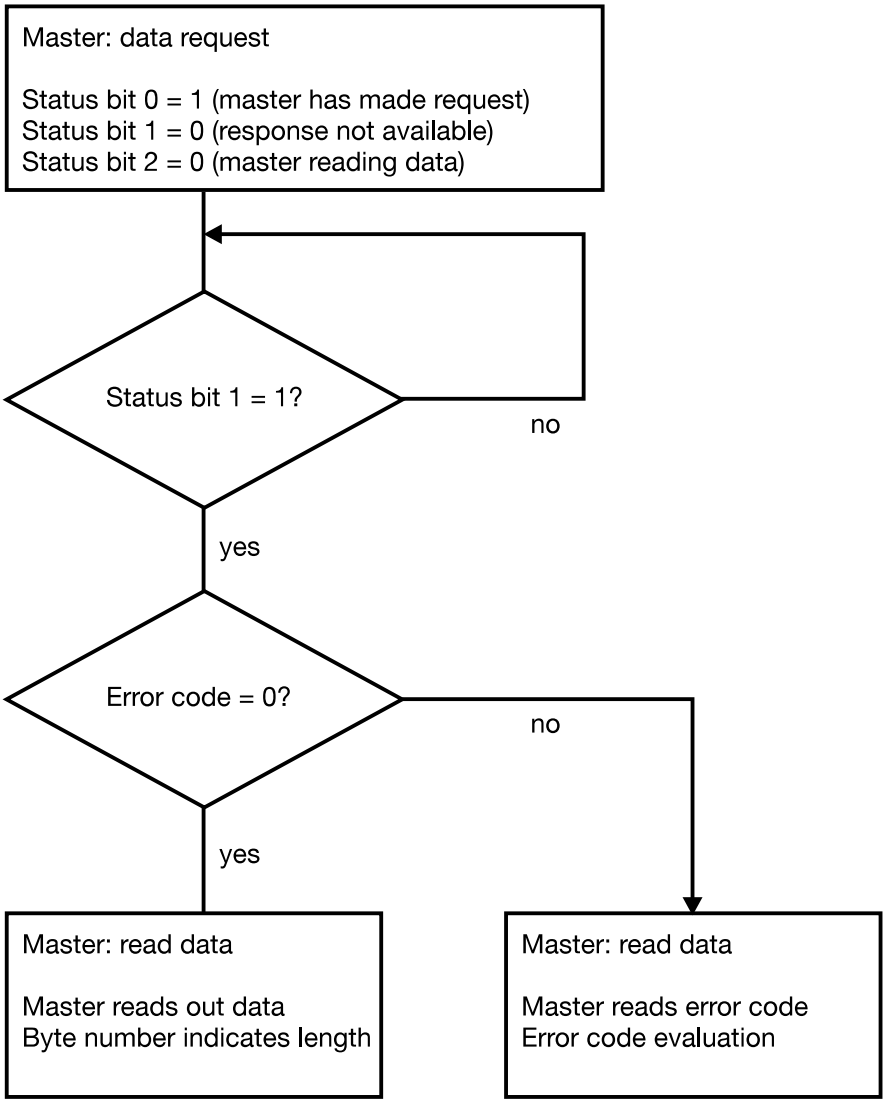
This area is not updated cyclically by the system processor (non-cyclic data).

Variables in this data area must be requested by the MODbus driver. They are available only after processing by the system processor.

Non-cyclic data can be read and written (R/W - read/write). The data request by the master is always made with reference to a complete set of data (table), such as “Flags for display control”.

If variables are altered by the master, the rest of the table remains unchanged. After completion of the data alteration, the master has to initiate the transfer of the table from the dual-port RAM to the system processor.

4.1 Receive non-cyclic data from the instrument



Example

Read the flag for the display text 2
⇨ Section 6.2.3 “Flags for display control”

Step 1: The data structure “Flags for display control” is requested

Set status bit 0 = 1, status bit 1 = 0, and status bit 2 = 0

MODbus command: Write 1 word

01	06	0126	0001	CRC16

Response:

01	06	0126	0001	CRC16

Step 2: Cyclic polling whether the corresponding data structure is available

Read status bit 1

MODbus command: Read n bit

01	01	1261	0001	CRC16

Response:

01	01	01	00	CRC16

Status bit 1 = 0 (data structure is not yet available)

01	01	01	01	CRC16

Status bit 1 = 1 (data structure is available)

Step 3: Read error code of the structure requested

MODbus command: Read 1 word

01	03	0127	0001	CRC16

Response:

01	03	02	0000	CRC16

No error occurred.

Step 4: Poll flag for display text 2

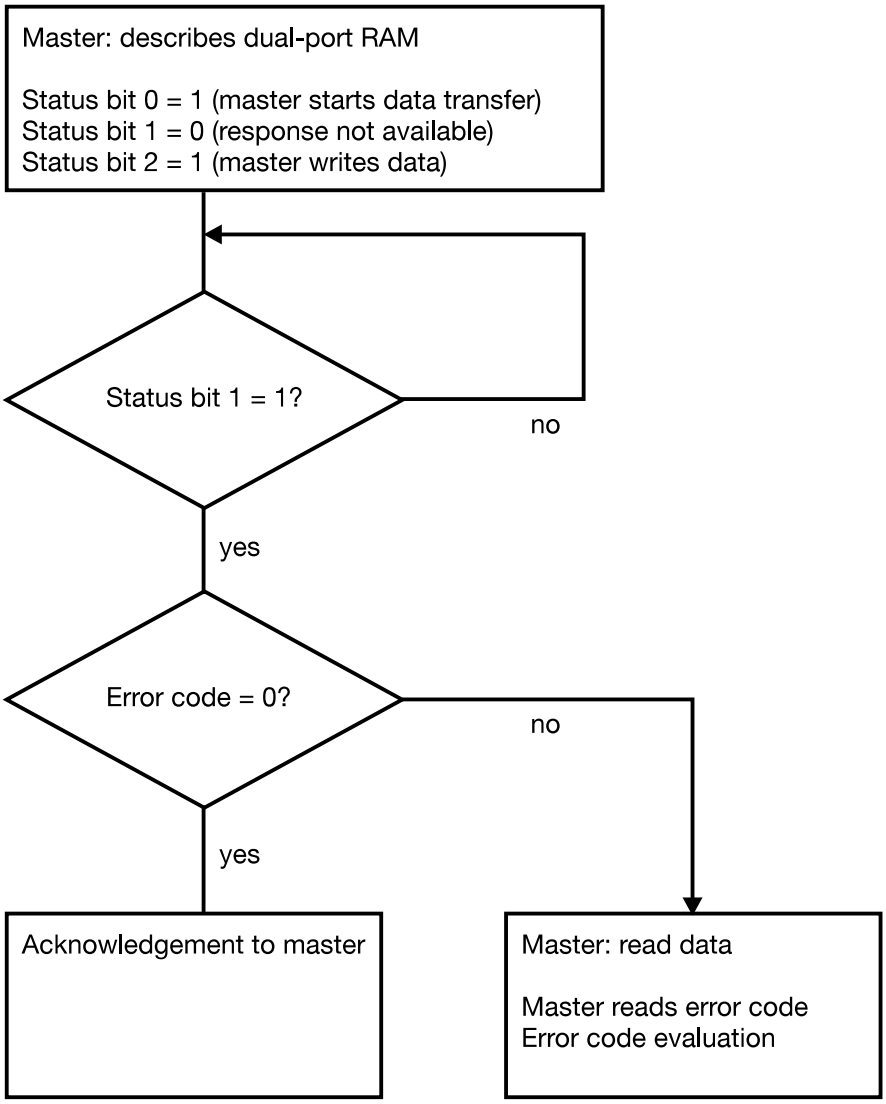
MODbus command: Read 1 word

01	03	0129	0001	CRC16

Response:

01	03	02	0001	CRC16
			display text 2 is shown	

4.2 Transmit non-cyclic data to the instrument



Example

Write one float value for the maths module
(float value 1 = 20.32)

Step 1: The data structure is requested

Set status: bit0=1, bit1 = 0, bit2 = 0

MODbus command: Write 1 word

01	06	0139	0001	CRC16

Response:

01	06	0139	0001	CRC16

Step 2: Cyclic polling whether the corresponding data structure is available

Read status bit1

MODbus instruction: Read n bit

01	01	1391	0001	CRC16

Response:

01	01	01	00	CRC16

Status bit1 = 0 (data structure is not yet available)

01	01	01	01	CRC16

Status bit1 = 1 (data structure is available)

Step 3: Read error code of the structure requested

MODbus command: Read 1 word

01	03	013A	0001	CRC16

Response:

01	03	02	0000	CRC16

No error occurred

Step 4: Write 20.32 to float value 1
(20.32 corresponds to 0x41A28F5C in the IEEE format)

MODbus command: Write 2 words

01	10	013B	0002	04	8F5C	41A2	CRC16

Response:

01	10	013B	0002	CRC16

Step 5: The data structure is transmitted

Set status: bit0 = 1, bit1 = 0, bit2 = 1

MODbus command: Write 1 word

01	06	0139	0005	CRC16

Response:

01	06	0139	0005	CRC16

Step 6: Cyclic polling whether the corresponding data structure has been transmitted

Read status bit1

MODbus command: Read n bit

01	01	1391	0001	CRC16

Response:

01	01	01	00	CRC16

Status bit1 = 0 (data structure not yet transmitted)

01	01	01	01	CRC16

Status bit1 = 1 (data structure has been transmitted)

Step 7: Read error code of transmitted structure

MODbus command: Read 1 word

01	03	013A	0001	CRC16

Response:

01	03	02	0000	CRC16

No error occurred

4.3 Transmission format

Integer values

Integer values are transmitted over the MODbus in the following format: first the high byte, then the low byte.

e. g.: write the integer value 1 (= 0x0001) to the address 0x012A:
0106012A**0001**683E

Float values

Float values are handled on the MODbus with the IEEE-754 standard format (32 bits), but with the difference that bytes 1 and 2 are swapped with bytes 3 and 4.

Single-float format (32 bits) to standard IEEE 754

SEEEEEEE	EMMMMMMM	MMMMMMMM	MMMMMMMM
byte 1	byte 2	byte 3	byte 4

S - sign bit
E - exponent (complement to base 2)
M - 23 bit normalised mantissa

Modbus-float format

MODbus address x		MODbus address x+1	
MMMMMMMM	MMMMMMMM	SEEEEEEE	EMMMMMMM
byte 3	byte 4	byte 1	byte 2

e. g.: write the float value 550.0
(= 0x44098000 in IEEE-754 format) to address 0x013B:
0110013B000204**80004409**665E

The bytes of the float value must be swapped accordingly before/after the transmission to/from the instrument.

Many compilers (e. g. Microsoft C++, Turbo C++, Turbo Pascal, Keil C51) record the float values in the following order:

float value

Address x	Address x+1	Address x+2	Address x+3
MMMMMMMM	MMMMMMMM	EMMMMMMM	SEEEEEEE
byte 4	byte 3	byte 2	byte 1

Texts

Texts are transmitted in the ASCII format.

☞ The last sign must always be a “\0” (ASCII code 0x00) as stop marker.

Since texts are transmitted word by word, an additional 00x00 has to be transmitted when there is an uneven character number (including “\0”).

E. g.: Write text “abcd” (= 61626364) to the address 0x007E (text for printing):
0110007E000306**6162636400002**B7C

If texts are transmitted to the printer for output to paper, it is sufficient to send them to the corresponding address (⇨ Section 6.2.1 “Text for printing”). They will be printed automatically.

The polling flag can be used to check whether the text printing has been completed. Only then should a new text follow.

If texts are transmitted to the printer for display, the following two steps are necessary:

- Transmit text (⇨ Section 6.2.2 “Text for display”)
- Display text by operating a flag
  (⇨ Section 6.2.3 “Flags for display control”)

Altogether 6 texts and 6 control variables (flags) are available.

If the text is to be displayed, then the corresponding flag must be set (transmit 1). When the flag is cancelled (transmit 0), the text disappears again. All that is required to display the text once more, is to set the flag again. The text itself is still available in the printer.

5 Error messages

5.1 Error messages from the interface

The error numbers can be found under the error code in the data blocks of the non-cyclic data.

⇨ Section 6.2.1 - Section 6.2.4

Error code	Error: Setup command processing
0x0014	Command busy flag not reset by master
0x0015	Invalid command
0x0016	Error on data acceptance
0x0017	No cyclic data available
0x0018	Invalid structure length
0x0019	Invalid header
0x001C	Write error in the serial EEPROM (Calib.)

5.2 Error messages for invalid values

For measurements, the convention is that the error number is represented in the value itself, i.e. the error number is recorded instead of the measurement.

Error number	Error
200000.0	Measurement overrange/underrange

5.3 System and run-time errors

The system or run-time errors are part of the process data (cyclic data).

⇨ Section 6.1.2

The numbers which are recorded there have the following meaning:

System errors

If one of the following errors occurs, then the measurement acquisition and recording are interrupted. All other process data are no longer valid. If error number 13 occurs, then the instrument is still in the initialisation phase, during which valid process data are not yet available. The instruction to the instrument should be repeated after a short delay.

Error number	Error
0	no error
1	reserved
2	reserved
3	reserved
4	EEPROM fault
5	reserved
6	reserved
7	reserved
8	reserved
9	reserved
10	print-head operation faulty
11	reserved
12	reserved
13	initialisation phase

Run-time errors

Please see Operating Instructions B 95.4012 for the response of the instrument to one or more of the following errors.

All other data can continue to be read out from the instrument.

Error number	Error
0	no error
1	reserved
2	reserved
3	no paper
4	relay module does not respond
5	clock must be reset
6	battery low
7	reserved
8	A/D converter faulty

☞ It is possible for several run-time errors to occur at the same time. The run-time error which is displayed is always the one with the highest priority.

Event	**Priority**
no paper	higher
relay module does not respond	
clock must be reset	
battery low	
A/D converter faulty	
no error	lower

If, for instance, all the paper has been used up and the battery is low, then error no. 3 (no paper) will be signalled as a run-time error.

6 Address tables

All process values (variables) together with their addresses, the data type and the access mode are described below.

References are as follows:

R/O	access read only
R/W	access read and write
char	ASCII character (8 bits)
byte	byte (8 bits)
int	integer (16 bits)
char xx	character string of length xx; xx = length including string stop character “\0”
bit x	bit No. x
float	float value (4 bytes)

The process values are divided into logical areas.

The absolute MODbus address is given by the base address of the appropriate area and the address offset.

In the address tables below, bit 0 is always the least significant bit.

6 Address tables

6.1 Cyclic data

6.1.1 Instrument data

Base address: 0x0000

Address	Access	Data type	Signal designation
0x0000	R/O	int	Instrument group (13)
0x0001	R/O	int	Instrument type (0)
0x0002	R/O	char 9	Instrument name (“LP 500”)
0x0007	R/O	char 11	Software version
0x000D	R/O	char 13	VdN number
0x0014	R/O	char 10	Production number
0x0019	R/O	char 15	Date/time of last change configuration
0x0021	R/O	char 15	Date/time of last change parameter

6.1.2 Process data

Base address: 0x002F

Address	Access	Data type	Signal designation
0x0000	R/O	int	System error ⇨ Section 5.3 “System and run-time errors”
0x0001	R/O	int	Run-time error ⇨ Section 5.3 “System and run-time errors”
0x0002	R/O	float	Measurement input 1
0x0004	R/O	float	Measurement input 2
0x0006	R/O	float	Measurement input 3
0x0008	R/O	float	Measurement input 4
0x000A	R/O	float	Measurement input 5
0x000C	R/O	float	Measurement input 6
0x000E	R/O	int	State of the logic inputs: 0 = open / 1 = closed
	R/O	bit0	Logic input 1
	R/O	bit1	Logic input 2

Address	Access	Data type	Signal designation
	R/O	bit2	Logic input 3
	R/O	bit3	Logic input 4
	R/O	bit4	Logic input 5
	R/O	bit5	Logic input 6
	R/O	bit6	Logic input 7
	R/O	bit7	Logic input 8
	R/O	bit8—15	free
0x000F	R/O	int	Limit comparator states 1 — 8 (for relays 1 — 8): 0 = inactive / 1 = active
	R/O	bit0	Limit comparator output 1
	R/O	bit1	Limit comparator output 2
	R/O	bit2	Limit comparator output 3
	R/O	bit3	Limit comparator output 4
	R/O	bit4	Limit comparator output 5
	R/O	bit5	Limit comparator output 6
	R/O	bit6	Limit comparator output 7
	R/O	bit7	Limit comparator output 8
	R/O	bit8—15	free
0x0010	R/O	int	Open-collector outputs 1 — 3: logic 0 = not conducting, no current flows logic 1 = conducting, current flows Open-collector output 4: logic 1 = not conducting, no current flows logic 0 = conducting, current flows
	R/O	bit0	Open collector 1
	R/O	bit1	Open collector 2
	R/O	bit2	Open collector 3
	R/O	bit 3	Open collector 4 (fault message)
	R/O	bit4	free
	R/O	bit5	free
	R/O	bit6	free

<table>
<tr><th>Address</th><th>Access</th><th>Data type</th><th>Signal designation</th></tr>
<tr><td></td><td>R/O</td><td>bit7</td><td>free</td></tr>
<tr><td></td><td>R/O</td><td>bit8 — 15</td><td>free</td></tr>
<tr><td>0x0011</td><td>R/O</td><td>float</td><td>Event counter 1</td></tr>
<tr><td>0x0013</td><td>R/O</td><td>float</td><td>Event counter 2</td></tr>
<tr><td>0x0015</td><td>R/O</td><td>int</td><td>Current chart speed
<table>
<tr><th>Value</th><th>Speed [mm/h]</th></tr>
<tr><td>0
1
2
3
4
5
6
7
8
9
10</td><td>0
5
10
20
60
120
240
300
360
600
720</td></tr>
</table></td></tr>
<tr><td>0x0016</td><td>R/O</td><td>int</td><td>Stop status:
0 = no stop / 1 = stop</td></tr>
<tr><td>0x0017</td><td>R/O</td><td>int</td><td>Paper end: 0 = paper available /
1 = no paper</td></tr>
<tr><td>0x0018</td><td>R/O</td><td>char 20</td><td>Current date and time
(“yyyy-mm-dd-hh:mm:ss”)</td></tr>
<tr><td>0x0022</td><td>R/O</td><td>char 20</td><td>Last switch-off of the printer
(date/time)</td></tr>
<tr><td>0x002C</td><td>R/O</td><td>char 20</td><td>Last switch-on of the printer
(date/time)</td></tr>
<tr><td>0x0036</td><td>R/O</td><td>int</td><td>Number of mains switch-off events</td></tr>
<tr><td>0x0037</td><td>R/O</td><td>int</td><td>Number of operating hours</td></tr>
<tr><td>0x0038</td><td>R/O</td><td>int</td><td>Polling flag for the text transmitted for printing:
0 = no print request
1 = print request for the text
is still present</td></tr>
</table>

6.2 Non-cyclic data

Status bits

The tables below (for non-cyclic data) are only read from or written to on data request from the master. The state of these address tables (non-cyclic data) is shown in the status word. Status word and error code are each placed at the start of the data blocks for the non-cyclic data.

Status			Significance for the master
bit 2	bit 1	bit 0	
X	0	0	Master has not made a data request
X	0	1	Master has made a data request to the instrument, the data request is being processed
X	1	0	Processing is finished, the response is available in the buffer, ready for the master
X	1	1	not valid
0			Data transmission from instrument to dual-port RAM
1			Data transmission from dual-port RAM to instrument
bit 3— bit 15 are not used			

6.2.1 Text for printing

Base address: 0x007C

Address	Access	Data type	Signal designation
0x0000	R/W	int	Status of the data structure
	R/W	bit0	1 = process data request
	R/W	bit1	0 = response not yet present / 1 = processing ended, response is present in the buffer
	R/W	bit2	0 = transmission printer ➔ interface 1 = transmission interface ➔ printer
		bit3—15	free
0x0001	R/W	int	Error code ⇨ Section 5.1 “Error messages from the interface”
0x0002	R/W	char36	Text for output to paper

6.2.2 Text for display

Base address: 0x00A4

Address	Access	Data type	Signal designation
0x0000	R/W	int	Status of the data structure
	R/W	bit0	1 = process data request
	R/W	bit1	0 = response not yet present 1 = processing finished, response is present in the buffer
	R/W	bit2	0 = transmission printer ➔ interface 1 = transmission interface ➔ printer
		bit3—15	free
0x0001	R/W	int	Error code ⇨ Section 5.1 “Error messages from the interface”
0x0002	R/W	char 36	Text 1 for display
0x0014	R/W	char 36	Text 2 for display
0x0026	R/W	char 36	Text 3 for display
0x0038	R/W	char 36	Text 4 for display
0x004A	R/W	char 36	Text 5 for display
0x005C	R/W	char 36	Text 6 for display

6.2.3 Flags for display control

Base address: 0x0126

Address	Access	Data type	Signal designation
0x0000	R/W	int	Status of the data structure
	R/W	bit0	1 = processing data request
	R/W	bit1	0 = response not yet present 1 = response ended, response is present in the buffer
	R/W	bit2	0 = transmission printer ➔ interface 1 = transmission interface ➔ printer
		bit3—15	free
0x0001	R/W	int	Error code ⇨ Section 5.1 “Error messages from the interface
0x0002	R/W	int	Flag 1: 0 = do not display text 1 1 = display text 1
0x0003	R/W	int	Flag 2: 0 = do not display text 2 1 = display text 2
0x0004	R/W	int	Flag 3: 0 = do not display text 3 1 = display text 3
0x0005	R/W	int	Flag 4: 0 = do not display text 4 1 = display text 4
0x0006	R/W	int	Flag 5: 0 = do not display text 5 1 = display text 5
0x0007	R/W	int	Flag 6: 0 = do not display text 6 1 = display text 6
0x0008	R/W	int	Flag 7: freely usable in the maths module

6.2.4 Float values for the maths module

Base address: 0x0139

Address	Access	Data type	Signal designation
0x0000	R/W	int	Status of the data structure
	R/W	bit0	1 = processing data request
	R/W	bit1	0 = response not yet present / 1 = processing ended, response present in the buffer
	R/W	bit2	0 = transmission printer ➔ interface 1 = transmission interface ➔ printer
		bit3—15	free
0x0001	R/W	int	Error code ⇨ Section 5.1 “Error messages from the interface”
0x0002	R/W	float	Float value 1 = freely usable in the maths module
0x0004	R/W	float	Float value 2 = freely usable in the maths module
0x0006	R/W	float	Float value 3 = freely usable in the maths module
0x0008	R/W	float	Float value 4 = freely usable in the maths module
0x000A	R/W	float	Float value 5 = freely usable in the maths module
0x000C	R/W	float	Float value 6 = freely usable in the maths module

6.2.5 Reset recognition

Base address: 0x0408

Address	Access	Data type	Signal designation
0x0000	R/W	int	Reset recognition of the interface module

This storage address can be used for a reset recognition of the interface module. A value other than zero must be entered and then cyclically polled. If a reset appears at the interface module of the printer, then the byte will be cancelled (set to zero). The cyclic polling thus enables a reset to be recognised.

A reset of the interface module can have the following causes:

- Mains off
- Configuration of the printer from the keypad
  (if the codeword has been entered correctly)
- Setup connector is plugged into the printer

7 Further examples

As far as the checksum (CRC 16) is concerned, with all the examples listed below it is presumed that the slave connected has the address 1.

7.1 Read process data

Data request: Poll system error

01	03	002F	0001	B5C3

Response: No system error occurred

01	03	02	0000	B844

Data request: Poll run-time error

01	03	0030	0001	8405

Response: No run-time error occured

01	03	02	0000	B844

Data request: Read measurement input 1

01	03	0031	0002	95C4

Response: Current value at measurement input 1 = 200.1

01	03	04	1999	4348	1C46
			200.1		

Data request: Polling flag for text printing

01	03	0067	0001	35D5

Response: Print request is still present

01	03	02	0001	7984

7.2 Send text to the instrument and print

Instruction: Request data structure “Text for printing”
(from the instrument into the dual-port RAM)

01	06	007C	0001	89D2

Response:

01	06	007C	0001	89D2

Data request: Structure available?

01	01	07C1	0001	AD42

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	007D	0001	1412

Response: No

01	03	02	0000	B844
			no	

Instruction: Send text “23455” to instrument

01	10	007E	0003	06	3233	3435	3500	8C26
					23	45	5	

Response:

01	10	007E	0003	E010

Instruction: Transmit data structure from dual-port RAM to instrument

01	06	007C	0005	8811

Response:

01	06	007C	0005	8811

Data request: Transmit structure?

01	01	07C1	0001	AD42

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	007D	0001	1412

Response: No

01	03	02	0000	B844
			no	

Data request: Has text been printed already?

01	03	0067	0001	35D5

Response: Yes

01	03	02	0000	B844
			yes	

7.3 Send text to instrument and display

Instruction: Request structure “Text for display”

01	06	00A4	0001	09E9

Response:

01	06	00A4	0001	09E9

Data request: Structure available?

01	01	0A41	0001	AE06

Response: No

01	01	01	00	5188
			no	

Since the structure is not yet available in the dual-port RAM, the data request has to be repeated.

Data request: Structure available?

01	01	0A41	0001	AE06

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	00A5	0001	9429

Response: No

01	03	02	0000	B844
			no	

Instruction: Send display text 1 “Please wait”

01	10	00A6	0007	0E	4249	5454	4520	5741	5254	454E	0000	940F
					PLEASE WAIT							

Response:

01	10	00A6	0007	61E8

Instruction: Send display text 2 “Stop!”

01	10	00B8	0003	06	4861	6C74	2100	9727
					Stop!			

Response:

01	10	00B8	0003	002D

Instruction: Transmit data structure from dual-port RAM to instrument

01	06	00A4	0005	082A

Response:

01	06	00A4	0005	082A

Data request: Transmit structure?

01	01	0A41	0001	AE06

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	00A5	0001	9429

Response: No

01	03	02	0000	B844
			no	

Instruction: Request structure “Flags for display control”

01	06	0126	0001	A83D

Response:

01	06	0126	0001	A83D

Data request: Data structure available?

01	01	1261	0001	A96C

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	0127	0001	35FD

Response: No

01	03	02	0000	B844
			no	

Instruction: Display text 1 and 2

01	10	0128	0002	04	0001	0001	6D81

Response:

01	10	0128	0002	C03C

Instruction: Transmit data structure from dual-port RAM to instrument

01	06	0126	0005	A9FE

Response:

01	06	0126	0005	A9FE

Data request: Transmit structure?

01	01	1261	0001	A96C

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	0127	0001	35FD

Response: No

01	03	02	0000	B844
			no	

7.4 Send measurements to the instrument

Instruction: Request data structure “Float values for the maths module”

01	06	0139	0001	99FB

Response:

01	06	0139	0001	99FB

Data request: Structure available?

01	01	1391	0001	A8A3

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	013A	0001	A5FB

Response: No

01	03	02	0000	B844
			no	

Instruction: Send value 1 (20.32)

01	10	013B	0002	04	8F5C	41A2	E7B7
					20.32		

Response:

01	10	013B	0002	31F9

Instruction: Send value 2 (25.00)

01	10	013D	0002	04	0000	41C8	0D74
					25.00		

Response:

01	10	013D	0002	D1F8

Instruction: Transmit data structure from dual-port RAM to instrument

01	06	0139	0005	9838

Response:

01	06	0139	0005	9838

Data request: Transmit structure?

01	01	1391	0001	A8A3

Response: Yes

01	01	01	01	9048
			yes	

Data request: Error occurred?

01	03	013A	0001	A5FB

Response: No

01	03	02	0000	B844
			no	

MEASUREMENT AND CONTROL

M. K. JUCHHEIM GmbH & Co

36035 Fulda
Germany
Phone (int. +49) 661 6003-0
Fax (int. +49) 661 6003-607
Telex 49701 juf d
email JUMO_de@e-mail.com