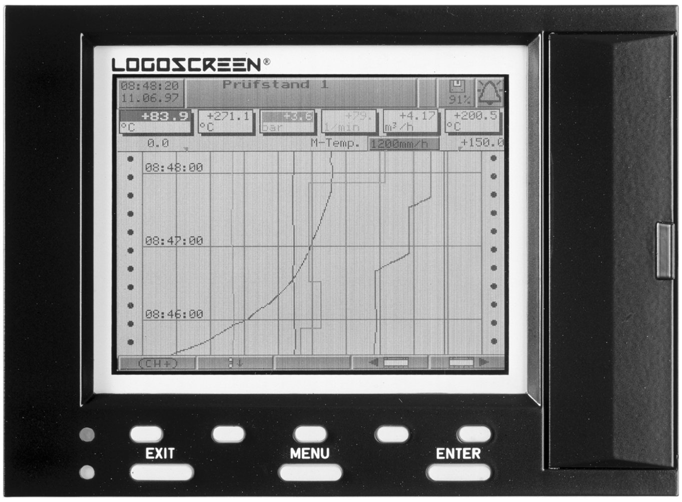


LOGOSCREEN®

Screen recorder

B 95.5010.2

Interface description

8.99/00356849

Contents

Contents

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 assists us to improve these operating instructions where necessary.

Your suggestions will be most welcome.

Phone	in Germany	(0661) 6003-725
	from abroad	(+49) 661 6003-0
Fax	in Germany	(0661) 6003-681
	from abroad	(+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** packaging material for transport.

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

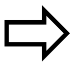
1.2 Typographical conventions

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

2.1 Applications

The RS232 or RS422/RS485 serial interfaces are available for communication with supervisory systems (e.g. bus system or PC). They can be used to read out

- measurements and
- instrument and process data

from the screen recorder.

2.2 System requirements

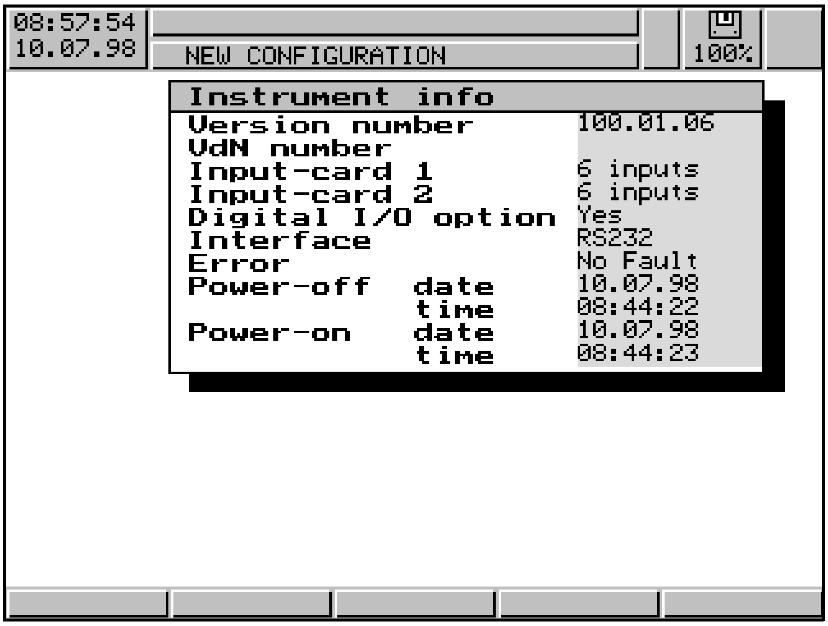
The following items are necessary for operating the serial interface:

- screen recorder with program version[1] from 100.01.06 (including serial interface)
- master (e. g. PC)
- connecting cable
- JUMO-PCVUE (SVS2000) evaluation program or others

2.3 Identifying the interface

The screen recorder is supplied with an integral RS232 interface as standard (RS422/RS485 interface is under development).

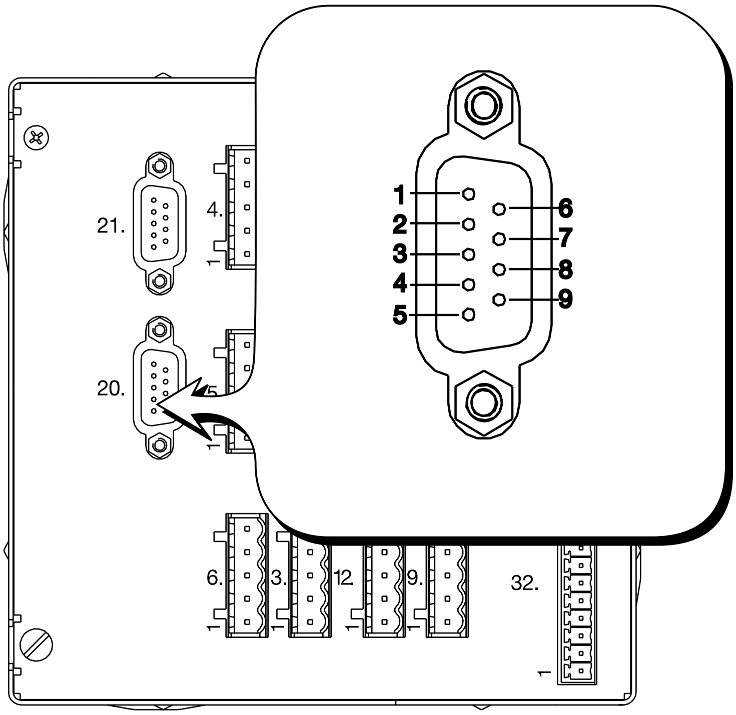
Which interface is implemented in the screen recorder can be requested via the menu *“Instrument info ➔ Interface”.*



1. The program version can be requested in the menus of the screen recorder via *Instrument info ➔ Version number*

3.1 Connection diagram

Rear view of the screen recorder



Connector 20.

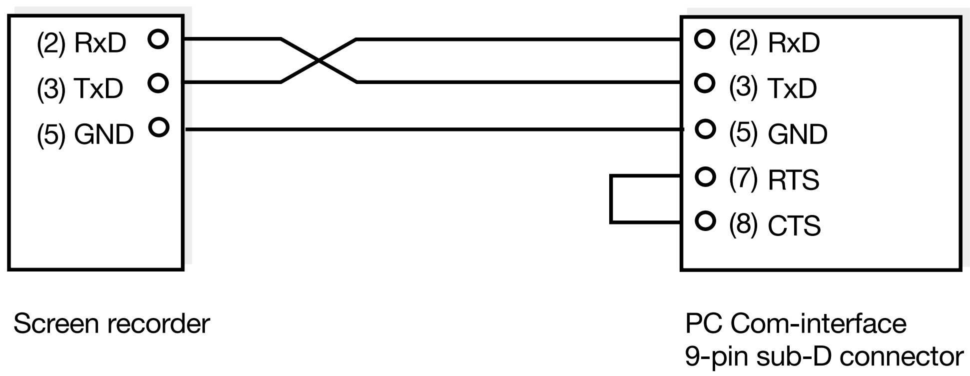
Interface

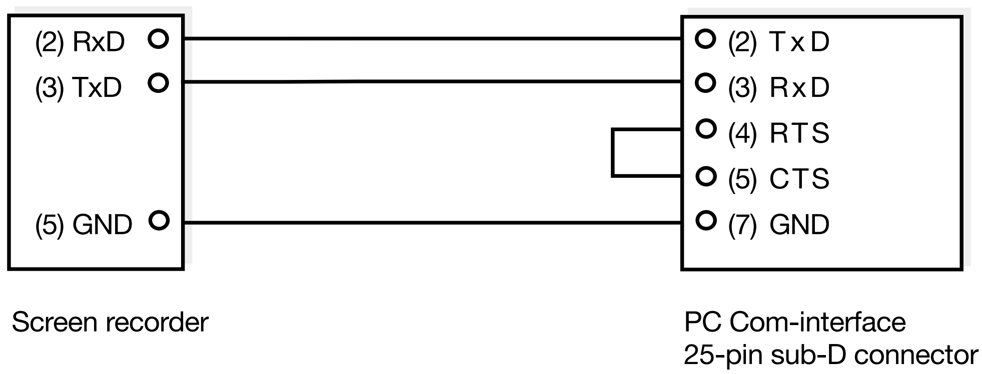
Connection diagram

RS232		**RS422**		**RS485**	
1 ○		1 ○		1 ○	
2 ○	RxD	2 ○		2 ○	
3 ○	TxD	3 ○	TxD+	3 ○	TxD+/RxD+
4 ○		4 ○	RxD+	4 ○	
5 ○	GND	5 ○	GND	5 ○	GND
6 ○		6 ○		6 ○	
7 ○		7 ○		7 ○	
8 ○		8 ○	TxD-	8 ○	TxD-/RxD-
9 ○		9 ○	RxD-	9 ○	

3.2 RS232

The handshake connections (RTS, CTS) are not used in the RS232 interface. The RTS connection coming from the master (CTS on screen recorder) is ignored, the answer is sent directly from the screen recorder. The CTS connection of the master (RTS on the screen recorder) remains open. If the master evaluates the handshake connections, they have to be bridged in the cable.



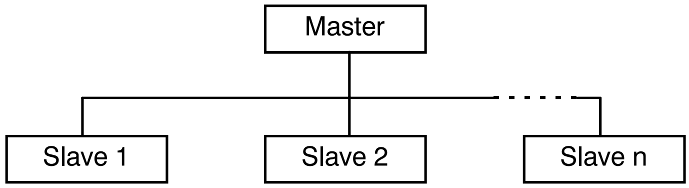


3.3 RS422/RS485

The RS422 and RS485 interfaces are changed over automatically by the screen recorder, according to the type of connection (2-wire or 4-wire connection).

4.1 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 a 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.

4.2 Transmission mode (RTU)

The transmission mode used is the RTU (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	Bit number
8 bits	—	1	9
8 bits	—	2	10
8 bits	even	1	10
8 bits	odd	1	10

4.3 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 form (hexadecimal) in the transfer protocol.

4.4 Timing of the communication

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

The character transmission time (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 other data formats it is:

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

Timing

Data request from master transmission time = n characters * 1000 * x bits / (baud rate)
Marker for end of data request 3.5 characters * 1000 * x bits / (baud rate)
Processing of the data request by the slave (50 msec max.)
Response of the slave transmission time = n characters * 1000 * x bits / (baud rate)
Marker for end of response 3.5 characters * 1000 * x bits / (baud rate)

Example

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

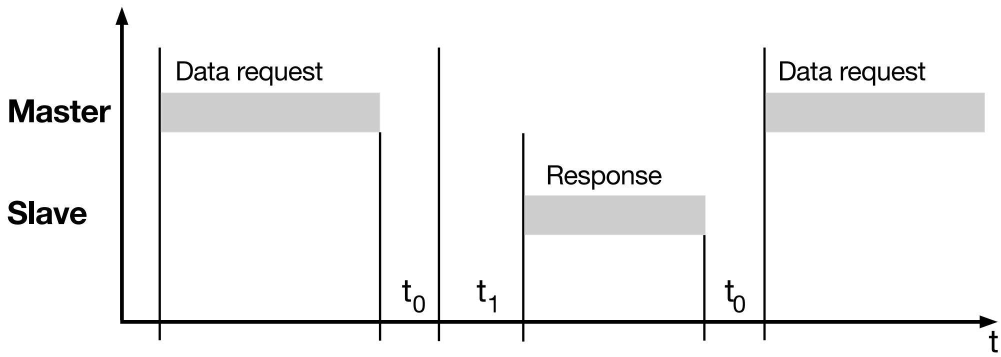
Waiting time = 3.5 characters * 1000 * x bits / (baud rate)

Baud rate [baud]	**Data format** [bit]	**Waiting time** [msec]
38400	10	0.911
	9	0.820
19200	10	1.823
	9	1.641
9600	10	3.646
	9	3.281

4.4.1 Timing sequence of a data request

Timing scheme

A data request runs according to the following timing scheme:



t_0 At least 3.5 times the transmission time for 1 character (the time depends on the baud rate)

t_1 This time depends on the internal processing. The maximum processing time is 50 msec.

☞ A minimum response time can be set in the instrument, under the menu item *Configuration* ➔ *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 for the RS422 or RS232 interface.

4.4.2 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.

4.4.3 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.

4.5 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)

Check sum recognition of transmission errors

4.6 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
...	...	...

4.7 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                          else
        CRC = CRC XOR 0xA001
while (not all ByteOfMessage processed);
```

The low byte of the checksum is transmitted first.

Example 1

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

Data request: Read two words, starting at address 0x57 (CRC16 = 0x1E77)

14	03	0057	0002	771E

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

Poll the status of the relay outputs.

Instruction: Read one word of address 0x31 (CRC16 = 0x00D7)

14	03	0031	0001	D700

Response (CRC = 0x4774):

14	03	02	0001	7447
			Word 1	

Word 1 means that only output 1 is active.

4.8 Configuration of the interface

The interface is configured using the keys of the screen recorder or with the aid of the setup program.

Configuration from the keys

Firstly, the menu *Configuration* must be called up and the parameter *Interface* has to be selected. The parameters for configuring the interface are now available.

	Parameter	Value/selection	Description
Protocol	Configuration → Interface → Protocol	**MODBUS**, JBUS	Select protocol ⇨ Section 4.6 “Distinction Modbus/ Jbus”
Baud rate	Configuration → Interface → Baud rate	**9600 baud**, 19200 baud, 38400 baud	Select baud rate
Data format	Configuration → Interface → Data format	8-1- none, 8-1- odd, **8-1- even**, 8-2- none	Select data format (Data bit - stop bit - parity)
Instrument address	Configuration → Interface → Instrument address	**1** — 255	Select address
Min. response time	Configuration → Interface → Min. response time	**0** — 500msec	Select minimum response time

The instrument address must also be observed when communicating via the RS232 interface, even though it is not a bus interface.

Configuration using the setup program

The menu item *Edit* → *Interface* (*RS232-RS422/485)* is available for configuration with the aid of the setup software.

The measurements and other instrument and process data can be read out from the screen recorder using the functions below.

Overview of functions

Function number	Function	
0x01/0x02	read n bits	(max. 256 bits)
0x03/0x04	read n words	(max. 127 words)

There are no separated 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 for the measurement of analogue input 6:

word address = 0x0035 + 0x000A = 0x003F

Example: bit address of the open-collector output:

bit address = (0x002F + 0x0002) · 0x0010 + 0x0005 = 0x0315

5 Functions

5.1 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 8.2 “Process data”

Bit address = (base address + process data address) * 16 + bit number
= (0x002F + 0x0000) * 0x10 + 0x08 = 0x02F8

Data request: (CRC16 = 0xFBBC)

0A	01	02F8	0004	BCFB

Response: (CRC16 = A813)

0A	01	01	0F	13A8

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

In the example above this means that the bits 0x02F8—0x02FF are read.

0x02FF	0x02FE	0x02FD	0x02FC	0x02FB	0x02FA	0x02F9	0x02F8

8 bits = 1 byte

For all irrelevant bits (0x02FC—0x02FF) the response is the value 0.

5.2 Read n words

This function reads n words from a defined address.

Data request

Slave address	Function 0x03 or 0x04	Address first word	Word number	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 8.2 “Process data”

Word address= base address + process data address
= 0x0035 + 0x0000 = 0x0035

Data request: (CRC16 = 037D)

14	03	0035	0006	D703

Response: (CRC16 = 4750)

14	03	0C	1999	4348	4CCC	4348	2666	4396	5047
			Measurement 1 200.1		Measurement 2 200.3		Measurement 3 300.3		

6.1 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.: Polling the integer value of address 0x0000 if the value 12 (0x000C) is written below this address.
Request: 010300000001840A (CRC16 = 0x0A84)
Response: 010302**000C**B841 (CRC16 = 41B8)

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.: Polling the float value of address 0x0035 if the value 550.0 (0x44098000 in the IEEE-754 format) is written below this address.
Request: 010300350002D405 (CRC16 = 05D4)
Response: 010304**80004409**20F5 (CRC16 = F520)

After the transmission from the instrument, the bytes of the float value have to be swapped accordingly.

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

Please find out how float values are saved in your application. If required, the bytes have to be swapped accordingly in the interface program after acquiring them from the screen recorder.

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 also transmitted word by word (16 bit), an additional 0x00 is transmitted when there is an uneven number of characters (including “\0”).

e. g.: Polling text of address 0x0002 if the character string “L-SCREEN” (ASCII code: 0x4C, 0x2D, 0x53, 0x43, 0x52, 0x45, 0x45, 0x4E, 0x00) is below this address.
Request: 0103000200052409 (CRC = 0924)
Response: 01030A**4C2D53435245454E0000**A587 (CRC16 = 87A5)

7.1 Error handling

Error codes

If the data request by the master was received by the screen recorder (without transmission error), but could not be processed, the screen recorder responds with an error code. There are two error codes:

1 invalid function

2 invalid parameter address

If a bit or word number which is read by the master is larger than the maximum permitted one, the screen recorder also sends the error code 1.

Response on error

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

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

Example

Data request: (CRC16 = 0B1C)

01	09	0000	0001	1C0B

Response: (CRC16 = 5086)

01	89	01	8650

Special cases

The slave does not respond to the following errors:

- the baud rate and /or data format for master and screen recorder do not match
- the instrument address of the screen recorder does not correspond to the one in the protocol (in this case, the data request by the master has to be sent again after a time-out of 2 sec has elapsed).
- 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
- there is ongoing communication via the setup interface at the front

7.2 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 entered instead of the measurement.

Error number	Error
200000.0	Measurement underrange
200001.0	Measurement overrange
200003.0	other invalid value

Example

Data request: (CRC16 = 05D4)

01	03	0035	0002	D405

Response: (CRC16 = C29C)

01	03	04	5000	4843	9CC2

The measurement (0x48435000 = 200000.0) supplied by analogue input 1 indicates an underrange condition.

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.

8.1 Instrument data

Base address: 0x0000

Address	Access	Data type	Signal designation
0x0000	R/O	int	Instrument group (12)
0x0001	R/O	int	Instrument type (0)
0x0002	R/O	char 9	Instrument name ("L-SCREEN")
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

8.2 Process data

Base address: 0x002F

Address	Access	Data type	Signal designation
0x0000	R/O	int	Group alarms and position of the logic inputs
	R/O	bit0	Alarm group 1 0 = no alarm 1 = at least 1 limit infringed in group
	R/O	bit1	Alarm group 2
	R/O	bit2	Alarm group 3
	R/O	bit3	Alarm group 4
	R/O	bit4	Alarm group 5
	R/O	bit5	Alarm group 6
	R/O	bit6-7	free
	R/O	bit8	Logic input 1 0 = open / 1 = closed
	R/O	bit9	Logic input 2
	R/O	bit10	Logic input 3
	R/O	bit11	Logic input 4
	R/O	bit12	Logic input 5
	R/O	bit13	Logic input 6
	R/O	bit14	Logic input 7
	R/O	bit15	free
0x0001	R/O	int	Logic signals
	R/O	bit0-7	free
	R/O	bit8	Combination alarm 0 = no alarm 1 = at least 1 limit infringed in instrument
	R/O	bit9	Disk reserve signal 0 = disk reserve not yet reached 1 = change diskette
	R/O	bit10	Error 0 = no error / 1 = error
	R/O	bit11-15	free

Address	Access	Data type	Signal designation
0x0002	R/O	int	Logic outputs
	R/O	bit0	Relay output 1 0 = not active / 1 = active
	R/O	bit1	Relay output 2
	R/O	bit2	Relay output 3
	R/O	bit3	Relay output 4
	R/O	bit4	Relay output 5
	R/O	bit5	Open-collector output 0 = not active / 1 = active
	R/O	bit6-15	free

Base address: 0x0035

Address	Access	Data type	Signal designation
0x0000	R/O	float	Measurement input 1 (analogue input 1)
0x0002	R/O	float	Measurement input 2 (analogue input 2)
0x0004	R/O	float	Measurement input 3 (analogue input 3)
0x0006	R/O	float	Measurement input 4 (analogue input 4)
0x0008	R/O	float	Measurement input 5 (analogue input 5)
0x000A	R/O	float	Measurement input 6 (analogue input 6)
0x000C	R/O	float	Measurement input 7 (analogue input 7)
0x000E	R/O	float	Measurement input 8 (analogue input 8)
0x0010	R/O	float	Measurement input 9 (analogue input 9)
0x0012	R/O	float	Measurement input 10 (analogue input 10)
0x0014	R/O	float	Measurement input 11 (analogue input 11)
0x0016	R/O	float	Measurement input 12 (analogue input 12)
0x0018	R/O	float	free
0x001A	R/O	float	free
0x001C	R/O	float	free
0x001E	R/O	float	free
0x0020	R/O	float	Counter value 1
0x0022	R/O	float	Counter value 2

A

C

D

E

F

I

J

M

P

R

S

T

M.K. JUCHHEIM GmbH & Co.

36035 Fulda
Germany
Phone ++49 6 61-60 03-0
Fax ++49 6 61-60 03-6 07
Telex 49 701 juf d
email JUMO_de@e-mail.com

United Kingdom

JUMO Instrument Co. Ltd.
Temple Bank, Riverway
GB-Harlow,
Essex CM20 2TT
Phone (0 12 79) 63 55 33
Fax (0 12 79) 63 52 62

USA

JUMO PROCESS CONTROL INC.
735 Fox Chase,
Coatesville, PA 19320
Phone 610-380-8002,
800-554 JUMO
Fax 610-380-8009