Description of Modbus interfaces

Modbus protocol

M0c, M3c

April 2020 (EN)
V1

Table of contents

1 Information about this operating manual 3

1.1 Symbols and signs..3
1.2 Related documents 4

2 Interfaces 5

2.1 Interface allocation..5
2.2 Continuous current when supply voltage is applied.....................5
2.3 Settings for the serial interface 6

3 Description of the Modbus protocol 7

3.1 Function codes 7
3.2 Data types 7
3.3 Floating point values 8
3.4 Long integer values (32 bits) 9
3.5 Checksum (CRC16)..9
3.6 Modbus error codes 10

4 Modbus address tables 11

5 Calibration..14

5.1 Principle of function of the history...14
5.2 Calibration of the sensor slope 15
5.3 Recovery of the nominal slope 16

1 Information about this operating manual

1.1 Symbols and signs

1.1.1 Safety instructions and warnings

This operating manual uses the warning signs and signal words listed below. They help you in the safe handling of the product, protect the operating staff from injuries and the owner against material damage and additional costs.

	Signal word	Meaning
⚠	**DANGER!**	DANGER designates a hazard with a high risk which, if not avoided, will result in death or serious injury.
⚠	**WARNING!**	WARNING designates a hazard with a medium risk which, if not avoided, could result in death or serious injury.
⚠	**CAUTION!**	CAUTION designates a hazard with a low risk level which, if not avoided, could result in minor or moderate injury.
	NOTE	NOTE gives a warning regarding potential material damage.

Tab. 1: Signal words

1.1.2 Signs used in the text

Symbol	Meaning
⚠	This symbol is the general warning sign and warns you about the risk of injury. Observe all instructions marked with this warning sign.
ℹ	This symbol designates tips and helpful information regarding the optimum and economical operation of the product.
▶	This symbol designates an activity to be carried out by the staff.
↳	This symbol designates the result of an activity.
•	This symbol designates bullet points in a list.
✓	This symbol designates a prerequisite for the performance of an activity.
→	This symbol refers to further information contained in other sections, chapters or other operating manuals.

Tab. 2: Signs used in the text

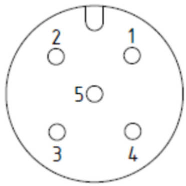
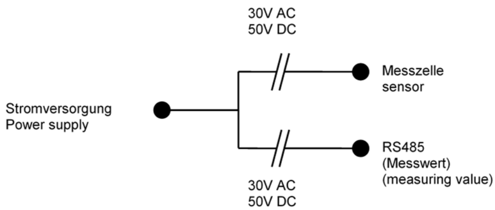
1.2 Related documents

Datasheets for the individual sensor types can be found at the following internet address:
https://www.reiss-gmbh.com/english/datasheets.htm

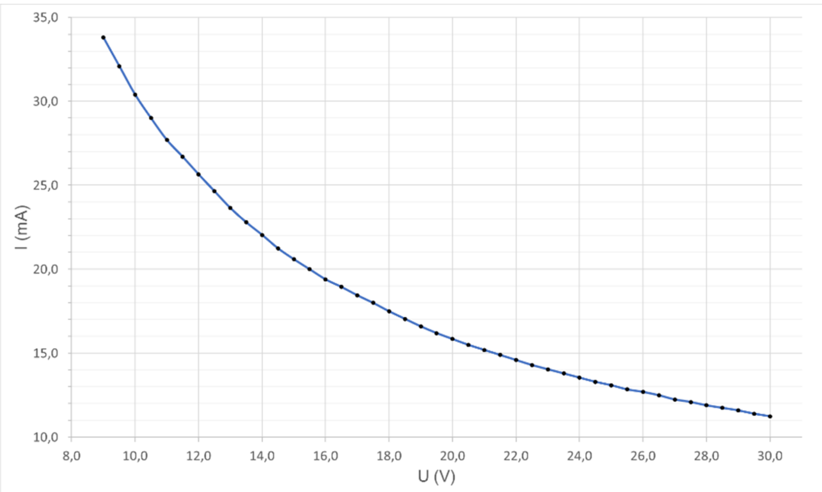
The associated operating manuals can be found at the following internet address:
https://www.reiss-gmbh.com/english/manuals.htm

2 Interfaces

2.1 Interface allocation

5-pole M12 plug-in connector, A-coded • 1 reserved • 2 +9 – +30 V • 3 GND • 4 RS485 B [1] • 5 RS485 A [1]	
Galvanic isolation	
Power supply	9-30 V, max. 56 mA

2.2 Continuous current when supply voltage is applied



[1] There are no terminating resistors in the sensor!

2.3 Settings for the serial interface

Sensor types M0c and M3c support the **RTU mode** only.

2.3.1 Settings for M0c

Configuration item	Settings	Description
Baud rate	**19200**	Transmission rate (symbol rate) of the serial interface
Data format	**8 - 1 - even**	Format of the data word Data bit – stop bit – parity
Device address	**10**	Unique identifier of a bus device 0 = Broadcast address[2] 1 to 247 = Unicast addresses[3]

2.3.2 Settings for M3c

Configuration item	Settings	Description
Baud rate	**9600**	Transmission rate (symbol rate) of the serial interface
Data format	**8 - 2 - none**	Format of the data word Date bit – stop bit – parity
Device address	**20**	Unique identifier of a bus device 0 = Broadcast address[2] 1 to 247 = Unicast addresses[3]

[2] In the Modbus standard, the device address is defined. The broadcast address must not be used as a slave address. It is provided for broadcast messages.
[3] Unicast addresses are provided for use as slave addresses. They serve as the unique identifier of slave devices so that these can be explicitly addressed by the master.

3 Description of the Modbus protocol

3.1 Function codes

Code Hex	Function
0x03	Read Holding Registers [16bit]
0x04	Read Input Registers [16bit]
0x06	Write Single Register
0x10	Write Multiple Registers

3.2 Data types

Data type	Description	Access	Possible function codes Hex	Number of Modbus registers
float (32 bits)	Transmission does not take place in the order defined according to the IEEE 754 standard coding. The order is in the mid-little-endian (CDAB) format instead. → Chapter 3.3, p. 8	R / O R / W	0x03, 0x04 0x03, 0x04, 0x10	2
int	Word (16 bits) as an unsigned integer. Transmission takes places in the big-endian format. Value range: 0 to 65,535	R / O R / W	0x03, 0x04 0x03, 0x04, 0x06	1
unsigned longint	Double word (32 bits) as an unsigned integer. Transmission takes place in the big-endian format. Value range: 0 to 4,294,967,295	R / O R / W	0x03, 0x04 0x03, 0x04, 0x10	2
char [...]	Characters/bytes (8 bits) as an unsigned integer. 2 characters are contained in 1 word. The 1st character is transmitted in the most significant bit (MSB) and the 2nd character in the least significant bit (LSB). Value range: 0 to 255	R / O	0x03, 0x04	1

3.3 Floating point values

When processing floating point values, the sensors use an order **deviating** from the IEEE 754 standard format (32 bits).

S = Sign bit
E = Exponent (two's complement)
M = 23 bits normalised mantissa

Modbus register x+1																**Modbus register x**															
31							**24**	**23**							**16**	**15**							**8**	**7**							**0**
S	**E**	**E**	**E**	**E**	**E**	**E**	**E**	**E**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**
C								**D**								**A**								**B**							

Example
In this example, the concentration value of the disinfectant is to be read out at address 0x0000 of the device. Here, the value is to be 0.168 ppm (0x08313E2C in the mid-little endian (CDAB format)).

Data request:

01	03	00 00	00 02	C4 0B
Slave	Function	Address 1st word	Number of words	CRC

Response (values in the Modbus floating point format):

01	03	04	08 31 3E 2C	B8 21
Slave	Function	Bytes read	Floating point value	CRC

Many compilers (e.g. Microsoft® Visual C++®) save the floating point values in the following order:

Address x								**Address x+1**								**Address x+2**								**Address x+3**							
M	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**E**	**M**	**M**	**M**	**M**	**M**	**M**	**M**	**S**	**E**	**E**	**E**	**E**	**E**	**E**	**E**
D								**C**								**B**								**A**							

The order of the bytes depends on how the floating point values are saved in the respective application. The bytes may have to be interchanged in the interface program.

For PLC applications (Function Block Diagramm):

$Low{-}/Highword\ WORD;\ V\ DINT;\ M\ REAL;\ E\ DINT;\ float\ REAL$

$float = (-1)^{V} \cdot (1 + M) \cdot 2^{E-127}$

Separate the S, E, M of the highword by application of e.g. the bit shift.

3.4 Long integer values (32 bits)

Example

In this example, the time stamp of the most recent calibration of a sensor saved in the calibration memory (calibration 0) is to be read out at address 0x0214 (→ Chapter 4, p11 under “History”). The value in this example is to correspond to the date 08.03.2019, 13:10 h, i.e. to be a decimal value of 1903081310 and thus a hexadecimal value of 716EB75E.

Data request:

01	03	02 14	00 02	85 B7
Slave	Function	Address 1st word	Number of words	CRC

Response (value in the Modbus int format):

01	03	04	71 6E B7 5E	76 DA
Slave	Function	Bytes read	Long integer value	CRC

Limits for the date

- Largest representable number: 2^{32} = 42|94|96|72|96
  corresponding (theoretically) to year: (20)42, month: 94, day: 96, hour: 72, minute: 96
- Last possible date: 31.12.2042, 23:59 h (42|12|31|23|59)

3.5 Checksum (CRC16)

Calculation scheme

Transmission errors are detected by way of the checksum (CRC16). If an error is detected during the evaluation, the device concerned does not respond.

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

The low byte of the checksum is transmitted first!
Example: The CRC16 checksum “DB 25” is transmitted and represented in the 25 DB order.

3.6 Modbus error codes

Prerequisites for Modbus communication

The following prerequisites must be fulfilled for a slave to be able to receive, process and respond to requests:

- The baud rate and data format of master and slave must be the same.
- The correct slave address must be used in the request.
- Slave devices will only respond after a successful checksum check of the request. Otherwise the slave will reject the request.
- The master's instruction must be complete and in conformity with the Modbus protocol.
- The number of words to be read must be greater than 0.

Error codes

If the slave received the master's data request without any transmission errors but could not process it, the slave responds with an error code. The following error codes can occur:

01	Invalid function. The function codes supported by our sensors are specified in Chapter 3.1, p. 7.
02	Invalid address, or an excessive number of words and/or bits are to be read or written.
03	The format of the data cannot be read.
255	There is a communication problem.

Modbus error responses can be recognised by the MSB of the function code having been set to 1.

Response in case of an error

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

The function code is OR'd with 0x80. This sets the most significant bit (MSB) to 1.

Example

Data request:

01	06	23 45	00 01	52 5B
Slave	Write word	Word address	Word value	CRC

Response (with error code 02):

01	86	02	C3 A1
Slave	Function OR	Error	CRC

Response with error code 02, because address 0x2345 is not available.

4 Modbus address tables

Definitions	• **X_Zero in nA**	Current, if no disinfectant is present in the medium to be measured. The start address for this value is 0x0206.
	• **X_Span in nA/unit**	Current value of the latest calibration in relation to the concentration of the disinfectant in the latest calibration. The start address for this value is 0x0208. The unit is provided by reading out address 0x0200.
	• **Concentration**	The software calculates the concentration as follows: $\frac{\text{actual current} - \text{X\_Zero}}{\text{X\_Span}}$

All register information is stated in hexadecimal format.

Device data

Register	Access	Data type	Parameters	Example
0x0300	R / O	char [16]	Sensor type	CP4.0N-M0
0x0308	R / O	int	Hardware	1130 (1.130)
0x0309	R / O	int	Firmware	1503 (1.503)
0x030a	R / O	float	Nominal slope L	7.5
0x030b	R / O		Nominal slope H	
0x030c	R / O	char [20]	Serial number	S19010593
0x0317	R / O	char [10]	Part number	10501004.2

<table>
<tr><th colspan="7">Communication parameters</th></tr>
<tr><th>Register</th><th>Access</th><th>Data type</th><th>Parameters</th><th colspan="2">Value range</th><th>Default</th></tr>
<tr><td>0x0400</td><td>R / W</td><td>int</td><td>Slave address</td><td colspan="2">1 … 247</td><td rowspan="12">→ 2.3.1 and 2.3.2, p. 6</td></tr>
<tr><td rowspan="7">0x0401</td><td rowspan="7">R / W</td><td rowspan="7">int</td><td rowspan="7">Baud rate</td><td>0</td><td>2400</td></tr>
<tr><td>1</td><td>4800</td></tr>
<tr><td>2</td><td>9600</td></tr>
<tr><td>3</td><td>19200</td></tr>
<tr><td>4</td><td>38400</td></tr>
<tr><td>5</td><td>57600</td></tr>
<tr><td>6</td><td>115200</td></tr>
<tr><td rowspan="4">0x0402</td><td rowspan="4">R / W</td><td rowspan="4">int</td><td rowspan="4">Parity/Stop bit</td><td>0</td><td>none/2</td></tr>
<tr><td>1</td><td>even/1</td></tr>
<tr><td>2</td><td>odd/1</td></tr>
<tr><td>3</td><td>none/1</td></tr>
</table>

Process data parameters

Register	Access	Data type	Parameters	Value range		Default
0x200	R / O	int	Unit	0	%	
				1	‰	
				2	g/l	
				3	ppm	
				4	mg/l	
				5	ppb	
				6	µg/l	
0x201	R / O	int	Decimal places	0	0000	
				1	000.0	
				2	00.00	
				3	0.000	
0x0206	R / W	float	X_Zero L			0 [4]
0x0207			X_Zero H			
0x0208	R / W	float	X_Span L			→ 0x030a[4]
0x0209			X_Span H			
0x020a	R / W	unsigned long int	DateTime L	yymmddhhmm		0 [4]
0x020b			DateTime H			

History

Register	Access	Data type	Parameters	Value range	Default
0x0210	R / O	float	X_Zero [0] L		
0x0211			X_Zero [0] H		
0x0212	R / O	float	X_Span [0] L		
0x0213			X_Span [0] H		
0x0214	R / O	unsigned long int	DateTime [0] L	yymmddhhmm	
0x0215			DateTime [0] H		
0x0216	R / O	float	X_Zero [1] L		
0x0217			X_Zero [1] H		
0x0218	R / O	float	X_Span [1] L		
0x0219			X_Span [1] H		
0x021a	R / O	unsigned long int	DateTime [1] L	yymmddhhmm	
0x021b			DateTime [1] H		
0x021c	R / O	float	X_Zero [2] L		
0x021d			X_Zero [2] H		
0x021e	R / O	float	X_Span [2] L		
0x021f			X_Span [2] H		

[4] Initial values

History					
Register	**Access**	**Data type**	**Parameters**	**Value range**	**Default**
0x0220	R / O	unsigned long int	DateTime [2] L	yymmddhhmm	
0x0221			DateTime [2] H		
0x0222	R / O	float	X_Zero [3] L		
0x0223			X_Zero [3] H		
0x0224	R / O	float	X_Span [3] L		
0x0225			X_Span [3] H		
0x0226	R / O	unsigned long int	DateTime [3] L	yymmddhhmm	
0x0227			DateTime [3] H		
0x0228	R / O	float	X_Zero [4] L		
0x0229			X_Zero [4] H		
0x022a	R / O	float	X_Span [4] L		
0x022b			X_Span [4] H		
0x022c	R / O	unsigned long int	DateTime [4] L	yymmddhhmm	
0x022d			DateTime [4] H		
0x022e	R / O	float	Measuring range L		
0x022f			Measuring range H		

Process data of measuring values				
Register	**Access**	**Data type**	**Parameters**	**Value range**
0x0000	R / O	float	Concentration L (unit)	→ Calculation p. 10 → (Unit) 0x200
0x0001			Concentration H (unit)	
0x0002	R / O	float	Cell current L [nA]	
0x0003			Cell current H [nA]	
0x0004	R / O	float	Temperature L	
0x0005			Temperature H	

Note

Write operations on some R/W parameters result in storage in the EEPROM or flash memory. These memory modules only have storage capacity for a limited number of write cycles (about 100,000 or 10,000 respectively).

This is why frequent writing of corresponding variables may result in a memory error.

▶ The number of write operations should therefore be kept as small as possible.

5 Calibration

5.1 Principle of function of the history

As can be seen from the Modbus address tables of the sensor (→ chapter 4, p. 11 under "History"), the current calibration data are stored at addresses 0x0206 to 0x020A.

The storage system of the history functions according to the following rules:

- A completed calibration becomes active when the date and time (time stamp) are written.
- If the current calibration has a greater (more recent) time stamp than the previous calibration, it is stored at the addresses for the current calibration (0x0206 to 0x020A).
  - The data of the current calibration are also stored at addresses 0x0210 to 0x0214 (X_Zero (0), X_Span (0) and Date/Time (0)), i.e. in location 0 of the history.
  - The data of the old calibration are shifted from the previous location 0 of the history to location 1; all further data are also shifted by one location.
  - The oldest data so far (location 4) cannot be shifted anymore; they are deleted.
- If the current calibration has a time stamp that is smaller than or equal to the time stamp of the last calibration, the data of the last calibration (at addresses 0x0206 to 0x020A) and the data in location 0 of the history (at addresses 0x0210 to 0x0214) are overwritten. This does not affect the data in locations 1 to 4 in the history in this case!

As the minute is the smallest unit of the time stamp, the time between 2 calibrations should be at least 1 minute.

5.2 Calibration of the sensor slope

5.2.1 Verification of the sensor slope at the measuring point

✓ The prerequisites for a calibration listed in Chapter "Calibration" of the operating manual must be fulfilled.

▶ Read out the cell current (in nA) at address 0x0002 and note it down.

▶ Perform the calibration as described in Chapter "Calibration" of the operating manual.

▶ Calculate the sensor slope by dividing the cell current noted down by the concentration of the disinfectant measured.

If the current determined for the zero point of the sensor is not zero, it must be subtracted from the cell current noted down before the division.

5.2.2 Saving the calibration data in the sensor

▶ Write the value "0" to address 0x0206. In case of a zero point calibration, write the current value determined in disinfectant- free process water in address 0x0206.

▶ Write the value for the slope calculated as described in Chapter 5.2.1, p. **Fehler! Textmarke nicht definiert.** to address 0x0208.

▶ Write the value for "Date/Time" to address 0x020A. The structure for this variable is "YYMMDDHHMM". An example for "Date/Time" is 1904101430 (→ Chapter 3.4, p. 9).

When "Date/Time" is written, the calibration is active.

5.2.3 Verification of the calibration

Verification of the data stored in the sensor

▶ Verify the values for the zero point and slope by reading out the values in addresses 0x0210 (X_Zero (0)) and 0x0212 (X_Span (0)).

▶ The value displayed at address 0x0210 should be "0" (or the value from the zero point calculation).

▶ The value from the calculation of the slope according to Chapter 5.2.1, p. 15 should be displayed at address 0x0212.

Analytical verification

- The concentration measured by the sensor and the concentration determined by analysis should largely be the same.

The calibration log book (history) can be used to view the values of the last 5 calibrations. Please note that the 6th calibration will overwrite the previously oldest calibration in the calibration log book.

Check the calibration regularly as described in Chapter “Maintenance” of the operating manual. For this purpose, perform the steps described from Chapter 5.2.1, p. 15 on. For a comparison, the value of the nominal slope can be read out at address 0x030A.

5.3 Recovery of the nominal slope

5.3.1 Setting the standard values for nominal slope and zero point

The sensor is not calibrated when delivered from our factory. If the sensor is new and has not been calibrated yet, the adjustment of the nominal slope as described below is not necessary.

- Read out the nominal slope at address 0x030A.

  Examples of nominal slopes are e.g. 22 nA/ppm for the sensor measuring free chlorine with a measuring range from 0 to 200 ppm (TARAbase Chlorine Measuring Cell CL4.2L) or 7.5 nA/ppm for the sensor measuring the total chlorine content with a measuring range from 0 to 20 ppm (TARAline Chlorine Measuring Cell CP4.0N).
- Write the nominal slope to address 0x0208.
- ↪ This step prepares the replacement of the slope determined before by calibration by the nominal slope and by storing the most recent calibration in the history.
- Write the value “0” to address 0x0206. This is the standard value for the zero point.
- Write the value for “Date/Time” to address 0x020A. The structure of the time stamp is explained in Chapter 3.4, p. 9.

The writing of the time stamp completes the setting of the nominal slope and the storage of the most recent calibration in the history.

5.3.2 Verification of the sensor slope value transmitted

If the sensor is new and has not been calibrated yet, the verification of the nominal slope as described below is not necessary.

- Verify the values in addresses 0x0210 (X_Zero (0)) and 0x0212 (X_Span (0)). The value in address 0x0210 should be “0” while the value for the nominal slope should be available in address 0x0212 (e.g. “7.5” for the sensor TARAline Chlorine Measuring Cell CP4.0N).

Reiss GmbH
Elektrochemische Messtechnik
Eisleber Str. 5
D - 69469 Weinheim
Germany