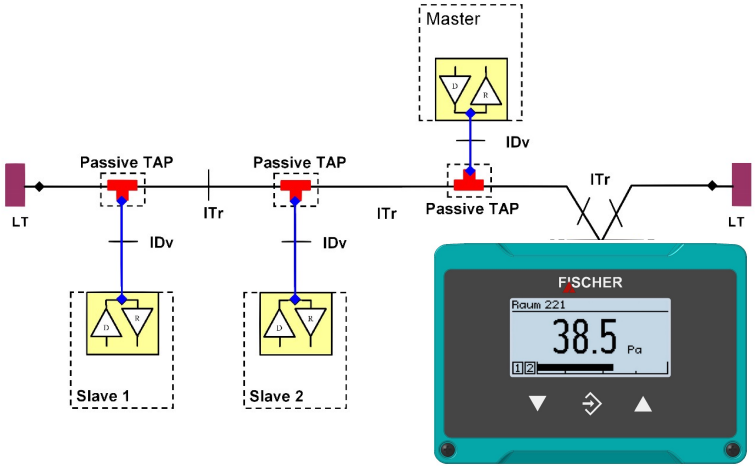
developing solutions

FISCHER
MESS- UND REGELTECHNIK

Modbus

CE

RoHS II COMPLIANT



Manual

Modbus RTU

Protocol description
for the product line DE90

09015216 • HB_EN_Modbus_DE90 • Rev. ST4-A • 08/19

Masthead

Manufacturer: **FISCHER Mess- und Regeltechnik GmbH**

Bielefelderstr. 37a
D-32107 Bad Salzuflen

Telephone: +49 5222 974 0
Telefax: +49 5222 7170

eMail: info@fischermesstechnik.de
web: www.fischermesstechnik.de

Technical editorial team: Documentation representative: T. Malischewski
Technical editor: R. Kleemann

Version history

Rev. ST4-A 08/19	Version 1 (first edition)

Table of contents

1 Introduction

The Modbus protocol is a communication protocol that is based on a Master/Slave architecture. All FISCHER products work in the operating mode Modbus RTU.

This manual is designed for readers with a basic understanding of the Modbus protocol. There are references to relevant specialist literature about this topic at the end of this manual.

1.1 Modbus infrastructure

Communication with the FISCHER units requires a serial two-wire bus (2W) in compliance with the EIA/TIA-485 standard. All connected units must be connected to a joint reference potential by means of a third (common) line. The bus is connected using a 150Ω 0.5W resistor. The pull up/down resistors are usually set on the master. Usually, up to 32 slaves can be connected without a repeater.

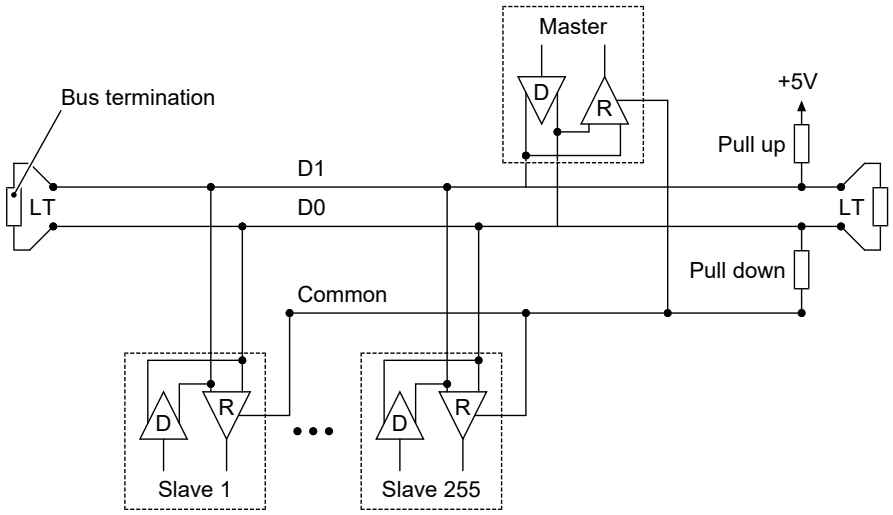


Fig. 1: Modbus infrastructure

NOTICE

Passive TAP

If the units are connected via a Passive TAP (e.g. T-adapter connection), the units can be disconnected from the bus without interrupting the bus.

1.2 Modbus RTU Protocol

The Modbus RTU transfers data in a binary form. A single master and up to 255 slaves can be connected at the same time to the serial Modbus.

The following basic rules apply.

- A Modbus transaction is only initialised by the Master.
- At the same time, only one Modbus transaction takes place.
- The Slave never sends data without a request from the Master.
- Slaves cannot communicate with each other.

1.3 Modbus Transaction

A Modbus transaction comprises two parts. A request from the Master and a response from the Slave.

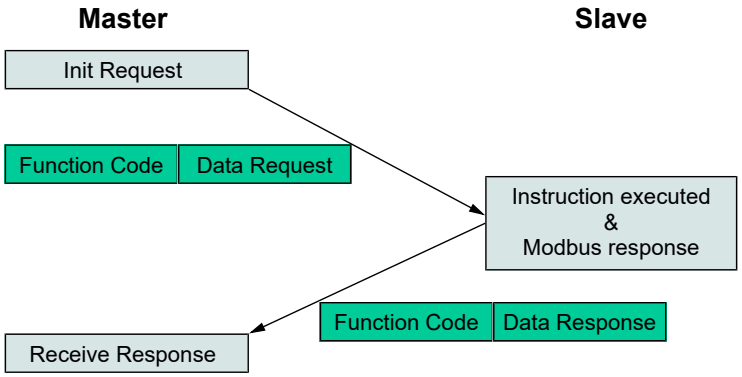


Fig. 2: Error-free request/response cycle

If an error occurs during a Modbus transaction, the Function Code is replaced with a special Function Code with an error indicator in the Modbus Response message and a more detailed description of the error in the data field is sent.

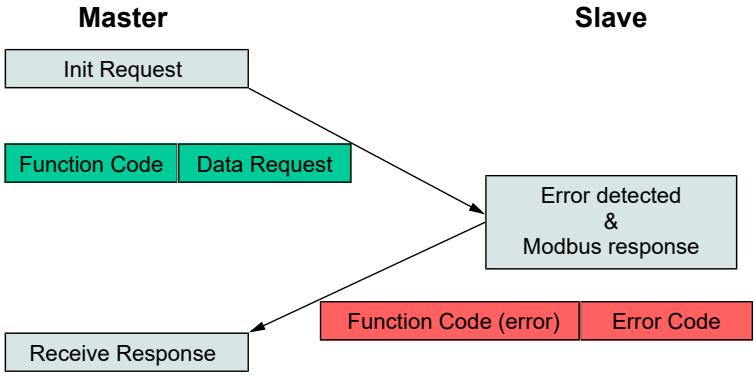


Fig. 3: Faulty request/response cycle

1.4 Modbus Frame

A Modbus data frame comprises two components.

- Protocoll Data Unit (PDU)
- Application Data Unit (ADU)

The inner data structure is the PDU and additional data fields are added for the encapsulation of the frame in the respective protocol of the data transmission.

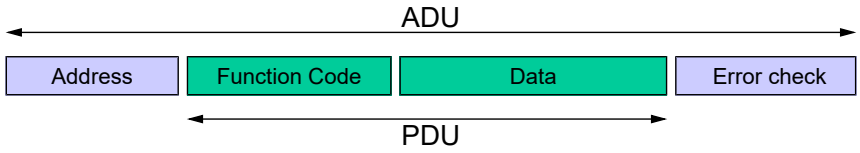


Fig. 4: MODBUS Frame

In the Modbus RTU protocol, the address field contains the Slave address. The address space comprises the addresses 1 to 255. If the Slave sends a Response, it positions its own address in the address field. This means that the Master 'knows' which Slave is sending. The Function Code states which action needs to be carried out. The following data field contains the Request and Response parameters. The error check field contains the result of a CRC review of the content of the transmission.

1.5 Modbus data transmission

In the RTU mode, each message is sent as a continuous binary flow of characters via the serial bus.

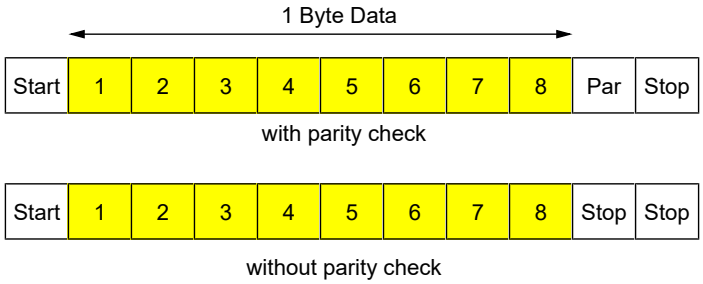


Fig. 5: Bit Sequence

The Even Parity is set as the default value for the parity bit. But an Odd Parity and No Parity can also be used. If No Parity is used, a further stop bit is added.

A Modbus message is set by the transferring unit in a so-called frame. The maximum size of a message is 256 byte. Start and end point of a frame are well defined. This allows the receiving unit to recognise the start and end of a message.

A transmission starts with a break of at least 3.5 characters (char). Then the frames are sent. Each frame must be followed by a pause interval (t3.5) with a length of at least 3.5 characters before the next frame is sent. There must be a pause interval (t1.5) between two characters with a maximum length of 1.5 characters. The entire transmission must be sent as a continuous flow of characters.

If the idle intervals are not satisfied, the character flow stops and the transmission is declared invalid.

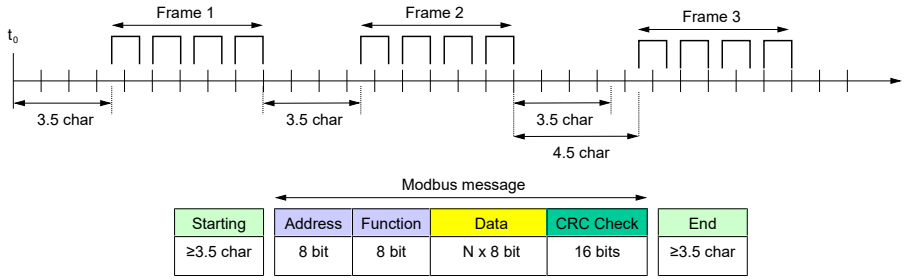


Fig. 6: Modbus Message Frame

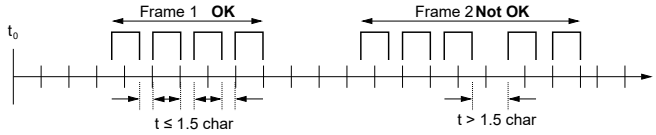


Fig. 7: Faulty transmission (example)

2 Functions

2.1 General

The Modus protocol has a series of options for access to the data:

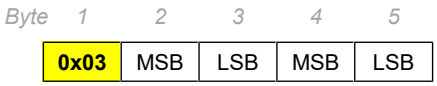
Type	Access	File	Code
Details	16 Bit	Read Holding Register	03
		Read Input Register	04
		Write Single Register	06
		Write Multiple Register	16
Dia-gnostics		Report Server ID	17

2.2 Function Code [03] "Read Holding Register"

This Function Code is used to read the Holding Register. The maximum possible number of registers that can be addressed in one message is 125.

Request

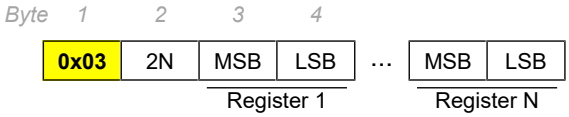
The request contains the address of the first register that is to be read and the number of registers that need to be read. The addressing of the register starts with 0; the numbering of the registers starts with 1.



Byte	Field name	Size	Value range
1	Function Code	1 byte	0x03
2.3	Start Address	2 bytes	0x0000 to 0xFFFF
4.5	Number of registers	2 bytes	0x0001 to 0x007D (1…125)

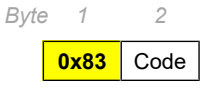
Response

The response contains two bytes for each read register; therefore the number of bytes is twice the number of registers (N).



Byte	Field name	Size	Value range
1	Function Code	1 byte	0x03
2	Number of bytes	2 bytes	2N
3.4	Holding Register	N x 2 Byte	16 Bit value

Error



Byte	Field name	Size	Value range
1	Function Code (error)	1 byte	0x83
2	Error code	1 byte	Code see table

The following error codes are possible:

0x01	The function is not supported
0x02	An invalid address is referenced
0x03	The request does not correspond to the expected format; the number of requested registers is greater than 125

Example:

- Export Holding Register 108 to 110
- Content Register 108= 0x000A
- Content Register 109= 0x000B
- Content Register 110= 0x000C

Byte	1	2	3	4	5	6	7	8
Request	**0x03**	0x00	0x6B	0x00	0x03			
Response	**0x03**	0x06	0x00	0x0A	0x00	0x0B	0x00	0x0C

Request		Response	
Field name	**Value**	**Field name**	**Value**
Function Code	0x03	Function Code	0x03
Start Address MSB	0x00	Number of bytes	0x06
Start Address LSB	0x6B	Holding Register 108 MSB	0x00
Number of Registers MSB	0x00	Holding Register 108 LSB	0x0A
Number of Registers LSB	0x03	Holding Register 109 MSB	0x00
		Holding Register 109 LSB	0x0B
		Holding Register 110 MSB	0x00
		Holding Register 110 LSB	0x0C

2.3 Function Code [04] "Read Input Register"

This Function Code is used to read the input register. The maximum possible number of registers that can be addressed in one message is 125.

Request

The request contains the address of the first register that is to be read and the number of registers that need to be read. The addressing of the register starts with 0; the numbering of the registers starts with 1.

Byte	1	2	3	4	5
	0x04	MSB	LSB	MSB	LSB

Byte	Field name	Size	Value range
1	Function Code	1 byte	0x04
2.3	Start Address	2 bytes	0x0000 to 0xFFFF
4.5	Number of registers	2 bytes	0x0001 to 0x007D (1…125)

Response

The response contains two bytes for each read register; therefore the number of bytes is twice the number of registers.

Byte	1	2	3	4			
	0x04	2N	MSB	LSB	…	MSB	LSB
			Register 1			Register N	

Byte	Field name	Size	Value range
1	Function Code	1 byte	0x04
2	Number of bytes	2 bytes	2N
3.4	Content Register	N x 2 Byte	16 Bit value

Error

Byte	1	2
	0x84	Code

Byte		Size	Value range
1	Function Code (error)	1 byte	0x84
2	Error code	1 byte	Code see table

The following error codes are possible:

0x01	The function is not supported
0x02	An invalid address is referenced
0x03	The request does not correspond to the expected format; the number of requested registers is greater than 125

Example:

- Export Content Register 9
- Content Register 9= 0x000A

Byte	1	2	3	4	5
Request	**0x04**	0x00	0x08	0x00	0x01
Response	**0x04**	0x02	0x00	0x0A	

Request		Response	
Field name	**Value**	**Field name**	**Value**
Function Code	0x04	Function Code	0x04
Start Address MSB	0x00	Number of bytes	0x02
Start Address LSB	0x08	Input Register 9 MSB	0x00
Number of Registers MSB	0x00	Input Register 9 LSB	0x0A
Number of Registers LSB	0x01		

2.4 Function Code [06] "Write Single Register"

This Function Code is used to write a single Holding register.

Request

The request contains the address of the register that is to be written and value that is to be written.

Byte	1	2	3	4	5
	0x06	MSB	LSB	MSB	LSB

Byte	Field name	Size	Value range
1	Function Code	1 byte	0x06
2.3	Register Address	2 bytes	0x0000 to 0xFFFF
4.5	Register Value	2 bytes	0x0000 to 0xFFFF

Response

The response contains the address regster and the written value.

Byte	1	2	3	4	5
	0x06	MSB	LSB	MSB	LSB

Byte	Field name	Size	Value range
1	Function Code	1 byte	0x06
2	Register Address	2 bytes	0x0000 to 0xFFFF
3.4	Register Value	2 bytes	0x0000 to 0xFFFF

Error

Byte	1	2
	0x86	Code

Byte		Size	Value range
1	Function Code (error)	1 byte	0x86
2	Error code	1 byte	Code see table

The following error codes are possible:

0x01	The function is not supported
0x02	An invalid address is referenced
0x03	The request does not correspond to the expected format

Example:

- Write register 2
- Value that needs to be written = 0x0003

Byte	1	2	3	4	5
Request	**0x06**	0x00	0x01	0x00	0x03
Response	**0x06**	0x00	0x01	0x00	0x03

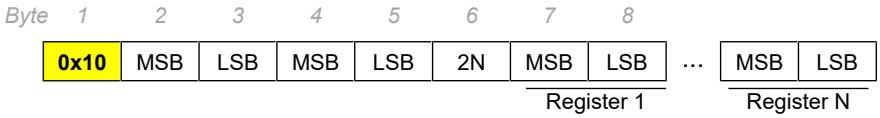
Request		Response	
Field name	**Value**	**Field name**	**Value**
Function Code	0x06	Function Code	0x06
Register Address MSB	0x00	Register Address MSB	0x00
Register Address LSB	0x01	Register Address LSB	0x01
Register Value MSB	0x00	Register Value MSB	0x00
Register Value LSB	0x03	Register Value LSB	0x03

2.5 Function Code [16] "Write Multiple Registers"

This Function Code is used to write a block of sequential registers. The maximum possible number of registers that can be addressed in one message is 123.

Request

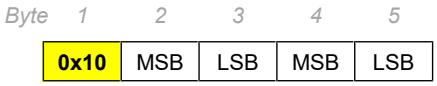
The request contains the address of the first register that is to be written and the number of registers that need to be written. The addressing of the register starts with 0; the numbering of the registers starts with 1.



Byte	Field name	Size	Value range
1	Function Code	1 byte	0x10
2.3	Start Address	2 bytes	0x0000 to 0xFFFF
4.5	Number of registers	2 bytes	0x0001 to 0x007B (1…123)
6	Number of bytes	1 byte	2 x N
7.8	Register Value	N x 2 Byte	Value

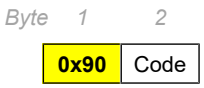
Response

The response contains the start address and the number of written registers.



Byte	Field name	Size	Value range
1	Function Code	1 byte	0x10
2.3	Start Address	2 bytes	0x0000 to 0xFFFF
4.5	Number of Registers	2 bytes	0x0001 to 0x007B (1…123)

Error



Byte		Size	Value range
1	Function Code (error)	1 byte	0x90
2	Error code	1 byte	Code see table

The following error codes are possible:

0x01	The function is not supported
0x02	An invalid address is referenced
0x03	The request does not correspond to the expected format; the number of requested registers is greater than 123; the number of data bytes does not match the number of registers

Example:

- Write 2 registers
- Start Address = 0x0001
- Content Register 2= 0x000A
- Content Register 3= 0x0102

Byte	1	2	3	4	5	6	7	8	9	10
Request	0x10	0x00	0x01	0x00	0x02	0x04	0x00	0x0A	0x01	0x02
Response	0x10	0x00	0x01	0x00	0x02					

Request		Response	
Field name	**Value**	**Field name**	**Value**
Function Code	0x10	Function Code	0x10
Start Address MSB	0x00	Start Address MSB	0x00
Start Address LSB	0x01	Start Address LSB	0x01
Number of Registers MSB	0x00	Number of Registers MSB	0x00
Number of Registers LSB	0x02	Number of Registers LSB	0x02
Number of bytes	0x04		
Register Value MSB	0x00		
Register Value LSB	0x0A		
Register Value MSB	0x01		
Register Value LSB	0x02		

2.6 Function Code [17] "Report Server ID"

The Function Code allows the query of unit-specific data.

Request

The response only contains the Function Code.

Byte	1
	0x11

Byte	Field name	Size	Value range
1	Function Code	1 byte	0x11

Response

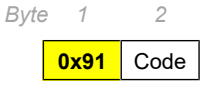
The response comprises two data bytes. The last byte is always 0xFF.

Byte	1	2	3	4
	0x11	0x02	Byte1	0xFF

Byte	Field name	Size	Value range
1	Function Code	1 byte	0x11
2	Number of bytes	1 byte	0x02
3	Byte 1	1 byte	Code see table
4	End	1 byte	0xFF

Dev.	Byte 1
DE90	0x01

Error



Byte	Field name	Size	Value range
1	Function Code (error)	1 byte	0x91
2	Error code	1 byte	Code see table

The following error codes are possible:

0x01	The function is not supported
0x03	The request does not correspond to the expected format

3 Data types

3.1 Integer (16 Bit)

- Standard format for register
- Comprises two bytes in a Modbus message
- The higher-valued byte (Bits 8 to 15) is always sent first
- The two-complement format is used for integers with a preceding sign.

	Value range
unsigned Integer	0 … 65535
signed Integer	-32768 … +32767

3.2 Float

- Floating point numbers are transferred in the IEEE-475 Single Precision format.
- It comprises two registers (four bytes).
- The byte sequence can be changed between the Big Endian- (the value with the highest value first) and the Little Endian format (the value with the lowest value first).

For example:

$1234.56_{10} = 449A51EC_{16}$

Format	Reg. 1 MSB	Reg. 1 LSB	Reg. 2 MSB	Reg. 2 LSB
Big Endian	0x44	0x9A	0x51	0xEC
Little Endian	0xEC	0x51	0x9A	0x44

3.3 Character

- Character strings are transmitted with two characters per register.
- The preceding character is stored in the most significant byte (MSB) and the following character in the least significant byte (LSB) of the register.
- For strings with odd length the last character is always a zero character (0x00).

Example:

Character string = "FISCHER"

Reg. 1 MSB	Reg. 1 LSB	Reg. 2 MSB	Reg. 2 LSB	Reg. 3 MSB	Reg. 3 LSB	Reg. 4 MSB	Reg. 4 LSB
'F'	'I'	'S'	'C'	'H'	'E'	'R'	'\0'
0x46	0x49	0x53	0x43	0x48	0x45	0x52	0x00

4 Addresses

The following abbreviations are used:

Data type	Abbrevi-ation	Discription
Float	Float	Floating point number
Unsigned integer	UInt	Integer without sign
Signed integer	SInt	Integer with sign
Character	Char	Character string

Other abbreviations		
	BMR	Basic measuring range (see type plate)
	MR	Measuring range determined by the parameters MBA and MBE
	MRS	Measuring range start
	MRE	Measuring range end
	ColTH	Colour change (threshold)

4.1 Measured values

Reg.	Address		length	Format	Category	Measured values	Access	
	Dec.	Hex.					Read	Write
1	0	0x0000	2	Float		**Measured value channel 1**	x	
2	1	0x0001						
3	2	0x0002	1	SInt		**Colour change channel 1**	x	
					Value	–2: low / red		
					Value	–1: low / yellow		
					Value	0: ok /green		
					Value	1: high / yellow		
					Value	2: high / red		
4	3	0x0003	1	SInt		**Error signal channel 1**	x	
					Value	0: no error		
					Value	1: error		
5	4	0x0004	2	Float		**Measured value channel 2**	x	
6	5	0x0005						
7	6	0x0006	1	SInt		**Colour change channel 2**	x	
					Value	–2: low / red		
					Value	–1: low / yellow		
					Value	0: ok /green		
					Value	1: high / yellow		
					Value	2: high / red		
8	7	0x0007	1	SInt		**Error signal channel 2**	x	
					Value	0: no error		
					Value	1: error		
9	8	0x0008	2	Float		**Measured value channel 3**	x	
10	9	0x0009						
11	10	0x000A	1	SInt		**Colour change channel 3**	x	
					Value	–2: low / red		
					Value	–1: low / yellow		
					Value	0: ok /green		
					Value	1: high / yellow		
					Value	2: high / red		
12	11	0x000B	1	SInt		**Error signal channel 3**	x	
					Value	0: no error		
					Value	1: error		

4.2 Configuration channel 1 (differential pressure)

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
1001	1000	0x03E8	1	UInt		**Mode channel 1**	x	x
					Value	0: Linear characteristic curve		
					Value	1: Flow rate		
					Value	2: Table		
					Value	3: Volume flow		
					Value	4: Linear function		
1002	1001	0x03E9	1	UInt		**Measuring range channel 1 unit**	x	x
					Value	0: Pa		
					Value	1: kPa		
					Value	2: MPa		
					Value	3: bar		
					Value	4: mbar		
					Value	5: mmHg		
					Value	6: mmH_2O		
					Value	7: inH_2O		
					Value	8: psi		
1003	1002	0x03EA	2	Float		**Measuring range channel 1 start**	x	x
1004	1003	0x03EB						
					Value	BMR start…end		
1005	1004	0x03EC	2	Float		**Measuring range channel 1 end**	x	x
1006	1005	0x03ED						
					Value	BMR start…end		
1007	1006	0x03EE	1	UInt		**Damping channel 1**	x	x
					Value	0 … 30 s		
1008	1007	0x03EF	2	Float		**Offset channel 1**	x	x
1009	1008	0x03F0						
					Value	–⅓ BMR ... +⅓ BMR		
1010	1009	0x03F1	2	Float		**Zero-point window channel 1**	x	x
1011	1010	0x03F2						
					Value	0 ... +⅓ BMR		
1012	1011	0x03F3	1	UInt		**Limit channel 1**	x	x
					Value	0: Off		
					Value	1: On		
1013	1012	0x03F4	1	SInt		**Number format channel 1**	x	x
					Value	0: ±123456		
					Value	1: ±12345.6		
					Value	2: ±1234.45		
					Value	3: ±123,456		
					Value	4: ±12.3456		
					Value	5: ±1.23456		
1014	1013	0x03F5	2	Float		**Colour change channel 1 red-green**	x	x
1015	1014	0x03F6						
					Value	MRS -50 % … ColTH green-red		

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
1016	1015	0x03F7	2	Float		**Colour change channel 1 green-red**	x	x
1017	1016	0x03F8						
					Value	ColTH red-green … MRE +50%		
1018	1017	0x03F9	2	Float		**Colour change channel 1 red-yellow**	x	x
1019	1018	0x03FA						
					Value	MRS -50 % … ColTH yellow-green		
1020	1019	0x03FB	2	Float		**Colour change channel 1 yellow-green**	x	x
1021	1020	0x03FC						
					Value	ColTH red-yellow … ColTH green-yellow		
1022	1021	0x03FD	2	Float		**Colour change channel 1 green-yellow**	x	x
1023	1022	0x03FE						
					Value	ColTH yellow-green … ColTH yellow-red		
1024	1023	0x03FF	2	Float		**Colour change channel 1 yellow-red**	x	x
1025	1024	0x0400						
					Value	ColTH green-yellow … MR +50 %		
1026	1025	0x0401	2	Float		**Colour change channel 1 hysteresis**	x	x
1027	1026	0x0402						
					Value	0.1% … 10% MR		
1028	1027	0x0403	1	UInt		**Colour change channel 1 delay on**	x	x
					Value	0 … 100 s		
1029	1028	0x0404	1	UInt		**Colour change channel 1 delay off**	x	x
					Value	0 … 100 s		
Characteristic: flow rate								
1030	1029	0x0405	3	Char		**Display channel 1 unit**	x	x
1031	1030	0x0406						
1032	1031	0x0407						
					Value	5 characters		
1033	1032	0x0408	2	Float		**Display channel 1 start**	x	x
1034	1033	0x0409						
					Value	–999999 ... +999999		
1035	1034	0x040A	2	Float		**Display channel 1 end**	x	x
1036	1035	0x040B						
					Value	–999999 ... +999999		
Characteristic: Table								
1037	1036	0x040C	3	Char		**Display channel 1 unit**	x	x
1038	1037	0x040D						
1039	1038	0x040E						
					Value	5 characters		
1040	1039	0x040F	1	UInt		**Number of value pairs**	x	x
					Value	2 … 30		
1041	1040	0x0410	2	Float		**Input value 1**	x	x
1042	1041	0x0411			For all values	MRS … MRE		
1043	1042	0x0412	2	Float		**Input value 2**	x	x
1044	1043	0x0413						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
1045	1044	0x0414	2	Float		**Input value 3**	x	x
1046	1045	0x0415						
1047	1046	0x0416	2	Float		**Input value 4**	x	x
1048	1047	0x0417						
1049	1048	0x0418	2	Float		**Input value 5**	x	x
1050	1049	0x0419						
1051	1050	0x041A	2	Float		**Input value 6**	x	x
1052	1051	0x041B						
1053	1052	0x041C	2	Float		**Input value 7**	x	x
1054	1053	0x041D						
1055	1054	0x041E	2	Float		**Input value 8**	x	x
1056	1055	0x041F						
1057	1056	0x0420	2	Float		**Input value 9**	x	x
1058	1057	0x0421						
1059	1058	0x0422	2	Float		**Input value 10**	x	x
1060	1059	0x0423						
1061	1060	0x0424	2	Float		**Input value 11**	x	x
1062	1061	0x0425						
1063	1062	0x0426	2	Float		**Input value 12**	x	x
1064	1063	0x0427						
1065	1064	0x0428	2	Float		**Input value 13**	x	x
1066	1065	0x0429						
1067	1066	0x042A	2	Float		**Input value 14**	x	x
1068	1067	0x042B						
1069	1068	0x042C	2	Float		**Input value 15**	x	x
1070	1069	0x042D						
1071	1070	0x042E	2	Float		**Input value 16**	x	x
1072	1071	0x042F						
1073	1072	0x0430	2	Float		**Input value 17**	x	x
1074	1073	0x0431						
1075	1074	0x0432	2	Float		**Input value 18**	x	x
1076	1075	0x0433						
1077	1076	0x0434	2	Float		**Input value 19**	x	x
1078	1077	0x0435						
1079	1078	0x0436	2	Float		**Input value 20**	x	x
1080	1079	0x0437						
1081	1080	0x0438	2	Float		**Input value 21**	x	x
1082	1081	0x0439						
1083	1082	0x043A	2	Float		**Input value 22**	x	x
1084	1083	0x043B						
1085	1084	0x043C	2	Float		**Input value 23**	x	x
1086	1085	0x043D						
1087	1086	0x043E	2	Float		**Input value 24**	x	x
1088	1087	0x043F						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
1089	1088	0x0440	2	Float		**Input value 25**	x	x
1090	1089	0x0441						
1091	1090	0x0442	2	Float		**Input value 26**	x	x
1092	1091	0x0443						
1093	1092	0x0444	2	Float		**Input value 27**	x	x
1094	1093	0x0445						
1095	1094	0x0446	2	Float		**Input value 28**	x	x
1096	1095	0x0447						
1097	1096	0x0448	2	Float		**Input value 29**	x	x
1098	1097	0x0449						
1099	1098	0x044A	2	Float		**Input value 30**	x	x
1100	1099	0x044B						
1101	1100	0x044C	2	Float		**Display value 1**	x	x
1102	1101	0x044D			For all values	–999999 ... +99999		
1103	1102	0x044E	2	Float		**Display value 2**	x	x
1104	1103	0x044F						
1105	1104	0x0450	2	Float		**Display value 3**	x	x
1106	1105	0x0451						
1107	1106	0x0452	2	Float		**Display value 4**	x	x
1108	1107	0x0453						
1109	1108	0x0454	2	Float		**Display value 5**	x	x
1110	1109	0x0455						
1111	1110	0x0456	2	Float		**Display value 6**	x	x
1112	1111	0x0457						
1113	1112	0x0458	2	Float		**Display value 7**	x	x
1114	1113	0x0459						
1115	1114	0x045A	2	Float		**Display value 8**	x	x
1116	1115	0x045B						
1117	1116	0x045C	2	Float		**Display value 9**	x	x
1118	1117	0x045D						
1119	1118	0x045E	2	Float		**Display value 10**	x	x
1120	1119	0x045F						
1121	1120	0x0460	2	Float		**Display value 11**	x	x
1122	1121	0x0461						
1123	1122	0x0462	2	Float		**Display value 12**	x	x
1124	1123	0x0463						
1125	1124	0x0464	2	Float		**Display value 13**	x	x
1126	1125	0x0465						
1127	1126	0x0466	2	Float		**Display value 14**	x	x
1128	1127	0x0467						
1129	1128	0x0468	2	Float		**Display value 15**	x	x
1130	1129	0x0469						
1131	1130	0x046A	2	Float		**Display value 16**	x	x
1132	1131	0x046B						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
1133	1132	0x046C	2	Float		**Display value 17**	x	x
1134	1133	0x046D						
1135	1134	0x046E	2	Float		**Display value 18**	x	x
1136	1135	0x046F						
1137	1136	0x0470	2	Float		**Display value 19**	x	x
1138	1137	0x0471						
1139	1138	0x0472	2	Float		**Display value 20**	x	x
1140	1139	0x0473						
1141	1140	0x0474	2	Float		**Display value 21**	x	x
1142	1141	0x0475						
1143	1142	0x0476	2	Float		**Display value 22**	x	x
1144	1143	0x0477						
1145	1144	0x0478	2	Float		**Display value 23**	x	x
1146	1145	0x0479						
1147	1146	0x047A	2	Float		**Display value 24**	x	x
1148	1147	0x047B						
1149	1148	0x047C	2	Float		**Display value 25**	x	x
1150	1149	0x047D						
1151	1150	0x047E	2	Float		**Display value 26**	x	x
1152	1151	0x047F						
1153	1152	0x0480	2	Float		**Display value 27**	x	x
1154	1153	0x0481						
1155	1154	0x0482	2	Float		**Display value 28**	x	x
1156	1155	0x0483						
1157	1156	0x0484	2	Float		**Display value 29**	x	x
1158	1157	0x0485						
1159	1158	0x0486	2	Float		**Display value 30**	x	x
1160	1159	0x0487						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
Characteristic: Volume flow								
1161	1160	0x0488	1	UInt		**Display channel 1 unit**	x	x
					Value	0: m^3/h		
					Value	1: l/min		
					Value	2: cfm		
1162	1161	0x0489	2	Float		**Display channel 1 end**	x	x
1163	1162	0x048A						
					Value	0 ... 999999		
1164	1163	0x048B	2	Float		**K factor channel 1**	x	x
1165	1164	0x048C						
					Value	0 ... 9999		
1166	1165	0x048D	2	Float		**Air density channel 1**	x	x
1167	1166	0x048E						
					Value	0.25 … 2		
1168	1167	0x048F	1	UInt		**Formula channel 1**	x	x
					Value	0: Standard		
					Value	1: Comefri		
					Value	2: EBM Papst		
					Value	3: Fläkt Woods		
					Value	4: Nicotra Gebhardt		
					Value	5: Rosenberg		
					Value	6: Ziehl-Abegg		
Characteristic curve: Linear function								
1169	1168	0x0490	3	Char		**Display channel 1 unit**	x	x
1170	1169	0x0491						
1171	1170	0x0492						
					Value	5 characters		
1172	1171	0x493	2	Float		**Display channel 1 start**	x	x
1173	1172	0x494						
					Value	-999999 … +999999		
1174	1173	0x495	2	Float		**Display channel 1 end**	x	x
1175	1174	0x496						
					Value	-999999 … +999999		
1176	1175	0x497	2	Float		**Slope channel 1**	x	x
1177	1176	0x498						
					Value	-10 … +10		
1178	1177	0x499	2	Float		**Offset channel 1**	x	x
1179	1178	0x59A						
					Value	-999999 … +999999		

4.3 Configuration channel 2 (differential pressure)

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
2001	2000	0x07D0	1	UInt		**Mode channel 2**	x	x
					Value	0: Linear characteristic curve		
					Value	1: Flow rate		
					Value	2: Table		
					Value	3: Volume flow		
					Value	4: Linear function		
2002	2001	0x07D1	1	UInt		**Measuring range channel 2 unit**	x	x
					Value	0: Pa		
					Value	1: kPa		
					Value	2: MPa		
					Value	3: bar		
					Value	4: mbar		
					Value	5: mmHg		
					Value	6: mmH_2O		
					Value	7: inH_2O		
					Value	8: psi		
2003	2002	0x07D2	2	Float		**Measuring range channel 2 start**	x	x
2004	2003	0x07D3						
					Value	BMR start…end		
2005	2004	0x07D4	2	Float		**Measuring range channel 2 end**	x	x
2006	2005	0x07D5						
					Value	BMR start…end		
2007	2006	0x07D6	1	UInt		**Damping channel 2**	x	x
					Value	0 … 30 s		
2008	2007	0x07D7	2	Float		**Offset channel 2**	x	x
2009	2008	0x07D8						
					Value	–⅓ BMR ... +⅓ BMR		
2010	2009	0x07D9	2	Float		**Zero-point window channel 2**	x	x
2011	2010	0x07DA						
					Value	0 ... +⅓ BMR		
2012	2011	0x07DB	1	UInt		**Limit channel 2**	x	x
					Value	0: Off		
					Value	1: On		
2013	2012	0x07DC	1	SInt		**Number format channel 2**	x	x
					Value	0: ±123456		
					Value	1: ±12345.6		
					Value	2: ±1234.45		
					Value	3: ±123,456		
					Value	4: ±12.3456		
					Value	5: ±1.23456		
2014	2013	0x07DD	2	Float		**Colour change channel 2 red-green**	x	x
2015	2014	0x07DE						
					Value	MRS -50 % … ColTH green-red		

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
2016	2015	0x07DF	2	Float		**Colour change channel 2 green-red**	x	x
2017	2016	0x07E0						
					Value	ColTH red-green … MRE +50%		
2018	2017	0x07E1	2	Float		**Colour change channel 2 red-yellow**	x	x
2019	2018	0x07E2						
					Value	MRS -50 % … ColTH yellow-green		
2020	2019	0x07E3	2	Float		**Colour change channel 2 yellow-green**	x	x
2021	2020	0x07E4						
					Value	ColTH red-yellow … ColTH green-yellow		
2022	2021	0x07E5	2	Float		**Colour change channel 2 green-yellow**	x	x
2023	2022	0x07E6						
					Value	ColTH yellow-green … ColTH yellow-red		
2024	2023	0x07E7	2	Float		**Colour change channel 2 yellow-red**	x	x
2025	2024	0x07E8						
					Value	ColTH green-yellow … MR +50 %		
2026	2025	0x07E9	2	Float		**Colour change channel 2 hysteresis**	x	x
2027	2026	0x07EA						
					Value	0.1% … 10% MR		
2028	2027	0x07EB	1	UInt		**Colour change channel 2 delay on**	x	x
					Value	0 … 100 s		
2029	2028	0x07EC	1	UInt		**Colour change channel 2 delay off**	x	x
					Value	0 … 100 s		
Characteristic: flow rate								
2030	2029	0x07ED	3	Char		**Display channel 2 unit**	x	x
2031	2030	0x07EE						
2032	2031	0x07EF						
					Value	5 characters		
2033	2032	0x07F0	2	Float		**Display channel 2 start**	x	x
2034	2033	0x07F1						
					Value	–999999 ... +99999		
2035	2034	0x07F2	2	Float		**Display channel 2 end**	x	x
2036	2035	0x07F3						
					Value	–999999 ... +99999		
Characteristic: Table								
2037	2036	0x07F4	3	Char		**Display channel 2 unit**	x	x
2038	2037	0x07F5						
2039	2038	0x07F6						
					Value	5 characters		
2040	2039	0x07F7	1	UInt		**Number of value pairs**	x	x
					Value	2 … 30		
2041	2040	0x07F8	2	Float		**Input value 1**	x	x
2042	2041	0x07F9			For all values	MRS … MRE		
2043	2042	0x07FA	2	Float		**Input value 2**	x	x
2044	2043	0x07FB						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
2045	2044	0x07FC	2	Float		**Input value 3**	x	x
2046	2045	0x07FD						
2047	2046	0x07FE	2	Float		**Input value 4**	x	x
2048	2047	0x07FF						
2049	2048	0x0800	2	Float		**Input value 5**	x	x
2050	2049	0x0801						
2051	2050	0x0802	2	Float		**Input value 6**	x	x
2052	2051	0x0803						
2053	2052	0x0804	2	Float		**Input value 7**	x	x
2054	2053	0x0805						
2055	2054	0x0806	2	Float		**Input value 8**	x	x
2056	2055	0x0807						
2057	2056	0x0808	2	Float		**Input value 9**	x	x
2058	2057	0x0809						
2059	2058	0x080A	2	Float		**Input value 10**	x	x
2060	2059	0x080B						
2061	2060	0x080C	2	Float		**Input value 11**	x	x
2062	2061	0x080D						
2063	2062	0x080E	2	Float		**Input value 12**	x	x
2064	2063	0x080F						
2065	2064	0x0810	2	Float		**Input value 13**	x	x
2066	2065	0x0811						
2067	2066	0x0812	2	Float		**Input value 14**	x	x
2068	2067	0x0813						
2069	2068	0x0814	2	Float		**Input value 15**	x	x
2070	2069	0x0815						
2071	2070	0x0816	2	Float		**Input value 16**	x	x
2072	2071	0x0817						
2073	2072	0x0818	2	Float		**Input value 17**	x	x
2074	2073	0x0819						
2075	2074	0x081A	2	Float		**Input value 18**	x	x
2076	2075	0x081B						
2077	2076	0x081C	2	Float		**Input value 19**	x	x
2078	2077	0x081D						
2079	2078	0x081E	2	Float		**Input value 20**	x	x
2080	2079	0x081F						
2081	2080	0x0820	2	Float		**Input value 21**	x	x
2082	2081	0x0821						
2083	2082	0x0822	2	Float		**Input value 22**	x	x
2084	2083	0x0823						
2085	2084	0x0824	2	Float		**Input value 23**	x	x
2086	2085	0x0825						
2087	2086	0x0826	2	Float		**Input value 24**	x	x
2088	2087	0x0827						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
2089	2088	0x0828	2	Float		**Input value 25**	x	x
2090	2089	0x0829						
2091	2090	0x082A	2	Float		**Input value 26**	x	x
2092	2091	0x082B						
2093	2092	0x082C	2	Float		**Input value 27**	x	x
2094	2093	0x082D						
2095	2094	0x082E	2	Float		**Input value 28**	x	x
2096	2095	0x082F						
2097	2096	0x0830	2	Float		**Input value 29**	x	x
2098	2097	0x0831						
2099	2098	0x0832	2	Float		**Input value 30**	x	x
2100	2099	0x0833						
2101	2100	0x0834	2	Float		**Display value 1**	x	x
2102	2101	0x0835			For all values	–999999 ... +99999		
2103	2102	0x0836	2	Float		**Display value 2**	x	x
2104	2103	0x0837						
2105	2104	0x0838	2	Float		**Display value 3**	x	x
2106	2105	0x0839						
2107	2106	0x083A	2	Float		**Display value 4**	x	x
2108	2107	0x083B						
2109	2108	0x083C	2	Float		**Display value 5**	x	x
2110	2109	0x083D						
2111	2110	0x083E	2	Float		**Display value 6**	x	x
2112	2111	0x083F						
2113	2112	0x0840	2	Float		**Display value 7**	x	x
2114	2113	0x0841						
2115	2114	0x0842	2	Float		**Display value 8**	x	x
2116	2115	0x0843						
2117	2116	0x0844	2	Float		**Display value 9**	x	x
2118	2117	0x0845						
2119	2118	0x0846	2	Float		**Display value 10**	x	x
2120	2119	0x0847						
2121	2120	0x0848	2	Float		**Display value 11**	x	x
2122	2121	0x0849						
2123	2122	0x084A	2	Float		**Display value 12**	x	x
2124	2123	0x084B						
2125	2124	0x084C	2	Float		**Display value 13**	x	x
2126	2125	0x084D						
2127	2126	0x084E	2	Float		**Display value 14**	x	x
2128	2127	0x084F						
2129	2128	0x0850	2	Float		**Display value 15**	x	x
2130	2129	0x0851						
2131	2130	0x0852	2	Float		**Display value 16**	x	x
2132	2131	0x0853						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
2133	2132	0x0854	2	Float		**Display value 17**	x	x
2134	2133	0x0855						
2135	2134	0x0856	2	Float		**Display value 18**	x	x
2136	2135	0x0857						
2137	2136	0x0858	2	Float		**Display value 19**	x	x
2138	2137	0x0859						
2139	2138	0x085A	2	Float		**Display value 20**	x	x
2140	2139	0x085B						
2141	2140	0x085C	2	Float		**Display value 21**	x	x
2142	2141	0x085D						
2143	2142	0x085E	2	Float		**Display value 22**	x	x
2144	2143	0x085F						
2145	2144	0x0860	2	Float		**Display value 23**	x	x
2146	2145	0x0861						
2147	2146	0x0862	2	Float		**Display value 24**	x	x
2148	2147	0x0863						
2149	2148	0x0864	2	Float		**Display value 25**	x	x
2150	2149	0x0865						
2151	2150	0x0866	2	Float		**Display value 26**	x	x
2152	2151	0x0867						
2153	2152	0x0868	2	Float		**Display value 27**	x	x
2154	2153	0x0869						
2155	2154	0x086A	2	Float		**Display value 28**	x	x
2156	2155	0x086B						
2157	2156	0x086C	2	Float		**Display value 29**	x	x
2158	2157	0x086D						
2159	2158	0x086E	2	Float		**Display value 30**	x	x
2160	2159	0x086F						

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
Characteristic: Volume flow								
2161	2160	0x0870	1	UInt		**Display channel 2 unit**	x	x
					Value	0: m^3/h		
					Value	1: l/min		
					Value	2: cfm		
2162	2161	0x0871	2	Float		**Display channel 2 end**	x	x
2163	2162	0x0872						
					Value	0 ... 999999		
2164	2163	0x0873	2	Float		**K factor channel 2**	x	x
2165	2164	0x0874						
					Value	0 ... 9999		
2166	2165	0x0875	2	Float		**Air density channel 2**	x	x
2167	2166	0x0876						
					Value	0.25 … 2		
2168	2167	0x0877	1	UInt		**Formula channel 2**	x	x
					Value	0: Standard		
					Value	1: Comefri		
					Value	2: EBM Papst		
					Value	3: Fläkt Woods		
					Value	4: Nicotra Gebhardt		
					Value	5: Rosenberg		
					Value	6: Ziehl-Abegg		
Characteristic curve: Linear function								
2169	2168	0x0878	3	Char		**Display channel 2 unit**	x	x
2170	2169	0x0879						
2171	2170	0x087A						
					Value	5 characters		
2172	2171	0x087B	2	Float		**Display channel 2 start**	x	x
2173	2172	0x087C						
					Value	-999999 … +999999		
2174	2173	0x087D	2	Float		**Display channel 2 end**	x	x
2175	2174	0x087E						
					Value	-999999 … +999999		
2176	2175	0x087F	2	Float		**Slope channel 2**	x	x
2177	2176	0x0880						
					Value	-10 … +10		
2178	2177	0x0881	2	Float		**Offset channel 2**	x	x
2179	2178	0x0882						
					Value	-999999 … +999999		

4.4 Configuration channel 3

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
3001	3000	0x0BB8	1	UInt		**Mode channel 3**	x	x
					Value	0: inactive		
					Value	1: Difference		
					Value	2: Dynamic filter monitoring		
3002	3001	0x0BB9	1	UInt		(reserved)		
3003	3002	0x0BBA	2	Float		(reserved)		
3004	3003	0x0BBB						
3005	3004	0x0BBC	2	Float		(reserved)		
3006	3005	0x0BBD						
3007	3006	0x0BBE	1	UInt		(reserved)		
3008	3007	0x0BBF	2	Float		(reserved)		
3009	3008	0x0BC0						
3010	3009	0x0BC1	2	Float		(reserved)		
3011	3010	0x0BC2						
3012	3011	0x0BC3	1	UInt		**Limit channel 3**		
					Value	0: Off		
					Value	1: On		
3013	3012	0x0BC4	1	UInt		**Number format channel 3**		
					Value	0: ±123456		
					Value	1: ±12345.6		
					Value	2: ±1234.45		
					Value	3: ±123,456		
					Value	4: ±12.3456		
					Value	5: ±1.23456		
3014	3013	0x0BC5	2	Float		**Colour change channel 3 red-green**	x	x
3015	3014	0x0BC6						
					Value	MRS -50 % … ColTW green-red		
3016	3015	0x0BC7	2	Float		**Colour change channel 3 green-red**	x	x
3017	3016	0x0BC8						
					Value	ColTW red-green … MRE +50%		
3018	3017	0x0BC9	2	Float		**Colour change channel 3 red-yellow**	x	x
3019	3018	0x0BCA						
					Value	MRS -50 % … ColTW yellow-green		
3020	3019	0x0BCB	2	Float		**Colour change channel 3 yellow-green**	x	x
3021	3020	0x0BCC						
					Value	ColTW red-yellow … ColTW green-yellow		
3022	3021	0x0BCD	2	Float		**Colour change channel 3 green-yellow**	x	x
3023	3022	0x0BCE						
					Value	ColTW yellow-green … ColTW yellow-red		
3024	3023	0x0BCF	2	Float		**Colour change channel 3 yellow-red**	x	x
3025	3024	0x0BD0						
					Value	ColTW green-yellow … MR +50 %		

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
3026	3025	0x0BD1	2	Float		**Colour change channel 3 hysteresis**	x	x
3027	3026	0x0BD2						
					Value	0.1% … 10% MR		
3028	3027	0x0BD3	1	UInt		**Colour change channel 3 delay on**	x	x
					Value	0 … 100 s		
3029	3028	0x0BD4	1	UInt		**Colour change channel 3 delay off**	x	x
					Value	0 … 100 s		
Difference of the input channels								
3030	3029	0x0BD5	1	UInt		**Display channel 3 unit**	x	x
					Value	0: Pa		
					Value	1: kPa		
					Value	2: MPa		
					Value	3: bar		
					Value	4: mbar		
					Value	5: mmHg		
					Value	6: mmH_2O		
					Value	7: inH_2O		
					Value	8: psi		
3031	3030	0x0BD6	2	Float		**Display channel 3 start**	x	x
3032	3031	0x0BD7						
					Value	-999999 … +999999		
3033	3032	0x0BD8	2	Float		**Display channel 3 end**	x	x
3034	3033	0x0BD9						
					Value	-999999 … +999999		
3035	3034	0x0BDA	1	UInt		**Formula channel 3**	x	x
					Value	0: Channel 1 – Channel 2		
					Value	1: Channel 2 – Channel 1		
Dynamic filter monitoring								
3036	3035	0x0BDB	2	Float		**Display channel 3 start**	x	x
3037	3036	0x0BDC						
					Value	-999999 … +999999		
3038	3037	0x0BDD	2	Float		**Display channel 3 end**	x	x
3039	3038	0x0BDE						
					Value	-999999 … +999999		
3040	3039	0x0BDF	1	UInt		**Channel Δp**	x	x
					Value	1: Channel 1		
					Value	2: Channel 2		
3041	3040	0x0BE0	1	UInt		**Channel Q**	x	x
					Value	1: Channel 1		
					Value	2: Channel 2		
3042	3041	0x0BE1	1	UInt		**Approximation**	x	x
					Value	0: linear		
					Value	1: square root		

Reg.	Address		length	Format	Category	Parameters/value	Access	
	Dec.	Hex.					Read	Write
3043	3042	0x0BE2	2	Float		**Δp clean**	x	x
3044	3043	0x0BE3						
					Value	0 ... 999999		
3045	3044	0x0BE4	2	Float		**Δp soiled**	x	x
3046	3045	0x0BE5						
					Value	0 ... 999999		
3047	3046	0x0BE6	2	Float		**Δp Correction value**	x	x
3048	3047	0x0BE7						
					Value	-999999 ... +999999		
3049	3048	0x0BE8	2	Float		**Max. volume flow**	x	x
3050	3049	0x0BE9						
					Value	0 ... 999999		
3051	3050	0x0BEA	2	Float		**Min. volume flow**	x	x
3052	3051	0x0BEB						
					Value	0 … 50 % max. volume flow		

4.5 Configuration display

Reg.	Address		length	Format	Category	Measured values	Access	
	Dec.	Hex.					Read	Write
7001	7000	0x1B58	1	UInt		**Language**	x	x
					Value	0: German		
					Value	1: English		
					Value	2: Spanish		
					Value	3: French		
					Value	4: Italian		
					Value	5: Portuguese		
					Value	6: Hungarian		
7002	7001	0x1B59	10	Char		**Designation**	x	x
7003	7002	0x1B5A			Value	20 characters		
7004	7003	0x1B5B						
7005	7004	0x1B5C						
7006	7005	0x1B5D						
7007	7006	0x1B5E						
7008	7007	0x1B5F						
7009	7008	0x1B60						
7010	7009	0x1B61						
7011	7010	0x1B62						
7012	7011	0x1B63	1	UInt		(reserved)	x	(x)
7013	7012	0x1B64	1	UInt		**Meas.data display**	x	x
					Value	1: Channel 1		
					Value	2: Channel 2		
					Value	4: Channel 3		
					Value	255: All channels		
7014	7013	0x1B65	1	UInt		**Colour change assignment**	x	x
					Bit 0	Chnl 1		
					Bit 1	Chnl 2		
					Bit 2	Channel 3		
7015	7014	0x1B66	1	UInt		**LCD colour**	x	x
					Value	0: Off		
					Value	1: green		
					Value	2: Blue		
					Value	3: white		
					Value	4: Red		
					Value	5: red/green (colour-change)		
					Value	6: red/yellow/green (colour change		
					Value	7: Cyan		
					Value	8: Yellow		
					Value	9: Magenta		
7016	7015	0x1B67	1	UInt		**LCD lighting**	x	x
					Value	0 … 600 s		
7017	7016	0x1B68	1	UInt		**LCD contrast**	x	x
					Value	-15 … +15		

Reg.	Address		length	Format	Category	Measured values	Access	
	Dec.	Hex.					Read	Write
9999	9998	0x270E	1	UInt		**Error indicator configuration**	x	

If an invalid value is written in a register, its original value is retained. The number of the first invalid parameter can be called up via the Register 9999. If a zero (0) is saved here, the last written configuration is valid.

5 Attachments

5.1 Literature

„IEEE Standard for Floating-Point Arithmetic.“ 29. 08 2008. <http://ieeexplore.ieee.org/document/4610935/>.

„Modbus Application Protocol v1.1b3.“ 26. 04 2012. <http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf>.

Glossary

ADU

The Application Data Unit (ADU) is the complete command / data block of the communication protocol.

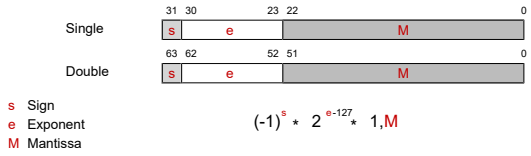
char

Abbreviation for character Char as a data type defines that the individual characters of a memory area each comprise (usually) 8 bits that represent a displayable character (letter, digit, special character...). The content of the memory point states which sign this is.

EIA-485

EIA-485, also called RS-485, is an industrial standard for an interface with asynchronous serial data transmission.

IEEE-475



The IEEE754 standard stipulates several data formats. The most important are the single and the double format. These formats comprise a sign bit s, the exponent e and the mantissa m.

Master/Slave

Master/Slave is a type of hierarchical administration of access to a common resource usually in the shape of a common data channel. A participant is the Master, all others are the Slaves. The Master is the only one authorised to access the joint resource without request. The Save cannot access the common resource on its own; it must wait until it is requested by the Master.

Message

Process of the transmission of data between a sender and one or more recipients.

PDU

The Protocol Data Unit (PDU) is the data block of a message.

Request

The request by a Master to a Slave to carry out the Function Code the transmission contains.

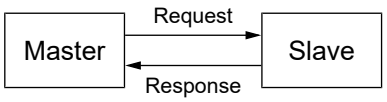
Response

Response by the Slave to the Master to a Request.

RTU

Remote Terminal Unit

Transaction



A transaction comprises a request from the Master and a response from the Slave.

FISCHER Mess- und Regeltechnik GmbH
Bielefelder Str. 37a
D-32107 Bad Salzuflen

Tel. +49 5222 974-0
Fax +49 5222 7170
www.fischermesstechnik.de
info@fischermesstechnik.de