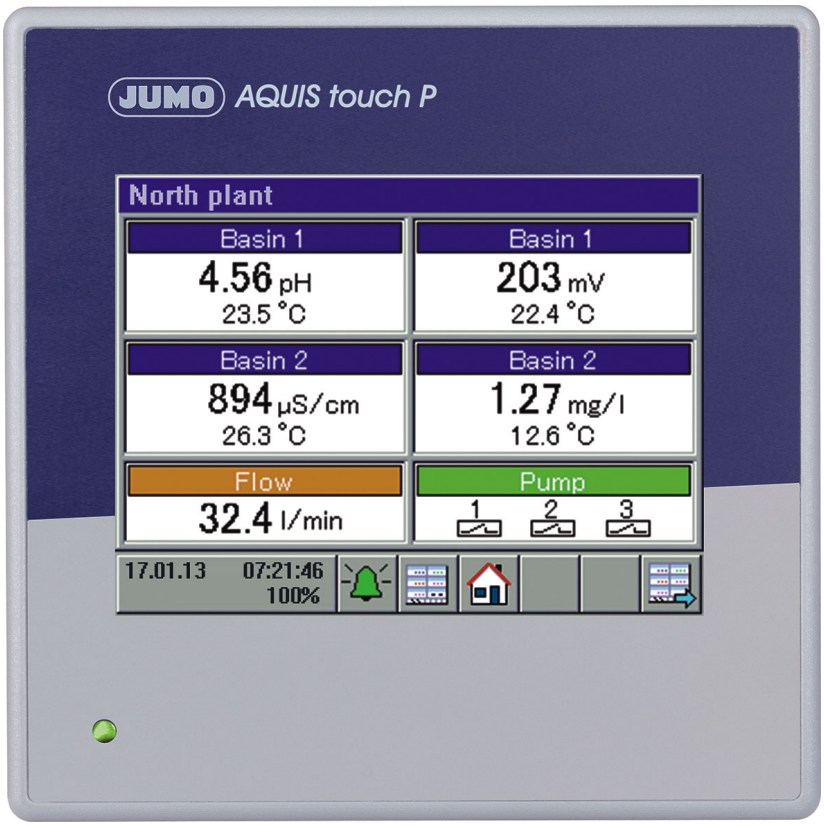
JUMO AQUIS touch P

Modular Multichannel Measuring Device for Liquid Analysis with Integrated Controller and Paperless Recorder



Modbus Interface description

20258000T92Z001K000

V4.00/EN/00614107/2025-10-22

Table of contents

1 Safety information . . . 7

1.1 Warning symbols . . . 7
1.2 Note signs . . . 7

2 Modbus protocol description . . . 9

2.1 Master-slave principle . . . 9
2.2 Transmission media for Modbus . . . 9
2.3 Structure of an RTU Modbus telegram . . . 10
2.4 Function codes . . . 10
2.4.1 Reading n words . . . 10
2.4.2 Writing one word . . . 11
2.4.3 Writing n words . . . 12
2.5 Data types . . . 12
2.6 Examples of data transmission . . . 14
2.6.1 Integer values . . . 14
2.6.2 Float values . . . 15
2.6.3 Character strings (texts) . . . 16
2.7 Checksum (CRC16) . . . 16
2.8 Error messages . . . 18
2.8.1 Modbus error codes . . . 18
2.8.2 Error messages for invalid values . . . 19
2.8.3 Error codes as integer feedback values . . . 19

3 Interfaces . . . 25

3.1 Location of interfaces . . . 25
3.2 Interface assignment . . . 26
3.2.1 Terminating resistors . . . 28

4 Configuring interfaces . . . 29

5 Modbus over a serial interface . . . 35

5.1 Modbus slave communication over an RS422/485 serial interface . . . 35
5.2 Modbus master operation via serial interface . . . 37

6 Modbus over Ethernet . . . 39

6.1 Modbus/TCP . . . 39
6.2 Networking with Modbus/TCP . . . 41
6.3 Modbus master with Modbus/TCP . . . 41
6.4 Modbus slave with Modbus/TCP . . . 41

Table of contents

7 Modbus frames 43

7.1 General information 43
7.2 Structure of the reading and writing processes 43
7.3 Compiling Modbus frames 45
7.3.1 Modbus frames for reading 45
7.3.2 Modbus frames for writing 48
7.4 Examples for the data transmission options with frames 53

8 Modbus address tables 55

8.1 Modbus master Actual cycle times 55
8.2 Configuration data and parameters 55
8.2.1 Basic settings 55
8.2.2 Temperature inputs 56
8.2.3 Universal inputs 56
8.2.4 Analysis inputs pH/Redox/NH 56
8.2.5 Analysis inputs CR (conductive conductivity) 57
8.2.6 Analysis inputs Ci (inductive conductivity) 58
8.2.7 External analog inputs 59
8.2.8 Manual values 61
8.2.9 Flow 61
8.2.10 Limit value monitoring 61
8.2.11 Controller parameter proportional band 1 63
8.2.12 Controller parameter proportional band 2 63
8.2.13 Controller parameter derivative time 1 63
8.2.14 Controller parameter derivative time 2 64
8.2.15 Controller parameter reset time 1 64
8.2.16 Controller parameter reset time 2 64
8.2.17 Controller parameter switching period 1 65
8.2.18 Controller parameter switching period 2 65
8.2.19 Controller parameter contact spacing 65
8.2.20 Controller parameter switch hyst. 1 66
8.2.21 Controller parameter switch hyst. 2 66
8.2.22 Controller parameter actuator time 66
8.2.23 Controller parameter working point 67
8.2.24 Controller parameter maximum output level 67
8.2.25 Controller parameter minimum output level 67
8.2.26 Controller parameter minimum relay energize time 1 68
8.2.27 Controller parameter minimum relay energize time 2 68
8.2.28 Controller parameter maximum pulse frequency 1 68
8.2.29 Controller parameter maximum pulse frequency 2 69
8.2.30 Controller parameter switch-on delay 1 69
8.2.31 Controller parameter switch-on delay 2 69
8.2.32 Controller parameter switch-off delay 1 70

8.2.33 Controller parameter switch-off delay 2 70
8.2.34 Controller parameter alarm tolerance 70
8.2.35 Controller parameter alarm delay 71
8.2.36 Controller parameter setpoint values 71
8.3 Process values 72
8.3.1 Date and time 72
8.3.2 Limit value monitoring alarm 1 72
8.3.3 Limit value monitoring alarm 2 72
8.3.4 Flow 73
8.3.5 Salinity 73
8.3.6 Washtimer 74
8.3.7 Controller 74
8.3.8 Timers 75
8.3.9 Counters 77
8.3.10 Calibration timers 77
8.3.11 Formula 78
8.3.12 Logic formula 79
8.3.13 analysis inputs 80
8.3.14 Universal inputs 81
8.3.15 Temperature inputs 81
8.3.16 Binary inputs 82
8.3.17 Analog outputs 82
8.3.18 Binary outputs 84
8.3.19 Service data 84
8.3.20 Hardware information 84
8.3.21 External analog inputs 84
8.3.22 External binary inputs inputs 87
8.3.23 Collective alarm 87
8.3.24 Ethernet 88
8.3.25 Modbus error 88
8.3.26 JUMO digiLine 89
8.4 Modbus frames 99
8.4.1 Reading frames 99
8.4.2 Writing frames 99

Table of contents

1.1 Warning symbols



DANGER!

This symbol indicates that **personal injury from electrocution** may occur if the appropriate precautionary measures are not taken.



WARNING!

This symbol in connection with the signal word indicates that **personal injury** may occur if the respective precautionary measures are not carried out.



CAUTION!

This symbol in connection with the signal word indicates that **material damage or data loss** will occur if the respective precautionary measures are not taken.



CAUTION!

This symbol indicates that **components could be destroyed** by electrostatic discharge (ESD = Electro Static Discharge) if the respective cautionary measures are not taken.

Only use the ESD packages intended for this purpose to return device inserts, assembly groups, or assembly components.



READ THE DOCUMENTATION!

This symbol, which is attached to the device, indicates that the associated **documentation for the device** must be **observed**. This is necessary to identify the nature of the potential hazard, and to take measures to prevent it.

1.2 Note signs



NOTE!

This symbol refers to **important information** about the product, its handling, or additional benefits.



REFERENCE!

This symbol refers to **additional information** in other sections, chapters, or other manuals.



FURTHER INFORMATION!

This symbol is used in tables and indicates that **further information** is provided after the table.



DISPOSAL!

At the end of its service life, the device and any batteries present do not belong in the trash! Please ensure that they are **disposed of** properly and in an **environmentally friendly** manner.

1 Safety information

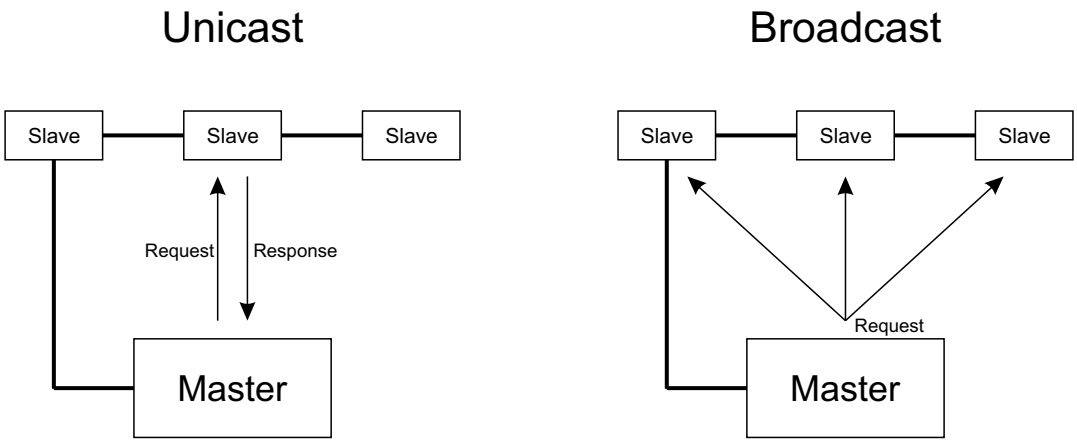
2.1 Master-slave principle

Communication between a master (e.g. a SCADA system or PLC) and a JUMO AQUIS touch P as slave via Modbus takes place on the basis of the master-slave principle in the form of a data inquiry/instruction – response. Depending on the transmission technology, the bus users (master and slaves) are addressed as follows:

- For **Modbus via a serial interface**, all slaves are identified by their device address (1 to 254). Master devices do not need an address.
- For **Modbus via Ethernet**, users are identified by their IP address. Slave responses are sent to the IP address for the master.

⇨ chapter 5 "Modbus over a serial interface", Page 35

⇨ chapter 6 "Modbus over Ethernet", Page 39



The master controls data exchange by cyclically requesting the slaves in the overall bus. The slaves (e.g. JUMO AQUIS touch P) have only a response function. The master has write and read access to the slaves. This enables data to be communicated between master and slave devices in real-time. Slaves cannot communicate with one another directly. In order to transmit data from slave to slave, the master must extract data from one slave and transfer it to the next.

As a rule, the master directs its queries to individual slaves. For this, it must communicate with the individual slaves by means of their unicast address. Queries can be directed to all slaves on the bus also in the form of a broadcast message. In this case, the broadcast address „0“ is used as the slave address. Slaves do not respond to broadcast queries. Data collisions would otherwise result in serial bus systems. For this reason, use of broadcast addresses only makes sense with function codes for writing data. Broadcasts cannot be used with function codes for reading data.

NOTE!

The JUMO AQUIS touch P can be operated as a master or slave.

2.2 Transmission media for Modbus

Serial interface

For data communication via a **serial interface**, the Modbus specification provides the transmission modes **RTU mode** (**R**emote **T**erminal **U**nit) and ASCII mode (transmission of data in ASCII format). The JUMO AQUIS touch P supports only the **RTU mode**. Here, the data are transmitted over the serial bus in binary format (RS422/485).

⇨ chapter 5 "Modbus over a serial interface", Page 35

2 Modbus protocol description

Ethernet

Use of an **Ethernet network** is possible with the aid of **Modbus/TCP**. The Modbus data are encapsulated in the form of Modbus/TCP telegrams in TCP segments of the TCP/IP protocol family. In this way, it is possible to transmit Mod bus/TCP telegrams via Ethernet.

⇨ chapter 6 "Modbus over Ethernet", Page 39

2.3 Structure of an RTU Modbus telegram

Modbus telegrams have the following structure:

Slave address	Function code	Data field	Checksum CRC
1 byte	1 byte	x bytes	2 bytes

Every telegram has four fields:

Slave address	Device address of a specific slave
Function code	Function selection (read/write words)
Data field	Contains the information (depending on the function code) - word address/bit address - number of words/number of bits - word value(s)/bit value(s)
Checksum	Detection of transmission errors

2.4 Function codes

Function overview

The functions described in the following (from the Modbus standard) are available for extracting measured values, device and process data, and for writing data.

Function code		**Function**	**Limit**
Hex	**Dec.**		
03 or 04	3 or 4	Reading n words	Max. 127 words (254 bytes)
06	6	Writing one word	Max. 1 word (2 bytes)
10	16	Writing n words	Max. 127 words (254 bytes)

NOTE!

If the device does not respond to these functions or generates an error code, these can be evaluated.

⇨ chapter 2.8 "Error messages", Page 18

2.4.1 Reading n words

This function is used to read n words, starting from a specific address.

Data request

Slave address	Function 0x03 or 0x04	Address of first word	Number of words x	Checksum CRC
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 CRC

1 byte	1 byte	1 byte	2 x bytes	2 bytes

Example

Reading the IP address of the device. The address used in the example is 10.10.1.69. Since each octet of the IP address is saved in a word, it is necessary here to read 4 words, that is, 8 bytes. For these and other Modbus addresses, see chapter 8.3.22 "External binary inputs inputs", Page 87.

Hex code for the data query:

01	03	19 C9	00 04	93 6B
Slave	Function	address 1st word	Number of words	CRC

Hex code for the response (values in byte format):

01	03	08	00 0A	00 0A	00 01	00 45	37 E5
Slave	Function	Bytes read	10	10	1	69	CRC
			IP address				

2.4.2 Writing one word

The data blocks for the instruction and response are identical when writing a word.

CAUTION!

Write operations in some R/W parameters result in them being saved in the EEPROM or flash memory. These memory modules have only a limited number of write cycles (approx. 100,000 or 10,000).

Thus, frequent writing of certain variables can result in a memory error.

▶ The number of write operations should therefore be kept as low as possible. Write operations can also be carried out using "external analog inputs". "External analog inputs" are not saved in the EEPROM or flash memory, and are suitable for fast write cycles.

Instruction

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

Response

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

Example

This example describes how to write a command for the value "Binary value of external binary input 1" of the device. In this case, the slave address of the device is 1, the word address is 0x17E2 (chapter 8.2.8 "Manual values", Page 61) and the word to write should be "1".

Hex code for the instruction:

01	06	17 E2	00 01	ED 88
Slave	function	Word address	Value	CRC

Hex code for the response:

01	06	17 E2	00 01	ED 88
Slave	function	Word address	Value	CRC

2.4.3 Writing n words

Instruction

Slave address	Function 0x10	Address of first word	Number of words x	Number of bytes 2 x	x word value(s)	Checksum CRC
1 byte	1 byte	2 bytes	2 bytes	1 byte	2 x bytes	2 bytes

Response

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

Example

Writing the word "North plant" (ASCII coding with terminating code: 0x41 0x6E 0x6C 0x61 0x67 0x65 0x20 0x4E 0x6F 0x72 0x64 0x00) from word address 0x1000 as device name.

⇨ chapter 8.2.1 "Basic settings", Page 55

Hex code for the instruction:

01	10	10 00	00 06	0C	41 6E 6C 61 67 65 20 4E 6F 72 64 00	DD 7D
Slave	Function	address 1st word	Number of words	Number of bytes	Text in UTF-8 encoding	CRC

Hex code for the response:

01	10	10 00	00 06	44 CB
Slave	Function	address 1st word	Number of words	CRC

2.5 Data types

Data type	Description	Access	Possible Function codes	Quantity Modbus register
Byte	Low byte of a word as an integer value; the high byte is not used. Value ranges: 0 to 255 for unsigned data -128 to 127 for signed data	read only	03, 04	1
		read/ write	03, 04, 06, 16	
Byte[n]	Byte array with n bytes (maximum 254 bytes) The bytes are written in a contiguous area of Modbus registers, with up to 2 bytes per register.	read only	03, 04	maximum 127
		read/ write	03, 04, 06, 16	

<table>
<tr><th>Data type</th><th>Description</th><th>Access</th><th>Possible Function codes</th><th>Quantity Modbus register</th></tr>
<tr><td>Word</td><td>Word (16-bit) as integer value

Value ranges:
0 to 65535 for unsigned data
-32768 to 32767 for signed data</td><td>read only</td><td>03, 04</td><td>1</td></tr>
<tr><td></td><td></td><td>read/ write</td><td>03, 04, 06, 16</td><td></td></tr>
<tr><td>Float</td><td>2 words as 32-bit floating-point number with coding to IEEE 754, where it must be kept in mind that bytes 1 and 2 are interchanged with bytes 3 and 4 during transmission.
S = Sign bit
E = Exponent (two's complement)
M = 23-bit normalized mantissa

IEEE 754 standard coding
<table><tr><td>Byte 1</td><td>Byte 2</td><td>Byte 3</td><td>Byte 4</td></tr><tr><td>SEEEEEEE</td><td>EMMMMMMM</td><td>MMMMMMMM</td><td>MMMMMMMM</td></tr></table>
Modbus coding of floating-point variables in the JUMO AQUIS touch S/P
<table><tr><td colspan="2">Address of the 1st Modbus register for variables</td><td colspan="2">Address of the 2nd Modbus register for variables</td></tr><tr><td>Byte 3</td><td>Byte 4</td><td>Byte 1</td><td>Byte 2</td></tr><tr><td>MMMMMMMM</td><td>MMMMMMMM</td><td>SEEEEEEE</td><td>EMMMMMMM</td></tr></table>
When creating customer-specific applications, the correct byte order in storage format needs to be checked. Many compilers use the following storage format:
Compiler coding
<table><tr><td>Byte 4</td><td>Byte 3</td><td>Byte 2</td><td>Byte 1</td></tr><tr><td>MMMMMMMM</td><td>MMMMMMMM</td><td>EMMMMMMM</td><td>SEEEEEEE</td></tr><tr><td>Address x</td><td>Address x+1</td><td>Address x+2</td><td>Address x+3</td></tr></table></td><td>read only</td><td>03, 04</td><td>2</td></tr>
<tr><td></td><td></td><td>read/ write</td><td>03, 04, 16</td><td></td></tr>
<tr><td>Uint32</td><td>Double word (32-bit) as unsigned integer value

Value range: 0 to 4.294.967.295</td><td>read only</td><td>03, 04</td><td>2</td></tr>
<tr><td></td><td></td><td>read/ write</td><td>03, 04, 16</td><td></td></tr>
</table>

Data type	Description	Access	Possible Function codes	Quantity Modbus register
Char[60]	Character string for up to 20 Unicode characters in UTF-8 coding with up to 3 bytes per character The character string thus consists of up to 60 bytes. Each of the 30 Modbus register words contains 2 successive bytes of the character string. Note that the character string must always have "\0" (ASCII code 0x00) as the terminating code.	read only	03, 04	30
		read/ write	03, 04, 06, 16	
Bool	Least significant bit in a word as bit value 0000 0000 0000 000**1** = 1 or TRUE 0000 0000 0000 000**0** = 0 or FALSE	read only	03, 04	1
		read/ write	03, 04, 06, 16	

2.6 Examples of data transmission

The function 0x03 or 0x04 (writing of n words) is used to extract integers, floating-point values and text values.

Data request

Slave address	Function 0x03 or 0x04	address First word	Number of words	Checksum CRC
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Integer values are transmitted via the Modbus in the following format:
The high byte first, followed by the low byte.

Response

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

2.6.1 Integer values

Example

In this example, the value of the maximum output level will be read out from parameter set 1 of controller channel 1 at address 0x13C1 (chapter 8.2.24 "Controller parameter maximum output level", Page 67). The value here should be "100" (word value 0x0064).

Data query:

01	03	13 C1	00 01	D1 72
Slave	function	Address of 1st word	Number of words	CRC

Response (values in Modbus floating-point format):

01	03	02	00 64	B9 AF
Slave	function	Bytes read	Integer value	CRC

2.6.2 Float values

The JUMO AQUIS touch P operates using the IEEE 754 standard format for floating-point values (32-bit), but with the difference that bytes 1 and 2 are interchanged with bytes 3 and 4.

Single float format (32-bit) acc. to Standard IEEE 754

SEEEEEEE	EMMMMMMM	MMMMMMMM	MMMMMMMM
Byte 1	Byte 2	Byte 3	Byte 4

S - Sign bit
E - Exponent (two's complement)
M - 23-bit normalized mantissa

Modbus floating-point format

Modbus address x		**Modbus address x+1**	
MMMMMMMM	MMMMMMMM	SEEEEEEE	EMMMMMMM
Byte 3	Byte 4	Byte 1	Byte 2

Example

In this example, the value "IN 7 limit alarm 1" is to be extracted at address 0x10C9 of the device. The value here is to be 550.0 (0x44098000 in the IEEE 754 format).

Data request:

01	03	10 C9	00 02	10 F5
Slave	Function	address 1st word	Number of words	CRC

Response (values in Modbus floating-point format):

01	03	04	80 00	44 09	20 F5
Slave	Function	Bytes read	Floating-point value		CRC

After being transmitted from the device, the bytes for the floating-point value must be interchanged accordingly. Many compilers (for example, Microsoft® Visual C++) store the floating-point values in the following sequence:

Floating-point value

Address x	**Address x+1**	**Address x+2**	**Address x+3**
MMMMMMMM	MMMMMMMM	EMMMMMMM	SEEEEEEE
Byte 4	Byte 3	Byte 2	Byte 1

NOTE!

The sequence of the bytes depends on how floating-point values are saved in the application concerned. It may be necessary for the bytes to be interchanged in the interface program accordingly.

2.6.3 Character strings (texts)

Character strings are coded in UTF8 for transmission.

NOTE!

A "\0" (ASCII code 0x00) must always be transmitted as the terminating code. Subsequent characters have no meaning.
Since texts are transmitted word by word (16-bit register), 0x00 is also added to the end for an uneven number of bytes (incl. "\0").
The maximum lengths for character strings specified in the address tables contain the final "\0". This means that, in the case of "char[60]", the text, including "\0", must not be longer than 60 bytes.
In the case of 19 Unicode characters having a length of 3 bytes each, only 2 bytes remain for the 20th character. 1 byte is needed for the terminating code.

Example

Querying the text from address 0x1000, this address holds the character string for the device name "North system" (ASCII code: 0x41 0x6E 0x6C 0x61 0x67 0x65 0x20 0x4E 0x6F 0x72 0x64).

Hex code for the query:

01	03	10 00	00 07	00 C8
Slave	function	Address 1st word	Number of words	CRC

Hex code for the response:

01	03	00 0E	41 6E 6C 61 67 65 20 4E 6F 72 64 00 00 AA	C5 DF
Slave	function	Bytes read	Word values (ASCII characters)	CRC

NOTE!

The value (here: AA) before the CRC sum (here: C5DF) is not considered, since it follows the terminating code "\0".

2.7 Checksum (CRC16)

Calculation principle

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

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

NOTE!

The low byte of the checksum is transmitted first!
Example: The CRC16 checksum CC DD is transmitted and represented in the sequence DD CC.

Example

Query the binary value of the wash timer 1 at address 0x14E5:

Instruction: Read one word from address 0x14E5

01	03	14 E5	00 01	90 0D
Slave	Function	address	Read one word	CRC

Response (CRC16 = 0x8479)

01	03	02	00 01	79 84
Slave	Function	Number of bytes	Word 1	CRC

Word 1 = 1 means that the binary value of wash timer 1 = 1.

2 Modbus protocol description

2.8 Error messages

2.8.1 Modbus error codes

Requirements for Modbus communication

The following conditions must be met for a slave to receive, process, and respond to queries:

- Baud rate and data format of master and slave must match.
- The correct slave address must be used in the query.
- Slave devices respond only after a successful checksum check of the query by the slave. Otherwise, the query is rejected by the slave.
- The instruction from the master must be complete and conform to the Modbus protocol.
- The number of words to be read must be greater than 0.

Error codes

If the data request from the master has been received by the slave without transmission errors but cannot be processed, the slave responds with an error code. The following error codes may occur:

- 01 = Invalid function; the function codes supported by the JUMO AQUIS touch P are listed in chapter 2.4 "Function codes", Page 10.
- 02 = Invalid address or too many words or bits are to be read or written
- 03 = Value is outside the admissible range
- 08 = Value is write-protected

Response to malfunction

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

The function code is OR-gated with 0x80. As a result, the highest-value bit (msb) is set to 1.

Example

Data query:

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

Response (with error code 2):

01	86	02	C3 A1
Slave	OR function	Errors	CRC

Response with error code 02, because the address 0x2345 does not exist.

2.8.2 Error messages for invalid values

For measured values in floating-point format, the error number itself is displayed as a value, i.e. instead of the measured value, the error number is returned.

Error code for floating-point values	Error
1.0×10^{37}	Measuring range underflow
2.0×10^{37}	Measuring range overflow
3.0×10^{37}	No valid input value
4.0×10^{37}	Division by zero
5.0×10^{37}	Mathematical error
6.0×10^{37}	Invalid compensation temperature
7.0×10^{37}	Probe short circuit
8.0×10^{37}	Probe break
9.0×10^{37}	Timeout

Example

Reading in the external analog input 1 at Modbus address 0x17B2:

Data request:

08	03	17 B2	00 02	61 01
Slave	Function	Word address	Number of words	CRC

Response:

08	03	04	8E 52	7D B4	C8 ED
Slave	Function	Bytes read	Error code		CRC

The measured value 0x7DB48E52 (=3.0×10^{37}) received from analog input 1 indicates that this is an invalid input value.

2.8.3 Error codes as integer feedback values

For some longer processes (for example, sending an email or active transmission of frames as Modbus master), an error code is entered in a result field or event list at the end. The error codes can be queried at the Modbus addresses starting at 0x19CD.

⇨ chapter 8.3.25 "Modbus error", Page 88

Error codes for faults on the JUMO digiLine bus can be found from address 0x1B46.

⇨ chapter "Last error code", Page 99

Error codes

Error code	Description
Error list: Program memory management	
1	Program cannot be created
2	Program does not exist
3	Program cannot be deleted
4	Section cannot be deleted
5	Checksum cannot be stored
6	Checksum cannot be read
7	Program cannot be copied
8	Section cannot be copied

Error code	Description
9	Program checksum error
10	Program pointer tab. checksum error
11	Program memory end
12	Section does not exist
13	Repeat jump marks
Error list: General input and output	
14	Please use the ENTER key to confirm
15	Invalid number of digits
16	The entry contains invalid characters
17	Value outside of limits
18	Section not programmed correctly
19	Password error
Error list: Keypad and program lock	
26	Keypad is locked
27	Programming is locked
28	Write error in the ser. EEprom (Calib)
29	Hardware error: MANUAL + AUTO locked
30	Editing during active program inadmissible
31	Copying during active program inadmissible
32	MANUAL is inadmissible during AUTO lead time
33	Section change! Image update needed
34	No DB number, image update from PLC
35	No DB number for process values from PLC
36	Printer busy or not ready
37	Setpoint value 1 not programmed
38	Set up printer (config. / interface)
39	Possible only when device in MANUAL mode
40	Autotuning already running
41	Time axis elapsed or not programmed
42	Time axis cannot be copied
43	Time axis does not exist
44	Program change is locked
45	MANUAL operation is locked
46	Program start is locked
Error list: Interface processing	
47	Incorrect response length
48	Timeout error (no response)
49	Error reported in telegram protocol
50	Checksum error
51	Parity error
52	Framing error
53	Interface buffer full
54	Addressing error (e.g. address does not exist)
55	Incorrect or unexpected command

Error code	Description
Error list: Event processing	
60	Event could not be created
61	Event setting failed
62	Event clear failed
63	Event wait failed
64	Event close failed
65	Event open failed
66	Sync error between group and data manager
Error list: Message processing	
70	No queue memory available
71	Message queue cannot be opened
72	Message pool cannot be created
73	Memory cannot be requested from memory pool
74	Message cannot be sent
Error list: Processing of MQX functions	
80	Task creation failed
81	Hardware timer not created
Error list: Flash memory processing	
90	Flash memory write error
Error list: Other errors	
100	Undefined error
101	Division by zero
102	Cannot find RAM
103	RTC runtime exceeded
104	ID does not exist
105	Index too large (overflow)
106	Invalid data
107	Invalid parameter
109	String without 0 characters
110	Timeout during initialization
111	Value must not be written to
112	Log entry with error bits that trigger debug mode
Error list: Sending email via modem and Ethernet	
120	Step error in state automation
121	Invalid response length
122	No CONNECT from modem
123	FCS checksum incorrect
124	Unexpected value or response
125	Conf request not accepted
126	No conf request from opposite end
127	No CHAP request from opposite end
128	Response timeout
129	Unknown modem response
130	Unexpected OK from modem
131	Unexpected CONNECT from modem

Error code	Description
132	Unknown frame received
133	Unexpected PROTOCOL from modem
134	Unexpected COMPRESS from modem
135	Invalid PPP packet received
136	Unexpected BUSY from modem
137	Unknown authentication protocol
138	Ignored LCP option
139	Unexpected DELAYED from modem
140	Unexpected NODIALTONE
141	Unknown PPP protocol
142	Unknown PAP code
143	Ignored IPCP option
144	Ignored IPCP code
145	Unknown CHAP code
146	IP checksum incorrect
147	Unknown IP protocol
148	Unknown ICMP type
149	Unknown LCP type
150	Received as client DNS request
151	Unknown DNS error
152	DNS response is split
153	No IP received via DNS
154	Unknown UDP port
155	TCP checksum incorrect
156	Incorrect TCP port
157	Unknown TCP-SYN option
158	TCP port not used
159	Unknown POP3 response
160	Unknown SMTP response
161	Unknown DNS name
162	No MD5 requested for CHAP
163	Authentication error
164	Canceled from opposite end
165	Error creating TCP socket
166	Error binding TCP socket
167	Error during TCP connect
168	Error sending TCP telegram
169	Error closing TCP socket
170	Error during TCP listen
171	Reset during TCP accept
172	Error during TCP accept
173	SMTP server reports syntax error
174	TCP socket is already closed
175	Incorrect frame configuration
176	Already sent from the gateway

Error code	Description
Error list: Processing digital sensors	
180	Invalid device address
181	Invalid dig. HW address
182	Invalid sensor type
183	Invalid sensor subtype
184	Invalid sensor VdN no.
185	Invalid sensor software version
186	Invalid sensor TAG no.
187	Invalid calibration routine
188	Invalid calibration step
189	Calibration not permitted
190	Sensor reports error
191	Collision in device address range 10 to 19 (reserved address range for JUMO ecoLine O-DO)
192	Collision in device address range 40 to 49 (reserved address range for JUMO ecoLine NTU)
193	Calibration value outside valid limits
194	Calibration signal has been reset (see chapter "Calibration signals", Page 93)
195	Collision in device address ranges 20 to 39 or 50 to 89 (reserved address range for membrane-covered sensors from JUMO with product group numbers 20263x)
Error list: File system processing	
200	Error installing partition manager
201	Error installing MFS file system
202	Error uninstalling partition manager
203	Error uninstalling MFS file system
300	Parameter is not a config. type
301	Buffer too small
302	Function ID not transmitted
303	The COE transmission is possible only for individual instances
304	Parameter is not an "Art_P" type
305	Function index in system IO not transmitted via COE
Error list: File systems	
400	General errors in the JUMO file system
401	General errors in the boot loader file system

2 Modbus protocol description

3.1 Location of interfaces

The JUMO AQUIS touch P has a RS485 interface (COM1 on the base unit as standard). This is designed for transmission using the Modbus protocol or for connecting digital sensors from JUMO.

Optionally, an Ethernet interface, PROFINET-IO interface, RS422/485 or PROFIBUS DP interface can be retrofitted in the COM2 expansion slot. The following optional boards are available:

- Serial interface RS422/485 for Modbus RTU or JUMO digiLine (part no. 00581172)
- PROFIBUS DP (part no. 00581173)
- Ethernet (part no. 00581174)

Both serial interfaces can be operated using the Modbus Slave (RTU) and Modbus Master (RTU) protocols. Alternatively, one of the two interfaces can be operated as a JUMO digiLine Master.

NOTE!

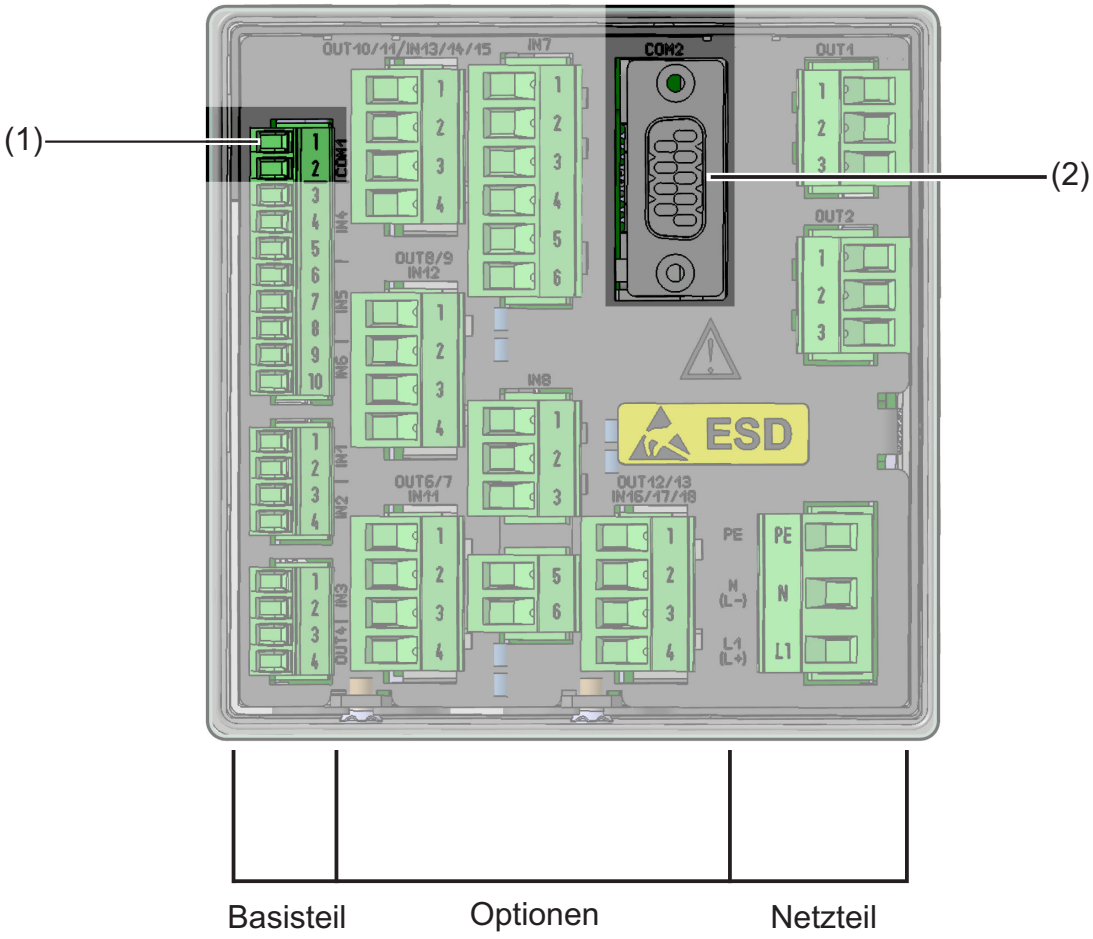
The type designation on the nameplate of the device indicates the optional interfaces that were assembled **by default**.
The relevant information can be found in the chapter entitled “Identifying device version” in the operating manual or installation instructions of the device (the installation instructions are supplied with the device).

NOTE!

Optional interfaces can also be added **by the user**.
The relevant information can be found in the chapter entitled “Retrofitting optional boards” in the operating manual or installation instructions of the device (the installation instructions are supplied with the device).

3 Interfaces

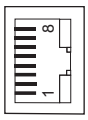
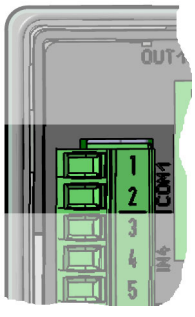
Overview of connections

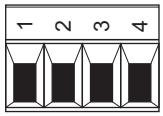


1 Base unit COM 1 for RS485 serial interfaces (Modbus RTU)

2 COM 2 expansion slot for Ethernet, PROFINET-IO, RS422/485 serial interfaces (Modbus RTU) or PROFIBUS-DP (the figure shows PROFIBUS-DP fitted)

3.2 Interface assignment

Connector/ terminal	Connec- tion variant	Terminal assignment		
LAN	Ethernet	1 TX+ 2 TX- 3 RX+ 6 RX-	Transmission data + Transmission data - Received data + Received data -	
COM 1	Serial inter- face (RS485)	1 RxD/TxD+ 2 RxD/TxD-	Transmission/received data + Transmission/received data -	

Connector/ terminal	Connec- tion variant	Terminal assignment		
COM 2	Serial inter- face (RS422)	1 RxD+ 2 RxD- 3 TxD+ 4 TxD-	Received data + Received data - Transmission data + Transmission data -	
	Serial inter- face (RS485)	3 RxD/TxD+ 4 RxD/TxD-	Transmission/received data + Transmission/received data -	

NOTE!

A twisted connecting cable with shielding must be used to connect the RS422/485 interface.

NOTE!

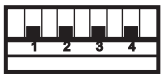
An RJ45 patch/crossover cable (CAT5 or higher) must be used to connect the LAN interface.

3.2.1 Terminating resistors

Optional boards for serial interfaces RS422/485 have integrated terminating resistors. The terminating resistors can be activated and deactivated via the DIP switch next to the connection port on the optional board. For the RS485 interface "COM 1" on the base unit, terminating resistors must be installed by the customer.

Correct installation/configuration of the terminating resistors at the ends of the bus line must be assured before commissioning an RS422/485 interface.

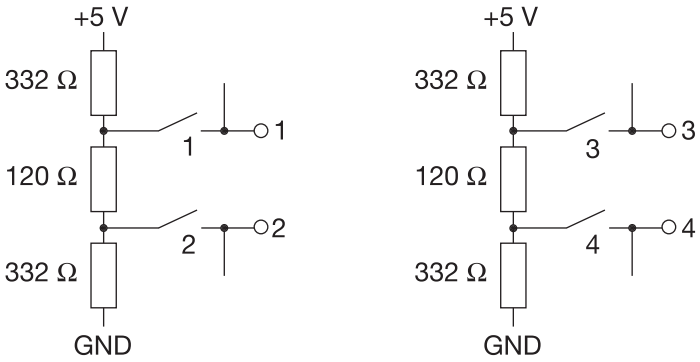
The following table shows the positions of the DIP switches on the RS422/485 optional board for activating and deactivating the terminating resistors.

RS422/485 **without** terminating resistors (deactivated)	
RS422/485 **with** terminating resistors (activated)	

NOTE!

Trouble-free operation requires terminating resistors at the beginning and end of an RS422/485 transmission path.

Terminating resistors for optional boards



Notes



CAUTION!

After every configuration change, the device restarts functions affected by the changes.

Analog and digital inputs may assume undesired states during the startup process.

▶ For this reason, configuration changes must never be made while a system is operating.



CAUTION!

Improper installation or the wrong settings on the device can result in unexpected operating states of a plant.

This can disrupt processes from functioning properly or result in damage.

▶ For this reason, it is always necessary to provide safety devices that are independent of the device and to allow settings to be made only by qualified personnel.



NOTE!

When changing configuration data that are relevant for the data monitoring and recording function, data recording is terminated and a new recording session begun.



NOTE!

Changes to the configuration settings described in this chapter can be made directly on the device or via the JUMO PC setup program.



NOTE!

Changing settings in the "Configuration" menu is possible only if a user is logged in with the appropriate user rights.

Operation, configuration and parameterization procedures are explained in detail in the operating manual for the JUMO AQUIS touch P. Information about user login is also provided there.

4 Configuring interfaces

Settings for serial interfaces

Using the “Modbus Frames” function, the device can be configured to exchange user-defined combinations of Modbus addresses with other devices. For more details, refer to the chapter on Modbus Frames.

Open: Device menu > Configuration > Serial interface >Serial interface 1 to 2

Configuration item	Selection/settings	Description
Protocol	Modbus Slave Modbus Master Modbus digital sensors	Communication protocol **Modbus Slave:** For operation of the device as a slave in a Modbus system **Modbus master:** For operation of the device as a master in a Modbus system, for example for querying measured values from sensors with a Modbus slave function. **Modbus digital sensors:** For operation of JUMO digital sensors on the serial interface (see order details for extra code "JUMO digiLine protocol activated") In the JUMO AQUIS touch P, either the interface on the base unit or the optional serial interface (if present) can be configured for digital sensors (digiLine operation). Simultaneous use of both interfaces for digiLine operation is not possible.
Baud rate	9600 19200 38400	Transmission speed (symbol rate) of the serial interface[a] When connecting digital sensors of product groups 2026xx, the baud rate must be set to 9600 baud prior to startup. Otherwise, the sensors will not begin operation.
Data format	8 - 1 - no parity 8 - 1 - odd parity 8 - 1 - even parity	Format of the data word[a] Useful bit - stop bit - parity
Minimum response time	0 to 500 ms	**For Modbus slave:** Minimum time from receiving a request to sending the response **For Modbus master:** Minimum time from receiving a response to sending the next request. This parameter can be used to optimize bus timing.

Configuration item	Selection/settings	Description
Modbus slave		
Timeout master	60 to 60000 ms	Master monitoring time After this time, a failure of the Modbus master is identified. An internal digital signal is set in the event of a timeout.
Device address	1 to 254	A bus user's unique identifier 0 = Broadcast address[b] 1 to 247 = Unicast addresses[c] 248 to 254 reserved addresses[d]
Modbus master		
Timeout	60 to 10000 ms	A request sent by the master is defined as faulty if no answer is received within this time.
Scan cycle	60 to 99999 ms	The Modbus master requests data from the Modbus slaves at these intervals.

[a] For all users to communicate with one another, these setting be the same on all users.

[b] Device addressing is specified in the Modbus standard. The broadcast address must not be used as a slave address. It is intended for broadcast messages.

[c] Unicast addresses are intended for use as slave addresses. They are used to identify the slave devices uniquely so that the master can communicate with them explicitly.

[d] In the standard Modbus, the address range 248 to 254 is reserved for future use. In the device, these addresses can be used as slave addresses.

Settings for Ethernet interface

The following protocols can be used for communication over Ethernet:

- Modbus/TCP slave for communication with Modbus masters over Ethernet
- Modbus/TCP master for communication with Modbus masters over Ethernet
- Communication with the PC setup program via the HTTP protocol
- E-mail sent using the SMTP protocol

DHCP and DNS are also supported. It is possible to obtain the IP configuration automatically via DHCP. If necessary, the IP configuration can also be set up manually.

Using the “Modbus Frames” function, the device can be configured to exchange user-defined combinations of Modbus addresses with other devices. For more details, refer to the chapter on Modbus Frames.

Open: Device Menu > Configuration > Ethernet

Configuration item	Selection/settings	Description
Assign IP address	Manually Automatically	**Manually:** If no DHCP server is installed on the network and the IP address configuration is known (e.g. by the network administrator), the data are entered manually. **Automatically:** A DHCP server is installed on the network. After being switched on and booted , the JUMO AQUIS touch P receives the IP configuration from the DHCP server; IP configuration takes place automatically.
Manual IP address	Valid IP address[a]	IP address of the device to use for Modbus/TCP (Modbus communication via Ethernet)

Configuration item	Selection/settings	Description
Subnet mask	Valid subnet mask[a]	The subnet mask specifies which part of the IP address functions as the network address and which part as the host address. It can be used to create subnetworks on a network.
Standard gateway	Valid IP address[a]	IP address of the standard gateway The standard gateway is required for routing into other networks. Without the standard gateway, the JUMO AQUIS touch P can communicate only with hosts in its own subnetwork.
DNS server	Valid IP address[a]	IP address of the DNS server DNS =**D**omain **N**ame **S**ystem This service permits the DNS name resolution. The JUMO AQUIS touch S requires DNS to resolve the IP address of the outgoing email server based on its server name.
Transfer rate	Automatic 10 Mbit/s half duplex 10 Mbit/s full duplex 100 Mbit/s half duplex 100 Mbit/s full duplex	Bit rate and Duplex mode of the Ethernet optional board This setting must match the setting of the switch or router port via which the JUMO AQUIS touch P is connected.

[a] To enter an IP configuration manually, a valid available IP address on the network must be known. Contact your network administrator to obtain the parameters for manual IP configuration.

Modbus TCP

The settings for the Modbus-TCP operating mode are implemented in a separate configuration menu.

If the device functions as a Modbus slave, two external devices (masters 1 and 2) can simultaneously access the paperless recorder. If it functions as a Modbus master, it can communicate with up to four external devices (Modbus slaves: device 1 to 4).

Open: Device Menu > Configuration > Modbus TCP

Configuration item	Selection/settings	Description
Port	0 to 1024	TCP port of the type 202580 as a Modbus slave. The change of the TCP port for the slave function must be made with the device manufacturer's set-up program. In the **delivery state**, the **TCP port number "502"** is configured.
Modbus master		
Timeout	4000 to 10000 ms	A request sent by the master is defined as faulty if no answer is received within this time.
Scan cycle	60 to 99999 ms	The Modbus master requests data from the Modbus slaves at these intervals.
External devices (device 1 to 4)[a]		
Manual IP address	Valid IP address[b]	IP address of the Modbus slave device
Port	0 to 1024	TCP port of Modbus slave device

[a] Settings for up to 4 target slave devices to which the type 202580 is to send its Modbus commands are created for the Modbus master function of the type 202580 via Ethernet. It can be selected as the target device for the respective Modbus frame in the settings of the Modbus frames.

[b] To enter an IP configuration for type 202580 manually, a valid available IP address on the network must be known. Please contact your network administrator in this regard.

4 Configuring interfaces

In the basic type, the JUMO AQUIS touch P has one RS485 interface (COM 1 on the base unit as standard). In addition, a RS422/485 interface can be retrofitted in slot COM2 as an optional board.

These interfaces can be used to connect the device to up to 2 bus systems, each as either master or slave.

On each bus, all slave devices must have unique device addresses in the range of 1 to 254. The master is not addressed.

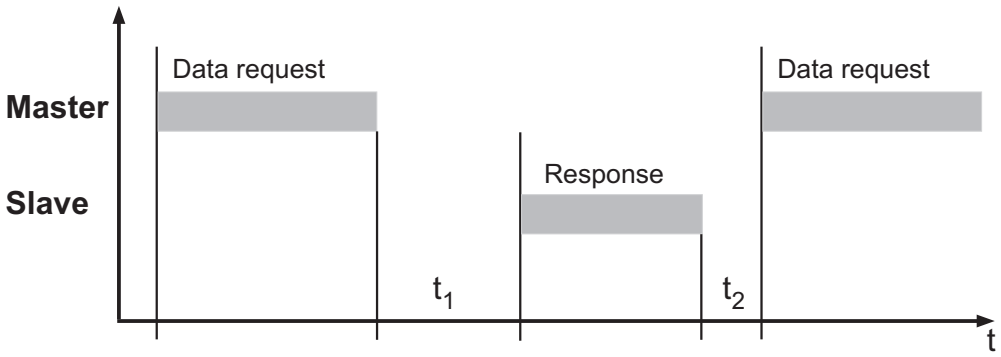
5.1 Modbus slave communication over an RS422/485 serial interface

If the device is configured as a slave, it responds on the bus to Modbus requests from the master. The master controls the data exchange and the slaves only have a response function. They are identified by their device address.
The master can request all device variables according to the Modbus address tables of this slave.

⇨ chapter 8 "Modbus address tables", Page 55

Chronological sequence of communication

A scanning cycle on a bus proceeds with the following timing:



t_1	Internal waiting period of the device before checking the data query and the internal processing time: min.: 5 ms typically: 5 to 35 ms max.: 35 ms or the set "minimum response time"
t_2	Waiting period that the master must observe before it starts a new data query with RS485: 35 ms

NOTE!

The minimum response time can be set in the configuration.
Device menu > Configuration > Serial interface > Serial interface 1 to 2
This set time must elapse before a response is sent (0 to 500 ms). If processing of a query from a master is completed in the slave before the minimum response time has elapsed, the response is not sent until the "minimum response time" has elapsed.
With an RS485 interface, the minimum response time that can be set is needed by the master to switch the interface driver from send to receive.

NOTE!

Within t_1 and t_2 and during the response time of the slave, no data queries may be generated by the master. Queries during t_1 and t_2 are ignored by the slave. Requests during the response time invalidate all the data currently on the bus.

5 Modbus over a serial interface

NOTE!

The terminating code after a data query or data response is 3 characters long. The time needed for these 3 characters depends on the baud rate.

Character transmission time

The beginning and end of a data block are identified by pauses in transmission. The character transmission time (time to transmit one character) depends on the baud rate and the data format used.
The following results for a data format of one start bit, 8 data bits, no parity bit, and one stop bit:

Character transmission time [ms] = 1000 × 10 bits ÷ baud rate

For other data formats, the following is the result:

Character transmission time [ms] = 1000 × 11 bits ÷ baud rate

Example

Identifier for end of data request or response for data format 11/10 bits

Waiting period = 3 characters * 1000 * 10 bits ÷ baud rate

Baud rate [baud]	**Data format** [bit]	**Character transmission time**[ms]
38400	11	0.268
	10	0.260
19200	11	0.573
	10	0.521
9600	11	1.146
	10	1.042

Timeoutmonitoring

This function monitors the communication between the Modbus master and Modbus slave from the perspective of the Modbus slave. When the first request is received, a timer starts running (the timer time is the time set in the configuration under “Timeout”). The timer starts to run again after each new request. If the request does not appear, an internal binary signal is activated after the timer has elapsed, and its status can be queried via Modbus (see chapter 8.3.25 "Modbus error", Page 88). The signal is reset with the next request.

5.2 Modbus master operation via serial interface

If device type has been configured as a master, it can send queries on the bus to slaves. The frames transmitted here are always complete. The corresponding Modbus and device addresses must be specified when configuring the frames in the setup program. For each frame, it is possible to enter the device address and Modbus address used to query the value. Each programmed frame can be disabled by selecting the "Modbus slave" setting for the "Interface" parameter.

A maximum of 8 frames can be enabled for reading and a maximum of 8 frames for writing. The activated frames are cyclically processed in sequence. This also applies to writing frames regardless of whether a process value has changed or not. An event-driven transmission dependent on process value changes is therefore not possible. The complete frame is always transmitted with the frame length configured in the setup program.

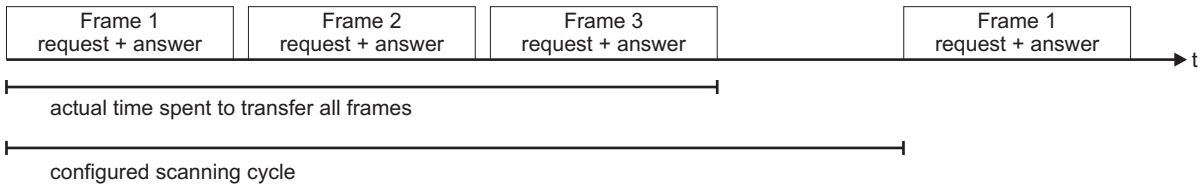


NOTE!

Double writing of a target variable, for example, the same variable selector in two read frames, will lead to undefined states and must be avoided.

Chronological sequence

The relevant interface (setup program: ***ONLY SETUP > MODBUS FRAMES FOR READING/WRITING*** > Parameter ***INTERFACE***) searches for all frames configured for it and cyclically transmits them in sequence, to frame 1, frame 2, and frame 3, such as in the example below.



If the configured scanning cycle is greater than the elapsed actual time required for scanning, the device waits to process the next cycle and thus minimizes the bus load. If the configured scanning cycle is too short, all configured frames are completely processed cyclically without a break.

NOTE!

With the parameter “minimum response time” in the serial interface configuration, the device’s waiting time as master from receiving a response until the next transmission can be adjusted to the slaves’ reaction times, allowing them sufficient time for processing.

6.1 Modbus/TCP

Modbus/TCP uses Ethernet as a transmission standard. Two transmission options can be used here:

- Modbus/TCP slave for transmitting individual values
- Modbus/TCP master for transmitting entire data frames

The advantage of using Modbus/TCP and the Ethernet interface is in the high speed and the company-wide availability of the connected devices.

Modbus/TCP is a standardized process, in which a Modbus telegram is packed in a TCP frame (tunneled) and transmitted via Ethernet. The Modbus telegram (without CRC) is transmitted with an additional 6 or 7 byte "MBAP header" (Modbus Application Header). The seventh byte corresponds to the first serial byte, but has a different designation here.

Structure of a Modbus/TCP telegram

MBAP header				**Modbus telegram**
2 bytes Transaction ID	2 bytes Protocol ID	2 bytes Length	1 byte Unit ID	Additional bytes as below, but without CRC
Identical in request and response	Must be 0 for Modbus	Length of request/response in bytes starting with (incl.) "Unit ID"	Corresponds to device address and must be TCP 0xFF or 0 (0 = broadcast)	

For comparison: The "normal" Modbus telegram

Slave address 1 byte	Function code 1 byte	Data field x bytes	CRC16 2 bytes

With Modbus/TCP, for example, a suitable process data visualization program can act as the master over a company-wide Ethernet network to read and write values from the device configured as a slave. All device variables from the Modbus address tables of the device can be accessed.

⇨ chapter 8 "Modbus address tables", Page 55

Example: Reading n words

Reading the IP address of the device. The address used in the example is 10.10.1.69. Since each octet of the IP address is saved in a word, it is necessary here to read 4 words, that is, 8 bytes.

See also the Modbus example in chapter 2.4.1 "Reading n words", Page 10.

Query:

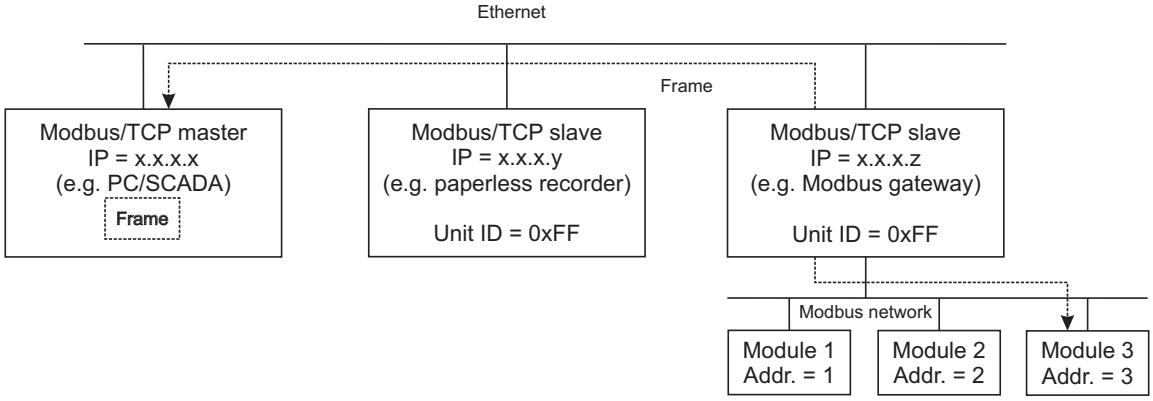
MBAP header				**Modbus telegram (without slave address and CRC)**		
0x0001	0x0000	0x0006	0xFF	0x03	0x19C9	0x0004
2 bytes Transaction ID	2 bytes Protocol ID	2 bytes Length	1 byte Unit ID	1 byte Function code	2 bytes address First word	2 bytes Number of words
Assignment of response to request (consecutive numbering)	For Modbus always 0x0000	Length of the request in bytes starting with (incl.) "Unit ID"; 6 bytes here (0x06)	For TCP always FF or 255	Function code for "Reading n words"	First word of IP address to be read	4 words should be read

Response:

MBAP header				**Modbus telegram (without slave address and CRC)**					
0x0001	0x0000	0x000B	0xFF	0x03	0x08	0x000A	0x000A	0x0001	0x0045
2 bytes Transaction ID	2 bytes Protocol ID	2 bytes Length	1 byte Unit ID	1 byte Function code	1 byte Number of bytes read	8 bytes 1 word per octet 4 octets × 2 bytes = 8 bytes			
Assignment of response to query	For Modbus always 0x00	Length of response in bytes (incl.) "Unit ID"; in this case 11 bytes	For TCP always 0xFF	Function code for "Reading n words"	8 bytes were read	IP address consisting of 4 octets 10.	10.	1.	69

6.2 Networking with Modbus/TCP

The image below provides an overview of the networking options when using the Modbus/TCP protocol:



Each Modbus/TCP node is accessible via a unique IP address. The "Unit ID" also contained in the protocol (= device address) must be 0xFF for the node directly attached to the network. However, a node can present itself virtually as several subordinate devices. For example, this may be the case with a Modbus gateway with the connected modules. These modules can then be accessed with device addresses 1 to 254.

6.3 Modbus master with Modbus/TCP

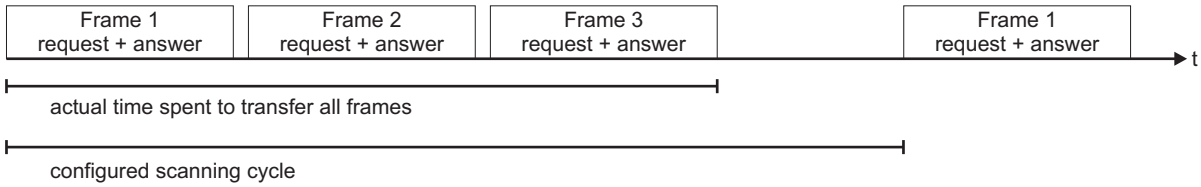
Up to 4 connections are possible to different Modbus/TCP devices which allow the exchange of a maximum of 8 reading and 8 writing frames. A frame can be used to transmit up to 254 bytes. It operates just like the Modbus master function via the serial interface.

Frames with 4 opposing ends (devices 1 to 4) can be exchanged using 4 sockets. The Modbus device address that can be edited inside each frame mask and that is transmitted in the Modbus/TCP protocol can also be used to address modules behind Modbus gateways. The frame from Modbus gateway can thus be forwarded to the module.

⇨ chapter 4 "Configuring interfaces", Page 29

Chronological sequence

The relevant interface (setup program: ***Only setup*** > ***Modbus frames for reading/writing*** > Parameter ***Interface***) searches for all frames configured for it and cyclically transmits them in sequence, to frame 1, frame 2, and frame 3, such as in the example below.



If the configured scanning cycle is longer than the elapsed actual time required for scanning, the device waits to process the next cycle and thus minimizes the bus load. If the configured scanning cycle is too short, all configured frames are completely processed cyclically without a break.

6.4 Modbus slave with Modbus/TCP

Here, the device functions as a slave and is available for requests on the bus from the Modbus master. A master can query all device variables according to the Modbus address tables for this slave.

⇨ chapter 8 "Modbus address tables", Page 55

6 Modbus over Ethernet

NOTE!

Only two Modbus masters (clients) can access this slave (server) simultaneously via Modbus/TCP. A connection opened by a master is closed by the slave after 30 seconds of inactivity. If a Modbus/TCP port is closed (by a slave or from the opposite end), it can only be reopened after 10 seconds have elapsed.

The TCP port no. is preset to the value 502_{DEC}. This value can be edited.

Timeoutmonitoring

This function monitors the communication between the Modbus master and Modbus slave from the perspective of the Modbus slave. When the first request is received, a timer starts running (the timer time is the time set in the configuration under “Timeout”). The timer starts to run again after each new request. If the request does not appear, an internal binary signal is activated after the timer has elapsed, and its status can be queried via Modbus (see chapter 8.3.25 "Modbus error", Page 88). The signal is reset with the next request.

7.1 General information

The device allows the user to individually compile Modbus frames for his application. As a result, the highest level of flexibility is reached and the data exchange on the bus is reduced.

This offers a considerable advantage with regard to the transfer speed. Numerous variables can be compiled in a large data packet, thus improving the transmission speed, which, in turn, means that the bus is subject to less protocol overhead and fewer changes between the request and response.

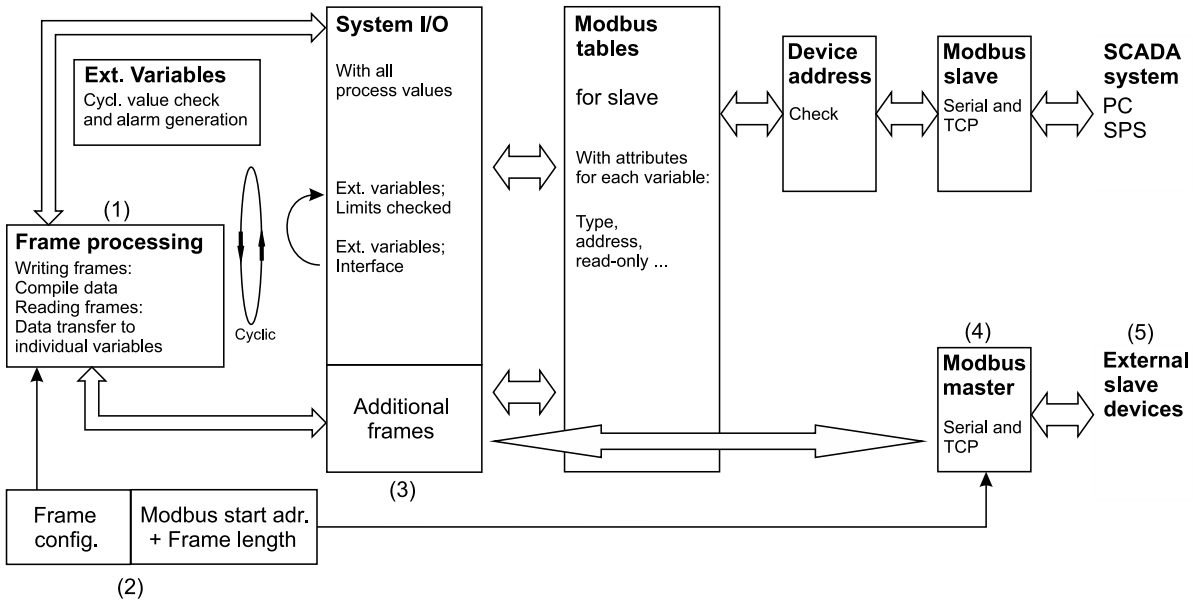
The maximum size of a frame is set to 254 bytes. This corresponds to 127 words for each command. The activated reading or writing frames are cyclically processed in sequence. The frame length specified in the setup program is always used. This length is automatically calculated using the number of entries (up to 16 per frame) in the respective frame. All frames can be used for the Modbus master or Modbus slave.

NOTE!

There is no locking function against multiple use of the same input variables within several reading frames. The user must prevent this from happening to avoid a data collision.

7.2 Structure of the reading and writing processes

Block diagram of the inputs and outputs for the master function



Reading/writing Modbus (master)

(1)	A cyclical function compiles the individual variables into frames.
(2)	To do this, it accesses the frame arrangement configured in the setup program.
(3)	The complete cyclically updated frames are available.
(4)	Each interface configured as a master transmits (reading or writing) the frames configured for it to the opposite end.
(5)	The opposite end receives these frames and responds.

7 Modbus frames

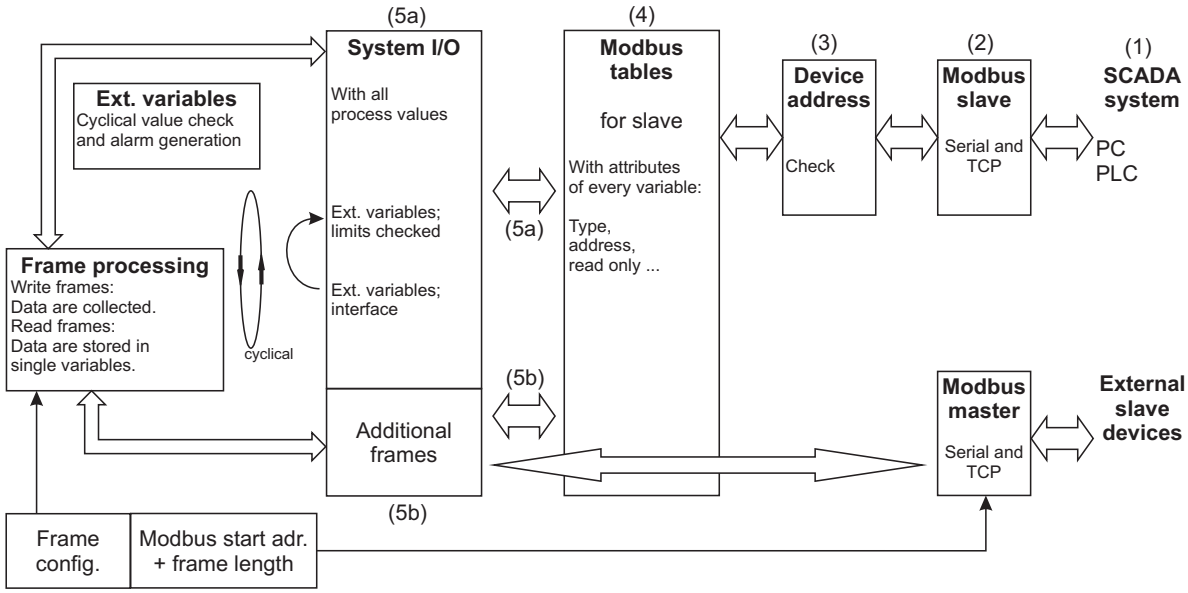
NOTE!

When actively transmitting as a Modbus master, repetitive errors are entered in the event list. Corresponding error codes can be found in chapter 2.8.3 "Error codes as integer feedback values", Page 19.

Error monitoring

Each frame is monitored for data transmission. If an error occurs, the corresponding error flag is set, an error code is also stored for each frame, and an entry is written in the event list. This entry is not cyclically generated and set; instead, it is generated and set once when the error occurs and then again when the transmission functions again. The following applies for all read frames:
If there is no valid response after 3 recurrences, all values for this frame are set to "No input value".

Block diagram of the inputs and outputs for the slave function



Reading/writing Modbus (slave)

(1)	An external master issues a request (reading or writing).
(2)	The interface receives the request.
(3)	The device address is checked for validity.
(4)	The Modbus table assigns the Modbus addresses to the device variables.
(5a)	The variable value is accessed and the Modbus response is generated.
(5b)	The frames configured for the Modbus master can also be accessed for checking or variable scaling/type conversions.

7.3 Compiling Modbus frames

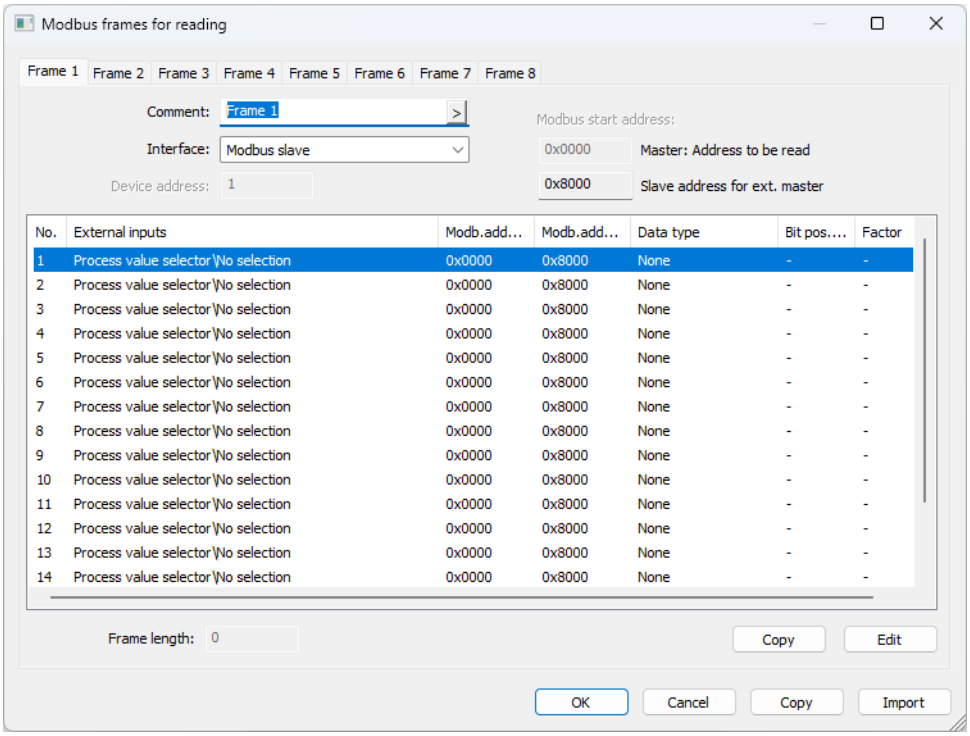
The functions "Modbus frames for reading" and "Modbus frames for writing" are in the setup program in the navigation tree under ***Only setup***.

7.3.1 Modbus frames for reading

This function is used to compile up to 8 Modbus frames for reading process values from external devices (via interface) individually for each opposite end. The process values (analog and digital values) are written to the selected variables (external inputs) from the received Modbus telegram and are available for use in the device.

Each frame can be used to configure up to 16 frame entries with a process value each, which are then grouped and transmitted in a Modbus telegram.

Setup dialog box



CAUTION!

A variable must not be used in multiple frames.

This means that different process values are written to the same variable.

▶ The user has to make sure that a variable (external input) is not used in multiple frames. In particular, after copying frames or frame entries, the frames must be edited to ensure that duplicate use of external inputs is excluded.

7 Modbus frames

Parameters

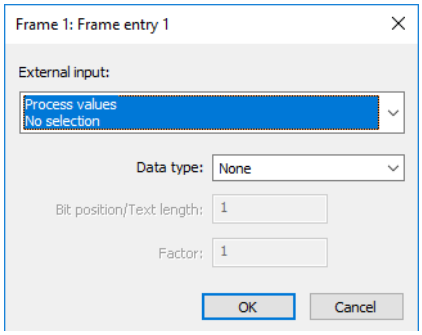
Parameters	Selection/settings	Description
Comment	**Frame 01** Use default text or edit text.	Comment on the more detailed description of the frame
Interface	The selection determines whether the frame is actively transmitted as a master or only available for queries as a slave. In the case of a Modbus master, the interface on which the relevant frame is used is also specified. If it is an Ethernet interface, the external device to be addressed must also be selected.	
	Modbus slave	Compiled frame only available for queries as a slave
	Modbus master TCP 1	Modbus master; Modbus/TCP via Ethernet, device 1 addressed
	Modbus master TCP 2	Modbus master; Modbus/TCP via Ethernet, device 2 addressed
	Modbus master TCP 3	Modbus master; Modbus/TCP via Ethernet, device 3 addressed
	Modbus master TCP 4	Modbus master; Modbus/TCP via Ethernet, device 4 addressed
	Modbus master serial 1	Modbus master; Modbus RTU via serial interface COM1 (base unit)
	Modbus master serial 2	Modbus master; Modbus RTU via serial interface COM2 (optional board)
Device address	**1** to 255	Device address of the external device (Modbus slave) For Modbus/TCP, to query a direct slave via Modbus/TCP, address 255 must be set. The address range 1 to 254 can be used for forwarding to slaves via a Modbus gateway (see chapter 6.2 "Networking with Modbus/TCP", Page 41).
Modbus start adress master (The device is a Modbus master)	**0x0000** to 0xFFFF	Modbus start address (offset) of the external slave device The setup program uses the start address as an initial value for the addresses of the entries. The Modbus description of the external device must be observed for this.
Modbus start adress slave (The device is a Modbus slave)	**0x8000**	Display of the Modbus start address (offset) of this frame in the device The start address is used for the setting in the external Modbus master.
Entry 1 to entry 16	Select the desired entry (double-click the line with the entry or mark and then click the "Edit" button).	
Frame length	**0** to 254	Displays the frame length (byte(s))

Copy (entries)

With this function entries within a Modbus frame can be copied.

Editing

When the "Edit" button is clicked, this window appears:



Parameters

Parameters	Selection/settings	Description
External input	External process values (analog inputs, digital inputs, texts)	
	No selection	No process value selected
	Select process value	Selector for selecting a process value
Data type	The data type is preset depending on the type of external input. It can subsequently be changed.	
	None	No data type selected
	Float (AB CD) – (LSB)	Floating point number, IEEE 754 Big Endian; old designation: Float (LSB)
	Float (CD AB) – (MSB)	Floating point number, Modbus standard coding; old designation: Float (MSB)
	Float (BA DC)	Floating point number, IEEE 754 Little Endian Byte Swap
	Float (DC BA)	Floating point number, IEEE 754 Little Endian
	Integer (1 byte)	Integer with length of 1 byte
	Integer (2 bytes)	Integer with length of 2 bytes
	Integer (4 bytes)	Integer with length of 4 bytes
	Unsigned int. (1 byte)	Integer without prefix sign, with length of 1 byte
	Unsigned int. (2 bytes)	Integer without prefix sign, with length of 2 bytes
	Unsigned int. (4 bytes)	Integer without prefix sign, with length of 4 bytes
Bit pos./text length	Integer (1 byte): **0** to 7 Integer (2 bytes): **0** to 15 Integer (4 bytes): **0** to 31	The setting ranges of the bit position or text length depend on the data type selected. (This parameter is inactive with an external analog input.)
Factor	Using the factor makes it possible to transmit floating-point values in the integer format, for example. The transmitter must multiply the data with the corresponding factor before transmission. The data must be divided by the same value in the receiver.	
	Complete floating-point range allowed; default value = 1.0	This factor is used to rescale values during the transmission, in particular for simultaneous type conversions. (This parameter is not active with an external digital input or text.)

Copy (frames)

With this function Modbus frames within a project can be copied.

Import

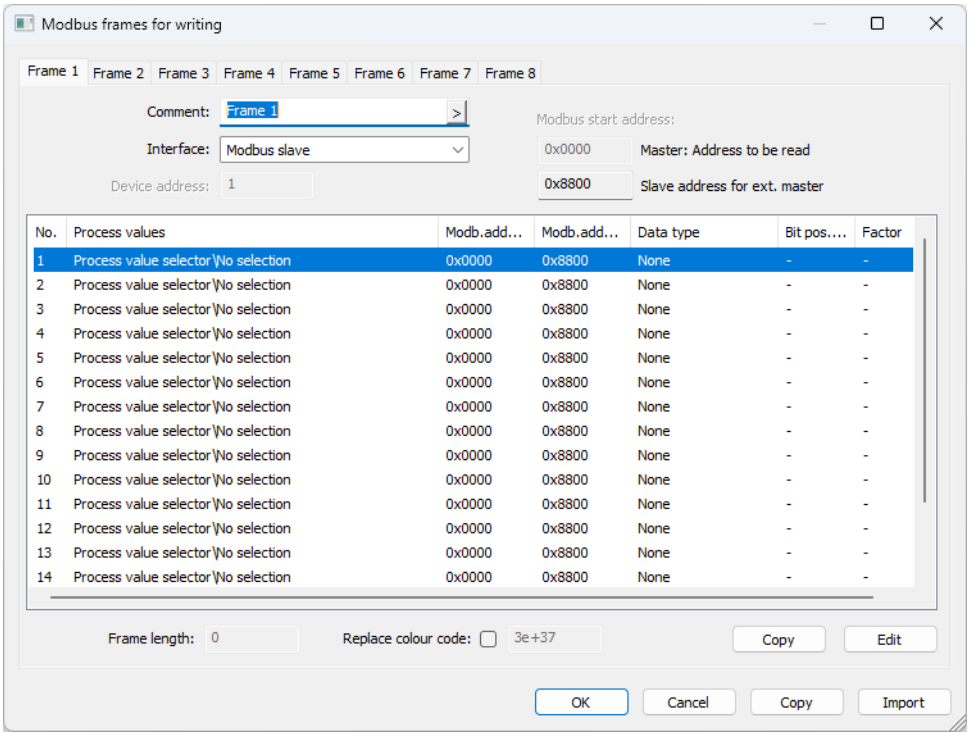
With this function Modbus frames can be imported from another project. First, the setup file from which you want to import is temporarily opened. Then all frames to be copied have to be selected (individual frames or all). As a result, all entries of the relevant frames are transferred into your own project.

7.3.2 Modbus frames for writing

This function can be used to compile up to 8 Modbus frames for writing the device's process values to external devices (via interface) individually for each opposite end. The process values (analog and digital values) are written into the frames by this device and are available for external devices.

Each frame can be used to configure up to 16 frame entries with a process value each, which are then grouped and transmitted in a Modbus telegram.

Setup dialog box



Parameters

Parameters	Selection/settings	Description
Comment	**Frame 1** Use default text or edit text.	Comment on the more detailed description of the frame
Interface	The selection determines whether the frame is actively transmitted as a master or only available for queries as a slave. In the case of a Modbus master, the interface on which the relevant frame is used is also specified. If it is an Ethernet interface, the external device to be addressed must also be selected.	
	Modbus slave	Compiled frame only available for queries as a slave
	Modbus master TCP 1	Modbus master; Modbus/TCP via Ethernet, device 1 addressed
	Modbus master TCP 2	Modbus master; Modbus/TCP via Ethernet, device 2 addressed
	Modbus master TCP 3	Modbus master; Modbus/TCP via Ethernet, device 3 addressed
	Modbus master TCP 4	Modbus master; Modbus/TCP via Ethernet, device 4 addressed
	Modbus master serial 1	Modbus master; Modbus RTU via serial interface COM1 (base unit)
	Modbus master serial 2	Modbus master; Modbus RTU via serial interface COM2 (optional board)
Device address	0 to **1** to 255	Device address of the external device (Modbus slave) For Modbus/TCP, to query a direct slave via Modbus/TCP, address 255 must be set. The address range 1 to 254 can be used for forwarding to slaves via a Modbus gateway (see chapter 6.2 "Networking with Modbus/TCP", Page 41).
Modbus start adress master (The device is a Modbus master)	**0x0000** to 0xFFFF	Modbus start address (offset) of the external slave device The setup program uses the start address as an initial value for the addresses of the entries. The Modbus description of the external device must be observed for this.
Modbus start adress slave (The device is a Modbus slave)	**0x8800**	Display of the Modbus start address (offset) of this frame in the device The start address is used for the setting in the external Modbus master.
Entry 1 to entry 16	Select the desired entry (double-click the line with the entry or mark and then click the "Edit" button).	
Frame length	**0** to 254	Displays the frame length (byte(s))

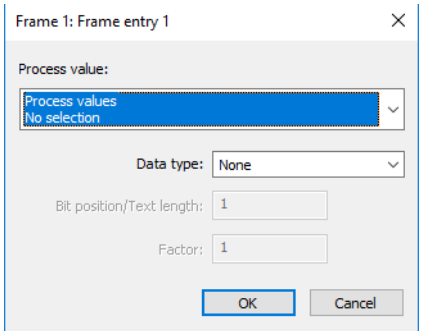
Parameters	Selection/settings	Description
Error code replace	-3.37E+38 to **+3.0E+37** to +3.37E+38	Value (code) that is transmitted instead of the measured value if an error occurs (for measured values in the floating-point format). By activating the error code, the user can specify an error value for all malfunctions. If “Replace error code” is not active, the device’s differentiating standard error values apply (see chapter 2.8.2 "Error messages for invalid values", Page 19).

Copy (entries)

With this function entries within a Modbus frame can be copied.

Editing

When the "Edit" button is clicked, this window appears:



Parameters

Parameters	Selection/settings	Description
Process value	Process values of the device (analog signals, digital signals, texts)	
	No selection	No process value selected
	Select process value	Selector for selecting a process value
Data type	The data type is preset depending on the type of process value. It can subsequently be changed.	
	None	No data type selected
	Float (AB CD) – (LSB)	Floating point number, IEEE 754 Big Endian; old designation: Float (LSB)
	Float (CD AB) – (MSB)	Floating point number, Modbus standard coding; old designation: Float (MSB)
	Float (BA DC)	Floating point number, IEEE 754 Little Endian Byte Swap
	Float (DC BA)	Floating point number, IEEE 754 Little Endian
	Integer (1 byte)	Integer with length of 1 byte
	Integer (2 bytes)	Integer with length of 2 bytes
	Integer (4 bytes)	Integer with length of 4 bytes
	Unsigned int. (1 byte)	Integer without prefix sign, with length of 1 byte
	Unsigned int. (2 bytes)	Integer without prefix sign, with length of 2 bytes
	Unsigned int. (4 bytes)	Integer without prefix sign, with length of 4 bytes
Bit pos./text length	Integer (1 byte): **0** to 7 Integer (2 bytes): **0** to 15 Integer (4 bytes): **0** to 31	The setting ranges of the bit position or text length depend on the data type selected. (This parameter is inactive with an analog signal.)
Factor	Using the factor makes it possible to transmit floating-point values in the integer format, for example. The transmitter must multiply the data with the corresponding factor before transmission. The data must be divided by the same value in the receiver.	
	Complete floating-point range allowed; default value = 1.0	This factor is used to rescale values during the transmission, in particular for simultaneous type conversions. (This parameter is not active with a digital signal or text.)

7 Modbus frames

Copy (frames)

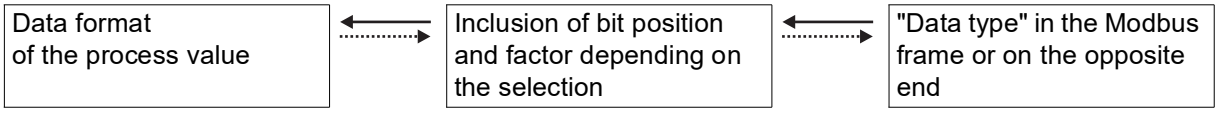
With this function Modbus frames within a project can be copied.

Import

With this function Modbus frames can be imported from another project. First, the setup file from which you want to import is temporarily opened. Then all frames to be copied have to be selected (individual frames or all). As a result, all entries of the relevant frames are transferred into your own project.

7.4 Examples for the data transmission options with frames

There are flexible customization options between the content of the variables in the device and the data format on the opposite end. The process value selected from the selector determines the data format within this device; the selection under "Data type" determines the data format in the Modbus frame or on the opposite end. The two data formats do not have to match. As a result, type conversion can be carried out. The logical direction is displayed in the following diagram:



NOTE!

For reading frames, the logical direction is the arrow to the left and for writing frames, the arrow to the right.

The admissible setting options are:

Floating-point value	⟷	x factor	⟷	Floating-point value
Floating-point value	⟷	x factor	⟷	Integer value
Integer value	⟷	x factor	⟷	Floating-point value
Integer value	⟷	x factor	⟷	Integer value
Binary/Boolean value	⟷	Bit position	⟷	Integer value[a]
Binary/Boolean value	⟷	Bit position	⟷	"None"[b]

[a] Only the configured bit (bit position within the integer value) is used.

[b] Other bits from the previous frame entry is used with the data type "Integer..." In each frame entry with the data type "None" the bit position must be specified. This determines at which point the bit of the relevant frame entry is transmitted within the previous integer value.

7 Modbus frames

The tables in this chapter list all process and device data for the JUMO AQUIS touch P with their Modbus address, the data type, and the possible access options (Modbus function codes).

8.1 Modbus master Actual cycle times

Modbus PDU address		Data type	Access	Data
Hex.	**Dec.**			
01B0	432	Uint32	r/o	Actual cycle time in 5-ms ticks serial interface 1[a] (see chapter 5.2 "Modbus master operation via serial interface", Page 37)
01B9	441	Uint32	r/o	Actual cycle time in 5-ms ticks serial interface 2[a] (see chapter 5.2 "Modbus master operation via serial interface", Page 37)
01C6	454	Uint32	r/o	Actual cycle time in 5-ms ticks interface Modbus TCP 1[a] (see chapter 6.3 "Modbus master with Modbus/TCP", Page 41)
01D9	473	Uint32	r/o	Actual cycle time in 5-ms ticks interface Modbus TCP 2[a] (see chapter 6.3 "Modbus master with Modbus/TCP", Page 41)
01EC	492	Uint32	r/o	Actual cycle time in 5-ms ticks interface Modbus TCP 3[a] (see chapter 6.3 "Modbus master with Modbus/TCP", Page 41)
01FF	511	Uint32	r/o	Actual cycle time in 5-ms ticks interface Modbus TCP 4[a] (see chapter 6.3 "Modbus master with Modbus/TCP", Page 41)

[a] The actual cycle times of Modbus master operation are available only from device software version 304.08.01 onward.

8.2 Configuration data and parameters

8.2.1 Basic settings

Modbus PDU address		Data type	Access	Data
Hex.	**Dec.**			
1000	4096	Char[60]	r/o	The 1st and 2nd bytes of the character string consisting of Unicode characters with UTF 8 coding
...	...			...
101D	4125			Device name 59th and 60th bytes of the character string consisting of Unicode characters with UTF 8 coding
101E	4126	byte	r/w	Temperature unit device operation 0 = °C 1 = °F
101F	4127	byte	r/w	Temperature unit interface 0 = °C 1 = °F

8.2.2 Temperature inputs

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1021	4129	float	r/w	IN 4 limit value alarm 1
1023	4131	float	r/w	IN 4 limit value alarm 2
1025	4133	float	r/w	IN 5 limit value alarm 1
1027	4135	float	r/w	IN 5 limit value alarm 2
1029	4137	float	r/w	IN 4 screen width alarm 1
102B	4139	float	r/w	IN 4 screen width alarm 2
102D	4141	float	r/w	IN 5 screen width alarm 1
102F	4143	float	r/w	IN 5 screen width alarm 2

[a] The temperature unit for the values corresponds to the configured temperature unit for the interfaces of the device, and is specified in the "Basic settings".

8.2.3 Universal inputs

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1031	4145	float	r/w	IN 6 limit value alarm 1
1033	4147	float	r/w	IN 6 limit value alarm 2
1035	4149	float	r/w	IN 11 limit value alarm 1
1037	4151	float	r/w	IN 11 limit value alarm 2
1039	4153	float	r/w	IN 12 limit value alarm 1
103B	4155	float	r/w	IN 12 limit value alarm 2
103D	4157	float	r/w	IN 6 screen width alarm 1
103F	4159	float	r/w	IN 6 screen width alarm 2
1041	4161	float	r/w	IN 11 screen width alarm 1
1043	4163	float	r/w	IN 11 screen width alarm 2
1045	4165	float	r/w	IN 12 screen width alarm 1
1047	4167	float	r/w	IN 12 screen width alarm 2

[a] The unit for the values is based on the configuration for the respective analysis input.

8.2.4 Analysis inputs pH/Redox/NH

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
10C9	4297	float	r/w	IN 7 limit value alarm 1
10D1	4305	float	r/w	IN 7 limit value alarm 2
10D9	4313	float	r/w	IN 8 limit value alarm 1
10E1	4321	float	r/w	IN 8 limit value alarm 2
1109	4361	float	r/w	IN 7 screen width alarm 1
1111	4369	float	r/w	IN 7 screen width alarm 2
1119	4377	float	r/w	IN 8 screen width alarm 1
1121	4385	float	r/w	IN 8 screen width alarm 2

[a] The unit for the values is based on the configuration for the respective analysis input.

8.2.5 Analysis inputs CR (conductive conductivity)

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1149	4425	float	r/w	IN 7 limit value measuring range 1 alarm 1
114B	4427	float	r/w	IN 7 limit value measuring range 2 alarm 1
114D	4429	float	r/w	IN 7 limit value measuring range 3 alarm 1
114F	4431	float	r/w	IN 7 limit value measuring range 4 alarm 1
1151	4433	float	r/w	IN 7 limit value measuring range 1 alarm 2
1153	4435	float	r/w	IN 7 limit value measuring range 2 alarm 2
1155	4437	float	r/w	IN 7 limit value measuring range 3 alarm 2
1157	4439	float	r/w	IN 7 limit value measuring range 4 alarm 2
1159	4441	float	r/w	IN 8 limit value measuring range 1 alarm 1
115B	4443	float	r/w	IN 8 limit value measuring range 2 alarm 1
115D	4445	float	r/w	IN 8 limit value measuring range 3 alarm 1
115F	4447	float	r/w	IN 8 limit value measuring range 4 alarm 1
1161	4449	float	r/w	IN 8 limit value measuring range 1 alarm 2
1163	4451	float	r/w	IN 8 limit value measuring range 2 alarm 2
1165	4453	float	r/w	IN 8 limit value measuring range 3 alarm 2
1167	4455	float	r/w	IN 8 limit value measuring range 4 alarm 2
1189	4489	float	r/w	IN 7 screen width measuring range 1 alarm 1
118B	4491	float	r/w	IN 7 screen width measuring range 2 alarm 1
118D	4493	float	r/w	IN 7 screen width measuring range 3 alarm 1
118F	4495	float	r/w	IN 7 screen width measuring range 4 alarm 1
1191	4497	float	r/w	IN 7 screen width measuring range 1 alarm 2
1193	4499	float	r/w	IN 7 screen width measuring range 2 alarm 2
1195	4501	float	r/w	IN 7 screen width measuring range 3 alarm 2
1197	4503	float	r/w	IN 7 screen width measuring range 4 alarm 2
1199	4505	float	r/w	IN 8 screen width measuring range 1 alarm 1
119B	4507	float	r/w	IN 8 screen width measuring range 2 alarm 1
119D	4509	float	r/w	IN 8 screen width measuring range 3 alarm 1
119F	4511	float	r/w	IN 8 screen width measuring range 4 alarm 1
11A1	4513	float	r/w	IN 8 screen width measuring range 1 alarm 2
11A3	4515	float	r/w	IN 8 screen width measuring range 2 alarm 2
11A5	4517	float	r/w	IN 8 screen width measuring range 3 alarm 2
11A7	4519	float	r/w	IN 8 screen width measuring range 4 alarm 2

[a] The unit for the values is based on the configuration for the respective analysis input.

8.2.6 Analysis inputs Ci (inductive conductivity)

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
11C9	4553	float	r/w	IN 7 limit value measuring range 1 alarm 1
11CB	4555	float	r/w	IN 7 limit value measuring range 2 alarm 1
11CD	4557	float	r/w	IN 7 limit value measuring range 3 alarm 1
11CF	4559	float	r/w	IN 7 limit value measuring range 4 alarm 1
11D1	4561	float	r/w	IN 7 limit value measuring range 1 alarm 2
11D3	4563	float	r/w	IN 7 limit value measuring range 2 alarm 2
11D5	4565	float	r/w	IN 7 limit value measuring range 3 alarm 2
11D7	4567	float	r/w	IN 7 limit value measuring range 4 alarm 2
11D9	4569	float	r/w	IN 8 limit value measuring range 1 alarm 1
11DB	4571	float	r/w	IN 8 limit value measuring range 2 alarm 1
11DD	4573	float	r/w	IN 8 limit value measuring range 3 alarm 1
11DF	4575	float	r/w	IN 8 limit value measuring range 4 alarm 1
11E1	4577	float	r/w	IN 8 limit value measuring range 1 alarm 2
11E3	4579	float	r/w	IN 8 limit value measuring range 2 alarm 2
11E5	4581	float	r/w	IN 8 limit value measuring range 3 alarm 2
11E7	4583	float	r/w	IN 8 limit value measuring range 4 alarm 2
1209	4617	float	r/w	IN 7 screen width measuring range 1 alarm 1
120B	4619	float	r/w	IN 7 screen width measuring range 2 alarm 1
120D	4621	float	r/w	IN 7 screen width measuring range 3 alarm 1
120F	4623	float	r/w	IN 7 screen width measuring range 4 alarm 1
1211	4625	float	r/w	IN 7 screen width measuring range 1 alarm 2
1213	4627	float	r/w	IN 7 screen width measuring range 2 alarm 2
1215	4629	float	r/w	IN 7 screen width measuring range 3 alarm 2
1217	4631	float	r/w	IN 7 screen width measuring range 4 alarm 2
1219	4633	float	r/w	IN 8 screen width measuring range 1 alarm 1
121B	4635	float	r/w	IN 8 screen width measuring range 2 alarm 1
121D	4637	float	r/w	IN 8 screen width measuring range 3 alarm 1
121F	4639	float	r/w	IN 8 screen width measuring range 4 alarm 1
1221	4641	float	r/w	IN 8 screen width measuring range 1 alarm 2
1223	4643	float	r/w	IN 8 screen width measuring range 2 alarm 2
1225	4645	float	r/w	IN 8 screen width measuring range 3 alarm 2
1227	4647	float	r/w	IN 8 screen width measuring range 4 alarm 2

[a] The unit for the values is based on the configuration for the respective analysis input.

8.2.7 External analog inputs

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1251	4689	float	r/w	External analog input 1 limit value alarm 1
1253	4691	float	r/w	External analog input 1 limit value alarm 2
1255	4693	float	r/w	External analog input 2 limit value alarm 1
1257	4695	float	r/w	External analog input 2 limit value alarm 2
1259	4697	float	r/w	External analog input 3 limit value alarm 1
125B	4699	float	r/w	External analog input 3 limit value alarm 2
125D	4701	float	r/w	External analog input 4 limit value alarm 1
125F	4703	float	r/w	External analog input 4 limit value alarm 2
1261	4705	float	r/w	External analog input 5 limit value alarm 1
1263	4707	float	r/w	External analog input 5 limit value alarm 2
1265	4709	float	r/w	External analog input 6 limit value alarm 1
1267	4711	float	r/w	External analog input 6 limit value alarm 2
1269	4713	float	r/w	External analog input 7 limit value alarm 1
126B	4715	float	r/w	External analog input 7 limit value alarm 2
126D	4717	float	r/w	External analog input 8 limit value alarm 1
126F	4719	float	r/w	External analog input 8 limit value alarm 2
1C83	7299	float	r/w	External analog input 9 limit value alarm 1[b]
1C85	7301	float	r/w	External analog input 9 limit value alarm 2[b]
1C87	7303	float	r/w	External analog input 10 limit value alarm 1[b]
1C89	7305	float	r/w	External analog input 10 limit value alarm 2[b]
1C8B	7307	float	r/w	External analog input 11 limit value alarm 1[b]
1C8D	7309	float	r/w	External analog input 11 limit value alarm 2[b]
1C8F	7311	float	r/w	External analog input 12 limit value alarm 1[b]
1C91	7313	float	r/w	External analog input 12 limit value alarm 2[b]
1C93	7315	float	r/w	External analog input 13 limit value alarm 1[b]
1C95	7317	float	r/w	External analog input 13 limit value alarm 2[b]
1C97	7319	float	r/w	External analog input 14 limit value alarm 1[b]
1C99	7321	float	r/w	External analog input 14 limit value alarm 2[b]
1C9B	7323	float	r/w	External analog input 15 limit value alarm 1[b]
1C9D	7325	float	r/w	External analog input 15 limit value alarm 2[b]
1C9F	7327	float	r/w	External analog input 16 limit value alarm 1[b]
1CA1	7329	float	r/w	External analog input 16 limit value alarm 2[b]

[a] The unit for the values is based on the configuration for the respective external analog input.

[b] The limit values of external analog inputs 9 to 16 are only available starting from device software version 304.08.01.

8 Modbus address tables

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1271	4721	float	r/w	External analog input 1 screen width alarm 1
1273	4723	float	r/w	External analog input 1 screen width alarm 2
1275	4725	float	r/w	External analog input 2 window range alarm 1
1277	4727	float	r/w	External analog input 2 window range alarm 2
1279	4729	float	r/w	External analog input 3 screen width alarm 1
127B	4731	float	r/w	External analog input 3 screen width alarm 2
127D	4733	float	r/w	External analog input 4 screen width alarm 1
127F	4735	float	r/w	External analog input 4 screen width alarm 2
1281	4737	float	r/w	External analog input 5 screen width alarm 1
1283	4739	float	r/w	External analog input 5 screen width alarm 2
1285	4741	float	r/w	External analog input 6 screen width alarm 1
1287	4743	float	r/w	External analog input 6 screen width alarm 2
1289	4745	float	r/w	External analog input 7 screen width alarm 1
128B	4747	float	r/w	External analog input 7 screen width alarm 2
128D	4749	float	r/w	External analog input 8 screen width alarm 1
128F	4751	float	r/w	External analog input 8 screen width alarm 2
1CA3	7331	float	r/w	External analog input 9 screen width alarm 1[b]
1CA5	7333	float	r/w	External analog input 9 screen width alarm 2
1CA7	7335	float	r/w	External analog input 10 screen width alarm 1
1CA9	7337	float	r/w	External analog input 10 screen width alarm 2
1CAB	7339	float	r/w	External analog input 11 screen width alarm 1
1CAD	7341	float	r/w	External analog input 11 screen width alarm 2
1CAF	7343	float	r/w	External analog input 12 screen width alarm 1[b]
1CB1	7345	float	r/w	External analog input 12 screen width alarm 2[b]
1CB3	7347	float	r/w	External analog input 13 screen width alarm 1[b]
1CB5	7349	float	r/w	External analog input 13 screen width alarm 2[b]
1CB7	7351	float	r/w	External analog input 14 screen width alarm 1[b]
1CB9	7353	float	r/w	External analog input 14 screen width alarm 2[b]
1CBB	7355	float	r/w	External analog input 15 screen width alarm 1[b]
1CBD	7357	float	r/w	External analog input 15 screen width alarm 2[b]
1CBF	7359	float	r/w	External analog input 16 screen width alarm 1[b]
1CC1	7361	float	r/w	External analog input 16 screen width alarm 2[b]

[a] The unit for the values is based on the configuration for the respective external analog input.

[b] The screen widths of external analog inputs 9 to 16 are only available starting from device software version 304.08.01.

8.2.8 Manual values



CAUTION!

Write operations in some R/W parameters result in them being saved in the EEPROM or flash memory. These memory modules have only a limited number of write cycles (approx. 10 000 or 100 000).

Frequent writing of variables can thus result in a memory error in the event of a power failure.

▶ Fast write cycles should thus be avoided or performed with the aid of the "external analog inputs". "External analog inputs" are not saved in the EEPROM or flash memory, and are suitable for fast write cycles.

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1291	4753	float	r/w	Manual value 1
1293	4755	float	r/w	Manual value 2
1295	4757	float	r/w	Manual value 3
1297	4759	float	r/w	Manual value 4
1299	4761	float	r/w	Manual value 5
129B	4763	float	r/w	Manual value 6
129D	4765	float	r/w	Manual value 7
129F	4767	float	r/w	Manual value 8
12A1	4769	float	r/w	Manual value 9
12A3	4771	float	r/w	Manual value 10
12A5	4773	float	r/w	Manual value 11
12A7	4775	float	r/w	Manual value 12
12A9	4777	float	r/w	Manual value 13
12AB	4779	float	r/w	Manual value 14
12AD	4781	float	r/w	Manual value 15
12AF	4783	float	r/w	Manual value 16

[a] The unit for the values is based on the configuration for the respective manual value.

8.2.9 Flow

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
12B1	4785	float	r/w	Flow rate 1 limit value alarm 1
12B3	4787	float	r/w	Flow rate 1 limit value alarm 2
12B5	4789	float	r/w	Flow rate 2 limit value alarm 1
12B7	4791	float	r/w	Flow rate 2 limit value alarm 2
12B9	4793	float	r/w	Flow rate 1 screen width alarm 1
12BB	4795	float	r/w	Flow rate 1 screen width alarm 2
12BD	4797	float	r/w	Flow rate 2 screen width alarm 1
12BF	4799	float	r/w	Flow rate 2 screen width alarm 2

[a] The unit for the values is based on the configuration for the respective flow rate function.

8.2.10 Limit value monitoring

8 Modbus address tables

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
12C1	4801	float	r/w	Limit value monitoring 1 limit value alarm 1
12C3	4803	float	r/w	Limit value monitoring 1 limit value alarm 2
12C5	4805	float	r/w	Limit value monitoring 2 limit value alarm 1
12C7	4807	float	r/w	Limit value monitoring 2 limit value alarm 2
12C9	4809	float	r/w	Limit value monitoring 3 limit value alarm 1
12CB	4811	float	r/w	Limit value monitoring 3 limit value alarm 2
12CD	4813	float	r/w	Limit value monitoring 4 limit value alarm 1
12CF	4815	float	r/w	Limit value monitoring 4 limit value alarm 2
12D1	4817	float	r/w	Limit value monitoring 5 limit value alarm 1
12D3	4819	float	r/w	Limit value monitoring 5 limit value alarm 2
12D5	4821	float	r/w	Limit value monitoring 6 limit value alarm 1
12D7	4823	float	r/w	Limit value monitoring 6 limit value alarm 2
12D9	4825	float	r/w	Limit value monitoring 7 limit value alarm 1
12DB	4827	float	r/w	Limit value monitoring 7 limit value alarm 2
12DD	4829	float	r/w	Limit value monitoring 8 limit value alarm 1
12DF	4831	float	r/w	Limit value monitoring 8 limit value alarm 2
12E1	4833	float	r/w	Limit value monitoring 1 screen width alarm 1
12E3	4835	float	r/w	Limit value monitoring 1 screen width alarm 2
12E5	4837	float	r/w	Limit value monitoring 2 screen width alarm 1
12E7	4839	float	r/w	Limit value monitoring 2 screen width alarm 2
12E9	4841	float	r/w	Limit value monitoring 3 screen width alarm 1
12EB	4843	float	r/w	Limit value monitoring 3 screen width alarm 2
12ED	4845	float	r/w	Limit value monitoring 4 screen width alarm 1
12EF	4847	float	r/w	Limit value monitoring 4 screen width alarm 2
12F1	4849	float	r/w	Limit value monitoring 5 screen width alarm 1
12F3	4851	float	r/w	Limit value monitoring 5 screen width alarm 2
12F5	4853	float	r/w	Limit value monitoring 6 screen width alarm 1
12F7	4855	float	r/w	Limit value monitoring 6 screen width alarm 2
12F9	4857	float	r/w	Limit value monitoring 7 screen width alarm 1
12FB	4859	float	r/w	Limit value monitoring 7 screen width alarm 2
12FD	4861	float	r/w	Limit value monitoring 8 screen width alarm 1
12FF	4863	float	r/w	Limit value monitoring 8 screen width alarm 2

[a] The unit for the values is based on the configuration for the respective limit value monitoring.

8.2.11 Controller parameter proportional band 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1301	4865	float	r/w	Controller 1 parameter block 1
1303	4867	float	r/w	Controller 1 parameter block 2
1305	4869	float	r/w	Controller 2 parameter block 1
1307	4871	float	r/w	Controller 2 parameter block 2
1309	4873	float	r/w	Controller 3 parameter block 1
130B	4875	float	r/w	Controller 3 parameter block 2
130D	4877	float	r/w	Controller 4 parameter block 1
130F	4879	float	r/w	Controller 4 parameter block 2

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

8.2.12 Controller parameter proportional band 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1311	4881	float	r/w	Controller 1 parameter block 1
1313	4883	float	r/w	Controller 1 parameter block 2
1315	4885	float	r/w	Controller 2 parameter block 1
1317	4887	float	r/w	Controller 2 parameter block 2
1319	4889	float	r/w	Controller 3 parameter block 1
131B	4891	float	r/w	Controller 3 parameter block 2
131D	4893	float	r/w	Controller 4 parameter block 1
131F	4895	float	r/w	Controller 4 parameter block 2

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

8.2.13 Controller parameter derivative time 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1321	4897	float	r/w	Controller 1 parameter block 1
1323	4899	float	r/w	Controller 1 parameter block 2
1325	4901	float	r/w	Controller 2 parameter block 1
1327	4903	float	r/w	Controller 2 parameter block 2
1329	4905	float	r/w	Controller 3 parameter block 1
132B	4907	float	r/w	Controller 3 parameter block 2
132D	4909	float	r/w	Controller 4 parameter block 1
132F	4911	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.14 Controller parameter derivative time 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1331	4913	float	r/w	Controller 1 parameter block 1
1333	4915	float	r/w	Controller 1 parameter block 2
1335	4917	float	r/w	Controller 2 parameter block 1
1337	4919	float	r/w	Controller 2 parameter block 2
1339	4921	float	r/w	Controller 3 parameter block 1
133B	4923	float	r/w	Controller 3 parameter block 2
133D	4925	float	r/w	Controller 4 parameter block 1
133F	4927	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.15 Controller parameter reset time 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1341	4929	float	r/w	Controller 1 parameter block 1
1343	4931	float	r/w	Controller 1 parameter block 2
1345	4933	float	r/w	Controller 2 parameter block 1
1347	4935	float	r/w	Controller 2 parameter block 2
1349	4937	float	r/w	Controller 3 parameter block 1
134B	4939	float	r/w	Controller 3 parameter block 2
134D	4941	float	r/w	Controller 4 parameter block 1
134F	4943	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.16 Controller parameter reset time 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1351	4945	float	r/w	Controller 1 parameter block 1
1353	4947	float	r/w	Controller 1 parameter block 2
1355	4949	float	r/w	Controller 2 parameter block 1
1357	4951	float	r/w	Controller 2 parameter block 2
1359	4953	float	r/w	Controller 3 parameter block 1
135B	4955	float	r/w	Controller 3 parameter block 2
135D	4957	float	r/w	Controller 4 parameter block 1
135F	4959	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.17 Controller parameter switching period 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1361	4961	float	r/w	Controller 1 parameter block 1
1363	4963	float	r/w	Controller 1 parameter block 2
1365	4965	float	r/w	Controller 2 parameter block 1
1367	4967	float	r/w	Controller 2 parameter block 2
1369	4969	float	r/w	Controller 3 parameter block 1
136B	4971	float	r/w	Controller 3 parameter block 2
136D	4973	float	r/w	Controller 4 parameter block 1
136F	4975	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.18 Controller parameter switching period 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1371	4977	float	r/w	Controller 1 parameter block 1
1373	4979	float	r/w	Controller 1 parameter block 2
1375	4981	float	r/w	Controller 2 parameter block 1
1377	4983	float	r/w	Controller 2 parameter block 2
1379	4985	float	r/w	Controller 3 parameter block 1
137B	4987	float	r/w	Controller 3 parameter block 2
137D	4989	float	r/w	Controller 4 parameter block 1
137F	4991	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.19 Controller parameter contact spacing

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1381	4993	float	r/w	Controller 1 parameter block 1
1383	4995	float	r/w	Controller 1 parameter block 2
1385	4997	float	r/w	Controller 2 parameter block 1
1387	4999	float	r/w	Controller 2 parameter block 2
1389	5001	float	r/w	Controller 3 parameter block 1
138B	5003	float	r/w	Controller 3 parameter block 2
138D	5005	float	r/w	Controller 4 parameter block 1
138F	5007	float	r/w	Controller 4 parameter block 2

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

8.2.20 Controller parameter switch hyst. 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1391	5009	float	r/w	Controller 1 parameter block 1
1393	5011	float	r/w	Controller 1 parameter block 2
1395	5013	float	r/w	Controller 2 parameter block 1
1397	5015	float	r/w	Controller 2 parameter block 2
1399	5017	float	r/w	Controller 3 parameter block 1
139B	5019	float	r/w	Controller 3 parameter block 2
139D	5021	float	r/w	Controller 4 parameter block 1
139F	5023	float	r/w	Controller 4 parameter block 2

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

8.2.21 Controller parameter switch hyst. 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13A1	5025	float	r/w	Controller 1 parameter block 1
13A3	5027	float	r/w	Controller 1 parameter block 2
13A5	5029	float	r/w	Controller 2 parameter block 1
13A7	5031	float	r/w	Controller 2 parameter block 2
13A9	5033	float	r/w	Controller 3 parameter block 1
13AB	5035	float	r/w	Controller 3 parameter block 2
13AD	5037	float	r/w	Controller 4 parameter block 1
13AF	5039	float	r/w	Controller 4 parameter block 2

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

8.2.22 Controller parameter actuator time

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13B1	5041	word	r/w	Controller 1 parameter block 1
13B2	5042	word	r/w	Controller 1 parameter block 2
13B3	5043	word	r/w	Controller 2 parameter block 1
13B4	5044	word	r/w	Controller 2 parameter block 2
13B5	5045	word	r/w	Controller 3 parameter block 1
13B6	5046	word	r/w	Controller 3 parameter block 2
13B7	5047	word	r/w	Controller 4 parameter block 1
13B8	5048	word	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.23 Controller parameter working point

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13B9	5049	byte	r/w	Controller 1 parameter block 1
13BA	5050	byte	r/w	Controller 1 parameter block 2
13BB	5051	byte	r/w	Controller 2 parameter block 1
13BC	5052	byte	r/w	Controller 2 parameter block 2
13BD	5053	byte	r/w	Controller 3 parameter block 1
13BE	5054	byte	r/w	Controller 3 parameter block 2
13BF	5055	byte	r/w	Controller 4 parameter block 1
13C0	5056	byte	r/w	Controller 4 parameter block 2

[a] Unit: percent

8.2.24 Controller parameter maximum output level

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13C1	5057	byte	r/w	Controller 1 parameter block 1
13C2	5058	byte	r/w	Controller 1 parameter block 2
13C3	5059	byte	r/w	Controller 2 parameter block 1
13C4	5060	byte	r/w	Controller 2 parameter block 2
13C5	5061	byte	r/w	Controller 3 parameter block 1
13C6	5062	byte	r/w	Controller 3 parameter block 2
13C7	5063	byte	r/w	Controller 4 parameter block 1
13C8	5064	byte	r/w	Controller 4 parameter block 2

[a] Unit: percent

8.2.25 Controller parameter minimum output level

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13C9	5065	byte	r/w	Controller 1 parameter block 1
13CA	5066	byte	r/w	Controller 1 parameter block 2
13CB	5067	byte	r/w	Controller 2 parameter block 1
13CC	5068	byte	r/w	Controller 2 parameter block 2
13CD	5069	byte	r/w	Controller 3 parameter block 1
13CE	5070	byte	r/w	Controller 3 parameter block 2
13CF	5071	byte	r/w	Controller 4 parameter block 1
13D0	5072	byte	r/w	Controller 4 parameter block 2

[a] Unit: percent

8.2.26 Controller parameter minimum relay energize time 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13D1	5073	float	r/w	Controller 1 parameter block 1
13D3	5075	float	r/w	Controller 1 parameter block 2
13D5	5077	float	r/w	Controller 2 parameter block 1
13D7	5079	float	r/w	Controller 2 parameter block 2
13D9	5081	float	r/w	Controller 3 parameter block 1
13DB	5083	float	r/w	Controller 3 parameter block 2
13DD	5085	float	r/w	Controller 4 parameter block 1
13DF	5087	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.27 Controller parameter minimum relay energize time 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13E1	5089	float	r/w	Controller 1 parameter block 1
13E3	5091	float	r/w	Controller 1 parameter block 2
13E5	5093	float	r/w	Controller 2 parameter block 1
13E7	5095	float	r/w	Controller 2 parameter block 2
13E9	5097	float	r/w	Controller 3 parameter block 1
13EB	5099	float	r/w	Controller 3 parameter block 2
13ED	5101	float	r/w	Controller 4 parameter block 1
13EF	5103	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.28 Controller parameter maximum pulse frequency 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13F1	5105	byte	r/w	Controller 1 parameter block 1
13F2	5106	byte	r/w	Controller 1 parameter block 2
13F3	5107	byte	r/w	Controller 2 parameter block 1
13F4	5108	byte	r/w	Controller 2 parameter block 2
13F5	5109	byte	r/w	Controller 3 parameter block 1
13F6	5110	byte	r/w	Controller 3 parameter block 2
13F7	5111	byte	r/w	Controller 4 parameter block 1
13F8	5112	byte	r/w	Controller 4 parameter block 2

[a] Unit: 1/min

8.2.29 Controller parameter maximum pulse frequency 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
13F9	5113	byte	r/w	Controller 1 parameter block 1
13FA	5114	byte	r/w	Controller 1 parameter block 2
13FB	5115	byte	r/w	Controller 2 parameter block 1
13FC	5116	byte	r/w	Controller 2 parameter block 2
13FD	5117	byte	r/w	Controller 3 parameter block 1
13FE	5118	byte	r/w	Controller 3 parameter block 2
13FF	5119	byte	r/w	Controller 4 parameter block 1
1400	5120	byte	r/w	Controller 4 parameter block 2

[a] Unit: 1/min

8.2.30 Controller parameter switch-on delay 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1401	5121	float	r/w	Controller 1 parameter block 1
1403	5123	float	r/w	Controller 1 parameter block 2
1405	5125	float	r/w	Controller 2 parameter block 1
1407	5127	float	r/w	Controller 2 parameter block 2
1409	5129	float	r/w	Controller 3 parameter block 1
140B	5131	float	r/w	Controller 3 parameter block 2
140D	5133	float	r/w	Controller 4 parameter block 1
140F	5135	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.31 Controller parameter switch-on delay 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1411	5137	float	r/w	Controller 1 parameter block 1
1413	5139	float	r/w	Controller 1 parameter block 2
1415	5141	float	r/w	Controller 2 parameter block 1
1417	5143	float	r/w	Controller 2 parameter block 2
1419	5145	float	r/w	Controller 3 parameter block 1
141B	5147	float	r/w	Controller 3 parameter block 2
141D	5149	float	r/w	Controller 4 parameter block 1
141F	5151	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.32 Controller parameter switch-off delay 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1421	5153	float	r/w	Controller 1 parameter block 1
1423	5155	float	r/w	Controller 1 parameter block 2
1425	5157	float	r/w	Controller 2 parameter block 1
1427	5159	float	r/w	Controller 2 parameter block 2
1429	5161	float	r/w	Controller 3 parameter block 1
142B	5163	float	r/w	Controller 3 parameter block 2
142D	5165	float	r/w	Controller 4 parameter block 1
142F	5167	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.33 Controller parameter switch-off delay 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1431	5169	float	r/w	Controller 1 parameter block 1
1433	5171	float	r/w	Controller 1 parameter block 2
1435	5173	float	r/w	Controller 2 parameter block 1
1437	5175	float	r/w	Controller 2 parameter block 2
1439	5177	float	r/w	Controller 3 parameter block 1
143B	5179	float	r/w	Controller 3 parameter block 2
143D	5181	float	r/w	Controller 4 parameter block 1
143F	5183	float	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.34 Controller parameter alarm tolerance

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1441	5185	float	r/w	Controller 1 parameter block 1
1443	5187	float	r/w	Controller 1 parameter block 2
1445	5189	float	r/w	Controller 2 parameter block 1
1447	5191	float	r/w	Controller 2 parameter block 2
1449	5193	float	r/w	Controller 3 parameter block 1
144B	5195	float	r/w	Controller 3 parameter block 2
144D	5197	float	r/w	Controller 4 parameter block 1
144F	5199	float	r/w	Controller 4 parameter block 2

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

8.2.35 Controller parameter alarm delay

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1461	5217	word	r/w	Controller 1 parameter block 1
1462	5218	word	r/w	Controller 1 parameter block 2
1463	5219	word	r/w	Controller 2 parameter block 1
1464	5220	word	r/w	Controller 2 parameter block 2
1465	5221	word	r/w	Controller 3 parameter block 1
1466	5222	word	r/w	Controller 3 parameter block 2
1467	5223	word	r/w	Controller 4 parameter block 1
1468	5224	word	r/w	Controller 4 parameter block 2

[a] Unit: seconds

8.2.36 Controller parameter setpoint values



CAUTION!

Write operations involving many R/W parameters are associated with saving in an EEPROM or flash memory. These memory modules have only a limited number of write cycles (approx. 10 000 or 100 000).

Frequent writing of variables can thus result in a memory error in the event of a power failure.

▶ Fast write cycles should thus be avoided or performed with the aid of the "external analog inputs". "External analog inputs" are not saved in an EEPROM or flash memory, and are ideal for fast write cycles.

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
1471	5233	float	r/w	Setpoint value 1 controller 1
1473	5235	float	r/w	Setpoint value 1 controller 2
1475	5237	float	r/w	Setpoint value 1 controller 3
1477	5239	float	r/w	Setpoint value 1 controller 4
1479	5241	float	r/w	Setpoint value 2 controller 1
147B	5243	float	r/w	Setpoint value 2 controller 2
147D	5245	float	r/w	Setpoint value 2 controller 3
147F	5247	float	r/w	Setpoint value 2 controller 4

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

8 Modbus address tables

8.3 Process values

8.3.1 Date and time

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
14A5	5285	word	r/o	Year
14A6	5286	word	r/o	Month
14A7	5287	word	r/o	Day
14A8	5288	word	r/o	Hour
14A9	5289	word	r/o	Minute
14AA	5290	word	r/o	Second

8.3.2 Limit value monitoring alarm 1

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
14BB	5307	bool	r/o	Limit value monitoring 1
14BC	5308	bool	r/o	Limit value monitoring 2
14BD	5309	bool	r/o	Limit value monitoring 3
14BE	5310	bool	r/o	Limit value monitoring 4
14BF	5311	bool	r/o	Limit value monitoring 5
14C0	5312	bool	r/o	Limit value monitoring 6
14C1	5313	bool	r/o	Limit value monitoring 7
14C2	5314	bool	r/o	Limit value monitoring 8

[a] The unit for the values is based on the configuration for the analog value source selected for the input value.

8.3.3 Limit value monitoring alarm 2

Modbus PDU address		Data type	Access	Data[a]
Hex.	Dec.			
14C3	5315	bool	r/o	Limit value monitoring 1
14C4	5316	bool	r/o	Limit value monitoring 2
14C5	5317	bool	r/o	Limit value monitoring 3
14C6	5318	bool	r/o	Limit value monitoring 4
14C7	5319	bool	r/o	Limit value monitoring 5
14C8	5320	bool	r/o	Limit value monitoring 6
14C9	5321	bool	r/o	Limit value monitoring 7
14CA	5322	bool	r/o	Limit value monitoring 8

[a] The unit for the values is based on the configuration for the analog value source selected for the input value.

8.3.4 Flow

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
14CB	5323	float	r/o	Flow 1
14CD	5325	float	r/o	Flow 2
14CF	5327	float	r/o	Total quantity flow 1[a]
14D1	5329	float	r/o	Total quantity flow 2[a]
14D3	5331	float	r/o	Total quantity last period flow 1[a]
14D5	5333	float	r/o	Total quantity last period flow 2[a]
14D7	5335	float	r/o	Pulse frequency flow 1[b]
14D9	5337	float	r/o	Pulse frequency flow 2[b]
14DB	5339	bool	r/o	Alarm 1 flow rate 1
14DC	5340	bool	r/o	Alarm 1 flow-through 2
14DD	5341	bool	r/o	Alarm 2 flow rate 1
14DE	5342	bool	r/o	Alarm 2 flow-through 2
14DF	5343	bool	r/w	Reset flow 1 only possible for unlimited pulse period, see configuration in the operating manual or installation instructions
14E0	5344	bool	r/w	Reset flow 2 only possible for unlimited pulse period, see configuration in the operating manual or installation instructions

[a] The unit for the values is based on the configuration for the respective flow function.

[b] Unit: Hertz

8.3.5 Salinity

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1C7A	7290	float	r/o	Salinity value 1
1C7C	7292	float	r/o	Salinity value 2
1C7E	7294	float	r/o	Salinity value 3
1C80	7296	float	r/o	Salinity value 4

8.3.6 Washtimer

Modbus PDU address Hex.	Dec.	Data type	Access	Data	
14E5	5349	bool	r/o	Binary value wash timer 1	
14E6	5350	bool	r/o	Binary value wash timer 2	
14E7	5351	bool	r/o	Hold time active wash timer 1	
14E8	5352	bool	r/o	Hold time active wash timer 2	
14E9	5353	uint32	r/o	Time to next washing process in ms wash timer 1[a]	
14EB	5355	uint32	r/o	Time to next washing process in ms wash timer 2[a]	
14ED	5357	uint32	r/o	Washing time elapsed wash timer 1[a]	
14EF	5359	uint32	r/o	Washing time elapsed wash timer 2[a]	
14F1	5361	uint32	r/o	Hold time elapsed wash timer 1[a]	
14F3	5363	uint32	r/o	Hold time elapsed wash timer 2[a]	
14F5	5365	byte	r/o	Current state wash timer 1	Meaning of the integer values: 0 = not started 1 = Time interval running 2 = Washing process running 3 = Hold time running 4 = Washing process running
14F6	5366	byte	r/o	Current state wash timer 2	
14F7	5367	uint32	r/o	Time to next washing process in min. wash timer 1[b]	
14F9	5369	uint32	r/o	Time to next washing process in min. wash timer 2[b]	
14FB	5371	uint32	r/o	Remaining time washing process in s wash timer 1[b]	
14FD	5373	uint32	r/o	Remaining time washing process in s wash timer 2[b]	

[a] Unit: milliseconds

[b] Unit: seconds

8.3.7 Controller

Modbus PDU address Hex.	Dec.	Data type	Access	Data
1503	5379	float	r/o	Actual value controller 1[a]
1505	5381	float	r/o	Actual value controller 2[a]
1507	5383	float	r/o	Actual value controller 3[a]
1509	5385	float	r/o	Actual value controller 4[a]
150B	5387	float	r/o	Setpoint value controller 1[a]
150D	5389	float	r/o	Setpoint value controller 2[a]
150F	5391	float	r/o	Setpoint value controller 3[a]
1511	5393	float	r/o	Setpoint value controller 4[a]
1513	5395	float	r/o	Output level display Y1 controller 1[b]
1515	5397	float	r/o	Output level display Y1 controller 2[b]
1517	5399	float	r/o	Output level display Y1 controller 3[b]

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1519	5401	float	r/o	Output level display Y1 controller 4[b]
151B	5403	float	r/o	Output level display Y2 controller 1[b]
151D	5405	float	r/o	Output level display Y2 controller 2[b]
151F	5407	float	r/o	Output level display Y2 controller 3[b]
1521	5409	float	r/o	Output level display Y2 controller 4[b]
1523	5411	float	r/o	Continuous output 1 controller 1[b]
1525	5413	float	r/o	Continuous output 1 controller 2[b]
1527	5415	float	r/o	Continuous output 1 controller 3[b]
1529	5417	float	r/o	Continuous output 1 controller 4[b]
152B	5419	float	r/o	Continuous output 2 controller 1[b]
152D	5421	float	r/o	Continuous output 2 controller 2[b]
152F	5423	float	r/o	Continuous output 2 controller 3[b]
1531	5425	float	r/o	Continuous output 2 controller 4[b]
1533	5427	bool	r/o	Binary output K1 controller 1
1534	5428	bool	r/o	Binary output K1 controller 2
1535	5429	bool	r/o	Binary output K1 controller 3
1536	5430	bool	r/o	Binary output K1 controller 4
1537	5431	bool	r/o	Binary output K2 controller 1
1538	5432	bool	r/o	Binary output K2 controller 2
1539	5433	bool	r/o	Binary output K2 controller 3
153A	5434	bool	r/o	Binary output K2 controller 4
153B	5435	bool	r/o	Dosing alarm controller 1
153C	5436	bool	r/o	Dosing alarm controller 2
153D	5437	bool	r/o	Dosing alarm controller 3
153E	5438	bool	r/o	Dosing alarm controller 4
153F	5439	bool	r/o	Manual mode controller 1 active
1540	5440	bool	r/o	Manual mode controller 2 active
1541	5441	bool	r/o	Manual mode controller 3 active
1542	5442	bool	r/o	Manual mode controller 4 active
1543	5443	bool	r/o	Autotuning controller 1 active
1544	5444	bool	r/o	Autotuning controller 2 active
1545	5445	bool	r/o	Autotuning controller 3 active
1546	5446	bool	r/o	Autotuning controller 4 active

[a] The unit for the values is based on the configuration of the analog value source selected for the controller actual value input.

[b] Unit: percent

8.3.8 Timers

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1567	5479	bool	r/o	Output timer 1
1568	5480	bool	r/o	Output timer 2
1C18	7192	bool	r/o	Output timer 3[a]

8 Modbus address tables

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1C19	7193	bool	r/o	Output timer 4[a]
1C1A	7194	bool	r/o	Output timer 5[a]
1C1B	7195	bool	r/o	Output timer 6[a]
1C1C	7196	bool	r/o	Output timer 7[a]
1C1D	7197	bool	r/o	Output timer 8[a]
1C1E	7198	bool	r/o	Output timer 9[a]
1C1F	7199	bool	r/o	Output timer 10[a]
1569	5481	bool	r/o	Delay timer 1
156A	5482	bool	r/o	Delay timer 2
1C22	7202	bool	r/o	Delay timer 3[a]
1C23	7203	bool	r/o	Delay timer 4[a]
1C24	7204	bool	r/o	Delay timer 5[a]
1C25	7205	bool	r/o	Delay timer 6[a]
1C26	7206	bool	r/o	Delay timer 7[a]
1C27	7207	bool	r/o	Delay timer 8[a]
1C28	7208	bool	r/o	Delay timer 9[a]
1C29	7209	bool	r/o	Delay timer 10[a]
156B	5483	uint32	r/o	Elapsed time timer 1[b]
156D	5485	uint32	r/o	Elapsed time timer 2[b]
1C2C	7212	uint32	r/o	Elapsed time timer 3[ab]
1C2E	7214	uint32	r/o	Elapsed time timer 4[ab]
1C30	7216	uint32	r/o	Elapsed time timer 5[ab]
1C32	7218	uint32	r/o	Elapsed time timer 6[ab]
1C34	7220	uint32	r/o	Elapsed time timer 7[ab]
1C36	7222	uint32	r/o	Elapsed time timer 8[ab]
1C38	7224	uint32	r/o	Elapsed time timer 9[ab]
1C3A	7226	uint32	r/o	Elapsed time timer 10[ab]
156F	5487	uint32	r/o	Remaining time timer 1[b]
1571	5489	uint32	r/o	Remaining time timer 2[b]
1C40	7232	uint32	r/o	Remaining time timer 3[ab]
1C42	7234	uint32	r/o	Remaining time timer 4[ab]
1C44	7236	uint32	r/o	Remaining time timer 5[ab]
1C46	7238	uint32	r/o	Remaining time timer 6[ab]
1C48	7240	uint32	r/o	Remaining time timer 7[ab]
1C4A	7242	uint32	r/o	Remaining time timer 8[ab]
1C4C	7244	uint32	r/o	Remaining time timer 9[ab]
1C4E	7246	uint32	r/o	Remaining time timer 10[ab]

[a] Timers 3 to 10 are available only from device software version 304.04.01 onwards.

[b] Unit: seconds

8.3.9 Counters

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
158B	5515	uint32	r/o	Output counter 1
158D	5517	uint32	r/o	Output counter 2
158F	5519	uint32	r/o	Output counter 3
1591	5521	uint32	r/o	Output counter 4
1593	5523	bool	r/o	Alarm counter 1
1594	5524	bool	r/o	Alarm counter 2
1595	5525	bool	r/o	Alarm counter 3
1596	5526	bool	r/o	Alarm counter 4
1597	5527	bool	r/w	Reset counter 1
1598	5528	bool	r/w	Reset counter 2
1599	5529	bool	r/w	Reset counter 3
159A	5530	bool	r/w	Reset counter 4

8.3.10 Calibration timers

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
159B	5531	uint32	r/o	Remaining time to calibration IN 6[a]
159D	5533	uint32	r/o	Remaining time to calibration IN 7[a]
159F	5535	uint32	r/o	Remaining time to calibration IN 8[a]
15A5	5541	uint32	r/o	Remaining time to calibration IN 11[a]
15A7	5543	uint32	r/o	Remaining time to calibration IN 12[a]
1C54	7252	uint32	r/o	Remaining time to calibration of digital sensor 1[ab]
1C56	7254	uint32	r/o	Remaining time to calibration of digital sensor 2[ab]
1C58	7256	uint32	r/o	Remaining time to calibration of digital sensor 3[ab]
1C5A	7258	uint32	r/o	Remaining time to calibration of digital sensor 4[ab]
1C5C	7260	uint32	r/o	Remaining time to calibration of digital sensor 5[ab]
1C5E	7262	uint32	r/o	Remaining time to calibration of digital sensor 6[ab]
15B7	5559	bool	r/o	Calibration alarm IN 6
15B8	5560	bool	r/o	Calibration alarm IN 7
15B9	5561	bool	r/o	Calibration alarm IN 8
15BC	5564	bool	r/o	Calibration alarm IN 11
15BD	5565	bool	r/o	Calibration alarm IN 12
1C6C	7276	bool	r/o	Calibration alarm of digital sensor 1[b]
1C6D	7277	bool	r/o	Calibration alarm of digital sensor 2[b]
1C6E	7278	bool	r/o	Calibration alarm of digital sensor 3[b]
1C6F	7279	bool	r/o	Calibration alarm of digital sensor 4[b]
1C70	7280	bool	r/o	Calibration alarm of digital sensor 5[b]
1C71	7281	bool	r/o	Calibration alarm of digital sensor 6[b]

[a] Unit: seconds

[b] The calibration timers for digital sensors are available only from device software version 304.04.01 onwards.

8.3.11 Formula

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
15C5	5573	float	r/o	Calculated result formula 1
15C7	5575	float	r/o	Calculated result formula 2
15C9	5577	float	r/o	Calculated result formula 3
15CB	5579	float	r/o	Calculated result formula 4
15CD	5581	float	r/o	Calculated result formula 5
15CF	5583	float	r/o	Calculated result formula 6
15D1	5585	float	r/o	Calculated result formula 7
15D3	5587	float	r/o	Calculated result formula 8
1BD4	7124	float	r/o	Calculated result formula 9[a]
1BD6	7126	float	r/o	Calculated result formula 10[a]
1BD8	7128	float	r/o	Calculated result formula 11[a]
1BDA	7130	float	r/o	Calculated result formula 12[a]
1BDC	7132	float	r/o	Calculated result formula 13[a]
1BDE	7134	float	r/o	Calculated result formula 14[a]
1BE0	7136	float	r/o	Calculated result formula 15[a]
1BE2	7138	float	r/o	Calculated result formula 16[a]
15D5	5589	bool	r/o	Comparison function result formula 1
15D6	5590	bool	r/o	Comparison function result formula 2
15D7	5591	bool	r/o	Comparison function result formula 3
15D8	5592	bool	r/o	Comparison function result formula 4
15D9	5593	bool	r/o	Comparison function result formula 5
15DA	5594	bool	r/o	Comparison function result formula 6
15DB	5595	bool	r/o	Comparison function result formula 7
15DC	5596	bool	r/o	Comparison function result formula 8
1BE4	7140	bool	r/o	Comparison function result formula 9[a]
1BE5	7141	bool	r/o	Comparison function result formula 10[a]
1BE6	7142	bool	r/o	Comparison function result formula 11[a]
1BE7	7143	bool	r/o	Comparison function result formula 12[a]
1BE8	7144	bool	r/o	Comparison function result formula 13[a]
1BE9	7145	bool	r/o	Comparison function result formula 14[a]
1BEA	7146	bool	r/o	Comparison function result formula 15[a]
1BEB	7147	bool	r/o	Comparison function result formula 16[a]

[a] Mathematical formulas 9 to 16 are available only from device software version 304.04.01 onwards.

8.3.12 Logic formula

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
15DD	5597	bool	r/o	Result logic formula 1
15DE	5598	bool	r/o	Result logic formula 2
15DF	5599	bool	r/o	Result logic formula 3
15E0	5600	bool	r/o	Result logic formula 4
15E1	5601	bool	r/o	Result logic formula 5
15E2	5602	bool	r/o	Result logic formula 6
15E3	5603	bool	r/o	Result logic formula 7
15E4	5604	bool	r/o	Result logic formula 8
1BEC	7148	bool	r/o	Result logic formula 9[a]
1BED	7149	bool	r/o	Result logic formula 10[a]
1BEE	7150	bool	r/o	Result logic formula 11[a]
1BEF	7151	bool	r/o	Result logic formula 12[a]
1BF0	7152	bool	r/o	Result logic formula 13[a]
1BF1	7153	bool	r/o	Result logic formula 14[a]
1BF2	7154	bool	r/o	Result logic formula 15[a]
1BF3	7155	bool	r/o	Result logic formula 16[a]
1BF4	7156	bool	r/o	Result logic formula 17[a]
1BF5	7157	bool	r/o	Result logic formula 18[a]
1BF6	7158	bool	r/o	Result logic formula 19[a]
1BF7	7159	bool	r/o	Result logic formula 20[a]
1BF8	7160	bool	r/o	Result logic formula 21[a]
1BF9	7161	bool	r/o	Result logic formula 22[a]
1BFA	7162	bool	r/o	Result logic formula 23[a]
1BFB	7163	bool	r/o	Result logic formula 24[a]
1BFC	7164	bool	r/o	Result logic formula 25[a]
1BFD	7165	bool	r/o	Result logic formula 26[a]
1BFE	7166	bool	r/o	Result logic formula 27[a]
1BFF	7167	bool	r/o	Result logic formula 28[a]
1C00	7168	bool	r/o	Result logic formula 29[a]
1C01	7169	bool	r/o	Result logic formula 30[a]
15E5	5605	bool	r/o	Alarm logic formula 1
15E6	5606	bool	r/o	Alarm logic formula 2
15E7	5607	bool	r/o	Alarm logic formula 3
15E8	5608	bool	r/o	Alarm logic formula 4
15E9	5609	bool	r/o	Alarm logic formula 5
15EA	5610	bool	r/o	Alarm logic formula 6
15EB	5611	bool	r/o	Alarm logic formula 7
15EC	5612	bool	r/o	Alarm logic formula 8
1C02	7170	bool	r/o	Alarm logic formula 9[a]
1C03	7171	bool	r/o	Alarm logic formula 10[a]
1C04	7172	bool	r/o	Alarm logic formula 11[a]
1C05	7173	bool	r/o	Alarm logic formula 12[a]

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1C06	7174	bool	r/o	Alarm logic formula 13[a]
1C07	7175	bool	r/o	Alarm logic formula 14[a]
1C08	7176	bool	r/o	Alarm logic formula 15[a]
1C09	7177	bool	r/o	Alarm logic formula 16[a]
1C0A	7178	bool	r/o	Alarm logic formula 17[a]
1C0B	7179	bool	r/o	Alarm logic formula 18[a]
1C0C	7180	bool	r/o	Alarm logic formula 19[a]
1C0D	7181	bool	r/o	Alarm logic formula 20[a]
1C0E	7182	bool	r/o	Alarm logic formula 21[a]
1C0F	7183	bool	r/o	Alarm logic formula 22[a]
1C10	7184	bool	r/o	Alarm logic formula 23[a]
1C11	7185	bool	r/o	Alarm logic formula 24[a]
1C12	7186	bool	r/o	Alarm logic formula 25[a]
1C13	7187	bool	r/o	Alarm logic formula 26[a]
1C14	7188	bool	r/o	Alarm logic formula 27[a]
1C15	7189	bool	r/o	Alarm logic formula 28[a]
1C16	7190	bool	r/o	Alarm logic formula 29[a]
1C17	7191	bool	r/o	Alarm logic formula 30[a]

[a] Logic formulas 9 to 30 are available only from device software version 304.04.01 onwards.

8.3.13 analysis inputs

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1645	5701	float	r/o	Measured value IN 7 uncomp.[a]
1647	5703	float	r/o	Measured value IN 8 uncomp.[a]
164D	5709	float	r/o	Measured value IN 7 compensated[b]
164F	5711	float	r/o	Measured value IN 8 compensated[b]
165D	5725	bool	r/o	Alarm 1 IN 7
165E	5726	bool	r/o	Alarm 1 IN 8
1661	5729	bool	r/o	Alarm 2 IN 7
1662	5730	bool	r/o	Alarm 2 IN 8
1665	5733	bool	r/o	Sensor alarm IN 7
1666	5734	bool	r/o	Sensor alarm IN 8
1669	5737	bool	r/o	Calibration IN 7 active
166A	5738	bool	r/o	Calibration IN 8 active
1679	5753	byte	r/o	Active measuring range IN 7
167A	5754	byte	r/o	Active measuring range IN 8

[a] The unit for uncompensated values (value of the electrical sensor signal) depends on the type of the connected sensor for which the respective analysis input is configured.

[b] The unit for the values is based on the configuration for the respective analysis input.

8.3.14 Universal inputs

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
169A	5786	float	r/o	Measured value IN 6 uncomp.[a]
169C	5788	float	r/o	Measured value IN 11 uncomp.[a]
169E	5790	float	r/o	Measured value IN 12 uncomp.[a]
16A0	5792	float	r/o	Measured value IN 6 compensated[b]
16A2	5794	float	r/o	Measured value IN 11 compensated[b]
16A4	5796	float	r/o	Measured value IN 12 compensated[b]
16A6	5798	bool	r/o	Alarm 1 IN 6
16A7	5799	bool	r/o	Alarm 1 IN 11
16A8	5800	bool	r/o	Alarm 1 IN 12
16A9	5801	bool	r/o	Alarm 2 IN 6
16AA	5802	bool	r/o	Alarm 2 IN 11
16AB	5803	bool	r/o	Alarm 2 IN 12
16AC	5804	bool	r/o	Calibration IN 6 active
16AD	5805	bool	r/o	Calibration IN 11 active
16AE	5806	bool	r/o	Calibration IN 12 active

[a] The unit for uncompensated values (value of the electrical sensor signal) depends on the type of the connected sensor for which the respective analysis input is configured.

[b] The unit for the values is based on the configuration for the respective analysis input.

8.3.15 Temperature inputs

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
16BB	5819	float	r/o	Measured temperature value IN 4[a]
16BD	5821	float	r/o	Measured temperature value IN 5[a]
16BF	5823	float	r/o	Sensor resistance IN 4[a]
16C1	5825	float	r/o	Sensor resistance IN 5[a]
16C3	5827	bool	r/o	Alarm 1 IN 4
16C4	5828	bool	r/o	Alarm 1 IN 5
16C5	5829	bool	r/o	Alarm 2 IN 4
16C6	5830	bool	r/o	Alarm 2 IN 5
16C7	5831	bool	r/o	Calibration IN 4 active
16C8	5832	bool	r/o	Calibration IN 5 active

[a] and is specified in the "Basic settings".

8.3.16 Binary inputs

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
16CC	5836	word	r/o	Pulse frequency flow rate 1
16CD	5837	word	r/o	Pulse frequency flow rate 2
16CF	5839	word	r/o	Pulse period flow rate 1
16D0	5840	word	r/o	Pulse period flow rate 2
16D1	5841	bool	r/o	Electrical logic signal IN 1
16D2	5842	bool	r/o	Electrical logic signal IN 2
16D3	5843	bool	r/o	Electrical logic signal IN 3
16D4	5844	bool	r/o	Electrical logic signal IN 13
16D5	5845	bool	r/o	Electrical logic signal IN 14
16D6	5846	bool	r/o	Electrical logic signal IN 15
16D7	5847	bool	r/o	Electrical logic signal IN 16
16D8	5848	bool	r/o	Electrical logic signal IN 17
16D9	5849	bool	r/o	Electrical logic signal IN 18
16DA	5850	bool	r/o	Binary value IN 1
16DB	5851	bool	r/o	Binary value IN 2
16DC	5852	bool	r/o	Binary value IN 3
16DD	5853	bool	r/o	Binary value IN 13
16DE	5854	bool	r/o	Binary value IN 14
16DF	5855	bool	r/o	Binary value IN 15
16E0	5856	bool	r/o	Binary value IN 16
16E1	5857	bool	r/o	Binary value IN 17
16E2	5858	bool	r/o	Binary value IN 18
16E3	5859	bool	r/o	Alarm binary input 1
16E4	5860	bool	r/o	Alarm binary input 2
16E5	5861	bool	r/o	Alarm binary input 3
16E6	5862	bool	r/o	Alarm binary input 13
16E7	5863	bool	r/o	Alarm binary input 14
16E8	5864	bool	r/o	Alarm binary input 15
16E9	5865	bool	r/o	Alarm binary input 16
16EA	5866	bool	r/o	Alarm binary input 17
16EB	5867	bool	r/o	Alarm binary input 18

8.3.17 Analog outputs

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
16EC	5868	float	r/o	Value manual input OUT 4[a]
16F0	5872	float	r/o	Value manual input OUT 6[a]
16F2	5874	float	r/o	Value manual input OUT 8[a]
16F4	5876	float	r/o	Value manual input OUT 10[a]
16F6	5878	float	r/o	Value manual input OUT 12[a]

Modbus PDU address		Data type	Access	Data	
Hex.	Dec.				
16FE	5886	bool	r/o	Manual mode OUT 4 active	
1700	5888	bool	r/o	Manual mode OUT 6 active	
1701	5889	bool	r/o	Manual mode OUT 8 active	
1702	5890	bool	r/o	Manual mode OUT 10 active	
1703	5891	bool	r/o	Manual mode OUT 12 active	
1707	5895	byte	r/o	Signal type with value range OUT 4	Meaning of integer values:0 = 0 to 10 V 1 = 0 to 20 mA 2 = 4 to 20 mA 3 = 10 to 0 V 4 = 20 to 0 mA 5 = 20 to 4 mA
1709	5897	byte	r/o	Signal type with value range OUT 6	
170A	5898	byte	r/o	Signal type with value range OUT 8	
170B	5899	byte	r/o	Signal type with value range OUT 10	
170C	5900	byte	r/o	Signal type with value range OUT 12	
1710	5904	byte	r/o	Signal type OUT 4	Meaning of integer values: 0 = Voltage signal 1 = Current signal
1712	5906	byte	r/o	Signal type OUT 6	
1713	5907	byte	r/o	Signal type OUT 8	
1714	5908	byte	r/o	Signal type OUT 10	
1715	5909	byte	r/o	Signal type OUT 12	
1719	5913	float	r/o	Analog value electrical standard signal OUT 4[b]	
171D	5917	float	r/o	Analog value electrical standard signal OUT 6[b]	
171F	5919	float	r/o	Analog value electrical standard signal OUT 8[b]	
1721	5921	float	r/o	Analog value electrical standard signal OUT 10[b]	
1723	5923	float	r/o	Analog value electrical standard signal OUT 12[b]	
172B	5931	float	r/o	Analog value in percent OUT 4[a]	
172F	5935	float	r/o	Analog value in percent OUT 6[a]	
1731	5937	float	r/o	Analog value in percent OUT 8[a]	
1733	5939	float	r/o	Analog value in percent OUT 10[a]	
1735	5941	float	r/o	Analog value in percent OUT 12[a]	
173D	5949	bool	r/o	Behavior during calibration active OUT 4	
173F	5951	bool	r/o	Behavior during calibration active OUT 6	
1740	5952	bool	r/o	Behavior during calibration active OUT 8	
1741	5953	bool	r/o	Behavior during calibration active OUT 10	
1742	5954	bool	r/o	Behavior during calibration active OUT 12	

[a] Unit: percent

[b] The unit depends on the configuration of the respective analog input.fm:Linefeed/>Voltage signals: Volt; Current signals: Milliamperes

8 Modbus address tables

8.3.18 Binary outputs

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1746	5958	bool	r/o	Binary value OUT 1
1747	5959	bool	r/o	Binary value OUT 2
1748	5960	bool	r/o	Binary value OUT 3
1749	5961	bool	r/o	Binary value OUT 6
174A	5962	bool	r/o	Binary value OUT 7
174B	5963	bool	r/o	Binary value OUT 8
174C	5964	bool	r/o	Binary value OUT 9
174D	5965	bool	r/o	Binary value OUT 10
174E	5966	bool	r/o	Binary value OUT 11
174F	5967	bool	r/o	Binary value OUT 12
1750	5968	bool	r/o	Binary value OUT 13

8.3.19 Service data

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1757	5975	float	r/o	Battery voltage[a]
1759	5977	float	r/o	Board temperature[b]

[a] Unit: Volts

[b] The temperature unit for the board temperature corresponds to the configured temperature unit for device operation and is specified in the "Basic settings".

8.3.20 Hardware information

Modbus PDU address		Data type	Access	Data	Meaning of the integer values
Hex.	Dec.				
175B	5979	byte	r/o	HW act.: Device type [0=surface-mount, 1=control cabinet]	0 = AQUIS touch S 1 = AQUIS touch P
175C	5980	byte	r/o	Display type	0 = 5.5“ 1 = 3.5“

8.3.21 External analog inputs

Modbus PDU address		Data type	Access	Data	
Hex.	Dec.				
17B2	6066	float	r/w	External analog input 1[a]	For reading from and writing to the external analog inputs
17B4	6068	float	r/w	External analog input 2[a]	
17B6	6070	float	r/w	External analog input 3[a]	
17B8	6072	float	r/w	External analog input 4[a]	
17BA	6074	float	r/w	External analog input 5[a]	
17BC	6076	float	r/w	External analog input 6[a]	
17BE	6078	float	r/w	External analog input 7[a]	
17C0	6080	float	r/w	External analog input 8[a]	
1CC3	7363	float	r/w	External analog input 9[ab]	
1CC5	7365	float	r/w	External analog input 10[ab]	
1CC7	7367	float	r/w	External analog input 11[ab]	
1CC9	7369	float	r/w	External analog input 12[ab]	
1CCB	7371	float	r/w	External analog input 13[ab]	
1CCD	7373	float	r/w	External analog input 14[ab]	
1CCF	7375	float	r/w	External analog input 15[ab]	
1CD1	7377	float	r/w	External analog input 16[ab]	
17C2	6082	float	r/o	Analog value external analog input 1[a]	Solely for reading the external analog inputs after display range check The values stored here are copied from the addresses 0x17B2 to 0x17C0 and 0x1CC3 to 0x1CD1, and subjected to a display range check.
17C4	6084	float	r/o	Analog value external analog input 2[a]	
17C6	6086	float	r/o	Analog value external analog input 3[a]	
17C8	6088	float	r/o	Analog value external analog input 4[a]	
17CA	6090	float	r/o	Analog value external analog input 5[a]	
17CC	6092	float	r/o	Analog value external analog input 6[a]	
17CE	6094	float	r/o	Analog value external analog input 7[a]	
17D0	6096	float	r/o	Analog value external analog input 8[a]	
1CD3	7379	float	r/o	Analog value external analog input 9[ab]	
1CD5	7381	float	r/o	Analog value external analog input 10a[b]	
1CD7	7383	float	r/o	Analog value external analog input 11[ab]	
1CD9	7385	float	r/o	Analog value external analog input 12[ab]	
1CDB	7387	float	r/o	Analog value external analog input 13[ab]	
1CDD	7389	float	r/o	Analog value external analog input 14[ab]	
1CDF	7391	float	r/o	Analog value external analog input 15[ab]	
1CE1	7393	float	r/o	Analog value external analog input 16[ab]	

8 Modbus address tables

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
17D2	6098	bool	r/o	Alarm 1 external analog input 1
17D3	6099	bool	r/o	Alarm 1 external analog input 2
17D4	6100	bool	r/o	Alarm 1 external analog input 3
17D5	6101	bool	r/o	Alarm 1 external analog input 4
17D6	6102	bool	r/o	Alarm 1 external analog input 5
17D7	6103	bool	r/o	Alarm 1 external analog input 6
17D8	6104	bool	r/o	Alarm 1 external analog input 7
17D9	6105	bool	r/o	Alarm 1 external analog input 8
1CE3	7395	bool	r/o	Alarm 1 of external analog input 9[b]
1CE4	7396	bool	r/o	Alarm 1 of external analog input 10[b]
1CE5	7397	bool	r/o	Alarm 1 of external analog input 11[b]
1CE6	7398	bool	r/o	Alarm 1 of external analog input 12[b]
1CE7	7399	bool	r/o	Alarm 1 of external analog input 13[b]
1CE8	7400	bool	r/o	Alarm 1 of external analog input 14[b]
1CE9	7401	bool	r/o	Alarm 1 of external analog input 15[b]
1CEA	7402	bool	r/o	Alarm 1 of external analog input 16[b]
17DA	6106	bool	r/o	Alarm 2 external analog input 1
17DB	6107	bool	r/o	Alarm 2 external analog input 2
17DC	6108	bool	r/o	Alarm 2 external analog input 3
17DD	6109	bool	r/o	Alarm 2 external analog input 4
17DE	6110	bool	r/o	Alarm 2 external analog input 5
17DF	6111	bool	r/o	Alarm 2 external analog input 6
17E0	6112	bool	r/o	Alarm 2 external analog input 7
17E1	6113	bool	r/o	Alarm 2 external analog input 8
1CEB	7403	bool	r/o	Alarm 2 of external analog input 9[b]
1CEC	7404	bool	r/o	Alarm 2 of external analog input 10[b]
1CED	7405	bool	r/o	Alarm 2 of external analog input 11[b]
1CEE	7406	bool	r/o	Alarm 2 of external analog input 12[b]
1CEF	7407	bool	r/o	Alarm 2 of external analog input 13[b]
1CF0	7408	bool	r/o	Alarm 2 of external analog input 14[b]
1CF1	7409	bool	r/o	Alarm 2 of external analog input 15[b]
1CF2	7410	bool	r/o	Alarm 2 of external analog input 16[b]

[a] The unit for the values is based on the configuration for the respective external analog input.

[b] The external analog inputs 9 to 16 are only available starting from device software version 304.08.01.

8.3.22 External binary inputs inputs

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
17E2	6114	bool	r/w	Binary value external digital input 1
17E3	6115	bool	r/w	Binary value external digital input 2
17E4	6116	bool	r/w	Binary value external digital input 3
17E5	6117	bool	r/w	Binary value external digital input 4
17E6	6118	bool	r/w	Binary value external digital input 5
17E7	6119	bool	r/w	Binary value external digital input 6
17E8	6120	bool	r/w	Binary value external digital input 7
17E9	6121	bool	r/w	Binary value external digital input 8
19DF	6623	bool	r/w	Binary value external digital input 9[a]
19E0	6624	bool	r/w	Binary value external digital input 10[a]
19E1	6625	bool	r/w	Binary value external digital input 11[a]
19E2	6626	bool	r/w	Binary value external digital input 12[a]
19E3	6627	bool	r/w	Binary value external digital input 13[a]
19E4	6628	bool	r/w	Binary value external digital input 14[a]
19E5	6629	bool	r/w	Binary value external digital input 15[a]
19E6	6630	bool	r/w	Binary value external digital input 16[a]
17EA	6122	bool	r/o	Alarm external digital input 1
17EB	6123	bool	r/o	Alarm external digital input 2
17EC	6124	bool	r/o	Alarm external digital input 3
17ED	6125	bool	r/o	Alarm external digital input 4
17EE	6126	bool	r/o	Alarm external digital input 5
17EF	6127	bool	r/o	Alarm external digital input 6
17F0	6128	bool	r/o	Alarm external digital input 7
17F1	6129	bool	r/o	Alarm external digital input 8
19E7	6631	bool	r/o	Alarm external digital input 9[a]
19E8	6632	bool	r/o	Alarm external digital input 10[a]
19E9	6633	bool	r/o	Alarm external digital input 11[a]
19EA	6634	bool	r/o	Alarm external digital input 12[a]
19EB	6635	bool	r/o	Alarm external digital input 13[a]
19EC	6636	bool	r/o	Alarm external digital input 14[a]
19ED	6637	bool	r/o	Alarm external digital input 15[a]
19EE	6638	bool	r/o	Alarm external digital input 16[a]

[a] The external binary inputs 9 to 16 are only available starting from device software version 304.08.01.

8.3.23 Collective alarm

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
19A1	6561	bool	r/o	Collective alarm
19A2	6562	bool	r/o	Collective alarm acknowledged
19A4	6564	bool	r/o	Memory alarm
19A5	6565	bool	r/o	User logged on

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
19A9	6569	bool	r/o	USB flash drive inserted into host interface
19AA	6570	bool	r/o	Profibus error
19AB	6571	bool	r/o	Battery empty
19AC	6572	bool	r/o	Battery weak
19AD	6573	bool	r/o	Calibration active
19AE	6574	bool	r/o	Controller manual mode active
19AF	6575	bool	r/o	Temperature unit device operation is °F
1C82	7298	bool	r/o	Internal housing temperature too high

8.3.24 Ethernet

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
19C9	6601	byte	r/o	IP address 1st octet
19CA	6602	byte	r/o	IP address 2nd octet
19CB	6603	byte	r/o	IP address 3rd octet
19CC	6604	byte	r/o	IP address 4th octet

8.3.25 Modbus error

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
19CD	6605	bool	r/o	COM1 error state
19CE	6606	word	r/o	COM1 last error code
19CF	6607	bool	r/o	COM2 error state
19D0	6608	word	r/o	COM2 last error code
19D1	6609	bool	r/o	Modbus/TCP slave 1 error state
19D2	6610	word	r/o	Modbus/TCP slave 1 last error code
19D3	6611	bool	r/o	Modbus/TCP slave 2 error state
19D4	6612	word	r/o	Modbus/TCP slave 2 last error code
19D7	6615	bool	r/o	Modbus/TCP master 1 error state[a]
19D8	6616	word	r/o	Modbus/TCP master 1 last error code[a]
19D9	6617	bool	r/o	Modbus/TCP master 2 error state[a]
19DA	6618	word	r/o	Modbus/TCP master 2 last error code[a]
19DB	6619	bool	r/o	Modbus/TCP master 3 error state[a]
19DC	6620	word	r/o	Modbus/TCP master 3 last error code[a]
19DD	6621	bool	r/o	Modbus/TCP master 4 error state[a]
19DE	6622	word	r/o	Modbus/TCP master 4 last error code[a]

[a] Error data for Modbus/TCP master are available only from device software version 304.08.01 onwards.

8.3.26 JUMO digiLine

Introduction

The JUMO AQUIS touch P supports, from device software version 304.03.01 with the JUMO digiLine protocol, the operation of digital sensors. The JUMO AQUIS touch P communicates, as the JUMO digiLine master device, with the digital sensors on the JUMO digiLine bus and provisions data of the JUMO digiLine bus as Modbus slave. The sensor-specific variable assignments and Modbus addresses of the JUMO digiLine data are indicated in this chapter.

Analog values

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1A1F	6687	float	r/o	digital sensor 1: Analog value 1
1A21	6689	float	r/o	digital sensor 1: Analog value 2
1A23	6691	float	r/o	digital sensor 1: Analog value 3
1A25	6693	float	r/o	digital sensor 1: Analog value 4
1A27	6695	float	r/o	digital sensor 2: Analog value 1
1A29	6697	float	r/o	digital sensor 2: Analog value 2
1A2B	6699	float	r/o	digital sensor 2: Analog value 3
1A2D	6701	float	r/o	digital sensor 2: Analog value 4
1A2F	6703	float	r/o	digital sensor 3: Analog value 1
1A31	6705	float	r/o	digital sensor 3: Analog value 2
1A33	6707	float	r/o	digital sensor 3: Analog value 3
1A35	6709	float	r/o	digital sensor 3: Analog value 4
1A37	6711	float	r/o	digital sensor 4: Analog value 1
1A39	6713	float	r/o	digital sensor 4: Analog value 2
1A3B	6715	float	r/o	digital sensor 4: Analog value 3
1A3D	6717	float	r/o	digital sensor 4: Analog value 4
1A3F	6719	float	r/o	digital sensor 5: Analog value 1
1A41	6721	float	r/o	digital sensor 5: Analog value 2
1A43	6723	float	r/o	digital sensor 5: Analog value 3
1A45	6725	float	r/o	digital sensor 5: Analog value 4
1A47	6727	float	r/o	digital sensor 6: Analog value 1
1A49	6729	float	r/o	digital sensor 6: Analog value 2
1A4B	6731	float	r/o	digital sensor 6: Analog value 3
1A4D	6733	float	r/o	digital sensor 6: Analog value 4

The variables in the address table above have different data assignments for the different available sensor types. The table below indicates the sensor-specific assignment of the analog values.

8 Modbus address tables

Sensor type	Analog value 1	Analog value 2	Analog value 3	Analog value 4
JUMO digiLine pH	pH measured value uncomp.	pH measured value compensated	Temperature measured value[a]	Sensor stress
JUMO digiLine ORP	Input voltage[b]	measured value[c]	-	-
JUMO digiLine T	Temperature measured value	-	-	-
JUMO digiLine CR/Ci	Uncompensated conductivity measured value[d]	Compensated conductivity measured value[d]	Measured temperature value	Sensor stress
JUMO digiLine O-DO S10	Compensated measured oxygen value [e]	Measured temperature value	-	-
JUMO digiLine O-DO H10/H20[f]	Compensated measured oxygen value [e]	Measured temperature value	-	-
JUMO flowTRANS US W02 digiLine[f]	Flow rate value	Measured temperature value	Pressure measurement value	-
JUMO ecoLine O-DO	Compensated measured oxygen value[g]	Temperature measured value	-	-
JUMO ecoLine NTU	Turbidity measured value	Temperature measured value	-	-
JUMO tecLine (sensors for disinfection measurands)	Cell current in nanoamperes	Concentration measured value	Measured temperature value	-

[a] Only on sensors with integrated temperature probe and device version with Variopin connection.

[b] The input voltage is indicated as an uncalibrated raw value in mV.

[c] Measured value based on the configuration as a redox potential in mV or as a percentage concentration (see operating manual or installation instructions of the JUMO AQUIS touch P).

[d] Measured value based on the configuration of the active measuring range (see operating manual or installation instructions of the JUMO AQUIS touch P).

[e] Measured value based on the configuration as concentration value, saturation value, or as oxygen partial pressure (see operating manual or installation instructions of the JUMO AQUIS touch P).

[f] Available from device software version 304.09.01.

[g] Measured value based on the configuration as concentration value or as saturation value (see operating manual or installation instructions of the JUMO AQUIS touch P).

Count values

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1A4F	6735	float	r/o	Digital sensor 1 count value 1
1A51	6737	float	r/o	Digital sensor 1 count value 2
1A53	6739	float	r/o	Digital sensor 1 count value 3
1A55	6741	float	r/o	Digital sensor 1 count value 4
1A57	6743	float	r/o	Digital sensor 1 count value 5
1A59	6745	float	r/o	Digital sensor 1 count value 6
1A5B	6747	float	r/o	Digital sensor 1 count value 7
1A5D	6749	float	r/o	Digital sensor 1 count value 8
1A5F	6751	float	r/o	Digital sensor 1 count value 9
1A61	6753	float	r/o	Digital sensor 1 count value 10
1A63	6755	float	r/o	Digital sensor 2 count value 1
1A65	6757	float	r/o	Digital sensor 2 count value 2
1A67	6759	float	r/o	Digital sensor 2 count value 3
1A69	6761	float	r/o	Digital sensor 2 count value 4
1A6B	6763	float	r/o	Digital sensor 2 count value 5
1A6D	6765	float	r/o	Digital sensor 2 count value 6
1A6F	6767	float	r/o	Digital sensor 2 count value 7
1A71	6769	float	r/o	Digital sensor 2 count value 8
1A73	6771	float	r/o	Digital sensor 2 count value 9
1A75	6773	float	r/o	Digital sensor 2 count value 10
1A77	6775	float	r/o	Digital sensor 3 count value 1
1A79	6777	float	r/o	Digital sensor 3 count value 2
1A7B	6779	float	r/o	Digital sensor 3 count value 3
1A7D	6781	float	r/o	Digital sensor 3 count value 4
1A7F	6783	float	r/o	Digital sensor 3 count value 5
1A81	6785	float	r/o	Digital sensor 3 count value 6
1A83	6787	float	r/o	Digital sensor 3 count value 7
1A85	6789	float	r/o	Digital sensor 3 count value 8
1A87	6791	float	r/o	Digital sensor 3 count value 9
1A89	6793	float	r/o	Digital sensor 3 count value 10
1A8B	6795	float	r/o	Digital sensor 4 count value 1
1A8D	6797	float	r/o	Digital sensor 4 count value 2
1A8F	6799	float	r/o	Digital sensor 4 count value 3
1A91	6801	float	r/o	Digital sensor 4 count value 4
1A93	6803	float	r/o	Digital sensor 4 count value 5
1A95	6805	float	r/o	Digital sensor 4 count value 6
1A97	6807	float	r/o	Digital sensor 4 count value 7
1A99	6809	float	r/o	Digital sensor 4 count value 8
1A9B	6811	float	r/o	Digital sensor 4 count value 9
1A9D	6813	float	r/o	Digital sensor 4 count value 10
1A9F	6815	float	r/o	Digital sensor 5 count value 1
1AA1	6817	float	r/o	Digital sensor 5 count value 2
1AA3	6819	float	r/o	Digital sensor 5 count value 3

8 Modbus address tables

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1AA5	6821	float	r/o	Digital sensor 5 count value 4
1AA7	6823	float	r/o	Digital sensor 5 count value 5
1AA9	6825	float	r/o	Digital sensor 5 count value 6
1AAB	6827	float	r/o	Digital sensor 5 count value 7
1AAD	6829	float	r/o	Digital sensor 5 count value 8
1AAF	6831	float	r/o	Digital sensor 5 count value 9
1AB1	6833	float	r/o	Digital sensor 5 count value 10
1AB3	6835	float	r/o	Digital sensor 6 count value 1
1AB5	6837	float	r/o	Digital sensor 6 count value 2
1AB7	6839	float	r/o	Digital sensor 6 count value 3
1AB9	6841	float	r/o	Digital sensor 6 count value 4
1ABB	6843	float	r/o	Digital sensor 6 count value 5
1ABD	6845	float	r/o	Digital sensor 6 count value 6
1ABF	6847	float	r/o	Digital sensor 6 count value 7
1AC1	6849	float	r/o	Digital sensor 6 count value 8
1AC3	6851	float	r/o	Digital sensor 6 count value 9
1AC5	6853	float	r/o	Digital sensor 6 count value 10

The data allocation of the variables in the address table varies depending on the sensor or device type. Sensors of the types JUMO ecoLine, JUMO tecLine, as well as JUMO digiLine ORP and digiLine T do not provide count values. The following table shows the assignment of count values for the respective types.

Count values	Sensor type JUMO digiLine				JUMO flowTRANS US W02 digiLine[a]
	pH	CR/Ci	O-DO S10	O-DO H10/H20[a]	
1	Number CIP	Number CIP	Cap quality[b],[a]	Number of CIP total	Totalizer 1[c]
2	Number SIP	Number SIP	-	Number of SIP total	Carry of totalizer 1
3	Number min. temp.	Duration min. temp.[d]	-	Number of CIP since last calibration	Totalizer 2[c]
4	Number max. temp.	Duration max. temp.[d]	-	Number of SIP since last calibration	Carry of totalizer 2
5	Number min. pH	Number min. temp.	-	Operating hours outside the measured temperature	-
6	Number max. pH	Number max. temp.	-	Operating hours outside the operating temperature	-
7	Duration min. temp.[d]	Duration of max. conductivity[d]	-	Pulse counter	-
8	Duration max. temp.[d]	Number of max. conductivity			
9	Duration min. pH[d]	-	-	-	-
10	Duration max. pH[d]	-	-	-	-

[a] Available from device software version 304.09.01.

[b] Specification in %.

[c] Specification: positive, negative, or balanced.

[d] Specification in minutes.

Alarms

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1AC7	6855	bool	r/o	Digital sensor 1 alarm 1
1AC8	6856	bool	r/o	Digital sensor 2 alarm 1
1AC9	6857	bool	r/o	Digital sensor 3 alarm 1
1ACA	6858	bool	r/o	Digital sensor 4 alarm 1
1ACB	6859	bool	r/o	Digital sensor 5 alarm 1
1ACC	6860	bool	r/o	Digital sensor 6 alarm 1
1ACD	6861	bool	r/o	Digital sensor 1 alarm 2
1ACE	6862	bool	r/o	Digital sensor 2 alarm 2
1ACF	6863	bool	r/o	Digital sensor 3 alarm 2
1AD0	6864	bool	r/o	Digital sensor 4 alarm 2
1AD1	6865	bool	r/o	Digital sensor 5 alarm 2
1AD2	6866	bool	r/o	Digital sensor 6 alarm 2

Sensor failure

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1AD3	6867	bool	r/o	Digital sensor 1 sensor failure
1AD4	6868	bool	r/o	Digital sensor 2 sensor failure
1AD5	6869	bool	r/o	Digital sensor 3 sensor failure
1AD6	6870	bool	r/o	Digital sensor 4 sensor failure
1AD7	6871	bool	r/o	Digital sensor 5 sensor failure
1AD8	6872	bool	r/o	Digital sensor 6 sensor failure

Calibration signals

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1AD9	6873	bool	r/o	Digital sensor 1 calibration signal
1ADA	6874	bool	r/o	Digital sensor 2 calibration signal
1ADB	6875	bool	r/o	Digital sensor 3 calibration signal
1ADC	6876	bool	r/o	Digital sensor 4 calibration signal
1ADD	6877	bool	r/o	Digital sensor 5 calibration signal
1ADE	6878	bool	r/o	Digital sensor 6 calibration signal

8 Modbus address tables

Binary values

Modbus PDU address		Data type	Access	Data
Hex.	**Dec.**			
1ADF	6879	bool	r/o	Digital sensor 1 binary value 1
1AE0	6880	bool	r/o	Digital sensor 1 binary value 2
1AE1	6881	bool	r/o	Digital sensor 1 binary value 3
1AE2	6882	bool	r/o	Digital sensor 1 binary value 4
1AE3	6883	bool	r/o	Digital sensor 1 binary value 5
1AE4	6884	bool	r/o	Digital sensor 1 binary value 6
1AE5	6885	bool	r/o	Digital sensor 1 binary value 7
1AE6	6886	bool	r/o	Digital sensor 1 binary value 8
1AE7	6887	bool	r/o	Digital sensor 1 binary value 9
1AE8	6888	bool	r/o	Digital sensor 1 binary value 10
1AE9	6889	bool	r/o	Digital sensor 1 binary value 11
1AEA	6890	bool	r/o	Digital sensor 1 binary value 12
1AEB	6891	bool	r/o	Digital sensor 1 binary value 13
1AEC	6892	bool	r/o	Digital sensor 1 binary value 14
1AED	6893	bool	r/o	Digital sensor 1 binary value 15
1AEE	6894	bool	r/o	Digital sensor 1 binary value 16
1AEF	6895	bool	r/o	Digital sensor 2 binary value 1
1AF0	6896	bool	r/o	Digital sensor 2 binary value 2
1AF1	6897	bool	r/o	Digital sensor 2 binary value 3
1AF2	6898	bool	r/o	Digital sensor 2 binary value 4
1AF3	6899	bool	r/o	Digital sensor 2 binary value 5
1AF4	6900	bool	r/o	Digital sensor 2 binary value 6
1AF5	6901	bool	r/o	Digital sensor 2 binary value 7
1AF6	6902	bool	r/o	Digital sensor 2 binary value 8
1AF7	6903	bool	r/o	Digital sensor 2 binary value 9
1AF8	6904	bool	r/o	Digital sensor 2 binary value 10
1AF9	6905	bool	r/o	Digital sensor 2 binary value 11
1AFA	6906	bool	r/o	Digital sensor 2 binary value 12
1AFB	6907	bool	r/o	Digital sensor 2 binary value 13
1AFC	6908	bool	r/o	Digital sensor 2 binary value 14
1AFD	6909	bool	r/o	Digital sensor 2 binary value 15
1AFE	6910	bool	r/o	Digital sensor 2 binary value 16
1AFF	6911	bool	r/o	Digital sensor 3 binary value 1
1B00	6912	bool	r/o	Digital sensor 3 binary value 2
1B01	6913	bool	r/o	Digital sensor 3 binary value 3
1B02	6914	bool	r/o	Digital sensor 3 binary value 4
1B03	6915	bool	r/o	Digital sensor 3 binary value 5
1B04	6916	bool	r/o	Digital sensor 3 binary value 6
1B05	6917	bool	r/o	Digital sensor 3 binary value 7
1B06	6918	bool	r/o	Digital sensor 3 binary value 8
1B07	6919	bool	r/o	Digital sensor 3 binary value 9
1B08	6920	bool	r/o	Digital sensor 3 binary value 10

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1B09	6921	bool	r/o	Digital sensor 3 binary value 11
1B0A	6922	bool	r/o	Digital sensor 3 binary value 12
1B0B	6923	bool	r/o	Digital sensor 3 binary value 13
1B0C	6924	bool	r/o	Digital sensor 3 binary value 14
1B0D	6925	bool	r/o	Digital sensor 3 binary value 15
1B0E	6926	bool	r/o	Digital sensor 3 binary value 16
1B0F	6927	bool	r/o	Digital sensor 4 binary value 1
1B10	6928	bool	r/o	Digital sensor 4 binary value 2
1B11	6929	bool	r/o	Digital sensor 4 binary value 3
1B12	6930	bool	r/o	Digital sensor 4 binary value 4
1B13	6931	bool	r/o	Digital sensor 4 binary value 5
1B14	6932	bool	r/o	Digital sensor 4 binary value 6
1B15	6933	bool	r/o	Digital sensor 4 binary value 7
1B16	6934	bool	r/o	Digital sensor 4 binary value 8
1B17	6935	bool	r/o	Digital sensor 4 binary value 9
1B18	6936	bool	r/o	Digital sensor 4 binary value 10
1B19	6937	bool	r/o	Digital sensor 4 binary value 11
1B1A	6938	bool	r/o	Digital sensor 4 binary value 12
1B1B	6939	bool	r/o	Digital sensor 4 binary value 13
1B1C	6940	bool	r/o	Digital sensor 4 binary value 14
1B1D	6941	bool	r/o	Digital sensor 4 binary value 15
1B1E	6942	bool	r/o	Digital sensor 4 binary value 16
1B1F	6943	bool	r/o	Digital sensor 5 binary value 1
1B20	6944	bool	r/o	Digital sensor 5 binary value 2
1B21	6945	bool	r/o	Digital sensor 5 binary value 3
1B22	6946	bool	r/o	Digital sensor 5 binary value 4
1B23	6947	bool	r/o	Digital sensor 5 binary value 5
1B24	6948	bool	r/o	Digital sensor 5 binary value 6
1B25	6949	bool	r/o	Digital sensor 5 binary value 7
1B26	6950	bool	r/o	Digital sensor 5 binary value 8
1B27	6951	bool	r/o	Digital sensor 5 binary value 9
1B28	6952	bool	r/o	Digital sensor 5 binary value 10
1B29	6953	bool	r/o	Digital sensor 5 binary value 11
1B2A	6954	bool	r/o	Digital sensor 5 binary value 12
1B2B	6955	bool	r/o	Digital sensor 5 binary value 13
1B2C	6956	bool	r/o	Digital sensor 5 binary value 14
1B2D	6957	bool	r/o	Digital sensor 5 binary value 15
1B2E	6958	bool	r/o	Digital sensor 5 binary value 16
1B2F	6959	bool	r/o	Digital sensor 6 binary value 1
1B30	6960	bool	r/o	Digital sensor 6 binary value 2
1B31	6961	bool	r/o	Digital sensor 6 binary value 3
1B32	6962	bool	r/o	Digital sensor 6 binary value 4
1B33	6963	bool	r/o	Digital sensor 6 binary value 5
1B34	6964	bool	r/o	Digital sensor 6 binary value 6

8 Modbus address tables

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1B35	6965	bool	r/o	Digital sensor 6 binary value 7
1B36	6966	bool	r/o	Digital sensor 6 binary value 8
1B37	6967	bool	r/o	Digital sensor 6 binary value 9
1B38	6968	bool	r/o	Digital sensor 6 binary value 10
1B39	6969	bool	r/o	Digital sensor 6 binary value 11
1B3A	6970	bool	r/o	Digital sensor 6 binary value 12
1B3B	6971	bool	r/o	Digital sensor 6 binary value 13
1B3C	6972	bool	r/o	Digital sensor 6 binary value 14
1B3D	6973	bool	r/o	Digital sensor 6 binary value 15
1B3E	6974	bool	r/o	Digital sensor 6 binary value 16

The data allocation of the variables in the address table varies depending on the sensor or device type. JUMO tecLine sensors do not provide binary values. The following tables show the assignment of binary values for the respective types.

Binary values	Sensor types JUMO digiLine					
	pH	ORP	T	CR/Ci	O-DO S10	O-DO H10/H20[a]
1	pH sensor alarm	ORP sensor alarm	Temperature sensor alarm	Alarm conductivity	Alarm oxygen	Error-Flag oxygen
2	Temperature sensor alarm	Calibration timer alarm	Binary input state	Alarm temperature	Alarm temperature	Error-Flag temperature
3	pH min. warning	Binary input state	-	Compensation range left	Alarm sensor-element (cap missing)	Switch-off temp. exceeded
4	pH min. alarm	-	-	Pre-alarm max. CIP/SIP cycles	Oxygen outside the measuring range	Max. pulse reached
5	pH max. warning	-	-	Alarm max. CIP/SIP cycles	Temperature outside the measuring range	Cap missing
6	pH max. alarm		-	Sensor stress pre-alarm	Warning: State of the sensor element (cap replacement) is recommended	Max. CIP exceeded
7	Min. temperature warning		-	Sensor stress alarm	-	Max. SIP exceeded
8	Min. temperature alarm	-	-	Calibration timer expired	-	-
9	Max. temperature warning	-	-	Operating data acquisition alarm	-	-
10	Max. temperature alarm	-	-	**only for CR:** contamination alarm	-	-
11	Calibration timer alarm	-	-	-	-	-
12	CIP/SIP/Autoclaving warning	-	-	-	-	-
13	CIP/SIP/Autoclaving alarm	-	-	-	-	-
14	Sensor stress warning	-	-	-	-	-
15	Sensor stress alarm	-	-	-	-	-
16	Binary input state	-	-	-	-	-

[a] Available from device software version 304.09.01.

8 Modbus address tables

Binary values	Sensor types JUMO ecoLine	
	O-DO	NTU
1	Warning: Measured value outside the specification	Warning: Measured value outside the specification
2	Warning: Measurement interrupted	Warning: Extraneous light
3	Error: Measurement impossible	Error: Measurement impossible
4	Error: Membrane cap missing	Error: Extraneous light

Binary values	Device type JUMO flowTRANS US W02 digiLine[a]
1	Alarm, flow
2	Temperature alarm
3	Alarm, pressure[b]
4	Empty conduit
5	Air bubbles present

[a] Available from device software version 304.09.01.

[b] Only for devices with an integrated pressure sensor.

Bus status

Modbus PDU address		Data type	Access	Data	
Hex.	Dec.				
1B3F	6975	byte	r/o	digiLine bus status	Meaning of the byte values[a]: 0 = Error (red) 1 = fault (yellow) 2 = Ok (green)

[a] Further information on the bus status can be found in the operating manual for the JUMO AQUIS touch P.

Sensor status

Modbus PDU address		Data type	Access	Data	
Hex.	Dec.				
1B40	6976	byte	r/o	Digital sensor, 1 link status	Meaning of the byte values[a]: 0 = NoLink (red) 1 = Install (yellow) 2 = LinkActive (green)
1B41	6977	byte	r/o	Digital sensor, 2 link status	
1B42	6978	byte	r/o	Digital sensor, 3 link status	
1B43	6979	byte	r/o	Digital sensor, 4 link status	
1B44	6980	byte	r/o	Digital sensor, 5 link status	
1B45	6981	byte	r/o	Digital sensor, 6 link status	

[a] Further information on the link status of digital sensors can be found in the operating manual for the JUMO AQUIS touch P and the respective JUMO digiLine electronics (Type 202705).

Last error code

Modbus PDU address		Data type	Access	Data
Hex.	Dec.			
1B46	6982	word	r/o	Digital sensor 1 last error code
1B47	6983	word	r/o	Digital sensor 2 last error code
1B48	6984	word	r/o	Digital sensor 3 last error code
1B49	6985	word	r/o	Digital sensor 4 last error code
1B4A	6986	word	r/o	Digital sensor 5 last error code
1B4B	6987	word	r/o	Digital sensor 6 last error code

For further information regarding the handling of the error codes listed here, see chapter 2.8.3 "Error codes as integer feedback values", Page 19.

8.4 Modbus frames

8.4.1 Reading frames

address		Data type	Access	Designation
Hex.	Dec.			
8000	32768	Byte[254]	r/w	Frame 1 reading[a]
8080	32896	Byte[254]	r/w	Frame 2 reading[a]
8100	33024	Byte[254]	r/w	Frame 3 reading[a]
8180	33152	Byte[254]	r/w	Frame 4 reading[a]
8200	33280	Byte[254]	r/w	Frame 5 reading[a]
8280	33408	Byte[254]	r/w	Frame 6 reading[a]
8300	33536	Byte[254]	r/w	Frame 7 reading[a]
8380	33664	Byte[254]	r/w	Frame 8 reading[a]

[a] The Modbus frames are available only from device software version 304.08.01 onwards.

8.4.2 Writing frames

address		Data type	Access	Designation
Hex.	Dec.			
8800	34816	Byte[254]	r/o	Frame 1 writing[a]
8880	34944	Byte[254]	r/o	Frame 2 writing[a]
8900	35072	Byte[254]	r/o	Frame 3 writing[a]
8980	35200	Byte[254]	r/o	Frame 4 writing[a]
8A00	35328	Byte[254]	r/o	Frame 5 writing[a]
8A80	35456	Byte[254]	r/o	Frame 6 writing[a]
8B00	35584	Byte[254]	r/o	Frame 7 writing[a]
8B80	35712	Byte[254]	r/o	Frame 8 writing[a]

[a] The Modbus frames are available only from device software version 304.08.01 onwards.

8 Modbus address tables

JUMO GmbH & Co. KG
Street address:
Moritz-Juchheim-Straße 1
36039 Fulda, Germany
Delivery address:
Mackenrodtstraße 14
36039 Fulda, Germany
Postal address:
36035 Fulda, Germany
Phone: +49 661 6003-0
Fax: +49 661 6003-607
Email: mail@jumo.net
Internet: www.jumo.net

JUMO UK LTD
JUMO House
Temple Bank, Riverway
Harlow, Essex, CM20 2DY, UK
Phone: +44 1279 63 55 33
Fax: +44 1279 62 50 29
Email: sales@jumo.co.uk
Internet: www.jumo.co.uk

JUMO Process Control, Inc.
6724 Joy Road
East Syracuse, NY 13057, USA
Phone: +1 315 437 5866
Fax: +1 315 437 5860
Email: info.us@jumo.net
Internet: www.jumousa.com

