

JUMO digiLine O-DO H10/H20

Digital optical sensors for dissolved oxygen
in hygienic applications



Modbus interface description



20261200T92Z001K000

V1.00/EN/2025-04-16

Further information and downloads



qr-202612-en.jumo.info

Table of contents

1	About this documentation	5
1.1	Validity	5
1.2	Applicable documentation	5
1.3	Purpose	5
1.4	Target group	5
1.5	Abbreviations	5
1.6	Trademark information	5
1.7	Definition of terms	6
1.8	Symbols	6
2	Modbus protocol description	7
2.1	Master-slave principle	7
2.2	Transmission media for Modbus	8
2.3	Structure of an RTU Modbus telegram	8
2.4	Function codes	8
2.4.1	Reading n words	9
2.4.2	Writing one word	10
2.4.3	Writing n words	11
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	17
2.8.1	Modbus error codes	17
2.8.2	Error messages for invalid values	18
3	Interfaces	19
3.1	Interface assignment	19
3.1.1	Terminating resistors	20
4	Configuring interfaces	21
5	Modbus over a serial interface	22
5.1	Modbus slave operation via RS485 serial interface	22
6	Modbus address tables	24
6.1	General digiLine Modbus addresses	24
6.1.1	Nameplate	24
6.1.2	Manufacturer's data	24
6.1.3	Measuring point info	25
6.1.4	Configuration of the interface	25

Table of contents

6.1.5	Operating data	26
6.1.6	Calibration data	27
6.2	Address tables for use cases	38
6.2.1	Reading detailed measured values	38
6.2.2	Measurement configuration	39
6.2.3	New sensor/changing sensor values manually	40
6.2.4	Simple automatic calibrations	41
6.2.5	SIP and CIP settings	42
6.2.6	Measurement and operating temperatures	42
6.2.7	Configuring analog 4 to 20 mA interface	43
6.2.8	Calibrating internal temperature probes	44
6.2.9	Configuring the measurements counter	44
6.2.10	Error evaluation with status word	44
6.2.11	Data for the optical sensor cap	45

1 About this documentation

1.1 Validity

This manual is valid for all device software versions and hardware versions.

1.2 Applicable documentation

This document is supplemented by the documents listed below:

Product group	Document type
202612	Operating manual
203599	JUMO DSM software operating manual

The documents listed are available online on our website **www.jumo.de**.

Enter the number of the product group or the title of the document in the search field.

1.3 Purpose

This documentation describes the technical details and configuration of the serial interface's Modbus RTU mode for data transmission between devices and also explains the requisite registers on the basis of use cases.

Contact the manufacturer if you have any questions about the device and documentation.

1.4 Target group

This documentation is intended for specialists and users who are familiar with the configuration and operation of Modbus-capable devices.

1.5 Abbreviations

Abbreviation	Meaning
RTU	Remote Terminal Unit
RS485	Two-wire interface with differential signal, where the request and response are transmitted on the same wire pair

1.6 Trademark information

All trademarks and trade and company names used are the property of their rightful owners or authors.

1 About this documentation

1.7 Definition of terms

Use in document	Definition
Master	Interface user that is authorized to send requests or commands to slaves.
Slave	Interface user that responds to a master's requests or commands.

1.8 Symbols



DANGER!

The signal word **"DANGER"** indicates an immediate danger.

Non-observance will lead to death or serious injury.

- The instructions in the warning notice must be observed and followed!
-



WARNING!

The signal word **"WARNING"** indicates an imminent danger.

Non-observance can lead to death or serious injury.

- The instructions in the warning notice must be observed and followed!
-



CAUTION!

The signal word **"CAUTION"** indicates an imminent danger.

Non-observance can lead to minor or moderate injury.

- The instructions in the warning notice must be observed and followed!
-

NOTICE!

The signal word **"NOTICE"** indicates possible damage to property.

Non-observance can lead to damage to devices, systems or the environment.

- Observe the instructions in the note for avoiding damage!
-

NOTE!

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



REFERENCE!

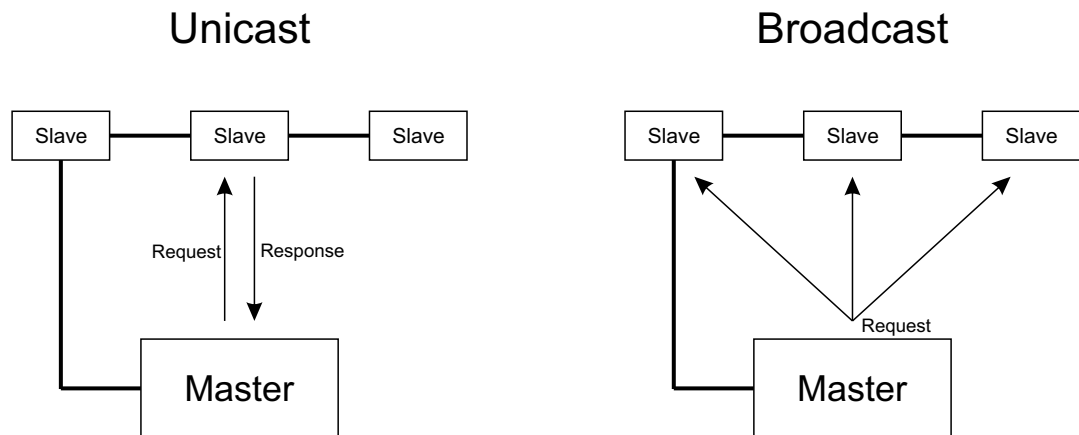
This symbol refers to **further information** in other sections, chapters, or other manuals.



2.1 Master-slave principle

Modbus communication between a master, such as a SCADA system or PLC, and a JUMO digiLine O-DO H10/H20 as a slave according to the master-slave principle. During this process, the master sends data requests or instructions that the slave then answers with a response. Slaves are identified by their device address, while master devices do not need an address.

⇒ chapter 5 "Modbus over a serial interface", Page 22



The master controls data exchange in the bus by cyclically sending requests to the slaves, such as the JUMO digiLine O-DO H10/H20, which are only able to respond. The master has read and write access to the slaves, which enables real-time communication between master and slaves. Slaves cannot communicate with one another directly; the master must extract data from one slave and transfer it to another.

The master sends its requests specifically to individual slaves using their **Unicast** addresses. Alternatively, requests can be sent to all slaves in the bus as a **broadcast**, using the broadcast address. To avoid data collisions in serial bus systems, slaves do not respond to broadcast requests. For this reason, broadcast addresses are only appropriate for function codes for writing data, not for reading data.



NOTE!

The JUMO digiLine O-DO H10 and H20 sensors can only be operated as slaves.

2 Modbus protocol description

2.2 Transmission media for Modbus

Serial interface

Modbus specifications consider **RTU (Remote Terminal Unit)** and ASCII modes for **serial data transmission**. The JUMO digiLine O-DO H10/H20 only supports **RTU mode**, in which data is transferred via the serial interface (RS485) in binary format.

⇒ chapter 5 "Modbus over a serial interface", Page 22

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. 125 words (250 bytes)
06	6	Writing one word	Max. 1 word (2 bytes)
10	16	Writing n words	Max. 123 words (246 bytes)

2 Modbus protocol description

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 calibration date

The calibration date (February 10, 2025) is saved in three consecutive addresses for the year, month and day. For this purpose, 3 words (6 bytes) have to be loaded.

Hex code for the data request:

01	03	30 0A	00 03	2A C9
Slave	Function	Address of 1st word	Number of words	CRC

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

01	03	06	07 E9 00 02 00 0A	5C D2
Slave	Function	Bytes read	Date (YYYY MM DD) in hex format	CRC

2 Modbus protocol description

2.4.2 Writing one word

In the case of this function, the data blocks are identical for the instruction and response.

NOTICE!

Write operations on specific R/W parameters save data in the EEPROM or flash memory. These memory modules have a limited number of write cycles (approx. 100,000 or 10,000).

Frequent writing can cause a memory error.

- Keep the number of write operations to a minimum.
-

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: Setting up calibration mode

Here, the calibration mode at word address 0x3009 should be set to "humid" (value = 1) for the sensor with device address 1.

Hex code for the instruction:

01	06	30 09	00 01	97 08
Slave	Function	Word address	Value	CRC

Hex code for the response:

01	06	30 09	00 01	97 08
Slave	Function	Word address	Value	CRC

2 Modbus protocol description

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 val- ue(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: Activating the 4 to 20 mA interface

From word address 0x207D, the interface is activated with the value (command) "02" and set to "linear". (chapter 6.2.7 "Configuring analog 4 to 20 mA interface", Page 43). In this example, the sensor's slave address should be "1".

Hex code for the instruction:

01	10	20 7D	00 02	04	00 00 00 02	2D 12
Slave	Function	Address 1st word	Number of words	Number of bytes	Value	CRC

Hex code for the response:

01	10	20 7D	00 02	DA 10
Slave	Function	Address 1st word	Number of words	CRC

2 Modbus protocol description

2.5 Data types

Data type	Description	Access	Possible function codes	Number of Modbus registers																												
uint8 int8 bool	Low byte (8-bit) of a word as an integer value; the high byte is not used. Value range: 0 to 255 (uint8) -128 to 127 (int8) For interpretation as bool: lowest significant bit in a word as bit value 0000 0000 0000 0001 = 1 or TRUE 0000 0000 0000 0000 = 0 or FALSE	read only	03, 04	1																												
		read/write	03, 04, 06, 16																													
uint16 int16	Word (16-bit) as integer value Value range: 0 to 65535 (uint16) -32768 to 32767 (int16)	read only	03, 04	1																												
		read/write	03, 04, 06, 16																													
uint32 int32	Double word (32-bit) as integer value Value range: 0 to 4294967295 (uint32) -2147483648 to 2147483647 (int32)	read only	03, 04	2																												
		read/write	03, 04, 16																													
enum	Word (16-bit) as integer value for consecutive numbering of settings Value range: 0 to 65535	read only	03, 04	1																												
		read/write	03, 04, 06, 16																													
float	2 words as 32-bit floating-point number with encoding according to IEEE 754, where the correct order of the 4 bytes' transmission must be observed. When configuring the RS485 interface ⇒chapter 6.1.4 "Configuration of the interface", Page 25, the desired floating-point format can be selected. S = Prefix sign bit E = Exponent (two's complement) M = 23-bit normalized mantissa IEEE 754 Big Endian <table><tr><td>Byte 1</td><td>Byte 2</td><td>Byte 3</td><td>Byte 4</td></tr><tr><td>SE7EEEEEE1</td><td>E0M23MMMMMM16</td><td>M15MMMMMMM8</td><td>M7MMMMMMM0</td></tr></table> IEEE 754 Little Endian <table><tr><td>Byte 1</td><td>Byte 2</td><td>Byte 3</td><td>Byte 4</td></tr><tr><td>M7MMMMMMM0</td><td>M15MMMMMMM8</td><td>E0M23MMMMMM16</td><td>SE7EEEEEE1</td></tr></table> Standard modbus coding <table><tr><td>Byte 1</td><td>Byte 2</td><td>Byte 3</td><td>Byte 4</td></tr><tr><td colspan="2">1. Modbus register</td><td colspan="2">2. Modbus register</td></tr><tr><td>M15MMMMMMM8</td><td>M7MMMMMMM0</td><td>SE7EEEEEE1</td><td>E0M23MMMMMM16</td></tr></table>	Byte 1	Byte 2	Byte 3	Byte 4	SE7EEEEEE1	E0M23MMMMMM16	M15MMMMMMM8	M7MMMMMMM0	Byte 1	Byte 2	Byte 3	Byte 4	M7MMMMMMM0	M15MMMMMMM8	E0M23MMMMMM16	SE7EEEEEE1	Byte 1	Byte 2	Byte 3	Byte 4	1. Modbus register		2. Modbus register		M15MMMMMMM8	M7MMMMMMM0	SE7EEEEEE1	E0M23MMMMMM16	read only	03, 04	2
		Byte 1	Byte 2	Byte 3	Byte 4																											
		SE7EEEEEE1	E0M23MMMMMM16	M15MMMMMMM8	M7MMMMMMM0																											
		Byte 1	Byte 2	Byte 3	Byte 4																											
		M7MMMMMMM0	M15MMMMMMM8	E0M23MMMMMM16	SE7EEEEEE1																											
		Byte 1	Byte 2	Byte 3	Byte 4																											
		1. Modbus register		2. Modbus register																												
		M15MMMMMMM8	M7MMMMMMM0	SE7EEEEEE1	E0M23MMMMMM16																											
		read/write	03, 04, 16																													

2 Modbus protocol description

Data type	Description	Access	Possible function codes	Number of Modbus registers
string[n]	Character string with a maximum length of n bytes. The character encoding of the variables is specified in the Modbus address tables in the last chapter of this document. UTF-8, ISO 8859-1 or ASCII can be used for character set encoding. Each of the n Modbus register words contains 2 consecutive bytes from the character string. The character string must always be concluded with "\0" (hexadecimal: 0x00) as the end coding. Furthermore, the character string's total length must have an even number of bytes. If the number of bytes is odd, an additional "\0" must be used in addition to the first "\0" to compensate for the length.	read only	03, 04	n/2
		read/write	03, 04, 06, 16	

2 Modbus protocol description

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

Slaveaddress	Function 0x03 or 0x04	Address of first word	Number of words	Checksum CRC
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Integer values are transferred via Modbus in the following format:
First the high, then the low byte.

Response

Slaveaddress	Function 0x03 or 0x04	Number of 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 number of CIP cycles since the last calibration needs to be extracted at Uint32 address 0x2061 (chapter 6.2.5 "SIP and CIP settings", Page 42). The value here should be "25" (word value 0x00000019).

Data request:

01	03	20 61	00 02	9E 15
Slave	Function	Address of 1st word	Number of words	CRC

Response (values in Modbus floating-point format):

01	03	04	00 00 00 19	3B F9
Slave	Function	Bytes read	Integer value	CRC

2 Modbus protocol description

2.6.2 Float values

For floating-point values, this device uses the IEEE 754 format (32-bit) in the standard Modbus encoding, whereby bytes 1 and 2 are swapped with bytes 3 and 4. This setting can be reconfigured if necessary.

Single floating-point format (32-bit) according to standard IEEE 754

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

S - Prefix 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 "compensation temperature" value needs to be extracted at the sensor address 0x2013. The value here should be 20.25 (0x41A20000 in IEEE 754 format).

Data request:

01	03	20 13	00 02	3E 0E
Slave	Function	Address of 1st word	Number of words	CRC

Response (values in Modbus floating-point format):

01	03	04	00 00	41 A2	4B DA
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. Conventional standardized floating-point formats can be selected in the settings of the serial interface. However, under certain circumstances it may be necessary to adjust the byte sequence by making changes in the interface program of the master.

2 Modbus protocol description

2.6.3 Character strings (texts)

Character strings are encoded in UTF8 for transmission.



NOTE!

A "\0" (hexadecimal: 0x00) must always be transmitted as the terminating code. Subsequent characters have no meaning.

Since texts are transmitted word by word (16-bit register), they must always have an even number of bytes.

The maximum lengths for character strings specified in the address tables contain the final "\0". This means that, in the case of "string[20]", the text, including "\0", must not be longer than 20 bytes.

Example:

Requesting of the text from address 0x2136. From this address onwards, the character string "00123456" (ASCII code: 0x30 0x30 0x31 0x32 0x33 0x34 0x35 0x36) stands for the part number (material number) of the optical sensor cap.

Hex code for the request:

01	03	21 36	00 06	2F FA
Slave	Function	Address 1st word	Number of words	CRC

Hex code for the response:

01	03	00 0C	30 30 31 32 33 34 35 36 00 FF FF FF	44 55
Slave	Function	Bytes read	Hex word values of the ASCII characters	CRC



NOTE!

The values (here: 00 FF FF FF) before the CRC sum (here: 44 55) are not taken into consideration, since they are after the end 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
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.

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 this device are listed in chapter 2.4 "Function codes", Page 8.
- 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 Modbus protocol description

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

Loading of the measured oxygen value at Modbus address 0x2001:

Data request:

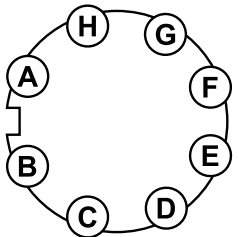
01	03	20 01	00 02	9E 0B
Slave	Function	Word address	Number of words	CRC

Response:

01	03	04	BD C2	7D 70	5F 17
Slave	Function	Bytes read	Error code	CRC	

The measured value provided by the analog input 0x7D70BDC2 ($=2.0 \times 10^{37}$) indicates that an over-range situation exists.

3.1 Interface assignment

Function	Pin	Figure Sensor connection
NC	A	
4 to 20 mA current sink	B	
+24 V	C	
GND	D	
NC	E	
NC	F	
RS485 (RxD/TxD+)	G	
RS485 (RxD/TxD-)	H	



NOTE!

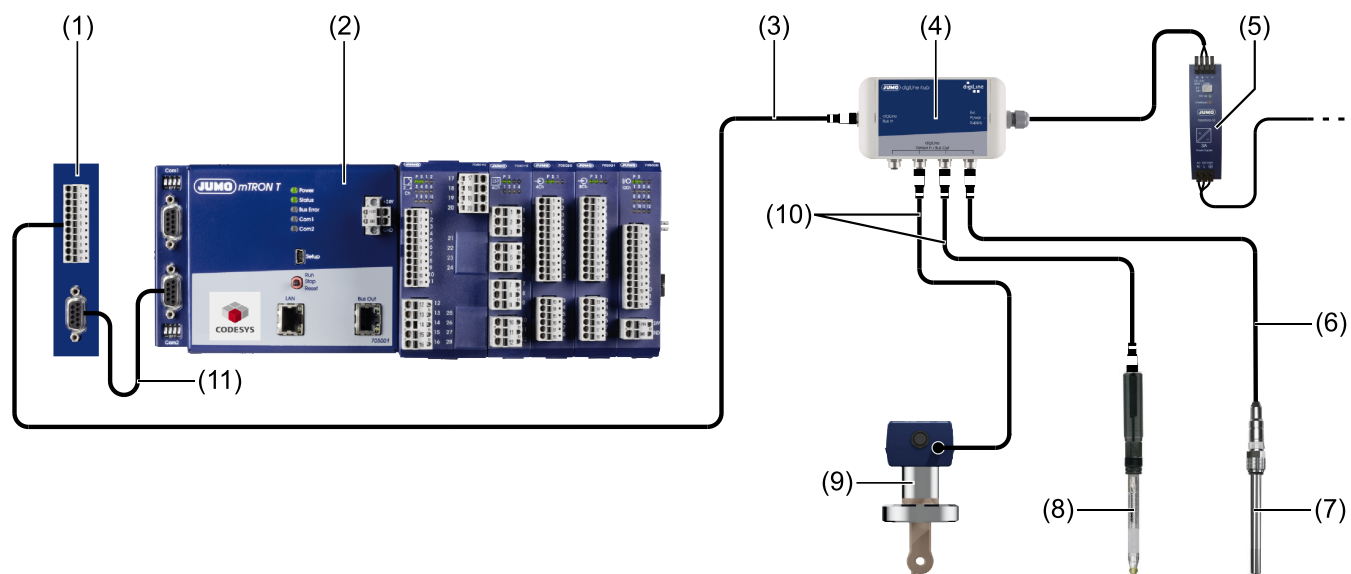
Preassembled connecting cables are available for connecting sensors to JUMO digiLine electronics. Use these to assure a reliable cable connection and data transmission in the bus system. In the order details provided in the operating manual, you can find a list of cables and connectors available as accessories.

3 Interfaces

3.1.1 Terminating resistors

A JUMO digiLine bus cable needs to be terminated with terminating connectors at the ends distant from the master unit (part no.: 00461591). These are simply connected to the last free M12 coupling at the final Y-splitter (part no.: 00638327) in the bus. At the master unit end of the bus cable, the RS485 bus cable must be terminated according to the operating manual for the Modbus master device.

Connection example for 2 sensors with JUMO digiLine electronic components on a JUMO mTRON T



- 1 Interface module for digiLine master connecting cable, with screw terminals and 9-pole D-sub pin header
- 2 mTRON T central processing unit with RS485 interface as digiLine master (Modbus master)
- 3 JUMO M12 digiLine master connecting cable **for 705001** with exposed wire ends at one end for connection to devices with screw or spring-cage terminals
- 4 JUMO digiLine hub
- 5 Separate DC 24 V power supply unit for the voltage supply to the JUMO digiLine bus system
- 6 VarioPin adapter cable to M12, 5-pole
- 7 JUMO digiLine O-DO H10/H20 – optical sensor for dissolved oxygen in hygienic applications
- 8 JUMO digiLine pH – sensor for hygienic applications
- 9 JUMO digiLine Ci – head transmitter for hygienic applications
- 10 JUMO M12 connecting cable, 5-pole and A-coded
- 11 D-sub connection cable

4 Configuring interfaces

For all user devices on a bus to be able to communicate with one another, their interface settings must be correct. The table in chapter 6.1.4 "Configuration of the interface", Page 25 shows the setting options for the serial interfaces on the JUMO digiLine electronic components.

While the JUMO digiLine protocol automatically issues the interface parameters during startup (Plug and Play), **the interface parameters must be configured using JUMO DSM software prior to initial startup for operation on a Modbus master device.**

NOTICE!

Unexpected plant operating statuses possible due to incorrect installation or incorrect settings for equipment.

Failure to observe this information can cause damage or disrupt processes from functioning properly.

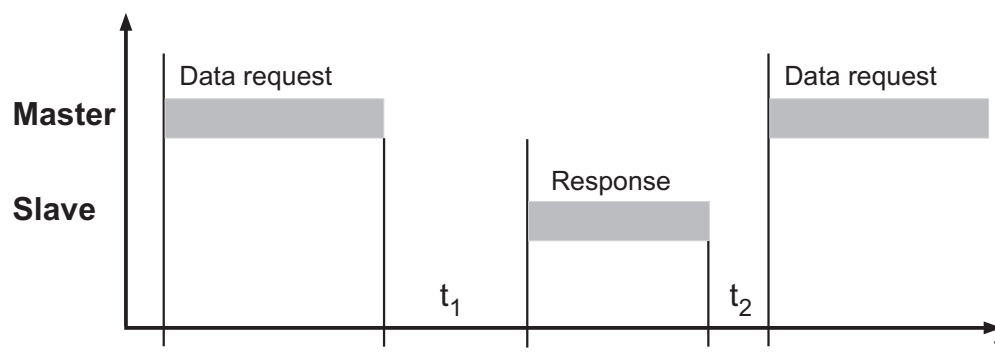
- Make sure that independent safety equipment is always available and that settings are always configured by qualified personnel.
-

5 Modbus over a serial interface

5.1 Modbus slave operation via RS485 serial interface

Chronological sequence of communication

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



t_1	Device's internal waiting period prior to checking of the data request and the internal processing time: min.: 3.5 byte times + minimum response time typically: 5 ms max.: 25 ms + minimum response time
t_2	Waiting period that the master must observe before starting a new data request: 3.5 characters or at least the time needed for the slowest bus user to check the response, to reject it and to be ready to receive requests again.



NOTE!

The minimum response time can be set in the configuration in order to adjust the waiting period t_1 to the slowest bus user.

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.



NOTE!

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



NOTE!

The terminating code after a data request or data response is 3.5 characters long. The time needed for these 3.5 characters depends on the baud rate.

5 Modbus over a serial interface

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.

The following results for a data format of 8 data bits, no parity bit, and one stop bit:

$$\text{Character transmission time [ms]} = 1000 \times 10 \text{ bits} \div \text{baud rate}$$

For other data formats, the following is the result:

$$\text{Character transmission time [ms]} = 1000 \times 11 \text{ bits} \div \text{baud rate}$$

Example

Identifier for end of data request or end of response

Waiting period = 3.5 characters * 1000 * 11 bits ÷ baud rate

Baud rate [baud]	Data format [bit]	Character transmission time[ms]
38400	11	0.287
	10	0.260
19200	11	0.573
	10	0.521
9600	11	1.146
	10	1.042

6 Modbus address tables

6.1 General digiLine Modbus addresses

6.1.1 Nameplate

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
0	0000	string[10]	5	r	Device name , ASCII coding with "\0" as end code
5	0005	string[12]	6	r	Software version , ASCII coding with "\0" as end code
11	000B	uint16	1	r	Software number (basic number)
12	000C	uint8	1	r	Major version Sensor
13	000D	uint8	1	r	Minor version Sensor
14	000E	uint16	1	r	Trial version
15	000F	uint8	1	r	Major version Hardware
16	0010	uint8	1	r	Minor version Hardware
17	0011	string[14]	7	r	VDN version , ASCII coding with "\0" as end mark
24	0018	uint16	1	r	VDN number
25	0019	uint16	1	r	VDN version number

6.1.2 Manufacturer's data

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
256	0100	string[20]	10	r	Serial number , ASCII coding with "\0" as end code
266	010A	enum	1	r	Sensor type , enumeration: • 0 = not defined • 1 = JUMO digiLine pH/ORP/T • 2 = JUMO digiLine CR/Ci • 3 = JUMO digiLine O-DO H10/H20 • 4 = JUMO digiLine O-DO S10 • 5 = not defined
267	010B	enum	1	r	Oxygen sensor sub-type , enumeration: • 0 = Standard measuring range (JUMO digiLine O-DO H10) • 1 = Trace measuring range (JUMO digiLine O-DO H20)
273	0111	string[12]	6	r	Part number , ASCII coding with "\0" as end code
279	0117	string[64]	32	r	Order code , ASCII coding with "\0" as end code
335	014F	uint32	2	r	Hardware address 24 bit
347	015B	uint16	1	r	Date of manufacture, year
348	015C	int8	1	r	Date of manufacture, month
349	015D	int8	1	r	Date of manufacture, day
350	015E	int8	1	r	Date of manufacture, hour
351	015F	int8	1	r	Date of manufacture, minute
352	0160	int8	1	r	Date of manufacture, second
353	0161	int16	1	r	Date of manufacture, time zone
354	0162	uint16	1	r	Calibration status

6 Modbus address tables

6.1.3 Measuring point info

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
512	0200	string[60]	30	r/w	Tag number , UTF8 coding with "\0" as end code The number of bytes must be even (if necessary, append an additional "0")
542	021E	string[192]	96	r/w	Description , UTF8 coding with "\0" as end code The number of bytes must be even (if necessary, append an additional "0")
638	027E	string[110]	55	r/w	Sensor origin , UTF8 coding with "\0" as end code This variable is managed automatically by the JUMO digiLine master and may not be described otherwise! The number of bytes must be even (if necessary, append an additional "0")

6.1.4 Configuration of the interface

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
768	0300	uint8	1	r/w	Device address , unambiguous specification of a bus user 0 = Broadcast address^a 1 to 247 = Unicast addresses^b
769	0301	enum	1	r/w	Baud rate , enumeration: 0 = 9600 1 = 19200 2 = 38400 Must match for all bus users
770	0302	enum	1	r/w	Data format , enumeration: 0 = 8 – 1 – no parity 1 = 8 – 1 – odd parity 2 = 8 – 1 – even parity 3 = 8 – 2 – no parity Must match for all bus users
773	0305	uint16	1	r/w	Minimum response time , used to adjust the sensor's response speed to slower bus users Value range: 0 to 500 ms
775	0307	enum	1	r/w	Floating-point format , enumeration: 0 = Standard Modbus format 1 = IEEE 754 Little Endian 2 = IEEE 754 Big Endian

^a The broadcast address is intended for broadcast messages and must not be used as a slave address.

^b Unicast addresses serve as unique slave addresses that allow the master to address specific slave devices.

6 Modbus address tables

6.1.5 Operating data



NOTE!

A timeout period of 250 ms is required after **all** write functions.

An extended timeout period of 650 ms is required after write functions for the addresses 0x0402, 0x2031 et. seq., 0x3003 and 0x3005.

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1024	0400	uint32	2	r	Operating hours , current counter status in seconds
1026	0402	uint16	1	r/w	Sensor change , current counter status for sensor cap changes

Initial startup

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1280	0500	uint16	1	r	Year of initial startup
1281	0501	uint8	1	r	Month of initial startup
1282	0502	uint8	1	r	Day of initial startup
1283	0503	uint8	1	r	Hour of initial startup
1284	0504	uint8	1	r	Minute of initial startup
1285	0505	uint8	1	r	Second of initial startup
1286	0506	int16	1	r	Time zone of initial startup, offset to UTC time in minutes
1287	0507	uint32	2	r	Status of operating hours counter at time of initial startup, in seconds

Drag indicator function

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1289	0509	float	2	r	Lowest temperature value
1291	050B	float	2	r	Highest temperature value

Date of lowest temperature value

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1293	050D	uint16	1	r	Year
1294	050E	uint8	1	r	Month
1295	050F	uint8	1	r	Day
1296	0510	uint8	1	r	Hour
1297	0511	uint8	1	r	Minute
1298	0512	uint8	1	r	Second
1299	0513	int16	1	r	Time zone , offset to UTC time in minutes
1300	0514	uint32	2	r	Status of operating hours counter at time of lowest temperature value, in seconds

6 Modbus address tables

Date of highest temperature value

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1302	0516	uint16	1	r	Year
1303	0517	uint8	1	r	Month
1304	0518	uint8	1	r	Day
1305	0519	uint8	1	r	Hour
1306	051A	uint8	1	r	Minute
1307	051B	uint8	1	r	Second
1308	051C	int16	1	r	Time zone , offset to UTC time in minutes
1309	051D	uint32	2	r	Status of operating hours counter at time of highest temperature value, in seconds

6.1.6 Calibration data

A calibration logbook with 10 entry data records is stored in the electronic components. The addresses of the respective data records are listed in the following tables.

Since all data records have the same storage capacity, addresses can also be calculated. The address range of one data record is 0x0031 in hexadecimal or 49 in decimal notation. Using multiples of this value, jumps between different data records can be calculated.

Example: $0 \times 0600 + (n - 1) \times 0 \times 0031$ yields the Modbus address of the parameter "Entry type" in the n-th data record.

Calibration logbook entry 1

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1536	0600	enum	1	r	Entry type , enumeration: • 0 = No entry • 20 = digiLine ODO Hxx zero point • 21 = digiLine ODO Hxx end value
1537	0601	enum	1	r	Manual change , enumeration: • 0 = Manual • 1 = Calibration
1538	0602	enum	1	r	Evaluation , enumeration: • 0 = No evaluation • 1 = OK • 2 = Warning
1539	0603	uint32	2	r	Operating hours counter , counter status in seconds
1541	0605	uint16	1	r	Year of calibration
1542	0606	uint8	1	r	Month of calibration
1543	0607	uint8	1	r	Day of calibration
1544	0608	uint8	1	r	Hour of calibration
1545	0609	uint8	1	r	Minute of calibration
1546	060A	uint8	1	r	Second of calibration
1547	060B	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1548	060C	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1549	060D	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1561	0619	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 2

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1585	0631	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1586	0632	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1587	0633	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1588	0634	uint32	2	r	Operating hours counter , counter status in seconds
1590	0636	uint16	1	r	Year of calibration
1591	0637	uint8	1	r	Month of calibration
1592	0638	uint8	1	r	Day of calibration
1593	0639	uint8	1	r	Hour of calibration
1594	063A	uint8	1	r	Minute of calibration
1595	063B	uint8	1	r	Second of calibration
1596	063C	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1597	063D	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1598	063E	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1610	064A	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 3

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1634	0662	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1635	0663	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1636	0664	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1637	0665	uint32	2	r	Operating hours counter , counter status in seconds
1639	0667	uint16	1	r	Year of calibration
1640	0668	uint8	1	r	Month of calibration
1641	0669	uint8	1	r	Day of calibration
1642	066A	uint8	1	r	Hour of calibration
1643	066B	uint8	1	r	Minute of calibration
1644	066C	uint8	1	r	Second of calibration
1645	066D	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1646	066E	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1647	066F	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1659	067B	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 4

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1683	0693	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1684	0694	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1685	0695	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1686	0696	uint32	2	r	Operating hours counter , counter status in seconds
1688	0698	uint16	1	r	Year of calibration
1689	0699	uint8	1	r	Month of calibration
1690	069A	uint8	1	r	Day of calibration
1691	069B	uint8	1	r	Hour of calibration
1692	069C	uint8	1	r	Minute of calibration
1693	069D	uint8	1	r	Second of calibration
1694	069E	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1695	069F	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1696	06A0	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1708	06AC	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 5

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1732	06C4	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1733	06C5	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1734	06C6	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1735	06C7	uint32	2	r	Operating hours counter , counter status in seconds
1737	06C9	uint16	1	r	Year of calibration
1738	06CA	uint8	1	r	Month of calibration
1739	06CB	uint8	1	r	Day of calibration
1740	06CC	uint8	1	r	Hour of calibration
1741	06CD	uint8	1	r	Minute of calibration
1742	06CE	uint8	1	r	Second of calibration
1743	06CF	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1744	06D0	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1745	06D1	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1757	06DD	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 6

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1781	06F5	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1782	06F6	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1783	06F7	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1784	06F8	uint32	2	r	Operating hours counter , counter status in seconds
1786	06FA	uint16	1	r	Year of calibration
1787	06FB	uint8	1	r	Month of calibration
1788	06FC	uint8	1	r	Day of calibration
1789	06FD	uint8	1	r	Hour of calibration
1790	06FE	uint8	1	r	Minute of calibration
1791	06FF	uint8	1	r	Second of calibration
1792	0700	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1793	0701	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1794	0702	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1806	070E	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 7

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1830	06F5	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1831	06F6	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1832	06F7	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1833	06F8	uint32	2	r	Operating hours counter , counter status in seconds
1835	06FA	uint16	1	r	Year of calibration
1836	06FB	uint8	1	r	Month of calibration
1837	06FC	uint8	1	r	Day of calibration
1838	06FD	uint8	1	r	Hour of calibration
1839	06FE	uint8	1	r	Minute of calibration
1840	06FF	uint8	1	r	Second of calibration
1841	0700	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1841	0701	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1843	0702	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1855	070E	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 8

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1879	0757	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1880	0758	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1881	0759	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1882	075A	uint32	2	r	Operating hours counter , counter status in seconds
1884	075C	uint16	1	r	Year of calibration
1885	075D	uint8	1	r	Month of calibration
1886	075E	uint8	1	r	Day of calibration
1887	075F	uint8	1	r	Hour of calibration
1888	0760	uint8	1	r	Minute of calibration
1889	0761	uint8	1	r	Second of calibration
1890	0762	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1891	0763	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1892	0764	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1904	0770	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 9

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1928	0788	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1929	0789	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1930	078A	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1931	078B	uint32	2	r	Operating hours counter , counter status in seconds
1933	078D	uint16	1	r	Year of calibration
1934	078E	uint8	1	r	Month of calibration
1935	078F	uint8	1	r	Day of calibration
1936	0790	uint8	1	r	Hour of calibration
1937	0791	uint8	1	r	Minute of calibration
1938	0792	uint8	1	r	Second of calibration
1939	0793	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1940	0794	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1941	0795	float[6]	12	r	Calibration values , for entry type 20 (zero point): 0 = Phase 1 = Temperature 2 = Calibration mode for entry type 21 (end value): 0 = Phase 1 = Temperature 2 = Calibration mode 3 = Oxygen concentration 4 = Pressure
1953	07A1	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

Calibration logbook entry 10

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1977	07B9	enum	1	r	Entry type , enumeration: 0 = No entry 20 = digiLine ODO Hxx zero point 21 = digiLine ODO Hxx end value
1978	07BA	enum	1	r	Manual change , enumeration: 0 = Manual 1 = Calibration
1979	07BB	enum	1	r	Evaluation , enumeration: 0 = No evaluation 1 = OK 2 = Warning
1980	07BC	uint32	2	r	Operating hours counter , counter status in seconds
1982	07BE	uint16	1	r	Year of calibration
1983	07BF	uint8	1	r	Month of calibration
1984	07C0	uint8	1	r	Day of calibration
1985	07C1	uint8	1	r	Hour of calibration
1986	07C2	uint8	1	r	Minute of calibration
1987	07C3	uint8	1	r	Second of calibration
1988	07C4	int16	1	r	Time zone of calibration , offset to UTC time in minutes
1989	07C5	uint8	1	r	Sensor counter , status of sensor change counter at time of calibration

6 Modbus address tables

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
1990	07C6	float[6]	12	r	Calibration values, for entry type 20 (zero point): • 0 = Phase • 1 = Temperature • 2 = Calibration mode for entry type 21 (end value): • 0 = Phase • 1 = Temperature • 2 = Calibration mode • 3 = Oxygen concentration • 4 = Pressure
2002	07D2	string[6][8]	24	r	Units of calibration values A unit in text form is stored for each of the 6 calibration values. Up to 8 bytes, including termination (end code: \0), are available for this purpose for each unit text. The texts are UTF8-coded and, when 5 characters long, are permitted to contain a maximum of 2 UTF8 characters in 2-byte coding or 1 character in 3-byte coding.

6 Modbus address tables

6.2 Address tables for use cases



NOTE!

A timeout period of 250 ms is required after **all** write functions.

An extended timeout period of 650 ms is required after write functions for the addresses 0x0402, 0x2031 et. seqq., 0x3003 and 0x3005.

6.2.1 Reading detailed measured values

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
8193	2001	float	2	r	Calculated oxygen value
8195	2003	float	2	r	Measured/set temperature in °C
8197	2005	uint32	2	r	Error register (bit-coded, see table 1)
8199	2007	float	2	r	Set pressure value in hPa
8201	2009	float	2	r	Raw reference amplitude in µV
8203	200B	float	2	r	Raw measured oxygen amplitude in µV
8205	200D	float	2	r	Measured oxygen phase shift in µV

Table 1: Error codes for error register

Bit	Error	Description
0	Reference channel overflow	Reference amplitude too high. Reduce reference current.
1	Temperature too high	The current temperature is too high to take a measurement.
2	Reference photodetector overflow A	Reference amplitude is too low or too high. Adjust reference current. This error can also occur when the sensor cap is missing or the ambient light is too strong.
3	Signal channel overflow	Signal amplitude is too high. Reduce reference current.
4	Reserved	
5	Reserved	
6	No sensor calculation/amplitude too low	Signal amplitude is too low. Adjust signal current. This error can also occur when the sensor cap is missing.
7	Measurement counter overflow	The measurement counter has reached its maximum value.
8	Reference amplitude outside admissible limits	Reference amplitude is too low or too high. Adjust reference current. This error can also occur when the sensor cap is missing.
9	Signal photodetector overflow	Signal amplitude is too high. Adjust signal current. This error can also occur when the sensor cap is missing.
10	Reference photodetector overflow B	Reference amplitude is too low or too high. Adjust reference current. This error can also occur when the sensor cap is missing or the ambient light is too strong.
11	Storage write error detected	A storage write error has occurred. Please contact JUMO Service.
12	PME timeout	Internal error. Please contact JUMO Service.

6 Modbus address tables

6.2.2 Measurement configuration

Address		Data type	Number of registers	Access	Description	Value range
Dec.	Hex.					
8209	2011	uint32	2	r/w	Oxygen unit used for measurements. See table 2 for admissible values (codes). Not all units can be selected for every combination. See table 3 for more information.	-
8211	2013	float	2	r/w	Temperature value used for temperature compensation. <u>Wird nicht in den Flash-Speicher geschrieben.</u>	0 to 99.99
8213	2015	float	2	r/w	Salt content value used for salt content compensation.	0 to 999.9
8215	2017	float	2	r/w	Volatile pressure compensation value (in mbar) for oxygen calculation. If the value is included in a multi-register write process, no changes are made. <u>Is not written to the flash memory.</u>	1 to 9999
8217	2019	float	2	r/w	Standard pressure compensation value (in mbar) for oxygen calculation. If the value is included in a multi-register write process, no changes are made. <u>This value is used after a reset.</u>	1 to 9999
8219	201B	float	2	r/w	Interval value for oxygen and temperature measurement in seconds. If the value is zero, continuous oxygen measurement is disabled; temperature measurement then takes place every 5 seconds.	0 to 9999
8221	201D	float	2	r/w	Maximum temperature at which oxygen measurement can take place. Default setting: 100 °C. <u>Sensor accuracy is only specified for a range of 0 to 50 °C.</u>	-100 to 1000
8223	201F	uint32	2	r/w	Automatic temperature compensation. Writing any value activates the sensor's internal temperature measurement. If the value is included in a multi-register write process, no changes are made. <u>Is not written to the flash memory.</u>	-
8225	2021	uint32	2	r/w	Measurement mode, 0 = humid 1 = dry	-
12289	3001	uint32	2	r/w	Measurement status and measurement control. Maintains the measurement status and triggers a single measurement using the following code values: Bit 0: Continuous measurement on/off (read only; setting the interval to "0" deletes the bit; any other value activates the bit.) Bit 1: If 1, exactly one measurement takes place (read only) Bit 2: Setting triggers a single measurement (deletion has no effect)	-

6 Modbus address tables

Table 2: Codes of units available for oxygen measurement

Code (Hex)	Unit
10	% vol.
20	% sat.
40	ppb (µg/l)
80	ppm (mg/l)
20000000	hPa
40000000	ppm gas

Table 3: Units for oxygen measurement depending on the sensor type and measurement mode

Unit	Type H10		Type H20	
	Dry	Humid	Dry	Humid
% sat.	X	X	X	X
% vol.	X	X	X	X
ppb (µg/l)		X		X
ppm (mg/l)		X		X
hPa	X	X	X	X
ppm gas			X	X

6.2.3 New sensor/changing sensor values manually



NOTE!

A timeout period of 250 ms is required after **all** write functions.

An extended timeout period of 650 ms is required after write functions for the addresses 0x0402, 0x2031 et. seq., 0x3003 and 0x3005.

Address		Data type	Number of registers	Access	Description	Value range
Dec.	Hex.					
8241	2031	float	2	r/w	Sensor constant f1	-10 to 10
8243	2033	float	2	r/w	Sensor constant dPhi1	-10 to 10
8245	2035	float	2	r/w	Sensor constant dPhi2	-10 to 10
8247	2037	float	2	r/w	Sensor constant dKSV1	-10 to 10
8249	2039	float	2	r/w	Sensor constant dKSV2	-10 to 10
8251	203B	float	2	r/w	Sensor constant m	-100 to 100
8253	203D	uint32	2	r/w	Measurement method during calibration , either "0" (humid) or 1 (dry)	0 or 1
8255	203F	float	2	r/w	Phase shift at zero point of oxygen calibration in °	0 to 90
8257	2041	float	2	r/w	Temperature at zero point of oxygen calibration in °C	0 to 99.99
8259	2043	float	2	r/w	Oxygen concentration at end value of oxygen calibration in the unit defined in register 204B.	0 to 99.99
8261	2045	float	2	r/w	Phase shift at end value of oxygen calibration in °	0 to 90
8263	2047	float	2	r/w	Temperature at end value of oxygen calibration in °C	0 to 99.99

6 Modbus address tables

Address		Data type	Number of registers	Access	Description	Value range
Dec.	Hex.					
8265	2049	float	2	r/w	Pressure at end value of oxygen calibration in mbar	1 to 9999
8267	204B	uint32	2	r/w	Unit for oxygen concentration at end value of oxygen calibration (see table 3)	-

6.2.4 Simple automatic calibrations



NOTE!

A long timeout of 650 ms is needed to describe the register 0x3003 and 0x3005.

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
12291	3003	float	2	r/w	Zero-point oxygen calibration. Trigger: Writing any value. Automatic update of the following register using the last values measured or set: 203F (phase shift) 2041 (temperature)
12293	3005	float	2	r/w	End-value oxygen calibration. Trigger: Writing a new value in register 2043 (oxygen concentration). Automatic update of the following register using the last values measured or set: 2045 (phase shift) 2047 (temperature) 2049 (pressure)
12297	3009	uint16	1	r/w	Calibration mode , is set to "0" as standard upon startup. 0 = Mode as configured in register 2021 1 = Mode "humid" 2 = Mode "dry" Is set to "0" when either register 3003 or 3005 is activated for calibration. <u>Is not written to the flash memory.</u>
12298	300A	uint16	1	r/w	Year of calibration. Is written in subsequent logbook entries. <u>Is not written to the flash memory.</u>
12299	300B	uint16	1	r/w	Month of calibration. Is written in subsequent logbook entries. <u>Is not written to the flash memory.</u>
12300	300C	uint16	1	r/w	Day of calibration. Is written in subsequent logbook entries. <u>Is not written to the flash memory.</u>
12301	300D	uint16	1	r/w	Hour of calibration. Is written in subsequent logbook entries. <u>Is not written to the flash memory.</u>
12302	300E	uint16	1	r/w	Minute of calibration. Is written in subsequent logbook entries. <u>Is not written to the flash memory.</u>
12303	300F	uint16	1	r/w	Second of calibration. Is written in subsequent logbook entries. <u>Is not written to the flash memory.</u>
12304	3010	uint16	1	r/w	Time zone of calibration. Is written in subsequent logbook entries. <u>Is not written to the flash memory.</u>

6 Modbus address tables

6.2.5 SIP and CIP settings

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
8275	2053	float	2	r/w	Lower limit of temperature range for detecting a SIP cycle in °C
8277	2055	float	2	r/w	Upper limit of temperature range for detecting a SIP cycle in °C
8279	2057	float	2	r/w	Sensor's minimum dwell time in the defined temperature range for detecting a SIP cycle in minutes
8281	2059	float	2	r/w	Lower limit of temperature range for detecting a CIP cycle in °C
8283	205B	float	2	r/w	Upper limit of temperature range for detecting a CIP cycle in °C
8285	205D	float	2	r/w	Sensor's minimum dwell time in the defined temperature range for detecting a CIP cycle in minutes
8287	205F	uint32	2	r	Number of SIP cycles since last calibration. The register is reset during each calibration, so the value may deviate from register 2063 , which contains the total number of SIP cycles.
8289	2061	uint32	2	r	Number of CIP cycles since last calibration. The register is reset during each calibration, so the value may deviate from register 2065 , which contains the total number of CIP cycles.
8291	2063	uint32	2	r	Total number of SIP cycles
8293	2065	uint32	2	r	Total number of CIP cycles
8295	2067	uint32	2	r/w	Manual acquisition of autoclaves. No automatic counting takes place. If used, the user must update the entries themselves.

6.2.6 Measurement and operating temperatures

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
8297	2069	float	2	r	Minimum operating temperature of the sensor in °C Default setting: -50
8299	206B	float	2	r	Maximum operating temperature of the sensor in °C Default setting: 130
8301	206D	float	2	r/w	Minimum temperature for oxygen measurements in °C Default setting: -40 Sensory accuracy is only specified within the calibration range (0 to 50 °C).
8303	206F	float	2	r/w	Maximum temperature for oxygen measurements in °C Default setting: 60 Sensory accuracy is only specified within the calibration range (0 to 50 °C).
8305	2071	float	2	r	Total number of hours during which the sensor has been operated outside of the defined operating temperature range (register 2069 to 206C).
8307	2073	float	2	r	Total number of hours during which the sensor has been operated outside of the defined range for oxygen measurements (register 206D to 2070).

6 Modbus address tables

6.2.7 Configuring analog 4 to 20 mA interface

Address		Data type	Number of registers	Access	Description	Value range
Dec.	Hex.					
8309	2075	float	2	r/w	Actual (measured) current value of the 4 to 20 mA interface for the low value (zero point) in mA.	0 to 30
8311	2077	float	2	r/w	Reference current value (from register 2085) of the 4 to 20 mA interface for the low value (zero point) in mA	0 to 30
8313	2079	float	2	r/w	Actual (measured) current value of the 4 to 20 mA interface for the high value (end value) in mA.	0 to 30
8315	207B	float	2	r/w	Reference current value (from register 2085) of the 4 to 20 mA interface for the high value (end value) in mA.	0 to 30
8317	207D	uint32	2	r/w	Code for the mode of the 4 to 20 mA interface: 0x00 = Off 0x01 = Fixed value (required to calibrate the interface current) 0x02 = Linear	-
8319	207F	uint32	2	r/w	Code for allocation of the 4 to 20 mA interface: 0x01 = Oxygen 0x20 = Temperature	-
8321	2081	float	2	r/w	Output value , which corresponds to 4 mA.	-999 to 9999
8323	2083	float	2	r/w	Output value , which corresponds to 20 mA.	-999 to 9999
8325	2085	float	2	r/w	Fixed value in mA, which is allocated to the output , if this has been set in the "Fixed value" mode (in register 207D).	0 to 25
8327	2087	uint32	2	r/w	Response of the 4 to 20 mA interface in the event of a fault. The following codes can be combined: 0x000001 = Error (continuous) 0x010000 = Warning (continuous) Example: To activate both the error response and the warning response, "0x010001" must be written in this register.	-
8329	2089	float	2	r/w	Current value in the event of a warning. Is only used if register 2087 is set to "0x010000".	0 to 30
8331	208B	float	2	r/w	Current value in the event of an error. Is only used if register 2087 is set to "0x000001".	0 to 30

6 Modbus address tables

6.2.8 Calibrating internal temperature probes

Address		Data type	Number of registers	Access	Description	Value range
Dec.	Hex.					
8333	208D	float	2	r/w	Low temperature reference value for calibrating the lower reference point , measured with a reference device.	0 to 60
8335	208F	float	2	r/w	High temperature reference value for calibrating the upper reference point , measured with a reference device. <u>It is important to calibrate the low reference point first!</u>	30.1 to 75

6.2.9 Configuring the measurements counter

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
8337	2091	uint32	2	r	Measurement counter , is increased after each successful measurement. Is not increased if the temperature value exceeds the maximum temperature defined in register 201D. After a restart, this register is reset to the last stored value, which is stored after every 100 measurements. <u>Example:</u> During a reset after the 1536th measurement, the value is reset to "1500".
8339	2093	uint32	2	r/w	Controls for the measurement counter 0 = Counter disabled, count is not increased 1 = Counter enabled, count is increased after every measurement 2 = Counter (register 2091) is set to "0", register 2093 is reset to the last defined value ("0" or "1")
8341	2095	uint32	2	r/w	Maximum value of the measurement counter . If this value is reached, the error "Measurement counter overflow" (bit 7 in register 2005) is triggered.

6.2.10 Error evaluation with status word

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
12295	3007	uint32	2	r	Error messages in status words 0x0001: Oxygen error (invalid measured value) 0x0002: Temperature error (invalid measured value) 0x0004: Temperature too high during measurement 0x0008: Maximum number of measurements exceeded 0x0010: No sensor cap detected 0x8000: New measured values

6 Modbus address tables

6.2.11 Data for the optical sensor cap

Address		Data type	Number of registers	Access	Description
Dec.	Hex.				
8492	212C	string[20]	10	r/w	Fabrication number (F-No.) of the sensor cap
8502	2136	string[12]	6	r/w	Part no. of the sensor cap

6 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

