

JUMO digiLine O-DO S10

Sauerstoffsensor mit digitaler Schnittstelle



Schnittstellenbeschreibung Modbus



20261400T92Z000K000

V2.00/DE/00753240/2021-08-20

1	Sicherheitshinweise	5
1.1	Warnende Zeichen	5
1.2	Hinweisende Zeichen	5
2	Modbus-Protokollbeschreibung	7
2.1	Master-Slave-Prinzip	7
2.2	Übertragungsmedien für Modbus	8
2.3	Aufbau eines RTU-Modbus-Telegramms	8
2.4	Funktionscodes	8
2.4.1	Lesen von n Worten	9
2.4.2	Schreiben eines Wortes	10
2.4.3	Schreiben von n Worten	11
2.5	Datentypen	11
2.6	Beispiele für die Übertragung von Daten	14
2.6.1	Ganzzahl-Werte	14
2.6.2	Float-Werte	14
2.6.3	Zeichenketten (Texte)	16
2.7	Checksumme (CRC16)	17
2.8	Fehlermeldungen	18
2.8.1	Modbus-Fehlercodes	18
2.8.2	Fehlermeldungen bei ungültigen Werten	19
3	Schnittstellen	21
3.1	Schnittstellenbelegung	21
3.1.1	Abschlusswiderstände	21
4	Schnittstellen konfigurieren	23
5	Modbus über serielle Schnittstelle	25
5.1	Modbus-Slave-Betrieb über serielle Schnittstelle RS 485	25
6	Modbus-Adresstabellen	27
6.1	Typenschild	27
6.2	Herstellerdaten	27
6.3	Messstelleninfo	28
6.4	Konfiguration Schnittstelle	28
6.5	Konfiguration Messung	29
6.6	Konfiguration Analogausgang	29
6.7	Betriebsdaten	30
6.7.1	Erstinbetriebnahme	30
6.7.2	Schleppzeigerfunktion	30

Inhalt

6.8	Prozesswerte32
6.8.1	Kompensation32
6.8.2	Messwerte34
6.8.3	Statuswort34
6.9	Datum und Uhrzeit für Zeitstempel35

1 Sicherheitshinweise

1.1 Warnende Zeichen



GEFAHR!

Dieses Zeichen weist darauf hin, dass ein **Personenschaden durch Stromschlag** eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



WARNUNG!

Dieses Zeichen in Verbindung mit dem Signalwort weist darauf hin, dass ein **Personenschaden** eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



VORSICHT!

Dieses Zeichen in Verbindung mit dem Signalwort weist darauf hin, dass ein **Sachschaden oder ein Datenverlust** auftritt, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



VORSICHT!

Dieses Zeichen weist darauf hin, dass durch elektrostatische Entladungen (ESD = Electro Static Discharge) **Bauteile zerstört werden** können, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Bei Rücksendungen von Geräteeinschüben, Baugruppen oder Bauelementen nur dafür vorgesehene ESD-Verpackungen verwenden.



DOKUMENTATION LESEN!

Dieses Zeichen – angebracht auf dem Gerät – weist darauf hin, dass die zugehörige **Gerätedokumentation** zu **beachten** ist. Dies ist erforderlich, um die Art der potenziellen Gefährdung zu erkennen und Maßnahmen zu deren Vermeidung zu ergreifen.

1.2 Hinweisende Zeichen



HINWEIS!

Dieses Zeichen weist auf eine **wichtige Information** über das Produkt oder dessen Handhabung oder Zusatznutzen hin.



VERWEIS!

Dieses Zeichen weist auf **weitere Informationen** in anderen Abschnitten, Kapiteln oder anderen Anleitungen hin.



WEITERE INFORMATION!

Dieses Zeichen wird in Tabellen verwendet und weist auf **weitere Informationen** im Anschluss an die Tabelle hin.



ENTSORGUNG!

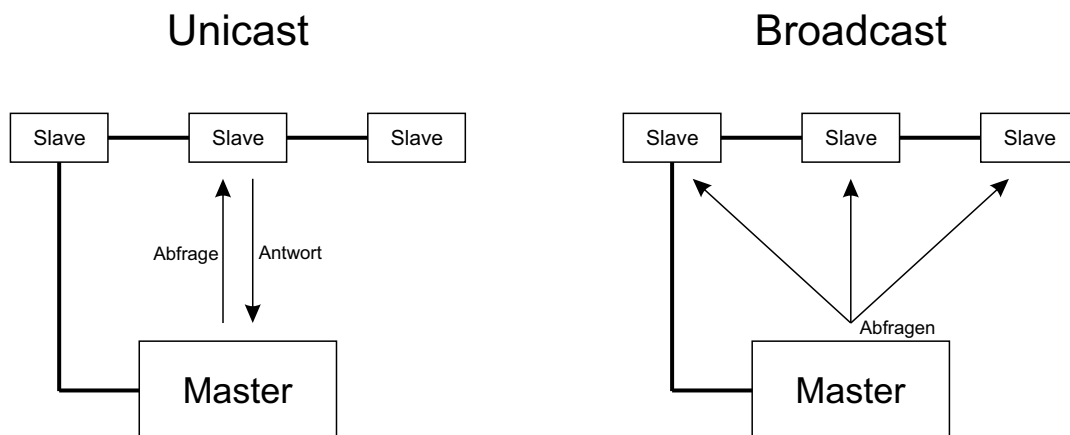
Dieses Gerät und, falls vorhanden, Batterien gehören nach Beendigung der Nutzung nicht in die Mülltonne! Bitte lassen Sie sie ordnungsgemäß und **umweltschonend entsorgen**.

1 Sicherheitshinweise

2.1 Master-Slave-Prinzip

Die Kommunikation zwischen einem Master (z. B. SCADA-System oder SPS) und einem JUMO digiLine O-DO S10 als Slave in einem Modbus findet nach dem Master-Slave-Prinzip in Form von Datenanfrage/Anweisung - Antwort statt. Slaves werden anhand ihrer Geräteadresse identifiziert. Master-Geräte benötigen keine Adresse.

⇒ Kapitel 5 „Modbus über serielle Schnittstelle“, Seite 25



Der Master steuert den Datenaustausch durch zyklische Anfragen der Slaves im gesamten Bus. Die Slaves (z. B. JUMO digiLine O-DO S10) haben lediglich Antwortfunktion. Der Master kann dabei schreibend und lesend auf die Slaves zugreifen. Auf diese Weise können Daten in Echtzeit zwischen Master und Slave-Geräten kommuniziert werden. Slaves können nicht direkt miteinander kommunizieren. Um Daten von Slave zu Slave zu übertragen, muss der Master die Daten aus dem einen Slave auslesen und dann an den nächsten übertragen.

In der Regel richtet der Master seine Anfragen gezielt an einzelne Slaves. Dazu muss er die jeweiligen Slaves mit ihrer Unicast-Adresse ansprechen. Anfragen können aber auch als Rundsendungsnachricht an alle Slaves im Bus gerichtet werden. Hierfür wird als Slave-Adresse die Broadcast-Adresse verwendet. Broadcast-Anfragen werden von den Slaves nicht beantwortet. In seriellen Bussystemen würden sonst Datenkollisionen entstehen. Daher macht die Verwendung von Broadcast-Adressen nur mit Funktionscodes zum Schreiben von Daten Sinn. Broadcasts können nicht mit Funktionscodes zum Lesen von Daten verwendet werden.



HINWEIS!

Der JUMO digiLine O-DO S10 kann nur als Slave betrieben werden.

2 Modbus-Protokollbeschreibung

2.2 Übertragungsmedien für Modbus

Serielle Schnittstelle

Die Modbus-Spezifikation sieht für die Datenkommunikation über **serielle Schnittstelle** die Übertragungsmodi **RTU-Modus (Remote Terminal Unit)** und ASCII-Modus (Übertragung der Daten im ASCII-Format) vor. Der JUMO digiLine O-DO S10 unterstützt nur den **RTU-Modus**. Hierbei werden die Daten im Binärformat über den seriellen Bus (RS485) übertragen.

⇒ Kapitel 5 „Modbus über serielle Schnittstelle“, Seite 25

2.3 Aufbau eines RTU-Modbus-Telegramms

Modbus-Telegramme sind nach folgendem Muster aufgebaut:

Slave-Adresse	Funktionscode	Datenfeld	Checksumme CRC
1 Byte	1 Byte	x Bytes	2 Bytes

Jedes Telegramm enthält vier Felder:

Slave-Adresse	Geräteadresse eines bestimmten Slaves
Funktionscode	Funktionsauswahl (Lesen/Schreiben von Worten)
Datenfeld	Enthält die Informationen (je nach Funktionscode) - Wortadresse/Bitadresse - Wortanzahl/Bitanzahl - Wortwert(e)/Bitwert(e)
Checksumme	Erkennung von Übertragungsfehlern

2.4 Funktionscodes

Funktionsübersicht

Die nachfolgend beschriebenen Funktionen des Modbus-Standards stehen zum Auslesen von Messwerten, Geräte- und Prozessdaten sowie zum Schreiben von Daten zur Verfügung.

Funktionscode		Funktion	Begrenzung
Hex	Dez.		
03 oder 04	3 oder 4	Lesen von n Worten	Max. 125 Worte (250 Bytes)
06	6	Schreiben eines Wortes	Max. 1 Wort (2 Bytes)
10	16	Schreiben von n Worten	Max. 123 Worte (246 Bytes)

2 Modbus-Protokollbeschreibung

2.4.1 Lesen von n Worten

Mit dieser Funktion werden n Worte ab einer bestimmten Adresse gelesen.

Datenanfrage

Slave-Adresse	Funktion 0x03 oder 0x04	Adresse erstes Wort	Wortanzahl x	Checksumme CRC
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

Antwort

Slave-Adresse	Funktion 0x03 oder 0x04	Anzahl gelesener Bytes	Wert(e)	Checksumme CRC
1 Byte	1 Byte	1 Byte	2 x Bytes	2 Bytes

Beispiel

Lesen des Gerätenamens. Es handelt sich hier im Beispiel um den Gerätenamen „dl CR“. Da der Gerätenamenname bis zu 10 Zeichen lang ist (inklusive NULL-Zeichen als Ende-Kennung) und 2 ASCII-Zeichen des Gerätenamens pro Wort abgelegt werden, ist es hier erforderlich, dass 5 Worte, das sind 10 Bytes, eingelesen werden. Diese und weitere Modbus-Adressen sind dem Kapitel 6 „Modbus-Adresstabellen“, Seite 27 zu entnehmen.

Hex-Code der Datenanfrage:

01	03	00 00	00 05	85 C9
Slave	Funktion	Adresse 1. Wort	Wortanzahl	CRC

Hex-Code der Antwort (Werte im Byte-Format):

01	03	0A	64 6C 20 43 52 00 00 00 00 00	E6 B1
Slave	Funktion	Bytes gelesen	Text in ASCII	CRC

2 Modbus-Protokollbeschreibung

2.4.2 Schreiben eines Wortes

Bei der Funktion Wortschreiben sind die Datenblöcke für Anweisung und Antwort identisch.



VORSICHT!

Schreiboperationen auf manche R/W-Parameter bewirken ein Abspeichern im EEPROM oder Flash-Speicher. Diese Speicherbausteine haben nur eine begrenzte Anzahl von Schreibzyklen (ca. 100.000 bzw. 10.000).

Häufiges Beschreiben entsprechender Variablen kann daher dazu führen, dass ein Speicherfehler auftritt.

► Die Anzahl der Schreibvorgänge sollte daher möglichst klein gehalten werden.

Anweisung

Slave-Adresse	Funktion 0x06	Wort-Adresse	Wort-Wert	Checksumme CRC
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

Antwort

Slave-Adresse	Funktion 0x06	Wort-Adresse	Wort-Wert	Checksumme CRC
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

Beispiel

In diesem Beispiel soll ein Kommando die Jahreszahl „2021“ (hexadezimaler Wert: 0x07E5) in die Daten für Datum und Uhrzeit des Gerätes schreiben. Die Slave-Adresse des Gerätes soll hierbei 1 sein und die betreffende Wortadresse der Jahreszahl ist 0x2840 (Kapitel 6.9 „Datum und Uhrzeit für Zeitstempel“, Seite 35).

Hex-Code der Anweisung:

01	06	28 40	07 E5	42 05
Slave	Funktion	Wortadresse	Wert	CRC

Hex-Code der Antwort:

01	06	28 40	07 E5	42 05
Slave	Funktion	Wortadresse	Wert	CRC

2 Modbus-Protokollbeschreibung

2.4.3 Schreiben von n Worten

Anweisung

Slave-Adresse	Funktion 0x10	Adresse erstes Wort	Wortanzahl x	Byte-Anzahl 2 x	x Wortwert(e)	Checksumme CRC
1 Byte	1 Byte	2 Bytes	2 Bytes	1 Byte	2 x Bytes	2 Bytes

Antwort

Slave-Adresse	Funktion 0x10	Adresse erstes Wort	Wortanzahl x	Checksumme CRC
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

Beispiel

Schreiben des Wortes „Anlage Nord“ (UTF-8-Kodierung mit Ende-Kennung: 0x41 0x6E 0x6C 0x61 0x67 0x65 0x20 0x4E 0x6F 0x72 0x64 0x00) ab Wortadresse 0x021E als Beschreibung für die Messstelle.

⇒ Kapitel 6.3 „Messstelleninfo“, Seite 28

Hex-Code der Anweisung:

01	10	02 1E	00 06	0C	41 6E 6C 61 67 65 20 4E 6F 72 64 00	F8 70
Slave	Funktion	Adresse 1. Wort	Wortanzahl	Byte-Anzahl	Text in UTF-8-Kodierung	CRC

Hex-Code der Antwort:

01	10	02 1E	00 06	21 B5
Slave	Funktion	Adresse 1. Wort	Wortanzahl	CRC

2.5 Datentypen

Datentyp	Beschreibung	Zugriff	Mögliche Funktionscodes	Anzahl Modbus-Register
bool	niederwertigstes Bit eines Wortes als Bit-Wert 0000 0000 0000 0001 = 1 bzw. TRUE (wahr) 0000 0000 0000 0000 = 0 bzw. FALSE (unwahr)	read only	03, 04	1
		read/write	03, 04, 06, 16	
word	Wort (16 Bit) als Bitfeld Die individuellen Bedeutungen der einzelnen Bits ist den Modbus-Adresstabellen im letzten Kapitel dieses Dokuments zu entnehmen.	read only	03, 04	1
		read/write	03, 04, 06, 16	

2 Modbus-Protokollbeschreibung

Datentyp	Beschreibung	Zugriff	Mögliche Funktionscodes	Anzahl Modbus-Register
char	<p>Low-Byte eines Wortes als ganzzahliger Wert; das High-Byte wird nicht verwendet.</p> <p>Wertebereiche bei numerischer Interpretation: 0 bis 255 für vorzeichenlose Daten -128 bis 127 für vorzeichenbehaftete Daten</p> <p>Bei Interpretation als Bool: niederwertigstes Bit eines Wortes als Bit-Wert</p> <p>0000 0000 0000 0001 = 1 bzw. TRUE (wahr) 0000 0000 0000 0000 = 0 bzw. FALSE (unwahr)</p>	read only	03, 04	1
		read/write	03, 04, 06, 16	
short	<p>Wort (16 Bit) als ganzzahliger Wert</p> <p>Wertebereiche: 0 bis 65535 für vorzeichenlose Daten -32768 bis 32767 für vorzeichenbehaftete Daten</p>	read only	03, 04	1
		read/write	03, 04, 06, 16	
uint32	<p>Doppelwort (32 Bit) als vorzeichenloser ganzzahliger Wert</p> <p>Wertebereich: 0 bis 4.294.967.295</p>	read only	03, 04	2
		read/write	03, 04, 16	
enum	<p>Wort (16 Bit) als ganzzahliger Wert</p> <p>Durchnummerierungen von Einstellungen (mögliche Nummerierungsweite: 0 bis 65535). Die Bedeutungen der einzelnen Durchnummerierungen für Enumerations-Variablen ist den Modbus-Adresstabellen im letzten Kapitel dieses Dokuments zu entnehmen.</p>	read only	03, 04	1
		read/write	03, 04, 06, 16	

2 Modbus-Protokollbeschreibung

Datentyp	Beschreibung	Zugriff	Mögliche Funktionscodes	Anzahl Modbus-Register																												
float	<p>2 Wörter als 32-Bit-Fließkommazahl mit Kodierung nach IEEE 754, wobei die korrekte Reihenfolge der Übertragung der 4 Bytes bei der Übertragung beachtet werden muss. In der Konfiguration der RS485-Schnittstelle kann die gewünschte Kodierung in der Einstellung „Gleitkommaformat“ ausgewählt werden.</p> <p>S = Vorzeichenbit E = Exponent (2er-Komplement) M = 23 Bit normalisierte Mantisse</p> <p style="text-align: center;">IEEE 754 Big Endian</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Byte 1</th> <th>Byte 2</th> <th>Byte 3</th> <th>Byte 4</th> </tr> </thead> <tbody> <tr> <td>SE7EEEEEE1</td> <td>E0M23MMMMMM16</td> <td>M15MMMMMMM8</td> <td>M7MMMMMMM0</td> </tr> </tbody> </table> <p style="text-align: center;">IEEE 754 Little Endian</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Byte 1</th> <th>Byte 2</th> <th>Byte 3</th> <th>Byte 4</th> </tr> </thead> <tbody> <tr> <td>M7MMMMMMM0</td> <td>M15MMMMMMM8</td> <td>E0M23MMMMMM16</td> <td>SE7EEEEEE1</td> </tr> </tbody> </table> <p style="text-align: center;">Standard-Modbus-Kodierung</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Byte 1</th> <th>Byte 2</th> <th>Byte 3</th> <th>Byte 4</th> </tr> </thead> <tbody> <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> </tbody> </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																															
string[n]	<p>Zeichenkette für eine Länge von bis zu n Bytes. Die Zeichenkodierung der einzelnen Variablen ist den Modbus-Adresstabelle im letzten Kapitel dieses Dokuments zu entnehmen. Als Zeichensatz-Kodierungen treten UTF-8, ISO 8859-1 oder ASCII-Kodierung auf.</p> <p>Jedes der n Modbus-Register-Worte enthält 2 aufeinanderfolgende Bytes der Zeichenkette.</p> <p>Es ist zu beachten, dass die Zeichenkette als letztes Zeichen immer ein „\0“ (hexadezimal: 0x00) als Ende-Kennung enthalten muss. Außerdem muss die Zeichenkettenlänge insgesamt eine gerade Byte-Anzahl aufweisen. Um das zu gewährleisten muss bei ungerader Byte-Anzahl inklusive „\0“ ein zusätzliches „\0“ angehängt werden.</p>	read only	03, 04	n/2																												
		read/write	03, 04, 06, 16																													

2 Modbus-Protokollbeschreibung

2.6 Beispiele für die Übertragung von Daten

Zum Auslesen von Ganzzahl-, Float- und Text-Werten wird die Funktion 0x03 oder 0x04 (Einlesen von n Worten) verwendet.

Datenanfrage

Slave-Adresse	Funktion 0x03 oder 0x04	Adresse erstes Wort	Wortanzahl	Checksumme CRC
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

Ganzzahl-Werte werden über Modbus im folgenden Format übertragen:
Zuerst das High-, dann das Low-Byte.

Antwort

Slave-Adresse	Funktion 0x03 oder 0x04	Anzahl gelesener Bytes	Wert(e)	Checksumme CRC
1 Byte	1 Byte	1 Byte	x Bytes	2 Bytes

2.6.1 Ganzzahl-Werte

Beispiel

In diesem Beispiel soll Jahr der Erstinbetriebnahme an Adresse 0x0500 (Kapitel 6.7.1 „Erstinbetriebnahme“, Seite 30) ausgelesen werden. Der Wert soll hier „2020“ (Wort-Wert 0x07E4) sein.

Datenanfrage:

01	03	05 00	00 01	84 C6
Slave	Funktion	Adresse 1. Wort	Wortanzahl	CRC

Antwort (Werte im Modbus-Float-Format):

01	03	02	07 E4	BA 3F
Slave	Funktion	Bytes gelesen	Integerwert	CRC

2.6.2 Float-Werte

Dieses Gerät arbeitet bei Float-Werten mit dem IEEE-754-Standard-Format (32 Bit), allerdings mit dem Unterschied, dass Byte 1 und 2 mit Byte 3 und 4 vertauscht sind.

Single-Float-Format (32 Bit) nach Standard IEEE 754

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

S - Vorzeichen-Bit

E - Exponent (2er-Komplement)

M - 23 Bit normalisierte Mantisse

Modbus-Float-Format

Modbus-Adresse x		Modbus-Adresse x+1	
MMMMMMMM	MMMMMMMM	SEEEEEEE	EMMMMMMM

2 Modbus-Protokollbeschreibung

Modbus-Float-Format

Modbus-Adresse x		Modbus-Adresse x+1	
Byte 3	Byte 4	Byte 1	Byte 2

Beispiel

In diesem Beispiel soll der Wert „Temperaturmesswert“ an Adresse 0x2620 des Gerätes ausgelesen werden. Der Wert soll hier 20,25 (0x41A20000 im IEEE-754-Format) sein.

Datenanfrage:

01	03	26 20	00 02	CE 89
Slave	Funktion	Adresse 1. Wort	Wortanzahl	CRC

Antwort (Werte im Modbus-Float-Format):

01	03	04	00 00	41 A2	4B DA
Slave	Funktion	Bytes gelesen	Float-Wert		CRC

Nach der Übertragung vom Gerät müssen die Bytes des Float-Wertes entsprechend vertauscht werden. Viele Compiler (z.B. Microsoft Visual C++) legen die Float-Werte in folgender Reihenfolge ab:

Float-Wert

Adresse x	Adresse x+1	Adresse x+2	Adresse x+3
MMMMMMMM	MMMMMMMM	EMMMMMMM	SEEEEEEE
Byte 4	Byte 3	Byte 2	Byte 1



HINWEIS!

Die Reihenfolge der Bytes hängt davon ab, wie Float-Werte in der betreffenden Anwendung gespeichert werden. In den Einstellungen der seriellen Schnittstelle können gängige genormte Gleitkommaformate gewählt werden. Unter Umständen kann es aber erforderlich sein, die Byte-Reihenfolge durch entsprechende Vertauschungen im Schnittstellenprogramm des Masters anzupassen.

2 Modbus-Protokollbeschreibung

2.6.3 Zeichenketten (Texte)

Zeichenketten werden im UTF8-kodiert übertragen.



HINWEIS!

Als letztes Zeichen muss immer ein „\0“ (hexadezimal: 0x00) als Ende-Kennung übertragen werden. Danach folgende Zeichen haben keine Bedeutung.

Da die Übertragung von Texten wortweise (16-Bit-Register) erfolgt, wird bei einer ungeraden Byte-Anzahl (inkl. „\0“) zusätzlich 0x00 angehängt.

Die in den Adresstabellen angegebenen Maximallängen für Zeichenketten beinhalten das abschließende „\0“. Das heißt, bei „String[60]“ darf der Text inklusive „\0“ maximal 60 Byte lang sein.

Bei 19 Unicode-Zeichen mit einer Länge von je 3 Bytes bleiben nur 2 Bytes für das zwanzigste Zeichen. 1 Byte wird für die Ende-Kennung benötigt.

Beispiel

Abfrage des Textes von Adresse 0x021E, unter dieser Adresse steht die Zeichenkette für den Beschreibung der Messstelle „Anlage Nord“ (ASCII-Code: 0x41 0x6E 0x6C 0x61 0x67 0x65 0x20 0x4E 0x6F 0x72 0x64).

Hex-Code der Anfrage:

01	03	02 1E	00 07	65 B6
Slave	Funktion	Adresse 1. Wort	Wortanzahl	CRC

Hex-Code der Antwort:

01	03	00 0E	41 6E 6C 61 67 65 20 4E 6F 72 64 00 00 AA	C5 DF
Slave	Funktion	Bytes ge- lesen	Wortwerte (ASCII-Zeichen)	CRC



HINWEIS!

Die Werte (hier: 00 AA) vor der CRC-Summe (hier: C5DF) werden nicht berücksichtigt, da sie hinter der Endekennung „\0“ liegen.

2 Modbus-Protokollbeschreibung

2.7 Checksumme (CRC16)

Berechnungsschema

Anhand der Checksumme (CRC16) werden Übertragungsfehler erkannt. Wird bei der Auswertung ein Fehler festgestellt, antwortet das entsprechende Gerät nicht.

CRC = 0xFFFF	
CRC = CRC XOR ByteOfMessage	
For (1 bis 8)	
CRC = SHR(CRC)	
if (rechts hinausgeschobenes Flag = 1)	
then	else
CRC = CRC XOR 0xA001	
while (nicht alle ByteOfMessage bearbeitet);	



HINWEIS!

Das Low-Byte der Checksumme wird zuerst übertragen!

Beispiel: Die CRC16-Checksumme CC DD wird in der Reihenfolge DD CC übertragen und dargestellt.

2 Modbus-Protokollbeschreibung

2.8 Fehlermeldungen

2.8.1 Modbus-Fehlercodes

Voraussetzungen für die Modbus-Kommunikation

Folgende Bedingungen müssen erfüllt sein, damit ein Slave Anfragen empfangen, bearbeiten und beantworten kann:

- Baudrate und Datenformat von Master und Slave müssen übereinstimmen.
- In der Anfrage muss die korrekte Slave-Adresse verwendet werden.
- Slave-Geräte antworten nur bei erfolgreichem Prüfsummen-Check der Anfrage durch den Slave. Anderenfalls wird die Anfrage vom Slave verworfen.
- Die Anweisung des Masters muss vollständig und konform zum Modbus-Protokoll sein.
- Die Anzahl der zu lesenden Worte muss größer 0 sein.

Fehlercodes

Wurde die Datenanfrage des Masters vom Slave ohne Übertragungsfehler empfangen, konnte aber nicht bearbeitet werden, antwortet der Slave mit einem Fehlercode. Folgende Fehlercodes können auftreten:

- 01 = ungültige Funktion; Die Funktionscodes, die von diesem Gerät unterstützt werden, sind im Kapitel 2.4 „Funktionscodes“, Seite 8 aufgeführt.
- 02 = ungültige Adresse oder eine zu große Anzahl von Worten bzw. Bits soll gelesen oder geschrieben werden
- 03 = Wert ist außerhalb des zulässigen Bereichs
- 08 = Wert ist schreibgeschützt

Antwort im Fehlerfall

Slave-Adresse	Funktion XX OR 80h	Fehlercode	Checksumme CRC
1 Byte	1 Byte	1 Byte	2 Bytes

Der Funktionscode wird mit 0x80 verODERT. Dadurch wird das höchstwertige Bit (msb) auf 1 gesetzt.

Beispiel

Datenanfrage:

01	06	23 45	00 01	52 5B
Slave	Wort schreiben	Wortadresse	Wort-Wert	CRC

Antwort (mit Fehlercode 2):

01	86	02	C3 A1
Slave	Funktion OR	Fehler	CRC

Antwort mit Fehlercode 02, weil die Adresse 0x2345 nicht vorhanden ist.

2 Modbus-Protokollbeschreibung

2.8.2 Fehlermeldungen bei ungültigen Werten

Bei Messwerten im Float-Format wird die Fehlernummer im Wert selbst dargestellt, d.h. anstatt des Messwerts ist die Fehlernummer enthalten.

Fehlercode bei Float-Werten	Fehler
$1,0 \times 10^{37}$	Messbereichsunterschreitung
$2,0 \times 10^{37}$	Messbereichsüberschreitung
$3,0 \times 10^{37}$	kein gültiger Eingangswert
$4,0 \times 10^{37}$	Division durch Null
$5,0 \times 10^{37}$	Mathematikfehler
$6,0 \times 10^{37}$	Ungültige Kompensationstemperatur
$7,0 \times 10^{37}$	Fühlerkurzschluss
$8,0 \times 10^{37}$	Fühlerbruch
$9,0 \times 10^{37}$	Timeout

Beispiel

Einlesen des Sauerstoffmesswertes an Modbus-Adresse 0x2600:

Datenanfrage:

08	03	26 00	00 02	CF DA
Slave	Funktion	Wortadresse	Wortanzahl	CRC

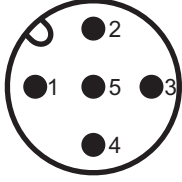
Antwort:

08	03	04	BD C2	7D 70	C6 17
Slave	Funktion	Gelesene Bytes	Fehlercode		CRC

Der vom Analogeingang gelieferte Messwert 0x7D70BDC2 ($=2,0 \times 10^{37}$) zeigt an, dass es sich um einen Messbereichsüberschreitung handelt.

2 Modbus-Protokollbeschreibung

3.1 Schnittstellenbelegung

Funktion	Pin	Abbildung (Buchse)
nicht angeschlossen	1	
+24 V Spannungsversorgung vom Anzeigegerät/Regler	2	
GND	3	
RS485 B (RxD/TxD-)	4	
RS485 A (RxD/TxD+)	5	
Der Anschluss an der seriellen Schnittstelle eines Mastergerätes oder eines Messumformers mit Schraub- oder Federzugklemmen erfolgt mit Hilfe eines JUMO M12-digiLine-Master-Anschlusskabels (siehe Zubehör in Typenblatt / Betriebsanleitung).		



HINWEIS!

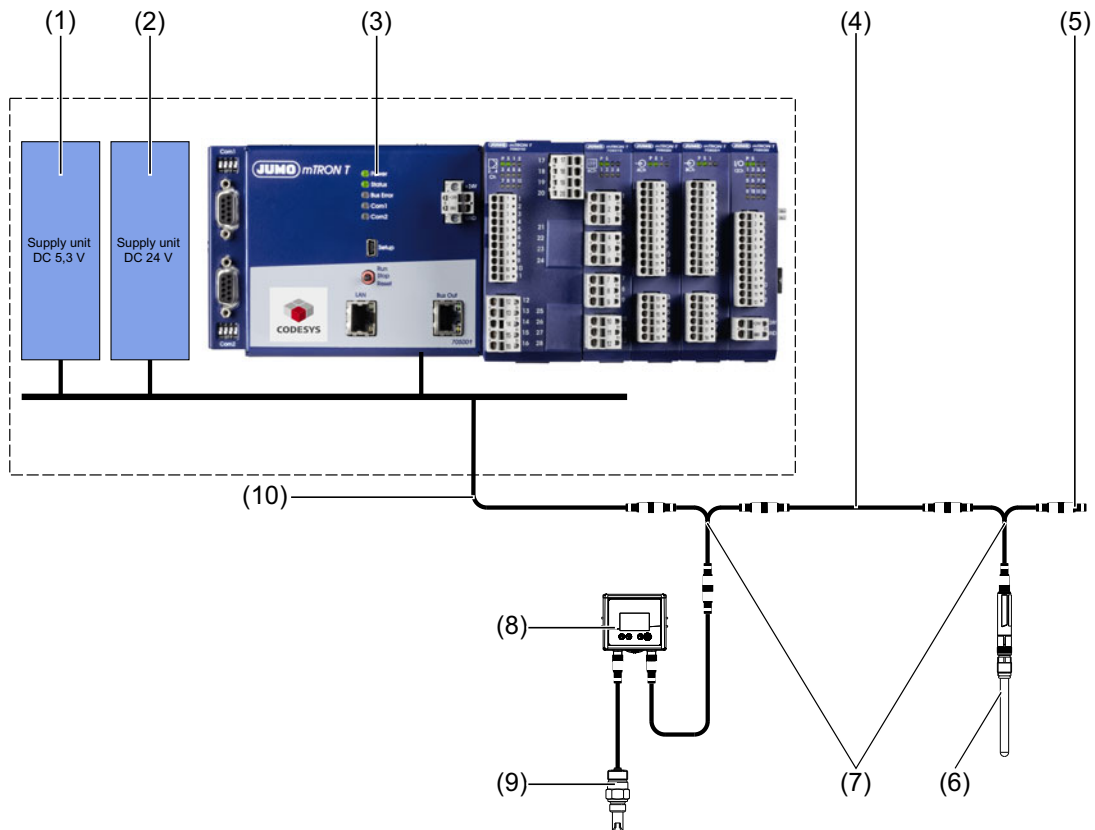
Zum Anschluss von Sensoren mit JUMO-digiLine-Elektronik sind konfektionierte Anschlussleitungen erhältlich. Verwenden Sie diese, um eine zuverlässige Leitungsverbindung und Datenübertragung im Bus-System zu gewährleisten. In den Bestellangaben der Betriebsanleitung finden Sie eine Auflistung, der als Zubehör erhältlichen Kabel und Stecker.

3.1.1 Abschlusswiderstände

Eine JUMO digiLine-Bus-Leitung wird an vom Mastergerät entfernten Enden mit Terminierungssteckern abgeschlossen (Teile-Nr.: 00461591). Diese werden einfach an der letzten freien M12-Kupplung des letzten Y-Verteilers (Teile-Nr.: 00638327) im Bus angeschlossen. Bei Mastergeräteseitigen Busleitungsenden muss der Busabschluss der RS485-Busleitung gemäß der Betriebsanleitung des Modbus-Mastergerätes erfolgen.

3 Schnittstellen

Anschlussbeispiel von 2 Sensoren mit JUMO digiLine-Elektronik an einem JUMO mTRON T



- (1) stabilisiertes Netzteil mit DC 5,3 V-Ausgang zur Speisung von Sensoren mit JUMO digiLine pH/ORP/T 5-polig (Strombegrenzung mit Sicherung 3 A erforderlich)
- (2) stabilisiertes Netzteil mit DC 24 V-Ausgang zur Speisung von Sensoren mit JUMO-digiLine-Elektronik und des JUMO mTRON T (Strombegrenzung mit Sicherung 3 A erforderlich)
- (3) JUMO mTRON T - Zentraleinheit mit freigeschalteter SPS-Funktion und RS422/485 Modbus RTU (siehe Bestellangabe JUMO mTRON T)
- (4) JUMO M12-Verbindungskabel 5-polig und A-kodiert
- (5) JUMO M12-Terminierungsstecker 5-polig zum Busabschluss
- (6) JUMO Sensoren mit JUMO digiLine pH/ORP/T 5-polig
- (7) JUMO Y-Verteiler 5-polig mit 2× M12-Kabelbuchsen und 1× M12-Stecker jeweils A-kodiert
- (8) JUMO digiLine CR in Geräteausführung mit RS485-Schnittstelle
- (9) konduktiver Leitfähigkeitssensor
- (10) JUMO digiLine-Master-Anschlusskabel mit einseitig offenen Aderenden zum Anschluss an Geräte mit Schraub- oder Federzugklemmen (siehe Zubehör)

Hinweise



VORSICHT!

Durch fehlerhafte Installation oder falsche Einstellungen von Betriebsmitteln können unerwartete Betriebszustände einer Anlage auftreten.

Dies kann Prozesse in ihrer ordnungsgemäßen Funktion beeinträchtigen oder zu Schäden führen.

- ▶ Daher immer vom Gerät unabhängige Sicherheitseinrichtungen vorsehen und die Einstellungen nur von Fachpersonal durchführen lassen.
-



HINWEIS!

Änderungen der in diesem Kapitel beschriebenen Konfigurationseinstellungen der JUMO digiLine-Elektronik können am PC mit der JUMO DSM-Software vorgenommen werden.

4 Schnittstellen konfigurieren

Einstellungen für serielle Schnittstellen

Damit alle Teilnehmergeräte in einem Bus miteinander kommunizieren können, müssen ihre Schnittstellen-Einstellungen korrekt vorgenommen werden. Die folgende Tabelle stellt die Einstellmöglichkeiten der seriellen Schnittstellen der JUMO digiLine-Elektronik dar.



HINWEIS!

Das JUMO digiLine-Protokoll vergibt die Schnittstellenparameter bei der Inbetriebnahme automatisch (Plug & Play). Beim Betrieb an einem Modbus-Master-Gerät müssen die Schnittstellenparameter vor der ersten Inbetriebnahme mit der JUMO DSM-Software eingestellt werden.

Konfigurationspunkt	Auswahl/Einstellungen	Beschreibung
Baudrate	9600 19200 38400	Übertragungsgeschwindigkeit (Symbolrate) der seriellen Schnittstelle (muss bei allen Busteilnehmern übereinstimmen)
Datenformat	8 - 1 - no Parity 8 - 1 - odd Parity 8 - 1 - even Parity 8 - 2 - no Parity	Format des Datenwortes (muss bei allen Busteilnehmern übereinstimmen) Nutzbit - Stoppbit - Parität
Gleitkommaformat	Standard IEEE754_LITTLE IEEE754_BIG	wählbares Übertragungsformat für Float-Werte (Gleitkommazahlen)
minimale Antwortzeit	0 bis 500 ms	Minstdauer vom Empfang einer Anfrage bis zum Senden einer Antwort Dieser Parameter dient dazu, die Antwort-Geschwindigkeit des Gerätes an langsamere Busteilnehmer anzupassen.
Geräteadresse	1 bis 247	eindeutige Kennung eines Busteilnehmers 0 = Broadcast-Adresse ^a 1 bis 247 = Unicast-Adressen ^b

^a Im Modbus-Standard ist die Geräteadressierung festgelegt. Die Broadcast-Adresse darf nicht als Slave-Adresse verwendet werden. Sie ist für Rundsendungsnachrichten vorgesehen.

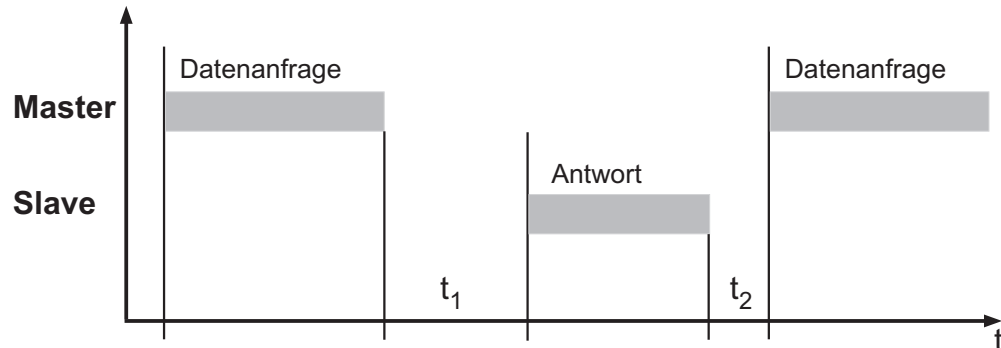
^b Unicast-Adressen sind für die Verwendung als Slave-Adressen vorgesehen. Sie dienen der eindeutigen Kennung der Slave-Geräte, damit diese vom Master explizit angesprochen werden können.

5 Modbus über serielle Schnittstelle

5.1 Modbus-Slave-Betrieb über serielle Schnittstelle RS 485

Zeitlicher Ablauf der Kommunikation

Eine Abfragezyklus im Bus läuft nach folgendem Zeitschema ab:



t_1	Interne Wartezeit des Geräts vor der Überprüfung der Datenanfrage und der internen Bearbeitungszeit: min.: 3,5 Byte-Zeiten + minimale Antwortzeit typisch: 5 ms max.: 25 ms + minimale Antwortzeit
t_2	Wartezeit, die der Master einhalten muss, bevor er eine neue Datenanfrage startet: 3,5 Zeichen bzw. mindestens 2 ms



HINWEIS!

In der Konfiguration kann die minimale Antwortzeit eingestellt werden. Diese eingestellte Zeit wird mindestens eingehalten, bevor eine Antwort gesendet wird (0 bis 500 ms). Ist die Bearbeitung einer Master-Anfrage im Slave vor Ablauf der minimalen Antwortzeit abgeschlossen, wird die Antwort erst nach Ablauf der „minimalen Antwortzeit“ übertragen.



HINWEIS!

Innerhalb von t_1 und t_2 und während der Antwortzeit des Slaves dürfen vom Master keine Datenanfragen gestellt werden. Anfragen während t_1 und t_2 werden vom Slave ignoriert. Anfragen während der Antwortzeit führen dazu, dass alle gerade auf dem Bus befindlichen Daten ungültig werden.



HINWEIS!

Das Ende-Kennzeichen nach einer Datenanfrage oder Datenantwort ist 3,5 Zeichen lang. Die Dauer dieser 3,5 Zeichen ist abhängig von der Baudrate.

Zeichenübertragungszeit

Anfang und Ende eines Datenblocks sind durch Übertragungspausen gekennzeichnet. Die Zeichenübertragungszeit (Zeit für die Übertragung eines Zeichens) ist abhängig von der Baudrate. Bei einem Datenformat von 8 Datenbits keinem Paritätsbit und einem Stoppbit ergibt sich:

$$\text{Zeichenübertragungszeit [ms]} = 1000 \times 10 \text{ Bit} \div \text{Baudrate}$$

Bei den anderen Datenformaten ergibt sich:

$$\text{Zeichenübertragungszeit [ms]} = 1000 \times 11 \text{ Bit} \div \text{Baudrate}$$

5 Modbus über serielle Schnittstelle

Beispiel

Kennzeichen für Datenanfrage- oder Antwort-Ende

Wartezeit = 3,5 Zeichen * 1000 * 11 Bit ÷ Baudrate

Baudrate[Baud]	Datenformat[Bit]	Zeichenübertragungszeit[ms]
38400	11	0,287
	10	0,260
19200	11	0,573
	10	0,521
9600	11	1,146
	10	1,042

6 Modbus-Adresstabellen

6.1 Typenschild

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
0	0000	string[10]	5	r	Gerätename	ASCII-Kodierung mit „NULL“ als Ende-Kennung
5	0005	string[12]	6	r	Software-Version	ASCII-Kodierung mit „NULL“ als Ende-Kennung
11	000B	short	1	r	Software-Nummer	
12	000C	char	1	r	Sensor-Major-Version	
13	000D	char	1	r	Sensor-Minor-Version	
14	000E	short	1	r	Testversion	
15	000F	char	1	r	Hardware-Major-Version	
16	0010	char	1	r	Hardware-Minor-Version	
17	0011	string[14]	7	r	VDN-Version	ASCII-Kodierung mit „NULL“ als Ende-Kennung
24	0018	short	1	r	VDN-Nummer	
25	0019	short	1	r	VDN-Versionsnummer	

6.2 Herstellerdaten

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
256	0100	char[20]	10	r	Seriennummer	ASCII-Kodierung mit „NULL“ als Ende-Kennung
266	010A	enum	1	r	Sensortyp	Enumeration: 0 = - 1 = JUMO digiLine pH/ORP/T 2 = JUMO digiLine CR/Ci 3 = - 4 = JUMO digiLine O-DO S10 5 = -
267	010B	enum	1	r	Sauerstoffsensoren-Subtyp	Enumeration: 0 = JUMO digiLine O-DO S10
273	0111	char[12]	6	r	Teilenummer	ASCII-Kodierung mit „NULL“ als Ende-Kennung
279	0117	char[64]	32	r	Bestellschlüssel	ASCII-Kodierung mit „NULL“ als Ende-Kennung
335	014F	Uint32	2	r	Hardwareadresse	
347	015B	short	1	r	Fertigungsdatum Jahr	
348	015C	char	1	r	Fertigungsdatum Monat	
349	015D	char	1	r	Fertigungsdatum Tag	
350	015E	char	1	r	Fertigungsdatum Stunde	
351	015F	char	1	r	Fertigungsdatum Minute	
352	0160	char	1	r	Fertigungsdatum Sekunde	
353	0161	short	1	r	Fertigungsdatum Zeitzone	Offset zur UTC in Minuten

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
354	0162	Uint32	2	r	Kalibrierstatus	

6.3 Messstelleninfo

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
512	0200	string[60]	30	r/w	Tag-Nummer	UTF-8-Kodierung + NULL“ als Ende-Kennung Die Byte-Anzahl muss gerade sein (ggf. mit einem weiteren NULL-Zeichen auffüllen)
542	021E	string[192]	96	r/w	Beschreibung	UTF-8-Kodierung + NULL“ als Ende-Kennung Die Byte-Anzahl muss gerade sein (ggf. mit einem weiteren NULL-Zeichen auffüllen)
638	027E	string[110]	55	r/w	Sensorherkunft	Diese Variable wird automatisch vom JUMO-digiLine-Mastergerät verwaltet und darf nicht anderweitig beschrieben werden! UTF-8-Kodierung + NULL“ als Ende-Kennung Die Byte-Anzahl muss gerade sein (ggf. mit einem weiteren NULL-Zeichen auffüllen)

6.4 Konfiguration Schnittstelle

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
768	0300	char	1	r/w	Geräteadresse	
769	0301	enum	1	r/w	Baudrate	Enumeration: 0 = 9600 1 = 19200 2 = 38400
770	0302	enum	1	r/w	Datenformat	Enumeration: 0 = 8 – 1 – no parity 1 = 8 – 1 – odd parity 2 = 8 – 1 – even parity 3 = 8 – 2 – no parity
773	0305	short	1	r/w	minimale Antwortzeit	

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
775	0307	enum	1	r/w	Floatwert-Format	Enumeration: 0 = Standard-Modbus-Format 1 = IEEE 754 Little Endian 2 = IEEE 754 Big Endian

6.5 Konfiguration Messung

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung / Hinweis
Dez.	Hex.					
9520	2530	enum	1	r/w	Einheit Sauerstoffmessung	0 = % Vol 1 = % SAT 2 = - 3 = ppm 4 = mbar (Sauerstoffpartialdruck)
9600	2580	float	2	r/w	Kompensationssalzgehalt (elektrolytische Leitfähigkeit in mS/cm)	nur wirksam bei Einheit „ppm“
9602	2582	float	2	r/w	Kompensationsdruck in hPa	nur wirksam bei Einheiten „% Vol“ oder „% SAT“

6.6 Konfiguration Analogausgang

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung / Hinweis
Dez.	Hex.					
8769	2241	enum	1	r/w	Funktion	0 = inaktiv 1 = Festwert 2 = Einheitssignal 4 bis 20 mA
8770	2242	float	2	r/w	Skalierungsanfang	Messwert bei 4 mA
8772	2244	float	2	r/w	Skalierungsende	Messwert bei 20 mA
8776	2248	float	2	r/w	Festwert	nur wenn „Funktion“ = 1
8778	224A	char	1	r/w	Warnausgabe aktiv	Boolwert für Warnausgabe true = aktiv, false = inaktiv
8779	224B	char	1	r/w	Fehlerausgabe aktiv	Boolwert für Fehlerausgabe true = aktiv, false = inaktiv
8780	224C	float	2	r/w	Warnwert	Ausgabewert von 3,5 mA bis 21,5 mA für Warnausgabe
8782	224E	float	2	r/w	Fehlerwert	Ausgabewert von 3,5 mA bis 21,5 mA für Fehlerausgabe
8784	2250	float	2	r/w	Ausgabewert bei Temperatur außerhalb erlaubtem Bereich	Ausgabewert von 3,5 mA bis 21,5 mA für Temperatur außerhalb erlaubtem Bereich

6 Modbus-Adresstabellen

6.7 Betriebsdaten

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1024	0400	Uint32	2	r	Betriebsstundenzähler	Betriebsdauer in Sekunden
1026	0402	short	1	r/w	Sensorwechselzähler	

6.7.1 Erstinbetriebnahme

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1280	0500	short	1	r	Jahr	
1281	0501	char	1	r	Monat	
1282	0502	char	1	r	Tag	
1283	0503	char	1	r	Stunde	
1284	0504	char	1	r	Minute	
1285	0505	char	1	r	Sekunde	
1286	0506	char	1	r	Zeitzone der Erstinbetriebnahme	
1287	0507	Uint32	2	r	Betriebsstundenzählerstand bei Erstinbetriebnahme	Betriebsdauer in Sekunden

6.7.2 Schleppzeigerfunktion

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1289	0509	float	2	r	niedrigster Temperaturwert	
1291	050B	float	2	r	höchster Temperaturwert	

Datum niedrigster Temperaturwert

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1293	050D	short	1	r	Jahr	
1294	050E	char	1	r	Monat	
1295	050F	char	1	r	Tag	
1296	0510	char	1	r	Stunde	
1297	0511	char	1	r	Minute	
1298	0512	char	1	r	Sekunde	
1299	0513	char	1	r	Zeitzone	
1300	0514	Uint32	2	r	Betriebsstundenzählerstand	Betriebsdauer in Sekunden

Datum höchster Temperaturwert

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1302	0516	short	1	r	Jahr	
1303	0517	char	1	r	Monat	
1304	0518	char	1	r	Tag	
1305	0519	char	1	r	Stunde	
1306	051A	char	1	r	Minute	
1307	051B	char	1	r	Sekunde	
1308	051C	char	1	r	Zeitzone	
1309	051D	Uint32	2	r	Betriebsstundenzählerstand	Betriebsdauer in Sekunden

6 Modbus-Adresstabellen

6.8 Prozesswerte

6.8.1 Kompensation

Es gibt 2 Quellen für Kompensationswerte, die für die Berechnung des Sauerstoffmesswertes herangezogen werden können:

- **Flashspeicher:** feste Kompensationswerte, die nichtflüchtig gespeichert werden und über Netz-Aus erhalten bleiben. Beachten Sie, dass Flashspeicher nicht zyklisch beschrieben werden dürfen. Hierfür sind die Kompensationsdaten im RAM vorgesehen.
- **RAM-Speicher:** Diese Kompensationswerte können vom Mastergerät zyklisch geschrieben werden, um aktuelle Kompensationswerte von externen Quellen (in der Regel Messdaten, die von weiteren Sensoren am Mastergerät stammen) zu erhalten. Hier befindet sich auch das „Control-Byte“ zur Steuerung der Kompensationsquellenwahl.

Die festen Kompensationswerte aus dem Flashspeicher dienen als Initialisierungswert der Kompensationswerte im RAM. Sie können aber auch als fest vorgegebene Kompensationswerte für die Sauerstoffmessung genutzt werden.

Die Auswahl, der für die Messung heranzuziehenden Kompensationsdaten zwischen **Flash** (feststehende Werte) oder **RAM** (Messwerte externer Quellen) muss mit dem **Control-Byte** gesteuert werden.

feste Kompensationswerte im Flashspeicher

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
9600	2580	float	2	r/w	fester Kompensationssalzgehalt	
9602	2582	float	2	r/w	fester Umgebungsdruck	

Kompensationswerte im RAM-Speicher

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
9647	25AF	word	1	r/w	Control-Byte Steuerung der Kompensationsquellenwahl	Low-Byte des Registers bestehend aus booleschen Werten; Bedeutung der einzelnen Bits, siehe nachfolgende Tabelle
9648	25B0	float	2	r/w	Kompensationstemperatur in °C	
9650	25B2	float	2	r/w	Kompensationssalzgehalt	
9652	25B4	float	2	r/w	Umgebungsdruck	

6 Modbus-Adresstabellen

Registerinhalt des Control-Byte

Bit-Nummer	Bedeutung
0 (lsb)	Quelle Kompensationstemperatur: False = interner Temperatursensor True = RAM-Kompensationstemperatur
1	Quelle Kompensationssalzgehalt: False = fester Kompensationssalzgehalt aus Flash True = RAM-Kompensationssalzgehalt
2	Quelle Umgebungsdruck: False = fester Umgebungsdruck aus Flash True = RAM-Umgebungsdruck
3	Reserve
4 bis 7 (msb)	nicht relevant für Modbus (Steuerung der zyklischen Übertragung der Kompensationswerte von JUMO-digiLine-Mastergeräten)
8 bis 15	-

6 Modbus-Adresstabellen

6.8.2 Messwerte

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
9728	2600	float	2	r	Sauerstoffmesswert (Einheit wie in Konfiguration Messung eingestellt)	
9760	2620	float	2	r	Temperaturmesswert in °C	

6.8.3 Statuswort

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
9792	2640	word	1	r	Statuswort	16-Bit-Wort bestehend aus booleschen Werten Bedeutung der einzelnen Bits, siehe nachfolgende Tabelle

Registerinhalt Statuswort

Bit-Nummer	Bedeutung
0 (lsb)	True = Warnung Sauerstoffmesswert ungültig
1	True = Warnung Temperaturmesswert ungültig
2	True = Alarm Zustand Sensorelement, Kappe fehlt
3	Reserve
4	Reserve
5	Reserve
6	Reserve
7	Reserve
8	True = Warnung Sauerstoff außerhalb Messbereich
9	True = Warnung Temperatur außerhalb Messbereich
10	True = Warnung Zustand Sensorelement, Tausch der Kappe wird empfohlen
11	Reserve
12	Reserve
13	Reserve
14	Reserve
15 (msb)	Reserve

6.9 Datum und Uhrzeit für Zeitstempel

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
10304	2840	short	1	r/w	Jahr	
10305	2841	char	1	r/w	Monat	
10306	2842	char	1	r/w	Tag	
10307	2843	char	1	r/w	Stunde	
10308	2844	char	1	r/w	Minute	
10309	2845	char	1	r/w	Sekunde	
10310	2846	short	1	r/w	Zeitzone	Offset zur UTC in Minuten

6 Modbus-Adresstabelle



JUMO GmbH & Co. KG

Moritz-Juchheim-Straße 1
36039 Fulda, Germany

Telefon: +49 661 6003-714
Telefax: +49 661 6003-605
E-Mail: mail@jumo.net
Internet: www.jumo.net

Lieferadresse:
Mackenrodtstraße 14
36039 Fulda, Germany

Postadresse:
36035 Fulda, Germany

Technischer Support Deutschland:

Telefon: +49 661 6003-9135
Telefax: +49 661 6003-881899
E-Mail: support@jumo.net

JUMO Mess- und Regelgeräte GmbH

Pfarrgasse 48
1230 Wien, Austria

Telefon: +43 1 610610
Telefax: +43 1 6106140
E-Mail: info.at@jumo.net
Internet: www.jumo.at

Technischer Support Österreich:

Telefon: +43 1 610610
Telefax: +43 1 6106140
E-Mail: info.at@jumo.net

JUMO Mess- und Regeltechnik AG

Laubisrütistrasse 70
8712 Stäfa, Switzerland

Telefon: +41 44 928 24 44
Telefax: +41 44 928 24 48
E-Mail: info@jumo.ch
Internet: www.jumo.ch

Technischer Support Schweiz:

Telefon: +41 44 928 24 44
Telefax: +41 44 928 24 48
E-Mail: info@jumo.ch

