

JUMO digiLine CR/Ci

Intelligente Elektronik mit digitaler Schnittstelle für
Leitfähigkeitssensoren von JUMO



Schnittstellenbeschreibung Modbus



20276000T92Z000K000

V1.00/DE/00726128/2019-10-09

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	12
2.6.1	Ganzzahl-Werte	13
2.6.2	Float-Werte	13
2.6.3	Zeichenketten (Texte)	15
2.7	Checksumme (CRC16)	16
2.8	Fehlermeldungen	17
2.8.1	Modbus-Fehlercodes	17
2.8.2	Fehlermeldungen bei ungültigen Werten	18
3	Schnittstellen	19
3.1	Schnittstellenbelegung	19
3.1.1	Abschlusswiderstände	19
4	Schnittstellen konfigurieren	21
5	Modbus über serielle Schnittstelle	23
5.1	Modbus-Slave-Betrieb über serielle Schnittstelle RS 485	23
6	Modbus-Adresstabellen	25
6.1	Typenschild	25
6.2	Messstelleninfo	26
6.3	Konfiguration Schnittstelle	26
6.4	Kalibrierdaten	27
6.4.1	Kalibrierlogbucheintrag 1	27
6.4.2	Kalibrierlogbucheintrag 2	28
6.4.3	Kalibrierlogbucheintrag 3	29
6.4.4	Kalibrierlogbucheintrag 4	30
6.4.5	Kalibrierlogbucheintrag 5	31
6.4.6	Kalibrierlogbucheintrag 6	33
6.4.7	Kalibrierlogbucheintrag 7	34

Inhalt

6.4.8	Kalibrierlogbucheintrag 8	.35
6.4.9	Kalibrierlogbucheintrag 9	.36
6.4.10	Kalibrierlogbucheintrag 10	.37
6.4.11	Relative Zellenkonstanten	.38
6.5	Konfiguration Leitfähigkeitseingang CR	.38
6.6	Konfiguration Leitfähigkeitseingang Ci	.40
6.7	Konfiguration Temperatureingang	.42
6.8	Konfiguration Kalibriertimer	.42
6.9	Konfiguration Sensorstress	.42
6.10	Konfiguration Sensorüberwachung	.42
6.11	Konfiguration Display	.43
6.12	Prozesswerte	.45
6.12.1	Eingänge	.45
6.12.2	Kalibrieralarm	.46
6.12.3	Sensorstress	.46
6.12.4	Sensorüberwachung	.46
6.12.5	Datum und Uhrzeit für Zeitstempel	.47
6.12.6	Zusammenfassung wichtiger Prozesswerte	.47

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 **Geräte-Dokumentation** 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!

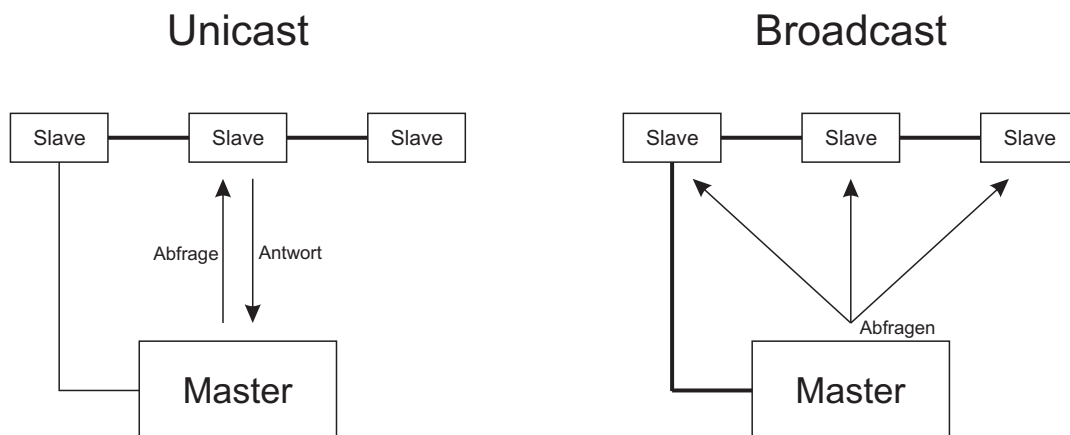
Dieses Gerät und, falls vorhanden, Batterien gehören nach Beendigung der Nutzung nicht in die Mülltonne! Bitte lassen Sie sie ordnungsgemäß und **umweltschonend entsorgen**.

1 Sicherheitshinweise

2.1 Master-Slave-Prinzip

Die Kommunikation zwischen einem Master (z. B. SCADA-System oder SPS) und einem JUMO digiLine CR/Ci 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 23



Der Master steuert den Datenaustausch durch zyklische Anfragen der Slaves im gesamten Bus. Die Slaves (z. B. JUMO digiLine CR/Ci) 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 CR/Ci 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 CR/Ci unterstützt nur den **RTU-Modus**. Hierbei werden die Daten im Binärformat über den seriellen Bus (RS422/485) übertragen.

⇒ Kapitel 5 „Modbus über serielle Schnittstelle“, Seite 23

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 25 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 für den Wert „Dauer bis CIP-Erkennung“ des Gerätes geschrieben werden. Die Slave-Adresse des Gerätes ist hier 1, die Wortadresse ist 0x5A42 (Kapitel 6.10 „Konfiguration Sensorüberwachung“, Seite 42) und der zu schreibende Wert soll „2700“ (2700 s entsprechend 45 min) sein.

Hex-Code der Anweisung:

01	06	5A 42	0A 8C	3C 03
Slave	Funktion	Wortadresse	Wert	CRC

Hex-Code der Antwort:

01	06	5A 42	0A 8C	3C 03
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 6.2 „Messstelleninfo“, Seite 26

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
char	Low-Byte eines Wortes als ganzzahliger Wert; das High-Byte wird nicht verwendet. Wertebereiche bei numerischer Interpretation: 0 bis 255 für vorzeichenlose Daten -128 bis 127 für vorzeichenbehaftete Daten Bei Interpretation als 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	
short	Wort (16 Bit) als ganzzahliger Wert Wertebereiche: 0 bis 65535 für vorzeichenlose Daten -32768 bis 32767 für vorzeichenbehaftete Daten	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="width: 100%; text-align: center;"> <tr> <th>Byte 1</th> <th>Byte 2</th> <th>Byte 3</th> <th>Byte 4</th> </tr> <tr> <td>SE7EEEEEE1</td> <td>E0M23MMMMMM16</td> <td>M15MMMMMMM8</td> <td>M7MMMMMMM0</td> </tr> </table> <p style="text-align: center;">IEEE 754 Little Endian</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <th>Byte 1</th> <th>Byte 2</th> <th>Byte 3</th> <th>Byte 4</th> </tr> <tr> <td>M7MMMMMMM0</td> <td>M15MMMMMMM8</td> <td>E0M23MMMMMM16</td> <td>SE7EEEEEE1</td> </tr> </table> <p style="text-align: center;">Standard-Modbus-Kodierung</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <th>Byte 1</th> <th>Byte 2</th> <th>Byte 3</th> <th>Byte 4</th> </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																															
long	<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																													
string[n]	<p>Zeichenkette für eine Länge von bis zu n Bytes. Die Zeichenkodierung der einzelnen Variablen ist den Modbus-Adresstabellen 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“ (ASCII-Code 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																													

2.6 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

2 Modbus-Protokollbeschreibung

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	Wertwert(e)	Checksumme CRC
1 Byte	1 Byte	1 Byte	x Bytes	2 Bytes

2.6.1 Ganzzahl-Werte

Beispiel

In diesem Beispiel soll die Dauer der CIP-Erkennung an Adresse 0x5A42 (Kapitel 6.10 „Konfiguration Sensorüberwachung“, Seite 42) ausgelesen werden. Der Wert soll hier „2400“ (Wort-Wert 0x0960) sein.

Datenanfrage:

01	03	5A 42	00 01	36 C6
Slave	Funktion	Adresse 1. Wort	Wortanzahl	CRC

Antwort (Werte im Modbus-Float-Format):

01	03	02	09 60	BE 3C
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
Byte 3	Byte 4	Byte 1	Byte 2

Beispiel

In diesem Beispiel soll der Wert „Temperaturwert CIP-Erkennung“ an Adresse 0x5A40 des Gerätes ausgelesen werden. Der Wert soll hier 70,25 (0x428C8000 im IEEE-754-Format) sein.

Datenanfrage:

01	03	5A 40	00 02	D7 07
Slave	Funktion	Adresse 1. Wort	Wortanzahl	CRC

2 Modbus-Protokollbeschreibung

Antwort (Werte im Modbus-Float-Format):

01	03	04	80 00	42 8C	E2 F6
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. Eventuell müssen die Bytes im Schnittstellenprogramm entsprechend vertauscht werden.

2 Modbus-Protokollbeschreibung

2.6.3 Zeichenketten (Texte)

Zeichenketten werden im UTF8-kodiert übertragen.



HINWEIS!

Als letztes Zeichen muss immer ein „\0“ (ASCII-Code 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.

Beispiel

Binärwert des Sensorstress-Voralarms an Adresse 0x2601 abfragen:

Anweisung: Lese ein Wort von Adresse 0x2601

01	03	26 01	00 01	DE 82
Slave	Funktion	Adresse	Ein Wort lesen	CRC

Antwort (CRC16 = 0x8479)

01	03	02	00 01	79 84
Slave	Funktion	Anzahl Bytes	Wort 1	CRC

Wort 1 = 1 bedeutet, dass der Binärwert des Voralarms für den Sensorstress = 1 ist.

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
$7,0 \times 10^{37}$	Ungültiger Float-Wert
$8,0 \times 10^{37}$	Integrator oder Statistik zerstört

Beispiel

Einlesen des kompensierten Leitfähigkeits-Messwertes an Modbus-Adresse 0x8002:

Datenanfrage:

08	03	80 02	00 02	4C 92
Slave	Funktion	Wortadresse	Wortanzahl	CRC

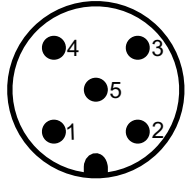
Antwort:

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.

3.1 Schnittstellenbelegung

5-poliger M12-Steckanschluss für RS485 (JUMO digiLine oder Modbus RTU)

Pin	Potenzial	Symbol
1	+5 V	
2	+ 24 V	
3	GND	
4	RS485 B (RxD/TxD-)	
5	RS485 A (RxD/TxD+)	
Der Anschluss an der seriellen Schnittstellen eines Mastergerätes mit Schraub- oder Federzugklemmen erfolgt mithilfe der JUMO digiLine-Geräteanschlussleitung (siehe Zubehör)		



HINWEIS!

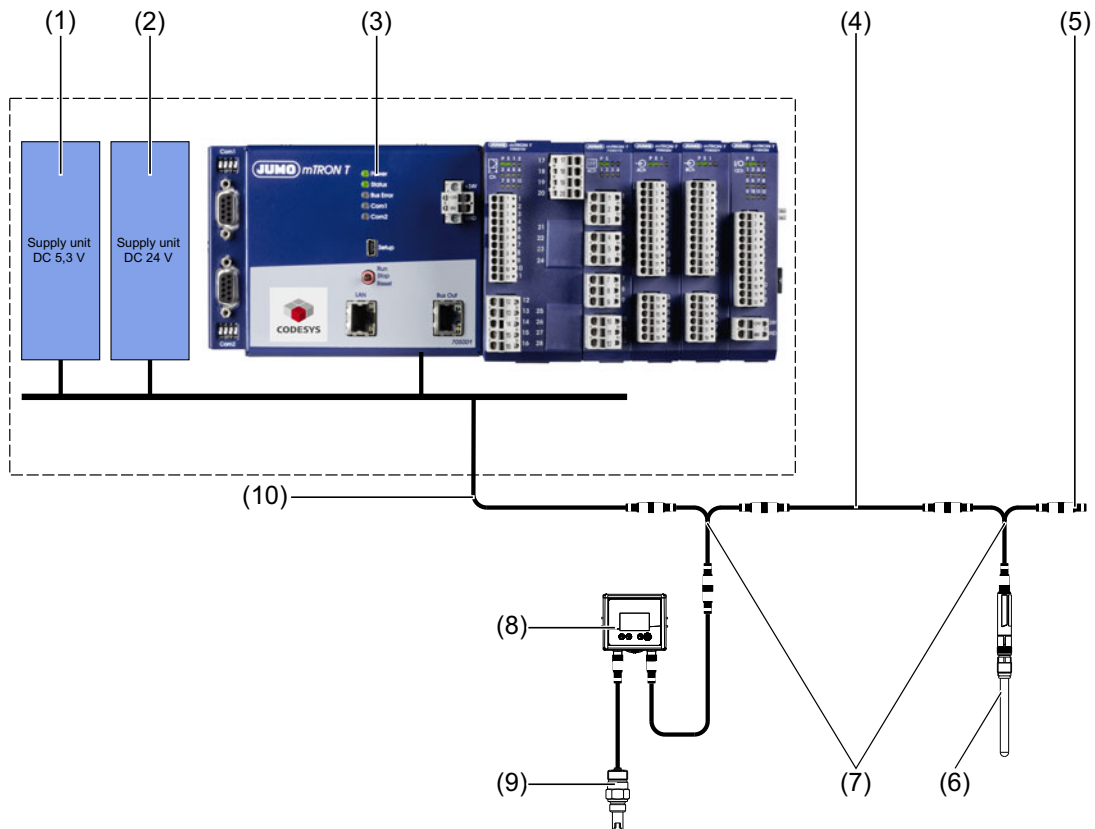
Zum Anschluss von Sensoren mit JUMO-digiLine-Elektronik sind konfektionierte Anschlussleitungen erhältlich. Verwenden Sie diese, um eine zuverlässige Leitungsverbindung und Datenübertragung im Bus-System zu gewährleisten. In den Bestellangaben der Betriebsanleitung finden Sie eine Auflistung, der als 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 JUMO 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 CR 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); für den Anschluss an das Modbus-System ist das im Folgenden aufgeführte Verdrahtungsschema zu berücksichtigen.

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.

In der Betriebsanleitung des JUMO digiLine CR/Ci ist die Bedienung, Konfiguration und Parametrierung detailliert erklärt.

- ⇒ Betriebsanleitung JUMO digiLine CR/Ci

4 Schnittstellen konfigurieren

Einstellungen für serielle Schnittstellen

Damit alle Teilnehmergeräte in einem Bus miteinander kommunizieren können, müssen ihre Schnittstellen-Einstellungen übereinstimmen. 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
Datenformat	8 - 1 - no Parity 8 - 1 - odd Parity 8 - 1 - even Parity 8 - 2 - no Parity	Format des Datenwortes 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.

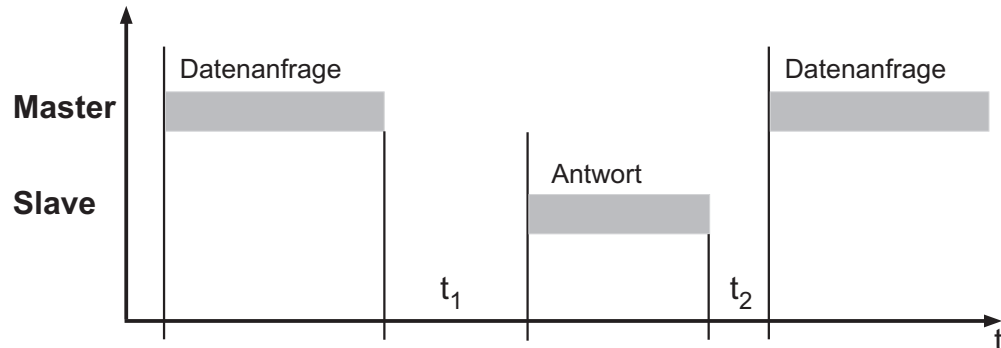
^b Unicast-Adressen sind für die Verwendung als Slave-Adressen vorgesehen. Sie dienen der eindeutigen Kennung der Slave-Geräte, damit diese vom Master explizit angesprochen werden können.

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 Stopppbit 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	
64	0040	char	1	r	Major-Version JUMOdigiLine-Protokoll	
65	0041	char	1	r	Minor-Version JUMOdigiLine-Protokoll	
128	0080	short	1	r	Hardwaretyp-Basis	
129	0081	short	1	r	Softwarenummer Messmikrocontroller	
130	0082	char	1	r	Major-Softwareversion Messmikrocontroller	
131	0083	char	1	r	Minor-Softwareversion Messmikrocontroller	
132	0084	char	1	r	Testversion Messmikrocontroller	
133	0085	char	1	r	Fehlerindex Messmikrocontroller	
134	0086	short	1	r	VDN-Version Messmikrocontroller	
135	0087	short	1	r	VDN-Nummer Messmikrocontroller	
136	0088	char	1	r	Hardwaretyp Messmikrocontroller	
137	0089	char	1	r	Hardware-Major-Version Messmikrocontroller	
138	008A	char	1	r	Hardware-Minor-Version Messmikrocontroller	

6 Modbus-Adresstabellen

6.2 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[108]	54	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)

6.3 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	
775	0307	enum	1	r/w	Floatwert-Format	Enumeration: 0 = Standard-Modbus-Format 1 = IEEE 754 Little Endian 2 = IEEE 754 Big Endian

6.4 Kalibrierdaten



HINWEIS!

In der JUMO-digiLine-Elektronik ist 1 Kalibrierlogbuch mit 10 Eintrags-Datensätzen hinterlegt. Die Adressen der Kalibrierlogbuchdaten sind in den folgenden Tabellen explizit angegeben. Da alle Kalibrierlogbücher gleich große Speicherbereiche haben, können Adressen auch berechnet werden. Die Größe des Adressbereiches eines Kalibrierlogbuchs beträgt 0x0031 in hexadezimaler bzw. 49 in dezimaler Zahlenangabe. Mit dem Vielfachen dieses Wertes können Sprünge aus einem Kalibrierlogbuch in andere berechnet werden.

Beispiel: $0x0600 + (n - 1) * 0x0031$ ergibt die Modbusadresse des Parameters „Eintragstyp“ des n-ten Kalibrierlogbuchs.

6.4.1 Kalibrierlogbucheintrag 1

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1536	0600	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient
1537	0601	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1538	0602	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1539	0603	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1541	0605	short	1	r	Jahr	
1542	0606	char	1	r	Monat	
1543	0607	char	1	r	Tag	
1544	0608	char	1	r	Stunde	
1545	0609	char	1	r	Minute	
1546	060A	char	1	r	Sekunden	
1547	060B	short	1	r	Zeitzone	
1548	060C	char	1	r	Sensorwechselfählerstand	
1549	060D	float[6]	12	r	Kalibrierwerte	

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1561	0619	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6.4.2 Kalibrierlogbucheintrag 2

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1585	0631	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient
1586	0632	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1587	0633	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1588	0634	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1590	0636	short	1	r	Jahr	
1591	0637	char	1	r	Monat	
1592	0638	char	1	r	Tag	
1593	0639	char	1	r	Stunde	
1594	063A	char	1	r	Minute	
1595	063B	char	1	r	Sekunden	
1596	063C	short	1	r	Zeitzone	
1597	063D	char	1	r	Sensorwechselfählerstand	

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1598	063E	float[6]	12	r	Kalibrierwerte	<p>Array-Elemente bei Eintragstyp „relative Zellenkonstante“: [0] = relative Zellenkonstante [1] = Messbereich</p> <p>Array-Elemente bei Eintragstyp „Temperaturkoeffizient“: [0] = Temperaturkoeffizient [1] = Temperatur 1 [2] = Temperatur 2 [3] = Messbereich</p>
1610	064A	string[6][8]	24	r	Einheiten	<p>Einheiten der Kalibrierwerte in folgendem Format:</p> <p>Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.</p>

6.4.3 Kalibrierlogbucheintrag 3

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1634	0662	enum	1	r	Eintragstyp	<p>Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient</p>
1635	0663	enum	1	r	manuelle Änderung	<p>Enumeration: 0 = manuell 1 = Kalibrierung</p>
1636	0664	enum	1	r	Bewertung der Kalibrierung	<p>Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung</p>

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1637	0665	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1639	0667	short	1	r	Jahr	
1640	0668	char	1	r	Monat	
1641	0669	char	1	r	Tag	
1642	066A	char	1	r	Stunde	
1643	066B	char	1	r	Minute	
1644	066C	char	1	r	Sekunden	
1645	066D	short	1	r	Zeitzone	
1646	066E	char	1	r	Sensorwechselzählerstand	
1647	066F	float[6]	12	r	Kalibrierwerte	<p>Array-Elemente bei Eintragstyp „relative Zellenkonstante“: [0] = relative Zellenkonstante [1] = Messbereich</p> <p>Array-Elemente bei Eintragstyp „Temperaturkoeffizient“: [0] = Temperaturkoeffizient [1] = Temperatur 1 [2] = Temperatur 2 [3] = Messbereich</p>
1659	067B	string[6][8]	24	r	Einheiten	<p>Einheiten der Kalibrierwerte in folgendem Format:</p> <p>Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.</p>

6.4.4 Kalibrierlogbucheintrag 4

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1683	0693	enum	1	r	Eintragstyp	<p>Enumeration:</p> <p>0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient</p>
1684	0694	enum	1	r	manuelle Änderung	<p>Enumeration:</p> <p>0 = manuell 1 = Kalibrierung</p>

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1685	0695	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1686	0696	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1688	0698	short	1	r	Jahr	
1689	0699	char	1	r	Monat	
1690	069A	char	1	r	Tag	
1691	069B	char	1	r	Stunde	
1692	069C	char	1	r	Minute	
1693	069D	char	1	r	Sekunden	
1694	069E	short	1	r	Zeitzone	
1695	069F	char	1	r	Sensorwechselzählerstand	
1696	06A0	float[6]	12	r	Kalibrierwerte	Array-Elemente bei Eintragstyp „relative Zellenkonstante“: [0] = relative Zellenkonstante [1] = Messbereich Array-Elemente bei Eintragstyp „Temperaturkoeffizient“: [0] = Temperaturkoeffizient [1] = Temperatur 1 [2] = Temperatur 2 [3] = Messbereich
1708	06AC	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6.4.5 Kalibrierlogbucheintrag 5

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1732	06C4	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1733	06C5	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1734	06C6	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1735	06C7	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1737	06C9	short	1	r	Jahr	
1738	06CA	char	1	r	Monat	
1739	06CB	char	1	r	Tag	
1740	06CC	char	1	r	Stunde	
1741	06CD	char	1	r	Minute	
1742	06CE	char	1	r	Sekunden	
1743	06CF	short	1	r	Zeitzone	
1744	06D0	char	1	r	Sensorwechselfählerstand	
1745	06D1	float[6]	12	r	Kalibrierwerte	
1757	06DD	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6 Modbus-Adresstabellen

6.4.6 Kalibrierlogbucheintrag 6

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1781	06F5	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient
1782	06F6	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1783	06F7	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1784	06F8	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1786	06FA	short	1	r	Jahr	
1787	06FB	char	1	r	Monat	
1788	06FC	char	1	r	Tag	
1789	06FD	char	1	r	Stunde	
1790	06FE	char	1	r	Minute	
1791	06FF	char	1	r	Sekunden	
1792	0700	short	1	r	Zeitzone	
1793	0701	char	1	r	Sensorwechselzählerstand	
1794	0702	float[6]	12	r	Kalibrierwerte	
1806	070E	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6 Modbus-Adresstabellen

6.4.7 Kalibrierlogbucheintrag 7

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1830	0726	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient
1831	0727	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1832	0728	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1833	0729	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1835	072B	short	1	r	Jahr	
1836	072C	char	1	r	Monat	
1837	072D	char	1	r	Tag	
1838	072E	char	1	r	Stunde	
1839	072F	char	1	r	Minute	
1840	0730	char	1	r	Sekunden	
1841	0731	short	1	r	Zeitzone	
1842	0732	char	1	r	Sensorwechselzählerstand	
1843	0733	float[6]	12	r	Kalibrierwerte	
1855	073F	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6 Modbus-Adresstabellen

6.4.8 Kalibrierlogbucheintrag 8

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1879	0757	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient
1880	0758	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1881	0759	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1882	075A	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1884	075C	short	1	r	Jahr	
1885	075D	char	1	r	Monat	
1886	075E	char	1	r	Tag	
1887	075F	char	1	r	Stunde	
1888	0760	char	1	r	Minute	
1889	0761	char	1	r	Sekunden	
1890	0762	short	1	r	Zeitzone	
1891	0763	char	1	r	Sensorwechselzählerstand	
1892	0764	float[6]	12	r	Kalibrierwerte	
1904	0770	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6 Modbus-Adresstabellen

6.4.9 Kalibrierlogbucheintrag 9

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1928	0788	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient
1929	0789	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1930	078A	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1931	078B	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1933	078D	short	1	r	Jahr	
1934	078E	char	1	r	Monat	
1935	078F	char	1	r	Tag	
1936	0790	char	1	r	Stunde	
1937	0791	char	1	r	Minute	
1938	0792	char	1	r	Sekunden	
1939	0793	short	1	r	Zeitzone	
1940	0794	char	1	r	Sensorwechselzählerstand	
1941	0795	float[6]	12	r	Kalibrierwerte	
1953	07A1	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6 Modbus-Adresstabellen

6.4.10 Kalibrierlogbucheintrag 10

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
1977	07B9	enum	1	r	Eintragstyp	Enumeration: 0 = kein Eintrag 1 = relative Zellenkonstante 2 = Temperaturkoeffizient
1978	07BA	enum	1	r	manuelle Änderung	Enumeration: 0 = manuell 1 = Kalibrierung
1979	07BB	enum	1	r	Bewertung der Kalibrierung	Enumeration: 0 = keine Bewertung 1 = OK 2 = Warnung
1980	07BC	long	2	r	Betriebstundenzählerstand	Datum-/Zeitstempel
1982	07BE	short	1	r	Jahr	
1983	07BF	char	1	r	Monat	
1984	07C0	char	1	r	Tag	
1985	07C1	char	1	r	Stunde	
1986	07C2	char	1	r	Minute	
1987	07C3	char	1	r	Sekunden	
1988	07C4	short	1	r	Zeitzone	
1989	07C5	char	1	r	Sensorwechselzählerstand	
1990	07C6	float[6]	12	r	Kalibrierwerte	
2002	07D2	string[6][8]	24	r	Einheiten	Einheiten der Kalibrierwerte in folgendem Format: Für jede Einheit der 6 Kalibrierwerte stehen 8 Bytes für die Einheiten-Zeichenkette inklusive dem NULL-Zeichen als Endekennung in UTF-8-Kodierung zur Verfügung. Die Zeichenkette darf bis zu 5 Zeichen lang sein.

6 Modbus-Adresstabellen

6.4.11 Relative Zellenkonstanten

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
25600	6400	float[4]	8	r/w	relative Ci-Zellenkonstante je Messbereich	Die Array-Elemente 0 bis 3 entsprechen den relativen Zellenkonstanten der Messbereiche 1 bis 4
25608	6408	float	2	r/w	relative Ci-Zellenkonstante für alle Messbereiche	
25610	640A	float[4]	8	r/w	relative CR-Zellenkonstante je Messbereich	Die Array-Elemente 0 bis 3 entsprechen den relativen Zellenkonstanten der Messbereiche 1 bis 4
25618	6412	float	2	r/w	relative CR-Zellenkonstante für alle Messbereiche	
25620	6414	float	2	r/w	relative Zellenkonstante CR junior	

6.5 Konfiguration Leitfähigkeitseingang CR

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
20738	5102	enum	1	r/w	Kompensationsquelle	Enumeration: 0 = manuelle Temperatur 1 = Temperatureingang 2 = Schnittstelle
20739	5103	enum	1	r/w	Modus relative Zellenkonstante	Enumeration: 0 = eine Zellenkonstante für alle Messbereiche 1 = separate relative Zellenkonstante für jeden Messbereich
20741	5105	enum	1	r/w	Leitungsbruchererkennung	Enumeration: 0 = Aus 1 = Ein
20742	5106	float	2	r/w	manuelle Temperatur	
20744	5108	float	2	r/w	Bezugstemperatur für lin. TK und TK-Kurve	
20746	510A	float	2	r/w	Filterzeit	
20748	510C	float	2	r/w	nominelle Zellenkonstante	
20750	510E	enum[4]	4	r/w	Aktivierung Linearisierungstabelle	Array-Elemente [0] bis [3] für Messbereiche 1 bis 4 Enumeration: 0 = Nein 1 = Ja

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
20754	5112	enum[4]	4	r/w	Einheit für Berechnung	Array-Elemente [0] bis [3] für Messbereiche 1 bis 4 Enumeration: 0 = $\mu\text{S}/\text{cm}$ 1 = mS/cm 2 = $\text{k}\Omega \times \text{cm}$ 3 = $\text{M}\Omega \times \text{cm}$
20758	5116	T_KOMP_CR[4]	4	r/w	Kompensation	Die Array-Elemente [0] bis [3] entsprechen den Enumerationen für die Kompensationseinstellungen der Messbereiche 1 bis 4. Enumeration: 0 = Keine 1 = TK-Linear 2 = TK-Kurve 3 = natürliche Wasser 4 = natürliche Wasser mit erweitertem Temperaturbereich 5 = TDS 6 = ASTM neutral 7 = ASTM sauer 8 = ASTM alkalisch
20762	511A	float[4]	8	r/w	TK	Die Array-Elemente [0] bis [3] entsprechen den Temperaturkoeffizienten der Messbereiche 1 bis 4
20770	5122	float[4]	8	r/w	Offset	Die Array-Elemente [0] bis [3] entsprechen den Offsets der Messbereiche 1 bis 4
20778	512A	float[4]	8	r/w	TDS-Faktor	Die Array-Elemente [0] bis [3] entsprechen den TDS-Faktoren der Messbereiche 1 bis 4
20786	5132	string[4][6]	12	r/w	Einheitentext bei TDS und kundenspezifischer Linearisierung mit Linearisierungstabellen	Die Array-Elemente [0][x] bis [3][x] entsprechen den TDS-Einheitentexten der Messbereiche 1 bis 4 in ISO-8859-1-Kodierung mit bis zu 6 Zeichen inklusive „NULL“ als Ende-Kennung.

6 Modbus-Adresstabellen

6.6 Konfiguration Leitfähigkeitseingang Ci

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
20994	5202	enum	1	r/w	Kompensationsquelle	Enumeration: 0 = manuelle Temperatur 1 = Temperatureingang 2 = Schnittstelle
20995	5203	enum	1	r/w	Modus relative Zellenkonstante	Enumeration: 0 = eine Zellenkonstante für alle Messbereiche 1 = separate relative Zellenkonstante für jeden Messbereich
20996	5204	float	2	r/w	manuelle Temperatur	
20998	5206	float	2	r/w	Bezugstemperatur für lin. TK und TK-Kurve	
21000	5208	float	2	r/w	Filterzeit	
21002	520A	float	2	r/w	nominelle Zellenkonstante	
21004	520C	float	2	r/w	Einbaufaktor	
21006	520E	enum[4]	4	r/w	Aktivierung Linearisierungstabelle	Array-Elemente [0] bis [3] für Messbereiche 1 bis 4 Enumeration: 0 = Nein 1 = Ja
21010	5212	enum[4]	4	r/w	Einheit für Berechnung	Array-Elemente [0] bis [3] für Messbereiche 1 bis 4 Enumeration: 0 = $\mu\text{S/cm}$ 1 = mS/cm 2 = $\text{k}\Omega \times \text{cm}$ 3 = $\text{M}\Omega \times \text{cm}$

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
21014	5216	enum[4]	4	r/w	Kompensation	<p>Array-Elemente [0] bis [3] für Messbereiche 1 bis 4</p> <p>Enumeration:</p> <p>0 = Keine 1 = TK-Linear 2 = TK-Kurve 3 = natürliche Wässer 4 = natürliche Wässer mit erweitertem Temperaturbereich 5 = TDS 6 = NaOH 0 bis 12 % 7 = NaOH 25 bis 50 % 8 = HNO₃ 0 bis 25 % 9 = HNO₃ 36 bis 82 % 10 = H₂SO₄ 0 bis 28 % 11 = H₂SO₄ 36 bis 85 % 12 = H₂SO₄ 92 bis 99 % 13 = HCl 0 bis 18 % 14 = HCl 22 bis 44 % 15 = NaCl 0 bis 25 % 16 = MgCl₂ 0 bis 17,5 % 17 = MgCl₂ 18,5 bis 25 %</p>
21018	521A	float[4]	8	r/w	TK	Die Array-Elemente [0] bis [3] entsprechen den Temperaturkoeffizienten der Messbereiche 1 bis 4
21026	5222	float[4]	8	r/w	Offset	Die Array-Elemente [0] bis [3] entsprechen den Offsets der Messbereiche 1 bis 4
21034	522A	float[4]	8	r/w	TDS-Faktor	Die Array-Elemente [0] bis [3] entsprechen den TDS-Faktoren der Messbereiche 1 bis 4
21042	5232	string[4][6]	12	r/w	Einheitentext bei TDS und kundenspezifischer Linearisierung mit Linearisierungstabellen	Die Array-Elemente [0][x] bis [3][x] entsprechen den TDS-Einheitentexten der Messbereiche 1 bis 4 in ISO-8859-1-Kodierung mit bis zu 6 Zeichen inklusive „NULL“ als Ende-Kennung.

6 Modbus-Adresstabellen

6.7 Konfiguration Temperatureingang

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
21504	5400	float	2	r/w	Filterzeit	Einheit: Sekunden
21506	5402	float	2	r/w	Offset	
21509	5405	enum	1	r/w	Anschlussart	Enumeration: 0 = 2-Leiter 1 = 3-Leiter
21510	5406	enum	1	r/w	Aktivierung Temperatureingang	Enumeration: 0 = inaktiv 1 = aktiv

6.8 Konfiguration Kalibriertimer

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
22784	5900	short	1	r/w	Kalibrierintervall	Einheit: Tage

6.9 Konfiguration Sensorstress

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
23040	5A00	enum	1	r/w	Aktivierung Sensorstressalarm	Enumeration: 0 = inaktiv 1 = aktiv

6.10 Konfiguration Sensorüberwachung

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
23104	5A40	float	2	r/w	CIP-Temperatur	
23106	5A42	short	1	r/w	CIP-Dauer	Einheit: Tage
23107	5A43	float	2	r/w	SIP-Temperatur	
23109	5A45	short	1	r/w	SIP-Dauer	Einheit: Tage
23110	5A46	enum	1	r/w	Alarm Sensorüberwachung	Enumeration: 0 = aktiv 1 = inaktiv
23111	5A47	short	1	r/w	Höchstanzahl an CIP-Zyklen bis zum Alarm	
23112	5A48	short	1	r/w	Höchstanzahl an SIP-Zyklen bis zum Alarm	
23113	5A49	short	1	r/w	Höchstanzahl an CIP-Zyklen bis zum Voralarm	

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
23114	5A4A	short	1	r/w	Höchstanzahl an SIP-Zyklen bis zum Voralarm	

6.11 Konfiguration Display

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
23809	5D01	enum	1	r/w	Sprache	Enumeration: 0 = Deutsch 1 = Englisch 2 = Französisch 3 = Spanisch
23810	5D02	short	1	r/w	Zeit automatische Abmeldung	Einheit: Minuten 0= keine automatische Abmeldung
23811	5D03	enum	1	r/w	Anzeigeart	Enumeration: 0 = Standardanzeige 1 = Großanzeige 2 = Bargraph
23812	5D04	enum	1	r/w	Temperatureinheit	Enumeration: 0 = °C 1 = °F
23813	5D05	float[4]	8	r/w	Bargraph Anfang	Die Array-Elemente [0] bis [3] entsprechen den Anfangswerten des Bargraphs für die Messbereiche 1 bis 4
23821	5D0D	float[4]	8	r/w	Bargraph Ende	Die Array-Elemente [0] bis [3] entsprechen den Endwerten des Bargraphs für die Messbereiche 1 bis 4
23829	5D15	enum[4]	4	r/w	Kommaformat	Array-Elemente [0] bis [3] für Messbereiche 1 bis 4 Enumeration: 0 = XXXX 1 = XXX.x 2 = XX.xx 3 = X.xxx
23835	5D1B	short	1	r/w	Displaykontrast	

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
23836	5D1C	enum	1	r/w	Signal für Hauptwert-anzeige	Enumeration 0 = Keine 1 = Temperatureingang 2 = Kompensationstemperatur 3 = unkompensierte Leitfähigkeit 4 = kompensierte Leitfähigkeit
23837	5D1D	enum	1	r/w	Signal für Nebenwert-anzeige	Enumeration 0 = Keine 1 = Temperatureingang 2 = Kompensationstemperatur 3 = unkompensierte Leitfähigkeit 4 = kompensierte Leitfähigkeit

6 Modbus-Adresstabellen

6.12 Prozesswerte

6.12.1 Eingänge

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
8192	2000	float	2	r	unkompensierter Leitfähigkeitsmesswert in Mikro-Siemens	
8194	2002	float	2	r	unkompensierter Leitfähigkeitsmesswert	
8196	2004	float	2	r	kompensierter Leitfähigkeitsmesswert	
8198	2006	float	2	r	Kompensationstemperaturwert	
8200	2008	short	1	r	aktueller Messbereich des Leitfähigkeitsmesswertes	
8202	200A	char	1	r	Kompensationsbereich verlassen	
8203	200B	char	1	r	Ungültige Leitfähigkeit	0 = kein Fehler vorhanden 1 = ungültige Leitfähigkeit
8204	200C	string[6]	3	r	Einheitentext unkomp. Leitfähigkeit	ASCII-Kodierung mit bis zu 6 Zeichen inklusive „NULL“ als Ende-Kennung. Die Byte-Anzahl muss gerade sein (ggf. mit einem weiteren NULL-Zeichen auffüllen)
8207	200F	string[6]	3	r	Einheitentext komp. Leitfähigkeit	ASCII-Kodierung mit bis zu 6 Zeichen inklusive „NULL“ als Ende-Kennung. Die Byte-Anzahl muss gerade sein (ggf. mit einem weiteren NULL-Zeichen auffüllen)
8210	2012	short	1	r/w	aktueller Messbereich des Leitfähigkeitsmesswertes über Schnittstelle	0 = Messbereich 1 1 = Messbereich 2 2 = Messbereich 3 3 = Messbereich 4
8211	2013	float	2	r/w	Kompensationstemperatur Schnittstelle	Kompensationstemperatur, die vom JUMO-digiLine-Mastergerät an den JUMO digiLine CR/Ci übertragen wird
8450	2102	float	2	r	Messwert Temperatureingang in °C	Temperaturmesswert des JUMO digiLine CR/Ci in °C
8452	2104	float	2	r	Messwert Temperatureingang in °F	Temperaturmesswert des JUMO digiLine CR/Ci in °F
8454	2106	char	1	r	Alarm Temperatureingang	0 = kein Fehler vorhanden 1 = Alarm Temperatureingang

6 Modbus-Adresstabellen

6.12.2 Kalibrieralarm

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
9472	2500	char	1	r	Kalibrieralarm	

6.12.3 Sensorstress

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
9728	2600	char	1	r	Alarmzustand Sensorstress	Der Sensorstresslevel gibt die thermische und chemische Beanspruchung des Sensors als Index in Abhängigkeit von Temperatur- und Leitfähigkeitsmesswert wieder. Mehr hierzu finden Sie in der Betriebsanleitung des JUMO digi-Line CR/Ci.
9729	2601	char	1	r	Vor-Alarmzustand Sensorstress	
9730	2602	short	1	r	aktueller Sensorstresslevel	

6.12.4 Sensorüberwachung

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
9792	2640	char	1	r	CIP-/SIP-Alarm	
9793	2641	char	1	r	CIP-/SIP-Voralarm	

6 Modbus-Adresstabellen

6.12.5 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

6.12.6 Zusammenfassung wichtiger Prozesswerte



HINWEIS!

Die Variablen aus der zusammengefassten Prozesswerte werden während der Bootphase auf ungültig gesetzt, um zu verhindern, dass instabile Messwerte Probleme bei Master-Geräten hervorrufen, die auf diese Messwerte zugreifen. Im Gegensatz dazu werden die Original-Variablen immer transparent (also auch schwingend) abgebildet.

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
32768	8000	float	2	r	unkompensierter Leitfähigkeitsmesswert	
32770	8002	float	2	r	kompensierter Leitfähigkeitsmesswert	
32772	8004	float	2	r	Temperatur in °C	
32774	8006	short	1	r	aktiver Parametersatz	
32775	8007	short	1	r	Statuswort	Das Statuswort ist ein Bit-Feld bestehend aus 16 Statusbits. Der Aufbau des Statuswortes ist in der nachfolgenden Tabelle dargestellt.
32776	8008	short	1	r	Sensorstresslevel	Der Sensorstresslevel gibt die thermische und chemische Beanspruchung des Sensors als Index in Abhängigkeit von Temperatur- und Leitfähigkeitsmesswert wieder. Mehr hierzu finden Sie in der Betriebsanleitung des JUMO digi-Line CR/Ci.
32777	8009	long	2	r	CIP-Zyklenzähler	
32779	800B	long	2	r	SIP-Zyklenzähler	

6 Modbus-Adresstabellen

Modbus-PDU-Adresse		Datentyp	Anzahl Modbus-Register	Zugriff	Daten	Kodierung
Dez.	Hex.					
32781	800D	long	2	r	summierte Gesamtdauer der Verletzung des unteren Temperaturgrenzwertes	Einheit: Sekunden
32783	800F	long	2	r	summierte Gesamtdauer der Verletzung des oberen Temperaturgrenzwertes	
32785	8011	long	2	r	summierte Gesamtdauer der Verletzung des oberen Leitfähigkeitsgrenzwertes	
32787	8013	long	2	r	Anzahl der Verletzungen des unteren Temperaturwertes	
32789	8015	long	2	r	Anzahl der Verletzungen des oberen Temperaturwertes	
32791	8017	long	2	r	Anzahl der Verletzungen des oberen Leitfähigkeitsgrenzwertes	

Aufbau des Statuswortes und Bedeutung dessen 16-Statusbits

Bit-Nummer	#16 (msb)	#15	#14	#13	#12	#11	#10	#9
Funktion	Kalibrierdatenfehler	Konfigurationsdatenfehler	-	-	-	-	-	Alarm Betriebsdaten erfassung

#8	#7	#6	#5	#4	#3	#2	#1 (lsb)
Kalibriertimer abgelaufen	Alarm Sensorstress	Voralarm Sensorstress	Alarm max. CIP-/SIP-Zyklen	Voralarm max. CIP-/SIP-Zyklen	Kompensations-temperaturbereich verlassen	Alarm ungültiger Temperaturmesswert	Alarm ungültiger Leitfähigkeitsmesswert



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: service@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

