

JUMO cTRON 04/08/16

Kompaktregler mit Timer und Rampenfunktion



702071



702072



702074

B 70.2070.2.0
Schnittstellenbeschreibung
Modbus

Inhalt

1	Einleitung	5
1.1	Vorwort	5
1.2	Typografische Konventionen	5
2	Protokollbeschreibung	7
2.1	Master-Slave-Prinzip	7
2.2	Übertragungsmodus (RTU)	7
2.3	Geräteadresse	8
2.4	Zeitlicher Ablauf der Kommunikation	8
2.5	Aufbau der Datenblöcke	11
2.6	Funktionscodes	12
2.6.1	Lesen von n Worten	12
2.6.2	Schreiben eines Wortes	13
2.6.3	Schreiben von n Worten	14
2.7	Übertragungsformat (Integer-, Float- und Text-Werte)	15
2.8	Checksumme (CRC16)	17
2.9	Fehlerbehandlung	18
3	RS485-Schnittstelle	19
3.1	Anschlussplan	19
3.2	Konfiguration	20
4	Modbus-Adressen	21
4.1	Prozessdaten	21
4.2	Sollwerte	23
4.3	Reglerparameter	23
4.4	Konfiguration	23
4.5	Kommandos	24
4.6	RAM-Speicher	25

1.1 Vorwort

Diese Anleitung wendet sich an den Anlagenhersteller mit fachbezogener Ausbildung und PC-Kenntnissen.

Lesen Sie diese Anleitung, bevor Sie mit Ihrer Arbeit am Gerät beginnen. Bewahren Sie die Anleitung an einem für alle Benutzer jederzeit zugänglichen Platz auf. Mit Ihren Anregungen können Sie uns helfen, diese Anleitung zu verbessern.

Alle erforderlichen Einstellungen sind in der vorliegenden Anleitung beschrieben. Sollten bei der Inbetriebnahme Schwierigkeiten auftreten, bitten wir Sie, keine eigenmächtigen Manipulationen vorzunehmen, die nicht in der Anleitung beschrieben sind. Sie gefährden dadurch Ihren Gewährleistungsanspruch. Bitte setzen Sie sich mit der nächsten Niederlassung oder dem Stammhaus in Verbindung.

1.2 Typografische Konventionen

Warnende Zeichen:

Achtung



Dieses Zeichen wird benutzt, wenn es durch ungenaues Befolgen oder Nichtbefolgen von Anweisungen zu **Beschädigungen von Geräten oder Daten** kommen kann!

Hinweisende Zeichen:

Hinweis



Dieses Zeichen wird benutzt, wenn Sie auf **etwas Besonderes** aufmerksam gemacht werden sollen.

Verweis



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

Darstellungsarten:

Hexadezimalzahl

0x0010

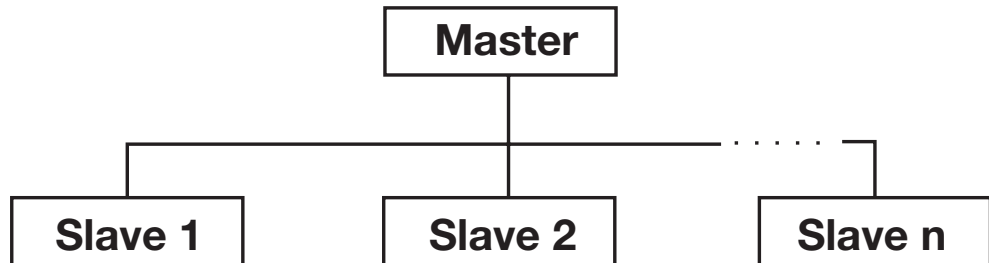
Eine Hexadezimalzahl wird durch ein vorangestelltes „0x“ gekennzeichnet (hier: 16 dezimal).

1 Einleitung

2 Protokollbeschreibung

2.1 Master-Slave-Prinzip

Die Kommunikation zwischen einem Master (z. B. PC) und einem Slave (z. B. Mess- und Regelsystem) mit Modbus findet nach dem Master-Slave-Prinzip in Form von Datenanfrage/Anweisung - Antwort statt.



Der Master steuert den Datenaustausch, die Slaves haben lediglich Antwortfunktion. Sie werden anhand ihrer Geräteadresse identifiziert.

2.2 Übertragungsmodus (RTU)

Als Übertragungsmodus wird der RTU-Modus (Remote Terminal Unit) verwendet. Die Übertragung der Daten erfolgt im Binärformat (hexadezimal) mit 8 Bits. Das LSB (least significant bit, engl. das niederwertigste Bit) wird zuerst übertragen. Die Betriebsart ASCII-Modus wird nicht unterstützt.

Datenformat

Mit dem Datenformat wird der Aufbau eines übertragenen Zeichens beschrieben. Es sind folgende Möglichkeiten des Datenformats gegeben:

Datenwort	Paritätsbit	Stoppbit 1/2 Bit	Bitanzahl
8 Bit	—	1	9
8 Bit	gerade (even)	1	10
8 Bit	ungerade (odd)	1	10
8 Bit	—	2	10

2 Protokollbeschreibung

2.3 Geräteadresse

Die Geräteadresse des Slaves ist zwischen 0 und 254 einstellbar. Die Geräteadresse 0 ist reserviert.



Über die RS485-Schnittstelle können maximal 31 Slaves angesprochen werden.

Es gibt zwei Varianten des Datenaustausches:

Query

Datenanfrage/Anweisung des Masters an einen Slave über die entsprechende Geräteadresse.

Der angesprochene Slave antwortet.

Broadcast

Anweisung des Masters an alle Slaves über die Geräteadresse 0 (z. B. zur Übertragung eines bestimmten Wertes an alle Slaves).

Die angeschlossenen Slaves antworten nicht. Die richtige Übernahme des Wertes durch die Slaves sollte in diesem Fall durch anschließendes Auslesen an jedem einzelnen Slave kontrolliert werden.

Eine Datenanfrage mit der Geräteadresse 0 ist nicht sinnvoll.

2.4 Zeitlicher Ablauf der Kommunikation

Anfang und Ende eines Datenblocks sind durch Übertragungspausen gekennzeichnet. Zwischen zwei aufeinanderfolgenden Zeichen darf maximal das Dreifache der Zeit zum Übertragen eines Zeichens vergehen.

Die Zeichenübertragungszeit (Zeit für die Übertragung eines Zeichens) ist abhängig von der Baudrate und dem verwendeten Datenformat (Stoppbits und Paritätsbit).

Bei einem Datenformat von 8 Datenbits, keinem Paritätsbit und einem Stoppbit ergibt sich:

$$\text{Zeichenübertragungszeit [ms]} = 1000 * 9 \text{ Bit/Baudrate}$$

Bei den anderen Datenformaten ergibt sich:

$$\begin{aligned} \text{Zeichenübertragungszeit [ms]} \\ = 1000 * (8 \text{ Bits} + \text{Paritätsbit} + \text{Stoppbit(s)}) \text{ Bit/Baudrate} \end{aligned}$$

2 Protokollbeschreibung

Ablauf

Datenanfrage vom Master Übertragungszeit = $n \text{ Zeichen} * 1000 * x \text{ Bit/Baudrate}$
Kennzeichen für Datenanfrage-Ende $3 \text{ Zeichen} * 1000 * x \text{ Bit/Baudrate}$
Bearbeitung der Datenanfrage durch den Slave ($\leq 250\text{ms}$)
Antwort des Slaves Übertragungszeit = $n \text{ Zeichen} * 1000 * x \text{ Bit/Baudrate}$
Kennzeichen für Antwort-Ende $3 \text{ Zeichen} * 1000 * x \text{ Bit/Baudrate}$

Beispiel

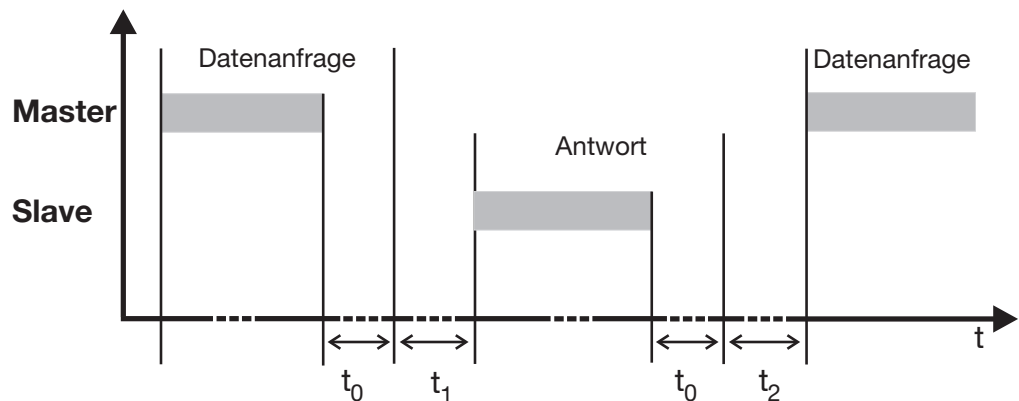
Kennzeichen für Datenanfrage- oder Antwort-Ende bei Datenformat 10/9 Bits
 Wartezeit = $3 \text{ Zeichen} * 1000 * 10 \text{ Bit/Baudrate}$

Baudrate [Baud]	Datenformat [Bit]	Wartezeit [ms] (3 Zeichen)
38400	10	0.79
	9	0.71
19200	10	1.57
	9	1.41
9600	10	3.13
	9	2.82

2 Protokollbeschreibung

Zeitschema

Eine Datenanfrage läuft nach folgendem Zeitschema ab:



- t_0 Endekennzeichen = 3 Zeichen
(die Zeit ist von der Baudrate abhängig)
- t_1 Diese Zeit ist von der internen Bearbeitungszeit abhängig.
Die maximale Bearbeitungszeit liegt bei 250 ms.



In dem Gerät kann unter dem Menüpunkt „Schnittstelle“ eine minimale Antwortzeit eingestellt werden. Diese eingestellte Zeit wird mindestens eingehalten, bevor eine Antwort gesendet wird (0...500 ms). Wird ein kleiner Wert eingestellt, so kann die Antwortzeit größer sein als der eingestellte Wert (die interne Bearbeitungszeit ist länger), das Gerät antwortet dann unmittelbar nachdem die interne Bearbeitung abgeschlossen ist. Eine eingestellte Zeit von 0 ms bedeutet, dass das Gerät mit der maximal möglichen Geschwindigkeit antwortet.

Die minimal einstellbare Antwortzeit wird bei der RS485-Schnittstelle vom Master benötigt, um die Schnittstellentreiber von Senden auf Empfangen umzustellen.

- t_2 Diese Zeit braucht der Slave, um von Senden wieder auf Empfangen umzuschalten. Diese Zeit muß der Master einhalten, bevor er eine neue Datenanfrage stellt. Sie muß immer eingehalten werden, auch wenn die neue Datenanfrage an ein anderes Gerät gerichtet ist.

RS485-Schnittstelle: $t_2 = 10\text{ms}$

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.

2.5 Aufbau der Datenblöcke

Alle Datenblöcke haben die gleiche Struktur:

Datenstruktur

Slave-Adresse	Funktionscode	Datenfeld	Checksumme CRC16
1 Byte	1 Byte	x Byte	2 Bytes

Jeder Datenblock enthält vier Felder:

Slave-Adresse	Geräteadresse eines bestimmten Slaves
Funktionscode	Funktionsauswahl (Lesen, Schreiben von Worten)
Datenfeld	Enthält die Informationen: <ul style="list-style-type: none">- Wortadresse- Wortanzahl- Wortwert(e)
Checksumme	Erkennung von Übertragungsfehlern

2 Protokollbeschreibung

2.6 Funktionscodes

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

Funktionsübersicht

Funktionsnummer	Funktion	Begrenzung
0x03 oder 0x04	Lesen von n Worten	max. 32 Worte (64 Byte)
0x06	Schreiben eines Wortes	max. 1 Wort (2 Byte)
0x10	Schreiben von n Worten	max. 32 Worte (64 Byte)



Wenn das Gerät auf diese Funktionen nicht reagiert oder einen Fehlercode ausgibt, siehe Kapitel 2.9 Fehlerbehandlung, Seite 18.

2.6.1 Lesen von n Worten

Mit dieser Funktion werden n ($n \leq 32$) Worte ab einer bestimmten Adresse gelesen.

Datenanfrage

Slave-Adresse	Funktion x03 oder 0x04	Adresse erstes Wort	Wortanzahl (max. 32)	Checksumme CRC16
1 Byte	1 Byte	2 Byte	2 Byte	2 Byte

Antwort

Slave-Adresse	Funktion 0x03 oder 0x04	Anzahl gelesener Bytes	Wort- wert(e)	Checksumme CRC16
1 Byte	1 Byte	1 Byte	x Byte	2 Byte

Beispiel

Lesen der Sollwerte W1 und W2 (jeweils 2 Worte)

Adresse erstes Wort = 0x3100 (Sollwert W1)

Datenanfrage:

01	03	3100	0004	4AF5
----	----	------	------	------

Antwort (Werte im Modbus-Float-Format):

01	03	08	0000	41C8	0000	4120	4A9E
			Sollwert W1 (25.0)		Sollwert W2 (10.0)		

2 Protokollbeschreibung

2.6.2 Schreiben eines Wortes

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

Anweisung

Slave-Adresse	Funktion 0x06	Wortadresse	Wortwert	Checksumme CRC16
1 Byte	1 Byte	2 Byte	2 Bytes	2 Byte

Antwort

Slave-Adresse	Funktion 0x06	Wortadresse	Wortwert	Checksumme CRC16
1 Byte	1 Byte	2 Byte	2 Byte	2 Byte

Beispiel

Schreiben des Grenzwertes AL von Limitkomparator 1 = 275.0
(Wert = 0x80004389 im Modbus-Float-Format)

Wortadresse = 0x0057

Anweisung: Schreiben des ersten Teils des Wertes

01	06	0057	8000	59DA
----	----	------	------	------

Antwort (wie Anweisung):

01	06	0057	8000	59DA
----	----	------	------	------

Anweisung: Schreiben des zweiten Teils des Wertes (nächste Wortadresse)

01	06	0058	4389	F88F
----	----	------	------	------

Antwort (wie Anweisung):

01	06	0058	4389	F88F
----	----	------	------	------

2 Protokollbeschreibung

2.6.3 Schreiben von n Worten

Mit dieser Funktion werden n ($n \leq 32$) Worte ab einer bestimmten Adresse geschrieben.

Anweisung

Slave-Adresse	Funktion 0x10	Adresse erstes Wort	Wortanzahl (max. 32)	Byteanzahl	Wortwert(e)	Checksumme CRC16
1 Byte	1 Byte	2 Byte	2 Byte	1 Byte	x Byte	2 Byte

Antwort

Slave-Adresse	Funktion 0x10	Adresse erstes Wort	Wortanzahl	Checksumme CRC16
1 Byte	1 Byte	2 Byte	2 Byte	2 Byte

Beispiel

Schreiben der Sollwerte W1 und W2 (jeweils 2 Worte)

Wortadresse = 0x3100 (Sollwert W1)

Anweisung:

01	10	3100	0004	08	0000	41C8	0000	4120	2A42
					Sollwert W1 (25.0)		Sollwert W2 (10.0)		

Antwort:

01	10	3100	0004	CF36
----	----	------	------	------

2 Protokollbeschreibung

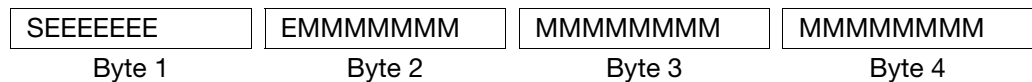
2.7 Übertragungsformat (Integer-, Float- und Text-Werte)

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

Beispiel Abfrage des Integer-Wertes von Adresse 0x0021, wenn unter dieser Adresse der Wert "4" (Wortwert 0x0004) steht.
Anfrage: 01 03 0021 0001 (+ 2 Byte CRC16)
Antwort: 01 03 02 **0004** (+ 2 Byte CRC16)

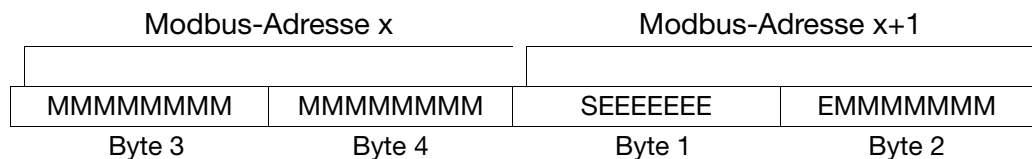
Float-Werte Bei Float-Werten wird im Modbus mit dem IEEE-754-Standard-Format (32bit) gearbeitet, allerdings mit dem Unterschied, dass Byte 1 und 2 mit Byte 3 und 4 vertauscht sind.

Single-float-Format (32bit) nach Standard IEEE 754



S - Vorzeichen-Bit
E - Exponent (2er-Komplement)
M - 23Bit normalisierte Mantisse

Modbus-float-Format

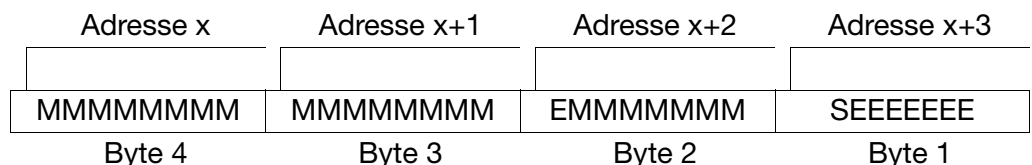



Beispiel Abfrage des Float-Wertes von Adresse 0x0035, wenn unter dieser Adresse der Wert 550.0 (0x44098000 im IEEE-754-Format) steht.
Anfrage: 01 03 0035 0002 (+ 2 Byte CRC16)
Antwort: 01 03 04 **8000 4409** (+ 2 Byte CRC16)

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



 Bitte ermitteln Sie, wie in Ihrer Anwendung Float-Werte gespeichert werden. Ggf. müssen die Bytes nach der Abfrage in Ihrem Schnittstellenprogramm entsprechend getauscht werden.

2 Protokollbeschreibung

Zeichenketten (Texte)

Zeichenketten (Texte) werden im ASCII-Format übertragen.



Als letztes Zeichen kann ein "\0" (ASCII-Code 0x00) als Endekennung übertragen werden. Danach folgende Zeichen haben keine Bedeutung.

In den Adresstabellen ist die max. mögliche Zeichenanzahl im Datentyp angegeben, z. B. "TEXT24" (24 Zeichen). Bei Verwendung einer Endekennung stehen bei diesem Beispiel nur noch 23 lesbare Zeichen für den Text zur Verfügung.

Wird keine Endekennung verwendet, muss beim Schreiben die im Datentyp angegebene max. Zeichenanzahl (z. B. TEXT8 = 8 Zeichen) genutzt werden. So wird verhindert, dass noch im Speicher vorhandene Zeichen an den Text angehängt werden.

Da die Übertragung von Texten wortweise (16 Bit) erfolgt, wird bei einer ungeraden Zeichenanzahl (inkl. "\0") zusätzlich 0x00 angehängt.

Beispiel für Datentyp TEXT4

Lesen des Textes (hier: "AbC ") unter Adresse 0x0067 (max. 4 Zeichen können gespeichert werden)

ASCII-Code für "AbC " (mit einem Leerzeichen am Ende):
0x41, 0x62, 0x43, 0x20

Anfrage: 01 03 0067 0002 (+ 2 Byte CRC16)

Slave-Adresse = 01

Funktion = 03, d. h. Lesen von n Worten

Adresse = 0067

Anzahl der zu lesenden Worte = 0002, da max. 4 Zeichen

Antwort: 01 03 04 **41 62 43 20** (+ 2 Byte CRC16)

Slave-Adresse = 01

Funktion = 03, d. h. Lesen von n Worten

Anzahl der gelesenen Bytes = 04

Variante:

ASCII-Code für "Ab" (ohne Leerzeichen am Ende):

0x41, 0x62, 0x00

ASCII 0x00 ("\0") bedeutet, dass die Zeichenkette hier endet.

Bei der Übertragung wird ein zusätzliches **0x00** angehängt, um eine gerade Anzahl von Zeichen zu erhalten.

Antwort in diesem Fall: 01 03 04 **41 62 00 00** (+ 2 Byte CRC16)

2.8 Checksumme (CRC16)

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

Berechnungs- schema

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);	



Das Low-Byte der Checksumme wird zuerst übertragen, dann folgt das High-Byte.

Beispiel

Datenanfrage: Lesen von zwei Worten ab Adresse 0x00CE
(CRC16 = 0x92A5)

07	03	00	CE	00	02	A5	92
							CRC16

Antwort: (CRC16 = 0xF5AD)

07	03	04	00	00	41	C8	AD	F5
			Wort 1	Wort 2	CRC16			

2 Protokollbeschreibung

2.9 Fehlerbehandlung

Fehlercodes Es existieren folgende Fehlercodes:

- 1 ungültige Funktion
- 2 ungültige Parameteradresse oder zu große Anzahl von Worten soll gelesen oder geschrieben werden
- 8 Schreibzugriff auf Parameter verweigert

Antwort im Fehlerfall

Slave-Adresse	Funktion XX OR 80h	Fehlercode	Checksumme CRC16
1 Byte	1 Byte	1 Byte	2 Bytes

Der Funktionscode wird mit 0x80 verODERT, d. h., das MSB (most significant bit, engl. das höchstwertige Bit) wird auf 1 gesetzt.

Beispiel

Datenanfrage:

01	03	40	00	00	04	CRC16
----	----	----	----	----	----	-------

Antwort (mit Fehlercode 2):

01	83	02	CRC16
----	----	----	-------

Sonderfälle

Wenn der Slave nicht antwortet, können folgende Ursachen vorliegen:

- Baudrate und/oder Datenformat stimmen beim Master und beim Slave nicht überein
- die verwendete Geräteadresse stimmt nicht mit der Slaveadresse überein
- die Checksumme (CRC16) ist nicht korrekt
- die Anweisung des Masters ist unvollständig oder überdefiniert
- die Anzahl der zu lesenden Worte ist Null

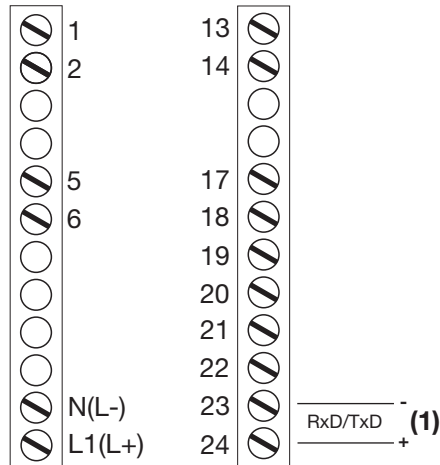
In diesen Fällen sollte die Datenanfrage nach Ablauf der Timeout-Zeit (2s) erneut gesendet werden.

3.1 Anschlussplan

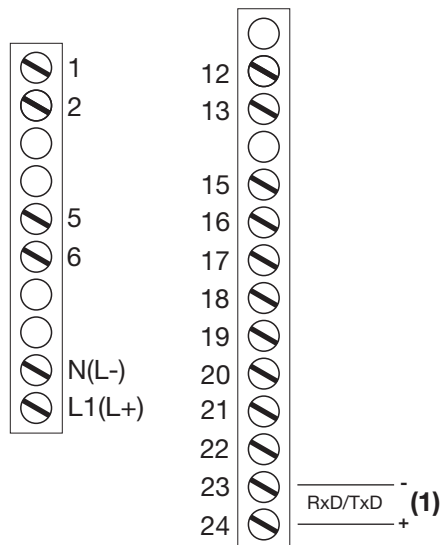


Die Geräte dieser Reglerreihe können optional mit einer RS485-Schnittstelle bestellt werden. Bestellangaben finden Sie im Typenblatt 70.2070 oder in der Betriebsanleitung B 70.2070.0 (Typenerklärung).

Anschlussplan Typ 702071



Anschlussplan Typ 702072 und Typ 702074



(1) RS485-Schnittstelle



Die Schirmung der Schnittstellenleitung ist einseitig im Schalt-schrank zu erden.

3 RS485-Schnittstelle

3.2 Konfiguration

Die folgende Tabelle zeigt die möglichen Einstellungen der Modbus-Schnittstelle, die in der Konfigurationsebene (CONF → INTF) bzw. im Setup-Programm vorgenommen werden.



Weitere Informationen zur Konfiguration können der Betriebsanleitung B 70.2070.0 entnommen werden.

	Symbol	Wert/ Auswahl	Beschreibung
Baudrate	bdr t	0	9600 Baud
		1	19200 Baud
		2	38400 Baud
Datenformat	dft t	0	8 Datenbits, 1 Stoppbit, keine Parität
		1	8 Datenbits, 1 Stoppbit, ungerade Parität
		2	8 Datenbits, 1 Stoppbit, gerade Parität
		3	8 Datenbits, 2 Stoppbits, keine Parität
Geräteadresse	Adr r	0... 1 ...255	Adresse im Datenverbund Die Adressen 0 und 255 sind für bestimmte Zwecke vorgesehen und sollten hier nicht verwendet werden.
Minimale Antwortzeit	(Setup)	0 ...500ms	Zeitspanne, die von der Anfrage eines Gerätes in einem Datenverbund bis zur Antwort des Reglers mindestens vergeht (nur im Setup-Programm einstellbar).

Werkseitige Einstellungen sind **fett** dargestellt.



Bei Kommunikation über die Setup-Schnittstelle ist die RS485-Schnittstelle inaktiv.

4 Modbus-Adressen

Datentyp, Zugriffsart

In den folgenden Tabellen sind alle Prozess- und Gerätedaten mit ihren Adressen, dem Datentyp und der Zugriffsart aufgeführt.

Hierbei bedeutet:

- R/O** Zugriff nur lesend
- W/O** Zugriff nur schreibend
- R/W** Zugriff lesend und schreibend
- INT** Integer (8 oder 16 Bit)
- Bit x** Bit Nr. x (Bit 0 ist das niederwertigste Bit)
- LONG** Long Integer (4 Byte)
- FLOAT** Float-Wert (4 Byte) nach IEEE 754
- TEXT4** Text 4 Zeichen



Schreiboperationen auf R/W-Parameter bewirken ein Abspeichern im EEPROM. Diese Speicherbausteine haben nur eine begrenzte Anzahl von Schreibzyklen (ca. 10000), weshalb bei häufiger Programmierung diese Funktion abgeschaltet werden kann. Die Parameterwerte sind dann nur im flüchtigen Speicher (RAM) gespeichert und nach einem Netzausfall verloren.

⇒ *Setup-Programm (Undokumentierte Parameter -> Bitparameter -> Parameter 2 setzen)*

4.1 Prozessdaten

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x001F	INT	R/O	Interne Binärwerte
	Bit 12		Binärwert L1 (= 0x1000)
	Bit 13		Binärwert L2 (= 0x2000)
0x0020	INT	R/O	Reglerstatus
	Bit 12		Handbetrieb aktiv (= 0x1000)
	Bit 15		Selbstoptimierung aktiv (= 0x8000)
0x0021	INT	R/O	Binärausgänge 1...4 (Schaltzustände 0 = aus / 1 = ein)
	Bit 0		Ausgang K1: Relais (= 0x0001)
	Bit 1		Ausgang K2: Relais (= 0x0002)
	Bit 2		Ausgang K3: Logik (= 0x0004)
	Bit 3		Ausgang K4: Relais (= 0x0008)

4 Modbus-Adressen

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0023	INT	R/O	Binäreingänge 1...2 (Schaltzustände 0 = offen / 1 = geschlossen)
	Bit 0		Eingang 1 (= 0x0001)
	Bit 1		Eingang 2 (= 0x0002)
0x0024	INT	R/O	Limitkomparatoren 1...2
	Bit 0		Limitkomparator 1 (= 0x0001)
	Bit 1		Limitkomparator 2 (= 0x0002)
0x0025	INT	R/W	Ansteuerung Binärausgänge (einzeln)
	Bit 0 + Bit 8		Ausgang K1 (= 0x0101)
	Bit 1 + Bit 9		Ausgang K2 (= 0x0202)
	Bit 2 + Bit 10		Ausgang K3 (= 0x0404)
	Bit 3 + Bit 11		Ausgang K4 (= 0x0808)
0x0026	FLOAT	R/O	Analogeingang [mV]
0x0028	FLOAT	R/O	Interner Pt100 [Ohm]
0x002A	INT	R/O	Abtastzeit
0x002B	FLOAT	R/O	Analogeingang [Anzeigewert]
0x002D	FLOAT	R/O	Interner Analogwert 1
0x002F	FLOAT	R/O	Interner Analogwert 2
0x0031	FLOAT	R/O	Regler Rampenendwert
0x0033	FLOAT	R/O	Regler Istwert GEFILTERT
0x0035	FLOAT	R/O	Regler Istwert UNGEFILTERT
0x0037	FLOAT	R/W	Regler Sollwert
0x0039	FLOAT	R/O	Regler Stellgradanzeige
0x003B	FLOAT	R/O	Regler Stellgrad HEIZEN
0x003D	FLOAT	R/O	Regler Stellgrad KÜHLEN
0x003F	FLOAT	R/O	Regler Regeldifferenz
0x0041	FLOAT	R/O	Regler Regelabweichung
0x0043	INT	R/O	Regler Schaltstellung HEIZEN
0x0044	INT	R/O	Regler Schaltstellung KÜHLEN
0x0046	INT	R/O	Handstellgrad
0x0047	LONG	R/O	Timer-Laufzeit
0x0049	LONG	R/O	Timer-Restzeit
0x004B	INT	R/O	Timer-Status
	Bit 1		Timer angehalten (= 0x0002)
	Bit 5		Timer läuft (= 0x0020)
	Bit 6		Timer-Ende (= 0x0040)
	Bit 15		Timer-Signal (= 0x8000)

4.2 Sollwerte

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x3100	FLOAT	R/W	Sollwert W1
0x3102	FLOAT	R/W	Sollwert W2

4.3 Reglerparameter

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x3000	FLOAT	R/W	Regler-Parameter XP1
0x3002	FLOAT	R/W	Regler-Parameter XP2
0x3004	FLOAT	R/W	Regler-Parameter TV
0x3006	FLOAT	R/W	Regler-Parameter TN
0x300C	FLOAT	R/W	Regler-Parameter CY1
0x300E	FLOAT	R/W	Regler-Parameter CY2
0x3010	FLOAT	R/W	Regler-Parameter XSH
0x3012	FLOAT	R/W	Regler-Parameter XD1
0x3014	FLOAT	R/W	Regler-Parameter XD2
0x3016	INT	R/W	Regler-Parameter TT
0x3017	INT	R/W	Regler-Parameter Y0
0x3018	INT	R/W	Regler-Parameter Y1
0x3019	INT	R/W	Regler-Parameter Y2

4.4 Konfiguration

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0053	FLOAT	R/W	Rampenfunktion Steigung
0x0055	FLOAT	R/W	Filterzeitkonstante
0x0057	FLOAT	R/W	Limitkomparator 1 Grenzwert AL
0x0059	FLOAT	R/W	Limitkomparator 1 Schaltdifferenz
0x005D	FLOAT	R/W	Limitkomparator 2 Grenzwert AL
0x005F	FLOAT	R/W	Limitkomparator 2 Schaltdifferenz
0x0063	LONG	R/W	Timer-Wert
0x0065	LONG	R/W	Service-Grenzwert
0x0067	TEXT4	R/W	Alarmtext
0x0069	LONG	R/W	Servicezähler

4 Modbus-Adressen

4.5 Kommandos

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x004D	INT	W/O	Binärfunktionen REGLER
	Bit 0		Selbstoptimierung Start (= 0x0001)
	Bit 1		Selbstoptimierung Abbruch (= 0x0002)
	Bit 2		Handbetrieb (= 0x0004)
	Bit 3		Automatikbetrieb (= 0x0008)
	Bit 4		Regler aus (= 0x0010)
	Bit 5		Verriegelung Handbetrieb (= 0x0020)
	Bit 6		Rampe Halt (= 0x0040)
	Bit 7		Rampe Abbruch (= 0x0080)
	Bit 8		Rampe Neustart (= 0x0100)
	Bit 9		Timer Start (= 0x0200)
	Bit 10		Timer Abbruch (= 0x0400)
	Bit 11		Timer Halt (= 0x0800)
0x004E	INT	W/O	Binärfunktionen BEDIENUNG
	Bit 0		Verriegelung Tastatur (= 0x0001)
	Bit 1		Verriegelung Konfigurations- und Parameterebene (= 0x0002)
	Bit 3		Anzeige AUS (= 0x0008)
	Bit 5		Textanzeige (= 0x0020)
0x004F	INT	W/O	Binärfunktionen TIMER
	Bit 9		Timer starten (= 0x0200)
	Bit 10		Timer abrechnen (= 0x0400)
	Bit 11		Timer anhalten (= 0x0800)
0x0050	INT	R/W	Sollwertumschaltung
	Bit 0		Sollwert 1 (= 0x0001)
	Bit 1		Sollwert 2 (= 0x0002)

4.6 RAM-Speicher

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x3200	FLOAT	W/O	Regler-Sollwert (beschreibbar)
0x3202	FLOAT	W/O	Regler-Istwert (beschreibbar)
0x3204	FLOAT	W/O	Interner Analogwert 1 (beschreibbar)
0x3206	FLOAT	W/O	Interner Analogwert 2 (beschreibbar)
0x3208	INT	W/O	Interne Binärwerte (beschreibbar)
	Bit 0 + Bit 7		Binärwert L1 (= 0x0081)
	Bit 1 + Bit 7		Binärwert L2 (= 0x0082)



Über Modbus ist der direkte Zugriff auf den RAM-Speicher des Gerätes möglich, um den Regler-Sollwert (0x3200), den Regler-Istwert (0x3202) und die Internen Analogwerte (0x3204, 0x3206) sowie die Internen Binärwerte (0x3208) zu schreiben.

Beim Schreiben steht für die Werte von Regler-Sollwert, Regler-Istwert und für die Internen Analogwerte ein Bereich von -1999 bis +9999 zur Verfügung. Statt des Originalwertes wird dann der ins Gerät geschriebene Wert verwendet.

Soll am Gerät wieder der Originalwert verwendet werden, muss der Wert 200001 über Modbus in die jeweilige Speicherstelle geschrieben werden.

4 Modbus-Adressen



JUMO GmbH & Co. KG

Moritz-Juchheim-Straße 1
36039 Fulda, Germany
Telefon: +49 661 6003-727
Telefax: +49 661 6003-508
E-Mail: mail@jumo.net
Internet: www.jumo.net

Lieferadresse:
Mackenrodtstraße 14
36039 Fulda, Germany

Postadresse:
36035 Fulda, Germany

Kundendienst

Deutschland:
Telefon: +49 661 6003-300
oder -653 oder -899
Telefax: +49 661 6003-881729
E-Mail: service@jumo.net

JUMO Mess- und Regelgeräte Ges.m.b.H.

Pfarrgasse 48
1232 Wien, Austria
Telefon: +43 1 610610
Telefax: +43 1 6106140
E-Mail: info@jumo.at
Internet: www.jumo.at

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

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

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