

**JUMO dTRON 04.1**

**JUMO dTRON 08.1**

**JUMO dTRON 16.1**

Kompakter Mikroprozessorregler  
Compact microprocessor  
controller

**B 70.3030.2**

Schnittstellenbeschreibung  
Interface description

11.97/00325533



# Inhalt

---

## 1 Einleitung

1.1	Vorwort .....	4
1.2	Typografische Konventionen .....	5
1.2.1	Warnende Zeichen .....	5
1.2.2	Hinweisende Zeichen .....	5
1.2.3	Darstellungsarten .....	5

## 2 Protokollbeschreibung



2.1	Master-Slave-Prinzip .....	6
2.2	Übertragungsmodus (RTU) .....	6
2.3	Geräteadresse .....	7
2.4	Zeitlicher Ablauf der Kommunikation .....	7
2.4.1	Zeitlicher Ablauf einer Datenanfrage .....	9
2.4.2	Kommunikation während der internen Bearbeitungszeit des Slaves .....	10
2.4.3	Kommunikation während der Antwortzeit des Slaves .....	10
2.5	Aufbau der Datenblöcke .....	10
2.6	Fehlerbehandlung .....	10
2.7	Unterscheidung MOD-Bus/J-Bus .....	11
2.8	Checksumme (CRC16) .....	12

## 3 Funktionen

3.1	Lesen von n Worten .....	14
3.2	Schreiben eines Worts .....	15
3.3	Schreiben von n Worten .....	16

## 4 Datenfluß

## 5 Adresstabellen

5.1	Adresstabellen für  dTRON 04.1/08.1 .....	20
5.2	Adresstabellen für  dTRON 16.1 .....	24

---

# 1 Einleitung

---

## 1.1 Vorwort

Lesen Sie diese Betriebsanleitung, bevor Sie die Schnittstelle in Betrieb nehmen. Bewahren Sie die Betriebsanleitung an einem für alle Benutzer jederzeit zugänglichen Platz auf.

Bitte unterstützen Sie uns, diese Betriebsanleitung zu verbessern.

Für Ihre Anregungen sind wir dankbar.

Telefon (06 61) 60 03-7 27

Telefax (06 61) 60 03-5 08



Alle erforderlichen Informationen zum Betrieb der Schnittstelle sind in der vorliegenden Betriebsanleitung beschrieben. Sollten bei der Inbetriebnahme trotzdem Schwierigkeiten auftreten, bitten wir Sie, keine unzulässigen Manipulationen vorzunehmen. Sie können Ihren Garantieanspruch gefährden!

Bitte Setzen Sie sich mit der nächsten Niederlassung oder mit dem Stammhaus in Verbindung.



Bei Rücksendungen von Geräteeinschüben, Baugruppen oder Bauelementen sind die Regelungen nach DIN EN 100 015 „Schutz von elektrostatisch gefährdeten Bauelementen“ einzuhalten. Verwenden Sie nur dafür vorgesehene ESD-Verpackungen für den Transport.

Bitte beachten Sie, daß für Schäden, die durch ESD verursacht werden, keine Haftung übernommen werden kann.

ESD= Elektrostatische Entladungen

# 1 Einleitung

---

## 1.2 Typografische Konventionen

### 1.2.1 Warnende Zeichen

Die Zeichen für *Vorsicht* und *Achtung* werden in dieser Betriebsanleitung unter folgenden Bedingungen verwendet:



**Vorsicht** Dieses Zeichen wird benutzt, wenn es durch ungenaues Befolgen oder Nichtbefolgen von Anweisungen zu Personenschäden kommen kann!



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



**Achtung** Diese Zeichen wird benutzt, wenn Vorsichtsmaßnahmen bei der Handhabung elektrostatisch entladungsgefährdeter Bauelemente zu beachten sind.

### 1.2.2 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.

abc<sup>1</sup>

**Fußnote** Fußnoten sind Anmerkungen, die auf bestimmte Textstellen Bezug nehmen. Fußnoten bestehen aus zwei Teilen:

Kennzeichnung im Text und Fußnotentext.

Die Kennzeichnung im Text geschieht durch hochstehende fortlaufende Zahlen.

Der Fußnotentext (2 Schriftgrade kleiner als die Grundschrift) steht am unteren Seitenende und beginnt mit einer Zahl und einem Punkt.

### 1.2.3 Darstellungsarten

0x0010

**Hexadezimalzahl**

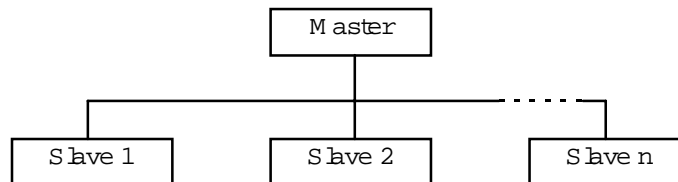
Eine Hexadezimalzahl wird durch ein vorgestelltes „0x“ gekennzeichnet (hier: 16 dezimal).

## 2 Protokollbeschreibung

---

### 2.1 Master-Slave-Prinzip

Die Kommunikation zwischen einem PC (Master) und einem Gerät (Slave) mit MOD-/J-Bus 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. Es können maximal 255 Slaves angesprochen werden.

### 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, 16 Bits bei Integerwerten und 32 Bits bei Floatwerten. 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 Bytes 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

## 2 Protokollbeschreibung

---

### 2.3 Geräteadresse

Die Geräteadresse des Slaves ist zwischen 1 und 31 einstellbar. Die Geräteadresse 0 ist reserviert.



Über die RS422-/RS485-Schnittstelle können maximal 31 Slaves angesprochen werden.

Man unterscheidet zwei Möglichkeiten 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. Die angeschlossenen Slaves antworten nicht. Eine Datenanfrage mit der Geräteadresse 0 ist nicht sinnvoll. So kann z. B. allen Slaves ein bestimmter Sollwert übertragen werden. Die richtige Übernahme des Wertes durch die Slaves sollte in diesem Fall durch anschließendes Auslesen des Sollwertes kontrolliert werden.

### 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.

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

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

Bei den anderen Datenformaten ergibt sich:

$$\text{Zeichenübertragungszeit [ms]} = 1000 * 10 \text{ Bits/Baudrate}$$

#### Ablauf

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

## 2 Protokollbeschreibung

---

### Beispiel

Kennzeichen für Datenanfrage- oder Antwort-Ende bei Datenformat 10/9 Bits

Wartezeit = 3 Zeichen \* 1000 \* 10 Bits/Baudrate[

Baudrate [Baud]	Datenformat [Bit]	Wartezeit [ms]
9600	10	3,125
	9	2,813
4800	10	6,250
	9	5,625
2400*	10	12,500
	9	11,250
1200*	10	25,000
	9	22,500

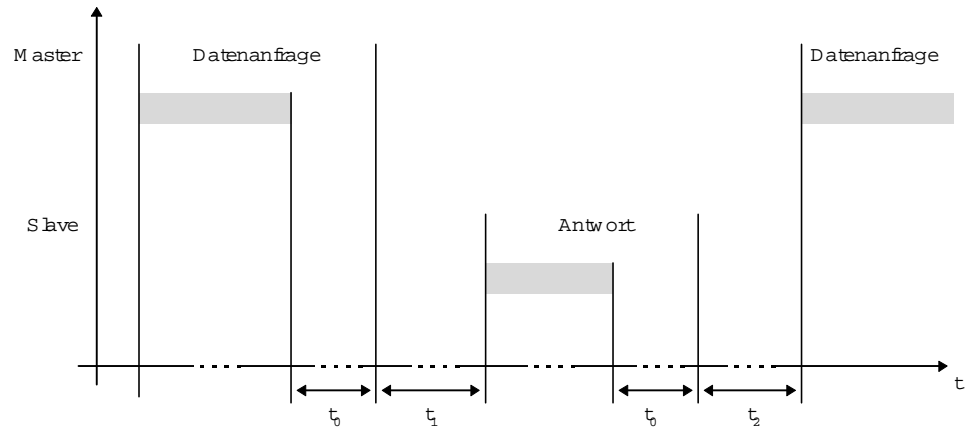
\* Nur **JUMO** dTRON 04.1/08.1.

## 2 Protokollbeschreibung

---

### 2.4.1 Zeitlicher Ablauf einer Datenanfrage

**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 Bearbeitung abhängig.  
Die maximale Bearbeitungszeit liegt bei 250 ms.
  - $t_2$  Diese Zeit braucht der Gerät, 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.
- $t_2 \geq 20\text{ms}$

## 2 Protokollbeschreibung

---

### 2.4.2 Kommunikation während der internen Bearbeitungszeit des Slaves

Während der internen Bearbeitungszeit des Slaves dürfen vom Master keine Datenanfragen gestellt werden. In dieser Zeit gestellte Datenanfragen werden vom Slave ignoriert.

### 2.4.3 Kommunikation während der Antwortzeit des Slaves

Während der Antwortzeit des Slaves dürfen vom Master keine Datenanfragen gestellt werden. In dieser Zeit gestellte Datenanfragen 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(s)	2 Bytes

Jeder Datenblock enthält vier Felder:

**Slave-Adresse** Geräteadresse eines bestimmten Slaves

**Funktionscode** Funktionsauswahl (Lesen, Schreiben von Worten)

**Datenfeld** Enthält die Informationen:  
- Wortadresse  
- Wortanzahl  
- Wortwert

**Checksumme** Erkennung von Übertragungsfehlern

## 2.6 Fehlerbehandlung

### Fehlercodes

Es existieren fünf Fehlercodes:

- 1 ungültige Funktion
- 2 ungültige Parameteradresse
- 3 Parameterwert außerhalb Wertebereich<sup>1</sup>
- 4 Slave nicht bereit
- 8 Schreibzugriff auf Parameter verweigert

1. Die Parameter werden nicht auf Plausibilität geprüft.

## 2 Protokollbeschreibung

---

### 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 geODERT, d. h., das MSB (most significant bit, engl. das höchstwertige Bit) wird auf 1 gesetzt.

### Beispiel

Datenanfrage:

01	02	00	00	00	04	CRC16
----	----	----	----	----	----	-------

Antwort:

01	82	01	CRC16
----	----	----	-------

### Sonderfälle

In folgenden Fehlerfällen antwortet der Slave nicht:

- die Checksumme (CRC16) ist nicht korrekt
- die Anweisung des Masters ist unvollständig oder überdefiniert
- die Anzahl der zu lesenden Worte oder Bits ist Null

## 2.7 Unterscheidung MOD-Bus/J-Bus

Das MOD-Bus-Protokoll ist zu dem J-Bus-Protokoll kompatibel. Die Struktur der Datenblöcke ist identisch.



Der Unterschied zwischen MOD-Bus und J-Bus besteht darin, daß die absoluten Adressen der Daten verschieden sind. Die Adressen des MOD-Bus sind gegenüber denen des J-Bus um eins verschoben.

Das J-Bus-Protokoll wird nur von **JUMO** dTRON 04.1/08.1 unterstützt.

Absolute Adresse	Adresse J-Bus	Adresse MOD-Bus
1	1	0
2	2	1
3	3	2
...	...	...

## 2 Protokollbeschreibung

### 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.

#### Beispiel

Datenanfrage: Lesen von zwei Worten ab Adresse 6 (CRC16 = 0x24A0)

0B	03	00	06	00	02	24	A0
						CRC16	

Antwort: (CRC16 = 0x6105)

0B	03	04	00	00	42	C8	61	05
			Wort 1		Wort 2		CRC16	

## 3 Funktionen

---

Die folgenden Funktionen stehen für das Gerät zur Verfügung:

Funktionsnummer	Funktion
0x03/0x04	Lesen von n Worten
0x06	Schreiben eines Wortes
0x10	Schreiben von n Worten



Alle Beispiele beziehen sich auf Adressen des **JUMO** dTRON 04.1/08.1.

# 3 Funktionen

---

## 3.1 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 (max. 6)	Checksumme CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

### Antwort

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

### Beispiel

Lesen der 2 Sollwerte des Reglers

Wortadresse = 0x0008 (1. Sollwert SP1)

Datenanfrage:

0B	03	00	08	00	04	CRC16
----	----	----	----	----	----	-------

Antwort:

0B	03	08	0000	42C8	0000	4316	CRC16
			Sollwert 1 (100)		Sollwert 2 (150)		

# 3 Funktionen

---

## 3.2 Schreiben eines Worts

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

### Anweisung

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

### Antwort

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

### Beispiel

Schreibe Grenzwert Limitkomparator 1 (AL1) (= 275)

Wortadresse = 0x000C

Anweisung: Schreiben des ersten Teils des Wertes

0B	06	00	0C	80	00	CRC16
----	----	----	----	----	----	-------

Antwort (wie Anweisung):

0B	06	00	0C	80	00	CRC16
----	----	----	----	----	----	-------

Anweisung: Schreiben des zweiten Teils des Wertes

0B	06	00	0D	43	89	CRC16
----	----	----	----	----	----	-------

Antwort (wie Anweisung):

0B	06	00	0D	43	89	CRC16
----	----	----	----	----	----	-------

# 3 Funktionen

---

## 3.3 Schreiben von n Worten

### Anweisung

Slave-Adresse	Funktion 0x10	Adresse erstes Wort	Wortan- zahl	Byte- anzahl	Wort- wert(e)	Checksumme CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	1 Byte	x Byte(s)	2 Bytes

### Antwort

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

### Beispiel

Schreibe Gradient der Rampenfunktion ( $rASd = 100$ ) des zweiten Parametersatzes

Wortadresse = 0x0072

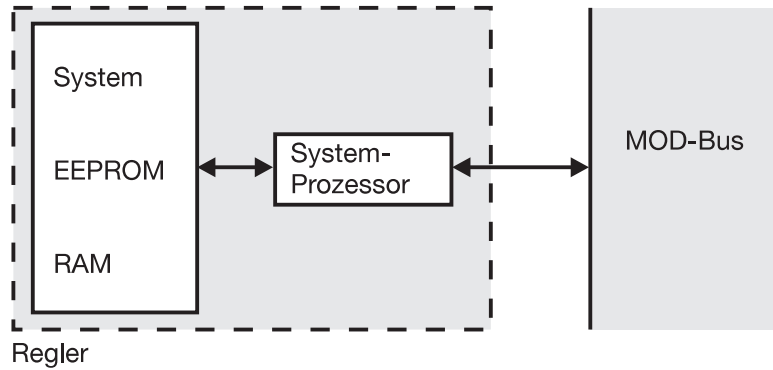
Anweisung:

0B	10	00	72	00	02	04	00	00	42	C8	CRC16
----	----	----	----	----	----	----	----	----	----	----	-------

Antwort:

0B	10	00	72	00	02	CRC16
----	----	----	----	----	----	-------

# 4 Datenfluß



Bei aktiver Selbstoptimierung ist die Schnittstelle abgeschaltet. Der Regler meldet keinen Fehler.

## Datentyp „char“

Übertragen werden grundsätzlich nur ASCII-Zeichen. Diese werden entsprechend der Reihenfolge im Speicher gesendet.

Beispiel:

Text: „075.02.01“

MOD-Bus:

0x30, 0x37, 0x35, 0x2E, 0x30, 0x31, 0x2E, 0x30, 0x31, 0x20, 0x20, 0x00

## Datentyp „integer“

Es muß zuerst das High-Byte übertragen werden.

Beispiel:

Konfigurationscode C211: „7702“

MOD-Bus: 0x07, 0x07, 0x00, 0x02

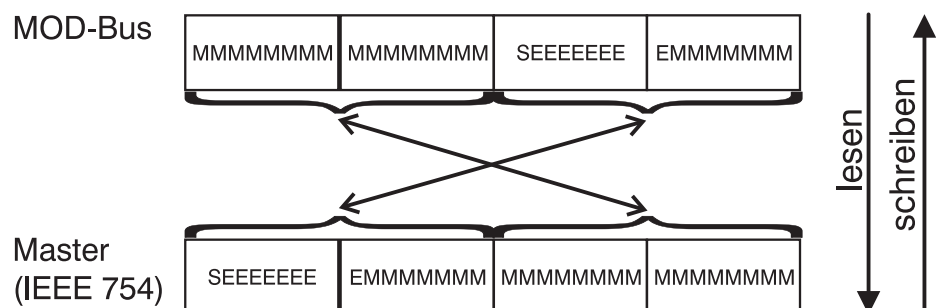
## Datentyp „float“

Die folgenden Erklärungen gelten unter der Bedingung, daß der Master mit dem IEEE-754-Format arbeitet. Vor der Übertragung eines Wertes müssen die Bytes so vertauscht werden, daß die Reihenfolge der Darstellung für den MOD-Bus entspricht (siehe Grafik).

M - 23 Bit normalisierte Mantisse

E - Exponent (2er Komplement)

S - Vorzeichen-Bit; 1 = negativ, 0 = positiv



Beispiel:

Übertragung des Dezimalwertes „550“:

MOD-Bus: 0x80, 0x00, 0x44, 0x09

## 4 Datenfluß

---

**Datentyp  
„byte“  
(Status-  
kennzeichen)**

Kennzeichen können nur in Gruppen zu 8 Bit übertragen werden. Diesen 8 Bit wird ein Null-Byte (0x00) vorangestellt.

Beispiel:

Die Status-Kennzeichen der Speicherstelle 0x81 sollen ausgelesen werden. Sollwert 2 und Parametersatz 2 sind aktiv (B01001000 = 0x48).

MOD-Bus: 0x00, 0x48



Jede Veränderung eines Prozeßwertes, der im EEPROM abgelegt ist, hat zur Folge, daß die Daten im EEPROM aktualisiert werden. Bitte beachten Sie, daß das EEPROM bis zu ca. 10000 mal neu beschrieben werden kann.

## 5 Adresstabellen

---

Im folgenden sind alle Prozeßwerte (Variablen) mit ihren Adressen, dem Datentyp und der Zugriffsart beschrieben.

Hierbei bedeutet:

R/O            Zugriff nur lesend

R/W            Zugriff schreibend und lesend

byte            Byte (8 Bit)

char xx        Zeichenkette mit Länge xx;  
xx = Länge **inklusive** Zeichenkettenende-Zeichen /0

float           Float-Wert (4 Byte/2 Worte)

int             Integer-Wert (2 Byte/1 Wort)

Die Prozeßwerte sind in logische Bereiche unterteilt.

## 5 Adresstabellen

### 5.1 Adresstabellen für **JUMO** dTRON 04.1/08.1

#### Gerätedaten

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0100	char 12	R/O	Software-Version 075.xx.xx
0x0106	char 14	R/O	VdN-Nummer

Gerätedaten können nur gelesen werden.

#### Prozeßdaten

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0000	float	R/O	Istwert Analogeingang 1
0x0002	float	R/O	Istwert Analogeingang 2
0x0006	float	R/O  R/W	- Reglerausgang im Automatikbetrieb  - Stellgradrückmeldung bei Dreipunktschrittregler  - Stellgrad im Handbetrieb
0x0004	float	R/W  R/O	- aktueller Sollwert  - Rampensollwert (bei Ram- penfunktion)  - externer Sollwert (bei exter- ner Sollwertvorgabe)  Dieser Wert wird nicht ins EEPROM übernommen.
0x0008	float	R/W	Sollwert 1
0x000A	float	R/W	Sollwert 2

#### Parameter- satz 1

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x000C	float	R/W	AL1 - Grenzwert Limitkompa- rator 1
0x000E	float	R/W	AL2 - Grenzwert Limitkompa- rator 2
0x0010	float	R/W	Pb1 - Proportionalbereich 1
0x0012	float	R/W	Pb2 - Proportionalbereich 2
0x0014	float	R/W	dt - Vorhaltzeit
0x0016	float	R/W	rt - Nachstellzeit
0x0018	float	R/W	tt - Stellgliedlaufzeit
0x001A	float	R/W	Cy1 - Schaltperiodendauer 1
0x001C	float	R/W	Cy2 - Schaltperiodendauer 2
0x001E	float	R/W	db - Kontaktabstand

## 5 Adresstabellen

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0020	float	R/W	HYS.1 - Schaltdifferenz 1
0x0022	float	R/W	HYS.2 - Schaltdifferenz 2
0x0024	float	R/W	Y0 - Arbeitspunkt
0x0026	float	R/W	Y1 - max. Stellgrad
0x0028	float	R/W	Y2 - min. Stellgrad
0x002A	float	R/W	dF - Filterzeitkonstante
0x002C	float	R/W	rASd - Rampensteigung
0x002E	float	R/W	tS - Zeitdauer der Haltphase (Nur bei Reglern mit Anfahr- rampe!)

### Parameter- satz 2

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0056	float	R/W	Pb1 - Proportionalbereich 1
0x0058	float	R/W	Pb2 - Proportionalbereich 2
0x005A	float	R/W	dt - Vorhaltzeit
0x005C	float	R/W	rt - Nachstellzeit
0x005E	float	R/W	tt - Stellgliedlaufzeit
0x0060	float	R/W	Cy1 - Schaltperiodendauer 1
0x0062	float	R/W	Cy2 - Schaltperiodendauer 2
0x0064	float	R/W	db - Kontaktabstand
0x0066	float	R/W	HYS.1 - Schaltdifferenz 1
0x0068	float	R/W	HYS.2 - Schaltdifferenz 2
0x006A	float	R/W	Y0 - Arbeitspunkt
0x006C	float	R/W	Y1 - max. Stellgrad
0x006E	float	R/W	Y2 - min. Stellgrad
0x0070	float	R/W	dF - Filterzeitkonstante
0x0072	float	R/W	rASd - Rampensteigung

### Konfigurati- onsebene

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x030	int 02	R/W	C111 (Analogeingang 1 + 2)
0x0032	int 02	R/W	C112 (allgemeine Gerätedaten)
0x0034	int 02	R/W	C113 (Schnittstelle)
0x0036	int 02	R/W	C211 (Limitkomparatoren)
0x0038	int 02	R/W	C212 (Reglereinstellungen und Ausgang 3)
0x003A	int 02	R/W	C213 (Ausgang 1, 2, 4 und 5)

## 5 Adresstabellen

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x003C	float	R/W	SCL (Einheitssignalskalierung: Meßbereichsanfang)
0x003E	float	R/W	SCH (Einheitssignalskalie- rung: meßbereichsende)
0x0040	float	R/W	SPL (untere Sollwertgrenze)
0x0042	float	R/W	SPH (obere Sollwertgrenze)
0x0044	float	R/W	OFFS (Istwertkorrektur)

### Status- Kennzeichen

Status-Kennzeichen werden in Gruppen zu 8 Bit im Datenformat „byte“ ausgelesen. Das Verändern der Kennzeichen über die Schnittstelle ist nicht möglich.

Kennzeichen	Adresse MOD-Bus	Position	Bedeutung
Ausgang 1	0x80	_____ 1	Ausgang aus (Kontakt offen)
Ausgang 2		_____ 1 _	Ausgang aus (Kontakt offen)
Ausgang 3		_____ 1 __	Ausgang aus (Kontakt offen)
Ausgang 4		_____ 1 ___	Ausgang aus (Kontakt offen)
Ausgang 5		_____ 1 ____	Ausgang aus (Kontakt offen)
Handbetrieb		_____ 1 _____	Handbetrieb aktiv
Tastatur- verriegelung	0x81	_____ 1	Tastatur verriegelt
Ebenen- verriegelung		_____ 1 _	Parameter- und Konfigurationsebe- ne verriegelt
Rampenstopp		_____ 1 __	Rampenfunktion oder Anfahrrampe angehalten
Sollwert- umschaltung		_____ 1 ___	Sollwert SP2 aktiv
Schnittstellen- sollwert		_____ 1 ____	Schnittstellensoll- wert aktiv
Haltphase (Anfahrrampe)		_____ 1 _____	Haltphase erreicht
Parametersatz- umschaltung		_____ 1 _____	Parametersatz 2 aktiv

## 5 Adresstabellen

Kennzeichen	Adresse MOD-Bus	Position	Bedeutung
Meßbereichsüber- oder -unterschrei- tung Eingang 1	0x82	----- 1	Meßbereichsüber- oder -unterschrei- tung Eingang 1
Meßbereichsüber- oder -unterschrei- tung Eingang 2		----- 1 _	Meßbereichsüber- oder -unterschrei- tung Eingang 2
Binäreingang 1		----- 1 --	Kontakt geschlossen
Binäreingang 2		----- 1 ---	Kontakt geschlossen
1. Reglerausgang		--- 1 -----	Reglerausgang aktiv
2. Reglerausgang		-- 1 -----	Reglerausgang aktiv
Limitkomparator 1		_ 1 -----	Schaltzustand „EIN“
Limitkomparator 2		1 -----	Schaltzustand „EIN“

## 5 Adresstabellen

### 5.2 Adresstabellen für **JUMO** dTRON 16.1

#### Gerätedaten

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0301	char[10]	R/O	Gerätename
0x0306	char[12]	R/O	Software-Version
0x030C	char[14]	R/O	VdN-Nummer

#### Prozeßdaten

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0000	float	R/O	Istwert Analogeingang 1
0x0002	float	R/W	aktueller Sollwert
0x0004	float	R/W	SP1
0x0006	float	R/W	SP2

#### Parameter- satz 1

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0008	float	R/W	AL1 - Grenzwert Limitkomparator 1
0x000A	float	R/W	AL2 - Grenzwert Limitkomparator 2
0x000C	float	R/W	Pb1 - Proportionalbereich 1
0x000E	float	R/W	Pb2 - Proportionalbereich 2
0x0010	float	R/W	dt - Vorhaltzeit
0x0012	float	R/W	rt - Nachstellzeit
0x0014	float	R/W	Cy1 - Schaltperiodendauer 1
0x0016	float	R/W	Cy2 - Schaltperiodendauer 2
0x0018	float	R/W	db - Kontaktabstand
0x001A	float	R/W	HYS.1 - Schaltdifferenz 1
0x001C	float	R/W	HYS.2 - Schaltdifferenz 2
0x001E	float	R/W	Y0 - Arbeitspunkt
0x0020	float	R/W	Y1 - max. Stellgrad
0x0022	float	R/W	Y2 - min. Stellgrad
0x0024	float	R/W	dF - Filterzeitkonstante
0x0026	float	R/W	rASd - Rampensteigung

#### Parameter- satz 2

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x003E	float	R/W	Pb1 - Proportionalbereich 1
0x0040	float	R/W	Pb2 - Proportionalbereich 2
0x0042	float	R/W	dt - Vorhaltzeit
0x0044	float	R/W	rt - Nachstellzeit

## 5 Adresstabellen

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0046	float	R/W	Cy1 - Schaltperiodendauer 1
0x0048	float	R/W	Cy2 - Schaltperiodendauer 2
0x004A	float	R/W	db - Kontaktabstand
0x004C	float	R/W	HYS.1 - Schaltdifferenz 1
0x004E	float	R/W	HYS.2 - Schaltdifferenz 2
0x0050	float	R/W	Y0 - Arbeitspunkt
0x0052	float	R/W	Y1 - max. Stellgrad
0x0054	float	R/W	Y2 - min. Stellgrad
0x0056	float	R/W	dF - Filterzeitkonstante
0x0058	float	R/W	rASd - Rampensteigung

### Konfigurations- onsebene

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0028	char[4]	R/W	C111
0x002A	char[4]	R/W	C112
0x002C	char[4]	R/W	C113
0x002E	char[4]	R/W	C114
0x0030	char[4]	R/W	C000 (nur über Schnittstelle einstellbar)
0x0032	float	R/W	SCL
0x0034	float	R/W	SCH
0x0036	float	R/W	SPL
0x0038	float	R/W	SPH
0x003A	float	R/W	OFFS
0x003C	float	R/W	HYST

### Programm- daten

Adresse MOD-Bus	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x005A	float	R/W	SP00
0x005C	float	R/W	t00
0x005E	float	R/W	SP01
0x0060	float	R/W	t01
0x0062	float	R/W	SP02
0x0064	float	R/W	t02
0x0066	float	R/W	SP03
0x0068	float	R/W	t03

## 5 Adresstabellen

Status-  
Kennzeichen

Adresse MOD-Bus	Datentyp	Signalbezeichnung
0x0200	word	Ausgänge und Binärfunktionen
	_____ 1	Ausgang 1 aus
	_____ 1 _	Ausgang 2 aus
	_____ 1 _ _	Ausgang 3 aus
	_____ 1 _ _ _	Ausgang 4 aus
	_____ 1 _ _ _ _	Ausgang 5 aus
	_____ 1 _ _ _ _ _	Tastatur verriegelt
	_____ 1 _ _ _ _ _ _	Parameter- und Konfigurationsebene verriegelt
	_____ 1 _ _ _ _ _ _ _	Programm oder Rampenfunktion angehalten
	_____ 1 _ _ _ _ _ _ _ _	Zweiter Sollwert aktiv
	_____ 1 _ _ _ _ _ _ _ _ _	Schnittstellensollwert aktiv
	_____ 1 _ _ _ _ _ _ _ _ _ _	Kalibriermode
	_____ 1 _ _ _ _ _ _ _ _ _ _ _	Zweiter Parametersatz aktiv
	1 _ _ _ _ _ _ _ _ _ _ _ _ _	Limitkomparatoren freigegeben

## 5 Adresstabellen

Adresse MOD-Bus	Datentyp	Signalbezeichnung
0x0201	word	Binärsignale und Hardwarekennung
	----- 1	Meßbereichsüberschreitung Eingang 1
	----- 1 _	Meßbereichsüberschreitung Eingang 2
	----- 1 _ _	Binäreingang 1 geschlossen
	----- 1 _ _ _	Binäreingang 2 geschlossen
	----- 1 _ _ _ _	1. Reglerausgang aktiv
	----- 1 _ _ _ _ _	2. Reglerausgang aktiv
	----- 1 _ _ _ _ _ _	1. Limitkomparator aktiv
	----- 1 _ _ _ _ _ _ _	2. Limitkomparator aktiv
	--- 1 0 _ _ _ _ _ _ _	Relaisausgang 3 vorhanden
	--- 0 1 _ _ _ _ _ _ _	Binärausgang 3 vorhanden
	--- 1 1 _ _ _ _ _ _ _	Transistorausgang 3 vorhanden
	-- 1 _ _ _ _ _ _ _ _	Analogausgang 3 vorhanden
	_ 1 _ _ _ _ _ _ _ _ _	Analogausgang Spannung
	----- 1 _ _ _ _ _ _ _	Analogeingang 2 vorhanden
	----- 1 _ _ _ _ _ _ _ _	Schnittstelle vorhanden
	1 _ _ _ _ _ _ _ _ _ _	Lötbrücke S4 geschlossen



**M. K. JUCHHEIM GmbH & Co**

Hausadresse:

Moltkestraße 13 - 31  
36039 Fulda, Germany

Lieferadresse:

Mackenrodtstraße 14  
36039 Fulda, Germany

Postadresse:

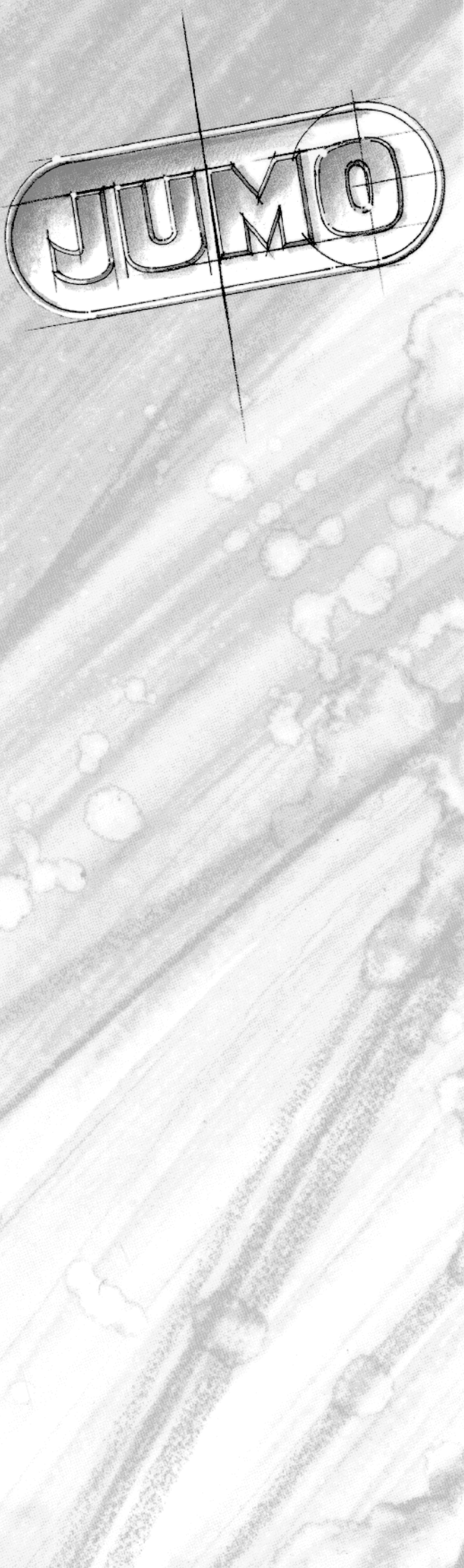
36035 Fulda, Germany

Telefon: (06 61) 60 03-0

Telefax: (06 61) 60 03-5 00

E-Mail: [mail@jumo.net](mailto:mail@jumo.net)

Internet: [www.jumo.de](http://www.jumo.de)



**JUMO** dTRON 04.1

**JUMO** dTRON 08.1

**JUMO** dTRON 16.1

Compact microprocessor  
controller

B 70.3030.2  
Interface description



# Contents

---

## 1 Introduction

1.1	Preface .....	4
1.2	Typographical conventions .....	5
1.2.1	Warning signs .....	5
1.2.2	Note signs .....	5
1.2.3	Presentation .....	5

## 2 Protocol description

2.1	Master-slave principle .....	6
2.2	Transfer mode (RTU) .....	6
2.3	Instrument address .....	7
2.4	Timing of the communication .....	7
2.4.1	Timing of a data request .....	9
2.4.2	Communication during the internal processing time of the slave .....	10
2.4.3	Communication during the response time of the slave .....	10
2.5	Arrangement of the data blocks .....	10
2.6	Error handling .....	10
2.7	Distinction MODbus/Jbus .....	11
2.8	Checksum (CRC16) .....	12

## 3 Functions

3.1	Reading n words .....	14
3.2	Writing one word .....	15
3.3	Writing n words .....	16

## 4 Data flow

## 5 Address tables

5.1	Address tables for <b>JUMO</b> dTRON 04.1/08.1 .....	20
5.2	Address tables for <b>JUMO</b> dTRON 16.1 .....	24

---

# 1 Introduction

---

## 1.1 Preface

Please read this Manual before commissioning the interface. Keep the Manual in a place which is at all times accessible to all users.

Please assist us to improve this Manual.

Your suggestions will be most welcome.

Phone Germany (06 61) 60 03-7 27  
abroad (int.+49) 661 60 03-0  
Fax Germany (06 61) 60 03-5 08  
abroad (int.+49) 661 60 03-607



All the necessary information for operating the interface is contained in this Manual. If any difficulties should still arise during start-up, you are asked not to carry out any manipulations which are not permitted. You could endanger your rights under the instrument warranty!

Please contact the nearest JUMO office or the main factory.



When returning chassis, assemblies or components, the rules of EN 100 015 "Protection of electrostatically sensitive devices" have to be observed. Use only the appropriate ESD packaging material for transport.

Please note that we can not be held liable for damages caused by ESD (electrostatic discharges).

# 1 Introduction

---

## 1.2 Typographical conventions

### 1.2.1 Warning signs

The signs for Danger and Warning are used in this Manual under the following conditions:



**Danger** This symbol may be used when there may be danger to personnel if the instructions are disregarded or not followed accurately!



**Warning** This symbol is used when there may be **danger to equipment or data** if the instructions are disregarded or not followed accurately!



**Warning** This symbol is used where special care is required when handling components liable to damage through electrostatic discharges.

### 1.2.2 Note signs



**Note** This symbol is used when your **special attention** is drawn to a remark.



**Reference** This sign refers to further information in other handbooks, chapters or sections.

abc<sup>1</sup>

**Footnote** Footnotes are notes which refer to certain points in the text. Footnotes consist of two parts:

Marking in the text and the footnote text.

The markings in the text are arranged as continuous raised numbers.

The footnote text (in smaller typeface) is placed at the bottom of the text and starts with a number and a full stop.

### 1.2.3 Presentation

0x0010

**Hexa-  
decimal  
number**

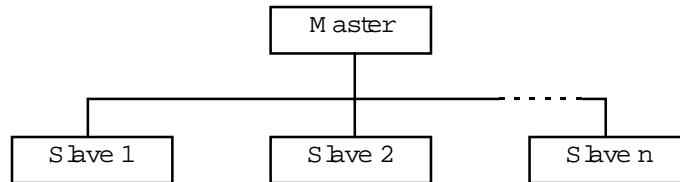
A hexadecimal number is identified by being preceded by “0x” (here 16 decimal).

## 2 Protocol description

---

### 2.1 Master-slave principle

The communication between a PC (master) and an instrument (slave) using MODbus/Jbus takes place according to the master-slave principle in the form of a data request/instruction - response.



The master controls the data exchange, the slaves only have response function. They are identified by their instrument address. A maximum of 255 slaves can be addressed.

### 2.2 Transfer mode (RTU)

The transfer mode used is the RTU (Remote Terminal Unit). Data are transferred in binary format (hexadecimal) with 8 bits, 16 bits for integers, and 32 bits for float values. The LSB (least significant bit) is transferred first. The ASCII operating mode is not supported.

#### Data format

The data format describes the arrangement of a transferred byte. The data format can be as follows:

Data word	Parity bit	Stop bit 1/2 bit	No. of bits
8 bits	—	1	9
8 bits	even	1	10
8 bits	odd	1	10

## 2 Protocol description

---

### 2.3 Instrument address

The address of the slaves can be set between 1 and 31. Address 0 is reserved.



A maximum of 31 slaves can be accessed via the RS422/RS485 interface.

There are two possibilities for data exchange:

**Query** Data request/instruction from the master to a slave via the appropriate address.

The slave which was accessed responds.

**Broadcast** Instruction from the master to all slaves via address 0. The slaves in the system do not respond. A data request with address 0 is not appropriate. For example, a certain setpoint can be transmitted to all slaves. The correct acceptance of the value by the slaves should in this case be checked by a subsequent read-out of the setpoint.

### 2.4 Timing of the communication

Start and end of a data block are identified by transmission pauses. The maximum permitted interval between two consecutive characters is three times the time for transmitting one character.

The character transmission time (time for transmitting a character) depends on the baud rate and the data format used.

If the data format has 8 data bits, no parity bit and one stop bit, then:

$$\text{character transmission time [msec]} = 1000 * 9 \text{ bits/baud rate}$$

For other data formats:

$$\text{character transmission time [msec]} = 1000 * 10 \text{ bits/baud rate}$$

#### Procedure

<b>Data request from master</b> transmission time = n characters * 1000 * x bits/baud rate
Marker for data request end 3 characters * 1000 * x bits/baud rate
Processing the data request by the slave (250msec max.)
<b>Response of slave</b> transmission time = n characters * 1000 * x bits/baud rate
Marker for response end 3 characters * 1000 * x bits/baud rate

## 2 Protocol description

---

### Example

Marker for end of data request or end of response for 10/9 bits data format.

Waiting time = 3 characters \* 1000 \* 10 bits/ baud rate

Baud rate [Baud]	Data format [bit]	Waiting time [msec]
9600	10	3.125
	9	2.813
4800	10	6.250
	9	5.625
2400*	10	12.500
	9	11.250
1200*	10	25.000
	9	22.500

\* **JUMO** dTRON 04.1/08.1 only.

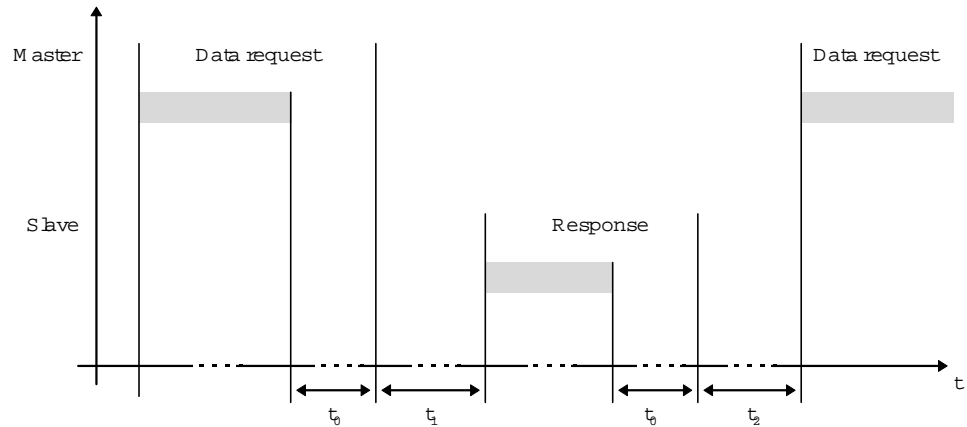
## 2 Protocol description

---

### 2.4.1 Timing of a data request

#### Timing scheme

A data request runs according to the following timing scheme:



- $t_0$  End marker = 3 characters  
(this time depends on the baud rate)
- $t_1$  This time depends on the internal processing.  
The maximum processing time is 250 msec.
- $t_2$  This is the time required by the instrument to switch from transmitting back to receiving. The master must wait this time before making a fresh data request. It must always be maintained, even when the new data request is addressed to a different instrument.  
 $t_2 = 20\text{msec}$  or more

## 2 Protocol description

---

### 2.4.2 Communication during the internal processing time of the slave

The master must not make any data requests during the internal processing time of the slave. Any data requests during this time are ignored by the slave.

### 2.4.3 Communication during the response time of the slave

The master must not make any data requests during the response time of the slave. Any data requests during this period cause all data currently on the bus to become invalid.

## 2.5 Arrangement of the data blocks

All data blocks have the same structure:

### Data structure

Slave address	Function code	Data field	Checksum CRC16
1 byte	1 byte	x byte(s)	2 bytes

Each data block consists of four fields:

<b>Slave address</b>	instrument address of a particular slave
<b>Function code</b>	function selection (reading, writing of words)
<b>Data field</b>	contains the information: <ul style="list-style-type: none"><li>- word address</li><li>- word number</li><li>- word value</li></ul>
<b>Checksum</b>	recognition of transmission errors

## 2.6 Error handling

### Error codes

There are five error codes:

- 1 invalid function
- 2 invalid parameter address
- 3 parameter outside value range<sup>1</sup>
- 4 slave not ready
- 8 write access to parameter inhibited

1. The parameters are not tested for plausibility.

## 2 Protocol description

### Response in case of an error

Slave address	Function XX OR 80h	Error code	Checksum CRC16
1 byte	1 byte	1 byte	2 bytes

The function code is linked by OR with 0x80, i.e. the most significant bit is set to 1.

### Example

Data request:

01	02	00	00	00	04	CRC16
----	----	----	----	----	----	-------

Response:

01	82	01	CRC16
----	----	----	-------

### Specific cases

The slave does not respond in the event of the following errors:

- the checksum (CRC16) is incorrect
- the instruction of the master is incomplete or over-defined
- the number of the words or bits is zero

## 2.7 Distinction MODbus/Jbus

The MODbus protocol is compatible with the Jbus protocol. The structure of the data blocks is identical.



MODbus differs from Jbus in the absolute addresses of the data. The addresses of the MODbus are shifted by one against those of the Jbus.

The Jbus protocol is only supported by **JUMO** dTRON 04.1/08.1.

Absolute address	Jbus address	MODbus address
1	1	0
2	2	1
3	3	2
...	...	...

## 2 Protocol description

---

### 2.8 Checksum (CRC16)

The checksum (CRC16) serves to recognize transmission errors. If an error is identified during processing, the appropriate instrument does not respond.

#### Calculation scheme

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



The low byte of the checksum is transmitted first.

#### Example

Data request: reading two words starting from address 6  
(CRC16 = 0x24A0)

0B	03	00	06	00	02	24	A0
						CRC16	

Response: (CRC16 = 0x6105)

0B	03	04	00	00	42	C8	61	05
			Word 1		Word 2		CRC16	

## 3 Functions

---

The following functions are available to the instrument:

Function number	Function
0x03/0x04	Reading n words
0x06	Writing one word
0x10	Writing n words



All examples refer to addresses of the **JUMO** dTRON 04.1/08.1.

## 3 Functions

---

### 3.1 Reading n words

This function reads n words starting from a defined address.

#### Data request

Slave address	Function 0x03 or 0x04	Address first word	Number of words (6 max.)	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

#### Response

Slave address	Function 0x03 or 0x04	Number of bytes read	Word value(s)	Checksum CRC16
1 byte	1 byte	1 byte	x byte(s)	2 bytes

#### Example

Reading the two setpoints of the controller

Word address = 0x0008 (first setpoint SP1)

Data request:

0B	03	00	08	00	04	CRC16
----	----	----	----	----	----	-------

Response:

0B	03	08	0000	42C8	0000	4316	CRC16
			Setpoint 1 (100)	Setpoint 2 (150)			

# 3 Functions

---

## 3.2 Writing one word

In the “writing word” function the data blocks for instruction and response are identical.

### Instruction

Slave address	Function 0x06	Word address	Word value	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

### Response

Slave address	Function 0x06	Word address	Word value	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

### Example

Write limit of limit comparator 1 (AL1) (= 275)

Word address = 0x000C

Instruction: Write first part of the value

0B	06	00	0C	80	00	CRC16
----	----	----	----	----	----	-------

Response (as instruction):

0B	06	00	0C	80	00	CRC16
----	----	----	----	----	----	-------

Instruction: write second part of the value

0B	06	00	0D	43	89	CRC16
----	----	----	----	----	----	-------

Response (as instruction):

0B	06	00	0D	43	89	CRC16
----	----	----	----	----	----	-------

# 3 Functions

---

## 3.3 Writing n words

### Instruction

Slave address	Function 0x10	Address first word	Number of words	Number of bytes	Word value(s)	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	1 byte	x byte(s)	2 bytes

### Response

Slave address	Function 0x10	Address first word	Number of words	Checksum CRC16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

### Example

Write gradient of ramp function ( $rASd = 100$ ) of the second parameter set

Word address = 0x0072

Instruction:

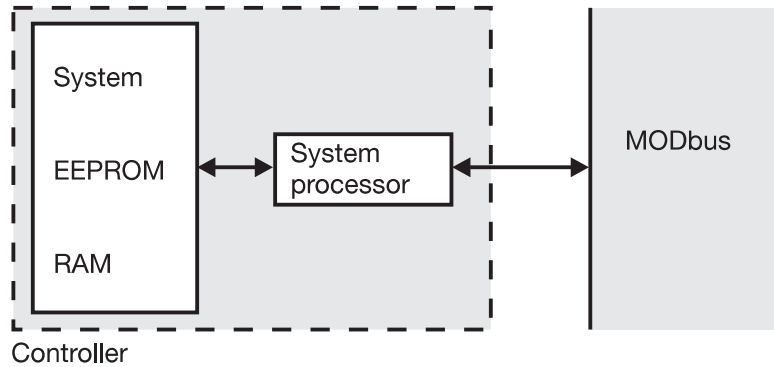
0B	10	00	72	00	02	04	00	00	42	C8	CRC16
----	----	----	----	----	----	----	----	----	----	----	-------

Response:

0B	10	00	72	00	02	CRC16
----	----	----	----	----	----	-------

## 4 Data flow

---



When self-optimisation is activated the interface is switched off. The controller does not report any error.

### Char-type data

In principle, only ASCII characters are transmitted. These are transmitted according to their sequence in the memory.

Example:

Text: 075.02.01

MODbus:

0x30, 0x37, 0x35, 0x2E, 0x30, 0x31, 0x2E, 0x30, 0x31, 0x20, 0x20, 0x00

### Integer-type data

The high byte has to be transmitted first.

Example:

Configuration code C211: 7702

MODbus: 0x07, 0x07, 0x00, 0x02

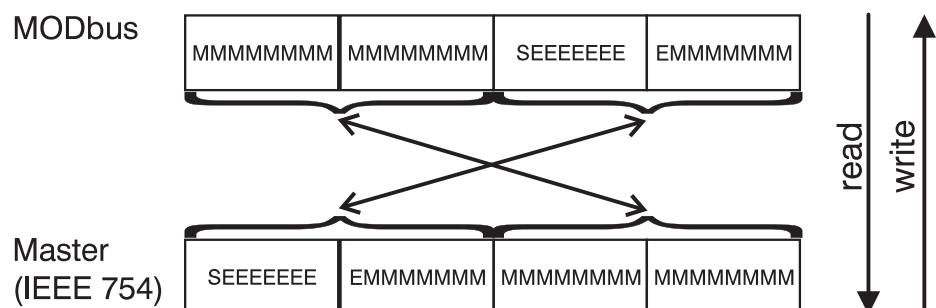
### Float-type data

The explanation below applies under the condition that the master is operating with the IEEE-754 format. Before transmitting a value the bytes have to be rearranged so that their order corresponds to the representation for MODbus (see diagram).

M - 23 bit normalised mantissa

E - exponent (complement base 2)

S - sign bit; 1 = negative, 0 = positive



Example:

Transferring the decimal value 550:

MODbus: 0x80, 0x00, 0x44, 0x09

## 4 Data flow

---

### Byte-type data (status identifiers)

Symbols can only be transferred in 8 bit groups. These 8 bits are preceded by a zero byte (0x00).

Example:

To read the status identifiers of the memory location 0x81. Setpoint 2 and parameter set 2 are activated (B01001000 = 0x48).

MODbus: 0x00, 0x48



Every alteration of a process value which is stored in EEPROM results in updating of the data in EEPROM. Please note that the EEPROM can be rewritten up to 10000 times approx.

## 5 Address tables

---

All process values (variables) with their addresses, the data type and the access mode are described below.

References are as follows:

R/O            read access only

R/W            read and write access

byte            byte (8 bits)

char xx        character chain of length xx;  
xx = length **including** chain end character /0

float           float value (4 bytes/2 words)

int             integer value (2 bytes/1 word)

The process values are divided into logical areas.

## 5 Address tables

### 5.1 Address tables for **JUMO** dTRON 04.1/08.1

#### Instrument data

Address MODbus	Data type/bit number	Access	Signal designation
0x0100	char 12	R/O	Software version 075.xx.xx
0x0106	char 14	R/O	VdN number

Instrument data are read only.

#### Process data

Address MODbus	Data type/bit number	Access	Signal designation
0x0000	float	R/O	Process value analogue input 1
0x0002	float	R/O	Process value analogue input 2
0x0006	float	R/O  R/W	<ul style="list-style-type: none"> <li>- Controller output in automatic operation</li> <li>- Output retransmission on modulating controller</li> <li>- Output in manual operation</li> </ul>
0x0004	float	R/W  R/O	<ul style="list-style-type: none"> <li>- Current setpoint</li> <li>- Ramp setpoint (during ramp function)</li> <li>- external setpoint (with external setpoint input)</li> </ul> <p>This value is not transferred to EEPROM.</p>
0x0008	float	R/W	Setpoint 1
0x000A	float	R/W	Setpoint 2

#### Parameter set 1

Address MODbus	Data type/bit number	Access	Signal designation
0x000C	float	R/W	AL1 - Limit for limit comparator 1
0x000E	float	R/W	AL2 - Limit for limit comparator 2
0x0010	float	R/W	Pb1 - Proportional band 1
0x0012	float	R/W	Pb2 - Proportional band 2
0x0014	float	R/W	dt - Derivative time
0x0016	float	R/W	rt - Reset time
0x0018	float	R/W	tt - Stroke time
0x001A	float	R/W	Cy1 - Cycle time 1
0x001C	float	R/W	Cy2 - Cycle time 2
0x001E	float	R/W	db - Contact spacing

## 5 Address tables

Address MODbus	Data type/ bit number	Access	Signal designation
0x0020	float	R/W	HYS.1 - Differential 1
0x0022	float	R/W	HYS.2 - Differential 2
0x0024	float	R/W	Y0 - Working point
0x0026	float	R/W	Y1 - Maximum output
0x0028	float	R/W	Y2 - Minimum output
0x002A	float	R/W	dF - Filter time constant
0x002C	float	R/W	rASd - Ramp slope
0x002E	float	R/W	tS - Time of hold phase  (only on controllers with start-up ramp!)

### Parameter set 2

Address MODbus	Data type/ bit number	Access	Signal designation
0x0056	float	R/W	Pb1 - Proportional band 1
0x0058	float	R/W	Pb2 - Proportional band 2
0x005A	float	R/W	dt - Derivative time
0x005C	float	R/W	rt - Reset time
0x005E	float	R/W	tt - Stroke time
0x0060	float	R/W	Cy1 - Cycle time 1
0x0062	float	R/W	Cy2 - Cycle time 2
0x0064	float	R/W	db - Contact spacing
0x0066	float	R/W	HYS.1 - Differential 1
0x0068	float	R/W	HYS.2 - Differential 2
0x006A	float	R/W	Y0 - Working point
0x006C	float	R/W	Y1 - Maximum output
0x006E	float	R/W	Y2 - Minimum output
0x0070	float	R/W	dF - Filter time constant
0x0072	float	R/W	rASd - Ramp slope

### Configuration level

Address MODbus	Data type/ bit number	Access	Signal designation
0x030	int 02	R/W	C111 (analogue input 1 + 2)
0x0032	int 02	R/W	C112 (general instrument data)
0x0034	int 02	R/W	C113 (interface)
0x0036	int 02	R/W	C211 (limit comparators)
0x0038	int 02	R/W	C212 (controller settings and output 3)
0x003A	int 02	R/W	C213 (outputs 1, 2, 4 and 5)

## 5 Address tables

Address MODbus	Data type/bit number	Access	Signal designation
0x003C	float	R/W	SCL (standard signal scaling: start of measurement range)
0x003E	float	R/W	SCH (standard signal scaling: end of measurement range)
0x0040	float	R/W	SPL (low setpoint limit)
0x0042	float	R/W	SPH (high setpoint limit)
0x0044	float	R/W	OFFS (process value correction)

### Status identifiers

Status identifiers are read in 8-bit groups in byte data format. Alteration of the identifiers via the interface is not possible.

Identifier	Address MODbus	Position	Meaning
Output 1	0x80	----- 1	Output off (contact open)
Output 2		----- 1 _	Output off (contact open)
Output 3		----- 1 _ _	Output off (contact open)
Output 4		--- 1 _ _ _ _	Output off (contact open)
Output 5		-- 1 _ _ _ _ _	Output off (contact open)
Manual operation		_ 1 _ _ _ _ _ _	Manual operation activated
Key inhibit	0x81	----- 1	Keys inhibited
Level inhibit		----- 1 _	Parameter and configuration levels inhibited
Ramp stop		----- 1 _ _	Ramp function or start-up ramp held
Setpoint switching		----- 1 _ _ _	Setpoint SP2 activated
Interface setpoint		--- 1 _ _ _ _	Interface setpoint activated
Hold phase (start-up ramp)		-- 1 _ _ _ _ _	Hold phase reached
Parameter set switching		_ 1 _ _ _ _ _ _	Parameter set 2 activated

## 5 Address tables

---

Identifier	Address MODbus	Position	Meaning
Over/underrange input 1	0x82	_____ 1	Over/underrange input 1
Over/underrange input 2		_____ 1 _	Over/underrange input 2
Logic input 1		_____ 1 __	Contact closed
Logic input 2		_____ 1 ___	Contact closed
Controller output 1		___ 1 _____	Controller output activated
Controller output 2		__ 1 _____	Controller output activated
Limit comparator 1		_ 1 _____	Switching status ON
Limit comparator 2		1 _____	Switching status ON

## 5 Address tables

### 5.2 Address tables for **JUMO** dTRON 16.1

#### Instrument data

Address MODbus	Data type/ bit number	Access	Signal designation
0x0301	char[10]	R/O	Instrument name
0x0306	char[12]	R/O	Software version
0x030C	char[14]	R/O	VdN number

#### Process data

Address MODbus	Data type/ bit number	Access	Signal designation
0x0000	float	R/O	Process value analogue input 1
0x0002	float	R/W	current setpoint
0x0004	float	R/W	SP1
0x0006	float	R/W	SP2

#### Parameter set 1

Address MODbus	Data type/ bit number	Access	Signal designation
0x0008	float	R/W	AL1 - Limit for limit comparator 1
0x000A	float	R/W	AL2 - Limit for limit comparator 2
0x000C	float	R/W	Pb1 - Proportional band 1
0x000E	float	R/W	Pb2 - Proportional band 2
0x0010	float	R/W	dt - Derivative time
0x0012	float	R/W	rt - Reset time
0x0014	float	R/W	Cy1 - Cycle time 1
0x0016	float	R/W	Cy2 - Cycle time 2
0x0018	float	R/W	db - Contact spacing
0x001A	float	R/W	HYS.1 - Differential 1
0x001C	float	R/W	HYS.2 - Differential 2
0x001E	float	R/W	Y0 - Working point
0x0020	float	R/W	Y1 - Maximum output
0x0022	float	R/W	Y2 - Minimum output
0x0024	float	R/W	dF - Filter time constant
0x0026	float	R/W	rASd - Ramp slope

#### Parameter set 2

Address MODbus	Data type/ bit number	Access	Signal designation
0x003E	float	R/W	Pb1 - Proportional band 1
0x0040	float	R/W	Pb2 - Proportional band 2
0x0042	float	R/W	dt - Derivative time

## 5 Address tables

Address MODbus	Data type/ bit number	Access	Signal designation
0x0044	float	R/W	rt - Reset time
0x0046	float	R/W	Cy1 - Cycle time 1
0x0048	float	R/W	Cy2 - Cycle time 2
0x004A	float	R/W	db - Contact spacing
0x004C	float	R/W	HYS.1 - Differential 1
0x004E	float	R/W	HYS.2 - Differential 2
0x0050	float	R/W	Y0 - Working point
0x0052	float	R/W	Y1 - Maximum output
0x0054	float	R/W	Y2 - Minimum output
0x0056	float	R/W	dF - Filter time constant
0x0058	float	R/W	rASd - Ramp slope

### Configuration level

Address MODbus	Data type/ bit number	Access	Signal designation
0x0028	char[4]	R/W	C111
0x002A	char[4]	R/W	C112
0x002C	char[4]	R/W	C113
0x002E	char[4]	R/W	C114
0x0030	char[4]	R/W	C000 (can only be set via the interface)
0x0032	float	R/W	SCL
0x0034	float	R/W	SCH
0x0036	float	R/W	SPL
0x0038	float	R/W	SPH
0x003A	float	R/W	OFFS
0x003C	float	R/W	HYST

### Program data

Address MODbus	Data type/ bit number	Access	Signal designation
0x005A	float	R/W	SP00
0x005C	float	R/W	t00
0x005E	float	R/W	SP01
0x0060	float	R/W	t01
0x0062	float	R/W	SP02
0x0064	float	R/W	t02
0x0066	float	R/W	SP03
0x0068	float	R/W	t03

## 5 Address tables

**Status  
identifiers**

Address MODbus	Data type	Signal designation
0x0200	word	Outputs and logic functions
	----- 1	Output 1 off
	----- 1 _	Output 2 off
	----- 1 __	Output 3 off
	----- 1 ___	Output 4 off
	----- 1 ____	Output 5 off
	----- 1	Keys inhibited
	----- 1 _	Parameter and configuration levels inhibited
	----- 1 __	Program or ramp function held
	----- 1 ___	Second setpoint activated
	----- 1 ____	Interface setpoint activated
	----- 1	Calibration mode
	----- 1 _	Second parameter set activated
	1 -----	Limit comparators enabled

## 5 Address tables

Address MODbus	Data type	Signal designation
0x0201	word	Logic signals and hardware ID
	----- 1	Out-of-range input 1
	----- 1 _	Out-of-range input 2
	----- 1 _ _	Logic input 1 closed
	----- 1 _ _ _	Logic input 2 closed
	----- 1 _ _ _ _	Controller output 1 activated
	----- 1 _ _ _ _ _	Controller output 2 activated
	----- 1 _ _ _ _ _	Limit comparator 1 activated
	----- 1 _ _ _ _ _	Limit comparator 2 activated
	---- 1 0 _ _ _ _ _	Relay output 3 available
	---- 0 1 _ _ _ _ _	Logic output 3 available
	---- 1 1 _ _ _ _ _	Transistor output 3 available
	--- 1 _ _ _ _ _	Analogue output 3 available
	--- 1 _ _ _ _ _	Analogue output voltage
	----- 1 _ _ _ _ _	Analogue input 2 available
	----- 1 _ _ _ _ _	Interface available
1 _ _ _ _ _	Solder link S4 closed	



**M. K. JUCHHEIM GmbH & Co**

Street address:  
Moltkestraße 13 - 31  
36039 Fulda, Germany  
Delivery address:  
Mackenrodtstraße 14  
36039 Fulda, Germany  
Postal address:  
36035 Fulda, Germany  
Phone: +49 661 60 03-0  
Fax: +49 661 60 03-6 07  
E-mail: [mail@jumo.net](mailto:mail@jumo.net)  
Internet: [www.jumo.de](http://www.jumo.de)

**JUMO Instrument Co. Ltd.**

JUMO House  
Temple Bank, Riverway  
Harlow, Essex CM20 2TT, UK  
Phone: +44 12 79 63 55 33  
Fax: +44 12 79 63 52 62  
E-mail: [sales@jumo.co.uk](mailto:sales@jumo.co.uk)

**JUMO PROCESS CONTROL INC.**

885 Fox Chase, Suite 103  
Coatesville, PA 19320, USA  
Phone: 610-380-8002  
1-800-554-JUMO  
Fax: 610-380-8009  
E-mail: [info@JumoUSA.com](mailto:info@JumoUSA.com)  
Internet: [www.JumoUSA.com](http://www.JumoUSA.com)