

MODBUS – Schnittstelle

Schnittstellenbeschreibung und Datenübertragungsprotokoll

- MODBUS RTU / ASCII
- RS485 / 2-Draht Anschluss / Halb-Duplex Übertragung
- Anschluss über frontseitigen 9-DSUB-Stecker
- 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 Baud

Inhaltsangabe

1. ALLGEMEINES.....	2
2. RS485 GRUNDLAGEN.....	2
2.1. Beschreibung.....	2
2.2. Abschlusswiderstände.....	2
2.3. Bezugsleiter.....	3
2.4. Anschluss RS485 an einen Master (SPS / Bedienpanel / PC).....	3
2.5. Anschluss RS485 an einen Messumformer.....	3
3. MODBUS GRUNDLAGEN.....	3
3.1. Master-Slave Verfahren.....	3
3.2. Timeout.....	3
3.3. Beschreibung des MODBUS – Rahmen.....	3
3.4. Funktions-Codes in MODBUS.....	4
3.5. Alternative Übertragungsmodi (RTU / ASCII).....	4
3.6. Register.....	5
3.7. Parametertabelle.....	6
3.8. Adressen der Parameter.....	6
3.9. Datentyp.....	6
3.10. Fehlermeldungen.....	6
3.11. Beispiel Telegramme.....	7
3.12. Wortreihenfolge.....	8
3.13. Baudrate.....	8
3.14. Weitere Informationen über MODBUS.....	8
4. BUSKONFIGURATION.....	9
4.1. Konfiguration des Masters.....	9
4.2. Konfiguration des Messumformers.....	9
5. BESONDERHEITEN UND VEREINBARUNGEN.....	11
5.1. Dezimalstellen.....	11
5.2. Übertragung von Tabellen.....	11
6. ANHANG.....	12
6.1. ASCII Tabelle.....	12
6.2. Parametertabellen.....	12

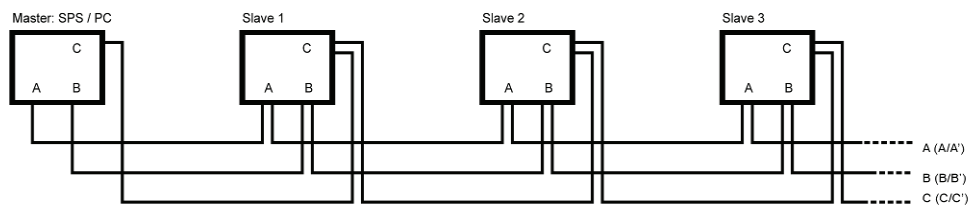
1. Allgemeines

Messumformer können optional mit einer RS485-Schnittstelle geliefert werden. Bereits bei der Bestellung muss berücksichtigt werden, ob die serielle Schnittstelle benötigt wird, eine Nachrüstung ist nicht möglich. Über die Schnittstelle können Parameter gelesen oder geschrieben werden. Zusätzlich zur manuellen Konfiguration, können die Messumformer dann auch per Schnittstelle konfiguriert werden. In dieser Beschreibung wird speziell auf die serielle Schnittstelle mittels MODBUS eingegangen.

2. RS485 Grundlagen

2.1. Beschreibung

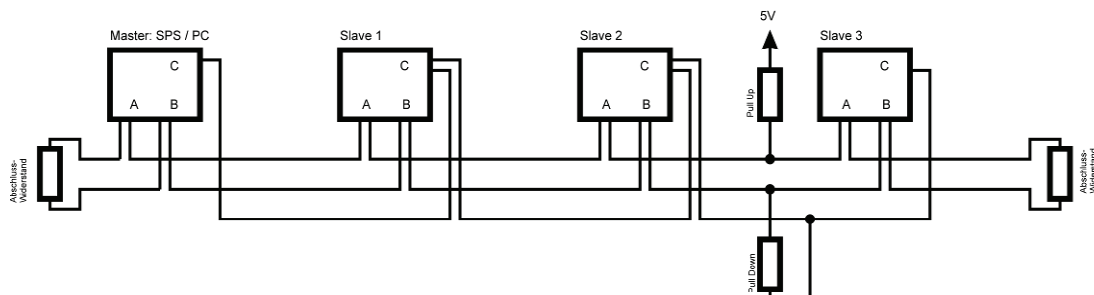
Die MODBUS Geräte werden über eine RS485-Schnittstelle (EIA-485) miteinander verbunden. Dies ist ein Schnittstellen-Standard für leitungsgebundene, differentielle Datenübertragung, wobei an einem Adernpaar mehrere Sender / Empfänger angeschlossen sein dürfen. Bei vorliegenden Geräten wird ein 2-Draht-Bus verwendet (halbduplexfähig). Da nur ein Übertragungsweg zur Verfügung steht, kann immer nur ein Teilnehmer Daten senden. Der RS485-2-Draht-Bus besteht gemäß Skizze aus dem eigentlichen Buskabel mit einer max. Länge von 1200m. Die dritte Leitung (C / C') dient als Bezugsleiter.



Bei dem vorliegenden Bussystem sind ein Master (SPS / Bedienpanel / PC) und maximal 32 Slaves vorgesehen. Die maximale Anzahl der Slaves kann über Segmentkoppler erhöht werden. Die einzelnen Geräte / Messumformer (Slaves) sind in Linien- bzw. Bustopologien abgeordnet. Das bedeutet, dass die Leitung A (häufig gekennzeichnet als A/A') vom ersten Gerät auf den zweiten, vom zweiten auf den dritten, etc., geführt wird. Gleiches gilt für B (B/B'). Eine sternförmige Verteilung und Stichleitungen sind zwingend zu vermeiden (kein Einsatz von Verteilerdosen).

2.2. Abschlusswiderstände

An den offenen Kabelenden (erster und letzter Teilnehmer in einem Bussystem) entstehen immer Leitungsreflexionen. Diese sind um so stärker, je größer die gewählte Baudrate ist. Um die Reflexionen möglichst gering zu halten, findet ein Abschlußwiderstand Einsatz. Durch die Pull Up / Down Widerstände wird ein definiertes Ruhepotential erreicht.



In der Praxis haben sich für die Abschlußwiderstände 150 Ω , als Pull-Up/Down Widerstände 390 Ω als sinnvoll erwiesen. Eine generelle Aussage kann aber nicht getroffen werden, da dies von vielen Parametern abhängt (verwendete Master / Slaves, Anzahl, Kabellängen, Kabelart, Übertragungsgeschwindigkeit). Folglich muss der Anwender dies einfach ausprobieren.

MODBUS – Schnittstelle

2.3. Bezugsleiter

Viele Geräte von Drittanbietern (auch RS485-Schnittstellen-Converter für PCs) führen die Leitung C (Common, Bezugsleiter) nicht nach aussen. Dies entspricht nicht den Empfehlungen der Spezifikation *MODBUS over serial line specification and implementation guide*. Da das differentielle Signal zwischen A/A' und B/B' ausgewertet wird, funktioniert die Kommunikation auch, wenn der Bezugsleiter weggelassen wird.

2.4. Anschluss RS485 an einen Master (SPS / Bedienpanel / PC)

Viele SPS verfügen über eine RS485 Schnittstelle. Sofern auch die entsprechenden Treiber für MODBUS RTU bzw. ASCII vorliegen ist eine einfache Einbindung möglich. Gleiches gilt für diverse Bedienpanels, die häufig standardmäßig MODBUS RTU / ASCII (siehe 3.5) über RS485 unterstützen.

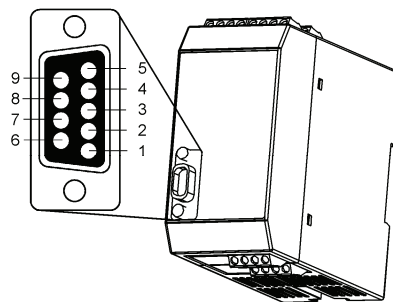
Aber auch ein Standard-PC kann als Master genutzt werden. Hierzu sind RS485 Schnittstellen-Karten von Drittanbietern verfügbar. Außerdem gibt es eine Vielzahl an externen Schnittstellenkonvertern (z.B. RS232 auf RS485, USB auf RS485). Es wird empfohlen hier Lösungen zu wählen, die das BUS-System und die eigentliche PC-Schnittstelle galvanisch trennen.

Die Anschlussbelegungen der Master-Schnittstellen sind den Dokumentationen der entsprechenden Geräte zu entnehmen.

2.5. Anschluss RS485 an einen Messumformer

Die Geräte verfügen in der Front über eine 9 poligen DSUB Buchse mit folgender Pinbelegung (MODBUS-Version):

PIN	EIA-485 Bezeichnung
A / A'	9
B / B'	5
C / C'	1



3. MODBUS Grundlagen

3.1. Master-Slave Verfahren

Die Daten werden per MODBUS-Protokoll übertragen. Die Kommunikation erfolgt nach dem Master-Slave-Verfahren. Die Kommunikation wird immer vom Master (PC, SPS, o.ä.) durch eine Anfrage begonnen. Jeder Slave hat eine Adresse (gültiger Bereich 1 – 247), die einmalig vergeben ist. Erkennt ein Slave, dass seine Adresse vom Master angesprochen wurde, reagiert er entsprechend. Der Slave sendet daraufhin immer eine Antwort. Die Slaves kommunizieren niemals untereinander. Sie sind auch nicht in der Lage eine Kommunikation mit dem Master zu beginnen.

3.2. Timeout

Nach einer Anfrage wartet der Master auf eine Antwort. Hierbei wird eine Timeout-Funktion realisiert. So wird verhindert, dass der Master ewig wartet, wenn ein Slave vom Bus entfernt wurde, keine Betriebsspannung hat oder ein anderer Fehler vorliegt.

3.3. Beschreibung des MODBUS – Rahmen

Die zu übertragenden Daten liegen immer in einem festgelegtem Rahmen (Frame) vor und sind wie folgt definiert:

Feld 1	Feld 2	Feld 3	Feld 4
Adresse	Funktions-Code	Daten	CRC / LRC

Beschreibung der einzelnen Felder:

Feld 1)	Adresse	Hier wird die Adresse des Slave angegeben (gültiger Bereich 1 – 247 _{Dez})
Feld 2)	Funktions-Code	Hier wird festgelegt, ob Parameter gelesen oder geschrieben werden sollen.
Feld 3)	Daten	<ul style="list-style-type: none"> - z.B. vom Master: Welche Parameter werden angefragt? - z.B. vom Slave: Inhalt der abgefragten Parameter,
Feld 4)	CRC / LRC	<p>Die zyklische Redundanzprüfung; Prüfwert für Daten um Fehler bei der Datenübertragung zu erkennen. 2 mögliche Verfahren: CRC (engl. Cyclic Redundancy Check) LRC (engl. Longitudinal Redundancy Check)</p>

3.4. Funktions-Codes in MODBUS

In MODBUS sind mehrere Funktions-Codes definiert. Sie lauten:

Funktions-Code	Name	Beschreibung
01 _{Hex}	Read Coils	Geräteparameter im Bit-Format lesen
02 _{Hex}	Read Discrete Inputs	Istwerte im Bit-Format lesen
03 _{Hex}	Read Hold. Register	Geräteparameter lesen (Integer / Float)
04 _{Hex}	Read Input Register	Istwerte lesen (Integer / Float)
05 _{Hex}	Write Single Coil	Geräteparameter im Bit-Format schreiben
06 _{Hex}	Write Single Register	Geräteparameter wortweise schreiben
15 _{Hex}	Write Multiple Coils	mehrere Geräteparameter im Bit-Format schreiben
16 _{Hex}	Write Multiple Register	mehrere Geräteparameter wortweise schreiben

3.5. Alternative Übertragungsmodi (RTU / ASCII)

RTU

Im RTU Modus sind die einzelnen Bytes wie folgt definiert:

Codiert in: 8 Bit Binär

Ein Byte setzt sich aus 11 Bits zusammen, da z.B. Start/Stopp-Bits verwendet werden:

Start Bit	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7	BIT 8	Parität	Stopp Bit
-----------	-------	-------	-------	-------	-------	-------	-------	-------	---------	-----------

Die Parität dient der Erkennung einer fehlerhaft übertragener Folge von Bits. "Parität" bezeichnet die Anzahl der mit 1 belegten Bits im Informationswort und heißt gerade (engl. "even"), wenn die Anzahl dieser Bits gerade ist, andernfalls ungerade (engl. "odd"). Soll keine Parität verwendet werden setzt sich ein Byte trotzdem aus 11 Bits zusammen, es wird einfach ein weiteres Stopp-Bit angehängt:

Start Bit	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7	BIT 8	Stopp Bit	Stopp Bit
-----------	-------	-------	-------	-------	-------	-------	-------	-------	-----------	-----------

MODBUS – Schnittstelle

Im RTU Modus sieht dann ein Frame wie folgt aus:

Feld 1 Adresse	Feld 2 Funktions-Code	Feld 3 Daten	Feld 4 CRC
1 Byte	1 Byte	0 ... 252 Byte(s)	2 Bytes CRC Low CRC High

Zwischen den Frames muss eine minimale Pause von 3,5 Zeichen liegen, so wird das Ende einer Nachricht erkannt, RTU verwendet immer CRC (Feld 4).

ASCII

Im ASCII Modus sind die einzelnen Bytes wie folgt definiert:

Codiert in: Hexadezimal, ASCII Zeichen („0“ - „9“ ; „A“ - „F“)

Ein Byte setzt sich aus 10 Bits zusammen. Wenn Parität verwendet wird gilt:

Start Bit	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7	Parität	Stopp Bit
-----------	-------	-------	-------	-------	-------	-------	-------	---------	-----------

Ohne Parität werden 2 Stopp-Bit verwendet, so dass gilt:

Start Bit	BIT 1	BIT 2	BIT 3	BIT 4	BIT 5	BIT 6	BIT 7	Stopp Bit	Stopp Bit
-----------	-------	-------	-------	-------	-------	-------	-------	-----------	-----------

Im ASCII Modus sieht dann ein Frame wie folgt aus:

Start	Feld 1 Adresse	Feld 2 Funktions-Code	Feld 3 Daten	Feld 4 LRC	Ende
1 Zeichen „:“	2 Zeichen	2 Zeichen	0 ... 2 x 252 Zeichen	2 Zeichen	2 Zeichen „CR“ + „LF“

Der Beginn der Nachricht wird durch 3A_{HEX} „:“ gekennzeichnet und endet mit den beiden Zeichen 0x0D_{HEX} + 0x0A_{HEX} „carriage return + line feed“.

Der ASCII Modus verwendet LRC (Feld 4) als Redundanzprüfung.

Der Vergleich zwischen RTU und ASCII macht deutlich, dass bei RTU ein höherer Datendurchsatz bei gleicher Übertragungsgeschwindigkeit möglich ist, als bei ASCII Übertragung.

Bspl.: Das Byte 5B_{Hex} kann in RTU direkt als ein Byte übertragen werden.

In ASCII wird daraus: 5B_{Hex} → „5“ (Code: 35_{Hex}) und „B“ (Code: 42_{Hex})

3.6. Register

MODBUS sieht vor, die Daten in verschiedenen Registern zu speichern. Ein Register speichert jeweils 2 Byte. Folgende Register werden unterschieden:

Registernummer / Register-Adresse	Art	MODBUS Bezeichnung	Beschreibung
1 – 9999	lesen / schreiben	Discrete Output Coils	Diskrete Ausgänge
10001 – 19999	nur lesen	Discrete Input Contacts	Diskrete Eingänge
30001 – 39999	nur lesen	Input Registers	Eingangsregister, Messwerte des Messumformers
40001 – 49999	lesen / schreiben	Output Holding Registers	Halteregister für Parameter, Gerätekfiguration, etc.

MODBUS – Schnittstelle

3.7. Parametertabelle

Zu jedem Slave gehört eine Parametertabelle und eine Istwerttabelle. Aus diesen Tabellen kann entnommen werden, unter welchen Adressen welche Parameter zu finden sind.

Auszug aus einer Parametertabelle:

Register-Adresse	Protokoll-Adresse	Parameter Name	Wertebereich	Bedeutung	Datentyp	Berechtigung
40001	0	Sprache	0...1	0: Deutsch 1: Englisch	Short	R/W
40002	1	Kontrast	0...100	[%]	Short	R/W
40003	2	Dezimalstellen	0...3	0: 0000 1: 000,0 2: 00,00 3: 0,000	Short	R/W
...	

Auszug aus einer Istwerttabelle:

Register-Adresse	Protokoll-Adresse	Parameter Name	Wertebereich	Datentyp	Berechtigung
30001	0	Istwert 1	-9999...9999	Float	R
30003	2	Istwert 2	-9999...9999	Float	R
30005	4	Minimaler Istwert-Speicher	-9999...9999	Float	R
30007	6	Maximaler Istwert-Speicher	-9999...9999	Float	R
...	

Die Spalte Datentyp gibt an, in welchem Format ein Parameter/Istwert geschrieben bzw. gelesen werden kann. Die Spalte Berechtigung gibt an ob ein Parameter/Istwert nur lesbar (R) oder schreib- und lesbar (R/W) ist.

3.8. Adressen der Parameter

Aus obigen Tabellen können die Register-Adressen und die Protokoll-Adressen entnommen werden. Eine SPS oder ein Bedienpanel arbeitet in der Regel mit den Register-Adressen. In dem eigentlichen Telegramm wird die Protokoll-Adresse verwendet.

3.9. Datentyp

Den Tabellen Parametertabelle / Istwerttabelle kann der entsprechende Datentyp entnommen werden.

SHORT ist hierbei ein 16 Bit Integer – Wert. Der SHORT-Wert belegt 1 Register.

FLOAT ist hierbei eine 32 Bit Fließkommazahl, die Darstellung erfolgt nach IEEE 754 / IEC-60559. Der FLOAT-Wert belegt 2 Register.

3.10. Fehlermeldungen

Der Slave antwortet mit Fehlermeldungen, wenn der Master z.B. auf Parameteradressen zugreift, die nicht definiert sind. Dies gilt auch dann, wenn der Master an den Slave Werte übergibt, die ausserhalb des definierten Wertebereichs liegen (z.B. Master fordert 4 Nachkommastellen, maximal sind aber nur 3 möglich).

Im Fehlerfall sendet der Slave folgendes Telegramm:

Feld 1	Feld 2	Feld 3	Feld 4
Adresse	Funktions-Code + 80 _{Hex}	Fehler-Code	CRC

zu Feld 2) Zu dem Funktionscode, der in der Anfrage des Masters verwendet wurde, werden 80_{Hex} addiert. So kann der Master sofort erkennen, dass ein Fehler vorliegt und dann gegebenenfalls Feld 3 mit dem Fehler-Code auswerten.

Folgende Fehlermeldungen werden unterstützt:

Fehler-Code	Modbus-Bezeichnung	Beschreibung
01 _{Hex}	Illegal Function	Nicht unterstützter Funktionscode
02 _{Hex}	Illegal Data Address	Geräteparameter nicht vorhanden
03 _{Hex}	Illegal Data Value	Schreibzugriff liegt außerhalb des Wertebereichs oder Parameter ist schreibgeschützt
06 _{Hex}	Slave Device Busy	Slave ist gerade beschäftigt, kann nicht reagieren

Bei Geräten aus der X50 Familie blinkt die BUS-LED in der Gerätefront rot, wenn ein Fehler vorliegt.

3.11. Beispiel Telegramme

Der Master soll von einem Slave mit der Adresse 1_{Dez} (01_{Hex}) den Maximalen-Istwert-Speicher einlesen. Verwendet wird nach [3.4 Funktions-Codes] Code Nr. 04_{Hex}. Der Maximal-Wert liegt im Slave unter der Protokoll-Adresse 06_{Hex}.

Da der Istwert-Speicher als FLOAT vorliegt, müssen 2 Register gelesen werden. Für den Rahmen gilt:

Feld 1	Feld 2	Feld 3	Feld 4
Adresse	Funktions-Code	Daten	CRC

Daraus erhalten wir die Anfrage, die der Master stellen muss:

Feld 1	Feld 2	Feld 3	Feld 4
01 _{Hex}	04 _{Hex}	00 _{Hex} 06 _{Hex} 00 _{Hex} 02 _{Hex}	91 _{Hex} CA _{Hex}

Interpretation der Daten in Feld 3:

00 _{Hex}	}	(00 06) _{Hex}	Startadresse / Protokolladresse: 6
06 _{Hex}			
00 _{Hex}	}	(00 02) _{Hex}	Anzahl der zu lesenden Register: 2
02 _{Hex}			

Der Slave mit der Adresse 1 antwortet darauf mit:

Feld 1	Feld 2	Feld 3	Feld 4
01 _{Hex}	04 _{Hex}	04 _{Hex} 00 _{Hex} 00 _{Hex} 41 _{Hex} 10 _{Hex}	CA _{Hex} 18 _{Hex}

Interpretation der Daten in Feld 3:

04 _{Hex}	}	(00 00 41 10) _{Hex}	Anzahl der folgenden Bytes
00 _{Hex}			
00 _{Hex}			
41 _{Hex}			Istwert als Float, 2 Register
10 _{Hex}			

Die 32 Bit hexadezimale Zahl (00 00 41 10)_{Hex} ergibt in Intelbyte-Reihenfolge 9,000.

MODBUS – Schnittstelle

3.12. Wortreihenfolge

Unterschieden werden 2 mögliche Wortreihenfolgen:

INTEL: gewohnte Darstellungsform von LONG- und FLOAT-Zahlen, zuerst Low-Word, dann High-Word.

MODBUS: MODBUS – konforme Darstellungsform von LONG- und FLOAT-Zahlen, zuerst High-Word, dann Low-Word.

Das Feld für die Adressen und Funktions-Codes bleiben davon unberührt, da dies nur Auswirkungen auf die LONG- und FLOAT-Zahlen hat. Um dies zu verdeutlichen auch hier wieder ein Beispiel-Telegramm:

Format	Feld 1 Adresse	Feld 2 Funktions-Code	Feld 3 Daten		Feld 4 CRC
			Anzahl d. Bytes	FLOAT-Zahl	
Modbus	01 _{Hex}	04 _{Hex}	04 _{Hex}	D4 _{Hex} 96 _{Hex} C1 _{Hex} 8B _{Hex}	33 _{Hex} AF _{Hex}
Intel	01 _{Hex}	04 _{Hex}	04 _{Hex}	C1 _{Hex} 8B _{Hex} D4 _{Hex} 96 _{Hex}	69 _{Hex} 3C _{Hex}

In beiden Fällen wird die 32 Bit FLOAT-Zahl –17,479 dargestellt.

Im gesamten Bussystem muss daher eine Festlegung getroffen werden, welches Format verwendet wird, da die Daten sonst falsch interpretiert werden. Einige Master nennen das INTEL-Format auch LONG/FLOAT Inverse.

Die MODBUS Spezifikation sieht nur das MODBUS-Format vor, daher ist dies auch die Werkseinstellung!

3.13. Baudrate

Die Baudrate gibt die Schrittgeschwindigkeit einer Datenübertragung an. Die Baudrate muss bei allen Teilnehmern im Bussystem (Master und Slaves) gleich sein. Die X50 Familie unterstützt Baudraten mit 300, 600, 1200, 2400, 4800, 9600, 19200 und 38400 Baud.

3.14. Weitere Informationen über MODBUS

Weitere Informationen über das MODBUS Protokoll finden Sie in den frei zugänglichen Spezifikationen der Modbus-Nutzerorganisation (Internetadresse WWW.MODBUS.ORG).

- MODBUS application protocol specification
- MODBUS over serial line specification and implementation guide

In diesen Dokumenten wird auch erläutert, wie z.B. die CRC / LRC Berechnung durchgeführt werden.

4. Buskonfiguration

4.1. Konfiguration des Masters

Mit den entsprechenden SPS / Panel Konfigurations- und Programmier-Softwaretools ist ein einfacher Zugriff möglich. Lediglich die Konfiguration des Bussystems (RTU oder ASCII, welche Baudrate, welche Art der Parität wird verwendet, etc.) ist zu erstellen. Natürlich muss noch eingestellt werden, welcher Parameter nun gelesen / geschrieben werden muss und in welchem Format er vorliegt (siehe 3.6). Mit dem eigentlichen Telegrammaufbau, oder den CRC Berechnungen, hat der Anwender dann nichts mehr zu tun.

Aber auch ein Standard PC kann als Master genutzt werden. Dies ist z.B. interessant, wenn eine Vielzahl an Geräten immer mit der gleichen Konfiguration parametrisiert werden sollen oder eine einfache Visualisierung realisiert werden soll. Die Daten können mit entsprechender Software auch in Tabellenkalkulationen weiterverwendet werden. Dies ermöglicht ein einfaches Mitloggen der Daten und eine spätere Weiterverarbeitung. Hierzu stehen eine Reihe an Softwarelösungen von Drittanbietern zur Verfügung.

Aufgrund der Vielzahl der Lösungsmöglichkeiten kann im Rahmen dieser Dokumentation auf die eigentliche Programmierung nicht näher eingegangen werden.

4.2. Konfiguration des Messumformers

Die Modbus-Gerätekonfiguration kann bei den Messumformern / Anzeigegegeräten in der Konfigurationsebene im Untermenü BUS eingestellt werden.

Inbetriebnahmehinweis!

Das Gerät ist werksseitig mit einer Standardeinstellung vorbelegt.
Es muss daher noch an den speziellen Einsatzfall angepasst werden.

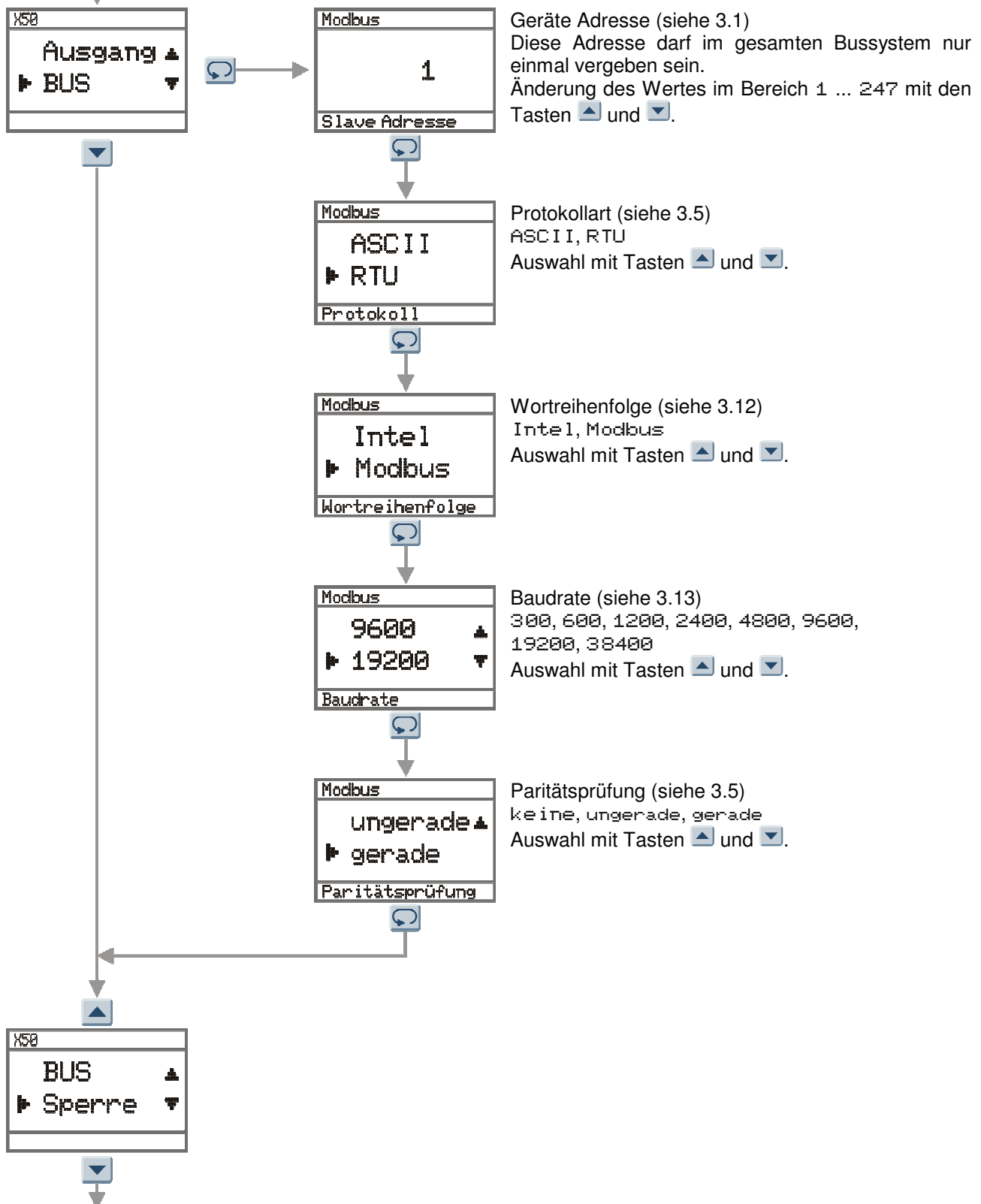
Dabei sind folgende Punkte zu beachten:

Geräteadresse	Die Adresse eines Slaves darf im Bussystem nur einmal vergeben sein.
Protokollart Wortreihenfolge Baudrate Paritätsprüfung	Diese Parameter müssen bei allen Geräten im Bussystem (Master und Slaves) identisch eingestellt werden.

Zur eigentlichen Konfiguration des Messumformers, siehe gesonderte Bedienungsanleitung.

Beschreibung

(eingetragene Werte sind Werkseinstellungen)



5. Besonderheiten und Vereinbarungen

5.1. Dezimalstellen

Wird der Parameter „Dezimalstellen“ verändert, erfolgt automatisch eine Anpassung aller betroffenen Parameter. Es wird automatisch gerundet.

Bspl.:	<i>Vorher</i>		<i>Nachher</i>	
	Dezimalstellen	= 2	Dezimalstellen	= 0
	„Grenzwert Alarmkontakt 1“	= 9,65	„Grenzwert Alarmkontakt 1“	= 10

Aber auch folgender Fall ist denkbar:

Bspl.:	<i>Vorher</i>		<i>Nachher</i>	
	Dezimalstellen	= 1	Dezimalstellen	= 3
	„Grenzwert Alarmkontakt 1“	= 500,8	„Grenzwert Alarmkontakt 1“	= 9,999

Die umgerechneten Parameter werden möglichst dicht an den ursprünglichen Parametern dargestellt. Aus diesem Grund ist es zwingend erforderlich, nach Verstellen der Dezimalstellen die gesamte Konfiguration des Messumformers zu überprüfen.

5.2. Übertragung von Tabellen

Bei einigen Messumformern ist es möglich, Tabellen Werte zu übergeben. Ein Beispiel ist hier der Kennlinienkonverter PMT50 / PMT50Ex. Soll über MODBUS diese Tabelle programmiert oder verändert werden, muss jeder Parameter einzeln übergeben werden.

Bspl: Folgende XY Tabelle aus 2 Wertepaaren soll übergeben werden

		<i>X-Wert</i>	<i>Y-Wert</i>
Zeile 1	X_1, Y_1	3	5
Zeile 2	X_2, Y_2	4	8

Der Master muss zur Übergabe der Tabelle an den Slave nun folgende Schritte durchführen:

	Beschreibung	Register Name*	Inhalt
Telegramm 1)	Operations Code „Neue Tabelle“ übermitteln	Tabellenoperation	= 1
Telegramm 2)	Zeilen Nummer angeben	Tabelle aktuelle Zeile	= 1
Telegramm 3)	X ₁ -Wert übergeben	Tabelle X-Wert	= 3
Telegramm 4)	Y ₁ -Wert übergeben	Tabelle Y-Wert	= 5
Telegramm 5)	Zeilen Nummer angeben	Tabelle aktuelle Zeile	= 2
Telegramm 6)	X ₂ -Wert übergeben	Tabelle X-Wert	= 4
Telegramm 7)	Y ₂ -Wert übergeben	Tabelle Y-Wert	= 8
Telegramm 8)	Operations Code „Ende der Tabelle“ übermitteln	Tabellenoperation	= 3

* Register-Adresse siehe gerätespezifische Parametertabelle

Folgende Operations Codes werden verwendet:

Code	Beschreibung
0	Default
1	Neue Tabelle
3	Ende der Übertragung, letzter Wert wurde übermittelt, Tabelle prüfen
6	Wertepaar ersetzen

Der Master kann außerdem den Status einer Tabelle abfragen (Register Name Tabellenstatus):

Status-Code	Beschreibung
0	Keine Tabelle vorhanden
1	Gültige Tabelle vorhanden
4	Zu wenige Stellen, alte Werte sind gültig
5	Zu viele Stellen
7	Übergebene Wertepaare sind nicht gültig, die vorherigen sind weiterhin gültig
26	Die Tabelle wird verarbeitet
27	Die Tabelle wird geprüft

6. Anhang

6.1. ASCII Tabelle

Die ASCII-Tabelle stellt die Verbindung zwischen ASCII-Zeichen und Hexadezimal-Wert dar. Üblicherweise findet man die Zeichen der ASCII-Tabelle auf der Tastatur eines PC wieder. Eine Ausnahme stellen Sonderzeichen und Steuerzeichen dar. Diese Zeichen kann man mit einer Tastenkombination eingeben. Hierzu betätigt man die <ALT>-Taste und gibt dann den Dezimal-Wert des ASCII-Zeichens ein.

Die verwendeten Zeichen im ASCII-Mode:

ASCII Zeichen	Hexadezimal Code	Dezimal Code
---------------	------------------	--------------

CR Carriage return	0D	13
LF Line feed	0A	10
:	3A	58

A	41	65
B	42	66
C	43	67
D	44	68
E	45	69
F	46	70

ASCII Zeichen	Hexadezimal Code	Dezimal Code
---------------	------------------	--------------

0	30	48
1	31	49
2	32	50
3	33	51
4	34	52
5	35	53
6	36	54
7	37	55
8	38	56
9	39	57

6.2. Parametertabellen

Die Parametertabellen sind den gerätespezifischen Tabellen zu entnehmen.
Sie werden mit den BUS-Geräten ausgeliefert und stehen auch im Internet zur Verfügung.

MODBUS Parametertabellen für DMS50 und DMS50 Ex

Parametertabelle Haltereister

Register Adresse	Protokoll Adresse	Rubrik	Parameter Name	Datentyp	Default	Wertebereich
40001	0	Sprache	Sprache	short r/w	0	0: Deutsch, 1: Englisch
40002	1	Anzeige	Kontrast	short r/w	35	0 ... 100 [%]
40003	2		Dezimalstellen	short r/w	0	0 ... 3
40004	3		Einheit Anzeige	short r/w	0	0: kg 1: t 2: N 3: kN 4: Nm 5: bar
40005	4	Eingang	Eingangsfiler	short r/w	0	0: aus 1: 0,1 s 2: 1 s 3: 5 s 4: 10 s 5: 20 s 6: 40 s
40006	5		Nennlast Messbrücke	float r/w	1000	1...9999
40008	7		Brückenspeisung	short r/w	1	0: 2,5 V 1: 5 V 2: 10 V (nicht vorhanden bei DMS50 Ex)
40009	8		Brückenempfindlichkeit	float r/w	2,000	DMS50: 0,100 ... 5,000 [mV /V] DMS50 Ex: 0,500 ... 5,000 [mV /V]
40011	10		Tara (Offset der Brückendiagonalspannung)	float r/w	0	-50,000 ... 50,000 [mV]
40013	12	Ausgang	Analogausgang Einstellung	short r/w	1	0: 0-20 mA / 0-10 V 1: 4-20 mA / 2-10 V
40014	13		Analogausgang Anfangswert	float r/w	0,0	-9999 ... 9999
40016	15		Analogausgang Endwert	float r/w	1000	-9999 ... 9999
40018	17		Alarmausgang 1 Einstellung	short r/w	0	0: aus 1: min 2: max
40019	18		Alarmausgang 1 Schaltpunkt	float r/w	1000	-9999 ... 9999
40021	20		Alarmausgang 1 Hysterese	float r/w	10	1 ... 9999
40023	22		Alarmausgang 1 Schaltverzögerung	short r/w	0	0 ... 32400 [s]
40024	23		Alarmausgang 1 Rückfallverzögerung	short r/w	0	0 ... 32400 [s]
40025	24		Alarmausgang 2 Einstellung	short r/w	0	0: aus 1: min 2: max
40026	25		Alarmausgang 2 Schaltpunkt	float r/w	1000	-9999 ... 9999
40028	27		Alarmausgang 2 Hysterese	float r/w	10	1 ... 9999
40030	29		Alarmausgang 2 Schaltverzögerung	short r/w	0	0 ... 32400 [s]
40031	30		Alarmausgang 2 Rückfallverzögerung	short r/w	0	0 ... 32400 [s]
40032	31		Simulationswert	float r/w	0	-9999 ... 9999
40034	33		Korrektur Analogausgang Startwert	float r/w	0,0	-2,000 ... +2,000 [mA]
40036	35		Korrektur Analogausgang Endwert	float r/w	0,0	-2,000 ... +2,000 [mA]
40038	37	Sperre	Parametersperre Tastenfeld	short r/w	0	0: aus 1: Parametersperre Konfigurationsebene 2: Parametersperre allgemein
40038	38	Code	Code	short r/w	0	0: default, Simulation aus 1: LCD Hintergrundbeleuchtung (2 Minuten an) 2: Min/Max Speicher löschen 3: Simulation an, Simulationswert Register-Adresse 40032 10: Tara speichern
40040	39	Bus	Adresse	short r/w	1	1 ... 247
40041	40		Protokoll	short r/w	1	0: ASCII 1: RTU
40042	41		Wortreihenfolge	short r/w	1	0: Intel 1: Modbus
40043	42		Baudrate	short r/w	6	0: 300 Baud 1: 600 Baud 2: 1200 Baud 3: 2400 Baud 4: 4800 Baud 5: 9600 Baud 6: 19200 Baud 7: 38400 Baud
40044	43		Parität	short r/w	2	0: keine 1: ungerade 2: gerade

Istwerttabelle

Register Adresse	Protokoll Adresse	Rubrik	Parameter Name	Datentyp	Wertebereich
30001	0	Messwert	Eingangsmesswert	float r	-9999 ... 9999
30003	2		Messwert Signal Brückendiagonalspannung	float r	-99,99 ... 99,99 [mV]
30005	4		Minimaler Eingangsmesswert	float r	-9999 ... 9999
30007	6		Maximaler Eingangsmesswert	float r	-9999 ... 9999

Hinweise r/w read/write
 r read

08/14-00

Headquarter

GHM Messtechnik GmbH
GHM GROUP CORPORATE
Tenter Weg 2-8
42897 Remscheid | GERMANY
Phone +49 2191 9672-0
info@ghm-group.de
www.ghm-group.de

Centers of Competences

GHM Messtechnik GmbH
GHM GROUP – Greisinger
Hans-Sachs-Straße 26
93128 Regensburg | GERMANY
Phone +49 9402 9383-0
info@greisinger.de | www.greisinger.de

GHM Messtechnik GmbH
GHM GROUP – Honsberg
Tenter Weg 2-8
42897 Remscheid | GERMANY

GHM Messtechnik GmbH
GHM GROUP – Martens
Kiebitzhörn 18
22885 Barsbüttel | GERMANY

GHM Messtechnik GmbH
GHM GROUP – Imtron
Carl-Benz-Straße 11
88696 Owingen | GERMANY

Delta OHM S.r.l. a socio unico
GHM GROUP – Delta OHM
Via Marconi 5
35030 Caselle di Selvazzano
Padova (PD) | ITALY
Phone +39 049 8977150
info@deltaohm.com
www.deltaohm.com

Valco srl
GHM GROUP – VAL.CO
Via Rovereto 9/11
20014 S. Ilario di Nerviano
Milano (MI) | ITALY
Phone +39 0331 53 59 20
valco@valco.it
www.valco.it

GHM GROUP International

Austria
GHM Messtechnik GmbH
Office Austria
Breitenseer Str. 76/1/36
1140 Vienna | AUSTRIA
Phone +43 660 7335603
a.froestl@ghm-messtechnik.de

Brazil & Latin America
GHM Messtechnik do Brasil Ltda
Av. José de Souza Campos, 1073, cj 06
Campinas, SP
13025 320 | BRAZIL
Phone +55 19 3304 3408
info@grupoghm.com.br

Czech Republic/Slovakia
GHM Greisinger s.r.o.
Ovci hájek 2/2153
158 00 Prague 5
Nove Butovice | CZECH REPUBLIC
Phone +420 251 613828
Fax +420 251 612607
info@greisinger.cz | www.greisinger.cz

Denmark
GHM Maaletchnik ApS
Maarslet Byvej 2
8320 Maarslet | DENMARK
Phone +45 646492-00
Fax +45 646492-01
info@ghm.dk | www.ghm.dk

France
GHM GROUP France SAS
Parc des Pivoles
9 Rue de Catalogne
69150 Décines (Lyon) | FRANCE
Phone +33 6 60 32 06 35
a.jouanillou@ghm-group.fr

India
GHM Messtechnik India Pvt Ltd.
209 | Udyog Bhavan | Sonowala Road
Gurgaon (E) | Mumbai - 400 063
INDIA
Phone +91 22 40236235
info@ghmgroup.in | www.ghmgroup.in

Italy for Greisinger & Delta OHM
GHM GROUP – Delta OHM
Via Marconi 5
35030 Caselle di Selvazzano
Padova (PD) | ITALY
Phone +39 049 8977150
a.casati@ghm-messtechnik.de

Italy for Honsberg, Martens, Valco
GHM GROUP – Valco
Via Rovereto 9/11
20014 S. Ilario di Nerviano
Milano (MI) | ITALY
Phone +39 0331 53 59 20
alessandro.perego@valco.it

Netherlands
GHM Meettechniek BV
Zeeltweg 30
3755 KA Eemnes | NETHERLANDS
Phone +31 35 53805-40
Fax +31 35 53805-41
info@ghm-nl.com | www.ghm-nl.com

South Africa
GHM Messtechnik SA (Pty) Ltd
16 Olivier Street
Verwoerdpark, Alberton 1453
SOUTH AFRICA
Phone +27 74 4590040
j.grobler@ghm-sa.co.za

**...and more than
100 qualified distributors!**

www.ghm-group.de

Bildnachweis: Fotolia, Getty Images, iStock, Thinkstock