

# Digitale Schnittstellen und Bussysteme

Grundlagen und praktische Hinweise  
zur Anbindung von Feldgeräten

Manfred Schleicher

**PROFIBUS-DP**

**HART®**

**Modbus**

**CANopen**

**ETHERNET**



# Digitale Schnittstellen und Bussysteme

*Grundlagen und praktische Hinweise  
zur Anbindung von Feldgeräten  
an Modbus, PROFIBUS-DP, ETHERNET,  
CANopen und HART®*

*Manfred Schleicher*

## Bemerkung

Diese Broschüre wurde nach bestem Wissen und Gewissen erstellt. Für mögliche Irrtümer übernehmen wir keine Gewähr. Maßgebend sind in jedem Fall die Betriebsanleitungen zu den entsprechenden Geräten.

## **Vorwort**

Seit vielen Jahren bietet JUMO Komponenten mit der Möglichkeit, diese an Bussysteme anzuschließen.

Mit diesem Buch vermitteln wir wichtige Grundlagen zur Anbindung. Bei der Erstellung haben wir uns immer gefragt, welche Informationen der Anwender hierzu benötigt und uns auf das Wesentliche beschränkt:

Nach einer allgemeinen Information in **Kapitel 1** schaffen **Kapitel 2 bis 4** einige Grundlagen auf dem Gebiet der digitalen Kommunikation.

Vor allem in **Kapitel 5** geben wir konkrete Hinweise zur Anbindung. Wir verstehen die Beschreibung jedoch als Wegweiser und „roten Faden“. Für weitere Informationen verweisen wir an dieser Stelle auf die entsprechenden Betriebsanleitungen.

**Kapitel 6** zeigt prinzipiell, wie eine Modem-Verbindung aufgebaut wird.

Diese Broschüre wurde nach bestem Wissen und Gewissen erstellt. Für mögliche Irrtümer übernehmen wir jedoch keine Gewähr.

Ein besonderer Dank gilt allen Kollegen, die durch ihre Mitarbeit eine Hilfe bei der Überarbeitung dieses Buches waren.

Fulda, im Juni 2008

Manfred Schleicher



### **JUMO GmbH & Co. KG**

Moritz-Juchheim-Straße 1

36039 Fulda, Germany

Telefon: +49 661 6003-396

Telefax: +49 661 6003-500

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

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

Nachdruck mit Quellennachweis gestattet!

Teilenummer: 00339287

Buchnummer: FAS 603

Druckdatum: 06.08

ISBN: 978-3-935742-02-3

---

<b>1</b>	<b>Allgemeine Information</b>	<b>5</b>
<b>2</b>	<b>Signalarten und Datencodierung</b>	<b>7</b>
2.1	Analoge und digitale Signale	7
2.2	Datenformate und Codierung	9
2.3	Hexadezimalsystem	12
<b>3</b>	<b>Serielle Schnittstellen RS232, RS422, RS485 und USB</b>	<b>13</b>
3.1	Serielle und parallele Schnittstellen	13
3.2	RS-232-, RS-422- und RS-485-Schnittstellen	14
3.2.1	Die RS-232-Schnittstelle	14
3.2.2	Die RS-422-Schnittstelle	17
3.2.3	Die RS-485-Schnittstelle	19
3.2.4	Die Anschlussleitung für die RS-485-/RS-422-Schnittstelle	20
3.2.5	Verdrahtung von Teilnehmern mit RS-485-Schnittstelle	20
3.2.6	Schnittstellenumsetzer	21
3.2.7	Konfiguration von Schnittstellen, der UART	22
3.3	USB	25
<b>4</b>	<b>Zugriffsverfahren und Übertragungsmedien</b>	<b>27</b>
4.1	Zugriffsverfahren	27
4.1.1	Master-/Slave-Verfahren	27
4.1.2	Token-/Passing-Verfahren	28
4.1.3	CSMA-Verfahren	28
4.2	Übertragungsmedien in der Automatisierungstechnik	29
4.3	Der Abschlusswiderstand	30
<b>5</b>	<b>Bussysteme</b>	<b>31</b>
5.1	Modbus	31
5.1.1	Der Protokollaufbau	32
5.2	PROFIBUS-DP	42
5.3	ETHERNET	49
5.3.1	TCP/IP	50
5.3.2	IP - Internet-Protokoll	50
5.3.3	Vergabe von IP-Adressen	52
5.3.4	Anbindung eines Teilnehmers an ETHERNET/Aufbau eines A-Klasse-Netzes	53
5.3.6	COM-Umlenkung	58
5.4	CAN und CANopen	65
5.4.1	CAN	65
5.4.2	CANopen	67

# Inhalt

---

<b>5.5</b>	<b>HART®</b> .....	<b>74</b>
5.5.1	HART®-Protokoll .....	74
5.5.2	FSK-Verfahren .....	74
5.5.3	Multidropbetrieb .....	75
5.5.4	JUMO-Geräte mit HART® .....	75
5.5.5	Kommunikation über PC .....	75
5.5.6	HHT .....	76
5.5.7	Tastatur/LCD .....	76
<b>6</b>	<b>Modems</b> .....	<b>77</b>
<b>6.1</b>	<b>Kommandos für das Modem</b> .....	<b>78</b>

## Allgemeine Information

Einleitend wird ein Überblick bezüglich der Gerätearten von JUMO gegeben, welche über Bussysteme kommunizieren:

**Temperaturmessumformer** bereiten die Signale von Widerstandsthermometern/Thermoelementen auf und stellen diese als Einheitssignale zur Verfügung.

Die Einheit Temperaturfühler - Messumformer liefert beispielsweise ein Signal 4 ... 20mA, was 0 ... 500 °C entspricht.



**Abbildung 1: JUMO dTRANS T01 HART®**

Verfügt der Temperaturmessumformer über eine HART®-Schnittstelle, erfolgt über diese meist ausschließlich die Konfiguration des Gerätes.

**Druckmessumformer** liefern proportional zum Druck Einheitssignale (meist 4 ... 20mA). Die Umformer sind ebenfalls in einigen Ausführungen mit HART®-Schnittstelle lieferbar.



**Abbildung 2: JUMO dTRANS p02**

**Messumformer für Leitfähigkeit und pH-Wert** stellen zusammen mit der Sensorik die genannten Größen als Einheitssignale zur Verfügung. Die Prozessgrößen werden in zahlreichen Anwendungen über Bussysteme an Visualisierungssysteme (Modbus) oder Speicherprogrammierbare Steuerungen (PROFIBUS-DP) übertragen.

# 1 Allgemeine Information



Abbildung 3: JUMO dTRANS pH

Die Grundfunktion eines **Kompaktreglers** ist die exakte Regelung einer Prozessgröße. Auch in diesem Fall erfolgt die Übertragung von Prozesswerten über Bussysteme (PROFIBUS-DP und Modbus).



Abbildung 4: Universeller Programmregler/-geber JUMO IMAGO 500

Die Aufzeichnung von Prozessgrößen ist mit **Bildschirmschreibern** möglich. Die Messwerte werden in einem internen Speicher des Gerätes abgelegt. Die Abholung der aufgezeichneten Daten erfolgt im einfachsten Fall über eine externe Compact-Flash Card oder zeitweise mit einem USB-Stick. Alternativ kann der interne Speicher automatisiert von einem PC über Bussysteme ausgelesen werden. Eine Anbindung erfolgt häufig über ETHERNET aber auch PROFIBUS-DP und Modbus sind möglich.



Abbildung 5: Bildschirmschreiber JUMO LOGOSCREEN nt

# 2 Signalarten und Datencodierung

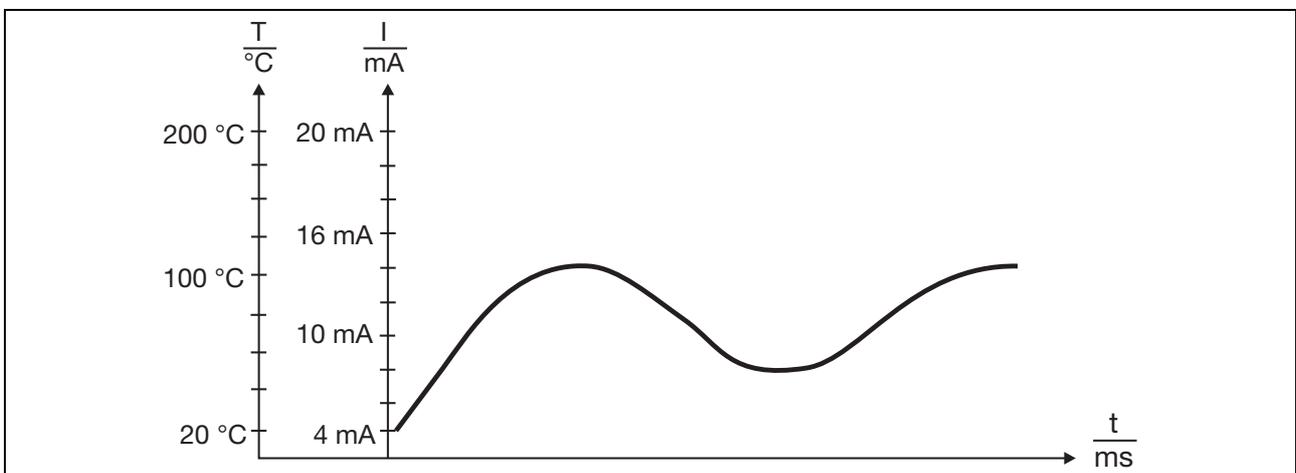
## Signalarten und Datencodierung

### 2.1 Analoge und digitale Signale

Produzenten von Komponenten der Automatisierungstechnik liefern kaum noch Geräte mit einer analogen Signalverarbeitung. Es werden fast ausschließlich mikroprozessorgesteuerte Geräte angeboten. Neben anderen Vorteilen, wie einer möglichen Programmierung, kann mit diesen Geräten eine digitale Schnittstelle realisiert werden. Über eine Schnittstelle können die Prozessgrößen gesendet werden.

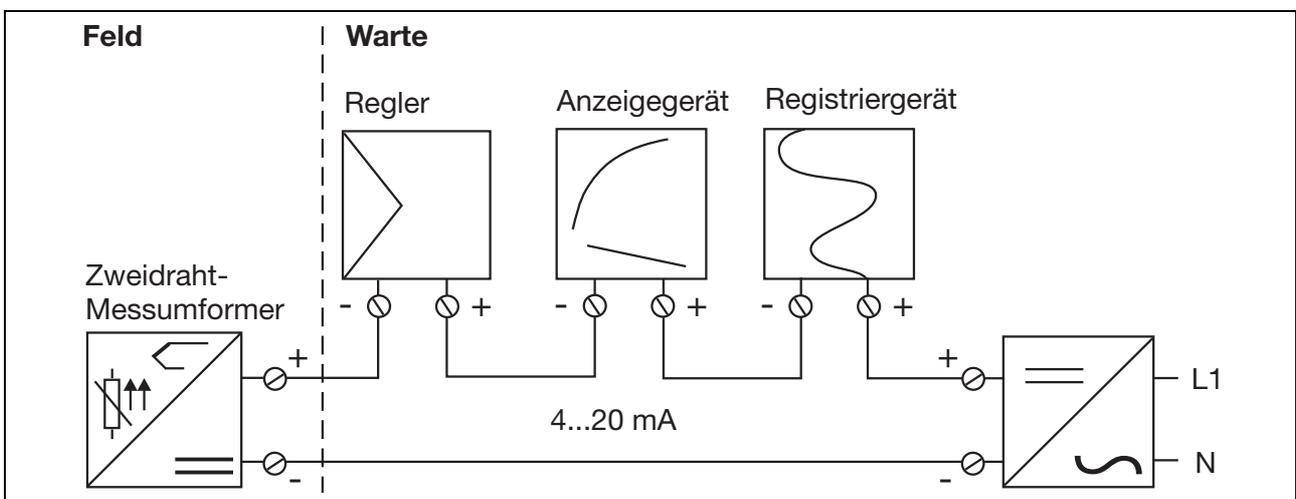
Wir schauen uns das Prinzip von analogen und digitalen Signalen an.

Ein Messwert, z. B. eine Temperatur, wird von einem Sensor mit Messumformer in ein dieser Temperatur proportionales Signal umgewandelt. Das Signal kann beispielsweise ein Strom 4 ... 20mA sein. Jedem Wert der Temperatur entspricht eine eindeutige Signalthöhe (z. B. 20 ... 200°C/ 4 ... 20mA). Ändert sich die Temperatur, so ändert sich in gleichem Maße das Signal (Abbildung 6).



**Abbildung 6:** Analoges Signal mit einer sich kontinuierlich ändernden Amplitude

Das Einheitssignal (in der Abbildung 4 ... 20mA - weiterhin sind die Signale 0 ... 20mA, 2 ... 10V und 0 ... 10V üblich) kann über die unterschiedlichsten Geräte geführt werden. Jede Messwertänderung wird über den Strom sofort durch alle angeschlossenen Geräte registriert.



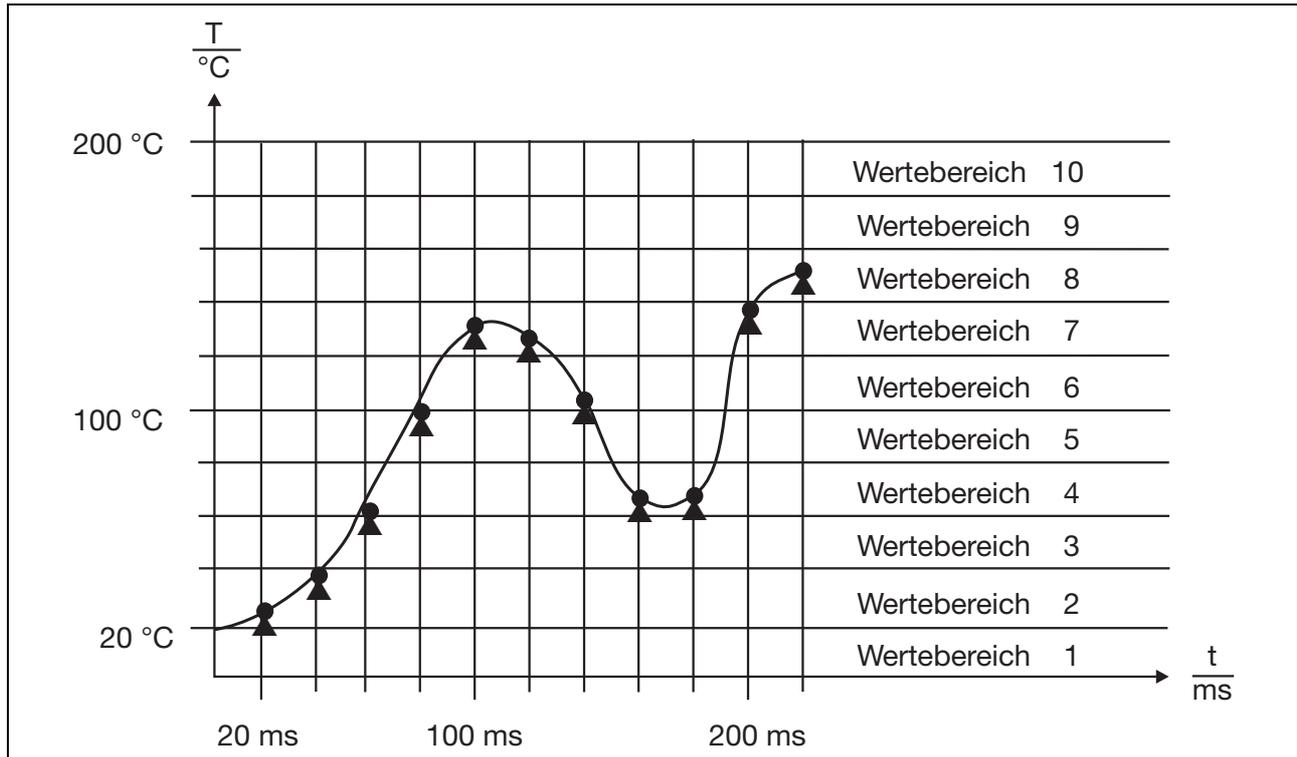
**Abbildung 7:** Analoge Signalübertragung im Fall eines Stromsignales

## 2 Signalarten und Datencodierung

### Bemerkung:

Die gezeigte Signalübertragung erfolgt analog, dennoch arbeiten Messumformer und die angeschlossenen Komponenten in den meisten Fällen auf der Basis eines Mikroprozessors (der Zweidrahtmessumformer erfasst das Signal am Sensor analog, es erfolgt eine Wandlung in ein digitales Signal mit anschließender Verarbeitung; vor der Ausgabe geschieht die Umwandlung in ein analoges Signal).

Findet eine digitale Signalverarbeitung statt, muss der Analogwert in Wertebereiche eingeteilt werden. Die Werte werden innerhalb einer festgelegten Zeit, der Abtastzeit, abgefragt. Die Aufgabe der Signalumwandlung übernimmt ein Analog-/Digitalwandler (A/D-Wandler).



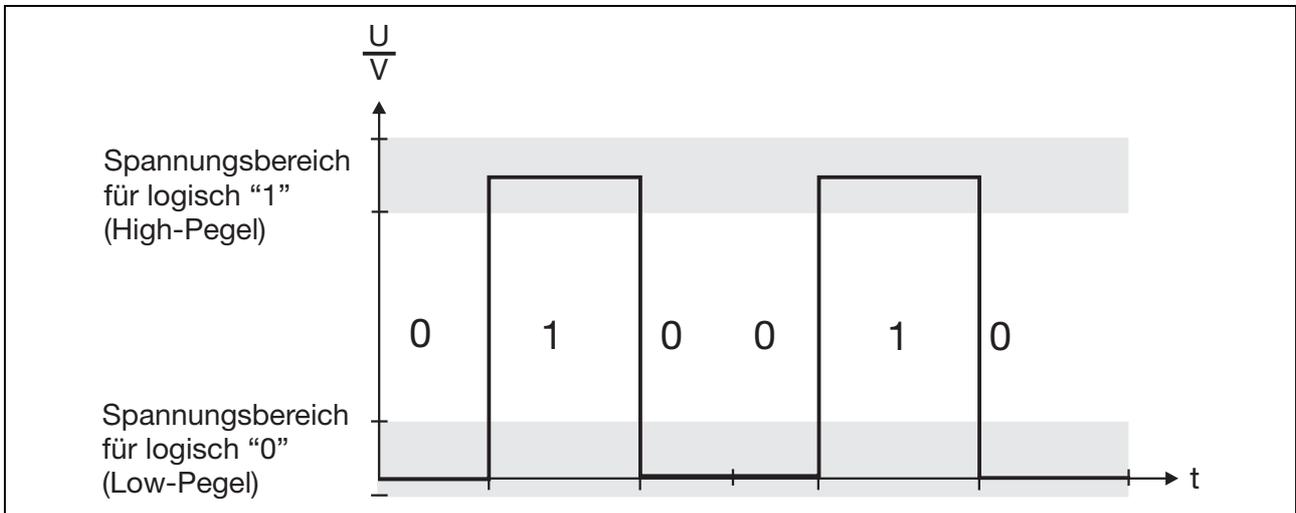
**Abbildung 8: Digitalisiertes Messsignal**

Abbildung 8 zeigt 10 Wertebereiche und eine Abtastzeit von 20ms. Das digitalisierte Signal wäre sehr ungenau. JUMO arbeitet zum Zeitpunkt des Druckes u. a. mit A/D-Wandlern, welche eine 16-Bit-Auflösung besitzen. Es existieren somit  $2^{16} = 65536$  Wertebereiche, was verdeutlicht, dass hier ein sehr geringer Fehler durch die Einteilung in Wertebereiche auftritt. Übliche Abtastzeiten liegen von 50 ... 500ms.

# 2 Signalarten und Datencodierung

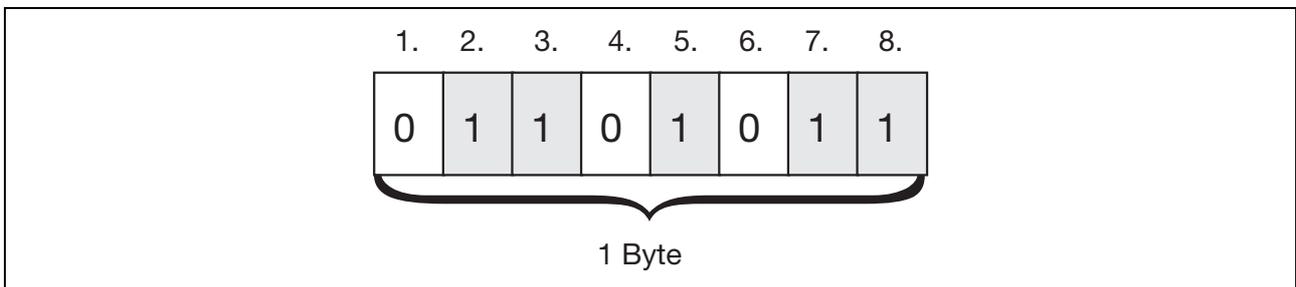
## 2.2 Datenformate und Codierung

Über eine Schnittstelle können nur zwei Zustände ausgegeben werden (Low- oder High-Pegel). Diese kleinste Informationseinheit wird Bit genannt. Die verwendeten Spannungspegel (bei Low oder High) sind von der Schnittstelle abhängig.



**Abbildung 9: Binäre Datenübertragung in „Bits“ durch verschiedene Spannungspegel**

Prinzipiell sendet eine Komponente der Automatisierungstechnik nie ein Bit alleine, es werden immer mindestens acht Bit in einem Block versendet. Hierbei handelt es sich um ein Byte.



**Abbildung 10: Zusammenfassen von „8 Bit“ zu „1 Byte“**

In Form eines Bytes können z. B. die Stellungen von acht schaltenden Ausgängen eines Reglers übertragen werden.

Weiterhin werden mit einem Byte Zeichen gesendet. Hier bedient man sich beispielsweise des ASCII-Codes. Zum Ermitteln des entsprechenden Wertes muss das Byte als binäre Zahl gedeutet werden:

Beispiel: es wurde das Byte  $00100110_2$  gesendet.

Der binäre Wert muss nun in eine Dezimalzahl gewandelt werden. Jeder Ziffer der binären Zahl ist eine Zweierpotenz zugeordnet:

	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$							
Wertigkeit der Stelle	128	64	32	16	8	4	2	1							
Mit dem oben genannten Byte	0	0	1	0	0	1	1	0							
wird der Zahlenwert	0	+	0	+	32	+	0	+	0	+	4	+	2	+	0
gesendet, dies entspricht:	$38_{10}$														

## 2 Signalarten und Datencodierung

Codiert der Sender den Zahlenwert als ASCII-Wert, kann aus folgender Tabelle das entsprechende Zeichen ermittelt werden:

ASCII-Wert	Zeichen	Steuerzeichen	ASCII-Wert	Zeichen	ASCII-Wert	Zeichen	ASCII-Wert	Zeichen
000	leer	NUL	032	(Leerstelle)	064	@	096	
001	☺	SOH	033	!	065	A	097	a
002	☹	STX	034	„	066	B	098	b
003	♥	ETX	035	#	067	C	099	c
004	♦	EOT	036	\$	068	d	100	d
005	♣	ENQ	037	%	069	E	101	e
006	♠	ACK	038	&	070	F	102	f
007	(Alarm)	BEL	039	´	071	G	103	g
008	☒	BS	040	(	072	H	104	h
009	(Tabulator)	HT	041	)	073	I	105	i
010	(Zeilenvorschub)	LF	042	*	074	J	106	j
011	(Grundstellung)	VT	043	+	075	K	107	k
012	(Seitenvorschub)	FF	044	,	076	L	108	l
013	(Schreibkopfrücklauf)	CR	045	-	077	M	109	m
014	♪	SO	046	.	078	N	110	n
015	☼	SI	047	/	079	O	111	o
016	▶	DLE	048	0	080	P	112	p
017	◀	DC1	049	1	081	Q	113	q
018	↕	DC2	050	2	082	R	114	r
019	!!	DC3	051	3	083	S	115	s
020	¶	DC4	052	4	084	T	116	t
021	§	NAK	053	5	085	U	117	u
022	–	SYN	054	6	086	V	118	v
023	↕	ETB	055	7	087	W	119	w
024	↑	CAN	056	8	088	X	120	x
025	↓	EM	057	9	089	Y	121	y
026	→	SUB	058	:	090	Z	122	z
027	←	ESC	059	;	091	[	123	{
028	(Positionsanzeiger nach rechts)	FS	060	<	092	\	124	
029	(Positionsanzeiger nach links)	GS	061	=	093	]	125	}
030	(Positionsanzeiger nach oben)	RS	062	>	094	↑	126	~
031	(Positionsanzeiger nach unten)	US	063	?	095	–	127	△

**Tabelle 1: ASCII-Code**

## 2 Signalarten und Datencodierung

Aus Tabelle 1 kann entnommen werden, dass das Zeichen „&“ gesendet wurde. Die in der Tabelle gezeigten 128 Zeichen können mit sieben Bits codiert werden. In einem Byte stehen jedoch acht Bits zur Verfügung. Weiterhin sind Umlaute und viele Sonderzeichen nicht enthalten. Deshalb erweiterte man den Code auf acht Bits und kann so insgesamt  $2^8 = 256$  Zeichen codieren.

Die mit dem ASCII-Wert ab 128 codierten Zeichen (erweiterter ASCII-Code) sind in der Tabelle nicht angegeben.

Neben ASCII existieren noch andere Codes, z. B. von der amerikanischen Normengesellschaft ANSI (**A**merican **N**ational **S**tandard **I**nstitute). Windows verwendet diesen ANSI-Zeichensatz, der für die Nummern 32 bis 127 mit dem ASCII-Zeichensatz übereinstimmt.

Moderne Feldgeräte sind häufig zwischen verschiedenen Sprachen umschaltbar. Mit einem auf 8 Bits basierenden Code ist die Darstellung aller weltweit verfügbaren Sprachen unmöglich. Aus diesem Grund arbeiten Feldgeräte und Anwendungsprogramme von JUMO häufig mit Unicode. Mit Unicode sind prinzipiell alle weltweit verfügbaren Schriftzeichen darstellbar (zur Drucklegung dieses Buches sind bereits über 100.000 Zeichen definiert).



**Abbildung 11: Gruppenansicht des Bildschirmschreibers JUMO LOGOSCREEN nt in russischer (links) und chinesischer Sprache (rechts) (Sprachenvielfalt wird erst durch Unicode möglich)**

Gelegentlich werden auch Zahlen zu zwei Bytes codiert. Mit zwei Bytes, für die man auch den Begriff „Wort“ verwendet, werden ganzzahlige Werte aus dem Bereich 0 ... 65535 codiert. Dieses so genannte Integerformat wird z. B. für Zeiten und Zählwerte verwendet.

In der Automatisierungstechnik werden sehr häufig Messwerte übertragen (Temperaturen, Drücke etc.). Diese werden nach dem Standard IEEE 754 codiert und mit vier Byte übertragen. Als so genannter Float-Wert kann ein Zahlenbereich von  $1.0E\pm 38$  codiert werden.

## 2 Signalarten und Datencodierung

### 2.3 Hexadezimalsystem

Sehr wichtig in der Informationstechnologie ist das Hexadezimalsystem, denn häufig werden vier Bits (ein halbes Byte) zusammengefasst. Mit vier Bits können 16 unterschiedliche Zustände codiert werden. Soll die Darstellung mit einer Ziffer geschehen, ist dies nicht im Dezimalsystem möglich (größter Wert der Ziffer: 10).

Alle Zustände von vier Bits, den Wert im Dezimalsystem und Hexadezimalsystem zeigt die folgende Tabelle:

4 Bits	Wertigkeit dezimal	Wertigkeit hexadezimal
0000	$0_{10}$	$0_{16}$
0001	$1_{10}$	$1_{16}$
0010	$2_{10}$	$2_{16}$
0011	$3_{10}$	$3_{16}$
0100	$4_{10}$	$4_{16}$
0101	$5_{10}$	$5_{16}$
0110	$6_{10}$	$6_{16}$
0111	$7_{10}$	$7_{16}$
1000	$8_{10}$	$8_{16}$
1001	$9_{10}$	$9_{16}$
1010	$10_{10}$	$A_{16}$
1011	$11_{10}$	$B_{16}$
1100	$12_{10}$	$C_{16}$
1101	$13_{10}$	$D_{16}$
1110	$14_{10}$	$E_{16}$
1111	$15_{10}$	$F_{16}$

**Tabelle 2: Binär-, Dezimal- und Hexadezimalzahlen**

Der Index  $9_{10}$  bzw.  $9_{16}$  gibt an, dass es sich um eine Zahl im Dezimalsystem bzw. im Hexadezimalsystem handelt. Bei einer hexadezimalen Zahl wird auch häufig 0x vor die Zahl gestellt:

$0xA = A_{16}$  entspricht der Zahl A im Hexadezimalsystem. Diese hat den Wert 10 im Dezimalsystem ( $10_{10}$ ).

Ein Byte wird oft als zweistellige, hexadezimale Zahl dargestellt, so entspricht z. B.:

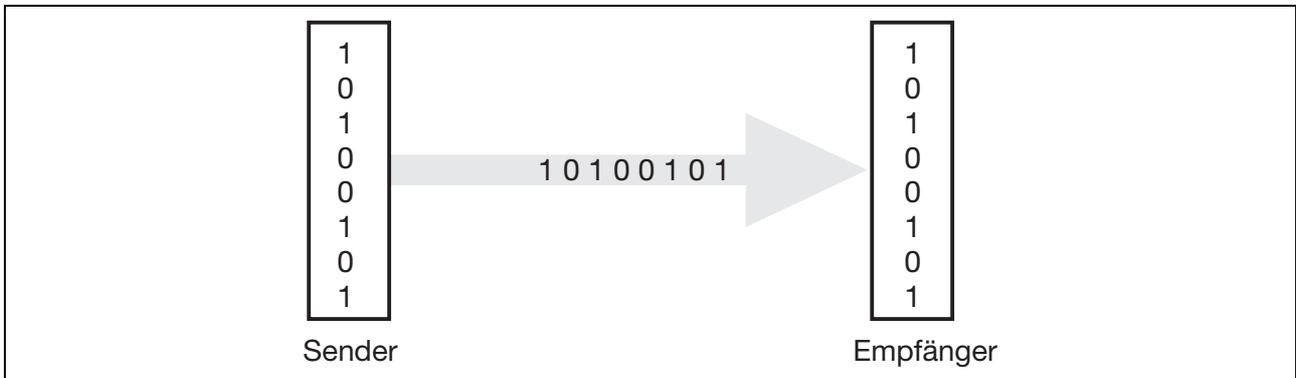
Byte: 0110 1101  
Darstellung durch zwei hexadezimale Zahlen 0x 6 D = 0x6D

# 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

## Serielle Schnittstellen RS232, RS422, RS485 und USB

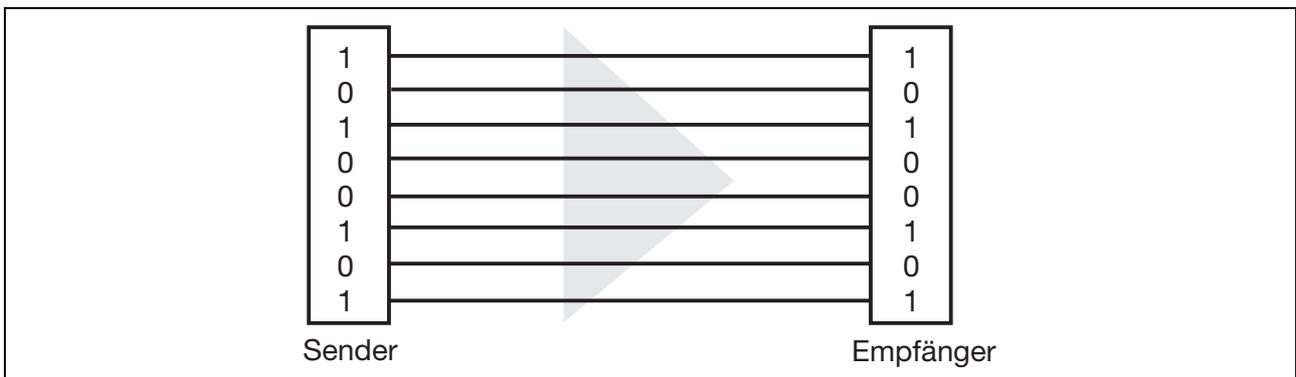
### 3.1 Serielle und parallele Schnittstellen

Im Regelfall geschieht die Datenübertragung in einem Bussystem seriell: die einzelnen Bits werden nacheinander, meist über eine Zweidrahtleitung, versendet.



**Abbildung 12: Serielle Datenübertragung**

Selten werden in einem Bussystem die Daten parallel übertragen: bei einer parallelen Datenübertragung werden immer mehrere Bits gleichzeitig versendet. Parallele Bussysteme existieren im Bereich der Messtechnik (IEC-Bus IEEE 488).



**Abbildung 13: Parallele Signalübertragung**

Der große Vorteil des parallelen Busses ist die hohe Übertragungsgeschwindigkeit, da immer mehrere Bits gleichzeitig übertragen werden. Demgegenüber steht eine hohe Störanfälligkeit und hohe Kosten bei der Verdrahtung. Geeignet ist die parallele Datenübertragung bei kurzen Wegen.

Die nachfolgend beschriebenen RS-232-, RS-422- und RS-485-Schnittstellen arbeiten allesamt seriell und werden bei JUMO für das Modbus-Protokoll genutzt.

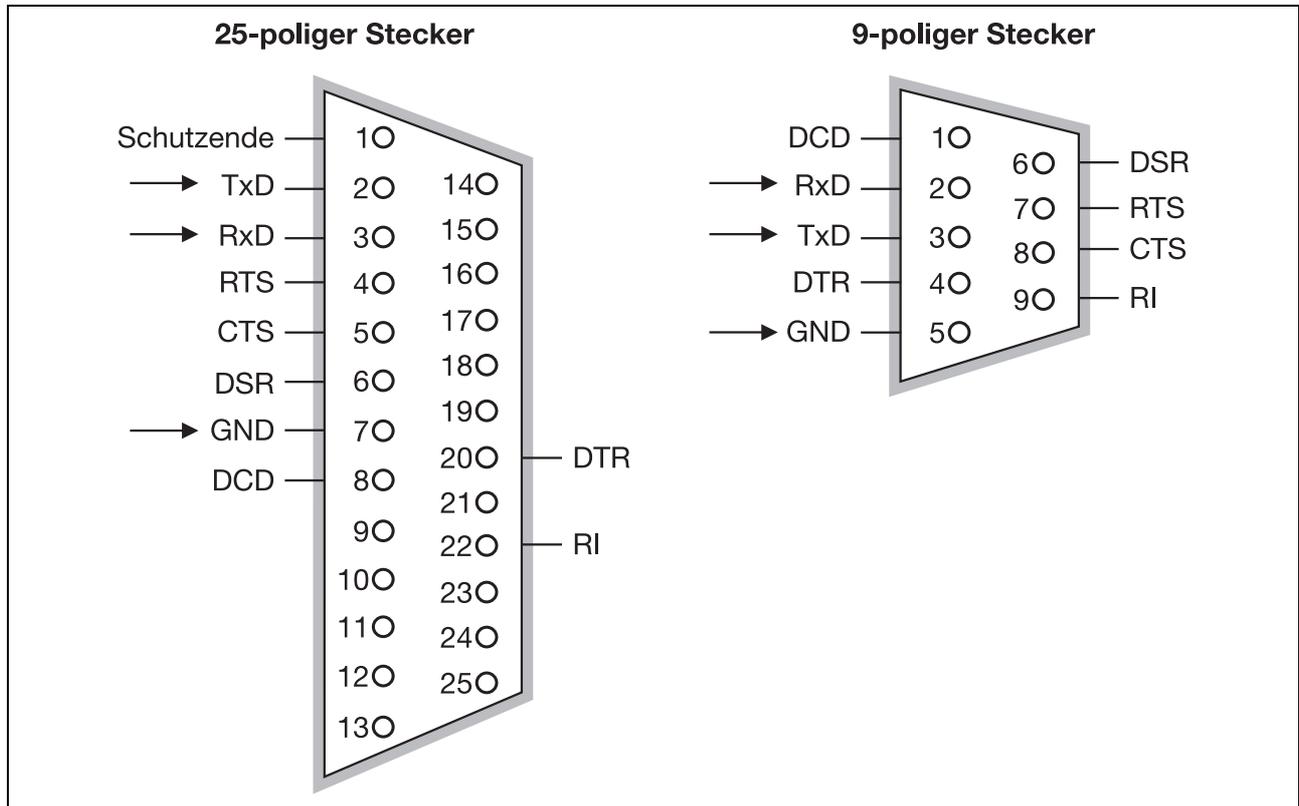
# 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

## 3.2 RS-232-, RS-422- und RS-485-Schnittstellen

### 3.2.1 Die RS-232-Schnittstelle

Eine serielle Schnittstelle befindet sich nahezu in jedem PC (die COM-Ports stellen jeweils eine RS-232-Schnittstelle dar).

Abbildung 14 zeigt die Belegung für die am meisten verwendeten Stecker (Sub-D):

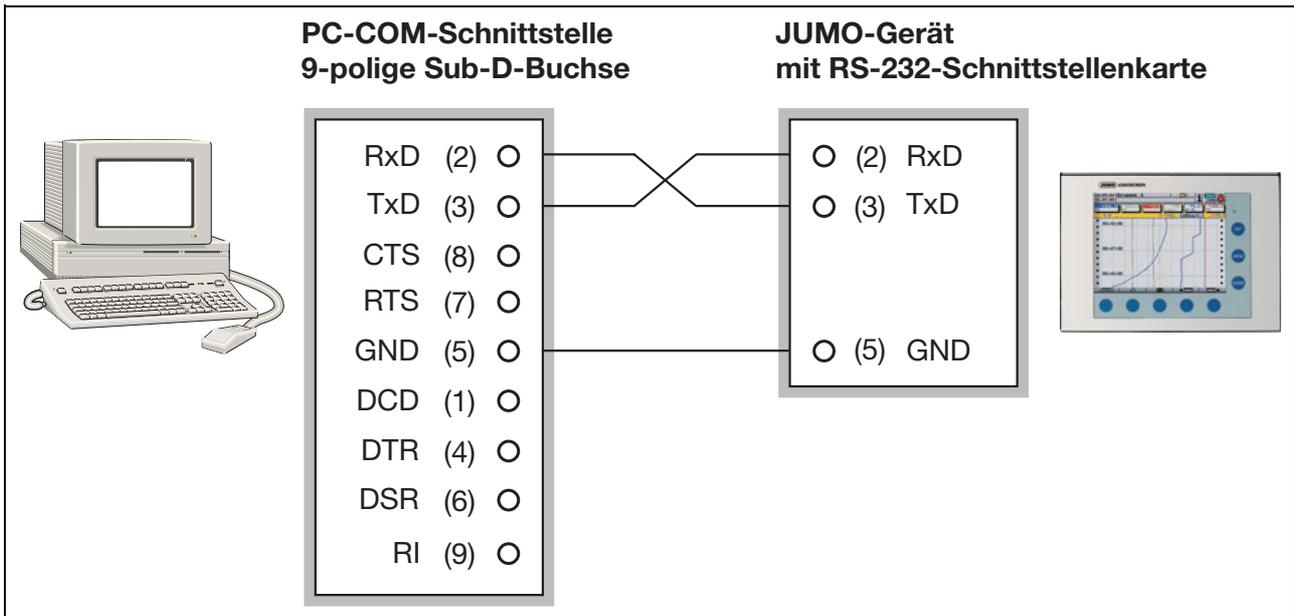


**Abbildung 14: Typische Belegung von SUB-D (25- und 9-polig) für die RS-232-Schnittstelle**

Die RS-232-Schnittstelle wurde ursprünglich entworfen, um Rechner über Telefonleitungen zu verbinden. Aus diesem Grund ergeben sich die Vielzahl der gezeigten Signale.

In der Automatisierungstechnik werden von der RS-232-Schnittstelle meist nur drei Leitungen verwendet: TxD (Transmit Data) zum Senden, RxD (Receive Data) zum Empfangen und die GND-Leitung für das gemeinsame Bezugspotenzial (ohne Anschluss der GND-Leitung keine Datenübertragung).

### 3 Serielle Schnittstellen RS232, RS422, RS485 und USB



**Abbildung 15: Verdrahtung eines PC's mit einem JUMO-Gerät über RS-232-Schnittstelle**

In Abbildung 15 fällt auf, dass TxD des PC's mit RxD des Gerätes verbunden wird. Dies ist auch verständlich, wenn man überlegt, dass der PC über die Leitung TxD sendet, das Gerät dieses Signal empfängt (RxD). Gleiches gilt für TxD (Gerät) und RxD (PC).

Der Low-Pegel (logisch 0) der RS-232-Schnittstelle liegt in einem Bereich von 3 ... 15V, der High-Pegel (logisch 1) in einem Bereich von -3 ... -15V.

Die maximale Leitungslänge der RS-232-Schnittstelle liegt bei 30m. Die RS-232-Schnittstelle ist nicht busfähig und kann nur für Punkt-zu-Punkt-Verbindungen verwendet werden (Anschluss von nur zwei Teilnehmern).

JUMO nutzt die RS-232-Schnittstelle in folgenden Fällen:

1. Geräte können mit Hilfe eines Konfigurationsprogrammes (Setup) konfiguriert werden, die Daten werden vom PC über die RS-232-Schnittstelle mit einem Setupkabel (PC-Interface) übertragen:



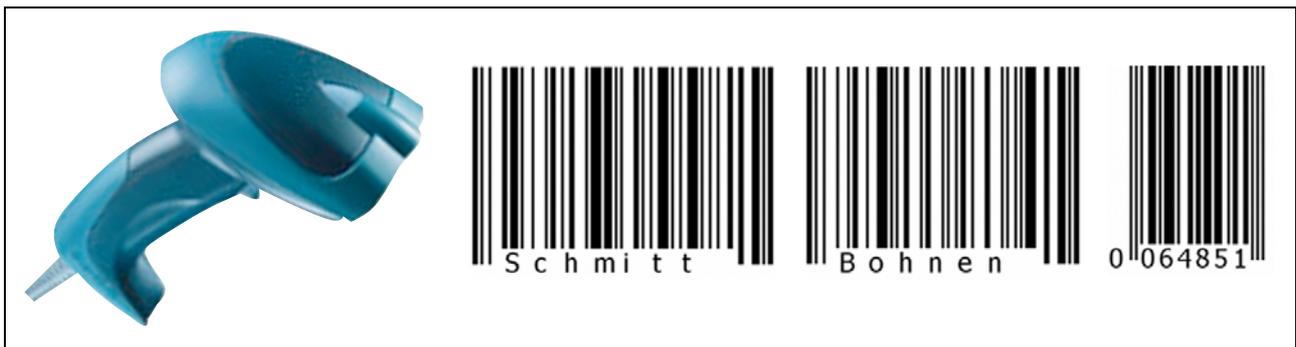
**Abbildung 16: Regler JUMO DICON 500 mit PC-Interface**

### 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

Der im vorderen Bildbereich dargestellte Stecker wird auf einen freien COM-Port des PC's gesteckt (RS-232-Schnittstelle) und wandelt den Pegel der RS-232-Schnittstelle in TTL-Logik um. Die Schnittstelle am Regler arbeitet mit TTL-Logik. Das PC-Interface ist auch für die Anbindung an USB lieferbar.

2. JUMO liefert Bildschirmschreiber, die Messdaten von den angeschlossenen Sensoren in einem internen elektronischen Speicher ablegen. Die Messdaten können zeitgesteuert über Schnittstelle auf Anfrage vom Bildschirmschreiber zum PC übertragen werden. Existiert in der Anlage nur ein Bildschirmschreiber, kann dieser über die RS-232-Schnittstelle angebunden werden.
3. Einige Bildschirmschreiber ermöglichen das Erstellen von Chargenprotokollen. Der Start des Produktionsvorganges wird dem Schreiber z. B. durch das Schließen eines binären Einganges signalisiert. Die Charge läuft und die Eingabe von Chargendaten kann über die Gerätefront erfolgen (beispielsweise Bediener, Produkt und Produktnummer). Das Ende des Produktionsvorganges wird durch Öffnen des erwähnten Binäreinganges signalisiert. Aufgrund der Chargendaten kann später in der Auswertesoftware in einfacher Weise nach dem Chargenprotokoll gesucht und die Messwerte während des Produktionsvorganges eingesehen werden.

Alternativ zur Eingabe der Chargendaten am Gerät kann diese über einen Barcodeleser erfolgen. Das Lesegerät wird an die **RS232** des Bildschirmschreibers angeschlossen und nach einer Konfigurationsänderung am Gerät werden die so genannten Chargendaten in einfacher Weise eingescannt:

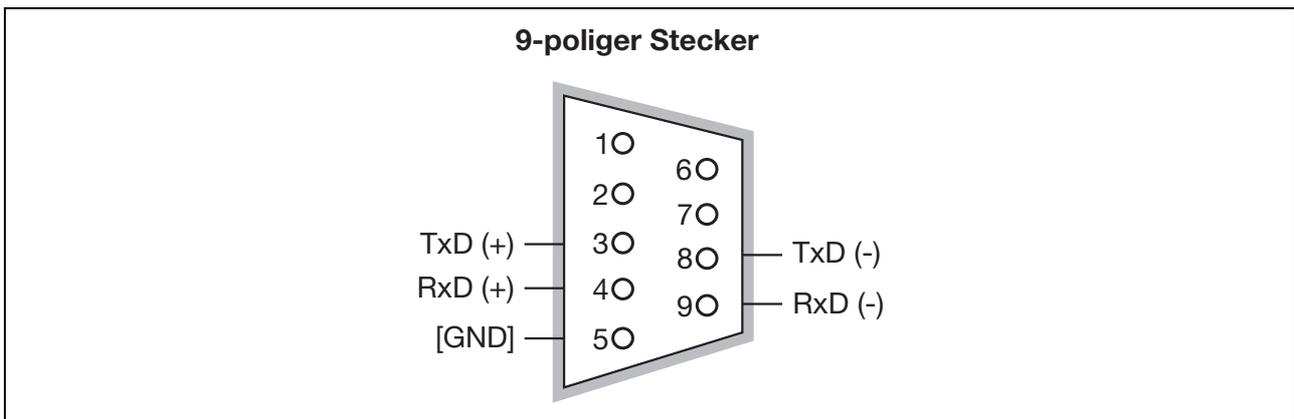


**Abbildung 17: Einlesen von Chargendaten mit Barcodeleser**

Die RS-232-Schnittstelle ist im Vergleich zu den folgenden RS-422- und RS-485-Schnittstellen störanfällig. Den Grund hierfür beschreiben wir in den folgenden Zeilen:

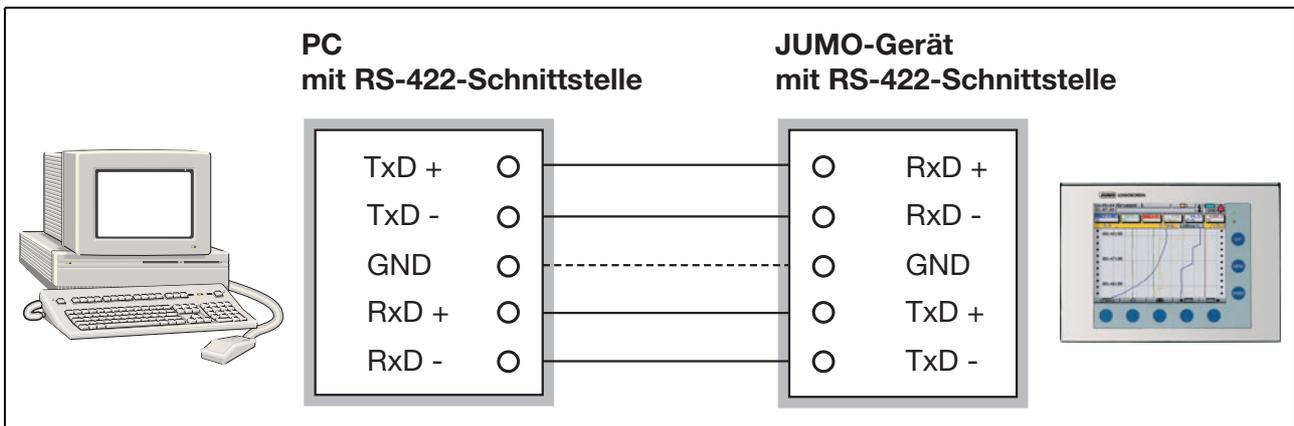
# 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

## 3.2.2 Die RS-422-Schnittstelle



**Abbildung 18:** Typische Belegung eines 9-poligen Sub-D-Steckers bei einer RS-422-Schnittstelle

Die Datenübertragung mit der RS-422-Schnittstelle erfolgt über eine Vierdrahtleitung:



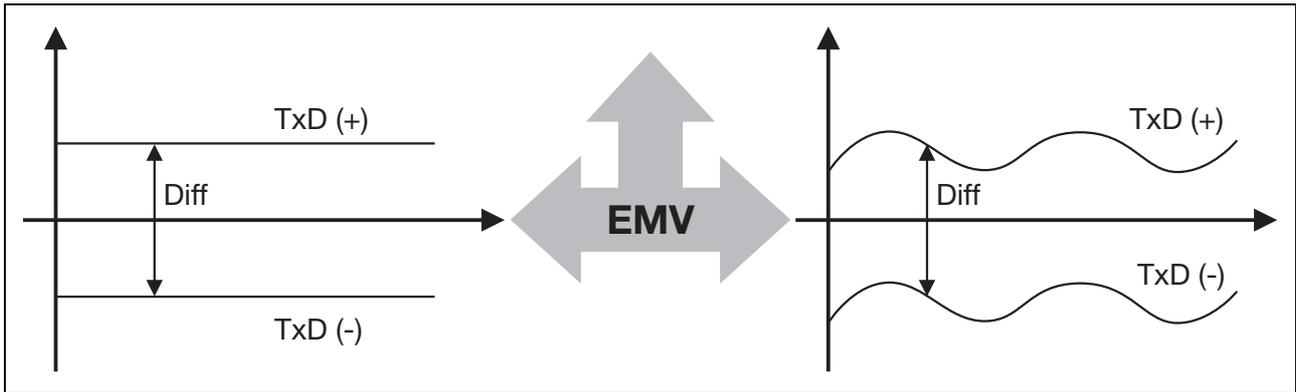
**Abbildung 19:** Datenübertragung mit der RS-422-Schnittstelle

Alle Teilnehmer senden über die Datenleitungen TxD + und TxD - und empfangen über RxD + und RxD -. Die GND-Leitung ist nur bei großen Potenzialdifferenzen zu verwenden und sollte nicht notwendig sein.

Die RS-422-Schnittstelle (auch die RS-485-Schnittstelle) ist unempfindlicher gegenüber Störungen als die RS-232-Schnittstelle. Der Grund liegt darin, dass bei den Leitungspegeln immer die Differenz betrachtet wird: liegt z. B. an TxD + +5V und TxD - -5V, entspricht dies dem Pegel logisch „0“.

Liegt entsprechend an TxD + -5V und TxD - +5V an, entspricht dies dem Pegel logisch „1“. Eine elektromagnetische Störung würde das Potenzial an beiden Leitungen im gleichen Maß erhöhen. Somit hätte die Störung keinen Einfluss auf das Differenzsignal (zum Vergleich: bei der RS-232-Schnittstelle hätte die Störung nur Einfluss auf die Signalleitungen, nicht aber auf das Bezugspotenzial).

### 3 Serielle Schnittstellen RS232, RS422, RS485 und USB



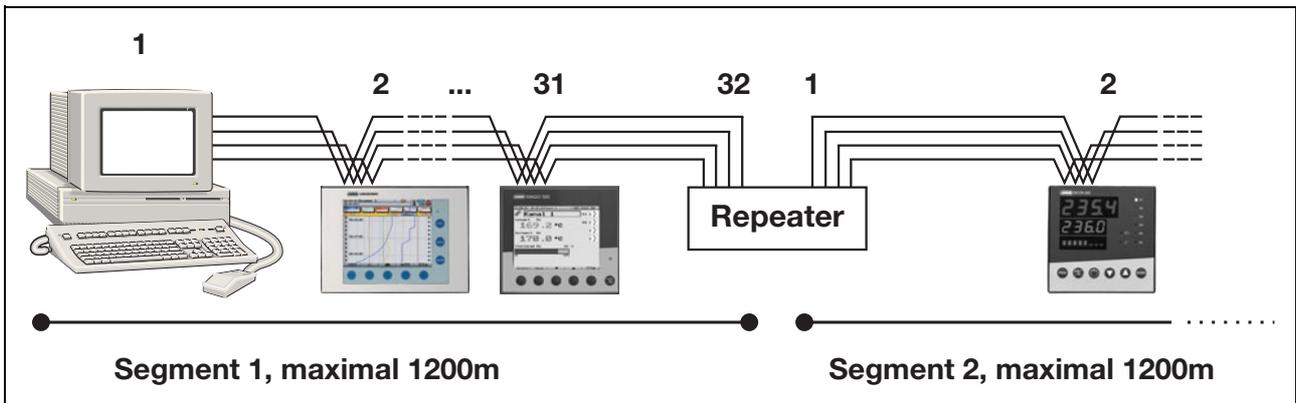
**Abbildung 20: Störungsunempfindliche RS-422-Schnittstelle**

Die RS-422-Schnittstelle ist weiterhin busfähig. Es können 32 Teilnehmer miteinander verbunden werden.

Wie sind nun weitere Teilnehmer an die RS-422-Schnittstelle anzuschließen?

Am Feldgerät (Abbildung 19) sind alle Klemmen (TxD +, TxD -, RxD + und RxD -) mit den gleichen Klemmen des nächsten Gerätes zu verbinden.

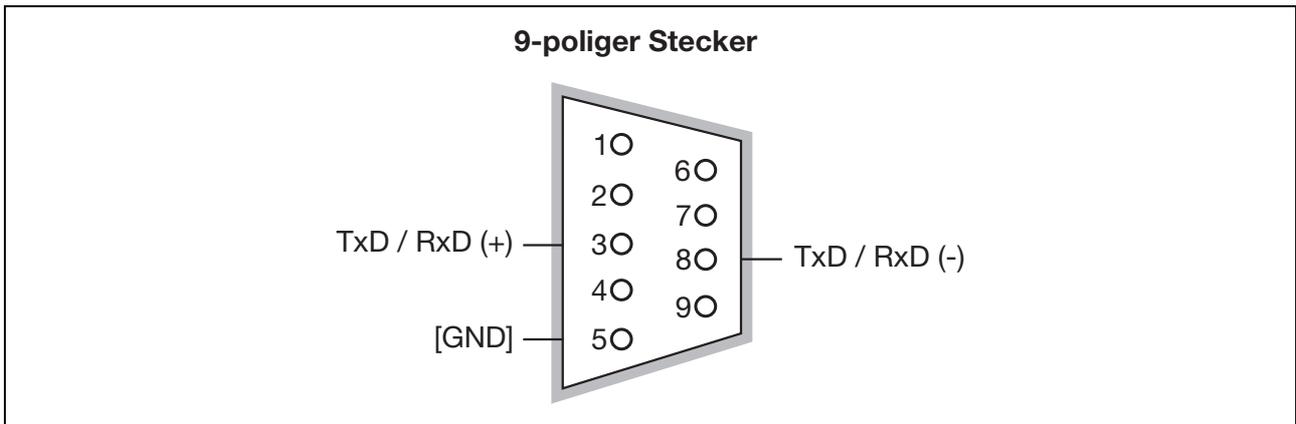
Der Abstand vom ersten bis zum letzten Teilnehmer kann 1200m betragen. Sollten mehr als 32 Teilnehmer vorhanden sein oder sich die Teilnehmer mehr als 1200m voneinander entfernt befinden, wird mit Repeatern (Leistungsverstärkern) gearbeitet (Abbildung 21). Die Repeater zählen in beiden Segmenten als Gerät.



**Abbildung 21: Einsatz eines Repeaters bei einer RS-422-Schnittstelle**

# 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

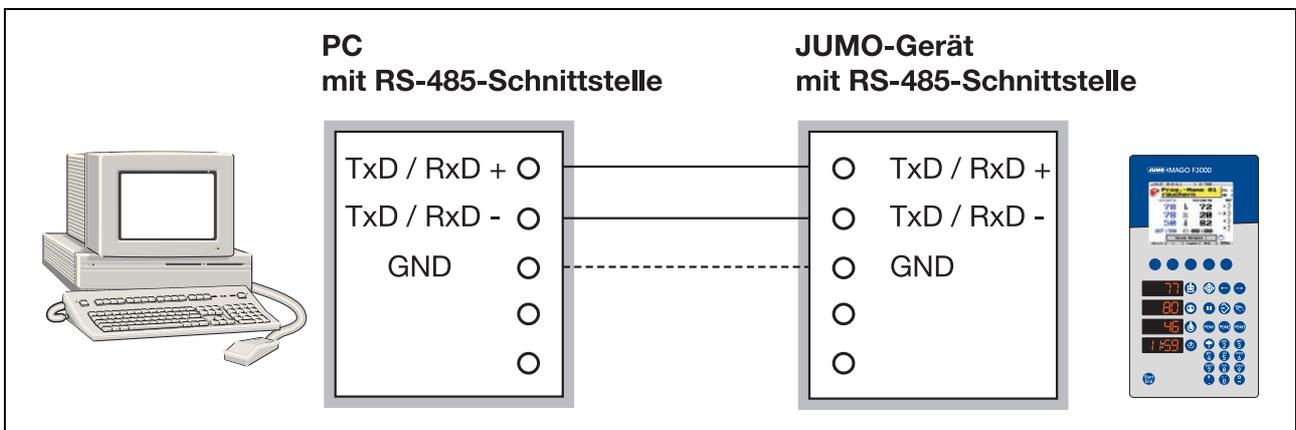
## 3.2.3 Die RS-485-Schnittstelle



**Abbildung 22: Typische Belegung eines 9-poligen Sub-D-Steckers bei einer RS-485-Schnittstelle**

Die RS-485-Schnittstelle hat ähnliche Eigenschaften wie die RS-422-Schnittstelle. Die Pegel liegen in der gleichen Höhe. Es können Ausdehnungen von bis zu 1200m erreicht und 32 Teilnehmer miteinander verbunden werden. Auch bezüglich der Repeater, der Störungsunempfindlichkeit und der Spannungspegel gilt das über die Schnittstelle RS422 Beschriebene.

Der große Unterschied zur RS-422-Schnittstelle und gleichzeitig der Grund für eine viel stärkere Verbreitung ist, dass die Teilnehmer bei der RS-485-Schnittstelle lediglich über zwei Datenleitungen miteinander verbunden werden.



**Abbildung 23: Datenübertragung mit der RS-485-Schnittstelle**

Die Teilnehmer senden und empfangen über die Leitungen TxD/RxD + und TxD/RxD -. Somit ergibt sich ein sehr geringer Installationsaufwand. Bei der RS-485-Schnittstelle kommt es bei den Teilnehmern zum Umschalten: so sendet z. B. der PC Daten über das Leitungspaar zum Gerät. Zu dieser Zeit nutzt das Gerät das Leitungspaar zum Empfangen. Ist die Nachricht übertragen, wird im Gerät von Empfangen auf Senden, beim PC von Senden auf Empfangen umgeschaltet.

# 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

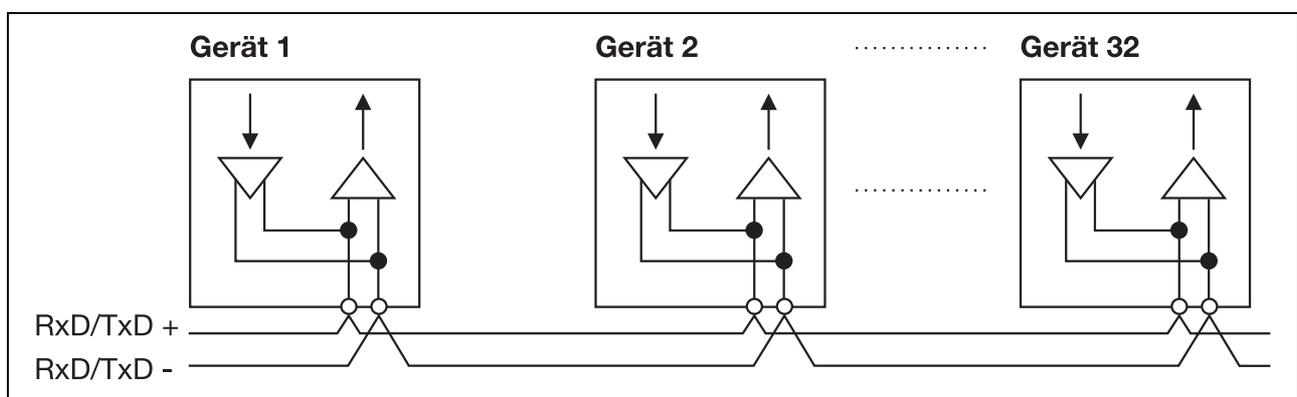
## 3.2.4 Die Anschlussleitung für die RS-485-/RS-422-Schnittstelle

Kapitel 4 „Zugriffverfahren und Übertragungsmedien“ beschreibt die Anschlusskabel detailliert. Im Bereich der RS-485-/RS-422-Schnittstelle kommt eine paarweise verdrehte, abgeschirmte Anschlussleitung zum Einsatz. Im Leitungspaar sind die RxD/TxD + und RxD/TxD - der RS-485- bzw. bei der RS-422-Schnittstelle erstes Leitungspaar RxD + und RxD - und zweites Leitungspaar TxD + und TxD - zusammengefasst.

Warum eine paarweise verdrehte Leitung eingesetzt wird, erörtert Kapitel 4 „Zugriffverfahren und Übertragungsmedien“.

## 3.2.5 Verdrahtung von Teilnehmern mit RS-485-Schnittstelle

Bei der RS-422- oder RS-485-Schnittstelle werden die Teilnehmer in Linien- bzw. Bustopologien angeordnet.



**Abbildung 24: Linientopologie mit RS-485-Schnittstelle**

Für die RS-485-Schnittstelle bedeutet dies: die Leitung RxD/TxD + wird vom ersten Teilnehmer auf den zweiten geführt, vom zweiten auf den dritten etc. Entsprechend wird die RxD/TxD - vom ersten Teilnehmer auf den zweiten geführt, vom zweiten auf den dritten etc.

Bei der RS-422-Schnittstelle werden die anderen Anschlussklemmen ebenfalls in gleicher Weise verbunden (Abbildung 21).

Auf Stichleitungen ist unbedingt zu verzichten. Die ankommende und abgehende Leitung ist bis zum Feldgerät zu führen (kein Einsatz von Verteilerdosen).

Bei EMV-gerechter Verdrahtung (separater Potenzialausgleich zu jedem Schaltschrank bzw. Feldgerät geführt) erzielt man meist die besten Ergebnisse, wenn an jedem Gerät bzw. Schaltschrank der Schirm der Busleitung großflächig auf PA aufgelegt ist.

Am ersten und letzten Teilnehmer ist ein Abschlusswiderstand zu aktivieren. Kommen Feldgeräte von JUMO zum Einsatz, wird auf den Abschlusswiderstand verzichtet, da dieser bereits in die Schnittstelle integriert ist.

# 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

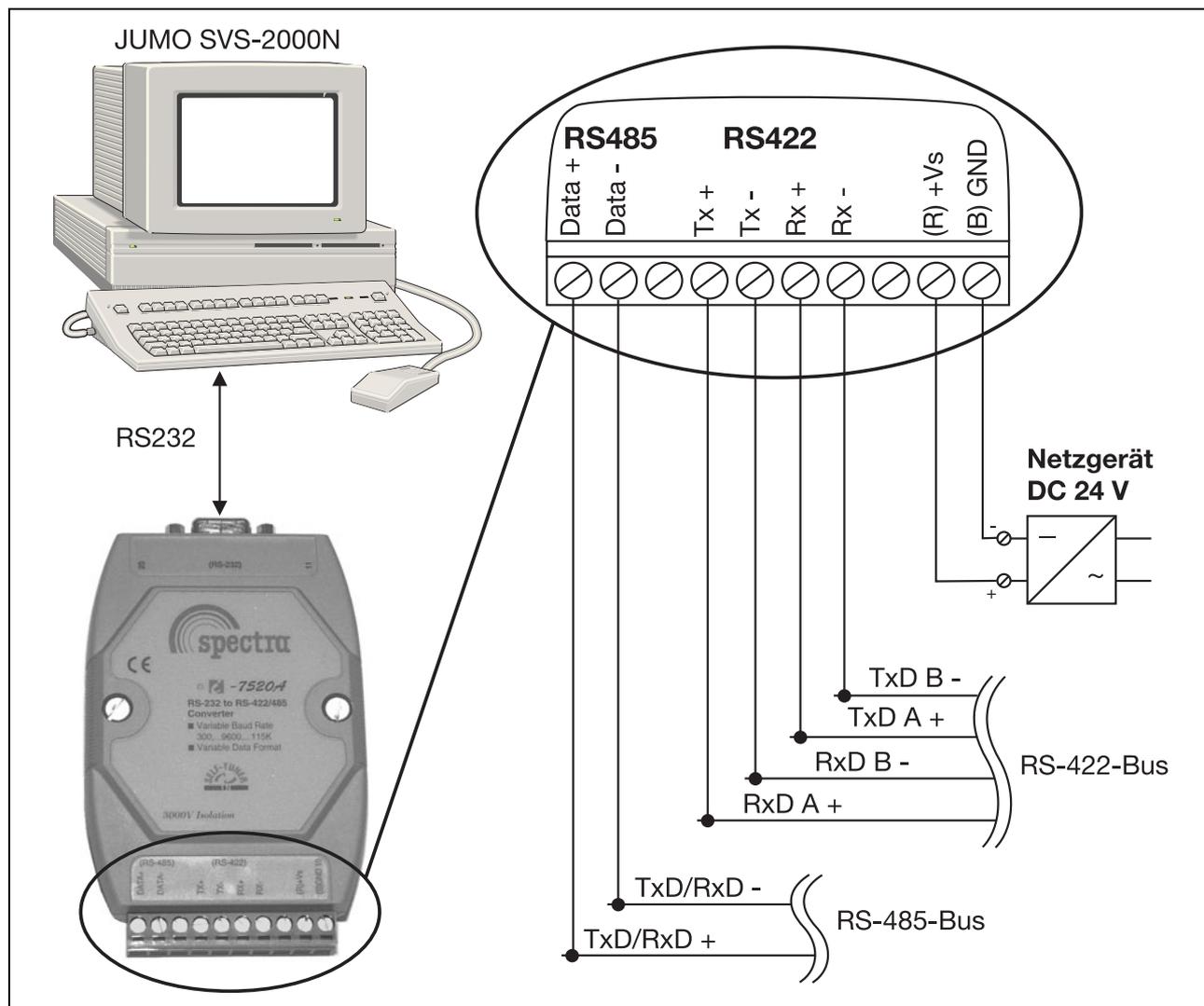
## 3.2.6 Schnittstellenumsetzer

Die meisten PC's werden mit einer RS-232-Schnittstelle geliefert (COM-Port).

Möchte man Feldgeräte mit RS-485-Schnittstellen anbinden, hat man prinzipiell zwei Möglichkeiten:

1. Einsatz eines Schnittstellenumsetzers (RS232 - RS485/RS422)

An den COM-Port wird ein Schnittstellenumsetzer angeschlossen. Er wandelt die Pegel der RS-232- auf die Pegel der RS-485- bzw. RS-422-Schnittstelle.



**Abbildung 25: Schnittstellenumsetzer RS-232- auf RS-422- und RS-485-Schnittstelle**

Abbildung 25 zeigt einen Schnittstellenumsetzer der Firma Spectra. Der Anschluss des Schnittstellenumsetzers ist einfach: die Leitung für die RS-232-Schnittstelle gehört zum Lieferumfang des Umsetzers. Nun können wahlweise die Feldgeräte mit RS-485-Schnittstelle (RxD/TxD + an Data + und RxD/TxD - an Data -) oder entsprechend Feldgeräte an die RS-422-Schnittstelle angeschlossen werden. Am Schnittstellenumsetzer müssen keine Einstellungen vorgenommen werden. Bei Einsatz eines Schnittstellenumsetzers ist wichtig, dass dieser die Datenflussrichtung automatisch erkennt.

2. Einsatz einer Karte mit RS-485-Schnittstelle

Der Markt bietet Karten für PC's mit RS-485- bzw. RS-422-Schnittstellen. Sind diese eingebaut, können die Feldgeräte direkt an den PC angeschlossen werden.

# 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

## 3.2.7 Konfiguration von Schnittstellen, der UART

Die Datenübertragung zwischen den Bauteilen eines Feldgerätes geschieht parallel. Die Daten in einem Bussystem werden seriell übertragen. Der UART ist in einem Feldgerät für die Umwandlung der parallelen in serielle Datenübertragung verantwortlich.

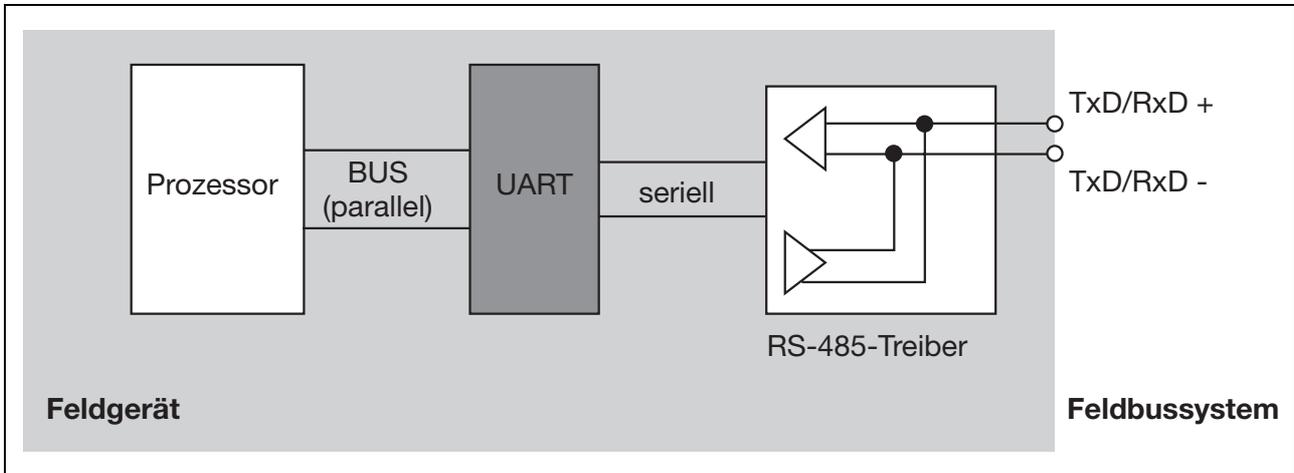


Abbildung 26: UART im Fall einer RS-485-Schnittstelle

**Die UART's der an den Bus angeschlossenen Feldgeräte müssen alle mit der gleichen Geschwindigkeit arbeiten.** Die Geschwindigkeit der Datenübertragung ist definiert in Baud. Als Baud bezeichnet man die Anzahl der Zustandsänderungen pro Sekunde. Da in einem seriellen Bussystem pro Zustandsänderung ein Bit übertragen wird, entspricht die Baudrate der übertragenen Bits pro Sekunde (Bps).

Jeder Teilnehmer muss eine Adresse erhalten, jede Adresse darf nur einmal im System vorhanden sein.

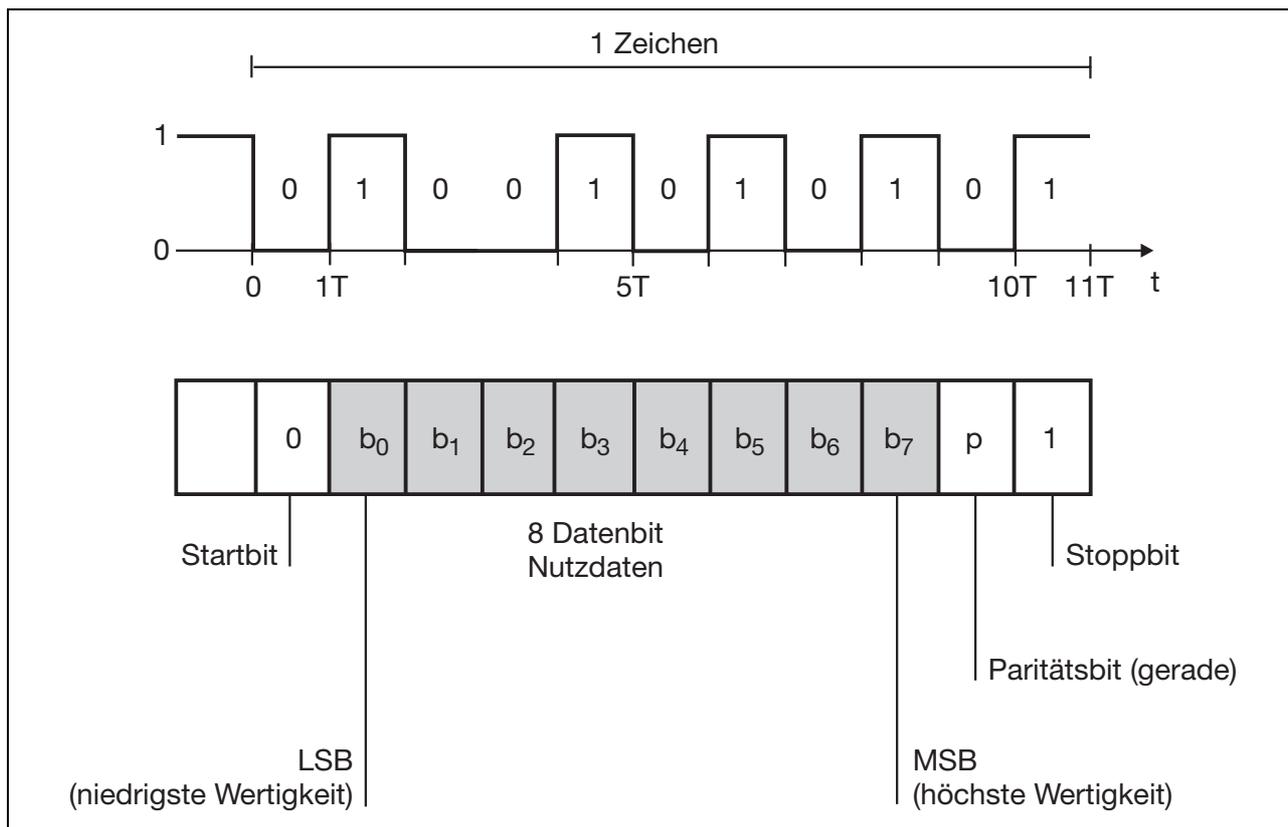
Bei den meisten Bussystemen muss keine weitere Konfiguration des UART's vorgenommen werden.

### Weitere Konfiguration des UART's:

Bei dem später vorgestellten Modbus-Protokoll erfolgen weitere Einstellungen. Der UART sendet die Informationen byteweise: auch wenn ein Messwert (bestehend aus einem Doppelwort oder vier Bytes) gesendet wird, erfolgt die Datenübertragung Byte für Byte.

Schauen wir uns diese Byte's genauer an:

### 3 Serielle Schnittstellen RS232, RS422, RS485 und USB



**Abbildung 27: Übertragung eines Bytes**

Wie beschrieben, muss an allen Teilnehmern die gleiche Baudrate eingestellt sein. Die Teilnehmer arbeiten jedoch nicht unbedingt synchron. Daher wird vor den Nutzdaten ein Startbit zur Synchronisation gesendet (Abbildung 27).

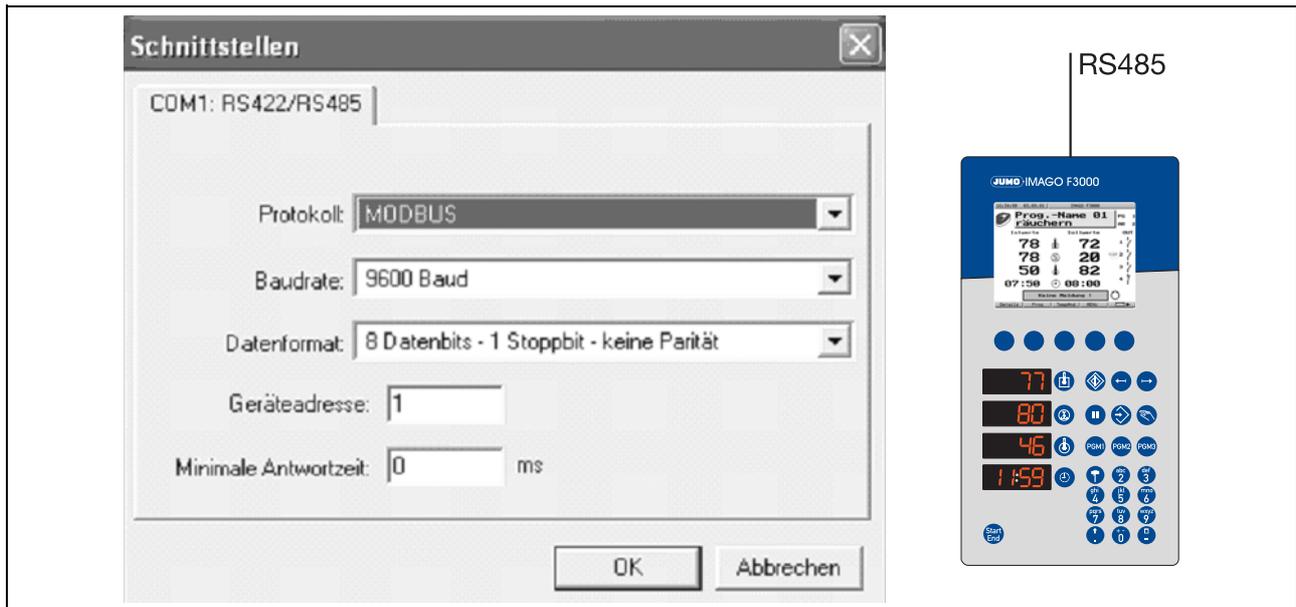
Danach folgt ein Byte Nutzdaten.

Nun folgt das Paritätsbit. Es kann z. B. auf ungerade eingestellt werden: die in dem Paket gesendeten Bits mit der Wertigkeit 1 werden auf eine ungerade Anzahl ergänzt. In diesem Fall muss beim Empfänger ebenfalls „ungerade“ Parität eingestellt werden: nur wenn der Empfänger eine ungerade Anzahl von Bits mit der Wertigkeit 1 empfängt, wird die Information übernommen, anderenfalls wird sie verworfen. Eine fehlerhafte Datenübertragung eines Bits würde abgefangen. Die Paritätsprüfung könnte weiterhin auf „gerade“ (wie in Abbildung 27) gestellt werden. Nun würde auf eine gerade Anzahl von Einsen ergänzt. Wird die Einstellung „keine“ verwendet, erfolgt keine Paritätsprüfung. Wie wir später sehen werden, erfolgen übergeordnet bedeutend sicherere Prüfungen auf eine fehlerfreie Übertragung. Gleich welche Einstellung vorgenommen wird (ungerade, gerade, keine): die Einstellung muss bei allen Teilnehmern gleich sein.

Mit dem Stoppbit erfolgt die Information, dass die Nachricht nun übertragen ist. Es können ebenfalls unterschiedliche Einstellungen (1, 2 Stoppbits etc.) vorgenommen werden.

Auch hier gilt: die Einstellungen sind für alle Teilnehmer gleich vorzunehmen.

### 3 Serielle Schnittstellen RS232, RS422, RS485 und USB



**Abbildung 28: Konfiguration der Schnittstelle (Modbus über RS-485-Schnittstelle) des JUMO IMAGO F3000**

**Bemerkung:**

Fragt im Fall einer RS-485-Schnittstelle ein Master (z. B. ein PC) bei einem Feldgerät einen Wert an, muss beim Master von Senden auf Empfangen umgeschaltet werden. Mit dem Parameter „Minimale Antwortzeit“ (Abbildung 28) kann eine minimale Zeit eingestellt werden, die das Gerät wartet, bis es mit dem Senden beginnt.

## 3 Serielle Schnittstellen RS232, RS422, RS485 und USB

### 3.3 USB

Auch USB (**U**niversal **S**erial **B**us) ist ein serieller Bus. Die Datenübertragung erfolgt über zwei verdrehte Leitungen, zwei weitere Leitungen dienen zur Stromversorgung der angeschlossenen Geräte.

Die Bus-Spezifikation sieht einen zentralen Host (Master) vor, der die Koordination der angeschlossenen Peripherie-Geräte (den so genannten Clients) übernimmt. Daran können theoretisch bis zu 127 verschiedene Geräte angeschlossen werden. An einem USB-Port kann immer nur ein USB-Gerät angeschlossen werden. Wenn an einem Host mehrere Geräte angeschlossen werden sollen, muss deshalb ein Verteiler (Hub) für deren Kopplung sorgen. Durch den Einsatz von Hubs entstehen Baumstrukturen, die alle im Host-Controller enden.

Im Zusammenhang mit JUMO-Geräten findet die USB-Schnittstelle in folgenden Fällen Anwendung:

1. PC-Interface zur Konfiguration der Feldgeräte mit USB-Schnittstelle:



**Abbildung 29: PC-Interface mit USB-Schnittstelle**

2. Einige JUMO-Geräte verfügen über eine USB-Schnittstelle (Client). In diesem Fall können Standard-USB-Kabel für die Konfiguration verwendet werden.
3. Bildschirmschreiber (beispielsweise LOGOSCREEN nt) verfügen neben der Client-Schnittstelle über eine Host-Schnittstelle. Die Host-Schnittstelle wird für die Speicherung von aufgezeichneten Messdaten auf USB-Stick verwendet.

Allgemein werden in Richtung des Hosts flache Stecker (Typ A „DIN IEC 61 076-3-107“) verwendet:



**Abbildung 30: Stecker Typ A**

## **3 Serielle Schnittstellen RS232, RS422, RS485 und USB**

---

In Richtung des Clients werden die Kabel entweder fix montiert oder über annähernd quadratische Stecker angeschlossen, vereinzelt auch mit Typ A-Steckern.

# 4 Zugriffsverfahren und Übertragungsmedien

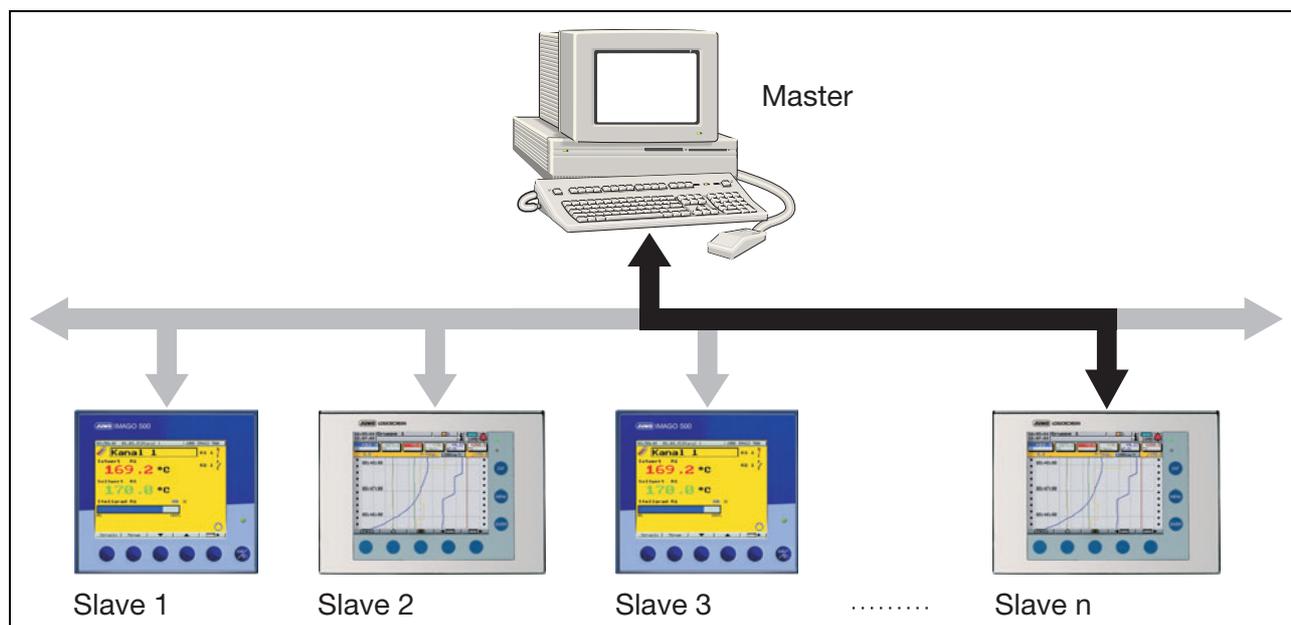
## Zugriffsverfahren und Übertragungsmedien

### 4.1 Zugriffsverfahren

In einem Bussystem sind die Teilnehmer meist in Bustopologie angeordnet. Alle Komponenten arbeiten z. B. über eine Zweidrahtleitung. Die Leitung kann jedoch zur gleichen Zeit immer nur einem Teilnehmer zur Verfügung stehen. Durch die Zugriffsverfahren ist geregelt, welcher Teilnehmer zu welcher Zeit auf den Bus zugreifen darf.

Schauen wir uns die unterschiedlichen Zugriffsverfahren an:

#### 4.1.1 Master-/Slave-Verfahren



**Abbildung 31: Master-/Slave-Verfahren**

Beim Master-/Slave-Verfahren steuert ein Master die Kommunikation auf dem Bus: er beginnt z. B. mit Anweisungen für den Slave 1. Der Slave folgt den Anweisungen und bestätigt die Ausführung (z. B. Übernahme Sollwert). Hiernach fragt der Master die ihn interessierenden Werte vom gleichen Slave ab (der Slave sendet z. B. 2 Analogwerte). In dieser Weise kommuniziert der Master mit allen angeschlossenen Slaves. Wurden die Daten mit dem letzten Slave ausgetauscht, beginnt der Master erneut mit Slave 1. Die Zeit, die für den Austausch mit allen Slaves benötigt wird, nennt man Buszykluszeit. Die Buszykluszeit kann von ms (PROFIBUS-DP) bis zu mehreren Sekunden (Modbus) betragen.

# 4 Zugriffsverfahren und Übertragungsmedien

## 4.1.2 Token-/Passing-Verfahren

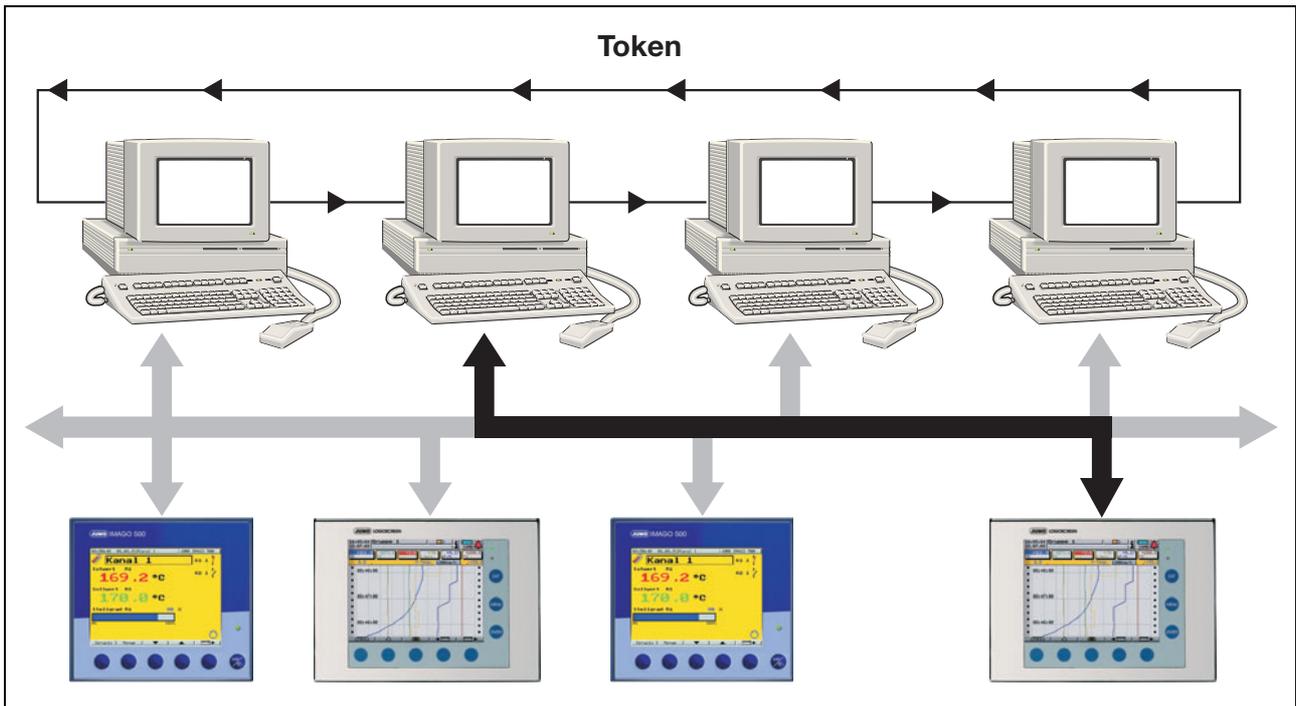


Abbildung 32: Token-/Passing-Verfahren

Auch beim Token-/Passing-Verfahren hat zur gleichen Zeit ein Master die Kontrolle über den Bus. Hier ist jedoch die Besonderheit, dass mehrere Master im System vorhanden sind. Im System existiert ein Token (Zugriffsrecht). Dieser wird von Master zu Master weitergereicht. Nur der Master, der im Besitz des Token ist, darf die Daten mit den Slaves austauschen. Das Token-/Passing-Verfahren kommt relativ selten zum Einsatz. Ein Beispiel ist PROFIBUS-DP.

## 4.1.3 CSMA-Verfahren

Arbeitet ein System nach dem CSMA(**C**arrier **S**ense **M**ultiple **A**ccess)-Verfahren, sind alle Teilnehmer auf dem Bus gleichberechtigt: jede Station kann von sich aus anfangen zu senden. Prinzipiell prüft jede Station, bevor sie zu senden anfängt, ob im Moment ein anderer Teilnehmer sendet. Problematisch wird es, wenn zwei Teilnehmer gleichzeitig anfangen zu senden. Nun gibt es unterschiedliche Möglichkeiten, wie mit dieser so genannten Kollision umgegangen wird:

### CSMA-CD

Senden beim Zugriffsverfahren CSMA-CD(**C**arrier **S**ense **M**ultiple **A**ccess **C**ollision **D**etection) zwei Teilnehmer gleichzeitig, wird die Nachricht zerstört. Die Teilnehmer erkennen das gleichzeitige Senden und wiederholen die Sendung nach dem Zufallsprinzip. Sollte es wieder zu einer Kollision kommen, erfolgt ein neues Senden usw. ETHERNET arbeitet z. B. nach diesem Verfahren.

### CSMA-CA

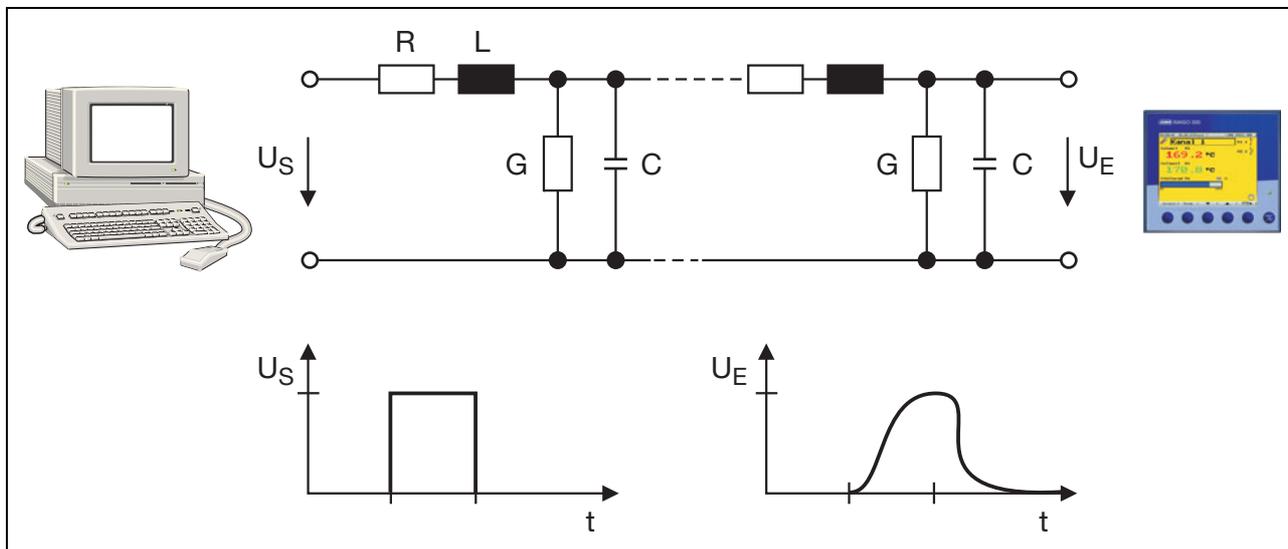
Beim CSMA-CA(**C**arrier **S**ense **M**ultiple **A**ccess **C**ollision **A**void)-Verfahren zieht sich der Teilnehmer, welcher die Nachricht mit geringerer Priorität sendet, vom Bus zurück. Durch den Protokollaufbau und die elektrischen Eigenschaften der Schnittstelle setzt sich die Nachricht mit der höheren Priorität durch und wird zerstörungsfrei übertragen.

Lesen Sie mehr zu diesem Verfahren in Kapitel 5.4 „CAN und CANopen“.

# 4 Zugriffsverfahren und Übertragungsmedien

## 4.2 Übertragungsmedien in der Automatisierungstechnik

Vereinzelt kommen in der Automatisierungstechnik Koaxialkabel und Lichtwellenleiter zum Einsatz. Gelegentlich werden auch Daten über Funk oder Infrarot übertragen. Das Standardübertragungsmedium ist jedoch die paarweise verdrehte, geschirmte Zweidrahtleitung:



**Abbildung 33: Ersatzschaltbild eines Kabels**

Die in Abbildung 33 gezeigten Induktivitäten und Kapazitäten einer Leitung wirken als Tiefpass. Das bedeutet: hohe Frequenzen werden nicht übertragen bzw. bedämpft.

Induktivitäten und vor allem die Kapazitäten des Kabels dürfen nur in begrenztem Maß vorhanden sein, da die Tiefpasswirkung sonst zu groß ist und Informationen verloren gehen (z. B. muss die Leitung für eine RS-485-Schnittstelle bei Einsatz für PROFIBUS-DP einen Kapazitätsbelag  $<30\text{pF/m}$  besitzen).

Die Tiefpasswirkung steigt mit größerer Länge des Kabels und höherer Baudrate. Mit größer werdender Leitungslänge ist entsprechend die Baudrate zu reduzieren.

### Beispiel:

Wir werden später sehen, dass PROFIBUS-DP mit der RS-485-Schnittstelle arbeitet und die maximal mögliche Baudrate 12Mbaud beträgt. Tabelle 3 zeigt, wie bei entsprechenden Leitungslängen die Baudrate zu reduzieren ist:

Segmentlänge [m]	100	200	400	1000	1200
Baudrate [kbaud]	12000	1500	500	187,5	9,6

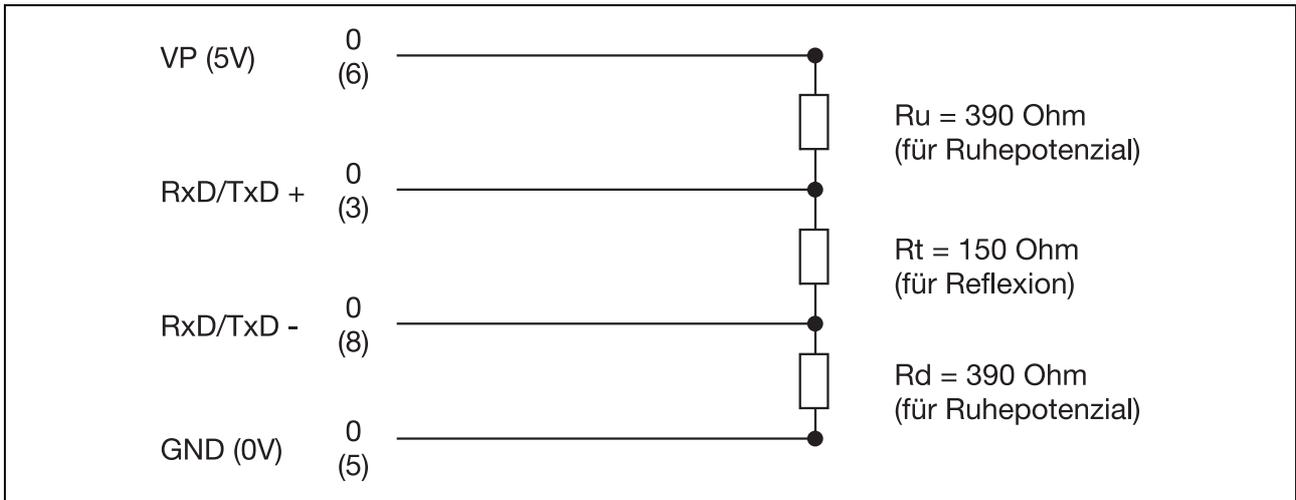
**Tabelle 3: Wahl der Baudrate bei entsprechenden Leitungslängen**

### Bemerkung:

Wie in Kapitel 3 „Serielle Schnittstellen RS232, RS422, RS485 und USB“ beschrieben, kommt in den meisten Fällen eine paarweise verdrehte und abgeschirmte Leitung zum Einsatz. Das Kabel sollte deshalb paarweise verdreht sein, da so die durch ein elektromagnetisches Feld induzierten Spannungen entgegengesetzt wirken und sich aufheben.

# 4 Zugriffsverfahren und Übertragungsmedien

## 4.3 Der Abschlusswiderstand



**Abbildung 34: Abschlusswiderstand am Beispiel einer RS-485-Schnittstelle bei PROFIBUS-DP**

An den offenen Kabelenden (erster und letzter Teilnehmer in einem Bussystem) entstehen immer Leitungsreflexionen. Diese sind umso stärker, je größer die gewählte Baudrate ist. Um die Reflexionen möglichst gering zu halten, findet ein Abschlusswiderstand Einsatz.

Weiterhin wird beispielsweise bei PROFIBUS-DP durch die Verbindung des Abschlusswiderstandes mit dem Bezugspotenzial ein definiertes Ruhepotenzial erreicht.

## Bussysteme

In diesem Kapitel beschreiben wir die Bussysteme, für die JUMO Komponenten liefert. Wir erläutern Grundlagen zu den einzelnen Systemen, geben aber auch Hinweise zur Anbindung. Wir bitten um Beachtung, dass die Ausführungen nicht als Ersatz für die entsprechenden Betriebsanleitungen zu sehen sind.

Im Vorfeld geben wir eine Übersicht zu den in diesem Kapitel beschriebenen Feldbussen:

	Modbus	PROFIBUS-DP	ETHERNET	CANopen	HART®
Zugriffsverfahren	Master/Slave	Master/Slave	CSMA-CD	CSMA-CA	Master/Slave
Medium	beliebig	2-Draht-Leitung oder Lichtwellenleiter	2-Draht-Leitung	2-Draht-Leitung	2-Draht-Leitung
Teilnehmer/Segment	247	32	16.777.216 (A-Klasse-Netz)	127	15
Datenübertragungsrate	je nach Schnittstelle bis zu 187,5 kBit/s	9600 bit/s bis 12 Mbit/s	100 Mbit	50 kbit/s bis 1 Mbit/s	1200 bit/s
Busausdehnung	je nach Schnittstelle bis zu 1200m/Segment	1200m/Segment	maximaler Abstand zwischen zwei Punkten 100m	bis 5000m	bis 2000m

**Tabelle 4: Übersicht der beschriebenen Feldbusse**

### 5.1 Modbus

Modbus ist ein Übertragungsprotokoll, welches 1979 von der amerikanischen Firma Gould-Modicon entwickelt wurde. Modbus findet überwiegend Anwendung im Bereich der Prozessvisualisierung. Es ist ein einfaches und sicheres Protokoll und wird spezifiziert im „Modicon Modbus Protocol Reference Guide“. Der große Vorteil von Modbus ist, dass dieses Protokoll von jedem Programmierer realisiert werden kann und keinerlei Lizenzgebühren fällig werden. Hierdurch entstehen im Vergleich zu anderen Bussystemen kommerzielle Vorteile.

Beim Einsatz von Modbus kann der Anwender selbst bestimmen, welche Schnittstelle er verwendet (RS422/RS485, RS232, LWL etc.). Sicher findet in den meisten Fällen die RS-485-Schnittstelle Anwendung. Wird die RS-485-Schnittstelle verwendet, sind die Teilnehmer in Linie anzuordnen und ggf. der erste und letzte Teilnehmer mit einem Abschlusswiderstand zu versehen (siehe auch Kapitel 3 „*Serielle Schnittstellen RS232, RS422, RS485 und USB*“).

Die Schnittstellen aller Teilnehmer sind hinsichtlich Baudrate, Paritätsprüfung und Anzahl Stoppbits gleich einzustellen. Jedem Teilnehmer ist eine Adresse zu vergeben, diese darf im System nur einmal vorhanden sein.

Hinsichtlich der Baudrate verwendet JUMO die folgenden Standardraten:  
9600, 19200, 38400 Baud.

Modbus arbeitet nach dem Master-Slave-Verfahren. Wie bereits beschrieben, tauscht bei diesem Verfahren ein Master (bei Modbus meist eine Visualisierungssoftware auf einem PC) zyklisch die Daten mit allen Slaves aus.

# 5 Bussysteme



**Abbildung 35: Mehrere Feldgeräte, RS-485-Schnittstelle und PC mit der JUMO SVS3000**

Der Master beginnt mit dem ersten Slave (Abbildung 35) und sendet seine Anfragen einzeln zu diesem Teilnehmer. Der Slave beantwortet die Anfragen (er sendet z. B. den Messwert an Analogeingang 1). Entsprechend sendet der Master nun weitere Anfragen, die der Slave in gleicher Weise beantwortet. Nun sendet der Master seine Anweisungen, z. B. „setze den 1. Sollwert auf 50“. Der Teilnehmer setzt den Sollwert auf 50 und bestätigt dem Master die Sollwertänderung. Die benötigte Zeit, bis die Daten mit allen Slaves ausgetauscht sind und die Datenübertragung von vorne beginnt, wird Buszykluszeit genannt. Diese kann bei Modbus mehrere Sekunden dauern, was im Bereich der Prozessvisualisierungssysteme meist akzeptabel ist.

## 5.1.1 Der Protokollaufbau

Die Anfragen und Anweisungen sind fest definiert und wie folgt aufgebaut:

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

Wir möchten uns den Aufbau einer Anweisung an einem konkreten Beispiel anschauen: Der Regler JUMO dTRON 304 (dieser soll die Adresse 1 besitzen) soll den gefilterten Istwert senden. Die Anfrage des PC's beginnt mit der Slave-Adresse. Die Adresse wird mit 1 Byte übertragen, also wird der Master „00000001“ senden. Hexadezimal ausgedrückt beträgt der Wert 0x01 (mit 0x gibt man an, dass die folgende Zahl eine hexadezimale Zahl ist: 0x01 ist auch dezimal 1, aber 0x10 ist dezimal 16).

Im folgenden Funktionscode wird dem Regler mitgeteilt, ob ein Wert gelesen oder geschrieben werden soll. Es existieren folgende, wichtige Funktionscodes:

- 0x03: Lesen von n Worten
- 0x06: Schreiben eines Wortes
- 0x10: Schreiben von n Worten

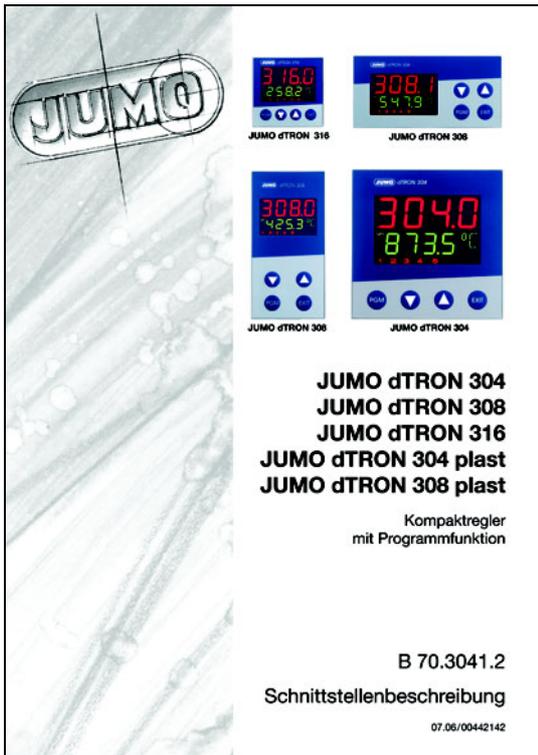
Wir wollen einen Wert lesen, der aus mehreren Worten besteht. Aus diesem Grund wählen wir den Funktionscode 0x03.

Das bisherige Protokoll ergibt sich (als hexadezimale Zahl) zu

01	03
Nachricht	Lesen
Für Slave 1	n Worte

Nun folgt das Datenfeld, welches mit der Adresse des ersten Wortes beginnt.

Hierzu benötigen wir die Schnittstellenbeschreibung des JUMO dTRON 304:



**Abbildung 36: Schnittstellenbeschreibung JUMO dTRON 304**

In der Schnittstellenbeschreibung kann eingesehen werden, ab welcher Adresse der gefilterte Istwert abliegt:

**5 Adresstabellen**

Adresse	Datentyp/ Bitnummer	Zugriff	Signalbezeichnung
0x0026	Bit 3	R/O	Steuerkontakt 4 (=0x0008)
	INT	R/O	Binärsignale
	Bit 0	R/O	Steuerkontakt 1 (=0x0001)
	Bit 1	R/O	Steuerkontakt 2 (=0x0002)
	Bit 2	R/O	Steuerkontakt 3 (=0x0004)
	Bit 3	R/O	Steuerkontakt 4 (=0x0008)
	Bit 4	R/O	Timer 1 (=0x0010)
	Bit 5	R/O	Timer 2 (=0x0020)
	Bit 8	R/O	1. Limitkomparator (=0x0040)
	Bit 9	R/O	2. Limitkomparator (=0x0080)
	Bit 10	R/O	3. Limitkomparator (=0x0100)
	Bit 11	R/O	4. Limitkomparator (=0x0200)
	Bit 12	R/O	Logik-Formel 1 (=0x0400)
	Bit 13	R/O	Logik-Formel 2 (=0x0800)
0x0027	INT	R/W	Ansteuerung der Binärausgänge
	Bit 0	R/W	Ausgang 1 (=0x8001)
	...	R/W	
	Bit 9	R/W	Ausgang 10 (=0x8200)
	Bit 15	R/W	Aktivierung = 1 (=0x8000)
0x0028	FLOAT	R/O	Analogeingang 1 [mV]
0x002A	FLOAT	R/O	Analogeingang 2 [mV]
0x0030	FLOAT	R/O	interner Pt100 [Ohm]
0x0034	INT	R/O	Abtastzeit
0x0035	FLOAT	R/O	Analogeingang 1 [Grad]
0x0037	FLOAT	R/O	Analogeingang 2 [Grad]
0x003D	FLOAT	R/O	Mathematik 1
0x003F	FLOAT	R/O	Mathematik 2
0x0041	FLOAT	R/O	Rampenendwert (W)
0x0043	FLOAT	R/O	Istwert gefiltert
0x0045	FLOAT	R/O	Istwert ungefiltert
0x0047	FLOAT	R/W	Softwert
0x0049	FLOAT	R/O	Stellgrad -100 ... 100% (Anzeigewert)
0x004B	FLOAT	R/O	Stellgrad heizen 0 ... 100%
0x004D	FLOAT	R/O	Stellgrad kühlen -100 ... 0%
0x004F	FLOAT	R/O	Regeldifferenz
0x0053	INT	R/O	Schaltstellung Heizen (1 = Kontakt geschlossen/EIN)
0x0054	INT	R/O	Schaltstellung Kühlen (1 = Kontakt geschlossen/EIN)
0x0055	INT	R/O	Parametersatznummer 0...1
0x0056	INT	R/W	Handstellgrad
0x0057	INT	R/O	Abschnittsnummer (1 ... 8)
0x0058	INT	R/O	Abschnittszahl (1 ... 8)
0x0059	FLOAT	R/O	Programm-Softwert

0x0041	FLOAT	R/O	Rampenendwert (W)
0x0043	FLOAT	R/O	Istwert gefiltert
0x0045	FLOAT	R/O	Istwert ungefiltert

**Abbildung 37: Adresstabellen des JUMO dTRON 304**

## 5 Bussysteme

---

Wie wir sehen können, kann der gefilterte Istwert ab der Adresse 0x0043 erreicht werden. Unser Datenfeld beginnt also mit dem Wert 0x0043. Nun folgt im Datenfeld über 2 Byte die Information, wie viele Worte ab der Adresse 0x0043 gelesen werden sollen. Bei dem Istwert handelt es sich um einen Floatwert, dieser besteht aus zwei Worten, somit lautet das Datenfeld:

0043	0002
Lese ab	lese ab der genannten
Adresse 0x0043	Adresse 2 Worte

Die bisherige Anfrage ergibt sich zu:

01	03	0043	0002
Nachricht	Lese	Lese ab	lese ab der genannten
Für Slave 1	n Worte	Adresse 0x0043	Adresse 2 Worte

Tatsächlich sendet der Master folgende Bits über den Bus:

00000001	00000011	000000001000011	0000000000000010
----------	----------	-----------------	------------------

Aus der Anfrage wird der CRC-Check gebildet und direkt nach der Anfrage gesendet. Der Empfänger erstellt den CRC-Check in gleicher Weise. Kommt er nicht auf das gleiche Ergebnis, wird die Information verworfen. Wie der CRC-Check gebildet wird, ist am Ende dieses Kapitels gezeigt.

Wie bereits erwähnt, handelt es sich bei Modbus um ein altes, jedoch auch bewährtes Protokoll. Es erfreut sich immer noch sehr hoher Akzeptanz bei den Programmierern. Nachteilig ist sicher, dass Programmieraufwand entsteht. Es wird jedoch fast immer auf existierende Treiber zurückgegriffen.

JUMO entwickelt und vertreibt eine Prozessvisualisierungssoftware, welche auf Basis von Modbus arbeitet. Die JUMO SVS3000 ist interessant für Unternehmen, welche eine Vielzahl von JUMO-Geräten einsetzen und die Prozessgrößen dieser Komponenten auf dem Bildschirm eines PC's (z. B. in einer Warte) darstellen möchten. Die Temperaturen, Drücke, pH-Werte können neben der Anzeige auf der Festplatte des PC's abgelegt werden.

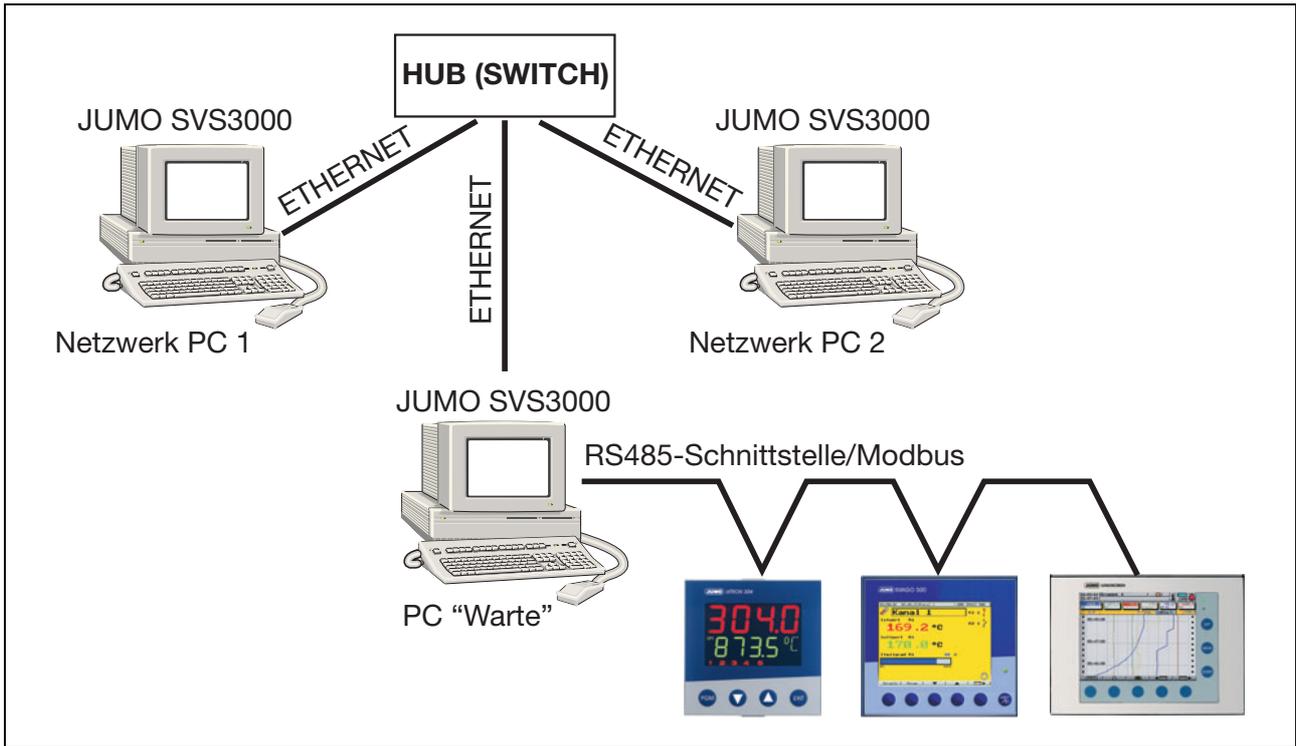
Mit dem vorab beschriebenen Protokollaufbau wird der Anwender nicht konfrontiert. Er muss lediglich die Schnittstelle des PC's definieren (COM-Port-Nr., Baudrate, Parität und Anzahl Stoppbits), die Geräte aus einem Katalog entnehmen und bestimmen, welche Größen zwischen JUMO SVS3000 und Gerät übertragen werden sollen. Nach der Einbindung der Geräte bzw. Funktionen in so genannte Gruppenbilder werden die Größen auf dem Bildschirm des PC's dargestellt:



**Abbildung 38: Gruppenbild der JUMO SVS3000**

Weiterhin ist die JUMO SVS3000 netzwerktauglich, das bedeutet: die Feldgeräte sind möglicherweise lediglich an einen PC in einer Warte angeschlossen. Die JUMO SVS3000 wird auf weiteren PC's installiert, welche über das Netzwerk mit dem PC „Warte“ verbunden sind. Die Prozessdaten werden über das Netzwerk zu diesen PC's gesendet und stehen hier ebenfalls zur Verfügung.

# 5 Bussysteme

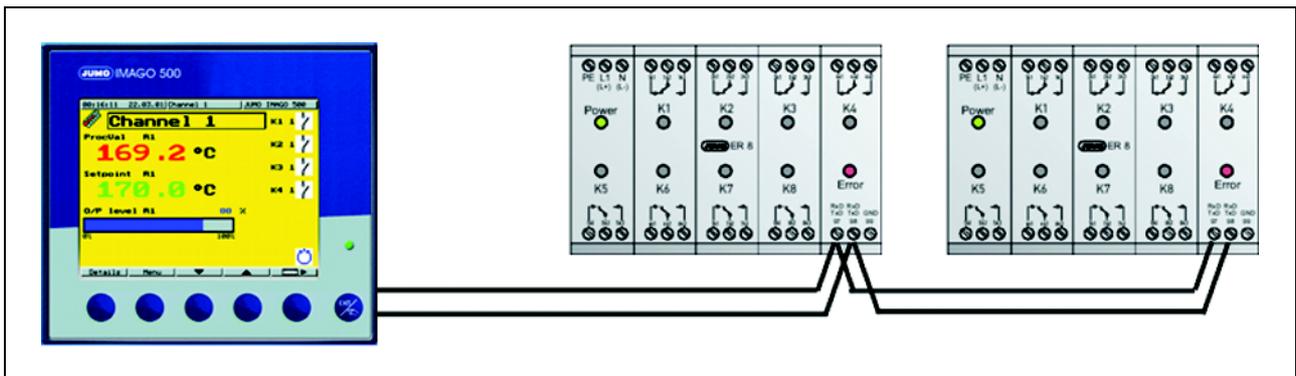


**Abbildung 39: JUMO SVS3000 im Netzwerk auf mehreren PC's**

Im Folgenden werden zwei weitere Anwendungen für Modbus vorgestellt:

## 1. Ansteuerung von externen Relaisbaugruppen

Bei der Arbeit mit Programmreglern werden während eines Programmablaufs in einigen Applikationen sehr viele Aggregate (Klappen, Ventile, Ventilatoren etc.) angesteuert. Ist die im Regler vorhandene Anzahl von Ausgängen nicht ausreichend, können externe Relaismodule über RS-485-Schnittstelle angebunden werden. Die Schnittstelle arbeitet in diesem Fall über Modbus-Protokoll. Anstelle der internen Relais werden nun die Relais der externen Baugruppe angesteuert:



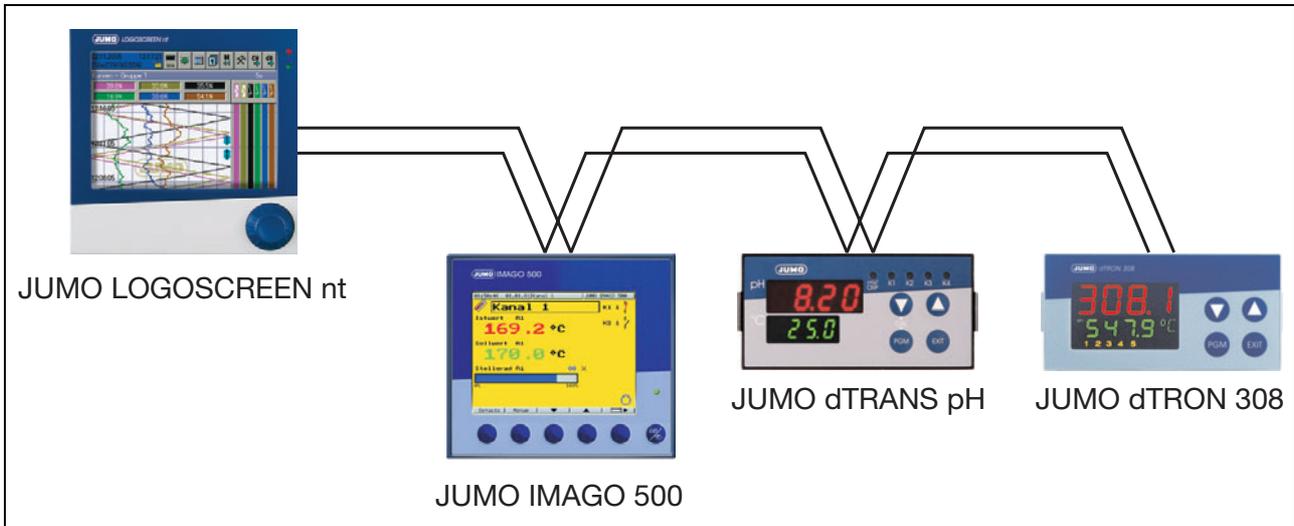
**Abbildung 40: JUMO IMAGO 500 mit zwei externen Relaisbaugruppen ER8**

## 2. Modbus-Master-Funktion

Die Schnittstellen von JUMO-Geräten sind meist als Slaves ausgeführt. Die Geräte führen die Anweisungen des Masters aus bzw. senden Prozessgrößen aufgrund von Anfragen.

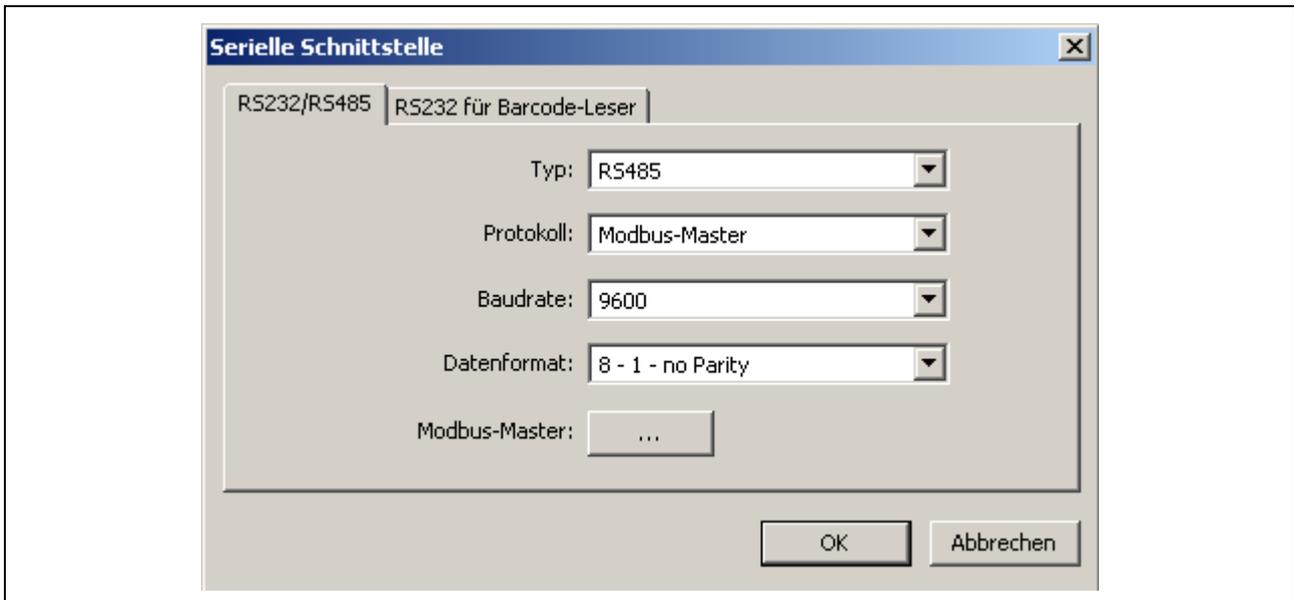
Einige Schnittstellen können als Master-Funktion eingestellt werden; ein Vertreter ist der Bildschirmstreiber LOGOSCREEN nt:

An den Schreiber können bis zu 18 Sensoren/Signale direkt angeschlossen werden. Weitere 24 Signale können beispielsweise mittels RS-485-Schnittstelle und Modbus-Protokoll übertragen werden.



**Abbildung 41: JUMO LOGOSCREEN nt mit Master-Funktion**

In Abbildung 41 sind die Geräte über eine Zweidrahtleitung miteinander zu verdrahten. Die Schnittstelle des LOGOSCREEN nt ist wie folgt zu konfigurieren:



**Abbildung 42: Konfiguration der Schnittstelle des JUMO LOGOSCREEN nt 1**

Die Schnittstelle RS485 findet Verwendung (Typ). Sie soll mit Modbus-Master-Protokoll arbeiten. Baudrate, Anzahl der Stoppbits und Paritätsprüfung (Datenformat) sind bei allen Geräten gleich einzustellen.

Bei den Slave-Geräten ist neben Baudrate, Anzahl der Stoppbits und Paritätsprüfung jeweils eine einmalige Geräteadresse zu vergeben.

Stellen wir uns vor, für den dTRON 308 (Abbildung 41, rechtes Gerät) wurde die Geräteadresse 1 vergeben. Der LOGOSCREEN nt soll den Istwert (gefiltert) dieses Reglers abfragen.

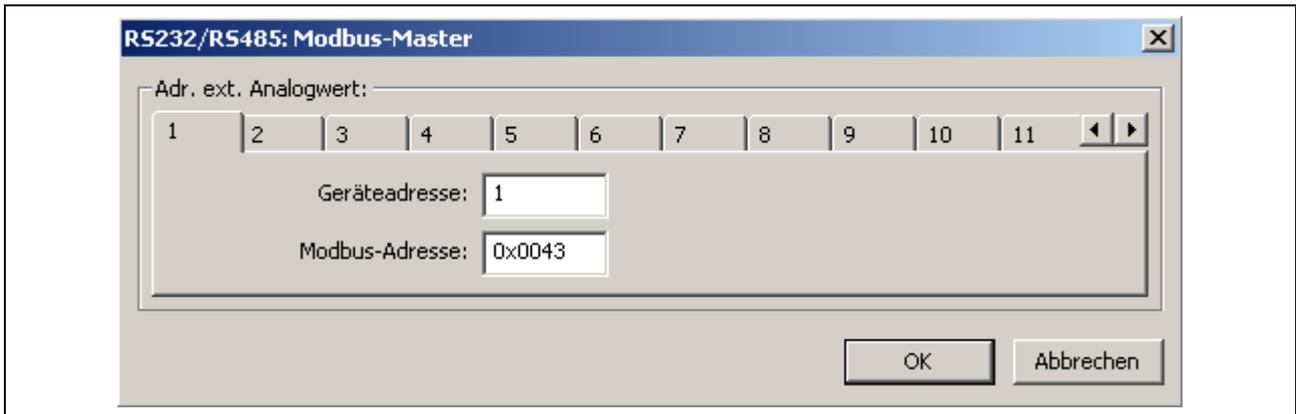
# 5 Bussysteme

Aus der Schnittstellenbeschreibung des Reglers ist ersichtlich, dass der Istwert unter der Adresse 0x0043 zu erreichen ist:

0x0041	FLOAT	R/O	Rampenendwert (W)
0x0043	FLOAT	R/O	Istwert gefiltert
0x0045	FLOAT	R/O	Istwert ungefiltert

**Abbildung 43: Auszug aus der Schnittstellenbeschreibung des JUMO dTRON**

Der Istwert des Reglers ist der erste externe Wert für den Bildschirmschreiber. Für den LOGOSCREEN nt sind folgende Einstellungen durchzuführen:



**Abbildung 44: Konfiguration der Schnittstelle des LOGOSCREEN nt 2**

Der LOGOSCREEN nt fragt nach der Konfiguration immer wiederkehrend den Istwert des Reglers ab. Die Größe steht im Schreiber als erster externer Analogwert zur Verfügung.

## 5.1.2 CRC-Check

Anbei wird verdeutlicht, wie für eine Modbus-Anfrage ein CRC-Check generiert wird.

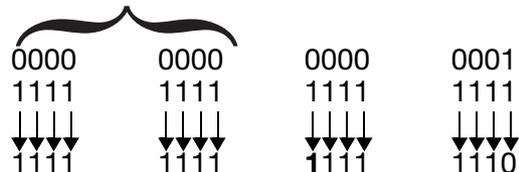
Stellen wir uns vor, ein Feldgerät sendet folgendes Telegramm:

01 03 10260002

Nun wird mit dem ersten Byte (01) begonnen. Es beträgt binär ausgedrückt:  $00000001_2$ .

wird mit 0 aufgefüllt

Für die 8 Bits  
und  
wird bit-weise das Ergebnis einer  
Exklusiv-Oder-Verknüpfung erstellt:



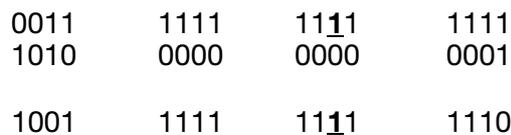
### Bemerkung:

Werden zwei Bits Exklusiv-Oder verknüpft, ist das Ergebnis immer dann 1, wenn die Bits unterschiedlich sind. Nur die erste Exklusiv-Oder-Verknüpfung erfolgt mit 1111 1111 1111 1111 (0xFFFF). Wie wir sehen werden, geschieht die Verknüpfung gleich mit anderen Werten.

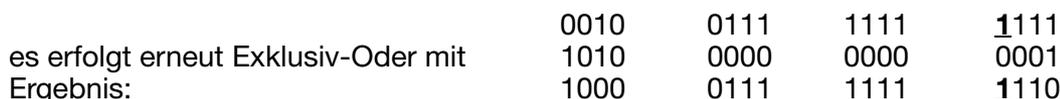
Beim Ergebnis (1111 1111 1111 1110) beträgt der Wert für das rechte Bit 0. Alle Bits werden nach rechts herausgeschoben, bis die erste 1 herausfällt. Die frei werdenden Bits auf der linken Seite werden mit 0 aufgefüllt:

Dies ergibt: 0011 1111 1111 1111 **‡**

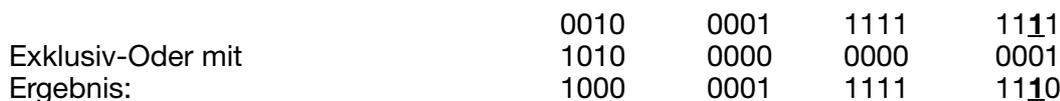
Für die 8 Bits  
und  
wird bit-weise das Ergebnis einer  
Exklusiv-Oder-Verknüpfung erstellt:



Die Bits werden zweimal nach rechts verschoben (dann fällt die 1 heraus). Die frei werdenden Bits auf der linken Seite werden mit 0 aufgefüllt. Das Ergebnis lautet:



Es wird wieder geschoben, bis die erste 1 rechts herausfällt:



Es wird wieder geschoben, bis die erste 1 rechts herausfällt.  
Zu beachten ist, dass insgesamt nur 8-mal geschoben wird:



Der CRC-16-Check lautet bisher:  
(oder hexadezimal: 0x807E)

# 5 Bussysteme

---

**Nun wird das 2. Byte der Anfrage (0x03 = 00000011) mit einbezogen:**

Für das 2. Byte	0000	0000	0000	0011
und der bisherigen CRC-Check	1000	0000	0111	1110
wird bit-weise das Ergebnis einer				
Exklusiv-Oder-Verknüpfung erstellt:	1000	0000	<b>0111</b>	1101

Es wird wieder geschoben, bis die erste 1 rechts herausfällt:

	0100	0000	<b>0011</b>	1110
--	------	------	-------------	------

Die weiteren Exklusiv-Oder-Verknüpfungen werden wieder mit

	1010	0000	0000	0001
--	------	------	------	------

durchgeführt. Es wird immer wieder nach rechts geschoben, bis eine 1 herausfällt und dann die Exklusiv-Oder-Verknüpfung erstellt. Nach dem achten Schieben wird, nur wenn eine 1 herausfällt, das letzte mal die Exklusiv-Oder-Verknüpfung durchgeführt, anderenfalls das Ergebnis so übernommen.

**Das Ergebnis nach Einbeziehen des 2. Bytes würde betragen:**

0010    0001    0100    0000    (oder hexadezimal: 0x2140)

**Nach Einbeziehen des 3. Bytes (10) ergibt CRC-16:**

0011    1100    0010    0001    (oder hexadezimal: 0x3C21)

**Nach Einbeziehen des 4. Bytes (26) ergibt CRC-16:**

1100    0010    0111    1101    (oder hexadezimal: 0xC27D)

**Nach Einbeziehen des 5. Bytes (00) ergibt CRC-16:**

0010    0001    0000    0010    (oder hexadezimal: 0x2102)

**Nach Einbeziehen des 6. Bytes (02) ergibt CRC-16:**

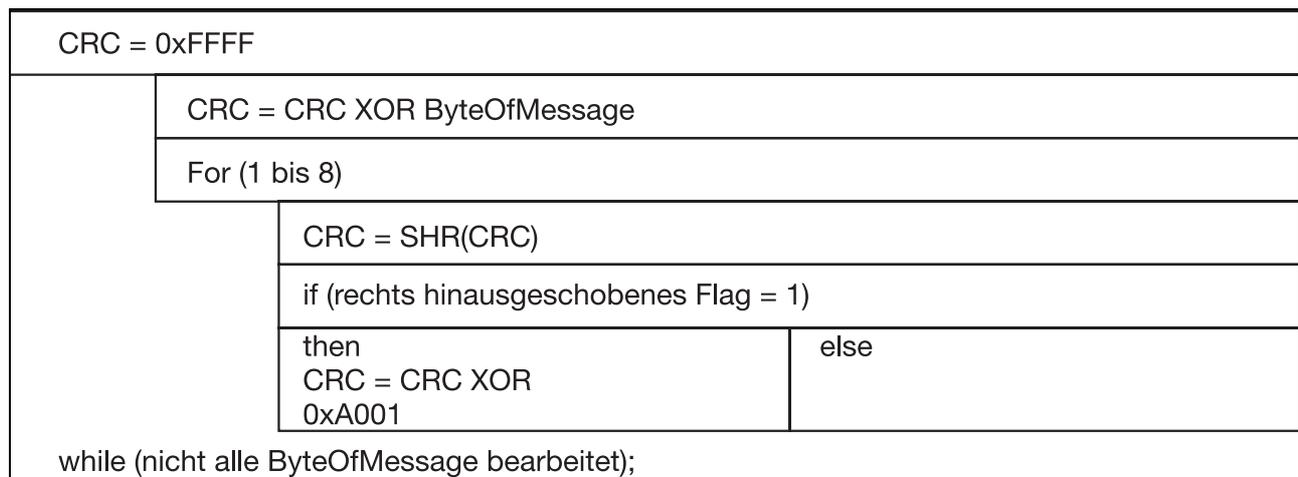
0000    0000    0010    0001    (oder hexadezimal: 0x0021)

Die vollständige Anfrage des Masters lautet wie folgt:

<b>01</b>	<b>03</b>	<b>10260002</b>	<b>2100</b>
			<hr/>
			<b>CRC-16</b>

**Das Low-Byte der Checksumme wird zuerst gesendet. Aus diesem Grund wird 0x2100 (und nicht 0x0021) übertragen.**

Folgende Abbildung zeigt das Berechnungsschema für den CRC-Check:



**Abbildung 45: Berechnungsschema für CRC-Check**

Abschließend ist ein Beispielprogramm zur Berechnung der Checksumme aufgeführt:

## PASCAL/DELPHI

```
// Prüfsumme CRC16
function CRC16 (anweisung:String): String;
VAR CRC: word;           //CRC-Wert
    n,m,i: Integer;
    manz: Integer;       // Anzahl der Bytes
    buff: array [1..256] of byte; // Message-Bytes
begin
    manz := ROUND (Length(anweisung) / 2);
// Umwandlung des Eingangsstrings in Binär-Werte
    i:=1;m:=1;
    for n:=1 to manz do begin
        buff[m] := HEXtoINT (COPY(anweisung,i,2)); m:=m+1; i:=i+2;
    end;
// Berechnung CRC16
    CRC := $FFFF;
    For m:=1 to manz do begin
        CRC := (CRC xor buff[m]);
        For n:=1 to 8 do begin
            if ((CRC and $0001) = 1) then begin
                CRC := ((CRC shr 1) xor $A001);
            end
            else begin
                CRC := (CRC shr 1);
            end;
        end;
        ShowMessage(IntToBin(CRC));
    end;
end;
// Umwandlung CRC16 in gedrehten String
result := COPY (IntToHex(CRC,4),3,2) + COPY (IntToHex(CRC,4),1,2);
end;
```

# 5 Bussysteme

## 5.2 PROFIBUS-DP

PROFIBUS (**PRO**cess **FI**eld **BUS**) ist ein internationaler, offener Feldbusstandard, der in der Feldbusnorm EN 50 170 standardisiert wurde.

Die PROFIBUS-Technologie wurde in Zusammenarbeit mehrerer Firmen - unter Führung von Siemens, Klöckner Möller und Bosch - entwickelt und wird von der PROFIBUS-Nutzerorganisation (PNO) gepflegt.

Derzeit sind zwei Varianten der PROFIBUS-Familie von Bedeutung:

- **PROFIBUS-DP (Decentralized Peripherie)**

ist konzipiert für den Einsatz im Bereich der dezentralen Peripherie, bei dem kurze Systemreaktionszeiten wichtig sind. Er dient fast immer dazu, die dezentralen Automatisierungsgeräte (Regler, Bildschirmschreiber etc.) über eine serielle Schnittstelle mit einer SPS zu verbinden.

- **PROFIBUS-PA (Process Automation)**

wurde speziell für die Verfahrenstechnik konzipiert und erlaubt die Anbindung von Sensoren und Aktoren im Ex-Bereich. PROFIBUS-PA ermöglicht die Kommunikation und Energieversorgung der Geräte in 2-Drahttechnik gemäß dem internationalen Standard IEC 1158-2.

PROFIBUS-DP und -PA sind für die schnelle Datenübertragung kleiner Datenmengen konzipiert.

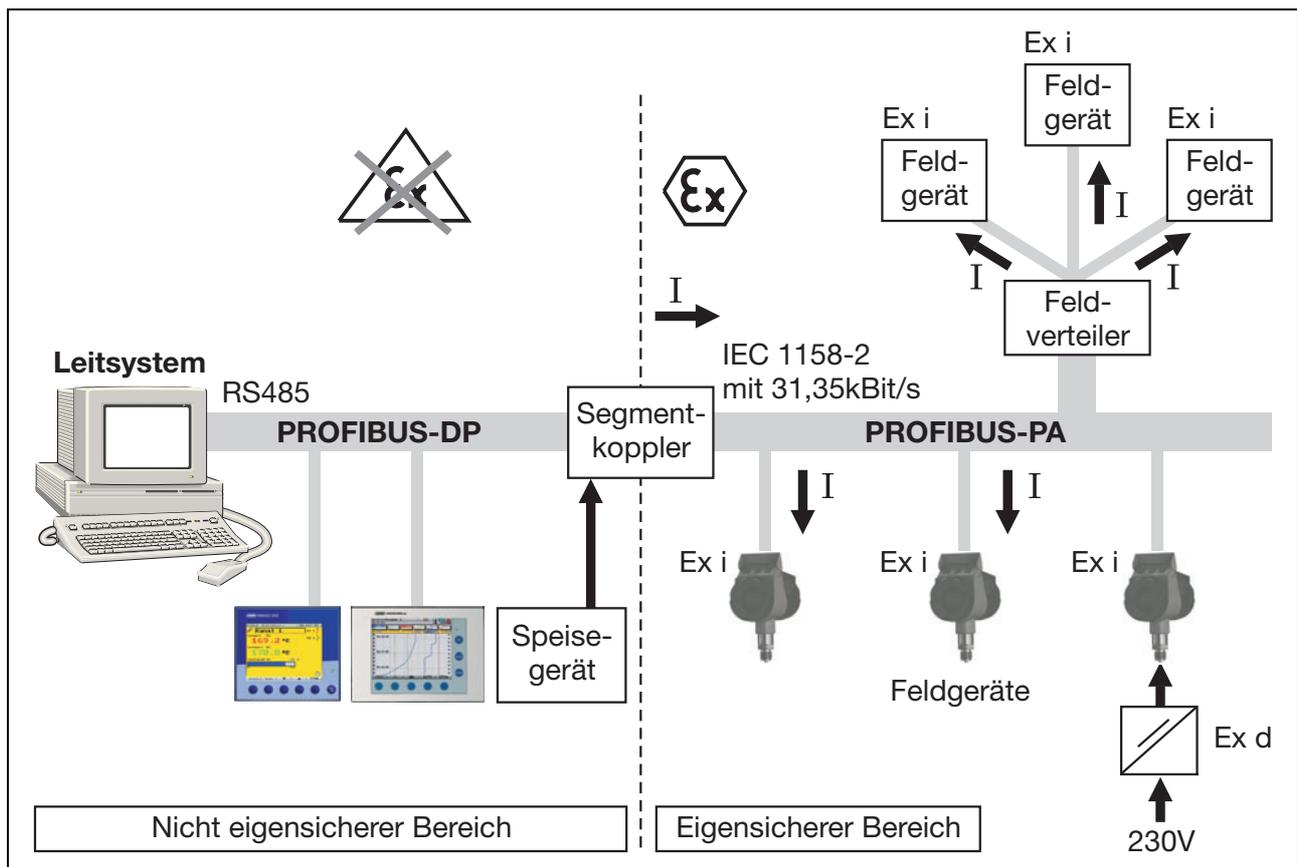
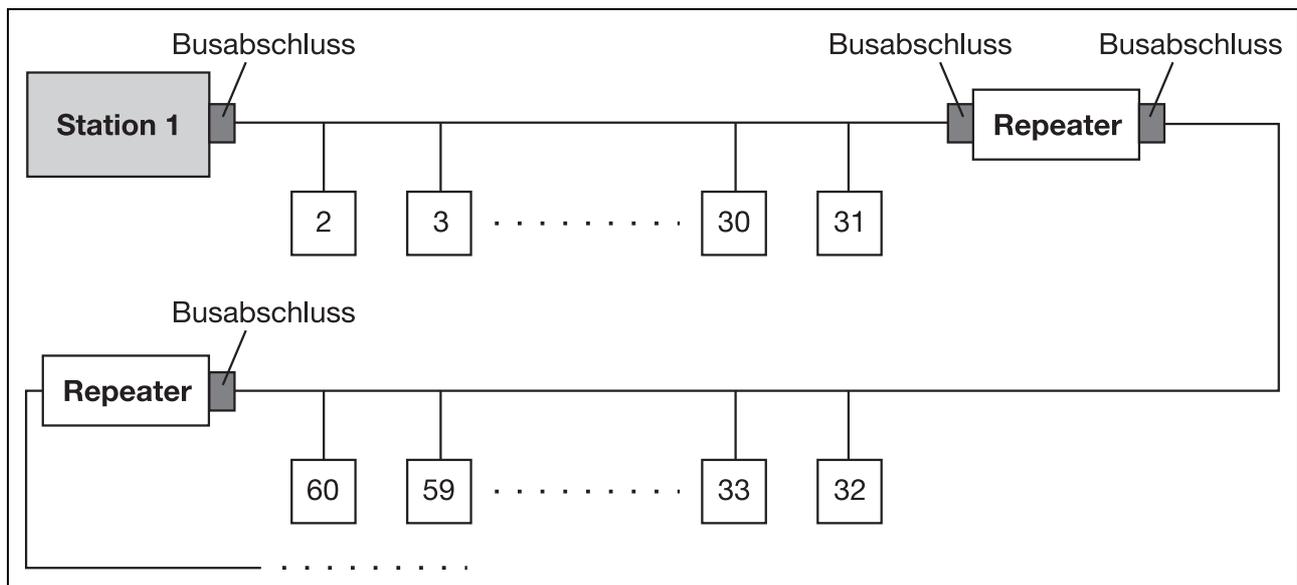


Abbildung 46: PROFIBUS-DP und PROFIBUS-PA

Von der SPS aus betrachtet, werden die Teilnehmer immer über PROFIBUS-DP angebunden. Aus diesem Grund beginnen wir mit dieser Ausprägung von PROFIBUS und wollen hier erwähnen, dass JUMO ausschließlich Geräte mit PROFIBUS-DP liefert. Maximal 32 Teilnehmer werden in Lini-

entopologie angeordnet. Sollen mehr Teilnehmer an eine Schnittstelle angebunden werden, stehen Repeater zur Verfügung:



**Abbildung 47: Linientopologie und Einsatz von Repeatern**

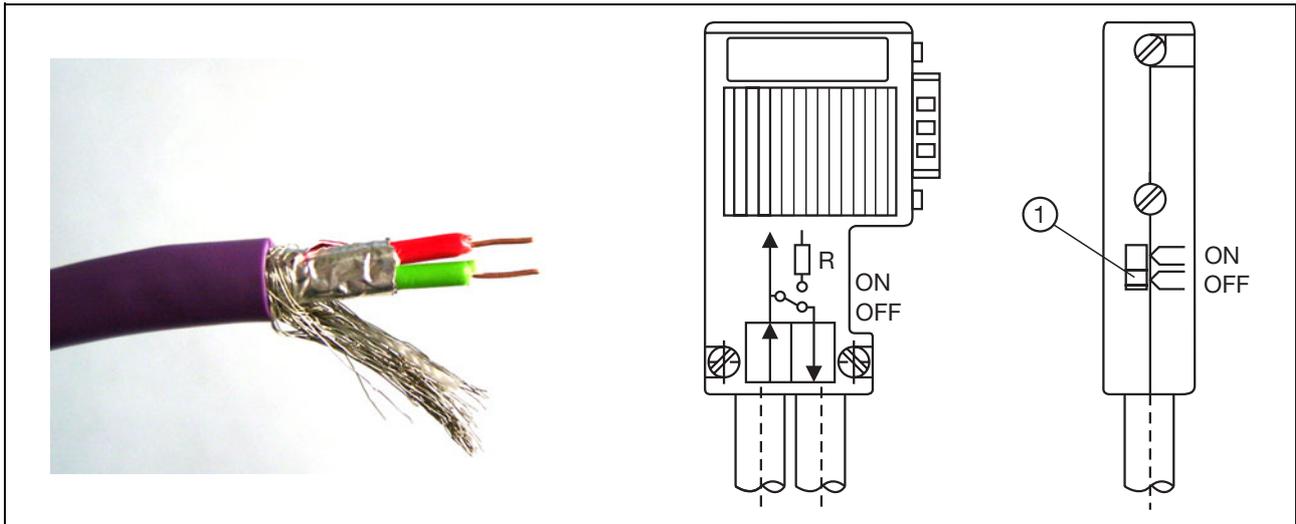
Zwischen Repeatern entstehen Segmente. In diesen befinden sich immer maximal 32 Teilnehmer (Repeater werden mitgezählt). In der Summe können über die Schnittstelle 127 Teilnehmer angeschlossen werden. Neben der höheren Anzahl von Teilnehmern kommen Repeater auch dann zum Einsatz, wenn man die maximal erreichbare Ausdehnung erhöhen möchte: wird das PROFIBUS-DP-System z. B. mit der typischen Baudrate von 1,5 MBaud betrieben, kann gemäß Spezifikation eine Ausdehnung von 200 m erreicht werden. Durch Einführung eines Repeaters entstehen zwei Segmente. Jedes Segment erreicht nun eine Länge von 200 m und die gesamte Netzausdehnung (erster bis letzter Teilnehmer) beträgt 400 m.

Bei PROFIBUS-DP kommt meist die RS-485-Schnittstelle zum Einsatz. Die Baudrate kann in einem Bereich von 9600 Baud - 12 MBaud gewählt werden. Die erreichbaren Netzausdehnungen liegen im Bereich von 1200 m (bei 9600 Baud) bis 100 m (bei 12 MBaud).

In der Regel wird das in EN 50 170 Part 8-2 als Leitungstyp A definierte Kabel eingesetzt.

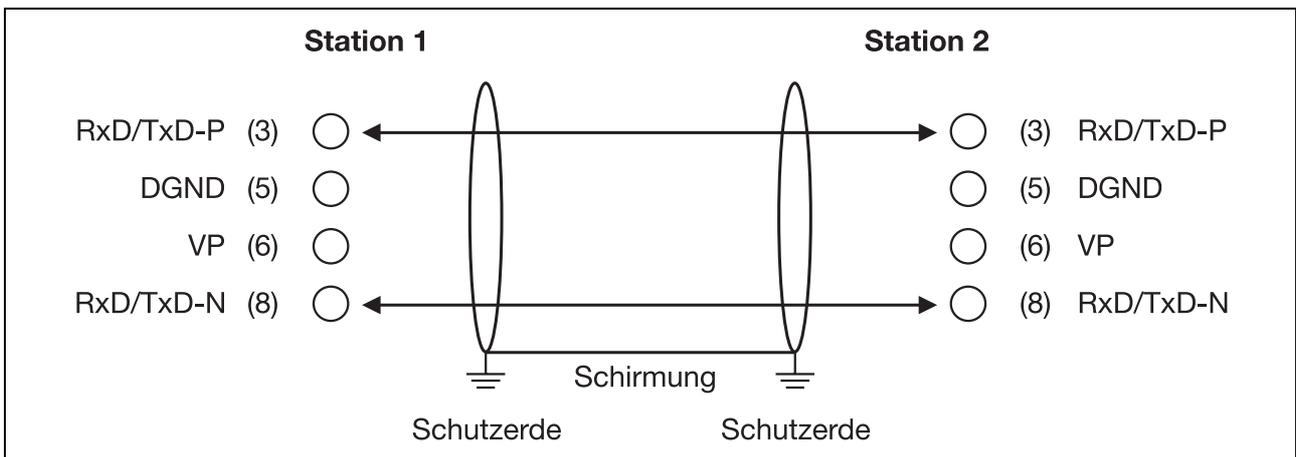
Die Verdrahtung ist sehr einfach: Die beiden Adern des Kabels (Abbildung 48) sind rot bzw. grün gefärbt. Die Klemmen im Stecker sind durch die gleichen Farben markiert. Die ankommende und die abgehende Leitung werden auf die entsprechend gefärbten Klemmen gelegt (die Klemmen für die ankommenden Leitungen sind meist mit A1 und B1, die für die abgehenden Leitungen mit A2 und B2 gekennzeichnet).

## 5 Bussysteme



**Abbildung 48: Kabeltyp A und schematische Darstellung des PROFIBUS-Steckers**

Die Abbildung 49 zeigt die Belegung des PROFIBUS-Steckers.



**Abbildung 49: Belegung des PROFIBUS-Steckers**

Der erste und letzte Teilnehmer ist im Fall einer RS-485-Schnittstelle mit einem Abschlusswiderstand zu versehen. Die Stecker (Abbildung 48) beinhalten diesen Abschlusswiderstand bereits, er muss nur noch über den Schiebeschalter aktiviert werden.

PROFIBUS-DP arbeitet meist nach dem Master-Slave-Prinzip. Der Master (überwiegend eine SPS) tauscht zyklisch die Daten mit den Slaves (Regler, Bildschirmschreiber etc.) aus, die Buszykluszeit liegt hierbei im ms-Bereich. Bei PROFIBUS-DP können aber auch mehrere Master zum Einsatz kommen. In diesem Fall kann nur ein Master zur gleichen Zeit mit den Slaves kommunizieren. Das Zugriffsrecht (der Token) wird dann zwischen den Mastern hin- und hergereicht und nur die Station, die den Token besitzt, darf von sich aus auf den Bus senden.

Bei PROFIBUS-DP beschränken sich die Einstellungen der Slaves bezüglich der Schnittstelle auf die Geräteadresse. Besitzt der Slave eine automatische Baudrate-Erkennung, muss nicht einmal die Geschwindigkeit an diesem eingestellt werden.

Für die Anbindung der Slaves an eine SPS muss der Lieferant eines PROFIBUS-DP-Gerätes eine GSD (**Geräte-Stamm-Daten**)-Datei mitliefern. Die GSD-Datei wird später in das Konfigurationswerkzeug der SPS eingelesen und legt unter anderem die Daten fest, welche von der SPS zum Gerät übertragen werden bzw. in umgekehrter Richtung fließen.

Beispielhaft wollen wir uns anschauen, was für Schritte notwendig sind, um den Istwert eines Reglers vom Typ JUMO IMAGO 500 in eine S7 von Siemens einzulesen:

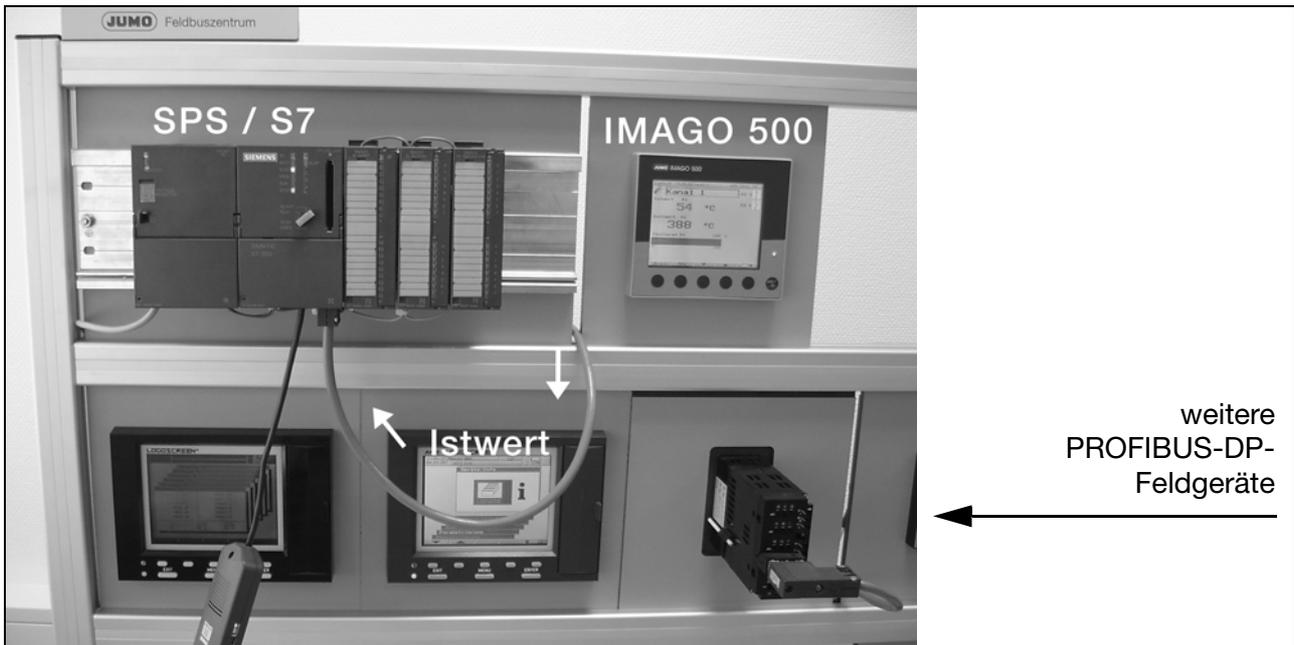


Abbildung 50: JUMO IMAGO 500 und SPS

Wird von JUMO ein PROFIBUS-DP-Gerät bezogen, ist der so genannte GSD-Generator Teil der Lieferung. Mit diesem Programm erstellt sich der Anwender eine GSD-Datei für das Gerät und bestimmt, welche Daten über PROFIBUS-DP gesendet werden. Mit einer festen GSD-Datei würden generell bestimmte Daten übertragen werden - gleich, ob diese benötigt werden oder nicht. Das Bussystem würde unnötig belastet werden.

Um eine GSD-Datei für das genannte Beispiel zu erzeugen, wird der GSD-Generator gestartet und bestimmt, dass für einen JUMO IMAGO 500 eine GSD-Datei erstellt werden soll.

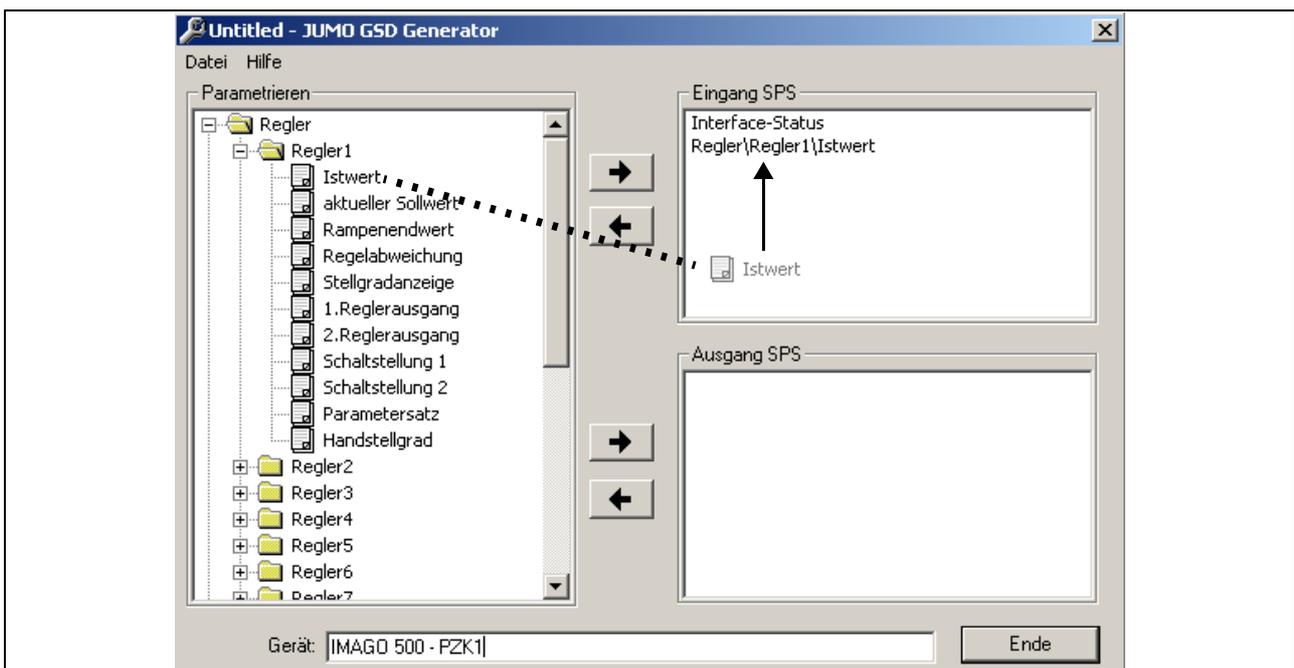


Abbildung 51: Erstellen einer GSD-Datei

# 5 Bussysteme

Im Fenster „Parametrieren“ (Abbildung 51) sind alle Parameter aufgelistet, welche auf diesem einfachen Weg über PROFIBUS-DP gesendet werden können. Nun kann man sich z. B. vom Regler 1 (mit dem JUMO IMAGO 500 sind bis zu acht Reglerkanäle möglich) den Istwert in das Fenster „Eingang SPS“ stellen, dies bedeutet: der Istwert wird später zyklisch in den Eingangsspeicher der SPS gelesen. In gleicher Weise können weitere Parameter in die beiden Fenster „Eingang SPS“ und „Ausgang SPS“ gestellt werden.

Der unter Gerät angegebene Name „IMAGO 500 - PZK1“ ist editierbar und taucht später im Gerätekatalog des SIMATIC-Manager auf.

Abschließend ist die GSD-Datei abzuspeichern.

Am JUMO IMAGO 500 ist die Adresse zu definieren. Diese stellen wir auf „1“ ein. Weitere Einstellungen hinsichtlich der Schnittstelle sind nicht notwendig.

Nun beginnt die Arbeit mit dem SIMATIC-Manager. Dieses Programm dient der Konfiguration (Hard- und Software) der SPS. Starten wir mit der Hardware. Wir gehen davon aus, dass die Hardware der SPS bereits konfiguriert wurde (CPU, Netzteil, Ein-/Ausgabemodule etc.). Weiterhin wurde PROFIBUS-DP aktiviert und eine Baudrate von 1,5 Mbaud eingestellt.

Schauen wir uns nun die SPS in der Hardwarekonfiguration an:

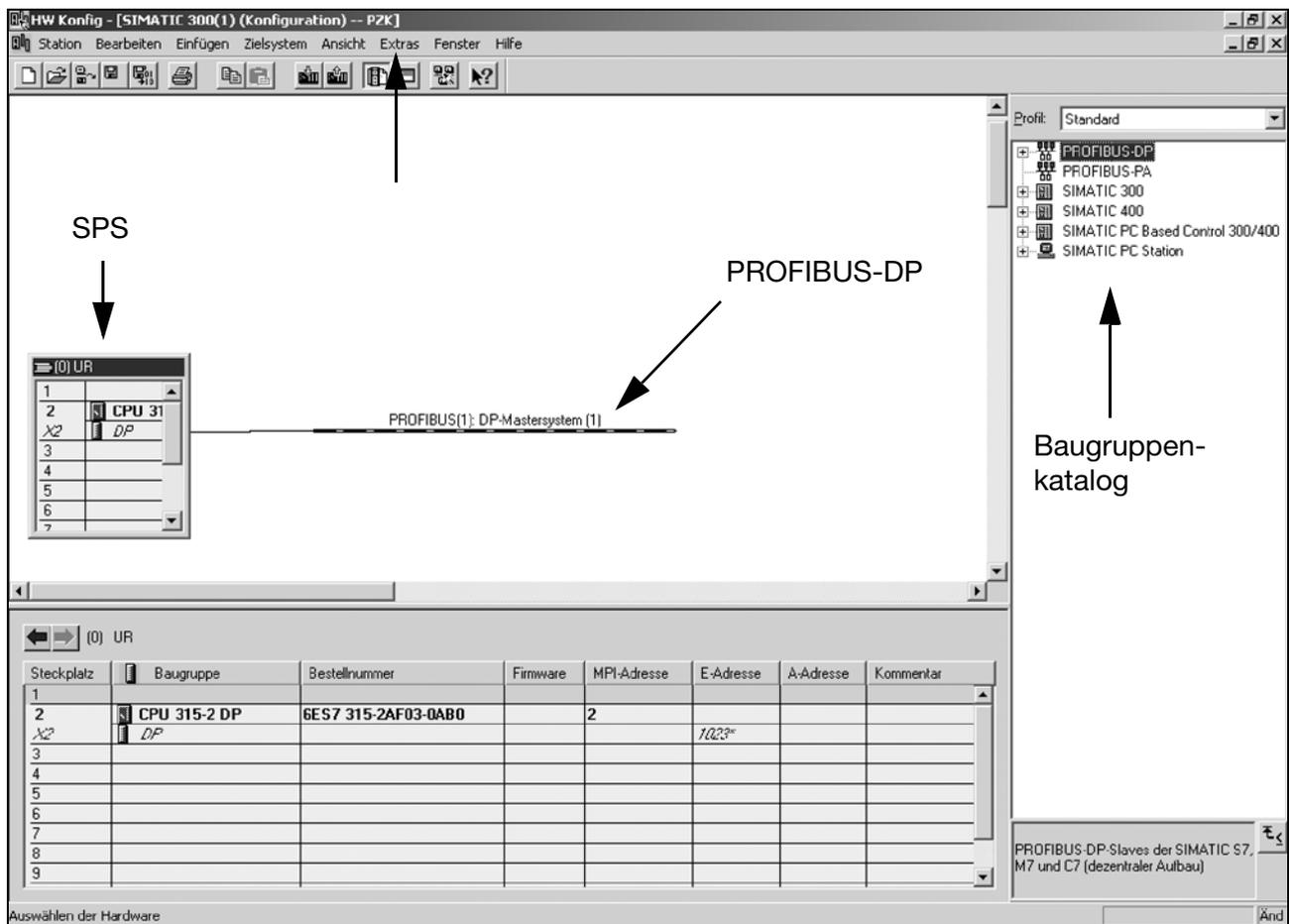
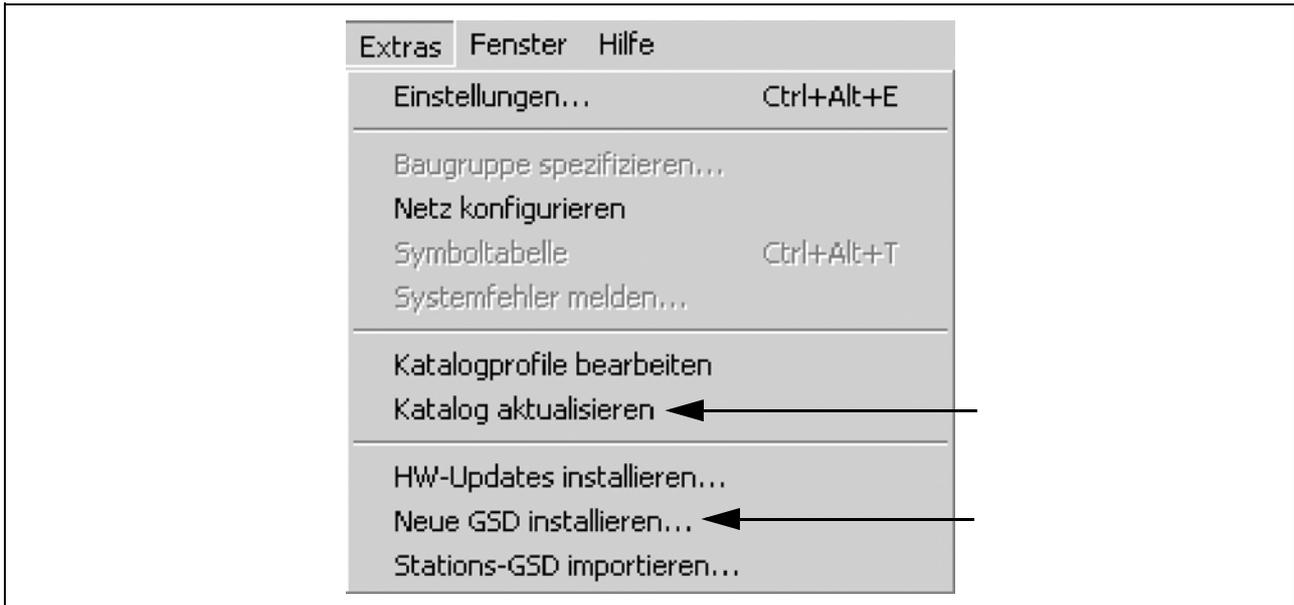


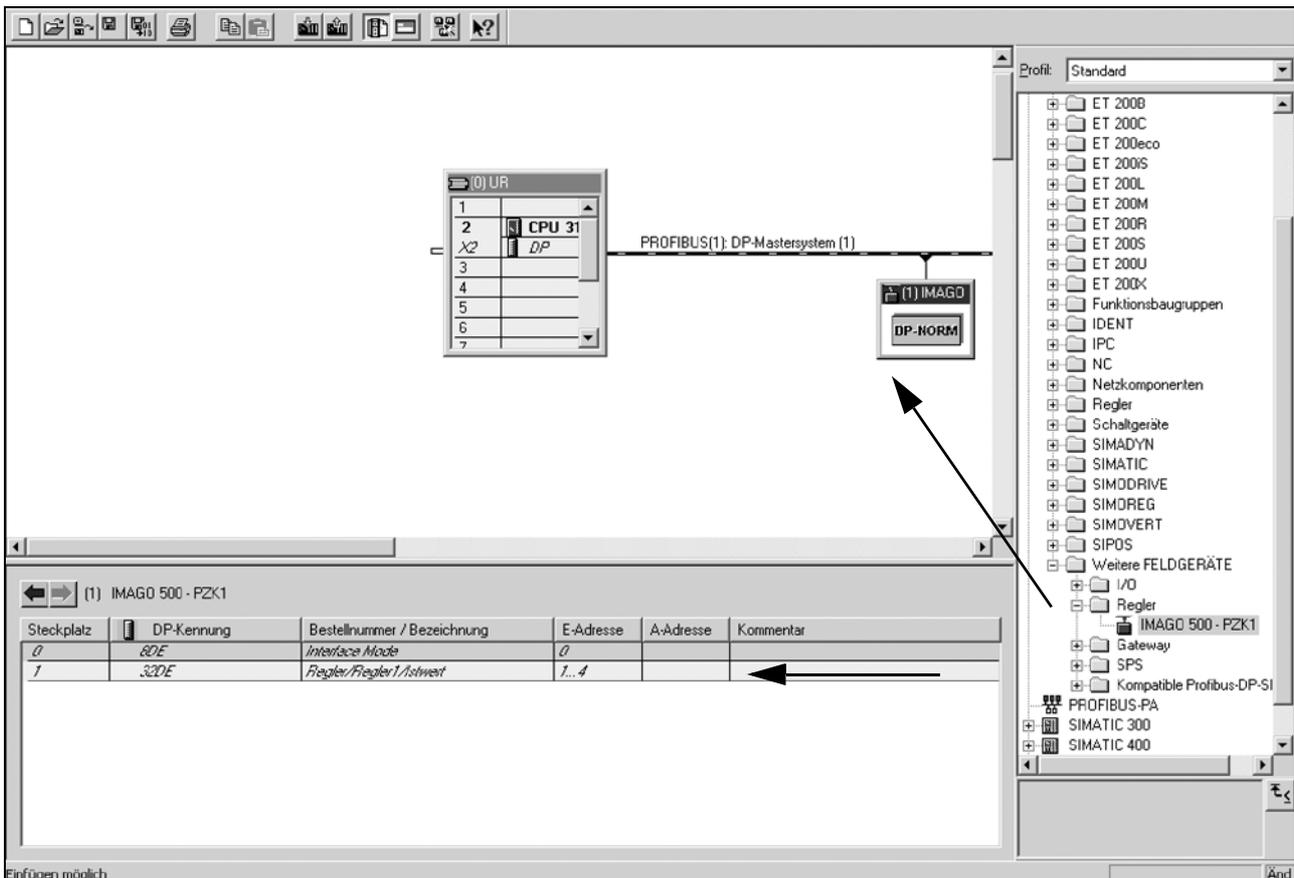
Abbildung 52: SPS mit PROFIBUS-DP in der Hardwarekonfiguration

Abbildung 52 zeigt die SPS mit dem PROFIBUS-DP-Strang. Um den Regler an diesen Strang anzubinden, muss der JUMO IMAGO 500 in den Baugruppenkatalog gebracht werden. Dies geschieht über das Menü „Extras“:



**Abbildung 53: Menü „Extras“ in der Hardwarekonfiguration**

Im Untermenü „Neue GSD installieren“ wird nun die zuvor erstellte GSD-Datei aufgerufen. Nach Auswahl von „Katalog aktualisieren“ wird der JUMO IMAGO 500 unter der vorher angegebenen Bezeichnung im Baugruppenkatalog angelegt:



**Abbildung 54: JUMO IMAGO 500 an PROFIBUS-DP anbinden**

## 5 Bussysteme

---

Der Regler kann nun dem Baugruppenkatalog entnommen und auf den PROFIBUS-DP-Strang gezogen werden. Für den Regler muss an dieser Stelle im SIMATIC-Manager ebenfalls die Adresse definiert werden (in unserem Beispiel „1“). Wird diese Konfiguration in die SPS übertragen, fragt die SPS zyklisch den Istwert des Reglers ab und legt diesen in ihren Eingangsspeicher ab dem Byte „1“ ab (Abbildung 54). Ab diesem Moment kann der Istwert über die Adresse des Speicherbereiches (liegt im Eingangsspeicher im Bereich Byte 1 ... 4) in das Programm der SPS mit einbezogen werden.

Wie bereits erläutert, ist das Standardmedium für PROFIBUS-DP die RS-485-Schnittstelle. Im Fall einer störbehafteten Umgebung sowie zur Erreichung größerer Reichweiten kommen auch Lichtwellenleiter zum Einsatz. Auf dem Markt sind spezielle Busstecker mit integrierter Wandlung der RS-485-Signale auf LWL und umgekehrt erhältlich. Somit kann innerhalb der Anlage auf einfache Weise zwischen den beiden Übertragungsarten gewechselt werden.

### **Abschließend noch einige Informationen zu PROFIBUS-PA:**

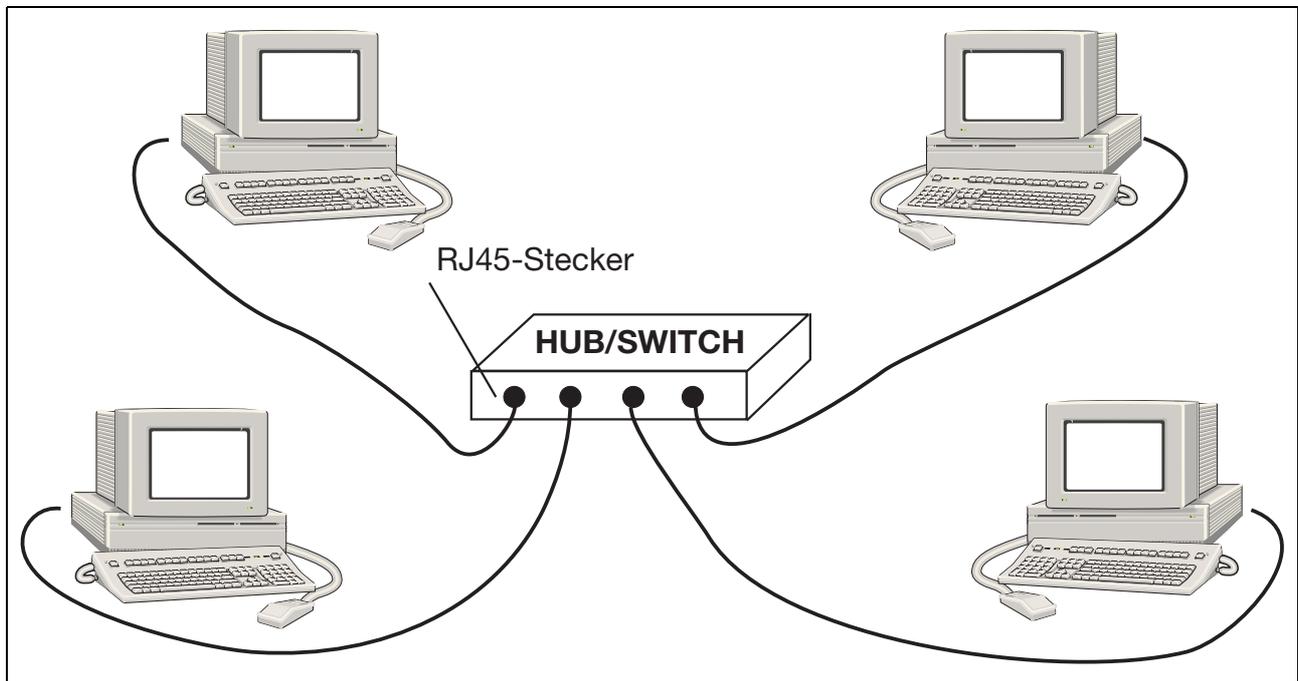
Wie aus Abbildung 46 zu sehen ist, geschieht die Anbindung einer SPS an PROFIBUS immer über PROFIBUS-DP (meist RS-485-Schnittstelle). Durch Segmentkoppler wird der Pegel der RS-485-Schnittstelle in ein Signal gemäß IEC 1158-2 umgeformt. Diese Schnittstelle ermöglicht den Einsatz der Feldgeräte im Ex-Bereich. Es sind Netze in Linien-, Baum- und Sterntopologie möglich. Die Übertragungsgeschwindigkeit beträgt 31,25 kBaud (fest) für die Zweileiterverbindung (Abbildung 46).

Bei dieser Technik gibt es in jedem Segment nur eine einspeisende Quelle, das Speisegerät. Das Speisegerät ist häufig im Segmentkoppler enthalten, der die Anbindung der Busabschnitte an Segmente der RS-485-Übertragung realisiert. In einem Segment ist der Anschluss auf 32 Teilnehmer beschränkt, die Anzahl wird jedoch durch die gewählte Zündschutzart und eventuelle Busspeisung weiter eingeschränkt.

Interessant bei PROFIBUS-PA ist weiterhin, dass beim Senden eines Teilnehmers keine Leistung in den Bus gespeist wird, da die Geräte als Stromsenken wirken: jeder Teilnehmer nimmt einen konstanten Grundstrom auf (z. B. 10 mA) und das Kommunikationssignal wird vom sendenden Gerät durch Aufmodulieren von  $\pm 9$  mA erzeugt.

## 5.3 ETHERNET

Bei ETHERNET werden alle Teilnehmer über Patch-Kabel und HUB's/SWITCHES miteinander verbunden.



**Abbildung 55: Teilnehmer über HUB/SWITCH an ETHERNET angebunden**

HUB's oder SWITCHES bieten die Möglichkeit, mehrere Netzteilnehmer sternförmig miteinander zu verbinden. Datenpakete, die auf einem Port empfangen werden, leitet beispielsweise der HUB auf alle Ports weiter. Die Information steht somit allen Teilnehmern zur Verfügung. Sendet ein Teilnehmer Daten in das Netzwerk, muss sichergestellt sein, dass auch der richtige Computer die Daten empfängt. Der Zielcomputer braucht eine eindeutige Adresse. Dazu wird in die Netzwerkkarte eine weltweit eindeutige Adresse eingebrannt. Diese so genannte MAC(**M**edia **A**ccess **C**ontrol)-Adresse hat einen einheitlichen Aufbau und besteht aus 6 Bytes:

Hersteller (z. B. JUMO)			Netzwerkkarte		
00	0C	D8	5A	CD	D0

Die ersten drei Bytes stehen für den Hersteller. Die nächsten drei Bytes kennzeichnen die einzelne Karte.

Die Kennziffern des Herstelleranteils werden von der IEEE (Institute of **E**lectrical and **E**lectronic **E**ngineers) zentral vergeben. Die Kennung der einzelnen Karte wird von dem Hersteller selbst vorgenommen.

Die MAC-Adresse ist üblicherweise auf einem Label abgedruckt, das direkt auf der Netzwerkkarte zu finden ist. Bei Bildschirmschreibern von JUMO mit ETHERNET-Schnittstelle kann die Adresse z. B. in der Geräteinfo ausgelesen werden.

ETHERNET schafft die physikalische Grundlage für TCP/IP:

# 5 Bussysteme

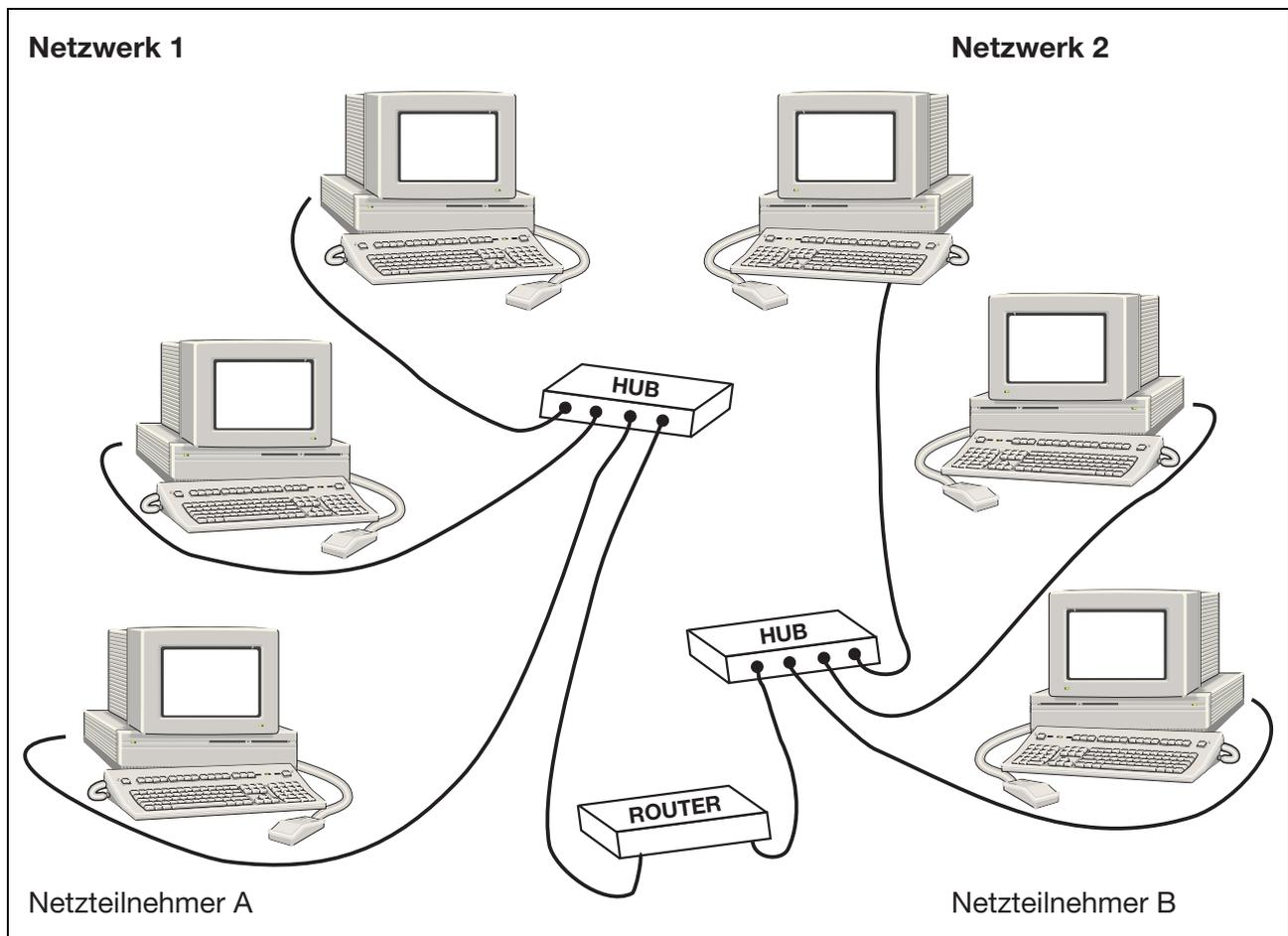
## 5.3.1 TCP/IP

Bereits in den 60er Jahren vergab das amerikanische Militär den Auftrag, ein Protokoll zu schaffen, das unabhängig von der verwendeten Hard- und Software einen standardisierten Informationsaustausch zwischen einer beliebigen Anzahl verschiedener Netzwerke möglich machen sollte. Aus dieser Vorgabe entstand im Jahr 1974 das Protokoll TCP/IP.

Obwohl TCP und IP immer in einem Wort genannt werden, handelt es sich um zwei aufeinander setzende Protokolle. Das **Internet-Protokoll IP** übernimmt die richtige Adressierung und Zustellung der Datenpakete, während das darauf aufsetzende **Transport-Control-Protokoll TCP** für den Transport und die Sicherung der Daten zuständig ist.

## 5.3.2 IP - Internet-Protokoll

Das Internet-Protokoll macht es möglich, eine unbestimmte Anzahl von Einzelgeräten zu einem Gesamtnetzwerk zusammenfügen. Es ermöglicht den Datenaustausch zwischen zwei beliebigen Netzteilnehmern, die jeweils in beliebigen Einzelnetzen positioniert sind:



**Abbildung 56: Zwei Einzelnetze verbunden über Router**

Die Adressierung von Endgeräten mit ETHERNET-Schnittstelle (meist PC's) erfolgt jedoch nicht über die MAC-Adresse direkt, sondern über die so genannte IP-Adresse. Wie wir gleich sehen werden, kann durch Verwendung der IP-Adresse beispielsweise eine logische Zuordnung zu unterschiedlichen Netzen erfolgen.

Den vorher genannten, physikalischen Netzwerkadressen (MAC-Adressen) werden die logischen IP-Adressen zugeordnet.

IP-Adressen sind 32 Bit lange Binärzahlen (4 Byte), die zur besseren Lesbarkeit im „Dotted-Decimal-Format“ angegeben werden: vier durch einen Punkt getrennte Dezimalzahlen (Oktette), wie z. B. 193.96.1.200.

Ein Oktett kann zur Netzadressierung oder zur Adressierung des Teilnehmers (Host) verwendet werden. Je nachdem, wieviele Oktette für die Netz- bzw. Teilnehmeradressierung verwendet werden, spricht man von unterschiedlichen Netzklassen.

Für den Praktiker sind im Wesentlichen drei Netzklassen von Bedeutung: A-, B- und C-Klasse.

Im A-Klasse-Netz wird über das erste Oktett das Netz adressiert. Für alle Teilnehmer (Hosts) im Netz ist diese Netzadresse (das erste Oktett) gleich.

Beispiel A-Klasse-Netz (in diesem Beispiel haben alle Teilnehmer die Netzadresse oder Netz-ID 1):

1.	<u>0.0.1</u>
Netz-ID	Host-ID

Sind beispielsweise alle Hosts dem Netz 1 zugeordnet, könnte dem ersten Teilnehmer die IP 1.0.0.1, dem letzten Teilnehmer die IP 1.255.255.254 vergeben werden.

Die Definition der Netzklasse erfolgt über die Subnetmaske (oder Subnetmask): Die Subnetmaske besteht ebenfalls aus 4 Oktetten. Wird das erste Oktett auf 255, die verbleibenden drei Oktette auf 0 gesetzt (255.0.0.0), bedeutet dies: das erste Oktett der IP wird zur Netzadressierung, die verbleibenden drei Oktette zur Hostadressierung verwendet. Mit der genannten Subnetmaske (255.0.0.0) erhalten wir ein A-Klasse-Netz.

Für ein B-Klasse-Netz müsste die Subnetmaske wie folgt lauten: 255.255.0.0.

In einem B-Klasse-Netz werden die ersten beiden Oktette zur Netzadressierung, die beiden verbleibenden zur Host-Adressierung verwendet.

Befinden sich die Teilnehmer im ersten möglichen B-Klasse-Netz, hätte der erste Teilnehmer die IP

<u>128.1.</u>	<u>0.1</u>
Netz-ID	Host-ID

Der letzte Teilnehmer dieses B-Klasse-Netzes hätte die Adresse 128.1.255.254.

Entsprechend würde mit der Subnetmaske 255.255.255.0 ein C-Klasse-Netz definiert.

# 5 Bussysteme

## Adressierungsregeln:

Prinzipiell kann ein Oktett (ein Byte) jeden Wert im Bereich von 0 ... 255 annehmen. Bei den IP-Adressen wird auf Grund spezieller Adressierungsregeln jedoch nicht der ganze Zahlenbereich verwendet. Auf die Adressierungsregeln möchten wir im Einzelnen nicht eingehen, jedoch werden in der folgenden Tabelle die Adressierungsmöglichkeiten aufgezeigt:

A-Klasse					
		Netz-ID		Host-ID	
1. Netz/1. Host	binär	00000001	00000000	00000000	00000001
	dezimal	1	0	0	1
1. Netz/letzter Host	binär	00000001	11111111	11111111	11111110
	dezimal	1	255	255	254
Letztes Netz/1. Host	binär	011111110	00000000	00000000	00000001
	dezimal	126	0	0	1
Letztes Netz/letzter Host	binär	011111110	11111111	11111111	11111110
	dezimal	126	255	255	254
B-Klasse					
		Netz-ID		Host-ID	
1. Netz/1. Host	binär	10000000	00000001	00000000	00000001
	dezimal	128	1	0	1
1. Netz/letzter Host	binär	10000000	00000001	11111111	11111110
	dezimal	128	1	255	254
Letztes Netz/1. Host	binär	10111111	11111111	00000000	00000001
	dezimal	191	255	0	1
Letztes Netz/letzter Host	binär	10111111	11111111	11111111	11111110
	dezimal	191	255	255	254
C-Klasse					
		Netz-ID			Host-ID
1. Netz/1. Host	binär	11000000	00000000	00000001	00000001
	dezimal	192	0	1	1
1. Netz/letzter Host	binär	11000000	00000000	00000001	11111110
	dezimal	192	0	1	254
Letztes Netz/1. Host	binär	11011111	11111111	11111111	00000001
	dezimal	191	255	255	1
Letztes Netz/letzter Host	binär	11011111	11111111	11111111	11111110
	dezimal	223	255	255	254

**Tabelle 5: Adressierungsmöglichkeiten**

## Bemerkung:

Unterschiedliche Netze können über so genannte Bridges oder Router miteinander verbunden werden.

### 5.3.3 Vergabe von IP-Adressen

Für die Vergabe von IP-Adressen in einem Netzwerk existieren unterschiedliche Möglichkeiten:

#### Vergabe aus einem Adresspool:

Mit DHCP (**D**ynamic **H**ost **C**onfiguration **P**rotocol) wird dem Netzwerkadministrator ein Werkzeug

angeboten, mit dem die Netzwerkeinstellungen der einzelnen Endgeräte automatisch, einheitlich und zentral konfigurierbar sind. Für die Nutzung von DHCP wird im Netzwerk mindestens ein DHCP-Server benötigt: auf dem Server wird ein Bereich von IP-Adressen festgelegt, aus dem einem anfragenden Teilnehmer (meist ein PC) eine zurzeit nicht benutzte IP-Adresse zugeteilt wird. Die Zuteilung ist in der Regel zeitlich begrenzt, wobei die Nutzungsdauer (Lease-Time) vom Netzwerkadministrator festgelegt werden kann. Diese automatische Vergabe von IP-Adressen ist in Netzwerken Standard.

Die automatische Vergabe der IP-Adresse ist von Vorteil, wenn Teilnehmer ihren Standort in den Netzwerken ändern. In diesem Fall erhalten die Teilnehmer immer eine für den Standort gültige IP-Adresse.

### **Ausschluss bestimmter IP-Adressen aus der DHCP-Konfiguration:**

Für Endgeräte, die nicht DHCP-fähig sind, hat der Netzwerkadministrator die Möglichkeit, einzelne IP-Adressen oder auch ganze Adressbereiche von der Vergabe durch DHCP auszuschließen. Die aus der automatischen Vergabe ausgeschlossenen IP-Adressen stehen dann den entsprechenden Geräten zur Verfügung.

Die Konfiguration muss in diesem Fall entweder am Endgerät oder durch Einsatz mitgelieferter Tools erfolgen.

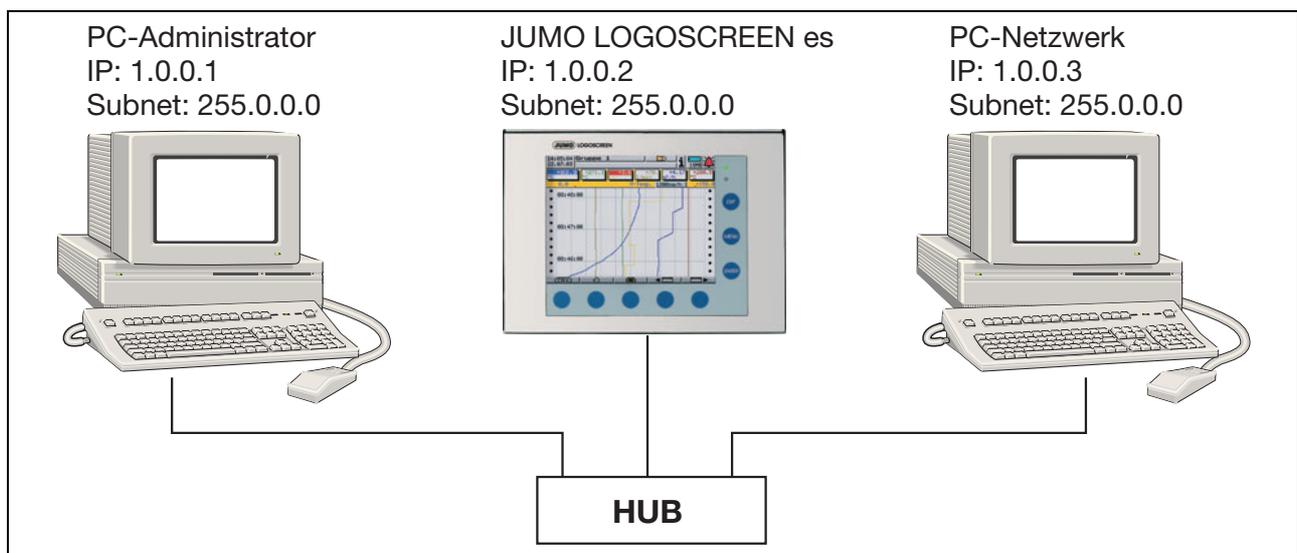
Bei einigen JUMO-Geräten mit ETHERNET-Schnittstelle und den später beschriebenen COM-Servern muss eine feste IP-Adresse vorliegen. Das bedeutet hier: der Netzwerkadministrator muss die entsprechende IP, wie beschrieben, von der Vergabe durch DHCP ausschließen. Die IP ist für das entsprechende Feldgerät reserviert und wird an diesem fest eingestellt.

### **5.3.4 Anbindung eines Teilnehmers an ETHERNET/Aufbau eines A-Klasse-Netzes**

Um ein Feldgerät an ein Firmennetz anzuschließen, müssen prinzipiell nur IP-Adresse und die Subnetmaske definiert werden.

Wir möchten aber ganz von vorne beginnen:

Es soll ein kleines Netzwerk, bestehend aus zwei PC's und einem Bildschirmschreiber JUMO LOGOSCREEN es, aufgebaut werden. Der Bildschirmschreiber kann eine Datenaufzeichnung von bis zu 36 Prozessgrößen (Temperaturen, Drücke etc.) durchführen und die Messwerte über seine ETHERNET-Schnittstelle zur Verfügung stellen.



**Abbildung 57: Prinzipieller Aufbau des Netzes**

## 5 Bussysteme

Die PC's sind mit Hilfe von Patch-Kabeln über einen HUB/SWITCH miteinander verbunden:

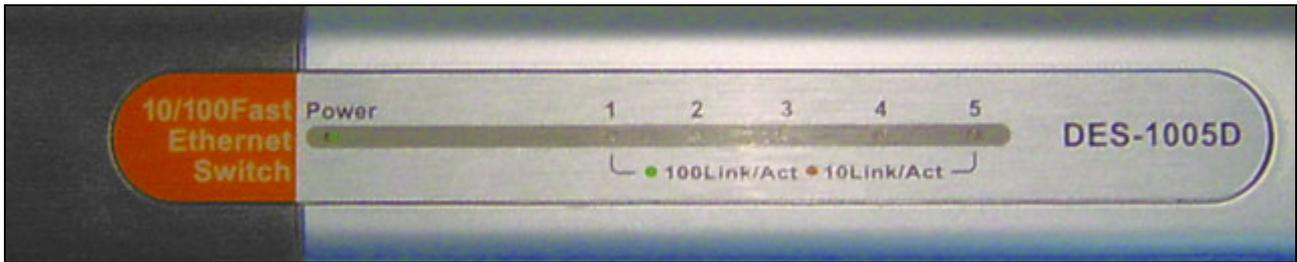


Abbildung 58: Frontansicht SWITCH



Abbildung 59: SWITCH-Rückansicht mit Patch-Kabeln für die drei Teilnehmer und Netzteil

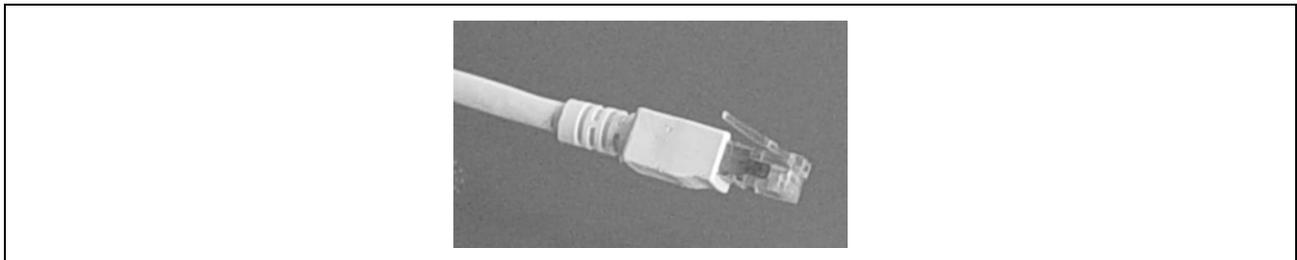


Abbildung 60: RJ45-Stecker der Patch-Kabel

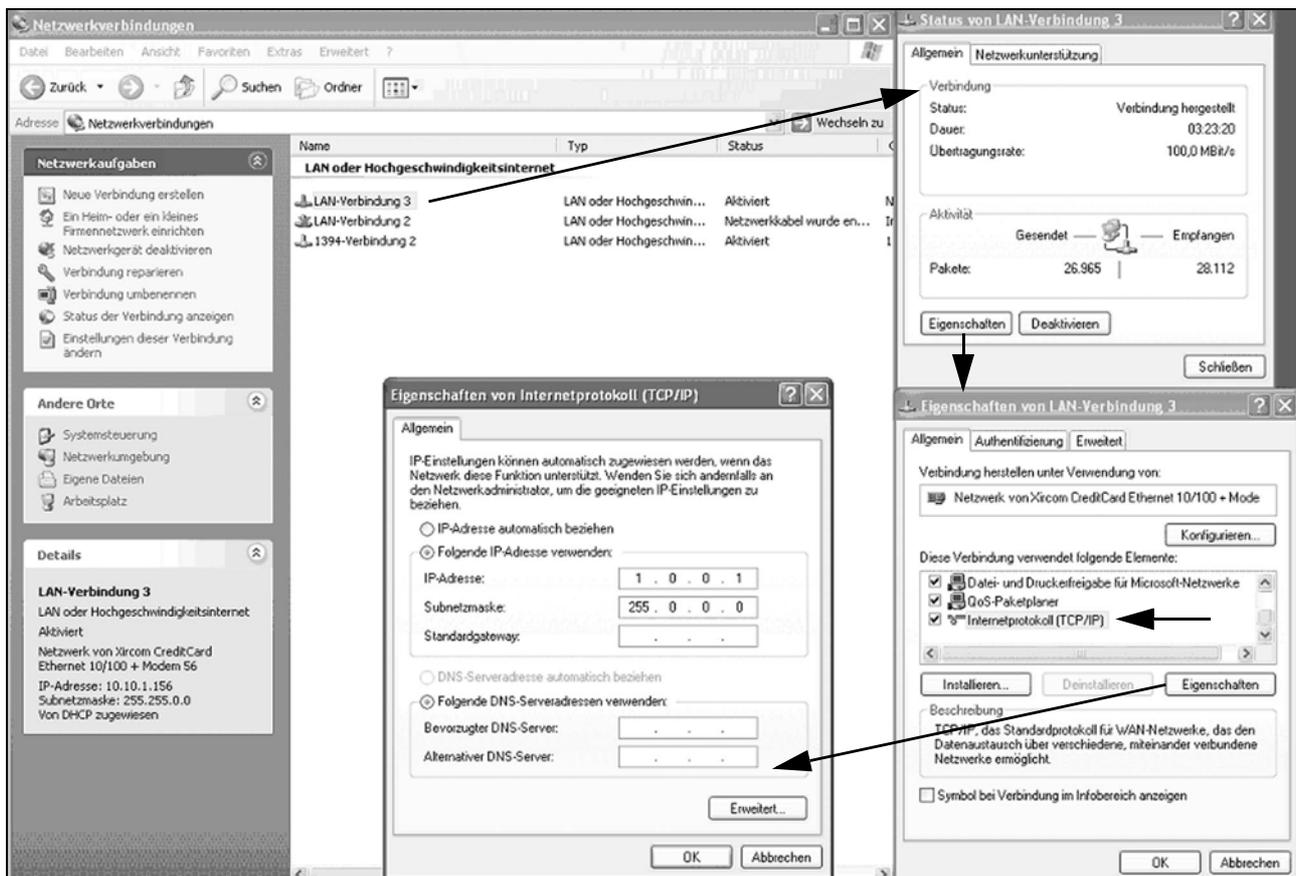
Nun kann die Vergabe der Subnetmaske und der IP-Adresse erfolgen:

Es soll ein A-Klasse-Netz geschaffen werden.

Wir definieren die Subnetmaske zu: 255.0.0.0 (Abbildung 57).

Wie zuvor beschrieben, ergibt sich die erste Netzadresse zu 1 und die erste Hostadresse zu X.0.0.1. Aus diesem Grund erhält der erste Teilnehmer die IP-Adresse 1.0.0.1.

Die Vergabe der IP-Adresse kann für die PC's unter „Systemsteuerung → Netzwerkverbindung“ erfolgen:



**Abbildung 61: Vergabe der IP-Adresse und Subnetmaske im Fall der PC's**

Im Fenster „Eigenschaften von Internetprotokoll (TCP/IP)“ kann nun die IP-Adresse und die Subnetmaske des PC's eingetragen werden. Üblicherweise besteht ETHERNET aus Teilnetzen, die über Router oder Bridges miteinander verbunden sind. Sendet ein Teilnehmer eine Nachricht für einen Endteilnehmer, welcher sich nicht im gleichen Netz befindet, wird die Nachricht direkt zum Router bzw. zur Bridge gesendet. Die IP-Adresse des Routers oder der Bridge ist unter Standardgateway anzugeben. Da unser Netz mit keinem weiteren Netz verbunden ist und folglich auch kein Gateway zum Einsatz kommt, kann der entsprechende Eintrag leer bleiben.

### Bemerkung:

„IP-Adresse automatisch beziehen“ ist dann auszuwählen, wenn im Netzwerk die IP-Adressen automatisch über DHCP vergeben werden (dies ist in der Regel der Fall, wir vergeben die Parameter jedoch fest).

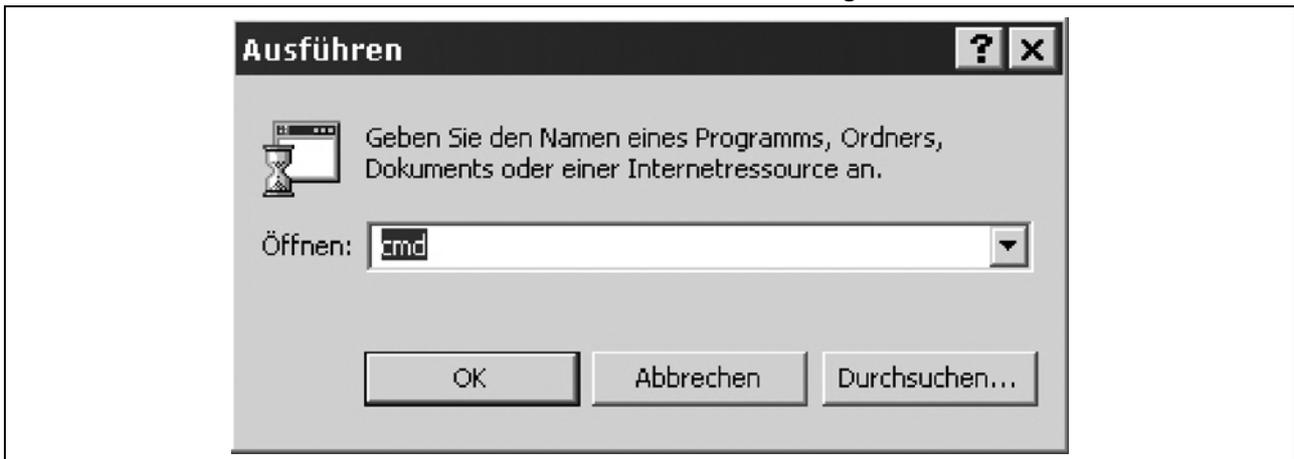
Für den LOGOSCREEN es kann die Vergabe der IP-Adresse und Subnetmaske am Gerät bzw. über ein Konfigurationsprogramm erfolgen.

Nach der Parametrierung der Schnittstellen stellt sich die Frage: Kommunizieren die Teilnehmer miteinander?

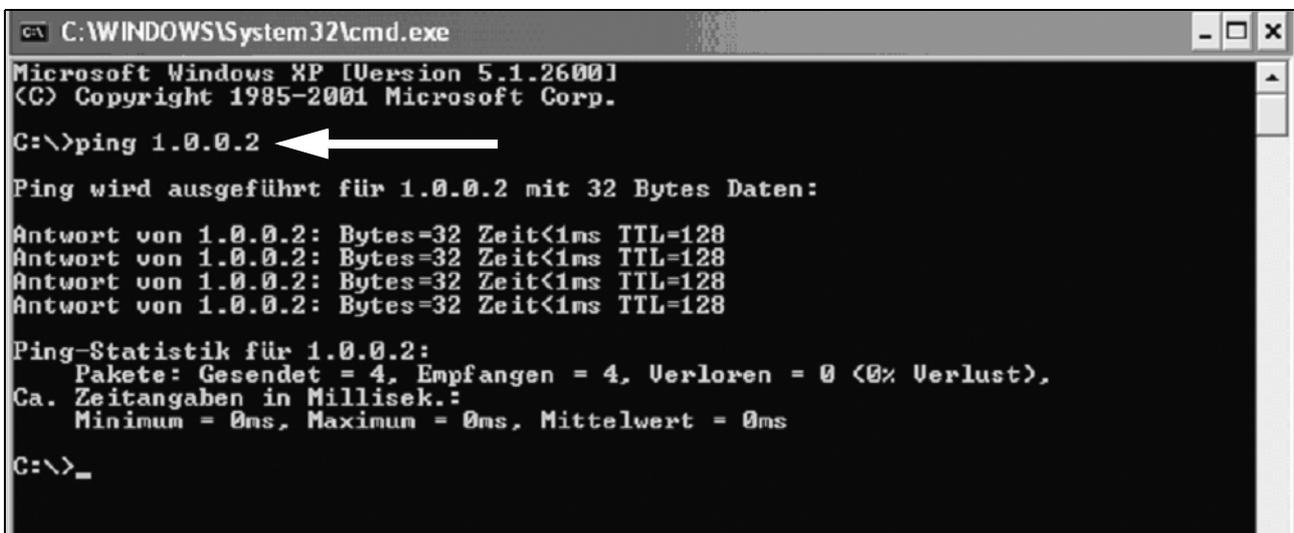
Im Folgenden ist eine Testmöglichkeit aufgezeigt:

## 5 Bussysteme

Nach Auswahl von „Ausführen“ im Windows-Startmenü und Eingabe von „cmd“ ...



... kann im DOS-Fenster mit „Ping IP“ die Kommunikation zu einem Teilnehmer getestet werden:



**Abbildung 62: Prüfen der Kommunikation**

Abbildung 62 zeigt die Antwort des Teilnehmers mit der IP-Adresse 1.0.0.2, wenn die Kommunikation steht. Können die Teilnehmer über PING erreicht werden, ist davon auszugehen, dass die Verbindung in Ordnung ist.

Der JUMO LOGOSCREEN es kann nun von den PC's angesprochen werden (in den entsprechenden Anwendungsprogrammen muss lediglich die IP-Adresse des Schreibers angegeben werden).

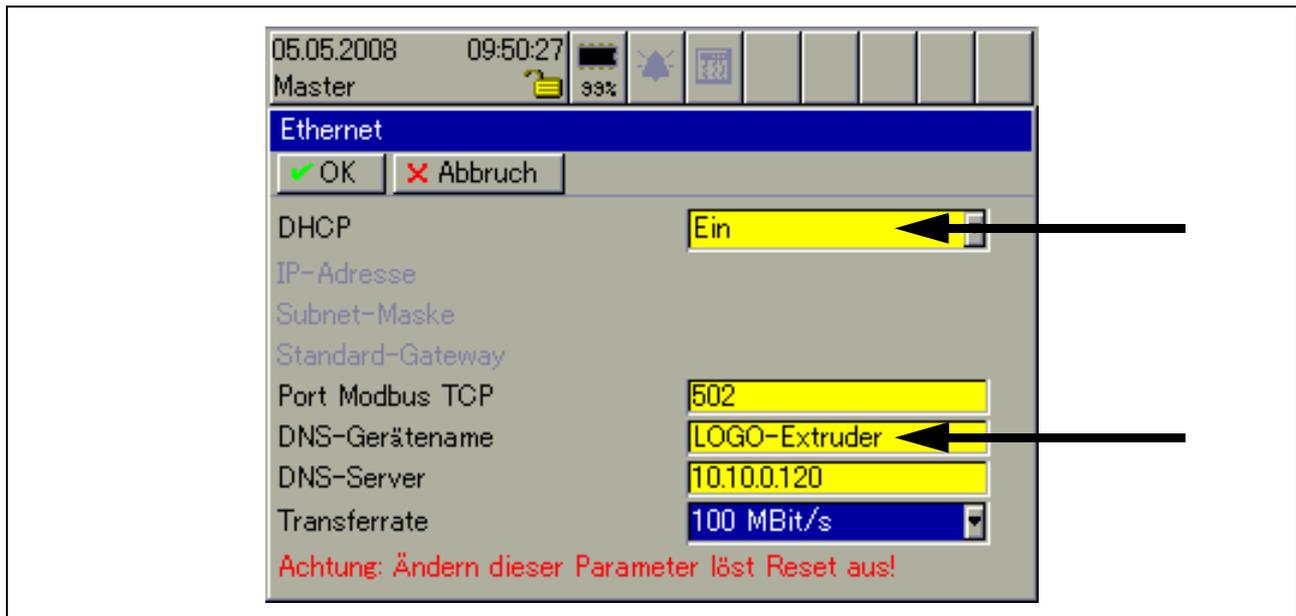
### Einbinden von JUMO-Feldgeräten in ein bestehendes Firmennetz

Die Einbindung eines entsprechenden ETHERNET-Gerätes ist prinzipiell sehr einfach: Benötigt wird eine freie Netzwerkdose. Die Verbindung zwischen dieser und dem ETHERNET-Gerät ist mit einem Patch-Kabel zu realisieren. Vom Netzwerkadministrator ist eine IP-Adresse anzugeben, die für dieses Gerät reserviert ist (wie bereits beschrieben, ist diese von der Vergabe durch DHCP auszuschließen). Die IP-Adresse, die Subnetmaske und das Standardgateway sind im JUMO-Gerät einzustellen. Nun kann das Gerät mit den entsprechenden Anwendungsprogrammen von jedem PC im Netzwerk angesprochen werden.

## 5.3.5 DNS

Hauptsächlich wird DNS (**D**omain **N**ame **S**ystem) zur Umsetzung von Domainnamen in IP-Adressen benutzt. Dies ist vergleichbar mit einem Telefonbuch, das die Namen der Teilnehmer in ihre Telefonnummer auflöst. DNS bietet somit eine Vereinfachung, weil Menschen sich Namen weitaus besser merken können als Zahlenkolonnen. So kann man sich einen DNS-Namen, z. B. LOGO-Extruder, in der Regel leichter merken als die dazugehörige IP-Adresse, z. B. 10.10.90.41.

JUMO-Geräten, wie beispielsweise dem LOGOSCREEN nt, kann ein DNS-Name vergeben werden:



**Abbildung 63: ETHERNET-Konfiguration mit DNS und DHCP**

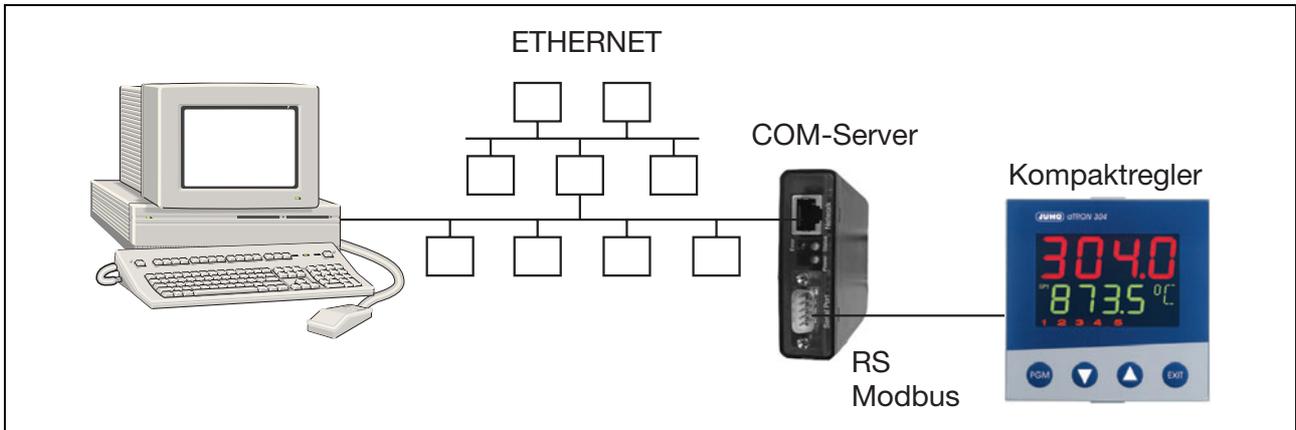
DNS ist meist nur mit DHCP-sinnvoll, aus diesem Grund wird dies am Schreiber aktiviert: Nach Einschalten des Schreibers erhält der Schreiber vom DHCP-Server seine IP-Konfiguration (IP-Adresse, Subnet und Standardgateway). Ein so genannter DNS-Server pflegt die Zuordnung der DNS-Namen zu den IP-Adressen (Die IP-Adresse des DNS-Servers ist anzugeben).

In Anwendungsprogrammen, welche auf den Schreiber zugreifen sollen, erfolgt ab sofort der Zugriff auf das Gerät über den DNS-Gerätenamen.

# 5 Bussysteme

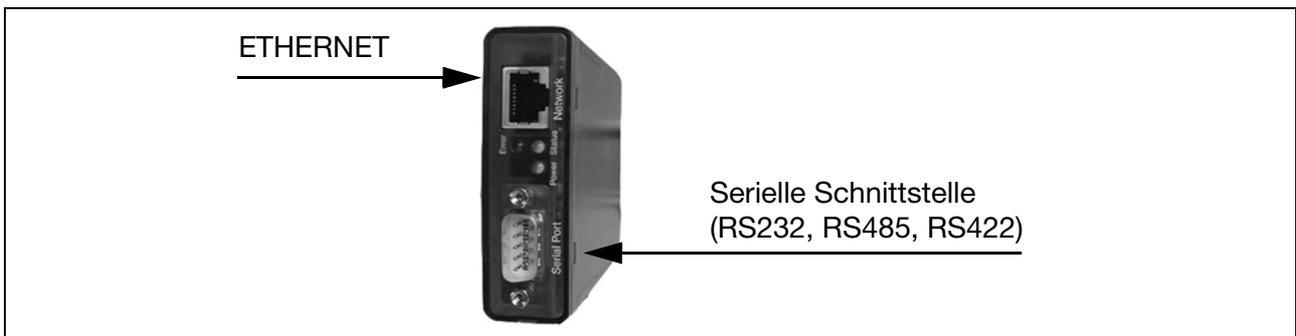
## 5.3.6 COM-Umlenkung

Mit der COM-Umlenkung kann die Anbindung von Modbus-Geräten an ETHERNET erfolgen:



**Abbildung 64: COM-Server im Netzwerk**

Bei der COM-Umlenkung wird ein COM-Server (z. B. COM-Server Highspeed Industry, siehe WuT.de) im Netzwerk platziert. Diesem wird eine IP-Adresse vergeben und über ein Anwendungsprogramm (COM-Umlenkung) für einen PC eine virtuelle COM-Schnittstelle definiert. Alle Daten, die nun vom PC aus an dessen virtuelle COM-Schnittstelle gesendet werden, gelangen über ETHERNET an die serielle Schnittstelle des COM-Servers:



**Abbildung 65: COM-Server Highspeed Industry**

Mit dem genannten COM-Server können die Schnittstellen RS232, RS485 und RS422 realisiert werden. Die Auswahl der Schnittstelle erfolgt über DIP-Switches im COM-Server.

### Konfiguration des COM-Servers

Mit den Werkseinstellungen des COM-Servers hat dieser noch keine gültige IP-Adresse. Mit ARP (**A**dress **R**esolution **P**rotocol) kann in diesem Fall einmalig (!) eine IP-Adresse vergeben werden:

ARP kann in Windows unter „Start → Ausführen“ gestartet werden. Die Anweisung für einen COM-Server gestaltet sich z. B. wie folgt:

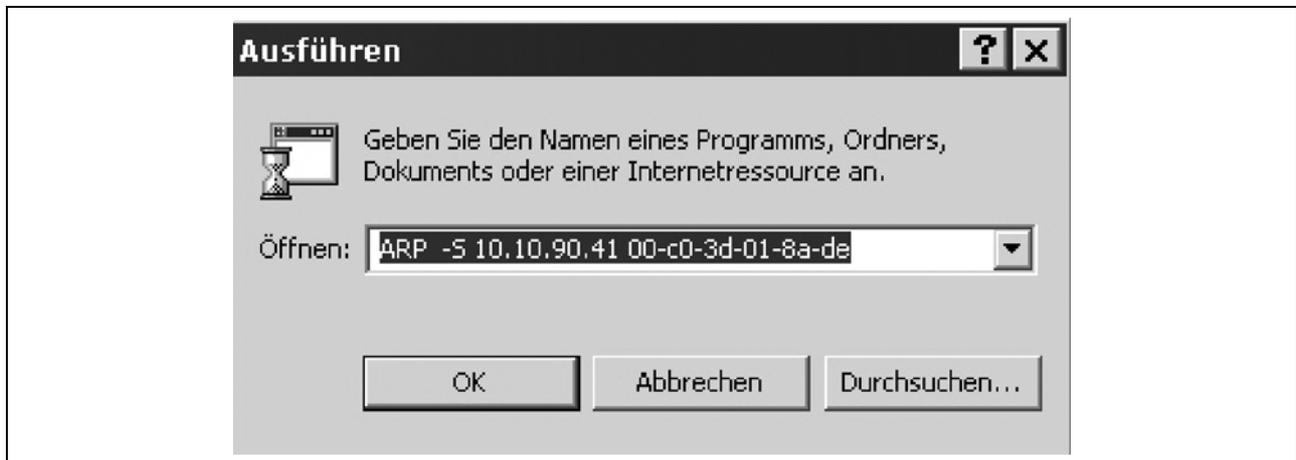


Abbildung 66: Zuweisung einer IP-Adresse

<b>-S</b>	IP-Adresse der physischen Adresse zuordnen
<b>10.10.90.41</b>	IP-Adresse, die dem COM-Server zugeteilt werden soll. Die IP-Adresse muss vom Netzwerkadministrator genannt werden.
<b>00-c0-3d-01-8a-de</b>	Physische Adresse (MAC-Adresse) des COM-Servers. Die Adresse befindet sich am COM-Server.

Nun muss die Konfiguration des COM-Servers hinsichtlich des Formates der zu übertragenden Daten (Baudrate, Stoppbits etc.) mit Telnet erfolgen. Telnet ist ein Textfenster bzw. ein textorientiertes Programm, über das ein Endgerät im Netzwerk fernkonfiguriert werden kann.

Telnet wird durch Eingabe von „Telnet IP 1111“ gestartet (1111 = Port-Nummer des COM-Servers). Möchten wir von einem PC aus einen COM-Server mit der IP 10.10.90.41 konfigurieren, gestaltet sich der Start von Telnet wie folgt:

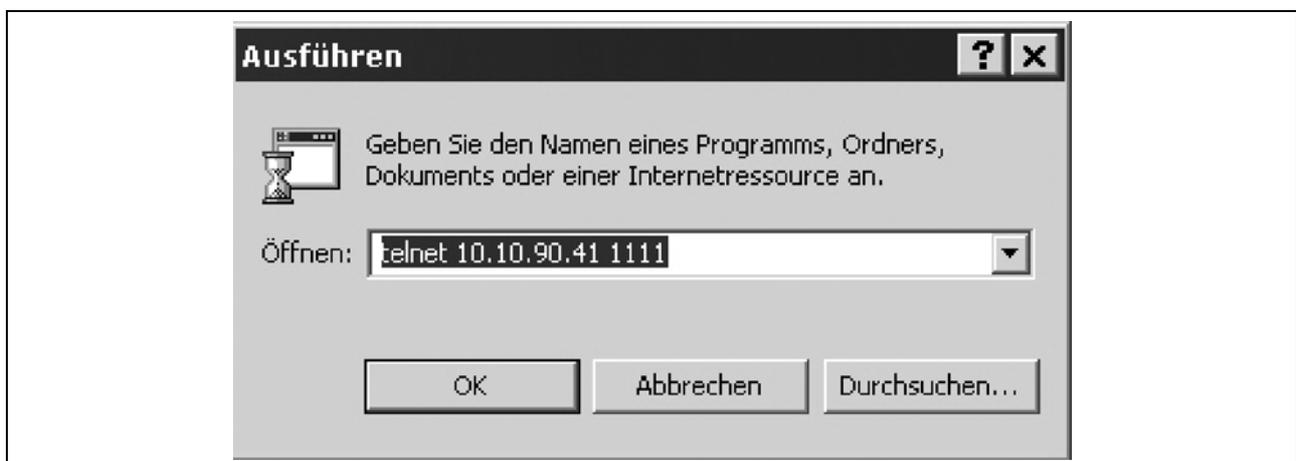
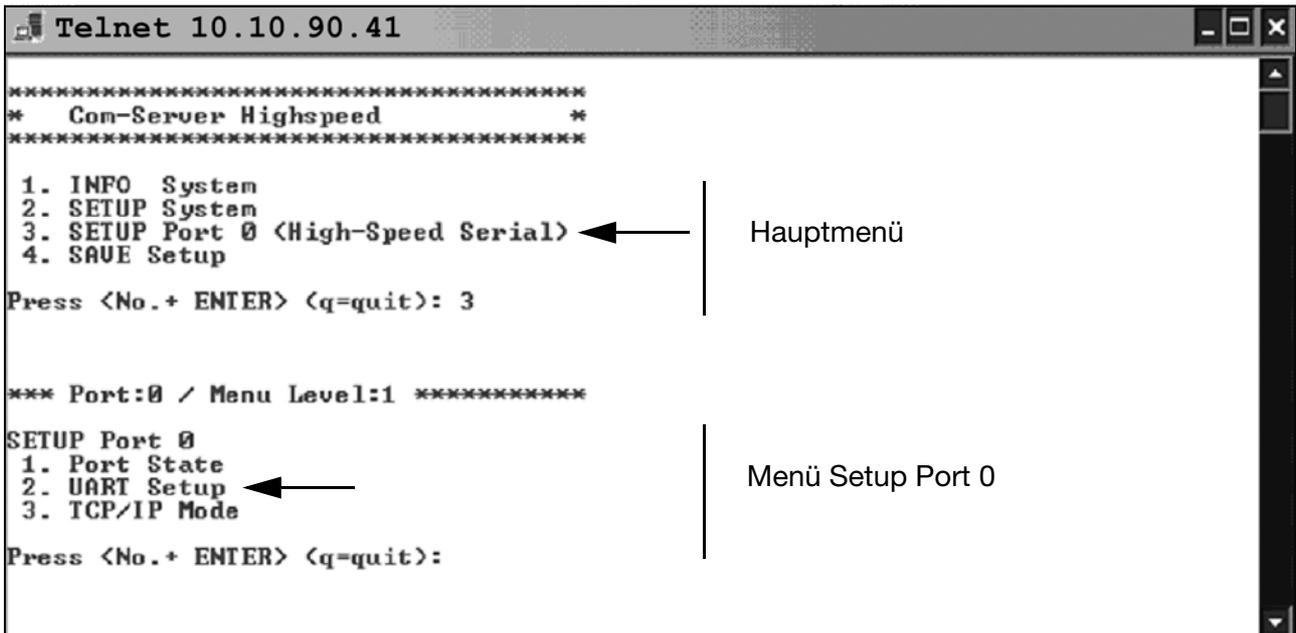


Abbildung 67: Start von Telnet

# 5 Bussysteme



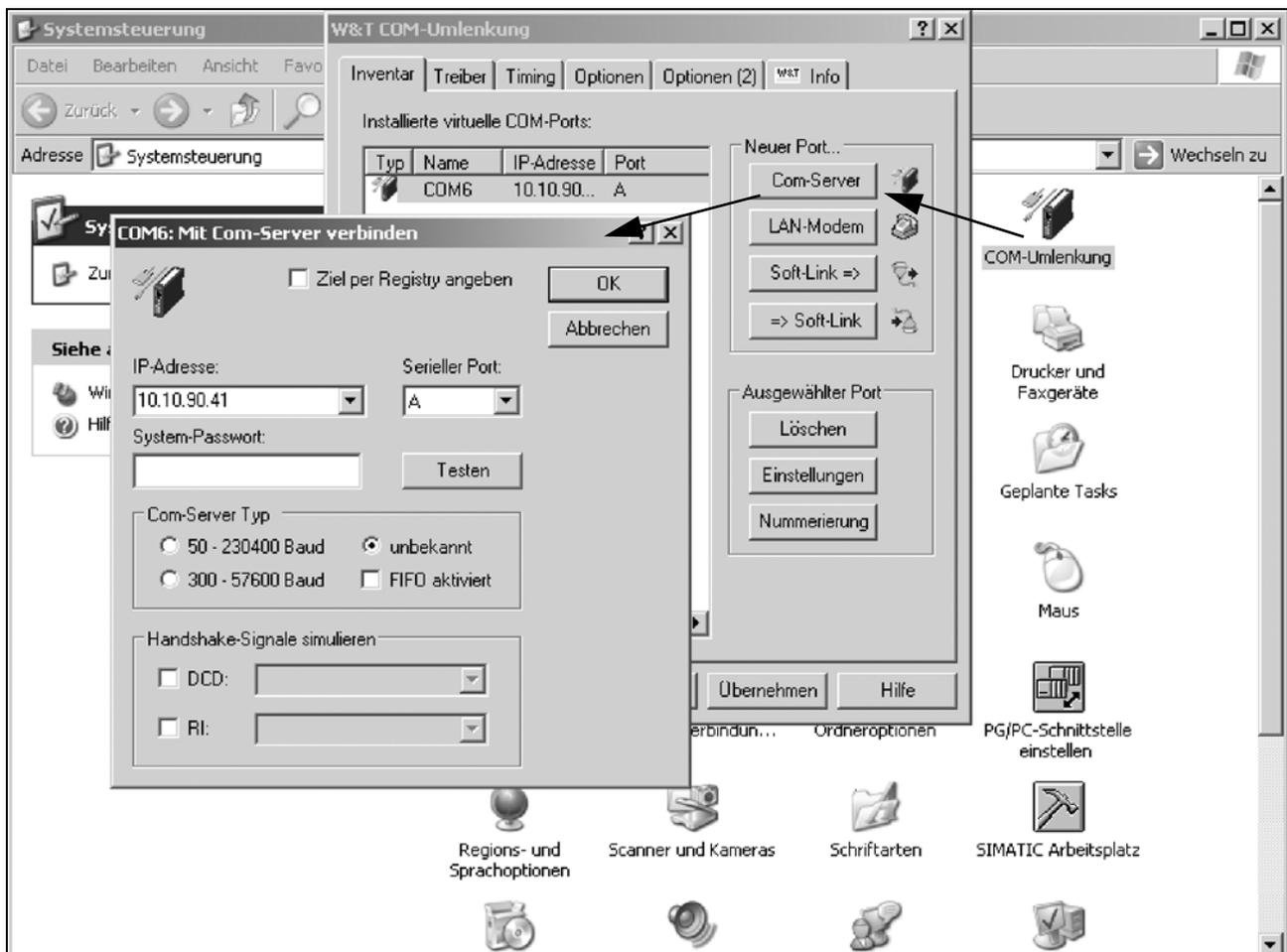
**Abbildung 68: Hauptmenü Telnet**

Nach dem Start erscheint das Hauptmenü von Telnet. Nach Anwahl des Menüs „SETUP Port 0“ kann in dessen Untermenü „UART Setup“ die serielle Schnittstelle des COM-Servers eingestellt werden (Baudrate, Parity, Stoppbits etc.). Die Einstellung muss mit der Konfiguration der Geräte auf der Modbus-Seite (RS-232-, RS-485- oder RS-422-Schnittstelle) übereinstimmen.

Der COM-Server unterstützt BootP. Über BootP kann ein Endgerät ebenfalls eine IP-Adresse bei einem DHCP-Server erfragen. Für Anwendungen von COM-Servern in Verbindung mit JUMO-Geräten empfehlen wir, diese Funktion mit Hilfe von Telnet abzuschalten.

Mit Telnet ist weiterhin die nachträgliche Änderung der IP-Adresse möglich.

Nun kann der COM-Server installiert werden. Nach der Installation ist am PC in der Systemsteuerung der Eintrag „COM-Umlenkung“ ersichtlich:



**Abbildung 69: COM-Server in der Systemsteuerung**

In der COM-Umlenkung kann nun die IP-Adresse des COM-Servers angegeben werden. Alle Daten, die an die virtuelle COM-Schnittstelle (im Beispiel COM6) gesendet werden, gehen über ETHERNET an die serielle Schnittstelle des COM-Servers.

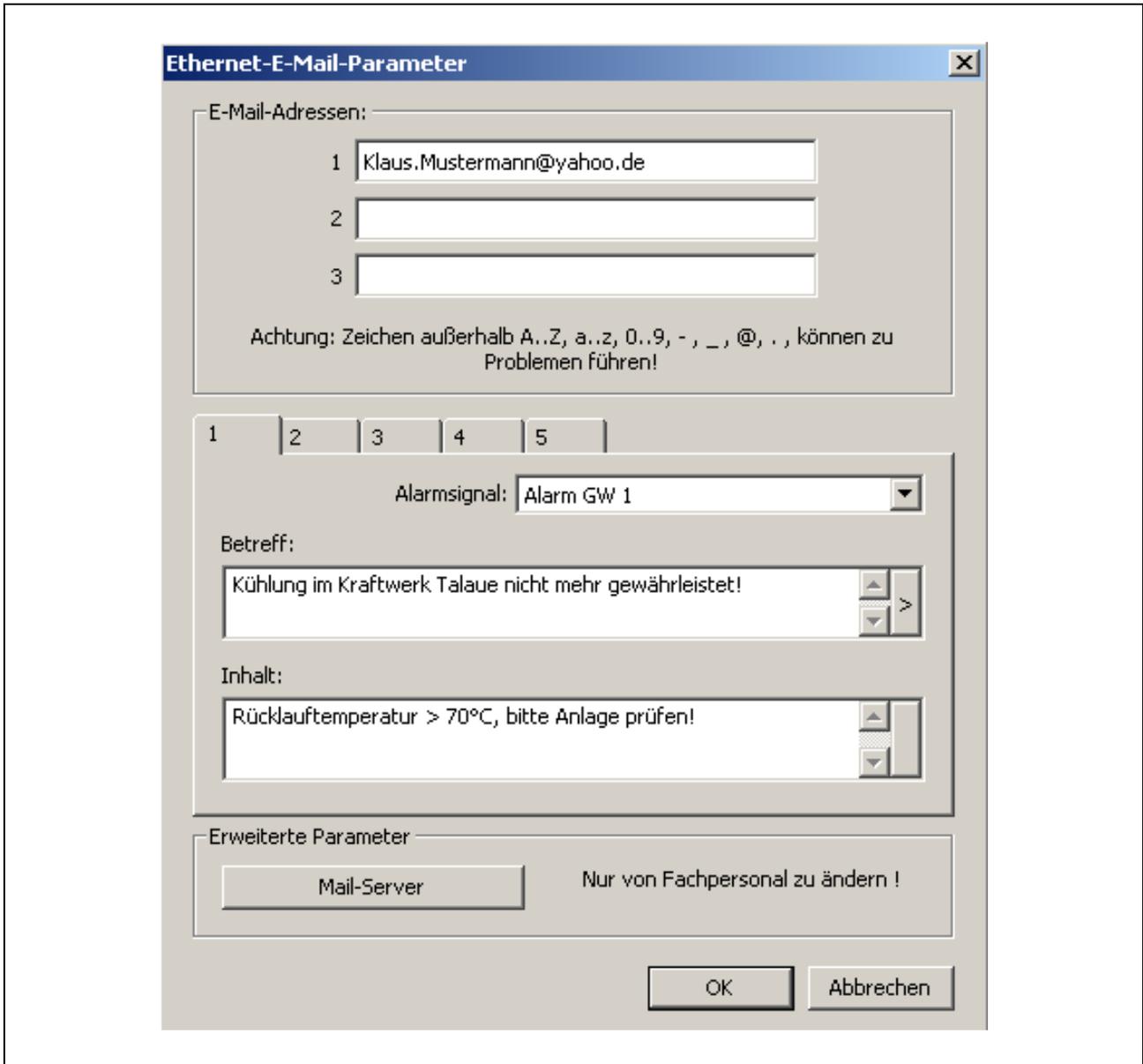
JUMO hat den COM-Server in einige Anwendungsprogramme eingearbeitet. In den Programmen kann direkt eingegeben werden, dass sich ein Feldgerät an einer seriellen Schnittstelle eines COM-Servers befindet, für welchen man wiederum die IP-Adresse angeben muss. In diesem Fall entfällt die Installation der COM-Umlenkung.

# 5 Bussysteme

## 5.3.7 E-Mail-Versand durch JUMO-Geräte

Mit einigen JUMO-Geräten können E-Mails, ausgelöst durch Binärsignale, versendet werden. Binärsignale werden beispielsweise durch Überschreitung von Grenzwerten (Temperaturen, Drücke etc.) gesteuert.

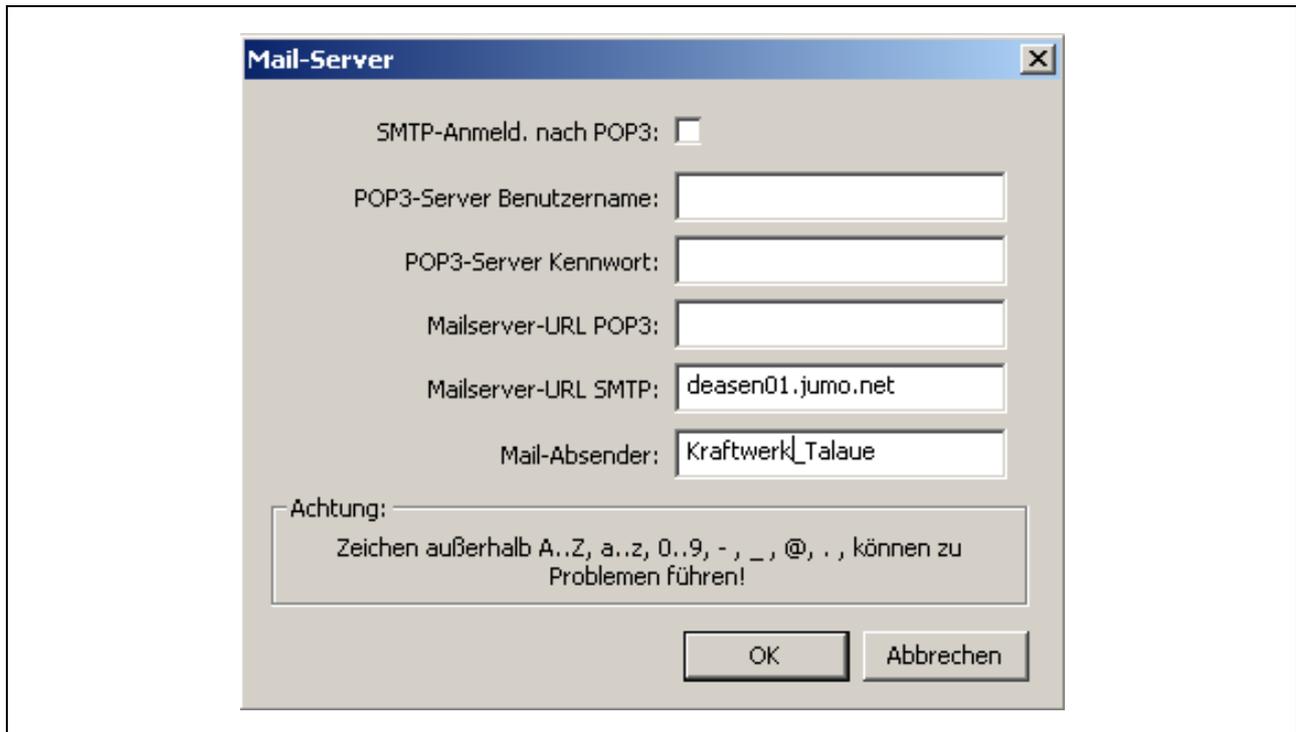
Beim Bildschirmschreiber LOGOSCREEN nt können aufgrund fünf unterschiedlicher Binärsignale Meldungen an 3 unterschiedliche Adressaten gesendet werden:



**Abbildung 70: Konfiguration E-Mail-Versand 1**

Ein Gerät mit der in Abbildung 70 dargestellten Konfiguration wird bei aktivem Alarmsignal (Alarm GW 1 kann z. B. bedeuten, dass die Temperatur an Eingang 1 > 70°C beträgt) eine E-Mail an die angegebene Adresse senden.

Ist im vorhandenen Netzwerk ein Mailserver vorhanden, muss weiterhin die Mailserver-Adresse und der Absender des Gerätes angegeben werden:



**Abbildung 71: Konfiguration E-Mail-Versand 2**

Die Einstellungen zu POP3 (**P**ost **O**ffice **P**rotokoll) müssen vorgenommen werden, wenn der Schreiber die E-Mails direkt über das Internet absetzt (POP3 ist ein Übertragungsprotokoll, über welches eine Verbindung zu einem E-Mail-Server aufgebaut wird).

# 5 Bussysteme

## 5.3.8 Web-Server

Ein Web-Server ist ein Endgerät, welches Dokumente an Webbrowser überträgt. Als Webserver bezeichnet man das Endgerät mit Websoftware oder nur die Websoftware. Der LOGOSCREEN nt ist das erste Gerät bei JUMO, welches einen Webserver beinhaltet. Durch JUMO wurden dynamische HTML-Seiten im Bildschirmschreiber bereitgestellt, über welche der Schreiber weltweit fernbedient werden kann.

Zwecks Verbindung wird die IP-Adresse oder der DNS-Name des Bildschirmschreibers im Browser eingegeben. Wenn gewünscht, können vier Schreiber gleichzeitig oder ein Schreiber in vier Ansichten mit dem Standard-Browser dargestellt werden:

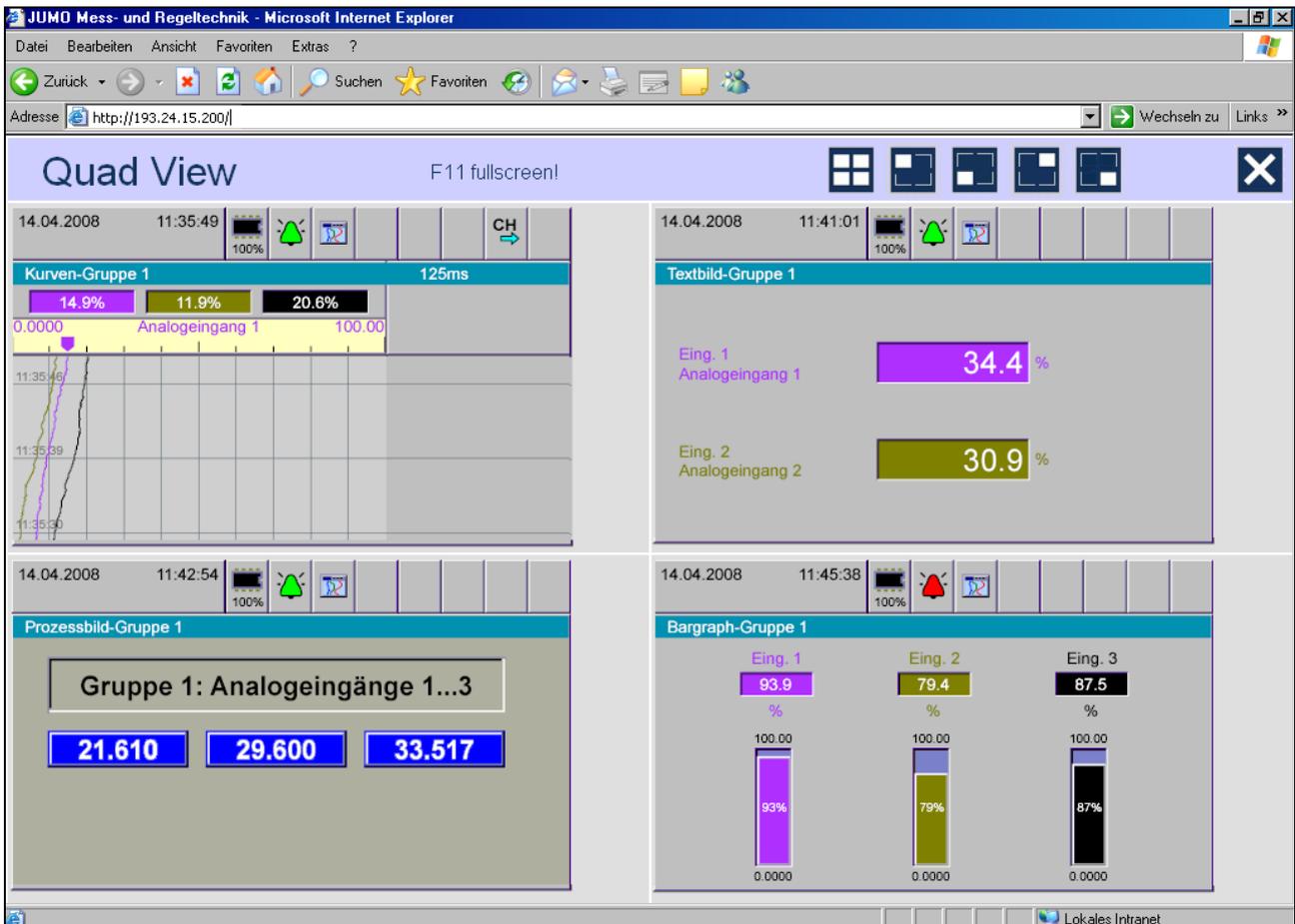


Abbildung 72: Vier Ansichten eines Schreibers über Internet-Browser (Quad-View)

## 5.4 CAN und CANopen

### 5.4.1 CAN

CAN ist ein sehr sicheres Bussystem, welches 1986 von der Firma Robert Bosch entwickelt wurde. Ursprünglich wurde CAN für den Einsatz in Kraftfahrzeugen entwickelt, findet aber auch verstärkt im industriellen Bereich Einsatz.

CAN steht für **C**ontroller **A**rea **N**etwork und ist ein serieller Datenbus. Bei CAN werden kurze Nachrichten zerstörungsfrei übertragen. Die zerstörungsfreie Übertragung wird unter anderem durch die Adressierung der Nachrichten ermöglicht.

Das Signal wird über zwei Leitungen (CAN\_H und CAN\_L) übertragen. Die Teilnehmer werden über Linientopologie über eine geschirmte Zweidrahtleitung miteinander verbunden und die Leitung am ersten und letzten Teilnehmer mit einem Abschlusswiderstand versehen:

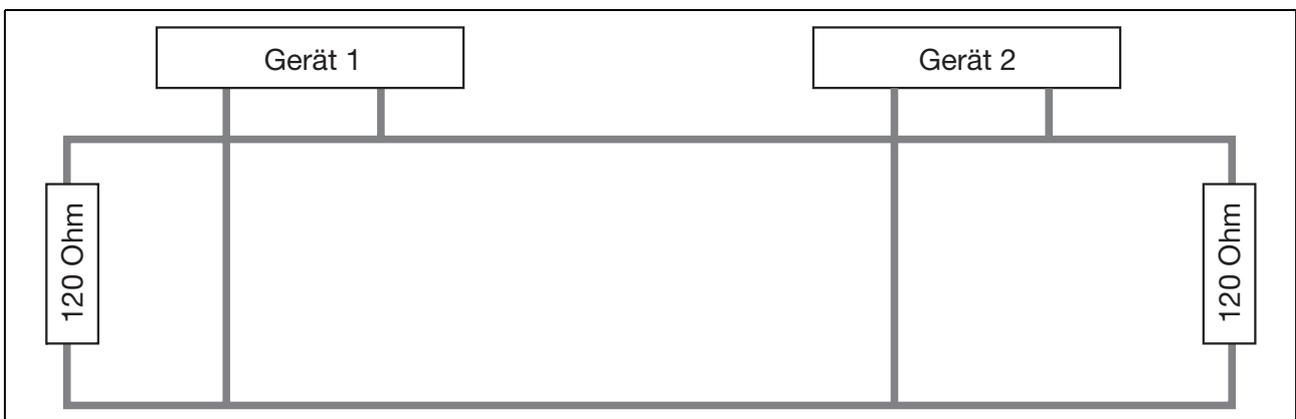


Abbildung 73: Anschluss der Teilnehmer bei CAN

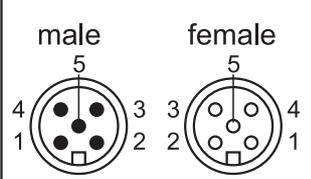
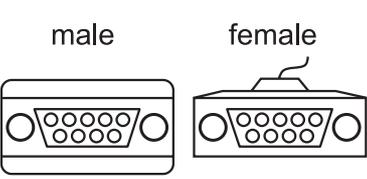
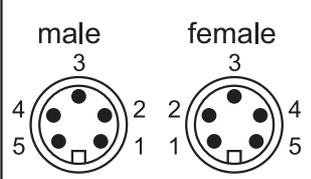
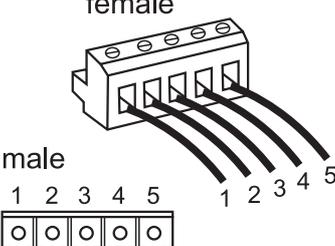
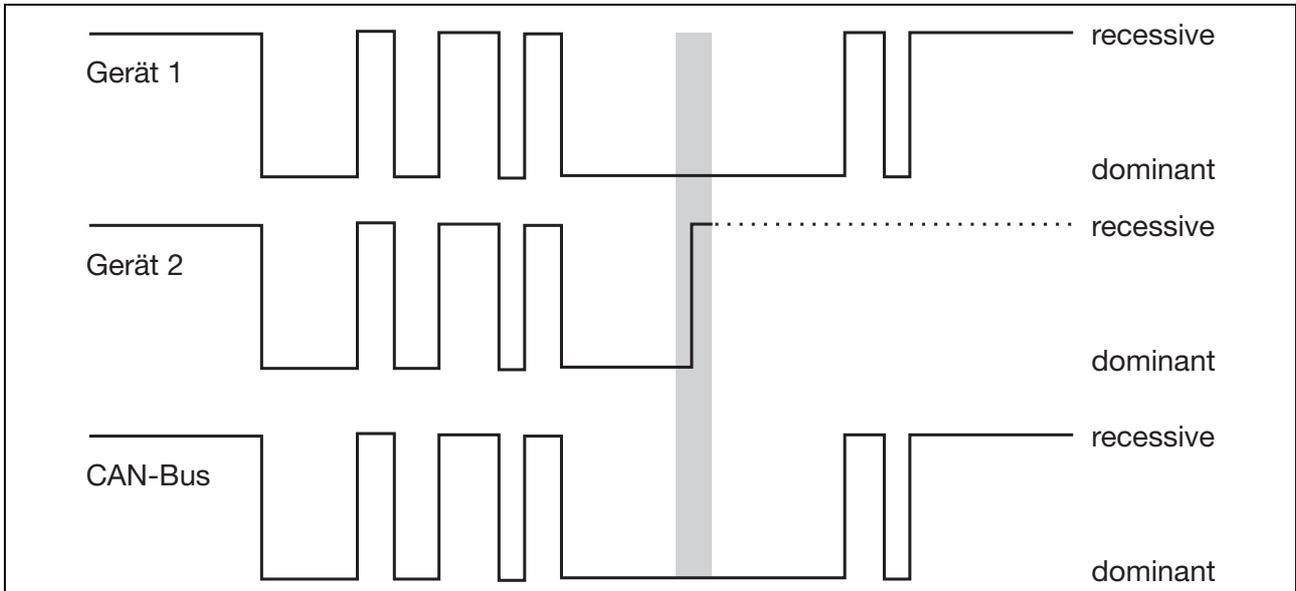
<p>Rundstecker M 12x1; 5-polig nach IEC 60 947-5-2</p>  <table border="1"> <thead> <tr> <th>PIN</th> <th>SIGNAL</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>CAN_GND</td> </tr> <tr> <td>2</td> <td>V+</td> </tr> <tr> <td>3</td> <td>V-</td> </tr> <tr> <td>4</td> <td>CAN_H</td> </tr> <tr> <td>5</td> <td>CAN_L</td> </tr> </tbody> </table>	PIN	SIGNAL	1	CAN_GND	2	V+	3	V-	4	CAN_H	5	CAN_L	<p>9-poliger D-Sub-Stecker nach DIN 41 652</p>  <table border="1"> <thead> <tr> <th>PIN</th> <th>SIGNAL</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>-</td> </tr> <tr> <td>2</td> <td>CAN_L</td> </tr> <tr> <td>3</td> <td>CAN_GND</td> </tr> <tr> <td>4</td> <td>-</td> </tr> <tr> <td>5</td> <td>CAN_Schirm</td> </tr> <tr> <td>6</td> <td>GND</td> </tr> <tr> <td>7</td> <td>CAN_H</td> </tr> <tr> <td>8</td> <td>-</td> </tr> <tr> <td>9</td> <td>CAN_V+</td> </tr> </tbody> </table>	PIN	SIGNAL	1	-	2	CAN_L	3	CAN_GND	4	-	5	CAN_Schirm	6	GND	7	CAN_H	8	-	9	CAN_V+
PIN	SIGNAL																																
1	CAN_GND																																
2	V+																																
3	V-																																
4	CAN_H																																
5	CAN_L																																
PIN	SIGNAL																																
1	-																																
2	CAN_L																																
3	CAN_GND																																
4	-																																
5	CAN_Schirm																																
6	GND																																
7	CAN_H																																
8	-																																
9	CAN_V+																																
<p>Rundstecker ANSI/B93.55M-1981</p>  <table border="1"> <thead> <tr> <th>PIN</th> <th>SIGNAL</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>CAN_Schirm</td> </tr> <tr> <td>2</td> <td>V+</td> </tr> <tr> <td>3</td> <td>V-</td> </tr> <tr> <td>4</td> <td>CAN_H</td> </tr> <tr> <td>5</td> <td>CAN_L</td> </tr> </tbody> </table>	PIN	SIGNAL	1	CAN_Schirm	2	V+	3	V-	4	CAN_H	5	CAN_L	<p>Schraub-Steckklemme</p>  <table border="1"> <thead> <tr> <th>PIN</th> <th>SIGNAL</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>CAN_GND</td> </tr> <tr> <td>2</td> <td>CAN_L</td> </tr> <tr> <td>3</td> <td>CAN_Schirm</td> </tr> <tr> <td>4</td> <td>CAN_H</td> </tr> <tr> <td>5</td> <td>CAN_V+</td> </tr> </tbody> </table>	PIN	SIGNAL	1	CAN_GND	2	CAN_L	3	CAN_Schirm	4	CAN_H	5	CAN_V+								
PIN	SIGNAL																																
1	CAN_Schirm																																
2	V+																																
3	V-																																
4	CAN_H																																
5	CAN_L																																
PIN	SIGNAL																																
1	CAN_GND																																
2	CAN_L																																
3	CAN_Schirm																																
4	CAN_H																																
5	CAN_V+																																

Abbildung 74: Auswahl von Steckern für CAN

# 5 Bussysteme

Wie aus Abbildung 74 ersichtlich ist, erfolgt bei CAN über das Anschlusskabel häufig auch zugleich die Spannungsversorgung der Teilnehmer (V+ und V-).

Bei CAN kann jeder Teilnehmer von sich aus mit dem Senden beginnen. Alle Nachrichten erhalten eine Priorität. Eine Priorität ist in jedem System nur einmal vorhanden. Beginnen nun zwei Teilnehmer mit dem Senden, setzt sich die Nachricht mit der höheren Priorität durch. In Abbildung 75 beginnen zwei Teilnehmer gleichzeitig, eine Nachricht zu senden. Während des Sendens hören die Teilnehmer auf den Bus, ob die Information der von ihnen gesendeten Information entspricht:



**Abbildung 75: CSMA-CA-Verfahren**

Gerät 1 und 2 beginnen gleichzeitig mit dem Senden der Nachricht. Bis zur Markierung ist die Information gleich - die Daten auf dem Bus entsprechen der Nachricht, welche von den beiden Geräten gesendet wurde. An der Markierung unterscheidet sich nun die Information (Gerät 1 sendet Pegel 0, Gerät 2 sendet Pegel 1). Pegel 0 ist der dominante Pegel, dieser liegt am Bus an, wenn nur ein Teilnehmer diesen Pegel sendet. Gerät 2 stellt fest, dass sich auf dem Bus eine andere Information (Pegel 0) als das gesendete Bit (Pegel 1) befindet. Gerät 2 zieht sich vom Bus zurück und sendet seine Information später erneut. In unserem Beispiel sendet Gerät 1 die Nachricht mit der höheren Priorität, diese setzt sich durch. Wie bereits in 4 „Zugriffsverfahren und Übertragungsmedien“ beschrieben, handelt es sich hierbei um das CSMA-CA-Verfahren.

Die Priorität der Nachricht wird im so genannten Identifier festgelegt. Abbildung 76 zeigt den Aufbau eines CAN-Protokolls:

Identifier	Datenfeld (0 ... 8 Bytes)	CRC-Feld
------------	---------------------------	----------

**Abbildung 76: Protokollaufbau eines CAN-Telegramms**

Der Identifier (COB-ID) bei JUMO-Sensoren besteht aus 11 Bit. In diesem Format können  $2^{11} = 2048$  unterschiedliche Nachrichten adressiert werden. Neben diesem Standardformat existiert noch ein erweitertes Format mit 29 Bit. In diesem Format können  $2^{29} = 536.870.912$  unterschiedliche Nachrichten adressiert werden.

Dem Identifier folgt die eigentliche Information, wie z. B. ein Druck- oder Temperaturmesswert. Hier ist zu sehen, dass in einem Telegramm immer sehr geringe Datenmengen übertragen werden (maximal 8 Bytes). Dem Datenfeld folgt das CRC-Feld. Hier wird die Checksumme übertragen.

Diese Prüfung ermöglicht, dass bei CAN innerhalb eines Protokolls bis zu sechs falsch übertragene Bits sicher erkannt werden (Hammingdistanz  $HD = 6$ ).

Die maximale Übertragungsgeschwindigkeit von CAN beträgt 1 Mbit/s bei 25m Busausdehnung. Bei Übertragungsraten von 20 kbit/s können bis zu 2500m Netzausdehnung erreicht werden.

Um die Kommunikation zwischen Geräten über CAN zu ermöglichen, sind verschiedene Festlegungen notwendig. An oberster Stelle steht hier die Zuteilung der im Netz verfügbaren CAN-Identifizier an die einzelnen Geräte. Weiterhin müssen Festlegungen in Bezug auf die übertragenen Dateninhalte getroffen werden. Wichtig ist dabei vor allem die Interpretation der Dateninhalte. Und nicht zuletzt ist auch eine übergeordnete Geräteüberwachung notwendig. All dies ist in der Anwendungsschicht zusammengefasst.

Protokolle, die die genannten Punkte klar definieren, sind CANopen, DeviceNet, CANKingdom, Smart Distributed System, In-Vehicle Networking, J1939.

### 5.4.2 CANopen

Wir wollen uns nun mit dem in Deutschland und Europa verbreiteten CANopen-Standard beschäftigen.

Wir beschreiben die Arbeitsweise eines CANopen-Knotens und zeigen, wie eine solche Komponente konfiguriert wird. JUMO liefert Druck- und Temperaturmessumformer mit CANopen. Die Erklärungen werden am Beispiel des Temperaturmessumformers JUMO CANtrans T verdeutlicht.

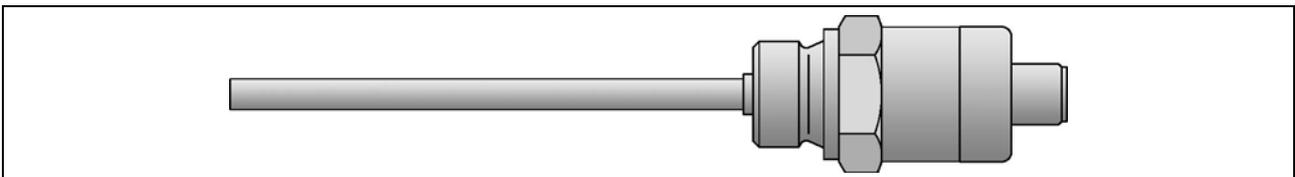


Abbildung 77: JUMO CANtrans T

Bei CANopen werden max. 127 Knoten in Linie miteinander verbunden (wie bei CAN beschrieben).

Die Konfiguration der Teilnehmer wird mit einem herstellerunabhängigen Anwenderprogramm erstellt und über den Bus zum Teilnehmer gesendet. Hierfür ist ein Interface zur Verbindung zwischen dem PC und den CAN-Knoten notwendig.

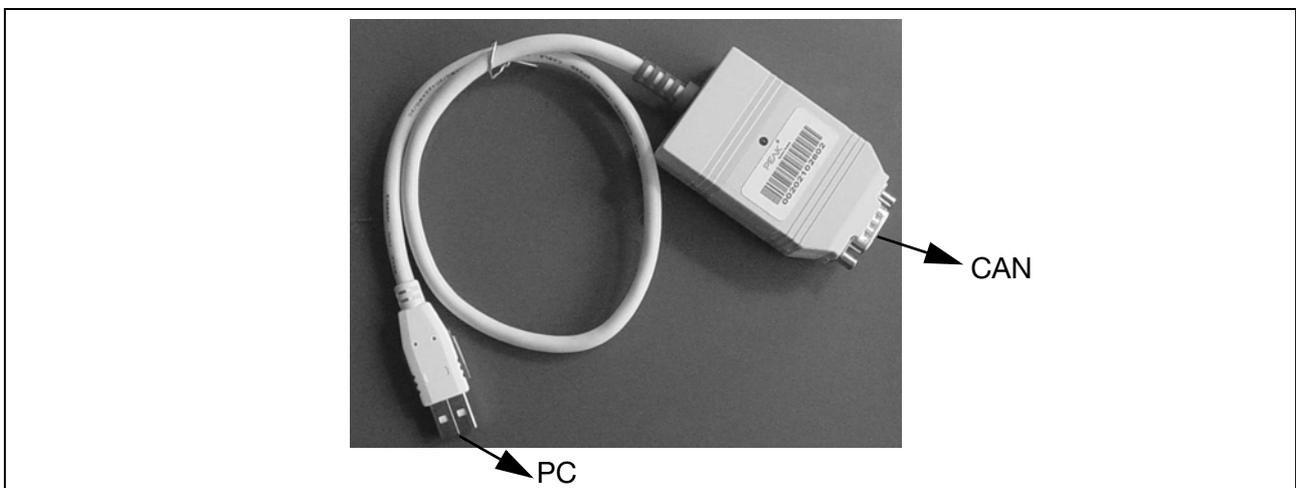


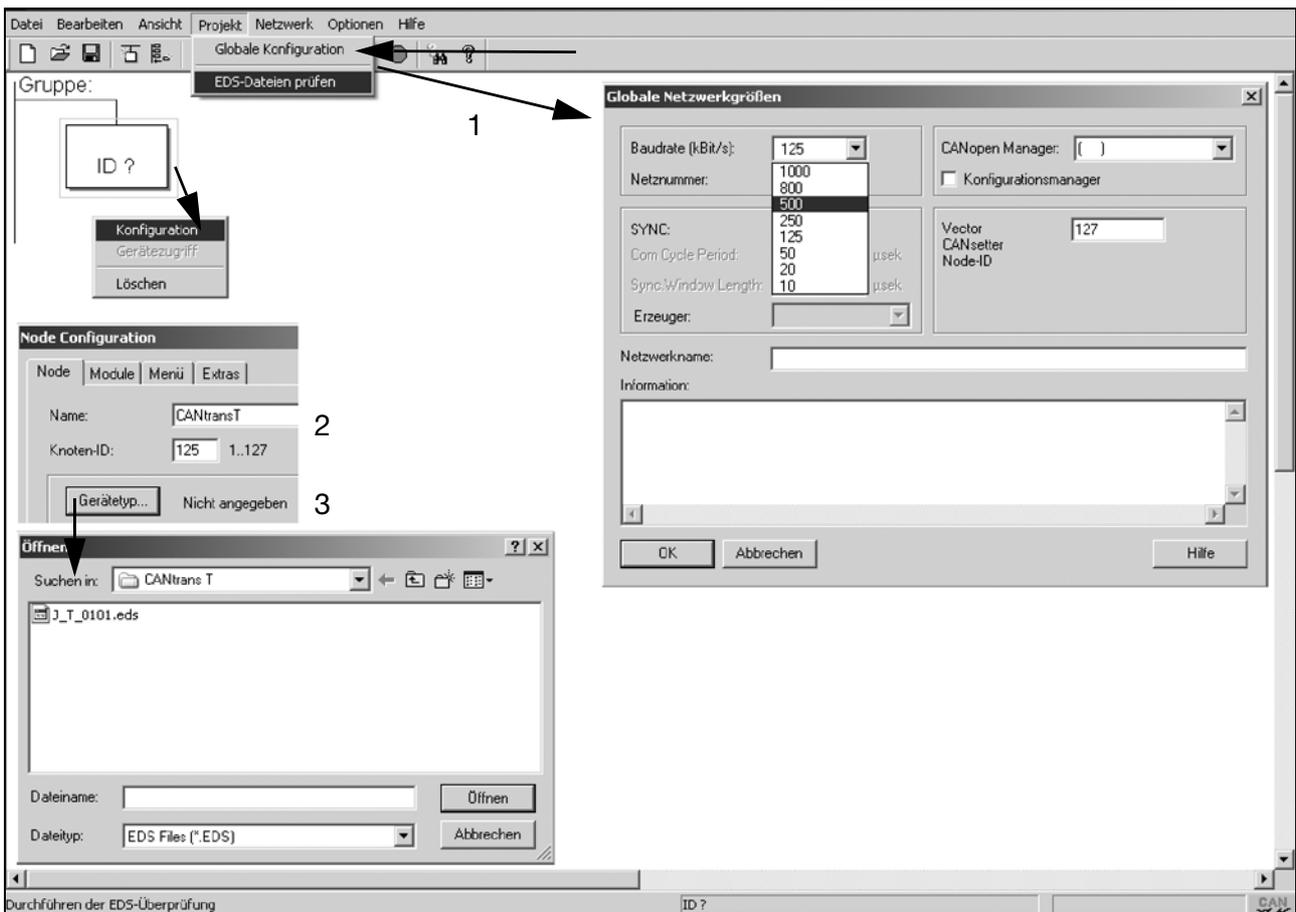
Abbildung 78: CAN-Interface

# 5 Bussysteme

Abbildung 78 zeigt ein CAN-Interface, das an die USB-Schnittstelle des PC's angeschlossen wird und auf der Gegenseite den Zugriff auf den CAN-Bus ermöglicht.

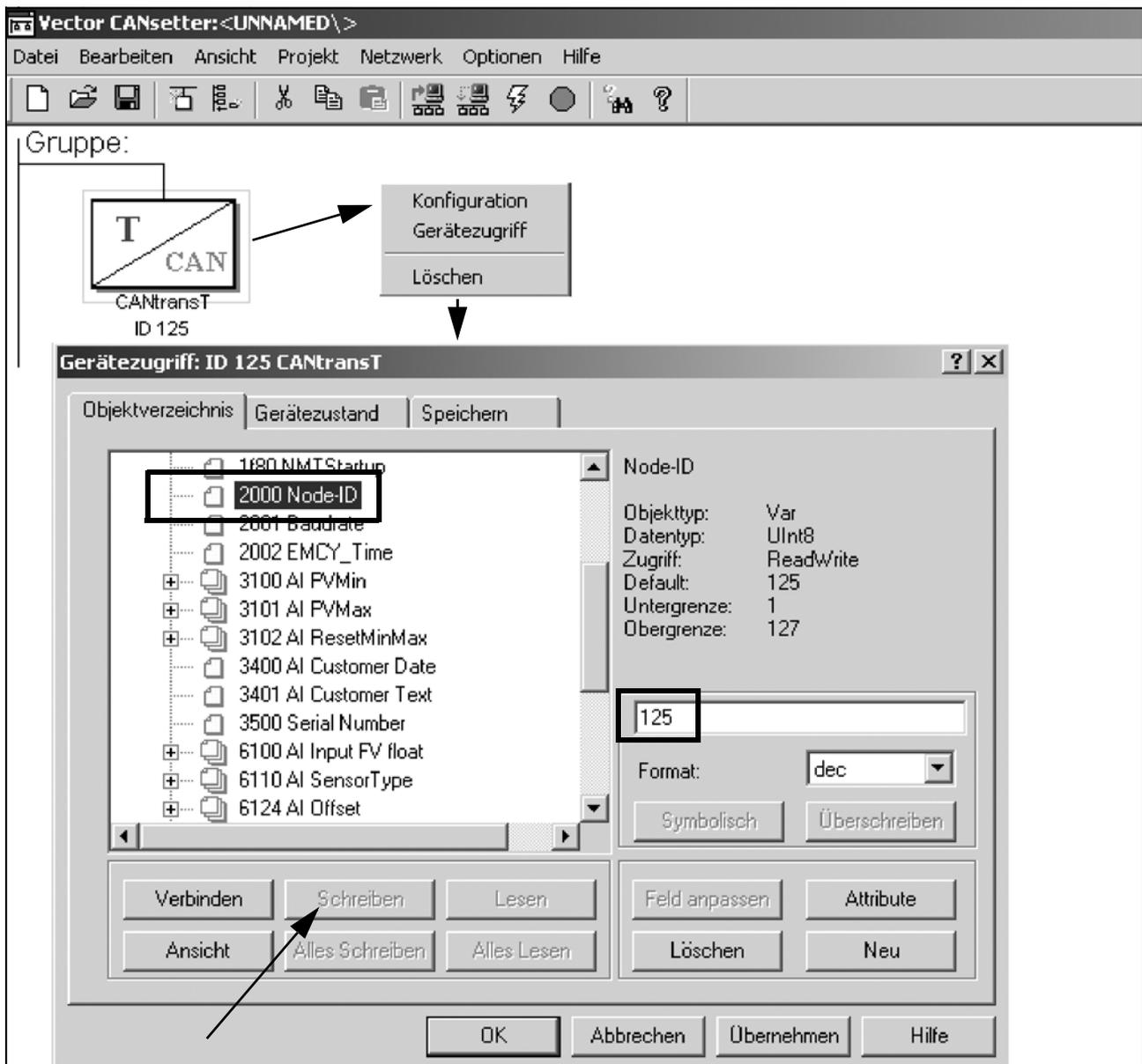
Wir wollen uns die Konfiguration eines Knotens, beispielhaft mit dem Programm CANsetter der Firma Vector, anschauen. Die Konfigurationsparameter werden über ein Objektverzeichnis adressiert. Um Zugriff auf dieses Verzeichnis zu erhalten, muss der Hersteller eines CANopen-Gerätes eine EDS(Electronic Data Sheet)-Datei bereitstellen.

In Abbildung 79 „starten“ wir CANsetter und konfigurieren den Messumformer JUMO CANtrans T. Der Messumformer arbeitet im Standard mit 500 kBaud. Die Geräteadresse beträgt werkseitig 125.



**Abbildung 79: Zugriff auf Objektverzeichnis eines JUMO CANtrans T**

1. Um den JUMO CANtrans T erreichen zu können, wird die Datenübertragungsrate auf die Geschwindigkeit des JUMO CANtrans T (500 kBit/s) unter „Projekt - Globale Konfiguration“ eingestellt.
2. Direkt nach dem Start des Programmes ist ein neuer Knoten angelegt (ID?). Diesem wird im Fenster „Node Configuration“ die Adresse 125 vergeben.
3. Unter „Gerätetyp“ wird nun die vom Hersteller mitgelieferte EDS-Datei ausgewählt.



**Abbildung 80: Konfiguration eines JUMO CANtrans T**

Nachdem der Knoten angelegt ist, gelangt man durch Anwahl von „Gerätezugriff“ in sein Objektverzeichnis. Das Objektverzeichnis ist eine Tabelle, in der alle Konfigurationsparameter über eine vierstellige, hexadezimale Zahl angesprochen werden. Wie in Abbildung 80 zu sehen ist, kann z. B. die Adresse des Teilnehmers über 0x2000 angesprochen werden (0x = Kennung für Zahl im hexadezimalen System, Node-ID = Adresse des Teilnehmers). Im markierten Feld ist die Adresse (125) ersichtlich. Diese könnte geändert und die neue Adresse nach Anwahl von „Schreiben“ an den JUMO CANtrans T gesendet werden. In gleicher Weise werden alle Parameter geändert.

### Process Data Object (PDO)

Nach der vollständigen Konfiguration des Knotens kann der Messumformer mit der Messwertübertragung beginnen. Der Messwert (im Fall eines JUMO CANtrans T ist dies die Temperatur) wird in einem so genannten PDO gesendet. Dem PDO wird ein in diesem Kapitel bereits beschriebener Identifier (COB-ID) vorangestellt. Die COB-ID kann konfiguriert werden. Werkseitig ist 0x180+Node-ID definiert. Bei einem Temperaturmessumformer mit der Adresse 125 (dezimal) wür-

# 5 Bussysteme

---

de werkseitig die COB-ID 0x180+125 (dezimal) = 0x1FD = 00111111101 (binär) betragen.

In unserem Fall würde sich das Telegramm wie folgt gestalten:

0	0	1	1	1	1	1	1	1	0	1	Temperatur	CRC-Field
COB-ID												

**Abbildung 81: Telegramm eines JUMO CANtrans T mit COB-ID 0x1FD**

## Wann sendet ein Messumformer den Messwert, das PDO?

Der Zeitpunkt, wann ein Messumformer das PDO sendet, ist konfigurierbar. Man unterscheidet hierbei zwischen Ereignis gesteuertem und synchronem Senden.

Ein ereignisgesteuertes Senden erfolgt,

- wenn sich der Messwert seit dem letzten Senden um eine einstellbare Differenz ändert (Delta),
- wiederholend zu einer einstellbaren Zeit (Eventtimer),
- beim Auftreten eines Hardwarefehlers bzw. Overrange, Underrange (Toggle),
- beim Umschalten in Operational (Erklärung folgt),
- auf Grund der Anfrage eines Masters (RTR empfangen).

Wie aus der Aufzählung deutlich wird, gibt es viele Bedingungen für das Aussenden des PDO. Deshalb kann im Messumformer eine Mindestzeit zwischen zwei PDO's festgelegt werden (Inhibitimer). Durch diesen Parameter kann beispielsweise festgelegt werden, dass der Messwert höchstens einmal in der Sekunde gesendet wird (auch wenn z. B. fünfmal die Bedingung für das Senden vorlag).

Alternativ erfolgt das Senden synchronisiert:

Der Messumformer kann in der Weise konfiguriert werden, dass er sein PDO nur noch im Fall einer zyklischen Anfrage eines Masters sendet (SYNC empfangen). In diesem Fall sendet der JUMO CANtrans T nicht mehr bei den Ereignis gesteuerten Bedingungen, wie Delta, Eventtimer, Toggle, Wechsel zu operational.

## Wie verbindet man bei CANopen zwei Teilnehmer?

Der JUMO CANtrans T besitzt ein PDO, mit dem er den Messwert sendet. Alle Teilnehmer hören die Nachricht mit, aber nur der (die) Teilnehmer, für den (die) die Nachricht bestimmt ist, verarbeitet(n) diese. Teilnehmer, die eine Nachricht über CANopen empfangen können, benötigen mindestens ein Empfangs-PDO. Über dieses Empfangs-PDO kann z. B. ein Regler mit CANopen-Schnittstelle seinen Istwert empfangen. Für das Eingangs-PDO des Reglers wird nun die gleiche COB-ID eingestellt, unter der z. B. ein Messumformer den Wert sendet.

### Beispiel:

Soll der Regler den Temperaturmesswert des JUMO CANtrans T (COB-ID 0x1FD) als Istwert übernehmen, wird für sein Eingangs-PDO ebenfalls die COB-ID 0x1FD eingestellt.

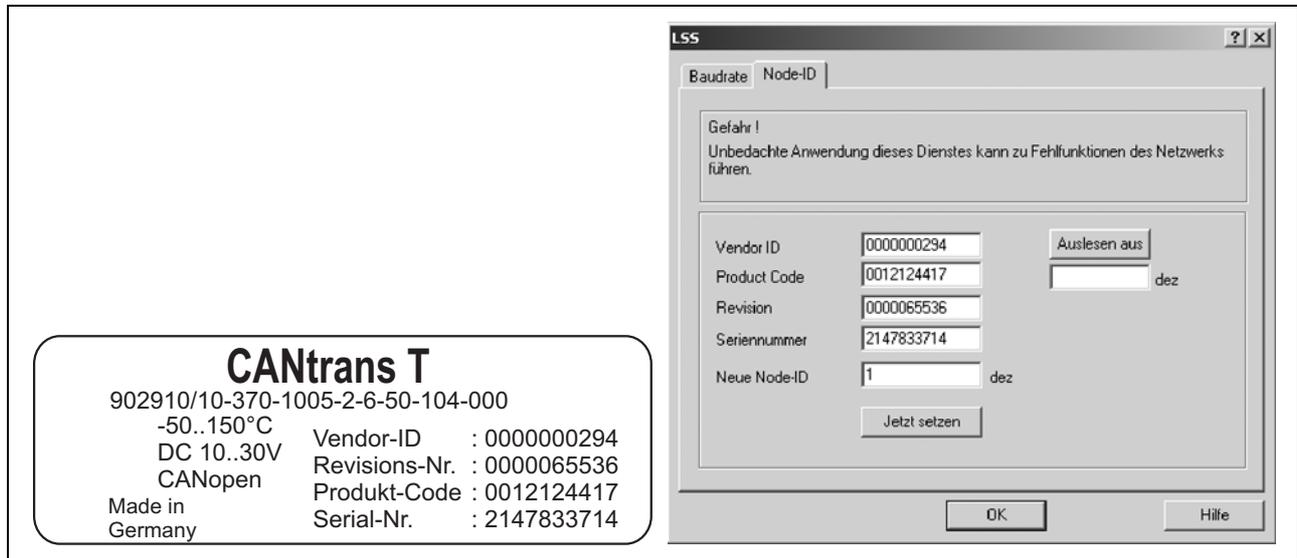
## Emergency Object (Fehlermeldung)

Im Fall der Statusänderung von Hardwarefehlern sendet der JUMO CANtrans T eine hochpriorie Fehlermeldung. Im Objektverzeichnis wird die Zeit in ms eingestellt, in welcher das Emergency Object zyklisch wiederholend gesendet wird.

## Vergabe der Node-ID (Geräteadresse) und Baudrate über LSS (Layer Setting Services)

Ein Beispiel:

Der Anwender setzt 10 Temperaturmessumformer JUMO CANtrans T ein. Wie bereits erwähnt, haben diese Aufnehmer werkseitig die Geräteadresse 125. Prinzipiell können die Teilnehmer nicht gleichzeitig über den Bus angesprochen werden (sie haben die gleiche Adresse). Eine Veränderung der Geräteadresse (und auch Baudrate) ist im genannten Fall über „LSS“ möglich. Mit „LSS“ können die Teilnehmer über Daten angesprochen werden, welche sich auf dem Typenschild befinden:



**Abbildung 82: Typenschild JUMO CANtrans T und Unter Menü LSS aus dem Konfigurationsprogramm**

Mit den Angaben auf dem Typenschild (Vendor-ID, Revisions-Nr. etc., Abbildung 82, links), ist der Messumformer über das herstellerunabhängige Konfigurationswerkzeug (Abbildung 82, rechts) identifizierbar. Der Messumformer kann angesprochen, die Baudrate und Adresse verändert werden. Nach einem Neustart des Systems (Unterbrechung der Spannungsversorgung) haben die Knoten die eingestellte Adresse und Baudrate übernommen und können weiter konfiguriert werden.

### Pre-Operational, Operational

Prinzipiell befindet sich der Messumformer nach Anlegen der Netzspannung im Modus „Pre-Operational“, das bedeutet, er sendet noch keine Daten. Üblicherweise werden die Sensoren durch eine übergeordnete Steuerung (ein Master) Operational (aktiv) geschaltet. Dieses Verhalten liegt werksseitig vor. Der Messumformer kann in der Weise konfiguriert werden, dass er nach Aufschalten der Netzspannung automatisch in den Modus „Operational“ schaltet (er ist aktiv, sendet sein PDO etc.).

Die gesamte Konfiguration des Messumformers geschieht über das Objektverzeichnis.

Abbildung 83 zeigt die Arbeitsweise des JUMO CANtrans T.

Weiterhin wird deutlich, wo im Objektverzeichnis die einzelnen Größen verändert werden:

# 5 Bussysteme

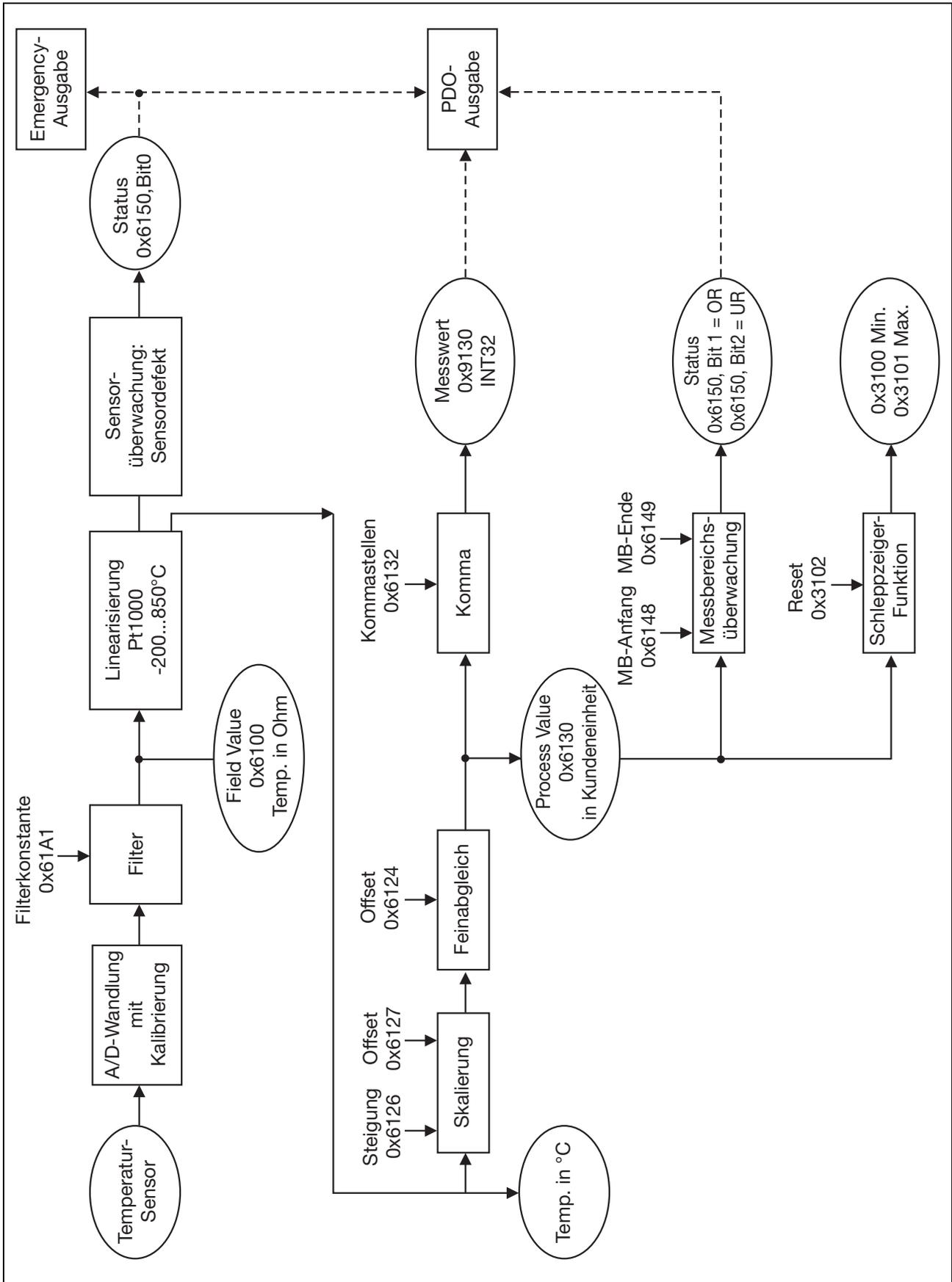


Abbildung 83: Datenfluss Temperaturkanal

Verfolgen Sie den Messwert ab Temperatursensor (Abbildung 83). Hinter dem Filter (die Dimensionierung des Filters kann im Objektverzeichnis unter 0x61A1 erfolgen) steht der Messwert in Ohm zur Verfügung (im Messumformer befindet sich ein Pt1000-Widerstandssensor). Der Messwert kann im Objektverzeichnis unter 0x6100 ausgelesen werden.

Nach der Linearisierung kann der Kunde eine Skalierung mit Feinabgleich durchführen. So kann z. B. die Temperatur (steht unter 0x6130 an) von Celsius in Fahrenheit gewandelt werden.

Mit dem JUMO CANtrans T kann eine Messbereichsüberwachung durchgeführt werden. Im Fall der Überschreitung (siehe MB-Ende 0x6149) oder Unterschreitung (siehe MB-Anfang 0x6148) wird das PDO versendet. Mit dem PDO werden neben dem Messwert auch zusätzliche Statusbits übertragen, die anzeigen, ob eine Überschreitung von MB-Ende oder Unterschreitung von MB-Anfang vorliegt.

# 5 Bussysteme

## 5.5 HART®

### 5.5.1 HART®-Protokoll

Die Grundidee dieser Technik beruht darauf, dass auf einem Leitungspaar sowohl das analoge 4 ... 20mA-Signal als auch digitale Kommunikationssignale übertragen werden. HART® steht für „**H**ighway **A**dressable **R**emote **T**ransducer“ und ist ein eingetragenes Warenzeichen der Firma Rosemount Inc.

Die HART®-Technik ist weitgehend standardisiert und bietet den Vorteil, dass man mit einem universellen Bediengerät in der Lage ist, die HART®-Geräte unterschiedlicher Hersteller zu konfigurieren. Dies ist möglich durch die Festlegung von definierten Kommandos, die bei allen HART®-Geräten gleich sind.

HART® ist für den Einsatz im Ex-Bereich geeignet.

Weltweite Unterstützung findet die weit verbreitete HART®-Technik durch die 1993 gegründete HART Communication Foundation (HCF).

### 5.5.2 FSK-Verfahren

Die Kommunikationstechnik von HART® basiert auf dem FSK-Verfahren (**F**requency **S**hift **K**eysing). Dem eingepprägten Stromsignal 4 ... 20mA auf der Zweidrahtleitung, welches den Messwert repräsentiert, wird auf dem gleichen Kabel das Kommunikationssignal als mittelwertfreie Wechselspannung mit niedriger Amplitude (500mVSS) überlagert, wobei der logischen „0“ = 2200Hz und der logischen „1“ = 1200Hz entspricht. Über die digitale Kommunikation werden neben der primären Messgröße die Konfigurationsparameter übertragen.

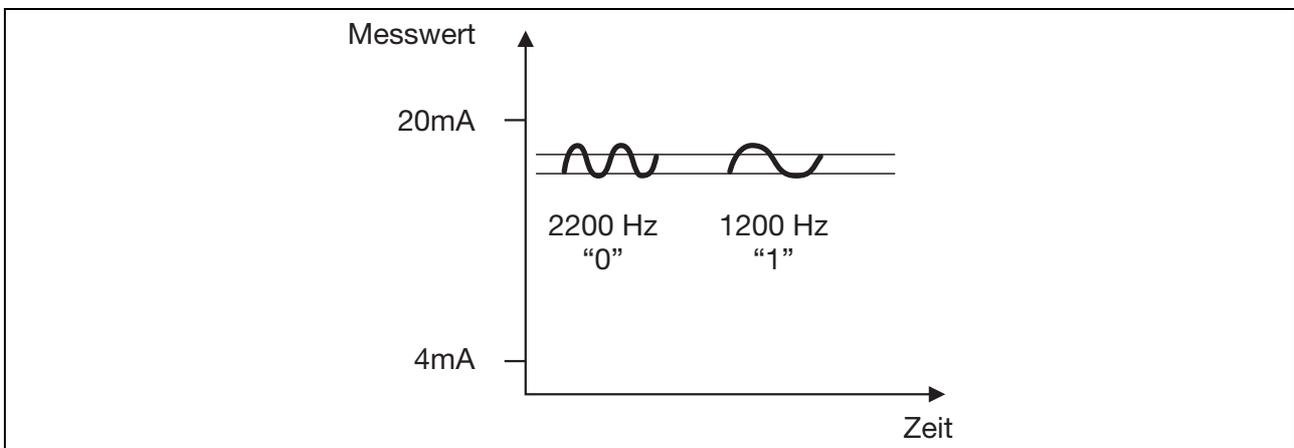


Abbildung 84: HART®-Kommunikation

HART®-Geräte werden üblicherweise mit Punkt-zu-Punkt-Verbindung betrieben oder auch sternförmig über entsprechende Feldmultiplexer.

### 5.5.3 Multidropbetrieb

Weiterhin ist mit HART® ein Busbetrieb möglich:

Im so genannten Multidropbetrieb werden bis zu 15 Geräte parallel an eine Zweidrahtleitung angeschlossen und mit einem Automatisierungssystem verbunden (Linientopologie). Hierzu erhält jedes Gerät eine Busadresse und das analoge Ausgangssignal wird auf einen festen Wert von 4mA eingestellt, wodurch lediglich die Speisung der Geräte erfolgt. Es findet nur noch ein digitaler Informationsaustausch mit dem FSK-Signal statt. Der Master (bis zu zwei Master sind möglich) fragt die Werte der Geräte nacheinander ab. Mit der HART®-Kommunikation werden ein bis zwei Telegramme je Sekunde übertragen. Befindet sich an einem Bus die maximale Anzahl von 15 HART®-Geräten, dauert ein Abfragezyklus mehrere Sekunden!

Die HART®-Kommunikation ist im allgemeinen (und besonders im Multidropbetrieb) nicht unbedingt für das Regeln und Steuern einer Anlage geeignet, sondern kommt eher bei der Inbetriebnahme und Wartung zum Einsatz. Im laufenden Betrieb wird meist über das Stromsignal gearbeitet.

### 5.5.4 JUMO-Geräte mit HART®

JUMO produziert und liefert Druck- und Temperaturmessumformer mit HART® - auch für den Ex-Bereich. Die Vorgehensweise bei der Konfiguration eines JUMO-Gerätes mit HART®-Kommunikation wird am Beispiel des Druckmessumformers JUMO dTRANS p02 gezeigt.



**Abbildung 85: Druckmessumformer JUMO dTRANS p02 und Temperaturmessumformer JUMO dTRANS T01 HART®**

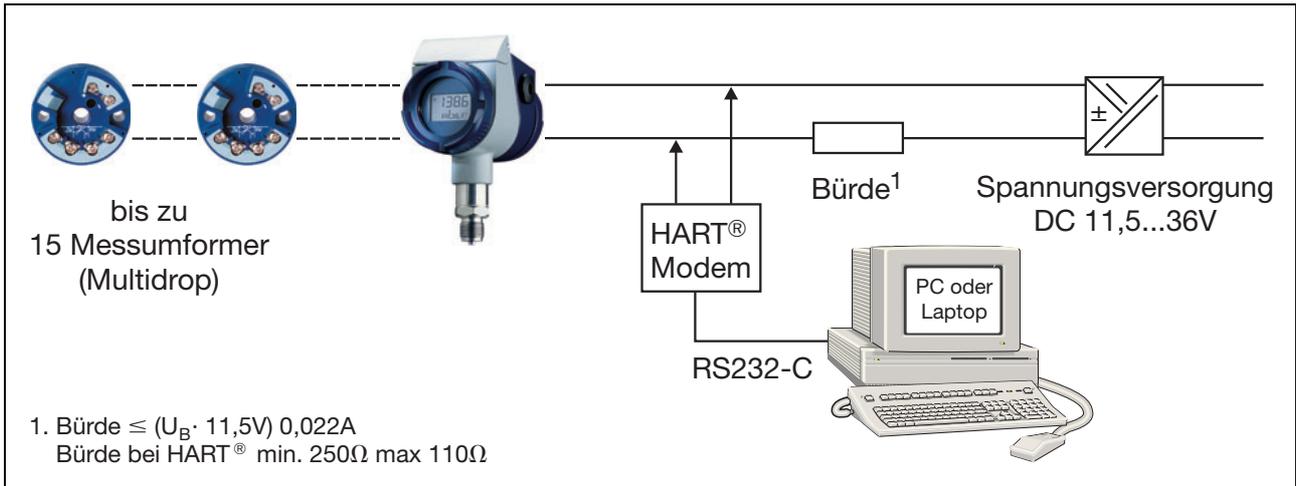
Die Messumformer verfügen über einen analogen Ausgang 4 ... 20mA. Diesem Signal wird, wie bei Sensoren mit HART®-Protokoll üblich, das digitale Kommunikationssignal überlagert.

Neben der Primärgröße Druck erfolgt beim JUMO dTRANS p02 die Übertragung weiterer Daten, wie Temperatur, maximaler Druck, minimaler Druck etc., über das digitale Signal.

### 5.5.5 Kommunikation über PC

Zur Kommunikation zwischen PC und JUMO dTRANS p02 ist ein handelsübliches HART®-Modem notwendig, welches das Frequenzsignal von HART® auf die RS-232-Schnittstelle umsetzt. Ergänzend wird eine Bürde eingefügt, über der als Ergebnis des Stromwechselsignales ein Spannungssignals entsteht.

## 5 Bussysteme

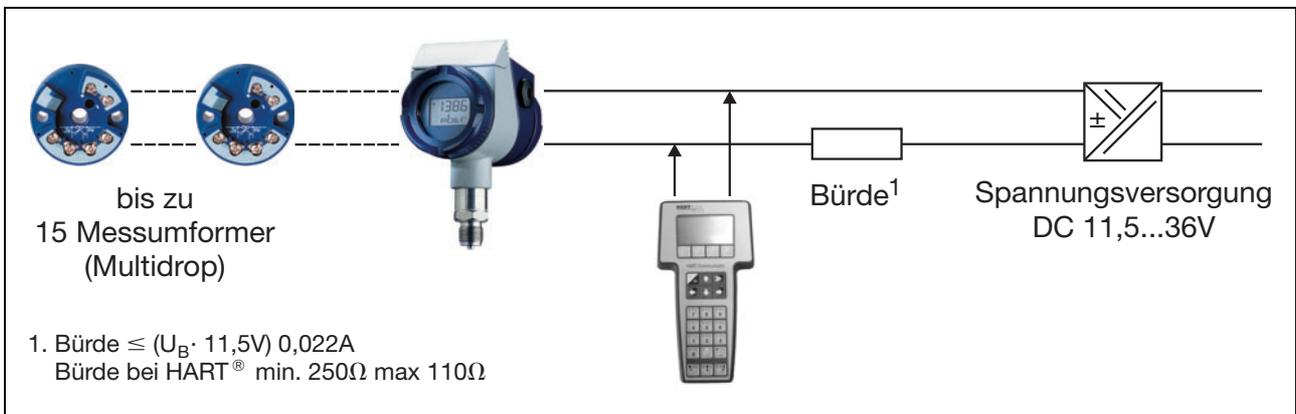


**Abbildung 86: Kommunikation zwischen PC und Messumformer**

Mit dem Setupprogramm für den JUMO dTRANS p02 kann nach dem Verbindungsaufbau über den PC die Parametrierung des Messumformers erfolgen. Das Setupprogramm ist mit einer VDI/VDE2187-Oberfläche für Feldgeräte ausgestattet. Mit dieser Oberfläche werden die Parameterdaten, wie Nullpunkt, Verstärkung, Geräteadresse etc., für das Gerät erstellt und über die HART®-Kommunikation in das Gerät übertragen. Zusätzlich können die Messwerte von HART®-Geräten visualisiert und aufgezeichnet werden.

### 5.5.6 HHT

Des Weiteren wird ein HHT (**H**and**H**eld **T**erminal) zur Parametrierung des Messumformers verwendet:



**Abbildung 87: Kommunikation zwischen HHT und Messumformer**

Das HHT besitzt ein integriertes Modem und wird über eine Bürde an den Messumformer angeschlossen. Es bedient herstellerübergreifend alle HART®-Geräte und kommt auch im Ex-Bereich zum Einsatz.

### 5.5.7 Tastatur/LCD

Ohne HART®-Kommunikation erfolgt die Parametrierung des JUMO dTRANS p02 klassisch per Tastenbedienung am Gerät.

## Modems

Das Wort „Modem“ setzt sich zusammen aus den Begriffen **Modulator/Demodulator**. Ein Modem empfängt Daten (z. B. von einem PC) in digitaler Form und wandelt diese z. B. in ein analoges Signal um (im Beispiel eines analogen Telefonnetzes). Das Empfangsmodem demoduliert die Daten wieder, womit sie dem Empfänger erneut in digitaler Form zur Verfügung stehen.

JUMO nutzt Modems beispielsweise für

- Teleservice:

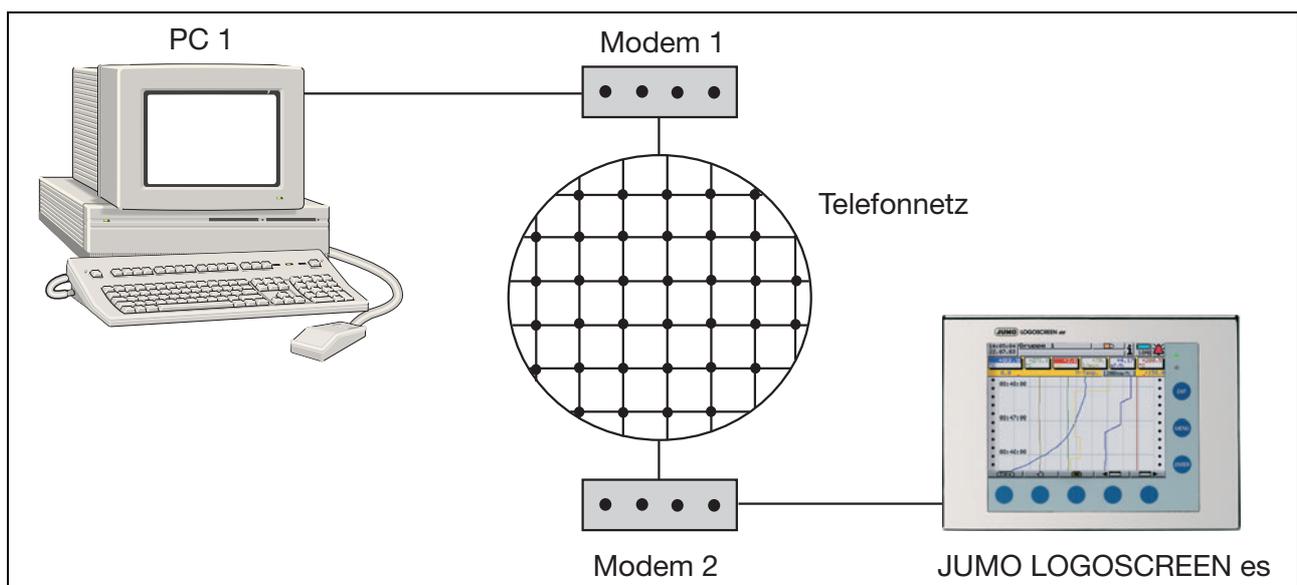
Man kann sich z. B. über ein Setupprogramm den Status aller Ein- und Ausgänge eines Reglers anzeigen lassen und das Gerät von der Ferne aus konfigurieren.

- Anwendungen mit der Prozessvisualisierungssoftware JUMO SVS-2000N:

Die Software gibt einen Überblick über die Prozessgrößen von Anlagen, welche unterschiedliche Standorte besitzen. Ein PC wählt sich von Zeit zu Zeit in die Anlagen ein und aktualisiert die in der Software visualisierten Größen (z. B. Vorräte von Betriebsstoffen etc.).

- die durch einen Bildschirmschreiber aufgezeichneten Messdaten werden über eine Modem-Verbindung von einem PC abgefragt.

Wir wollen nun einige Grundlagen zu Modems anschneiden und uns am letztgenannten Beispiel (Datenfernübertragung von Messdaten eines Bildschirmschreibers) orientieren.



**Abbildung 88: Datenfernübertragung von Messdaten des Bildschirmschreibers JUMO LOGOSCREEN es**

Prinzipiell kann die Verbindung über eine Analog- bzw. ISDN-Verbindung erfolgen. Weiterhin kann die Datenübertragung über Mobilfunk geschehen. Entsprechend können vom Markt analoge, ISDN- oder GSM-Modems bezogen werden. **Bitte beachten Sie jedoch die Bemerkungen unter „Support von JUMO bezüglich Modemverbindung“ am Ende dieses Kapitels.**

Auf der PC-Seite können sowohl externe (RS-232-Schnittstelle, USB etc.) als auch interne (Steckkarten ISA, PCI etc.) Modems verwendet werden. Auf der PC-Seite muss der passende Windows-Treiber verwendet werden. Nach erfolgreicher Installation erscheint das Modem in der Liste der Modems (Gerätemanager). Im Fall eines externen Modems mit RS-232-Schnittstelle wird ein einzu-eins-verdrahtetes Kabel genutzt. Das Kabel gehört üblicherweise zum Lieferumfang des Modems. Ebenso wird in den meisten Fällen das Verbindungskabel zur Telefonsteckdose (TAE-Dose) mitgeliefert.

# 6 Modems

---

Das Modem auf der Geräteseite (hier befindet sich im Beispiel der Bildschirmschreiber JUMO LOGOSCREEN es) muss eine serielle Schnittstelle zum Anschluss eines Bildschirmschreibers bieten. Soll ein Gerät angeschlossen werden, ist die RS-232-Schnittstelle ausreichend, bei mehreren Geräten kommt die RS-485- bzw. RS-422-Schnittstelle zum Einsatz.

## 6.1 Kommandos für das Modem

Die Steuerung des Modems geschieht nach einem De-Facto-Standard, der von der amerikanischen Firma Hayes entwickelt und dann von anderen Herstellern übernommen wurde. Gelegentlich hört man auch von „Hayes-kompatiblen Modems“.

Das Modem ist in der Lage, auf bestimmte Befehle zu reagieren, die über die serielle Schnittstelle gesendet werden. Diese Befehle beginnen immer mit den beiden Buchstaben „AT“ (**A**ttention). Dann folgen die Befehlscodes für bestimmte Funktionen, wie z. B.

ATH0	Hörer auflegen
ATH1	Hörer abheben
ATDP142	Nummer 142 wählen

### **Bemerkung:**

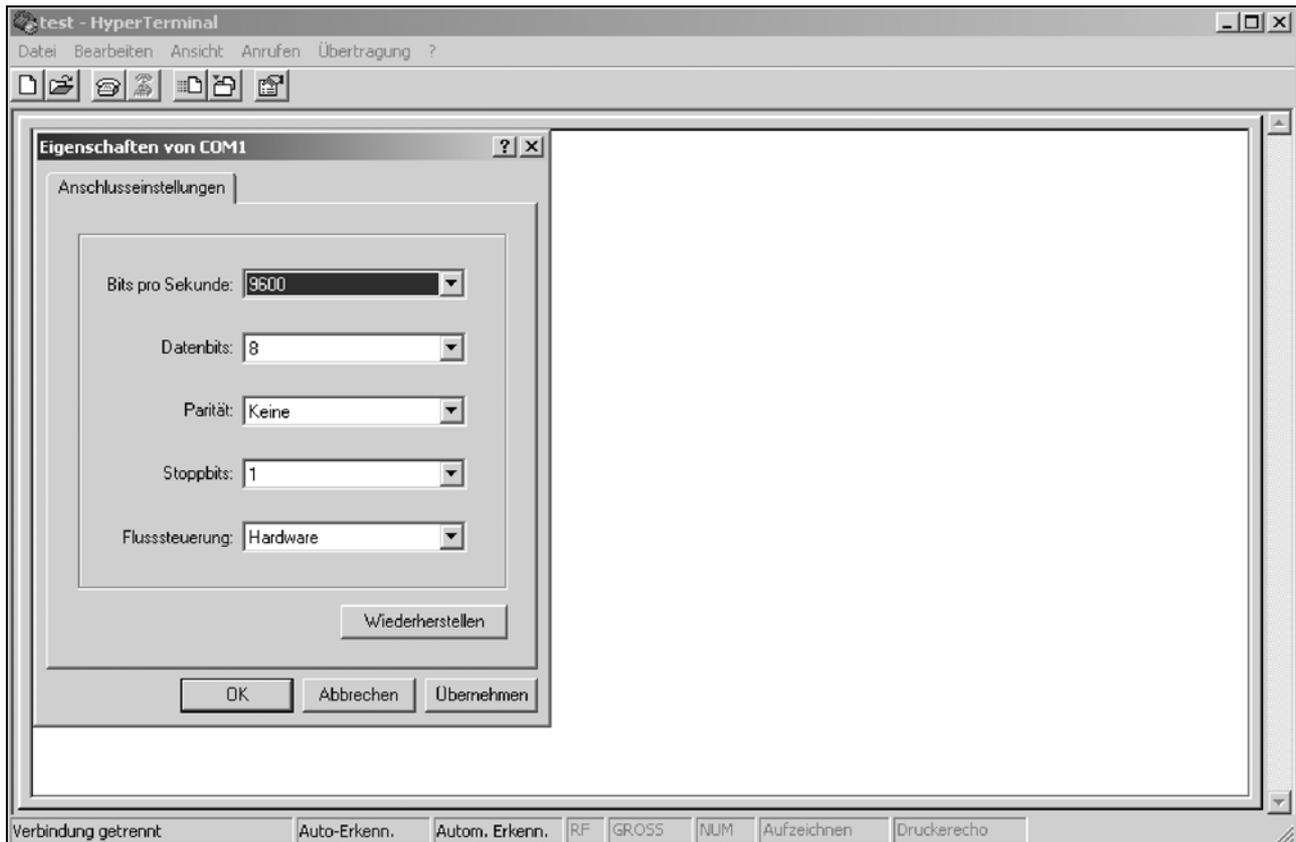
Zusätzlich gibt es noch den Standard V.25bis. Der Hayes-Befehlssatz ist aber der bekanntere und wird von fast allen Modems verstanden.

Der Verbindungsaufbau zwischen zwei Modems geschieht durch Anwahl der Gegenstation: das Modem wählt die Nummer der Gegenstation, diese nimmt das Gespräch automatisch an. Bei Modems mit höheren Übertragungsraten und automatischer Anpassung der Baudrate tauschen nun beide Stationen verschiedene Tonsignale aus (kann über die Lautsprecher der Modems verfolgt werden). Hiermit wird die entsprechende Baudrate ausgehandelt.

Bevor eine Verbindung über Modem realisiert werden kann, muss die Konfiguration der Modems erfolgen: das Modem auf der PC-Seite (Abbildung 88) dient lediglich der Einwahl in das Telefonnetz. Für dieses kann meist die werkseitige Konfiguration verwendet werden.

Für das geräteseitige Modem (dieses befindet sich in unserem Beispiel auf der Seite des Bildschirmschreibers) muss eine Konfiguration erfolgen. Hierzu wird das „Gerätmodem“ über die serielle Schnittstelle an den PC angebunden. Die Konfiguration erfolgt über ein Terminalprogramm, wie z. B. Hyperterminal (gehört zum Standardzubehör von Windows und kann unter „Programme → Zubehör → Kommunikation“ erreicht werden).

Bevor mit Hyperterminal eine Verbindung aufgebaut wird, erfolgt die Einstellung für das Datenformat (Baudrate, Anzahl Stoppbits, Paritätscheck). Diese Einstellung übernimmt das Modem für seine serielle Schnittstelle. Am Bildschirmschreiber muss die Schnittstelle gleich eingestellt sein.



**Abbildung 89: Einstellen der Baudrate und des Datenformates mit Hyperterminal**

Folgend wollen wir uns die AT-Befehle, welche für das Modem auf der Geräteseite benötigt werden, anschauen (diese sind nur für das Modem vom Typ HSM 33.6 gültig).

### **AT&F**

Das Modem wird auf die werkseitige Konfiguration gesetzt.

### **AT&K0**

Wenn der Computer bzw. das Modem mit dem Datenfluss nicht mehr zurecht kommt, besteht die Möglichkeit, den Datenfluss anzuhalten:

#### **- Hardwarehandshake**

Vom Hardwarehandshake wird gesprochen, wenn der Datenfluss durch Spannungspegel auf speziellen Steuerleitungen geregelt wird. Das Modem setzt das CTS-Signal auf „OFF“ und erst dann wieder auf „ON“, wenn es aufnahmebereit ist. Dadurch wird die Datenübertragung vom Computer an das Modem gestoppt. Durch „RTS auf OFF setzen“ kann der Computer die Daten vom Modem anhalten.

#### **- Beim Verbindungskabel Bildschirmreiber - Modem sind die genannten Pins (CTS und RTS) nicht verdrahtet. Aus diesem Grund wird mit AT&K0 der Hardwarehandshake ignoriert.**

Im Fall der Datenübertragung von Messdaten handelt es sich um kurze Modbus-Protokolle.

Daher ist eine Übersättigung mit Daten relativ unwahrscheinlich. Sollte es trotzdem zu einem Überlauf kommen, wird dies z. B. durch die übergeordneten CRC-Checks geprüft und die Daten ggf. noch einmal abgefragt.

# 6 Modems

---

## **AT&D0**

Das Modem ignoriert das DTR-Signal.

## **ATS0=1**

Das Modem besitzt interne Register. Mit diesen wird die Konfiguration ebenfalls beeinflusst. Mit ATS0=1 nimmt das Modem nach dem ersten Klingelimpuls den kommenden Ruf an.

## **ATT**

Das Frequenzwahlverfahren (Tonwahl) wird eingestellt.

## **ATE0**

Die zum Modem gesendeten Kommandos werden nicht als Echo zurückgesendet. Hiermit wird sichergestellt, dass sich das Modem im späteren Betrieb auf das Durchreichen der Daten beschränkt und sich sonst „ruhig verhält“. Somit sendet das Modem keine weiteren Daten auf den Bus (RS-232- oder RS-485-Schnittstelle).

## **ATQ1**

Rückmeldungen aus: Nach Senden eines AT-Befehls, z. B. über Hyperterminal, antwortet das Modem nicht mehr mit „OKAY“. Es wird ebenfalls sichergestellt, dass sich das Modem im späteren Betrieb auf das Durchreichen der Daten beschränkt und sich sonst „ruhig verhält“.

## **ATX0**

Das Modem wartet nicht auf ein Freizeichen und ignoriert Besetzt-Zeichen.

## **AT&W0**

Die aktuelle Konfiguration des Modems wird in Profil 0 gespeichert (es existiert noch Profil 1). Die Werte bleiben auch nach Abschalten des Modems erhalten.

## **AT&Y0**

Das Konfigurationsprofil 0 wird beim Einschalten des Modems geladen.

## **ATZ**

Das Modem wird zurückgesetzt: die Verbindung wird unterbrochen und - wie vorher angegeben - Konfigurationsprofil 0 geladen.

Wie bereits erwähnt, handeln die Modems vor der Datenübertragung die Baudrate aus. Bei schlechten Verbindungen kann es von Vorteil sein, wenn die Modems auf eine kleinere Baudrate fest eingestellt werden. Um die Baudrate fest einzustellen, gelten für das HSM 33.6 folgende AT-Befehle:

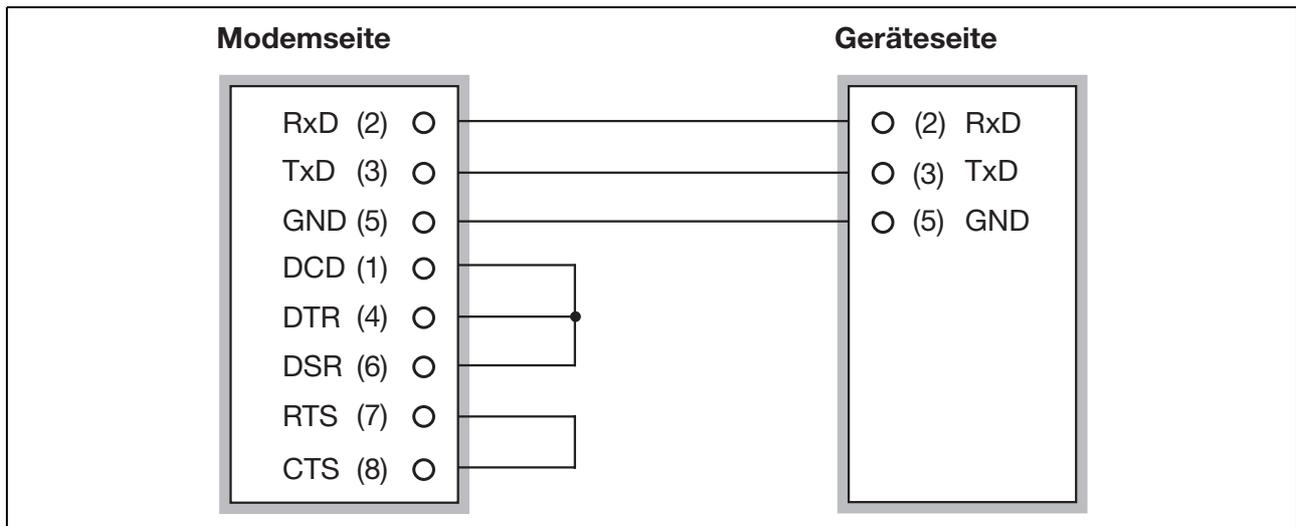
ATB8 für 9600 Baud

ATB11 für 19200 Baud

Um das entsprechende Verhalten für ein anderes Modem zu erreichen, gelten andere AT-Befehle.

Wird die Baudrate auf der Geräteseite reduziert, sollte diese auf der PC-Seite ebenfalls auf die entsprechend geringere Baudrate eingestellt werden (Abbildung 89).

Beim Verbindungskabel vom Modem zum Bildschirmschreiber ist auf dessen Belegung zu achten:



**Abbildung 90: Verbindungskabel Modem/Bildschirmschreiber im Fall einer RS-232-Schnittstelle**

### Modem auf der PC-Seite

Das Modem auf der PC-Seite dient lediglich für die Einwahl in das Telefonnetz. Hier können meist die Werksdaten übernommen werden.

### PCA-Kommunikations-Software PCC

Für das genannte Beispiel entwickelte JUMO eine Software, welche zeitgesteuert eine Modemverbindung aufbaut und die im Bildschirmschreiber gespeicherten Messdaten abfragt. Mit der PCC können die von JUMO empfohlenen Modems konfiguriert (initialisiert) werden. Die Modems können in der Software aufgerufen und die notwendigen AT-Befehle übertragen werden. Es entfällt die Arbeit mit Hyperterminal.

### Support von JUMO bezüglich Modemverbindung

Nicht alle AT-Befehle sind genormt: AT-Befehle, die mit einem Modem die Verbindung über das Telefonnetz ermöglichen, müssen für ein weiteres Modem nicht gelten. JUMO prüft jeweils ein analoges, ISDN- und GSM-Modem und gibt für diese die geeigneten AT-Befehle bekannt. Auf Wunsch liefert JUMO diese Modems auch vorkonfiguriert aus.

Für Modems, welche von JUMO nicht getestet und empfohlen werden, leistet JUMO keinen Support.

# 6 Modems

---

# Fachliteratur von JUMO - lehrreiches für Einsteiger und Praktiker

Nicht nur bei der Herstellung von JUMO-Produkten, auch beim späteren Einsatz ist Know-How gefragt. Deshalb bieten wir unseren Anwendern eigene Publikationen zu Themen der Mess- und Regelungstechnik an.

Die Publikationen sollen Einsteigern und Praktikern die unterschiedlichsten Anwendungsgebiete schrittweise näher bringen. Hierbei werden überwiegend allgemeine Themenbereiche, zum Teil auch JUMO-spezifische Anwendungen, erläutert.

Zusätzlich zur JUMO-Fachliteratur, bieten wir Ihnen neben unseren Software-Downloads die Möglichkeit der direkten Online-Bestellung von Prospekten und CD-ROM-Katalogen.



## Elektrische Temperaturmessung mit Thermoelementen und Widerstandsthermometern Matthias Nau

FAS 146  
deutsche Ausgabe, gebunden  
Verkaufs-Artikel-Nr.: 00074750  
ISBN-13: 978-3-935742-06-1  
zum Preis von 14,- EUR netto



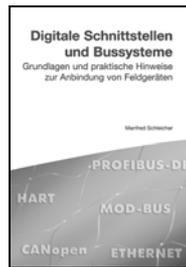
## Regelungstechnik für den Praktiker Manfred Schleicher

FAS 525  
deutsche Ausgabe, gebunden  
Verkaufs-Artikel-Nr.: 00314836  
ISBN-13: 978-3-935742-00-2  
zum Preis von 14,- EUR netto



## Explosionsschutz in Europa Elektrische Betriebsmittel Grundlagen, Richtlinien, Normen Jürgen Kuhlmei

FAS 547  
deutsche Ausgabe, gebunden  
Verkaufs-Artikel-Nr.: 00324966  
ISBN-10: 3-935742-08-8  
ISBN-13: 978-3-935742-08-5  
zum Preis von 9,- EUR netto



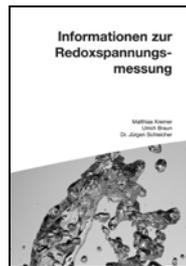
## Digitale Schnittstellen u. Bussysteme Grundlagen und praktische Hinweise zur Anbindung von Feldgeräten Manfred Schleicher

FAS 603  
deutsche Ausgabe, gebunden  
Verkaufs-Artikel-Nr.: 00339287  
ISBN-10: 3-935742-02-9  
ISBN-13: 978-3-935742-02-3  
zum Preis von 9,- EUR netto



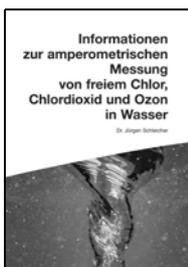
## Informationen zur Reinstwassermessung Reinhard Manns, Dr. Jürgen Schleicher

FAS 614  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00369643  
kostenfrei



## Informationen zur Redoxspannungsmessung Matthias Kremer, Ulrich Braun, Dr. Jürgen Schleicher

FAS 615  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00373848  
kostenfrei



## Informationen zur amperometrischen Messung von freiem Chlor, Chlordioxid und Ozon in Wasser Dr. Jürgen Schleicher

FAS 619  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00394969  
kostenfrei



## Elektronische Leistungssteller Manfred Schleicher, Winfried Schneider

FAS 620  
deutsche Ausgabe, gebunden  
Verkaufs-Artikel-Nr.: 00398728  
ISBN-13: 978-3-935742-04-7  
zum Preis von 9,- EUR netto

# Fachliteratur von JUMO - lehrreiches für Einsteiger und Praktiker



## Informationen zur pH-Messung Dr. Jürgen Schleicher

FAS 622  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00403231  
kostenfrei



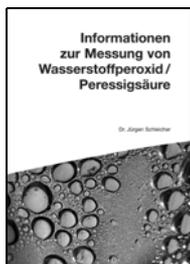
## Informationen zur Leitfähigkeitsmessung Reinhard Manns, Dr. Jürgen Schleicher

FAS 624  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00411335  
kostenfrei



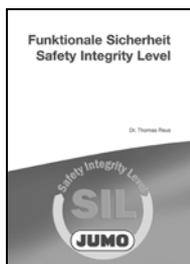
## Messunsicherheit einer Temperaturmesskette mit Beispielrechnungen Gerd Scheller

FAS 625  
deutsche Ausgabe, gebunden  
Verkaufs-Artikel-Nr.: 00413510  
ISBN-13: 978-3-935742-12-2  
zum Preis von 3,- EUR netto



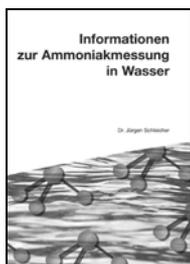
## Informationen zur Messung von Wasserstoffperoxid / Peressigsäure Dr. Jürgen Schleicher

FAS 628  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00420695  
kostenfrei



## Funktionale Sicherheit SIL Dr. Thomas Reus

FAS 630  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00463374  
kostenfrei



## Informationen zur Ammoniakmessung in Wasser Dr. Jürgen Schleicher

FAS 631  
deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00481786  
kostenfrei

Besuchen Sie unsere deutsche Website auf [www.jumo.de](http://www.jumo.de) (für Österreich [www.jumo.at](http://www.jumo.at), für die Schweiz [www.jumo.ch](http://www.jumo.ch)) und überzeugen Sie sich von der umfangreichen Produktpalette für die verschiedensten Einsatzgebiete. Dort finden Sie weitere Informationen und die dazugehörigen Ansprechpartner für Ihre Wünsche, Fragen, Anregungen und Bestellungen.

## Kataloge auf CD-ROM

Unsere Kataloge sind - außer in gedruckten Versionen - auch in digitaler Form erhältlich. Die CD-ROM mit deutschen oder englischen Daten enthalten strukturierte Kataloge im pdf-Format, die JUMO-Produktübersicht, die Kontaktadressen der JUMO-Ansprechpartner sowie den kostenlosen Download des Acrobat Readers.



## JUMO Produkte + Preise

deutsche Ausgabe  
Verkaufs-Artikel-Nr.: 00397668  
kostenfrei



## JUMO Products

englische Ausgabe  
Verkaufs-Artikel-Nr.: 00404116  
kostenfrei





[www.jumo.net](http://www.jumo.net)

