

**Einrichtung einer intelligenten  
Ausleseinheit für Verbrauchsmesszähler**

Diplomarbeit von Carsten Bories, März 1995



# Inhaltsverzeichnis

<b>1. Einleitung .....</b>	<b>3</b>
<b>2. Grundlagen zum M-Bus.....</b>	<b>4</b>
2.1. Konzept .....	4
2.2. Die physikalische Schicht .....	5
2.3. Die Verbindungsebene .....	5
2.3.1. Datenübertragung.....	6
2.3.2. Telegrammformate.....	6
2.3.3. Bedeutung der Felder .....	7
2.4. Die Netzwerkebene .....	8
2.5. Die Transportebene .....	8
2.6. Die Sitzungsebene .....	9
2.7. Die Darstellungsebene.....	9
2.8. Die Anwendungsebene.....	9
2.8.1. Das CI-Feld.....	9
2.8.2. Die feste Datenstruktur .....	10
2.8.3. Die variable Datenstruktur.....	11
2.8.4. Senden von Daten in Aufrufichtung.....	18
<b>3. Eine intelligente Ausleseeinheit auf dem M-Bus.....</b>	<b>25</b>
3.1. Einleitung .....	25
3.2. Die primären Informationen .....	27
3.2.1. Auslesen.....	27
3.2.2. Konfiguration.....	28
3.2.3. Initialisierung .....	28
3.3. Die sekundären Informationen .....	29
3.3.1. Auslesen.....	29
3.3.2. Fehlermeldungen .....	31
3.3.3. Konfiguration.....	32
3.3.4. Initialisierung .....	57
<b>4. Installierung und Betrieb der Ausleseeinheit.....</b>	<b>58</b>
4.1. Aufbau und Durchführung .....	58
4.2. Ergebnisse und Auswertung .....	65

---

<b>5. Messungen der Repeaterigenschaften .....</b>	<b>67</b>
5.1. Repeaterspezifikationen .....	67
5.2. Versuchsaufbau und Durchführung .....	68
5.3. Ergebnisse und Auswertung der Messungen .....	73
5.3.1. Der Analog-Repeater .....	73
5.3.2. Digital-Repeater.....	83
5.4. Tabellarischer Vergleich der beiden Repeater .....	98
<b>6. Zusammenfassung.....</b>	<b>99</b>
<b>7. Anhang .....</b>	<b>100</b>
7.1. Codierung von Datentypen .....	100
7.2. Tabellen zur festen Datenstruktur .....	103
7.2.1. Feld Medium/Einheit.....	103
7.2.2. Das Statusfeld .....	105
7.3. Tabellen zur variablen Datenstruktur.....	106
7.3.1. Das Statusfeld .....	106
7.3.2. Das Mediumfeld .....	106
7.3.3. Primäre VIF-Codes .....	107
7.3.4. Sekundäre VIF-Codes.....	108
7.3.5. VIFE-Codes .....	110
<b>Literaturverzeichnis.....</b>	<b>113</b>

## 1. Einleitung

Die Grundlage der vorliegenden Diplomarbeit bildet ein 1992 von Prof. Dr. Ziegler im Zusammenarbeit mit Texas Instruments entwickeltes Bussystem, daß den Namen M-Bus trägt. Es ist ausgerichtet auf die Fernauslesung und Fernspeisung von Verbrauchsmeßzählern (Endgeräten, Slaves), wie z.B. Wasser-, Gas- und Wärmemengenzähler. Diese senden bei Aufforderung ihre Meßwerte bzw. Zählerstände an eine übergeordnete Komponente auf dem Bus, dem sogenannten Master.

Diese Diplomarbeit hat zwei Ziele:

1. Entwicklung einer neuen, programmierbaren Komponente auf dem M-Bus, die als intelligente Ausleseeinheit bezeichnet wird. Diese entlastet den Master und erweitert die eingeschränkten Fähigkeiten der Endgeräte.
2. Messung und Auswertung von Eigenschaften zweier M-Bus Repeater (Bustreibern), die für den Betrieb mit 250 Endgeräten geeignet sind.

## 2. Grundlagen zum M-Bus

Die in diesem Kapitel vorgestellten Fähigkeiten des M-Bus Systems decken nur die Teile ab, die für die Diplomarbeit relevant sind. Eine vollständige Behandlung des M-Bus befindet sich in dem Werk "The M-Bus: A Documentation" von Andreas Papenheim, das die Grundlage zu diesem Kapitel bildet.

### 2.1. Konzept

Ziel bei der Entwicklung des M-Bus ist es, möglichst viele (bis zu 250) Endgeräte (i.a. Verbrauchsmeßzähler) über ein zweiadriges Kabel zu vernetzen und fernzuspeisen. Hierbei steht vor allen Dingen die Datensicherheit, und weniger die Datenübertragungsgeschwindigkeit im Vordergrund. Die Realisierung erfolgt durch ein reines Master-Slave Konzept. Dies bedeutet, daß ein Teilnehmer, der Master (typischerweise ein PC oder Handheld Computer), die Kommunikation auf dem Bus steuert, indem er die Slaves (Endgeräte) zur Datenübertragung auffordert. Die Slaves können untereinander nicht kommunizieren.

Der M-Bus ist ein sogenanntes offenes Verbindungssystem, das dem ISO-OSI-Referenzmodul genügt. ISO steht für "International Organization of Standardization". OSI bedeutet "Open System Interconnection". Ziel des Moduls ist es sicherzustellen, daß Informationen von unterschiedlichen Systemen nach standardisierten Prozeduren ausgetauscht und interpretiert werden können. Hierzu werden die Kommunikationsfunktionen in sieben Schichten (Ebenen, Layer) aufgeteilt (siehe Abb. 1). Der Aufbau ist hierarchisch. Jede Schicht ist mit der direkt darunterunterliegenden Ebene verbunden und benutzt deren Dienstleistungen.

7	Anwendungsebene	Anwendungsorientiert Ebenen
6	Darstellungsebene	
5	Sitzungsebene	
4	Transportebene	Transportorientierte Ebenen
3	Netzwerkebene	
2	Verbindungsebene	
1	Physikalische Ebene	

**Abb. 1:** Die sieben Ebenen des OSI-Referenzmoduls (aus [1])

In bestimmten Fällen läßt es das M-Bus Protokoll zu, daß die hierarchische Struktur umgangen wird. Deshalb wurde eine übergeordnete Ebene eingeführt, welcher der direkte Zugriff auf einzelne Schichten des Modells erlaubt ist. Diese Ebene wird als Management-Layer (Schicht, Ebene) bezeichnet und wird benutzt, um eine bestimmte Art der Adressierung von Slaves und das Wechseln der Baudrate auf dem Bus von der Anwendungsebene aus zu ermöglichen. In den folgenden Kapiteln wird die Funktionsweise des M-Bus anhand dieses Modells erklärt.

## 2.2. Die physikalische Schicht

Die unterste Schicht stellt die eigentliche physikalische Verbindung zwischen den Kommunikationspartnern dar. Diese wird beim M-Bus durch ein Aderpaar eines Standard-Telefonkabels realisiert. Bis zu 250 Slaves werden parallel zum Kabel in beliebiger Topologie angeschlossen. Die Gesamtlänge der Verbindung soll 1000 m, die weiteste Entfernung zwischen Master und Slave 350 m nicht überschreiten.

Um größere Entfernungen und eine erhöhte Anzahl von Slaves zu ermöglichen, kann der Bus um Segmente gemäß den obigen Bedingungen erweitert werden. Hierzu ist es aber notwendig, jedes Segment mit einem eigenen, sogenannten Remote Repeater zu betreiben, der die Aufgabe hat, die Signale auf dem M-Bus wiederherzustellen, da diese aufgrund der vorhandenen Kapazitäten und Widerstände verzerrt werden. Typischerweise besteht ein M-Bus System nur aus einem Segment mit einem Local Repeater.

Die Übertragung von Daten erfolgt bidirektional. In Richtung Master zum Slave (Aufrufichtung) wird mit Spannungsunterschieden gearbeitet. Der Ruhezustand auf dem Bus ist die logische 1 (Mark), welche durch eine Busspannung  $U_{\text{Mark}}$  im Bereich von 35 V bis 40 V am Repeaterausgang ausgedrückt wird. Die logische 0 (Space) wird mit  $U_{\text{Space}} = U_{\text{Mark}} - 12\text{V}$  dargestellt. In Antwortichtung (Slave zum Master) wird mit Stromunterschieden gearbeitet. Einer logischen 1, dem Ruhezustand, entspricht ein Strombedarf von höchstens 1,5 mA pro Slave, einer logischen 0 ein erhöhter Strombedarf im Bereich von 11-20mA.

Als Bustreiber wird (siehe oben) ein sogenannter Repeater eingesetzt, welcher unter anderem die Aufgabe hat, die Stromversorgung des Busses zu gewährleisten. Da als Master zumeist ein Personal Computer (PC) eingesetzt wird, ist es notwendig, daß die Pegel der seriellen Schnittstelle (+3 bis +15V entsprechen einer logischen 0, -3 bis -15V einer logischen 1) in M-Bus Pegel und umgekehrt gewandelt werden können.

## 2.3. Die Verbindungsebene

Die Ebene 2 des Referenz-Moduls ist zuständig für eine sichere Verbindung zwischen zwei Teilnehmern. Ferner werden Telegrammstrukturen, Verfahren zur Synchronisation und Adressierung der Teilnehmer festgelegt.

### 2.3.1. Datenübertragung

Die Übertragung erfolgt zeichenweise, asynchron und seriell. Jedes Zeichen besteht aus 11 Bits in folgender Reihenfolge: Ein Startbit (Space), 8 Datenbits, ein Paritybit (gerade Parität) und 1 Stopbit (Mark).

Die Synchronisation jedes Zeichens erfolgt anhand des Start- und Stopbits. Die Datenbits werden beginnend mit dem Bit der niedrigsten Wertigkeit (LSB: Least Significant Bit) mit einer Baudrate von mindestens 300 Baud übertragen.

### 2.3.2. Telegrammformate

Die Übertragung der Daten erfolgt in Telegrammen, welche aus den oben spezifizierten Zeichen bestehen. Abb. 2 zeigt die zur Verfügung stehenden Telegramme.

Einzelzeichen	Kurzsatz	Steuersatz	Langsatz
E5h	Start 10h	Startzeichen 68h	Startzeichen 68h
	C-Feld	L-Feld = 3	L-Feld
	A-Feld	L-Feld = 3	L-Feld
	Checksumme	Startzeichen 68h	Startzeichen 68h
	Stopzeichen 16h	C-Feld	C-Feld
		A-Feld	A-Feld
		CI-Feld	CI-Feld
		Checksumme	Anwenderdaten (0 bis 252 Byte)
		Stopzeichen 16h	Checksumme
			Stopzeichen 16h

**Abb. 2:** Telegrammformate des M-Bus Protokolls, pro Zelle 1 Byte (aus [2])

- Das Einzelzeichen dient zur Bestätigung einer, gemäß der Verbindungsschicht, korrekt empfangenen Nachricht.
- Der Kurzsatz besteht immer aus fünf Zeichen, Startzeichen ist 10h. Die Checksumme wird durch Addition ohne Übertrag von C- und A-Feld gebildet. Stopzeichen ist, wie auch bei Steuer- und Langsatz, 16h.
- Mit dem Steuersatz werden immer neun Zeichen übertragen, als Startzeichen dient hier 68h. Das L-Feld (Längenfeld) besitzt immer den Wert drei. Die Checksumme wird über C-, A- und CI-Feld gebildet.



- Der Unterschied zum Steuersatz besteht beim Langsatz darin, daß Anwenderdaten nach dem CI-Feld übertragen werden können. Dementsprechend variieren die Länge des Telegramms und der Inhalt des L-Feldes, welches aus der Anzahl Anwenderdaten zuzüglich drei gebildet wird.

### 2.3.3. Bedeutung der Felder

- **C-Feld (Funktionsfeld, Control-Field)**

Das C-Feld dient dazu, die Funktion des Telegramms zu kennzeichnen. Es ermöglicht ferner, auf Verbindungsebene Aufruf- und Antwortrichtung zu unterscheiden. Tabelle 1 gibt einen Überblick über die erlaubten Funktionscodes.

Bezeichnung	C-Feld binär	C-Feld hex.	Telegramm	Beschreibung
SND_NKE	0100 0000	40	Kurzsatz	Initialisierung der Slaves
SND_UD	01FV 0011	43/53/63/73	Lang-/ Steuersatz	Anwenderdaten senden
REQ_UD2	01FV 1011	4B/5B/6B/7B	Kurzsatz	Abfrage von Daten der Klasse 2
RSP_UD	0000 1000	08	Lang-/ Steuersatz	Datenübertragung nach Abfrage

**Tabelle 1:** Bedeutung des C-Feldes (F: FCB-, V: FCV-Bit)

Die ersten drei Funktionen der obigen Tabelle können nur in Aufrufrichtung genutzt werden und besitzen folgende Bedeutung:

- 1) Ein SND\_NKE bewirkt bei erfolgreichem Empfang eine Initialisierung des Slaves auf Verbindungsebene, was einem Löschen des sogenannten FCB (frame count bit, Erklärung siehe unten) entspricht, und die Quittierung durch ein Einzelzeichen.
- 2) Ein SND\_UD wird dazu genutzt, dem Slave Anwendungsdaten zu senden. Besteht für das Endgerät die Möglichkeit, die Daten zu verarbeiten, erfolgreicher Empfang vorausgesetzt, so wird ein Einzelzeichen gesendet.
- 3) Ein REQ\_UD2 fordert das Endgerät auf, mit Daten der Klasse 2 zu antworten. Besitzt der Slave solche Daten nicht, antwortet er mit einem Einzelzeichen. Anderenfalls schickt dieser ein RSP\_UD. Bei fehlerhafter Übertragung auf Ebene 2 bleibt eine Antwort aus.

Das FCB alterniert bei erfolgreicher Kommunikation. Ein gleichbleibendes FCB fordert den Slave auf, nochmals das zuletzt gesendete Telegramm zu wiederholen. Mit gesetztem FCV-Bit (frame count valid bit) zeigt der Master an, daß das FCB Gültigkeit besitzt, anderenfalls ist es vom Slave zu ignorieren.

Das Ausbleiben einer Antwort des Slaves wird nach 330 Bitzeiten zuzüglich 50 ms angenommen. Der Master geht zunächst davon aus, daß ein Fehler in der Verbindungsschicht aufgetreten ist. Er wiederholt die Übertragung des gleichen Telegramms bis zu zweimal. Liegt die Antwort des Slaves bis dahin immer noch nicht vor, so wird eine Pause von 33 Bitzeiten auf dem Bus eingelegt. Genauso wird verfahren, wenn der Master ein fehlerhafte oder gestörte Antwort des Slaves empfängt.

- **A-Feld (address field, Adressfeld)**

Das M-Bus Protokoll läßt zwei verschiedene Arten der Adressierung zu. Besitzt das A-Feld den Wert 253, so handelt es sich um eine Sekundäradressierung auf Anwendungs- und Netzwerkebene (siehe entsprechende Kapitel).

Für die Adressierung auf Verbindungsebene stehen die Werte 1 bis 250 zur Auswahl. Diese Adresse wird in Auf- und Antwortrichtung zur Kennzeichnung des anzusprechenden bzw. antwortenden Slaves benutzt. Unkonfigurierte Slaves besitzen die Primäradresse null. Die Adressen 254 und 255 werden in Aufrufichtung eingesetzt, um alle angeschlossenen Slaves anzusprechen. Die Adresse 255 verlangt, daß keiner der Slaves, die Adresse 254 hingegen, daß alle Slaves antworten. Dieses führt bei zwei oder mehr angeschlossenen Endgeräten unweigerlich zur Kollision und ist daher nur für Testzwecke vorgesehen.

- **CI-Feld (control information field, Kennungsfeld)**

Das Kennungsfeld gehört zur Anwendungsebene und wird daher im Kapitel 2.8. behandelt.

## 2.4. Die Netzwerkebene

Die Netzwerkebene sorgt für den bestmöglichen Transportweg innerhalb eines Netzwerkes. Die Funktionen dieser Ebene werden bei der sogenannten Sekundäradressierung (notwendige Bedingung A-Feld = 253) benötigt. Die Adresse des Slaves ist in diesem Fall auf Anwendungsebene, nicht aber auf Verbindungsebene bekannt. Die Schicht 3 sorgt nun für die entsprechende Verbindung. Auf diese Form der Adressierung wird in Kapitel 2.8.4.3. *Sekundäradressierung* näher eingegangen.

## 2.5. Die Transportebene

Die letzte der transportorientierten Schichten dient dazu, die zu übertragene Daten in geeignete Pakete einzuteilen, und die Informationen durch das Netzwerk zu führen. Diese Ebene wird vom M-Bus Protokoll nicht benutzt.

## 2.6. Die Sitzungsebene

Diese Ebene bleibt im M-Bus Protokoll ebenfalls leer. Sie stellt Dienstleistungen zur Eröffnung, Durchführung und Beendigung einer Kommunikation bereit.

## 2.7. Die Darstellungsebene

Die Darstellungsebene, vom M-Bus Protokoll nicht genutzt, wandelt unter anderem die empfangenen Daten in ein Format um, daß die Anwendung interpretieren kann.

## 2.8. Die Anwendungsebene

Diese Schicht dient als Schnittstelle zum Anwender. Die Dienstleistungen beziehen sich auf den Umgang mit dem System. Zum Entwickeln von Anwendungsprogrammen ist die Kenntnis des Protokolls dieser Schicht nötig.

### 2.8.1. Das CI-Feld

Das CI-Feld macht Aussagen über die Anwendung des gesendeten Telegramms und kennzeichnet den Modus mit dem Felder codiert werden, die aus mehreren Bytes bestehen. Der Modus 1 sagt aus, daß in solchen Feldern immer das niederwertigste Byte zuerst gesendet wird. Modus 2 hingegen legt fest, daß das höchstwertigste Byte als erstes übertragen wird. Beispiel: Ein Feld besteht aus 4 Bytes. Es enthält die Nummer 12345678. Die Übertragung im Mode 1 erfolgt dann in der Reihenfolge: 78, 56, 34, 12. Eine ausführliche Besprechung und Darstellung der CI-Felder in Aufrufichtung erfolgt in dem Kapitel 2.8.4. *Senden von Daten in Aufrufichtung*.

Die Tabelle 2 zeigt die benutzten CI-Codes in Antwortichtung. Mit dem CI-Feld 70h ist es möglich einen den gesamten Slave betreffenden allgemeinen Fehler an den Master zu übermitteln. Das Senden eines Alarmzustandes wird mit dem CI-Feld 71h gekennzeichnet. Beide Anwendungen sind für diese Diplomarbeit nicht relevant. Die Kennungen 72h/76h und 73h/77h werden in den folgenden beiden Kapiteln erläutert.

CI-Feld (hex.)		Anwendung
Mode 1	Mode 2	
70		Senden eines allgemeinen Fehlerzustandes des Slaves
71		Senden eines Alarmzustandes

72	76	Antwort mit variabler Datenstruktur
73	77	Antwort mit fester Datenstruktur

**Tabelle 2:** CI-Codes in Antwortrichtung

### 2.8.2. Die feste Datenstruktur

Das M-Bus Protokoll bietet die Möglichkeit, variable und feste Datenstrukturen zu benutzen. In Aufrufrichtung kann, bei einem SND\_UD, aber nur die variable benutzt werden. Diese Struktur ist für zukünftige Anwendungen auch auf Slave-seite zu empfehlen. Die Merkmale der festen Datenstruktur sind die festgelegte Länge, die Einschränkung, nur zwei Zählerstände gleichzeitig übertragen zu können, und natürlich das CI-Feld, das je nach Modus den Wert 72h (Mode 1) oder 76h (Mode 2) annehmen kann. Der Aufbau der Anwenderdaten ist Abb. 3 zu entnehmen. Die Daten werden in dieser Reihenfolge von links nach rechts gesendet.

Identifikationsnr	Zugriffszahl	Status	Medium/Einheit	Zähler 1	Zähler 2
4 Byte	1 Byte	1 Byte	2 Byte	4 Byte	4 Byte

**Abb. 3:** Datenaufbau bei fester Struktur in Antwortrichtung (aus [2])

Die Identifikationsnr. wird mit 4 Byte BCD codiert. Das folgende Byte bezieht sich auf die Anzahl der Zugriffe des Masters auf Daten des jeweiligen Slaves. Die Bedeutung und Codierung des Statusfeldes ist der Tabelle im Anhang zu entnehmen. Im Medium/Einheit Feld wird immer das niederwertige Byte voran gestellt. Die Codierung geht aus der Abb. 4 hervor. Die Bits 7, 8, 15, 16 bilden zusammen den Code für das Medium beider Zähler. Die Tabellen zu den darstellbaren Medien und Einheiten befinden sich im Anhang.

Bit Nr.	höherwertiges Byte								niederwertiges Byte							
	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Codierung	Medium		Einheit Zähler 2						Medium		Einheit Zähler 1					

**Abb. 4:** Bedeutung der Bits im Feld Medium/Einheit



**Abb. 5:** *Fester Datenblock der variablen Struktur (aus [2])*

Die ersten 8 Bytes dieses Blocks bilden eine slavespezifische Sekundäradresse, die vom Hersteller vergeben wird. Mit dieser ist es für den Master möglich, eine Adressierung auf Anwendungsebene durchzuführen (siehe auch Kapitel 2.8.4.3. *Sekundäradressierung*).

Das Feld Hersteller ermöglicht die Codierung von drei Großbuchstaben, die nach folgender Formel einen Wert im Word-Format ergeben:

$$\begin{aligned} \text{Hersteller} &= [\text{ASCII (1. Buchstabe)} - 64] * 32 * 32 \\ &+ [\text{ASCII (2. Buchstabe)} - 64] * 32 \\ &+ [\text{ASCII (3. Buchstabe)} - 64] \end{aligned}$$

Die Codierungen der Felder Status und Medium sind den Tabellen im Anhang zu entnehmen. Signatur wird bisher noch nicht genutzt und erhält den Wert 00 00h.

### Aufbau der variablen Datenblöcke

Abbildung 6 zeigt den Aufbau eines Datenrecords. Die Übertragung erfolgt von links nach rechts. Es wird jeweils ein Wert (Datum) übertragen, dessen Bedeutung aus dem vorangehenden DRH hervorgeht. Dieser besteht aus zwei Teilen, dem Dateninformationsblock (DIB) und dem Wertinformationsblock (VIB).

Data Record Header (DRH)				
Data Information Block (DIB)		Value Information Block (VIB)		
DIF	DIFE	VIF	VIFE	Datum
1 Byte	0 - 10 Byte	1 Byte oder String	0 - 10 Byte	0 - 191 Byte

**Abb. 6:** *Struktur eines Datenrecords (aus [2])*

#### a) Der Dateninformationsblock:

Der DIB besteht aus einem DIF (siehe Abb. 7), dem bis zu 10 DIFE (siehe Abb. 8) folgen können. Der Übertragungswunsch einer DIFE ist am gesetzten Erweiterungsbit des Vorgängerzeichens (DIF bzw. DIFE) zu erkennen.

Bit Nr.	7	6	5	4	3	2	1	0
Bedeutung	Erweiterung des DIF	LSB der Speichernr.	Funktionsfeld		Datenfeld			

**Abb. 7:** Codierung des Data Information Field (DIF)

Bit Nr.	7	6	5	4	3	2	1	0
Bedeutung	Erweiterung der DIFE	Unitnummer	Tarifnummer		Speichernummer (höhere Bits)			

**Abb. 8:** Codierung der Data Information Field Extension (DIFE)

Die Unitnummer ermöglicht es, Untereinheiten eines Slaves zu unterscheiden. Die Nummer wird aus bis zu 10 Bits gebildet, wobei das LSB in der ersten DIFE übertragen wird. Mit Hilfe der Speichernummer lassen sich historische Werte voneinander unterscheiden. Das LSB der Speichernummer wird im DIF, bis zu 40 weitere Bits in den folgenden 10 DIFE übertragen. Bis zu 20 Bits sind für die Tarifnummer vorgesehen. Sie kennzeichnet, daß ein Wert über eine bestimmte Zeitspanne aufgenommen wurde (z.B. Tag und Nachttarif). Das Funktionsfeld charakterisiert die Art der Daten. Die Codierung gibt Tabelle 3 wieder.

Code	Bedeutung	Code	Bedeutung
00b	augenblicklicher Wert	01b	Maximum
01b	Minimum	11b	fehlerbehafteter Wert

**Tabelle 3:** Codierung des Funktionsfeldes (aus [2])

Die Codierung des 4 Bit umfassenden Datenfeldes geht aus Tabelle 4 hervor. Es bestimmt Länge und Darstellung der übertragenen Daten.

Länge der Daten in Byte	Code dual	Bedeutung	Code dual	Bedeutung
0	0000	keine Daten	1000	Selektion
1	0001	8 Bit Integer	1001	2 Stellen BCD
2	0010	16 Bit Integer	1010	4 Stellen BCD
3	0011	24 Bit Integer	1011	6 Stellen BCD
4	0100	32 Bit Integer	1100	8 Stellen BCD
4	0101	32 Bit Real	1101	variable Länge
6	0110	48 Bit Integer	1110	12 Stellen BCD

8	0111	64 Bit Integer	1111	Spezielle Funktionen
---	------	----------------	------	----------------------

**Tabelle 4:** Codierung des Datenfeldes (aus [2])

Bemerkungen zu Tabelle 4:

- Alle Integer-Typen sind vorzeichenbehaftet.
- Die Codierung der einzelnen Typen (Real, Integer, BCD) wird im Anhang unter Codierung von Daten dargestellt.
- Das Datenfeld Selektion (Code 1000b) wird in dieser Diplomarbeit nicht genutzt.
- Aus dem Datenfeld " variable Länge " geht weder Art noch Länge der Daten hervor. Deshalb wird dem Datum ein zusätzliches Byte (hier LVAR genannt) vorangestellt, welches die fehlenden Aussagen gemäß Tabelle 5 übernimmt.

LVAR hexadezimal	Bedeutung	
	Art der Datencodierung	Länge
00..BF	ASCII-String	LVAR Zeichen
C0..CF	BCD positiv	(LVAR - C0) * 2 Stellen
D0..DF	BCD negativ	(LVAR - D0) * 2 Stellen
E0..EF	binär	(LVAR - E0) Bytes
F0..FA	Fließkomma	(LVAR - F0) Bytes (noch näher zu definieren)
FB..FF	reserviert	

**Tabelle 5:** Bedeutung des ersten Datenbytes bei variabler Länge

Entspricht die Codierung des Datenfeldes 1111b, so ist der DIF nicht wie oben erklärt, sondern wie folgt zu deuten:

- DIF = 0Fh: Hiernach folgen herstellerspezifische Daten. Da diese nur am Ende der Anwenderdaten erlaubt sind, kann ihre Länge aus dem L-Feld berechnet werden. Der VIB wird in diesem Fall nicht benutzt.
- DIF = 1Fh: Entspricht DIF = 0Fh mit dem Zusatz, daß mehr Datenrecords im nächsten Telegramm folgen. Der Einsatz dieses DIF ist immer dann sinnvoll, wenn ein Slave mehr Anwenderdaten besitzt, als in einem Telegramm Platz finden. Der Master erhält die noch fehlenden Informationen durch Auslesen mittels eines weiteren REQ\_UD2.



- DIF = 2Fh: Dieses DIF wird vom Empfänger ignoriert. Das nächste Zeichen, für den Fall, daß noch Anwenderdaten folgen, ist wiederum ein DIF.
- DIF = 3Fh..6Fh: reserviert
- DIF = 7Fh: Der Master fordert den Slave auf, bei einem REQ\_UD2 alle Anwenderdaten zu senden, die dieser besitzt. Der VIB bleibt leer. (siehe auch Kapitel 2.8.4. *Senden von Daten in Aufrufrichtung*).

#### b) Der Wertinformationsblock

Der zweite Teil des DRH legt im allgemeinen die physikalische Einheit und die Wertigkeit des gesendeten Datums fest. Er besteht, analog zum Aufbau des DIB, aus einem VIF und bis zu 10 VIFE. Die Benutzung der VIFE wird durch ein gesetztes Erweiterungsbit des Vorgängerzeichens eingeleitet. Die Codierung der Felder gibt Abb. 9 wieder.

Bit Nr.	7	6	5	4	3	2	1	0
Bedeutung	Erweiterungsbit	physikalische Einheit und Wertigkeit						

**Abb. 9:** Codierung von VIF und VIFE

Im Anhang befindet sich eine Tabelle der definierten, sogenannten primären VIF-Codes. Solche, die eine besondere Bedeutung besitzen, werden im folgenden erklärt:

- VIF = 7Ch / FCh (VIF im folgenden String): Die Bedeutung des VIF ist einem String zu entnehmen, der direkt anschließend folgen muß. Die Länge ist dem ersten Zeichen dieses Strings zu entnehmen. Modus 1 bedeutet in diesem Fall, daß die Übertragung der Zeichenkette von rechts nach links verläuft.
- VIF = FDh (VIF Erweiterung): Das eigentliche VIF wird erst mit dem nächsten Zeichen übertragen. Dies ergibt die Möglichkeit der Definition von weiteren 128 "sekundären" VIF- Codes . Die Tabelle hierzu befindet sich ebenfalls im Anhang.
- VIF = 7Eh (jedes VIF): Im Zusammenhang mit dem Datenfeld = 1000b kann der Master hiermit den Slave konfigurieren. (siehe auch 2.2.7.4. *Senden von Daten in Aufrufrichtung*) .
- VIF = 7Fh (herstellerspezifisch): Das Datum dieses Records ist herstellerspezifisch. Die Position innerhalb der Anwenderdaten ist nicht festgelegt. (Im Gegensatz zu DIF = 0Fh / 1Fh , siehe oben)

Bei gesetztem Erweiterungsbit des VIF folgen ein oder mehrere VIFE, die zu drei verschiedenen Zwecken eingesetzt werden:

### 1. Erweiterung der Bedeutung des VIF

Hierzu sind Codes aus der Tabelle im Anhang zu entnehmen. Die mit diesem Zeichen ▲ markierten VIFE nehmen die Rolle der VIF bezüglich des übertragenen Wertes des Records ein.

### 2. Übermitteln eines recordspezifischen Fehlers

Mit einer solchen VIFE wird ein Fehler ausgedrückt, der sich nur auf diesen Datenrecord bezieht. Im Anhang befindet sich eine Tabelle der möglichen Fehlercodes.

### 3. Bilden eines Objektes

Ein Objekt (im Sinne des M-Bus) besteht aus einem Datum und einer Aktion (eben der VIFE), die die Verwendung des Datums angibt. Diese Art von Record wird nur vom Master in Richtung Slave eingesetzt, und ist eine Möglichkeit diesen zu konfigurieren. Weitere Informationen und die Tabelle der Aktionen folgen im Kapitel 2.8.4.1. *Konfiguration von Slaves.*

Beispiel 2: Senden eines RSP\_UD mit variabler Datenstruktur (Werte links hex.)

68            Start    : Beginn des Telegramms, Format Lang- oder Steuersatz  
 20 20        L-Feld   : Länge der Anwenderdaten plus 3  
 68            Start    : 2. Startzeichen  
 08            C-Feld   : RSP\_UD  
 05            A-Feld   : Primäradresse = 5

*Beginn Anwenderdaten:*

72            CI-Feld : variable Datenstruktur, Modus 1

*fester Datenblock:*

39 12 52 37 Identifikationsnr. = 37521239  
 22 04        Hersteller        = 0422h entspricht ABC  
 01 02        Version            = 1            Medium        = Elektrizität  
 06 00        Zugriffszahl       = 6            Status        = 00h  
 00 00        Signatur            = 00 00h

*1. Datenrecord:*

02            DIF     : Datenformat: 16 Bit Integer  
 FC            VIF     : VIF im folgenden String, Erweiterungsbit gesetzt  
 04            String   : Länge = 4 Zeichen  
 6D 6A 4F 6D String   : Inhalt = "mOhm"  
 2F            VIFE    : pro K (Kelvin)  
 02 00        Datum    = 2 (mΩ/K)

*2. Datenrecord:*

0D            DIF     : keine Erweiterung, Datenformat: variabel  
 28            VIF     : Leistung in mW  
 C2            LVAR    : das folgende Datum wird mit 4 Stellen BCD codiert  
 12 55        Datum    = 5512 (mW )

*3. Datenrecord:*

0F            DIF     : herstellerspezifische Daten folgen (kein VIB)  
 02 03 07     Datum    : Länge des Datums läßt sich aus dem L-Feld berechnet

*Ende der Anwenderdaten*

BF            Checksumme

16            Stop    : Ende des Telegramms

### 2.8.4. Senden von Daten in Aufrufriichtung

Ein SND\_UD vom Master wird für verschiedene Anwendungen eingesetzt, die sich über das CI-Feld unterscheiden lassen. Tabelle 6 zeigt die verwendeten Codes, die in den folgenden Unterkapiteln erläutert werden.

CI-Feld		Anwendung
Mode 1 hex.	Mode 2 hex.	
50		Reset auf Anwendungsebene
51	55	Daten werden vom Master zum Slave gesendet (2.2.7.4)
52	56	Selektion von Slaves über Sekundäradressen (2.2.7.6)
B8		Setzen der Baudrate auf 300 Baud
B9		Setzen der Baudrate auf 600 Baud (nicht empfohlen)
BA		Setzen der Baudrate auf 1200 Baud (nicht empfohlen)
BB		Setzen der Baudrate auf 2400 Baud
BC		Setzen der Baudrate auf 4800 Baud (nicht empfohlen)
BD		Setzen der Baudrate auf 9600 Baud
BE		Setzen der Baudrate auf 19200 Baud (nicht empfohlen)
BF		Setzen der Baudrate auf 38400 Baud

**Tabelle 6:** CI-Codes in Aufrufriichtung

#### 2.8.4.1. Konfiguration der Slaves

##### Ändern der Baudrate

Jeder Slave muß in der Lage sein mit der Baudrate von 300 Bd zu kommunizieren. Das Ändern der Baudrate durch den Master erfolgt mit einem Steuersatz, der ein entsprechendes CI-Feld (siehe Tabelle 6 untere Hälfte) enthält. Den korrektem Empfang dieses Telegramms quittiert der Slave durch ein Einzelzeichen mit der bisher verwendeten Baudrate, unabhängig davon, ob er überhaupt in der Lage ist mit der neuen zu kommunizieren.

### Konfiguration der Anwendungsdaten

Die CI-Felder 51h (Modus 1) und 52h (Modus 2) dienen dazu, Daten zum Slave zu senden, um diesen zu konfigurieren.

- Auslesen aller Daten

Um den Slave aufzufordern mit seinem gesamten Datenvorrat zu antworten, schickt der Master ein Telegramm, das ein Datenrecord enthält, das nur aus dem DIF = 7Fh besteht.

- Konfiguration mit Aktionen

Jeder Datenrecord in Aufrufrichtung kann eine Aktion beinhalten, die angibt wie der Slave diesen Datenrecord verarbeiten soll. Eine Aktion ist eine VIFE und demnach Bestandteil des VIB. Tabelle 7 gibt Aufschluß über die einsetzbaren Aktionen.

VIFE-Code (binär)	Aktion	Beschreibung
E000 0000	Schreibe(/Ersetze) Daten	schreiben (ersetzen) von Daten
E000 0001	Addiere Wert	addiere Daten zu alten Daten
E000 0010	Subtrahiere Wert	subtrahiere Daten von alten Daten
E000 0011	OR (Setze Bits)	Daten OR alte Daten
E000 0100	AND	Daten AND alte Daten
E000 0101	XOR (Toggle Bits)	Daten XOR alte Daten
E000 0110	AND NOT (Lösche Bits)	NOT Daten AND alte Daten
E000 0111	Lösche	lösche Daten
E000 1000	Erzeuge Eintrag	erzeuge neuen Datenrecord
E000 1001	Lösche Eintrag	lösche bestehenden Datenrecord
E000 1010	Reserviert	
E000 1011	Friere Daten ein	speichere Daten mit Speichernr.
E000 1100	Füge zur Ausgabeliste hinzu	Datenrecord bei RSP_UD hinzufügen
E000 1101	Lösche von Ausgabeliste	Datenrecord bei RSP_UD löschen
E000 111x	Reserviert	
E001 xxxx	Reserviert	

**Tabelle 7:** VIFE-Codes für Aktionen (aus [1])

Der erste Teil der Tabelle (Trennung durch Doppelstrich), beeinflusst bereits vorhandene Informationen bzw. Werte des Slaves. Wird keine Aktion im Telegramm benutzt, ist der Grundzustand die Aktion "Schreibe(/Ersetze) Daten". Einige Slaves lassen es zu, daß ihre Primäradresse, die Identifikationsnummer, oder die komplette Sekundäradresse geändert werden können. Hierzu sind folgende Telegramme zu nutzen:

Änderung der Primäradresse (alle Werte hex.):

68 06 06 68	Kopf des Telegramms		
53	C-Feld	:	SND_UD
XX	A-Feld	:	XX bisherige Primäradresse eintragen
51	CI-Feld	:	Master sendet Daten
01	DIF	:	Datenformat : 8 Bit Integer
7A	VIF	:	Busadresse
YY	Datum	:	YY ist die neue Primäradresse
CS 16	Checksumme, Stop		

Änderung der Identifikationsnummer (alle Werte hex.):

68 09 09 68	Kopf des Telegramms		
53	C-Feld	:	SND_UD
PADR	A-Feld	:	Primäradresse
51	CI-Feld	:	Master sendet Daten
0C	DIF	:	Datenformat : 8 Stellen BCD
79	VIF	:	Identifikation $\bar{\quad}$
X4 X3 X2 X1	Datum	:	Identifikationsnummer = X1 X2 X3 X4
CS 16	Checksumme, Stop		

Änderung der kompletten Sekundäradresse (alle Werte hex.):

68 0D 0D 68	Kopf des Telegramms		
53	C-Feld	:	SND_UD
PADR	A-Feld	:	Primäradresse
51	CI-Feld	:	Master sendet Daten
07	DIF	:	Datenformat : 64 Bit Integer
79	VIF	:	Identifikation $\bar{\quad}$

X4 X3 X2 X1	Datum	:	Identifikationsnummer = X1 X2 X3 X4
H2 H1	Datum	:	Hersteller = H1 H2
VV	Datum	:	Version
MM	Datum	:	Medium
CS 16	Checksumme, Stop		

- ▭ Die Unterscheidung ob die Änderung der Identifikationsnummer oder der gesamten Sekundäradresse erwünscht ist, erfolgt über das Datenfeld (8 Stellen BCD bzw. 64 Bit Integer).

Mit den Aktionen des zweiten Teils ist es möglich, einen intelligenten Slave zu veranlassen neue Einträge d.h. Datenrecords zu erzeugen ("Erzeuge Eintrag") oder zu löschen ("Lösche Eintrag"). Mit "Friere Daten ein" ist es möglich einen Slave Werte zusammen mit der übergebenen Speichernummer (aus dem DIB) sichern zu lassen. Der zu speichernde Wert wird durch den gesamten DRH (mit Ausnahme der Aktion, des Datenfeldes und der Speichernummer) charakterisiert. Voraussetzung ist, das der zu speichernde Wert vormals mit der Speichernummer 0 dargestellt wurde.

Der letzte Teil beeinflusst die Datenausgabe der Slaves. Es kann ihnen mitgeteilt werden, daß er bestimmte Records senden ("Füge zur Ausgabeliste hinzu"), andere hingegen weglassen soll ("Lösche von Ausgabeliste").

Besonders wirkungsvoll sind die Aktionen im Zusammenhang mit dem VIF-Code 7Eh (Jedes VIF). Mit einem Datenrecord des Masters können somit alle Informationen des Slaves, mit dem entsprechenden DIB, angesprochen und konfiguriert werden.

Beispiel 3: Konfiguration eines Slaves mittels Aktionen (Werte hexadezimal):

Bei einem RSP\_UD schickt der Slave folgende Anwenderdaten:

72	CI-Feld	:	variable Datenstruktur, Modus 1
----	---------	---	---------------------------------

*fester Datenblock:*

44 33 22 11	Identifikationsnr	=	11223344
24 40	Hersteller	=	PAD
02 01	Version	=	1 Medium : Öl
03 00	Zugriffszahl	=	3 Status = 00h
00 00	Signatur	=	00 00h

*1. Datenrecord:*

82	DIF	:	Speichernummer = 0, Datenformat: 16 Bit Integer
40	DIFE	:	Unitnummer = 1
13	VIF	:	Volumen in l
10 00	Datum	:	16 Liter (dezimal)

*2. Datenrecord:*

0C            DIF     : Speichernummer = 0, Datenformat: 8 Stellen BCD  
 03            VIF     : Energie in Wh  
 27 36 03 00   Datum   : 33627 Wh (dezimal)

*Ende der Anwenderdaten*

Zur Konfiguration dieses Slaves schickt der Master folgende Anwenderdaten (hex.):

51            CI-Feld : Sende Daten

*1. Datenrecord: Es wird der Bezug zum 2.Datenrecord des Slaves hergestellt, der augenblickliche Wert soll in der Speichernummer 1 abgelegt werden*

40            DIF     : Speichernummer: 1, keine Daten  
 83            VIF     : Energie in Wh  
 0B            VIFE    : Friere Daten ein

*2. Datenrecord: Es wird der Bezug zum 1. Datenrecord des Slaves hergestellt, um die Ausgabe dieses Records bei dem nächsten RSP\_UD zu unterdrücken*

80            DIF     : Datenformat        : keine Daten  
 40            DIFE    : Unitnummer         = 1  
 93            VIF     : Volumen in l  
 0D            VIFE    : Lösche von Ausgabeliste

*Ende der Anwenderdaten*

#### **2.8.4.2. Reset auf Anwendungsebene**

Mit einem solchen Reset wird der Slave aufgefordert in seinen Grundzustand auf Anwendungsebene überzugehen. Es wird ein Steuersatz mit CI-Feld 50h genutzt. Der Reset kann beispielsweise die Ausgabe der Daten beeinflussen.

#### **2.8.4.3. Selektion und Sekundäradressierung**

Die hier beschriebene Form der Adressierung findet auf Netzwerkebene statt. Die Felder Identifikationsnummer, Hersteller, Version und Medium der variablen Datenstruktur dienen zusammen als sogenannte Sekundäradresse die sich in der Anwendungsebene befindet. Um einen Slave über diese Adresse anzusprechen, muß sich dieser im selektierten Zustand befinden. Zu diesem Zweck sendet der Master ein Langsatz-Telegramm an die Primäradresse 253 (A-Feld), daß als Anwenderdaten das CI-Feld 52h (Modus 1) oder 56h (Modus 2) und daran anschließend die oben beschriebene Sekundäradresse enthält. Diese



Kombination aus CI- und A-Feld veranlasst die Slaves, ihre Sekundäradresse mit der gesendeten zu vergleichen. Bei Übereinstimmung schickt der entsprechende Slave ein Einzelzeichen. Der so selektierte Slave ist nun über die Netzwerkebene durch die Adresse 253 mit dem Master verbunden. Dieser kann dem Slave Telegramme jeder Art an die Adresse 253 schicken. Mögliche Antworten des Slaves (per RSP\_UD) werden ebenfalls mit der Primäradresse 253 versehen. Eine erneute Selektion mit einer geänderten Sekundäradresse oder ein SND\_NKE an 253 bewirkt, daß der Slave deselektiert wird. Die Verbindung zwischen Master und diesem Slave ist gelöst.

Auf Seite des Masters ist es möglich, in der Sekundäradresse sogenannte Wildcards, eine Art Platzhalter, zu vergeben. Die Entscheidung des Slaves, ob er selektiert ist oder nicht, ist unabhängig von solchen Zeichen. Als Wildcard wird für die Identifikationsnummer der Wert Fh für die anderen Felder FFh benutzt. Hiermit ist es möglich, mehrere Slaves gleichzeitig zu selektieren, was allerdings zu Kollisionen auf dem Bus führen kann, wenn z.B. der Master die selektierten Slaves auffordert mit ihren Daten zu antworten. In solchen Fällen muß der Master dafür sorgen, daß nur ein Slave selektiert ist.

Der Vorteil der Sekundäradressierung besteht darin, daß die Anzahl der möglichen Adressen und somit der Anzahl der möglichen Slaves auf dem Bus enorm anwächst. Allein mit der Identifikationsnummer (8 Stellen BCD) lassen sich 100 Millionen verschiedene Werte bilden. Weiterhin entfällt die Vergabe von Primäradressen.

Beispiel 4: Selektion eines Slaves

Die fett markierten Daten stellen die Sekundäradresse dar. Alle Werte der linken Seite hexadezimal.

68 0B 0B 68	Kopf des Telegramms
53	C-Feld : SND_UD
FD	A-Feld : Adresse 253
52	CI- Feld : Selektion von Slaves (Modus 1)
<b>79 68 35 24</b>	Identifikationsnr = 24356879
<b>24 40</b>	Hersteller = 4024h: PAD
<b>01 07</b>	Version = 1 Medium : Wasser
48	Checksumme
16	Stopzeichen

Beispiel 5: Selektion mit Wildcards

Die Sekundäradresse ist fett, die Wildcards kursiv dargestellt. Alle Slaves des Herstellers PAD jeder Version, die als Medium Wasser und eine Identifikationsnummer im Bereich von 32753970 bis 32753979 besitzen werden selektiert.

Alle Werte der linken Seite hexadezimal.

68 0B 0B 68	Kopf des Telegramms
53	C-Feld : SND_UD
FD	A-Feld : Adresse 253
52	CI- Feld : Selektion von Slaves (Modus 1)
<b>7F 39 75 32</b>	Identifikationsnr = 3275397F (F ist Platzhalter für Werte 0..9)
<b>24 40</b>	Hersteller = 4024h: PAD
<b>FF 07</b>	Version : jede Medium : Wasser
6B	Checksumme
16	Stopzeichen

## 3. Eine intelligente Ausleseeinheit auf dem M-Bus

### 3.1. Einleitung

Das Ziel dieses Teils der Diplomarbeit war die Konzeption und Entwicklung einer sogenannten intelligenten Ausleseeinheit auf dem M-Bus. Mit Hilfe dieser soll es möglich sein, möglichst viele Meßwerte bzw. Zählerstände von Endgeräten über einen beliebig langen Zeitraum zu speichern und auf Grundlage der in den Slaves existierenden Informationen, neue innerhalb der Ausleseeinheit zu bilden. Die gespeicherten und gebildeten Daten sollen durch einen Master jederzeit abgefragt und beeinflusst werden können. Dieser braucht nicht ständig an die Ausleseeinheit angeschlossen zu sein. Eine Kommunikation über ein Modem ist möglich.

Die Ausleseeinheit wird als Schnittstelle zwischen dem übergeordneten Master und den Slaves plaziert (siehe Abb. 10). Sie konzentriert Daten bzw. Informationen von bis zu 250 Endgeräten in ihrem Speicher, und wird deshalb auch als Konzentrator bezeichnet. Eine Information besteht aus einem Wert (Zählerstand, Meßwert) und dessen Bedeutung (spezifiziert durch den DRH). Jede dieser Informationen wird zusammen mit der Sekundäradresse des entsprechenden Slaves abgelegt und im folgenden als sekundär bezeichnet. Sie lassen sich einer oder mehrerer der folgenden Gruppen zuordnen:

- Tarifinformationen
- Extremwertinformationen
- historische Informationen

Die Kennzeichnung der Elemente einer jeden Gruppe erfolgt durch den DIB. Historische Informationen sind durch eine Speichernummer, tarifliche durch eine Tarifnummer größer null gekennzeichnet. Eine Extremwertinformation wird durch ein entsprechendes Funktionsfeld bezeichnet.

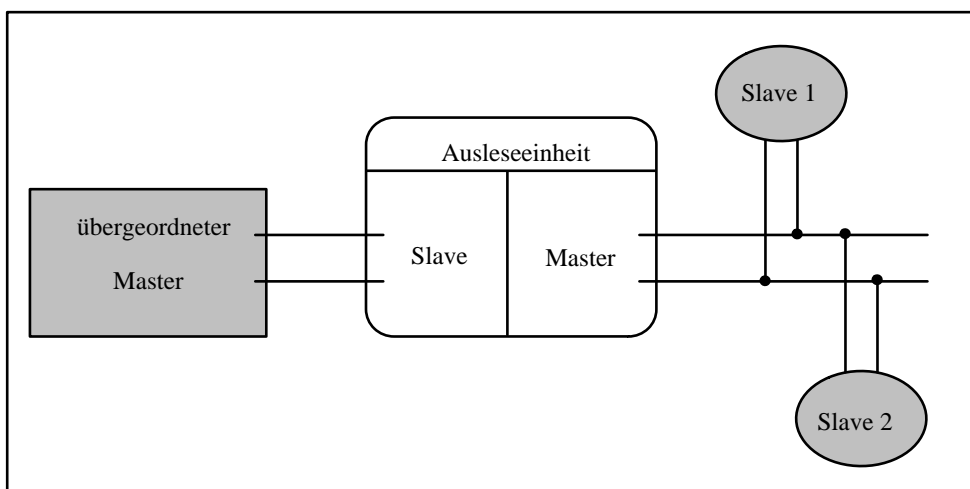


Abb. 10: Aufbau des M-Bus mit Ausleseeinheit

Ein Endgerät sendet i.a. bei Auslesung immer die gleiche Art von Informationen. Aufgabe der Ausleseeinheit (des Konzentrators) ist es auf Grundlage dieser neuen Informationen (Einträge) zu bilden und zu speichern, die eine Einteilung in die oben angegebenen Gruppen zulassen. Hinsichtlich dieser Fähigkeit läßt sich die Ausleseeinheit vom übergeordneten Master programmieren. Diese Art von Informationen werden als virtuell bezeichnet, da sie scheinbar von den Endgeräten selbst stammen. Angestrebt wird ein hohes Maß an Universalität. Jeder Slaveeintrag soll individuell konfiguriert und ausgelesen werden können (siehe Kapitel 3.3. *Die sekundären Informationen*).

Der Konzentrador beinhaltet auch folgende sogenannte primäre Informationen, die die Ausleseeinheit als solche und nicht die Slaveeinträge betreffen:

- Systemzeit
- Zahl der angeschlossenen Slaves
- freier Speicherplatz

Diese Informationen lassen sich ebenfalls auslesen und teilweise konfigurieren (siehe Kapitel 3.2 *Die primären Informationen*).

Der übergeordnete Master kann nur mit der Ausleseeinheit, nicht mit den Slaves direkt kommunizieren. Telegramme werden mit einer festen Baudrate von 9600 Baud im M-Bus Protokoll mit variabler Datenstruktur und Modus 1 gesendet.

In Richtung der Slaves fungiert die Ausleseeinheit als Master und kann diese auslesen. Auslesezeitpunkt und die Verwendung der empfangenen sekundären Daten hängen von der Programmierung der Ausleseeinheit durch den übergeordneten Master ab. Die Ausleseeinheit versteht sowohl variable als auch feste Datenstruktur im Modus eins und zwei und alle Datenformate. Feste Datenstrukturen werden in variable, Modus zwei in eins umgewandelt. Die Kommunikation zwischen Ausleseeinheit und den Slaves findet mit deren höchst möglicher Baudrate (max. 9600 Bd) statt. Antwortet ein Slave nicht, so wird eine Fehlermeldung erzeugt, die als sekundäre Information zu diesem Slave in der Ausleseeinheit abgelegt wird und somit vom übergeordneten Master mit ausgelesen wird.

Als Ausleseeinheit wird ein V25 Einplatinenrechner eingesetzt, der unter anderem über zwei serielle Schnittstellen, einen Timerbaustein und bis zu 768 kByte Speicher verfügt. Die Software der Ausleseeinheit ist in Turbo Pascal geschrieben. Das Programm besteht aus mehreren Units die sich an das OSI-Referenz Modul anlehnen.

Als Mastersoftware wird das Programm "M-Bus Application" von Andreas Papenheim verwendet, das durch zwei Punkte erweitert zur Programmierung der Ausleseeinheit geeignet ist.

## 3.2. Die primären Informationen

Um primäre Informationen, die die Ausleseeinheit selbst betreffen, auszulesen, zu konfigurieren und zu initialisieren ist die Primäradresse der Ausleseeinheit (Grundzustand: null) im jeweiligen Telegramm zu benutzen.

### 3.2.1. Auslesen

Bei einem REQ\_UD2 des übergeordneten Masters an die Primäradresse der Ausleseeinheit, antwortet diese mit den oben angegebenen primären Informationen. Es wird variables Protokoll im Modus 1 benutzt. Der freie Speicherplatz ist abhängig von der Anzahl der angeschlossenen Slaves und der Menge an Informationen zu jedem Endgerät.

Beispiel 6: Antwort der Ausleseeinheit nach einem REQ\_UD2 an die Primäradresse (nur Anwenderdaten, alle Werte links hexadezimal)

02	DIF	: 16 Bit Integer
7C	VIF	: VIF im folgenden String
04		
65 74 79 42	String	: "Byte"
20 4E	Datum	: 20.000 Byte
04	DIF	: 32 Bit Integer
6D	VIF	: Datum / Uhrzeit
00 09 E2 B5	Datum	: 02.05.1995 09:00 Uhr Winterzeit
01	DIF	: 8 Bit Integer
7C	VIF	: VIF im folgenden String
08		
29 73 28 65		
76 61 6C 53	String	: "Slave(s)"
BE	Datum	: 190 Slaves

### 3.2.2. Konfiguration

Die Konfiguration erfolgt mittels Aktionen. Verwendet wird nur die Aktion "Schreibe(/Ersetze) Daten". Grundzustand der konfigurierbaren Daten ist der folgende (Darstellung im Modus 1):

- Primäradresse : 0
- Sekundäradresse : Identifikationsnr. = 00 00 00 00  
Hersteller (hex.) = C9 55 (UNI)  
Version = 1  
Medium = 0Eh (Bus / System)

Die Systemzeit, ebenfalls konfigurierbar, ist eine fortlaufende Größe. Eine Voreinstellung dieser Zeit ist auch über das Betriebssystem des V25-Einplatinenrechners möglich. Die zu sendenden Telegramme sind dem Kapitel 2.8.4.1. *Konfiguration der Slaves* zu entnehmen. Es ist zu beachten, daß das Ändern der Systemzeit mit dem Verlust sämtlicher sekundärer Informationen verbunden ist.

### 3.2.3. Initialisierung

Die Ausleseeinheit als Endgerät kann sowohl auf Verbindungsebene als auch auf Anwendungsebene vom übergeordneten Master initialisiert werden.

Ein Reset auf Anwendungsebene veranlaßt die Ausleseeinheit eine Primär- und Sekundäradressensuche der angeschlossenen Slaves mit 9600 bis 300 Baud durchzuführen. Die Anzahl der gefundenen Endgeräte wird als primäre Information gespeichert. Die Slaves werden vom Konzentrator ausgelesen und eine Liste der Slavemerkmale angelegt. Hierzu gehören unter anderem Angabe über den festen Datenblock, über die Art der Datenstruktur und über die Art der Informationen, die der Slave bei Auslesung sendet. Bei erstmaligen Betrieb der Ausleseeinheit oder nach Anschluß eines oder mehrerer neuer Endgeräte ist ein solcher Anwendungsreset durchzuführen.

Die Initialisierung auf Verbindungsebene durch ein SND\_NKE führt dazu, daß das primäre FCB (das FCB, das zur Primäradresse gehört) der Ausleseeinheit gelöscht wird. Es besitzt allerdings keine praktische Bedeutung, da die Informationen, die die Ausleseeinheit liefert, in einem Telegramm Platz finden.

### 3.3. Die sekundären Informationen

Die sekundären Informationen der Ausleseeinheit werden mit dem gesamten festen Datenblock des jeweiligen Slaves abgespeichert. Die Ausleseeinheit kann hiermit eine Kommunikation zwischen den Slaves und dem übergeordneten Master simulieren. Welche Slaves ausgelesen, konfiguriert oder initialisiert werden sollen wird durch eine Selektion über Sekundäradressen durch den übergeordneten Master festgelegt. Die Ausleseeinheit legt eine nach Sekundäradressen sortierte Liste der selektierten Slaves an. Schickt nun der Master ein Telegramm mit A-Feld = FDh, so wird das Telegramm auf alle in der Liste aufgeführten Endgeräte angewandt.

#### 3.3.1. Auslesen

Um die Informationen zu den Slaves auszulesen wird vom übergeordneten Master ein REQ\_UD2 Telegramm an die Adresse 253 gesendet. Pro selektierten Slave wird ein Telegramm geschickt, das variable Datenstruktur und Modus 1 besitzt. Folgen noch Slaves in der Selektionsliste oder ist die Menge an Daten eines Slaves zu groß für ein Telegramm, so ist das letzte Zeichen der Anwenderdaten das DIF = 1Fh. Es sagt aus, daß mehr Daten nach einem erneuten REQ\_UD2 folgen.

Die Ausleseeinheit unterstützt das FCB. Es wird solange das gleiche Telegramm übertragen, bis das FCB im REQ\_UD2 Telegramm getoggelt wird.

Unkonfiguriert schickt die Ausleseeinheit als variable Datenblöcke die realen Informationen der Endgeräte ohne Angabe von Daten. Anhand dieser kann der übergeordnete Master entscheiden, wie die Slaveinformationen zu konfigurieren sind.

Beispiel 7: Auslesen des unkonfigurierten Konzentrators, 2 Slaves angeschlossen*1. Schritt: Selektion aller Slaves per Wildcards*

68 0B 0B 68 | 53 FD 52 | FF FF FF FF | FF FF | FF | FF | 9A | 16

*2. Schritt: REQ\_UD2 an die Adresse 253*

10 | 5B | FD | 58 | 16

*Antwort des Konzentrators mit folgenden Daten:*

68 12 12 68 Kopf des Telegramms  
 08 FD 72 RSP\_UD, A-Feld 253, variable Datenstruktur Modus 1  
 80 65 39 15 Identifikationsnr. : 15396580  
 24 40 Hersteller : 4024h (PAD)  
 01 07 Version = 1, Medium = 7 (Wasser)  
 01 00 Zugriffszahl = 1, Status = 00h  
 00 00 Signatur = 0000h  
 00 DIF = keine Daten  
 13 VIF = Volumen in l  
 1F DIF : mehr Daten folgen, bei erneutem REQ\_UD2  
 39 16 Checksumme, Stop

*3. Schritt: Erneutes REQ\_UD2 des Masters, FCB getoggelt*

10 | 7B | FD | 78 | 16

*Antwort des Konzentrators:*

68 11 11 68 Kopf des Telegramms  
 08 FD 72 RSP\_UD, A-Feld 253, variable Datenstruktur Modus 1  
 13 58 34 17 Identifikationsnr. : 17345813  
 22 04 Hersteller : 0422h (ABC)  
 01 06 Version = 1, Medium = 6 (Warmwasser)  
 01 00 Zugriffszahl = 1, Status = 00h  
 00 00 Signatur = 0000h  
 00 DIF = keine Daten  
 B6 VIF = Temperatur in °C  
 16 Checksumme, Stop

(Es wird kein DIF = 1Fh gesetzt, da nur zwei Slaves angeschlossen sind.)



### 3.3.2. Fehlermeldungen

Antwortet ein Slave nach einem REQ\_UD2 der Ausleseeinheit nicht, so wird dies von der Ausleseeinheit ebenso festgehalten, wie der Zeitpunkt ab dem die Kommunikation nach diesem Fehlverhalten wieder erfolgreich verläuft. Die Mitteilung des Anfangs und des Endes der gestörten Kommunikation wird jeweils in Form eines Datenrecord mitgeteilt, wenn der übergeordnete Master die Ausleseeinheit durch ein REQ\_UD2 ausliest.

Die Ausleseeinheit speichert den ersten und den letzten aufgetretenen Fehler zu jedem Slave. Diese Informationen können gelöscht werden (siehe Kapitel 3.3.3.4. *Löschen von Daten*).

Die Ausgabe der Fehlerinformationen erfolgt mit Hilfe folgender Anwenderdaten:

DIB : DIF = 34h (Funktionsfeld: Fehler, Datenfeld: 32 Bit Integer)  
VIB : VIF = FEh (jedes VIF)  
+VIFE = 95h (keine Daten erhalten)  
+VIFE = 6Ah (Datum/Uhrzeit vom Beginn des ersten...) oder  
= 6Bh (Datum/Uhrzeit vom Ende des ersten...) oder  
= 6Eh (Datum/Uhrzeit vom Beginn des letzten...) oder  
= 6Fh (Datum/Uhrzeit vom Ende des letzten...)  
Datum : 4 Byte codiert nach Typ F der Norm IEC 870-5-4

Die ... sind Platzhalter für die Information des gesamten DRH. Für sie ist dementsprechend einzusetzen "...Fehlers zu jedem VIF-Code".

Der VIFE-Code 95h bzw. 15h stammt aus der Gruppe der recordspezifischen Fehlercodes, ist in diesem Falle aber für den gesamten Slave gültig, was aus der Verwendung des VIF-Codes 7Eh (jedes VIF) hervorgeht.

Beispiel 8: Antwort der Ausleseeinheit auf ein REQ\_UD2 des Masters. Es ist ein Slave selektiert, der in der Zeit von 15:00 Uhr bis 17:15 Uhr am 01.06.95 nicht geantwortet hat. Es werden nur die variablen Datenblöcke dargestellt. Alle Werte links hexadezimal.

*1. Datenrecord: Eine reale Information*

00            DIF     : Datenfeld = 0000b (keine Daten)  
2B            VIF     : Leistung in W

*2. Datenrecord: Anfang des ersten und einzigen Fehlers*

34            DIF     : Datenfeld: 32 Bit Integer, Funktionsfeld: Fehler  
FE            VIF     : jedes VIF  
95            VIFE    : keine Daten erhältlich  
6A            VIFE    : Beginn des ersten Fehlers zu jedem VIF  
00 0F E1 B6 Datum : 01.06.1995 15:00 Uhr Winterzeit

*3. Datenrecord: Ende des ersten Fehlers*

34            DIF     : siehe oben  
FE            VIF     : jedes VIF  
95            VIFE    : keine Daten erhältlich  
6B            VIFE    : Ende des ersten Fehlers zu jedem VIF  
0F 11 E1 B6 Datum : 01.06.1995 17:15 Uhr Winterzeit

### 3.3.3. Konfiguration

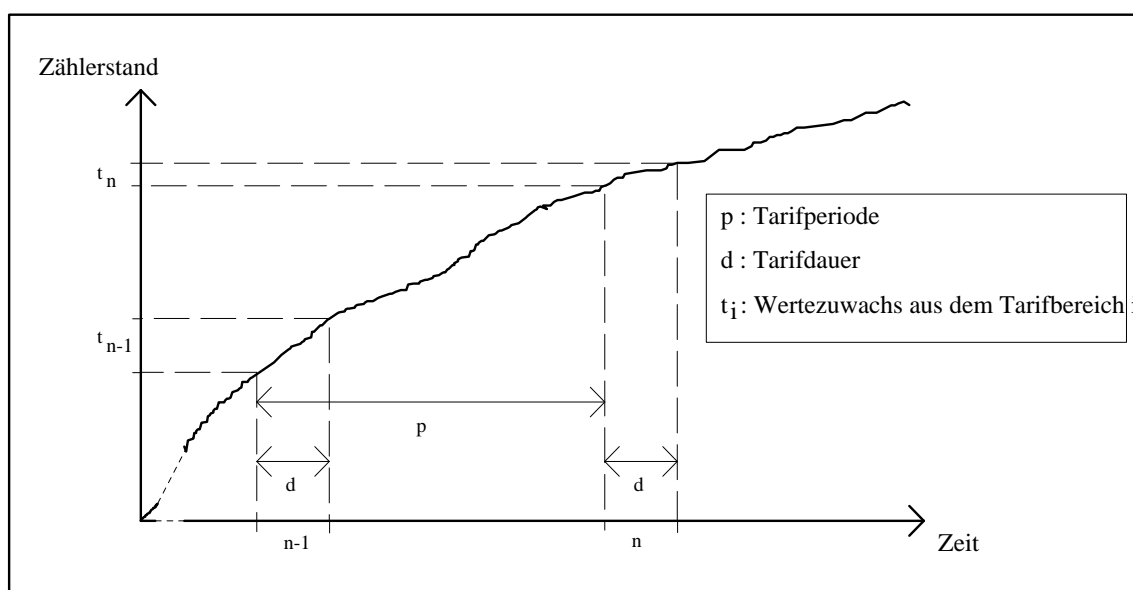
Die Konfiguration erfolgt immer mit Objekten wie im Kapitel 2.8.4.2 *Konfiguration von Slaves* beschrieben. Die Ausleseeinheit unterstützt die Aktionen "Schreibe(/Ersetze) Daten", "Lösche Daten", "Friere Daten ein", "Erzeuge Eintrag", "Lösche Eintrag", "Lösche von Ausgabeliste" und "Füge zur Ausgabeliste hinzu". Wie die Aktionen einzusetzen sind ist den folgenden Kapiteln zu entnehmen. Grundsätzlich gilt, daß die Datenrecords, die zur Programmierung eingesetzt werden, in einem oder mehreren SND\_UD Telegramm gesendet werden können. Um gleichzeitig mehrere Informationen konfigurieren zu können, unterstützt die Ausleseeinheit den VIF-Code 7Eh ("jedes VIF").

In den folgenden beiden Unterkapiteln wird erläutert, wie die Ausleseeinheit zu programmieren ist, um aus bestehenden Informationen neue (virtuelle Extrem- und Tarifwerte) zu bilden. Das dritte Unterkapitel beschäftigt sich mit der Speicherung von Daten.

### 3.3.3.1. Bildung von Tarifinformationen

#### Beschreibung einer Tarifinformation

Eine Tarifinformation gibt den Verbrauch eines Meßzählers, in periodisch wiederkehrenden Tarifbereichen wieder. Die zeitliche Ausdehnung des Bereiches wird als Tarafdauer, die Periode als Tarifperiode bezeichnet. Abb. 11 verdeutlicht den Meßvorgang. Es wird der Wertezuwachs  $t_i$  innerhalb der Bereiche  $i$  berechnet. Die augenblickliche Tarifinformation ist dann die Summe der  $t_i$ . Die Aktualisierung erfolgt jeweils am Ende eines Tarifbereiches.



**Abb. 11:** Bildung einer Tarifinformation

Eine Tarifinformation ist durch eine Tarifnummer gekennzeichnet und von folgenden Parametern abhängig, die sich durch den übergeordneten Master einstellen lassen:

- Tarafdauer
- Tarifperiode
- Beginn des Tarifes

Beispiel: Der in der Nacht von 22.00 bis 6.00 verbrauchte Strom wird nach einem besonderen Tarif, dem sogenannten Nachttarif, bezahlt. Eine solche Tarifinformation, besitzt als Tarifperiode 1 Tag und als Tarafdauer 8 Stunden. Der Beginn des Tarifes ist um 22.00 Uhr.

### Vorraussetzungen

Eine Tarifinformation läßt sich nur zu bestehenden augenblicklichen aufsummierenden Slaveinformationen einrichten. Daraus ergeben sich als Vorraussetzungen für die Bezugsinformation:

- Tarifnummer = 0
- Speichernummer = 0
- Funktionsfeld = 00b
- VIF-Code ist in der Tabelle im Anhang durch das Zeichen <sup>2</sup> gekennzeichnet

Tabelle 8 zeigt die zur Verfügung stehenden sekundären VIF-Codes zur Bildung eines Tarifes, die von der Ausleseeinheit unterstützt werden. Ferner werden die Aktionen "Erzeuge Eintrag", "Schreibe(/Ersetze) Daten" verwendet.

VIF-Code (binär)	Beschreibung
E011 0000	Startzeitpunkt des Tarifes (Datum/Uhrzeit)
E011 0001	Tarifdauer in Minuten
E011 0010	Tarifdauer in Stunden
E011 0011	Tarifdauer in Tagen
E011 0100	Tarifperiode in Sekunden
E011 0101	Tarifperiode in Minuten
E011 0110	Tarifperiode in Stunden
E011 0111	Tarifperiode in Tagen
E011 1000	Tarifperiode in Monaten
E011 1001	Tarifperiode in Jahren
E011 1010	Dimensionslos/kein VIF

**Tabelle 8:** VIF-Codes zur Einrichtung eines Tarifes

### Programmierung

Der erste Schritt zur Programmierung ist die Selektion eines oder mehrerer Slaves über deren Sekundäradresse.

Dann wird ein sogenanntes Tarifmuster zu jedem der selektierten Slaves erstellt, welches aus den oben angegebenen Parametern besteht. Das Muster bezieht sich zunächst einmal auf keine bestimmte Information. Der Ausleseeinheit wird als erstes mitgeteilt, daß ein neuer Eintrag im Speicher mit der Tarifnummer t erzeugt werden soll.

Dazu werden folgende Anwenderdaten verwendet:

DIB	: Datenfeld	= 0000b (keine Daten)
	Tarifnr.	= t (t > 0)
	Speichernr.	= 0
	Unitnr.	: beliebig
	Funktionsfeld	: beliebig
VIB	: VIF	= BAh (kein VIF)
	+ Aktion	= 08h ("Erzeuge Eintrag")

Die mitgeteilte Tarifnummer darf in keiner der bereits existierenden Dateneinträge vorkommen. Es findet keine Zuordnung zu irgendeiner bestehenden Information statt. Demzufolge wird das VIF "kein VIF" benutzt und Teile des DRH (Unitnummer, Funktionsfeld) nicht beachtet. Existiert eine Information mit derselben Tarifnummer zu einem bestimmten Slave, so wird der Datenrecord für diesen ignoriert, anderenfalls richtet die Ausleseinheit einen neuen Dateneintrag mit dem Wert der Tarifnummer ein.

Der nächste Schritt betrifft die, in der Reihenfolge beliebige, Parametrisierung des Tarifes. Der Bezug zwischen den verschiedenen Parametern eines Tarifes wird über die gemeinsame Tarifnummer hergestellt. Zur Übergabe des Startzeitpunktes werden folgende Anwenderdaten gesendet:

DIB	: Datenfeld	= 0100b (32 Bit Integer)
	Speichernr.	= 0
	Tarifnr.	= t
	Unitnr.	: beliebig
	Funktionsfeld	: beliebig
VIB	: VIF	= FDh B0h (Startzeitpunkt des Tarifes)
	+ Aktion	= 00h ("Schreibe(/Ersetze) Daten")
Datum	: Datum/Uhrzeit	: 4 Byte codiert nach Typ F der Norm IEC 870-5-4

Da die Aktion "Schreibe(/Ersetze) Daten", wie bereits beschrieben, den Grundzustand bei der Sendung von Daten seitens des Masters darstellt, kann auf die Übertragung des entsprechenden Bytes (00h) auch verzichtet werden.

Es ist darauf zu achten, daß nur Startzeitpunkte die in der Zukunft liegen, benutzt werden können.

Die Tarifdauer wird mit folgenden Anwenderdaten übermittelt:

DIB	: Datenfeld	: alle Integer und BCD Typen außer der variablen Länge
	Speichernr.	= 0
	Tarifnr.	= t
	Unitnr.	: beliebig
	Funktionsfeld	: beliebig
VIB	: VIF	= FDh B1h (Tarifdauer in Minuten) oder = FDh B2h (Tarifdauer in Stunden) oder = FDh B3h (Tarifdauer in Tagen)
	+ Aktion	= 00h ("Schreibe(/Ersetze) Daten")
Datum	: ergibt zusammen mit dem VIF die entsprechende Tarifdauer	

Das Datum muß einen Wert größer null und kleiner gleich 65535 besitzen, um von der Ausleseeinheit akzeptiert zu werden.

Der dritte Parameter, der einzustellen ist, betrifft die Tarifperiode. Hierzu wird der folgende Datenrecord gesendet:

DIB	: Datenfeld	: alle Integer und BCD Typen außer der variablen Länge
	Speichernr.	= 0
	Tarifnr.	= t
	Unitnr.	: beliebig
	Funktionsfeld	: beliebig
VIB	: VIF	= FDh B4h (Tarifperiode in Sekunden) oder = FDh B5h (Tarifperiode in Minuten) oder = FDh B6h (Tarifperiode in Stunden) oder = FDh B7h (Tarifperiode in Tagen) oder = FDh B8h (Tarifperiode in Monate) oder = FDh B9h (Tarifperiode in Jahre)
	+ Aktion	= 00h ("Schreibe(/Ersetze) Daten")
Datum	: ergibt zusammen mit dem VIF die entsprechende Tarifperiode	

Für das Datum gelten gleiche Bedingungen, wie bei der Zuteilung der Tarifdauer.

Die Parameter werden anhand der Tarifnummer dem entsprechendem Muster zugeordnet. Sie können nur gesetzt, aber nicht überschrieben werden. Bei Änderung müssen sie zunächst gelöscht und dann erneut geschrieben werden (siehe auch Kapitel 3.3.3.4. *Löschen von Daten*).

Die Ausleseeinheit verbindet mit jeder Tarifnummer bei vollständiger Parametrisierung Tarafdauer, Tarifperiode und Beginn des Tarifes. Der Master kann nun einer oder mehreren Slaveinformationen diese Nummer zuordnen. Die Ausleseeinheit erzeugt dann eine Tarifinformation mit den Parameter des Tarifmusters auf Basis der bestehenden Slaveinformation. Die Anwenderdaten hierzu haben folgendes Aussehen:

DIB	: Datenfeld	: alle BCD und Integer Typen außer variable Länge
	Unitnr.	: entsprechend der Bezugsinformation
	Speichernr.	= 0 (entsprechend der Bezugsinformation)
	Funktionsfeld	= 00b (entsprechend der Bezugsinformation)
	Tarifnr.	= t ( $t > 0$ )
VIB	: phys. Einheit	: entsprechend der Bezugsinformation
Datum	: enthält den Wert der Tarifnummer	im Word-Format

Die Zuordnung eines Musters zu einer Information ist nur dann erfolgreich, wenn alle Parameter geschrieben sind und wenn der Zeitpunkt der Zuordnung vor dem Beginn des Tarifes liegt.

#### Kurzfassung der Programmierung

1. Selektion des oder der Slaves
2. Erstellung eines Tarifmusters zu einem Slave
  - 2.1. neuen Eintrag mit Tarifnummer erzeugen
  - 2.2. Parametrisierung (Reihenfolge beliebig)
    - a) Tarafdauer
    - b) Tarifperiode
    - c) Beginn des Tarifes
3. Zuordnung des Tarifmusters zu einer oder mehrerer bestehender Informationen

#### Datenausgabe

- Es werden sämtliche Parameter des Tarifmusters ausgegeben. Die Anwenderdaten entsprechen denen der Programmierung, ohne Aktionen.
- Die Tarifinformationen selbst werden gesendet. Diese lassen sich durch die Tarifnummer den Parametern zuordnen. Sind noch keine Daten vorhanden wird dies durch das Datenfeld 0000b (keine Daten) ausgedrückt. Anderenfalls benutzt die Ausleseeinheit das Datenformat der Bezugsinformation.

### Besonderheiten

Wie bereits erwähnt ist es möglich die Tarifperiode gleich der Tarifdauer zu wählen. Die zugehörige augenblickliche Tarifinformation beinhaltet dann den Verbrauch des Endgerätes seit Beginn ihrer Bildung. Dieser Wert wird in Zeitintervallen (entsprechend der Tarifperiode) aktualisiert. Eine solche Information wird als Verbrauchstarifinformation bezeichnet.

Beispiel: Es wird ein Tarif mit der Nummer 1 gebildet dessen Dauer und Periode auf 24 Stunden festgelegt wird. Tarifbeginn sei der 01.01.1995 um 00:00 Uhr. Wird eine zugeordnete Tarifinformation am 03.01.1995 um 00:01 Uhr ausgelesen, so entspricht der Wert dem Verbrauch der letzten zwei Tage. Das Ergebnis bleibt 24 Stunden erhalten, um dann durch den Verbrauch des zurückliegenden Tages ergänzt zu werden.

### Beispiel 9: Erstellung und Zuordnung eines Tarifmusters

#### *1. Schritt: Selektion aller angeschlossenen Slaves*

68 0B 0B 68 | 53 FD 52 | FF FF FF FF | FF FF | FF | FF | 9A | 16

#### *2. Schritt: Erzeugung eines neuen Tarifmustereintrages mit der Tarifnummer 2*

68 23 23 68    Kopf des Telegramms  
 53 FD 51      SND\_UD, Adresse 253, Master sendet Daten  
 80            DIF     : keine Daten  
 20            DIFE    : Tarifnummer 2  
 FD BA        VIF     : kein VIF  
 08            Aktion  : Erzeuge neuen Eintrag

#### *3. Schritt: Parametrisierung (gleiches Telegramm)*

##### *a) Tarifdauer*

82            DIF     : Datenfeld            : 16 Bit Integer  
 20            DIFE    : Tarifnr.                 = 2  
 FD B1        VIF     : Tarifdauer in Minuten  
 00            Aktion  : Schreibe(/Ersetze) Daten  
 2C 01        Datum  = 300 (Minuten)



*b) Tarifperiode*

82	DIF	: Datenfeld	: 16 Bit Integer
20	DIFE	: Tarifnr.	= 2
FD B7	VIF	: Periodendauer in Tagen	
00	Aktion	: Schreibe(/Ersetze) Daten	
01 00	Datum	= 1 (Tag)	

*c) Anfangszeitpunkt des Tarifes*

84	DIF	: Datenfeld	: 32 Bit Integer
20	DIFE	: Tarifnr.	= 2
FD B0	VIF	: Startzeitpunkt des Tarifes	
00	Aktion	: Schreibe(/Ersetze) Daten	
00 00 E1 B4	Datum	: 01.11.95 00:00 Uhr Winterzeit	

*4.Schritt: Zuordnung des Tarifmusters zu einer Slaveinformation*

80	DIF	: Datenfeld	: keine Daten
20	DIFE	: Tarifnr.	= 2
86	VIF	: Energie in kWh	
08	Aktion	: Erzeuge Eintrag	
E8	Checksumme		
16	Ende des Telegramms		

### 3.3.3.2. Bildung von Extremwerten

#### Beschreibung eines Extremwertes

Zur Bildung von Extremwerten wird eine Slaveinformation in äquidistanten Zeitabständen ausgelesen. Es besteht eine Abhängigkeit zu folgenden Parametern:

- Art des VIF der ausgelesenen Information (aufsummiert, augenblicklich, durchschnittlich)
- Zeitpunkt der ersten Auslesung (Phasenangabe)
- Zeit zwischen den Auslesungen

Die letzten beiden Parameter können durch den Master bestimmt werden. Der erste ist durch die Information, zu der ein Extremwert eingerichtet werden soll, fest vorgegeben. Die Tabelle der VIF-Codes zur variablen Datenstruktur im Anhang teilt die phys. Einheiten in drei Gruppen:

1. aufsummierte Werte
2. durchschnittliche Werte
3. augenblickliche Werte.

Um einen Extremwert zu einem durchschnittlichen oder augenblicklichen Wert zu bilden wird dieser einfach ausgelesen, mit dem bisherigen verglichen und ersetzt, wenn dieser echt größer bzw. kleiner als der bestehende ist. Die Genauigkeit des Extremwertes hängt von der Zeit zwischen den Auslesungen ab. Je kleiner diese Zeit gewählt wird, desto genauer ist der Extremwert.

Eine Slaveinformation, die aufsummierte Werte liefert, zeichnet sich dadurch aus, daß der gesendete Wert mit der Zeit nie abnimmt. Spricht man von einem Extremwert im Zusammenhang mit solchen Informationen, so meint man den größten oder kleinsten Wertezuwachs, der zwischen den einzelnen Auslesungen festgestellt worden ist. Ein Maximum ist also der größte Wertezuwachs, ein Minimum der kleinste Wertezuwachs seit Beginn der Bildung des Extremwertes.

#### Voraussetzungen

Ein Extremwert kann nur auf Grundlage einer bestehenden Information eines Slaves gebildet werden. Voraussetzungen für die Bezugsinformation sind:

- Speichernummer = 0
- Funktionsfeld = 00b
- VIF-Code ist in der Tabelle im Anhang durch die Zeichen <sup>2</sup> oder « gekennzeichnet

Es ist nicht möglich Extremwerte zu realen Tarifinformationen einzurichten.

Zur Bildung eines solchen Extremwertes werden die Aktionen "Erzeuge Eintrag" und "Schreibe(/Ersetze) Daten" sowie mehrere modifizierte VIFE genutzt, deren Bedeutung der folgenden Tabelle zu entnehmen ist:

VIFE-Code (binär)	Beschreibung
E011 1001	Das Datum des Datenrecords beinhaltet die Startzeit der Information, die durch den DRH beschrieben wird
E110 0100	Das Datum des Datenrecords beinhaltet eine Zeitdauer in Sekunden, die sich auf die Information im DRH bezieht
E110 0101	wie oben, Zeitdauer in Minuten
E110 0110	wie oben, Zeitdauer in Stunden
E110 0111	wie oben, Zeitdauer in Tagen

**Tabelle 9:** VIFE-Codes zur Bildung eines Extremwertes

#### Programmierung

Ein Extremwert setzt sich aus drei Informationen zusammen:

1. dem augenblicklichen Wert
2. dem Parameter zum Beginn der Extremwertbildung
3. dem Parameter zum zeitlichen Abstand der Auslesungen

Um kenntlich zu machen, daß diese Informationen zusammengehören, besitzen alle den gleichen DRH, der bei den Parametern durch eine entsprechende VIFE (siehe oben) ergänzt wird.

Der erste Schritt der Programmierung erfolgt durch die Selektion des oder der Slaves, zu denen Maximum oder Minimuminformationen gebildet werden sollen.

Danach sendet der Master ein (oder mehrere) SND\_UD Telegramm(e) mit variabler Datenstruktur an die Adresse 253. Zunächst wird ein neuer Eintrag in der Ausleseinheit erzeugt. Dazu muß der Master den Bezug zu einer bestehenden Information herstellen, die Art des einzurichtenden Extremwertes (Maximum oder Minimum) angeben und die Aktion "Erzeuge Eintrag" am VIB anhängen. Die entsprechenden Anwenderdaten hierzu bestehen nur aus einem DRH.

Dieser enthält folgende Informationen:

DIB	: Datenfeld	= 0000b ( keine Daten)
	Funktionsfeld	= 01b für Maximum / 10b für Minimum
	Speichernr.	= 0 (immer)
	Unitnr.	: entsprechend der Bezugsinformation
	Tarifnr.	: entsprechend der Bezugsinformation
VIB	: phys. Einheit	: entsprechend der Bezugsinformation
	+ Aktion	= 08h ("Erzeuge Eintrag")

Aus der phys. Einheit der Unit- und Tarifnummer stellt die Ausleseeinheit den Bezug zu einer existierenden Information her. Anhand des Funktionsfeldes und der Aktion erkennt die Ausleseeinheit, daß ein Extremwert zu dieser einzurichten ist. Zu jeder phys. Einheit mit gleicher Tarif- und Unitnummer läßt sich ein Maximum und ein Minimum einrichten. Hiernach erfolgt die Parametrisierung des Extremwertes. Dazu wird die Aktion "Schreibe(/Ersetze) Daten" genutzt. Die Ausleseeinheit muß am DRH der Datenrecords erkennen können, zu welche Slaveinformation der Parameter gehört. Zur Einstellung der Phasenangabe, also dem Datum und der Uhrzeit des Beginns der Messungen, werden folgende Anwenderdaten verwendet.

DIB	: Datenfeld	= 0100b (32 Bit Integer)
	Funktionsfeld	= 01b für Maximum / 10b für Minimum
	Speichernr.	= 0 (immer)
	Unitnr.	: entsprechend der Bezugsinformation
	Tarifnr.	: entsprechend der Bezugsinformation
VIB	: phys. Einheit	: siehe oben
	+ VIFE	= B9h (Startzeitpunkt der Information im DRH)
	+ Aktion	= 00h ("Schreibe(/Ersetze) Daten")
Datum	: Datum/Uhrzeit	: 4 Byte codiert nach Typ F der Norm IEC 870-5-4

Es ist darauf zu achten, daß der Startzeitpunkt des Extremwertes in der Zukunft liegt. Ansonsten wird der Datenrecord ignoriert.

Der zweite Parameter, die Angabe zum zeitlichen Abstand zwischen zwei Auslesungen, wird durch folgende Anwenderdaten dargestellt:

DIB	: Datenfeld	: alle Integer und BCD Typen außer der variablen Länge
	Funktionsfeld	= 01b für Maximum / 10b für Minimum
	Speichernr.	= 0 (immer)
	Unitnr.	: entsprechend der Bezugsinformation
	Tarifnr.	: entsprechend der Bezugsinformation
VIB	: phys. Einheit	: siehe oben
	+ VIFE= E0h	(Zeitdauer in Sekunden) oder
		= E1h (Zeitdauer in Minuten) oder
		= E2h (Zeitdauer in Stunden) oder
		= E3h (Zeitdauer in Tagen)
	+ Aktion	= 00h ("Schreibe(/Ersetze) Daten")
Datum	: ergibt zusammen mit der VIFE	die entsprechende Zeitdauer

Das Datum muß einen Wert größer null und kleiner gleich 65535 enthalten, sonst wird dieser Record nicht von der Ausleseinheit akzeptiert.

Um Parameter zu ändern, ist es nicht möglich, diese zu überschreiben. Sie müssen zunächst einmal gelöscht werden. Die Beschreibung befindet sich in dem Kapitel *Löschen von Parametern und Informationen*.

### Kurzfassung der Programmierung

1. Selektion des oder der Slaves
2. Erzeugung eines neuen Datenrecords mit der Aktion "Erzeuge Eintrag" und dem Funktionsfeld
3. Parametrisierung
  - a) Phasenangabe (Datum/Uhrzeit der ersten Messung)
  - b) Meßdauer (Abstand zwischen den Auslesungen)

Die Schritte 3a) und 3b) können in ihrer Reihenfolge vertauscht werden.

### Ausgabe der Informationen zum Extremwert

Der fehlerfreie Empfang auf Anwendungsebene kann kontrolliert werden, indem der zugehörige Slave ausgelesen wird. Es werden folgende Angaben in Form von jeweils einem Datenrecord zu dem Extremwert ausgegeben:

- Der eigentliche Extremwert. Ist noch keiner gebildet worden (kein Fehler), so wird dies durch ein Datenfeld 0000b (keine Daten) ausgedrückt. Anderenfalls benutzt die Ausleseeinheit das Datenfeld der Bezugsinformation zur Ausgabe des Wertes.
- Die Zeitdauer zwischen den Auslesungen des Slaves. Das Telegramm entspricht dem der Programmierung ohne Aktion. Das Wert wird immer als 24 Bit Integer codiert.
- Der Zeitpunkt der ersten Auslesung des Slaves zur Bildung eines Extremwertes. Das Telegramm ist gleich dem der Programmierung ohne die Aktion.
- Der Zeitpunkt der Bildung oder der Auslesung des aktuellen Extremwertes. Dieser Datenrecord wird nur dann gesendet, wenn ein Extremwert existiert. Es wird der modifizierte VIFE-Code E1101011b verwendet. Diese VIFE sagt aus, daß der übertragene Wert des Datenrecords dem Aufnahmezeitpunkt der Information entspricht, die durch den DRH beschrieben wird. Der Aufbau der Daten sieht wie folgt aus:

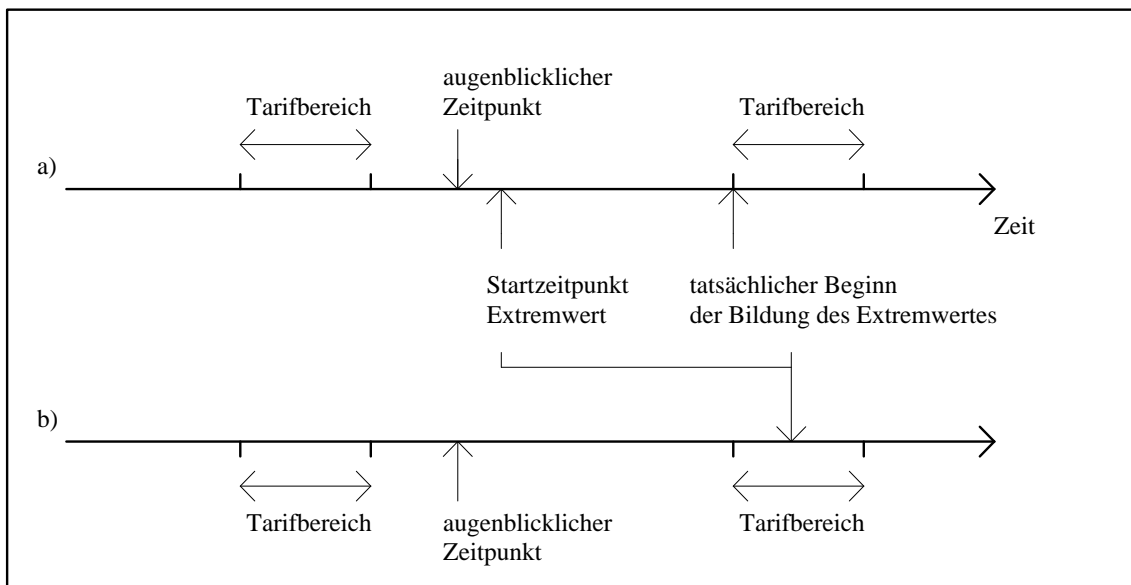
DIB	: Datenfeld	= 0100b (32 Bit Integer)
	Funktionsfeld	= 01b für Maximum / 10b für Minimum
	Speichernr.	= 0 (immer)
	Unitnr.	: entsprechend der Information
	Tarifnr.	: entsprechend der Information
VIB	: phys. Einheit	: entsprechend der Information
	+ VIFE=	6Bh
Datum	: Datum/Uhrzeit	: 4 Byte codiert nach Typ F der Norm IEC 870-5-4

### Besonderheiten:

Der erste Extremwert zu einer aufsummierenden Bezugsinformation wird im Vergleich zu dem ersten Extremwert mit gleichen Parametern zu einer augenblicklichen oder durchschnittlichen Bezugsinformation um eine Auslesungszeitspanne später ermittelt.

Es lassen sich Extremwerte zu virtuellen Tarifinformationen einrichten. Diese zeichnen sich dadurch aus, daß ihre Bildung nur innerhalb des Tarifbereiches erfolgt. Die Programmierung erfolgt wie oben beschrieben. Die Phasenangabe wird allerdings vom Tarif übernommen, d.h. der Anfang eines Tarifbereiches ist gleichzeitig auch Beginn der Bildung oder der Auslesung der Extremwerte.

Beispiel: Es existiert eine Tarifinformation, die jeden Tag von 00:00 Uhr bis 06:00 Uhr den Verbrauch eines Slaves feststellt. In diesem Tarifbereich soll der größte Wertezuwachs pro Stunde festgestellt werden. Setzt der übergeordnete Master den Startzeitpunkt des Extremwertes auf 15:00 Uhr dieses Tages fest, so beginnt die Bildung des ersten Wertes um 00:00 Uhr des folgenden (vgl. Abb. 12a)). Liegt der Startzeitpunkt des Extremwertes innerhalb des Tarifbereiches, so entspricht dieser auch dem tatsächlichen Beginn der Extremwertbildung (vgl. Abb. 12b)).



**Abb. 12a), b):** Bildung eines Extremwertes zu einer virtuellen Tarifinformation

Eine weitere Besonderheit besteht darin, daß Extremwerte zu sogenannten Verbrauchstarifinformationen (Tarifdauer entspricht Tarifperiode, siehe oben) eingerichtet werden können. Ein Maximum bzw. Minimum zu einer solchen Information liefert das gleiche Ergebnis, wie zu der Bezugsinformation des Tarifes. Über den Umweg der Tarifbildung ist es also möglich, mehrere Extremwerte zu einer Information einzurichten.

Beispiel 10: Eine Information liefert bei Auslesung den augenblicklichen Wasserverbrauch mit der Einheit Volumen in l und der Unitnummer 1. Tarifnummer und Speichernummer besitzen den Wert 0. Der maximale Verbrauch pro Minute soll ab dem 01.04.1995 um 10.00 Uhr festgestellt werden.

*1. Schritt: Selektion des entsprechenden Slaves. Hier mit der Identifikationsnummer 12345678, Hersteller PAD, Version 1, Medium: Wasser*

68 0B 0B 68 | 53 FD 52 | 78 56 34 12 | 24 40 | 01 | 07 | 22 | 16

*2. Schritt: Erzeugung eines neuen Informationseintrages zu der Information mit der Unitnr. 1, der Tarif- und Speichernr. 0 und der phys. Einheit Volumen in l.*

68 16 16 68 Kopf des Telegramms

53 FD 51 C-, A-, CI- Feld : SND\_UD, Adresse 253, Master sendet Daten

90 DIF : keine Daten, Funktionsfeld: Maximum

40 DIFE : Unitnr. = 1, Speichernr. = 0, Tarifnr. = 0

93 VIF : Volumen in l

08 Aktion : Erzeuge Eintrag

*3. Schritt: Parametrisierung (gleiches Telegramm)*

*a) Übergabe des Anfangszeitpunktes*

94 DIF : Datenfeld: 32 Bit Integer, Funktionsfeld: Maximum

40 DIFE : Unitnr. = 1, Speichernr. = 0, Tarifnr. = 0

93 VIF : Volumen in l

B9 VIFE : Startzeitpunkt der Information im DRH

00 Aktion : Schreibe(/Ersetze) Daten

0A 0A E1 B4 Datum : 01.04.1995 10:10 Uhr

*b) Übergabe des Zeitabstandes zwischen den Auslesungen der Bezugsinformation*

99 DIF : Datenfeld: 2 Stellen BCD, Funktionsfeld: Maximum

40 DIFE : Unitnr. = 1, Speichernr. = 0, Tarifnr. = 0

93 VIF : Volumen in l

E0 VIFE : Dauer in Sekunden

00 Aktion : Schreibe(/Ersetze) Daten

60 Datum : 60 Sekunden

81 Checksumme

16 Stop : Ende des Telegramms



Beispiel 11: Gesendete Daten der Ausleseeinheit hinsichtlich des obigen Beispiels. Es werden nur die variablen Datenblöcke angegeben.

*1. Datenrecord: Parameter zur Meßdauer*

92            DIF     : Datenfeld: 16 Bit Integer, Funktionsfeld: Maximum  
 40            DIFE    : Unitnr. = 1, Speichernr. = 0, Tarifnr. = 0  
 93            VIF     : Volumen in l  
 60            VIFE    : Dauer in Sekunden  
 60 00        Datum   : 60 Sekunden

*2. Datenrecord: Parameter zum Anfangszeitpunkt*

94            DIF     : Datenfeld: 32 Bit Integer, Funktionsfeld: Maximum  
 40            DIFE    : Unitnr. = 1, Speichernr. = 0, Tarifnr. = 0  
 93            VIF     : Volumen in l  
 39            VIFE    : Startzeitpunkt der Information im DRH  
 0A 0A E1 B4 Datum   : 01.04.1995 10:10 Uhr

*3. Datenrecord: Der eigentliche Extremwert*

9B            DIF     : Datenfeld: 6 Stellen BCD, Funktionsfeld: Maximum  
 40            DIFE    : Unitnr. = 1, Speichernr. = 0, Tarifnr. = 0  
 13            VIF     : Volumen in l  
 14 00 00    Datum   : 14 (Liter)

*4. Datenrecord: Zeitpunkt der Bildung des Extremwertes*

94            DIF     : Datenfeld: 32 Bit Integer, Funktionsfeld: Maximum  
 40            DIFE    : Unitnr. = 1, Speichernr. = 0, Tarifnr. = 0  
 93            VIF     : Volumen in l  
 6B            VIFE    : Datum/Uhrzeit der im DRH beschriebenen Information  
 0B 0A E1 B4 Datum   : 01.04.1995 10:11 Uhr

### 3.3.3.3. Historisierung augenblicklicher Information

Die Ausleseeinheit unterstützt zwei verschiedene Arten der Speicherung von augenblicklichen Werten:

1. Einfrieren von Daten
2. Bildung historischer Profile

#### 3.3.3.3.1. Einfrieren von Daten

Die einzufrierenden Daten werden zusammen mit ihrem Speicherzeitpunkt in der Ausleseeinheit augenblicklich abgelegt, wenn die entsprechende Aufforderung vom übergeordneten Master erfolgt. Kennzeichen und Verknüpfung von Zeitpunkt und Information ist die gemeinsame Speichernummer, welche vom übergeordneten Master vergeben wird.

Die Slaves deren Daten einzufrieren sind werden über ihre Sekundäradresse selektiert. Der eigentliche Vorgang der Speicherung wird durch folgende Anwenderdaten ausgelöst:

DIB	: Datenfeld	: 0000b (keine Daten)
	Speichernr.	: s (s > 0)
	Unitnr.	: entsprechend der zu speichernden Information
	Tarifnr.	: entsprechend der zu speichernden Information
	Funktionsfeld	: entsprechend der zu speichernden Information
VIB	: phys. Einheit	: entsprechend der zu speichernden Information
	+ Aktion	= 0Bh ("Friere Daten ein")

Anhand des Funktionsfeldes, der Unit- und Tarifnummer und dem VIB erkennt die Ausleseeinheit, auf welche Information Bezug zu nehmen ist. Die Aktion "Friere Daten ein" veranlaßt den Konzentrador zur sofortigen Speicherung des Wertes dieser Information zusammen mit dem augenblicklichem Datum und der Uhrzeit unter der angegebenen Speichernummer. Es können nur Speichernummern verwendet werden, die bisher noch nicht benötigt wurden.

Die Ausgabe eines eingefrorenen Wertes beinhaltet zwei Datenrecords. Einen für den Wert selbst, den anderen für den Aufnahmezeitpunkt.

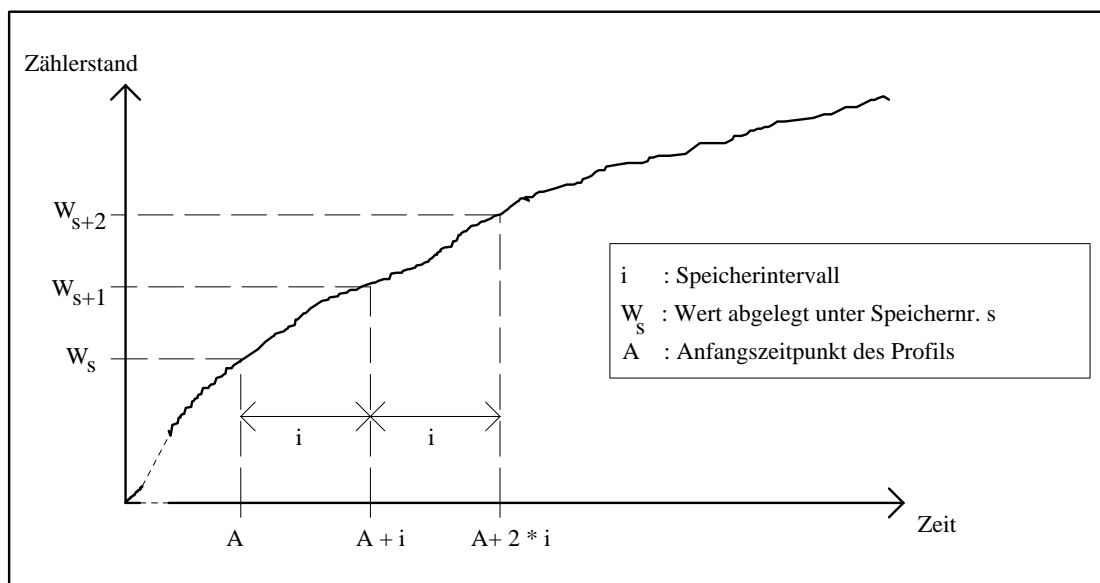
### 3.3.3.3.2. Bildung historischer Profile

#### Beschreibung eines historischen Profils

Ein historisches Profil wird gebildet, indem eine Information von der Ausleseeinheit ab einem Startzeitpunkt in äquidistanten Zeitabständen ausgelesen und der empfangene Wert abgespeichert wird (siehe Abb. 13). Um die historisierten Werte zeitlich unterscheiden zu können werden sie mit Hilfe von Speichernummern durchnummeriert. Hierbei gilt, daß der älteste Wert die kleinste Speichernummer erhält. Ist die maximale Größe eines Profils erreicht, und ein weiterer Wert wird ausgelesen, so wird die älteste Information gelöscht. Die Speichernummern werden um eine Position verschoben.

Folgende Parameter lassen sich durch den übergeordneten Master einstellen:

- erste Speichernummer
- Anzahl Speichernummern
- Speicherintervall (Zeitraum zwischen den Abspeicherungen)
- Zeitpunkt der ersten Speicherung



**Abb. 13:** Bildung eines historischen Profils

#### Vorraussetzungen

Ein historisches Profil läßt sich zu jeder bestehenden Slaveinformationen einrichten die vollständig parametrisiert ist, die Speichernummer 0 besitzt und einen primären oder sekundären VIF-Code besitzt , der in der Tabelle im Anhang durch die Zeichen « oder <sup>2</sup> gekennzeichnet ist.

Tabelle 10 zeigt die sekundären VIF-Codes zur Bildung eines historischen Profils, die von der Ausleseeinheit unterstützt werden. Weiterhin werden die Aktionen "Erzeuge Eintrag" und "Schreibe(/Ersetze) Daten" benutzt.

VIF-Code (binär)	Beschreibung
E010 0000	Erste Speichernummer für zyklische Speicherung
E010 0010	Anzahl Speichernummern
E010 0100	Speicherintervall in Sekunden
E010 0101	Speicherintervall in Minuten
E010 0110	Speicherintervall in Stunden
E010 0111	Speicherintervall in Tagen
E010 1000	Speicherintervall in Monaten
E010 1001	Speicherintervall in Jahren

**Tabelle 10:** Sekundäre VIF-Codes zur Bildung eines historischen Profils

### Programmierung

Der erste Schritt ist die Selektion der oder des Slaves dessen Informationen gespeichert werden sollen.

Analog zur Bildung einer Tarifinformation wird zunächst ein sogenanntes Profilmuster erstellt, welches aus der Angabe der Speichernummern, dem Speicherintervall und dem Zeitpunkt der ersten Speicherung besteht. Diese Muster ist für den gesamten Slave und nicht nur für eine bestimmte Information gültig.

Der Wunsch zur Einrichtung eines neuen Profilmusters wird durch folgenden Datenrecord ausgedrückt:

DIB : Datenfeld : alle BCD- und Integer-Typen, außer variable Länge  
Speichernr. = s ( $s > 0$ , erste Speichernummer des Profilmusters)  
Unitnr. : beliebig  
Tarifnr. : beliebig  
Funktionsfeld : beliebig  
VIB : VIF = FDh A2h (Anzahl der Speichernummern)  
+ Aktion = 08h ("Erzeuge Eintrag")  
Datum : enthält die Anzahl der Speichernummern (max. 65535)

Wie den Angaben zu entnehmen ist, erfolgt gleichzeitig eine Parametrisierung hinsichtlich der Anfangsspeichernummer und der Größe des Profils. Die Speichernummern werden nur

dann für das Profil reserviert, wenn sie nicht schon belegt sind. Im anderen Fall ignoriert die Ausleseeinheit den Datenrecord. Das Kennzeichen eines Profils ist die erste Speicher- nummer (hier s). Bei Einstellung der Parameter (Reihenfolge beliebig) wird auf diese Nummer Bezug genommen.

Der Aufbau der Anwenderdaten zur Parametrisierung des Speicherintervalls besitzt folgende Form:

DIB	: Datenfeld	: alle BCD- und Integer-Typen, außer variable Länge
	Speichernr.	= s
	Unitnr.	: beliebig
	Tarifnr.	: beliebig
	Funktionsfeld	: beliebig
VIB	: VIF	= FDh A4h (Speicherintervall in Sekunden) oder = FDh A5h (Speicherintervall in Minuten) oder = FDh A6h (Speicherintervall in Stunden) oder = FDh A7h (Speicherintervall in Tagen) oder = FDh A8h (Speicherintervall in Monaten) oder = FDh A9h (Speicherintervall in Jahren)
	+ Aktion	= 00h ("Schreibe(/Ersetze) Daten")
Datum	: ergibt zusammen mit dem VIF	das Speicherintervall

Die Ausleseeinheit erkennt an der Speichernummer, der Aktion und dem VIF, welchem Profil das Datum zuzuordnen ist und welche Bedeutung es besitzt. Es sind Werte größer 0 bis einschließlich 65535 erlaubt.

Zur Parametrisierung des Anfangszeitpunktes werden die nachstehenden Anwenderdaten benutzt:

DIB	: Datenfeld	= 0100b (32 Bit Integer)
	Speichernr.	= s
	Unitnr.	: beliebig
	Tarifnr.	: beliebig
	Funktionsfeld	: beliebig
VIB	: VIF	= EDh (Datum und Zeit)
	+ Aktion	= 00h ("Schreibe(/Ersetze) Daten")
Datum	: 4 Byte codiert nach Typ F	der Norm IEC 870-5-4

Wie immer gilt, daß der übergebene Zeitpunkt in der Zukunft liegen muß und das Parameter nicht überschrieben werden können, sondern zunächst gelöscht werden müssen, um sie zu ändern.

Hiermit ist die Erstellung des Profilmusters abgeschlossen. Es kann nun ein oder mehreren Slaveinformationen zugeordnet werden. Nur zu diesen wird dann ein historisches Profil mit den entsprechenden Parametern gebildet. Die Zuordnung erfordert folgende Anwenderdaten und ist nur dann erfolgreich, wenn der Zeitpunkt der ersten Speichernummer in der Zukunft liegt:

DIB	: Datenfeld	= 0000b (keine Daten)
	Speichernr.	= s (erste Speichernummer des zuzuordnenden Profils)
	Unitnr.	: entsprechend der Bezugsinformation
	Tarifnr.	: entsprechend der Bezugsinformation
	Funktionsfeld	: entsprechend der Bezugsinformation
VIB	: phys. Einheit	: entsprechend der Bezugsinformation
	+ Aktion	= 08h ("Erzeuge Eintrag")

### Kurzfassung der Programmierung

1. Selektion des oder der Slaves
2. Erstellung eines Profilmusters
  - 2.1. Erzeugung eines neuen Eintrages
  - 2.2. Parametrisierung
    - a) Speicherintervall
    - b) Beginn der Speicherung
3. Zuordnung des Profilmusters zu einem oder mehrerer Slaves

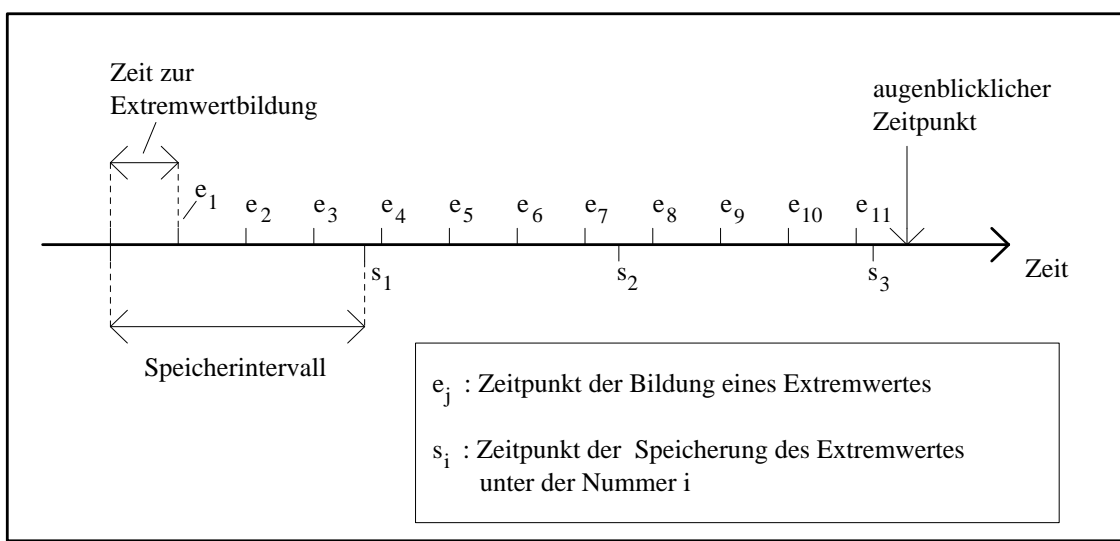
### Ausgabe des historischen Profils

Die Ausleseeinheit sendet bei Aufforderung zu jedem Profilmuster den Aufnahmezeitpunkt des ersten Speicherwertes, die Anzahl der Speichernummern und das Speicherintervall. Ist das Profilmuster bereits einer Information zugeordnet, so sendet die Ausleseeinheit die gespeicherten Werte (wenn vorhanden), die durch Speichernummern gekennzeichnet sind.

### Besonderheiten

Wird ein historisches Profil zu einer virtuellen Extremwertinformation angelegt, so wird nicht der augenblickliche Extremwert gespeichert, sondern der jeweilige Extremwert des zurückliegenden Speicherintervalls. Es ist darauf zu achten, daß das Speicherintervall größer oder gleich der Auslesungszeit bzgl. der Extremwertinformation ist. Zur Verdeutlichung dient Abb. 14. Zu den mit  $e_j$  ( $j = 1..11$ ) gekennzeichneten Zeiten werden

Maxima (Minima) gebildet. Es liegt eine aufsummierende Information zugrunde. Zum augenblicklichen Zeitpunkt sind 11 Maxima (Minima) gebildet worden. Das größte (kleinste) Wert aus dieser Menge befindet sich als augenblickliche Information im Speicher der Ausleseeinheit. Die Speichernummer 1 enthält den größten (kleinsten) Wert aus der Menge der Maxima (Minima), die zu den Zeitpunkten  $e_1, e_2, e_3$  gebildet wurden. Speichernummer 2 enthält den zu den Zeiten  $e_5, e_6, e_7$  und Speichernummer 3 den zu den Zeiten  $e_9, e_{10}, e_{11}$  gebildeten größten (kleinsten) Wert. Die Zeitpunkte  $e_4$  und  $e_8$  bleiben unberücksichtigt, weil die Bezugsinformation aufsummierend ist und deshalb die Bildung des Maximums (Minimums) schon innerhalb des vorhergehenden Speicherintervalls beginnt.



**Abb. 14:** Bildung eines historischen Profils zu einer virtuellen Extremwertinformation

Bei der Zuordnung eines Profilmusters zu einer Extremwertinformation ist darauf zu achten, daß die Ausleseeinheit mit der Suche nach dem ersten Speicherwert schon ein Speicherintervall vor dem eigentlichen Speicherzeitpunkt beginnen wird. Das bedeutet, daß der Zeitpunkt der Zuordnung noch vor dieser Zeit liegen muß.

Beispiel 12: Erstellung eines Profilmusters zu allen angeschlossenen Slaves und Zuordnung zu einer bestimmten Information. Alle Daten links hexadezimal.

*1. Schritt: Selektion aller angeschlossenen Slaves*

68 0B 0B 68 | 53 FD 52 | FF FF FF FF | FF FF | FF | FF | 9A | 16

*2. Schritt: Erzeugung eines Profilmusters mit den Speichernummern von 2 bis 10*

68 1B 1B 68 Kopf des Telegramms  
 53 FD 51 SND\_UD, Adresse 253, Master sendet Daten  
 89 DIF : Datenfeld : 2 Stellen BCD  
 01 DIFE : Speichernr. = 2  
 FD A2 VIF : Anzahl Speichernummern  
 08 Aktion : Erzeuge Eintrag  
 09 Datum : 9 Speichernummern werden benötigt

*2.Schritt: Parametrisierung (gleiches Telegramm)*

*a) Speicherintervall: Es soll alle 10 Minuten ein Wert gespeichert werden*

89 DIF : Datenfeld : 2 Stellen BCD  
 10 DIFE : Speichernr. = 2  
 FD A5 VIF : Speicherintervall in Minuten  
 00 Aktion : Schreibe(/Ersetze) Daten  
 10 Datum = 10 (Minuten)

*b) Zeitpunkt der ersten Speicherung: 01.04.1995 10:11 Uhr*

84 DIF : Datenfeld = 0100b (32 Bit Integer)  
 01 DIFE : Speichernr. = 2  
 ED VIF : Datum/Zeit  
 00 Aktion : Schreibe(/Ersetze) Daten  
 0B 0A E1 B4 Datum : 01.04.1995 10:11 Uhr Winterzeit

*3. Schritt: Zuordnung des Profilmusters zu einer Slaveinformation (gleiches Telegramm)*

80 DIF : Datenfeld : keine Daten  
 01 DIFE : Speichernr. = 2  
 9B VIF : Masse in kg  
 08 Aktion : Erzeuge Eintrag



### 3.3.3.4. Löschen von Daten

Alle Informationen, die durch Konfiguration der Ausleseinheit und durch Fehler entstanden sind, lassen sich durch den übergeordneten Master löschen. Zur Erschaffung von neuen Datenrecords werden Anwenderdaten benutzt, die eine der drei Aktionen "Erzeuge Eintrag", "Friere Daten ein" oder "Schreibe(/Ersetze) Daten" beinhalten. Zur Löschung der jeweiligen Information werden die Anwenderdaten beibehalten, die Aktionen jedoch gemäß Tabelle 11 ersetzt. Als Datenfeld ist 0000b (keine Daten) zu verwenden.

Erzeugen von Daten		Löschen von Daten	
VIFE-Code	Aktion	VIFE-Code	Aktion
E000 1000b	Erzeuge Eintrag	E000 1001b	Lösche Eintrag
E000 0000b	Schreibe(/Ersetze) Daten	E000 0111b	Lösche Daten
E000 1011b	Friere Daten ein	E000 1001b	Lösche Eintrag

**Tabelle 11:** Gegenüberstellung der Aktionen zur Löschung und Erzeugung von Daten

Die Löschung einer augenblicklichen virtuellen Information, bedeutet auch die Löschung der historischen Werte zu dieser. Wird ein Tarif- oder Profilmuster bzw. ein Parameter hierzu gelöscht, so zieht dies auch die Löschung der augenblicklichen Informationen die auf Basis des Musters entstanden sind nach sich.

Zur Löschung der Informationen, die durch fehlerhafte Kommunikation entstanden sind, werden folgende Anwenderdaten verwendet:

30            DIF     : Funktionsfeld = 11b (Fehler)  
 FE            VIF     : Jedes VIF  
 09            Aktion : "Lösche Eintrag"

Beispiel 13: Löschen eines Profilmusters mit der Anfangsspeichernummer zwei

68 08 08 68    Kopf des Telegramms  
53 FD 51      SND\_UD, Adresse 253, Master sendet Daten  
80            DIF     : Datenfeld            : keine Daten  
01            DIFE    : Speichernr.                = 2  
FD A2        VIF     : Anzahl Speichernummern  
09            Aktion : Lösche Eintrag  
CA            Checksumme  
16            Stop

Beispiel 14: Löschen des Parameters Tariffdauer zum Tarif Nummer zwei

68 08 08 68    Kopf des Telegramms  
53 FD 51      SND\_UD, Adresse 253, Master sendet Daten  
80            DIF     : Datenfeld            : keine Daten  
20            DIFE    : Tarifnr.                    = 2  
FD B7        VIF     : Periodendauer in Tagen  
07            Aktion : Lösche Daten  
FC            Checksumme  
16            Stop

### 3.3.3.5. Konfiguration der Datenausgabe

Die Entscheidung, welche Informationen die Ausleseeinheit zu einem Slave sendet erfolgt nach einer Ausgabeliste. Darin werden alle Informationen notiert, die bei einem RSP\_UD ausgegeben werden. Im Grundzustand sendet die Ausleseeinheit alle verfügbaren Informationen zu den selektierten Slaves. Jede neu gebildete Information wird automatisch in der Liste vermerkt. Mit den im Kapitel 2 beschriebenen Aktionen "Lösche von der Ausgabeliste" und "Füge zur Ausgabeliste hinzu" lässt sich diese beeinflussen. Hierzu werden die Anwenderdaten der Information mit der entsprechenden Aktion versehen. Das übergebene Datum besitzt keine Bedeutung, daher wird die Verwendung des Datenfeldes 0000b (keine Daten) empfohlen.

Beispiel 15: Löschen einer Information von der Ausgabeliste der Ausleseeinheit

00	DIF	: Datenfeld = 0000b (keine Daten)
AB	VIF	: Leistung in kW
0D	Aktion	: Lösche von Ausgabeliste

### 3.3.4. Initialisierung

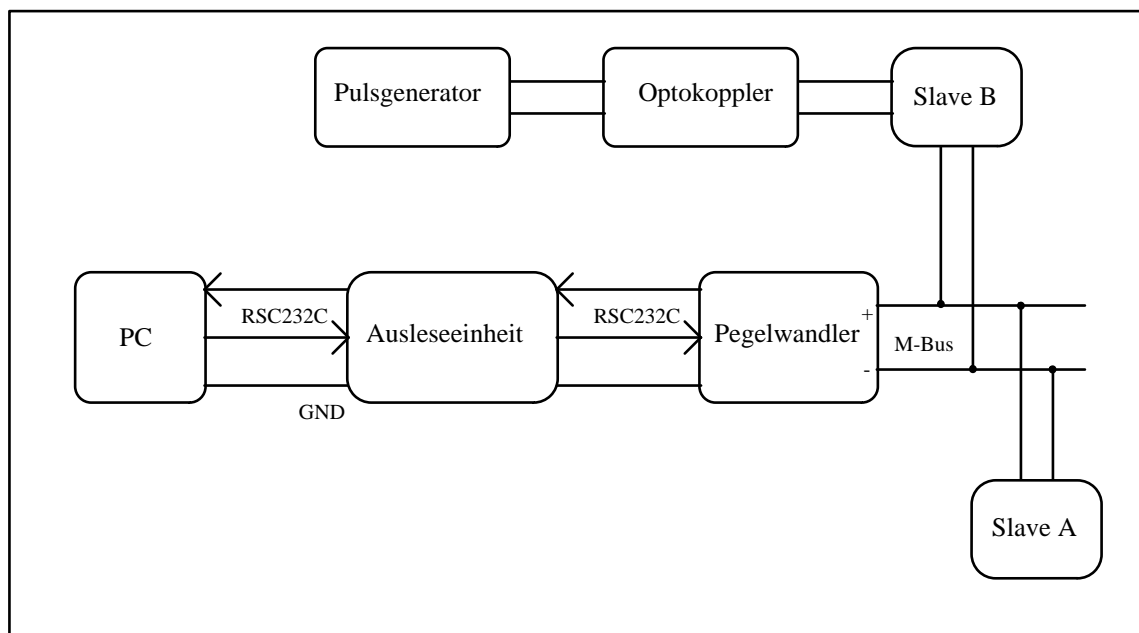
Die Initialisierung auf Verbindungsebene durch ein SND\_NKE an die Adresse 253 hat die Deselektion der angeschlossenen Endgeräte zur Folge. Ein Reset auf Anwendungsebene an die Adresse 253 wird bisher noch nicht unterstützt.

## 4. Installierung und Betrieb der Ausleseeinheit

Ziel dieses Kapitels ist die Beschreibung eines Dauertestes, bei dem die Ausleseeinheit zwei Tage lang auf ihre Funktionsfähigkeit überprüft wurde. Die in dieser Zeit gesendeten und empfangenen Daten ab dem Zeitpunkt des Anschlusses werden angegeben und erläutert.

### 4.1. Aufbau und Durchführung

Den Aufbau gibt Abb. 15 wieder. An die eine Schnittstelle der Ausleseeinheit wird ein übergeordneter Master in Form eines PC's angeschlossen. Als Mastersoftware wird das modifizierte Programm "M-Bus Application Vers. 1.17" benutzt. An die zweite Schnittstelle wird ein Pegelwandler angeschlossen, über den zwei Endgeräte betrieben werden. Bei dem Slave A handelt es sich um einen Wasserzähler der Firma Techem mit der Identifikationsnummer 38570130, der seine Informationen mit festem Datenprotokoll im Modus 1 und bis zu 2400 Baud sendet. Der Zählerstand bleibt während der Versuchsdauer konstant.



**Abb. 15:** Versuchsaufbau

Bei dem Slave B handelt es sich um einen Padpuls der Version 2, der Impulse zählt und seine Daten im Format 48 Bit Integer mit bis zu 9600 Baud, variablen Protokoll und Modus 1 überträgt. Die Angaben Identifikationsnummer, Hersteller, Medium, Primäradresse, VIF-Code, Impulswertigkeit und Zählerstand lassen sich vom Nutzer programmieren.

In diesem Fall werden folgende Werte eingestellt:

Identifikationsnr. = 12345678  
 Hersteller = UNI  
 Primäradresse = 1  
 VIF-Code = 10h (Volumen in ml)  
 Impulswertigkeit = 1  
 Zählerstand = 0 (ml)

An dieses Endgerät wird ein Pulsgenerator angeschlossen, der alle 500 ms einen Impuls liefert und somit den Zählerstand um 2 ml pro Sekunde erhöht.

Die Ausleseeinheit wird so programmiert, daß zu diesem Slave ein Profilmuster erstellt und der einzigen bestehenden Information zugeordnet wird. Das Muster ist durch folgende Parameter gekennzeichnet:

- erste Speichernr. : 1
- Anzahl Speichernr. : 25
- Speicherintervall : 2 Stunden
- Beginn des Profils : 03.03.1995 um 12:00 Uhr

Der Pulsgenerator wird kurz vor Beginn des historischen Profils aktiviert.

Die im folgenden dargestellten und erläuterten Daten stammen aus einer Protokolldatei, die von der Mastersoftware angelegt wird. Darin sind alle gesendeten und empfangenen Daten bezüglich dieses Testes enthalten. Die Daten des Masters werden fett, die der Ausleseeinheit normal, Zeitpunkte und Beschreibungen kursiv dargestellt.

- Initialisierung der Ausleseeinheit

<i>03.03.95 11:33:12</i>	<i>Abfrage von primären Daten</i>
<b>10 7B 00 7B 16</b>	<b>REQ_UD2 an die Adresse 0</b>
68 2A 2A 68 08 00 72	RSP_UD, Adresse 0, variables Protokoll Modus 1
00 00 00 00	Identifikationsnr.= 00000000
C9 55	Hersteller: UNI
01 0E 01	Version 1, Medium: Bus/System, erster Zugriff
00 00 00	Status = 00h, Signatur = 0000h
04 6D 12 0C E3 B3	Systemzeit: 03.03.1995 12:12
04 7C 04 65 74 79 42	DIF: 32 Bit Integer, VIF im String: "Byte"
D0 48 00 00	Datum: 18640 (Byte)
01 7C 06 73 65 76 61 6C 53	8 Bit Integer, VIF im String: "Slaves"

00	Datum: 0 (Slaves)
EE 16	Checksumme, Stopzeichen
03.03.95 11:34:00	<i>Setzen der Primäradresse der Ausleseinheit auf 1</i>
<b>68 07 07 68 53 00 51</b>	<b>SND_UD an die Adresse 0, Master sendet Daten</b>
<b>01 FA</b>	<b>DIF: 8 Bit Integer, VIF: Busadresse,</b>
<b>00 01</b>	<b>Aktion: Schreibe(/Ersetze) Daten, Datum: 1</b>
<b>A0 16</b>	<b>Checksumme, Stopzeichen</b>
E5	Einzelzeichen
03.03.95 11:34:33	<i>Setzen der Identifikationsnr. auf 1</i>
<b>68 0A 0A 68 53 01 51</b>	<b>SND_UD an die Adresse 1, Master sendet Daten</b>
<b>0C F9</b>	<b>DIF: 8 Stellen BCD, VIF: Identifikation</b>
<b>00 01 00 00 00</b>	<b>Aktion: Schreibe(/Ersetze) Daten, Datum: 00000001h</b>
<b>AB 16</b>	<b>Checksumme, Einzelzeichen</b>
E5	Einzelzeichen
03.03.95 11:50:01	<i>Setzen der Systemzeit</i>
<b>68 0A 0A 68 73 01 51</b>	<b>SND_UD an die Adresse 1, Master sendet Daten</b>
<b>04 ED</b>	<b>DIF: 32 Bit Integer, VIF: Zeitpunkt</b>
<b>00</b>	<b>Aktion: Schreibe(/Ersetze) Daten</b>
<b>32 0B E3 B3</b>	<b>Datum: 03.03.1995 11:50 Uhr</b>
<b>89 16</b>	<b>Checksumme, Stopzeichen</b>
E5	Einzelzeichen
03.03.95 11:50:11	<i>Anwendungsreset</i>
<b>68 03 03 68 53 01 50</b>	<b>SND_UD an die Adresse 1, Anwendungsreset</b>
<b>A4 16</b>	<b>Checksumme, Stopzeichen</b>
E5	Einzelzeichen
03.03.95 11:50:25	<i>Abfrage von primären Daten</i>
<b>10 5B 01 5C 16</b>	<b>REQ_UD2 an die Adresse 1</b>
68 2A 2A 68 08 01 72	RSP_UD, Adresse 1, variables Protokoll, Modus 1
01 00 00 00	Identifikationsnr. = 00000001
C9 55	Hersteller: UNI
01 0E 02	Version 1, Medium: Bus/System, zweiter Zugriff
00 00 00	Status: 00h, Signatur: 0000h
04 6D 32 0B E3 B3	Systemzeit: 03.03.1995 11:50
04 7C 04 65 74 79 42	DIF: 32 Bit Integer, VIF im String: "Byte"
10 48 00 00	Datum: 18448 (Byte)

01 7C 06 73 65 76 61 6C 53     DIF: 8 Bit Integer, VIF im String: "Slaves"  
 02                                     Datum: 2 (Slaves)  
 52 16                                  Checksumme und Stopzeichen

- Konfiguration der sekundären Informationen

*03.03.95 11:50:32*                     *Abfrage von sekundären Daten*  
**10 7B FD 78 16**                         **REQ\_UD2 an die Adresse 253**  
 68 12 12 68 08 FD 72                 RSP\_UD, Adresse 253, variables Protokoll, Modus 1  
 78 56 34 12                             Identifikationsnr. = 1234567 (Padpuls)  
 C9 55                                     Hersteller: UNI  
 01 07 01                                 Version 1, Medium: Wasser, erster Zugriff  
 00 00 00                                 Status: 00h, Signatur: 0000h  
 00 10                                     DIF: keine Daten, VIF: Volumen in ml  
 1F                                         DIF: mehr Daten folgen nach einem weiteren REQ\_UD2  
 E1 16                                     Checksumme, Stopzeichen

*03.03.95 11:50:33*                     *Erneute Abfrage von sekundären Daten*  
**10 5B FD 58 16**                         **REQ\_UD2 an die Adresse 253**  
 68 13 13 68 08 FD 72                 RSP\_UD, Adresse 253, variables Protokoll, Modus 1  
 30 01 57 38                             Identifikationsnr. = 38570130 (Techem Wasserzähler)  
 00 00                                     Hersteller: geht aus fester Datenstruktur nicht hervor  
 00                                         Version: geht aus fester Datenstruktur nicht hervor  
 07 01                                     Medium: Wasser, erster Zugriff  
 00 00 00                                 Status: 00h, Signatur: 0000h  
 00 13                                     DIF: keine Daten, VIF: Volumen in l  
 40 13                                     DIF: keine Daten, Speichernr. 1, VIF: Volumen in l  
 A5 16                                     Checksumme, Stopzeichen

*03.03.95 11:50:40*                     *Selektion des Padpuls*  
**68 0B 0B 68 73 FD 52**                 **SND\_UD an die Adresse 253, Selektion**  
**78 56 34 12**                             **Identifikationsnr. = 12345678**  
**FF FF FF FF**                             **Hersteller, Version, Medium sind kein Bestandteil**  
**der Selektion**  
**D2 16**                                     **Checksumme, Stopzeichen**  
 E5                                         Einzelzeichen

*03.03.95 11:51:12*                     *Erzeugen eines Speicherblocks mit den Speichernr. 1 - 25*  
**68 08 08 68 53 FD 51**                 **SND\_UD an die Adresse 253, Master sendet Daten**

<b>41</b>	<b>DIF: 8 Bit Integer, Speichernr. 1</b>
<b>FD A2 08</b>	<b>VIF: Anzahl Speichernr., Aktion: Erzeuge Eintrag</b>
<b>19</b>	<b>Datum: 25 (Speichernummern)</b>
<b>A2 16</b>	<b>Checksumme, Stopzeichen</b>
<b>E5</b>	Einzelzeichen
<i>03.03.95 11:51:44</i>	<i>Parametrisierung des Speicherintervalls</i>
<b>68 08 08 68 73 FD 51</b>	<b>SND_UD an die Adresse 253, Master sendet Daten</b>
<b>41</b>	<b>DIF: 8 Bit Integer, Speichernr. 1</b>
<b>FD A6</b>	<b>VIF: Speicherintervall in Stunden</b>
<b>00 02</b>	<b>Aktion: Schreibe(/Ersetze), Datum: 2 (Stunden)</b>
<b>A7 16</b>	<b>Checksumme, Stopzeichen</b>
<b>E5</b>	Einzelzeichen
<i>03.03.95 11:52:17</i>	<i>Parametrisierung des Startzeitpunktes</i>
<b>68 0A 0A 68 53 FD 51</b>	<b>SND_UD an die Adresse 253, Master sendet Daten</b>
<b>44 ED</b>	<b>DIF: 32 Bit Integer, VIF: Zeitpunkt</b>
<b>00</b>	<b>Aktion Schreibe(/Ersetze)</b>
<b>00 0C E3 B3</b>	<b>Datum: 03.03.1995 12:00</b>
<b>74 16</b>	<b>Checksumme, Stopzeichen</b>
<b>E5</b>	Einzelzeichen
<i>03.03.95 11:52:32</i>	<i>Zuordnung des Profilmusters zu der bestehenden Information</i>
<b>68 06 06 68 73 FD 51</b>	<b>SND_UD an die Adresse 253, Master sendet Daten</b>
<b>40 90</b>	<b>DIF: keine Daten, Speichernr. 1, VIF: Volumen in ml</b>
<b>08</b>	<b>Aktion: Erzeuge Eintrag</b>
<b>99 16</b>	<b>Checksumme, Stopzeichen</b>
<b>E5</b>	Einzelchen
• Auslesung	
<i>03.03.95 13:06:06</i>	<i>Abfrage von sekundären Daten</i>
<b>10 7B FD 78 16</b>	<b>REQ_UD2 an die Adresse 253</b>
<b>68 2B 2B 68 08 FD 72</b>	RSP_UD, Adresse 253
<b>78 56 34 12 C9 55 01 07 02</b>	fester Datenblock
<b>00 00 00</b>	fester Datenblock
<b>00 10</b>	DIF: keine Daten, VIF: Volumen in ml
<b>44 6D</b>	DIF: Speichernr. 1, VIF: Zeitpunkt



00 0C E3 B3 Datum: 03.03.1995 12:00  
 43 FD 26 DIF: Speichernr. 1, VIF: Speicherintervall in Stunden  
 02 00 00 Datum: 2 (Stunden)  
 43 FD 22 DIF: Speichernr. 1, VIF: Anzahl Speichernummern  
 19 00 00 Datum: 25 (Speichernummern)  
 46 10 DIF: Speichernr. 1, VIF: Volumen in ml  
 73 03 00 00 00 00 Datum: 883 (ml)  
 C5 16 Checksumme, Stopzeichen

04.03.95 13:08:10

*Abfrage von sekundären Daten*

**10 7B FD 78 16**

**REQ\_UD2 an die Adresse 253**

68 97 97 68 08 FD 72

RSP\_UD, Adresse 253

78 56 34 12 C9 55 01 07 03

fester Datenblock

00 00 00

fester Datenblock

00 10

DIF: keine Daten, VIF: Volumen in ml

44 6D 00 0C E3 B3

Speichernr. 1, Zeit: 03.03.1995 12:00

43 FD 26 02 00 00

Speichernr. 1, Speicherintervall: 2 Stunden

43 FD 22 19 00 00

Speichernr. 1, Anzahl Speichernr.: 25

46 10 73 03 00 00 00 00

Speichernr. 1: 883 ml

86 01 10 7F 3B 00 00 00 00

Speichernr. 2: 15231 ml

C6 01 10 93 73 00 00 00 00

Speichernr. 3: 29587 ml

86 02 10 9F AB 00 00 00 00

Speichernr. 4: 43935 ml

C6 02 10 AE E3 00 00 00 00

Speichernr. 5: 58286 ml

86 03 10 BA 1B 01 00 00 00

Speichernr. 6: 72634 ml

C6 03 10 C2 53 01 00 00 00

Speichernr. 7: 86978 ml

86 04 10 C9 8B 01 00 00 00

Speichernr. 8: 101321 ml

C6 04 10 D0 C3 01 00 00 00

Speichernr. 9: 115664 ml

86 05 10 D6 FB 01 00 00 00

Speichernr. 10: 130006 ml

C6 05 10 DB 33 02 00 00 00

Speichernr. 11: 144347 ml

86 06 10 E0 6B 02 00 00 00

Speichernr. 12: 158688 ml

C6 06 10 ED A3 02 00 00 00

Speichernr. 13: 173037 ml

A9 16

Checksumme, Stopzeichen

05.03.95 12:01:41

*Abfrage von sekundären Informationen*

**10 7B FD 78 16**

**REQ\_UD2 an die Adresse 253**

68 F2 F2 68 08 FD 72

RSP\_UD, Adresse 253

78 56 34 12 C9 55 01 07 04

fester Datenblock

00 00 00

fester Datenblock

00 10

DIF: keine Daten, VIF: Volumen in ml

44 6D 00 0C E3 B3	Speichernr. 1, Zeit: 03.03.1995 12:00
43 FD 26 02 00 00	Speichernr. 1, Speicherintervall: 2 Stunden
43 FD 22 19 00 00	Speichernr. 1, Anzahl Speichernr.: 25
46 10 73 03 00 00 00 00	86 01 10 7F 3B 00 00 00 00 C6 01 10 93 73 00 00 00 00
86 02 10 9F AB 00 00 00 00	C6 02 10 AE E3 00 00 00 00 86 03 10 BA 1B 01 00 00 00
C6 03 10 C2 53 01 00 00 00	86 04 10 C9 8B 01 00 00 00 C6 04 10 D0 C3 01 00 00 00
86 05 10 D6 FB 01 00 00 00	C6 05 10 DB 33 02 00 00 00 86 06 10 E0 6B 02 00 00 00
C6 06 10 ED A3 02 00 00 00	Werte der Speichernummern 1 bis 13 (siehe oben)
86 07 10 FE DB 02 00 00 00	Speichernr. 14: 187390 ml
C6 07 10 11 14 03 00 00 00	Speichernr. 15: 201745 ml
86 08 10 1F 4C 03 00 00 00	Speichernr. 16: 216095 ml
C6 08 10 2E 84 03 00 00 00	Speichernr. 17: 230446 ml
86 09 10 3A BC 03 00 00 00	Speichernr. 18: 244794 ml
C6 09 10 43 F4 03 00 00 00	Speichernr. 19: 259139 ml
86 0A 10 4C 2C 04 00 00 00	Speichernr. 20: 273484 ml
C6 0A 10 56 64 04 00 00 00	Speichernr. 21: 287830 ml
86 0B 10 5F 9C 04 00 00 00	Speichernr. 22: 302175 ml
C6 0B 10 68 D4 04 00 00 00	Speichernr. 23: 316520 ml
1F	DIF: mehr Daten folgen nach einem erneuten REQ_UD2
11 16	Checksumme, Stopzeichen

05.03.95 12:01:49

**10 5B FD 58 16**

68 21 21 68 08 FD 72  
 78 56 34 12 C9 55 01 07 05  
 00 00 00  
 86 0C 10 74 0C 05 00 00 00  
 C6 0C 10 81 44 05 00 00 00  
 89 16

*erneute Abrage von sekundären Informationen*

**REQ\_UD2 an die Adresse 253**

RSP\_UD, Adresse 253  
 fester Datenblock  
 fester Datenblock  
 Speichernr. 24:330868 ml  
 Speichernr. 25:345217 ml  
 Checksumme, Stopzeichen

## 4.2. Ergebnisse und Auswertung

Die erhaltenen Werte zu den einzelnen Speichernummern sind in der Tabelle 12 zusammengefaßt. Es werden die Differenzen zweier aufeinanderfolgender Nummern gebildet und festgehalten.

Speichernr. n	Zählerstand <sub>n</sub> in ml	Zählerstand <sub>n</sub> - Zählerstand <sub>n-1</sub> in ml
1	883	-----
2	15231	14348
3	29587	14356
4	43935	14348
5	58286	14351
6	72634	14348
7	86978	14344
8	101321	14343
9	115664	14343
10	130006	14342
11	144347	14341
12	158688	14341
13	173037	14349
14	187390	14353
15	201745	14355
16	216095	14350
17	230446	14351
18	244794	14348
19	259139	14345
20	273484	14345
21	287830	14346
22	302175	14345
23	316520	14345
24	330868	14348
25	345217	14349

**Tabelle 12:** Zählerstände der Speichernummern

Als Mittelwert der Zählerstandsdifferenzen ergibt sich: 14347 ml

Die Standardabweichung beträgt: 4 ml

Der Pulsgenerator sendet mit einer Frequenz von 2 Hz Impulse. In zwei Stunden sind demzufolge 14400 Pulse bzw. ml zu erwarten. Die Genauigkeit dieses Gerätes beträgt 5 % des eingestellten Wertes. Die Wiederholgenauigkeit ist um den Faktor vier besser. Damit ergeben sich folgende Werte:

Genauigkeit:  $\pm 720$  Pulse

Wiederholgenauigkeit:  $\pm 180$  Pulse

Die von der Ausleseinheit ermittelten Zählerstände liegen damit deutlich innerhalb der Grenzen der Ungenauigkeit des Pulsgenerators. Somit kann davon ausgegangen werden, daß die Ausleseinheit auch über einen längeren Zeitraum zuverlässig arbeitet.



## 5. Messungen der Repeatereigenschaften

In diesem Kapitel werden die Eigenschaften zweier vom Konzept grundsätzlich verschiedener Repeater verglichen. Um die noch zu spezifizierenden Anforderungen zu erfüllen werden für den einen analoge für den anderen digitale Bauelemente verwendet. Beide sind sogenannte Vollrepeater, an die bis zu 250 Standardlasten angeschlossen werden können.

Der Analog-Repeater soll Baudraten im Bereich von 300 bis 9600 Baud beherrschen, muß dazu aber über einen Widerstand auf die jeweilige Baudrate eingestellt werden. Für die Messungen stand ein auf 2400 Baud eingestellter Repeater zur Verfügung.

Der Digital-Repeater soll bei Baudraten von 300 bis 9600 Baud, 19200 Baud und 38400 Baud arbeiten können. Er verfügt über ein Bitrecovery, d.h. er ist in der Lage Bitverzerrungen wieder auszugleichen.

Beide sind für den Einsatz als Pegelwandler vorgesehen, d.h. sie wandeln Signale der seriellen Schnittstelle in M-Bus Pegel und umgekehrt.

### 5.1. Repeaterspezifikationen

Aufgrund der physikalischen Schicht, ergeben sich nachstehende Anforderungen an einen Repeater auf dem M-Bus:

- **Aufrufrichtung**

Im Ruhezustand (Mark) soll die Ausgangsspannung des Repeaters im Bereich von 35V bis 40V liegen. Die durch das Senden eines Space durch ein oder zwei Slaves hervorgerufenen Spannungseinbrüche, dürfen nicht mehr als 1,5 V betragen. Die Space-Spannung muß mindestens 12 V unter der Mark-Spannung liegen. Gleichzeitig darf aber, um die Fernspeisung der Slaves zu gewährleisten, die Busspannung nicht unter 23 V sinken.

- **Antwortrichtung**

Der Repeater soll sich auf den konstanten Ruhestrom der Slaves einstellen. Erhöht sich der Bustrom um mehr als 8 mA so soll ein Space-, bei einer Erhöhung um weniger als 5 mA ein Markzustand erkannt werden.

- **Anstiegsgeschwindigkeiten**

Die Anstiegsgeschwindigkeiten der Flanken in Aufrufrichtung sollen auch bei Variation der Kabellänge und des Ruhestroms immer zwischen 0,1 und 30 V/ $\mu$ s liegen. Im Bereich der ersten und letzten 10% der Flanke darf die Anstiegsgeschwindigkeit unter 0,1 V/ $\mu$ s liegen.

- **Bitverzerrung**

Die Verzerrung, d.h. die Verlängerung bzw. die Verkürzung einer Bitzeit durch den Repeater, darf in Antwort- und Aufrufzeit höchstens 25 % betragen. Beispiel: Bei 2400 Baud beträgt die errechnete Bitzeit für Space und Mark 416  $\mu$ s. Eine Abweichung um 104  $\mu$ s durch den Repeater ist noch erlaubt.

- **Fernspeisung der Slaves**

Bei Kurzschluß eines Endgerätes, soll der Bus funktionsfähig bleiben. Hierzu wird ein 430  $\Omega$  Widerstand längs in eine der Buszuleitungen jedes Slaves gelegt. Die obigen Bedingungen bezüglich der Busspannungen müssen für diesen Fall und bis 400 mA Ruhestrom erfüllt werden. Bei Empfang von 10 Bit Space der niedrigsten Baudrate (300 Bd => 33,3 ms Space) darf die Ruhestromreferenz des Repeaters nur um 2mA ansteigen. Die Regelung des Repeaters an höhere Ströme darf also nicht zu schnell erfolgen.

- **Überstrom- und Kurzschlußschutz**

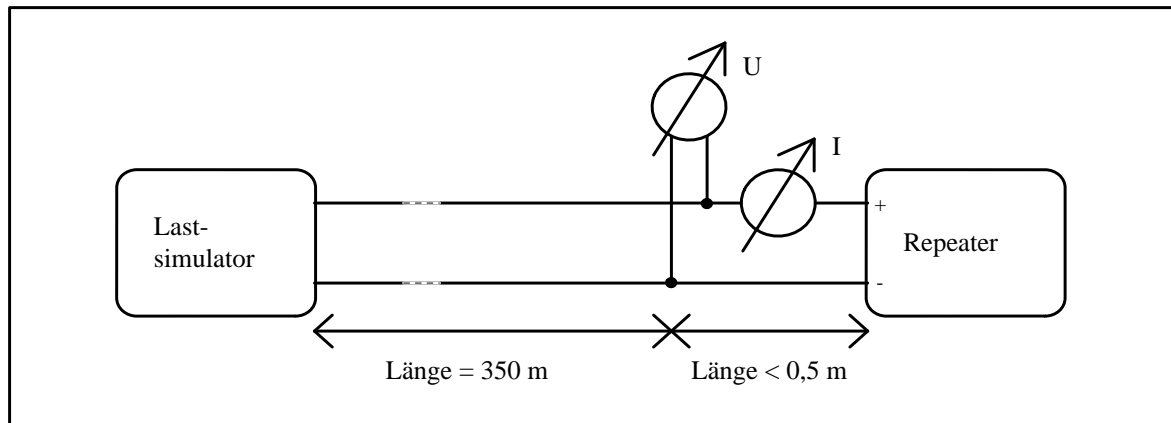
Ab einem Ruhestrom von 80% des vom Repeater lieferbaren Stroms soll eine Anzeige (z.B. eine LED) einen Überstromzustand signalisieren. Nach einem beliebig langem Kurzschluß soll der Repeater selbstständig in den normalen Betrieb zurückkehren.

## 5.2. Versuchsaufbau und Durchführung

Im folgenden werden die Messungen beschrieben, die in den beiden anschließenden Kapiteln mit beiden Repeatern durchgeführt wurden. Die Messungen orientieren sich an den oben beschriebenen Repeaterspezifikationen.

- **Busspannung am Repeater in Abhängigkeit vom Ruhestrom**

Mit Hilfe eines Lastsimulators wird der aus dem Bus entnommene Ruhestrom variiert, und die vom Repeater gelieferte Spannung gemessen. Hiermit kann überprüft werden, ob der Repeater im Ruhezustand mindestens die geforderten 35 V liefert und über eine Überstromanzeige und Warnstromanzeige verfügt. Den Meßaufbau gibt Abb. 16 wieder.



**Abb. 16:** Aufbau zur Messung der Busspannung in Abhängigkeit des Ruhestroms

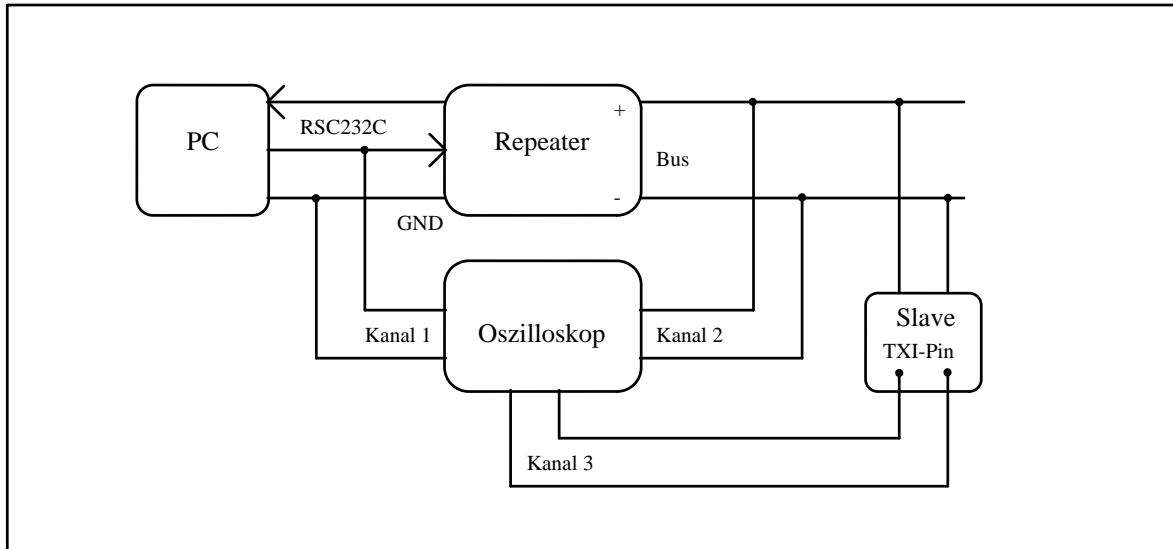
### • Übertragungsverhalten

Das Übertragungsverhalten in Auf- und Antwortrichtung wird unter Variation der Kabelänge und der angeschlossenen Slaves gemessen. Es lassen sich Anstiegsgeschwindigkeiten der Flanken und die Verzerrungen der Signale bestimmen.

Jedes Endgerät verfügt über eine Schnittstelle zum Bus, einen sogenannten Transceiver, der für Sendung und Empfang der M-Bus Signale verantwortlich ist. Dieser besitzt Schwellen, bei denen er ein empfangenes Space bzw. Mark erkennt. Liegt die Spannung des Busses um 7,9 V unterhalb des Mark-Zustandes, so erkennt er sicher ein Space. Der Übergang von Space zu Mark wird sicher oberhalb der Mark-Spannung abzüglich 5,7 V erkannt. Anhand dieser Detektionsschwellen, werden die Berechnungen über Mark und Space-Bitzeiten durchgeführt. Die M-Bus Pegel werden im Transceiver umgewandelt und liegen am TXI-Pin des Endgerätes in der Form 0 V (Mark) und 3,2 V (Space) vor.

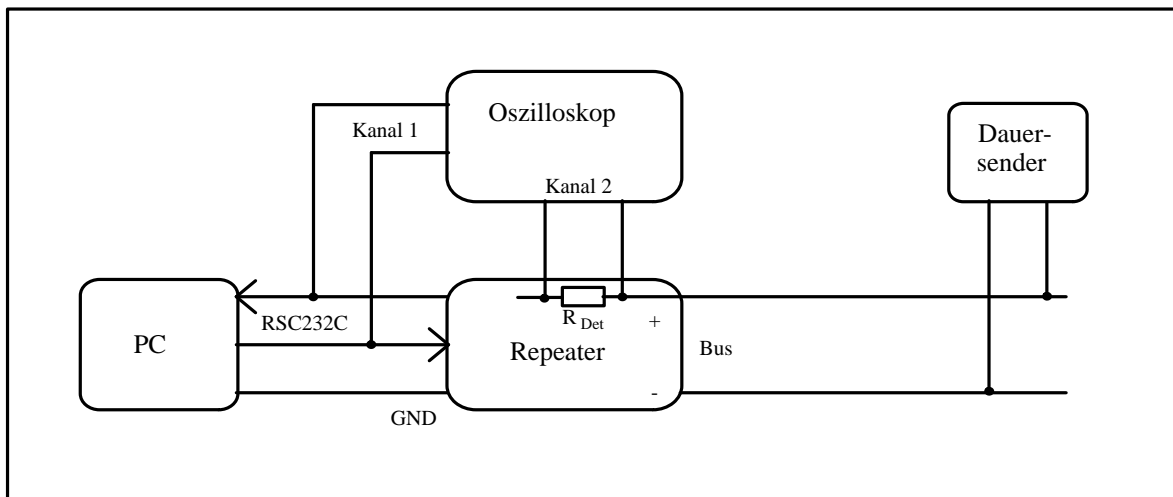
In Aufrufrichtung wird ein Rechtecksignal von einem PC über die serielle Schnittstelle auf den Repeater gegeben, und mit dem Ausgangssignal und dem Signal am Empfangspin (TXI-Pin) eines Slaves mit Hilfe eines Oszilloskopes verglichen. Abb. 17 zeigt den Aufbau mit einem Slave und 1 m Kabel.





**Abb. 17:** Aufbau der Messung zum Übertragungsverhalten in Aufrufrichtung

In Antwortrichtung wird als Sender ein spezieller Slave genutzt, der ständig ein Rechtecksignal liefert. Beim Digitalrepeater wird zusätzlich zum Ausgangssignal an der seriellen Schnittstelle noch das Signal am Detektionswiderstand ( $3,75 \Omega$ ) gemessen. Für diesen Fall zeigt Abb. 18 den Aufbau. Zur Feststellung der Bitzeiten, werden die Nulldurchgänge des Signals betrachtet.

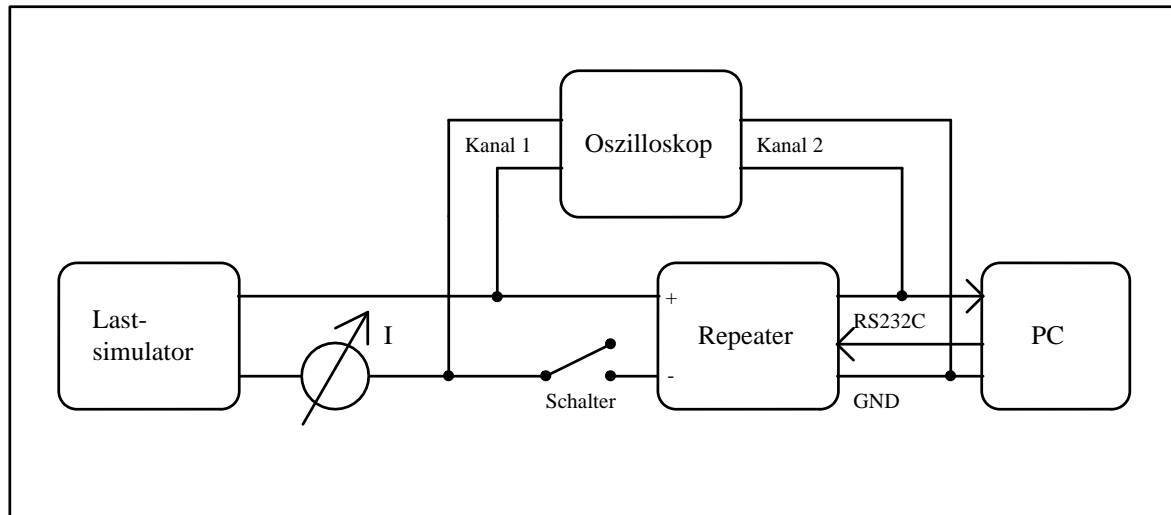


**Abb. 18:** Aufbau zur Messung des Übertragungsverhaltens in Antwortrichtung

- **Nachregelzeit**

Unter diesem Begriff wird die Zeit verstanden die der Repeater benötigt, um sich an erhöhten bzw. erniedrigten Ruhestrom aufgrund von zugeschalteten bzw. abgeschalteten Endgeräten einzustellen. In diesem Zeitraum ist der Repeater nicht übertragungsbereit.

Das Zuschalten von Endgeräten und der damit verbundene Anstieg des Stromes auf dem Bus wird vom Repeater zunächst als Space erkannt und somit auch über die serielle Schnittstelle an den PC weitergegeben. Sobald er sich auf den neuen Ruhestrom eingestellt hat, geht er in den Ruhezustand (Mark) über. Es wird die Spannung über der zugeschalteten Last und das Ausgangssignal an der seriellen Schnittstelle am Oszilloskop gemessen (siehe Abb. 19).



**Abb. 19:** Aufbau der Messung zum Zuschalten von Lasten

Die Messung zum Abschalten eines Ruhestroms wird mit Hilfe des dauernd sendenden Endgerätes sichtbar gemacht. Meßpunkte und Aufbau sind analog zur Abb. 19. Der Dauer-sender wird zwischen Schalter und Repeater plaziert.

- **Kollisionserkennung**

Ein dauernd sendendes Endgerät dessen Sendestrom sich variieren läßt wird verwendet, um festzustellen, wann der Repeater eine sogenannte Kollision, d.h. das Senden zweier oder mehrerer Endgeräte gleichzeitig, feststellt. Hierauf sollte der Repeater mit dem Senden eines Space über mindestens 50 ms in Richtung Master und Slaves reagieren. Dieser Vorgang wird auch als Break bezeichnet.

- **Schaltchwelle in Stromrichtung**

Das gleiche Endgerät wie oben wird eingesetzt, um festzustellen wann der Repeater ein Space detektiert. Hierzu wird sowohl das vom Slave gesendete Signal über dessen 430  $\Omega$  Widerstand, als auch das Ausgangssignal an der seriellen Schnittstelle aufgenommen.

- **Abfall der Busspannung bei Space in Antwortrichtung**

Um den Abfall der Markspannung bei sendenden Slaves zu messen, wird das oben beschriebene Endgerät auf seinen maximalen Sendestrom von 38,5 mA eingestellt. Das Ausgangssignal am Repeater und das gesendete Signal werden mittels einem Oszilloskop festgehalten.

- **Übertragung bei 300 Baud**

Bei dieser Messung werden 2048 Zeichen vom Wert 0 in Antwortrichtung mit 300 Baud übertragen, um die Funktion des Repeaters bei der niedrigsten möglichen Baudrate zu überprüfen.

- **Überstromanzeige/Warnstromanzeige**

Durch Erhöhung des Ruhestroms über einen Lastsimulator wird festgestellt, ob und wann die Überstrom bzw. Warnstromanzeige aktiviert wird.

- **Kurzschlußverhalten**

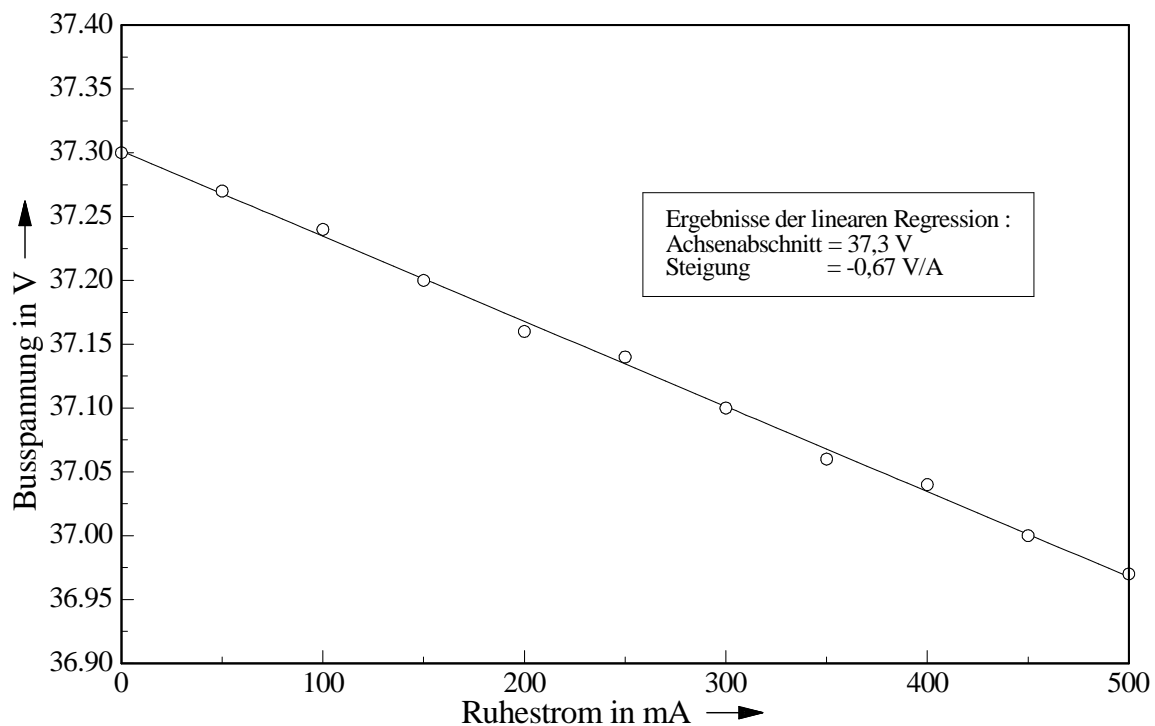
Zur Messung der Erholzeit nach Beendigung eines Kurzschlusses wird in Aufruf- und Antwortrichtung ständig ein Signal gesendet, welches mit Hilfe des Oszilloskopes sichtbar gemacht wird.

## 5.3. Ergebnisse und Auswertung der Messungen

### 5.3.1. Der Analog-Repeater

#### Busspannung am Repeater in Abhängigkeit vom Ruhestrom

Abb. 20 zeigt die Abhängigkeit zwischen der Busspannung und dem eingestellten Ruhestrom. Der Verlauf der Kurve ist linear. Die Ergebnisse der linearen Regression sind der Abbildung zu entnehmen. Die Busspannung fällt nicht unter den Mindestwert von 35 V. Selbst bei einem Ruhestrom von 500 mA liegt die Busspannung noch bei 36,97 V. Zum Vergleich: 250 Standardlasten + ein Kurzschluß ergeben einen Stromverbrauch von 465 mA.



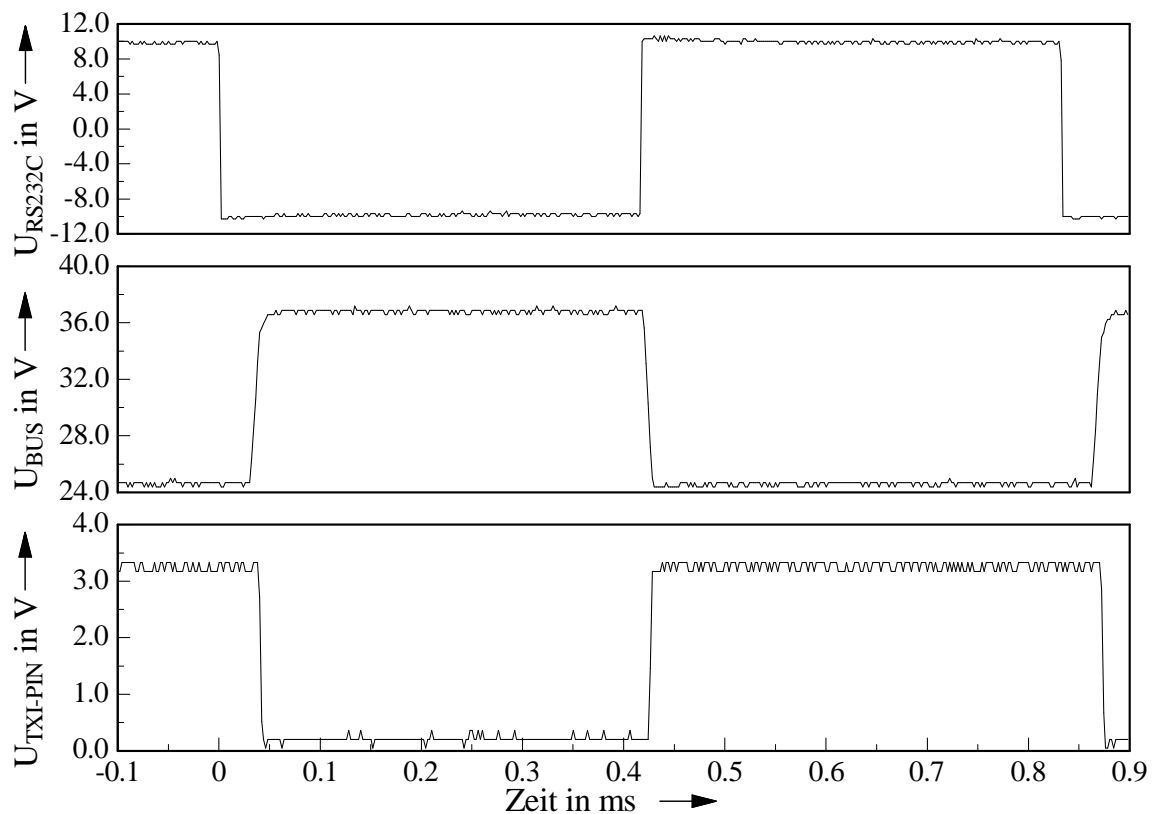
**Abb. 20:** Busspannung in Abhängigkeit vom Ruhestrom

#### Übertragungsverhalten

Die nachstehende Abb. 21 zeigt das Übertragungsverhalten in Aufrufichtung bei 2400 Baud, 1 m Kabel und einer Last. Der obere Teil der Abbildung zeigt das Eingangssignal, der untere Teil das Signal am Empfangspin des Endgerätes und der mittlere das Ausgangssignal des Repeaters, aus dem die Länge von Mark ( $388,5 \mu\text{s}$ ) und Space ( $444,5 \mu\text{s}$ ) bestimmt werden. Die Verkürzung einer Mark- und die Verlängerung einer Space-Bitzeit beträgt in diesem Fall  $28,0 \mu\text{s}$ . Hieraus resultiert eine Bitverzerrung von 6,7 % bei 2400 Bd und 27 % bei 9600 Baud. Der letzte Wert überschreitet die Bedingung, daß die

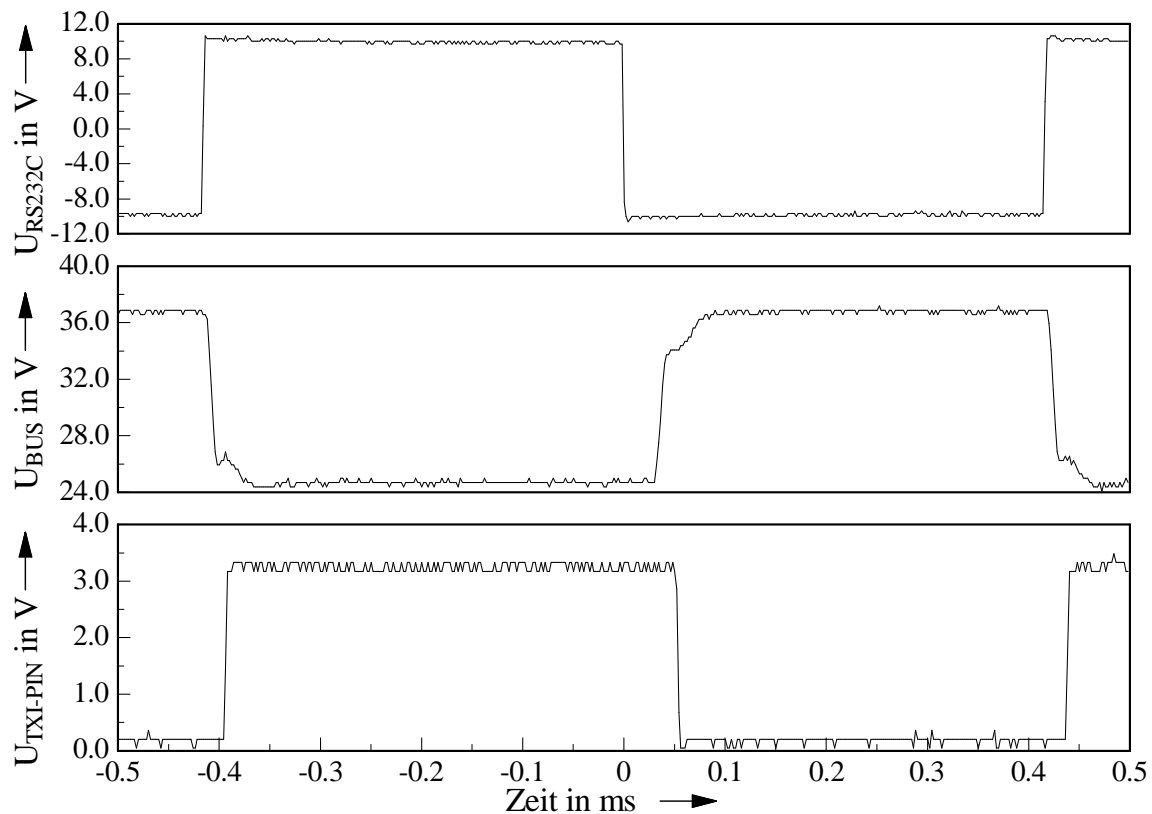
Bitverzerrung höchstens ein Viertel der Bitzeit betragen darf deutlich. Der Analog-Repeater ist für die Kommunikation bei 9600 Baud nicht geeignet. Die Anstiegsgeschwindigkeit der steigenden Flanke beträgt  $1,03 \text{ V}/\mu\text{s}$ , die der fallenden  $1,39 \text{ V}/\mu\text{s}$ . Diese Werte entsprechen den Spezifikationen.

Betrachtet man das Signal am Empfangspin des Endgerätes (Abb. 21 unten), so ergibt sich mit einer Bitverzerrung von  $31,5 \mu\text{s}$  (7,6 % bei 2400, 30,2 % bei 9600 Baud) ein noch schlechterer Wert.



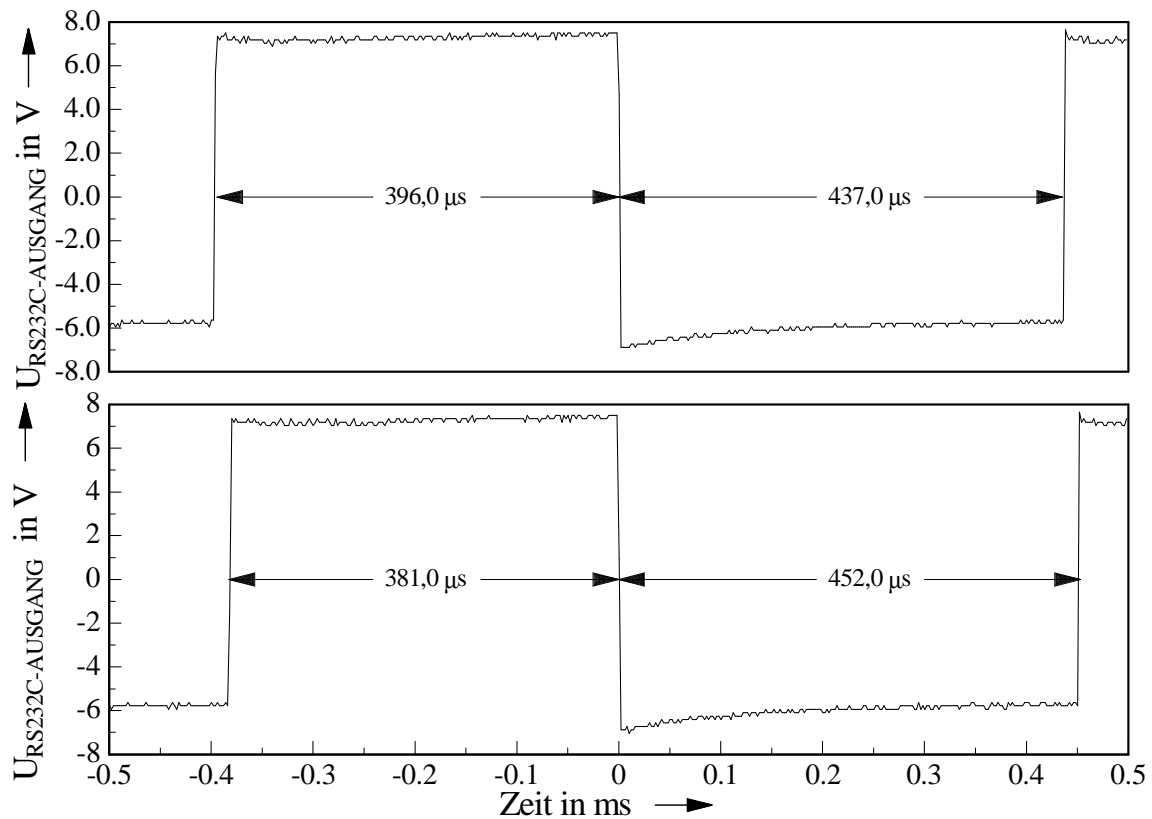
**Abb. 21:** Übertragungsverhalten in Aufrufriechung, 1 m Kabel, 1 Standardlast

Die Abb. 22 zeigt das Übertragungsverhalten in Aufrufriechung bei Anschluß von 250 Endgeräten, 1000 m Kabel und einer Baudrate von 2400 Baud. Unter diesen Bedingungen bleibt die Bitverzerrung gleich, für die Anstiegsgeschwindigkeiten der steigenden Flanke wurde ein Wert von  $0,28 \text{ V}/\mu\text{s}$  ermittelt, der Wert der fallende Flanke beträgt  $0,35 \text{ V}/\mu\text{s}$ . Dies entspricht den Spezifikationen.



**Abb. 22:** Übertragungsverhalten in Aufrufichtung, 1000 m Kabel, 250 Standardlasten

Das Verhalten in Antwortrichtung wurde unter Berücksichtigung der gleichen Parameter wie in Aufrufichtung gemessen. Abb. 23 zeigt das Ausgangssignal des Repeaters für die zwei gemessenen Fälle, wobei dem oberen Teil die besseren Bedingungen zugrunde liegen. Es ergibt sich eine Verlängerung der Mark-Bitzeit von  $35,5 \mu\text{s}$  (unten) bzw.  $20,5 \mu\text{s}$  (oben). Dies entspricht bei 2400 Baud einer Bitverzerrung von 8,5 % bzw. 4,9 %, und bei 9600 Baud einer Bitverzerrung von 34 % bzw. 19,6 %. Auch in dieser Richtung gilt, daß die Bitverzerrung bei den ungünstigen Bedingungen zu groß ist.

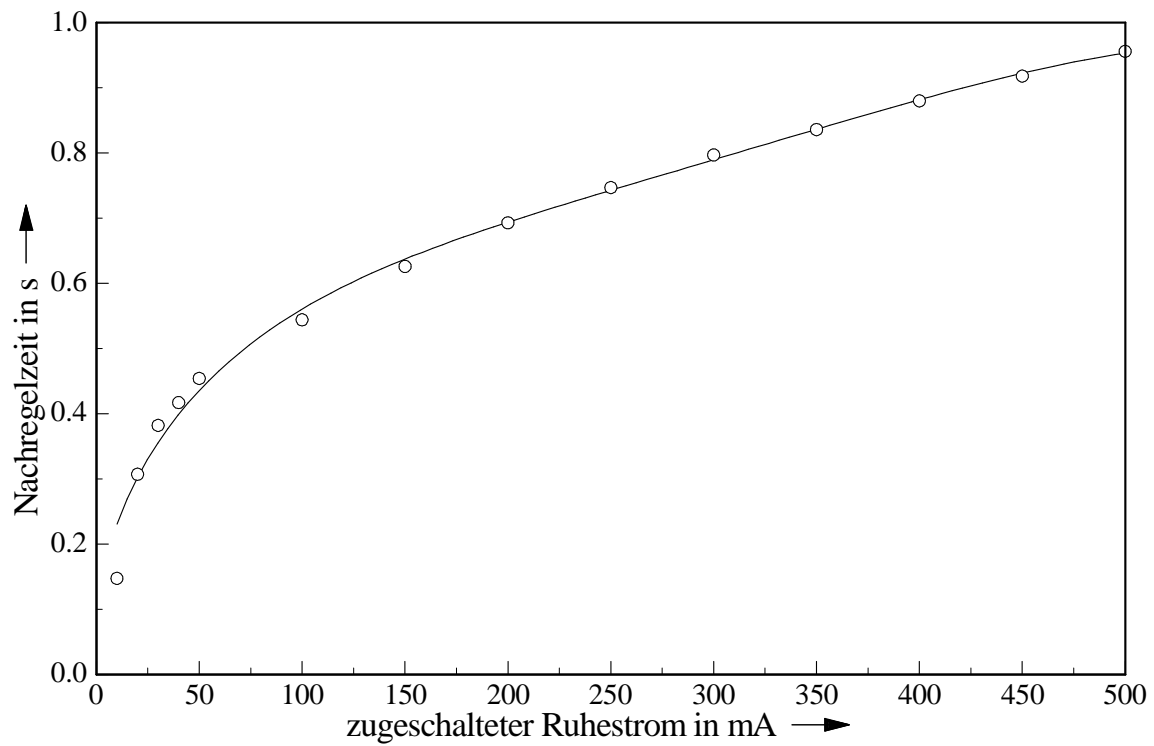


**Abb. 23:** Übertragungsverhalten in Antwortrichtungoben: 1 m Kabel, 1 Last  
 unten: 1 km Kabel, 250 Lasten

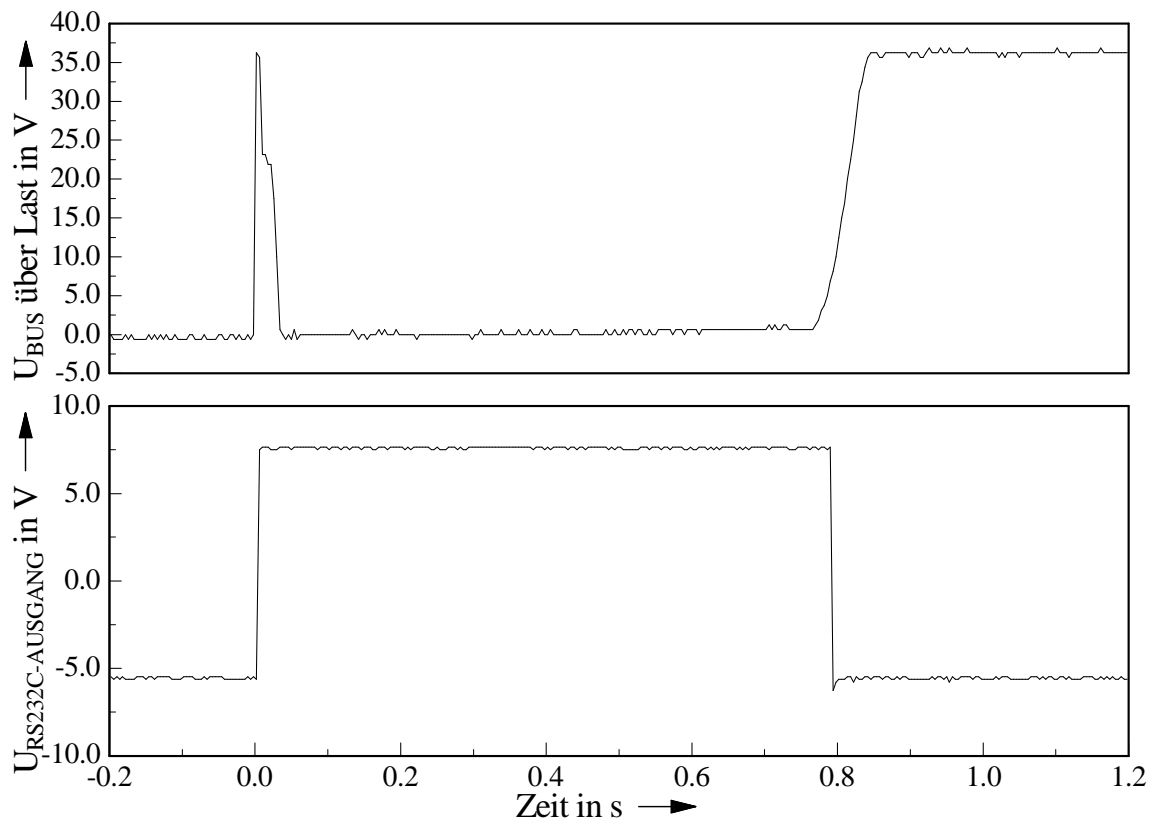
### Nachregelzeit

Die Abbildung 24 zeigt die Abhängigkeit der Nachregelzeit von dem zugeschalteten Ruhestrom. Der Verlauf der Kurve entspricht dem einer Wurzelfunktion. Abbildung 25 zeigt das Beispiel einer Meßwertaufnahme. Ein Strom von 300 mA wird zugeschaltet. Die obere Hälfte zeigt die Busspannung über der Last. Im Moment des Zuschaltens erhöht sich diese auf die nominelle Busspannung, fällt dann aber sofort wieder ab. Die Nachregelzeit entspricht der Zeit in der am RS232C-Ausgang ein Space anliegt.

Den Kurvenverlauf für die Nachregelzeit in Abhängigkeit des abgeschalteten Ruhestroms gibt Abb. 26 wieder. Abb. 27 zeigt ein Beispiel für das Abschalten eines Ruhestroms von 300 mA. Deutlich ist zu erkennen, daß ca. 2,5 Sekunden keine Übertragung des dauernd gesendeten Signales möglich ist.



**Abb. 24:** Nachregelzeit in Abhängigkeit des zugeschalteten Ruhestroms



**Abb. 25:** Nachregelzeit für das Zuschalten eines Ruhestroms von 300 mA, 0,5 m Kabel



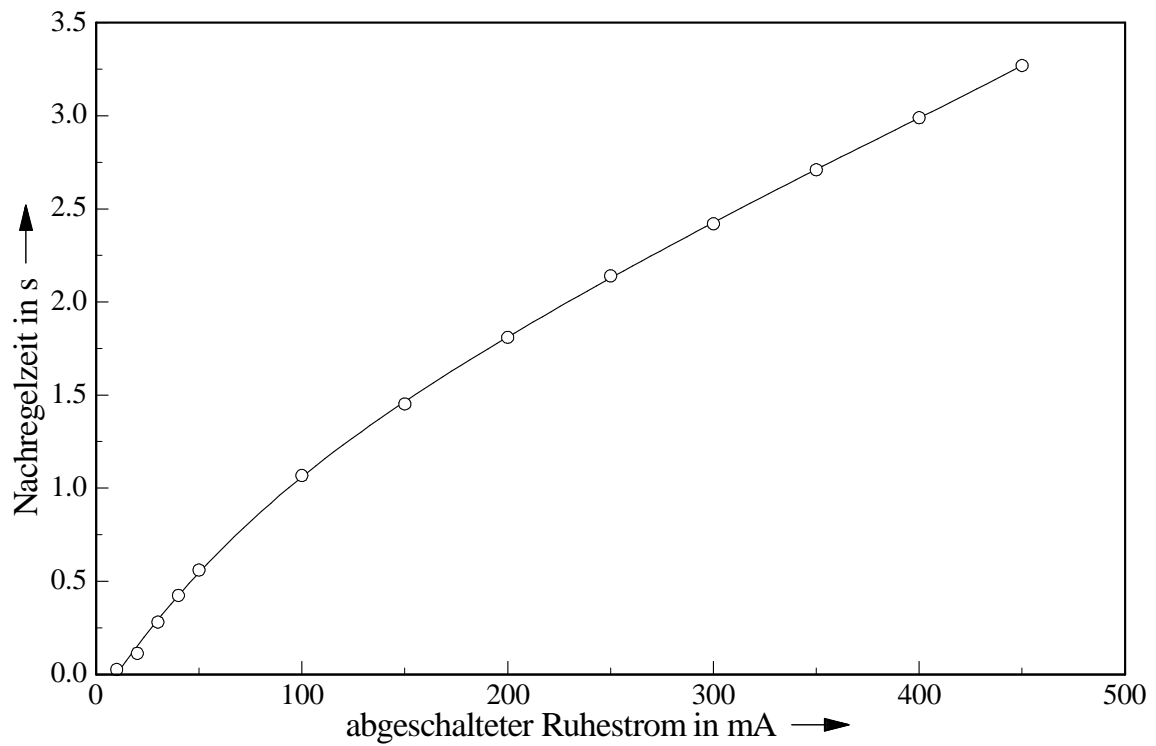


Abb. 26: Nachregelzeit in Abhängigkeit des abgeschalteten Ruhestroms

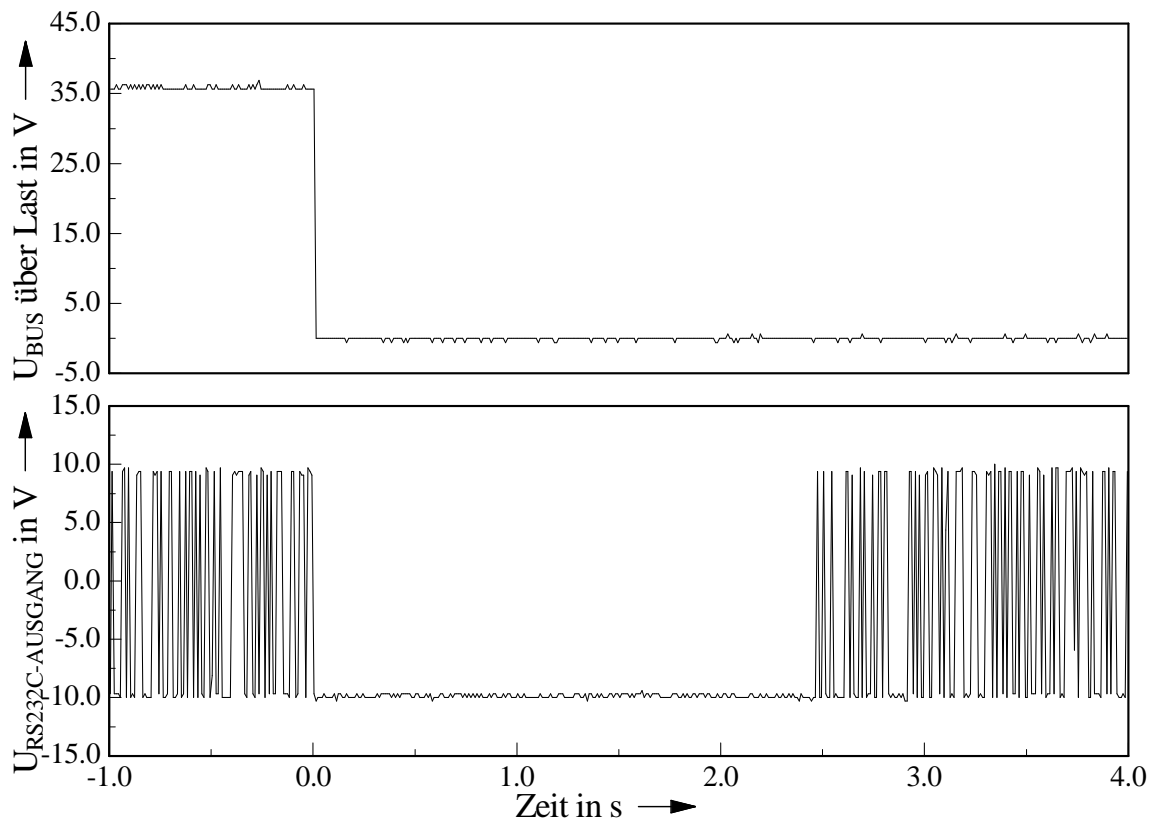
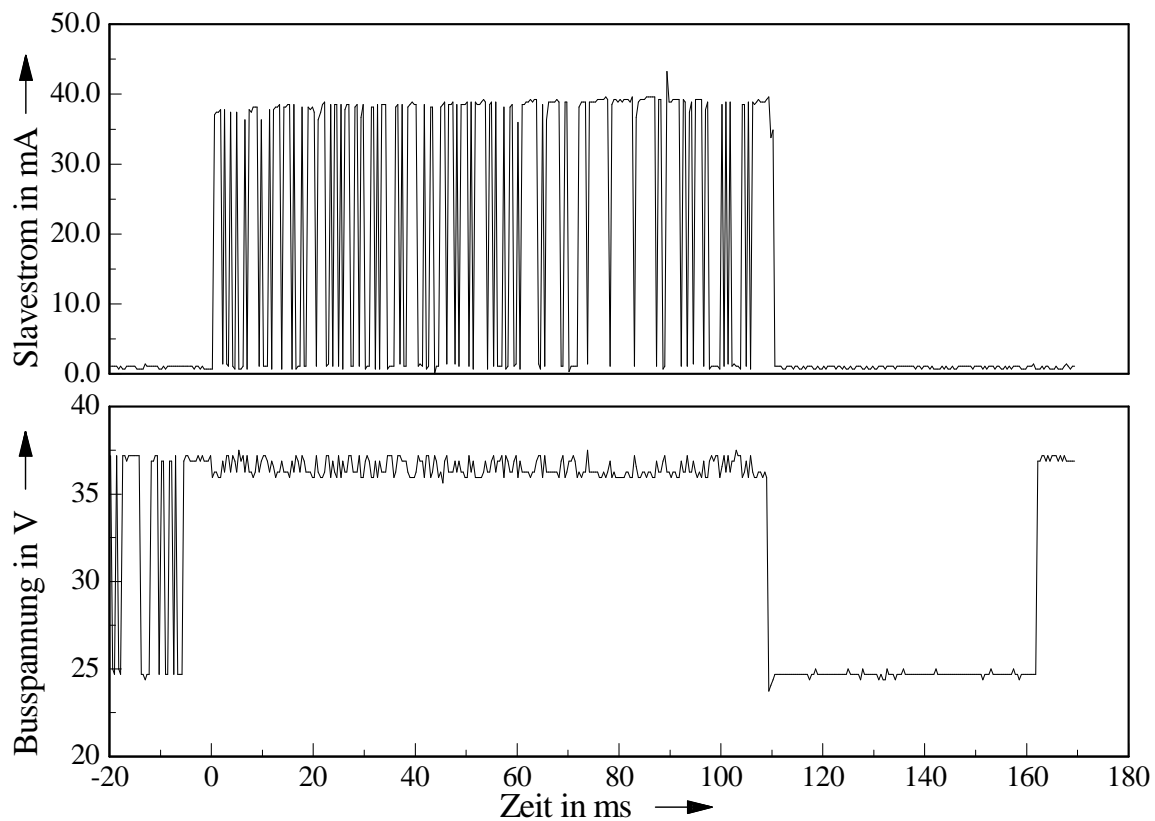


Abb. 27: Nachregelzeit für das Abschalten eines Ruhestroms von 300 mA, 0,5 m Kabel

### Kollisionserkennung

Die Kollisionserkennung erfolgt ab einem Sendestrom von 38,1 mA. Die geforderten 50 ms Space in Antwort und Aufrufriechung werden eingehalten. Abb. 28 zeigt eine Beispielmessung. Per REQ\_UD2 werden Daten eines Slaves angefordert, der so präpariert ist, daß er mit einem Sendestrom von 38,1 mA sendet. Die Kollisionserkennung erfolgt nach ca. 110 Millisekunden. Danach wird über 50 ms ein Space auf dem Bus gesendet.



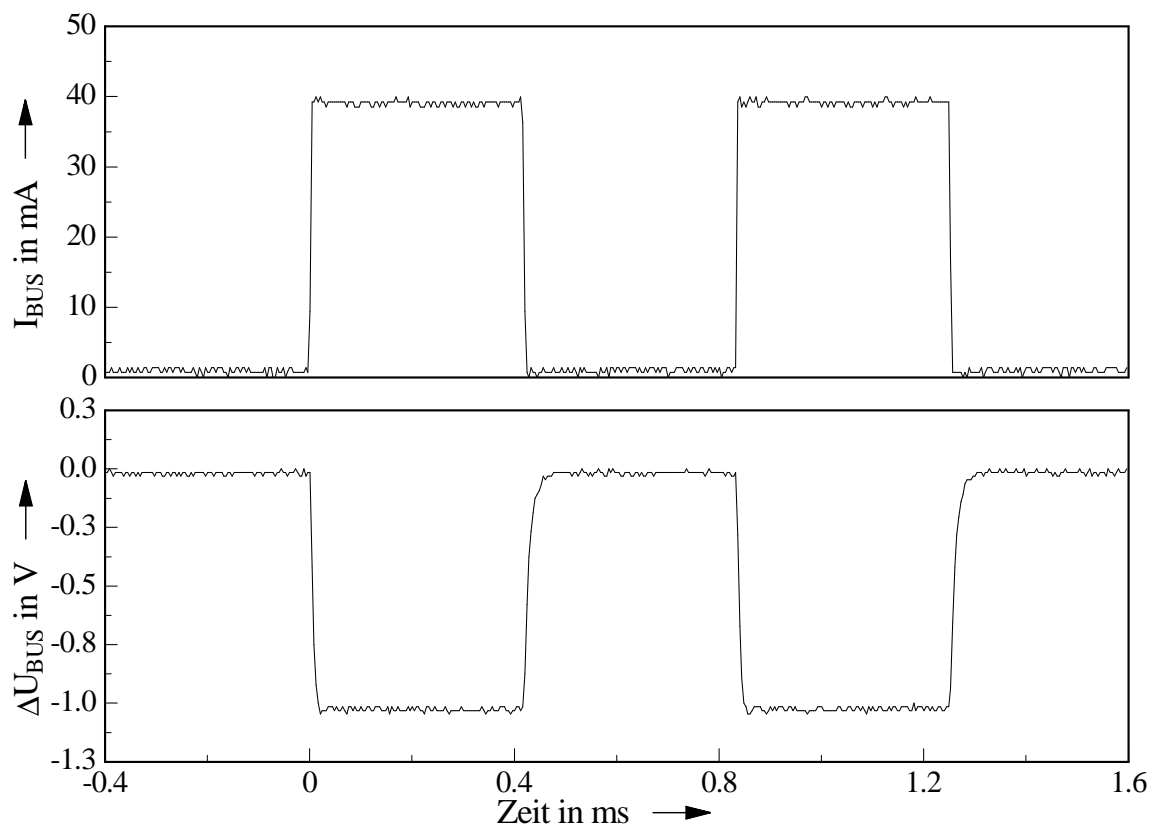
**Abb. 28:** Kollisionserkennung

### Schaltschwelle in Stromrichtung

Die Messung wurde mit verschiedenen Parametern durchgeführt. Bei einer angeschlossenen Last und 0,5 m Kabel liegt die Schwelle zur Detektion eines Space bei 7,2 mA. Bei einem Ruhestrom von 390 mA, voller Kapazität und 0,5 m Kabel bei 7,4 mA. Dies entspricht den Spezifikationen.

### Abfall der Busspannung bei einem Space in Antwortrichtung

Bei einem Sendestrom von 38,5 mA sinkt die Busspannung um 1,0 V und erfüllt damit die gestellten Anforderungen von einer maximalen Absenkung von 1,5 V. Abb. 29 veranschaulicht das Ergebnis. Bei der Aufnahme konnte davon profitiert werden, daß die Kollisionserkennung erst mit zeitlicher Verzögerung erfolgt.



**Abb. 29:** Abfall Markspannung bei Space in Antwortrichtung

### Übertragung bei 300 Baud

Bei einem Sendestrom von 11 mA des Slaves waren 236 der 2048 Zeichen fehlerhaft. Das bedeutet, daß 88,5 % der Zeichen (Wert 0) richtig übertragen wurden. Bei einem Sendestrom von 20 mA hingegen, beträgt die Rate der erfolgreichen Datenübermittlung schon 99,9 %. Zwei Zeichen wurden fehlerhaft übermittelt. Bei diesen beiden Messungen betrug die insgesamt verwendete Kabellänge 1000 m. Der Ruhestrom lag bei 390 mA.

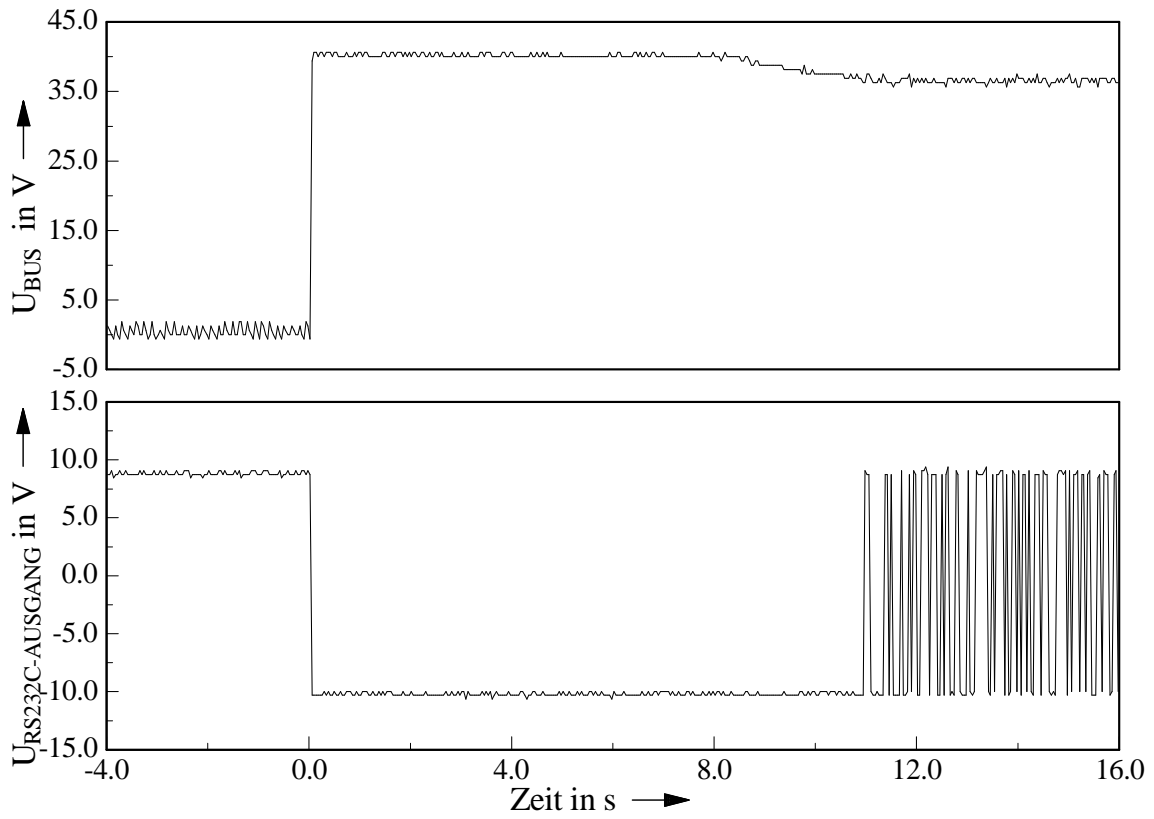
### Überstromanzeige/Warnstromanzeige

Die Warnstromanzeige wird bei 418 mA die Überstromanzeige bei 535 mA aktiviert. Beide werden durch jeweils eine leuchtende LED realisiert.

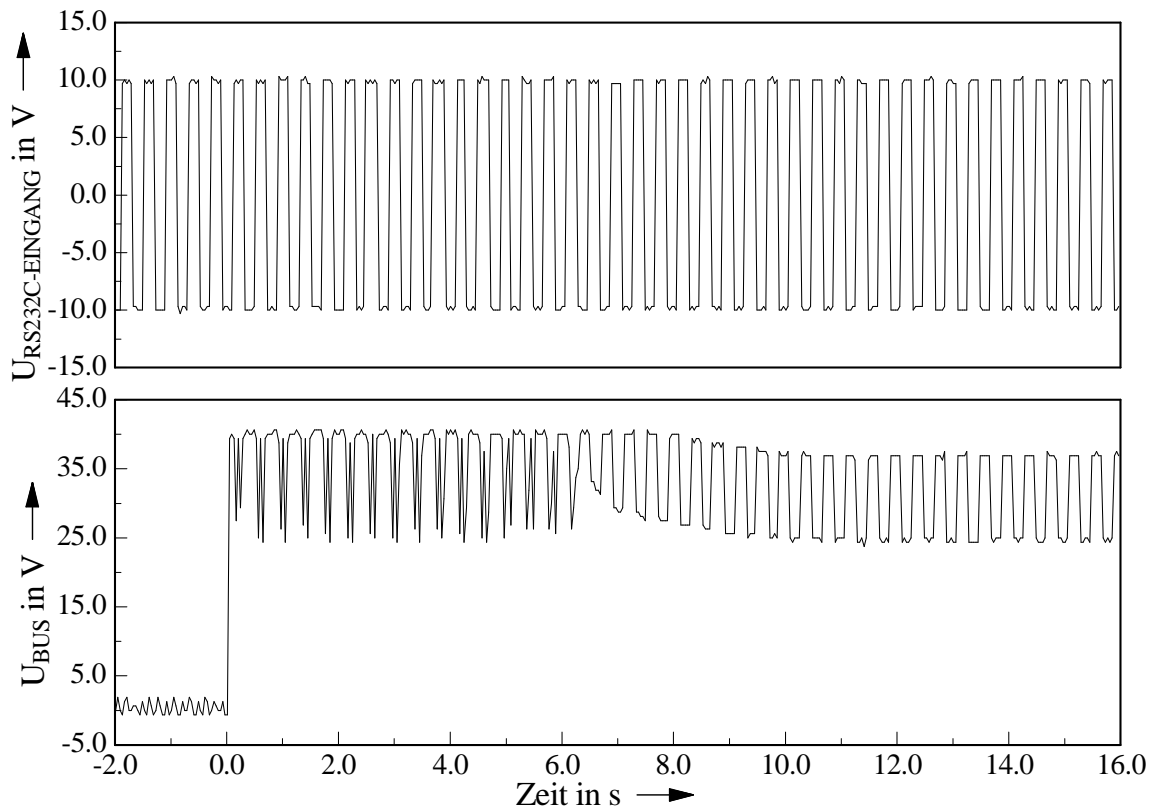
### Kurzschlußverhalten

Nach einem Kurzschluß bedarf es einer gewissen Zeit, bis sich der Repeater hiervon erholt und wieder übertragungsbereit ist. Aus Abb. 30 unten ist zu entnehmen, daß diese Zeit in Antwortrichtung 11,5 Sekunden beträgt. Die daran anschließende Abbildung zeigt das Verhalten des Repeaters in Aufrufrichtung. Die Feststellung der Erholzeit ist hier nicht genau möglich. Sie liegt im Bereich von 8 Sekunden. Beide Messungen wurden bei 1 m Kabel und einer angeschlossenen Standardlast gemacht.

Beim "Reinfahren" in einen Überstromzustand läßt sich ein mittlerer Strom auf dem Bus von 347 mA messen. Der Repeater schaltet den Bus immer wieder kurz ein, um festzustellen, ob der Überstromzustand noch Gültigkeit besitzt.



**Abb. 30:** Verhalten eines Repeaters nach einem Überstromzustand in Antwortrichtung

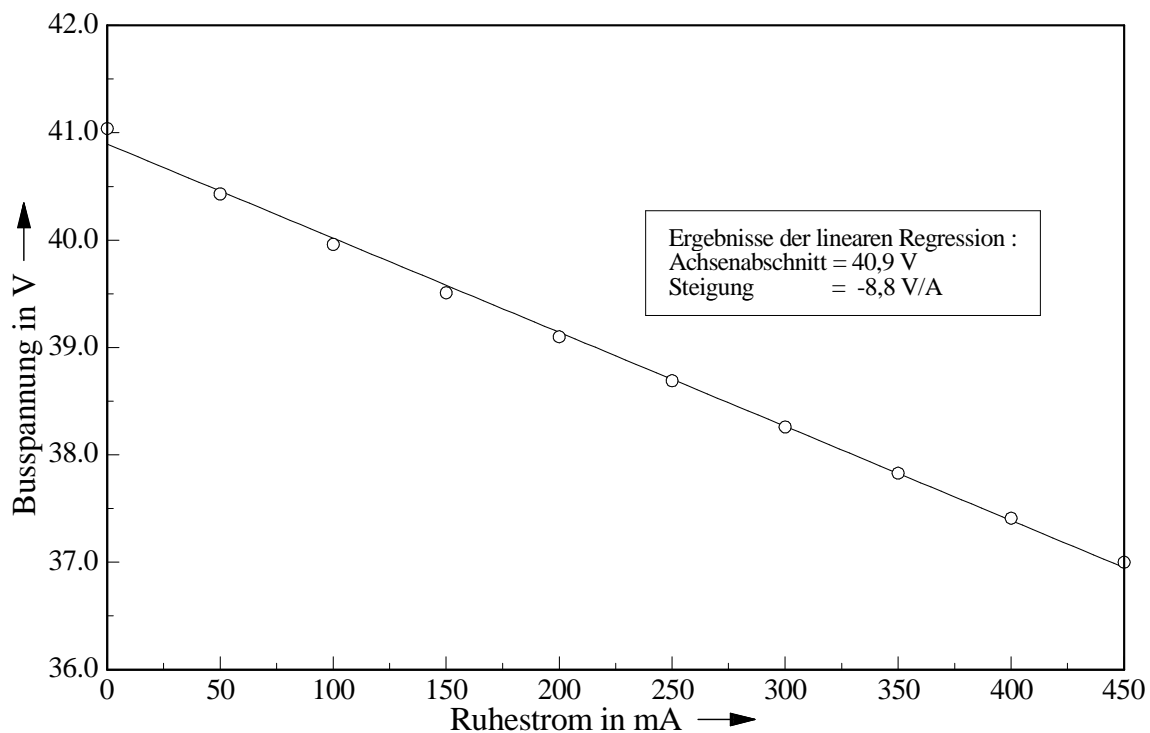


**Abb. 31:** Verhalten eines Repeaters nach einem Überstromzustand in Aufrufichtung

### 5.3.2. Digital-Repeater

#### Busspannung in Abhängigkeit des Ruhestroms

Die Abb. 32 zeigt die Busspannung in Abhängigkeit des Ruhestroms. Es ergibt sich ein linearer Verlauf. Die Ergebnisse der linearen Regression sind in der Abbildung festgehalten. Die Anforderung, daß die Busspannung am Repeater-Ausgang mindestens 35 V betragen muß, wird über den gesamten Bereich erfüllt.



**Abb. 32:** Busspannung in Abhängigkeit des Ruhestroms

#### Übertragungsverhalten

Der Digital-Repeater ist im Gegensatz zum Analog-Repeater dazu in der Lage stark verzerrte Signale wiederherzustellen (Bitrecovery). Er besitzt eine sogenannte Repeaterfunktion. Zur Überprüfung dieser Funktion werden definierte asymmetrische Signale von einem Pulsgenerator in Auf- und Antwortrichtung auf den Repeater gegeben und die ausgegebene Space-Bitzeit in Abhängigkeit der Eingangs-Space-Bitzeit notiert. Als Randbedingungen wurden 1 m Kabel und 1 Standardlast notiert.

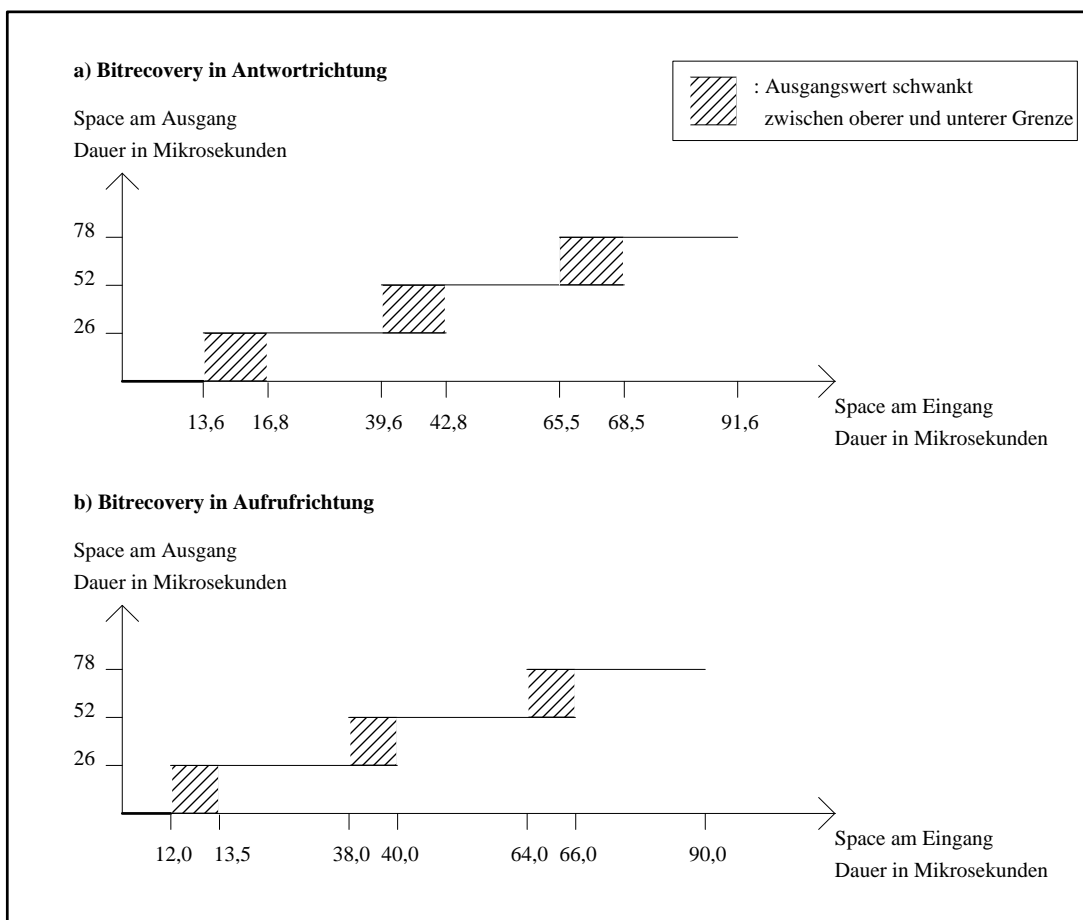
Die Messung ergibt in Antwortrichtung folgende Werte:

- $13,6 \mu\text{s} > \text{Space}_{\text{Eingang}}$   $\Rightarrow$  sicher nicht erkannt  
 $16,8 \mu\text{s} \leq \text{Space}_{\text{Eingang}} \leq 39,6 \mu\text{s}$   $\Rightarrow$  26  $\mu\text{s}$  Space am Ausgang  
 $42,8 \mu\text{s} \leq \text{Space}_{\text{Eingang}} \leq 65,5 \mu\text{s}$   $\Rightarrow$  52  $\mu\text{s}$  Space am Ausgang  
 $68,5 \mu\text{s} \leq \text{Space}_{\text{Eingang}} \leq 91,6 \mu\text{s}$   $\Rightarrow$  78  $\mu\text{s}$  Space am Ausgang

In Aufrufrichtung konnten folgende Werte beobachtet werden:

- $12,0 \mu\text{s} > \text{Space}_{\text{Eingang}}$   $\Rightarrow$  sicher nicht erkannt  
 $13,5 \mu\text{s} \leq \text{Space}_{\text{Eingang}} \leq 38,0 \mu\text{s}$   $\Rightarrow$  26  $\mu\text{s}$  Space am Ausgang  
 $40,0 \mu\text{s} \leq \text{Space}_{\text{Eingang}} \leq 64,0 \mu\text{s}$   $\Rightarrow$  52  $\mu\text{s}$  Space am Ausgang  
 $66,0 \mu\text{s} \leq \text{Space}_{\text{Eingang}} \leq 90,0 \mu\text{s}$   $\Rightarrow$  78  $\mu\text{s}$  Space am Ausgang

Diesen Sachverhalt gibt Abb. 33 graphisch wieder.



**Abb. 33:** Bitrecovery in Antwort- und Aufrufrichtung (nicht maßstabsgetreu)

Die Abbildungen 34 und 35 zeigen jeweils ein Beispiel für den korrigierenden Einfluß des Digital-Repeaters auf die Bitzeiten in Antwort und Aufrufrichtung.

Die folgenden vier Abbildungen (36, 37, 38, 39) beschäftigen sich mit dem Übertragungsverhalten in Aufrufrichtung. Sie zeigen die Übergänge von Mark zu Space bzw. Space zu Mark bei unterschiedlichen Bedingungen. Den ersten beiden Abbildungen liegen Messungen zugrunde, die bei 1 m Kabel, einer Standardlast und 9600 Baud gemacht wurden. Die daran anschließenden Abbildungen beziehen sich auf Messungen unter erschwerten Bedingungen, nämlich mit 1000 m Kabel, 250 Standardlasten und 9600 Baud.

Aufgrund des Bitrecovery dieses Repeaters kommt es nur zu geringen Bitverzerrungen. Benutzt man das Ausgangssignal des Repeaters (mittlere Teile der Abbildungen) zur Berechnung, so erhält man eine Verlängerung der Space-Bitzeit bzw. eine Verkürzung der Mark-Bitzeit um  $0,3 \mu\text{s}$  unter den leichteren und  $1,5 \mu\text{s}$  unter den erschwerten Bedingungen. Bei 9600 Baud entspricht der letzte Wert  $1,4 \%$ , bei 38400 Baud  $5,76 \%$  einer Bitzeit. Dies entspricht den Anforderungen.

Die Anstiegsgeschwindigkeiten der Flanken liegen ebenfalls innerhalb der Anforderungen. Unter den leichteren Bedingungen ergibt sich für die steigende Flanke ein Wert von  $3,56 \text{ V}/\mu\text{s}$  und für die fallende Flanke ein Wert von  $4,13 \text{ V}/\mu\text{s}$ . Unter den erschwerten Bedingungen ergibt sich für die Anstiegsgeschwindigkeit der steigenden Flanke  $2,96 \text{ V}/\mu\text{s}$  und für die fallende Flanke  $4,08 \text{ V}/\mu\text{s}$ .

Die Berechnung der Bitverzerrung aus den Signalen am Empfangspin des Endgerätes liefert andere Werte, da hier die physikalischen Eigenschaften des Busses, insbesondere unter den erschwerten Bedingungen, eine größere Rolle spielen. Unter den leichten Bedingungen ergibt sich die Bitverzerrung zu  $0,4 \mu\text{s}$ , unter den erschwerten Bedingungen beträgt sie  $2,4 \mu\text{s}$ . Das entspricht bei 38400 Baud  $1,5 \%$  bzw.  $9,1 \%$  einer Bitzeit. Auch diese Angaben entsprechen den Spezifikationen.



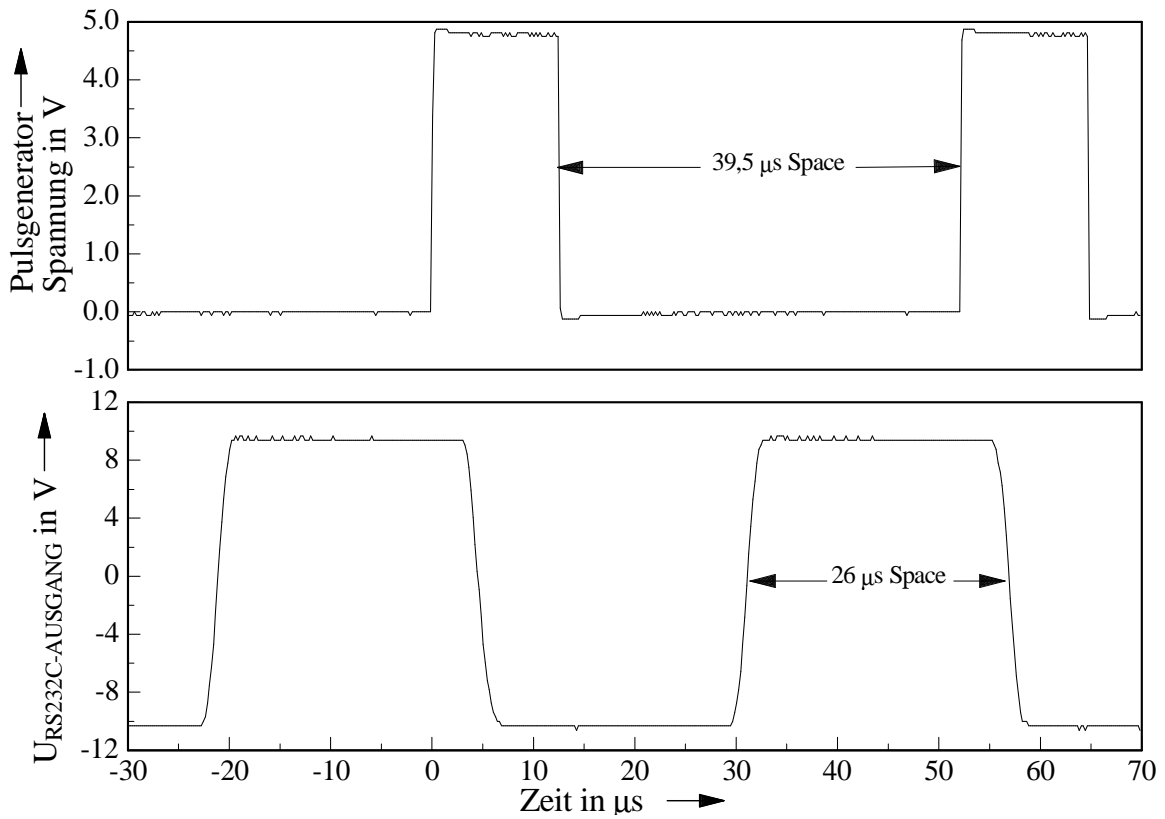


Abb. 34: Bitrecovery in Antwortrichtung

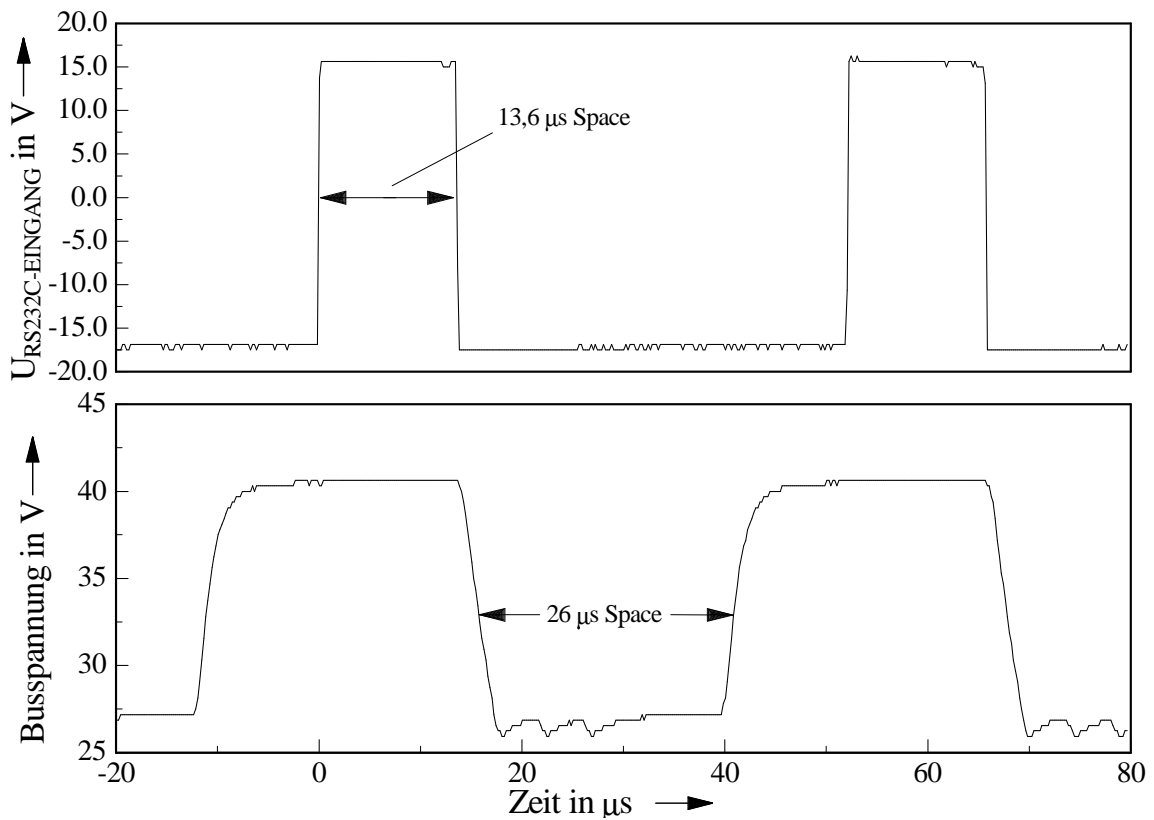
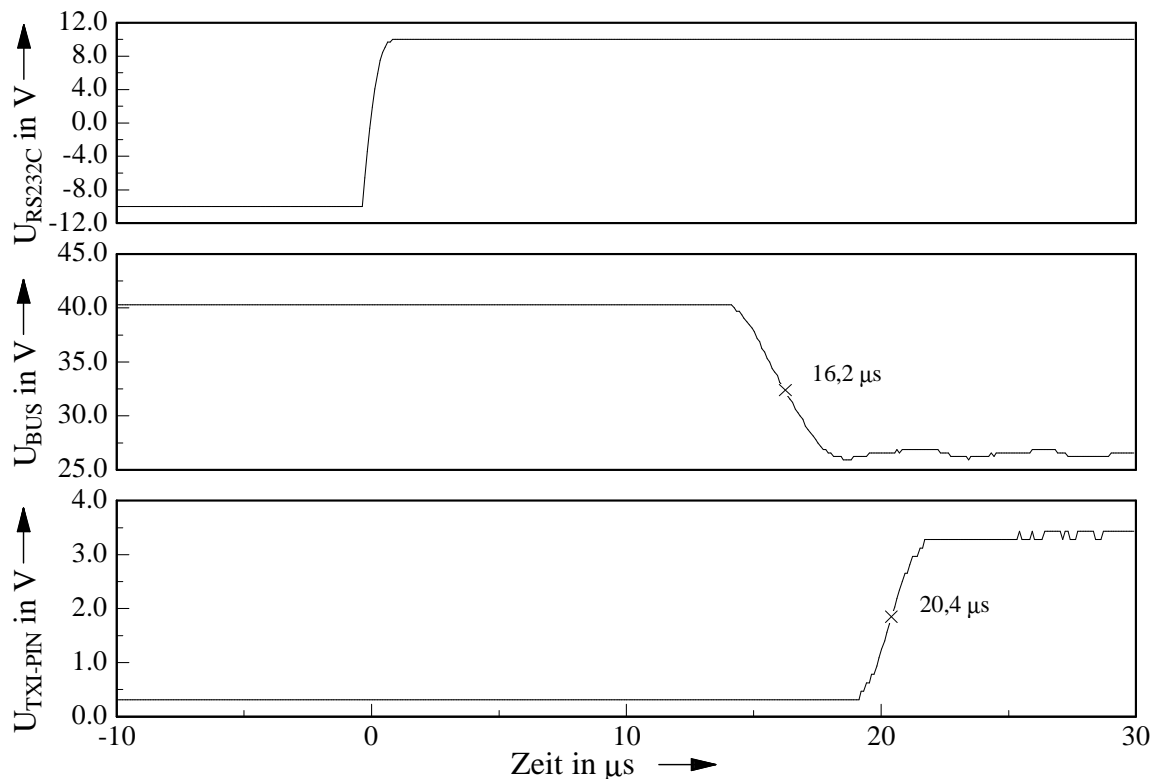
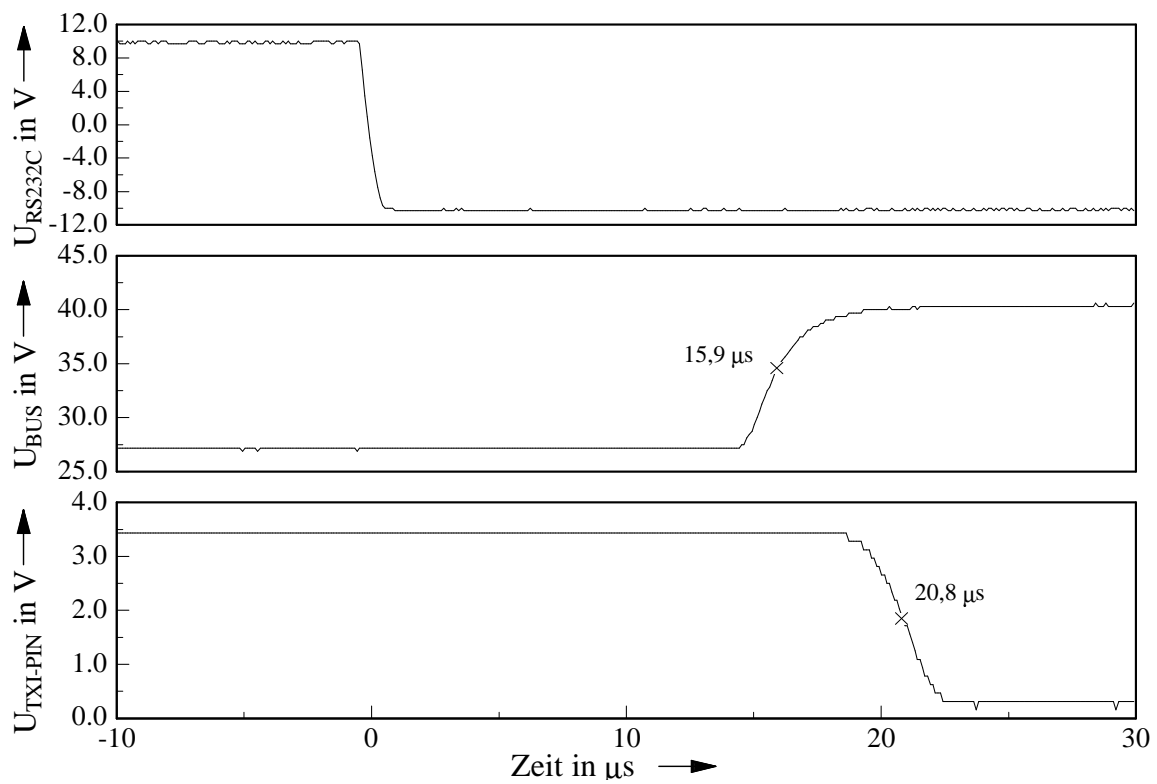


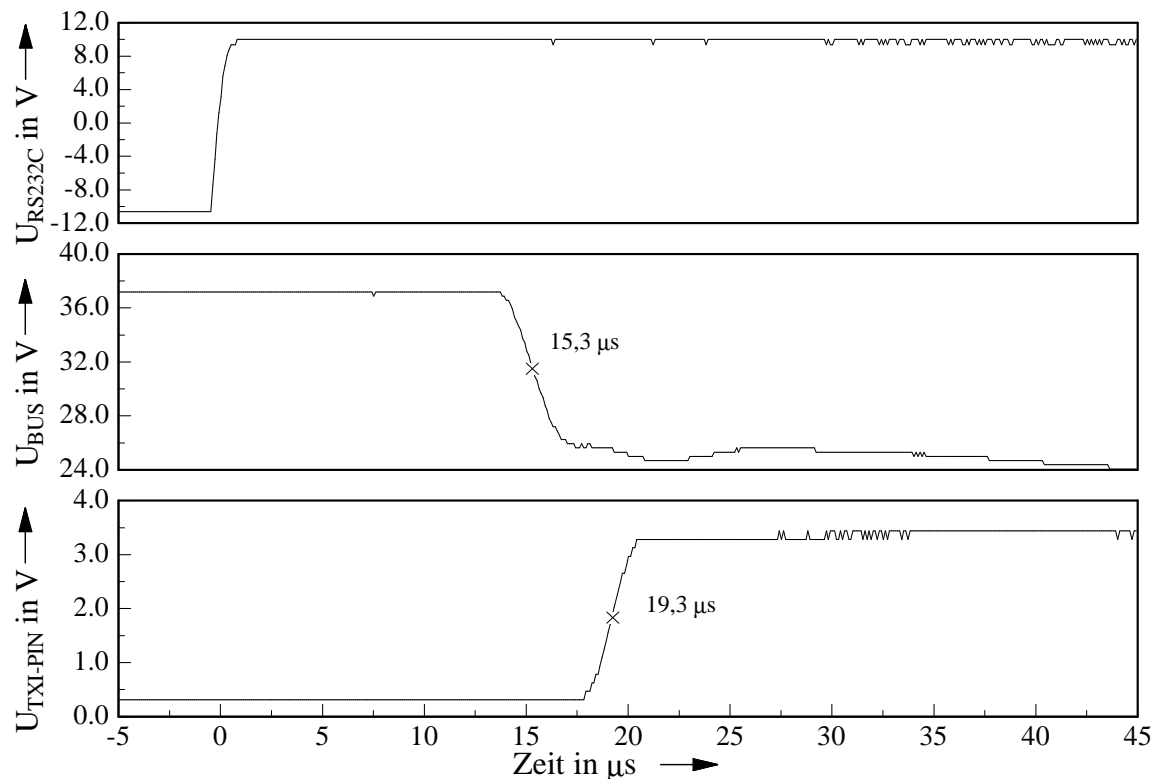
Abb. 35: Bitrecovery in Aufrufrichtung



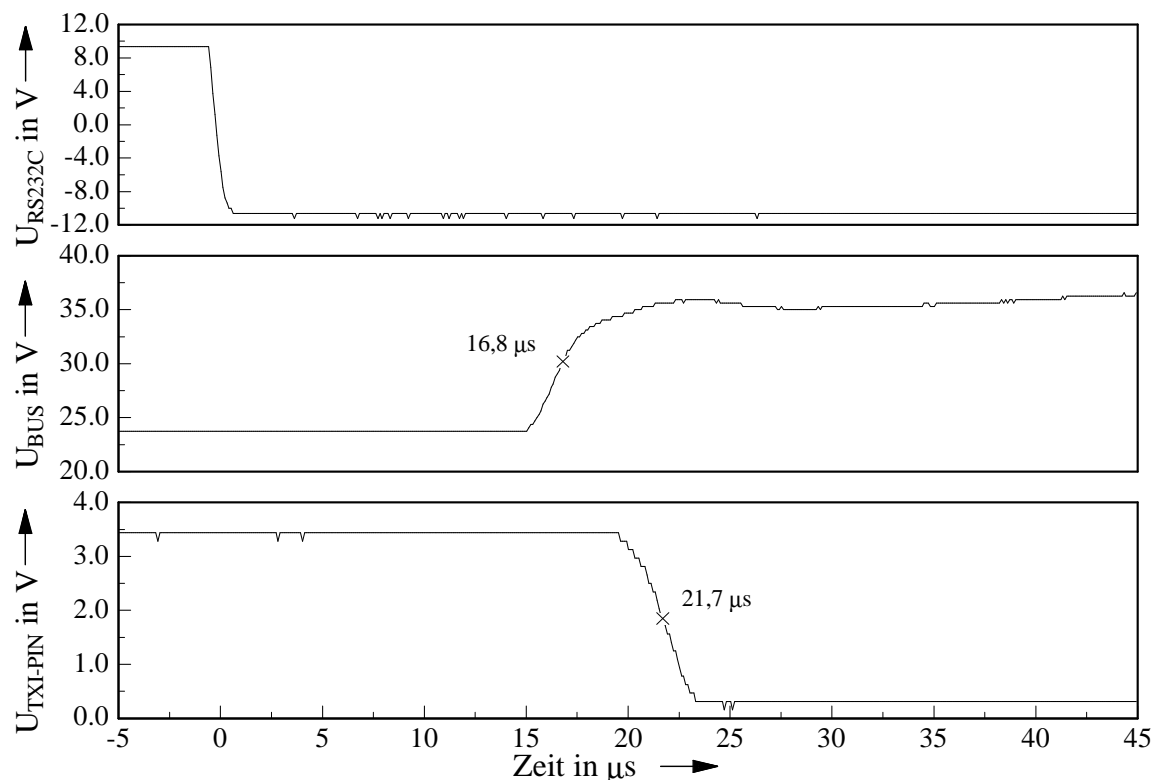
**Abb. 36:** Übertragungsverhalten in Aufrufrichtung, 1 m Kabel, Übergang Mark zu Space



**Abb. 37:** Übertragungsverhalten in Aufrufrichtung, 1 m Kabel, Übergang Space zu Mark



**Abb. 38:** Übertragungsverhalten in Aufrufichtung, 1 km Kabel, Übergang Mark zu Space



**Abb. 39:** Übertragungsverhalten in Aufrufichtung, 1 km Kabel, Übergang Space zu Mark

Die beiden Abbildungen 40 und 41 zeigen das Übertragungsverhalten in Antwortrichtung. Die erste wurde unter den Bedingungen von einer angeschlossenen Standardlast, 1 m Kabel

und 38400 Baud aufgenommen. Der zweiten liegt eine Messung zugrunde, bei der 250 Standardlasten, 1000 m Kabel und eine Baudrate von 38400 Baud eingesetzt wurden. Es wird jeweils das Signal am Detektionswiderstand des Repeaters ( $3,75 \Omega$ ), mit dem Signal an der seriellen Schnittstelle verglichen. Aufgrund des Bitrecovery erzeugt der Repeater symmetrische Signale, die die Anforderungen bei 38400 Baud erfüllen.

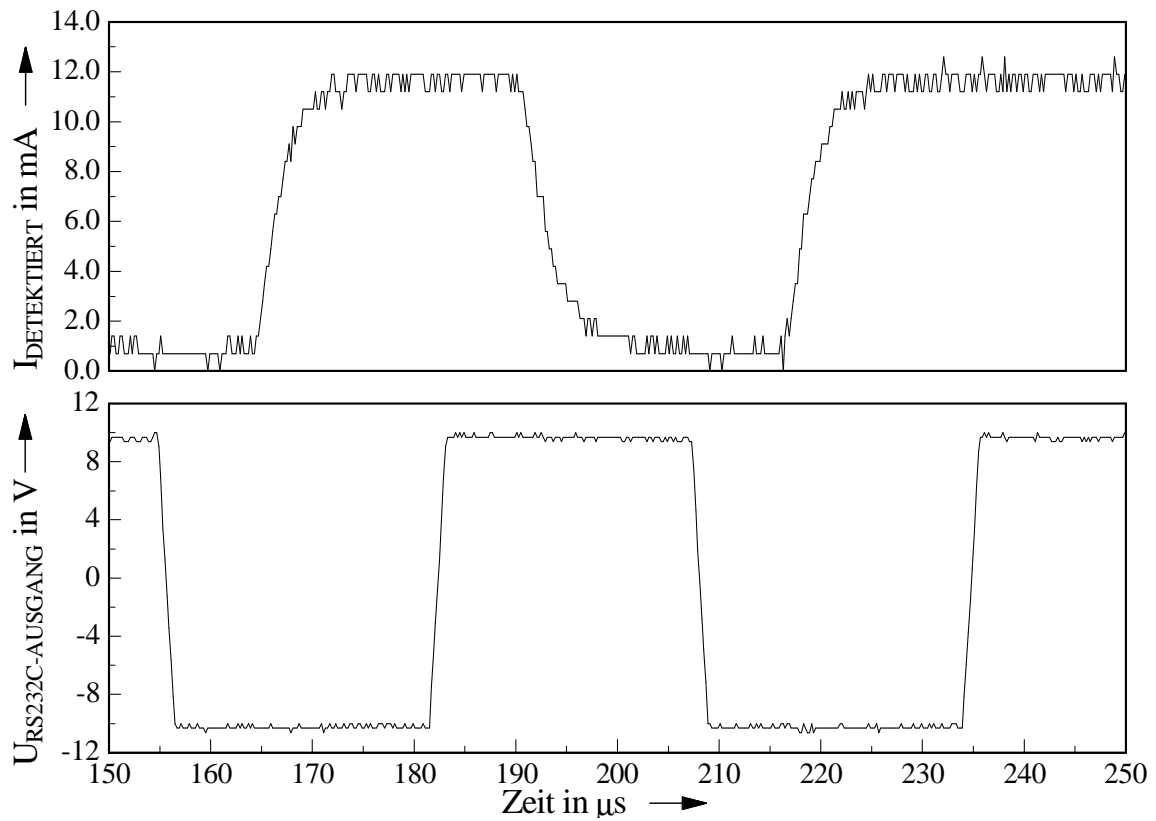
### Kollisionserkennung

Ab einem um 30,5 mA erhöhten Strombedarf schließt der Digital-Repeater auf eine Kollision auf dem Bus. Hiernach wird 50,2 ms ein Space in Richtung Slaves und Master gesendet. Diesen Sachverhalt gibt Abb. 42 wieder. Per REQ\_UD2 werden Daten von einem Slave angefordert, der mit einem Sendestrom von 30,5 mA antwortet. Die Kollisionserkennung erfolgt wesentlich schneller (ca. 2 ms) als bei dem Analog-Repeater.

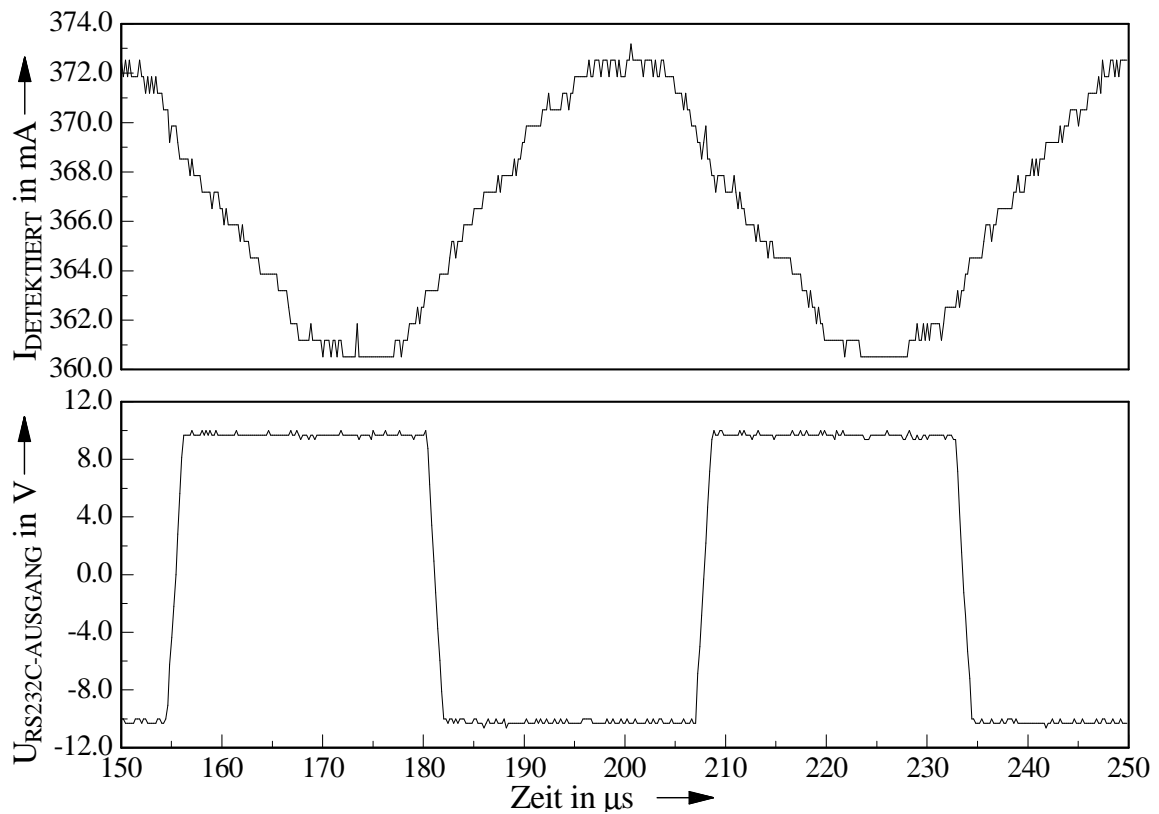
### Nachregelzeit

Die Abhängigkeit der Nachregelzeit von dem abgeschalteten Ruhestrom zeigt die Abb. 43. Man erkennt zwei verschiedene lineare Kurvenverläufe. Die untere Gerade entsteht, da der Repeater eine Kollision erkennt. Das Abschalten des Ruhestroms ist verbunden mit einer Art Tastenprellen. Der Bus wird im Bereich einiger Mikrosekunden an und abgeschaltet (Abb. 44 unten). Es werden kurzfristig hohe Ströme bis zu 350 mA vom Repeater detektiert (Abb. 44 oben). Ist ein so erzeugter Strompeak um 30,5 mA größer als der abgeschaltete Ruhestrom, so erkennt der Repeater ein Break (siehe Abb. 45a)). Die Meßwertaufnahme für diesen Fall zeigt die Abb. 46 am Beispiel des Abschaltens eines Ruhestroms von 50 mA.

Für den Fall, daß der aus dem Tastenprellen resultierende Strompeak unterhalb der Kollisionsschwelle bleibt (Abb. 45b)), ergibt sich die obere Gerade in Abb. 43. Im Bereich von 200 mA bis 300 mA lassen sich beide Fälle beobachten, da die erzeugten Ströme durch das Tastenprellen in ihrer Größe variieren.



**Abb. 40:** Übertragungsverhalten in Antwortrichtung, 1 m Kabel, 1 Standardlast



**Abb. 41:** Übertragungsverhalten in Antwortrichtung, 1000 m Kabel, 250 Standardlasten

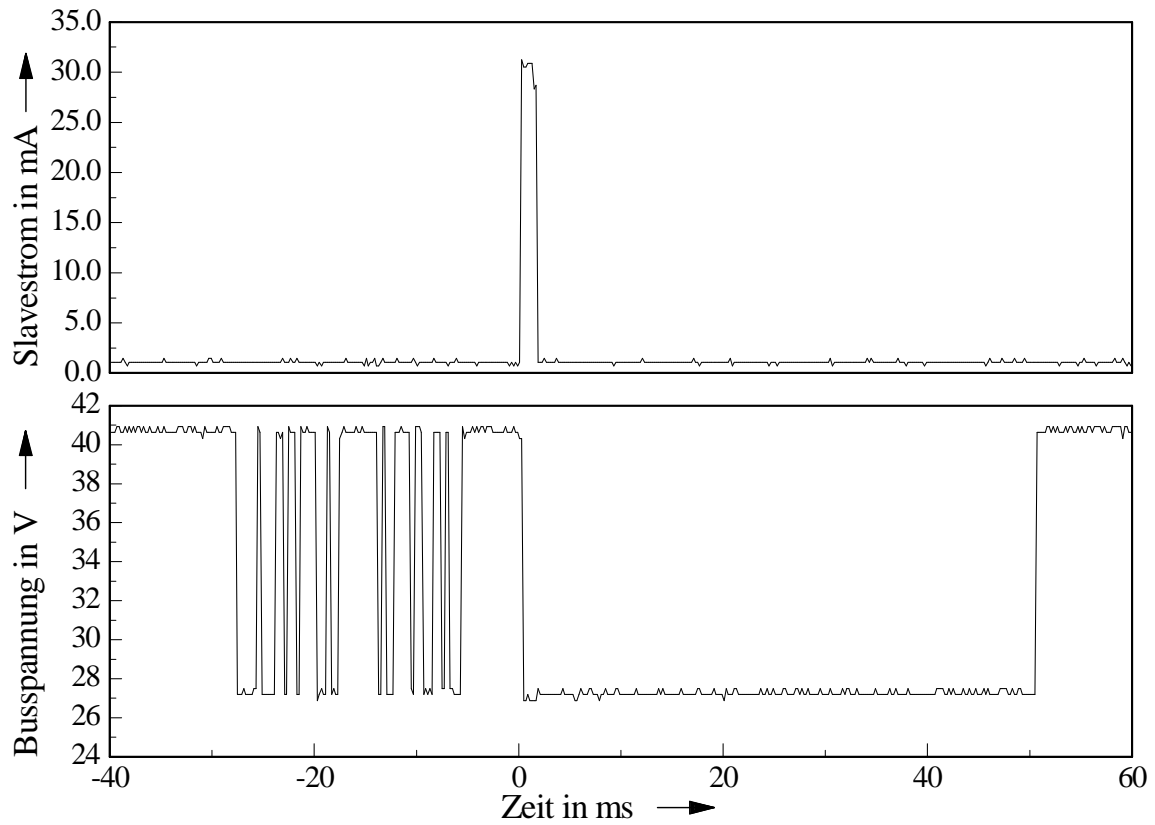


Abb. 42: Messung zur Kollisionserkennung

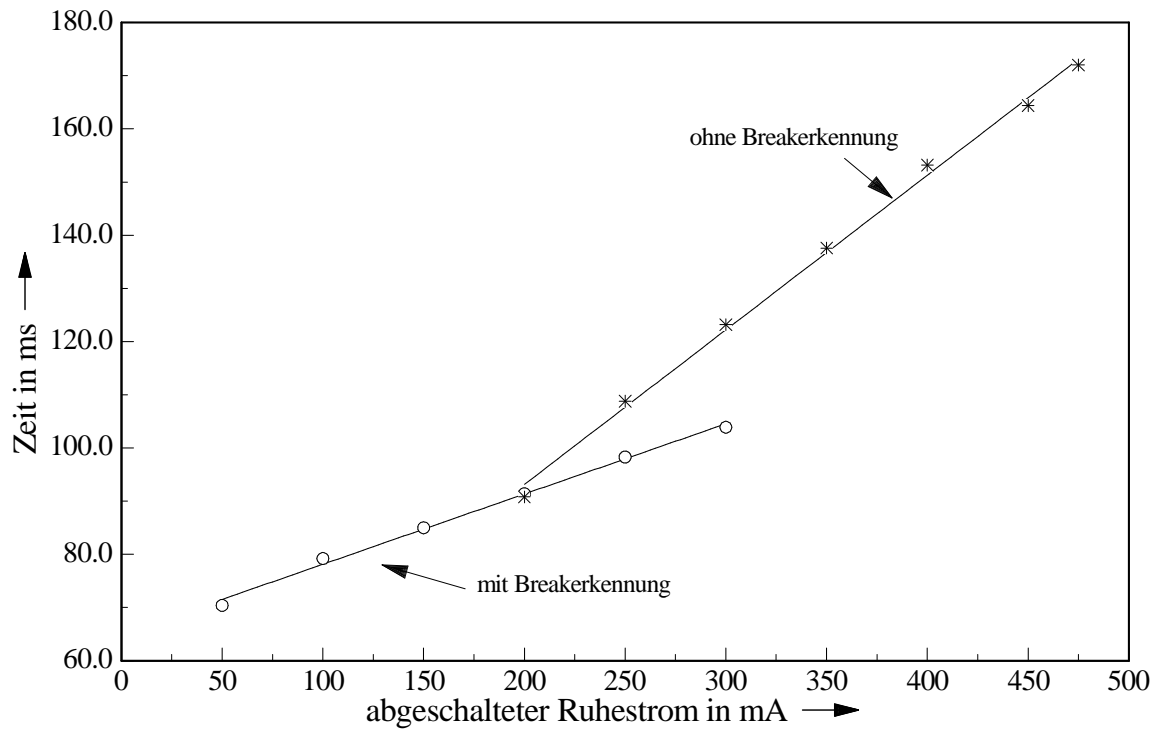


Abb. 43: Nachregelzeit in Abhängigkeit des abgeschalteten Ruhestroms

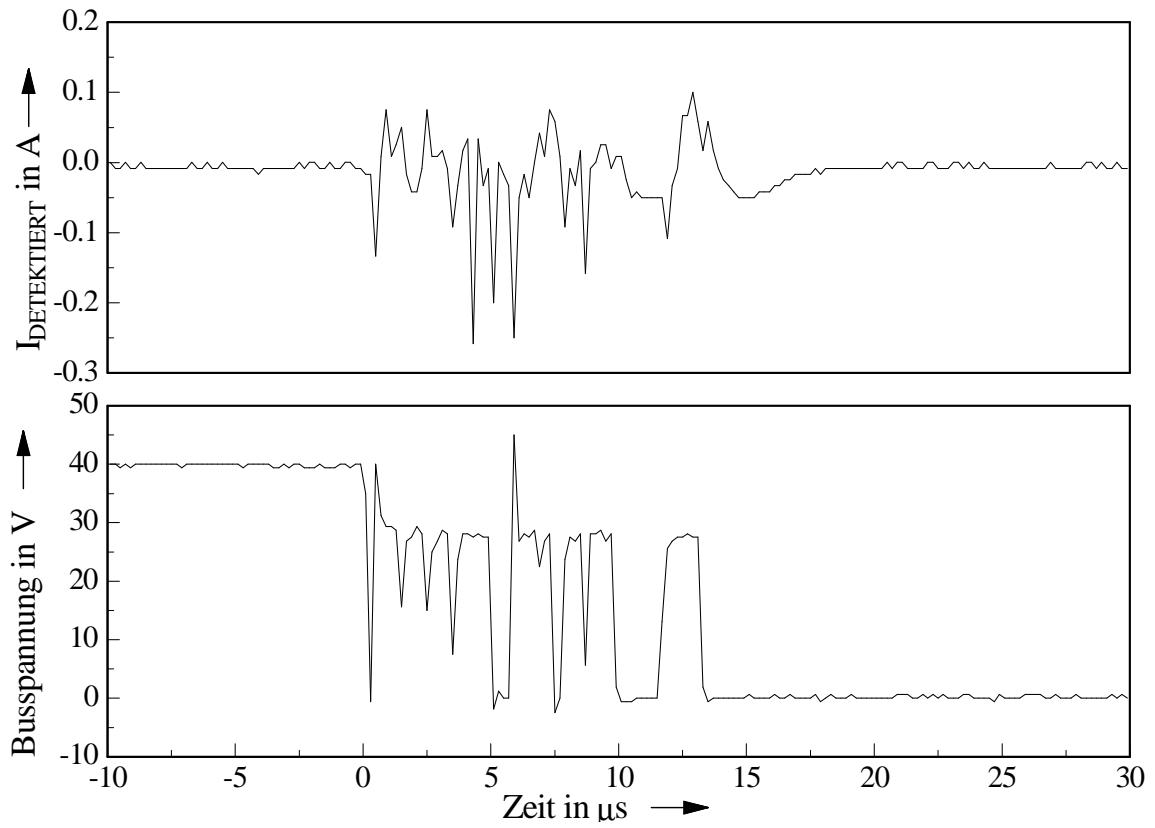


Abb. 44: Tastenprellen erzeugt Peakströme

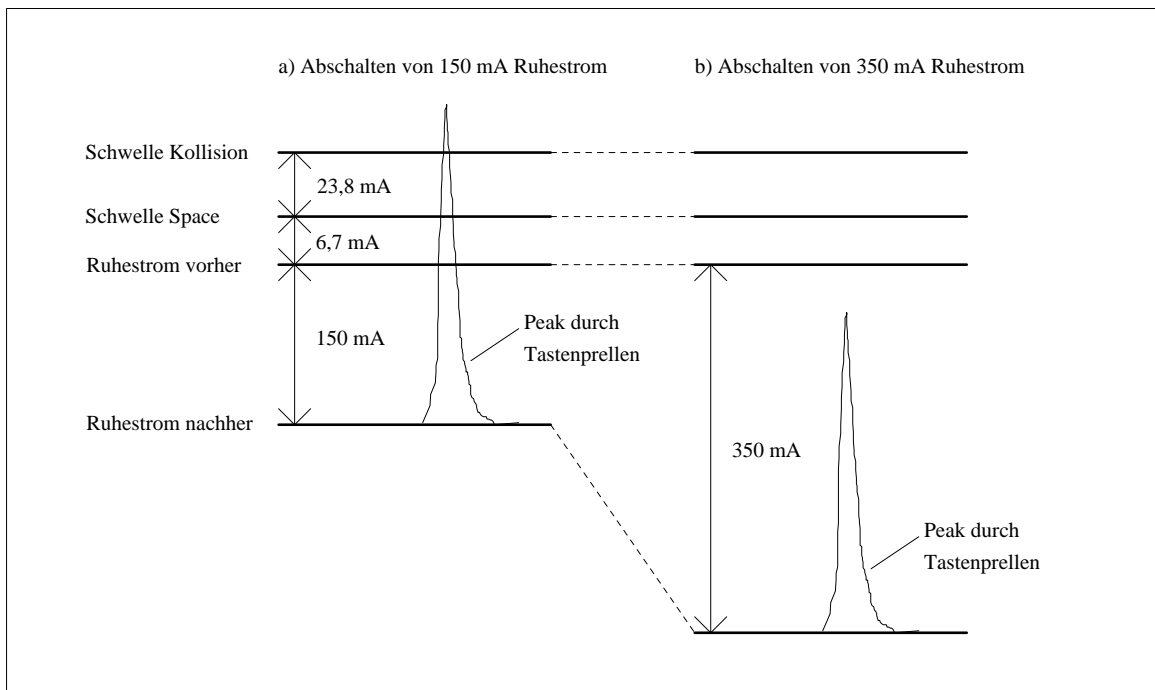
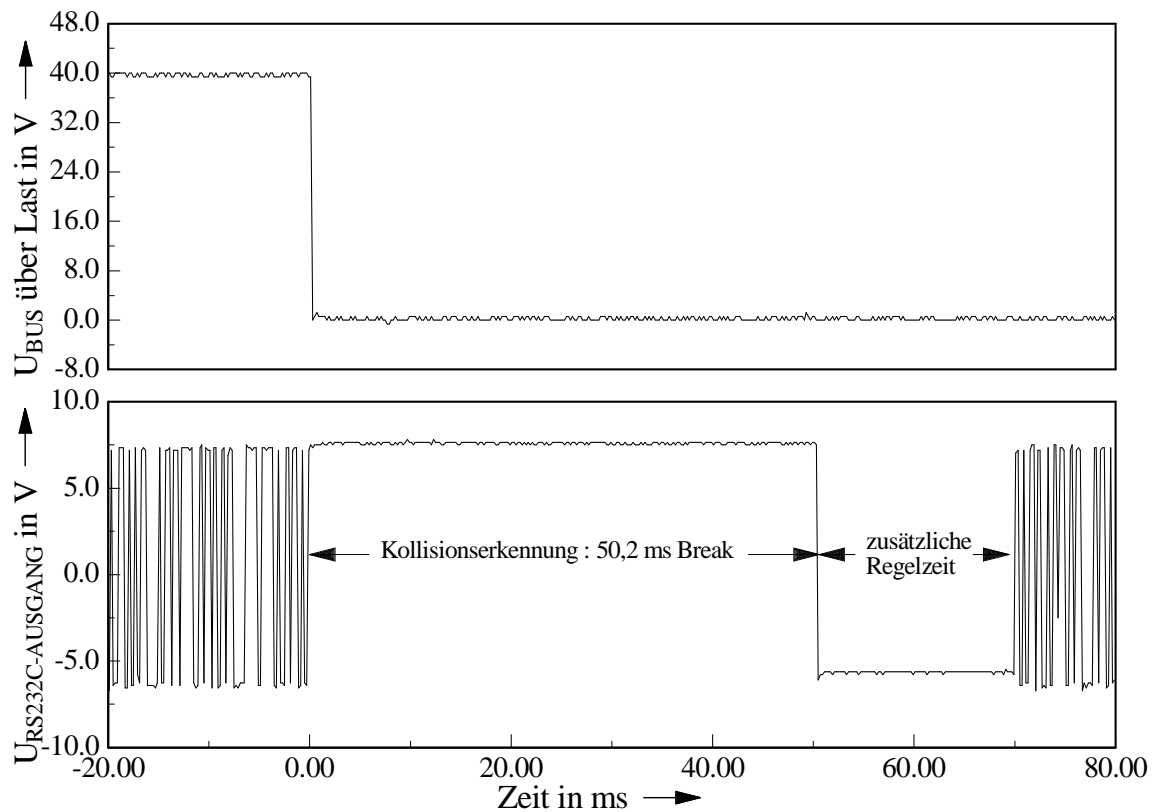


Abb. 45: Detektionsschwellen des Digital-Repeater im Bezug zu Peakströmen



**Abb. 46:** Nachregelzeit bei Abschalten eines Ruhestroms von 50 mA

Auch beim Zuschalten von Standardlasten, kann der Effekt der Kollisionserkennung beobachtet werden. Oberhalb von 30,5 mA zugeschalteten Ruhestrom (Schwelle der Kollisionserkennung) ist dies zu erwarten, unterhalb hiervon ist der Effekt des Tastenprellens dafür verantwortlich. Nach den 50,2 ms Space erfolgt die eigentliche Regelung an den erhöhten Ruhestrom. Aus Abb. 47 läßt sich entnehmen, daß die Nachregelzeit abhängig ist von der Anzahl der zugeschalteten Standardlasten. Die daran anschließende Abbildung zeigt das Beispiel einer Meßwertaufnahme bei Zuschalten eines Ruhestroms von 50 mA. Der zu Anfang beschriebene Sachverhalt läßt sich hier deutlich erkennen.



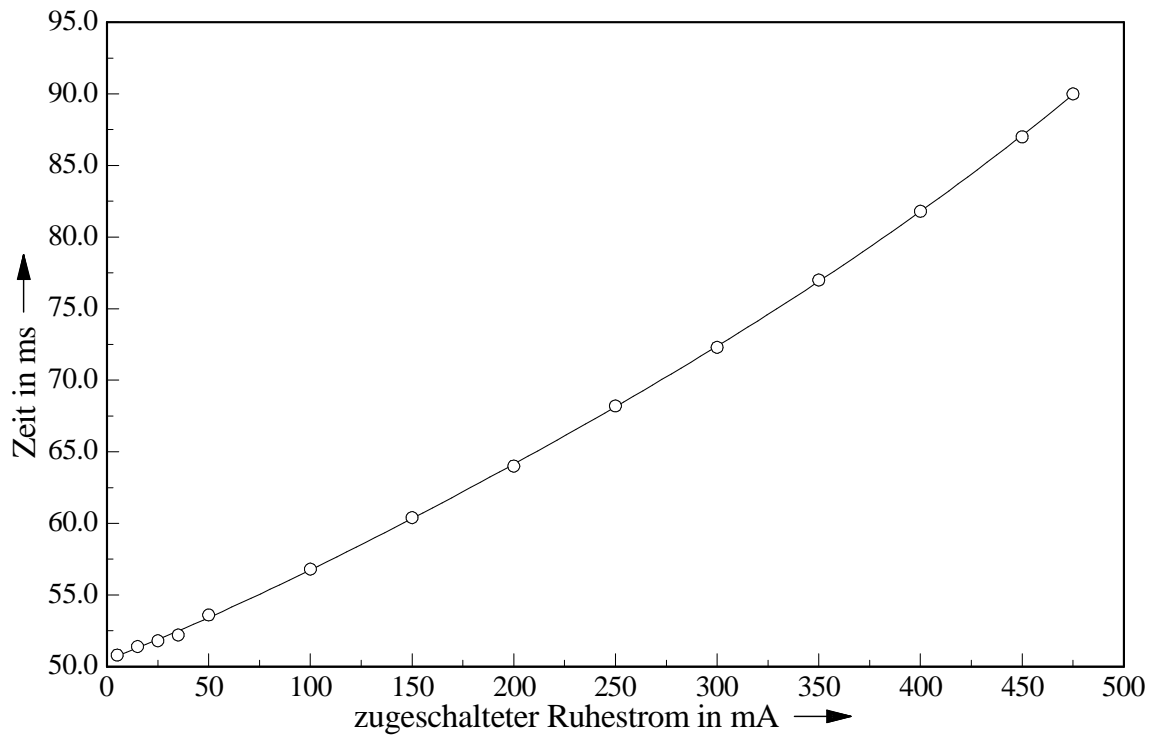


Abb. 47: Nachregelzeit in Abhängigkeit des zugeschalteter Standardlasten

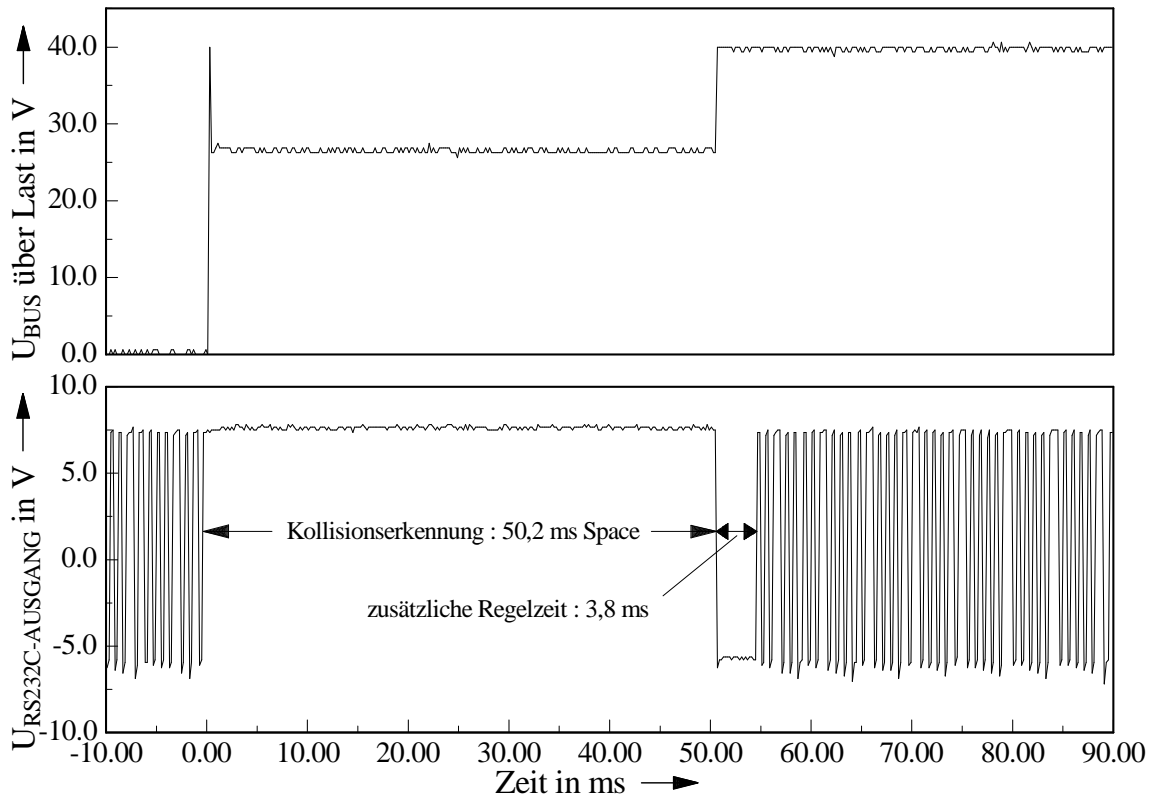


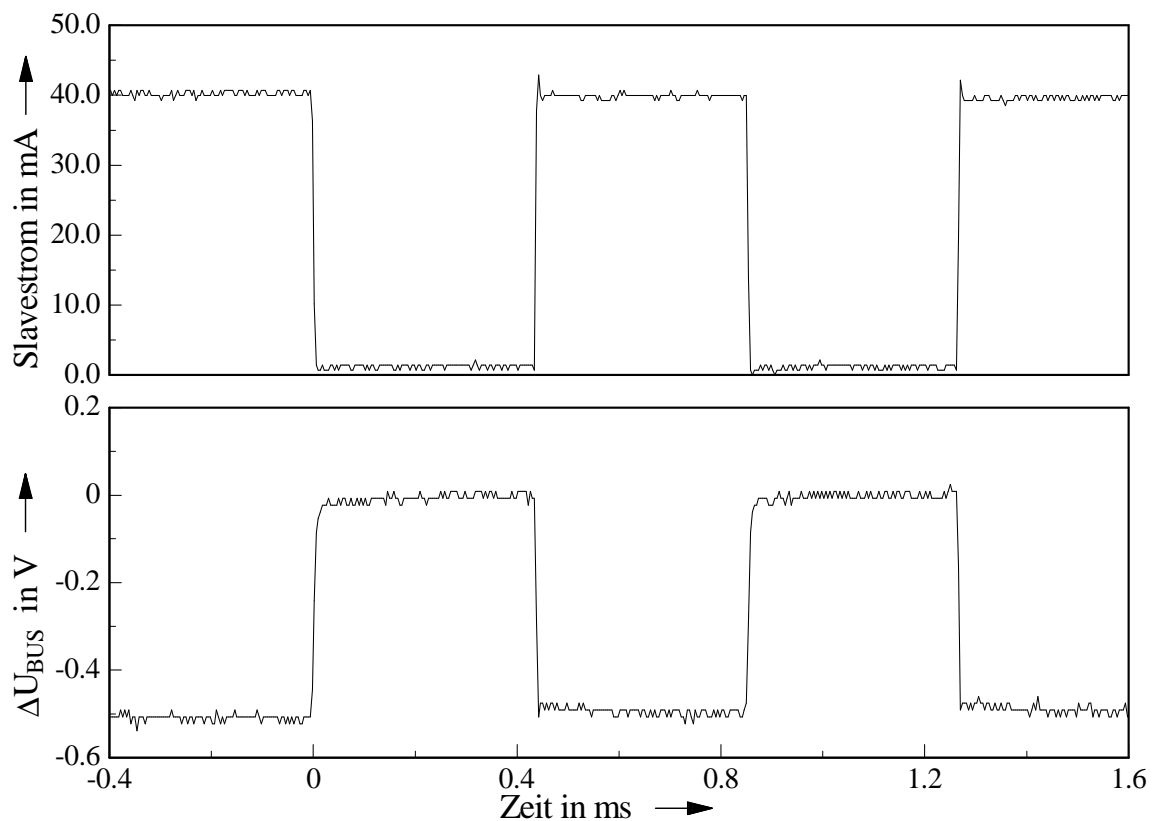
Abb. 48: Nachregelzeit bei Zuschalten eines Ruhestroms von 50mA

### Schaltswelle in Stromrichtung

Bei einer Baudrate von 2400 Baud, 1m Kabel und einer Standardlast erkennt der Repeater ab 6,7 mA sicher ein Space und unter 5,2 mA sicher nicht mehr. Bei einem Ruhestrom von 390 mA und einer Kapazität von 680 nF auf dem Bus wird ein Space ab 6,8 mA sicher erkannt, und unter 5,0 mA sicher nicht mehr erkannt. Diese Meßergebnisse erfüllen die Anforderungen.

### Abfall der Busspannung bei einem Space in Antwortrichtung

Die Messung des Abfalls der Busspannung wird bei einem Sendestrom von 38,5 mA durchgeführt. Hierzu ist es nötig die Grenze der Kollisionserkennung durch Variation eines Widerstandes nach oben zu verschieben, da der Digital-Repeater normalerweise bei 30,5 mA eine Kollision erkennt. Unter diesen neuen Voraussetzungen kann man aus Abb. 49 erkennen, daß die Busspannung um weniger als 0,5 V abfällt und somit die Spezifikationen erfüllt.



**Abb. 49:** Abfall der Busspannung bei Space in Antwortrichtung

### Übertragung bei 300 Baud

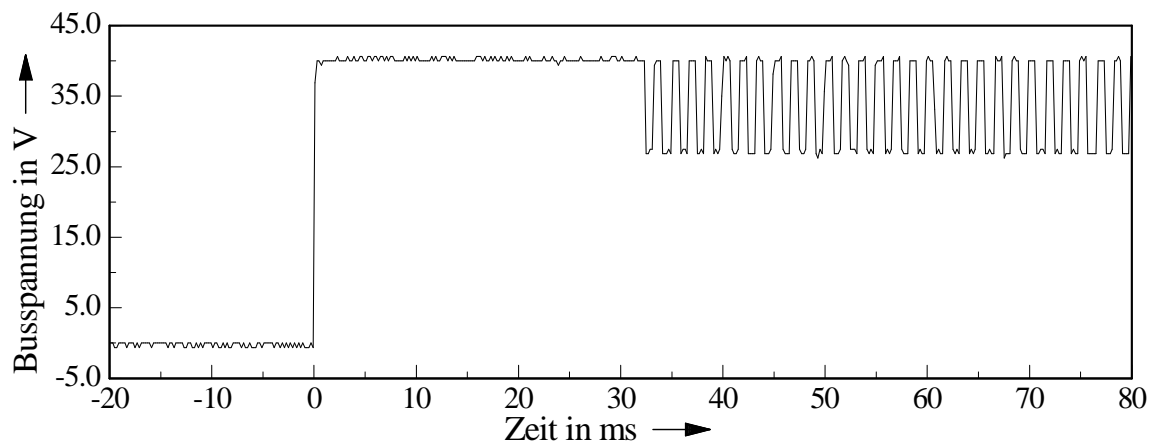
Bei der Übertragung von 2048 Zeichen mit dem Wert 0 ergeben sich keine Fehler, selbst unter der Bedingung, daß am Bus 250 Standardlasten und 1000 m Kabel angeschlossen sind.

### Überstromanzeige/Warnstromanzeige

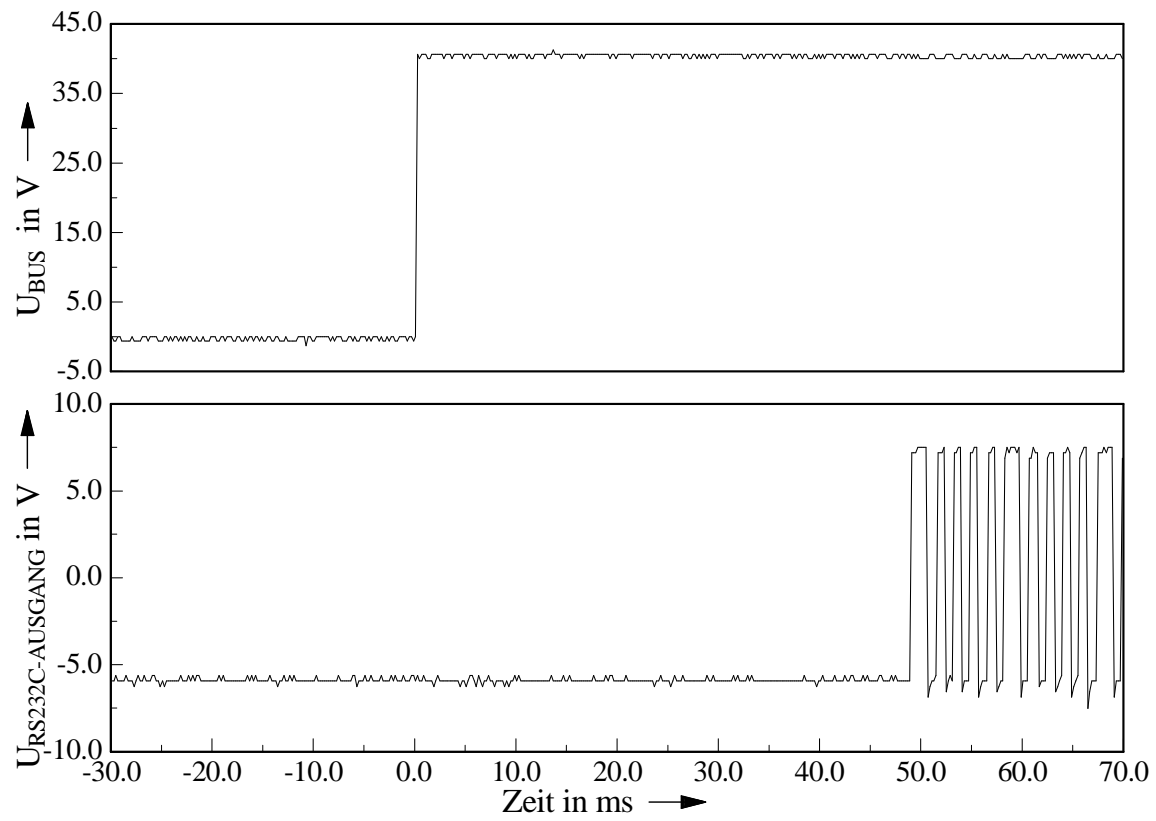
Die Warnstromanzeige (gelbe LED) wird ab 418 mA, die Überstromanzeige (rote LED) ab 498 mA aktiviert.

### Kurzschlußverhalten

Aus den folgenden beiden Abbildungen lassen sich die Erholungszeiten des Repeaters in Aufruf- (32,2 ms) und Antwortrichtung (48,8 ms) bestimmen. In beiden Fällen wurde ständig ein Signal gesendet, um die Zeit zu bestimmen, die nach der Abschaltung des Kurzschlusses (Zeitpunkt: 0 s) vergeht, bis das Signal wieder übertragen wird. Am Bus waren 1 Standardlast und 1 m Kabel angeschlossen. Der mittlere Strom auf dem Bus ist so gering, daß er mit einem Multimeter nicht gemessen werden konnte.



**Abb. 50:** Erholungszeit in Aufrufrichtung nach einem Überstromzustand



**Abb. 51:** Erholungszeit in Antwortrichtung nach einem Überstromzustand

## 5.4. Tabellarischer Vergleich der beiden Repeater

Im folgenden werden die Spezifikationen der beiden Repeater in einer Tabelle zusammengefaßt.

Eigenschaft	Analog-Repeater	Digital-Repeater
max. Anzahl Standardlasten	250	250
Baudraten	300 Bd bis 9600 Bd	300 Bd bis 9600 Bd, 19200 Bd, 38400 Bd
Kurzschlußschutz	ja	ja
Echounterdrückung	ja	ja
Kollisionserkennung	ja	ja
Repeaterfunktionen (Bitrecovery)	nein	ja
Anzeigefunktionen		
- Warnstrom	ja	ja
- Überstrom	ja	ja
- Bus an	ja	ja
- Datenfluß	nein	ja

**Tabelle 13:** *Eigenschaften des Analog- und Digital-Repeater im Vergleich*

### Bemerkungen:

9600 Bd sind nur bedingt möglich, nämlich in dem Fall, daß der Repeater auf diese Baudrate ausgelegt ist. Für die Messungen stand nur ein Repeater zur Verfügung der auf 2400 Bd eingestellt ist.

Für 38400 Bd erfüllt der Repeater alle Anforderungen, allerdings kommt es bei ungünstigen Bedingungen (250 Standardlasten, 1000 m Kabel) in der Kommunikation zwischen Master und Slaves dennoch zu Störungen. Die Ursache hierfür liegt allerdings beim Bus selbst, der ein RC-Glied darstellt und somit die Signale auf dem Bus stark verzerrt.

## 6. Zusammenfassung

In der vorliegenden Diplomarbeit wurden folgende Punkte behandelt:

1. Entwicklung einer intelligenten Ausleseeinheit auf dem M-Bus, die den Master entlastet, sich von diesem programmieren läßt, und fehlende Fähigkeiten der an ihr angeschlossenen Endgeräte simulieren kann. Die neue Komponente ermöglicht die Fernauslesung sehr großer Datenmengen, die über einen längeren Zeitraum gebildet und gespeichert worden sind, über ein Modem zu übertragen.  
Einige denkbare Erweiterungen des bestehenden Konzeptes sind, der Anschluß von mehr als einer Ausleseeinheit auf dem M-Bus, die Fehleranalyse auf Anwendungsebene, das Setzen und Löschen von Zählerständen, sowie die Konfiguration der Datenformate bei Ausgabe von sekundären Informationen an den übergeordneten Master.
2. Ein Analog- und ein Digital-Repeater wurden auf Einhaltung der in Kapitel 5.1. *Repeaterspezifikationen* angegebenen Anforderungen überprüft. Hierbei konnte festgestellt werden, daß der Analog-Repeater nur für den Betrieb bis zu 2400 Baud bei 250 Slaves geeignet ist. Ferner entstanden Regelprobleme bei Sendung von vielen Space-Bits in Antwortrichtung bei 300 Baud.  
Der Digital-Repeater hingegen erfüllt alle Spezifikationen, auch bei 38400 Baud. Da bei so hohen Baudraten die physikalischen Eigenschaften des Busses eine fehlerfreie Kommunikation nicht ermöglichen, sollte für diesen Fall die Anzahl der Geräte auf maximal 120 und die maximale Kabellänge auf 500 m eingeschränkt werden.

## 7. Anhang

### 7.1. Codierung von Datentypen

In der Norm IEC 870-5-4 sind folgende Datentypen zur Benutzung in der Anwendungsebene definiert:

**Daten-Typ A = Vorzeichenlose, ganze BCD Zahl** := XUI4 [1 bis 4] <0 bis 9 BCD>

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>		
digit 10				digit 1					
8	4	2	1	8	4	2	1		
...	...	...	...	...	...	...	...		
8	4	2	1	8	4	2	1		

1UI4 [1 bis 4] <0 bis 9 BCD> := Stelle 10<sup>0</sup>  
 2UI4 [5 bis 8] <0 bis 9 BCD> := Stelle 10<sup>1</sup>  
 ...  
 XUI4 [5 bis 8] <0 bis 9 BCD> := Stelle 10<sup>X-1</sup>

**Daten-Typ B = Ganze, binärcodierte Zahl** := I[1..X] <-2<sup>X-1</sup> bis +2<sup>X-1</sup>-1>

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>		
...	...	...	...	...	...	...	...		
S	2 <sup>X-2</sup>							2 <sup>X-8</sup>	

1B1 [X] := S=Vorzeichen: S<0> := positiv  
 S<1> := negativ

Negative Werte in Zweierkomplementdarstellung

**Daten-Typ C = Vorzeichenlose, ganze binärcodierte Zahl** := UI[1 bis X] <0 bis 2<sup>X</sup>-1>

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>		
...	...	...	...	...	...	...	...		
2 <sup>X-1</sup>								2 <sup>X-8</sup>	

UI8 [1 bis 8] <0 bis 255>

**Daten-Typ D = Boolesche Zahl (1 Bit binär dargestellter Daten):**

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>		
...	...	...	...	...	...	...	...		
2 <sup>X-1</sup>								2 <sup>X-8</sup>	

XB1: B1[i] <0 bis 1>  
 B1[i] <0> := unwahr  
 B1[i] <1> := wahr

**Daten-Typ F = Zusammengesetzter Typ CP32: Datum und Zeit**

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
2 <sup>15</sup>	2 <sup>14</sup>	2 <sup>13</sup>	2 <sup>12</sup>	2 <sup>11</sup>	2 <sup>10</sup>	2 <sup>9</sup>	2 <sup>8</sup>
2 <sup>23</sup>	2 <sup>22</sup>	2 <sup>21</sup>	2 <sup>20</sup>	2 <sup>19</sup>	2 <sup>18</sup>	2 <sup>17</sup>	2 <sup>16</sup>
2 <sup>31</sup>	2 <sup>30</sup>	2 <sup>29</sup>	2 <sup>28</sup>	2 <sup>27</sup>	2 <sup>26</sup>	2 <sup>25</sup>	2 <sup>24</sup>

Min: UI6 [1 bis 6] <0 bis 59>

Std: UI5 [9 bis 13] <0 bis 23>

Tag: UI5 [17 bis 21] <1 bis 31>

Monat: UI4 [25 bis 28] <1 bis 12>

Jahr: UI7[22 bis 24, 29 bis 32] <0 bis 99>

IV: B1[8] {Zeit ungültig}: IV<0> := gültig,  
IV>1> := ungültig

SU: B1[16] {Sommerzeit}: SU<0> := Standardzeit  
SU<1> := Sommerzeit

RES1: B1[7] {reserviert}: <0>

RES2: B1[14] {reserviert}: <0>

RES3: B1[15] {reserviert}: <0>

**Daten-Typ G: Zusammengesetzter Typ CP16: Datum**

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
2 <sup>15</sup>	2 <sup>14</sup>	2 <sup>13</sup>	2 <sup>12</sup>	2 <sup>11</sup>	2 <sup>10</sup>	2 <sup>9</sup>	2 <sup>8</sup>

Tag:

UI5 [1 bis 5] <1 bis 31>

Monat:

UI4 [9 bis 12] <1 bis 12>

Jahr:

UI7[6 bis 8, 13 bis 16] <0 bis 99>





## 7.2. Tabellen zur festen Datenstruktur

### 7.2.1. Feld Medium/Einheit

#### 7.2.1.1. Tabelle der Medien

Medium	Feld Medium/Einheit				Wert hexadecimal
	Bit 16	Bit 15	Bit 8	Bit 7	
Anderes	0	0	0	0	0
Öl	0	0	0	1	1
Elektrizität	0	0	1	0	2
Gas	0	0	1	1	3
Wärme	0	1	0	0	4
Dampf	0	1	0	1	5
Warmwasser	0	1	1	0	6
Wasser	0	1	1	1	7
Wärmemenge	1	0	0	0	8
reserviert	1	0	0	1	9
Gas Modus 2	1	0	1	0	A
Wärme Modus 2	1	0	1	1	B
Warmwasser Modus 2	1	1	0	0	C
Wasser Modus 2	1	1	0	1	D
Wärmemenge Modus 2	1	1	1	0	E
reserviert	1	1	1	1	F

#### Bemerkung:

Bei Nutzung von Medien die mit Modus 2 gekennzeichnet sind, ist darauf zu achten, daß das höherwertigste Byte der Zählerstände zuerst gesendet wird, unabhängig vom Wert des CI-Feldes.

## 7.2.1.2. Tabelle der physikalischen Einheiten

Unit	MSB..LSB	Unit	MSB..LSB
h,m,s	000000	MJ/h	100000
D,M,Y	000001	MJ/h * 10	100001
Wh	000010	MJ/h * 100	100010
Wh * 10	000011	GJ/h	100011
Wh * 100	000100	GJ/h * 10	100100
kWh	000101	GJ/h * 100	100101
kWh * 10	000110	ml	100110
kWh * 100	000111	ml * 10	100111
MWh	001000	ml * 100	101000
MWh * 10	001001	l	101001
MWh * 100	001010	l * 10	101010
kJ	001011	l * 100	101011
kJ * 10	001100	m <sup>3</sup>	101100
kJ * 100	001101	m <sup>3</sup> * 10	101101
MJ	001110	m <sup>3</sup> * 100	101110
MJ * 10	001111	ml/h	101111
MJ * 100	010000	ml/h * 10	110000
GJ	010001	ml/h * 100	110001
GJ * 10	010010	l/h	110010
GJ * 100	010011	l/h * 10	110011
W	010100	l/h * 100	110100
W * 10	010101	m <sup>3</sup> /h	110101
W * 100	010110	m <sup>3</sup> /h * 10	110110
kW	010111	m <sup>3</sup> /h * 100	110111
kW * 10	011000	°C * 10 <sup>-3</sup>	111000
kW * 100	011001	Einheiten für Wärmemenge	111001
MW	011010	reserviert	111010
MW * 10	011011	reserviert	111011
MW * 100	011100	reserviert	111100
kJ/h	011101	reserviert	111101
kJ/h * 10	011110	wie Zähler 1, aber historisch	111110
kJ/h * 100	011111	ohne Einheit	111111

### 7.2.2. Das Statusfeld

Bit Nr	Bedeutung	
	gesetztes Bit	nicht gesetztes Bit
0	Zählerstände binär mit Vorzeichen	Zählerstände BCD codiert
1	Zählerstände sind historisch	Zählerstände sind Momentanwerte
2	Spannungsausfall aufgetreten	kein Spannungsausfall
3	dauerhafter Fehler	kein dauerhafter Fehler
4	temporärer Fehler aufgetreten	kein temporärer Fehler
5	herstellerspezifisch	herstellerspezifisch
6	herstellerspezifisch	herstellerspezifisch
7	herstellerspezifisch	herstellerspezifisch

## 7.3. Tabellen zur variablen Datenstruktur

### 7.3.1. Das Statusfeld

Bit 1	Bit 0	Bedeutung
0	0	kein Fehler
0	1	Anwendung beschäftigt
1	0	Anwendungsfehler
1	1	reserviert

Bemerkung:

Die Bedeutung der restlichen Bits ist analog zur festen Datenstruktur.

### 7.3.2. Das Mediumfeld

Medium	Code hexadezimal
Anderes	00
Öl	01
Elektrizität	02
Gas	03
Wärme (Rücklauftemperatur)	04
Dampf	05
Warmwasser	06
Wasser	07
Wärmemengenzähler	08
Druckluft	09
reserviert	0A
reserviert	0B
Wärme (Vorlauftemperatur)	0C
reserviert	0D
Bus/System	0E
unbekanntes Medium	0F
reserviert	10..FF

### 7.3.3. Primäre VIF-Codes

Der erste Block der Tabelle enthält aufsummierte, der zweite durchschnittliche, der dritte augenblickliche Werte und der vierte Parameter.

Codierung	Beschreibung	Bereichscodierung	
E000 0nnn	Energie <sup>2</sup>	10 <sup>(nnn-3)</sup>	Wh
E000 1nnn	Energie <sup>2</sup>	10 <sup>(nnn)</sup>	J
E001 0nnn	Volumen <sup>2</sup>	10 <sup>(nnn-6)</sup>	m
E001 1nnn	Masse <sup>2</sup>	10 <sup>(nnn-3)</sup>	kg
E010 00nn	Laufzeit <sup>2</sup>	nn = 00	Sekunden
		nn = 01	Minuten
		nn = 10	Stunden
		nn = 11	Tage
E010 01nn	Meßzeit <sup>2</sup>	siehe Laufzeit	
E010 1nnn	Leistung <sup>4</sup>	10 <sup>(nnn-3)</sup>	W
E011 0nnn	Leistung <sup>4</sup>	10 <sup>(nnn)</sup>	J/h
E011 1nnn	Volumenfluß <sup>4</sup>	10 <sup>(nnn-6)</sup>	m <sup>3</sup> /h
E100 0nnn	Volumenfluß ext. <sup>4</sup>	10 <sup>(nnn-7)</sup>	m <sup>3</sup> /min
E100 1nnn	Volumenfluß ext. <sup>4</sup>	10 <sup>(nnn-9)</sup>	m <sup>3</sup> /s
E101 0nnn	Massenfluß <sup>4</sup>	10 <sup>(nnn-3)</sup>	kg/h
E101 10nn	Vorlauftemperatur <sup>4</sup>	10 <sup>(nn-3)</sup>	°C
E101 11nn	Rücklauftemperatur <sup>4</sup>	10 <sup>(nn-3)</sup>	°C
E110 00nn	Temperaturdifferenz <sup>4</sup>	10 <sup>(nn-3)</sup>	K
E110 01nn	Außentemperatur <sup>4</sup>	10 <sup>(nn-3)</sup>	°C
E110 10nn	Druck <sup>4</sup>	10 <sup>(nn-3)</sup>	bar
E110 110n	Zeitpunkt	n = 0	Datum
		n = 1	Datum & Zeit
E110 1110	Einheiten Wärmemengezähler <sup>4</sup>		
E110 1111	reserviert		
E111 00nn	Mittelungszeit	siehe Laufzeit	
E111 01nn	Zeit seit Auslesung	siehe Laufzeit	
E111 1000	Fabrikationsnummer		
E111 1001	Identifikation <sup>±</sup>		
E111 1010	Busadresse		
E111 1011	reserviert		

Bemerkungen:

- <sup>2</sup> Die so markierten VIF-Codes lassen die Bildung von historischen Profilen, virtuellen Extremwerten und Tarifinformationen zu
- ⌈ Mit den so markierten VIF-Codes lassen sich durch die Ausleseinheit Extremwerte und historische Profile bilden
- ± Wird dieser VIF-Code zusammen mit dem Datenfeld 8 Stellen BCD benutzt, so ist die Identifikationsnr., bei 64 Bit Integer hingegen die gesamte Sekundäradresse gemeint.

**7.3.4. Sekundäre VIF-Codes**

Kennzeichen dieser Codes ist das vorangehende Zeichen FDh. Es ergibt sich hiermit die Möglichkeit 128 weitere Codes zu bilden.

VIF-Code (binär)	Beschreibung	Gruppe
E000 00nn	Haben in 10 <sup>nn-3</sup> der örtlichen legalen Währungseinheit ⌈	Währungseinheit
E000 01nn	Soll in 10 <sup>nn-3</sup> der örtlichen legalen Währungseinheit ⌈	
E000 1000	Anzahl Zugriffe	Erweiterte Identifikation
E000 1001	Medium (wie im festen Datenblock)	
E000 1010	Hersteller (wie im festen Datenblock)	
E000 1011	reserviert	
E000 1100	Modell/Version	
E000 1101	Hardwareversionsnummer	
E000 1110	Firmwareversionsnummer	
E000 1111	Softwareversionsnummer	
E001 0000	Abnehmerort	Implementation aller TC 292 WG1 Anforderungen (verbesserte Selektion)
E001 0001	Abnehmer	
E001 0010	Zugriffscod Benutzer	
E001 0011	Zugriffscod Operator	
E001 0100	Zugriffscod Systemoperator	
E001 0101	Zugriffscod Entwickler	
E001 0110	Paßwort	
E001 0111	Fehlerflags (binär)	
E001 1000	Fehlermaske	
E001 1001	reserviert	

VIF-Code (binär)	Beschreibung	Gruppe	
E001 1xxx	reserviert		
E010 0000	Erste Speichernummer für zyklische Speicherung	Erweitertes Speicher- management	
E010 0001	Letzte Speichernummer für zyklische Speicherung		
E010 0010	Größe des Speicherblocks		
E010 0011	reserviert		
E010 01nn	Speicherintervall [Sek. bis Tag(e)]		
E010 1000	Speicherintervall Monat(e)		
E010 1001	Speicherintervall Jahr(e)		
E010 1010	reserviert		
E010 1011	reserviert		
E010 11nn	Zeit seit letzter Auslesung [Sek. bis Tag(e)]		
E011 0000	Start (Datum/Zeit) des Tarifes		Erweitertes Tarifmanagement
E011 00nn	Tarifdauer [Min. bis Tag(e)]		
E011 01nn	Tarifperiode [Sek. bis Tag(e)]		
E011 1000	Tarifperiode Monat(e)		
E011 1001	Tarifperiode Jahr(e)		
E011 1010	Dimensionslos / kein VIF		
E011 1011	reserviert		
E011 11xx	reserviert		
E100 nnnn	$10^{nnn-9}$ Volts <sup>⌈</sup>	Elektrische Einheiten	
E101 nnnn	$10^{nnn-12}$ Ampere <sup>⌈</sup>		
E110 xxxx	reserviert		
E111 xxxx	reserviert		

Bemerkungen:

nn = 00: Sekunde(n) / 01: Minute(n) / 10: Stunde(n) / 11: Tag(e)

Ob der Datentyp F für Datum und Uhrzeit oder der Datentyp G nur für das Datum verwendet wird läßt sich aus dem Datenfeld schließen. (0010b für Typ G oder 0100b für Typ F)

nn = 01: Minute(n) / 10: Stunde(n) / 11: Tag(e)

⌈ Zu diesen VIF-Codes lassen sich historische Profile und Extremwerte einrichten



### 7.3.5. VIFE-Codes

#### 7.3.5.1. Erweiterung der Bedeutung des VIF

VIFE-Code (binär)	Beschreibung
E00x xxxx	reserviert für Aktionen und Fehlercodes
E010 0000	pro Sekunde
E010 0001	pro Minute
E010 0010	pro Stunde
E010 0011	pro Tag
E010 0100	pro Woche
E010 0101	pro Monat
E010 0110	pro Jahr
E010 0111	pro Umdrehung / Meßung
E010 100p	Erhöhung pro Eingangsimpulse am Eingangskanal Nummer p
E010 101p	Erhöhung pro Ausgangsimpuls am Ausgangskanal Nummer p
E010 1100	pro l (Liter)
E010 1101	pro m <sup>3</sup>
E010 1110	pro kg
E010 1111	pro K (Kelvin)
E011 0000	pro kWh
E011 0001	pro GJ
E011 0010	pro kW
E011 0011	pro (K * l) (Kelvin * Liter)
E011 0100	pro V (Volt)
E011 0101	pro A (Ampere)
E011 0110	multipliziert mit Sekunden
E011 0111	multipliziert mit Sekunden / V
E011 1000	multipliziert mit Sekunden / A
E011 1001	Startzeitpunkt (Uhrzeit und/oder Datum) <sup>▲</sup>
E011 1010	VIF beinhaltet unkorrigierte Einheit
E011 1011	Berechnung nur bei positiven Beitrag
E011 1100	Berechnung der absoluten Werte nur bei negativen Beitrag
E011 1101	reserviert
E011 1110	reserviert

E011 1111	reserviert
<b>VIFE-Code (binär)</b>	<b>Beschreibung</b>
E100 u000	u = 1: oberes, u = 0: unteres Limit des Wertes
E100 u001	Anzahl der Überschreitungen des unteren (u = 0) / oberen (u=1) Limits <sup>♣</sup>
E100 uf1b	Datum (/Zeit) vom Anfang (b = 0)/Ende (b = 1) der ersten (f = 0)/letzten (f = 1) Überschreitung des unteren (u = 0)/oberen (u = 1) Limits <sup>♣</sup>
E101 ufnn	Dauer der Limitüberschreitung [Sek. bis Tage] (u,f: wie oben) <sup>♣</sup>
E110 0fnn	Dauer von [Sek. bis Tage] (f: wie oben) <sup>♣</sup>
E110 1x1x	reserviert
E110 1f1b	Datum (/Zeit) von <sup>♣</sup>
E111 0nnn	Multiplikativer Korrekturfaktor: $10^{nnn-6}$
E111 10nn	Additive Korrekturkonstante: $10^{nn-3} * \text{Einheit des VIF}$
E111 11xx	reserviert

Bemerkungen:

nn = 00: Sekunde(n) / 01: Minute(n) / 10: Stunde(n) / 11: Tag(e)

Ob der Datentyp F für Datum und Uhrzeit oder der Datentyp G nur für das Datum verwendet wird läßt sich aus dem Datenfeld schließen. (0010b für Typ G oder 0100b für Typ F)

"Datum (/Uhrzeit)" bzw. "Dauer von" beziehen sich auf die Information, die durch den DRH spezifiziert wird

- ♣ Die so markierten VIFE-Codes nehmen die Rolle des VIF im Bezug auf die Bedeutung des übertragenen Datums ein.

## 7.3.5.2. Recordspezifische Fehler

VIFE-Code	Art des Fehlers	Fehler Gruppe
E000 0000	Kein Fehler	DIB-Fehler
E000 0001	Zu viel DIFE's	
E000 0010	Speichernummer nicht implementiert	
E000 0011	Unitnummer nicht implementiert	
E000 0100	Tarifnummer nicht implementiert	
E000 0101	Funktion nicht implementiert	
E000 0110	Datenart nicht implementiert	
E000 0111	Datengröße nicht implementiert	
E000 1000 bis E000 1010	reserviert	
E000 1011	Zu viel VIFE's	
E000 1100	Illegale VIF-Gruppe	
E000 1101	Illegaler VIF-Exponent	
E000 1110	VIF paßt nicht zum DIF	
E000 1111	Aktion nicht implementiert	
E001 0000 bis E001 0100	reserviert	
E001 0101	Keine Daten erhältlich (undefinierter Wert)	Datenfehler
E001 0110	Datenüberlauf	
E001 0111	Datenunterlauf	
E001 1000	Datenfehler	
E001 1001 bis E001 1011	reserviert	
E001 1100	vorzeitiges Ende des Records	Andere Fehler
E001 1101 bis E001 1111	reserviert	



## Literaturverzeichnis

- [1] Papenheim, Andreas: Dokumentation "The M-Bus: A Documentation Version 4.2", Universität-GH Paderborn, Fachbereich Physik, 1994
- [2] Papenheim, Andreas: Diplomarbeit "Anwendungsbeispiele für den M-Bus", Universität-GH Paderborn, Fachbereich Physik, 1992
- [3] prEN 1434-3: Wärmehähler, Teil 3: Datenaustausch und Schnittstellen, 1994
- [4] DIN 19244, Teil 52 (IEC 57-5-2): Fernwirkeinrichtungen und Fernwirksysteme, Übertragungsverfahren der Verbindungsschicht, 1990