

SIEMENS

Universelles serielles Schnittstellen-Protokoll USS[®]-Protokoll

Spezifikation

Verfasser:

Walter Möller-Nehring, Siemens AG, ASI 1 D SP, Erlangen
Wolfgang Bohrer, Siemens AG, ASI 1 D SP, Erlangen

Weitergabe sowie Verfielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiederhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten insbesondere für den Fall der Patenterteilung oder GM Eintragung.

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so daß wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden jedoch regelmäßig überprüft und notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten. Für Verbesserungsvorschläge sind wir dankbar.

USS® ist ein Warenzeichen von Siemens.
SIMOREG® ist ein Warenzeichen von Siemens.
SIMOVERT® ist ein Warenzeichen von Siemens.

Inhalt

Kapitel A: Protokollspezifikation

1.	Einleitung.....	1
2.	Telegrammverkehr.....	2
2.1.	Zyklischer Telegrammverkehr	2
2.2.	Azyklischer Telegrammverkehr	2
3.	Broadcast	2
4.	Telegrammaufbau.....	3
4.1.	Datencodierung	3
4.2.	Telegrammlänge (LGE).....	3
4.2.1.	Variable Telegrammlänge	3
4.2.2.	Feste Telegrammlänge	4
4.3.	Belegung des Adreßbytes (ADR)	4
4.4.	BCC-Bildungsgesetz.....	5
5.	Übertragungsprozedur	6
5.1.	Abwicklung des Datenaustausches	6
5.1.1.	Zykluszeit	7
5.1.2.	Startpause	8
5.2.	Überwachungsmechanismen und Fehlerreaktionen	8
5.2.1.	Zeitüberwachung	9
5.2.1.1.	Antwortverzugszeit	9
5.2.1.2.	Telegramm-Restlaufzeit.....	9
5.2.2.	Weiterverarbeitung von empfangenen Telegrammen	9
5.2.3.	Diagnose-Hilfsmittel zur Überprüfung des Empfangs.....	10
5.3.	Spiegeltelegramm	11
5.4.	Sondertelegramm	11
6.	Definitionen.....	12
6.1.	Zeichenlaufzeit.....	12
6.2.	Bündige Telegramm-Restlaufzeit	12
6.3.	Maximale Telegramm-Restlaufzeit	12

Kapitel B: Physikalische Schnittstelle und Busaufbau

1.	Topologie.....	1
2.	Übertragungstechnik.....	2
2.1.	Leitungseigenschaften	2
2.2.	Leitungslängen	3
2.3.	Schnittstelleneigenschaften.....	3
2.4.	Übertragungsraten	4
3.	Übertragungsverfahren	5
3.1.	Bitcodierung.....	5
3.2.	Zeichenrahmen	5
4.	Aufbaurichtlinien	6
4.1.	Leitungsführung	6
4.2.	Potentialausgleich.....	6
4.3.	Schirmung	6
4.4.	Anschlußtechnik, Steckerbelegungen.....	7
4.5.	Busabschluß.....	8
4.6.	Schaltungsvorschlag.....	10

Kapitel C: Festlegung der Nutzdaten für die Anwendung in der Antriebstechnik

1. Einleitung..... 1

2. Allgemeiner Aufbau des Nutzdatenblocks 2

3. Parametrierung des USS®-Protokolls an einer seriellen Schnittstelle..... 4

3.1. Parametereinstellung für Geräte SIMOVERT P 6SE21, 6SE30 (MircoMaster) und SIMOREG K 6RA244

3.2. Parametereinstellung für SIMOVERT Master Drives 7

4. PKW-Bereich..... 10

4.1. Aufbau des PKW-Bereiches (Parameter-Kennung-Wert) 10

4.1.1. PKW-Bereich bei fester Telegrammlänge..... 10

4.1.2. PKW-Bereich mit variabler Telegrammlänge 11

4.2. Beschreibung der einzelnen PKW-Elemente 11

4.2.1. Parameterkennung (PKE) 11

4.2.1.1. Auftrags- und Antwortkennung..... 11

4.2.1.2. Spontanmeldung..... 16

4.2.1.3. Parameternummer (PNU) 18

4.2.2. Index (IND) 18

4.2.3. Parameterwert (PWE)..... 23

5. PZD-Bereich..... 25

5.1. Aufbau des PZD-Bereichs..... 25

5.2. Beschreibung der einzelnen PZD-Elemente 26

5.2.1. Das Steuerwort und Zustandswort 26

5.2.2. Soll- / Istwerte..... 32

5.2.3. Broadcast-Mechanismus..... 32

6. Übertragungsformat der Nutzdaten 33

7. Projektierung des Protokolls am Bussystem..... 36

8. Beispiele..... 38

8.1. Feste Telegrammlänge 38

8.2. Variable Telegrammlänge 40

Anhang

Übersicht: Aufbau des Telegramms beim USS®-Protokoll 1

The Optional Broadcast Mechanism of the USS-Protocol..... 2

Änderungsstand

Ausgabe	Bestellnummer	Datum	Stand
Erstausgabe Kapitel A	E31930 - T9011 - X - A1	Januar '92	veröffentlicht
Erstausgabe englisch Kapitel A	E31930 - T9011 - X - A1-7600	January '92	veröffentlicht
Erstausgabe Applikation	E31930-T9012-X-A1	April '92	veröffentlicht
Fehlerkorrektur zur Erstausgabe Applikation	E31930-T9012-X-A2	Oktober '92	veröffentlicht
Funktionserweiterung für SIMOVERT Master Drives und generelle Überarbeitung. Nicht als Applikation veröffentlicht.	E31930-T9012-X-AXX	August '93	vorläufig
Zusammenfassung der Dokumente E31930-T9011 und E31930-T9012	E20125-D001-S302-A1	September '94	Neue Ausgabe

Kurzbeschreibung der Änderungen:

August 1993:

- Spezifikation für SIMOVERT Master Drives.
- Aufheben der Beschränkung auf feste Telegrammlänge bei Auftragstelegrammen vom Master zum Slave. Es wird bei Parametrierung auf variable Telegrammlänge, diese in beiden Richtungen (Master → Slave → Master) zugelassen.
- Erweiterung im Index-Wort (IND) im High-Byte für Textübertragung bei SIMOVERT FC/VC/SC
- Änderung der Reihenfolge der Textzeichen bei der Übertragung über den Bus.

September 1994:

- Überarbeitung von Kapitel C: Punkt 3: Parametrierung
- Einbringen von Kapitel C: Punkt 6: Übertragungsformate der Nutzdaten
- Ergänzung des Kapitel B: Physikalische Schnittstelle und Busaufbau
- Master kann an beliebiger Stelle der Buslinie angeordnet sein.
- Vorschlag für Anzeigeparameter zur BUS-und Schnittstellendiagnose.
- Belegung des höchstwertigen Bit im Adreßbyte für Sondertelegramme
- Broadcastdefinition Kapitel C
- Bit 11-15 im Index gerätespezifisch definert
- Kapitel B: Bezeichnungen (A) und (B) durch RS485P und RS485N ersetzt

Hinweis

Diese Spezifikation erhebt nicht den Anspruch, alle Gerätedetails oder -varianten zu erfassen oder jeden denkbaren Fall des Betriebes oder der Anwendung zu berücksichtigen.

Sollten Sie weitere Informationen benötigen oder sollten spezielle Probleme auftreten, die für das Anwendungsgebiet nicht ausführlich genug behandelt werden, wenden Sie sich bitte an die örtliche Siemens-Niederlassung.

Der Inhalt dieser Spezifikation wird nicht Bestandteil einer früheren oder bestehenden Vereinbarung, Zusage oder eines Rechtsverhältnisses und ändert diese auch nicht ab. Der jeweilige Kaufvertrag stellt die gesamte Verpflichtung des Geschäftsgebiet ASI 1 Antriebstechnik der Siemens AG dar. Die in dem Vertrag zwischen den Parteien festgelegte Gewährleistung ist die einzige vom Geschäftsbereich ASI 1 Antriebstechnik übernommene Gewährleistung. Die vertraglichen Gewährleistungsbestimmungen werden durch die Ausführung dieser Spezifikation weder erweitert noch abgeändert.

Literaturverzeichnis

- /1/: PROFIBUS-Profil: Drehzahlveränderbare Antriebe
VDI / VDE-Richtlinie 3689
Entwurf, Januar 1993
- /2/: RS485 Recommended Standard
EIA 485: STANDARD FOR ELECTRICAL CHARACTERISTICS
OF GENERATOR AND RECEIVER FOR USE IN
BALANCED DIGITAL MULTIPOINT SYSTEMS
EIA Standard April 1983

Files: Alle Dokumente sind mit Winword 2.0b und Designer 3.1 PLUS OLE verfaßt.

Deckblatt.doc	Titelseite ohne Bild
Ussbild.drw	Designer 3.1 Zeichnung für Titelseite
Vorspann.doc	Dieses Dokument: Copyright, Inhaltsverzeichnis, Änderungsstand, Literatur
KapitelA.doc	Kapitel A: Protokollspezifikation
KapitelB.doc	Kapitel B: Physik
KapitelC.doc	Kapitel C: Festlegung der Nutzdaten
Anhang.doc	Anhang
Ruecken.doc	Letzte Seite
Tabellen.doc	3 Tabellen im Querformat für Kapitel C
Broadcast.drw	Designer 3.1 Zeichnung für Anhang Thema Broadcast

A: Protokollspezifikation

1. Einleitung

Das USS[®]-Protokoll (Universelles-serielles-Schnittstellen-Protokoll) definiert ein Zugriffsverfahren nach dem Master-Slave-Prinzip für die Kommunikation über einen seriellen Bus. Als Untermenge ist darin auch die Punkt-zu-Punkt-Verbindung eingeschlossen

Wesentliche Merkmale des USS[®]-Protokolls sind:

- Unterstützung einer mehrpunktfähigen Kopplung, z. B. EIA RS 485-Hardware
- Master-Slave-Zugriffsverfahren
- Single Master-System
- Maximal 32 Teilnehmer (max. 31 Slaves)
- Einfacher, sicherer Telegrammrahmen
- Einfach implementierbar
- Wahlweiser Betrieb mit variablen oder festen Telegrammlängen.

Am Bus können ein Master und max. 31 Slaves angeschlossen werden. Die einzelnen Slaves werden vom Master über ein Adreßzeichen im Telegramm angewählt. Ein Slave kann niemals von sich aus die Sendeinitiative ergreifen, ein direkter Nachrichtenaustausch zwischen den einzelnen Slaves ist nicht möglich. Die Kommunikation erfolgt im Halbduplex-Betrieb.

Die Masterfunktion kann nicht weitergegeben werden (Single-Master-System).

Das folgende Diagramm zeigt eine Buskonfiguration am Beispiel der Antriebstechnik.

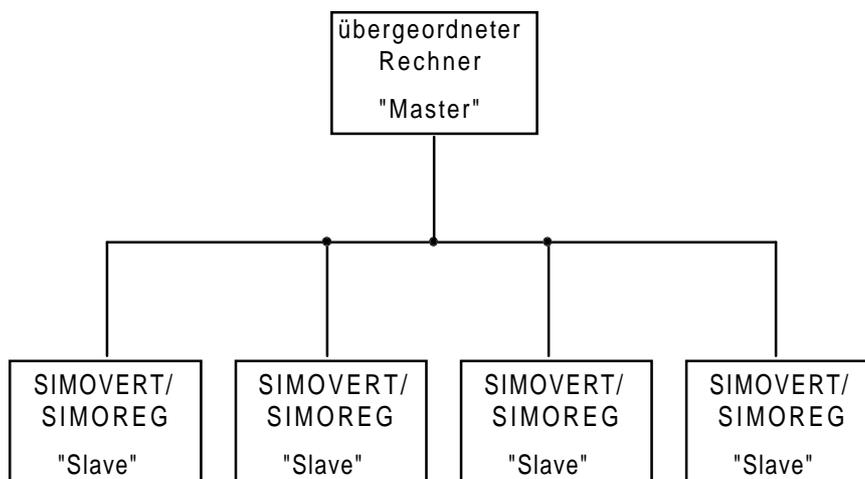


Bild 1.1: *Serielle Kopplung von SIMOREG- / SIMOVERT-Geräten (Slaves) mit einem übergeordneten Rechner als Master.*

Die elektrischen und mechanischen Schnittstelleneigenschaften (Hardware) sind nicht Gegenstand der Protokollspezifikation. Die Festlegungen und Empfehlungen zur Übertragungstechnik, dem Übertragungsverfahren und dem Busaufbau werden im Kapitel B: 'Physikalische Schnittstelle und Busaufbau' dieser Spezifikation beschrieben.

2. Telegrammverkehr

Generell kann zwischen einem zyklischen und einem azyklischen Telegrammverkehr unterschieden werden. **In der Antriebstechnik wird ausschließlich der zyklische Telegrammverkehr eingesetzt.** Die Masterstation hat den zyklischen Telegrammverkehr sicherzustellen, wobei nacheinander in gleichbleibenden Zeitabständen alle Slaveteilnehmer angesprochen werden.

2.1. Zyklischer Telegrammverkehr

Die Antriebstechnik fordert für die zu bewältigenden Steuerungs- und Regelungsaufgaben definierte Reaktionszeiten und damit einen streng zyklischen Telegrammverkehr:

Der Master sendet kontinuierlich Telegramme (Auftragstelegramme) an die Slaves und erwartet jeweils ein Antworttelegramm vom angesprochenen Slave.

Ein Slave muß ein Antworttelegramm senden, wenn

- er ein Telegramm fehlerfrei empfangen hat und
- er in diesem Telegramm adressiert wurde.

Ein Slave darf nicht senden, wenn diese Bedingungen nicht erfüllt sind oder der Slave im Broadcast angesprochen wurde.

Für den Master besteht die Verbindung zu dem betreffenden Slave dann, wenn er nach einer definierten Bearbeitungszeit (Antwortverzugszeit) vom Slave ein Antworttelegramm erhält.

Im zyklischen Telegrammverkehr muß von den Slave-Teilnehmern der Telegrammverkehr auf Ausfall überwacht werden.

Für Service- und Diagnoseaufgaben ist ein Teil der Nutzdaten, die innerhalb des zyklischen Telegramms enthalten sind, vorgesehen. Der Nutzdatenverkehr für die Datentechnik ist im Kapitel C: 'Festlegung der Nutzdaten für die Anwendung in der Antriebstechnik' beschrieben.

2.2. Azyklischer Telegrammverkehr

Generell wird der Telegrammverkehr zyklisch abgewickelt.

Zyklischer und azyklischer Telegrammverkehr schließen sich gegenseitig aus.

Service- und Diagnose-Aufgaben können auch im azyklischen Betrieb gefahren werden.

Im azyklischen Betrieb sendet der Master in unregelmäßigen Abständen Telegramme an die Slaves. Der Slave antwortet entsprechend den für den zyklischen Betrieb festgelegten Bedingungen.

Beim azyklischen Telegrammverkehr kann vom Slave keine Telegrammausfall-Überwachung durchgeführt werden.

3. Broadcast

Im Broadcast-Modus sendet der Master ein Telegramm an alle am Bus befindlichen Slaves. Dabei ist im Auftragstelegramm das "Broadcast-Bit" im Adreßbyte auf log. 1 gesetzt (siehe Abschnitt 4.3 Belegung des Adreßbyte (ADR)). Die Adreßbits sind unwirksam. Auf ein Broadcast-Telegramm darf von den einzelnen Slaves kein Antworttelegramm gesendet werden.

Die Anwendung des Broadcast-Telegramms setzt weitere Definitionen auf Anwendungsebene (gemeinsame Telegrammlänge, Zuordnung der Nutzdateninhalte zu Teilnehmern, etc..) voraus. Zur Festlegung der Nutzdateninhalte bei Broadcast siehe in Kapitel C, Punkt 5: Prozeßdaten, Unterpunkt Broadcast.

4. Telegrammaufbau

Jedes Telegramm (Bild 4.1) beginnt mit dem Startzeichen STX (= 02 Hex), gefolgt von der Längenangabe (LGE) und dem Adreßbyte (ADR). Die Nutzzeichen folgen anschließend. Abgeschlossen wird das Telegramm durch das Datensicherungszeichen BCC (Block Check Character).

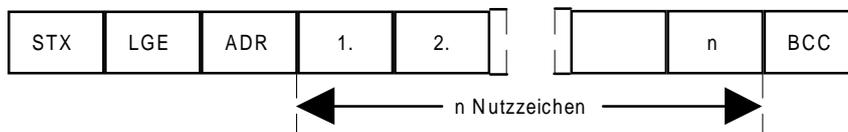


Bild 4.1: Telegrammaufbau

Bei Wortinformationen (16 Bit) im Nutzdatenblock (= Nutzzeichenblock) wird stets zuerst das High-Byte (erstes Zeichen) und dann das Low-Byte (zweites Zeichen) gesendet. Entsprechendes gilt bei Doppelwortinformationen: Zuerst wird das High-Word gesendet, dann folgt das Low-Word.

Die Kennzeichnung von Aufträgen in den Nutzzeichen ist nicht Bestandteil des Protokolls. Der Inhalt der Nutzdaten für die Geräte der Antriebstechnik wird im Kapitel C behandelt.

4.1. Datencodierung

Die Informationen sind wie folgt codiert:

- STX (Start of Text): ASCII-Zeichen: 02 Hex
- LGE (Telegrammlänge): 1 Byte, enthält die Telegrammlänge als Binärzahl, siehe Abschnitt 4.2
- ADR (Adreßbyte): 1 Byte, enthält die Slave-Adresse und den Telegrammtyp. Binär codiert, siehe Abschnitt 4.3
- Nutzzeichen: Je ein Byte, Inhalt auftragsabhängig
- BCC: 1 Byte, Datensicherungszeichen (Block Check Charakter), Bildungsgesetz, siehe Abschnitt 4.4

4.2. Telegrammlänge (LGE)

Die Telegrammlänge ist variabel.

Die Länge eines Telegramms wird im 2. Byte des Telegramms angegeben.

Projektierungsabhängig kann auch eine feste Telegrammlänge vereinbart werden.

Bei fester Telegrammlänge können für jeden Slave-Teilnehmer an einem Bus unterschiedliche Telegrammlängen verwendet werden.

Die maximale Gesamtlänge eines Telegramms ist 256 Byte.

Die tatsächliche Länge des Gesamttelegramms ist um zwei Zeichen länger als LGE, weil die ersten beiden Zeichen (STX und LGE) nicht mitgerechnet werden.

Zur Telegrammlänge zählen nur die Nutzzeichen (Anzahl n), das Adreßbyte (ADR) und das Datensicherungszeichen (BCC). Somit ergibt sich für die Telegrammlänge:

$$\mathbf{LGE = n + 2.} \quad \{ 1 \leq LGE \leq 254 \}$$

Es können pro Telegramm maximal $n = 252$ Nutzzeichen (252 Nutzdatenbytes) übertragen werden.

4.2.1. Variable Telegrammlänge

Bei Telegrammen mit variabler Länge ist die Anzahl der Nutzzeichen vom gestellten Auftrag (Master → Slave) abhängig.

4.2.2. Feste Telegrammlänge

Bei einem Telegrammverkehr mit vorher vereinbarter fester Länge ist die Anzahl der Nutzzeichen innerhalb eines Telegramms fest, z. B. 6 Worte-Telegramm, d. h. 12 Nutzzeichen.

Die Einschränkung des Protokolls auf feste Länge muß in der Anwenderebene bei der Projektierung des Bussystems erfolgen. (siehe hierzu auch Kapitel C)

Für die Slaves an einem Bus können jeweils unterschiedliche Telegrammlängen vereinbart werden.

4.3. Belegung des Adreßbytes (ADR)

Im Adreßbyte werden zusätzlich zur Teilnehmernummer weitere Informationen verschlüsselt: Die einzelnen Bits im Adreßbyte sind wie dargestellt belegt.

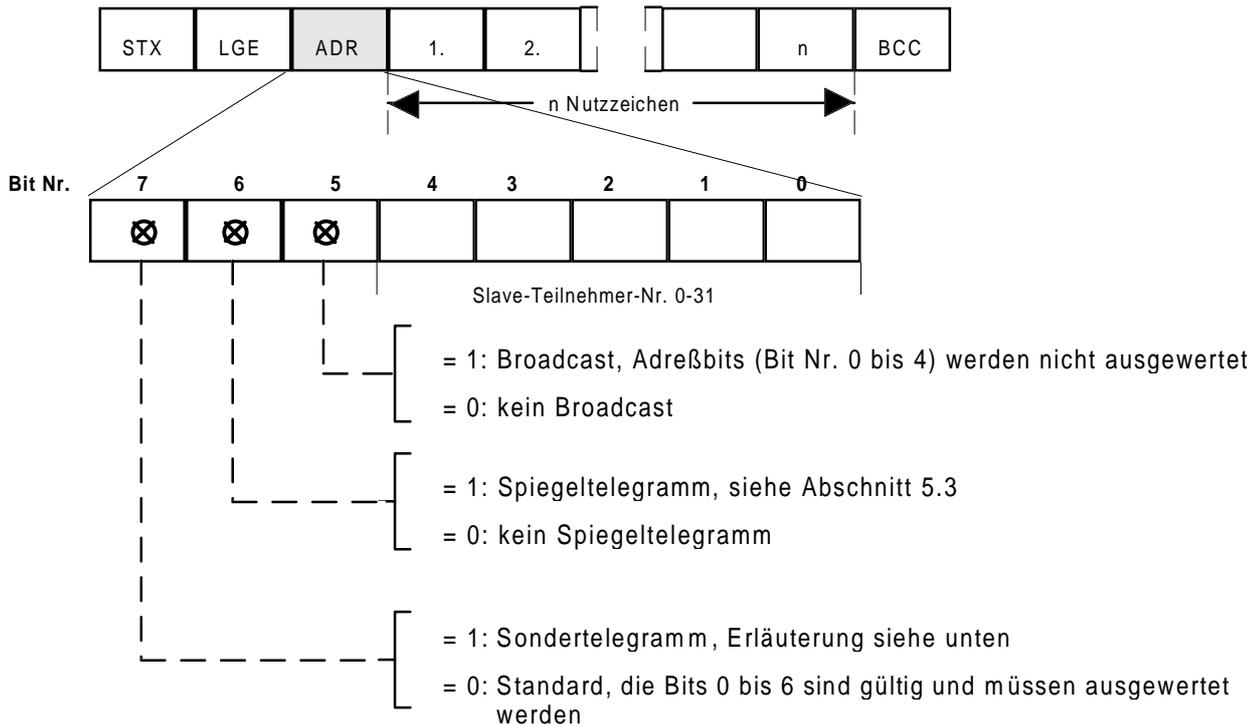


Bild 4.2: Belegung des Adreßbytes (ADR)

Bit 7	Bit 6	Bit 5	Bedeutung
0	0	0	Normaler Datenaustausch für Geräte. Die Teilnehmernummer (Bit 0 bis Bit 4 wird ausgewertet)
0	1	0	Spiegeltelegramm: Die Teilnehmernummer wird ausgewertet und der adressierte Slave sendet das Telegramm unverändert an den Master zurück (siehe 5.3 Spiegeltelegramm)
0	0	1	Broadcast: Die Teilnehmernummer wird nicht ausgewertet. (siehe unter 3. Broadcast)
1	x	x	Sondertelegramm: Das Telegramm wird von allen Slaves, für die keine Sondertelegramme definiert sind, verworfen. Das Telegramm darf nicht ausgewertet werden (siehe 5.4 Sondertelegramm)

Tabelle 4.1: Wahrheitstabelle der möglichen Kombinationen der Bits 5, 6 und 7 im Adreßbyte (ADR)

Nicht definierte Kombinationen dürfen von Mastern nicht gesendet und bei Slaves zu keiner Reaktion führen

4.4. BCC-Bildungsgesetz

Das BCC-Bildungsgesetz wird im folgenden anhand eines Beispiels erklärt:

Vor Empfang des ersten Zeichens eines Telegramms (STX) ist $BCC = 0$.

BCC 0 0 0 0 0 0 0 0

Nach Empfang des ersten Zeichens: $BCC_{neu} = BCC_{alt} \text{ EXOR "erstes Zeichen"}$
(EXOR = exklusiv-ODER-Verknüpfung)

BCC_{alt} = 0 0 0 0 0 0 0 0
EXOR
"erstes Zeichen" = 0 0 0 0 0 0 1 0 ($\underline{\text{STX}}$)

BCC_{neu} = 0 0 0 0 0 0 1 0

Nach Empfang jedes weiteren Zeichens wird dieses mit BCC_{alt} EXOR verknüpft, um BCC_{neu} zu bilden, z. B.:

BCC_{alt} = 0 0 0 0 0 0 1 0
EXOR
"zweites Zeichen" = 1 1 0 1 0 1 1 0

BCC_{neu} = 1 1 0 1 0 1 0 0

Das Ergebnis nach dem letzten Nutzzeichen ist BCC.

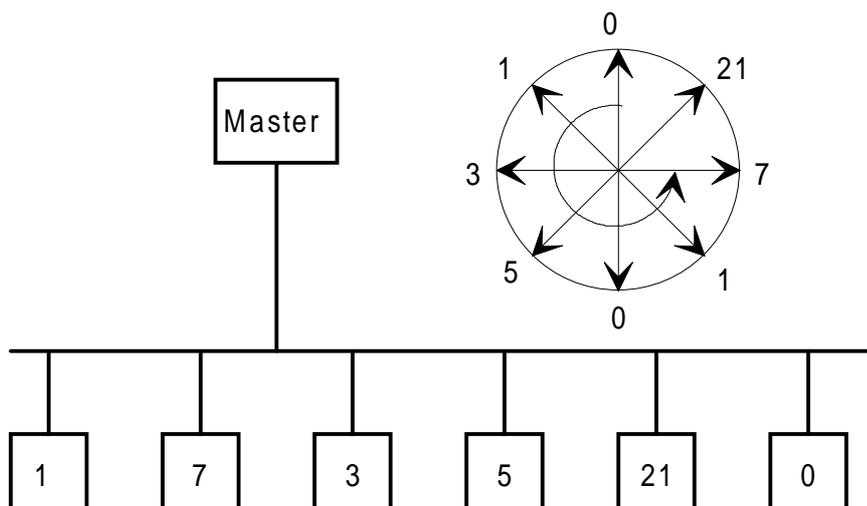
5. Übertragungsprozedur

Der zyklische Telegrammverkehr wird durch den Master sichergestellt. Der Master spricht nacheinander alle Slaveteilnehmer mit einem Auftragstelegramm an. Die angesprochenen Teilnehmer senden jeweils ein Antworttelegramm zurück. Gemäß der Master-Slave-Prozedur muß der Slave nach Empfang des für ihn bestimmten Auftragstelegramms das Antworttelegramm an den Master senden, bevor der Master den nächsten Slaveteilnehmer anspricht.

5.1. Abwicklung des Datenaustausches

Die Reihenfolge der angesprochenen Slaveteilnehmer kann z.B. durch Eintrag der Teilnehmernummern (ADR) in einer Umlaufliste angegeben werden. Müssen einige Slaves in einem schnelleren Zyklus als andere angesprochen werden, so kann deren Teilnehmernummer mehrmals in der Umlaufliste vorkommen. Über die Umlaufliste kann auch eine **Punkt-zu-Punkt Verbindung** realisiert werden, in diesem Fall ist nur ein Teilnehmer in der Umlaufliste eingetragen.

Beispiel für eine Konfiguration



Beispiel für die Umlaufliste

0
1
3
5
0
1
7
21

Die Teilnehmer 0 und 1 werden doppelt so oft angesprochen, wie die anderen.

Bild 5.1: Umlaufliste

5.1.1. Zykluszeit

Der Betrag einer Zykluszeit entsteht durch die zeitliche Aufeinanderfolge des Datenaustausches mit den einzelnen Teilnehmern.

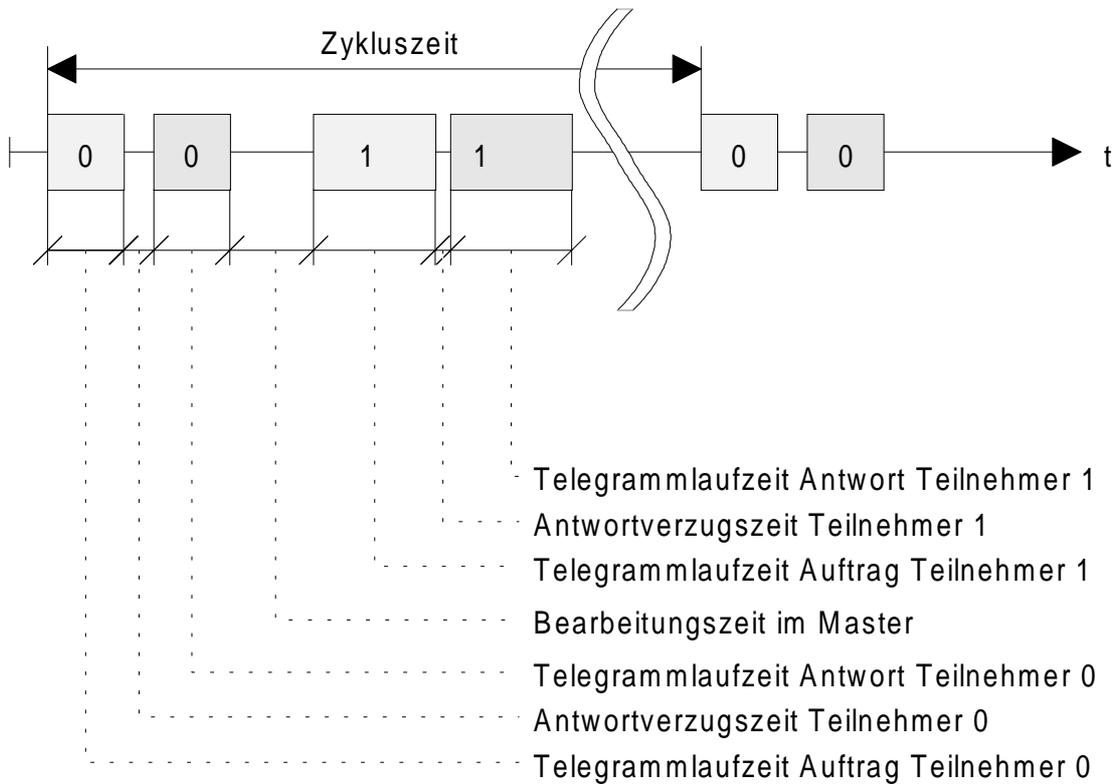


Bild 5.2: Zykluszeit

Auf Grund von nicht konstanten Antwortverzugs- und Bearbeitungszeiten ist die Zykluszeit nicht determiniert. Ein Master kann eine feste Zykluszeit realisieren, in dem eine maximale Zykluszeit für eine Konfiguration ermittelt und als absolute Zykluszeit definiert wird. Nach dem Datenaustausch mit dem letzten Teilnehmer muß er warten, bis die definierte Zykluszeit abgelaufen ist und ein neuer Umlauf beginnt.

5.1.2. Startpause

Das Startzeichen STX (= 02 Hex) allein reicht den Slaves nicht aus, um den Beginn eines Telegramms eindeutig zu erkennen, weil die Bitkombination 02/Hex auch in den Nutzzeichen vorkommen kann. Daher ist vor dem STX eine zeichenlose **Startpause** von mindestens **2 Zeichenlaufzeiten** (siehe Abschnitt 6) für den Master vorgeschrieben. Die Startpause ist Bestandteil des Auftragstelegramms.

Erst ein STX mit vorangegangener Startpause kennzeichnet einen gültigen Telegrammbeginn.

Baudrate in bit/s	Startpause in ms
9600	2,30 ms
19200	1,15 ms
38400	0,58 ms
187500	0,12 ms

Tabelle 5.1: Wert der minimalen Startpause bei verschiedenen Baudraten

Der Datenaustausch verläuft immer nach dem im Bild 5.3 dargestellten Schema (Halbduplexbetrieb):

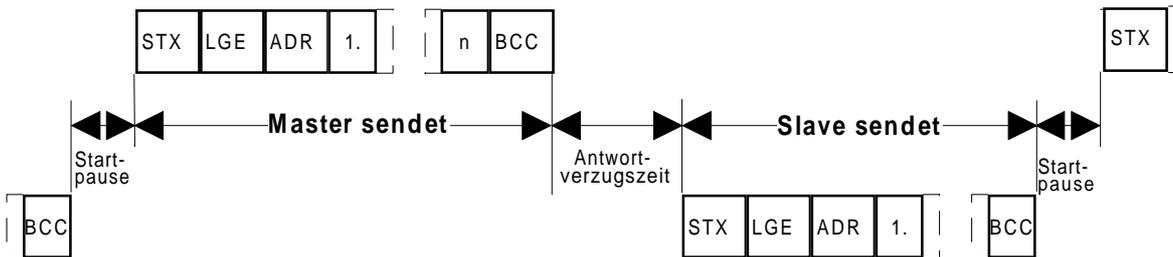


Bild 5.3: Sendefolge

5.2. Überwachungsmechanismen und Fehlerreaktionen

Beim Empfang eines Telegramms muß zuerst der richtige Start eines Telegramms (Startpause + STX) erkannt und anschließend die Länge (LGE) ausgewertet werden. Entspricht die Angabe der Länge nicht dem eingestellten Wert bei fester Telegrammlänge oder hat sie keinen gültigen Wert bei variabler Telegrammlänge, wird das Telegramm verworfen.

Vor und während des Empfangs von Telegrammen sind Zeiten zu überwachen (siehe unten).

Während des Empfangs wird das *block check character* (BCC) gebildet und nach dem Einlesen des gesamten Telegramms mit dem empfangenen BCC verglichen. Stimmen diese nicht überein, wird das Telegramm nicht ausgewertet.

Ist in keinem empfangenen Zeichen ein Zeichenrahmenfehler oder ein Parity-Fehler aufgetreten, so kann die Teilnehmernummer (ADR) des empfangenen Telegramms ausgewertet werden.

Entspricht das Adreßbyte (ADR) nicht der Teilnehmernummer (beim Slave) oder der erwarteten Slave-Teilnehmernummer (beim Master), so wird das Telegramm verworfen. (Zur Auswertung von ADR siehe 4.3)

5.2.1. Zeitüberwachung

Der **Master** muß folgende Zeiten überwachen:

- Antwortverzugszeit (Bearbeitungszeit des Slaves)
- Restlaufzeit des Antworttelegramms (siehe Abschnitt 6)

Der **Slave** überwacht folgende Zeiten:

- Startpause
- Restlaufzeit des Auftragstelegramms (siehe Abschnitt 6)

5.2.1.1. Antwortverzugszeit

Das Zeitintervall zwischen dem letzten Zeichen des Auftragstelegramms (BCC) und dem Beginn des Antworttelegramms (STX) heißt **Antwortverzugszeit** (Bild 5.3). Die maximal zulässige Antwortverzugszeit beträgt **20 ms, darf jedoch nicht kleiner als die Startpause sein**. Antwortet der Teilnehmer x nicht innerhalb der maximal zulässigen Antwortverzugszeit, wird im Master die Fehlermeldung "Teilnehmer x sendet nicht" hinterlegt.

Der Master sendet dann das für den nächsten Slave-Teilnehmer vorgesehene Telegramm.

Die Fehlermeldung "Teilnehmer x sendet nicht" wird erst nach einem fehlerfreien Telegramm vom Teilnehmer x gelöscht. Der Teilnehmer x wird nicht aus der Umlaufliste gelöscht.

5.2.1.2. Telegramm-Restlaufzeit

Die Überwachung der Telegramm-Restlaufzeit ist abhängig von der vereinbarten Telegrammlänge.

- Variable Telegrammlänge
Die Überwachung der max. Telegramm-Restlaufzeit wird zunächst mit dem Empfang vom STX mit dem Wert geladen, der sich bei einem Telegramm mit 252 Nutzdaten ergibt. Nach Empfang von LGE (Abschnitt 4.2), also dem nächsten Zeichen, wird diese Überwachungszeit mit dem korrekten Wert geladen.
- Feste Telegrammlänge
Die Überwachung der max. Telegramm-Restlaufzeit kann direkt mit dem korrekten Wert gestartet werden (keine Auswertung von LGE notwendig).

5.2.2. Weiterverarbeitung von empfangenen Telegrammen

Nur fehlerfrei empfangene und richtig adressierte Telegramme werden weiterverarbeitet. Folgende Empfangsfehler werden erkannt (gilt gleichermaßen für Master und Slave):

- Parity-Fehler
- Zeichenrahmenfehler
- Telegrammlänge (LGE) falsch
- BCC falsch
- Telegramm-Restlaufzeit überschritten
- Verbindung unterbrochen:
 - Slave: Keine Überwachung der Masteraktivitäten auf Protokollebene, die Anwenderebene kann eine parametrierbare Überwachung (Busüberwachungszeit) vorsehen.
 - Master: Slave antwortet nicht innerhalb der maximal zulässigen Antwortverzugszeit .

Auf fehlerhaft empfangene Telegramme wird vom Slave kein Antworttelegramm gesendet.

5.2.3. Diagnose-Hilfsmittel zur Überprüfung des Empfangs

Für die Diagnose der Kommunikationsschnittstellen können von der Schnittstellensoftware Informationen über den Status der Kommunikation bereitgestellt werden.

Die Diagnose-Informationen sollten, bei Slaves, auf dem geräteeigenen Bedienfeld sichtbar gemacht werden können (Diagnose-Parameter USS[®]-Schnittstelle).

Folgende Diagnose-Informationen werden vorgeschlagen:

In einem indizierten Diagnose-Parameterfeld sollte das empfangene Telegramm Zeichen für Zeichen eingesehen werden können.

Weitere, jeweils 16 Bit breite Diagnose-Parameter:

1. Hardwareausführungsstand der Schnittstelle
2. Softwarestand der Schnittstellensoftware
3. Reserve
4. Ein 16 Bit breites Wort für Fehlerstatus
 - Bit 15 bis Bit 8: reserviert für spätere Verwendung
 - Bit 7: Master: Antwortverzugszeit abgelaufen
Slave: Auf Anwendererebene eingestellte Überwachungszeit für zyklische Telegramme abgelaufen
 - Bit 6: Falsche Telegrammlänge (LGE)
 - Bit 5: Telegrammrestlaufzeit abgelaufen
 - Bit 4: Falscher *block check character* (BCC)
 - Bit 3: Telegrammstart nicht erkannt (erstes Zeichen kein STX)
 - Bit 2: Parity-Fehler
 - Bit 1: Puffer-Überlauf
 - Bit 0: Zeichenrahmenfehler
5. Anzahl der fehlerfrei empfangenen Telegramme je Minute
6. Anzahl der verworfenen Telegramme je Minute
7. Zähler: Fehlerfreie Telegramme
8. Zähler: Verworfenen Telegramme
9. Zähler: Zeichenrahmenfehler
10. Zähler: Überlauffehler
11. Zähler: Parity-Fehler.
12. Zähler: Telegrammstart nicht erkannt
13. Zähler: Telegrammrestlaufzeit abgelaufen
14. Zähler: BCC-Fehler
15. Zähler: Falsche Telegrammlänge
16. Zähler: Überwachungszeit abgelaufen

Weitere Zähler und Felder können vorgesehen werden.

5.3. Spiegeltelegramm

Der Master hat die Möglichkeit, vom Slave ein sogenanntes Spiegeltelegramm anzufordern.

Der Ablauf ist dann wie folgt:

Der Master sendet ein Telegramm an den entsprechenden Slave-Teilnehmer. Dieses Telegramm unterscheidet sich vom normalen Telegramm nur dadurch, daß das Bit Nr. 6 des Adreßbytes gesetzt (= log. 1) ist. Der Slave sendet (spiegelt) dieses Telegramm ohne irgendwelche Veränderungen unmittelbar danach an den Master zurück. Hierbei gelten die Bedingungen von Abschnitt 5.1.

Durch das Spiegeltelegramm kann die Datenübertragung zwischen Master und Slave geprüft werden. Dies ist bei einer schrittweisen Inbetriebsetzung bzw. eine Fehlereingrenzung von Vorteil.

5.4. Sondertelegramm

Für Sonderanwendungen, die einen vom Geräteprofil abweichenden Nutzdatenaufbau erfordern (siehe Kapitel C), kann das USS[®]-Protokoll ebenfalls eingesetzt werden.

Um einen Mischbetrieb von Anwendungen mit Geräteprofil und Sonderanwendungen an einer Buslinie mit einem Master zu ermöglichen, müssen diese Sondertelegramme eindeutig gekennzeichnet sein, um von allen Slaves mit Geräteprofil verworfen werden zu können.

Der Aufbau des Telegrammrahmens der Sondertelegramme entspricht vollständig dem Aufbau aller anderen Telegramme (STX, LGE, ADR Nutzdaten, BCC, Maximale Nutzdatenlänge = 252 Bytes).

Die Adressierung der Teilnehmer mit Sondertelegrammbehandlung erfolgt wie unter 4.3 (Belegung des Adreßbytes (ADR)) definiert. Eine Adresse (Bit 0 bis 4) darf nur einmal am Bus vorkommen. D.h. ein Teilnehmer mit Sondertelegrammbehandlung kann sowohl mit normalen Telegrammen nach Geräteprofil (Bit 7=0), als auch als Sonderanwendung (Bit 7=1) angesprochen werden.

Ein Broadcast, der nur an Teilnehmer mit Sondertelegrammbehandlung gerichtet ist, ist mit dem Broadcast-Bit (Bit 5) möglich.

Teilnehmer am Bus, die Sondertelegramme behandeln können (Bit 7=1), müssen nicht auf normale Telegramme (Bit 7=0) antworten können.

Spiegeltelegramm ist nur mit Bit 7=0 möglich.

Bit 7	Bit 6	Bit 5	Bedeutung
1	0	0	Sondertelegramm: Das Telegramm wird von dem mit Bit 0 bis Bit 4 adressierten Teilnehmer (Teilnehmer mit Sondertelegrammbearbeitung) empfangen und beantwortet.
1	0	1	Sondertelegramm mit Broadcast: Die Teilnehmernummer wird nicht ausgewertet. Alle Teilnehmer mit Sondertelegrammbearbeitung empfangen das Telegramm und senden kein Antworttelegramm an den Master zurück
0	x	x	siehe 4.3

Tabelle 5.2: Wahrheitstabelle der möglichen Kombinationen der Bits 5, 6 und 7 im Adreßbyte (ADR) bei Sondertelegrammen (Bit 7 = log. 1)

Nicht definierte Kombinationen dürfen zu keiner Reaktion führen

6. Definitionen

6.1. Zeichenlaufzeit

Die Zeichenlaufzeit ist die Zeit, die benötigt wird, um ein Zeichen (Zeichenrahmen 11 Bit) zu übertragen. Diese Zeit ist abhängig von der Baudrate.

$$tz = \frac{11 \cdot 1000}{\text{Übertragungsrate}} [\text{ms}]$$

6.2. Bündige Telegramm-Restlaufzeit

Die bündige Telegramm-Restlaufzeit ist definiert als die Laufzeit, die benötigt wird, um LGE, ADR, die n Nutzzeichen und BCC bündig aufeinanderfolgend zu senden (d. h. auf das Stopbit des einen Zeichens folgt unverzögert das Startbit des nächsten Zeichens).

Die bündige Telegramm-Rücklaufzeit beträgt somit:

$$(n + 3) * \text{Zeichenlaufzeit.}$$

6.3. Maximale Telegramm-Restlaufzeit

Die maximale Telegramm-Restlaufzeit beinhaltet, wie in Bild 6.1 dargestellt, zusätzlich zur bündigen Telegramm-Restlaufzeit Zeichenverzugszeiten. Die Summe der Zeichenverzugszeiten beträgt 50 % der bündigen Telegramm-Restlaufzeit.

Die maximale Telegramm-Restlaufzeit beträgt somit:

$$1,5 * \text{bündige Telegramm-Restlaufzeit}$$

entsprechend:

$$1,5 * (n + 3) * \text{Zeichenlaufzeit}$$

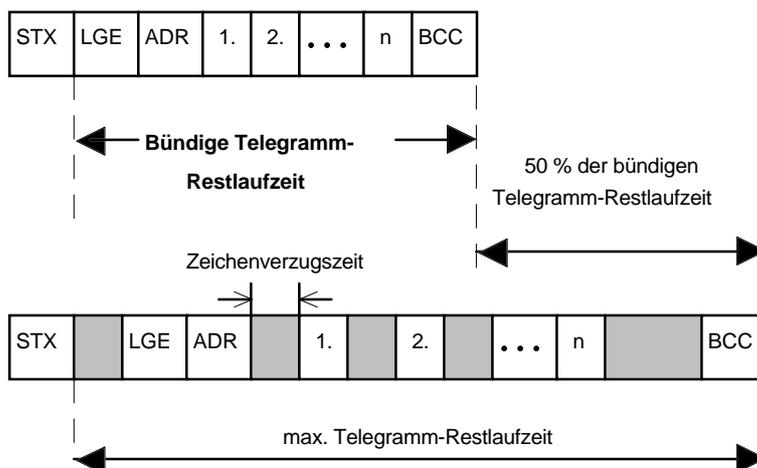


Bild 6.1: Telegramm-Restlaufzeit

Die Verzugszeit zwischen zwei Zeichen (= Zeichenverzugszeit) muß kleiner als die Startpause sein und kann beliebig zwischen den Zeichen verteilt auftreten. Eine Überwachung der Zeichenverzugszeit muß nicht erfolgen.

B: Physikalische Schnittstelle und Busaufbau

Das Übertragungsmedium und die physikalische Busschnittstelle werden wesentlich durch den Anwendungsbereich des Bussystems bestimmt.

Grundsätzlich ist es möglich, für das USS[®]-Protokoll verschiedene physikalische Schnittstellen einzusetzen. Im Anwendungsfall ist jedoch bei der Auswahl der physikalischen Schnittstelle die erforderliche Übertragungssicherheit zu beachten.

Grundlage für die physikalische Schnittstelle des USS[®]-Protokolls ist der "Recommended Standard RS-485 nach /2/. Bei Punkt-zu-Punkt-Verbindungen kann auch eine Untermenge von EIA RS-232 (CCITT V.24), TTY (20 mA Stromschleife) oder Lichtwellenleiter als physikalische Schnittstelle verwendet werden.

In diesem Kapitel wird beschrieben, wie ein USS[®]-Feldbus aufgebaut werden muß, um in Standardanwendungen einen sicheren Datentransport über das Übertragungsmedium zu gewährleisten. Unter besonderen Einsatzbedingungen müssen zusätzliche Einflüsse berücksichtigt werden, die weitere Maßnahmen oder Einschränkungen, die nicht in dieser Spezifikation beschrieben sind, erfordern.

1. Topologie

Der USS[®]-Bus basiert auf einer Linientopologie ohne Stichleitungen.

Beide Enden der Linie enden an einem Teilnehmer.

Die maximale Leitungslänge und damit der maximale Abstand zwischen Master und dem letzten Slave ist durch die Leitungseigenschaften, die Umgebungsbedingungen und die Übertragungsrate begrenzt. Bei einer Übertragungsrate < 100 kbit/s ist eine maximale Länge von 1200 m möglich. [EIA Standard RS-422-A Dezember 1978, Appendix, Page 14]

Die Teilnehmerzahl ist auf maximal 32 Teilnehmer (1 Master, 31 Slaves) beschränkt.

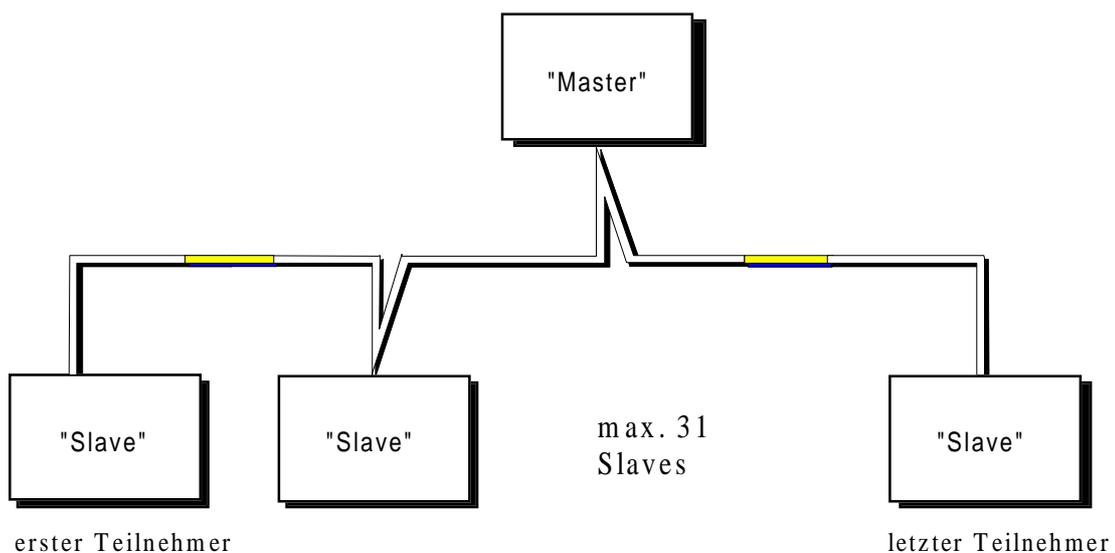


Bild 1.1: Topologie USS[®]-Bus

Punkt-zu-Punkt-Verbindungen werden wie Busverbindungen behandelt. Ein Teilnehmer hat die Masterfunktion, der andere hat die Slavefunktion.

2. Übertragungstechnik

Die Datenübertragung erfolgt nach dem Standard EIA 485 (/2/).

Für Punkt-zu-Punkt-Kopplungen kann auch RS232 oder TTY eingesetzt werden.

Die Übertragung ist grundsätzlich halbduplex, d.h. Senden und Empfangen erfolgen im Wechsel und müssen von der Software gesteuert werden. Das Halbduplexverfahren erlaubt die Verwendung der gleichen Leitungen für beide Übertragungsrichtungen. Dies ermöglicht eine einfache und kostengünstige Busverkabelung, Betrieb in gestörter Umgebung und eine hohe Datenübertragungsrate.

2.1. Leitungseigenschaften

Zur Busverkabelung wird eine geschirmte, verdrehte Zweidrahtleitung verwendet.

Anmerkung:

Alle Angaben sind nur Empfehlungen. Je nach den Erfordernissen und Gegebenheiten des spezifischen Einsatzes und den Bedingungen auf der Anlage können Abweichungen notwendig sein.

Aufbaudaten:

Leiterquerschnitt:	2 x ≈0,5 mm ²
Litze:	≥ 16 x ≤0,2mm
Verseilung:	≥ 20 Verseilschläge / m
Gesamtabschirmung:	Geflecht, verzinnter Kupferdraht Ø ≥ 1,1 mm ² 85 % optische Bedeckung
Gesamt-Ø:	≥ 5 mm
Außenmantel:	je nach Anforderungen an Entflammbarkeit, Verbrennungsrückstände etc.

Thermische / elektrische Eigenschaften:

Leiterwiderstand (20°C):	≤ 40 Ω/km
Isolationswiderstand (20°C):	≥ 200 MΩ/km
Betriebsspannung (20°C):	≥ 300V
Prüfspannung (20°C):	≥ 1500V
Temperaturbereich:	-40C ≤ T ≤ 80°C
Belastbarkeit:	≥ 5A
Kapazität:	≤ 120 pF/m

Mechanische Eigenschaften:

Einmalige Biegung:	≤ 5 x Außendurchmesser
Wiederholte Biegung:	≤ 20 x Außendurchmesser

Empfehlungen:

1. Standard, ohne besondere Anforderungen:

Zweiadrige, flexible, abgeschirmte Schlitze nach VDE 0812 mit farbiger PVC-Ummantelung.
PVC-Isolierung ölbeständig und benzinfest.

Typ: LIYCY 2 x 0,5 mm²
z.B. Fa. Metrofunk Kabel-Union GmbH 12111 Berlin Postfach 41 01 09
Tel 030-831 40 52, Fax: 030-792 53 43

2. Halogenfreie Leitung (kein Salzsäurenebel bei Bränden):

Halogenfrei, hochflexibel, hochhitze- und kältebeständig. Mantel aus ASS-Spezialmischung auf Silikonbasis

Typ: ASS 1 x 2 x 0,5 mm²
bei: Fa. Metrofunk Kabel-Union GmbH 12111 Berlin Postfach 41 01 09
Tel 030-831 40 52, Fax: 030-792 53 43

3. Empfehlung, wenn halogen- und silikonfreie Leitungen gefordert sind:

Typ: BETAflam G-M/G-G-B1 flex 2 x 0,5 mm²
bei: Fa. Studer-Kabel-AG, CH 4658 Däniken

2.2. Leitungslängen

Die Leitungslänge ist abhängig von der Übertragungsrate und der Zahl der angeschlossenen Teilnehmer. Unter den in 2.1 genannten Leitungseigenschaften sind folgende Kabellängen möglich:

Übertragungsrate	max. Teilnehmerzahl	max. Leitungslänge
9,6 kbit/s	32	1200 m
19,2 kbit/s	32	1200 m
38,4 kbit/s	32	1200 m
187,5 kbit/s	30	1000 m

Tabelle 2.1: Leitungslängen in Abhängigkeit von der Übertragungsrate

2.3. Schnittstelleneigenschaften

Im folgenden wird nur auf die Realisierung der physikalischen Schnittstelle nach EIA RS-485 eingegangen.

Die Schnittstelle kann potentialbehaftet oder potentialfrei gegenüber der internen Elektronikversorgungsspannung aufgebaut sein.

Das Buszugriffsverfahren nach dem Master-Slave-Prinzip ermöglicht es, den Bus mit nur zwei Drähten aufzubauen.

Dies erfordert, daß zu einer Zeit nur ein Sender auf den Bus zugreifen darf. Alle anderen Teilnehmer müssen ihre Sender hochohmig schalten ("Hörbetrieb").

Die logischen 0- und 1-Zustände werden bei der RS-485-Technik durch das Vorzeichen der Spannungsdifferenz zwischen den beiden Datenleitungen gekennzeichnet.

Für die Eigenschaften der Sender- und Empfängerbausteine gelten grundsätzlich die Angaben aus den Standards EIA RS-485 und EIA RS-422. Die im folgenden aus diesen Standards entnommenen Werte sollen Eckwerte angeben, wie die ordnungsgemäße Signalform an einem installierten Bus geprüft werden kann.

Nach dem RS-485-Standard muß ein Sender unter Testbedingungen (der Bus ist mit 54Ω abgeschlossen) eine bestimmte Spannungsdifferenz V_O zwischen den Signalleitungen RS485P und RS485N erzeugen:

logischer 1-Zustand: $1,5 \text{ V} \leq V_O \leq 5 \text{ V}$ RS485P ist positiv gegenüber RS485N

logischer 0-Zustand: $-5 \text{ V} \leq V_O \leq -1,5 \text{ V}$ RS485P ist negativ gegenüber RS485N

Die Anstiegs- und Abfallzeit, das heißt die Zeit für den Wechsel zwischen zwei logischen Zuständen, darf auf dem Bus nicht größer sein als:

$$0,3 / \text{Übertragungsrate}$$

Sendet kein Teilnehmer, so wird die Spannungsdifferenz durch das Basisnetzwerk (siehe Kapitel 4.5 Busabschluß) auf positiven Pegel definiert.

Die Übertragung erfolgt asynchron.

Die Abweichung des Empfangs- und Sendetaktes vom Nominalwert darf maximal $\pm 0,3\%$ betragen.

Die Zeichen eines Telegramms werden als bitserielle UART-Zeichen gesendet und empfangen.

Die Zeit zwischen dem Stopbit eines Zeichens und dem Startbit des nächsten Zeichens innerhalb eines Telegramms muß kürzer als zwei Zeichenlaufzeiten sein.

2.4. Übertragungsraten

Folgende Übertragungsraten kommen für USS[®]-Schnittstellen in Frage:

300 bit / s
600 bit / s
1200 bit / s
2400 bit / s
4800 bit / s
9600 bit / s
19200 bit / s
38400 bit / s
57600 bit / s
76800 bit / s
93750 bit / s
115200 bit / s
187500 bit / s

Die fett markierten Übertragungsraten werden für USS[®] empfohlen und sollten von allen Anschaltungen realisiert werden.

3. Übertragungsverfahren

Zur Datenübertragung werden die in der Digitaltechnik allgemein üblichen UART-Bausteine für die serielle asynchrone Datenübertragung verwendet.

3.1. Bitcodierung

Die Zeichen werden so übertragen, wie sie vom UART-Baustein codiert werden. Der entstehende Code wird als Non-Return-to-Zero-Code (NRZ-Code) bezeichnet. Ein Bit besteht aus einem Rechteckimpuls, dessen Breite dem Takt entspricht ($1/\text{Übertragungsrate}$).

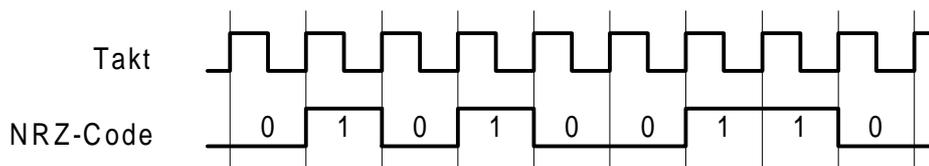


Bild 3.1: NRZ-Code

3.2. Zeichenrahmen

Jedes übertragene Zeichen beginnt mit einem Startbit und endet mit einem Stopbit. Es werden 8 Datenbits übertragen. Jedes Zeichen (Byte) ist durch ein Paritätsbit gesichert (gerade Parität, d. h. die Anzahl der log. Einsen in den Datenbits einschließlich Paritätsbit ist geradzahlig). Bei Nichteinhaltung des Zeichenrahmens (siehe Bild 3.1) wird eine Fehlermeldung erzeugt.

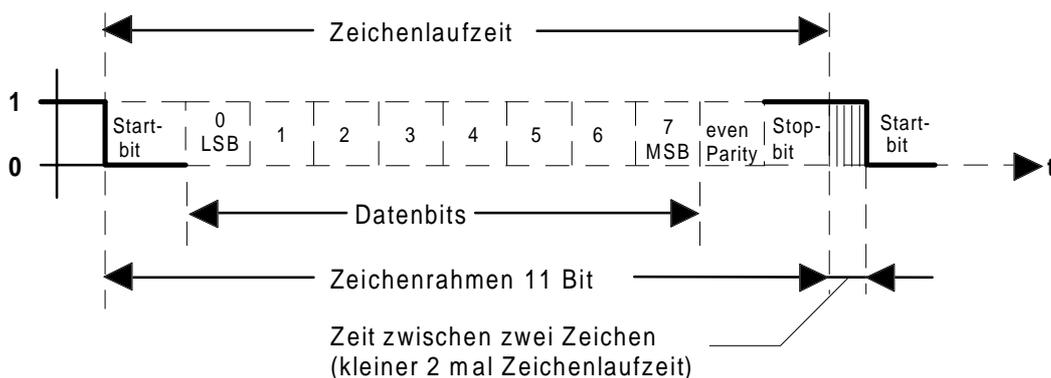


Bild 3.2: Zeichenrahmen

Das Startbit ist immer logisch 0, die 8 Datenbits haben ein beliebiges Bitmuster, das Paritätsbit ist entweder 1 oder 0, das Stopbit ist immer logisch 1. Der Signalpegel bleibt bis zum Startbit des nächsten Zeichens desselben Telegramms logisch 1.

LSB ist das niederwertigste Bit, MSB ist das höchstwertigste Bit.

4. Aufbaurichtlinien

4.1. Leitungsführung

Bei der Verlegung des USS[®]-Bus-Kabels ist insbesondere auf einen EMV-gerechten Aufbau der Anlage zu achten. Folgende Punkte sind besonders zu beachten:

- Die Busleitung darf zusammen mit anderen Datenleitungen (PG, Drucker, SINEC L2-DP), ungeschirmten Leitungen für Gleichspannung ≤ 60 V und ungeschirmten Leitungen für Wechselspannungen ≤ 25 V in einem Bündel oder Kabelkanal verlegt werden.
- Die Busleitung darf nicht mit ungeschirmten Leitungen für Gleichspannung > 60 V und ≤ 400 V zusammen in einem Bündel oder Kabelkanal verlegt werden.
- Zwischen der Busleitung und nicht geschirmten Leitungen für Wechselspannungen > 25 V und ≤ 400 V muß ein Mindestabstand von 10 cm eingehalten werden.
- Zwischen der Busleitung für USS[®] und Leitungen für Datennetze (z.B. SINEC H1) ist ein Mindestabstand von 50 cm einzuhalten.
- Von Störquellen (Transformator, Schütz, Motor, Elektroschweißgerät) ist ein Mindestabstand von 1 m einzuhalten.
- Die Kabel müssen auf möglichst kurzem Weg zwischen 2 Teilnehmern verlegt werden.
- Potentialausgleichsleitungen und Signalkabel sollen in möglichst kleinem Abstand zueinander verlegt werden.
- Kabelverlängerungen über Klemmen oder Stecker sollten vermieden werden.
- Leitungen sind eng an Masseflächen zu führen.

4.2. Potentialausgleich

Sind Teilnehmer am Bus über potentialgebundene Schnittstellen verbunden oder sind angeschlossene Teilnehmer an unterschiedlichen Anlagenteilen geerdet oder ist die Differenz zwischen den 0 V Potentialen der Schnittstellenelektroniken ≥ 7 V, so kann die Datenübertragung gestört sein und es können Schäden an den Baugruppen entstehen.

In allen Fällen ist für einen Potentialausgleich zwischen den Erdungspotentialen der Umrichter zu sorgen.

Querschnitt der Potentialausgleichsleitung aus Cu:

- 16 mm² bei Leitungslängen bis 200 m
- 25 mm² bei Leitungslängen über 200 m

Die Potentialausgleichsleitung ist bei allen Teilnehmern großflächig mit der Masse/Erde zu verbinden.

(siehe auch Bild 4.2: Schirmung und Potentialausgleich)

4.3. Schirmung

Die Schirmung ist eine Maßnahme zur Dämpfung von magnetischen, elektrischen und elektromagnetischen Störfeldern. Störströme werden durch das Schirmgeflecht über die Gehäusemasse zur Erde abgeleitet.

- Die Schirme sind bei allen Teilnehmern großflächig mit der Gehäusemasse/Erde (geerdete Schirmschiene) zu verbinden.
- Ist kein Potentialausgleich verlegt (Ausnahmefall), so darf der Schirm nur einseitig aufgelegt werden.
- Der Schirm sollte mit dem metallenen Steckergehäuse verbunden sein.
- Das Steckergehäuse in der Schnittstellenelektronik darf keinen Kontakt zur Gehäuse-/Elektronikversorgungs-masse haben (andernfalls Schirm nicht mit Steckergehäuse verbinden).
- Der Schirm darf mit keinem Pin (z.B. Pin 1) verbunden werden, weil sonst die Störströme über die Schnittstellenelektronik zur Erde abgeleitet werden (Zerstörung der Schnittstellen-Elektronik).

(siehe auch Bild 4.2: Schirmung und Potentialausgleich)

4.4. Anschlußtechnik, Steckerbelegungen

Die Ausführung der Stecker / Klemmen und die Belegung wird nicht festgeschrieben. Im Einzelfall ist auf die Besonderheiten des Gerätes (Platzbedarf, Zugänglichkeit etc..) Rücksicht zu nehmen. Es empfiehlt sich jedoch, sich so weit wie möglich an der Anschlußtechnik des SINEC L2-DP zu orientieren.

Pinbelegung der Busschnittstelle mit 9-poligem SUB-D-Stecker:

An der Schnittstellenseite sind die Buchsen, auf der Kabelseite die Stifte.

PIN 1	-	frei
PIN 2	-	frei
PIN 3	RS485P	Empfangs- und Sendesignal (+)
PIN 4	(-)	Reserve (bei SINEC L2-DP Richtungssignale für Repeater)
PIN 5	0 V	Datenbezugspotential
PIN 6	5 V	Versorgungsspannung
PIN 7	-	frei
PIN 8	RS485N	Empfangs- und Sendesignal (-)
PIN 9	(-)	Reserve (bei SINEC L2-DP Richtungssignale für Repeater)

Belegung der Busschnittstelle bei Klemmenanschluß:

Klemme 1	RS485P	Anschluß Leitung 1, positiver Pegel
Klemme 2	RS485N	Anschluß Leitung 1, negativer Pegel
Klemme 3	RS485P	Anschluß Leitung 2, positiver Pegel
Klemme 4	RS485P	Anschluß Leitung 2, negativer Pegel
Klemme 5	0Vext	
Klemme 6	5Vext	optional, wird nur bei externem Basisnetzwerk gebraucht

**Klemme 1 und Klemme 3 sind intern verbunden;
Klemme 2 und Klemme 4 sind intern verbunden**

Leitung 1: Leitung zum vorherigen Teilnehmer in der Buslinie

Leitung 2: Leitung zum nachfolgenden Teilnehmer in der Buslinie

Falls der Bus nicht unterbrochen werden darf, wenn der Stecker von der Schnittstelle abgezogen wird, müssen die Adern RS485P von Leitung 1 und 2 unter der Klemme 1 und die Adern RS485N von Leitung 1 und 2 unter der Klemme 2 angeschlossen werden.

4.5. Busabschluß

Die Busleitung muß beidseitig abgeschlossen werden.

Dazu ist sind am ersten und am letzten Teilnehmer jeweils ein Widerstand mit 150Ω zwischen die Datensignalleitungen RS485P und RS485N zu schalten.

Der Busabschlußwiderstand ist auf der Schnittstellenbaugruppe, durch Brückeneinstellungen aktivierbar, einzubauen. Im Auslieferungszustand ist der Busabschlußwiderstand nicht aktiviert.

Kann der Busabschluß nicht auf der Schnittstellenbaugruppe untergebracht werden, so ist dieser im Steckergehäuse vorzusehen.

Basisnetzwerk

Sendet kein Teilnehmer, so liegt auf dem Bus ein undefiniertes Potential, weil alle Sender hochohmig geschaltet sind. Zur Unterdrückung von Signalstörungen in diesem Zustand wird der Bus mit einem Basisnetzwerk beschaltet, damit ein definierter positiver Signalpegel anliegt.

Das Basisnetzwerk ist an den Teilnehmern anzuschließen, an denen der Busabschluß ausgeführt ist.

Für die Anordnung der Widerstände des Basisnetzwerkes gilt das gleiche wie für den Busabschlußwiderstand.

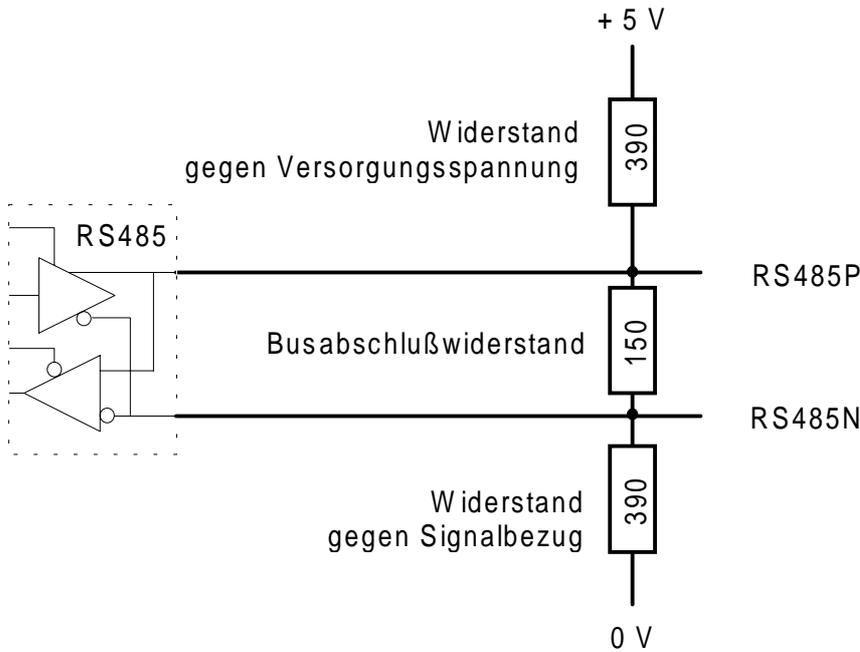


Bild 4.1 : Basisnetzwerk und Busabschluß

Beträgt die Versorgungsspannung der Schnittstelle 5 V, so werden 390Ω Widerstände empfohlen (bei 15V ca. $1 k\Omega$ gegen Versorgungsspannung und 390Ω gegen 0 V).

Die Schnittstelle muß dafür die Versorgungsspannung auf Pin oder Klemme herausführen. Die Spannungsquelle muß einen Kurzschlußstrom von 10 mA treiben können.

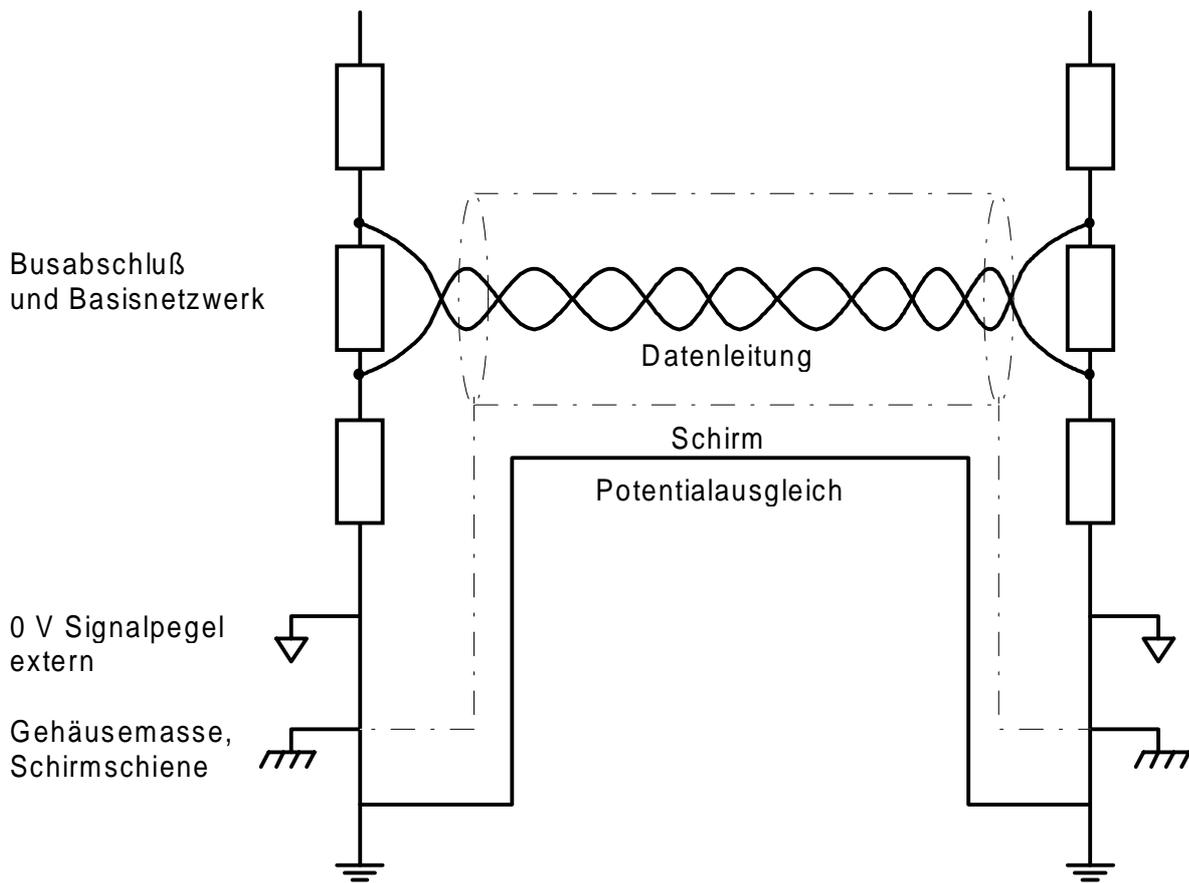
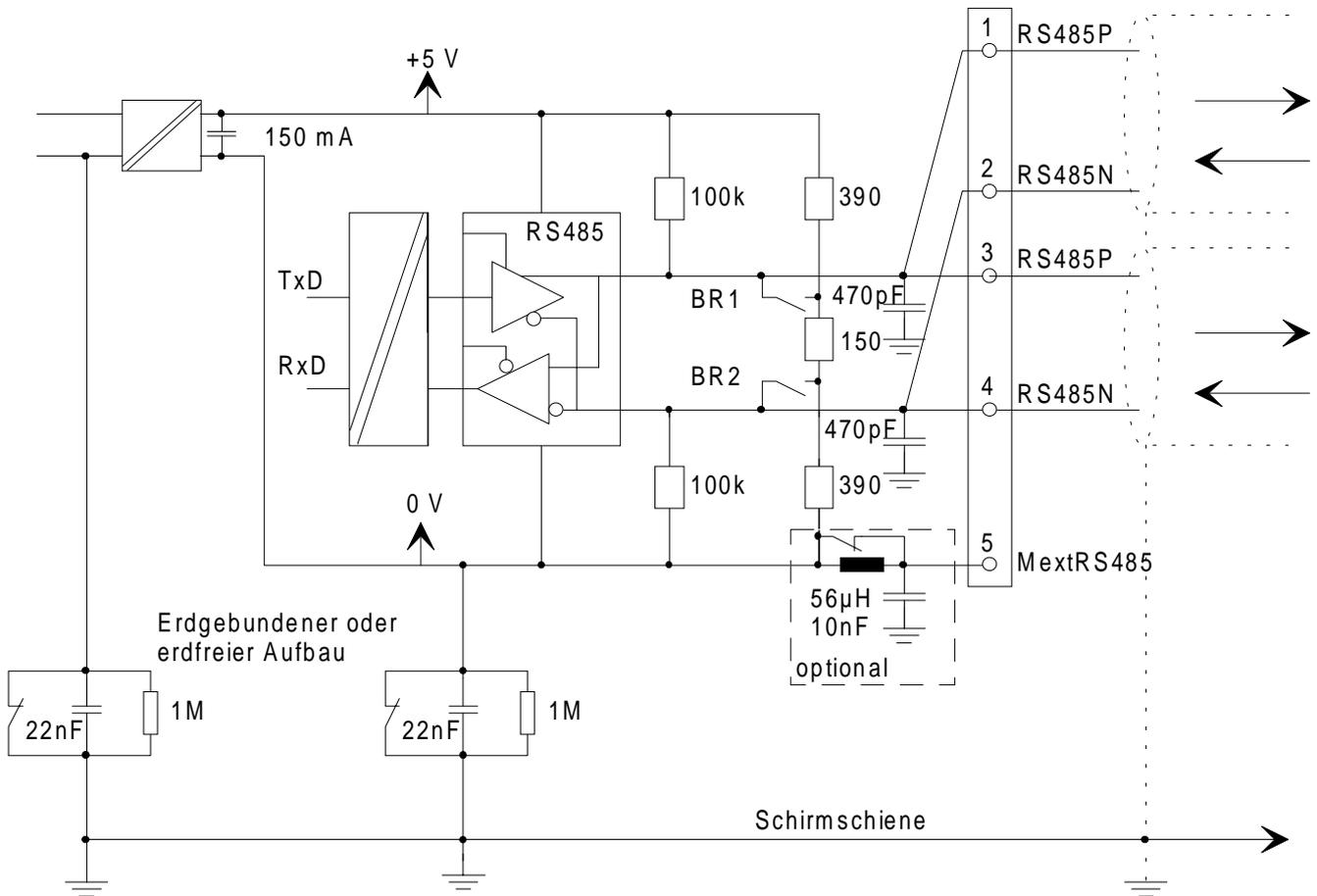


Bild 4.2: Schirmung und Potentialausgleich

4.6. Schaltungsvorschlag

Nachfolgendes Bild zeigt einen Schaltungsvorschlag für eine potentialgetrennte USS[®]-Schnittstelle, deren Anschlüsse auf Klemmen geführt sind:



BR1 und BR2 bei Busabschluß geschlossen, Auslieferungszustand offen.
 Potentialtrennung ist optional.

Bild 4.3: Aufbau der EIA RS485-Schnittstelle 2-Draht potentialgetrennt

C: Festlegung der Nutzdaten für die Anwendung in der Antriebstechnik

1. Einleitung

Mit Hilfe des USS[®]-Protokolls kann ein Anwender eine serielle Buskopplung zwischen einem übergeordneten Master- und mehreren Slavesystemen aufbauen. Mastersysteme können z. B. speicherprogrammierbare Steuerungen (SPS) oder PC's sein. Die Antriebe SIMOVERT und SIMOREG sind dabei immer Slaves am Bussystem.

Das USS[®]-Protokoll bietet dem Anwender die Möglichkeit, sowohl Automatisierungsaufgaben mit Forderung nach einem zeitzyklischen Telegrammverkehr (⇒ feste Telegrammlänge notwendig), als auch Visualisierungsaufgaben zu realisieren. In diesem Fall ist das Protokoll mit variabler Telegrammlänge von Vorteil, da Texte und Parameterbeschreibungen ohne "Zerstückelung" der Information mit einem Telegramm übertragen werden können.

Automatisierungsaufgaben in der Antriebstechnik (Steuern, Regeln) fordern einen zyklischen Telegrammverkehr, der nur realisiert werden kann, wenn die Telegramme auf eine feste Länge beschränkt werden. Die eingestellte Telegrammlänge darf während des Betriebs nicht verändert werden. Die Festlegung auf eine feste Telegrammlänge begrenzt die Anzahl der Zeichen im Nutzdatenblock des Telegramms.

Die nachfolgenden Kapitel beschreiben detailliert den **Aufbau des im Telegramm enthaltenen Nutzdatenblocks**, sowie die daraus resultierenden notwendigen Einstellungen (Parametrierungen) der Schnittstelle, über die der Datenaustausch erfolgen soll. Die Kommunikation kann über eine Schnittstelle am Grundgerät oder auch über eine separate Schnittstellenbaugruppe erfolgen.

Der Aufbau des Nutzdatenblocks im Telegramm ist losgelöst von der Spezifizierung des Protokolls mit dem die Nutzdaten übertragen werden. Der Aufbau, d. h. Inhalt und Struktur der Nutzdaten entsprechen im wesentlichen den Festlegungen für den zyklischen Datenaustausch über PROFIBUS (PROFIBUS-Profil "Drehzahlveränderbare Antriebe " /1/).

Damit ist für den Anwender sichergestellt, daß er mit den gleichen Zugriffsmechanismen auf die Prozeßdaten (= Steuer- / Zustandsworte und Soll- / Istwerte) und Parameter eines Gerätes zugreifen kann, unabhängig davon, ob dies über USS[®] oder PROFIBUS-DP/-FMS erfolgt.

Die nachfolgenden Beschreibungen sind geräteunabhängig gehalten und sind eine allgemeingültige Richtlinie für die Festlegung des Nutzdatenaufbaus und der Datenübertragung mit dem USS[®]-Protokoll. Für die Dokumentation des USS[®]-Protokolls bei den einzelnen Geräten ist eine gerätespezifische Applikation zu erstellen. In dieser Applikation ist zu beschreiben, wie der Bus aufzubauen und das Protokoll zu parametrieren ist, ferner ist zu definieren, welche Nutzdateninhalte das Gerät "versteh".

Die Einhaltung der Spezifikation ist für alle Implementierungen bindend.

2. Allgemeiner Aufbau des Nutzdatenblocks

Der Nutzdatenblock teilt sich in die zwei Bereiche, den

- PKW (**P**arameter-**K**ennung-**W**ert)-Bereich und den
 - PZD (**P**rozeß**d**aten)-Bereich
- auf.

- Der **PKW-Bereich** bezieht sich dabei auf das Handling der **P**arameter-**K**ennung-**W**ert (PKW)-Schnittstelle. Unter dem Begriff PKW-Schnittstelle ist keine physikalische Schnittstelle zu verstehen, sondern es wird damit ein Mechanismus beschrieben, der den Parameteraustausch zwischen zwei Kommunikationspartnern regelt. D. h., Lesen und Schreiben von Parameterwerten, Parameterbeschreibungen und zugehörige Texten sowie die Behandlung von Parameteränderung durch Spontanmeldungen. Alle Aufgaben, die über die PKW-Schnittstelle erfolgen, sind im wesentlichen Aufgaben für Bedienen und Beobachten, Service und Diagnose.
- Der **PZD-Bereich** beinhaltet die für die **Automatisierung** notwendigen Signale: Steuerwort(e) und Sollwert(e) vom Master zum Slave
Zustandswort(e) und Istwert(e) vom Slave zum Master.

Beide Bereiche zusammen ergeben den Nutzdatenblock. Dieser Aufbau gilt für das **Auftragstelegramm** (Master → Slave) als auch für das **Antworttelegramm** (Slave → Master).

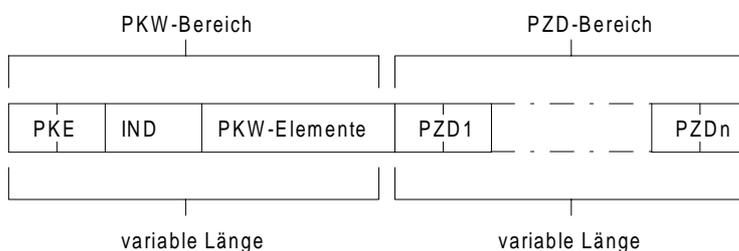


Bild 2.1: Nutzdatenblock

Unter einem **Auftragstelegramm** wird die Übertragung des gesamten Nutzdatenblocks vom Master an den Slave verstanden.

Unter einem **Antworttelegramm** wird die Übertragung des gesamten Nutzdatenblocks vom Slave an den Master verstanden.

PKW-Bereich:

Mit Hilfe von sogenannten "Aufträgen" und "Antworten" wird in diesem Bereich der Zugriff auf die in einem Gerät befindlichen und über eine USS[®]-Schnittstelle zugänglichen Parameter behandelt.

- PKE = **Parameter-Kennung** (PKE); dient zur Kennzeichnung und Veranlassung von Aufträgen und Antworten zur Bearbeitung von Parametern und besitzt immer die Länge **ein Wort** (≤ 16 Bit). In PKE ist auch die Parameternummer enthalten.
- IND = Index hat immer die Länge ein Wort. Die Bedeutung des Index wird in Kap. 4.2.2 ausführlich beschrieben.
- PWE-Element = Hier werden die zu einem Auftrag oder einer Antwort (in PKE festgelegt) gehörenden Informationen, wie Parameterwerte, Texte oder Parameterbeschreibungsdaten übertragen. Dieser Bereich kann je nach Auftrag/Antwort in der Länge variieren. Sollen nur PZD-Daten im Nutzdatenblock übertragen werden, kann die Anzahl der PKW-Elemente (PKE + Index + PWE-Elemente) auf 0 gesetzt werden!

PZD-Bereich:

In diesem Bereich werden **ständig** Prozeßdaten zwischen dem Master und den Slaves ausgetauscht. Welche Prozeßdaten mit einem Slave ausgetauscht werden, wird zu Beginn der Kommunikation fest projektiert. Zum Beispiel wird an den Slave x im zweiten PZD (= PZD2) der Stromsollwert übertragen. Diese Einstellung bleibt für die gesamte Übertragung fest.

- PZD1-PZDn = **Prozeßdaten** (= Steuer- / Zustandswort(e) und Soll- / Istwert(e));
In diesem Bereich werden die für die Automatisierung notwendigen Steuer- / Zustandswort(e), Soll- und Istwerte übertragen.
Die Länge des PZD-Bereichs wird bestimmt durch die Anzahl der PZD-Elemente und deren Größe (z. B. Wort, Doppelwort). Im Gegensatz zum PKW-Bereich, der variabel sein kann, muß die Länge dieses Bereichs zwischen den Kommunikationspartnern immer fest vereinbart werden! Die maximale Anzahl der PZD-Worte je Telegramm ist auf 16 Worte begrenzt. Sollen nur PKW-Daten im Nutzdatenblock übertragen werden, dann kann die Anzahl der PZD auch 0 sein!
Im PZD1 ist je nach Übertragungsrichtung immer das Steuerwort oder das Zustandswort zu übertragen. Im PZD 2 entsprechend der Übertragungsrichtung immer der Hauptsollwert bzw. der Hauptistwert. In den nachfolgenden Prozeßdaten PZD3 bis PZDn werden weitere Sollwerte bzw. Istwerte oder Steuer- / Zustandsworte gesendet.

Vereinbarungen über die Länge der einzelnen Bereiche müssen zwischen den Kommunikationspartnern getroffen werden, siehe Kap. 7. Hierfür sind im Grundgerät verschiedene Parameter zur Verfügung zu stellen, mit denen die PKW- und PZD-Anteile bei den Protokollen an seriellen Schnittstellen einstellbar sind.

3. Parametrierung des USS[®]-Protokolls an einer seriellen Schnittstelle

Jede serielle Schnittstelle, an der das USS[®]-Protokoll implementiert werden soll, muß neben den Parametern für die Busadresse, Baudrate und Telegrammausfallzeit, zwei Parameter haben, an denen die Länge des PKW-Bereiches und des PZD-Bereiches unabhängig voneinander eingestellt werden kann. Die entsprechende Schnittstelle kann am Grundgerät oder auf einer Kommunikationsbaugruppe vorhanden sein. Letzteres setzt voraus, daß ein entsprechender Mechanismus existiert, um die notwendigen Parametrierungen auf der Kommunikationsbaugruppe vornehmen zu können. Zum Beispiel kann die Übergabe der relevanten Parameter vom Grundgerät über eine Dual Port-RAM-Kopplung zur Kommunikationsbaugruppe erfolgen.

3.1. Parametereinstellung für Geräte SIMOVERT P 6SE21, 6SE30 (MircoMaster) und SIMOREG K 6RA24

Baudrate

PNU: gerätespezifisch		→ Parameternummer (PNU), z. B. P783 beim SIMOREG K 6RA24
Bezeichner: gerätespezifisch		→ Dient zur Klartext-Anzeige am Bedienfeld z. B. BD_RATE
Typ: 02		→ Datentyp: siehe /1/
Funktion: Baudrate		
Parameterwert (PWE)		
1 [^] 300 Baud		
2 [^] 600 Baud		
3 [^] 1200 Baud		
4 [^] 2400 Baud		
5 [^] 4800 Baud		
6 [^] 9600 Baud		
7 [^] 19200 Baud		
8 [^] 38400 Baud		
9 [^] 93750 Baud		
10 [^] 187500 Baud		

Anmerkung: Spezielle Baudraten, die über diese Festlegungen hinausgehen werden > 20 hinterlegt!

Busadresse

PNU:	gerätespezifisch	→ z. B. BUS_ADR
Bezeichner:	gerätespezifisch	
Typ:	O2	
Funktion:	Busadresse	
PWE:	0 - 31	

Telegrammausfallzeit

Um auf dem Grundgerät eine Überwachung der zeitlichen Ansprache durch den Master durchführen zu können, soll über diesen Parameter der maximale Zeitabstand zwischen zwei, für das Grundgerät adressierte, gültige Telegramme parametrisiert werden. Die Angabe bezieht sich dabei auf die Einheit "Sekunden [sec]". Soll keine Überwachung stattfinden, ist der Parameter auf den Wert 0 einzustellen.

PNU:	gerätespezifisch	→ z. B. TLG_AUS
Bezeichner:	gerätespezifisch	
Typ:	O2	
Funktion:	Telegrammausfallzeit in Sekunden für Grundgeräteschnittstelle	
PWE:	0 - 32 (0 : keine Überwachung) (1 : Werkseinstellung)	

Anmerkung:

Der Slave beginnt nach dem Einschalten seiner Stromversorgung erst dann mit der Zeitüberwachung, nachdem das erste fehlerfreie Auftragstelegramm empfangen wurde.

PKW-Elemente-Anzahl

Wieviele PKW-Elemente im PKW-Bereich des Nutzdatenblocks vorhanden sein sollen, ist über diesen Parameter einstellbar. Die Angabe bezieht sich immer auf PKW-Elemente mit Wortlänge.

PNU:	gerätespezifisch	→ z. B. PKW_ANZ
Bezeichner:	gerätespezifisch	
Typ:	O2	
Funktion:	PKW-Anzahl	
PWE:	0 → 0 Worte 3 → konstant 3 Worte 4 → konstant 4 Worte 127 → variable Länge	

Achtung:

Sollen Telegramme nur mit konstanter Nutzdatenanzahl verwendet werden, darf die Addition von PKW-Anzahl und PZD-Anzahl den Wert 126 nicht überschreiten. Gemäß der USS[®]-Protokoll-Spezifikation sind für die Nutzdaten maximal 252 Byte (126 Worte) zugelassen. Sollen Telegramme mit variablen PKW-Anteilen verwendet werden, muß dieser Parameter (PKW_ANZ) auf 127 gesetzt werden, unabhängig davon, wie der Parameter PZD_ANZ steht.

PZD-Elemente-Anzahl

Die im Nutzdatenblock enthaltene Anzahl von Prozeßdaten, kann durch diesen Parameter beeinflusst werden. Die Angabe bezieht sich immer auf PZD-Element mit Wortlänge.

PNU:	gerätespezifisch	→ z. B. PZD_ANZ
Bezeichner:	gerätespezifisch	
Typ:	O2	
Funktion:	PZD-Anzahl	
PWE:	0 - 16 (Worte)	

3.2. Parametereinstellung für SIMOVERT Master Drives

Im Gegensatz zu den bisherigen Parameterdefinitionen sind die Parameter Baudrate, Busadresse, PKW_ANZ und PZD _ANZ Parameter von Datentyp "Array" mit Index 1 bis 3 für die jeweiligen USS[®]-Protokolle auf dem Grundgerät SST1, bzw. SST2 und auf der Baugruppe SCB (**S**erial **C**ommunication **B**oard).

PNU: P683	
Bezeichner: SST/SCB Busadr.	
Typ: Array mit Datentyp 02	
Funktion: Busadresse für USS [®] an den Schnittstelle am Grundgerät und auf der Kommunikationsbaugruppe SCB	
Index	Parameterwert (PWE)
1 \wedge SST 1	0 bis 30
2 \wedge SCB	
3 \wedge SST 2	

→ Datentyp: siehe /1/

PNU: P684	
Bezeichner: SST/SCB Baudrate	
Typ: Array mit Datentyp 02	
Funktion: Baudrate für USS [®] an den Schnittstelle am Grundgerät und auf der Kommunikationsbaugruppe SCB	
Index	Parameterwert (PWE)
1 \wedge SST 1	1 \wedge 300 Baud
2 \wedge SCB	2 \wedge 600 Baud
3 \wedge SST 2	3 \wedge 1200 Baud
	4 \wedge 2400 Baud
	5 \wedge 4800 Baud
	6 \wedge 9600 Baud
	7 \wedge 19200 Baud
	8 \wedge 38400 Baud

→ Datentyp: siehe /1/

Beispiel: die Baudrate von 38,4 kbit/s an der Schnittstelle 1 am Grundgerät wird über setzen des Parameters P684 mit Index =1 auf den Parameterwert = 8 eingestellt.

PNU: P685	
Bezeichner: SST/SCB PKW-Anz	
Typ: Array mit Datentyp 02	
Funktion: Anzahl der Worte (16 Bit) im PKW-Bereich im Nutzdatenblock an den Schnittstellen am Grundgerät und auf der Kommunikationsbaugruppen SCB	
Index	Parameterwert (PWE)
1 $\hat{=}$ SST 1	0 \rightarrow 0 Worte
2 $\hat{=}$ SCB	3 \rightarrow konstant 3 Worte
3 $\hat{=}$ SST 2	4 \rightarrow konstant 4 Worte
	127 \rightarrow variable Länge

→ Datentyp: siehe /1/

PNU: P686	
Bezeichner: SST/SCB PZD-Anz	
Typ: Array mit Datentyp 02	
Funktion: Anzahl der Wort (16 Bit) im PZD-Bereich des Nutzdatenbereich an den Schnittstellen am Grundgerät und auf der Kommunikationsbaugruppe SCB	
Index	Parameterwert (PWE)
1 $\hat{=}$ SST 1	0 bis 16
2 $\hat{=}$ SCB	
3 $\hat{=}$ SST 2	

→ Datentyp: siehe /1/

PNU: P687	
Bezeichner: SST/SCB TGL-Ausz.	
Typ: Array mit Datentyp 02	
Funktion: Telegrammausfallzeit für USS [®] an den Schnittstellen am Grundgerät und auf der Kommunikationsbaugruppe SCB	
Index	Parameterwert (PWE)
1 <u>^</u> SST 1	0 bis 6500 [ms]
2 <u>^</u> SCB	
3 <u>^</u> SST 2	

→ Datentyp: siehe /1/

4. PKW-Bereich

4.1. Aufbau des PKW-Bereiches (Parameter-Kennung-Wert)

Der Aufbau des PKW-Bereiches ist in der Reihenfolge seiner Elemente immer gleich und unterscheidet sich in seinem Standardaufbau nur durch die Anzahl seiner Parameterwerte (PWE).

Es gibt über den Parameter PKW_ANZ, siehe Kap. 3.1 die Möglichkeit, den PKW-Bereich mit fester Länge (3 Worte oder 4 Worte groß) oder mit variabler Länge einzustellen.

Soll kein PKW-Bereich im Nutzdatenblock vorhanden sein, so muß **PKW_ANZ = 0** sein, dann ist **auch keine Parametrierung des Gerätes über diese Schnittstelle möglich!**

4.1.1. PKW-Bereich bei fester Telegrammlänge

Standard-Aufbau bei Parameterwerten als Wortgrößen (16 Bit):

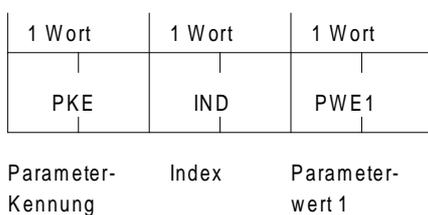


Bild 4.1: Aufbau PKW-Bereich 3 Worte

Ist der PKW-Bereich nicht auf 0 gesetzt, siehe Parametrierung der Schnittstelle, so muß dieser Bereich **3 Worte** groß sein.

Bei fester Telegrammlänge ist die Anzahl der Worte im PKW-Bereich sowohl für das Auftragstelegramm (Master an Slave) als auch für das Antworttelegramm (Slave an Master) immer konstant und gleich groß. Parametrierung des festen PKW-Bereichs, siehe Kap. 3.1; hier muß PKW_ANZ = 3 gesetzt werden.

Anmerkung:

Die Festlegung bei fester Telegrammlänge auf gleich PKW_ANZ und PZD_ANZ ist notwendig, um eine korrekte und optimale Telegrammüberwachung "Telegrammausfallzeit" parametrieren zu können.

Standard-Aufbau bei Parameterwerten als Doppelwortgrößen (32 Bit):

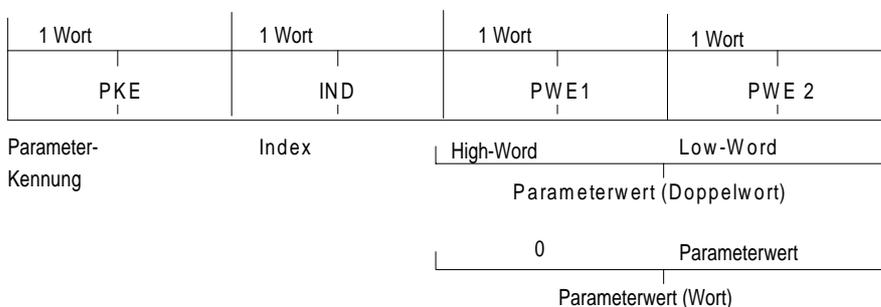


Bild 4.2: Aufbau PKW-Bereich 4 Worte

In diesem Fall ist, falls nicht PKW-Bereich gleich 0 parametrieren ist, die Anzahl der Worte im PKW-Bereich = 4 zu parametrieren. Dies gilt auch hier sowohl für das Auftrags- als auch für das Antworttelegramm.

4.1.2. PKW-Bereich mit variabler Telegrammlänge

Standard-Aufbau:



Bild 4.3: Aufbau PKW-Bereich variable Länge

mit: $1 \text{ Wort} \leq m \leq 108 \text{ Worte (maximal)}$, wenn 16 PZD-Worte (maximal) im Nutzdatenblock sind.
 $1 \text{ Wort} \leq m \leq 124 \text{ Worte (maximal)}$, wenn kein PZD vorhanden.

Telegrammverkehr mit variabler Telegrammlänge bedeutet, daß auf ein Telegramm vom Master der Slave mit einem Telegramm antwortet, dessen Länge nicht mehr mit der Länge des Auftragstelegramms übereinstimmt. Die Länge und die Besetzung der Elemente PWE 1 bis PWE m im Antworttelegramm, ist abhängig vom gestellten Auftrag des Masters. Damit der Slave bei Aufträgen, die eine variable Telegrammlänge verlangen, entsprechend reagieren kann, muß der Parameter $\text{PKW_ANZ} = 127$ gesetzt sein.

Nur bei der Parametrierung auf variabler Telegrammlänge ($\text{PKW_ANZ} = 127$) kann der Master auch mit variabler Telegrammlänge auf den Slave zugreifen. Wobei sich "variabel" generell auf einen variablen PKW-Bereich bezieht. Die Länge des PZD-Bereiches muß für Auftrags- und Antworttelegramme immer gleich groß sein. In diesem Fall muß geprüft werden, ob die Einstellung der Telegrammausfallzeit sinnvoll ist.

4.2. Beschreibung der einzelnen PKW-Elemente

4.2.1. Parameterkennung (PKE)

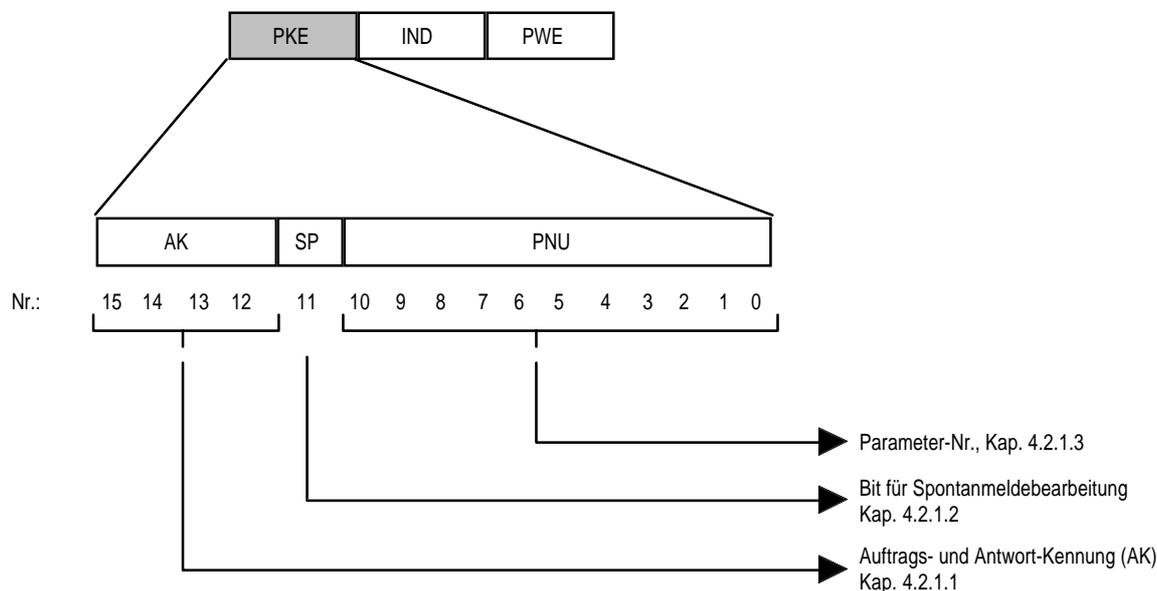


Bild 4.4: Aufbau der Parameterkennung

4.2.1.1. Auftrags- und Antwortkennung

In der Auftragskennung (AK) sind die Aufträge verschlüsselt die vom Auftraggeber ($\hat{=}$ Master) an den Auftragnehmer ($\hat{=}$ Slave) gestellt werden. Der Slave bearbeitet den Auftrag und formuliert die entsprechende Antwort dazu, die dann in verschlüsselter Form (Antwortkennung) dem Master mitgeteilt wird. Die Auftrags- / Antwortkennungen sind so definiert, daß ein Auftrag, bzw. eine Antwort eindeutig durch die PKE (AK + PNU) und bei bestimmten Aufträgen / Antworten zusätzlich durch das Indexwort IND definiert sind.

Auftragskennung (Master → Slave)

Bit-Nr. 15 14 13 12	Funktion	Beschreibung
0 0 0 0	Kein Auftrag	Kein Auftrag für die PKW-Schnittstelle
0 0 0 1	PWE anfordern	Anfordern eines Parameterwertes (PWE)
0 0 1 0	PWE ändern (Wort)	Schreiben eines Parameterwertes in Wortgröße (16 Bit)
0 0 1 1	PWE ändern (Doppelwort)	Schreiben eines Parameterwertes in Doppelwortgröße (32 Bit)
0 1 0 0	PBE-Element anfordern 1)	Lesen eines Elementes aus der Parameterbeschreibung (PBE). Welches Element gelesen werden soll, steht in IND. Ist IND = 255, wird die gesamte PBE angefordert
0 1 0 1	PBE-Element ändern 1)	Schreiben eines Elementes aus PBE; wie bei Lesen von PBE
0 1 1 0	PWE anfordern (Array) 1)	Lesen eines Parameterwertes aus einem eindimensionalen Feld (Δ Array). Die Stelle innerhalb des Feldes, von dem der Wert gelesen werden soll, steht in IND. Z. B. ist IND = 4, so wird der PWE übertragen, der an der 4. Stelle im Array steht.
0 1 1 1	PWE ändern (Array Wort) 1)	Schreiben eines Wertes (als Wort) auf eine bestimmte Stelle in einem eindimensionalen Feld (Array); wie beim Lesen.
1 0 0 0	PWE ändern (Array Doppelwort) 1)	Schreiben eines Wertes als Doppelwort, wie Kennung 0111.
1 0 0 1	Anzahl der Array-Elemente anfordern	Anzahl der Elemente eines Feldes lesen.
1 0 1 0	Reserve	
1 0 1 1	PWE (Array Doppelwort) ändern und ins EEPROM abspeichern 1), 2)	Parameterwert (Doppelwort) auf eine Stelle in einem Array ins EEPROM schreiben
1 1 0 0	PWE (Array Wort) ändern und ins EEPROM abspeichern 1), 2)	Parameterwert (Wort) auf eine bestimmte Stelle in einem Array ins EEPROM schreiben
1 1 0 1	PWE (Doppelwort) ändern und ins EEPROM abspeichern 2)	Parameterwert (Doppelwort) ins EEPROM schreiben
1 1 1 0	PWE (Wort) ändern und ins EEPROM abspeichern 2)	Parameterwert (Wort) ins EEPROM schreiben.
1 1 1 1	Text anfordern oder ändern 1), 2) 3)	Lesen oder schreiben eines Textes

- 1) Bei diesen Aufträgen gehört zur Eindeutigkeit des Auftrages der Wert dazu, der im IND im Nettodatenblock steht, siehe Kap. 4.2.2.
- 2) Diese Auftragskennungen gelten nur für das USS-Protokoll. In den zyklischen Auftragskennungen des PROFIBUS-Profiles /1/ sind diese Kennungen nicht vorhanden.
- 3) Der Zusatz "Text ändern" gilt nur für SIMOVERT Master Drives

Tabelle 4.1: Auftragskennungen

Antwortkennungen (Slave → Master)

Bit-Nr. 15 14 13 12	Funktion	Beschreibung
0 0 0 0	Keine Antwort	Keine Antwort
0 0 0 1	PWE übertragen (Wort)	Übertragen eines Parameterwertes (PWE) als Wort (16 Bit)
0 0 1 0	PWE übertragen (Doppelwort)	Übertragen eines Parameterwertes (PWE) als Doppelwort (32 Bit)
0 0 1 1	PBE-Element übertragen 1)	Übertragen eines Elementes aus der Parameterbeschreibung (PBE). Das wievielte Element der PBE übertragen wird, steht in IND. Ist IND = 255, wird die gesamte PBE übertragen.
0 1 0 0	PWE übertragen (Array Wort) 1)	Übertragen eines Parameterwertes (PWE) von der in IND angegebenen Stelle innerhalb eines eindimensionalen Feldes (△ Array)
0 1 0 1	PWE übertragen (Array Doppelwort) 1)	Wie vorher, nur PWE in Doppelwortgröße
0 1 1 0	Anzahl der Array-Elemente übertragen	Anzahl der Elemente eines Feldes übertragen.
0 1 1 1	Auftrag nicht ausführbar (mit Fehlernummer)	Der gestellte Auftrag kann vom Slave nicht ausgeführt werden. Grund siehe Fehlernummer
1 0 0 0	keine PKW-Bedienhoheit	Von der Schnittstelle auf der dieses Protokoll läuft, dürfen Parameterwerte, Parameterbeschreibungen oder zugehörige Texte nicht geändert, sondern nur gelesen werden.
1 0 0 1	Spontanmeldung (Wort))
1 0 1 0	Spontanmeldung (Doppelwort))
1 0 1 1	Spontanmeldung (Array Wort) 1)) Siehe Kap. 4.2.1.2
1 1 0 0	Spontanmeldung (Array Doppelwort) 1))
1 1 0 1	Reserve	
1 1 1 0	Reserve	
1 1 1 1	Text übertragen 1) 2)	Text wird übertragen

1) Bei diesen Aufträgen gehört zur Eindeutigkeit des Auftrages der Wert dazu, der im Wert IND im Nettodatenblock steht, siehe Kap. 4.2.2.

2) Diese Antwortkennungen gelten nur für das USS-Protokoll. In den zyklischen Antwortkennungen des PROFIBUS-Profiles /1/sind diese Kennungen nicht vorhanden.

Tabelle 4.2: Antwortkennungen

Zusammenhang zwischen gestelltem Auftrag und der zugehörigen Antwort

Auftragskennung		Antwortkennung (positiv)	
Kennung	Funktion	Kennung	Funktion
0 0 0 0	Kein Auftrag	0 0 0 0	Keine Antwort
0 0 0 1	PWE anfordern	0 0 0 1	PWE übertragen (Wort)
		0 0 1 0	PWE übertragen (Doppelwort)
0 0 1 0	PWE ändern (Wort)	0 0 0 1	PWE übertragen (Wort)
0 0 1 1	PWE ändern (Doppelwort)	0 0 1 0	PWE übertragen (Doppelwort)
0 1 0 0	PBE-Element anfordern	0 0 1 1	PBE-Element übertragen
0 1 0 1	PBE-Element ändern	0 0 1 1	PBE-Element übertragen
0 1 1 0	PWE anfordern (Array)	0 1 0 0	PWE übertragen (Array Wort)
		0 1 0 1	PWE übertragen (Array Doppelwort)
0 1 1 1	PWE ändern (Array Wort)	0 1 0 0	PWE übertragen (Array Wort)
1 0 0 0	PWE ändern (Array Doppelwort)	0 1 0 1	PWE übertragen (Array Doppelwort)
1 0 0 1	Anzahl Array-Elemente anfordern	0 1 1 0	Anzahl der Array-Elemente übertragen
1 0 1 1	PWE (Array Doppelwort) ändern und ins EEPROM abspeichern	0 1 0 1	PWE übertragen (Array Doppelwort)
1 1 0 0	PWE (Array Wort) ändern und ins EEPROM abspeichern	0 1 0 0	PWE übertragen (Array Wort)
1 1 0 1	PWE (Doppelwort) ändern und ins EEPROM abspeichern	0 0 1 0	PWE übertragen (Doppelwort)
1 1 1 0	PWE (Wort) ändern und ins EEPROM abspeichern	0 0 0 1	PWE übertragen (Wort)
1 1 1 1	Text anfordern oder ändern	1 1 1 1	Text übertragen
1 0 1 0	Reserve	0 1 1 1	Auftrag nicht ausführbar

Können Aufträge nicht ausgeführt werden, sendet der Auftragnehmer die Antwortkennung "Auftrag nicht ausführbar" und übergibt im Parameterwert (PWE) die entsprechende Fehlerkennung:

Fehlerkennung	Beschreibung
0	unzulässige PNU
1	Parameter nicht änderbar
2	Untere oder obere Wertgrenze überschritten
3	fehlerhafter IND
4	kein Array
5	falscher Datentyp
6	kein Setzen erlaubt
7	Beschreibungselement nicht änderbar
:	:
100	reserviert
> 100	Fehlerkennung größer 100 können gerätespezifisch belegt werden.

Die Fehlerkennungen 0 bis 100 sind identisch mit den Fehlerkennungen des PROFIBUS-Profiles "Drehzahlveränderbare Antriebe" /1/.

Weitere festgelegte Fehlernummern sind in/1/ dokumentiert.

Die in einem Gerät hinterlegten Fehlerkennungen sind der Betriebsanleitung zu entnehmen

Tabelle 4.3: Auftragskennung und zugehörige Antwortkennungen

Auftrags- / Antwortbearbeitung

Die Auftrags- / Antwortbearbeitung beschreibt den zeitlichen und funktionellen Ablauf der Datenübertragung für die PKW-Schnittstelle zwischen Master (Auftraggeber) und den Slaves (Auftragnehmer)

- Ein Auftrag oder eine Antwort für die PKW-Schnittstelle besteht aus den Angaben für die Auftragskennung, die Parameternummer, den Parameterindex und den Parameterwert. Werden einzelne Angaben nicht benötigt, werden diese mit 0 vorbesetzt.
- **Ein** Auftrag oder **eine** Antwort kann sich immer nur auf **einen** Parameterwert beziehen (Ausnahme: Indexwert 255).
- Der Auftraggeber darf **immer nur einen Auftrag an eine Schnittstelle** stellen und muß auf die entsprechende Antwortkennung warten. Solange auf die Antwortkennung gewartet wird, muß der Auftraggeber seinen Auftrag wiederholen!
- Der Auftrag muß in einem Telegramm komplett gesendet werden; gesplittete Auftragstelegramme sind nicht erlaubt. Gleiches gilt für die Antwort!
- Jede Änderung des Auftrags bedeutet einen Neuauftrag, auf den die zugehörige Antwort stattfinden muß! Die Auftragskennung "Kein Auftrag" ist wie jede andere Auftragskennung zu betrachten und muß mit der Antwortkennung "Keine Antwort" beantwortet werden!
- Werden im zyklischen Betrieb keine Informationen von der PKW-Schnittstelle benötigt (nur PZD-Daten sind wichtig), so muß der Auftrag "Kein Auftrag" gestellt werden.
- Liegen zwischen der zyklischen Telegrammfolge und der Antwortbereitstellung im Gerät große Zeitunterschiede, so wird in der Übergangsphase zwischen "Alt-Auftrag" und "Neu-Auftrag" vom Slave noch solange die Antwort zum "Alt-Auftrag" gesendet, bis der "Neu-Auftrag" erkannt und die dazu gehörende Antwort bereitgestellt wird.
Bei Antworten die Parameterwerte enthalten, antwortet der Slave bei der Wiederholung der Antworttelegramme immer **mit dem aktuellen Wert**.
- Bei Erstaufnahme der Kommunikation zwischen Master und Slave (Erstmaliges Ansprechen des Slaves) kann der Slave in der Übergangsphase, in der eine Antwortbereitstellung im Gerät erfolgt, nur mit der Kennung "Keine Antwort" antworten.
- Erhält der Auftraggeber von seinem angesprochenen Auftragnehmer keine zum Auftrag gehörende Antwortkennung, so ist im Auftraggeber für eine entsprechende Reaktion zu sorgen.
- Besitzt der Auftraggeber nicht die PKW-Bedienhoheit, so werden alle Änderung- und EEPROM-Aufträge vom Gerät nicht bearbeitet und mit der Kennung "Keine PKW- Bedienhoheit" beantwortet. Alle Lese-Aufträge werden bearbeitet.
- Der Auftragnehmer erwartet vom Auftraggeber keine Quittung ob die Antwort empfangen wurde.
- Erkennung der Antwort auf einen gestellten Auftrag im Master:
Der Auftraggeber erkennt die richtige Antwort im Antworttelegramm durch die Auswertung der Antwortkennung (Tabelle 4.2), der Parameternummer (PNU), gegebenenfalls durch den Wert in IND und dem Parameterwert.
- Erkennung eines Neuauftrages im Slave:
Jeder Auftrag, den der Auftraggeber nach Erhalt einer gültigen Antwort auf den alten Auftrag stellt, wird als Neuauftrag erkannt.
- Sendet der Master ein Broadcast-Telegramm, so dürfen die Slaves auf dieses Broadcast-Telegramm kein Antworttelegramm an den Master senden.

4.2.1.2. Spontanmeldung

In der PKE ist das Bit Nr. 11 als Toggle-Bit für die Bearbeitung von Spontanmeldungen definiert. Ein Slave sendet eine Spontanmeldung an den Master bei Änderung eines Parameterwertes (PWE), wenn diese Änderung nicht über die Schnittstelle erfolgt ist, über die der Master mit dem Slave kommuniziert.

Damit nicht jede Wertänderung eines Parameters zu einer Spontanmeldung führt, sind alle Parameter in ihrer Parameterbeschreibung (PBE) über ein Bit als Aktiv- oder als Passivparameter gekennzeichnet. Dieses Bit ist in der Beschreibung im Element "Kennzeichen" enthalten (siehe Bild 6.1). Nur Änderung des PWE bei Aktivparametern führt zu einer Spontanmeldung!

Bei einer Spontanmeldung läuft folgender Bearbeitungsmechanismus ab:

Der normale Auftrags- / Antwortablauf, wie in Kap. 4.2.1.1 beschrieben, wird vom Slave sofort unterbrochen, indem er im Antworttelegramm die Antwortkennung "Spontanmeldung" (Kennung 9, 10, 11 und 12 aus Tabelle 4.2) mit der entsprechenden **Parameter**nummer (PNU) und dem geänderten Parameterwert (PWE) sendet. Gleichzeitig wird vom Slave das Spontanmelde-Toggle-Bit (Nr. 11) geändert, hierdurch markiert der Slave, daß die vorliegende Spontanmeldung für den Master neu ist. Die Spontanmeldung wird solange an den Master gesendet, bis dieser durch ändern des Toggle-Bits (Bit 11) den sicheren Empfang der Spontanmeldung quittiert hat. Danach führt der Slave mit der **unterbrochenen Antwortbearbeitung** fort oder er überträgt die nächste anstehende Spontanmeldung. Das heißt, bei mehreren vorliegenden Spontanmeldungen, darf die nächste Spontanmeldung erst nach quittieren der vorherigen Spontanmeldung gesendet werden.

Ändern aller Parameterwerte in einem Array:

Wird über eine Schnittstelle mit einem Telegramm alle Werte eines Parameterarrays verändert (Index Low-Byte = 255) und sind an einer zweiten Schnittstelle nur feste Telegrammlängen möglich, so wird als Spontanmeldung über diese Schnittstelle nur die entsprechende Antwortkennung (AK 9 bis 12) die Parameternummer und der Index Low-Byte = 255 übertragen. Die geänderten Parameterwerte werden dabei **nicht** übertragen.

Beispiel:

Datenverkehr zwischen Master und Slave bei Vorliegen von zwei Spontanmeldungen (SPM 1, SPM 2), betrachtet wird **Bit 11** der Parameterkennung (in Klammern):

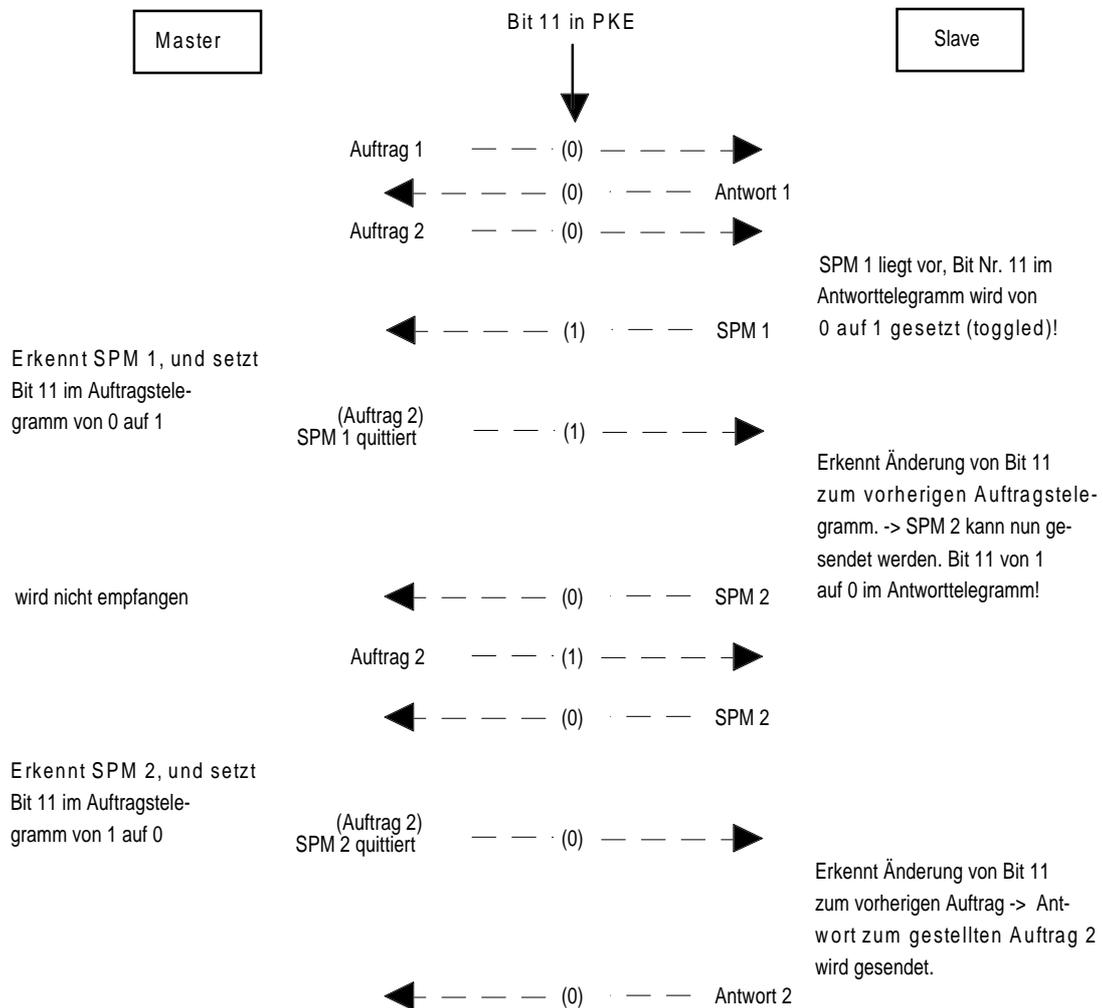


Bild 4.5: Datenfolge bei Spontanmeldungen

4.2.1.3 Parameternummer (PNU)

In den Bits 0 bis 10 ist in der Parameterkennung (PKE) die Parameternummer (PNU) enthalten. Die Zuordnung einer Funktion/Bedeutung zu einer PNU (z. B. die Ankerstrom-Nachstellzeit TN ist beim SIMOREG K 6RA24 unter der Parameternummer P156 abgelegt) ist von den Geräten abhängig und ist in den einschlägigen Betriebsanleitungen zu dokumentieren.

4.2.2. Index (IND)

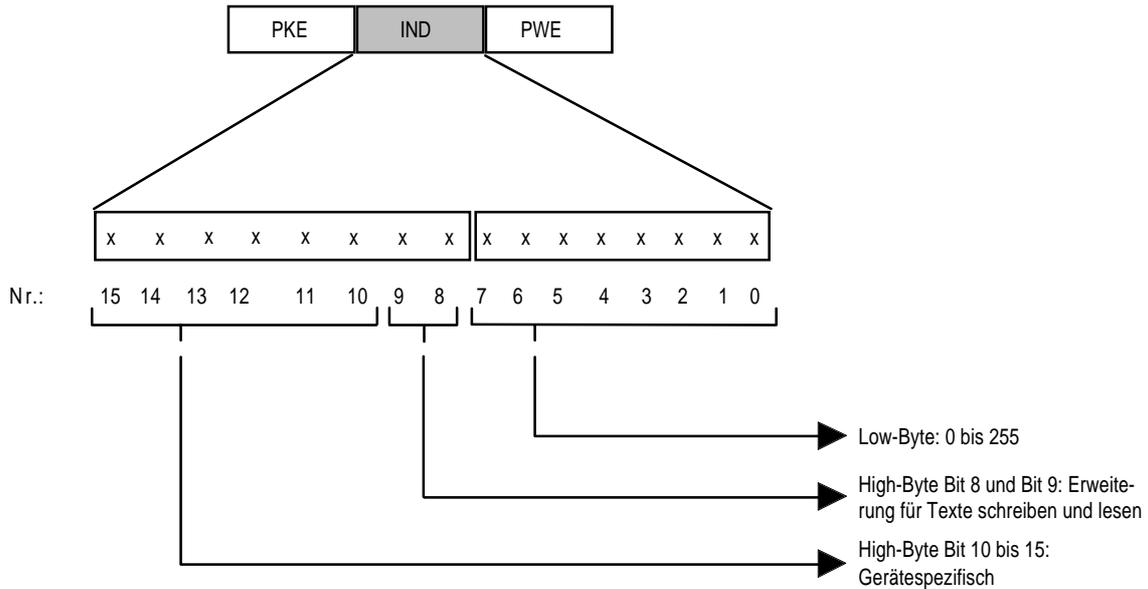


Bild 4.6: Aufbau des Index

Verwendung des IND bei der Auftrags- / Antwortkennung aus Kap. 4.2.1.1.

Der Index wird bei den folgenden Aufträgen benutzt:

- Schreiben und Lesen der Parameterbeschreibung
- Schreiben und Lesen von Werten, die in einem Array (= eindimensionales Feld) stehen.
- Schreiben und Lesen von Texten.

In allen anderen Fällen sind die Bits 0 bis 9 auf logisch "0" gesetzt.

Die gerätespezifischen Bits 10 bis 15 dürfen nur von den Geräten ausgewertet werden, für die diese Bits definiert sind. Bei Geräten, bei denen unter den Bits 10 bis 15 keine Funktionen hinterlegt ist, sind die Bits auf logisch 0 gesetzt. Der Master muß dies bei der Auftragsstellung an die entsprechenden Slaves berücksichtigen.

Auftragskennung (Master → Slave):

- Schreiben und Lesen der Parameterbeschreibung
- Schreiben und Lesen von Werten aus einem Array

Index Low-Byte: 0 - 254

PKE				IND	
Auftragskennung Bit: 15 14 13 12	Funktion	Low-Byte	Funktion		
0 1 0 0	PBE-Element anfordern	y (\leq 254)	y-te Elemente lesen		
0 1 0 1	PBE-Element ändern	y (\leq 254)	y-te Elemente ändern		
0 1 1 0	PWE anfordern (Array)	y (\leq 254)	Parameterwert von der y-ten Stelle des Array lesen		
0 1 1 1	PWE ändern (Array Wort)	y (\leq 254)	PWE in Wortgröße auf y-te Stelle im Array schreiben		
1 0 0 0	PWE ändern (Array Doppelwort)	y (\leq 254)	PWE in Doppelwortgröße auf y-te Stelle im Array schreiben		
1 1 0 0	PWE (Array Wort) ändern und ins EEPROM abspeichern	y (\leq 254)	PWE (Wort auf y-te Stelle im Array in EEPROM schreiben		
1 0 1 1	PWE (Array Doppelwort) ändern und ins EEPROM abspeichern	y (\leq 254)	PWE (Doppelwort) auf y-te Stelle im Array ins EEPROM schreiben		

Tabelle 4.4: Auftragskennung und Index bei indizierten Parametern

Index Low-Byte: 255

Der Index Low-Byte = 255 hat eine Sonderstellung. Bei diesem Indexwert bezieht sich der Auftrag auf das gesamte Array oder die gesamte Parameterbeschreibung.

Die fehlerfreie Bearbeitung eines solchen Auftrages setzt die Parameterierung auf variable Telegrammlänge (PKW_ANZ = 127) voraus.

Überschreitet die Länge (in Worten) einer Parameterbeschreibung oder eines Arrays die maximal mögliche Länge des PKW-Bereiches, siehe Kap. 4.1.2, so ist der Auftrag mit Antwortkennung 7 (Auftrag nicht ausführbar) und eine Fehlermeldung (\triangle Fehlernummer) zu beantworten.

Ebenso ist zu verfahren, wenn bei Low-Byte = 255 feste Telegrammlänge parametrier ist.

Auftragskennung (Master → Slave)

- Lesen und Schreiben eine Textes (△ Textelement)

Gilt nur für SIMOVERT Master Drives

Der Aufbau von Texten im Gerät erfolgt nach den Festlegungen aus dem PROFIBUS-Profil "Drehzahlveränderbare Antriebe" /1/. Zusätzlich ist bei SIMOVERT Master Drives eine Erweiterung gegenüber /1/ definiert worden, die es ermöglicht, daß zu einem indizierten Parameter (= Array) sowohl ein Textfeld mit Textelementen zu jedem Index, als auch ein Textfeld mit Textelementen zu jedem Parameterwert hinterlegt werden kann. Nach /1/ ist es nur möglich, daß zu einem Parameter immer nur ein Textfeld existieren kann. Um diese Textelemente schreiben oder lesen zu können, bzw. um auswählen zu können welches der beiden Textfelder angesprochen werden soll, wird das High-Byte im Index (IND) benutzt.

Die fehlerfreie Bearbeitung mit Auftragskennung 15 erfordert immer eine Parametrierung auf variable Telegrammlänge (PKW_ANZ = 127). Ist bei diesem gestellten Auftrag feste Telegrammlänge parametrierung, so muß ebenfalls mit Antwortkennung 7 und Fehlernummer geantwortet werden.

Index Low-Byte: 1 - 254 und Index High-Byte Bit 8 und 9

PKE				IND		
Auftragskennung Bit: 15 14 13 12	Funktion	Index High-Byte Bit-Nr.: 9 8	Index Low-Byte	Anmerkung		
1 1 1 1	Lesen des y-ten Textelementes in einem Textfeld, das zu Parametern vom Datentyp: - Array - Unsigned 8/16/32 - V2 existiert.	0 0	$1 \leq y \leq 254$	Die Ermittlung des y-ten Textelementes ist in /1/ Kap. 5.1.2.4 beschrieben.		
1 1 1 1	Schreiben des y-ten Textelementes im Textfeld, das zu den Datentypen: - Array - Unsigned 8/16/32 - V2 existiert.	0 1	$1 \leq y \leq 254$	"wie oben"		

Tabelle 4.5: Auftragskennung und Index bei Lesen und Schreiben von Texten

Inhalte der Textelemente bei den Parametern vom Datentyp:

- Array: Text der die Bedeutung der einzelnen Indizes des Array beschreibt.
- Unsigned 8/16/32: Text der die Bedeutung der einzelnen Parameterwerte dieser Parameter beschreibt.
- V2: Text der die Bedeutung der Bitwerte "0" und "1" jedes Bits im V2 Parameter beschreibt.

Index Low-Byte: 255

Die Sonderstellung Low-Byte = 255 ist ohne Bedeutung! Ein diesbezüglich gestellter Auftrag muß mit Antwortkennung 7 und Fehlernummer beantwortet werden.

Für **Parameter vom Datentyp Array** ist ein "zweites" zusätzliches Textfeld definiert, indem Textelemente hinterlegt sind, die die Bedeutung der einzelnen Parameterwerte beschreiben.

PKE				IND			
Auftragskennung Bit: 15 14 13 12				Index High-Byte Bit-Nr.: 9 8		Index Low-Byte	Anmerkung
1	1	1	1	1	0	$1 \leq y \leq 254$	y = Parameterwert +1
1	1	1	1	1	1	$1 \leq y \leq 254$	y = Parameterwert +1

Tabelle 4.6: Auftragskennung und Index bei Lesen und Schreiben der Textelemente eines Arrays

Auftragskennung (Master → Slave)

Index High-Byte Bit 10-15:

Diese Bits können gerätespezifisch benutzt werden und müssen in den jeweiligen Betriebsanleitungen eines Gerätes beschrieben sein. Bei Geräten, die diese Bits nicht verwenden, ist als Defaultwert "0" zu senden.

Gerätefamilie	Bedeutung der Bits 10-15
SIMOVERT P 6SE21	keine
Micromaster	keine
SIMOVERT P 6SE12 /6SE35 mit Kommunikationsbaugruppe C51/61	keine
SIMOVERT K 6RA24	keine
SIMOREG K 6RA23	keine
SIMOVERT Master Drives	keine
SIMOVERT PM	Adressierung der Einspeiseeinheit und der Wechselrichter
SIMOVERT P 6SE48	Kennliniennummer

Tabelle 4.7: Bedeutung der gerätespezifischen Bits 10 bis 15 im Index

Antwortkennung (Slave → Master):

PKE				IND	
Antwortkennung Bit: 15 14 13 12	Funktion	Index Low-Byte	Funktion		
0 0 1 1	PBE-Element übertragen	y (≤ 254)	y-te Element übertragen		
		255	gesamte PBE übertragen		
0 1 0 0	PWE übertragen (Array Wort)	y (≤ 254)	Übertragen des PWE, der an der y-ten Stelle in Array steht (Wortgröße)		
		255	gesamte Array übertragen (Wortgröße)		
0 1 0 1	PWE übertragen (Array Doppelwort)	y (≤ 254)	wie oben, nur PWE als Doppelwort		
		255	gesamte Array übertragen (Doppelwortgröße)		
1 0 1 1	Spontanmeldung (Array Wort)	y (≤ 254)	Übertragen eines geänderten PWE als Wortgröße durch Spontanmeldung, wobei PWE an der y-ten Stelle in einem Array steht.		
		255	Gesamtes Array übertragen, falls an dieser Schnittstelle variable Telegrammlänge aktiv.		
1 1 0 0	Spontanmeldung (Array Doppelwort)	y (≤ 254)	wie oben, nur PWE als Doppelwortgröße		
		255	Gesamtes Array übertragen, falls an dieser Schnittstelle variable Telegrammlänge aktiv.		

Tabelle 4.8: Antwortkennung und Index bei indizierten Parametern

Antwortkennung bei Textelement übertragen bei SIMOVERT Master Drives:

PKE				IND		
Antwortkennung Bit: 15 14 13 12	Funktion	Index High-Byte Bit-Nr.: 9 8	Index Low-Byte	Funktion		
1 1 1 1	Textelement übertragen	x x	1 ≤ y ≤ 254	Textelemente übertragen, gemäß den gestellten Auftrag		

Tabelle 4.9: Antwortkennung und Index bei Lesen und Schreiben von Texten

Anmerkung: x: 0 oder 1, je nach gestelltem Auftrag.

4.2.3. Parameterwert (PWE)

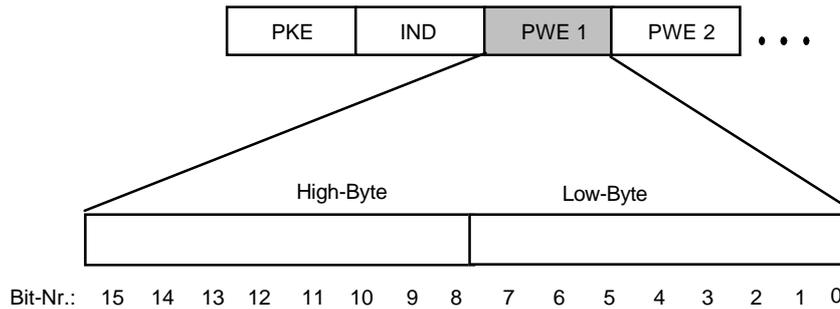


Bild 4.7: Aufbau Parameterwert

Die Besetzung von PWE ist abhängig vom gestellten Auftrag, respektive von der zugehörigen Antwort.

Auftragstelegramm (Master → Slave)

PKE				PWE	Länge des PKW-Bereichs im Nutzdatenblock		
Auftragskennung Bit: 15 14 13 12		Funktion	Inhalt	fest (minimal)	variabel notwendig		
0	0	0	0	kein Auftrag	$x \triangleq$ nicht belegt	3 Worte	-
0	0	0	1	PWE anfordern	x	3 Worte	
0	0	1	0	PWE ändern (Worte)	Wert	3 Worte	
0	0	1	1	PWE ändern (Doppelwort)	PWE 1 = High-Word PWE2 = Low-Word	4 Worte	
0	1	0	0	PBE-Element anfordern	x	3 Worte/4 Worte	Ja, falls
0	1	0	1	PBE-Element ändern	PBE-Element	3 Worte/4 Worte	Index Low-Byte = 255
0	1	1	0	PWE anfordern (Array)	x	3 Worte	Ja, falls
0	1	1	1	PWE ändern (Array Wort)	Wert	3 Worte	Index Low-Byte = 255
1	0	0	0	PWE ändern (Array Doppelwort)	PWE 1 = High-Word PWE 2 = Low-Word	4 Worte	Ja, falls Index Low-Byte = 255
1	0	0	1	Anzahl der Array-Elemente anfordern	x	3 Worte	-
1	1	0	0	geänderten Wert (Array Wort) in EEPROM abspeichern	Wert	3 Worte	Ja, falls Index Low-Byte = 255
1	1	0	1	geänderten Wert (Doppelwort) ins EEPROM abspeichern	PWE 1 = High-Word PWE 2 = Low-Word	4 Worte	-
1	1	1	0	geänderten Wert (Wort) ins EEPROM abspeichern	Wert	3 Worte	
1	1	1	1	Text anfordern oder ändern	- X = bei "anfordern" - Textelement: bei "ändern"	Nein	Ja

Tab. 4.10: Länge des Parameterblock im Auftragstelegramm

Antworttelegramm (Slave → Master)

PKE				PWE	Länge des PKW-Bereichs im Nutzdatenblock		
Antwortkennung Bit: 15 14 13 12				Funktion	Inhalt	fest (minimal)	variabel notwendig
0	0	0	0	keine Antwort	$x \hat{=}$ nicht belegt	3 Worte	-
0	0	0	1	PWE übertragen (Wort)	Wert	3 Worte	-
0	0	1	0	PWE übertragen (Doppelwort)	PWE 1 = High-Word PWE 2 = Low-Word	4 Worte	-
0	0	1	1	PBE-Element übertragen $IND \leq 254$	Element	3 Worte / 4 Worte	-
				$IND = 255$	Alle Elemente	Nein	Ja
0	1	0	0	PWE übertragen (Array Wort) $IND = y$	Wert (Wort) von der y-ten Stelle im Array	3 Worte	Ja, falls Index Low- Byte = 255
0	1	0	1	PWE übertragen (Array Doppelwort) $IND = y$	Wert (Doppelwort) von der y-ten Stelle im Array	4 Worte	Ja, falls Index Low- Byte = 255
0	1	1	0	Anzahl der Array-Elemente übertragen	Anzahl der Array- Elemente	3 Worte	-
0	1	1	1	Auftrag nicht ausführbar	Fehlernummer	3 Worte	-
1	0	0	0	Keine PKW-Bedienhoheit	x	3 Worte	-
1	0	0	1	Spontanmeldung (Wort)	Wert (Wort)	3 Worte	-
1	0	1	0	Spontanmeldung (Doppelwort)	Wert (Doppelwort)	4 Worte	-
1	0	1	1	Spontanmeldung (Array-Wort) $IND = y$	Wert (Wort) von der y-ten Stelle im Array	3 Worte	-
1	1	0	0	Spontanmeldung (Array Doppelwort) $IND = y$	Wert (Doppelwort) von der y-ten Stelle im Array	4 Worte	-
1	1	1	1	Text übertragen	Textelemente	Nein	Ja

Tab. 4.11: Länge des Parameterblock im Antworttelegramm

Hinweis:

Ist $PKW_ANZ = 4$ eingestellt, so werden Wortgrößen im Low-Word (PWE2) gesendet. Das High-Word (PWE1) ist dabei 0.

Ist $PKW_ANZ = 127$ (variable Länge) eingestellt, so werden Wortgrößen im PWE1 gesendet, Doppelwortgrößen in PWE1 (High-Word) und PWE2 (Low-Word).

5. PZD-Bereich

Der **Prozeßdaten**-(PZD)-Bereich ist unabhängig vom PKW-Bereich der zweite Teil im Nutzdatenblock.

5.1. Aufbau des PZD-Bereichs

Der Aufbau des PZD-Bereichs ist in der Reihenfolge seiner Elemente (1 Worten) immer gleich und unterscheidet sich von seinem Standardaufbau, nur durch die Anzahl der übertragenen Soll- / Istwerte.

Standardaufbau:



Bild 5.1: Aufbau des Prozeßdatenbereichs PZD

	PZD1	PZD2	PZD3 ... PZD16
Auftragstelegramm (Master → Slave)	Steuerwort	Hauptsollwert	Sollwerte / Zusatzsteuerworte
Antworttelegramm (Slave → Master)	Zustandswort	Hauptistwert ¹⁾	Istwerte ¹⁾ / Zusatz- zustandsworte

Tabelle 5.1: Aufbau des Prozeßdatenbereichs PZD

Anmerkung:

Die Bearbeitung der empfangenen PZD muß im Master und im Slave immer hochprior erfolgen. Die PZD-Verarbeitung hat Vorrang vor der PKW-Bearbeitung und überträgt immer die an der Schnittstelle "anliegenden", aktuellsten Daten.

- 1) Die Zuordnung Sollwert zu Istwert ist wahlfrei, d. h. zum Beispiel wird im Auftragstelegramm im PZD2 der Drehzahlsollwert übertragen, so kann im Antworttelegramm im PZD2 der Drehzahlwert zurückgemeldet werden (technologisch sinnvoll), oder aber auch ein anderer Istwert wie Momentenistwert, Lageistwert oder Stromistwert.

5.2. Beschreibung der einzelnen PZD-Elemente

5.2.1. Das Steuerwort und Zustandswort

Steuer- und Zustandswort sind identisch mit der Festlegung aus dem PROFIBUS-Profil "Drehzahlveränderbare Antriebe" /1/.

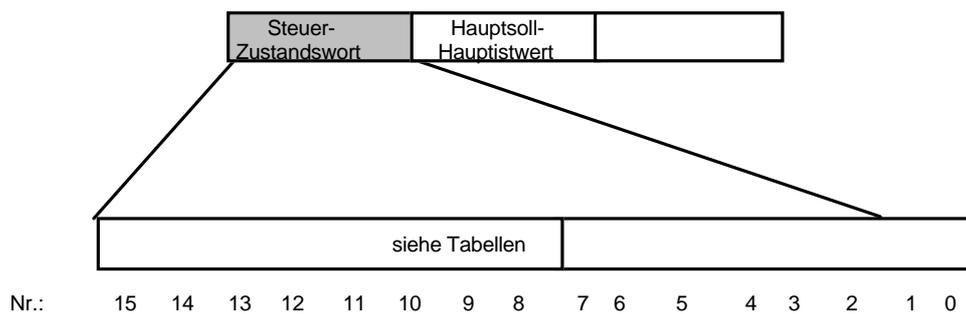


Bild 5.2: Aufbau Steuerwort und Zustandswort

Das Steuer- und Zustandswort, das über den "USS[®]-Bus" von einer übergeordneten Automatisierung vorgegeben wird, entspricht für die Bits 0 bis 10 genau den Festlegungen, die für die Übertragung über den PROFIBUS im Profil "Drehzahlveränderbare Antriebe" |2| definiert wurden. Die anderen, restlichen Bits sind gerätespezifisch belegt.

Steuerwort (Bit 0 bis Bit 2)

Bit	Wert	Bedeutung	Bemerkungen
0	1	EIN	<p>Betriebsbereit; Spannung an Stromrichter, d. h. Hauptschütz ein (wenn vorhanden); Feld wird aufgebaut; erfolgt innerhalb einer parametrierbaren Wartezeit (t_{wart}) kein Weiterschalten, wird in den Zustand "Einschaltsperr" geschaltet.</p> <p>$t_{\text{wart}} = 0 \dots 20 \text{ min}$, 20 min wird wie unendlich interpretiert!</p> <p>Gerätespezifisch:</p> <p>Variante 1: Feld wird aufgebaut (Stillstandserregung); Impulse gesperrt</p> <p>Variante 2: Zwischenkreis wird geladen; WR-Impulse gesperrt.</p> <p>Variante 3: GR- und WR-Impulse gesperrt, Kommutierkondensatoren werden nicht geladen oder nachgeladen.</p> <p>Variante 4: ER-, GR- und WR-Impulse gesperrt.</p>
	0	AUS 1	<p>Stillsetzen (abhängig von der Stellung der Steuerwortbits 0,1 und 2 zurück in den Zustand "Einschaltsperr", "Nicht Einschaltbereit" bzw. "Einschaltbereit"); Herunterfahren an der Hochfahrrampe (HLG-Rampe) oder an der UZK-Grenze; bei $n/f = 0$ und $I = 0$ Spannungsfreischaltung; Hauptschütz aus (wenn vorhanden).</p>
1	1	Betriebsbedingung	Alle "AUS 2" Befehle sind aufgehoben.
	0	AUS 2	<p>Spannungsfreischaltung:</p> <p>Gerätespezifisch:</p> <p>Variante 1: Impulse an Aussteuergränze α_{max} verschieben; bei $I = 0$ Impulse sperren.</p> <p>Variante 2: Impulse sperren.</p> <p>Variante 3: Impulse an Aussteuergränze α_{max} verschieben; bei $I = 0$; GR- und WR-Impulse sperren.</p> <p>Variante 4: Wie Variante 3, zusätzlich Erregung und Erregerschütz aus.</p> <p>Dann Hauptschütz aus (wenn vorhanden) und in Einschaltsperr; Motor trudelt aus.</p>
2	1	Betriebsbedingung	Alle "AUS 3" Befehle sind ausgehoben.
	0	AUS 3	<p>Schnellhalt, falls notwendig Betriebssperre aufheben, schnellstmögliches Herunterfahren z. B. an Stromgränze oder an UZK-Grenze bei $n/f = 0$; GR-Impulse sperren, dann Spannungsfreischaltung (Schütz aus) und in Einschaltsperr.</p>

Tabelle 5.2: Belegung Steuerwort-Bits 0 bis 2

Steuerwort (Bit 3 bis Bit 7)

Bit	Wert	Bedeutung	Bemerkungen
3	1	Betrieb freigeben	Freigabe Elektronik + Impulse Gerätespezifisch: Variante 1: Erregung freigeben. Variante 2: falls projektiert, wird Entregungszeit abgewartet, dann WR-Impulse freigeben und Erregerstrom einprägen. Variante 3: falls projektiert, wird Entregungszeit abgewartet, dann GR- und WR-Impulse freigeben. Kommutierungskondensatoren vorladen. Variante 4: Impulse freigeben für Ortung, dann GR- und WR-Impulse freigeben. Dann Hochlauf auf anliegenden Sollwert.
	0	Betrieb sperren	Gerätespezifisch: Variante 1: Impulse an Aussteuergrenze α_{max} verschieben; bei $I = 0$ Impulse sperren und Erregung auf Stillstandserregung. Variante 2: WR-Impulse sperren. Variante 3: GR-Impulse an Aussteuergrenze α_{max} erschieben; bei $I = 0$ GR- und WR- Impulse sperren. Variante 4: GR-Impulse an Aussteuergrenze α_{max} verschieben; bei $I = 0$ GR-, WR- und ER-Impulse löschen. Antrieb trudelt aus (Hochlaufgeber auf Null oder Nachführung) und geht in Zustand "Betriebsbereit" (s. Steuerwort Bit 0).
4	1	Betriebsbedingung	
	0	Hochlaufgeber sperren	Ausgang HLG wird auf 0 gesetzt. Hauptschütz bleibt ein, Stromrichter wird nicht vom Netz getrennt, Antrieb fährt an Stromgrenze oder an UZK-Grenze herunter.
5	1	Hochlaufgeber freigeben	
	0	Hochlaufgeber stoppen	Einfrieren des aktuellen vom HLG vorgegebenen Sollwertes.
6	1	Sollwert freigeben	Angewählter Wert am Eingang des HLG wird eingeschaltet.
	0	Sollwert sperren	Angewählter Wert am Eingang des HLG wird zu 0 gesetzt.
7	1	Quittieren	Sammelmeldung wird bei positiver Flanke quittiert; Umrichter befindet sich bis Fehler erfolgreich behoben wurde, in "Störung" geht anschließend in "Einschaltsperr"
	0	keine Bedeutung	

Tabelle 5.3: Belegung Steuerwort-Bits 3 bis 7

Steuerwort (Bit 8 bis Bit 15)

Bit	Wert	Bedeutung	Bemerkungen
8 ¹⁾	1	Tippen 1 EIN	Voraussetzung: Betrieb ist freigegeben und n (soll) = 0 Antrieb fährt so schnell wie möglich auf Tipp-Sollwert 1
	0	Tippen 1 AUS	Antrieb bremst so schnell wie möglich, wenn vorher "Tippen 1" EIN war und geht bei n/f = 0 und l = 0 in "Betrieb frei"
9 ¹⁾	1	Tippen 2 EIN	Voraussetzung: Betrieb ist freigegeben und n (soll) = 0 Antrieb fährt so schnell wie möglich auf Tipp-Sollwert 2
	0	Tippen 2 AUS	Antrieb bremst so schnell wie möglich, wenn vorher "Tippen 2" EIN war und geht bei n/f = 0 und l = 0 in "Betrieb frei"
10	1	Führung vom AG	Führung über Schnittstelle, Prozeßdaten gültig
	0	Keine Führung	Prozeßdaten ungültig, d. h., die "alten" Prozeßdaten bleiben erhalten
11-15		Gerätespezifisch	Bedeutung nicht vorgegeben

Tabelle 5.4: Belegung Steuerwort-Bits 8 bis 15

1) Die Belegung von Bit 8 und Bit 9 mit Tipp-Funktion sind optional.

Erläuterungen:

AG	Automatisierungsgerät	SM	Synchronmaschine
ASM	Asynchronmaschine	SPM	Spontanmeldungen
ER	Erregung	UZK	Zwischenkreisspannung
GR	Gleichrichter	WR	Wechselrichter
HLG	Hochlaufgeber		
α_{\max}	max. Steuerwinkel		
Variante 1:	Gleichstromgerät		
Variante 2:	Pulsumrichter mit Spannungszwischenkreis		
Variante 3:	Stromzwischenkreisumrichter mit ASM		
Variante 4:	Stromzwischenkreisumrichter mit SM (Stromrichteromotor)		

Zustandswort (Bit 0 bis Bit 7)

Bit	Wert	Bedeutung	Bemerkungen
0	1	Einschaltbereit	Stromversorgung eingeschaltet, Elektronik initialisiert, Hauptschütz ggf. abgefallen, Impulse gesperrt
	0	Nicht einschaltbereit	
1	1	Betriebsbereit	Siehe Steuerwort Bit 0
	0	Nicht betriebsbereit	
2	1	Betrieb freigegeben	Siehe Steuerwort Bit 3
	0	Betrieb gesperrt	
3	1	Störung	Antrieb gestört und dadurch außer Betrieb, geht nach Quittierung und erfolgreicher Fehlerbehebung in Einschaltsperrre. Fehlernum- mern im Störungsparameter
	0	Störungsfrei	
4	1	Kein AUS 2	
	0	AUS 2	"AUS 2" Befehl steht an
5	1	Kein AUS 3	
	0	AUS 3	"AUS 3" Befehl steht an
6	1	Einschaltsperrre	Wiedereinschalten nur durch "AUS 1" und anschließendem "EIN"
	0	Kein Einschaltsperrre	
7	1	Warnung	Antrieb weiter in Betrieb; Warnung im Wartungsparameter; keine Quittierung
	0	Keine Warnung	Es liegt keine Warnung an bzw. Warnung ist wieder verschwunden.

Tabelle 5.5: Belegung Zustandswort-Bits 0 bis 7

Zustandswort (Bit 8 bis Bit 15)

Bit	Wert	Bedeutung	Bemerkungen
8	1	Soll- / Ist-Überwachung im Toleranzbereich	Istwert innerhalb eines Toleranzbandes; dynamische Über- oder Unterschreitungen für $t < t_{\max}$ zulässig z. B. $n = n_{\text{SOLL}} \pm n$, $f = f_{\text{SOLL}} \pm f$, usw. t_{\max} ist parametrierbar
	0	Desgleichen, nicht im Toleranzbereich	
9	1	Führung gefordert	Das Automatisierungssystem wird aufgefordert, die Führung zu übernehmen.
	0	Betrieb vor Ort	Führung nur am Gerät möglich
10	1	f oder n erreicht	Istwert \geq Vergleichswert (Sollwert), der über Parameternummer einstellbar ist
	0	f/n unterschritten	Istwert $<$ Vergleichswert
11-15		Gerätespezifisch	Bedeutung nicht vorgegeben.

Tabelle 5.6: Belegung Zustandswort-Bits 8 bis 15

5.2.2. Soll- / Istwerte

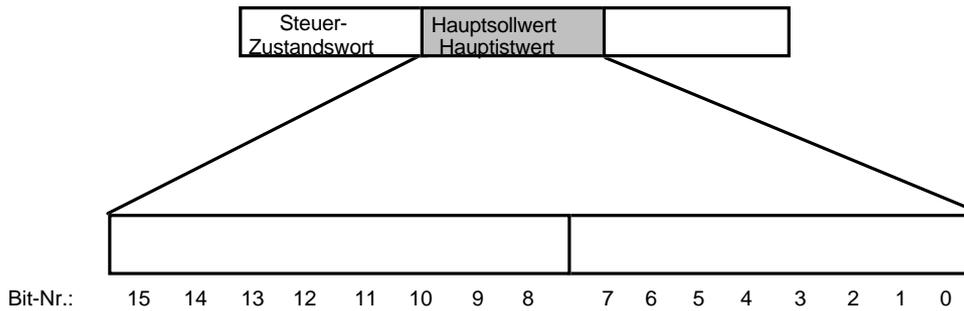


Bild 5.3: Aufbau Hauptsollwert und Hauptistwert

Übertragung von normierten Soll- und Istwerten. Die Normierung ist von der Bedeutung des Wertes und dem jeweiligen Gerätetyp abhängig ist.

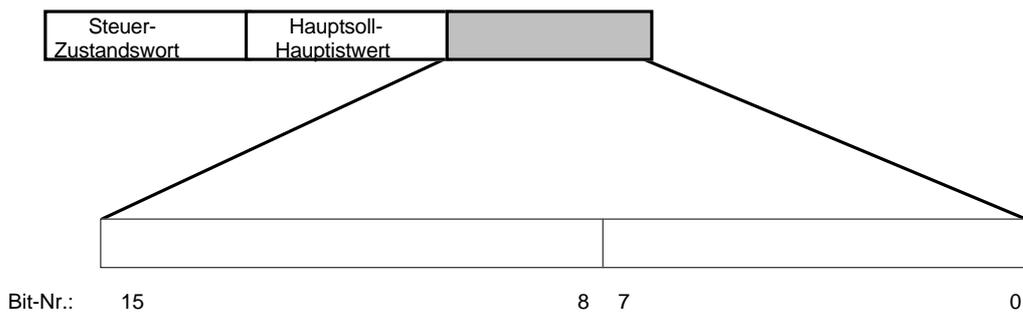


Bild 5.4: Anordnung zusätzlicher Soll- und Istwerte, sowie gegebenenfalls zusätzliche Steuer- und Zustandsworte.

5.2.3. Broadcast-Mechanismus

Für SIMOVERT Master Drives wurde für die Übertragung von Prozeßdaten ein Mechanismus definiert, der es dem Master erlaubt, mit einem Telegramm Steuerwort(e) und Sollwerte für alle am Bus angeschlossenen Antriebe gleichzeitig zu übertragen. Dabei ist es durch Maskierung möglich festzulegen, welche Sollwerte und welche Steuerwortbits tatsächlich mit dem Broadcasttelegramm beeinflusst werden sollen. Die Maskierung wird im jeweiligen Broadcasttelegramm immer mitgesendet. Hierzu wird der PKW-Bereich mit einer festen Länge von 4 Worten benutzt. Das heißt, beim Broadcasttelegramm ist eine "normale" PKW-Bearbeitung nicht möglich. Eine ausführliche Beschreibung des Broadcast-Mechanismus ist im Anhang beigefügt.

6. Übertragungsformat der Nutzdaten

Bei Nutzdaten, die aus mehr als einem Byte bestehen, wird bei der Übertragung über den Bus der höherwertigere Teil immer zuerst übertragen. Diese Festlegung ist identisch mit der Übertragung von Nutzdaten über den PROFIBUS. Im einzelnen gilt:

Übertragung von Wortgrößen:

Es wird High-Byte vor Low-Byte übertragen!

Dies gilt für alle 16-bit-großen Datentypen: z.B. V2, Unsigned 16, Integer 16, etc.(siehe /1/)

Übertragung von Doppelwortgrößen:

Es wird High-Word vor Low-Word übertragen! Für die Übertragung von High- und Low-Word gilt die Übertragungsregel der Wortgrößen.

Dies gilt für alle 32-bit-großen Datentypen: z. B. Unsigned 32, Integer 32, Floating Point, etc. (siehe /1/)

Übertragung von Bytegrößen:

Hier gibt es keine Reihenfolge! Übertragung wie die Daten im "internen Speicher" des Gerätes abgelegt sind.

Dies gilt für Daten vom Typ: Byte-String (= Octet String gemäß /1/)

Übertragung von Texten:

Texte setzen sich aus einzelnen Zeichen (= Charakters) zusammen. Jedes Zeichen hat die Größe eines Bytes. Für die Übertragung der einzelnen Zeichen gibt es keine Reihenfolge! Die Übertragung der Zeichen erfolgt so, wie diese im "internen Speicher" des Gerätes abgelegt sind.

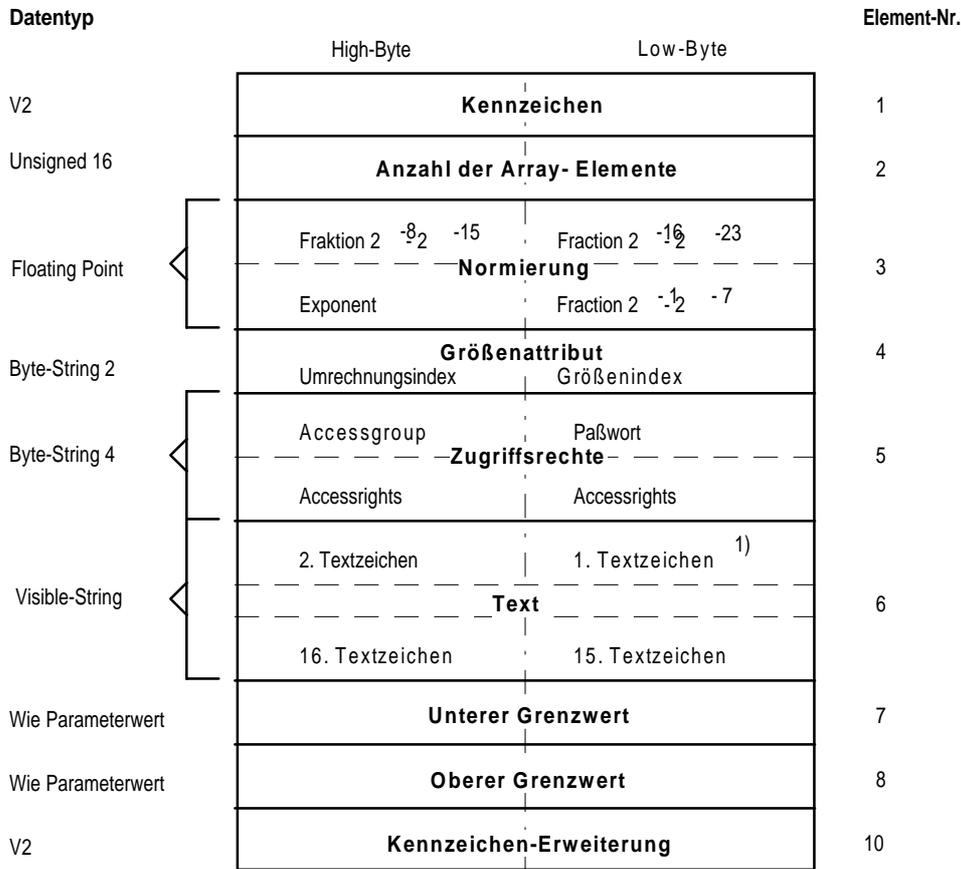
Der Datentyp für Textzeichen ist Visible-String, siehe /1/.

Die Übertragungsvorschrift wird im folgenden Beispiel anhand der Parameterbeschreibung erläutert.

Gemäß dem PROFIBUS-Profil /1/ gehört zu jedem Parameter eines Gerätes eine Parameterbeschreibung. Die Parameterbeschreibung selbst besteht aus mehreren Elementen, z.B. dem Kennzeichen, der Normierung etc.

Je nach Ausbaugrad eines Gerätes ist die Parameterbeschreibung vollständig oder nur teilweise vorhanden. In /1/ wird Aufbau, Umfang und Bedeutung der Parameterbeschreibung ausführlich beschrieben.

Im Bild 6.1 ist die vollständige Parameterbeschreibung eines Parameters dargestellt, so wie diese im "internen Speicher" des Gerätes abgelegt ist. Das hier gewählte Beispiel zeigt die Ablage bei SIMOVERT Master Drives. Die Abspeicherung im Gerät erfolgt nach dem "Intel"-Format, das heißt, daß zum Beispiel bei einem Wort vom Datentyp "unsigned 16" das niederwertigste Byte auf der niederwertigeren Adresse abgelegt ist.



1) 1. Textzeichen von links auf einem Display!

Bild 6.1: Ablage der gesamten Parameterbeschreibung im "internen Speicher" bei SIMOVERT Master Drives

Die einzelnen **Elemente** der Parameterbeschreibung sind im Bild 6.1 "fett"-gedruckt. Der entsprechende Datentyp des jeweiligen Elementes ist auf der linken Seite dargestellt. Die Element-Nummer in der Parameterbeschreibung auf der rechten Seite dargestellt.

Bild 6.2 zeigt die Übertragung der Parameterbeschreibung über den Bus.

Die Parameterbeschreibung kann mit dem Auftrag "PBE anfordern", siehe Tabelle 4.1 in Kapitel 4.2.1.1 vom Master gelesen werden. Um die gesamte Parameterbeschreibung zu lesen muß im IND das Low-Byte = 255 gesetzt sein, siehe Kap. 4.2.2. Um nur ein Element der Parameterbeschreibung zu lesen, muß die Nummer des Elements im Low-Byte von IND gesetzt sein. Das Element "Kennzeichen" ist Nummer 1, das Element "Anzahl der Array-Elemente" ist Nummer 2, ..., Nummer 9 ist reserviert, das Element "Kennzeichen-Erweiterung" ist Nummer 10 (siehe auch Kap. 8, Beispiel 4). Der Slave überträgt die Parameterbeschreibung im PKW-Bereich des Nutzdatentelegramms. Ist die Länge des PKW-Bereichs (in Worten) kleiner definiert (siehe Kap. 3) als die Länge des zu übertragenden Elements, wird der Auftrag vom Slave mit der negativen Antwortkennung "Auftrag nicht ausführbar" (Tabelle 4.2) beantwortet. Bild 6.2 zeigt ausschließlich die Parameterbeschreibung, die anderen Teile der Nutzdaten werden nicht dargestellt.

Die zeitliche Reihenfolge der Übertragung ist in Bild 6.2 von oben nach unten dargestellt. Das heißt, zuerst wird das High-Byte des Kennzeichens übertragen. Danach folgt das Low-Byte des Kennzeichens u.s.w.

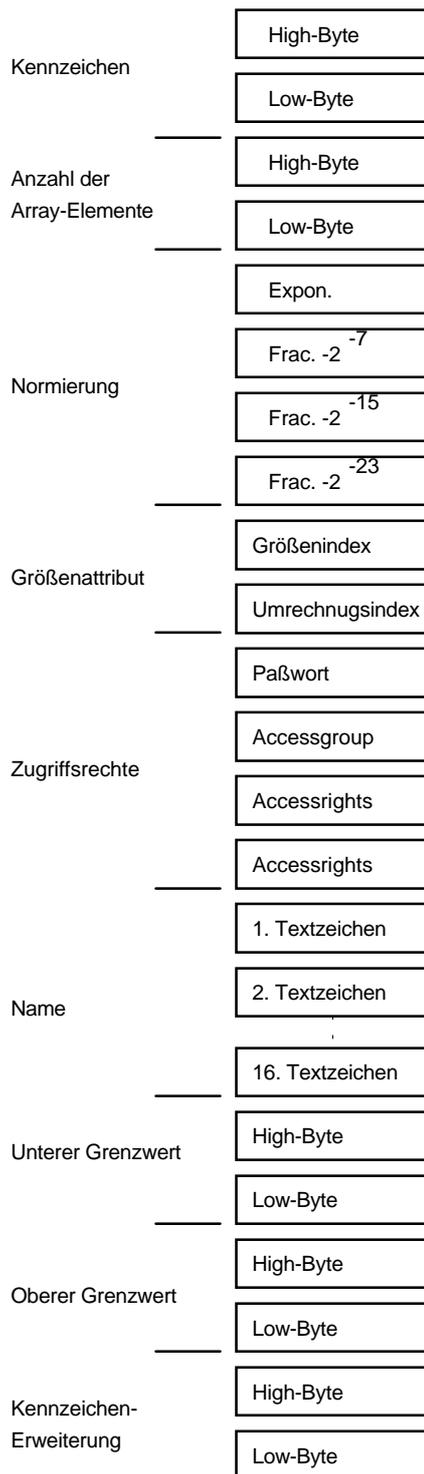


Bild 6.2: Übertragungsreihenfolge der Parameterbeschreibung über den Bus

7. Projektierung des Protokolls am Bussystem

Wie in der Einleitung bereits erwähnt, besteht die Möglichkeit, die Kommunikation zwischen Master und Slaves mit fester oder variabler Telegrammlänge zu konfigurieren.

Feste Telegrammlänge

Die Projektierung der Kommunikation mit dem USS[®]-Protokoll bei fester Telegrammlänge bedeutet:

- Bei der Kommunikation zwischen Master und einem Slave gilt:
Auftrags- und Antworttelegramme haben die selbe Länge, d. h., gleiche Länge bezüglich des PKW-Bereiches und des PZD-Bereiches.
- Diese Länge muß vor Erstinbetriebnahme des Bussystems fest eingestellt werden und darf während des Betriebes nicht mehr verändert werden.
- Feste Telegrammlänge bedeutet feste Größe des Nutzdatenblocks.
- Die Einstellung der Größe des Nutzdatenblocks erfolgt über zwei Parameter, siehe Kap. 3.1
Die Größe des PKW-Bereichs (in Worten) wird eingestellt über den Parameter "PKW_ANZ", z. B. ist PKW_ANZ = 3 gesetzt, wird der PKW-Bereich im Nutzdatenblock immer 3 Worte umfassen. Entsprechend wird die Größe des PZD-Bereichs (in Worten) über den Parameter PZD_ANZ eingestellt. Ist z. B. PZD_ANZ = 2 so umfaßt der PZD-Bereich 2 Worte im Nutzdatenblock.
- Wird vom Master ein Auftrag erstellt, der eine Antwort zur Folge hätte, die über die eingestellte Größe des PKW-Bereiches hinausgeht, muß dieser Auftrag mit der Antwortkennung "Auftrag nicht ausführbar" beantwortet werden.

Beispiel: Der Auftrag "PWE anfordern (Doppelwort)" kann bei
PKW_ANZ = 3 nicht ausgeführt werden. In diesem Fall muß
PKW_ANZ= 4 gesetzt sein.
- Vor der Einstellung der Größe des Nutzdatenblocks, muß festgelegt werden, welche Aufträge von Master gestellt werden sollen. Darauf basierend muß die Größe des PKW-Bereichs festgelegt werden. Das heißt, ist eine Doppelwortbearbeitung vorgesehen, dann muß vor Erstinbetriebnahme der PKW-Bereich grundsätzlich auf 4 Worte gesetzt sein, auch wenn die Wortbearbeitung überwiegt.

Variable Telegrammlänge

Ein Datenverkehr zwischen Master und Slave mit variabler Telegrammlänge bedeutet:

- Variable Telegrammlänge vom Master zum Slave (Auftragstelegramm) und
- Variable Telegrammlänge vom Slave zum Master (Antworttelegramm).

Hierbei gelten folgende Bedingungen:

Variable Telegrammlänge = variable Länge des PKW-Bereichs (PKW-ANZ = 127)

- Der PZD-Bereich (Prozeßdaten) muß für das Auftrags- und für das Antworttelegramm immer gleich groß sein. Das heißt, der Slave erwartet und sendet unabhängig von der getroffenen Parametrierung des PKW-Bereichs die Anzahl der im Parameter PZD_ANZ festgelegten Prozeßdaten.
- Im Slave muß bei Parametrierung auf variable Telegrammlänge immer das "Längen-Byte" LGE im Telegrammrahmen ausgewertet werden. Mit dieser Information, und mit der festen Parametrierung der Prozeßdaten (Parameter PZD_ANZ) ist die Länge des empfangenen, variablen Auftragstelegramms eindeutig bestimmbar.
- Ein entsprechendes Programm auf Masterseite muß ebenfalls so ausgelegt sein, daß variable Antworttelegramme von Slave erkannt und fehlerfrei ausgewertet werden können. Auf Grund der gestellten Aufträge sowie über die Kenntnis von der Einstellung der Parameter der einzelnen Slaves ist dem Master bekannt ob der angesprochene Slave mit einem variablen Telegramm antwortet.
- Von der Projektierung her ist es möglich, daß Slaves, die mit fester und Slaves, die mit variabler Telegrammlänge parametrierung sind über den gleichen Bus mit einem Master kommunizieren können. Dies kann jedoch im Master zu einem erhöhten Software-Aufwand führen.
- Struktur des PKW-Bereichs
Bei variabler Telegrammlänge werden immer nur die tatsächlich für den Auftrag bzw. Antwort notwendigen Daten übertragen.
Beispiel:
Master: Auftrag Textelement y anfordern →
PKW-Bereich: PKE, IND, PWE 1 (Minimum 3 Worte)
Slave: Antwort Textelement y (16 Zeichen) übertragen
PKW-Bereich: PKE, IND, PWE 1, PWE 2... PWE 8
- Überwachung der Telegrammausfälle
Die Einstellung einer optimalen Telegrammausfallzeit im Slave ist bei Projektierung von variablen Telegrammlängen schwierig. Dies wird um so kritischer je vielfältiger die möglichen Auftragsvarianten vom Master an die Slaves sein können.
Im worst case-Fall (niedrige Baudrate, hohe Teilnehmerzahl, lange variable Telegramme) muß die Telegrammausfallzeit in den Slaves inaktiv gesetzt werden.

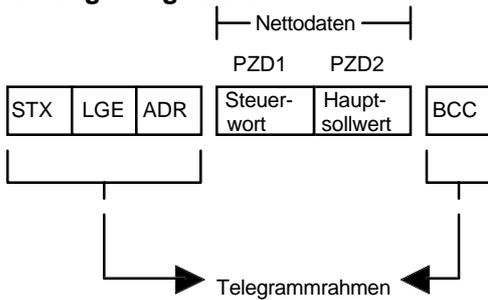
8. Beispiele

8.1. Feste Telegrammlänge

Beispiel 1: Übertragung von zwei Worten Prozeßdaten (Steuerwort / Zustandswort, Sollwert / Istwert)

- Parametrierung
PKW_ANZ = 0
PZD_ANZ = 2
- Auftrag für PKW-Schnittstelle nicht möglich
- Nur PZD-Bereich mit Steuerwort / Zustandswort und einem Soll- / Istwert im Telegramm

Auftragstelegramm:



Antworttelegramm:

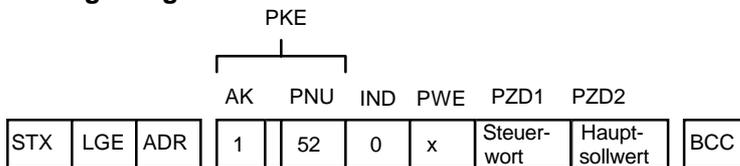


Bild 8.1: Telegrammaufbau zu Beispiel 1

Beispiel 2: Übertragung von einem Parameter (Wortgröße) und 2 Worten Prozeßdaten

- Parametrierung
PKW_ANZ = 3
PZD_ANZ = 2
- Auftrag: Wert von Parameter-Nr. 52 (dezimal) lesen; Wert = 4000 Wortgröße als Hex-Wert)
- Ständige Übertragung von Steuer-/ und Zustandswort und Hauptsoll- / Istwert.

Auftragstelegramm:



x $\hat{=}$ unbesetzt, nicht relevant!

Antworttelegramm:

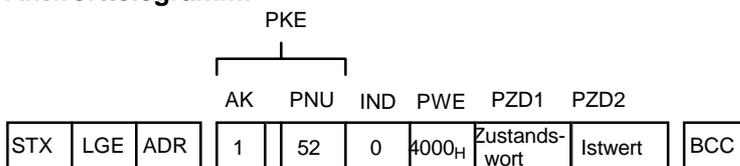
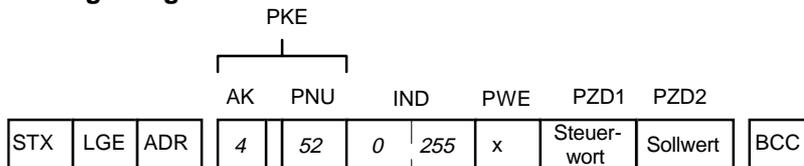
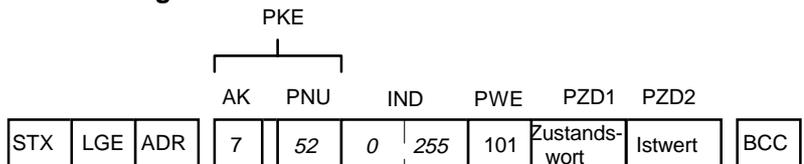


Bild 8.2: Telegrammaufbau zu Beispiel 2

Beispiel 3: Fehlerhafter Parameter-Lese-Auftrag

Wie Beispiel 2, jedoch

- Auftrag: gesamte Parameterbeschreibung zum Parameter P 52 lesen

Auftragstelegramm:**Antworttelegramm:****Fehlerkennung**

Z. B. geräteabhängige Fehlernummer:

Mit fester Telegrammlänge
nicht möglichAnmerkung: Diese Fehlernummer
muß z. B. im Gerät implementiert
werden; ist keine Standardfehler-
codierung)**Antwortkennung:**

Auftrag nicht ausführbar.

Bild 8.3: Telegrammaufbau zu Beispiel 3

Beispiel 4: Lesen eines Elements aus der Parameterbeschreibung

Wie Beispiel 2, jedoch

- Auftrag: Element "Unterer Grenzwert" (Element-Nr. 7) aus der Parameterbeschreibung zum Parameter P52 lesen. Unterer Grenzwert = 5FFF_H.

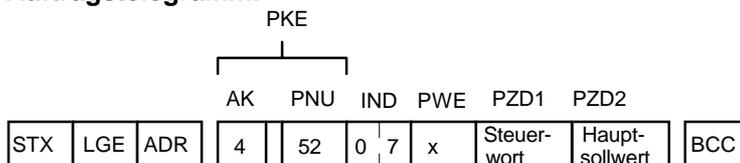
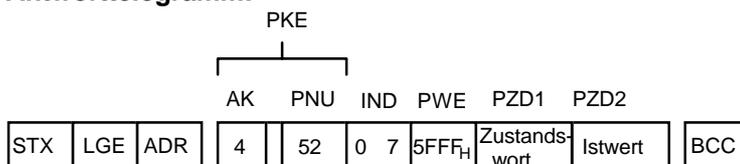
Auftragstelegramm:x \triangleq unbesetzt, nicht relevant!**Antworttelegramm:**

Bild 8.4: Telegrammaufbau zu Beispiel 4

Beispiel 5: Schreiben eines Doppelwortparameters in einem Parameterfeld

- Feste Telegrammlänge:
PKW_ANZ = 4
PZD_ANZ = 0
- Auftrag: Schreiben eines Wertes (Doppelwort) = 4000 0000 (Hex) auf Parameter-Nr. 4 an die 2-te Stelle des unter PNU= 4 hinterlegt eindimensionalen Feldes (= Array).
- Keine PZD-Daten im Telegramm

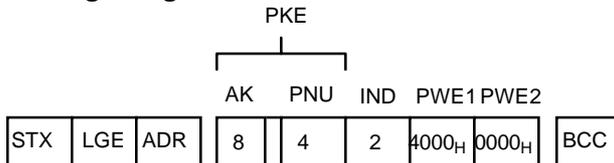
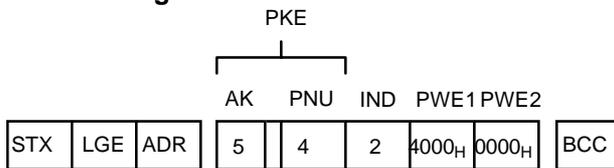
Auftragstelegramm:**Antworttelegramm:**

Bild 8.5: Telegrammaufbau zu Beispiel 5

Beispiel 6: Parameter in Wortgröße bei 4 Worten PKW

wie Beispiel 5, jedoch

- Auftrag: Schreiben eines Wertes (Wortgröße) = 7000 (Hex) auf Parameter-Nr. 18.

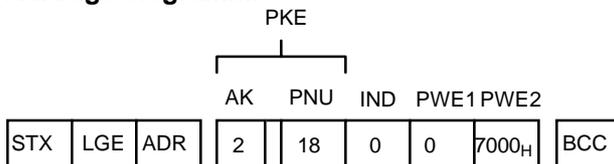
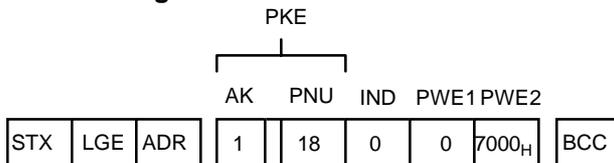
Auftragstelegramm:**Antworttelegramm:**

Bild 8.6: Telegrammaufbau zu Beispiel 6

8.2. Variable Telegrammlänge

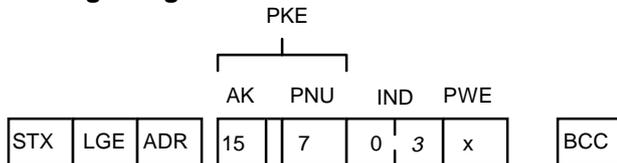
Beispiele für SIMOVERT Master Drives

Beispiel 7: Text aus einem Text-Array lesen das zum Parameter 7 gehört.

P 7 ist vom Datentyp "unsigned 16" ($\hat{=}$ 02) und hat einen Wertebereich von 1 bis 10. P 7 hat ein Textfeld indem jedem Wert des Parameters ein "16 Zeichen"-langes Textelement zugeordnet ist. Index des Textfeldes = Parameterwert + 1 (Bildungsgesetz nach /1/).

- Parametrierung
Slave: PKW_ANZ = 127
PZD_ANZ = 0
- Auftrag Textelement zum Parameterwert 2 des P7 lesen
3. Textelement: 300 BAUD
(Nach /1/ immer 16 Zeichen groß)

Auftragstelegramm:



Antworttelegramm:

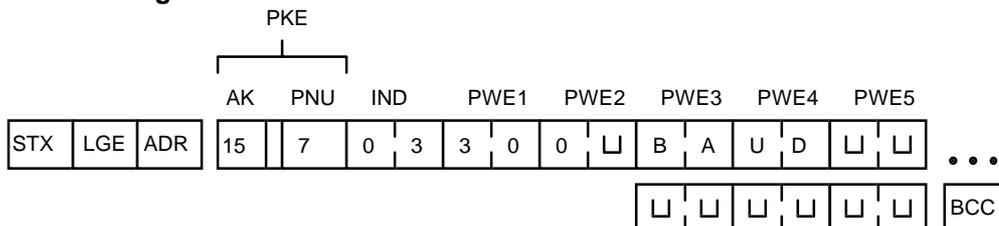
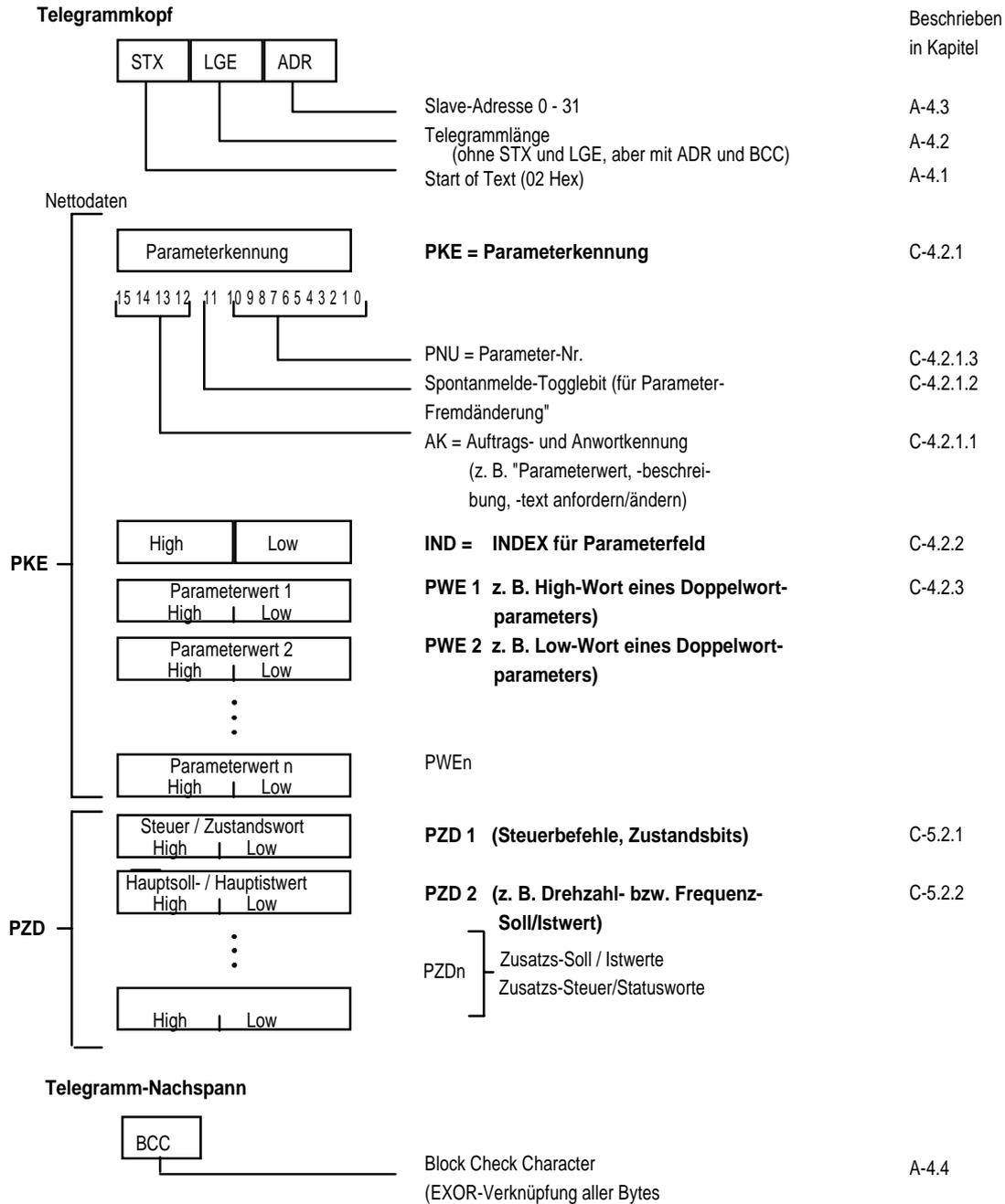


Bild 8.7: Telegrammaufbau zu Beispiel 7

ANHANG

Übersicht: Aufbau des Telegramms beim USS[®]-Protokoll



The Optional Broadcast Mechanism of the USS-Protocol

The Broadcast Mechanism is not obligatory for the USS slave interface. But if a Broadcast Mechanism is implemented for a slave communication interface, it must be realized according to this specification.

Whenever the term "Broadcast Telegram" is used in this document, it describes a telegram that has to be accepted by all drives connected via a Bus configuration; all drives fetch exactly the same identical net data from this telegram. A Broadcast Telegram has the same telegram length as a normal individual telegram. (A different type of Broadcast Telegram is defined in the PROFIBUS standard: A PROFIBUS Broadcast Telegram is a "multiple length" telegram whereby the individual information blocks required for all drives connected to the bus are contained in one extra long "group-telegram".)

It is a typical application of a Broadcast Telegram to enable the ramp generators of all drives at the same point of time by means of only one telegram in order to let all drives ramp up simultaneously.

Selective Reaction to a Broadcast Telegram

According to the experience with other protocols, a Broadcast Telegram from an external communication partner only makes sense if a 'selective reaction' to a Broadcast Telegram is made possible. This means that the drives must be told in the net data what REF VALUES and what Command Bits should be influenced by the Broadcast Telegram and what PROCESS DATA (PZD) should not be affected (the latter mentioned data must be dropped and instead the signals transmitted in the last "Non-Broadcast Telegram" remain effective).

The following construction is specified for this purpose: A Broadcast Telegram is marked by a "1" in bit 5 of the Address-Byte in the telegram header. In case of a Broadcast Telegram the PARAMETER DATA area (PKW) is "misused" for transmitting the information of how to react to the Broadcast Telegram instead of transmitting the normal PARAMETER DATA.

If the master station issues a Broadcast Telegram it is not allowed to enter a parameter into the PARAMETER DATA area - as it does with normal telegrams - but instead it has to enter a 4 Words "BROADCAST ENABLE ARRAY" into this space (refer to Fig. 2.5.1.6). This does not cause any problems because reading and writing parameters via a Broadcast Telegram is nonsense since: Broadcast Telegrams, as a general principle, are never answered by the drives.

The BROADCAST ENABLE ARRAY contains Enable-Bits for each of the PROCESS DATA WORDS 1 to 15 and for each of the max. possible 48 Command Bits in the Broadcast Telegram. If a PROCESS data block originates from a Broadcast Telegram, then a Command Bit or a REF VALUE is only accepted by the drive if the related Enable Bits are set to 1. If for a Command Bit or a REF VALUE the related bits in the BROADCAST ENABLE ARRAY are reset, the Command Bit or REF VALUE remains in the old state which has been set by the last Non-Broadcast-Telegram. The rules for processing the PROCESS DATA WORDS of a Broadcast Telegram according to the Bits in the BROADCAST ENABLE ARRAY are described below.

Rules for processing the PROCESS DATA WORDS of a Broadcast Telegram

The 16 PROCESS DATA WORDS 1 to 16 can be classified into the following 3 groups:

- PROCESS DATA WORDS 2, 3 and 6 to 15:

These PROCESS DATA WORDS can contain REF VALUES only ; they cannot act as COMMAND WORDS. A PROCESS DATA WORD of this group is only picked up by the LOWER SIDE BOARD from a Broadcast Telegram if the corresponding enable bit in Word a of the BROADCAST ENABLE ARRAY is set to "1".

- PROCESS DATA WORDS 1, 4 and 5:

These PROCESS DATA WORDS can act as COMMAND WORDS: WORD 1 is always a COMMAND WORD, according to the PROFIBUS-Profile (Lit. [2]). Words 4 and 5 may either be COMMAND WORDS or REF VALUES, e.g. depending on the software type of the TECH BOARD.

For each of the 48 bits of these PROCESS DATA WORDS there is a corresponding bit in the words b, c and d of the BROADCAST ENABLE ARRAY.

A PROCESS DATA Bit in the PROCESS DATA WORDS 1, 4 and 5 is only taken over from a Broadcast Telegram if

- I) the whole PROCESS DATA WORD is enabled via a "1" Information in the corresponding bit in Word a and additionally
- II) the PROCESS DATA Bit itself is enabled by the corresponding bit of word b, c or d respectively.

As a general rule it is prescribed that in case of a Broadcast Telegram the BROADCAST ENABLE WORDS b, c and d must always be processed by the master station and the slave station irrespective of the fact whether the corresponding PROCESS DATA WORDS 1 4 and 5 are REF VALUES or COMMAND WORDS. This rule makes the implementation of a general-purpose PROCESS DATA receive function block easier (e.g. for a SIMADYN D fashioned TECH BOARD). As a consequence of this e.g. all bits in Bit mask d have to be set by the host system even if PROCESS DATA WORD 5 is a REF VALUE and is to be affected by the Broadcast Telegram.

- PROCESS DATA WORD 16

The 16th PROCESS DATA WORD is not supported by the Broadcast mechanism; this means: Word 16 is not Broadcast-capable and never taken over from a Broadcast Telegram. The master station can modify this word via a Non-Broadcast Telegram only.

The rules for processing the PROCESS DATA WORDS of a Broadcast Telegram are summarized in the following table:

PROCESS DATA WORD	BROADCAST ENABLE bits which must be taken into account
WORD 1	Bit mask a (Bit 1) and Bit mask b
WORD 2	Bit mask a (Bit 2)
WORD 3	Bit mask a (Bit 3)
WORD 4	Bit mask a (Bit 4) and Bit mask c
WORD 5	Bit mask a (Bit 5) and Bit mask d
WORDS 6 to 15	Bit mask a (Bits 6 to 15)
WORD 16	not taken over from a Broadcast Telegram (not Broadcast-capable)

Example 1 of a Broadcast Telegram:

The host system wants to give the same Main Reference Value 'REF1.' to all drives connected to the bus via a Broadcast Telegram. All other PROCESS DATA of the telegram should be ignored by the drives.

⇒ In this case the host system has to transmit the following information in the BROADCAST ENABLE ARRAY:

Word a = 0000 0000 0000 0101 (Broadcast Flag set and
PROCESS DATA WORD 2
enabled)

Words b to d = 0000 0000 0000 0000

Example 2 of a Broadcast Telegram:

The host system wants to start the ramp generators of all drives connected to the bus simultaneously via one Broadcast Telegram. All other PROCESS DATA must not be processed by the drives.

According to the PROFIBUS Profile (Lit [2]) the ramp generator can be enabled by setting bit 5 of COMMAND WORD 1 to a "1" value.

⇒ In this case the host system has to transmit the following information in the BROADCAST ENABLE ARRAY:

Word a = 0000 0000 0000 0011 (Broadcast Flag set and
PROCESS DATA WORD 1
(= COMMAND WORD 1)
enabled)

Word b = 0000 0000 0010 0000 (Command Bit 5 enabled)

Words c to d = 0000 0000 0000 0000

Remarks on the Broadcast Telegram:

- Broadcast Telegrams are only possible with telegram types containing a PARAMETER DATA area (e.g. PROFIBUS Adjustable-Speed Drives Profile , PPO types 1, 2 or 5)
- For a double-word REF VALUE (32 bit information) the Enable Bits of both: High Word and LOW Word have commonly to be set or reset by the master station.

Bereich
Antriebs-, Schalt- und Installationstechnik
Geschäftsgebiet Antriebstechnik
Postfach 3269, D-91050 Erlangen