onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

INTERBUS-S

Anwenderhandbuch

Hard- und Firmware der Anschaltbaugruppe für IBM®-kompatible PCs

Typ: IBS PC CB HW UM

Revision: C

Artikel-Nr.: 27 47 15 4

Dieses Handbuch ist gültig für folgende Anschaltbaugruppen mit der Firmware-Version 3.72:

IBS PC CB/I-T	Artikel-Nr. 27 80 84 9
IBS PC CB/COP/I-T	Artikel-Nr. 27 54 51 6
IBS PC CB/RTX486/I-T	Artikel-Nr. 27 61 47 0

Copyright by Phoenix Contact 07/1995

PHOENIX CONTACT

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Bitte beachten Sie folgende Hinweise:

Um Ihnen einen möglichst unproblematischen Gebrauch dieses Handbuchs und einen sicheren Geräteeinsatz in den Phasen der Installation, des Betriebs und der Wartung zu gewährleisten, sollen Sie folgende Hinweise aufmerksam lesen und beachten:

Erklärungen zu den verwendeten Symbolen

Das Symbol *Achtung* bezieht sich auf fehlerhafte Handlungen, die einen Schaden der Hard- oder Software oder, in indirektem Zusammenhang mit gefährlicher Prozeßperipherie (z.B. ungeschützt laufende Wellen von Motoren mit Aktorfunktion), leichten bis schweren Personenschaden zur Folge haben können. Das Symbol steht immer links vom zu kennzeichnenden Text.

Die *Hinweis-Hand* gibt Ihnen Tips und Ratschläge für den effizienten Geräteeinsatz und die Softwareoptimierung, um Ihnen zum Beispiel eine mögliche Mehrarbeit zu ersparen. Außerdem vermittelt Ihnen der so gekennzeichnete Text systembedingte Randbedingungen, die unbedingt beachtet werden müssen, um einen fehlerfreien Betrieb zu ermöglichen. Die Hand steht aber auch für begriffliche Klarstellungen.

Das Symbol *Text* verweist Sie auf weiterführende Informationsquellen (Handbücher, Datenblätter, Literatur, etc.) zu dem angesprochenen Thema, Produkt, o.ä. Ebenso gibt dieser Text hilfreiche Hinweise zur Orientierung, Lesereihenfolge etc . im Handbuch.

Ihre Meinung interessiert uns

Wir sind ständig bemüht, die Qualtität unserer Handbücher zu verbessern. Sollten Sie Anregungen und Verbesserungsvorschläge zu Inhalt und Gestaltung unseres Handbuchs haben, würden wir uns freuen, wenn Sie uns Ihre Vorschläge zusenden. Verwenden Sie dazu bitte den universellen Telefax-Vordruck am Ende dieses Handbuchs.

Erklärungen zu den rechtlichen Grundlagen

Dieses Handbuch, einschließlich aller darin enthaltenen Abbildungen, ist urheberrechtlich geschützt. Jede Drittverwendung dieses Handbuchs, die von den urheberrechtlichten Bestimmungen abweicht, ist verboten. Die Reproduktion, Übersetzung sowie die elektronische und photographische Archivierung und Veränderung bedarf der schriftlichen Genehmigung der Firma Phoenix Contact. Zuwiderhandlungen verpflichten zu Schadensersatz.

Phoenix Contact behält sich jegliche Änderungen, die dem technischen Fortschritt dienen, vor.

Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustereintragung sind Phoenix Contact vorbehalten. Fremdprodukte werden stets ohne Vermerk auf Patentrechte genannt. Die Existenz solcher Rechte ist daher nicht auszuschließen.

Der in diesem Handbuch beschriebene Produktgebrauch richtet sich ausschließlich an qualifizierte Anwendungsprogrammierer und Software-Ingenieu-

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

re, die mit den Sicherheitskonzepten der Automatisierungstechnik und den geltenden nationalen Normen vertraut sind. Für Fehlhandlungen und Schäden, die durch Mißachtung der Informationen dieses Handbuchs an Produkten von Phoenix Contact oder Produkten anderer Hersteller entstehen, übernimmt Phoenix Contact keine Haftung.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Inhaltsverzeichnis

1 Einleitung und Übersicht 1-3

1.1 Schnellstart unter DOS 1-3
1.2 Die Programmierung - Grundlegendes 1-4
1.3 Dokumentation 1-6
1.4 Modularer Aufbau der Anschaltbaugruppe 1-8

2 Technische Beschreibung des Grundboards 2-3

2.1 Kurzbeschreibung 2-3
2.2 Mechanischer Aufbau 2-4
2.2.1 Aufbau der Anschaltbaugruppe 2-4
2.2.2 Funktion der LEDs 2-7
2.3 Schnittstellen der Anschaltbaugruppen 2-8
2.3.1 Verbindung des Grundboards mit Ihrem PC 2-8
2.3.2 Verbindung des Grundboards mit den Aufsteckboards 2-10
2.3.2.1 Schnittstellen des IBS-Masterboards 2-10
2.4 Funktionseinheiten der Anschaltbaugruppen 2-13
2.4.1 AT-Bus-Interface 2-13
2.4.1.1 I/O-Basisadresse im Host 2-13
2.4.2 Multi-Port-Memory 2-14
2.4.2.1 Zugriffsverfahren auf das MPM 2-14
2.4.3 Interrupt-Funktionen 2-15
2.4.4 Spannungsüberwachung, Reset-System 2-15
2.4.5 Watchdog zur Überwachung des Hosts 2-16
2.4.6 Stromversorgung 2-16
2.4.6.1 Galvanisch getrennte IBS-Spannungsversorgung 2-16

3 Technische Beschreibung der Coprozessorboards 3-3

3.1 Kurzbeschreibung 3-3
3.2 Mechanischer Aufbau 3-5
3.3 Schnittstellen des Coprozessorboards 3-6
3.3.1 Grundboard-Schnittstelle 3-6
3.3.2 Serielle Schnittstelle 3-6
3.4 Funktionseinheiten des Coprozessorboards 3-8
3.4.1 Prozessor/Chipset 3-8
3.4.1.1 Komponenten des Chipsets 3-9
3.4.1.2 I/O-Adreßbereich des Coprozessorboards 3-9
3.4.1.3 Interruptbelegung des Coprozessorboards 3-10
3.4.2 Speicher des Coprozessorboards 3-10
3.4.2.1 EPROM 3-10
3.4.2.2 Statisches RAM 3-11
3.4.2.3 Dynamisches RAM 3-12
3.4.3 MPM-Interface 3-13
3.4.4 Mapping-Register 3-14
3.4.5 Serielle Schnittstelle des Coprozessorboards 3-14
3.4.6 Sicherheitseinrichtungen des Coprozessorboards 3-14
3.4.6.1 Spannungsüberwachung des Coprozessorboards 3-14

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3.4.6.2 Resetsystem des Coprozessorboards 3-14
3.4.6.3 Watchdog des Coprozessorboards 3-15
3.4.7 Echtzeituhr des Coprozessorboards 3-15
3.4.8 Stromversorgung des Coprozessorboards 3-16
3.4.8.1 Batterie-Pufferung 3-16

4 Einbau und erste Inbetriebnahme 4-3

4.1 Einstellen der Adressierung 4-3
4.1.1 Basisadresse im IO-Bereich des PCs (IO-Address) 4-3
4.1.2 Boardnummer (Board No.) 4-4
4.2 Einstellen der Boot-Konfiguration 4-6
4.2.1 Kontrolle des InterBus-S (IBS-Control) 4-6
4.2.2 Anlaufverhalten des InterBus-S (IBS-Autostart) 4-7
4.2.3 Automatischer Programmstart aus EPROM (EPROM-Start) . . 4-8
4.2.4 Einstellen des Boot-Laufwerks (Boot Disk) 4-8
4.2.5 Bootverhalten in Bezug auf RFSERVER (Wait for RFSERVER) 4-10
4.2.6 Boot-Verhalten in Bezug auf DPCON (Wait for DPCON) . . . 4-11
4.2.7 Datenübertragung zwischen COP und Entwicklungsumgebung 4-12
4.3 Einstellung der Steckbrücken (Jumper) 4-13
4.3.1 Auswahl der Spannungsversorgung (Power Supply) 4-13
4.3.2 Trennung vom Hardware-Reset des Hosts (PC HW RESET) . 4-14
4.3.3 Deaktivieren der Reset-Taster (Enable/Disable RESET-Button) 4-14
4.4 Anschluß des Batterie-Packs 4-15
4.5 Einbau der Anschaltbaugruppe in den PC 4-15
4.5.1 Serielle Schnittstelle des Coprozessorboards 4-16
4.6 Installation des Device-Drivers 4-16
4.6.1 Device-Driver unter MS-DOS 4-18
4.6.1.1 Installationshilfe für DOS 4-19
4.6.2 Device-Driver unter Microsoft Windows 4-20
4.6.2.1 Installationshilfe für Microsoft Windows 4-21
4.6.3 Device-Driver unter IBM OS/2 4-22
4.6.3.1 Installationshilfe für OS/2 4-22
4.7 Installation der Peripherie 4-23
4.8 Software-Tools für die Inbetriebnahme 4-23
4.9 Inbetriebnahme mit Prozeßdaten-Monitorprogramm 4-24
4.9.1 Das Pull-Down-Menü Functions 4-25
4.9.2 Absetzen von Kommandos mit PCCBMONI 4-28
4.9.3 Das Pull-Down-Menü Options 4-30

5 Schnittstellen zwischen Hardware und Software 5-3

5.1 Multi-Port-Memory 5-3
5.1.1 Das MPM im Adreßraum des Hosts 5-3
5.1.2 Aufteilung des MPMs 5-4
5.2 Allgemeine Struktur der Treibersoftware 5-7
5.2.1 Realisierung des DDI und des DD 5-9
5.2.2 Struktur der Treibersoftware auf dem Coprozessorboard . . . 5-9
5.2.3 Begriffserklärungen zur Treibersoftware 5-10
5.2.3.1 Verwaltung von Datenkanälen 5-10
5.2.3.2 Mailbox-Interface 5-12
5.2.3.3 Data-Interface 5-12
5.2.3.4 Diagnosefunktion 5-12

5.3 Nutzung des statischen RAMs 5-13
5.4 Datenübertragung zwischen Host und COP 5-13
5.4.1 Aufbau einer Nachricht zwischen Host und COP 5-14
5.5 Überwachung durch Watchdogs 5-14
5.5.1 Watchdog des IBS-Masterboards 5-14
5.5.2 Watchdog zur Überwachung des Hosts 5-15
5.5.3 Watchdog des Coprozessorboards 5-15
5.5.4 Das Signal SysFail 5-16
5.6 Download des Anwenderprogramms auf den COP 5-16

6 InterBus-S-spezifische Programmierung 6-3

6.1 Identifikation der angeschlossenen IBS-Teilnehmer 6-3
6.1.1 Physikalische Zählweise für Bussegmente und IBS-Teilnehmer 6-3
6.1.2 Beispielaufbau 6-4
6.1.3 Adressierungsarten des InterBus-S 6-6
6.2 Physikalische Adressierung von IBS-Teilnehmern 6-7
6.2.1 Adressen bei physikalischer Adressierung 6-8
6.2.1.1 Vergabe der Eingangsadressen durch die Anschaltbaugruppe . 6-8
6.2.1.2 Vergabe der Ausgangsadressen durch die Anschaltbaugruppe 6-10
6.2.2 Kommandofolge für den Anlauf bei physikalischer Adressierung 6-12
6.3 Logische Adressierung von IBS-Teilnehmern 6-13
6.3.1 Ermitteln der aktuell angeschlossenen Buskonfiguration . . . 6-14
6.3.2 Überprüfen der Buskonfiguration 6-14
6.3.3 Vergabe von logischen Bussegmentnummern 6-16
6.3.4 Vergabe der logischen Adressen durch den Programmierer . . 6-19
6.3.4.1 Vergabe der logischen Eingangsadressen 6-20
6.3.4.2 Vergabe der logischen Ausgangsadressen 6-24
6.3.4.3 Gültigkeit der Zuordnungslisten prüfen 6-27
6.3.5 Kommandofolge für den Anlauf bei logischer Adressierung . . 6-28
6.4 Gruppendefinition 6-29
6.4.1 Bilden von funktionalen Gruppen 6-29
6.4.2 Abschalten von Gruppen 6-32
6.4.3 Einschalten von Gruppen 6-33
6.4.4 Behandlung der Gruppen für den Fehlerfall definieren 6-33

7 Diagnose im Fehlerfall 7-3

7.1 Diagnose an der Hardware 7-3
7.1.1 Diagnoseanzeigen an der Anschaltbaugruppe 7-3
7.1.2 Diagnoseanzeigen an Busklemmen 7-4
7.1.3 Diagnoseanzeigen an IBS-Teilnehmern mit E/A-Funktion . . . 7-5
7.1.4 Diagnose an IBS-Teilnehmern von anderen Geräteherstellern . 7-5
7.2 Diagnose mit Softwaretools 7-6
7.2.1 Das Prozeßdaten-Monitorprogramm 7-6
7.2.2 Die Diagnose und Konfigurations-Software IBS SYS SWT . . 7-6
7.2.3 Der InterBus-Manager IBS CMD SWT 7-7
7.3 Diagnose durch das Anwenderprogramm 7-8
7.3.1 Diagnose von Anschaltbaugruppe und Busaufbau 7-8
7.3.1.1 Fehlerart 7-9
7.3.1.2 Bedeutung von Controller-Fehlernummern 7-10

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8 Kommandos für das IBS-Masterboard 8-3

8.1 Aufbau einer Kommandobeschreibung 8-5
8.2 Kommandos zur Konfiguration 8-6
8.3 Kommandos zur Adressierung 8-13
8.4 Kommandos für den Betrieb 8-16
8.5 Kommandos zur Fehlerbehandlung 8-19
8.6 Kommandos für die Anwendungsschnittstelle 8-26
8.7 Kommandos zur Systemkontrolle 8-27
8.8 Kommandos zur Prozeßdatenverknüpfung 8-28
8.9 Kommandos zur Event-Bearbeitung 8-38

9 Meldungen des IBS-Masterboards 9-3

9.1 Aufbau einer Meldungsbeschreibung 9-5
9.2 Meldungen zur Konfiguration 9-6
9.3 Meldungen zur Adressierung 9-11
9.4 Meldungen für den Betrieb 9-14
9.5 Meldungen zur Fehlerbehandlung 9-15
9.6 Meldungen der Anwenderschnittstelle 9-31
9.7 Meldungen zur Systemkontrolle 9-32
9.8 Meldungen zur Prozeßdatenverknüpfung 9-33
9.9 Meldungen zur Event-Bearbeitung 9-34

A Technischer Anhang A-3

A.1 Technische Daten der Anschaltbaugruppen A-3

B Verzeichnisanhang B-3

B.1 Abbildungsverzeichnis B-3
B.2 Tabellenverzeichnis B-6
B.3 Stichwortverzeichnnis B-9

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel **1**

Einleitung und Übersicht

Dieses Kapitel gibt Ihnen:

- eine kurze Einführung zu Parametrierung und Programmierung der Anschaltbaugruppen
- eine Übersicht über die für InterBus-S erhältliche Dokumentation

1 Einleitung und Übersicht 1-3

1.1 Schnellstart unter DOS 1-3
1.2 Die Programmierung - Grundlegendes 1-4
1.3 Dokumentation 1-6
1.4 Modularer Aufbau der Anschaltbaugruppe 1-8

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

1 Einleitung und Übersicht

Die Anschaltbaugruppen der Baureihe IBS PC CB/.../I-T dienen zur Anbindung des InterBus-S an IBM-kompatible PCs. Die zugehörige Dokumentation soll Ihnen Informationen zu allen Funktionalitäten und Anwendungsfällen geben. Das bedingt einen entsprechenden Umfang, der es nicht immer einfach macht, die gewünschten Informationen schnell zu finden.

Wir möchten Ihnen daher an dieser Stelle eine kurze Übersicht über die Handhabung der Anschaltbaugruppe geben. Die einzelnen Schritte sind in den folgenden Kapiteln detailliert beschrieben. Falls nach dem Durcharbeiten dieses Handbuchs noch Fragen offen bleiben sollten, steht Ihnen unter der Telefonnummer 05235 / 34 18 88 unsere technische Hotline zur Verfügung.

1.1 Schnellstart unter DOS

Die Anschaltbaugruppe benötigt 8 Bytes im I/O-Bereich Ihres PCs, einen freien Interrupt sowie 4 KByte freien Adreßraum. Die Standardeinstellungen des Auslieferungszustands sind so gewählt, daß sie oft nicht verändert werden müssen.

1. Stellen Sie fest, ob die Standard-Einstellung (default) zu Ihrer PC-Konfiguration paßt:

I/O-Adresse	120_{hex}
Interrupt	15
Speicheradresse	$D000_{hex}$

Achten Sie unbedingt darauf, daß die für die Anschaltbaugruppe gewählte Einstellungen nicht schon durch andere Komponenten Ihres PCs belegt sind. Der Speicherbereich zwischen 640 KByte und 1 MByte z.B. wird häufig von Treibern o.ä. genutzt. Wählen Sie, falls notwendig, andere Werte (siehe Kapitel 4). Doppelbelegungen sind die häufigste Fehlerursache bei der Inbetriebnahme!

2. Schalten Sie den PC Spannungsfrei (Ausschalten und Netzkabel abziehen). und Bauen Sie die Anschaltbaugruppe ein. Die Anschaltbaugruppe darf nicht auf anderen Bauteilen im PC aufliegen oder Kontakt zu ihnen haben (Kurzschlußgefahr)!
3. Schalten Sie den PC wieder ein. Die Inbetriebnahme der Anschaltbaugruppe erfolgt unter DOS am einfachsten mit dem auf der Tool-Diskette mitgelieferten Monitorprogramm *PCCBMONI.EXE* (siehe Kapitel 4).

   Vor dem Start des Monitorprogramms muß der Treiber *IBSPCCB.EXE* für die Anschaltbaugruppe, gestartet werden. Falls Sie für Ihre Anschaltbaugruppe die Standard-Einstellungen übernommen haben, brauchen Sie beim Aufruf des Treibers keine Parameter übergeben. Das ebenfalls auf der Tool-Diskette enthaltene Batch-File *MONI.BAT* bietet Ihnen dann eine einfache Möglichkeit, Treiber und Monitorprogramm mit einem Aufruf zu starten. Wenn Sie eine Buskonfiguration angeschlossen haben, können Sie diese mit dem Monitorprogramm überprüfen ohne eine einzige Zeile zu programmieren.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

1.2 Die Programmierung - Grundlegendes

Die Treibersoftware greift über ein 4 KByte großes Fenster, das zwischen 640 KB und 1 MB in den Speicherbereich des PCs eingeblendet wird, auf das Multi-Port-Memory (MPM) der Anschaltbaugruppe zu. Hierzu stehen folgende Funktionen zur Verfügung:

- Funktionen zum Öffnen und Schließen von Datenkanälen
- Funktionen zum Schreiben von Kommandos und Lesen von Meldungen (Mailbox-Interface, arbeitet mit Handshake-Signalen und Interruptsteuerung)
- Funktionen zum Lesen und Schreiben von E/A-Daten (Data-Interface, arbeitet unquittiert)
- Diagnosefunktion zur Kontrolle des Betriebszustands der Anschaltbaugruppe

Beim Öffnen eines Datenkanals erhalten Sie einen *Node-Handle* zurück, der ähnlich wie der *Handle* bei einem Dateizugriff den Datenkanal spezifiziert und beim Lesen und Schreiben von Daten angegeben werden muß (siehe Handbuch zur Treibersoftware *IBS PC CB SWD UM*, Artikel-Nr.: 27 53 94 4).

Steuerung des InterBus-S

Ihr Anwenderprogramm steuert die Anschaltbaugruppe über Kommandos (z.B. Starten des Bussystems, Buskonfiguration einlesen, siehe Kapitel 8).

Die Anschaltbaugruppe betreibt den Bus nach ihrer Initialisierung und dem Start der Datenübertragung selbsttätig und gibt entsprechende Meldungen (siehe Kapitel 9) zurück.

Das Kapitel 6 zeigt die InterBus-spezifische Programmierung wie
- die physikalische Adressierung der IBS-Teilnehmer,
- die logische Adressierung der IBS-Teilnehmer und
- das Zusammenfassen von IBS-Teilnehmern zu Gruppen

an einem Beispielaufbau.

Im Falle eines ernsten Fehlers (z.B. unterbrochenes Buskabel) gehen alle angeschlossenen IBS-Teilnehmer automatisch in den RESET und setzen ihre Ausgänge auf Null. Die Anschaltbaugruppe untersucht daraufhin den Fehler und gibt über die Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$, siehe Kapitel 9) eine detaillierte Beschreibung von Fehlerursache und Fehlerort.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Beispielprogramme

Die Programmierung der Anschaltbaugruppe ist unter DOS, Microsoft Windows® und IBM OS/2® weitgehend identisch. So kann z.B. für die Programmierung unter OS/2 ohne weiteres auf ein DOS-Beispielprogramm aufgebaut werden. Auf der Treiberdiskette stehen Ihnen Beispielprogramme zur Verfügung, die bereits das notwendige Bus-Handling enthalten. Sie können sich voll und ganz auf die Programmierung ihrer E/A-Verknüpfungen konzentrieren.

Schulungen

Sie haben sicherlich auch schon die Erfahrung gemacht, daß Sie eine längere Einarbeitungszeit benötigen, um sich mit allen Möglichkeiten umfangreicher Softwarepakete wie z.B. *Microsoft Winword®* oder *Excel®* vertraut zu machen.

Ähnliches gilt für InterBus-S. Sie können sich selbstverständlich leicht an Hand der Dokumentation und der mitgelieferten Beispielprogramme in die Programmierung der Anschaltbaugruppen einarbeiten. Um deren volle Leistungsfähigkeit auch ganz ausnutzen zu können, empfehlen wir Ihnen zusätzlich eine unserer Programmierschulungen, die Ihnen ein umfangreiches Praxiswissen vermittelt. Inhalte und Termine entnehmen Sie bitte unserer Seminarbroschüre, die Ihnen Ihre nächstgelegene Phoenix-Vertretung gerne zusendet. Wir kommen auch, auf Wunsch auch mit einer speziell auf Ihre Bedürfnisse zugeschnittenen Schulung, zu Ihnen ins Haus. Sprechen Sie uns an!

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

1.3 Dokumentation

Zu den Anschaltbaugruppen

- IBS PC CB/I-T,
- IBS PC CB/COP/I-T und
- IBS PC CB/RTX486/I-T

sind die folgenden Dokumentationen verfügbar.

Beschreibung der Hardware

Das vorliegende Handbuch (IBS PC CB HW UM) beschreibt die Hardware der drei Anschaltbaugruppen und die InterBus-spezifische Programmierung. Außerdem ist eine Bibliothek mit Kommandos und Meldungen für das IBS-Masterboard (Firmware-Version 3.72) angefügt.

Beschreibung der Treibersoftware

Das Handbuch *IBS PC CB SWD UM* (Artikel-Nr. 27 53 94 4) beschreibt Treibersoftware für die Betriebssysteme *DOS*, *Microsoft Windows®* und *IBM OS/2®* in Verbindung mit verschiedenen Compilern. Es gehört auch zum Lieferumfang des Treibersoftwarepakets *IBS PC CB SWD* (Artikel-Nr. 27 64 70 7)

Beschreibung der Entwicklungsumgebung TDOS-PRO

(nur für IBS PC CB/COP/I-T)

Die Entwicklungsumgebung *TDOS-PRO* (IBS PC COP SWT, Artikel-Nr. 27 52 12 3*)* ermöglicht den Download eines Programms auf den *COP386*. Die zugehörige Dokumentation wird mitgeliefert.

Beschreibung des Betriebssystems RTXDOS

(nur für IBS PC CB/RTX486/I-T)

Das Handbuch zu RTX-DOS beschreibt die speziellen Erweiterungen und Möglichkeiten dieses DOS-kompatiblen Betriebssystems für den *COP486*.

Kommunikation über InterBus-S (PCP)

Für die Übertragung von Parametrierungsdaten zu intelligenten IBS-Teilnehmern oder die Kommunikation zu einem IBS-Teilnehmer mit V24-Schnittstelle steht das **P**eripherals **C**ommunication **P**rotocol (kurz PCP) zur Verfügung. PCP ist eine auf das Basisprotokoll des InterBus-S aufsetzende Softwareschnittstelle und ermöglicht die Übertragung von zeitunkritischen größeren Datensätzen nahezu unabhängig von den Prozeßdaten.

Das Handbuch *IBS PCP UM* (Artikel-Nr. 27 53 92 8) beschreibt die Grundlagen und die Anwendung des Peripherals-Communication-Protokolls.

Projektierung Ihrer InterBus-S-Anlage

Das Projektierungshandbuch *IBS SYS PRO UM* (Artikel-Nr. 27 53 82 1) gibt Ihnen Informationen zur Auswahl von Komponenten für Ihre IBS-Anlage. Es beschreibt folgende Betriebsmerkmale:

- Elektrische Eigenschaften (Spannungs- und Strombereiche)
- Mechanische Eigenschaften (Schutzart, Anschlußart, Montagemöglichkeiten usw.)
- Programmtechnische Eigenschaften (z.B. benötigter Adreßraum im Host)

Das Handbuch stellt, die Bauformen der von Phoenix Contact lieferbaren Produktfamilien sowie deren Merkmale und Bestelldaten vor.

Es ist außerdem eine Sammlung aller aktuellen Datenblätter beigefügt.

Installation Ihres InterBus-S-Systems

Detallierte Hinweise zur Installation der Peripherie (IBS-Teilnehmer, Module, Kabel usw.) entnehmen Sie bitte dem Installationshandbuch *IBS SYS INST UM* (Artikel-Nr. 27 54 28 6). Das Handbuch informiert Sie in verschiedenen Kapiteln zu folgenden Themen:

- Systemübersicht
- Montage der E/A-Komponenten
- Verkabelungsempfehlungen
- Inbetriebnahme und Funktionstest der Peripherie
- Fehlerbeseitigung
- Austausch von IBS-Komponenten
- Kabelpläne

Tabelle 1-1: Bestelldaten für Dokumentation

Beschreibung	Typ	Artikel-Nr.
System-Ordner zu den Anschaltbaugruppen, enthält: - Anwenderhandbuch zu den Anschaltbaugruppen - Tool-Diskette - Handbuch zu Grundlagen und Anwendung des Peripherals-Communication-Protokolls (PCP)	IBS PC CB UM	27 54 74 9
Anwenderhandbuch zu den Anschaltbaugruppen, einzeln (gehört auch zum System-Ordner)	IBS PC CB HW UM	27 47 15 4
Handbuch zur Treibersoftware (gehört auch zum Lieferumfang des Treibersoftwarepakets *IBS PC CB SWD*)	IBS PC CB SWD UM	27 53 94 4
Handbuch zu Grundlagen und Anwendung des Peripherals-Communication-Protokolls (PCP), einzeln (gehört auch zum System-Ordner)	IBS PCP UM	27 53 92 8
Projektierungshandbuch für IBS-Anlagen	IBS SYS PRO UM	27 53 82 1
Installationshandbuch für IBS-Komponennten	IBS SYS INST UM	27 54 28 6
Datenblätter zu neuen digitalen und analogen E/A-Modulen	auf Anfrage	

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

1.4 Modularer Aufbau der Anschaltbaugruppe

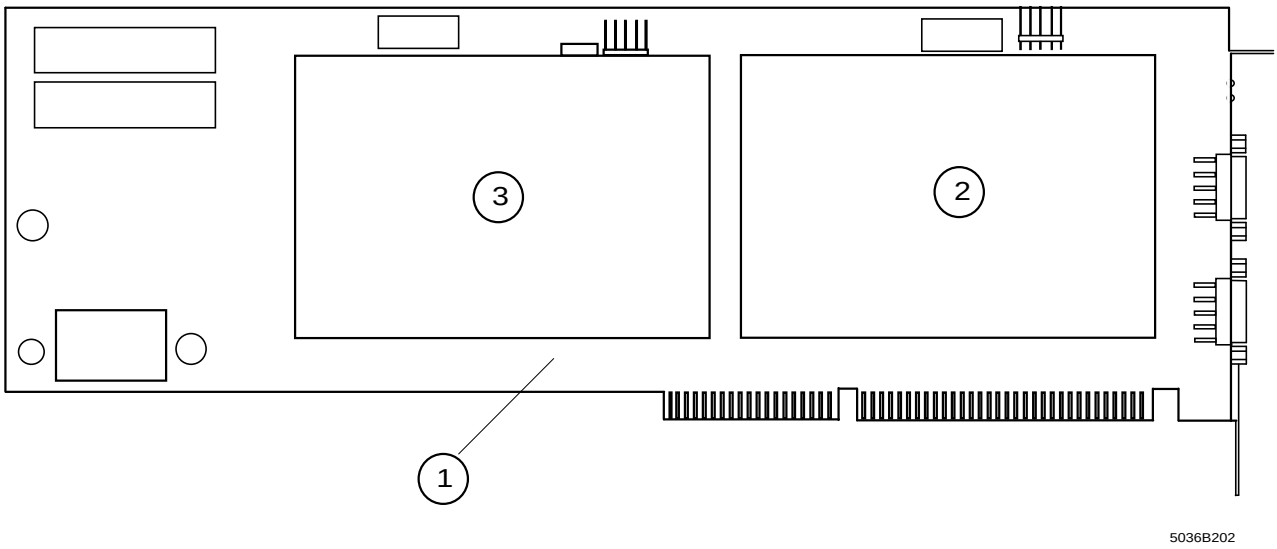

Bild 1-1: Modularer Aufbau der Anschaltbaugruppe

Die Anschaltbaugruppen IBS PC CB/I-T, IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T sind modular aufgebaut und bestehen aus folgenden Komponenten:

1 Grundboard
2 IBS-Masterboard (MA)
3 Coprozessorboard (COP386 oder COP486)

Ein Coprozessorboard befindet sich nur auf den Anschaltbaugruppen IBS PC CB/ COP/I-T und IBS PC CB/RTX486/I-T. Es ist für die Anschaltbaugruppe IBS PC CB I-T nicht nachrüstbar, da auf dessen Grundboard die entprechenden Schnittstellen und die Spannungsversorgung fehlen.

Die folgenden Kapitel beschreiben die Anschaltbaugruppen genauer.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel **2**

Technische Beschreibung des Grundboards

Dieses Kapitel informiert Sie über

- den Aufbau und die Komponenten des Grundboards.

2 Technische Beschreibung des Grundboards 2-3

2.1 Kurzbeschreibung 2-3
2.2 Mechanischer Aufbau 2-4
2.2.1 Aufbau der Anschaltbaugruppe 2-4
2.2.2 Funktion der LEDs 2-7
2.3 Schnittstellen der Anschaltbaugruppen 2-8
2.3.1 Verbindung des Grundboards mit Ihrem PC 2-8
2.3.2 Verbindung des Grundboards mit den Aufsteckboards 2-10
2.3.2.1 Schnittstellen des IBS-Masterboards 2-10
2.4 Funktionseinheiten der Anschaltbaugruppen 2-13
2.4.1 AT-Bus-Interface 2-13
2.4.1.1 I/O-Basisadresse im Host 2-13
2.4.2 Multi-Port-Memory 2-14
2.4.2.1 Zugriffsverfahren auf das MPM 2-14
2.4.3 Interrupt-Funktionen 2-15
2.4.4 Spannungsüberwachung, Reset-System 2-15
2.4.5 Watchdog zur Überwachung des Hosts 2-16
2.4.6 Stromversorgung 2-16
2.4.6.1 Galvanisch getrennte IBS-Spannungsversorgung 2-16

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

2 Technische Beschreibung des Grundboards

2.1 Kurzbeschreibung

Die Anschaltbaugruppen der Baureihe IBS PC CB/.../I-T dienen zur Ankopplung des InterBus-S an einen 100% IBM-kompatiblen Standard-PC (AT, 80386, 80486 etc.). Dieser PC wird im folgenden Host genannt. Die Anschaltbaugruppen sind als Einsteckkarte für einen langen AT-Bus-Steckplatz ausgelegt. Als Frontblende dient ein üblicher PC-Kartenhalter.

Das Grundboard der Anschaltbaugruppen stellt zwei Steckplätze für Aufsteckboards zur Verfügung. Das erste Aufsteckboard ist der IBS-Master (kurz MA) für die Ankopplung an den InterBus-S. Als zweites Aufsteckboard kann ein Coprozessorboard (kurz COP) als schneller I/O-Rechner für den InterBus-S genutzt werden. Für jedes Aufsteckboard sind auf der Frontblende zwei LEDs zur Anzeige des Betriebszustandes vorhanden.

Zentrale Funktionseinheit des Grundboards ist ein Multi-Port-Memory (MPM) mit integrierter Zugriffsverwaltung. Auf diesen Speicher kann vom Host wie auch von den beiden Aufsteckboards aus zugegriffen werden, d.h. das MPM hat drei Teilnehmer (Nodes). Das MPM besteht aus 64 KByte SRAM mit 16 Bit Datenwortbreite. Es dient zum Austausch von Kommandos, Meldungen und Daten zwischen den beiden Aufsteckboards und dem Host. Weiterhin sind die Mechanismen für die MPM-Zugriffssteuerung auf dem Grundboard enthalten.

In einem Host können bis zu vier Anschaltbaugruppen betrieben werden. Durch die Einstellung unterschiedlicher Boardnummern (von 1 bis 4) bei gleicher I/O-Basisadresse unterscheiden sich die Anschaltbaugruppen um einen von der Boardnummer abhängigen I/O-Adreßoffset von 0_{hex}, 8_{hex}, 10_{hex} und 18_{hex}.

Eine Anschaltbaugruppe belegt 8 Byte im I/O-Adreßbereich, sowie ein Memory-Fenster von 4 KByte im Speicherbereich des Hosts. Die Basis-Adresse des I/O-Adreßbereichs ist über einen DIP-Schalter auf eine von 16 möglichen Adressen einstellbar. Die Basisadresse des 4 KByte großen Memory-Fensters zum MPM ist vom Anwender durch die Treiber-Software über eine I/O-Adresse einstellbar.

Eine Spannungsüberwachungsschaltung gewährleistet den sicheren Betrieb der Anschaltbaugruppe. Ein umfangreiche Reset-System wird ergänzt durch die Möglichkeit, jedes der Aufsteckboards gezielt durch einen Software-Befehl zurückzusetzen. Als weitere Sicherheitsfunktion sind Watchdogs integriert, die das IBS-Masterboard, das Coprozessorboard und über den AT-Bus den Host überwachen.

Für den Interrupt-gesteuerten Betrieb der Anschaltbaugruppe werden durch die MPM-Steuerlogik für jeden der Teilnehmer spezifische Interruptsignale aktiviert. Der Anwender hat die Möglichkeit, das Interruptsystem an die Konfiguration seines Hosts anzupassen.

Das Grundboard der IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T enthält zusätzlich einen Batterie-Pack (6V) zur Pufferung des SRAMs und der Echtzeituhr auf dem Coprozessorboard.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

2.2 Mechanischer Aufbau

Das Grundboard der Anschaltbaugruppen der Baureihe IBS PC CB/.../I-T ist als PC-Einsteckkarte mit den Standardabmessungen von 338,5 mm * 114,3 mm aufgebaut. Zum Anschluß an den AT-Bus (ISA) Ihres PCs befinden sich am unteren Rand des Grundboards die für PC-Einsteckkarten typischen Direktsteckverbinder.

2.2.1 Aufbau der Anschaltbaugruppe

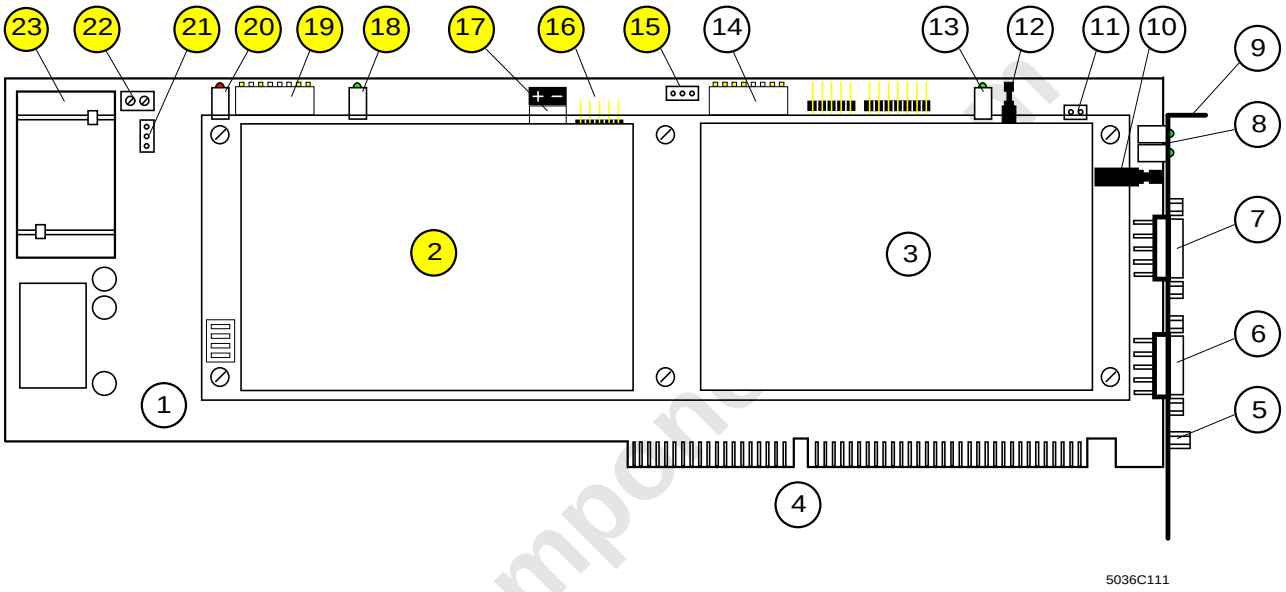

Bild 2-1: Aufbau der Anschaltbaugruppen der Baureihe IBS PC CB/.../I-T

Die Anschaltbaugruppen bestehen aus folgenden Komponenten (die grau hinterlegten Positionen entfallen für das Controllerboard IBS PC CB/I-T):

1 Grundboard
Das Grundboard trägt bis zu zwei Aufsteckboards und stellt die Schnittstelle zum AT-Bus des Hosts zur Verfügung.

2 Coprozessorboard
Zur Datenvorverarbeitung und Steuerung (nicht IBS PC CB/I-T).

3 IBS-Masterboard
Das IBS-Masterboard (MA-Board) betreibt den Datenverkehr auf dem InterBus-S selbständig.

4 AT-Bus-Direktsteckverbinder
Der Steckverbinder stellt die Verbindung der Anschaltbaugruppe mit dem AT-Bus (ISA) des Hosts her.

5 Erdungsbolzen
Erden Sie die Anschaltbaugruppe über den Erdungsbolzen.

6 InterBus-S
Fernbusschnittstelle in Zweileitertechnik

7 Diagnoseschnittstelle
Serielle Schnittstelle (RS-232-Pegel) für das IBS-Masterboard, zum Anschluß eines PCs mit den Softwaretools IBS CMD SWT oder IBS SYS

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

SWT.

8 Status-LEDs, extern
Die grünen LEDs zeigen den Status der Anschaltbaugruppe an, siehe Kapitel 2.2.2.

9 Frontblende
PC-Kartenhalter

10 Externer Reset-Taster
Verdeckt angeordneter Reset-Taster zur Bedienung bei geschlossenem PC-Gehäuse.

11 Steckbrücke für die Resettaster
Zum Deaktivieren des internen und des externen Resettasters

12 Interner Reset-Taster
Zur komfortablen Bedienung bei geöffnetem PC-Gehäuse.

13 Interne Status-LEDs des IBS-Masterboards (grün)
Zur komfortablen Anzeige des Betriebszustandes bei geöffnetem PC-Gehäuse.

14 DIP-Schalter
Zum Einstellen der Basisadresse im I/O-Bereich des PCs und der Boardnummer

15 Steckbrücke, Reset (nicht IBS PC CB/I-T)
Zur Trennung der Anschaltbaugruppe vom Hardware-Reset des PCs. Über diese Steckbrücke können Sie die Anschaltbaugruppe so konfigurieren, daß sie bei einem Hardware-Reset des PCs weiterläuft.

16 Serielle Schnittstelle (nicht IBS PC CB/I-T)
RS-232-Schnittstelle des Coprozessorboards

17 Anschluß für das Batterie-Pack zur Pufferung der RAM-Speicher auf dem Coprozessorboard (nicht IBS PC CB/I-T)

18 Interne Status LEDs des Coprozessorboards (grün, nicht IBS PC CB/I-T)
Zur komfortablen Anzeige des Betriebszustandes bei geöffnetem PC-Gehäuse.

19 DIP-Schalter (nicht IBS PC CB/I-T)
zur Einstellung der Bootkonfiguration des Coprozessorboards (nicht IBS PC CB/I-T)

20 LEDs für die Spannungsversorgung des Coprozessorboards (nicht IBS PC CB/I-T)
Die grüne LED zeigt an, daß die externe Versorgungsspannung für die Anschaltbaugruppe anliegt.
Wenn die rote LED dauerhaft leuchtet, ist die Spannung des Batterie-Packs unter den zulässigen Mindestwert gefallen. Tauschen Sie in diesem Fall das Batteriepack aus.

21 Steckbrücke Versorgung (nicht IBS PC CB/I-T)
Wählen Sie mit der Steckbrücke aus, ob die Anschaltbaugruppe ihre Versorgungsspannung vom AT-Bus oder über die Klemmen zur externen Versorgung beziehen soll.

22 Klemme Versorgung (nicht IBS PC CB/I-T)
Klemme zur Einspeisung der externen Versorgungsspannung. Über diese Klemmen können Sie die Anschaltbaugruppe von einem externen Netzteil aus mit Spannung (5V DC 1,5A) versorgen.

23 Batterie-Pack (nicht IBS PC CB/I-T)
zur Pufferung des CMOS-SRAM-Speichers und der Echtzeituhr des COP.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Aufbau der Frontblende

Auf der Frontblende (PC-Kartenhalter) befinden sich zwei 9-polige Sub-D-Steckverbinder für die Peripherie-Schnittstellen des IBS-Masterboards. Als Kontrollemente sind vier Funktions-LEDs und ein Reset-Taster hinter dem Kartenhalter plaziert. Unterhalb der Sub-D-Steckverbinder ist ein Rundbolzen zur Erdung der Anschaltbaugruppe. Bild 2-3 zeigt die Anordnung der Elemente auf dem PC-Kartenhalter.

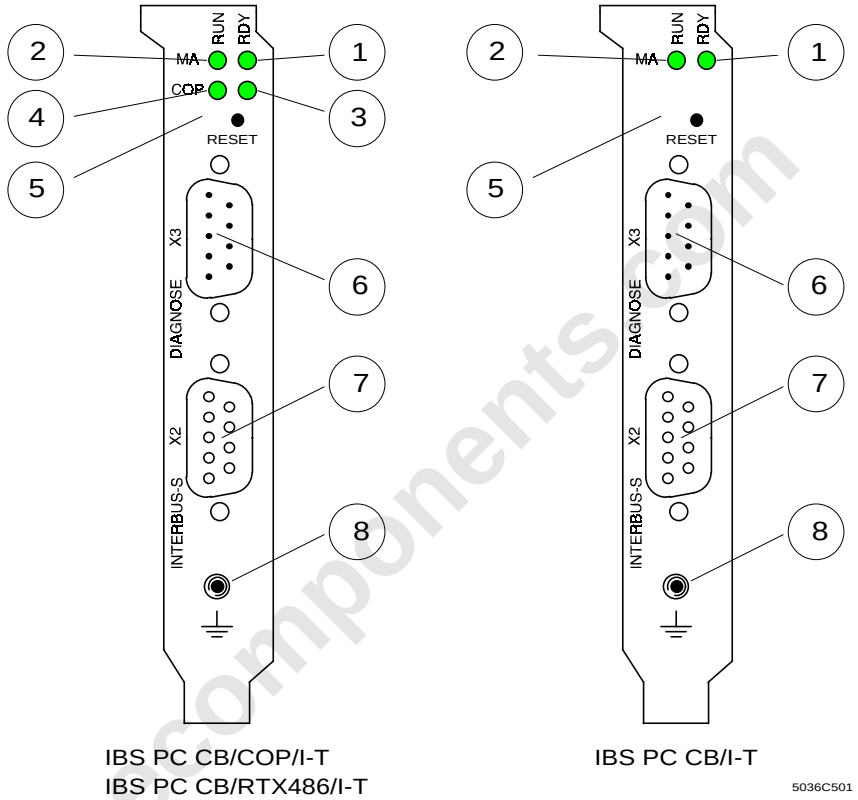

Bild 2-2: Elemente auf dem PC-Kartenhalter

1 Externe LED *MA READY* (grün)
2 Externe LED *MA RUN* (grün)
3 Externe LED *COP READY* (grün)
4 Externe LED *COP RUN* (grün)
5 Externer Reset-Taster
6 IBS-Diagnoseschnittstelle (9-poliger Sub-D-Stecker, RS232-Pegel)
7 IBS-Fernbusschnittstelle (9-polige Sub-D-Buchse)
8 Erdungsbolzen

Erden Sie die Anschaltbaugruppe über den Erdungsbolzen (8).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

2.2.2 Funktion der LEDs

Zur Anzeige des aktuellen Betriebszustandes stehen auf der Frontblende (PC-Kartenhalter) je Aufsteckboard 2 LEDs zur Verfügung (Bild 2-2), welche die Zustände *Ready* und *Run* signalisieren:

IBS-Masterboard

MA READY:	Das IBS-Masterboard hat nach dem Einschalten einen Bootcheck für alle Funktionseinheiten inklusive MPM fehlerfrei ausgeführt und ist betriebsbereit.
MA RUN:	Das IBS-Masterboard hat den InterBus-S gestartet. Es werden ID- oder Datenzyklen übertragen.

Coprozessor-Board
(nicht IBS PC CB/I-T)

COP RUN:	Das Betriebssystem des Coprozessorboards hat gebootet; es kann ein Anwendungsprogramm gestartet werden.
COP READY:	Auf dem Coprozessorboard läuft ein Programm.

Nach dem Booten des Coprozessorboards werden verschiedene Dienstprogramme automatisch gestartet, die teilweise als TSR-Programm im Speicher des COPs verbleiben. Daher leuchtet nach dem Hochfahren des Gesamtsystems die LED *COP READY*.

Die LEDs stehen an der dem AT-Bus gegenüberliegenden Kartenseite noch einmal zur Verfügung und können so bei für die Inbetriebnahme geöffnetem PC-Gehäuse komfortabel beobachtet werden.

Zur Einstellung der I/O-Basisadresse, der Boardnummer und zur Power-Up-Konfiguration sind an der Oberkante der Anschaltbaugruppe zwei 8-fach DIP-Schalter plaziert, die sich auch im eingebauten Zustand einfach bedienen lassen.

Die Einstellungen der DIP-Schalter werden nur beim Booten der Anschaltbaugruppe eingelesen. Nach einer Änderung der Einstellung muß ein Reset der Anschaltbaugruppe erfolgen, um die Änderung wirksam werden zu lassen.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

2.3 Schnittstellen der Anschaltbaugruppen

Die folgenden Kapitel beschreiben die verschiedenen Schnittstellen der Anschaltbaugruppe.

2.3.1 Verbindung des Grundboards mit Ihrem PC

Über die AT-Bus-Direktsteckverbinder wird die Anschaltbaugruppe mit dem AT-Bus des Hosts verbunden. Die Signalbelegung ist zum ISA-Standard kompatibel, wobei die Anschaltbaugruppe mit einer Datenbusbreite von 8 Bit auskommt.

Die Tabellen 2-1 und 2-2 zeigen die Anschlußbelegungen der Direktsteckverbinder zum AT-Bus. Bei den Signalen der Schnittstelle handelt es sich ausschließlich um standardisierte AT-Bus-Signale.

Tabelle 2-1: Belegung des kurzen AT-Bus-Direktsteckverbinders

Pin	Signal	Funktion
C1		
C2	LA23	I, T, PD100
C3	LA22	I, T, PD100
C4	LA21	I, T, PD100
C5	LA20	I, T, PD100
C6		
C7		
C8		
C9		
C10		
C11		
C12		
C13		
C14		
C15		
C16		
C17		
C18		

Pin	Signal	Funktion
D1		
D2		
D3	IRQ10	O, C, TS
D4	IRQ11	O, C, TS
D5	IRQ12	O, C, TS
D6	IRQ15	O, C, TS
D7		
D8		
D9		
D10		
D11		
D12		
D13		
D14		
D15		
D16	+5 V	V
D17		
D18	GND	V

Legende:

I = Eingang	V = Versorgungs-spannung	TS = Tristate
O = Ausgang	T = TTL-Pegel	PU ... = Pullup ... [kΩ]
B = Bidirektional	C = CMOS-Pegel	PD ... = Pulldown ... [kΩ]
M = nur zu Meßzwecken	OC = Open-Collector	UN ... = Nennspannung [V]

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 2-2: Belegung des langen AT-Bus-Direktsteckverbinders

Pin	Signal	Funktion
A1		
A2	SD7	B, T, TS
A3	SD6	B, T, TS
A4	SD5	B, T, TS
A5	SD4	B, T, TS
A6	SD3	B, T, TS
A7	SD2	B, T, TS
A8	SD1	B, T, TS
A9	SD0	B, T, TS
A10	IOCHRDY	O, C, PU10
A11	AEN	I, T
A12	SA19	I, T
A13	SA18	I, T
A14	SA17	I, T
A15	SA16	I, T
A16	SA15	I, T
A17	SA14	I, T
A18	SA13	I, T
A19	SA12	I, T
A20	SA11	I, T
A21	SA10	I, T
A22	SA9	I, T
A23	SA8	I, T
A24	SA7	I, T
A25	SA6	I, T
A26	SA5	I, T
A27	SA4	I, T
A28	SA3	I, T
A29	SA2	I, T
A30	SA1	I, T
A31	SA0	I, T

Pin	Signal	Funktion
B1	GND	V
B2	RESET	I, T
B3	+5 V	V
B4	IRQ2/9	O, C, T, S
B5		
B6		
B7		
B8		
B9	+12 V	V
B10	GND	V
B11	SMEMWL	I, T
B12	SMEMRL	I, T
B13	IOWL	I, T
B14	IORL	I, T
B15		
B16		
B17		
B18		
B19	REFRESHL	I, T
B20		
B21	IRQ7	O, C, TS
B22		
B23	IRQ5	O, C, TS
B24		
B25	IRQ3	O, C, TS
B26		
B27		
B28	BALE	I, T
B29	+5 V	V
B30	OSC	I, T
B31	GND	V

Legende:

I = Eingang	V = Versorgungs-spannung	TS = Tristate
O = Ausgang	T = TTL-Pegel	PU ... = Pullup ... [kΩ]
B = Bidirektional	C = CMOS-Pegel	PD ... = Pulldown ... [kΩ]
M = nur zu Meßzwecken	OC = Open-Collector	UN ... = Nennspannung [V]

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

2.3.2 Verbindung des Grundboards mit den Aufsteckboards

Jeweils zwei 58-polige Buchsenleisten stellen die Verbindung des Grundboards mit den Aufsteckboards her. Beide Schnittstellen sind in gleicher Weise mit den Adreß-, Daten- und Steuerleitungen des Multi-Port-Memorys (MPM) und mit steckplatzspezifischen MPM-Signalen belegt. Außerdem liegen auf den Steckverbindern beider Schnittstellen die Signale eventueller Peripherie-Schnittstellen. Dabei ist zwischen den auf System-Potential liegenden Peripherie-Signalen und den galvanisch getrennten IBS-Signalen zu unterscheiden.

2.3.2.1 Schnittstellen des IBS-Masterboards

Das IBS-Masterboard hat zwei Schnittstellen, deren Signale über die 58-poligen Buchsenleisten auf die im folgenden beschriebenen Steckverbinder des Grundboards geführt werden.

Diagnoseschnittstelle (seriell)

Über die Diagnoseschnittstelle (RS232-Pegel) der Anschaltbaugruppe kann ein IBM-kompatibler PC mit der Software *IBS SYS SWT* oder *IBS CMD SWT* als Diagnosegerät angeschlossen werden. Die Diagnoseschnittstelle ist auf der Frontblende der Anschaltbaugruppe als 9-polige Sub-D-Stiftleiste ausgeführt.

Die Verbindung der Diagnoseschnittstelle mit dem IBM-kompatiblen PC erfolgt über das Diagnosekabel *IBS PRG CAB* (Artikel-Nr. 28 06 86 2), das der folgenden Darstellung entspricht.

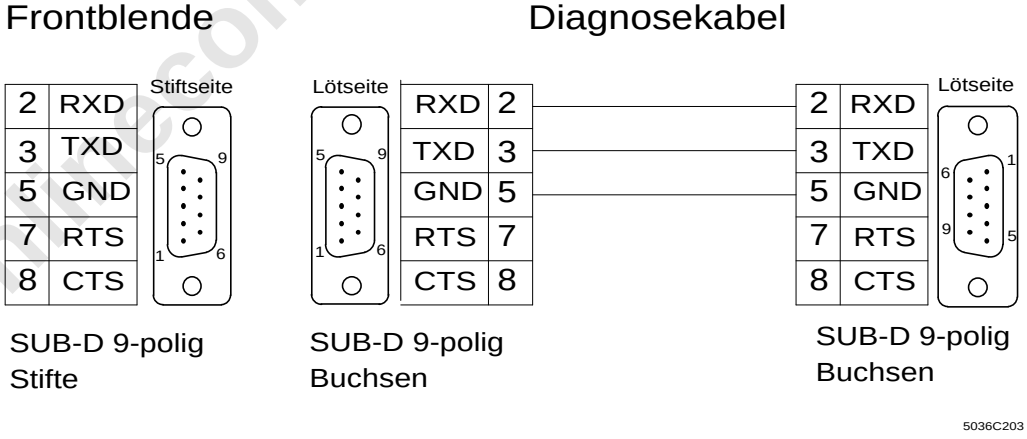

Bild 2-3: Diagnoseschnittstelle und Diagnosekabel zur Verbindung mit einem PC

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 2-3: Anschlußbelegung der Diagnoseschnittstelle

Pin	Signal	Funktion
1	-	reserviert
2	RXD	Received Data
3	TXD	Transmitted Data
4	-	reserviert
5	GND	Digital Ground
6	-	reserviert
7	RTS	Request to send (nicht firmwareunterstützt)
8	CTS	Clear to send (nicht firmwareunterstützt)
9	-	reserviert

Zur Verbindung der Anschaltbaugruppe mit einem PC als Diagnosegerät sind nur die Signale *TXD*, *RXD* und *GND* erforderlich.

Fernbusschnittstelle

Über die Fernbusschnittstelle wird der Fernbus (2-Leiter) des IBS-Systems angeschlossen. Sie ist auf der Frontblende (Kartenhalter) der Anschaltbaugruppe als 9-polige Sub-D-Buchse ausgeführt und galvanisch vom Potential des Hosts getrennt. Das Gehäuse der Buchse ist leitend mit dem PC-Kartenhalter verbunden.

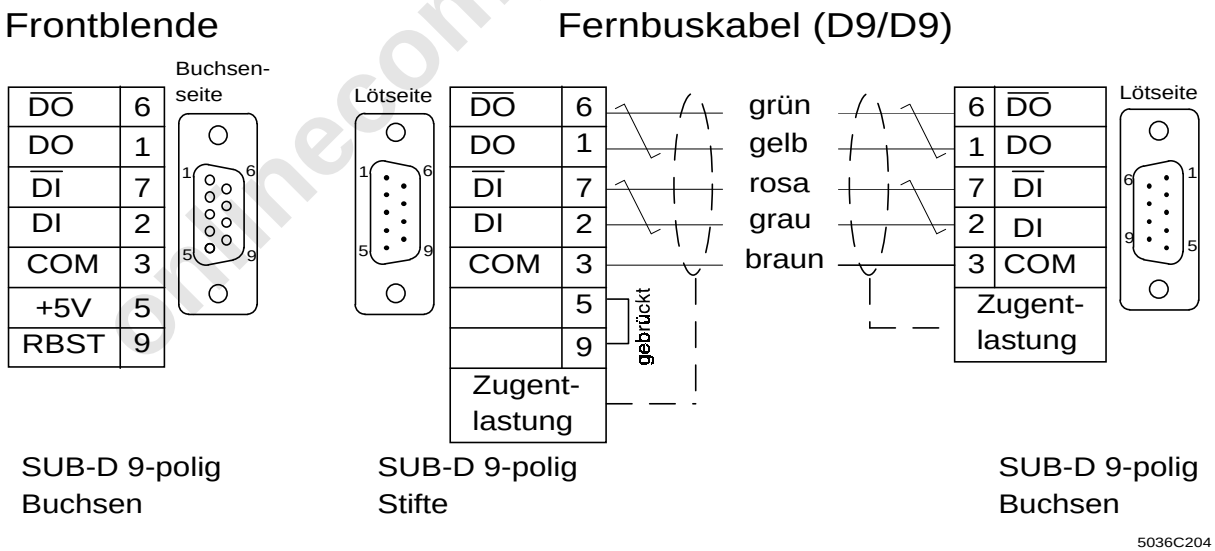

Bild 2-4: Fernbusschnittstelle und Beispiel für ein Fernbuskabel (Kabeltyp D9/D9)

Die Brücke zwischen Pin 5 und Pin 9 zeigt der weiterführenden Fernbusschnittstelle eines IBS-Teilnehmers, daß das weiterführende Fernbuskabel angeschlossen ist.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 2-4: Anschlußbelegung der Fernbusschnittstelle

Pin	Signal
1	DO
2	DI
3	COM
4	reserviert
5	+5 V
6	DO
7	DI
8	reserviert
9	RBST

Detailierte Hinweise zur Kabelspezifikation, allen IBS-Kabeltypen und zur Installation Ihres IBS-Systems entnehmen Sie bitte dem IBS-Installationshandbuch *IBS SYS INST UM* (Artikel-Nr. 27 54 28 6).

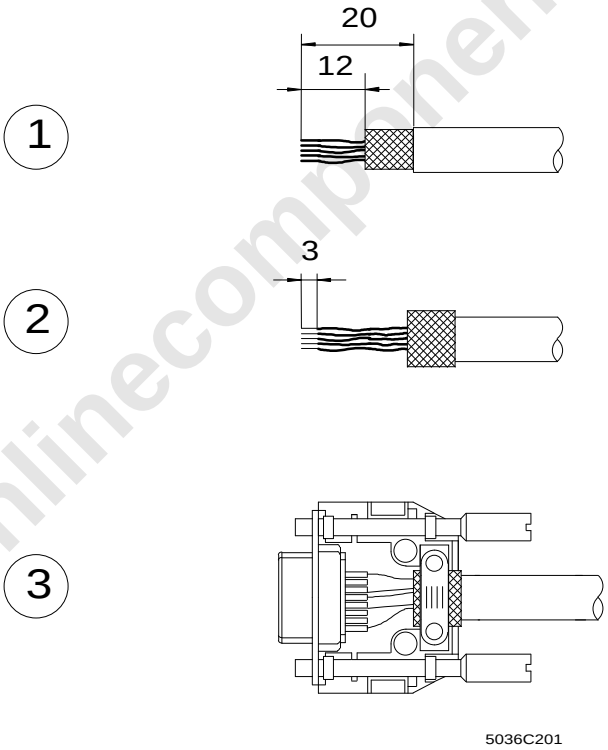

Bild 2-5: Fachgerechtes Anschließen eines Fernbussteckers (Sub-D9)

Schließen Sie den Stecker auf folgende Weise an:

1 Setzen sie den Kabelmantel um 20 mm ab und kürzen Sie das Schirmgeflecht um 12 mm.

2 Isolieren Sie die Aderenden um 3 mm ab und legen Sie das Schirmgeflecht gleichmäßig nach hinten um den Kabelmantel.

3 Nach dem Anlöten der Aderenden klemmen Sie das auf dem Kabelmantel liegende Schirmgeflecht großflächig unter die leitende Zugentlastung.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Für einen störungssicheren Betrieb des InterBus-S muß das Schirmgeflecht des Fernbuskabels an der Anschaltbaugruppe geerdet werden. Verwenden Sie hierzu ausschließlich Sub-D-Stecker mit metallisiertem Gehäuse oder Metallgehäuse. Achten Sie unbedingt darauf, daß das Schirmgeflecht des Fernbuskabels großflächig unter der leitenden, mit dem Steckergehäuse leitend verbundenen Zugentlastung aufliegt. Erden Sie die Anschaltbaugruppe über den Erdungsbolzen (Bild 2-2).

2.4 Funktionseinheiten der Anschaltbaugruppen

2.4.1 AT-Bus-Interface

Das AT-Bus-Interface ist mit einem 8 Bit breiten Datenbus sowie einem 24 Bit breiten Adreßbus realisiert. Es umfaßt neben der Pufferung der Daten-, Adreß- und Steuerleitungen die Adreßdekodierung sowie die Einstellmöglichkeiten für den 8 Byte I/O-Adreßbereich und das 4 KByte Memory-Fenster zum MPM. Für die Datenübertragung zwischen den Aufsteckboards liegen außerdem Interruptsignale aus der MPM-Steuerlogik am AT-Bus an, die auf Interrupteingänge des PC aufgeschaltet werden können. Im folgenden wird die Funktionalität des AT-Bus-Interfaces beschrieben.

2.4.1.1 I/O-Basisadresse im Host

Pro Anschaltbaugruppe wird ein I/O-Adreßbereich von 8 Byte für Status-Register, Mapping-Register und Steuer-Ports benötigt. Die Basisadresse dieses Adreßbereichs ist über DIP-Schalter auf eine von sechzehn vorgegebenen Adressen einstellbar und muß vor Inbetriebnahme der Anschaltbaugruppe festgelegt werden. Achten Sie darauf, daß kein Adreßkonflikt mit anderen PC-Einschubkarten des Hosts entsteht. Kapitel 5 zeigt die einstellbaren Basisadressen sowie die zugehörige Einstellung des DIP-Schalters auf dem Grundboard.

Das AT-Bus-Interface ist so ausgelegt, daß vier Anschaltbaugruppen auf einer gemeinsamen I/O-Basisadresse des Hosts betrieben werden können. Die vier Anschaltbaugruppen werden dann in Abhängigkeit von einer über einen DIP-Schalter einstellbaren Boardnummer (1 bis 4) mit einem Offset von 0_{hex}, 8_{hex}, 10_{hex} oder 18_{hex} zur Basisadresse angesprochen. In Kapitel 5 wird der Zusammenhang zwischen Boardnummer und dem Offset zur eingestellten Basisadresse sowie die zugehörige Einstellung des DIP-Schalters dargestellt.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

2.4.2 Multi-Port-Memory

Das Multi-Port-Memory (MPM) der Anschaltbaugruppen dient zum Austausch von Daten zwischen dem Host und den beiden Aufsteckboards. Das MPM ist als statisches RAM (SRAM) ausgeführt und hat eine Speicherkapazität von 64 KByte.

Beachten Sie bei Zugriffen auf das MPM folgendes. Das IBS-Masterboard legt seine Daten im Motorola-Format (68xxx-Familie) in das MPM. Die Prozessoren des Hosts und des COPs erwarten Daten stets im Intel-Format. Die Reihenfolge der Byte-Adressen (Numerierung) innerhalb eines Datenwortes ist bei diesen Formaten entgegengesetzt. Verwenden Sie für Zugriffe auf das MPM die mitgelieferten Makros, die beim Zugriff auf ein Datenwort das High- und das Low-Byte entsprechend vertauschen (siehe Handbuch zur Treibersoftware IBS PC CB SWD UM).

Im MPM befinden sich das Mailbox-Interface (MXI) für Kommandos und Meldungen sowie das Data-Interface (DTI) für Prozeßdaten. Außerdem werden eine Reihe von Hardware-Registern in den Adreßbereich des MPM eingeblendet. Sie dienen zum Austausch von Status- und Handshake-Informationen.

Für jeden Teilnehmer (Node) am Datenverkehr über das MPM sind eigene Speicherbereiche vorgesehen. Jeder Teilnehmer darf in seinem eigenen Bereich schreiben, in den Bereichen der anderen Teilnehmer nur lesen.

2.4.2.1 Zugriffsverfahren auf das MPM

Da die Teilnehmer vollkommen asynchron auf das MPM zugreifen können, wären folgende Zugriffskonflikte denkbar:

- das gleichzeitige oder sich überlappende Lesen/Schreiben der gleichen Speicherstelle,
- das Lesen eines logisch zusammengehörigen Datenbereichs durch einen Teilnehmer, während dieser vom anderen Teilnehmer beschrieben wird.

Der erste Konflikt wird durch schaltungstechnische Maßnahmen innerhalb der Zugriffsverwaltung des MPM verhindert. Bei gleichzeitigen Zugriffen gilt folgende Prioritätsfolge:

- Priorität 1 hat der Host
- Priorität 2 hat Aufsteckboard 1 (IBS-Masterboard)
- Priorität 3 hat Aufsteckboard 2 (Coprozessorboard)

Bei sich überlappenden Zugriffen wird immer erst der laufende Zugriff beendet.

Der zweite Konflikt wird dadurch gelöst, daß die Anzahl der zusammenhängenden Byte-Zugriffe pro Datentransfer variabel einstellbar ist. Die MPM-Steuerlogik stellt dafür Datenkonsistenzen von 8, 16, 32 und 48 Bit zur Verfügung. Die Zugriffsverwaltung des MPMs hält dabei die MPM-Zugriffe der anderen Teilnehmer zurück, bis der letzte Byte-Zugriff eines Datentransfers beendet ist.

Bei einem Zugriffskonflikt im MPM wird ein Teilnehmer über ein Ready-Signal so lange angehalten, bis der andere Teilnehmer seinen Zugriff beendet hat. Der Zugriff eines Teilnehmers ist erst dann beendet, wenn er alle Daten-Bytes ent-

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

sprechend der eingestellten Datenkonsistenz abgeholt hat. Die Byte-Zugriffe eines MPM-Datentransfers müssen zeitlich unmittelbar hintereinander erfolgen, da die MPM-Steuerlogik nach einer bestimmten Zeit die Zugriffssperre zum MPM wieder aufhebt (Timeout). Die Funktionen der Treibersoftware berücksichtigen dies automatisch, so daß Sie in Ihrem Anwenderprogramm keine besonderen Vorkehrungen vornehmen müssen.

2.4.3 Interrupt-Funktionen

Zwischen den Teilnehmern des MPM sind Zugriffsverfahren mit unterschiedlichen Zugriffsprotokollen definiert. Die Hardware des MPM unterstützt die Protokolle mit *Handshake-Interrupts*. Außerdem gibt es auf den Schnittstellen der Teilnehmer je eine Interruptleitung, die den Systemausfall eines der Teilnehmer meldet. Der Host kann von beiden Aufsteckboards durch den *Host-Interrupt* unterbrochen werden.

2.4.4 Spannungsüberwachung, Reset-System

Für die sichere Funktion der Anschaltbaugruppe ist eine ausreichende Betriebsspannungsversorgung (5V) über den AT-Bus erforderlich. Bei Unterschreiten der Mindestspannung löst eine integrierte Spannungsüberwachung einen *Reset* der Anschaltbaugruppe aus. Somit werden Fehlfunktionen am InterBus-S durch undefinierte Zustände während des Ein-/Ausschaltens des Hosts und durch Spannungseinbrüche vermieden. Ein Reset der Anschaltbaugruppe wird unter folgenden Bedingungen aktiviert:

- Ansprechen der Spannungsüberwachung
- Reset-Signal auf dem AT-Bus (für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T abschaltbar, siehe Bild 2-1, Position 15)
- Betätigung des Reset-Tasters auf der Frontblende (PC-Kartenhalter) oder des Reset-Tasters an der oberen Kante der Anschaltbaugruppe

Um eine unbeabsichtigte Betätigung des Reset-Tasters auf der Frontblende zu vermeiden, ist dieser verdeckt angeordnet (siehe Bild 2-2). Außerdem besteht die Möglichkeit, beide Taster durch Ziehen einer Steckbrücke (siehe Bild 2-1) außer Funktion zu setzen.

Tabelle 2-5: Funktion der Steckbrücke für den Reset-Taster

Steckbrücke S105	Reset-Taster
gesteckt	in Funktion
offen	gesperrt

Durch diesen Reset werden folgende Funktionseinheiten zurückgesetzt:
- das AT-Bus-Interface der Anschaltbaugruppe
- die MPM-Steuerlogik
- die beiden Aufsteckboards

Die Aufsteckboards können außerdem einzeln durch die Software zurückgesetzt werden.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

2.4.5 Watchdog zur Überwachung des Hosts

Während die Aufsteckboards eigene Watchdogschaltungen zur Überwachung ihrer Funktion haben, wurde zur Überwachung des Hosts eine Watchdogschaltung in das AT-Bus-Interface auf dem Grundboard der Anschaltbaugruppen eingebaut. Die Funktion dieses Watchdogs ist in Kapitel 4 näher beschrieben.

2.4.6 Stromversorgung

Die Stromversorgung der Anschaltbaugruppe erfolgt normalerweise über die Direktsteckverbinder aus der 5V-Systemspannung des Hosts. Bei Verwendung bestimmter Flash-EPROM-Typen auf dem Coprozessorboard wird zusätzlich die 12V-Spannungsversorgung des AT-Bus zur Programmierung dieser Flash-Eproms benötigt (Stromverbrauch ca. 40mA).

2.4.6.1 Galvanisch getrennte IBS-Spannungsversorgung

Zum galvanisch getrennten Betrieb des InterBus-S benötigen die Treiberbausteine eine potentialfreie Spannungsversorgung. Auf dem Grundboard befindet sich ein DC/DC-Wandler, der diese Spannung aus der 5V-Systemspannung des PCs erzeugt. Er stellt eine Ausgangsspannung von 5 V bei einem maximalem Strom von 200 mA zur Verfügung, die aus Gründen der Störableitung kapazitiv an PE gekoppelt ist.

Erden Sie die Anschaltbaugruppe über den Erdungsbolzen (Bild 2-2).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel **3**

Technische Beschreibung der Coprozessorboards

Nur für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T.

Dieses Kapitel informiert Sie über

- den Aufbau und die Komponenten der Coprozessorboards

3 Technische Beschreibung der Coprozessorboards 3-3

3.1 Kurzbeschreibung 3-3
3.2 Mechanischer Aufbau 3-5
3.3 Schnittstellen des Coprozessorboards 3-6
3.3.1 Grundboard-Schnittstelle 3-6
3.3.2 Serielle Schnittstelle 3-6
3.4 Funktionseinheiten des Coprozessorboards 3-8
3.4.1 Prozessor/Chipset 3-8
3.4.1.1 Komponenten des Chipsets 3-9
3.4.1.2 I/O-Adreßbereich des Coprozessorboards 3-9
3.4.1.3 Interruptbelegung des Coprozessorboards 3-10
3.4.2 Speicher des Coprozessorboards 3-10
3.4.2.1 EPROM 3-10
3.4.2.2 Statisches RAM 3-11
3.4.2.3 Dynamisches RAM 3-12
3.4.3 MPM-Interface 3-13
3.4.4 Mapping-Register 3-14
3.4.5 Serielle Schnittstelle des Coprozessorboards 3-14
3.4.6 Sicherheitseinrichtungen des Coprozessorboards 3-14
3.4.6.1 Spannungsüberwachung des Coprozessorboards 3-14
3.4.6.2 Resetsystem des Coprozessorboards 3-14
3.4.6.3 Watchdog des Coprozessorboards 3-15
3.4.7 Echtzeituhr des Coprozessorboards 3-15
3.4.8 Stromversorgung des Coprozessorboards 3-16
3.4.8.1 Batterie-Pufferung 3-16

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3 Technische Beschreibung der Coprozessorboards

(für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

3.1 Kurzbeschreibung

Die Coprozessorboards (kurz COP) sind für Steuerungsaufgaben optimierte Industrie-PCs in Form von Aufsteckkarten für die Grundboards der Controller IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T.

Tabelle 3-1: Übersicht Coprozessorboards

	IBS PC CB/COP/I-T	IBS PC CB/RTX486/I-T
Coprozessorboard	COP386	COP486
CPU	386SX-25, 25 MHz	486 SXLC-40, 40 MHz double clock, 8 KByte Cache
Arbeitsspeicher	2 MByte	2 MByte
Statisches RAM (batteriegepuffert)	128 KByte	128 KByte - als RAM-Disk (D:\)
EPROM 1	128 KByte EPROM für Betriebssystem	256 KByte Flash-EPROM: - 128 KByte für Betriebssystem - 128 KByte für Device-Driver und Anwender
EPROM 2	—	256 KByte Flash-EPROM für Anwender
Betriebsystem	TDOS	RTXDOS
Zugriff vom PC über	Entwicklungsumgebung TDOS-PRO*	Terminalprogramm DPCON.EXE (befindet sich auf der Tool-Diskette)

* Die Entwicklungsumgebung TDOS-PRO (IBS PC COP SWT) ist nicht kompatibel zum COP486 des IBS PC CB/RTX486/I-T und wird dafür auch nicht benötigt.

Mit Hilfe dieser Coprozessorboards ist es möglich, den Prozessor des Hosts zu entlasten und somit zu einem deutlichen Performance-Gewinn des PC-Systems beizutragen.

Zentrale Funktionseinheit der Coprozessorboards ist ein Intel-kompatibler Mikroprozessor. Die Zugriff auf das Coprozessorboard erfolgt entweder über das Multi-Port-Memory (MPM) oder über eine PC-kompatible serielle Schnittstelle. Die Coprozessorboards sind weitgehend PC-kompatibel aufgebaut (ausgenommen Tastatur, Bildschirm und Hard-Disks), wodurch eine Programmierung mit Standard-Tools wie z.B. Turbo Pascal oder Turbo C ermöglicht wird. Der 2 MByte große Arbeitsspeicher bietet auch umfangreichen Software-Projekten genügend Platz. Die im PC-Bereich bekannten EMS-Features sind auf den Coprozessorboards im vollen Umfang nutzbar.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Zur dauerhaften Ablage von Informationen oder Programmen stehen dem Anwender 128KByte batteriegepuffertes SRAM, sowie auf dem COP486 zusätzlich 384 KByte Flash-EPROM zur Verfügung (128KByte im Flash-EPROM 1 und 256KByte im Flash-EPROM 2).

Das Betriebssystem des Coprozessorboards ist ebenfalls im Flash-EPROM 1 gespeichert.

Zur zeitabhängigen Steuerung von Maschinen und Automaten über den Inter-Bus-S ist auf dem Coprozessorboard eine quarzgesteuerte Echtzeituhr (PC-kompatibel) integriert.

Um im Industrieeinsatz eine hohe Datensicherheit zu gewährleisten, besitzen die Coprozessorboards eine aufwendige Watchdog- und Reset-Schaltung. Diese ermöglicht dem Nutzer, Systemfehler rechtzeitig zu erkennen und gegebenenfalls Anlagen abzuschalten. Nach dem Auslösen des Schutzmechanismus sind die Coprozessorboards in der Lage, das Anwenderprogramm selbständig wieder zu starten.

Die auf den Coprozessorboards enthaltene Interrupt-Steuerung informiert weitere MPM-Teilnehmer (Host, IBS-Master) über eventuell aufgetretene Funktionsfehler. Auf der Frontblende des Grundboards informieren zwei Leuchtdioden den Anwender visuell über den Funktionszustand des Coprozessorboards.

Die MS-DOS-kompatiblen Betriebssysteme *TDOS* und *RTXDOS* stellen eine speziell auf die Belange der Coprozessorboards abgestimmte Software-Basis zur Verfügung. Somit ist es möglich, bereits bestehende DOS-Applikationen auf den Coprozessorboards zu implementieren. Für Echtzeitanwendungen können zusätzliche Real-Time-Multitasking-Betriebssysteme eingesetzt werden (auf Anfrage).

Die Datenübertragung zwischen dem Betriebssystem TDOS des COP386 und dem Entwicklungs-Tool *TDOS-PRO (IBS PC COP SWT,* Artikel-Nr. 27 52 12 3*)* kann im Software-Entwicklungsstadium über das MPM oder die serielle Schnittstelle des COPs erfolgen.

Das Betriebsystem *RTXDOS* des COP486 kann über das auf dem Host zu startende Programm *RFSERVER.EXE* direkt auf dessen Massenspeicher (Diskettenlaufwerke, Festplatten) zugreifen. *RFSERVER.EXE* befindet sich auf der Diskette, die zum Lieferumfang dieses Handbuchs gehört.

Tabelle 3-2: Laufwerke des COP 486

Laufwerk	Datenträger	Nutzung
A:\	128 KByte im Flash-EPROM1	Device-Driver und Anwenderprogramme
B:\	—	nicht genutzt
C:\	256 KByte Flash-EPROM2	Anwenderprogramme
D:\	128 KByte statisches RAM (batteriegepuffert)	Anwenderprogramme, Daten
E:\	Laufwerk C:\ des PCs (Festplatte des Hosts)	Anwenderprogramme, Daten

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

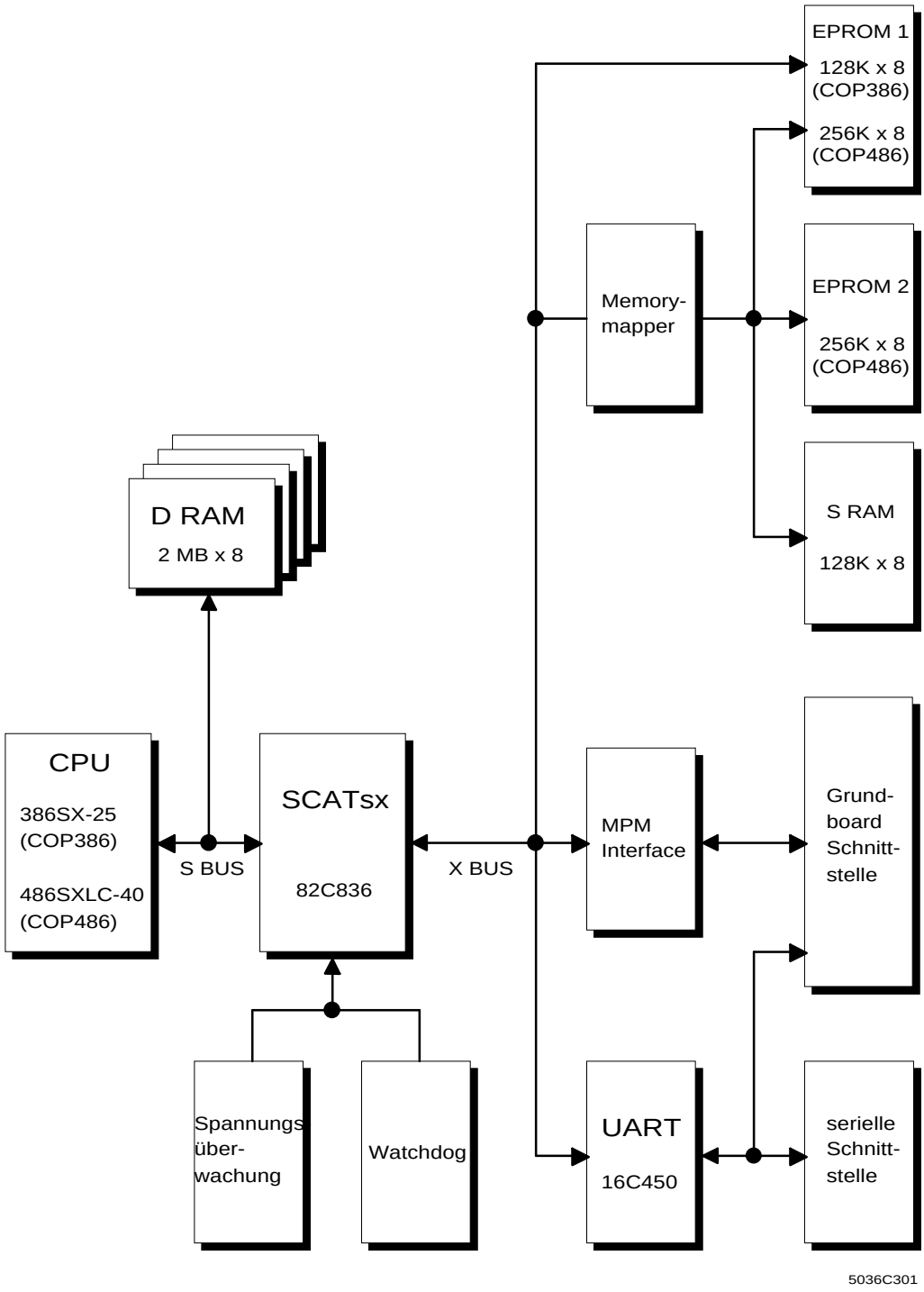

Bild 3-1: Blockschaltbild des Coprozessorboards

3.2 Mechanischer Aufbau

Das Coprozessorboard (COP) besitzt zur Verbindung mit dem Grundboard an zwei Seiten eine 58-polige Stiftleiste. Die Orientierung zur Montage des COPs ergibt sich aus den asymmetrisch liegenden Bohrungen (je eine auf dem Coprozessorboard und dem Grundboard). An der unteren Seite des COPs ist ein vierpoliger Stecker zum Anschluß des Batterie-Packs und ein 10-poliger Stecker für die serielle Schnittstelle vorhanden.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Um eine hohe Flexibilität des Coprozessorboards zu erreichen, wurde auf Konfigurationsmöglichkeiten über DIP-Schalter oder Steckbrücken verzichtet. Die hardwarespezifischen Einstellungen werden automatisch durch die Treibersoftware vorgenommen.

3.3 Schnittstellen des Coprozessorboards

Dieses Kapitel beschreibt die verschiedenen Schnittstellen des Coprozessorboards und ihre Signal-Belegung.

3.3.1 Grundboard-Schnittstelle

Funktionen der Schnittstelle:

- Datenaustausch zwischen Grundboard und Coprozessorboard: Die Datenbreite beträgt 16 Bit. Alle Daten- und Adreßleitungen sind gepuffert.
- Bereitstellung von Steuersignalen: Interruptsteuersignale, Schreib- und Lesesteuerung, Reset- und Watchdog-Steuerung
- Versorgungs- und Sonderleitungen: Spannungsversorgung, Batteriepufferung, LED-Ansteuerung, RS232-Schnittstelle

3.3.2 Serielle Schnittstelle

Das Coprozessorboard besitzt eine serielle Schnittstelle (RS-232-Pegel), die der COM 1 eines Standard-PCs entspricht, und die Sie z.B. für folgende Anwendungen nutzen können:

- Wenn der Zugriff der Entwicklungsumgebung *TDOS-PRO (IBS PC COP SWT)* auf den COP386 nicht über das MPM erfolgen soll. Genauere Informationen finden Sie im Handbuch zur Entwicklungsumgebung *TDOS-PRO*, das zum Lieferumfang dieser Software gehört.
- Zum Remote-Debugging (z.B.: mit Turbo/Borland C Remote-Debugger). Genauere Informationen hierzu finden Sie im Handbuch Ihres Compilers.
- eigene Anwendungen

Adapterkabel *IBS PC COP RS 232 CAB*

Diese voll PC-kompatible Schnittstelle ist am Rand des Coprozessorboards als 10-polige Stiftleiste realisiert. Sie kann mit dem Adapterkabel *IBS PC COP RS232 CAB* auf einen Sub-D-Stecker geführt werden, den Sie z.B. auf einem zusätzlichen PC-Kartenhalter (freier Steckplatz) montieren. Das Adapterkabel kann unter der Artikel-Nr. 27 51 65 8 bestellt werden. Es gehört außerdem zum Lieferumfang der Entwicklungsumgebung *TDOS-PRO (IBS PC COP SWT*).

Die Buchsenleiste dieses Adapterkabels ist durch einen Zapfen neben Pin 5 gekennzeichnet.

- Falls die 10-polige Stiftleiste in einer verpolungssicheren Steckerwanne sitzt, stecken Sie die Buchsenleiste so in diese Steckerwanne, daß der Zapfen zum Grundboard zeigt (siehe Bild 3-2).

- Falls Sie eine älteres Coprozessorboard ohne verpolungssichere Steckerwanne besitzen, stecken Sie die Buchsenleiste so auf die Stiftleiste, daß der Zapfen zum Coprozessorboard zeigt (180° gedreht zur Darstellung in Bild 3-2).

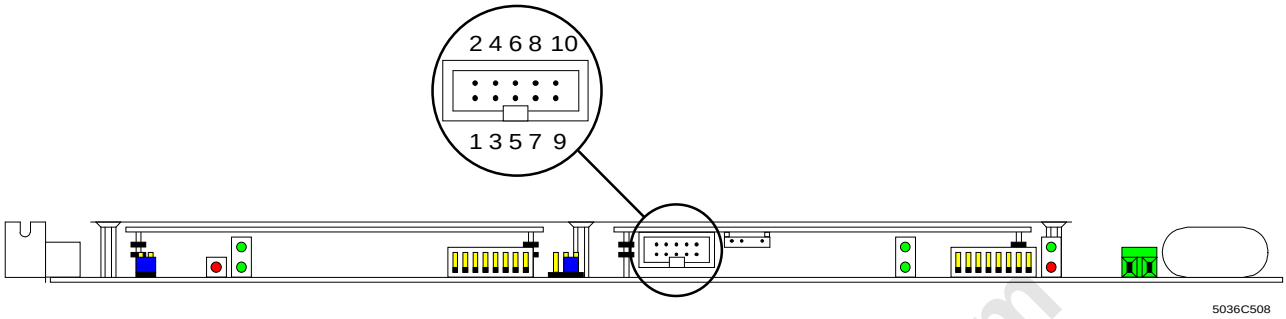

Bild 3-2: Pinanordnung der in der Steckerwanne des COPs

Tabelle 3-3: Belegung des Adapterkabels

Pinbelegung Coprozessorboard	Funktion	Signal	Pinbelegung Sub-D 9
1	Data Carrier Detect (Eingang)	DCD	1
2	Data Set Ready (Eingang)	DSR	6
3	Received Data (Eingang)	RXD	2
4	Request To Send (Ausgang)	RTS	7
5	Transmitted Data (Ausgang)	TXD	3
6	Clear To Send (Eingang)	CTS	8
7	Data Terminal Ready (Ausgang)	DTR	4
8	Ring Indicator (Eingang)	RI	9
9	Signal Ground	GND	5
10	reserviert	-	-

Entwicklungskabel

Die Verbindung der auf einen Sub-D-Stecker herausgeführten seriellen Schnittstelle des COP386 und des PCs, auf dem die Entwicklungsumgebung *IBS PC COP SWT (TDOS-PRO)* läuft, erfolgt über ein Entwicklungskabel. Dieses ist als sogenanntes Null-Modemkabel ausgeführt und entspricht der folgenden Darstellung.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

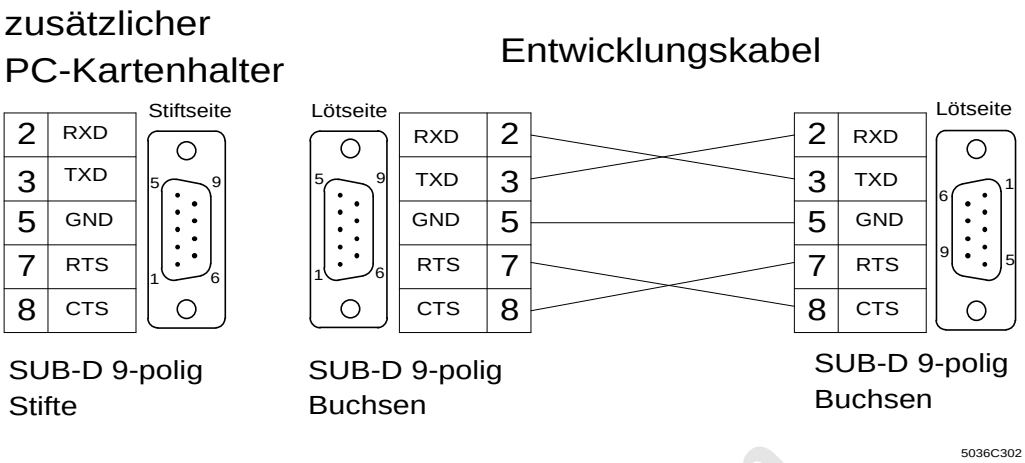

Bild 3-3: Entwicklungskabel für den COP386

Das Entwicklungskabel gehört zum Lieferumfang der Entwicklungsumgebung TDOS-PRO (IBS PC CB COP SWT, Artikel-Nr. 27 52 12 3).

3.4 Funktionseinheiten des Coprozessorboards

Die Funktionalität des Coprozessorboards läßt sich in sieben Bereiche gliedern:

- Prozessor/Chipset
- Speicher (DRAM; SRAM; EPROM)
- MPM-Interface (Multi-Port-Memory)
- Memory-Mapper
- Serielle Schnittstelle (UART)
- Spannungsüberwachung und Reset-System
- Stromversorgung

In den folgenden Abschnitten werden diese Funktionseinheiten, deren Einstellung und Bedeutung näher beschrieben.

3.4.1 Prozessor/Chipset

Den Rechnerkern stellt für den *COP386* ein 386SX-25 und für den COP486 ein 486 SXLC-40 Mikroprozessor jeweils in Verbindung mit dem Single-Chipset 82C836 dar.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3.4.1.1 Komponenten des Chipsets

Bei dem Baustein 82C836 handelt es sich um einen hochintegrierten Single-Chip-AT-Chipset. Er beinhaltet alle für einen PC notwendigen Schaltungskomponenten wie z.B:

- eine 146818-kompatible Real-Time-Clock mit 114 Bytes CMOS-RAM
- zwei 8237-kompatible DMA-Controller
- zwei 8259-kompatible Interrupt-Controller
- einen 8254-kompatiblen Zähler/Zeitgeber
- einen 82284-kompatiblen Clock-Generator und Ready-Interface
- einen 82288-kompatiblen Bus-Controller
- einen DRAM-Controller / DRAM-Refresh-Controller
- vier EMS-Page-Register (LIM EMS 4.0 und 3.2)

3.4.1.2 I/O-Adreßbereich des Coprozessorboards

Der I/O-Adreßbereich des Coprozessorboards ist neben den im IBM-PC üblichen Standard-Adressen um einige Sonderfunktionen erweitert. Im Adreßbereich von 0200_{hex} bis $02BF_{hex}$ (laut IBM reserviert für die Prototypenkarte) werden verschiedene Funktionen des Watchdogs aktiviert. Im Adreßraum von 0280_{hex} bis $02BF_{hex}$ liegt das Mapping-Register (ein Register zur Steuerung von kontrollerspezifischen Memory-Funktionen) inklusive einigen Bits für Sonderfunktionen wie z.B. zur Steuerung des Watchdogs.

Tabelle 3-4: I/O-Adreßbereich des Coprozessorboards

I/O-Adresse:			Funktion:
$03C0_{hex}$	bis	$03FF_{hex}$	Bereich für die serielle Schnittstelle COM1
0340_{hex}	bis	$03BF_{hex}$	frei
0300_{hex}	bis	$033F_{hex}$	Watchdog-Clear
$02C0_{hex}$	bis	$02FF_{hex}$	frei
0280_{hex}	bis	$2BFF_{hex}$	Mapping-Register
0240_{hex}	bis	$027F_{hex}$	Watchdog Enable
0200_{hex}	bis	$023F_{hex}$	Watchdog Trigger
$00E0_{hex}$	bis	$01FF_{hex}$	frei
$00C0_{hex}$	bis	$00DF_{hex}$	DMA-Controller 2
$00A0_{hex}$	bis	$00BF_{hex}$	Interrupt-Controller 2 (Slave)
0080_{hex}	bis	$009F_{hex}$	DMA-Seitenregister / NMI-Mask
0070_{hex}	bis	$007F_{hex}$	Real-Time-Clock
0060_{hex}	bis	$006F_{hex}$	Systemregister
0040_{hex}	bis	$005F_{hex}$	Timer / Counter
0020_{hex}	bis	$003F_{hex}$	Interrupt-Controller 1 (Master)
0000_{hex}	bis	$001F_{hex}$	DMA-Controller 1

Die Basisadresse für die serielle Schnittstelle COM1 liegt, wie bei IBM-kompatiblen PCs üblich, auf der Adresse $03F8_{hex}$.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3.4.1.3 Interruptbelegung des Coprozessorboards

Auch das Interruptsystem entspricht weitgehend dem üblichen Standard eines IBM-PCs. Bei der Ansprache des MPMs wird ein sogenannter Handshake-Interrupt benutzt, der die Interrupt-Nr. 5 besitzt. Im Falle eines Systemfehlers eines anderen MPM-Teilnehmers (Host, IBS-Master) wird auf dem Coprozessorboard ein *Non-Maskable-Interrupt* (NMI) ausgelöst. Tritt hingegen ein Fehler auf dem Coprozessorboard auf, wird bei eingeschaltetem Watchdog ein Interruptsignal zu den anderen Teilnehmern erzeugt. Dieses Signal gelangt auf die Grundboard-Schnittstelle und wird mit *SRQ2L* bezeichnet. Die folgende Tabelle gibt eine Übersicht über die verwendete Interruptbelegung des Coprozessorboards.

Tabelle 3-5: Interruptbelegung des Coprozessorboards

Interrupt	Funktion
IRQ0	Timer Kanal 0
IRQ1	frei
IRQ2	Slave-Interrupt-Controller
IRQ3	frei
IRQ4	COM1
IRQ5	Handshake-Interrupt MPM
IRQ6	Timer Kanal 2
IRQ7	frei
IRQ8	Real-Time-Clock
IRQ9	frei
IRQ10	frei
IRQ11	frei
IRQ12	frei
IRQ13	frei
IRQ14	frei
IRQ15	frei

onlinecomponents.com

3.4.2 Speicher des Coprozessorboards

Die Aufteilung des Speichers ergibt sich wie folgt:

- 128 KByte statisches RAM (SRAM), batteriegepuffert mit einer Datenbreite von 8 Bit
- 2 MByte DRAM mit einer Datenbreite von 16 Bit

3.4.2.1 EPROM

Auf dem Coprozessorboard sind zwei 32-polige PLCC EPROM-Sockel zur Aufnahme von 128 KByte oder 256 KByte EPROMs oder Flash-EPROMs vorgesehen.

IBS PC CB/COP/I-T

Im Auslieferungszustand ist Sockel 1 mit einem 128 KByte EPROM für das Betriebssystem bestückt. Sockel 2 ist nicht bestückt.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

IBS PC CB/RTX486/I-T

Im Auslieferungszustand sind Sockel 1 und Sockel 2 mit jeweils 256 KByte Flash-EPROM bestückt.

3.4.2.2 Statisches RAM

Zur Aufbewahrung von Daten (z.B. Anlagenzustände) stehen auf dem Coprozessorboard 128 KByte statisches RAM (SRAM) zur Verfügung. Dieser Speicher ist batteriegepuffert und 8 Bit breit organisiert.

IBS PC CB/COP/I-T

Das SRAM auf dem COP386 ist über das Speichersegment $B0000_{hex}$ in zwei 64 KByte-Blöcken adressierbar. Der Zugriff erfolgt über Funktionen der Treibersoftware.

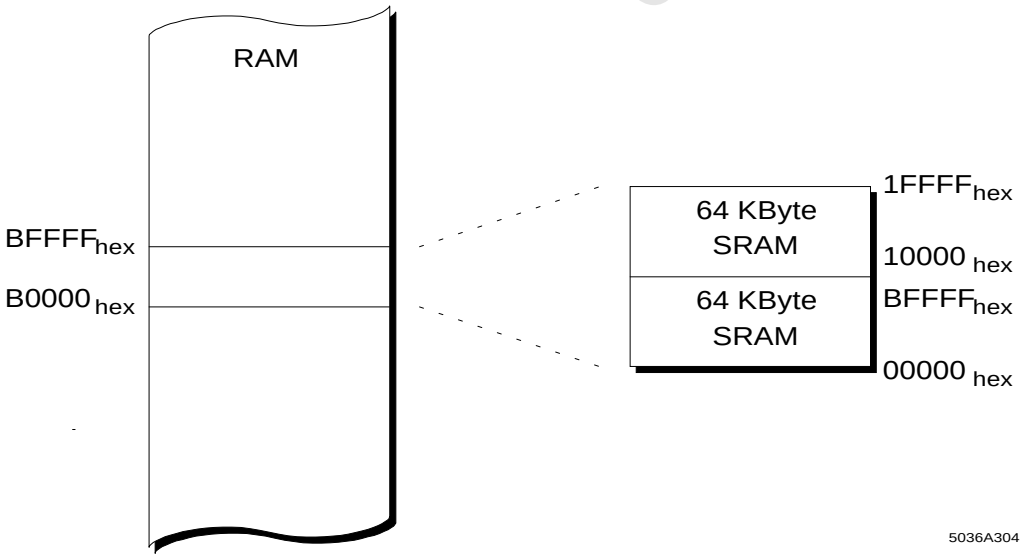

Bild 3-4: SRAM-Aufteilung auf dem COP386 der IBS PC CB/COP/I-T

IBS PC CB/RTX486/I-T

Das SRAM auf dem COP486 wird als Laufwerk D:\ angesprochen und kann so über übliche DOS-Funktionen als RAM-Disk verwendet werden.

Mit dem Programm *RDRCONF.EXE* läßt sich die Zugriffssart auf das SRAM des COP486 parametrieren. Sie können z.B. auch die für den COP386 beschriebene Adressierung in zwei 64 KByte-Blöcken einstellen. *RDRCONF.EXE* gehört zum Lieferumfang dieses Handbuchs.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3.4.2.3 Dynamisches RAM

Das Coprozessorboard enthält als Arbeitsspeicher 2 MByte dynamisches RAM (DRAM). Das DRAM wird mit 0 RAM-Wait-States und im Page-Interleave-Mode betrieben, was eine Steigerung der Rechengeschwindigkeit bedeutet.

Da das Coprozessorboard wegen des fehlenden Tastatur-Controllers im Real-Mode arbeitet, ist die Nutzung des zweiten Megabytes (100000_{hex} bis $1FFFFF_{hex}$) nur durch Verwendung der EMS-Funktionen möglich. Hierzu befindet sich auf der Tool-Diskette ein entsprechender EMS-Treiber.

Tabelle 3-6: Aufteilung des DRAMs

Megabyte	Speicherbereich			Bank	Zuordnung	Ergänzung
2.	100000_{hex}	bis	$1FFFFF_{hex}$	10 bis 1F	DRAM	XMS/EMS-Speicher
1.	$0F0000_{hex}$	bis	$0FFFFF_{hex}$	F		Bios
	$0E0000_{hex}$	bis	$0EFFFF_{hex}$	E		
	$0D0000_{hex}$	bis	$0DFFFF_{hex}$	D		reserviert
	$0C0000_{hex}$	bis	$0CFFFF_{hex}$	C		
	$0B0000_{hex}$	bis	$0BFFFF_{hex}$	B	Mapping-Fenster	
	$0A0000_{hex}$	bis	$0AFFFF_{hex}$	A	DRAM	Arbeitsspeicher/ Download-RAM
	090000_{hex}	bis	$09FFFF_{hex}$	9		
	080000_{hex}	bis	$08FFFF_{hex}$	8		
	070000_{hex}	bis	$07FFFF_{hex}$	7		
	060000_{hex}	bis	$06FFFF_{hex}$	6		
	050000_{hex}	bis	$05FFFF_{hex}$	5		
	040000_{hex}	bis	$04FFFF_{hex}$	4		
	030000_{hex}	bis	$03FFFF_{hex}$	3		
	020000_{hex}	bis	$02FFFF_{hex}$	2		
	010000_{hex}	bis	$01FFFF_{hex}$	1		Arbeitsspeicher
	000000_{hex}	bis	$00FFFF_{hex}$	0		Systembereich TDOS

Der 4 KByte große Speicherbereich von $B0000_{hex}$ bis $BFFFF_{hex}$ ist als Mapping-Fenster konfiguriert, über das die Funktionen der Treibersoftware auf das MPM, das statische RAM und die EEPROMS zugreifen.

Sie dürfen im Bereich des Mapping-Fensters nicht direkt auf das DRAM des Coprozessorboards zugreifen!

Die Funktionen der Treibersoftware wählen über das Mapping-Register im IO-Bereich des Coprozessorboards automatisch aus, welcher Speicher über das Mapping-Fenster eingeblendet wird.

onlinecomponents.com

3.4.3 MPM-Interface

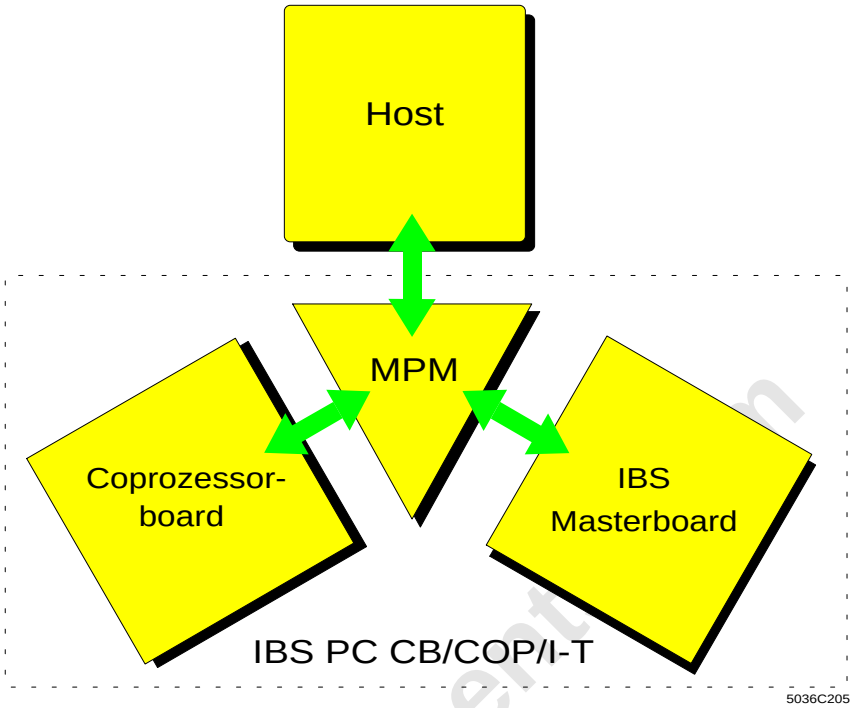

Bild 3-5: Das MPM als zentrale Schnittstelle

Das MPM stellt die zentrale Schnittstelle zwischen den Teilnehmern Host, IBS-Masterboard (MA) und Coprozessorboard (COP) dar. Um eine reibungslose Datenübertragung zwischen den MPM-Teilnehmern sicherzustellen, sind alle Daten- und Adreßleitungen des COPs gepuffert. Die Puffer werden nur freigegeben, wenn das Coprozessorboard das Zugriffsrecht erhält. Die Zugriffsberechtigung wird durch eine Prioritätssteuerung von der MPM-Logik (sie befindet sich auf dem Grundboard) erteilt. Im Zeitraum von der MPM-Anforderung durch das Coprozessorboard bis zur Zugriffserlaubnis des Grundboards wird der Prozessor des COPs in den *Halt-Zustand* versetzt.

Erhält das Coprozessorboard die Zugriffsberechtigung, werden die Daten und Adreßpuffer freigeschaltet, und in Abhängigkeit von der eingestellten MPM-Adresse (die Einstellung erfolgt über das Mapping-Register) beginnt der Datentransfer mit einer Datenkonsistenz von 8, 16, 32 oder 48 Bit. Bei Datenkonsisenzen von 16, 32 oder 48 Bit werden mehrfach 8-Bit-Zugriffe getätigt.

Beachten Sie bei Zugriffen auf das MPM folgendes. Das IBS-Masterboard legt seine Daten im Motorola-Format (68xxx-Familie) in das MPM. Die Prozessoren des Hosts und des COPs erwarten Daten stets im Intel-Format. Die Reihenfolge der Byte-Adressen (Numerierung) innerhalb eines Datenwortes ist bei diesen Formaten entgegengesetzt. Verwenden Sie für Zugriffe auf das MPM die mitgelieferten Makros zur Datenkonvertierung, die High- und Low-Byte entsprechend vertauschen (siehe Handbuch zur Treibersoftware *IBS PC CB SWD UM;* das Handbuch gehört zum Lieferumfang der Treibersoftware *IBS PC CB SWD*).

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3.4.4 Mapping-Register

Eine zentrale Rolle im Hardware-Design des Coprozessorboards spielt der Memory-Mapper. Er liegt im I/O-Bereich des Coprozessorboards. Hierbei handelt es sich um ein 8 Bit breites Register, welches die Umschaltung der einzelnen Memory-Bereiche in das Speichersegment $B0000_{hex}$ bis $BFFFF_{hex}$ ermöglicht. Über diesen Speicherbereich können die EPROMs, das RAM und das MPM-Fenster erreicht werden. Der Zugriff erfolgt jeweils über Funktionen der Treibersoftware.

3.4.5 Serielle Schnittstelle des Coprozessorboards

Die serielle Schnittstelle des Coprozessorboards ermöglicht die Softwareentwicklung und den Programmtest ohne die Benutzung der MPM-Schnittstelle. So ist es z.B. möglich, mit Hilfe des Turbo Debuggers von Borland ein Remote-Debugging zur schnellen Fehlererkennung eigener Software-Applikationen durchzuführen. Die serielle Schnittstelle ist voll PC-kompatibel als COM 1 spezifiziert. Die Basisadresse liegt, wie bei IBM-kompatiblen PCs üblich, auf der Adresse $03F8_{hex}$ und es wird der Interrupt IRQ 4 benutzt. Diese Einstellungen sind nicht konfigurierbar.

3.4.6 Sicherheitseinrichtungen des Coprozessorboards

Die Spannungsüberwachung- und Watchdog-Einheit stellen in Verbindung mit dem Resetsystem beim Ein- und Ausschalten der Versorgungsspannung ein einwandfreies Verhalten des Coprozessorboards sicher. Sie schützen das System außerdem bei einem Einbruch der Versorgungsspannung oder bei Laufzeitfehlern des Anwenderprogramms vor unkontrolliertem Verhalten.

3.4.6.1 Spannungsüberwachung des Coprozessorboards

Die Spannungsüberwachung des Coprozessorboards kontrolliert die +5 V-Versorgungsspannung. Sinkt die Versorgungsspannung unter einen Wert von 4,65 V, löst die Spannungsüberwachung einen System-Reset aus und die Chip-Enable-Leitung des CMOS-RAM wird deaktiviert. Ab diesem Moment wird die externe Akkuspannung zur Pufferung der Real-Time-Clock und des statischen RAMs benutzt. Die einwandfreie Funktion der Spannungsüberwachungsschaltung ist für einen Spannungsbereich von 1,0 V bis 5,25 V gewährleistet.

3.4.6.2 Resetsystem des Coprozessorboards

Das Coprozessorboard besitzt ein umfangreiches Resetsystem für die interne Hardware. Ein Reset wird zum einen durch die internen Funktionseinheiten, Spannungsüberwachung und das Watchdog-System automatisch erzeugt, zum anderen kann ein Reset über ein MPM-Signal ausgelöst werden. Die Resetlogik des Chipsets führt dann die gezielte Zurücksetzung der einzelnen Komponenten des Coprozessorboards wie z.B. Prozessor, serieller Schnittstelle und Mapping-Register durch.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3.4.6.3 Watchdog des Coprozessorboards

Mit Hilfe des Watchdogs kann die einwandfreie Funktion des Anwenderprogramms überwacht werden. Nach dem Einschalten des Watchdogs muß dieser regelmäßig getriggert werden. Bleibt dieser Triggerimpuls länger als 125 ms aus, führt das Coprozessorboard einen Neustart durch. Nach erfolgreichem Booten des Systems kann der Anwender über eine Funktion den Status des Watchdogs ermitteln. Andere Teilnehmer des IBS-Systems werden über das Auslösen des Coprozessor-Watchdogs durch eine Interruptleitung benachrichtigt. Ein Zurücksetzen des Watchdog-Status erfolgt durch Ausschalten des Hosts oder durch den Aufruf der entsprechenden Funktion. Die Freigabe des Watchdogs und die Triggerung werden ebenfalls durch Funktionen durchgeführt. Die Triggerzeit ist fest eingestellt und beträgt 125ms. Nach jeden System-Reset ist die Überwachungsschaltung ausgeschaltet. Ein einmal eingeschalteter Watchdog kann nicht durch die Software ausgeschaltet werden.

Funktionen zur Steuerung des Watchdogs:

- EnableWatchDog()	Watchdog einschalten
- TriggerWatchDog()	Watchdog triggern
- GetWatchDogState()	Watchdog-Statusbit abfragen
- ClearWatchDog()	Watchdog-Statusbit zurücksetzen

Das Handbuch zur Treibersoftware *IBS PC CB SWD UM* beschreibt diese Funktionen detailliert.

3.4.7 Echtzeituhr des Coprozessorboards

Die Coprozessorboards verfügen über eine quarzgesteuerte Echtzeituhr. Sie wird durch das Batteriepack auf dem Grundboard gepuffert.

Die Echtzeituhr ist PC-kompatibel.

- Zur Einstellung von Datum und Uhrzeit auf dem COP386 stellt die Entwicklungsumgebung *TDOS-PRO (IBS PC COP SWT)* das Programm *RTCINIT.EXE* zur Verfügung. Laden Sie das Programm mit dem Entwicklungsumgebung auf das Coprozessorboard herunter und starten Sie es. Danach können Sie Datum und Uhrzeit einstellen.
- Auf dem *COP486* ermöglicht *RTXDOS* die Einstellung von Datum und Uhrzeit mit den üblichen DOS-Befehlen.

Nutzung von Timern

Wie bei einem normalen PC kann der Timer 0, der normalerweise 18 mal pro Sekunde den Interrupt 1C aufruft, auch auf andere Werte eingestellt werden. Der Timer wird wie üblich über die I/O-Adressen 40_{hex} bis 43_{hex} angesprochen. Detaillierte Informationen zur Programmierung des Timers finden Sie in zahlreichen Beiträgen in der PC-Literatur (z.B. Zeitschrift c't, Ausgabe 1988, Heft 4).

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

3.4.8 Stromversorgung des Coprozessorboards

Das Coprozessorboard ist so konzipiert worden, daß im Normalfall eine Versorgungspannung von +5 Volt ausreicht. Eine Ausnahme stellt der Einsatz von Flash-EPROMs mit einer Programmierspannung von 12V dar. Bei Verwendung dieser EPROMs ist neben der 5V-Versorgung zusätzlich eine 12V-Versorgungsspannung notwendig. Auf dem Grundboard der Anschaltbaugruppe ist diese Spannungsversorgung vorgesehen. Die Gesamtstromaufnahme des Coprozessorboards beträgt dank der eingesetzten CMOS-Komponenten nur ca. 400mA. Bei der Flash-EPROM-Programmierung mit 12V Programmierspannung muß dort mit einer zusätzlichen Stromaufnahme von ca. 40mA gerechnet werden.

3.4.8.1 Batterie-Pufferung

Zur Pufferung der RAM-Speicher auf dem Coprozessorboard ist das Grundboard mit einem Standard-Batterie-Pack ausgerüstet, wie er in PCs zur Pufferung der BIOS-Einstellungen verwendet wird. Das Batterie-Pack stellt eine Spannung von 6V zur Verfügung und hat im Betrieb eine Lebensdauer von ca. 2,5 Jahren. Verwenden Sie beim Austausch einen Batteriepack gleicher Baugröße.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel 4

Einbau und erste Inbetriebnahme

Dieses Kapitel informiert Sie über

- Adreßeinstellungen an der Anschaltbaugruppe
- den Einbau der Anschaltbaugruppe in Ihren PC
- die Installation der Treiber-Software

4 Einbau und erste Inbetriebnahme 4-3

4.1 Einstellen der Adressierung 4-3
4.1.1 Basisadresse im IO-Bereich des PCs (IO-Address) 4-3
4.1.2 Boardnummer (Board No.) 4-4
4.2 Einstellen der Boot-Konfiguration 4-6
4.2.1 Kontrolle des InterBus-S (IBS-Control) 4-6
4.2.2 Anlaufverhalten des InterBus-S (IBS-Autostart) 4-7
4.2.3 Automatischer Programmstart aus EPROM (EPROM-Start) . . 4-8
4.2.4 Einstellen des Boot-Laufwerks (Boot Disk) 4-8
4.2.5 Bootverhalten in Bezug auf RFSERVER (Wait for RFSERVER) 4-10
4.2.6 Boot-Verhalten in Bezug auf DPCON (Wait for DPCON) . . . 4-11
4.2.7 Datenübertragung zwischen COP und Entwicklungsumgebung 4-12
4.3 Einstellung der Steckbrücken (Jumper) 4-13
4.3.1 Auswahl der Spannungsversorgung (Power Supply) 4-13
4.3.2 Trennung vom Hardware-Reset des Hosts (PC HW RESET) . 4-14
4.3.3 Deaktivieren der Reset-Taster (Enable/Disable RESET-Button) 4-14
4.4 Anschluß des Batterie-Packs 4-15
4.5 Einbau der Anschaltbaugruppe in den PC 4-15
4.5.1 Serielle Schnittstelle des Coprozessorboards 4-16
4.6 Installation des Device-Drivers 4-16
4.6.1 Device-Driver unter MS-DOS 4-18
4.6.1.1 Installationshilfe für DOS 4-19
4.6.2 Device-Driver unter Microsoft Windows 4-20
4.6.2.1 Installationshilfe für Microsoft Windows 4-21
4.6.3 Device-Driver unter IBM OS/2 4-22
4.6.3.1 Installationshilfe für OS/2 4-22
4.7 Installation der Peripherie 4-23
4.8 Software-Tools für die Inbetriebnahme 4-23
4.9 Inbetriebnahme mit Prozeßdaten-Monitorprogramm 4-24
4.9.1 Das Pull-Down-Menü Functions 4-25
4.9.2 Absetzen von Kommandos mit PCCBMONI 4-28
4.9.3 Das Pull-Down-Menü Options 4-30

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4 Einbau und erste Inbetriebnahme

Die folgenden Unterkapitel beschreiben der Reihe nach alle notwendigen Vorbereitungen für die Inbetriebnahme der Anschaltbaugruppe.

Die Anschaltbaugruppen können in 100% IBM-kompatiblen PCs (AT, 80386, 80486 etc.) eingesetzt werden. Die notwendige Größe des Arbeitsspeichers richtet sich ausschließlich nach dem Umfang des Anwenderprogramms.

Die Einstellungen der DIP-Schalter werden nur beim Booten der Anschaltbaugruppe eingelesen. Nach einer Änderung der Einstellung muß ein Reset der Anschaltbaugruppe erfolgen, um die Änderung wirksam werden zu lassen.

Bei der Bestückung der Anschaltbaugruppen können u.U. DIP-Schalter unterschiedlicher Ausführung zum Einsatz kommen. Bitte beachten Sie deshalb die direkt an den DIP-Schaltern aufgedruckte Beschriftung für ON/OFF.

Die für die jeweilig beschriebene Funktion nicht relevanten Schalter sind grau dargestellt.

4.1 Einstellen der Adressierung

4.1.1 Basisadresse im IO-Bereich des PCs (IO-Address)

Die Anschaltbaugruppen werden über je 8 Bytes im IO-Bereich des PCs parametriert. Die Basisadresse ist die Adresse des ersten dieser 8 Bytes. Sie wird über den in Bild 4-1 dargestellten DIP-Schalter eingestellt. Die Standardeinstellung (default) ist 120_{hex} und braucht normalerweise nicht verändert zu werden. Bitte stellen Sie sicher, daß der eingestellte Adreßbereich nicht bereits von anderen Komponenten Ihres PCs genutzt wird.

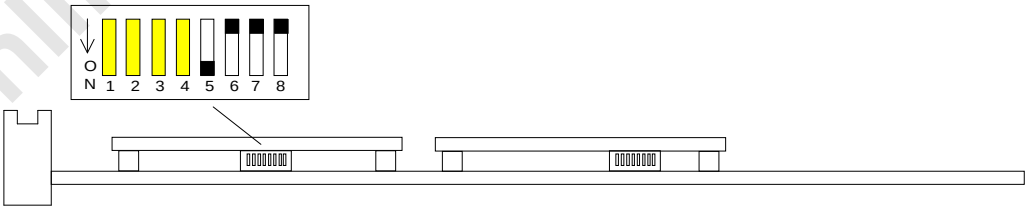

Bild 4-1: Schalter zur Einstellung der IO-Adresse

Für die PC-IO-Adresse stehen 16 Adressen im IO-Adreßraum des PCs zur Verfügung, aus denen Sie über DIP-Schalter die gewünschte Adresse auswählen können. In der nachfolgenden Tabelle 4-1 sind die zulässigen PC-IO-Adressen mit den zugehörigen Einstellungen beschrieben.

Die Schalter Nr.3 und Nr.4 stehen im Auslieferungszustand in Stellung *OFF*. Nehmen Sie keine Veränderungen dieser Einstellungen vor!

online-components.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 4-1: Mögliche Einstellungen für die Basisadresse im IO-Bereich

PC-IO-Adresse	Schalter Nr. 5	Schalter Nr. 6	Schalter Nr. 7	Schalter Nr. 8
100_{hex}	OFF	OFF	OFF	OFF
120_{hex} (default)	**ON**	OFF	OFF	OFF
140_{hex}	OFF	**ON**	OFF	OFF
160_{hex}	**ON**	**ON**	OFF	OFF
180_{hex}	OFF	OFF	**ON**	OFF
$1A0_{hex}$	**ON**	OFF	**ON**	OFF
200_{hex}	OFF	**ON**	**ON**	OFF
220_{hex}	**ON**	**ON**	**ON**	OFF
240_{hex}	OFF	OFF	OFF	**ON**
280_{hex}	**ON**	OFF	OFF	**ON**
$2A0_{hex}$	OFF	**ON**	OFF	**ON**
300_{hex}	**ON**	**ON**	OFF	**ON**
320_{hex}	OFF	OFF	**ON**	**ON**
340_{hex}	**ON**	OFF	**ON**	**ON**
380_{hex}	OFF	**ON**	**ON**	**ON**
$3A0_{hex}$	**ON**	**ON**	**ON**	**ON**

Es können bis zu vier Anschaltbaugruppen in einem Host (PC) verwendet werden. Dabei muß bei allen vier Anschaltbaugruppen die gleiche Basisadresse eingestellt sein. Die Unterscheidung erfolgt über die Boardnummer.

4.1.2 Boardnummer (Board No.)

Wenn mehrere Anschaltbaugruppen in einem Host (PC) betrieben werden, erfolgt die Unterscheidung an Hand einer auf dem Grundboard per DIP-Schalter eingestellten Boardnummer (1 bis 4). Auf den Anschaltbaugruppen muß in diesem Fall die gleiche Basisadresse eingestellt sein. Die Boardnummer bewirkt dann einen automatischen Offset von 0_{hex}, 8_{hex}, 10_{hex} oder 18_{hex} zur Basisadresse im IO-Bereich des PCs.

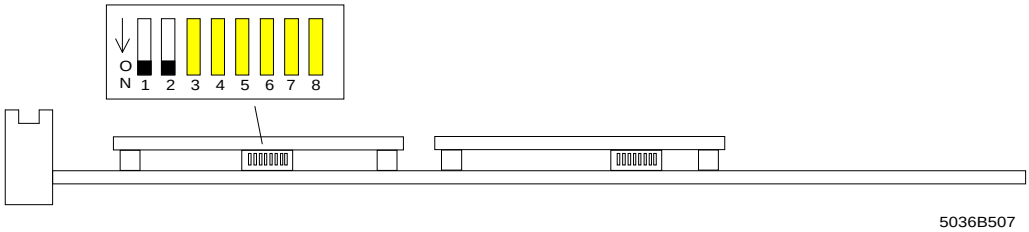

Bild 4-2: Schalter zur Einstellung der Boardnummer (Board No.)

Wenn Sie vier Anschaltbaugruppen einsetzten möchten, müssen ab der eingestellten Basisadresse 32 Bytes (4 mal 8 Bytes) im IO-Bereich des PCs frei sein.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Die Tabelle 4-2 zeigt die Boardnummer, den sich damit ergebenden (automatischen) Offset zur gemeinsamen Basisadresse sowie die zugehörige Einstellung der DIP-Schalter.

Tabelle 4-2: Einstellung der Boardnummer von Anschaltbaugruppen

Schalter Nr. 1	Schalter Nr. 2	Boardnummer	entspricht Offset zur Basisadresse
ON	ON	1	00_{hex}
OFF	ON	2	08_{hex}
ON	OFF	3	10_{hex}
OFF	OFF	4	18_{hex}

Stellen Sie bei Verwendung von nur einer Anschaltbaugruppe sicher, daß die Boardnummer *1* eingestellt ist (Auslieferungszustand).

Falls Sie z.B. mit den DIP-Schaltern Nr.5 bis Nr.8 einer Basisadresse im IO-Bereich von $3A0_{hex}$ einstellen, ergibt sich für die Anschaltbaugruppe mit der Boardnummer 4 ein I/O Adreßbereich von $3B8_{hex}$ bis von $3C0_{hex}$:

Basisadresse **+** Offset **+** I/O-Bereich für Anschaltbaugruppe Nr.4

$3A0_{hex}$ **+** 18_{hex} **+** 8_{hex} **=** $3C0_{hex}$

Stellen Sie sicher, daß der von den Anschaltbaugruppen beanspruchte Adreßraum nicht bereits von anderen Komponenten Ihres PCs genutzt wird.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.2 Einstellen der Boot-Konfiguration

Auf dem Grundboard können mit einem DIP-Schalter bestimmte Boot-Konfigurationen ausgewählt werden. Die Einstellungen des DIP-Schalters werden nur beim Booten der Anschaltbaugruppe eingelesen. Nach einer Änderung der Einstellung muß ein Reset der Anschaltbaugruppe erfolgen, um die Änderung wirksam werden zu lassen.

4.2.1 Kontrolle des InterBus-S (IBS-Control)

(nur IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Bei Verwendung der IBS-Contollerboards mit Coprozessorboard kann die Kontrolle des IBS-Masterboards entweder vom Host (PC) oder vom Coprozessorboard (COP) aus erfolgen. Nur von dieser Stelle aus ist das Absetzen von Kommandos (z.B. Starten des Busses) und ein Verändern der IBS-Ausgangsdaten möglich. Bestimmen Sie mit dem DIP-Schalter Nr.1, ob die Steuerung des IBS-Masterboards vom Host oder vom Coprozessorboard (COP) erfolgt. Die IBS-Eingangsdaten können in jedem Fall sowohl vom Host wie auch vom COP gelesen werden.

Stellen Sie den Schalter Nr.1 bei Verwendung der IBS PC CB/I-T (ohne Coprozessorboard) immer auf Stellung *ON*.

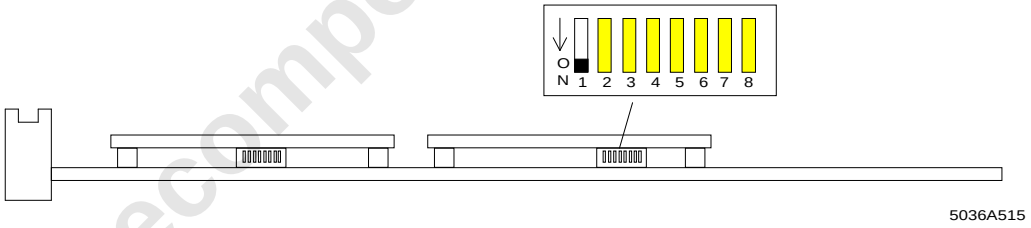

Bild 4-3: Schalter für die Kontrolle des IBS-Masterboards (IBS-Control)

Tabelle 4-3: Kontrolle des IBS-Masterboards vom Host oder COP

Schalter Nr. 1	Kontrolle des IBS-Masterboards vom
ON (Auslieferungszustand)	Host (PC)
OFF	Coprozessorboard

Die Anschaltbaugruppe liest den Schalter Nr.1 nur nach einem Hardware-Reset aus. Führen Sie nach einer Änderung der Schalterstellung einen Hardware-Reset durch um, die Änderung wirksam werden zu lassen.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.2.2 Anlaufverhalten des InterBus-S (IBS-Autostart)

Der DIP-Schalter legt das Anlaufverhalten des InterBus-S nach dem Booten des IBS-Masterboards fest. Steht der Schalter auf OFF, werden die IBS-Datenzyklen automatisch gestartet, falls eine lauffähige Buskonfiguration angeschlossen ist. Das Controllerboard überträgt ab diesem Zeitpunkt Daten vom MPM zur Peripherie und umgekehrt.

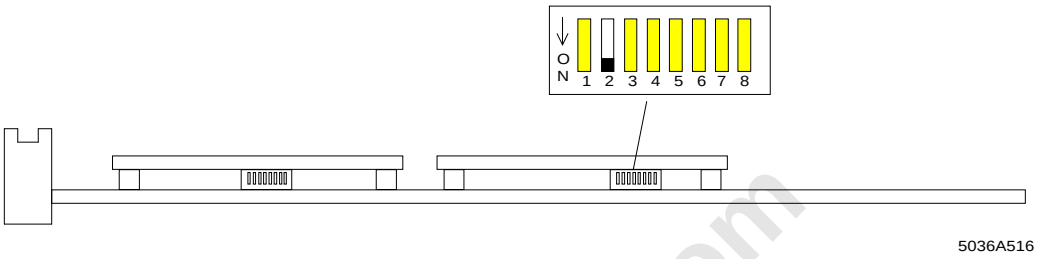

Bild 4-4: Schalter für das Anlaufverhalten (IBS-Autostart)

Tabelle 4-4: Festlegung des Anlaufverhaltens

Schalter Nr. 2	Anlaufverhalten
ON (Auslieferungszustand)	kein automatischer Start
OFF	automatischer Start

Wenn nach dem automatischen Starten der IBS-Datenzyklen ein Fehler im InterBus-S auftritt, der zum Stop des Busses führt, wartet das Anschaltbaugruppe auf einen manuellen Hardware-Reset durch den Reset-Taster auf der Frontblende. Ein erneutes Starten der IBS-Datenzyklen durch das Anwenderprogramm ist in diesem Fall nicht möglich. Berücksichtigen Sie dies bei Verwendung des automatischen Starts der IBS-Datenzyklen.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.2.3 Automatischer Programmstart aus EPROM (EPROM-Start)

(nur IBS PC CB/COP/I-T mit TDOS)

Mit dem Schalter 6 legen Sie unter TDOS fest, ob im EPROM abgelegte Programme direkt nach dem Booten des COPs automatisch gestartet werden sollen oder nicht. Das Abschalten des automatischen Starts ist z.B. sinnvoll, wenn die im EPROM abgelegten Programme fehlerhaft sind und deshalb das Coprozessorboard sofort nach dem Booten abstürzt.

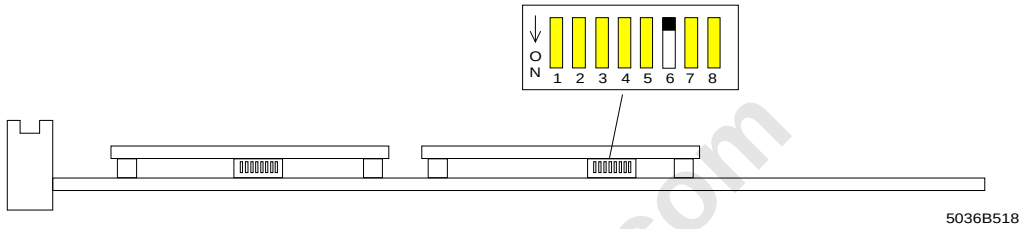

Bild 4-5: Schalter für den Programmstart aus dem EPROM (EPROM-Start)

Tabelle 4-5: Programmstart aus Flash-EPROM

Schalter Nr. 6	Programme aus EPROM
OFF (Auslieferungszustand)	werden nicht automatisch gestartet
ON	werden automatisch gestartet

Die Schalter Nr. 3, Nr. 4, Nr. 5 und Nr. 7 des IBS PC CB/COP/I-T stehen im Auslieferungszustand in Stellung *OFF*. Nehmen Sie keine Veränderungen dieser Einstellungen vor!

4.2.4 Einstellen des Boot-Laufwerks (Boot Disk)

(nur IBS PC CB/RTX486/I-T mit *RTXDOS*)

Mit dem Schalter 6 legen Sie unter *RTXDOS* das Boot-Laufwerk für das Coprozessorboard fest (die Laufwerke des Coprozessorboards sind im Kapitel 2 und der Dokumentation zu *RTXDOS* beschrieben).

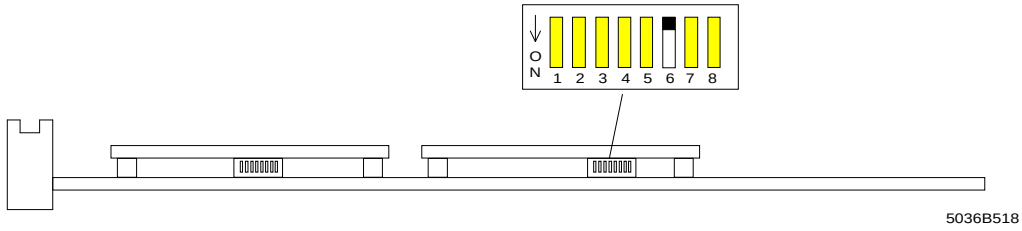

Bild 4-6: Schalter für den die Auswahl des Boot-Laufwerks (Boot Disk)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 4-6: Auswahl des Boot-Laufwerks für das Coprozessorboard

Schalter Nr. 6	Bootlaufwerk
OFF (Auslieferungszustand)	A:\ (Flash-EPROM1)
ON	D:\ (statisches RAM)

Automatisches Starten von Programmen unter *RTXDOS*

Im Auslieferungszustand des IBS PC CB/RTX486/I-T befindet sich im Stammverzeichnis von auf Laufwerk A:\ (Flash-EPROM) eine *AUTOEXEC.BAT* und eine *CONFIG.SYS* mit den grundsätzlichen Einträgen. Um Programme nach dem Booten des Coprozessorboards automatisch zu starten, gehen Sie folgendermaßen vor:

1. Starten Sie auf dem Host das Programm *DPCON.EXE* (befindet sich auf der Tool-Diskette).
2. Kopieren Sie die Dateien *AUTOEXEC.BAT* und *CONFIG.SYS* von Laufwerk A:\ (Flash-EPROM) nach Laufwerk D:\ (SRAM). Die Laufwerke des Coprozessorboards sind im Kapitel 2 und der Dokumentation zu *RTXDOS* beschrieben.
3. Tragen Sie mit einem Editor die automatisch zu startenden Programme in die *AUTOEXEC.BAT* auf Laufwerk D:\ ein.
4. Stellen Sie mit dem Schalter Nr. 6 das Laufwerk D:\ als Boot-Laufwerk ein.
5. Booten Sie das Coprozessorboard. Die in der *AUTOEXEC.BAT* auf Laufwerk D:\ eingetragenen Programme werden, falls sie fehlerfrei sind, nach dem Booten des Coprozessorboards automatisch gestartet.

Das Umschalten des Boot-Laufwerks ist auch sinnvoll, wenn die von der durch das Coprozessorboard abzuarbeitenden *AUTOEXEC.BAT* aufgerufenen Programme fehlerhaft sind und deshalb das Coprozessorboard sofort nach dem Booten abstürzt. Nach Umschalten auf das andere Boot-Laufwerk kann das Coprozessorboard dann ohne Aufruf der fehlerhaften Programme booten.

Automatisch zu startende Programme sollten nie gleichzeitig auf Laufwerk A:\ und Laufwerk D:\ in der *AUTOEXEC.BAT* eingetragen sein, da Sie dann den Aufruf von fehlerhaften Programmen durch Schalter Nr.6 nicht verhindern können.

Sie können die automatisch zu startenden Programme an folgenden Stellen hinterlegen:

- im Flash-EEPROM 1, das vom COP als Laufwerk A:\ anzusprechen ist.
- im Flash-EEPROM 2, das vom COP als Laufwerk C:\ anzusprechen ist.
- im batteriegepufferten SRAM, das vom COP als Laufwerk D:\ anzusprechen ist.
- Auf der Festplatte (C:\ ) des Hosts, die vom COP als Laufwerk E:\ anzusprechen ist.

Die Schalter Nr. 3 und Nr. 4 des IBS PC CB/RTX486/I-T stehen im Auslieferungszustand in Stellung *OFF*. Nehmen Sie keine Veränderungen dieser Einstellungen vor!

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.2.5 Bootverhalten in Bezug auf *RFSERVER* (Wait for RFSERVER)

(nur IBS PC CB/RTX486/I-T mit *RTXDOS*)

Das TSR-Programm *RFSERVER.EXE* ermöglicht dem Coprozessorboard, auf die Laufwerke des Hosts (PC) zuzugreifen. Nach dem Start von *RFSERVER* auf dem Host kann das Coprozessorboard z.B. Programme von der Festplatte des Hosts laden und dort Daten lesen oder speichern. Die Verbindung zwischen *RFSERVER* und *RTXDOS* geschieht über das MPM.
Das Programm *RFSERVER.EXE* befindet sich auf der Tool-Diskette.

RTXDOS versucht, während das Coprozessorboard bootet, Verbindung zum Programm *RFSERVER.EXE* auf dem Host aufzunehmen. Da das Coprozessorboard sehr schnell bootet, ist bei einem Neustart des gesamten Systems wahrscheinlich *RFSERVER.EXE* auf dem Host noch nicht gestartet. Mit dem Schalter Nr. 5 können Sie nun festlegen, ob das Coprozessorboard in diesem Fall den Boot-Vorgang fortsetzt, oder wartet, bis *RFSERVER.EXE* auf dem Host gestartet wurde. Es ist z.B. im Stand-Alone-Betrieb des *IBS PC CB/RTX486/I-T* sinnvoll, nicht auf den Start von *RFSERVER.EXE* zu warten .

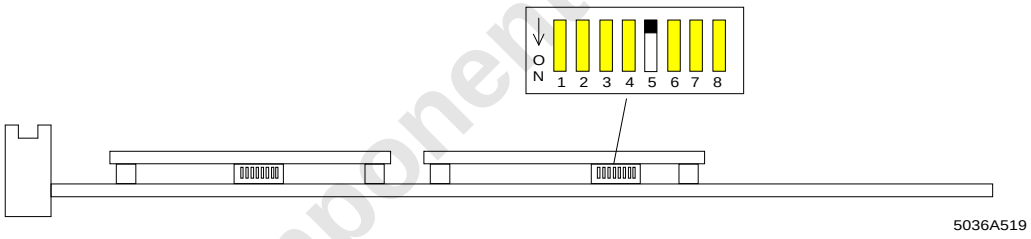

Bild 4-7: Schalter zur Einstellung des Bootverhaltens in Bezug auf *RFSERVER*

Tabelle 4-7: Bootverhalten des Coprozessorboards in Bezug auf *RFSERVER*

Schalter Nr. 5	Bootverhalten
OFF (Auslieferungszustand)	Coprozessorboard wartet beim Booten auf den Start von *RFSERVER*
ON	Coprozessorboard wartet beim Booten nicht auf den Start von *RFSERVER*

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.2.6 Boot-Verhalten in Bezug auf *DPCON* (Wait for DPCON)

(nur IBS PC CB/RTX486/I-T mit *RTXDOS*)

Das Programm *DPCON.EXE* ermöglicht es, Tastatur und Bildschirm des Hosts (PC) als „Terminal“ für das Coprozessorboard zu verwenden. Nach dem Start von *DPCON.EXE* auf dem Host steht dem Anwender dort für das Coprozessorboard eine DOS-Oberfläche zur Verfügung. Die Verbindung zwischen *DPCON.EXE* und *RTXDOS* geschieht über das MPM. Das Programm *DPCON.EXE* befindet sich auf der Tool-Diskette.

Während das Coprozessorboard bootet, schreibt es die für einen PC üblichen Bildschirmausgaben in das MPM. Diese können dort vom Programm *DPCON.EXE* abgeholt und auf dem Monitor des Hosts dargestellt werden.

Falls *DPCON.EXE* die Bildschirmausgaben nicht aus dem MPM abholt, kann der hierfür reservierte MPM-Bereich voll werden. Mit dem Schalter Nr. 7 können Sie nun festlegen, ob das Coprozessorboard in diesem Fall den Boot-Vorgang fortsetzt oder wartet, bis der MPM-Bereich wieder frei ist.

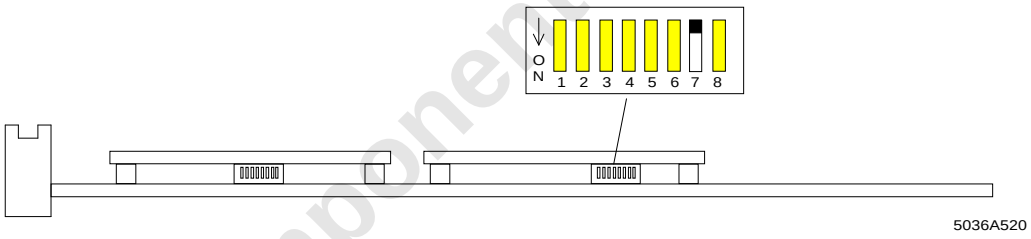

Bild 4-8: Schalter zur Einstellung des Boot-Verhaltens in Bezug auf DPCON

Tabelle 4-8: Bootverhalten in Bezug auf DPCON

Schalter Nr. 7	Bootverhalten
OFF (Auslieferungszustand)	Das Coprozessorboard führt, wenn der MPM-Bereich „voll“ ist, den Boot-Vorgang fort, obwohl DPCON die Zeichen aus dem MPM noch nicht gelesen hat. Es können Zeichen verlorengehen.
ON	Das Coprozessorboard wartet, wenn der MPM-Bereich „voll“ ist, mit dem Fortsetzen des Boot-Vorgangs, bis DPCON die Zeichen aus dem MPM gelesen hat.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.2.7 Datenübertragung zwischen COP und Entwicklungsumgebung

(Terminal-Mode, nur IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Der Datenübertragungsweg (Terminal-Mode) zwischen dem Coprozessorboard und

- der Entwicklungsumgebung *TDOS-PRO (IBS PC COP SWT)* oder
- dem Terminalprogramm *DPCON* (befindet sich auf der Tool-Diskette)

kann wahlweise über das MPM oder über die seriellen Schnittstellen des Coprozessorboards und des PCs erfolgen. Dies wird über DIP-Schalter Nr. 8 eingestellt.

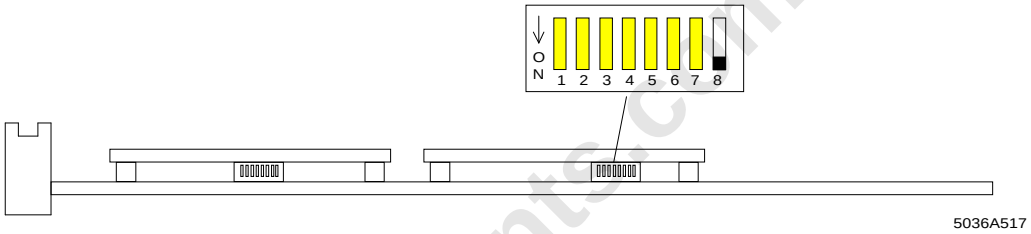

Bild 4-9: Schalter zur Einstellung des Datenübertragungswegs (Terminal-Mode)

Tabelle 4-9: Einstellung des Terminal-Modes

Schalter Nr. 8	Datenübertragungsweg
ON (Auslieferungszustand)	über das MPM
OFF	über die seriellen Schnittstellen (RS232)

Die Entwicklungsumgebung *TDOS-PRO (IBS PC COP SWT)* ist nicht kompatibel zum COP486 des *IBS PC CB/RTX486/I-T* und wird dafür auch nicht benötigt.

Wenn die Verbindung über die seriellen Schnittstellen erfolgen soll, müssen diese über das Adapterkabel *IBS PC COP RS 232 CAB* und das Entwicklungskabel miteinander verbunden werden (siehe Kapitel 3). Beide Kabel gehören zum Lieferumfang der Entwicklungsumgebung *TDOS-PRO*.

Stellen Sie den Schalter Nr. 8 auf ON, falls Sie die serielle Schnittstelle des Coprozessorboards für Funktionen Ihres Anwenderprogramms nutzen möchten.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.3 Einstellung der Steckbrücken (Jumper)

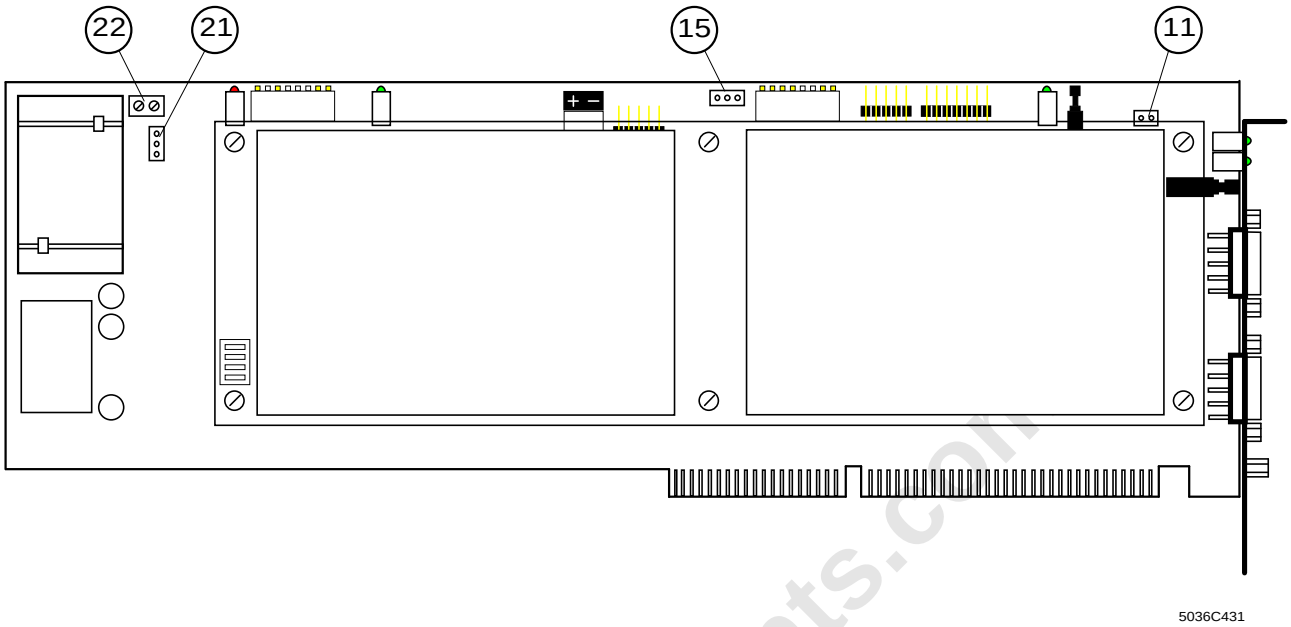

Bild 4-10: Steckbrücken zur Parametrierung der Anschaltbaugruppen

22 Klemme zur Einspeisung der externen Versorgungsspannung (nicht IBS PC CB I-T)

Die Anschaltbaugruppe wird über folgende Steckbrücken parametriert:
21 Steckbrücke für die Auswahl der Spannungsversorgung (nicht IBS PC CB I-T)
15 Steckbrücke zur Trennung vom Hardware-Reset des PCs (nicht IBS PC CB I-T)
11 Steckbrücke zum Deaktivieren der Reset-Taster

4.3.1 Auswahl der Spannungsversorgung (Power Supply)

(nur IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Über die Steckbrücke *Power Supply* wählen Sie aus, ob die Anschaltbaugruppen IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T ihre Versorgungsspannung über den AT-Bus des Hosts (PC) oder über die Klemmen zur externen Versorgung beziehen sollen.

Tabelle 4-10: Funktion der Steckbrücke zur Auswahl der Spannungsversorgung (siehe Bedruckung auf der Folie der Anschaltbaugruppe)

Steckbrücke *Power Supply*	**Spannungsversorgung**
auf PC gesteckt (Auslieferungszustand)	durch den Host (PC) über den AT-Bus
auf Ext. gesteckt	von einem externen Netzteil über die grünen CombiCon-Klemmen neben dem Batterie-Pack (siehe Bild 4-10, Position 22)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.3.2 Trennung vom Hardware-Reset des Hosts (PC HW RESET)

(nur IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Über die Steckbrücke *Enable/Disable PC HW RESET* (Bild 4-10, Position 15) können Sie die Anschaltbaugruppen IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T so konfigurieren, daß sie von einem Hardware-Reset des Hosts (PC) nicht beeinflußt werden.

Tabelle 4-11: Funktion der Steckbrücke zur Trennung vom Hardware-Reset des Hosts (siehe Bedruckung auf der Folie der Anschaltbaugruppe)

Steckbrücke Enable/ Disable PC HW RESET	**Resetverhalten der Anschaltbaugruppes**
auf ON gesteckt (enabled, Auslieferungszustand)	Ein Reset des Hosts veranlaßt die Anschaltbaugruppe zum Booten.
auf OFF gesteckt (disabled)	Ein Reset des Hosts beeinflußt die Anschaltbaugruppe nicht.

4.3.3 Deaktivieren der Reset-Taster (Enable/Disable RESET-Button)

Um eine unbeabsichtigte Betätigung der Reset-Taster zu vermeiden, besteht die Möglichkeit, die Taster durch Ziehen der Steckbrücke *Enable/Disable RESET Button* (Bild 4-10, Position 11) außer Funktion zu setzen.

Tabelle 4-12: Funktion der Steckbrücke zum Deaktivieren der Reset-Taster (siehe Bedruckung auf der Folie der Anschaltbaugruppe)

Steckbrücke *Enable/Disable RESET-Button*	**Reset-Taster**
Enable (Brücke gesteckt, Auslieferungszustand)	in Funktion
Disable (Brücke offen)	gesperrt

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.4 Anschluß des Batterie-Packs

(nur IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Das Batterie-Pack zur Versorgung des SRAMs und der Echtzeituhr auf dem Coprozessorboard ist im Auslieferungszustand nicht mit dem Coprozessorboard verbunden. Stecken Sie vor dem Einbau der Anschaltbaugruppe in Ihren PC den Stecker des Batterie-Packs auf die mit *SRAM Battery 6V* gekennzeichnete Stiftleiste am Rand des Coprozessorboards.

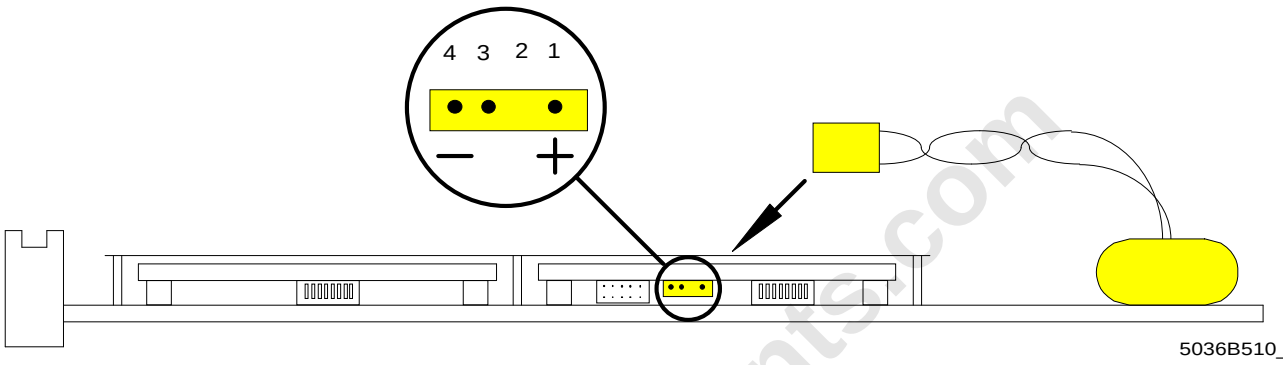

Bild 4-11: Position der Stiftleiste für das Batterie-Pack

Tabelle 3-13: Anschlußbelegung der Stiftleiste für das Batterie-Pack

Pin	Signal	Farbe des Anschlußkabels
1	+6V Batterie	rot
2	Code-Pin	—
3	GND	—
4	GND	schwarz

Das Batterie-Pack hat im Betrieb eine Lebensdauer von ca. 2,5 Jahren

4.5 Einbau der Anschaltbaugruppe in den PC

Um Beschädigungen der PC-Karte und des Hosts (PC) zu vermeiden, darf der Ein- bzw. Ausbau der Anschaltbaugruppe nur bei spannungsfreiem PC erfolgen.

Achten Sie darauf, daß der Abstand zwischen Anschaltbaugruppe und anderen PC-Karten ausreichend ist. Vermeiden Sie unbedingt den Kontakt zwischen PC-Karten. Bauen Sie Anschaltbaugruppen nur in AT-Bus-Steckplätze ein, an denen ausreichend Platz zur Verfügung steht (z.B. nicht über hochbauenden Speicherbausteinen, SIMMs etc.).

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Die Anschaltbaugruppen benötigen einen langen AT-Bus-Steckplatz im Host (PC). Die Stromversorgung des PCs muß, insbesondere zum Betrieb von mehreren Anschaltbaugruppen in einem PC, ausreichend bemessen sein.

Tabelle 4-14: Stromaufname der Anschaltbaugruppen

Anschaltbaugruppe	Versorgung
IBS PC CB/I-T	1,2 A (5V)
IBS PC CB/COP/I-T	1,5 A (5V)
IBS PC CB/RTX486/I-T	1,5 A (5V)

Erden Sie die Anschaltbaugruppe über den Erdungsbolzen auf dem PC-Kartenhalter.

4.5.1 Serielle Schnittstelle des Coprozessorboards

(nur IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Das Coprozessorboard besitzt eine serielle Schnittstelle (RS 232-Pegel), die der COM 1 eines Standard-PCs entspricht. Diese voll PC-kompatible Schnittstelle ist am Rand der Coprozessorboards als 10-polige Stiftleiste realisiert und kann mit dem Adapterkabel *IBS PC COP RS 232 CAB* auf einen Sub-D-Stekker geführt werden, den Sie z.B. auf einem zusätzlichen PC-Kartenhalter (freier Steckplatz) montieren. Das Adapterkabel kann unter der Artikel-Nr. 27 51 65 8 bestellt werden. Es gehört außerdem zum Lieferumfang der Entwicklungsumgebung *IBS PC COP SWT (TDOS-PRO)*. Beachten Sie bei der Montage die korrekte Belegung des Adapterkabels (siehe Kapitel 3).

4.6 Installation des Device-Drivers

Bei der Installation des Device-Drivers sind 4 Parameter von Bedeutung.

- Basisadresse im IO-Bereich des PCs (IO-Address)
- Boardnummer (Board No.)
- Basisdresse des 4Kbyte großen MPM-Fensters im Speicherbereich des PCs (MPM-Address)
- Interruptnummer

Die einzelnen Parameter haben dabei folgende Bedeutung:

Basisadresse im IO-Bereich des PCs (IO-Address)

Dieser Parameter steht für die Adresse, unter der die Anschaltbaugruppe im IO-Adreßbereich des PCs angesprochen wird. Die PC-IO-Adresse wird auf der Anschaltbaugruppe mit DIP-Schaltern eingestellt (siehe Kapitel 4.1.1). Der Standardwert für die PC-IO-Adresse ist 120_{hex}. Hinweise zu alternativen IO-Adressen können Sie dem Kapitel 4.1.1 entnehmen.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Boardnummer (Board No.)

Die Boardnummer gibt an, für welche Anschaltbaugruppe der Device-Driver geladen werden soll. Der Standardwert ist 1, d.h. falls der Treiber für die Anschaltbaugruppe Nr.1 (über DIP-Schalter einstellbar, siehe Kapitel 4.1.2) geladen werden soll, brauchen Sie diesen Parameter nicht anzugeben. Zulässige Werte für die Boardnummer sind 1, 2, 3 und 4.

Speicheradresse für das 4 KByte große MPM-Fenster (MPM Address)

Die Zugriffe auf das Multi-Port-Memory (MPM) erfolgen vom Host (PC) aus über ein 4 KByte großes MPM-Fenster im Speicherbereich unterhalb 1MB. Der Parameter *MPM-Address* steht für die Basisadresse dieses MPM-Fensters.

Die Karte belegt ab dieser Basisadresse einen Adreßraum von 4 KBytes. Stellen Sie sicher, daß dieser Bereich nicht bereits von anderen Karten belegt wird. Eine automatische Überprüfung findet **nicht** statt.

Tabelle 4-15: Typisches Memory Mapping eines Standard-PCs

Speicher	Adreßbereich			belegt von
64 KByte	$F0000_{hex}$	bis	$FFFFF_{hex}$	ROM-BIOS
64 KByte	$E0000_{hex}$	bis	$EFFFF_{hex}$	Erweiterungen
64 KByte	$D0000_{hex}$	bis	$DFFFF_{hex}$	EMS-Fenster
32 KByte	$C8000_{hex}$	bis	$CFFFF_{hex}$	SCSI-Controller oder LAN-BIOS
32 KByte	$C0000_{hex}$	bis	$C7FFF_{hex}$	EGA/VGA-BIOS
32 KByte	$B8000_{hex}$	bis	$BFFFF_{hex}$	Video-Speicher (z.B. VGA-Text oder EGA)
32 KByte	$B0000_{hex}$	bis	$B7FFF_{hex}$	Video-Speicher (z.B. Monochrom oder EGA)
64 KByte	$A0000_{hex}$	bis	$AFFFF_{hex}$	Video-Speicher (z.B. VGA-Grafik oder EGA)
640 KByte	00000_{hex}	bis	$9FFFF_{hex}$	konventioneller Speicher

Die Basisadresse des MPM-Fensters kann in den Bereich zwischen 80000_{hex} bis $FF000_{hex}$ gelegt werden. Da dieser Speicherbereich in der Praxis schon recht weitgehend belegt ist (BIOS usw.), beschränkt sich der zur Verfügung stehende Adreßbereich in der Regel auf Teile der Adreß-Segmente D und E (Adressen von $D0000_{hex}$ bis $EFFFF_{hex}$). Das Adreßfenster sollte möglichst an den Anfang oder das Ende des verfügbaren Adreßbereiches gelegt werden, um diesen nicht unnötig zu zerstückeln. Der Standardwert (default) ist $D0000_{hex}$.

Interruptnummer

Jede Anschaltbaugruppe benutzt zur Ansprache ihres Device-Drivers einen Hardware-Interrupt, der beim Start des jeweiligen Device-Drivers vorgegeben wird.

Die Anschaltbaugruppen unterstützen die Interrupts IRQ 3, 5, 7, 9, 10, 11, 12 und 15.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Bei Verwendung mehrerer Anschaltbaugruppen in einem Host (PC) muß für jede installierte Anschaltbaugruppe ein anderer Interrupt verwendet werden. Auf einem Standard-PC sind die Interrupts IRQ 10, 11, 12 und 15 meist nicht benutzt und bieten sich somit für den Device-Driver an. Die anderen Interrupts werden häufig von Standard-PC-Komponenten (serielle Schnittstellen COM1 und COM2, Netzwerkkarten usw.) verwendet und sollten deshalb nicht von Anschaltbaugruppen benutzt werden.

Stellen Sie sicher, daß der verwendete Interrupt nicht bereits von anderen Komponenten Ihres PCs belegt ist.

Falls Sie die Angabe eines Parameters unterlassen, wird für diesen Parameter automatisch der Standardwert (Default) verwendet.

4.6.1 Device-Driver unter MS-DOS

Der Device-Driver für die Anschaltbaugruppe ist unter DOS als TSR-Programm (englisch: **T**erminate and **S**tay **R**esident) realisiert, also als resident im Hintergrund laufendes Programm.

- Das TSR-Programm für den Host (PC) hat den Namen *IBSPCCB.EXE*.
- Das TSR-Programm für das Coprozessorboard (COP) hat den Namen *IBSCOP.EXE*.

Device-Driver auf dem Host (PC)

Starten Sie den Device-Driver auf dem Host (PC) durch Aufruf von *IBSPCCB*. Wechseln Sie zum Aufruf des Device-Drivers von der MS-DOS-Kommandozeile aus in das Verzeichnis, in dem sich der Device-Driver befindet oder geben Sie beim Aufruf den kompletten Pfad mit an.

Device-Driver auf dem Coprozessorboard

- Übertragen Sie den Device-Driver *IBSCOP.EXE* mit der Entwicklungsumgebung *TDOS-PRO* auf den *COP386* des *IBS PC CB/COP/I-T*. Die genaue Beschreibung hierzu finden Sie im Handbuch zur Entwicklungsumgebung.
- Der Device-Driver *IBSCOP.EXE* ist im Auslieferungszustand des *IBS PC CB/RTX486/I-T* bereits auf Laufwerk A:\ (Flash-EPROM) des *COP486* abgelegt und wird über einen Eintrag in der *AUTOEXEC.BAT* automatisch gestartet.

Der Device-Driver kann (z.B. unter MS-DOS 5.0) in den Speicherbereich oberhalb von 640 KByte geladen werden. Er belegt dann unterhalb von 640 KByte keinen Speicherplatz. Hinweise zur Installation von Treibern im oberen Speicherbereich finden Sie in Ihrem DOS-Handbuch.

Sie können dem Device-Driver beim Aufruf verschiedene Parameter übergeben. Durch den Aufruf von *IBSPCCB /?* oder *IBSPCCB /HELP* wird eine Liste der möglichen Parameter angezeigt.

onlinecomponents.com

Tabelle 4-16: Beipiele für die Parameterübergabe beim Aufruf des Device-Drivers

Aufruf	Wirkung
IBSPCCB IO=100	stellt die Basisadresse im IO-Bereich des PCs auf 100_{hex}
IBSPCCB BN=2	stellt die Boardnummer auf *2*
IBSPCCB MPM=C800	stellt die Basisadresse des MPM-Fensters auf $C8000_{hex}$
IBSPCCB IRQ=10	stellt die Interruptnummer auf *10*
IBSPCCB IO=100 BN=2	stellt die Basisadresse im IO-Bereich des PCs auf 100_{hex} und die Boardnummer auf *2*

Durch den Aufruf mit *IBSPCCB /INFO* können Sie die eingestellten Parameter aller Device-Driver überprüfen.

Wenn Sie *IBSPCCB* aufrufen, ohne weitere Parameter anzugeben, werden folgende Voreinstellungen (Defaults) benutzt:

- Basisadresse im IO-Bereich des PCs (IO-Address): 120_{hex}
- Boardnummer (Board No.): 1
- Basisdresse des 4Kbyte großen MPM-Fensters im Speicherbereich des PCs (MPM-Address): $D0000_{hex}$
- Interruptnummer bei Treibersoftware Version ≤ 0.9: IRQ3
- Interruptnummer bei Treibersoftware Version ≥ 0.91: IRQ15

Sie müssen beim Aufruf des Device-Drivers nur die Werte angeben, die Sie ändern möchten. Dabei ist die Reihenfolge der Parameterangaben ohne Bedeutung. Durch Eingabe von *IBSPCCB /UNINSTALL* werden alle Device-Driver deinstalliert.

4.6.1.1 Installationshilfe für DOS

Der Device-Driver *IBSPCCB.EXE* prüft bei seiner Installation, ob unter der angegeben IO-Adresse ein Anschaltbaugruppe zu finden ist. Kann unter der IO-Adresse keine Anschaltbaugruppe gefunden werden, wird eine Fehlermeldung ausgegeben und der Device-Driver nicht installiert.

Der Versuch einer Mehrfachinstallation von Device-Drivern mit identischen Parametern wird ebenfalls mit einer Fehlermeldung abgebrochen. Nur der zuerst installierte Device-Driver bleibt in Funktion.

Mit Hilfe des Device-Drivers können Sie auch auf einfache Weise überprüfen, ob die Anschaltbaugruppe richtig im PC installiert ist.

Eine Doppelbelegung von MPM-Adressen, IO-Adressen oder Interrupts wird vom Device-Driver nicht erkannt. Stellen Sie sicher, daß der eingestellte IO-Adreßbereich, der Speicherbereich für das 4 KByte große MPM-Fenster und der Interrupt für die Anschaltbaugruppe nicht bereits von anderen PC-Komponenten genutzt werden!

Bei Verwendung des Speichermanagers *EMM386.EXE* (Microsoft) muß im Adreßraum des Hosts (PC) das 4 KByte große MPM-Fensters vor Zugriffen die-

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

ses Speichermanagers geschützt werden. Erweitern Sie hierzu in der *CONFIG.SYS* Ihres Hosts (PC) folgende Zeile um einen entsprechenden Eintrag.

Beispiel:

```
DEVICE = C:\DOS\EMM386.EXE      x=D000-D100
```

Durch den fett dargestellten Eintrag wird in diesem Beispiel ab Adresse $D0000_{hex}$ ein 4 KByte großer Speicherbereich vor Zugriffen des *EMM386.EXE* geschützt. Eine nähere Beschreibung des Speichermanagers *EMM386* finden Sie in Ihrem DOS-Handbuch.

Stellen Sie außerdem sicher, die Verwendung von SHADOW RAM oder VIDEO CACHE nicht mit dem 4 KByte großen MPM-Fenster in Konflikt gerät. Deaktivieren Sie im Konfliktfall im BIOS-Setup Ihres PCs diese Optionen durch folgende Einstellungen:

SHADOW RAM	disabled
VIDEO CACHE	disabled

Der Device-Driver *IBSPCCB.EXE* darf nicht innerhalb eines DOS-Fensters von Microsoft-Windows® installiert oder deinstalliert werden!

4.6.2 Device-Driver unter Microsoft Windows

Die Treibersoftware für Microsoft Windows® ist als **D**ynamic-**L**ink-**L**ibrary (DLL) ausgeführt (IBSPCCB.DLL). Diese DLL beinhaltet das Device-Driver-Interface sowie Device-Driver für vier Anschaltbaugruppen. Kopieren Sie die Datei *IBSPCCB.DLL*, wie unter Windows üblich, in das Verzeichnis, in dem sich Ihr Anwenderprogramm befindet, oder in das Windows-Stammverzeichnis.

Die Verwendung zusätzlicher Treiber (wie z.B. Aufrufen von TSR-Programmen unter DOS oder Eintragen von OS/2-Device-Driver unter IBM OS/2®) ist unter Microsoft Windows® nicht erforderlich!

Im Windows-Stammverzeichnis muß sich außerdem die Datei IBSPCCB.INI befinden, die zur Parametrierung dient. Tragen Sie die zur Initialisierung der Anschaltbaugruppe erforderlichen Parameter (IO-Adresse, MPM-Adresse und Interruptnummer) in die Datei *IBSPCCB.INI* ein. Das folgende Beispiel zeigt die Einträge für den Betrieb einer Anschaltbaugruppe.

onlinecomponents.com

```
[GENERAL]
EnableInitErrorMessage=TRUE

[BOARD1]
BoardInUseFlag=TRUE
IOAddress=120
MPMAddress=D000
IRQ=10

[BOARD2]
BoardInUseFlag=FALSE
IOAddress=120
MPMAddress=D100
IRQ=11

[BOARD3]
BoardInUseFlag=FALSE
IOAddress=120
MPMAddress=D200
IRQ=12

[BOARD4]
BoardInUseFlag=FALSE
IOAddress=120
MPMAddress=D300
IRQ=15
```

Bild 4-12: Beispieleinträge in der Datei *IBSPCCB.INI*

4.6.2.1 Installationshilfe für Microsoft Windows

Setzen Sie z.B. für Board 1 den Eintrag *BoardInUseFlag,* auf *TRUE*, damit die Anschaltbaugruppe Nr. 1 während der Initialisierungsphase erkannt wird. Andernfalls wird, selbst wenn die Anschaltbaugruppe Nr. 1 vorhanden ist, diese als nicht vorhanden gekennzeichnet und nicht initialisiert.

Wenn Sie einen ungültigen Wert angeben, wird die Initialisierung der DLL **nicht** abgebrochen.

- Falls der Eintrag *EnableInitErrorMessage=TRUE* lautet, wird beim Laden der DLL (Start des Anwenderprogramms) eine Fehlermeldung in einer Windows-Message-Box ausgegeben.
- Falls der Eintrag *EnableInitErrorMessage=FALSE* lautet, wird beim Laden der DLL (Start des Anwenderprogramms) **keine** Fehlermeldung ausgegeben. In diesem Fall können Sie nur anhand der Fehlermeldungen der Treiber- bzw. DDI-Funktionen (z.B. beim ersten Öffnen eines Datenkanals) erkennen, daß ein Fehler während der Initialisierung aufgetreten ist.

Die Parameter entsprechen bis auf die Einschränkung, daß die Werte für *MPMAddress* nur im Bereich von $A0000_{hex}$ bis $FF000_{hex}$ liegen dürfen, denen der Treibersoftware für DOS (siehe Seite 4-17).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.6.3 Device-Driver unter IBM OS/2

Der Device-Driver für die Anschaltbaugruppe ist für IBM OS/2® als OS/2-Device-Driver realisiert.

Für jede Anschaltbaugruppe muß ein OS/2-Device-Driver installiert werden!

Die OS/2-Device-Driver für die Anschaltbaugruppen müssen beim Booten des Hosts (PC) geladen werden. Tragen Sie dazu für jede Anschaltbaugruppe einen OS/2-Device-Driver in die Datei *CONFIG.SYS* ihres Hosts ein.

Der OS/2-Device-Driver für die Anschaltbaugruppe hat den Namen *OS2_IBS.DRV*. Geben Sie den kompletten Pfad an, falls sich der Treiber nicht im OS/2-Stammverzeichnis befindet.

Wenn Sie *OS2_IBS.DRV* eintragen, ohne weitere Parameter anzugeben, werden folgende Voreinstellungen (Defaults) benutzt:

- Basisadresse im IO-Bereich des PCs (IO Address): 120_{hex}
- Boardnummer (Board No.): 1
- Basisdresse des 4Kbyte großen MPM-Fensters im Speicherbereich des PCs (MPM-Address): $D0000_{hex}$
- Interruptnummer: IRQ15

Tabelle 4-17: Beipiele für das Eintragen von Parametern

Eintrag	Auswirkung
Device=OS2_IBS.DRV IO=100	stellt die Basisadresse auf 100_{hex}
Device=OS2_IBS.DRV IBSPCCB BN=2	stellt die Boardnummer auf *2*
Device=OS2_IBS.DRV IBSPCCB MPM=C800	stellt die MPM-Adresse auf $C8000_{hex}$
Device=OS2_IBS.DRV IBSPCCB IRQ=10	stellt die Interruptnummer auf *10*
Device=OS2_IBS.DRV IBSPCCB IO=100 BN=2	stellt die Basisadresse auf 100_{hex} und die Boardnummer auf *2*

Die Reihenfolge der Parameterangaben ist ohne Bedeutung.

Falls Sie einen Parameter nicht angeben, wird automatisch der Standardwert verwendet.

4.6.3.1 Installationshilfe für OS/2

Stimmt die angegebene PC-IO-Adresse nicht mit der auf der Anschaltbaugruppe über DIP-Schalter eingestellten Adresse überein, wird die Initialisierung des Boards abgebrochen und eine Fehlermeldung ausgegeben.

Die Anschaltbaugruppe belegt einen Adreßbereich von 4 KBytes. Dieser Bereich darf nicht bereits von anderen PC-Karten (z.B. Netzwerk-Karten) belegt sein. Eine Überprüfung findet nicht statt.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Interruptnummer

OS/2 gestattet es auf dem ISA-Bus nicht , daß ein Interrupt von mehreren Karten verwendet wird. Achten Sie darauf, daß der verwendete Interrupt nicht bereits belegt ist. Andernfalls wird die Initialisierung abgebrochen und eine Fehlermeldung (Error at SetIRQ) ausgegeben. Der Standardwert für den Interrupt ist 15.

4.7 Installation der Peripherie

Detallierte Hinweise zur Installation der Peripherie (IBS-Teilnehmer, Module, Kabel) entnehmen Sie bitte dem IBS-Installationshandbuch *IBS SYS INST UM* (Artikel-Nr. 27 54 28 6). Das Handbuch informiert Sie in verschiedenen Kapiteln zu folgenden Themen:

- Systemübersicht
- Montage der E/A-Komponenten
- Verkabelungsempfehlungen
- Inbetriebnahme und Funktionstest
- Fehlerbeseitigung
- Austausch von IBS-Komponenten
- Kabelpläne

Außerdem erhalten Sie auf Anfrage Datenblätter zu allen neuen IBS-Teilnehmern von Phoenix-Contact.

4.8 Software-Tools für die Inbetriebnahme

Zur ersten Inbetriebnahme stehen Ihnen verschiedene Software-Tools zur Verfügung, die das Ansprechen Ihres Busaufbaus ohne großen Programmieraufwand ermöglichen.

- Das Prozeßdaten-Monitorprogramm *PCCBMONI.EXE* (gehört zum Lieferumfang der Treibersoftware)
- Die Diagnose und Konfigurations-Software *IBS SYS SWT* unter DOS (Artikel-Nr. 27 80 88 1)
- Die Bedien-Software *IBS CMD SWT*, die Projektierung, Konfiguration, Monitoring und Diagnose ihres InterBus-S-Systems unter Microsoft Windows® ermöglicht.

Die Software-Tools ermöglichen das Setzen von Ausgängen. Stellen Sie vor dem Setzen von Ausgängen sicher, daß in direktem oder indirektem Zusammenhang mit gefährlicher Prozeßperipherie (z.B. ungeschützt laufende Wellen, Motoren, Pressen usw.) kein Sach- oder Personenschaden erfolgen kann.

Weitere Hinweise zu den verschiedenen Software-Tools finden Sie im Kapitel 7 und in den jeweiligen Handbüchern.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.9 Inbetriebnahme mit Prozeßdaten-Monitorprogramm

Zum Lieferumfang der Treibersoftware für IBS PC CB/.../I-T gehört ein Prozeß-daten-Monitorprogramm. Das Programm leistet folgende Dienste:

- Erste Inbetriebnahme eines InterBus-S-Systems ohne Programmieraufwand
- Test ihres Busaufbaus
- Kontrolle der angeschlossenen Konfiguration
- Anzeigen des Status von einzelnen Eingängen (binär)
- Anzeigen des Status von Eingangsworten (hexadezimal)
- Setzen von einzelnen Ausgängen (binär)
- Setzen von Ausgangsworten (hexadezimal)
- Rücksetzen des InterBus-S-Systems

Spezialfunktionen wie das Generieren eines Datenfiles mit allen wichtigen Daten über die an die Anschaltbaugruppe angeschlossnen Module ergänzen die Funktionalität dieses Programms.

Nach der Installation ihrer Anschaltbaugruppe und der Treibersoftware übertragen Sie die Datei *PCCBMONI.EXE* von der Treiberdiskette auf ihre Festplatte Starten Sie das Prozeßdaten-Monitorprogramm von dort durch den Aufruf *PCCBMONI*. Es erscheint folgendes Bild:

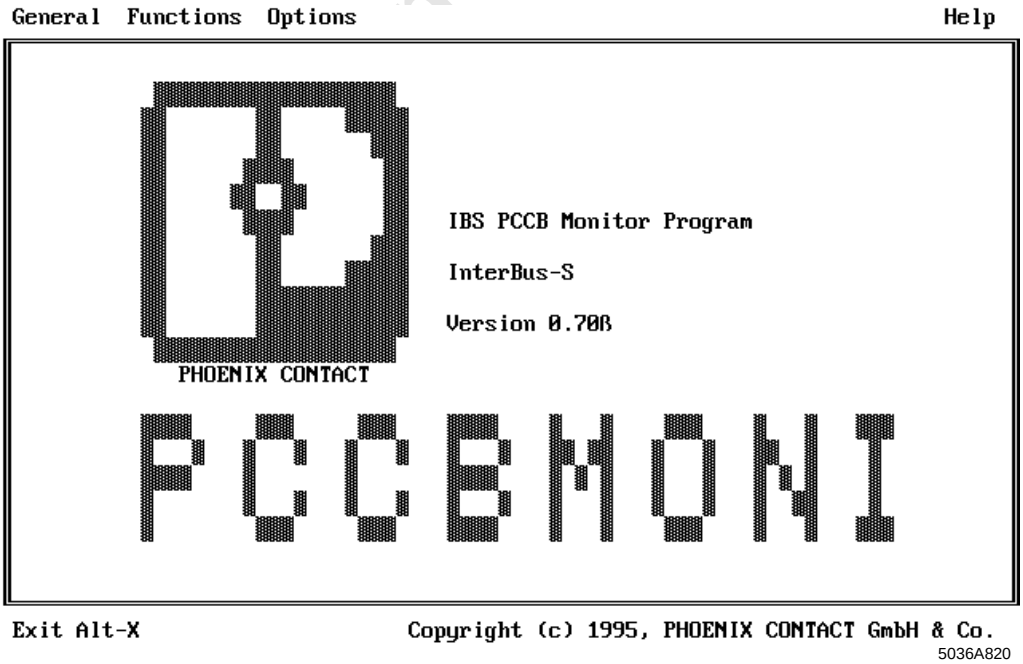

Bild 4-13: Hauptmenü des Monitorprogramms

Sie können beim Aufruf von *PCCBMONI.EXE* folgende Parameter angeben:

/MONO	Schwarz/Weiß-Darstellung für monochrome Monitore
/NOTCOMPATIBLE	Bei Problemen Ihres Grafikadapters bei der Cursorsteuerung (z.B. kein Cursor vorhanden.)
/WARMSTART	Bei jedem Start des InterBus-S löst die Anschaltbaugruppe zuvor einen Warmstart aus.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

/SWITCH	Zeigt die zur Zeit wirksamen DIP-Schalterstellungen an. Die Einstellungen der DIP-Schalter werden nur beim Booten der Anschaltbaugruppe eingelesen. Nach einer Änderung der Einstellung muß ein Reset der Anschaltbaugruppe erfolgen, um die Änderung wirksam werden zu lassen.
/HELP oder /?	Informationen über Kommandozeilenoptionen
/BOARDNO = n	Auswahl der Anschaltbaugruppe, auf die das Monitorprogramm zugreifen soll. Gültige Werte für BOARDNO sind 1, 2, 3 oder 4.

Das Prozeßdaten-Monitorprogramm bietet unter drei Pull-Down-Menüs verschiedene Menüpunkte.

General: Allgemeine Programmfunktionen
Functions: Funktionen, um IBS zu steuern und IBS-Daten anzuzeigen
Options: Verschiedene Programm- und Busfunktionen

Über die Taste **[F1]** oder die Tastenkombination **[ALT] [H]** erreichen Sie die Hilfe zu den Menüpunkten.

4.9.1 Das Pull-Down-Menü *Functions*

Durch Betätigen der Taste **[F]** bzw. der Tastenkombination **[ALT] [F]** gelangen Sie in das Pull-Down-Menü *Functions*. Es bietet Funktionen, um InterBus-S zu steuern und Daten anzuzeigen.

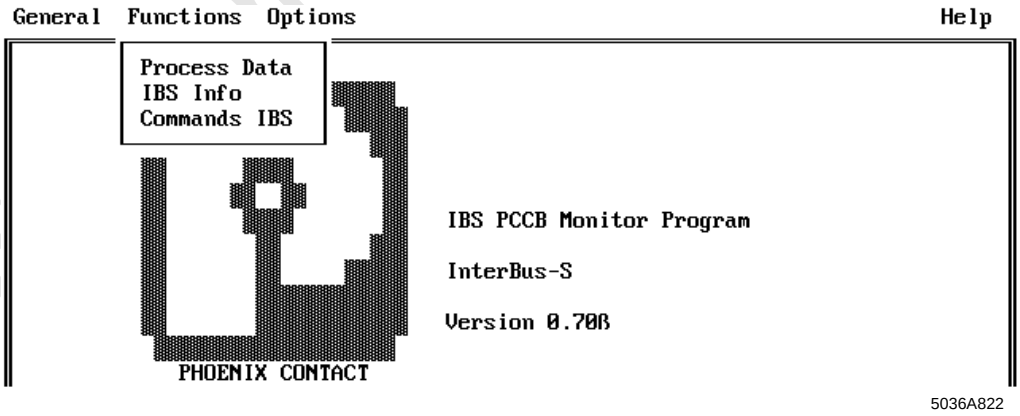

Bild 4-14: Das Pull-Down-Menü *Functions* des Monitorprogramms

Process Data	Zeigt die Eingangsdaten an und setzt Ausgangsdaten.
IBS Info	Zeigt den Buszustand und Busfehler an.
IBS Commands	Ermöglicht (ab Version 0.70) das Absetzen von Kommandos und zeigt eingegangene Meldungen an

Process Data

Durch Betätigen der Taste **[P]** für *Process Data* gelangen Sie in die Monitor-Maske.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

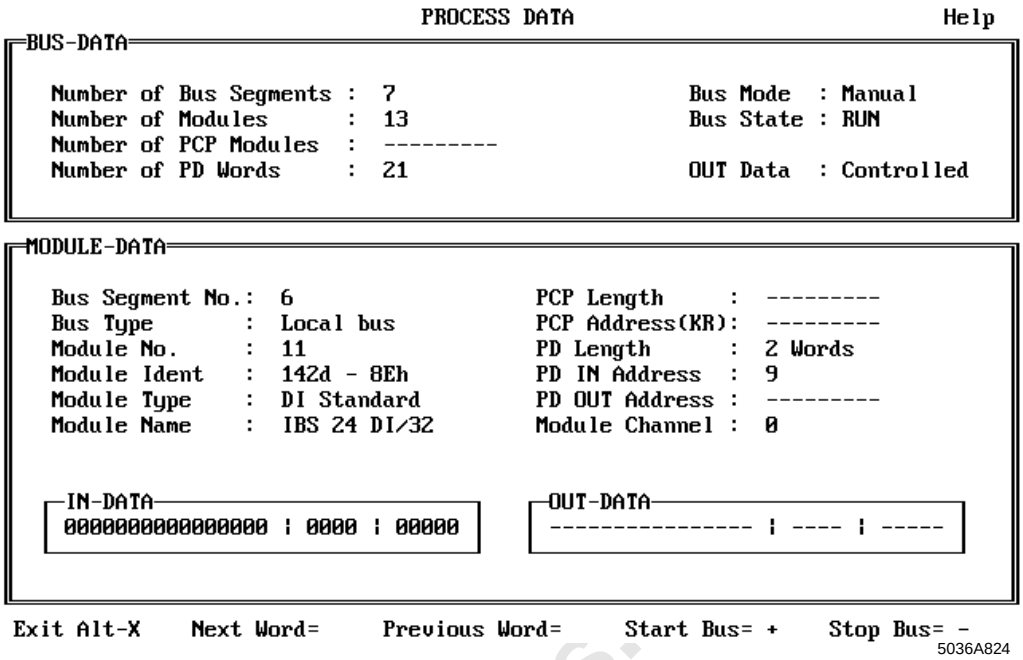

Bild 4-15: Monitor-Maske des Monitorprogramms

Drücken Sie nun die Taste **[+]**, um die Datenübertragung auf dem InterBus-S zu starten. Die Anzeige *Bus State* im Fenster *BUS-DATA* wechselt von *STOP* (rot) nach *RUN* (grün). Liegt kein Fehler vor, so können Sie über die Cursortasten alle angeschlossenen InterBus-S-Module anwählen und deren E/A-Zustand überprüfen bzw. verändern.

Tastatur-belegung

[+]	Start der Datenübertragung auf dem InterBus-S
[–]	Stop der Datenübertragung auf dem InterBus-S
[↑]	Wechselt zum nächsten Prozeßdatenwort
[↓]	Wechselt zum vorherigen Prozeßdatenwort
[Bild↑]	Sprung zum nächsten Bussegment
[Bild↓]	Sprung zum vorherigen Bussegment
[Pos1]	Sprung zum physikalisch ersten InterBus-S Modul
[Ende]	Sprung zum physikalisch letzten InterBus-S Modul
[ESC]	Zurückspringen zur Startmaske (Buszyklen werden nicht beendet)
[ALT] [X]	Beenden des Monitorprogrammes (Buszyklen werden automatisch gestoppt.)

Das obere Fenster zeigt auf der linken Seite die allgemeinen Systemdaten:

- Anzahl der angeschlossenen Bussegmente
- Anzahl der angeschlossenen Module (IBS-Teilnehmer)
- Anzahl der angeschlossenen PCP-Module (kommunikationsfähige IBS-Teilnehmer)
- Anzahl der Prozeßdatenworte des angeschlossenen Busaufbaus

Das obere Fenster zeigt auf der rechten Seite Informationen zu Programmeinstellungen und ob der Bus Prozeßdaten überträgt.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Das untere Fenster zeigt Informationen zum aktuell angewählten Modul:

Bus Segment No.:	physikalische Nummer des Bussegments, in dem sich das aktuelle Modul befindet (beginnend mit 0)
Bus Type:	Busart (*Local Bus* für Peripheriebus, *Remote Bus* für Fernbus)
Module No.:	physikalische Nummer des Moduls im Busaufbau (beginnend mit 0)
Module Ident:	ID-Code des Moduls in dezimaler und hexadezimaler Darstellung
Module Type:	Modulart
Module Name:	Bezeichnung des Moduls
PCP Length:	PCP-Adreßraum, den das aktuelle Modul für PCP im Host belegt
PCP Adress:	PCP-Adresse, entspricht der Kommunikations-Referenz (KR)
PD Length:	Registerlänge
PD IN Address:	physikalische Adresse des angezeigten Eingangsworts
PD OUT Address:	physikalische Adresse des angezeigten Ausgangsworts
Module Channel:	Kanalnummer für Module mit mehreren Prozeßdatenworten
IN-DATA:	aktuelles Eingangswort in binärer, hexadezimaler und dezimaler Darstellung
OUT-DATA:	aktuelles Ausgangswort in binärer, hexadezimaler und dezimaler Darstellung

Fehlermeldung

Wenn im Busaufbau ein Fehler auftritt, erscheint ein rotes Fenster mit einer Fehlermeldung. Die Fehlermeldung gibt die Ursache der Störung an (z.B. Remote Bus-Fehler 0, wenn das Fernbuskabel nicht angeschlossen ist).

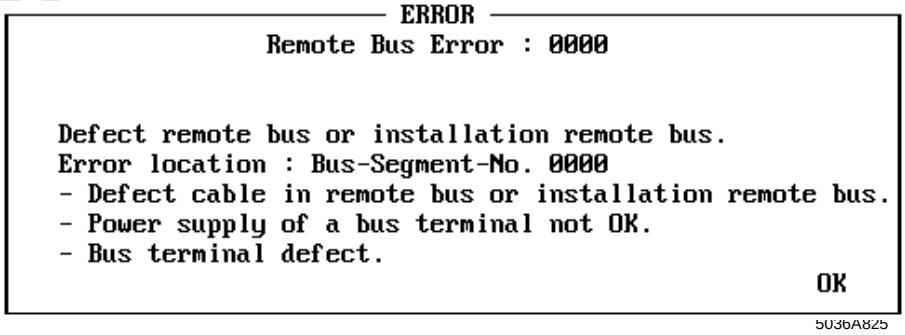

Bild 4-16: Eine Fehlermeldung im Monitorprogramm

Informationen zu Bedeutung, Ursache und Abhilfe bei folgenden Fehlermeldungen finden Sie im Kapitel 9.

Info-Fenster

Der Menüpunkt *IBS-Info* ruft das Info-Fenster auf, das Ihnen auf einen Blick den Zustand des InterBus-S und etwaige Fehlermeldungen anzeigt.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

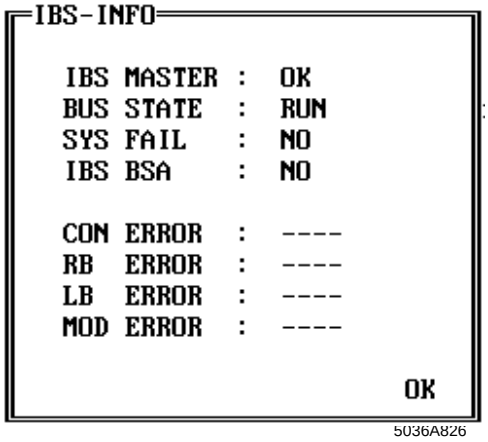

Bild 4-17: Das Info-Fenster des Monitorprogramms

Controller Error	Controllerfehler (CON ERR)
Remote Bus Error	Fernbusfehler (RB ERR)
Local Bus Error	Peripheriebusfehler (LB ERR)
Module Error	Modulfehler (MOD ERR)

4.9.2 Absetzen von Kommandos mit PCCBMONI

Kommandos

Der Menüpunkt *Commands IBS* ermöglicht das manuelle Absetzen von IBS-Kommandos (Requests) sowie das Empfangen von Meldungen (Confirmations) inkl. Parametern. Sie können auch PCP-Kommandos absetzen und die zugehörigen Meldungen empfangen. PCP-Kommandos werden z.B. zum Parametrieren von intelligenten IBS-Teilnehmern wie IBS V.24 oder Frequenzumrichtern benötigt.

Der Menüpunkt *IBS Commands* wird erst ab Version 0.70 des PCCBMONI unterstützt.

Die Maske besteht aus zwei Fenstern:

Das linke Fenster ist zum Eintragen des gewünschten Kommandos (Request) mit den zugehörigen Parametern. Tragen Sie in die Zeile *REQ* den Kommando-Code in hexadezimaler Form ein. Falls Parameter zum Kommando gehören, müssen Sie in der Zeile *PC* (Parameter Counter) die Anzahl der nachfolgenden Parameter und in die folgenden Zeilen diese Parameter eintragen. Durch Betätigen von **[ALT] [S]** wird das Kommando zur Anschaltbaugruppe übertragen.

Meldungen

Im rechten Fenster wird die zuletzt empfangene Meldung (Last Confirmation) angezeigt. Die Zeile *CNF* zeigt den empfangenen Meldungs-Code an. Falls Parameter zur Meldung gehören, zeigt die Zeile *PC* (Parameter Counter) die Anzahl der nachfolgenden Parameter und die folgenden Zeilen diese Parameter an.

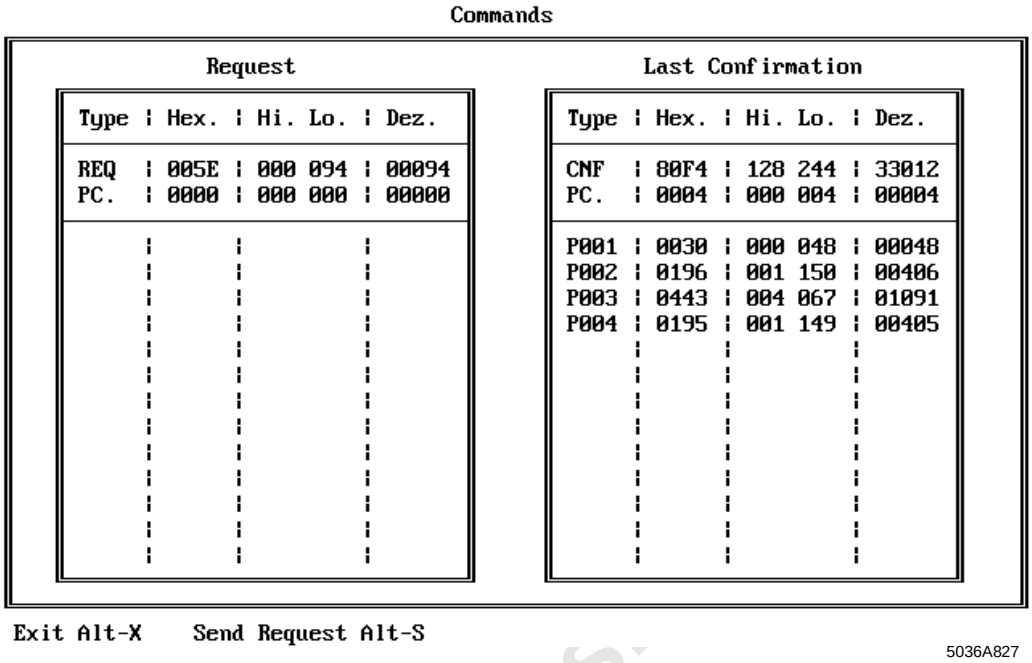

Bild 4-18: Maske für Kommandos und Meldungen

Drei Spalten stellen die einzelnen Parameter in folgender Form dar:

Hex: hexadezimale Darstellung
Hi. Lo.: dezimale Darstellung des höherwertigen (Hi.) und niederwertigen (Lo.) Bytes. Hier können Sie z.B. Längen-Code (Hi.) und ID-Code (Lo.) für einen IBS-Teilnehmer in dezimaler Form eingeben.
Dez: Dezimale Darstellung

Mit **[TAB]** bzw. **[SHIFT] [TAB]** können Sie zwischen den Spalten für die verschiedenen Darstellungsformen wechseln. Die Umrechnung erfolgt automatisch.

Scrollen Sie mit **[↑]** oder **[↓]** durch die Zeilen. Wenn im rechten Fenster die Anzahl der anzuzeigenden Parameter den darstellbaren Bereich (Parameteranzahl > 13) überschreitet, dann können Sie mit **[SHIFT] [↑]** oder **[SHIFT] [↓]** durch die Parameterliste scrollen.

Mit folgenden Tastenkombinationen können Sie direkt zwischen den Fenstern des Pull-Down-Menüs *Funktions* wechseln:

[ALT] [P] Wechsel zum Fenster *Process Data*
[ALT] [I] Wechsel zum Fenster *Info*
[ALT] [O] Wechsel zum Fenster*Command*

Über die Tasten **[Esc]**, **[X]** oder **[Alt] [X]** gelangen Sie zur nächsthöheren Menueebene.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4.9.3 Das Pull-Down-Menü *Options*

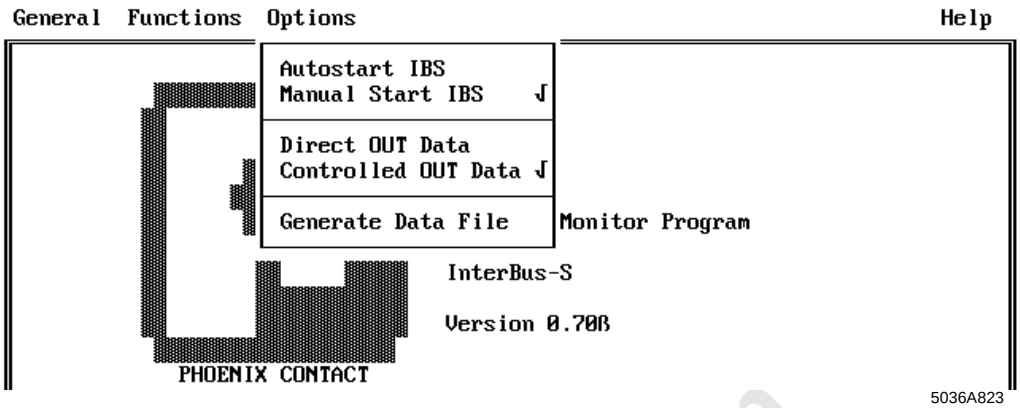

Bild 4-19: Das Pull-Down-Menü *Options*

Mit dem Pull-Down-Menü *Options* läßt sich das Monitorprogramm konfigurieren. Über die Menüpunkte *Autostart IBS* oder *Manual Start IBS* wählen Sie, ob die Anschaltbaugruppe nach Auftreten eines Fehlers automatisch versucht das angeschlossene Bussystem wieder in Betrieb zu nehmen (Autostart IBS). Bei *Manual Start IBS* erfolgt der Neustart erst nach Drücken der Taste **[+]**.
Über die Menüpunkte *Direct OUT Data* oder *Controlled OUT Data* wählen Sie, ob das Ändern des entsprechenden Ausganges unmittelbar nach Eintrag eines Wertes in die Maske erfolgt (Direct OUT Data) oder erst nach Quittierung mit der Taste **[↵]** (Controlled OUT Data).
Der Menüpunkt *Generate Data File* generiert eine Datei (IDLST.TXT), die Informationen zum aktuell angeschlossenen Busaufbau (Längen-Code, ID-Code, Adresse etc.) enthält.

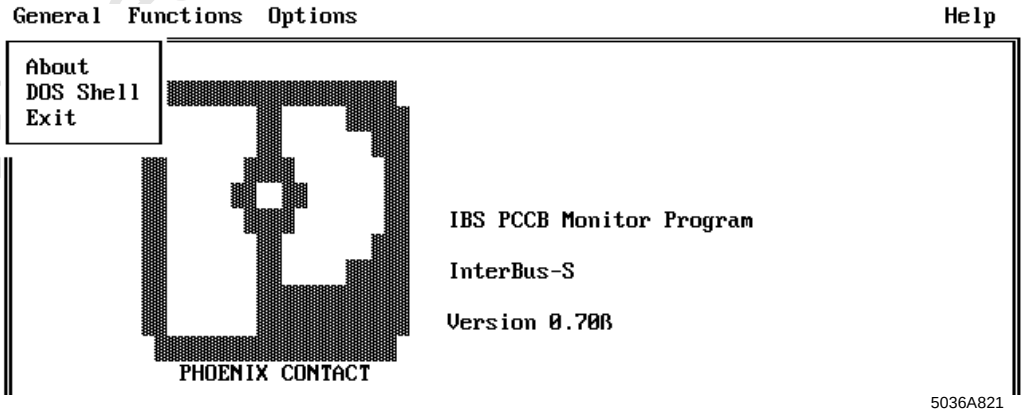

Bild 4-20: Das Pull-Down-Menü *General*

About	Informationen über die Version des Monitorprogramms
DOS shell	Wechselt zur DOS-Oberfläche. Der InterBus-S behält den aktuellen zustand bei. Verlassen Sie die DOS-Oberfläche durch Eingabe von *EXIT* und **[↵]**
Exit	Stoppt die Datenübertragung auf dem InterBus-S (Alarm_Stop_Request) und beendet das Monitorprogramm

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel 5

Schnittstellen zwischen Hardware und Software

Dieses Kapitel informiert Sie über

- die Schnittstellen der Anschaltbaugruppen
- die allgemeine Struktur der Treibersoftware

5 Schnittstellen zwischen Hardware und Software 5-3

5.1 Multi-Port-Memory 5-3
5.1.1 Das MPM im Adreßraum des Hosts 5-3
5.1.2 Aufteilung des MPMs 5-4
5.2 Allgemeine Struktur der Treibersoftware 5-7
5.2.1 Realisierung des DDI und des DD 5-9
5.2.2 Struktur der Treibersoftware auf dem Coprozessorboard . . . 5-9
5.2.3 Begriffserklärungen zur Treibersoftware 5-10
5.2.3.1 Verwaltung von Datenkanälen 5-10
5.2.3.2 Mailbox-Interface 5-12
5.2.3.3 Data-Interface 5-12
5.2.3.4 Diagnosefunktion 5-12
5.3 Nutzung des statischen RAMs 5-13
5.4 Kommunikation zwischen Host und COP 5-13
5.4.1 Aufbau einer Nachricht zwischen Host und COP 5-14
5.5 Überwachung durch Watchdogs 5-14
5.5.1 Watchdog des IBS-Masterboards 5-14
5.5.2 Watchdog zur Überwachung des Hosts 5-15
5.5.3 Watchdog des Coprozessorboards 5-15
5.5.4 Das Signal SysFail 5-16
5.6 Download des Anwenderprogramms auf den COP 5-16

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

5 Schnittstellen zwischen Hardware und Software

Dieses Kapitel beschreibt allgemein die Schnittstellen zwischen der Anschaltbaugruppe und der Treibersoftware.

5.1 Multi-Port-Memory

Die zentrale Schnittstelle der Anschaltbaugruppen ist das **M**ulti-**P**ort-**M**emory (MPM). Das MPM ist ein Speicher auf dem Grundboard, auf den alle MPM-Teilnehmer (Host, Coprozessorboard und IBS-Masterboard) zugreifen können. Die MPM-Teilnehmer speichern im MPM alle Daten, die gemeinsam genutzt werden sollen. Das MPM ist die einzige Verbindung zwischen den MPM-Teilnehmern.

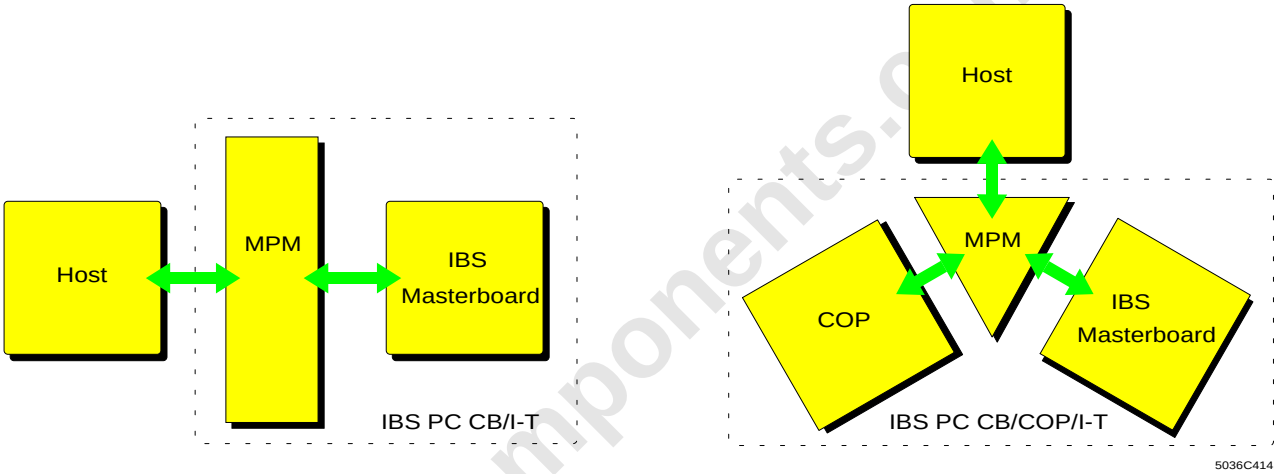

Bild 5-1: Das MPM als zentrale Schnittstelle einer Anschaltbaugruppe

In Bild 5-1 ist die zentrale Position des MPMs dargestellt. Die gestrichelten Linien umfassen die auf den Anschaltbaugruppen integrierte Hardware.

Ein direkter Datenaustausch zwischen den MPM-Teilnehmer kann nur über das MPM stattfinden. Das MPM besitzt eine vorgegebene Struktur, die vom Anwender nicht verändert werden darf.

Zugriffe auf das MPM dürfen nur über die Funktionen des Device-Drivers erfolgen. Ein direktes Lesen oder Beschreiben des MPMs ist nicht erlaubt!

5.1.1 Das MPM im Adreßraum des Hosts

Das MPM wird in den Adreßraum des Hosts (zwischen 640 KB und 1MB) eingeblendet. Da der dort zur Verfügung stehende Platz beschränkt ist, wird nur ein 4 KByte großes Memory-Fenster verwendet. Der Device-Driver verschiebt dieses Fenster automatisch (und schnell, durch Programmierung der Adressierungslogik der Anschaltbaugruppe) auf den jeweils benötigten Adreßbereich des MPMs. Die Adresse des Fensters im Speicherbereich des Hosts bleibt dabei konstant!

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

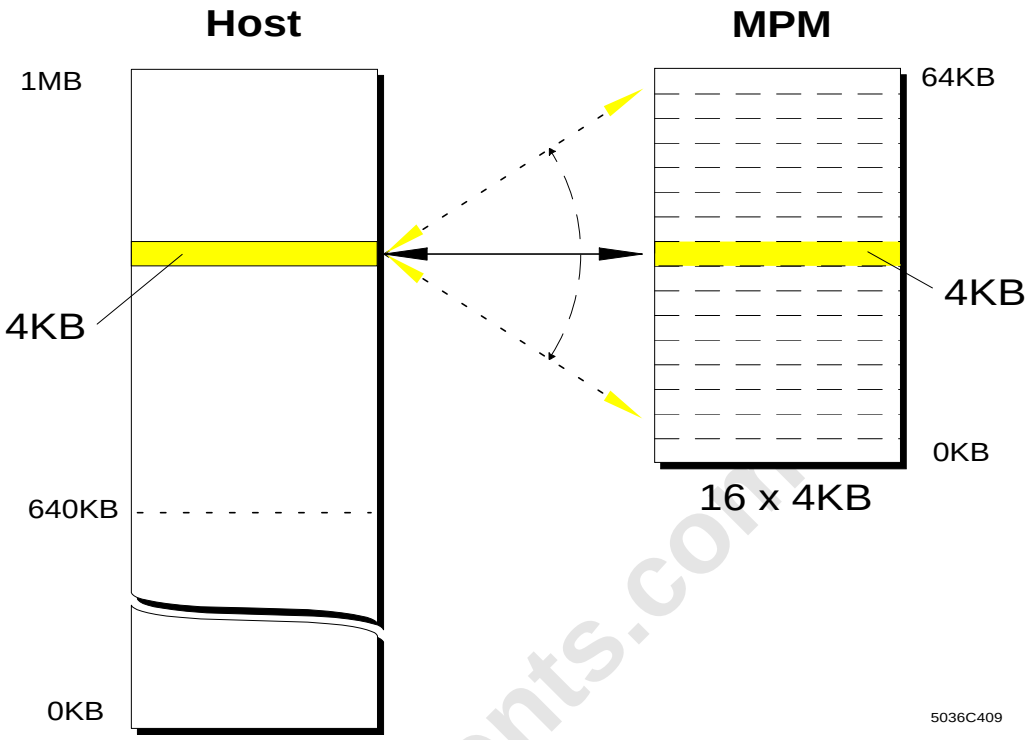

Bild 5-2: Verschieben des MPM-Fensters durch den Device-Driver

5.1.2 Aufteilung des MPMs

Das 64KByte große MPM ist in vier Teilnehmerbereiche zu je 16KByte aufgeteilt. Drei Teilnehmerbereiche sind an die MPM-Teilnehmer Host (PC), IBS-Masterboard (IBS-MA) und Coprozessorboard (COP) vergeben. Der vierte Teilnehmerbereich ist für spätere Erweiterungen reserviert. Jeder MPM-Teilnehmer darf in seinem Bereich lesen und schreiben, in den Bereichen der anderen aber nur lesen. Der Device-Driver wählt beim Öffnen eines Datenkanals anhand des Device-Names automatisch den richtigen Teilnehmerbereich aus. Ein 16KByte großer Teilnehmerbereich besteht aus 4 Übertragungsbereichen von je 1KByte Größe für das Data-Interface (DTI) und 12 weiteren KByte, die u.a. für das Mailbox-Interface (MXI) genutzt werden.

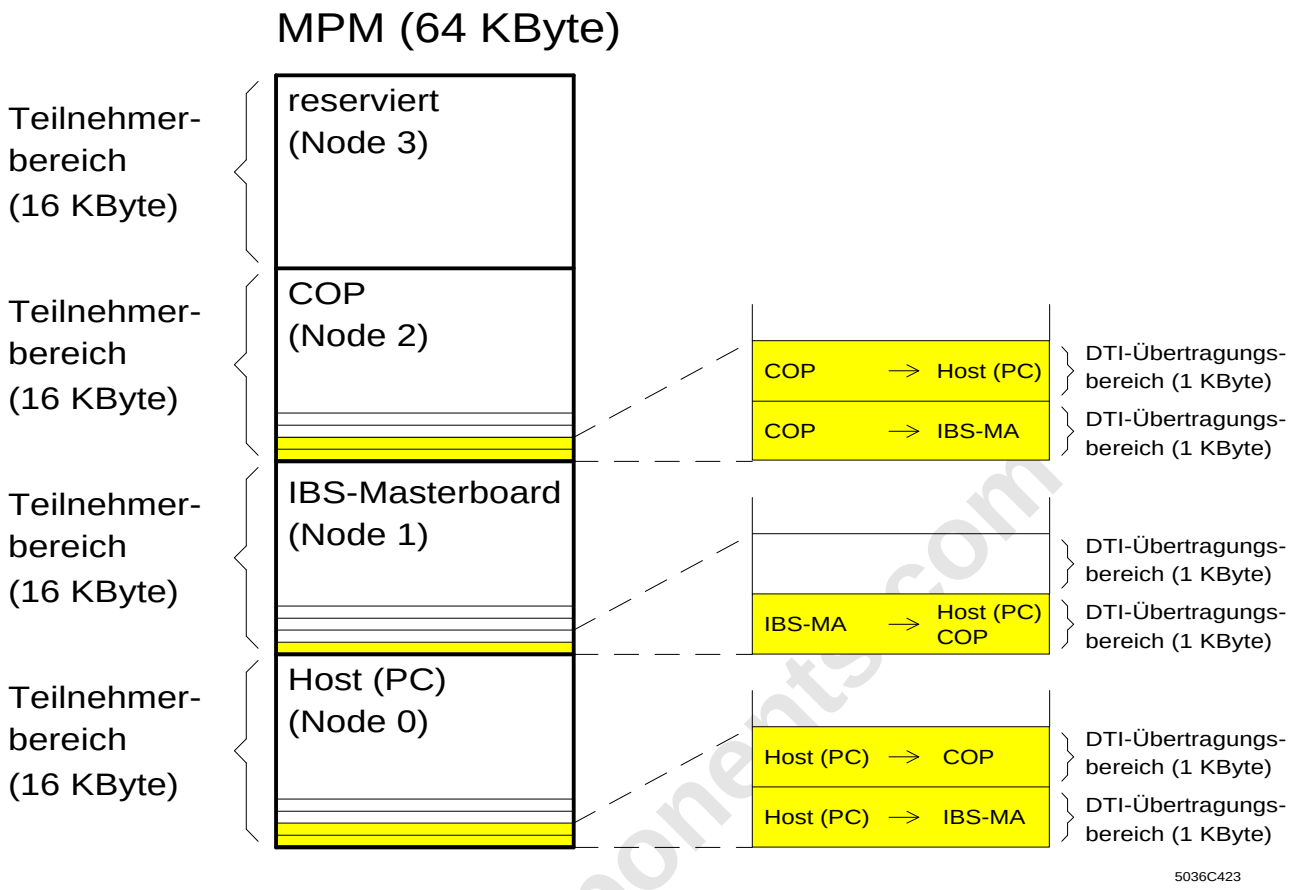

Bild 5-3: Aufteilung des MPMs

Die DTI-Adresse eines Prozeßdatenworts innerhalb eines Teilnehmerbereiches setzt sich aus zwei Komponenten zusammen:

- Dem Adreß-Offset des DTI-Übertragungsbereiches
- Der relativen Adresse innerhalb dieses DTI-Übertragungsbereiches.

Adreß-Offset: Der Adreß-Offset ist der Abstand eines 1 KByte großen DTI-Übertragungsbereiches vom Beginn des 16 Kbyte großen Teilnehmerbereiches.

Relative Adresse: Die relative Adresse ist die Adresse eines Prozeßdatenwortes innerhalb dieses DTI-Übertragungsbereiches. Sie wird wie der Parameter *Length* in Bytes angegeben, weshalb nur gerade Byte-Adressen möglich sind. Das erste Prozeßdatenwort liegt demnach auf der relativen Adresse 0, das zweite auf der relativen Adresse 2, das dritte auf der relativen Adresse 4 usw.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

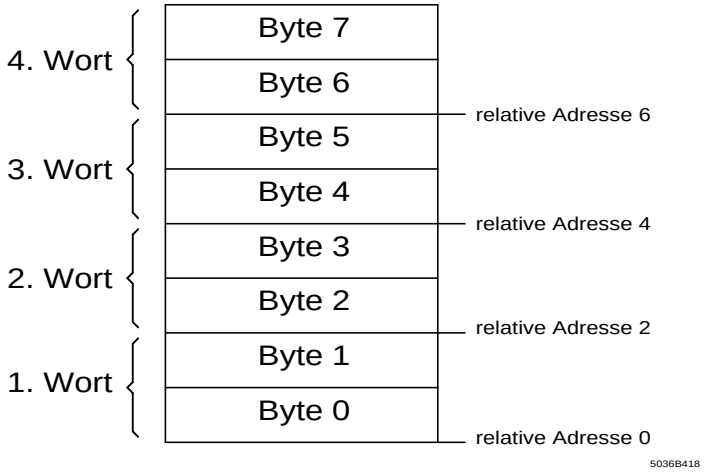

Bild 5-4: Relative Adressen von Prozeßdatenworten im DTI-Übertragungsbereich

Zur Ansprache eines Prozeßdatenwortes müssen Adreß-Offset und relative Adresse addiert werden:

DTI-Adresse = Adreß-Offset + relative Adresse

Zur einfachen Handhabung der Adreß-Offsets für die verschiedenen Datenbereiche stehen folgende Offset-Konstanten zur Verfügung, die in der Include-Datei *DDI_USR.H* definiert sind:

Tabelle 5-1: Offset-Konstanten

Zugriff		**Offset-Konstante**
von	**auf**	
Host	IBS-Masterboard	DTA_OFFST_IBS_MASTER
COP	IBS-Masterboard	DTA_OFFST_IBS_MASTER
Host	COP	DTA_OFFST_PC_TO_COP
COP	Host	DTA_OFFST_COP_TO_PC

Sie müssen also nur die dem Zugriff entsprechende Offset-Konstante zur relativen Adresse addieren.

DTI-Adresse = Offset-Konstante + relative Adresse

Geben Sie den Parameter *DTI-Adresse* beim Aufruf der Funktionen *DDI_DTI_WriteData* und *DDI_DTI_ReadData* an (siehe Handbuch zur Treibersoftware *IBS PC CB SWD UM* Artikel-Nr. 27 53 94 4).

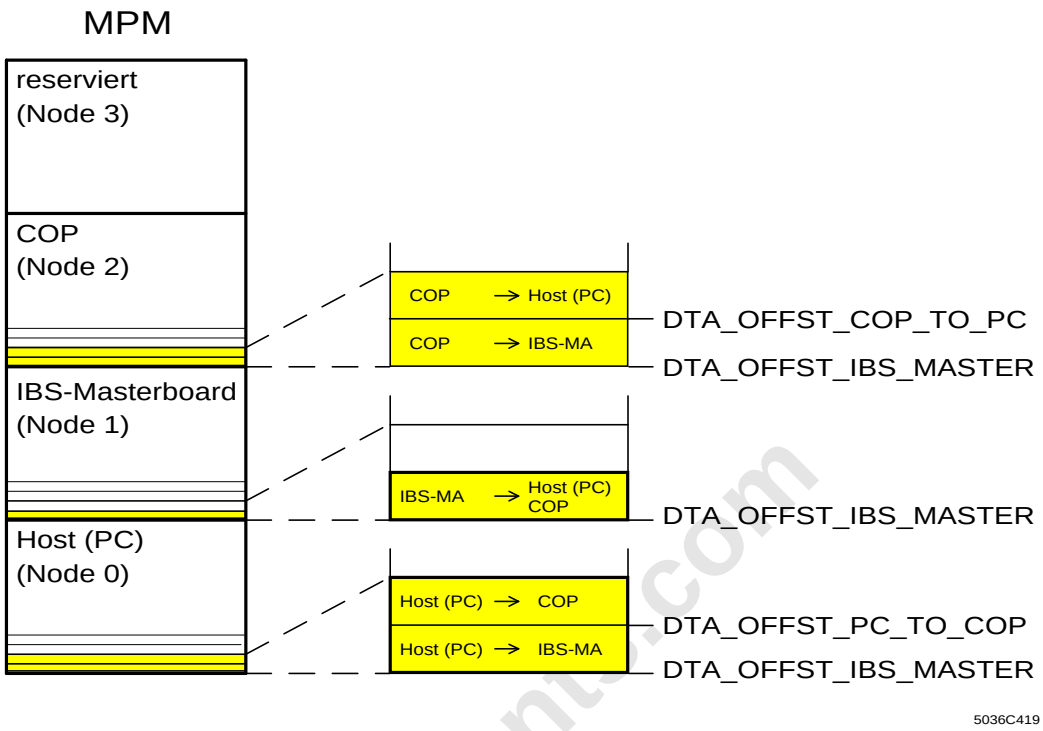

Bild 5-5: DTI-Übertragungsbereiche und Offset-Konstanten

5.2 Allgemeine Struktur der Treibersoftware

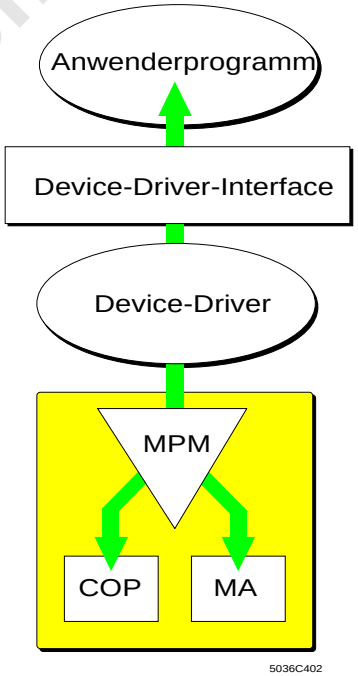

Bild 5-6: Struktur der Treibersoftware

Für die Anschaltbaugruppen steht Ihnen Treibersoftware für die Betriebssysteme DOS, Microsoft Windows® und IBM OS/2® zur Verfügung.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Die Treibersoftware besteht aus zwei Teilen:

1. Dem Device-Driver-Interface (DDI), einer Compiler-spezifischen Schnittstelle zum Anwenderprogramm.
2. Dem betriebssystemspezifischen Device-Driver (DD).
   Der Device-Driver stellt über das MPM die Verbindung zwischen Host (PC) bzw. Coprozessorboard (COP) und IBS-Masterboard (MA) her.

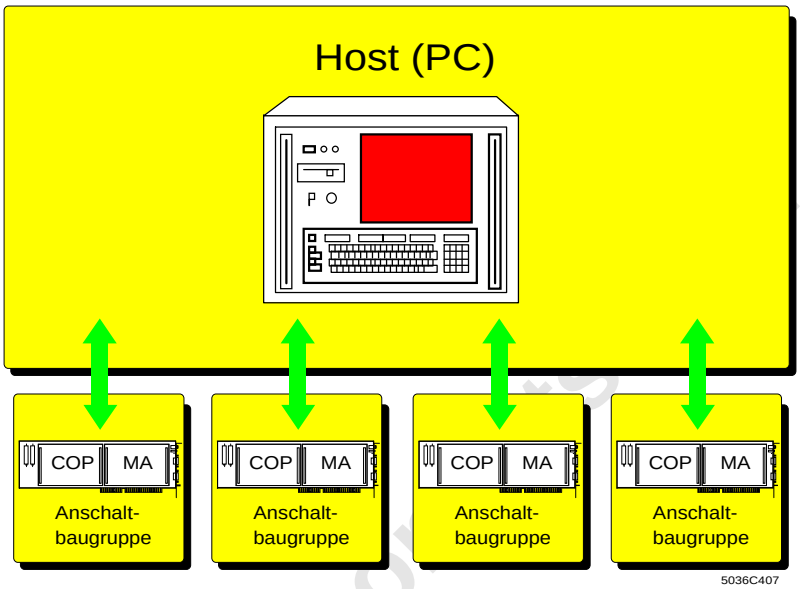

Bild 5-7: Einsatz von vier Anschaltbaugruppen in einem Host

In einem Host können bis zu vier Anschaltbaugruppen eingesetzt werden. Pro Anschaltbaugruppe muß ein Device-Driver installiert werden. Das Device-Driver-Interface übernimmt dabei die Verwaltung und Kontrolle von allen Device-Drivern:

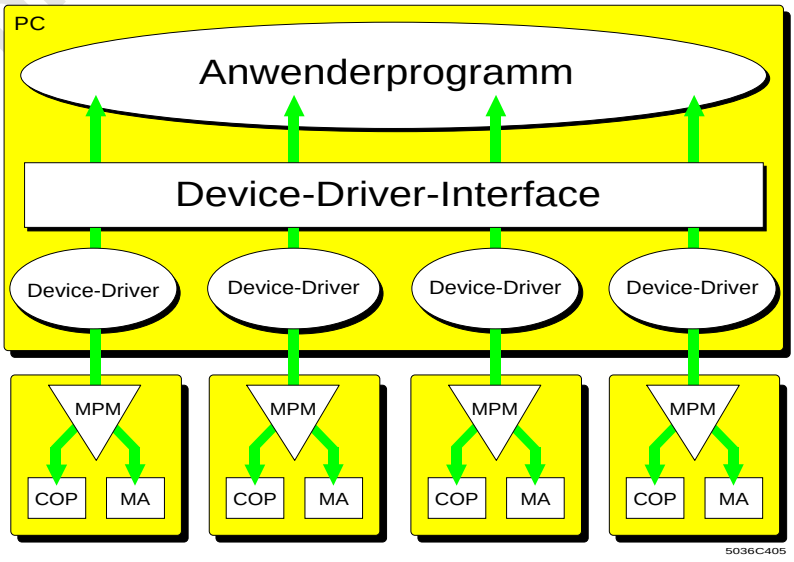

Bild 5-8: Kontrolle von vier Device-Drivern durch das Device-Driver-Interface

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

5.2.1 Realisierung des DDI und des DD

Die konkrete Realisierung der Compiler-spezifischen Device-Driver-Interfaces und der Device-Driver für die verschiedenen Betriebssysteme entnehmen Sie bitte dem Handbuch zur Treibersoftware (IBS PC CB SWD UM, Artikel-Nr. 27 53 94 4).

5.2.2 Struktur der Treibersoftware auf dem Coprozessorboard

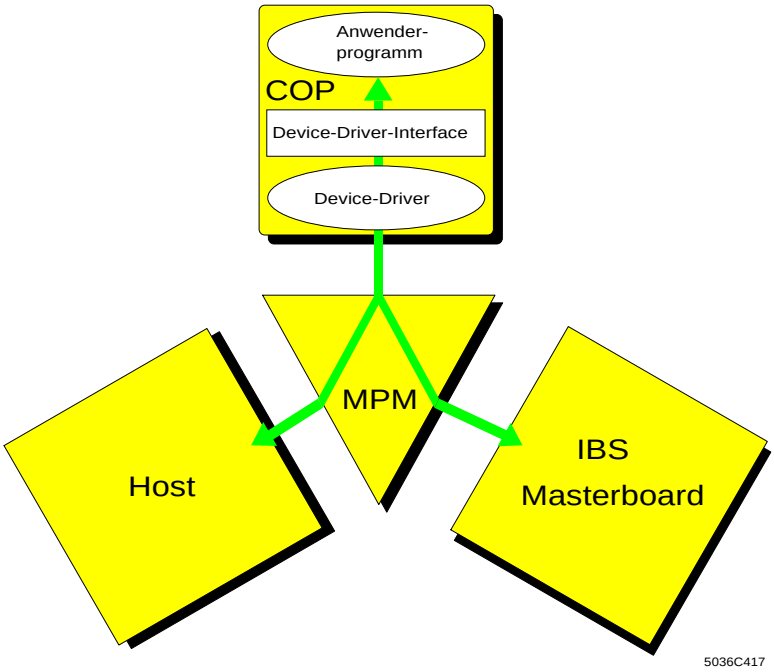

Bild 5-9: Struktur der Treibersoftware auf dem Coprozessorboard (COP)

Die Programmierung des Coprozessorboard unterscheidet sich nicht von der Programmierung auf dem Host unter DOS. Ein für den Host unter DOS erstelltes Programm kann auch auf dem Coprozessorboard ablaufen. Nutzen Sie auf dem COP anstelle von *IBSPCCB.EXE* das TSR-Progamm *IBSCOP.EXE* als Device-Driver:

- Übertragen Sie den Device-Driver *IBSCOP.EXE* mit der Entwicklungsumgebung *TDOS-PRO* auf den *COP386* des *IBS PC CB/COP/I-T*. Die genaue Beschreibung hierzu finden Sie im Handbuch zur Entwicklungsumgebung.
- Der Device-Driver *IBSCOP.EXE* ist im Auslieferungszustand des *IBS PC CB/RTX486/I-T* bereits auf Laufwerk A:\ (Flash-EPROM) des *COP486* abgelegt und wird über einen Eintrag in der *AUTOEXEC.BAT* automatisch gestartet.

Beachten Sie, daß das Coprozessorboard prinzipbedingt nicht über eigene Tastatur, Bildschirm oder Hard-Disks verfügt. Die Möglichkeiten, Tastatur und Bildschirm (Terminal-Mode) sowie die Hard-Disks des Hosts für das Coprozessorboard zu nutzen, sind in den Dokumentationen zu dessen Betriebssystemen (TDOS/RTXDOS) beschrieben.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

5.2.3 Begriffserklärungen zur Treibersoftware

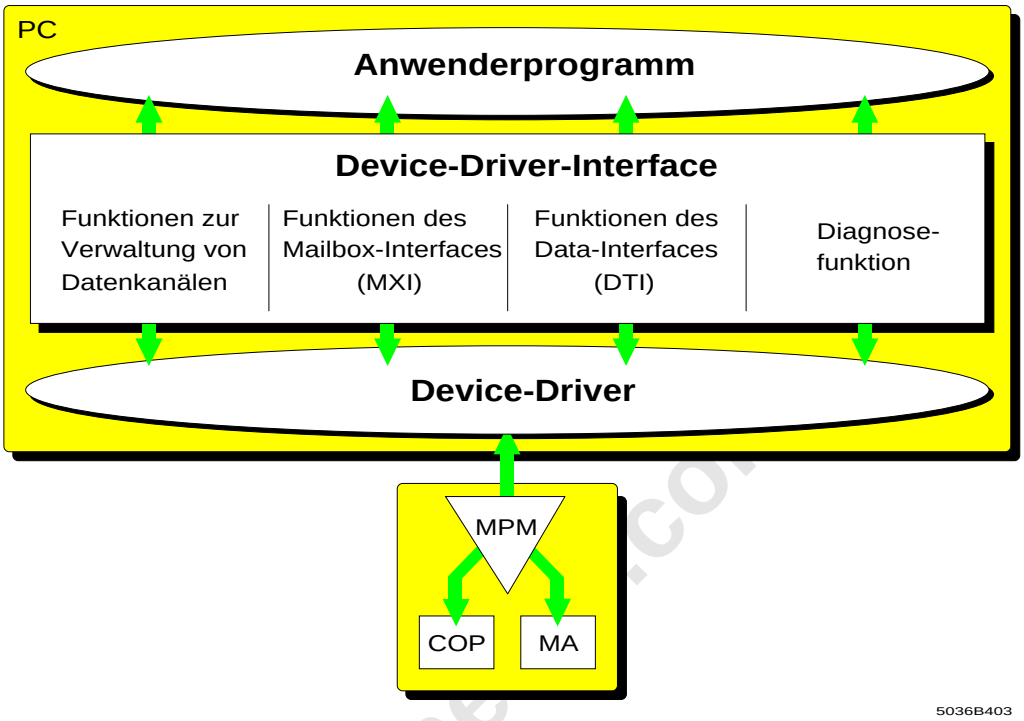

Bild 5-10: Das DDI als Interface zu einem Device-Driver

5.2.3.1 Verwaltung von Datenkanälen

Node

Ein MPM-Teilnehmer mit zugehörigem Device-Driver wird als Node bezeichnet. Es finden folgende Nodes Verwendung:

Node 0: Host mit zugehörigem Device-Driver
Node 1: IBS-Masterboard mit zugehörigem Device-Driver (Der Device-Driver des IBS-Masterboards ist Bestandteil von dessen Firmware.)
Node 2: Coprozessorboard mit zugehörigem Device-Driver

Node-Handle

Ein Node-Handle identifiziert einen geöffneten Datenkanal zu einem Node.

Device-Name

Name des Gerätes, zu dem ein Datenkanal geöffnet werden soll. Der Name legt die Anschaltbaugruppe (Boardnumber 1 bis 4) und den dortigen MPM-Teilnehmer (Host, IBS-Masterboard oder Coprozessorboard) fest.

Siehe auch Beschreibungen zu den Funktionen *DDI_DevOpenNode* und *DDI_DevCloseNode* im Handbuch zur Treibersoftware *IBS PC CB SWD UM*.

Zur einfacheren Handhabung steht zum Öffnen eines Datenkanals für den Parameter *Device-Name* ein String zur Verfügung, der Boardnumber und MPM-Teilnehmer zuordnet.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 5-2: Öffnen eines Datenkanals vom Host zum IBS-Masterboard

Anschaltbaugruppe (Boardnummer)	MPM-Teilnehmer	Interface	String für den Parameter *Device-Name* (devName)
Board 1	IBS-Masterboard	Mailbox-Interface	IBB1N1_M
Board 1		Data-Interface	IBB1N1_D
Board 2		Mailbox-Interface	IBB2N1_M
Board 2		Data-Interface	IBB2N1_D
Board 3		Mailbox-Interface	IBB3N1_M
Board 3		Data-Interface	IBB3N1_D
Board 4		Mailbox-Interface	IBB4N1_M
Board 4		Data-Interface	IBB4N1_D

Tabelle 5-3: Öffnen eines Datenkanals vom Host zum Coprozessorboard (für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Anschaltbaugruppe (Boardnummer)	MPM-Teilnehmer	Interface	String für den Parameter *Device-Name* (devName)
Board 1	Coprozessorboard	Mailbox-Interface	IBB1N2_M
Board 1		Data-Interface	IBB1N2_D
Board 2		Mailbox-Interface	IBB2N2_M
Board 2		Data-Interface	IBB2N2_D
Board 3		Mailbox-Interface	IBB3N2_M
Board 3		Data-Interface	IBB3N2_D
Board 4		Mailbox-Interface	IBB4N2_M
Board 4		Data-Interface	IBB4N2_D

Tabelle 5-4: Öffnen eines Datenkanals vom Coprozessorboard zum Host (für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Anschaltbaugruppe (Boardnummer)	MPM-Teilnehmer	Interface	String für den Parameter *Device-Name* (devName)
Controllerboard 1 bis Controllerboard 4	Host	Mailbox-Interface	IBB1N0_M
		Data-Interface	IBB1N0_D

Tabelle 5-5: Öffnen eines Datenkanals vom Coprozessorboard zum IBS-Masterboard (für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Anschaltbaugruppe (Boardnummer)	MPM-Teilnehmer	Interface	String für den Parameter *Device-Name* (devName)
Controllerboard 1 bis Controllerboard 4	IBS-Masterboard	Mailbox-Interface	IBB1N1_M
		Data-Interface	IBB1N1_D

Setzen Sie die Strings für den Parameter *Device-Name* bei Verwendung von *C* in doppelte (z.B. "IBB1N1_M") und bei Verwendung von *Turbo-Pascal* in einfache Anführungszeichen (z.B. ‘IBB1N1_M‘).

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Datenkonsistenz

Die Datenkonsistenz (Data-Consistency) ist die Anzahl von Bytes, die ein Node lesen oder schreiben kann, ohne daß ein anderer Node gleichzeitig auf diese Bytes zugreift. Das bedeutet, daß der Zugriff eines Nodes auf eine durch die Datenkonsistenz festgelegte Anzahl von Bytes erst abgeschlossen sein muß, bevor ein anderer Node auf diese Bytes zugreifen kann. Die MPM-Logik verhindert so gleichzeitige Zugriffe der Nodes auf Daten des MPMs.

Mögliche Werte für die Datenkonsistenz sind:

- Word-Datenkonsistenz (2 Byte) DTI_DATA_WORD
- Longword-Datenkonsistenz (4 Byte) DTI_DATA_LWORD
- Byte-Datenkonsistenz (1 Byte) DTI_DATA_BYTE
- 48 Bit-Datenkonsistenz (6 Byte) DTI_DATA_48BIT

5.2.3.2 Mailbox-Interface

Das Mailbox-Interface (MXI) dient zur Übertragung von Kommandos und Meldungen zwischen den MPM-Teilnehmern.

Auf einige Kommandos erfolgt als Quittung eine Meldung über die erfolgreiche oder fehlerhafte Ausführung des Kommandos. Die Übertragung der Quittung findet ebenfalls über das Mailbox-Interface statt.

Falls eine Meldung vom Anwenderprogramm nicht abgeholt wird, erfolgt nach *8* Minuten ein Time-Out-Fehler. Diese Fehlermeldung ist ebenfalls eine Meldung und überschreibt die bisher eingetragene Meldung. Liegt mehr als eine Meldung an, verkürzt sich die Time-Out-Zeit auf 8,1 Sekunden.

5.2.3.3 Data-Interface

Das Data-Interface (DTI) dient zur Übertragung von E/A-Daten zwischen den MPM-Teilnehmern. Die Übertragung geschieht ohne Rückmeldung (Quittung)

5.2.3.4 Diagnosefunktion

Die Anschaltbaugruppe besitzt im MPM zwei Register zur Auswertung von Fehleranzeigen durch das Anwenderprogramm:

- Das für Diagnosebitregister gibt Betriebs- und Fehlerzustände der Anschaltbaugruppe wieder.
- Das Diagnoseparameterregister liefert zusätzliche Informationen zu Fehlerart oder Fehlerort.

Die Register belegen jeweils ein Wort im Adreßbereich des MPMs. Sie können beide Register mit der Diagnosefunktion auswerten. Die Diagnosefunktion ist im Handbuch zur Treibersoftware ausführlich beschrieben.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

5.3 Nutzung des statischen RAMs

(für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T)

Das Coprozessorboard des Controllers IBS PC CB/COP/I-T stellt 128 KByte statisches RAM (SRAM) zur Verfügung. Das SRAM wird über die Batterien des Grundboards gepuffert. Bei ausgeschaltetem Host oder nicht eingebauter Anschaltbaugruppe bleiben die im SRAM gespeicherten Daten bis zu 2,5 Jahre erhalten.

SRAM auf dem COP386

Auf dem COP386 stehen Ihnen für das Beschreiben und Lesen des SRAMs zwei Funktionen zur Verfügung, die in ihrer Syntax an die Funktionen des Data-Interfaces angelehnt sind.

Pro Funktionsaufruf können maximal 64 KByte übertragen werden. Um größere Datenmengen zu übertragen, rufen Sie die Funktion zweimal auf. Geben Sie beim zweiten Aufruf einem Offset von 64 KByte an. Die für die Nutzung des SRAMs erforderlichen Include-Dateien und Libraries (C) bzw. Units (Pascal) werden im Handbuch zur Treibersoftware beschrieben.

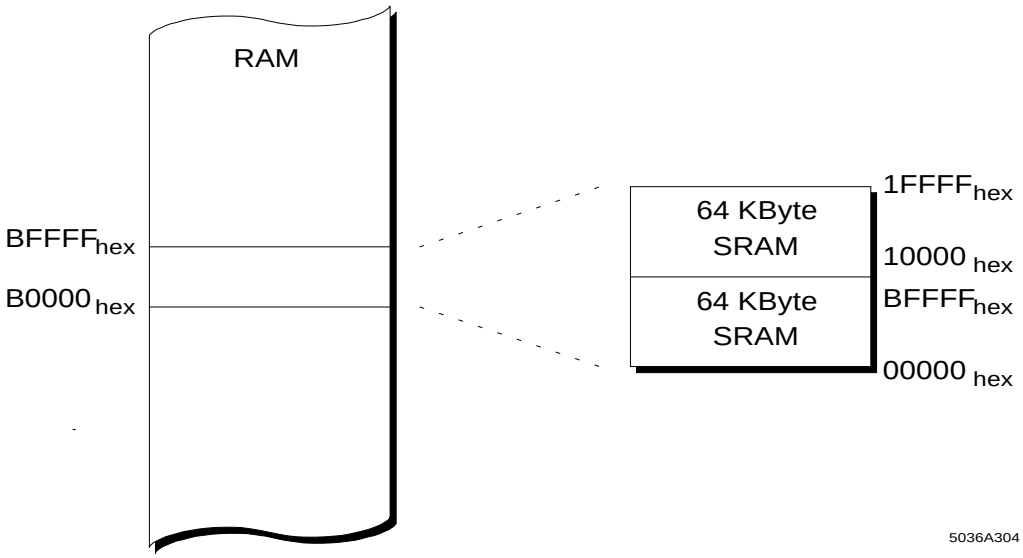

Bild 5-11: Aufteilung des SRAMs

SRAM auf dem COP486

Auf dem *COP 486* ermöglicht *RTXDOS*, das SRAM über übliche Dateifunktionen als Laufwerk D:\ anzusprechen.

5.4 Datenübertragung zwischen Host und COP

Die Datenübertragung zwischen Host und COP erfolgt prinzipiell auf die gleiche Weise wie zwischen Host und IBS-Masterboard bzw. COP und IBS-Masterboard. Verwenden Sie zum Öffnen eines Datenkanals zwischen Host und COP die in Tabelle 5-3 angegebenen Device-Namen (devName).

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

5.4.1 Aufbau einer Nachricht zwischen Host und COP

Die Kommandos und Meldungen zur Datenübertragung zwischen Host und COP sind nicht vorgegeben. Sie können sich diese Kommandos/Meldungen, unter Einhaltung einer vorgegebenen Struktur, selbst definieren. Diese Struktur ist die gleiche, wie die der Kommandos und Meldungen zur Steuerung des IBS-Masterboards.

Bei Kommandos bzw. Meldungen ohne Parameter wird nur der eigentliche Kommando- bzw. Meldungs-Code angegeben. Weitere Parameter folgen nicht.

Bei Kommandos bzw. Meldungen mit Parametern folgt unmittelbar nach dem Kommando- bzw. Meldungs-Code der Parameterzähler (PZ). Der Parameterzähler gibt die Anzahl der nachfolgenden Worte an.

Alle Nachrichtenblöcke werden im sogenannten Mailboxformat übergeben bzw. entgegengenommen. Der Aufbau dieser Blöcke vom Datentyp *USIGN16* stellt sich folgendermaßen dar :

Kommandos bzw. Meldungen ohne Parameter

Wort 1	*Kommando-Code$_{hex}$*		Wort 1	*Meldungs-Code$_{hex}$*	
Bit	15 8	7 0	Bit	15 8	7 0

Kommandos bzw. Meldungen mit Parametern

Wort 1	*Kommando-Code$_{hex}$*		Wort 1	*Meldungs-Code$_{hex}$*	
Wort 2	*Parameterzähler (n)*		Wort 2	*Parameterzähler (n)*	
Wort 3	*Parameter 1*		Wort 3	*Parameter 1*	
Wort 4	*Parameter 2*		Wort 4	*Parameter 2*	
	...			*...*	
Wort n+2	*Parameter n*		Wort n+2	*Parameter n*	
Bit	15 8	7 0	Bit	15 8	7 0

5.5 Überwachung durch Watchdogs

Die für die Nutzung der Watchdogs erforderlichen Include-Dateien und Libraries werden in Kapitel 4.6 beschrieben.

Sie können den Host und das Coprozessorboard (COP) jeweils von einer eigenen Watchdog-Schaltung überwachen lassen. Das IBS-Masterboard (MA) wird immer durch eine Watchdog-Schaltung überwacht.

5.5.1 Watchdog des IBS-Masterboards

Wenn der Watchdog des IBS-Masterboards auslöst, erfolgt ein Reset des IBS-Systems, alle Ausgänge des IBS-Systems werden zurückgesetzt und das Si-

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

gnal *SysFail* des Masterboards ist gesetzt. Der Watchdog ist immer eingeschaltet und kann vom Anwender nicht beeinflußt werden.

5.5.2 Watchdog zur Überwachung des Hosts

Auf dem Grundboard der Anschaltbaugruppe befindet sich eine Watchdogschaltung, die Sie zur Überwachung Ihres PC-Programms („Absturz“ des PCs, „Aufhängen" eines Programms) einsetzen können. Der Watchdog versetzt bei seinem Auslösen das IBS-System in einen definierten Zustand (Zurücksetzen aller Ausgänge).

Der Watchdog beeinflußt nicht den Host, es wird z.B. kein Reset des Hosts ausgelöst!

Der Watchdog muß, wenn Sie ihn nutzen möchten, von Anwenderprogramm aus eingeschaltet werden. Er ist standardmäßig nicht eingeschaltet.

Sie schalten den Watchdog des Hosts durch den Aufruf der Funktion *EnableWatchdog ()* ein. Der Watchdog läßt sich nach dem Einschalten nicht mehr durch die Software ausschalten, und kann nur über das Ausschalten des Hosts oder einen Hardware-Reset deaktiviert werden.

Der Watchdog des Hosts ist fest auf eine Zeit von 146 msec eingestellt. Innerhalb dieser Zeit muß der Watchdog durch Aufruf der Funktion *TriggerWatchDog ()* im Anwenderprogramm getriggert werden, sonst löst er einen Reset des IBS-Systems aus.

Der Einsatz des Watchdogs zur Überwachung des Host ist unter Windows nicht sinnvoll. Es ist nicht gewährleistet, daß Ihr Programm innerhalb der vorgeschriebenen Zeit ein Triggern des Watchdogs leisten kann. Ein Beispiel: Sie verschieben mit der Maus den Rahmen eines Fensters. Solange Sie den Rahmen des Fensters mit der Maus festhalten, läuft Ihr Programm nicht weiter!

5.5.3 Watchdog des Coprozessorboards

Das Coprozessorboard kann ebenfalls durch eine eigene Watchdogschaltung überwacht werden. Die Funktion und Bedienung des Watchdogs ist, bis auf den beim Ansprechen ausgelösten Reset, identisch mit der zur Überwachung des PCs.

Sie schalten den Watchdog des Coprozessorboards durch den Aufruf der Funktion *EnableWatchDog* ein. Der Watchdog läßt sich nach dem Einschalten nicht mehr durch die Software ausschalten, und kann nur über einen Hardware-Reset deaktiviert werden.

Der Watchdog des Coprozessorboards ist fest auf eine Zeit von 125 Millisekunden eingestellt. Innerhalb dieser Zeit muß der Watchdog durch Aufruf der Funktion *TriggerWatchDog* im Anwenderprogramm getriggert werden, sonst löst er einen Reset des Coprozessorboards aus.

Der Status des Watchdogs ist über die Funktion *GetWatchDogState* auslesbar. Mit Hilfe dieser Funktion können Sie beim Starten des Anwenderprogramms feststellen, ob der Watchdog des COPs einen Reset des Coprozessorboards

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

ausgelöst hatte. Die Funktion *ClearWatchDog* setzt den Status des COP-Watchdogs zurück.

5.5.4 Das Signal *SysFail*

Für jeden an das MPM angeschlossene Teilnehmer ist innerhalb des MPMs ein eigener Bereich reserviert, der z.B. für Statusmeldungen oder Rückmeldungs-Signale genutzt wird. Eines dieser Statussignale ist das Signal *SysFail* (System Failure). Es wird bei einem Systemfehler des entsprechenden Teilnehmers gesetzt, z.B. wenn dessen Watchdog auslöst. Mit der Funktion *GetSysFailRegister* können Sie das Signal *SysFail* eines beliebigen MPM-Teilnehmers auslesen.

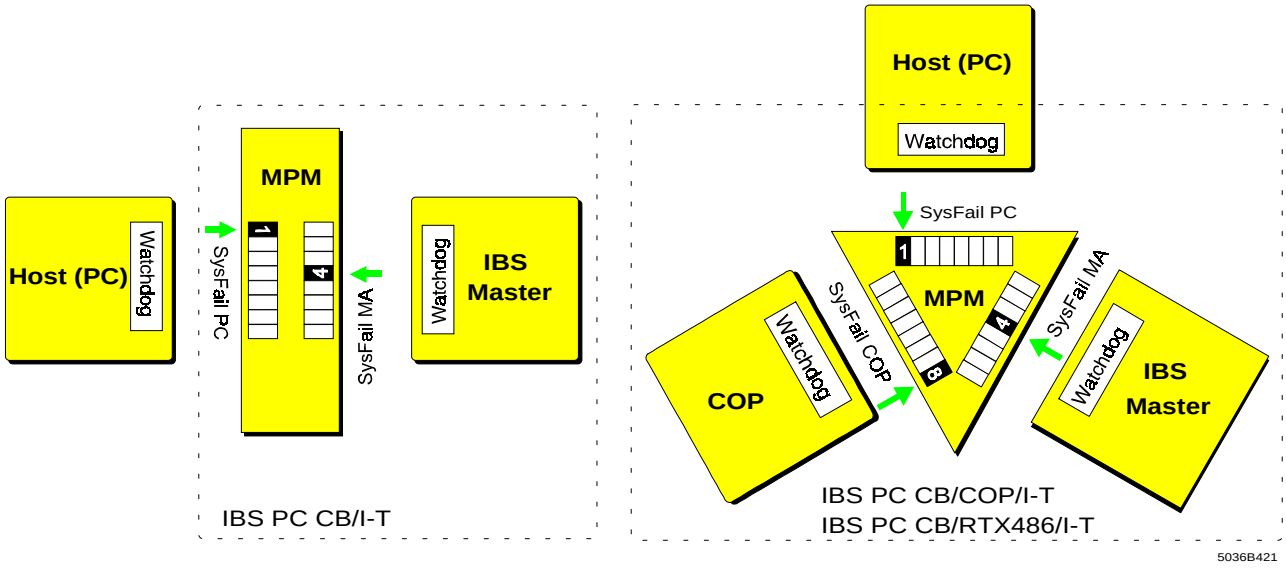

Bild 5-12: Das Signal *SysFail* im MPM

5.6 Download des Anwenderprogramms auf den COP

(nur für IBS PC CB/COP/I-T)

Zur komfortablen Handhabung des Coprozessorboards während der Programmentwicklung steht Ihnen die Entwicklungsumgebung *TDOS-PRO* (IBS PC COP SWT, Artikel-Nr. 27 52 12 3) zur Verfügung. Mit ihrer Hilfe können folgende Aktionen komfortabel realisiert werden:

- Download von Anwenderprogrammen
- Auswertung eines Statusberichts der Anschaltbaugruppe
- Reset der Anschaltbaugruppe
- Programmierung der Flash-EPROM
- Terminalfunktion für die Anschaltbaugruppe

Im Lieferumfang von *TDOS-PRO* ist neben dem Handbuch zu dieser Entwicklungsumgebung das Adapterkabel zum Herausführen der seriellen Schnittstelle des COPs (siehe Kapitel 3) enthalten.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel 6

INTERBUS-S-spezifische Programmierung

Dieses Kapitel zeigt Ihnen an einem Beispielaufbau die Programmierung von typischen Funktionalitäten des InterBus-S, wie

- Physikalische Adressierung und logische Adressierung
- Gruppendefinition sowie das Ab- und Wiederzuschalten von Gruppen

6 InterBus-S-spezifische Programmierung 6-3

6.1 Identifikation der angeschlossenen IBS-Teilnehmer 6-3
6.1.1 Physikalische Zählweise für Bussegmente und IBS-Teilnehmer 6-3
6.1.2 Beispielaufbau 6-4
6.1.3 Adressierungsarten des InterBus-S 6-6
6.2 Physikalische Adressierung von IBS-Teilnehmern 6-7
6.2.1 Adressen bei physikalischer Adressierung 6-8
6.2.1.1 Vergabe der Eingangsadressen durch die Anschaltbaugruppe . 6-8
6.2.1.2 Vergabe der Ausgangsadressen durch die Anschaltbaugruppe 6-10
6.2.2 Kommandofolge für den Anlauf bei physikalischer Adressierung 6-12
6.3 Logische Adressierung von IBS-Teilnehmern 6-13
6.3.1 Ermitteln der aktuell angeschlossenen Buskonfiguration . . . 6-14
6.3.2 Überprüfen der Buskonfiguration 6-14
6.3.3 Vergabe von logischen Bussegmentnummern 6-16
6.3.4 Vergabe der logischen Adressen durch den Programmierer . . 6-19
6.3.4.1 Vergabe der logischen Eingangsadressen 6-20
6.3.4.2 Vergabe der logischen Ausgangsadressen 6-24
6.3.4.3 Gültigkeit der Zuordnungslisten prüfen 6-27
6.3.5 Kommandofolge für den Anlauf bei logischer Adressierung . . 6-28
6.4 Gruppendefinition 6-29
6.4.1 Bilden von funktionalen Gruppen 6-29
6.4.2 Abschalten von Gruppen 6-32
6.4.3 Einschalten von Gruppen 6-33
6.4.4 Behandlung der Gruppen für den Fehlerfall definieren 6-33

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6 InterBus-S-spezifische Programmierung

6.1 Identifikation der angeschlossenen IBS-Teilnehmer

Die Identifikation der angeschlossenen IBS-Teilnehmer erfolgt an Hand dreier Merkmale:

- physikalische Lage (Plazierung) des IBS-Teilnehmers im Busaufbau
- Längen-Code des IBS-Teilnehmers (beschreibt den Adreßraumbedarf im Host)
- Identifikations-Code (kurz ID-Code) des IBS-Teilnehmers

Eine Tabelle (Teilnehmerliste) beschreibt den Aufbau des IBS-Systems.

Längen-Code	ID-Code	für Teilnehmer Nr. 1
Längen-Code	ID-Code	für Teilnehmer Nr. 2
Längen-Code	ID-Code	für Teilnehmer Nr. 3
...	...	
Längen-Code	ID-Code	für Teilnehmer Nr. n

Bit | 15 ..... 8 | 7 ..... 0 |

Bild 6-1: Prinzipieller Aufbau einer Teilnehmerliste

Legende:

Längen-Code: Der Längen-Code beschreibt den Adreßraumbedarf des IBS-Teilnehmers im Host.

ID-Code: Identifikations-Code (kurz ID-Code) des IBS-Teilnehmers

In der Ident-Liste wird die Konfiguration des IBS-Systems eingetragen. Dazu müssen die Längen-Codes und die ID-Codes der einzelnen IBS-Teilnehmer bekannt sein (siehe IBS-Teilnehmerübersicht).

6.1.1 Physikalische Zählweise für Bussegmente und IBS-Teilnehmer

Ein Bussegment besteht aus einem Fernbusteilnehmer und dem ankommenden Fernbuskabel. Falls dieser Fernbusteilnehmer eine Busklemme ist, gehören auch die Teilnehmer eines von Ihm abzweigenden Peripheriebusses zu seinem Bussegment. Bussegment 0 (kurz BS 0) ist das Bussegment, dessen Busklemme über den Fernbus direkt mit der Anschaltbaugruppe im Host verbunden ist. Bussegment 1 ist über den Fernbus an die Busklemme des Bussegmentes 0 gekoppelt usw.

Busklemmen für den Installationsfernbus und Installationsfernbus-Teilnehmer besitzen eine eigene Bussegmentnummer
(siehe Bild 6-2, Bussegmente 2 bis 5)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.1.2 Beispielaufbau

Auf den folgenden Seiten sind in einigen Beispielen die unterschiedlichen Adressierungsarten beschrieben. Der dabei dargestellte Busaufbau ist für alle Beispiele gleich und besteht aus folgender Zusammenstellung:

Tabelle 6-1: IBS-Teilnehmer des Beispielaufbaus

Modul-Nr.	Bussegment	Modul	Längen-Code*	ID-Code*
1	BS 0	IBS ST 24 BK-T	00_{hex}	08_{hex}
2	BS 0	IB ST 24 DI 16/4	01_{hex}	BE_{hex}
3	BS 0	IB ST 24 AO 4/SF	04_{hex}	$7D_{hex}$
4	BS 0	IB ST 24 DO 16/3	01_{hex}	BD_{hex}
5	BS 0	IB ST 24 PT 100 4/4	04_{hex}	$7B_{hex}$
6	BS 1	IBS IP DIO 1/24-P	01_{hex}	07_{hex}
7	BS 2	IBS IP CBK 1/24-F	00_{hex}	$0C_{hex}$
8	BS 3	IBS IP CDI 1/24-F	81_{hex}	$0A_{hex}$
9	BS 4	IBS IP CDO 1/24-F	81_{hex}	09_{hex}
10	BS 5	IBS IP CDI 1/24-F	81_{hex}	$0A_{hex}$
11	BS 6	IBS 24 BK I/O-T	01_{hex}	$0B_{hex}$
12	BS 6	IBS 24 DI/32	02_{hex}	$8E_{hex}$
13	BS 6	IBS AI 3	04_{hex}	47_{hex}

* Längen- und ID-Codes für alle IBS-Teilnehmer von Phoenix Contact finden Sie in der IBS-Teilnehmerübersicht.

onlinecomponents.com

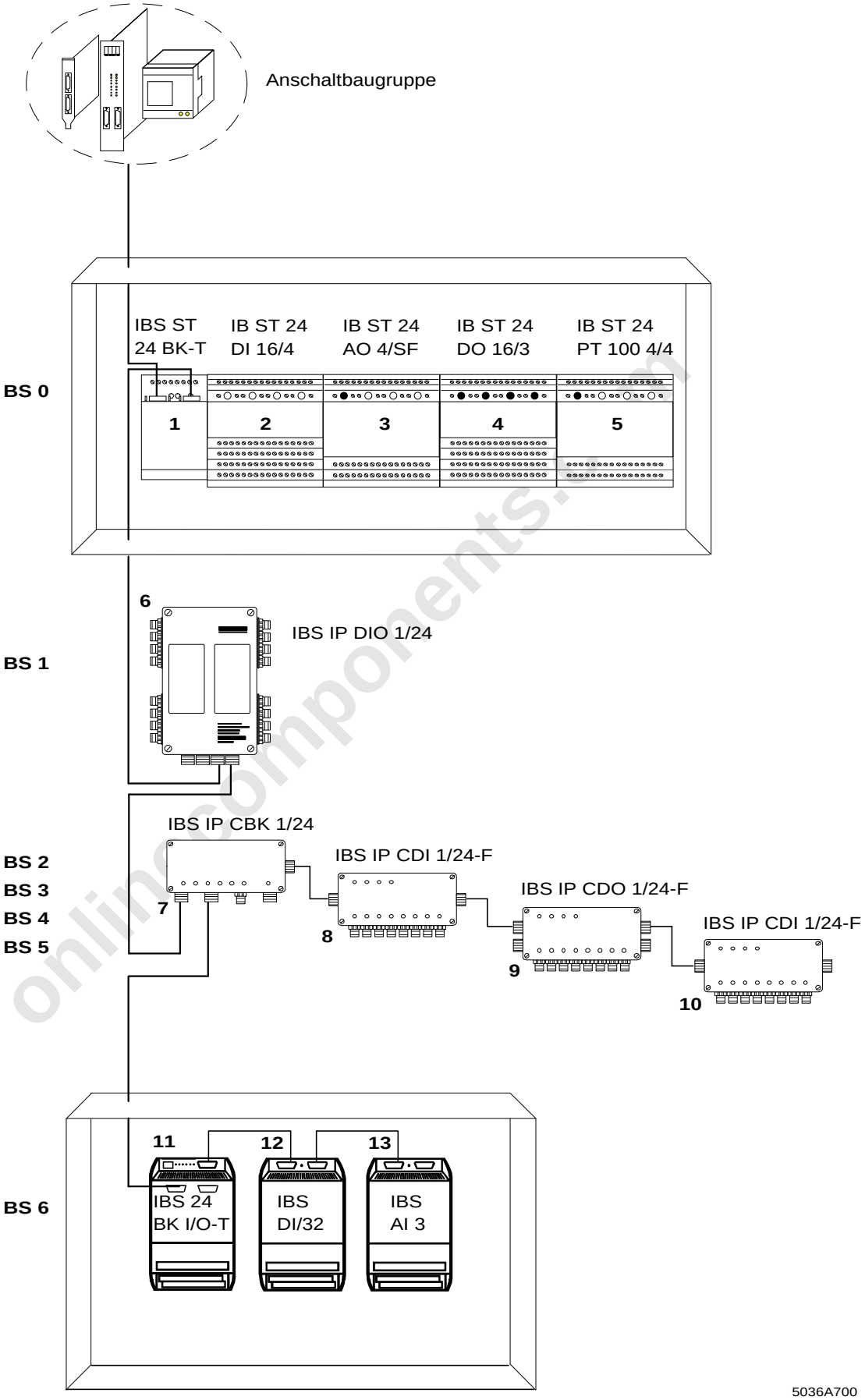

Bild 6-2: Busaufbau für die Beispiele zur Adressierung

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.1.3 Adressierungsarten des InterBus-S

Es sind zwei Adressierungsarten möglich.

Physikalische Adressierung

In dieser Betriebsart legt die Anschaltbaugruppe die Adressen der IBS-Teilnehmer automatisch an Hand der physikalischen Reihenfolge der IBS-Teilnehmer im Busaufbau fest.

Logische Adressierung

In dieser Betriebsart vergibt der Programmierer die Adressen der IBS-Teilnehmer nach seiner Wahl. Die Anschaltbaugruppe adressiert dann die IBS-Teilnehmer auf die im Anwenderprogramm festgelegte Weise.

Die logische Adressierung ist sinnvoll, um

- die Speicheraufteilung zu optimieren;
- bei Erweiterung ihrer Anlage Adreßverschiebungen auszuschließen;
- eine Änderung der Konfiguration ohne komplette Umstellung der Adressierung zu ermöglichen;
- die Übersichtlichkeit der Anlage zu verbessern. Sie können die Bussegmente eines Anlagenteils (z.B. eines Schaltschrankes, einer Maschine) einer definierten Gruppe zuweisen.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.2 Physikalische Adressierung von IBS-Teilnehmern

Bei der physikalischen Adressierung adressiert die Anschaltbaugruppe die IBS-Teilnehmer automatisch an Hand der physikalischen Reihenfolge im Busaufbau. Unter Firmware 3.x reserviert die Anschaltbaugruppe für jeden IBS-Teilnehmer, der Prozeßdaten liefert oder verarbeitet, mindestens 1 Wort (16 Bit) im Multi-Port-Memory.

Bei physikalischer Adressierung adressiert die Anschaltbaugruppe, nach einer Erweiterung oder Änderung des Busaufbaus, automatisch die IBS-Teilnehmer entsprechend der geänderten Anordnung neu. Treffen Sie im Anwenderprogramm entsprechende Vorkehrungen (z.B. Abfrage der Konfiguration), da es hierbei zu Adreßverschiebungen kommen kann!

Im Gegensatz zur physikalischen Adressierung können Sie bei logischer Adressierung in ihrem Programm die Adressen der IBS-Teilnehmer unabhängig von deren tatsächlichen physikalischen Reihenfolge im Bussystem vergeben (siehe Kapitel 7.2.2).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.2.1 Adressen bei physikalischer Adressierung

6.2.1.1 Vergabe der Eingangsadressen durch die Anschaltbaugruppe

Tabelle 6-2: Zuordnung der Eingangsworte zu den IBS-Teilnehmern

Modul-Nr.	Modul	Eingangswort (EW)
1	IBS ST 24 BK-T	—
2	IB ST 24 DI 16/4	EW 0
3	IB ST 24 AO 4/SF	—
4	IB ST 24 DO 16/3	—
5	IB ST 24 PT 100 4/4	EW 1, EW 2 , EW 3, EW 4
6	IBS IP DIO	EW 5
7	IBS IP CBK	—
8	IBS IP CDI	EW 6
9	IBS IP CDO	—
10	IBS IP CDI	EW 7
11	IBS 24 BK I/O-T	EW 8
12	IBS 24 DI/32	EW 9, EW 10
13	IBS AI 3	EW 11, EW 12, EW 13, EW 14

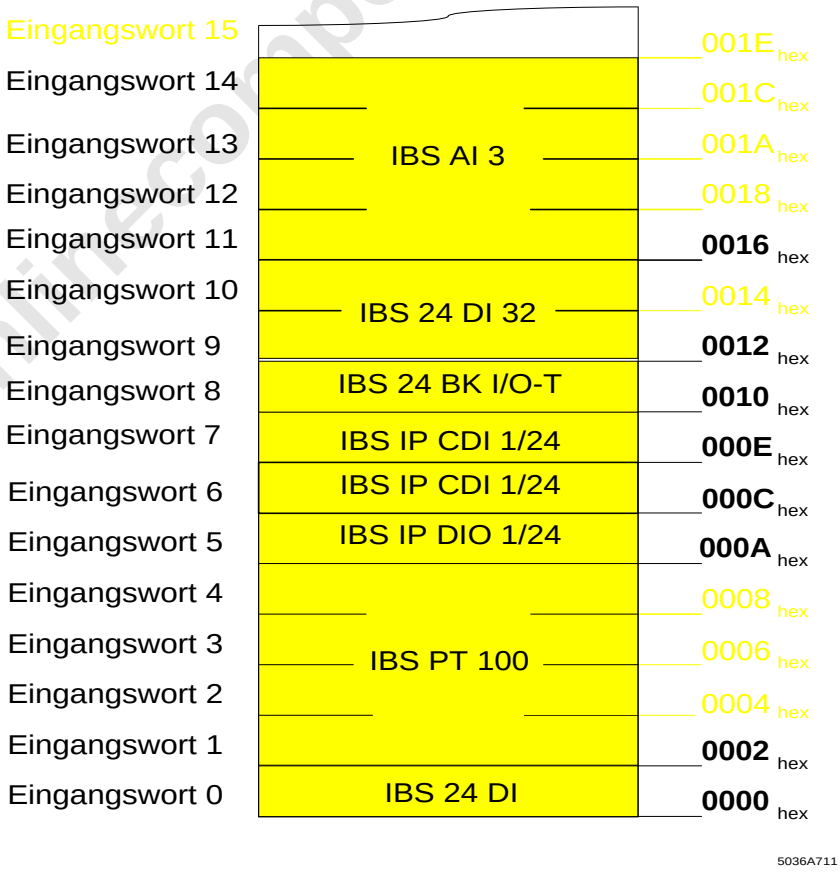

Bild 6-3: Eingangsadressen im Speicherabbild (IN-Buffer)
Die Startadressen der Module sind fett hervorgehoben.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

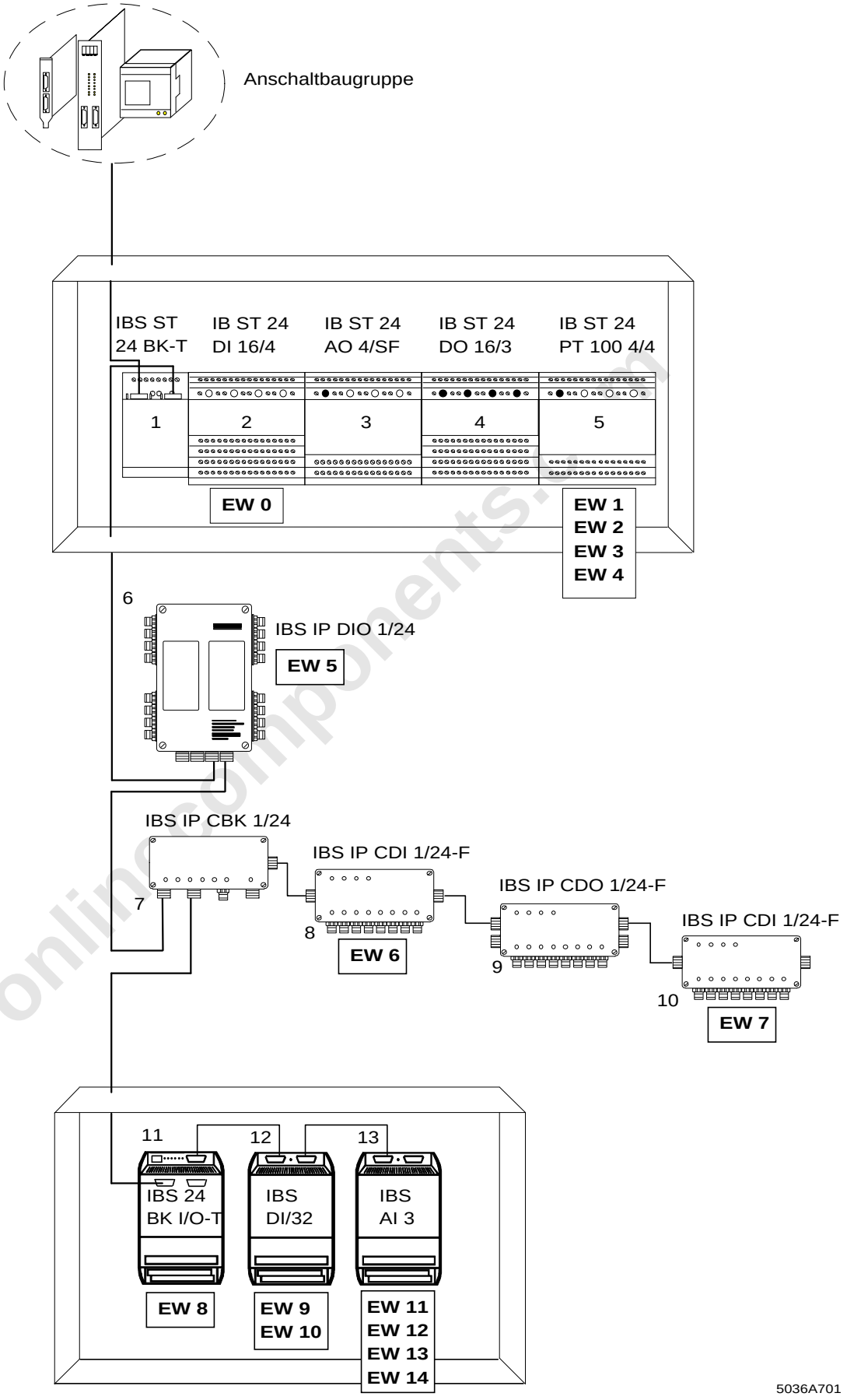

Bild 6-4: Eingangsadressen bei physikalischer Adressierung

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.2.1.2 Vergabe der Ausgangsadressen durch die Anschaltbaugruppe

Tabelle 6-3: Zuordnung der Ausgangsworte zu den IBS-Teilnehmern

Modul-Nr.	Modul	Ausgangswort (AW)
1	IBS ST 24 BK-T	-
2	IB ST 24 DI 16/4	-
3	IB ST 24 AO 4/SF	AW 0, AW 1, AW 2, AW 3
4	IB ST 24 DO 16/3	AW 4
5	IB ST 24 PT 100 4/4	-
6	IBS IP DIO	AW 5
7	IBS IP CBK	-
8	IBS IP CDI	-
9	IBS IP CDO	AW 6
10	IBS IP CDI	-
11	IBS 24 BK I/O-T	AW 7
12	IBS 24 DI/32	-
13	IBS AI 3	AW 8, AW 9, AW 10, AW 11

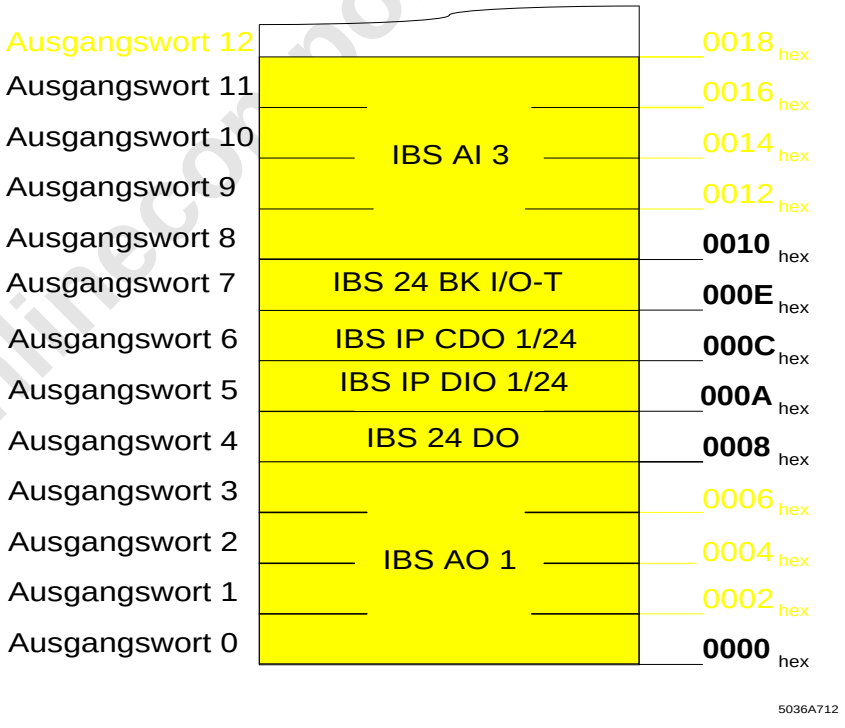

Bild 6-5: Ausgangsadressen im Speicherabbild (OUT-Buffer)
Die Startadressen der Module sind fett hervorgehoben.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

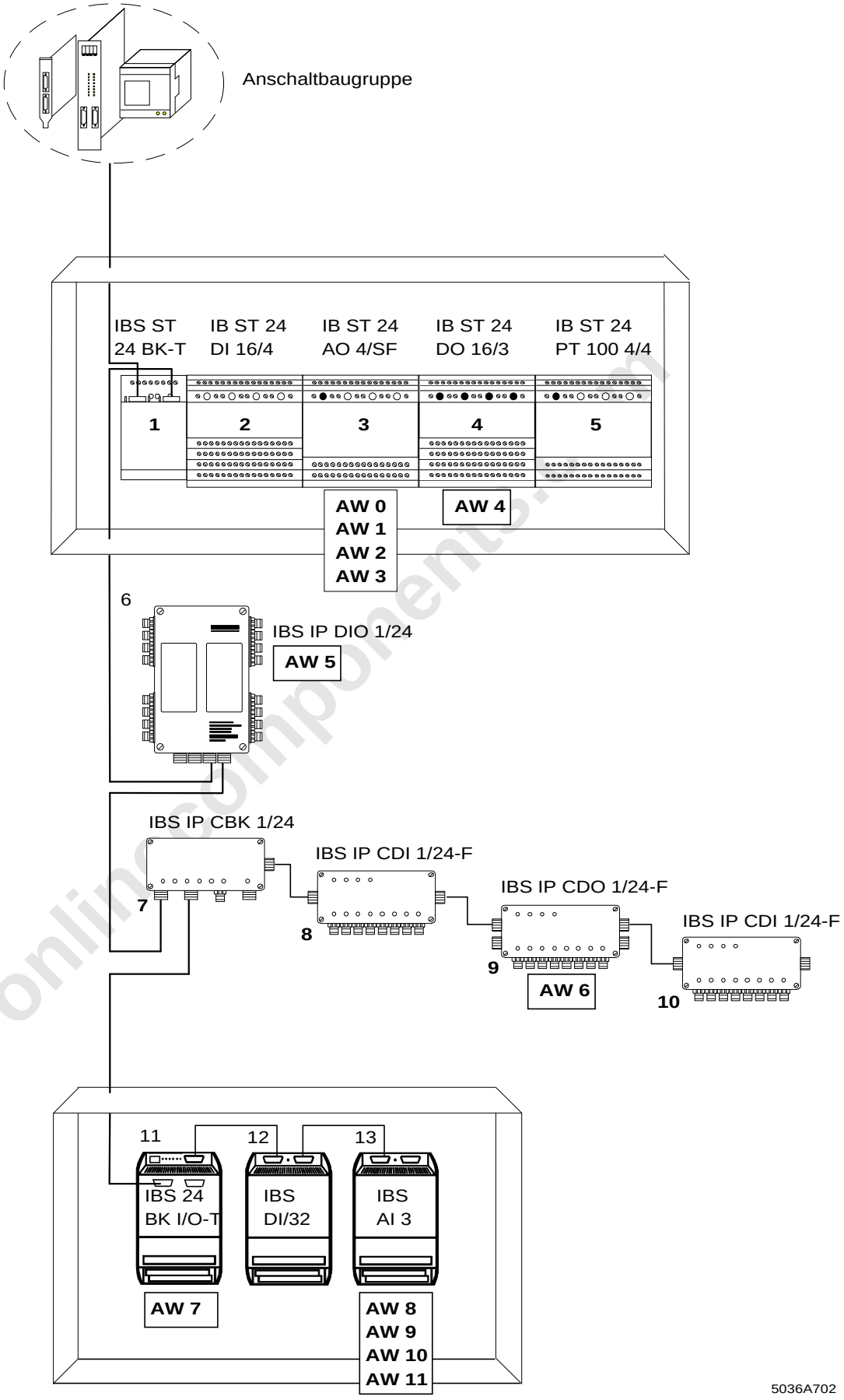

Bild 6-6: Ausgangsadressen bei physikalischer Adressierung

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.2.2 Kommandofolge für den Anlauf bei physikalischer Adressierung

Tabelle 6-4: Kommandofolge für den Anlauf bei physikalischer Adressierung

Aktion	Kommando	Code
Ausgänge in sicheren Zustand setzen, Datenübertragung auf dem Bus stoppen	Alarmstop_Request	$004A_{hex}$
Register für Diagnosebits löschen	Clear_Display_Request	$004E_{hex}$
Bussystem konfigurieren	Configure_BUS_Request	0023_{hex}
Konfiguration überprüfen	Check_Physical_Configuration_Request	0058_{hex}
Starten der Datenzyklen	Start_BUS_Cycle_Request	0001_{hex}

Kapitel 8 beschreibt die Kommandos detailliert.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.3 Logische Adressierung von IBS-Teilnehmern

Im Gegensatz zur physikalischen Adressierung können Sie bei logischer Adressierung die Adressen der IBS-Teilnehmer unabhängig von deren tatsächlicher physikalischen Reihenfolge im Bussystem vergeben. Durch Eintrag in der physikalischen Reihenfolge in Zuordnungslisten vergeben Sie die gewünschten Adressen.

Sie benötigen für die logische Adressierung folgende Zuordnungslisten:
- Liste mit Längen- und ID-Codes von allen Teilnehmern Ihres Busaufbaus
- Liste der logischen Bussegmente
- Liste der logischen Eingangsadressen
- Liste der logischen Ausgangsadressen

Die Listen werden zur Anschaltbaugruppe übertragen, dort geprüft und im RAM gespeichert. Im Anlauf adressiert die Anschaltbaugruppe dann die IBS-Teilnehmer anhand dieser Listen.

Die logische Adressierung ermöglicht somit die wahlfreie Zuordnung der E/A-Daten aus dem InterBus-S-System zu den Speicheradressen im Multi-Port-Memory und somit im Host. Die freie Wahl der Adressen erleichtert das Ab- oder Zuschalten von Anlagenteilen im Busaufbau, da Sie so beim Einfügen oder Entfernen von IBS-Teilnehmern nur die Adreßliste, und nicht sämtliche Adressen in Ihrem Anwenderprogramm ändern müssen.

Außerdem sind weitere, anlagenspezifische Zuordnungen möglich. Sie können z.B. die Bussegmentnummer dem jeweiligen Anlagenteil (z.B. einer Schaltschranknummer) fest zuordnen oder einzelne Bussegmente zu Gruppen zusammenzufassen.

Vorgehensweise

1. Klären Sie folgenden Fragen:
   - In welcher Reihenfolge sollen die IBS-Teilnehmer logisch adressiert werden?
   - Welche IBS-Teilnehmer sollen welche logische Adressen erhalten?
   - Welche IBS-Teilnehmer sollen in einer Gruppe zusammengefaßt werden?
2. Erstellen Sie entsprechend den Vorüberlegungen die logische IN-Adreßliste.
3. Erstellen Sie entsprechend den Vorüberlegungen die logische OUT-Adreßliste.
4. Numerieren Sie gegebenenfalls die Bussegmente und führen Sie eine Gruppendefinition durch.
5. Die Anschaltbaugruppe überprüft die Zuordnungslisten bei der Übernahme (Kommando *Implement_All_Logical_Address_Maps_Request* (0040_{hex})) automatisch auf Konformität und Plausibilität.

Wenn die Anschaltbaugruppe die Zuordnungslisten fehlerfrei gespeichert hat, kann es den InterBus-S mit der logischen Adressierung betreiben.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.3.1 Ermitteln der aktuell angeschlossenen Buskonfiguration

Das IBS-Kommando *Configure_BUS_Request* (0023_{hex}) veranlaßt die Anschaltbaugruppe dazu, die aktuell angeschlossene Buskonfiguration zu ermitteln und im RAM der Anschaltbaugruppe zu speichern. Dabei werden für alle angeschlossenen IBS-Teilnehmer die Längen- und ID-Codes eingelesen.

Das Kommando *Configure_BUS_Request* (0023_{hex}) wird im Kapitel 8, *Kommandos für das IBS-Masterboard* detailliert beschrieben.

Alle zuvor gespeicherten Listen (Adressierung, Gruppendefinitionen, Prozeßdatenverknüpfungen und Event-Definitionen) werden gelöscht.

6.3.2 Überprüfen der Buskonfiguration

Das IBS-Kommando *Check_Physical_Configuration_Request* (0058_{hex}) überträgt eine für den Betrieb gewünschte Buskonfiguration in Form von Längen- und ID-Codes zur Anschaltbaugruppe. Die Anschaltbaugruppe vergleicht die übertragene Konfiguration automatisch mit der durch das Kommando *Configure_BUS_Request* (0023_{hex}) im RAM gespeicherten Konfiguration.

Das Kommando *Check_Physical_Configuration_Request* (0058_{hex}) wird im Kapitel 8, *Kommandos für das IBS-Masterboard* detalliert beschrieben.

Stimmen die übertragene und die gespeicherte Konfiguration nicht überein, so generiert die Anschaltbaugruppe eine entsprechende Fehlermeldung (CTRL ERR). Mit dem Kommando *Send_Log_Address_Error_Request* ($005F_{hex}$) können Sie dann eine Meldung mit detaillierten Informationen zu Fehlerart und Fehlerort abrufen.

Wort 1	Code		
Wort 2	Parameterzähler (n)		
Wort 3	Längen-Code	ID-Code	für Teilnehmer Nr. 1
Wort 4	Längen-Code	ID-Code	für Teilnehmer Nr. 2
Wort 5	Längen-Code	ID-Code	für Teilnehmer Nr. 3
	...	...	
Wort n+2	Längen-Code	ID-Code	für Teilnehmer Nr. n
Bit	15 8	7 0	

Bild 6-7: Aufbau des Kommandos *Check_Physical_Configuration_Request*

Legende:

Code:	Kommando-Code (hier 0058_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).
Längen-Code:	Der Längen-Code beschreibt den Adreßraumbedarf des IBS-Teilnehmers im Host.
ID-Code:	Identifikations-Code des IBS-Teilnehmers.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Verwenden Sie zusätzlich das Kommando *Receive_Local_Bus_Code_Map_Request*, falls ihr Busaufbau Installationsfernbus-Teilnehmer enthält.

Buffer in der Programmiersprache "C":

```
/*****************************************************************/
/* "Check_Physical_Configuration_Request"                        */
/*  as a list for the bus configuration                          */
/*****************************************************************/
USIGN16 chk_phy_cnf[] = //Data field of the unsigned-integer type
             {0x0058,  //Command Code
              0x000D,  //Parameter Counter
              0x0008,  //IBS ST 24 BK-T
              0x01BE,  //IB ST 24 DI 16/4
              0x047D,  //IB ST 24 AO 4/SF
              0x01BD,  //IB ST 24 DO 16/3
              0x047B,  //IBS PT 100
              0x0107,  //IBS IP DIO 1/24
              0x000C,  //IBS IP CBK 1/24
              0x810A,  //IBS IP CDI 1/24
              0x8109,  //IBS IP CDO 1/24
              0x810A,  //IBS IP CDI 1/24
              0x010B,  //IBS 24 BK I/O-T
              0x028E,  //IBS 24 DI/32
              0x0447}; //IBS AI 3
```

Buffer in der Programmiersprache "Pascal":

```
{*****************************************************************}
{* "Check_Physical_Configuration_Request"                        *}
{*  as a list for the bus configuration                          *}
{*****************************************************************}

const chk_phy_cnf : array[1..15] of word =
                    {Data field of the unsigned-integer type}
             ($0058,  {Command Code}
              $000D,  {Parameter Counter}
              $0008,  {IBS ST 24 BK-T}
              $01BE,  {IB ST 24 DI 16/4}
              $047D,  {IB ST 24 AO 4/SF}
              $01BD,  {IB ST 24 DO 16/3}
              $047B,  {IBS PT 100}
              $0107,  {IBS IP DIO 1/24}
              $000C,  {IBS IP CBK 1/24}
              $810A,  {IBS IP CDI 1/24}
              $8109,  {IBS IP CDO 1/24}
              $810A,  {IBS IP CDI 1/24}
              $010B,  {IBS 24 BK I/O-T}
              $028E,  {IBS 24 DI/32}
              $0447); {IBS AI 3}
```

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.3.3 Vergabe von logischen Bussegmentnummern

Das IBS-Kommando *Receive_Local_Bus_Code_Map_Request* (0069_{hex}) nimmt die wahlfreie Zuordnung der Bussegmentnummer zu den InterBus-S-Teilnehmern mit Busklemmenfunktionalität (Fernbus- und Installationsfernbusteilnehmer) vor. Dies ist sinnvoll, um bei Erweiterungen der Anlage nicht die Numerierung der Bussegmente in der bestehenden Anlage ändern zu müssen.

Wort 1	Code			
Wort 2	Parameterzähler (n)			
Wort 3	FB-Ebene	0_{hex}	Bussegmentnummer	Teilnehmer Nr. 1
Wort 4	FB-Ebene	0_{hex}	Bussegmentnummer	Teilnehmer Nr. 2
	...	...	...	
Wort n+2	FB-Ebene	0_{hex}	Bussegmentnummer	Teilnehmer Nr. n

Bit	15 12	11 8	7 4	3 0

Bild 6-8: Aufbau des Kommandos *Receive_Local_Bus_Code_Map_Request*

Legende:

Code: Kommando-Code (hier 0069_{hex})

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Bussegmente).

FB-Ebene: Tragen Sie hier die FB-Ebene ein:
- Hauptstrang: FB-Ebene = 0_{hex} (0000_{bin})
- Fernbus-Stich (z.B. Abzweig von IBS IP CBK): FB-Ebene = 1_{hex} (0001_{bin})

Bussegmentnummer: Tragen Sie hier in der Reihenfolge der physikalischen Anordnung die gewünschten logischen Bussegmentnummern ein.

Die Liste wird erst nach erfolgreicher Ausführung des Kommandos *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) gültig.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

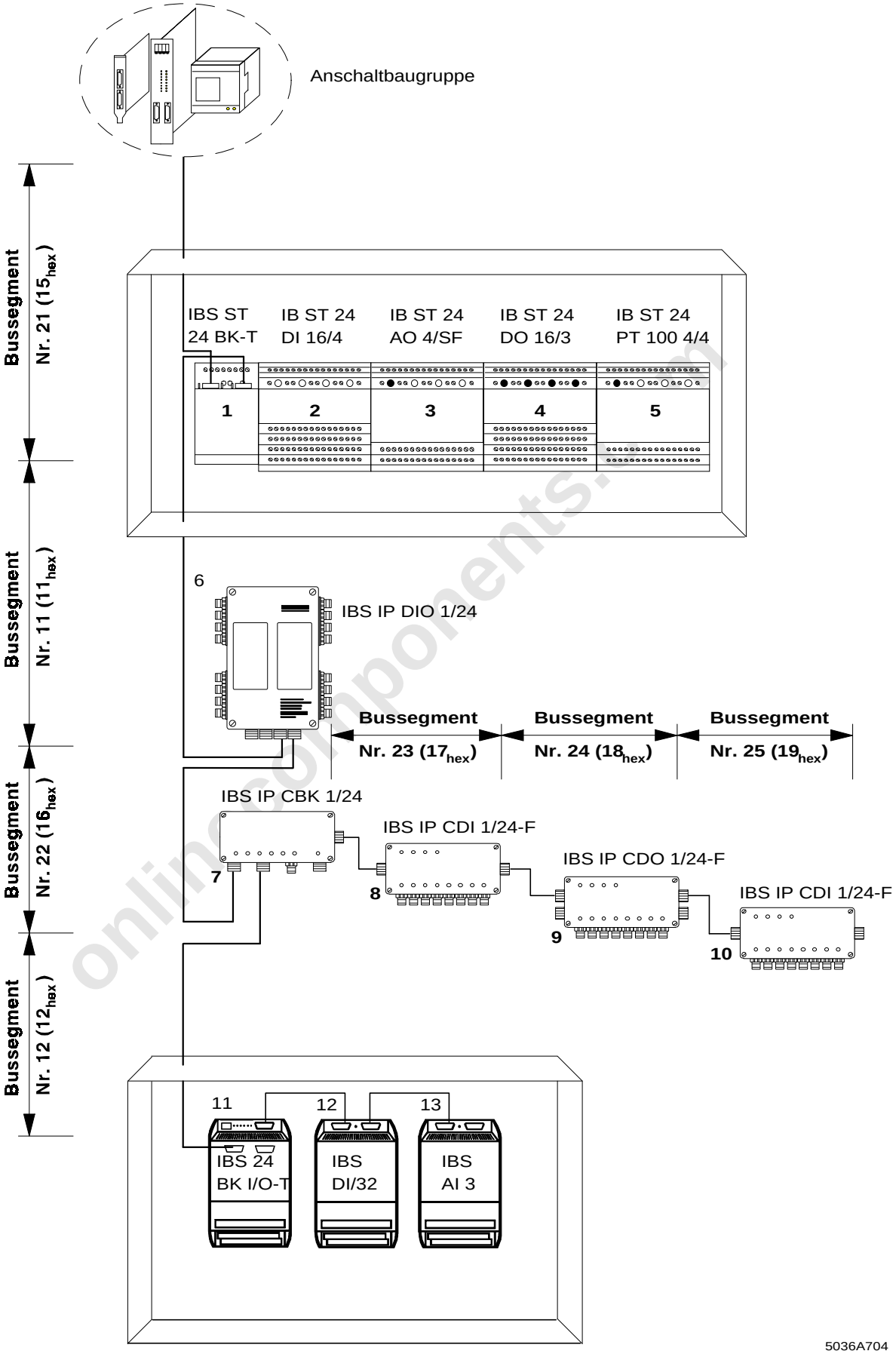

Bild 6-9: Numerierung der Bussegmente bei logischer Adressierung

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Buffer in der Programmiersprache "C":

```
/*********************************************************************/
/* "Receive_Local_Bus_Code_Map_Request"                              */
/*  to implement logical bus segment numbers                         */
/*********************************************************************/
USIGN16 log_lb_adr_map[] =
              {0x0069,  //Command Code
               0x000D,  //Parameter Counter
               0x0015,  //IBS ST 24 BK-T
               0x0000,  //IB ST 24 DI 16/4
               0x0000,  //IB ST 24 AO 4/SF
               0x0000,  //IB ST 24 DO 16/3
               0x0000,  //IBS PT 100
               0x000B,  //IBS IP DIO 1/24
               0x0016,  //IBS IP CBK 1/24
               0x1017,  //IBS IP CDI 1/24
               0x1018,  //IBS IP CDO 1/24
               0x1019,  //IBS IP CDI 1/24
               0x000C,  //IBS 24 BK I/O-T
               0x0000,  //IBS 24 DI/32
               0x0000}; //IBS AI 3
```

Buffer in der Programmiersprache "Pascal":

```
{*********************************************************************}
{* "Receive_Local_Bus_Code_Map_Request"                              *}
{*  to implement logical bus segment numbers                         *}
{*********************************************************************}
const log_lb_adr_map :array[1..15] of word =
              ($0069,  {Command Code}
               $000D,  {Parameter Counter}
               $0015,  {IBS ST 24 BK-T}
               $0000,  {IB ST 24 DI 16/4}
               $0000,  {IB ST 24 AO 4/SF}
               $0000,  {IB ST 24 DO 16/3}
               $0000,  {IBS PT 100}
               $000B,  {IBS IP DIO 1/24}
               $0016,  {IBS IP CBK 1/24}
               $1017,  {IBS IP CDI 1/24}
               $1018,  {IBS IP CDO 1/24}
               $1019,  {IBS IP CDI 1/24}
               $000C,  {IBS 24 BK I/O-T}
               $0000,  {IBS 24 DI/32}
               $0000); {IBS AI 3}
```

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.3.4 Vergabe der logischen Adressen durch den Programmierer

Numerierung von Worten bei der Vergabe von logischen Adreßlisten

InterBus-S numeriert Worte in Anlehnung an die bei **s**peicher**p**rogrammierbaren **S**teuerungen (SPS) übliche Zählweise. Da aufeinanderfolgende Worte auf geradzahligen Byteadressen beginnen, werden Sie nach diesen geraden Byteadressen numeriert. So bekommt z.B. das Wort welches aus den Bytes 6 und 7 besteht, die Nummer 6.

Tabelle 6-5: Unterschiedliche Numerierung von Worten bei Speicherprogrammierbaren Steuerungen (SPS) und PCs

SPS		PC	
Wort	**Byte**	**Wort**	**Byte**
Wort n	Byte n+1	Wort n/2	Byte n+1
	Byte n		Byte n
...	...	...	...
	...		...
Wort 6	Byte 7	Wort 3	Byte 7
	Byte 6		Byte 6
Wort 4	Byte 5	Wort 2	Byte 5
	Byte 4		Byte 4
Wort 2	Byte 3	Wort 1	Byte 3
	Byte 2		Byte 2
Wort 0	Byte 1	Wort 0	Byte 1
	Byte 0		Byte 0

Verwenden deshalb Sie bei der Erstellung von Adreßlisten mit den Kommandos
- *Receive_Logical_IN_Address_Map_Request* ($003A_{hex}$) und
- *Receive_Logical_OUT_Address_Map_Request* ($003B_{hex}$)
  für die Startadressen von IBS-Teilnehmern mit 16 Bit (oder mehr) Adreßraum nur gerade Adressen.

IBS-Teilnehmer mit 8 Bit Adreßraum (Längen-Code 81_{hex}) können Sie wahlweise auf gerade oder ungerade Startadressen legen. Auf diese Weise können Sie, bei logischer Adressierung, zwei 8 Bit-Module in einem Wort zusammenfassen.

Die InterBus-typische Zählweise für Worte (Tabelle 6-5, links) müssen Sie nur bei der Erstellung der Zuordnungslisten (logische Adressierung, Prozeßdatenverknüpfung, Event-Definition) berücksichtigen. Die Treibersoftware (Funktionen *DDI_DTI_ReadData* und *DDI_DTI_WriteData*) stellt die Daten in der für Hochsprachen üblichen Wortzählweise (Tabelle 6-5, rechts) in den Buffer (siehe z.B. Bild 6-12).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.3.4.1 Vergabe der logischen Eingangsadressen

Das IBS-Kommando *Receive_Logical_IN_Address_Map_Request* ($003A_{hex}$) überträgt die Liste der logischen Eingangsadressen zur Anschaltbaugruppe.

Wort 1	Code			
Wort 2	Parameterzähler (n)			
Wort 3	DC	0000_{bin}	Eingangsadresse	Teilnehmer Nr. 1
Wort 4	DC	0000_{bin}	Eingangsadresse	Teilnehmer Nr. 2
	...	...	...	
Wort n+2	DC	0000_{bin}	Eingangsadresse	Teilnehmer Nr. n

Bit	15 14	13 9	9 0

Bild 6-10: Aufbau des Kommandos *Receive_Logical_IN_Address_Map_Req.*

Legende:

Code: Kommando-Code (hier $003A_{hex}$)

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).

Eingangsadresse: Tragen Sie hier (10 Bit) für alle Module mit Prozeßeingangsdaten die gewünschte Adresse (als Byteadresse) ein. Bei Modulen mit mehr als einem Prozeßeingangswort (z.B. mit 32 Bit), werden automatisch die nächsthöheren Adressen mit belegt.

DC: Tragen Sie hier die Datenkonsistenz für den Prozeßdatenzugriff ein. Die Datenkonsistenz garantiert, daß die angegebene Datenbreite aus einem IBS-Zyklus stammt. Der Standardwert für E/A-Module ist 16 Bit (00_{bin}). Für IBS-Teilnehmer, welche die zusammenhängende Übertragung größerer Datenbreiten verlangen, müssen Sie die Datenkonsistenz erhöhen. Dies sind z.B. Encoder, Bedienterminals oder Analogmodule mit mehr als 16 Bit Auflösung. Für die Datenkonsistenz sind folgende Werte zulässig:

- 00_{bin} 16 Bit (Standard)
- 01_{bin} 32 Bit (z.B. Encoder, Bedienterminals)
- 10_{bin} 8 Bit (nur für Module mit Längencode 81_{hex}, auf die Bytezugriffe erfolgen sollen!)
- 11_{bin} 48 Bit (z.B. Encoder, Bedienterminals)

Tragen Sie für alle Module ohne Prozeßeingangsdaten (z.B. reine Busklemmen oder reine Ausgangsmodule) den Eintrag 0000_{hex} ein.

Legen Sie die Startadressen von IBS-Teilnehmern mit 16 Bit (oder mehr) Adreßraum nur auf gerade Adressen.

Die Liste wird erst nach erfolgreicher Ausführung des Kommandos *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) gültig.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Buffer in der Programmiersprache "C":

```
/*****************************************************************************/
/* "Receive_Logical_IN_Address_Map_Request"                                  */
/*  to implement all in addresses                                            */
/*****************************************************************************/
USIGN16 in_adr_map[] =
              {0x003A,  //Command Code
               0x000D,  //Parameter Counter
               0x0000,  //IBS ST 24 BK-T
               0x0000,  //IB ST 24 DI 16/4
               0x0000,  //IB ST 24 AO 4/SF
               0x0000,  //IB ST 24 DO 16/3
               0x001C,  //IBS PT 100
               0x000A,  //IP DIO 1/24
               0x0000,  //IP CBK 1/24
               0x0002,  //IP CDI 1/24
               0x0000,  //IP CDO 1/24
               0x0004,  //IP CDI 1/24
               0x000C,  //IBS 24 BK I/O-T
               0x0006,  //IBS 24 DI/32
               0x0014}; //IBS AI 3
```

Buffer in der Programmiersprache "Pascal":

```
{*****************************************************************************}
{* "Receive_Logical_IN_Address_Map_Request"                                  *}
{*  to implement all in addresses                                            *}
{*****************************************************************************}
const in_adr_map : array [1..15] of word =
              ($003A,  {Command Code}
               $000D,  {Parameter Counter}
               $0000,  {IBS ST 24 BK-T}
               $0000,  {IB ST 24 DI 16/4}
               $0000,  {IB ST 24 AO 4/SF}
               $0000,  {IB ST 24 DO 16/3}
               $001C,  {IBS PT 100}
               $000A,  {IP DIO 1/24}
               $0000,  {IP CBK 1/24}
               $0002,  {IP CDI 1/24}
               $0000,  {IP CDO 1/24}
               $0004,  {IP CDI 1/24}
               $000C,  {IBS 24 BK I/O-T}
               $0006,  {IBS 24 DI/32}
               $0014); {IBS AI 3}
```

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

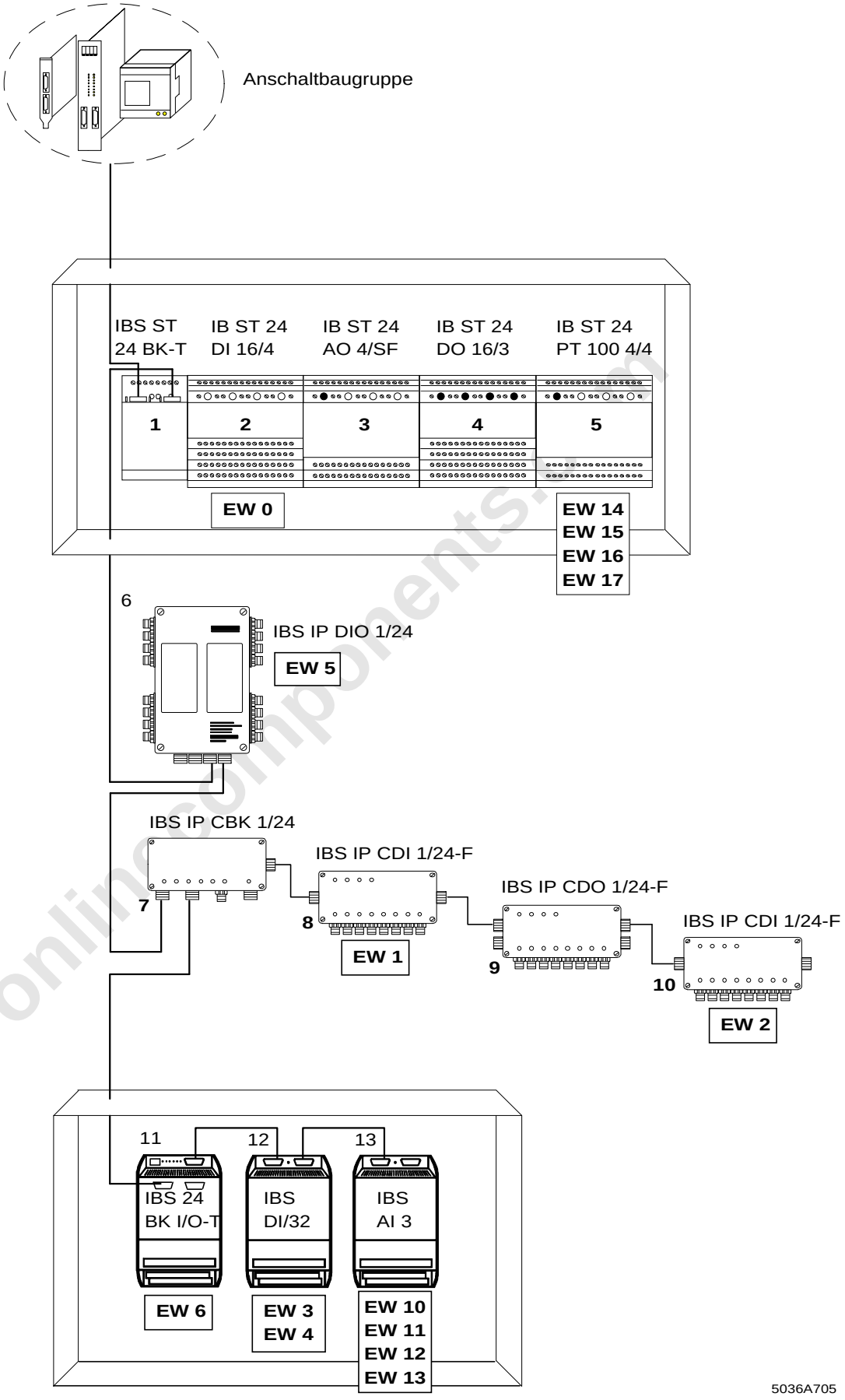

Bild 6-11: Eingangsadressen bei logischer Adressierung

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

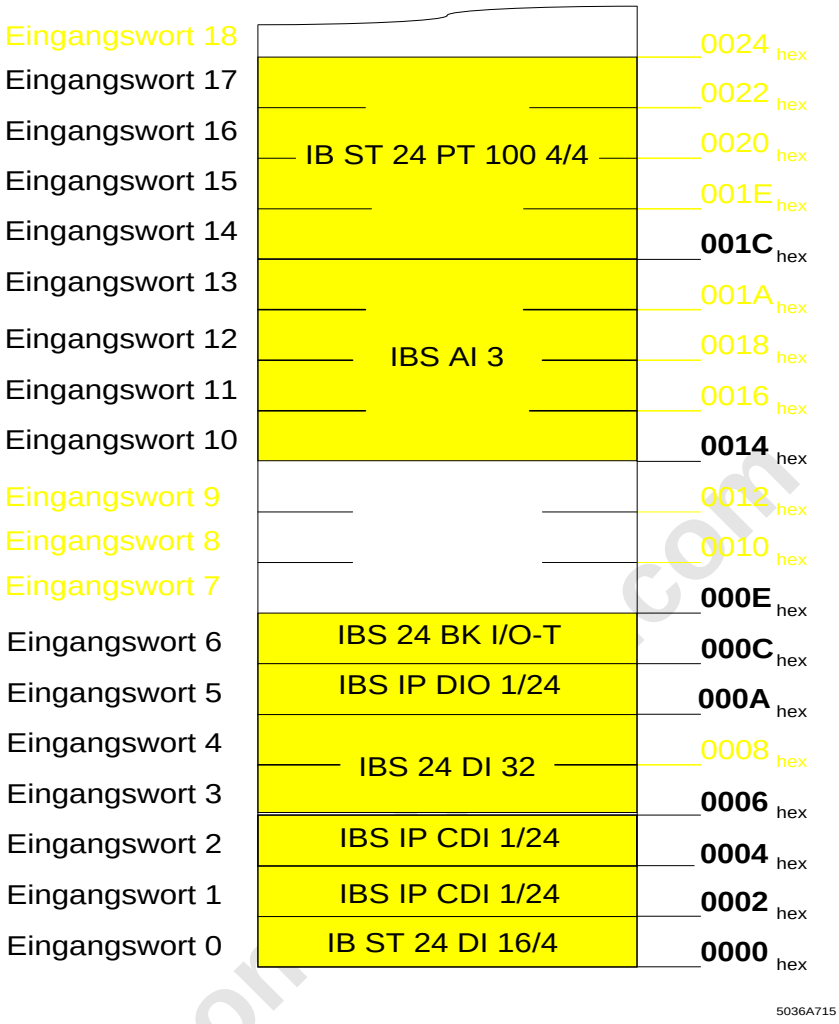

Bild 6-12: Eingangsadressen im Speicherabbild (IN-Buffer)
Die Startadressen der Module sind fett hervorgehoben.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.3.4.2 Vergabe der logischen Ausgangsadressen

Das IBS-Kommando *Receive_Logical_OUT_Address_Map_Request* ($003B_{hex}$) überträgt die Liste der logischen Ausgangsadressen zur Anschaltbaugruppe.

Wort 1	Code			
Wort 2	Parameterzähler (n)			
Wort 3	DC	0000_{bin}	Ausgangsadresse	Teilnehmer Nr. 1
Wort 4	DC	0000_{bin}	Ausgangsadresse	Teilnehmer Nr. 2
	...	...	...	
Wort n+2	DC	0000_{bin}	Ausgangsadresse	Teilnehmer Nr. n

Bit	15 14	13 9	9 0

Bild 6-13: Aufbau des Kommandos *Receive_Logical_OUT_Address_Map*

Legende:

Code: Kommando-Code (hier $003B_{hex}$)

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).

Ausgangsadresse: Tragen Sie hier (10 Bit) für alle Module mit Prozeßausgangsdaten die gewünschte Adresse (als Byteadresse) ein. Bei Modulen mit mehr als einem Prozeßausgangswort (z.B. mit 32 Bit) werden automatisch die nächsthöheren Adressen mit belegt.

DC: Tragen Sie hier die Datenkonsistenz für den Prozeßdatenzugriff ein. Die Datenkonsistenz garantiert, daß die angegebene Datenbreite aus einem IBS-Zyklus stammt. Der Standardwert für E/A-Module ist 16 Bit (00_{bin}). Für IBS-Teilnehmer, welche die zusammenhängende Übertragung größerer Datenbreiten verlangen, müssen Sie die Datenkonsistenz erhöhen. Dies sind z.B. Encoder, Bedienterminals oder Analogmodule mit mehr als 16 Bit Auflösung. Für die Datenkonsistenz sind folgende Werte zulässig:

- 00_{bin} 16 Bit (Standard)
- 01_{bin} 32 Bit (z.B. Encoder, Bedienterminals)
- 10_{bin} 8 Bit (nur für Module mit Längencode 81_{hex}, auf die Bytezugriffe erfolgen sollen!)
- 11_{bin} 48 Bit (z.B. Encoder, Bedienterminals)

Tragen Sie für alle Module ohne Prozeßausgangsdaten (z.B. reine Busklemmen oder reine Eingangsmodule) den Eintrag 0000_{hex} ein.

Legen Sie die Startadressen von IBS-Teilnehmern mit 16 Bit (oder mehr) Adreßraum nur auf gerade Adressen.

Die Liste wird erst nach erfolgreicher Ausführung des Kommandos *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) gültig.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Buffer in der Programmiersprache "C":

```
/*****************************************************************/
/* "Receive_Logical_OUT_Address_Map_Request"                     */
/*  to implement all out addresses                               */
/*****************************************************************/
USIGN16 out_adr_map[] =
             {0x003B,  //Command Code
              0x000D,  //Parameter Counter
              0x0000,  //IBS ST 24 BK-T
              0x0000,  //IB ST 24 DI 16/4
              0x0014,  //IB ST 24 AO 4/SF
              0x0000,  //IB ST 24 DO 16/3
              0x0000,  //IBS PT 100
              0x0004,  //IBS IP DIO 1/24
              0x0000,  //IBS IP CBK 1/24
              0x0000,  //IBS IP CDI 1/24
              0x0002,  //IBS IP CDO 1/24
              0x0000,  //IBS IP CDI 1/24
              0x0006,  //IBS 24 BK I/O-T
              0x0000,  //IBS 24 DI/32
              0x001C}; //IBS AI 3
```

Buffer in der Programmiersprache "Pascal":

```
{*****************************************************************}
{* "Receive_Logical_OUT_Address_Map_Request"                     *}
{*  to implement all out addresses                               *}
{*****************************************************************}
const out_adr_map : array[1..15] of word =
             ($003B,  {Command Code}
              $000D,  {Parameter Counter}
              $0000,  {IBS ST 24 BK-T}
              $0000,  {IB ST 24 DI 16/4}
              $0014,  {IB ST 24 AO 4/SF}
              $0000,  {IB ST 24 DO 16/3}
              $0000,  {IBS PT 100}
              $0004,  {IBS IP DIO 1/24}
              $0000,  {IBS IP CBK 1/24}
              $0000,  {IBS IP CDI 1/24}
              $0002,  {IBS IP CDO 1/24}
              $0000,  {IBS IP CDI 1/24}
              $0006,  {IBS 24 BK I/O-T}
              $0000,  {IBS 24 DI/32}
              $001C); {IBS AI 3}
```

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

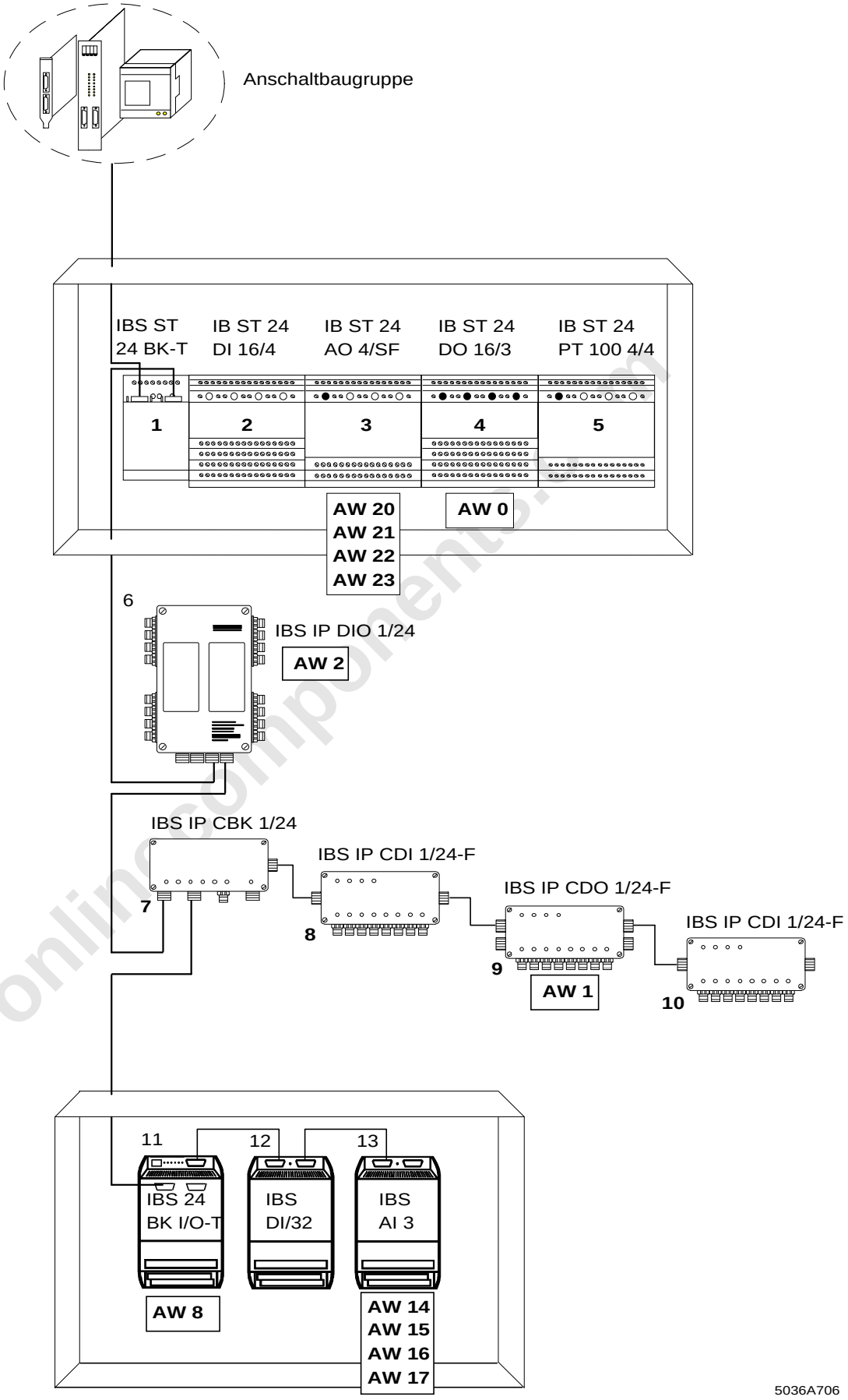

Bild 6-14: Ausgangsadressen bei logischer Adressierung

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

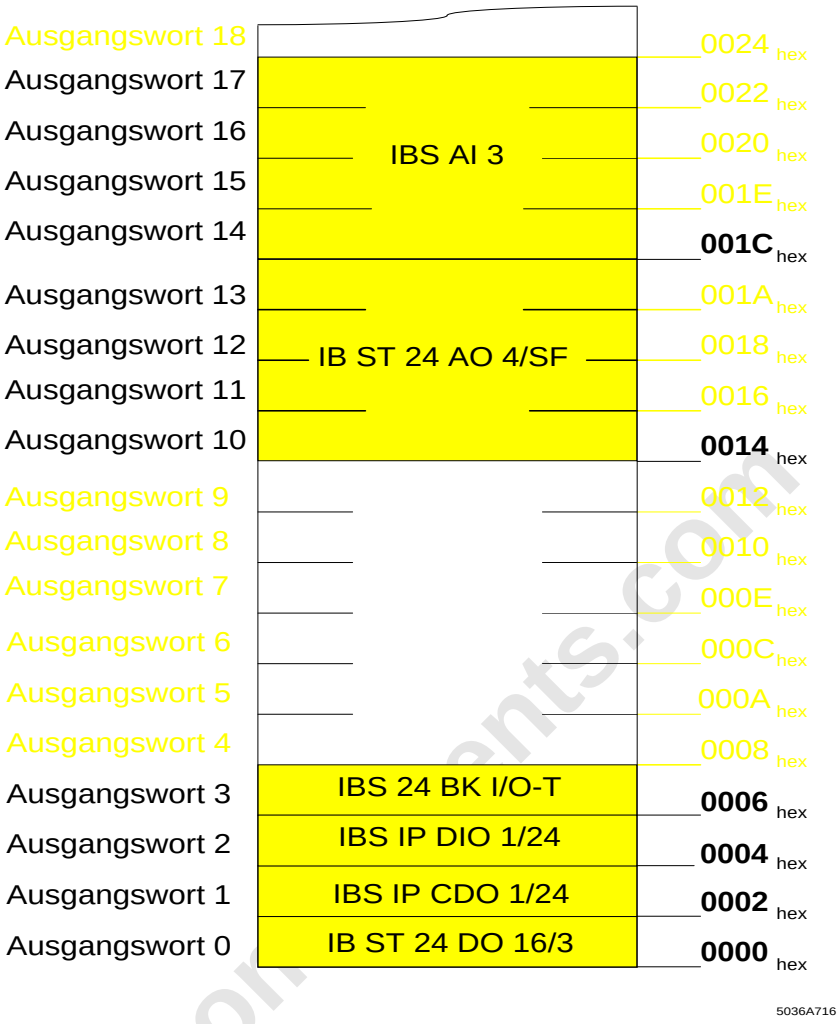

Bild 6-15: Ausgangsadressen im Speicherabbild (Out-Buffer)
Die Startadressen der Module sind fett hervorgehoben.

6.3.4.3 Gültigkeit der Zuordnungslisten prüfen

Das InterBus-S-Kommando *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) überprüft mit IBS-Kommandos zur Anschaltbaugruppe übertragene Zuordnungslisten auf Konsistenz und Plausibilität. Falls die Listen fehlerfrei sind werden sie von der Anschaltbaugruppe übernommen und im RAM gespeichert.

Es ist sinnvoll das Kommando *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) nach jeder Übertragung von Adreßlisten (z.B. den Kommandos 0069_{hex}, $003A_{hex}$, $003B_{hex}$) erneut aufzurufen um im Fehlerfall exakt sagen zu können, welche Adreßliste fehlerhaft ist.

Im Fehlerfall werden zuvor übertragene Listen nicht übernommen. Die negative Quittung für dieses Kommando lautet *Logical_Address_Error* ($002B_{hex}$).

Das Kommando *Send_Log_Address_Error_Request* ($005F_{hex}$) erzeugt eine Meldung mit detaillierte Informationen zur Fehlerursache.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Das Kommando *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) darf nur angewendet werden, wenn sich der Bus im STOP-Zustand befindet. Rufen Sie gegebenenfalls vor der Ausführung dieses Kommandos das Kommando *Alarm_Stop_Request* ($004A_{hex}$) auf.

6.3.5 Kommandofolge für den Anlauf bei logischer Adressierung

Tabelle 6-6: Kommandofolge für den Anlauf bei logischer Adressierung

Aktion	Kommando	Code
Ausgänge in sicheren Zustand setzen, Datenübertragung auf dem Bus stoppen	Alarmstop_Request	$004A_{hex}$
Register für Diagnosebits löschen	Clear_Display_Request	$004E_{hex}$
Bussystem konfigurieren	Configure_Bus_Request	0023_{hex}
Konfiguration überprüfen	Check_Physical_Configuration_Request	0058_{hex}
Bussegmente numerieren	Receive_Logical_Local_Bus_Address_Map_Request	0069_{hex}
Gültigkeit der Zuordnungslisten prüfen	Implement_All_Logical_Address_Maps_Request	0040_{hex}
Logische IN-Adreßliste übertragen	Receive_Logical_IN_Address_Map_Request	$003A_{hex}$
Gültigkeit der Zuordnungslisten prüfen	Implement_All_Logical_Address_Maps_Request	0040_{hex}
Logische OUT-Adreßliste übertragen	Receive_Logical_OUT_Address_Map_Request	$003B_{hex}$
Gültigkeit der Zuordnungslisten prüfen	Implement_All_Logical_Address_Maps_Request	0040_{hex}
Starten der Datenzyklen	Start_Bus_Cycle_Request	0001_{hex}

Kapitel 8 beschreibt die Kommandos detailliert.

Es ist sinnvoll das Kommando *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) nach jeder Übertragung von Adreßlisten (z.B. den Kommandos 0069_{hex}, $003A_{hex}$, $003B_{hex}$) erneut aufzurufen um im Fehlerfall exakt sagen zu können, welche Adreßliste fehlerhaft ist.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.4 Gruppendefinition

6.4.1 Bilden von funktionalen Gruppen

Wenn z.B. bestimmte Anlageteile abgeschaltet werden sollen, während andere Anlageteile weiterlaufen, müssen Sie vorher Bussegmente zu funktionale Gruppen zusammenfassen. Anschließend können einzelne Gruppen durch Ausführung der InterBus-S-Kommandos *Switch_Group_Off_Request* (0021_{hex}) und *Switch_Group_On_Request* (0020_{hex}) ab- bzw. zugeschaltet werden. Das IBS-Kommando *Receive_Group_Numbers_Request* (0049_{hex}) bildet Gruppen. Ordnen Sie jedem Bussegment eine Gruppennummer von 0 bis 255 zu.

Wort 1	Code		
Wort 2	Parameterzähler (n)		
Wort 3	00_{hex}	Bussegment	für 1. Bussegment
Wort 4	00_{hex}	Gruppe	für 1. Bussegment
Wort 5	00_{hex}	Bussegment	für 2. Bussegment
Wort 6	00_{hex}	Gruppe	für 2. Bussegment
	...	...	
	...	...	
Wort n+1	00_{hex}	Bussegment	für x. Bussegment
Wort n+2	00_{hex}	Gruppe	für x. Bussegment
Bit	15 8	7 0	

Bild 6-16: Aufbau des Kommandos *Receive_Group_Numbers_Request*

Legende:

Code:	Kommando-Code (hier 0049_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Bussegmente multipliziert mit 2)
Bussegment:	Nummer des Bussegmentes, das der in der folgenden Zeile benannte Gruppe zugewiesen wird
Gruppe:	Nummer der Gruppe, der das in der vorhergehen Zeile benannte Bussegment zugewiesen wird (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

Tragen Sie nur IBS-Teilnehmer ein, die sich im Hauptstrang des Busaufbaus (FB-Ebene 0) befinden. Teilnehmer im Fernbus-Stich (FB-Ebene 1) werden in die Gruppendefinition einbezogen, indem die Busklemme, welche diese Stichleitung eröffnet, zu einer Gruppe zugewiesen wird.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

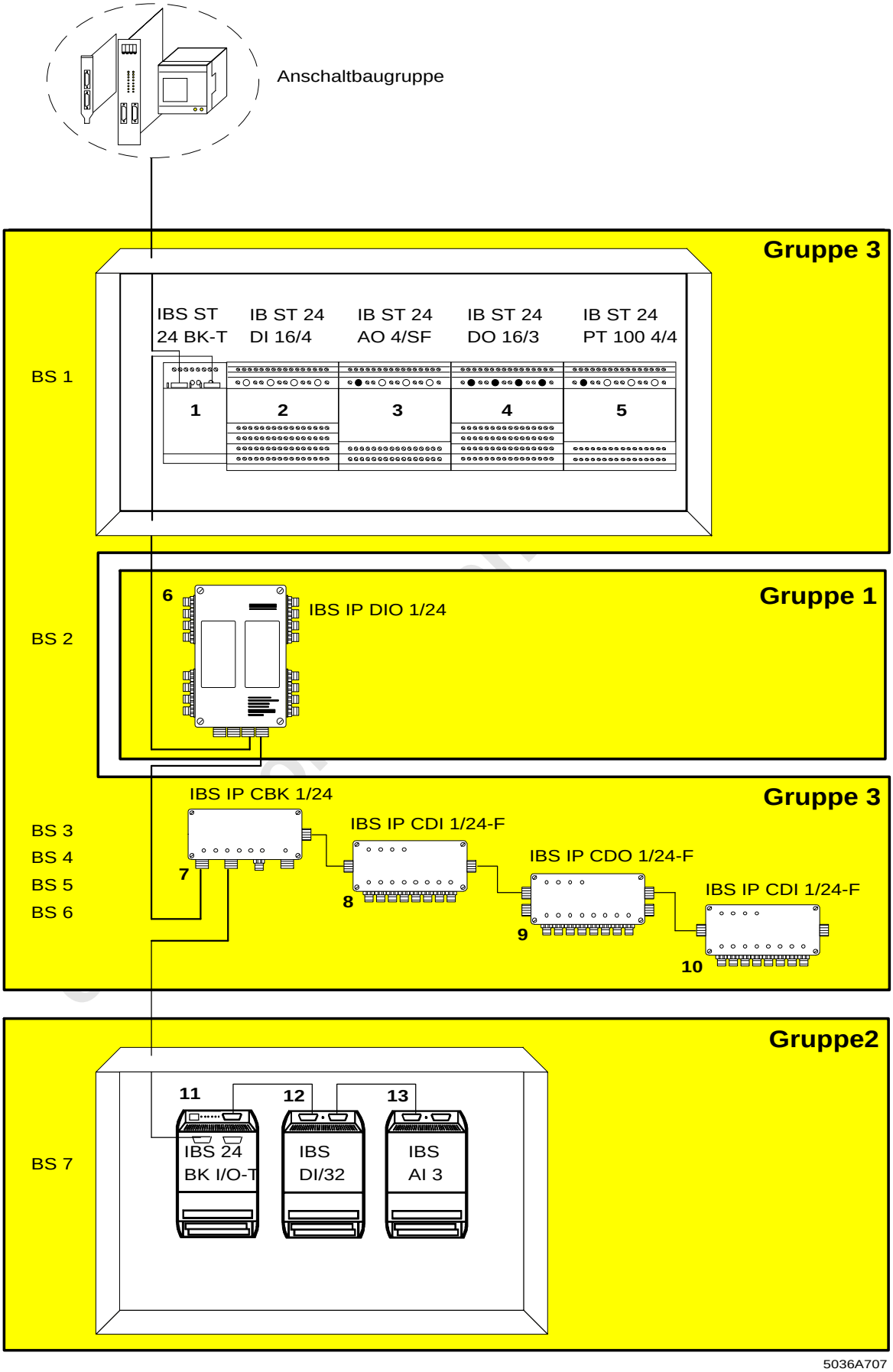

Bild 6-17: Einteilung in logische Funktionsgruppen

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Buffer in der Programmiersprache "C":

```
/*****************************************************************/
/* "Receive_Group_Numbers_Request" to implement logical groups   */
/*****************************************************************/
USIGN16 grp_num[] =
             {0x0049, // Command Code
              0x0008, // Parameter Counter
              0x0015, // lb number
              0x0003, // group number to which this lb should belong
              0x000B, // lb number
              0x0001, // group number to which this lb should belong
              0x0016, // lb number
              0x0003, // group number to which this lb should belong
              0x000C, // lb number
              0x0002};// group number to which this lb should belong
```

Buffer in der Programmiersprache "Pascal":

```
{*************************************************************}
{* "Receive_Group_Numbers_Request" to implement logical groups *}
{*************************************************************}
const grp_num : array[1..10] of word =
             ($0049, {Command Code}
              $0008, {Parameter Counter}
              $0015, {lb number}
              $0003, {group number to which this lb should belong}
              $000B, {lb number}
              $0001, {group number to which this lb should belong}
              $0016, {lb number}
              $0003, {group number to which this lb should belong}
              $000C, {lb number}
              $0002);{group number to which this lb should belong}
```

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.4.2 Abschalten von Gruppen

Mit dem Kommando *Switch_Group_Off_Request* (0021_{hex}) schalten Sie eine bestimmte Gruppe ab. Die Ausgänge der sich im abgeschalteten Bussegment befindenden IBS-Teilnehmer und die zugehörigen Eingänge des Steuerungs bzw. Rechnersystems werden auf 0 gesetzt.

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Gruppennummer
Bit	15 8	7 0

Bild 6-18: Aufbau des Kommandos *Switch_Group_Off_Request*

Legende:

Code:	Kommando-Code (hier 0021_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Gruppennummer:	Nummer der Gruppe die abgeschaltet werden soll. (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

Beim Abschalten eines Anlagenteils muß für die darin enthaltenen IBS-Teilnehmer mit Busklemmenfunktionalität die Versorgungsspannung für die Modulelektronik (Logikspannung) erhalten bleiben, damit der restliche Busaufbau weiter betrieben werden kann.

Buffer in der Programmiersprache "C":

```
/*********************************************************************/
/* "Switch_Group_Off_Request" to switch off a group                  */
/*********************************************************************/
USIGN16 grp_off[] =
             {0x0021,  // Command Code
              0x0001,  // Parameter Counter
              0x0003}; // switch off group number 1
```

Buffer in der Programmiersprache "Pascal":

```
{*********************************************************************}
{* "Switch_Group_Off_Request" to switch off a group                  *}
{*********************************************************************}
const grp_off : array[1..3] of word =
             ($0021,  {Command Code}
              $0001,  {Parameter Counter}
              $0003); {switch off group number 1}
```

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

6.4.3 Einschalten von Gruppen

Mit dem Kommando *Switch_Group_On_Request* (0020_{hex}) schalten Sie eine bestimmte, zuvor abgeschaltete Gruppe wieder ein.

Wort 1	Code		
Wort 2	Parameterzähler		
Wort 3	0_{hex}	0_{hex}	Gruppennummer
Bit	15 12	11 8	7 0

Bild 6-19: Aufbau des Kommandos *Switch_Group_On_Request*

Legende:

Code:	Kommando-Code (hier 0020_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Gruppennummer:	Nummer der Gruppe die eingeschaltet werden soll. (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

Buffer in der Programmiersprache "C":

```
/*****************************************************************/
/* "Switch_Group_On_Request" to switch on a group                */
/*****************************************************************/
USIGN16 grp_on[] =
             {0x0020,  // Command Code
              0x0001,  // Parameter Counter
              0x0003}; // switch on group number 1
```

Buffer in der Programmiersprache "Pascal":

```
{*****************************************************************}
{* "Switch_Group_On_Request" to switch on a group                *}
{*****************************************************************}
const grp_on : array[1..3] of word =
              ($0020,  {Command Code}
               $0001,  {Parameter Counter}
               $0003); {switch on group number 1}
```

6.4.4 Behandlung der Gruppen für den Fehlerfall definieren

Mit dem Kommando *Define_Group_Error_Characteristic_Request* (0060_{hex}) legen Sie für jede Gruppe fest, ob der Bus nach einem Fehler in dieser Gruppe (Bus im STOP-Zustand) ohne diese Gruppe automatisch wieder gestartet werden soll.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

.

Wort 1	Code			
Wort 2	Parameterzähler (n)			
Wort 3	Mode	0_{hex}	Gruppennummer	für Gruppe Nr. 1
Wort 4	Mode	0_{hex}	Gruppennummer	für Gruppe Nr. 2
	...	...	...	
Wort n+2	Mode	0_{hex}	Gruppennummer	für Gruppe Nr. n
Bit	15 12	11 8	7 0	

Bild 6-20: Aufbau des Kommandos *Define_Group_Error_Characteristic_Req.*

Legende:

Code: Kommando-Code

Parameterzähler: Anzahl der nachfolgenden Worte

Mode: Handhabung des restlichen Bussystems bei einem Fehler in der der durch diese Zeile beschriebenen Gruppe
0_{hex} Der gesamte Bus bleibt im Fehlerfall im STOP-Zustand.
8_{hex} Der Bus wird ohne die fehlerhafte Gruppe wieder gestartet.

Gruppennummer: Gruppennummer der durch diese Zeile beschriebenen Gruppe
(00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

Buffer in der Programmiersprache "C":

```
/*****************************************************************/
/* "Define_Groups_Error_Characteristic_Request"                  */
/*  for restart after an error                                    */
/*****************************************************************/
USIGN16 grp_err_chr[] =
            {0x0060,  // Command Code
             0x0003,  // Parameter Counter
             0x8001,  // Group 1
             0x8002,  // Group 2
             0x8003}; // Group 3
```

Buffer in der Programmiersprache "Pascal":

```
{*****************************************************************}
{* "Define_Groups_Error_Characteristic_Request"                 *}
{*  for restart after an error                                   *}
{*****************************************************************}
const grp_err_chr : array[1..5] of word =
             ($0060,  {Command Code}
              $0003,  {Parameter Counter}
              $8001,  {Group 1}
              $8002,  {Group 2}
              $8003); {Group 3}
```

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel 7

Diagnose im Fehlerfall

Dieses Kapitel informiert Sie über die Diagnose und Behebung

- von Fehlern im InterBus-S-System (Anschaltbaugruppe, Busaufbau)
- von Fehlern im Anwenderprogramm

7 Diagnose im Fehlerfall 7-3

7.1 Diagnose an der Hardware. 7-3
7.1.1 Diagnoseanzeigen an der Anschaltbaugruppe 7-3
7.1.2 Diagnoseanzeigen an Busklemmen 7-4
7.1.3 Diagnoseanzeigen an IBS-Teilnehmern mit E/A-Funktion . . . 7-4
7.1.4 Diagnose an IBS-Teilnehmern von anderen Geräteherstellern . 7-5
7.2 Diagnose mit Softwaretools 7-6
7.2.1 Das Prozeßdaten-Monitorprogramm 7-6
7.2.2 Die Diagnose und Konfigurations-Software IBS SYS SWT . . 7-6
7.2.3 Der InterBus-Manager IBS CMD SWT 7-7
7.3 Diagnose durch das Anwenderprogramm 7-8
7.3.1 Diagnose von Anschaltbaugruppe und Busaufbau 7-8
7.3.1.1 Fehlerart 7-9
7.3.1.2 Bedeutung von Controller-Fehlernummern 7-10

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

7 Diagnose im Fehlerfall

InterBus-S ermöglicht im Fehlerfall eine umfangreiche Diagnose.

7.1 Diagnose an der Hardware

7.1.1 Diagnoseanzeigen an der Anschaltbaugruppe

Die Anschaltbaugruppe bietet auf dem PC-Kartenhalter 4 LEDs zur einfachen Diagnose des Betriebszustands.

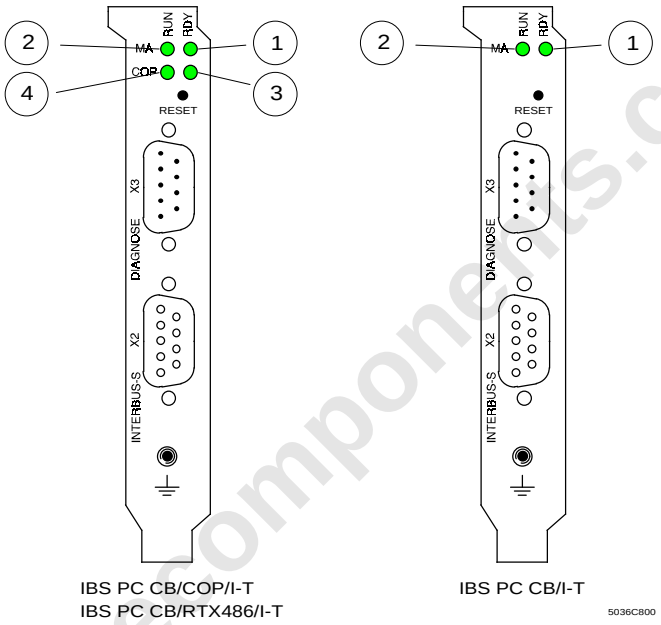

Bild 7-1: Diagnose-Anzeigen auf dem PC-Kartenhalter

1 Externe LED *MA READY* (grün)
2 Externe LED *MA RUN* (grün)
3 Externe LED *COP READY* (grün)
4 Externe LED *COP RUN* (grün)

IBS-Masterboard

MA READY: Das IBS-Masterboard hat nach dem Einschalten einen Bootcheck für alle Funktionseinheiten inklusive MPM fehlerfrei ausgeführt und ist betriebsbereit.

MA RUN: Das IBS-Masterboard hat den InterBus-S gestartet. Es werden ID- oder Datenzyklen übertragen.

Coprozessor-Board
(nicht IBS PC CB/I-T)

COP RUN: Das Betriebssystem des Coprozessorboards hat gebootet; es kann ein Anwendungsprogramm gestartet werden.

COP READY: Auf dem Coprozessorboard läuft ein Programm.

online-components.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Nach dem Booten des Coprozessorboards werden verschiedene Dienstprogramme automatisch gestartet, die teilweise als TSR-Programm im Speicher des COPs verbleiben. Daher leuchtet nach dem Hochfahren des Gesamtsystems die LED *COP READY*.

7.1.2 Diagnoseanzeigen an Busklemmen

Auch die Busklemmen bieten vor Ort über LEDs eine lokale Diagnosemöglichkeit ihres Betriebszustandes. Bild 8-2 zeigt beispielhaft die Diagnose an der Busklemme IBS 24 BK-T.

Typische Anzeigen auf Busklemmen:
- Versorgungsspannung
- Fernbusüberwachung
- Peripheriebusüberwachung
- Betriebsanzeige des InterBus-S

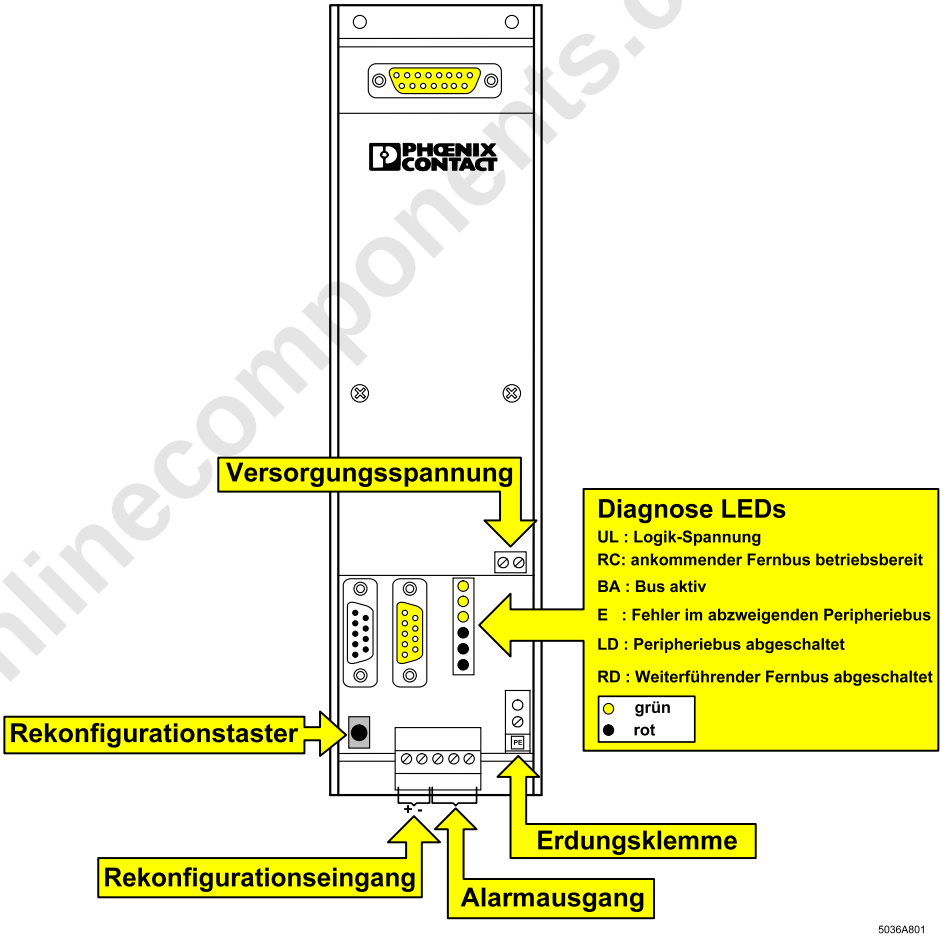

Bild 7-2: LED-Diagnose an Busklemmen

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

7.1.3 Diagnoseanzeigen an IBS-Teilnehmern mit E/A-Funktion

Auch an den IBS-Teilnehmern mit E/A-Funktion sind Status- und Diagnose-LEDs angebracht, um eine schnelle Diagnose vor Ort zu ermöglichen. Die Diagnose-LEDs geben Hinweise auf die Art und den Ort des aufgetretenen Fehlers. Die Status-LEDs zeigen die Schaltzustände der Ein- und Ausgänge (E/As) an.

Typische Anzeigen an IBS-Teilnehmern mit E/A-Funktion:
- Versorgungsspannung
- Ausgangsüberlastung/Kurzschluß (gruppenweise, einzelne DI/DO)
- Status-LED je E/A-Kanal

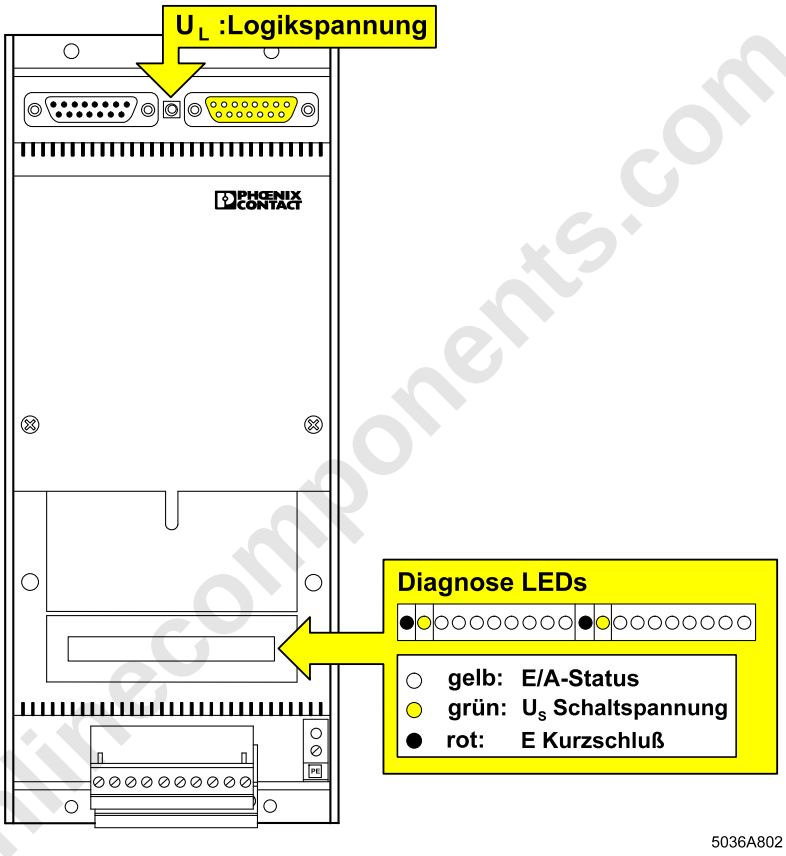

Bild 7-3: LED-Diagnose an E/A-Modulen

7.1.4 Diagnose an IBS-Teilnehmern von anderen Geräteherstellern

Die Diagnoseanzeigen an IBS-Teilnehmern von anderen Geräteherstellern gliedern sich in zwei Anzeigegruppen:

- Busspezifische Anzeigen:
  Diese Anzeigen entsprechen weitgehend den Anzeigen auf Busklemmen oder IBS-Teilnehmern mit E/A-Funktion von Phoenix Contact.
- Gerätespezifische Anzeigen:
  Diese Anzeigen sind vom jeweiligen IBS-Teilnehmern abhängig. Schlagen Sie diese bitte in der zugehörigen Dokumentation nach oder wenden Sie sich an den Gerätehersteller.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

7.2 Diagnose mit Softwaretools

Zur Diagnose stehen Ihnen verschiedene Software-Tools zur Verfügung, die das Ansprechen Ihres Busaufbaus ohne großen Programmieraufwand ermöglichen.

- Das Prozeßdaten-Monitorprogramm *PCCBMONI.EXE* (gehört zum Lieferumfang der Treibersoftware)
- Die Diagnose und Konfigurations-Software *IBS SYS SWT* unter DOS (Artikel-Nr. 27 80 88 1)
- Der InterBus-Manager *IBS CMD SWT*, ein grafikorientiertes Programm, das Konfiguration, Monitoring und Diagnose ihres IBS-Systems unter Microsoft Windows® ermöglicht (Artikel-Nr 27 53 17 7).

7.2.1 Das Prozeßdaten-Monitorprogramm

Zum Lieferumfang der Treibersoftware für IBS PC CB/.../I-T gehört ein Prozeßdaten-Monitorprogramm.

Übertragen Sie die Datei *PCCBMONI.EXE* von der Treiberdiskette in ein Verzeichnis auf ihrer Festplatte und starten Sie das Prozeßdaten-Monitorprogramm von dort durch den Aufruf *PCCBMONI*. Das Programm leistet folgende Dienste:

- Erste Inbetriebnahme eines InterBus-S-Systems ohne Programmieraufwand
- Test ihres Busaufbaus
- Kontrolle der angeschlossenen Konfiguration
- Anzeigen des Status von einzelnen Eingängen (binär)
- Anzeigen des Status von Eingangsworten (hexadezimal)
- Setzen von einzelnen Ausgängen (binär)
- Setzen von Ausgangsworten (hexadezimal)
- Rücksetzen des InterBus-S-Systems

Die Bedienung des Monitorprogramms wird im Kapitel 4 ausführlich beschrieben.

7.2.2 Die Diagnose und Konfigurations-Software IBS SYS SWT

Das Programm *IBS SYS SWT* (Artikel-Nr. 27 80 88 1) ist eine Bedienoberfläche für InterBus-S auf IBM-kompatiblen PCs unter DOS. Es überträgt und speichert Parametrierungsdaten zur bzw. auf der Anschaltbaugruppe und ermöglicht eine einfache, menügeführte Parametrierung und Steuerung der Anschaltbaugruppe. Die Funktionen der IBS-Komponenten (Anschaltbaugruppen, Module usw.) sind durch die Verwendung von *IBS SYS SWT* ohne Programmieraufwand zu nutzen.

Die Kopplung des PCs mit *IBS SYS SWT* und der Anschaltbaugruppe erfolgt über die seriellen Schnittstellen (RS-232) beider Geräte.

Die Funktionen des Programms im einzelnen:

- Übertragen von Kommandos zur Anschaltbaugruppe.
- Empfang von Meldungen von der Anschaltbaugruppe.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

- Protokollieren der empfangenen Meldungen und der übertragenen Kommandos.
- Speichern von Kommandofolgen in einer Datei und Übertragen dieser Kommandos zur Anschaltbaugruppe.
- Erstellen und Übertragen von logischen Adreßlisten zur Anschaltbaugruppe. Diese Adreßlisten können auf der Anschaltbaugruppe nicht dauerhaft gespeichert werden, und gehen bei einem Reset der Anschaltbaugruppes verloren.
- Diagnose und Bewertung der Busübertragung

IBS SYS SWT bietet dem Anwender unter DOS eine einfache Schnittstelle zur Anschaltbaugruppe. Weitere Hinweise zu *IBS SYS SWT* finden Sie im Handbuch IBS SYS SWT UM (Artikel-Nr. 27 58 77 4).

7.2.3 Der InterBus-Manager IBS CMD SWT

Der InterBus-Manager *IBS CMD SWT* ist eine grafikorientiertes Programm unter Microsoft Windows®. Es stellt Funktionen zur Projektierung, Inbetriebnahme und Diagnose des InterBus-S zur Verfügung. Dialogfunktionen ermöglichen das Bedienen und Anzeigen (Monitoring) aller angeschlossenen IBS-Teilnehmer. Darüberhinaus ermöglicht die offene Struktur der Oberfläche die Integration von Hersteller- oder gerätespezifischen Bedien- und Parametrierungsfunktionen.

Die Funktionen des Programms im einzelnen:

- Entwurf eines Busaufbaus zur Erstellung der Adreßzuordnung zwischen dem Host und den angeschlossenen IBS-Teilnehmern
- Parametrierung von komplexen IBS-Teilnehmern
- Dialogfunktionen zur Datenausgabe (z.B. Setzen von Ausgängen) sowie Lesen und Darstellen von aktuellen Eingangsdaten (Monitoring von Eingängen)
- Diagnosefunktionen zum Erkennen und Lokalisieren von defekten Anlagenteilen (IBS-Teilnehmer, Leitungsstränge, Spannungsversorgungen etc.)
- Dokumentation Ihres IBS-Systems durch das Erstellen einer Anlagenbeschreibung inklusive der parametrierten Geräteeinstellungen

Zusätzliche Funktionen werden durch Dienstprogramme ermöglicht, die von anderen IBS-Geräteherstellern oder IBS-Nutzergruppen (z.B. DRIVECOM) erstellt werden. Diese Dienstprogramme können vom Anwender einfach in die Benutzeroberfläche von *IBS CMD SWT* eingebunden werden.

Die Bedien-Software *IBS CMD SWT* stellt somit eine für alle IBS-Geräteherstellen und IBS-Anwender offene Plattform dar, die übersichtlich Projektierung, Konfiguration, Monitoring und Diagnose ihres InterBus-S-Systems unter Microsoft Windows® ermöglicht.

Weitere Hinweise zu *IBS CMD SWT* finden Sie im Handbuch IBS CMD SWT UM (Artikel-Nr. 27 52 79 6).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

7.3 Diagnose durch das Anwenderprogramm

7.3.1 Diagnose von Anschaltbaugruppe und Busaufbau

Nach dem Start des IBS-Systems durch das Kommando *Start_Bus_Cycle_Request* (0001_{hex}) betreibt das IBS-Masterboard das IBS-System selbständig und aktualisiert zyklisch die E/A-Daten im MPM der Anschaltbaugruppe. Bei Auftreten eines Fehlers innerhalb des IBS-Systems sendet das IBS-Masterboard eine Meldung. Je nach Art und Schwere des Fehlers sind zwei Reaktionen zu unterscheiden:

Leichte Fehler, die den weiteren Betrieb des IBS-Systems nicht stören z.B.:
- Ausfall der Peripherie-Versorgungsspannung an einem IBS-Teilnehmer mit Fehlerrückmeldung
- Überlastung eines Ausgangs an einem IBS-Teilnehmer mit Fehlerrückmeldung

In diesem Fall reagiert das IBS-Masterboard z.B. mit der Meldung *Modul-Error-Indication* ($80A0_{hex}$). Mit der Diagnosefunktion *GetIBSDiagnostic* kann das Bussegment, in dem das defekte IBS-Teilnehmer enthalten ist, festgestellt werden. Mit dem Kommando *Send_Localbus_Module_Error_Request* ($005B_{hex}$) können Sie, in dem als Parameter anzugebenden Bussegment, den fehlerhaften IBS-Teilnehmer ermitteln. Das IBS-System wird weiterbetrieben.

Schwere Fehler, die den weiteren Betrieb des IBS-Systems nicht zulassen z.B.:
- Veränderung der Bus-Konfiguration, z.B. durch Abziehen eines Bus-Steckers
- Defekt (z.B. Durchtrennen) des Buskabels
- Ausfall der Logik-Versorgungsspannung (nicht Peripherie-Versorgungsspannung) einer Busklemme oder eines IBS-Teilnehmers mit Busklemmenfunktionalität

Das IBS-Masterboard stoppt in diesem Fall sofort den zyklischen Betrieb des IBS-Systems, setzt alle Ausgänge zurück und sendet die Meldung *Bus_System_Error_Indication* (0038_{hex}). Danach untersucht das IBS-Masterboard das IBS-System und gibt die Fehlerursachen in der Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$) an. Nähere Informationen zu dieser Meldung finden Sie im Kapitel 9.

Die Meldung *Bus_Error_Information_Indication* (80C4) beschreibt die Fehler sehr detailliert. Einfacher ist die Nutzung der Diagnosefunktion *GetIBSDiagnostic*. Sie stellt nur die wesentlichen Informationen zur Verfügung und erleichtert eine schnelle Diagnose des IBS-Systems. Nach Beseitigung der Fehlerursache (z.B. Abschalten des defekten Bussegments über ein Kommando) kann das IBS-System wieder gestartet werden (siehe Kapitel 6 *Kommandofolge zur Inbetriebnahme des InterBus-S*).

Zusätzlich kann die Diagnosefunktion dazu genutzt werden,

- durch Abfrage des *Ready-Bits* festzustellen, ob das IBS-Masterboard beim Neuanlauf oder nach einem Warmstart mit dem Kommando

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Warmstart_Request (004C) wieder betriebsbereit ist oder
- durch Abfrage des *Run-Bits* festzustellen, ob das IBS-Masterboard den Bus betreibt und die E/A-Daten zyklisch aktualisiert werden.

Nähere Informationen zu den Meldungen

- Modul_Error_Ind ($80A0_{hex}$),
- Bus_System_Error_Indication (0038_{hex}),
- Bus_Error_Information_Indication ($80C4_{hex}$)

und zu weiteren Fehlermeldungen finden Sie im Kapitel 9 *Meldungen des IBS-Masterboards*.

7.3.1.1 Fehlerart

Die Anschaltbaugruppe zeigt das Auftreten eines Fehlers auf dem InterBus-S durch die Meldung *Bus_System_Error_Indication* (0038_{hex}) an. Dieser Meldung folgt die Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$), welche detaillierte Information über den Fehlerzustand des Bussystems enthält. Der erste Parameter dieser Meldung gibt die Fehlerart an. Dabei unterscheidet die Firmware-Version 3.72 zwischen sechs verschiedene Fehlerarten.

Tabelle 7-1: Fehlerarten unter Firmware-Version 3.72

Fehlerart	Bedeutung	Fehlerursache	Fehlerbehebung
EE01	Nach dem Auftreten des Fehlers wurde kein Fehler beim Erfassen und Vergleichen der Konfiguration gefunden.	- Verkabelungsfehler - Schirmungsfehler	- Überprüfen Sie die Fern- und Peripheriebusverkabelung. - Überprüfen Sie die Spannungsversorgung. Hilfsmittel: IBS SYS SWT oder IBS CMD SWT
EE02	Die maximal mögliche Konfiguration wurde überschritten.	Zahl der Teilnehmer oder der Registerplätze zu groß.	Überprüfen Sie die Adressierungslisten und die angeschlossenen Konfiguration. Es sind maximal 256 Busklemmen und 256 Worte (Summe aller Registerlängen und der PCP-Worte) auf dem Datenring zulässig.
EE03	Fehler in einem Bussegment	- Busklemme oder vorgelagertes Fernbuskabel defekt - Peripheriebuskabel oder Modul defekt	Beheben Sie die Fehlerursachen. Anstelle der Fehlerart steht beim Fehler EE03 die Nummer des betroffenen Bussegmentes im mit der Diagnosefunktion *GetIBSDiagnostic* auslesbaren *Register für Diagnoseparameter*.
EE04	Die Konfiguration konnte nicht erfaßt werden.	IBS-Teilnehmer meldet sich nicht.	Überprüfen Sie die Spannungsversorgung der Teilnehmer auf Einbrüche.
EE05	Alle Gruppen wurden abgeschaltet.	Fehler im Anwendungsprogramm.	Überprüfen Sie das Anwendungsprogramm.
EE06	Kein Datenzyklus möglich, obwohl kein Fehler beim Erfassen und Vergleichen der Konfiguration gefunden erden kann.	Modulfehler	Lokalisieren Sie den Fehler mit IBS SYS SWT oder IBS CMD SWT und Informieren Sie den technischen Support von Phoenix Contact.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Auch das Register für Diagnoseparameter zeigt im Fehlerfall die Fehlerarten EE01, EE02, EE04, EE05 und EE06 an. Anstelle der Fehlerart steht dort beim Fehler EE03 die Nummer des betroffenen Bussegmentes.

Die Beschreibung der Meldung *Bus_Error_Information_Indication* (80C4) im Kapitel 9 enthält weitere Erläuterungen über die Bedeutung, Ursache und Abhilfe bei Auftreten der sechs verschiedenen Fehlerarten.

7.3.1.2 Bedeutung von Controller-Fehlernummern

Wenn die Anschaltbaugruppe fehlerhaft arbeitet, generiert sie eine Controller-Fehlernummer. Mit der Diagnosefunktion *GetIBSDiagnostic* (siehe Handbuch zur Treibersoftware) können Sie diese Controller-Fehlernummer auslesen. Die Fehlernummern haben folgende Bedeutung:

00, 05, 06

Bedeutung: Es wurde ein Firmware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Setzen Sie sich mit Phoenix Contact in Verbindung.

07

Bedeutung: Es ist ein unzulässiges Kommando oder PCP-Kommando abgesetzt worden.

Abhilfe: Überprüfen Sie das Anwenderprogramm und entfernen Sie nicht definierte Kommandos.

08

Bedeutung: Nach einem Kommando werden Parameter erwartet, es folgt aber ein weiteres Kommando.

Abhilfe: Überprüfen Sie im Anwenderprogramm die Kommandosequenzen. Ergänzen Sie eventuell Parameter oder korrigieren Sie die Kommandoaufrufe.

09

Bedeutung: Die Anzahl von Parametern in den Zuordnungslisten ist gleich null.

Abhilfe: Überprüfen Sie die Anzahl der Parameter in den Definitionstlisten.

0A

Bedeutung: Es wurde ein nicht definiertes Kommando abgeschickt.

Abhilfe: Überprüfen Sie das Anwenderprogramm und beseitigen Sie nicht definierte Kommandos.

Setzen Sie sich mit Phoenix Contact in Verbindung, falls der Fehler wiederholt auftritt.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

0B

Bedeutung: Es wurde ein Firmware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Setzen Sie sich mit Phoenix Contact in Verbindung.

0C

Bedeutung: Eine Meldung oder die Quittung auf ein Kommando steht auf der Anschaltbaugruppe an und wird nicht rechtzeitig abgeholt.

Abhilfe: Rufen Sie die Funktion *DDI_MXI_RcvMessage* zyklisch im Anwenderprogramm auf.

Bemerkung: Wenn Sie keine Meldungen wünschen, können Sie das Kommando *Disable_all_Messages_Request* (0048_{hex}) absetzen.

0D

Bedeutung: Die Anschaltbaugruppe erwartet Parameter die innerhalb einer Überwachungszeit nicht empfangen wurden.

Ursache: Der Handshake erfolgte nicht rechtzeitig.

Abhilfe: Lösen Sie einen Reset des Steuerungsrechners aus.

23

Bedeutung: Ein Busfehler führte zum Auslösen des Watchdogs.

Abhilfe: Lösen Sie einen Reset des Steuerungsrechners aus.

Setzen Sie sich mit Phoenix Contact in Verbindung, falls der Fehler wiederholt auftritt.

26, 27, 28, 29

Bedeutung: Es wurde ein Hardware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Setzen Sie sich mit Phoenix Contact in Verbindung.

2A

Bedeutung: Es wurde ein Firmware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Setzen Sie sich mit Phoenix Contact in Verbindung.

2B

Bedeutung: Es wurde ein Fehler in den Adreßlisten festgestellt.

Folgen: Der Bus geht ohne Reset in den Stop, die Ausgänge werden **nicht** zurückgesetzt.

Abhilfe: Überprüfen Sie die Zuordnungslisten im Anwenderprogramm.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Bedeutung: Es wurde versucht, Adreßlisten zur Anschaltbaugruppe zu senden, während der Bus läuft.

Abhilfe: Schalten Sie den Bus in den Stop, bevor Sie Adreßlisten zur Anschaltbaugruppe senden.

38

Bedeutung: Dieser Fehler tritt im Zusammenhang mit einem Peripheriebus oder Fernbus-fehler auf. Eine Testroutine ermittelt das defekte Segment.

Abhilfe: Überprüfen Sie den Busaufbau. Beachten Sie nachfolgende Diagnoseanzei-gen.

3B

Bedeutung: Zyklusende durch externes Signal.

Ursache: Störeinflüsse im Busaufbau.

Abhilfe: Überprüfen Sie den Busaufbau auf Störeinflüße und beseitigen Sie diese. Falls dies keinen Erfolg bringt, setzen Sie sich mit Phoenix Contact in Verbindung.

3C

Bedeutung: Es wurde ein FiFo-Overflow festgestellt.

Abhilfe: Lösen Sie einen Reset des Steuerungsrechners aus.

3D

Bedeutung: Es wurde ein Hardware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Lösen Sie einen Reset des Steuerungsrechners aus.

3E, 3F, 40, 41, 42, 43, 44, 45

Bedeutung: Es wurde ein Firmware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Setzen Sie sich mit Phoenix Contact in Verbindung.

4A

Bedeutung: Die im RAM der Anschaltbaugruppe gespeicherte Konfiguration (Ident-Liste) und die aktuell angeschlossene Konfiguration stimmen nicht überein.

Ursache: Fehlerhafte oder nicht vorhandene Ident-Liste.

Abhilfe: Überprüfen Sie die Ident-Liste

Ursache: Nicht vorhandene InterBus-S-Konfiguration.

Abhilfe: Schließen Sie das Fernbuskabel an die Anschaltbaugruppe an.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

4B

Bedeutung: Zu viel oder zu wenig Parameter im Kommando oder PCP-Kommando.

Abhilfe: Überprüfen Sie die Parameteranzahl im Anwenderprogramm.

4C

Bedeutung: Es wurde ein Fehler in der Event-Definition festgestellt.

Abhilfe: Überprüfen Sie die Event-Definition im Anwenderprogramm.

4D

Bedeutung: Es wurde ein Hardware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Lösen Sie einen Reset der Anschaltbaugruppe aus.

4E

Bedeutung: Sie haben zuviele Kommandos innerhalb kurzer Zeit gesendet.

Abhilfe: Reduzieren Sie Zahl der abgesetzten Kommandos pro Zeit.

50, 51

Bedeutung: Es wurde ein Firmware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Setzen Sie sich mit Phoenix Contact in Verbindung.

55

Bedeutung: Es wurde ein Fehler in der Gruppendefinition festgestellt.

Abhilfe: Überprüfen Sie die Gruppendefinition im Anwenderprogramm.

57, 58

Bedeutung: Es wurde ein Firmware-Fehler auf der Anschaltbaugruppe festgestellt.

Abhilfe: Setzen Sie sich mit Phoenix Contact in Verbindung.

59

Bedeutung: Es wurde in einem Kommando eine nicht definierte Gruppennummer angegeben.

Abhilfe: Verwenden Sie nur definierte Gruppennummern. Überprüfen Sie die Gruppendefinitionen im Anwenderprogramm.

Bedeutung: Es wurde versucht ein Kommando zur Gruppenabschaltung auf eine nicht abschaltbare Gruppe anzuwenden.

Abhilfe: Wenden Sie die Kommandos zur Gruppenabschaltung nur auf abschaltbare Gruppen an.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

5A

Bedeutung: Bei Ein- oder Abschalten eines Busklemmenalarms wurde eine falsche Bussegmentnummer angegeben.

Abhilfe: Überprüfen Sie das Anwenderprogramm. Benutzen Sie nur definierte bzw. vorhandene Bussegmentnummern.

5B

Bedeutung: Im Anwenderprogramm wird ein nicht zugelassener Längen-Code verwendet.

Abhilfe: Überprüfen Sie die Längen-Codes in den Zuordnungslisten.

65

Bedeutung: Die maximale Anzahl von Kommunikationsmodulen im Bus ist überschritten.

Abhilfe: Reduzieren Sie Zahl der Kommunikationmodule auf maximal 62.

66

Bedeutung: Ungültige Kommunikationsreferenz oder falsche Parameteranzahl

Abhilfe: Überprüfen Sie die Kommunikationsreferenzliste (KR-Liste):
- Der Parameter *Parameterzähler* muß mit der Teilnehmerzahl übereinstimmen.
- Die KR-Liste muß von aufsteigender Reihenfolge und ohne Lücken sein.

68

Bedeutung: Die Ident-Liste und die aktuell angeschlossene Konfiguration stimmen nicht überein.

Ursache: Wenn der Fehler bei der Inbetriebnahme auftritt, liegt wahrscheinlich ein Fehler in der Ident-Liste vor.

Abhilfe: Überprüfen Sie die Ident-Liste.

Ursache: Wenn der Fehler im laufenden Betrieb auftritt, ist wahrscheinlich der Busaufbau nicht mehr komplett oder verändert worden.

Abhilfe: Kontrollieren Sie ihren Busaufbau.

69

Bedeutung: Es wurde ein PCP-Kommando abgesetzt, ohne daß zuvor die Kommunikation initialisiert worden ist.

Abhilfe: Überprüfen Sie im Anwenderprogramm die Initialisierung der Kommunikation.

6B

Bedeutung: Es wurde ein Fehler auf der Anschaltbaugruppe festgestellt. Setzen Sie sich mit Phoenix Contact in Verbindung.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel **8**

Kommandos für das IBS-Masterboard

Dieses Kapitel informiert Sie über

- die Aufgaben, und den Aufruf von Kommandos für das IBS-Masterboard
- die Parameter der Kommandos
- die Bedeutung und Ursachen von Meldungen, die auf diese Kommandos erfolgen können

8 Kommandos für das IBS-Masterboard 8-3

8.1 Aufbau einer Kommandobeschreibung 8-5
8.2 Kommandos zur Konfiguration 8-6
8.3 Kommandos zur Adressierung 8-13
8.4 Kommandos für den Betrieb 8-16
8.5 Kommandos zur Fehlerbehandlung 8-19
8.6 Kommandos für die Anwendungsschnittstelle 8-26
8.7 Kommandos zur Systemkontrolle 8-27
8.8 Kommandos zur Prozeßdatenverknüpfung 8-28
8.9 Kommandos zur Event-Bearbeitung 8-38

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8 Kommandos für das IBS-Masterboard

Tabelle 8-1: Kommandos für das IBS-Masterboard

Code	IBS-Kommandos	Seite
0001_{hex}	Start_Bus_Cycle_Request	8-16
0002_{hex}	Stop_Bus_Cycle_Request	8-17
0006_{hex}	Send_Bus_Cycle_Counter_Request	8-27
0008_{hex}	Send_Software_Revision_Request	8-18
0020_{hex}	Switch_Group_On_Request	8-11
0021_{hex}	Switch_Group_Off_Request	8-10
0023_{hex}	Configure_Bus_Request	8-6
0024_{hex}	Set_BK_Alarm_Logical_Request	8-23
0025_{hex}	Reset_BK_Alarm_Logical_Request	8-24
0026_{hex}	Set_BK_Alarm_Physical_Request	8-24
0027_{hex}	Reset_BK_Alarm_Physical_Request	8-25
$002A_{hex}$	Receive_Events_Request	8-38
$002B_{hex}$	Enable_Event_Number_Request	8-40
$002C_{hex}$	Disableable_Event_Number_Request	8-41
$002D_{hex}$	Enable_All_Events_Request	8-39
$002E_{hex}$	Disable_All_Events_Request	8-40
0034_{hex}	Set_Parameter_Timeout_Constant_Request	8-26
0036_{hex}	Enable_Event_Logical_Address_Request	8-41
0037_{hex}	Disable_Event_Logical_Address_Request	8-42
$003A_{hex}$	Receive_Logical_IN_Address_Map_Request	8-14
$003B_{hex}$	Receive_Logical_OUT_Address_Map_Request	8-15
0040_{hex}	Implement_All_Logical_Address_Maps_Request	8-16
0047_{hex}	Enable_All_Messages_Request	8-26
0048_{hex}	Disable_All_Messages_Request	8-27
0049_{hex}	Receive_Group_Numbers_Request	8-9
$004A_{hex}$	Alarm_Stop_Request	8-18
$004C_{hex}$	Warmstart_Request	8-18
$004D_{hex}$	Set_Event_Message_Type_Request	8-43
$004E_{hex}$	Clear_Display_Request	8-19
0058_{hex}	Check_Physical_Configuration_Map_Request	8-8
0059_{hex}	Bus_Delay_Request	8-19
$005A_{hex}$	Send_Bus_Error_Information_Request	8-20
$005B_{hex}$	Send_Local_Bus_Module_Error_Request	8-21

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 8-1: Kommandos für das IBS-Masterboard

Code	IBS-Kommandos	Seite
$005C_{hex}$	Send_All_Module_Error_Request	8-20
$005D_{hex}$	Receive_Processing_Instructions_Request	8-30
$005E_{hex}$	Send_Physical_Configuration_Request	8-6
$005F_{hex}$	Send_Log_Address_Error_Request	8-16
0060_{hex}	Define_Groups_Error_Characteristic_Request	8-12
0064_{hex}	Quit_Module_Error_Request	8-22
0065_{hex}	Quit_Module_Error_All_Request	8-23
0069_{hex}	Receive_Localbus_Code_Map_Request	8-13
$010D_{hex}$	Send_Actuall_Configuration_Request	8-7
0114_{hex}	Read_Event_Counter_Request	8-44

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8.1 Aufbau einer Kommandobeschreibung

Dieses Kapitel beschreibt die Kommandos für das IBS-Masterboard. Die Kommandos werden wortweise, die Kommando- und Meldungs-Codes hexadezimal dargestellt. Die Beschreibungen sind nach dem folgenden Schema aufgebaut.

Name des Kommandos **Kommando-Code$_{hex}$**

Aufgabe: *Beschreibt die Funktionalität des Kommandos.*

Voraussetzung: *Alle Bedingungen, die für eine erfolgreiche Bearbeitung eines Kommandos vor dessen Aufruf erfüllt sein müssen.*

Syntax: Für ein Kommando ohne Parameter wird nur der Kommando-Code angegeben.

Die Syntax eines Kommandos mit Parametern wird folgendermaßen als Parameterblock dargestellt:

Wort 1	*Kommando-Code*
Wort 2	*Anzahl nachfolgender Worte (Parameterzähler)*
Wort 3	*Parameter 1*
Wort 4	*Parameter 2*
Wort 5	*Parameter 3*
	...
Wort n+2	*Parameter n*

Bit	15 0

Legende: Parameter *Beschreibung der einzelnen Parameter*

positive Quittung: *Meldung, die eine erfolgreiche Durchführung des Kommandos bestätigt.*
Bedeutung: *Erklärung der Meldung.*
Bemerkung: *Weitere Informationen zur Meldung.*

negative Quittung: *Meldung die einen Fehler bei der Durchführung des Kommandos anzeigt.*
Bedeutung: *Erklärung der Fehlerursache.*
Bemerkung: *Weitere Informationen zur Meldung.*

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8.2 Kommandos zur Konfiguration

Configure_Bus_Request **0023_{hex}**

Aufgabe: Dieses Kommando veranlaßt das IBS-Masterboard dazu, die aktuell angeschlossene Buskonfiguration einzulesen und im RAM der Anschaltbaugruppe als *Urkonfiguration* zu speichern.

Voraussetzung: Bei Anwendung dieses Kommandos muß sich der Bus im STOP-Zustand befinden.

Das Kommando darf nicht im laufenden Betrieb der Anlage aufgerufen werden, da es alle auf der Anschaltbaugruppe gespeicherten Definitionen und Listen löscht.

Auswirkung: Alle zuvor auf dem Controllerboard gespeicherten Definitionen und Listen werden gelöscht:
- logische Adreßlisten
- Gruppendefinition
- Prozeßdatenverknüpfung
- Eventdefinition

Die Konfigurierung beschäftigt die Anschaltbaugruppe für eine gewisse Zeit. Dadurch wird nach Absetzen der positiven Quittung eine erneute Kommandoausführung für diese Zeitspanne blockiert. Die Zeit hängt von der Buslänge ab.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code (0023_{hex}). Es folgen keine weiteren Parameter.

positive Quittung: Quit_Configure_Bus_Confirmation ($00CA_{hex}$)

negative Quittung: Bus_System_Error_Indication (0038_{hex})
Bedeutung: Es ist ein Fehler aufgetreten, der keinen weiteren Betrieb zuläßt.
Bemerkung: Die Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$) gibt eine detaillierte Beschreibung des Fehlers.

Send_Physical_Configuration_Request **$005E_{hex}$**

Aufgabe: Das Kommando liest die im RAM der Anschaltbaugruppe als *Urkonfiguration* gespeicherte Buskonfiguration in Form von Längen- und ID-Codes aus. Diese Konfiguration ist dort durch Ausführung der Kommandos *Configure_Bus_Request* (0023_{hex}) oder *Check_Physical_Configuration* (0058_{hex}) hinterlegt worden.

Bemerkung: Falls die aktuell angeschlossene Buskonfiguration ausgelesen werden soll, muß vor Aufruf des Kommandos *Send_Physical_Configuration_Request* das *Kommando Configure_Bus_Request (0023_{hex})* ausgeführt werden.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code 005E($_{hex}$). Es folgen keine weiteren Parameter.

positive Quittung: Send_Physical_Configuration ($80F4_{hex}$)
Bedeutung: Die Quittung übermittelt die im RAM der Anschaltbau-

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

gruppe abgespeicherte Buskonfiguration in Form von Längen- und ID-Codes.

negative Quittung: No_Executable_Configuration ($004A_{hex}$)
Bedeutung: Während des Hardware-Anlaufs (Einschalten der Spannung, Konfiguration oder nach dem letzten Reset) war kein Bussystem angeschlossen.

Send_Actual_Configuration_Request **$010D_{hex}$**

Aufgabe: Dieses Kommando veranlaßt das IBS-Masterboard dazu, die aktuell angeschlossene Buskonfiguration mit der Meldung *Send_Actual_Configuration_Confirmation* (8119_{hex}) in Form von Längen- und ID-Codes zum Host zu übertragen.

Bemerkung: Im Gegensatz zum Kommando *Send_Physical_Configuration_Request* ($05E_{hex}$) wird nicht die im RAM der Anschaltbaugruppe abgespeicherte *Urkonfiguration*, sondern die tatsächlich angeschlossene Konfiguration übertragen.

Falls sich der Bus bei Aufruf des Kommandos *Send_Actual_Configuration_Request* im STOP-Zustand befindet, kann die aktuell angeschlossene Buskonfiguration nicht übertragen werden. Stattdessen wird die zuletzt vor dem Stop angeschlossene Konfiguration übertragen. Nachträglich Veränderungen erkennt das IBS-Masterboard nicht! Erst durch Starten des Busses oder durch ein erneutes Konfigurieren mit dem Kommando *Configure_Bus_Request* (0023_{hex}) oder *Check_Physical_Configuration* (0058_{hex}) werden diese Änderungen berücksichtigt. Falls der Bus aufgrund einer Konfigurationsänderung in den STOP-Zustand geht, so wird die neue (fehlerhafte) Konfiguration übertragen.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($010D_{hex}$). Es folgen keine weiteren Parameter.

Quittung: Send_Actual_Configuration_Confirmation (8119_{hex})
Bedeutung: Die Quittung übermittelt die aktuell angeschlossene Buskonfiguration in Form von Längen- und ID-Codes. Falls bei Aufruf des Kommandos *Send_Actual_Configuration* z.B das Fernbuskabel nicht an der Anschaltbaugruppe angeschlossen erfolgt ebenfalls diese Meldung, allerdings ohne Längen- und ID-Codes.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Check_Physical_Configuration_Map_Request 0058_{hex}

Aufgabe: Das Kommando überträgt eine gewünschte Buskonfiguration in Form von Längen- und ID-Codes auf die Anschaltbaugruppe. Diese gewünschte Buskonfiguration wird zuerst daraufhin überprüft, ob die eingetragenen Parameter prinzipiell zulässig sind (z.B. auf zulässige Längen- oder ID-Codes) und wenn ja, im RAM der Anschaltbaugruppe gespeichert. Danach wird die gewünschte Buskonfiguration mit der aktuell angeschlossenen Buskonfiguration verglichen.

Voraussetzung: Bei Anwendung dieses Kommandos muß sich der Bus im STOP-Zustand befinden.

Syntax:

Wort 1	Code		
Wort 2	Parameterzähler (n)		
Wort 3	Längen-Code	ID-Code	für Teilnehmer Nr. 1
Wort 4	Längen-Code	ID-Code	für Teilnehmer Nr. 2
Wort 5	Längen-Code	ID-Code	für Teilnehmer Nr. 3
	...	...	
Wort n+2	Längen-Code	ID-Code	für Teilnehmer Nr. n
Bit	15 8	7 0	

Legende:

Code: Kommando-Code (hier 0058_{hex})

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).

Längen-Code: Der Längen-Code beschreibt den Adreßraumbedarf des IBS-Teilnehmers im Host.

ID-Code: Identifikations-Code des IBS-Teilnehmers. Er ist als *Module Ident.* den Modulen in dezimaler Form aufgedruckt. (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

Nach Kommandoanwendung müssen Sie alle Adresslisten, Gruppen- und Eventdefinitionen neu laden! Der Längen-Code für PCP-Teilnehmer enthält nur die Anzahl der Prozeßein-/Ausgabedaten; die durch PCP benutzten Daten sind nicht anzugeben. Beispiel: Das Modul *IBSV.24* besitzt nur ein Wort für PCP und keine Prozeßein-/Ausgabedaten. Tragen Sie für das Modul *IBSV.24* somit als Längen-Code eine 0 ein.

positive Quittung: Physical_Configuration_Map_Valid_Confirmation ($00AB_{hex}$)

Bedeutung: Die gewünschte Buskonfiguration ist zulässig und wurde im RAM der Anschaltbaugruppe gespeichert. Sie stimmt außerdem mit der aktuell angeschlossenen Buskonfiguration überein.

negative Quittung: Check_Configuration_Error_Confirmation (0068_{hex})

Bedeutung: Die gewünschte Buskonfiguration ist zulässig und wurde im RAM der Anschaltbaugruppe gespeichert. Sie stimmt allerdings **nicht** mit der aktuell angeschlossenen Buskonfiguration überein.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

negative Quittung: Unknown_Bus_Module ($005B_{hex}$)

Bedeutung: Es wurden unzulässige Längen- oder ID-Codes übertragen;
Es wurde eine unzulässige Anzahl von Prozeßdatenworten angegeben;
Das erste Modul des Busaufbaus ist keine Busklemme.

Bemerkung: Nur in diesem Fall wird die gewünschte Konfiguration nicht im RAM der Anschaltbaugruppe gespeichert.

Receive_Group_Numbers_Request **0049_{hex}**

Aufgabe: Das Kommando fast logisch zusammenhängende Bussegmente zu einer Gruppe zusammen, indem die Gruppennummer zusammen mit den entsprechenden Bussegmentnummern genannt wird. Weisen Sie alle Bussegmenten mit Ausnahme der Segmente in Installationsfernbussen einer Gruppe zu! Im Grundzustand (nach dem Einschalten der Spannung, betätigen des Reset-Tasters oder Aufruf des Kommandos *Configure_Bus_Request* (0023_{hex})) sind alle Bussegmente in Gruppe 0 zusammengefaßt.

Voraussetzung: Bei Anwendung dieses Kommandos muß sich der Bus im STOP-Zustand befinden.

Bemerkung: Die Aufteilung des Busaufbaus in Gruppen ermöglicht es, die vom Hauptstrang abgehenden Bussegmente (Fernbus-Stich, Installationsfernbus, Peripheriebus) ab- bzw. wieder anzuschalten (Kommandos: *Switch_Group_Off_Request* bzw. *Switch_Group_On_Request*). Eine Gruppe kann aus einem oder mehreren Bussegmenten bestehen.

Syntax:

Wort 1	Code		
Wort 2	Parameterzähler (n)		
Wort 3	00_{hex}	Bussegment	für 1. Bussegment
Wort 4	00_{hex}	Gruppe	für 1. Bussegment
Wort 5	00_{hex}	Bussegment	für 2. Bussegment
Wort 6	00_{hex}	Gruppe	für 2. Bussegment
	...	...	
	...	...	
Wort n+1	00_{hex}	Bussegment	für x. Bussegment
Wort n+2	00_{hex}	Gruppe	für x. Bussegment

Bit	15 8	7 0

Legende:

Code: Kommando-Code (hier 0049_{hex})

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Bussegmente multipliziert mit 2)

Bussegment: Nummer des Bussegmentes, das der in der folgenden Zeile benannte Gruppe zugewiesen wird

Gruppe: Nummer der Gruppe, der das in der vorhergehen Zeile benannte Bussegment zugewiesen wird.
(0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Ordnen Sie allen Bussegmenten (BS), mit Ausnahme der Segmente in Stichabzweigen (FB-Ebene 1), eine Gruppennummer von 0 bis 255 zu.

- Ordnen Sie nicht abschaltbaren Fernbusteilnehmern (z.B. Busklemmen mit E/A-Funktion) eine eigene Gruppennummer zu.
- IBS-Teilnehmer ohne Weichenfunktion (kein Peripheriebusanschluß, kein Fernbus-Stich) bekommen ebenfalls eine eigene Gruppennummer!
- Tragen Sie, je nach gewählter Adressierungsart, in den Parameter *Bussegment* die logische oder physikalische Bussegmentnummer ein.
- Bussegmentnummern dürfen nur einmal vorhanden sein!
- Tragen Sie nur IBS-Teilnehmer ein, die sich im Hauptstrang des Busaufbaus (FB-Ebene 0) befinden. Teilnehmer im Fernbus-Stich oder Peripheriebus (FB-Ebene 1) werden in die Gruppendefinition einbezogen, indem die Busklemme, welche diese Stichleitung eröffnet, einer Gruppe zugewiesen wird.

positive Quittung: Quit_Receive_Group_Numbers_Confirmation ($00BD_{hex}$)

negative Quittung: Receive_Group_Numbers_Failed_Confirmation (0055_{hex})
Bedeutung: Bei der Überprüfung der Liste wurde ein Fehler erkannt.

Switch_Group_Off_Request **0021_{hex}**

Aufgabe: Das Kommando schaltet eine zuvor definierte Gruppe von Bussegmenten ab. Die Ausgänge der sich im abgeschalteten Bussegment befindenden IBS-Teilnehmer und die zugehörigen Eingänge des Steuerungs bzw. Rechnersystems werden auf 0 gesetzt.

Voraussetzung: Vor Aufruf dieses Kommandos muß die Gruppendefinition (*Receive_Group_Numbers_Request* 0049_{hex}) erfolgt sein.

Die Befehle von Prozeßdatenverknüpfungen, die mit Adressen innerhalb der abgeschalteten Gruppen arbeiten sind nicht mehr aktiv. Stellen Sie durch das Anwendungsprogramm sicher, daß der Anlagenzustand dies zuläßt. Die Eingangsprozeßdaten werden auf Null gesetzt.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Gruppennummer

Bit	15 8	7 0

Legende:
Code: Kommando-Code (hier 0021_{hex})
Parameterzähler: Anzahl der nachfolgenden Worte (hier 1)
Gruppennummer: Nummer der Gruppe, die abgeschaltet werden soll. (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

Beim Abschalten eines Anlagenteils muß für die darin enthaltenen IBS-Teilnehmer mit Busklemmenfunktionalität die Versorgungsspannung für die Modulelektronik (Logikspannung) erhalten bleiben, damit der restliche Busaufbau weiter betrieben werden kann.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

positive Quittung: Switch_Group_Off_Confirmation ($809D_{hex}$)

negative Quittung: Not_Expected_Group_Number (0059_{hex})
Bedeutung: Unbekannte oder unzulässige Gruppennummern; Versuch ein nicht abschaltbares Modul abzuschalten

Switch_Group_On_Request **0020_{hex}**

Aufgabe: Das Kommando schaltet eine zuvor definierte Gruppe von Bussegmenten ein. Dabei wird keine Überprüfung der zugeschalteten Segmente vorgenommen.

Voraussetzung: Vor Aufruf dieses Kommandos muß die Gruppendefinition (*Receive_Group_Numbers_Request* (0049_{hex}) erfolgt sein.

Die in der Bitmanipulation benutzten Prozeßausgänge werden beim Einschalten einer Gruppe wieder in die Prozeßdatenübertragung aufgenommen. Stellen Sie durch das Anwendungsprogramm sicher, daß diese Daten in den korrekten Zustand gesetzt werden.

positive Quittung: Switch_Group_On_Confirmation ($809E_{hex}$)
Bemerkung: Das Bussystem läuft mit den zugeschalteten Gruppen weiter. Falls eine zugeschaltete Gruppe defekt ist, können die Fehlermeldungen *Bus_System_Error_Indication* (0038_{hex}) und *Bus_Error_Information_Indication* ($80C4_{hex}$) folgen.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Gruppennummer

Bit	15 8	7 0

Legende:
Code: Kommando-Code (hier 0020_{hex})
Parameterzähler: Anzahl der nachfolgenden Worte (hier 1)
Gruppennummer: Nummer der Gruppe, die eingeschaltet werden soll.

negative Quittung: Not_Expected_Group_Number_Indication (0059_{hex})
Bedeutung: Es wurde eine unbekannte Gruppennummer benutzt, oder die Nummer liegt nicht im zulässigen Bereich.

negative Quittung: Switch_Group_On_Failed_Confirmation ($80C5_{hex}$)
Bedeutung: Die Gruppe kann nicht zugeschaltet werden, weil die Konfiguration geändert wurde oder Module im Hauptstrang (FB-Ebene 0) fehlen.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Define_Groups_Error_Characteristic_Request 0060_{hex}

Aufgabe: Falls in einer Gruppe ein Fehler auftritt, der den Weiterbetrieb des Busses nicht zuläßt, geht der Bus in der RESET. Mit dem Kommando *Define_Group_Error_Characteristic_Request* legen Sie für jede Gruppe fest, ob der Bus nach dem RESET ohne diese Gruppe wieder gestartet werden soll.

Voraussetzung: Vor Aufruf dieses Kommandos muß mit dem Kommando (*Receive_Group_Numbers_Request* (0049_{hex}) die Gruppendefinition erfolgt sein.

Syntax:

Wort 1	Code			
Wort 2	Parameterzähler			
Wort 3	Mode	0_{hex}	Gruppennummer	für Gruppe Nr. 1
Wort 4	Mode	0_{hex}	Gruppennummer	für Gruppe Nr. 2
	...	...	...	
Wort n+2	Mode	0_{hex}	Gruppennummer	für Gruppe Nr. n

Bit	15 12	11 8	7 0

Legende:

Code: Kommando-Code (hier 0060_{hex})

Parameterzähler: Anzahl der nachfolgenden Worte

Mode: Handhabung der durch diese Zeile beschriebenen Gruppe im Fehlerfall:

- 0_{hex} Der gesamte Bus bleibt nach einem durch diese Gruppe verursachten RESET im STOP-Zustand.
- 8_{hex} Der Bus wird nach dem RESET ohne diese Gruppe wieder gestartet.

Gruppennummer: Gruppennummer der durch diese Zeile beschriebenen Gruppe (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

Sie müssen nicht alle Gruppennummern angeben. Falls ein Fehler in einer Gruppe auftritt, die Sie nicht angegeben haben, bleibt der gesamte Bus nach dem RESET im STOP-Zustand.

Die Anschaltbaugruppe stellt dem Host für die Dauer der Fehlerdiagnose keine neuen Eingangsdaten zur Verfügung. Die Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$) gibt eine detaillierte Beschreibung des Fehlers.

positive Quittung: Quit_Groups_Error_Characteristic_Confirmation ($00F6_{hex}$)

negative Quittung: Groups_Error_Characteristic_Failed_Confirmation ($80F7_{hex}$)

Bemerkung: Bei negativer Quittung sind die mit diesem Kommando zuvor getätigten und positiv bestätigten Definitionen weiterhin gültig!

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8.3 Kommandos zur Adressierung

Receive_Localbus_Code_Map_Request **0069_{hex}**

Aufgabe: Das Kommando nimmt die wahlfreie Zuordnung der Bussegmentnummer zu den InterBus-S-Teilnehmern mit Busklemmenfunktionalität (Fernbus- und Installationsfernbusteilnehmer) vor. Dies ist sinnvoll, um bei Erweiterungen der Anlage nicht die Numerierung der Bussegmente in der bestehenden Anlage ändern zu müssen.

Voraussetzung: keine

Syntax:

Wort 1	Code			
Wort 2	Parameterzähler			
Wort 3	FB-Ebene	0_{hex}	Bussegmentnummer	Teilnehmer Nr. 1
Wort 4	FB-Ebene	0_{hex}	Bussegmentnummer	Teilnehmer Nr. 2
	...	...	...	
Wort n+2	FB-Ebene	0_{hex}	Bussegmentnummer	Teilnehmer Nr. n

Bit	15 12	11 8	7 0

Legende:

Code: Kommando-Code (hier 0069_{hex})

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Bussegmente).

FB-Ebene: Tragen Sie hier die FB-Ebene ein:
- Hauptstrang: FB-Ebene = 0_{hex} (0000_{bin})
- Fernbus-Stich (z.B. Abzweig von IBS IP CBK): FB-Ebene = 1_{hex} (0001_{bin})

Bussegmentnummer: Tragen Sie hier in der Reihenfolge der physikalischen Anordnung die gewünschten logischen Bussegmentnummern ein (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex}). Jede Adresse darf nur einmal vergeben werden!

Die Liste wird erst nach erfolgreicher Ausführung des Kommandos *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) gültig.

positive Quittung: Quit_Receive_Localbus_Code_Map_Confirmation (0105_{hex})

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Receive_Logical_IN_Address_Map_Request 003A$_{hex}$

Aufgabe: Das Kommando überträgt die Liste der logischen Eingangsadressen zur Anschaltbaugruppe.

Syntax:

Wort 1	Code			
Wort 2	Parameterzähler			
Wort 3	DC	0000_{bin}	Eingangsadresse	Teilnehmer Nr. 1
Wort 4	DC	0000_{bin}	Eingangsadresse	Teilnehmer Nr. 2
	...	...	...	
Wort n+2	DC	0000_{bin}	Eingangsadresse	Teilnehmer Nr. n

Bit	15 14	13 9	9 0

Legende:

Code: Kommando-Code (hier 003A$_{hex}$)

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).

Eingangsadresse: Tragen Sie hier (10 Bit) für alle Module mit Prozeßeingangsdaten die gewünschte Adresse (als Byteadresse) ein. Bei Modulen mit mehr als einem Prozeßeingangswort (z.B. mit 32 Bit), werden automatisch die nächsthöheren Adressen mit belegt.

DC: Tragen Sie hier die Datenkonsistenz für den Prozeßdatenzugriff ein. Die Datenkonsistenz garantiert, daß die angegebene Datenbreite aus einem IBS-Zyklus stammt. Der Standardwert für E/A-Module ist 16 Bit (00_{bin}). Für IBS-Teilnehmer, welche die zusammenhängende Übertragung größerer Datenbreiten verlangen, müssen Sie die Datenkonsistenz erhöhen. Dies sind z.B. Encoder, Bedienterminals oder Analogmodule mit mehr als 16 Bit Auflösung. Für die Datenkonsistenz sind folgende Werte zulässig:

- 00_{bin} 16 Bit (Standard)
- 01_{bin} 32 Bit (z.B. Encoder, Bedienterminals)
- 10_{bin} 8 Bit (nur für Module mit Längencode 81_{hex}, auf die Bytezugriffe erfolgen sollen!)
- 11_{bin} 48 Bit (z.B. Encoder, Bedienterminals)

Tragen Sie für alle Module ohne Prozeßeingangsdaten (z.B. reine Busklemmen oder reine Ausgangsmodule) den Eintrag 0000_{hex} ein.

Legen Sie die Startadressen von IBS-Teilnehmern mit 16 Bit (oder mehr) Adreßraum nur auf gerade Adressen.

positive Quittung: Quit_Receive_Logical_In Address_Map_ Confirmation ($00D1_{hex}$)

Die Liste wird erst nach erfolgreicher Ausführung des Kommandos *Implement_All_Logical_Address_Maps_Request* (0040_{hex}) gültig.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Receive_Logical_OUT_Address_Map_Request 003B$_{hex}$

Aufgabe: Das Kommando überträgt die Liste der logischen Ausgangsadressen zur Anschaltbaugruppe.

Syntax:

Wort 1	Code			
Wort 2	Parameterzähler			
Wort 3	DC	0000_{bin}	Ausgangsadresse	Teilnehmer Nr. 1
Wort 4	DC	0000_{bin}	Ausgangsadresse	Teilnehmer Nr. 2
	...	...	...	
Wort n+2	DC	0000_{bin}	Ausgangsadresse	Teilnehmer Nr. n

Bit	15 14	13 9	9 0

Legende:

Code: Kommando-Code (hier 003B$_{hex}$)

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).

Ausgangsadresse: Tragen Sie hier (10 Bit) für alle Module mit Prozeßausgangsdaten die gewünschte Adresse (als Byteadresse) ein. Bei Modulen mit mehr als einem Prozeßausgangswort (z.B. mit 32 Bit), werden automatisch die nächsthöheren Adressen mit belegt.

DC: Siehe Parameterbeschreibung zum Kommando *Receive_Logical_IN_Address_Map_Request* (003A$_{hex}$)

Tragen Sie für alle Module ohne Prozeßausgangsdaten (z.B. reine Busklemmen oder reine Eingangsmodule) den Eintrag 0000$_{hex}$ ein.

Legen Sie die Startadressen von IBS-Teilnehmern mit 16 Bit (oder mehr) Adreßraum nur auf gerade Adressen.

positive Quittung: Quit_Receive_Logical_Out Address_Map_ Confirmation (00D2$_{hex}$)

Die Liste wird erst nach erfolgreicher Ausführung des Kommandos *Implement_All_Logical_Address_Maps_Request* (0040$_{hex}$) gültig.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Implement_All_Logical_Address_Maps_Request **0040_{hex}**

Aufgabe: Die Anschaltbaugruppe überprüft die zuvor übertragenen Adreßlisten und übernimmt Sie, falls sie fehlerfrei sind.

Das Kommando darf nur benutzt werden, wenn der Datenzyklus des Busses nicht betrieben wird! Die zuvor gesendeten Listen werden geprüft und, wenn fehlerfrei, übernommen. (Im Fehlerfall können Sie mit dem Kommando *Send_Log_Address_Error_Request* ($005F_{hex}$) die Fehlerursachen abfragen.)

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code $0040(_{hex})$. Es folgen keine weiteren Parameter.

positive Quittung: Quit_Implement_Confirmation ($00D3_{hex}$)
Bedeutung: Die Adressen sind gültig und übernommen.

negative Quittung: Logical_Address_Error_Confirmation ($002B_{hex}$)
Bedeutung: In den zu implementierenden Listen (logische Adressen) stehen fehlerhafte Werte. Der Bus ist im STOP-Zustand. Es erfolgt **kein** Schalten der Prozeßausgangsdaten in einen definierten, sicheren Zustand!

Send_Log_Address_Error_Request **$005F_{hex}$**

Aufgabe: Anforderung des Ergebnisses einer vorhergehenden Ausführung der Kommandos:
- Implement_All_Logical_Address_Maps_Request (0040_{hex})
- Check_Physical_Configuration_Request (0058_{hex})

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($005F_{hex}$). Es folgen keine weiteren Parameter.

positive Quittung: Send_Log_Address_Error_Confirmation ($80F5_{hex}$)
Bedeutung: Das Ergebnis wird über diese Meldung bekanntgegeben (siehe Kapitel 9).

8.4 Kommandos für den Betrieb

Start_Bus_Cycle_Request **0001_{hex}**

Aufgabe: Das Kommando aktiviert den zyklischen Datenverkehr auf dem Bus. Die Anschaltbaugruppe nimmt den Busaufbau in Betrieb und stellt im MPM die E/A-Daten zur Verfügung.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code (0001_{hex}). Es folgen keine weiteren Parameter.

positive Quittung: Start_Bus_Confirmation (0088_{hex})
Bedeutung: Das System konnte gestartet werden. Die Anschaltbaugruppe beginnt mit dem zyklischen Einlesen und Setzen der Ein- und Ausgänge (Prozeßdaten- und eventuell PCP-Betrieb).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Bemerkung: Die LED *RUN Masterboard* leuchtet.

negative Quittung: Start_Bus_Not_Possible_Confirmation ($00E3_{hex}$)
Bedeutung: Überschreiten der zulässigen, maximal möglichen Konfiguration des Busses,
Alle Gruppen sind abgeschaltet,
Hardwarefehler der Anschaltbaugruppe,
Bemerkung: Die LED *RUN Masterboard* leuchtet nicht.

negative Quittung: Bus_System_Error_Indication (0038_{hex})
Bedeutung: Es ist ein Fehler aufgetreten, der keinen weiteren Betrieb zuläßt.
Bemerkung: Die Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$) gibt eine detaillierte Beschreibung des Fehlers.

Bei der Fehlermeldung *Module_Error_Indication* ($80A0_{hex}$), die nach der positiven oder negativen Quittung auftreten kann, wird der Datenverkehr nicht abgebrochen. Bei Änderung der Konfiguration des Busses vor dem Starten des Datenverkehrs wird zunächst der Datenverkehr gestartet und sofort danach abgebrochen.

Stop_Bus_Cycle_Request **0002_{hex}**

Aufgabe: Das Kommando schaltet den Bus in den STOP-Zustand.

Folgen: Prozeßdatenkanal: Der zyklische Datenverkehr auf dem Bus wird gestoppt. Das bestehende Prozeßdaten-Abbild der am Bus angeschlossenen Module bleibt statisch erhalten und wird nicht erneuert.
PCP-Kanal: Das Peripherals-Communication-Protocol (PCP) wird nicht weiter betrieben. Es erfolgt **kein** automatischer Abbau der hergestellten Verbindungen. Eventuell ausstehende Dienste werden nach dem erneuten Starten des Datenverkehrs weiter bearbeitet.

Das Kommando schaltet die Prozeßausgabedaten **nicht** in den sicheren Zustand (Rücksetzen der Ausgänge). Dies ist nur mit dem Kommando *Alarm_Stop_Request* ($004A_{hex}$) möglich.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code (0002_{hex}). Es folgen keine weiteren Parameter.

positive Quittung: Stop_Bus_Confirmation ($00C6_{hex}$)
Bedeutung: Der Prozeßdaten- und der eventuell betriebene Kommunikations-Kanal (PCP) wird nicht weiter betrieben.
Bemerkung: Die LED *RUN Masterboard* verlischt.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Alarm_Stop_Request **$004A_{hex}$**

Aufgabe: Das Kommando löst einen Reset aus und schaltet den Bus in den STOP-Zustand. Die Ausführung des Kommandos erfolgt direkt nach der Beendigung des laufenden Datenzyklusses.

Folgen:
- Prozeßdatenkanal: Der zyklische Datenverkehr des Busses wird gestoppt. Das bestehende Prozeßdaten-Abbild der am Bus angeschlossenen Module (Eingänge des Steuerungs bzw. Rechnersystems) wird auf 0 gesetzt. Das Kommando schaltet die Prozeßausgabedaten (Ausgänge der IBS-Teilnehmer) in den sicheren Zustand (auf 0).
- PCP-Kanal: Das Peripherals-Communication-Protocol (PCP) wird nicht weiter betrieben. Es erfolgt **kein** automatischer Abbau der hergestellten Verbindungen. Eventuell ausstehende Dienste werden nach dem erneuten Starten des Datenverkehrs weiter bearbeitet.

weiterer Betrieb: folgende Definitionen und Listen werden auf dem Controllerboard gelöscht:
- logische Adreßlisten
- Gruppendefinition
- Prozeßdatenverknüpfung
- Eventdefinition

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($004A_{hex}$). Es folgen keine weiteren Parameter.

positive Quittung: Quit_Alarm_Stop_Confirmation ($00D8_{hex}$)
Bedeutung: Der Datenverkehr des Busses ist gestoppt.
Bemerkung: Die Parameterlisten und die LED *Run* werden gelöscht.

Warmstart_Request **$004C_{hex}$**

Aufgabe: Das Kommando führt einen Warmstart der Firmware aus. Nach dem Hardware-Anlauf wird die Hard- und Software der Anschaltbaugruppe initialisiert. Alle zuvor vorgenommenen Einstellungen werden ungültig und auf den Stand gesetzt, der auch nach dem Betätigen des Reset-Tasters erreicht wird.

Folgen: Auf dem Bus wird ein Reset ausgelöst. Dies führt bei Modulen mit Prozeßdaten zum Rücksetzen der Prozeßdaten auf den Wert 0. Der Datenverkehr des Busses wird gestoppt.

Die Anschaltbaugruppe ist erst wieder betriebsbereit, wenn die LED *Ready Masterboard* leuchtet!

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($004C_{hex}$). Es folgen keine weiteren Parameter.

Quittung: Die Ausführung des Kommandos wird nicht quittiert. Die LED *Ready Masterboard* wird zunächst ausgeschaltet und nach der Beendigung des Warmstarts wieder eingeschaltet.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Bus_Delay_Request **0059_{hex}**

Aufgabe: Bei Betrieb des Bussystems mit Kommunikation (PCP-Kanal) ist zwischen den Datenzyklen eine Verzögerungszeit von 1ms eingebaut. Die Zeit wird bei der Initialisierung der Kommunikationsschichten mit dem Kommando *Init_Comm_Service_Request* (0054_{hex}) eingeschaltet. Wenn sich keine Kommunikationsteilnehmer im Bussystem befinden, so ist die Verzögerung automatisch abgeschaltet. Mit dem Kommando *Bus_Delay_Request* (0059_{hex}) kann die Verzögerungszeit auch für den Betrieb mit Kommunikation abgeschaltet werden.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code $0059(_{hex})$. Es folgen keine weiteren Parameter.

Quittung: Quit_Bus_Delay_Confirmation ($00EC_{hex}$).

8.5 Kommandos zur Fehlerbehandlung

Clear_Display_Request **$004E_{hex}$**

Aufgabe: Das Kommando löscht die Statusanzeige auf der Frontblende der Anschaltbaugruppe. Die Einträge im Diagnosebitregister und im Diagnoseparameterregister werden ebenfalls zurückgesetzt.

Voraussetzung: keine

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($004E_{hex}$). Es folgen keine weiteren Parameter.

positive Quittung: Quit_Clear_Display_Confirmation ($00E2_{hex}$)

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Send_Bus_Error_Information_Request **$005A_{hex}$**

Aufgabe: Das Kommando fordert die Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$) an. Diese Meldung überträgt aus dem RAM der Anschaltbaugruppe eine Liste mit Informationen zum Fehlerzustand des Busses (Bus-Error-Information-Map). Die Liste wurde beim letzten Auftreten eines Fernbus- oder Peripheriebusfehlers erstellt.

Voraussetzung: Benutzen Sie dieses Kommando nur nach Auftreten der Fehlermeldung *Bus_System_Error_Indication* (0038_{hex}). Es wird nicht überprüft, ob sich der Fehlerzustand des Busses seit Erstellung der *Bus-Error-Information-Map* schon wieder geändert hat (z.B bereits erfolgte Fehlerbehebung durch den Anwender), und somit der Inhalt der letzten *Bus-Error-Information-Map* nicht mehr aktuell ist.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($005A_{hex}$). Es folgen keine weiteren Parameter.

positive Quittung: Bus_Error_Information_Indication ($80C4_{hex}$)
Bedeutung: Liste mit Informationen zum Fehlerzustand des Busses (Bus-Error-Information-Map)

negative Quittung: No_Map_Entry_Confirmation ($00ED_{hex}$)
Bedeutung: Es ist kein Eintrag vorhanden.

Send_All_Module_Error_Request **$005C_{hex}$**

Aufgabe: Das Kommando überprüft bei aktivem Bussystem alle Module auf anstehende Fehlermeldungen und aktualisiert die Fehleranzeige der Modulfehler in der Diagnoseschnittstelle.

Voraussetzung: Modulfehler

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($005C_{hex}$). Es folgen keine weiteren Parameter.

positive Quittung: Send_All_Module_Error_Confirmation ($80EF_{hex}$)
Bedeutung: Mit dieser Quittung wird eine Liste aller Bussegmente mit anstehenden Modulfehlern übermittelt.

negative Quittung: No_Map_Entry_Confirmation ($00ED_{hex}$)
Bedeutung: Es stehen keine Modulfehler an. Die Fehleranzeigen in der Diagnoseschnittstelle werden zurückgesetzt.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Send_Local_Bus_Module_Error_Request 005B$_{hex}$

Aufgabe: Das Kommando sucht in dem angegebenen Peripheriebus nach Modulen, die einen Modulfehler melden.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Bussegment

Bit	15 8	7 0

Legende:

Code: Kommando-Code (hier 005B$_{hex}$)
Parameterzähler: Anzahl der nachfolgenden Worte (hier 1).
Bussegment: Tragen Sie hier die Bussegmentnummer des Peripheriebusses ein, in dem nach Modulen mit Modulfehler gesucht werden soll.

positive Quittung: Local_Bus_Module_Error_Confirmation (80EE$_{hex}$)
Bedeutung: Eine Liste mit Position und ID-Codes aller Module, bei denen ein Modulfehler aufgetreten ist. Die Position ist die physikalische Platznummer im angegebenen Peripheriebus (siehe auch Kapitel 9):
- Busklemme: 0
- Peripheriebusteilnehmer: 1 bis 8

negative Quittung: No_Map_Entry_Confirmation (00ED$_{hex}$)
Bedeutung: Es wurden im angegebenen Peripheriebus keine Modulfehler festgestellt.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Quit_Module_Error_Request 0064_{hex}

Aufgabe: IBS-Teilnehmer mit speichernder Anzeige zeigen einen Modulfehler auch nach dessen Behebung weiterhin an. Die Fehlermeldung muß deshalb quittiert werden. Das Kommando quittiert die Modulfehler der angegebenen IBS-Teilnehmer. Geben Sie für jeden zu quittierenden IBS-Teilnehmer die Bussegmentnummer und die Position im Peripheriebus an.

Voraussetzung: Modulfehler auf einem IBS-Teilnehmer mit speichernder Anzeige.

Syntax:

Wort 1	Code		
Wort 2	Parameterzähler		
Wort 3	00_{hex}	Bussegment	Teilnehmer Nr. 1
Wort 4	00_{hex}	Position	Teilnehmer Nr. 1
Wort 5	00_{hex}	Bussegment	Teilnehmer Nr. 2
Wort 6	00_{hex}	Position	Teilnehmer Nr. 2
	...	...	
	...	...	
Wort n+1	00_{hex}	Bussegment	Teilnehmer Nr. x
Wort n+2	00_{hex}	Position	Teilnehmer Nr. x
Bit	15 8	7 0	

Legende:

Code: Kommando-Code (hier 0064_{hex})

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl (x) der Teilnehmer, für die eine Modulfehlermeldung quittiert werden soll, multipliziert mit 2).

Bussegment: Logische Bussegmentnummer (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

Position: Physikalische Platznummer im durch das vorhergehende Wort ausgewählten Bussegment:

- Busklemmen, reine Fernbusteilnehmer und Installationsfernbusteilnehmer: 0;
- Peripheriebusteilnehmer: 1 bis 8

positive Quittung: Quit_Module_Error_Ok_Confirmation ($00FE_{hex}$)

Bedeutung: Das Kommando wurde erfolgreich ausgeführt.

Bemerkung: Auch die Fehler-LEDs auf dem IBS-Teilnehmer wurden gelöscht.

negative Quittung: Quit_Module_Error_Not_Possible_Confirmation ($80FF_{hex}$)

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Quit_Module_Error_All_Request 0065$_{hex}$

Aufgabe: Dieses Kommando quittiert die Modulfehlermeldungen aller IBS-Teilnehmer mit speichernder Anzeige in der aktuell erreichbaren Buskonfiguration.

Voraussetzung: Modulfehler auf IBS-Teilnehmern mit speichernder Anzeige

Bemerkung: Auch die Fehler-LEDs auf den IBS-Teilnehmer werden gelöscht.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code (0065$_{hex}$). Es folgen keine weiteren Parameter.

positive Quittung: Quit_Module_Error_Ok_Confirmation (00FE$_{hex}$)
Bedeutung: Das Kommando wurde erfolgreich ausgeführt.

negative Quittung: Quit_Module_Error_Not_Possible_Confirmation (80FF$_{hex}$)
Bedeutung: Es wurde nicht für alle erreichbaren IBS-Teilnehmer mit speichernder Anzeige die Modulfehlermeldung quittiert.
Bemerkung: Die Meldung enthält für jeden IBS-Teilnehmer dessen Modulfehler nicht Quittiert werden konnte zwei Parameter, welche die Ursache näher beschreiben (siehe Beschreibung der Meldung (80FF$_{hex}$).

Set_BK_Alarm_Logical_Request 0024$_{hex}$

Aufgabe: Das Kommando setzt den Alarmausgang einer Busklemme.

Voraussetzung: Auf der angesprochenen Busklemme muß ein Alarmausgang vorhanden sein.

Wenn das angesprochene Modul keinen Alarmausgang besitzt, wird die positive Quittung generiert!

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00$_{hex}$	log. Bussegment

Bit	15 8	7 0

Legende:
Code: Kommando-Code (hier 0024$_{hex}$)
Parameterzähler: Anzahl der nachfolgenden Worte (hier 1).
log. Bussegment: Tragen Sie hier die logische Bussegmentnummer (0 bis 255$_{dez}$ entspricht 00$_{hex}$ bis FF$_{hex}$) der Busklemme ein, deren Alarmausgang Sie setzen möchten.

positive Quittung: Quit_Set_BK_Alarm_Logical_Confirmation (00CB$_{hex}$)

negative Quittung: BK_Alarm_Failed_Confirmation (005A$_{hex}$)
Bedeutung: Ungültige Peripheriebusadresse.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Reset_BK_Alarm_Logical_Request 0025_{hex}

Aufgabe: Das Kommando setzt den Alarmausgang einer Busklemme zurück.

Voraussetzung: Auf der angesprochenen Busklemme muß ein Alarmausgang vorhanden und gesetzt sein.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	log. Bussegment

Bit	15 8	7 0

Legende:

Code:	Kommando-Code (hier 0025_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1).
log. Bussegment:	Tragen Sie hier die logische Bussegmentnummer (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})der Busklemme ein, deren Alarmausgang Sie zurücksetzen möchten.

positive Quittung: Quit_Reset_BK_Alarm_Logical_Confirmation ($00CC_{hex}$)

negative Quittung: BK_Alarm_Failed_Confirmation ($005A_{hex}$)
Bedeutung: Ungültige Peripheriebusadresse oder zu viele Parameter.

Set_BK_Alarm_Physical_Request 0026_{hex}

Aufgabe: Das Kommando setzt den Alarmausgang einer Busklemme.

Voraussetzung: Auf der angesprochenen Busklemme muß ein Alarmausgang vorhanden sein.

Wenn das angesprochene Modul keinen Alarmausgang besitzt, wird die positive Quittung generiert!

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	phys. Bussegment

Bit	15 8	7 0

Legende:

Code:	Kommando-Code (hier 0026_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1).
phys. Bussegment:	Tragen Sie hier die physikalische Bussegmentnummer (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex}) der Busklemme ein, deren Alarmausgang Sie setzen möchten.

positive Quittung: Quit_Set_BK_Alarm_Physical_Confirmation ($00CD_{hex}$)

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

negative Quittung: BK_Alarm_Failed_Confirmation ($005A_{hex}$)
Bedeutung: Ungültige Peripheriebusadresse.

Reset_BK_Alarm_Physical_Request **0027_{hex}**

Aufgabe: Das Kommando setzt den Alarmausgang einer Busklemme zurück.

Voraussetzung: Auf der angesprochenen Busklemme muß ein Alarmausgang vorhanden und gesetzt sein.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	phys. Bussegment

Bit	15 8	7 0

Legende:
Code: Kommando-Code (hier 0027_{hex})
Parameterzähler: Anzahl der nachfolgenden Worte (hier 1).
phys. Bussegment: Tragen Sie hier die physikalische Bussegmentnummer (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex}) der Busklemme ein, deren Alarmausgang Sie zurücksetzen möchten.

positive Quittung: Quit_Reset_BK_Alarm_Physical_Confirmation ($00CE_{hex}$)

negative Quittung: BK_Alarm_Failed_Confirmation ($005A_{hex}$)
Bedeutung: Ungültige Peripheriebusadresse oder zu viele Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8.6 Kommandos für die Anwendungsschnittstelle

Set_Parameter_Timeout_Constant_Request **0034_{hex}**

Aufgabe: Dieses Kommando legt die Zeitkonstante für den Kommando- und Meldungs-Timeout fest.

Der Austausch von Kommandos und Meldungen zwischen Steuerungsrechner und Anschaltbaugruppe wird von der Anschaltbaugruppe durch einen Timeout überwacht. Wenn eine Meldung innerhalb der Timeout-Zeit nicht abgeholt wird oder der interne Meldungsspeicher überläuft, generiert die Anschaltbaugruppe die Fehlermeldung *Controller_Parameter_Timeout_Indication* ($000C_{hex}$) ohne Berücksichtigung des Übertragungsprotokolls gemeldet.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	reserviert	
Wort 4	00_{hex}	Timeout
Bit	15 8	7 0

Legende:

Code:	Kommando-Code (hier 0034_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 2)
reserviert:	Dieser Parameter ist für spätere Erweiterungen reserviert. Tragen Sie den Wert 0000_{hex} ein.
Timeout:	Tragen Sie hier die gewünschte Timeout-Zeit in Millisekunden ein.

positive Quittung: Quit_Set_Parameter_Timeout_Confirmation ($00CF_{hex}$)

Enable_All_Messages_Request **0047_{hex}**

Aufgabe: Das Kommando gibt die Übertragung von Meldungen von der Anschaltbaugruppe zum Steuerungsrechner frei.

Die Übertragung von Meldungen ist solange freigegeben, bis sie durch den Anwender gesperrt wird. Nach RESET ist sie standardmäßig gesperrt!

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code (0047_{hex}). Es folgen keine weiteren Parameter.

positive Quittung: Quit_Enable_All_Messages_Confirmation ($00D6_{hex}$)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Disable_All_Messages_Request **0048_{hex}**

Aufgabe: Das Kommando sperrt die Übertragung von Meldungen von der Anschaltbaugruppe zum Steuerungsrechner und leitet sie auf die Diagnose-Schnittstelle (V24) um.

Das Kommando *Enable_All_Messages_Request* (0047_{hex}) macht Umleitung wieder rückgängig.

Syntax: Das Kommando *Disable_All_Messages_Request* besteht nur aus einem Wort, dem Kommando-Code (0048_{hex}). Es folgen keine weiteren Parameter.

positive Quittung: Quit_Disable_All_Messages_Confirmation (00D6)

8.7 Kommandos zur Systemkontrolle

Send_Software_Revision_Request **0008_{hex}**

Aufgabe: Das Kommando fragt nach der Firmware-Version auf dem IBS-Master.

Voraussetzung: keine

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code (0008_{hex}). Es folgen keine weiteren Parameter.

positive Quittung: Software_Revision_Confirmation (8087_{hex})

Send_Bus_Cycle_Counter_Request **0006_{hex}**

Aufgabe: Auf der Anschaltbaugruppe läuft intern ein 32 Bit breiter Zähler, der fortlaufend die Datenzyklen des InterBus-S zählt. Dieser Zähler wird beim Hochlaufen der Anschaltbaugruppe auf 0 gesetzt. Das Kommando liest den aktuellen Zählerstand aus.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code (0006_{hex}). Es folgen keine weiteren Parameter.

positive Quittung: Bus_Cycle_Counter_Confirmation
Bedeutung: Die Meldung gibt den aktuellen Zählerstand an.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8.8 Kommandos zur Prozeßdatenverknüpfung

In einigen Anwendungen ist es sinnvoll, auf Signalwechsel von Eingängen schneller zu reagieren als es das Steuerungs- oder Rechnersystem (Host) vermag. Die Reaktionszeit des Hosts auf einen Signalwechsel ist abhängig von der Zykluszeit des Anwendungsprogramms. Die Prozeßdatenverknüpfung bietet die Möglichkeit, schon auf Signalwechsel an Eingängen zu reagieren, bevor diese in das Prozeßabbild des Hosts aufgenommen werden. Wenn die Anschaltbaugruppe den Signalwechsel eines Eingangs erkennt, kann sie

- diesen wie üblich für die weitere Verarbeitung zum Host durchreichen, und
- unter Verwendung eines Befehls der Prozeßdatenverknüpfung sofort eine direkte Reaktion auf einen Ausgang bewirken.

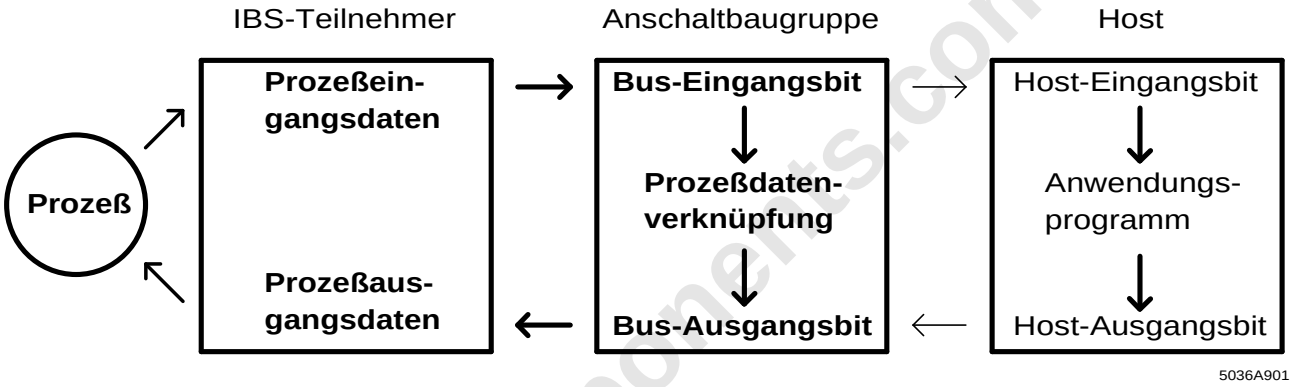

Bild 8-1: Schnelle Reaktion über die Prozeßdatenverknüpfung

online-components.com

Verwenden Sie im Zusammenhang mit Prozeßdatenverknüpfung möglichst die logische Adressierung. Zwar läßt sich die Prozeßdatenverknüpfung auch unter physikalischer Adressierung betreiben, bei späteren Erweiterungen ihrer Anlage müssen Sie dann aber in sämtlichen Zuordnungslisten alle Adreßangaben anpassen. Bei logischer Adressierung müssen Sie die Zuordnungslisten nur um die neu hinzugekommenen IBS-Teilnehmer erweitern.

Allgemeine Funktion der Befehle zur Prozeßdatenverknüpfung

Die Befehle steuern bestimmbare Bits der Prozeßausgangsdaten (Bus-Ausgangsbit) unter Berücksichtigung bestimmbarer Bits der Prozeßeingangsdaten (Bus-Eingangsbit) und der vom Host geschriebenen Ausgabedaten (Host-Ausgangsbit).

Die Darstellung des Bits im Bus-Eingangsworts in Bild 8-2 gilt:
a) für Befehle, die das Rücksetzen des Bus-Ausgangsbits bei Auftreten der positiven Flanke des Bus-Eingangsbits bewirken.
b) für Befehle, die das Rücksetzen des Bus-Ausgangsbits bei Auftreten der negativen Flanke des Bus-Eingangsbits bewirken.

Das Bit im Bitmanipulationsregister ist nur bei den Befehlen zur bedingten Prozeßdatenverknüpfung (*Reset_Out_In01_Switch* und *Reset_Out_In10_Switch*) von Bedeutung.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

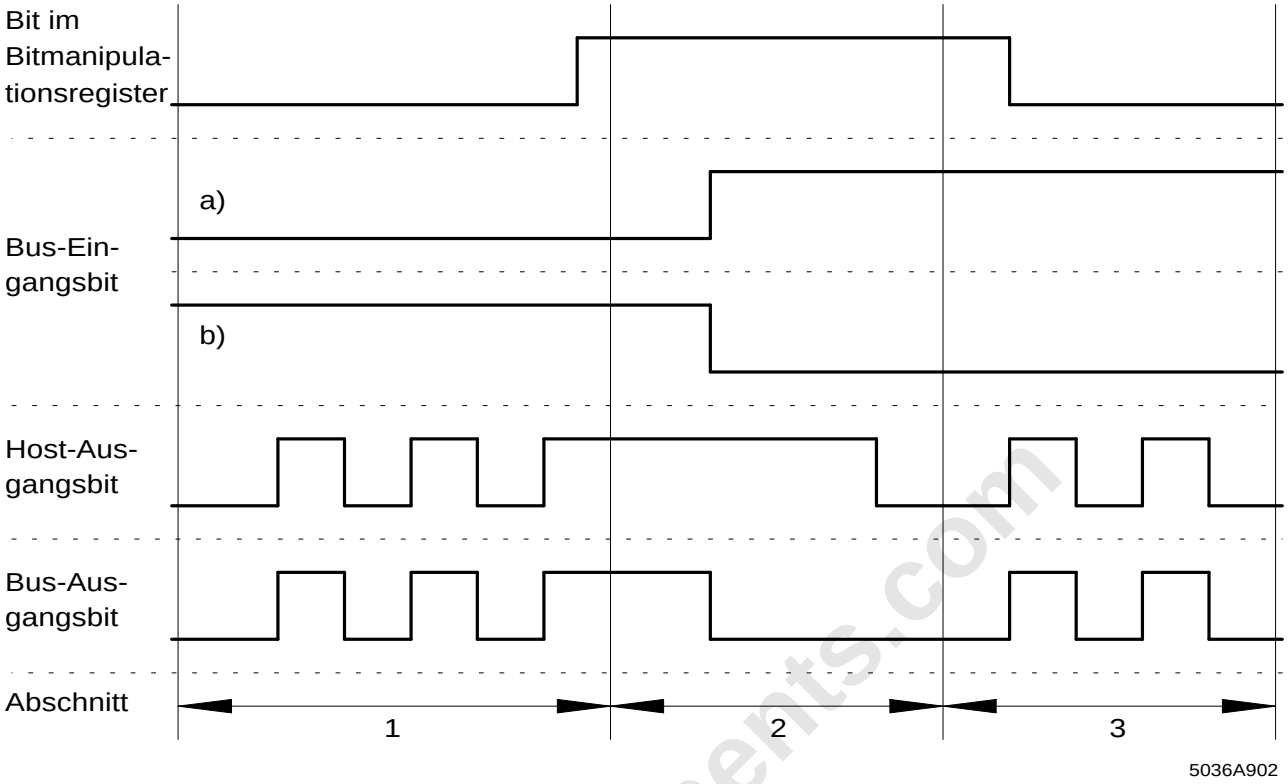

Bild 8-2: Impulsdiagramm zur Prozeßdatenverknüpfung

Allgemeine Funktion der Befehle RESET...

Das Zurücksetzen des Bus-Ausgangsbits erfolgt nach der Auswertung der Flanke des Bus-Eingangsbits (im Bild Abschnitt 2). Der Befehl bestimmt die Auswertung des Bus-Eingangsbits. Gesetzt wird das Bus-Ausgangsbit über das Setzen des Host-Ausgangsbits (im Bild Abschnitte 1 und 3). Solange das Bus-Ausgangsbit nicht durch die Auswertung des Bus-Eingangsbits zurückgesetzt ist, wird das Bus-Ausgangsbit nur vom Host-Ausgangsbit bestimmt (im Bild Abschnitte 1 und 3).

Tabelle 8-2: Befehle für die Prozeßdatenverknüpfung

Code	Befehl	Bemerkung	Seite
0000_{hex}	Clear_All_Processing_ Instructions	löscht alle bereits gespeicherten Befehle	8-32
0005_{hex}	Reset_Out_In01	reagiert auf positive Flanke (0→1)	8-32
0006_{hex}	Reset_Out_In10	reagiert auf negative Flanke (1→0)	8-32
0007_{hex}	Reset2_Out_In01	reagiert auf positive Flanke (0→1)	8-33
0008_{hex}	Reset2_Out_In10	reagiert auf negative Flanke (1→0)	8-34
0009_{hex}	Reset2_Out_In01_Switch	reagiert auf positive Flanke (0→1)	8-35
$000A_{hex}$	Reset2_Out_In10_Switch	reagiert auf negative Flanke (1→0)	8-36
$000D_{hex}$	Bit_Copy	kopiert ein Ein- auf ein Ausgangsbit	8-37
$000E_{hex}$	Word_Copy_Bit_Mask	Verknüpfung eines Eingangsworts mit einer Maske	8-37

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Die Prozeßdatenverknüpfung ist nur im Adreßbereich der binären Ein- und Ausgänge durchführbar. Die Zykluszeit des Datenverkehrs auf dem Bus kann sich um bis zu 40 µs pro Befehl erhöhen. Die Befehle werden mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe übertragen.

Adreßangaben inerhalb der Befehle zur Prozeßdatenverknüpfung

Geben Sie für jede Prozeßdatenverknüpfung das Eingangs- und das Ausgangsbit an. Adressieren Sie diese Bits über die Angabe der Adresse und der Bitnummer. Geben Sie für

- IBS-Teilnehmer mit 16 Bit (oder mehr) Adreßraum die gerade Adresse und das gewünschte Bit (0 bis 15) an;
- IBS-Teilnehmern mit 8 Bit Adreßraum (Längen-Code 81_{hex}) die gerade oder ungerade Adresse und das gewünschte Bit (0 bis 7) an

Die InterBus-typische Numerierung von Worten und Bytes ist im Kapitel 6.3.4 erläutert.

Ein Wort eines IBS-Teilnehmer mit 16 Bit (oder mehr) Adreßraum::

	Wort n
Bit	15 0

Ein Wort mit zwei IBS-Teilnehmern mit je 8 Bit Adreßraum::

	Byte n	Byte n+1
Bit	7 0	7 0

Verwenden Sie dabei, je nach Adressierungsart, die physikalische oder die logische Adresse, die Sie mit dem Kommandos
- *Receive_Logical_IN_Address_Map_Request* ($003A_{hex}$)
- *Receive_Logical_OUT_Address_Map_Request* ($003B_{hex}$)

vergeben haben.

Receive_Processing_Instructions_Request **$005D_{hex}$**

Aufgabe:

Das Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) überträgt einen oder mehrere Befehle zur Verknüpfung von Prozeßdaten zur Anschaltbaugruppe. Die Befehle werden in Form einer Befehlsliste zur Anschaltbaugruppe gesendet und dort im RAM gespeichert.

Übertragen Sie die Befehlsliste vor dem Start des IBS-Systems und -wenn verwendet- nach dem Kommando *Implement_All_Logical_Address_Maps_Request*. Die Prozeßdatenverknüpfung darf während des Betriebs **nicht** neu definiert und nicht in Verbindung mit Gruppendefinitionen verwendet werden, da hierbei Bits der von der Prozeßdatenverknüpfung verwendeten Prozeßausgangsdaten zurückgesetzt werden

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

können.
Für die Befehlsliste steht ein Speicher zur Verfügung, der bis zu 1000 Einträge aufnehmen kann. Dies entspricht z.B.:

- 332 Befehlen von 3 Wort Länge (Befehls-Code 0005_{hex} oder 0006_{hex})
- 199 Befehlen von 5 Wort Länge (Befehls-Code 0007_{hex} oder 0008_{hex})
- 142 Befehlen von 7 Wort Länge (Befehls-Code 0009_{hex} oder $000A_{hex}$)

Befehle mit ungültigen Adressen kommen nicht zur Ausführung. Es wird **keine** gesonderte Meldung für diesen Fall gegeben. Adressen (und damit Befehle) werden ungültig bei

- der logischen Adressierung, wenn zuvor verwendete Adressen nicht mehr vorhanden sind, oder
- bei abgeschalteten Gruppen, wenn die Adresse in dem Bereich einer abgeschalteten Gruppe liegt.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	Befehls-Code	1. Befehl
Wort 4	1. Befehlsparameter	
Wort 5	2. Befehlsparameter	
usw.	...	
	letzter Befehlsparameter	
	Befehls-Code	2. Befehl
	1. Befehlsparameter	
	2. Befehlsparameter	
	...	
	letzter Befehlsparameter	
	...	
	...	
	Befehls-Code	n. Befehl
	1. Befehlsparameter	
	2. Befehlsparameter	
	...	
Wort n+2	letzter Befehlsparameter	

Bit	15 0

Legende:
Code: Kommando-Code (hier $005D_{hex}$)
Parameterzähler: Anzahl der nachfolgenden Worte (hier Summe der einzelnen Befehlslängen).

positive Quittung: Quit_Receive_Instructions_Confirmation (00F2)
Bedeutung: Die Parameter sind fehlerfrei. Alle aufgeführten Befehle sind angenommen und an eine schon eventuell bestehende Liste angehängt.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

negative Quittung: Processing_Instructions_Error_Confirmation (80F3)
Bedeutung: Die Parameter sind fehlerhaft. Die gesendete Liste wird nicht übernommen (siehe Kapitel 10).

Befehl *Clear_All_Processing_Instructions* (0000_{hex})

Aufgabe: Der Befehl löscht alle im RAM der Anschaltbaugruppe vorhandene Befehle. Setzen Sie den Befehl an den Anfang einer zu übertragenden Befehlsliste, wenn diese die zuvor im RAM gespeicherte Befehle überschreiben soll. Wenn Sie den Befehl *Clear_Instruction_List* unterlassen werden die bereits gespeicherten Befehle um die neu übertragenen Befehle ergänzt.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax: Der Befehl besteht nur aus einem Wort, dem Befehls-Code (0006_{hex}). Es folgen keine weiteren Parameter.

Befehl *Reset_Out_In01* (0005_{hex})

Aufgabe: Dieser Befehl bewirkt das Rücksetzen eines bestimmten Bits im angegebenen Ausgangswort bei Auftreten der **positiven** Flanke eines bestimmten Bits im angegebenen Eingangswort.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax:

Wort 1	Code		
Wort 2	Bitnummer	res.	Ausgabeadresse
Wort 3	Bitnummer	res.	Eingangsadresse

Bit	15 12	11	10 0

Legende:
Code: Befehls-Code (hier 0005_{hex})
Bitnummer: Geben Sie hier jeweils die Nummer des gewünschten Bits an (0_{hex} bis F_{hex} entspricht 0_{dez} bis 15_{dez}).
res.: Dieses Bit ist für spätere Erweiterungen reserviert. Setzen Sie es auf 0.
Ausgabeadresse: Adresse des Worts mit dem durch die Bitmanipulation manipulierten Bit
Eingangsadresse: Adresse des Worts mit dem die Bitmanipulation bestimmenden Bit

Befehl *Reset_Out_In10* (0006_{hex})

Aufgabe: Dieser Befehl bewirkt das Rücksetzen eines bestimmten Bits im angegebenen Ausgangswort bei Auftreten der **negativen** Flanke eines bestimmten Bits im angegebenen Eingangswort.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

online-components.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Syntax:

Wort 1	Code		
Wort 2	Bitnummer	res.	Ausgabeadresse
Wort 3	Bitnummer	res.	Eingangsadresse

Bit	15 12	11	10 0

Legende:

Code:	Befehls-Code (hier 0006_{hex})
Bitnummer:	Geben Sie hier jeweils die Nummer des gewünschten Bits an (0_{hex} bis F_{hex} entspricht 0_{dez} bis 15_{dez}).
res.:	Dieses Bit ist für spätere Erweiterungen reserviert. Setzen Sie es auf 0.
Ausgabeadresse:	Adresse des Worts mit dem durch die Bitmanipulation manipulierten Bit
Eingangsadresse:	Adresse des Worts mit dem die Bitmanipulation bestimmenden Bit

Befehl *Reset2_Out_In01* (0007_{hex})

Aufgabe:

Dieser Befehl bewirkt das Rücksetzen eines bestimmten Bits im angegebenen Ausgangswort bei Auftreten der **positiven** Flanke eines bestimmten Bits im angegebenen Eingangswort.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax:

Wort 1	Code	
Wort 2	Ausgabeadresse	
Wort 3	000_{hex}	Bitnummer
Wort 4	Eingangsadresse	
Wort 5	000_{hex}	Bitnummer

Bit	15 4	3 0

Legende:

Code:	Befehls-Code (hier 0007_{hex})
Bitnummer:	Geben Sie hier jeweils die Nummer des gewünschten Bits an (0_{hex} bis F_{hex} entspricht 0_{dez} bis 15_{dez}).
Ausgabeadresse:	Adresse des Worts mit dem durch die Bitmanipulation manipulierten Bit
Eingangsadresse:	Adresse des Worts mit dem die Bitmanipulation bestimmenden Bit

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Befehl *Reset2_Out_In10* (0008_{hex})

Aufgabe:

Dieser Befehl bewirkt das Rücksetzen eines bestimmten Bits im angegebenen Ausgangswort bei Auftreten der **negativen** Flanke eines bestimmten Bits im angegebenen Eingangswort.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax:

Wort 1	Code	
Wort 2	Ausgabeadresse	
Wort 3	000_{hex}	Bitnummer
Wort 4	Eingangsadresse	
Wort 5	000_{hex}	Bitnummer

Bit	15 4	3 0

Legende:

Code:	Befehls-Code (hier 0008_{hex})
Bitnummer:	Geben Sie hier jeweils die Nummer des gewünschten Bits an (0_{hex} bis F_{hex} entspricht 0_{dez} bis 15_{dez}).
Ausgabeadresse:	Adresse des Worts, mit dem durch die Bitmanipulation manipulierten Bits
Eingangsadresse:	Adresse des Worts, mit dem die Bitmanipulation bestimmenden Bits

☞ Die Befehle *Reset2_Out_In01* und *Reset2_Out_In10* sind funktionsgleich mit den Befehlen *Reset_Out_In01* und *Reset_Out_In10*. Die Befehle sind allerdings 5 Worte lang, so daß 16 Bit für die Adressen zur Verfügung stehen.

Hinweise zu den Befehlen für die bedingte Prozeßdatenverknüpfung

☞ Wenn das angegebene Bit im Bitmanipulationsregister gesetzt ist, verhalten sich die Befehle wie *Reset...01* und *Reset...10*. Ist das angegebene Bit im Bitmanipulationsregister nicht gesetzt, ist der Befehl nicht wirksam und es erfolgt keine Prozeßdatenverknüpfung.

Das Bitmanipulationsregister muß vor Anwendung der Befehle *Reset_Out_In01_Switch* und *Reset_Out_In10_Switch,* definiert werden. Tragen Sie dazu im Kommando *Check_Physical_Configuration* (0058_{hex}) ein "Pseudomodul" (ID-Code 0013_{hex}) als letztes Modul der Konfiguration ein. Weisen Sie diesem "Pseudomodul" anschließend mit der logischen Adressierung die Adresse des Bitmanipulationsregisters zu.

☞ Wenn Sie anstelle des Bitmanipulationsregisters das Prozeßeingangswort eines real vorhandenen, digitalen Eingabemoduls verwenden, können Sie aus der Anlage heraus (z.B. mit einem Schlüsselschalter) bestimmen, ob die Prozeßdatenverknüpfung wirksam sein soll oder nicht.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Befehl *Reset2_Out_In01_Switch* (0009_{hex})

Aufgabe:

Dieser Befehl bewirkt das Rücksetzen eines bestimmten Bits im angegebenen Ausgangswort bei Auftreten der **positiven** Flanke eines bestimmten Bits im angegebenen Eingangswort, wenn ein bestimmtes Bit im Bitmanipulationsregister gesetzt ist.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax:

Wort 1	Code	
Wort 2	BitmanRegAdr	
Wort 3	000_{hex}	Bitnummer
Wort 4	Ausgabeadresse	
Wort 5	000_{hex}	Bitnummer
Wort 6	Eingangsadresse	
Wort 7	000_{hex}	Bitnummer

Bit	15 4	3 0

Legende:

Code:	Befehls-Code (hier 0009_{hex})
Bitnummer:	Geben Sie hier jeweils die Nummer des gewünschten Bits an (0_{hex} bis F_{hex} entspricht 0_{dez} bis 15_{dez}).
BitmanRegAdr:	Adresse des Bitmanipulationsregisters
Ausgabeadresse:	Adresse des Worts mit dem durch die Bitmanipulation manipulierten Bit
Eingangsadresse:	Adresse des Worts mit dem die Bitmanipulation bestimmenden Bit

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Befehl *Reset2_Out_In10_Switch* ($000A_{hex}$)

Aufgabe:

Dieser Befehl bewirkt das Rücksetzen eines bestimmten Bits im angegebenen Ausgangswort bei Auftreten der **negativen** Flanke eines bestimmten Bits im angegebenen Eingangswort, wenn ein bestimmtes Bit im Bitmanipulationsregister gesetzt ist.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax:

Wort 1	Code	
Wort 2	BitmanRegAdr	
Wort 3	000_{hex}	Bitnummer
Wort 4	Ausgabeadresse	
Wort 5	000_{hex}	Bitnummer
Wort 6	Eingangsadresse	
Wort 7	000_{hex}	Bitnummer

Bit	15 4	3 0

Legende:

Code:	Befehls-Code (hier $000A_{hex}$)
Bitnummer:	Geben Sie hier jeweils die Nummer des gewünschten Bits an (0_{hex} bis F_{hex} entspricht 0_{dez} bis 15_{dez}).
BitmanRegAdr:	Adresse des Bitmanipulationsregisters
Ausgabeadresse:	Adresse des Worts mit dem durch die Prozeßdatenverknüpfung manipulierten Bit
Eingangsadresse:	Adresse des Worts mit dem die Prozeßdatenverknüpfung bestimmenden Bit

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Befehl *Bit_Copy* ($000D_{hex}$)

Aufgabe: Dieser Befehl bewirkt das Kopieren eines bestimmten Bits im angegebenen Eingangswort auf ein bestimmtes Bit im angegebenen Ausgangswort.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax:

Wort 1	Code	
Wort 2	Ausgabeadresse	
Wort 3	000_{hex}	Bitnummer
Wort 4	Eingangsadresse	
Wort 5	000_{hex}	Bitnummer

Bit	15 4	3 0

Legende:

Code:	Befehls-Code (hier $000D_{hex}$)
Bitnummer:	Geben Sie hier jeweils die Nummer des gewünschten Bits an (0_{hex} bis F_{hex} entspricht 0_{dez} bis 15_{dez}).
Ausgabeadresse:	Adresse des Worts, in das das Bit kopiert wird
Eingangsadresse:	Adresse des Worts, aus dem das Bit kopiert wird

Befehl *Word_Copy_Bit_Mask* ($000E_{hex}$)

Aufgabe: Dieser Befehl bewirkt die bitweise UND-Verknüpfung eines bestimmten Eingangsworts mit einem Maskierungswort und schreibt das Ergebnis auf ein bestimmtes Ausgangswort.

Übertragen Sie den Befehl mit dem Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) zur Anschaltbaugruppe.

Syntax:

Wort 1	Code
Wort 2	Ausgabeadresse
Wort 4	Eingangsadresse
	Maske

Bit	15 0

Legende:

Code:	Befehls-Code (hier $000E_{hex}$)
Ausgabeadresse:	Adresse des Ausgangsworts, in das das Verknüpfungsergebnis kopiert wird
Eingangsadresse:	Adresse des zu verknüpfenden Eingangsworts
Maske:	Maskierungswort

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

8.9 Kommandos zur Event-Bearbeitung

In einigen Anwendungen ist es sinnvoll, auf bestimmte Ereignisse (englisch Events) gezielt zu reagieren. Die Event-Definition ermöglicht es, hierzu vordefinierte Signalwechsel bestimmter Eingänge explizit gemeldet zu bekommen.

Vorbereitende Schritte zur Event-Bearbeitung:
1. Definieren Sie die gewünschten Events mit dem Kommando *Receive_Events_Request* ($002A_{hex}$).
2. Legen Sie mit dem Kommando *Set_Event_Message_Type_Request* ($004D_{hex}$) den Meldungsttyp fest. Standardeinstellung ist *Event_Number_Indication* (8099_{hex})
3. Geben Sie die gewünschten Events frei (z.B. mit dem Kommando *Enable_Event_Number_Request* ($002B_{hex}$)).

Das Auftreten aller definierten und freigegebenen Events wird in zugeordneten Countern gezählt. Diese Event-Counter werden mit der Freigabe des Events zurückgesetzt. Sie können mit dem Kommando *Read_Event_Counter_Request* (0114_{hex}) ausgelesen werden.

Verwenden Sie im Zusammenhang mit der Event-Bearbeitung möglichst die logische Adressierung. Zwar läßt sich die Event-Bearbeitung auch unter physikalischer Adressierung betreiben, bei späteren Erweiterungen ihrer Anlage müssen Sie dann aber in sämtlichen Zuordnungslisten alle Adreßangaben anpassen. Bei logischer Adressierung müssen Sie die Zuordnungslisten nur um die neu hinzugekommenen IBS-Teilnehmer erweitern.

Receive_Events_Request **$002A_{hex}$**

Aufgabe:

Die Parameter des Kommandos erlauben die Definition von bis zu 16 Events. Jede Event-Definition belegt 5 Worte.

Syntax:

Wort 1	Code				
Wort 2	Parameterzähler				
Wort 3	1. Event-Kennung				
Wort 4	Event-Code				
Wort 5	0_{hex}	00_{bin}	Adresse		1. Eventdefinition
Wort 6	Maske 1				
Wort 7	Maske 2				
	...				
	...				
	x. Event-Kennung				
	Event-Code				
	0_{hex}	00_{bin}	Adresse		x. Eventdefinition
	Maske 1				
Wort (5z+2)	Maske 2				
Bit	15 12	11 10	9 0		

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Legende:	Code:	Kommando-Code (hier $002A_{hex}$)
	Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl (z) der Eventdefinitionen multipliziert mit 5).
	x:	Anzahl der Eventdefinitionen (1 bis 16)
	Event-Kennung:	Sie bezeichnet die einzelnen Events. Vergeben Sie für die 16 möglichen Event-Definitionen nur Event-Kennungen, die aus einem Wort bestehen, in dem ein Bit gesetzt und 15 Bits nicht gesetzt sind. Hierdurch kann Ihnen die Meldung *Event_Number_Indication* (8099_{hex}) durch bitweise ODER-Verknüpfung der Event-Kennungen, in einem Wort alle aufgetretenen Events anzeigen (siehe Meldungsbeschreibung).
	Event-Code:	Vergeben Sie den Event-Code 0004_{hex} für die Erkennung von Bit-Zustandswechseln bei digitalen Modulen.
	Adresse:	Tragen Sie hier, je nach verwendeter Adressierungsart, (siehe Kapitel 6) die logische oder physikalische Adresse des Eingangsworts ein, für das Sie einen Event definieren möchten. Benutzen Sie dazu gerade Byteadressen, wie sie z.B. bei der Vergabe der logischen Adressen mit dem Kommandos *Receive_Logical_IN_Address_Map_Req.* ($003A_{hex}$) *Receive_Logical_OUT_Address_Map_Req.* ($003B_{hex}$) verwendet werden (siehe Seite 6-19).
	Maske 1:	Spezifiziert innerhalb des durch die Adresse bestimmten Eingangsworts die Bits, für die bei einer positiven Flanke die Meldung eines Events ausgelöst wird. Setzen Sie die gewünschten Bits auf 1.
	Maske 2:	Spezifiziert innerhalb des durch die Adresse bestimmten Eingangsworts die Bits, für die bei einer negativen Flanke die Meldung eines Events ausgelöst wird. Setzen Sie die gewünschten Bits auf 1.

positive Quittung: Quit_Receive_Event_Confirmation ($00BF_{hex}$)

Bedeutung: Die Event-Definitionen wurden übernommen, sind aber noch nicht wirksam. Zuvor mit diesem Kommando definierte Event-Definitionen wurden überschrieben.

negative Quittung: Event_Error_Confirmation ($004C_{hex}$)

Bedeutung: Fehler bei der Eventdefinition. Es liegen keine gültigen Eventdefinitionen vor, alte Definitionen sind gelöscht.

Enable_All_Events_Request $002D_{hex}$

Aufgabe: Das Kommando schaltet alle zuvor definierten Event-Definitionen aktiv.

Voraussetzung: Die entsprechenden Event-Definitionen müssen vorhanden und gültig (Kommando *Receive_Events_Request* ($002A_{hex}$)) sein.

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($002D_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

positive Quittung: Quit_Enable_All_Events_Confirmation ($00DC_{hex}$)
Bedeutung: Alle Event-Definitionen wurden aktiv geschaltet.

negative Quittung: Event_Error_Confirmation ($004C_{hex}$)
Bedeutung: Fehler bei der Eventdefinition. Es liegen ungültige Eventdefinitionen vor.

Disable_All_Events_Request **$002E_{hex}$**

Aufgabe: Das Kommando schaltet alle zuvor definierten Event-Definitionen inaktiv.

Voraussetzung: Die entsprechenden Event-Definitionen müssen vorhanden und gültig (Kommando *Receive_Events_Request* ($002A_{hex}$)) sein.

positive Quittung: Quit_Disable_All_Events_Confirmation ($00DC_{hex}$)
Bedeutung: Alle Event-Definitionen wurden inaktiv geschaltet.

negative Quittung: Eine negative Quittung erfolgt nicht!

Syntax: Das Kommando besteht nur aus einem Wort, dem Kommando-Code ($002E_{hex}$). Es folgen keine weiteren Parameter.

Enable_Event_Number_Request **$002B_{hex}$**

Aufgabe: Das Kommando schaltet eine bestimmte Event-Definition aktiv.

Voraussetzung: Die entsprechende Event-Definition muß vorhanden und gültig (Kommando Receive_Events_Request ($002A_{hex}$)) sein.

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Event-Kennung

Bit 15 0

Legende:
Code: Kommando-Code (hier $002B_{hex}$)
Parameterzähler: Anzahl der nachfolgenden Worte (hier 1).
Event-Kennung: Tragen Sie hier die Kennung der Event-Definition ein, die Sie aktiv schalten möchten.

positive Quittung: Quit_Enable_Event_Number_Confirmation ($00DA_{hex}$)
Bedeutung: Die angegebene Event-Definition wurde aktiv geschaltet.

negative Quittung: Event_Error_Confirmation ($004C_{hex}$)
Bedeutung: Die angegebene Event-Definition wird nicht wirksam.
- Die angegebene Event-Definition existiert nicht.
- Die angegebene Event-Definition liegt im Bereich einer abgeschalteten Gruppe.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Disable_Event_Number_Request **$002C_{hex}$**

Aufgabe: Das Kommando schaltet eine bestimmte Event-Definition inaktiv.

Voraussetzung: Die entsprechende Event-Definition muß vorhanden und gültig (Kommando *Receive_Events_Request* ($002A_{hex}$)) sein.

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Event-Kennung

Bit	15 0

Legende:

Code:	Kommando-Code (hier $002C_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1).
Event-Kennung:	Tragen Sie hier die Kennung der Event-Definition ein, die Sie inaktiv schalten möchten.

positive Quittung: Quit_Disable_Event_Number_Confirmation ($00DA_{hex}$)
Bedeutung: Die angegebene Event-Definition wurde inaktiv geschaltet.

negative Quittung: Event_Error_Confirmation ($004C_{hex}$)
Bedeutung: Die angegebene Event-Definition verbleibt im vorherigen Zustand.
- Die angegebene Event-Definition liegt im Bereich einer abgeschalteten Gruppe.

Enable_Event_Logical_Address_Request **0036_{hex}**

Aufgabe: Das Kommando schaltet die Event-Definition für das über die Adresse angegebene Eingangswort aktiv.

Voraussetzung: Die entsprechenden Event-Definition muß vorhanden und gültig (Kommando *Receive_Events_Request* ($002A_{hex}$)) sein.

Syntax:

Wort 1	Code		
Wort 2	Parameterzähler		
Wort 3	0_{hex}	00_{bin}	Eingangsadresse

Bit	15 12	11 10	9 0

Legende:

Code:	Kommando-Code (hier 0036_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1).
Eingangsadresse:	Tragen Sie hier, je nach verwendeter Adressierungsart, die logische oder physikalische Adresse (10Bit) des Eingangsworts ein, für das Sie die Event-Definition aktiv schalten möchten. Benutzen Sie dazu gerade Byte-

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

adressen, wie sie z.B. bei der Vergabe der logischen Adressen mit dem Kommandos *Receive_Logical_IN_Address_Map_Req.* ($003A_{hex}$) *Receive_Logical_OUT_Address_Map_Req.* ($003B_{hex}$) verwendet werden (siehe Seite 6-19).

positive Quittung: Quit_Enable_Event_Logical_Address_Confirmation ($00E0_{hex}$)
Bedeutung: Die ausgewählte Event-Definition wurde aktiv geschaltet.

negative Quittung: Event_Error_Confirmation ($004C_{hex}$)
Bedeutung: Die ausgewählte Event-Definition wird nicht wirksam.
- Die angegebene Eingangsadresse existiert nicht.
- Für die angegebene Eingangsadresse existiert keine Event-Definition.
- Die angegebene Eingangsadresse liegt im Bereich einer abgeschalteten Gruppe.

Disable_Event_Logical_Address_Request **0037_{hex}**

Aufgabe: Das Kommando schaltet die Event-Definition für das über die Adresse angegebene Eingangswort inaktiv.

Voraussetzung: Die entsprechenden Event-Definition muß vorhanden und gültig (Kommando *Receive_Events_Request* ($002A_{hex}$)) sein.

Syntax:

Wort 1	Code		
Wort 2	Parameterzähler		
Wort 3	0_{hex}	00_{bin}	Eingangsadresse

Bit	15 12	11 10	9 0

Legende:
Code: Kommando-Code (hier 0037_{hex})
Parameterzähler: Anzahl der nachfolgenden Worte (hier 1).
Eingangsadresse: Tragen Sie hier, je nach verwendeter Adressierungsart, die logische oder physikalische Adresse (10Bit) des Eingangsworts ein, für das Sie die Event-Definition inaktiv schalten möchten. Benutzen Sie dazu gerade Byteadressen, wie sie z.B. bei der Vergabe der logischen Adressen mit dem Kommandos *Receive_Logical_IN_Address_Map_Req.* ($003A_{hex}$) *Receive_Logical_OUT_Address_Map_Req.* ($003B_{hex}$) verwendet werden (siehe Seite 6-19).

positive Quittung: Quit_Enable_Event_Logical_Address_Confirmation ($00E1_{hex}$)
Bedeutung: Die ausgewählte Event-Definition wurde inaktiv geschaltet.

negative Quittung: Event_Error_Confirmation ($004C_{hex}$)
Bedeutung: Die angegebene Event-Definition verbleibt im vorherigen Zustand.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

- Die angegebene Eingangsadresse existiert nicht.
- Die angegebene Eingangsadresse liegt im Bereich einer abgeschalteten Gruppe.
- Für die angegebene Eingangsadresse existiert keine Event-Definition.

Set_Event_Message_Type_Request 004D$_{hex}$

Aufgabe:

Dieses Kommando definiert den Meldungstyp, der nach dem Auftreten eines Events ausgelöst wird. Generierte Meldungstypen:

1 Event_Number_Indication (8099$_{hex}$)
In dieser Einstellung werden alle aufgetretenen Events in einem Wort übergeben. Dieses Wort wird durch die ODER-Verknüpfung der Event-Kennungen aller aufgetretenen Events generiert.

2 Event_Data_Indication (809B$_{hex}$)
In dieser Einstellung werden für jeden aufgetretenen Event die Event-Kennung und das Datenwort des Eingabe-Moduls, bei dem der Event ausgelöst worden ist, übergeben.

Standardeinstellung nach dem Hochlaufen der Anschaltbaugruppe ist der Meldungstyp *Event_Number_Indication* (8099$_{hex}$). Verwenden Sie den Meldungstyp *Event_Data_Indication* (809B$_{hex}$), wenn die zum Event-Zeitpunkt zugehörigen Daten für eine weitergehende Behandlung verwendet werden sollen. Dadurch ist die zeitliche Zugehörigkeit des Datums zum Event garantiert.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	000$_{hex}$	Meldungstyp
Bit	15 4	3 0

Legende:

Code:	Kommando-Code (hier 004D$_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1).
Meldungstyp:	Tragen Sie hier den gewünschten Meldungstyp ein: 1: Event_Number_Indication (8099$_{hex}$) 2: Event_Data_Indication (809B$_{hex}$)

positive Quittung: Quit_Set_Event_Message_Typ_Confirmation (00DE$_{hex}$)

negative Quittung: Set_Event_Message_Typ_Failed_Confirmation (00DF$_{hex}$)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Read_Event_Counter_Request 0114_{hex}

Aufgabe:

Zu jedem der maximal 16 Events ist ein Event-Zähler angelegt. Das Kommando *Read_Event_Counter_Request* (0114_{hex}) liest die Event-Counter aus. Ein Event-Zähler wird inkrementiert, wenn der Event mit der zugehörigen Kennung erkannt wird.

Vorausetzung:

Die Events müssen definiert und freigegeben sein. Ein Event-Counter wird mit der Freigabe (Kommandos $002C_{hex}$, $002E_{hex}$ oder 0036_{hex}) des zugehörigen Events auf 0 gesetzt.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	Event-Kennung	für 1. Event
Wort 4	Event-Kennung	für 2. Event
	...	
Wort n+2	Event-Kennung	für n. Event

Bit	15 0

Legende:

Code: Kommando-Code (hier 0114_{hex})

Parameterzähler: Anzahl der nachfolgenden Worte (hier Anzahl der auszulesenden Event-Counter, maximal 16_{dez}).

Event-Kennung: Tragen Sie hier die Kennungen der Event-Definitionen ein, deren Zähler Sie auslesen möchten. Die Ausgabereihenfolge in der Quittung entspricht der Reihenfolge, die Sie hier vorgeben.
Falls Sie als einzigen Parameter eine 0 eintragen, werden die Zählerstände aller Events ausgegeben. Die Ausgabereihenfolge der Zählerstände in der Quittung entspricht dann der bei der Event-Definition mit dem Kommando *Receive_Events_Request* ($0002A_{hex}$) vorgegebenen Reihenfolge.

Quittung:

Quit_Read_Event_Counter_Confirmation ($811D_{hex}$)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel **9**

Meldungen des IBS-Masterboards

Dieses Kapitel informiert Sie über

- die Bedeutung und Ursachen der Meldungen des IBS-Masterboards,
- die Parameter der Meldungen.

9 Meldungen des IBS-Masterboards 9-3

9.1 Aufbau einer Meldungsbeschreibung 9-5
9.2 Meldungen zur Konfiguration 9-6
9.3 Meldungen zur Adressierung 9-11
9.4 Meldungen für den Betrieb 9-14
9.5 Meldungen zur Fehlerbehandlung 9-15
9.6 Meldungen der Anwenderschnittstelle 9-31
9.7 Meldungen zur Systemkontrolle 9-32
9.8 Meldungen zur Prozeßdatenverknüpfung 9-33
9.9 Meldungen zur Event-Bearbeitung 9-34

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9 Meldungen des IBS-Masterboards

Tabelle 9-1: Meldungen des IBS-Masterboards

Code	Meldung	Seite
0007_{hex}	No_Command_Code_Indication	9-15
0008_{hex}	Command_Write_Error_Indication	9-15
0009_{hex}	Parameter_Write_Error_Indication	9-15
$000A_{hex}$	No_Command_Parameter_Routine_Indication	9-15
$000B_{hex}$	No_Error_Code_Indication	9-16
$000C_{hex}$	Controller_Parameter_Timeout_Indication	9-16
$000D_{hex}$	Host_Parameter_Timeout_Indication	9-16
0023_{hex}	CPU_Bus_Error_Indication	9-17
$002B_{hex}$	Logical_Address_Error_Confirmation	9-11
0038_{hex}	Bus_System_Error_Indication	9-17
$004A_{hex}$	No_Executable_Configuration_Confirmation	9-17
$004B_{hex}$	Command_Parameter_Error_Confirmation	9-17
$004C_{hex}$	Event_Error_Confirmation	9-34
$004D_{hex}$	IPMS_No_Error_Indication	9-18
$004E_{hex}$	Communication_Invalid_Indication	9-18
0059_{hex}	Unexpected_Group_Number_Confirmation	9-6
$005A_{hex}$	BK_Alarm_Failed_Confirmation	9-18
$005B_{hex}$	Unknown_Bus_Module_Confirmation	9-9
0066_{hex}	Receive_CR_Error_Confirmation	9-19
0068_{hex}	Check_Configuration_Error_Confirmation	9-8
0069_{hex}	Communication_Not_Ready_Confirmation	9-19
0088_{hex}	Start_Bus_Confirmation	9-14
$009C_{hex}$	Command_Disabled_Confirmation	9-31
$00AB_{hex}$	Physical_Configuration_Map_Valid_Confirmation	9-9
$00BD_{hex}$	Quit_Receive_Group_Numbers_Confirmation	9-6
$00BF_{hex}$	Quit_Receive_Events_Confirmation	9-34
$00C6_{hex}$	Stop_Bus_Confirmation	9-14
$00CA_{hex}$	Quit_Configure_Bus_Confirmation	9-10
$00CF_{hex}$	Quit_Set_Parameter_Timeout_Confirmation	9-31
$00D0_{hex}$	Quit_Receive_Logical_Localbus_Addressmap_Confirmation	9-11
$00D1_{hex}$	Quit_Receive_Logical_In_Addressmap_Confirmation	9-11
$00D2_{hex}$	Quit_Receive_Logical_Out_Addressmap_Confirmation	9-11
$00D3_{hex}$	Quit_Implement_Confirmation	9-11
$00D6_{hex}$	Quit_Enable_All_Messages_Confirmation	9-31
$00D7_{hex}$	Quit_Disable_All_Messages_Confirmation	9-31

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 9-1: Meldungen des IBS-Masterboards

Code	Meldung	Seite
$00D8_{hex}$	Quit_Alarm_Stop_Confirmation	9-14
$00DA_{hex}$	Quit_Enable_Event_Number_Confirmation	9-34
$00DB_{hex}$	Quit_Disable_Event_Number_Confirmation	9-34
$00DC_{hex}$	Quit_Enable_All_Events_Confirmation	9-34
$00DD_{hex}$	Quit_Disable_All_Events_Confirmation	9-34
$00DE_{hex}$	Quit_Set_Event_Message_Type_Confirmation	9-35
$00DF_{hex}$	Set_Event_Message_Type_Failed_Confirmation	9-35
$00E0_{hex}$	Quit_Enable_Event_Logical_Address_Confirmation	9-35
$00E1_{hex}$	Quit_Disable_Event_Logical_Address_Confirmation	9-35
$00E2_{hex}$	Quit_Clear_Display_Confirmation	9-15
$00EC_{hex}$	Quit_Bus_Delay_Confirmation	9-15
$00ED_{hex}$	No_Map_Entry_Confirmation	9-19
$00F2_{hex}$	Quit_Receive_Instructions_Confirmation	9-33
$00F6_{hex}$	Quit_Groups_Error_Characteristic_Confirmation	9-8
$00FE_{hex}$	Quit_Module_Error_Ok_Confirmation	9-19
0105_{hex}	Quit_Receive_Localbus_Code_Map_Confirmation	9-13
8087_{hex}	Software_Revision_Confirmation	9-32
$809D_{hex}$	Switch_Group_Off_Confirmation	9-6
$809E_{hex}$	Switch_Group_On_Confirmation	9-7
$80A0_{hex}$	Module_Error_Indication	9-20
$80A1_{hex}$	Power_Fail_Indication	9-20
$80A2_{hex}$	Battery_Indication	9-21
$80C4_{hex}$	Bus_Error_Information_Indication	9-22
EE01	Erklärung der Fehlerart und Hinweise zur Abhilfe	9-23
EE02	Erklärung der Fehlerart und Hinweise zur Abhilfe	9-24
EE03	Erklärung der Fehlerart und Hinweise zur Abhilfe	9-26
EE04	Erklärung der Fehlerart und Hinweise zur Abhilfe	9-28
EE05	Erklärung der Fehlerart und Hinweise zur Abhilfe	9-28
EE06	Erklärung der Fehlerart und Hinweise zur Abhilfe	9-29
$80C5_{hex}$	Switch_Group_On_Failed_Confirmation	9-7
$80EE_{hex}$	Localbus_Module_Error_Confirmation	9-30
$80EF_{hex}$	Send_All_Module_Error_Confirmation	9-29
$80F3_{hex}$	Processing_Instructions_Error_Confirmation	9-33
$80F4_{hex}$	Send_Physical_Configuration_Confirmation	9-9
$80F5_{hex}$	Send_Logical_Address_Error_Confirmation	9-12
$80F7_{hex}$	Groups_Error_Characteristic_Failed_Confirmation	9-8
$80FF_{hex}$	Quit_Module_Error_Not_Possible_Confirmation	9-21
8119_{hex}	Send_Actual_Configuration_Confirmation	9-10
$811D_{hex}$	Quit_Read_Event_Counter_Confirmation	9-37

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.1 Aufbau einer Meldungsbeschreibung

Dieses Kapitel beschreibt die Meldungen des IBS-Masterboards. Die Meldungen werden wortweise, die Meldungs-Codes hexadezimal dargestellt. Die Beschreibungen sind nach dem folgenden Schema aufgebaut.

Name der Meldung **Meldungs-Code$_{hex}$**

Bedeutung: *Beschreibt die Aussage der Meldung.*

Ursache: *Beschreibt die Gründe, die zu der Meldung geführt haben.*

Abhilfe: *Gibt z.B. bei Fehlermeldungen Hinweise zur Behebung der Fehlerursache.*

Syntax: Für eine Meldung ohne Parameter wird nur der Meldungs-Code angegeben.

Die Syntax einer Meldung mit Parametern wird folgendermaßen als Parameterblock dargestellt:

Wort 1	*Meldungs-Code*
Wort 2	*Anzahl nachfolgender Worte (Parameterzähler)*
Wort 3	*Parameter 1*
Wort 4	*Parameter 2*
Wort 5	*Parameter 3*
	...
Wort n+2	*Parameter n*

Bit	15 0

Legende: Parameter *Beschreibung der einzelnen Parameter*

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.2 Meldungen zur Konfiguration

Receive_Group_Number_Failed_Confirmation **0055_{hex}**

Ursache: Die mit dem Kommando *Receive_Group_Numbers_Req*uest (0049_{hex}) übertragene Liste enthält fehlerhafte Einträge. Fehlerursachen können sein:
- Ungültige Bussegment-Nummern.
- Nummern für Bussegmente mehrfach vergeben.
- Ungültige Gruppennummer.
- Es wurde versucht, IBS-Teilnehmer ohne Weichenfunktion (kein Peripheriebusanschluß, kein Fernbus-Stich) mit anderen IBS-Teilnehmern zu einer Gruppe zusammenzufassen. Weisen Sie IBS-Teilnehmern ohne Weichenfunktion jeweils eine eigene Gruppennummer zu.
- Falsche Parameteranzahl

Abhilfe: Überprüfen Sie die Definitionslisten der Gruppendefinition.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0055_{hex}). Es folgen keine weiteren Parameter.

Unexpected_Group_Number_Confirmation **0059_{hex}**

Bedeutung: Fehler bei der Durchführung der Kommandos *Switch_Group_Off_Request* (0021_{hex}) oder *Switch_Group_On_Request* (0020_{hex})
- Es wurde versucht, einen nicht abschaltbaren Teilnehmer abzuschalten.
- Beim Ein- oder Abschalten einer Gruppe wurde eine Gruppennummer angegeben, die vorher nicht definiert worden war.
- Die übergebene Gruppennummer liegt außerhalb des zulässigen Bereichs.

Der Fehler wird nur gemeldet und beeinflußt nicht das weitere Verhalten der Anschaltbaugruppe.

Abhilfe: Überprüfen Sie das Anwendungsprogramm auf nicht existierende Gruppennummern und falsche Parameteranzahlen.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0059_{hex}). Es folgen keine weiteren Parameter.

Quit_Receive_Group_Numbers_Confirmation **$00BD_{hex}$**

Bedeutung: Das Kommando *Receive_Group_Numbers_Request* (0049_{hex}) wurde erfolgreich ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00BD_{hex}$). Es folgen keine weiteren Parameter.

Switch_Group_Off_Confirmation **$809D_{hex}$**

Bedeutung: Eine Gruppe von Bussegmenten wurde abgeschaltet. Als Parameter wird die Nummer der abgeschalteten Gruppe übergeben.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Gruppennummer

Bit	15 8	7 0

Legende:

Code:	Meldungs-Code (hier $809D_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Gruppennummer:	Nummer der Gruppe, die abgeschaltet wurde. (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

Switch_Group_On_Confirmation **$809E_{hex}$**

Bedeutung: Eine Gruppe von Bussegmenten wurde eingeschaltet. Als Parameter wird die Nummer der eingeschalteten Gruppe übergeben.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Gruppennummer

Bit	15 8	7 0

Legende:

Code:	Meldungs-Code (hier $809E_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Gruppennummer:	Nummer der Gruppe, die eingeschaltet wurde. (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

Switch_Group_On_Failed_Confirmation **$80C5_{hex}$**

Bedeutung: Der Versuch eine Gruppe einzuschalten ist nicht erfolgreich verlaufen. Als Parameter wird die Gruppennummer übergeben.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Gruppennummer

Bit	15 8	7 0

Legende:

Code:	Meldungs-Code (hier $80C5_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Gruppennummer:	Nummer der Gruppe die nicht eingeschaltet werden konnte (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez}).

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Quit_Groups_Error_Characteristic_Confirmation **$00F6_{hex}$**

Bedeutung: Das Kommando *Define_Groups_Error_Characteristic_Request* (0060_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00F6_{hex}$). Es folgen keine weiteren Parameter.

Groups_Error_Characteristic_Failed_Confirmation **$80F7_{hex}$**

Bedeutung: Das Kommando *Define_Groups_Error_Characteristic_Request* (0060_{hex}) wurde nicht fehlerfrei ausgeführt. Die darin gemachten Angaben sind nicht übernommen worden. Die Fehlernummer benennt die Art des Fehlers. Die Parameternummer kennzeichnet den fehlerhaften Parameter.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Fehlerummer
Wort 4	00_{hex}	Parameternummer

Bit	15 8	7 0

Legende:

Code: Meldungs-Code (hier $80F7_{hex}$)
Parameterzähler: Anzahl der nachfolgenden Worte (hier 2)
Fehlerummer: Der Parameter *Fehlernummer* benennt die Art des Fehlers:
1 Es sind Elemente doppelt vorhanden.
2 Die Gruppennummer existiert nicht.
Parameternummer: Der Parameter *Parameternummer* kennzeichnet den fehlerhaften Parameter im Kommando *Define_Groups_Error_Characteristic_Request* (0060_{hex}).

Check_Configuration_Error_Confirmation **0068_{hex}**

Bedeutung: Dieser Fehler wird gemeldet, wenn beim Kommando *Check_Physical_Configuration_Map_Request* (0058_{hex}) die zur Anschaltbaugruppe übertragene und die aktuell angeschlossene Buskonfiguration nicht übereinstimmen.

Abhilfe: Überprüfen Sie die ID-Codeliste auf
- die Anzahl der Parameter,
- Übereinstimmung der Reihenfolge von ID-Codes und Modulen sowie
- die Positionierung der Register und Sonder-ID-Codes.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0068_{hex}). Es folgen keine weiteren Parameter.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Unknown_Bus_Module_Confirmation **$005B_{hex}$**

Bedeutung: Im Kommando *Check_Physical_Configuration* (0058_{hex}) wurde ein nicht zugelassener Längencode angegeben;
Es wurde eine nicht zulässige Anzahl von Prozeßdaten angegeben;
Das erste Modul ist kein Modul mit Busklemmenfunktionalität.

Abhilfe: Überprüfen Sie die Parametrierungsliste der ID-Codes auf nicht zugelassene ID-Codes und nicht existierende Längencodes und die Anzahl der Prozeßdaten.

Der Bus geht ohne *Reset* in den STOP-Zustand. Die Ausgabedaten werden nicht zurückgesetzt. Der Datenzyklus des Busses kann nur durch das Kommando *Warmstart_Request* ($004C_{hex}$) oder den Reset-Taster wieder initialisiert werden.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($005B_{hex}$). Es folgen keine weiteren Parameter.

Physical_Configuration_Map_Valid_Confirmation **$00AB_{hex}$**

Bedeutung: Die der Anschaltbaugruppe mit dem Kommando *Receive_Physical_Configuration_Map_Request* übermittelte Konfiguration des Busses stimmt mit der aktuell angeschlossenen Konfiguration überein.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00AB_{hex}$). Es folgen keine weiteren Parameter.

Send_Physical_Configuration_Confirmation **$80F4_{hex}$**

Bedeutung: Positive Quittung auf das Kommando *Send_Physical_Configuration_Request* ($005E_{hex}$). Sie enthält in Form von Längen- und ID-Codes die im RAM der Anschaltbaugruppe abgespeicherte physikalische Buskonfiguration.

Syntax:

Wort 1	Code		
Wort 2	Parameterzähler (n)		
Wort 3	Längen-Code	ID-Code	für Teilnehmer Nr. 1
Wort 4	Längen-Code	ID-Code	für Teilnehmer Nr. 2
Wort 5	Längen-Code	ID-Code	für Teilnehmer Nr. 3
	...	...	
Wort n+2	Längen-Code	ID-Code	für Teilnehmer Nr. n
Bit	15 8	7 0	

Legende:

Code:	Meldungs-Code (hier $80F4_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).
Längen-Code:	Der Längen-Code beschreibt den Adreßraumbedarf des IBS-Teilnehmers im Host.
ID-Code:	Identifikations-Code des IBS-Teilnehmers. Er ist als *Module Ident* den Modulen in dezimaler Form aufgedruckt. (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Der Längen-Code von IBS-Teilnehmern, die Kommunikation (PCP) unterstützen, beinhaltet nur deren Prozeßdatenworte.

Send_Actual_Configuration_Confirmation **8119_{hex}**

Bedeutung: Positive Quittung auf das Kommando *Send_Actual_Configuration_Request* ($010D_{hex}$). Sie enthält in Form von Längen- und ID-Codes die aktuell angeschlossene Buskonfiguration.

Syntax:

Wort 1	Code		
Wort 2	Parameterzähler (n)		
Wort 3	Result	00_{hex}	
Wort 4	Längen-Code	ID-Code	für Teilnehmer Nr. 1
Wort 5	Längen-Code	ID-Code	für Teilnehmer Nr. 2
Wort 6	Längen-Code	ID-Code	für Teilnehmer Nr. 3
	...	...	
Wort n+2	Längen-Code	ID-Code	für Teilnehmer Nr. n

Bit	15 8	7 0

Legende:

Code:	Meldungs-Code (hier 8119_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Teilnehmer).
Result:	Der Parameter *Result* ist immer 00_{hex}. Wenn die Anschaltbaugruppe nur einen Teil der aktuell angeschlossenen Buskonfiguration einlesen kann (weil z.B das Fernbuskabel irgendwo defekt ist), enthält die Meldung nur die Längen- und ID-Codes der erreichbaren IBS-Teilnehmer. Falls die Anschaltbaugruppe die aktuell angeschlossene Buskonfiguration gar nicht einlesen konnte (weil z.B das Fernbuskabel nicht an der Anschaltbaugruppe angeschlossen ist), erfolgt ebenfalls diese Meldung, allerdings ohne Längen- und ID-Codes.
Längen-Code:	Der Längen-Code beschreibt den Adreßraumbedarf des IBS-Teilnehmers im Host.
ID-Code:	Identifikations-Code des IBS-Teilnehmers. Er ist als *Module Ident* den Modulen in dezimaler Form aufgedruckt. (0 bis 255_{dez} entspricht 00_{hex} bis FF_{hex})

Der Längen-Code von IBS-Teilnehmern, die Kommunikation (PCP) unterstützen, beinhaltet nur deren Prozeßdatenworte.

Quit_Configure_Bus_Confirmation **$00CA_{hex}$**

Bedeutung: Das Kommando *Configure_Bus_Request* (0023_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00CA_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.3 Meldungen zur Adressierung

Logical_Address_Error_Confirmation **$002B_{hex}$**

Bedeutung: In den Listen der logischen Adressierung, die zur Anschaltbaugruppe geschickt wurden, ist bei der Überprüfung ein Fehler erkannt worden.

Der Bus geht ohne Reset in den STOP-Zustand. Die Ausgänge werden nicht zurückgesetzt.

Ergänzend bietet das Kommando *Send_Log_Address_Error_Request* ($005F_{hex}$) die Möglichkeit der Fehlerdiagnose.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($002B_{hex}$). Es folgen keine weiteren Parameter.

Quit_Receive_Logical_Localbus_Addressmap_Confirmation **$00D0_{hex}$**

Bedeutung: Das Kommando *Receive_Logical_Localbus_Addressmap_Request* (0039_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00D0_{hex}$). Es folgen keine weiteren Parameter.

Quit_Receive_Logical_In_Addressmap_Confirmation **$00D1_{hex}$**

Bedeutung: Das Kommando *Receive_Logical_In_Addressmap_Request* ($003A_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00D1_{hex}$). Es folgen keine weiteren Parameter.

Quit_Receive_Logical_Out_Addressmap_Confirmation **$00D2_{hex}$**

Bedeutung: Das Kommando *Receive_Logical_Out_Address_Map_Request* ($003B_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00D2_{hex}$). Es folgen keine weiteren Parameter.

Quit_Implement_Confirmation **$00D3_{hex}$**

Bedeutung: Das Kommando Implement_All_Logical_Address_Maps_Request (0040_{hex}) wurde ausgeführt. Die Listen der logischen Adressierung, die zur Anschaltbaugruppe geschickt wurden, sind fehlerfrei.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00D3_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Send_Logical_Address_Error_Confirmation 80F5$_{hex}$

Bedeutung:

Diese Meldung ist die positive Quittung auf das Kommando *Send_Log_Address_Error_Request* (005F$_{hex}$) und übermittelt den letzten bei der logischen Adressierung aufgetretenen Fehler. Falls kein Fehler erkannt wurde, sind die Parameter *Liste*, *Parameternummer* und *Fehlernummer* gleich 0.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00$_{hex}$	Liste
Wort 4	00$_{hex}$	Parameternummer
	00$_{hex}$	Fehlernummer

Bit	15 8	7 0

Legende:

Code: Meldungs-Code (hier 80F5$_{hex}$)

Parameterzähler: Anzahl der nachfolgenden Worte

Liste: Der Parameter *Liste* kennzeichnet die fehlerhafte Liste:

0 kein Fehler erkannt
2 Logische IN-Adreßliste (Kommando *Receive_Logical_IN_Address_Map_Request* (003A$_{hex}$))
3 Logische OUT-Adreßliste (Kommando *Receive_Logical_OUT_Address_Map_Request* (003B$_{hex}$))
4 ID-Liste (Kommando *Check_Physical_Configuration_Request* (0058$_{hex}$))
6 Logical-Bus-Code-Map (Kommando Receive_Localbus_Code_Map_Request (0069$_{hex}$)

Parameternummer: Der Parameter *Parameternummer* kennzeichnet den fehlerhaften Parameter in dieser Liste:

- Ist die Fehlernummer 1, nennt Parameternummer die Nummer des ersten fehlenden Parameters in der Liste der logischen Adressen.
- Ist die Fehlernummer 2, nennt Parameternummer die Nummer des ersten Parameters in der Liste der logischen Adressen, der zuviel angegeben wurde.
- andernfalls wird die Nummer des fehlerhaften Eintrages angegeben.

Fehlerummer: Der Parameter *Fehlernummer* benennt die Art des Fehlers:

0 kein Fehler erkannt
1 Zu wenige Einträge in der Liste, die mit dem fehlerhaften Kommando gesendet wurde.
2 Zu viele Einträge in der Liste, die mit dem fehlerhaften Kommando gesendet wurde.
3 Es wurde eine unzulässige Bussegmentnummer vergeben. Zulässig sind 0 bis 255$_{dez}$ (entspricht 00$_{hex}$ bis FF$_{hex}$).
4 Es wurde eine Bussegmentnummer mehrfach vergeben.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

5 Unzulässiger Host-Adreßbereich
7 Die Adreßbereiche von zwei IBS-Teilnehmern überlappen sich.
8 Es wurde eine ungerade Adresse für einen IBS-Teilnehmer mit mehr als 8 Bit Prozeßdaten vergeben. Vergeben Sie eine gerade Adresse.
9 Es wurde ein unzulässiger Längencode verwendet.
10 Ein gesendeter ID-Code stimmt nicht mit dem in der Anlage vorhandenem IBS-Teilnehmer überein.
12 Die angegebene Adresse ist unzulässig.
14 Es wurde eine unzulässige Fernbusebene angegeben. Unter Firmware 3.x sind nur die Fernbusebenen *0* und *1* zulässig.
15 Es wurde für einen IBS-Teilnehmer die falsche Fernbusebene angegeben. Geben Sie für IBS-Teilnehmer im Hauptstrang die *Fernbusebene 0* und für IBS-Teilnehmer in Fernbusstich die *Fernbusebene 1* an.
16 Kommandoausführung nicht möglich, da der Bus im Run-Zustand ist. In diesem Fall sind die Parameter *Liste* und *Parameternummer* gleich 0.
17 Es wurde eine unzulässige Datenkonsistenz angegeben.

Quit_Receive_Localbus_Code_Map_Confirmation **0105_{hex}**

Bedeutung: Das Kommando *Receive_Localbus_Code_Map_Request* (0069_{hex}) wurde erfolgreich ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0105_{hex}). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.4 Meldungen für den Betrieb

Start_Bus_Confirmation **0088_{hex}**

Bedeutung: Der zyklische Datenverkehr auf dem Bus ist gestartet. Die Anschaltbaugruppe betreibt ab diesem Zeitpunkt den Prozeßdatenverkehr. Dies bedeutet zyklisches Erstellen eines Prozeßabbilds der Eingangsdaten für die Steuerung und zyklischer Transfer der Prozeßausgabedaten von der Steuerung zu den am Bus angeschlossenen Modulen. Außerdem betreibt die Anschaltbaugruppe das **P**eripherals-**C**ommunication-**P**rotocol (PCP) für Module, die mit PCP arbeiten.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0088_{hex}). Es folgen keine weiteren Parameter.

Stop_Bus_Confirmation **$00C6_{hex}$**

Bedeutung: Das Kommando *Stop_Bus_Cycle_Request* (0002_{hex}) wurde ausgeführt. Der Bus befindet sich im STOP-Zustand.

Prozeßdaten: Der zyklische Datenverkehr auf dem Bus wurde gestoppt. Das bestehende Prozeßdaten-Abbild der am Bus angeschlossenen Module bleibt statisch erhalten und wird nicht erneuert.

Das Kommando schaltet die Prozeßausgabedaten **nicht** in den sicheren Zustand (Rücksetzen der Ausgänge). Dies ist nur mit dem Kommando *Alarm_Stop_Request* ($004A_{hex}$) möglich.

PCP-Kanal: Das **P**eripherals-**C**ommunication-**P**rotocol (PCP) wird nicht weiter betrieben. Es erfolgt kein automatischer Abbau der hergestellten Verbindungen. Eventuell ausstehende Services werden nach dem erneuten Starten des Datenverkehrs weiter bearbeitet.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00C6_{hex}$). Es folgen keine weiteren Parameter.

Quit_Alarm_Stop_Confirmation **$00D8_{hex}$**

Bedeutung: Das Kommando *Alarm_Stop_Request* ($004A_{hex}$) wurde erfolgreich ausgeführt. Der Bus befindet sich im STOP-Zustand.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00D8_{hex}$). Es folgen keine weiteren Parameter.

Start_Bus_Not_Possible_Confirmation **$00E3_{hex}$**

Bedeutung: Dies ist eine negative Quittung auf das Kommando *Start_Bus_Cycle_Request* (0001_{hex}). Sie wird gesendet, wenn kein Datenverkehr auf dem Bus möglich ist.

Ursachen: Hardwarefehler auf der Anschaltbaugruppe;
Konfigurationsänderung am Fernbus (es wurden Module entfernt oder hinzugefügt).

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00E3_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Quit_Bus_Delay_Confirmation **$00EC_{hex}$**

Bedeutung: Das Kommando *Bus_Delay_Request* (0059_{hex}) wurde erfolgreich ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00EC_{hex}$). Es folgen keine weiteren Parameter.

9.5 Meldungen zur Fehlerbehandlung

Quit_Clear_Display_Confirmation **$00E2_{hex}$**

Bedeutung: Das Kommando *Clear_Display_Request* ($004E_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00E2_{hex}$). Es folgen keine weiteren Parameter.

No_Command_Code_Indication **0007_{hex}**

Bedeutung: Es wurde ein nicht definierter Kommando-Code zur Anschaltbaugruppe geschickt.

Abhilfe: Überprüfen Sie das Anwendungsprogramm.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0007_{hex}). Es folgen keine weiteren Parameter.

Command_Write_Error_Indication **0008_{hex}**

Bedeutung: Es wurde ein Kommando mit einer falschen Anzahl von Parametern zur Anschaltbaugruppe geschickt.

Abhilfe: Überprüfen Sie das Anwendungsprogramm (z.B. PCP-Kommandos)

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0008_{hex}). Es folgen keine weiteren Parameter.

Parameter_Write_Error_Indication **0009_{hex}**

Bedeutung: Ein Kommando mit Parametern wurde mit dem Parameterzähler *0* gesendet.

Abhilfe: Überprüfen Sie die Parameter im Anwenderprogramm.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0009_{hex}). Es folgen keine weiteren Parameter.

No_Command_Parameter_Routine_Indication **$000A_{hex}$**

Bedeutung: Es wurde ein unbekanntes Kommando geschickt.

Abhilfe: Überprüfen Sie das Anwendungsprogramm .

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($000A_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

No_Error_Code_Indication **$000B_{hex}$**

Bedeutung: Unbekannter Fehlercode aufgetreten.

Abhilfe: Setzen sie sich mit Phoenix Contact in Verbindung.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($000B_{hex}$). Es folgen keine weiteren Parameter.

Controller_Parameter_Timeout_Indication **$000C_{hex}$**

Bedeutung: Die Fehlermeldung wird ausgelöst, wenn:
- eine anstehende Meldung nicht innerhalb der ersten 8 Minuten abgeholt (*TIME-OUT*) wurde und somit den entsprechenden MPM-Bereich blockiert;
- eine anstehende Meldung nicht abgeholt wurde und eine zweite bereits 8,1 Sekunden ansteht.

Abhilfe: Überprüfen Sie, ob in Ihrem Programm in ausreichend kurzen Abständen das Anstehen einer Meldung abgefragt wird.

Die Meldung *Controller_Parameter_Timeout_Indication* überschreibt alle Meldungen, die nach der nicht aus dem MPM abgeholten Meldung aufgetreten sind, auf dem IBS-Masterboard zur Übergabe an das MPM anstehen, aber nicht übergeben werden konnten, da die erste nicht abgeholte Meldung den MPM-Bereich blockiert.

Weitere Fehlermeldungen werden durch diese Meldung gesperrt. Eine Bus-Diagnose ist nur noch mit der Funktion *GetIBSDiagnostics* möglich.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($000C_{hex}$). Es folgen keine weiteren Parameter.

Host_Parameter_Timeout_Indication **$000D_{hex}$**

Bedeutung: Die Anschaltbaugruppe kann eine eingegangenes Kommando nicht quittieren, weil der Host das Handshake-Bit für die vorherige Kommando-Quittung noch nicht zurückgenommen hat. Diese Handshake-Bit wurde beim Senden des vorherigen Kommando vom Host gesetzt.

Abhilfe: Überprüfen Sie das Anwendungsprogramm.

Weitere Fehlermeldungen werden durch diese Meldung gesperrt. Eine Bus-Diagnose ist nur noch mit der Funktion *GetIBSDiagnostics* möglich.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($000D_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

CPU_Bus_Error_Indication **0023_{hex}**

Bedeutung: Auslösen des Watchdogs

Abhilfe: Setzen sie sich mit Phoenix Contact in Verbindung.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0023_{hex}). Es folgen keine weiteren Parameter.

Bus_System_Error_Indication **0038_{hex}**

Bedeutung: Im IBS-System ist ein Fehler aufgetreten, der dazu führt, daß das System mit der letzten Konfiguration nicht mehr lauffähig ist. Der Fehler wird von der Anschaltbaugruppe sofort gemeldet. Anschließend läuft automatisch eine Testroutine ab, die den Fehler im System lokalisiert.

Eine detaillierte Fehlermeldung erfolgt nach Abschluß dieser Testroutine über die Meldung *Bus_Error_Information_Indication* ($80C4_{hex}$)

Der Fehler wird nur als Meldung abgesetzt. Während der Fehlerlokalisierung wird der Datenzyklus des Busses nicht betrieben.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0038_{hex}). Es folgen keine weiteren Parameter.

No_Executable_Configuration_Confirmation **$004A_{hex}$**

Bedeutung: Eine nicht vorhandene Konfiguration wurde zur Anschaltbaugruppe gesendet.

Abhilfe: Überprüfen Sie den Anschluß des Fernbuskabels an der Anschaltbaugruppe und lösen Sie einen Reset auf der Anschaltbaugruppe aus.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($004A_{hex}$). Es folgen keine weiteren Parameter.

Command_Parameter_Error_Confirmation **$004B_{hex}$**

Bedeutung: Die Anzahl der Parameter für das direkt vor dieser Meldung abgesendete Kommando ist zu groß oder zu klein.

Die Kommandoausführung wird abgebrochen.

Abhilfe: Überprüfen Sie in Ihrem Programm das zuletzt gesendete Kommando auf eine falsche Parameteranzahl.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($004B_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

IPMS_No_Error_Indication **$004D_{hex}$**

Bedeutung: Der IPMS-Protokoll-Chip hat einen Busfehler erkannt, es wurde aber kein Bit im IPMS-Error-Register gesetzt.

Abhilfe: Setzen sie sich mit Phoenix Contact in Verbindung.

Führen Sie auf der Anschaltbaugruppe einen Reset aus.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($004D_{hex}$). Es folgen keine weiteren Parameter.

Communication_Invalid_Indication **$004E_{hex}$**

Bedeutung: Die Anschaltbaugruppe benutzt für die Kommandobearbeitung einen festgelegten Speicherbereich. Sind diese Ressourcen erschöpft, wird diese Fehlermeldung generiert. Das kann z.B. der Fall sein, wenn eine große Anzahl Kommandos in kurzer Zeit abgesetzt wird und die Anschaltbaugruppe nicht in der Lage ist, diese schnell genug abzuarbeiten.

Abhilfe: Überprüfen Sie Ihr Programm.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($004E_{hex}$). Es folgen keine weiteren Parameter.

BK_Alarm_Failed_Confirmation **$005A_{hex}$**

Bedeutung: Eines der folgenden Kommandos wurde nicht erfolgreich ausgeführt:
- Set-BK-Alarm-Logical-Request (0024_{hex})
- Reset-BK-Alarm-Logical-Request (0025_{hex})
- Set-BK-Alarm-Physical-Request (0026_{hex})
- Reset-BK-Alarm-Physical-Request (0027_{hex})

Mögliche Fehlerursachen sind:
a) Sie haben eine nicht existierende Bussegmentnummer angegeben.
b) Sie haben eine unzulässige Anzahl von Parametern angegeben.

Abhilfe: Überprüfen Sie Ihr Anwenderprogramm.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($005A_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Receive_CR_Error_Confirmation **0066_{hex}**

Bedeutung: Dieser Fehler wird beim Überprüfen des Kommandos *Receive Communication Reference* gesendet, wenn
- eine ungültige Kommunikationsreferenz benutzt wurde,
- zu viele Parameter gesendet wurden oder
- zu wenige Parameter gesendet wurden.

Abhilfe: Überprüfen Sie die Kommunikationsbeziehungsliste (KBL) auf zugelassene Kommunikationsreferenzen (2 bis 62). Die vergebenen Kommunikationsreferenzen müssen lückenlos (z.B. 2, 3, 4, 5,...) aufeinander folgen. Die Reihenfolge des Aufrufs ist nicht von Bedeutung (z.B. auch 4, 2, 5, 3 ist erlaubt).

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0066_{hex}). Es folgen keine weiteren Parameter.

Communication_Not_Ready_Confirmation **0069_{hex}**

Bedeutung: Obwohl die Kommunikation (kurz PCP) noch nicht initialisiert worden ist, wurde ein Request-, oder Response-Kommando gesendet.

Abhilfe: Initialisieren Sie in Ihrem Anwenderprogramm die Kommunikation.

Siehe Handbuch zu Grundlagen und Anwendung des Peripherals-Communication-Protokolls *IBS PCP UM* (Artikel-Nr. 27 53 92 8).

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0069_{hex}). Es folgen keine weiteren Parameter.

No_Map_Entry_Confirmation **$00ED_{hex}$**

Bedeutung: Diese Meldung wird gesendet, wenn bei Kommandos, die in irgendeiner Form eine Liste oder Tabelle senden sollen, keine Einträge vorhanden sind.

Abhilfe: Überprüfen Sie Ihr Anwenderprogramm

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00ED_{hex}$). Es folgen keine weiteren Parameter.

Quit_Module_Error_Ok_Confirmation **$00FE_{hex}$**

Bedeutung: Diese Meldung ist die positive Quittung auf die Kommandos *Quit_Module_Error_Request* (0064_{hex}) und *Quit_Module_Error_All_Request* (0065_{hex}).

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00FE_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Module_Error_Indication $80A0_{hex}$

Bedeutung: Auf einem oder mehreren IBS-Teilnehmern mit Fehlerrückmeldung wurde ein Fehler festgestellt. Die Parameter geben, je nach angewandter Adressierungsart, die logischen oder die physikalischen Adressen der betroffenen Bussegmente an.

Ursachen:
- Peripheriespannungsausfall
- Sicherungsdefekt
- Ansprechen der Strombegrenzung

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler (n)	
Wort 3	00_{hex}	Bussegment
Wort 4	00_{hex}	Bussegment
	...	
Wort n+2	00_{hex}	Bussegment

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80A0_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Bussegmente, in denen ein Modulfehler festgestellt wurde)
Bussegment:	Bussegmentnummer des betroffenen Bussegments (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

Power_Fail_Indication $80A1_{hex}$

Bedeutung: An einer Busklemme ist die Spannungsversorgung (9 Volt) eines Peripheriebusses ausgefallen. Die Parameter geben, je nach angewandter Adressierungsart, die logischen oder die physikalischen Adressen der betroffenen Bussegmente an.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Bussegment

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80A1_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Bussegment:	Bussegmentnummer des betroffenen Bussegments (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Battery_Indication **$80A2_{hex}$**

Bedeutung: Auf einigen Sondermodulen ist die Logikspannung (U_L) mit einem Akku gepuffert. Fällt diese Versorgungsspannung aus, so wird das Umschalten auf Akkubetrieb durch diese Meldung angezeigt. Der Parameter gibt die logische oder physikalische Bussegmentnummer der betroffenen Busklemme an.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Bussegment

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80A2_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Bussegment:	Bussegmentnummer des betroffenen Bussegments (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

Quit_Module_Error_Not_Possible_Confirmation **$80FF_{hex}$**

Bedeutung: Diese Meldung ist die negative Quittung auf die Kommandos *Quit_Module_Error_Request* (0064_{hex}) und *Quit_Module_Error_All_Request* (0065_{hex}).

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Parameternummer
Wort 4	Fehlerummer

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80FF_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 2)
Parameternummer:	Der Parameter *Parameternummer* kennzeichnet den fehlerhaften Parameter.
Fehlerummer:	Der Parameter *Fehlernummer* benennt die Art des Fehlers: 1 Ungerade Anzahl von Parametern (Parameternummer gleich 1). 2 Unbekannte Bussegment-Adresse. 3 Das angegebene Modul (Nummer) existiert nicht. 4 Das angegebene Modul ist zur Zeit nicht erreichbar, da das entsprechende Segment abgeschaltet ist. 5 Die notwendigen ID-Zyklen konnten nicht erfolgreich durchgeführt werden.

Bei der Antwort auf das Kommando *Quit_Module_Error_All_Request* (0065_{hex}) wird nur die Fehlernummer *5* benutzt.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Bus_Error_Information_Indication 80C4$_{hex}$

Es folgt zunächst der allgemeine Aufbau der Meldung. Der Aufbau ist abhängig von dem Parameter *Fehlerart*, der im Detail weiter unten beschrieben wird.

Bedeutung: Diese Meldung folgt auf die Fehlermeldung *Bus_System_Error_Indication* (0038$_{hex}$). Sie gibt eine detailierte Information über den Fehlerzustand des Bussystems.

Folge: Der Bus befindet sich im STOP-Zustand und die Ausgänge sind auf *0* gesetzt.

Syntax:

Wort 1	Code
Wort 2	Parameterzähler (n)
Wort 3	Fehlerart
Wort 4	Fehlerbeschreibung
	...
Wort n+2	Fehlerbeschreibung

Bit	15 0

Legende:

Code:	Meldungs-Code (hier 80C4$_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte
Fehlerart:	EE01, EE02, EE03, EE04, EE05 oder EE06. In Abhängigkeit der Fehlerart schlüsseln die weiteren Parameter als Folgeinformation den Fehler auf.
Fehlerbeschreibung:	Optionale Folgeinformationen bei den Fehlerarten EE02 und EE03. Die einzelnen Parameter werden in den nachstehenden Beschreibungen zu den einzelnen Fehlerarten erklärt.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Fehlerart EE01

Bedeutung: Bei der Überprüfung des Busses wurde kein Fehler in der augenblicklich zulässigen Konfiguration festgestellt. Es liegt aber ein Fehler in der Businstallation vor.

Ursache: Eine kurze Störung der Datenübertragung.

Abhilfe: Überprüfen Sie die Anlage auf:
- fehlende oder nicht korrekte Schirmung der Buskabel (Steckverbinder),
- fehlende oder nicht korrekte Erdung/Potentialausgleich,
- Spannungseinbrüche auf der Logikversorgung der Fernbusteilnehmer.

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Fehlerart

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80C4_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Fehlerart:	hier EE01

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Fehlerart EE02

Bedeutung: Konfigurationsänderungen, die einen Weiterbetrieb des Datenverkehrs über den Bus nicht zulassen.

Ursache:
- Die maximal zulässige Anzahl der IBS-Worte wurde überschritten
- Die maximal zulässige Anzahl der IBS-Module wurde überschritten

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Fehlerart
Wort 4	Fehlenummer

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80C4_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 2)
Fehlerart:	hier EE02
Fehlernummer:	Die einzelnen System-Fehler sind im Wort 4 als Fehlernummer kodiert. Die folgende Tabelle führt alle möglichen Fehlernummern mit einer Kurzbeschreibung des Fehlers auf.

Tabelle 9-2: Bedeutung des Parameters *Fehlernummer*

Fehlernummer	Bedeutung
$DD01_{hex}$	Ein im RAM der Anschaltbaugruppe gespeicherter ID-Code paßt nicht zur zugehörigen Busklemme in der Anlage (z.B. aufgrund Modulaustausch oder Gerätefehler).
$DD02_{hex}$	Ein im RAM der Anschaltbaugruppe gespeicherter ID-Code paßt nicht zur zugehörigen E/A-Busklemme in der Anlage (z.B. aufgrund Modulaustausch oder Gerätefehler).
$DD03_{hex}$	Ein angegebener Peripheriebus ist in der Anlage nicht vorhanden oder ein Peripheriebuskabel ist defekt.
$DD04_{hex}$	Ein Peripheriebus beinhaltet mehr Teilnehmer als erwartet, weil z.B. ein Teilnehmer während des Betriebs hinzugefügt wurde.
$DD05_{hex}$	Ein Peripheriebus beinhaltet weniger Teilnehmer als erwartet, weil z.B. ein Teilnehmer entfernt wurde.
$DD06_{hex}$	Ein im RAM der Anschaltbaugruppe gespeicherter ID-Code paßt nicht zu einem in der Anlage vorhandenen Peripheriebusteilnehmer (z.B. aufgrund Modulaustausch oder Gerätefehler)
$DD07_{hex}$	Die Konfiguration konnte nicht eingelesen werden, weil das Fernbuskabel nicht an der Anschaltbaugruppe angeschlossen ist.
$DD08_{hex}$	An einer Busklemme mit E/A-Funktionalität ist ein Peripheriebus angeschlossen, der nicht im RAM der Anschaltbaugruppe gespeichert ist (weil z.B. Peripheriebusteilnehmer während des Betriebs hinzugefügt wurden).

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Tabelle 9-2: Bedeutung des Parameters *Fehlernummer*

Fehlernummer	Bedeutung
$DD09_{hex}$	Die angeschlossenen Konfiguration ist kürzer als erwartet, weil der Fernbus gegenüber der im RAM der Anschaltbaugruppe gespeicherten Konfiguration verkürzt wurde (Fernbusteilnehmer entfernt oder Fernbuskabel nicht angeschlossen).
$DD0A_{hex}$	Mehrfache Übertragungsstörungen zwischen 2 fehlerfreien Datenzyklen (entspricht Fehlerart EE06). Beim Erfassen und Vergleichen des Busaufbaus wurde kein Fehler in der Konfiguration festgestellt.
$DD0B_{hex}$	Die angeschlossenen Konfiguration ist länger als erwartet, weil der Fernbus gegenüber der im RAM der Anschaltbaugruppe gespeicherten Konfiguration erweitert wurde.
$DD0C_{hex}$	Die maximal zulässige Konfiguration ist überschritten. Es sind maximal 256 Busklemmen und 256 Worte (Summe aller Registerlängen und der PCP-Worte) auf dem Datenring zulässig.
$DD11_{hex}$	- Das Datenregister eines Fernbusteilnehmers ist unterbrochen. - Ein im RAM der Anschaltbaugruppe gespeicherter Längencode paßt nicht zum zugehörigen Fernbusteilnehmer in der Anlage (z.B. aufgrund Modulaustausch oder Gerätefehler).
$DD12_{hex}$	- Das Datenregister eines Peripheriebusteilnehmers unterbrochen - Ein im RAM der Anschaltbaugruppe gespeicherter Längencode paßt nicht zum zugehörigen Peripheriebusteilnehmer in der Anlage (z.B. aufgrund Modulaustausch oder Gerätefehler).
$DD15_{hex}$	Kurzzeitige Veränderung eines ID-Codes im Betrieb bei angegebebener Busklemme oder Modul in angegebebenem Peripheriebus (ähnlich zu DD01/02/06)
$DD18_{hex}$	Kurzzeitiger Fehler in einem Peripheriebus (Achtleitertechnik) während des Betriebs bedingt durch Kabel- bzw. Modulfehler (ähnlich zu DD03/05/08).
$DD19_{hex}$	Es wurde eine Busunterbrechung oder der Spannungsreset eines Teilnehmers in der zusätzlichen Diagnosephase erkannt.
$DD1A_{hex}$	Es wurden mehrfache Übertragungsstörungen zwischen 2 fehlerfreien Datenzyklen (entspricht Fehlerart EE06) in der zusätzlichen Diagnosephase erkannt.
$DD2B_{hex}$	Die angeschlossenen Konfiguration ist länger als erwartet, weil der Fernbus gegenüber der im RAM der Anschaltbaugruppe gespeicherten Konfiguration erweitert wurde. Der Fehler wurde in der zusäztlichen Diagnosephase erkannt.
$DD42_{hex}$	Kurzzeitige Busunterbrechung aufgrund des Spannungsreset eines Teilnehmers oder Brücke (RBST) in einem weiterführenden Fernbusstecker defekt.
$DD50_{hex}$	Es wurde ein Fern- oder Peripheriebusfehler in der zusätzlichen Diagnosephase erkannt.
$DD51_{hex}$	Es wurde ein Peripheriebusfehler in der zusätzlichen Diagnosephase erkannt.
$DD52_{hex}$	Es wurde ein Fernbusfehler in der zusätzlichen Diagnosephase erkannt.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Fehlerart EE03

Bedeutung: Während des laufenden Betriebs des Datenverkehrs auf dem Bus trat ein Fehler auf, der auf eine elektrische Veränderung der Buskonfiguration (z.B. Kabelbruch) zurückzuführen ist.

Ursache:

Fernbus:	Busklemme oder vorgelagertes Fernbuskabel defekt
Peripheriebus:	Peripheriebuskabel oder Modul defekt

Folgen: Der Datenzyklus ist ausgeschaltet. Es sei denn, die Fehler traten nur in den Gruppen auf, für die zuvor mit dem Kommando *Define_Groups_Error_Characteristic_Request* (0060_{hex}) definiert wurde, daß der restliche Busaufbau ohne sie weiter betrieben werden darf.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler (n)	
Wort 3	Fehlerart	
Wort 4	$FF01_{hex}$	Parameterblock *Gruppen*
Wort 5	Gruppenanzahl	
Wort 6	1. Gruppen-Nr.	
Wort 7	...	
Wort 8	x. Gruppen-Nr.	
	...	
	...	
	$FF02_{hex}$	Parameterblock *Fernbus*
	RB-Anzahl	
	1. Fernbus-Nr.	
	1. Fehler-Nr.	
	...	
	x. Fernbus-Nr.	
	x. Fehler-Nr.	
	...	
	...	
	$FF03_{hex}$	Parameterblock *Peripheriebus*
	Peripheriebusanzahl	
	1. Peripheriebus-Nr.	
	1. Fehler-Nr.	
Wort n	...	
Wort n+1	x. Peripheriebus-Nr.	
Wort n+2	x. Fehler-Nr.	

Bit	15 0

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Legende:	Code:	Meldungs-Code (hier $80C4_{hex}$)
	Parameterzähler:	Anzahl der nachfolgenden Worte
	Fehlerart:	hier EE03
	FF01	Beginn des Parameterblocks, der die fehlerhaften Gruppen anzeigt.
	Gruppenanzahl:	Anzahl der fehlerhaften Gruppen.
	1. Gruppen-Nr:	Nummer der 1. fehlerhaften Gruppe.
	x. Gruppen-Nr:	Nummer der letzten fehlerhaften Gruppe.
	FF02	Beginn des Parameterblocks, der die fehlerhaften Fernbussegmente anzeigt.
	Fernbusanzahl:	Anzahl der fehlerhaften Fernbussegmente
	1. Fernbus-Nr.:	Bussegmentnummer des 1. fehlerhaften Fernbusses
	1. Fehler-Nr.:	Diese Fehlernummer beschreibt den Fehler des in der vorherigen Zeile angegebenen Fernbusses (siehe Tabelle 9-2).
	x. Fernbus-Nr.:	Bussegmentnummer des letzten fehlerhaften Fernbusses
	x. Fehler-Nr.:	Diese Fehlernummer beschreibt den Fehler des in der vorherigen Zeile angegebenen Fernbusses (siehe Tabelle 9-2).
	FF03:	Beginn des Parameterblocks, der die Bussegmente mit einem Fehler im Peripheriebus anzeigt.
	Peripheriebusanzahl:	Anzahl der Bussegmente mit einem Fehler im Peripheriebus.
	1. Peripheriebus-Nr.:	Bussegmentnummer des 1. fehlerhaften Peripheriebusses.
	1. Fehlernummer:	Diese Fehlernummer beschreibt den Fehler des in der vorherigen Zeile angegebenen Peripheriebusses (siehe Tabelle 9-2).
	x. Peripheriebus-Nr.:	Bussegmentnummer des letzten fehlerhaften Peripheriebusses.
	x. Fehlernummer:	Diese Fehlernummer beschreibt den Fehler des in der vorherigen Zeile angegebenen Peripheriebusses (siehe Tabelle 9-2).

Die Länge der Parameterblöcke *Gruppen*, *Fernbus* und *Peripheriebus* hängt von der Anzahl der aufgetretenen Fehler ab. Alle Einträge sind vierstellig hexadezimal.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Fehlerart EE04

Bedeutung: Die Buskonfiguration konnte nicht eingelesen werden.

Ursache: Die Spannungsversorgung für die Elektronik (Logikspannung) eines oder mehrerer Fernbusteilnehmer ist gestört.

Abhilfe: Überprüfen Sie die Anlage auf Spannungseinbrüche auf der Versorgung der Fernbusteilnehmer (Logikspannung).

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Fehlerart

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80C4_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Fehlerart:	hier EE04

Fehlerart EE05

Bedeutung: Alle Gruppen des InterBus-S sind abgeschaltet.

Ursache: Die letzte sich noch im Betrieb befindende Gruppe wurde abgeschaltet.

Abhilfe: Überprüfen Sie ihr Anwendungsprogramm.

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Fehlerart

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80C4_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Fehlerart:	hier EE05

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Fehlerart EE06

Bedeutung: Beim Erfassen und Vergleichen des Busaufbaus wurde kein Fehler in der Konfiguration festgestellt. Datenzyklen sind jedoch nicht möglich.

Ursache:
- Defekt eines IBS-Teilnehmers
- Installationsfehler (siehe auch Fehlerart EE01)
- Auftreten nicht lokalisierbarer Mehrfachfehler
- Nicht feststellbare Fehler

Abhilfe: Überprüfen Sie Ihren Busaufbau

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	Fehlerart

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80C4_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
Fehlerart:	hier EE06

Send_All_Module_Error_Confirmation **$80EF_{hex}$**

Bedeutung: Diese Meldung wird vom Kommando *Send_All_Module_Error_Request* ($005C_{hex}$) ausgelöst. Die Liste enthält alle Bussegmente, in denen ein Modul einen Fehler meldet.

Syntax:

Wort 1	Code
Wort 2	Parameterzähler (n)
Wort 3	1. Bussegment
Wort 4	2. Bussegment
	...
Wort n+2	x. Bussegment

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $80EF_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Bussegmente, in denen ein Modulfehler festgestellt wurde)
Bussegment:	Bussegmentnummer zur Angabe der Fehlerorte (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez})

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Localbus_Module_Error_Confirmation 80EE$_{hex}$

Bedeutung:

Diese Meldung wird vom Kommando *Send_Localbus_Module_Error_Request* (005B$_{hex}$) ausgelöst. Sie enthält eine Liste aller fehlermeldenden Module des im Kommando *Send_Localbus_Module_Error_Request* angegebenen Peripheriebusses (mit Position und ID-Code).

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler (n)	
Wort 3	1. Position	für 1. fehlermeldendes Modul
Wort 4	1. ID-Code	
	...	
	x. Position	für x. fehlermeldendes Modul
Wort n+2	x. ID-Code	

Bit	15 0

Legende:

Code:	Meldungs-Code (hier 80EE$_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Bussegmente, in denen ein Modulfehler festgestellt wurde, multipliziert mit 2)
Position:	Physikalische Platznummer im ausgewählten Peripheriebus: - Busklemme: 0; - Peripheriebusteilnehmer: 1 bis 8
ID-Code:	Identifikations-Code des Moduls. Er ist als *Module Ident* den Modulen in dezimaler Form aufgedruckt. (0 bis 255$_{dez}$ entspricht 00$_{hex}$ bis FF$_{hex}$)

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.6 Meldungen der Anwenderschnittstelle

Command_Disabled_Confirmation **$009C_{hex}$**

Bedeutung: Das zuvor abgesetzte Kommando ist nicht zur Benutzung freigegeben.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($009C_{hex}$). Es folgen keine weiteren Parameter.

Quit_Set_Parameter_Timeout_Confirmation **$00CF_{hex}$**

Bedeutung: Das Kommando *Set_Parameter_Timeout_Constant_Request* (0034_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code (0034_{hex}). Es folgen keine weiteren Parameter.

Quit_Enable_All_Messages_Confirmation **$00D6_{hex}$**

Bedeutung: Das Kommando *Enable_All_Messages_Request* (0047_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00D6_{hex}$). Es folgen keine weiteren Parameter.

Quit_Disable_All_Messages_Confirmation **$00D7_{hex}$**

Bedeutung: Das Kommando *Enable_All_Messages_Request* (0048_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00D7_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.7 Meldungen zur Systemkontrolle

Software_Revision_Confirmation **8087_{hex}**

Bedeutung: Ausgabe der Version der Anschaltbaugruppe. Die Daten der Parameter sind im ASCII-Code eingetragen.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler (n)	
Wort 3	Parameter 1	Parameterblock *Firmenname*
...	...	
Wort 10	Parameter 8	
Wort 11	Parameter 9	Parameterblock *Host-Typ*
...	...	
Wort 26	Parameter 24	
Wort 27	Parameter 25	Parameterblock *Anschaltbaugr.*
...	...	
Wort 42	Parameter 40	
Wort 43	Parameter 41	Parameterblock *Option*
...	...	
Wort 58	Parameter 56	
Wort 59	Parameter 57	Parameterblock *Firmware*
...	...	
Wort 65	Parameter 63	
Wort 66	Parameter 64	Parameterblock *Datum*
...	...	
Wort 79	Parameter 77	
Bit	15 ..... 8 | 7 0	

Legende:

Code:	Meldungs-Code (hier 8087_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier $4D_{hex}$ entspricht 77_{dez})
Firmenname:	Hersteller der Anschaltbaugruppe (ASCII-String von 16 Byte)
Host-Typ:	Steuerungs- oder Rechnersystem (ASCII-String von 32 Byte)
Anschaltbaugr.:	Anschaltbaugruppen-Type (ASCII-String von 32 Byte)
Option:	Weitere Information zum Anschaltbaugruppen-Typ (ASCII-String von 32 Byte)
Firmware:	Firmware-Version der Anschaltbaugruppe (ASCII-String von 14 Byte)
Datum:	Datum der Firmware-Version (ASCII-String von 28 Byte)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.8 Meldungen zur Prozeßdatenverknüpfung

Quit_Receive_Instructions_Confirmation **$00F2_{hex}$**

Bedeutung: Das Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00F2_{hex}$). Es folgen keine weiteren Parameter.

Processing_Instructions_Error_Confirmation **$80F3_{hex}$**

Bedeutung: Das Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$) konnte nicht erfolgreich ausgeführt werden.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler	
Wort 3	00_{hex}	Fehlerummer
Wort 4	00_{hex}	Parameternummer

Bit	15 8	7 0

Legende:

Code: Meldungs-Code (hier $80F3_{hex}$)
Parameterzähler: Anzahl der nachfolgenden Worte (hier 2)
Fehlerummer: Der Parameter *Fehlernummer* benennt die Art des Fehlers (siehe nachfolgendeTabelle).
Parameternummer: Der Parameter *Parameternummer* kennzeichnet den fehlerhaften Parameter (00_{hex} bis FF_{hex} entspricht 0 bis 255_{dez}) im Kommando *Receive_Processing_Instructions_Request* ($005D_{hex}$).

Tabelle 9-3: Bedeutung des Parameters *Fehlernummer*

Fehlernummer	Bedeutung
01_{hex}	Sämtlicher Speicher ist belegt. Es können keine weiteren Befehle aufgenommen werden. Der Parameter *Parameternummer* ist in diesem Fall gleich 0.
02_{hex}	Unbekannter Befehls-Code
03_{hex}	Zu wenige Parameter bei einem der letzten Befehle. Der Parameter *Parameternummer* ist in diesem Fall gleich 0.
04_{hex}	Kein internes RAM frei
05_{hex}	Reserviert
06_{hex}	Die logische Ausgabeadresse existiert nicht.
07_{hex}	Die logische Eingangsadresse existiert nicht.
08_{hex}	Ungültige Bitnummer. Es wurde eine Bitnummer größer 15 (bei Worten) bzw größer 7 (bei Bytes) angegeben.

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

9.9 Meldungen zur Event-Bearbeitung

Event_Error_Confirmation **$004C_{hex}$**

Bedeutung: Bei der Parameterüberprüfung zur Programmierung der Events ist ein Fehler erkannt worden. Der Fehler kann bei folgenden Kommandos auftreten:
- Receive_Event_Request ($002A_{hex}$)
- Enable_Event_Number_Request ($002B_{hex}$)
- Disable_Event_Number_Request ($002C_{hex}$)
- Enable_All_Event_Request ($002D_{hex}$)
- Ensable_Event_Logical_Adress_Request (0036_{hex})
- Disable_Event_Logical_Adress_Request (0037_{hex})

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($004C_{hex}$). Es folgen keine weiteren Parameter.

Quit_Receive_Events_Confirmation **$00BF_{hex}$**

Bedeutung: Das Kommando *Receive_Events_Request* ($002A_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00BF_{hex}$). Es folgen keine weiteren Parameter.

Quit_Enable_Event_Number_Confirmation **$00DA_{hex}$**

Bedeutung: Das Kommando *Enable_Event_Number_Request* ($002B_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00DA_{hex}$). Es folgen keine weiteren Parameter.

Quit_Disable_Event_Number_Confirmation **$00DB_{hex}$**

Bedeutung: Das Kommando *Disable_Event_Number_Request* ($002C_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00DB_{hex}$). Es folgen keine weiteren Parameter.

Quit_Enable_All_Events_Confirmation **$00DC_{hex}$**

Bedeutung: Das Kommando *Enable_All_Events_Request* ($002D_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00DC_{hex}$). Es folgen keine weiteren Parameter.

Quit_Disable_All_Events_Confirmation **$00DD_{hex}$**

Bedeutung: Das Kommando *Disable_All_Events_Request* ($002E_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00DD_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Quit_Set_Event_Message_Type_Confirmation **$00DE_{hex}$**

Bedeutung: Das Kommando *Set_Event_Message_Type_Request* ($004D_{hex}$) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00DE_{hex}$). Es folgen keine weiteren Parameter.

Set_Event_Message_Type_Failed_Confirmation **$00DF_{hex}$**

Bedeutung: Das Kommando *Set_Event_Message_Type_Request* ($004D_{hex}$) konnte nicht ausgeführt werden.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00DF_{hex}$). Es folgen keine weiteren Parameter.

Quit_Enable_Event_Logical_Address_Confirmation **$00E0_{hex}$**

Bedeutung: Das Kommando *Enable_Event_Logical_Address_Request* (0036_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00E0_{hex}$). Es folgen keine weiteren Parameter.

Quit_Disable_Event_Logical_Address_Confirmation **$00E1_{hex}$**

Bedeutung: Das Kommando *Disable_Event_Logical_Address_Request* (0037_{hex}) wurde ausgeführt.

Syntax: Die Meldung besteht nur aus einem Wort, dem Meldungs-Code ($00E1_{hex}$). Es folgen keine weiteren Parameter.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Event_Number_Indication 8099_{hex}

Bedeutung: Diese Meldung zeigt Ihnen durch bitweise ODER-Verknüpfung der Event-Kennungen alle Events in einem Wort an, die während eines Datenzyklus aufgetreten sind.

Voraussetzung: Sie dürfen für die 16 möglichen Event-Definitionen nur Event-Kennungen verwenden, die aus einem Wort bestehen, in dem nur ein Bit gesetzt ist:

0000 0000 0000 0001$_{bin}$ (0001$_{hex}$)
0000 0000 0000 0010$_{bin}$ (0002$_{hex}$)
0000 0000 0000 0100$_{bin}$ (0004$_{hex}$)
0000 0000 0000 1000$_{bin}$ (0008$_{hex}$)
0000 0000 0001 0000$_{bin}$ (0010$_{hex}$)
0000 0000 0010 0000$_{bin}$ (0020$_{hex}$)
0000 0000 0100 0000$_{bin}$ (0040$_{hex}$)
0000 0000 1000 0000$_{bin}$ (0080$_{hex}$)
0000 0001 0000 0000$_{bin}$ (0100$_{hex}$)
0000 0010 0000 0000$_{bin}$ (0200$_{hex}$)
0000 0100 0000 0000$_{bin}$ (0400$_{hex}$)
0000 1000 0000 0000$_{bin}$ (0800$_{hex}$)
0001 0000 0000 0000$_{bin}$ (1000$_{hex}$)
0010 0000 0000 0000$_{bin}$ (2000$_{hex}$)
0100 0000 0000 0000$_{bin}$ (4000$_{hex}$)
1000 0000 0000 0000$_{bin}$ (8000$_{hex}$)

Syntax:

Wort 1	Code
Wort 2	Parameterzähler
Wort 3	All Events

Bit	15 0

Legende:

Code:	Meldungs-Code (hier 8099_{hex})
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 1)
All Events:	Kennungen aller in einem Datenzyklus aufgetretenen Events (in Bitkodierung). Dieses Wort wird durch bitweise ODER-Verknüpfung der Event-Kennungen aller Aufgetretenen Events generiert.

Falls die zum Event-Zeitpunkt zugehörigen Daten für eine weitergehende Behandlung des/der Event(s) vom Steuerungs- oder Rechnersystem benötigt werden, müssen Sie eine Event-Meldung auswerten, die das Datenwort mitliefert (Event_Data_Indication $809B_{hex}$). Sonst ist die zeitliche Zugehörigkeit der Daten zum Event nicht gewährleistet.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Event_Data_Indication **$809B_{hex}$**

Bedeutung: Diese Meldung zeigt Ihnen alle Events an, die während eines Datenzyklus aufgetreten sind. Als Parameter werden für jeden Event die Event-Kennung und das Datenwort des Eingabe-Moduls, bei dem der Event ausgelöst worden ist, übergeben. Die Event-Kennung muß vorher definiert worden sein.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler (n)	
Wort 3	Event-Kennung	für 1. Event
Wort 4	Datenwort	
	...	
	Event-Kennung	für x. Event
Wort n+2	Datenwort	

Bit	15 0

Legende:

Code:	Meldungs-Code (hier $809B_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der Events multipliziert mit 2)
Event-Kennung:	Kennungen der in einem Datenzyklus aufgetretenen Events (in Bitkodierung).
Datenwort:	Datenwort des Eingabe-Moduls, bei dem der in der vorherigen Zeile angegebene Event ausgelöst worden ist.

Quit_Read_Event_Counter_Confirmation **$811D_{hex}$**

Bedeutung: Diese Meldung ist die positive oder die negative Quittung auf das Kommando *Read_Event_Counter_Request* (0114_{hex}).

Positive Quittung: Die Meldung übermittelt als positive Quittung die Zählerstände der angeforderten Event-Counter. Bei der positiven Quittung ist der Parameter *Result* gleich 00_{hex}.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler (n)	
Wort 3	Result	ID-Code
Wort 4	Event-Counter	
Wort 4	Event-Counter	
	...	
Wort n+2	Event-Counter	

Bit	15 8	7 0

Legende:

Code:	Meldungs-Code (hier $80F4_{hex}$)

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Parameterzähler:	Anzahl der nachfolgenden Worte (hier Anzahl der ausgelesenen Event-Counter, maximal 16_{dez}).
Event-Counter:	Zählerstände der ausgelesenen Event-Counter. Die Ausgabereihenfolge entspricht der bei der Anforderung (Kommando *Read_Event_Counter_Request* (0114_{hex})) vorgegebenen Reihenfolge. Falls Sie dort eine 0 eingetragen hatten um alle Zählerstände auszulesen, entspricht die Ausgabereihenfolge der bei der Event-Definition (Kommando *Receive_Events_Request* ($0002A_{hex}$)) vorgegebenen Reihenfolge.

Negative Quittung: Die Meldung übermittelt als negative Quittung keine Zählerstände sondern Fehler-Codes. Bei der negativen Quittung ist der Parameter *Result* gleich FF_{hex}.

Syntax:

Wort 1	Code	
Wort 2	Parameterzähler (n)	
Wort 3	Result	Error-Class
Wort 4	Error-Code	Additional-Code
Wort 5	Additional-Code	00_{hex}

Bit	15 8	7 0

Legende:

Code:	Meldungs-Code (hier $811D_{hex}$)
Parameterzähler:	Anzahl der nachfolgenden Worte (hier 3).
Result:	Für die negative Quittung FF_{hex}.
Error-Class:	reserviert (00_{hex}), zur Zeit ohne Bedeutung
Error-Code:	reserviert (00_{hex}), zur Zeit ohne Bedeutung
Additional-Code:	Event-Kennung, bei der ein Fehler erkannt wurde.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Anhang A

Technischer Anhang

A Technischer Anhang A-3

A.1 Technische Daten der Anschaltbaugruppen A-3

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

A Technischer Anhang

A.1 Technische Daten der Anschaltbaugruppen

Allgemeine Daten

zulässiger Temperaturbereich	
- Betrieb (Zulufttemperatur):	0^{o}C bis $+55^{o}$C
- Lagerung:	-20^{o}C bis $+70^{o}$C
zulässige Feuchte	
- Betrieb:	75% (keine Betauung)
- Lagerung:	95% (keine Betauung)
Isolationsfestigkeit:	0,5 kV

Mechanischer Aufbau

Format:	lange Standard-PC-AT-Einsteckkarte
Frontplatte:	PC-Kartenhalter
Abmessungen:	338,5 mm * 114,3 mm
Masse	
- IBS PC CB/I-T:	ca. 340 g
- IBS PC CB/COP/I-T:	ca. 500 g
- IBS PC CB/RTX486/I-T:	ca. 500 g

Hostinterface

AT-Bus-Schnittstelle	
- Steckverbinder:	62- und 36- poliger Direkt-steckverbinder
Adreßbedarf je Anschaltbaugruppe (siehe Kapitel 4)	
- im Speicher des Hosts:	4 KByte
- im I/O-Bereich:	8 Byte
- Interrupt:	einer
zulässige Anzahl von Anschaltbaugruppen:	maximal vier pro Host

Versorgung

zulässige Spannung (Welligkeit eingeschlossen):	+5 V DC ± 5 %
zulässige Welligkeit:	100 mV_{SS}
Stromaufnahme	
- IBS PC CB/I-T:	typisch 1,2 A
- IBS PC CB/COP/I-T:	typisch 1,5 A
- IBS PC CB/RTX486/I-T:	typisch 1,5 A
Batterie:	6 Volt,1350 mAh z.B.: Varta 2/CR 2/3 AA

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

InterBus-S

IBS-Masterboard:	MA5-Board
Fernbusanschluß	
- Schnittstelle:	2-Leiter Fernbus
- Steckverbinder:	SUB-D9, Buchsenleiste
- Übertragungsgeschwindigkeit:	500 kBits/s
- max. Anzahl der IBS-Teilnehmer:	320
- max. Anzahl der Fernbusteilnehmer:	256
- max. Anzahl der Bussegmente:	256
- max. Anzahl der Prozeßdatenworte:	256
- max. Anzahl der PCP-Teilnehmer:	256
Diagnose- und Parametrierungsschnittstelle	
- Protokoll:	kompatibel zu RS-232C
- Steckverbinder:	SUB-D9, Stftleiste

Coprozessorboard *COP 386* (IBS PC CB/COP/I-T)

Betriebssystem:	TDOS
Abmessungen:	78,74 x 113,03 mm
Prozessor:	80386 SX
Taktfrequenz:	25 MHz
Chipset:	Chips & Technologies 82C836
Arbeitsspeicher:	2 MByte DRAM 128 KByte CMOS-RAM (batteriegepuffert)
EPROM 1:	128 KByte
EPROM 2:	nicht bestückt
Stromversorgung:	+5 V ±5%; typ. 380 mA
Serielle Schnittstelle:	COM1, 9-polig
Schnittstellenbaustein:	UART NS 16C450

Coprozessorboard *COP 486* (IBS PC CB/RTX486/I-T)

Betriebssystem:	RTXDOS
Abmessungen:	78,74 x 113,03 mm
Prozessor:	486 SXLC-40, double clock, 8 KByte Cache
Taktfrequenz:	40 MHz intern 20 MHz extern
Chipset:	Chips & Technologies 82C836
Arbeitsspeicher:	2 MByte DRAM, 128 KByte CMOS-RAM (batteriegepuffert)
Flash-EPROM 1:	256 KByte
Flash-EPROM 2:	256 KByte
Stromversorgung:	+5 V ±5%; typ. 380 mA
Serielle Schnittstelle:	COM1 mit FIFO, 9polig
Schnittstellenbaustein:	ST16C550JW-44

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Anhang B

Verzeichnisanhang

B Verzeichnisanhang B-3

B.1 Abbildungsverzeichnis B-3
B.2 Tabellenverzeichnis B-6
B.3 Stichwortverzeichnnis. B-9

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

B Verzeichnisanhang

B.1 Abbildungsverzeichnis

Kapitel 1

Modularer Aufbau der Anschaltbaugruppe 1-10

Kapitel 2

Aufbau der Anschaltbaugruppen der Baureihe IBS PC CB/.../I-T . . . 2-4
Elemente auf dem PC-Kartenhalter 2-6
Diagnoseschnittstelle und Diagnosekabel zur Verbindung mit einem PC 2-10
Fernbusschnittstelle und Beispiel für ein Fernbuskabel (Kabeltyp D9/D9) 2-11
Fachgerechtes Anschließen eines Fernbussteckers (Sub-D9) 2-12

Kapitel 3

Blockschaltbild des Coprozessorboards 3-5
Pinanordnung der in der Steckerwanne des COPs 3-7
Entwicklungskabel für den COP386 3-8
SRAM-Aufteilung auf dem COP386 der IBS PC CB/COP/I-T 3-11
Das MPM als zentrale Schnittstelle 3-13

Kapitel 4

Schalter zur Einstellung der IO-Adresse 4-3
Schalter zur Einstellung der Boardnummer (Board No.) 4-4
Schalter für die Kontrolle des IBS-Masterboards (IBS-Control) 4-6
Schalter für das Anlaufverhalten (IBS-Autostart) 4-7
Schalter für den Programmstart aus dem EPROM (EPROM-Start) . . 4-8
Schalter für den die Auswahl des Boot-Laufwerks (Boot Disk) 4-8
Schalter zur Einstellung des Bootverhaltens in Bezug auf RFSERVER . 4-10
Schalter zur Einstellung des Boot-Verhaltens in Bezug auf DPCON . . 4-11
Schalter zur Einstellung des Datenübertragungswegs (Terminal-Mode) 4-12
Steckbrücken zur Parametrierung der Anschaltbaugruppen 4-13

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Position der Stiftleiste für das Batterie-Pack 4-15
Beispieleinträge in der Datei IBSPCCB.INI 4-21
Hauptmenü des Monitorprogramms 4-24
Das Pull-Down-Menü Functions des Monitorprogramms 4-25
Monitor-Maske des Monitorprogramms 4-26
Eine Fehlermeldung im Monitorprogramm 4-27
Das Info-Fenster des Monitorprogramms 4-28
Maske für Kommandos und Meldungen 4-29
Das Pull-Down-Menü Options 4-30
Das Pull-Down-Menü General 4-30

Kapitel 5

Das MPM als zentrale Schnittstelle einer Anschaltbaugruppe 5-3
Verschieben des MPM-Fensters durch den Device-Driver 5-4
Aufteilung des MPMs 5-5
Relative Adressen von Prozeßdatenworten im DTI-Übertragungsbereich 5-6
DTI-Übertragungsbereiche und Offset-Konstanten 5-7
Struktur der Treibersoftware 5-7
Einsatz von vier Anschaltbaugruppen in einem Host 5-8
Kontrolle von vier Device-Drivern durch das Device-Driver-Interface . . 5-8
Struktur der Treibersoftware auf dem Coprozessorboard (COP) . . . 5-9
Das DDI als Interface zu einem Device-Driver 5-10
Aufteilung des SRAMs 5-13
Das Signal SysFail im MPM 5-16

Kapitel 6

Prinzipieller Aufbau einer Teilnehmerliste 6-3
Busaufbau für die Beispiele zur Adressierung 6-5
Eingangsadressen im Speicherabbild (IN-Buffer) 6-8
Eingangsadressen bei physikalischer Adressierung 6-9
Ausgangsadressen im Speicherabbild (OUT-Buffer) 6-10

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Ausgangsadressen bei physikalischer Adressierung 6-11
Aufbau des Kommandos Check_Physical_Configuration_Request . . 6-14
Aufbau des Kommandos Receive_Local_Bus_Code_Map_Request . . 6-16
Numerierung der Bussegmente bei logischer Adressierung 6-17
Aufbau des Kommandos Receive_Logical_IN_Address_Map_Request . 6-20
Eingangsadressen bei logischer Adressierung 6-22
Eingangsadressen im Speicherabbild (IN-Buffer) 6-23
Aufbau des Kommandos Receive_Logical_OUT_Address_Map_Request 6-24
Ausgangsadressen bei logischer Adressierung 6-26
Ausgangsadressen im Speicherabbild (Out-Buffer) 6-27
Aufbau des Kommandos Receive_Group_Numbers_Request 6-29
Einteilung in logische Funktionsgruppen 6-30
Aufbau des Kommandos Switch_Group_Off_Request 6-32
Aufbau des Kommandos Switch_Group_On_Request 6-33
Aufbau des Kommandos Define_Group_Error_Characteristic_Request 6-34

Kapitel 7

Diagnose-Anzeigen auf dem PC-Kartenhalter 7-3
LED-Diagnose an Busklemmen 7-4
LED-Diagnose an E/A-Modulen 7-5

Kapitel 8

Schnelle Reaktion über die Prozeßdatenverknüpfung 8-28
Impulsdiagramm zur Prozeßdatenverknüpfung 8-29

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

B.2 Tabellenverzeichnis

Kapitel 1

Bestelldaten für Dokumentation 1-7

Kapitel 2

Belegung des kurzen AT-Bus-Direktsteckverbinders 2-8

Belegung des langen AT-Bus-Direktsteckverbinders 2-9

Anschlußbelegung der Diagnoseschnittstelle 2-11

Anschlußbelegung der Fernbusschnittstelle 2-12

Funktion der Steckbrücke für den Reset-Taster 2-15

Kapitel 3

Übersicht Coprozessorboards 3-3

Laufwerke des COP 486 3-4

Belegung des Adapterkabels 3-7

I/O-Adreßbereich des Coprozessorboards 3-9

Interruptbelegung des Coprozessorboards 3-10

Aufteilung des DRAMs 3-12

Kapitel 4

Mögliche Einstellungen für die Basisadresse im IO-Bereich 4-4

Einstellung der Boardnummer von Anschaltbaugruppen 4-5

Kontrolle des IBS-Masterboards vom Host oder COP 4-6

Festlegung des Anlaufverhaltens 4-7

Programmstart aus Flash-EPROM 4-8

Auswahl des Boot-Laufwerks für das Coprozessorboard 4-9

Bootverhalten des Coprozessorboards in Bezug auf RFSERVER . . . 4-10

Bootverhalten in Bezug auf DPCON 4-11

Einstellung des Terminal-Modes 4-12

Funktion der Steckbrücke zur Auswahl der Spannungsversorgung (siehe Bedruckung auf der Folie der Anschaltbaugruppe) 4-13

Funktion der Steckbrücke zur Trennung vom Hardware-Reset des Hosts

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

(siehe Bedruckung auf der Folie der Anschaltbaugruppe) 4-14

Funktion der Steckbrücke zum Deaktivieren der Reset-Taster (siehe Bedruckung auf der Folie der Anschaltbaugruppe) 4-14

Anschlußbelegung der Stiftleiste für das Batterie-Pack 4-15

Stromaufname der Anschaltbaugruppen 4-16

Typisches Memory Mapping eines Standard-PCs 4-17

Beipiele für die Übergabe von Parametern beim Aufruf des Device-Drivers 4-19

Beipiele für das Eintragen von Parametern 4-22

Kapitel 5

Offset-Konstanten 5-6

Öffnen eines Datenkanals vom Host zum IBS-Masterboard 5-11

Öffnen eines Datenkanals vom Host zum Coprozessorboard (für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T) 5-11

Öffnen eines Datenkanals vom Coprozessorboard zum Host (für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T) 5-11

Öffnen eines Datenkanals vom Coprozessorboard zum IBS-Masterboard (für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T) 5-11

Kapitel 6

IBS-Teilnehmer des Beispielaufbaus 6-4

Zuordnung der Eingangsworte zu den IBS-Teilnehmern 6-8

Zuordnung der Ausgangsworte zu den IBS-Teilnehmern 6-10

Kommandofolge für den Anlauf bei physikalischer Adressierung . . . 6-12

Unterschiedliche Numerierung von Worten bei Speicherprogrammierbaren Steuerungen (SPS) und PCs 6-19

Kommandofolge für den Anlauf bei logischer Adressierung 6-28

Kapitel 7

Fehlerarten unter Firmware-Version 3.72 7-9

Kapitel 8

Kommandos für das IBS-Masterboard 8-3

Befehle für die Prozeßdatenverknüpfung 8-29

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel 9

Meldungen des IBS-Masterboards 9-3

Bedeutung des Parameters Fehlernummer 9-24

Bedeutung des Parameters Fehlernummer 9-33

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

B.3 Stichwortverzeichnnis

A

Adapterkabel für COP 3-6
Adressierungsarten 6-6
Adreßlisten überprüfen 8-16
AT-Bus-Direktsteckverbinder 2-8
AT-Bus-Interface 2-13

B

Batterie-Pack 2-5, 4-15
Boardnummer 4-4, 4-17
Boot Disk 4-8
Bus starten 8-16
Buskonfiguration einlesen 8-6
Buskonfiguration überprüfen 6-14, 8-8
Bussegment 6-3
Bussegmente numerieren 8-13
Bussegmentnummer vergeben 6-16

C

Controller- Fehlernummer 7-10
COP 3-3
Coprozessorboard 2-4, 3-3

D

Data-Interface (DTI) 5-12
Datenkonsistenz 2-14, 5-12, 6-20, 6-24
Device-Driver 5-8
Device-Driver, Installation 4-16, 4-18
Device-Driver-Interface 5-8
Device-Name 5-10, 5-13
Diagnose 7-3
Diagnose an Busklemmen 7-4
Diagnose an E/A-Modulen 7-5
Diagnose-Anzeigen 7-3
Diagnosebitregister 5-12
Diagnosefunktion 5-12, 7-8
Diagnosekabel 2-10
Diagnoseparameterregister 5-12
Diagnoseschnittstelle 2-4, 2-10
Dokumentation 1-6
Download 5-16
DPCON 4-9, 4-11
DTI-Adresse 5-6
DTI-Übertragungsbereich 5-7
Dynamic-Link-Libary für Windows 4-20
dynamisches RAM des COPs 3-12

E

Echtzeituhr des COPs 3-15
EMM386.EXE 4-19
Enable/Disable RESET Button 4-14
Entwicklungskabel für COP 3-7
EPROM des COPs 3-10
EPROM-Start 4-8
Erdungsbolzen 2-4, 2-6
Event-Definition 8-38
externen Versorgungsspannung 4-13

F

Fehlerart 7-9, 9-22
Fernbuskabel 2-11
Fernbusschnittstelle 2-4, 2-11
Fernbussteckers 2-12
Firmware-Version bestimmen 8-27
Frontblende 2-6

G

Grundboard 2-4
Gruppe abschalten 6-32, 8-10
Gruppe einschalten 6-33, 8-11
Gruppen bilden 6-29, 8-9
Gruppendefinition 6-29
Gruppenverhalten im Fehlerfall 6-33, 8-12

H

Hardware-Interrupt 4-17

I

I/O-Adreßbereich des COPs 3-9
IBS CMD SWT 7-7
IBS SYS SWT 7-6
IBS-Autostart 4-7
IBS-Control 4-6
IBS-Masterboard 2-4
ID-Code 6-3
Inbetriebnahme 4-24
Interrupt (OS/2) 4-23
Interruptbelegung des COPs 3-10
IO-Address 4-3, 4-16

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

K

Kabeltypen 2-12
Kartennummer 4-4
Kommandos 8-3
Kommunikation 1-6

L

Längen-Code 6-3
LED 2-7
logische Adressierung 6-6, 6-13
logische Ausgangsadressen 6-24
logische Eingangsadressen 6-20

M

Mailbox-Interface (MXI) 5-12
Mapping-Fenster 3-12
Mapping-Register 3-12
Meldungen 9-3
Memory-Mapper 3-14
Microsoft-Windows 4-20
MONI.BAT 1-1
Monitorprogramm 4-24, 7-6
MPM, Adreß-Offset 5-5
MPM, Basisadresse 4-17
MPM, Memory- Fenster 5-3
MPM, Offset-Konstanten 5-6
MPM, Relative Adresse 5-5
MPM, Teilnehmerbereich 5-4
MPM-Offset-Konstanten 5-6
MPM-Teilnehmer 5-3
MPM-Teilnehmerbereich 5-4
MS-DOS 4-18
Multi-Port-Memory 5-3

N

Node 5-10
Node-Handle 1-2, 5-10

P

Parameterzähler 5-14
PC HW RESET 4-14
PCCBMONI 4-24
PC-Kartenhalter 2-6
physikalische Adressierung 6-6, 6-7
Position im Peripheriebus 8-21, 8-22, 9-30
Power Supply 4-13
Prozeßdatenverknüpfung 8-28

R

Remote-Debugging 3-14
Resetsystem des COPs 3-14
Reset-Taster 2-5, 2-15, 4-14
RFSERVER 4-10
RFSERVER.EXE 3-4

S

serielle Schnittstelle des COPs 3-6, 4-16
SHADOW RAM 4-20
Software-Tools 4-23, 7-6
Spannungsüberwachung des COPs 3-14
Spannungsüberwachung, AT-Bus 2-15
statisches RAM des COPs 3-11, 5-13
STOP-Zustand 8-17, 8-18, 9-14
SysFail 5-16

T

TDOS-PRO 5-16
Terminal-Mode 4-12
Treibersoftware 5-7
TSR-Programm 4-18

U

Urkonfiguration 8-6

V

VIDEO CACHE 4-20

W

Watchdog 5-14
Watchdog des COPs 3-15
Watchdogs 5-14

Z

Zugriffsverwaltung des MPM 2-14
Zuordnungslisten 6-13

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel 1

Modulerer Aufbau der Anschaltbaugruppe 1-10

Kapitel 2

Aufbau der Anschaltbaugruppen der Baureihe IBS PC CB/.../I-T . . . 2-4
Elemente auf dem PC-Kartenhalter 2-6
Diagnoseschnittstelle und Diagnosekabel zur Verbindung mit einem PC 2-10
Fernbusschnittstelle und Beispiel für ein Fernbuskabel (Kabeltyp D9/D9) 2-11
Fachgerechtes Anschließen eines Fernbussteckers (Sub-D9) 2-12

Kapitel 3

Blockschaltbild des Coprozessorboards 3-5
Pinanordnung der in der Steckerwanne des COPs 3-7
Entwicklungskabel für den COP386 3-8
SRAM-Aufteilung auf dem COP386 der IBS PC CB/COP/I-T 3-11
Das MPM als zentrale Schnittstelle 3-13

Kapitel 4

Schalter zur Einstellung der IO-Adresse 4-3
Schalter zur Einstellung der Boardnummer (Board No.) 4-4
Schalter für die Kontrolle des IBS-Masterboards (IBS-Control) 4-6
Schalter für das Anlaufverhalten (IBS-Autostart) 4-7
Schalter für den Programmstart aus dem EPROM (EPROM-Start) . . 4-8
Schalter für den die Auswahl des Boot-Laufwerks (Boot Disk) 4-8
Schalter zur Einstellung des Bootverhaltens in Bezug auf RFSERVER . 4-10
Schalter zur Einstellung des Boot-Verhaltens in Bezug auf DPCON . . 4-11
Schalter zur Einstellung des Datenübertragungswegs (Terminal-Mode) 4-12
Steckbrücken zur Parametrierung der Anschaltbaugruppen 4-13
Position der Stiftleiste für das Batterie-Pack 4-15
Beispieleinträge in der Datei IBSPCCB.INI 4-21
Hauptmenü des Monitorprogramms 4-24
Das Pull-Down-Menü Functions des Monitorprogramms 4-25

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Monitor-Maske des Monitorprogramms 4-26
Eine Fehlermeldung im Monitorprogramm 4-27
Das Info-Fenster des Monitorprogramms 4-28
Maske für Kommandos und Meldungen 4-29
Das Pull-Down-Menü Options 4-30
Das Pull-Down-Menü General 4-30

Kapitel 5

Das MPM als zentrale Schnittstelle einer Anschaltbaugruppe 5-3
Verschieben des MPM-Fensters durch den Device-Driver 5-4
Aufteilung des MPMs 5-5
Relative Adressen von Prozeßdatenworten im DTI-Übertragungsbereich 5-6
DTI-Übertragungsbereiche und Offset-Konstanten 5-7
Struktur der Treibersoftware 5-7
Einsatz von vier Anschaltbaugruppen in einem Host 5-8
Kontrolle von vier Device-Drivern durch das Device-Driver-Interface . . 5-8
Struktur der Treibersoftware auf dem Coprozessorboard (COP) . . . 5-9
Das DDI als Interface zu einem Device-Driver 5-10
Aufteilung des SRAMs 5-13
Das Signal SysFail im MPM 5-16

Kapitel 6

Prinzipieller Aufbau einer Teilnehmerliste 6-3
Busaufbau für die Beispiele zur Adressierung 6-5
Eingangsadressen im Speicherabbild (IN-Buffer) 6-8
Eingangsadressen bei physikalischer Adressierung 6-9
Ausgangsadressen im Speicherabbild (OUT-Buffer) 6-10
Ausgangsadressen bei physikalischer Adressierung 6-11
Aufbau des Kommandos Check_Physical_Configuration_Request . . 6-14
Aufbau des Kommandos Receive_Local_Bus_Code_Map_Request . . 6-16
Numerierung der Bussegmente bei logischer Adressierung 6-17
Aufbau des Kommandos Receive_Logical_IN_Address_Map_Request . 6-20

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Eingangsadressen bei logischer Adressierung 6-22

Eingangsadressen im Speicherabbild (IN-Buffer) 6-23

Aufbau des Kommandos Receive_Logical_OUT_Address_Map_Request 6-24

Ausgangsadressen bei logischer Adressierung 6-26

Ausgangsadressen im Speicherabbild (Out-Buffer) 6-27

Aufbau des Kommandos Receive_Group_Numbers_Request 6-29

Einteilung in logische Funktionsgruppen 6-30

Aufbau des Kommandos Switch_Group_Off_Request 6-32

Aufbau des Kommandos Switch_Group_On_Request 6-33

Aufbau des Kommandos Define_Group_Error_Characteristic_Request 6-34

Kapitel 7

Diagnose-Anzeigen auf dem PC-Kartenhalter 7-3

LED-Diagnose an Busklemmen 7-4

LED-Diagnose an E/A-Modulen 7-5

Kapitel 8

Schnelle Reaktion über die Prozeßdatenverknüpfung 8-28

Impulsdiagramm zur Prozeßdatenverknüpfung 8-29

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Kapitel 1

Bestelldaten für Dokumentation 1-7

Kapitel 2

Belegung des kurzen AT-Bus-Direktsteckverbinders 2-8

Belegung des langen AT-Bus-Direktsteckverbinders 2-9

Anschlußbelegung der Diagnoseschnittstelle 2-11

Anschlußbelegung der Fernbusschnittstelle 2-12

Funktion der Steckbrücke für den Reset-Taster 2-15

Kapitel 3

Übersicht Coprozessorboards 3-3

Laufwerke des COP 486 3-4

Belegung des Adapterkabels 3-7

I/O-Adreßbereich des Coprozessorboards 3-9

Interruptbelegung des Coprozessorboards 3-10

Aufteilung des DRAMs 3-12

Kapitel 4

Mögliche Einstellungen für die Basisadresse im IO-Bereich 4-4

Einstellung der Boardnummer von Anschaltbaugruppen 4-5

Kontrolle des IBS-Masterboards vom Host oder COP 4-6

Festlegung des Anlaufverhaltens 4-7

Programmstart aus Flash-EPROM 4-8

Auswahl des Boot-Laufwerks für das Coprozessorboard 4-9

Bootverhalten des Coprozessorboards in Bezug auf RFSERVER . . . 4-10

Bootverhalten in Bezug auf DPCON 4-11

Einstellung des Terminal-Modes 4-12

Funktion der Steckbrücke zur Auswahl der Spannungsversorgung (siehe Bedruckung auf der Folie der Anschaltbaugruppe) 4-13

Funktion der Steckbrücke zur Trennung vom Hardware-Reset des Hosts (siehe Bedruckung auf der Folie der Anschaltbaugruppe) 4-14

Funktion der Steckbrücke zum Deaktivieren der Reset-Taster

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

(siehe Bedruckung auf der Folie der Anschaltbaugruppe) 4-14

Anschlußbelegung der Stiftleiste für das Batterie-Pack 4-15

Stromaufname der Anschaltbaugruppen 4-16

Typisches Memory Mapping eines Standard-PCs 4-17

Beipiele für die Übergabe von Parametern beim Aufruf des Device-Drivers4-19

Beipiele für das Eintragen von Parametern 4-22

Kapitel 5

Offset-Konstanten 5-6

Öffnen eines Datenkanals vom Host zum IBS-Masterboard 5-11

Öffnen eines Datenkanals vom Host zum Coprozessorboard
(für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T) 5-11

Öffnen eines Datenkanals vom Coprozessorboard zum Host
(für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T) 5-11

Öffnen eines Datenkanals vom Coprozessorboard zum IBS-Masterboard
(für IBS PC CB/COP/I-T und IBS PC CB/RTX486/I-T) 5-11

Kapitel 6

IBS-Teilnehmer des Beispielaufbaus 6-4

Zuordnung der Eingangsworte zu den IBS-Teilnehmern 6-8

Zuordnung der Ausgangsworte zu den IBS-Teilnehmern 6-10

Kommandofolge für den Anlauf bei physikalischer Adressierung . . . 6-12

Unterschiedliche Numerierung von Worten bei
Speicherprogrammierbaren Steuerungen (SPS) und PCs 6-19

Kommandofolge für den Anlauf bei logischer Adressierung 6-28

Kapitel 7

Fehlerarten unter Firmware-Version 3.72 7-9

Kapitel 8

Kommandos für das IBS-Masterboard 8-3

Befehle für die Prozeßdatenverknüpfung 8-29

Kapitel 9

Meldungen des IBS-Masterboards 9-3

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Bedeutung des Parameters Fehlernummer 9-24

Bedeutung des Parameters Fehlernummer 9-33

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

A

Adapterkabel 3-6
Adapterkabel für COP 3-6
Adressierungsarten 6-6
Adreßlisten überprüfen 8-16
AT-Bus-Direktsteckverbinder 2-8
AT-Bus-Interface 2-13

B

Batterie-Pack 2-5, 4-15
Boardnummer 4-4, 4-17
Boot Disk 4-8
Bus starten 8-16
Buskonfiguration einlesen 8-6
Buskonfiguration überprüfen 6-14, 8-8
Bussegment 6-3
Bussegmente numerieren 8-13
Bussegmentnummer vergeben 6-16

C

Controller- Fehlernummer 7-10
COP 3-3
Coprozessorboard 2-4, 3-3

D

Data-Interface (DTI) 5-12
Datenkonsistenz 2-14, 5-12, 6-20, 6-24
Device-Driver 5-8
Device-Driver, Installation 4-16, 4-18
Device-Driver-Interface 5-8
Device-Name 5-10, 5-13
Diagnose 7-3
Diagnose an Busklemmen 7-4
Diagnose an E/A-Modulen 7-5
Diagnose-Anzeigen 7-3
Diagnosebitregister 5-12
Diagnosefunktion 5-12, 7-8
Diagnosekabel 2-10
Diagnoseparameterregister 5-12
Diagnoseschnittstelle 2-4, 2-10
Dokumentation 1-4
Download 5-16
DPCON 4-9, 4-11
DTI-Adresse 5-6
DTI-Übertragungsbereich 5-7
Dynamic-Link-Libary für Windows 4-20
dynamisches RAM des COPs 3-12

E

Echtzeituhr des COPs 3-15
EMM386.EXE 4-19
Enable/Disable RESET Button 4-14
Entwicklungskabel für COP 3-7
EPROM des COPs 3-10
EPROM-Start 4-8
Erdungsbolzen 2-4, 2-6
Event-Definition 8-38
externen Versorgungsspannung 4-13

F

Fehlerart 7-9, 9-22
Fernbuskabel 2-11
Fernbusschnittstelle 2-4, 2-11
Fernbussteckers 2-12
Firmware-Version bestimmen 8-27
Frontblende 2-6

G

Grundboard 2-4
Gruppe abschalten 6-32, 8-10
Gruppe einschalten 6-33, 8-11
Gruppen bilden 6-29, 8-9
Gruppendefinition 6-29
Gruppenverhalten im Fehlerfall 6-33, 8-12

H

Hardware-Interrupt 4-17

I

I/O-Adreßbereich des COPs 3-9
IBS CMD SWT 7-7
IBS SYS SWT 7-6
IBS-Autostart 4-7
IBS-Control 4-6
IBS-Masterboard 2-4
ID-Code 6-3
Inbetriebnahme 4-24
Interrupt (OS/2) 4-23
Interruptbelegung des COPs 3-10
IO-Address 4-3, 4-16

K

Kabeltypen 2-12
Kartennummer 4-4
Kommandos 8-3

Kommunikation 1-4

L

Längen-Code 6-3
LED 2-7
logische Adressierung 6-6, 6-13
logische Ausgangsadressen 6-24
logische Eingangsadressen 6-20

M

Mailbox-Interface (MXI) 5-12
Mapping-Fenster 3-12
Mapping-Register 3-12
Meldungen 9-3
Memory-Mapper 3-14
Microsoft-Windows 4-20
MONI.BAT 1-1
Monitorprogramm 4-24, 7-6
MPM, Adreß-Offset 5-5
MPM, Basisadresse 4-17
MPM, Memory- Fenster 5-3
MPM, Offset-Konstanten 5-6
MPM, Relative Adresse 5-5
MPM, Teilnehmerbereich 5-4
MPM-Offset-Konstanten 5-6
MPM-Teilnehmer 5-3
MPM-Teilnehmerbereich 5-4
MS-DOS 4-18
Multi-Port-Memory 5-3

N

Node 5-10
Node-Handle 1-2, 5-10

P

Parameterzähler 5-14
PC HW RESET 4-14
PCCBMONI 4-24
PC-Kartenhalter 2-6
physikalische Adressierung 6-6, 6-7
Position im Peripheriebus 8-21, 8-22, 9-30
Power Supply 4-13
Prozeßdatenverknüpfung 8-28

R

Remote-Debugging 3-14
Resetsystem des COPs 3-14
Reset-Taster 2-5, 2-15, 4-14
RFSERVER 4-10
RFSERVER.EXE 3-4

S

serielle Schnittstelle des COPs 3-6, 4-16
SHADOW RAM 4-20
Software-Tools 4-23, 7-6
Spannungsüberwachung des COPs 3-14
Spannungsüberwachung, AT-Bus 2-15
statisches RAM des COPs 3-11, 5-13
STOP-Zustand 8-17, 8-18, 9-14
SysFail 5-16

T

TDOS-PRO 5-16
Terminal-Mode 4-12
Treibersoftware 5-7
TSR-Programm 4-18

U

Urkonfiguration 8-6

V

VIDEO CACHE 4-20

W

Watchdog 5-14
Watchdog des COPs 3-15
Watchdogs 5-14

Z

Zugriffsverwaltung des MPM 2-14
Zuordnungslisten 6-13

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

Ihre Meinung interessiert uns!

Geben Sie uns die Möglichkeit, Ihre Anregungen, Wünsche und Kritikpunkte zum vorliegenden InterBus-Handbuch zu erfahren. Aus diesem Grund beantworten Sie bitte den Fragebogen auf der Rückseite. Jeder noch so kleine Hinweis oder Kommentar wird von uns bearbeitet.

Dieses Antwortschreiben können Sie als FAX-Vorlage, im Fensterbrief oder auch - gefaltet (1. / 2.) und zusammengeklebt (mit Tesafilm) - direkt versenden.

Falls Sie einen Briefumschlag verwenden, kennzeichnen Sie diesen im rechten, oberen Teil der Vorderseite mit dem Vermerk "Entgelt bezahlt Empfänger".

Vielen Dank !

1.

Entgelt
bezahlt
Empfänger

Antwort

Phoenix Contact GmbH & Co. KG
Produktmarketing InterBus/ME-DOK
Flachsmarktstraße 8 - 28
D-32825 Blomberg

2.

onlinecomponents.com

onlinecomponents.com
THE ONLINE DISTRIBUTOR OF ELECTRONIC COMPONENTS

FAX-NR.: *49-(0)5235-331199

FAX-Antwort
Phoenix Contact GmbH & Co. KG
Produktmarketing InterBus/ME-DOK

Datum: .....

Anzahl Seiten: .....

Absender:

Firma: ..... Name: .....

..... Abteilung: .....

Straße: ..... Funktion: .....

Ort: .... Tel.: ..... / .....

Fax: ..... / .....

Angaben zum Handbuch:

Typ: ..... Revision: ..... Art.-Nr.: .....

Meine Meinung zum Handbuch

Gestaltung

	Ja	z.Teil	Nein
Ist das Inhaltsverzeichnis übersichtlich gestaltet?	❒	❒	❒
Sind die gewählten Bilder/Grafiken verständlich/aussagekräftig?	❒	❒	❒
Fördert die Seitengestaltung die schnelle Infosuche?	❒	❒	❒
Ist die Kapitelinhaltsübersicht ausführlich?	❒	❒	❒
Sind die Texterklärungen zu den Bildern ausreichend?	❒	❒	❒
Entspricht die Qualität der Bilder (Übersichtlichkeit/selbsterklärend) Ihren Erwartungen/Ansprüchen?	❒	❒	❒
Führen die Indexeinträge zu den informativen Stellen?	❒	❒	❒

Inhalt

	Ja	z.Teil	Nein
Sind die gewählten Fachbegriffe verständlich/aussagekräftig?	❒	❒	❒
Sind die gewählten Formulierungen verständlich/aussagekräftig?	❒	❒	❒
Sind die gewählten Verzeichniseinträge verständlich/aussagekräftig?	❒	❒	❒
Welches Verzeichnis benutzen Sie am häufigsten, um Informationen zu suchen?			
Fehlen wichtige Informationen? Wenn ja, welche?	❒	❒	❒
Sind die Beispiele praxisgerecht?	❒	❒	❒
Ist das Handbuch gut zu handhaben?	❒	❒	❒
Wünschen Sie mehr Platz für Notizen?	❒	❒	❒

Ich habe folgende Anregung(en)/Anmerkung(en):

.....
.....
.....
.....
.....
.....
.....

DNR 2532

5050CF17

onlinecomponents.com