

Anwenderhandbuch

Ankopplung an INTERBUS MMICOM Schnittstelle 06, 14, 18, 21, 76

Teilenummer: 80 860.658
Version: 3
Datum: 08.12.2009
Gültig für: TSwin .net 4.x

Version	Datum	Änderungen
1	26.07.2005	Erstausgabe
2	24.11.2005	Gültigkeit erweitert, Kapitel „Wichtige Hinweise“ eingefügt, Übersicht Funktionsbausteine korrigiert
3	08.12.2009	Erweiterung um Schnittstelle 76, MPM-Beschreibung FC66 / FC86

Dieses Handbuch ist einschließlich aller darin enthaltenen Abbildungen urheberrechtlich geschützt. Jede Drittverwendung dieses Handbuchs, die von den urheberrechtlichen Bestimmungen abweicht, ist verboten. Die Reproduktion, Übersetzung sowie die elektronische und fotografische Archivierung und Veränderung bedarf der schriftlichen Genehmigung der Firma Süttron electronic GmbH. Zuwiderhandlungen verpflichten zu Schadenersatz.

Süttron electronic behält sich jegliche Änderungen, die dem technischen Fortschritt dienen, vor.

Gesamtinhaltsverzeichnis

1	Wichtige Hinweise	1-1
	1.1 Symbole	1-1
	1.2 Sicherheitshinweise	1-1
	1.3 Bestimmungsgemäßer Gebrauch	1-1
	1.4 Zielgruppe	1-1
2	INTERBUS MMICOM Schnittstelle 06, 14, 18, 21, 76	2-1
	2.1 Integration der Bediengeräte	2-1
	2.2 MMICOM-Profil	2-1
	2.2.1 Direkter Prozessdatenkanal	2-1
	2.2.2 Indirekter Prozessdatenkanal	2-2
	2.2.3 Parameterkanal	2-2
	2.3 Anschaltung des Bediengeräts	2-2
	2.3.1 Spezifikation innerhalb des INTERBUS	2-3
	2.4 Datentypen	2-4
	2.5 Randbedingungen für optimierte Datenübertragung	2-4
	2.6 Projektieren	2-7
	2.6.1 Protokollparameter	2-7
	2.6.1.1 MMICOM Handshake Timeout	2-7
	2.6.1.2 Verzögerung bis Verbindungsaufbau	2-8
	2.6.1.3 Gleitkommazahlen im Siemens-Format	2-8
	2.6.1.4 Optimierte Datenübertragung	2-8
	2.6.2 Pollbereich	2-9
	2.6.3 Zustandsmeldungen	2-9
	2.6.4 Eingabesyntax	2-9
	2.6.5 Physikalische Ankopplung	2-11
	2.6.5.1 Steckerbelegung	2-11
	2.6.5.2 2 - Leiter Fernbuskabel	2-12
	2.6.5.3 Umsetzung von 8 - Leiterprotokoll auf 2 - Leiterprotokoll	2-13
	2.7 Fehlermeldungen	2-14
	2.8 Anwendungen	2-16
	2.8.1 Siemens S7 Steuerung	2-16
	2.8.1.1 Betriebsart Asynchron	2-16
	2.8.1.2 Betriebsart Asynchron mit Synchronisationsimpuls	2-16
	2.8.1.3 Bediengerät als INTERBUS-Teilnehmer konfigurieren	2-17
	2.8.1.4 Funktionsbausteine von Phoenix Contact	2-17
	2.8.1.5 FC20 (INIT_IB)	2-18
	2.8.1.6 FC21 (MEM_READ)	2-22
	2.8.1.7 FC22 (MEM_WRIT)	2-23
	2.8.1.8 FC28 (IB_SYNC)	2-24
	2.8.1.9 Funktionsbausteine von Süttron electronic	2-24
	2.8.1.10 Dekodierung der Variablennummer	2-24
	2.8.1.11 Übersicht über Funktionsbausteine von Süttron electronic	2-26
	2.8.1.12 Quellen und Bausteine einfügen	2-26

Gesamtinhaltsverzeichnis

2.8.1.13	FC60 für Betriebsart Asynchron an S7-400.....	2-28
2.8.1.14	FC66 für Betriebsart Asynchron an S7-300.....	2-29
2.8.1.15	FB80 für Betriebsart Asynchron an S7-400.....	2-30
2.8.1.16	FC86 für Betriebsart Asynchron an S7-300.....	2-31
2.8.1.17	FC65 für Betriebsart Asynchron mit Synchronisationsimpuls an S7-300	2-33
2.8.1.18	FC85 für Betriebsart Asynchron mit Synchronisationsimpuls an S7-300	2-34
2.8.1.19	Beispiel für Betriebsart Asynchron	2-34
2.8.1.20	Beispiel für Betriebsart Asynchron mit Synchronisationsimpuls.....	2-35
A	Index	A-1

1 Wichtige Hinweise

1.1 Symbole

In diesem Handbuch werden Symbole verwendet, um Sie auf Hinweise und Gefahren aufmerksam zu machen.

**Gefahr**

Dieses Symbol wird benutzt, wenn es durch ungenaues Befolgen oder Nichtbefolgen von Anweisungen zu Personenschäden kommen kann.

**Hinweis**

Dieses Symbol kennzeichnet Anwendungsratschläge oder ergänzende Hinweise.

**Verweis auf Informationsquelle**

Dieses Symbol kennzeichnet Verweise auf weiterführende Informationsquellen zu dem aktuellen Thema.

1.2 Sicherheitshinweise

- Lesen Sie dieses Handbuch, bevor Sie die Software in Betrieb nehmen. Bewahren Sie dieses Handbuch an einem, für alle Benutzer jederzeit zugänglichen, Platz auf.
- Das Anwenderhandbuch, insbesondere die Sicherheitshinweise, sind von allen Personen zu beachten, die mit der Software und dem projektierten Bediengerät arbeiten.
- Bitte beachten Sie die für den Einsatzort des Bediengeräts geltenden Regeln und Vorschriften zur Unfallverhütung.
- Die Installation und Bedienung des Bediengeräts darf nur von ausgebildetem und geschultem Personal erfolgen.

1.3 Bestimmungsgemäßer Gebrauch

- Die Software ist ausschließlich für das Projektieren von Bediengeräten zu verwenden. Jede andere Verwendung ist nicht zulässig.

1.4 Zielgruppe

Alle Projektier- und Programmierarbeiten in Verbindung mit dem Automatisierungssystem dürfen nur von geschultem Personal ausgeführt werden (z.B. Elektrofachkräfte, Elektroingenieure).

Das Projektier- und Programmierpersonal muss mit den Sicherheitskonzepten der Automatisierungstechnik vertraut sein.

2 INTERBUS MMICOM Schnittstelle 06, 14, 18, 21, 76

Beachten Sie den Aufdruck auf dem Typenschild des Bediengeräts, um die richtige Schnittstellenvariante in der Projektierungssoftware auswählen zu können.

Die Schnittstellenvariante wird durch die zwei Ziffern hinter dem Schrägstrich gekennzeichnet (im Beispiel fett gedruckt).

Beispiel:

TP32ET/**159032**

2.1 Integration der Bediengeräte

Der INTERBUS ist ein offenes Feldbuskonzept, der von einer Vielzahl von Geräteherstellern des Bereichs Sensorik/Aktorik unterstützt wird. Das bedeutet, dass neben der Anschaltung der Bediengeräte über denselben Bus eine Vielzahl von weiteren Teilnehmern an die Steuerung angekoppelt werden können (dezentrale Ein- und Ausgänge usw.).

Das Bediengerät wird als Slave im 2-Leiter-Fernbus (9-polig D-SUB) integriert. Dabei belegt das Bediengerät auf dem INTERBUS eine Datenbreite von 4 Worten oder 8 Bytes. Die Steuerung bildet jedes Bediengerät mit 8 Byte IN-Daten und 8-Byte OUT-Daten im Speicherabbild ab.

Falls auf der Anschaltbaugruppe die Betriebsart „Physikalische Adressierung“ gewählt ist, werden die Ein- und Ausgangsbereiche der Teilnehmer sequentiell so in den Speicher der Steuerung gelegt, wie sie physikalisch am Bus installiert sind.

Falls auf der Anschaltbaugruppe die Betriebsart „Logische Adressierung“ gewählt ist, kann die Lage der Ein- und Ausgangsbereiche wahlfrei im Speicher der Steuerung platziert werden. Außerdem ist die Lage dieser Bereiche unabhängig von der Reihenfolge der Busteilnehmer.

2.2 MMICOM-Profil

Laut MMICOM-Profil werden auf dem INTERBUS grundsätzlich drei Datenkanäle unterschieden:

1. Direkter Prozessdatenkanal,
2. Indirekter Prozessdatenkanal und
3. Parameterkanal.

2.2.1 Direkter Prozessdatenkanal

Direkte Prozessdaten sind zyklische Daten, die über die Betriebsdauer des Geräts konstant sind. Sie werden unquittiert und zyklisch in den Prozessdatenkanal eingeblendet (z.B. 16 Eingänge eines Eingangsmoduls oder Bitinformationen eines Tastenfeldes - Geräteklasse A1 oder B1).

2.2.2 Indirekter Prozessdatenkanal

Indirekte Prozessdaten sind Daten, die über die Betriebsdauer des Geräts nicht konstant sind und je nach äußeren Ereignissen oder Anforderungen wechseln. Über Prozessdatenkennungen (PD-Index) wird die Struktur dieser Daten festgelegt. Der Handshake zwischen den beiden Kommunikationspartnern wird über ein Status/ Steuerbyte definiert.

2.2.3 Parameterkanal

Über den Parameterkanal kann zusätzlich zum Prozessdatenkanal eine FMS-ähnliche Hintergrundkommunikation betrieben werden. Dabei werden die Bytes eines gesicherten Protokolls sequentiell über den INTERBUS übertragen. Dieses Verfahren ist eher für langsame Kommunikationen mit größeren Datenmengen von Vorteil und verzögert die äquidistante Übertragung der zeitkritischen Daten nicht.

2.3 Anschaltung des Bediengeräts

Das Bediengerät wird am INTERBUS im indirekten Prozessdatenkanal betrieben. Es wird weder der Parameterkanal noch der direkte Prozessdatenkanal verwendet. Die Nutzdaten werden gemäß dem MMICOM Profil interpretiert.

Die Anschaltung an den Bus wird durch galvanisch entkoppelte RS485-Treiber realisiert. Das eigentliche Busprotokoll wird durch den INTERBUS Protokollchip abgewickelt.

Auf der Steuerungsseite ist eine herstellerspezifische INTERBUS Master Anschaltbaugruppe notwendig, die vorzugsweise eine galvanische Trennung besitzen sollte.

Falls die Anschaltbaugruppe in der Steuerung eine 8-Leiter-Fernbusschnittstelle (25-polig) besitzt, müssen Sie mittels der Busklemme IBS 24 BK/LC2 eine Umsetzung auf den 2-Leiter-Fernbus (9-polig) durchführen.

Sowohl die INTERBUS Anschaltung, als auch das MMICOM-Protokoll, sind steuerungsunabhängig. Das hat zur Folge, dass innerhalb des MMICOM Protokolls nur Variablennummern, nicht aber reale Steuerungsadressen, übertragen werden. Die Referenzierung der Variablennummer auf die Variable findet innerhalb der Steuerung statt.

Grundsätzlich muss im Hauptprogramm der Steuerung ein Funktionsbaustein aufgerufen werden. Dieser Funktionsbaustein wird mit der Lage der IN- und OUT-Daten parametrisiert und muss für jedes angeschlossene Bediengerät aufgerufen werden. Bei Bedarf wertet er die vom Bediengerät empfangenen Anforderungen aus. Dabei wird entweder eine Lese- oder eine Schreib-Funktion ausgeführt, die die Zuordnung zwischen der Variablennummer und der realen Steuerungsvariable vornimmt. Die Daten werden in der Steuerung ereignisgesteuert bei Bedarf umkopiert. Dadurch hat die Steuerung auch eine Kontrolle darüber, welche Daten vom Bediengerät geschrieben und gelesen werden.

Sämtliche für den Betrieb des Bediengeräts erforderlichen Dienste (Anforderungen) gehen vom Bediengerät aus. Das Bediengerät besitzt Client-Funktionalität. Die Steuerung reagiert lediglich auf die Anforderungen des Bediengeräts und hat somit Serverfunktion.

2.3.1 Spezifikation innerhalb des INTERBUS

Die Bediengeräte werden am INTERBUS mit folgenden Parametern betrieben:

Tabelle 2-1 Parameter der Bediengeräte im INTERBUS

Parameter		Wert
Breite des Datenkanals	indirekte Prozessdaten mit Statuswort	4 Worte / 8 Byte
	direkte Prozessdaten	keine
	Parameterkanal	keiner
Identcode	PD-Kanal mit Ein- und Ausgangsdaten	2F _h / 47 _d
Funktionsgruppenspezifikation innerhalb des MMI-COM	B3	Variableneingabe
	G1	Variablenanforderung
Verwendete Dienste des MMICOM	PD-Index 0x14	Variable 1 Byte schreiben (mandatorisch)
	PD-Index 0x15	Variable 2 Byte schreiben (mandatorisch)
	PD-Index 0x16	Variable 4 Byte schreiben
	PD-Index 0x40	Variable 1 Byte lesen (mandatorisch)
	PD-Index 0x41	Variable 2 Byte lesen (mandatorisch)
	PD-Index 0x42	Variable 4 Byte lesen
	PD-Index 0x84	Bit auf Byte schreiben
	PD-Index 0x85	Bit auf Wort schreiben

2.4 Datentypen

Die Datentypen definieren, wieviele Bytes einer Einzelvariablen zugeordnet werden.

Tabelle 2-2 Datentypen, MMICOM

Daten- typ	Zugriff auf	Länge Ein- zelvariable	Bemerkung
BY	Byte (und auch Bit)	1 Byte	nächstes Byte steht 1 Adresse höher
W	Wort	2 Bytes	nächstes Wort steht 1 Adresse höher
W-2	Wort	2 Bytes	nächstes Wort steht 2 Adressen höher
DW	Doppelwort	4 Bytes	nächstes Doppelwort steht 1 Adresse höher
DW-2	Doppelwort	4 Bytes	nächstes Doppelwort steht 2 Adressen höher
DW-4	Doppelwort	4 Bytes	nächstes Doppelwort steht 4 Adressen höher

Der Zugriffsparameter bestimmt zum einen die Datenlänge, die vom Bediengerät ausgewertet wird und zum anderen mit welchem MMICOM-Dienst die Variable abgehandelt wird.

Beispiel:

Ein Text mit 10 Zeichen beginnend bei der Variable 100 (Zugriff W - Wort) setzt sich aus den Variablen 100 bis 104 zusammen.



Für Daten, die eine Datenlänge von mehr als 4 Bytes besitzen (alphanumerische Texte, Tabellen, Pollbereich, Zustandsmeldungen usw.) gilt folgende Konvention: Die Daten werden je nach Zugriffsart mit dem entsprechenden 1, 2 oder 4 Byte Zugriff bearbeitet. Falls weitere Daten bearbeitet werden müssen, wird davon ausgegangen, dass diese weiteren Daten unter der jeweils nächst höheren Variablennummer zu finden sind.

2.5 Randbedingungen für optimierte Datenübertragung



Die Verwendung von optimierter Datenübertragung ist nicht zulässig für Projekte mit PC WORX.

Um die Datenübertragung zu optimieren und damit gleichzeitig den Bildaufbau am Bediengerät zu beschleunigen, können Sie in den Protokollparametern die **optimierte Datenübertragung** aktivieren. Dabei wird die zur Verfügung stehende Nutzdatenbreite von 4 Byte (32 Bit) effizienter genutzt und die Anzahl der Kommunikationstelegramme verringert. Um dies zu ermöglichen, müssen Sie die Variablenadressen fortlaufend als Bit-, Byte- oder Wort-Zugriff im jeweiligen Speicherbereich angeben.

Die folgende Tabelle zeigt die möglichen Kombinationen.

Tabelle 2-3 Kombinationen für optimierte Datenübertragung

DB X Offset Y	DB X Offset Y+1	DB X Offset Y+2	DB X Offset Y+3	Anzahl der Telegramme	
				Ohne Optim.	Mit Optim.
BY				1	1
BY	BY			2	1
BY		BY		2	2
BY			BY	2	1
BY	BY	BY		3	2
BY	BY		BY	3	1
BY		BY	BY	3	1
BY	BY	BY	BY	4	1
W low	W high			1	1
W low	W high	W low	W high	2	1



Die Syntax für INTERBUS MMICOM ist steuerungsneutral.

Beispiele für Siemens S7-Steuerungen:

BY 100, 10 = DB100.DBB10,

W 100, 10 = DB100.DBW10,

DW 100, 10 = DB100.DBD10.

Beispiele für PC WORX:

BY100 = Bytearray, Byte 100,

W100 = Wortarray, Wort 100,

DW100 = Doppelwortarray, Doppelwort 100

BIT-Zugriff:

Ohne optimierte Datenübertragung werden jeweils bis zu 8/16/32 Bits, die dieselbe Byte-, Wort- oder Doppelwortadresse haben zusammengefasst übertragen.

Mit optimierter Datenübertragung werden auch Bits in hintereinander adressierten Bytes, bzw. Worten zusammengefasst übertragen.

Tabelle 2-4 BIT-Zugriff

Bitvariablen in einem Bild				Anzahl der Bits	Anzahl der Telegramme	
Variable	Adresse	Low-Bit Nr.	High-Bit Nr.		Ohne Optim.	Mit Optim.
1 : 8	BY 100, 10 : BY 100, 10	0 : 7	0 : 7	8	1	1
1 : 8 9 : 16	BY 100, 10 : BY 100, 10 BY 100, 11 : BY 100, 11	0 : 7 0 : 7	0 : 7 0 : 7	16	2	1
1 : 8 9 : 16 17 : 24 25 : 32	BY 100, 10 : BY 100, 10 BY 100, 11 : BY 100, 11 BY 100, 12 : BY 100, 12 BY 100, 13 : BY 100, 13	0 : 7 0 : 7 0 : 7 0 : 7	0 : 7 0 : 7 0 : 7 0 : 7	32	4	1
1 : 16	W 100, 10 : W 100, 10	0 : 15	0 : 15	16	1	1
1 : 16 17 : 32	W 100, 10 : W 100, 10 W 100, 12 : W 100, 12	0 : 15 0 : 15	0 : 15 0 : 15	32	2	1
1 : 32	DW 100, 10 : DW 100, 10	0 : 31	0 : 31	32	1	1

BYTE-Zugriff:

Tabelle 2-5 BYTE-Zugriff

Bytevariablen in einem Bild	Anzahl der Bytes	Anzahl der Telegramme	
		Ohne Optim.	Mit Optim.
BY 100, 10	1	1	1
BY 100, 10 + BY 100, 11	2	2	1
BY 100, 10 + BY 100, 12	2	2	2
BY 100, 10 + BY 100, 11 + BY 100, 12	3	3	2
BY 100, 10 + BY 100, 11 + BY 100, 13	3	3	1
BY 100, 10 + BY 100, 11 + BY 100, 12 + BY 100, 13	4	4	1



Für 3 aufeinander folgende Bytes werden 2 Zugriffe gemacht. Für die ersten 2 Bytes wird ein WORT-Zugriff gemacht und für das dritte Byte wird ein BYTE-Zugriff gemacht. Falls das dritte Byte am Ende eines Datenbausteins ist, entsteht kein Kommunikationsfehler.

WORT-Zugriff:

Tabelle 2-6 WORT-Zugriff

Wortvariablen in einem Bild	Anzahl der Worte	Anzahl der Telegramme	
		Ohne Optim.	Mit Optim.
W 100, 10	1	1	1
W 100, 10 + W 100, 12	2	2	1

2.6 Projektieren

2.6.1 Protokollparameter

Mit den Protokollparametern passen Sie die Kommunikation an die verwendete Steuerung an.

2.6.1.1 MMICOM Handshake Timeout

Dieser Parameter gibt an, wie lange das Bediengerät auf die Quittung von der Steuerung wartet.

Tabelle 2-7 MMICOM Handshake Timeout

Einstellbare Werte	Standardwert
0 ms bis 65535 ms	100 ms

2.6.1.2 Verzögerung bis Verbindungsaufbau

Dieser Parameter gibt eine Wartezeit an, bis vom Bediengerät die Kommunikation gestartet wird.

Tabelle 2-8 Verzögerung bis Verbindungsaufbau

Einstellbare Werte	Standardwert
0 s bis 20 s	10 s

2.6.1.3 Gleitkommazahlen im Siemens-Format

Dieser Parameter gibt an, ob Gleitkommazahlen im Siemens-Format oder im IEEE-Format ausgetauscht werden.

Tabelle 2-9 Gleitkommazahlen im Siemens-Format

Einstellbare Werte	Standardwert
IEEE-Format	
Siemens-Format	X

2.6.1.4 Optimierte Datenübertragung

Dieser Parameter gibt an, ob eine optimierte Datenübertragung verwendet werden soll. Für die Nutzung der optimierten Datenübertragung müssen Sie bei der Programmierung der Steuerung bestimmte Randbedingungen einhalten!

Tabelle 2-10 Optimierte Datenübertragung

Einstellbare Werte	Standardwert
EIN	
AUS	X



See chapter "Randbedingungen für optimierte Datenübertragung" on page 2-4.

2.6.2 Pollbereich

Über den Pollbereich wird das KBS, der serielle Meldekanal und die LEDs der Funktionstasten verwaltet. Das Bediengerät pollt diesen Bereich ständig. Der Pollbereich hat folgende Struktur:

Tabelle 2-11 Zusatzfunktion Pollbereich, MMICOM

Birnummer	15	8	7	0
Variable X	KBS		Reserviert	
Variable X+1	Serieller Meldekanal High Byte		Serieller Meldekanal Low Byte	
Variable X+2	LED 1 bis 4		LED 5 bis 8	
Variable X+3	LED 9 bis 12		LED 13 bis 16	
Variable X+4	LED 17 bis 20		LED 21 bis 24	
Variable X+5	LED 25 bis 28		LED 29 bis 32	

2.6.3 Zustandsmeldungen

Zustandsmeldungen sind statische Zuordnung von Flags (Bits) in der Steuerung zu Klartextmeldungen im Bediengerät.

Der Bereich der Zustandsmeldungen hat folgende Struktur

Tabelle 2-12 Zustandsmeldungen, MMICOM

Bitnummer	15	8	7	0
Variable X	Meldung 16 bis 9		Meldung 8 bis 1	
Variable X+1	Meldung 32 bis 25		Meldung 24 bis 17	
Variable X+2	Meldung 48 bis 41		Meldung 40 bis 33	

2.6.4 Eingabesyntax

Die folgende Grafik zeigt den Aufbau der Eingabesyntax für Variablen in der Projektiersoftware.

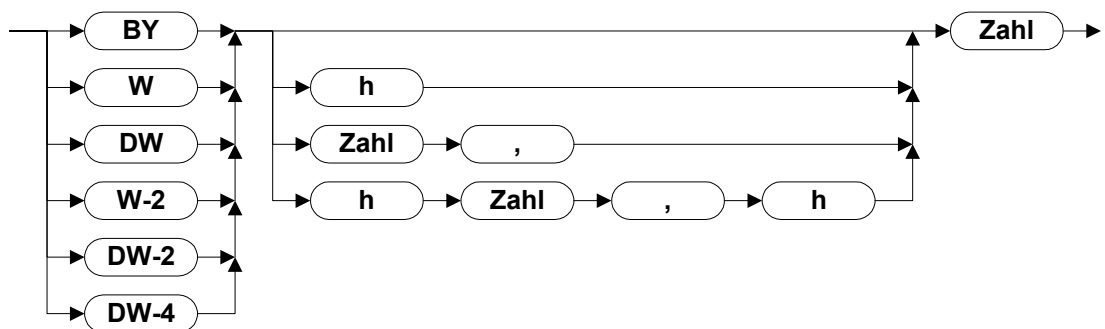


Bild 2-1 Syntaxdiagramm INTERBUS MMICOM

In der Projektiersoftware öffnet sich der Dialog **Variable**, um eine Steuerungsvariable anzulegen.

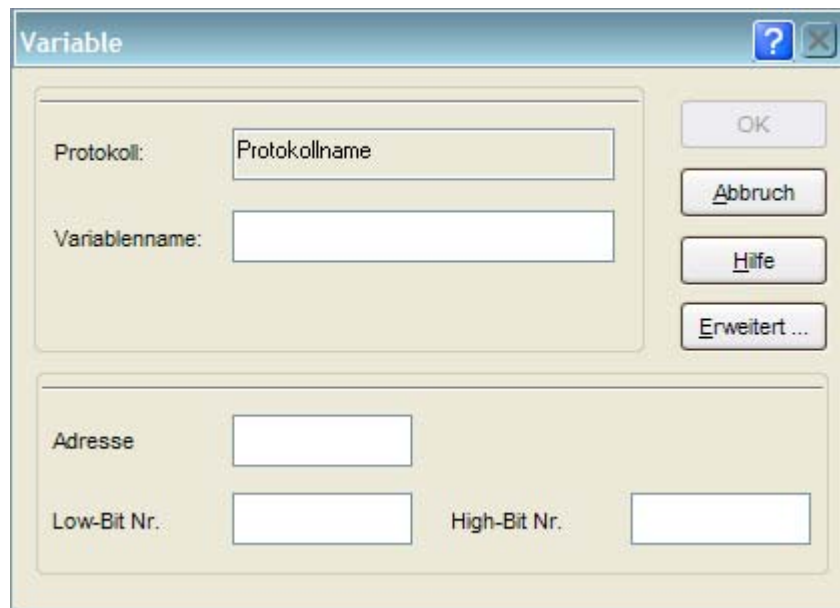


Bild 2-2 Dialog Variable

Variablenname:

Geben Sie in diesem Feld einen beliebigen Namen ein.

Adresse:

Geben Sie eine Adresse nach der oben angegebenen Eingabesyntax ein.

Low-Bit Nr. und High-Bit Nr.:

Wenn Sie **ein einzelnes Bit** von einem Byte, Wort oder Doppelwort adressieren möchten, geben Sie in beiden Feldern die gleiche Bitnummer an.

Beispiel:

Um im Datenbaustein 100, im Byte 10 das Bit 0 zu adressieren geben Sie folgende Werte an.

Adresse = BY100,10

Low-Bit Nr. = 0

High-Bit Nr. = 0

Wenn Sie **mehrere Bits** (Bitstream) von einem Byte, Wort oder Doppelwort adressieren möchten, geben Sie im Feld Low-Bit Nr. das niederwertigere Bit und im Feld High-Bit Nr. das höherwertigere Bit an.

Beispiel:

Um im Datenbaustein 100, im Byte 10 die Bits 0 bis 3 zu adressieren, geben Sie folgende Werte an.
 Adresse = BY 100,10
 Low-Bit Nr. = 0
 High-Bit Nr. = 3



Die Syntax für INTERBUS MMICOM ist steuerungsneutral.
 Beispiele für Siemens S7-Steuerungen:
 BY 100, 10 = DB100.DBB10,
 W 100, 10 = DB100.DBW10,
 DW 100, 10 = DB100.DBD10.
 Beispiele für PC WORX:
 BY100 = Bytearray, Byte 100,
 W100 = Wortarray, Wort 100,
 DW100 = Doppelwortarray, Doppelwort 100

2.6.5 Physikalische Ankopplung

2.6.5.1 Steckerbelegung

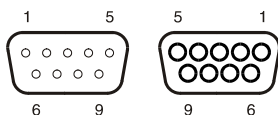


Bild 2-3 9-polige D-SUB-Stiftleiste und Buchsenleiste

Steckverbinder im Terminal: 9-polige D-SUB-Stiftleiste für Remotebus In.

Tabelle 2-13 Steckerbelegung Remotebus In (INTERBUS)

Pin	Bezeichnung	Funktion
1	DO	Dateneingang
2	DI	Datenausgang
3	GND	Betriebserde
4	nc	Nicht verbunden
5	nc	Nicht verbunden
6	/DO	Dateneingang invertiert
7	/DI	Datenausgang invertiert
8	nc	Nicht verbunden
9	nc	Nicht verbunden

Steckverbinder im Terminal: 9-polige D-SUB-Buchsenleiste für Remotebus Out.

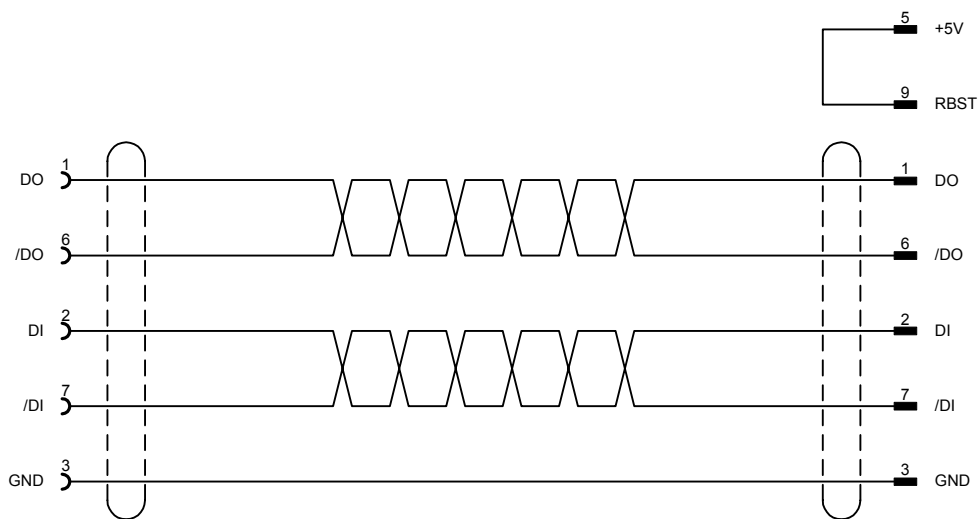
Tabelle 2-14 Steckerbelegung Remotebus Out (INTERBUS)

Pin	Bezeichnung	Funktion
1	DO	Datenausgang
2	DI	Dateneingang
3	GND	Betriebserde
4	nc	Nicht verbunden
5	+5 V	Spannungsversorgung +5 VDC
6	/DO	Datenausgang invertiert
7	/DI	Dateneingang invertiert
8	nc	Nicht verbunden
9	RBST	Fernbusstatus

2.6.5.2 2 - Leiter Fernbuskabel

Bediengerät
Remotebus IN

Remotebus OUT



D-SUB
Buchsenstecker
9-polig

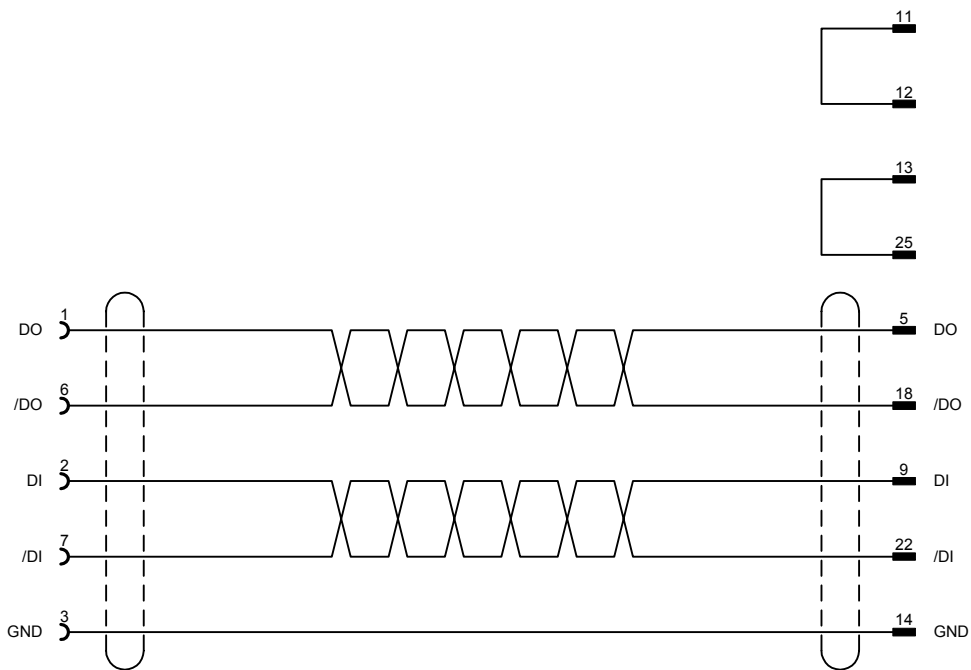
D-SUB
Stiftstecker
9-polig

2.6.5.3 Umsetzung von 8 - Leiterprotokoll auf 2 - Leiterprotokoll

Mit Busklemme IBS 24 BK.

Bediengerät
Remotebus IN

Remotebus OUT



D-SUB
Buchsenstecker
9-polig

D-SUB
Stiftstecker
25-polig

2.7 Fehlermeldungen

Fehlermeldungen werden am Bediengerät mit Code und Subcode angezeigt. Eine Fehlermeldung hat das folgende Schema:

```

Communication Error
Code           XXXXX
Subcode        XXXXX
Retries        XXXXX
    
```

Tabelle 2-15 Fehlermeldungen INTERBUS MMICOM, Schnittstelle 06, 14, 18, 21, 76

Code	Subcode	Bezeichnung	Mögliche Ursache
2		Die Steuerung hat Daten an das Bediengerät gesendet, die nicht angefordert wurden. Der Subcode gibt den empfangenen PD-Index dezimal an.	
40		Illegale Systemvariable.	Im Projekt befindet sich eine illegale Systemvariable.
50		Störungsbit von Steuerung ist gesetzt. Der Subcode gibt den empfangenen PD-Index dezimal an.	
51		Online-Bit von Steuerung ist nicht gesetzt. Der Subcode gibt das empfangene Status-Steuerbyte dezimal an.	Bus läuft nicht, SPS-Programm fehlt oder Steuerung befindet sich im STOPP-Zustand.
52		Standard-Bit und / oder Index-Bit von Steuerung ist nicht gesetzt. Der Subcode gibt das empfangene Status-Steuerbyte dezimal an.	Bus läuft nicht, SPS-Programm fehlt oder Steuerung befindet sich im STOPP-Zustand.
53		Falscher PD-Index empfangen. Der Subcode gibt den empfangenen PD-Index dezimal an.	
54		Handshake-Fehler (Empfangs-Timeout). Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.	SPS-Programm fehlt oder Steuerung befindet sich im STOPP-Zustand oder der Funktionsbaustein für das Bediengerät ist nicht aktiv.
55		Handshake-Fehler (Sende-Timeout). Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.	SPS-Programm fehlt oder Steuerung befindet sich im STOPP-Zustand.
56		Falsche Basisgröße der Variablen. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.	Falsche Zugriffsart angegeben.
57		Handshake-Fehler. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.	Handshake-Bits wurden von der Steuerung falsch gesetzt.

Tabelle 2-15 Fehlermeldungen INTERBUS MMICOM, Schnittstelle 06, 14, 18, 21, 76

Code	Subcode	Bezeichnung	Mögliche Ursache
58		Zugriffs-Fehler. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.	Auf dem INTERBUS werden keine gültigen Datenzyklen abgewickelt.
59		Falsche Variablennummer empfangen. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.	Die 8 Byte Nutzdaten sind nicht konsistent. Es liegt ein Fehler in der Anschaltbaugruppe vor.
60		Falscher PD-Index empfangen. Der Subcode gibt den empfangenen PD-Index dezimal an.	Die 8 Byte Nutzdaten sind nicht konsistent. Es liegt ein Fehler in der Anschaltbaugruppe vor.

Sie können die Fehlermeldungen mit dem Bediengerät protokollieren, indem Sie in einem Bild des Projekts die folgenden Systemvariablen einrichten.

- **ComParityCount** für die Anzahl der Fehlercodes 59.
- **ComOverrunCount** für die Anzahl der Fehlercodes 60.
- **ComFrameCount** für die Anzahl aller übrigen Fehlercodes.

2.8 Anwendungen

2.8.1 Siemens S7 Steuerung

2.8.1.1 Betriebsart Asynchron

Bei den Steuerungen S7-300 und S7-400 werden die Anschaltbaugruppen in der Regel in der Betriebsart Asynchron betrieben.

So stellen Sie die Betriebsart mit IBS CMD G4 ein:

1. Markieren Sie die Anschaltbaugruppe.
2. Wählen Sie aus dem Kontextmenü den Menüpunkt **Einstellungen**.

Der Dialog **Einstellungen Anschaltbaugruppe** öffnet sich.

3. Öffnen Sie die Karteikarte **Busbetrieb**.
4. Wählen Sie im Feld **Busbetrieb** die Betriebsart **Asynchron**.
5. Bestätigen Sie die Einstellungen mit **OK**.

2.8.1.2 Betriebsart Asynchron mit Synchronisationsimpuls

Sie können den INTERBUS sowohl an der S7-300, als auch an der S7-400 in der Betriebsart **Asynchron mit Synchronisationsimpuls** betreiben.

Empfehlenswert ist an der S7-400 jedoch nur die Betriebsart **Asynchron**.

So stellen Sie die Betriebsart Asynchron mit Synchronisationsimpuls ein:

1. Markieren Sie die Anschaltbaugruppe.
2. Wählen Sie aus dem Kontextmenü den Menüpunkt **Einstellungen**.

Der Dialog **Einstellungen Anschaltbaugruppe** öffnet sich.

3. Öffnen Sie die Karteikarte **Busbetrieb**.
4. Wählen Sie im Feld **Busbetrieb** die Betriebsart **Asynchron mit Synchronisationsimpuls**.
5. Klicken Sie auf den Schalter **Betriebsart Optionen**.

Der Dialog **Betriebsart Optionen** öffnet sich.

6. Markieren Sie das Kontrollkästchen **Interrupt-Modus parametrieren**.

Dadurch werden dem Steuerungssystem die Parameter Prozessalarm, Interrupt-Leitung 1 und Alarm OB 40 übergeben.

7. Bestätigen Sie die Einstellungen mit **OK**.

So parametrieren Sie Datensätze für die Übertragung der Prozessdaten:

1. Markieren Sie die Anschaltbaugruppe.
2. Wählen Sie aus dem Kontextmenü den Menüpunkt **Einstellungen**.

Der Dialog **Einstellungen Anschaltbaugruppe** öffnet sich.

3. Öffnen Sie die Karteikarte **IBS Datensätze**.

Auf der Karteikarte **IBS Datensätze** können Sie die folgenden Parameter einstellen:

Tabelle 2-16 Parameter für IBS Datensätze

Parameter	Beschreibung
Satz	Wählen Sie die Nummer des Datensatzes, den Sie parametrieren wollen.
Bereich	Stellen Sie den Zielbereich (DB, E, M), bzw. den Quellbereich (A, DB, M) in der Steuerung ein.
Datenbaustein	Wenn Sie als Ziel-/Quellbereich DB ausgewählt haben, tragen Sie die Nummer Ihres Ziel-/Quelldatenbausteins ein.
Offset (Byte)	Tragen Sie die Startadresse der zu übertragenden Daten im ausgewählten Ziel-/Quellbereich ein.
Länge (Byte)	Geben Sie die Länge der zu übertragenden Daten ein (für Bediengeräte 8 Byte).

4. Bestätigen Sie die Einstellungen mit **OK**.

2.8.1.3 Bediengerät als INTERBUS-Teilnehmer konfigurieren

So fügen Sie ein Bediengerät dem INTERBUS hinzu:

1. Markieren Sie die Anschaltbaugruppe.
2. Wählen Sie aus dem Kontextmenü den Menüpunkt **Einfügen mit ID-Code**.

Der Dialog **Teilnehmer einfügen** öffnet sich.

3. Geben Sie den **ID-Code** entweder hexadezimal (**2F**) oder dezimal (**47**) ein.
4. Geben Sie für die **Breite des Prozessdatenkanals** den Wert **8** ein.
5. Markieren Sie für die **Teilnehmerart** den Auswahlknopf **Fernbusteilnehmer**.
6. Bestätigen Sie Ihre Eingaben mit **OK**.

Der Dialog **Teilnehmerbeschreibung** öffnet sich. Sie können in diesen Dialog genauere Angaben zur Beschreibung des Bediengeräts machen.

7. Bestätigen Sie diesen Dialog ebenfalls mit **OK**.

Die Konfiguration ist abgeschlossen. Sichern Sie die Konfiguration auf dem Parametrisierungsspeicher.

2.8.1.4 Funktionsbausteine von Phoenix Contact

Für die Betriebsart Asynchron mit Synchronisationsimpuls werden von der Firma Phoenix Contact die folgenden Funktionen und Datenbausteine zur Verfügung gestellt:

- FC20 (INIT_IB),
- FC21 (MEM_READ),
- FC22 (MEM_WRIT),
- FC24 (IB_DIAG),
- FC28 (IB_SYNC) und
- DB20 (IBDB).

Für die Steuerung S7-300 benötigen Sie in der Betriebsart **Asynchron** den INTERBUS-Datenbaustein ebenfalls.

2.8.1.5 FC20 (INIT_IB)

Die Funktion FC20 wird beim Anlauf der Steuerung (OB100) einmalig aufgerufen. Sie regelt das Anlaufverhalten der Anschaltbaugruppe und initialisiert den INTERBUS-Datenbaustein (IBDB, im folgenden Beispiel DB20) mit den erforderlichen Daten. Der Timer T1 wird hart verwendet.

Parameter:

Das Busy-Bit muss beim Aufruf gesetzt sein, da sonst keine Aktion erfolgt.

```

UN #FC20BusyBit
S #FC21BusyBit
CALL FC 20
IBDB           := DB20
COM_ADR        := 256
DIAG_STATE     := 0
DIAG_PARA      := 0
FKN_START      := 0
FKN_PARA       := 0
FKN_STATE      := 0
MEM_READ       := 21
MEM_WRITE      := 22
LOAD           := 0
BOOT           := 0
MODE           := 0
TIMER_NR       := T1
SOURCE         := 0
CONFIGURATION  := DW#16#8000000F
RET            := #FC20RetBit
BUSY           := #FC20BusyBit
    
```

Tabelle 2-17 Parameter der Funktion FC20

Parameter	Wert	Bedeutung
IBDB	DB20	Nummer des globalen INTERBUS-Datenbausteins. In diesem Baustein werden von der Funktion FC20 die Einstellungen vorgenommen.
COM_ADR		S7-Basisadresse der Anschaltbaugruppe
DIAG_STATE	0	S7-Adresse des Diagnosestatus-Registers. Diese ist von der Basisadresse der Anschaltbaugruppe abhängig und wird automatisch vergeben. Tragen Sie deshalb hier den Wert 0 (Null) ein.
DIAG_PARA	0	S7-Adresse des Diagnoseparameter-Registers. Diese ist von der Basisadresse der Anschaltbaugruppe abhängig und wird automatisch vergeben. Tragen Sie deshalb hier den Wert 0 (Null) ein.
FKN_START	0	S7-Adresse des Standard-Funktions-Startregisters. Diese ist von der Basisadresse der Anschaltbaugruppe abhängig und wird automatisch vergeben. Tragen Sie deshalb hier den Wert 0 (Null) ein.
FKN_PARA	0	S7-Adresse des Standard-Funktions-Parameterregisters. Diese ist von der Basisadresse der Anschaltbaugruppe abhängig und wird automatisch vergeben. Tragen Sie deshalb hier den Wert 0 (Null) ein.

Tabelle 2-17 Parameter der Funktion FC20

Parameter	Wert	Bedeutung
FKN_STATE	0	S7-Adresse des Standard-Funktions-Statusregisters. Diese ist von der Basisadresse der Anschaltbaugruppe abhängig und wird automatisch vergeben. Tragen Sie deshalb hier den Wert 0 (Null) ein.
MEM_READ		Bausteinnummer (21) des Lesetreibers FC21 (MEM_READ)
MEM_WRITE		Bausteinnummer (22) des Schreibtreibers FC22 (MEM_WRITE)
LOAD		Laden der Konfiguration der Anschaltbaugruppe
	0 (Flash)	Laden der Konfiguration aus dem Parametrierungsspeicher. Voraussetzung ist die Projektierung über IBS CMD/Parametrierungsspeicher.
	1 (APPLIC)	Laden der Konfiguration über Funktionsbausteine der Applikation
	2 (CMD_IB)	Laden der Konfiguration über eine CMD-Datei und Datenbausteine
IN_BOOT		Dieser Parameter regelt den Anlauf des INTERBUS (nur bei LOAD = 0 relevant)
	0	Start des Bussystems aus der Projektierung heraus (die auf der Flash-Karte hinterlegte Parametrierung enthält den Start des Bussystems)
	1	Start des Bussystems mit Konfigurationsrahmen 1 über den Initialisierungstreiber
MODE		Betriebsart (BA) der Anschaltbaugruppe
	0 (A_SYN)	Asynchron
	1 (A_SYN_IMP)	Asynchron mit Synchronisationsimpuls als Interrupt
TIMER_NR		Nummer des Timers, der von der Funktion benutzt wird
SOURCE		Nummer des Parametrierungs-DB bei LOAD = 2

Tabelle 2-17 Parameter der Funktion FC20

Parameter	Wert	Bedeutung
CONFIGURATION		32 Konfigurations-Bits für INTERBUS-Bausteine
	Bit 0 = 0	Peripheriefehler-Einträge (PF-Einträge) werden von der Funktion IB_DIAG im IBDB eingetragen
	Bit 0 = 1	Peripheriefehler-Einträge (PF-Einträge) werden von der Funktion IB_DIAG im IBDB nicht eingetragen
	Bit 1 = 0	Anwenderfehler-Einträge (USER-Einträge) werden von der Funktion IB_DIAG im IBDB eingetragen
	Bit 1 = 1	Anwenderfehler-Einträge (USER-Einträge) werden von der Funktion IB_DIAG im IBDB nicht eingetragen
	Bit 2 = 0	Busfehler-Einträge (BUS-Einträge) werden von der Funktion IB_DIAG im IBDB eingetragen
	Bit 2 = 1	Busfehler-Einträge (BUS-Einträge) werden von der Funktion IB_DIAG im IBDB nicht eingetragen
	Bit 3 = 0	Controller-Einträge (CTRL-Einträge) werden von der Funktion IB_DIAG im IBDB eingetragen
	Bit 3 = 1	Controller-Einträge (CTRL-Einträge) werden von der Funktion IB_DIAG im IBDB nicht eingetragen
	Bit 4 = 0	Reserviert
	Bit 4 = 1	Reserviert
	Bit 31 = 0	Nicht relevant
	Bit 31 = 1	Dieses Bit muss immer gesetzt sein. Dadurch erkennen alle Treiberbausteine, dass es sich um eine S7-300 DSC-Anschaltbaugruppe handelt
	DW#16#8000000F	Es werden keine Fehler im IBDB eingetragen
	DW#16#80000000	Es werden alle Fehler im IBDB eingetragen

Tabelle 2-17 Parameter der Funktion FC20

Parameter	Wert	Bedeutung
RET		Ist das Ergebnisbit gesetzt (1), steht eine Fehlerkennung im DBW56 des IBDB mit folgender Bedeutung: High-Byte DBB56 = Fehlernummer Low-Byte DBB57 = Nummer des fehlerhaften Schritts
	Fehlernummer 1	Timeout
	Fehlernummer 2	Negatives Ergebnis bei „Stopp und Anforderung der Neukonfiguration“
	Fehlernummer 3	Negatives Ergebnis bei „Start Data-Transfer“
	Fehlernummer 4	Parameter „LOAD“ nicht gültig
	Fehlernummer 5	Kommunikationsregister nicht ansprechbar (Basisadresse falsch, Hardware defekt)
	Fehlernummer 6	Fehler in Parametrierungssequenz (über Parameter SOURCE): <ul style="list-style-type: none"> – das angezeigte Diagnose-Statusregister ist im IBDB an DBW 76 eingetragen – das angezeigte Diagnose-Parameterregister ist im IBDB an DBW 78 eingetragen – der fehlerhafte Dienst-Code ist im IBDB an DBW 74 eingetragen und dient zu internen Zwecken
BUSY	1	Funktion aktiv. Dieses Bit muss beim Aufruf gesetzt sein.

2.8.1.6 FC21 (MEM_READ)

Die Funktion FC21 (MEM_READ) liest Daten von der Anschaltbaugruppe und kopiert diese in den Zielbereich der Steuerung.

Parameter:

Als Parameter wird die Nummer des INTERBUS-Datenbausteins (IBDB, im folgenden Beispiel DB20), übergeben. Das Busy-Bit muss beim Aufruf gesetzt sein, da sonst keine Aktion erfolgt.

```

UN #FC21BusyBit
S #FC21BusyBit
CALL FC21
IBDB           := DB20
MODE           := 1
SOURCE         := 0
DEST_AREA     := 0
DEST_AREA_NR  := 0
DEST_OFFSET   := 0
DEST_LENGTH   := 0
RET           := #FC21RetBit
BUSY          := #FC21BusyBit
    
```

Tabelle 2-18 Parameter der Funktion FC21

Parameter	Wert	Bedeutung
IBDB	DB20	Nummer des globalen INTERBUS-Datenbausteins. Hier DB20.
MODE	1	Betriebsart der Funktion. Hier Mode 1 (Lesen von Datensatz 1. Dieser muss in IBS CMD parametrisiert werden.
SOURCE	0	Datenquelle. Im Mode 1 ohne Bedeutung. Immer 0.
DEST_AREA	0	Datenzielbereich. Im Mode 1 ohne Bedeutung. Immer 0.
DEST_AREA_NR	0	Nummer des Datenbausteins, wenn Zielbereich Daten sind. Im Mode 1 ohne Bedeutung. Immer 0.
DEST_OFFSET	0	Startadresse in der DEST_AREA. Im Mode 1 ohne Bedeutung. Immer 0.
DEST_LENGTH	0	Anzahl der zu lesenden Worte. Im Mode 1 ohne Bedeutung. Immer 0.
RET		Ergebnisbit. Bei gesetztem Ergebnisbit steht eine Fehlererkennung im DBW58 des IBDB mit folgender Bedeutung: DBW58 = Fehlernummer 1 bis 5.
	Fehlernummer = 1	Über den Parameter DEST_AREA_NR wurde DB0 als Ziel eingegeben
	Fehlernummer = 2	Über den Parameter DEST_AREA wurde ein unzulässiger Zielbereich angegeben
	Fehlernummer = 3	Über den Parameter MODE wurde eine unzulässige Betriebsart angegeben
	Fehlernummer = 4	Über den Parameter SOURCE wurde eine ungültige CR-Nummer (Nummer eines PCP-fähigen INTERBUS-Teilnehmers) angegeben
	Fehlernummer = 5	Es wurde ein Datensatz angegeben, der in IBS CMD nicht aktiviert wurde
Busy	1	Funktion aktiv. Dieses Bit muss beim Aufruf gesetzt sein.

2.8.1.7 FC22 (MEM_WRIT)

Die Funktion FC22 schreibt Daten aus dem Quellbereich der Steuerung in den angegebenen Zielbereich auf der Anschaltbaugruppe.

Parameter:

Als Parameter wird die Nummer des INTERBUS-Datenbausteins (IBDB, im folgenden Beispiel DB20), übergeben. Das Busy-Bit muss beim Aufruf gesetzt sein, da sonst keine Aktion erfolgt.

```

UN #FC22BusyBit
S #FC22BusyBit
CALL FC 22
IBDB          := DB20
MODE          := 1
SOURCE_AREA   := 0
SOURCE_AREA_NR := 0
SOURCE_OFFSET := 0
SOURCE_LENGTH := 0
DESTINATION   := 0
RET          := #FC22RetBit
BUSY         := #FC22BusyBit
    
```

Tabelle 2-19 Parameter der Funktion FC22

Parameter	Wert	Bedeutung
IBDB	DB20	Nummer des globalen INTERBUS-Datenbausteins. Hier DB20.
MODE	1	Betriebsart der Funktion. Hier Mode 1 (Schreiben von Datensatz 1. Dieser ist in IBS CMD zu parametrieren.
SOURCE_AREA	0	Datenquelle. Im Mode 1 ohne Bedeutung. Immer 0.
SOURCE_AREA_NR	0	Nummer des Datenbausteins, wenn Quellbereich Daten sind. Im Mode 1 ohne Bedeutung. Immer 0.
SOURCE_OFFSET	0	Startadresse in der SOURCE_AREA. Im Mode 1 ohne Bedeutung. Immer 0.
SOURCE_LENGTH	0	Anzahl der zu schreibenden Worte. Im Mode 1 ohne Bedeutung. Immer 0.
DESTINATION	0	Datenziel. Im Mode 1 ohne Bedeutung. Immer 0.
RET		Ergebnisbit. Bei gesetztem Ergebnisbit steht eine Fehlerkennung im DBW60 des IBDB mit folgender Bedeutung: DBW60 = Fehlernummer 1 bis 5.
	Fehlernummer = 1	Über den Parameter SOURCE_AREA_NR wurde DB0 als Quelle eingegeben
	Fehlernummer = 2	Über den Parameter SOURCE_AREA wurde ein unzulässiger Quellbereich angegeben
	Fehlernummer = 3	Über den Parameter MODE wurde eine unzulässige Betriebsart angegeben
	Fehlernummer = 4	Über den Parameter DESTINATION wurde eine ungültige CR-Nummer (Nummer eines PCP-fähigen INTERBUS-Teilnehmers) angegeben
	Fehlernummer = 5	Es wurde ein Datensatz angegeben, der in IBS CMD nicht aktiviert wurde
Busy	1	Funktion aktiv. Dieses Bit muss beim Aufruf gesetzt sein.

2.8.1.8 FC28 (IB_SYNC)

Die Funktion FC28 regelt in der Betriebsart „Asynchron mit Synchronisationsimpuls“ das Protokoll und überwacht die Übertragung konsistenter Daten größer 16 Bit. Außerdem benötigt sie Funktionen zum Lesen und Schreiben der E/A-Daten (im folgenden Beispiel FC50 und FC51) Diese Funktionen müssen ohne Parameter programmiert werden.

Parameter:

```
Call FC 28
IBDB:=      DB20
READ_FC:=   FC50
WRITE_FC:=  FC51
RET:=      M10.1
```

Tabelle 2-20 Parameter der Funktion FC28

Parameter	Wert	Bedeutung
IBDB	DB20	Nummer des globalen INTERBUS-Datenbausteins. Hier DB20.
READ_FC		Funktion, die zum Lesen von Eingangsdaten aufgerufen werden soll
WRITE_FC		Funktion, die zum Schreiben von Ausgangsdaten aufgerufen werden soll
RET	0	Funktion wurde fehlerfrei durchlaufen.
	1	Fehler. Die Funktion wurde verlassen.

2.8.1.9 Funktionsbausteine von Süttron electronic

Im Gegensatz zu einer S5-Steuerung verfügt die S7-Steuerung in allen Speicherbereichen über eine Byte-Struktur. Daher müssen Sie für Variablen, die mehr als 4 Byte in der Steuerung belegen, folgende Bezeichner verwenden:

W-2 anstelle W für:

- den Pollbereich
- das parallele Meldungssystem
- das Uhrzeitupdate von und zur Steuerung
- die Rezepturpuffer

DW-4 anstelle DW für:

- alphanumerische Variablen; die Länge der Variable muss durch 4 teilbar sein.

Wird eine Variable mit mehr als 4 Byte Speicherbedarf versehentlich mit der falschen Datenkennung versehen, werden die Nutzdaten falsch interpretiert. Die Variable gibt keine korrekten Werte aus.

Achtung :

Beim Hochlauf des Bediengeräts wird in einer Testkommunikation ein 1-Byte-Lesezugriff auf die Variable mit der Nummer 0 durchgeführt. Dabei werden die gelesenen Daten nicht interpretiert, das heißt der Lesefunktionsbaustein muss diese Variablennummer nicht unbedingt dekodieren.

2.8.1.10 Dekodierung der Variablennummer

Die Schreib- und Lesefunktion der Funktionsbausteine dekodiert die empfangene Variablennummer nach folgendem Schema:

Beispiel: W h 4BC8

4				B				C				8			
0	1	0	0	1	0	1	1	1	1	0	0	1	0	0	0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Datenbaustein-Nummer								Byte-/Wort-/Doppelwort-Nummer							

Kürzel	Wert	Bedeutung
W		Datenbreite
	DW	Doppelwort = 32 Bit
	DW-4	Doppelwort = 32 Bit
	W	Wort = 16 Bit
	W-2	Wort = 16 Bit
	BY	Byte = 8 Bit
	W	Bit-Zugriff auf Wort-Adresse (bei Angabe der Bit-Adresse)
	BY	Bit-Zugriff auf Byte-Adresse (bei Angabe der Bit-Adresse)
h		Hexadezimalzahl
4BC8		Zahlencode für Datenkennung 4B = Datenbausteinnummer (hier DB75) C8 = Doppelwort-/Wort-/Byte-Nummer (hier Wort 200)

2.8.1.11 Übersicht über Funktionsbausteine von Süttron electronic

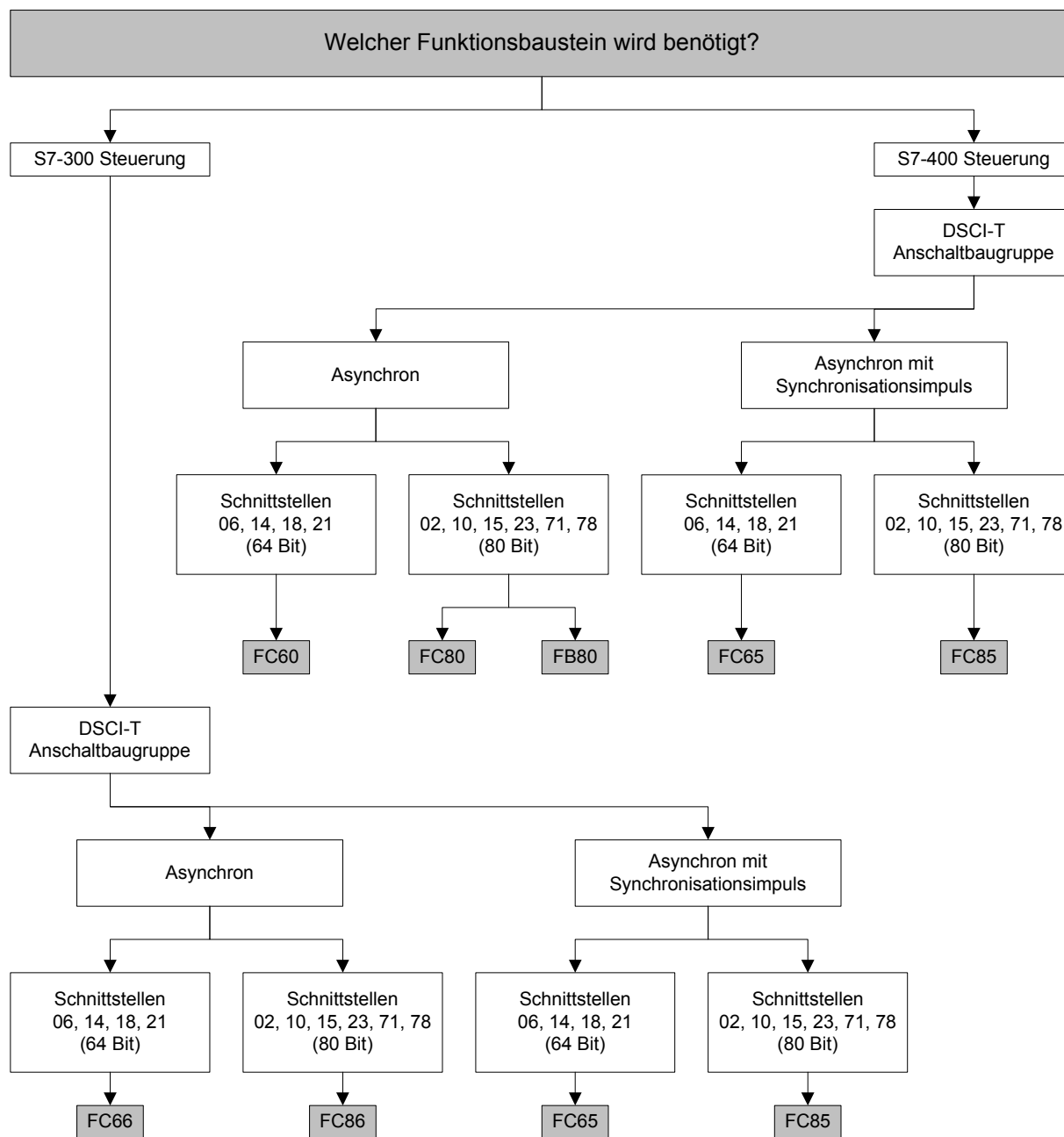


Bild 2-4 Übersicht über Funktionsbausteine von Süttron electronic

2.8.1.12 Quellen und Bausteine einfügen

Die mitgelieferten AWL-Dateien von Süttron werden im SIMATIC-Manager über den Eintrag **Quellen** importiert. Dazu wählen Sie den aus dem Menü **Einfügen** den Menüpunkt **Externe Quelle**.

Welche Bausteine Sie benötigen, hängt vom Steuerungstyp und der Anschaltung des Bediengeräts ab (siehe Diagramm).

Die Dateien finden Sie im Verzeichnis der Projektierungssoftware unter

...FBs\INTERBUS\SIEMENS\S7\MMI COM Schnittstelle [02,10,15, 23,71,78]\300 ASYNC

...FBs\INTERBUS\SIEMENS\S7\MMI COM Schnittstelle [02,10,15,23,71,78]\300_400 ASYMSI

...\FBs\INTERBUS\SIEMENS\S7\MMI COM Schnittstelle [02,10,15,23,71,78]\400 ASYNC

...\FBs\INTERBUS\SIEMENS\S7\MMI COM Schnittstelle [06,14,18,21]\300 ASYNC

...\FBs\INTERBUS\SIEMENS\S7\MMI COM Schnittstelle [06,14,18,21]\300_400 ASYMSI

...\FBs\INTERBUS\SIEMENS\S7\MMI COM Schnittstelle [06,14,18,21]\400 ASYNC

Die Zahlen in den eckigen Klammern bezeichnen den Schnittstellentyp des Bediengeräts.

Die eingelesenen AWL-Quellen müssen anschließend mit dem AWL-Editor in das S7 Bausteinformat übersetzt werden.



Die Funktionen und Funktionsbausteine der Fa. Süttron verwenden die registerindirekte, bereichsübergreifende Adressierungsform. Werden innerhalb des Programmzyklus interruptgesteuerte Programmaufrufe zugelassen, müssen neben den Inhalten der beiden Akkumulatoren die Inhalte der Adressregister 1+2 und unter Umständen auch der Lokaldatenstack gesichert werden. Weitere Informationen hierzu finden Sie auch im Programmierhandbuch Ihrer S7-Steuerung (Systemsoftware S7-300/400, Programmentwurf) unter dem Kapitel "Vermeiden von Fehlern beim Aufrufen von Bausteinen".



Auf die Steuerungsdaten kann das Bediengerät Bit-, Byte-, Wort- oder Doppelwortweise zugreifen. Der Bit-Zugriff ist auf eine Byte- oder Wortadresse möglich. Da das MMICOM-Profil selbst keine Übertragung von Datentypkennungen vorsieht, wird dies mittels der Variablennummer realisiert. Jede Variable wird durch eine solche Nummer eindeutig gekennzeichnet. Sie wird während der Projektierung in der Projektierungssoftware festgelegt.



Die Funktionen müssen im Anwenderprogramm zyklisch aufgerufen werden. Für jedes angeschlossene Bediengerät muss ein eigener Funktionsbaustein aufgerufen werden.

2.8.1.13 FC60 für Betriebsart Asynchron an S7-400

Diese Funktion dekodiert die vom Bediengerät übertragenen Daten und führt dementsprechend Lese- und Schreibzugriffe aus. Die Datenbreite beträgt 8 Byte. Zur Optimierung der Übertragungszeit greift der Baustein direkt auf die Peripherieadressen und nicht auf das Prozessabbild zu. DBOFFSET und VAROFFSET erlauben auch Zugriffe auf Datenbausteine und Byte-Adressen größer 255.

Parameter:

```
Call FC 60
PIN:=      P#P10.0
POUT:=     P#P10.0
DEST:=P#DB101.DBX0.0 Byte 8
SOURCE:=P#DB101.DBX8.0 Byte 8
VAROFFSET:=      0
DBOFFSET:=       0
```

Tabelle 2-21 Parameter der Funktion FC60

Parameter	Wert	Bedeutung
PIN	P#P10.0	Anfangsadresse der P-Eingänge des Bediengeräts (Beispiel: P#P10.0 = PEB10)
POUT	P#P10.0	Anfangsadresse der P-Ausgänge des Bediengeräts (Beispiel: P#P10.0 = PAB10)
DEST		Speicheradresse der Eingangsdaten. Diese Adresse darf nicht von anderen Programmteilen überschrieben werden.
SOURCE		Speicheradresse der Ausgangsdaten. Diese Adresse darf nicht von anderen Programmteilen überschrieben werden.
DBOFFSET		DB-Offset der zur Datenbausteinadresse der Variablen addiert wird
VAROFFSET		Variablen-Offset, der zur Byte-Adresse der Variablen addiert wird

2.8.1.14 FC66 für Betriebsart Asynchron an S7-300

Diese Funktion dekodiert die vom Bediengerät übertragenen Daten und führt dementsprechend Lese- und Schreibzugriffe aus. Die Datenbreite beträgt 8 Byte.

Da dieser Baustein direkt auf das MPM (Multi Port Memory) der Anschaltbaugruppe zugreift, muss das Bediengerät mit dem INTERBUS Projektierwerkzeug CMD in einen eigenen INTERBUS Datensatz parametrieren werden.

Dieser INTERBUS Datensatz darf nicht über Aufrufe von FC21 (MEM_READ) / FC22 (MEM_WRIT) innerhalb des OB1 auf den E/A-Bereich aktualisiert werden. Die Berechnung der Parameter PIN / POUT erfolgt mit folgender Formel:

$$PIN/POUT = MPM_Offset_Interbus_Datensatz + (CMD_Prozessdaten_Zuordnung_IBS_Teilnehmer - CMD_Interbus_Datensatz_Offset)$$

Erläuterung:

Der **MPM_Offset_Interbus_Datensatz** ist für die INTERBUS Datensätze wie folgt definiert:

DS1 = 0 / DS2 = 128 / DS3 = 256 / DS4 = 384 / DS5 = 512 (alle Werte dezimal)

CMD_Prozessdaten_Zuordnung_IBS_Teilnehmer:

Siehe CMD Projektierung --> Bedienterminal --> Prozessdaten --> Parameter Zuordnung

CMD_Interbus_Datensatz_Offset:

Siehe CMD-Projektierung --> Anschaltbaugruppe --> Einstellungen --> IBS Datensätze für Eingänge / Ausgänge --> Parameter Offset

Beispiel (alle Werte sind dezimal):

Ein Bedienterminal im INTERBUS Datensatz 3

MPM_Offset_Interbus_Datensatz = 256

CMD_Interbus_Datensatz_Offset = 256

CMD_Prozessdaten_Zuordnung_IBS_Teilnehmer = 256 für E/A

PIN = POUT = 256 + (256 - 256) = 256

DBOFFSET und VAROFFSET erlauben auch Zugriffe auf Datenabusteine und Byte-Adressen größer 255.

Parameter:

```
Call FC 66
PIN:=          10
POUT:=         10
IBDB:=         DB20
DEST:= DB10.DBX0.0
SOURCE:=DB10.DBX8.0
VAROFFSET:=    0
DBOFFSET:=    0
```

Tabelle 2-22 Parameter der Funktion FC66

Parameter	Wert	Bedeutung
PIN	10	Offset innerhalb der INTERBUS-Eingangsdaten (Startadresse der Bediengerät-Eingangsdaten) (Beispiel: 10 = E10)
POUT	P#P10.0	Anfangsadresse der P-Ausgänge des Bediengeräts (Beispiel: P#P10.0 = PAB10)
IBDB	DB20	INTERBUS Datenbaustein
DEST		Zieladresse der Bediengerätedaten in der Steuerung
SOURCE		Quelladresse der Bediengerätedaten in der Steuerung
DBOFFSET		DB-Offset der zur Datenbausteinadresse der Variablen addiert wird
VAROFFSET		Variablen-Offset, der zur Byte-Adresse der Variablen addiert wird

2.8.1.15 FB80 für Betriebsart Asynchron an S7-400

Beim Aufruf des Funktionsbausteins muss ein Instanzdatenbaustein angegeben werden. Die darin gespeicherten Protokoll Daten für Ein- und Ausgänge sind nur für den FB88 sichtbar.

Dieser Baustein dekodiert die vom Bediengerät übertragenen Daten und führt dementsprechend Lese- und Schreibzugriffe aus. Die Datenbreite beträgt 10 Byte. Der Baustein greift direkt auf die Peripherieadressen zu.

DBOFFSET und VAROFFSET erlauben auch Zugriffe auf Datenbausteine und Byte-Adressen größer 255.

Die verfügbaren E-/A-Punkte des Moduls werden jeweils an Byte-Position 10 im IN- und OUT-Rahmen des Moduls abgebildet und durch den Funktionsbaustein nicht beeinflusst.

Parameter:

```
Call FB 80, DB101
P_IN:=          10
P_OUT:=         10
DBOFFSET:=      0
VAROFFSET:=     0
```

Tabelle 2-23 Parameter der Funktion FB80

Parameter	Wert	Bedeutung
P_IN	10	Anfangsadresse der Eingänge des Bediengeräts (INTEGER) (Beispiel: 10 = PEB10)
P_OUT	10	Anfangsadresse der Ausgänge des Bediengeräts (Beispiel: 10 = PAB10)
IBDB	DB20	INTERBUS Datenbaustein
DEST		Zieladresse der Bediengerätedaten in der Steuerung
SOURCE		Quelladresse der Bediengerätedaten in der Steuerung
DBOFFSET		DB-Offset der zur Datenbausteinadresse der Variablen addiert wird
VAROFFSET		Variablen-Offset, der zur Byte-Adresse der Variablen addiert wird

2.8.1.16 FC86 für Betriebsart Asynchron an S7-300

Diese Funktion dekodiert die vom Bediengerät übertragenen Daten und führt dementsprechend Lese- und Schreibzugriffe aus. Die Datenbreite beträgt 10 Byte.

Da dieser Baustein direkt auf das MPM (Multi Port Memory) der Anschaltbaugruppe zugreift, muss das Bediengerät mit dem INTERBUS Projektierwerkzeug CMD in einen eigenen INTERBUS Datensatz parametrieren werden.

Dieser INTERBUS Datensatz darf nicht über Aufrufe von FC21 (MEM_READ) / FC22 (MEM_WRIT) innerhalb des OB1 auf den E/A-Bereich aktualisiert werden. Die Berechnung der Parameter PIN / POUT erfolgt mit folgender Formel:

$$PIN/POUT = MPM_Offset_Interbus_Datensatz + (CMD_Prozessdaten_Zuordnung_IBS_Teilnehmer - CMD_Interbus_Datensatz_Offset)$$

Erläuterung:

Der **MPM_Offset_Interbus_Datensatz** ist für die INTERBUS Datensätze wie folgt definiert:

DS1 = 0 / DS2 = 128 / DS3 = 256 / DS4 = 384 / DS5 = 512 (alle Werte dezimal)

CMD_Prozessdaten_Zuordnung_IBS_Teilnehmer:

Siehe CMD Projektierung --> Bedienterminal --> Prozessdaten --> Parameter Zuordnung

CMD_Interbus_Datensatz_Offset:

Siehe CMD-Projektierung --> Anschaltbaugruppe --> Einstellungen --> IBS Datensätze für Eingänge / Ausgänge --> Parameter Offset

Beispiel (alle Werte sind dezimal):

Ein Bedienterminal im INTERBUS Datensatz 3

MPM_Offset_Interbus_Datensatz = 256

CMD_Interbus_Datensatz_Offset = 256

CMD_Prozessdaten_Zuordnung_IBS_Teilnehmer = 256 für E/A

PIN = POUT = 256 + (256 - 256) = 256

DBOFFSET und VAROFFSET erlauben auch Zugriffe auf DBs und Byteadressen größer 255.

Parameter:

```
Call FC 86
PIN:=          10
POUT:=         10
IBDB:=         DB20
DEST:= DB10.DBX0.0
SOURCE:=DB10.DBX10.0
DBOFFSET:=     0
VAROFFSET:=    0
```

Tabelle 2-24 Parameter der Funktion FC86

Parameter	Wert	Bedeutung
PIN	10	Offset innerhalb der INTERBUS-Eingangsdaten (Startadresse der Bediengerät-Eingangsdaten) (Beispiel: 10 = E10)
POUT	10	Offset innerhalb der INTERBUS-Ausgangsdaten (Startadresse der Bediengerät-Ausgangsdaten) (Beispiel: 10 = PAB10)
IBDB	DB20	INTERBUS Datenbaustein (Beispiel: DB20)
DEST	DB10.DBX0.0	Zieladresse der Bediengerätedaten in der Steuerung (Beispiel: DB10.DBX0.0)

Tabelle 2-24 Parameter der Funktion FC86

Parameter	Wert	Bedeutung
SOURCE	DB10.DBX10.0	Quelladresse der Bediengerätedaten in der Steuerung (Beispiel: DB10.DBX10.0)
DBOFFSET	0	DB-Offset der zur Datenbausteinadresse der Variablen addiert wird (Beispiel: 0)
VAROFFSET	0	Variablen-Offset, der zur Byte-Adresse der Variablen addiert wird (Beispiel: 0)

2.8.1.17 FC65 für Betriebsart Asynchron mit Synchronisationsimpuls an S7-300

Diese Funktion dekodiert die vom Bediengerät übertragenen Daten und führt dementsprechend Lese- und Schreibzugriffe aus. Die Datenbreite beträgt 8 Byte. Es wird auf die Datensätze zugegriffen, die in IBS CMD parametrisiert wurden.

DBOFFSET und VAROFFSET erlauben auch Zugriffe auf DBs und Byteadressen größer 255.

Parameter:

```
Call FC 65 PIN := EB10 POUT:=AB10 DBOFFSET:=0 VAROFFSET:=0
```

Tabelle 2-25 Parameter der Funktion FC65

Parameter	Wert	Bedeutung
PIN	EB10	Anfangsadresse der Eingänge des Bediengeräts (Beispiel: EB10)
POUT	AB10	Anfangsadresse der Ausgänge des Bediengeräts (Beispiel: AB10)
DBOFFSET	0	DB-Offset der zur Datenbausteinadresse der Variablen addiert wird (Beispiel: 0)
VAROFFSET	0	Variablen-Offset, der zur Byte-Adresse der Variablen addiert wird (Beispiel: 0)

2.8.1.18 FC85 für Betriebsart Asynchron mit Synchronisationsimpuls an S7-300

Diese Funktion dekodiert die vom Bediengerät übertragenen Daten und führt dementsprechend Lese- und Schreibzugriffe aus. Die Datenbreite beträgt 10 Byte. Es wird auf die Datensätze zugegriffen, die in IBS CMD parametrieren wurden.

DBOFFSET und VAROFFSET erlauben auch Zugriffe auf DBs und Byteadressen größer 255.

Parameter:

```
Call FC 85PIN := EB10POUT:=AB10DBOFFSET:=0VAROFFSET:=0
```

Tabelle 2-26 Parameter der Funktion FC85

Parameter	Wert	Bedeutung
PIN	EB10	Anfangsadresse der Eingänge des Bediengeräts (Beispiel: EB10)
POUT	AB10	Anfangsadresse der Ausgänge des Bediengeräts (Beispiel: AB10)
DBOFFSET	0	DB-Offset der zur Datenbausteinadresse der Variablen addiert wird (Beispiel: 0)
VAROFFSET	0	Variablen-Offset, der zur Byte-Adresse der Variablen addiert wird (Beispiel: 0)

2.8.1.19 Beispiel für Betriebsart Asynchron

OB1 und OB100 für S7-300

Beim Aufruf der Funktion FC20 (INIT_IB) im OB100 müssen Sie den Parameter MODE auf Null (0) setzen.

OB1

```
...
Call FC 66
PIN          10
POUT         10
IBDB         DB20
DEST        DB10.DBX0.0
SOURCE      DB10.DBX8.0
DBOFFSET    0
VAROFFSET   0
...
BE
```

OB100

```
UN M          10.0
S M           10.0
R M           10.1
CALL FC 20
IBDB          := DB20
COM_ADR       := 256
DIAG_STATE    := 0
DIAG_PARA     := 0
FKN_START     := 0
FKN_PARA      := 0
FKN_STATE     := 0
MEM_READ      := 21
MEM_WRITE     := 22
LOAD          := 0
```



```

BOOT           := 0
MODE           := 0
TIMER_NR      := T1
SOURCE        := 0
CONFIGURATION := DW#16#8000000F
RET           := M10.1
BUSY          := M10.0
UN M          100.7
S M           100.7
UN M          10.0
UN M          10.1
BE

```

OB1 für S7-400

OB1

```

...
Call FC 60
PIN          P#P10.0
DESTP#DB101.DBX0.0 Byte 8
SOURCEP#DB101.DBX8.0 Byte 8
POUT        P#P10.0
DBOFFSET    0
VAROFFSET   0
...

```

2.8.1.20 Beispiel für Betriebsart Asynchron mit Synchronisationsimpuls

OB1, OB40, OB100, FC50 und FC51 für S7-300

OB1

```

...
CALL FC 28
IBDB:=      DB20
READ_FC:=   FC50
WRITE_FC:=  FC51
CYCLE_OB:=  TRUE
RET:=       M14.0
...
BE

```

OB40

```

CALL FC 28
IBDB:=      DB20
READ_FC:=   FC50
WRITE_FC:=  FC51
CYCLE_OB:=  FALSE
RET:=       M12.0
BE

```

OB100

```

UN M          10.0
S M           10.0
R M           10.1
CALL FC 20
IBDB:=      DB20
COM_ADR:=   256
DIAG_STATE:= 0
DIAG_PARA:= 0
FKN_START:= 0
FKN_PARA:= 0
FKN_STATE:= 0
MEM_READ:=  21
MEM_WRITE:= 22
LOAD:=      0
BOOT:=      0

```

```
MODE:=          2
TIMER_NR:=      T1
SOURCE:=        0
CONFIGURATION:=DW#16#8000000F
RET:=          M10.1
BUSY:=         M10.0
UN M           10.0
UN M           10.1
BE
```

FC50

```
UN M           10.0
S M            10.0
CALL FC 21
IBDB:=         DB20
MODE:=         1
SOURCE:=       0
DEST_AREA:=   0
DEST_AREA_NR:= 0
DEST_OFFSET:= 0
DEST_LENGTH:= 0
RET:=         M10.1
BUSY:=        M10.0
CALL FC 65
PIN:=         EB10
POUT:=        AB10
DBOFFSET:=    0
VAROFFSET:=   0
BE
```

FC51

```
UN M           11.0
S M            11.0
CALL FC 22
IBDB           := DB20
MODE           := 1
SOURCE_AREA    := 0
SOURCE_AREA_NR := 0
SOURCE_OFFSET  := 0
SOURCE_LENGTH  := 0
DESTINATION    := 0
RET            := M11.1
BUSY           := M11.0
BE
```

A Index

B

Bestimmungsgemäßer Gebrauch 1-1

F

Fehlermeldungen

INTERBUS MMICOM, Schnittstelle 06, 14, 18,
21, 76 2-14

I

INTERBUS MMICOM, Schnittstelle 06, 14, 18, 21,
76 2-1

M

MMICOM

MMI-COM-Profil 2-1

O

Optimierte Datenübertragung 2-8

Randbedingungen 2-4

P

Protokollparameter

INTERBUS MMICOM, Schnittstelle 06, 14, 18,
21, 76 2-7

R

Randbedingungen für optimierte Datenübertragung
2-4

S

Sicherheitshinweise 1-1

Symbole 1-1

Syntaxdiagramm

INTERBUS MMICOM 2-9

W

Wichtige Hinweise 1-1

Z

Zielgruppe 1-1



