Inhaltsverzeichnis

5.15 TesiMod - InterBus 5.15-3

5.15.1 Technische Beschreibung 5.15-3
5.15.2 Spezifikation innerhalb des InterBus 5.15-5
5.15.3 MMI-COM-Profil 5.15-5
5.15.3.1 Allgemeine Struktur 5.15-6
5.15.3.2 Verwendete MMI-COM Standarddienste 5.15-7
5.15.3.3 Sütron spezifische Bit Schreibdienste 5.15-7
5.15.4 Aufgaben des SPS-Programms 5.15-7
5.15.4.1 Abwicklung des MMI-COM-Protokolls 5.15-8
5.15.4.2 Wahlfreier Datenzugriff 5.15-9
5.15.4.3 Struktur des MMI-Funktionsbausteins 5.15-10
5.15.4.4 Struktur des Lese-Funktionsbausteins 5.15-12
5.15.4.5 Struktur des Schreib-Funktionsbausteins 5.15-13
5.15.4.6 Vereinfachte Kommunikation über einen Datenbaustein 5.15-13
5.15.5 Parametrierung des Protokolls 5.15-15
5.15.5.1 Parameter MMI-COM Handshake Timeout 5.15-15
5.15.5.2 Parameter Verzögerung bis Verbindungsaufbau 5.15-15
5.15.5.3 Parameter Gleitkommazahlen im Siemens-Format 5.15-15
5.15.5.4 Datentypen 5.15-16
5.15.5.5 Variablennummer 5.15-16
5.15.6 Zusatzfunktionen 5.15-17
5.15.6.1 Pollbereich 5.15-17
5.15.6.2 Zustandsmeldungen 5.15-17
5.15.6.3 Siemens Timer/Zähler-Formate 5.15-18
5.15.7 Diagnose-LEDs 5.15-19
5.15.8 Physikalische Ankopplung 5.15-19
5.15.8.1 2 - Leiter Fernbuskabel 5.15-20
5.15.8.2 Umsetzung von 8 - Leiterprotokoll auf 2 - Leiterprotokoll 5.15-21
5.15.9 Fehlermeldungen 5.15-22

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

5.15 TesiMod - InterBus

5.15.1 Technische Beschreibung

TesiMod Bedienterminals lassen sich auf einfache Art und Weise im 2-Leiter-Fernbussystem an den InterBus ankoppeln. Deshalb sind TesiMod Bedienterminals als Mensch-Maschine-Schnittstelle für Anlagen mit InterBus-Bussystem bestens geeignet. Des weiteren können über den InterBus eine Vielzahl von Bedienterminals an eine Master-Steuerung gekoppelt werden.

Grundsätzlich ergeben sich durch den InterBus folgende Vorteile:

Bustopologie

Der InterBus verwendet als Bustopologie eine Ringstruktur mit aktiver Kopplung von max. 512 Busteilnehmern. Dabei sind die Hin- und Rückleitungen in einem einzigen Buskabel geführt. Jeder Fernbusteilnehmer stellt gleichzeitig einen Repeater dar, so daß insgesamt ca. 12,8 km überbrückbar sind.
Im Fehlerfall kann an jedem Fernbusteilnehmer der weiterführende Fernbus abgeschaltet und der Ring kurzgeschlossen werden. Das ermöglicht dem Busmaster eine gezielte Fehlerdiagnose bezüglich Fehlerart und Fehlerort, selbst bei einem Busleitungskurzschluß.

Schnelles Übertragungsmedium

Der InterBus wird mit RS485-Schnittstellenpegeln und einer Übertragungsrate von 500 kBit/s betrieben. Durch ein deterministisches Summenrahmenverfahren können äquidistante Zykluszeiten auf dem Busmedium garantiert werden. Es ergibt sich bei einer Gesamtrahmenbreite von 256 Bytes (2048 E/A-Punkte) ein max. InterBus-Zyklus von 7,2 Millisekunden.

Jedes Bedienterminal trägt mit 8 Byte Datenbreite auf dem InterBus mit ca. 0,4 Millisekunden zur Gesamtzykluszeit bei.

Integration in offenes Feldbuskonzept

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 Bedienterminals über denselben Bus eine Vielzahl von weiteren Teilnehmern an die Steuerung angekoppelt werden können (dezentrale Ein- und Ausgänge usw.).
Der InterBus liegt als Norm DIN 19258 vor.

Das TesiMod Bedienterminal wird als Slavegerät im 2-Leiter-Fernbus (9-polig SubminD) des InterBus integriert. Dabei belegt das Bedienterminal auf dem InterBus eine Datenbreite von 4 Worten oder 8 Bytes. Das hat zur Folge, daß in der Steuerung jedes Bedienterminal mit 8 Bytes-IN-Daten und 8 Byte OUT-Daten im Speicherabbild abgebildet wird.

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

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

Falls auf der Anschaltbaugruppe die Betriebsart "Logische Adressierung" gewählt wurde, so kann die Lage der Ein- und Ausgangsbereiche wahlfrei in den Speicher der Steuerung plaziert werden. Außerdem ist die Lage dieser Bereiche unabhängig von der Reihenfolge der Busteilnehmer.

Nach dem MMI-COM-Profil werden auf dem InterBus grundsätzlich drei Datenkanäle unterschieden:

Direkter Prozeßdatenkanal

Direkte Prozeßdaten sind zyklische Daten, die über die Betriebsdauer des Geräts konstant sind. Sie werden unquittiert und zyklisch in den Prozeßdatenkanal eingeblendet (zum Beispiel: 16 Eingänge eines Eingangsmoduls oder Bitinformationen eines Tastenfeldes - Geräteklasse A1 oder B1).

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.

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.

Das Bedienterminal wird auf dem InterBus im **indirekten Prozeßdatenkanal** betrieben. Es wird kein Parameterkanal und kein direkter Prozeßdatenkanal verwendet. Die Nutzdaten werden gemäß des MMI-COM Profils interpretiert.

Die Anschaltung vom Bedienterminal 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, muß mittels der Busklemme **IBS 24 BK/LC2** eine Umsetzung auf den 2-Leiter-Fernbus (9-polig) durchgeführt werden.

Sowohl die Bedienterminal-InterBus Anschaltung, als auch das MMI-COM-Protokoll, ist steuerungsunabhängig. Das hat zur Folge, daß innerhalb des MMI-COM-Protokolls nur Variablennummern, nicht aber reale SPS-Adressen, übertragen werden. Die Zuordnung der Variablennummer zur Variable innerhalb der Steuerung, muß **in der Steuerung** durchgeführt werden.

Grundsätzlich muß im Hauptprogramm der Steuerung ein Funktionsbaustein aufgerufen werden. Dieser Funktionsbaustein wird mit der Lage der IN- und OUT-Daten parametriert und muß für jedes

angeschlossene Bedienterminal aufgerufen werden. Bei Bedarf wertet er die vom Bedienterminal empfangenen Anforderungen aus. Dabei wird entweder ein Lese- oder ein Schreib-Funktionsbaustein aufgerufen, der die Zuordnung zwischen der Variablennummer und der realen SPS-Variable vornimmt. Das heißt die Daten werden in der SPS ereignisgesteuert bei Bedarf umkopiert. Dadurch hat die SPS auch eine Kontrolle darüber, welche Daten vom Bedienterminal geschrieben und gelesen werden.

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

5.15.2 Spezifikation innerhalb des InterBus

TesiMod Bedienterminals werden am InterBus mit folgenden Parametern betrieben:

Breite des Datenkanals:

indirekte Prozeßdaten mit Statuswort	: **4 Worte** / 8 Bytes
direkte Prozeßdaten	: keine
Parameterkanal	: keiner

Identcode:

0x2F(hex) / 47(dez) PD-Kanal mit Ein- und Ausgangsdaten

Funktionsgruppenspezifikation innerhalb des MMI-COM:

B3 Variableneingabe
G1 Variablenanforderung

Verwendete Dienste des MMI-COM-Profils:

PD-Index 0x14: Variable 1 Byte senden (mandatorisch)
PD-Index 0x15: Variable 2 Byte senden (mandatorisch)
PD-Index 0x16: Variable 4 Byte senden
PD-Index 0x40: Variable 1 Byte anfordern (mandatorisch)
PD-Index 0x41: Variable 2 Byte anfordern (mandatorisch)
PD-Index 0x42: Variable 4 Byte anfordern

Sütron spezifische Dienste:

PD-Index 0x84: Bitzugriff in Bytevariable senden
PD-Index 0x85: Bitzugriff in Wortvariable senden

5.15.3 MMI-COM-Profil

Das MMI-COM-Profil spezifiziert die einheitliche Interpretation der Nutzdaten auf dem InterBus für die Geräteklasse Bedienen und Beobachten (**M**an - **M**achine - **I**nterface - **COM**munication).

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

5.15.3.1 Allgemeine Struktur

Das MMI-COM-Profil definiert einen 8 Byte breiten Bereich innerhalb der indirekten Prozeßdaten mit folgendem Aufbau:

Byte 1 Status / SteuerByte	Byte 2 PD-Index	Byte 3 VariablenNr High	Byte 4 VariablenNr Low	Byte 5 Nutzdaten	Byte 6 Nutzdaten	Byte 7 Nutzdaten	Byte 8 Nutzdaten

Nutzdaten 1 Byte : b7 .... b0
Nutzdaten 2 Byte : b15 ..... b0
Nutzdaten 4 Byte : b31 ..... b0

Da der Datentransfer in beiden Richtungen durchgeführt werden muß, spiegelt sich jedes am InterBus angeschlossene Bedienterminal im Speicher der Steuerung mit 8 Bytes-IN-Daten und 8 Byte OUT-Daten wieder.

Belegung des Status / Steuer-Bytes:

Bit 7 : Onlinebit
Bit 6 : Störungsbit
Bit 5 : Handshake Empfangen
Bit 4 : Handshake Senden
Bit 3 : Standard-Bit (hat immer den Zustand 1)
Bit 2 : Index-Bit (hat immer den Zustand 1)
Bit 1 : reserviert
Bit 0 : reserviert

PD-Index:

Der PD-Index spezifiziert den jeweils verwendeten MMI-COM-Dienst (vgl. 5.15.3.2).

Variablennummer:

Gibt die Nummer der Variable an, auf die zugegriffen werden soll. Die Interpretation der Variablennummer ist vollständig frei und muß zwischen der Bedienterminal-Applikation und der SPS-Applikation abgestimmt werden.

Nutzdaten:

Gibt die Nutzdaten an, die für den Zugriff auf eine Variable verwendet werden sollen.

Byteorder

Das Bit 7 im Byte 5 stellt immer das MSB (**M**ost -**S**ignificant-**B**it) dar. Die Byte-Order ist bei aufsteigenden Adressen immer HIGH-->LOW.

Achtung :

Bei einigen Anschaltbaugruppen sind jeweils die Bytes paarweise vertauscht, damit per Wortzugriff aus dem SPS-Programm auf die Daten zugegriffen werden kann (Beispiel: Bosch-Anschaltbaugruppe).

Gleitkommadaten
Gleitkommadaten werden nach dem IEEE-Datenformat in den 4 Bytes Nutzdaten kodiert.

5.15.3.2 Verwendete MMI-COM Standarddienste

Schreibzugriff des Bedienterminals oder Antwort der Steuerung auf eine Anforderung:
PD-Index 0x14: Variable senden (1 Byte)
PD-Index 0x15: Variable senden (2 Byte)
PD-Index 0x16: Variable senden (4 Byte)

Leseanforderung des Bedienterminals:
PD-Index 0x40: Variable anfordern (1 Byte)
PD-Index 0x41: Variable anfordern (2 Byte)
PD-Index 0x42: Variable anfordern (4 Byte)

5.15.3.3 Sütron spezifische Bit Schreibdienste

Das Schreiben von Bits ist kein Standarddienst des MMI-COM-Profils. Sütron hat daher einen eigenen Bit Schreibdienst definiert. Es wird ein NON-Standarddienst generiert und mit folgenden PD-Indexen gekoppelt. Deshalb wird bei diesen Diensten im Steuerbyte das Standardbit gelöscht.

PD-Index 0x84: Bitzugriff in Bytevariable senden (1 Byte)
PD-Index 0x85: Bitzugriff in Wortvariable senden (2 Byte)

Steuerbyte	PD-Index	Variable	Variable	Byte 5 High	Byte 6	Byte 7 Low	Byte 8
XX	0x84	XXXX	XXXX	And Mask		Or Mask	
XX	0x85	XXXX	XXXX	And Mask High	And Mask Low	Or Mask High	Or Mask Low

Die Steuerung quittiert diese Dienste in Ihrem Handshake empfangen Bit und mit gesetztem Standard Bit.
Die Steuerung muß zunächst die entsprechende Variable auslesen, die gelesenen Daten mit der AND-Maske verknüpfen, dann die OR-Maske mit dem Ergebnis verknüpfen und die Daten wieder zurückspeichern.
Beispiele:
1.) Aus einem Datenwort soll das niederwertigste Bit auf den Wert 1 gesetzt werden
AND Maske 0xFE
OR Maske 0x01
2.) Aus einem Datenwort soll das niederwertigste Bit auf den Wert 0 gesetzt werden
AND Maske 0xFE
OR Maske 0x00

5.15.4 Aufgaben des SPS-Programms

Das SPS-Programm muß bezüglich der Bedienterminalfunktion die Funktionen und Dienste des MMI-COM-Profils abhandeln. Dazu muß für jedes angeschlossene Bedienterminal ein Funktionsbaustein

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

aufgerufen werden. Dieser Funktionsbaustein interpretiert die eingetroffenen Daten, verwaltet die Handshakebits und dekodiert die Anforderungen des Bedienterminals.

Da bei der Anschaltung von mehreren Bedienterminals die Nutzdaten auf unterschiedlichen Speicherbereichen der Steuerung liegen, muß der Funktionsbaustein bezüglich der Adreßlage der IN- bzw. OUT-Daten parametrierbar sein. Falls die OUT-Daten nicht durch das SPS-Programm rücklesbar sind, muß
der aktuelle Zustand der OUT-Daten (nur Status/Steuerbyte und PD-Index) in einem Zwischenpuffer gespeichert werden. (So sind zum Beispiel die OUT-Daten im P-Bereich einer Siemens-Steuerung nicht rücklesbar).

Falls gültige Anforderungen des Bedienterminals erkannt wurden, muß durch Aufruf eines Schreib- oder Lese-Funktionsbausteins die übertragene Variablennummer der realen SPS-Adresse zugeordnet und die erforderlichen Daten umkopiert werden. Das hat den Vorteil, daß ereignisgesteuert in der SPS nur diejenigen Daten verwaltet werden müssen, die das Bedienterminal aktuell gerade benötigt. Zusätzlich hat die SPS die Kontrolle darüber, auf welche Daten durch das Bedienterminal zur Laufzeit zugegriffen wird.

Wenn das Bedienterminal das Online-Bit wegnimmt, müssen in der Steuerung die Handshakebits gelöscht werden. Dadurch kann sich das Bedienterminal zu jedem beliebigen Zeitpunkt auf die Steuerung synchronisieren.

5.15.4.1 Abwicklung des MMI-COM-Protokolls

Onlinebit (Bit 7 des Status / Steuerbyte)

Wenn der Bus ordnungsgemäß arbeitet, und die Steuerung aktiv ist, muß das Online-Bit gesetzt werden. Das Online-Bit zeigt die Empfangsbereitschaft des Geräts an. Im Normalzustand hat das Bit den Wert 1.

Störungsbit (Bit 6 des Status / Steuerbyte)

Dieses Bit zeigt an, daß auf dem Gerät eine Störung vorliegt. Im Normalzustand hat das Bit den Wert 0.

Handshake-Empfangen-Bit (Bit 5 des Status / Steuerbyte)

Mit diesem Bit wird der Empfang eines Telegramms quittiert. Dieses Bit wird immer dann invertiert, wenn die Gegenstation durch Invertieren ihres Handshake-Senden-Bits (Bit 4) ein neues Telegramm anzeigt und dieses von der empfangenden Station ausgewertet wurde.

Mit diesem Bit wird lediglich auf Protokollebene angezeigt, daß der betreffende Übertragungskanal wieder frei ist, nicht jedoch, daß die betreffende Anwendung das Telegramm bereits umgesetzt hat. Das Telegramm kann also von der betreffenden Anwendung zu einem späteren Zeitpunkt noch als fehlerhaft eingestuft werden.

Handshake-Senden-Bit (Bit 4 des Status / Steuerbyte)

Mit diesem Bit wird ein neues Telegramm signalisiert. Dieses Bit darf nur dann invertiert werden, wenn es mit dem Handshake Empfangen-Bit (Bit 5) der Gegenstation übereinstimmt und wenn

die Gegenstation ONLINE ist (Bit 7). Der Übertragungskanal ist dann solange gesperrt, bis die Gegenstation ihr Handshake-Empfangen-Bit mit dem neuen Wert des eigenen Handshake-Senden-Bits wieder in Übereinstimmung bringt.

Standard- und Indexbit (Bits 3 / 2 des Status / Steuerbyte)
Diese beiden Bits zeigen an, daß es sich um Standardfunktionen und nicht um herstellerspezifische Funktion handelt. Diese beiden Bits haben immer den Zustand 1.

Reservierte Bits (Bits 1 / 0 des Status / Steuerbyte)
Die reservierten Bits sind mit dem Zustand 0 zu beaufschlagen.

Anforderungen des Bedienterminals :
Sämtliche für den Betrieb des Bedienterminals erforderlichen Dienste werden vom Bedienterminal initiiert.

Wenn das Handshake-Senden-Bit des Bedienterminals (IN-Daten-Bereich) mit dem Handshake-Empfangen-Bit der Steuerung (OUT-Daten-Bereich) nicht übereinstimmt, so ist eine neue Anforderung des Bedienterminals eingetroffen, die durch die SPS bearbeitet werden muß. Dabei muß der Empfang der Anforderung durch Invertieren des Handshake-Empfangen-Bits (OUT-Daten-Bereich) in der SPS quittiert werden und evtl. Daten zurück an das Bedienterminal übertragen werden. Diese neuen Sendedaten sind dann im Sendedatenbereich (OUT-Bereich) der Steuerung einzutragen und zusätzlich das Handshake-Senden-Bit zu invertieren (OUT-Bereich).

Lesezugriff des Bedienterminals (Dienst Variable anfordern)

Bedienterminal		SPS
---- Variable anfordern	------->	
<--- Variable senden	--------	

Schreibzugriff des Bedienterminals (Dienst Variable senden)

Bedienterminal		SPS
---- Variable senden	------->	

5.15.4.2 Wahlfreier Datenzugriff

Wenn eine Anforderung des Bedienterminals erkannt wurde, müssen Programmteile (Funktionsbausteine) aufgerufen werden, die die Lese- oder Schreibfunktion realisieren. Innerhalb dieser Funktionsbausteine wird eine Zuordnung zwischen der übertragenen Variablennummer und der realen SPS-Adresse vorgenommen. Als Aufrufparameter sind hier der PD-Index, die Variablennummer und die Nutzdaten zu sehen. Beim Schreibzugriff des Bedienterminals müssen die Nutzdaten (IN-Bereich)

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

auf die SPS-Variablen geschrieben werden, beim Lesezugriff die Nutzdaten aus dem SPS-Speicher geholt und in den Sendebereich (OUT-Bereich) der Steuerung geschrieben werden.
In Abhängigkeit des in der Applikation verwendeten Zugriffs (vgl 5.15.5.4) müssen 1-, 2- und 4 Bytezugriffe auf die Variablen unterstützt werden.

Es können geeignete Variablennummernbereiche zu logischen Bereichen zusammengefaßt werden. Dadurch vereinfacht sich der Decodieraufwand innerhalb der Schreib- und Lese-Funktionsbausteine. (Beispielsweise die Variablennummern 100 bis 199 könnten den Merkerworten 0 bis 99 entsprechen)

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 (vgl 5.15.5.4).

Beim Hochlauf des Bedienterminals wird in einer Testkommunikation ein 1 Byte Lesezugriff auf die Variable mit der Nummer 0 durchgeführt. Dabei werden die gelesen Daten nicht interpretiert, das heißt der Lesefunktionsbaustein muß diese Variablennummer nicht notwendigerweise decodieren.

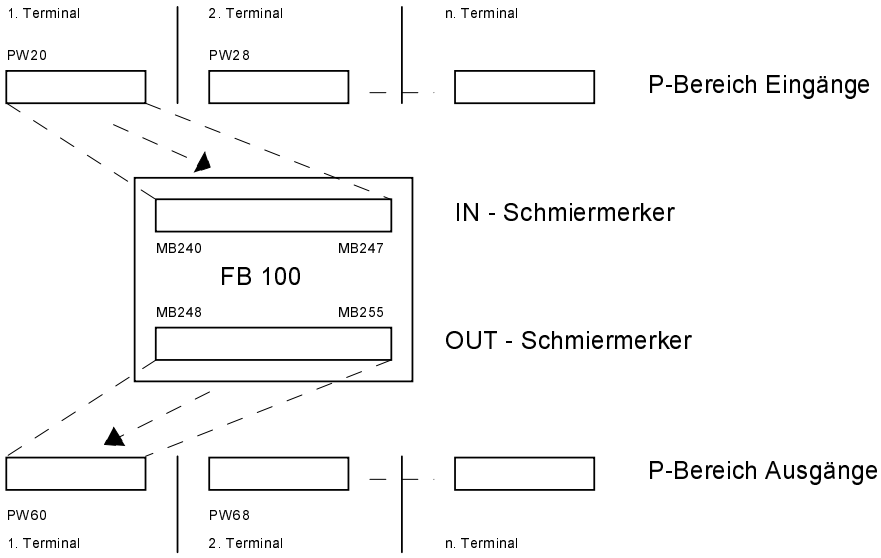
Alle editierbaren Variablen müssen sowohl in dem Schreib- als auch in dem Lese-Funktionsbaustein decodiert werden, da sie im allgemeinen zuerst gelesen und dann auch geschrieben werden.

5.15.4.3 Struktur des MMI-Funktionsbausteins

Die Struktur des MMI-Funktionsbaustein wird hier am Fallbeispiel der InterBus-Anschaltung an eine Siemens-SPS mit dem FB100 erläutert. Bei der Anschaltung an andere Steuerungen hat der Funktionsbaustein eine ähnliche Struktur.

Der Funktionsbaustein FB100 hat folgende Aufgaben :

- wird für jedes im Netz befindliche Bedienterminal im OB1 aufgerufen
- wird mit der Adreßlage der Daten im P-Bereich der Steuerung parametriert (Parameter PEIN und PAUS)
- verwaltet die Handshakebits und den PD-Index
- sichert das 1. Wort der OUT-Daten in einem Datenbaustein (Parameter DBNR und WDNR)
- kopiert die IN-/OUT-Peripheriedaten in IN-/OUT-Schmiermerker MB240 bis MB255 um
- verzweigt bei Bedarf in einen Schreib- / Lese-Funktionsbaustein



MW238 wird für die Bearbeite-Funktion verwendet !

```
    BAUSTEIN#OB1
    BIB     #15033
00000     :
00002     :
00004     :SPA FB 100    1. Bedienterminal
    NAME #FB100
    PEIN =KF +20         Adresse IN-Daten
    PAUS =KF +60              Adresse OUT-Daten
    DBNR =KF +100             Datenbausteinnummer
    WDNR =KF +0               Wortnummer
00010     :
00012     :SPA FB 100    2. Bedienterminal
    NAME #FB100
    PEIN =KF +28         Adresse IN-Daten
    PAUS =KF +68              Adresse OUT-Daten
    DBNR =KF +100             Datenbausteinnummer
    WDNR =KF +1               Wortnummer
0001E     :BE
```

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

5.15.4.4 Struktur des Lese-Funktionsbausteins

Die Struktur des Lese-Funktionsbaustein wird hier am Fallbeispiel der InterBus-Anschaltung an eine Siemens-SPS mit dem FB101 erläutert. Bei der Anschaltung an andere Steuerungen hat der Funktionsbaustein eine ähnliche Struktur.

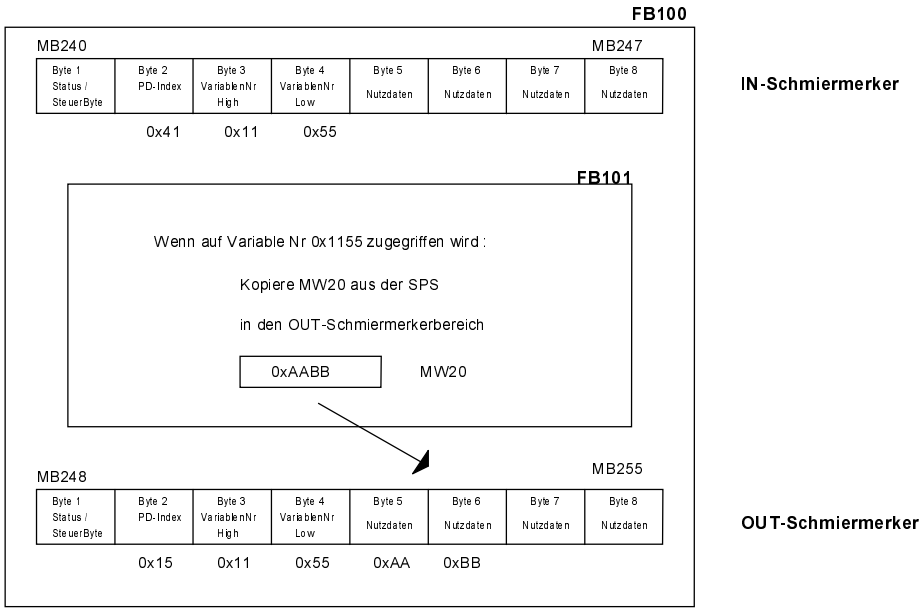
Der Funktionsbaustein FB101 hat folgende Aufgaben :

- realisiert den ereignisgesteuerten Lesezugriff
- wertet PD-Index, Variablennummer aus
- kopiert Nutz-Daten aus dem Speicher der SPS in einen Schmiermerkerbereich

Beispiel :

Variablendatei : SOLLWERT W 4437

Variablennummer : 0x1155 oder 4437 (dez)
Lage in der SPS : MW 20 mit dem Inhalt : 0xAABB oder 43707 (dez)
PD - Index : 0x41 (2 Bytes anfordern) /
0x15 (2 Bytes senden)



5.15.4.5 Struktur des Schreib-Funktionsbausteins

Die Struktur des Schreib-Funktionsbaustein wird hier am Fallbeispiel der InterBus-Anschaltung an eine Siemens-SPS mit dem FB102 erläutert. Bei der Anschaltung an andere Steuerungen hat der Funktionsbaustein eine ähnliche Struktur.

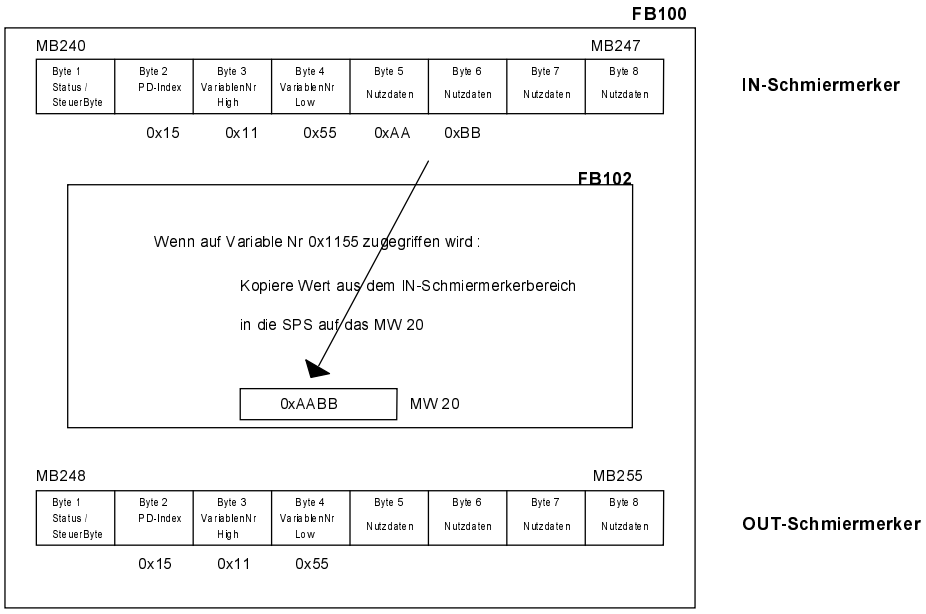
Der Funktionsbaustein FB102 hat folgende Aufgaben :

- realisiert den ereignisgesteuerten Schreibzugriff
- wertet PD-Index, Variablennummer und Nutzdaten aus
- kopiert Nutz-Daten aus dem Schmiermerkerbereich in den Speicher der SPS

Beispiel :

Variablendatei : SOLLWERT W 4437

Variablennummer : 0x1155 oder 4437 (dez)
Lage in der SPS : MW 20
Wert : 0xAABB oder 43707 (dez)
PD - Index : 0x15 (2 Bytes senden)



5.15.4.6 Vereinfachte Kommunikation über einen Datenbaustein

Der Decodieraufwand innerhalb der Schreib- und Lesefunktionsbausteine kann wesentlich vereinfacht werden, wenn nur über einen Datenbaustein kommuniziert wird.

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

Dazu wird die übertragene Variablennummer direkt als Wortnummer innerhalb eines Datenbausteins interpretiert. In diesem Fall werden nur Wortzugriffe durchgeführt. Die Funktionsbausteine (für das Fallbeispiel Siemens-Anschaltung FB101 und FB102) sind dann applikationsunabhängig. Der Anwender hat die volle Kontrolle über die Variablen in der Projektierumgebung des Bedienterminals.

In diesem Fall sehen die Funktionsbausteine FB101 und FB102 wie folgt aus:

```
    BAUSTEIN#FB101
    BSTNAME #FB101
    BIB     #4010
0000A       :                       ---------------------------------
0000C       :                       Dieser Baustein liest Variablen
0000E       :                       - - - - - - - - - - - - - - - -
00010       :                       Übergabe der Daten in folgenden
00012       :                       Schmiermerkern
00014       :
00016       :                       Aufrufparameter :
00018       :                       PD-Index            : MB 241
0001A       :                       Variablennummer  : MW 242
0001C       :
0001E       :                       Returnparameter :
00020       :                       Nutzdaten           : MB252 .. MB255
00022       :                       ---------------------------------
00024       :
00026       :A   DB XYZ             Datenbaustein aktivieren
00028       :B   MW 242
0002A       :L   DW 0               Wort aus Datenbaustein lesen
0002C       :T   MW 252             in den Ziel-Schmiermerkerbereich
0002E       :                       transferieren
00030       :BE
```

```
    BAUSTEIN#FB102
    BSTNAME #FB102
    BIB     #4010
0000A       :                       ------------------------------------------
0000C       :                       Dieser Baustein schreibt Variablen
0000E       :                       - - - - - - - - - - - - - - - -
00010       :                       Übergabe der Daten in folgenden
00012       :                       Schmiermerkern
00014       :
00016       :                       Aufrufparameter :
00018       :                       PD-Index       : MB 241
0001A       :                       Variablennummer : MW 242
0001C       :                       Nutzdaten      : MB244 .. MB247
```

```
0001E     :                  ------------------------------------------
00020     :
00022     :A   DB XYZ
00024     :L   MW 244        Nutzdaten in Akku laden
00026     :B   MW 242
00028     :T   DW 0          In den Datenbaustein schreiben
0002A     :
0002C     :BE
```

5.15.5 Parametrierung des Protokolls

5.15.5.1 Parameter MMI-COM Handshake Timeout

Spezifiziert die Wartezeit in Millisekunden, die das Bedienterminal auf die Quittung der SPS wartet (Änderung in den Handshakebits). Falls in dieser Zeit keine Quittung durch die SPS vorgenommen wird, geht das Bedienterminal in den Kommunikationsfehler (vgl. 5.15.9)

Der zulässige Wertebereich ist : 0 bis 65000 Millisekunden.

5.15.5.2 Parameter Verzögerung bis Verbindungsaufbau

Mit dieser Verzögerungszeit (bezüglich der Einheit Sekunden) wird das erstmalige Absetzen von MMI-COM-Diensten verzögert. Dadurch wird vermieden, daß das Bedienterminal in der Hochlaufphase einen Kommunikationsfehler entdeckt, der darin begründet ist, daß der Bus und / oder die Steuerung noch nicht betriebsbereit ist. Wenn die SPS und die entsprechende Anschaltbaugruppe für den Power-Up-Anlauf länger als das Bedienterminal benötigen, kann dies durch eine erhöhte Verzögerungszeit kompensiert werden.

Der zulässige Wertebereich ist : 1 bis 20 Sekunden.

5.15.5.3 Parameter Gleitkommazahlen im Siemens-Format

Mit dieser Option wird festgelegt, ob die Gleitkommadaten im IEEE-Format oder im Siemens-Spezialformat an die Steuerung übertragen werden.

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

5.15.5.4 Datentypen

Die Datentypen definieren, wieviele Bytes einer Einzelvariablen zugeordnet werden :

Datentyp	Zugriff auf	Länge Einzelvariable	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	8 Bytes	nächstes Doppelwort steht 4 Adressen höher

Die Datentypen W-2, DW-2 und DW-4 stehen erst ab der Firmware-Version 6.0 zur Verfügung.

Der Zugriffsparameter bestimmt zum einen die Datenlänge, die vom Bedienterminal ausgewertet wird. Ferner definiert dieser Parameter, mit welchem MMI-COM-Dienst die Variable abgehandelt wird.

Achtung :
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.

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

5.15.5.5 Variablennummer

Über das MMI-COM-Profil werden die Daten steuerungsunabhängig über eine Variablennummer zugeordnet. Das heißt erst auf der Steuerung wird einer Variablennummer eine harte Steuerungsadresse zugeordnet (zum Beispiel soll der Variablennummer 50 der Merker 32.0 zugeordnet werden). Die Variablennummer kann entweder dezimal (ohne Kennung) oder hexadezimal (mit nachgestelltem H) eingegeben werden (hexadezimale Eingabe erst ab Treiberversion 4.5).

Beispiel: Variablennummer 100 oder 64H

5.15.6 Zusatzfunktionen

Da innerhalb des MMI-COM-Profils nur Zugriffe bis max. 4 Bytes unterstützt werden, müssen folgende Zusatzfunktionen durch mehrfache Zugriffe realisiert werden.

Um eine definierte Byteorder zu erzielen, müssen die folgenden Zusatzfunktionen immer mit 2 Byte - Zugriffen realisiert werden. Das heißt bei der Zuordnung der Start-Variablennummer muß immer der Zugriff W (Wort) angewählt werden.

5.15.6.1 Pollbereich

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

Aufbau mit 2 Byte - Variablen

15 8	7 0	Bitnummer
KBS	Reserviert	Variable X
Meldekanal Highbyte	Meldekanal Lowbyte	Variable X+1
LEDs 1 bis 4	LEDs 5 bis 8	Variable X+2
LEDs 9 bis 12	LEDs 13 bis 16	Variable X+3
LEDs 17 bis 20	LEDs 21 bis 24	Variable X+4
LEDs 25 bis 28	LEDs 29 bis 32	Variable X+5

5.15.6.2 Zustandsmeldungen

Unter Zustandsmeldungen versteht man die statische Zuordnung von Flags (Bits) in der Steuerung auf Klartextmeldungen im Bedienterminal. Nachfolgend wird der Aufbau der Zustandsmeldungen beschrieben.

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

Der Bereich der Zustandsmeldungen hat folgende Struktur:

Aufbau der 2 Byte - Variablen

15 8	7 0	Bitnummer
Meldung 16 bis 9	Meldung 8 bis 1	Variable X
Meldung 32 bis 25	Meldung 24 bis 17	Variable X+1
Meldung usw.	Meldung usw.	Variable X+2

5.15.6.3 Siemens Timer/Zähler-Formate

Format Siemens Zähler
Falls als Darstellungsart Zählerformat angewählt wurde, wird binär auf die Daten zugegriffen. Evtl. vorhandene Statusbits des Timers werden beim Lesen ausgeblendet bzw. beim Schreiben überschrieben.

Format Siemens Timer
Falls als Darstellungsart Timerformat angewählt wurde, wird beim Lesen aus dem eigentlichen Timerwert und dem Bereich eine terminalinterne 4 Byte Zahl ohne Vorzeichen gebildet, die den Timerwert bezüglich der Basis von 0,01 Sekunden darstellt.

Beispiel: Bei Range 10 (1,0 Sekunden Auflösung) und Timerwert 999 wird terminalintern der Wert 99900 dargestellt bzw. editiert.

Beim Schreiben von Timervariablen wird aus der terminalinternen vorzeichenlosen 4 Byte Zahl der Timerwert und die kleinstmögliche Bereichseinstellung gebildet.

5.15.7 Diagnose-LEDs

Diagnose-LED +5V (Betriebsspannung)
Die grüne LED +5V zeigt die interne Versorgungsspannung der InterBus-Logik an.

Diagnose-LED RC (**R**emotebus **C**heck)
Die grüne LED RC stellt eine Überwachung des ankommenden Fernbus dar. Ist die Kabelverbindung in Ordnung und ist der InterBus-Master (Anschaltbaugruppe der Steuerung) nicht im Reset, so ist RC aktiv. Bei einem InterBus-Reset oder Power-Up-Reset wird die LED RC inaktiv.

Diagnose-LED BA (**B**us-**A**ktiv)
Die grüne LED BA signalisiert eine InterBus Übertragung. Die LED hat eine Ausschaltverzögerung von 640 ms.

Diagnose-LED TR (**T**ransmit/**R**eceive)
Die grüne LED TR wird aktiv, wenn dieser Teilnehmer PCP-Kommunikation betreibt. Da das Bedienterminal innerhalb des MMI-COM-Profils kein PCP über den Parameterkanal betreibt, bleibt diese LED inaktiv.

Diagnose-LED RD (**R**emotebus **D**isable)
Die rote LED RD zeigt die Abschaltung des weiterführenden Fernbus an. Entweder wurde ein Reset durchgeführt oder der Master versucht innerhalb des ID-Zyklus die weiterführende Schnittstelle abzuschalten.

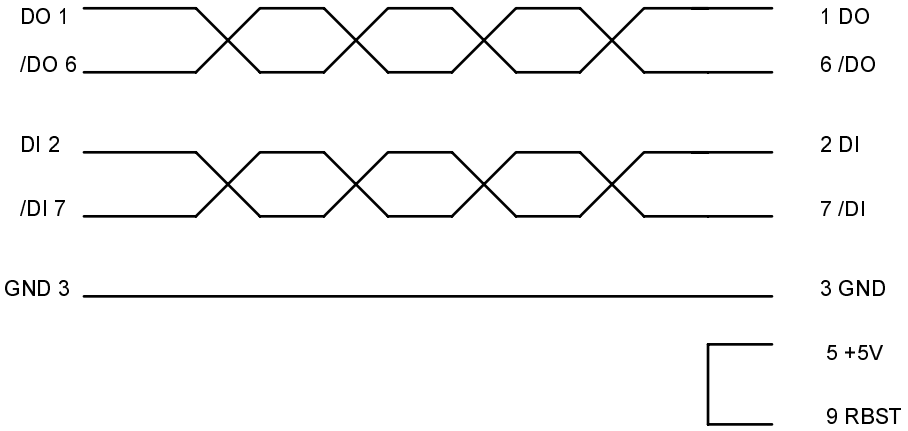
5.15.8 Physikalische Ankopplung

Steckverbindungen am TesiMod Bedienterminal für Anschluß an Fernbus

	Remotebus In	**Remotebus Out**
1	DO	DO
2	DI	DI
3	GND	GND
4	-	-
5	-	+5V
6	/DO	/DO
7	/DI	/DI
8	-	-
9	-	RBST

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

5.15.8.1 2 - Leiter Fernbuskabel



D-SUB 9 pol
Buchsen

Remotebus-In

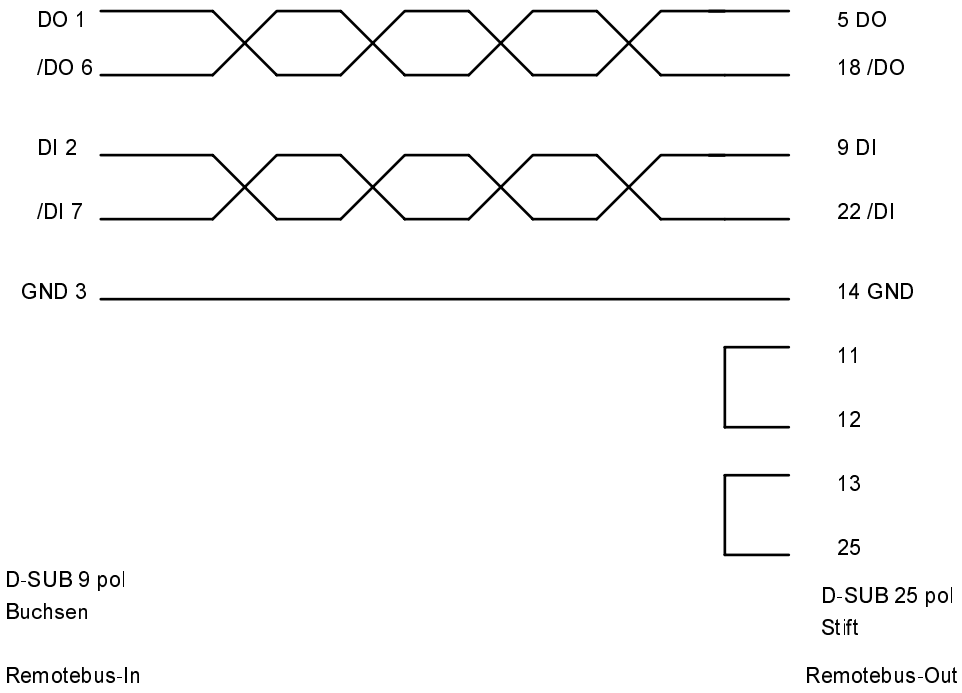
D-SUB 9 pol
Stift

Remotebus-Out

LI-YCY- 3 x 2 x 0.25 mm²
Adern paarweise verseilt
gemeinsamer Schirm beidseitig auf Steckergehäuse

5.15.8.2 Umsetzung von 8 - Leiterprotokoll auf 2 - Leiterprotokoll

Mit Busklemme IBS 24 BK/LC2 :



LI-YCY- 3 x 2 x 0.25 mm²
Adern paarweise verseilt
gemeinsamer Schirm beidseitig auf Steckergehäuse

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

5.15.9 Fehlermeldungen

Mögliche Fehlermeldungen :

Code	2	E_PROTOKOL	Die Steuerung hat Daten an das Bedienterminal geschickt, die nicht vom Bedienterminal angefordert wurden. Der Subcode gibt den empfangenen PD-Index dezimal an.
	40	E_SYS_ADDRESS	Falsche Systemvariable.
	50	E_STOERUNG	Steuerung hat als Gegenstation das Störungsbit gesetzt. Der Subcode gibt das empfangene Status-Steuerbyte dezimal an.(**)
	51	E_OFFLINE	Steuerung ist als Gegenstation Offline. Der Subcode gibt das empfangene Status-Steuerbyte dezimal an.(**)
	52	E_NON_STANDARD	Steuerung hat einen Dienst übertragen, bei dem das Index-Bit und/oder das Standard-Bit nicht gesetzt war. Der Subcode gibt das empfangene Status-Steuerbyte dezimal an.(**)
	53	E_ILLEGAL_PDINDEX	Es wurde ein nicht unterstützter PD-Index empfangen. Der Subcode gibt den empfangenen PD-Index dezimal an.(**)
	54	E_TIMEOUT_HANDSHAKEBITS	Die Steuerung hat als Gegenstation nicht innerhalb der Timeoutzeit durch toggeln der Handshakebits reagiert.

Mögliche Fehlerursachen:
- OB13 geladen, zeigt auf Schmiermerker
- OB1 mit falscher PAA- und PAE-Adresse
- Daten überschreiben Sicherungswort der PAA-Daten (Steuer-Statusbyte, parametriert zum Beispiel als OB1, DW0, DB100)

Mögliche Fehlerursachen:

- Entweder läuft der InterBus nicht, die SPS ist im Stop-Mode oder der für den Betrieb des Bedienterminals notwendige Funktionsbaustein ist in der SPS nicht aktiv. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.

55 E_TIMEOUT_ANSWER ..... Die Steuerung hat als Gegenstation nicht innerhalb der Timeoutzeit einen Variablen-Anforderungs-Dienst mit einem Variablen-Senden-Dienst beantwortet. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.

56 E_BASESIZE ..... Unzulässiger Zugriffstyp und damit unzulässige Basisdatengröße in der Variablenliste. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.

57 E_HANDSHAKEBITS ..... Die Handshakebits der Gegenstation waren so gestellt, daß das Bedienterminal keine weiteren Anforderungen absetzen konnte. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.

58 E_SUPI ..... Der SUPI Protokollchip ist nicht mehr ansprechbar.

Mögliche Fehlerursachen:

- Evtl. werden auf dem Bus keine gültigen Datenzyklen abgewickelt. Der Subcode gibt die Variablennummer dezimal an, bei deren Bearbeitung der Fehler aufgetreten ist.

/006-8099/ Technische Änderungen vorbehalten
TESI_T_ger_V20.300515FQK0

59 E_WRONG_VARNR ..... Die SPS hat bei einem Leseauftrag eine falsche Variablennummer zurückübertragen.

Mögliche Fehlerursachen:
- Evtl. sind die 8 Bytes Nutzdaten nicht konsistent. Der Subcode gibt die empfangene Variablennummer dezimal an.(**)

60 E_WRONG_PDINDEX ..... Die SPS hat bei einem Leseauftrag einen falschen PD-Index zurückübertragen.

Mögliche Fehlerursachen:
- Evtl. sind die 8 Bytes Nutzdaten nicht konsistent. Der Subcode gibt den empfangenen PD-Index dezimal an.(**)

(**) Subcode erst ab Treiberversion 4.3