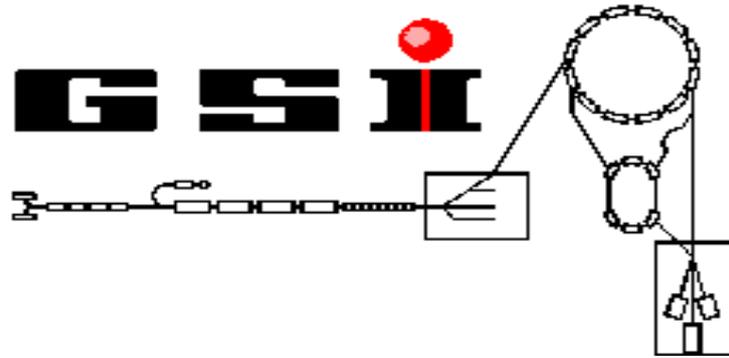


Common Systems
Control Systems
Hardware

CSCOHW



Interface-Karte für
die Netzgeräte-Backplane,
den Modul- und SCU-Bus.

FG 380.221

R. Hartmann

Änderungsliste

| Vers. | Datum | Name | Kommentar |
|--------------|--------------|-------------|---|
| 0.1 | 23.02.2009 | R. Hartmann | Dokument erstellt |
| 0.2 | 05.10.2011 | R. Hartmann | FKTC-Liste: wegen FG-Trigger(FG 380.751) und SCU-Bus |
| 0.3 | 08.02.2013 | R. Hartmann | Betriebsart SCU-ACU, Daten zur VG-Leiste bei Power-UP = "0" |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Inhaltsverzeichnis:

| | | |
|----------|---|-----------|
| 1 | BESCHREIBUNG..... | 5 |
| 1.1 | Allgemeines | 5 |
| 1.2 | Kompatibilität..... | 5 |
| 1.3 | Blockschaltbild | 6 |
| 1.4 | Funktionsbeschreibung..... | 7 |
| 1.5 | Betriebsarten..... | 8 |
| 1.5.1 | NG-Backplane..... | 8 |
| 1.5.2 | SD-Backplane (Blockmode) | 8 |
| 1.5.2.1 | Lesen im Blockmode | 8 |
| 1.5.3 | Modul-Bus-Mode..... | 10 |
| 1.5.4 | SCU-Bus-Mode..... | 11 |
| 1.5.5 | SCU-ACU-Mode (V0.3)..... | 12 |
| 1.5.6 | Funktionsgenerator-Mode..... | 12 |
| 1.5.7 | Funktionsgenerator-Mode mit DDS-Interface | 12 |
| 1.5.8 | Sweeper..... | 12 |
| 1.6 | Technische Daten: (Kurzdaten)..... | 13 |
| 1.7 | Front- und Seitenansicht der Baugruppe | 14 |
| 2 | DEVICE-BUS..... | 15 |
| 2.1 | Schreibe Daten mit Funktions-Code..... | 16 |
| 2.1.1 | Schreibe Daten | 16 |
| 2.1.2 | Schreibe Funktions-Code..... | 16 |
| 2.1.3 | Timing auf dem Device-Bus, Ausgang des Manchester-Decoders..... | 16 |
| 2.2 | Lese Daten oder Status | 17 |
| 2.2.1 | Timing beim Lesen auf dem Device-Bus und am Manchester-Encoder..... | 17 |
| 3 | LISTE DER FUNKTIONS-CODES | 18 |
| 4 | REGISTER | 19 |
| 4.1 | Steuerregister (Funktions-Codes siehe Kap. 3)..... | 19 |
| 4.1.1 | Mode-Register..... | 19 |
| 4.1.2 | Interrupt-Maske..... | 19 |
| 4.1.3 | Echo-Register..... | 20 |
| 4.1.4 | I2C-Bus | 20 |
| 4.2 | Status-Register (Funktions-Codes siehe Kap. 3)..... | 22 |
| 4.2.1 | Interlock-Status-Register..... | 22 |
| 4.2.2 | IFA-Status-Register..... | 22 |
| 4.2.3 | Modul-Bus-Status-Register | 22 |
| 4.2.4 | ID-Register..... | 23 |
| 4.2.5 | EPLD-Versions-Register..... | 23 |
| 5 | MODUL-BUS..... | 24 |
| 5.1 | Modul-Bus-Adressierung | 24 |
| 5.1.1 | Standard-Modul-Bus-Zugriff (2 Schritte)..... | 24 |
| 5.1.2 | Schneller Modul-Bus-Zugriff (1 Schritt) | 24 |
| 5.1.2.1 | Initialisierung des Funktions-Code-Decoders | 24 |
| 5.1.2.2 | Initialisierung des MB-Group-Count | 24 |

| | | |
|----------|--|-----------|
| 6 | SCU-BUS | 25 |
| 6.1 | SCU-Bus-Mode-Adressierung..... | 25 |
| 6.1.1 | Standard-SCU-Bus-Zugriff (2 Schritte)..... | 25 |
| 6.1.2 | Schneller SCU-Bus-Zugriff (1 Schritt)..... | 25 |
| 6.1.2.1 | Initialisierung des Funktions-Code-Decoder | 25 |
| 6.1.2.2 | Initialisierung des SCU-Bus-Group-Cont | 25 |
| 6.2 | SCU-ACU-Mode-Adressierung (V0.3)..... | 26 |
| 6.2.1 | Standard-SCU-ACU-Bus-Zugriff)..... | 26 |
| | | |
| 7 | MANUALS | 26 |
| 7.1 | Modul-Bus | 26 |
| 7.2 | SCU-Bus | 26 |
| 7.3 | Funktionsgenerator | 26 |
| | | |
| 8 | ANHANG | 27 |
| 8.1 | Pinbelegung der VG-Steckerleiste | 27 |
| 8.2 | Signalbeschreibung: | 28 |
| 8.3 | Netzgeräte-Backplane..... | 29 |
| 8.4 | Modul-Bus-Backplane..... | 29 |
| 8.5 | SCU-Bus-Backplane..... | 29 |
| 8.6 | Produktionsunterlagen..... | 30 |

1 Beschreibung

1.1 Allgemeines

Die Geräte am SIS und ESR (z.T. auch UNILAC) werden von Steuereinheiten ("SEs") kontrolliert. Die SEs sind über den Device-Bus mit den Geräten verbunden. Da die Geräte nicht direkt an den Device-Bus angeschlossen werden können, werden Anpass-Elektroniken, so genannte Interface-Karten benötigt. Die Interface-Karten setzen die Daten vom Device-Bus in gerätespezifische Signale um.

Die Interface-Karten werden von der SE-Software über den Device-Bus gesteuert.

Die Adresse der Interface-Karte wird über die VG-Leiste eingestellt.

Die allgemeine digitale Interface-Karte FG 380.221 (im weiteren **IFA** genannt) ist für den universellen Einsatz entwickelt worden. Sie kann als Interface für die Netzgeräte-Backplane, als Controller für den Modul-Bus oder als "stand-alone"- Interface-Karte mit Piggy-Back-Platine verwendet werden.

Es ist ebenfalls ein Funktionsgenerator sowie ein I2C-Bus Controller integriert.

Die unterschiedlichen Betriebsarten der **IFA** können nur zum Teil per SE-Software eingestellt werden, einige Betriebsarten müssen aber wegen der Sicherstellung der richtigen Treiberrichtung zwingend per Hardware über die VG-Leiste bzw. einen Piggy-Stecker-Pin festgelegt werden.

Achtung! (V0.3)

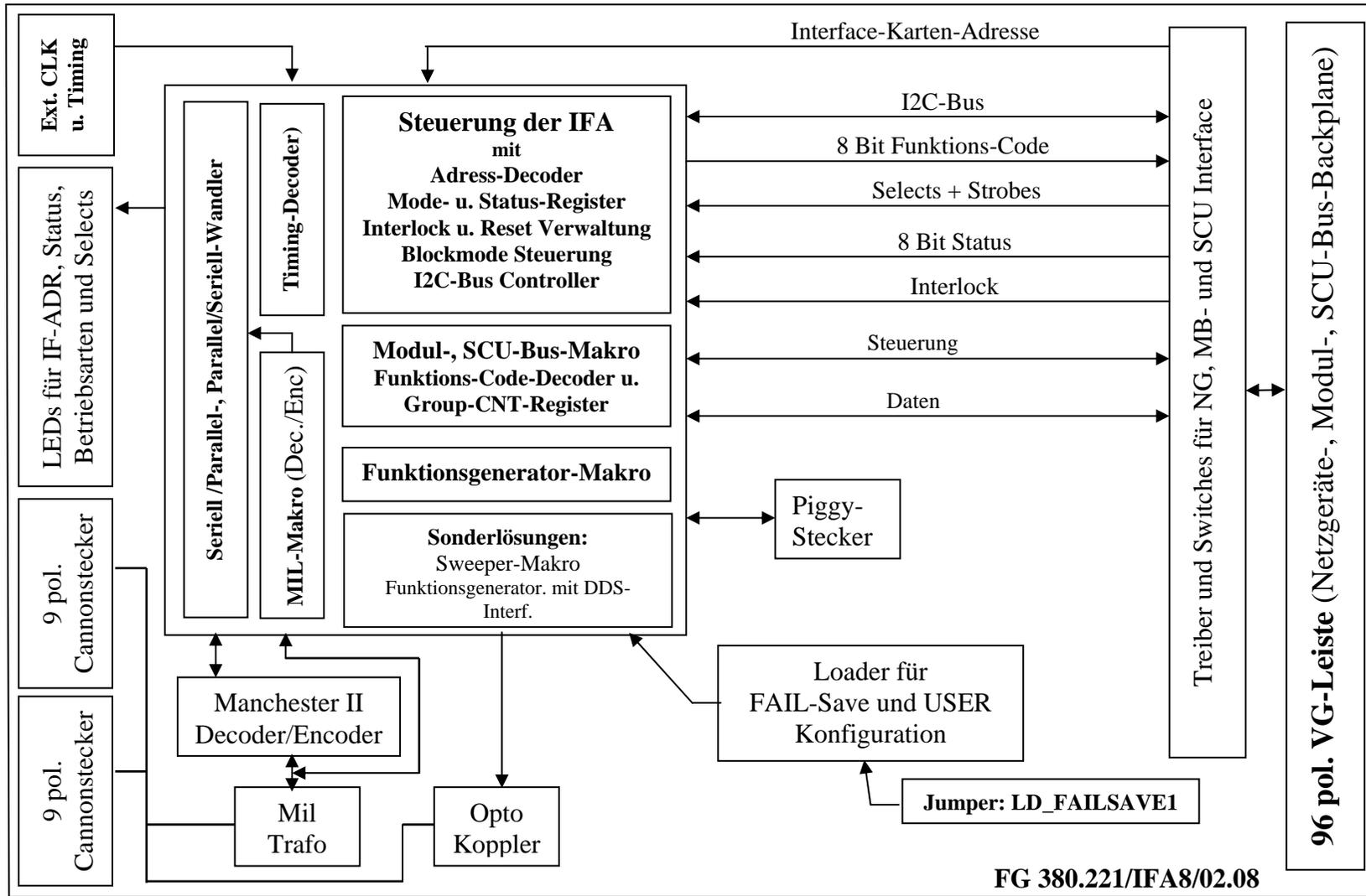
Die Datenleitungen zur VG-Leiste werden in den "Software-Betriebsarten" IFA-, FG-, FG-DDS-Mode solange auf "0" gehalten, bis Power-UP zurückgesetzt wurde.

1.2 Kompatibilität

Die Interface-Karten Generation ab **FG 380.22x** ist nur noch für die Verwendung auf den Backplanes für die Netzgeräte, den Modul- und den SCU-Bus entwickelt worden.

Auf die **IFA** können keine Piggys der Vorgängerversionen **FG 380.1xx - FG 380.211x** aufgesteckt werden.

1.3 Blockschaltbild



1.4 Funktionsbeschreibung

Die Datenübertragung von der **SE** zur **IFA** erfolgt über den **Device-Bus**, eine serielle Datenübertragung im Manchester-II-Code mit einer Datenrate von 1Mbit/s.

Das **Device-Bus-Kabel** verbindet die **SE** mit einem der beiden 9pol. Cannon-Stecker auf der Frontplatte der **IFA**. An dem zweiten Cannon-Stecker wird entweder ein Verbindungskabel zur nächsten Interface-Karte oder ein Abschlusswiderstand angeschlossen. Diese seriellen Manchester-II codierten Daten werden dann über einen Transformator galvanisch entkoppelt und von einem Decoder/Encoder (6408) in serielle digitale Daten umgewandelt, oder direkt an den zentralen Logik-Baustein vom Type **EP1F400C-7** der Firma ALTERA übergeben.

Der Logik-Baustein enthält:

- digitaler Decoder/Encoder für den Device-Bus (MIL-Makro)
- digitaler Encoder für den Timing-Bus
- serielle/parallele Datenwandlung und umgekehrt
- Adress-Decoder
- Steuer- und Statusregister
- Interlock und Reset-Verwaltung
- Blockmodesteuerung (für die Strahldiagnose)
- I2C-Controller.
- Makros für Modul-Bus, SCU-Bus und Funktionsgenerator
- Optionale Sonderlösungen, wie Sweeper-Makro und DDS-Interface für **FG**

Bei der Datenübertragung auf dem Device-Bus, unterscheidet man zwei Arten von 16 Bit Worten, das Daten- und das Command-Word.

Die Datenworte werden parallel an alle Interface-Karten an einem Device-Bus gesendet und in einem internen Datenregister zwischengespeichert.

Das Command-Word ist in 8 Bit Funktions-Code (D[15..8]) und 8 Bit Interface-Karten-Adresse(D[7..0]) aufgeteilt.

Mit den Funktions-Codes wird die Interface-Karte eingestellt, die Status-Register gelesen, die Datenrichtung festgelegt und die angeschlossenen Geräte gesteuert.

Die Funktions-Codes sind aber nur gültig, wenn der Adress-Decoder auf der **IFA** feststellt, dass die Adresse im Command-Word mit der Adresse der Interface-Karte von der VG-Leiste übereinstimmt.

Ausnahme!!

Die Adresse 255 (FF_{Hex}) im Command-Word ist für die Broadcast-Funktion reserviert. Diese wird verwendet, um Funktions-Codes per SE-Software an alle Interface-Karten an einem Device-Bus zu schicken.

Die digitalen Daten der **IFA** werden dann entsprechend der Betriebsart auf die Stecker und die VG-Leite aufgeteilt.

Über Optokoppler können bis zu drei Interlock-Meldungen (Summen-Interlock, Data-Request und Data-Ready) an die SE geschickt werden. Diese Interlockmeldungen sind über das Masken-Register zuzulassen oder zu sperren. Nach dem Einschalten ist nur der Summen-Interlock freigegeben, mit dem die Power-UP-Meldung der **IFA** der SE angezeigt wird.

1.5 Betriebsarten

Die IFA besteht aus einem allgemeinen Schaltungsteil und den betriebsartabhängigen Teil. Zum allgemeinen Schaltungsteil zählen:

Funktions-Code-Ausgänge, Status-Eingänge und I2C-Bus.

Diese allgemeinen Schaltungsteile sind in allen Betriebsarten, **außer in der Betriebsart "SCU-Bus-Controller"**, jederzeit verwendbar.

Die digitalen Ein- und Ausgänge werden in Abhängigkeit von der Betriebsart verwendet.

1.5.1 NG-Backplane

Für den Anschluss von Magnetnetzgeräten, wurde eine Netzgeräte-Backplane entwickelt. Für diese Backplane, sind die digitalen Ein- und Ausgänge der IFA in 16 Bit Sollwert und 16 Bit Istwert aufgeteilt. Der Funktions-Code aus dem allgemeinen Teil, wird hierbei als "Adresse" für die Leiterplatten auf dieser Backplane verwendet.

Diese Betriebsart ist die Betriebsart nach dem Einschalten (Power-UP).

1.5.2 SD-Backplane (Blockmode)

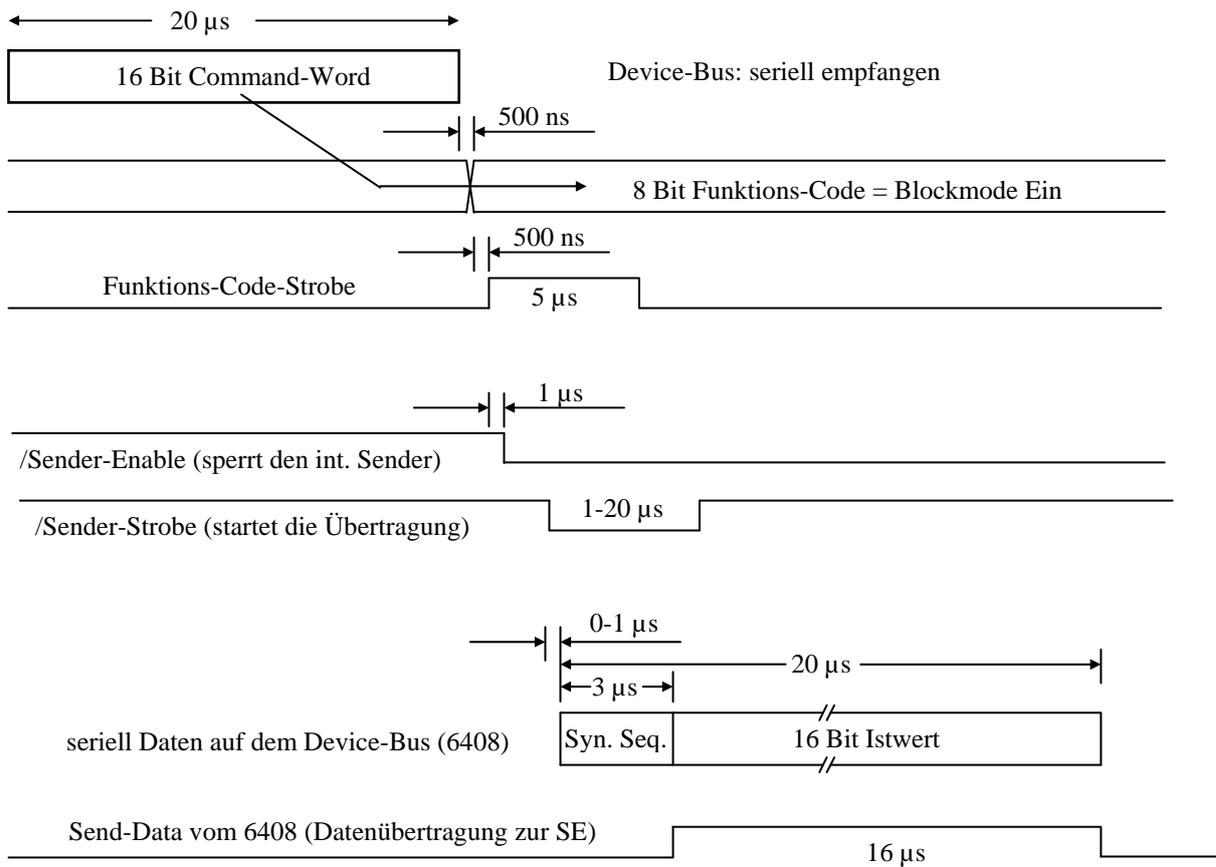
Der Betrieb der SD-Backplane erfolgt in der gleichen Konfiguration der IFA wie bei der NG-Backplane. Um die großen Daten-Pakete vom Dualport-RAM für die Strahldiagnose schneller und effektiver an die SE übertragen zu können, wird die externe Steuerung des Senders der IFA benutzt (Blockmode)

1.5.2.1 Lesen im Blockmode

Die Funktion "Lesen im Blockmode" muss durch den Befehl (Funktions-Code: Blockmode-Ein) an eine externe Ablaufsteuerung in der Hardware 'hinter' der Interface-Karte, eingeschaltet werden.

Die Sendersteuerung übernimmt dann durch ein "LOW-Signal" am Pin 5b (**/SEND_EN**) der VG-Leiste die Kontrolle über den Senderteil der **IFA**. Mit einem "LOW-Puls" am Pin 6b (**/SEND_STR**) der VG-Leiste, wird ein Datenwort von der **IFA** zur **SE** geschickt. An dem Pin 9b (**SD** vom 6408) der VG-Leiste kann festgestellt werden, ob die Datenübertragung gestartet wurde \uparrow und wann sie beendet ist \downarrow .

Timing:



Achtung:!!

Ab diesem Zeitpunkt hat die **SE keine Kontrolle** mehr über die **IFA**. Das bedeutet, der Datenbus ist mit der Übertragen der Daten zur **SE** belegt und es können keine Daten oder Funktions-Codes zur **IFA** gesendet werden. Die externe Sendersteuerung muss dafür sorgen, dass der Anschluss **/SEND_EN** nur für die **max. errechnete Übertragungszeit** der Daten auf "LOW" gehalten wird.

1.5.3 Modul-Bus-Mode

Der Mangel an Signalen für den Anwender auf der VG-Leiste der IFA führte zur Definition eines Busses mit zugehöriger Backplane.

- Adressierung der Module mit 5Bit Moduladresse und 8 Bit Subadresse
- Datentransfer 8-Bit bidirektional von der IFA zum Modul
(16 Bit Daten vom Device-Bus werden automatisch als 2 x 8 Bit übertragen)
- Identifizierung eines Moduls durch eindeutigen 8-Bit-Code (Modul-ID)
- Hardwaremäßige Überprüfung des Modul-Idents mit dem in der Verdrahtung vorgesehenen Ident-Code auf der VG96 oder VG160 -Leiste
(richtiger Kartentyp auf dem richtigen Platz?)
- 36 freie Pins auf der 96pol. VG-Leiste für Ein-/Aus-Signale des Anwenders.
Die Ein-/Aus-Signale werden über einen auf der Rückseite der Backplane steckbaren Adapter entnommen
- Busleitungen mit Abschlusswiderständen für hohe Signalqualität und Störsicherheit auf der Multilayer-Backplane, auch in Hinsicht auf EMV.

Standard-Modul-Bus-Zugriff (2 Schritte)

Datentransfers von/zu Modulen erfordern im Normalfall *zwei* Device-Bus-Transfers von je 2x16-Bit:

1. Modul-Karte-Adr.(Modul- u. Subadresse) als 16-Bit Daten + (Fct 11_{Hex} + IFK-Adr.) 16-Bit.
2. 16 Bit Moduldaten mit Funktions-Code = 10_{Hex} schreiben, oder
16 Bit Moduldaten mit Funktions-Code = 90_{Hex} lesen.

Durch die Initialisierung des **Funktions-Decoder-RAMs**, kann der direkte Zugriff über einen Funktions-Code ermöglicht werden (Zeitgewinn).

Für den Anwender ist entscheidend, dass er mit einem Schritt auf eine Ressource im System zugreifen kann. Dieser Zugriff erfolgt über eine über den Device-Bus ladbare Tabelle, das **Funktions-Decoder-RAM**. Der Vorteil dieses Zugriffs besteht auch

Der Vorteil dieses Zugriffs besteht auch darin, dass bestehende Gerätesoftware durch Hinzufügen der Tabelle als Initialisierung ansonsten unverändert bleiben kann, wenn auf Modul-Bus umgestellt wird. Wichtige Eigenschaft des RAMs ist seine Flüchtigkeit, die zu einer Re-Initialisierung nach jedem Power-Up zwingt.

Um mehrere Geräte mit einer einzigen Interface-Karte in einem Modul-Bus-Einschub realisieren zu können, wurde die virtuelle Interfacekarten-Adresse eingeführt.

Mit dem **MB Group-CNT-Register** kann die Anzahl der virtuellen Interface-Karten-Adressen festgelegt werden.

Nun sind mit dem **MB Group-CNT** und über das **Funktions-Decoder-RAM**, mehrere auf dem gleichen Gerätemodell basierende Geräte mit einer Interface-Karte und mit den gleichen Funktions-Codes in einem Modul-Bus-Einschub möglich.

1.5.4 SCU-Bus-Mode

Der Mangel an Signalen für den Anwender auf der VG-Leiste der IFA führte zur Definition eines weiteren Busses mit zugehöriger Backplane.

- Selektierung der bis zu 12 Slaves über separate Selektleitungen.
- Adressierung innerhalb der Slaves 16 Bit Adresse (bei der **IFA** auf 12 Bit begrenzt)
- Datentransfer 16-Bit bidirektional zwischen IFA und Slave
- Verteilung von parallelen Events über den **ADR**- und **Datenbus** inkl. Strobe
- 36 freie Pins auf der 96pol. VG-Leiste für Ein-/Aus-Signale des Anwenders.
Die Ein-/Aus-Signale werden über einen auf der Rückseite der Backplane steckbaren Adapter entnommen
- Busleitungen mit Abschlusswiderständen für hohe Signalqualität und Störsicherheit auf der Multilayer-Backplane, auch in Hinsicht auf EMV.

Standard-SCU-Bus-Zugriff (2 Schritte)

Datentransfers von/zum Slave erfordern im Normalfall **zwei** Device-Bus-Transfers von je 2x16-Bit:

1. Slave-Karten-Adr. (Slave # - u. Adresse) als 16-Bit Daten + (Fct 11_{Hex} + IFK-Adr.) 16-Bit.
2. 16 Bit Moduldaten mit Funktions-Code = 10_{Hex} schreiben, oder
16 Bit Moduldaten mit Funktions-Code = 90_{Hex} lesen.

Durch die Initialisierung des **Funktions-Decoder-RAMs**, kann der direkte Zugriff über einen Funktions-Code ermöglicht werden (Zeitgewinn).

Für den Anwender ist entscheidend, dass er mit einem Schritt auf eine Ressource im System zugreifen kann. Dieser Zugriff erfolgt über eine über den Device-Bus ladbare Tabelle, das **Funktions-Decoder-RAM**. Der Vorteil dieses Zugriffs besteht auch

Der Vorteil dieses Zugriffs besteht auch darin, dass bestehende Gerätesoftware durch Hinzufügen der Tabelle als Initialisierung ansonsten unverändert bleiben kann, wenn auf Modul-Bus umgestellt wird. Wichtige Eigenschaft des RAMs ist seine Flüchtigkeit, die zu einer Re-Initialisierung nach jedem Power-Up zwingt.

Um mehrere Geräte mit einer einzigen Interface-Karte in einem SCU-Bus-Einschub realisieren zu können, wurde die virtuelle Interfacekarten-Adresse eingeführt. Mit dem **SCU Group-CNT-Register** kann die Anzahl der virtuellen Interface-Karten-Adressen festgelegt werden.

Nun sind mit dem **SCU Group-CNT** und über das **Funktions-Decoder-RAM**, mehrere auf dem gleichen Gerätemodell basierende Geräte mit einer Interface-Karte und mit den gleichen Funktions-Codes in einem SCU-Bus-Einschub möglich.

1.5.5 SCU-ACU-Mode (V0.3)

Im SCU-ACU-Mode, ist ein Datentransfer nur mit dem Slave Nr. 1 (ACU) möglich. Durch die Beschränkung auf den Slave Nr. 1, kann auf alle 16 Adressleitungen zugegriffen werden.

Standard- SCU-ACU-Zugriff (2 Schritte)

Datentransfers von/zum Slave erfordern im Normalfall *zwei* Device-Bus-Transfers von je 2x16-Bit:

1. Adresse (von Slave #1) als 16-Bit Daten + (Fct 11_{Hex} + IFK-Adr.) 16-Bit.
2. 16 Bit Moduldaten mit Funktions-Code = 10_{Hex} schreiben, oder
16 Bit Moduldaten mit Funktions-Code = 90_{Hex} lesen.

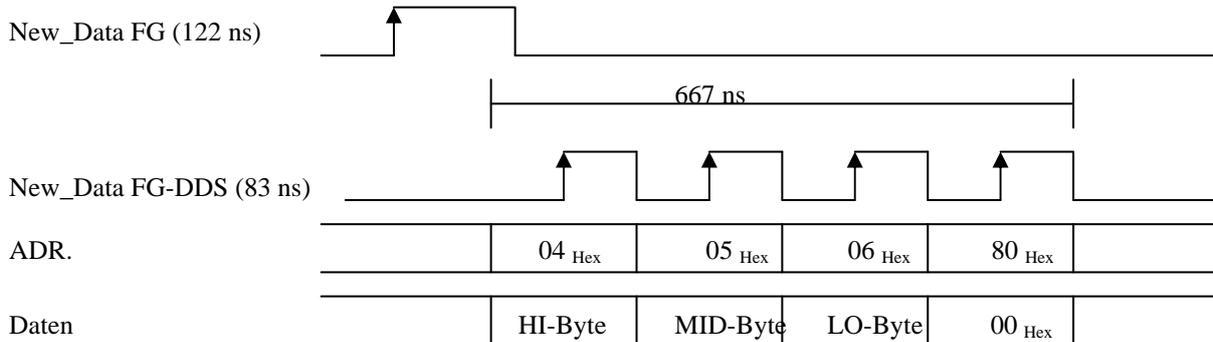
1.5.6 Funktionsgenerator-Mode

Mit dem **Funktionsgenerator-Mode**, ersetzt die **IFA** den alten Funktionsgenerator FG 380.540. Wird ein zusätzlicher externer Takt benötigt, kann dieser über eine 1pol. Lemo-Buchse auf der Frontplatte der **IFA** angeschlossen werden.

1.5.7 Funktionsgenerator-Mode mit DDS-Interface

Im Unterschied zur Betriebsart Funktionsgenerator-Mode, werden die 24 Bit Ausgangsdaten Byte-Seriell über die Pins vom 16 Bit Sollwert zum Anwender übertragen.

Timing für DDS-Interface



1.5.8 Sweeper

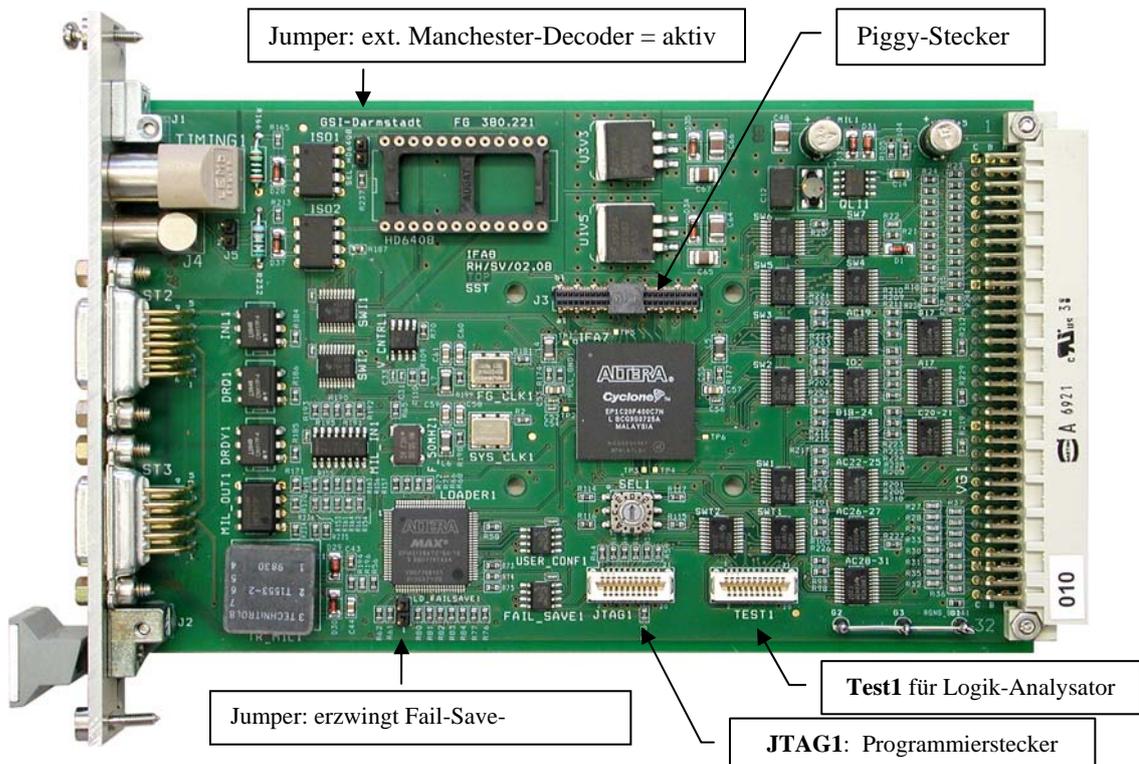
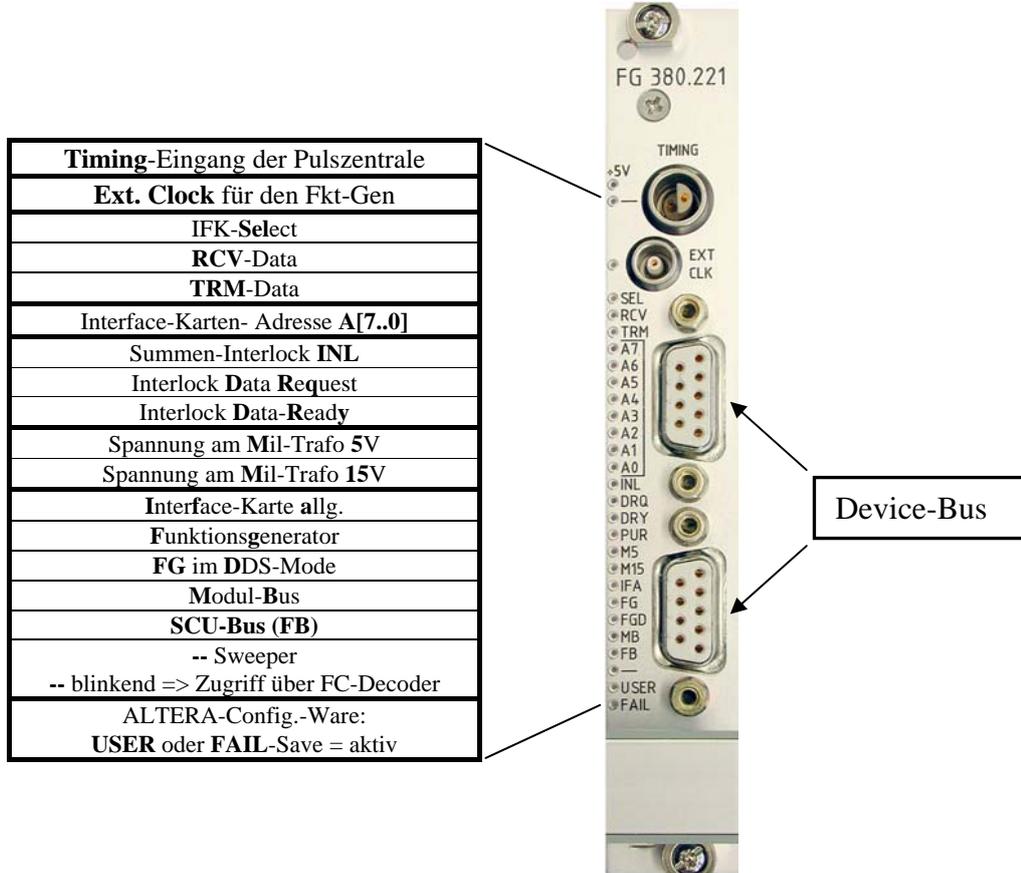
Der Sweeper-Mode ist eine abgeänderte Funktionsgenerator Betriebsart.

??

1.6 Technische Daten: (Kurzdaten)

- Interface-Karte im Europaformat (160x100mm).
- Frontplatte 3HE 4TE, mit Auswerfer.
- Anschluss über 96 pol. VG-Leiste für die Standard Netzgeräte, Modul-Bus- oder SCU-Bus-Backplane.
- Stromversorgung 5V/500mA, +15V/100mA (Device-Bus).
- Eingangsspannungsbereich für Summen-Interlock 5-15V, (offen, oder Spannung unter 4,0V = Interlock).
- alle anderen Ein- und Ausgänge TTL-kompatibel
- alle Busleitungen (**MB/SCU**) sind mit 2 Busabschlüssen je 330/220Ohm belastbar
- 2 x 9pol. Cannon-Buchsen auf der Frontplatte für den Device-Bus-Anschluss,
- Endknoten müssen mit einem Busabschluss (2*62 Ohm?) versehen werden.
- 3 optisch entkoppelte Interlockleitungen zur SE.
- 1pol. Lemo-Buchse für externen Takt des Funktionsgenerators (Optional).
- 2pol. Lemo-Buchse für das GSI-Timing-System (Event + virt. ACC).
- LED-Anzeigen, für die Interface-Karten-Adresse, den Betriebszustand (SEL, Interlocks, Power-Up-Reset, Mil (M5/M15), (RCV und TRM) und die Betriebsarten (Mode).
- Temperaturbereich 0-70°C.
- Betriebsart (Mode) per Software bzw. Hardware einstellbar, aber immer rücklesbar,
- Power-Down/Power-Up und Interlock feststellbar,
- Echo-Register zur Fehler-Diagnose.

1.7 Front- und Seitenansicht der Baugruppe



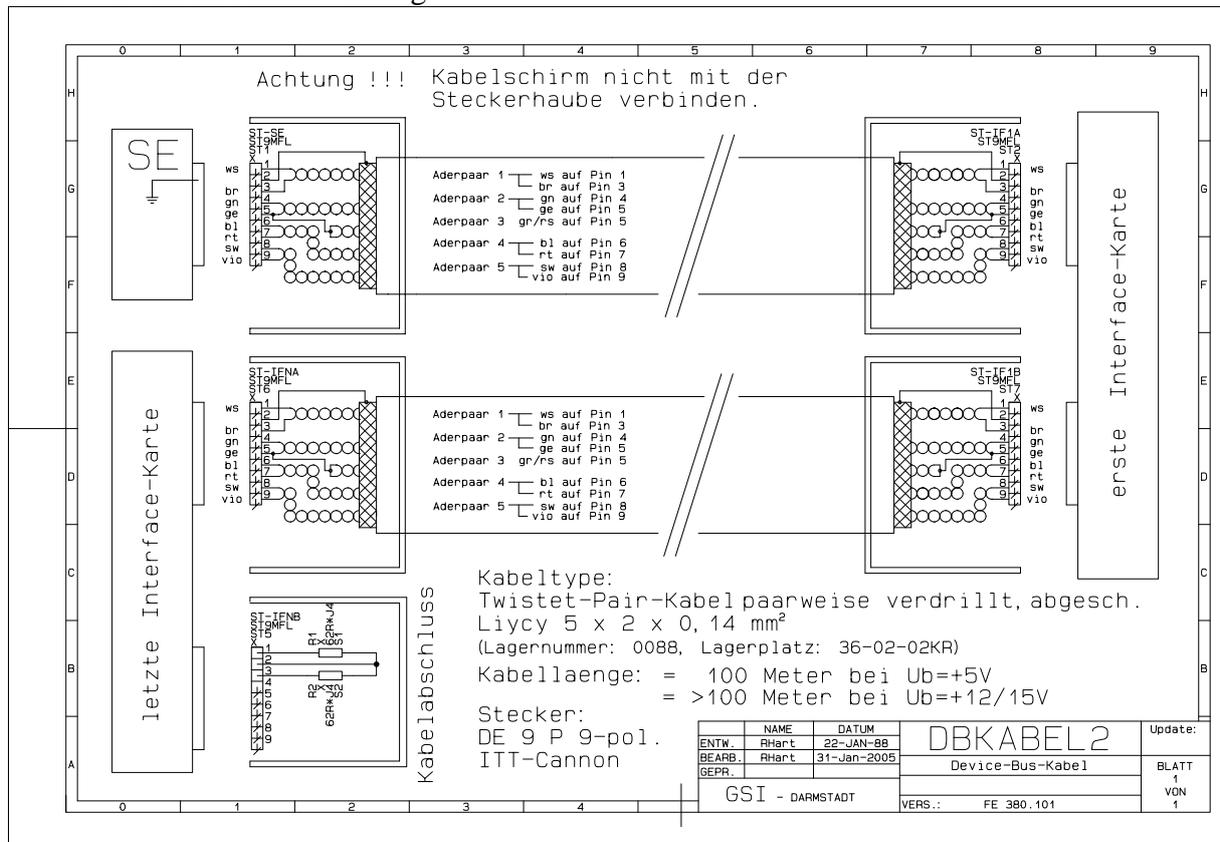
2 Device-Bus

Der Device-Bus ist ein serieller Bus zur Übertragung von 16-Bit-Daten in Manchester-II-Codierung mit 1 Mbit/s. Er entspricht physikalisch den Vorschriften der Mil-Std 1553 B-Spezifikation. Abweichend von der MIL1553 gibt es außer dem Parity-Bit keine Datensicherung, kein Echo für jedes Datum. Rückmeldung können die Slaves nur über zusätzliche GSI-spezifische Interrupt-Leitungen geben.

Die Einzelheiten:

- Der Device-Bus ist ein Multi-Drop Bus (Single-Master).
- Es können mehrere Teilnehmer verbunden werden (bei GSI normalerweise 16).
- Alle Slaves können über 3 potentialgetrennte Interrupt-Leitungen auf den Master (die SE) einwirken
- Die Kopplung ist transformatorisch und passiv.
- Es besteht dadurch eine Potentialtrennung zwischen Rechner und Gerät, eine defekte Koppelstelle kann den Bus nicht lahm legen.
- Für einen 16-bit-Daten-Transfer werden 20 μ s benötigt.
- Es kann zwischen Datentransfer und Command-Transfer unterschieden werden.
- Das Device-Bus-Kabel darf bis 200 m lang sein. (bei GSI-Standard-Datenkabeln, mit anderen Kabeln sind größere Entfernungen möglich)

EMV-Hinweis: Durch das nicht Verbinden des Schirms an den Slave-Interfaces vermeidet man Ausgleichsströme zwischen Interfaces an verschiedenen Erdpotentialen, verschlechtert aber evtl. die Abschirmwirkung



2.1 Schreibe Daten mit Funktions-Code

Die Datentransfer-Zeit für das Schreiben von 16 Bit Daten oder Kommandos beträgt jeweils 20 μ s. (Übertragungszeit für Daten + Funktions-Code = 40 μ s)

2.1.1 Schreibe Daten

Die parallelen Daten (16 Bit) stehen, nach der Übertragung über den Device-Bus, gleichzeitig an der VG-Leiste aller am gleichen Device-Bus angeschlossenen Interface-Karten und somit auf der Netzgeräte-Backplane zur Verfügung.

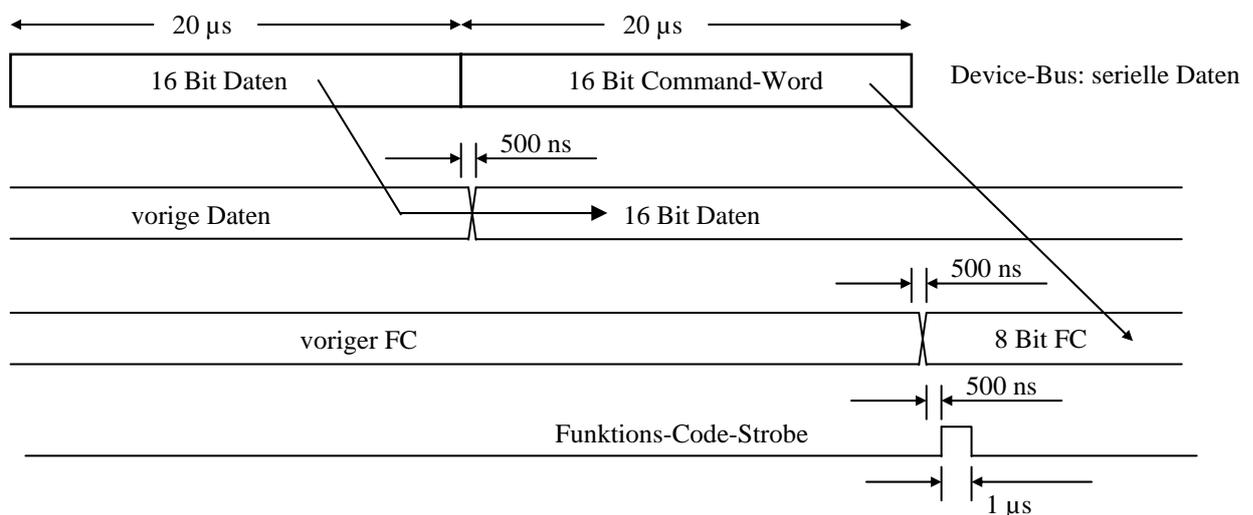
2.1.2 Schreibe Funktions-Code

Um die Daten entsprechend zuordnen zu können, muss ein Funktions-Code gesendet und auf den Karten der Netzgeräte-Backplane decodiert werden (ADC/DAC-Karte, Schaltkarte usw.). Der Funktions-Code wird zusammen mit der IFK-Adresse im zweiten Datenwort, dem "Command Word" übertragen.

| Funktions-Code | | | | | | | | IFK-Adresse (1-254, 255 = Broadcast) | | | | | | | |
|----------------|----|----|----|----|----|---|---|--------------------------------------|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Der Funktions-Code ist mit der positiven Flanke des FC-Stobes gültig.

2.1.3 Timing auf dem Device-Bus, Ausgang des Manchester-Decoders



2.2 Lese Daten oder Status

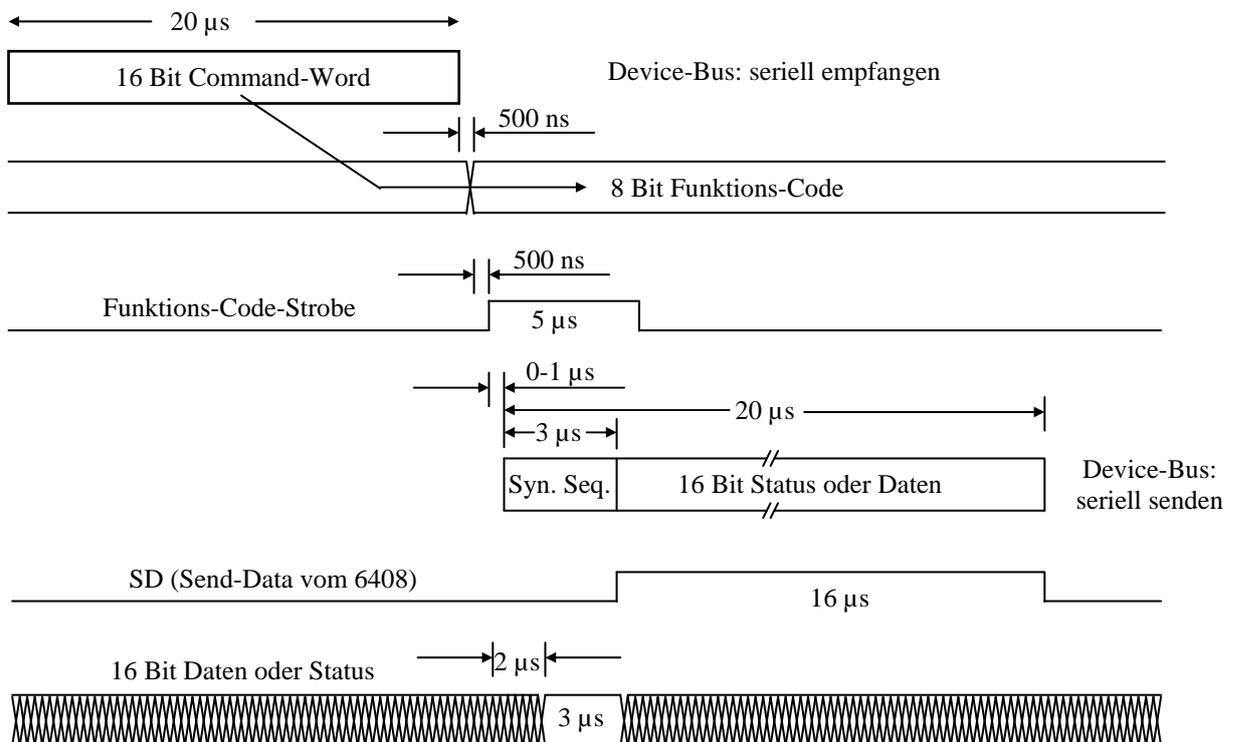
Die Datentransfer-Zeit für das Lesen von 16 Bit Daten oder Status, die Interface-Karte ist Slave der SE, d.h. alle Daten müssen per Funktions-Code angefordert werden ($20 \mu\text{s}$), beträgt deshalb ($20 \mu\text{s} + 20 \mu\text{s}$) $40 \mu\text{s}$.

Um die Daten von den Karten der Netzgeräte-Backplane oder der IFA, lesen zu können, muss ein Funktions-Code zur Interface-Karte gesendet werden. Mit der positiven Flanke des FC-Strobes wird die Übertragung gestartet.

Die Übertragung beginnt zwischen $0-1 \mu\text{s}$ nach der positiven Flanke des FC-Strobes mit der Sync-Sequence von $3 \mu\text{s}$. Während der Sync-Sequence werden die parallel anstehenden Daten in ein Schieberegister geladen, anschließend wird dann das Schieberegister von 'laden' auf 'schieben' umgeschaltet.

Daraus folgt, dass die Daten bzw. der Status zwischen 2 und $5 \mu\text{s}$ nach der positiven Flanke des FC-Strobes stabil sein müssen.

2.2.1 Timing beim Lesen auf dem Device-Bus und am Manchester-Encoder



3 Liste der Funktions-Codes

| Schreiben (SE ⇒ IFK) | | | | Lesen (IFK ⇒ SE) | | | |
|----------------------|-----|-----------------------------------|-----------|------------------|-----|--|---------------|
| Dez | Hex | Funktion | Betr.-Art | Dez | Hex | Funktion | Betr.-Art |
| 0 | 0 | keine Funktion | | 128 | 80 | Daten lesen | |
| 1 | 1 | Schalter auf Standby (Reset) | FG | 129 | 81 | Istwert 1 lese | |
| 2 | 2 | Gerät Einschalten | | | | | |
| 3 | 3 | Gerät Ausschalten | | 136 | 88 | Istwert 8 lesen | |
| 4 | 4 | Polarität + setzen | | 137 | 89 | Echo-Sollwert rücklesen | IFA |
| 5 | 5 | Polarität - setzen | | | | | |
| 6 | 6 | Sollwert 1 setzen | FG | 141 | 8D | Piggy-EPLD-Vers. lesen | MB/SCU |
| 7 | 7 | Sollwert 2 setzen | FG | 142 | 8E | Piggy-Back ID lesen | IFA/MB/SCU/FG |
| 8 | 8 | Sollwert 3 setzen | FG | 143 | 8F | Datenblock lesen | |
| 9 | 9 | Sollwert 4 setzen | FG | | | | |
| 10 | A | Sollwert 5 setzen | FG | 144 | 90 | Datenbus lesen | MB/SCU |
| 11 | B | Sollwert 6 setzen | | | | | |
| 12 | C | Sollwert 7 setzen | | 145 | 91 | G-Daten 1 / Stat_1 | FG |
| 13 | D | Sollwert 8 setzen | | 146 | 92 | G-Daten 2 / Stat_2 | FG |
| 14 | E | Wait | | 147 | 93 | G-Daten 3 / Sollwert_1 | FG |
| 15 | F | Release | | 148 | 94 | G-Daten 4 / Sollwert_2 | FG |
| | | | | 149 | 95 | G-Daten 5 / SP-Sollwert_1 | FG |
| 16 | 10 | Datenbus setzen | MB/SCU | 150 | 96 | G-Daten 6 / SP-Sollwert_2 | FG |
| 17 | 11 | Adressbus setzen | MB/SCU | | | | |
| | | | | 151 | 97 | G-Daten 7 / IFA-Mode | IFA |
| 18 | 12 | Set INR-Maske | IFA | 152 | 98 | G-Daten 8 / I ² C-CMD/Daten | IFA |
| 19 | 13 | Echo-Sollwert setzen | IFA | | | | |
| | | | | 153 | 99 | FC-Decoder Daten (RAM) | MB/SCU |
| 32 | 20 | Broadcast | FG | 154 | 9A | MB-Status-Register | MB/SCU |
| | | | | 155 | 9B | Group-Count-Register | MB/SCU |
| 48 | 30 | Modul-Bus-Funktions-Code | MB/SCU | 156 | 9C | Firmware-Loader-Data | IFA |
| | | Modul-Bus-Funktions-Code | MB/SCU | 157 | 9D | Firmware-Loader-Status | IFA |
| 79 | 4F | Modul-Bus-Funktions-Code | MB/SCU | 158 | 9E | FG-Trig. "FG 380.751" LED's | IFA-Piggy |
| | | | | 159 | 9F | FG-Trig. "FG 380.751" Out/LED's | IFA-Piggy |
| 95 | 5F | Conv.-Command für ADC's | IFA | | | | |
| 96 | 60 | IFA-Mode setzen | IFA | 176 | B0 | Modul-Bus-Funktions-Code | MB/SCU |
| 97 | 61 | I ² C-CMD/Daten setzen | IFA | | | Modul-Bus-Funktions-Code | MB/SCU |
| 98 | 62 | FC-Dec. RAM-ADR setzen | MB/SCU | 191 | BF | Modul-Bus-Funktions-Code | MB/SCU |
| 99 | 63 | FC-Dec.-RAM-Daten setzen | MB/SCU | | | | |
| 100 | 64 | Group-Count setzen | MB/SCU | 192 | C0 | Status-Byte1 lesen | |
| 101 | 65 | Firmware-Loader-Data | IFA | 193 | C1 | Status-Byte2 lesen | |
| 102 | 66 | Firmware-Loader-Ctrl | IFA | 194 | C2 | Status-Byte3 lesen | |
| 103 | 67 | SCU-Status | SCU | | | | |
| 104 | 68 | SCU-SRQ-Enable | SCU | 201 | C9 | Interrupt-Maske lesen | IFA |
| 105 | 69 | FG-Trig. "FG 380.751" LED's | IFA-Piggy | 202 | CA | Ctrl-Register lesen | IFA |
| 106 | 6a | FG-Trig. "FG 380.751" Out/LED's | IFA-Piggy | 203 | CB | Reserve für Maske-HI-INL | IFA |
| | | | | 204 | CC | IFK-Identcode lesen | IFA |
| 117 | 75 | Res. Piggy-Err./DTACK-Err. | IFA | 205 | CD | IFKEPLD-Version lesen | IFA |
| | | Reserve für HI-INL | IFA | | | | |
| 125 | 7D | Reset RCV-Error(VWTest) | IFA | | | | |
| 126 | 7E | Reset Broadcast | IFA | 208 | D0 | SCU-SRQ-Enable | SCU |
| 127 | 7F | Set Broadcast | IFA | 209 | D1 | SCU-SRQ-Activ | SCU |
| | | | | 210 | D2 | SCU-SRQ-In | SCU |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |

Alle Funktions-Codes stehen in allen Betriebsarten, **außer in der Betriebsart "SCU-Bus-Controller"**, an der VG-Leiste zur Verfügung.

Einige Funktions-Codes werden für die internen Daten-, Steuer- und Statusregister der **IFA** bzw. innerhalb der "betriebsartabhängigen" Makros verwendet.

IFA = Interface-Karte, **MB** = Modul-Bus, **SCU** = SCU-Bus, **FG** = Funktionsgenerator

4 Register

4.1 Steuerregister (Funktions-Codes siehe Kap. 3)

4.1.1 Mode-Register

Die unterschiedlichen Betriebsarten der **IFA** können nur zum Teil per SE-Software eingestellt werden, einige Betriebsarten müssen aber wegen der Sicherstellung der richtigen Treiberrichtung zwingend per Hardware über die VG-Leite bzw. einen Piggy-Stecker festgelegt werden.

Nach dem Einschalten der **IFA** ist der Interface-Karten-Mode, für die Netzgeräte (**NG**)-Backplane eingeschaltet. Sollte aber eine "Hardware-Betriebsart" z.B. Modul-Bus-, SCU-Bus oder IO-Bus-Mode gesetzt sein, ist die selektierte Betriebsart sofort aktiv.

| IFA-Mode-Register (write Daten FC=60 Hex) / (read Daten FC=97 Hex) | | | |
|---|------|---------|--|
| Daten | | Mode | |
| write | read | | |
| 0000 | 0000 | IFA | allg. Interface-Karte (nach dem Einschalten) |
| 0001 | 0001 | FG | FunktionsGenerator |
| --- | 0002 | MB | Modul-Bus-IF (Mode ist nur über VG-Leisten Pin 7b einschaltbar). |
| 0004 | 0004 | Sweeper | Sweeper (ist nur möglich, wenn Bit im Mode-STS rücklesbar). |
| --- | 0008 | IO-Bus | IO-Bus-IF (Mode ist nur über VG-Leisten Pin 7b und mit Piggy "IFPIO" einschaltbar. Modul-Bus Funktion mit IO-Bus Interface) |
| 0010 | 0010 | FG-DDS | Funktionsgenerator mit DDS-Interface (ist nur möglich, wenn Bit im Mode-STS rücklesbar). |
| --- | 0020 | SCUB | SCU-Bus-IF (Mode ist nur über VG-Leisten Pin 8b einschaltbar). |
| 0040 | 0040 | SCU-ACU | SCU-ACU nur einschaltbar, im SCU-Mode (VG- Pin 8b = 0). (V0.3) |

4.1.2 Interrupt-Maske

Mit der Interrupt-Maske können die Interrupts der IFA oder der angeschlossenen Geräte zugelassen oder gesperrt werden.

Nach dem Einschalten sind die Interrupts Data-Request (**DRQ**) und Data-Ready (**DRDY**) ausgeschaltet, und der Interrupt Summen-Interlock (**S-INL**) eingeschaltet.

Die Daten für die Interrupt-Maske werden mit dem Funktions-Code **FC=12_{Hex}** geschrieben.

| Bit | Interrupt-Masken-Register (write mit FC=12_{Hex}) | |
|------|--|------------------------|
| 15 | Sum.-Interlock | 1=enable / 0 = disable |
| 14 | Data-Ready | 1=enable / 0 = disable |
| 13 | Data-Request | 1=enable / 0 = disable |
| 12-0 | | frei |

Die Interrupt-Maske wird mit dem Funktions-Code **FC=C9_{Hex}** ausgelesen.

4.1.3 Echo-Register

Um die Datenübertragung zwischen SE und IFA zu testen, wird ein 16 Bit Datenwort mit dem Funktions-Code FC=13_{Hex} geschrieben und mit dem Funktions-Code FC=89_{Hex} gelesen.

4.1.4 I2C-Bus

| Bit | I ² C-CMD und Daten-Register(write Daten FC=61 Hex) | | |
|--------|--|--|-----------|
| 15..13 | frei | | |
| 12 | CMD-Receive | Liest ein Byte vom I2C-Bus und setzt Acknowledge | 1 aktiv |
| 11 | CMD-Send | Sendet die Daten vom Input-Port zum I2C-Bus. | 1 aktiv |
| 10 | CMD-Start | Erzeugt eine Start-Bedingung auf dem I2C-Bus. | 1 aktiv |
| 9 | CMD-Stop | Erzeugt eine Stopp-Bedingung auf dem I2C-Bus. | 1 aktiv |
| 8 | ACK-TX | Acknowledge. Wird nach dem lesen der 8 Datenbits, vom Master zum Slave gesendet. | 0 = ackn. |

| | |
|------|--------------------------------------|
| 7..0 | I ² C-Adressen oder Daten |
|------|--------------------------------------|

| Bit | I ² C-Status und Daten-Register(read mit FC=96 Hex) | |
|--------|--|---------|
| 15..13 | Frei | |
| 12 | Busy | 1 aktiv |
| 11 | Denable | 1 aktiv |
| 10 | Dvalid | 1 aktiv |
| 9 | Status | 1 aktiv |
| 8 | ACK-RX | 1 aktiv |

| | |
|------|------------------------|
| 7..0 | I ² C-Daten |
|------|------------------------|

Beispiel: Lese Daten vom Temperatur-Sensor LM75 (A2+A1=0, A0=1)

| frei | frei | frei | CMD-Receive | CMD-Send | CMD-Start u. Send Data | CMD-Stop | ACK-TX | A6 vom LM75 (fest) | A5 vom LM75 (fest) | A4 vom LM75 (fest) | A3 vom LM75 (fest) | A2 vom LM75 (Pin-5) | A1 vom LM75 (Pin-6) | A0 vom LM75 (Pin-7) | 1 = read / 0 = write | Schreibe mit FC = 61 Hex | Beschreibung |
|----------|------|------|-------------|----------|------------------------|----------|---------|--------------------|--------------------|--------------------|--------------------|---------------------|---------------------|---------------------|----------------------|--------------------------|---------------------------------|
| D[15..8] | | | | | | | D[7..0] | | | | | | | HEX | | | |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0491 | Start I2C-Bus, write Slave-ADR. |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | x | x | x | x | x | x | x | x | 1100 | Read MSB-Byte, mit Ackn.=1 |

Lese I²C-Daten-Register(read mit FC=96 Hex) D[7..0] = MSB-Daten

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|------|----------------------------|
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | x | x | x | x | x | x | x | x | 1100 | Read LSB-Byte, mit Ackn.=1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | x | x | x | x | x | x | x | x | 0200 | Stop I2C-Bus |

Lese I²C-Daten-Register(read mit FC=96 Hex) D[7..0] = LSB-Daten

| Beispiel, als Temperatur ergibt sich: | | | | | | | | | | | | | | | |
|---------------------------------------|----|----|----|----|----|----|----|--------------------------|---|---|---|---|---|---|---|
| MSB-Daten (gelesen = 1A) | | | | | | | | LSB-Daten (gelesen = FF) | | | | | | | |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Daten für Temp-Wert | | | | | | | | undef. | | | | | | | |
| MSB | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | x | x | x | x | x | x | x |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | x | x | x | x | x | x | x |

So ergibt sich ein Wert von 35 Hex, bei 0,5 °C/Bit, entspricht das: 26,5 °C.

4.2 Status-Register (Funktions-Codes siehe Kap. 3)

4.2.1 Interlock-Status-Register

| Bit | Status-Register (read mit FC=C9 Hex) | | |
|------|--------------------------------------|------------------|----------|
| 15 | Interlock-Maske | Sum.-Interlock | 1 Enable |
| 14 | Interlock-Maske | Data-Ready | 1 Enable |
| 13 | Interlock-Maske | Data-Request | 1 Enable |
| 12 | Signal | Summen-Interlock | 0 aktiv |
| 11 | Signal | Data-Ready | 0 aktiv |
| 10 | Signal | Data-Request | 0 aktiv |
| 9 | Slave-Ready | | 1 aktiv |
| 8 | Power-UP Flag | | 0 aktiv |
| 7..0 | | | “00“ |

4.2.2 IFA-Status-Register

| Bit | Control-Register (read mit FC=CA Hex) | | |
|------|--|------------------|------|
| 15 | Mil-Spannung 15V | 1 aktiv | |
| 14 | Piggy-Error (IFA) / DTACK-Error (MB) | 0 aktiv | |
| 13 | IRQ VG-Leiste (Pin 11a) | 1 aktiv | |
| 12 | Jumper, Select_6408 (HD6408 ist aktiv) | 0 = gest.(aktiv) | |
| 11 | frei | 0V | |
| 10 | frei | 0V | |
| 9 | Broadcast | 1 aktiv | |
| 8 | Receive-Error | 1 aktiv | |
| 7..0 | | | “00“ |

4.2.3 Modul-Bus-Status-Register

| Bit | Modul-Bus-Status-Register (read mit FC=9A Hex) | | |
|------|--|--|------|
| 15 | GND (0V) | | |
| 14 | GND (0V) | | |
| 13 | GND (0V) | | |
| 12 | GND (0V) | | |
| 11 | GND (0V) | | |
| 10 | GND (0V) | | |
| 9 | GND (0V) | | |
| 8 | GND (0V) | | |
| 7..0 | | | “00“ |

4.2.4 ID-Register

Alle Interface-Karten ab FG 380.2xx und FG 450.01x sind mit einem ID-Register ausgerüstet, um die Karten eindeutig identifizieren zu können.

| Bit | Interface-Karten ID-Register (read mit FC=CC Hex) | |
|-------|---|------|
| 15..8 | | “FB“ |
| 7..0 | | “00“ |

| ID-Code | Nummer | Beschreibung |
|-----------------------|------------|--|
| 254/FE _{Hex} | FG 380.201 | allgemeine digitale Interface-Karte (Job: IFA2) |
| 253/FD _{Hex} | FG 450.011 | Interface-Karte für Modul-Bus im Ionenquellen-Terminal) |
| 253/FD _{Hex} | FG 450.012 | Interface-Karte für Modul-Bus im Ionenquellen-Terminal) |
| 252/FC _{Hex} | FG 380.203 | allgemeine digitale Interface-Karte (Job: IFA4) |
| 251/FB _{Hex} | FG 380.211 | neue allg. dig. Interf. Karte und Modul-Bus Interf. (Job: IFA6) |
| 250/FA _{Hex} | FG 380.221 | IF-Karte mit Modul- und SCU-Bus Interf. (Job: IFA8) |
| | | |
| | | |

4.2.5 EPLD-Versions-Register

Das EPLD-Versions-Register wird mit dem Funktions-Code **FC=CD**_{Hex} ausgelesen.

| Bit | EPLD-Version | | EPLD-Revision | | frei (Default = 0) | |
|-----|--------------|------|---------------|-----|--------------------|-----|
| Bit | 15 | - 12 | 11 | - 8 | 7 | - 0 |

Die Version 1.6 ist die z.Z. aktuelle Version.

5 Modul-Bus

5.1 Modul-Bus-Adressierung

5.1.1 Standard-Modul-Bus-Zugriff (2 Schritte)

Datentransfers von/zu Modulen erfordern im Normalfall *zwei* Device-Bus-Transfers von je 2x16-Bit:

3. Modul-Karten-Adr.(Modul- u. Subadresse) als 16-Bit Daten + (Fct 11_{Hex} + IFK-Adr.) 16-Bit.
4. 16 Bit Moduldaten mit Funktions-Code = 10_{Hex} schreiben, oder
16 Bit Moduldaten mit Funktions-Code = 90_{Hex} lesen.

5.1.2 Schneller Modul-Bus-Zugriff (1 Schritt)

Schnellere Zugriffe auf Module mit nur einem Device-Bus-Transfer sind nur möglich, wenn das **Funktions-Code-Decoder-RAM** auf der **IFA** verwendet wird, das Zugriffe auf Module mit Hilfe von Funktions-Codes erlaubt

5.1.2.1 Initialisierung des Funktions-Code-Decoders

Die Initialisierung des Funktions-Code-Decoders erfolgt in zwei Stufen.

1. Die Adress-Position, die dem zu decodierenden Funktions-Code entspricht, wird mit dem Funktions-Code **FC=62**_{Hex} geschrieben.

| | | | | | | | | | | |
|------|---|---|---|---|---|---|---|---|---|---|
| frei | | | Funktions-Code der decodiert werden soll. (=RAM-ADR.) | | | | | | | |
| 15 | - | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

2. Die Modul-Bus-Adresse, die dann auf dem Modul-Bus über den Funktions-Code angesprochen werden soll, wird mit dem Funktions-Code **FC=63**_{Hex} geschrieben.

| | | | | | | | | | | | | | | | |
|----------------|---|----|----|----|----|---|-----------|---|---|---|---|---|---|---|---|
| Modul-Bus-ADR. | | | | | | | Sub.-ADR. | | | | | | | | |
| x | x | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

5.1.2.2 Initialisierung des MB-Group-Count

| MB-Group-Cnt.-Reg. (write Daten FC=64 Hex) / (read Daten FC=9B Hex) | | |
|---|----------|---|
| Daten | MB-Group | |
| 0000 | 0 | → MB-read/write mit Basis-ADR |
| 0002 | 2 | → MB-read/write mit Basis-ADR und 1 virt. ADR. |
| 0004 | 4 | → MB-read/write mit Basis-ADR und 3 virt. ADR. |
| 0008 | 8 | → MB-read/write mit Basis-ADR und 7 virt. ADR. |
| 000F | 16 | → MB-read/write mit Basis-ADR und 15 virt. ADR. |

Achtung: !!

Beim Betrieb der Interface-Karte im Modul-Bus-Mode und einem Group-Count >0 **müssen die entsprechenden untersten 1-4 Bits der Basisadr. 0 sein., weil sie direkt zur Unterdekodierung herangezogen werden** (z.B.: Group-Count = 4 → IFK-ADR. Bit2, Bit1 und Bit0 = Low (0V)). Das Lesen der Konfiguration mit den Funktions-Codes Adress- oder Datenbus schreiben bzw. Datenbus lesen ist sonst, auch mit der Basis-Adr., nicht mehr möglich.

6 SCU-Bus

6.1 SCU-Bus-Mode-Adressierung

6.1.1 Standard-SCU-Bus-Zugriff (2 Schritte)

Datentransfers von/zum Slave erfordern im Normalfall **zwei** Device-Bus-Transfers von je 2x16-Bit:

1. SCU-Bus-Adr. [Slave # (D15-D12) + Slave-Adr. (D11-D0)] als 16-Bit Daten + (Fct 11_{Hex} + IFK-Adr.) 16-Bit.
2. 16 Bit SCU-Bus-Daten mit Funktions-Code = 10_{Hex} schreiben, oder
16 Bit SCU-Bus-Daten mit Funktions-Code = 90_{Hex} lesen.

6.1.2 Schneller SCU-Bus-Zugriff (1 Schritt)

Schnellere Zugriffe auf Slaves mit nur einem Device-Bus-Transfer sind nur möglich, wenn das **Funktions-Code-Decoder-RAM** auf der **IFA** verwendet wird, das Zugriffe auf Slaves mit Hilfe von Funktions-Codes erlaubt

6.1.2.1 Initialisierung des Funktions-Code-Decoder

Die Initialisierung des Funktions-Code-Decoders erfolgt in zwei Stufen.

1. Die Adress-Position, die dem zu decodieren Funktions-Code entspricht, wird mit dem Funktions-Code **FC=62**_{Hex} geschrieben.

| frei | | | Funktions-Code der decodiert werden soll. (=RAM-ADR.) | | | | | | | |
|------|---|---|---|---|---|---|---|---|---|---|
| 15 | - | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

2. Die Slave-Nr. und die -Adresse, die dann auf dem SCU-Bus über den Funktions-Code angesprochen werden soll, wird mit dem Funktions-Code **FC=63**_{Hex} geschrieben.

| Slave # | | | | Adresse | | | | | | | | | | | |
|---------|----|----|----|---------|----|---|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

6.1.2.2 Initialisierung des SCU-Bus-Group-Cont

| SCU-Bus-Group-Cnt.-Reg. (write Daten FC=64 Hex) / (read Daten FC=9B Hex) | | |
|--|-----------|--|
| Daten | SCU-Group | |
| 0000 | 0 | → SCU-read/write mit Basis-ADR |
| 0002 | 2 | → SCU-read/write mit Basis-ADR und 1 virt. ADR. |
| 0004 | 4 | → SCU-read/write mit Basis-ADR und 3 virt. ADR. |
| 0008 | 8 | → SCU-read/write mit Basis-ADR und 7 virt. ADR. |
| 000F | 16 | → SCU-read/write mit Basis-ADR und 15 virt. ADR. |

Achtung: !!

Beim Betrieb der Interface-Karte im SCU-Bus-Mode und einem Group-Count >0 **müssen die entsprechenden untersten 1-4 Bits der Basisadr. 0 sein., weil sie direkt zur Unterdekodierung herangezogen werden** (z.B.: Group-Count = 4 → IFK-ADR. Bit2, Bit1 und Bit0 = Low (0V)). Das Lesen der Konfiguration mit den Funktions-Codes Adress- oder Datenbus schreiben bzw. Datenbus lesen ist sonst, auch mit der Basis-Adr., nicht mehr möglich.

6.2 SCU-ACU-Mode-Adressierung (V0.3)

6.2.1 Standard-SCU-ACU-Bus-Zugriff

Alle Zugriffe auf den SCU-Bus erfolgen wie im SCU-Bus-Mode.

Da aber im SCU-ACU-Mode nur auf den Slave #1 (ACU) zugegriffen werden kann, entfällt die Slave# bei der Adressierung.

Somit steht der Gesamte 16-Bit-Adressraum zur Verfügung

SCU-Adresse im SCU-Bus-Mode: [Slave # (D15-D12) + Slave-Adr. (D11-D0)]
SCU-Adresse im SCU-ACU-Mode: [D15..D0]

7 Manuals

7.1 Modul-Bus

7.2 SCU-Bus

7.3 Funktionsgenerator

8 Anhang

8.1 Pinbelegung der VG-Steckerleiste

| PIN | NG-IFK | T | SCU-Master | T | SCU-Slave | |
|-----|--------|----|------------|----|------------|----|
| 1A | +5V | - | +5V/2A | - | +5V/2A | |
| 2A | +15V | - | +15V/1A | - | +15V/1A | |
| 3A | A 0 | SI | IFK-A0 | SI | DGND | |
| 4A | A 2 | SI | IFK-A2 | SI | User-0 | |
| 5A | A 4 | SI | IFK-A4 | SI | User-3 | |
| 6A | A 6 | SI | IFK-A6 | SI | User-6 | |
| 7A | STS 0 | SI | Master-12 | S | User-9 | |
| 8A | STS 2 | SI | Master-15 | S | User-12 | |
| 9A | STS 4 | SI | Master-18 | S | User-15 | |
| 10A | STS 6 | SI | nSRQ-1 | SI | User-18 | |
| 11A | nIRQ | SI | nBSel-1 | SO | User-21 | |
| 12A | FC 0 | SO | nSRQ-4 | SI | User-24 | |
| 13A | FC 2 | SO | nBSsel-4 | SO | User-27 | |
| 14A | FC 4 | SO | DGND | - | DGND | |
| 15A | FC 6 | O | R/W | BO | R/W | I |
| 16A | nc | | nc | - | Spare0 | |
| 17A | SW-GNA | | nEvent-Str | BO | nEvent-Str | I |
| 18A | SW 0 | O | nReset | BO | nReset | I |
| 19A | SW 2 | O | nBSel-8 | SO | DGND | |
| 20A | SW 4 | O | A14 | BO | A14 | I |
| 21A | SW 6 | O | A12 | BO | A12 | I |
| 22A | SW 8 | O | A10 | BO | A10 | I |
| 23A | SW 10 | O | A8 | BO | A8 | I |
| 24A | SW 12 | O | A6 | BO | A6 | I |
| 25A | SW 14 | O | A4 | BO | A4 | I |
| 26A | IW 4 | I | A2 | BO | A2 | I |
| 27A | IW 6 | I | A0 | BO | A0 | I |
| 28A | IW 8 | I | D6 | BB | D6 | BB |
| 29A | IW 10 | I | D4 | BB | D4 | BB |
| 30A | IW 12 | I | D2 | BB | D2 | BB |
| 31A | IW 14 | I | D0 | BB | D0 | BB |
| 32A | GND | | DGND | - | DGND | |

| PIN | NG-IFK | T | SCU-MA | T | SCU-Slave | |
|-----|---------------|----|-----------|----|-----------|----|
| 1B | I2C-LDA | S | +5V | - | +5V/2A | |
| 2B | I2C-LCL | S | AGND | - | AGND | |
| 3B | New-Data | SO | DGND | - | DGND | |
| 4B | FC-ST-Piggy | SO | Master-4 | S | User-1 | |
| 5B | nSend-Enable | SI | Master-7 | S | User-4 | |
| 6B | nSend-STR-Ex. | SI | Master-10 | S | User-7 | |
| 7B | nSel ModBus | SI | nSEL-MB | SI | User-10 | |
| 8B | ST4-3 | SI | nSEL-SCU | SI | User-13 | |
| 9B | Send-Data | SO | Master-19 | S | User-16 | |
| 10B | nDRQ | SI | nSRQ-2 | SI | User-19 | |
| 11B | nDRDY | SI | nBSel-2 | SO | User-22 | |
| 12B | nPu.-Reset | SO | nSRQ-5 | SI | User-25 | |
| 13B | FC-Strobe | SO | nBSel-5 | SO | User-28 | |
| 14B | 6Mhz | SO | DGND | - | DGND | |
| 15B | DRQ | O | nDS | BO | nDS | I |
| 16B | ST4-4 | S | res. | S | res. | |
| 17B | Output (frei) | O | D15 | BB | D15 | BB |
| 18B | SWF 0 | O | D14 | BB | D14 | BB |
| 19B | SWF 1 | O | D13 | BB | D13 | BB |
| 20B | SWF 2 | O | D12 | BB | D12 | BB |
| 21B | SWF 3 | O | D11 | BB | D11 | BB |
| 22B | SWF 4 | O | D10 | BB | D10 | BB |
| 23B | SWF 5 | O | D9 | BB | D9 | BB |
| 24B | SWF 6 | O | D8 | BB | D8 | BB |
| 25B | SWF 7 | O | nSRQ-9 | SI | User-30 | |
| 26B | ST4-5 | O | nBSel-9 | SO | User-31 | |
| 27B | ST4-6 | O | nSRQ-10 | SI | User-32 | |
| 28B | IW 0 | I | nBSel-10 | SO | User-33 | |
| 29B | IW 1 | I | nSRQ-11 | SI | User-34 | |
| 30B | IW 2 | I | nBSel-11 | SO | User-35 | |
| 31B | IW 3 | I | nSRQ-12 | SI | User-36 | |
| 32B | ST4-7 | S | nBSel-12 | SO | DGND | |

| PIN | NG-IFK | T | SCU-MA | T | SCU-Slave | |
|-----|--------|----|-----------|----|-----------|----|
| 1C | /S-INL | S | +5V | | +5V/2A | |
| 2C | -15V | | -15V | | -15V / 1A | |
| 3C | A 1 | SI | IFK-A1 | SI | DGND | |
| 4C | A 3 | SI | IFK-A3 | SI | User-2 | |
| 5C | A 5 | SI | IFK-A5 | SI | User-5 | |
| 6C | A 7 | SI | IFK-A7 | SI | User-8 | |
| 7C | STS 1 | SI | Master-14 | S | User-11 | |
| 8C | STS 3 | SI | Master-17 | SI | User-14 | |
| 9C | STS 5 | SI | One-Wire | SO | User-17 | |
| 10C | STS 7 | SI | nSRQ-3 | SI | User-20 | |
| 11C | nIACK | SI | nBSel-3 | SO | User-23 | |
| 12C | FC 1 | SO | nSRQ-6 | SI | User-26 | |
| 13C | FC 3 | SO | nBSel-6 | SO | User-29 | |
| 14C | FC 5 | SO | DGND | S | DGND | |
| 15C | FC 7 | SO | nSRQ-7 | SI | Boardsel. | I |
| 16C | nc | - | nc | - | Spare1 | |
| 17C | IW-GNA | S | nDtack | BI | nDtack | oK |
| 18C | SW 1 | O | nBSsel-7 | SO | nSRQ | O |
| 19C | SW 3 | O | nSRQ-8 | SI | DGND | |
| 20C | SW 5 | O | A15 | BO | A15 | I |
| 21C | SW 7 | O | A13 | BO | A13 | I |
| 22C | SW 9 | O | A11 | BO | A11 | I |
| 23C | SW 11 | O | A9 | BO | A9 | I |
| 24C | SW 13 | O | A7 | BO | A7 | I |
| 25C | SW 15 | O | A5 | BO | A5 | I |
| 26C | IW 5 | I | A3 | BO | A3 | I |
| 27C | IW 7 | I | A1 | BO | A1 | I |
| 28C | IW 9 | I | D7 | BB | D7 | BB |
| 29C | IW 11 | I | D5 | BB | D5 | BB |
| 30C | IW 13 | I | D3 | BB | D3 | BB |
| 31C | IW 15 | I | D1 | BB | D1 | BB |
| 32C | GNA | - | AGND | | AGND | |

I = Input, O = Output, B = BiDir, S = Switch, SI = Switch-Input, SO = Switch-Output, BI= Bus-Input, BO = Bus-Out, BB = Bus-BiDir, oK = Open-Koll.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|--------|
| A | B | C | D | E | F | G | H | I | Switch |
|---|---|---|---|---|---|---|---|---|--------|

8.2 Signalbeschreibung:

| Signale für alle Betriebsarten | | |
|--|-----------|--|
| A[7..0] | I | 8 Bit Interface-Karten-Adresse (Interne Pullup-Widerstände (4k7)) |
| STS[7..0] | I | 8 Bit Geräte-Status (Interne Pulldown-Widerstände (10k)) |
| FC[7..0] | O | 8 Bit Funktions-Code |
| FC-Strobe | O | Funktions-Code-Strobe, positiver Puls (write = 1 µs, read = 5 µs) |
| FC-ST-Piggy | O | Funktions-Code-Strobe, positiver Puls (write = 1 µs, read = 5 µs) |
| Send-Data | O | Send-Data des Decoders 6408, zeigt die Übertragung der Daten an (high-aktiv) |
| /Sel ModBus | I | Hardware-Selekt zur Umschaltung auf Modul-Bus-Mode (low-aktiv). |
| /Pu.-Reset | O | Powerup-Reset (low-aktiv) |
| DRQ | O | invertiertes Interlocksignal /DRQ (high-aktiv) |
| New-Data | O | Übernahmepuls für neue Stützpunktdaten (24 Bit) vom FG (high-aktiv) |
| IRQ | O | Interrupt zum Gerät, wird vom FC-Strobe gesetzt (high-aktiv) |
| ACK | I | Quittungssignal für Geräte-Interrupt (high-aktiv) |
| AnOut | O | Verbindung zum Stecker STPiggy2 (analoger Sollwert) |
| AnOutS | O | Verbindung zum Stecker STPiggy2 (Schirm des analogen Sollwertes) |
| AnIn | I | Verbindung zum Stecker STPiggy2 (analogen Istwert) |
| AnInS | I | Verbindung zum Stecker STPiggy2 (Schirm des analogen Istwertes) |
| Output (frei) | O | Reserve Output vom ALTERA |
| Input (frei) | I | Verbindung zum Stecker STPiggy4-Pin3 und Reserve Input zum ALTERA |
| | | |
| | | |
| | | |
| | | |
| I²C-LDA | IO | Daten für den I2C-Bus |
| I²C-LCL | O | Clock für den I2C-Bus |
| Signale, die beim "Busbetrieb" inaktiv sind | | |
| /S-INL | I | S-Interlock 5-15V,(offen, oder > 4,0V = Interlock). |
| /DRQ | I | Interlock Data-Request (low-aktiv) |
| /DRDY | I | Interlock Data-Ready (low-aktiv) |
| SW[15..0] | O | 16 Bit Sollwert (beim FG, SW[15..0] = SW-FG[23..8]) |
| SWF[7..0] | O | 8 Bit Sollwerterw. auf 24Bit beim FG, SWF[7..0] = SW-FG[7..0]) |
| IW[15..0] | I | 16 Bit Istwert |
| /Send-Enable | I | Sender-Enable für den Blockmode (low-aktiv) |
| /Send-STR | I | Send-Strobe für den Blockmode (low-aktiv). |

8.3 *Netzgeräte-Backplane*

8.4 *Modul-Bus-Backplane*

8.5 *SCU-Bus-Backplane*

8.6 Produktionsunterlagen

Die Fertigungsunterlagen für die **Platine** wurden mit dem CAD-Programm "ORCAD" unter Windows-XP erstellt. Die aktuelle Version der Interface-Karte hat die Versions-Nr. 8. Das Projekt wurde im ORCAD-System unter dem Job-Namen **FG380221_IFA8** angelegt.

Verzeichnis: CadenceEDA auf "Winfilesva\CAD\$Group"
K:\GsiJob\FG38022x_IFA\FG380221_IFA8

Das **Manual** (FG380221_Manual.doc) der IF-Karte wurde in MS-WORD geschrieben und ist im

Verzeichnis: Belab on "Winfilesve\Belab\$Root"
Z:\MANUALS\BAUGRUPPEN_Geraete\FG380\22X\221\Doku\FG380221_Manual.doc
abgelegt.

Die Fertigungsunterlagen für die **Frontplatte** und die **Leuchtdiodenabdeckung** wurden mit dem Programm "IsyCAD 3.0" erstellt.

Die Zeichnungen sind auch als PDF-, DXF- DWG- Datei sind unter

Verzeichnis: Belab on "Winfilesve\Belab\$Root"
Z:\MANUALS\BAUGRUPPEN_Geraete\FG380\22X\221\Mechanik
abgelegt.

Die **Design-Unterlagen für die ALTERA-EPLD's (Loader und IFA)** sind unter:

Verzeichnis: Belab on "Winfilesve\Belab\$Root"
Z:\MANUALS\BAUGRUPPEN_Geraete\FG380\22X\221\Firmware\PLD\
IFA8_V1_R0_2008_11_24 (Version 1, Revision 0 vom 24.11.2009)

abgelegt.