



TFS - Transversales Feedback-System SIS

Gerätemodell und Softwareentwurf

L. Hechler

Gerätemodell und Softwareentwurf des Transversalen Feedback-Systems am SIS.

Ein ? an der Seite weist auf eine bisher noch ungeklärte Frage hin.

Ein tbs an der Seite weist auf eine Stelle der Dokumentation hin, die noch ausgeführt oder vervollständigt werden muss.

Änderungsprotokoll			
Datum	GM-Version	Name	Kommentar
24. 03. 2000	TFS_01	LH	Rahmen erstellt
30. 03. 2000	TFS_01	LH	Erster Ansatz für Teil I
25. 01. 2001	TFS_01	RP	Erweiterung
Nov. 2001	–	MK	Überarbeitete und erweiterte T _E X-Version, die sowohl in PostScript als auch in HTML konvertiert werden kann.

Inhaltsverzeichnis

I	Das Gerätemodell	7
1	Die Aufgabe des Gerätes	7
2	Die Hardware des Gerätes	7
3	Die Schnittstelle zum Gerät	7
3.1	Hardware Verstärker	7
3.2	Hardware Relaissteuerung	8
3.3	Interlock Interrupt	9
3.4	Data Request (DRQ) Interrupts	9
3.5	Data Ready (DRD) Interrupts	9
3.6	Funktionscodes der Interfacekarte	9
3.7	Funktionscodes der 32Bit-IO-Karte	9
4	Die Ansteuerung des Gerätes	10
4.1	Einordnung in das Timing	10
4.2	Festlegung von Startwerten	10
4.2.1	Kaltstarts	10
4.2.2	Warmstarts	10
4.3	Handbetrieb	10
4.4	Verhalten bei Störungen	10
4.4.1	Geräteinterlock	10
4.4.2	Event-Sequenzfehler	10
4.4.3	Event-Overrun	11
4.4.4	Emergency-Event	11
4.4.5	Ausfall der Kommunikation EC – Gerät	11
4.5	Bedienungsfehler vom Operating	11
5	Therapiebetrieb	11
6	Die Repräsentation des Gerätes	11
6.1	Kennzeichnung des Gerätemodells	11
6.2	Die Master-Properties	11
6.2.1	INFOSTAT	12
6.2.2	INIT	13
6.2.3	POWER	13
6.2.4	RESET	14
6.2.5	STATUS	14
6.2.6	VERSION	15
6.2.7	RELAIS	16
6.3	Die Slave-Properties	16
6.3.1	ACTIV	16
6.3.2	COPYSET	16
6.3.3	EQMERROR	17
6.3.4	DYNSTAT	17
6.3.5	WORKMODE	18
II	Die Gerätesoftware	19

7	Softwareentwurf	19
8	Lokale Datenbasis	19
9	Dualport-RAM	19
9.1	Master-Daten	19
9.2	Slave-Daten	19
10	USRs - User Service Routines	19
10.1	Obligatorische USRs	19
10.2	Gerätespezifische USRs	19
10.2.1	W_Relais	19
10.2.2	R_Relais	20
10.2.3	W_Workmode	20
10.2.4	R_Workmode	20
10.2.5	R_Dynstat	20
10.3	Globale Routinen	20
11	EQMs - Equipment Modules	20
11.1	Interne Zustände	20
11.1.1	Bedeutung der internen Zustände	20
11.1.2	Übergänge zwischen den Zuständen	20
11.1.3	Standard-Zustandsübergänge	21
11.2	Eventkonnectierte EQMs	21
11.2.1	Workmode_EQM	21
11.2.2	Dynstat_EQM	21
11.3	Periodisch konnectierte EQMs	21
11.3.1	Update_Config_EQM	21
11.4	An externe Interrupts konnectierte EQMs	22
11.4.1	Interlock_EQM	22
11.4.2	DRD_EQM	22
11.4.3	DRQ_EQM	22
11.5	Kommandogetriggerte EQMs	22
11.5.1	Dev_Init_EQM	22
11.5.2	Dev_Reset_EQM	22
11.5.3	Status_EQM	23
11.5.4	Active_EQM	23
11.5.5	Power_EQM	23
11.5.6	SetMedDataS_EQM	23
11.5.7	GetMedDataS_EQM	24
11.5.8	MedDataI_EQM	24
11.6	EQMs für die Diagnose vor Ort	24
11.6.1	Display_DPR_EQM	24
11.6.2	Display_DevErr_EQM	24
11.7	Globale Routinen	24
11.7.1	Read_and_Update_Status	24
11.7.2	Do_Intr_Service_Prep	25
11.8	Therapie-Routinen	25
11.8.1	CheckPowerState	25
11.9	Sonstige Routinen	25
11.9.1	Startup_EQM	25
11.9.2	UserIni	25

11.10MIL-Treiber	25
A Statusbits der DSPs	26
Literatur	26
Index	27

Teil I

Das Gerätemodell

1 Die Aufgabe des Gerätes

Am Schwerionensynchrotron der GSI soll durch ein Intensitätserhöhungsprogramm die Zahl der zu beschleunigenden Schwerionen um bis zu 2 Größenordnungen erhöht werden. Mit dieser Maßnahme verbunden ist das Auftreten von größeren Raumladungskräften als bisher. Diese können unter anderem dazu führen, dass kohärente, transversale Strahlschwingungen angeregt werden. Neben einer Beeinträchtigung der Strahlqualität können diese Instabilitäten zu erheblichen Strahlverlusten führen. Um diese unerwünschten kohärenten Strahlschwingungen zu dämpfen, soll ein transversales Feedbacksystem aufgebaut werden, welches sinnvollerweise nur im Hochstrombetrieb eingesetzt werden soll.

Das Feedbacksystem besteht im wesentlichen aus den vier Hauptkomponenten

- Positionssonden zur Detektion der Strahlablage,
- Digitale Signalprozessoren (DSP's),
- Leistungsverstärker und
- Exciter

Das Gerätemodell TFS realisiert nur einen Teil des Transversalen Feedback-Systems SIS. Der wesentliche Teil der Steuerung erfolgt mit Funktionsgeneratoren, die die Sollwerte für die DSPs vorgeben und durch die Gerätesoftware FG bedient werden. Die DSPs liefern sechs dynamische Statusbits, die in Kapitel A auf Seite 26 dargestellt sind.

Zum Gesamtsystem siehe die Beschreibungen in [1] und [2].

2 Die Hardware des Gerätes

Mit dem Gerätemodell TFS werden pro Gerät zwei Verstärker sowie eine Relais-Steuerung mit einem Relais bedient.

Die Geräte müssen nicht mit Sollwerten (im herkömmlichen Sinn) versorgt werden¹ und liefern keine Istwerte. Lediglich wenige Funktionen müssen ausgeführt sowie der Status gelesen werden.

Es gibt zwei Geräte, je eins für horizontales und vertikales TFS.

3 Die Schnittstelle zum Gerät

3.1 Hardware Verstärker

Jeder der beiden Verstärker liefert 4 statische und 2 dynamische Statusinformationen.

Das Statusbyte, wie es von einem Verstärker geliefert wird, ist in der folgenden Tabelle zusammengefasst.

¹Da der Betriebsmodus (working/standby) beschleuniger-spezifisch ist, muss er im Dualport-RAM gehalten und von der Gerätesoftware wie ein Sollwert behandelt werden. Einen entsprechenden Istwert gibt es nicht. Der Betriebsmodus erscheint im Status.

Bit	Name	Bedeutung	
		High (1)	Low (0)
0	Netz	ein	aus
1	Hoheit	Rechner	Hand
2	Gleichspannung	ok	Störung
3	Temperatur	ok	zu hoch
4	not used	–	–
⋮	⋮	⋮	⋮
7	not used	–	–
8	Betriebsmodus	Working	Standby
9	Verstärkereingang	Signal ok	übersteuert

‘Betriebsmodus’ und ‘Verstärkereingang’ sind die dynamischen, d. h. beschleuniger-spezifischen Bits.

Die statischen Bits der beiden Verstärker werden von der Property STATUS geliefert. Siehe Kapitel 6.2.5 auf Seite 14.

Die dynamischen Bits der beiden Verstärker werden von der Property DYNSTAT geliefert. Siehe Kapitel 6.3.4 auf Seite 17.

Der Status ‘übersteuert’ des Bits Verstärkereingang wird von der Hardware gelatcht, da es in einem Zyklus mehrmals den Status wechseln kann. Es muss am Zyklusende zurückgesetzt werden mit Hilfe der Funktion ‘Reset des Übersteuert-Bits’.

3.2 Hardware Relaissteuerung

Das Statusbyte (der Istwert), wie es von der Relaissteuerung geliefert wird, ist in der folgenden Tabelle zusammengefasst.

Bit	Name	Bedeutung	
		High (1)	Low (0)
0	Relaiskontakt Betrieb	kein Betrieb	Betrieb
1	Relaiskontakt Kalibrierung	nicht Kalibrierung	Kalibrierung
2	not used	–	–
⋮	⋮	⋮	⋮
7	not used	–	–

Die Statusbits der Relaissteuerung werden ebenso von der Property STATUS geliefert. Siehe Kapitel 6.2.5 auf Seite 14. Diese Bits erscheinen dort, im STATUS, so wie sie am Relais ausgelesen werden. (Bit 0 = Bit 24, Bit 1 = Bit 25)

Eine EINS am Relais führt dazu, daß der Relaiskontakt Kalibrierung geschlossen wird, wogegen bei einer NULL (auch Ruhestellung) der Relaiskontakt Betrieb geschlossen wird. Das Schließen des einen Kontaktes führt also automatisch zum Öffnen des anderen Kontaktes. Bit 1 und Bit 0 haben also im Normalfall unterschiedliche Stellung.

Die LED-Anzeige auf der Anpaßkarte zeigt jeweils den EINS-Zustand an.

3.3 Interlock Interrupt

Interlocks werden gepollt. Sie treten auf bei Übertemperatur oder DC-Fehler eines Verstärkers. Ein Interlock kann nur durch einen Reset zurückgesetzt werden.

3.4 Data Request (DRQ) Interrupts

DRQ-Interrupts werden nicht erwartet und verarbeitet.

3.5 Data Ready (DRD) Interrupts

DRD-Interrupts werden nicht erwartet und verarbeitet.

3.6 Funktionscodes der Interfacekarte

Funktionscode		Modus	Bedeutung
Name	Hex		
ifb_reset	01	Funktion	Reset
ifb_power_on	02	Funktion	Netz einschalten
ifb_power_off	03	Funktion	Netz ausschalten
ifb_intr_mask	12	Schreiben	Interruptmaske
ifb_dev_fct_1	14	Funktion	Übersteuert-Bits Resetpuls ein
ifb_dev_fct_6	19	Funktion	Übersteuert-Bits Resetpuls aus
ifb_pol_plus	04	Funktion	Betriebsmodus Working
ifb_pol_min	05	Funktion	Betriebsmodus Standby
ifb_rdstat	C0	Lesen	Gerätestatus lesen
ifb_rdstat_int	C9	Lesen	Status der Interfacekarte
ifb_rdstat_int_3	CC	Lesen	Version der Interfacekarte

3.7 Funktionscodes der 32Bit-IO-Karte

Funktionscode		Modus	Bedeutung
Name	Hex		
fct_io32_w_kanal_0	30	Schreiben	Kanal 0 schreiben
fct_io32_reset	32	Funktion	Reset Modulbuskarte 32Bit
fct_modadr_ID	B0	Lesen	Modul-ID lesen
fct_modadr_scal	B1	Lesen	Modul-Skalierung lesen
fct_io32_r_kanal_1	B4	Lesen	Kanal 1 lesen
fct_io32_stsAPK0	B5	Lesen	Status-Regi. APK Kanal 0 lesen
fct_io32_stsAPK1	B6	Lesen	Status-Regi. APK Kanal 1 lesen

4 Die Ansteuerung des Gerätes

Hier wird beschrieben, wie das Gerät (die Hardware) angesteuert werden muss. Das beinhaltet die Anforderungen *vom* Gerät als auch die Anforderungen *an* das Gerät.

4.1 Einordnung in das Timing

Am Zyklusanfang (Event `Start_Cycle`) müssen die Verstärker auf `working` oder `standby` geschaltet werden.

Nach der Extraktion (Event `Extr_End`) muß der dynamische Status gelesen und anschließend das Übersteuert-Bit zurückgesetzt werden.

4.2 Festlegung von Startwerten

4.2.1 Kaltstarts

Bei einem Kaltstart werden folgende Aktionen durchgeführt:

- Es wird ein Gerätereset durchgeführt.
- Das Gerät wird für alle virtuellen Beschleuniger auf `standby` geschaltet.
- Das Gerät wird für alle virtuellen Beschleuniger inaktiv geschaltet.
- Die Interlockbehandlung wird aktiviert.
- Die SE wird in den Eventmode-Betrieb geschaltet.
- Die Standard-Eventkonn timerungen werden gesetzt.

4.2.2 Warmstarts

Bei einem Warmstart werden folgende Aktionen durchgeführt:

- Es wird ein Gerätereset durchgeführt.
- Die Interlockbehandlung wird aktiviert.

4.3 Handbetrieb

Das Gerät kann auf Handbetrieb (`local`) geschaltet werden. Der Zustand wird im Status angezeigt.

4.4 Verhalten bei Störungen

4.4.1 Geräteinterlock

Bei einem Interlock geht die Gerätesoftware in den internen Zustand `dev_interlock` und liest den aktuellen Gerätestatus. Weitere Aktionen sind nicht notwendig.

4.4.2 Event-Sequenzfehler

Event-Sequenzfehler werden gemeldet, der aktuelle Zyklus aber nicht abgebrochen.

4.4.3 Event-Overrun

Die Ansteuerung des Gerätes ist nicht zeitkritisch. Im Overrun-Fall werden die normalen Aktionen ausgeführt, aber trotzdem ein Event-Overrun als Warnung gemeldet.

4.4.4 Emergency-Event

Im Emergency-Fall geht die Gerätesoftware in den internen Zustand `emergency` und liest den aktuellen Gerätestatus. Weitere Aktionen sind nicht notwendig.

4.4.5 Ausfall der Kommunikation EC – Gerät

Der Ausfall der Kommunikation zwischen EC und Gerät führt zu Timeouts. Ein Timeout wird als Fehler gemeldet, führt aber nicht zum Abbruch des aktuellen Zyklus. Aktionen die später im Zyklus noch folgen werden dadurch kaum beeinflusst.

4.5 Bedienungsfehler vom Operating

Ungültige Werte werden zurückgewiesen.

5 Therapiebetrieb

Im Therapiebetrieb darf das Gerät ein oder aus sein. Ist es ein, dann muß der workmode der Verstärker auf standby stehen.

6 Die Repräsentation des Gerätes

Dieses Kapitel beschreibt, wie das Gerät nach höheren Ebenen hin abgebildet wird.

6.1 Kennzeichnung des Gerätemodells

Das Gerätemodell hat die Bezeichnung **TFS**.

Die Gerätemodellnummer ist 35_{dez} (bzw. 23_{Hex}).

6.2 Die Master-Properties

In der Tabelle angegeben sind Name und Klasse der Property, Anzahl und Typ der Parameter, Anzahl und Typ der Daten sowie die physikalisch technische Einheit der Daten und der zugehörige Exponent zur Basis 10.

INFOSTAT, INIT, POWER, RESET, STATUS und VERSION sind obligatorische Properties, die jedes Gerätemodell hat.

Property	Klasse	Parameter		Daten		Größe	
		Anz.	Typ	Anz.	Typ	Einh.	Exp.
INFOSTAT	RA	0	–	25	BitSet32	1	0

Forts. auf nächster Seite

Forts. von letzter Seite

Property	Klasse	Parameter		Daten		Größe	
		Anz.	Typ	Anz.	Typ	Einh.	Exp.
INIT	N	0	–	0	–	–	–
POWER	R/W	0	–	1	BitSet16	1	0
RESET	N	0	–	0	–	–	–
STATUS	R	0	–	1	BitSet32	1	0
VERSION	RA	0	–	36	BitSet8	1	0
RELAIS	R	0	–	2	BitSet16	1	0
RELAIS	W	0	–	1	BitSet16	1	0

6.2.1 INFOSTAT

Bedeutung: Diese Property liefert einige wichtige Geräteinformationen in einem Zugriff. Die Informationen werden direkt aus dem Dualport-RAM gelesen, also ohne den expliziten Aufruf eines EQMs, und sind daher in der Abarbeitung nicht abhängig von Kommandoevents.

Parameter: Keine.

Daten: Die 25 Langworte enthalten im Einzelnen:

- 1: Gerätestatus (wie in der Property STATUS)
- 2: Gibt in den oberen 16 Bits an, welcher virtuelle Beschleuniger aktiv gesetzt ist (ein Bit pro Beschleuniger). Das niederwertigste Bit (Bit 16) gibt den Beschleuniger 15 an, das Bit 31 den Beschleuniger 0. Die unteren 16 Bit sind nicht verwendet. Dabei bedeutet Null, daß der Beschleuniger inaktiv ist und Eins, daß der Beschleuniger aktiv ist.
- 3: Master-Fehler. Hier ist derjenige Master-Gerätefehlercode mit dem schwersten Fehlergrad eingetragen. Bei mehreren Fehlern mit dem gleichen Fehlergrad wird der erste eingetragen, der gefunden wurde.
- 4: Slave Fehler für virtuellen Beschleuniger 0. Entsprechend dem Master-Fehler wird hier der nach dem Fehlergrad schwerste Slave-Gerätefehlercode für den Beschleuniger 0 eingetragen.
- 5: Entsprechend Punkt 4, aber für virtuellen Beschleuniger 1.
- ⋮
- 19: Entsprechend Punkt 4, aber für virtuellen Beschleuniger 15.
- 20: EC-Mode. In den oberen 16 Bit des Langwortes steht der von der Geräte-software eingestellte Default-EC-Mode, in den unteren 16 Bit der aktuelle EC-Mode. Folgende Modi sind definiert:
 - 0: *not set*
 - 1: *Preset_Command* Der ECM hat das Umschalten in Command-Mode vorbereitet aber noch nicht beendet.
 - 2: *Command* Der ECM läuft im Command-Mode.
 - 3: *Preset_Event* Der ECM hat das Umschalten in Event-Mode vorbereitet aber noch nicht beendet.
 - 4: *Event* Der ECM läuft im Event-Mode.

- 21:** EC-Performance-Mode. In den oberen 16 Bit des Langwortes steht der von der Gerätesoftware eingestellte Default-Performance-Mode, in den unteren 16 Bit der aktuelle Performance-Mode. Folgende Modi sind definiert:
- 0:** *not set*
 - 1:** *Display* Der ECM läuft im Display-Mode.
 - 2:** *Preset_Turbo* Der ECM hat das Umschalten in den Turbo-Mode vorbereitet aber noch nicht beendet.
 - 3:** *Turbo* Der ECM läuft im Turbo-Mode.
- 22:** HW_Warning_Maske. Die 32 Bits geben an aus welchen Bits im Gerätestatus das HW-Warning-Bit im Status abgeleitet wird.
- 23:** Pulszentralen-Identifikation:
- 0:** TIF
 - 1:** SIS-PZ
 - 2:** ESR-PZ
 - 3...6:** undefiniert
 - 7:** Software-PZ
 - 8:** UNILAC, Master-PZ
 - 9:** UNILAC-PZ 1
 - 10:** UNILAC-PZ 2
 - 11:** UNILAC-PZ 3
 - 12:** UNILAC-PZ 4
 - 13:** UNILAC-PZ 5
 - 14:** UNILAC-PZ 6
 - 15:** UNILAC-PZ 7
- 24:** Reserviert für Erweiterungen.
- 25:** Reserviert für Erweiterungen.

6.2.2 INIT

Bedeutung: Initialisierung des Gerätes (Kaltstart). Für die dabei durchzuführenden Aktionen siehe Kapitel 4.2.1 auf Seite 10.

Parameter: Keine.

Daten: Keine.

6.2.3 POWER

Bedeutung: Gibt an, ob beide Verstärker ein- oder ausgeschaltet sind bzw. werden sollen.

Parameter: Keine.

Daten: Das Datum kann nur zwei Werte annehmen. Eins heißt, beide Verstärker sind eingeschaltet bzw. sollen eingeschaltet werden. Null heißt, mindestens ein Verstärker ist ausgeschaltet bzw. die Verstärker sollen ausgeschaltet werden.

6.2.4 RESET

Bedeutung: Reset des Gerätes (Warmstart). Für die dabei durchzuführenden Aktionen siehe Kapitel 4.2.2 auf Seite 10.

Parameter: Keine.

Daten: Keine.

6.2.5 STATUS

Bedeutung: Auslesen des Gerätestatus.

Parameter: Keine.

Daten: Das 32 Bits breite Statuswort. Die statischen Bits des Status der beiden Verstärker (Kapitel 3.1 auf Seite 7) sowie die zwei Statusbits der Relais (Kapitel 3.2 auf Seite 8) werden wie folgt dem Statuswort zugeordnet:

Die Bits 0 bis 7 sind die systemweit einheitlichen Software-Statusbits. Einen Schalter zur Hand Bedienung gibt es im Moment nicht. Das Bit 1 liegt derzeit also ständig auf eins.

Jeder der beiden Verstärker liefert 4 statische Statusinformationen. Die Stellung der beiden Relais wird ebenfalls im Status angezeigt.

Die Statusbits im Einzelnen sind in der folgenden Tabelle zusammengefasst.

Bit	Name	Bedeutung	
		High (1)	Low (0)
0	Netz	ein	aus
1	Hoheit	Rechner	Hand
2	(reserved)	–	–
3	(reserved)	–	–
4	Emergency	no	yes
5	Interlock	no	yes
6	HW Warning	no	yes
7	SW Warning	no	yes
8	V1 Netz	ein	aus
9	V1 Hoheit	Rechner	Hand
10	V1 Gleichspannung	ok	Störung
11	V1 Temperatur	ok	zu hoch
12	not used	–	–
⋮	⋮	⋮	⋮
15	not used	–	–
16	V2 Netz	ein	aus
17	V2 Hoheit	Rechner	Hand
18	V2 Gleichspannung	ok	Störung
19	V2 Temperatur	ok	zu hoch
20	not used	–	–
⋮	⋮	⋮	⋮
23	not used	–	–

Forts. auf nächster Seite

Forts. von letzter Seite

Bit	Name	Bedeutung	
		High (1)	Low (0)
24	Relaiskontakt Betrieb	kein Betrieb	Betrieb
25	Relaiskontakt Kalibrierung	nicht Kalibrierung	Kalibrierung
26	id von 32-Bit-I/O	ok	falsch
27	not used	–	–
⋮	⋮	⋮	⋮
29	not used	–	–
30	id von APK-1	ok	falsch
31	id von APK-0	ok	falsch

Das Bit 0 zeigt nur dann ‘Netz ein’ an, wenn *beide* Verstärker eingeschaltet sind.
Das Bit 1 zeigt nur dann ‘Hoheit Rechner’ an, wenn *beide* Verstärker auf Rechnersteuerung geschaltet sind.

Bit 24 steht auf EINS und Bit 25 auf Null, wenn der Kalibriermode gewählt wurde, im Betriebsmode hingegen steht Bit 25 auf EINS und Bit 24 auf NULL. Dadurch weist der Status immer eine NULL auf.

Eine Hardware-Warnung liegt vor, wenn eines der folgenden Bits den angegebenen Status anzeigt:

Bit	Name	Status
10	V1 Gleichspannung	Störung
11	V1 Temperatur	zu hoch
18	V2 Gleichspannung	Störung
19	V2 Temperatur	zu hoch
24+25	Relaiskontakte undefiniert	kein Betrieb und nicht Kalibrierung
24+25	Relaiskontakte undefiniert	Betrieb und Kalibrierung
26	id von 32-Bit-I/O	falsch
30	id von APK-1	falsch
31	id von APK-0	falsch

Bit 24 und Bit 25 führen zu einer HW-Warnung, wenn *sie beide* Null sind.

6.2.6 VERSION

Bedeutung: Lesen der Versionskennung der Gerätesoftware.

Parameter: Keine.

Daten: Versionskennung als ASCII-String, pro Datum ein ASCII-Zeichen.

Bytes	Inhalt
1...12	Version der USRs
13...24	Version der EQMs
25...36	Version des Standard-MIL-Treibers
37...48	Variante der EQMs

6.2.7 RELAIS

Bedeutung: Lesen der Relaisstellung oder Schreiben des Relaismode.

Parameter: Keine.

Daten: Die Anzahl der Daten ist beim Lesen und Schreiben gleich. Ein Datum kann nur zwei Werte annehmen. NULL ist die Kalibrierstellung des Relais. EINS ist die Betriebsstellung des Relais.

Schreiben: 1 Datum gibt den gewünschten Mode und somit die gewünschte Stellung des Relais an.

Lesen: 1 Datum gibt die aktuelle Stellung des Relais an.

6.3 Die Slave-Properties

In der Tabelle angegeben sind Name und Klasse der Property, Anzahl und Typ der Parameter, Anzahl und Typ der Daten sowie die physikalisch technische Einheit der Daten und der zugehörige Exponent zur Basis 10.

Liefert eine Property z. B. Milliampere (mA), dann ist die physikalisch technische Einheit „A“ und der Exponent „-3“.

ACTIV, COPYSET und EQMERROR sind obligatorische Properties, die jedes Gerätemodell hat.

Property	Klasse	Parameter		Daten		Größe	
		Anz.	Typ	Anz.	Typ	Einh.	Exp.
ACTIV	R/W	0	-	1	BitSet16	1	0
COPYSET	W	0	-	1	BitSet16	1	0
EQMERROR	RA	217	Integer32	348	Integer32	1	0
DYNSTAT	R	0	-	1	BitSet16	1	0
WORKMODE	R	0	-	2	BitSet16	1	0
WORKMODE	W	0	-	1	BitSet16	1	0

6.3.1 ACTIV

Bedeutung: Gibt an, ob das Gerät für den zugehörigen virtuellen Beschleuniger an der Puls-zu-Puls-Modulation teilnehmen soll bzw. teilnimmt.

Parameter: Keine.

Daten: Das Datum kann nur zwei Werte annehmen. Null heißt, das Gerät nimmt für den zugeordneten Beschleuniger *nicht* an der PPM teil bzw. soll *nicht* an der PPM teilnehmen. Eins heißt, das Gerät nimmt für den zugeordneten Beschleuniger an der PPM teil bzw. soll an der PPM teilnehmen.

6.3.2 COPYSET

Bedeutung: Kopiert alle Geräteeinstellungen (Sollwerte) eines („fremden“) virtuellen Beschleunigers in den zugehörigen („eigenen“) virtuellen Beschleuniger.

Parameter: Keine.

Daten: Nummer des („fremden“) virtuellen Beschleunigers, von dem die Einstellungen (Sollwerte) kopiert werden sollen.

Bemerkung: Die Nummer des zugehörigen („eigenen“) virtuellen Beschleunigers, in den die Geräteeinstellungen kopiert werden sollen, wird dem Standard entsprechend im XSR-Header geliefert.

6.3.3 EQMERROR

Bedeutung: Fehlermeldungen der auf der SE installierten Gerätesoftware. Es werden die aktuellen Fehlermeldungen sowohl für die Masterfehler als auch für die Slavefehler der Geräteebene geliefert. Dazu wird auch der Inhalt des Fehlerpuffers zurückgegeben, in dem die letzten aufgetretenen Fehler abgespeichert wurden.

Parameter: Keine.

Daten: Die Anzahl der Fehlermeldungen sei bezeichnet durch:

- m Zahl der Master-Fehlermeldungen
- s Zahl der Slave-Fehlermeldungen
- b Größe des Fehlerpuffers

Weiterhin soll gelten:

$$l = m + s$$

$$t = m + s + b$$

Die Daten im Einzelnen:

- 1 : In den unteren beiden Bytes sind die Anzahl der Master-Fehlermeldungen m und die Anzahl der Slave-Fehlermeldungen s angegeben:

0	0	s	m
---	---	---	---

- 2 : erste Master-Fehlermeldung

⋮

- $m + 1$: letzte Master-Fehlermeldung

- $m + 2$: erste Slave-Fehlermeldung

⋮

- $l + 1$: letzte Slave-Fehlermeldung

- $l + 2$: Länge b des Fehlerpuffers

- $l + 3$: Zahl der Einträge im Fehlerpuffer

- $l + 4$: Index des ersten freien Platzes im Fehlerpuffer (der Fehlerpuffer ist ein Ringpuffer)

- $l + 5$: Erster Speicherplatz im Fehlerpuffer

⋮

- $t + 4$: Letzter Speicherplatz im Fehlerpuffer

6.3.4 DYNSTAT

Bedeutung: Lesen des beschleuniger-spezifischen, dynamischen Gerätestatus der Verstärker. (Gelesen wird aus dem DPR, wo diese Daten am Ende jedes Zyklus abgelegt werden.)

Parameter: Keine.

Daten: Das 16 Bits breite Statuswort. Die dynamischen Bits des Status der beiden Verstärker (Kapitel 3.1 auf Seite 7) werden wie folgt dem Statuswort zugeordnet:

Bit	Name	Bedeutung	
		High (1)	Low (0)
0	V1 Betriebsmodus	Working	Standby
1	V1 Verstärkereingang	Signal ok	übersteuert
2	not used	–	–
⋮	⋮	⋮	⋮
7	not used	–	–
8	V2 Betriebsmodus	Working	Standby
9	V2 Verstärkereingang	Signal ok	übersteuert
10	not used	–	–
⋮	⋮	⋮	⋮
15	not used	–	–

6.3.5 WORKMODE

Bedeutung: Gibt an, ob der Betriebsmodus der Verstärker standby oder working ist oder sein soll.

Parameter: Keine.

Daten: Die Anzahl der Daten ist unterschiedlich, je nachdem ob geschrieben oder gelesen wird. Ein Datum kann nur zwei Werte annehmen. Null heißt standby. Eins heißt working.

Schreiben: 1 Datum gibt den gewünschten Betriebsmodus für beide Verstärker an.

Lesen: 2 Daten geben den aktuellen Betriebsmodus der beiden Verstärker an (Datum 1 - Verstärker 1; Datum 2 - Verstärker 2).

Teil II

Die Gerätesoftware

7 Softwareentwurf

Dies ist noch ein sehr allgemeiner Punkt. Hier sollte unter anderem hingehören:

- Datenstrukturen,
- Datenflussdiagramme,
- Kontrollflussdiagramme
- ...

8 Lokale Datenbasis

9 Dualport-RAM

9.1 Master-Daten

Es gibt keine gerätespezifischen Master-Werte im DPR

9.2 Slave-Daten

Es gibt zwei Slave-Werte im DPR:

Datum	Typ	Bedeutung
workmode	uns_word	Betriebsmodus der Verstärker
dynstat	uns_word	dynamischer Gerätestatus der Verstärker

10 USRs - User Service Routines

10.1 Obligatorische USRs

Obligatorische USRs werden hier nicht beschrieben. Sie werden als `DEFAULT_USRS.PIN` inkludiert und sind an anderer Stelle dokumentiert.

10.2 Gerätespezifische USRs

10.2.1 W_Relais

Setzen der gewünschten Position für beide Relais.

10.2.2 R_Relais

Lesen der Relaispositionen. Die Positionsdaten werden für die Relais getrennt zurückgegeben.

10.2.3 W_Workmode

Setzen des Betriebsmodus (Standby/Working) der beiden Verstärker.

10.2.4 R_Workmode

Der Betriebsmodus der Verstärker wird direkt aus der Slave-Variablen dynstat im DPR ausgelesen. Die Betriebsmoden werden für die Verstärker getrennt zurückgegeben.

10.2.5 R_Dynstat

Der dynamische Status wird direkt aus dem DPR ausgelesen.

10.3 Globale Routinen

Hier werden alle Routinen aufgeführt, die im Modul USRs global definiert sind und von verschiedenen USRs benutzt werden.

11 EQMs - Equipment Modules

11.1 Interne Zustände

11.1.1 Bedeutung der internen Zustände

Für die Gerätesoftware sind folgende interne Zustände definiert:

not_set Initzustand. Dieser Zustand sollte nie auftreten.
xxx ...

11.1.2 Übergänge zwischen den Zuständen

Legende zur Tabelle:

U:	Auslösende Ursache.	
	SI	Summeninterlock des Gerätes steht an.
	Evt_Emerg	Pulszentrale verschickte Emergency-Event.
	RESET	Reset wird per Kommando oder Knöpfchendrücken ausgelöst.
	Power=1	Power wird per Kommando eingeschaltet.
	Power=0	Power wird per Kommando ausgeschaltet.
B:	Abzuprüfende Bedingung.	
	R	Remotebit des Status steht auf Remote.
	r	Remotebit des Status steht auf Local.
	P	Powerbit des Status steht auf Power on.
	p	Powerbit des Status steht auf Power off.

- A:** Ausführende Stelle des Zustandsübergangs.
- | | |
|--------------|---|
| Status lesen | Beim periodischen (oder zumindest regelmäßigen) |
| (periodisch) | Lesen des Status. |
| x_EQM | Innerhalb des EQMs x_EQM. |
- Die Priorität der Zustände (höchste Priorität zuerst): emergency, interlock, local, power_off und power_seq, error, ready und busy.
- Liegen mehrere Bedingungen für verschiedene Zustände gleichzeitig vor (z. B. Netz aus und Gerät auf Handbetrieb), muss der jeweils wichtigste Zustand eingenommen werden.

11.1.3 Standard-Zustandsübergänge

Zur Verdeutlichung hier einige Standard-Zustandsübergänge. Sie kommen Zustände, wenn eine Sequenz ohne Fehler abläuft.

Das Gerät nimmt an der Puls-zu-Puls-Modulation teil. ...

ready -> busy -> ready

11.2 Eventkonnektierte EQMs

11.2.1 Workmode_EQM

Event: Evt_Start_Cycle(32_{dez})

Aktion: Der Betriebsmode der Verstärker wird auf „working“ oder „standby“ gesetzt.

11.2.2 Dynstat_EQM

Event: Evt_Extr_End(51_{dez})

Aktion: Der dynamische Status wird ausgelesen und anschließend das Übersteuerbit zurückgesetzt.

11.3 Periodisch konnektierte EQMs

11.3.1 Update_Config_EQM

Zeit: 60s

Anzahl: Unendlich.

Aktion: Aktualisieren der Geräteverfügbarkeit: Es wird versucht, von möglichen Geräteadressen den Status zu lesen. Erfolgt eine Reaktion, wird das Gerät als „online“ geführt.

11.4 An externe Interrupts konnektierte EQMs

11.4.1 Interlock_EQM

Interrupt: Summen-Interlock.

Aktion: Internen Zustand auf „Interlock“ setzen, falls er nicht „Emergency“ ist. Sollwert Null setzen.

11.4.2 DRD_EQM

Interrupt: Data Ready Interrupt.

Aktion: Keine. Sollte bei MS nicht vorkommen.

11.4.3 DRQ_EQM

Interrupt: Data Request Interrupt.

Aktion: Keine. Sollte bei MS nicht vorkommen.

11.5 Kommandogetriggerte EQMs

11.5.1 Dev_Init_EQM

Folgende Aktionen werden ausgeführt:

- Fehlerpuffer initialisieren.
- Reset ans Gerät schicken.
- Interlock-Service präparieren (Do_Intr_Service_Prep).
- Status vom Gerät lesen (Read_and_Update_Status).
- Konstanten in Dualport-RAM initialisieren.
- Internen Zustand setzen; mit not-set (Set_InternalState).
- Aktivzustand für alle virtuellen Beschleuniger initialisieren.
- Alle Slave-Soll- und -Istdaten initialisieren. (workmode = standby, dynstat = FFFF)
- Fehlerbehandlung.

11.5.2 Dev_Reset_EQM

Folgende Aktionen werden ausgeführt:

- Fehlerpuffer initialisieren.
- Reset ans Gerät schicken.
- Interlock-Service präparieren (Do_Intr_Service_Prep).

- Status vom Gerät lesen (Read_and_Update_Status).
- Konstanten in Dualport-RAM initialisieren.
- Internen Zustand setzen (Set_InternalState).
- Aktivzustand für alle virtuellen Beschleuniger initialisieren.
- Alle Slave-Solldaten validieren.
- Alle Slave-Istdaten initialisieren.
- Fehlerbehandlung.

11.5.3 Status_EQM

Folgende Aktionen werden ausgeführt:

- Status vom Gerät lesen (Read_and_Update_Status).
- Internen Zustand setzen (Set_InternalState).
- Fehlerbehandlung.

11.5.4 Active_EQM

Folgende Aktionen werden ausgeführt:

- Status vom Gerät lesen (Read_and_Update_Status).
- Internen Zustand setzen (Set_InternalState).
- Gerät aktiv bzw. inaktiv setzen.
- Fehlerbehandlung.

11.5.5 Power_EQM

Folgende Aktionen werden ausgeführt:

- Status vom Gerät lesen (Read_and_Update_Status).
- ...
- Fehlerbehandlung.

11.5.6 SetMedDataS_EQM

Folgende Aktionen werden ausgeführt:

- Empfangenen Therapie-Datensatz in den lokalen Speicher kopieren (SaveSlave-Set).
- Fehlerbehandlung

11.5.7 GetMedDataS_EQM

Folgende Aktionen werden ausgeführt:

- Unabhängige EFI-Parameter Null setzen
- EFI-Parameter testen
- Falls ein gültiger Datensatz angefordert wurde, diesen zurückschicken.
- Fehlerbehandlung

11.5.8 MedDataI_EQM

Folgende Aktionen werden ausgeführt:

- Angeforderten Datensatz zurückschicken (ReadActVal).
- Fehlerbehandlung

11.6 EQMs für die Diagnose vor Ort

11.6.1 Display_DPR_EQM

Parameter: Das EQM benötigt 2 Parameter.

P1: virtueller Beschleuniger (in Hex angeben)

P2: logische Gerätenummer (in Hex angeben)

Daten: Keine.

Aktion: Zeigt am Bildschirm vor Ort die wichtigsten Daten aus dem DPRAM für das gewählte Gerät und den gewählten virtuellen Beschleuniger an.

11.6.2 Display_DevErr_EQM

Parameter: Das EQM benötigt 2 Parameter.

P1: virtueller Beschleuniger (in Hex angeben)

P2: logische Gerätenummer (in Hex angeben)

Daten: Keine.

Aktion: Zeigt am Bildschirm vor Ort die Fehlercodes aus der Datenstruktur im Dualport-RAM für das gewählte Gerät und den gewählten virtuellen Beschleuniger an.

11.7 Globale Routinen

11.7.1 Read_and_Update_Status

Liest den Hardware-Status des Gerätes, leitet die Software-Statusbits daraus ab und schreibt den neuen Status ins Dualport-RAM. Bei einer Statusänderung wird automatisch ein Statusalarm verschickt.

11.7.2 Do_Intr_Service_Prep

Setzt das Summeninterlock-Display im Dualport-RAM zurück und ermöglicht den Interlock-Interrupt auf der Inetrface-Karte.

11.8 Therapie-Routinen

Da das Gerät immer auf activ steht, gibt es kein SetActiveState-EQM.

Werden Therapie-EQMs inkludiert, dann müssen die EQMs einige sogenannte Callback-Routinen zur Verfügung stellen, die von den Therapie-EQMs aufgerufen werden. Definitionen für den Therapiebetrieb sind in Kapitel 5 beschrieben.

Es gibt keine Therapie-Slave-Properties und keine Therapie-Master-Properties.

Die Therapiedaten stehen nichtim Flash, sondern sind fest verdrahtet.

11.8.1 CheckPowerState

Wird diese Routine durch das Med-Prep-event aufgerufen, dann wird in dem Beschleuniger Me-dAcc für den workmode der Verstärker der Wert 0, also standby im DPR abgelegt, so daß die Verstärker wenn dieser Beschleuniger läuft nach standby geschaltet werden.

11.9 Sonstige Routinen

11.9.1 Startup_EQM

Installiert die Event-EQM-Konnektierung für alle virtuellen Beschleuniger (siehe hierzu auch Kapitel 4.1 auf Seite 10), konnektiert die Therapie-EQMs und schaltet die SE in den Event-Mode.

11.9.2 UserIni

Initialisiert die Dualport-RAM-Konstanten, setzt die periodischen Aufträge auf und macht die EQMs dem ECM bekannt.

11.10 MIL-Treiber

Zum Ansteuern der Interfacekarte wird der Standard-MIL-Treiber eingesetzt.

A Statusbits der DSPs

Die Statusbits der DSPs werden über den Funktionsgenerator für den Verstärkungsfaktor (V_v bzw. V_h) eingelesen. Folgende Bits sind definiert:

Bit	Name	Bedeutung	
		High (1)	Low (0)
0	Pickup-Signal	vorhanden	nicht vorhanden
1	DSP	läuft während Rampe	läuft nicht (mehr)
2	alle DSPs	laufen	min. einer läuft nicht
3	DSP-Input-Daten	vollständig vorhanden	(noch) nicht vollständig
4	Umlaufzeit	Input-Daten vorhanden	nicht vorhanden
5	Koeffizienten	vorhanden	nicht vorhanden

Literatur

- [1] U. Blell. Ein transversales Feedback-System für das SIS - Aufbau und Funktion des Gerätes. Technical report, GSI, Darmstadt, März 2000.
- [2] U. Blell. Transversales Feedback-System SIS - Vorschlag zur Anbindung an das Kontrollsystem. Technical report, GSI, Darmstadt, März 2000.
- [3] Peter Kainberger. Therapieprojekt: Die SE-Ebene. Accelerator Controls Documentation F-MBP-02, Gesellschaft für Schwerionenforschung, Darmstadt, Mai 1995. (Source: sis_med:dd_se.tex).

Index

—Symbole —

32Bit-IO-Karte 9

—A—

Active_EQM 23
Ansteuerung des Gerätes 10
Aufgabe des Gerätes 7

—B—

Bedienungsfehler 11

—D—

Datenbasis 19
Dev_Init_EQM 22
Dev_Reset_EQM 22
Display_DevErr_EQM 24
Display_DPR_EQM 24
DPR
 • Master-Daten 19
DRD-Interrupt 9
DRD_EQM 22
DRQ-Interrupt 9
DRQ_EQM 22
Dualport-RAM 19
Dynstat_EQM 21

—E—

Emergency-Event 11
EQMs 20
 • An Interrupts konnektierte 22
 • Eventkonnektierte 21
 • für die Diagnose vor Ort 24
 – Display_DevErr_EQM 24
 – Display_DPR_EQM 24
 • Globale Routinen 24
 – Do_Intr_Service_Prep 25
 – Read_and_Update_Status 24
 • Interruptkonnektierte
 – DRD_EQM 22
 – DRQ_EQM 22
 – Interlock_EQM 22
 • Kommandogetriggerte 22
 – Active_EQM 23
 – Dev_Init_EQM 22

 – Dev_Reset_EQM 22
 – GetMedDataS_EQM 24
 – MedDataL_EQM 24
 – Power_EQM 23
 – SetMedDataS_EQM 23
 – Status_EQM 23
 • MIL-Treiber 25
 • Periodisch konnektierte 21
 – Update_Config_EQM 21
 • Sonstige Routinen 25
 – Startup_EQM 25
 – UserIni 25
 • Therapie-Routinen 25
 – CheckPowerState 25

Event-Overrun 11
Event-Sequenzfehler 10
Eventkonnektierte EQMs 21
Eventkonnektierungen 10

—F—

Funktionscodes 9

—G—

Gerät

- Ansteuerung 10
- Aufgabe 7
- Hardware 7
- Repräsentation 11
- Schnittstelle 7
- Therapiebetrieb 11

Gerätemodell 7

- Kennzeichnung 11
- Master-Properties 11
- Slave-Properties 16

Gerätestatus

- Dynamischer 17
- Statischer 14

GetMedDataS_EQM 24

Globale Routinen

- der EQMs 24
- der USRs 20

—H—

Handbetrieb 10
Hardware des Gerätes 7

Hardware-Status	7, 8
Hardware-Warnung	15

—I—

Init	10
Interfacekarte	9
• MIL-Treiber	25
Interlock	9, 10
Interlock_EQM	22
Interne Zustände	20
Interrupt	
• DRD-Interrupt	9
• DRQ-Interrupt	9
• Interlock	9
Interruptkonnektierte EQMs	22

—K—

Kaltstarts	10
Kommandogetriggerte EQMs	22

—L—

Lokale Datenbasis	19
-------------------------	----

—M—

Master-Daten	19
Master-Properties	11
MedDataI_EQM	24
MIL-Treiber	25

—O—

Overrun	11
---------------	----

—P—

Periodisch konnektierte EQMs	21
Power_EQM	23
Properties	
• ACTIV	16
• COPYSET	16
• DYNSTAT	17
• EQMERROR	17
• INFOSTAT	12
• INIT	13
• Master-	11
• POWER	13
• RELAIS	16

• RESET	14
• Slave-	16
• STATUS	14
• VERSION	15
• WORKMODE	18

—R—

R_Dynstat	20
R_Relais	20
R_Workmode	20
Repräsentation des Gerätes	11
Reset	10

—S—

Schnittstelle zum Gerät	7
Sequenzfehler	10
SetMedDataS_EQM	23
Slave-Properties	16
Software-Status	14
Softwareentwurf	19
Sonstige Routinen	25
Störungen	10
• Emergency-Event	11
• Event-Overrun	11
• Event-Sequenzfehler	10
• Interlock	10
• Kommunikation EC – Gerät	11
Startup_EQM	25
Startwerte	10
Status	
• der DSPs	26
• Dynamischer -	17
• Hardware-	7, 8
• Statischer -	14
Status_EQM	23

—T—

Therapie-Routinen	25
Therapiebetrieb	11
Timing	10

—U—

Update_Config_EQM	21
UserIni	25
USRs	19
• gerätespezifische	19
– R_Dynstat	20

- R_Relais 20
- R_Workmode.....20
- W_Relais.....19
- W_Workmode.....20
- Globale Routinen.....20
- obligatorische 19

—**W**—

- W_Relais 19
- W_Workmode.....20
- Warmstarts10
- Workmode_EQM.....21

—**Z**—

Zustände

- Interne 20
 - Übergänge 20
 - Bedeutung 20
 - Standard-Übergänge 21