

MX - Gepulste Magnete

Gerätemodell und Softwareentwurf

L. Hechler, P. Kainberger

Diese Papier ist entstanden mit Hilfe eines uralten EPOS-Entwurfs. Es ist das erste Papier, das mit dem neuen Rahmen zur Beschreibung eines Gerätemodells und zum Entwurf der Gerätesoftware erstellt wurde. Obwohl es keinen Anspruch auf Vollständigkeit erhebt soll es hauptsächlich als Beispiel für weitere Gerätemodelle und Softwareentwürfe dienen.

Änderungsprotokoll			
Datum	GM-Version	Name	Kommentar
1. Mar. 88	MX_01	Kai	1. Diskussionsgrundlage
4. Mar. 88	MX_01	Eik, LH, CR, Kai	GSM-1
17. Mar. 88	MX_02	Kai	2. Diskussionsgrundlage
1. Nov. 88	MX_03	LH, Bd, Stn, Schaa	GSM-2, EQ-Modell MX_03
22. Nov. 91	MX_04	L. Hechler	Trennung Gerätemodell/SW-Entwurf
29. Jan. 92	MX_05	L. Hechler	Endgültige Version mit neuem Rahmen. Eq-Modell MX_05.
23. Aug. 93	MX_06	L. Hechler	Weitere Varianten, keine 'OLDIFB'-Besonderheit mehr
21. Feb. 01	MX_18	L. Hechler	Anpassung an L ^A T _E X ₂ e PotiInfo raus; MagnInfo, Calc rein und noch'n paar Ergänzungen
17. Feb. 03	MX_19	L. Hechler	Unterscheidung Strom-/Feldregelung
26. Apr. 07	MX_19	L. Hechler	MAGNSVCS/I und CONNECT hinzugefügt

Inhaltsverzeichnis

I	Das Gerätemodell	7
1	Die Aufgabe des Gerätes	7
2	Die Hardware des Gerätes	7
2.1	Gerätevarianten	7
3	Die Schnittstelle zum Gerät	8
3.1	Funktionscodes der Interfacekarte	8
3.2	D/A- und A/D-Wandler	8
3.3	Interlock Interrupt	9
3.4	Data Request (DRQ) Interrupts	9
3.5	Data Ready (DRD) Interrupts	9
3.6	Definition der Bits des Hardwarestatus	9
4	Die Steuerung des Gerätes	10
4.1	Aufgaben im Normalbetrieb	10
4.1.1	Soll- und Istwert	10
4.1.2	Polwender	11
4.1.3	Lastumschaltung	11
4.1.4	Einschalten	12
4.1.5	Ausschalten	13
4.2	Genauigkeitsanforderungen	13
4.3	Zeitkritische Anforderungen	13
4.4	Einordnung in das Timing	13
4.5	Festlegung von Startwerten	14
4.5.1	Kaltstarts	14
4.5.2	Warmstarts	15
4.6	Handbetrieb	15
4.7	Ableitung des Hardwarefehler-Bits aus dem Gerätestatus	15
4.8	Verhalten bei Störungen	15
4.8.1	Geräteinterlock	15
4.8.2	Event-Sequenzfehler	15
4.8.3	Event-Overrun	15
4.8.4	Emergency-Event	15
4.8.5	Ausfall der Kommunikation EC – Gerät	16
4.8.6	Sonstiges	16
4.9	Bedienungsfehler vom Operating	16
4.10	Sonstige Anforderungen	16
5	Die Repräsentation des Gerätes	16
5.1	Kennzeichnung des Gerätemodells	16
5.2	Die Master-Properties	16
5.2.1	INFOSTAT	16
5.2.2	INIT	17
5.2.3	POWER	17
5.2.4	RESET	17
5.2.5	STATUS	18
5.2.6	VERSION	18
5.2.7	CALC	18

5.2.8	CONNECT	18
5.2.9	CONSTANT	19
5.2.10	INVERTER	19
5.2.11	MAGNSVCI	20
5.2.12	MAGNSVCS	20
5.3	Die Slave-Properties	20
5.3.1	ACTIV	21
5.3.2	COPYSET	21
5.3.3	EQMERROR	21
5.3.4	CURRENTI	22
5.3.5	CURRENTS	22
5.3.6	DSTATUS	22
5.3.7	FIELDI	22
5.3.8	FIELDS	23
5.3.9	MAGNINFO	23
5.3.10	MEDDATAI	24
5.3.11	MEDDATAS	24
5.3.12	VOLTS	24
5.3.13	VOLTI	25
II Der Entwurf der Software		27
6	Softwareentwurf	27
7	Lokale Datenbasis	27
7.1	Tabelle der Konstanten	27
8	Dualport RAM	27
9	USRs - User Service Routinen	28
9.1	Obligatorische USRs	28
9.1.1	N_Init	28
9.1.2	N_Reset	28
9.1.3	R_Status	28
9.1.4	R_Power	28
9.1.5	W_Power	28
9.1.6	R_Active	28
9.1.7	W_Active	28
9.1.8	W_CopySet	28
9.1.9	R_EQMErr	28
9.1.10	R_Version	28
9.1.11	R_InfoStat	28
9.2	Gerätespezifische USRs	28
9.2.1	W_FieldS	28
9.2.2	R_FieldS	28
9.2.3	R_FieldI	28
9.2.4	W_CurrentS	29
9.2.5	R_CurrentS	29
9.2.6	R_CurrentI	29
9.2.7	W_VoltS	29
9.2.8	R_VoltS	29
9.2.9	R_VoltI	29

9.2.10	W_MagnSvcS	29
9.2.11	R_MagnSvcS	29
9.2.12	R_MagnSvcI	29
9.2.13	W_Inverter	30
9.2.14	R_Inverter	30
9.2.15	R_Constant	30
9.2.16	R_PotiInfo	30
9.3	Globale Routinen	30
10	EQMs - Equipment Module	30
10.1	Interne Zustände	30
10.1.1	Bedeutung der internen Zustände	30
10.1.2	Übergänge zwischen den Zuständen	31
10.1.3	Standard-Zustandsübergänge	31
10.2	Eventkonnektierte EQMs	31
10.2.1	CurrentS_EQM	31
10.2.2	CurrentI_EQM	33
10.2.3	Turbo_CurrentS_EQM	33
10.2.4	Turbo_CurrentI_EQM	33
10.2.5	CleanUp_EQM	33
10.2.6	Emerg_EQM	33
10.3	Periodisch konnektierte EQMs	33
10.3.1	CheckPower_EQM	33
10.3.2	CheckConnect_EQM	33
10.3.3	CheckInverter_EQM	33
10.3.4	Update_Config_EQM	34
10.4	An externe Interrupts konnektierte EQMs	34
10.4.1	Interlock_EQM	34
10.4.2	DRD_EQM	34
10.4.3	DRQ_EQM	34
10.5	Kommandogetriggerte EQMs	34
10.5.1	Dev_Init_EQM	34
10.5.2	Dev_Reset_EQM	34
10.5.3	Status_EQM	34
10.5.4	Active_EQM	34
10.5.5	Power_EQM	34
10.5.6	Inverter_EQM	34
10.5.7	MagnSvcS_EQM	35
10.5.8	MagnSvcI_EQM	35
10.6	EQMs für die Diagnose vor Ort	35
10.6.1	Display_DPR_EQM	35
10.6.2	Display_DevErr_EQM	35
10.7	Sonstige EQMs	35
10.7.1	Startup_EQM	35
10.8	Globale Routinen	35
10.8.1	Read_and_Update_Status	36
10.8.2	Set_InternalState	36
10.8.3	Do_Intr_Service_Prep	36
10.8.4	Power_on	36
10.8.5	Power_off	36
10.8.6	Set_Active_State	36
10.9	MIL-Treiber	36

11 Varianten	36
Literatur	38
Index	39

Teil I

Das Gerätemodell

1 Die Aufgabe des Gerätes

In allen Beschleunigerabschnitten (UNILAC, SIS, ESR, Transferstrecken) gibt es Netzgeräte, die gepulst betrieben werden.

Gepulst heißt, dass diese Geräte mit einem neuen Sollwert für den Strom angesteuert werden, wenn eine neue Teilchensorte oder Teilchenenergie eingeschossen wird. Mit anderen Worten, bei jedem neuen virtuellen Beschleuniger wird ein neuer Sollwert gesetzt, der dann während der Dauer dieses Beschleunigungszyklus konstant bleibt.

Während bei den gerampten Geräten von einer relativ kleinen Sollwertänderung pro Ansteuerintervall ausgegangen werden kann (d.h. kleine Zeitverzögerung zwischen den Sollwerten, $< 1ms$), können bei gepulsten Geräten große Änderungen des Sollwertes zwischen zwei Ansteuerungen erfolgen, sodass bei der Definition des Ansteuerzeitpunktes berücksichtigt werden muss, wann bei einem Sprung des Sollwertes der Istwert am Gerät erreicht ist.

Zur Ansteuerung der gepulsten Geräte ist die Kenntnis sowohl der minimalen und maximalen Einstellwerte, als auch die der maximalen Zeitverzögerung zwischen dem Bereitstellen des Sollwertes und dem Erreichen des entsprechenden Istwertes erforderlich. Ebenfalls muss bekannt sein, zu welchem Zeitpunkt, bezogen auf den Zyklus, der Istwert anstehen muss, um Aussagen über die Bereitstellungszeit des Sollwertes machen zu können.

Alle gepulsten Magnet-Netzgeräte verhalten sich im Prinzip gleich. Für diese wird das Gerätemodell 'MX-Magnete' definiert. Unterschiede wie unipolar/bipolar, Polwender oder nicht, Lastumschaltung oder nicht werden zum Teil als Gerätekonstanten getroffen und zum Teil als Variante dieses Gerätemodells behandelt.

2 Die Hardware des Gerätes

Hier wird ein Überblick über die Funktionsweise des Gerätes gegeben, also wie es aufgebaut ist und wie es intern funktioniert.

2.1 Gerätevarianten

Geräte können sich in folgenden prinzipiellen Eigenschaften unterscheiden.

Unipolare Geräte: Ihr Arbeitsbereich liegt zwischen 0 und $+max$ Ampère.

Bipolare Geräte: Ihr Arbeitsbereich liegt zwischen $-max$ und $+max$ Ampère.

Geräte mit Polwender: Die Stromrichtung der Geräte kann umgeschaltet werden. Ihr Arbeitsbereich liegt je nach Stellung des Polwenders entweder zwischen $-max$ und 0 Ampère (negative Polarität) oder zwischen 0 und $+max$ Ampère (positive Polarität).

Geräte mit Lastumschaltung: Ein Magnet (Last) teilt sich mit maximal 4 anderen Magneten (Lasten) eine Stromversorgungseinheit (Netzgerät). Diese Magneten sind *nur* alternativ zu betreiben. Sie besitzen weiterhin eine der oben genannten Eigenschaften (unipolar, bipolar oder Polwender).

3 Die Schnittstelle zum Gerät

Dieses Kapitel beschreibt den Aufbau der digitalen Schnittstelle zum Rechnersystem.

3.1 Funktionscodes der Interfacekarte

Die für die Geräteansteuerung definierten Funktionscodes sind in der folgenden Tabelle aufgelistet. Als Modus ist angegeben, ob Daten von der Interfacekarte gelesen werden, ob Daten zu der Interfacekarte geschrieben werden, oder ob nur eine Funktion ausgeführt wird.

Die Codes und ihre Bedeutung sind in der folgenden Tabelle zusammengefasst.

Funktionscode		Modus	Bedeutung
Name	Hex		
<code>ifb_reset</code>	01	Funktion	Reset
<code>ifb_power_on</code>	02	Funktion	Netz einschalten
<code>ifb_power_off</code>	03	Funktion	Netz ausschalten
<code>ifb_pol_plus</code>	04	Funktion	Polwender auf normale Polarität
<code>ifb_pol_minus</code>	05	Funktion	Polwender auf inverse Polarität
<code>ifb_soll_1</code>	06	Schreiben	Strom setzen
<code>ifb_intr_mask</code>	12	Schreiben	Interruptmaske setzen
<code>ifb_dev_fct_1</code>	14	Funktion	Magnet 1 (Last 1) anschalten
<code>ifb_dev_fct_2</code>	15	Funktion	Magnet 2 (Last 2) anschalten
<code>ifb_dev_fct_3</code>	16	Funktion	Magnet 3 (Last 3) anschalten
<code>ifb_dev_fct_4</code>	17	Funktion	Magnet 4 (Last 4) anschalten
<code>ifb_dev_fct_5</code>	18	Funktion	Magnet 5 (Last 5) anschalten
<code>ifb_ist_1</code>	81	Lesen	Strom lesen
<code>ifb_rdstat</code>	C0	Lesen	Gerätestatus, 1. Byte lesen
<code>ifb_rdstat_1</code>	C1	Lesen	Gerätestatus, 2. Byte lesen
<code>ifb_rdstat_2</code>	C2	Lesen	Gerätestatus, 3. Byte lesen
<code>ifb_rdstat_int</code>	C9	Lesen	Status der Interfacekarte lesen

`ifb_reset`

Das Gerät wird in einen definierten Ausgangszustand gebracht. Entsprechende Hardwarestatusbits und das Summeninterlock werden zurückgesetzt. Das Signal muss mindestens 200ms anstehen, bevor ein neuer Funktionscode geschickt werden darf.

`ifb_power_on`, `ifb_power_off`

Einschalten bzw. Ausschalten des Gerätes. Das Signal muss mindestens 200ms anstehen, bevor ein neuer Funktionscode geschickt werden darf.

3.2 D/A- und A/D-Wandler

Zum Setzen der Sollwerte und Lesen der Istwerte des Stromes arbeitet das Gerät mit 16bit breiten D/A- und A/D-Wandlern. Die Normierung ist wie folgt.

$$8000_{\text{hex}} \hat{=} -10 V$$

$$0 \hat{=} 0V$$

$$7FFF_{\text{hex}} \hat{=} 10V$$

Diese Normierung gilt für *alle* Geräte. So haben z. B. unipolare Geräte nur einen Wertebereich von 0 bis $7FFF_{\text{hex}}$ entsprechend 0 bis 10 V..

3.3 Interlock Interrupt

Das Gerät arbeitet im interrupt-orientierten Summeninterlock-Service.

3.4 Data Request (DRQ) Interrupts

DRQ-Interrupts werden nicht erwartet bzw. verarbeitet.

3.5 Data Ready (DRD) Interrupts

DRD-Interrupts werden nicht erwartet bzw. verarbeitet.

3.6 Definition der Bits des Hardwarestatus

Die Geräte liefern 3 Bytes Statusinformation. Die Funktionscodes zum Lesen der entsprechenden Bytes sind in der folgenden Tabelle zusammengefasst.

Statusbits	Funktionscode
8 ... 15	ifb_rdstat
16 ... 23	ifb_rdstat_1
24 ... 31	ifb_rdstat_2

Die Bits 0 ... 7 sind die systemweiten sogenannten generierten Softwarestatusbits (in engl. derived status bits). Die Statusbits im Einzelnen sind in der folgenden Tabelle zusammengefasst.

Bit	Name	Bedeutung	
		High (1)	Low (0)
0	Power	on	off
1	Remote/Local	Remote	Local
2		reserved	
3		reserved	
4	Emergency	no	yes
5	Interlock	no	yes
6	HW Error	no	yes
7	SW Error	no	yes
8	Power	on	off
9	Voltage	ok	low
10	Temperature PS	ok	high
11	Cooling Water PS	ok	fail
12	Current	ok	exceeded
13	Inverter	default	invers
14	Temperature Magnet	ok	high

Forts. auf nächster Seite

Forts. von letzter Seite		Bedeutung	
Bit	Name	High (1)	Low (0)
15	Cooling Water Magnet	ok	fail
16	Current I_{p2}	ok	exceeded
17	Current I_{p1}	ok	exceeded
18	Transistor $T_1 - T_4$	ok	failure
19	Transistor T_5	ok	failure
20	Voltage U_{d1}	ok	exceeded
21	Voltage U_{b0}	ok	exceeded
22	Stromwandler	ok	saturiert
23	Erdschluss	no	yes
24	Remote/Local	Remote	Local
25	Radiation Protection	none	Interlock
26	Magnet 1	connected	not connected
27	Magnet 2	connected	not connected
28	Magnet 3	connected	not connected
29	Magnet 4	connected	not connected
30	Magnet 5	connected	not connected
31	Actual Value	ok	deviation

4 Die Steuerung des Gerätes

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

4.1 Aufgaben im Normalbetrieb

4.1.1 Soll- und Istwert

Der Sollwert kann als Feld in Teslameter (Realwert), als Strom in Ampère (Realwert) oder direkt als Spannung für die D/A-Wandler in Millivolt (Integerwert) angegeben werden. Dabei gilt folgende Normierung.

$$\begin{aligned}
 -10\text{ V} &\hat{=} -max\text{ A} \\
 0\text{ V} &\hat{=} 0\text{ A} \\
 10\text{ V} &\hat{=} max\text{ A}
 \end{aligned}$$

Der minimale und maximale *Stromsollwert*, der vom Operating vorgegeben werden darf, ist geräteabhängig und der entsprechenden Netzgeräteliste zu entnehmen.

Der minimale und maximale *Spannungssollwert* liegt bei unipolare Magneten zwischen 0 Volt und +10 Volt, bei bipolaren Magneten zwischen -10 Volt und +10 Volt.

Für Magnete mit Polwender gilt sowohl für den Strom- als auch für den Spannungssollwert: Der Sollwert kann zwischen $-max$ und $+max$ liegen (wobei max geräteabhängig für den Strom- bzw. 10V für den Spannungssollwert ist). Steht der Polwender in normaler (positiver) Stellung, so ist der minimale Sollwert gleich Null, steht der Polwender in inverser (negativer) Stellung, so ist der maximale Sollwert gleich Null.

Spezielle Behandlung bei FIELD, da durch Polynome Grenzwertüberschreitungen möglich sind. Das gilt für Bereichsüberprüfungen (min - max), als auch für Polwenderbehandlung.

Es soll möglich sein den Sollwert in allen Formaten zurücklesen zu können.

Der Istwert kann ebenfalls in allen Formaten gelesen werden. Dabei gilt die gleiche Normierung.

4.1.2 Polwender

Nicht alle Geräte sind mit Polwender ausgerüstet.

Polwender dürfen nur geschaltet werden, wenn die Power des Gerätes ausgeschaltet ist. Es gibt die zwei Sollwerte *Polwender invers* und *Polwender normal*.

Steht der Polwender in inverser (negativer) Stellung, dürfen vom Operating nur negative Sollwerte gesetzt werden. Steht der Polwender in normaler (positiver) Stellung, dürfen vom Operating nur positive Sollwerte gesetzt werden.

Die Stellung des Polwenders kann ohne Fehler von *jedem* Gerät gelesen werden. Besitzt das Gerät keinen Polwender, wird diese Information als Istwert zurückgeliefert.

Somit ergeben sich die drei möglichen Istwerte *Polwender invers*, *kein Polwender* und *Polwender normal*.

4.1.3 Lastumschaltung

Zur näheren Erklärung ist es hier kurzzeitig erforderlich, das bisher als ganzes betrachtete *Gerät* in zwei Komponenten, nämlich in *Netzgerät* und *Magnet*, zu zerlegen. Üblicherweise besteht ein Gerät vom Typ MX genau aus je einer dieser Komponenten (siehe Abbildung 1).



Abbildung 1: MX ohne Lastumschaltung

Nun gibt es Magnete, bei denen es keinen Sinn macht, sie gleichzeitig zu betreiben, weil sich Teilchen nur in dem einen *oder* dem anderen Strahlzweig bewegen können. Aus diesem Grund gibt es für alle alternativ zu betreibenden Magnete nur jeweils *ein* Netzgerät, das an den aktuell zu betreibenden Magneten geschaltet werden kann (siehe Abbildung 2). Maximal ist es möglich fünf verschiedene Magnete (Lasten) an ein Netzgerät zu schalten.

Auch hier muss, wie üblich, der entsprechende *Magnet* (MX 1 . . . MX 5) und nicht das Netzgerät (NG) über die Nomenklatur erreicht werden (obwohl es für 5 Magneten nur eine Interfacekarte gibt).

Diese Methode erfordert die Einhaltung einiger Bedingungen und führt zu einigen Einschränkungen, die im folgenden aufgezählt werden.

Voraussetzungen sind:

- An einer SE dürfen nur Magnete mit Lastumschaltung hängen.
- Jedes Netzgerät hat die gleiche Anzahl Magnete (Lasten, 5 pro Netzgerät).
- Das Netzgerät muss mit Grund- und Sekundärgeräteadressen erreicht werden können.
- Der Gerätestatus liefert eine Information, welcher Magnet am Netzgerät hängt.

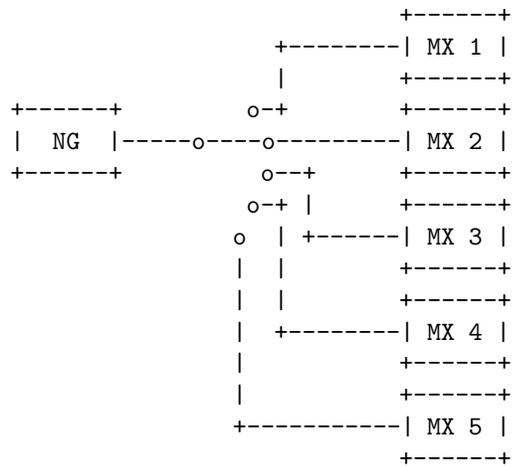


Abbildung 2: MX mit Lastumschaltung

- Die Grundadresse (phys. IFB-Adresse des Netzgerätes) muss immer ein Vielfaches von 5 sein.

Einschränkungen sind:

- Es kann nur jeweils ein Magnet pro Netzgerät erreicht werden. Der Versuch einen anderen als den angeschalteten zu erreichen führt zu einem Fehler (MX-F-NoPowerSupply).
- Es ist nicht offensichtlich, welche Magnete sich ein Netzgerät teilen.
- Will man einen anderen als den aktuell angeschalteten Magneten betreiben, so muss zunächst der angeschaltete Magnet ausgeschaltet und anschließend der neu gewünschte Magnet eingeschaltet werden. Die notwendige Umschaltung der Last erfolgt dann automatisch.

4.1.4 Einschalten

Für die Netzgeräte müssen folgende Sequenzen zum Einschalten beachtet werden.

Sequenz für Geräte ohne Lastumschaltung:

1. Sollwert 0 setzen
2. Power einschalten
3. 200 Millisekunden ohne Unterbrechung warten
4. nach 2 Sekunden EIN-Status prüfen

Sequenz für Geräte mit Lastumschaltung:

1. Falls Netzgerät nicht am Magnet hängt:
 - (a) Last an Netzgerät schalten
 - (b) 2 Sekunden warten
 - (c) Reset an Gerät schicken

- (d) 200 Millisekunden warten
 - (e) Anschalten überprüfen
2. Sollwert 0 setzen
 3. Power einschalten
 4. 200 Millisekunden ohne Unterbrechung warten
 5. nach 2 Sekunden EIN-Status prüfen

4.1.5 Ausschalten

Für die Netzgeräte müssen folgende Sequenzen zum Ausschalten beachtet werden.

Sequenz für alle Geräte:

1. Sollwert 0 setzen
2. Power ausschalten
3. 200 Millisekunden ohne Unterbrechung warten
4. nach 2 Sekunden AUS-Status prüfen

4.2 Genauigkeitsanforderungen

Durch die Wahl der Formate der Soll- und Istwerte werden die Anforderungen eines 16bit DACs bzw. ADCs, die für diese Geräte eingesetzt werden, erreicht.

4.3 Zeitkritische Anforderungen

Das Erreichen der gewünschten Istwerte nach dem relevanten Event zum Sollwert setzen ist nur bei den Magneten im UNILAC kritisch. Die Istwerte dort müssen nach maximal $7ms$ erreicht sein.

4.4 Einordnung in das Timing

Je nach Einsatzort des Gerätes gibt es unterschiedliche Betriebsvarianten.

Magnete am Hochstrominjektor (HSI) erfordern eine spezielle Betriebsart. Erklärungen und Messungen dazu finden sich in [7], [4], [5], [6], [2], [3] und [8].

Für Magnete am UNILAC gilt: Das Gerät soll bei Event `Evt_Prep_Next_Acc` auf den Sollwert fahren und bei Event `Evt_Beam_On` den aktuellen Istwert erfassen.

Für Magnete am SIS-Timing *vor* dem SIS (Injektion) gilt: Das Gerät soll bei Event `Evt_Prep_Inj` auf den Sollwert fahren und bei Event `Evt_UNI_Ready` den aktuellen Istwert erfassen.

Für Magnete am SIS-Timing *nach* dem SIS (Extraktion) gilt: Das Gerät soll bei Event `Evt_Prep_Inj` auf den Sollwert fahren und bei Event `Evt_Flattop` den aktuellen Istwert erfassen.

Für Magnete am Elektronenkühler (ESR-Timing) gilt: Das Gerät soll bei Event `Evt_ECE_HV_On` auf den Sollwert fahren und bei Event `Evt_ECE_HV_Mess` den aktuellen Istwert erfassen.

Ein Beispiel eines SIS-Standardzyklus mit relevanten Events für MX-Magnete zur SIS-Injektion ist in Abbildung 3 angegeben.

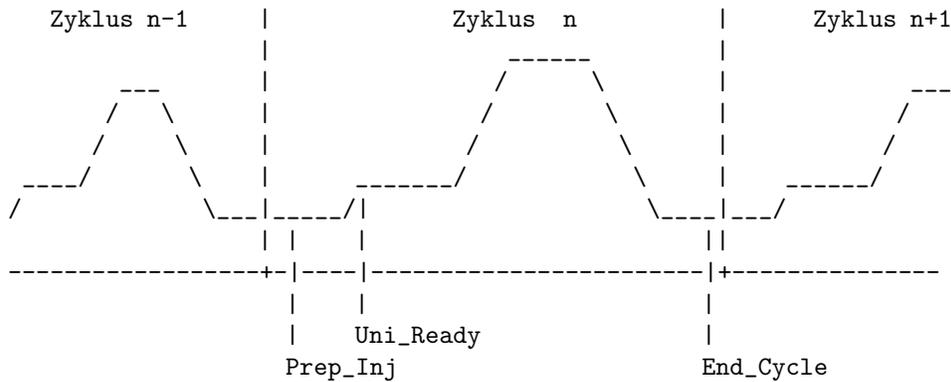


Abbildung 3: Standardzyklus mit relevanten Events für MX-Magnete

Am Ende eines Zyklus muss die Software den internen Zustand testen und eventuell Aufräumarbeiten vornehmen. Dazu ist eine weitere Eventkonnectierung nötig. Aus Zeitgründen entfällt diese Konnectierung für die UNILAC-Variante. Die Konnectierungen sind in Tabelle 3 zusammengefasst.

Aktion	UNILAC	SIS-Inj	SIS-Ext	E-Cooler
Sollwert	Evt_Prep_next_Acc	Evt_Prep_Inj	Evt_Prep_Inj	Evt_ECE_HV_On
Istwert	Evt_Beam_On	Evt_Uni_Ready	Evt_Flattform	Evt_ECE_HV_Mess
Zustand	entfällt	Evt_Ramp_Start	Evt_Ext_End	Evt_End_Cycle

Tabelle 3: Standard-Eventkonnectierungen für MX-Magnete

4.5 Festlegung von Startwerten

4.5.1 Kaltstarts

Bei einem Kaltstart werden folgende Aktionen durchgeführt:

- Es wird ein Gerätereset durchgeführt.
- Alle Sollwerte werden für alle virtuellen Beschleuniger auf Null gesetzt und am Gerät eingestellt.
- Die Istwerte werden auf Null gesetzt, wobei sowohl ein Eventstamp als auch ein Timestamp von Null eingetragen wird.
- Das Gerät wird für alle virtuellen Beschleuniger aktiv geschaltet.
- Die Interlockbehandlung wird aktiviert.
- Die SE wird in den Eventmode-Betrieb geschaltet.
- Die Standard-Eventkonnectierungen müssen gesetzt werden (siehe Tabelle 3 auf Seite 14).

Power, Polwender und Lastumschaltung werden bei einem Kaltstart *nicht* verändert.

4.5.2 Warmstarts

Bei einem Warmstart werden folgende Aktionen durchgeführt:

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

Power, Polwender und Lastumschaltung werden bei einem Warmstart *nicht* verändert.

4.6 Handbetrieb

Das Gerät kann auf Handbetrieb (local) geschaltet werden. Der Zustand wird im Status des Gerätes angezeigt. Nach dem Zurückschalten auf Rechnerbetrieb (remote) werden die zuletzt geschickten Sollwerte wieder realisiert. Per Hand veränderte Werte werden nicht übernommen.

4.7 Ableitung des Hardwarefehler-Bits aus dem Gerätestatus

Ein Hardwarefehler (angezeigt im Hardwarefehler-Bit des Status) liegt vor, wenn eines der folgenden Bits des Hardwarestatus *nicht* den angegebenen Wert (nicht OK) anzeigt.

Bit	Name	Wert
9	Voltage	1
10	Temperature PS	1
11	Cooling Water PS	1
12	Current	1
14	Temperature Magnet	1
15	Cooling Water Magnet	1

4.8 Verhalten bei Störungen

4.8.1 Geräteinterlock

Tritt ein Interlock auf, so ist der Strom des Gerätes auf 0 Ampère zu setzen.

4.8.2 Event-Sequenzfehler

Kommen die relevanten Events nicht in der richtigen Reihenfolge (Sollwert setzen, Istwert lesen, Aufräumen (fällt weg für UNILAC)), so ist der aktuelle Zyklus abubrechen.

4.8.3 Event-Overrun

Bei einem Overrun-Fehler ist der Zyklus abubrechen.

4.8.4 Emergency-Event

Beim Auftreten eines Emergency-Events ist der Strom des Gerätes auf 0 Ampère zu setzen.

4.8.5 Ausfall der Kommunikation EC – Gerät

Der Ausfall der Kommunikation zwischen EC und Gerät führt zu Timeouts. Tritt ein Timeout auf so ist der Zyklus abzuberechnen.

4.8.6 Sonstiges

Keine weiteren Störungen vorhanden.

4.9 Bedienungsfehler vom Operating

Sollwerte außerhalb des zulässigen Bereichs sind zurückzuweisen.

4.10 Sonstige Anforderungen

Keine.

5 Die Repräsentation des Gerätes

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

5.1 Kennzeichnung des Gerätemodells

Das Gerätemodell hat die Bezeichnung **MX**.

Die Gerätemodellnummer ist 19_{dez} .

5.2 Die Master-Properties

Property	Klasse	Parameter		Daten		Größe	
		Anz.	Typ	Anz.	Typ	Einh.	Exp.
INFOSTAT	RA	0	–	25	BitSet32	1	0
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
CALC	RA	2	Structure	3	Structure	div.	div.
CONNECT	R	0	–	1	BitSet16	0	1
CONSTANT	RA	0	–	3	RealF	1	0
INVERTER	R/W	0	–	1	Integer16	1	0
MAGNSVCI	R	0	–	1	Integer32	1	0
MAGNSVCS	R/W	0	–	1	Integer32	1	0

5.2.1 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:** Reserviert für Erweiterungen.
- ⋮
- 25:** Reserviert für Erweiterungen.

5.2.2 INIT

Bedeutung: Initialisierung des Gerätes (Kaltstart). Für die dabei durchzuführenden Aktionen siehe Abschnitt 4.5.1 auf Seite 14.

Parameter: Keine.

Daten: Keine.

5.2.3 POWER

Bedeutung: Gibt an, ob der Leistungsteil des Gerätes ein- oder ausgeschaltet ist bzw. werden soll. Beim Einschalten eines Gerätes mit Lastumschaltung muss gegebenenfalls vorher die Last umgeschaltet werden.

Parameter: Keine.

Daten: Das Datum kann nur zwei Werte annehmen. Null heißt, das Gerät ist eingeschaltet bzw. soll eingeschaltet werden. Eins heißt, das Gerät ist ausgeschaltet bzw. soll ausgeschaltet werden.

5.2.4 RESET

Bedeutung: Reset des Gerätes (Warmstart). Für die dabei durchzuführenden Aktionen siehe Abschnitt 4.5.2 auf Seite 15.

Parameter: Keine.

Daten: Keine.

5.2.5 STATUS

Bedeutung: Auslesen des 32bit Gerätestatus.

Parameter: Keine.

Daten: Das 32bit Statuswort. Die Bits entsprechen den Statusbits, wie sie in Abschnitt 3.6 auf Seite 9 und in der Tabelle 1 auf Seite 8 erklärt sind.

5.2.6 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

5.2.7 CALC

Bedeutung: Feld, Strom und Spannung aus gegebenem Feld, Strom oder Spannung berechnen. Dies ist eine reine Rechen-Property. Sie beeinflusst in keiner Weise das Gerät.

Parameter: Die 2 Parameter haben folgende Struktur:

Vorgabetyp: Integer32
Vorgabe: RealF

Der Vorgabetyp kann 3 Werte annehmen:

Vorgabetyp = 1 \Rightarrow Vorgabe ist Feld in Tm

Vorgabetyp = 2 \Rightarrow Vorgabe ist Strom in A

Vorgabetyp = 3 \Rightarrow Vorgabe ist Spannung in mV

Daten: Die zurück gelieferten Daten haben folgende Struktur und Bedeutung:

Komponente	Einheit	Format
Feld	Tm	RealF
Strom	A	RealF
Spannung	mV	Integer32

5.2.8 CONNECT

Bedeutung: Gibt an, ob der Magnet (mit Lastumschaltung) mit dem Netzgerät verbunden ist, oder verbunden bzw. getrennt werden soll. Die Lastumschaltung ist nur möglich, wenn das Netzgerät ausgeschaltet ist.

Parameter: Keine.

Daten: Das Datum kann nur zwei Werte annehmen. Null heißt, der Magnet ist vom Netzgerät getrennt bzw. soll vom Netzgerät getrennt werden. Eins heißt, der Magnet ist mit dem Netzgerät verbunden bzw. soll mit dem Netzgerät verbunden werden.

5.2.9 CONSTANT

Bedeutung: Liefert die gerätespezifischen Konstanten.

Parameter: Keine.

Daten: Die 4 Realwerte bedeuten im Einzelnen:

- 1: Minimal einstellbarer Wert des Stroms in Ampère.
- 2: Maximal einstellbarer Wert des Stroms in Ampère.
- 3: Eichkonstante. Sie wird mit dem gemessenen Istwert des Stroms multipliziert.
- 4: Gerätesubtyp. Zur Unterscheidung verschiedener Geräteeigenschaften. Die Werte haben folgende Bedeutung (NG = Netzgerät):

Binärwert	Geräteeigenschaft
xxxx xxxx xxxx xx01	Unipolares NG
xxxx xxxx xxxx xx10	Bipolares NG
xxxx xxxx xxxx xx11	NG mit Polwender
xxxx xxxx xxxx 00xx	Keine Aktion bei Event Magn_Down
xxxx xxxx xxxx 01xx	Setze I = 0 bei Event Magn_Down
xxxx xxxx xx00 xxxx	Magnet hat permanentes NG
xxxx xxxx xx01 xxxx	Magnet teilt sich NG mit anderen Magneten
xxxx xxxx xx10 xxxx	Magnet teilt sich NG, manuelle Umschaltung
xxxx xxxx 00xx xxxx	Gerät hat 1 ADC
xxxx xxxx 01xx xxxx	Gerät hat 2 ADCs
xxxx xx00 xxxx xxxx	Magnet ohne Hallsonde
xxxx xx01 xxxx xxxx	Magnet mit Hallsonde
xxxx 00xx xxxx xxxx	NG ist stromgeregelt
xxxx 01xx xxxx xxxx	NG ist feldgeregelt
xxxx 10xx xxxx xxxx	NG mit manuell umschaltbarer Regelung
xx00 xxxx xxxx xxxx	Nennwerte DAC/ADC1/ADC2 = 10V (DAC/ADC)
xx01 xxxx xxxx xxxx	Nennwerte DAC/ADC1 = 11V, ADC2 = 10V (ACU)

5.2.10 INVERTER

Bedeutung: Lesen der Stellung oder Umschalten des Polwenders.

Parameter: Keine.

Daten: Das Datum kann beim Schreiben (Schalten des Polwenders) *zwei* sowie beim Lesen des Polwenders *drei* Werte annehmen.

Wert	Bedeutung
-1	inverse (negative) Polarität
0	kein Polwender vorhanden (Wert kann nur gelesen werden)
1	normale (positive) Polarität

5.2.11 MAGNSVCI

Bedeutung: Lese den Istwert eines Gerätes unabhängig vom virtuellen Beschleuniger. Der Istwert ist für den DAC/ADC normiert auf $10000mV$.

Dies ist eine Property für den Magnet-Service zum Lesen des DC-Istwertes. Das Gerät muss für alle virtuellen Beschleuniger inaktiv sein! Ist dies nicht der Fall, wird ein Fehler zurück geliefert.

Parameter: Keine.

Daten: Der gelesene Istwert zwischen $-10000mV$ und $10000mV$ bzw. zwischen $0mV$ und $10000mV$.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, werden nur negative Istwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.2.12 MAGNSVCS

Bedeutung: Setze oder lese den Sollwert eines Gerätes unabhängig vom virtuellen Beschleuniger. Der Sollwert ist für den DAC/ADC normiert auf $10000mV$.

Dies ist eine Property für den Magnet-Service zum Setzen eines DC-Sollwertes. Das Gerät muss für alle virtuellen Beschleuniger inaktiv sein! Ist dies nicht der Fall, wird ein Fehler zurück geliefert.

Parameter: Keine.

Daten: Der zu setzende bzw. gelesene Sollwert zwischen $-10000mV$ und $10000mV$ bzw. zwischen $0mV$ und $10000mV$.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, dürfen nur negative Sollwerte gesetzt werden bzw. werden nur negative Sollwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3 Die Slave-Properties

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
CURRENTI	R	0	-	1	RealF	Ampere	0
CURRENTS	R/W	0	-	1	RealF	Ampere	0
DSTATUS	R	0	-	1	BitSet16	1	0
FIELDI	R	0	-	1	RealF	Tm	0
FIELDS	R/W	0	-	1	RealF	Tm	0
MAGNINFO	RA	0	-	80	Structure	div.	0
MEDDATAI	RA	3	BitSet32	2	RealF	Tm	0
MEDDATAS	WA	3	BitSet16	3	RealF	Tm, s	0, -6
MEDDATAS	RA	4	BitSet32	3	RealF	Tm, s	0, -6
VOLTS	R/W	0	-	1	Integer16	1	0
VOLTI	R	0	-	1	Integer16	1	0

5.3.1 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.

5.3.2 COPYSET

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

Parameter: Keine.

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

5.3.3 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: Hier hat nur der erste der 217 Parameter eine Bedeutung.

1: Wird bei konnektierten Aufträgen ausgewertet. 0: Es wird bei jeder Ausführung des Auftrages eine Antwort verschickt. 1: Es wird bei jeder Ausführung des Auftrages nur dann eine Antwort verschickt, wenn sich seit dem letzten Aufruf der Inhalt der Daten geändert hat.

2...217: Dummy, sie werden vom MOPS intern verwendet und können vom Benutzer beliebig gesetzt werden.

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

5.3.4 CURRENTI

Bedeutung: Lese den Istwert eines Gerätes.

Parameter: Keine.

Daten: Der gelesene Istwert in Ampère.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, werden nur negative Istwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.5 CURRENTS

Bedeutung: Setze oder lese den Sollwert eines Gerätes.

Parameter: Keine.

Daten: Der zu setzende bzw. gelesene Sollwert in Ampère.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, dürfen nur negative Sollwerte gesetzt werden bzw. werden nur negative Sollwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.6 DSTATUS

Bedeutung: Lese den Degaussing-Status eines Gerätes.

Parameter: Keine.

Daten: Der gelesene Degaussing-Status.

5.3.7 FIELDI

Bedeutung: Lese den Istwert eines Gerätes.

Parameter: Keine.

Daten: Der gelesene Istwert in Teslameter.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, werden nur negative Istwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.8 FIELDS

Bedeutung: Setze oder lese den Sollwert eines Gerätes.

Parameter: Keine.

Daten: Der zu setzende bzw. gelesene Sollwert in Teslameter.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, dürfen nur negative Sollwerte gesetzt werden bzw. werden nur negative Sollwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.9 MAGNINFO

Bedeutung: Diese Property ist für spezielle Operatingprogramme designed. Sie liefert mit einem Lesezyklus verschiedene, ständig benötigte Informationen an diese Programme.

Parameter: Keine.

Daten: Insgesamt 80 Bytes vom SIS-Datentyp Structure, die in 20 Elementen wie folgt strukturiert sind:

Master-Status (BitSet32) Gerätestatus.

Deg.-Status (BitSet32) Dynamischer Degaussing-Status eines Degaussing-Magneten.

Aktivzustand (BitSet32) Aktivzustand. 0 = inaktiv, 1 = aktiv.

Polwender (Integer32) Polwenderstellung. 1 = normal, 0 = kein Polwender, -1 = invers.

ActValMode (Integer32) Berechnungsmodus der Istwerte. Im Low-Byte (Bits 0 bis 7) befindet sich die Information, wie der Feldistwert berechnet wurde. Im zweiten Byte (Bits 8 bis 15) befindet sich die Information, wie der Stromistwert berechnet wurde. Im dritten Byte (Bits 16 bis 23) befindet sich die Information, wie der Spannungsistwert berechnet wurde. Die Werte in den einzelnen Bytes haben die Bedeutung:

1 = Istwert aus Magnetstrom (DCCT) berechnet,

2 = Istwert aus Hallspannung berechnet,

3 = kein Polynom zur Berechnung vorhanden.

Das High-Byte (Bits 24 bis 31) ist nicht belegt.

(reserviert) (Integer32)

(reserviert) (Integer32)

(reserviert) (Integer32)

VoltS (Integer32) Flattop-Sollwert in mV.

VoltI (Integer32) Istwert in mV.

CurrentS (Real) Sollwert in A.

CurrentI (Real) Istwert in A.

FieldS (Real) Sollwert in Tm.

FieldI (Real) Istwert in Tm.

(reserviert) (Real)

(reserviert) (Real)

(reserviert) (Real)

(reserviert) (Real)

(reserviert) (Real)

(reserviert) (Real)

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, werden nur negative Soll- und Istwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.10 MEDDATAI

Bedeutung: Lesen eines Istwertsatzes aus dem Therapiebetrieb.

Parameter: Die Struktur der Parameter ist in [1] beschrieben.

Daten: Insgesamt 12 Bytes vom SIS-Datentyp Structure, die in 3 Elementen wie folgt strukturiert sind:

VoltI (Integer32) Istwert in mV.

CurrentI (Real) Istwert in A.

FieldI (Real) Istwert in Tm.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, werden nur negative Istwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.11 MEDDATAS

Bedeutung: Schreiben oder Lesen eines Sollwertsatzes für den Therapiebetrieb.

Parameter: Die Struktur der Parameter ist unterschiedlich, je nachdem ob geschrieben oder gelesen wird. Die Strukturen sind in [1] beschrieben.

Daten: Die Struktur der Daten ist unterschiedlich, je nachdem ob geschrieben oder gelesen wird.

Daten schreiben: Der zu setzende Sollwert in Teslameter.

Daten lesen: Insgesamt 12 Bytes vom SIS-Datentyp Structure, die in 3 Elementen wie folgt strukturiert sind:

VoltS (Integer32) Sollwert in mV.

CurrentS (Real) Sollwert in A.

FieldS (Real) Sollwert in Tm.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, dürfen nur negative Sollwerte gesetzt werden bzw. werden nur negative Sollwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.12 VOLTS

Bedeutung: Setze oder lese den Sollwert eines Gerätes. Der Sollwert ist für den DAC/ADC normiert auf $10000mV$.

Parameter: Keine.

Daten: Der zu setzende bzw. gelesene Sollwert zwischen $-10000mV$ und $10000mV$ bzw. zwischen $0mV$ und $10000mV$.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, dürfen nur negative Sollwerte gesetzt werden bzw. werden nur negative Sollwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

5.3.13 VOLTI

Bedeutung: Lese den Istwert eines Gerätes. Der Istwert ist für den DAC/ADC normiert auf $10000mV$.

Parameter: Keine.

Daten: Der gelesene Istwert zwischen $-10000mV$ und $10000mV$ bzw. zwischen $0mV$ und $10000mV$.

Für Netzgeräte mit Polwender gilt: Steht der Polwender in inverser (negativer) Stellung, werden nur negative Istwerte gelesen. Entsprechendes gilt für die Normstellung des Polwenders.

Teil II

Der Entwurf der Software

6 Softwareentwurf

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

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

7 Lokale Datenbasis

7.1 Tabelle der Konstanten

Für jedes Gerät gibt es eine Beschreibung aus 4 Elementen in der Konstantentabelle der lokalen Datenbasis. Die Elemente haben in der Reihenfolge folgende Bedeutung:

- 1:** Minimal einstellbarer Wert des Stroms in Ampère.
- 2:** Maximal einstellbarer Wert des Stroms in Ampère.
- 3:** Eichkonstante. Sie wird mit dem gemessenen Istwert des Stroms multipliziert.
- 4:** Gerätesubtyp. Zur Unterscheidung in unipolare, bipolare, Geräte mit Polwender oder Lastumschaltung.

Siehe auch die Beschreibung der Property CONSTANT.

8 Dualport RAM

Hierher kommt die Beschreibung des gerätespezifischen Teils des Dualport RAM (`m_data_type`, `s_data_type`, `Dev_Common_Buf_Type`, ...).

9 USRs - User Service Routinen

9.1 Obligatorische USRs

9.1.1 N_Init

9.1.2 N_Reset

9.1.3 R_Status

9.1.4 R_Power

9.1.5 W_Power

9.1.6 R_Active

9.1.7 W_Active

9.1.8 W_CopySet

9.1.9 R_EQMErr

9.1.10 R_Version

9.1.11 R_InfoStat

9.2 Gerätespezifische USRs

Zuzüglich der obligatorischen USRs werden für die Steuerung der MX-Magnete folgende gerätespezifischen USRs benötigt:

9.2.1 W_FieldS

Umrechnung Field (Real) in Bitmuster für DAC mit Hilfe eines Polynoms dritten Grades, das die Funktion $I(BI)$ beschreibt. Test, ob I innerhalb des definierten Bereichs liegt mit eventueller Fehlermeldung. Überprüfung der Master- und Slave-Errors mit eventueller Fehlermeldung.

9.2.2 R_FieldS

Umrechnung Bitmuster für DAC in Field (Real) mit Hilfe eines Polynoms dritten Grades, das die Funktion $(BI)(I)$ beschreibt. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.3 R_FieldI

Berechnung von Field (Real) aus Bitmuster des ADC unter Berücksichtigung einer Eichkonstanten und mit Hilfe eines Polynoms dritten Grades, das die Funktion $(BI)(I)$ beschreibt.. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.4 W_CurrentS

Test, ob I innerhalb des definierten Bereichs liegt mit eventueller Fehlermeldung. Umrechnung Current (Real) in Bitmuster für DAC. Überprüfung der Master- und Slave-Errors mit eventueller Fehlermeldung.

9.2.5 R_CurrentS

Umrechnung Bitmuster für DAC in Current (Real) Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.6 R_CurrentI

Berechnung von Current (Real) aus Bitmuster des ADC unter Berücksichtigung einer Eichkonstanten. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.7 W_VoltS

Sollwert in Millivolt (I16) in Bitmuster für DAC. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.8 R_VoltS

Sollwert in Millivolt (I16) aus Bitmuster für DAC. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.9 R_VoltI

Berechnung des Istwertes in Millivolt (I16) aus Bitmuster ADC. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.10 W_MagnSvcS

Sollwert in Millivolt (I16) in Bitmuster für DAC. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.11 R_MagnSvcS

Sollwert in Millivolt (I16) aus Bitmuster für DAC. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.12 R_MagnSvcI

Berechnung des Istwertes in Millivolt (I16) aus Bitmuster ADC. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.13 W_Inverter

Test, ob Inverter vorhanden. Test, ob Power OFF ist. Inverter_EQM anstoßen. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.14 R_Inverter

Test, ob Inverter vorhanden. Inverterstellung lesen. Überprüfung der Master- und Slave-Errors und eventuell Fehlermeldung.

9.2.15 R_Constant

Lesen der Gerätekonstanten.

9.2.16 R_PotiInfo

Spezielle USR für Poti-Operatingprogramm.

9.3 Globale Routinen

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

Im Modul USRs existieren keine globalen Routinen.

10 EQMs - Equipment Module

10.1 Interne Zustände

10.1.1 Bedeutung der internen Zustände

Für die Gerätesoftware sind folgende interne Zustände definiert:

Not_Set	Initzustand. Dieser Zustand sollte nie auftreten.
emergency	Ein Emergency-Event wurde empfangen. Dieser Zustand darf nur durch Rücksetzen vom Operating verlassen werden.
interlock	Ein Interlock wurde gemeldet. In einem periodisch ablaufenden Auftrag wird überprüft, ob die Interlock-Ursache noch vorliegt. Falls nein, Übergang nach ready.
local	Das Gerät wird mit Handsteuerung betrieben.
power_off	Das Gerät ist ausgeschaltet.
power_seq	Das Gerät schaltet gerade ein oder aus.
inverting	Der Polwender schaltet gerade.
error	Während der Abarbeitung eines EQMs wurde ein Fehler erkannt. (besser Erklären).
ready	Das Gerät ist bereit für Aktionen. Ausgangszustand am Beginn eines virtuellen Beschleunigers.
busy	Das Gerät realisiert gerade einen Sollwert.

10.1.2 Übergänge zwischen den Zuständen

Die Zustände und die Übergänge zwischen denselben sind aus Platzgründen in zwei Tabellen (Tabelle 6 und Tabelle 7) aufgeteilt. Die Legende zu diesen Tabellen ist in Tabelle 8 zu finden.

Tabelle der Zustandsübergänge						
von \ nach →	emergency	interlock	local	power_off	power_seq	inverting
emergency	U:	-	RESET, SI	RESET	RESET	-
	B:	-	-	r	Rp	-
	A:	-	Interl_EQM	Reset_EQM	Reset_EQM	-
interlock	U:	Evt_Emerg	-	RESET	RESET	-
	B:	-	-	r	Rp	-
	A:	Emerg_EQM	-	SI_Off_Chk, Reset_EQM	SI_Off_Chk, Reset_EQM	-
local	U:	Evt_Emerg	SI	-	-	-
	B:	-	-	-	Rp	-
	A:	Emerg_EQM	Interl_EQM	-	Status lesen (periodisch)	-
power_off	U:	Evt_Emerg	SI	-	-	Power=1
	B:	-	-	r	-	-
	A:	Emerg_EQM	Interl_EQM	Status lesen (periodisch)	-	Power_EQM Inverter_EQM
power_seq	U:	Evt_Emerg	SI	-	-	-
	B:	-	-	r	Rp	-
	A:	Emerg_EQM	Interl_EQM	Status lesen (periodisch)	ChkPwr_EQM	-
inverting	U:	Evt_Emerg	SI	-	-	-
	B:	-	-	r	Rp	-
	A:	Emerg_EQM	Interl_EQM	Status lesen (periodisch)	ChkInv_EQM	-
error	U:	Evt_Emerg	SI	-	-	-
	B:	-	-	r	Rp	-
	A:	Emerg_EQM	Interl_EQM	Status lesen (periodisch)	Status lesen (periodisch)	-
ready	U:	Evt_Emerg	SI	-	-	Power=0
	B:	-	-	r	Rp	-
	A:	Emerg_EQM	Interl_EQM	Status lesen (periodisch)	Status lesen (periodisch)	Power_EQM
busy	U:	Evt_Emerg	SI	-	-	-
	B:	-	-	-	-	-
	A:	Emerg_EQM	Interl_EQM	-	-	-

Tabelle 6: Zustandsübergangsdiagramm 1

10.1.3 Standard-Zustandsübergänge

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

Das Gerät nimmt an der Puls-zu-Puls-Modulation teil. Innerhalb eines virtuellen Beschleunigers wird zum Sollwert setzen der Zustand `ready` erwartet und zum Zustand `busy` weitergeschaltet. Zum Istwert lesen wird der Zustand `busy` erwartet und zum Zustand `ready` weitergeschaltet.

`ready -> busy -> ready`

Das Gerät schaltet ein bzw. aus.

`power_off -> power_seq -> ready`
`ready -> power_seq -> power_off`

Das Gerät schaltet den Polwender:

`power_off -> inverting -> power_off`

10.2 Eventkonnektierte EQMs

10.2.1 CurrentS_EQM

Event: Evt_Prep_Inj.

Aktion: (Sollwert setzen, genauer Ablauf, nicht UNILAC).

Tabelle der Zustandsübergänge				
von ↓	nach →	error	ready	busy
emergency	U:	-	RESET	-
	B:	-	-	-
	A:	-	Reset_EQM	-
interlock	U:	-	RESET	-
	B:	-	-	-
	A:	-	SI_Off_Chk, Reset_EQM	-
local	U:	-	-	-
	B:	-	RP	-
	A:	-	Status lesen (periodisch)	-
power_off	U:	-	-	-
	B:	-	RP	-
	A:	-	Status lesen (periodisch)	-
power_seq	U:	MIL timeout	-	-
	B:	-	RP	-
	A:	ChkPwr_EQM	ChkPwr_EQM	-
inverting	U:	MIL timeout	-	-
	B:	-	RP	-
	A:	ChkInv_EQM	ChkInv_EQM	-
error	U:	-	RESET, Zyklusende	-
	B:	-	RP	-
	A:	-	Reset_EQM, CleanUp_EQM	-
ready	U:	overrun etc.	-	'Evt_Set_Sollwert'
	B:	-	-	-
	A:	div. EQMs	-	CurrentS_EQM
busy	U:	overrun etc.	'Evt_Read_Istwert'	-
	B:	-	-	-
	A:	CleanUP_EQM	CurrentI_EQM	-

Tabelle 7: Zustandsübergangsdiagramm 2

Legende

- Die Priorität der Zustände (höchste Priorität zuerst): emergency, interlock, local, power_off und power_seq und inverting, 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.
- 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.
 - Inverter=i Polwender wird per Kommando geschaltet (i=1,-1).
- 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 (period.) Beim periodischen (oder zumindest regelmäßigen) Lesen des Status.
 - ..._EQM Innerhalb des EQMs ..._EQM.

Tabelle 8: Legende zu den Zustandsübergangsdiagrammen

10.2.2 CurrentI_EQM

Event: Evt_Uni_Ready.

Aktion: (Istwert lesen, genauer Ablauf, nicht UNILAC)

10.2.3 Turbo_CurrentS_EQM

Event: Evt_Prep_Next_Acc.

Aktion: (Sollwert setzen, genauer Ablauf, nur UNILAC)

10.2.4 Turbo_CurrentI_EQM

Event: Evt_Beam_On.

Aktion: (Istwert lesen, genauer Ablauf, nur UNILAC)

10.2.5 CleanUp_EQM

Event: (Entfällt für UNILAC); Evt_Ramp_Start (Rest).

Aktion: Aufräumarbeiten am Ende eines Zyklus. Steht der interne Zustand auf 'error', ist er auf 'ready' zu setzen.

10.2.6 Emerg_EQM

Event: Evt_Emergency.

Aktion: Internen Zustand auf 'Emergency' setzen. Sollwert Null setzen.

10.3 Periodisch konnektierte EQMs

10.3.1 CheckPower_EQM

Zeit: 2s

Anzahl: Einmalige Ausführung.

Aktion: Prüfen, ob Einschalten der Power fertig. Das Einschalten der Power kann bis zu 2s dauern.

10.3.2 CheckConnect_EQM

Zeit: 2s

Anzahl: Einmalige Ausführung.

Aktion: Prüfen, ob Anschalten einer Last an das NG geschehen ist. Das Anschalten einer Last kann bis zu 2s dauern.

10.3.3 CheckInverter_EQM

Zeit: 2s

Anzahl: Einmalige Ausführung.

Aktion: Prüfen, ob Inverter umgeschaltet ist. Das Umschalten des Inverters kann bis zu 2s dauern.

10.3.4 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.

10.4 An externe Interrupts konnektierte EQMs

10.4.1 Interlock_EQM

Interrupt: Summen-Interlock.

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

10.4.2 DRD_EQM

Interrupt: Data Ready Interrupt.

Aktion: Keine. Sollte bei MX nicht vorkommen.

10.4.3 DRQ_EQM

Interrupt: Data Request Interrupt.

Aktion: Keine. Sollte bei MX nicht vorkommen.

10.5 Kommandogetriggerte EQMs

10.5.1 Dev_Init_EQM

10.5.2 Dev_Reset_EQM

10.5.3 Status_EQM

10.5.4 Active_EQM

10.5.5 Power_EQM

10.5.6 Inverter_EQM

Schalten des Polwenders, nur wenn er nicht wie gewünscht steht und die Power aus ist.

10.5.7 MagnSvcS_EQM

10.5.8 MagnSvcI_EQM

10.6 EQMs für die Diagnose vor Ort

10.6.1 Display_DPR_EQM

Parameter: Das EQM benötigt 2 Parameter.

1. virtueller Beschleuniger (in Hex angeben)
2. 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.

10.6.2 Display_DevErr_EQM

Parameter: Das EQM benötigt 2 Parameter.

1. virtueller Beschleuniger (in Hex angeben)
2. logische Gerätenummer (in Hex angeben)

Daten: Keine.

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

10.7 Sonstige EQMs

10.7.1 Startup_EQM

Installiert die Event-EQM-Konnektierung für alle virtuellen Beschleuniger (siehe hierzu auch Abschnitt 4.4 auf Seite 13) und schaltet die SE in den Event-Mode.

10.8 Globale Routinen

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

10.8.1 Read_and_Update_Status

10.8.2 Set_InternalState

10.8.3 Do_Intr_Service_Prep

10.8.4 Power_on

10.8.5 Power_off

10.8.6 Set_Active_State

10.9 MIL-Treiber

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

11 Varianten

Zur Zeit sind 19 EQM-Varianten für unterschiedlichen Beschleunigerbetrieb realisiert. Weitere Kombinationen sind möglich.

Die Unilac-Varianten 3 und 7 werden seit der Hochstromerweiterung des Unilac nicht mehr benötigt.

Die HEST-Varianten 18 und 19 werden nicht mehr benötigt, da alle fraglichen Magnete ein eigenes Netzgerät bekommen haben.

Die Varianten im Einzelnen:

- Variante 1 (MX\$PERMANENT\$SIS\$24BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an die Events des SIS-Timings zur SIS-Extraktion konnektiert und lesen 24 Bit Hardwarestatus.
- Variante 2 (MX\$SHARED\$SIS\$24BIT) Ein Magnet teilt sich ein Netzteil mit anderen Magneten (shared). Die EQMs sind an die Events des SIS-Timings zur SIS-Extraktion konnektiert und lesen 24 Bit Hardwarestatus.
- Variante 3 (MX\$PERMANENT\$UNI\$8BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an das UNILAC-Timing konnektiert und lesen 8 Bit Hardwarestatus. Zusätzlich ist der Summeninterlock disabled (wg. uralten Interfacekarten).
- Variante 4 (MX\$PERMANENT\$SIS\$8BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an die Events des SIS-Timings zur SIS-Extraktion konnektiert und lesen 24 Bit Hardwarestatus.
- Variante 5 (MX\$PERMANENT\$TK9\$24BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an die Events des SIS-Timings zur SIS-Injektion konnektiert und lesen 24 Bit Hardwarestatus.
- Variante 6 (MX\$PERMANENT\$ECOOOL\$24BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an die Events des ESR-Timings zur Steuerung des Elektrokühlers konnektiert und lesen 24 Bit Hardwarestatus.
- Variante 7 (MX\$PERMANENT\$UNI\$24BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an das UNILAC-Timing konnektiert und lesen 8 Bit Hardwarestatus.

- Variante 8 (MX\$PERMANENT\$SIS\$E\$24BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an die Events des SIS-Timings zur SIS-Extraktion konnektiert, verarbeiten während der Therapie energie-abhängige Sollwerte und lesen 24 Bit Hardwarestatus.
- Variante 9 (MX\$PERMANENT\$SIS\$EF\$24BIT) Ein Magnet ist permanent mit seinem Netzteil verbunden. Die EQMs sind an die Events des SIS-Timings zur SIS-Extraktion konnektiert, verarbeiten während der Therapie energie- und fokus-abhängige Sollwerte und lesen 24 Bit Hardwarestatus.
- Variante 10 (MX\$PERMANENT\$HTM\$E\$24BIT) Variante für HTMMU1 und HTMMU2. Sie können auch mit dem „Event Connected Command“ ECC_MNGEIN eingeschaltet werden. Ansonsten sind es permanent mit einem Netzteil verbundene Magnete, deren EQMs an die Events des SIS-Timings zur SIS-Extraktion konnektiert sind, die während der Therapie energie-abhängige Sollwerte verarbeiten und die 24 Bit Hardwarestatus lesen.
- Variante 11 (MX\$PERMANENT\$SIS\$DEG) Degaussing-Magnete mit sehr eingeschränkter Bedienmöglichkeit. Im Grunde kann man nur Status und Istwerte lesen.
- Variante 12 (MX\$PERMANENT\$ESR\$INJ) Variante für das ESR-Injektionsseptum E01MU4I. Dieses realisiert zwei „Schüsse“ in einem virtuellen Beschleuniger und hat daher zwei Istwerte statt einem.
- Variante 13 (MX\$PERMANENT\$SIS\$MK1) Spezielle Variante für den Spillabbruch-Kicker TH3MK1. Istwerte werden von einem ADC gelesen, der hardware-mäßig getriggert wird und per Software wieder zurückgesetzt wird.
- Variante 14 (MX\$PERMANENT\$HSI\$QUAD) Magnet ist ein Quadrupol im Hochstrominjektor HSI. Er benötigt spezielle Behandlung der Soll- und Istwerte bezüglich dem Unilac-Timing und zusätzliche, von $B\rho$ abhängige, Default-Sollwerte am Anfang und Ende eines 20ms-Zyklus. ADCs und DACs werden mit MIL-Broadcasts betrieben.
- Variante 15 (MX\$PERMANENT\$HSI\$STEER) Magnet ist ein Steerer im Hochstrominjektor HSI. Er benötigt spezielle Behandlung der Soll- und Istwerte bezüglich dem Unilac-Timing und zusätzliche, von $B\rho$ abhängige, Default-Sollwerte am Anfang und Ende eines 20ms-Zyklus. ADCs und DACs werden mit MIL-Broadcasts betrieben.
- Variante 16 (MX\$PERMANENT\$UNI\$POST) Magnet liegt im Poststripperbereich des Unilac. Er benötigt spezielle Behandlung der Soll- und Istwerte bezüglich dem Unilac-Timing. ADCs und DACs werden mit MIL-Broadcasts betrieben.
- Variante 17 (MX\$PERMANENT\$TK) Magnet liegt im Transferkanal zwischen Unilac und SIS. Er benötigt spezielle Behandlung der Soll- und Istwerte bezüglich dem Unilac-Timing. ADCs und DACs laufen frei und werden ohne MIL-Broadcasts betrieben.
- Variante 18 (MX\$SHARED\$SIS\$E\$24BIT) Ein Magnet teilt sich ein Netzteil mit anderen Magneten (shared). Die Magnete können aber nicht per Rechner umgeschaltet werden, das muss mit einem Schraubenschlüssel manuell gemacht werden. Die EQMs sind an die Events des SIS-Timings zur SIS-Extraktion konnektiert, verarbeiten während der Therapie energie-abhängige Sollwerte und lesen 24 Bit Hardwarestatus.
- Variante 19 (MX\$SHARED\$SIS\$EF\$24BIT) Ein Magnet teilt sich ein Netzteil mit anderen Magneten (shared). Die Magnete können aber nicht per Rechner umgeschaltet werden, das muss mit einem Schraubenschlüssel manuell gemacht werden. Die EQMs sind an die Events des SIS-Timings zur SIS-Extraktion konnektiert, verarbeiten während der Therapie energie- und fokus-abhängige Sollwerte und lesen 24 Bit Hardwarestatus.

Literatur

- [1] Ludwig Hechler. Therapieprojekt: Die $G\mu P$ -Ebene. Accelerator Controls Documentation F-MBP-01, Gesellschaft für Schwerionenforschung, Darmstadt, April 1995. (Source: dd_gup.tex).
- [2] Ludwig Hechler. Laufzeiten der MX-EQMs, Version 1. Accelerator Controls Note, Gesellschaft für Schwerionenforschung, Darmstadt, September 1998. (Source: hsi/timing_v1.tex).
- [3] Ludwig Hechler. Laufzeiten der MX-EQMs, Version 2. Accelerator Controls Note, Gesellschaft für Schwerionenforschung, Darmstadt, November 1998. (Source: hsi/timing_v2.tex).
- [4] Ludwig Hechler. Laufzeitmessungen der MX-EQMs im Hochstromtiming. Accelerator Controls Note, Gesellschaft für Schwerionenforschung, Darmstadt, September 1998. (Source: hsi/messungen_v1.tex).
- [5] Ludwig Hechler. Laufzeitmessungen der MX-EQMs im Hochstromtiming, Ergänzungen. Accelerator Controls Note, Gesellschaft für Schwerionenforschung, Darmstadt, Oktober 1998. (Source: hsi/messungen_v2.tex).
- [6] Ludwig Hechler. Laufzeitmessungen der MX-EQMs im Hochstromtiming, Weitere Ergänzungen. Accelerator Controls Note, Gesellschaft für Schwerionenforschung, Darmstadt, Dezember 1998. (Source: hsi/messungen_v3.tex).
- [7] Ludwig Hechler. Timing gepulster Magnete. Accelerator Controls Note, Gesellschaft für Schwerionenforschung, Darmstadt, August 1998. (Source: timing/messungen.tex).
- [8] Ludwig Hechler. Laufzeiten der MX-EQMs mit 4ms-Delay-Erweiterung. Accelerator Controls Note, Gesellschaft für Schwerionenforschung, Darmstadt, März 1999. (Source: hsi/timing_v3.tex).
- [9] Peter Kainberger. Therapieprojekt: Die SE-Ebene. Accelerator Controls Documentation F-MBP-02, Gesellschaft für Schwerionenforschung, Darmstadt, Mai 1995. (Source: dd_se.tex).

Index

—A—	
A/D-Wandler	8
Active_EQM	34
ADC	8
An Interrupts konnektierte EQMs	34
Aufgabe des Gerätes	7
Ausschalten	13
—B—	
Bedienungsfehler	16
—C—	
CheckConnect_EQM	33
CheckInverter_EQM	33
CheckPower_EQM	33
CleanUp_EQM	33
CurrentLEQM	33
CurrentSEQM	31
—D—	
D/A-Wandler	8
DAC	8
Datenbasis	27
Dev_Init_EQM	34
Dev_Reset_EQM	34
Display_DevErr_EQM	35
Display_DPR_EQM	35
DRD Interrupt	9
DRD_EQM	34
DRQ Interrupt	9
DRQ_EQM	34
Dualport RAM	27
—E—	
Einschalten	12
Emerg_EQM	33
Emergency-Event	15
EQMs	30
• An Interrupts konnektierte	34
– DRD_EQM	34
– DRQ_EQM	34
– Interlock_EQM	34
• Eventkonnektierte	31
– CleanUp_EQM	33
– CurrentLEQM	33
– CurrentSEQM	31
– Emerg_EQM	33
– Turbo_CurrentLEQM	33
– Turbo_CurrentSEQM	33
• für die Diagnose vor Ort	35
– Display_DevErr_EQM	35
– Display_DPR_EQM	35
• Globale Routinen	35
– Do_Intr_Service_Prep	36
– Power_off	36
– Power_on	36
– Read_and_Update_Status	36
– Set_Active_State	36
– Set_InternalState	36
• Kommandogetriggerte	34
– Active_EQM	34
– Dev_Init_EQM	34
– Dev_Reset_EQM	34
– Inverter_EQM	34
– MagnSvcLEQM	35
– MagnSvcSEQM	35
– Power_EQM	34
– Status_EQM	34
• MIL-Treiber	36
• Periodisch konnektierte	33
– CheckConnect_EQM	33
– CheckInverter_EQM	33
– CheckPower_EQM	33
– Update_Config_EQM	34
• Sonstige	35
– Startup_EQM	35
• Varianten	36
Event-Overrun	15
Event-Sequenzfehler	15
Eventkonnektierte EQMs	31
Eventkonnektierungen	13
—F—	
Funktionscodes	8
• ifb_power_off	8
• ifb_power_on	8
• ifb_reset	8
—G—	
Genauigkeitsanforderungen	13

Gerät	Lokalen Datenbasis
• Aufgabe 7	• Tabelle der Konstanten 27
• Hardware 7	
• Repräsentation 16	
• Schnittstelle 8	—M—
• Steuerung 10	MagnSvcLEQM 35
Gerätemodell 7	MagnSvcSEQM 35
• Kennzeichnung 16	Master-Properties 16
• Master-Properties 16	MIL-Treiber 36
• Slave-Properties 20	
Gerätevarianten 7	—N—
• bipolar 7	N_Init 28
• Lastumschaltung 7	N_Reset 28
• Polwender 7	Normalbetrieb 10
• unipolar 7	Normierung 8, 10
Globale Routinen 30, 35	
—H—	—O—
Handbetrieb 15	Overrun 15
Hardware des Gerätes 7	
Hardwarefehler-Bit 15	—P—
Hardwarestatus 9	Periodisch konnektierte EQMs 33
—I—	Polwender 11
ifb_power_on, ifb_power_off 8	Power_EQM 34
ifb_reset 8	Properties
Init 14	• ACTIV 21
Interfacekarte 8	• CALC 18
• MIL-Treiber 36	• CONNECT 18
Interlock 9, 15	• CONSTANT 19
Interlock_EQM 34	• COPYSET 21
Interne Zustände 30	• CURRENTI 22
Interrupt	• CURRENTS 22
• DRD Interrupt 9	• DSTATUS 22
• DRQ Interrupt 9	• EQMERROR 21
• Interlock 9	• FIELDI 22
Inverter_EQM 34	• FIELDS 23
Istwert 10	• INFOSTAT 16
—K—	• INIT 17
Kaltstarts 14	• INVERTER 19
Kommandogetriggerte EQMs 34	• MAGNINFO 23
—L—	• MAGNSVCI 20
Lastumschaltung 11	• MAGNSVCS 20
Lokale Datenbasis 27	• Master- 16
	• MEDDATAI 24
	• MEDDATAS 24
	• POWER 17
	• RESET 17
	• Slave- 20
	• STATUS 18

- VERSION 18
- VOLTI 25
- VOLTS 24

- Turbo_CurrentI_EQM 33
- Turbo_CurrentS_EQM 33

—R—

- R_Active 28
- R_Constant 30
- R_CurrentI 29
- R_CurrentS 29
- R_EQMErr 28
- R_FieldI 28
- R_FieldS 28
- R_InfoStat 28
- R_Inverter 30
- R_MagnSvcI 29
- R_MagnSvcS 29
- R_PotiInfo 30
- R_Power 28
- R_Status 28
- R_Version 28
- R_VoltI 29
- R_VoltS 29
- Repräsentation des Gerätes 16
- Reset 15

—S—

- Schnittstelle zum Gerät 8
- Sequenzfehler 15
- Slave-Properties 20
- Softwareentwurf 27
- Softwarestatus 9
- Sollwert 10
- Sonstige EQMs 35
- Störungen 15
 - Emergency-Event 15
 - Event-Overrun 15
 - Event-Sequenzfehler 15
 - Interlock 15
 - Kommunikation EC – Gerät 16
- Standardzyklus SIS 13
- Startup_EQM 35
- Startwerte 14
- Status_EQM 34
- Statusbits 9
- Steuerung des Gerätes 10

—T—

- Timing 13

—U—

- Update_Config_EQM 34
- USRs 28
 - gerätespezifische 28
 - R_Constant 30
 - R_CurrentI 29
 - R_CurrentS 29
 - R_FieldI 28
 - R_FieldS 28
 - R_Inverter 30
 - R_MagnSvcI 29
 - R_MagnSvcS 29
 - R_PotiInfo 30
 - R_VoltI 29
 - R_VoltS 29
 - W_CurrentS 29
 - W_FieldS 28
 - W_Inverter 30
 - W_MagnSvcS 29
 - W_VoltS 29
 - Globale Routinen 30
 - obligatorische 28
 - N_Init 28
 - N_Reset 28
 - R_Active 28
 - R_EQMErr 28
 - R_InfoStat 28
 - R_Power 28
 - R_Status 28
 - R_Version 28
 - W_Active 28
 - W_CopySet 28
 - W_Power 28

—V—

- Varianten 36
 - Betriebs- 13
 - EQM- 36
 - Geräte- 7

—W—

- W_Active 28
- W_CopySet 28
- W_CurrentS 29
- W_FieldS 28

W_Inverter	30
W_MagnSvcS	29
W_Power	28
W_VoltS	29
Wandler	
• D/A- und A/D-	8
Warmstarts	15

—Z—

Zeitkritische Anforderungen	13
Zustände	
• Interne	30
– Übergänge	31
– Bedeutung	30
– Standard-Übergänge	31