



**HVDM -
Multi-Hochspannungs-Netzgeräte
Gerätemodell und Softwareentwurf**

P. Kainberger

Dieses Papier enthält die Beschreibung des Gerätemodells „HVDM - Multi-Hochspannungs-Netzgeräte“ und den Entwurf der Gerätesoftware für dieses Gerät.

Überarbeitungen des Dokuments			
Datum	GM-Version	Name	Kommentar
29. Jan. 92	–	L. Hechler	Rahmen erstellt
13. Feb. 92	–	P. Kainberger	Adaptierung für HVDM
24. Jul. 92	–	P. Kainberger	diverse Korrekturen
27. Aug. 92	–	P. Kainberger	Korrekturen und Ergänzungen
20. Sep. 92	–	P. Kainberger	Korrekturen und Ergänzungen
April 2001	–	M. Kühn	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
2.1	Gerätevarianten	7
3	Die Schnittstelle zum Gerät	7
3.1	Funktionscodes der Interfacekarte	8
3.2	Adressen des VME-Interfaces	10
3.3	Interlock Interrupt	13
3.4	Data Request (DRQ) Interrupts	14
3.5	Data Ready (DRD) Interrupts	14
3.6	Definition der Bits des Hardwarestatus	14
3.7	Onlinetest	14
4	Die Bedienung des Gerätes	15
4.1	Aufgaben im Normalbetrieb	15
4.1.1	Soll- und Istwert	15
4.1.2	Einschalten	15
4.1.3	Ausschalten	15
4.2	Genauigkeitsanforderungen	15
4.3	Zeitkritische Anforderungen	15
4.4	Einordnung in das Timing	15
4.5	Festlegung von Startwerten	16
4.5.1	Kaltstarts	16
4.5.2	Warmstarts	16
4.6	Handbetrieb	16
4.7	Ableitung des Hardwarefehler-Bits aus dem Gerätestatus	16
4.8	Verhalten bei Störungen	16
4.8.1	Geräteinterlock	16
4.8.2	Event-Sequenzfehler	16
4.8.3	Event-Overrun	16
4.8.4	Emergency-Event	17
4.8.5	Ausfall der Kommunikation EC – Gerät	17
4.9	Bedienungsfehler vom Operating	17
4.10	Supergeräte	17
5	Die Repräsentation des Gerätes	17
5.1	Kennzeichnung des Gerätemodells	17
5.2	Die Master-Properties	18
5.2.1	POWER	18
5.2.2	STATUS	18
5.2.3	INIT	18
5.2.4	RESET	19
5.2.5	VERSION	19
5.2.6	INFOSTAT	19
5.2.7	CONSTANT	20
5.2.8	VOLTAGES	20
5.2.9	VOLTAGEI	20

5.2.10	CURRENTS	21
5.2.11	CURRENTI	21
5.2.12	RAMPRATE	21
5.2.13	TRIPTIME	22
5.3	Die Slave-Properties	22
5.3.1	ACTIV	22
5.3.2	EQMERROR	22
5.3.3	COPYSET	23
II Der Entwurf der Software		25
6	Softwareentwurf	25
7	Lokale Datenbasis	25
7.1	Tabelle der Konstanten	25
8	Dualport RAM	25
9	USRs - User Service Routinen	26
9.1	Obligatorische USRs	26
9.1.1	N_Init	26
9.1.2	N_Reset	26
9.1.3	R_Status	26
9.1.4	R_Power	26
9.1.5	W_Power	26
9.1.6	R_Active	26
9.1.7	W_Active	26
9.1.8	W_CopySet	26
9.1.9	R_EQMErr	26
9.1.10	R_Version	26
9.1.11	R_InfoStat	26
9.2	Gerätespezifische USRs	26
9.2.1	W_VoltageS	27
9.2.2	R_VoltageS	27
9.2.3	R_VoltageI	27
9.2.4	W_CurrentS	27
9.2.5	R_CurrentS	27
9.2.6	R_CurrentI	27
9.2.7	W_RampRate	27
9.2.8	R_RampRate	27
9.2.9	W_TripTime	27
9.2.10	R_TripTime	27
9.2.11	R_Constant	27
9.3	Globale Routinen	27
9.3.1	get_device_parameter	27
9.3.2	get_super_constants	27
9.3.3	get_device_constants	27

10 EQMs - Equipment Module	27
10.1 Interne Zustände	27
10.1.1 Bedeutung der internen Zustände	27
10.1.2 Übergänge zwischen den Zuständen	27
10.1.3 Standard-Zustandsübergänge	27
10.2 Eventkonnectierte EQMs	27
10.2.1 Emerg_EQM	27
10.3 Periodisch konnectierte EQMs	29
10.3.1 Update_Config_EQM	29
10.4 An externe Interrupts konnectierte EQMs	29
10.5 Kommandogetriggerte EQMs	29
10.5.1 Dev_Init_EQM	29
10.5.2 Dev_Reset_EQM	29
10.5.3 Status_EQM	29
10.5.4 Active_EQM	29
10.5.5 Power_EQM	29
10.5.6 Set_Soll_EQM	29
10.5.7 Get_Ist_EQM	29
10.6 EQMs für die Diagnose vor Ort	30
10.6.1 Display_DPR_EQM	30
10.6.2 Display_DevErr_EQM	30
10.7 Sonstige EQMs	30
10.7.1 Startup_EQM	30
10.8 Globale Routinen	30
10.8.1 Read_and_Update_Status	30
10.8.2 Do_Intr_Service_Prep	30
10.8.3 VME_Select	30
10.8.4 VME_Write	30
10.8.5 VME_Read	30
10.8.6 Check_VME_Error	31
10.8.7 check_access	31
10.8.8 Select_Target	31
10.8.9 Select_Protect	31
10.8.10 write_value	31
10.8.11 read_value	31
11 Varianten	31
12 Besonderheiten	31
Index	33

Teil I

Das Gerätemodell

1 Die Aufgabe des Gerätes

Für die Versorgung von diversen Strahldiagnose-Elementen mit Hochspannung werden in zunehmendem Maße Hochspannungs-Netzgeräte der Firma *CAEN* verwendet. Diese Netzgeräte (die für verschiedene Leistungsanforderungen unterschiedlich dimensioniert sein können) sind als einzelne HV-Module (bis zu 40 Stück) in sog. 'Crates' zusammengefaßt. In den Crates wiederum sitzt ein Controller, der die einzelnen HV-Module versorgt und überwacht ('lokale Intelligenz'). Bis zu 100 Crates können über eine serielle Schnittstelle ('Serial Line' mit 48 kBaud Übertragungsrate und etwas eigenwilligem CAENET-Protokoll) an einem einzigen VME-Controller angeschlossen werden.

2 Die Hardware des Gerätes

Jedes HV-Modul kann zwei verschiedene Spannungssollwerte (V_0 , V_1) haben, von denen einer über ein externes Signal (VSEL) ausgewählt wird. Ebenso können zwei verschiedene Strombegrenzungen (I_0 und I_1 , über das externe Signal ISEL wählbar) eingestellt werden. Bei einer Sollwertänderung 'fährt' der Crate-Controller das HV-Module selbständig mit einer Rampe (Steprate in Volt/s und Maximalstrom sind für beide Richtungen getrennt einstellbar) auf den neuen Sollwert. Dabei kann eingestellt werden wie lange der Maximalstrom überschritten werden darf (TRIP-time), bevor automatisch abgeschaltet wird. Ist eine automatische Abschaltung erfolgt, so kann man dies im Gerätestatus erkennen (TRIP OFF), das HV-Modul muß dann explizit wieder eingeschaltet werden.

Von jedem einzelnen HV-Modul sind die Leistungsdaten (Maximalspannung und Maximalstrom) und die aktuelle Einstellung aus einem EEPROM auslesbar (siehe hierzu auch Abschnitt 3.2 auf Seite 13). Ebenso kann über einen Modulstatus festgestellt werden, ob ein HV-Modul vorhanden und in Ordnung ist. Damit ist eine Autokonfiguration im Sinne des Kontrollsystems möglich (siehe hierzu Abschnitt 3.7 auf Seite 14), wobei jedes HV-Modul auf ein Gerät und damit auch eine Nomenklatur abgebildet wird.

In manchen Spezialanwendungen (z.B. bei gasgefüllten Gittern) werden mehr als eine Hochspannung für ein einzelnes Gerät benötigt (bei gasgefüllten Gittern 3). Für diesen Fall müssen mehrere HV-Module zu einem 'Supergerät' zusammengefaßt werden, wobei für das Sollwert setzen und das Ein/Aus-Schalten der einzelnen Module eine bestimmte Reihenfolge einzuhalten ist. Siehe hierzu auch Abschnitt 4.10 auf Seite 17.

2.1 Gerätevarianten

Bisher erscheint es nicht notwendig, Gerätevarianten zu unterscheiden. Möglicherweise aber doch, wenn sich bei der Definition der auszulösenden Aktionen beim Auftreten eines Emergency-Events herausstellt, daß in diesem Punkt mehrere Varianten notwendig sind.

3 Die Schnittstelle zum Gerät

Die Schnittstelle zu den HV-Modulen bilden eine Interfacekarte mit VME-Piggy und der VME-Controller A200 von CAENET, wobei jedes HV-Modul als einzelnes Gerät mit eigener Nomenklatur abgebildet wird. Siehe auch *Technical Information Manual Mod. A200 H.V. CAENET VME-Controller*

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.

Da die HV-Module über eine Interfacekarte mit VME-Piggy (F443) angeschlossen sind, beschränken sich die verwendeten Funktionscodes auf diejenigen, die man zur Bedienung des VME-Bus über das VME-Piggy benötigt. Die einzelnen Module und ihre Teilfunktionen erreicht man über verschiedene VME-Adressen (siehe hierzu Abschnitt 3.2 auf Seite 10).

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

Funktionscode		Modus	Bedeutung
Name	hex		
ifb_wr_data_w	06	Schreiben	Datenwort auf VME-Bus schreiben
ifb_wr_data_b	07	Schreiben	Datenbyte auf VME-Bus schreiben
ifb_high_addr	08	Schreiben	Adressmodifier und A16-A23 setzen
ifb_low_addr	0A	Schreiben	A1-A15 setzen
ifb_VME_Intr_Mask	0C	Schreiben	VME-Interrupt-Maske setzen
ifb_VME_Reset	0E	Funktion	VME-Reset
ifb_IFK_intr_mask	12	Schreiben	Interruptmaske der Interfacekarte setzen
ifb_rd_data_w	80	Lesen	Datenwort von VME-Bus lesen
ifb_rd_data_b	81	Lesen	Datenbyte von VME-Bus lesen
ifb_rd_VME_Intr	84	Lesen	VME-Interruptregister lesen
ifb_rd_VME_Intr_Mask	86	Lesen	VME-interrupt-Maske lesen
ifb_rd_VME_Intr_Vect	88	Lesen	VME-Interrupt-Vektor lesen
ifb_rd_Ident	8E	Lesen	Identifikationsregister lesen
ifb_rd_IFK_Intr	C9	Lesen	Interruptstatus der Interfacekarte lesen

ifb_wr_data_w

Ein 16 Bit Datenwort wird auf den VME-Bus geschrieben. Die richtige Adresse muß vorher bereits eingestellt worden sein.

ifb_wr_data_b

Ein 8 Bit DatenByte (Low-byte im übertragenen Wort bei ungeraden Adressen, High-byte wenn die VME-Adresse gerade ist) wird auf den VME-Bus geschrieben. Die richtige VME-Adresse muß vorher bereits eingestellt worden sein.

ifb_low_addr

Die Bits 1 bis 15 der VME-Adresse werden geschrieben.

ifb_high_addr

Die Bits 16 bis 23 der VME-Adresse (Bits 0 bis 7 im übertragenen Wort) und der Adreßmodifier AM0 bis AM5 (Bits 8 bis 13 im übertragenen Wort) werden geschrieben. Der Adreßmodifier ist beim CAENET VME-Controller auf $3D_{hex}$ eingestellt.

ifb_VME_Intr_Mask

Die VME-Interrupt-Maske wird geschrieben. Die Bits der Interruptmaske sind wie folgt belegt:

Bit	Bedeutung
0	Interrupt-Request 1
1	Interrupt-Request 2
2	Interrupt-Request 3
3	Interrupt-Request 4
4	Interrupt-Request 5
5	Interrupt-Request 6
6	Interrupt-Request 7
7	Timeout oder Bus-Error

ifb_VME_Reset

Ein Reset für den VME-Bus und das VME-Piggy wird ausgeführt.

ifb_IFK_intr_mask

Die Interrupt-Maske der Interfacekarte wird gestzt.

ifb_rd_data_w

Ein 16 Bit Datenwort wird vom VME-Bus gelesen. Die richtige VME-Adresse muß bereits vorher eingestellt worden sein.

ifb_rd_data_b

Ein 8 Bit Datenbyte wird vom VME-Bus gelesen. Die richtige VME-Adresse muß bereits vorher eingestellt worden sein. Das gelesene Datenbyte ist bei einem Zugriff auf eine ungerade VME-Adresse in den unteren 8 Bit des übertragenen Wortes zu finden, bei Zugriff auf eine gerade VME-Adresse in den oberen 8 Bit.

ifb_rd_VME_Intr

Das VME-Interruptregister wird gelesen. Zusätzlich wird im Highbyte des übertragenen Datenwortes das Statusbyte des VME-Piggy übertragen.

VME-Piggy-Status:

Bit	Bedeutung
8	RDY (Ready-Interrupt-Signal)
9	REQ (Request-Interrupt-Signal)
10	TO (VME Timeout)
11	BERR (VME Buserror)
12	DTACK (VME Data Acknowledge)
13..15	unbenutzt

Die Belegung des Interruptregisters ist identisch mit der Belegung des Interrupt-Masken-Registers (siehe Tabelle auf Seite 8).

ifb_rd_VME_Intr_Mask

Das VME-Interrupt-Masken-Register wird gelesen. Zusätzlich wird im Highbyte des übertragenen Datenwortes das Statusbyte übertragen. Die Belegung des Interruptregisters ist identisch mit der Belegung des Interrupt-Masken-Registers (siehe Tabelle auf Seite 8).

ifb_rd_VME_Intr_Vect

Die VME-Interruptvektornummer wird gelesen. Durch Lesen der Vektornummer wird ein IACK (Interrupt-Acknowledge) am VME-Bus erzeugt.

ifb_rd_Ident

Das Identifikationsregister des VME-Piggy wird gelesen. Zusätzlich wird im Highbyte des übertragenen Datenwortes das Statusbyte übertragen. Das VME-Piggy hat die Identifikationsnummer 10_{hex} .

ifb_rd_IFK_Intr

Der Interruptstatus der Interfacekarte wird gelesen.

Bit	Bedeutung
0...7	Adresse der Interfacekarte
8...9	nicht belegt
10	Status der DRQ-Leitung (active low)
11	Status der DRD-Leitung (active low)
12	Status der SI-Leitung (active low)
13	DRQ-Maske (0: disabled)
14	DRD-Maske (0: disabled)
15	SI-Maske (0: disabled)

3.2 Adressen des VME-Interfaces

Die Bedienung des VME-Controllers erfolgt über mehrere VME-Adressen (eigentlich Adreßoffsets zur Grundadresse des VME-Controllers, siehe Abschnitt 3.7 *Onlinetest* auf Seite 14), deren Bedeutung nicht eindeutig festgelegt ist, sondern davon abhängt, welche Funktion aktuell auszuführen ist. Folgende Tabelle gibt einen Überblick, welche Adressen benutzbar sind und welche Bedeutung sie im Zusammenhang mit der angegebenen Funktion haben, als Modus ist angegeben, ob Daten von dieser Adresse gelesen (R) oder auf diese Adresse geschrieben (W) werden:

Adr. (hex)	Modus	Funktion	Bedeutung in der angegebenen Funktion
18	W	SELECT TARGET	Adresse des HV-Moduls (0...7: Modul, 8...15: Crate)
	W	START READOUT	$FF00_{hex}$ (Vorbereitung des Lesevorgangs)
1A	W	SELECT TARGET	“dummy”
	R	READ ERROR FLAG	Bit 0=0 gibt an, ob Schreiboperation erfolgreich war
	W	START READOUT	Gerätfunktion, deren aktueller Wert gelesen werden soll
1C	W	WRITE VALUE	16 Bit Sollwert
1E	W	WRITE PARAMETER	Gerätfunktion, die den Sollwert erhält
	R	READ ERROR FLAG	Bit 0=0 gibt an, ob Schreiboperation erfolgreich war
3C	R	READ VALUE	16 Bit Istwert
3E	R	READ ERROR FLAG	Bit 0=0 gibt an, ob von $3C_{hex}$ gelesene Daten gültig sind

Zwei grundlegende Aktionen werden zur Bedienung und Überwachung der HV-Module über den VME-Controller benötigt:

SET PARAMETER VALUE: (setze einen Geräteparameter auf einen bestimmten Wert)

SELECT TARGET
READ ERROR FLAG
WRITE VALUE
READ ERROR FLAG

READ PARAMETER VALUE: (lese den aktuellen Wert eines Geräteparameters)

SELECT TARGET
READ ERROR FLAG
START READOUT
READ VALUE(s)

Zu Beginn jeder der beiden Aktionen muß das HV-Modul selektiert werden. Dies geschieht durch folgende Befehlsfolge:

SELECT TARGET
READ ERROR FLAG

SELECT TARGET Durch Schreiben auf die VME-Adresse 18_{hex} (die Bits 0...7 selektieren das HV-Modul (0...39), die Bits 8...15 das Crate (0...99)) wird das gewünschte VME-Modul selektiert. Anschließend muß auf Adresse $1A_{hex}$ geschrieben werden, wobei die Daten nicht interpretiert werden.

Eine besondere Bedeutung kommt dem *TARGET* mit der Adresse 70_{dez} zu. Durch Selektieren dieser Adresse, kann man die "Protection Bits" eines Crates lesen oder schreiben. Folgende Protection Bits sind definiert:

Bit	Protection	Bedeutung	
		High (1)	Low (0)
0	Power ON Restart	enabled	disabled
1	Password enable	enabled	disabled
2	Keyboard enable	enabled	disabled
3	(Schreiben) Clear Alarm	Clear	–
	(Lesen) Alarmstatus	Alarm	no Alarm
4	VSEL	V1 active	V0 active
5	ISEL	I1 active	I0 active
6	Alarmstatus (wie Bit 3)	Alarm	no Alarm
7	HV-enable	enabled	disabled

Die Bits 4 bis 7 können nur gelesen werden.

READ ERROR FLAG Nach der *SELECT TARGET*-Operation muß durch Lesen von Adresse $1A_{hex}$ die Selektierung überprüft werden (Bit 0=0 im gelesenen Datum zeigt an, daß das Selektieren erfolgreich war).

Nachdem ein HV-Modul selektiert ist, können einzelne oder mehrere Parameter gesetzt werden:

WRITE VALUE
READ ERROR FLAG

oder

START READOUT
READ VALUE(s)

WRITE VALUE Auf die VME-Adresse $1C_{hex}$ ist der 16 Bit Sollwert zu schreiben, auf den der Geräteparameter gesetzt werden soll. Anschließend muß über Adresse $1E_{hex}$ der Geräteparameter angegeben werden, der den geschriebenen Sollwert erhalten soll. Folgende Geräteparameter sind möglich:

- 0 Hochspannung V0 in Volt (dieser Wert ist am Gerät aktiv, wenn VSEL=0 ist)
- 1 Hochspannung V1 in Volt (dieser Wert ist am Gerät aktiv, wenn VSEL=1 ist)
- 2 Strombegrenzung I0 in μ Ampere (dieser Wert ist am Gerät aktiv, wenn ISEL=0 ist)
- 3 Strombegrenzung I1 in μ Ampere (dieser Wert ist am Gerät aktiv, wenn ISEL=1 ist)
- 4 Maximale Steigung der Aufwärtsrampe in Volt/s
- 5 Maximale Steigung der Abwärtsrampe in Volt/s
- 6 Trip-Einstellung in 1/10s. Werte zwischen 1 und 9998 geben die Zeit an, wielange die Strombegrenzung (I0 oder I1) überschritten werden darf, bevor das Gerät mit der RAMP-DOWN-RATE automatisch auf 0 gefahren und abgeschaltet wird. Eine 0 bedeutet, daß die Strombegrenzung überhaupt nicht überschritten werden darf, während eine Zeit von 9999 eine unendlich lange Strombegrenzungsüberschreitung zuläßt.
- 7 Status bits (nur Bit 0 darf gesetzt werden zum Ein/Aus-Schalten)
- 8 Gruppenzugehörigkeit eines Moduls (wird nicht benutzt)

Bei V0, V1, I0 und I1 kann die Auflösung des Setzwertes durch Setzen von bit 14 im Sollwert verzehnfacht werden!

READ ERROR FLAG Durch Lesen von Adresse $1E_{hex}$ muß der Setzvorgang überprüft werden (Bit 0=0 im gelesenen Datum zeigt an, daß das Setzen erfolgreich war).

START READOUT Auf die VME-Adresse 18_{hex} ist der 16 Bit Wert $FF00_{hex}$ zu schreiben, wobei die unteren 8 Bit vom Controller ignoriert werden. Anschließend muß über Adresse $1A_{hex}$ der Geräteparameter angegeben werden, dessen aktuelle Einstellung gelesen werden soll. Folgende Geräteparameter können gelesen werden:

- 0 Hochspannung V0
- 1 Hochspannung V1
- 2 Strombegrenzung I0
- 3 Strombegrenzung I1
- 4 Steigung der Aufwärtsrampe
- 5 Steigung der Abwärtsrampe
- 6 Trip-Einstellung (max. Dauer der Strombegrenzungsüberschreitung)
- 7 8 Statusbits mit folgender Bedeutung:

Bit	Name	Bedeutung	
		High (1)	Low (0)
0	Power OFF	off	on
1	Power OFF by Trip	Trip off	normal Power off
2	Power ON	on	off
3	Over-Voltage	OVV	ok
4	Under-Voltage	UVV	ok
5	Over-Current	OVC	ok
6	undergoing ramp-up-phase	undergoing	ok
7	undergoing ramp-down-phase	undergoing	ok

Auf den Gerätestatus im Sinne des Kontrollsystems werden nur die Bits 0 bis 2 abgebildet (siehe Abschnitt 3.6 auf Seite 14). Alle anderen Bits haben mehr oder weniger dynamische Bedeutung und können deshalb nicht zu jeder Zeit für eine automatische Geräteüberwachung genutzt werden.

8 Gruppenzugehörigkeit eines Moduls (wird nicht benutzt)

9 Spannungswert VMON

10 Stromwert IMON

11,12 Konditionierungsparameter (nicht benutzt)

13 Modultyp (hieraus lassen sich die Grenzwerte eines Moduls erkennen). Folgende Modultypen sind verfügbar:

Model	Typ (hex)	Voltage max.	Current max.	Voltage resolution	Current resolution	Ramprate up/down	
A334P/N	1	2 kV	3 mA	0.5V	1 μ A	250 V/s	
A333P/N	2,E	3 kV	3 mA	1V	1 μ A	500 V/s	
A333P/N	3,F	4 kV	2 mA	1V	1 μ A	500 V/s	
A331P/N	4	8 kV	0.5 mA	2V	1 μ A	500 V/s	
A332P/N	5,C	6 kV	1 mA	2V	1 μ A	500 V/s	
A335P/N	6	800 V	0.5 mA	0.2V	0.2 μ A	50 V/s	
A431P/N	7,12	8 kV	0.2 mA	2V	0.1 μ A	500 V/s	
A432P/N	8	6 kV	0.2 mA	2V	0.1 μ A	500 V/s	
A435P/N	9	200 V	0.2 mA	0.1V	0.1 μ A	25 V/s	
A434P/N	A	2 kV	0.2 mA	0.5V	0.1 μ A	250 V/s	
A433P/N	B	4 kV	0.2 mA	1V	0.1 μ A	500 V/s	
k.A.	D	Special Module					
A436P/N	10	800 V	0.2 mA	0.2V	0.1 μ A	50 V/s	
A330P/N	13	10 kV	1 mA	3V	1 μ A	500 V/s	
A430P/N	16	10 kV	0.2 mA	3V	0.1 μ A	500 V/s	
A429P/N	17	15 kV	0.2 mA	4V	0.1 μ A	500 V/s	
A329P/N	18	15 kV	1 mA	4V	1 μ A	500 V/s	
k.A.	1F	I/O-Module					

Bit 7 gibt an, ob ein Modul positive (Bit 7=0) oder negative (Bit 7=1) Werte annehmen kann.

!!!! Achtung!!!!

Teilweise sind die Angaben in den Manuals widersprüchlich bezüglich der Normierung des Spannungswertes, deshalb wird man die richtigen Werte wohl praktisch ermitteln müssen !!!!!!!

14 nicht belegt

15 alle Parameter (0 bis 13) werden in einem Block gelesen

Anschließend muß durch Lesen von Adresse $1A_{hex}$ der Lesevorgang überprüft werden (Bit 0=0 im gelesenen Datum zeigt an, daß das Lesen erfolgreich war).

READ VALUE(s) Von der VME-Adresse $3C_{hex}$ wird der 16 Bit Istwert des gewählten Geräteparameters gelesen. Anschließend muß durch Lesen von Adresse $3E_{hex}$ überprüft werden, ob der Lesevorgang erfolgreich war (Bit 0=0 im gelesenen Datum). Beide Vorgänge müssen solange wiederholt werden, bis Bit 0 auf Adresse $3E_{hex}$ gleich 0 ist. Wielange das wirklich dauern kann geht aus den vorliegenden Unterlagen nicht hervor und muß wohl in der Praxis ermittelt werden.

3.3 Interlock Interrupt

Summeninterlock wird bei diesem Gerät nicht verwendet.

3.4 Data Request (DRQ) Interrupts

Ein Data Request Interrupt wird erzeugt, wenn ein VME-Reset durchgeführt worden ist. Dies ist z.B. lokal an der Interfacekarte per Knopfdruck möglich, aber auch wenn eine VME-Komponente einen VME-Reset ausführt (z.B. Watchdogkarte), wird dieser über den Data Request Interrupt an die SE mitgeteilt. Im speziellen Fall der Hochspannungsnetzgeräte muß darauf allerdings nicht reagiert werden.

3.5 Data Ready (DRD) Interrupts

Ein Data Ready Interrupt wird erzeugt, wenn ein maskierter VME-Interrupt, ein VME-Bus-Timeout oder ein VME-Bus-Error vorhanden ist. Im speziellen Fall der Hochspannungsnetzgeräte sind jedoch keine VME-Interrupts vorgesehen, VME-Bus-Timeout oder VME-Bus-Error werden nach jedem VME-Zugriff überprüft und evtl. Fehler als solche gemeldet.

3.6 Definition der Bits des Hardwarestatus

Das Gerät liefert 1 Byte Statusinformation. 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	on	off
9	Power OFF by Trip	no Trip off	Trip off
10	Alarm	no Alarm	Alarm
11 ... 31	unbenutzt	immer high	-

3.7 Onlinetest

Der VME-Controller erhält eine fest eingestellte Grundadresse ($3D010000_{hex}$, Adr.Modifier $3D_{hex}$, Adr.Offset 010000_{hex}), die in den oberen 16 Bit der VME-Adresse eingestellt werden muß (also $3D01_{hex}$), bevor man den VME-Controller und seine Sub-Adressen ansprechen kann. Ebenso wird vereinbart, daß die Adresse der Interfacekarte gleich dem obersten Byte des Adressoffset (also 01_{hex}) sein muß. Folglich darf pro SE nur 1 Interfacekarte mit VME-Piggy und pro Interfacekarte nur 1 VME-Controller angeschlossen werden. Für die Umsetzung von Crate-Adresse (0...99) plus HV-Modul-Adresse (0...39) auf 1...254 physikalische Geräteadressen im Sinne des Kontrollsystems, wird folgende Vereinbarung getroffen:

$$\text{physik. Geräteadresse} = (\text{Crate\#} * 40) + \text{Modul\#} + 1$$

wobei festgelegt wird, daß nur die Crate-Adressen 0...5 verwendet werden dürfen.

Der VME-Controller liefert nach der *SELECT TARGET*-Operation Fehlercodes, anhand derer festgestellt werden kann, ob das angewählte Crate vorhanden ist und ob ein HV-Modul existiert oder nicht:

Fehlercode	Bedeutung
0	ok, selektiertes HV-Modul existiert
-32	selektiertes HV-Modul existiert nicht
-256	selektiertes Crate existiert nicht

4 Die Bedienung des Gerätes

Hier wird beschrieben, wie das Gerät (die Hardware) bedient werden muß (die ‘dynamische’ Schnittstelle). Das beinhaltet die Anforderungen *vom* Gerät als auch die Anforderungen *an das* Gerät. Zudem werden Anfangszustände und das Verhalten bei Störungen festgelegt.

4.1 Aufgaben im Normalbetrieb

4.1.1 Soll- und Istwert

Der Sollwert kann als Spannung in Volt (Realwert) angegeben werden.

Der minimale und maximale *Spannungssollwert*, der vom Operating vorgegeben werden darf, ist modulabhängig und ebenso wie die Normierung der entsprechenden HV-Modulliste (Seite 13) zu entnehmen. Der Spannungsiswert wird ebenfalls in Volt (Realwert) angegeben.

4.1.2 Einschalten

Für die HV-Module müssen folgende Ein/Aus-Schaltsequenzen eingehalten werden:

1. Clear Alarm
2. Statusbit 0 setzen (1 für Einschalten, 0 für Ausschalten), alle anderen Bits müssen 0 sein
3. Status wiederholt lesen, bis das Modul den gewünschten Zustand erreicht hat. Wielange das wirklich dauern kann geht aus den vorliegenden Unterlagen nicht hervor und muß wohl in der Praxis ermittelt werden.

4.1.3 Ausschalten

Siehe Einschalten.

4.2 Genauigkeitsanforderungen

Die einzelnen HV-Module erfüllen die Genauigkeitsanforderungen, wie sie in der technischen Beschreibung der Module aufgeführt sind. Über das VME-Modul können die Sollwerte mit der in der Modul-Tabelle auf Seite 13 angegebenen Genauigkeit gesetzt werden.

4.3 Zeitkritische Anforderungen

Da die HV-Module nicht an der Puls-zu-Puls-Modulation (PPM) teilnehmen, sind bisher keine zeitkritischen Anforderungen bekannt.

4.4 Einordnung in das Timing

Das Gerät nimmt nicht an der PPM teil.

4.5 Festlegung von Startwerten

4.5.1 Kaltstarts

Bei einem Kaltstart werden folgende Aktionen durchgeführt:

- Es wird ein VME-Reset zur Initialisierung des VME-Piggy durchgeführt. Ein VME-Reset beeinflusst die einzelnen HV-Module nicht. Ein Reset der einzelnen Module ist nicht vorgesehen.
- Die Sollwerte für jedes HV-Modul werden auf Minimalwerte gesetzt.

4.5.2 Warmstarts

Bei einem Warmstart werden folgende Aktionen durchgeführt:

- Es wird ein VME-Reset zur Initialisierung des VME-Piggy durchgeführt.
- Von jedem HV-Modul wird die aktuelle Einstellung, die im Gerät gepuffert vorliegt, gelesen und als aktuelle Sollwerte übernommen.
- Die Istwerte und der Status werden aktualisiert.

4.6 Handbetrieb

Ein Handbetrieb der HV-Module ist Crate-weise über Terminal parallel zum Rechnerbetrieb möglich. Dabei ist jedoch zu beachten, daß der Handbetrieb nicht im Status ersichtlich ist und daß eine Betriebsart die andere solange ausschließt, bis die laufende Aktion am Gerät abgeschlossen ist. Um den Problemen, die dadurch zu erwarten sind (z.B. die SE macht gerade einen Konfigurationsupdate, während über Terminal der Status abgefragt wird) von vornherein aus dem Weg zu gehen, sollte auf einen Handbetrieb über Terminal so weit wie möglich verzichtet werden.

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
1	Trip OFF	0

Nach erfolgter Abschaltung eines Moduls durch die Trip-Überwachung, muß das Modul explizit wieder eingeschaltet werden, eine automatische Zuschaltung erfolgt nicht.

4.8 Verhalten bei Störungen

4.8.1 Geräteinterlock

Ein Geräte-Interlock ist nicht vorgesehen.

4.8.2 Event-Sequenzfehler

Gerät nimmt nicht am Eventbetrieb teil.

4.8.3 Event-Overrun

Gerät nimmt nicht am Eventbetrieb teil.

4.8.4 Emergency-Event

Was bei Empfang von Emergency-Events zu tun ist, hängt im wesentlichen vom Einsatzort der angeschlossenen HV-Module ab, deshalb sollten gleich zu behandelnde Module in möglicherweise mehreren Gerätemodellvarianten zusammengefaßt werden.

Allen HV-Modulen gemeinsam ist, daß Sollwert 0 mit der maximalen Rampensteigung “angefahren” werden soll.

4.8.5 Ausfall der Kommunikation EC – Gerät

Der Ausfall der Kommunikation zwischen EC und Gerät könnte, abgesehen von Hardware- oder Software-Fehlern, durch parallelen Handbetrieb eines HV-Moduls vorgetäuscht werden (siehe Abschnitt 4.6 auf Seite 16) und führt zu Timeout beim Aufbau der Kommunikation durch das Selektieren eines *TARGETS*.

Da ein Handbetrieb allerdings nur für Notfälle erlaubt sein soll, wird diese Unschönheit als tolerabel angesehen.

4.9 Bedienungsfehler vom Operating

Strom- und Spannungs-Sollwerte außerhalb des zulässigen Bereichs (siehe Liste der Modultypen auf Seite 13) sind zurückzuweisen. Für jedes HV-Modul kann eine minimale und eine maximale Rampensteilheit und eine maximal erlaubte Hochspannung definiert werden. Alle Sollwerte, die über diese individuellen Grenzen hinausgehen, werden ebenfalls zurückgewiesen.

4.10 Supergeräte

Bis zu 3 HV-Module können als ‘slave’-Geräte zu einem Supergerät zusammengefaßt werden. In der ‘constant’-Tabelle der VME-Datenbasis muß eingetragen werden, welche Geräte in welcher Bedienungsreihenfolge zu einem Supergerät gehören. Von den USRs aus werden die Supergerätkomponenten immer in dieser Reihenfolge bedient (Sollwert setzen, RESET, INIT, Einschalten, usw.), lediglich beim Ausschalten werden die Geräte in umgekehrter Reihenfolge bedient.

Alle Lese- und Schreibzugriffe auf ein Supergerät liefern bzw. erfordern immer Daten für alle Supergerätkomponenten. Auf einzelne Supergerätkomponenten kann nur lesend zugegriffen werden, Schreibversuche werden zurückgewiesen und sind damit nur über das Supergerät möglich.

Im Gerätemodell HVDM wird jedes HV-Modul auf eine Nomenklatur abgebildet. In der lokalen VME-Datenbasis steht ein Vermerk, ob dieses Modul unabhängig oder Teil eines Supergerätes ist. Bei Supergerätkomponenten können Soll- und Istwerte nur gelesen werden, jeder Schreibzugriff, wird abgewiesen. Einzelne HV-Module können zu Supergeräten mit eigener Nomenklatur zusammengefaßt werden. Bei Supergeräten steht in der VME-Datenbasis, aus welchen Geräteadressen ein Supergerät zusammengesetzt ist und in welcher Reihenfolge die Bedienung der einzelnen Komponenten erfolgen muß. Auf diesem Weg lassen sich die speziellen Anforderungen, die von den Netzgeräten der “Stelzer-Gitter” gestellt werden, erfüllen.

5 Die Repräsentation des Gerätes

Dieses Kapitel definiert das Gerätemodell, also wie das Gerät nach höheren Ebenen hin abgebildet wird. Es beschreibt die Schnittstelle zwischen Benutzerebene (Operatingprogrammen) und Geräteebene (Gerätehard- und -software).

5.1 Kennzeichnung des Gerätemodells

Das Gerätemodell hat die Bezeichnung **HVDM**. Die Gerätemodellnummer ist 26_{dez}.

5.2 Die Master-Properties

Master-Properties							
Property	Klasse	Parameter		Daten		Größe	
		Anz.	Typ	Anz.	Typ	Einh.	Exp.
POWER	R/W	0	–	1	BitSet16	1	0
STATUS	R	0	–	1	BitSet32	1	0
INIT	N	0	–	0	–	–	–
RESET	N	0	–	0	–	–	–
VERSION	RA	0	–	36	BitSet8	1	0
INFOSTAT	RA	0	–	25	BitSet32	1	0
CONSTANT	RA	0	–	30	RealF	1	0
VOLTAGES	RA/WA	0	–	6	RealF	V	1
VOLTAGEI	R	0	–	3	RealF	V	1
CURRENTS	RA/WA	0	–	6	RealF	A	-6
CURRENTI	R	0	–	3	RealF	A	-6
RAMPRATE	RA/WA	0	–	6	Integer16	V/s	1
TRIPTIME	R/W	0	–	3	Integer16	s	-1

5.2.1 POWER

Bedeutung: Gibt an, ob der Leistungsteil des Gerätes ein- oder ausgeschaltet ist bzw. werden soll. Ist das Gerät ein Supergerät, so werden seine Komponenten in der Reihenfolge eingeschaltet, wie die Geräte in der 'constant'-Tabelle der VME-Datenbasis eingetragen sind. Beim Ausschalten ist die Reihenfolge umgekehrt. Ein Supergerät wird nur dann als ein- bzw. ausgeschaltet gemeldet, wenn alle seine Komponenten ein- bzw. ausgeschaltet sind.

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.

5.2.2 STATUS

Bedeutung: Auslesen des 32bit Gerätestatus. Bei Supergeräten wird als Status die logische 'UND'-Verknüpfung der Status aller Einzelkomponenten geliefert.

Parameter: Keine.

Daten: Das 32bit Statuswort. Die Bits entsprechen den Statusbits, wie sie in Abschnitt 3.6 auf Seite 14 und in der Tabelle 3.6 auf Seite 14 erklärt sind.

5.2.3 INIT

Bedeutung: Initialisierung des Gerätes (Kaltstart). Für die dabei durchzuführenden Aktionen siehe Abschnitt 4.5.1 auf Seite 16. Bei Supergeräten werden alle Einzelkomponenten initialisiert.

Parameter: Keine.

Daten: Keine.

5.2.4 RESET

Bedeutung: Reset des Gerätes (Warmstart). Für die dabei durchzuführenden Aktionen siehe Abschnitt 4.5.2 auf Seite 16. Bei Supergeräten werden alle Einzelkomponenten reseted.

Parameter: Keine.

Daten: Keine.

5.2.5 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

5.2.6 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. Bei Supergeräten kann diese Property nicht benutzt werden.

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.

5.2.7 CONSTANT

Bedeutung: Lesen der gerätespezifischen Konstanten eines HV-Moduls.

Parameter: keine

Daten: 9 RealF

1. Geräteklasse (1: Master-Gerät, 2: Komponente eines Supergerätes 3: Supergerät)
2. Physikalische Geräteadresse
3. Nummer des Modultyp (siehe Seite 13)
4. minimale Spannung (in V)
5. maximale Spannung (in V)
6. maximaler Strom (in μA)
7. minimale Steilheit der Abwärtsrampe (in V/s)
8. maximale Steilheit der Rampe (in V/s)
9. maximale Auflösung der Spannungs-Sollwerte (in V)
10. maximale Auflösung der Strom-Sollwerte (in A)

5.2.8 VOLTAGES

Bedeutung: Setze oder lese die Spannungs-Sollwerte V0 und V1 (Siehe hierzu auch Abschnitt 2 auf Seite 7). Bei Supergeräten muß für jede Komponente des Supergeräts ein eigener Sollwert angegeben werden.

Parameter: keine

Daten: 2 bis 6 RealF

1. V0 in V (wenn VSEL=0) für ein Mastergerät oder die erste Komponente eines Supergeräts
2. V1 in V (wenn VSEL=1) für ein Mastergerät oder die erste Komponente eines Supergeräts
3. V0 für die zweite Komponente des Supergeräts
4. V1 für die zweite Komponente des Supergeräts
5. V0 für die dritte Komponente des Supergeräts
6. V1 für die dritte Komponente des Supergeräts

5.2.9 VOLTAGEI

Bedeutung: Lese den aktuellen Spannungs-Istwert VMON. Bei Supergeräten wird für jede Komponente des Supergeräts ein eigener Istwert geliefert.

Parameter: keine

Daten: 1 bis 3 RealF

1. Spannungs-Istwert VMON für ein Mastergerät oder die erste Komponente eines Supergeräts
2. VMON für die zweite Komponente des Supergeräts
3. VMON für die dritte Komponente des Supergeräts

5.2.10 CURRENTS

Bedeutung: Setze oder lese die Strombegrenzungswerte I0 und I1 (Siehe hierzu auch Abschnitt 2 auf Seite 7). Bei Supergeräten muß für jede Komponente des Supergeräts ein eigener Sollwert angegeben werden.

Parameter: keine

Daten: 2 bis 6 RealF

1. I0 in μ A (wenn ISEL=0) für ein Mastergerät oder die erste Komponente eines Supergeräts
2. I1 in μ A (wenn ISEL=1) für ein Mastergerät oder die erste Komponente eines Supergeräts
3. I0 für die zweite Komponente des Supergeräts
4. I1 für die zweite Komponente des Supergeräts
5. I0 für die dritte Komponente des Supergeräts
6. I1 für die dritte Komponente des Supergeräts

5.2.11 CURRENTI

Bedeutung: Lese den aktuellen Stromwert IMON. Bei Supergeräten wird für jede Komponente des Supergeräts ein eigener Istwert geliefert.

Parameter: keine

Daten: 1 RealF

1. aktueller Stromwert IMON für ein Mastergerät oder die erste Komponente eines Supergeräts
2. IMON für die zweite Komponente des Supergeräts
3. IMON für die dritte Komponente des Supergeräts

5.2.12 RAMPRATE

Bedeutung: Setze oder Lese die Steigung der Aufwärtsrampe und der Abwärtsrampe in V/s. Bei Supergeräten muß für jede Komponente des Supergeräts ein eigener Sollwert angegeben werden.

Parameter: keine

Daten: 2 bis 6 Integer16

1. Steigung der Aufwärtsrampe für ein Mastergerät oder die erste Komponente eines Supergeräts
2. Steigung der Abwärtsrampe für ein Mastergerät oder die erste Komponente eines Supergeräts
3. Steigung der Aufwärtsrampe für die zweite Komponente des Supergeräts
4. Steigung der Abwärtsrampe für die zweite Komponente des Supergeräts
5. Steigung der Aufwärtsrampe für die dritte Komponente des Supergeräts
6. Steigung der Abwärtsrampe für die dritte Komponente des Supergeräts

5.2.13 TRIPTIME

Bedeutung: Setze oder Lese die Trip-Einstellung in 1/10s. Werte zwischen 1 und 9998 geben die Zeit an, wie lange die Strombegrenzung (I0 oder I1) überschritten werden darf, bevor das Gerät mit der RAMP-DOWN-RATE automatisch auf 0 gefahren und abgeschaltet wird. Eine 0 bedeutet, daß die Strombegrenzung überhaupt nicht überschritten werden darf, während eine Zeit von 9999 eine unendlich lange Strombegrenzungsüberschreitung zuläßt. Bei Supergeräten muß für jede Komponente des Supergeräts ein eigener Sollwert angegeben werden.

Parameter: keine

Daten: 1 bis 3 Integer16

1. Triptime für ein Mastergerät oder die erste Komponente eines Supergeräts
2. Triptime für die zweite Komponente des Supergeräts
3. Triptime für die dritte Komponente des Supergeräts

5.3 Die Slave-Properties

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

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.

Da dieser Gerätetyp nicht im Event-Betrieb läuft, liefert der Versuch ACTIV zu Schreiben eine entsprechende Information.

Parameter: Keine.

Daten: immer 1

5.3.2 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

5.3.3 COPYSET

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

Da bei diesem Gerätetyp keine Beschleuniger-abhängigen Daten existieren, bleibt die Ausführung dieser Property ohne jegliche Wirkung.

Parameter: Keine.

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

Teil II

Der Entwurf der Software

6 Softwareentwurf

Zusammenfassung mehrerer Einzelgeräte zu einem Supergerät:

Bis zu 3 Einzelgeräte können zu einem Supergerät zusammengefaßt werden. Dies bedeutet jedoch nur, daß Schreibzugriffe auf Supergerätkomponenten nur über das Supergerät erlaubt sind. Lesezugriffe auf die Einzelkomponenten sind jedoch möglich. Bei einem Zugriff auf ein Supergerät müssen also Werte für alle Einzelkomponenten des Supergeräts gesetzt bzw. gelesen werden. In den jeweiligen USRs wird die Anzahl der empfangenen Daten mit der Anzahl der Supergerätkomponenten verglichen und ein ggf. ein Fehler gemeldet.

Auf Operating-Ebene muß darauf geachtet werden, daß die Anzahl der zu übertragenden Daten unterschiedlich sein kann, je nachdem ob das Gerät ein Mastergerät oder ein Supergerät ist.

Auf SE-Ebene werden alle Geräteklassen gleich behandelt, eine Unterscheidung nach Mastergeräten oder Supergeräten erfolgt nicht.

7 Lokale Datenbasis

7.1 Tabelle der Konstanten

Für jedes Gerät gibt es eine Beschreibung aus 6 Elementen in der Konstantentabelle der lokalen Datenbasis. Die Elemente haben in der Reihenfolge folgende Bedeutung:

Bei Supergeräten:

- 1:** Geräteklasse (1: Mastergerät, 2: Supergerätekomponente, 3: Supergerät)
- 2:** Anzahl der Supergerätekomponenten
- 3:** Geräteadresse der ersten Komponente
- 4:** Geräteadresse der zweiten Komponente
- 5:** Geräteadresse der dritten Komponente
- 6:** Reserve

Bei Mastergeräten:

- 1:** Geräteklasse (1: Mastergerät, 2: Supergerätekomponente, 3: Supergerät)
- 2:** CAEN-Modultyp-nummer
- 3:** Maximal erlaubte Hochspannung
- 4:** Maximal erlaubte Strombegrenzung
- 5:** Maximale erlaubte Steigung der Aufwärtsrampe
- 6:** Minimal erforderliche Steigung der Abwärtsrampe

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 Bedienung der CAEN-HV-Module folgende gerätespezifischen USRs benötigt:

- 9.2.1 W_VoltageS
- 9.2.2 R_VoltageS
- 9.2.3 R_VoltageI
- 9.2.4 W_CurrentS
- 9.2.5 R_CurrentS
- 9.2.6 R_CurrentI
- 9.2.7 W_RampRate
- 9.2.8 R_RampRate
- 9.2.9 W_TripTime
- 9.2.10 R_TripTime
- 9.2.11 R_Constant

9.3 Globale Routinen

- 9.3.1 get_device_parameter
- 9.3.2 get_super_constants
- 9.3.3 get_device_constants

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.
error	Während der Abarbeitung eines EQMs wurde ein Fehler erkannt.
ready	Das Gerät ist bereit für Aktionen. Ausgangszustand am Beginn eines virtuellen Beschleunigers.

10.1.2 Übergänge zwischen den Zuständen

Die Zustände und die Übergänge zwischen denselben sind in Tabelle 3 zusammengefaßt. Die Legende zu diesen Tabellen ist in Tabelle 4 zu finden.

10.1.3 Standard-Zustandsübergänge

Bei diesem Gerätetyp werden im Normalfall keine Zustandsübergänge vorgenommen.

ready -> ready

10.2 Eventkonnektierte EQMs

10.2.1 Emerg_EQM

Event: Evt_Emergency.

Tabelle der Zustandsübergänge				
von↓	nach→	emergency	error	ready
emergency	U:	–	–	RESET
	B:	–	–	P
	A:	–	–	Reset_EQM
error	U:	Evt_Emerg	–	div. EQMs
	B:	–	–	P
	A:	Emerg_EQM	–	Reset_EQM
ready	U:	Evt_Emerg	div. EQMs	–
	B:	–	–	–
	A:	Emerg_EQM	div. EQMs	–

Tabelle 3: Zustandsübergangsdiagramm

Legende

- Die Priorität der Zustände (höchste Priorität zuerst): emergency, error, ready.
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.
Evt_Emerg Pulszentrale verschickte Emergency-Event.
RESET Reset wird per Kommando oder Knöpfchendrücken ausgelöst.
- B: Abzuprüfende Bedingung.
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 4: Legende zu den Zustandsübergangsdiagrammen

Aktion: Internen Zustand auf 'Emergency' setzen. Hochspannung auf 0 fahren.

10.3 Periodisch konnektierte EQMs

10.3.1 Update_Config_EQM

Zeit: 60s

Anzahl: Unendlich.

Aktion: Aktualisieren der Geräteverfügbarkeit: Es wird versucht, von möglichen Geräteadressen den Onlinetest durchzuführen. Wird dieser Test erfolgreich ausgeführt, wird das Gerät als 'online' geführt.

10.4 An externe Interrupts konnektierte EQMs

Keine

10.5 Kommandogetriggerte EQMs

10.5.1 Dev_Init_EQM

Führt eine Geräteinitialisierung durch.

10.5.2 Dev_Reset_EQM

Führt einen Gerätereset durch.

10.5.3 Status_EQM

Liest den aktuellen Gerätestatus.

10.5.4 Active_EQM

Schaltet den Aktiv-Zustand eines Geräts um. Dies hat bei diesen DC-Geräten keine Auswirkung.

10.5.5 Power_EQM

Schaltet die Power eines Geräts ein oder aus.

10.5.6 Set_Soll_EQM

Setzt einen bestimmten Sollwert am Gerät.

10.5.7 Get_Ist_EQM

Liest den aktuellen Stromistwert, den aktuellen Spannungswert und die Protection-Bits eines Geräts.

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

Schaltet die SE in den Command-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

List den aktuellen Status vom Gerät und trägt ihn im Dualport-RAM ein.

10.8.2 Do_Intr_Service_Prep

Vorbereitung der Interrupt-Service-Routinen.

10.8.3 VME_Select

Selektiert den CAEN-VME-Controller am VME-Bus durch setzen des VME-Adreß-Modifiers und der Adreßbits A16 bis A23.

10.8.4 VME_Write

Schreibt Daten auf den VME-Bus.

10.8.5 VME_Read

Liest Daten vom VME-Bus.

10.8.6 Check_VME_Error

Überprüft, ob der letzte Transfer über den VME-Bus erfolgreich war.

10.8.7 check_access

Überprüft durch Lesen des Error-Codes, ob der letzte Gerätezugriff erfolgreich war.

10.8.8 Select_Target

Selektiert ein CAEN-HV-Modul für die weitere Kommunikation.

10.8.9 Select_Protect

Selektiert die Protection-Bits eines CAEN-Crates.

10.8.10 write_value

Schickt einen Sollwert an ein HV-Modul.

10.8.11 read_value

Liest einen Wert von einem HV-Modul.

11 Varianten

Keine

12 Besonderheiten

Als Besonderheit ist die Zusammenfassung mehrerer HV-Module zu einem Supergerät anzusehen (siehe Abschnitt 4.10 auf Seite 17).

Index

—Symbole —

Änderungsprotokoll 2

—A—

Abriß 2
Active_EQM 29
An externe Interrupts konnektierte EQMs 29
Aufgabe des Gerätes 7
Ausschalten 15

—B—

Bedienung des Gerätes 15
Bedienungsfehler 17
Besonderheiten 31

—C—

Controller 7
Crate 7

—D—

Datenbasis 25
Dev_Init_EQM 29
Dev_Reset_EQM 29
Display_DevErr_EQM 30
Display_DPR_EQM 30
DRD Interrupt 14
DRQ Interrupt 14
Dualport RAM 25

—E—

Einschalten 15
Emerg_EQM 27
Emergency-Event 17
EQMs 27

- An externe Interrupts konnektierte 29
- Eventkonnektierte 27
 - Emerg_EQM 27
- für die Diagnose vor Ort 30
 - Display_DevErr_EQM 30
 - Display_DPR_EQM 30
- Globale Routinen 30
 - check_access 31

- Check_VME_Error 31
- Do_Intr_Service_Prep 30
- Read_and_Update_Status 30
- read_value 31
- Select_Protect 31
- Select_Target 31
- VME_Read 30
- VME_Select 30
- VME_Write 30
- write_value 31
- Kommandogetriggerte 29
 - Active_EQM 29
 - Dev_Init_EQM 29
 - Dev_Reset_EQM 29
 - Get_Ist_EQM 29
 - Power_EQM 29
 - Set_Soll_EQM 29
 - Status_EQM 29
- Periodisch konnektierte 29
 - Update_Config_EQM 29
- Sonstige 30
 - Startup_EQM 30

Event-Overrun 16
Event-Sequenzfehler 16
Eventkonnektierte EQMs 27
Eventkonnektierungen 15

—F—

Funktionscodes 8

- ifb_high_addr 8
- ifb_IFK_intr_mask 9
- ifb_low_addr 8
- ifb_rd_data_b 9
- ifb_rd_data_w 9
- ifb_rd_Ident 10
- ifb_rd_IFK_Intr 10
- ifb_rd_VME_Intr 9
- ifb_rd_VME_Intr_Mask 9
- ifb_rd_VME_Intr_Vect 10
- ifb_VME_Intr_Mask 8
- ifb_VME_Reset 9
- ifb_wr_data_b 8
- ifb_wr_data_w 8

—G—

Genauigkeitsanforderungen 15

Gerät	
• Aufgabe	7
• Bedienung	15
• Hardware	7
– Controller	7
– Crate	7
– HV-Modul	7
– Modul	7
• Repräsentation	17
• Schnittstelle	7
Gerätemodell	7
• Kennzeichnung	17
• Master-Properties	18
• Slave-Properties	22
Gerätevarianten	7
get_device_constants	27
get_device_parameter	27
Get_Ist_EQM	29
get_super_constants	27
Globale Routinen	27, 30
—H—	
Handbetrieb	16
Hardware des Gerätes	7
Hardwarefehler-Bit	16
Hardwarestatus	14
HV-Modul	7
—I—	
ifb_high_addr	8
ifb_IFK_intr_mask	9
ifb_low_addr	8
ifb_rd_data_b	9
ifb_rd_data_w	9
ifb_rd_Ident	10
ifb_rd_IFK_Intr	10
ifb_rd_VME_Intr	9
ifb_rd_VME_Intr_Mask	9
ifb_rd_VME_Intr_Vect	10
ifb_VME_Intr_Mask	8
ifb_VME_Reset	9
ifb_wr_data_b	8
ifb_wr_data_w	8
Init	16
Interfacekarte	8
Interlock	13, 16
Interne Zustände	27
Interrupt	
• DRD Interrupt	14
• DRQ Interrupt	14
• Interlock	13
ISEL	7
Istwert	15
—K—	
Kaltstarts	16
Kommandogetriggerte EQMs	29
—L—	
Lokale Datenbasis	25
• Tabelle der Konstanten	25
—M—	
Master-Properties	18
Mastergeräte	25
Modul	7
Modul-Typen	13
—N—	
N_Init	26
N_Reset	26
Normalbetrieb	15
Normierung	15
—O—	
Onlinetest	14
Overrun	16
—P—	
Periodisch konnektierte EQMs	29
Power_EQM	29
Properties	
• ACTIV	22
• CONSTANT	20
• COPYSET	23
• CURRENTI	21
• CURRENTS	21
• EQMERROR	22
• INFOSTAT	19
• INIT	18
• Master-	18
• POWER	18
• RAMPRATE	21
• RESET	19
• Slave-	22

• STATUS	18
• TRIPTIME	22
• VERSION	19
• VOLTAGEI	20
• VOLTAGES	20
Protection Bits	11

—R—

R_Active	26
R_Constant	27
R_CurrentI	27
R_CurrentS	27
R_EQMErr	26
R_InfoStat	26
R_Power	26
R_RampRate	27
R_Status	26
R_TripTime	27
R_Version	26
R_VoltageI	27
R_VoltageS	27
Repräsentation des Gerätes	17
Reset	16

—S—

Schnittstelle zum Gerät	7
Sequenzfehler	16
Set_Soll_EQM	29
Slave-Properties	22
Softwareentwurf	25
Softwarestatus	14
Sollwert	15
Sonstige EQMs	30
Störungen	16
• Emergency-Event	17
• Event-Overrun	16
• Event-Sequenzfehler	16
• Interlock	16
• Kommunikation EC – Gerät	17
Startup_EQM	30
Startwerte	16
Status_EQM	29
Statusbits	14
Supergeräte	7, 17, 25
Supergerätkomponenten	17

—T—

Timing	15
--------------	----

TRIP	7
------------	---

—U—

Update_Config_EQM	29
USRs	26
• gerätespezifische	26
– R_Constant	27
– R_CurrentI	27
– R_CurrentS	27
– R_RampRate	27
– R_TripTime	27
– R_VoltageI	27
– R_VoltageS	27
– W_CurrentS	27
– W_RampRate	27
– W_TripTime	27
– W_VoltageS	27
• Globale Routinen	27
– get_device_constants	27
– get_device_parameter	27
– get_super_constants	27
• obligatorische	26
– N_Init	26
– N_Reset	26
– R_Active	26
– R_EQMErr	26
– R_InfoStat	26
– R_Power	26
– R_Status	26
– R_Version	26
– W_Active	26
– W_CopySet	26
– W_Power	26

—V—

Varianten	31
• Betriebs-	15
• Geräte-	7
• Software-	31
VME-Adressen	10
VME-Interface	10
VSEL	7

—W—

W_Active	26
W_CopySet	26
W_CurrentS	27
W_Power	26

W_RampRate.....	27
W_TripTime.....	27
W_voltageS.....	27

—Z—

Zeitkritische Anforderungen.....	15
Zustände	
• Interne.....	27
– Übergänge.....	27
– Bedeutung.....	27
– Standard-Übergänge.....	27