



# CEHV - Hochspannungschalter Elektronenkühler

Gerätemodell und Softwareentwurf

U. Krause

*In diesem Papier sollen die allerwichtigsten Informationen über die Gerätesoftware des Hochspannungsschalters der Elektronekühler dargelegt werden.*

<b>Änderungsprotokoll</b>			
Datum	GM-Version	Name	Kommentar
3. Jun. 98	CEHV_02	UK	1. Version
10. März 04	CEHV_02	UK	Format angepasst

# Inhaltsverzeichnis

<b>I</b>	<b>Das Gerätemodell</b>	<b>7</b>
<b>1</b>	<b>Die Aufgabe des Gerätes</b>	<b>7</b>
<b>2</b>	<b>Die Hardware des Gerätes</b>	<b>7</b>
2.1	Gerätebeschreibung . . . . .	7
2.2	Gerätevarianten . . . . .	7
<b>3</b>	<b>Die Schnittstelle zum Gerät</b>	<b>8</b>
3.1	Funktionscodes der Interfacekarte . . . . .	8
3.2	Interlock Interrupt . . . . .	8
3.3	Data Request (DRQ) Interrupts . . . . .	8
3.4	Data Ready (DRD) Interrupts . . . . .	8
3.5	Definition der Bits des Hardwarestatus . . . . .	8
<b>4</b>	<b>Die Bedienung des Gerätes</b>	<b>9</b>
4.1	Aufgaben im Normalbetrieb . . . . .	9
4.1.1	Betriebsarten . . . . .	9
4.1.2	Soll- und Istwert . . . . .	10
4.1.3	Einschalten . . . . .	10
4.2	Zeitkritische Anforderungen . . . . .	10
4.3	Einordnung in das Timing . . . . .	10
4.4	Festlegung von Startwerten . . . . .	10
4.4.1	Kaltstarts . . . . .	10
4.4.2	Warmstarts . . . . .	10
4.5	Handbetrieb . . . . .	11
4.6	Ableitung des Hardwarefehler-Bits aus dem Gerätestatus . . . . .	11
4.7	Verhalten bei Störungen . . . . .	11
4.7.1	Geräteinterlock . . . . .	11
4.7.2	Event-Sequenzfehler . . . . .	11
4.7.3	Event-Overrun . . . . .	11
4.7.4	Zyklus-Ende . . . . .	11
4.7.5	Emergency-Event . . . . .	11
4.7.6	Ausfall der Kommunikation EC – Gerät . . . . .	11
4.7.7	Sonstiges . . . . .	11
4.8	Bedienungsfehler vom Operating . . . . .	12
4.9	Sonstige Anforderungen . . . . .	12
<b>5</b>	<b>Die Repräsentation des Gerätes</b>	<b>12</b>
5.1	Kennzeichnung des Gerätemodells . . . . .	12
5.2	Die Master-Properties . . . . .	12
5.2.1	POWER . . . . .	12
5.2.2	STATUS . . . . .	12
5.2.3	INIT . . . . .	13
5.2.4	RESET . . . . .	13
5.2.5	VERSION . . . . .	13
5.2.6	INFOSTAT . . . . .	13
5.2.7	DCMODE . . . . .	14
5.2.8	HVDCS . . . . .	14
5.2.9	HVDCI . . . . .	14

5.2.10	EXTDC . . . . .	15
5.3	Die Slave-Properties . . . . .	15
5.3.1	ACTIV . . . . .	15
5.3.2	EQMERROR . . . . .	15
5.3.3	COPYSET . . . . .	16
5.3.4	HVPULSE . . . . .	16
<b>II</b>	<b>Der Entwurf der Software</b>	<b>17</b>
<b>6</b>	<b>Softwareentwurf</b>	<b>17</b>
<b>7</b>	<b>Lokale Datenbasis</b>	<b>17</b>
<b>8</b>	<b>Dualport RAM</b>	<b>17</b>
<b>9</b>	<b>USRs - User Service Routinen</b>	<b>17</b>
9.1	Obligatorische USRs . . . . .	17
9.1.1	N_Init . . . . .	17
9.1.2	N_Reset . . . . .	17
9.1.3	R_Status . . . . .	17
9.1.4	R_Power . . . . .	17
9.1.5	W_Power . . . . .	17
9.1.6	R_Active . . . . .	17
9.1.7	W_Active . . . . .	17
9.1.8	W_CopySet . . . . .	17
9.1.9	R_EQMerr . . . . .	17
9.1.10	R_Version . . . . .	17
9.1.11	R_InfoStat . . . . .	17
9.2	Gerätespezifische USRs . . . . .	17
9.3	Globale Routinen . . . . .	17
<b>10</b>	<b>EQMs - Equipment Module</b>	<b>18</b>
10.1	Interne Zustände . . . . .	18
10.1.1	Bedeutung der internen Zustände . . . . .	18
10.1.2	Übergänge zwischen den Zuständen . . . . .	18
10.1.3	Standard-Zustandsübergänge . . . . .	18
10.2	Eventkonnektierte EQMs . . . . .	20
10.2.1	CleanUp_EQM . . . . .	20
10.2.2	Emerg_EQM . . . . .	20
10.3	Periodisch konnektierte EQMs . . . . .	20
10.3.1	Update_Config_EQM . . . . .	20
10.4	An externe Interrupts konnektierte EQMs . . . . .	20
10.4.1	DRD_EQM . . . . .	20
10.4.2	DRQ_EQM . . . . .	20
10.5	Kommandogetriggerte EQMs . . . . .	21
10.5.1	Dev_Init_EQM . . . . .	21
10.5.2	Dev_Reset_EQM . . . . .	21
10.5.3	Status_EQM . . . . .	21
10.5.4	Active_EQM . . . . .	21
10.5.5	Power_EQM . . . . .	21
10.6	EQMs für die Diagnose vor Ort . . . . .	21
10.6.1	Display_DPR_EQM . . . . .	21

10.6.2	Display_DevErr_EQM . . . . .	21
10.7	Sonstige EQMs . . . . .	21
10.7.1	Startup_EQM . . . . .	21
10.8	Globale Routinen . . . . .	21
10.8.1	Read_and_Update_Status . . . . .	22
10.8.2	Set_InternalState . . . . .	22
10.8.3	Do_Intr_Service_Prep . . . . .	22
10.8.4	Power_on . . . . .	22
10.8.5	Power_off . . . . .	22
10.8.6	Set_Active_State . . . . .	22
10.9	MIL-Treiber . . . . .	22
<b>11</b>	<b>Varianten</b>	<b>22</b>
	<b>Index</b>	<b>23</b>



## Teil I

# Das Gerätemodell

## 1 Die Aufgabe des Gerätes

Der HV-Schalter (Hochspannungs-Schalter) ist der Knopf, über den der Elektronenstrahl des Elektronenkühlers schnell an- und abgeschaltet werden kann.

Zur Kühlung muss der Elektronenstrahl natürlich eingeschaltet sein. Ausserhalb des eigentlichen Kühlprozesses sollte der Elektronenstrahl dagegen aus sein. Das ist einerseits sinnvoll, um den umlaufenden Ionenstrahl nicht zu stören. Andererseits können die einzelnen Komponenten des Elektronenkühlers bei eingeschaltetem Elektronenstrahl nur streng synchronisiert verfahren werden. Bereits kleine Abweichungen der Einstellungen der Kühlerkomponenten untereinander können dazu führen, dass der Elektronenstrahl auf dem Weg zum Auffänger verloren geht. Das muss aber unbedingt vermieden werden, denn der Elektronenstrahl ist so intensiv, dass er an der Auftreffstelle Schäden hervorrufen kann, und nicht zuletzt würde dabei das Netzgerät zur Erzeugung des Elektronenstrahls überlastet.

Bei abgeschaltetem Elektronenstrahl dagegen können die Kühlerkomponenten unsynchronisiert beliebig verfahren werden.

Die Bedienung des HV-Schalters soll sowohl zyklussynchron, also gesteuert durch Events, als auch interaktiv, also per Operating-Kommando, erfolgen können.

## 2 Die Hardware des Gerätes

### 2.1 Gerätebeschreibung

Der eigentliche HV-Schalter befindet sich auf Hochspannungspotential. Zur Bedienung wird auf Erdpotential ein (TTL-?) Schaltsignal erzeugt, das über eine Lichtleiterstrecke an den eigentlichen Schalter übertragen wird.

Wenn im folgenden vom HV-Schalter gesprochen wird, ist in der Regel nur die erdseitige Einheit zur Erzeugung des Schaltsignals gemeint.

Das Schaltsignal zur Ansteuerung des HV-Schalters kann auf drei Arten erzeugt werden:

**Rechner:** Über einen Rechneranschluss. Per MIL-Kommando kann das Schaltsignal ein- und ausgeschaltet werden.

**Extern:** Das Schaltsignal kann auch direkt über eine Buchse extern eingespeist werden. Damit ist es möglich, den HV-Schalter per Hardware-Signal zu bedienen.

Um Konflikte über die Bedienhoheit zu vermeiden, ist dieser externe Steuereingang nur wirksam, wenn er zuvor per MIL-Kommando explizit freigegeben worden ist. Das Kontrollsystem kann die Hoheit jederzeit wieder zurückholen, indem per MIL-Kommando der externe Eingang gesperrt wird.

**Lokal:** Über einen Schalter am Steuergerät können Steuerbefehle des Kontrollsystems (und gleichzeitig auch der externe Steuereingang) gesperrt werden.

Die Bedienung erfolgt dann allein über einen weiteren Schalter vor Ort am Steuergerät.

### 2.2 Gerätevarianten

Es gibt derzeit zwei Geräte (eines im SIS-Kühler, eines im ESR-Kühler). Beide sind identisch aufgebaut. Sie unterscheiden sich allerdings etwas in der Ansteuerung (siehe Abschn. 4.7.4 auf Seite 11).

## 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 (nur IFK)
<code>ifb_extern_on</code>	04	Funktion	externer Steuereingang gesperrt
<code>ifb_extern_off</code>	05	Funktion	externer Steuereingang frei
<code>ifb_hv_on</code>	14	Funktion	Schalter „Ein“
<code>ifb_hv_off</code>	19	Funktion	Schalter „Aus“
<code>ifb_rdstat</code>	C0	Lesen	Gerätstatus, 1. Byte lesen
<code>ifb_rdstat_int</code>	C9	Lesen	Status der Interfacekarte lesen

#### `ifb_reset`

Es erfolgt nur ein Reset der Interfacekarte.

### 3.2 Interlock Interrupt

Nicht verwendet.

### 3.3 Data Request (DRQ) Interrupts

DRQ-Interrupts werden nicht erwartet bzw. verarbeitet.

### 3.4 Data Ready (DRD) Interrupts

DRD-Interrupts werden nicht erwartet bzw. verarbeitet.

### 3.5 Definition der Bits des Hardwarestatus

Das Gerät liefert 1 Byte relevante Statusinformation. Die Funktionscodes zum Lesen der entsprechenden Bytes sind in der folgenden Tabelle zusammengefasst.

Statusbits	Funktionscode
8 ... 15	<code>ifb_rdstat</code>

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

*Forts. auf nächster Seite*



Forts. von letzter Seite

Bit	Name	Bedeutung	
		High (1)	Low (0)
2	reserved		
3	reserved		
4	Emergency	no	yes
5	Interlock	no	yes
6	HW Error	no	yes
7	SW Error	no	yes
8	nicht belegt	1	–
9	nicht belegt	1	–
10	nicht belegt	1	–
11	nicht belegt	1	–
12	Schalterstellung	Ein	Aus
13	???	?	?
14	externer Eingang	frei	gesp.
15	Remote / local	Remote	Local
16	nicht belegt	1	–
17	nicht belegt	1	–
18	nicht belegt	1	–
19	nicht belegt	1	–
20	nicht belegt	1	–
21	nicht belegt	1	–
22	nicht belegt	1	–
23	nicht belegt	1	–
23	nicht belegt	1	–
24	nicht belegt	1	–
25	nicht belegt	1	–
26	nicht belegt	1	–
27	nicht belegt	1	–
28	nicht belegt	1	–
29	nicht belegt	1	–
30	nicht belegt	1	–
31	nicht belegt	1	–

## 4 Die Bedienung 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 Betriebsarten

Das Gerät kann auf zwei Weisen bedient werden:

**Pulsmode:** Das Gerät wird eventgesteuert betrieben.

Über Slave-Sollwerte ist einzustellen, ob der HV-Schalter in dem jeweiligen virtuellen Beschleuniger ein- oder ausgeschaltet sein soll.

Der externe ESteuereingang ist im Pulsmode *immer* gesperrt.

**DC-Mode:** Das Gerät kann per Kontrollsystem-Kommando interaktiv bedient werden.

Über Master-Sollwerte kann der HV-Schalter ein- oder ausgeschaltet werden und der externe Eingang freigegeben oder gesperrt werden.

Zwischen beiden Betriebsarten kann per Kontrollsystem-Kommando umgeschaltet werden.

#### 4.1.2 Soll- und Istwert

#### 4.1.3 Einschalten

Keine Einschaltung des Gerätes (Netzversorgung) über Rechner vorgesehen.

### 4.2 Zeitkritische Anforderungen

#### 4.3 Einordnung in das Timing

Wird das Gerät per Eventsteuerung betrieben, sind zwei Events (neben dem Standard-Event EVT\_End\_Cycle) von Bedeutung:

**EVT\_EBeam\_On (160):** Je nachdem, ob der HV-Schalter im laufenden virtuellen Beschleuniger ein- oder ausgeschaltet sein soll, wird das Steuersignal ein (HV-Schalter ein, Einschalten des Elektronenstrahls) oder ausgeschaltet (HV-Schalter aus, Ausschalten des Elektronenstrahls).

**EVT\_EBeam\_Off (161):** Steuersignal aus (HV-Schalter aus, Ausschalten des Elektronenstrahls) unabhängig vom Sollwert des Schalters.

**EVT\_End\_Cycle (055):** *Nur in der Variante für den SIS-Kühler!* Aufräumen am Ende des Zyklus. Sperren des externen Eingangs und Ausschalten des Steuersignals (Schalter / Elektronenstrahl aus). In der Variante für den ESR-Kühler bleibt der Zustand des Schalters bei diesem Event unverändert.

### 4.4 Festlegung von Startwerten

#### 4.4.1 Kaltstarts

Bei einem Kaltstart werden folgende Aktionen durchgeführt:

- Es wird ein Gerätereset durchgeführt.
- Das Gerät wird in den Pulsbetrieb geschaltet (Die SE wird in den Eventmode-Betrieb geschaltet).
- Die Sollwerte werden für alle virtuellen Beschleuniger wie folgt gesetzt und am Gerät eingestellt:
  - Externer Eingang gesperrt.
  - Schalter aus.
- 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.

#### 4.4.2 Warmstarts

Bei einem Warmstart werden folgende Aktionen durchgeführt:

- Es wird ein Gerätereset durchgeführt.
- Wenn sich das Gerät im DC-Modus (interaktive Bedienung) befindet, werden die jeweiligen Sollwerte am Gerät eingestellt.

## 4.5 Handbetrieb

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

In diesem Zustand kann die Rechnersteuerung das Gerät nicht mehr verändern. Die aktuellen Werte auszulesen ist aber weiterhin möglich.

## 4.6 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.

— nicht benutzt —

## 4.7 Verhalten bei Störungen

### 4.7.1 Geräteinterlock

— nicht von Bedeutung —

### 4.7.2 Event-Sequenzfehler

Im Pulsbetrieb reagiert das Gerät immer auf das Event EVT\_EBeam\_Off, unabhängig davon, ob im laufenden virtuellen Beschleuniger zuvor das Event EVT\_EBeam\_On gekommen ist oder nicht. Beim Event EVT\_EBeam\_Off wird immer das Steuersignal zurückgenommen (HV-Schalter und damit Elektronenstrahl aus).

### 4.7.3 Event-Overrun

Treten beim Event EVT\_EBeam\_Off Overrun-Fehler auf, ist der Zyklus abzuberechnen.

Overrun-Fehler beim Event EVT\_EBeam\_Off werden gemeldet. Auch bei Overrun-Fehlern wird das Steuersignal ausgeschaltet (HV-Schalter und damit Elektronenstrahl aus).

### 4.7.4 Zyklus-Ende

In der Variante für den Schalter im SIS-Kühler wird beim Event EVT\_End\_Cycle immer das Steuersignal zurückgenommen (HV-Schalter und damit Elektronenstrahl aus) sowie der externe Eingang gesperrt.

In der Variante für den Schalter im ESR-Kühler bleibt der Zustand des Schalters beim Event EVT\_End\_Cycle unverändert.

### 4.7.5 Emergency-Event

Beim Auftreten eines Emergency-Events ist der Elektronenstrahl abzuschalten.

### 4.7.6 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.

### 4.7.7 Sonstiges

Keine weiteren Störungen vorhanden.

## 4.8 Bedienungsfehler vom Operating

Sollwerte ausserhalb des zulässigen Bereichs sind zurückzuweisen.

## 4.9 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 **CEHV**.

Die Gerätemodellnummer ist  $47_{dez}$ .

## 5.2 Die 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
DCMODE	R/W	0	–	1	BitSet16	1	0
HVDCS	R/W	0	–	1	BitSet16	1	0
HVDCI	R/W	0	–	1	BitSet16	1	0
EXTDC	R/W	0	–	1	BitSet16	1	0

### 5.2.1 POWER

**Bedeutung:** Gibt an, ob der Leistungsteil des Gerätes ein- oder ausgeschaltet ist bzw. werden soll.

Diese Property ist nur aus Kompatibilitätsgründen vorhanden. Die Netzversorgung des Gerätes kann nicht geschaltet 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.

### 5.2.2 STATUS

**Bedeutung:** Auslesen des 32bit Gerätestatus.

**Parameter:** Keine.

**Daten:** Das 32bit Statuswort. Die Bits entsprechen den Statusbits, wie sie in Abschnitt 3.5 auf Seite 8 und in der Tabelle 1 auf Seite 8 erklärt sind.

### 5.2.3 INIT

**Bedeutung:** Initialisierung des Gerätes (Kaltstart). Für die dabei durchzuführenden Aktionen siehe Abschnitt 4.4.1 auf Seite 10.

**Parameter:** Keine.

**Daten:** Keine.

### 5.2.4 RESET

**Bedeutung:** Reset des Gerätes (Warmstart). Für die dabei durchzuführenden Aktionen siehe Abschnitt 4.4.2 auf Seite 10.

**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.

**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 DCMODE

**Bedeutung:** Gibt an, ob das Gerät interaktiv per Operating-Kommando bedient wird oder ob es per Event gesteuert wird und damit an der Puls-zu-Puls Umtastung teilnimmt. Bei der interaktiven Bedienung (DC-Modus) wird der Hochspannungsschalter über die Property HVDCS (und HVDCI) bedient. Läuft das Gerät mit Puls-zu-Puls Umtastung, gelten die Sollwerte der Property HVPULSE (spezifisch für die jeweiligen virtuellen Beschleuniger).  
Bei der Umschaltung der Betriebsmoden wird der Hochspannungsschalter abgeschaltet (entspricht HVDCS = 0 im DC-Modus), um definierte Ausgangsbedingungen zu gewährleisten.

**Parameter:** Keine.

**Daten:** Das Datum kann nur zwei Werte annehmen:

- 0:** Gerät wird durch Events gesteuert und nimmt an der Puls-zu-Puls Modulation teil.
- 1:** Gerät reagiert nicht auf Events. Die Bedienung erfolgt über interaktive Properties.

### 5.2.8 HVDCS

**Bedeutung:** Nur wirksam im DC-Mode (DCMODE=1). Über diese Property wird der HV-Schalter per Operating-Kommando bedient (Sollwert der Schalterstellung).

**Parameter:** Keine.

**Daten:** Das Datum kann nur zwei Werte annehmen:

- 0:** HV-Schalter aus.
- 1:** HV-Schalter ein.

### 5.2.9 HVDCI

**Bedeutung:** Über diese Property kann die aktuelle Stellung des HV-Schalters gelesen werden (Ist-Wert des HV-Schalters). Nur sinnvoll im DC-Mode (DCMODE=1), im Puls-Mode wird die Schalterstellung nur im nächsten Gap gelesen.

Die Property liefert auch den jeweils aktuellen Wert, wenn der Schalter im Handbetrieb gefahren wird oder wenn er über den externen Eingang gesteuert wird.

**Parameter:** Keine.

**Daten:** Das Datum kann nur zwei Werte annehmen:

- 0:** HV-Schalter aus.
- 1:** HV-Schalter ein.

### 5.2.10 EXTDC

**Bedeutung:** Nur wirksam im DC-Mode (DCMODE=1). Über diese Property wird der externe Steuereingang per Operating-Kommando freigegeben oder gesperrt.

**Parameter:** Keine.

**Daten:** Das Datum kann nur zwei Werte annehmen:

**0:** Externer Eingang gesperrt. Der HV-Schalter wird über die Property HVDCS gesteuert.

**1:** Externer Eingang freigegeben. Der HV-Schalter wird über das Signal an dem externen Eingang gesteuert. Die Property HVDCS ist dann nicht wirksam.

## 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
HVPULSE	R/W	0	–	1	BitSet16	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 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.

**Parameter:** Keine.

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

### 5.3.4 HVPULSE

**Bedeutung:** Nur wirksam, wenn der HV-Schalter im Pulsbetrieb gefahren wird (DCMODE=0). Sollwert des HV-Schalters für den jeweiligen virtuellen Beschleuniger.

**Parameter:** Keine.

**Daten:** Das Datum kann nur zwei Werte annehmen:

**0:** HV-Schalter aus.

**1:** HV-Schalter ein.



# 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

### 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 des HV-Schalters

#### 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.
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 5 und Tabelle 6) aufgeteilt. Die Legende zu diesen Tabellen ist in Tabelle 7 zu finden.

#### 10.1.3 Standard-Zustandsübergänge

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

Das Gerät nimmt an der Puls-zu-Puls-Modulation teil. 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 -> power_on`  
`power_on -> power_seq -> power_off`

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	Inverter=i
	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 5: Zustandsübergangsdiagramm 1

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	CurrentLEQM	-

Tabelle 6: 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 7: Legende zu den Zustandsübergangsdiagrammen

## 10.2 Eventkonnektierte EQMs

### 10.2.1 CleanUp\_EQM

**Event:** Evt\_End\_Cycle.

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

### 10.2.2 Emerg\_EQM

**Event:** Evt\_Emergency.

**Aktion:** Internen Zustand auf 'Emergency' setzen. HV-Schalter ausschalten, externen Eingang sperren.

## 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 Status zu lesen. Erfolgt eine Reaktion, wird das Gerät als 'online' geführt.

## 10.4 An externe Interrupts konnektierte EQMs

### 10.4.1 DRD\_EQM

**Interrupt:** Data Ready Interrupt.

**Aktion:** Keine. Sollte bei CEHV nicht vorkommen.

### 10.4.2 DRQ\_EQM

**Interrupt:** Data Request Interrupt.

**Aktion:** Keine. Sollte bei CEHV 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

## 10.6 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 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 4.3 auf Seite 10) 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 zwei Varianten der Software erforderlich. Die Varianten im Einzelnen:

**Variante 1** (SIS) Im Pulsbetrieb (Pulsmode) wird in virtuellen Beschleunigern, in denen das Gerät aktiv geschaltet ist, am Ende des Zyklus (Ende des virtuellen Beschleunigers) der Hochspannungsschalter immer geöffnet (Hochspannung ausgeschaltet).

Das Gerät ist als Therapie-Gerät konfiguriert.

**Variante 2** (ESR) Im Pulsbetrieb (Pulsmode) wird das Gerät nur durch die Events `EVT_EBeam_On` und `EVT_EBeam_Off` gesteuert. Der aktuelle Schaltzustand wird am Ende eines virtuellen Beschleunigers nicht geändert (ist die Hochspannung eingeschaltet, bleibt sie eingeschaltet).

Das Gerät ist nicht als Therapie-Gerät konfiguriert.

# Index

—A—	
Active_EQM .....	21
An Interrupts konnektierte EQMs .....	20
Aufgabe des Gerätes .....	7
—B—	
Bedienung des Gerätes .....	9
Bedienungsfehler .....	12
Betriebsarten .....	9
—C—	
CleanUp_EQM .....	20
—D—	
Datenbasis .....	17
DC-Mode .....	9
Dev_Init_EQM .....	21
Dev_Reset_EQM .....	21
Display_DevErr_EQM .....	21
Display_DPR_EQM .....	21
DRD Interrupt .....	8
DRD_EQM .....	20
DRQ Interrupt .....	8
DRQ_EQM .....	20
Dualport RAM .....	17
—E—	
Einschalten .....	10
Emerg_EQM .....	20
Emergency-Event .....	11
EQMs .....	18
EQMs; An Interrupts konnektierte .....	20
EQMs; An Interrupts konnektierte; DRD_EQM	20
EQMs; An Interrupts konnektierte; DRQ_EQM	20
EQMs; Eventkonnektierte .....	20
EQMs; Eventkonnektierte; CleanUp_EQM ...	20
EQMs; Eventkonnektierte; Emerg_EQM .....	20
EQMs; für die Diagnose vor Ort .....	21
EQMs; für die Diagnose vor Ort; Display_DevErr_EQM	21
EQMs; für die Diagnose vor Ort; Display_DPR_EQM	21
—F—	
Funktionscodes .....	8
Funktionscodes; ifb_reset .....	8
—G—	
Gerät; Aufgabe .....	7
Gerät; Bedienung .....	9
Gerät; Hardware .....	7
Gerät; Repräsentation .....	12
Gerät; Schnittstelle .....	8
Gerätemodell .....	7
• Slave-Properties .....	15
Gerätemodell; Kennzeichnung .....	12
Gerätemodell; Master-Properties .....	12
Gerätevarianten .....	7
Globale Routinen .....	17, 21
EQMs; Globale Routinen .....	21
EQMs; Globale Routinen; Do_Intr_Service_Prep	22
EQMs; Globale Routinen; Power_off .....	22
EQMs; Globale Routinen; Power_on .....	22
EQMs; Globale Routinen; Read_and_Update_Status	22
EQMs; Globale Routinen; Set_Active_State ..	22
EQMs; Globale Routinen; Set_InternalState ..	22
EQMs; Kommandogetriggerte .....	21
EQMs; Kommandogetriggerte; Active_EQM ..	21
EQMs; Kommandogetriggerte; Dev_Init_EQM	21
EQMs; Kommandogetriggerte; Dev_Reset_EQM	21
EQMs; Kommandogetriggerte; Power_EQM ..	21
EQMs; Kommandogetriggerte; Status_EQM ..	21
EQMs; MIL-Treiber .....	22
EQMs; Periodisch konnektierte .....	20
EQMs; Periodisch konnektierte; Update_Config_EQM	20
EQMs; Sonstige .....	21
EQMs; Sonstige; Startup_EQM .....	21
EQMs; Varianten .....	22
Event-Overrun .....	11
Event-Sequenzfehler .....	11
Eventkonnektierte EQMs .....	20
Eventkonnektierungen .....	10

—H—

Handbetrieb .....	11
Hardware des Gerätes .....	7
Hardwarefehler-Bit .....	11
Hardwarestatus .....	8

—I—

ifb_reset .....	8
Init .....	10
Interfacekarte .....	8
Interfacekarte;MIL-Treiber .....	22
Interlock .....	8, 11
Interne Zustände .....	18
Interrupt;DRD Interrupt .....	8
Interrupt;DRQ Interrupt .....	8
Interrupt;Interlock .....	8
Istwert .....	10

—K—

Kaltstarts .....	10
Kommandogetriggerte EQMs .....	21

—L—

Lokale Datenbasis .....	17
-------------------------	----

—M—

Master-Properties .....	12
MIL-Treiber .....	22

—N—

N_Init .....	17
N_Reset .....	17
Normalbetrieb .....	9

—O—

Overrun .....	11
---------------	----

—P—

Periodisch konnektierte EQMs .....	20
Power_EQM .....	21
Properties	
• EQMERROR .....	15
• Slave- .....	15

Properties;ACTIV .....	15
Properties;COPYSET .....	16
Properties;DCMODE .....	14
Properties;EXTDC .....	15
Properties;HVDCI .....	14
Properties;HVDCS .....	14
Properties;HVPULSE .....	16
Properties;INFOSTAT .....	13
Properties;INIT .....	13
Properties;Master- .....	12
Properties;POWER .....	12
Properties;RESET .....	13
Properties;STATUS .....	12
Properties;VERSION .....	13
Pulsmode .....	9

—R—

R_Active .....	17
R_EQMErr .....	17
R_InfoStat .....	17
R_Power .....	17
R_Status .....	17
R_Version .....	17
Repräsentation des Gerätes .....	12
Reset .....	10

—S—

Schnittstelle zum Gerät .....	8
Sequenzfehler .....	11
Slave-Properties .....	15
Softwareentwurf .....	17
Softwarestatus .....	8
Sollwert .....	10
Sonstige EQMs .....	21
Störungen .....	11
Störungen;Emergency-Event .....	11
Störungen;Event-Overrun .....	11
Störungen;Event-Sequenzfehler .....	11
Störungen;Interlock .....	11
Störungen;Kommunikation EC – Gerät .....	11
Störungen;Zyklus-Ende .....	11
Startup_EQM .....	21
Startwerte .....	10
Status_EQM .....	21
Statusbits .....	8

—T—

Timing .....	10
--------------	----



—U—

Update_Config_EQM.....	20
USRs .....	17
USRs <sub>;</sub> gerätespezifische .....	17
USRs <sub>;</sub> Globale Routinen .....	17
USRs <sub>;</sub> obligatorische .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> N_Init .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> N_Reset .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> R_Active .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> R_EQMErr .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> R_InfoStat .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> R_Power.....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> R_Status .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> R_Version .....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> W_Active.....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> W_CopySet.....	17
USRs <sub>;</sub> obligatorische <sub>;</sub> W_Power .....	17

—V—

Varianten.....	22
Varianten <sub>;</sub> Betriebs-.....	10
Varianten <sub>;</sub> EQM-.....	22
Varianten <sub>;</sub> Geräte-.....	7

—W—

W_Active.....	17
W_CopySet .....	17
W_Power .....	17
Warmstarts.....	10

—Z—

Zeitkritische Anforderungen.....	10
Zustände <sub>;</sub> Interne .....	18
Zustände <sub>;</sub> Interne <sub>;</sub> Übergänge .....	18
Zustände <sub>;</sub> Interne <sub>;</sub> Bedeutung .....	18
Zustände <sub>;</sub> Interne <sub>;</sub> Standard-Übergänge.....	18
Zyklus-Ende .....	11