



SCU exklusiver Registersatz Device Interface

Derek Schupp

Version vom: Donnerstag, 29. Februar 2024, 13:30:00

Aktuelle Versionen immer unter:

<https://wiki.gsi.de/EPS/ACUDocumentation>

Inhaltsverzeichnis

1. Änderungsliste	1
2. Abkürzungen	2
3. Überblick	3
4. Register (FSPs) innerhalb eines ACU Systems	4
5. Das Device Interface	4
6. Registerstruktur der SCU exklusiven Register im Device Interface	4
6.1. <i>ACU nach SCU Device Interface</i>	5
6.1.1. Quad_Funktionsgeneratoren (0300 _h , 033F _h , 0340 _h , 034F _h).....	7
6.1.2. Status_1 (1000 _h)	7
6.1.3. Status_2 (1001 _h)	8
6.1.4. Command_1 (1020 _h).....	9
6.1.5. Command_2 (1021 _h).....	9
6.1.6. Linearer FunctionGenerator_... (1030 _h , 104F _h , 1050 _h , 106F _h)	9
6.1.7. CurrentValue_1_... (1070 _h , 1071 _h)	10
6.1.8. CurrentValue_2_... (1080 _h , 1081 _h).....	11
6.1.9. SetValue_... (1090 _h ...110F _h)	11
6.1.10. CurrentValue_... (1110 _h , 112F _h).....	11
6.1.11. Interlocks_... (1200 _h ...123F _h).....	11
6.1.12. Errors_... (1240 _h ...125F _h).....	14
6.1.13. Warnings_... (1260 _h , 127F _h)	14
7. Register Kompatibilitäts-Block	15

1. Änderungsliste

<i>Datum</i>	<i>Name</i>	<i>Kommentar</i>
15.10.2020	D. Schupp	Dokument erstellt
26.04.2023	D. Schupp	Wiki-Link aktualisiert
25.10.2023	D. Schupp	Quadratische FG eingeführt, Kompatibilitätsregister eingefügt
29.02.2024	D. Schupp	Registeradressen bei Detailbeschreibung hinzugefügt

2. Abkürzungen

Um den Text übersichtlich und kurz zu gestalten werden bestimmte Begriffe abgekürzt:

ACU	<i>Adaptive Control Unit</i> - alle zur Steuer- und Kontrolleinheit eines Netzgerätes gehörenden Komponenten
ADC	Analog D igital C onverter
DCCT	DC Current Transformer
FPGA	<i>Field Programmable Gate Array</i> - ein programmierbarer Logikbaustein der die Funktionalitäten eines ACU Moduls beinhaltet
ICM	Interlock-and C ontrol M odule
MFU	<i>Multi Function Unit</i> - zentrale Grundeinheit eines ACU-Systems
NIOSII	Mikroprozessor in der MFU der die Benutzerschnittstelle zur Verfügung stellt
FW	<i>Firmware</i> - die notwendige Programmierung des FPGA. Es wird unterschieden zwischen Application und Factory Image. Das Application Image ist i.d.R. durch den Endanwender ersetzbar. Das Factory Image hingegen nur durch das Support-Team.
SCU	<i>Scalable Control Unit</i>
SW	<i>Software</i> - Betriebssoftware der NIOSII/VNC2 Mikroprozessoren innerhalb der MFU. Auch hier wird unterschieden zwischen Application und Factory Image. Das Application Image ist i.d.R. durch den Endanwender ersetzbar. Das Factory Image hingegen nur durch das Support-Team.
SVE	<i>Strom-Versorgungs-Einheit</i> (Netzgerät)
USI	<i>Universal Serial Interface</i>
PCA	<i>Power Config Advanced</i> (Software)
PPS	<i>Personal Protection System</i>
Quench	Sudden transition from superconducting to normal conditions

3. Überblick

Über das SCU/Nios-Interface ist es der SCU prinzipiell möglich jedes FSP (Register) innerhalb eines ACU-Systems direkt zu erreichen.

Dazu wird das USI-Protokoll verwendet.

Die Interfacekarte hingegen unterstützt diese Funktionalität nicht.

Im Weiteren gibt es eine Reihe SCU exklusiver Register, die ausschließlich dem Zugriff mittels SCU bzw. der Interfacekarte ab FW 1.8 vorbehalten sind.

Diese Register sind teilweise im Device Interface der MFU abgelegt, teilweise direkt adressierbar.

4. Register (FSPs) innerhalb eines ACU Systems

Die Beschreibung der FSPs findet sich in den modulbeschreibenden Dokumenten und können dem EPS ACU Wiki entnommen werden.

Für den Zugriff auf diese ist es nötig das USI Protokoll zu nutzen. Über das SCU/Nios-Interface ist die SCU in der Lage dies zu tun, die Interfacekarte hingegen nicht.

5. Das Device Interface

Über das Device Interface werden unterschiedlichste, in der MFU gesammelten Daten, exklusiv der SCU (oder der Interfacekarte ab FW 1.8) zur Verfügung gestellt, ohne dass diese das USI Protokoll für den direkten Zugriff auf FSPs innerhalb des ACU Systems nutzen müssen.

6. Registerstruktur der SCU exklusiven Register im Device Interface

Der Registersatz ist in Gruppen aufgeteilt, die jeweils einen bestimmten Adressbereich belegen. Alle Register haben eine Breite von 16 Bit.

Gelb hinterlegte Register sind nicht Bestandteil des Device Interfaces.

Die blau hinterlegten Register befinden sich im Device Interface.

Der Adressbereich der linearen Funktionsgeneratoren liegt/lag zwar innerhalb des Adressbereichs des Device Interfaces ist/war aber nicht Bestandteil des Device Interfaces.

Adresse(n)	Gruppe	Beschreibung
0300h...033Fh	Quadratischer Funktionsgenerator 1	Die Registerbeschreibung des quadratischen Funktionsgenerator entnehmen Sie der zugehörigen Dokumentation.
0340h...037Fh	Quadratischer Funktionsgenerator 2	
0x08xxh	Register Kompatibilitäts-Block	Diese Register sind veraltet und nur aus Kompatibilitätsgründen zur Interfacekarte noch unterstützt.
1000h...102Fh 1070h...127Fh	ACU nach SCUB Device Interface	Register zur Gerätesteuerung und Betriebszustandsüberwachung angeschlossener Geräte.
1030h...104Fh	Linearer Funktionsgenerator 1	bis einschließlich MFU FW 7.3.x Die Registerbeschreibung des linearen Funktionsgenerator entnehmen Sie der zugehörigen Dokumentation.
1050h...106Fh	Linearer Funktionsgenerator 1	

Tabelle 1: Struktur des Registersatzes

6.1. ACU nach SCU Device Interface

Nicht alle Register des ACU nach SCU Device Interfaces werden gegenwärtige benutzt.

Nicht genutzte Register sind nachfolgend farblich hervorgehoben.

Register deren Beschreibung nicht Bestandteil dieses Dokuments sind, sind bläulich oder grünlich hinterlegt.

	Register	Offset	Adresse	w/r	Beschreibung
	Quad_FunctionGenerator_1_1 ^{*)}		0300 _h	r/w	In diesem Adressraum ist der erste quadratische Funktionsgenerator (Stromsollwert) adressiert
	Quad_FunctionGenerator_1_32		033F _h	r/w	
	Quad_FunctionGenerator_2_1 ^{*)}		0340 _h	r/w	
	Quad_FunctionGenerator_2_32		037F _h	r/w	
Device Interface	Status_1	00 _d	1000 _h	r	In diesen Registern wird der Status des Gerätes angezeigt
	Status_2	01 _d	1001 _h	r	
	Status_3	02 _d	1002 _h	r	
	r	
	Status_32	31 _d	101F _h	r	Mit den Kommandoregistern wird das Gerät gesteuert.
	Command_1	32 _d	1020 _h	r/w	
	Command_2	33 _d	1021 _h	r/w	
	Command_3	34 _d	1022 _h	r/w	
	r/w	
	Command_32	47 _d	102F _h	r/w	
	Lin_FunctionGenerator_1_1 ^{**)}	48 _d	1030 _h	r/w	In diesem Adressraum ist der erste lineare Funktionsgenerator (Stromsollwert) adressiert
	Lin_FunctionGenerator_1_32	79 _d	104F _h	r/w	
	Lin_FunctionGenerator_2_1 ^{**)}	80 _d	1050 _h	r/w	In diesem Adressraum ist der zweite lineare Funktionsgenerator (Spannungssollwert) adressiert
	Lin_FunctionGenerator_2_32	111 _d	106F _h	r/w	
Device Interface	CurrentValue_1_LW	112 _d	1070 _h	r	Istwert 1 Bit 0 - 15
	CurrentValue_1_HW	113 _d	1071 _h	r	Istwert 1 Bit 16 - 31
	CurrentValue_2_LW	128 _d	1080 _h	r	Istwert 2 Bit 0 - 15
	CurrentValue_2_HW	129 _d	1081 _h	r	Istwert 2 Bit 16 - 31
	SetValue_1_Scaling_1	144 _d	1090 _h	r/w	Normierung, Skalierung und Einheit Sollwert 1
	SetValue_1_Scaling_16	159 _d	109F _h	r/w	
	SetValue_2_Scaling_1	256 _d	1100 _h	r/w	Normierung, Skalierung und Einheit Sollwert 2
	SetValue_2_Scaling_16	271 _d	110F _h	r/w	
	CurrentValue_1_Scaling_1	272 _d	1110 _h	r/w	Normierung, Skalierung und Einheit Istwert 1
	CurrentValue_1_Scaling_2	273 _d	1111 _h	r/w	
	CurrentValue_1_Scaling_16	287 _d	111F _h	r/w	
	CurrentValue_2_Scaling_1	288 _d	1120 _h	r/w	Normierung, Skalierung und Einheit Istwert 2
	CurrentValue_2_Scaling_2	289 _d	1121 _h	r/w	
	CurrentValue_2_Scaling_16	303 _d	112F _h	r/w	
	Interlocks_1	512 _d	1200 _h	r	Bit-weise Anzeige der Interlocks des Gerätes
	r	
	Interlocks_64	575 _d	123F _h	r	
	Errors_1_AddrOffset	576 _d	1240 _h	r	Bit-weise Anzeige von Fehlern in der Steuerhardware die zum Ausfall des Gerätes führen
	r	
	Errors_32_AddrOffset	607 _d	125F _h	r	
	Warnings_1_AddrOffset	608 _d	1260 _h	r	Bit-weise Anzeige von Warnungen die zwar vom einem Servicemitarbeiter überprüft werden sollten aber NICHT zum Ausfall des Gerätes führen (Temperatur im Gerät o. Ä.)
	r	
	Warnings_32_AddrOffset	639 _d	127F _h	r	

*) Die quadratischen Funktionsgeneratoren sind ab der MFU LE Firmware 7.0.0, MFU SE Firmware 7.0.0 und MFU LWL Firmware 7.0.0 verfügbar.

***) Die linearen Funktionsgeneratoren sind ab der MFU SE Firmware 7.4.0 und in der MFU LWL nicht mehr verfügbar.

Tabelle 2: Register des ACU nach SCUB Device Interfaces

6.1.1. Quad_Funktionsgeneratoren (0300_h, 033F_h, 0340_h, 034F_h)

Die Register des quadratischen Funktionsgenerators werden nicht in diesem Dokument beschrieben.

6.1.2. Status_1 (1000_h)

Bit	Quelle/Signal	#define	Beschreibung
0	VCC_BUS[0]	PSU_HasControlVoltage_IsAvailable	immer [1] sofern SVE mit Steuerspannung versorgt
1	FRONT_SWITCH_REMOTE_LOCALn_DB	PSU_IsLocalOrRemote	[0] bei Lokalbetrieb, [1] bei Remotebetrieb
2	CPU_STATUS[14]	PSU_BootSequenceFailedOrCompleted	[0] wenn Bootsequenz noch nicht abgeschlossen oder fehlgeschlagen, [1] wenn Bootsequenz erfolgreich abgeschlossen wurde
3	CPU_STATUS[17]	PSU_ParametersNotValidOrValid	[0] wenn Parameter nicht gültig sind, [1] wenn Parameter erfolgreich geladen wurden
4	CPU_STATUS[0]	PSU_IsSwitchedOFFOrON	[0] wenn SVE ausgeschaltet ist (per [OFF] Befehl), [1] wenn SVE eingeschaltet ist (per [ON] Befehl)
5	PSU1_Status1_ControllerEnabled	PSU_ControllerIsDisabledOrEnabled	[0] wenn Regler gesperrt, [1] wenn Regler freigegeben
6	CPU_STATUS[1]	PSU_IsCurrentControlledOrField	[0] wenn Stromgeregelt, [1] wenn Feldgeregelt
7	CPU_STATUS[20]	PSU_HasInterlocksOrNot	[0] wenn Interlocks anstehen, [1] wenn KEINE Interlocks anliegen
8	VCC_BUS[0]	PSU_HasHWWarningsOrNoWarning	[0] wenn HW Warnungen anstehen, [1] wenn KEINE HW Warnungen vorliegen
9	CPU_STATUS[8]	PSU_HasSWWarningsOrNoWarning	[0] wenn SW Warnungen anstehen, [1] wenn KEINE SW Warnungen vorliegen
10	VCC_BUS[0]	PSU_HasHWErrorsOrNoErrors	[0] wenn HW Fehler anstehen, [1] wenn KEINE HW Fehler aufgetreten sind
11	CPU_STATUS[9]	PSU_HasSWErrorsOrNoErrors	[0] wenn SW Fehler anstehen, [1] wenn KEINE SW Fehler aufgetreten sind
12..15	USixHS_ICM_WithUnitStatusInfo[3..0]	0x0: STATUS_UNIT_STATUS_NOSTATUS	No Status Readable / Kein Status lesbar
		0x1: STATUS_UNIT_STATUS_WAIT_FOR_PARAMETERS	Waiting for Parameters / Warten auf Parameter
		0x2: STATUS_UNIT_STATUS_UNITOFF	Unit is Switched Off / SVE ist AUS
		0x3: STATUS_UNIT_STATUS_CHARGING	Charging Capacitor Bank / Kondensatorbank wird geladen
		0x4: STATUS_UNIT_STATUS_SWITCHING_ON	Switching Unit On / SVE wird eingeschaltet
		0x5: STATUS_UNIT_STATUS_UNITON	Unit is On / SVE ist eingeschaltet, Regler ist noch gesperrt (FSM: folgt direkt auf 0x4)
		0x6: STATUS_UNIT_STATUS_CONTROLLER_DISABLED_BY_FPGA_INTERNAL_CAUSE	Contr. disab. FPGA int. / SVE ist eingeschaltet, Regler ist noch gesperrt (FSM: folgt direkt auf 0x5)
		0x7: STATUS_UNIT_STATUS_CONTROLLER_ENABLED	Controller Enabled / SVE ist eingeschaltet, Regler ist frei gegeben
		0x8: STATUS_UNIT_STATUS_SWITCHING_OFF	Switching Unit Off / SVE schaltet ab
		0x9: STATUS_UNIT_STATUS_CONTROLLER_DISABLED_BY_COMMAND	Contr. disab. by Cmd. / SVE ist eingeschaltet, Regler wurde via Kommando (z.B. über PCA) gesperrt
		0xA: STATUS_UNIT_STATUS_CONTROLLER_DISABLED_BY_FPGA_EXTERNAL_CAUSE	Contr. disab. FPGA int. / SVE ist eingeschaltet, Regler wurde ext. Eingang in eine Reglersperre geschaltet
		0xB: STATUS_UNIT_STATUS_RESET_INTERLOCKS	Resetting Interlocks / Interlock Reset wird durchgeführt
		0xC: STATUS_MACHINE_PROTECTION	Machine Protection
		0xD: --	
		0xE: STATUS_UNIT_STATUS_POWERON_RESET	Power On Reset / Power On Reset wird durchgeführt
0xF: STATUS_UNIT_STATUS_NOT_DEFINED	WhenOthers / Wenn alle Stricke reißen		

Tabelle 3: Bitbedeutung Status_1

6.1.3. Status_2 (1001h)

Bit	Quelle/Signal	#define	Beschreibung
13..15	GND_BUS[2..0]	immer [0]	
12	CPU_STATUS[19]	PSU_ReceivingSystemParametersRAM	[1] wenn Systemparameter ins RAM geschrieben werden, z.B. durch lokalen PC
11	CPU_STATUS [18]	PSU_AtLeasOneModuleNotVerifiedOrAllModulesVerfied	[0] wenn mind. ein Modul nicht verifizierbar, [1] wenn alle Module der ACU einsatzbereit sind
10	CPU_STATUS[16]	PSU_LoadingInternalParameters	[1] wenn gerade die internen Parameter aus dem eeflash geladen
9	CPU_STATUS[15]	PSU_NiosII_Watchdog	z.Zt. nicht benutzt
8	CPU_STATUS[13]	PSU_VNC2NotProgrammed	Der VNC2 (Hostcontrolelr) ist nicht programmiert
7	CPU_STATUS[12]	PSU_USBDevicePermitted	wenn [1] wurde eine "legitimierter" USB-Stick erkannt und es dürfen ALLE Menuezugriffe erfolgen
6	CPU_STATUS[11]	PSU_USBDeviceDetected	wenn [1] wurde ein USB-Stick am USB Port erkannt
5	CPU_STATUS[10]	PSU_UsingInternalParameters	wenn [1] werden die MFU internen Parameter verwendet
4	CPU_STATUS[7]	PSU_RecordingSystemParameters	wenn [1] empfängt die MFU Systemparameter und zeichnet diese ins int. EEFLASH auf
3	CPU_STATUS[6]	PSU_FetchingInterlocks	wenn [1] sammelt die MFU Interlockinformationen von den Modulen ein
2	CPU_STATUS[5]	PSU_PerformingUSIScan	wenn [1] führt die MFU einen USI Suchlauf durch
1	CPU_STATUS[4]	PSU_ResetButtonActive	wenn [1] wird die RESET-Taste an der MFU Front gedrückt
0	CPU_STATUS[3]	PSU_StandardScreenActive	wenn [1] zeigt die MFU den Standardbildschirm an

Tabelle 4: Bitbedeutung Status_2

6.1.4. Command_1 (1020h)

Bit	Quelle/Signal	#define	Beschreibung
15..4	n.u.		
3..0		0x0: cCmd_NoAction	
		0x1: cCmd_SwitchUnitON	
		0x2: cCmd_SwitchUnitOFF	
		0x3: cCmd_UnitRESET	
		0x4: cCmd_ControllerLocked	

Tabelle 5: Bitbedeutung Command_1**6.1.5. Command_2 (1021h)**

Hierrüber wird im Falle eines Lastumschaltmoduls die Ziel-Last ausgewählt.

Ist keine Last ausgewählt (Register ist 0x0000) und ist ein Lastumschaltmodul vorhanden, wird dieses Lastumschaltmodul die Reißleine nicht freigeben und dadurch das Einschalten der SVE verhindern.

Der Status der gewählten Last ist über das Statusregister Status_3 rücklesbar. Sind die Inhalte von Status_3 und Command_2 identisch, ist die ausgewählte Last angefahren.

Bit	Quelle/Signal	#define	Beschreibung
8..15	Nicht benutzt		
7	cLoad_8_Selected_BitPos	0x80	Last 8 anwählen
6	cLoad_7_Selected_BitPos	0x40	Last 7 anwählen
5	cLoad_6_Selected_BitPos	0x20	Last 6 anwählen
4	cLoad_5_Selected_BitPos	0x10	Last 5 anwählen
3	cLoad_4_Selected_BitPos	0x08	Last 4 anwählen
2	cLoad_3_Selected_BitPos	0x04	Last 3 anwählen
1	cLoad_2_Selected_BitPos	0x02	Last 2 anwählen
0	cLoad_1_Selected_BitPos	0x01	Last 1 anwählen

Tabelle 6: Bitbedeutung Command_2**6.1.6. Linearer FunctionGenerator_... (1030h, 104Fh, 1050h, 106Fh)**

Die Register der Funktionsgeneratoren werden nicht in diesem Dokument beschrieben.

6.1.7. CurrentValue_1_... (1070h, 1071h)

Liefert den aktuellen Istwert A entsprechend dem FSP020 der MFU. I.d.R. wird dies der aktuelle Stromwert der SVE sein.

CurrentValue_1_LW															lesen
1070 _H /Offset 112 _D															Reset:0x(siehe Beschreibung) _H
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CurrentValue_1_LW															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

CurrentValue_1_HW															lesen
1071 _H /Offset 113 _D															Reset:0x(siehe Beschreibung) _H
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CurrentValue_1_HW															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Zum aktuellen Zeitpunkt (MFU FW 7.5.1) liefert das ACU-System 20 Bit Istwerte. Es sind aber nur die oberen 16 davon an die CurrentValue_1_...-Register angeschlossen.

[15..0] CurrentValue_1_HW – obere 16 Bit des 20 Bit Istwertes A

[15..0] CurrentValue_1_LW – immer [0]

6.1.8. CurrentValue_2_... (1080_h, 1081_h)

Liefert den aktuellen Istwert B entsprechend dem FSP021 der MFU. I.d.R. wird dies der aktuelle Spannungswert der SVE sein.

CurrentValue_2_LW															lesen
1080 _H /Offset 128 _D															Reset:0x(siehe Beschreibung) _H
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CurrentValue_2_LW															
r	R	r	r	r	r	r	r	r	r	r	r	r	r	r	r

CurrentValue_2_HW																Lesen
1081 _H /Offset 129 _D																Reset:0x(siehe Beschreibung) _H
15	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
CurrentValue_2_HW																
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Zum aktuellen Zeitpunkt (MFU FW 7.5.1) liefert das ACU-System 20 Bit Istwerte. Es sind aber nur die oberen 16 davon an die CurrentValue_2_...-Register angeschlossen.

[31..16] CurrentValue_2_HW – obere 16 Bit des 20 Bit Istwertes B

[15..0] CurrentValue_2_LW – immer [0]

6.1.9. SetValue_... (1090_h...110F_h)

Die SetValue_...-Register werden gegenwärtig noch nicht genutzt.

6.1.10. CurrentValue_... (1110_h, 112F_h)

Die CurrentValue_...-Register werden gegenwärtig noch nicht genutzt.

6.1.11. Interlocks_... (1200_h...123F_h)

Kommunizieren die von den Modulen an den USIs abgefragten Interlocks nach außen zur weiteren Verarbeitung. Es werden die aktuell anstehenden Interlocks übermittelt.

Die genau verwendete Anzahl von Interlocks_...-Registern ist abhängig von der Gerätekonfiguration.

Es können max. 1024 Interlock Informationen mittels der 64 Interlocks_...-Register übermittelt werden.

Das Füllen der Interlocks_...-Register erfolgt dabei nach dem folgenden Prinzip.

Beispiel:

An einer MFU sind 4 Module angeschlossen

USI 1: Modul 1 mit 10 Interlocks

USI 2: Modul 1 mit 24 Interlocks

USI 4: Modul 1 mit 4 Interlocks, Modul 2 mit 24 Interlocks

In der Summe werden also mindestens $10+24+4+24=62$ Interlockbits benötigt. Interlocks werden aber immer in Speichertiefen mit vielfachen von 8 verarbeitet, d.h. für das Modul an USI 1 werden 16 Bit und das Modul 1 an USI 4 acht Bit verwendet. In diesem Beispiel werden also 72 Bit, bzw. die ersten 5 Interlocks_...-Register verwendet.

Das Füllen der Interlocks_...-Register geschieht wie folgt:

Beginnend von Interlocks_1[Bit 0] werden die Interlockinformationen der einzelnen USIs und deren Module aufsteigend zum MSB eingetragen.

Interlocks_1															lesen
1200 _H /Offset 512 _D															Reset:0x(siehe Beschreibung) _H
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
frei, immer [1]						USI 1, Modul 1: Interlockbits[9..0]									
r	R	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Interlocks_2															lesen
1201 _H /Offset 513 _D															Reset:0x(siehe Beschreibung) _H
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USI 2, Modul 1: Interlockbits[15..0]															
r	R	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Interlocks_3														lesen	
1202 _H /Offset 514 _D														Reset:0x(siehe Beschreibung) _H	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
frei, immer [1]				USI 4, Modul 1: Interlockbits [3..0]				USI 2, Modul 1: Interlockbits[23..16]							
r	R	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Interlocks_4														lesen	
1203 _H /Offset 515 _D														Reset:0x(siehe Beschreibung) _H	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USI 4, Modul 2: Interlockbits[15..0]															
r	R	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Interlocks_5														lesen	
1204 _H /Offset 516 _D														Reset:0x(siehe Beschreibung) _H	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Unbenutzt, immer [1]								USI 4, Modul 2: Interlockbits[23..16]							
r	R	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Die notwendige Information darüber, welches Interlockbit welchem Modulinterlock entspricht wird mittels PowerConfigAdvanced erstellt. Aus der Gerätekonfiguration lässt sich dort eine InterlockListFile_ . . csv Datei erstellen.

Eine derartige Datei kann wie folgt aussehen.

Interlocks_1-Register, Interlocks_2-Register, Interlocks_3-Register

Power Config Advanced Interlock list file								
Generated from "UnnamedPowerSupplyUnit.xpc7" configuration file.								
=====								
Global interlock bit number	Module name	USI number	Module number	Module interlock bit number	Interlock type	Interlock name	Interlock name on TFT displayed	interlock in use
[0]	18 Bit ADC Module II V7.1.x	1	1	[0]	Analog interlock	Comp Neg 2 Average (Channel 4)	Comp Neg 2 Average (Channel 4)	True
[1]		1	1	[1]	Analog interlock	Comp Neg 1 (Channel 3)	Comp Neg 1 (Channel 3)	True
[2]		1	1	[2]	Analog interlock	Comp Pos 2 Average (Channel 2)	Comp Pos 2 Average (Channel 2)	True
[3]		1	1	[3]	Analog interlock	Comp Pos 1 (Channel 1)	Comp Pos 1 (Channel 1)	True
[4]		1	1	[4]	Digital eletrical interlock	DCCT Error	DCCT Error	True
[5]		1	1	[5]	-	-	-	False

[6]		1	1	[6]	-	-	-	False
[7]		1	1	[7]	-	-	-	False
[8]	ADC DAC IO Module II V7.2.x	3	1	[0]	Digital software interlock	Digital Interlock 01	Digital Interlock 01	True
[9]		3	1	[1]	Digital software interlock	Digital Interlock 02	Digital Interlock 02	True
[10]		3	1	[2]	Digital software interlock	Digital Interlock 03	Digital Interlock 03	True
[11]		3	1	[3]	Digital software interlock	Digital Interlock 04	Digital Interlock 04	True
[12]		3	1	[4]	Digital software interlock	Digital Interlock 05	Digital Interlock 05	True
[13]		3	1	[5]	Digital software interlock	Digital Interlock 06	Digital Interlock 06	True
[14]		3	1	[6]	Digital software interlock	Digital Interlock 07	Digital Interlock 07	True
[15]		3	1	[7]	Digital software interlock	Digital Interlock 08	Digital Interlock 08	True
[16]		3	1	[8]	Digital software interlock	Digital Interlock 09	Digital Interlock 09	True
[17]		3	1	[9]	Digital software interlock	Digital Interlock 10	Digital Interlock 10	True
[18]		3	1	[10]	Digital software interlock	Digital Interlock 11	Digital Interlock 11	True
[19]		3	1	[11]	Digital software interlock	Digital Interlock 12	Digital Interlock 12	True
[20]		3	1	[12]	Digital eletrical interlock	Main Contactor Closed Failure	Main Contactor Closed Failure	True
[21]		3	1	[13]	Undefined or disabled	Unused Bit 13	Unused Bit 13	False
[22]		3	1	[14]	Analog interlock	Negative comparator 01	Negative comparator 01	True
[23]		3	1	[15]	Analog interlock	Positive comparator 01	Positive comparator 01	True
[24]		3	1	[16]	Analog Software interlock	Negative comparator 02	Negative comparator 02	True
[25]		3	1	[17]	Analog Software interlock	Positive comparator 02	Positive comparator 02	True
[26]		3	1	[18]	Analog Software interlock	Negative comparator 03	Negative comparator 03	True
[27]		3	1	[19]	Analog Software interlock	Positive comparator 03	Positive comparator 03	True
[28]		3	1	[20]	Analog Software interlock	Negative comparator 04	Negative comparator 04	True
[29]		3	1	[21]	Analog Software interlock	Positive comparator 04	Positive comparator 04	True
[30]		3	1	[22]	Analog Software interlock	Negative comparator 05	Negative comparator 05	True
[31]		3	1	[23]	Analog Software interlock	Positive comparator 05	Positive comparator 05	True
[32]		3	1	[24]	Digital software interlock	USI Highspeed 01	USI Highspeed 01	True
[33]		3	1	[25]	Digital software interlock	USI Highspeed 02	USI Highspeed 02	True
[34]		3	1	[26]	Digital software interlock	USI Highspeed 03	USI Highspeed 03	True
[35]		3	1	[27]	Undefined or disabled	Unused Bit 27	Unused Bit 27	False
[36]		3	1	[28]	Undefined or disabled	Unused Bit 28	Unused Bit 28	False
[37]		3	1	[29]	Undefined or disabled	Unused Bit 29	Unused Bit 29	False
[38]		3	1	[30]	Undefined or disabled	Unused Bit 30	Unused Bit 30	False
[39]		3	1	[31]	Undefined or disabled	Unused Bit 31	Unused Bit 31	False
=====								
END OF FILE								

Abbildung 1: Beispiel einer „InterlockListFile_..csv“ Datei

Wenn ein USI mit Modul(en) registriert ist und eines der Module oder das gesamte USI verloren geht, werden alle zugehörigen Interlockbits auf ,0] gesetzt.

D.h. stehen ALLE Interlocks für ein Modul zeitgleich an, ist vermutlich dieser Fall eingetreten.

6.1.12. Errors_... (1240h...125Fh)

Die Errors_...-Register werden gegenwärtig noch nicht genutzt.

6.1.13. Warnings_... (1260h, 127Fh)

Die Warnings_...-Register werden gegenwärtig noch nicht genutzt.

7. Register Kompatibilitäts-Block

Sowohl die SCU als auch die veraltetete Interfacekarte (IFK) kann auf den Register Kompatibilitäts-Block zugreifen um einen Parallelbetrieb älterer Geräte ohne Netzwerk zu gewährleisten. In der Regel wird es aber so sein, dass (sobald vorhanden) eine SCU den Register Kompatibilitäts-Block (0x0400er/0x0800er Adressbereich) nicht mehr bedient, sondern die deutlich umfangreicheren SCU exklusiven Register benutzt.

Der Vollständigkeit halber sind nachfolgend die Register des Kompatibilitäts-Blocks aufgelistet.

Register	Funktion	R/W	Beschreibung	Info
0x0410	ACU_SET_VALUE_TEMP	R	Strom SOLL-Wert	Undefiniert, wurde 0x0810 nicht zuvor beschrieben, andernfalls der Inhalt von 0x0800
0x0411		R	Kombination versch. Zustände	X"0" & sIFK_Command [0x0 = NOP, 0x1 = EIN, 0x2= AUS, 0x3 = RESET] & sCommutator [0 = normal, 1 = invers] & sIFK_LoadSwitch(3 downto 0) [0x0..0x8];
0x0800	ACU_RESET	W	Kommando RESET	Beliebigen Wert schreiben → löst RESET Kommando aus.
0x0802	ACU_POWER_ON	W	Kommando EIN	Beliebigen Wert schreiben → löst ON Kommando aus.
0x0804	ACU_POWER_OFF	W	Kommando AUS	Beliebigen Wert schreiben → löst OFF Kommando aus.
0x0806	ACU_POLARITY_NORMAL	W	Polwender Polarität positiv	
0x0808	ACU_POLARITY_INVERS	W	Polwender Polarität negativ	
0x0810	ACU_CURRENTS	R/W	Strom SOLL-Wert	Sollwertregister (16Bit) Ausgabe erfolgt abhängig von 0x0822 und ggf. 0x0812.
0x0812	ACU_REALIZE_CURRENTS	W	Getriggerte Ausgabe des SOLL-Wertes.	Beliebigen Wert schreiben → erzeugt die Ausgabe des SOLL-Wertes aus 0x0810, wenn zunächst 0x0822 entsprechend gesetzt ist.
0x0814	ACU_CONNECT_LOAD_1	W	Lastwahlschalter LAST1	Beliebigen Wert schreiben → wählt LAST1 (00001 ₆).
0x0816	ACU_CONNECT_LOAD_2	W	Lastwahlschalter LAST2	Beliebigen Wert schreiben → wählt LAST2 (00010 ₆).
0x0818	ACU_CONNECT_LOAD_3	W	Lastwahlschalter LAST3	Beliebigen Wert schreiben → wählt LAST3 (00100 ₆).
0x081A	ACU_CONNECT_LOAD_4	W	Lastwahlschalter LAST4	Beliebigen Wert schreiben → wählt LAST4 (01000 ₆).
0x081C	ACU_CONNECT_LOAD_5	W	Lastwahlschalter LAST5	Beliebigen Wert schreiben → wählt LAST5 (10000 ₆).
0x081E	ACU_CUT_LOAD	W	Lastwahlschalter KEINE	Beliebigen Wert schreiben → wählt deaktiviert alle Lasten (00000 ₆).
0x0820	ADC_SOFTTRIGGER	W	Getriggertes Einlesen von ADC Werten (Istwert)	Beliebigen Wert schreiben → erzeugt das Einlesen eines Istwertes in 0x0830, wenn zunächst 0x0824 entsprechend gesetzt ist.
0x0822	ACU_WR_DAC_MODE	R/W	DAC Modes	Definiert unter welchen Bedingungen der SOLL-Wert ausgegeben werden soll. 0xdd0 → Sollwert direkt über SCU 0xdd1 → Trigger über DACSofttrigger (Wert von 0x0810 erst ausgeben wenn 0x0812 geschrieben wird 0xdd2 → Trigger über ext. Trigger (Wert von 0x0810 erst ausgeben, wenn [DAC_ExternalTrigger] = [1] wird.
0x0824	ACU_WR_ADC_MODE	R/W	ADC Modes	Definiert unter welchen Bedingungen IST-Werte eingelesen werden. <u>DCCT Wert (vom ADC1):</u> 0xdbb0 → Trigger über Ausleseroutine (direkt lesen über 0x0830) 0xdbb1 → Trigger über ADCSofttrigger (lesen triggern wenn 0x0820 geschrieben wird, auslesen über 0x0830) 0xdbb2 → Trigger über ext. Trigger (lesen wenn [ADC_ExternalTrigger] = [1] wird, auslesen über 0x0830) <u>HALL1 Wert (vom ADC2):</u> 0xdb0b → Trigger über Ausleseroutine (direkt lesen über 0x0832) 0xdb1b → Trigger über ADCSofttrigger (lesen wenn 0x0820 geschrieben wird, auslesen über 0x0832) 0xdb2b → Trigger über ext. Trigger (lesen wenn [ADC_ExternalTrigger] = [1] wird, auslesen über 0x0832) <u>HALL2 Wert (vom ADC3):</u> 0xd0bb → Trigger über Ausleseroutine (direkt lesen über 0x0834)

				0xd1bb → Trigger über ADCSofttrigger (lesen wenn 0x0820 geschrieben wird, auslesen über 0x0834) 0xd2bb → Trigger über ext. Trigger (lesen wenn [ADC_ExternalTrigger] = [1] wird, auslesen über 0x0834)
0x0830	ACU_CURRENTI	R	Strom IST-Wert	
0x0832	ACU_FIELDI_1	R	(Feld)Istwert 2 (Hall)	
0x0834	ACU_FIELDI_2	R	(Feld)Istwert 3 (Hall)	
0x0836	ACU_FIELDI_3	R	--	undefiniert
0x0838	ACU_FIELDI_4	R	--	undefiniert
0x083A	ACU_FIELDI_5	R	--	undefiniert
0x083C	ACU_STATE_1	R	Statusbyte 1	IFK_Statusbyte 1 xxxx xxxx \----- [1] PSE eingeschaltet, [0] PSE ausgeschaltet \----- Interlock UNetz \----- Interlock Temperatur Kühlbank und Trafo \----- Interlock Kühlwasser \----- Interlock Lastüberstrom (ADC Modul) \----- Status Polwender, Status via [Commutator] Eingang \----- Interlock Temperatur Last \----- Interlock Kühlwasser Last
0x083E	ACU_STATE_2	R	Statusbyte 2	IFK_Statusbyte 2 xxxx xxxx \----- Interlock Primärstrom \----- Interlock Primärstrom 1 (z Zt. noch nicht unterstützt) \----- Interlock IGBT V1-V4 Auf das Bit für V1-V4 sind jetzt noch die Transistoren V5-V6 belegt. \----- Interlock IGBT V5-V6 V5-V6 zeigen jetzt an das das Gerät mit einer ext. Reglersperre versehen ist. \----- Interlock Zwischenkreis \----- Interlock Schwingungsüberwachung UND/ODER Sicherheits- Aus UND/ODER Sicherung \----- Interlock DCCT ERROR \----- Interlock Erdschluss
0x0840	ACU_STATE_3	R	Statusbyte 3	IFK_Statusbyte 3 xxxx xxxx \----- [1] PSE ist im Remotebetrieb, [0] PSE ist im Localbetrieb \----- n.u. \----- Magnet Nr.1 Aktiv \----- Magnet Nr.2 Aktiv \----- Magnet Nr.3 Aktiv \----- Magnet Nr.4 Aktiv \----- Magnet Nr.5 Aktiv \----- Stromregelung/Feldregelung, via [Field_Controlled] Eingang

Tabelle 7: Register Kompatibilitäts-Block