



# FSPs des ADCII Moduls (ab FW 7.0)

Version vom: Mittwoch, 4. Mai 2022, 12:02:00

## Inhaltsverzeichnis

1.	Änderungsliste .....	1
2.	FSPs ADC Modul II .....	2
	FSP001_ModuleStatus .....	3
	0x01 <sub>H</sub> /1 <sub>D</sub> /0x3031 <sub>ASCII</sub> .....	
	FSP002_ModuleWarnings .....	4
	0x02 <sub>H</sub> /2 <sub>D</sub> /0x3032 <sub>ASCII</sub> .....	
	FSP003_ModuleErrors.....	5
	0x03 <sub>H</sub> /3 <sub>D</sub> /0x3033 <sub>ASCII</sub> .....	
	FSP004_ModuleInterlocks.....	6
	0x04 <sub>H</sub> /4 <sub>D</sub> /0x3034 <sub>ASCII</sub> .....	
	FSP006_InterlocksArrivalSequence .....	7
	0x06 <sub>H</sub> /6 <sub>D</sub> /0x3036 <sub>AS</sub> .....	
	FSP009_ModuleSerialNumber .....	8
	0x09 <sub>H</sub> /9 <sub>D</sub> /0x3039 <sub>ASCII</sub> .....	
	FSP010_ModuleCommands .....	9
	0x0A <sub>H</sub> /10 <sub>D</sub> /0x3041 <sub>ASCII</sub> .....	
	FSP011_ModuleInterlocksMask_n .....	10
	0x0B <sub>H</sub> /11 <sub>D</sub> /0x3042 <sub>ASCII</sub> .....	
	FSP012_USIConfig.....	11
	0x0C <sub>H</sub> /12 <sub>D</sub> /0x3043 <sub>ASCII</sub> .....	
	FSP013_PeripheralConfig .....	12
	0x0D <sub>H</sub> /13 <sub>D</sub> /0x3044 <sub>ASCII</sub> .....	
	FSP030_SetValue_A .....	13
	0x1E <sub>H</sub> /30 <sub>D</sub> /0x3145 <sub>ASCII</sub> .....	
	FSP040_RemoteUpdateStatus.....	14
	0x28 <sub>H</sub> /40 <sub>D</sub> /0x3238 <sub>ASCII</sub> .....	
	FSP041_RemoteUpdateCommands.....	15
	0x29 <sub>H</sub> /41 <sub>D</sub> /0x3239 <sub>ASCII</sub> .....	
	FSP042_RemoteUpdateData.....	16
	0x2A <sub>H</sub> /42 <sub>D</sub> /0x3241 <sub>ASCII</sub> .....	
	FSP043_FSP_Image_Generator .....	18
	0x2B <sub>H</sub> /43 <sub>D</sub> /0x3242 <sub>ASCII</sub> .....	
	FSP044_FSP_Image_Status.....	19
	0x2C <sub>H</sub> /44 <sub>D</sub> /0x3243 <sub>ASCII</sub> .....	
	FSP045_AlteraRemoteUpdateCmd.....	20
	0x2D <sub>H</sub> /45 <sub>D</sub> /0x3244 <sub>ASCII</sub> .....	
	FSP046_AlteraRemoteUpdateStatus.....	21
	0x2E <sub>H</sub> /46 <sub>D</sub> /0x3245 <sub>ASCII</sub> .....	
	FSP050_ModuleSupplyValues .....	22
	0x32 <sub>H</sub> /50 <sub>D</sub> /0x3332 <sub>ASCII</sub> .....	
	FSP051_ModulePotiValues .....	23
	0x33 <sub>H</sub> /51 <sub>D</sub> /0x3333 <sub>ASCII</sub> .....	
	FSP052_ModuleComparatorValues.....	24
	0x34 <sub>H</sub> /52 <sub>D</sub> /0x3334 <sub>ASCII</sub> .....	
	FSP053_ModuleTemperatures.....	25
	0x35 <sub>H</sub> /53 <sub>D</sub> /0x3335 <sub>ASCII</sub> .....	
	FSP054_ModuleTemperaturesComparationThresholds.....	26
	0x36 <sub>H</sub> /54 <sub>D</sub> /0x3336 <sub>ASCII</sub> .....	
	FSP058_ParameterChecksumValue .....	27
	0x3A <sub>H</sub> /58 <sub>D</sub> /0x3341 <sub>ASCII</sub> .....	
	FSP059_ParameterChecksumValueCalculated .....	28
	0x3B <sub>H</sub> /59 <sub>D</sub> /0x3342 <sub>ASCII</sub> .....	
	FSP060_ADC_Configuration .....	29
	0x3C <sub>H</sub> /60 <sub>D</sub> /0x3343 <sub>ASCII</sub> .....	
	FSP061_ADCCalibrationGAIN .....	30
	0x3D <sub>H</sub> /61 <sub>D</sub> /0x3344 <sub>ASCII</sub> .....	
	FSP062_ADCCalibrationOFFSET .....	31
	0x3E <sub>H</sub> /62 <sub>D</sub> /0x3345 <sub>ASCII</sub> .....	
	FSP063_ADCAVRResult.....	32
	0x3F <sub>H</sub> /63 <sub>D</sub> /0x3346 <sub>ASCII</sub> .....	
	FSP073_Controller_Limits .....	33
	0x49 <sub>H</sub> /73 <sub>D</sub> /0x3439 <sub>ASCII</sub> .....	
	FSP074_Controller_PI_Settings.....	34
	0x4A <sub>H</sub> /60 <sub>D</sub> /0x3441 <sub>ASCII</sub> .....	
	FSP075_Controller_I_Part_ComparatorLimits.....	35

0x4B <sub>H</sub> /75 <sub>D</sub> /0x3442 <sub>ASCII</sub>	
FSP076_Controller_SetValueDeviation.....	36
0x4C <sub>H</sub> /76 <sub>D</sub> /0x3443 <sub>ASCII</sub>	
FSP077_Controller_PI_Output .....	37
0x4D <sub>H</sub> /77 <sub>D</sub> /0x3444 <sub>ASCII</sub>	

**1. Änderungsliste**

Datum	Name	Kommentar
01.07.2021	D. Schupp	Dokument erstellt aus ACU-FSP mUSlc TFT

## **2. FSPs ADC Modul II**

Dieses Dokument behandelt modulspezifische FSPs des ADC Moduls II FG660\_043 und höher.

Name	<b>FSP001_ModuleStatus</b>
Adresse	<b>0x01<sub>H</sub>/1<sub>D</sub>/0x3031<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

- [23..16]    ModuleStatus  
Ablaufsteuerungszustand des Moduls, mit diesem 8 Bit Wert wird der Zustand der im Modul integrierten Ablaufsteuerung dargestellt.  
n.u., immer 0
- [15..9]    Reserviert  
Reserviert für zukünftige Anwendungen  
n.u. immer ,0'
- [8]        USIsHighSpeed  
wenn ,1' ist USI im Highspeed Mode
- [7..6]    Reserviert  
Reserviert für zukünftige Anwendungen  
n.u. immer ,0'
- [5]        NoInterlocks  
wenn ,1' stehen keine Interlocks an  
Im Modul sind keine Interlocks gespeichert und es stehen auch keine Interlocks an.
- [4]        NoErrors  
wenn ,1' ist Modul fehlerfrei  
Im Modul sind keine Fehler gespeichert die den Betrieb stören.  
z. Zt. immer ,1'
- [3]        NoWarnings  
wenn ,1' ist Modul ohne Warnungen  
Im Modul sind keine Warnmeldungen vorhanden die den Betrieb zwar nicht stören aber trotzdem überprüft werden müssten (Details im FSP für die Warnungsbits) z.B. Temperatur zu hoch.  
z. Zt. immer ,1'
- [2]        ModuleReady  
wenn ,1' ist Modul betriebsbereit  
Das Modul ist voll betriebsbereit  
z. Zt. immer ,1'
- [1]        ChecksumOK  
wenn ,1' Parameter Checksumme OK  
Die Prüfsumme für die Modulparameter ist bestätigt.
- [0]        ParametersLoaded  
wenn ,1' sind die Parameter geladen  
Das Modul hat seine Konfigurationsparameter geladen.  
z. Zt. immer ,1'

Name	<b>FSP002_ModuleWarnings</b>
Adresse	<b>0x02<sub>H</sub>/2<sub>D</sub>/0x3032<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Im FSP werden alle Warnungen bitcodiert aufgelistet die den unmittelbaren Betrieb des Moduls nicht stören, aber trotzdem von einem Techniker untersucht werden müssen, dargestellt ( z.B. Temperatur des Moduls zu hoch ).

Liegt eine Warnung vor ist das korrespondierende Bit ,0' andernfalls ,1'. Außerdem ist Bit [3] das FSP001\_ModuleStatus = ,0'.

[23..0]      n.u., immer ,1'

Name	<b>FSP003_ModuleErrors</b>
Adresse	<b>0x03<sub>H</sub>/3<sub>D</sub>/0x3033<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Der FSP enthält alle Fehler die den unmittelbaren Betrieb des Moduls und damit des Übergeordneten Gerätes gefährdet und zu einer Abschaltung führt.

Liegt ein Fehler vor ist das korrespondierende Bit ,0' andernfalls ,1'. Außerdem ist Bit [4] das FSP001\_ModuleStatus = ,0'.

[23..0]      n.u., immer ,1'



Name	<b>FSP004_ModuleInterlocks</b>
Adresse	<b>0x04<sub>H</sub>/4<sub>D</sub>/0x3034<sub>ASCII</sub></b>
Tiefe	2 Byte / 16 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Im FSP sind alle Interlocks des Moduls abgebildet sowohl die aktuell anstehenden wie auch die gespeicherten Interlockmeldungen.

Die Grenze liegt in der Mitte des FSP, wobei immer ganze Bytes für die Gruppen verwendet werden müssen. d.h. für 3 Interlocks müssen trotzdem 2 Bytes verwendet werden. 1 Byte für den aktuellen Status und 1 Byte für die gespeicherte Meldung.

Die unteren Bytes [n/2..0] des FSP sind für den aktuellen Status bestimmt und die oberen Bytes [n .. n/2] für die gespeicherten Interlocks.

Liegt ein Interlock vor ist das korrespondierende Bit ,0' andernfalls ,1'. Außerdem ist Bit [5] das FSP001\_ModuleStatus = ,0'.

Nicht genutzte Interlockbits müssen ,1' sein!

#### **Gespeicherte Interlocks**

[15..13] n.u., immer ,1'

[12] wenn '0', Steckerüberwachung DCCT, wird auch ADC stirnseitig durch Wechselblinken der roten LEDs signalisiert.

[11] wenn '0', Komparator 1 Positiv-Überwachung

[10] wenn '0', Komparator 2 Positiv-Überwachung Mittelwert

[9] wenn '0', Komparator 3 Negativ-Überwachung

[8] wenn '0', Komparator 4 Negativ-Überwachung Mittelwert

#### **Aktuell anstehende Interlocks**

[7..5] n.u., immer ,1'

[4] wenn '0', Steckerüberwachung DCCT, wird auch ADC stirnseitig durch Wechselblinken der roten LEDs signalisiert.

[3] wenn '0', Komparator 1 Positiv-Überwachung

[2] wenn '0', Komparator 2 Positiv-Überwachung Mittelwert

[1] wenn '0', Komparator 3 Negativ-Überwachung

[0] wenn '0', Komparator 4 Negativ-Überwachung Mittelwert

Name	<b>FSP006_InterlocksArrivalSequence</b>
Adresse	<b>0x06<sub>H</sub>/6<sub>D</sub>/0x3036<sub>ASCII</sub></b>
Tiefe	4 Byte / 32 Byte
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) <sub>H</sub>

Dieser FSP liefert die zeitliche Abfolge auftretender Interlocks. Liegen mehrere Interlocks an, kann über diesen FSP die zeitliche Abfolge von deren Auftreten gelesen werden.

Zu beachten ist, treten Interlocks zeitgleich (also im selben Taktzyklus) auf, wird nur das hochwertigste Interlockbit erfasst und in diesem FSP abgelegt.

Werden also z.B. die Interlocks Bit[1], Bit[3] und Bit[7] im selben Takt erfasst, wird nur Bit[7] im FSP abgelegt.

Das FSP erfasst die 3 zuerst auftretenden Interlocks dieses Moduls.

- [31..24] Das Interlockbit des zuerst aufgetretenen Interlocks (i)
- [23..16] Das Interlockbit des vor (i) aufgetretenen Interlocks (i-1)
- [15..8] Das Interlockbit des zuletzt aufgetretenen Interlocks (i-2)
- [7..0] Anzahl der erfassten Interlock im FSP006 insgesamt (0..3)

Name	<b>FSP009_ModuleSerialNumber</b>
Adresse	<b>0x09<sub>H</sub>/9<sub>D</sub>/0x3039<sub>ASCII</sub></b>
Tiefe	6 Byte / 48 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Der FSP enthält die Modul Serien Nummer

Die Serien Nummer ist über einen One Wire Chip von Dallas zu erzeugen, da gewährleistet sein muss das die Serien Nummer weltweit nur einmal vergeben ist.

[47..0]            Seriennummer des Moduls (48 Bit)

Name	<b>FSP010_ModuleCommands</b>
Adresse	<b>0x0A<sub>H</sub>/10<sub>D</sub>/0x3041<sub>ASCII</sub></b>
Tiefe	1 Byte / 8 Bit
I/O	lesen / schreiben
Reset	0x00 <sub>H</sub>

Wenn ein Modul Kommandos unterstützt (Einschalten, Ausschalten, Reset usw.) dann werden diese über diesen FSP gesetzt.

Die Kommandos dieses FSPs werden nur ausgeführt, wenn das Modul mittels Standard-USI angebunden ist. Wird die USI HighSpeed Verbindung verwendet, dann werden die Kommandos aus diesem FSP ignoriert. Stattdessen erfolgt die Kommandoübertragung über die HighSpeed Anbindung.

[7..4] n.u.

[3..0] Diese Kommandos werden von der MFU oder PowerConfigAdvanced gesetzt und steuern die Module-/Gerätefunktionen

[3..0]	Kommando
0x0	<i>cCMDNoAction</i> keine Aktion
0x1	<i>cCMDSwitchUnitOn</i> Gerät einschalten (wenn möglich)
0x2	<i>cCMDSwitchUnitOff</i> Gerät abschalten
0x3	<i>cCMDResetUnit</i> Reset durchführen (z.B. Interlocks)
0x4	<i>cCMDDisableController</i>
0x5	<i>cCMDTriggerSomething</i> hiermit lassen sich Sonderfunktionen in Modulen auslösen

Für das ADC Modul ist nur das Kommando „0x03 (Reset)“ von Belangen um evtl. angefallene und nicht mehr anstehende Interlocks zu löschen.

Name	<b>FSP011_ModuleInterlocksMask_n</b>
Adresse	<b>0x0B<sub>H</sub>/11<sub>D</sub>/0x3042<sub>ASCII</sub></b>
Tiefe	2 Byte / 16 Bit
I/O	Lesen / schreiben
Reset	0xE0E0

Dieser FSP enthält eine Bitmaske, die nicht verwendete Interlocks ausmaskiert, d.h. alle nicht zu benutzenden Interlocks sind mit 1 zu setzen.

Wie beim FSP004\_ModuleInterlocks liegt die Grenze in der Mitte des FSP, wobei immer ganze Bytes für die Gruppen verwendet werden müssen. d.h. für 5 Interlocks müssen trotzdem 2 Bytes verwendet werden. 1 Byte für den aktuellen Status und 1 Byte für die gespeicherte Meldung.

Die unteren Bytes [n/2..0] des FSP sind für den aktuellen Status bestimmt und die oberen Bytes [n .. n/2] für die gespeicherten Interlocks.

#### **Gespeicherte Interlocks**

- [15..13]     n.u., immer ,1'
- [12]        wenn ,1' → Steckerüberwachung DCCT deaktiviert, wird auch ADC stirnseitig durch Wechselblinken der roten LEDs signalisiert.
- [11]        wenn ,1' → Komparator 1 Positiv-Überwachung deaktiviert
- [10]        wenn ,1' → Komparator 2 Positiv-Überwachung Mittelwert deaktiviert
- [9]         wenn ,1' → Komparator 3 Negativ-Überwachung deaktiviert
- [8]         wenn ,1' → Komparator 4 Negativ-Überwachung Mittelwert deaktiviert

#### **Aktuell anstehende Interlocks**

- [7..5]       n.u., immer ,1'
- [4]         wenn ,1' → Steckerüberwachung DCCT deaktiviert, wird auch ADC stirnseitig durch Wechselblinken der roten LEDs signalisiert.
- [3]         wenn ,1' → Komparator 1 Positiv-Überwachung deaktiviert
- [2]         wenn ,1' → Komparator 2 Positiv-Überwachung Mittelwert deaktiviert
- [1]         wenn ,1' → Komparator 3 Negativ-Überwachung deaktiviert
- [0]         wenn ,1' → Komparator 4 Negativ-Überwachung Mittelwert deaktiviert

Name	<b>FSP012_USIConfig</b>
Adresse	<b>0x0C<sub>H</sub>/12<sub>D</sub>/0x3043<sub>ASCII</sub></b>
Tiefe	1 Byte / 8 Bit
I/O	Lesen / schreiben
Reset	0x00 <sub>H</sub>

Dieser FSP definiert die USI Konfiguration

[7]                wenn ,1' USI im HighSpeed Modus, wenn ,0' USI im normalen Modus

[4..3]            n.u.

[2..0]            USI Bitrate

[2..0]	Bitrate
111	115,2 kBit (default)
110	1 MBit
101	2 MBit
100	5 MBit
011	10 MBit
010	16,6 MBit
001	20 MBit
000	25 MBit (Test only!)

Name	<b>FSP013_PeripheralConfig</b>
Adresse	<b>0x0D<sub>H</sub>/13<sub>D</sub>/0x3044<sub>ASCII</sub></b>
Tiefe	1 Byte / 8 Bit
I/O	lesen / schreiben
Reset	0x80 <sub>H</sub>

Auf dem Modul befindliche Peripherie kann mit diesem FSP konfiguriert werden

- [7] Nach einschalten der Spannungsversorgung wird dieses Bit automatisch auf ,1' gesetzt. Werden Parameter durch die MFU oder per PC geladen, die zur Prüfsummenbildung beitragen sollen, muss dieses Bit gelöscht werden, bevor der erste Parameter übertragen wird. Ist das Laden der Parameter beendet, muss dieses Bit wieder auf ,1' gesetzt werden. Im Anschluss daran wird die Vergleichs-Prüfsumme an „FSP058\_ParameterChecksumValue“ gesendet. Die Modul-Freigabe erfolgt aber nur, wenn die Vergleichs-Prüfsumme auch zu der aus den restlichen Parametern gebildeten Prüfsumme passt.  
Das Löschen dieses Bit löscht die zuvor errechnete Prüfsumme.
- [6..0] n.u.

Name	<b>FSP030_SetValue_A</b>
Adresse	<b>0x1E<sub>H</sub>/30<sub>D</sub>/0x3145<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen / schreiben
Reset	0x000000 <sub>H</sub>

Über diesen FSP wird der Sollwert\_A für die Temperatur-Regler PWM der ADC Heizung angegeben.

[23..0]      Vorzeichenbehafteter 20 Bit Wert (0xFFFF bis 0x7FFFF)



Name	<b>FSP040_RemoteUpdateStatus</b>
Adresse	<b>0x28<sub>H</sub>/40<sub>D</sub>/0x3238<sub>ASCII</sub></b>
Tiefe	1 Byte / 8 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Beim Fernupdate wird dieser FSP für das Rücklesen der Statusinformationen des Fernupdates verwendet.

[7..3] n.u.

[2] FSP042\_Busy  
wenn ,1' ist FSP42 beschäftigt (z.B. weil gerade Flashsektoren gelöscht oder programmiert werden) und es sollten KEIN Zugriffe darauf erfolgen

[1] FSP042\_ReadyToSendData,  
wenn ,1' können Daten vom Host an FSP42 abgeholt werden

[0] FSP042\_ReadyToReceiveData,  
wenn ,1' können Daten vom Host an FSP42 gesendet werden

Name	<b>FSP041_RemoteUpdateCommands</b>
Adresse	<b>0x29<sub>H</sub>/41<sub>D</sub>/0x3239<sub>ASCII</sub></b>
Tiefe	1 Byte / 8 Bit
I/O	lesen / schreiben
Reset	0x00 <sub>H</sub>

Dieser FSP überträgt die Kommandos für das Fernupdate

[7..3] n.u.

[2..0] Kommandos für den RemoteUpdateHandler

[2..0]	Kommando
000	NOP
001	Erase Bulk, das gesamte Flash löschen
010	Erase Sector, nur den an 'DataAddress' angegeben Sektor löschen
011	Write single bytes, ein einzelnes Bytes ins Flash schreiben
100	Write continuously, beliebige Anzahl Bytes ins Flash schreiben
101	Read single bytes, ein einzelnes Bytes aus dem Flash lesen
110	Read continuously, beliebige Anzahl Bytes aus dem Flash lesen
111	Init

Name	<b>FSP042_RemoteUpdateData</b>
Adresse	<b>0x2A<sub>H</sub>/42<sub>D</sub>/0x3241<sub>ASCII</sub></b>
Tiefe	256 Byte / 2048 Bit
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) <sub>H</sub>

Dieser FSP überträgt die Daten für das Fernupdate entweder vom Host zum Modul oder umgekehrt.

**Achtung:** Dieses FSP ist besonders im Hinblick auf Lesen und Schreiben. Da das FSP ein nachgeschalteter serieller Flash bedient und die empfangenen Daten direkt in diesen Flash programmiert, bzw. aus dem Flash ausgelesene Daten direkt an den Host versendet werden

#### Zugriffe einleiten

Zugriffe auf FSP042 müssen generell über FSP041 eingeleitet werden.

Das erste „Kommando“ an FSP041 lautet immer „000“ NOP. Der FSP041 muss mit ACK antworten. Darauf erfolgt das Kommando „111“ (Init). FSP041 muss auch hier mit ACK antworten.

#### Lesen

Bevor Daten aus dem FSP042 gelesen werden, muss das Lesen mit FSP041 eingeleitet werden.

Zum Lesen eines einzelnen Byte wird das Kommando: „101“ (Read single byte) an FSP041 gesendet. FSP041 muss mit ACK antworten.

Durch lesen von FSP040 lässt sich kontrollieren, ob FSP042 prinzipiell bereit ist Daten zu senden (Bit[1]).

Anschließend wird FSP042 einmalig gelesen. Dabei wird das erste Byte gesendet. Der Ausleseprozess beginnt an Adresse 0x0 und wird automatisch inkrementiert. D.h. wird ein weiteres Lesekommando an FSP042 geschickt, wird das folgende Byte ausgegeben.

Sollen hingegen die Daten seitenweise (jeweils 256 Byte) gelesen werden, erfolgt dies mit dem Kommando: „110“ an FSP041.

Anschließend wird mit jedem Lesebefehl an FSP042 jeweils eine Seite Daten übertragen. Die Seiten werden dabei automatisch inkrementiert.

#### Schreiben

Bevor Daten sinnvoll ins Flash geschrieben werden können, muss dieses gelöscht werden.

Das Kommando „001“ an FSP041 löscht dieses komplett, das Kommando „010“ an FSP041 hingegen nur die aktuell adressierte Page. Da ein direktes Adressieren der Page im ADCII nicht möglich ist, entfällt die Verwendung dieses Kommandos. In jedem Fall muss FSP042 ACK antworten. Der EPCS Controller beginnt dann unmittelbar mit dem Löschen des Flashs.

Jetzt kann sofort ein erneutes Init-Kommando („111“) an FSP041 gesendet werden. Dieser muss mit ACK antworten.

Jetzt erfolgt die Einleitung des Schreibkommandos.

Zum Schreiben eines einzelnen Byte wird das Kommando: „011“ (Write single bytes) an FSP041 gesendet. FSP041 muss mit ACK antworten.

Sollen hingegen die Daten seitenweise (jeweils 256 Byte) geschrieben werden, erfolgt dies mit dem Kommando: „100“.

Durch lesen von FSP040 lässt sich kontrollieren, ob FSP042 prinzipiell bereit ist Daten zu empfangen, sobald das Bit[0] gesetzt wird. Dieses wird gesetzt, wenn der Löschvorgang abgeschlossen und ein Schreibkommando geschickt wurde. Der Löschvorgang kann bis zu 20 Sekunden dauern.

Anschließend wird abhängig vom Schreibkommando mit dem Schreibbefehl an FSP042 jeweils entweder ein Byte oder jeweils eine Seite Daten ins Flash übertragen. Die Adressen, bzw. Seiten werden dabei automatisch inkrementiert.

Der Schreibvorgang beginnt dabei in jedem Fall bei Adresse 0x0.

#### Abbrechen/Beenden

Alle Zugriffe (schreiben/lesen) auf den Flash über FSP042 lassen sich mit einem „111“ (Init) an FSP041 abbrechen/beenden.

Name	<b>FSP043_FSP_Image_Generator</b>
Adresse	<b>0x2B<sub>H</sub>/43<sub>D</sub>/0x3242<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen / schreiben
Reset	0x00_00_00 <sub>H</sub>

Über diesen FSP wird ein FSP Image erzeugt und im seriellen Flash gespeichert, bzw. die Images der FSPs aus dem seriellen Flash geladen. Der serielle Flash ist hierdurch auch löschtbar. Statusrückmeldungen gibt das „FSP044\_FSP\_Image\_Status“.

#### Hinweis

Zwischen den Funktionen „FSPImgLoadImage“, „FSPImgMakelImage“ und „FSPImgEraseAll“ dürfen die Bitwechsel nicht unmittelbar erfolgen. D.h. sofern eine der Funktion benutzt wurde, muss das zugehörige Bit zuerst gelöscht werden, bevor das Bit einer anderen Funktion gesetzt wird.

- [23..16] FSPImgSectorAddress, sofern FSPImgUseSectorAddress; = ,1' wird der hier angegebene Sektor im ext. seriellen Flash gelöscht, zur Sicherung, bzw. zum Laden von FSP Daten genutzt.
- [15..8] FSPImgFSPNumber, gibt die FSP Nummer an, die mittels ,FSPImgMakelImage' ins ext. serielle Flash gesichert werden soll.
- [7..4] n.u.
- [3] FSPImgUseSectorAddress, wenn ,1' wird das Image in den ,FSPImgSectorAddress' Sektor geschrieben, wenn ,0' wird die Startadresse aus dem ,M25P\_Access' Modulgeneric ,gFSPImgStartAddress' verwendet.
- [2] FSPImgEraseAll, wenn ,1' wird ein Sektor im ext. seriellen Flash gelöscht. Dieser ist abhängig von ,FSPImgUseSectorAddress' und ,FSPImgSectorAddress' oder ,gFSPImgStartAddress'. Zum Starten des Löschvorgangs muss ein ,0' → ,1' Übergang erfolgen.
- [1] FSPImgMakelImage, wenn ,1' wird der Inhalt des FSP ,FSPImgFSPNumber' ins ext. serielle Flash gesichert. Zum Starten des Sicherungsvorgangs muss ein ,0' → ,1' Übergang erfolgen. Sollen mehrere FSP gesichert werden ist ,FSPImgMakelImage' mehrfach zu aktivieren und die jeweilige FSP Nummer in ,FSPImgFSPNumber' anzugeben.
- [0] FSPImgLoadImage, wenn ,1' wird der Inhalt des ext. seriellen Flashs in die FSPs geladen. Sind zuvor mehrere FSP gesichert worden, werden alle Sicherungen geladen. Zum Starten des Ladevorgangs muss ein ,0' → ,1' Übergang erfolgen.

Name	<b>FSP044_FSP_Image_Status</b>
Adresse	<b>0x2C<sub>H</sub>/44<sub>D</sub>/0x3243<sub>ASCII</sub></b>
Tiefe	1 Byte / 8 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Liefert Statusrückmeldungen bzgl. des Sicherungsvorgangs von FSP Daten. Die Sicherung wird über FSP043 gesteuert.

[7..1] n.u., immer ,0'

[0] FSPImgReady, ist ,1' wenn das ext. Flash Zugriffe zum sichern/laden von FSP Daten gestattet, ist ,0' wenn Daten gesichert oder gelesen werden, bzw. ein Löschvorgang durchgeführt wird. Ist diese Bit ,0' und mittels FSP043 werden Zugriffe auf das ext. Flash gestartet ist das Ergebnis undefiniert.

Name	<b>FSP045_AlteraRemoteUpdateCmd</b>
Adresse	<b>0x2D<sub>H</sub>/45<sub>D</sub>/0x3244<sub>ASCII</sub></b>
Tiefe	6/7 Byte / 48/56 Bit
I/O	lesen / schreiben
Reset	Reset:0x00100000_00_0_0_0_0 <sub>H</sub>

Dieser FSP dient als Kommando FSP für die Altera Remote Update Funktion

#### Imagetyp lesen

Bit[4] = ,0' (Read)

Bit[8] = ,1' (steigende Flanke startet lesen des Imagetyps)

FSP046[1..0] enthält nun den aktuellen Imagetyp.

#### Imagetyp wechseln

Bit[4] = ,1' (Write)

Bit[12] = ,1' (steigende Flanke wechselt das Image)

Mit Einführung des CycloneV änderte sich dieses FSP inhaltlich leicht.

Die Startadresse wanderte um 4 Bits nach links (beginnt nicht mehr bei Bit 20, sondern erst bei Bit 24) und wird um 4 weitere Bits ergänzt (hat also nun die Breite 32 Bits). (18.12.19 – DS)

#### Altes Format

[47..44] n.u.

[43..20] Flash Start Address (ab dieser Adresse wird das Image geschrieben)

[19..17] n.u.

#### Neues Format

[55..24] Flash Start Address (ab dieser Adresse wird das Image geschrieben)

[23..17] n.u.

#### Gemeinsam unverändert

[16] Reset WD Disable (only for debug)

[15..13] n.u.

[12] Start Write (steigende Flanke an diesem Bit startet die FSM zum Imagetyp-Wechsel)

[11..9] n.u.

[8] Start Read (steigende Flanke an diesem Bit startet die FSM zum lesen des Image-Type)

[7..5] n.u.

[4] Read\_n\_Write\_Enable (muss ,0' sein damit ,Start Read' überhaupt ausgeführt wird, muss ,1' sein damit ,Start Write' überhaupt ausgeführt wird)

[3..2] n.u.

[1..0] Read Source

Name	<b>FSP046_AlteraRemoteUpdateStatus</b>
Adresse	<b>0x2E<sub>H</sub>/46<sub>D</sub>/0x3245<sub>ASCII</sub></b>
Tiefe	10 Byte / 80 Bit
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) <sub>H</sub>

Dieser FSP dient als Status FSP für die Altera Remote Update Funktion

- [79..72] ReconfTriggerCondition
- [71..69] Force Osc\_int n.u.
- [68] Force Osc\_int
- [67..44] Boot Address
- [43..41] Watchdog Enable n.u.
- [40] Watchdog Enable
- [39..8] Watchdog timeout
- [7..5] Cd\_early n.u.
- [4] Cd\_early, wenn ,1' ist ein gültiges Application-Image an der Bootadresse zu finden
- [3..2] MSM State n.u.
- [1..0] MSM State ('00' = Factory Image, '11' = Application Image)



Name	<b>FSP050_ModuleSupplyValues</b>
Adresse	<b>0x32<sub>H</sub>/50<sub>D</sub>/0x3332<sub>ASCII</sub></b>
Tiefe	16 Byte / 128 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Liefert die vorzeichenbehafteten Betriebsspannungen des Moduls. Immer 2 Byte stehen für eine Spannung. Bei 8 Spannungen ist dieses FSP 16 Byte tief Die Spannungen sind dabei wie folgt sortiert.

- [127..112] vorzeichenbehaftete VREF, 2,75 Volt der 13 Bit ADCs (13 Bit)
- [111..96] vorzeichenbehaftete -12 Volt (13 Bit)
- [95..80] vorzeichenbehaftete 12 Volt (13 Bit)
- [79..64] vorzeichenbehaftete 5 Volt Analog (13 Bit)
- [63..48] vorzeichenbehaftete 5 Volt Digital (13 Bit)
- [47..32] vorzeichenbehaftete 3,3 Volt (13 Bit)
- [31..16] vorzeichenbehaftete 2,5 Volt (13 Bit)
- [15..0] vorzeichenbehaftete 1,2 Volt (13 Bit)

Name	<b>FSP051_ModulePotiValues</b>
Adresse	<b>0x33<sub>H</sub>/51<sub>D</sub>/0x3333<sub>ASCII</sub></b>
Tiefe	8 Byte / 64 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Liefert die vorzeichenbehafteten Poti-Spannungen des Moduls. Immer 2 Byte stehen für eine Spannung. Bei 4 Spannungen ist dieses FSP 8 Byte tief. Die Spannungen sind dabei wie folgt sortiert.

- [63..48]      vorzeichenbehaftete Poti-Spannung Komparator 4 (13 Bit)
- [47..32]      vorzeichenbehaftete Poti-Spannung Komparator 3 (13 Bit)
- [31..16]      vorzeichenbehaftete Poti-Spannung Komparator 2 (13 Bit)
- [15..0]       vorzeichenbehaftete Poti-Spannung Komparator 1 (13 Bit)

Name	<b>FSP052_ModuleComparatorValues</b>
Adresse	<b>0x34<sub>H</sub>/52<sub>D</sub>/0x3334<sub>ASCII</sub></b>
Tiefe	8 Byte / 64 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Liefert die vorzeichenbehafteten Komparatorspannungen des Moduls. Immer 2 Byte stehen für eine Spannung. Bei 4 Spannungen ist dieses FSP 8 Byte tief. Die Spannungen sind dabei wie folgt sortiert.

- [63..48]      vorzeichenbehaftete Spannung Komparator 4
- [47..32]      vorzeichenbehaftete Spannung Komparator 3
- [31..16]      vorzeichenbehaftete Spannung Komparator 2
- [15..0]       vorzeichenbehaftete Spannung Komparator 1

Name	<b>FSP053_ModuleTemperatures</b>
Adresse	<b>0x35<sub>H</sub>/53<sub>D</sub>/0x3335<sub>ASCII</sub></b>
Tiefe	7 Byte / 56 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Liefert verschiedene Temperaturen des Moduls.

3 Sensoren vom Typ TC74 liefern Statustemperaturen des Moduls mit 8 Bit Auflösung (inkl. Vorzeichen). Diese Sensoren können für Alarmzwecke verwendet werden. Die Alarmschwellen sind zur Laufzeit änderbar und im Standard FSP 054 hinterlegt.

1 Sensor vom Typ ADT7410 liefert die Temperatur im ADC Blechgehäuse mit 16 Bit Auflösung (inkl. Vorzeichen)

[55..48] TC74\_Control\_Alarm\_Info

[7..0]	Bedeutung
0x00	n.u., Resetzustand
0x01	Device 1 nicht bereit
0x02	Device 2 nicht bereit
0x04	Device 3 nicht bereit
0x08	DeviceSearchRunDone
0x10	Device 1 Grenze überschritten
0x20	Device 2 Grenze überschritten
0x40	Device 3 Grenze überschritten
0x80	Alarm Interrupt wenn Grenze bei einem Device überschritten

[47..40] TC74 Sensor 3: Temperatur Modul Mitte (8 Bit inkl. Vorzeichen)

[39..32] TC74 Sensor 2: Temperatur FPGA (8 Bit inkl. Vorzeichen)

[31..24] TC74 Sensor 1: Temperatur DC-DC Wandler (8 Bit inkl. Vorzeichen)

[23..16] ADT7410\_Status\_Info

[7..0]	Bedeutung
0x00	n.u., Resetzustand
0x01	Device 1 liefert Daten
0x02	Device 2 liefert Daten (hier ohne Funktion, da nur 1 Sensor)
0x04	Device 3 liefert Daten (hier ohne Funktion, da nur 1 Sensor)
0x08	Device 4 liefert Daten (hier ohne Funktion, da nur 1 Sensor)
0x10	n.u.
0x20	n.u.
0x40	n.u.
0x80	Device 1 wird gelesen

[15..0] ADT7410 Sensor: Temperatur am ADC (16 Bit inkl. Vorzeichen)

Name	<b>FSP054_ModuleTemperaturesComparisonThresholds</b>
Adresse	<b>0x36<sub>H</sub>/54<sub>D</sub>/0x3336<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	Lesen / schreiben
Reset	0x46_46_46 <sub>H</sub>

Stellt die vorzeichenbehafteten Vergleichswerte zur Verfügung bei denen die TC74 Temperatursensoren Alarm auslösen sollen, sofern die Temperatur überschritten wurde.

Je Temperatur 2 Byte ASCII, also je 1 Byte Vorzeichen behaftetes HEX Zeichen.

Als Standardwert ist 70° Celsius (70<sub>D</sub> = 46<sub>H</sub>) gewählt.

[23..16]      Sensor 3: Temperatur Modul Mitte (8 Bit)

[15..8]        Sensor 2: Temperatur FPGA (8 Bit)

[7..0]         Sensor 1: Temperatur DC-DC Wandler (8 Bit)

Name	<b>FSP058_ParameterChecksumValue</b>
Adresse	<b>0x3A<sub>H</sub>/58<sub>D</sub>/0x3341<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen / schreiben
Reset	0x000000 <sub>H</sub>

Repräsentiert die Vergleichs-Prüfsumme der vom Module empfangenen Parameter. Dieser Wert dient zum Vergleich der im Modul errechneten Prüfsumme.

Die Modul-Prüfsumme wird dabei aus den empfangenen Datenbytes durch aufaddieren gebildet und abschließend mit dem Eintrag von „FSP058\_ParameterChecksumValue“ verglichen.

[23..0]      Checksumme der Datenübertragung zum ADC.

Die Prüfsumme wird im Modul ChecksumBuilder der Teil von mUISC (modular-USI-control) ist aus den Daten der beschriebenen FSP gebildet und abschließend mit dem Wert dieses FSP verglichen.

Name	<b>FSP059_ParameterChecksumValueCalculated</b>
Adresse	<b>0x3B<sub>H</sub>/59<sub>D</sub>/0x3342<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Repräsentiert die errechnete Prüfsumme innerhalb des Moduls. Gibt die aktuell im Modul errechnete Prüfsumme zurück. Dadurch kann der Fortschritt der Prüfsummenbildung jederzeit verifiziert werden.

[23..0]        errechnete Checksumme der Datenübertragung vom PC, bzw. MFU.

Name	<b>FSP060_ADC_Configuration</b>
Adresse	<b>0x3C<sub>H</sub>/60<sub>D</sub>/0x3343<sub>ASCII</sub></b>
Tiefe	1 Byte / 8 Bit
I/O	Lesen / schreiben
Reset	0x00 <sub>H</sub>

Repräsentiert verschiedene Konfigurationszustände des ADC Moduls.

[7..6] n.u.

[5] ADC\_ExternalConStartSignal\_Enable (USI\_SyncMode), wenn ,1' wird der ADC über den HighSpeedReceiver getriggert, andernfalls läuft er frei. D.h. jedes Mal, wenn vom Host ein neuer HighSpeed Wert am ADC Module ankommt und erfolgreich ausdekodiert wurde, also im Modul zur Verfügung steht, startet eine neue ADC Messung.

[4] Remove Outliers, wenn ,1' werden der größte und der kleinste Wert bei der Mittelwertbildung ausgeblendet. Wenn ,0' werden alle Werte zur Mittelwertbildung (sofern diese nicht abgeschaltet ist) herangezogen.

[3..0] ADC Mittelwertbildung

[3..0]	Mittelwertbildung
0x0	Mittelwertbildung aus
0x1	2fach Mittelung
0x2	4fach Mittelung
0x3	8fach Mittelung
0x4	16fach Mittelung



Name	<b>FSP061_ ADCCalibrationGAIN</b>
Adresse	<b>0x3D<sub>H</sub>/61<sub>D</sub>/0x3344<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen / schreiben
Reset	0x4010E4 <sub>H</sub>

Wird als Hilfs-FSP zur Kalibrierung des ADC Moduls verwendet. Enthält die Verstärkungskorrektur (Skalierungsfaktor) des ADC, der über PowerConfigAdvanced beim Kalibriervorgang ermittelt und dann nicht flüchtig im ADC Modul gespeichert wird. Nach dem Einschalten der Spannungsversorgung wird dieses FSP im Falle einer durchgeführten Kalibrierung mit dem gespeicherten Kalibrierwert geladen. Der Resetwert wird dabei überschrieben.

[23..0] Vorzeichenbehafteter 18 Bit Skalierungswert zur Korrektur der ADC Verstärkung

Name	<b>FSP062_ADCCalibrationOFFSET</b>
Adresse	<b>0x3E<sub>H</sub>/62<sub>D</sub>/0x3345<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen / schreiben
Reset	0x000EC9 <sub>H</sub>

Wird als Hilfs-FSP zur Kalibrierung des ADC Moduls verwendet. Enthält die Offsetkorrektur des ADC, der über PowerConfigAdvanced beim Kalibriervorgang ermittelt und dann nicht flüchtig im ADC Moduls gespeichert wird. Nach dem Einschalten der Spannungsversorgung wird dieses FSP im Falle einer durchgeführten Kalibrierung mit dem gespeicherten Kalibrierwert geladen. Der Resetwert wird dabei überschrieben.

[23..0]      Vorzeichenbehaftete 18 Bit Offsetkorrektur des ADC

Name	<b>FSP063_ADCFAVResult</b>
Adresse	<b>0x3F<sub>H</sub>/63<sub>D</sub>/0x3346<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Gibt den gleitenden Mittelwert des ADC zurück (FAV: Floating Average Value). Der ADC läuft frei und erfasst permanent Werte. Diese werden bis zu 16fach gemittelt (siehe: „FSP060\_ADC\_Configuration“)

[23..0]      Vorzeichenbehafteter ADC Wert (20 Bit), bis zu 16fach gemittelt

Name	<b>FSP073_Controller_Limits</b>
Adresse	<b>0x49<sub>H</sub>/73<sub>D</sub>/0x3439<sub>ASCII</sub></b>
Tiefe	6 Byte / 48 Bit
I/O	lesen / schreiben
Reset	0x000000_000000 <sub>H</sub>

Über diesen FSP können die Bereichsgrenzen des PI Reglers festgelegt werden. Der Regler dient der Temperaturreglung des ADCs.

[47..24]      Controller\_MaxVal (20 Bit), repräsentiert den oberen (maximalen) Grenzwert des PI Regler

[23..0]       Controller\_MinVal (20 Bit), repräsentiert den unteren (minimalen) Grenzwert des PI Regler

Name	<b>FSP074_Controller_PI_Settings</b>
Adresse	<b>0x4A<sub>H</sub>/60<sub>D</sub>/0x3441<sub>ASCII</sub></b>
Tiefe	13 Byte / 104 Bit
I/O	lesen / schreiben
Reset	0x00_00000000_00000000_00000000 <sub>H</sub>

Über diesen FSP können die I und P1, P2 Anteile des PI Reglers gesetzt werden. Der Regler dient der Temperaturregelung des ADCs.

[103..97] n.u

[98] Wenn ,1' ist der P2 Anteil des Reglers aktiv.

[97] Wenn ,1' wird der Komparator zur Deaktivierung des I-Anteils aktiviert.  
Die Schwellwerte für diesen Komparator stellt  
„FSP075\_Controller\_I\_Part\_ComparatorLimits“ zur Verfügung.

[96] Wenn ,1' wird der I-Anteil des Reglers um den Faktor 1000 verlangsamt.

[95..64] Controller1\_I\_Part (32 Bit), repräsentiert den I Anteil des PI Reglers.

[63..32] Controller1\_P2\_Part (32 Bit), repräsentiert den 2. P Anteil des PI Reglers.

[31..0] Controller1\_P1\_Part (32 Bit), repräsentiert den 1. P Anteil des PI Reglers.

Name	<b>FSP075_Controller_I_Part_ComparatorLimits</b>
Adresse	<b>0x4B<sub>H</sub>/75<sub>D</sub>/0x3442<sub>ASCII</sub></b>
Tiefe	6 Byte / 48 Bit
I/O	lesen / schreiben
Reset	0x000000_000000 <sub>H</sub>

Repräsentiert die Bereichsgrenzen in denen der I Anteil bei der Regelung berücksichtigt werden soll. Der Regler dient der Temperaturregelung des ADCs.

[47..24]      Controller\_I\_Part\_ComparatorOFFThreshold (20 Bit)

[23..0]      Controller\_I\_Part\_ComparatorONThreshold (20 Bit)

Name	<b>FSP076_Controller_SetValueDeviation</b>
Adresse	<b>0x4C<sub>H</sub>/76<sub>D</sub>/0x3443<sub>ASCII</sub></b>
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Über diesen FSP kann die Regeldifferenz (Delta I) des PI Reglers gelesen werden. Der Regler dient der Temperaturregelung des ADCs. Die Regeldifferenz wird dabei im Differenzbildner des Reglers aus dem Minuend SetValue und dem Subtrahend ActualValue gebildet

[23..0]      SetValueDeviation (20 Bit)

Name	<b>FSP077_Controller_PI_Output</b>
Adresse	<b>0x4D<sub>H</sub>/77<sub>D</sub>/0x3444<sub>ASCII</sub></b>
Tiefe	9 Byte / 72 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) <sub>H</sub>

Über diesen FSP kann der P, I und PI-Anteil des PI Regler gelesen werden. Der Regler dient der Temperaturreglung des ADCs.

[71..48]      Controller\_P\_Output (20 Bit)

[47..24]      Controller\_I\_Output (20 Bit)

[23..0]      Controller\_PI\_Output (20 Bit)