



Allgemeingültige FSPs

Version vom: Freitag, 17. März 2023, 14:55:00

Inhaltsverzeichnis

1.	Änderungsliste	1
2.	Allgemeingültige FSPs	2
	FSP001_ModuleStatus	3
	0x01 _H /1 _D /0x3031 _{ASCII}	
	FSP002_ModuleWarnings	4
	0x02 _H /2 _D /0x3032 _{ASCII}	
	FSP003_ModuleErrors	5
	0x03 _H /3 _D /0x3033 _{ASCII}	
	FSP004_ModuleInterlocks	6
	0x04 _H /4 _D /0x3034 _{ASCII}	
	FSP005_InterlocksEnable	7
	0x05 _H /5 _D /0x3035 _{ASCII}	
	FSP006_InterlocksArrivalSequence	8
	0x06 _H /6 _D /0x3036 _{AS}	
	FSP007_Reserved	9
	0x07 _H /7 _D /0x3037 _{ASCII}	
	FSP008_Reserved	10
	0x08 _H /8 _D /0x3038 _{ASCII}	
	FSP009_ModuleSerialNumber	11
	0x09 _H /9 _D /0x3039 _{ASCII}	
	FSP010_ModuleCommands	12
	0x0A _H /10 _D /0x3041 _{ASCII}	
	FSP011_ModuleInterlocksMask_n	13
	0x0B _H /11 _D /0x3042 _{ASCII}	
	FSP012_ModuleUSIConfig	14
	0x0C _H /12 _D /0x3043 _{ASCII}	
	FSP013_PeripheralConfig	15
	0x0D _H /13 _D /0x3044 _{ASCII}	
	FSP014_CurrentScale	16
	0x0E _H /14 _D /0x3045 _{ASCII}	
	FSP015_VoltageScale	17
	0x0F _H /15 _D /0x3046 _{ASCII}	
	FSP016_BFieldScale	18
	0x10 _H /16 _D /0x3130 _{ASCII}	
	FSP017_DataStorageStatus	19
	0x11 _H /17 _D /0x3131 _{ASCII}	
	FSP018_DataStorageCfg	20
	0x12 _H /18 _D /0x3132 _{ASCII}	
	FSP019_Reserved	21
	0x13 _H /19 _D /0x3133 _{ASCII}	
	FSP020_ActualValue_A	22
	0x14 _H /20 _D /0x3134 _{ASCII}	
	FSP021_ActualValue_B	23
	0x15 _H /21 _D /0x3231 _{ASCII}	
	FSP022_ActualValue_C	24
	0x16 _H /22 _D /0x3232 _{ASCII}	
	FSP023_ActualValue_D	25
	0x17 _H /23 _D /0x3137 _{ASCII}	
	FSP024_Reserved	26
	0x18 _H /24 _D /0x3138 _{ASCII}	
	FSP025_Reserved	27
	0x19 _H /25 _D /0x3139 _{ASCII}	
	FSP026_Reserved	28
	0x1A _H /26 _D /0x3141 _{ASCII}	
	FSP027_Reserved	29
	0x1B _H /27 _D /0x3142 _{ASCII}	
	FSP028_Reserved	30
	0x1C _H /28 _D /0x3143 _{ASCII}	
	FSP029_ActualValuePhysicalQuantities	31
	0x1D _H /39 _D /0x3144 _{ASCII}	
	FSP030_SetValue_A	32
	0x1E _H /30 _D /0x3145 _{ASCII}	
	FSP031_SetValue_B	33
	0x1F _H /31 _D /0x3246 _{ASCII}	
	FSP032_SetValue_C	34

0x20 _H /32 _D /0x3230 _{ASCII}	
FSP033_SetValue_D	35
0x21 _H /33 _D /0x3231 _{ASCII}	
FSP034_Reserved	36
0x22 _H /34 _D /0x3232 _{ASCII}	
FSP035_Reserved	37
0x23 _H /35 _D /0x3233 _{ASCII}	
FSP036_Reserved	38
0x24 _H /36 _D /0x3234 _{ASCII}	
FSP037_Reserved	39
0x25 _H /37 _D /0x3235 _{ASCII}	
FSP038_Reserved	40
0x26 _H /38 _D /0x3236 _{ASCII}	
FSP039_SetValuePhysicalQuantities	41
0x27 _H /39 _D /0x3237 _{ASCII}	
FSP040_RemoteUpdateStatus	42
0x28 _H /40 _D /0x3238 _{ASCII}	
FSP041_RemoteUpdateCommands	43
0x29 _H /41 _D /0x3239 _{ASCII}	
FSP042_RemoteUpdateData	44
0x2A _H /42 _D /0x3241 _{ASCII}	
FSP043_FSP_Image_Generator	46
0x2B _H /43 _D /0x3242 _{ASCII}	
FSP044_FSP_Image_Status	47
0x2C _H /44 _D /0x3243 _{ASCII}	
FSP045_AlteraRemoteUpdateCmd	48
0x2D _H /45 _D /0x3244 _{ASCII}	
FSP046_AlteraRemoteUpdateStatus	49
0x2E _H /46 _D /0x3245 _{ASCII}	
FSP047_Reserved	50
0x2F _H /47 _D /0x3246 _{ASCII}	
FSP048_Reserved	51
0x30 _H /48 _D /0x3330 _{ASCII}	
FSP049_Reserved	52
0x31 _H /49 _D /0x3331 _{ASCII}	
FSP050_ModuleSupplyValues	53
0x32 _H /50 _D /0x3332 _{ASCII}	
FSP051_ModulePotiValues	54
0x33 _H /51 _D /0x3333 _{ASCII}	
FSP052_ModuleComparatorValues	55
0x34 _H /52 _D /0x3334 _{ASCII}	
FSP053_ModuleTemperatures	56
0x35 _H /53 _D /0x3335 _{ASCII}	
FSP054_ModuleTemperaturesComparationThresholds	57
0x36 _H /54 _D /0x3336 _{ASCII}	
FSP055_WaterFlowMeassurement	58
0x37 _H /55 _D /0x3337 _{ASCII}	
FSP056_Reserved	59
0x38 _H /56 _D /0x3338 _{ASCII}	
FSP057_Reserved	60
0x39 _H /57 _D /0x3339 _{ASCII}	
FSP058_ParameterChecksumValue	61
0x3A _H /58 _D /0x3341 _{ASCII}	
FSP059_ParameterChecksumValueCalculated	62
0x3B _H /59 _D /0x3342 _{ASCII}	

1. Änderungsliste

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

2. Allgemeingültige FSPs

Die FSPs 0 bis 59 sind in ihrer Funktion weitestgehend vereinheitlicht und sofern notwendig in die Module zu integrieren. Ab FSP 60 sind diese modulspezifisch und deshalb auch in den einzelnen Modulkapiteln beschrieben.

Name	FSP001_ModuleStatus
Adresse	0x01_H/1_D/0x3031_{ASCII}
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) _H

- [23..9] Module Status,
Kommando Status
z.B. Infos über die Ablaufsteuerungszustand oder Kommandos des Moduls
- [8] USIsHighSpeed
wenn ,1' ist USI im Highspeed Mode
- [7..6] Reserviert für zukünftige Anwendungen
- [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.
- [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.
- [2] ModuleReady
wenn ,1' ist Modul betriebsbereit
Das Modul ist voll betriebsbereit
- [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.

Name	FSP002_ModuleWarnings
Adresse	0x02_H/2_D/0x3032_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) _H

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] des FSP001_ModuleStatus = ,0'.

Name	FSP003_ModuleErrors
Adresse	0x03_H/3_D/0x3033_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) _H

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] des FSP001_ModuleStatus = ,0'.

Name	FSP004_ModuleInterlocks
Adresse	0x04_H/4_D/0x3034_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) _H

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

WICHTIG

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] des FSP001_ModuleStatus = ,0'.

Nicht genutzte Interlockbits müssen ,1' sein!

Definitionen für:

Name	FSP005_InterlocksEnable
Adresse	0x05_H/5_D/0x3035_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) _H

Im FSP sind alle Interlock Enable Informationen des Moduls abgebildet sofern darüber Informationen zur Verfügung gestellt werden. I.d.R. werden auf Modulen z.B. Jumperstellungen bzgl. der Zulässigkeit von Interlocks abgefragt und deren Einstellungen hier abgelegt.

Ist ein Interlock aktiviert (also zugelassen) ist das korrespondierende Bit in diesem FSP gesetzt, nicht zugelassene, also dauerhaft gesperrte Interlocks werden durch eine ,0' dargestellt.

Interlockdefinitionen für:

Name	FSP006_InterlocksArrivalSequence
Adresse	0x06_H/6_D/0x3036_{ASCII}
Tiefe	Modulabhängig
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) _H

Dieser FSP liefert die zeitliche Abfolge auftretender Interlocks. Liegen mehrere Interlocks an, kann über diesen FSP die zeitliche, sequentielle 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 kann die Tiefe aller im Modul erfassbaren Interlocks besitzen, oder aber nur eine bestimmte Anzahl sequenziell auftretender Interlocks erfassen.

I.d.R. wird die Erfassung der 5 zuerst auftretenden Interlocks ausreichend sein.

Name	FSP007_Reserved
Adresse	0x07_H/7_D/0x3037_{ASCII}
Tiefe	-
I/O	-
Reset	Reset:0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP008_Reserved
Adresse	0x08_H/8_D/0x3038_{ASCII}
Tiefe	-
I/O	-
Reset	Reset:0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP009_ModuleSerialNumber
Adresse	0x09_H/9_D/0x3039_{ASCII}
Tiefe	6 Byte / 48 Bit
I/O	lesen
Reset	Reset:0x(siehe Beschreibung) _H

Der FSP enthält die Modul Serien Nummer

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

Name	FSP010_ModuleCommands
Adresse	0x0A_H/10_D/0x3041_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	Reset:0x(siehe Beschreibung) _H

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

[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

Name	FSP011_ModuleInterlocksMask_n
Adresse	0x0B_H/11_D/0x3042_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	Reset:0x(siehe Beschreibung) _H

Dieser FSP enthält i.d.R. 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 normalerweise in der Mitte des FSP (Ausnahme siehe weiter unten im Text), 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.

Sonderfunktion: **Zugelassene Interlocks nach der Reglerfreigabe**

Dieses FSP kann eine weitere Bitmaske enthalten, die sich noch oberhalb der Masken für aktuell anstehende und gespeicherte Interlocks befindet

Das FSP enthält dann drei Bitmasken, die zum einen nicht verwendete Interlocks vollständig ausmaskiert, d.h. alle nicht zu benutzenden Interlocks sind mit ,1' zu setzen. Dies wird bereits weiter oben beschrieben. Zum anderen lassen sich Interlocks mit diesem FSP und der zusätzlichen Maske so maskieren, dass diese erst nach Freigabe des Reglers aktiviert werden. Deren Erfassung wird also während einer Reglersperre ignoriert. Hierbei ist das zugehörige Bit für verzögerte Freigabe des Interlocks (zusammen mit der Reglerfreigabe) mit ,1' zu setzen.

Die Verteilung der Masken ist dann wie folgt:

[Maske verzögerte Freigabe][Maske gespeicherte Interlocks][Maske aktuelle Interlocks]

Name	FSP012_ModuleUSIConfig
Adresse	0x0C_H/12_D/0x3043_{ASCII}
Tiefe	1 Byte / 8 Bit
I/O	lesen / schreiben
Reset	Reset:0x(siehe Beschreibung) _H

Hiermit wird die USI Schnittstelle des Moduls konfiguriert.

- [7] USIsHighSpeed
wenn ,1' schaltet USI in HighSpeed Modus um
- [6..3] n.u.
- [2..0] Bitrate des Moduls [111 = 115k]

Name	FSP013_PeripheralConfig
Adresse	0x0D_H/13_D/0x3044_{ASCII}
Tiefe	1 Byte / 8 Bit
I/O	lesen / schreiben
Reset	Reset:0x(siehe Beschreibung) _H

Auf dem Modul befindliche Peripherie kann mit diesem FSP konfiguriert werden

Name	FSP014_CurrentScale
Adresse	0x0E_H/14_D/0x3045_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	Reset:0x(siehe Beschreibung) _H

Repräsentiert die Stromskalierung des ACU Systems und des darin zur Anwendung kommenden DCCT (10V = CurrentScale Ampere).

Name	FSP015_VoltageScale
Adresse	0x0F_H/15_D/0x3046_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	Reset:0x(siehe Beschreibung) _H

Repräsentiert die Spannungsskalierung des ACU.

Name	FSP016_BFieldScale
Adresse	0x10_H/16_D/0x3130_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	Reset:0x(siehe Beschreibung) _H

Repräsentiert die Feldskalierung des ACU Systems.

Name	FSP017_DataStorageStatus
Adresse	0x11_H/17_D/0x3131_{ASCII}
Tiefe	2 Byte / 16 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Enthält den DataStorage Status

Name	FSP018_DataStorageCfg
Adresse	0x12_H/18_D/0x3132_{ASCII}
Tiefe	6 Byte / 48 Bit
I/O	Lesen /schreiben
Reset	0x(siehe Beschreibung) _H

Enthält die DataStorage Konfiguration

Name	FSP019_Reserved
Adresse	0x13_H/19_D/0x3133_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP020_ActualValue_A
Adresse	0x14_H/20_D/0x3134_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Dieser FSP liefert einen im Modul vorhandenen Istwert_A zurück

Name	FSP021_ActualValue_B
Adresse	0x15_H/21_D/0x3231_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Dieser FSP liefert einen im Modul vorhandenen Istwert_B zurück

Name	FSP022_ActualValue_C
Adresse	0x16_H/22_D/0x3232_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Dieser FSP liefert einen im Modul vorhandenen Istwert_C zurück

Name	FSP023_ActualValue_D
Adresse	0x17_H/23_D/0x3137_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Dieser FSP liefert einen im Modul vorhandenen Istwert_D zurück

Name	FSP024_Reserved
Adresse	0x18_H/24_D/0x3138_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP025_Reserved
Adresse	0x19_H/25_D/0x3139_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP026_Reserved
Adresse	0x1A_H/26_D/0x3141_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP027_Reserved
Adresse	0x1B_H/27_D/0x3142_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP028_Reserved
Adresse	0x1C_H/28_D/0x3143_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP029_ActualValuePhysicalQuantities
Adresse	0x1D_H/39_D/0x3144_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Gibt die physikalische Größe der Istwerte an.

Name	FSP030_SetValue_A
Adresse	0x1E_H/30_D/0x3145_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

Dieser FSP dient zur Übermittlung eines Sollwertes_A an das Modul

Name	FSP031_SetValue_B
Adresse	0x1F_H/31_D/0x3246_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

Dieser FSP dient zur Übermittlung eines Sollwertes_B an das Modul

Name	FSP032_SetValue_C
Adresse	0x20_H/32_D/0x3230_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

Dieser FSP dient zur Übermittlung eines Sollwertes_C an das Modul

Name	FSP033_SetValue_D
Adresse	0x21_H/33_D/0x3231_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

Dieser FSP dient zur Übermittlung eines Sollwertes_D an das Modul

Name	FSP034_Reserved
Adresse	0x22_H/34_D/0x3232_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP035_Reserved
Adresse	0x23_H/35_D/0x3233_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP036_Reserved
Adresse	0x24_H/36_D/0x3234_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP037_Reserved
Adresse	0x25_H/37_D/0x3235_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP038_Reserved
Adresse	0x26_H/38_D/0x3236_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP039_SetValuePhysicalQuantities
Adresse	0x27_H/39_D/0x3237_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

Gibt die physikalische Größe der Sollwerte an.

Name	FSP040_RemoteUpdateStatus
Adresse	0x28_H/40_D/0x3238_{ASCII}
Tiefe	1 Byte / 8 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Wenn ein Modul Remoteupdate unterstützt, ist dieser FSP für das Rücklesen der Statusinformationen des Fernupdates zu benutzen.

[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	FSP041_RemoteUpdateCommands
Adresse	0x29_H/41_D/0x3239_{ASCII}
Tiefe	1 Byte / 8 Bit
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

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	FSP042_RemoteUpdateData
Adresse	0x2A_H/42_D/0x3241_{ASCII}
Tiefe	256 Byte
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

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 nachgeschaltetes serielles 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 ASCII 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	FSP043_FSP_Image_Generator
Adresse	0x2B_H/43_D/0x3242_{ASCII}
Tiefe	3 Byte / 24 Bit
I/O	lesen / schreiben
Reset	0x00_0000 _H

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	FSP044_FSP_Image_Status
Adresse	0x2C_H/44_D/0x3243_{ASCII}
Tiefe	1 Byte / 8 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Liefert Statusrückmeldungen bzgl. des Sicherungsvorgangs von FSP Daten. Die Sicherung wird über „FSP043_FSP_Image_Generator“ 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 FSP070 werden Zugriffe auf das ext. Flash gestartet ist das Ergebnis undefiniert.

Name	FSP045_AlteraRemoteUpdateCmd
Adresse	0x2D_H/45_D/0x3244_{ASCII}
Tiefe	6/7 Byte / 48/56 Bit
I/O	lesen / schreiben
Reset	Reset:0x00100000_00_0_0_0_0 _H

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	FSP046_AlteraRemoteUpdateStatus
Adresse	0x2E_H/46_D/0x3245_{ASCII}
Tiefe	10 Byte / 80 Bit
I/O	lesen
Reset	0x(siehe Beschreibung) _H

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]	Wachdog Enable n.u.
[40]	Wachdog Enable
[39..8]	Wachdog 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	FSP047_Reserved
Adresse	0x2F_H/47_D/0x3246_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP048_Reserved
Adresse	0x30_H/48_D/0x3330_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP049_Reserved
Adresse	0x31_H/49_D/0x3331_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP050_ModuleSupplyValues
Adresse	0x32_H/50_D/0x3332_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Liefert die Betriebsspannungen des Moduls.

Name	FSP051_ModulePotiValues
Adresse	0x33_H/51_D/0x3333_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Liefert die Potispannungen des Moduls.

Name	FSP052_ModuleComparatorValues
Adresse	0x34_H/52_D/0x3334_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Liefert die Komparatorspannungen des Moduls.

Name	FSP053_ModuleTemperatures
Adresse	0x35_H/53_D/0x3335_{ASCII}
Tiefe	modulabhängig
I/O	lesen
Reset	0x(siehe Beschreibung) _H

Liefert verschiedenen Temperaturen des Moduls.

Sofern die Temperaturschwellen zur Laufzeit geändert werden sollen, werden diese im Standard FSP 054 hinterlegt.

Name	FSP054_ ModuleTemperaturesComparisonThresholds
Adresse	0x36_H/54_D/0x3336_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

Stellt die Vergleichswerte (Schwellen) für verschiedene Temperaturen des Moduls zur Verfügung bei denen ein Alarm erfolgen soll. Hiermit lassen sich Standardwerte im Modul zur Laufzeit ändern.

Name	FSP055_WaterFlowMeasurement
Adresse	0x37_H/55_D/0x3337_{ASCII}
Tiefe	modulabhängig
I/O	lesen / schreiben
Reset	0x(siehe Beschreibung) _H

Sind Wasserwächter im Modul vorhanden, werden diese über verschiedene Parameter beschrieben um deren aktuellen Durchfluss zu bestimmen und dazustellen.

FSPxxx_WaterFlowThresholds, wird mit den unteren und oberen Schwellwerten für den Durchfluss in uSekunden zwischen zwei Pulsen beschrieben. Zu beachten ist, dass der untere Grenzwert die minimal zulässige Zeitdauer in uSekunden angibt, die zwischen zwei Pulsen liegen darf, damit aber den maximalen Durchfluss beschreibt. Je kleiner die Zeitspanne zwischen zwei Pulsen ist, desto mehr Wasser fließt durch den Wasserwächter. Umgekehrt beschreibt der obere Schwellwert mit maximal zulässiger Zeitdauer in uSekunden, die zwischen zwei Pulsen liegen darf und damit den minimalen Durchfluss. Je größer die Zeitspanne zwischen zwei Pulsen ist, desto weniger Wasser fließt durch den Wasserwächter.

FSPxxx_InvertedWaterFlow_TimePeriodeBetweenTwoPulses_in_us, gibt die Zeitdauer zwischen zwei Messimpulsen des Wasserwächters in uSekunden zurück.

Diese FSP können beliebig im Modul zu finden sein.

Daher beschreibt FSP055 u. A. deren FSP-Adressen.

- [...] ggf. Index-Typ *) weitere Wasserwächter
- [31..28] ggf. Index-Typ *) des 2. Wasserwächters
- [27..24] Index-Typ *) des 1. Wasserwächters
- [23..16] Anzahl der verfügbaren Wasserwächterkanäle
- [15..8] Adresse des FSPxxx_InvertedWaterFlow_TimePeriodeBetweenTwoPulses_in_us
- [7..0] Adresse des FSPxxx_WaterFlowThresholds

*) Indextypen

0	Float_Type_or_norm_Interlock	(normaler Schaltkontakt, kein Durchfluss übermittelt)
1	Customized_Values	(Benutzerwerte)
2	RRI_010xxx020_2I	(GHM-Hornsberg)
3	RRI_010xxx050_12I	(GHM-Hornsberg)
4	RRI_010xxx070_14I	(GHM-Hornsberg)
5	RRI_025xxx180_36I	(GHM-Hornsberg)
6	RRI_025xxx120_72I	(GHM-Hornsberg)
7	RRI_010xxx160_120I	(GHM-Hornsberg)
8	DRS_9250XXF300	(Kobold)

Name	FSP056_Reserved
Adresse	0x38_H/56_D/0x3338_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP057_Reserved
Adresse	0x39_H/57_D/0x3339_{ASCII}
Tiefe	-
I/O	-
Reset	0x(siehe Beschreibung) _H

Dieser FSP ist für zukünftige Erweiterungen reserviert

Name	FSP058_ParameterChecksumValue
Adresse	0x3A_H/58_D/0x3341_{ASCII}
Tiefe	3 Byte / 24 Bit
I/O	lesen / schreiben
Reset	0x00_0000 _H

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] Prüfsumme der Datenübertragung zum Modul.

Name	FSP059_ParameterChecksumValueCalculated
Adresse	0x3B_H/59_D/0x3342_{ASCII}
Tiefe	3 Byte / 24 Bit
I/O	lesen
Reset	0x00_0000 _H

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 Prüfsumme der Datenübertragung zum Modul