

S1R72V27

テクニカルマニュアル

本資料のご使用につきましては、次の点にご留意願います。

本資料の内容については、予告無く変更することがあります。

1. 本資料の一部、または全部を弊社に無断で転載、または、複製など他の目的に使用することは堅くお断りいたします。
2. 本資料に掲載される応用回路、プログラム、使用方法等はあくまでも参考情報であり、これら起因する第三者の権利（工業所有権を含む）侵害あるいは損害の発生に対し、弊社はいかなる保証を行うものではありません。また、本資料によって第三者または弊社の工業所有権の実施権の許諾を行うものではありません。
3. 特性値の数値の大小は、数直線上の大小関係で表しています。
4. 本資料に掲載されている製品のうち「外国為替及び外国貿易法」に定める戦略物資に該当するものについては、輸出する場合、同法に基づく輸出許可が必要です。
5. 本資料に掲載されている製品は、生命維持装置その他、きわめて高い信頼性が要求される用途を前提としていません。よって、弊社は本（当該）製品をこれらの用途に用いた場合のいかなる責任についても負いかねます。

適用範囲

本ドキュメントは、USB2.0 デバイス・ホストコントローラ LSI「S1R72V27」に適用されます。

目次

1. 機能説明	1
1.1 USBデバイス／ホスト選択	1
1.1.1 USBデバイス／ホスト機能選択方法	1
1.1.2 USBポート状態変化検出ステータス	2
1.1.2.1 USBポート状態変化検出ステータスの使用例	2
1.1.2.1.1 デバイスポート変化ステータス	2
1.1.2.1.2 ホストポート変化ステータス	3
1.1.2.2 信号ライン変化ステータスの使用方法	3
1.2 USBデバイス制御	5
1.2.1 エンドポイント	5
1.2.2 トランザクション	7
1.2.2.1 SETUPトランザクション	9
1.2.2.2 バルク／インタラプトOUTトランザクション	10
1.2.2.3 アイソクロナスOUTトランザクション	11
1.2.2.4 バルク／インタラプトINトランザクション	11
1.2.2.5 アイソクロナスINトランザクション	12
1.2.2.6 PINGトランザクション	13
1.2.3 コントロール転送	14
1.2.3.1 セットアップステージ	15
1.2.3.2 データステージ／ステータスステージ	15
1.2.3.3 自動アドレス設定機能	16
1.2.3.4 デスクリプタ返信機能	16
1.2.4 バルク転送／インタラプト転送／アイソクロナス転送	16
1.2.5 データフロー	17
1.2.5.1 OUT転送	17
1.2.5.2 IN転送	17
1.2.6 バルクオンリーサポート	18
1.2.6.1 CBWサポート	18
1.2.6.2 CSWサポート	19
1.2.7 ケーブルネゴシエーション機能(オート・ネゴシエータ)	20
1.2.7.1 オート・ネゴシエータ	21
1.2.7.1.1 Disable	21
1.2.7.1.2 IDLE	21
1.2.7.1.3 WAIT_TIM3US	21
1.2.7.1.4 WAIT_CHIRP	21

1.2.7.1.5	WAIT_RSTEND	22
1.2.7.1.6	DET_SUSPEND	22
1.2.7.1.7	IN_SUSPEND.....	22
1.2.7.1.8	CHK_EVENT	22
1.2.7.1.9	WAIT_RESTORE.....	22
1.2.7.1.10	ERR	23
1.2.7.2	各バスイベント発生時の動作.....	23
1.2.7.2.1	HSモードにおけるサスペンド検出.....	23
1.2.7.2.2	FSモードにおけるサスペンド検出	25
1.2.7.2.3	HSモードにおけるリセット検出	27
1.2.7.2.4	FSモードにおけるリセット検出.....	28
1.2.7.2.5	HS Detection Handshake	29
1.2.7.2.5.1	FSのホストダウンストリームポートに繋がれた場合	30
1.2.7.2.5.2	HSのホストダウンストリームポートに繋がれた場合.....	32
1.2.7.2.5.3	スリープ中にリセットされた場合	34
1.2.7.2.6	レジュームの発行	36
1.2.7.2.7	レジュームの検出	38
1.2.7.2.8	ケーブル挿入	40
1.3	USBホスト制御	42
1.3.1	チャンネル	42
1.3.1.1	チャンネル概要.....	42
1.3.1.2	コントロール専用チャンネル	44
1.3.1.3	汎用チャンネル.....	45
1.3.1.4	チャンネル使用例	46
1.3.1.4.1	ストレージデバイスを 1 台接続した場合	46
1.3.1.4.2	コミュニケーションデバイスを 1 台接続した場合.....	47
1.3.1.4.3	ヒューマンインタフェイスデバイスを 1 台接続した場合	48
1.3.1.4.4	ハブを介してストレージデバイスを 2 台接続した場合.....	49
1.3.2	スケジューリング	50
1.3.3	トランザクション.....	51
1.3.3.1	SETUPトランザクション.....	52
1.3.3.2	バルクOUTトランザクション	53
1.3.3.3	インタラプトOUTトランザクション.....	54
1.3.3.4	アイソクロナスOUTトランザクション	55
1.3.3.5	バルクINトランザクション	56
1.3.3.6	インタラプトINトランザクション.....	58
1.3.3.7	アイソクロナスINトランザクション	59
1.3.3.8	PINGトランザクション	61

1.3.3.9	low-speed(LS)トランザクション	62
1.3.3.10	スプリットトランザクション	64
1.3.4	コントロール転送	65
1.3.4.1	セットアップステージ	66
1.3.4.2	データステージ／ステータスステージ	66
1.3.4.3	コントロール転送サポート機能	67
1.3.5	バルク転送／インタラプト転送／アイソクロナス転送	70
1.3.6	データフロー	70
1.3.6.1	OUT転送	70
1.3.6.2	IN転送	71
1.3.7	ゼロ長パケット自動発行機能	72
1.3.7.1	バルク／インタラプトOUT転送のゼロ長パケット自動発行機能	72
1.3.8	バルクオンリーサポート機能	73
1.3.9	オーディオクラス・アシスト機能	77
1.3.10	ホストステート管理サポート機能	78
1.3.10.1	ホストステート	78
1.3.10.1.1	IDLE	80
1.3.10.1.2	WAIT_CONNECT	80
1.3.10.1.3	DISABLED	81
1.3.10.1.4	RESET	81
1.3.10.1.5	OPERATIONAL	82
1.3.10.1.6	SUSPEND	83
1.3.10.1.7	RESUME	83
1.3.10.2	検出機能	84
1.3.10.2.1	VBUS異常検出	84
1.3.10.2.2	切断検出	85
1.3.10.2.2.1	HSデバイスが切断された場合	85
1.3.10.2.2.2	FSまたはLSデバイスが切断された場合	87
1.3.10.2.3	リモート・ウェークアップ検出	88
1.3.10.2.3.1	HSデバイスが接続されている場合	88
1.3.10.2.3.2	FSデバイスが接続されている場合	89
1.3.10.2.3.3	LSデバイスが接続されている場合	90
1.3.10.2.4	デバイスチャープ検出機能	91
1.3.10.2.4.1	正しいデバイスのChirpを検出した場合	91
1.3.10.2.4.2	異常なデバイスのChirpを検出した場合	92
1.3.10.2.5	ポートエラー検出	93
1.3.10.3	ホストステート管理サポート機能の単体説明	94
1.3.10.3.1	GoIDLE	94

1.3.10.3.2	GoWAIT_CONNECT	96
1.3.10.3.2.1	FSデバイスが接続された場合	96
1.3.10.3.2.2	LSデバイスが接続された場合	98
1.3.10.3.3	GoDISABLED	100
1.3.10.3.3.1	HSデバイスが接続されている場合	100
1.3.10.3.3.2	FSデバイスが接続されている場合	102
1.3.10.3.3.3	LSデバイスが接続されている場合	103
1.3.10.3.4	GoRESET	104
1.3.10.3.4.1	HSデバイスに対するリセット	104
1.3.10.3.4.2	異常なデバイスのChirp検出	106
1.3.10.3.4.2.1	チャープ完了ディセーブル(H_NegoControl_1.DisChirpFinish)設定が 0 の場合	106
1.3.10.3.4.2.2	チャープ完了ディセーブル(H_NegoControl_1.DisChirpFinish)設定が 1 の場合	108
1.3.10.3.4.3	FSデバイスに対するリセット	111
1.3.10.3.4.4	LSデバイスに対するリセット	113
1.3.10.3.5	GoOPERATIONAL	114
1.3.10.3.6	GoSUSPEND	115
1.3.10.3.6.1	HSデバイスが接続されている場合	115
1.3.10.3.6.2	FSデバイスが接続されている場合	117
1.3.10.3.6.3	LSデバイスが接続されている場合	119
1.3.10.3.7	GoRESUME	121
1.3.10.3.7.1	HSデバイスが接続されている場合	121
1.3.10.3.7.2	FSデバイスが接続されている場合	123
1.3.10.3.7.3	LSデバイスが接続されている場合	125
1.3.10.3.8	GoWAIT_CONNECTtoDIS	127
1.3.10.3.9	GoWAIT_CONNECTtoOP	128
1.3.10.3.9.1	HSデバイスが接続されている場合	128
1.3.10.3.9.2	FSまたはLSデバイスが接続されている場合	130
1.3.10.3.10	GoRESETtoOP	132
1.3.10.3.10.1	HSデバイスが接続されている場合	132
1.3.10.3.10.2	FSまたはLSデバイスが接続されている場合	133
1.3.10.3.11	GoSUSPENDtoOP	134
1.3.10.3.12	GoRESUMETOOP	135
1.4	パワーマネジメント機能	136
1.4.1	SLEEP（スリープ）	136
1.4.2	SNOOZE（スヌーズ）	137

1.4.3	ACTIVE (アクティブ)	137
1.4.4	CPU_Cutモード	138
1.5	FIFO管理	138
1.5.1	FIFOメモリマップ	138
1.5.2	デスクリプタエリア	139
1.5.2.1	デスクリプタエリアへのデータの書き込み	139
1.5.2.2	デスクリプタエリアでのデータステージ(IN)の実行	140
1.5.3	CBWエリア	140
1.5.3.1	CBWエリア(USBデバイス時)	140
1.5.3.2	CBWエリア(USBホスト時)	140
1.5.4	CSWエリア	141
1.5.4.1	CSWエリア(USBデバイス時)	141
1.5.4.2	CSWエリア(USBホスト時)	141
1.5.5	FIFOへのアクセス方法	141
1.5.5.1	RAMアクセスの方法(RAM_Rd)	141
1.5.5.2	RAMアクセスの方法(RAM_WrDoor)	142
1.5.5.3	FIFOアクセスの方法(レジスタアクセス)	142
1.5.5.4	FIFOアクセスの方法(DMA)	142
1.5.5.5	FIFOアクセスの制限	143
1.6	CPUIF	144
1.6.1	CPUIFモード	144
1.6.2	CPUIFのモード設定	145
1.6.3	ブロック構成	146
1.6.3.1	REG(Registers)	146
1.6.3.1.1	同期レジスタアクセス(ライト)	146
1.6.3.1.2	同期レジスタアクセス(リード)	146
1.6.3.1.3	FIFOアクセス(ライト)	146
1.6.3.1.4	FIFOアクセス(リード)	147
1.6.3.1.5	FIFOアクセスの端数処理	148
1.6.3.1.6	RAM_Rdアクセス	150
1.6.3.1.7	非同期レジスタアクセス(ライト)	150
1.6.3.1.8	非同期レジスタアクセス(リード)	150
1.6.3.2	DMA	151
1.6.3.2.1	基本機能	151
1.6.3.2.2	端子設定	153
1.6.3.2.3	カウントモード(ライト)	153
1.6.3.2.4	カウントモード(リード)	155
1.6.3.2.5	フリーランモード(ライト)	156

1.6.3.2.6	フリーランモード(リード).....	157
1.6.3.2.7	REQアサートカウンタオプション(ライト)	157
1.6.3.2.8	REQアサートカウンタオプション(リード)	158
1.6.3.2.9	DMAのFIFOアクセス端数処理	158
1.6.3.2.10	DMAのEdge Sense モード	159
2.	レジスタマップ	160
2.1	レジスタ構成	160
2.2	初期レジスタマップ	160
2.3	レジスタマップ	161
2.3.1	デバイス／ホスト共通レジスタマップ	161
2.3.2	デバイスレジスタマップ	167
2.3.3	ホストレジスタマップ	171
3.	レジスタ詳細説明	177
3.1	デバイス／ホスト共通レジスタ詳細説明	177
3.1.1	000h <i>MainIntStat</i> (<i>Main Interrupt Status</i>)	177
3.1.2	001h <i>USB_DeviceIntStat</i> (<i>USB Device Interrupt Status</i>)	179
3.1.3	002h <i>USB_HostIntStat</i> (<i>USB Host Interrupt Status</i>)	181
3.1.4	003h <i>CPU_IntStat</i> (<i>CPU Interrupt Status</i>)	183
3.1.5	004h <i>FIFO_IntStat</i> (<i>FIFO Interrupt Status</i>)	184
3.1.6	008h <i>MainIntEnb</i> (<i>Main Interrupt Enable</i>)	185
3.1.7	009h <i>USB_DeviceIntEnb</i> (<i>Device Interrupt Enable</i>)	186
3.1.8	00Ah <i>USB_HostIntEnb</i> (<i>Host Interrupt Enable</i>)	187
3.1.9	00Bh <i>CPU_IntEnb</i> (<i>CPU Interrupt Enable</i>)	188
3.1.10	00Ch <i>FIFO_IntEnb</i> (<i>FIFO Interrupt Enable</i>)	189
3.1.11	010h <i>RevisionNum</i> (<i>Revision Number</i>)	190
3.1.12	011h <i>ChipReset</i> (<i>Chip Reset</i>)	191
3.1.13	012h <i>PM_Control</i> (<i>Power Management Control</i>)	192
3.1.14	014h-015h <i>WakeupTim_H,L</i> (<i>Wakeup Time High, Low</i>)	194
3.1.15	016h <i>H_USB_Control</i> (<i>Host USB Control</i>)	195
3.1.16	017h <i>H_XcvtControl</i> (<i>Host Xcvt Control</i>)	196
3.1.17	018h <i>D_USB_Status</i> (<i>Device USB Status</i>)	198
3.1.18	019h <i>H_USB_Status</i> (<i>Host USB Status</i>)	199
3.1.19	01Bh <i>MTM_Config</i> (<i>Multi Transceiver Macro Config</i>)	200
3.1.20	01Fh <i>HostDeviceSel</i> (<i>Host Device Select</i>)	201
3.1.21	020h <i>FIFO_Rd_0</i> (<i>FIFO Read 0</i>)	202
3.1.21	021h <i>FIFO_Rd_1</i> (<i>FIFO Read 1</i>)	202
3.1.23	022h <i>FIFO_Wr_0</i> (<i>FIFO Write 0</i>)	203

3.1.24	023h FIFO_Wr_1(FIFO Write 1)	203
3.1.25	024h-025h FIFO_RdRemain_H,L (FIFO Read Remain High, Low).....	204
3.1.26	026h-027h FIFO_WrRemain_H,L (FIFO Write Remain High, Low).....	205
3.1.27	028h FIFO_ByteRd(FIFO Byte Read).....	206
3.1.28	02Ah FIFO_ByteWr(FIFO Byte Write)	207
3.1.29	02Ch CacheRemain (Host Device Select).....	208
3.1.30	030h-031h RAM_RdAdrs_H,L (RAM Read Address High, Low)	209
3.1.31	032h RAM_RdControl (RAM Read Control)	210
3.1.32	035h RAM_RdCount (RAM Read Counter).....	211
3.1.33	038h-039h RAM_WrAdrs_H,L (RAM Write Address High, Low).....	212
3.1.34	03Ah RAM_WrDoor_0 (RAM Write Door 0)	213
3.1.35	03Bh RAM_WrDoor_1 (RAM Write Door 1)	213
3.1.36	040h RAM_Rd_00 (RAM Read 00)	214
3.1.37	041h RAM_Rd_01 (RAM Read 01)	214
3.1.38	042h RAM_Rd_02 (RAM Read 02)	214
3.1.39	043h RAM_Rd_03 (RAM Read 03)	214
3.1.40	044h RAM_Rd_04 (RAM Read 04)	214
3.1.41	045h RAM_Rd_05 (RAM Read 05)	214
3.1.42	046h RAM_Rd_06 (RAM Read 06)	214
3.1.43	047h RAM_Rd_07 (RAM Read 07)	214
3.1.44	048h RAM_Rd_08 (RAM Read 08)	214
3.1.45	049h RAM_Rd_09 (RAM Read 09)	214
3.1.46	04Ah RAM_Rd_0A (RAM Read 0A)	214
3.1.47	04Bh RAM_Rd_0B (RAM Read 0B).....	214
3.1.48	04Ch RAM_Rd_0C (RAM Read 0C)	214
3.1.49	04Dh RAM_Rd_0D (RAM Read 0D)	214
3.1.50	04Eh RAM_Rd_0E (RAM Read 0E).....	214
3.1.51	04Fh RAM_Rd_0F (RAM Read 0F)	214
3.1.52	050h RAM_Rd_10 (RAM Read 10)	214
3.1.53	051h RAM_Rd_11 (RAM Read 11).....	214
3.1.54	052h RAM_Rd_12 (RAM Read 12)	214
3.1.55	053h RAM_Rd_13 (RAM Read 13)	214
3.1.56	054h RAM_Rd_14 (RAM Read 14)	214
3.1.57	055h RAM_Rd_15 (RAM Read 15)	214
3.1.58	056h RAM_Rd_16 (RAM Read 16)	214
3.1.59	057h RAM_Rd_17 (RAM Read 17)	214
3.1.60	058h RAM_Rd_18 (RAM Read 18)	214
3.1.61	059h RAM_Rd_19 (RAM Read 19)	214

3.1.62	05Ah RAM_Rd_1A (RAM Read 1A)	214
3.1.63	05Bh RAM_Rd_1B (RAM Read 1B)	214
3.1.64	05Ch RAM_Rd_1C (RAM Read 1C)	214
3.1.65	05Dh RAM_Rd_1D (RAM Read 1D)	214
3.1.66	05Eh RAM_Rd_1E (RAM Read 1E)	214
3.1.67	05Fh RAM_Rd_1F (RAM Read 1F)	215
3.1.68	061h DMA_Config (DMA Config)	216
3.1.69	062h DMA_Control (DMA Control)	218
3.1.70	064h-065h DMA_Remain_H,L (DMA FIFO Remain High, Low)	219
3.1.71	067h DMA_EdgeMode (DMA Edge Mode)	220
3.1.72	068h-069h DMA_Count_HH,HL (DMA Transfer Byte Counter High/High, High/Low)	221
3.1.73	06Ah-06Bh DMA_Count_LH,LL (DMA Transfer Byte Counter Low/High, Low/Low)	221
3.1.74	06Ch DMA_RdData_0 (DMA Read Data 0)	223
3.1.75	06Dh DMA_RdData_1 (DMA Read Data 1)	223
3.1.76	06Eh DMA_WrData_0 (DMA Write Data 0)	224
3.1.77	06Fh DMA_WrData_1 (DMA Write Data 1)	224
3.1.78	071h ModeProtect (Mode Protection)	225
3.1.79	073h ClkSelect (Clock Select)	226
3.1.80	075h CPU_Config (CPU Configuration)	227
3.1.81	07E-07Fh CPUIF_MODE (CPUIF Mode)	229
3.1.82	080h-081h AREA0StartAdrs_H,L (AREA 0 Start Address High, Low)	230
3.1.83	084h-085h AREA1StartAdrs_H,L (AREA 1 Start Address High, Low)	230
3.1.84	088h-089h AREA2StartAdrs_H,L (AREA 2 Start Address High, Low)	230
3.1.85	08Ch-08Dh AREA3StartAdrs_H,L (AREA 3 Start Address High, Low)	230
3.1.86	090h-091h AREA4StartAdrs_H,L (AREA 4 Start Address High, Low)	230
3.1.87	094h-095h AREA5StartAdrs_H,L (AREA 5 Start Address High, Low)	230
3.1.88	082h-083h AREA0EndAdrs_H,L (AREA 0 End Address High, Low)	231
3.1.89	086h-087h AREA1EndAdrs_H,L (AREA 1 End Address High, Low)	231
3.1.90	08Ah-08Bh AREA2EndAdrs_H,L (AREA 2 End Address High, Low)	231
3.1.91	08Eh-08Fh AREA3EndAdrs_H,L (AREA 3 End Address High, Low)	231
3.1.92	092h-093h AREA4EndAdrs_H,L (AREA 4 End Address High, Low)	231
3.1.93	096h-097h AREA5EndAdrs_H,L (AREA 5 End Address High,Low)	231
3.1.94	09Fh AREAnFIFO_Clr (AREA n FIFO Clear)	232
3.1.95	0A0h AREA0Join_0 (AREA 0 Join 0)	233
3.1.96	0A2h AREA1Join_0 (AREA 1 Join 0)	233
3.1.97	0A4h AREA2Join_0 (AREA 2 Join 0)	233
3.1.98	0A6h AREA3Join_0 (AREA 3 Join 0)	233
3.1.99	0A8h AREA4Join_0 (AREA 4 Join 0)	233

3.1.100	0AAh AREA5Join_0 (AREA 5 Join 0)	233
3.1.101	0A1h AREA0Join_1 (AREA 0 Join 1)	234
3.1.102	0A3h AREA1Join_1 (AREA 1 Join 1)	234
3.1.103	0A5h AREA2Join_1 (AREA 2 Join 1)	234
3.1.104	0A7h AREA3Join_1 (AREA 3 Join 1)	234
3.1.105	0A9h AREA4Join_1 (AREA 4 Join 1)	234
3.1.106	0ABh AREA5Join_1 (AREA 5 Join 1)	234
3.1.107	0AEh ClrAREAnJoin_0 (Clear AREA n Join 0)	236
3.1.108	0AFh ClrAREAnJoin_1 (Clear AREA n Join 1)	237
3.2	デバイスレジスタ詳細説明	238
3.2.1	0B0h D_SIE_IntStat (Device SIE Interrupt Status)	238
3.2.2	0B3h D_BulkIntStat (Device Bulk Interrupt Status)	240
3.2.3	0B4h D_EPrIntStat (Device EPr Interrupt Status)	241
3.2.4	0B5h D_EP0IntStat (Device EP0 Interrupt Status)	243
3.2.5	0B6h D_EPaIntStat (Device EPa Interrupt Status)	245
3.2.6	0B7h D_EPbIntStat (Device EPb Interrupt Status)	245
3.2.7	0B8h D_EPcIntStat (Device EPc Interrupt Status)	245
3.2.8	0B9h D_EPdIntStat (Device EPd Interrupt Status)	245
3.2.9	0BAh D_EPeIntStat (Device EPe Interrupt Status)	245
3.2.10	0BCh-0BDh D_AlarmIN_IntStat_H,L (Device AlarmIN Interrupt Status High, Low)	247
3.2.11	0BEh-0BFh D_AlarmOUT_IntStat_H,L (Device AlarmOUT Interrupt Status High, Low)	248
3.2.12	0C0h D_SIE_IntEnb (Device SIE Interrupt Enable)	249
3.2.13	0C3h D_BulkIntEnb (Device Bulk Interrupt Enable)	250
3.2.14	0C4h D_EPrIntEnb (Device EPr Interrupt Enable)	251
3.2.15	0C5h D_EP0IntEnb (Device EP0 Interrupt Enable)	252
3.2.16	0C6h D_EPaIntEnb (Device EPa Interrupt Enable)	253
3.2.17	0C7h D_EPbIntEnb (Device EPb Interrupt Enable)	253
3.2.18	0C8h D_EPcIntEnb (Device EPc Interrupt Enable)	253
3.2.19	0C9h D_EPdIntEnb (Device EPd Interrupt Enable)	253
3.2.20	0CAh D_EPeIntEnb (Device EPe Interrupt Enable)	253
3.2.21	0CCh-0CDh D_AlarmIN_IntEnb_H,L (Device AlarmIN Interrupt Enable High, Low)	254
3.2.22	0CEh-0CFh D_AlarmOUT_IntEnb_H,L (Device AlarmOUT Interrupt Enable High, Low)	255
3.2.23	0D0h D_NegoControl (Device Negotiation Control)	256
3.2.24	0D3h D_XcvrControl (Device Xcvr Control)	258
3.2.25	0D4h D_USB_Test (Device USB_Test)	259
3.2.26	0D6h D_EPnControl (Device Endpoint Control)	261
3.2.27	0D8h D_BulkOnlyControl (Device BulkOnly Control)	262

3.2.28	0D9h D_BulkOnlyConfig (Device BulkOnly Configuration).....	263
3.2.29	0E0h D_EP0SETUP_0 (Device EP0 SETUP 0).....	265
3.2.30	0E1h D_EP0SETUP_1 (Device EP0 SETUP 1).....	265
3.2.31	0E2h D_EP0SETUP_2 (Device EP0 SETUP 2).....	265
3.2.32	0E3h D_EP0SETUP_3 (Device EP0 SETUP 3).....	265
3.2.33	0E4h D_EP0SETUP_4 (Device EP0 SETUP 4).....	265
3.2.34	0E5h D_EP0SETUP_5 (Device EP0 SETUP 5).....	265
3.2.35	0E6h D_EP0SETUP_6 (Device EP0 SETUP 6).....	265
3.2.36	0E7h D_EP0SETUP_7 (Device EP0 SETUP 7).....	265
3.2.37	0E8h D_USB_Address (Device USB Address)	266
3.2.38	0EAh D_SETUP_Control(Device SETUP Control).....	267
3.2.39	0EEh-0EFh D_FrameNumber_H,L (Device FrameNumber High, Low).....	268
3.2.40	0F0h D_EP0MaxSize (Device EP0 Max Packet Size)	269
3.2.41	0F1h D_EP0Control (Device EP0 Control).....	270
3.2.42	0F2h D_EP0ControlIN (Device EP0 Control IN).....	271
3.2.43	0F3h D_EP0ControlOUT (Device EP0 Control OUT).....	273
3.2.44	0F8h-0F9h D_EPaMaxSize_H,L (Device EPa Max Packet Size High, Low).....	275
3.2.45	100h-101h D_EPbMaxSize_H,L (Device EPb Max Packet Size High, Low)	275
3.2.46	108h-109h D_EPcMaxSize_H,L (Device EPc Max Packet Size High, Low).....	275
3.2.47	110h-111h D_EPdMaxSize_H,L (Device EPd Max Packet Size High, Low)	275
3.2.48	118h-119h D_EPeMaxSize_H,L (Device EPe Max Packet Size High), Low.....	275
3.2.49	0FAh D_EPaConfig (Device EPa Configuration)	276
3.2.50	102h D_EPbConfig (Devie EPb Configuration)	276
3.2.51	10Ah D_EPcConfig (Device EPc Configuration)	276
3.2.52	112h D_EPdConfig (Devie EPd Configuration)	276
3.2.53	11Ah D_EPeConfig (Device EPe Configuration)	276
3.2.54	0FCh D_EPaControl (Device EPa Control)	278
3.2.55	104h D_EPbControl (Device EPb Control)	278
3.2.56	10Ch D_EPcControl (Device EPc Control).....	278
3.2.57	114h D_EPdControl (Device EPd Control)	278
3.2.58	11Ch D_EPeControl (Device EPc Control).....	278
3.2.59	120h-121h D_DescAdrs_H,L (Device Descriptor Address High, Low)	280
3.2.60	122h-123h D_DescSize_H,L (Device Descriptor Size High, Low).....	281
3.2.61	126h D_EP_DMA_Ctrl (Device EP DMA Control)	282
3.2.62	128h-129h D_EnEP_IN_H,L (Device Enable Endpoint-IN High, Low)	283
3.2.63	12Ah-12Bh D_EnEP_OUT_H,L (Device Enable Endpoint-OUT High, Low).....	284
3.2.64	12Ch-12Dh D_EnEP_IN_ISO_H,L (Device Enable Endpoint-IN Isocronouse High, Low).....	285

3.2.65	12Eh-12Fh D_EnEP_OUT_ISO_H,L (Device Enable Endpoint-OUT Isocronouse High, Low)	286
3.3	ホストレジスタ詳細説明	287
3.3.1	140h H_SIE_IntStat_0 (Host SIE Interrupt Status 0)	287
3.3.2	141h H_SIE_IntStat_1 (SIE Host Interrupt Status 1)	289
3.3.3	143h H_FrameIntStat (Host Frame Interrupt Status)	290
3.3.4	144h H_CHrIntStat (Host CHr Interrupt Status)	291
3.3.5	145h H_CH0IntStat (Host CH0 Interrupt Status)	292
3.3.6	146h H_CHaIntStat (Host CHa Interrupt Status)	294
3.3.7	147h H_CHbIntStat (Host CHb Interrupt Status)	296
3.3.8	148h H_CHcIntStat (Host CHc Interrupt Status)	296
3.3.9	149h H_CHdIntStat (Host CHd Interrupt Status)	296
3.3.10	14Ah H_CHeIntStat (Host CHe Interrupt Status)	296
3.3.11	150h H_SIE_IntEnb_0 (Host SIE Interrupt Enable)	298
3.3.12	151h H_SIE_IntEnb_1(SIE Host Interrupt Enable 1)	299
3.3.13	153h H_FrameIntEnb(Host Frame Interrupt Enable)	300
3.3.14	154h H_CHrIntEnb(Host CHr Interrupt Enable)	301
3.3.15	155h H_CH0IntEnb(Host CH0 Interrupt Enable)	302
3.3.16	156h H_CHaIntEnb (Host CHa Interrupt Enable)	303
3.3.17	157h H_CHbIntEnb (Host CHb Interrupt Enable)	304
3.3.18	158h H_CHcIntEnb (Host CHc Interrupt Enable)	304
3.3.19	159h H_CHdIntEnb (Host CHd Interrupt Enable)	304
3.3.20	15Ah H_CHeIntEnb (Host CHe Interrupt Enable)	304
3.3.21	160h H_NegoControl_0 (Host NegoControl 0)	305
3.3.22	162h H_NegoControl_1 (Host NegoControl 1)	307
3.3.23	164h H_USB_Test (Host USB_Test)	308
3.3.24	170h H_CH0SETUP_0 (Host CH0 SETUP 0)	310
3.3.25	171h H_CH0SETUP_1 (Host CH0 SETUP 1)	310
3.3.26	172h H_CH0SETUP_2 (Host CH0 SETUP 2)	310
3.3.27	173h H_CH0SETUP_3 (Host CH0 SETUP 3)	310
3.3.28	174h H_CH0SETUP_4 (Host CH0 SETUP 4)	310
3.3.29	175h H_CH0SETUP_5 (Host CH0 SETUP 5)	310
3.3.30	176h H_CH0SETUP_6 (Host CH0 SETUP 6)	310
3.3.31	177h H_CH0SETUP_7 (Host CH0 SETUP 7)	310
3.3.32	17Eh-17Fh H_FrameNumber_H,L (Host FrameNumber High, Low)	311
3.3.33	180h H_CH0Config_0(Host Channel 0 Configuration0)	312
3.3.34	181h H_CH0Config_1(Host Channel 0 Configuration1)	314
3.3.35	183h H_CH0MaxPktSize (Host Channel 0 Max Packet Size)	315

3.3.36	186h-187h H_CH0TotalSize_H,L (Host Channel 0 Total Size High, Low)	316
3.3.37	188h H_CH0HubAdrs (Host Channel 0 Hub Address)	317
3.3.38	189h H_CH0FuncAdrs (Host Channel 0 Function Address).....	318
3.3.39	18Bh H_CTL_SupportControl (Host ControlTransfer Support Control)	319
3.3.40	18Eh H_CH0ConditionCode (Host Channel 0 Condition Code)	320
3.3.41	190h H_CHaConfig_0(Host Channel a Configuration0)	321
3.3.42	191h H_CHaConfig_1(Host Channel a Configuration1)	323
3.3.43	192h-193h H_CHaMaxPktSize_H,L (Host Channel a Max Packet Size High, Low)	324
3.3.44	194h-195h H_CHaTotalSize_HH,HL (Host Channel a Total Size High-High, High-Low)	325
3.3.45	196h-197h H_CHaTotalSize_LH,LL (Host Channel a Total Size Low-High, Low-Low)	325
3.3.46	198h H_CHaHubAdrs (Host Channel a Hub Address)	327
3.3.47	199h H_CHaFuncAdrs (Host Channel a Function Address).....	328
3.3.48	19Ah H_CHaBO_SupportCtl (Host CHa Bulk Only Transfer Support Control)	329
3.3.49	19Bh H_CHaBO_CSW_RcvSize (Host CHa Bulk Only Transfer Support CSW Receive Data Size).....	330
3.3.50	19Ch H_CHaBO_OUT_EP_Ctl (Host CHa Bulk Only Transfer Support OUT Endpoint Control)	331
3.3.51	19Dh H_CHaBO_IN_EP_Ctl(Host CHa Bulk Only Transfer Support IN Endpoint Control)	332
3.3.52	19Eh H_CHaConditionCode (Host Channel a Condition Code)	333
3.3.53	1A0h H_CHbConfig_0(Host Channel b Configuration0).....	334
3.3.54	1B0h H_CHcConfig_0(Host Channel c Configuration0)	334
3.3.55	1C0h H_CHdConfig_0(Host Channel d Configuration0)	334
3.3.56	1D0h H_CHeConfig_0(Host Channel e Configuration0)	334
3.3.57	1A1h H_CHbConfig_1(Host Channel b Configuration1).....	336
3.3.58	1B1h H_CHcConfig_1(Host Channel c Configuration1)	336
3.3.59	1C1h H_CHdConfig_1(Host Channel d Configuration1)	336
3.3.60	1D1h H_CHeConfig_1(Host Channel e Configuration1)	336
3.3.61	1A2h-1A3h H_CHbMaxPktSize_H,L (Host Channel b Max Packet Size High, Low)	338
3.3.62	1B2h-1B3h H_CHcMaxPktSize_H,L (Host Channel c Max Packet Size High, Low)	338
3.3.63	1C2h-1C3h H_CHdMaxPktSize_H,L (Host Channel d Max Packet Size High, Low).....	338
3.3.64	1D2h-1D3h H_CHeMaxPktSize_H,L (Host Channel e Max Packet Size High, Low).....	338
3.3.65	1A4h-1A5h H_CHbTotalSize_HH,HL (Host Channel b Total Size High-High, High-Low)	340
3.3.66	1A6h-1A7h H_CHbTotalSize_LH,LL (Host Channel b Total Size Low-High, Low-Low)	340
3.3.67	1B4h-1B5h H_CHcTotalSize_HH,HL (Host Channel c Total Size High-High, High-Low)	340
3.3.68	1B6h-1B7h H_CHcTotalSize_LH,LL (Host Channel c Total Size Low-High, Low-Low).....	340

3.3.69	1C4h-1C5h H_CHdTotalSize_HH,HL (Host Channel d Total Size High-High, High-Low)	340
3.3.70	1C6h-1C7h H_CHdTotalSize_LH,LL (Host Channel d Total Size Low-High, Low-Low)	340
3.3.71	1D4h-1D5h H_CHeTotalSize_HH,HL (Host Channel e Total Size High-High, High-Low)	340
3.3.72	1D6h-1D7h H_CHeTotalSize_LH,LL (Host Channel e Total Size Low-High, Low-Low)	340
3.3.73	1A8h H_CHbHubAdrs (Host Channel b Hub Address)	342
3.3.74	1B8h H_CHcHubAdrs (Host Channel c Hub Address)	342
3.3.75	1C8h H_CHdHubAdrs (Host Channel d Hub Address)	342
3.3.76	1D8h H_CHeHubAdrs (Host Channel e Hub Address)	342
3.3.77	1A9h H_CHbFuncAdrs (Host Channel b Function Address)	343
3.3.78	1B9h H_CHcFuncAdrs (Host Channel c Function Address)	343
3.3.79	1C9h H_CHdFuncAdrs (Host Channel d Function Address)	343
3.3.80	1D9h H_CHeFuncAdrs (Host Channel d Function Address)	343
3.3.81	1AAh-1ABh H_CHbInterval_H,L(Host Channel b Interval High, Low)	344
3.3.82	1BAh-1BBh H_CHcInterval_H,L(Host Channel c Interval High, Low)	344
3.3.83	1CAh-1CBh H_CHdInterval_H,L(Host Channel d Interval High, Low)	344
3.3.84	1DAh-1DBh H_CHeInterval_H,L(Host Channel e Interval High, Low)	344
3.3.85	1ACh H_CHbTranPause(Host Channel b Transaction Pause)	345
3.3.86	1BCh H_CHcTranPause(Host Channel c Transaction Pause)	345
3.3.87	1CCh H_CHdTranPause(Host Channel d Transaction Pause)	345
3.3.88	1DCh H_CHeTranPause(Host Channel e Transaction Pause)	345
3.3.89	1AEh H_CHbConditionCode (Host Channel b Condition Code)	346
3.3.90	1BEh H_CHcConditionCode (Host Channel c Condition Code)	346
3.3.91	1CEh H_CHdConditionCode (Host Channel d Condition Code)	346
3.3.92	1DEh H_CHeConditionCode (Host Channel e Condition Code)	346
Appendix A リトルエンディアンのCPUとの接続		348
Appendix B S1R72V17 とS1R72V27 の相違点		350

1. 機能説明

本 LSI の動作について説明します。

以下では、レジスタについて下記のルールで標記します。

- 1 アドレス分のレジスタを指す場合の名称
レジスタ名+レジスタと標記します。
例：「MainIntStat レジスタ」
- 個々のビットを指す場合の名称
レジスタ名. ビット名+ビット、或いは、ビット名+ビットと標記します。
例：「MainIntStat.CPU_IntStat ビット」
- デバイスエンドポイント毎にあるレジスタ
D_EPx{x=... }～レジスタと標記します。
例：「D_EPx{x=0,a-e}IntStat レジスタ」
- ホストチャネル毎にあるレジスタ
H_CHx{x=... }～レジスタと標記します。
例：「H_CHx{x=0,a-e}IntStat レジスタ」

1.1 USBデバイス／ホスト選択

USB デバイス機能と USB ホスト機能のいずれか一方を選択して下さい。

USB デバイス機能を選択した場合（以下デバイスモード）、共通レジスタおよびデバイスレジスタのレジスタビット及び機能が有効となります。

USB ホスト機能を選択した場合（以下ホストモード）、共通レジスタおよびホストレジスタのレジスタビット及び機能が有効となります。

1.1.1 USBデバイス／ホスト機能選択方法

HostDeviceSel.HOSTxDEVICE ビットをクリアした場合にはデバイス機能が、セットした場合にはホスト機能が使用可能となります。ホストモードとデバイスモードとの相互のモード切り替えは、パワーマネジメントステートが SLEEP である時に行えます。USB デバイス／ホスト選択の設定項目を表 1.1 に示します。

表 1.1 USB デバイス／ホスト機能選択の設定項目

項目	レジスタ／ビット	説明
USB デバイス／ホスト 選択	HostDeviceSel.HOSTxDEVICE	USB デバイスモードとホストモードのいずれかを選択します。選択した側のレジスタ、及び機能が使用可能です。 1: ホストモード 0: デバイスモード

1. 機能説明

1.1.2 USBポート状態変化検出ステータス

本 LSI には、USB ポート状態の検出機能があります。

本機能は SLEEP / ACTIVE（「1.4 パワーマネージメント機能」項を参照）の両状態で使用できます。

1.1.2.1 USBポート状態変化検出ステータスの使用例

デバイスポート変化ステータスおよびホストポート変化ステータスの使用例を示します。

1.1.2.1.1 デバイスポート変化ステータス

本ステータスは、デバイスモードにおいて、VBUS 端子の状態が変化したことを示します。

デバイスポート変化ステータス関連レジスタを表 1.2 に示します。

表 1.2 デバイスポート変化ステータス関連レジスタ

項目	レジスタ/ビット	説明
VBUS 端子変化ステータス	USB_DeviceIntStat.VBUS_Changed	VBUS 端子の状態が変化したことを示します。
VBUS 端子変化ステータスイネーブル	USB_DeviceIntEnb.EnVBUS_Changed	USB_DeviceIntStat.VBUS_Changed による MainIntStat.USB_DeviceIntStat ビットのアサート許可/禁止します。
VBUS 端子状態	D_USB_Status.VBUS	VBUS 端子の状態を示します。

デバイスポート変化ステータスを使用する場合、F/W は(1)、(2)、(4)～(7)の処理を行います。

- (1) VBUS 端子変化ステータスをクリアします。また、MainIntEnb.EnUSB_DeviceIntStat ビットをセットします。(割り込みを使用する場合)
- (2) VBUS 端子変化ステータスイネーブルをセットします。
- (3) VBUS の供給があると、VBUS 端子変化ステータスが発行されます。
- (4) VBUS 端子変化ステータスを確認します。
- (5) VBUS 端子変化ステータスをクリアします。
- (6) VBUS 端子変化ステータスイネーブルをクリアします。
- (7) VBUS 端子状態を確認します。VBUS 端子状態が“1”であれば、VBUS が供給されている（ホストまたはハブが接続されている）状態と判断します。

1.1.2.1.2 ホストポート変化ステータス

本ステータスは、ホストモードにおいて、VBUS 給電を制御しているパワードライバの状態を示します。

ホストポート変化ステータス関連レジスタを表 1.4 に示します。

表 1.3 ホストポート変化ステータス関連レジスタ

項目	レジスタ/ビット	説明
VBUS 異常検出ステータス	USB_HostIntStat.VBUS_Err	VBUS に異常が発生したことを示します。
VBUS 異常検出ステータスイネーブル	USB_HostIntEnb.EnVBUS_Err	USB_HostIntStat.VBUS_Err による MainIntStat.USB_HostIntStat ビットのアサートを許可／禁止します。
VBUS イネーブル	H_USB_Control.VBUS_Enb	外部 USB パワースイッチをイネーブルにします。
ホストポート VBUS 状態	H_USB_Status.VBUS_State	ホストポートの VBUS の状態（正常／異常）を示します。

VBUS 異常検出ステータスを検出した場合には、直ちに VBUS イネーブルを OFF にして VBUS のドライブを停止してください。

1.1.2.2 信号ライン変化ステータスの使用方法

本ステータスは、ホストモードにおいて、DP 端子および DM 端子の状態が変化したことを示します。

信号ライン変化ステータス関連レジスタを表 1.4 に示します。

表 1.4 信号ライン変化ステータス関連レジスタ

項目	レジスタ/ビット	説明
信号ライン変化ステータス	USB_HostIntStat.LineStateChanged	ホストポートの DP 端子および DM 端子の状態が変化したことを示します。
信号ライン変化ステータス割り込みイネーブル	USB_HostIntEnb.EnLineStateChanged	信号ライン変化ステータスによる MainIntStat.USB_HostIntStat ビットのアサートを許可／禁止します。
VBUS 異常検出ステータス	USB_HostIntStat.VBUS_Err	VBUSFLG 端子をモニタし、VBUS 給電に異常が発生したことを示します。
VBUS 異常検出ステータス割り込みイネーブル	USB_HostIntEnb.EnVBUS_Err	VBUS 異常検出ステータスによる MainIntStat.USB_HostIntStat ビットのアサートを許可／禁止します。
オペレーションモード	H_XcvrControl.OpMode[1:0]	トランシーバマクロのオペレーションモードを設定します。
VBUS イネーブル	H_USB_Control.VBUS_Enb	外部 USB パワースイッチをイネーブルにするよう、VBUSEN 端子から信号を出力します。
ホストポート VBUS 状態	H_USB_Status.VBUS_State	ホストポートの VBUSFLG 端子の状態(正常／異常)を示します。
ホストポート信号ライン状態	H_USB_Status.LineState[1:0]	USB ケーブル上の信号状態を示します。
ホストステート遷移実行	H_NegoControl_0.AutoMode[3:0]	遷移させるホストステートを設定します。

1. 機能説明

信号ライン変化ステータスは、USB ホストとして動作中の接続検出ステータスとは異なり、USB ホスト機能を使用していないとき、すなわち、SLEEP ステート(「1.4 パワーマネージメント機能」参照)において、ホストポートの信号ライン変化を検出するための特殊なステータスです。USB ホスト動作時に `USB_HostIntEnb.EnLineStateChanged` がイネーブルになっていると、本ステータスは頻繁にアサートされます。そのため、USB ホストモードでの動作時は `USB_HostIntEnb.EnLineStateChanged` をディセーブルにしてください。

また、USB ホストモードでの動作時以外で `VBUS` 異常検出ステータスを検出した場合には、直ちに `VBUS` イネーブルを OFF して `VBUS` のドライブを停止してください。

信号ライン変化ステータスを使用する際に、ファームウェアは(1)～(4)、(6)～(9)の処理を行います。

- (1) 信号ライン変化ステータスおよび `VBUS` 異常検出ステータスをクリアします。
- (2) 信号ライン変化ステータスイネーブルおよび `VBUS` 異常検出ステータスイネーブルをセットします。また、`MainIntEnb.EnUSB_HostIntStat` ビットをセットします。(割り込みを使用する場合)
- (3) `H_XcvrControl` レジスタに `0x93` を設定します。
- (4) `VBUS` イネーブルを ON します。
- (5) ホストポートにデバイスの接続があるとラインステートが `SE0` から変化し、信号ライン変化ステータスが発行されます。
- (6) 信号ライン変化ステータスを確認します。
- (7) 信号ライン変化ステータスをクリアします。
- (8) 信号ライン変化ステータスイネーブルをクリアします。
- (9) ホストポート信号ライン状態を確認します。“01”または“10”であれば、ホストポートにデバイスが接続されている状態と判断します。

このあと、ホスト・モードを選択した際に、`H_XcvrControl` レジスタと `VBUS_Enb` ビットは、ここで設定した値が保持されています。これらの設定は、ファームウェアがホストステート遷移実行レジスタに適切なコードを設定することで、設定したホストステートに応じてハードウェアが自動設定します。詳細は、「1.3.10 ホストステート管理サポート機能」を参照してください。

1.2 USBデバイス制御

本 LSI の USB デバイス機能について説明します。

1.2.1 エンドポイント

本 LSI はコントロール転送用のエンドポイント(EP0)と、5 本の汎用エンドポイント(EPa, EPb, EPc, EPd, EPe)を持ちます。エンドポイント EPa, EPb, EPc, EPd, EPe は、それぞれに、バルク、インタラプトまたはアイソクロナス転送用のエンドポイントとして使用できます。さらに、USB ホストからトランザクションが発行された時に、アラームを発生するアラームエンドポイント機能により、エンドポイント EP0 を除き、最大 15 の IN エンドポイントと最大 15 の OUT エンドポイントを持つことが可能です。アラームエンドポイント機能は、EPa,EPb,EPc,EPd,EPe に設定されたアクティブなエンドポイント以外の、任意のエンドポイントに対して発行されたトランザクションに対して、NAK 応答し、F/W に通知することが出来ます。

本 LSI のハードウェアは、エンドポイントを提供し、トランザクションの管理を行います。一方、USB に定義されるインタフェイス(以下、USB 定義インタフェイス)の管理機能を提供しません。USB 定義インタフェイスは、ファームウェアで実装して下さい。デバイス固有のデスクリプタ定義に沿って、エンドポイントを適宜設定し、組み合わせて、USB 定義インタフェイスを構成してください。

各エンドポイントには、USB 定義インタフェイスによって決定される固定の基本設定項目と、転送毎に制御を行う可変の制御項目及びステータスがあります。基本設定項目は、チップ初期化時、または、USB 定義インタフェイスの切り替え時等に設定して下さい。

表 1.5 にエンドポイント EP0(デフォルトコントロールパイプ)の基本設定項目を示します。

エンドポイント EP0 は、IN 方向と OUT 方向とで、レジスタセットや FIFO 領域を共有します。エンドポイント EP0 におけるデータステージ及びステータスステージでは、その実行に際して、ファームウェアによって、適宜データトランザクションの方向を設定して下さい。

後述する FIFO エリアに関連付けることにより、トランザクションを実行することが出来るようになります。まず、FIFO 領域を AREAx{x=0-5}StartAdrs_H,L 及び AREAx{x=0-5}EndAdrs_H,L で確保し、AREAnFIFO_Clr で FIFO 領域を初期化した後、AREAx{x=0-5}Join_1.JoinEP0CH0 をセットして下さい。このジョイン処理によって関連付けが行われるまで、FIFO を使用してのデータ転送が行えません。

表 1.5 エンドポイント EP0 の基本設定項目

項目	レジスタ/ビット	説明
マックスパケットサイズ	D_EP0MaxSize	マックスパケットサイズを、FS 動作時には 8,16,32,64 のいずれかの値に設定します。また、HS 動作時には 64 に設定します。 エンドポイント EP0 を FIFO の任意の AREAx{x=0-5}領域に関連付けて使用して下さい。
FIFO 領域	AREAx{x=0-5}StartAdrs_H,L AREAx{x=0-5}EndAdrs_H,L	AREAx{x=0-5}領域に割り当てる FIFO 領域を、アドレスで指定します。FIFO 領域には、マックスパケットサイズ以上を確保してください。
FIFO 関連付け	AREAx{x=0-5}Join_1.JoinEP0CH0	エンドポイント EP0 を FIFO 領域に関連付け、EP0におけるデータ転送を行えるようにします。

表 1.6 に汎用エンドポイント(EPa, EPb, EPc, EPd, EPe)の基本設定項目を示します。エンドポイント EPa, EPb, EPc, EPd, EPe は、トランザクション方向とエンドポイントナンバーを任意に設定出来ますので、5 本までの独立したエンドポイントを同時にアクティブにして使用できます。USB 定義インタフェイスの定義内容に合わせて適宜設定し、また有効にすることによって、USB 定義インタフェイスを構成して下さい。

1. 機能説明

エンドポイント EPa, EPb, EPc, EPd, EPe は、それぞれ FIFO 領域に関連付けられることによって、トランザクションを実行できるようになります。FIFO 領域を AREAx{x=0-5}StartAdrs_H,L 及び AREAx{x=0-5}EndAdrs_H,L レジスタにより確保し、AREAnFIFO_Clr にて FIFO を初期化した後、AREAx{x=0-5}Join_1 レジスタを適宜セットして下さい。このジョイン処理によって関連付けが行われて居ないエンドポイントは、トランザクションを実行しません。従って、使用しないエンドポイントでは、ジョイン処理をしないで下さい。

表 1.6 汎用エンドポイントの基本設定項目

項目	レジスタ/ビット	説明
トランザクション方向	D_EPx{x=a-e}Config_0.INxOUT	各エンドポイントの転送方向を設定します。
マックスパケットサイズ	D_EPx{x=a-e}MaxSize_H, D_EPx{x=a-e}MaxSize_L	各エンドポイントのマックスパケットサイズを、8Byte,16Byte,32Byte,64Byte,512Byte のいずれかの値に設定します。 但し、バルク転送を行うエンドポイントでは、FS モード時は 8/16/32/64Byte のいずれか、HS モード時は 512Byte に設定して下さい。 また、アイソクロナス転送を行うエンドポイントでは、FS モード時は 1~1023Byte のいずれか、HS モード時は 1~1024Byte のいずれかに設定します。
エンドポイントナンバー	D_EPx{x=a-e}Config_0.EndpointNumber	各エンドポイントのエンドポイントナンバーを 0x1~0xF の間の任意の値に設定します。
トグルモード	D_EPx{x=a-e}Config_0.IntEP_Mode	インタラプト転送の動作モードを設定します。バルク転送を行うエンドポイントでは、方向に関らず、“0”に設定して下さい。 IN 方向のエンドポイントでは、トグルシーケンスのモードを設定します。 OUT 方向のエンドポイントでは、インタラプト転送を行う場合に、“1”にセットして下さい。
アイソクロナスモード	D_EPx{x=a-e}Config_0.ISO	アイソクロナス転送を行うエンドポイントで、“1”にセットして下さい。
FIFO 領域	AREAx{x=0-5}StartAdrs_H,L AREAx{x=0-5}EndAdrs_H,L	領域をアドレスで指定して確保します。各領域は、関連付けるエンドポイントのマックスパケットサイズ以上のサイズを確保してください。また、FIFO のサイズは転送のスループットに影響する場合があります。
FIFO 関連付け	AREAx{x=0-5}Join_1	各エンドポイントと FIFO 領域とを関連付けます。

表 1.7 にアラームエンドポイントの基本設定項目を示します。このアラームエンドポイントは、現在汎用エンドポイントに割り当てられてはいませんが、USB 定義インタフェースにて定義したエンドポイントを実装するものです。USB 定義インタフェースの定義内容に合わせて適宜設定し、また有効にすることによって、USB 定義インタフェースを構成して下さい。

このアラームエンドポイントには、FIFO 領域は必要ありません。

表 1.7 アラームエンドポイントの基本設定項目

項目	レジスタ/ビット	説明
アラームエンドポイント有効	D_EnEP_IN_H.EnEPn{n=8-15}IN, D_EnEP_IN_L.EnEPn{n=1-7}IN, D_EnEP_OUT_H.EnEPn{n=8-15}OUT, D_EnEP_OUT_L.EnEPn{n=1-7}OUT	アラームエンドポイント有効にします。
アイソクロナスモード	D_EnEP_IN_ISO_H.EnEPn{n=8-15}IN_ISO, D_EnEP_IN_ISO_L.EnEPn{n=1-7}IN_ISO, D_EnEP_OUT_ISO_H.EnEPn{n=8-15}OUT_ISO, D_EnEP_OUT_ISO_L.EnEPn{n=1-7}OUT_ISO	アイソクロナス転送モードに設定する際に“1”に設定します。バルク転送、インタラプト転送に設定するエンドポイントでは、“0”に設定してください。

1.2.2 トランザクション

本 LSI は H/W でトランザクション実行機能と、ファームウェアに対するトランザクション実行のためのインタフェイスを提供します。ファームウェアに対するインタフェイスは、制御レジスタとステータスレジスタ、及び、ステータスによりアサートされる割り込み信号として実装されています。ステータスにより割り込みをアサートする設定については、「2. レジスタ詳細説明」を参照して下さい。

本 LSI は個々のトランザクション毎に、ファームウェアに対してステータスを発行します。しかしながら、ファームウェアは必ずしも、個々のトランザクションを管理する必要はありません。LSI はトランザクションへの応答を行う時に、FIFO を参照し、そのデータ数または空き数によって、データ転送を行えるか否かを判断して自動的に処理を行います。

例えば、OUT のエンドポイントであれば、ファームウェアは、CPU インタフェイス(DMA リードまたはレジスタリード)により、FIFO からデータを読み出して FIFO に空き領域を作り出すことによって、OUT トランザクションを自動的に連続して実行させることが出来ます。また、IN のエンドポイントであれば、ファームウェアは、CPU インタフェイス(DMA ライトまたはレジスタライト)により、FIFO にデータを書き込んで FIFO に有効データを作り出すことによって、IN トランザクションを自動的に連続して実行させることが出来ます。

表 1.8 にエンドポイント EP0 のトランザクション制御に関する制御項目及びステータスを示します。

表 1.8 エンドポイント EP0 の制御項目及びステータス

項目	レジスタ/ビット	説明
トランザクション方向	D_EP0Control.INxOUT	データステージ及びステータスステージにおいて、転送方向を設定します。
デスクリプタ返信イネーブル	D_EP0Control.ReplyDescriptor	デスクリプタの自動応答を起動します。
デスクリプタ返信アドレス	D_DescAdrs_H, DescAdrs_L	デスクリプタの自動応答によって、返信を行うデータの FIFO 上の先頭アドレスを指定します。
デスクリプタサイズ	D_DescSize_H, DescSize_L	デスクリプタの自動応答によって、返信を行うデータ数を指定します。
制御禁止	D_SETUP_Control.ProtectEP0	このビットがセットされていると、EP0ControlIN 及び EP0ControlOUT レジスタの ForceNAK ビットと ForceSTALL ビットへのアクセスが行えません。 このビットは、RcvEP0SETUP ステータスが立つと、LSI の h/w によってセットされ、CPU によるレジスタアクセスでクリアできます。
ショートパケット送信イネーブル	D_EP0ControlIN.EnShortPkt	マックスパケットサイズに満たない、ショートパケットの送信を有効にします。ショートパケットを送信した IN トランザクションが完結すると、クリアされます。
トグルシーケンスビット	D_EP0ControlIN.ToggleStat, D_EP0ControlOUT.ToggleStat	トグルシーケンスビットの状態を示します。SETUP ステージにより、自動的に初期化されます。
トグルセット	D_EP0ControlIN.ToggleSet, D_EP0ControlOUT.ToggleSet	トグルシーケンスビットをセットします。
トグルクリア	D_EP0ControlIN.ToggleClr, D_EP0ControlOUT.ToggleClr	トグルシーケンスビットをクリアします。
強制 NAK 応答	D_EP0ControlIN.ForceNAK, D_EP0ControlOUT.ForceNAK	FIFO のデータ数/空き数に関わらず、IN 又は OUT(PING を含む)トランザクションに NAK 応答します。
STALL 応答	D_EP0ControlIN.ForceSTALL, D_EP0ControlOUT.ForceSTALL	IN 又は OUT(PING を含む)トランザクションに STALL 応答します。

1. 機能説明

自動 ForceNAK セット	D_EP0ControlOUT.AutoForceNAK	OUT トランザクションの完結毎に、D_EP0ControlOUT.ForceNAK ビットをセットします。
SETUP 受信ステータス	USB_DeviceIntStat.RcvEP0SETUP	SETUP トランザクションが実行されたことを示します。
トランザクションステータス	D_EP0IntStat.OUT_ShortACK, D_EP0IntStat.IN_TranACK, D_EP0IntStat.OUT_TranACK, D_EP0IntStat.IN_TranNAK, D_EP0IntStat.OUT_TranNAK, D_EP0IntStat.IN_TranErr, D_EP0IntStat.OUT_TranErr	トランザクションの結果を示します。
デスクリプタ返信データステージ終了ステータス	D_EP0IntStat.DescriptorCmp	デスクリプタ自動応答のデータステージが終了したことを示します。

表 1.9 に、汎用エンドポイント EPa, EPb, EPc, EPd, EPe のトランザクション処理に関する制御項目とステータスを示します。

表 1.9 汎用エンドポイントの制御項目とステータス

項目	レジスタ/ビット	説明
自動 ForceNAK セット	D_EPx{x=a-e}Control.AutoForceNAK	OUT トランザクションの完結毎に、そのエンドポイントの D_EPx{x=a-e}Control.ForceNAK ビットをセットします。
ショートパケット送信イネーブル	D_EPx{x=a-e}Control.EnShortPkt	IN トランザクションに対し、マックスパケットサイズに満たない、ショートパケットの送信を有効にします。ショートパケットを送信した IN トランザクションが完結すると、クリアされます。
ショートパケット受信による自動 ForceNAK セットの禁止	D_EPx{x=a-e}Control.DisAF_NAK_Short	OUT トランザクションにおいて、ショートパケットを受信すると、自動的にそのエンドポイントの D_EPx{x=a-e}Control.ForceNAK ビットをセットする機能(※)を禁止します。 ※ このビットにより禁止しない場合は有効になっています。
トグルシーケンスビット	D_EPx{x=a-e}Control.ToggleStat	トグルシーケンスビットの状態を示します。
トグルセット	D_EPx{x=a-e}Control.ToggleSet	トグルシーケンスビットをセットします。
トグルクリア	D_EPx{x=a-e}Control.ToggleClr	トグルシーケンスビットをクリアします。
強制 NAK 応答	D_EPx{x=a-e}Control.ForceNAK	FIFO のデータ数/空き数に関わらず、トランザクションに NAK 応答します。
STALL 応答	D_EPx{x=a-e}Control.ForceSTALL	トランザクションに STALL 応答します。
トランザクションステータス	D_EPx{x=a-e}IntStat.OUT_ShortACK, D_EPx{x=a-e}IntStat.IN_TranACK, D_EPx{x=a-e}IntStat.OUT_TranACK, D_EPx{x=a-e}IntStat.IN_TranNAK, D_EPx{x=a-e}IntStat.OUT_TranNAK, D_EPx{x=a-e}IntStat.IN_TranErr, D_EPx{x=a-e}IntStat.OUT_TranErr	トランザクションの結果を示します。

1.2.2.1 SETUP トランザクション

本 LSI は、自ノードのエンドポイント EP0 宛ての SETUP トランザクションを、無条件に実施します。(D_NegoControl.ActiveUSB ビットによって USB 機能は有効にされている必要が有ります。)

SETUP トランザクションが発行されると、データパケット(8Byte)の全ての内容を D_EP0SETUP_0~D_EP0SETUP_7 レジスタに格納し、ACK 応答します。また、SetAddress() リクエストを除き、ファームウェアに対し RcvEP0SETUP ステータスを発行します。

本 LSI は SETUP トランザクション中にエラーが発生した場合には、応答せず、ステータスを発行しません。

本 LSI は SETUP トランザクションが完結すると、D_EP0ControlIN レジスタ及び D_EP0ControlOUT レジスタの ForceNAK ビットをセットし、ForceSTALL ビットをクリアします。また、ToggleStat ビットをセットします。また、D_SETUP_Control.ProtectEP0 ビットをセットします。ファームウェアは、エンドポイント EP0 の設定を終え、次のステージに移行可能になったら、SETUP_Control.ProtectEP0 ビットをクリアし、D_EP0ControlIN レジスタまたは D_EP0ControlOUT レジスタにおいて、該当する方向の ForceNAK ビットをクリアして下さい。

図 1.1 に SETUP トランザクションの様子を図示します。(a)ホストが、このノードのエンドポイント 0 に宛てた SETUP トークンを発行します。(b)ホストは続けて、8Byte 長のデータパケットを送信します。LSI はこのデータを D_EP0SETUP_0~D_EP0SETUP_7 レジスタに書き込みます。(c)LSI は自動的に ACK 応答します。また、自動設定するレジスタを設定し、ファームウェアに対しステータスを発行します。

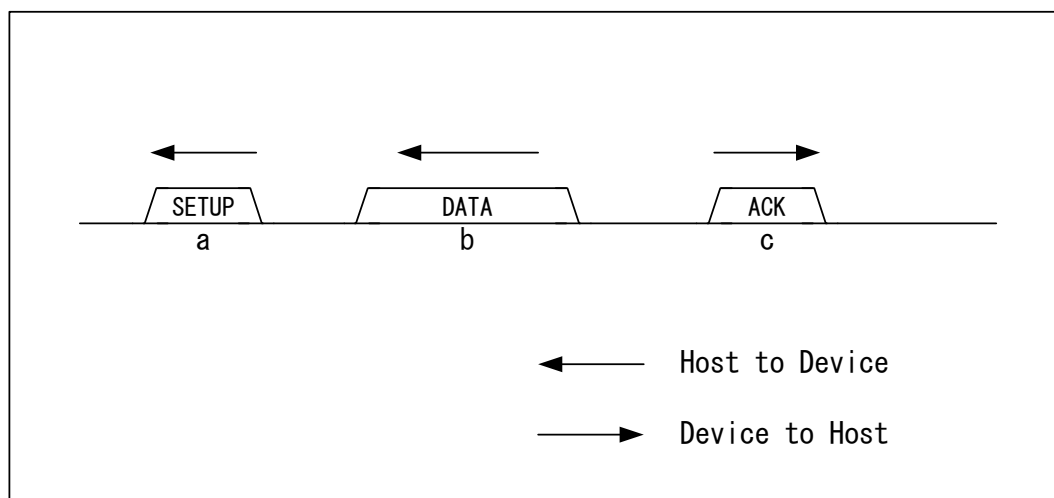


図 1.1 SETUP トランザクション

1. 機能説明

1.2.2.2 バルク／インタラプトOUTトランザクション

バルク及びインタラプト OUT トランザクションでは、FIFO の空き容量がマックスパケットサイズ以上有ると、データの受信を開始します。

本 LSI は、バルク及びインタラプト OUT トランザクションにおいて、全てのデータが正常に受信できると、トランザクションを完結し、ACK または NYET 応答します。また、ファームウェアに対し、該当するエンドポイントの OUT_TrانACK ステータス (D_EPx{x=0,a-c}IntStat.OUT_TrانACK ビット)を発行します。また、FIFO を更新して、データを受信済みとして、領域を確保します。

また、バルク及びインタラプト OUT トランザクションにおいて、ショートパケットの全てのデータを受信できると、上記のトランザクション完結処理に加え、OUT_ShortACK ステータス (D_EPx{x=0,a-e}IntStat.OUT_ShortACK ビット)を発行します。また、D_EPx{x=a-e}Control.DisAF_NAK_Short ビットがクリアされていると、そのエンドポイントの、D_EPx{x=a-e}Control.ForceNAK ビットをセットします。

バルク及びインタラプト OUT トランザクションにおいて、トグルミスマッチが発生した場合、トランザクションに ACK 応答しますが、ステータスを発行しません。FIFO は更新されません。

バルク及びインタラプト OUT トランザクションにエラーが発生した場合、トランザクションに応答しません。また、OUT_TrانErr ステータス(D_EPx{x=0,a-e}IntStat.OUT_TrانErr ビット)を発行します。FIFO は更新されません。

バルク及びインタラプト OUT トランザクションにおいて、全てのデータを受信できなかった場合、トランザクションに NAK 応答します。また、OUT_TrانNAK ステータス (D_EPx{x=0,a-e}IntStat.OUT_TrانNAK ビット)を発行します。FIFO は更新されません。

図 1.2 に、完結する場合のバルクまたはインタラプト OUT トランザクションの様子を図示します。(a)ホストが、このノードに存在する OUT 方向のエンドポイントに宛てた OUT トークンを発行します。(b)ホストは続けて、マックスパケットサイズ以内のデータパケットを送信します。LSI はこのデータを、該当するエンドポイントの FIFO に書き込みます。(c)LSI はデータを受信できると、自動的に ACK 応答します。また、自動設定するレジスタを設定し、ファームウェアに対しステータスを発行します。

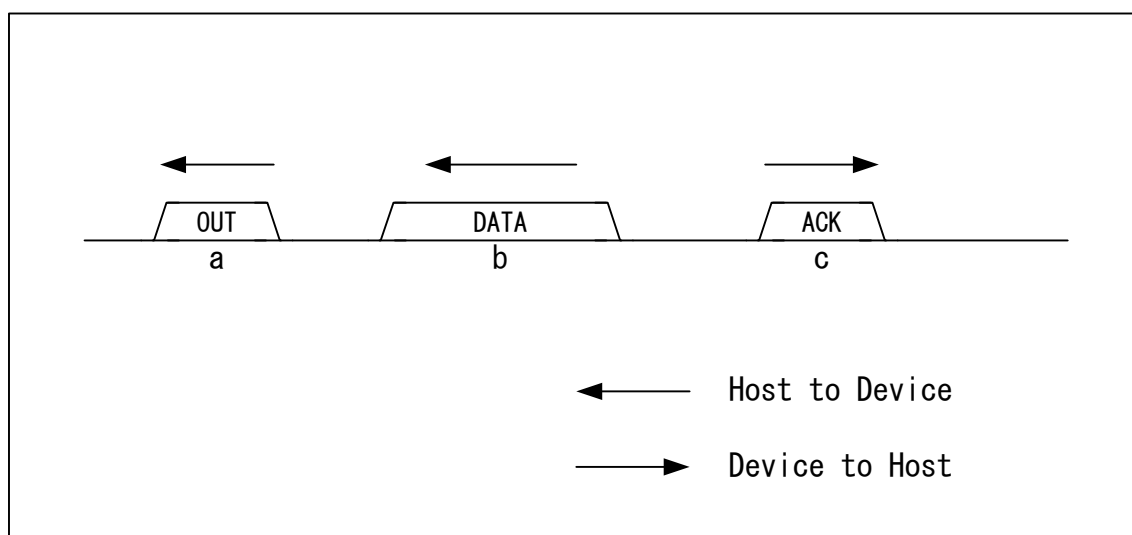


図 1.2 OUT トランザクション

1.2.2.3 アイソクロナスOUTトランザクション

アイソクロナス OUT トランザクションでは、FIFO の空き容量がマックスパケットサイズ以上有ると、データの受信を開始します。従いまして、CPU インタフェイスによるレジスタリードや DMA リード等により FIFO のデータを読み出して、空き領域を作りながら受信をすることが出来るよう、FIFO をマックスパケットサイズの倍程度割り当てることにより、良好なスループットを提供します。

アイソクロナス OUT トランザクションにおいて、全てのデータが正常に受信できると、F/W に対し、該当するエンドポイントの OUT_TrانACK ステータス (EPx{x=a-e}IntStat.OUT_TrانACK ビット)を発行します。また、FIFO を更新して、データを受信済みとして、領域を確保します。

アイソクロナス OUT トランザクションにおいて、マックスパケットサイズ未満の全てのデータを受信できると、上記のトランザクション完結処理に加え、OUT_ShortACK ステータス (EPx{x=a-e}IntStat.OUT_ShortACK ビット)を発行します。また、EPx{x=a-e}Control.DisAF_NAK_Short ビットがクリアされていると、そのエンドポイントの、EPx{x=a-e}ForceNAK ビットをセットします。

アイソクロナス OUT トランザクションにエラーが発生した場合、データを受信せず、FIFO を更新しません。また、OUT_TrانErr ステータス (EPx{x=a-e}IntStat.OUT_TrانErr ビット)を発行します。

アイソクロナス OUT トランザクションにおいて、1 パケット分の全てのデータを受信できなかった場合、OUT_TrانNAK ステータス (EPx{x=a-e}IntStat.OUT_TrانNAK ビット)を発行します。FIFO は更新されません。

1.2.2.4 バルク/インタラプトINトランザクション

IN 方向のバルク及びインタラプトのエンドポイントにおいて、FIFO にマックスパケットサイズ分のデータが有るか、または、ファームウェアによってショートパケットの送信が許可されていると、IN トランザクションに応答して、データパケットを返信します。

ショートパケット(データ長ゼロのパケットを含む)の送信許可は、D_EP0ControlIN.EnShortPkt ビットまたは D_EPx{x=a-e}Control.EnShortPkt ビットをセットすることで行います。ショートパケットを送信する場合、送信許可後、トランザクションが完結するまでの間、新たなデータをそのエンドポイントの FIFO に書き込まないようにして下さい。

エンドポイント EP0 では、ショートパケットを送信する IN トランザクションが完結すると、D_EP0ControlIN.ForceNAK ビットがセットされます。

データ返信した IN トランザクションで、ACK を受信すると、トランザクションを完結し、ファームウェアに対し、IN_TrانACK ステータス (D_EPx{x=0,a-e}IntStat.IN_TrانACK ビット)を発行します。また、FIFO を更新して、送信したデータを送信済みとして領域を開放します。

データ返信した IN トランザクションで、ACK を受信しないと、トランザクションを失敗と見なし、ファームウェアに対し、IN_TrانErr ステータス (D_EPx{x=0,a-e}IntStat.IN_TrانErr ビット)を発行します。また、FIFO を更新せず、領域を開放しません。

バルク及びインタラプトの IN 方向のエンドポイントにおいて、FIFO にマックスパケットサイズ分のデータが無く、かつ、ショートパケットの送信が許可されていないと、IN トランザクションに NAK 応答し、ファームウェアに対し、IN_TrانNAK ステータス (D_EPx{x=0,a-e}IntStat.IN_TrانNAK ビット)を発行します。また、FIFO を更新せず、領域を開放しません。

図 1.3 に、完結する場合のバルクまたはインタラプト IN トランザクションの様子を図示します。(a)ホストが、このノードに存在する IN 方向のエンドポイントに宛てた IN トークンを発行します。(b)LSI は、この IN トランザクションに応答できる場合、マックスパケットサイズ以内のデータパケットを送信します。(c)ホストは ACK 応答します。LSI は ACK を

1. 機能説明

受信すると、自動設定するレジスタを設定し、ファームウェアに対しステータスを発行します。

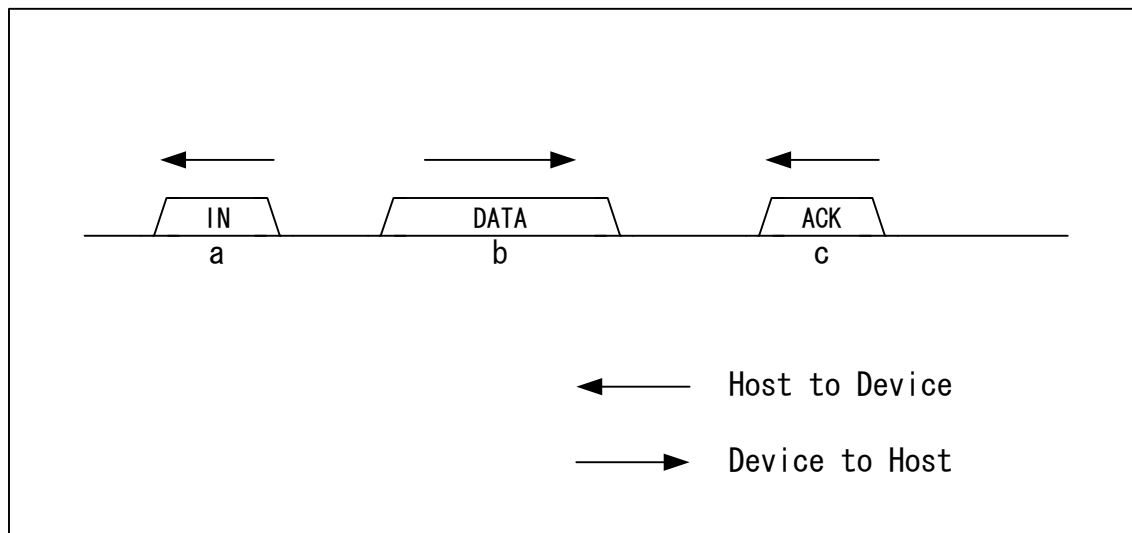


図 1.3 IN トランザクション

1.2.2.5 アイソクロナスINトランザクション

IN 方向のアイソクロナスのエンドポイントにおいて、FIFO にマックスパケットサイズ分のデータが在るか、または、F/W によってショートパケットの送信が許可されていると、IN トランザクションに応答して、データパケットを返信します。

ショートパケット(データ長ゼロのパケットを含む)の送信許可は、EPx{x=a-e}Control.EnShortPkt ビットをセットすることで行います。ショートパケットを送信する場合、送信許可後、トランザクションが完結するまでの間、新たなデータをそのエンドポイントの FIFO に書き込まないようにして下さい。

アイソクロナス IN トランザクションにデータパケットを返信すると、トランザクションを完結し、F/W に対し、IN_TranACK ステータス(EPx{x=a-e}IntStat.IN_TranACK ビット)を発行します。また、FIFO を更新して、送信したデータを送信済みとして領域を開放します。

アイソクロナスの IN 方向のエンドポイントにおいて、FIFO にマックスパケットサイズ分のデータが無く、かつ、ショートパケットの送信が許可されていないと、IN トランザクションにゼロ長データパケットで応答し、F/W に対し、IN_TranNAK ステータス(EPx{x=a-e}IntStat.IN_TranNAK ビット)を発行します。また、FIFO を更新せず、領域を開放しません。

1.2.2.6 PINGトランザクション

バルクの OUT 方向のエンドポイントでは、HS 動作時に、PING トランザクションを実行します。

該当するエンドポイントの FIFO 空き容量がマックスパケットサイズ以上であった場合に、PING トランザクションに対して ACK 応答します。また、ファームウェアに対して、ステータスを発行しません。

該当するエンドポイントの FIFO の空き容量がマックスパケットサイズ未満であった場合に、PING トランザクションに対して NAK 応答します。また、ファームウェアに対し、OUT_TrانNAK ステータス(D_EPx{x=0,a-e}IntStat.OUT_TrانNAK ビット)を発行します。

PING トランザクションにおいては、FIFO が更新されることはありません。

図 1.4 に、PING トランザクションに対して ACK 応答する様子を図示します。(a)ホストが、このノードに存在する OUT 方向のエンドポイントに宛てた PING トークンを発行します。(b)LSI は、FIFO にマックスパケットサイズ分の空きがある場合、この PING トランザクションに対し ACK 応答します。また、ファームウェアに対しステータスを発行します。

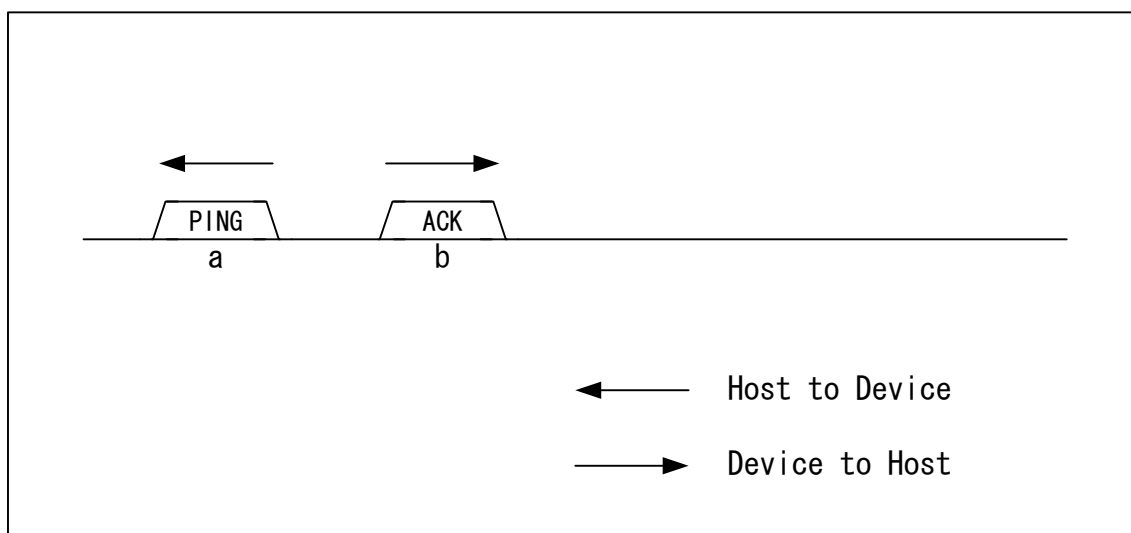


図 1.4 PING トランザクション

1. 機能説明

1.2.3 コントロール転送

エンドポイント EP0 におけるコントロール転送は、`SetAddress()` リクエストを除き、個々のトランザクションの組み合わせとして制御します。`SetAddress()` リクエストは、後述の自動アドレス設定機能により、自動的に処理されます。

図 1.5 にデータステージが OUT 方向である場合のコントロール転送の様子を図示します。(a) ホストは、SETUP トランザクションによって、コントロール転送を開始します。デバイスのファームウェアはリクエストの内容を解析して、データステージに応答する準備をします。(b) ホストは OUT トランザクションを発行して、データステージを行い、デバイスはデータを受信します。(c) ホストは IN トランザクションを発行して、ステータスステージを行い、デバイスはデータ長ゼロの packets を返信します。

データステージの無いコントロール転送は、この例におけるデータステージが無い状態で実施されます。

ステータスステージへの移行は、ホストがデータステージと逆方向のトランザクションを発行することによってなされます。ファームウェアは、`IN_TranNAK` ステータス (`D_EP0IntStat.IN_TranNAK` ビット) を監視して、データステージからステータスステージに移行するきっかけとして下さい。

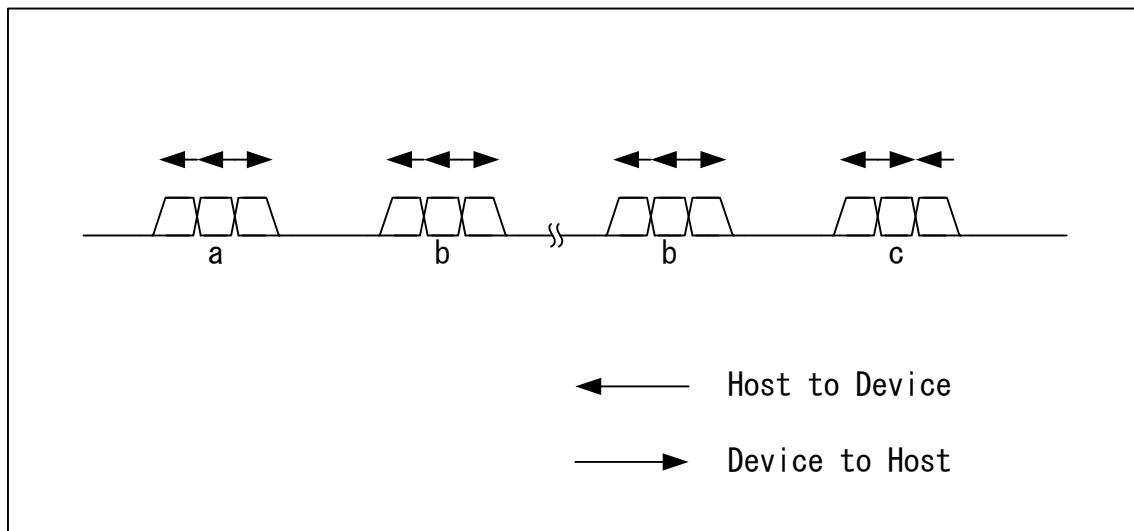


図 1.5 データステージが OUT 方向のコントロール転送

図 1.6 にデータステージが IN 方向である場合のコントロール転送の様子を図示します。(a) ホストは、SETUP トランザクションによって、コントロール転送を開始します。デバイスのファームウェアはリクエストの内容を解析して、データステージに応答する準備をします。(b) ホストは IN トランザクションを発行して、データステージを行い、デバイスはデータを送信します。(c) ホストは OUT トランザクションを発行して、ステータスステージを行い、デバイスは ACK 応答します。

ステータスステージへの移行は、ホストがデータステージと逆方向のトランザクションを発行することによってなされます。ファームウェアは、`OUT_TranNAK` ステータス (`D_EP0IntStat.OUT_TranNAK` ビット) を監視して、データステージからステータスステージに移行するきっかけとして下さい。

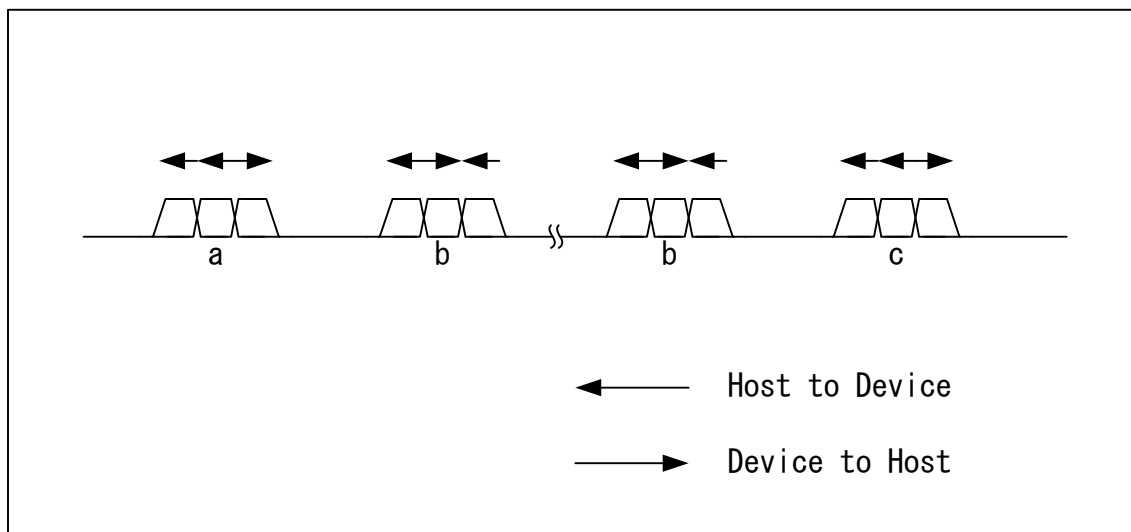


図 1.6 データステージが IN 方向のコントロール転送

コントロール転送のデータステージ及びステータスステージは、通常の OUT 及び IN トランザクションを行いますので、NAK によるフロー制御が有効です。デバイスは定められた時間内に、応答する準備をすることが許されています。

1.2.3.1 セットアップステージ

自ノードに宛てられた SETUP トークンを受信すると、自動的にセットアップトランザクションを実行します。

ファームウェアは、RcvEP0SETUP ステータスをモニタし、D_EP0SETUP_0～D_EP0SETUP_7 レジスタによりリクエストを解析して、コントロール転送を制御して下さい。

受信したリクエストが、OUT 方向のデータステージが有るものであった場合、データステージに移行するため、D_EP0Control レジスタの INxOUT ビットをクリアして、エンドポイント EP0 を OUT 方向に設定して下さい。

受信したリクエストが、IN 方向のデータステージが有るものであった場合、データステージに移行するため、D_EP0Control レジスタの INxOUT ビットをセットして、エンドポイント EP0 を IN 方向に設定して下さい。

受信したリクエストが、データステージの無いものであった場合、ステータスステージに移行するため、D_EP0Control レジスタの INxOUT ビットをセットして、エンドポイント EP0 を IN 方向に設定して下さい。

1.2.3.2 データステージ/ステータスステージ

D_EP0SETUP_0～D_EP0SETUP_7 レジスタを読み出してリクエストを解析した内容に従って、次のステージに移行して下さい。

そのステージが OUT 方向である場合、D_EP0Control レジスタの INxOUT をクリアして OUT 方向に設定し、D_EP0ControlOUT レジスタを適宜設定して、ステージをコントロールして下さい。SETUP ステージ終了時は、ForceNAK ビットがセットされています。また、D_SETUP_Control.ProtectEP0 ビットがセットされています。

そのステージが IN 方向である場合、D_EP0Control レジスタの INxOUT をセットして IN 方向に設定し、D_EP0ControlIN レジスタを適宜設定して、ステージをコントロールして下さい。SETUP ステージ終了時は、ForceNAK ビットがセットされています。また、D_SETUP_Control.ProtectEP0 ビットがセットされています。

1. 機能説明

1.2.3.3 自動アドレス設定機能

本 LSI には、エンドポイント EP0 におけるコントロール転送において、SetAddress() リクエストの処理を自動化する機能があります。

LSI の h/w は D_EP0SETUP_0～D_EP0SETUP_7 レジスタによってリクエストの内容を確認し、有効な SetAddress() リクエストで有った場合には、ファームウェアに通知することなく、このリクエストのステータスステージの処理に移行します。ステータスステージが完了すると、D_USB_Address レジスタにアドレスを設定し、ファームウェアに対し SetAddressCmp ステータス(D_SIE_IntStat.SetAddressCmp ビット)を発行します。

ファームウェアは SetAddressCmp ステータスを監視し、これが発行されたら D_USB_Address レジスタにより、アドレスを確認できます。

1.2.3.4 デスクリプタ返信機能

本 LSI には、エンドポイント EP0 におけるコントロール転送において、GetDescriptor()等の複数回発行され同じデータを要求するリクエストに有効な、デスクリプタ返信機能があります。

データステージが IN 転送であるリクエストにおいて、ファームウェアはこの機能を使用することが出来ます。

D_EP0ControlIN.ForceNAK ビットをクリアして、データステージへの応答を開始する前に、D_DescAdrs_H,L レジスタに FIFO のデスクリプタ領域内の、返信するデータの先頭アドレスを、また、D_DescSize_H,L レジスタに返信するデータの総バイト数を設定し、D_EP0Control.ReplyDescriptor ビットをセットして下さい。

デスクリプタ返信機能は、設定数のデータを送り終わるまで、データステージの IN トランザクションに응答してデータパケットを返信し、IN トランザクションを実行します。設定数のデータを送り終えた後に IN トランザクションが発行されると、NAK 応答します。マックスパケットサイズに対し端数のデータが存在すると、デスクリプタ返信機能は、D_EP0ControlIN.EnShortPkt をセットし、全てのデータを返信するまで、IN トランザクションに응答出来るようにします。

OUT トークンを受信し、ステータスステージへの移行を検知すると、D_EP0Control.ReplyDescriptor ビットをクリアし、ファームウェアに対し DescriptorCmp ステータス(D_EP0IntStat.DescriptorCmp ビット)を発行します。DescriptorCmp ステータスを検知したら、ファームウェアはステータスステージを行って下さい。

デスクリプタ領域については、「1.5 FIFO 管理」を参照して下さい。

1.2.4 バルク転送／インタラプト転送／アイソクロナス転送

汎用エンドポイント EPa, EPb, EPc, EPd, EPe におけるバルク転送、インタラプト転送、アイソクロナス転送は、データフロー(「1.2.5 データフロー」参照)としても、連続する個々のトランザクション(「1.2.2 トランザクション」参照)としても制御できます。

1.2.5 データフロー

OUT 転送及び IN 転送の一般的なデータフローの制御について、説明します。

1.2.5.1 OUT転送

OUT 転送によって受信したデータは、各エンドポイントに関連付けられた FIFO 上へ書き込まれます。FIFO のデータを読み出すには、CPU インタフェイスによるレジスタ読み出し、CPU インタフェイスによる DMA 読み出しの方法があります。

CPU インタフェイスのレジスタリードにより、FIFO のデータを読み出すには、`AREAx{x=0-5}.Join_0.JoinCPU_Rd` ビットにより、ただ一つの FIFO 領域を選択して下さい。選択した FIFO 領域は、`FIFO_Rd` レジスタ、または、`FIFO_ByteRd` レジスタにより、受信順に読み出すことが出来ます。また、読み出し可能な FIFO のデータ数を、`FIFO_RdRemain_H,L` レジスタにより参照できます。空の FIFO を読み出すことはできませんので、必ず `FIFO_RdRemain_H,L` レジスタによりデータ数を確認し、その数を超えないように読み出して下さい。

CPU インタフェイスの DMA リードにより、FIFO のデータを読み出すには、`AREAx{x=0-5}.Join_0.JoinDMA` ビットにより、ただ一つの FIFO 領域を選択し、`DMA_Control.Dir` ビットに“1”を設定して下さい。選択した FIFO 領域は、CPU インタフェイスにおいて DMA 手順を実行することにより、受信順に読み出されます。また、FIFO の残りデータ数を、`DMA_Remain_H,L` レジスタで参照できます。FIFO が空になると、CPU インタフェイスは自動的に DMA を一時停止してフロー制御を行います。

FIFO にデータパケットを受信できる空き領域があれば、OUT トランザクションに自動的に応答して、データを受信できます。従って、ファームウェアによって、個々のトランザクションについての制御を行うことなく、OUT 転送を行うことができます。但し、`D_EPx{x=a-e}.Control.DisAF_NAK_Short` ビットがクリアされている場合(初期値)、ショートパケット(データ長ゼロのパケットを含む)を受信した場合、そのエンドポイントの `D_EPx{x=a-e}.Control.ForceNAK` ビットをセットしますので、次のデータ転送を行う準備が出来たら、`D_EPx{x=a-e}.Control.ForceNAK` ビットをクリアして下さい。

1.2.5.2 IN転送

IN 転送により送信するデータを、各エンドポイントに関連付けられた FIFO 領域へ書き込んで下さい。FIFO にデータを書き込むには、CPU インタフェイスによるレジスタ書き込みと、CPU インタフェイスによる DMA 書き込みの方法があります。

CPU インタフェイスのレジスタライトにより、FIFO にデータを書き込むには、`AREAx{x=0-5}.Join_0.JoinCPU_Wr` ビットにより、ただ一つの FIFO 領域を選択して下さい。選択した FIFO 領域には、`FIFO_Wr_0,1` レジスタまたは `FIFO_ByteWr` レジスタにより書き込むことができ、書き込み順にデータパケットで送信されます。また、FIFO の空き容量を、`FIFO_WrRemain_H,L` レジスタにより参照できます。フル状態の FIFO へ書き込むことは出来ません。必ず `FIFO_WrRemain_H,L` レジスタにより空き数を確認し、その数を超えないように書き込んで下さい。

CPU インタフェイスの DMA ライトにより、FIFO にデータを書き込むには、`AREAx{x=0-5}.Join_0.JoinDMA` ビットにより、DMA のチャンネルにただ一つの FIFO 領域を選択し、`DMA_Control.Dir` ビットに“0”を設定して下さい。選択した FIFO 領域には、CPU インタフェイスにおいて DMA の手順を実行することにより、書き込まれ、書き込み順にデータパケットで送信されます。FIFO がフルになると、CPU インタフェイスは自動的に DMA を一時停止してフロー制御を行います。

FIFO にマックスパケットサイズ以上のデータがあれば、IN トランザクションに自動的に応答して、データを送信できます。従って、ファームウェアによって、個々のトランザクションについての制御を行うことなく、IN 転送を行うことが出来ます。但し、データ転送の最後にショートパケットを送信する必要がある場合、`EnShortPkt` ビットをセットして下さい。このビットはショートパケットを送信した IN トランザクションが完結することに

1. 機能説明

よってクリアされます。FIFO へのデータ書き込みが終了した時点でセットすることが可能です。また、DMA_FIFO_Control.AutoEnShort ビットがセットされていると、CPU インタフェ이스の DMA 書き込みが終了した時に、マックスパケットサイズに満たない端数データが FIFO に有ると、そのエンドポイントの EnShortPkt ビットを自動的にセットします。

1.2.6 バルクオンリーサポート

本 LSI には、エンドポイント EPa, EPb, EPc, EPd, EPe におけるバルク転送において、USB Mass Storage Class(BulkOnly Transport Protocol)に固有の Command Block Wrapper(CBW)の受信及び Command Status Wrapper(CSW)の送信を補助する、バルクオンリーサポート機能があります。

D_BulkOnlyConfig.EPx{x=a-e}BulkOnly ビットをセットすると、対象となるエンドポイントで、バルクオンリーサポート機能が有効になります。

バルクオンリーサポート機能が有効にされ、CBW サポート、または CSW サポートが実行されている間は、エンドポイントに通常割り当てられた FIFO 領域ではなく、CBW 領域または CSW 領域として割り当てられている領域を使用して、パケットの受信(CBW)または送信(CSW)を行います。

1.2.6.1 CBWサポート

ファームウェアは BulkOnly Transport Protocol のコマンドトランスポートを行うときに、CBW サポートを使用することが出来ます。D_BulkOnlyConfig.EPx{x=a-e}BulkOnly ビットがセットされると、対応する OUT のエンドポイントで CBW サポートが有効になります。CBW サポートは、ただひとつのエンドポイントで有効になるように設定して下さい。CBW サポートが有効であるときに、D_BulkOnlyControl.GoCBW_Mode ビットをセットすると、CBW サポートが実行され、対象となるエンドポイントにおける OUT トランザクションで受信したデータを CBW として扱います。

データパケットのデータ長が CBW として期待される 31 バイト長であった場合には、データを CBW 領域に保存し、ファームウェアに対し CBW 完了ステータス(D_BulkIntStat.CBW_Cmp ビット)を発行します。また、D_BulkOnlyControl.GoCBW_Mode ビットを自動的にクリアし、CBW サポートの実行が終了します。また、このとき D_BulkOnlyControl.GoCSW_Mode ビットがセットされていると、同時にクリアします。

データパケットのデータ長が、31 バイト長未満か、または、31 バイト長を超えた場合には、データを受信せず、ファームウェアに対し CBW データ長エラーステータス(D_BulkIntStat.CBW_LengthErr ビット)を発行します。また、D_BulkOnlyControl.GoCBW_Mode ビットを自動的にクリアし、CBW サポートの実行を終了します。また、このとき D_BulkOnlyControl.GoCSW_Mode ビットがセットされていると、同時にクリアします。CBW_Err ステータスが発行された場合、BulkOnly Transport Protocol でフェーズミスマッチが発生していますので、ファームウェアはエンドポイントを STALL するなどして、通信の復旧を行ってください。

対象となるエンドポイントで D_EPx{x=a-e}Control.ForceSTALL がセットされ、OUT トランザクションに STALL 応答した場合には、ファームウェアに対し CBW エラーステータス(D_BulkIntStat.CBW_Err ビット)を発行し、D_BulkOnlyControl.GoCBW_Mode ビットをクリアし、CBW サポートの実行が終了します。また、このとき D_BulkOnlyControl.GoCSW_Mode ビットがセットされていると、同時にクリアします。

OUT トランザクションに CRC エラーなどのトランザクションエラーが発生した場合は、データを受信せずファームウェアに対し CBW トランザクションエラーステータス(D_BulkIntStat.CBW_TranErr ビット)ステータスを発行します。この場合には、D_BulkOnlyControl.GoCBW_Mode ビットがクリアされず、CBW サポートの実行が継続します。また、このとき D_BulkOnlyControl.GoCSW_Mode ビットがセットされていてもクリアされません。

CBW 領域に受信したデータは、RAM_Rd 機能を用いて読み出すことが出来ます。

1.2.6.2 CSWサポート

ファームウェアは BulkOnly Transport Protocol のステータストランスポートを行うときに、CSW サポートを使用することが出来ます。D_BulkOnlyConfig.EPx{x=a-e}BulkOnly ビットがセットされると、対応する IN のエンドポイントで CSW サポートが有効になります。CSW サポートは、ただひとつのエンドポイントで有効になるように制御して下さい。CSW サポートが有効であるときに、D_BulkOnlyControl.GoCSW_Mode ビットをセットすると、CSW サポートが実行され、対象となるエンドポイントにおける IN トランザクションを CSW として扱い、CSW 領域から 13 バイトのデータを送信します。

IN トランザクションにおいて、13 バイトの CSW データをホストへ返信した後に、ホストからの ACK を受信してトランザクションを完結した場合には、ファームウェアに対し CSW 完了ステータス (D_BulkIntStat.CSW_Cmp ビット) を発行します。また、D_BulkOnlyControl.GoCSW_Mode ビットを自動的にクリアして CSW サポートの実行を終了します。また、同時に D_BulkOnlyControl.GoCBW_Mode ビットをセットして CBW サポートの実行を開始します。

IN トランザクションにおいて、13 バイトのデータをホストへ返信した後に、ホストからの ACK が受信できなかった場合は、ファームウェアに対し CSW エラーステータス (D_BulkIntStat.CSW_Err ビット) を発行します。この時、D_BulkOnlyControl.GoCSW_Mode ビットをクリアせずに CSW サポートの実行を継続します。また、同時にハードウェアが D_BulkOnlyControl.GoCBW_Mode ビットをセットして CBW サポートの実行を開始します。即ち、この場合には、CSW サポートの実行と、CBW サポートの実行が同時に行われている状態となります。もし、ホストが CSW を受信できずエラーとなっていた場合には、CSW のリトライが行われますが、CSW サポートが実行中なので応答することができます。また、デバイスが ACK を受信できずにエラーとなった場合には、次の CBW が行われますが、CBW サポートが実行中なので、応答することができ、また、CBW サポートが行われることによって CSW サポートの実行が終了されます。

CSW 領域へは、RAM_WrDoor 機能を用いてデータを書き込むことが出来ます。

1. 機能説明

1.2.7 ケーブルネゴシエーション機能(オート・ネゴシエータ)

オート・ネゴシエータは、サスペンド検出、リセット検出、HS Detection Handshake 実行、レジューム検出、等のバスイベントに対し、USB バスの状態を逐次チェックしながらケーブルネゴシエーションを自動的に行います。実行結果は、各割り込み (DetectRESET、DetectSUSPEND、ChirpCmp、RestoreCmp)により示されます。オート・ネゴシエータはケーブルの挿抜を扱いませんので、ケーブル接続後に有効にし、また、ケーブル切断後に無効にしてください。

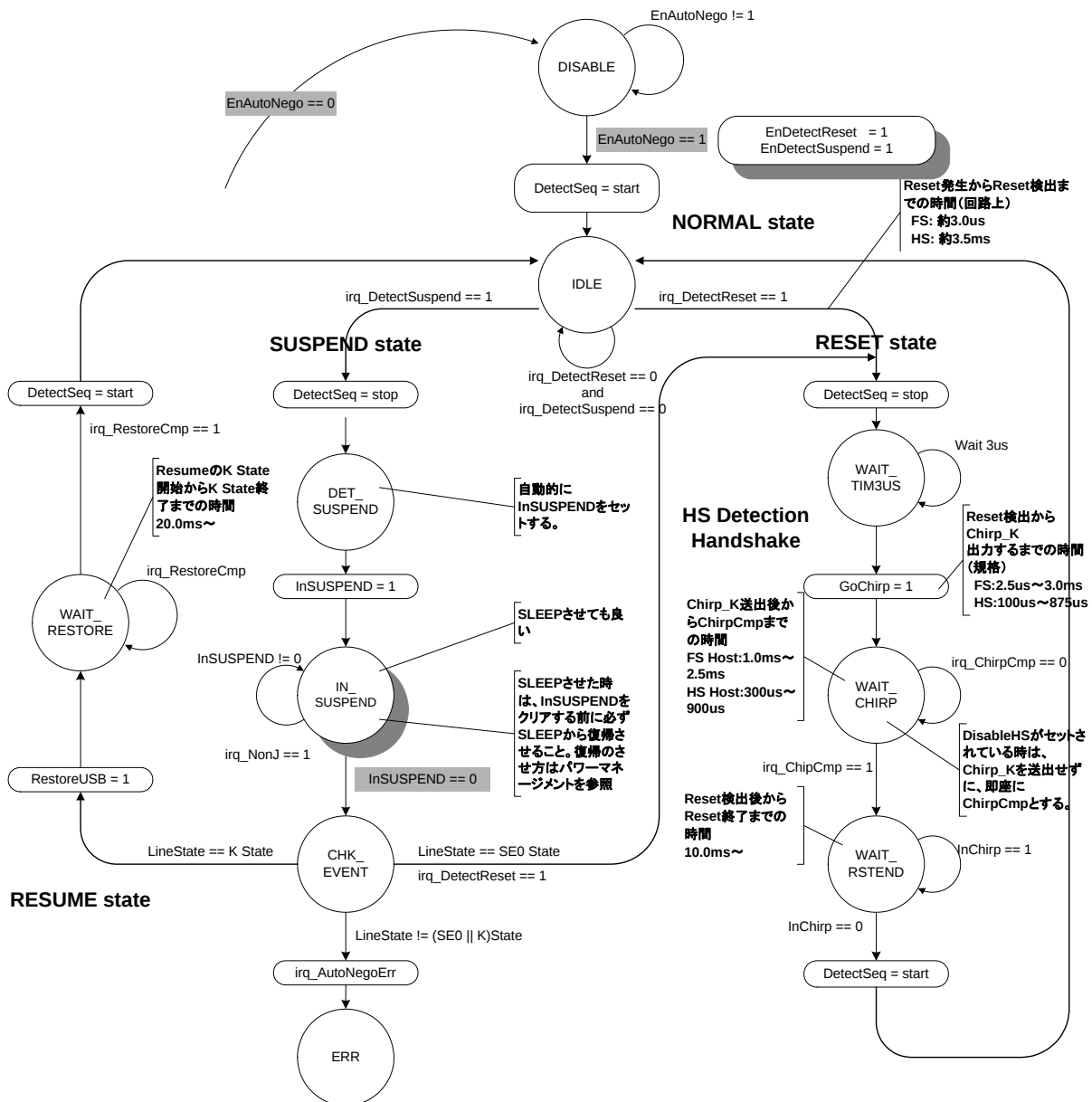


図 1.7 オート・ネゴシエータ

1.2.7.1 オート・ネゴシエータ

1.2.7.1.1 Disable

D_NegoControl.EnAutoNego ビットをクリアしている時に、このステートに入ります。

オート・ネゴシエータを有効にする時には、D_NegoControl.EnAutoNego ビットをセットする前に、リセット検出割り込み許可ビット(D_SIE_IntEnb.EnDetectRESET)、サスペンド検出割り込み許可ビット(D_SIE_IntEnb.EnDetectSUSPEND)をセットし、両イベント検出割り込みを許可してください。

オート・ネゴシエータを有効にすると、内部イベント検出機能を有効にします。オート・ネゴシエータを有効にしている間は、D_NegoControl.DisBusDetect ビットを絶対にセットしないでください。

1.2.7.1.2 IDLE

リセット検出待ち、サスペンド検出待ちをするステートです。

現在の USB スピードが HS の時には、USB バス上にバス・アクティビティが 3ms 以上検出できなかった場合に、一旦 FS のターミネーションを有効にし、FS-J が検出された場合はサスペンド、SE0 が検出された場合はリセットと判断します。現在のスピードが FS の時には、2.5us 以上の SE0 が検出された場合はリセット、3ms 以上バス・アクティビティが検出できなかった場合はサスペンドと判断します。これらの判断と同時に、リセット検出割り込み、またはサスペンド検出割り込みが発生し、D_SIE_IntStat.DetectRESET ビット、または D_SIE_IntStat.DetectSUSPEND ビットがセットされます。

サスペンドと判断した場合、一旦イベント検出機能を停止し、DET_SUSPEND ステートに入ります。

リセットと判断した場合、一旦イベント検出機能を停止し、WAIT_TIM3US ステートに入ります。

1.2.7.1.3 WAIT_TIM3US

リセット検出後、HS Detection Handshake を実行するまでの時間を調整しています。一定時間経過後(約 3us 後)、WAIT_CHIRP ステートに入ります。

1.2.7.1.4 WAIT_CHIRP

D_NegoControl.GoChirp ビットを自動的にセットし、HS Detection Handshake を実行します。HS Detection Handshake が終了すると、Chirp 終了割り込みステータス(D_SIE_IntStat.ChirpCmp)がセットされ、WAIT_RSTEND ステートに入ります。HS Detection Handshake の詳細については、「1.2.7.2.5 HS Detection Handshake」を参照してください。

また、D_NegoControl.DisableHS ビットをセットしている場合は、HS Detection Handshake を実行せずに、Chirp 終了割り込みステータス(D_SIE_IntStat.ChirpCmp)がセットされ、WAIT_RSTEND ステートに入ります。

なお、このステート終了後は、D_USB_Status.FSxHS ビットに設定された転送スピードにて動作することとなります。転送スピードが変化したことを検出する必要がある場合は、前述の Chirp 終了割り込みを有効にするために、D_SIE_IntEnb.EnChirpCmp ビットをセットしてください。

1. 機能説明

1.2.7.1.5 WAIT_RSTEND

リセット期間が終了するまで、このステートにて待ちます。HS 時はホストからの Chirp 送信(この IC にとっては受信)が終了したこと、FS 時は SE0 から J に遷移したことをもって、リセット期間の終了と判断します。

リセット期間終了と判断した後、イベント検出機能を有効にし、再度 IDLE ステートに入ります。

1.2.7.1.6 DET_SUSPEND

サスペンドと判断された場合に、自動的に D_NegoControl.InSUSPEND ビットがセットされ、IN_SUSPEND ステートに入ります。この D_NegoControl.InSUSPEND ビットによって、FS-J からのバスの遷移を検出する機能を有効にし、ホストからのレジューム及びリセットを検出できるようになります。

サスペンド中に実際に消費電流を軽減するかどうかは、アプリケーションに依存します。本 LSI では、パワーマネジメントにより、待機時の消費電流を軽減することが出来ます。パワーマネジメントの詳しい内容、及び、制御方法につきましては、「1.4 パワーマネジメント機能」を参照してください。

また、この時サスペンド終了指示であるレジューム(FS-K)を検出するために、NonJ 割り込みを許可してください。

1.2.7.1.7 IN_SUSPEND

NonJ 割り込みステータス(D_SIE_IntStat.NonJ)がセットされたら、サスペンドからの復帰指示であると判断して、D_NegoControl.InSUSPEND ビットを F/W にてクリアして下さい。パワーマネジメントにより、SLEEP にしている場合は、InSUSPEND をクリアするまえに、ACTIVE に戻して下さい。InSUSPEND がクリアされると、オート・ネゴシエータは CHK_EVENT ステートに入ります。

リモート・ウェイクアップ機能を有効にしているアプリケーションで、自発的にサスペンドから復帰する場合には、このステートの中で D_NegoControl.SendWakeup ビットをセットし、1ms 以上、15ms 以下の間 FS-K を出力してください。

1.2.7.1.8 CHK_EVENT

USB ケーブル上をチェックし、FS-K を検出した場合レジュームであると判断し、SE0 を検出した場合リセットであると判断します。レジュームと判断した場合は、D_NegoControl.RestoreUSB ビットをセットし、サスペンド前の転送スピード(D_USB_Status.FSxHS の値に従う)に戻ります。リセットと判断した場合は、IDLE ステートからの遷移と同じく、一旦イベント検出機能を停止し、WAIT_TIM3US ステートに入ります。

もし、FS-K でも SE0 でも無いステートを検出した場合には、オート・ネゴシエーション・エラー割り込みステータス(D_SIE_IntStat.AutoNegoErr)ビットをセットし、ERR ステートに入ります。

1.2.7.1.9 WAIT_RESTORE

D_SIE_IntStat.RestoreCmp ビットがセットされると、イベント検出機能を有効にし、IDLE ステートに入ります。

1.2.7.1.10 ERR

一旦このステートに遷移すると、オート・ネゴシエータを停止させない限り、このステートから抜けません。このステートは、USB 規格上在りえません。

なお、どのステートにおいても、USB ケーブルが抜かれたことによる判断を行っていないため、もし USB ケーブルが抜かれた場合には、すぐにオート・ネゴシエータを停止してください。

1.2.7.2 各バスイベント発生時の動作

各バスイベントの処理について、個別にその実施方法を説明します。オート・ネゴシエータは、本 LSI のレジスタインタフェースを通じて制御を行っていますので、オート・ネゴシエータで自動的に行われている処理を、f/w で実施することも出来ます。

1.2.7.2.1 HSモードにおけるサスペンド検出

HS モードでサスペンドを検出すると、DetectSUSPEND 割り込みステータスが発行され、オート・ネゴシエータは IN_SUSPEND ステートへ遷移します。この時、D_SIE_IntEnb.EnDetectSUSPEND ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat がセットされていると、XINT 信号がアサートされます。

オート・ネゴシエータは、HS モードで 3ms 以上送受信が何も検出されなかった場合 (T1)、FS モードに自動的に移行します (HS のターミネーションを無効にし、FS のターミネーション (Rpu) を有効にします)。この動作により DP は“H”になり、D_USB_Status.LineState[1:0] ビットで“J”を確認することができます (もし、“SE0”を検出した場合は、リセット (後述) となることに注意)。その後 T2 の時点で依然“J”が検出された場合、D_SIE_IntStat.DetectSUSPEND ビットがセットされますので、USB のサスペンドステートであると判断します。

以下の図では、USB サスペンド中にスリープを行う時の動作を表しています。

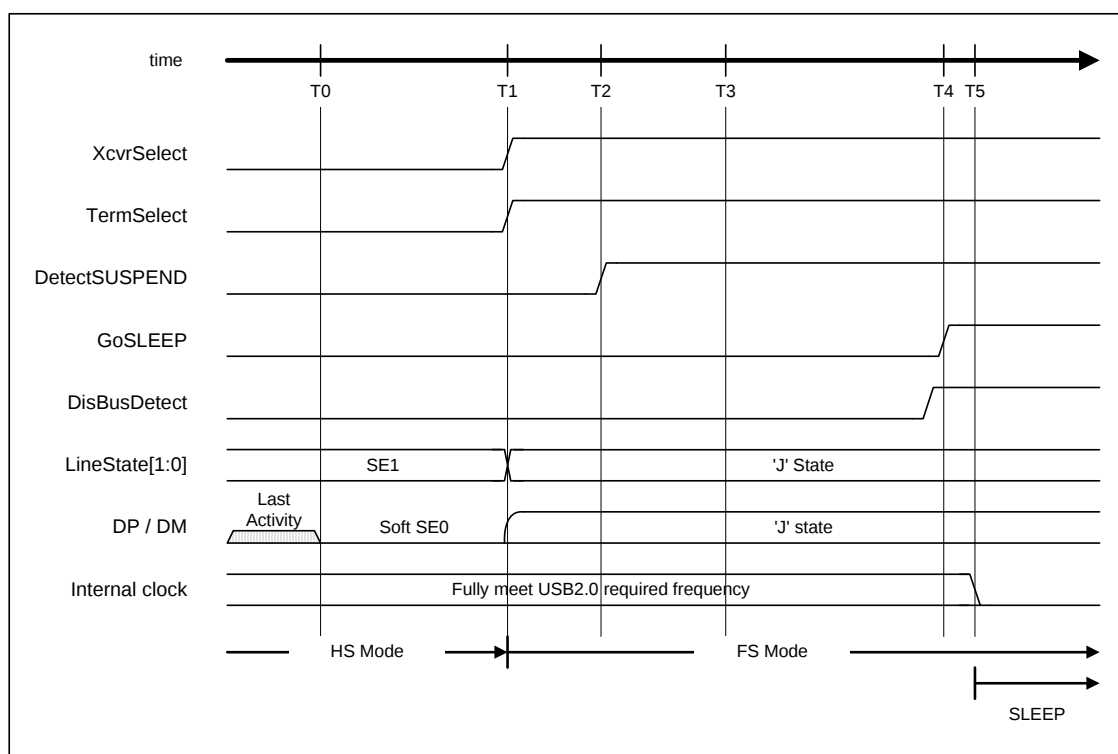


図 1.8 Suspend Timing (HS mode)

1. 機能説明

表 1.10 Suspend Timing Values (HS mode)

Timing Parameter	Description	Value
T0	最後のバスアクティビティ。	0 (reference)
T1	この時点で依然バスアクティビティが無い場合、XcvrSelect、TermSelect を“1”にセットし、HS モードから FS モードに切り替える。	HS Reset T0 + 3.0ms < T1 { T_{WTREV} } < HS Reset T0 + 3.125ms
T2	LineState[1:0] をサンプリングする。この時“J”なら、DetectSUSPEND が“1”になり、USB のサスペンドステートと判断する。	$T1 + 100\mu s < T2 \{T_{WTWRSTHS}\} < T1 + 875\mu s$
T3	これより前では、RESUME を発行してはいけない。	HS Reset T0 + 5ms { T_{WTRSM} }
T4	GoSLEEP を“1”にセットし、完全にスリープに移行。これ以降は VBUS から USB で規定されたサスペンド電流以上を消費してはいけない。 SLEEP 移行前に、DisBusDetect を“1”にセットし、再び SUSPEND が検出されることを防ぐ。	HS Reset T0 + 10ms { T_{2SUSP} }
T5	内部クロックが完全停止する。	$T5 < T4 + 10\mu s$

注： {} は、USB2.0 規格書で規格されている名称である。

1.2.7.2.2 FSモードにおけるサスペンド検出

FS モードでサスペンドを検出すると、DetectSUSPEND 割り込みステータスが発行され、オート・ネゴシエータは IN_SUSPEND ステートへ遷移します。この時、D_SIE_IntEnb.EnDetectSUSPEND ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat がセットされていると、XINT 信号がアサートされます。

オート・ネゴシエータは、3ms 以上送受信が何も検出されなかった場合、または D_USB_Status.LineState[1:0] ビットに“J”を検出し続け(T1)、さらに T2 の時点で依然“J”が検出された場合、USB のサスペンドステートであると判断し、SIE_IntStat.DetectSUSPEND ビットをセットします。

以下の図では、USB サスペンド中にスリープを行う時の動作を表しています。

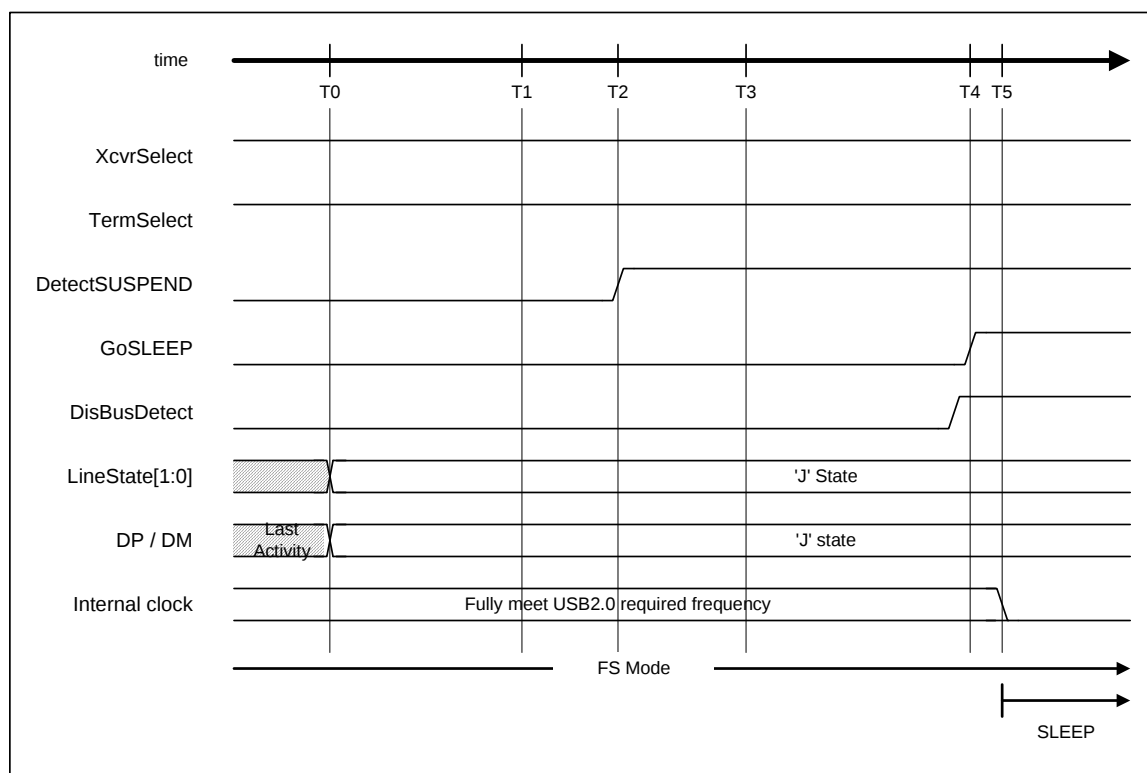


図 1.9 Suspend Timing (FS mode)

1. 機能説明

表 1.11 Suspend Timing Values (FS mode)

Timing Parameter	Description	Value
T0	最後のバスアクティビティ。	0 (reference)
T1	この時点で依然バスアクティビティが無い。	$T0 + 3.0\text{ms} < T1 \{T_{WTREV}\} < T0 + 3.125\text{ms}$
T2	LineState[1:0] をサンプリングする。この時 “J” なら、DetectSUSPEND が “1” になり、USB のサスペンドステートと判断する。	$T1 + 100\mu\text{s} < T2 \{T_{WTWRSTHS}\} < T1 + 875\mu\text{s}$
T3	これより前では、RESUME を発行してはいけない。	$T0 + 5\text{ms} \{T_{WTRSM}\}$
T4	GoSLEEP を “1” にセットし、完全に SLEEP に移行。これ以降はVBUS から USB で規定されたサスペンド電流以上を消費してはいけない。 スリープ移行前に、DisBusDetect を “1” にセットし、再びSUSPEND が検出されることを防ぐ。	$T0 + 10\text{ms} \{T_{2SUSP}\}$
T5	内部クロックが完全停止する。	$T5 < T4 + 10\mu\text{s}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.2.7.2.3 HSモードにおけるリセット検出

HS モードでリセットを検出すると、DetectRESET 割り込みステータスが発行され、オート・ネゴシエータはリセットシーケンスを行います。この時、D_SIE_IntEnb.EnDetectRESET ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat がセットされていると、XINT 信号がアサートされます。

オート・ネゴシエータは、3ms 以上送受信が何も検出されなかった場合、FS モードに自動的に移行します(HS のターミネーションを無効にし、FS のターミネーション(Rpu)を有効にします)。リセットされている場合、この動作が行われても DP ラインは“L”になったままで、この結果 D_USB_Status.LineState[1:0]ビットでも“SE0”を検出することができます。T2 の時点で依然“SE0”が検出された場合にはリセットとみなし、D_SIE_IntStat.DetectRESET ビットをセットします。以降は、D_NegoControl.DisBusDetect ビットをセットした後に、HS Detection Handshake(後述)を行います。

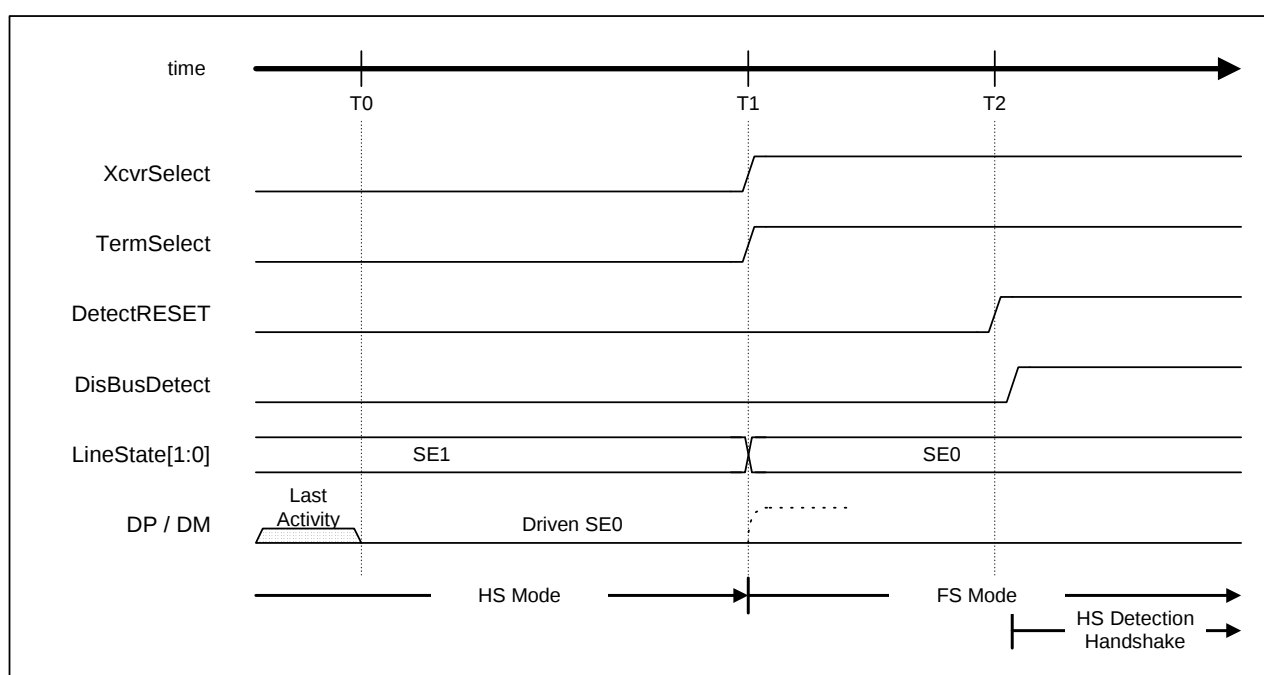


図 1.10 Reset Timing (HS mode)

表 1.12 Reset Timing Values (HS mode)

Timing Parameter	Description	Value
T0	最後のバスアクティビティ。	0 (reference)
T1	この時点で依然バスアクティビティが無い場合、XcvtSelect、TermSelect を“1”にセットし、HS モードから FS モードに切り替える。	HS Reset T0 + 3.0ms < T1 {T _{WTREV} } < HS Reset T0 + 3.125ms
T2	LineState[1:0] をサンプリングする。この時“SE0”なら、DetectRESET が“1”になり、リセットへの移行と判断する。リセット指示の検出後、DisBusDetect を“1”にセットし、以降 HS Detection Handshake を行う。	T1 + 100us < T2 {T _{WTWRSTHS} } < T1 + 875us

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.2.7.2.4 FSモードにおけるリセット検出

FS モードでリセットを検出すると、DetectRESET 割り込みステータスが発行され、オート・ネゴシエータはリセットシーケンスを行います。この時、D_SIE_IntEnb.EnDetectRESET ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat がセットされていると、XINT 信号がアサートされます。

オート・ネゴシエータは、2.5us 以上 D_USB_Status.LineState[1:0]ビットに“SE0”を検出し続けた場合には(T1)、リセットとみなし、D_SIE_IntStat.DetectRESET ビットをセットします。以降は D_NegoControl.DisBusDetect ビットをセットした後に、HS Detection Handshake(後述)を行います。

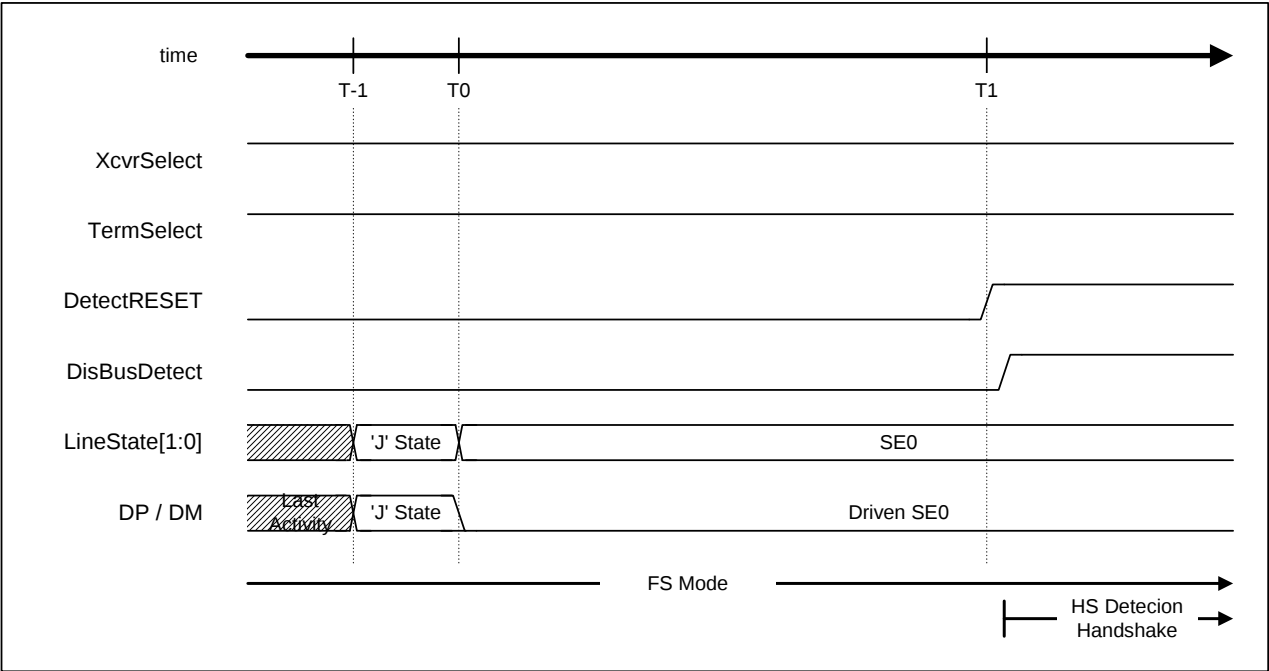


図 1.11 Reset Timing (FS mode)

表 1.13 Reset Timing Values (FS mode)

Timing Parameter	Description	Value
T-1	最後のバスアクティビティ。	
T0	ホスト Downstream port からのリセットの指示開始。	0 (reference)
T1	“SE0”が継続されている場合、DetectRESET が“1”になり、リセットへの移行と判断する。 リセット指示の検出後、DisBusDetect を“1”にセットし、以降 HS Detection Handshake を行う。	HS Reset T0 + 2.5us < T1 {T _{WTREV} }

注： {} は、USB2.0 規格書で規格されている名称である。

1.2.7.2.5 HS Detection Handshake

オート・ネゴシエータはリセットを検知すると、HS Detection Handshake を実行します。HS Detection Handshake が終了すると、D_SIE_IntStat.ChirpCmp 割り込みステータスがセットされます。この時、D_SIE_IntEnb.EnChirpCmp ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat がセットされている場合には、同時に XINT 信号がアサートされます。

HS Detection Handshake は、サスペンド中、FS 動作中、或いは HS 動作中の 3 状態のいずれかから、ホスト Downstream port からの“SE0”のアサートにより開始されます(上記状態からのリセットが開始された時)。詳細は、USB2.0 規格書を参照してください。

ここで、上記 3 状態からオート・ネゴシエータが HS Detection Handshake に移行する方法について説明します。

本 LSI がサスペンド状態では、バス上に“SE0”を検出すると直ちにリセットとみなし、HS Detection Handshake に移行します。

本 LSI が FS モードで動作している状態では、2.5us 以上の“SE0”を検出するとリセットとみなし、HS Detection Handshake に移行します。

本 LSI が HS モードで動作している状態では、3.0ms 以上の“SE0”を検出後、まず USB のサスペンドステートなのかリセットなのかを判断しなければならない為、一旦 FS モードに切り替わります。この時動作としては、D_XcvrControl.XcvrSelect、D_XcvrControl.TermSelect の両ビットを FS モードに切り替え、HS ターミネーションを無効にし、FS ターミネーションを有効にします。これらのモード切り替えは、h/w によって 3.125ms 以内に行われます。h/w は、このモード切り替えから 100us 以上 875us 以内に D_USB_Status.LineState[1:0] ビットをチェックし、“J”なら USB のサスペンドステートとして判断し、“SE0”ならリセットと判断します。この時、リセットと判断された場合には、その後 HS Detection Handshake に移行します。

いずれの場合も、リセットは最小 10ms 存在しますが、移行する前の状態(HS もしくは FS)により、タイミングが多少異なります。ここでは、リセットが開始された時間を“HS Reset T0”として定義し、以降は、この“HS Reset T0”からの動作について説明します。

動作中の場合は内部クロックも安定して発振していますが、サスペンド中にスリープさせていた場合には、リセット検出時には内部クロックが出力されていません。このため HS Detection Handshake を行うために、必ず PM_Control.GoACTIVE ビットを“1”にセットし、内部クロックを発振開始させてください。この動作の詳細は、「1.4 パワーマネジメント機能」を参照してください。

1.2.7.2.5.1 FSのホストダウンストリームポートに繋がれた場合

本 LSI が、HS をサポートしていないホスト downstream port に接続された時の HS Detection Handshake の動作を示します。HS Detection Handshake の開始時(T0)では、D_XcvtControl.XcvtSelect と D_XcvtControl.TermSelect は両ビットともに FS モードになっています。(FS ターミネーション、即ち DP のプルアップ抵抗(Rpu)が有効に、HS ターミネーションが無効にされています)。

オート・ネゴシエータは、まず D_NegoControl.GoChirp ビットをセットします。すると D_XcvtControl.OpMode[1:0] ビットが“Disable Bit Stuffing and NRZI encoding”になり、全て“0”のデータが準備されます(T1)。これは、バス上に“HS K”(Device Chirp) を送出するためのものです。また同時に、D_XcvtControl.XcvtSelect ビットが HS モードに設定され、かつ送信可能状態に設定されることで、ホスト downstream port に“HS K”(Device Chirp)が送出されます。送出終了後、ホスト downstream port からの Host Chirp を待ちます(T2)。通常、HS をサポートしているホスト downstream port は、T3 から“HS K”, “HS J”を連続的に送出してきますが(後述)、ホスト downstream port が HS をサポートしていない場合(本例の場合)は、T4 の時点でも Host Chirp を送出してこないため、D_XcvtControl.XcvtSelect ビットを FS モードに自動的に切り替え、D_NegoControl.GoChirp ビットがクリアされるとともに D_USB_Status.FSxHS ビットがセットされ、さらに D_SIE_IntStat.ChirpCmp ビットがセットされます。

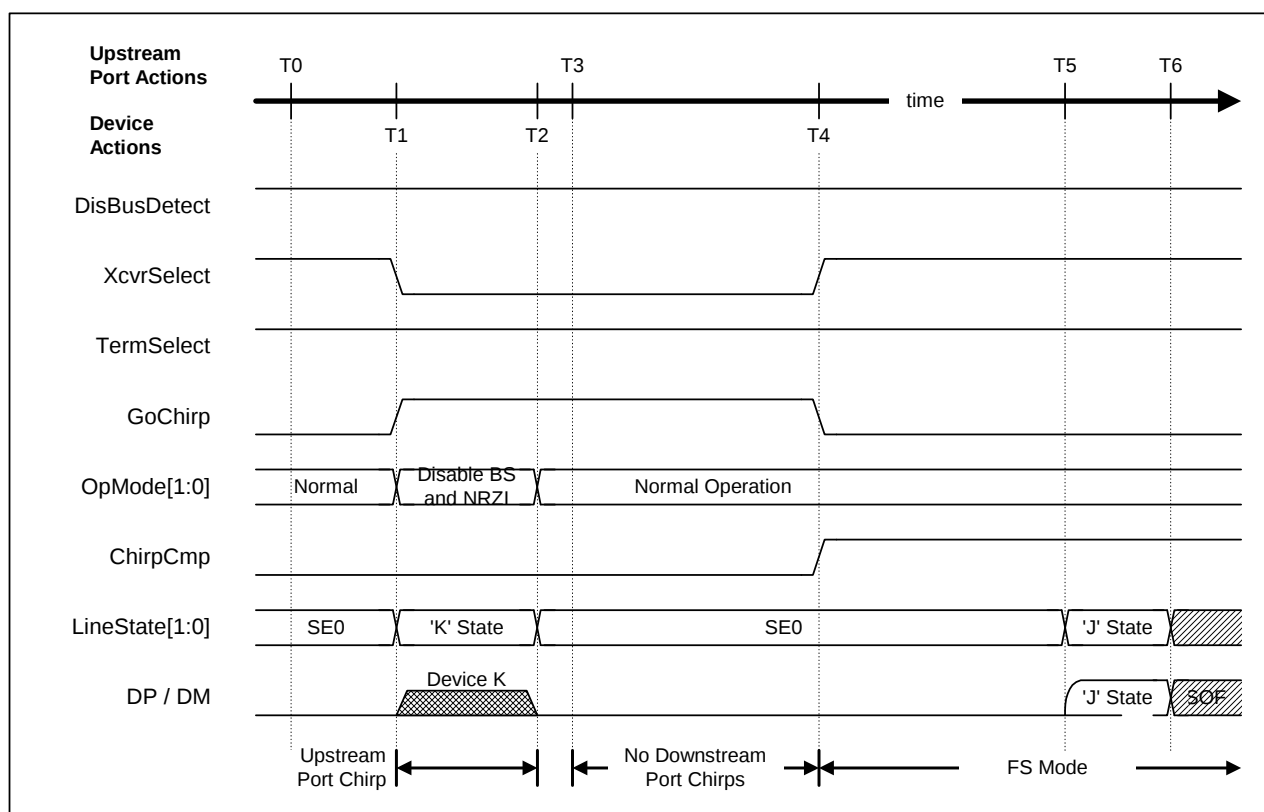


図 1.12 HS Detection Handshake Timing (FS mode)

表 1.14 HS Detection Handshake Timing Values (FS mode)

Timing Parameter	Description	Value
T0	HS Detection Handshake 開始。	0 (reference)
T1	HS トランシーバをイネーブルにし、GoChirp を“1”にセットして、Chirp K を送出開始。	$T0 < T1 < \text{HS Reset } T0 + 6.0\text{ms}$
T2	Chirp K 送出終了。最小 1ms の間は送出しなければならない。	$T1 + 1.0\text{ms } \{T_{\text{UCH}}\} < T2 < \text{HS Reset } T0 + 7.0\text{ms } \{T_{\text{UCHEND}}\}$
T3	ホスト downstream port が HS をサポートしている場合、ここから Chirp K を送出開始する。	$T2 < T3 < T2 + 100\mu\text{s } \{T_{\text{WTDCH}}\}$
T4	Chirp を検出出来ない場合、この時点で FS モードに戻り、ChirpCmp が“1”にセットされ、リセットシーケンスが終了するのを待つ。	$T2 + 1.0\text{ms} < T4 \{T_{\text{WTFS}}\} < T2 + 2.5\text{ms}$
T5	リセットシーケンスの終了。	$\text{HS Reset } T0 + 10\text{ms } \{T_{\text{DRST}} (\text{Min})\}$
T6	FS モードでの通常動作。	T6

注： {} は、USB2.0 規格書で規格されている名称である。

注： 最小 1ms の Chirp K を生成するために、66000 サイクル(内部クロック：60MHz)で判断する。

1.2.7.2.5.2 HSのホストダウンストリームポートに繋がれた場合

本 LSI が、HS をサポートしているホスト downstream port に接続された時の動作を示します。HS Detection Handshake の開始時 (T0) では、D_XcvrControl.XcvrSelect と D_XcvrControl.TermSelect は両ビットともに FS モードになっています (FS ターミネーション、即ち DP のプルアップ抵抗(Rpu) が有効に、HS ターミネーションが無効にされています)。

オート・ネゴシエータは、まず D_NegoControl.GoChirp ビットをセットします。すると D_XcvrControl.OpMode[1:0] ビットが "Disable Bit Stuffing and NRZI encoding" になり、全て "0" のデータが準備されます (T1)。これは、バス上に "HS K" (Device Chirp) を送出するためのものです。また同時に、D_XcvrControl.XcvrSelect ビットが HS モードに設定され、かつ送信可能状態に設定されることで、downstream port に "HS K" (Device Chirp) が送出されます。送出終了後、downstream port からの Host Chirp を待ちます (T2)。ここでは downstream port は HS をサポートしているので、"HS K" (Chirp K)、"HS J" (Chirp J) を交互に連続して送出してきます (T3)。この状態を USB_Status.LineState[1:0] ビットで Chirp K-J-K-J-K-J と最低 6 回検出したところで (T6)、XcvrControl.TermSelect ビットを HS モードに自動的に切り替え (T7)、完全な HS モードに移行します。これと同時に、D_NegoControl.GoChirp ビットがクリアされるとともに D_NegoStatus.FSxHS ビットがクリアされ、さらに D_SIE_IntStat.ChirpCmp ビットがセットされます。

このホスト downstream port からの Chirp K、Chirp J はバス・アクティビティとして認識し、USB のサスペンドステートと判断しないようにしなければなりません。そこで、HS モードでは、この Chirp K、Chirp J を逐次検出し、内部の Suspend Timer に取り込んでいます。

なお、Chirp K-J-K-J-K-J を検出する為に、USB_Status.LineState[1:0] ビットを使用しています。通常の HS パケットと違い、Chirp K、Chirp J は非常に遅い為、D_USB_Status.LineState[1:0] ビットを使用できます。しかし、本来のパケット受信時に D_USB_Status.LineState[1:0] ビットにバスの信号を載せると非常にノイズな為、D_XcvrControl.TermSelect ビットが HS モードの時、バス・アクティビティがあると判断される場合には、D_USB_Status.LineState[1:0] ビットは "J" を、バス・アクティビティが無いと判断される場合には "SE0" を出力するようになっています。

次図で、T6 の時点から Chirp の高さが変わっているのは、D_XcvrControl.TermSelect ビットによりデバイス側の HS ターミネーションが有効になったことを表しています。通常、D_XcvrControl.TermSelect が FS モード時の Chirp は約 800mV、D_XcvrControl.TermSelect ビットが HS モード時の Chirp (HS の通常送受信パケットも同様) では、約 400mV となります。

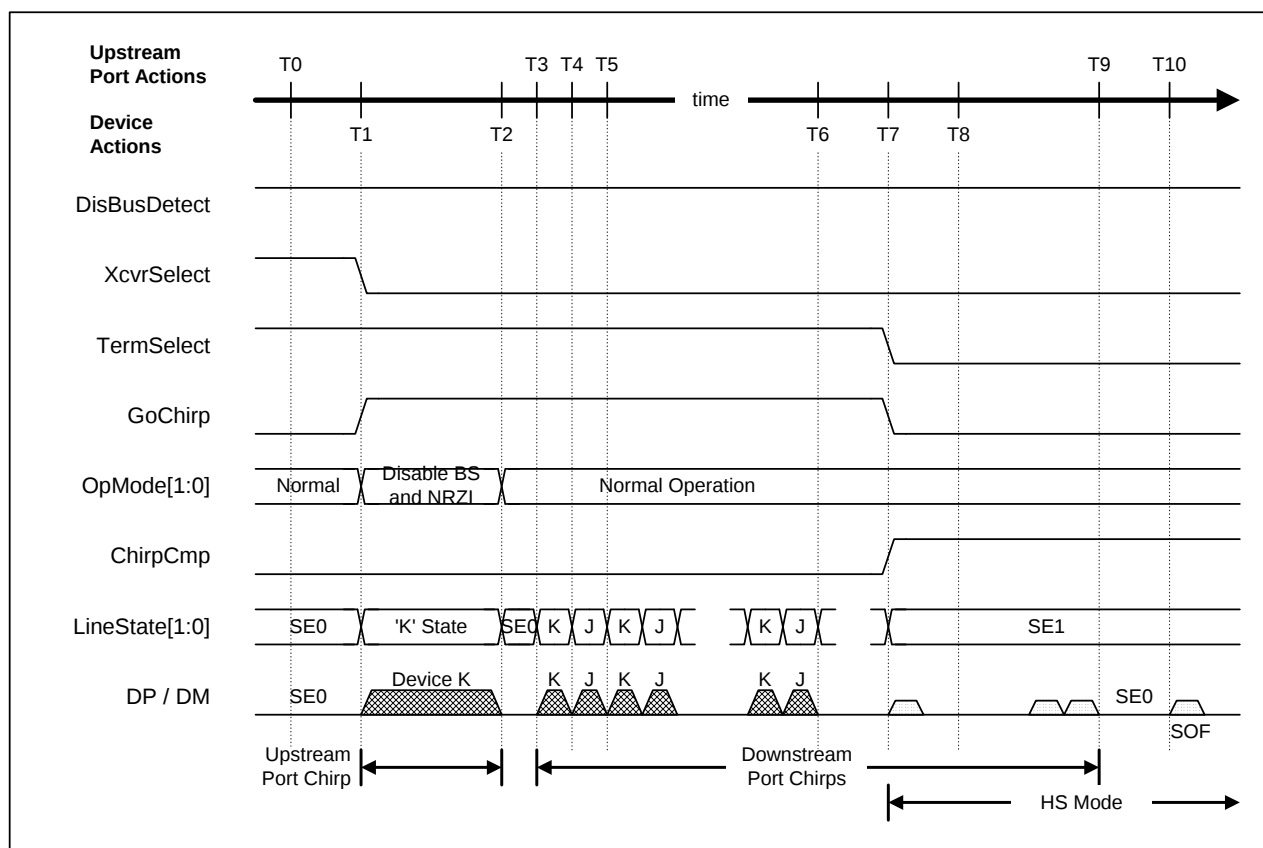


図 1.13 HS Detection Handshake Timing (HS mode)

表 1.15 HS Detection Handshake Timing Values (HS mode)

Timing Parameter	Description	Value
T0	HS Detection Handshake 開始。	0 (reference)
T1	HS トランシーバをイネーブルにし、GoChirp を“1”にセットして、Chirp K を送出開始。	$T0 < T1 < \text{HS Reset } T0 + 6.0\text{ms}$
T2	Chirp K 送出終了。最小 1ms の間は送出しなければならない。	$T1 + 1.0\text{ms} \{T_{\text{UCH}}\} < T2 < \text{HS Reset } T0 + 7.0\text{ms} \{T_{\text{UCHEND}}\}$
T3	ホスト downstream port が最初の Chirp K をバスに送出。	$T2 < T3 < T2 + 100\mu\text{s} \{T_{\text{WTDCH}}\}$
T4	ホスト downstream port が Chirp K から Chirp J に切り替え送出。	$T3 + 40\mu\text{s} \{T_{\text{DCHBIT}} (\text{Min})\} < T4 < T3 + 60\mu\text{s} \{T_{\text{DCHBIT}} (\text{Max})\}$
T5	ホスト downstream port が Chirp J から Chirp K に切り替え送出。	$T4 + 40\mu\text{s} \{T_{\text{DCHBIT}} (\text{Min})\} < T5 < T4 + 60\mu\text{s} \{T_{\text{DCHBIT}} (\text{Max})\}$
T6	Chirp K-J-K-J-K-J を検出。	T6
T7	Chirp K-J-K-J-K-J を検出したことを受けて、FS ターミネーションを無効に、HS ターミネーションを有効にする。ChirpCmp が“1”にセットされる。さらに、リセットの終了を待つ。	$T6 < T7 < T6 + 500\mu\text{s}$
T8	Chirp K、Chirp J によりバスアクティビティと認識される。ただし SYNC が検出できない為、パケット受信中和認識されることは無い。	T8
T9	ホスト downstream port からの Chirp K、Chirp J の送出終了。	$T10 - 500\mu\text{s} \{T_{\text{DCHSE0}} (\text{Max})\} < T9 < T10 - 100\mu\text{s} \{T_{\text{DCHSE0}} (\text{Min})\}$
T10	リセットシーケンスの終了。	$\text{HS Reset } T0 + 10\text{ms} \{T_{\text{DRST}} (\text{Min})\}$

注： {} は、USB2.0 規格書で規格されている名称である。

注： 最小 1ms の Chirp K を生成するために、66000 サイクル(内部クロック：60MHz)で判断する。

1. 機能説明

1.2.7.2.5.3 スリープ中にリセットされた場合

本 LSI は、スリープ状態では PLL による内部システムクロックは出力されていません。ここでは、スリープ状態でリセットを検出した場合の動作復帰について述べます。

スリープ状態でリセットが検出された場合(T0)、D_SIE_IntStat.NonJ ビットがセットされます。さらに、D_SIE_IntEnb.EnNonJ ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat ビットがセットされている場合には、同時に XINT 信号がアサートされます。この時、すぐさまスヌーズから復帰しリセットシーケンスに移行させる為に、f/w により、PM_Control.GoACTIVE ビットを“1”にセットしてください(T1)。PLL パワーアップ時間経過後(T2)、PM_Control.PM_State[1:0]が“ACTIVE”になり、内部クロックが出力され始めます。

この後、InSUSPEND がクリアされると、オート・ネゴシエータによって HS Detection Handshake(前述)が行われます。

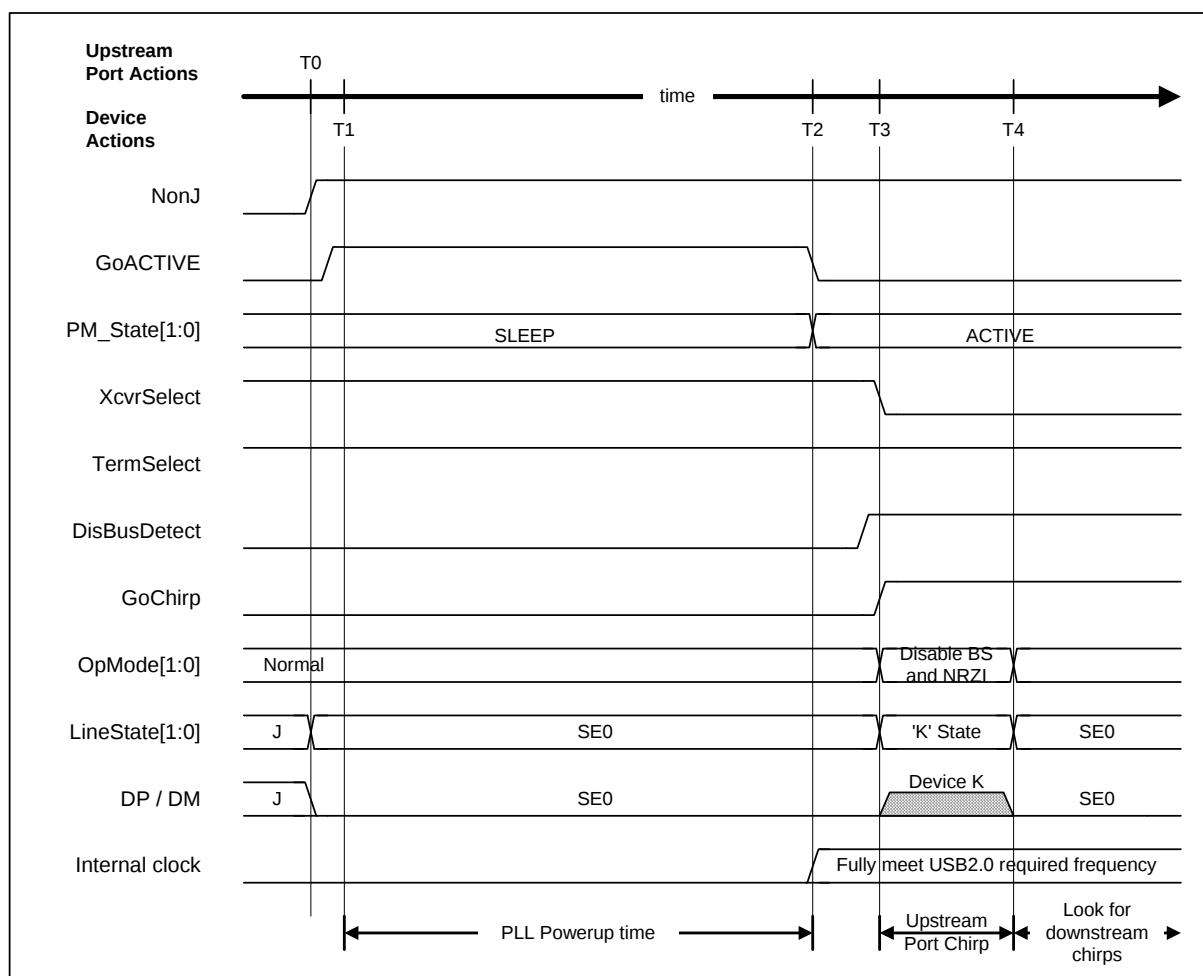


図 1.14 HS Detection Handshake Timing from Suspend

表 1.16 HS Detection Handshake Timing Values from Suspend

Timing Parameter	Description	Value
T0	NonJ が“1”にセットされ、LineState[1:0]で“SE0”を確認すると、スヌーズ中のリセットを検出。	0 (HS Reset T0)
T1	リセット検出後、GoActDevice を“1”にセット。	T1
T2	PM_State が“ACT_DEVICE”になる。内部クロック出力安定。	$T1 + 250\mu s < T2$
T3	GoChirp を“1”にセットし、chirp K をバスに送出。 (chirp K 送出前には DisBusDetect を“1”にセットする)	$T2 < T3 < \text{HS Reset T0} + 5.8\text{ms}$
T4	chirp K 送出を終了。	$T3 + 1.0\text{ms} \{T_{\text{UCH}}\} < T4 < \text{HS Reset T0} + 7.0\text{ms} \{T_{\text{UCHEND}}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

注： 最小 1ms の Chirp K を生成するために、66000 サイクル(内部クロック：60MHz)で判断する。

注： 発振回路も停止していた場合(スリープ状態)は、後述する(PLL パワーアップ時間の他に、OSC パワーアップ時間が必要)。

1.2.7.2.6 レジュームの発行

ここでは、リモート・ウェイクアップがサポートされていて、かつホストからこのリモート・ウェイクアップを有効にされている時に、何らかの要因で、自らレジュームする方法を説明します。ただしリモート・ウェイクアップを行うことが可能なのは、バスがアイドルになってから少なくとも 5ms 経過してからでなくてはなりません。さらに、レジューム信号を出力してから 10ms 経過以前は、USB のサスペンド状態に入る前の電流を VBUS から消費することはできません。

オート・ネゴシエータはリモート・ウェイクアップをサポートしていませんので、f/w でリモート・ウェイクアップを起動する必要があります。レジューム信号が発行された後は、通常の SUSPEND からの復帰と同じシーケンスになります。f/w は、D_NegoControl.SendWakeup ビットをセットして、レジューム信号を送信し、InSUSPEND をクリアします。また、送出開始から 1ms 後に、D_NegoControl.SendWakeup ビットをクリアして、レジューム信号の送信を止めます。レジュームが完了すると、D_SIE_IntStat.RestoreCmp ビットがセットされます。この時、D_SIE_IntEnb.EnRestoreCmp ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat ビットがセットされている場合には、同時に XINT 信号がアサートされます。なお、リモート・ウェイクアップする為には、まずスリープから復帰させている必要があります。D_SIE_IntEnb.EnNonJ ビットをクリアし、PM_Control.GoACTIVE ビットをセットし(T0)、PLL パワーアップ時間経過後(T1)、PM_Control.PM_State[1:0]ビットが"ACTIVE"になると同時に内部クロックが出力され始めます。

D_NegoControl.SendWakeup ビットをセットすると、レジューム信号を送出します(T2)。この時、h/w は、D_XcvtControl.OpMode[1:0]を“Disable Bit Stuffing and NRZI encoding”に設定し、送信データとして“0”を準備し、パケット送信状態にして、“K” (Resume 信号)を送出します。ホスト downstream port は、このレジューム信号を検出し、バス上に“K” (レジューム信号)を返してきます(T3)。D_NegoControl.SendWakeup ビットをクリアすることによってバスに送出していたレジューム信号が停止されますが(T4)、この時点ではホスト downstream port が依然バスをレジューム信号にホールドしています。

オート・ネゴシエータは、レジューム信号を検知して、D_NegoControl.RestoreUSB ビットをセットします。一定時間経過後、ホスト downstream port はレジューム信号の送出を停止し(T5)、2 ビットの LS-EOP(2*SE0)を送出し、USB のサスペンド以前のスピードモードに切り替わります。これを検出した(“K”でなくなった)ところで、D_XcvtControl.XcvtSelect、D_XcvtControl.TermSelect の両ビットが所望のモード(今回の場合 HS モード)に切り替えられ、D_NegoControl.RestoreUSB ビットがクリアされます。

ここで、USB のサスペンド開始時には、スピードモード(HS 又は FS)を、USB_Status.FSxHS ビットにて保存しており、レジュームにて復帰する場合には、この D_USB_Status.FSxHS ビットが示すモードに戻ります。この時、レジュームごとに HS Detection Handshake は行いません。ここでは USB のサスペンド以前のモードが HS モードであった場合についてのみ説明していることに注意してください。FS モードであった場合には、T5 以降が通常の FS モードとなり、特に大きなシーケンスの違いはありません。

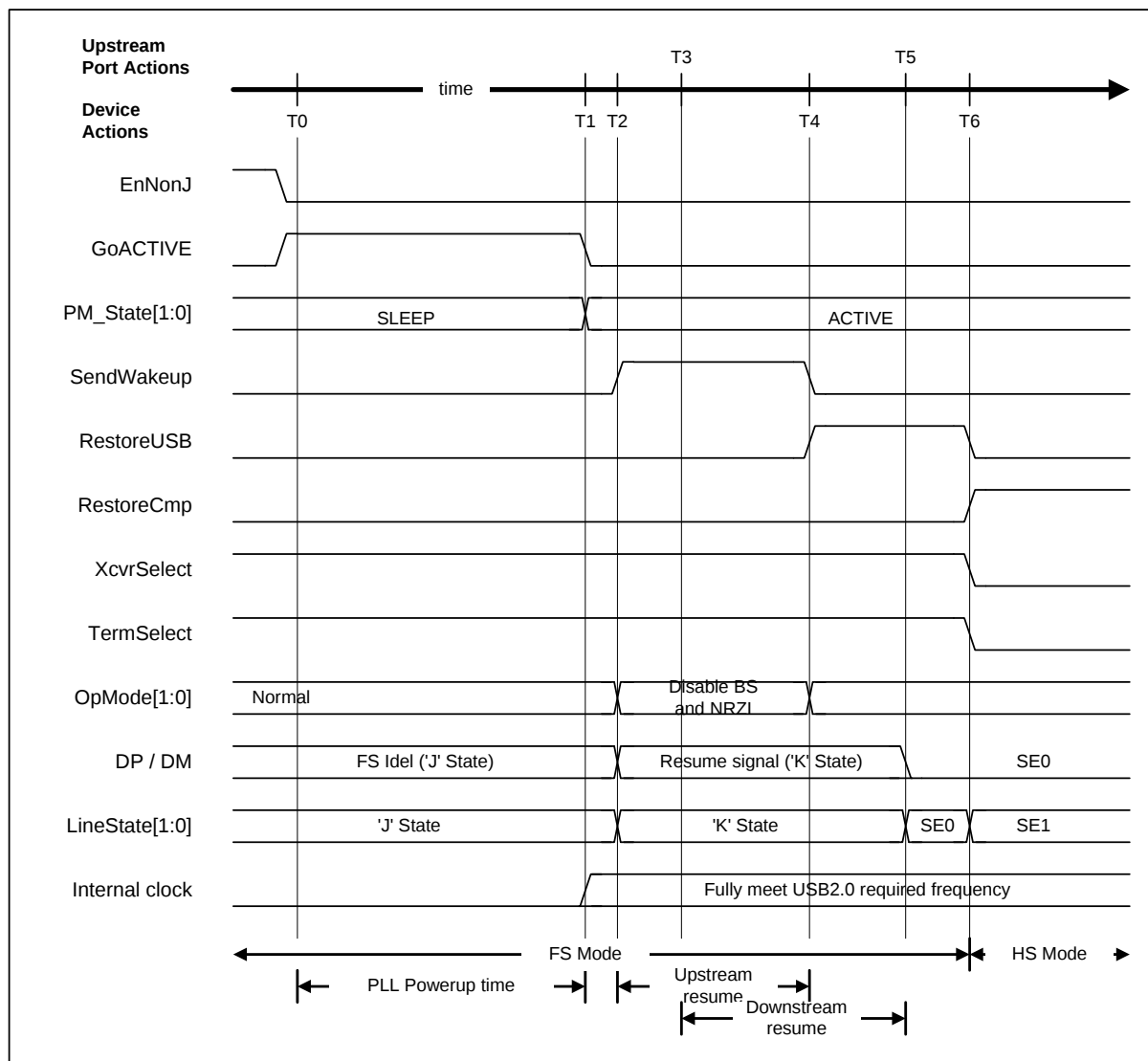


図 1.15 Assert Resume Timing (HS mode)

表 1.17 Assert Resume Timing Values (HS mode)

Timing Parameter	Description	Value
T0	レジューム開始。GoACTIVE を“1”にセット。 (レジューム開始前に EnNonJ を“0”にクリアすること)	0 (reference)
T1	PM_State が“ACTIVE”になる。内部クロック出力の安定。	$T0 + 250\mu s < T1$
T2	SendWakeup を“1”にセットし、FSの“K”を送出開始。ここで、10ms 以内は USB のサスペンド以前の電流を引っ張ってはいけない。	$T0 < T2 < T0 + 10ms$
T3	ホスト downstream port が FS の“K”を返す。	$T2 < T3 < T2 + 1.0ms$
T4	SendWakeup を“0”にクリアし、FS の“K”送を終了。 LineState[1:0]により“K”を確認後、RestoreUSB を“1”にセットする。	$T2 + 1.0ms \{T_{DRSMUP} (Min)\} < T4 < T2 + 15ms \{T_{DRSMUP} (Max)\}$
T5	ホスト downstream port が FS の“K”送を終了。	$T2 + 20ms \{T_{DRSMDN}\}$
T6	RestoreCmp が“1”になる。USB のサスペンド以前が HS モードで あった場合、自動的に HS モードに移行。	$T5 + 1.33\mu s \{2 \text{ Low-speed bit times}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.2.7.2.7 レジュームの検出

サスペンド中にレジュームを検出すると、NonJ 割り込みステータスが発行されます。この時、D_SIE_IntEnb.EnNonJ ビットおよび USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat ビットがセットされている場合には、同時に XINT 信号がアサートされます。NonJ を検知したら、f/w により InSUSPEND をクリアして、IN_SUSPEND ステートを抜けて下さい。また、サスペンド中に、SLEEP させていた場合は、InSUSPEND をクリアするまえに、ACTIVE に戻す必要があります。PM_Control.GoACTIVE ビットを“1”にセットし(T1)、PLL パワーアップ時間経過後(T2)、PM_Control.PM_State[1:0]が"ACTIVE"になると同時に内部クロックが出力され始めます。レジュームを完了すると、D_SIE_IntStat.RestoreCmp 割り込みステータスが発行されます。この時、D_SIE_IntEnb.EnRestoreCmp ビットと USB_DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnUSB_DeviceIntStat ビットがセットされている場合には、同時に XINT 信号がアサートされます。

サスペンド中、バス上には“J”(D_USB_Status.LineState[1:0]は“J”)が観測されます。バス上に“K”が観測された時は、ホスト downstream port からのウェイクアップの指示(レジューム指示)を受け取ったこととなります(T0)。この時、SIE_IntStat.NonJ ビットがセットされます。

オート・ネゴシエータは、IN_SUSPEND を抜けると、D_NegoControl.RestoreUSB をセットします。一定時間経過後、ホスト downstream port はレジューム信号の送出手を停止し(T3)、USB のサスペンド以前のスピードモードに切り替わります。これを検出した(“K”でなくなった)ところで、D_XcvrControl.XcvrSelect、D_XcvrControl.TermSelect の両ビットが所望のモード(今回の場合 HS モード)に切り替えられ、D_NegoControl.RestoreUSB ビットがクリアされるとともに D_SIE_IntStat.RestoreCmp ビットがセットされます。この時、D_SIE_IntEnb.EnRestoreCmp ビットがセットされます。

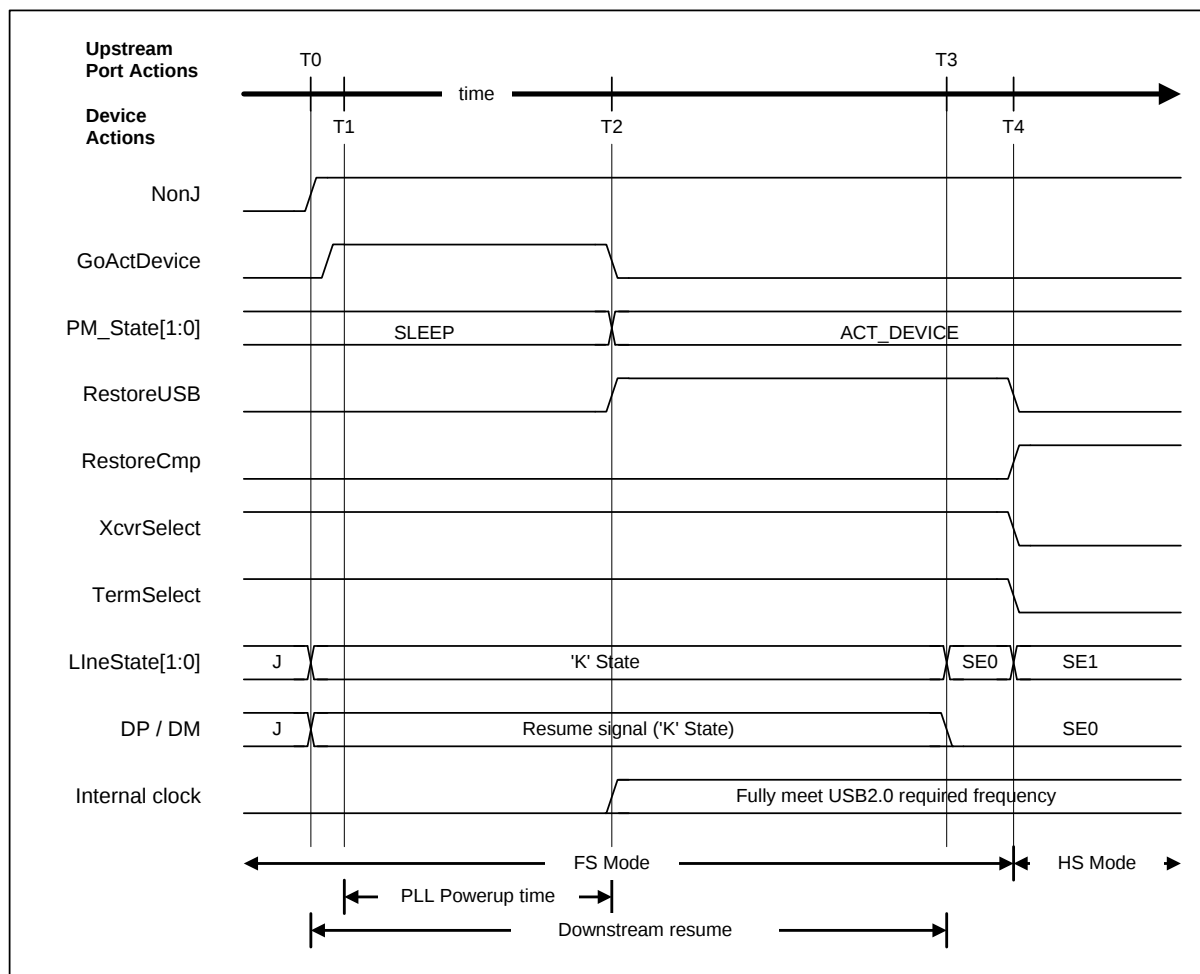


図 1.16 Detect Resume Timing (HS mode)

表 1.18 Detect Resume Timing Values (HS mode)

Timing Parameter	Description	Value
T0	ホスト downstream port が FS の“K”を送出。NonJ が“1”になる。	0 (reference)
T1	GoACTIVE を“1”にセット。	T1
T2	PM_State が“ACTIVE”になる。内部クロック出力安定。 LineState[1:0]で“K”を確認後、RestoreUSB が“1”にセットされる。	$T1 + 250\mu s < T2$
T3	ホスト downstream port が FS の“K”送出手を完了。同時にホスト downstream port は USB のサスペンド以前の HS モードに移行。	$T2 + 20ms \{T_{DRSMDN}\}$
T4	USB のサスペンド以前が HS モードであった場合、自動的に HS モードに移行。	$T5 + 1.33\mu s \{2 \text{ Low-speed bit times}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.2.7.2.8 ケーブル挿入

ここでは、ハブ又はホストに接続された場合、すなわちケーブルが挿入された場合を説明します。オート・ネゴシエータは、ケーブル挿抜を検知、及び、制御しません。ケーブル挿入シーケンスの完了後に、オート・ネゴシエータを有効にしてください。

ケーブルが抜かれている状態、もしくは繋がっていない状態に故意にしている時には、D_XcvtControl.XcvtSelect ビットは FS モード、D_XcvtControl.TermSelect ビットは HS モードを初期値としてください。

ケーブルが接続されていない状態(T0)でケーブルが接続されると、VBUS が“H”になり、同時に D_USB_Status.VBUS ビットがセットされます(T1)。この時、例えばスリープ状態にしていた場合は、PM_Control.GoACTIVE ビットを“1”にセットし(T2)、PLL パワーアップ時間経過後(T3)、PM_Control.PM_State[1:0]が"ACTIVE"になると同時に内部クロックが出力され始めます。その後、まずは FS デバイスが接続されたことにしなければならないため、一旦は FS モードになるために、D_XcvtControl.TermSelect ビットを FS モードにしてください(T4)。

その後、ホスト downstream port はリセットを送出し(T5)、ここから HS Detection Handshake が開始されます。

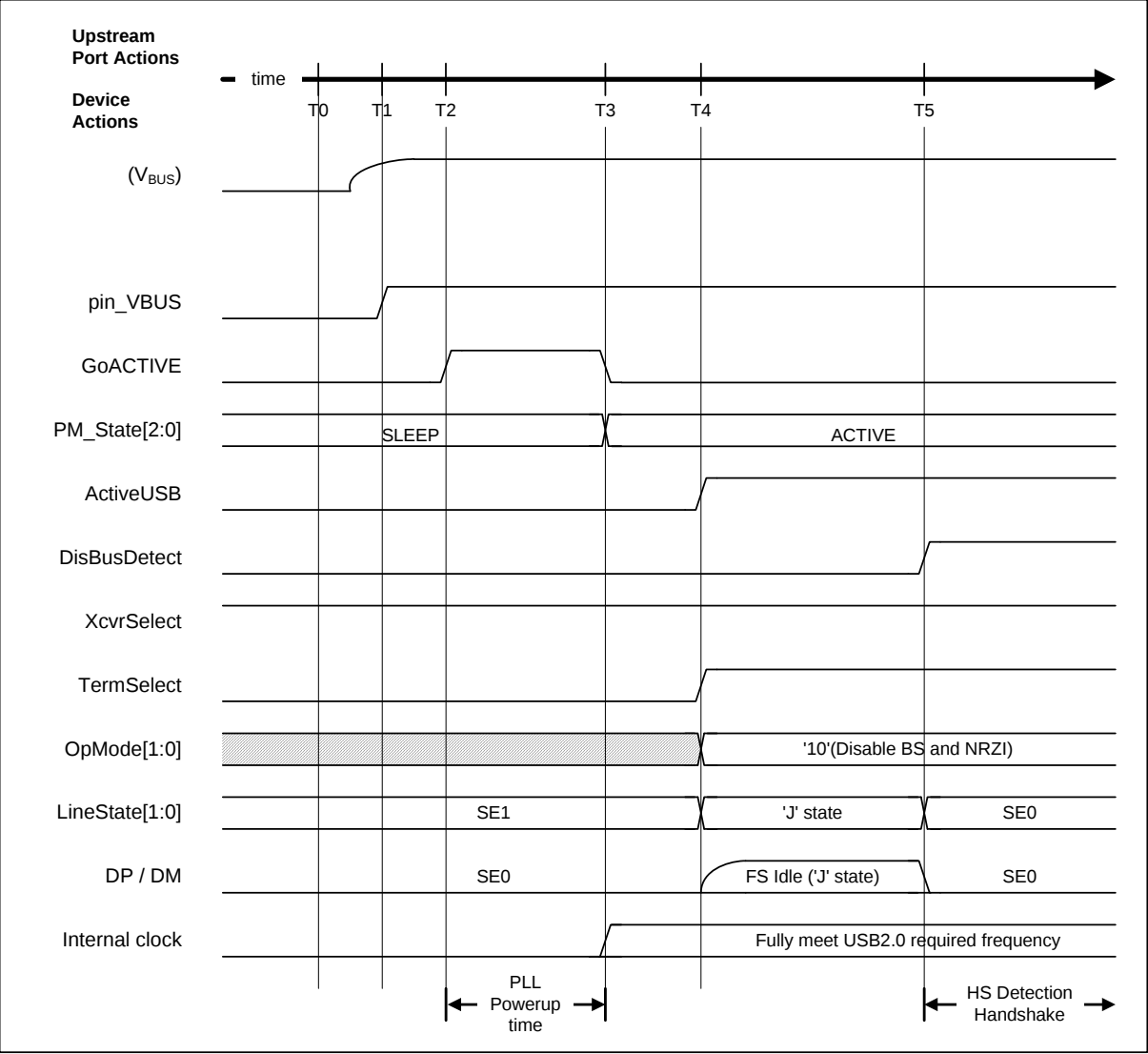


図 1.17 Device Attach Timing

表 1.19 Device Attach Timing Values

Timing Parameter	Description	Value
T0	ケーブルは挿入されていない。	0 (reference)
T1	ケーブルが挿入され、入力ピン VBUS_B が H になる。	T1
T2	GoACTIVE を“1”にセット。	T2
T3	PM_State が“ACTIVE”になる。内部クロック出力の安定。	$T2 + 250\mu s < T3$
T4	ActiveUSB を“1”にセット。TermSelect を“1”にセット。 OpMode[1:0]を“00”に設定。 FS モードに移行。FS ターミネーションが有効。	$T1 + 100ms \{T_{SIGATT}\} < T4$
T5	ホスト downstream port からリセットが送出される。 DisBusDetect を“1”にセット。	$T4 + 100ms \{T_{ATTDDB}\} < T5$

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.3 USBホスト制御

1.3.1 チャンネル

1.3.1.1 チャンネル概要

本 LSI では、パイプと 1 対 1 で対応するホスト側のバッファと、そのバッファを介して行われる転送の為の設定レジスタとを、まとめてチャンネルと呼びます。

チャンネルには、IRP(I/O Request Packet)単位で転送情報を設定します。またチャンネルのバッファとして FIFO 領域を割り当てます。

チャンネルは設定された情報に基づき、IRP を自動的に複数のトランザクションに分割して実行します。チャンネルは IRP 単位で設定を切り替えることができますので、1 本のチャンネルで複数のエンドポイントに対応することが可能です。

図 1.18 にチャンネルの概念を示します。

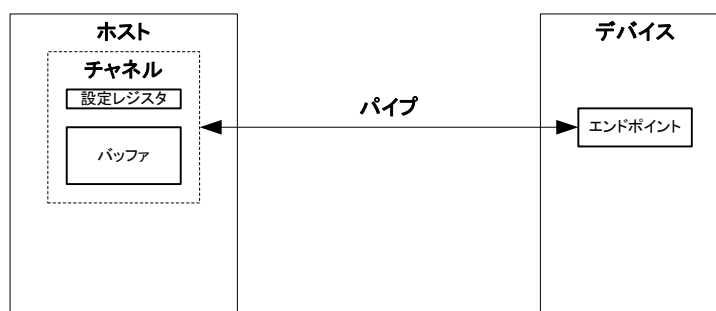


図 1.18 チャンネル概念図

ファームウェアは、バッファおよび転送情報の設定を行ったうえで転送実行(H_CHx{x=0,a-e}Config_0.TranGo)をセットします。転送実行をセットした後は、ファームウェアはIRPデータ数の処理が完了するまで、バッファにデータを書き込む(OUT転送時)、バッファからデータを読み出す(IN転送時)といった処理を行います。

一方、ハードウェア(チャンネル)は IRP を自動的に複数のトランザクションに分割して実行します。転送が完了すると、割り込みによってファームウェアに通知します。

図 1.19 に転送を行う際の基本的な流れを示します。

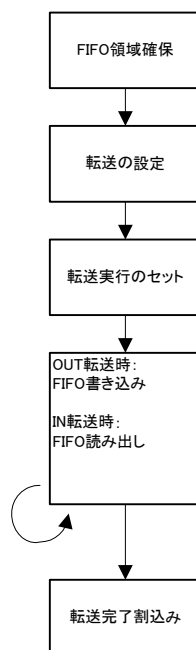


図 1.19 チャンネルの基本的な転送の手順

本 LSI はコントロール転送のみを行うチャンネル(CH0)、バルク転送のみを行うチャンネル(CHA)、バルク / インタラプト / アイソクロナス転送を行うチャンネル(CHb, CHc, CHd, CHe)の計 6 本のチャンネルを持ちます。ここでは、チャンネル CH0 をコントロール専用チャンネル、チャンネル CHa, CHb, CHc, CHd, CHe を汎用チャンネルと呼びます。

各チャンネルには、USB 定義インタフェースによって決定される固定の基本設定項目と、転送毎に制御を行う可変の制御項目及びステータスがあります。基本設定項目は、チップ初期化時、または、USB 定義インタフェースの切り替え時等に設定して下さい。

なお、同時に設定できるインタラプト転送とアイソクロナス転送の最大数は合わせて 4 本です。

各チャンネルが対応する転送タイプを表 1.20 に示します。

表 1.20 対応する転送タイプ

チャンネル	対応する転送タイプ	備考
CH0	コントロール転送	コントロール転送サポート 機能(1.3.4.3)使用可
CHa	バルク転送	バルクオンリーサポート機能(1.3.8)使用可
CHb, CHc, CHd, CHe	バルク転送 インタラプト転送 アイソクロナス転送	オーディオクラスアシスト機能(1.3.9)使用可

1. 機能説明

1.3.1.2 コントロール専用チャネル

本 LSI では、コントロール専用チャネル(CH0)にてコントロール転送を行います。その為、複数のエンドポイントに対してコントロール転送を行う場合はチャネルCH0を時分割して使用します。

表 1.21 にコントロール専用チャネル(CH0)の基本設定項目を示します。

表 1.21 コントロール専用チャネルの基本設定項目

項目	レジスタ/ビット	説明
転送速度	H_CH0Config_0.SpeedMode	チャネルCH0に対応するエンドポイントの転送速度(HS/FS/LS)を設定します。
トグルシーケンスビット	H_CH0Config_0.Toggle	トランザクションを開始する時のトグルシーケンスビットの初期値を設定します。また、トランザクション実行中、およびトランザクション完了後はトグルシーケンスビットの状態を示します。
トランザクション種別	H_CH0Config_1.TID	チャネルCH0で発行するトランザクションの種別(SETUP/IN/OUT)を設定します。
マックスパケットサイズ	H_CH0MaxPktSize	マックスパケットサイズを、LS動作時には8に、FS動作時には8,16,32,64のいずれかの値に設定します。また、HS動作時には64に設定します。
USB アドレス	H_CH0FuncAdrs.FuncAdrs	チャネルCH0で実行するIRPのあて先となるUSBアドレスを設定します。
エンドポイントナンバー	H_CH0FuncAdrs.EP_Number	チャネルCH0で実行するIRPのあて先となるエンドポイントナンバーを設定します。
ハブアドレス	H_CH0HubAdrs.HubAdrs	スプリットトランザクションを行うハブのUSBアドレスを設定します。
ポート番号	H_CH0HubAdrs.Port	スプリットトランザクションを行うハブのポート番号を設定します。
IRP データ数	H_CH0TotalSize_H, H_CH0TotalSize_L	チャネルCH0で実行するIRPのデータ転送数をバイト単位で設定します。
FIFO 領域	AREAn{n=0-5}StartAdrs_H, AREAn{n=0-5}StartAdrs_L, AREAn{n=0-5}EndAdrs_H, AREAn{n=0-5}EndAdrs_L,	チャネルCH0に割り当てる領域を、FIFOのアドレスで設定します。 FIFO領域は、チャネルCH0のマックスパケットサイズ以上の領域を割り当ててください。また、FIFO領域のサイズがデータ転送のスループットに影響します。 FIFOの領域割り当ての詳細は、「1.5 FIFO 管理」を参照してください。
FIFO 領域ジョイン	AREAn{n=0-5}Join_1.JoinEP0CH0	チャネルCH0をFIFO領域に結合します。 コントロール転送サポート機能を使用する場合、チャネルCH0をAREA0に結合してください。
セットアップデータ	H_CH0SETUP_x(x=0-7)	セットアップトランザクションで送信する8バイトのデータを設定します。

1.3.1.3 汎用チャンネル

汎用チャンネルは、トランザクション方向と USB アドレスおよびエンドポイントナンバーを任意に設定出来ますので、同時に 5 本までのエンドポイントに対して 1 対 1 に対応させることが出来ます。また、コントロール専用チャンネルと同様にチャンネルを IRP 単位で時分割して使用することで、5 本を越える数のエンドポイントとも転送を行うことが出来ます。

各チャンネルには、USB 定義インタフェースによって決定される固定の基本設定項目と、転送毎に制御を行う可変の制御項目及びステータスがあります。基本設定項目は、チップ初期化時、または、USB 定義インタフェースの切り替え時等に設定して下さい。

表 1.22 に汎用チャンネルの基本設定項目を示します。USB 定義インタフェースの定義内容に合わせて適宜設定し、また有効にすることによって、USB 定義インタフェースを構成して下さい。

表 1.22 汎用チャンネルの基本設定項目

項目	レジスタ/ビット	説明
転送速度	H_CHx{x=a-e}Config_0.SpeedMode	各チャンネルに対応するエンドポイントの転送速度 (HS/FS/LS)を設定します。
トグルシーケンスビット	H_CHx{x=a-e}Config_0.Toggle	トランザクションを開始する時のトグルシーケンスビットの初期値を設定します。また、トランザクション実行中、およびトランザクション完了後はトグルシーケンスビットの状態を示します。
トランザクション種別	H_CHx{x=a-e}Config_1.TID	各チャンネルで発行するトランザクションの種別 (IN/OUT)を設定します。
転送種別	H_CHx{x=b-e}Config_1.TranType	各チャンネルの転送種別(バルク、インタラプト、アイソクロナス)を設定します。
マックスパケットサイズ	H_CHx{x=a-e}MaxPktSize_H, H_CHx{x=a-e}MaxPktSize_L	各チャンネルで実行するトランザクションのマックスパケットサイズを、1Byte~1024Byte の間の任意の値に設定します。
ハブアドレス	H_CHx{x=a-e}HubAdrs.HubAdrs	スプリットトランザクションを行うハブの USB アドレスを設定します。
ポート番号	H_CHx{x=a-e}HubAdrs.Port	スプリットトランザクションを行うハブのポート番号を設定します。
USB アドレス	H_CHx{x=a-e}FuncAdrs.FuncAdrs	各チャンネルで実行する IRP のあて先となる USB アドレスを設定します。
エンドポイントナンバー	H_CHx{x=a-e}FuncAdrs.EP_Number	各チャンネルで実行する IRP のあて先となるエンドポイントナンバーを設定します。
IRP データ数	H_CHx{x=a-e}TotalSize_HH, H_CHx{x=a-e}TotalSize_HL, H_CHx{x=a-e}TotalSize_LH, H_CHx{x=a-e}TotalSize_LL	各チャンネルで実行する IRP のデータ転送数をバイト単位で設定します。
トークン発行間隔	H_CHx{x=b-e}Interval_H, H_CHx{x=b-e}Interval_L	インタラプト転送及びアイソクロナス転送におけるトークン発行間隔(周期)を設定します。
FIFO 領域	AREAn{n=0-5}StartAdrs_H, AREAn{n=0-5}StartAdrs_L, AREAn{n=0-5}EndAdrs_H, AREAn{n=0-5}EndAdrs_L,	各チャンネルに割り当てる領域を、FIFO のアドレスで設定します。 FIFO 領域は、各チャンネルのマックスパケットサイズ以上の領域を割り当て下さい。また、FIFO 領域のサイズがデータ転送のスループットに影響します。 FIFO の領域割り当ての詳細は、「1.5 FIFO 管理」を参照して下さい。
FIFO 領域ジョイン	AREAn{n=0-5}Join_1.JoinEPxCHx{x=a-e}	各チャンネルを FIFO 領域に結合します。 バルクオンリーサポート機能を使用する場合、チャンネル CHa を AREA1 に結合して下さい。

1. 機能説明

1.3.1.4 チャンネル使用例

1.3.1.4.1 ストレージデバイスを 1 台接続した場合

USB Mass Storage Class(BulkOnly Transport Protocol)に対応したストレージデバイス(例: ハードディスク)を接続した場合のチャンネル使用例を図 1.20 に示します。このクラスに対応したデバイスでは、コントロール転送、バルク IN 転送、バルク OUT 転送を使用します。

コントロール転送に対しては CH0 を使用します。

一方、バルク IN 転送およびバルク OUT 転送については汎用チャンネルを使用しますが、特に本 LSI のバルクオンリーサポート機能を使用する場合には CHa を使用します。

バルクオンリーサポート機能とは USB Mass Storage Class (BulkOnly Transport Protocol)のコマンドトランスポート(CBW)、データトランスポート、ステータストランスポート(CSW)の一連のトランスポート管理を自動で行う機能(「1.3.8 バルクオンリーサポート機能」参照)であり、CHa に実装しています。

バルクオンリーサポート機能を使用しない場合は、バルク IN 転送およびバルク OUT 転送を他の汎用チャンネルに個別(例えば CHb と CHc)に割り当てることも可能です。この場合、トランスポート管理はソフトウェアで行う必要があります。

ハードウェアは、チャンネルに対し転送のスケジューリング(「1.3.2 スケジューリング」参照)を行い、トランザクションを実行します。

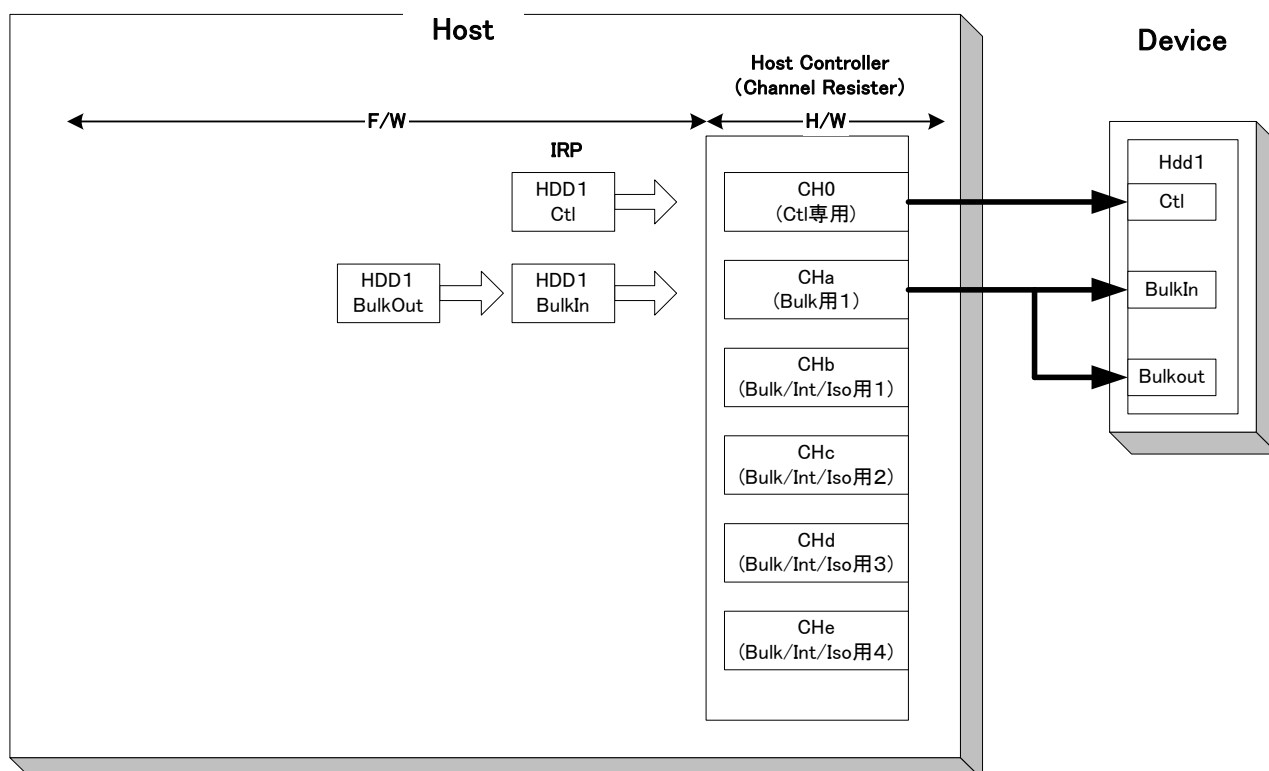


図 1.20 チャンネル使用例(ストレージデバイスを 1 台接続した場合)

1.3.1.4.2 コミュニケーションデバイスを 1 台接続した場合

USB Communication Device Class に対応したコミュニケーションデバイス(例：無線 LAN アダプタ)を接続した場合のチャンネル使用例を図 1.21 に示します。このクラスに対応したデバイスではコントロール転送、バルク IN 転送、バルク OUT 転送、インタラプト IN 転送を使用します。

コントロール転送には CH0 を使用します。

このクラスではバルク IN 転送とバルク OUT 転送は並列に動作する必要があるため、それぞれの転送に汎用チャンネル(例えば CHb, CHc)を個別に割り当てて使用します。

インタラプト IN 転送も同様に汎用チャンネル(例えば CHd)を個別に割り当てて使用します。

ハードウェアは、チャンネルに対し転送のスケジューリング(「1.3.2 スケジューリング」参照)を行い、トランザクションを実行します。

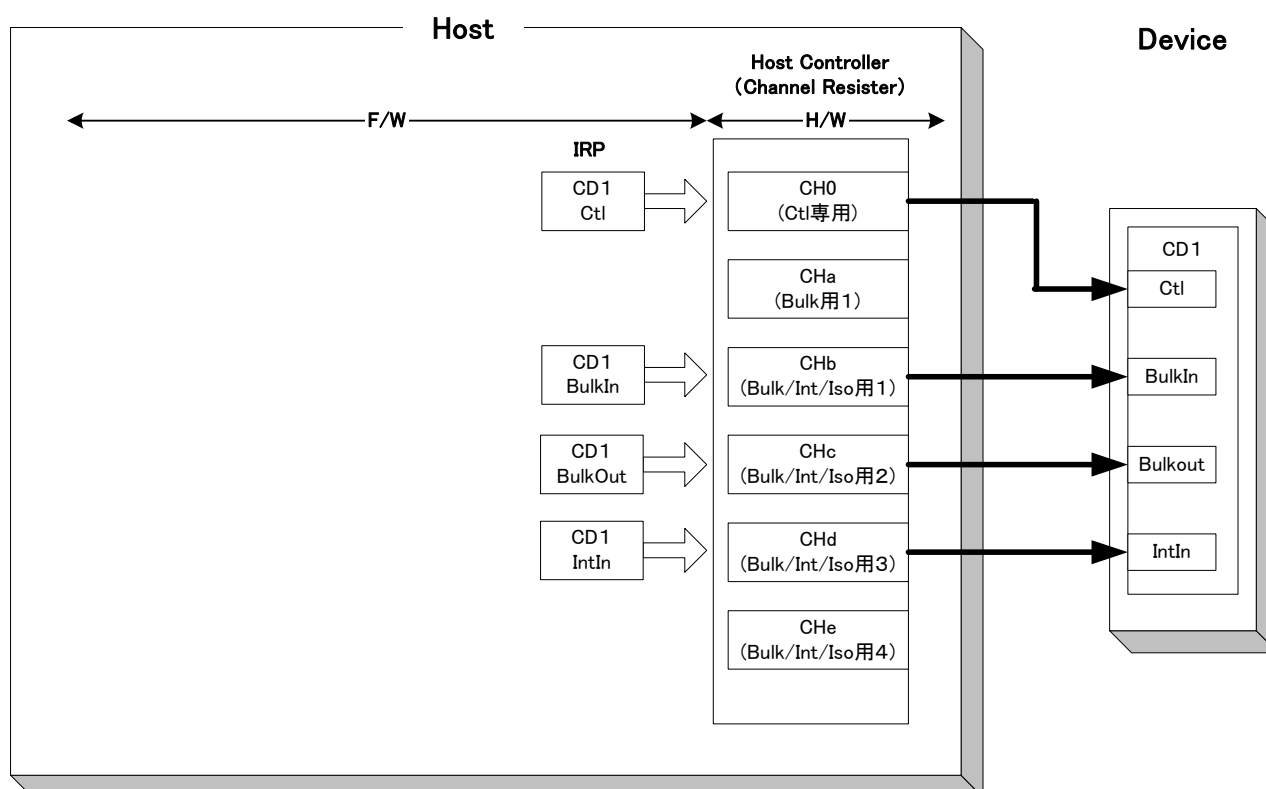


図 1.21 チャンネル使用例(コミュニケーションデバイスを 1 台接続した場合)

1. 機能説明

1.3.1.4.3 ヒューマンインタフェイスデバイスを1台接続した場合

USB Human Interface Device Class に対応したデバイス(例：マウス)を接続した場合のチャンネル使用例を図 1.22 に示します。このクラスに対応したデバイスではコントロール転送、インタラプト IN 転送を使用します。

コントロール転送には CH0 を使用します。

インタラプト IN 転送には汎用チャンネル(例えば CHd)を割り当てて使用します。

ハードウェアは、チャンネルに対し転送のスケジューリング(「1.3.2 スケジューリング」参照)を行い、トランザクションを実行します。

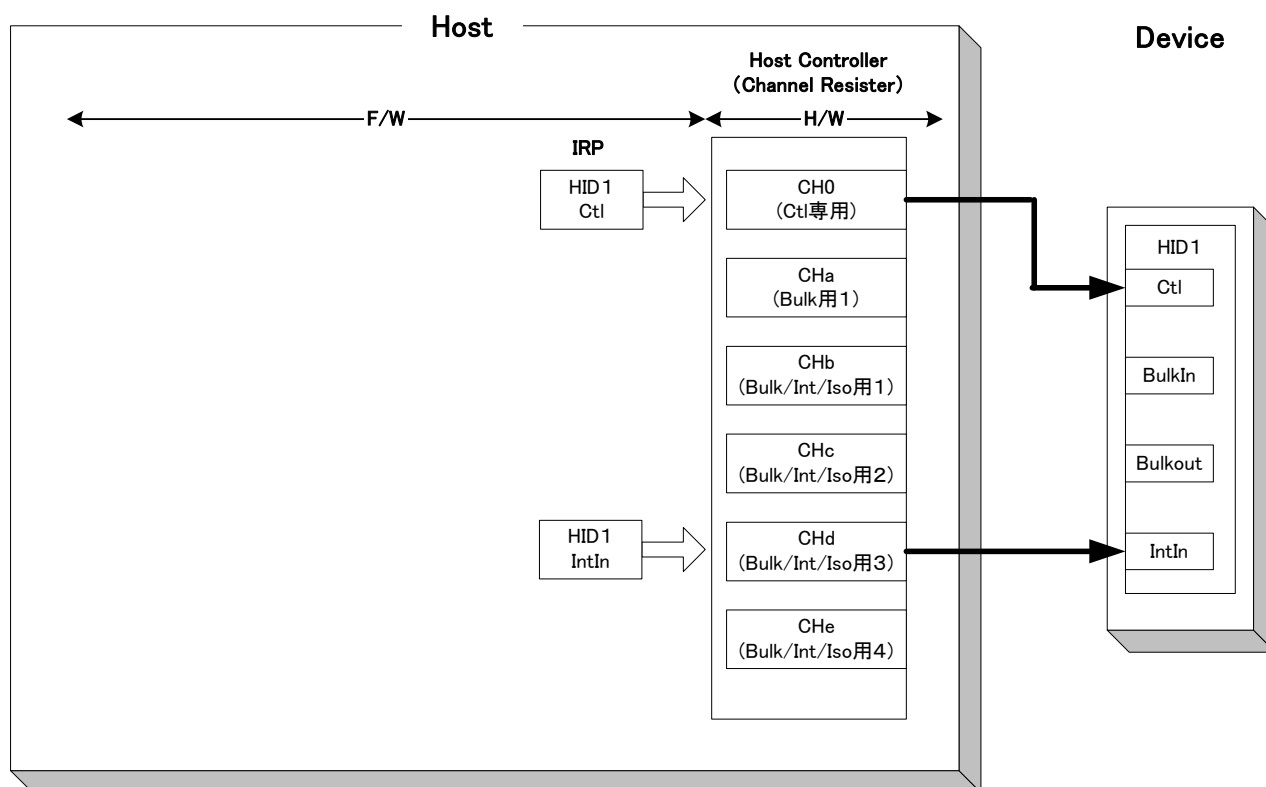


図 1.22 チャンネル使用例(ヒューマンインタフェイスデバイスを1台接続した場合)

1.3.1.4.4 ハブを介してストレージデバイスを 2 台接続した場合

USB Mass Storage Class(BulkOnly Transport Protocol)に対応したストレージデバイス(例: ハードディスク、USB メモリ)を、ハブを介して 2 台接続した場合のチャンネル使用例を図 1.23 に示します。USB Mass Storage Class ではバルク IN 転送、バルク OUT 転送を使用します。

ハブに関してはコントロール転送、インタラプト IN 転送を使用します。

USB メモリに関してはこの例ではハブを内蔵しているものを扱っています。

すべてのデバイスに対するコントロール転送は CH0 を時分割して使用します。

ハブ及び USB メモリのインタラプト IN 転送には、汎用チャンネル(例えば CHd, CHe)を個別に割り当てて使用します。

一方、ハードディスク及び USB メモリのバルク IN 転送およびバルク OUT 転送に対しては、本 LSI のバルクオンリーサポート機能を使用する場合は CHa を使用します。バルクオンリーサポート機能とは USB Mass Storage Class (BulkOnly Transport Protocol)のコマンドトランスポート(CBW)、データトランスポート、ステータストランスポート(CSW)の一連のトランスポート管理を自動で行う機能(「1.3.8 バルクオンリーサポート機能」参照)であり、本 LSI では CHa に実装しています。バルクオンリーサポート機能を使用する場合は転送相手(この例ではハードディスクと USB メモリ)をコマンドトランスポートからステータストランスポートまでを単位として切り替えて使用します。

バルクオンリーサポート機能を使用しない場合は、バルク IN 転送およびバルク OUT 転送を汎用チャンネルに個別に割り当てることも可能です。この場合、トランスポート管理はソフトウェアで行う必要があります。

ハードウェアは、チャンネルに対し転送のスケジューリング(「1.3.2 スケジューリング」参照)を行い、トランザクションを実行します。

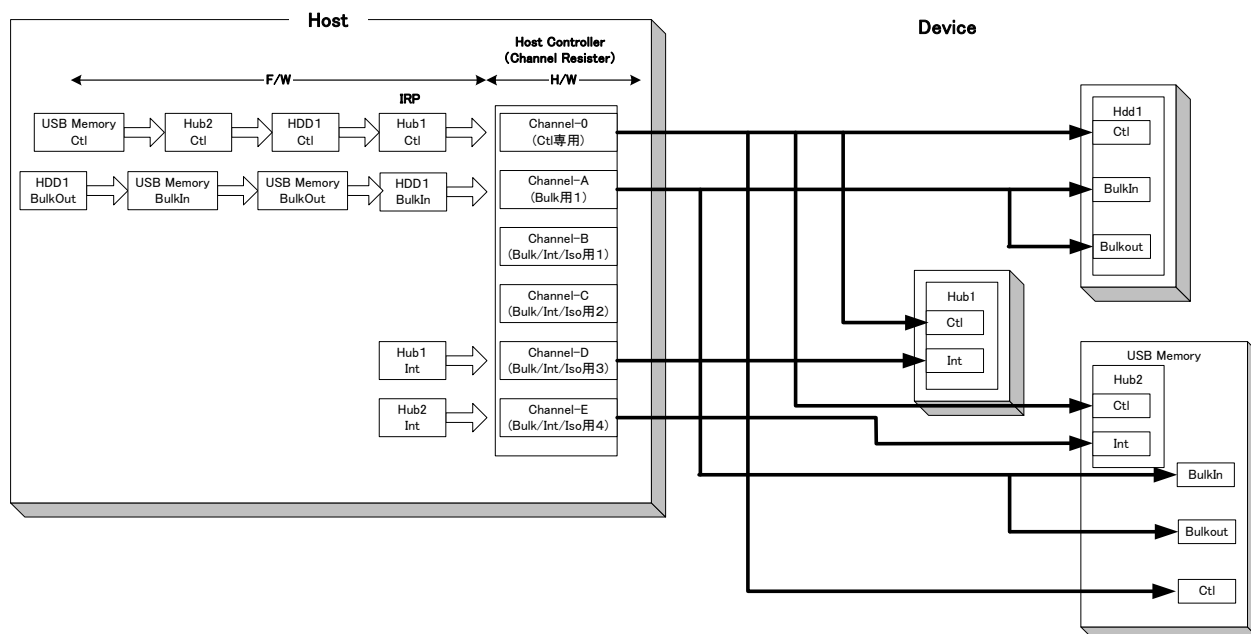


図 1.23 チャンネル使用例(ハブを介してストレージデバイスを 2 台接続した場合)

1. 機能説明

1.3.2 スケジューリング

ハードウェアは転送実行(H_CHx{x=0,a-e}Config_0.TranGo)がセットされたチャンネルの中から一つを選択し、そのチャンネルに設定された転送の実行可否判断を行います。転送が実行可能と判断すると、設定に従いトランザクションを一つ実行します。トランザクションが終了すると、ハードウェアは別のチャンネルを選択し、同様に実行可否判断を行い、トランザクションを実行します。

このようなチャンネルの選択、実行可否判断、トランザクションの実行を繰り返すことにより、ハードウェアは複数のチャンネルを使用して、複数のエンドポイントに対して転送を行います。

表 1.23 にチャンネル CH0 のスケジューリング制御に関する制御項目を示します。

表 1.23 チャンネル CH0 のスケジューリング設定項目

項目	レジスタ/ビット	説明
転送実行	H_CH0Config_0.TranGo	チャンネル CH0 の転送実行を設定します。 チャンネルの設定に従い転送を行います。

表 1.24 に汎用チャンネル(CHA, CHb, CHc, CHd, CHe)のスケジューリング制御に関する制御項目を示します。

表 1.24 汎用チャンネルのスケジューリング設定項目

項目	レジスタ/ビット	説明
転送実行	H_CHx{x=a-e}Config_0.TranGo	各チャンネルの転送実行を設定します。 各チャンネルの設定に従い転送を行います。

1.3.3 トランザクション

本 LSI は、ハードウェアでトランザクション実行機能と、ファームウェアに対するトランザクション実行のためのインタフェイスを提供します。ファームウェアに対するインタフェイスは、制御レジスタとステータスレジスタ、及び、ステータスによりアサートされる割り込み信号として実装されています。ステータスにより割り込みをアサートする設定については、「2. レジスタ」を参照して下さい。

ハードウェアはチャンネルを選択し、転送可否判断を行い、転送可能と判断した場合、チャンネルの設定に従ってトランザクションを実行します。

本 LSI は個々のトランザクション毎に、ファームウェアに対してステータスを発行します。しかしながら、ファームウェアは必ずしも、個々のトランザクションを管理する必要はありません。

例えば、IN のチャンネルであれば、ファームウェアは、CPU インタフェイス(DMA リードまたはレジスタリード)により、FIFO からデータを読み出して FIFO に空き領域を作り出すことによって、IN トランザクションを自動的に連続して実行させることが出来ます。また、OUT のチャンネルであれば、ファームウェアは、CPU インタフェイス(DMA ライトまたはレジスタライト)により、FIFO にデータを書き込んで FIFO に有効データを作り出すことによって、OUT トランザクションを自動的に連続して実行させることが出来ます。

表 1.25 にチャンネル CH0 のトランザクション制御に関する制御項目及びステータスを示します。

表 1.25 チャンネル CH0 の制御項目及びステータス

項目	レジスタ/ビット	説明
トランザクションステータス	H_CH0IntStat.TotalSizeCmp, H_CH0IntStat.TranACK, H_CH0IntStat.TranErr, H_CH0IntStat.ChangeCondition	トランザクションの結果を示します。
トランザクションコンディションコード	H_CH0ConditionCode	トランザクション結果の詳細を示します。

表 1.26 に、汎用チャンネル(CHA, CHb, CHc, CHd, CHE)のトランザクション処理に関する制御項目とステータスを示します。

表 1.26 汎用チャンネルの制御項目とステータス

項目	レジスタ/ビット	説明
トランザクションステータス	H_CHx{x=a-e}IntStat.TotalSizeCmp, H_CHx{x=a-e}IntStat.TranACK, H_CHx{x=a-e}IntStat.TranErr, H_CHx{x=a-e}IntStat.ChangeCondition	トランザクションの結果を示します。
トランザクションコンディションコード	H_CHx{x=a-e}ConditionCode	トランザクション結果の詳細を示します。

1. 機能説明

1.3.3.1 SETUP トランザクション

CH0 の基本設定レジスタにおいて、トランザクション種別(H_CH0Config_1.TID)を"SETUP"に設定します。その他の基本設定項目を適宜設定し、H_CH0SETUP_0~7 レジスタにセットアップデータ(8Byte)を書き込み、転送実行(H_CH0Config_0.TranGo)をセットすることにより、チャンネル CH0 が USB 転送スケジューリングの対象となります。スケジューリングにより該当チャンネルが選択されると、フレーム残り時間を判断し、SETUP トランザクションが実行されます。

SETUP トランザクションでは H_CH0SETUP_0~7 レジスタのデータが用いられ、データパケットのデータ長は 8Byte となります。

SETUP トランザクションに対して ACK を受信すると、ファームウェアに対し、ACK ステータス(H_CH0IntStat.TranACK ビット)を発行します。

SETUP トランザクションに対して正常な応答を受信しない場合、コンディションコード(H_CH0ConditionCode)を"RetryError"に設定し、ファームウェアに対し TranErr ステータス(H_CH0IntStat.TranErr ビット)を発行します。そしてリトライ処理を行いますが、エラーが 3 回連続して続く場合、H_CH0Control.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し ChangeCondition ステータス(H_CH0IntStat.ChangeCondition ビット)を発行します。

図 1.24 に SETUP トランザクションの様子を図示します。(a)LSI が、宛先ノードに存在するエンドポイント 0 に宛てた SETUP トークンを発行します。(b)LSI は続けて、8Byte 長のデータパケットを送信します。(c)LSI は ACK 受信により、該当するレジスタを自動設定し、ファームウェアに対しステータスを発行します。

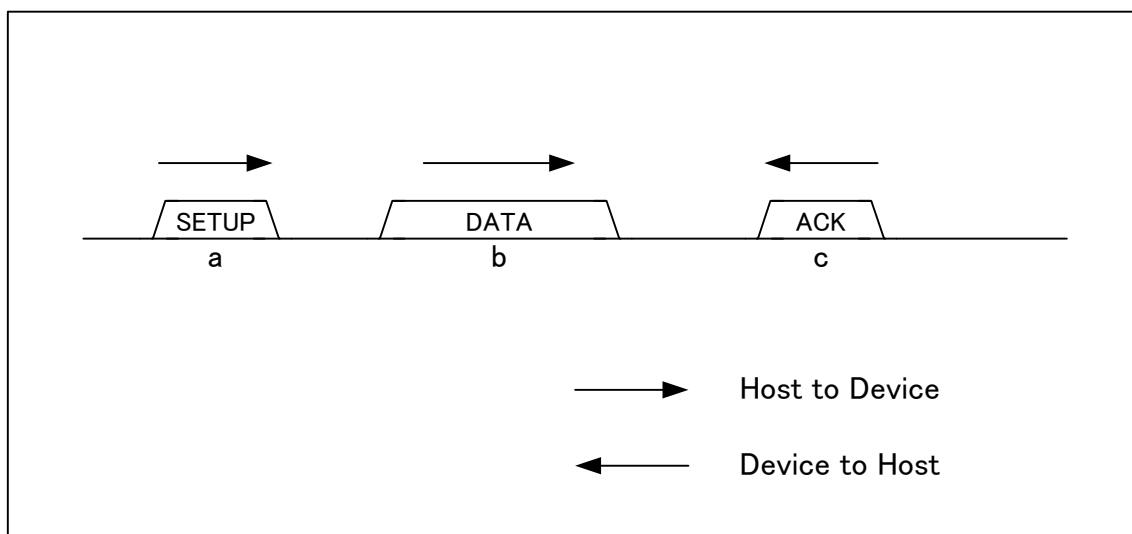


図 1.24 SETUP トランザクション

1.3.3.2 バルクOUTトランザクション

CHx{x=a-e}の基本設定レジスタにおいて、転送種別(H_CHx{x=b-e}Config_1.TranType)を“Bulk”、トランザクション種別(H_CHx{x=a-e}Config_1.TID)を“OUT”に設定します。その他の基本設定項目を適宜設定し、転送実行(H_CHx{x=a-e}Config_0.TranGo)をセットすることにより、チャンネルはハードウェアが行う USB 転送のスケジューリングの対象となります。スケジューリングにより該当チャンネルが選択されると、FIFO の有効データ数及びフレーム残り時間を判断し、バルク OUT トランザクションが実行されます。なお、チャンネル CHa はバルク転送専用となっている為に基本設定レジスタのうち転送種別は存在しません。

個々のデータパケットのデータ長は H_CHx{x=a-e}MaxPacketSize_H,L と H_CHx{x=a-e}TotalSize_HH,HL,LH,LL の値のうち小さい方の値となります。

バルク OUT トランザクションに対して ACK を受信すると、ファームウェアに対して ACK ステータス(H_CHx{x=a-e}IntStat.TranACK ビット)を発行します。また、FIFO を更新し、送信したデータを送信済みとして領域を開放します。

バルク OUT トランザクションに対して NAK を受信した場合、FIFO は更新せず、領域も開放しません。その為、該当チャンネルが再び選択された場合、同じトランザクションを実行します。

バルク OUT トランザクションに対して STALL を受信すると、H_CHx{x=a-e}Config_0.TranGo を自動クリアすることで転送を終了し、コンディションコード(H_CHx{x=a-e}ConditionCode)を“STALL”に設定します。そしてファームウェアに対し、ChangeCondition ステータス(H_CHx{x=a-e}IntStat.ChangeCondition ビット)を発行します。また、FIFO は更新せず、領域も開放しません。

バルク OUT トランザクションに対して正常な応答を受信しない場合、FIFO は更新せず、領域も開放しません。コンディションコード(H_CHx{x=a-e}ConditionCode)を“RetryError”に設定し、ファームウェアに対し、TranErr ステータス(H_CHx{x=a-e}IntStat.TranErr ビット)を発行します。そしてリトライ処理を行います。エラーが 3 回連続して続いた場合は H_CHx{x=a-e}Control.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し ChangeCondition ステータス(H_CHx{x=a-e}IntStat.ChangeCondition ビット)を発行します。

図 1.25 に、完結する場合のバルク OUT トランザクションの様子を図示します。(a)LSI が、宛先ノードに存在する OUT 方向のエンドポイントに宛てた OUT トークンを発行します。(b)LSI は続けて、マックスパケットサイズ以内のデータパケットを送信します。(c)LSI は ACK 受信により、該当するレジスタを自動設定し、ファームウェアに対しステータスを発行します。

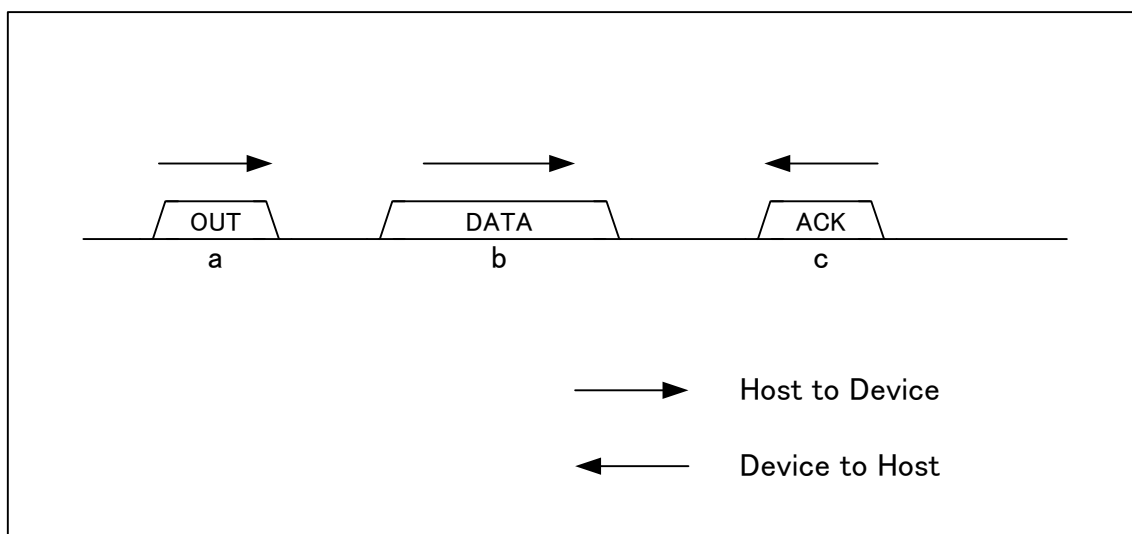


図 1.25 OUT トランザクション

1.3.3.3 インタラプトOUTトランザクション

CHx{x=b-e}の基本設定レジスタにおいて、転送種別(H_CHx{x=b-e}Config_1.TranType)を“Interrupt”、トランザクション種別(H_CHx{x=b-e}Config_1.TID)を“OUT”に設定します。また、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)を設定し、その他の基本設定項目を適宜設定した後に転送実行(H_CHx{x=b-e}Config_0.TranGo)をセットすることにより、チャンネルはハードウェアが行う USB 転送のスケジューリングの対象となります。スケジューリングにより該当チャンネルが選択されると、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)、FIFOの有効データ数及びフレーム残り時間を判断し、インタラプト OUT トランザクションが実行されます。

データパケットのデータ長は H_CHx{x=b-e}MaxPacketSize_H,L と H_CHx{x=b-e}TotalSize_HH,HL,LH,LL の値のうち小さい方の値となります。

インタラプトOUTトランザクションに対してACKを受信すると、ファームウェアに対し、ACK ステータス(H_CHx{x=b-e}IntStat.TranACK ビット)を発行します。また、FIFOを更新して、送信したデータを送信済みとして領域を開放します。

インタラプトOUTトランザクションに対してNAKを受信した場合、FIFOは更新せず、領域も開放しません。その為、該当チャンネルが再び選択された場合、同じトランザクションを実行します。

インタラプトOUTトランザクションに対してSTALLを受信すると、H_CHx{x=b-e}Config_0.TranGoを自動クリアすることで転送を終了し、そしてコンディションコード(H_CHx{x=b-e}ConditionCode)を“STALL”に設定します。そして、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。また、FIFOは更新せず、領域も開放しません。

インタラプトOUTトランザクションに対して正常な応答を受信しない場合、FIFOは更新せず、領域も開放しません。コンディションコード(H_CHx{x=b-e}ConditionCode)を“RetryError”に設定し、ファームウェアに対し、TranErr ステータス(H_CHx{x=b-e}IntStat.TranErr ビット)を発行します。そしてリトライ処理を行いますが、エラーが3回連続して続いた場合はH_CHx{x=b-e}Control.TranGoを自動クリアすることで転送を終了し、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。

1.3.3.4 アイソクロナスOUTトランザクション

CHx{x=b-e}の基本設定レジスタにおいて、転送種別(H_CHx{x=b-e}Config_1.TranType)を“Isochronous”、トランザクション種別(H_CHx{x=b-e}Config_1.TID)を“OUT”に設定します。また、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)を設定し、その他の基本設定項目を適宜設定した後に転送実行(H_CHx{x=b-e}Config_0.TranGo)をセットすることにより、チャンネルはハードウェアが行う USB 転送のスケジューリングの対象となります。スケジューリングにより該当チャンネルが選択された場合、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)、FIFOの有効データ数及びフレーム残り時間を判断し、アイソクロナス OUT トランザクションが実行されます。

データパケットのデータ長は H_CHx{x=b-e}MaxPacketSize_H,L と H_CHx{x=b-e}TotalSize_HH,HL,LH,LL の値のうち小さい方の値となります。

アイソクロナス OUT トランザクションが終了すると、ファームウェアに対し、ACK ステータス(H_CHx{x=b-e}IntStat.TranACK ビット)を発行します。また、FIFO を更新して、送信したデータを送信済みとして領域を開放します。

FIFO の有効データ数がデータパケットのデータ長に満たない場合はトランザクションを発行しません。その為 FIFO は更新せず、領域も開放しません。そして、ファームウェアに対し TranErr ステータスを発行し、コンディションコード(H_CHx{x=b-e}ConditionCode)を“BufferUnderrun”に設定します。

図 1.26 に、完結する場合のアイソクロナス OUT トランザクションの様子を図示します。(a)LSI が、宛先ノードに存在する OUT 方向のエンドポイントに宛てた OUT トークンを発行します。(b)LSI は続けて、マックスパケットサイズ以内のデータパケットを送信します。LSI はデータパケット送信後に該当するレジスタを自動設定し、ファームウェアに対しステータスを発行します。

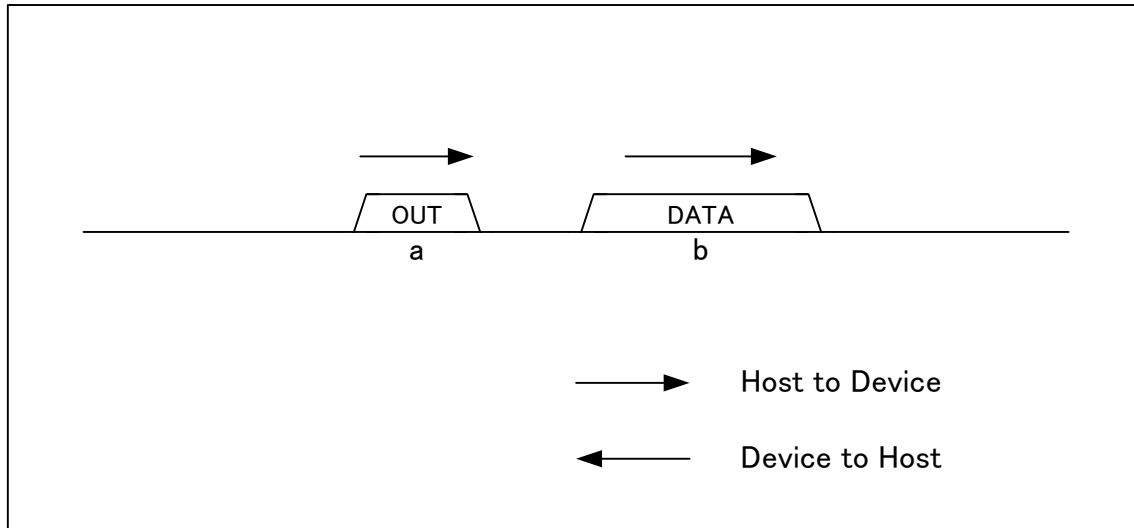


図 1.26 アイソクロナス OUT トランザクション

1. 機能説明

1.3.3.5 バルクINトランザクション

CHx{x=a-e}の基本設定レジスタにおいて、転送種別(H_CHx{x=b-e}Config_1.TranType)を“Bulk”、トランザクション種別(H_CHx{x=a-e}Config_1.TID)を“IN”に設定します。その他の基本設定項目を適宜設定し、転送実行(H_CHx{x=a-e}Config_0.TranGo)をセットすることにより、チャンネルはハードウェアが行う USB 転送のスケジューリングの対象となります。スケジューリングにより該当チャンネルが選択された場合、FIFO の空き容量及びフレーム残り時間を判断し、バルク IN トランザクションが実行されます。なお、チャンネル CHa はバルク転送専用となっている為に基本設定レジスタのうち転送種別は存在しません。

受信するデータパケットの期待データ長は H_CHx{x=a-e}MaxPacketSize_H,L と H_CHx{x=a-e}TotalSize_HH,HL,LH,LL の値のうち小さい方の値となります。

バルク IN トランザクションにおいて、全てのデータが正常に受信できると、ACK 応答し、トランザクションを完結します。また、ファームウェアに対し、ACK ステータス H_CHx{x=a-e}IntStat.TranACK ビット)を発行します。更に、FIFO を更新して、データを受信済みとして領域を確保します。

バルク IN トランザクションにおいて、受信データ長が期待データ長より小さい場合、H_CHx{x=a-e}Config_0.TranGo を自動クリアすることで転送を終了し、ACK 応答します。また、コンディションコード(H_CHx{x=a-e}ConditionCode)を“DataUnderrun”に設定します。そして、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=a-e}IntStat.ChangeCondition ビット)を発行します。更に、FIFO を更新して、データを受信済みとして領域を確保します。

バルク IN トランザクションにおいて、NAK を受信した場合、ステータスは発行しません。また、FIFO は更新しません。

バルク IN トランザクションに対して STALL を受信すると、H_CHx{x=a-e}Config_0.TranGo を自動クリアすることで転送を終了し、コンディションコード(H_CHx{x=a-e}ConditionCode)を“STALL”に設定します。そしてファームウェアに対し、ChangeCondition ステータス(H_CHx{x=a-e}IntStat.ChangeCondition ビット)を発行します。また、FIFO は更新しません。

バルク IN トランザクションにおいて、受信データ長が期待データ長より大きい場合、H_CHx{x=a-e}Config_0.TranGo を自動クリアすることで転送を終了します。応答は行いません。また、コンディションコード H_CHx{x=a-e}ConditionCode)を“DataOverrun”に設定します。そして、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=a-e}IntStat.ChangeCondition ビット)を発行します。FIFO は更新しません。

バルク IN トランザクションにおいて、トグルミスマッチが発生した場合、ACK 応答します。また、コンディションコード(H_CHx{x=a-e}ConditionCode)を“RetryError”に設定します。そして、ファームウェアに対し、TranErr ステータス(H_CHx{x=a-e}IntStat.TranErr ビット)を発行します。FIFO は更新しません。

バルク IN トランザクションにおいて、タイムアウトエラー、CRC エラー、ビットスタッフィングエラー、PID エラー(予期せぬ PID 含む)が発生した場合、応答は行いません。また、コンディションコード(H_CHx{x=a-e}ConditionCode)を“RetryError”に設定します。そして、ファームウェアに対し、TranErr ステータス(H_CHx{x=a-e}IntStat.TranErr ビット)を発行します。FIFO は更新しません。

コンディションコード(H_CHx{x=a-e}ConditionCode)が“RetryError”に設定されるエラーが発生した場合、リトライ処理を行います。そして、エラーが 3 回連続して続く場合 H_CHx{x=a-e}Control.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=a-e}IntStat.ChangeCondition ビット)を発行します。

図 1.27 に、完結する場合のバルク IN トランザクションの様子を図示します。(a)LSI が、宛先ノードに存在する IN 方向のエンドポイントに宛てた IN トークンを発行します。(b)エンドポイントは、この IN トランザクションに応答できる場合、マックスパケットサイズ以内のデータパケットを送信します。(c)LSI は ACK 応答します。そして該当するレジスタを自動設定し、ファームウェアに対しステータスを発行します。

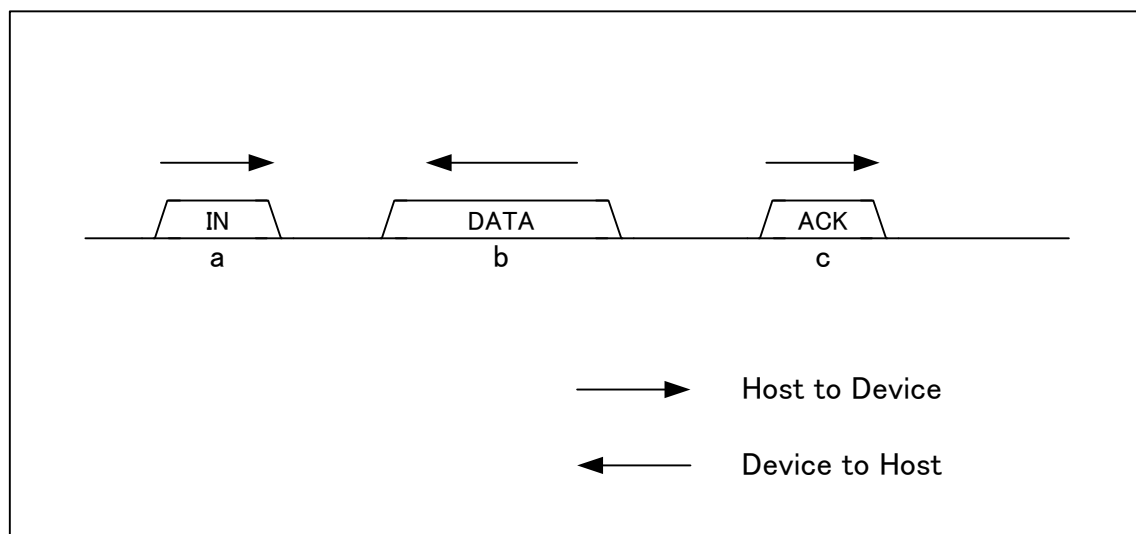


図 1.27 IN トランザクション

1. 機能説明

1.3.3.6 インタラプトINトランザクション

CHx{x=b-e}の基本設定レジスタにおいて、転送種別(H_CHx{x=b-e}Config_1.TranType)を“Interrupt”、トランザクション種別(H_CHx{x=b-e}Config_1.TID)を“IN”に設定します。また、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)を設定し、その他の基本設定項目を適宜設定し、転送実行(H_CHx{x=b-e}Config_0.TranGo)をセットすることにより、チャンネルはハードウェアが行う USB 転送のスケジューリングの対象となります。スケジューリングにより該当チャンネルが選択された場合、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)、FIFO の空き容量及びフレーム残り時間を判断し、インタラプト IN トランザクションが実行されます。

受信するデータパケットの期待データ長は H_CHx{x=b-e}MaxPacketSize_H,L と H_CHx{x=b-e}TotalSize_HH,HL,LH,LL の値のうち小さい方の値となります。

インタラプト IN トランザクションにおいて、全てのデータが正常に受信できると、ACK 応答し、トランザクションを完結します。また、ファームウェアに対し、ACK ステータス(H_CHx{x=b-e}IntStat.TranACK ビット)を発行します。また、FIFO を更新して、データを受信済みとして領域を確保します。

インタラプト IN トランザクションにおいて、受信データ長が期待データ長より小さい場合、H_CHx{x=b-e}Config_0.TranGo を自動クリアすることで転送を終了し、ACK 応答します。また、そしてコンディションコード(H_CHx{x=b-e}ConditionCode)を“DataUnderrun”に設定します。そして、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。また、FIFO を更新して、データを受信済みとして領域を確保します。

インタラプト IN トランザクションにおいて、NAK を受信した場合、ステータスは発行しません。また、FIFO は更新しません。次のトランザクションは次の周期にて行います。

インタラプト IN トランザクションに対して STALL を受信すると、H_CHx{x=b-e}Config_0.TranGo を自動クリアすることで転送を終了し、コンディションコード(H_CHx{x=b-e}ConditionCode)を“STALL”に設定します。そしてファームウェアに対し、ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。また、FIFO は更新しません。

バルク IN トランザクションにおいて、受信データ長が期待データ長より大きい場合、H_CHx{x=b-e}Config_0.TranGo を自動クリアすることで転送を終了します。応答は行いません。また、コンディションコード H_CHx{x=b-e}ConditionCode)を“DataOverrun”に設定します。そして、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。FIFO は更新しません。

インタラプト IN トランザクションにおいて、トグルミスマッチが発生した場合、ACK 応答します。また、コンディションコード(H_CHx{x=b-e}ConditionCode)を“RetryError”に設定します。そして、ファームウェアに対し、TranErr ステータス(H_CHx{x=b-e}IntStat.TranErr ビット)を発行します。FIFO は更新しません。

インタラプト IN トランザクションにおいて、タイムアウトエラー、CRC エラー、ビットスタッフィングエラー、PID エラー(予期せぬ PID 含む)が発生した場合、応答は行いません。また、コンディションコード(H_CHx{x=b-e}ConditionCode)を“RetryError”に設定します。そして、ファームウェアに対し、TranErr ステータス(H_CHx{x=b-e}IntStat.TranErr ビット)を発行します。FIFO は更新しません。

コンディションコード(H_CHx{x=b-e}ConditionCode)が“RetryError”に設定されるエラーが発生した場合、次の周期にてリトライ処理を行います。そして、エラーが 3 回連続して続く場合 H_CHx{x=b-e}Control.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。

1.3.3.7 アイソクロナスINトランザクション

CHx{x=b-e}の基本設定レジスタにおいて、転送種別(H_CHx{x=b-e}Config_1.TranType)を“Isochronous”、トランザクション種別(H_CHx{x=b-e}Config_1.TID)を“IN”に設定します。また、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)を設定し、その他の基本設定項目を適宜設定し、転送実行(H_CHx{x=b-e}Config_0.TranGo)をセットすることにより、チャンネルはハードウェアが行う USB 転送のスケジューリングの対象となります。スケジューリングにより該当チャンネルが選択された場合、トークン発行間隔(H_CHx{x=b-e}Interval_H,L)、FIFOの空き容量及びフレーム残り時間を判断し、アイソクロナス IN トランザクションが実行されます。

受信するデータパケットの期待データ長は H_CHx{x=b-e}MaxPacketSize_H,L と H_CHx{x=b-e}TotalSize_HH,HL,LH,LL の値のうち小さい方の値となります。

アイソクロナス IN トランザクションにおいて、全てのデータが正常に受信できると、トランザクションを完結します。また、ファームウェアに対し、ACK ステータス(H_CHx{x=b-e}IntStat.TranACK ビット)を発行します。また、FIFO を更新して、データを受信済みとして領域を確保します。

アイソクロナス IN トランザクションにおいて、受信データ長が期待データ長より小さい場合、ファームウェアに対し、ACK ステータス(H_CHx{x=b-e}IntStat.TranACK ビット)及び ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。そしてコンディションコード(H_CHx{x=b-e}ConditionCode)を“DataUnderrun”に設定します。また、FIFO を更新して、データを受信済みとして領域を確保します。

アイソクロナス IN トランザクションにおいて、受信データ長が期待データ長より大きい場合、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=b-e}IntStat.ChangeCondition ビット)を発行します。そしてコンディションコード(H_CHx{x=b-e}ConditionCode)を“DataOverrun”に設定します。、FIFO は更新しません。

アイソクロナス IN トランザクションにおいて、タイムアウトエラー、CRC エラー、ビットスタッフィングエラー、PID エラー(予期せぬ PID 含む)が発生した場合、ファームウェアに対し、TranErr ステータス(H_CHx{x=b-e}IntStat.TranErr ビット)を発行します。また、コンディションコード(H_CHx{x=b-e}ConditionCode)を“RetryError”に設定します。FIFO は更新しません。

FIFO の空き領域のサイズが H_CHx{x=b-e}MaxPacketSize_H,L が示す値に満たない場合はトランザクションを発行しません。そして、ファームウェアに対し TranErr ステータスを発行し、コンディションコード(H_CHx{x=b-e}ConditionCode)を“BufferOverrun”に設定します。

図 1.28 に、完結する場合のアイソクロナス IN トランザクションの様子を図示します。(a)LSI が、宛先ノードに存在する IN 方向のエンドポイントに宛てた IN トークンを発行します。(b)エンドポイントは、この IN トランザクションに応答できる場合、マックスパケットサイズ以内のデータパケットを送信します。LSI はデータパケット受信後に該当するレジスタを自動設定し、ファームウェアに対しステータスを発行します。

1. 機能説明

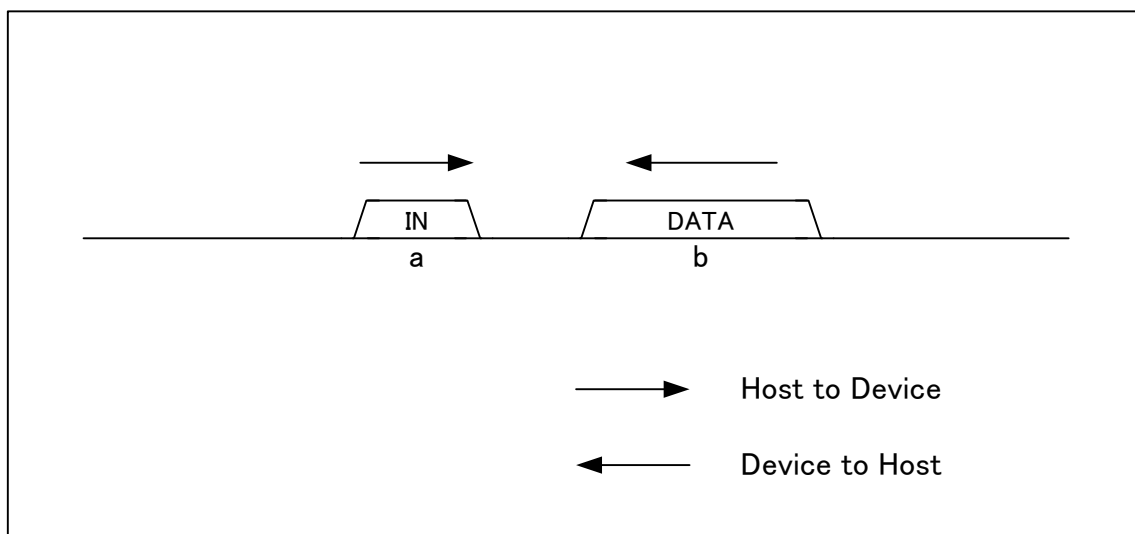


図 1.28 アイソクロナス IN トランザクション

1.3.3.8 PING トランザクション

バルク OUT または コントロール OUT トランザクションを行うチャネルでは、HS 動作時に、以下に示す条件で PING トランザクションを実行する状態に移行します。チャネルが PING トランザクションを実行する状態の時にスケジューリングにより選択されると PING トランザクションを実行します。

OUT トランザクションに対して、NYET を受信した場合、NAK を受信した場合、または応答を受信しない場合、チャネルは PING トランザクションを実行する状態に移行します。

PING トランザクションに対して NAK を受信した場合、チャネルは引き続き PING トランザクションを実行する状態となります。また、ファームウェアに対しステータスは発行しません。

PING トランザクションに対して ACK を受信した場合、チャネルは PING トランザクションを実行する状態から OUT トランザクションを行う状態に戻ります。ファームウェアに対しステータスは発行しません。

PING トランザクションに対して STALL を受信した場合、 $H_CHx\{x=0,a-e\}Config_0.TranGo$ を自動クリアすることで転送を終了し、コンディションコード ($H_CHx\{x=0,a-e\}ConditionCode$) を“STALL”に設定します。そして、ファームウェアに対し、ChangeCondition ステータス ($H_CHx\{x=0,a-e\}IntStat.ChangeCondition$ ビット) を発行します。

PING トランザクションに対して正常な応答を受信しない場合、コンディションコード ($H_CHx\{x=0,a-e\}ConditionCode$) を“RetryError”に設定します。そして、ファームウェアに対し、TranErr ステータス ($H_CHx\{x=0,a-e\}IntStat.TranErr$ ビット) を発行します。この場合、リトライ処理を行います。エラーが 3 回連続して続く場合、 $H_CHx\{x=0,a-e\}Control.TranGo$ を自動クリアすることで転送を終了し、ファームウェアに対し ChangeCondition ステータス ($H_CHx\{x=0,a-e\}IntStat.ChangeCondition$ ビット) を発行します。

PING トランザクションにおいては、FIFO が更新されることはありません。

図 1.29 に、PING トランザクションに対して ACK 応答する様子を図示します。(a) LSI は、このノードに存在する OUT 方向のエンドポイントに宛てた PING トークンを発行します。(b) デバイスは、エンドポイントにマックスパケットサイズ分の空きがある場合、この PING トランザクションに対し ACK 応答します。

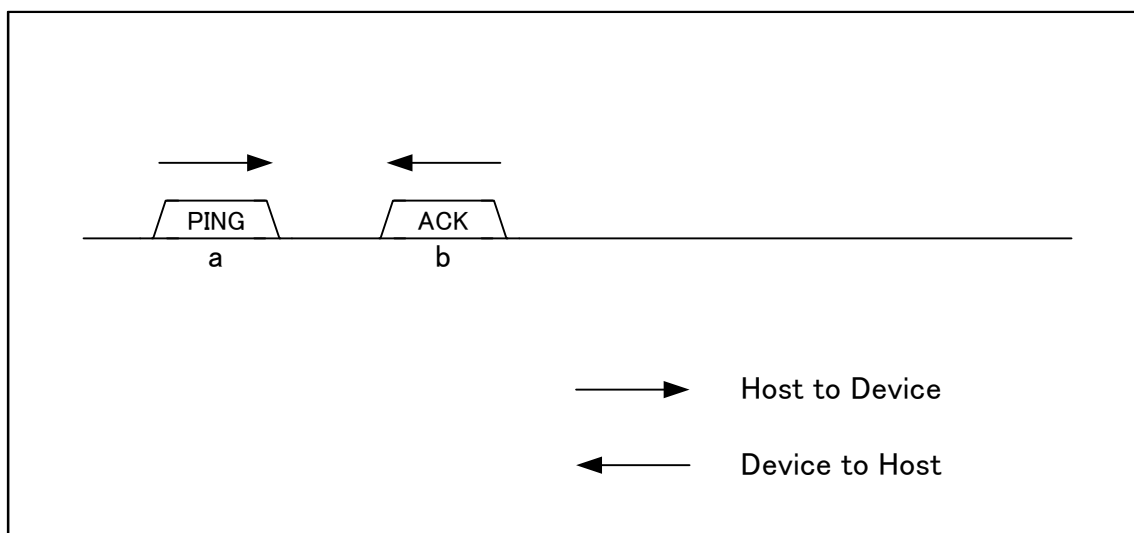


図 1.29 PING トランザクション

1. 機能説明

1.3.3.9 low-speed(LS)トランザクション

LS デバイスとの転送にはコントロール転送、またはインタラプト転送を使用します。

ダウンストリームポートに LS デバイスが接続された場合、ホストは LS にて動作します。使用するチャネルの転送速度(H_CHx{x=0,a-e}Config_0.SpeedMode)を“LS”に設定することで LS bit time にてトランザクションを実行します。

一方、ダウンストリームポートに full-speed(FS)ハブが接続され、そのハブのダウンストリームポートに LS デバイスが接続された場合、ホストは FS にて動作します。使用するチャネルの転送速度(H_CHx{x=0,a-e}Config_0.SpeedMode)を LS に設定することで、対応するエンドポイントへの全てのダウンストリーム・パケットの先頭にプリアンプルを伴って送信します。プリアンプルは FS bit time にて送信し、プリアンプルに続くダウンストリーム・パケットは LS bit time にて送信します。

図 1.30 に、完結する場合のインタラプト OUT トランザクションの様子を図示します。(a)LSI は、宛先ノードに存在する OUT 方向のエンドポイントに宛てた OUT トークンを先頭にプリアンプルを付与して発行します。(b)LSI は続けて、マックスパケットサイズ以内のデータパケットを先頭にプリアンプルを付与して送信します。(c)LSI は ACK 受信により、該当するレジスタを自動設定し、ファームウェアに対しステータスを発行します。

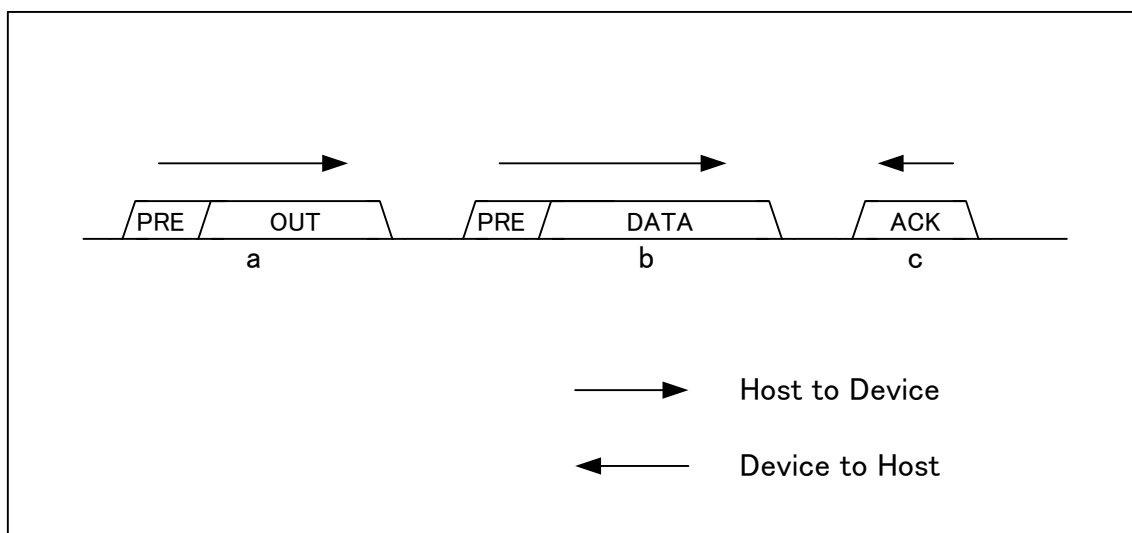


図 1.30 Preamble が付与された OUT トランザクション

図 1.31 に、完結する場合のインタラプト IN トランザクションの様子を図示します。(a)LSI は、宛先ノードに存在する IN 方向のエンドポイントに宛てた IN トークンを先頭にプリアンプルを付与して発行します。(b)デバイスはマックスパケットサイズ以内のデータパケットを送信してきます。LSI はこのデータを該当するチャネルの FIFO に書き込みます。(c)LSI はデータを受信出来ると ACK 応答を先頭にプリアンプルを付与して行います。また、該当するレジスタを自動設定し、ファームウェアに対しステータスを発行します。

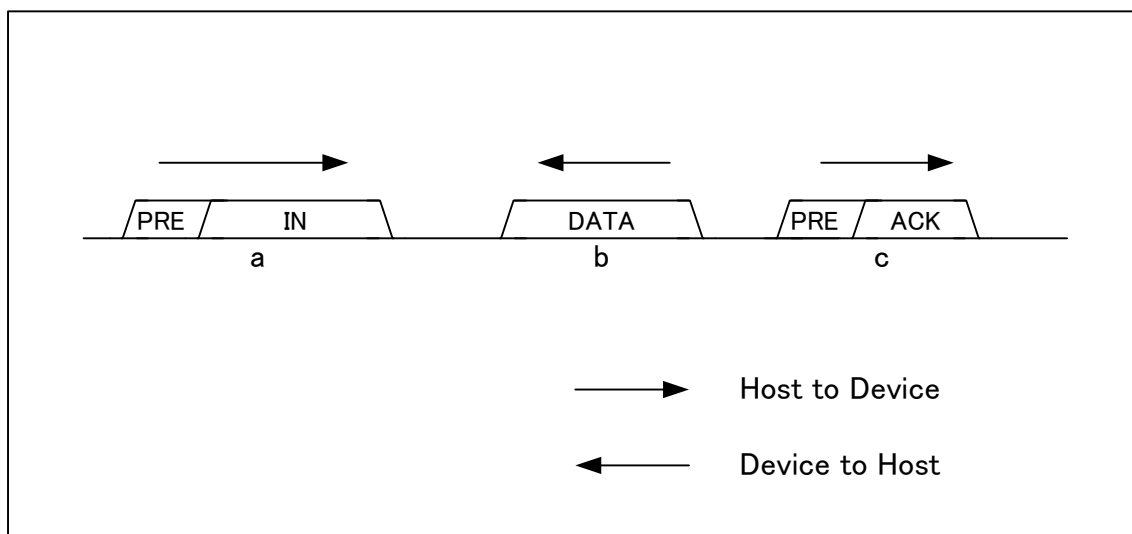


図 1.31 Preamble が付与された IN トランザクション

1.3.3.10 スプリットトランザクション

ダウンストリームポートに high-speed(HS)ハブが接続され、そのハブのダウンストリームポートに FS または LS デバイスが接続された場合、ホストは HS にて動作します。使用するチャネルの転送速度(H_CHx{x=0,a-e}Config_0.SpeedMode)を FS または LS に設定することで、ハブとの間にて、対応するエンドポイントへのトランザクションをスプリットトランザクションで実行します。

該当チャネルにはハブアドレス (H_CHx{x=0,a-e}HubAdrs.HubAdrs)、ポート番号 (H_CHx{x=0,a-e}HubAdrs.Port)を適切な値に設定します。

スプリットトランザクションにおけるスタートスプリットトランザクションからコンプリートスプリットトランザクションのシーケンスは、ハードウェアが制御します。その為ファームウェアはスプリットトランザクションの個々のトランザクションを意識する必要はありません。

コントロール転送、バルク転送、インタラプト転送、及び IN 方向のアイソクロナス転送の場合、スタートスプリットトランザクションからコンプリートスプリットトランザクションのシーケンスにおいて、最終のコンプリートスプリットトランザクションが正常に完結した場合、ファームウェアに対し ACK ステータス(H_CHx{x=0,a-e}IntStat.TranACK ビット)を発行し、FIFO を更新します。

一方、OUT 方向のアイソクロナス転送の場合、最終のスタートスプリットトランザクションが正常に完結した場合、ファームウェアに対し ACK ステータス(H_CHx{x=b-e}IntStat.TranACK ビット)を発行し、FIFO を更新します。

上記以外のスタートスプリットトランザクション及びコンプリートスプリットトランザクションが正常に完結した場合、ファームウェアに対してのステータスの発行は行いません。

スタートスプリットトランザクションからコンプリートスプリットトランザクションのシーケンスにおいて、個々のスプリットトランザクションでエラーが発生した場合、コンディションコード(H_CHx{x=0,a-e}ConditionCode)を“RetryError”に設定し、ファームウェアに対し、TranErr ステータス(H_CHx{x=0,a-e}IntStat.TranErr ビット)を発行します。FIFO は更新しません。そしてリトライ処理を行います。コントロール転送、バルク転送、インタラプト転送ではエラーが 3 回連続して続く場合、H_CHx{x=0,a-e}Control.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=0,a-e}IntStat.ChangeCondition ビット)を発行します。

1.3.4 コントロール転送

コントロール転送は、その各ステージを個々のトランザクションとして制御するか、または、後述(「1.3.4.3 コントロール転送サポート機能」)のコントロール転送サポート機能によって、自動的行えます。

図 1.32 にコントロール転送の制御の様子を図示します。ファームウェアは **SETUP**、**DATA**、**STATUS** の各ステージを適宜設定することでコントロール転送をハードウェアに実行させます。

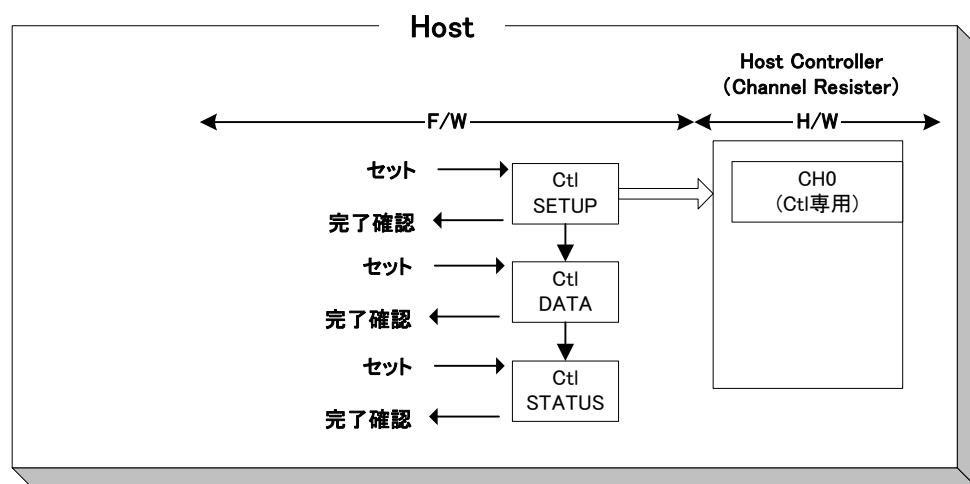


図 1.32 コントロール転送の制御

図 1.33 にデータステージが OUT 方向である場合のコントロール転送の様子を図示します。(a) ホストは、**SETUP** トランザクションによって、コントロール転送を開始します。(b) ホストは **OUT** トランザクションを発行して、データステージを行います。(c) ホストは **IN** トランザクションを発行して、ステータスステージを行います。

データステージの無いコントロール転送は、この例におけるデータステージが無い状態で実施されます。

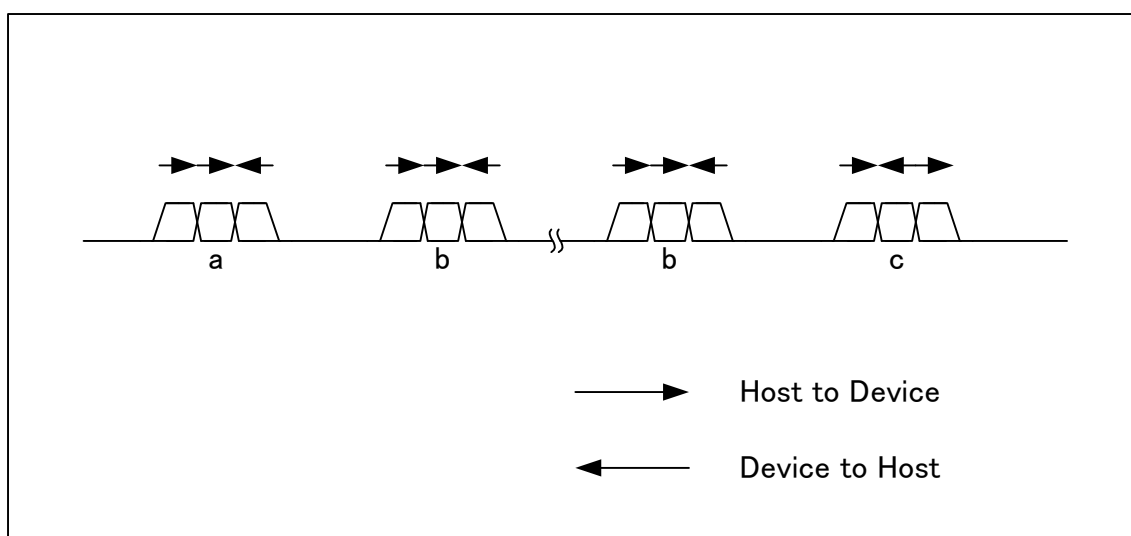


図 1.33 データステージが OUT 方向のコントロール転送

1. 機能説明

図 1.34 にデータステージが IN 方向である場合のコントロール転送の様子を図示します。(a) ホストは、SETUP トランザクションによって、コントロール転送を開始します。(b) ホストは IN トランザクションを発行して、データステージを行います。(c) ホストは OUT トランザクションを発行して、ステータスステージを行います。

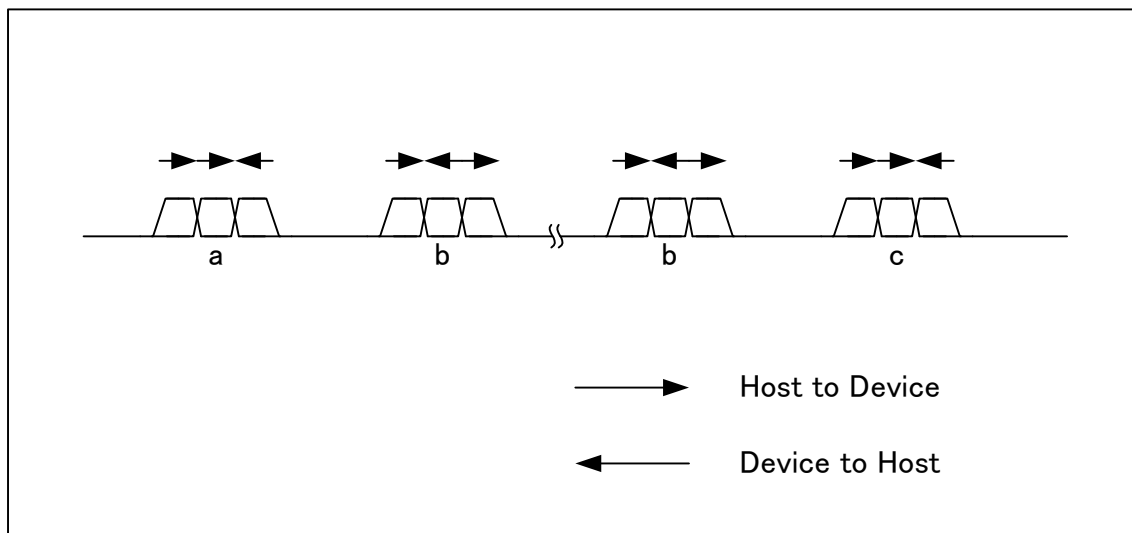


図 1.34 データステージが IN 方向のコントロール転送

1.3.4.1 セットアップステージ

セットアップステージはセットアップトランザクションによって行います。セットアップトランザクションの詳細は「1.3.3.1 SETUP トランザクション」を参照して下さい。

1.3.4.2 データステージ/ステータスステージ

セットアップステージ終了後、次のステージに移行してください。

そのステージが IN 方向である場合、トランザクション種別(H_CH0Config_1.TID)を“IN”に設定し、その他の基本設定レジスタを適宜設定して、トランザクションを実行してください。

一方、そのステージが OUT 方向である場合、トランザクション種別(H_CH0Config_1.TID)を“OUT”に設定し、その他の基本設定レジスタを適宜設定して、トランザクションを実行してください。

なお、ステータスステージを行う場合、IRP データ数(H_CH0TotalSize_H,L)を 0x0 に設定してトランザクションを実行してください。

1.3.4.3 コントロール転送サポート機能

本 LSI にはコントロール転送の一連のステージ管理を自動で行う機能があります。この機能を用いることにより、ファームウェアでは各ステージを個々のトランザクションとして管理する必要がなくなります。

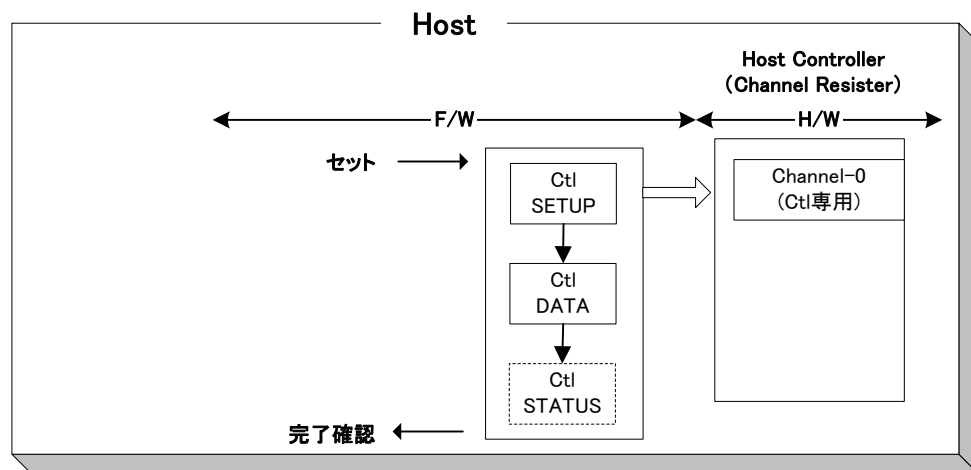


図 1.35 コントロール転送サポート機能の制御

コントロール転送サポート機能はチャンネル CH0 にて有効です。この機能を使用した場合のコントロール転送は以下ようになります。ファームウェアは(1)～(4)、(7)の処理を行います。

- (1) チャンネル CH0 の以下の基本設定レジスタに適宜設定を行います。
 転送速度(H_CH0Config_0.SpeedMode)、マックスパケットサイズ(H_CH0MaxPktSize)、
 USB アドレス(H_CH0FuncAdrs.FuncAdrs)、エンドポイントナンバー
 (H_CH0FuncAdrs.EP_Number)、FIFO 領域(AREAn{n=0-5}StartAdrs_H,L,
 AREAn{n=0-5}EndAdrs_H,L)、
 FIFO 領域ジョイン(AREAn{n=0-5}Join_1.JoinEP0CH0)
- (2) セットアップレジスタ(H_CH0SETUP_0～7)にセットアップデータ(8Byte)を書き込みます。
- (3) データステージが OUT 方向の場合は CH0 にジョインした FIFO 領域に送信すべきデータを書き込みます。データステージが IN 方向の場合は CH0 にジョインした FIFO 領域をクリアします。
- (4) コントロール転送サポート実行(H_CTL_SupportControl.CTL_SupportGo)をセットします。
 この時、コントロール転送ステージ(H_CTL_SupportControl.CTL_SupportState)の値は
 “Idle(00b)”として H_CTL_SupportControl レジスタにライトを行います。
- (5) SETUP レジスタのデータ(8Byte)を用いて SETUP トランザクションを実行します(セットアップステージ)。
- (6) SETUP データに基づき、データステージを実行します。
 - SETUP データの bmRequestType の Bit7 が 0 の場合、CH0 にジョインした FIFO 領域にあるデータを OUT トランザクションによって送信します(OUT 方向のデータステージ)。OUT 方向のデータステージは SETUP データの wLength で示されるデータ数を OUT トランザクションで送信すると終了します。

- SETUP データの bmRequestType の Bit7 が 1 の場合、IN トランザクションを発行し、受信したデータを CH0 にジョインした FIFO 領域に書き込みます(IN 方向のデータステージ)。IN 方向のデータステージは SETUP データの wLength で示されるデータ数を IN トランザクションで受信すると終了します。また、IN トランザクションでショートパケットを受信した場合も IN 方向のデータステージは終了します。
 - SETUP データの wLength の値が 0x0000 の場合はデータステージは実行しません。
- (7) CH0 にジョインした FIFO 領域が SETUP データの wLength で示される値より小さい場合、ファームウェアはデータステージのデータを分割して処理する必要があります。
- データステージが OUT 方向の場合、CH0 にジョインした FIFO 領域に送信するデータが無くなるとトランザクションは発行されなくなります。従ってファームウェアは FIFO の空き領域を確認しながら、順次残りの送信データを FIFO に書き込む必要があります。
 - データステージが IN 方向の場合、CH0 にジョインした FIFO 領域に空き領域が無くなるとトランザクションは発行されなくなります。従ってファームウェアは FIFO の有効データ数を確認しながら順次受信したデータを FIFO から読み出し、FIFO に空き領域を作る必要があります。
- (8) SETUP データに基づき、ステータスステージを実行します。
- OUT 方向のデータステージが終了すると、IN トランザクションを発行します(IN 方向のステータスステージ)。
 - IN 方向のデータステージが終了し、CH0 にジョインした FIFO 領域に存在する受信データがすべて読み出され FIFO が空になると、ゼロ長パケットの OUT トランザクションを発行します(OUT 方向のステータスステージ)。
- (9) コントロール転送が正常に完了するとコントロール転送サポート実行(H_CTL_SupportControl.CTL_SupportGo)を自動クリアし、コントロール転送終了ステータス(H_CH0IntStat.CTL_SupportCmp)を発行します。
- (10) コントロール転送の途中でトランザクションのエラーを検出すると、コントロール転送サポート実行(CTL_SupportControl.CTL_SupportGo)を自動クリアすることでコントロール転送を中止し、コントロール転送停止ステータス(H_CH0IntStat.CTL_SupportStop)を発行します。そして、コントロール転送ステージ(H_CTL_SupportControl.CTL_SupportState)により、エラーが発生したステージを示します。また、コンディションコード(H_CH0ConditionCode)を適切な値に設定し、ChangeCondition ステータス(H_CH0IntStat.ChangeCondition ビット)を発行します。

コントロール転送を中断する場合は、コントロール転送サポート実行

(H_CTL_SupportControl.CTL_SupportGo)をクリアします。コントロール転送の中断処理が終了するとステータスを発行します。

この時、コントロール転送がステータスステージまで完了して中断処理が終了した場合はコントロール転送終了ステータス(H_CH0IntStat.CTL_SupportCmp)を発行します。

一方、コントロール転送がステータスステージまで完了せずに中断処理が終了した場合はコントロール転送停止ステータス(H_CH0IntStat.CTL_SupportStop)を発行します。

コントロール転送の中断したステージはコントロール転送ステージ(H_CTL_SupportControl.CTL_SupportState)により示します。

中断したステージよりコントロール転送を再開する場合、コントロール転送ステージ(H_CTL_SupportControl.CTL_SupportState)を再開するステージに設定(即ち中断したステージに設定を保持)し、コントロール転送サポート実行(H_CTL_SupportControl.CTL_SupportGo)をセットします。

一方、新たなコントロール転送を行う場合は、コントロール転送ステージ(H_CTL_SupportControl.CTL_SupportState)に“Idle(00b)”を設定し、コントロール転送サポート実行(H_CTL_SupportControl.CTL_SupportGo)をセットします。

コントロール転送サポート機能を実行中は、転送実行ビット(H_CH0Config_0.TranGo)、トグルシーケンスビット(H_CH0Config_0.Toggle)、トランザクション種別(H_CH0Config_1.TID)、IRP データ数(H_CH0TotalSize_H,L)はハードウェアにより設定および更新しますので、これらに対する書き込みは行わないで下さい。

トランザクションのエラーについては、「1.3.3 トランザクション」を参照して下さい。

コントロール転送サポート機能の設定項目、ステータスを表 1.27 に示します。

表 1.27 コントロール転送サポート機能の制御項目及びステータス

項目	レジスタ/ビット	説明
コントロール転送サポート実行	H_CTL_SupportControl.CTL_SupportGo	コントロール転送のステージの管理を自動で行います。 詳細については、「1.3.4.3 コントロール転送サポート機能」を参照して下さい。
コントロール転送ステージ	H_CTL_SupportControl.CTL_SupportState	コントロール転送サポート機能において、実行中のステージを示します。また、コントロール転送がエラーで中止された場合は、エラーが発生したステージを示します。
コントロール転送実行結果	H_CH0IntStat.CTL_SupportCmp H_CH0IntStat.CTL_SupportStop	コントロール転送サポート機能によるコントロール転送の実行結果を示します。
トランザクションステータス	H_CH0IntStat.TotalSizeCmp, H_CH0IntStat.TranACK, H_CH0IntStat.TranErr, H_CH0IntStat.ChangeCondition	トランザクションの結果を示します。
トランザクションコンディションコード	H_CH0ConditionCode.ConditionCode	トランザクション結果の詳細を示します。

1. 機能説明

1.3.5 バルク転送／インタラプト転送／アイソクロナス転送

CHa におけるバルク転送、CHb, CHc, CHd, CHe におけるバルク転送、インタラプト転送及びアイソクロナス転送はデータフロー(「1.3.6 データフロー」参照)としても、連続する個々のトランザクション(「1.3.3 トランザクション」参照)としても制御できます。

1.3.6 データフロー

OUT 転送及び IN 転送の一般的なデータフローの制御について、説明します。

1.3.6.1 OUT転送

H_CH0TotalSize_H,L または H_CHx{x=a-e}TotalSize_HH,HL,LH,LL に OUT 転送の総データ数を設定し、OUT 転送により送信するデータを、各チャンネルにジョインした FIFO 領域上に書き込んで下さい。FIFO にデータを書き込むには、CPU インタフェイスによるレジスタ書き込みと、CPU インタフェイスによる DMA 書き込みの方法があります。

CPU インタフェイスのレジスタライトにより FIFO にデータを書き込むには、AREAn{n=0-5}Join_0.JoinCPU_Wr ビットにより、チャンネルにジョインした FIFO 領域を 1 つ選択して下さい。選択した FIFO 領域には、FIFO_Wr_0,1 レジスタ、または FIFO_ByteWr レジスタにより書き込むことができ、書き込み順にデータパケットで送信されます。また、FIFO の空き容量を、FIFO_WrRemain_H,L レジスタにより参照できます。フル状態の FIFO へ書き込むことは出来ません。必ず FIFO_WrRemain_H,L レジスタにより空き数を確認し、その数を超えないように書き込んで下さい。

OUT トランザクションで送信するデータパケットのサイズは、チャンネル CH0 では H_CH0TotalSize_H,L と H_CH0MaxPktSize の値のうち小さい方となります。チャンネル CHx{x=a-e}では H_CHx{x=a-e}TotalSize_HH,HL,LH,LL と H_CHx{x=a-e}MaxPktSize_H,L の値のうち小さい方となります。

FIFO にデータパケットのデータサイズ以上のデータが有れば、OUT トランザクションを実行し、データを送信します。また、H_CH0TotalSize_H,L または H_CHx{x=a-e}TotalSize_HH,HL,LH,LL は送信したデータサイズ分減算されます。そして、TotalSize がゼロになると、H_CHx{x=0,a-e}Config_0.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し、TotalSizeCmp ステータス(H_CHx{x=0,a-e}IntStat.TotalSizeCmp ビット)を発行します。

このように、ファームウェアによって個々のトランザクションについての制御を行うことなく、OUT 転送を行うことが出来ます。

1.3.6.2 IN転送

H_CH0TotalSize_H,L または H_CHx{x=a-e}TotalSize_HH,HL,LH,LL に IN 転送の総データ数を設定して下さい。

IN トランザクションで受信するデータパケットの期待サイズは、チャンネル CH0 では H_CH0TotalSize_H,L と H_CH0MaxPktSize の値のうち小さい方となります。チャンネル CHx{x=a-e}では H_CHx{x=a-e}TotalSize_HH,HL,LH,LL と H_CHx{x=a-e}MaxPktSize_H,L の値のうち小さい方となります。FIFO にマックスパケットサイズ以上の空き領域があれば、IN トランザクション実行して、データを受信します。また、H_CH0TotalSize_H,L または H_CHx{x=0,a-e}TotalSize_HH,HL,LH,LL は受信したデータサイズ分減算されます。そして、TotalSize がゼロになると、H_CHx{x=0,a-e}Config_0.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し、TotalSizeCmp ステータス (H_CHx{x=0,a-e}IntStat.TotalSizeCmp ビット)を発行します。

受信したデータサイズが期待データ長より大きい場合、コンディションコード H_CHx{x=0,a-e}ConditionCode)を"DataOverrun"に設定します。そして、ファームウェアに対し、ChangeCondition ステータス(H_CHx{x=0,a-e}IntStat.ChangeCondition ビット)を発行します。FIFO は更新しません。コントロール転送、バルク転送、インタラプト転送の場合は更に H_CHx{x=0,a-e}Config_0.TranGo を自動クリアし、そのチャンネルでの転送を終了します。

受信したデータサイズがデータパケットの期待サイズ未満であれば、コンディションコード(H_CHx{x=0,a-e}ConditionCode)を"DataUnderrun"に設定します。そしてファームウェアに対し、ChangeCondition ステータス(H_CHx{x=0,a-e}IntStat.ChangeCondition ビット)を発行します。また、FIFO を更新して、データを受信済みとして領域を確保します。コントロール転送、バルク転送、インタラプト転送の場合は更に H_CHx{x=0,a-e}Config_0.TranGo を自動クリアし、そのチャンネルでの転送を終了します。

このように、ファームウェアによって個々のトランザクションについての制御を行うことなく、IN 転送を行うことが出来ます。

IN 転送によって受信したデータは、各チャンネルのジョインした FIFO 領域上に書き込まれます。FIFO のデータを読み出すには、CPU インタフェイスによるレジスタ読み出し、CPU インタフェイスによる DMA 読み出しの方法があります。

CPU インタフェイスのレジスタリードにより FIFO のデータを読み出すには、該当するチャンネルとジョインされた FIFO 領域を AREAn{n=0-5}Join_0.JoinCPU_Rd ビットによりただ一つ選択して下さい。選択した FIFO 領域は、FIFO_Rd_0,1 レジスタ、または FIFO_ByteRd レジスタにより、受信順に読み出すことが出来ます。また、読み出し可能な FIFO のデータ数を、FIFO_RdRemain_H,L レジスタにより参照できます。空の FIFO を読み出すことはできませんので、必ず FIFO_RdRemain_H,L レジスタによりデータ数を確認し、その数を超えないように読み出して下さい。

CPU インタフェイスの DMA リードにより FIFO のデータを読み出すには、AREAn{n=0-5}Join_1.JoinDMA ビットにより、DMA のチャンネルにただ一つの FIFO 領域を選択し、DMA_Control.Dir ビットに 1 を設定して下さい。選択した FIFO 領域は、CPU インタフェイスにおいて DMA 手順を実行することにより、受信順に読み出されます。また、FIFO の残りデータ数を、DMA_Remain_H,L レジスタで参照できます。FIFO が空になると、CPU インタフェイスは自動的に DMA を一時停止してフロー制御を行います。

1. 機能説明

1.3.7 ゼロ長パケット自動発行機能

OUT 転送を行うチャンネルにおいて、H_CHx{x=a-e}Config_1.AutoZeroLen ビットをセットすることにより、ゼロ長パケットを自動で発行する機能が有効になります。

ゼロ長パケット自動発行機能の設定項目を表 1.28 に示します。

表 1.28 ゼロ長パケット自動発行機能の制御項目

項目	レジスタ/ビット	説明
ゼロ長パケット 自動発行	H_CHx{x=a-e}Config_1.AutoZeroLen	ゼロ長パケットの自動発行機能を有効にします。OUT 転送の場合のみこのビットは有効です。

1.3.7.1 バルク／インタラプトOUT転送のゼロ長パケット自動発行機能

バルク／インタラプト OUT 転送を実行しているチャンネルにおいて、H_CHx{x=a-e}TotalSize_HH,HL,LH,LL レジスタで設定したデータサイズの転送がちょうどマックスパケットサイズで完了しても H_CHx{x=a-e}Config_0.TranGo の自動クリアは行わず、転送を継続します。そして、再びこのチャンネルがスケジューリングされると OUT トランザクションをゼロ長パケットで実行します。このトランザクションが正常に完結すると、H_CHx{x=a-e}Config_0.TranGo を自動クリアすることで転送を終了し、ファームウェアに対し、TotalSizeCmp ステータス(H_CHx{x=a-e}IntStat.TotalSizeCmp)を発行します。

1. 機能説明

- OUT エンドポイントトグルシーケンス(H_CHaBO_OUT_EP_Control.OUT_Toggle)、
OUT エンドポイントナンバー(H_CHaBO_OUT_EP_Control.OUT_EP_Number)、
IN エンドポイントトグルシーケンス(H_CHaBO_IN_EP_Control.OUT_Toggle)、
IN エンドポイントナンバー(H_CHaBO_IN_EP_Control.IN_EP_Number)
- (3) FIFO の CBW エリアに CBW データ(31Byte)を書き込みます。
- (4) CHa にジョインした FIFO 領域に DMA、または PIO の設定を行います。
- (5) バルクオンリーサポート実行(H_CHaBO_SupportControl.BO_SupportGo)をセットします。
この時、トランスポート状態(H_CHaBO_SupportControl.BO_TransportState)の値は
“Idle(00b)”として H_CHaBO_SupportControl レジスタにライトを行います。
- (6) CBW エリアのデータ(31Byte)を、OUT エンドポイントナンバー
(H_CHaBO_OUT_EP_Control.OUT_EP_Number)で示される OUT 方向のエンドポイントに
対して、バルク OUT トランザクションによって送信します(コマンドトランスポート)。
- (7) CBW データに基づき、データトランスポートを実行します。
- CBW データの bmCBWFlags の Bit7 が 0 の場合、CHa にジョインした FIFO 領域にあるデータ
を、OUT エンドポイントナンバー(H_CHaBO_OUT_EP_Control.OUT_EP_Number)で示さ
れる OUT 方向のエンドポイントに対して、バルク OUT トランザクションによって送
信します(OUT 方向のデータトランスポート)。OUT 方向のデータトランスポートは
CBW データの dCBWDataTransferLength で示されるデータ数を OUT トランザクショ
ンで送信すると終了します。
 - CBW データの bmCBWFlags の Bit7 が 1 の場合、IN エンドポイントナンバー
(H_CHaBO_IN_EP_Control.IN_EP_Number)で示される IN 方向のエンドポイントに
対して、バルク IN トランザクションを発行し、受信したデータを CHa にジョインした FIFO
領域に書き込みます(IN 方向のデータトランスポート)。IN 方向のデータトランス
ポートはCBWデータのdCBWDataTransferLengthで示されるデータ数をINトランザク
ションで受信すると終了します。また、IN トランザクションでショートパケットを受信し
た場合も IN 方向のデータトランスポートは終了します
 - CBW データの dCBWDataTransferLength の値が 0x00000000 の場合はデータトランス
ポートは実行しません。
- (8) CHa にジョインした FIFO 領域が CBW データの wCBWDataTransferLength で示される値
より小さい場合、ファームウェアはデータトランスポートのデータを分割して処理する必
要があります。
- データトランスポートが OUT 方向の場合、FIFO に送信するデータが無くなるとラン
ザクションは発行されなくなります。従ってファームウェアは FIFO の空き領域を確認
しながら、順次残りの送信データを FIFO に書き込む必要があります。
 - データトランスポートが IN 方向の場合、FIFO に空き領域がなくなるとランザクシ
ョンは発行されなくなります。従ってファームウェアは FIFO の有効データ数を確認しな
がら順次受信したデータを FIFO から読み出し、FIFO に空き領域を作る必要がありま
す。
- (9) IN エンドポイントナンバー(H_CHaBO_IN_EP_Control.IN_EP_Number)で示される IN 方向
のエンドポイントに対して、バルク IN トランザクションを発行し、受信したデータを FIFO
の CSW エリアに書き込みます(ステータストランスポート)。ステータストランスポート
において受信したデータ数はステータストランスポート受信データ数
(H_CHaBO_CSW_RcvDataSize.CSW_RcvDataSize)に反映します。
- OUT 方向のデータトランスポートが終了した場合、ステータストランスポートを実行
する状態になります。
 - IN 方向のデータトランスポートが終了した場合、FIFO に存在する受信データがすべて
読み出され FIFO が空になると、ステータストランスポートを実行する状態になります。

(10) ステータストランスポートで受信した CSW データをチェックします。チェック内容は以下の通りです。

- 受信した CSW のデータ長が 13Byte であること。
- CSW の dCSWSignature が 0x53425355 であること。
- CSW の dCSWTag が CBW の dCBWTag と一致すること。
- BCSWStatus の値が 0x00 であること。

以上を一つでも満たさない場合、バルクオンリーサポート実行(H_CHaBO_SupportControl.BO_SupportGo)を自動クリアし、バルクオンリーサポート機能を停止します。またバルクオンリーサポート停止ステータス

(H_CHaIntStat.BO_SupportStop)を発行します。CSW エリアに受信したデータは、RAM_Monitor 機能を用いて読み出すことができます。

(11) ステータストランスポートが正しく完結すると、バルクオンリーサポート実行(H_CHaBO_SupportControl.BO_SupportGo)を自動クリアし、バルクオンリーサポート終了ステータス(H_CHaIntStat.BO_SupportCmp)を発行します。

(12) それぞれのトランスポートにおいて、トランザクションのエラーを検出すると、バルクオンリーサポート実行(H_CHaBO_SupportControl.BO_SupportGo)を自動クリアすることでバルクオンリーサポート機能を停止し、バルクオンリーサポート停止ステータス(H_CHaIntStat.BO_SupportStop)を発行します。そして、トランスポート状態(H_CHaBO_SupportControl.BO_TransportState)により、エラーが発生したトランスポートを示します。また、コンディションコード(H_CHaConditionCode)を適切な値に設定し、ChangeCondition ステータス(H_CHaIntStat.ChangeCondition ビット)を発行します。

バルクオンリーサポート機能を中断する場合は、バルクオンリーサポート実行(H_CHaBO_SupportControl.BO_SupportGo)をクリアします。バルクオンリーサポート機能の中断処理が終了するとステータスを発行します。

この時、トランスポートがステータストランスポートまで完了して中断処理が終了した場合はバルクオンリーサポート完了ステータス(CHaIntStat.BO_SupportCmp)を発行します。

一方、トランスポートがステータストランスポートまで完了せずに中断処理が終了した場合はバルクオンリーサポート停止ステータス(H_CHaIntStat.BO_SupportStop)を発行します。

中断したトランスポートはトランスポート状態(H_CHaBO_SupportControl.BO_TransportState)に示します。

中断したトランスポートよりバルクオンリーサポート機能を再開する場合、トランスポート状態(H_CHaBO_SupportControl.BO_TransportState)を再開するトランスポートに設定(即ち中断したトランスポートに設定を保持)し、バルクオンリーサポート実行(H_CHaBO_SupportControl.BO_SupportGo)をセットします。

一方、新たにバルクオンリーサポート機能を実行する場合は、トランスポート状態(H_CHaBO_SupportControl.BO_TransportState)に Idle(00b)を設定し、バルクオンリーサポート実行(H_CHaBO_SupportControl.BO_SupportGo)をセットします。

バルクオンリーサポート機能実行中は、転送実行ビット(H_CHaConfig_0.TranGo)、トグルシーケンスビット(H_CHaConfig_0.Toggle)、トランザクション種別(H_CHaConfig_1.TID)、トータルサイズフリービット(H_CHaConfig_1.TotalSizeFree)、エンドポイント番号(H_CHaFuncAdrs.EP_Number)、IRP データ数(H_CHaTotalSize_HH,HL,LH,LL)はハードウェアにより設定及び更新されていきますので、これらに対する書き込みは行わないで下さい。

トランザクションのエラーについては、「1.3.3 トランザクション」を参照して下さい。

FIFO の CBW エリア、CSW エリアについては、「1.5 FIFO 管理」を参照して下さい。

DMA については、「1.6.3.2 DMA」を参照して下さい。

バルクオンリーサポート機能の設定項目、ステータスを表 1.29 に示します。

1. 機能説明

表 1.29 バルクオンリーサポート機能の設定項目、ステータス

項目	レジスタ/ビット	説明
バルクオンリーサポート実行	H_CHaBO_SupportControl.BO_SupportGo	バルクオンリーサポート機能を実行します。詳細については、「1.3.8 バルクオンリーサポート機能」を参照して下さい。
OUT エンドポイントトグルシーケンス	H_CHaBO_OUT_EP_Control.OUT_Toggle	OUT エンドポイントのトグルシーケンスビットの初期値を設定します。また、トランザクション実行中、およびトランザクション完了後は OUT エンドポイントのトグルシーケンスビットの状態を示します。
OUT エンドポイントナンバー	H_CHaBO_OUT_EP_Control.OUT_EP_Number	OUT エンドポイントのエンドポイントナンバーを 0x0~0xF の間の任意の値に設定します。
IN エンドポイントトグルシーケンス	H_CHaBO_IN_EP_Control.IN_Toggle	IN エンドポイントのトグルシーケンスビットの初期値を設定します。また、トランザクション実行中、およびトランザクション完了後は IN エンドポイントのトグルシーケンスビットの状態を示します。
IN エンドポイントナンバー	H_CHaBO_IN_EP_Control.IN_EP_Number	IN エンドポイントのエンドポイントナンバーを 0x0~0xF の間の任意の値に設定します。
バルクオンリーサポート実行結果	H_CHaIntStat.BO_SupportCmp H_CHaIntStat.BO_SupportStop	バルクオンリーサポートの実行結果を示します。
トランザクションステータス	H_CHaIntStat.TotalSizeCmp, H_CHaIntStat.TranACK, H_CHaIntStat.TranErr, H_CHaIntStat.ChangeCondition	トランザクションの結果を示します。
トランザクションコンディションコード	H_CHaConditionCode	トランザクション結果の詳細を示します。
トランスポート状態	H_CHaBO_SupportControl.BO_TransportState	バルクオンリーサポート機能の実行において、実行中のトランスポートを示します。また、トランスポートがエラーで中止された場合は、エラーが発生したトランスポートを示します。
ステータス トランスポート 受信データ数	H_CSW_RcvDataSize	ステータストランスポートでの受信データ数 を示します。

1.3.9 オーディオクラス・アシスト機能

オーディオクラス・アシスト機能はサンプリング周波数 44.1kHz の 16bit 2 チャンネル PCM データをアイソクロナス転送で 1ms 周期で送信する際に使用します。この機能が有効の時、送信するデータパケットのサイズが 176Byte で 9 回連続トランザクションを行った後、180Byte で 1 回トランザクションを行います。その後再び 176Byte で 9 回連続トランザクションを行い、180Byte で 1 回トランザクションを行う、というデータパケットのサイズを自動で変更する動作を繰り返します。

このデータパケットサイズ変更のシーケンスは H_CHx{x=b-e}Config_1.Audio441 ビットが“1”である間保たれます。よってこのシーケンスを初期化する場合は H_CHx{x=b-e}Config_1.Audio441 ビットをクリアしてください。

表 1.30 オーディオクラスアシスト機能の制御項目

項目	レジスタ/ビット	説明
オーディオクラスアシスト機能	H_CHx{x=b-e}Config_1.Audio441	オーディオクラスアシスト機能を有効にします。OUT 方向のアイソクロナス転送の場合のみこの機能は使用できます。他の転送を行う場合はこのビットを“1”にセットしないで下さい。

1. 機能説明

1.3.10 ホストステート管理サポート機能

1.3.10.1 ホストステート

ホストでは、上位からの要求およびバスの状態によりステートを遷移させる必要があります。そのため、ステート管理はファームウェアが行います。ハードウェアは各ステートにおける各種設定およびネゴシエーションをサポートします。

ホストステートの遷移図を図 1.38 に示します。

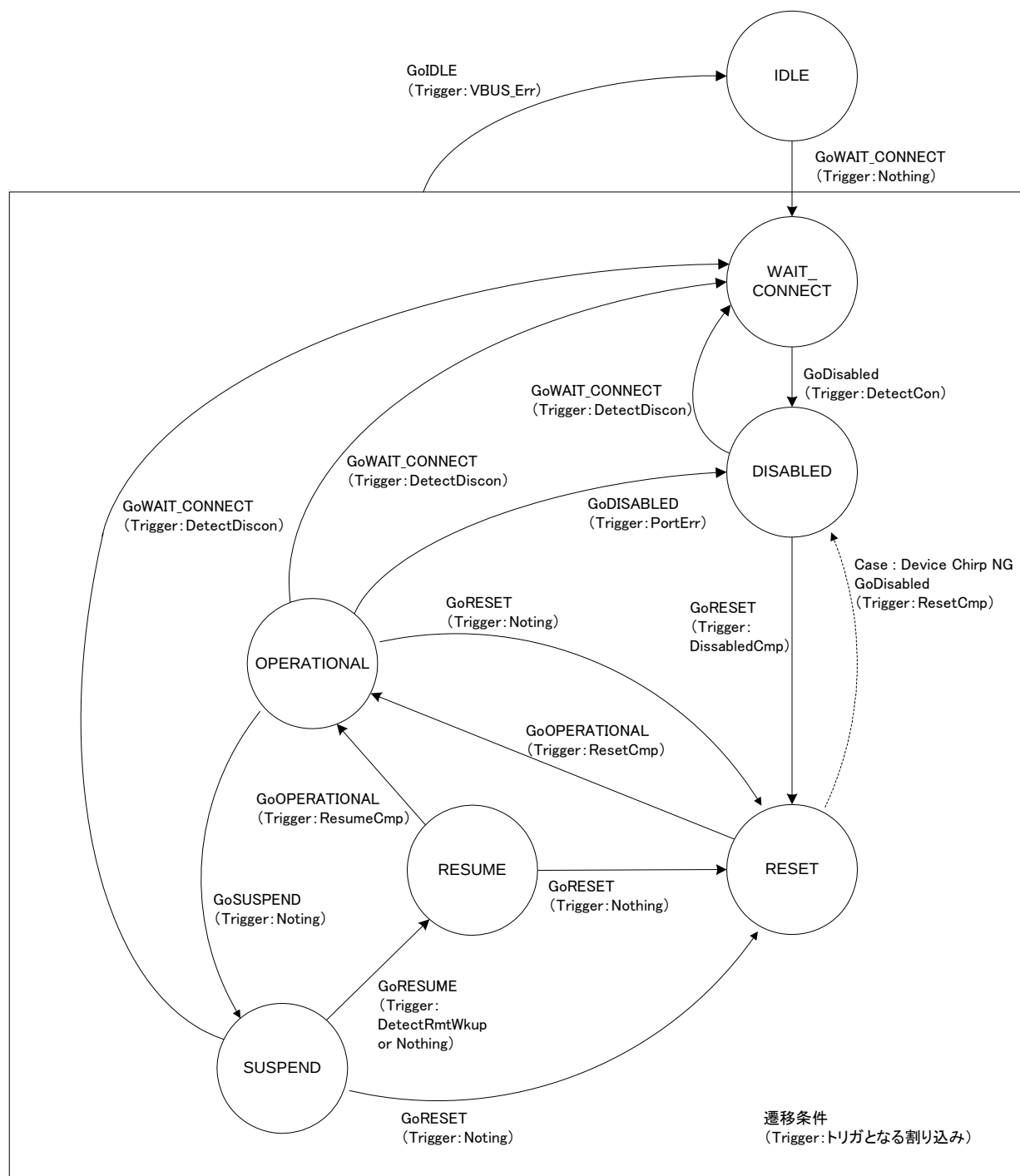


図 1.38 ホストステート遷移図

ホストステート管理サポート機能の設定項目およびステータスを表 1.31、表 1.32 に示します。

表 1.31 ホストステート管理サポート機能の設定項目およびステータス

項目	レジスタ/ビット	説明
ホストステート遷移実行	H_NegoControl_0.AutoMode	遷移させるホストステートを設定します。設定は以下のいずれかの項目になります。 GoIDLE GoWAIT_CONNECT GoDISABLED GoRESET GoOPERATIONAL GoSUSPEND GoRESUME GoWAIT_CONNECTtoDIS GoWAIT_CONNECTtoOP GoRESETtoOP GoRESUMetoOP GoSUSPENDtoOP
ホストステート遷移実行キャンセル	H_NegoControl_0.AutoModeCancel	現ホストステートでの処理を停止し、その状態で留まります。
ホストステートモニタ	H_NegoControl_0.HostState	現在のホストステート(下記)を示します。 IDLE WAIT_CONNECT DISABLED RESET OPERATIONAL SUSPEND RESUME
VBUS ステートモニタ	H_USB_Status.VBUS_State	VBUS の状態(正常/異常)を示します。
リモート・ウェークアップ受付許可	H_NegoControl_1.RmtWkupDetEnb	リモート・ウェークアップ受付を許可します。
チャープ完了ディセーブル	H_NegoControl_1.DisChirpFinish	デバイス Chirp が規定時間に完了しないときの動作モードを設定します。
VBUS 異常検出ステータス	H_SIE_IntStat_0.VBUS_Err	VBUS に異常が発生したことを示します。
接続検出ステータス	H_SIE_IntStat_0.DetectCon	ダウンストリームポートにデバイスが接続されたことを示します。
切断検出ステータス	H_SIE_IntStat_0.DetectDisCon	ダウンストリームポートからデバイスが切断されたことを示します。
リモート・ウェークアップ検出ステータス	H_SIE_IntStat_0.DetectRmtWkup	デバイスからのリモート・ウェークアップ信号を検出したことを示します。
デバイスチャープ正常検出ステータス	H_SIE_IntStat_0.DetectDevChirpOK	デバイスからのチャープ信号が正常であったことを示します。
デバイスチャープ異常検出ステータス	H_SIE_IntStat_0.DetectDevChirpNG	デバイスからのチャープ信号が異常であったことを示します。

(続く)

1. 機能説明

表 1.32 ホストステート管理サポート機能の設定項目およびステータス(続き)

項目	レジスタ/ビット	説明
リセット完了ステータス	H_SIE_IntStat_1.ResetCmp	USB リセットが正常に完了したことを示します。
サスペンド遷移完了ステータス	H_SIE_IntStat_1.SuspendCmp	サスペンドへの遷移が完了したことを示します。
レジューム完了ステータス	H_SIE_IntStat_1.ResumeCmp	レジュームが正常に完了したことを示します。
ポートスピード	H_NegoControl_1.PortSpeed	ダウンストリームポートの動作速度(HS/FS/LS)を示します。ポートに接続されたデバイスの動作
ラインステート	H_USB_Status.LineState	USB ケーブル上の信号状態を示します。
トランシーバ選択	H_XcvrControl.XcvrSelect	HS/FS/LSのいずれかのトランシーバを選択して有効にします。
ターミナル選択	H_XcvrControl.TermSelect	HS/FS のいずれかのターミナルを選択して有効にします。
オペレーションモード	H_XcvrControl.OpMode	HTM のオペレーションモードを設定します。

1.3.10.1.1 IDLE

USB ホスト機能を初期化するステートです。ホスト機能をイネーブルとしたときのデフォルトステートです。

本ステート以外のステートにおいて、VBUS_Err を検出した時に、本ステートに遷移させる必要があります。

遷移を実行させるためには、H_NegoControl_0 レジスタに 0x80(H_NegoControl_0.AutoModeCancel に“1”、かつ H_NegoControl_0.AutoMode に“0x0”)をライトして実行中ステートの動作を停止させます。

H_NegoControl_0.AutoModeCancel ビットは停止処理が完了することで“0”になります(60MHz クロックで 6 サイクル程度必要)ので、H_NegoControl_0.AutoModeCancel ビットが“0”になるのを確認してから、同レジスタに“0x01”をライト(すなわちホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoIDLE”を設定)してください。そうすることで本ステートに入ります。

本ステートでは、以下の設定を自動的に実行します。

- USB ホストのトランザクション実行機能を即時停止する。
- ポートを FS モードにして“NonDriving”とする。
- VBUSEN_A をオフする。
- 接続検出、切断検出、リモート・ウェークアップ検出、デバイスチャープ検出の全検出機能をオフする。

1.3.10.1.2 WAIT_CONNECT

ダウンストリームポートにデバイスが接続されるのを待つステートです。

“IDLE”ステートにおいては上位からの要求がある時に、また“DISABLED”、“OPERATIONAL”および“SUSPEND”の各ステートではデバイスの切断を検出した時に、一旦、本ステートに入り、相手の接続を待つ必要があります。

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECT”を設定した時に、本ステートに入ります。

本ステートでは、まず以下の設定を自動的に実行します。

- USB ホストのトランザクション実行機能を即時停止する。
- ポートを FS モードにして“PowerDown”とする。

- VBUSEN_A をオンする。
- 接続検出、切断検出、リモート・ウェークアップ検出、デバイスチャープ検出の全検出機能をオフする。

次に、バスパワーデバイスの内部電源安定時間を待って、接続検出機能を自動的にオンにし、デバイスの接続を待ちます。このとき、VBUSEN_A をオンしてからデバイスの接続検出までの時間は、ハードウェアでは管理しておりません。この時間は、必要に応じてファームウェアにて管理してください。

FS または HS デバイスが接続された場合、ラインステートが“J”として参照できます。一方、LS デバイスが接続された場合、ラインステートが“K”として参照できます。このいずれかの状態が 2.5us 以上継続されることでデバイスの接続を検出し、LS デバイスの接続を検出した場合には、ポートを LS モードにします。

接続が検出されると、切断検出機能を自動的にオンします。

その後、デバウンスインターバル期間に切断が検出されない場合、ファームウェアに対して接続検出ステータス(H_SIE_IntStat_0.DetectCon)を発行し、接続検出機能および切断検出機能を自動的にオフします。一方、切断が検出された場合、切断検出機能を自動的にオフし、自動的に接続検出からやり直します。

1.3.10.1.3 DISABLED

ダウンストリームポートにデバイスが接続された状態で、バス上での信号送受信を行わないステートです。

“WAIT_CONNECT”ステートで接続を検出した時、“RESET”ステートで異常なデバイスからの Chirp を検出した時、“OPERATIONAL”ステートでポートエラーを検出した時は、本ステートに入ります。

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoDISABLED”を設定した時に、本ステートに入ります。

本ステートでは、以下の設定を自動的に実行します。

- 現トランザクションの完了を待って、USB ホストのトランザクション実行機能を停止する。
- 本ステートに入る際に、HS モードであった場合はポートを FS モードにし、FS または LS モードの場合はポートは入る際のモードを持続する。
- ポートを“PowerDown”とする。

次に、切断検出ディセーブル期間完了後に、以下の処理を自動的に実行します。

- 切断検出機能をオンする。
- ディセーブルド遷移完了ステータス(H_SIE_IntStat_1.DisabledCmp)を発行する。

1.3.10.1.4 RESET

ダウンストリームポートに USB リセットを発行するステートです。

“DISABLED”ステートでディセーブルド遷移完了ステータスが発行された時は、本ステートに入り USB リセットを発行します。

また、上位から要求があった場合には、USB のいかなるステート(“OPERATIONAL”, “SUSPEND”, “RESUME”)からでも、本ステートに遷移することができます。

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定した時に、本ステートに入ります。

本ステートでは、まず以下の設定を自動的に実行します。

- 現トランザクションの完了を待って、USB ホストのトランザクション実行機能を停止する。

1. 機能説明

- ポートを HS モードにして“NormalOperation”とする(USB ケーブル上の信号にリセット信号 SE0 をドライブします)。
- 接続検出、切断検出およびリモート・ウェークアップ検出機能をオフする。
- デバイスチャープ検出機能をオンする。

デバイスからの Chirp は“HS K”により検出します。ラインステートが“K”として 2.5us 以上継続することにより検出され、USB リセットの発行から規定時間以内に終了された場合に正常な Chirp として検出します。また、規定時間以内に終了されない場合には、異常な Chirp として検出します。

検出結果により以下の処理を自動的に行います。

- (1) 正常なデバイスからの Chirp を検出した場合
デバイスからの Chirp 完了に伴い、ホストより“HS K”(Chirp K)、“HS J”(Chirp J)を交互に連続して送出します。ホストが Chirp の送出を完了した時点で、ファームウェアに対し、リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します。
ポートは HS モードのままです。
- (2) 異常なデバイスからの Chirp を検出した場合
規定時間を過ぎた時点で、ファームウェアに対し、デバイスチャープ異常検出ステータス(H_SIE_IntStat_0.DetectDevChirpNG)を発行します。その後の動作は、チャープ完了ディセーブル(H_NegoControl_1.DisChirpFinish)の設定により 2 つの動作モードが選択できます。詳細は「1.3.10.2.4.2 異常なデバイスの Chirp を検出した場合」を参照してください。
- (3) デバイスからの Chirp を検出せず、接続相手が FS の場合
規定時間の USB リセットを発行した後、ポートを FS モードにします。
ファームウェアに対し、リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します。
- (4) 接続相手が LS の場合
規定時間の USB リセットを発行した後、ポートを LS モードにします。
ファームウェアに対し、リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します。

1.3.10.1.5 OPERATIONAL

USB のトランザクションを実行するステートです。

“RESET”または“RESUME”の完了後、本ステートに入りトランザクションを実行します。

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoOPERATIONAL”を設定した時に、本ステートに入ります。

本ステートでは、以下の設定を自動的に実行します。

- ポートを“NormalOperation”とする。
- USB ホストのトランザクション実行機能をイネーブルにする。
- 切断検出機能をオンする。

1.3.10.1.6 SUSPEND

USB をサスペンドにするステートです。

USB のバスの使用を停止させるときに、“OPERATIONAL”から本ステートに遷移させます。

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoSUSPEND”を設定した時に、本ステートに入ります。

本ステートでは、まず以下の設定を自動的に実行します。

- 切断検出機能およびリモート・ウェークアップ検出機能をオフする。
- 現トランザクションの完了を待って、USB ホストのトランザクション実行機能を停止する。
- ポートが HS モードであった場合、FS モードに設定する。一方、FS モードまたは LS モードであった場合は、そのモードを維持する。
- ポートを“PowerDown”とする。

次に、切断およびリモート・ウェークアップ検出ディセーブル期間完了後に、以下の処理を自動的に実行します。

- 切断検出機能をオンする。
- リモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)がイネーブルの場合、リモート・ウェークアップ検出機能をオンする。
- サスペンド遷移完了ステータス(H_SIE_IntStat_1.SuspendCmp)を発行する。

さらに、リモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)がイネーブルの場合、リモート・ウェークアップ信号(2.5us 以上の“K”の継続)を検出することで、ファームウェアに対し、リモート・ウェークアップ検出ステータス(H_SIE_IntStat_0.DetectRmtWkup)を発行します。

1.3.10.1.7 RESUME

ダウンストリームポートに USB レジューム信号を発行するステートです。

USB デバイスをサスペンド状態から復帰させるために、“SUSPEND”から本ステートに遷移させます。

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUME”を設定した時に、本ステートに入ります。

本ステートでは、切断およびリモート・ウェークアップ検出機能を自動的にオフします。

そして、規定時間のレジューム信号(“K”)を発行します。

レジューム信号の発行完了時に、ポートを“SUSPEND”に入る前のモード設定に戻し、“NormalOperation”とします。

また、ファームウェアに対し、レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行します。

1. 機能説明

1.3.10.2 検出機能

1.3.10.2.1 VBUS異常検出

VBUSFLG_A 入力端子のレベル変化(H→L)により VBUS の異常を検出します。VBUS 異常を検出した場合の実行手順は以下のとおりです。下記(2)は本 LSI のハードウェアが自動的に実行します。

- (1) VBUSFLG_A(外付け USB パワースイッチのエラー発生フラグ)入力端子が L(エラー発生)になります(T0)。
- (2) ファームウェアに対し、VBUS 異常検出ステータス(USB_HostIntStat.VBUS_Err)を発行します(T0)。

なお、ホストは VBUS の異常を検出すると、直ちに VBUS をオフする必要があります。そのため、ファームウェアは、VBUS 異常検出ステータスを認識したら、H_NegoControl_0 レジスタに 0x80(H_NegoControl_0.AutoModeCancel に“1”、かつ H_NegoControl_0.AutoMode に“0x0”)をライトして実行中ステートの動作を停止させます。H_NegoControl_0.AutoModeCancel ビットは停止処理が完了することで“0”になります(60MHz クロックで 6 サイクル程度必要)ので、H_NegoControl_0.AutoModeCancel ビットが“0”になるのを確認してから、同レジスタに 0x01 をライト(すなわちホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoIDLE”を設定)してください。それにより IDLE ステートに遷移し、VBUSEN_A 端子論理がディセーブルとなり、VBUS をオフすることができます。

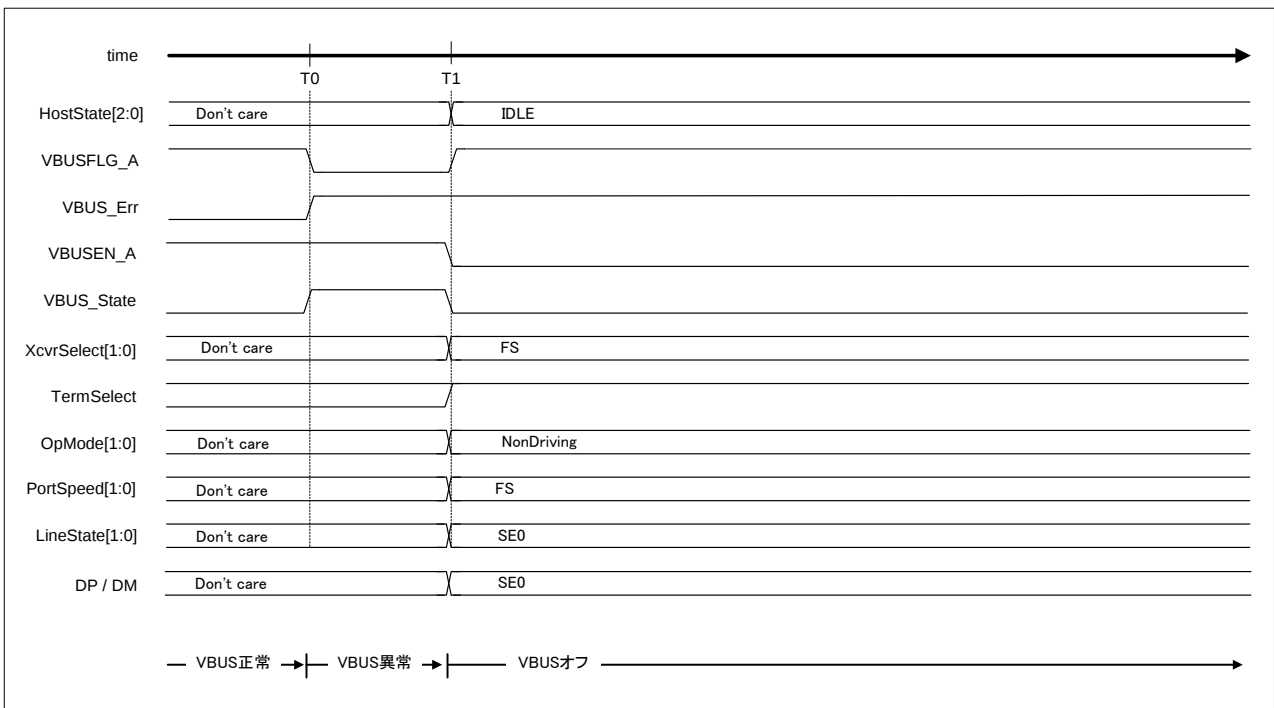


図 1.39 VBUS 異常検出 Timing

表 1.33 VBUS 異常検出 Timing Values

Timing Parameter	Description	Value
T0	VBUSFLG_A(外付け USB パワースイッチのエラー発生フラグ)入力端子が L(エラー)になる。 VBUS 異常検出ステータス(USB_HostIntStat.VBUS_Err)を発行する。 (H/W)	0 (reference)
T1(参考)	H_NegoControl_0 に 0x80 をライト後、0x01 をライトして“IDLE”ステートに遷移させる。(F/W)	T1

1.3.10.2.2 切断検出

デバイスの切断検出を“DISABLED”、“OPERATIONAL”、“SUSPEND”の各ステートにおいて行います。

切断が検出された時、VBUS をオフせず、接続検出からやり直す場合にはホストステートを“WAIT_CONNECT”に遷移させてください。一方、VBUS をオフする場合にはホストステートを“IDLE”に遷移させてください。

1.3.10.2.2.1 HSデバイスが切断された場合

HS デバイスの切断は、“OPERATIONAL”ステートにて検出します。

HS デバイスが切断された場合の実行手順は以下のとおりです。下記(2)～(3)は本 LSI のハードウェアが自動的に実行します。

- (1) デバイスが切断されます(T0)。
- (2) uSOF(HS_SOF)の EOP 期間において切断検出を行い、3 つの連続した uSOF で切断を検出した場合に切断状態と判断します(T1)。
- (3) ファームウェアに対し、切断検出ステータス(H_SIE_IntStat_0.DetectDiscon)を発行します(T1)。

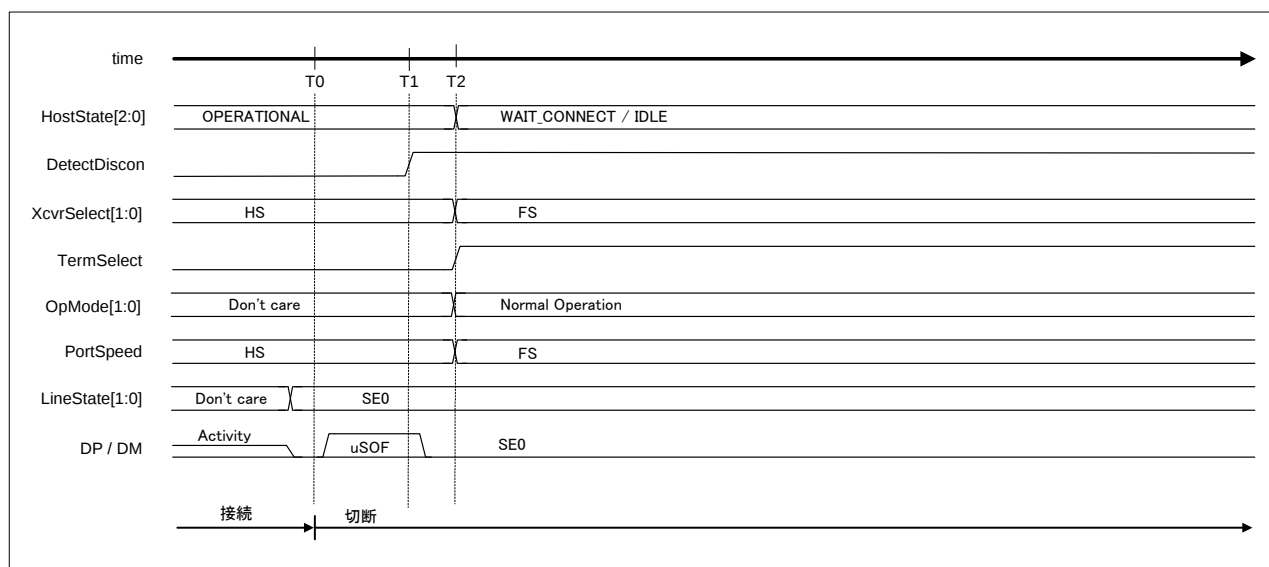


図 1.40 切断検出 Timing (HS mode)

1. 機能説明

表 1.34 切断検出 Timing Values (HS mode)

Timing Parameter	Description	Value
T0	デバイスが切断される。	0 (reference)
T1	切断検出ステータス(H_SIE_IntStat_0.DetectDiscon)を発行する。 (H/W)	T1
T2(参考)	ホス ト ス テ ー ト 遷 移 実 行 (H_NegoControl_0.AutoMode) に "GoWAIT_CONNECT"を設定する。(F/W)	T2

1.3.10.2.2.2 FSまたはLSデバイスが切断された場合

FS または LS デバイスの切断は、“DISABLED”、“OPERATIONAL”、“SUSPEND”ステートにて検出します。

FS または LS デバイスが切断された場合の実行手順は以下のとおりです。下記(2)～(3)は本 LSI のハードウェアが自動的に実行します。

- (1) デバイスが切断されます(T0)。
- (2) 信号線の状態から切断を検出します(T1)。
- (3) ファームウェアに対し、切断検出ステータス(H_SIE_IntStat_0.DetectDiscon)を発行します(T1)。

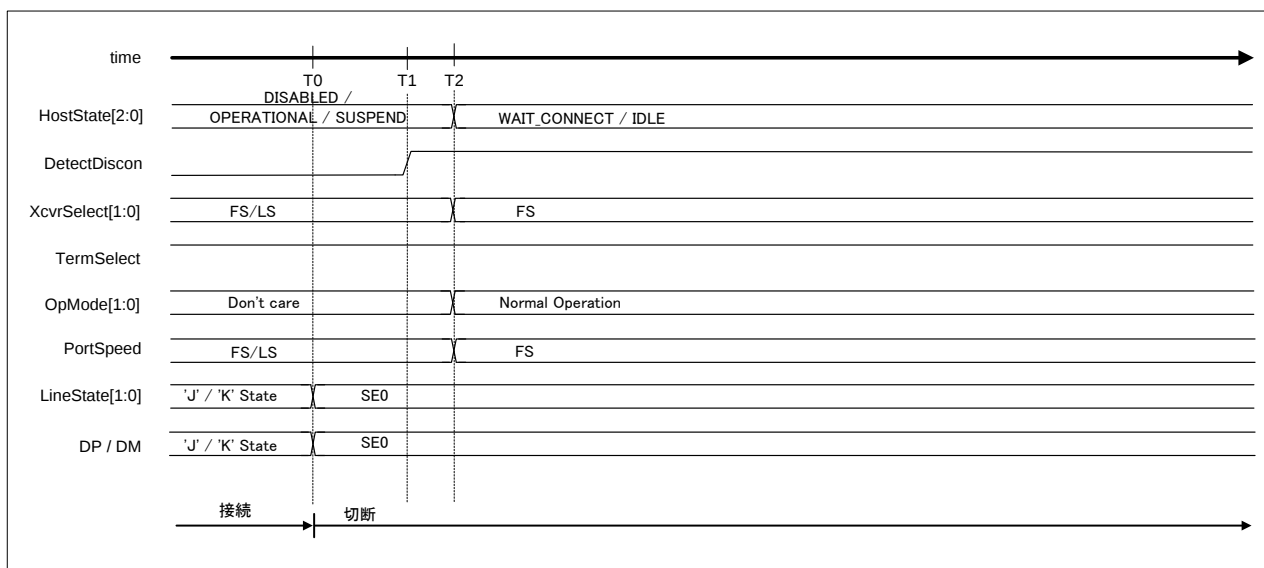


図 1.41 切断検出 Timing (FS または LS mode)

表 1.35 切断検出 Timing Values (FS または LS mode)

Timing Parameter	Description	Value
T0	デバイスが切断される。	0 (reference)
T1	切断検出ステータス(H_SIE_IntStat_0.DetectDiscon)を発行する。(H/W)	$T0 + 2.5\mu s < T1 \{T_{DDIS}\}$
T2(参考)	ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECT”を設定する。(F/W)	規定なし

1. 機能説明

1.3.10.2.3 リモート・ウェークアップ検出

リモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)がイネーブルの場合、“SUSPEND”ステートにおいてリモート・ウェークアップ検出を行います。

1.3.10.2.3.1 HSデバイスが接続されている場合

HS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(3)は本 LSI のハードウェアが自動的に実行します。

- (1) デバイスがリモート・ウェークアップ信号(K)を送出開始します(T0)。
- (2) ホストがリモート・ウェークアップ信号(K)を検出します(T1)。
- (3) ファームウェアに対し、リモート・ウェークアップ検出ステータス(H_SIE_IntStat_0.DetectRmtWkup)を発行します(T1)。

なお、ホストはデバイスのリモート・ウェークアップ検出から 1ms 以内にレジューム信号(“K”)を発行する必要があります。そのため、ファームウェアは、リモート・ウェークアップ検出ステータスを直ちに認識し、ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUME”を 900us 以内に設定してください。

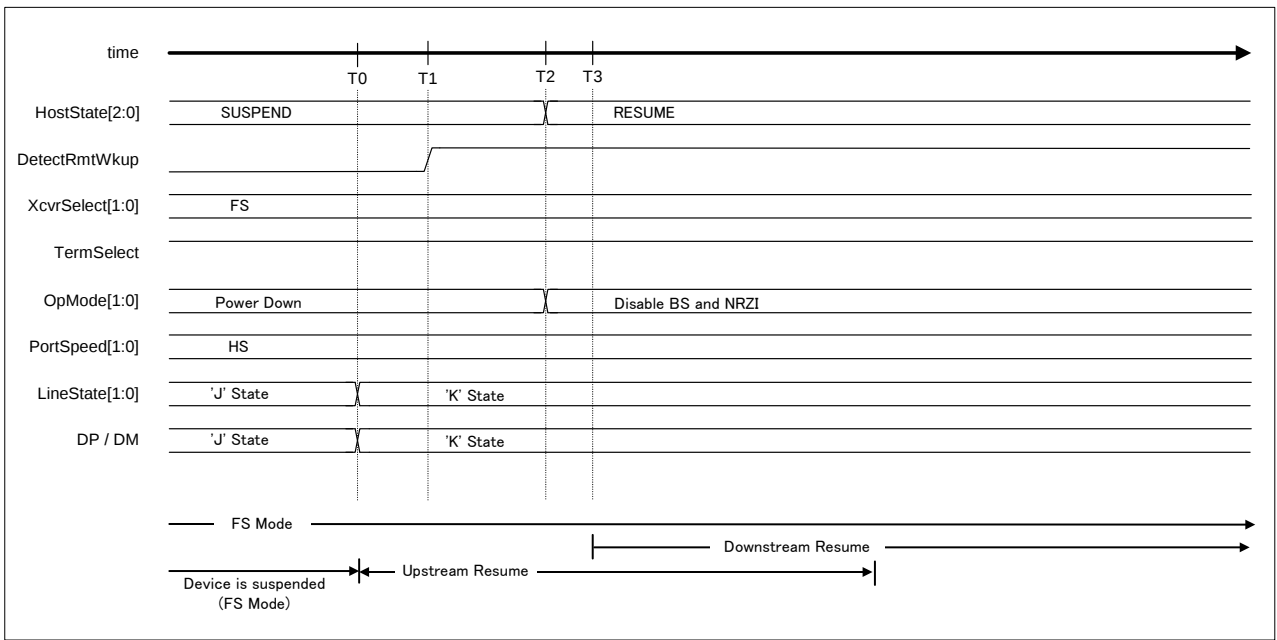


図 1.42 Remote Wakeup Timing (HS mode)

表 1.36 Remote Wakeup Timing Values (HS mode)

Timing Parameter	Description	Value
T0	デバイスがリモート・ウェークアップ信号(K)を送出開始する。	0 (reference)
T1	リモート・ウェークアップ信号(K)を検出する。 リモート・ウェークアップ検出ステータスを発行する。(H/W)	$T0 + 2.5\mu s\{T_{URLK}\} < T1$
T2(参考)	ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUME”を設定する。(F/W)	$T2 < T1 + 900\mu s$
T3(参考)	ホストがレジューム信号(“K”)の発行を開始する。(H/W)	$T3 < T0 + 1ms\{T_{URSM}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.2.3.2 FSデバイスが接続されている場合

FS デバイスが接続されている場合の実行手順は、HS デバイスが接続されている場合の実行手順と同じです。

実行手順は、「1.3.10.2.3.1 HS デバイスが接続されている場合」を参照してください。

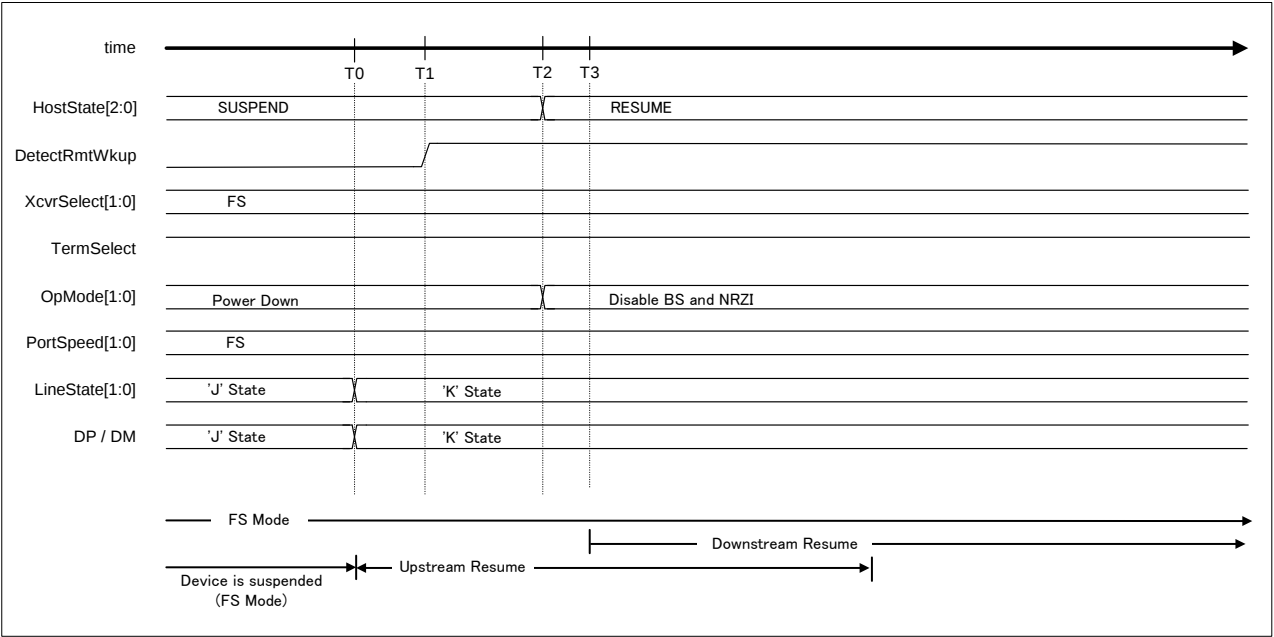


図 1.43 Remote Wakeup Timing (FS mode)

表 1.37 Remote Wakeup Timing Values (FS mode)

Timing Parameter	Description	Value
T0	デバイスがリモート・ウェークアップ信号(K)を送出開始する。	0 (reference)
T1	リモート・ウェークアップ信号(K)を検出する。 リモート・ウェークアップ検出ステータスを発行する。(H/W)	$T0 + 2.5\mu s < T1 \{T_{URLK}\}$
T2(参考)	ホストステート遷移実行 (H_NegoControl_0.AutoMode) に “GoRESUME”を設定する。(F/W)	$T2 < T1 + 900\mu s$
T3(参考)	ホストがレジューム信号(“K”)の発行を開始する。(H/W)	$T3 < T0 + 1ms\{T_{URSM}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.3.10.2.3.3 LSデバイスが接続されている場合

LS デバイスが接続されている場合の実行手順は、HS デバイスが接続されている場合の実行手順と同じです。

実行手順は、「1.3.10.2.3.1 HS デバイスが接続されている場合」を参照してください。

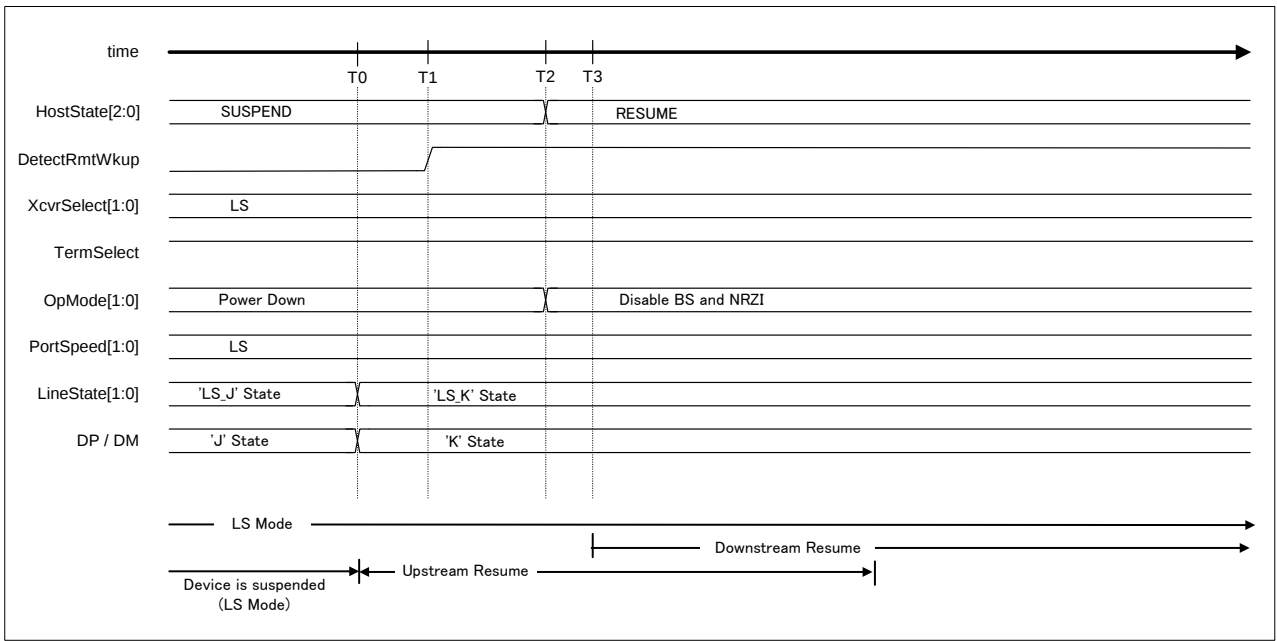


図 1.44 Remote Wakeup Timing (LS mode)

表 1.38 Remote Wakeup Timing Values (LS mode)

Timing Parameter	Description	Value
T0	デバイスがリモート・ウェークアップ信号("K")を送出開始する。	0 (reference)
T1	リモート・ウェークアップ信号("K")を検出する。 リモート・ウェークアップ検出ステータスを発行する。(H/W)	$T0 + 2.5\mu s < T1 \{T_{URLK}\}$
T2(参考)	ホストステート遷移実行 (H_NegoControl_0.AutoMode) に "GoRESUME"を設定する。(F/W)	$T2 < T1 + 900\mu s$
T3(参考)	ホストがレジューム信号("K")の発行を開始する。(H/W)	$T3 < T0 + 1ms\{T_{URSM}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.2.4 デバイスチャープ検出機能

デバイスの Chirp を検出します。

デバイスチャープ検出機能は“RESET”ステートにてオンします。

1.3.10.2.4.1 正しいデバイスのChirpを検出した場合

デバイスの Chirp を検出する手順は以下の通りです。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定します(T0)。
- (2) デバイスチャープ検出機能をオンします(T0)。
- (3) デバイスが Chirp を送出します(T1)。
- (4) ラインステート(H_USB_Status.LineState[1:0])に“K”が規定時間以上継続することでデバイスの Chirp を認識します(T2)。
- (5) デバイスの Chirp がリセット開始から規定時間以内に終了する(ラインステート(H_USB_Status.LineState[1:0])が“SE0”になる)ことをもってデバイスチャープ正常検出ステータス(H_SIE_IntStat0.DetectDevChirpOK)を発行します(T3)。
- (6) デバイスの Chirp を検出するとデバイスチャープ検出機能をオフします(T3)。

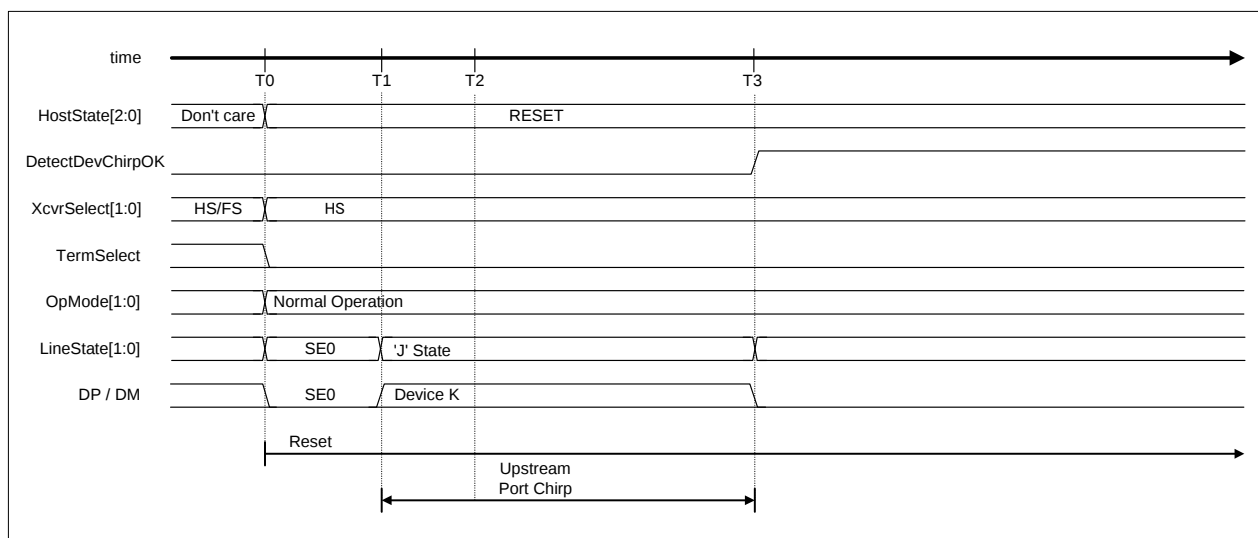


図 1.45 Device Chirp Timing

表 1.39 Device Chirp Timing Values

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESET”を設定する。(F/W)	0 (reference)
T1	デバイスが Chirp を開始。	$T0 < T1 < T0 + 6.0\text{ms}$
T2	デバイスの Chirp を認識する。(H/W)	$T1 + 2.5\mu\text{s} \{T_{\text{FILT}}\} < T2$
T3	デバイスが Chirp を終了。 デバイスチャープ検出機能をオフする。 デバイスチャープ正常検出ステータス(DetectDevChirpOK)を発行する。(H/W)	$T1 + 1.0\text{ms} \{T_{\text{UCH}}\} < T3 < T0 + 7.0\text{ms} \{T_{\text{UCHEND}}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.3.10.2.4.2 異常なデバイスのChirpを検出した場合

デバイスチャープ検出機能は、デバイスの Chirp が規定時間に終了しない場合を異常と見なし、ステータスを発行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定します(T0)。
- (2) デバイスチャープ検出機能をオンします(T0)。
- (3) デバイスが Chirp を送出します(T1)。
- (4) ラインステート(H_USB_Status.LineState[1:0])に“K”が規定時間以上継続することでデバイスの Chirp を認識します(T2)。
- (5) リセットの開始から規定時間以内にデバイスの Chirp が終了しないために異常と見なし、デバイスチャープ異常検出ステータス(H_SIE_IntStat_0.DetectDevChirpNG)を発行します(T3)
- (6) デバイスチャープ検出機能をオフします(T3)。

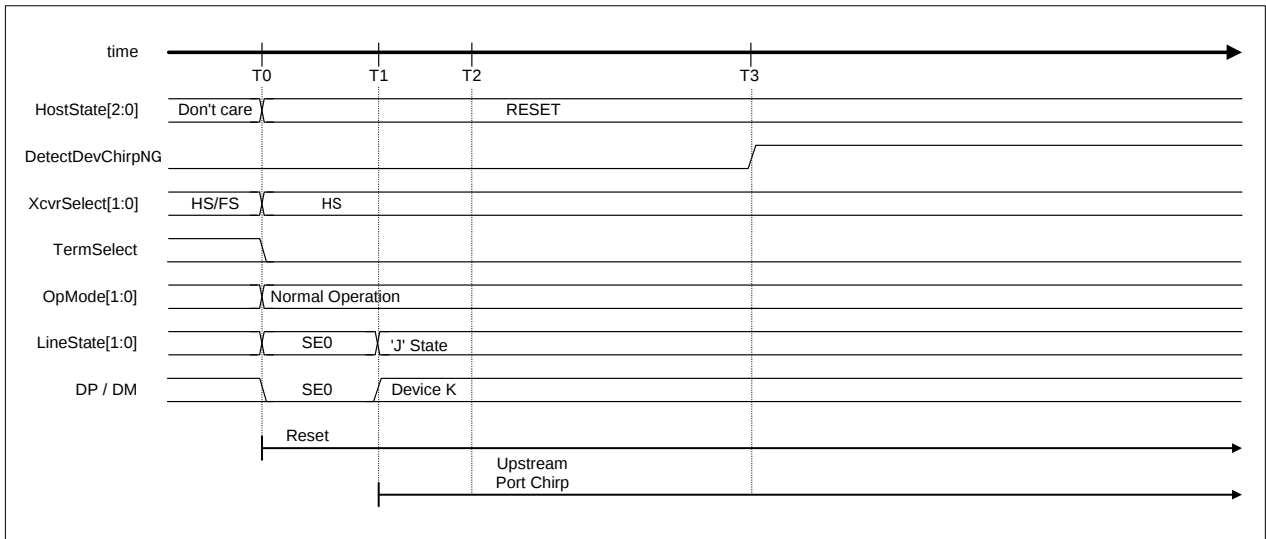


図 1.46 Device Chirp Timing(NG)

表 1.40 Device Chirp Timing(NG) Value

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESET”を設定する。(F/W)	0 (reference)
T1	デバイスが Chirp を開始。	$T0 < T1 < T0 + 6.0\text{ms}$
T2	デバイスの Chirp を認識する。(H/W)	$T1 + 2.5\mu\text{s} \{T_{\text{FILT}}\} < T2$
T3	デバイスチャープ異常検出ステータス(DetectDevChirpNG)を発行する。(H/W)	$T0 + 7\text{ms} \{T_{\text{UCHEND}}\} < T3$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.2.5 ポートエラー検出

“OPERATIONAL”ステートにおいてポートエラーを検出します。

受信中のパケットにおいて、(マイクロ)フレームの終端に達しても EOP を検出できない場合に、ポートエラーと判断します。

ホストはポートエラーを検出すると、ファームウェアに対してポートエラー検出ステータス(H_FrameIntStat.PortErr)を発行し、トランザクションを即時停止します。以降、SOF を含めトランザクションの発行は行いません。

ポートエラーが発生した場合、ファームウェアは以下の処理を行ってください。

- (1) H_NegoControl_0.AutoMode に“GoDISABLED”を設定します。
- (2) ChipReset.ResetMTM に“1”を設定し、トランシーバマクロをリセットします。
- (3) 60MHz クロックで 3 サイクル以上の時間経過後、ChipReset.ResetMTM に“0”を設定し、トランシーバマクロのリセットを解除します。

1. 機能説明

1.3.10.3 ホストステート管理サポート機能の単体説明

1.3.10.3.1 GoIDLE

H_NegoControl_0 レジスタに 0x80(H_NegoControl_0.AutoModeCancel に“1”、かつ H_NegoControl_0.AutoMode に 0x0)をライトして実行中ステートの動作を停止させます。H_NegoControl_0.AutoModeCancel ビットは停止処理が完了することで“0”になります(60MHz クロックで 6 サイクル程度必要)ので、H_NegoControl_0.AutoModeCancel ビットが“0”になるのを確認してから、同レジスタに 0x01 をライト(すなわちホストステート遷移実行(H_NegoControl_0.AutoMode)に GoIDLE を設定)してください。そうすることで、“IDLE”遷移に必要な処理を本 LSI のハードウェアが自動的に実行します。

下記(3)～(8)は本 LSI のハードウェアが自動的に実行します。

- (1) H_NegoControl_0.AutoModeCancel に 0x80(H_NegoControl_0.AutoModeCancel に “1”、かつ H_NegoControl_0.AutoMode に 0x0)をライトします(T0)。
- (2) H_NegoControl_0.AutoModeCancel ビットが “0” になったことを確認し、H_NegoControl_0 に 0x01(H_NegoControl_0.AutoMode に 0x1)をライトします(T1)。
- (3) ホストステートモニタ(H_NegoControl_0.HostState)を“IDLE”に設定します(T1)。
- (4) VBUSEN_A をオフします(T1)。
- (5) トランシーバ選択(H_XcvrControl.XcvrSelect)とターミナル選択(H_XcvrControl.TermSelect)を FS モードに設定します(T1)。
- (6) オペレーションモード(H_XcvrControl.OpMode[1:0])を“NonDriving”に設定します(T1)。
- (7) USB ホストのトランザクション実行機能を即時停止します(T1)。
- (8) 接続検出、切断検出、リモート・ウェークアップ検出、デバイスチャープ検出の全検出機能をオフします(T1)。

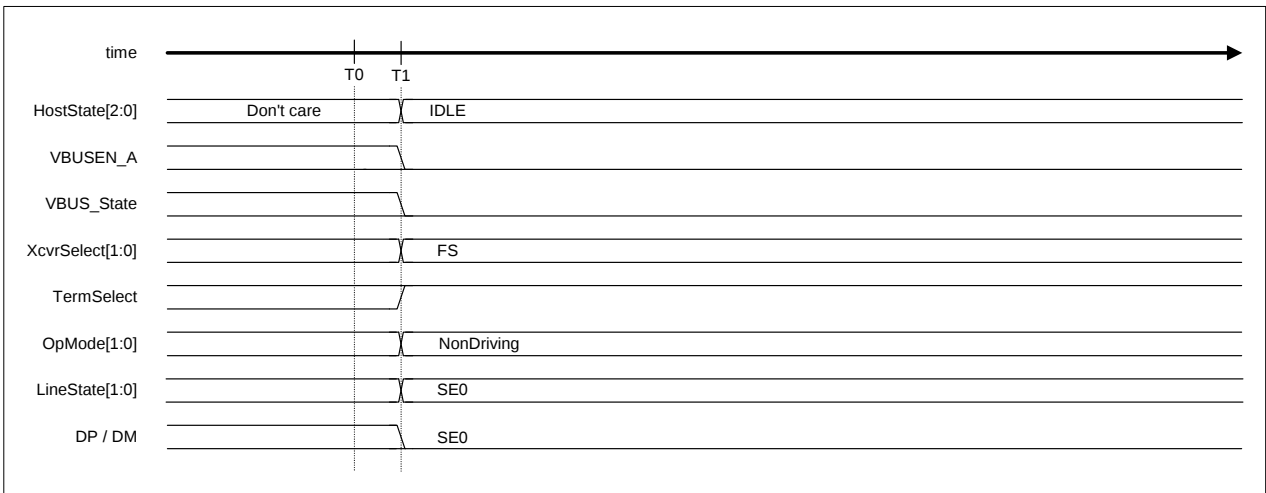


図 1.47 GoIDLE Timing

表 1.41 GoIDLE Timing Value

Timing Parameter	Description	Value
T0	実行中ステートの動作を停止する。(F/W)	0 (reference)
T1	ホストステート遷移実行キャンセルビットが“0”になったことを確認し、H_NegoControl_0.AutoMode に“GoIDLE”を設定する。 (F/W) VBUSEN_A をオフする。 トランシーバ選択を FS モードに設定する。 ターミナル選択を FS モードに設定する。 オペレーションモードを“NonDrivinvg”に設定する。 トランザクション実行機能を即時停止する。 接続検出、切断検出、リモート・ウェークアップ検出、デバイスチャージ検出をオフする。(H/W)	$T0 + 5\text{cycle}(60\text{MHz}) < T1$

1.3.10.3.2 GoWAIT_CONNECT

ホストステート遷移実行(H_NegoControl0.AutoMode)に“GoWAIT_CONNECT”を設定すると、“WAIT_CONNECT”遷移に必要な処理を本 LSI のハードウェアが自動的に実行します。

なお、HS デバイスはこの時点では FS デバイスとして接続します。その後に行われるリセット動作の HS Detection Handshake によって HS デバイスとして動作するようになります。

1.3.10.3.2.1 FSデバイスが接続された場合

FS デバイスが接続された場合の実行手順は以下のとおりです。下記(2)～(12)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECT”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“WAIT_CONNECT”に設定します(T0)。
- (3) VBUSEN_A をオンします(T0)。
- (4) トランシーバ選択(H_XcvrControl.XcvrSelect)とターミナル選択(H_XcvrControl.TermSelect)を FS モードに設定します(T0)。
- (5) オペレーションモード(H_XcvrControl.OpMode[1:0])を“PowerDown”に設定します(T0)。
- (6) ポートスピード(H_NegoControl_1.PortSpeed[1:0])を“FS”に設定します(T0)。
- (7) デバイス内部電源の安定化の為に一定期間待ち、接続検出機能をオンします(T1)。
- (8) FS デバイスが接続されるとラインステート(H_USB_Status.LineState[1:0])に“J”が現れます(T2)。
- (9) ラインステート(H_USB_Status.LineState[1:0])の“J”が 2.5us 以上継続することをもって FS デバイスの接続と見なします(T3)。
- (10) 切断検出機能をオンします(T3)。
- (11) デバウンスインターバル期間中に切断が検出されない場合に接続検出ステータス(H_SIE_IntStat_0.DetectCon)を発行します(T4)。もし、この期間中に切断が検出された場合には、切断検出機能をオフにし、(8)に戻っての接続検出からやり直します。切断検出ステータス(H_SIE_IntStat_0.DetectDiscon)は発行しません。
- (12) 切断検出機能と接続検出機能をオフします(T4)。

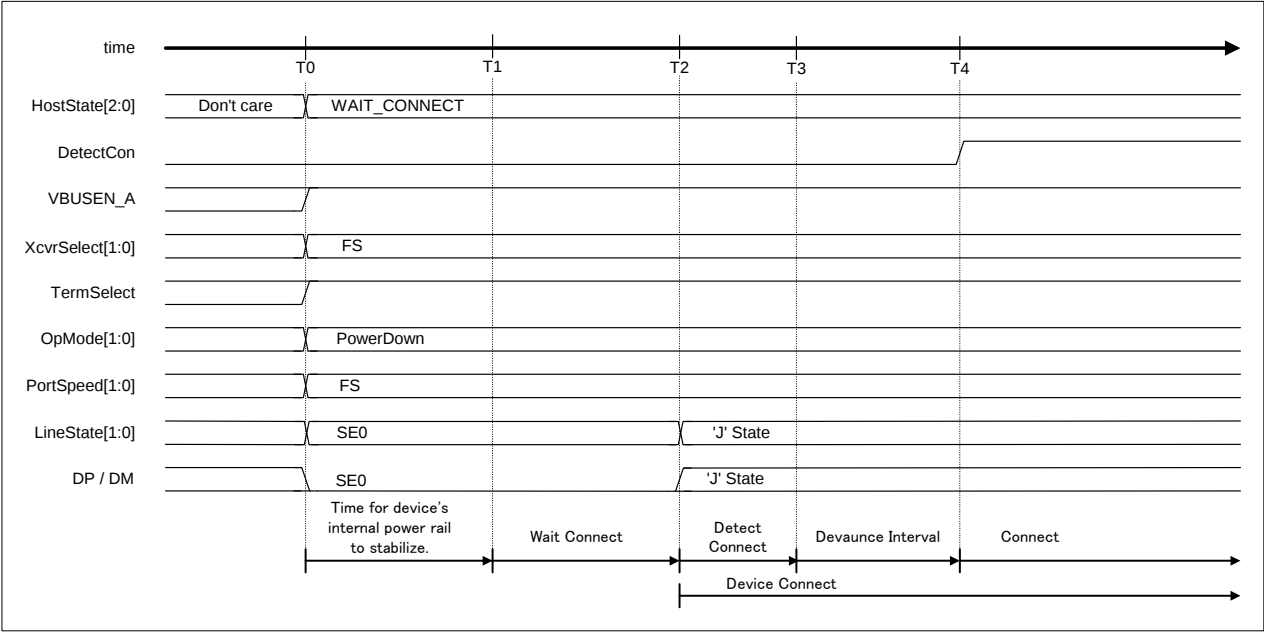


図 1.48 Device Attach Timing(FS mode)

表 1.42 Device Attach Timing Values(FS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoWAIT_CONNECT”を設定する。(F/W)	0 (reference)
T1	接続検出機能をオンする。(H/W)	$T0 + 100\text{ms}\{T_{\text{SIGATT}}\} < T1$
T2	デバイスが接続される。	T2
T3	切断検出機能をオンする。(H/W)	$T2 + 2.5\text{us}\{T_{\text{DCNN}}\} < T3$
T4	接続検出ステータス(DetectCon)を発行する。 切断検出機能と接続検出機能をオフする。(H/W)	$T3 + 100\text{ms}\{T_{\text{ATTDDB}}\} < T4$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.2.2 LSデバイスが接続された場合

LS デバイスが接続された場合の実行手順は以下のとおりです。下記(2)～(14)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECT”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“WAIT_CONNECT”に設定します(T0)。
- (3) VBUSEN_A をオンします(T0)。
- (4) トランシーバ選択(H_XcvtControl.XcvtSelect)とターミナル選択(H_XcvtControl.TermSelect)をFSモードに設定します(T0)。
- (5) オペレーションモード(H_XcvtControl.OpMode[1:0])を“PowerDown”に設定します(T0)。
- (6) ポートスピード(H_NegoControl_1.PortSpeed[1:0])を“FS”に設定します(T0)。
- (7) デバイス内部電源の安定化の為に 100ms 待ち、接続検出機能をオンします(T1)。
- (8) LS デバイスが接続されるとラインステート(H_USB_Status.LineState[1:0])に“K”が現れます(T2)。
- (9) ラインステート(H_USB_Status.LineState[1:0])の“K”が 2.5us 以上継続することをもって LS デバイスの接続と見なします(T3)。
- (10) トランシーバ選択(H_XcvtControl.XcvtSelect[1:0])を“LS”に設定します(T3)。このため、ラインステート(H_USB_Status.LineState[1:0])の極性が LS になり、ラインステート(H_USB_Status.LineState[1:0])には“J”が現れます。
- (11) ポートスピード(H_NegoControl_1.PortSpeed[1:0])を“LS”に設定します(T3)。
- (12) 切断検出機能をオンします(T3)。
- (13) デバウンスインターバル期間中に切断が検出されない場合に接続検出ステータス(H_SIE_IntStat_0.DetectCon)を発行します(T4)。もし、この期間中に切断が検出された場合には、切断検出機能をオフにし、トランシーバ選択(H_XcvtControl.XcvtSelect)とポートスピード(H_NegoControl_1.PortSpeed[1:0])をともに“FS”に設定し、(8)に戻っての接続検出からやり直します。切断検出ステータス(H_SIE_IntStat_0.DetectDiscon)は発行しません。
- (14) 切断検出機能と接続検出機能をオフします(T4)。

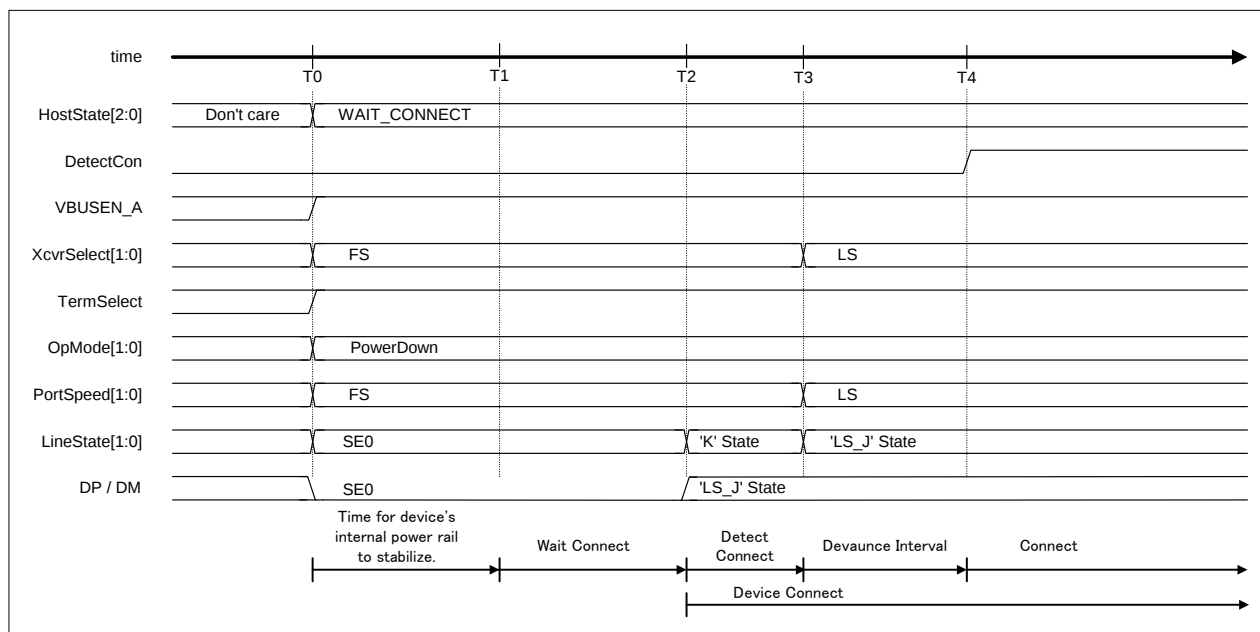


図 1.49 Device Attach Timing(LS mode)

表 1.43 Device Attach Timing Values(LS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に GoWAIT_CONNECT を設定する。(F/W)	0 (reference)
T1	接続検出機能をオンする。(H/W)	$T0 + 100\text{ms}\{T_{\text{SIGATT}}\} < T1$
T2	デバイスが接続される。	T2
T3	切断検出機能をオンする。(H/W)	$T2 + 2.5\text{us}\{T_{\text{DCNN}}\} < T3$
T4	接続検出ステータス(DetectCon)を発行する。(H/W) 切断検出機能と接続検出機能をオフする。(H/W)	$T3 + 100\text{ms}\{T_{\text{ATTDDB}}\} < T4$

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.3.10.3.3 GoDISABLED

ホストステート遷移実行(H_NegoControl0.AutoMode)に“GoDISABLED”を設定すると、ディセーブルド遷移に必要な処理を本 LSI のハードウェアが自動的に実行します。

“WAIT_CONNECT”ステートで接続を検出した時、“RESET”ステートで異常なデバイスからの Chirp を検出した時、“OPERATIONAL”ステートでポートエラーを検出した時は、本ステートに入ります。

1.3.10.3.3.1 HSデバイスが接続されている場合

HS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(6)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoDISABLED”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“DISABLED”に設定します(T0)。
- (3) 切断検出機能をオフします(T0)。
- (4) 現在実行中のトランザクション完了を待って、トランシーバ選択(H_XcvrControl.XcvrSelect)とターミナル選択(H_XcvrControl.TermSelect)とポートスピード(H_NegoControl_1.PortSpeed)を“FS”モードに設定し、オペレーションモード(H_XcvrControl.OpMode[1:0])を“PowerDown”に設定します(T1)。
- (5) 切断検出機能をオンします(T3)。
- (6) ファームウェアに対し、ディセーブルド遷移完了ステータス(H_SIE_IntStat_1.DisabledCmp)を発行します(T3)。

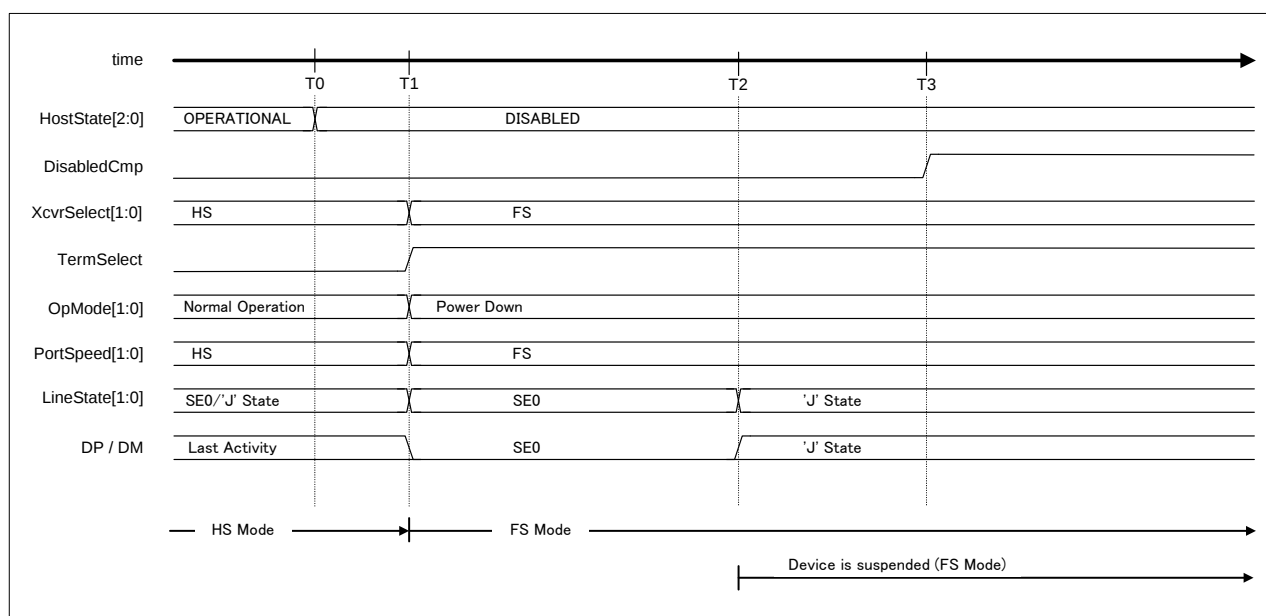


図 1.50 Disabled Timing (HS mode)

表 1.44 Disabled Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoDISABLED”を設定する。 (F/W) 切断検出機能をオフする。(H/W)	0 (reference)
T1	最後のバス・アクティビティ。完了後、トランシーバ選択とターミナル選択ポートスピードを“FS”モードに設定し、オペレーションモード(H_XcvrControl.OpMode[1:0])を“PowerDown”に設定する。(H/W)	T1
T2	デバイスがサスペンドを検出し、FS mode になる。	$T1 + 3.0\text{ms} < T2 \{T_{WTREV}\} < T1 + 3.125\text{ms}$
T3	切断検出機能をオンする。(H/W) ディセーブルド遷移完了ステータスを発行する。(H/W)	$T1 + 4\text{ms} < T3$

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.3.10.3.3.2 FSデバイスが接続されている場合

FS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(6)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoDISABLED”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“DISABLED”に設定します(T0)。
- (3) 切断検出機能をオフします(T0)。
- (4) 現在実行中のトランザクション完了を待って、オペレーションモード(H_XcvtControl.OpMode[1:0])を“PowerDown”に設定します(T1)。
- (5) 切断検出機能をオンします(T3)。
- (6) ファームウェアに対し、ディセーブルド遷移完了ステータス(H_SIE_IntStat_1.DisabledCmp)を発行します(T3)。

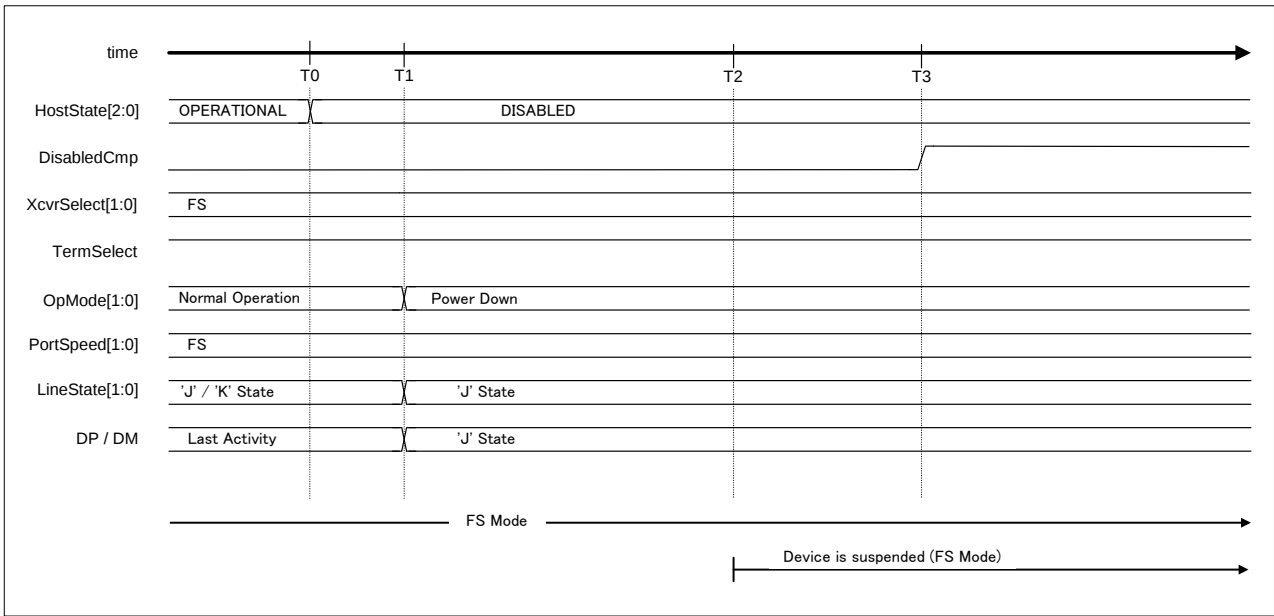


図 1.51 Disabled Timing (FS mode)

表 1.45 Disabled Timing Values (FS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に GoDisabled を設定する。(F/W) 切断検出機能をオフする。(H/W)	0 (reference)
T1	最後のバス・アクティビティ。完了後、オペレーションモード (H_XcvtControl.OpMode[1:0]) を “PowerDown” に設定する。(H/W)	T1
T2	デバイスがサスペンドを検出する。	$T1 + 3.0\text{ms} < T2 \{T_{WTREV}\} < T1 + 3.125\text{ms}$
T3	切断検出機能をオンする。 ディセーブルド遷移完了ステータスを発行する。(H/W)	$T1 + 4\text{ms} < T3$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.3.3 LSデバイスが接続されている場合

LS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(6)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoDISABLED”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“DISABLED”に設定します(T0)。
- (3) 切断検出機能をオフします(T0)。
- (4) 現在実行中のトランザクション完了を待って、オペレーションモード(H_XcvtControl.OpMode[1:0])を“PowerDown”に設定します(T1)。
- (5) 切断検出機能をオンします(T3)。
- (6) ファームウェアに対し、ディセーブルド遷移完了ステータス(H_SIE_IntStat_1.DisabledCmp)を発行します(T3)。

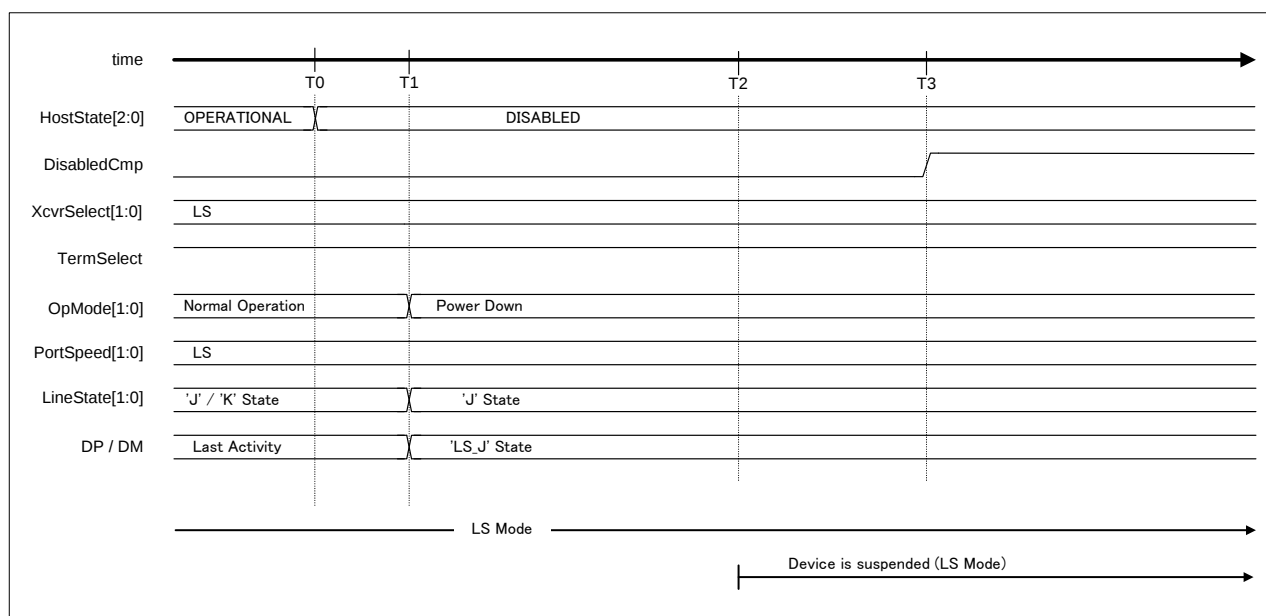


図 1.52 Disabled Timing (LS mode)

表 1.46 Disabled Timing Values (LS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoDISABLED”を設定する。(F/W) 切断検出機能をオフする。(H/W)	0 (reference)
T1	最後のバス・アクティビティ。完了後、オペレーションモード(H_XcvtControl.OpMode[1:0])を“PowerDown”に設定する。(H/W)	T1
T2	デバイスがサスペンドを検出する。	$T1 + 3.0\text{ms} < T2 \{T_{WTREV}\} < T1 + 3.125\text{ms}$
T3	切断検出機能をオンする。 ディセーブルド遷移完了ステータスを発行する。(H/W)	$T1 + 4\text{ms} < T3$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.4 GoRESET

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定すると、“RESET”遷移に必要な処理を本 LSI のハードウェアが自動的に実行します。なお、“OPERATIONAL”から本ステートに遷移する場合には、H/W が現在実行中のトランザクション完了を待って“RESET”処理を開始します。

1.3.10.3.4.1 HSデバイスに対するリセット

HS デバイスに対するリセットの実行手順は以下のとおりです。下記(2)～(14)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“RESET”に設定します(T0)。
- (3) トランシーバ選択(H_XcvtControl.XcvtSelect)とターミナル選択(H_XcvtControl.TermSelect)を“HS”モードに設定します(T0)。
- (4) オペレーションモード(H_XcvtControl.OpMode[1:0])を“Normal”に設定します(T0)。
- (5) デバイスチャープ検出機能をオンします(T0)。
- (6) ラインステート(H_USB_Status.LineState[1:0])にアクティビティ有(“J”ステートとして見える)が 2.5us 以上継続することでデバイスの Chirp を認識します。そしてデバイスの Chirp がリセット開始から規定時間以内に終了する(ラインステート(H_USB_Status.LineState[1:0])がアクティビティ無(“SE0”として見える)となる)ことをもってデバイスチャープ正常検出ステータス(H_SIE_IntStat_0.DetectDevChirpOK)を発行します(T2)。
ハードウェアはデバイスチャープ正常検出ステータス(H_SIE_IntStat_0.DetectDevChirpOK)が“1”にセットされている場合にデバイスから正常な Chirp 信号が送信されたと判断し以降の処理を行います。その為、HS デバイスが切断された時には必ずデバイスチャープ正常検出ステータス(H_SIE_IntStat_0.DetectDevChirpOK)を“0”にクリアしてください。クリアを行わずデバイスチャープ正常検出ステータス(H_SIE_IntStat_0.DetectDevChirpOK)が“1”にセットされた状態で FS デバイスが接続されると、そのデバイスを HS デバイスと誤認識して以降の処理を行ってしまいます。
- (7) デバイスチャープ検出機能をオフします(T2)。
- (8) デバイスの Chirp 終了後、ホストは“Chirp K”を出力し始めます(T3)。
- (9) ホストは“Chirp K”から“Chirp J”に切り替えて出力します(T4)。
- (10) ホストは“Chirp J”から“Chirp K”に切り替えて出力します(T5)。以降ホストは“Chirp K”と“Chirp J”のシーケンスを交互に出力します。
- (11) デバイスはホストの Chirp を検出すると HS モードに移行します(T6)。T7 の時点から Chirp の高さが変わっているのは、デバイス側の HS ターミネーションが有効になったことを表しています。通常、デバイスが FS モード時の Chirp は約 800mV、HS モード時の Chirp では、約 400mV となります。
- (12) ホストは Chirp を終了します(T8)。
- (13) リセットを終了します(T9)。
- (14) リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T9)。

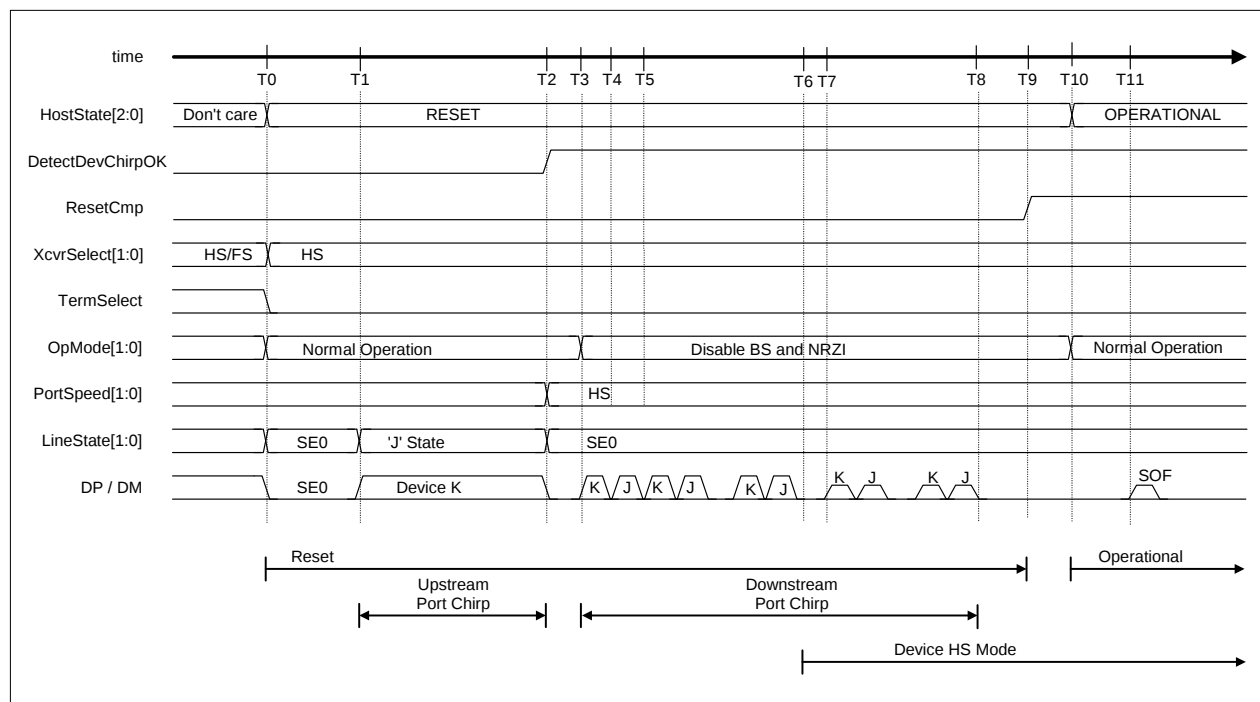


図 1.53 Reset Timing (HS mode)

表 1.47 Reset Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESET”を設定する。(F/W) デバイスチャープ検出機能をオンする。(H/W)	0 (reference)
T1	デバイスが Chirp を開始。	$T0 < T1 < T0 + 6.0\text{ms}$
T2	デバイスが Chirp を終了。 ポートスピードを HS に設定する。 デバイスチャープ検出機能をオフする。 デバイスチャープ正常検出ステータス(DetectDevChirpOK)を発行する。(H/W)	$T1 + 1.0\text{ms} \{T_{UCH}\} < T2 < T0 + 7.0\text{ms} \{T_{UCHEND}\}$
T3	ホストが最初の Chirp(“Chirp K”)を出力する。(H/W)	$T2 < T3 < T2 + 100\mu\text{s} \{T_{WTDCH}\}$
T4	ホストが“Chirp K”から“Chirp J”に切り替えて出力する。(H/W)	$T3 + 40\mu\text{s} \{T_{DCHBIT}\} < T4 < T3 + 60\mu\text{s} \{T_{DCHBIT}\}$
T5	ホストが“Chirp J”から“Chirp K”に切り替えて出力する。(H/W)	$T4 + 40\mu\text{s} \{T_{DCHBIT}\} < T5 < T4 + 60\mu\text{s} \{T_{DCHBIT}\}$
T6	デバイスがホストの Chirp を検出する。	T6
T7	デバイスが HS モードに遷移。	$T6 < T7 < T6 + 500\mu\text{s}$
T8	ホストが Chirp を終了。(H/W)	$T3 + 50\text{ms} \{T_{DRSTR}\} < T8$
T9	リセットの終了。リセット終了ステータス(ResetCmp)を発行する。(H/W)	$T8 < T9 < T8 + 150\mu\text{s}$
T10(参考)	H_NegoControl_0.AutoMode に“GoOPERATIONAL”を設定する。(F/W)	$T10 < T9 + 200\mu\text{s}$
T11(参考)	最初の SOF が送出。(H/W)	$T10 + 120\mu\text{s} < T11 < T10 + 130\mu\text{s}$ $T8 + 100\mu\text{s} \{T_{DCHSE0}\} < T11 < T8 + 500\mu\text{s} \{T_{DCHSE0}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.4.2 異常なデバイスのChirp検出

HS Detection Handshake において、デバイスの Chirp が異常だった場合の動作を示します。チャープ完了ディセーブル(H_NegoControl_1.DisChirpFinish)の設定により 2 つの動作モードが選択できます。

1.3.10.3.4.2.1 チャープ完了ディセーブル(H_NegoControl_1.DisChirpFinish)設定が 0 の場合

異常検出後にホストの Chirp を行いません。デバイスチャープ異常検出ステータスが発行された場合、ファームウェアはリセット完了ステータス(H_SIE_IntStat_1.ResetCmp)の発行を待って、ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoDISABLED”を設定し、ホストステートを“DISABLED”ステートに遷移させます。下記(2)～(9)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を"RESET"に設定します(T0)。
- (3) トランシーバ選択(H_XcvrControl.XcvrSelect)とターミナル選択(H_XcvrControl.TermSelect)を HS モードに設定します(T0)。
- (4) オペレーションモード(H_XcvrControl.OpMode[1:0])を“Normal”に設定します(T0)。
- (5) デバイスチャープ検出機能をオンします(T0)。
- (6) ラインステート(H_USB_Status.LineState[1:0])にアクティビティ有(“J”ステートとして見える)が 2.5us 以上継続することでデバイスの Chirp を認識します。しかし、リセット開始から規定時間以内にデバイスの Chirp が終了しないため異常とみなし、デバイスチャープ異常検出ステータス(H_SIE_IntStat_0.DetectDevChirpNG)を発行します(T2)。
- (7) デバイスチャープ検出機能をオフします(T2)。
- (8) リセットを終了します(T3)。
- (9) リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T3)。

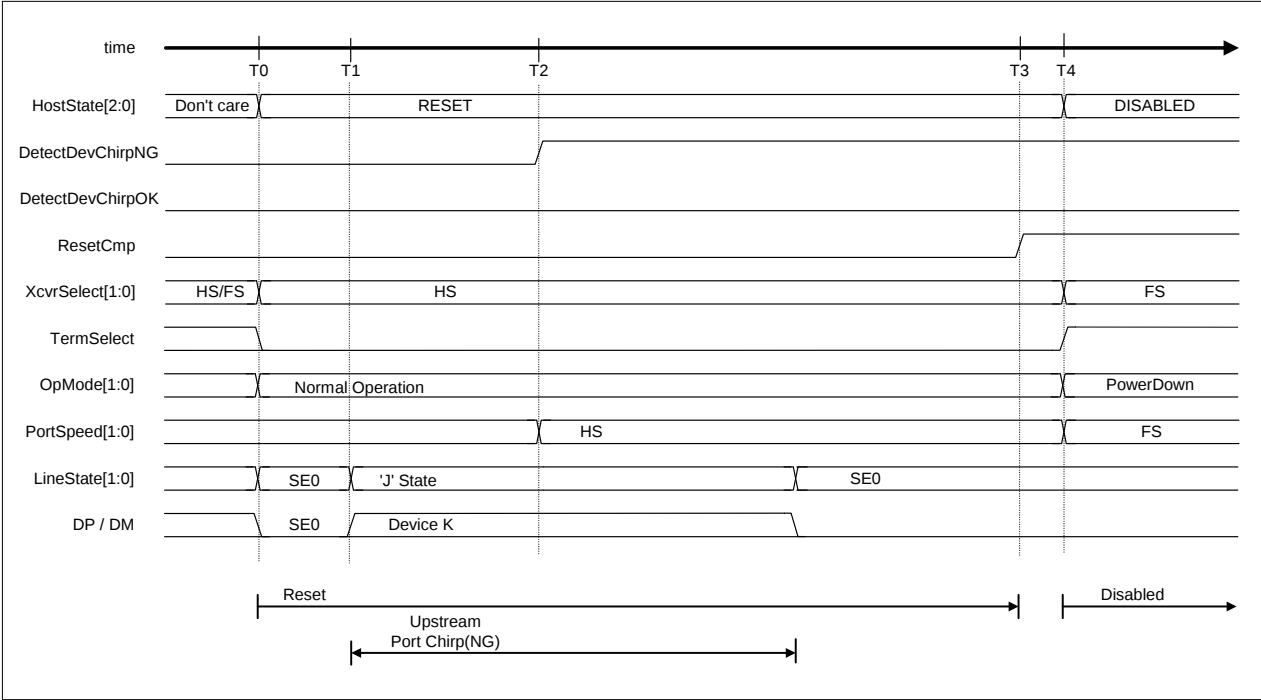


図 1.54 Detect Device Chirp NG Timing(チャープ完了ディセーブル設定 0)

表 1.48 Detect Device Chirp Timing Values(チャープ完了ディセーブル設定 0)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESET”を設定する。(F/W) デバイスチャープ検出機能をオンする。(H/W)	0 (reference)
T1	デバイスが Chirp を開始。	$T0 < T1 < T0 + 6.0\text{ms}$
T2	デバイスチャープ異常検出ステータス(DetectDevChirpNG)を発行する。デバイスチャープ検出機能をオフする。(H/W)	$T0 + 7\text{ms}\{T_{UCHEND}\} < T2$
T3	リセットの終了。リセット終了ステータス(ResetCmp)を発行する。(H/W)	$T2 + 50\text{ms}\{T_{DRSTR}\} < T3$
T4(参考)	H_NegoControl_0.AutoMode に“GoDISABLED”を設定する。(F/W)	T4

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.4.2.2

チャープ完了ディセーブル(H_NegoControl_1.DisChirpFinish)設定が 1 の場合

異常検出後にデバイスの Chirp 完了を待ってホストの Chirp を行います。

本モード使用時において、リセット完了ステータス

(H_SIE_IntStat_1.ResetCmp) を発行を待たずにホストステートを “DISABLED” ステートに遷移させる場合には、H_NegoControl_0 レジスタに 0x80(H_NegoControl_0.AutoModeCancel に “1”、かつ H_NegoControl_0.AutoMode に “0x0”) をライトして実行中ステートの動作を停止させます。H_NegoControl_0.AutoModeCancel ビットは停止処理が完了することで “0” になります (60MHz クロックで 6 サイクル程度必要) ので、H_NegoControl_0.AutoModeCancel ビットが “0” になるのを確認してから、同レジスタに 0x03 をライト (すなわちホストステート遷移実行 (H_NegoControl_0.AutoMode) に “GoDISABLED” を設定) してください。

下記(2)～(15)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行 (H_NegoControl_0.AutoMode) に “GoRESET” を設定します (T0)。
- (2) ホストステートモニタ (H_NegoControl_0.HostState) を “RESET” に設定します (T0)。
- (3) トランシーバ選択 (H_XcvtControl.XcvtSelect) とターミナル選択 (H_XcvtControl.TermSelect) を HS モードに設定します (T0)。
- (4) オペレーションモード (H_XcvtControl.OpMode[1:0]) を “Normal” に設定します (T0)。
- (5) デバイスチャープ検出機能をオンします (T0)。
- (6) ラインステート (USB_Host_Status.LineState[1:0]) にアクティビティ有 (“J” ステートとして見える) が 2.5us 以上継続することでデバイスの Chirp を認識します。しかし、リセット開始から規定時間以内にデバイスの Chirp が終了しないため異常とみなし、デバイスチャープ異常検出ステータス (H_SIE_IntStat_0.DetectDevChirpNG) を発行します (T2)。
- (7) デバイスチャープ検出機能をオフします (T2)。
- (8) デバイスの Chirp が終了する (ラインステート (H_USB_Status.LineState[1:0]) がアクティビティ無 (“SE0” として見える) になる) ことをもってデバイスチャープ正常検出ステータス (H_SIE_IntStat_0.DetectDevChirpOK) を発行します (T3)。
- (9) デバイスの Chirp 終了後、ホストは “Chirp K” を出力し始めます (T4)。
- (10) ホストは “Chirp K” から “Chirp J” に切り替えて出力します (T5)。
- (11) ホストは “Chirp J” から “Chirp K” に切り替えて出力します (T6)。以降ホストは “Chirp K” と “Chirp J” のシーケンスを交互に出力します。
- (12) デバイスはホストの Chirp を検出すると HS モードに移行します (T7)。T8 の時点から Chirp の高さが変わっているのは、デバイス側の HS ターミネーションが有効になったことを表しています。通常、デバイスが FS モード時の Chirp は約 800mV、HS モード時の Chirp では、約 400mV となります。
- (13) ホストは Chirp を終了します (T9)。
- (14) リセットを終了します (T10)。

(15) リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T10)。

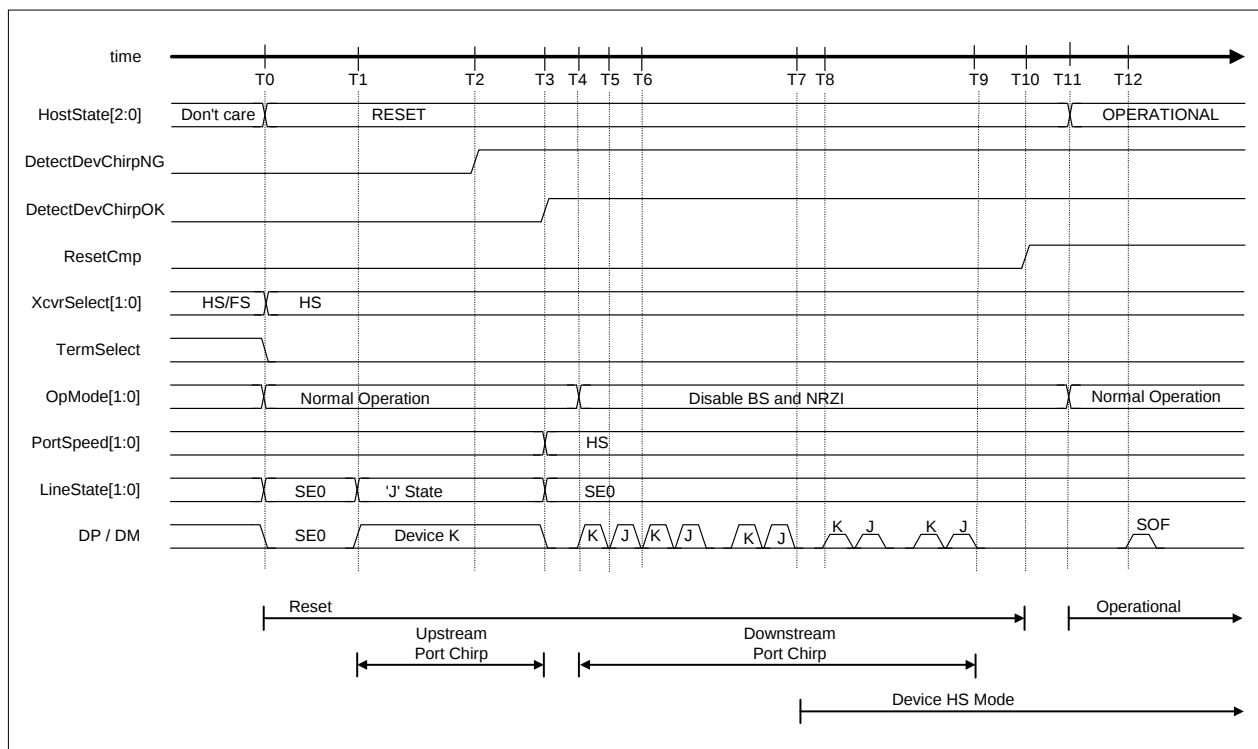


図 1.55 Detect Device Chirp NG Timing(チャープ完了ディセーブル設定 1)

1. 機能説明

表 1.48 Detect Device Chirp Timing Values(チャープ完了ディセーブル設定 1)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESET”を設定する。(F/W) デバイスチャープ検出機能をオンする。(H/W)	0 (reference)
T1	デバイスが Chirp を開始。	$T0 < T1 < T0 + 6.0\text{ms}$
T2	デバイスチャープ異常検出ステータス(DetectDevChirpNG)を発行する。 デバイスチャープ検出機能をオフする。(H/W)	$T0 + 7\text{ms}\{T_{UCHEND}\} < T2$
T3	デバイスが Chirp を終了。 ポートスピードを HS に設定する。 デバイスチャープ正常検出ステータス(DetectDevChirpOK)を発行する。(H/W)	T3
T4	ホストが最初の Chirp(“Chirp K”)を出力する。(H/W)	$T3 < T4 < T3 + 100\text{us}\{T_{WTDCH}\}$
T5	ホストが“Chirp K”から“Chirp J”に切り替えて出力する。(H/W)	$T4 + 40\text{us}\{T_{DCHBIT}\} < T45 < T4 + 60\text{us}\{T_{DCHBIT}\}$
T6	ホストが“Chirp J”から“Chirp K”に切り替えて出力する。(H/W)	$T5 + 40\text{us}\{T_{DCHBIT}\} < T6 < T5 + 60\text{us}\{T_{DCHBIT}\}$
T7	デバイスがホストの Chirp 検出する。	T7
T8	デバイスが HS モードに遷移。	$T7 < T8 < T6 + 500\text{us}$
T9	ホストが Chirp を終了。(H/W)	$T4 + 50\text{ms}\{T_{DRSTR}\} < T9$
T10	リセットの終了。リセット終了ステータス(ResetCmp)を発行する。(H/W)	$T9 < T10 < T9 + 150\text{us}$
T11(参考)	H_NegoControl0.AutoMode に“GoOPERATIONAL”を設定する。(F/W)	$T11 < T10 + 200\text{us}$
T12(参考)	最初の SOF が送出。(H/W)	$T11 + 120\text{us} < T12 < T11 + 130\text{us}$ $T9 + 100\text{us}\{T_{DCHSE0}\} < T12 < T9 + 500\text{us}\{T_{DCHSE0}\}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.4.3 FSデバイスに対するリセット

FS デバイスに対するリセットの実行手順は以下のとおりです。下記(2)～(9)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“RESET”に設定します(T0)。
- (3) トランシーバ選択(H_XcvrControl.XcvrSelect)とターミナル選択(H_XcvrControl.TermSelect)を HS モードに設定します(T0)。
- (4) オペレーションモード(H_XcvrControl.OpMode[1:0])を“Normal”に設定します(T0)。
- (5) デバイスチャープ検出機能をオンします(T0)。
- (6) デバイスの Chirp を検出せず、ポートスピード(H_NegoControl_1.PortSpeed[1:0])が HS/FS の為、相手デバイスが FS デバイスであると判断し、トランシーバ選択(H_XcvrControl.XcvrSelect)とポートスピード(H_NegoControl_1.PortSpeed[1:0])を“FS”に設定します(T1)。
- (7) デバイスチャープ検出機能をオフします(T1)。
- (8) ターミナル選択(H_XcvrControl1.TermSelect)を“FS”に設定します(T2)。
- (9) リセット終了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T3)。

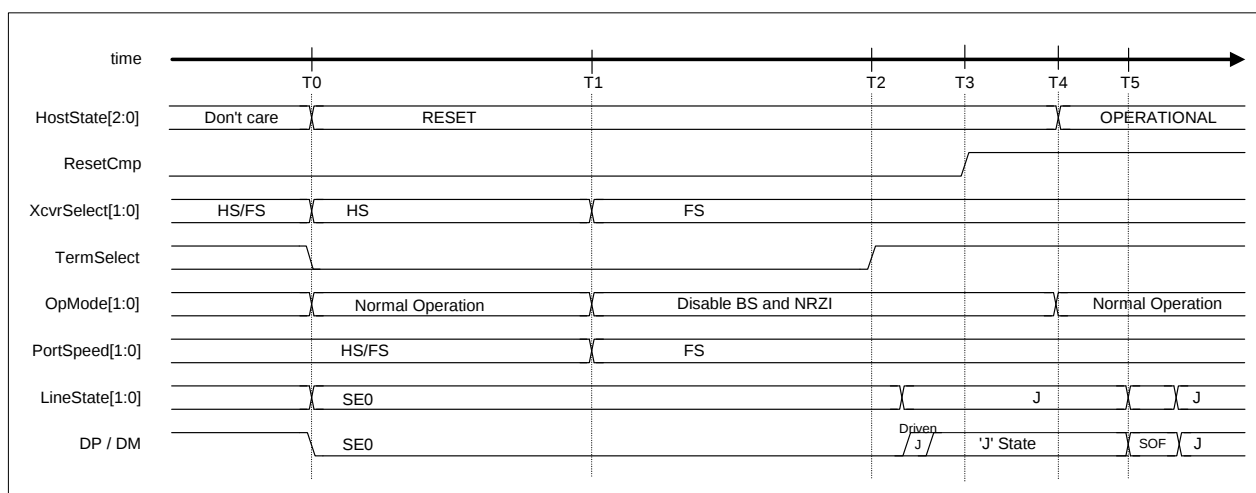


図 1.56 Reset Timing (FS mode)

1. 機能説明

表 1.50 Reset Timing Values (FS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESET”を設定する。(F/W) デバイスチャープ検出機能をオンする。(H/W)	0 (reference)
T1	トランシーバ選択を“FS”に設定する。 ポートスピードを“FS”に設定する。 デバイスチャープ検出機能をオフする。(H/W)	$T0 + 7.0\text{ms}\{T_{\text{UCHEND}}\} < T1$
T2	ターミナル選択を“FS”に設定する。(H/W)	$T0 + 50\text{ms}\{T_{\text{DRSTR}}\} < T2$
T3	リセット完了ステータスを発行する。(H/W)	$T2 + 150\text{us} < T3$
T4(参考)	H_NegoControl_0.AutoMode に“GoOPERATIONAL”を設定する。 (F/W)	T4
T5(参考)	最初の SOF が送出。(H/W)	$T4 + 0.9\text{ms} < T5 < T4 + 1.1\text{ms}$ ($T5 < T2 + 3\text{ms}$)

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.4.4 LSデバイスに対するリセット

LS デバイスに対するリセットの実行手順は以下のとおりです。下記(2)～(7)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESET”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“RESET”に設定します(T0)。
- (3) トランシーバ選択(H_XcvrControl.XcvrSelect)とターミナル選択(H_XcvrControl.TermSelect)をHSモードに設定します(T0)。
- (4) オペレーションモード(H_XcvrControl.OpMode[1:0])を“Normal”に設定します(T0)。
- (5) ポートスピード(H_NegoControl_1.PortSpeed[1:0])が“LS”の為、トランシーバ選択(H_XcvrControl.XcvrSelect)を“LS”に設定します(T1)。
- (6) ターミナル選択(H_XcvrControl.TermSelect)を“FS”に設定します(T2)。
- (7) リセット終了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T3)。

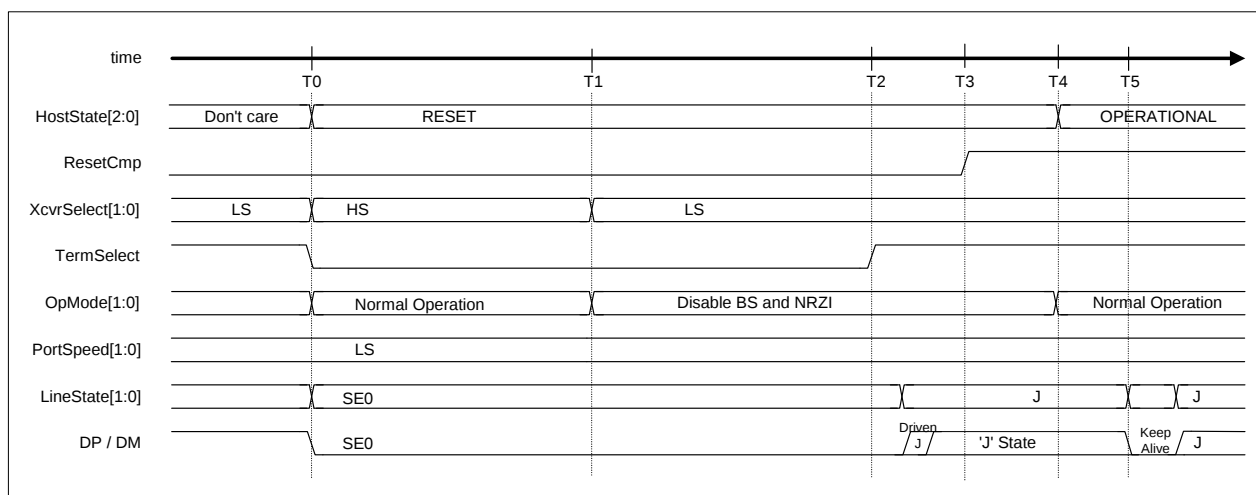


図 1.57 Reset Timing (LS mode)

表 1.51 Reset Timing Values (LS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESET”を設定する。(F/W)	0 (reference)
T1	トランシーバ選択を“LS”に設定する。(H/W)	$T0 + 7.0ms\{T_{UCHEND}\} < T1$
T2	ターミナル選択を“FS”に設定する。(H/W)	$T0 + 50ms\{T_{DRSTR}\} < T2$
T3	リセット完了ステータスを発行する。(H/W)	$T2 + 150\mu s < T3$
T4(参考)	H_NegoControl_0.AutoMode に“GoOPERATIONAL”を設定する。(F/W)	T4
T5(参考)	最初の KeepAlive が送出。(H/W)	$T4 + 0.9ms < T5 < T4 + 1.1ms$ ($T5 < T2 + 3ms$)

注： {} は、USB2.0 規格書で規格されている名称である。

1. 機能説明

1.3.10.3.5 GoOPERATIONAL

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoOPERATIONAL”を設定すると、“OPERATIONAL”遷移に必要な処理を本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoOPERATIONAL”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“OPERATIONAL”に設定します(T0)。
- (3) オペレーションモード(H_XcvtControl.OpMode[1:0])を“Normal”に設定し、USB のトランザクションを実行する状態になります(T0)。
- (4) 切断検出機能をオンします(T0)。
- (5) ポートスピード(H_NegoControl_1.PortSpeed[1:0])が“HS”または“FS”に設定されている場合は最初の SOF を発行します(T1)。“LS”に設定されている場合は最初の KeepAlive が発行します(T1)。以降はチャンネルの設定に従って転送が行われます。

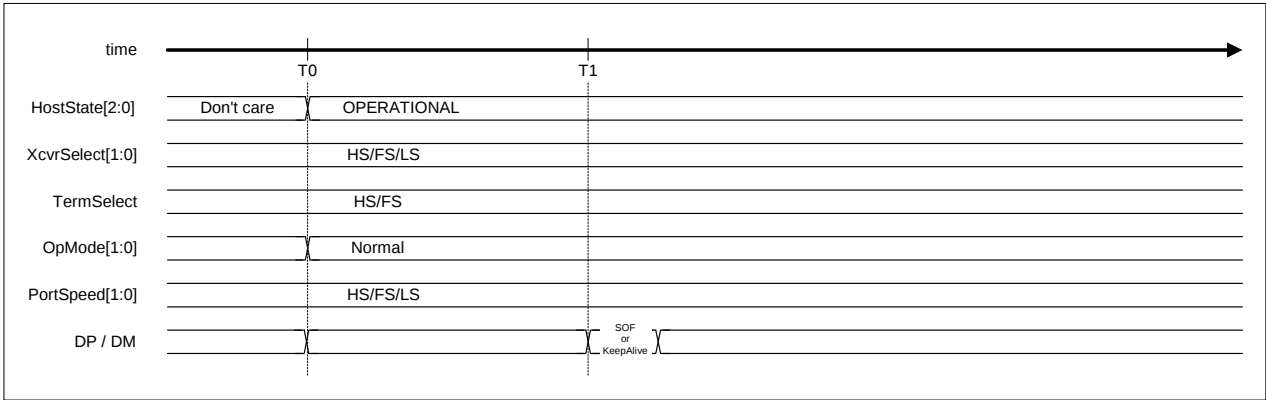


図 1.58 GoOPERATIONAL Timing

表 1.52 GoOPERATIONAL Timing Values

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoOPERATIONAL”を設定する。(F/W)	0 (reference)
T1	最初の SOF(HS／FS)、または KeepAlive(LS)が送出。	$T0+120\mu s < T1(HS) < T0 + 130\mu s$ $T0+0.9ms < T1(FS,LS) < T0 + 1.1ms$

1.3.10.3.6 GoSUSPEND

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoSUSPEND”を設定すると、サスペンド遷移に必要な処理を本 LSI のハードウェアが自動的に実行します。

1.3.10.3.6.1 HSデバイスが接続されている場合

HS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(7)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoSUSPEND”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“SUSPEND”に設定します(T0)。
- (3) 切断検出機能およびリモート・ウェークアップ検出機能をオフします(T0)。
- (4) 現在実行中のトランザクション完了を待って、トランシーバ選択(H_XcvrControl.XcvrSelect)とターミナル選択(H_XcvrControl.TermSelect)を FS モードに設定し、オペレーションモード(H_XcvrControl.OpMode[1:0])を“PowerDown”に設定します(T1)。
- (5) 切断検出機能をオンします(T3)。
- (6) リモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)がイネーブルの場合、リモート・ウェークアップ検出機能をオンします(T3)。
- (7) ファームウェアに対し、サスペンド遷移完了ステータス(H_SIE_IntStat_1.SuspendCmp)を発行します(T3)。

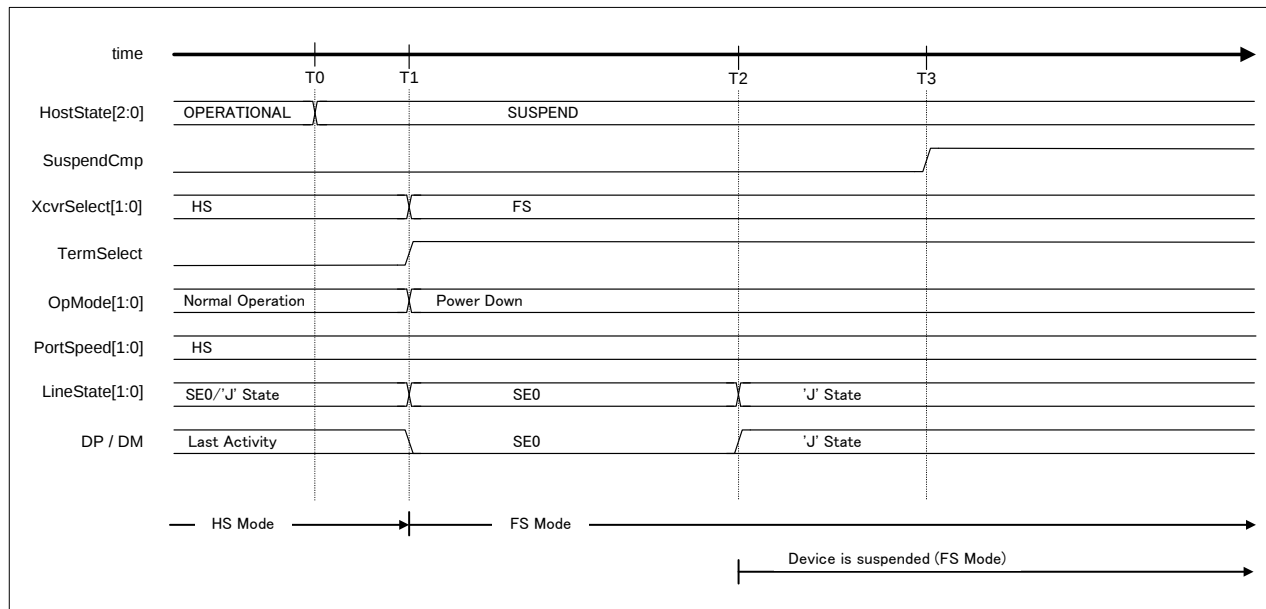


図 1.59 Suspend Timing (HS mode)

1. 機能説明

表 1.53 Suspend Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoSUSPEND”を設定する。 (F/W) 切断検出機能およびリモート・ウェークアップ検出機能をオフする。(H/W)	0 (reference)
T1	最後のバス・アクティビティ。完了後、トランシーバ選択とターミナル選択を FS モードに設定し、オペレーションモード (H_XcvtControl.OpMode[1:0]) を“PowerDown”に設定する。 (H/W)	T1
T2	デバイスがサスペンドを検出し、FS mode になる。	$T1 + 3.0\text{ms} < T2 \{T_{WTREV}\} < T1 + 3.125\text{ms}$
T3	切断検出機能をオンする。 リモート・ウェークアップ受付許可がイネーブルの場合、リモート・ウェークアップ検出機能をオンする。 サスペンド遷移完了ステータスを発行する。(H/W)	$T1 + 5\text{ms}\{T_{WTRSM}\} < T3$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.6.2 FSデバイスが接続されている場合

FS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(7)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoSUSPEND”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“SUSPEND”に設定します(T0)。
- (3) 切断検出機能およびリモート・ウェークアップ検出機能をオフします(T0)。
- (4) 現在実行中のトランザクション完了を待って、オペレーションモード(H_XcvtControl.OpMode[1:0])を“PowerDown”に設定します(T1)。
- (5) 切断検出機能をオンします(T3)。
- (6) リモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)がイネーブルの場合、リモート・ウェークアップ検出機能をオンします(T3)。
- (7) ファームウェアに対し、サスペンド遷移完了ステータス(H_SIE_IntStat_1.SuspendCmp)を発行します(T3)。

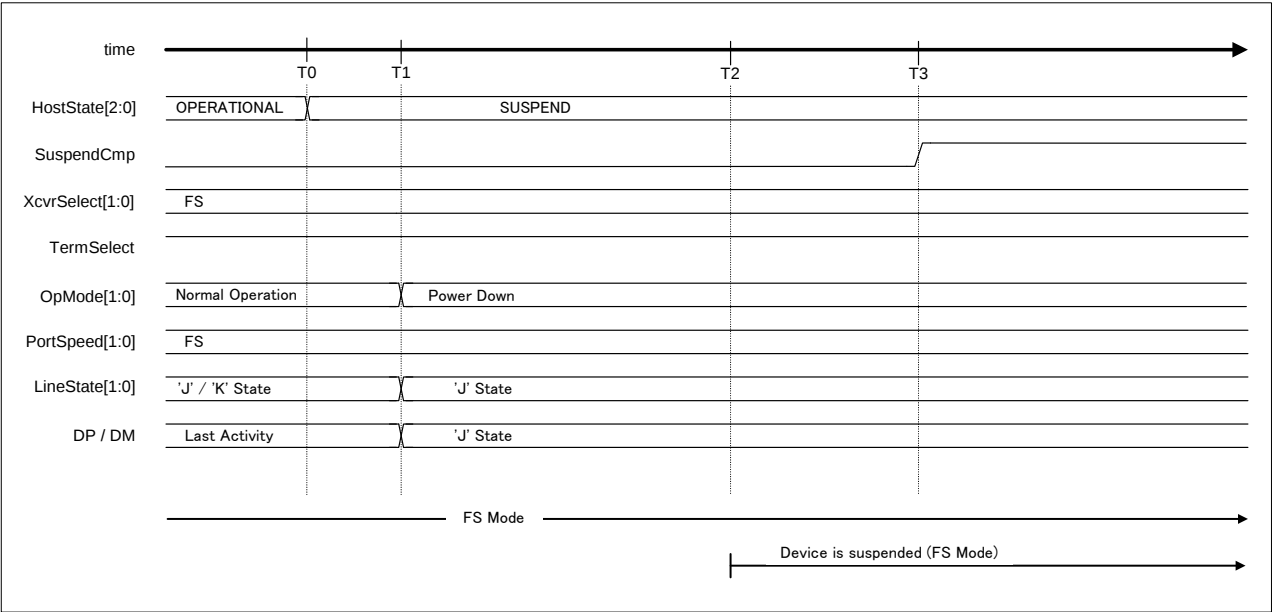


図 1.60 Suspend Timing (FS mode)

1. 機能説明

表 1.54 Suspend Timing Values (FS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoSUSPEND”を設定する。 (F/W) 切断検出機能およびリモート・ウェークアップ検出機能をオフする。(H/W)	0 (reference)
T1	最後のバス・アクティビティ。完了後、オペレーションモード (H_XcvtControl.OpMode[1:0]) を“PowerDown”に設定する。 (H/W)	T1
T2	デバイスがサスペンドを検出する。	$T1 + 3.0\text{ms} < T2 \{T_{WTREV}\} < T1 + 3.125\text{ms}$
T3	切断検出機能をオンする。 リモート・ウェークアップ受付許可がイネーブルの場合、リモート・ウェークアップ検出機能をオンする。 サスペンド遷移完了ステータスを発行する。(H/W)	$T1 + 5\text{ms}\{T_{WTRSM}\} < T3$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.6.3 LSデバイスが接続されている場合

LS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(7)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoSUSPEND”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl0.HostState)を“SUSPEND”に設定します(T0)。
- (3) 切断検出機能およびリモート・ウェークアップ検出機能をオフします(T0)。
- (4) 現在実行中のトランザクション完了を待って、オペレーションモード(H_XcvtControl.OpMode[1:0])を“PowerDown”に設定します(T1)。
- (5) 切断検出機能をオンします(T3)。
- (6) リモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)がイネーブルの場合、リモート・ウェークアップ検出機能をオンします(T3)。
- (7) ファームウェアに対し、サスペンド遷移完了ステータス(H_SIE_IntStat_1.SuspendCmp)を発行します(T3)。

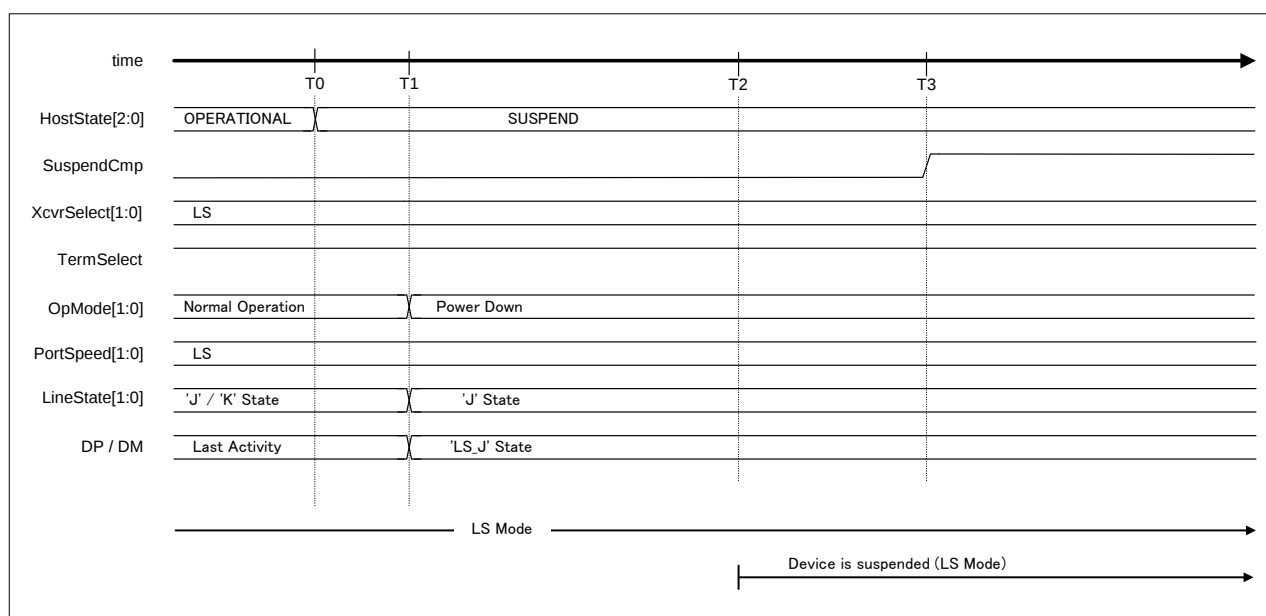


図 1.61 Suspend Timing (LS mode)

1. 機能説明

表 1.55 Suspend Timing Values (LS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoSUSPEND”を設定する。 (F/W) 切断検出機能およびリモート・ウェークアップ検出機能をオフする。(H/W)	0 (reference)
T1	最後のバス・アクティビティ。完了後、オペレーションモード (H_XcvtControl.OpMode[1:0]) を“PowerDown”に設定する。 (H/W)	T1
T2	デバイスがサスペンドを検出する。	$T1 + 3.0\text{ms} < T2 \{T_{WTREV}\} < T1 + 3.125\text{ms}$
T3	切断検出機能をオンする。 リモート・ウェークアップ受付許可がイネーブルの場合、リモート・ウェークアップ検出機能をオンする。 サスペンド遷移完了ステータスを発行する。(H/W)	$T1 + 5\text{ms}\{T_{WTRSM}\} < T3$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.7 GoRESUME

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUME”を設定すると、レジューム遷移に必要な処理を本 LSI のハードウェアが自動的に実行します。

1.3.10.3.7.1 HSデバイスが接続されている場合

HS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(8)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に "GoRESUME"を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“RESUME”に設定します(T0)。
- (3) 切断検出機能およびリモート・ウェークアップ検出機能をオフします(T0)。
- (4) オペレーションモード(H_XcvrControl.OpMode)を“Disable BS and NRZI”に設定し、レジューム“K”信号発行を開始します(T0)。
- (5) レジューム“K”信号発行を完了します(T1)。
- (6) ターミナル選択(H_XcvrControl.TermSelect)を“HS2”に設定します(T2)。
- (7) トランシーバ選択(H_XcvrControl.XcvrSelect)を“HS”に設定します(T3)。
- (8) ファームウェアに対し、レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行します(T3)。

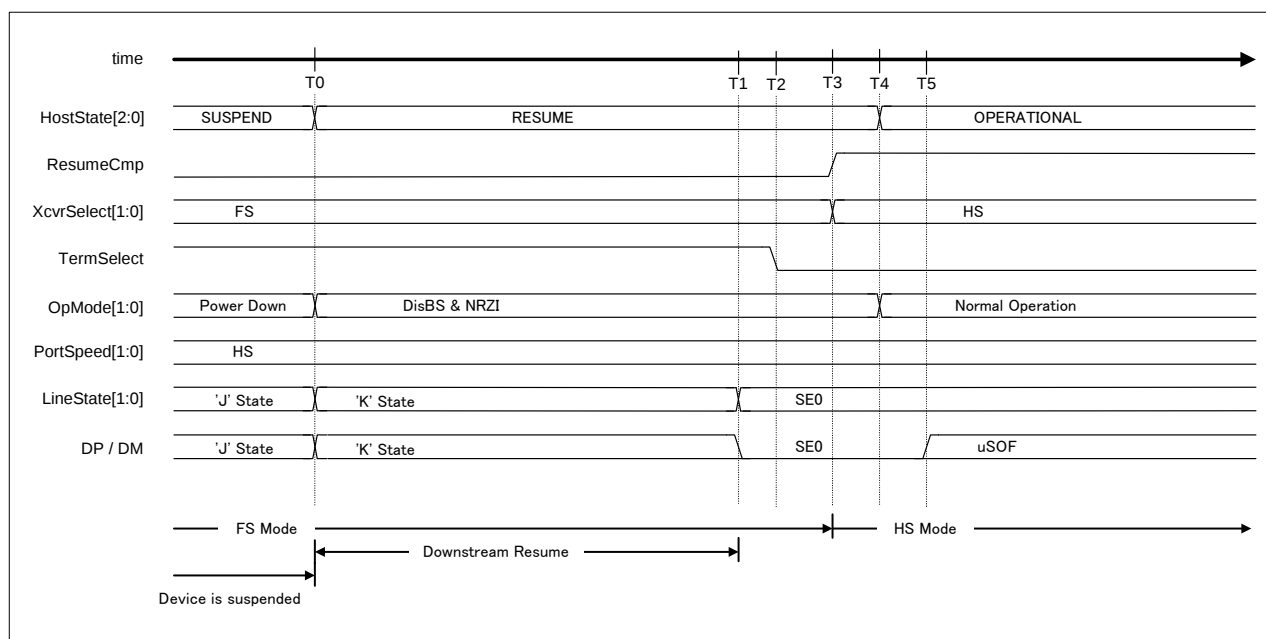


図 1.62 Resume Timing (HS mode)

1. 機能説明

表 1.56 Resume Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESUME”を設定する。(F/W) 切断検出機能およびリモート・ウェークアップ検出機能をオフする。 オペレーションモードを“Disable BS and NRZI”に設定し、レジューム“K”信号の発行を開始する。(H/W)	0 (reference)
T1	レジューム“K”信号発行を完了する。 ターミナル選択を“HS”に設定する。(H/W)	$T0 + 20\text{ms}\{T_{\text{DRSMDN}}\} < T1$
T2	トランシーバ選択を“HS”に設定する。(H/W)	$T1 + 100\text{ns} < T2 < T1 + 2.0\mu\text{s}$
T3	レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行する。(H/W)	$T1 + 90\mu\text{s} < T3 < T1 + 110\mu\text{s}$
T4(参考)	GoOPERATIONAL を設定する。(F/W) オペレーションモードを“NomalOperation”に設定する。(H/W)	T4
T5(参考)	最初のマイクロ SOF を発行する。(H/W)	$T5 < T1 + 3\text{ms}$ $T4 + 120\mu\text{s} < T5 < T4 + 130\mu\text{s}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.7.2 FSデバイスが接続されている場合

FS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(6)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUME”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“RESUME”に設定します(T0)。
- (3) 切断検出機能およびリモート・ウェイクアップ検出機能をオフします(T0)。
- (4) オペレーションモード(H_XcvrControl.OpMode)を“Disable BS and NRZI”に設定し、レジューム“K”信号の発行を開始します(T0)。
- (5) レジューム“K”信号の発行を完了し(T1)、最後に LS bit time の EOP を付与します。
- (6) ファームウェアに対し、レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行します(T2)。

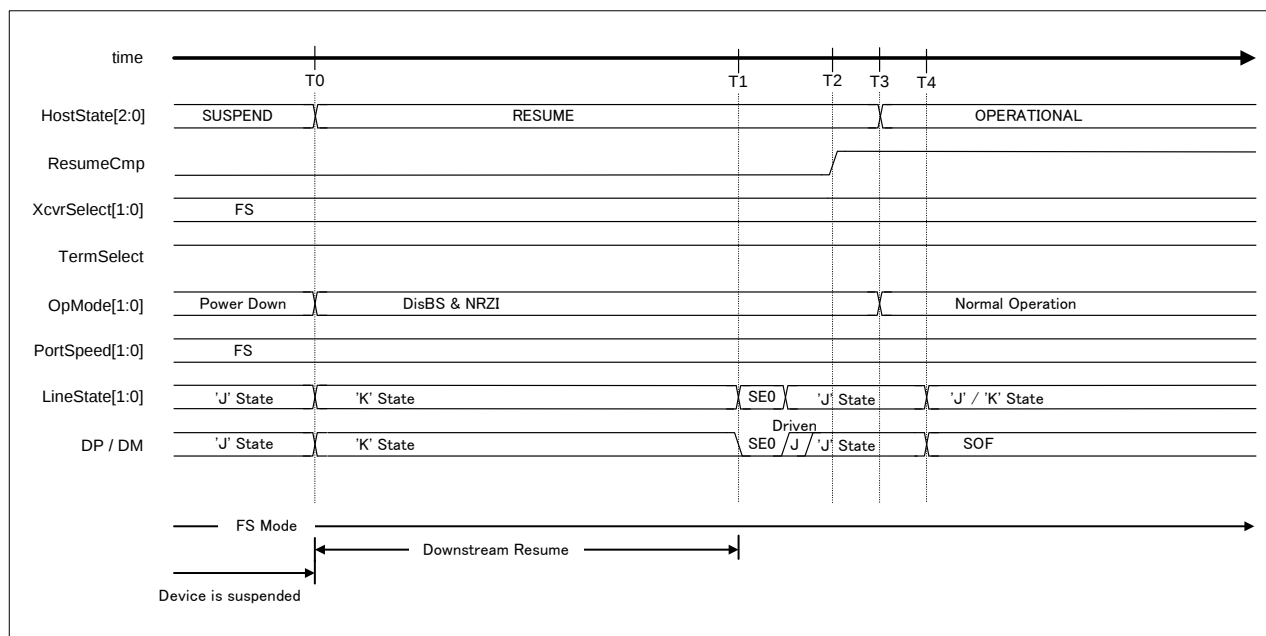


図 1.63 Resume Timing FS mode)

1. 機能説明

表 1.57 Resume Timing Values (FS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESUME”を設定する。(F/W) 切断検出機能およびリモート・ウェークアップ検出機能をオフする。 オペレーションモードを“Disable BS and NRZI”に設定し、レジューム“K”信号の発行を開始する。(H/W)	0 (reference)
T1	レジューム“K”信号の発行を完了し、最後に LS bit time の EOP を付与する。(H/W)	$T0 + 20\text{ms}\{T_{\text{DRSMDN}}\} < T1$
T2	レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行する。(H/W)	$T1 + 90\text{us} < T2 < T1 + 110\text{us}$
T3(参考)	GoOPERATIONAL を設定する。(F/W) オペレーションモードを“NomalOperation”に設定する。(H/W)	T3
T4(参考)	最初の SOF を発行する。(H/W)	$T4 < T1 + 3\text{ms}$ $T3 + 0.9\text{ms} < T4 < T3 + 1.1\text{ms}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.7.3 LSデバイスが接続されている場合

LS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(6)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUME”を設定します(T0)。
- (2) ホストステートモニタ(H_NegoControl_0.HostState)を“RESUME”に設定します(T0)。
- (3) 切断検出機能およびリモート・ウェイクアップ検出機能をオフします(T0)。
- (4) オペレーションモード(H_XcvrControl.OpMode)を“Disable BS and NRZI”に設定し、レジューム“K”信号の発行を開始します(T0)。
- (5) レジューム“K”信号の発行を完了し(T1)、最後に LS bit time の EOP を付与します。
- (6) ファームウェアに対し、レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行します(T2)。

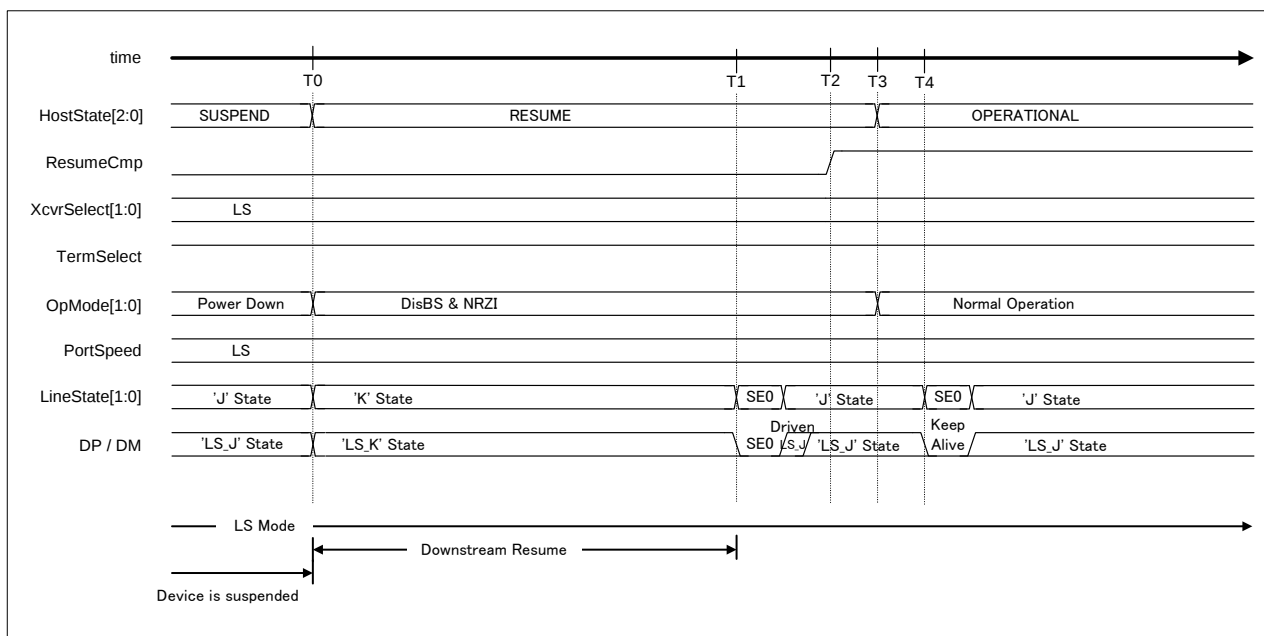


図 1.64 Resume Timing (LS mode)

1. 機能説明

表 1.58 Resume Timing Values (LS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESUME”を設定する。(F/W) 切断検出機能およびリモート・ウェークアップ検出機能をオフする。 オペレーションモードを“Disable BS and NRZI”に設定し、レジューム“K”信号の発行を開始する。(H/W)	0 (reference)
T1	レジューム“K”信号の発行を完了し、最後に LS bit time の EOP を付与する。(H/W)	$T0 + 20\text{ms}\{T_{\text{DRSMDN}}\} < T1$
T2	レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行する。(H/W)	$T1 + 90\text{us} < T2 < T1 + 110\text{us}$
T3(参考)	“GoOPERATIONAL”を設定する。(F/W) オペレーションモードを“NomalOperation”に設定する。(H/W)	T3
T4(参考)	最初の Keep Alive を発行する。(H/W)	$T4 < T1 + 3\text{ms}$ $T3 + 0.9\text{ms} < T4 < T3 + 1.1\text{ms}$

注： {} は、USB2.0 規格書で規格されている名称である。

1.3.10.3.8 GoWAIT_CONNECTtoDIS

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECTtoDIS”を設定すると、“WAIT_CONNECT”ステートから“DISABLED”ステートに至る必要な処理を本 LSI のハードウェアが自動的に実行します。

実行手順は以下のとおりです。下記(2)～(5)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECTtoDIS”を設定します(T0)。
- (2) “GoWAIT_CONNECT”と同等の処理を実行します(T0)。
- (3) 接続検出を行い、接続検出ステータス(H_SIE_IntStat_0.DetectCon)を発行します(T1)。
- (4) “GoDISABLED”と同等の処理を実行します(T1)。
- (5) ディセーブルド完了ステータス(H_SIE_IntStat_1.DisabledCmp)を発行します(T2)。

なお、各ステートにおけるタイミングは、“GoWAIT_CONNECT”、“GoDISABLED”を実行した場合と同じです。タイミングの詳細は、それぞれ「1.3.10.3.2 GoWAIT_CONNECT」、「1.3.10.3.3 GoDISABLED」を参照してください。

また、途中で異常(切断、VBUS 異常)を検出した場合の実行手順およびタイミングについては、「1.3.10.2.2.切断検出」、「1.3.10.2.1.VBUS 異常検出」を参照してください。

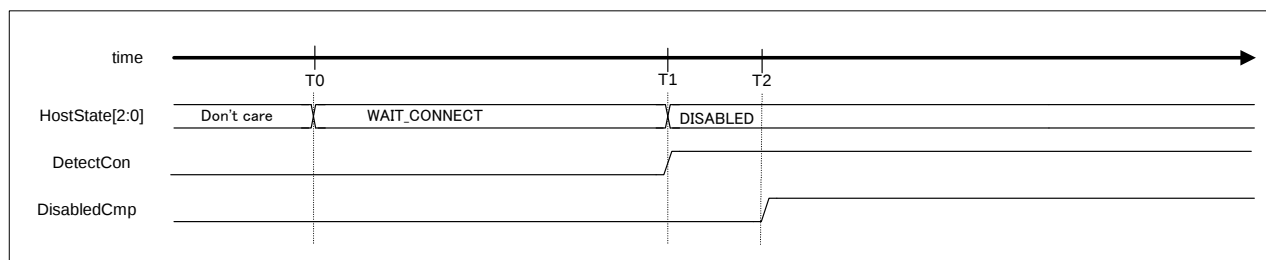


図 1.65 GoWAIT_CONNECTtoDIS Timing (HS mode)

表 1.59 GoWAIT_CONNECTtoDIS Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoWAIT_CONNECTtoDIS”を設定する。(F/W) GoWAIT_CONNECT と同等の処理を実行する。(H/W)	0 (reference)
T1	接続検出を行い、接続検出ステータスを発行する。 GoDISABLED と同等の処理を実行する。(H/W)	T1
T2	ディセーブルド完了ステータスを発行する。(H/W)	T2

1. 機能説明

1.3.10.3.9 GoWAIT_CONNECTtoOP

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECTtoOP”を設定すると、“WAIT_CONNECT”ステートから“OPERATIONAL”ステートに至る必要な処理を本 LSI のハードウェアが自動的に実行します。

1.3.10.3.9.1 HSデバイスが接続されている場合

HS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(9)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoWAIT_CONNECTtoOP”を設定します(T0)。
- (2) “GoWAIT_CONNECT”と同等の処理を実行します(T0)。
- (3) 接続検出を行い、接続検出ステータス(H_SIE_IntStat_0.DetectCon)を発行します(T1)。
- (4) “GoDISABLED”と同等の処理を実行します(T1)。
- (5) ディセーブルド完了ステータス(H_SIE_IntStat_1.DisabledCmp)を発行します(T2)。
- (6) “GoRESET”と同等の処理を実行します(T2)。
- (7) デバイスの Chirp を検出し、デバイスチャープ正常検出ステータス(H_SIE_IntStat_0.DetectDevChirpOK)を発行します(T3)。
- (8) リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T4)。
- (9) “GoOPERATIONAL”と同等の処理を実行します(T4)。

なお、各ステートにおけるタイミングは、“GoWAIT_CONNECT”、“GoDISABLED”、“GoRESET”、“GoOPERATIONAL”を実行した場合と同じです。タイミングの詳細は、それぞれ「1.3.10.3.2 GoWAIT_CONNECT」、「1.3.10.3.3 GoDISABLED」、「1.3.10.3.4 GoRESET」、「1.3.10.3.5 GoOPERATIONAL」を参照してください。

また、途中で異常(切断、VBUS 異常、デバイスチャープ異常)を検出した場合の実行手順およびタイミングについては、「1.3.10.2.2 切断検出」、「1.3.10.2.1 VBUS 異常検出」を参照してください。

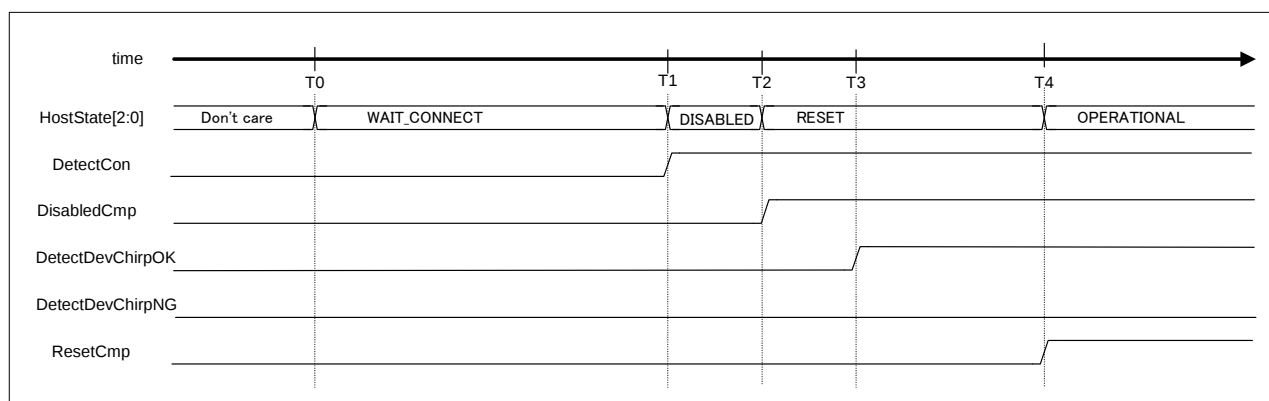


図 1.66 GoWAIT_CONNECTtoOP Timing (HS mode)

表 1.60 GoWAIT_CONNECTtoOP Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoWAIT_CONNECTtoOP”を設定する。(F/W) “GoWAIT_CONNECT”と同等の処理を実行する。(H/W)	0 (reference)
T1	接続検出を行い、接続検出ステータスを発行する。 “GoDISABLED”と同等の処理を実行する。(H/W)	T1
T2	ディセーブルド完了ステータスを発行する。 “GoRESET”と同等の処理を実行する。(H/W)	T2
T3	デバイスチャープ検出を行い、デバイスチャープ正常検出ステータスを発行する。(H/W)	T3
T4	リセット完了ステータスを発行する。 “GoOPERATIONAL”と同等の処理を実行する。(H/W)	T4

1. 機能説明

1.3.10.3.9.2 FSまたはLSデバイスが接続されている場合

FS または LS デバイスが接続されている場合の実行手順は以下のとおりです。
下記(2)～(9)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に
“GoWAIT_CONNECTtoOP”を設定します(T0)。
- (2) “GoWAIT_CONNECT”と同等の処理を実行します(T0)。
- (3) 接続検出を行い、接続検出ステータス(H_SIE_IntStat_0.DetectCon)を発行し
ます(T1)。
- (4) “GoDISABLED”と同等の処理を実行します(T1)。
- (5) ディセーブルド完了ステータス(H_SIE_IntStat_1.DisabledCmp)を発行しま
す(T2)。
- (6) “GoRESET”と同等の処理を実行します(T2)。
- (7) デバイスの Chirp を検出しないため、デバイスチャープ正常／異常検出ス
テータス(H_SIE_IntStat_0.DetectDevChirpOK/NG)は発行しません(T3)。
- (8) リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T4)。
- (9) “GoOPERATIONAL”と同等の処理を実行します(T4)。

なお、各ステートにおけるタイミングは、“GoWAIT_CONNECT”、
“GoDISABLED”、“GoRESET”、“GoOPERATIONAL”を実行した場合と同じで
す。タイミングの詳細はそれぞれ「1.3.10.3.2 GoWAIT_CONNECT」、「1.3.10.3.3
GoDISABLED」、「1.3.10.3.4 GoRESET」、「1.3.10.3.5 GoOPERATIONAL」を
参照してください。

また、途中で異常(切断、VBUS 異常)を検出した場合の実行手順およびタイミ
ングについては、「1.3.10.2.2 切断検出」、「1.3.10.2.1 VBUS 異常検出」を参
照してください。

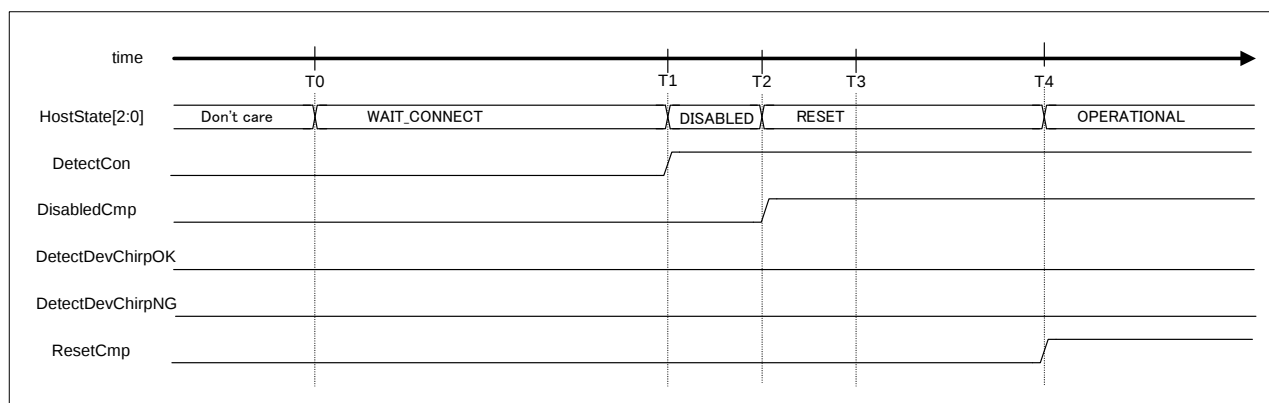


図 1.67 GoWAIT_CONNECTtoOP Timing (FS または LS mode)

表 1.61 GoWAIT_CONNECTtoOP Timing Values (FS または LS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に GoWAIT_CONNECTtoOP を設定する。(F/W) GoWAIT_CONNECT と同等の処理を実行する。(H/W)	0 (reference)
T1	接続検出を行い、接続検出ステータスを発行する。 GoDISABLED と同等の処理を実行する。(H/W)	T1
T2	ディセーブルド完了ステータスを発行する。 GoRESET と同等の処理を実行する。(H/W)	T2
T3	デバイスチャープ非検出のため、デバイスチャープ正常／異常検出ステータスを発行しない。(H/W)	T3
T4	リセット完了ステータスを発行する。 GoOPERATIONAL と同等の処理を実行する。(H/W)	T4

1. 機能説明

1.3.10.3.10 GoRESETtoOP

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESETtoOP”を設定すると、“RESET”ステートから“OPERATIONAL”ステートに至る必要な処理を本 LSI のハードウェアが自動的に実行します。

1.3.10.3.10.1 HSデバイスが接続されている場合

HS デバイスが接続されている場合の実行手順は以下のとおりです。下記(2)～(5)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESETtoOP”を設定します(T0)。
- (2) “GoRESET”と同等の処理を実行します(T0)。
- (3) デバイスの Chirp を検出し、デバイスチャープ正常検出ステータス(H_SIE_IntStat_0.DetectDevChirpOK)を発行します(T1)。
- (4) リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T2)。
- (5) “GoOPERATIONAL”と同等の処理を実行します(T2)。

なお、各ステートにおけるタイミングは、“GoRESET”、“GoOPERATIONAL”を実行した場合と同じです。タイミングの詳細はそれぞれ「1.3.10.3.4 GoRESET」、「1.3.10.3.5 GoOPERATIONAL」を参照してください。

また、途中で異常(VBUS 異常およびデバイスチャープ異常)を検出した場合の実行手順およびタイミングについては、「1.3.10.2.2 切断検出」、「1.3.10.2.1 VBUS 異常検出」を参照してください。

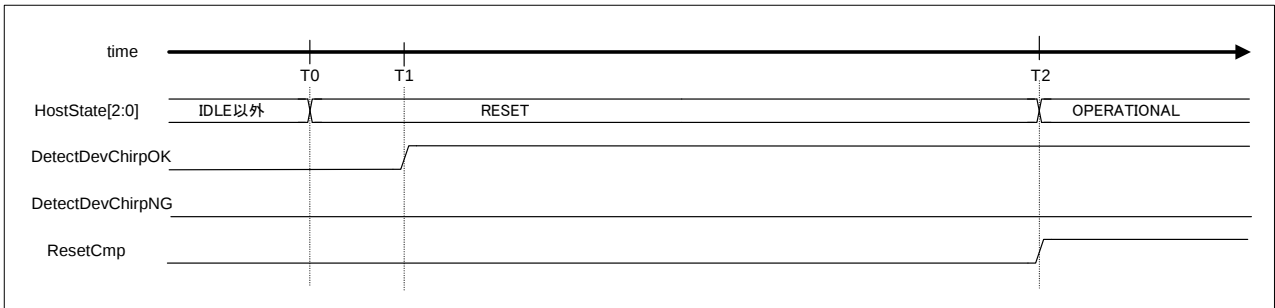


図 1.68 GoRESETtoOP Timing (HS mode)

表 1.62 GoRESETtoOP Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESETtoOP”を設定する。(F/W) “GoRESET”と同等の処理を実行する。(H/W)	0 (reference)
T1	デバイスチャープ検出を行い、デバイスチャープ正常検出ステータスを発行する。(H/W)	T1
T2	リセット完了ステータスを発行する。 “GoOPERATIONAL”と同等の処理を実行する。(H/W)	T2

1.3.10.3.10.2 FSまたはLSデバイスが接続されている場合

FS または LS デバイスが接続されている場合の実行手順は以下のとおりです。
下記(2)～(5)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に “GoRESETtoOP”を設定します(T0)。
- (2) “GoRESET”と同等の処理を実行します(T0)。
- (3) デバイスの Chirp を検出しないため、デバイスチャープ正常／異常検出ステータス(H_SIE_IntStat_0.DetectDevChirpOK/NG)は発行しません(T1)。
- (4) リセット完了ステータス(H_SIE_IntStat_1.ResetCmp)を発行します(T2)。
- (5) “GoOPERATIONAL”と同等の処理を実行します(T2)。

なお、各ステートにおけるタイミングは、GoRESET、GoOPERATIONAL を実行した場合と同じです。タイミングの詳細は、それぞれ「1.3.10.3.4 GoRESET」、
「1.3.10.3.5 GoOPERATIONAL」を参照してください。

また、途中で異常(VBUS 異常)を検出した場合の実行手順およびタイミングについては、「1.3.10.2.1 VBUS 異常検出」を参照してください。

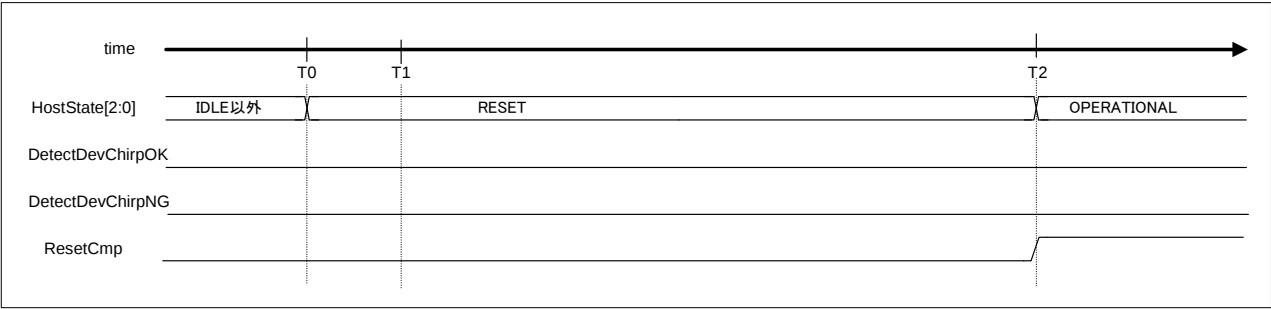


図 1.69 GoRESETtoOP Timing (FS または LS mode)

表 1.63 GoRESETtoOP Timing Values (FS または LS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に “GoRESETtoOP” を設定する。(F/W) “GoRESET”と同等の処理を実行する。(H/W)	0 (reference)
T1	デバイスチャープ非検出のため、デバイスチャープ正常／異常検出ステータスを発行しない。(H/W)	T1
T2	リセット完了ステータスを発行する。 “GoOPERATIONAL”と同等の処理を実行する。(H/W)	T2

1. 機能説明

1.3.10.3.11 GoSUSPENDtoOP

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoSUSPENDtoOP”を設定すると、“SUSPEND”ステートから“OPERATIONAL”ステートに至る必要な処理を本 LSI のハードウェアが自動的に実行します。

“GoSUSPENDtoOP”を設定した場合は、リモート・ウェークアップ検出機能をハードウェアが自動的にオン／オフします(ただしリモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)には反映しません)ので、ファームウェアがリモート・ウェークアップ受付許可(H_NegoControl_1.RmtWkupDetEnb)を操作する必要はありません。

なお、本設定を使用する際には、パワーマネジメント機能を使用しないでください。

本設定による実行手順は以下のとおりです。下記(2)～(7)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoSUSPENDtoOP”を設定します(T0)。
- (2) “GoSUSPEND”と同等の処理を実行します(T0)。
- (3) サスペンド遷移完了ステータス(H_SIE_IntStat_1.SuspendCmp)を発行します(T1)。
- (4) リモート・ウェークアップを検出し、リモート・ウェークアップを検出ステータス(H_SIE_IntStat_0.DetectRmtWkup)を発行します(T2)。
- (5) “GoRESUME”と同等の処理を実行します(T2)。
- (6) レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行します(T3)。
- (7) “GoOPERATIONAL”と同等の処理を実行します(T3)。

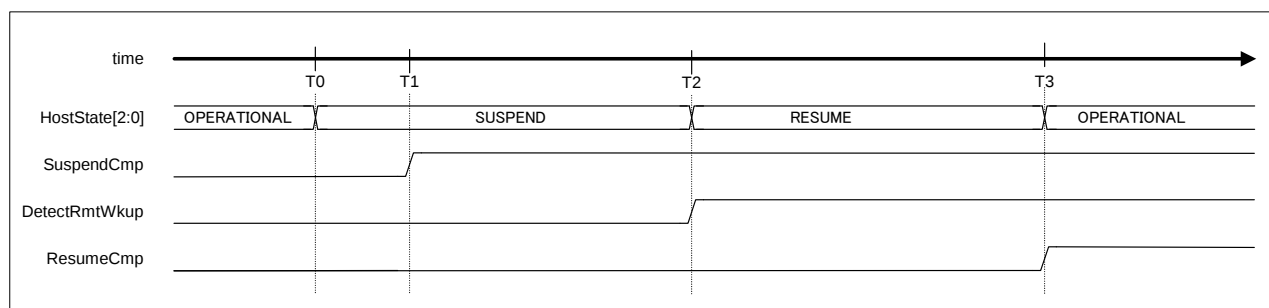


図 1.70 GoSUSPENDtoOP Timing

表 1.64 GoSUSPENDtoOP Timing Value

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoSUSPENDtoOP”を設定する。(F/W) “GoSUSPEND”と同等の処理を実行する。(H/W)	0 (reference)
T1	サスペンド遷移完了ステータスを発行する。(H/W)	T1
T2	リモート・ウェークアップを検出し、リモート・ウェークアップ検出ステータスを発行する。“GoRESUME”と同等の処理を実行する。(H/W)	T2
T3	レジューム完了ステータスを発行する。“GoOPERATIONAL”と同等の処理を実行する。(H/W)	T3

1.3.10.3.12 GoRESUMEtoOP

ホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUMEtoOP”を設定すると、“RESUME”ステートから“OPERATIONAL”ステートに至る必要な処理を本 LSI のハードウェアが自動的に実行します。

本設定による実行手順は以下のとおりです。下記(2)～(4)は本 LSI のハードウェアが自動的に実行します。

- (1) ファームウェアがホストステート遷移実行(H_NegoControl_0.AutoMode)に“GoRESUMEtoOP”を設定します(T0)。
- (2) “GoRESUME”と同等の処理を実行します(T0)。
- (3) レジューム完了ステータス(H_SIE_IntStat_1.ResumeCmp)を発行します(T1)。
- (4) “GoOPERATIONAL”と同等の処理を実行します(T1)。

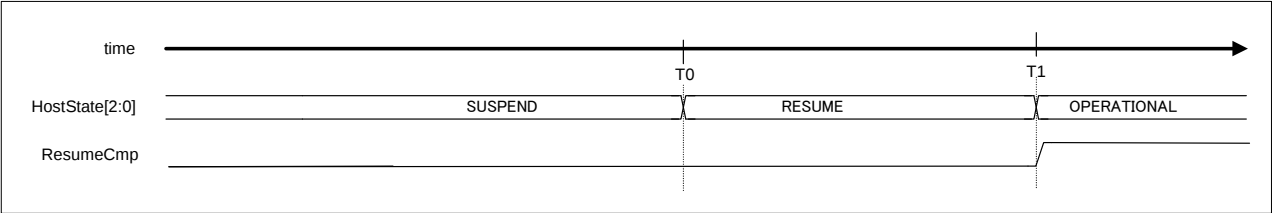


図 1.71 GoRESUMEtoOP

表 1.65 GoRESUMEtoOP Timing Value

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に“GoRESUMEtoOP”を設定する。(F/W) “GoRESUME”と同等の処理を実行する。(H/W)	0 (reference)
T1	レジューム完了ステータスを発行する。 “GoOPERATIONAL”と同等の処理を実行する。(H/W)	T1

1. 機能説明

1.4 パワーマネージメント機能

オシレータ、PLL の動作を制御し、スリープ、アクティブのステートを遷移します。ただし、スヌーズ・ステートはスリープ⇄アクティブのステート遷移時に経由するだけのステートです。他のステートに遷移する場合には、PM_Control レジスタの GoSLEEP、GoACTIVE ビットをセットすることで開始され、任意の処理が行われた後終了します。現在、どのステートにいるかを確認するには、PM_Control.PM_State [1:0]を確認してください。また遷移終了時、MainIntStat.FinishedPM イベントが発生します。この時、MainIntEnb.EnFinishedPM ビットをセットしていれば、割り込み XINT が発生します。ただし、この PM_Control.PM_State [1:0]は、PM_Control レジスタの GoSLEEP、GoACTIVE ビットをセットしてから、MainIntStat.FinishedPM イベントが発生するまでは、正確なステートを表示できないことに注意してください。

アクティブ・ステート時に PM_Control.GoSLEEP ビットをセットした場合は、スヌーズ・ステートを経由し、スリープ・ステートに遷移し、完全に遷移終了した時点で Main_IntStat.FinishedPM イベントが発生します。またスリープ・ステート時に PM_Control.GoACTIVE ビットをセットした場合も、スヌーズ・ステートを経由し、アクティブ・ステートに遷移し、完全に遷移終了した時点で MainIntStat.FinishedPM イベントが発生します。

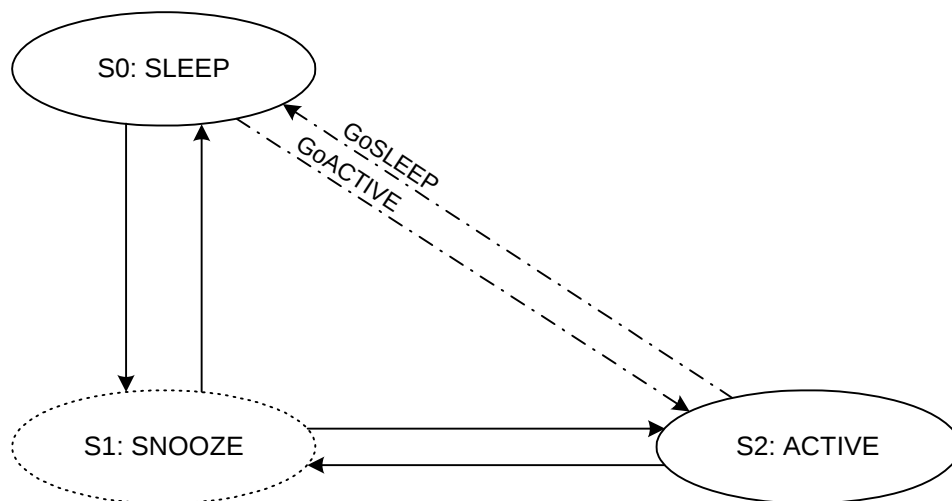


図 1.72 パワーマネージメント

1.4.1 SLEEP (スリープ)

オシレータが発振していない、または CLKIN 端子からの CLK 入力をゲートさせているステートです。したがって、このステートでは、PLL も発振していません。**斜体太字**のレジスタ及びビットはスリープ・ステートでも読み書き可能なレジスタです。**斜体文字**以外のレジスタにつきましては、アクティブ・ステートで無ければ読み書きできず、読むと全て 0 として読み出されてしまいます。

アクティブ・ステート中に PM_Control.GoSLEEP ビットをセットし、スリープに遷移する場合は、動作している内部 PLL を停止し、OSCCLK の内部への出力を停めた後に発振を停止します。

逆に、スリープ・ステート中に PM_Control.GoACTIVE ビットをセットし、スリープからスヌーズへ離脱する時には、オシレータが安定して発振するまで内部回路には OSCCLK を与えないように、発振安定時間ゲートしています。この発振安定時間は、発振セル、発振子、周辺回路及び基板によって変化しますので、WakeUpTim_H,L レジスタを用いて設定してください。

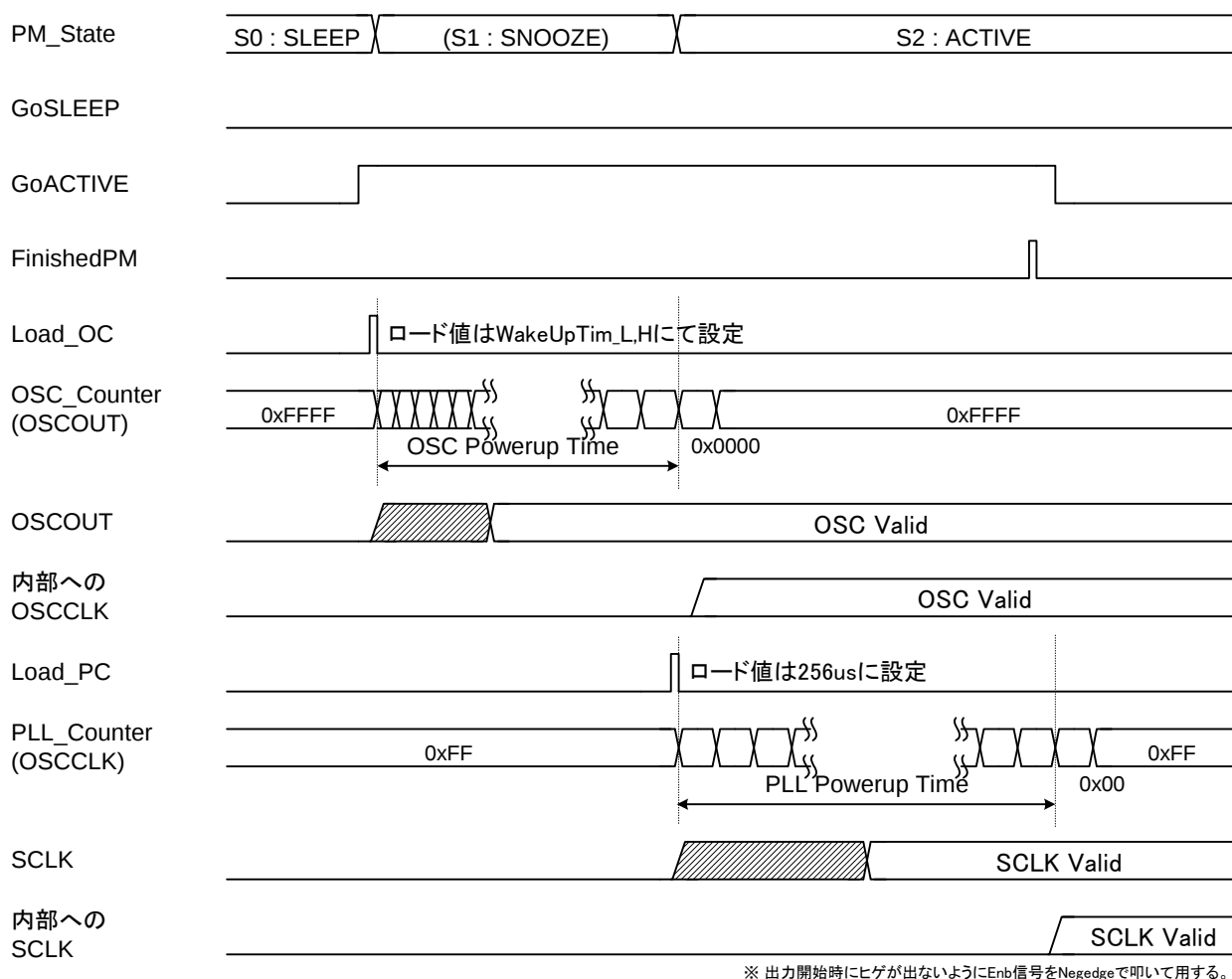


図 1.73 SLEEP ステートからの離脱 (GoACTIVE 時)

1.4.2 SNOOZE (スヌーズ)

オシレータは発振している状態で、PLL が発振していないステートです。スリープ⇄アクティブのステート遷移時に経由するだけのステートです。

スヌーズからアクティブへ離脱する時には、PLL が安定して発振するまで内部回路には SCLK を与えないように、PLL 安定時間 (約 250us) ゲートします。

1.4.3 ACTIVE (アクティブ)

オシレータ、PLL が動作しているステートです。USB 転送を含む全てのデータ転送が可能です。

1. 機能説明

1.4.4 CPU_Cutモード

スリープ・ステートにおいて、PM_State.GoCPU_Cut ビットをセットすることで、XCS を除く CPU-I/F の信号の入力を IC 初段から停止します。これにより、スリープ・ステートで CPU-I/F が動作している場合においても、本 LSI の入力端子のスイッチングは行われず、完全停止状態（IQ 状態）とほぼ同等の低消費電力化が図れます。

CPU_Cut モード状態から復帰するためには、PM_Control レジスタをダミーリードしてください。そのリード動作終了と同時に CPU_Cut モードから復帰します。この時、復帰動作を行うために XCS 信号の立ち上がり信号を使用しているため、ダミーリードの後には必ず XCS 信号がネゲート（“H”になる）されるようにしてください。ダミーリード後に XCS 信号がネゲートされずに、すぐに他のレジスタをリードする場合は、CPU_Cut モードが継続されています。

1.5 FIFO管理

FIFO 管理について説明します。

1.5.1 FIFOメモリマップ

FIFO のメモリマップを以下に示します。

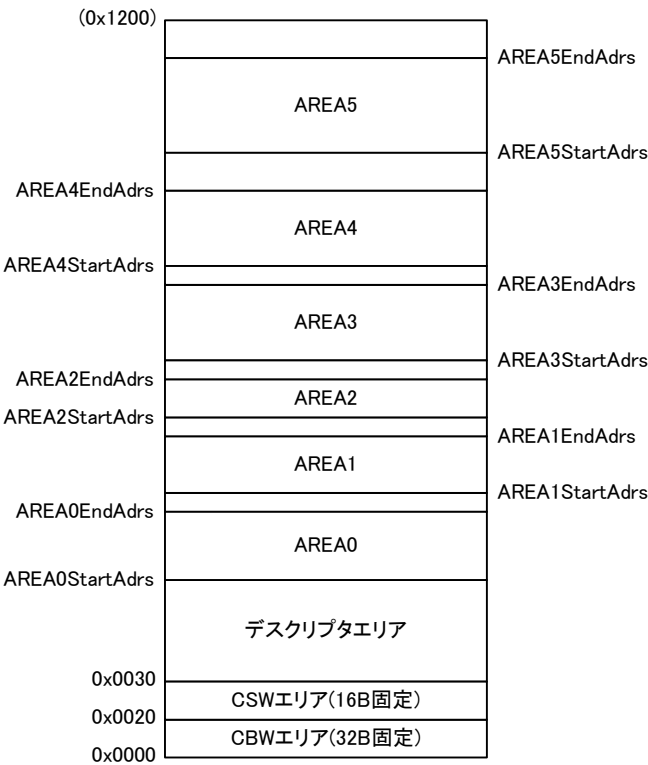


図 1.74 FIFO メモリマップ(例)

FIFO メモリは、CBW エリア、CSW エリア、デスクリプタエリア、AREA0、AREA1、AREA2、AREA3、AREA4、AREA5 に分割して使用できます。CBW エリア、CSW エリアについては図 1.74 のように固定領域が割り当てられます。一方その他の AREA_x{x=0-5}については FIFO エリア設定レジスタ (AREA_x{x=0-5}StartAdrs_H,L, AREA_x{x=0-5}EndAdrs_H,L)で領域を柔軟に設定することが可能です。デスクリプタエリアは未使用の領域を自由に使用できます。

デスクリプタエリアは、USB デバイス時にデスクリプタ返信機能で使用される領域です。FIFO の未使用領域のどの位置からでも使用することが出来ます。実際の使用方法については、「1.5.2 デスクリプタエリア」に後述します。全ての FIFO 領域をデスクリプタ返信機能に使用設定す

ることが可能となっていますが、競合を避けるために、ここで表したエリアでの使用を推奨します。

CBW エリアは、USB デバイス時にバルクオンリーサポート機能の CBW サポートで使用される領域です。32 バイト確保されていますが、そのうち、アドレス 0x0000 から始まる 31 バイトの領域を使用します。実際の使用方法については、「1.5.3.1 CBW エリア(USB デバイス時)」に後述します。また、この CBW エリアは USB ホスト時には CHa のバルクオンリーサポート機能で使用される領域です。実際の使用方法については「1.5.3.2 CBW エリア(USB ホスト時)」に後述します。

CSW エリアは、USB デバイス時にバルクオンリーサポート機能の CSW サポートで使用される領域です。16 バイト確保されていますが、そのうち、アドレス 0x0020 から始まる 13 バイトの領域を使用します。実際の使用方法については、「1.5.4.1 CSW エリア(USB デバイス時)」に後述します。また、この CSW エリアは USB ホスト時には CHa のバルクオンリーサポート機能で使用される領域です。実際の使用方法については「1.5.4.2 CSW エリア(USB ホスト時)」に後述します。

AREA0, AREA1, AREA2, AREA3, AREA4, AREA5 は、USB デバイス時はエンドポイント (EPx{x=0, a-e}) とジョインすることにより使用可能な、エンドポイントのエリアです。USB ホスト時には、同様にチャネル(CHx{x=0, a-e})に Join することで使用可能となります。ジョインするには、AREA ジョイン設定レジスタ (AREAx{x=0-5}Join_1) の JoinEPxCHx{x=0,a-e} ビットを設定します。なお、同じエンドポイント、チャネルを複数のエリアに同時にジョインしないで下さい。

AREA0, AREA1, AREA2, AREA3, AREA4, AREA5 それぞれのエリアは、FIFO として制御されており、データ格納数が保持されています。この保持された状態をクリアするためには、AREAnFIFO_Clr.ClearAREAx{x=0-5}の各ビットをセットして下さい。

なお、これらの状態クリアは、データ保持情報の初期化動作のみを行い、データは書き込みまたはクリアをしません。従って、このビットにより RAM 上のデータがクリアされることはありませんので、デスクリプタエリア内に記録された情報は消えることは無く、クリア後に再度データを書き込み直す必要はありません。

1.5.2 デスクリプタエリア

デスクリプタエリアは、USB デバイス時にデスクリプタ返信機能に使用するエリアです。デスクリプタ返信機能は、エンドポイント 0 において、データステージが IN 転送で行われる場合に使用することが出来ます。

IN 方向のデータステージを行う場合に、この領域内に書き込まれたデータの先頭アドレスと、返信するデータサイズを設定し、デスクリプタ返信機能を実行すると、自動的にデータステージが行われます。

デバイスデスクリプタ等、機器で一意に決定されるような内容を、電源投入後の初期設定時等にこのエリアに書き込んでおくことにより、リクエストを受け付けるとこのエリアのデータを返信するよう指示することが出来ます。リクエスト毎に EP0 の FIFO エリアにデータを書き込む必要が無く、高速に応答することが出来ます。

1.5.2.1 デスクリプタエリアへのデータの書き込み

デスクリプタエリアへデータを書き込むには、RAM_WrDoor 機能を使用します。書き込み開始アドレスを RAM_WrAdrs_H,L レジスタに設定し、RAM_WrDoor_0,1 レジスタにデータを書き込むことによって行います。RAM_WrAdrs_H,L レジスタの値は、一回の書き込み毎に、書き込みデータ数ずつ更新されますので、連続したアドレスにデータを書き込む場合には、RAM_WrDoor_0,1 レジスタに連続して書き込むことが出来ます。

なお、RAM_WrDoor_0,1 レジスタは書き込みのみ可能となっています。

1. 機能説明

1.5.2.2 デスクリプタエリアでのデータステージ(IN)の実行

書き込まれたデータを、デスクリプタ返信機能にて使用する場合には、D_DescAdrs_H,L レジスタに、データステージに送信するデータの先頭アドレスを設定し、返信するデータサイズを D_DescSize_H,L レジスタに設定した後、D_EP0Control.ReplyDescriptor ビットに 1 をセットします。また、D_EP0Control.INxOUT ビットに“1”をセットして、IN トランザクションを行えるようにします。また、データステージの IN トランザクションにデータパケットを返信出来るよう、D_SETUP_Control.ProtectEP0 をクリアしてから、D_EP0Control_IN.ForceNAK ビットをクリアするようご注意ください。

設定後、ホストからの IN トランザクションに応答し、マックスパケットサイズ (D_EP0MaxSize にて設定)に自動的に分割しながら、ホストに D_DescSize_H,L レジスタに設定されたデータ数までデータパケットを返信します。また、D_DescSize_H,L レジスタの値がマックスパケットサイズに満たない場合や、分割した後の残りのデータ数がマックスパケットサイズに満たない場合には、自動的にショートパケットとして送信します。

ホストから OUT トランザクションが発行されると、D_EP0Control.ReplyDescriptor をクリアし、D_EP0IntStat.DescriptorCmp がセットされます。F/W はステータスステージの処理に移行して下さい。

1.5.3 CBWエリア

1.5.3.1 CBWエリア(USBデバイス時)

CBW エリアは、USB デバイス時にバルクオンリーサポート機能の CBW サポートに使用するエリアです。Bulk OUT のエンドポイント(エンドポイント EPa, EPb, EPc, EPd, EPe)で、Bulk Only Transport Protocol のコマンドトランスポートを行うときに、このエリアに受信することが出来ます。こうすることにより、エンドポイントの FIFO では、データトランスポートで受信したデータのみを扱うことができます。

CBW サポートを実行している時に、対象となるエンドポイントで OUT トランザクションが行われ、データサイズが 31 バイトであると、CBW エリアに受信します。データ長が 31 バイトで無かった場合は、エラーステータスを発行し、データは破棄します。

CBW エリアに受信したデータを読み出すには、RAM_Rd 機能を使用します。RAM_RdControl.RAM_GoRdCBW_CSW ビットをセットすると、CBW エリアのデータが読み出されて RAM_Rd_00~RAM_Rd_1E レジスタにコピーされ、完了ステータス (CPU_IntStat.RAM_RdCmp ビット)が発行されます。

1.5.3.2 CBWエリア(USBホスト時)

CBW エリアは、USB ホスト時にバルクオンリーサポート機能で使用するエリアです。チャネル CHa で、Bulk Only Transport Protocol のコマンドトランスポートを行う際に、これらのエリアから CBW データをデータパケットとして送信します。こうすることにより、チャネルの FIFO では、データトランスポートで送信するデータのみを扱うことができます。

データパケットを送信する前に、CBW エリアには 0x0000 から CBW データ(31 バイト)を用意して下さい。

CBW エリアにデータを書き込むには、RAM_WrDoor 機能を使用します。RAM_WrAdrs_H,L レジスタに、CBW,エリアの先頭アドレス(0x0000)を書き、RAM_WrDoor_0,1 レジスタを介して 31 バイトの有効データを書き込みます。CBW エリアは 32 バイト確保されていますので、ワードアクセスで 32 バイト書き込んでも、他の領域を侵すことは無く、問題ありません。

1.5.4 CSWエリア

1.5.4.1 CSWエリア(USBデバイス時)

CSW エリアは、USB デバイス時、バルクオンリーサポート機能の CSW サポートに使用するエリアです。Bulk IN のエンドポイント(エンドポイント EPa, EPb, EPc, EPd, EPe)で、Bulk Only Transport Protocol のステータストランスポートを行うときに、このエリアから送信することが出来ます。こうすることにより、エンドポイントの FIFO では、データトランスポートで送信するデータのみを扱うことができます。

CSW サポートを実行している時に、対象となるエンドポイントで IN トランザクションが行われると、CSW エリアから 13 バイトのデータをデータパケットとして送信します。

CSW エリアにデータを書き込むには、RAM_WrDoor 機能を使用します。RAM_WrAdrs_H,L レジスタに、CSW エリアの先頭アドレス(0x0020)を書き、RAM_WrDoor_0,1 レジスタを介して 13 バイトの有効データを書き込みます。CSW エリアは 16Byte 確保されていますので、ワードアクセスで 14 バイト書き込んでも、他の領域を侵すことは無く、問題ありません。

1.5.4.2 CSWエリア(USBホスト時)

CSW エリアは、USB ホスト時バルクオンリーサポート機能で使用するエリアです。チャンネル CHa で、Bulk Only Transport Protocol のステータストランスポートを行う際に、このエリアに CSW データを受信します。こうすることにより、チャンネルの FIFO では、データトランスポートで受信するデータのみを扱うことができます。

CSW エリアに受信したデータを読み出すには、RAM_Rd 機能を使用します。RAM_RdControl.RAM_GoRdCBW_CSW ビットをセットすると、CSW エリアのデータが読み出され、RAM_Rd_00 ~ RAM_Rd_0C レジスタにコピーされ、完了ステータス(CPU_IntStat.RAM_RdCmp ビット)が発行されます。

1.5.5 FIFOへのアクセス方法

FIFO のデータにアクセスする方法は、RAM アクセスする方法と、FIFO アクセスする方法とがあります。FIFO アクセスの手段としては、CPU(レジスタ), CPU(DMA), USB が有ります。

RAM アクセスでは、FIFO の任意の場所からデータを読むことができ、また、任意の場所にデータを書くことができます。RAM アクセスによっては、FIFO のデータ数は更新されません。つまり、RAM_Rd 機能によって FIFO のデータを読み出しても、FIFO 領域は解放されません。また、RAM_WrDoor 機能によって FIFO にデータを書き込んでも、FIFO 領域を占有しません。

1.5.5.1 RAMアクセスの方法(RAM_Rd)

FIFO に RAM_Rd レジスタによってリードアクセスする場合には、読み出しを行いたい FIFO 領域の先頭アドレスとデータサイズを RAM_RdAdrs_H,L レジスタ、RAM_RdCount レジスタに設定し、RAM_RdControl.RAM_GoRd ビットを設定して下さい。指定された FIFO 領域のデータを RAM_Rd レジスタから読み出し可能になると CPU_IntStat.RAM_RdCmp ビットが“1”にセットされます。RAM_RdCmp ビットを確認した後 RAM_Rd_00 ~ RAM_Rd_1F レジスタからデータを読み出して下さい。読み出されるデータは RAM_Rd_00 から順に格納されます。RAM_RdCount レジスタに設定したサイズが 32 バイトより小さい場合は、設定サイズ以降の RAM_Rd レジスタの値は無効となります。

RAM_Rd レジスタによる FIFO データの読み出しは、FIFO 領域の設定に関係なく、任意に行う事が出来ます。

RAM_Rd 機能動作中は、RAM_RdAdrs_H,L レジスタ、RAM_RdCount レジスタの値が逐次更新されます。RAM_Rd 機能を開始した後は CPU_IntStat.RAM_RdCmp ビットがセットされるまで、これらのレジスタへアクセスしないで下さい。RAM_Rd 機能動作中にこれらのレジスタを読み出した場合の値は保証されません。また、これらのレジスタに書き込みを行った場合、誤動作の原因となります。

1. 機能説明

1.5.5.2 RAMアクセスの方法(RAM_WrDoor)

FIFO に RAM_WrDoor_0,1 レジスタによってライトアクセスする場合には、書き込み開始アドレスを RAM_WrAdrs_H,L レジスタに設定し、RAM_WrDoor_0,1 レジスタにてデータを書き込むことによって行います。一回のライトアクセス毎に、自動的に RAM_WrAdrs_H,L レジスタは書き込み数ずつインクリメントされますので、連続したアドレスにデータを書き込む場合には、RAM_WrDoor_0,1 レジスタに連続して書き込むことが出来ます。

RAM_WrDoor_0,1 レジスタによる FIFO への書き込みは、FIFO 領域の設定に関係なく、任意に行う事が出来ます。

なお、書き込み開始アドレスに以下の制限があります。

- (1) 書き込み開始アドレスを、RAM_WrAdrs_H,L == 0x0100 より下位とする。
- (2) 書き込み開始アドレスを、RAM_WrAdrs_H,L[1:0] == 0b00 (すなわち 4Byte 境界) とする。

1.5.5.3 FIFOアクセスの方法(レジスタアクセス)

FIFO にレジスタアクセスによってリードアクセスする場合には、いずれか1つのエリアに対して、AREAx{x=0-5}Join_0.JoinCPU_Rd に 1 をセットし、FIFO_Rd_0,1 レジスタ、または FIFO_ByteRd レジスタにて読み出しを行います。

また、FIFO にレジスタアクセスによってライトアクセスする場合には、いずれか1つのエリアに対して、AREAx{x=0-5}Join_0.JoinCPU_Wr に 1 をセットし、FIFO_Wr_0,1 レジスタ、または FIFO_ByteWr レジスタに書き込みを行います。

FIFO_RdRemain_H,L レジスタは、JoinCPU_Rd にてただ 1 つ設定されたエリアにおいて、FIFO から読み出し可能なデータの残り数を表しています。また、FIFO_WrRemain_H,L レジスタは、JoinCPU_Wr にてただ 1 つ設定されたエリアにおいて、FIFO に書き込み可能なエリアの残り数を表しています。

ICE 等を使用して F/W のデバッグを行うに際し、レジスタのダンプ等を行う場合に、JoinCPU_Rd ビットのいずれかがセットされていると、レジスタのダンプ時に FIFO からデータを読み出されてしまうことに注意してください。

CPU のレジスタアクセスは、FIFO とメディア FIFO とで排他的に Join する必要があります。つまり、メディア FIFO と FIFO のいずれかの領域の両方に、JoinCPU_Wr を同時に設定することは出来ません。同様に、メディア FIFO と FIFO のいずれかの領域の両方に、JoinCPU_Rd を同時に設定することは出来ません。

1.5.5.4 FIFOアクセスの方法(DMA)

FIFO に CPU の DMA アクセスによってリードアクセスする場合には、AREAx{x=0-5}Join_0.JoinDMA ビットによりただ一つのエリアを選択し、DMA_Control.Dir ビットに 1 を設定し、DMA 手順を実行して読み出しを行います。

また、FIFO に CPU の DMA アクセスによってライトアクセスする場合には、DMA のチャンネルに付き、AREAx{x=0-5}Join_0.JoinDMA ビットによりただ一つのエリアを選択し、DMA_Control.Dir ビットに 0 を設定し、DMA 手順を実行して書き込みを行います。

DMA_Remain_H,L レジスタは、DMA のチャンネルについて、AREAx{x=0-5}Join_0.JoinDMA ビットによりただ一つ選択されたエリアにおいて、FIFO から読み出し可能なデータの残り数を表しています。また、DMA のチャンネルについて、AREAx{x=0-5}Join_0.JoinDMA ビットによりただ一つ選択されたエリアにおいて、FIFO に書き込み可能な残り数を表しています。

1.5.5.5 FIFOアクセスの制限

本 LSI の FIFO には、USB との送受信、CPU バスからのレジスタ書き込み、読み出し、DMA を用いた書き込み、読み出しが同時に行われます。また、CPU バスからの読み出しについては、先読み処理を行っています。

これらのことから、各ポートの FIFO へのアクセス設定方法(Join)に対し、基本的に下記の排他ルールがあります。

- 同一の FIFO エリアに対し、複数の書き込み要因を同時に設定してはいけない。
- 同一の FIFO エリアに対し、複数の読み出し要因を同時に設定してはいけない。
- 一つのエリアには、JoinCPU_Wr, JoinCPU_Rd, JoinDMA のうちの一つしか設定できない。
- JoinCPU_Wr, JoinCPU_Rd, JoinDMA は、同時に一つのエリアにしか設定できない。

同一のエリアに対する書き込み／読み出し要因の重複に関し、USB(デバイス時)からのアクセス要因に関しては、例外的に下記の使用方法が可能です。例えば、OUT のエンドポイントの FIFO エリアに対し、JoinCPU_Wr や JoinDMA(DMA_Control.Dir==0)を設定して書き込むことが出来ます。この場合、必ず OUT トランザクションが行われない状況にした後に、JoinCPU_Wr または JoinDMA を設定し、CPU からの書き込みを行う必要があります。また、IN のエンドポイントの FIFO エリアに対し、JoinCPU_Rd や JoinDMA(DMA_Control.Dir==1)を設定して読み出すことが出来ます。この場合、必ず IN トランザクションが行われない状況にした後、JoinCPU_Rd または JoinDMA を設定し、CPU からの読み出しを行う必要があります。

トランザクションが行われない状況には、ActiveUSB ビットがクリアされている場合、各エンドポイントが FIFO 領域にジョインされていない場合や、ForceNAK がセットされている場合などがあります。

1. 機能説明

1.6 CPUIF

1.6.1 CPUIFモード

本 LSI の CPUIF は、BE モード／Strobe モードの 2 つの BUS モードをサポートします。各々の BUS モードで、ライトアクセスの方法が異なります。BE モードとは、ライトタイミングを示すライトストロブ信号、及び上位(CD[15:8])／下位(CD[7:0])バイトのイネーブル信号によってライトアクセスを行う方法です。一方、Strobe モードとは上位／下位バイトそれぞれに個別のライトストロブ信号を持ち、それらのライトストロブ信号によってライトアクセスを行う方法です(図 1.75 参照)。リードアクセスの方法は何れの BUS モードでも同一で、リードストロブ信号を用いて行います。

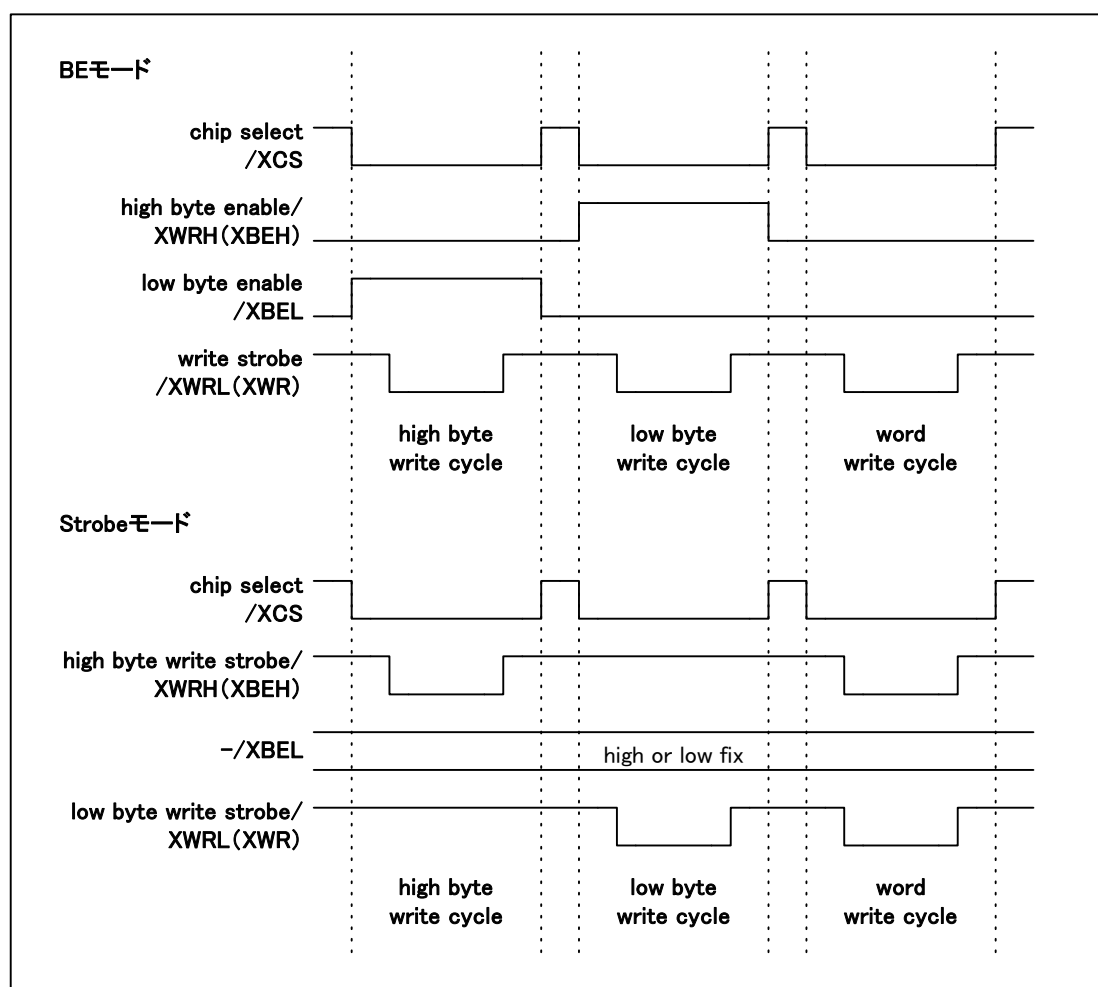


図 1.75 BUS モード

上述の BUS モードに加え、CPU のエンディアンに対応するため、本 LSI の CPUIF は BigEndian モード／LittleEndian モードの 2 つの ENDIAN モードをサポートします。BigEndian モードとは、偶数アドレスのレジスタが上位、奇数アドレスのレジスタが下位、また、1st バイトのデータが上位、2nd バイトのデータが下位になるモードです。LittleEndian モードとは、BigEndian モードの反対に、偶数アドレスのレジスタが下位、奇数アドレスのレジスタが上位、また、1st バイトのデータが下位、2nd バイトのデータが上位になるモードです。ENDIAN モードの詳細は「Appendix A リトルエンディアンの CPU との接続」をご参照ください。

接続する CPU に適合した、BUS モード、ENDIAN モードに設定した上で、本 LSI をご使用下さい。モードの設定方法に関しては、次項で説明します。

1.6.2 CPUIFのモード設定

本 LSI は、ハードウェアリセット後 CPUIF のモード設定待ちの状態(未初期化期間)となり、この状態においては、モード設定以外のアクセスを受け付けません。モード設定終了後にレジスタアクセスが可能になります(初期化済み期間)。モードの設定方法は以下の通りです。

- 1) XCS 及び XWRL 信号をアサートし、特定のデータパターンを用いて、モード設定データを書き込みます(モード設定)。例えば、本 LSI をマッピングしたメモリ空間の任意のアドレスにワードでデータライトします。
- 2) XCS 信号を一旦ネゲートします。例えば、NOP を挿入するか、或いは、本 LSI をマッピングしたメモリ空間以外にアクセスします。
- 3) XCS 信号を再度アサートします(モード確定)。例えば、1)と同様の動作をします。
- 4) XCS 信号を一旦ネゲートします。例えば、2)と同様の動作をします。

以上でモード設定は完了し、通常のレジスタアクセスが可能になります。モード設定の確認のため設定完了後に ChipConfig レジスタを読み出す事を推奨致します。

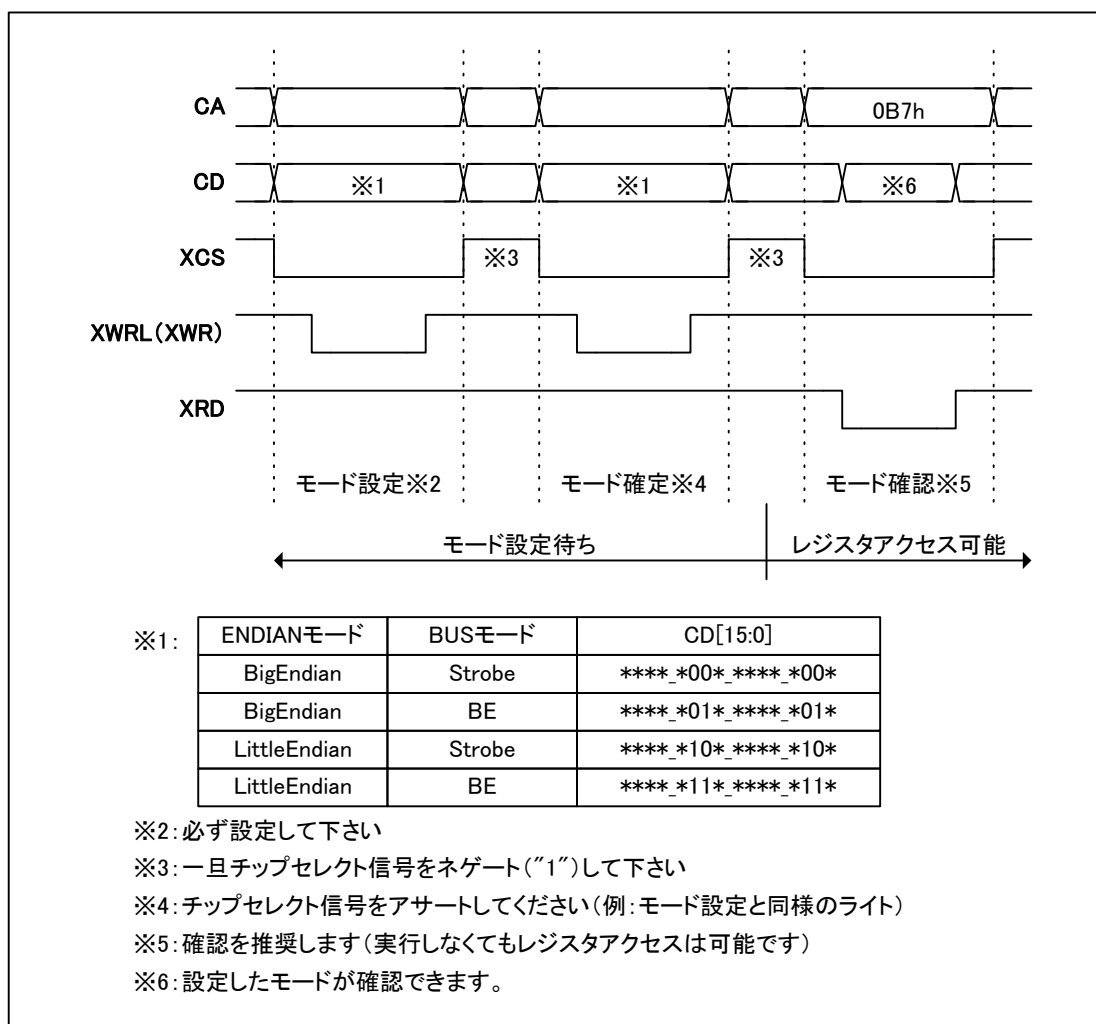


図 1.76 CPUIF のモード設定

モード設定の状態において、図 1.76 に記載以外の信号は、AC 規格を満足する限り、High/Low 何れの値でも問題ありません。

1. 機能説明

CPUIF のモード設定は、ハードリセット後に必ず行う必要があります、また、ハードリセット後の未初期化期間でのみ行えます。

以下の CPUIF の説明においては、Strobe モード／BigEndian モードを基本に説明致します。

1.6.3 ブロック構成

本 LSI の CPUIF(以下、CPUIF)は、REG/DMA の 2 ブロックから構成されます。

- REG : レジスタ領域へのアクセス制御
- DMA : DMA チャンネル

1.6.3.1 REG(Registers)

本 LSI のレジスタ領域のアクセスを制御します。以下のアクセス機能が含まれます。

- 同期レジスタアクセス
- FIFO アクセス
- RAM_Rd アクセス
- 非同期レジスタアクセス

1.6.3.1.1 同期レジスタアクセス(ライト)

内部クロックに同期して外部バスのデータをレジスタへ書き込みます。

1.6.3.1.2 同期レジスタアクセス(リード)

リード(XCS、XRD 共にアサート)期間を出力イネーブル期間としてレジスタのデータを外部バスへ出力します。

レジスタのリード動作において、カウント値など 3 バイト以上で意味を成すレジスタでは、アクセスサイクル間のカウント値の桁上がり等により誤ったカウント値をリードしない様、最上位バイトのリードタイミングで下位バイトのレジスタの値を保持し、下位バイトのリード時には、この値を外部バスへ出力するレジスタもあります。

1.6.3.1.3 FIFOアクセス(ライト)

FIFO ライトアクセスは FIFO_Wr_0,1、FIFO_ByteWr、RAM_WrDoor_0,1 レジスタへの書き込みを意味します。

FIFO ライトアクセスライトには以下の制限事項があります。

- AREAx{x=0-5}Join_0.JoinCPU_Wr ビットを設定した後、FIFO_WrRemain_H,L レジスタで書き込み可能なデータ数を確認した上でアクセスを行って下さい。RAM_WrDoor_0,1 レジスタにはこの制限はありません。
- 基本的にワード(2 バイト)単位でアクセスを行って下さい。端数(奇数)バイトの書き込みを行う場合の詳細は「1.6.3.1.5 FIFO アクセスの端数処理」を参照して下さい。
- FIFO_Wr_0,1、FIFO_ByteWr レジスタへのライト直後に FIFO_WrRemain_H,L レジスタを確認しても、正確な FIFO の空き領域を確認することが出来ません。必ずバスサイクル 1 回以上の間隔を空けて確認して下さい。
- RAM_WrDoor_0,1 レジスタへのライト直後に RAM_WrDoorAdrs_H,L レジスタを確認しても正確なアドレスを確認することが出来ません。必ずバスサイクル 1 回以上の間隔を空けて確認して下さい。

- FIFO_Wr_0,1、FIFO_ByteWr レジスタへのライト後、次にあげる動作を行う前に、データを FIFO に書き込み終えたことを、CacheRemain.WrFIFO_C_Remain ビットにより確認して下さい。守られなかった場合は、正しいデータの書き込みが保証されません。

＜WrFIFO_C_Remain ビットの確認が必要となる動作＞

- JoinCPU_Wr を切り替える、または、クリアする。
- AREAnFIFO_Clr にて、FIFO クリアする。
- RAM_WrDoor_0,1 レジスタへのライト後、次にあげる動作を行う前に、データを FIFO に書き込み終えたことを、CacheRemain.WrDOOR_C_Remain ビットにより確認して下さい。守られなかった場合は、正しいデータの書き込みが保証されません。

＜WrDOOR_C_Remain ビットの確認が必要となる動作＞

- RAM_WrDoorAdrs_H,L を書き換える。
- デバイス動作時、EnShortPkt をセットして、ショートパケットの送信を始める。
- デバイス動作時、ReplyDescriptor をセットして、データ送信を始める。
- デバイス動作時、GoCSW_Mode をセットして、CSW 送信を始める。
- ホスト動作時、BO_SupportGo をセットして、CBW 送信を始める。

1.6.3.1.4 FIFOアクセス(リード)

FIFO リードアクセスは FIFO_Rd_0,1 レジスタ、FIFO_ByteRd レジスタの読み出しを意味します。

FIFO リードアクセスには、以下の制限事項があります。

- AREAx{x=0-5}Join_0.JoinCPU_Rd ビットを設定した後、FIFO_RdRemain_H,L レジスタで読み出し可能データ数及び RdRemainValid ビットを確認した上でアクセスを行って下さい。
- ワード読みを行う場合は FIFO_Rd_0,1 レジスタを用いて行って下さい。バイト読みを行う場合は FIFO_ByteRd レジスタを用いて行って下さい。バイト境界が存在する場合はバイト読みを行って下さい。この場合に FIFO_Rd_0,1 レジスタを用いてワード読みを行った場合は片側にのみ有効なデータが出力されます。詳細は「1.6.3.1.5 FIFO アクセスの端数処理」ををご参照下さい。

1.6.3.1.5 FIFOアクセスの端数処理

端数(奇数)データを扱う場合の FIFO へのデータの格納状態と FIFO アクセスの関係を説明します。

【ライト動作】

基本的にはバイト境界の存在しない状態から書き込み動作を行う事を推奨致します。

AREAnFIFO_Clr.ClrAREAx{x=0-5}ビットをセットする等行って、バイト境界の存在しない状態からワード書き込みを行い、奇数データが存在する場合は、連続するデータの最終バイト(データ Z)のみ High 側に書き込んで下さい。この状態を図 1.77 の(1)に示します。USB 等からはデータ A, B, C, D, ・ ・ ・ X, Y, Z の順に出力されます。

FIFO にバイト境界がある状態から書き込みを行う場合は、最初に Low 側にデータを書き込み(データ K の書き込み)、バイト境界を解消した後、ワード書き込み(データ L, M)を行って下さい。この状態を図 1.77 の(2)に示します。

以上は、正常な書き込み動作です。

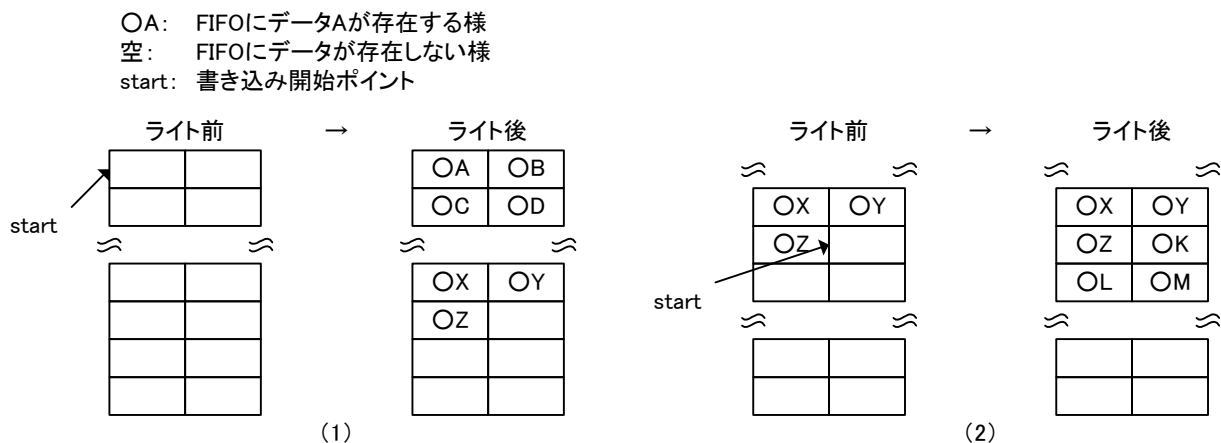


図 1.77 FIFO ライト処理(正常動作)

以下は注意が必要な書き込みを行った場合の動作です。

FIFO にバイト境界がある状態からワード書き込みを行った場合、High 側の書き込みは無視され Low 側のみ書き込みが行われます(図 1.78 の(3))。つまり、Low 側にバイト書き込みを行ったときと同じ動作を行います。また、FIFO にバイト境界のある状態から High 側にのみ書き込みを行った場合、その書き込みは無視されます(図 1.78 の(4))。

FIFO にバイト境界が無い状態から Low 側にのみ書き込みを行った場合、その書き込みは無視されます(図 1.78 の(5))。また、FIFO にバイト境界がなく、且つ書き込み可能数が“1”の状態からワード書き込みを行った場合、Low 側の書き込みは無視され、High 側のみ書き込みが行われます(図 1.78 の(6))。つまり、High 側にバイト書き込みを行ったときと同じ動作を行います。

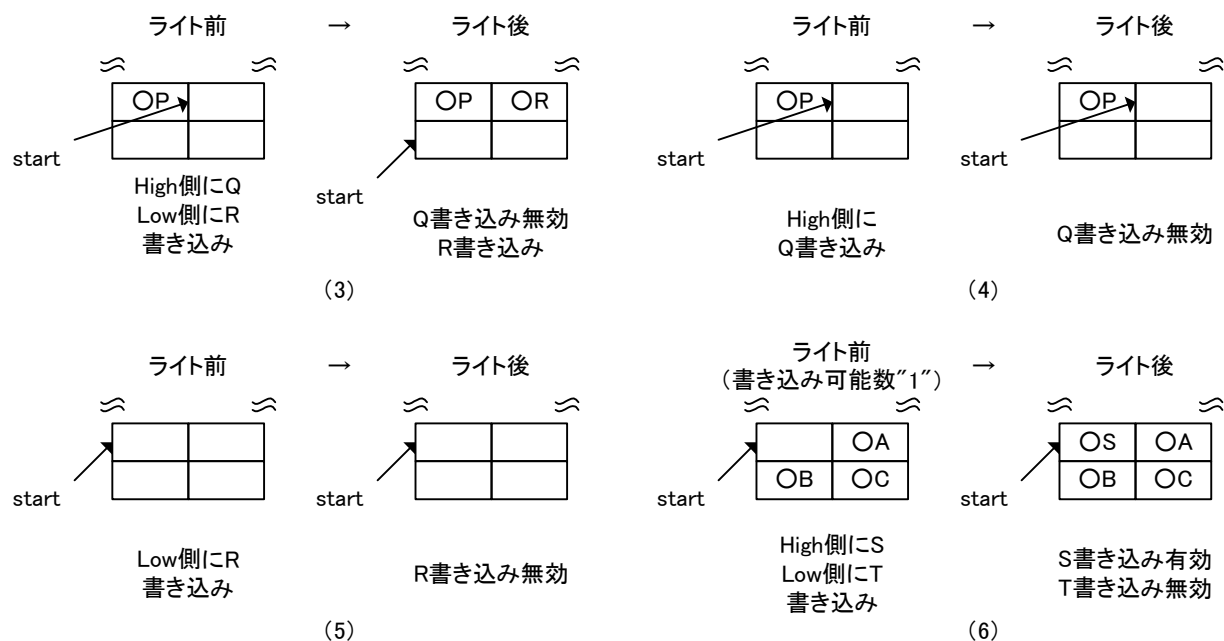


図 1.78 FIFO ライト処理(注意が必要な動作)

【リード動作】

バイト境界が無い場合は、FIFO_Rd_0,1 レジスタを用いたワード読み出し、FIFO_ByteRd レジスタを用いたバイト読み出し何れを行っても問題ありません。バイト境界がある場合は、FIFO_ByteRd レジスタを用いたバイト読み出しを行って下さい。一旦バイト境界を解消した後はワード読み出し、バイト読み出し何れを行っても問題ありません。

バイト境界が無い状態からのワード読み出しの様子を図 1.79 の(1)に示します。アクセス毎にデータ A, B → データ C, D と読み出されます。また、バイト読み出しの様子を図 1.79 の(2)に示します。アクセス毎にデータ A → データ B → データ C → データ D と読み出されます。以上は正常な読み出し動作です。

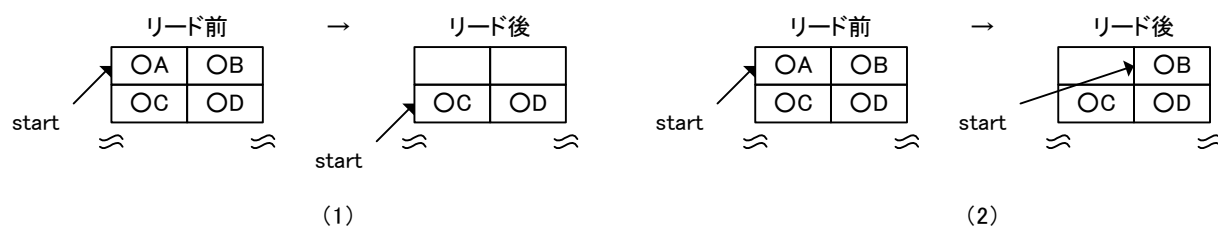


図 1.79 FIFO リード処理(正常動作)

以下は注意が必要な読み出しを行った場合の動作です。

図 1.80 の(3)はバイト境界がある状態から、FIFO_Rd_0,1 レジスタを用いてワード読み出しを行った場合の動作です。High 側には不定のデータが出力され、Low 側にデータ J が出力されます。リードのポインタは 1 バイト分のみ進みます。図 1.80 の(4)はバイト境界が無いが残りデータが 1 バイトの状態から、FIFO_Rd_0,1 レジスタを用いてワード読み出しを行った場合の動作です。High 側にはデータ X が出力され、Low 側には不定データが出力されます。リードのポインタは 1 バイト分のみ進みます。

1. 機能説明

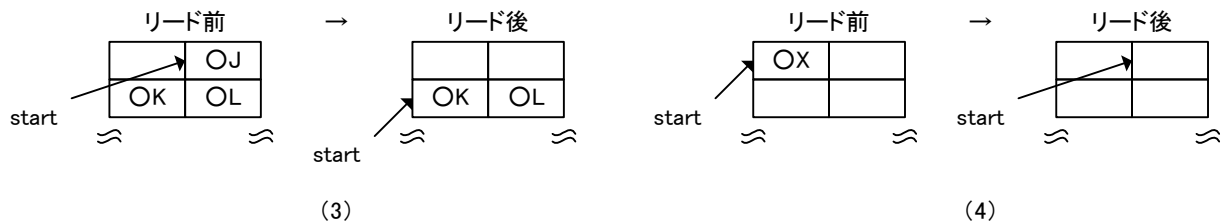


図 1.80 FIFO リード処理(注意が必要な動作)

上記より、端数処理のリード動作を行う場合の例を説明します。

- 1) USB から 64 バイト送られてきたデータを 31 バイト→33 バイトで読み出したい場合。
 - (1) CPUIF は 64 バイトのレディをラッチして一連のリードシーケンスを開始する。
 - (2) 30 バイト分のデータを FIFO_Rd_0,1 レジスタでワード読み、もしくは FIFO_ByteRd レジスタでバイト読みする。
 - (3) 31 バイト目のデータを FIFO_ByteRd レジスタでバイト読みする。→バイト境界発生。
 - (4) 32 バイト目のデータをバイト読みする。この場合 FIFO_ByteRd レジスタでのバイト読みを推奨します。FIFO_Rd_0,1 レジスタでのワード読みを行った場合は Low 側にデータが出力されます。→バイト境界解消。
 - (5) 残り 32 バイトのデータを FIFO_Rd_0,1 レジスタでワード読み、もしくは FIFO_ByteRd でバイト読みする。
- 2) JoinCPU_Rd がセットされている状態で USB から 31 バイト、33 バイトと送られてきたデータ 64 バイトを全て FIFO_Rd_0,1 レジスタでワード読みしたい場合。
 - (1) USB から 31 バイトのデータを受信した時点で CPUIF は 31 バイトのレディをラッチして一連の動作シーケンスを開始する。
 - (2) 30 バイト分のデータをワード読みする。
 - (3) キャッシュされている 31 バイト目のデータ(バイト境界)を解消するため、一旦ジョインを切り離す。
 - (4) 33 バイトのデータが USB から送られてきた後、再度ジョインする。(1+33 バイト)
 - (5) CPUIF は 34 バイトのレディをラッチして一連の動作シーケンスを開始する。
 - (6) 34 バイト分のデータをワード読みする。

1.6.3.1.6 RAM_Rdアクセス

同期レジスタリードと同様、リード(XCS、XRD 共にアサート)期間を出カイネーブル期間としてデータを外部バスへ出力します。詳細は「1.5.5.1 RAM アクセスの方法(RAM_Rd)」をご参照下さい。

1.6.3.1.7 非同期レジスタアクセス(ライト)

外部ライト信号(XCS、XWRL,H)からライトパルスを作成して外部バスのデータをレジスタへ書き込みます。

1.6.3.1.8 非同期レジスタアクセス(リード)

同期レジスタリードと同様、リード(XCS、XRD 共にアサート)期間を出カイネーブル期間としてレジスタのデータを外部バスへ出力します。

1.6.3.2 DMA

1.6.3.2.1 基本機能

DMA の基本動作は以下の通りです。

【ライト動作】

FIFO に書き込み可能な空き領域がある場合に XDREQ をアサートし DMA 転送が可能になります。

なお、DMA 転送によるライト動作を停止する場合は、必ず CPU_IntStat.DMA_Cmp ビットが 0b1 であることを確認してから、DMA_Control.DMA_Stop を設定してください。守られなかった場合は、正しいデータ書き込みが保証されません。

【リード動作】

FIFO に読み出し可能なデータがあり、読み出し動作可能になると XDREQ をアサートし DMA 転送が可能になります。

DMA のアクセス方法は以下の 3 通りあります。

- XDACK でアクセスする

ChipConfig.CS_Mode="0", DMA_Config.DMA_Mode="0" のとき、このアクセス方法となります。ChipConfig.DACK_Level に設定された論理レベルに XDACK 信号がアサートされると、DMA アクセスと見なして転送を行います。

- XDACK&XCS でアクセスする

ChipConfig.CS_Mode="1", DMA_Config.DMA_Mode="0" のとき、このアクセス方法となります。ChipConfig.DACK_Level に設定された論理レベルに XDACK 信号がアサートされ、かつ XCS がアサート("0")されると、DMA アクセスと見なして転送を行います。

- DMA_Mode でアクセスする

ChipConfig.CS_Mode="0", DMA_Config.DMA_Mode="1" のとき、このアクセス方法となります。DMA_Rd/WrData_H/L レジスタへのアクセスを DMA アクセスと見なして転送を行います。

何れの場合も、DMA_Config.ActiveDMA を"1"に設定して下さい。

DMA は二つの動作モードと一つの動作オプションを持ちます。

- カウントモード

設定されたカウント数分の DMA 転送を行います。

内部 FIFO に書き込み可能な空き領域／読み出し可能なデータがあり、且つ DMA_Count_HH,HL,LH,LL レジスタに残りカウントがある場合に XDREQ をアサートし DMA 転送が可能になります。

- フリーランモード

内部 FIFO に書き込み可能な空き領域／読み出し可能なデータがある場合に XDREQ をアサートし DMA 転送が可能になります。

- REQ アサートカウントオプション

CPU のバーストリード／バーストライトに対応するためのオプションです。このオプションは、カウントモード／フリーランモード何れでも使用が可能です。DMA_Config.ReqAssertCount[1:0] ビットに設定されたアサートカウント数以上の、書き込み可能な空き領域／読み出し可能なデータが FIFO にある場合に XDREQ をアサートし DMA 転送が可能になります。従って、基本的には一旦 XDREQ がアサートされると、設定されたアサートカウント数分の転送が保証されることに

1. 機能説明

なります。ただし、FIFO の空き領域／データが設定されたアサートカウント数未満の場合も、カウントモードに設定され、且つ FIFO の空き領域／データが残りカウント数以上の場合、XDREQ がアサートされます。この場合の保証される転送数は残りカウント数となります。

DMA の基本はワード単位でのデータ処理となります。バイト単位でのデータ処理はカウントモードで且つ残りカウント数が“1”の時のみ行うことができます。各々の動作モード、オプションにおける XDREQ のアサート条件、また XDREQ アサート時の転送可能数の関係を下表に示します。

表 1.66 動作モード、オプションと転送開始条件一覧

カウントモード・ReqAssertCountオプション使用時

条件	カウントモード (Count>0)			
	Count ≥ Req		Count < Req	
	Ready ≥ Req	Ready < Req	Ready ≥ Count	Ready < Count
XDREQ	アサート	ネゲート	アサート	ネゲート
転送可能数	Req	–	Count	–

フリーランモード・ReqAssertCountオプション使用時

条件	フリーランモード	
	–	
	Ready ≥ Req	Ready < Req
XDREQ	アサート	ネゲート
転送可能数	Req	–

カウントモード・ReqAssertCountオプション未使用時

条件	カウントモード (Count>0)		
	Count ≥ Ready		Count < Ready
	Ready ≥ 2	Ready < 2	Ready ≥ Count
XDREQ	アサート	ネゲート	アサート
転送可能数	Ready (Readyが奇数の場合はReady-1)	–	Count

フリーランモード・ReqAssertCountオプション未使用時

条件	フリーランモード	
	–	
	Ready ≥ 2	Ready < 2
XDREQ	アサート	ネゲート
転送可能数	Ready (Readyが奇数の場合はReady-1)	–

※ 表中の Req は DMA_Config.ReqAssertCount の設定値を、Ready は FIFO の空き領域／データ数を、Count は DMA_Count_HH,HL,LH,LL の値を指しています。

1.6.3.2.2 端子設定

ChipConfig レジスタの設定により、XDREQx{x=1,0}の論理レベル、XDACKx{x=1,0}の論理レベルを設定する事が出来ます。以下の説明では特に断りの無い限り、XDREQ,XDACK 共に負論理で記述しております。

1.6.3.2.3 カウントモード(ライト)

【動作開始】

DMA_Count_HH,HL,LH,LL レジスタにカウント値を設定した後、DMA_Control.DMA_Go ビットに“1”をセットして下さい。内部 FIFO に書き込み可能な空き領域(DMA_Ready)が 2 バイト以上あり、且つ残りカウントがある場合に XDREQ をアサートし DMA 転送が可能になります。FIFO に残り 1 バイトの空き領域しかない場合は、カウントモードに設定され、且つ残りカウント数が“1”の時のみ XDREQ がアサートされます。

奇数バイトの書き込みを行って FIFO にバイト境界が発生した場合には、USB 等からデータが転送された後に、FIFO クリアを行ってバイト境界を解消し、次のライト動作を開始して下さい。例えば DMA から 31 バイト毎に書き込みを行い、USB から 31 バイト毎にデータを転送したい場合、(1)DMA カウント値を 31 に設定し 31 バイトデータを書き込む、(2)USB に 31 バイト分のデータが転送されるのを待つ、(3)USB から 31 バイト分のデータが転送されたのを確認した後 FIFO クリアする、以上の動作を繰り返し行って下さい。

動作停止となるまで DMA_Control.DMA_Running ビットに“1”が読めます。

【動作停止】

動作停止条件は以下の 2 つです。

- DMA_Count_HH,HL,LH,LL レジスタに設定したカウント数分の DMA 転送が完了
- DMA_Control.DMA_Stop ビットに“1”をライト

DMA 動作停止時に CPU_IntStat.DMA_Cmp ビットがセットされます。

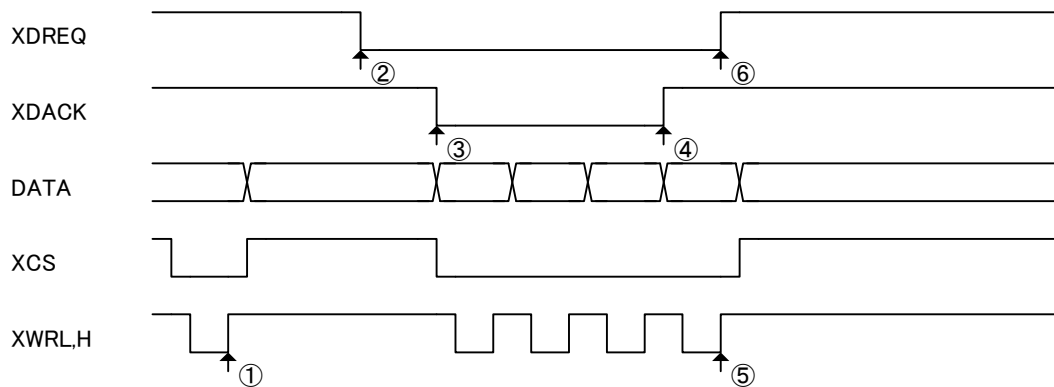
DMA_Count_HH,HL,LH,LL レジスタによる転送停止時は最終アクセスのストローブアサート期間に XDREQ をネゲートします。

DMA_Stop ビットによる転送停止時は、同期レジスタアクセスのライトタイミングでチップ内部動作を停止し、XDREQ をネゲートします。DMA_Stop ビットによる DMA の停止を行う場合は、CPU 側の DMAC(マスタ)を先に停止して下さい。

カウントモードで転送を開始し、設定されたカウント数の転送完了前に DMA_Control.DMA_Stop ビットで転送を停止した場合の動作タイミングを図 1.81 に示します。

1. 機能説明

ex1.【転送開始条件】カウント(8バイト) < FIFOの空き容量(16バイト) 【転送停止条件】DMA_Stop

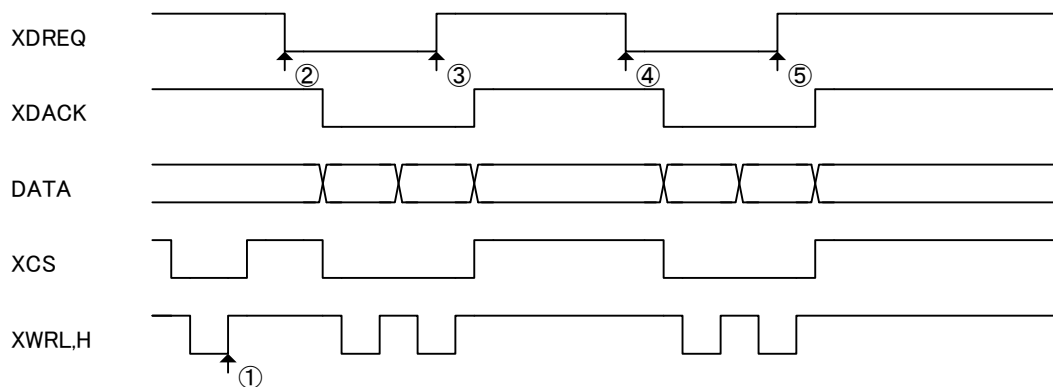


- ① DMA_Control.DMA_Goビットへの"1"書き込みによりDMA回路動作開始
- ② USB等からデータが転送される事によりFIFOに空き領域(DMA_Ready)が出来
DMA_Readyを受けてXDREQをアサート
- ③ XDACKがアサートされDMA転送開始
- ④ カウントモードの転送数完了前に、マスタ側を停止、XDACKをネゲート
- ⑤ DMA_Control.DMA_Stopビットへの"1"書き込みによりDMA回路停止
- ⑥ DMA回路停止を受けてXDREQネゲート

図 1.81 カウントモードライトタイミング 1

カウントモードで転送を開始し、設定されたカウント数分の転送を完了し DMA 転送が終了する場合の動作タイミングを図 1.82 に示します。

ex2.【転送開始条件】カウント(8バイト) > FIFOの空き容量(4バイト) 【転送停止条件】カウント0



- ① DMA_Control.DMA_Goビットへの"1"書き込みによりDMA回路動作開始
- ② USB等からデータが転送される事によりFIFOに空き領域(DMA_Ready)が出来
DMA_Readyを受けてXDREQをアサート
- ③ DMA_ReadyがなくなるタイミングでXDREQをネゲート
- ④ USB等からデータが転送される事によりFIFOに空き領域(DMA_Ready)が出来
DMA_Readyを受けてXDREQをアサート
- ⑤ DMA_Countの最終データタイミングでXDREQをネゲート
DMA_Count数分の転送終了によりDMA回路停止

図 1.82 カウントモードライトタイミング 2

1.6.3.2.4 カウントモード(リード)

【動作開始】

DMA_Count_HH,HL,LH,LL レジスタにカウント値を設定した後、DMA_Control.DMA_Go ビットに“1”をセットして下さい。内部 FIFO に読み出し可能なデータが 2 バイト以上あり、且つ残りカウントがある場合に、外部からのリードに対応可能になると XDREQ をアサートします。FIFO に残り 1 バイトしかデータがない場合はカウントモードに設定され、且つ残りカウント数が“1”の時のみ XDREQ がアサートされます。

デバイス動作時を例にしますと、カウントモードのリード動作では DMA_Count_HH,HL,LH,LL レジスタに設定されたカウント数以上のデータが、その DMA が接続されているエンドポイントの FIFO に貯まった時点で、ForceNAK ビットを自動的に“1”にセットし NAK 応答します。また、USB からショートパケットを受信した場合にも DisAF_NAK_Short ビットをセットしていなければ、該当するエンドポイントの ForceNAK ビットを自動的に“1”にセットし NAK 応答します。

奇数バイトのリードを行ってバイト境界が発生した場合には、FIFO クリアを行ってバイト境界を解消してから次の転送を行って下さい。例えば、USB から 31 バイト毎にデータが転送され、DMA から 31 バイト毎にデータを読み出す場合、(1)USB から 31 バイトのデータを受信する(この時点で ForceNAK がセットされ該当するエンドポイントは NAK 応答する)、(2)DMA から 31 バイトのデータを読み出す、(3)FIFO クリアした後 ForceNAK をクリアし USB からの転送を受信可能にする、以上の動作を繰り返して下さい。

動作停止となるまで DMA_Control.DMA_Running ビットに“1”が読めます。

【動作停止】

動作停止条件は以下の 2 つです。

- DMA_Count_HH,HL,LH,LL レジスタに設定したカウント数分の DMA 転送が完了
- DMA_Control.DMA_Stop ビットに“1”をライト

DMA_Count_HH,HL,LH,LL レジスタによる転送停止は最終アクセスのストローブ信号アサート期間に XDREQ をネゲートします。

DMA_Stop ビットによる転送停止は、同期レジスタアクセスのライトタイミングでチップ内部動作を停止し、XDREQ をネゲートします。DMA_Stop ビットによる DMA の停止を行う場合は、CPU 側の DMAC(マスタ)を先に停止して下さい。

カウントモードで転送を開始し、設定されたカウント数分の転送を完了して DMA 転送が終了する場合の動作タイミングを図 1.83 に示します。

1. 機能説明

ex.【転送開始】カウント(8バイト)>FIFOのデータ(4バイト) 【転送停止】カウント0

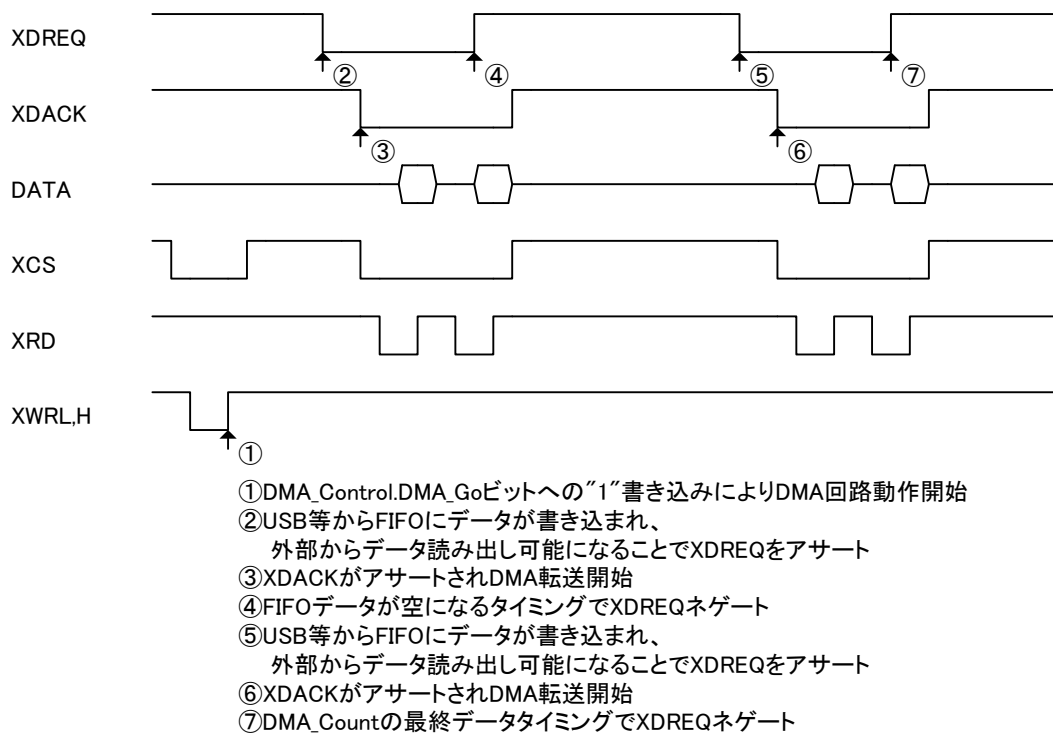


図 1.83 カウントモードリードタイミング

1.6.3.2.5 フリーランモード(ライト)

【動作開始】

DMA_Config.FreeRun ビットをセットした後、DMA_Control.DMA_Go ビットに“1”を書き込んで下さい。内部 FIFO に書き込み可能な空き領域が 2 バイト以上ある場合に XDREQ をアサートし DMA 転送が可能になります。FIFO に残り 1 バイトの空き領域しかない場合はフリーランモードでは XDREQ がアサートされません。転送を行う場合は「1.6.3.2.3 カウントモード(ライト)」をご参照ください。

動作停止となるまで DMA_Control.DMA_Running ビットに“1”が読めます。

【動作停止】

動作停止条件は以下です。

- DMA_Control.DMA_Stop ビットに“1”をライト

DMA_Stop ビットによる転送停止は、同期レジスタアクセスのライトタイミングでチップ内部動作を停止し、XDREQ をネゲートします。DMA_Stop ビットによる DMA の停止を行う場合は、CPU 側の DMAC(マスタ)を先に停止して下さい。

フリーランモードの DMA 転送中に DMA_Count_HH,HL,LH,LL レジスタの値がオーバーフローすると、CPU_IntStat.DMA_Countup ビットがセットされます。この場合も DMA 転送は継続され、DMA_Count_HH,HL,LH,LL も継続してカウントされます。

動作タイミングは DMA_Count_HH,HL,LH,LL による制限が無い事を除き、カウントモードと同等です。

1.6.3.2.6 フリーランモード(リード)

【動作開始】

DMA_Config.FreeRun ビットをセットした後、DMA_Control.DMA_Go ビットに“1”をセットして下さい。内部 FIFO に読み出し可能なデータが 2 バイト以上あり、外部からのリードに対応可能になると XDREQ をアサートします。FIFO に残り 1 バイトしか有効なデータがない場合は DMA 動作を開始しません。転送を行う場合は「1.6.3.2.4 カウントモード(リード)」をご参照ください。

動作停止となるまで DMA_Control.DMA_Running ビットに“1”が読めます。

【動作停止】

動作停止条件は以下です。

- DMA_Control.DMA_Stop ビットに“1”をライト

DMA_Stop ビットによる転送停止は、同期レジスタアクセスのライトタイミングでチップ内部動作を停止し、XDREQ をネゲートします。DMA_Stop ビットによる DMA の停止を行う場合は、CPU 側の DMAC(マスタ)を先に停止して下さい。

フリーランモードの DMA 転送中に DMA_Count_HH,HL,LH,LL レジスタの値がオーバーフローすると、CPU_IntStat.DMA_Countup ビットがセットされます。この場合も DMA 転送は継続され、DMA_Count_HH,HL,LH,LL も継続してカウントされます。

動作タイミングは DMA_Count_HH,HL,LH,LL による制限が無い事を除き、カウントモードと同等です。

1.6.3.2.7 REQアサートカウントオプション(ライト)

【動作開始】

DMA_Config.ReqAssertCount[1:0] ビットでアサートカウント数を設定した後、DMA_Control.DMA_Go ビットに“1”をセットして下さい。内部 FIFO に設定されたアサートカウント数以上の書き込み可能な空き領域がある場合に XDREQ をアサートし DMA 転送が可能になります。従って、一旦 XDREQ がアサートされると、設定されたアサートカウント数分の転送が保証されることになります。ただし、空き領域がアサートカウント数に満たない場合も、カウントモードに設定され、且つ残りカウント数以上の空き領域がある場合は XDREQ をアサートします。この場合、転送可能数は、残りカウント数となります。

このモードでは、ReqAssertCount[1:0] ビットに設定された転送数毎に一旦 XDREQ がネゲートされます。

動作停止となるまで DMA_Control.DMA_Running ビットに“1”が読めます。

【動作停止】

動作停止条件は「1.6.3.2.3 カウントモード(ライト)」、「1.6.3.2.5 フリーランモード(ライト)」をご参照ください。

1. 機能説明

ex.【転送開始】REQアサートカウント(8beat: 16byte)

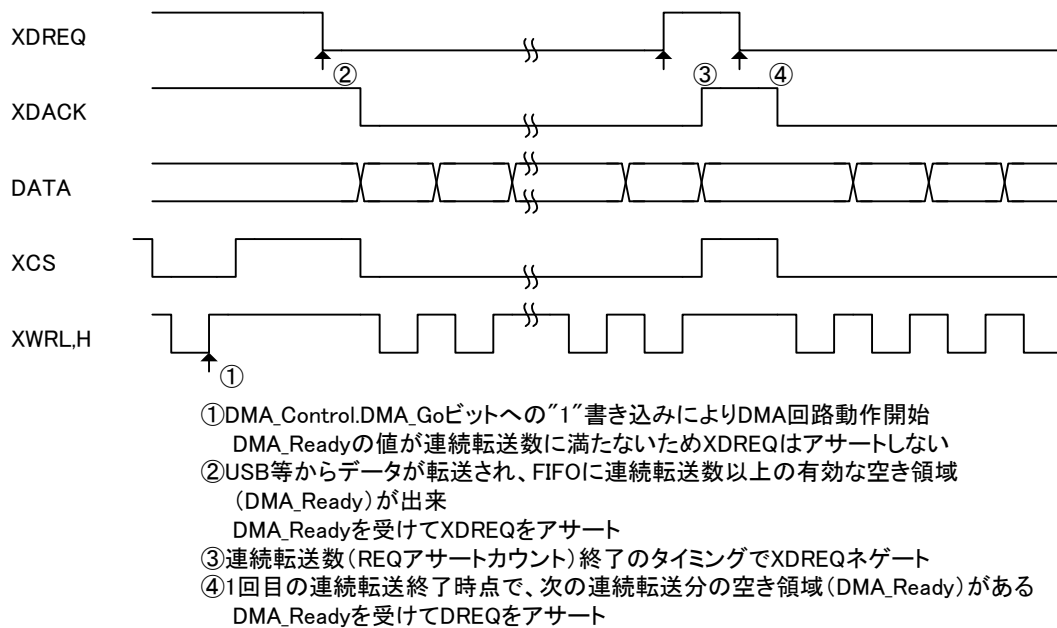


図 1.84 REQ アサートカウントオプションライトタイミング

1.6.3.2.8 REQアサートカウントオプション(リード)

【動作開始】

DMA_Config.ReqAssertCount[[1:0] ビットでアサートカウント数を設定した後、DMA_Control.DMA_Go ビットに“1”をセットして下さい。内部 FIFO に設定されたアサートカウント数以上の読み出し可能なデータがあり外部からのリードに対応可能になると XDREQ をアサートし DMA 転送が可能になります。従って、一旦 XDREQ がアサートされると、設定されたアサートカウント数分の転送が保証されることになります。ただし、FIFO のデータが REQ アサートカウント数に満たない場合も、カウントモードに設定され且つ残りカウント数以上のデータがある場合は、XDREQ をアサートします。この場合、転送可能数は、残りカウント数となります。

このモードでは、ReqAssertCount[1:0]ビットに設定された転送数毎に一旦 XDREQ がネゲートされます。

動作停止となるまで DMA_Control.DMA_Running ビットに“1”が読めます。

【動作停止】

動作停止条件は「1.6.3.2.4 カウントモード(リード)」、「1.6.3.2.6 フリーランモード(リード)」をご参照ください。

動作タイミングは図 1.83、図 1.84 をご参照下さい。

1.6.3.2.9 DMAのFIFOアクセス端数処理

「1.6.3.1.5 FIFO アクセスの端数処理」をご参照ください。DMA にはバイト読み出し／バイト書き込みの口はございませんのでご注意下さい。

1.6.3.2.10 DMAのEdge Sense モード

DMA_EdgeMode.EdgeSenseEn ビットに“1”をセットしてください。本ビットを設定後、1 ワード転送する毎に IC の XDREQ 信号がネゲートされます。すなわち、XDREQ のエッジをセンスして動作する DMA 転送に対応したモードです。本モードとカウントモード/フリーランモードの共用が可能です。但し、Req アサートカウントオプションと共用した場合は、DMA_Config.ReqAssertCount[1:0]ビットの設定値にかかわらず、1 ワードごとに XDREQ 信号がネゲートされます。

なお、DMA_EdgeMode.NgtCount[3:0]ビットにて、XDREQ の最小ネゲート期間を変更することが可能です。デフォルト値 0x0 が最も短い期間となっており、1 加算するたびに、内部クロックの 1 クロック（約 16.6ns）程度、最小ネゲート期間が延長されます。最小ネゲート期間は以下の式で表されます。

$$\text{XDREQ 最小ネゲート期間} = (\text{DMA_EdgeMode.NgtCount} + 3) \times 16.6 \text{ [ns]}$$

2. レジスタマップ

2. レジスタマップ

2.1 レジスタ構成

2.2 初期レジスタマップ

本 LSI は、ハードリセットされてから、本「初期レジスタ」が設定されてその設定が有効にされるまでの間(未初期化期間)は、本「初期レジスタ」のみがアクセス可能です。詳細は「1.6.2 CPUIF のモード設定」を参照して下さい。

Word Addr.	Register Name	R/W	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x07E	CPUIF_MODE	W	0xFFFF						CPU_Endian	BusMode	
									CPU_Endian	BusMode	

未初期化期間において、本 LSI へのライトアクセスは、全てこのレジスタへのライトアクセスと見なして処理します。初期化期間終了後は、本レジスタへのアクセスは無視されます。

2.3 レジスタマップ

レジスタマップを、「デバイス／ホスト共通レジスタ」、「デバイスレジスタ」、「ホストレジスタ」に分類して示します。

なお、下表の Reset 値は、リセット後のレジスタ値ですが、各レジスタの Reset 後のリード値は、IC が ACTIVE 時と SLEEP 時の場合で異なります。ACTIVE 時は全てのレジスタビットについて、下表の Reset 値と同じ値でリードされます。SLEEP 時については、**太字斜体**で示すレジスタビットは下表の Reset 値と同じ値でリードされますが、それ以外のレジスタビットは **0b0** がリードされます。本 IC の Reset 解除直後は SLEEP 状態となっておりますので、下表と異なる Reset 値が読み出されるレジスタにご注意ください。

2.3.1 デバイス／ホスト共通レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x000	MainIntStat	R/(W)	0x00	USB_DeviceIntStat	USB_HostIntStat	CPU_IntStat	FIFO_IntStat				FinishedPM
0x001	USB_DeviceIntStat	R/(W)	0x00	VBUS_Changed		D_SIE_IntStat	D_BulkIntStat	RcvEP0SETUP		D_EP0IntStat	D_EPIntStat
0x002	USB_HostIntStat	R/(W)	0x00	VBUS_Err	LineStateChanged	H_SIE_IntStat_1	H_SIE_IntStat_0	H_FrameIntStat		H_CH0IntStat	H_CHIntStat
0x003	CPU_IntStat	R/(W)	0x00	RAM_RdCmp						DMA_Countup	DMA_Cmp
0x004	FIFO_IntStat	R/(W)	0x00	FIFO_DMA_Cmp					FIFO_NotEmpty	FIFO_Full	FIFO_Empty
0x005			0xFF								
0x006											
0x007			0xFF								
0x008	MainIntEnb	R/W	0x00	EnUSB_DeviceIntStat	EnUSB_HostIntStat	EnCPU_IntStat	EnFIFO_IntStat				EnFinishedPM
0x009	USB_DeviceIntEnb	R/W	0x00	EnVBUS_Changed		EnD_SIE_IntStat	EnD_BulkIntStat	EnRcvEP0SETUP		EnD_EP0IntStat	EnD_EPIntStat
0x00A	USB_HostIntEnb	R/W	0x00	EnVBUS_Err	EnLineStateChanged	EnH_SIE_IntStat_1	EnH_SIE_IntStat_0	EnH_FrameIntStat		EnH_CH0IntStat	EnH_CHIntStat
0x00B	CPU_IntEnb	R/W	0x00	EnRAM_RdCmp						EnDMA_Countup	EnDMA_Cmp
0x00C	FIFO_IntEnb	R/W	0x00	EnFIFO_DMA_Cmp					EnFIFO_NotEmpty	EnFIFO_Full	EnFIFO_Empty
0x00D											
0x00E			0xFF								
0x00F			0xFF								

0x010	RevisionNum	R	0x08	RevisionNumber							
0x011	ChipReset	R/W	0x80	ResetMTM							AllReset
0x012	PM_Control	R/W	0x00	GoSLEEP	GoACTIVE	GoCPU_Cut				PM_State[1:0]	
0x013			0xFF								
0x014	WakeupTim_H	R/W	0x00	WakeupTim [15:8]							
0x015	WakeupTim_L	R/W	0x00	WakeupTim [7:0]							
0x016	H_USB_Control	R/W	0x00	VBUS_Enb							
0x017	H_XcvrControl	R/W	0x91	TermSelect	RemoveRPD	XcvrSelect[1:0]				OpMode[1:0]	
0x018	D_USB_Status	R/W	0xFF	VBUS	FSxHS					LineState[1:0]	
0x019	H_USB_Status	R	0xFF	VBUS_State						LineState[1:0]	
0x01A			0xFF								
0x01B	MTM_Config	R/W	0x00			MTM_SlopeValue [1:0]				MTM_TermValue [1:0]	
0x01C			0xFF								
0x01D			0xFF								
0x01E			0xFF								
0x01F	HostDeviceSel	R/W	0x00								HOSTxDEVICE

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x020	FIFO_Rd_0	R	0xXX	FIFO_Rd_0[7:0]							
0x021	FIFO_Rd_1	R	0xXX	FIFO_Rd_1[7:0]							
0x022	FIFO_Wr_0	W	0xXX	FIFO_Wr_0[7:0]							
0x023	FIFO_Wr_1	W	0xXX	FIFO_Wr_1[7:0]							
0x024	FIFO_RdRemain	R	0x00	RdRemainValid			RdRemain[12:8]				
0x025	FIFO_RdRemain	R	0x00	RdRemain[7:0]							
0x026	FIFO_WrRemain	R	0x00				WrRemain[12:8]				
0x027	FIFO_WrRemain	R	0x00	WrRemain[7:0]							
0x028	FIFO_ByteRd	R	0xXX	FIFO_ByteRd[7:0]							
0x029			0xXX								
0x02A	FIFO_ByteWr	W	0xXX	FIFO_ByteWr[7:0]							
0x02B			0xXX								
0x02C	CacheRemain	R	0x00	WrFIFO_C_Remain	WrDOOR_C_Remain						
0x02D			0xXX								
0x02E			0xXX								
0x02F			0xXX								

0x030	RAM_RdAdrs_H	R/W	0x00				RAM_RdAdrs[12:8]					
0x031	RAM_RdAdrs_L	R/W	0x00	RAM_RdAdrs[7:2]								
0x032	RAM_RdControl	R/W	0x00	RAM_GoRdCBW_CSW	RAM_GoRd							
0x033			0xXX									
0x034			0xXX									
0x035	RAM_RdCount	R/W	0x00	RAM_RdCount[5:2]								
0x036			0xXX									
0x037			0xXX									
0x038	RAM_WrAdrs_H	R/W	0x00				RAM_WrAdrs[12:8]					
0x039	RAM_WrAdrs_L	R/W	0x00	RAM_WrAdrs[7:0]								
0x03A	RAM_WrDoor_0	W	0xXX	RAM_WrDoor_0[7:0]								
0x03B	RAM_WrDoor_1	W	0xXX	RAM_WrDoor_1[7:0]								
0x03C												
0x03D			0xXX									
0x03E			0xXX									
0x03F			0xXX									

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x040	RAM_Rd_00	R	0x00	RAM_Rd_00[7:0]							
0x041	RAM_Rd_01	R	0x00	RAM_Rd_01[7:0]							
0x042	RAM_Rd_02	R	0x00	RAM_Rd_02[7:0]							
0x043	RAM_Rd_03	R	0x00	RAM_Rd_03[7:0]							
0x044	RAM_Rd_04	R	0x00	RAM_Rd_04[7:0]							
0x045	RAM_Rd_05	R	0x00	RAM_Rd_05[7:0]							
0x046	RAM_Rd_06	R	0x00	RAM_Rd_06[7:0]							
0x047	RAM_Rd_07	R	0x00	RAM_Rd_07[7:0]							
0x048	RAM_Rd_08	R	0x00	RAM_Rd_08[7:0]							
0x049	RAM_Rd_09	R	0x00	RAM_Rd_09[7:0]							
0x04A	RAM_Rd_0A	R	0x00	RAM_Rd_0A[7:0]							
0x04B	RAM_Rd_0B	R	0x00	RAM_Rd_0B[7:0]							
0x04C	RAM_Rd_0C	R	0x00	RAM_Rd_0C[7:0]							
0x04D	RAM_Rd_0D	R	0x00	RAM_Rd_0D[7:0]							
0x04E	RAM_Rd_0E	R	0x00	RAM_Rd_0E[7:0]							
0x04F	RAM_Rd_0F	R	0x00	RAM_Rd_0F[7:0]							

0x050	RAM_Rd_10	R	0x00	RAM_Rd_10[7:0]							
0x051	RAM_Rd_11	R	0x00	RAM_Rd_11[7:0]							
0x052	RAM_Rd_12	R	0x00	RAM_Rd_12[7:0]							
0x053	RAM_Rd_13	R	0x00	RAM_Rd_13[7:0]							
0x054	RAM_Rd_14	R	0x00	RAM_Rd_14[7:0]							
0x055	RAM_Rd_15	R	0x00	RAM_Rd_15[7:0]							
0x056	RAM_Rd_16	R	0x00	RAM_Rd_16[7:0]							
0x057	RAM_Rd_17	R	0x00	RAM_Rd_17[7:0]							
0x058	RAM_Rd_18	R	0x00	RAM_Rd_18[7:0]							
0x059	RAM_Rd_19	R	0x00	RAM_Rd_19[7:0]							
0x05A	RAM_Rd_1A	R	0x00	RAM_Rd_1A[7:0]							
0x05B	RAM_Rd_1B	R	0x00	RAM_Rd_1B[7:0]							
0x05C	RAM_Rd_1C	R	0x00	RAM_Rd_1C[7:0]							
0x05D	RAM_Rd_1D	R	0x00	RAM_Rd_1D[7:0]							
0x05E	RAM_Rd_1E	R	0x00	RAM_Rd_1E[7:0]							
0x05F	RAM_Rd_1F	R	0x00	RAM_Rd_1F[7:0]							

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0		
0x060			0xXX										
0x061	DMA_Config	R/W	0x00	FreeRun	DMA_Mode			ActiveDMA		ReqAssertCount [1:0]			
0x062	DMA_Control	R/W	0x00	DMA_Running			CounterClr	Dir		DMA_Stop	DMA_Go		
0x063													
0x064	DMA_Remain_H	R	0x00				DMA_Remain[12:8]						
0x065	DMA_Remain_L	R	0x00	DMA_Remain [7:0]									
0x066			0xXX										
0x067	DMA_EdgeMode	R/W	0x00	EdgeSenseEn				NegControl[3:0]					
0x068	DMA_Count_HH	R/W	0x00	DMA_Count [31:24]									
0x069	DMA_Count_HL	R/W	0x00	DMA_Count [23:16]									
0x06A	DMA_Count_LH	R/W	0x00	DMA_Count [15:8]									
0x06B	DMA_Count_LL	R/W	0x00	DMA_Count [7:0]									
0x06C	DMA_RdData_0	R	0xXX	DMA_RdData_0[7:0]									
0x06D	DMA_RdData_1	R	0xXX	DMA_RdData_1[7:0]									
0x06E	DMA_WrData_0	W	0xXX	DMA_WrData_0[7:0]									
0x06F	DMA_WrData_1	W	0xXX	DMA_WrData_1[7:0]									

0x070			0xXX								
0x071	ModeProtect	R/W	0x56	Protected[7:0](56 以外を書くとはプロテクト、0x56 で解除)							
0x072			0xXX								
0x073	ClkSelect	R/W	0x00	ClkSource						ClkFreq[1:0]	
0x074			0xXX								
0x075	CPU_Config	R/W	0x01	IntLevel	IntMode	DREQ_Level	DACK_Level	CS_Mode	CPU_Endian(R)	BusMode(R)	Inialized(R)
0x076			0xXX								
0x077			0xXX								
0x078			0xXX								
0x079			0xXX								
0x07A			0xXX								
0x07B			0xXX								
0x07C			0x00								
0x07D			0x00								
0x07E	CPUIF_MODE	W	0x00						CPU_Endian	BusMode	
0x07F	CPUIF_MODE	W	0x00						CPU_Endian	BusMode	

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x080	AREA0StartAdrs_	R/W	0x00			StartAdrs[12:8]					
0x081	AREA0StartAdrs_	R/W	0x00	StartAdrs[7:2]							
0x082	AREA0EndAdrs_	R/W	0x00			EndAdrs[12:8]					
0x083	AREA0EndAdrs_	R/W	0x00	EndAdrs[7:2]							
0x084	AREA1StartAdrs_	R/W	0x00			StartAdrs[12:8]					
0x085	AREA1StartAdrs_	R/W	0x00	StartAdrs[7:2]							
0x086	AREA1EndAdrs_	R/W	0x00			EndAdrs[12:8]					
0x087	AREA1EndAdrs_	R/W	0x00	EndAdrs[7:2]							
0x088	AREA2StartAdrs_	R/W	0x00			StartAdrs[12:8]					
0x089	AREA2StartAdrs_	R/W	0x00	StartAdrs[7:2]							
0x08A	AREA2EndAdrs_	R/W	0x00			EndAdrs[12:8]					
0x08B	AREA2EndAdrs_	R/W	0x00	EndAdrs[7:2]							
0x08C	AREA3StartAdrs_	R/W	0x00			StartAdrs[12:8]					
0x08D	AREA3StartAdrs_	R/W	0x00	StartAdrs[7:2]							
0x08E	AREA3EndAdrs_	R/W	0x00			EndAdrs[12:8]					
0x08F	AREA3EndAdrs_	R/W	0x00	EndAdrs[7:2]							
0x090	AREA4StartAdrs_	R/W	0x00			StartAdrs[12:8]					
0x091	AREA4StartAdrs_	R/W	0x00	StartAdrs[7:2]							
0x092	AREA4EndAdrs_	R/W	0x00			EndAdrs[12:8]					
0x093	AREA4EndAdrs_	R/W	0x00	EndAdrs[7:2]							
0x094	AREA5StartAdrs_	R/W	0x00			StartAdrs[12:8]					
0x095	AREA5StartAdrs_	R/W	0x00	StartAdrs[7:2]							
0x096	AREA5EndAdrs_	R/W	0x00			EndAdrs[12:8]					
0x097	AREA5EndAdrs_	R/W	0x00	EndAdrs[7:2]							
0x098			0xXX								
0x099			0xXX								
0x09A			0xXX								
0x09B			0xXX								
0x09C			0xXX								
0x09D			0xXX								
0x09E			0xXX								
0x09F	AREAnFIFO_Clr	W	0xXX			ClrAREA5	ClrAREA4	ClrAREA3	ClrAREA2	ClrAREA1	ClrAREA0

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0A0	AREA0Join_0	R/W	0x00	JoinFIFO_Stat					JoinDMA	JoinCPU_Rd	JoinCPU_Wr
0x0A1	AREA0Join_1	R/W	0x00			JoinEPeCHe	JoinEPdCHd	JoinEPcCHc	JoinEPbCHb	JoinEPaCHa	JoinEP0CH0
0x0A2	AREA1Join_0	R/W	0x00	JoinFIFO_Stat					JoinDMA	JoinCPU_Rd	JoinCPU_Wr
0x0A3	AREA1Join_1	R/W	0x00			JoinEPeCHe	JoinEPdCHd	JoinEPcCHc	JoinEPbCHb	JoinEPaCHa	JoinEP0CH0
0x0A4	AREA2Join_0	R/W	0x00	JoinFIFO_Stat					JoinDMA	JoinCPU_Rd	JoinCPU_Wr
0x0A5	AREA2Join_1	R/W	0x00			JoinEPeCHe	JoinEPdCHd	JoinEPcCHc	JoinEPbCHb	JoinEPaCHa	JoinEP0CH0
0x0A6	AREA3Join_0	R/W	0x00	JoinFIFO_Stat					JoinDMA	JoinCPU_Rd	JoinCPU_Wr
0x0A7	AREA3Join_1	R/W	0x00			JoinEPeCHe	JoinEPdCHd	JoinEPcCHc	JoinEPbCHb	JoinEPaCHa	JoinEP0CH0
0x0A8	AREA4Join_0	R/W	0x00	JoinFIFO_Stat					JoinDMA	JoinCPU_Rd	JoinCPU_Wr
0x0A9	AREA4Join_1	R/W	0x00			JoinEPeCHe	JoinEPdCHd	JoinEPcCHc	JoinEPbCHb	JoinEPaCHa	JoinEP0CH0
0x0AA	AREA5Join_0	R/W	0x00	JoinFIFO_Stat					JoinDMA	JoinCPU_Rd	JoinCPU_Wr
0x0AB	AREA5Join_1	R/W	0x00			JoinEPeCHe	JoinEPdCHd	JoinEPcCHc	JoinEPbCHb	JoinEPaCHa	JoinEP0CH0
0x0AC											
0x0AD											
0x0AE	ClrAREAnJoin_0	W	0x00	ClrJoinFIFO_Stat					ClrJoinDMA	ClrJoinCPU_Rd	ClrJoinCPU_Wr
0x0AF	ClrAREAnJoin_1	W	0x00			ClrJoinEPeCHe	ClrJoinEPdCHd	ClrJoinEPcCHc	ClrJoinEPbCHb	ClrJoinEPaCHa	ClrJoinEP0CH0

2.3.2 デバイスレジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0B0	<i>D_SIE_IntStat</i>	R/(W)	0x00		<i>NonJ</i>	RcvSOF	DetectRESET	DetectSUSPEND	ChirpCmp	RestoreCmp	SetAddressCmp
0x0B1			0xXX								
0x0B2		R/(W)	0x00								
0x0B3	<i>D_BulkIntStat</i>	R/(W)	0x00	CBW_Cmp	CBW_LengthErr	CBW_Err		CSW_Cmp	CSW_Err		
0x0B4	<i>D_EPrintStat</i>	R	0x00	D_AlarmIN_IntStat	D_AlarmOUT_IntStat		D_EPeIntStat	D_EPdIntStat	D_EPcIntStat	D_EPbIntStat	D_EPaIntStat
0x0B5	<i>D_EP0IntStat</i>	R/(W)	0x00	DescriptorCmp	OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0x0B6	<i>D_EPaIntStat</i>	R/(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0x0B7	<i>D_EPbIntStat</i>	R/(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0x0B8	<i>D_EPcIntStat</i>	R/(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0x0B9	<i>D_EPdIntStat</i>	R/(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0x0BA	<i>D_EPeIntStat</i>	R/(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0x0BB			0xXX								
0x0BC	<i>D_AlarmIN_IntStat_H</i>	R/(W)	0x00	AlarmEP15IN	AlarmEP14IN	AlarmEP13IN	AlarmEP12IN	AlarmEP11IN	AlarmEP10IN	AlarmEP9IN	AlarmEP8IN
0x0BD	<i>D_AlarmIN_IntStat_L</i>	R/(W)	0x00	AlarmEP7IN	AlarmEP6IN	AlarmEP5IN	AlarmEP4IN	AlarmEP3IN	AlarmEP2IN	AlarmEP1IN	
0x0BE	<i>D_AlarmOUT_IntStat_H</i>	R/(W)	0x00	AlarmEP15OUT	AlarmEP14OUT	AlarmEP13OUT	AlarmEP12OUT	AlarmEP11OUT	AlarmEP10OUT	AlarmEP9OUT	AlarmEP8OUT
0x0BF	<i>D_AlarmOUT_IntStat_L</i>	R/(W)	0x00	AlarmEP7OUT	AlarmEP6OUT	AlarmEP5OUT	AlarmEP4OUT	AlarmEP3OUT	AlarmEP2OUT	AlarmEP1OUT	

0x0C0	<i>D_SIE_IntEnb</i>	R/W	0x00		<i>EnNonJ</i>	EnRcvSOF	EnDetectRESET	EnDetectSUSPEND	EnChirpCmp	EnRestoreCmp	EnSetAddressCmp
0x0C1			0xXX								
0x0C2		R/W	0x00								
0x0C3	<i>D_BulkIntEnb</i>	R/W	0x00	EnCBW_Cmp	EnCBW_LengthErr	EnCBW_Err		EnCSW_Cmp	EnCSW_Err		
0x0C4	<i>D_EPrintEnb</i>	R/W	0x00	EnD_AlarmIN_IntStat	EnD_AlarmOUT_IntStat		EnD_EPeIntStat	EnD_EPdIntStat	EnD_EPcIntStat	EnD_EPbIntStat	EnD_EPaIntStat
0x0C5	<i>D_EP0IntEnb</i>	R/W	0x00	EnDescriptorCmp	EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0x0C6	<i>D_EPaIntEnb</i>	R/W	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0x0C7	<i>D_EPbIntEnb</i>	R/W	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0x0C8	<i>D_EPcIntEnb</i>	R/W	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0x0C9	<i>D_EPdIntEnb</i>	R/W	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0x0CA	<i>D_EPeIntEnb</i>	R/W	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0x0CB			0xXX								
0x0CC	<i>D_AlarmIN_IntEnb_H</i>	R/W	0x00	EnAlarmEP15IN	EnAlarmEP14IN	EnAlarmEP13IN	EnAlarmEP12IN	EnAlarmEP11IN	EnAlarmEP10IN	EnAlarmEP9IN	EnAlarmEP8IN
0x0CD	<i>D_AlarmIN_IntEnb_L</i>	R/W	0x00	EnAlarmEP7IN	EnAlarmEP6IN	EnAlarmEP5IN	EnAlarmEP4IN	EnAlarmEP3IN	EnAlarmEP2IN	EnAlarmEP1IN	
0x0CE	<i>D_AlarmOUT_IntEnb_H</i>	R/W	0x00	EnAlarmEP15OUT	EnAlarmEP14OUT	EnAlarmEP13OUT	EnAlarmEP12OUT	EnAlarmEP11OUT	EnAlarmEP10OUT	EnAlarmEP9OUT	EnAlarmEP8OUT
0x0CF	<i>D_AlarmOUT_IntEnb_L</i>	R/W	0x00	EnAlarmEP7OUT	EnAlarmEP6OUT	EnAlarmEP5OUT	EnAlarmEP4OUT	EnAlarmEP3OUT	EnAlarmEP2OUT	EnAlarmEP1OUT	

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0D0	D_NegoControl	R/W	0x00	DisBusDetect	EnAutoNego	InSUSPEND	DisableHS	SendWakeup	RestoreUSB	GoChirp	ActiveUSB
0x0D1			0xXX								
0x0D2			0xXX								
0x0D3	D_XcvtControl	R/W	0x41	TermSelect	XcvtSelect					OpMode [1:0]	
0x0D4	D_USB_Test	R/W	0x00	EnHS_Test				Test_SE0_NAK	Test_J	Test_K	Test_Packet
0x0D5			0xXX								
0x0D6	D_EPnControl	W	0xXX	AllForceNAK	EPnForceSTALL						
0x0D7			0xXX								
0x0D8	D_BulkOnlyControl	R/W	0x00	AutoForceNAK_CBW					GoCBW_Mode	GoCSW_Mode	
0x0D9		R/W	0x00				EPeBulkOnly	EPdBulkOnly	EPcBulkOnly	EPbBulkOnly	EPaBulkOnly
0x0DA			0xXX								
0x0DB			0xXX								
0x0DC			0xXX								
0x0DD			0xXX								
0x0DE			0xXX								
0x0DF			0xXX								

0x0E0	D_EP0SETUP_0	R	0x00	SETUP 0[7:0]								
0x0E1	D_EP0SETUP_1	R	0x00	SETUP 1[7:0]								
0x0E2	D_EP0SETUP_2	R	0x00	SETUP 2[7:0]								
0x0E3	D_EP0SETUP_3	R	0x00	SETUP 3[7:0]								
0x0E4	D_EP0SETUP_4	R	0x00	SETUP 4[7:0]								
0x0E5	D_EP0SETUP_5	R	0x00	SETUP 5[7:0]								
0x0E6	D_EP0SETUP_6	R	0x00	SETUP 6[7:0]								
0x0E7	D_EP0SETUP_7	R	0x00	SETUP 7[7:0]								
0x0E8	D_USB_Address	R/(W)	0x00	SetAddress	USB_Address [6:0]							
0x0E9			0xXX									
0x0EA	D_EP0SETUP_Control	R/W	0x00							ProtectEP0		
0x0EB			0xXX									
0x0EC			0xXX									
0x0ED			0xXX									
0x0EE	D_FrameNumber_H	R	0x80	Fn_Invalid					FrameNumber [10:8]			
0x0EF	D_FrameNumber_L	R	0x00	FrameNumber [7:0]								

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0F0	D_EP0MaxSize	R/W	0x00	EP0MaxSize[6:3]							
0x0F1	D_EP0Control	R/W	0x00	INxOUT							ReplyDescriptor
0x0F2	D_EP0ControlIN	R/W	0x00		EnShortPkt		ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x0F3	D_EP0ControlOUT	R/W	0x00	AutoForceNAK			ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x0F4			0xXX								
0x0F5			0xXX								
0x0F6			0xXX								
0x0F7			0x00								
0x0F8	D_EPaMaxSize_H	R/W	0x00	EPaMaxSize[10:8]							
0x0F9	D_EPaMaxSize_L	R/W	0x00	EPaMaxSize[7:0]							
0x0FA	D_EPaConfig	R/W	0x00	INxOUT	IntEP_Mode	ISO		EndpointNumber[3:0]			
0x0FB			0xXX								
0x0FC	D_EPaControl	R/W	0x00	AutoForceNAK	EnShortPkt	DisAF_NAK_Short	ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x0FD			0xXX								
0x0FE			0xXX								
0x0FF			0xXX								

0x100	D_EPbMaxSize_H	R/W	0x00	EPbMaxSize[10:8]							
0x101	D_EPbMaxSize_L	R/W	0x00	EPbMaxSize[7:0]							
0x102	D_EPbConfig	R/W	0x00	INxOUT	IntEP_Mode	ISO		EndpointNumber[3:0]			
0x103			0xXX								
0x104	D_EPbControl	R/W	0x00	AutoForceNAK	EnShortPkt	DisAF_NAK_Short	ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x105			0xXX								
0x106			0xXX								
0x107			0xXX								
0x108	D_EPcMaxSize_H	R/W	0x00	EPcMaxSize[10:8]							
0x109	D_EPcMaxSize_L	R/W	0x00	EPcMaxSize[7:0]							
0x10A	D_EPcConfig	R/W	0x00	INxOUT	IntEP_Mode	ISO		EndpointNumber[3:0]			
0x10B			0xXX								
0x10C	D_EPcControl	R/W	0x00	AutoForceNAK	EnShortPkt	DisAF_NAK_Short	ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x10D			0xXX								
0x10E			0xXX								
0x10F			0xXX								

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x110	D_EPdMaxSize_H	R/W	0x00						EPdMaxSize[10:8]		
0x111	D_EPdMaxSize_L	R/W	0x00	EPdMaxSize[7:0]							
0x112	D_EPdConfig	R/W	0x00	INxOUT	IntEP_Mode	ISO		EndpointNumber[3:0]			
0x113			0xXX								
0x114	D_EPdControl	R/W	0x00	AutoForceNAK	EnShortPkt	DisAF_NAK_Short	ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x115			0xXX								
0x116			0xXX								
0x117			0xXX								
0x118	D_EPeMaxSize_H	R/W	0x00						EPeMaxSize[10:8]		
0x119	D_EPeMaxSize_L	R/W	0x00	EPeMaxSize[7:0]							
0x11A	D_EPeConfig	R/W	0x00	INxOUT	IntEP_Mode	ISO		EndpointNumber[3:0]			
0x11B			0xXX								
0x11C	D_EPeControl	R/W	0x00	AutoForceNAK	EnShortPkt	DisAF_NAK_Short	ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x11D			0xXX								
0x11E			0xXX								
0x11F			0xXX								

0x120	D_DescAdrs_H	R/W	0x00				DescAdrs[12:8]				
0x121	D_DescAdrs_L	R/W	0x00	DescAdrs [7:0]							
0x122	D_DescSize_H	R/W	0x00						DescSize [9:8]		
0x123	D_DescSize_L	R/W	0x00	DescSize [7:0]							
0x124			0xXX								
0x125			0xXX								
0x126	D_EP_DMA_Ctrl	R/W	0xXX	FIFO_Running	AutoEnShort						
0x127			0xXX								
0x128	D_EnEP_IN_H	R/W	0x00	EnEP15IN	EnEP14IN	EnEP13IN	EnEP12IN	EnEP11IN	EnEP10IN	EnEP9IN	EnEP8IN
0x129	D_EnEP_IN_L	R/W	0x00	EnEP7IN	EnEP6IN	EnEP5IN	EnEP4IN	EnEP3IN	EnEP2IN	EnEP1IN	
0x12A	D_EnEP_OUT_H	R/W	0x00	EnEP15OUT	EnEP14OUT	EnEP13OUT	EnEP12OUT	EnEP11OUT	EnEP10OUT	EnEP9OUT	EnEP8OUT
0x12B	D_EnEP_OUT_L	R/W	0x00	EnEP7OUT	EnEP6OUT	EnEP5OUT	EnEP4OUT	EnEP3OUT	EnEP2OUT	EnEP1OUT	
0x12C	D_EnEP_IN_ISO_H	R/W	0x00	EnEP15IN_ISO	EnEP14IN_ISO	EnEP13IN_ISO	EnEP12IN_ISO	EnEP11IN_ISO	EnEP10IN_ISO	EnEP9IN_ISO	EnEP8IN_ISO
0x12D	D_EnEP_IN_ISO_L	R/W	0x00	EnEP7IN_ISO	EnEP6IN_ISO	EnEP5IN_ISO	EnEP4IN_ISO	EnEP3IN_ISO	EnEP2IN_ISO	EnEP1IN_ISO	
0x12E	D_EnEP_OUT_ISO_H	R/W	0x00	EnEP15OUT_ISO	EnEP14OUT_ISO	EnEP13OUT_ISO	EnEP12OUT_ISO	EnEP11OUT_ISO	EnEP10OUT_ISO	EnEP9OUT_ISO	EnEP8OUT_ISO
0x12F	D_EnEP_OUT_ISO_L	R/W	0x00	EnEP7OUT_ISO	EnEP6OUT_ISO	EnEP5OUT_ISO	EnEP4OUT_ISO	EnEP3OUT_ISO	EnEP2OUT_ISO	EnEP1OUT_ISO	

2.3.3 ホストレジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x140	H_SIE_IntStat_0	R(W)	0x00				DetectCon	DetectDiscon	DetectRmtWkup	DetectDevChirpOK	DetectDevChirpNG
0x141	H_SIE_IntStat_1	R(W)	0x00					DisabledCmp	ResumeCmp	SuspendCmp	ResetCmp
0x142		R(W)	0x00								
0x143	H_FrameIntStat	R(W)	0x00	TriggerFrame					PortErr	FrameNumOver	SOF
0x144	H_CHrIntStat	R	0x00				H_CHeIntStat	H_CHdIntStat	H_CHcIntStat	H_CHbIntStat	H_CHaIntStat
0x145	H_CH0IntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition			CTL_SupportCmp	CTL_SupportStop
0x146	H_CHaIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition			BO_SupportCmp	BO_SupportStop
0x147	H_CHbIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0x148	H_CHcIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0x149	H_CHdIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0x14A	H_CHeIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0x14B			0xFF								
0x14C			0xFF								
0x14D			0xFF								
0x14E			0xFF								
0x14F			0xFF								

0x150	H_SIE_IntEnb_0	R/W	0x00				EnDetectCon	EnDetectDiscon	EnDetectRmtWkup	EnDetectDevChirpOK	EnDetectDevChirpNG
0x151	H_SIE_IntEnb_1	R/W	0x00					EnDisabledCmp	EnResumeCmp	EnSuspendCmp	EnResetCmp
0x152		R/W	0x00								
0x153	H_FrameIntEnb	R/W	0x00	EnTriggerFrame					EnPortErr	EnFrameNumOver	EnSOF
0x154	H_CHrIntEnb	R/W	0x00				EnH_CHeIntStat	EnH_CHdIntStat	EnH_CHcIntStat	EnH_CHbIntStat	EnH_CHaIntStat
0x155	H_CH0IntEnb	R/W	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition			EnCTL_SupportCmp	EnCTL_SupportStop
0x156	H_CHaIntEnb	R/W	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition			EnBO_SupportCmp	EnBO_SupportStop
0x157	H_CHbIntEnb	R/W	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0x158	H_CHcIntEnb	R/W	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0x159	H_CHdIntEnb	R/W	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0x15A	H_CHeIntEnb	R/W	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0x15B			0xFF								
0x15C			0xFF								
0x15D			0xFF								
0x15E			0xFF								
0x15F			0xFF								

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x160	H_NegoControl_0	R/W	0x1X	AutoModeCancel	HostState[2:0]			AutoMode[3:0]			
0x161			0xXX								
0x162	H_NegoControl_1	R/W	0x10			PortSpeed[1:0]				DisChirpFinish	RmtWkupDetEnb
0x163			0xXX								
0x164	H_USB_Test	R/W	0x00	EnHS_Test			Test_Force_Enable	Test_SE0_NAK	Test_J	Test_K	Test_Packet
0x165			0xXX								
0x166			0xXX								
0x167			0xXX								
0x168			0xXX								
0x169			0xXX								
0x16A			0xXX								
0x16B			0xXX								
0x16C			0xXX								
0x16D			0xXX								
0x16E			0xXX								
0x16F			0xXX								

0x170	H_CH0SETUP_0	R/W	0x00	SETUP 0[7:0]								
0x171	H_CH0SETUP_1	R/W	0x00	SETUP 1[7:0]								
0x172	H_CH0SETUP_2	R/W	0x00	SETUP 2[7:0]								
0x173	H_CH0SETUP_3	R/W	0x00	SETUP 3[7:0]								
0x174	H_CH0SETUP_4	R/W	0x00	SETUP 4[7:0]								
0x175	H_CH0SETUP_5	R/W	0x00	SETUP 5[7:0]								
0x176	H_CH0SETUP_6	R/W	0x00	SETUP 6[7:0]								
0x177	H_CH0SETUP_7	R/W	0x00	SETUP 7[7:0]								
0x178			0xXX									
0x179			0xXX									
0x17A			0xXX									
0x17B			0xXX									
0x17C	Reserved		0xXX							Reserved		
0x17D			0xXX									
0x17E	H_FrameNumber_H	R	0x07					FrameNumber[10:8]				
0x17F	H_FrameNumber_L	R	0xFF	FrameNumber[7:0]								

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x180	H_CH0Config_0	R/W	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo
0x181	H_CH0Config_1	R/W	0x00	TID[1:0]							
0x182			0xXX								
0x183	H_CH0MaxPktSize	R/W	0x00	MaxPktSize[6:0]							
0x184			0xXX								
0x185			0xXX								
0x186	H_CH0TotalSize_H	R/W	0x00	TotalSize[15:8]							
0x187	H_CH0TotalSize_L	R/W	0x00	TotalSize[7:0]							
0x188	H_CH0HubAdrs	R/W	0x00	HubAdrs[3:0]					Port[2:0]		
0x189	H_CH0FuncAdrs	R/W	0x00	FuncAdrs[3:0]				EP_Number[3:0]			
0x18A			0xXX								
0x18B	H_CTL_SupportControl	R/W	0x00	CTL_SupportState[1:0]							CTL_SupportGo
0x18C			0xXX								
0x18D			0xXX								
0x18E	H_CH0ConditionCode	R	0x00	ConditonCode[2:0]							
0x18F			0xXX								

0x190	H_CHaConfig_0	R/W	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo
0x191	H_CHaConfig_1	R/W	0x00	TID[1:0]				AutoZeroLen			TotalSizeFree
0x192	H_CHaMaxPktSize_H	R/W	0x00					(MaxPktSize[10])	MaxPktSize[9:8]		
0x193	H_CHaMaxPktSize_L	R/W	0x00	MaxPktSize[7:0]							
0x194	H_CHaTotalSize_HH	R/W	0x00	TotalSize[31:24]							
0x195	H_CHaTotalSize_HL	R/W	0x00	TotalSize[23:16]							
0x196	H_CHaTotalSize_LH	R/W	0x00	TotalSize[15:8]							
0x197	H_CHaTotalSize_LL	R/W	0x00	TotalSize[7:0]							
0x198	H_CHaHubAdrs	R/W	0x00	HubAdrs[3:0]					Port[2:0]		
0x199	H_CHaFuncAdrs	R/W	0x00	FuncAdrs[3:0]				EP_Number[3:0]			
0x19A	H_CHaBO_SupportCtl	R/W	0x00			BO_TransportState[1:0]				BO_SupportGo	
0x19B	H_CHaBO_CSW_RcvSize	R	0x00					CSW_RcvDataSize[3:0]			
0x19C	H_CHaBO_OUT_EP_Ctl	R/W	0x00				OUT_Toggle	OUT_EP_Number[3:0]			
0x19D	H_CHaBO_IN_EP_Ctl	R/W	0x00				IN_Toggle	IN_EP_Number[3:0]			
0x19E	H_CHaConditionCode	R	0x00	ConditonCode[2:0]							
0x19F			0xXX								

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0		
0x1A0	H_CHbConfig_0	R/W	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo		
0x1A1	H_CHbConfig_1	R/W	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen	Audio441	TotalSizeFree[1]	TotalSizeFree[0]		
0x1A2	H_CHbMaxPktSize_H	R/W	0x00						(MaxPktSize[10])	MaxPktSize[9:8]			
0x1A3	H_CHbMaxPktSize_L	R/W	0x00	MaxPktSize[7:0]									
0x1A4	H_CHbTotalSize_HH	R/W	0x00	TotalSize[31:24]									
0x1A5	H_CHbTotalSize_HL	R/W	0x00	TotalSize[23:16]									
0x1A6	H_CHbTotalSize_LH	R/W	0x00	TotalSize[15:8]									
0x1A7	H_CHbTotalSize_LL	R/W	0x00	TotalSize[7:0]									
0x1A8	H_CHbHubAdrs	R/W	0x00	HubAdrs[3:0]					Port[2:0]				
0x1A9	H_CHbFuncAdrs	R/W	0x00	FuncAdrs[3:0]					EP_Number[3:0]				
0x1AA	H_CHbInterval_H	R/W	0x00						Interval[10:8]				
0x1AB	H_CHbInterval_L	R/W	0x00	Interval[7:0]									
0x1AC	H_CHbTranPause	R/W	0x00							EnTranPause	TranPause		
0x1AD			0xFF										
0x1AE	H_CHbConditionCode	R	0x00		ConditonCode[2:0]								
0x1AF			0xFF										

0x1B0	H_CHcConfig_0	R/W	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo
0x1B1	H_CHcConfig_1	R/W	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen	Audio441	TotalSizeFree[1]	TotalSizeFree[0]
0x1B2	H_CHcMaxPktSize_H	R/W	0x00					(MaxPktSize[10])	MaxPktSize[9:8]		
0x1B3	H_CHcMaxPktSize_L	R/W	0x00	MaxPktSize[7:0]							
0x1B4	H_CHcTotalSize_HH	R/W	0x00	TotalSize[31:24]							
0x1B5	H_CHcTotalSize_HL	R/W	0x00	TotalSize[23:16]							
0x1B6	H_CHcTotalSize_LH	R/W	0x00	TotalSize[15:8]							
0x1B7	H_CHcTotalSize_LL	R/W	0x00	TotalSize[7:0]							
0x1B8	H_CHcHubAdrs	R/W	0x00	HubAdrs[3:0]					Port[2:0]		
0x1B9	H_CHcFuncAdrs	R/W	0x00	FuncAdrs[3:0]				EP_Number[3:0]			
0x1BA	H_CHcInterval_H	R/W	0x00					Interval[10:8]			
0x1BB	H_CHcInterval_L	R/W	0x00	Interval[7:0]							
0x1BC	H_CHcTranPause	R/W	0x00						EnTranPause	TranPause	
0x1BD			0xXX								
0x1BE	H_CHcConditionCode	R	0x00	ConditonCode[2:0]							
0x1BF			0xXX								

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0		
0x1C0	H_CHdConfig_0	R/W	0x00	ACK_Cnt[3:0]					SpeedMode[1:0]		Toggle	TranGo	
0x1C1	H_CHdConfig_1	R/W	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen	Audio441	TotalSizeFree[1]	TotalSizeFree[0]		
0x1C2	H_CHdMaxPktSize_H	R/W	0x00						(MaxPktSize[10])	MaxPktSize[9:8]			
0x1C3	H_CHdMaxPktSize_L	R/W	0x00	MaxPktSize[7:0]									
0x1C4	H_CHdTotalSize_HH	R/W	0x00	TotalSize[31:24]									
0x1C5	H_CHdTotalSize_HL	R/W	0x00	TotalSize[23:16]									
0x1C6	H_CHdTotalSize_LH	R/W	0x00	TotalSize[15:8]									
0x1C7	H_CHdTotalSize_LL	R/W	0x00	TotalSize[7:0]									
0x1C8	H_CHdHubAdrs	R/W	0x00	HubAdrs[3:0]						Port[2:0]			
0x1C9	H_CHdFuncAdrs	R/W	0x00	FuncAdrs[3:0]					EP_Number[3:0]				
0x1CA	H_CHdInterval_H	R/W	0x00						Interval[10:8]				
0x1CB	H_CHdInterval_L	R/W	0x00	Interval[7:0]									
0x1CC	H_CHdTranPause	R/W	0x00							EnTranPause	TranPause		
0x1CD			0xFF										
0x1CE	H_CHdConditionCode	R	0x00		ConditonCode[2:0]								
0x1CF			0xFF										

0x1D0	H_CHeConfig_0	R/W	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo	
0x1D1	H_CHeConfig_1	R/W	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen	Audio441	TotalSizeFree[1]	TotalSizeFree[0]	
0x1D2	H_CHeMaxPktSize_H	R/W	0x00					(MaxPktSize[10])	MaxPktSize[9:8]			
0x1D3	H_CHeMaxPktSize_L	R/W	0x00	MaxPktSize[7:0]								
0x1D4	H_CHeTotalSize_HH	R/W	0x00	TotalSize[31:24]								
0x1D5	H_CHeTotalSize_HL	R/W	0x00	TotalSize[23:16]								
0x1D6	H_CHeTotalSize_LH	R/W	0x00	TotalSize[15:8]								
0x1D7	H_CHeTotalSize_LL	R/W	0x00	TotalSize[7:0]								
0x1D8	H_CHeHubAdrs	R/W	0x00	HubAdrs[3:0]					Port[2:0]			
0x1D9	H_CHeFuncAdrs	R/W	0x00	FuncAdrs[3:0]				EP_Number[3:0]				
0x1DA	H_CHeInterval_H	R/W	0x00						Interval[10:8]			
0x1DB	H_CHeInterval_L	R/W	0x00	Interval[7:0]								
0x1DC	H_CHeTranPause	R/W	0x00							EnTranPause	TranPause	
0x1DD			0xFF									
0x1DE	H_CHeConditionCode	R	0x00		ConditonCode[2:0]							
0x1DF			0xFF									

2. レジスタマップ

SLEEP 時にでも読み書きできるレジスタを**太字斜体**で示します。それ以外のレジスタは、ACTIVE 時に読み書きできます。

Address	Register Name	R/W	Reset	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x1F0											
0x1F1											
0x1F2	<i>H_TriggerFrameNum_H</i>	R/W	0x00	<i>UseTriggerFrame</i>					<i>TriggerFrameNum[10:8]</i>		
0x1F3	<i>H_TriggerFrameNum_L</i>	R/W	0x00	<i>TriggerFrameNum[7:0]</i>							
0x1F4											
0x1F5											
0x1F6											
0x1F7											
0x1F8											
0x1F9											
0x1FA											
0x1FB											
0x1FC											
0x1FD											
0x1FE											
0x1FF											

3. レジスタ詳細説明

3.1 デバイス／ホスト共通レジスタ詳細説明

3.1.1 000h MainIntStat (Main Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	000h	MainIntStat	R	7: USB_DeviceIntStat	0: None	1: USB Device Interrupts	00h
			R	6: USB_HostIntStat	0: None	1: USB Host Interrupts	
			R	5: CPU_IntStat	0: None	1: CPU Interrupts	
			R	4: FIFO_IntStat	0: None	1: FIFO Interrupts	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R (W)	0: FinishedPM	0: None	1: Detect FinishedPM	

本 LSI の割り込み要因を表示します。

このレジスタには割り込み要因を間接指示するビットと直接指示するビットがあります。割り込み要因を間接指示するビットは、それぞれに対応する割り込みステータスレジスタをリードすることにより、割り込み要因を直接指示するビットまで辿ることができます。割り込み要因を間接指示するビットは、リードオンリーであり、大元の割り込み要因を直接指示するビットをクリアすることにより、自動的にクリアされます。割り込み要因を直接指示しているビットは、書き込み可能であり、該当ビットに“1”を書き込むことにより、割り込み要因をクリアすることができます。MainIntEnb レジスタにより、対応するビットの割り込みがイネーブルにされている場合は、割り込み要因が“1”にセットされると XINT 端子がアサートされ、CPU に対して割り込みが発生します。該当する割り込み要因が全てクリアされると、XINT 端子がネゲートされます。

Bit7 **USB_DeviceIntStat**

割り込み要因を間接指示します。

USB_DeviceIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する USB_DeviceIntEnb レジスタのビットがイネーブルにされている時に“1”にセットされます。このビットは、SLEEP 中もリード有効です。

Bit6 **USB_HostIntStat**

割り込み要因を間接指示します。

USB_HostIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する USB_HostIntEnb レジスタのビットがイネーブルにされている時に“1”にセットされます。このビットは、SLEEP 中もリード有効です。

Bit5 **CPU_IntStat**

割り込み要因を間接指示します。

CPU_IntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する CPU_IntEnb レジスタのビットがイネーブルにされている時に“1”にセットされます。

Bit4 **FIFO_IntStat**

割り込み要因を間接指示します。

FIFO_IntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する FIFO_IntEnb レジスタのビットがイネーブルにされている時に“1”にセットされます。

Bit3-1 **Reserved**

3. レジスタ詳細説明

Bit0 ***FinishedPM***

割り込み要因を直接指示します。

PM_Control レジスタで、GoSLEEP,GoACTIVE を設定した場合、指示したそれぞれの状態に達したら、このビットは“1”にセットされます。このビットは SLEEP 中も有効です。

3.1.2 001h *USB_DeviceIntStat (USB Device Interrupt Status)*

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	001h	<i>USB_DeviceIntStat</i>	R (W)	7: <i>VBUS_Changed</i>	0: None	1: VBUS is Changed	00h
				6:	0:	1:	
			R	5: <i>D_SIE_IntStat</i>	0: None	1: SIE Interrupts	
			R	4: <i>D_BulkIntStat</i>	0: None	1: Bulk Interrupts	
			R (W)	3: <i>RcvEP0SETUP</i>	0: None	1: Receive EP0 SETUP	
				2:	0:	1:	
			R	1: <i>D_EP0IntStat</i>	0: None	1: EP0 Interrupts	
			R	0: <i>D_EPrintStat</i>	0: None	1: EPr Interrupts	

デバイス関連の割り込みを表示します。

このレジスタには割り込み要因を間接指示するビットと直接指示するビットがあります。割り込み要因を間接指示するビットは、それぞれに対応する割り込みステータスレジスタをリードすることにより、割り込み要因を直接指示するビットまで辿ることができます。割り込み要因を間接指示するビットは、リードオンリーであり、大元の割り込み要因を直接指示するビットをクリアすることにより、自動的にクリアされます。割り込み要因を直接指示しているビットは、書き込み可能であり、該当ビットに“1”を書き込むことにより、割り込み要因をクリアすることができます。

Bit7 *VBUS_Changed*

割り込み要因を直接指示します。

VBUS 端子の状態が変化した時に “1” にセットされます。

D_USB_Status レジスタの *VBUS* ビットによって *VBUS* の状態を確認して下さい。*VBUS* が“0”であれば、ケーブルが抜かれたことを示します。このビットは *SLEEP* 中も有効です。

Bit6 *Reserved***Bit5** *D_SIE_IntStat*

割り込み要因を間接指示します。

D_SIE_IntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する *D_SIE_IntEnb* レジスタのビットがイネーブルにされている時に“1”にセットされます。このビットは、*SLEEP* もリード有効です。

Bit4 *D_BulkIntStat*

割り込み要因を間接指示します。

D_BulkIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する *D_BulkIntEnb* レジスタのビットがイネーブルにされている時に “1” にセットされます。

Bit3 *RcvEP0SETUP*

割り込み要因を直接指示します。

コントロール転送のセットアップステージが終了し、受信したデータが *D_EP0SETUP_0*～*D_EP0SETUP_7* レジスタに格納されると“1”にセットされます。同時に *D_EP0ControlIN*, *D_EP0ControlOUT* レジスタの *ForceSTALL* ビットが “0” に *D_EP0ControlIN*, *D_EP0ControlOUT* レジスタの *ForceNAK* ビット、*ToggleStat* ビット、*D_SETUP_Control* レジスタの *ProtectEP0* ビットが“1”に、自動的に設定されます。*SetAddress()* リクエストに対しては、*AutoSetAddress* 機能が自動応答し、このステータスはセットされません。

Bit2 *Reserved*

3. レジスタ詳細説明

Bit1 D_EP0IntStat

割り込み要因を間接指示します。

D_EP0IntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する

D_EP0IntEnb レジスタのビットがイネーブルにされている時に“1”にセットされます。

Bit0 D_EPrIntStat

割り込み要因を間接指示します。

D_EPrIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する

D_EPrIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

3.1.3 002h *USB_HostIntStat (USB Host Interrupt Status)*

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	002h	<i>USB_HostIntStat</i>	R/(W)	7: <i>VBUS_Err</i>	0: None	1: VBUS Error	00h
			R/(W)	6: <i>LineStateChanged</i>	0: None	1: Line Status Changed	
			R	5: <i>H_SIE_IntStat1</i>	0: None	1: SIE Interrupts1	
			R	4: <i>H_SIE_IntStat0</i>	0: None	1: SIE Interrupts0	
			R	3: <i>H_FrameIntStat</i>	0: None	1: Frame Interrupts	
				2:	0:	1:	
			R	1: <i>H_CH0IntStat</i>	0: None	1: CH0 Interrupts	
			R	0: <i>H_CHrIntStat</i>	0: None	1: CHr Interrupts	

ホスト関連の割り込みを表示します。

このレジスタには割り込み要因を間接指示するビットと直接指示するビットがあります。割り込み要因を間接指示するビットは、それぞれに対応する割り込みステータスレジスタをリードすることにより、割り込み要因を直接指示するビットまで辿ることができます。割り込み要因を間接指示するビットは、リードオンリーであり、大元の割り込み要因を直接指示するビットをクリアすることにより、自動的にクリアされます。割り込み要因を直接指示しているビットは、書き込み可能であり、該当ビットに“1”を書き込むことにより、割り込み要因をクリアすることができます。

Bit7 *VBUS_Err*

割り込み要因を直接指示します。このビットは SLEEP 中も有効です。

VBUSFLG 端子に外部接続の VBUS パワースイッチから VBUS 異常を知らせる信号 (High から Low への変化 Edge) が入力された時に “1” にセットされます。

H_USB_Status レジスタの VBUS_State ビットによって VBUSFLG 端子の状態を確認して下さい。

上記の異常信号とは、外部接続の VBUS パワースイッチの仕様によって異なりますので、その仕様をご確認下さい。

Bit6 *LineStateChanged*

割り込み要因を直接指示します。このビットは SLEEP 中も有効です。

ホストポートの DP 端子及び DM 端子の状態が変化したことを示します。

この割り込みは、USB ホスト機能及び USB デバイス機能を有効にしていないとき、あるいは USB ホスト機能を有効にしている SUSPEND 状態にあるときの信号ライン変化を検出するために使用します。

Bit5 *H_SIE_IntStat1*

割り込み要因を間接指示します。

H_SIE_IntStat1 レジスタに割り込み要因があり、かつその割り込み要因に対応する H_SIE_IntEnb1 レジスタのビットがイネーブルにされている時に “1” にセットされます。

Bit4 *H_SIE_IntStat0*

割り込み要因を間接指示します。

H_SIE_IntStat0 レジスタに割り込み要因があり、かつその割り込み要因に対応する H_SIE_IntEnb0 レジスタのビットがイネーブルにされている時に “1” にセットされます。

3. レジスタ詳細説明

Bit3 H_FrameIntStat

割り込み要因を間接指示します。

H_FrameIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する
H_FrameIntEnb レジスタのビットがイネーブルにされている時に“1”にセットされます。

Bit2 Reserved

Bit1 H_CH0IntStat

割り込み要因を間接指示します。

H_CH0IntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する
H_CH0IntEnb レジスタのビットがイネーブルにされている時に “1” にセットされます。

Bit0 H_CHrIntStat

割り込み要因を間接指示します。

H_CHrIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する
H_CHrIntEnb レジスタのビットがイネーブルにされている時に “1” にセットされます。

3.1.4 003h CPU_IntStat (CPU Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	003h	CPU_IntStat	R (W)	7: RAM_RdCmp	0: None	1: RAM Read Complete	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R (W)	1: DMA_CountUp	0: None	1: DMA Counter Overflow	
			R (W)	0: DMA_Cmp	0: None	1: DMA Complete	

CPU インタフェース関連の割り込みを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 RAM_RdCmp

割り込み要因を直接指示します。

RAM_Rd 機能において、RAM からデータを読み出し、RAM_Rd_XX のデータが有効になったところで、“1”にセットされます。

Bit6-2 Reseerved**Bit1 DMA_CountUp**

割り込み要因を直接指示します。

転送モードがフリーランモードで動作している時に、DMA_Count_HH,HL,LH,LL の値がオーバーフローした時に、“1”にセットされます。DMA_Count_HH,HL,LH,LL の値は 0 に戻り、DMA 動作は継続します。

Bit0 DMA_Cmp

割り込み要因を直接指示します。

DMA 転送が停止されるか、或いは、指定された転送数が終了し、終了処理が完了した時に“1”にセットされます。

3. レジスタ詳細説明

3.1.5 004h FIFO_IntStat (FIFO Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	004h	FIFO_IntStat	R(W)	7: FIFO_DMA_Cmp	0: None	1: DMA Complete on FIFO	00h
				6:	0:	1	
				5:	0:	1	
				4:	0:	1	
				3:	0:	1:	
			R (W)	2: FIFO_NotEmpty	0: None	1: FIFO NotEmpty	
			R (W)	1: FIFO_Full	0: None	1: FIFO Full	
			R (W)	0: FIFO_Empty	0: None	1: FIFO Empty	

FIFO 関連の割り込みステータスを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 FIFO_DMA_Cmp

割り込み要因を直接指示します。

HostDeviceSel.HOSTxDEVICE=0 の時、DMA にジョインされているエンドポイントが IN 方向の場合、DMA 転送が終了した後に FIFO が空になると、“1”にセットされます。DMA にジョインされているエンドポイントが OUT 方向の場合、DMA 転送が終了すると、“1”にセットされます。

HostDeviceSel.HOSTxDEVICE=1 の時、DMA にジョインされているチャンネルが OUT 方向の場合、DMA の転送が終了した後に FIFO が空になると、“1”にセットされます。DMA にジョインされているチャンネルが IN 方向の場合、DMA の転送が終了すると、“1”にセットされます。

Bit6-3 Reseved

Bit2 FIFO_NotEmpty

割り込み要因を直接指示します。

AREAn{n=0-5}Join_0.JoinFIFO_Stat ビットが“1”にセットされているときに、該当するエリアの FIFO 領域にデータが存在する状態になると、“1”にセットされます。

Bit1 FIFO_Full

割り込み要因を直接指示します。

AREAn{n=0-5}Join_0.JoinFIFO_Stat ビットが“1”にセットされているときに、該当するエリアの FIFO 領域が Full になると、“1”にセットされます。

Bit0 FIFO_Empty

割り込み要因を直接指示します。

AREAn{n=0-5}Join_0.JoinFIFO_Stat ビットが“1”にセットされているときに、該当するエリアの FIFO 領域が Empty になると、“1”にセットされます。

3.1.6 008h MainIntEnb (Main Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	008h	MainIntEnb	R/W	7: EnUSB_DeviceIntStat	0: Disable	1: Enable	00h
			R/W	6: EnUSB_HostIntStat	0: Disable	1: Enable	
			R/W	5: EnCPU_IntStat	0: Disable	1: Enable	
			R/W	4: EnFIFO_IntStat	0: Disable	1: Enable	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1	
			R/W	0: EnFinishedPM	0: Disable	1: Enable	

MainIntStat レジスタの割り込み要因による割り込み信号(XINT)のアサートを許可／禁止するレジスタです。

対応するビットを“1”にセットすることで割り込みを許可します。

EnUSB_DeviceIntStat, EnUSB_HostIntStat, EnFinishedPM ビットは SLEEP 中も有効です。

3. レジスタ詳細説明

3.1.7 009h USB_DeviceIntEnb (Device Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	009h	<i>USB_DeviceIntEnb</i>	R/W	7: <i>EnVBUS_Changed</i>	0: Disable	1: Enable	00h
				6:	0:	1:	
			R/W	5: <i>EnD_SIE_IntStat</i>	0: Disable	1: Enable	
			R/W	4: EnD_BulkIntStat	0: Disable	1: Enable	
			R/W	3: EnRcvEP0SETUP	0: Disable	1: Enable	
				2:	0:	1:	
			R/W	1: EnD_EP0IntStat	0: Disable	1: Enable	
			R/W	0: EnD_EPIntStat	0: Disable	1: Enable	

USB_DeviceIntStat レジスタの割り込み要因による、MainIntStat レジスタの USB_DeviceIntStat ビットの
アサートを許可／禁止します。

EnVBUS_Changed, EnD_SIE_IntStat ビットは SLEEP 中も有効です。

3.1.8 00Ah *USB_HostIntEnb (Host Interrupt Enable)*

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	00Ah	USB_HostIntEnb	R/W	7: EnVBUS_Err	0: Disable	1: Enable	00h
			R/W	6: EnLineStateChanged	0: Disable	1: Enable	
			R/W	5: EnH_SIE_IntStat1	0: Disable	1: Enable	
			R/W	4: EnH_SIE_IntStat0	0: Disable	1: Enable	
			R/W	3: EnH_FrameIntStat	0: Disable	1: Enable	
			R/W	2:	0:	1:	
			R/W	1: EnH_CH0IntStat	0: Disable	1: Enable	
			R/W	0: EnH_CHrIntStat	0: Disable	1: Enable	

USB_HostIntStat レジスタの割り込み要因による、MainIntStat レジスタの USB_HostIntStat ビットのアサートを許可／禁止します。

EnVBUS_Err ビット、EnLineStateChanged ビットは SLEEP 中も有効です。

3. レジスタ詳細説明

3.1.9 00Bh CPU_IntEnb (CPU Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	00Bh	CPU_IntEnb	R/W	7: EnRAM_RdCmp	0: Disable	1: Enable	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: EnDMA_CountUp	0: Disable	1: Enable	
			R/W	0: EnDMA_Cmp	0: Disable	1: Enable	

CPU_IntStat レジスタの割り込み要因による、MainIntStat レジスタの CPU_IntStat ビットのアサートを許可／禁止します。

3.1.10 00Ch FIFO_IntEnb (FIFO Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	00Ch	FIFO_IntEnb	R/W	7: EnFIFO_DMA_Cmp	0: Disable	1: Enable	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R/W	2: EnFIFO_NotEmpty	0: Disable	1: Enable	
			R/W	1: EnFIFO_Full	0: Disable	1: Enable	
			R/W	0: EnFIFO_Empty	0: Disable	1: Enable	

FIFO_IntStat レジスタの割り込み要因による、MainIntStat レジスタの FIFO_IntStat ビットのアサートを許可／禁止します。

3. レジスタ詳細説明

3.1.11 010h RevisionNum (Revision Number)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	010h	RevisionNum	R	7: RevisionNum [7]	Revision Number	08h
				6: RevisionNum [6]		
				5: RevisionNum [5]		
				4: RevisionNum [4]		
				3: RevisionNum [3]		
				2: RevisionNum [2]		
				1: RevisionNum [1]		
				0: RevisionNum [0]		

本 LSI のリビジョンナンバーを示します。このレジスタは SLEEP 中でもアクセス有効です。
現仕様に関してのリビジョンナンバーは、0x08 となっています。

3.1.12 011h *ChipReset* (*Chip Reset*)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	011h	<i>ChipReset</i>	R/W	7: <i>ResetMTM</i>	0: None	1: MTM Reset	80h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			W	0: <i>AllReset</i>	0: None	1: Reset	

本 LSI をリセットします。

SLEEP 時でもアクセス有効です。

Bit7 *ResetMTM*

このビットに“1”をセットすると、本 LSI のトランシーバマクロ (MTM) を初期化します。
リセットを解除するには、このビットを“0”にクリアして下さい。

Bit6-1 *Reserved***Bit0 *AllReset***

CPUIF_MODE レジスタを除く、全回路を初期化します。本レジスタによるソフトリセットでは、未初期化期間には遷移しません。

なお、このレジスタにリセット用途以外で、書き込みしないで下さい。

AC スペックに違反してこのレジスタにリセット用途以外の書き込みを行った場合、誤動作の原因となりますのでご注意ください。

3. レジスタ詳細説明

3.1.13 012h PM_Control (Power Management Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	012h	<i>PM_Control</i>	R/W	7: <i>GoSLEEP</i>	0: Do nothing	1: Go to SLEEP	00h
			R/W	6: <i>GoACTIVE</i>	0: Do nothing	1: Go to ACTIVE	
			W	5: <i>GoCPU_Cut</i>	0: Do nothing	1: Go to CPU Cut mode	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R	1: <i>PM_State[1]</i>	Power Management State 00: SLEEP, 01: (SNOOZE), 11: ACTIVE		
				0: <i>PM_State[0]</i>			

本 LSI のパワーマネージメント関連の動作設定を行います。

このレジスタは SLEEP 中も有効です。

Bit7 **GoSLEEP**

ACTIVE ステートから SLEEP ステートへの移行を開始します。

ACTIVE ステート時に本ビットを“1”にセットすると、まず PLL の発振を停止し、その後オシレータの発振を停止し SLEEP ステートに移行します。

どのステートからの移行であっても、移行が完了し次第、本ビットは自動的にクリアされ、同時に MainIntStat.FinishedPM ビットがセットされます。

Bit6 **GoACTIVE**

SLEEP ステートから ACTIVE ステートへの移行を開始します。

SLEEP ステート時に本ビットを“1”にセットすると、オシレータの発振を開始し、オシレータ発振安定時間 (WakeupTim_H, L に設定された時間) 経過後、PLL の発振を開始し、PLL480 発振安定時間 (約 250us) 経過後、ACTIVE ステートに移行します。

どのステートからの移行であっても、移行が完了し次第、本ビットは自動的にクリアされ、同時に MainIntStat.FinishedPM ビットがセットされます。

Bit5 **GoCPU_Cut**

SLEEP ステート時、このビットをセットすることで、更に消費電流を低減する CPU_Cut モードに移行します。

SLEEP ステート完全移行後、このビットをセットすると、この CPU 書き込みステート終了と同時に、CPU インタフェイスの入力端子を XCS 端子を除き初段から OFF にします。これにより、XCS 端子以外の信号線が動作しようと、CPU インタフェイスの初段のドライバが OFF となっているため、不必要な電力消費を抑えることが出来ます。

このモードから復帰するには、本レジスタをダミーリードしてください。ただし、この時、読み出されるデータは、リードデータは確定しませんので、注意してください。また、この復帰動作は、ダミーリード終了と同時に行われるため、必ず XCS が一旦ネゲート (“L” → “H”) になるようにアクセスしてください。一般の CPU の場合、本 LSI 以外のアドレス空間をアクセスすると、このような動作が行われますので、ご参考下さい。

※ 本 LSI は SLEEP 中にアクセスできない割り込みステータス (以下、同期ステータス) により、SLEEP 中に XINT 信号がアサートされないようマスクしておりますが、SLEEP 解除と同時に XINT 端子がアサートされるのを回避するため、F/W により以下の処理を行って下さい。

＜SLEEP 開始前＞

同期ステータスを処理し、クリアする（～IntStat）。

同期ステータスをディスエーブルにする（～IntEnb）。

＜SLEEP 解除後（ACTIVE 開始後）＞

同期ステータスをクリアする（～IntStat）。

同期ステータスをイネーブルにする（～IntEnb）。

Bit4-2 **Reserved**

Bit1-0 ***PM_State [1:0]***

電力モードのステータを表します。

00: SLEEP ステート (OSC オフ、PLL オフ)

01: SLEEP ステート※ (OSC オン、PLL オフ)

11: ACTIVE ステート (OSC オン、PLL オン)

なお本ステータスは、GoSLEEP ビット或いは GoACTIVE ビットをセットしてから MainIntStat.FinishedPM 割り込みステータスがセットされるまでは、該当ステータに向かい逐次変化しますので、参照しないでください。

3. レジスタ詳細説明

3.1.14 014h-015h WakeupTim_H,L (Wakeup Time High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	014h -015h	WakeupTim_H WakeupTim_L	 R/W	 WakeupTim [15:0]	 Wakeup Time [15:0]	 0000h

SLEEP ステートから SNOOZE ステートへ復帰する際のオシレータ発振安定時間を設定します。本レジスタは SLEEP 時にもアクセスが可能です。

SLEEP ステート時に、PM_Control.GoACTIVE ビットに“1”が書き込まれた際に、発振セルをイネーブルにし、オシレータの発振を開始します。この時、カウンタにこの **WakeupTim_H,L** の設定値をロードし、OSC の立ち上がりにてカウントダウンを始めます。カウントダウン終了後、内部 OSCCLK のゲートを開き、PLL 等の回路に CLK を送出開始します。

このオシレータ発振安定時間は、発振子、発振セル、基板、負荷容量等により変化します。もし、デバイス動作時において、USB の SUSPEND 時に SLEEP ステートまで落とす場合は、USB の RESET 検出から 5.1ms 以内に 60MHz±10%に内部 SCLK が安定していなければなりません。

したがって、オシレータ発振安定時間 + PLL 安定時間（250us 未満）の合計が 5.1ms 以下とならなければなりません。

3.1.15 016h H_USB_Control (Host USB Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	016h	<i>H_USB_Control</i>	R/W	7: <i>VBUS_Enb</i>	0: Disable	1: Enable	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

ホスト に関する動作設定を行います。

このレジスタは SLEEP 中も有効です。

Bit7 *VBUS_Enb*

VBUSEN 端子（出力）の状態を設定します。デフォルトでローレベルです。

Bit6-0 *Reserved*

3. レジスタ詳細説明

3.1.16 017h H_XcvrControl (Host Xcvr Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	017h	<i>H_XcvrControl</i>	R/W	7: <i>TermSelect</i>	0: HS Termination	1: FS Termination	91h
			R/W	6: <i>RemoveRPD</i>	0: RPD ON	1: RPD OFF	
			R/W	4: <i>XcvrSelect[1]</i>	XcvrSelect[1:0]		
			R/W	4: <i>XcvrSelect[0]</i>			
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: <i>OpMode [1]</i>	OpMode [1:0]		
				0: <i>OpMode [0]</i>			

ホストトランシーバマクロに関する設定を行います。

このレジスタは SLEEP 中も有効です。

Bit7 **TermSelect**

HS または FS いずれかのターミネーションを選択して有効にします。

本ビットは、H_NogoControl_0.AutoMode を設定することにより H/W が自動的に設定を行いますので、設定を行わないでください。

Bit6 **RemoveRPD**

ホストデータラインである DP および DM 用内蔵プルダウン抵抗を ON/OFF します。

0: RPD ON

1: RPD OFF

本ビットは USB ホスト動作時（SUSPEND も含む）においては、必ず“0”（ON）の設定のままで使用してください。本ビットを“0”から変化させますと、ホストデータラインの特性が変わり、USB における誤動作を引き起こす可能性がありますのでご注意ください。

Bit5-4 **XcvrSelect[1:0]**

HS、FS、LS のいずれかのトランシーバを選択して有効にします。

00:High Speed トランシーバ

01:Full Speed トランシーバ

10: Reseved

11: Low Speed トランシーバ

本ビットは、H_NogoControl_0.AutoMode を設定することにより H/W が自動的に設定を行いますので、設定を行わないでください。

Bit3-2 **Reserved**

Bit1-0 **OpMode[1:0]**

MTM のオペレーションモードを設定します。

本ビットは、H_NogoControl_0.AutoMode を設定することにより H/W が自動的に設定を行いますので、設定を行わないでください。

ただし、信号ライン変化ステータスを検出する場合は、「1.1.2.2 信号ライン変化ステータスの使用方法」を参照のうえ、本ビットの設定を行ってください。

OpMode		
00	"Normal Operation"	通常使用状態
01	"Non-Driving"	非使用状態
10	"Disable Bitstuffing and NRZI encoding"	通常使用状態で Bitstuffing と NRZI encoding 機能ディセーブル状態
11	"Power-Down"	シングルエンドレシーバのみ使用する状態

3. レジスタ詳細説明

3.1.17 018h D_USB_Status (Device USB Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	018h	<i>D_USB_Status</i>	R	7: <i>VBUS</i>	0: VBUS = L	1: VBUS = H	XXh
			R/W	6: FSxHS	0: HS mode	1: FS mode	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R	1: <i>LineState</i> [1]	Line State [1:0]		
				0: <i>LineState</i> [0]			

デバイスに関するステータスを表示します。

Bit7 **VBUS**

VBUS 端子の状態が表示されます。このビットは SLEEP 中でも有効です。

Bit6 **FSxHS**

現在の動作モードを示します。D_NegoControl.GoChirp ビットにより“HS Detection Handshake”（機能説明参照）を実行すると、自動的に設定されます。このビットを書き込むことにより動作モードを強制的に変更することも可能ですが、シミュレーション等で“HS Detection Handshake”を行わずに動作モードを切り替えたい場合にのみ、このビットを操作してください。

ケーブルアタッチ時に、“FS (1)”にセットして下さい。

このビットは SLEEP 中でも読み出すことが出来ますが、書き込みは ACTIVE 時になります。

Bit5-2 **Reserved**

Bit1-0 **LineState [1:0]**

USB ケーブル上の信号状態を示します。このビットは SLEEP 中でも有効です。

D_XcvrControl レジスタの TermSelect ビットが“1”である時 (FS ターミネーション選択時)、XcvrSelect ビットが “1”

(FS トランシーバ選択時) であれば、DP/DM の FS レシーバの受信値を、XcvrSelect が“0” (HS トランシーバ選択時) であれば、HS レシーバの受信値を示します。

TermSelect が “0” である時は、USB のバス・アクティビティを示します。

LineState		
TermSelect	DP / DM	LineState [1:0]
0	Don't Care	バス・アクティビティ
1	SE0	0b00
1	J	0b01
1	K	0b10
1	SE1	0b11

3.1.18 019h *H_USB_Status* (Host USB Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	019h	<i>H_USB_Status</i>	R	7: <i>VBUS_State</i>	0: VBUSFLG = High	1: VBUSFLG= Low	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R	1: <i>LineState</i> [1]	Line State [1:0]		
				0: <i>LineState</i> [0]			

ホストに関するステータスを表示します。

このレジスタは SLEEP 中も有効です。

Bit7 *VBUS_State*

VBUSFLG 端子の状態が表示されます。

Bit6-2 *Reserved***Bit1-0** *LineState* [1:0]

USB ケーブル上の信号状態を示します。

H_XcvrControl レジスタの *XcvrSelect*[1:0]が“01”である時 (FS トランシーバ選択時) は DP/DM の FS レシーバの受信値を、“11”である時 (LS トランシーバ選択時) は DP/DM の LS レシーバの受信値を示します。

XcvrSelect[1:0]が“00”である時(HS トランシーバ選択時)は、USB のバス・アクティビティを示します。

LineState		
<i>XcvrSelect</i> [1:0]	DP / DM	<i>LineState</i> [1:0]
00	Don't Care	バス・アクティビティ アクティビティ有: 0b01 アクティビティ無: 0b00
01 または 11	SE0	0b00
01 または 11	J	0b01
01 または 11	K	0b10
01 または 11	SE1	0b11

(注) *XcvrSelect*[1:0]＝“10”のコードは Reserved のため、そのときの動作は保証いたしません。

3. レジスタ詳細説明

3.1.19 01Bh MTM_Config (Multi Transceiver Macro Config)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	01Bh	MTM_Config		7:	0:	1:	XXh
				6:	0:	1:	
			R/W	5: MTM_SlopeValue[1]	MTM Slope Value[1:0]		
			R/W	4: MTM_SlopeValue[0]			
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: MTM_TermValue[1]	MTM Termination Value[1:0]		
			R/W	0: MTM_TermValue[0]			

トランシーバマクロの調整値設定用レジスタです。

Bit7-6 **Reserved**

Bit5-4 **MTM_SlopeValue[1:0]**

HS トランスミッタのスルーレートを調整します。4 段階の調整が可能です。

00 : 緩

01 : ↑

10 : ↓

11 : 急

Bit3-2 **Reserved**

Bit1-0 **MTM_TermValue[1:0]**

HS 伝送路のターミネーションを調整します。4 段階の調整が可能です。

00 : 高

01 : ↑

10 : ↓

11 : 低

3.1.20 01Fh *HostDeviceSel* (Host Device Select)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	01Fh	<i>HostDeviceSel</i>		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R/W	0: <i>HOSTxDEVICE</i>	0: Device mode	1: Host mode	

USB のデバイス・モードまたはホスト・モードの設定を行います。

このレジスタは SLEEP 中も有効です。

Bit7-1 **Reserved**

Bit0 ***HOSTxDEVICE***

USB のデバイス・モードまたはホスト・モードの設定を行います。

0: USB デバイス・モード

1: USB ホスト・モード

このビットによって、内部システムクロックを操作しています。HOST x DEVICE が“0”の時、すなわち Device Mode の場合には、共通ブロックと USB のデバイスブロックへシステムクロックを供給し、USB のホストブロックへのシステムクロックを停止します。HOST x DEVICE が“1”の時、すなわち Host Mode の場合には、共通ブロックと USB のホストブロックへシステムクロックを供給し、USB のデバイスブロックへのシステムクロックを停止します。

3. レジスタ詳細説明

3.1.21 020h FIFO_Rd_0 (FIFO Read 0)

3.1.21 021h FIFO_Rd_1 (FIFO Read 1)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	020h	FIFO_Rd_0	R	7: FIFO_Rd_0 [7]	FIFO Read	XXh
				6: FIFO_Rd_0 [6]		
				5: FIFO_Rd_0 [5]		
				4: FIFO_Rd_0 [4]		
				3: FIFO_Rd_0 [3]		
				2: FIFO_Rd_0 [2]		
				1: FIFO_Rd_0 [1]		
				0: FIFO_Rd_0 [0]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	021h	FIFO_Rd_1	R	7: FIFO_Rd_1 [7]	FIFO Read	XXh
				6: FIFO_Rd_1 [6]		
				5: FIFO_Rd_1 [5]		
				4: FIFO_Rd_1 [4]		
				3: FIFO_Rd_1 [3]		
				2: FIFO_Rd_1 [2]		
				1: FIFO_Rd_1 [1]		
				0: FIFO_Rd_1 [0]		

020h.Bit7-0 FIFO_Rd_0 [7:0]

021h.Bit7-0 FIFO_Rd_1[7:0]

AREAn{n=0-5}Join_0.JoinCPU_Rd ビットがセットされている FIFO のデータを読み出せません。

FIFO にバイト境界がある場合に、このレジスタを読み出した場合は片側にのみ有効なデータが出力されます。詳細は機能説明“FIFO アクセスの端数処理”をご参照ください。

このレジスタを用いて FIFO のデータを読み出す場合は、必ず FIFO_RdRemain_H,L レジスタにより読み出し可能データ数を確認した後、リードを行ってください。

3.1.23 022h FIFO_Wr_0(FIFO Write 0)**3.1.24 023h FIFO_Wr_1(FIFO Write 1)**

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	022h	FIFO_Wr_0	W	7: FIFO_Wr_0 [7]	FIFO Write	XXh
				6: FIFO_Wr_0 [6]		
				5: FIFO_Wr_0 [5]		
				4: FIFO_Wr_0 [4]		
				3: FIFO_Wr_0 [3]		
				2: FIFO_Wr_0 [1]		
				1: FIFO_Wr_0 [1]		
				0: FIFO_Wr_0 [0]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	023h	FIFO_Wr_1	W	7: FIFO_Wr_1 [7]	FIFO Write	XXh
				6: FIFO_Wr_1 [6]		
				5: FIFO_Wr_1 [5]		
				4: FIFO_Wr_1 [4]		
				3: FIFO_Wr_1 [3]		
				2: FIFO_Wr_1 [2]		
				1: FIFO_Wr_1 [1]		
				0: FIFO_Wr_1 [0]		

022h.Bit7-0 FIFO_Wr_0 [7:0]**023h.Bit7-0 FIFO_Wr_1[7:0]**

AREAn{n=0-5}.Join_0.JoinCPU_Wr ビットがセットされている FIFO ヘデータを書き込みめます。

FIFO にバイト境界がある場合に、このレジスタへ書き込んだ場合は片側のみ書き込みが行われます。詳細は機能説明“FIFO アクセスの端数処理”をご参照ください。

このレジスタを用いて FIFO ヘデータを書き込む場合は、必ず FIFO_WrRemain_H,L レジスタにより、書き込み可能データ数を確認した後、ライトを行ってください。

3. レジスタ詳細説明

3.1.25 024h-025h FIFO_RdRemain_H,L (FIFO Read Remain High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset	
Device / Host	024h-025h	FIFO_RdRemain_H,L	R	15: RdRemainValid	0:None	1: Read Remain Valid	00h	
				14:	0:	1:		
				13:	0:	1:		
			R	12: RdRemain [12]	Read Remain			
				11: RdRemain [11]				
				10: RdRemain [10]				
				9: RdRemain [9]				
				8: RdRemain [8]				
				7: RdRemain [7]				
				6: RdRemain [6]				
				5: RdRemain [5]				
				4: RdRemain [4]				
				3: RdRemain [3]				
				2: RdRemain [2]				
				1: RdRemain [1]				
				0: RdRemain [0]				

Bit15 RdRemainValid

AREAn{n=0-5}Join_0.JoinCPU_Rd ビットによって、CPU I/F に FIFO がジョインされかつ、FIFO_RdRemain の値が有効な場合に“1”にセットされます。

このビットがクリアされている場合の RdRemain の値は無効です。

Bit14-13 Reserved

Bit12 -0 RdRemain [12:0]

AREAn{n=0-5}Join_0.JoinCPU_Rd ビットによって、CPU I/F に接続している FIFO 内のリード可能なデータ数を示します。FIFO のリード可能なデータ数を取得する場合は、FIFO_RdRemain_H レジスタと FIFO_RdRemain_L レジスタを対でアクセスする必要があります。

3.1.26 026h-027h FIFO_WrRemain_H,L (FIFO Write Remain High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	026h-027h	WrRemain_H,L		15:	0:	1:	00h
				14:	0:	1:	
				13:	0:	1:	
			R	12: WrRemain [12]	FIFO Write Remain		
				11: WrRemain [11]			
				10: WrRemain [10]			
				9: WrRemain [9]			
				8: WrRemain [8]			
				7: WrRemain [7]			
				6: WrRemain [6]			
				5: WrRemain [5]			
				4: WrRemain [4]			
				3: WrRemain [3]			
				2: WrRemain [2]			
				1: WrRemain [1]			
				0: WrRemain [0]			

Bit15-13 **Reserved****Bit12 -0** **WrRemain [12:0]**

AREAn{n=0-5}Join_0.JoinCPU_Wr ビットによって、CPU I/F に接続している FIFO の空き容量を示します。ただし、FIFO への書き込み直後には正確な FIFO の空き容量を確認することは出来ません。1CPU サイクル以上の間隔を開けて FIFO の空き容量を確認して下さい。FIFO の空き容量を取得する場合は、FIFO_WrRemain_H レジスタと FIFO_WrRemain_L レジスタを対でアクセスする必要があります。

3. レジスタ詳細説明

3.1.27 028h FIFO_ByteRd(FIFO Byte Read)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	028h	FIFO_ByteRd	R	7: FIFO_ByteRd [7]	FIFO Byte Read	XXh
				6: FIFO_ByteRd [6]		
				5: FIFO_ByteRd [5]		
				4: FIFO_ByteRd [4]		
				3: FIFO_ByteRd [3]		
				2: FIFO_ByteRd [2]		
				1: FIFO_ByteRd [1]		
				0: FIFO_ByteRd [0]		

Bit7-0 FIFO_ByteRd [7:0]

AREAn{n=0-5}.Join_0.JoinCPU_Rd ビットがセットされている FIFO のデータをバイト単位で読み出せます。このレジスタを用いて FIFO のデータを読み出す場合は、必ず FIFO_RdRemain_H,L レジスタにより読み出し可能データ数を確認した後、リードを行ってください。

3.1.28 02Ah FIFO_ByteWr(FIFO Byte Write)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	02Ah	FIFO_ByteWr	W	7: FIFO_ByteWr [7]	FIFO Byte Write	XXh
				6: FIFO_ByteWr [6]		
				5: FIFO_ByteWr [5]		
				4: FIFO_ByteWr [4]		
				3: FIFO_ByteWr [3]		
				2: FIFO_ByteWr [2]		
				1: FIFO_ByteWr [1]		
				0: FIFO_ByteWr [0]		

Bit7-0 FIFO_ByteWr[7:0]

AREAn{n=0-5}.Join_0.JoinCPU_Wr ビットがセットされている FIFO に、データをバイト単位で書き込めます。このレジスタを用いて FIFO のデータを書き込む場合は、必ず FIFO_WrRemain_H,L レジスタにより書き込み可能データ数を確認した後、ライトを行ってください。

3. レジスタ詳細説明

3.1.29 02Ch CacheRemain (Host Device Select)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	02Ch	CacheRemain		7:WrFIFO_C_Remain	0: Not Remain	1: Remain	00h
				6:WrDOOR_C_Remain	0: Not Remain	1: Remain	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R/W	0:	0:	1:	

ライトキャッシュの状態を示します。

Bit7 WrFIFO_C_Remain

FIFO_Wr_0,1 または FIFO_ByteWr レジスタ使用時、書き込みデータがキャッシュに残っていることを示すビットです。

0：キャッシュにデータが存在しません。

1：キャッシュにデータが存在します。

FIFO_Wr_0,1、FIFO_ByteWr レジスタへのライト後、次にあげる動作を行う前に、データを FIFO に書き込み終えたことを、本ビットにより確認して下さい。

守られなかった場合は、正しいデータの書き込みが保証されません。

＜WrFIFO_C_Remain ビットの確認が必要となる動作＞

- JoinCPU_Wr を切り替える、または、クリアする。
- AREAnFIFO_Clr にて、FIFO クリアする。

Bit6 WrDOOR_C_Remain

RAM_WrDoor_0,1 レジスタ使用時、書き込みデータがキャッシュに残っていることを示すビットです。

0：キャッシュにデータが存在しません。

1：キャッシュにデータが存在します。

RAM_WrDoor_0,1 レジスタへのライト後、次にあげる動作を行う前に、データを FIFO に書き込み終えたことを、本ビットにより確認して下さい。

守られなかった場合は、正しいデータの書き込みが保証されません。

＜WrDOOR_C_Remain ビットの確認が必要となる動作＞

- RAM_WrDoorAdrs_H,L を書き換える。
- デバイス動作時、EnShortPkt をセットして、ショートパケットの送信を始める。
- デバイス動作時、ReplyDescriptor をセットして、データ送信を始める。
- デバイス動作時、GoCSW_Mode をセットして、CSW 送信を始める。
- ホスト動作時、BO_SupportGo をセットして、CBW 送信を始める。

Bit5-0 Reserved

3.1.30 030h-031h RAM_RdAdrs_H,L (RAM Read Address High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset		
Device / Host	030h-031h	RAM_RdAdrs_H,L		15:	0:	1:	00h		
				14:	0:	1:			
				13:	0:	1:			
			R/W	12: RAM_RdAdrs [12]	RAM Read Address				
				11: RAM_RdAdrs [11]					
				10: RAM_RdAdrs [10]					
				9: RAM_RdAdrs [9]					
				8: RAM_RdAdrs [8]					
				7: RAM_RdAdrs [7]					
				6: RAM_RdAdrs [6]					
				5: RAM_RdAdrs [5]					
				4: RAM_RdAdrs [4]					
				3: RAM_RdAdrs [3]					
				2: RAM_RdAdrs [2]					
				1:				0:	1:
				0:				0:	1:

Bit15-13 Reserved**Bit12-2 RAM_RdAdrs[12:2]**

RAM_Rd を行う先頭アドレスを設定します。このレジスタを設定した後、RAM_RdCount レジスタを設定し、RAM_RdControl レジスタのビットをセットして下さい。RAM_Rd 機能が開始します。このレジスタの値は、RAM_Rd 機能作動中は内部動作に応じて変化します。従って、一旦 RAM_RdControl レジスタのビットをセットして、RAM_Rd 機能を開始させた後は CPU_IntStat.RAM_RdCmp ビットがセットされるまでこのレジスタの値を読み出さないで下さい。RAM_Rd 機能作動中にこのレジスタを読み出した場合の値は保証されません。また、RAM_Rd 機能作動中にこのレジスタに書き込んだ場合、誤動作の原因となりますのでご注意ください。

Bit1-0 Reserved

3. レジスタ詳細説明

3.1.31 032h RAM_RdControl (RAM Read Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	032h	RAM_RdControl	R/W	7: RAM_GoRdCBW_CSW	0: Do nothing	1: RAM Read CBW_CSW start	00h
			R/W	6: RAM_GoRd	0: Do nothing	1: RAM Read start	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

Bit7 RAM_GoRdCBW_CSW

デバイス動作時は CBW エリア、ホスト動作時は CSW エリアに受信したデータを読み出すために RAM_Rd 機能を開始するビットです。

デバイス動作時にこのビットに“1”を書き込むと、RAM_Rd 機能を開始し、CBW エリアからデータをリードします。RAM_Rd_00～RAM_Rd_1E レジスタの値が有効になると、CPU_IntStat.RAM_RdCmp ビットが“1”にセットされ、このビットが自動的にクリアされます。

ホスト動作時にこのビットに“1”を書き込むと、RAM_Rd 機能を開始し、CSW エリアからデータをリードします。RAM_Rd_00～RAM_Rd_0C レジスタの値が有効になると、CPU_IntStat.RAM_RdCmp ビットが“1”にセットされ、このビットが自動的にクリアされます。

いずれの場合も RAM_RdAdrs_H,L レジスタ、RAM_RdCount レジスタの設定は必要ありません。

RAM_GoRd ビットと同時にセットすると、本ビットの機能が優先されます。

Bit6 RAM_GoRd

RAM_Rd 機能を開始するビットです。

RAM_RdAdrs_H,L レジスタに RAM_Rd を行う先頭アドレスを設定した後、RAM_RdCount レジスタを設定し、このビットに“1”を書き込むと RAM_Rd 機能を開始します。指定された先頭アドレスから、指定されたカウント数分のデータをリードし RAM_Rd_xx{xx=00-1F} レジスタの値が有効になると CPU_IntStat.RAM_RdCmp ビットが“1”にセットされ、このビットが自動的にクリアされます。

RAM_GoRdCBW_CSW ビットと同時にセットすると、RAM_GoRdCBW_CSW ビットの機能が優先されます。

Bit5-0 Reserved

3.1.32 035h RAM_RdCount (RAM Read Counter)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	035h	RAM_RdCount		7:	0:	1:	00h
				6:	0:	1:	
			R/W	5: RAM_RdCount [5]	RAM Read Counter		
				4: RAM_RdCount [4]			
				3: RAM_RdCount [3]			
				2: RAM_RdCount [2]			
				1:	0:	1:	
				0:	0:	1:	

Bit7-6 Reserved

Bit5-2 RAM_RdCount [5:2]

RAM_Rd 機能を用いて RAM_Rd_xx{xx=00~1F}レジスタにリードするデータ数を設定します。RAM_RdAdrs_H,L レジスタを設定した後、このレジスタをセットし、RAM_RdControl レジスタのビットをセットして RAM_Rd 機能を開始して下さい。このレジスタの値は、RAM_Rd 機能作動中は内部動作に応じて変化します。従って、一旦 RAM_RdControl レジスタのビットをセットして、RAM_Rd 機能を開始させた後は CPU_IntStat.RAM_RdCmp ビットがセットされるまでこのレジスタの値を読み出さないで下さい。RAM_Rd 機能作動中にこのレジスタを読み出した場合の値は保証されません。また、RAM_Rd 機能作動中にこのレジスタに書き込んだ場合、誤動作の原因となりますのでご注意ください。

このレジスタの最大設定数は 32 バイトです。32 バイトを超えるデータ数の設定は誤動作の原因となりますのでご注意ください。

Bit1-0 Reserved

3. レジスタ詳細説明

3.1.33 038h-039h RAM_WrAdrs_H,L (RAM Write Address High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset	
Device / Host	038h-039h	RAM_WrAdrs_H,L		15:	0:	1:	00h	
				14:	0:	1:		
				13:	0:	1:		
			R/W	12: RAM_WrAdrs [12]		RAM Write Address		
				11: RAM_WrAdrs [11]				
				10: RAM_WrAdrs [10]				
				9: RAM_WrAdrs [9]				
				8: RAM_WrAdrs [8]				
				7: RAM_WrAdrs [7]				
				6: RAM_WrAdrs [6]				
				5: RAM_WrAdrs [5]				
				4: RAM_WrAdrs [4]				
				3: RAM_WrAdrs [3]				
				2: RAM_WrAdrs [2]				
				1: RAM_WrAdrs [1]				
				0: RAM_WrAdrs [0]				

RAM_WrDoor_0,1 レジスタによる RAM へのライトを行うアドレスを指定します。

Bit15-13 Reserved

Bit12-0 RAM_WrAdrs[12:0]

RAM への書き込みを行う際のアドレスを指定します。RAM_WrDoor_0,1 レジスタへの書き込みバイト数に応じてアドレスはインクリメントされます。RAM_WrDoor_0,1 レジスタ書き込み直後には正確な RAM_WrAdrs を確認する事は出来ませんので、1CPU サイクル以上の間隔を開けて、RAM_WrAdrs を確認して下さい。データの書き込みに関しては RAM_WrDoor_0,1 レジスタの項を参照して下さい。

なお、書き込み開始アドレスに以下の制限があります。

- (1) 書き込み開始アドレスを、RAM_WrAdrs_H,L == 0x0100 より下位とする。
- (2) 書き込み開始アドレスを、RAM_WrAdrs_H,L[1:0] == 0b00 (すなわち 4Byte 境界) とする。

3.1.34 03Ah RAM_WrDoor_0 (RAM Write Door 0)**3.1.35 03Bh RAM_WrDoor_1 (RAM Write Door 1)**

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	03Ah	RAM_WrDoor_0	W	7: RAM_WrDoor_0 [7]	RAM Write Door 0	XXh
				6: RAM_WrDoor_0 [6]		
				5: RAM_WrDoor_0 [5]		
				4: RAM_WrDoor_0 [4]		
				3: RAM_WrDoor_0 [3]		
				2: RAM_WrDoor_0 [2]		
				1: RAM_WrDoor_0 [1]		
				0: RAM_WrDoor_0 [0]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	03Bh	RAM_WrDoor_1	W	7: RAM_WrDoor_1 [7]	RAM Write Door 1	XXh
				6: RAM_WrDoor_1 [6]		
				5: RAM_WrDoor_1 [5]		
				4: RAM_WrDoor_1 [4]		
				3: RAM_WrDoor_1 [3]		
				2: RAM_WrDoor_1 [2]		
				1: RAM_WrDoor_1 [1]		
				0: RAM_WrDoor_1 [0]		

03Ah.Bit7-0 RAM_WrDoor_0 [7:0]**03Bh.Bit7-0 RAM_WrDoor_1[7:0]**

RAM へのライトを行う際のアクセスレジスタです。ライトオンリーのレジスタです。

書き込み開始前に、RAM_WrAdrs_H,L レジスタに、RAM のデータを書き込む先頭アドレスを設定して下さい。その後、このレジスタに書き込みを行えば、RAM_WrAdrs_H,L が自動的に書き込みバイト数に応じてインクリメントされ、順次書き込みが行えます。

USB デバイスの場合には、RAM_WrDoor_0,1 レジスタにより、デスク립タエリアおよび CSW エリアへのデータの書き込みを行えます。RAM_WrDoor_0,1 レジスタによりデスク립タエリアへ書き込んだデータは、ReplyDescriptor の機能によって何度でも使用できます。即ち、このデータは Descriptor 返信機能によって、消される、または上書きされることはありません。但し、Descriptor データを書き込んだ領域が、他のエンドポイントで確保されている領域と重なる場合には、データは上書きされることがあります。

USB ホストの場合には、RAM_WrDoor_0,1 レジスタにより、CBW エリアへのデータの書き込みを行えます。

3. レジスタ詳細説明

- 3.1.36 040h RAM_Rd_00 (RAM Read 00)
- 3.1.37 041h RAM_Rd_01 (RAM Read 01)
- 3.1.38 042h RAM_Rd_02 (RAM Read 02)
- 3.1.39 043h RAM_Rd_03 (RAM Read 03)
- 3.1.40 044h RAM_Rd_04 (RAM Read 04)
- 3.1.41 045h RAM_Rd_05 (RAM Read 05)
- 3.1.42 046h RAM_Rd_06 (RAM Read 06)
- 3.1.43 047h RAM_Rd_07 (RAM Read 07)
- 3.1.44 048h RAM_Rd_08 (RAM Read 08)
- 3.1.45 049h RAM_Rd_09 (RAM Read 09)
- 3.1.46 04Ah RAM_Rd_0A (RAM Read 0A)
- 3.1.47 04Bh RAM_Rd_0B (RAM Read 0B)
- 3.1.48 04Ch RAM_Rd_0C (RAM Read 0C)
- 3.1.49 04Dh RAM_Rd_0D (RAM Read 0D)
- 3.1.50 04Eh RAM_Rd_0E (RAM Read 0E)
- 3.1.51 04Fh RAM_Rd_0F (RAM Read 0F)
- 3.1.52 050h RAM_Rd_10 (RAM Read 10)
- 3.1.53 051h RAM_Rd_11 (RAM Read 11)
- 3.1.54 052h RAM_Rd_12 (RAM Read 12)
- 3.1.55 053h RAM_Rd_13 (RAM Read 13)
- 3.1.56 054h RAM_Rd_14 (RAM Read 14)
- 3.1.57 055h RAM_Rd_15 (RAM Read 15)
- 3.1.58 056h RAM_Rd_16 (RAM Read 16)
- 3.1.59 057h RAM_Rd_17 (RAM Read 17)
- 3.1.60 058h RAM_Rd_18 (RAM Read 18)
- 3.1.61 059h RAM_Rd_19 (RAM Read 19)
- 3.1.62 05Ah RAM_Rd_1A (RAM Read 1A)
- 3.1.63 05Bh RAM_Rd_1B (RAM Read 1B)
- 3.1.64 05Ch RAM_Rd_1C (RAM Read 1C)
- 3.1.65 05Dh RAM_Rd_1D (RAM Read 1D)
- 3.1.66 05Eh RAM_Rd_1E (RAM Read 1E)

3.1.67 05Fh RAM_Rd_1F (RAM Read 1F)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	040h -05Fh	RAM_Rd_00 ~ RAM_Rd_1F	R	7: RAM_Rd_xx [7]	RAM Read	00h
				6: RAM_Rd_xx [6]		
				5: RAM_Rd_xx [5]		
				4: RAM_Rd_xx [4]		
				3: RAM_Rd_xx [3]		
				2: RAM_Rd_xx [2]		
				1: RAM_Rd_xx [1]		
				0: RAM_Rd_xx [0]		

040h-05Fh.Bit7-0 RAM_Rd_xx [7:0]

RAM_Rd 機能を用いて RAM からリードしたデータを格納するレジスタです。RAM_RdAdrs_H,L レジスタ、RAM_RdCount レジスタを設定し、RAM_RdControl レジスタのビットを用いて RAM_Rd 機能を開始して下さい。本レジスタの値が有効になると FIFO_IntaStat.RAM_RdCmp ビットが“1”にセットされます。RAM_RdCount レジスタに設定した値が 32 バイト未満の場合、RAM からリードしたデータは RAM_Rd_00 から順に格納されます。RAM_RdCount レジスタに設定したカウント数以降のレジスタの値（例えば、カウント設定が“16”の場合、RAM_Rd_10～RAM_Rd_1F）は無効となります。

3. レジスタ詳細説明

3.1.68 061h DMA_Config (DMA Config)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	061h	DMA_Config	R/W	7: FreeRun	0: Count mode	1: FreeRun mode	00h
			R/W	6: DMA_Mode	0: Normal mode	1: Address Decode mode	
				5:	0:	1:	
				4:	0:	1:	
			R/W	3: ActiveDMA	0: DMA Inactive	1: DMA Active	
				2:	0:	1:	
			R/W	1: ReqAssertCount [1]	Request Assert Count		
				0: ReqAssertCount [0]			

DMA の動作モードを設定します。

Bit7 FreeRun

DMA の動作モードを設定します。

0 : カウントモード

1 : フリーランモード

Bit6 DMA_Mode

DMA のモードを設定します。

0 : ホストからの XDACK をアクノリッジとして DMA 動作します。

1 : ホストからの DMA_RdData/DMA_WrData レジスタに対するアクセスをアクノリッジとして DMA 動作します。

Bit5-4 Reserved

Bit3 ActiveDMA

DMA の DACK を有効にします。

0 : DMA (DACK) 無効

1 : DMA (DACK) 有効

Bit2 Reserved

Bit1-0 ReqAssertCount [1:0]

CPU のバーストリード／バーストライトに対応するための、REQ アサートカウントオプション設定ビットです。

XDREQ のアサートカウント数（転送バイト数）を設定します。設定されたアサートカウント数以上の書き込み可能な空き領域／読み出し可能なデータが FIFO にある場合に XDREQ をアサートします。設定されたアサートカウント数の DMA 転送を終了すると一旦 XDREQ をネゲートし、再度アサートカウント数以上の空き領域／データを確認した時点で XDREQ をアサートします。つまり、一回の XDREQ のアサートに対し、設定されたアサートカウント数分の転送を保証します。

ただし、カウントモードに設定され、且つ DMA_Count_HH,HL,LH,LL の残りカウント数が設定されたアサートカウント数より小さい場合は DMA_Count_HH,HL,LH,LL のカウント数が優先され、DMA_Count_HH,HL,LH,LL のカウント数以上の空き領域／データが FIFO にある場合に XDREQ をアサートします。

下表に DMA_Count_HH,HL,LH,LL（表では Count）、ReqAssertCount（表では Req）、FIFO の空き領域／データ（表では Ready）と XDREQ 信号及び転送可能数の関係を示します。

DMA_Count_HH,HL,LH,LL の残りカウントが“1”以上である事が必須条件です。

	Count>=Req		Count<Req	
	Ready>=Req	Ready<Req	Ready>=Count	Ready<Count
XDREQ	アサート	ネゲート	アサート	ネゲート
転送可能数	Req	-	Req	-

ReqAssertCount [1:0]	モード
	16bit mode
0b00	Normal
0b01	16Byte(8Count)
0b10	32Byte(16Count)
0b11	64Byte(32Count)

00 (Normal) 設定時には REQ アサートカウントオプション未使用となります。

3. レジスタ詳細説明

3.1.69 062h DMA_Control (DMA Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	062h	DMA_Control	R	7: DMA_Running	0: DMA is not running	1: DMA is running	00h
				6:	0:	1:	
				5:	0:	1:	
			W	4: CounterClr	0: Do nothing	1: Clear DMA counter	
			R/W	3: Dir	0: CPU-IF → FIFO RAM	1: CPU-IF ← FIFO RAM	
				2:	0:	1:	
			W	1: DMA_Stop	0: Do nothing	1: Finish DMA	
			W	0: DMA_Go	0: Do nothing	1: Start DMA	

DMA の制御、及び状態を表示します。

Bit7 DMA_Running

DMA の転送中、このビットが“1”にセットされます。このビットが“1”である間は、AREAn{n=0-5}.Join_0.JoinDMA ビットを書き換えることはできません。

Bit6-5 Reserved

Bit4 CounterClr

このビットに“1”をセットすると、DMA_Count_HH,HL,LH,LL レジスタが 0x00 にクリアされます。DMA_Running ビットが“1”である時は、このビットへの書き込みは無視されます。

Bit3 Dir

DMA の転送方向を設定します。

0: CPU-IF → FIFO RAM (DMA ライト)

1: CPU-IF ← FIFO RAM (DMA リード)

Bit2 Reserved

Bit1 DMA_Stop

このビットに“1”をセットすると、DMA の転送を終了します。DMA の転送を停止すると、DMA_Running ビットを“0”にクリアします。また、CPU_IntStat レジスタの DMA_Cmp ビットに“1”をセットします。DMA の転送を再開する場合、DMA_Running ビットまたは DMA_Cmp ビットを確認し、DMA が終了するのを待って行って下さい。

Bit0 DMA_Go

このビットを“1”にセットすると、DMA の転送を開始します。

3.1.70 064h-065h DMA_Remain_H,L (DMA FIFO Remain High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	064h-065h	DMA_Remain_H,L		15:	0:	1:	0000h
				14:	0:	1:	
				13:	0:	1:	
			R	12: DMA_Remain [12]	DMA FIFO Remain		
				11: DMA_Remain [11]			
				10: DMA_Remain [10]			
				9: DMA_Remain [9]			
				8: DMA_Remain [8]			
				7: DMA_Remain [7]			
				6: DMA_Remain [6]			
				5: DMA_Remain [5]			
				4: DMA_Remain [4]			
				3: DMA_Remain [3]			
				2: DMA_Remain [2]			
				1: DMA_Remain [1]			
				0: DMA_Remain [0]			

Bit15-13 Reserved**Bit12-0** DMA_Remain [12:0]

読み出しの場合、AREAn{n=0-5}Join_0.JoinDMA ビットによって DMA に接続している FIFO の残りデータ数を示します。

書き込みの場合、AREAn{n=0-5}Join_0.JoinDMA ビットによって DMA に接続している FIFO の空き容量を示します。

DMA 書き込みを行った直後にはこのレジスタで正しい FIFO の空き容量を参照する事が出来ません。1CPU サイクル以上の間隔を空けて FIFO の空き容量を確認してください。

3. レジスタ詳細説明

3.1.71 067h DMA_EdgeMode (DMA Edge Mode)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	067h	DMA_EdgeMode	R/W	7:EdgeSenseEn	0: Disable	1: Enable	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R/W	3:NgtControl[3]	NgtControl		
			R/W	2: NgtControl[2]			
			R/W	1: NgtControl[1]			
			R/W	0: NgtControl[0]			

DMA 転送における Edge Sense Mode(Single Word DMA)の設定を行います。

Bit7 EdgeSenseEn

DMA 転送において、Edge Sense Mode を有効にします。

0 : Edge Sense Mode 無効

1 : Edge Sense Mode 有効

Edge Sense Mode が有効となった場合、1 ワード転送する毎に IC の XDREQ 信号がネゲートされます。

DMA_Config.ReqAssertCount 以外の DMA 設定と本レジスタを同時に設定し、共用することが可能です。

DMA_Config.ReqAssertCount と同時に設定した場合、ReqAssertCount の設定値を無視して、1 ワード転送毎に、

XDREQ 信号をネゲートします。

Bit6-4 Reserved

Bit3-0 NgtCount[3:0]

Edge Sense Mode が有効な場合に、XDREQ の最小ネゲート期間を調整するために使用します。

本レジスタは DMA_EdgeMode.EdgeSenseEn が設定されている場合のみ有効です。

XDREQ 最小ネゲート期間は(本レジスタビット+3)×16.6ns です。

3.1.72 068h-069h DMA_Count_HH,HL (DMA Transfer Byte Counter High/High, High/Low)**3.1.73 06Ah-06Bh DMA_Count_LH,LL (DMA Transfer Byte Counter Low/High, Low/Low)**

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	068h-069h	DMA_Count_HH,HL	R/W	15: DMA_Count [31]	DMA Transfer Byte Counter High	0000h
				14: DMA_Count [30]		
				13: DMA_Count [29]		
				12: DMA_Count [28]		
				11: DMA_Count [27]		
				10: DMA_Count [26]		
				9: DMA_Count [25]		
				8: DMA_Count [24]		
				7: DMA_Count [23]		
				6: DMA_Count [22]		
				5: DMA_Count [21]		
				4: DMA_Count [20]		
				3: DMA_Count [19]		
				2: DMA_Count [18]		
				1: DMA_Count [17]		
				0: DMA_Count [16]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	06Ah-06Bh	DMA_Count_LH,LL	R/W	15: DMA_Count [15]	DMA Transfer Byte Counter Low	0000h
				14: DMA_Count [14]		
				13: DMA_Count [13]		
				12: DMA_Count [12]		
				11: DMA_Count [11]		
				10: DMA_Count [10]		
				9: DMA_Count [9]		
				8: DMA_Count [8]		
				7: DMA_Count [7]		
				6: DMA_Count [6]		
				5: DMA_Count [5]		
				4: DMA_Count [4]		
				3: DMA_Count [3]		
				2: DMA_Count [2]		
				1: DMA_Count [1]		
				0: DMA_Count [0]		

カウントモード時に DMA の、転送データ長をバイト単位で設定します。最大 0xFFFF_FFFF バイトまで設定可能です。設定された値からダウンカウントします。本レジスタに転送数を設定した後、DMA_Control.DMA_Go ビットに“1”をセットして DMA 転送を起動してください。本レジスタに設定された転送 Byte 数の転送が終了すると、DMA 転送は終了します。フリーランモードの場合、設定された値からカウントアップします。DMA_Count_HH,HL,LH,LL レジスタの値がオーバーフローすると、CPU_IntStat レジスタの DMA_CountUp ビットに“1”をセットします。オーバーフロー後もカウントは継続されます。このモードでは、DMA の転送数が参照できます。

3. レジスタ詳細説明

DMA 書き込みを行った直後は、このレジスタで正確なカウント数を確認することは出来ません。1CPU サイクル以上の間隔をあけて、カウント数を確認してください。このレジスタをリードする場合は DMA_Count_HH,HL が先、DMA_Count_LH,LL が後の順に読み出してください。

3.1.74 06Ch DMA_RdData_0 (DMA Read Data 0)**3.1.75 06Dh DMA_RdData_1 (DMA Read Data 1)**

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	06Ch	DMA_RdData_0	R	7: DMA_RdData_0 [7]	DMA Read Data 0	XXh
				6: DMA_RdData_0 [6]		
				5: DMA_RdData_0 [5]		
				4: DMA_RdData_0 [4]		
				3: DMA_RdData_0 [3]		
				2: DMA_RdData_0 [2]		
				1: DMA_RdData_0 [1]		
				0: DMA_RdData_0 [0]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	06Dh	DMA_RdData_1	R	7: DMA_RdData_1 [7]	DMA Read Data 1	XXh
				6: DMA_RdData_1 [6]		
				5: DMA_RdData_1 [5]		
				4: DMA_RdData_1 [4]		
				3: DMA_RdData_1 [3]		
				2: DMA_RdData_1 [2]		
				1: DMA_RdData_1 [1]		
				0: DMA_RdData_1 [0]		

06Ch.Bit7-0 DMA_RdData_0[7:0]**06Dh.Bit7-0 DMA_RdData_1[7:0]**

DMA_Config.DMA_Mode ビットが“1”にセットされている場合、このレジスタにアクセスする事で

AREAn{n=0-5}Join_0. JoinDMA ビットによって DMA に接続している FIFO のデータを読み出す事が出来ます。このとき、DMA_Control.Dir ビットは DMA リードに設定されている必要があります。

3. レジスタ詳細説明

3.1.76 06Eh DMA_WrData_0 (DMA Write Data 0)

3.1.77 06Fh DMA_WrData_1 (DMA Write Data 1)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	06Eh	DMA_WrData_0	W	7: DMA_WrData_0[7]	DMA Write Data 0	XXh
				6: DMA_WrData_0[6]		
				5: DMA_WrData_0[5]		
				4: DMA_WrData_0[4]		
				3: DMA_WrData_0[3]		
				2: DMA_WrData_0[2]		
				1: DMA_WrData_0[1]		
				0: DMA_WrData_0[0]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	06Fh	DMA_WrData_1	W	7: DMA_WrData_1[7]	DMA Write Data 1	XXh
				6: DMA_WrData_1[6]		
				5: DMA_WrData_1[5]		
				4: DMA_WrData_1[4]		
				3: DMA_WrData_1[3]		
				2: DMA_WrData_1[2]		
				1: DMA_WrData_1[1]		
				0: DMA_WrData_1[0]		

06Eh.Bit7-0 DMA_WrData_0[7:0]

06Fh.Bit7-0DMA_WrData_1[7:0]

DMA_Config.DMA_Mode ビットが“1”にセットされている場合、このレジスタにアクセスする事で

AREAn{n=0-5}Join_0.JoinDMA ビットによって DMA に接続している FIFO にデータを書き込む事が出来ます。このとき、DMA_Control.Dir ビットは DMA ライトに設定されている必要があります。

3.1.78 071h ModeProtect(Mode Protection)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	071h	ModeProtect	R/W	7: ModeProtect [7]	Mode Protection	56h
				6: ModeProtect [6]		
				5: ModeProtect [5]		
				4: ModeProtect [4]		
				3: ModeProtect [3]		
				2: ModeProtect [2]		
				1: ModeProtect [1]		
				0: ModeProtect [0]		

Bit7-0 ModeProtect [7:0]

CPU_Config レジスタ及び ClkSelect レジスタの値を保護します。このレジスタに 56h を書き込むと CPU_Config レジスタ及び ClkSelect レジスタへのライトアクセスが有効になります。

通常使用においては、CPU_Config レジスタ及び ClkSelect レジスタを任意に設定した後、このレジスタに 56h 以外の値 (例えば 00h) を設定して、CPU_Config レジスタ及び ClkSelect レジスタの設定を保護して下さい。

このビットは、SLEEP 中もアクセス有効です。

3. レジスタ詳細説明

3.1.79 073h ClkSelect (Clock Select)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	073h	ClkSelect	R/W	7: ClkSource	0: Xtal	1: CLKIN	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1: ClkFreq[1]	Input clock frequency 00:12MHz, 01: 24MHz, 11: 48MHz		
			R/W	0: ClkFreq[0]			

本 LSI のクロックの初期設定を行います。

本 LSI を動作させる前に、必ず本レジスタの設定を行ってください。

なお、このレジスタは SLEEP 中も有効です。

Bit7 ClkSource

本 LSI にて使用するクロックを設定します。

ClkSource と ClkFreq の関係を以下の表に示します。

CPU_Cut モード時、CLKIN 端子の入力初段も他の CPU インタフェイス端子と同様に初段から OFF しています。

Bit6-2 Reserved

Bit1-0 ClkFreq [1:0]

本 LSI にて使用するクロックの周波数を設定します。

ClkFreq と Cl k Source の関係を以下の表に示します。

		ClkSource	
		0: 水晶振動子	1: CLKIN 端子
00	12MHz	○	○
01	24MHz	○	○
11	48MHz	×	○

○ ご使用になれます。

× ご使用になれません。

3.1.80 075h CPU_Config (CPU Configuration)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	075h	CPU_Config	R/W	7: IntLevel	0: Low Active	1: High Active	01h
			R/W	6: IntMode	0: 1/0 mode	1: Hi-z/0 mode	
			R/W	5: DREQ_Level	0: Low Active	1: High Active	
			R/W	4: DACK_Level	0: Low Active	1: High Active	
			R/W	3: CS_Mode	0: DACK mode	1: CS mode	
			R	2: CPU_Endian	0: Do nothing	1: Bus Swap	
			R	1: BusMode	0: XWRH/L mode	1: XBEH/L mode	
			R	0: Iniialized	0: N/A	1: Initialized	

本 LSI の動作モードを設定します。

このビットは、SLEEP 中もアクセス有効です。

Bit7 IntLevel

XINT の論理レベルを設定します。

0 : 負論理

1 : 正論理

Bit6 IntMode

XINT の出力モードを設定します。

0 : 1/0 モード

1 : Hi-z/0 モード

Bit5 DREQ_Level

XDREQ の論理レベルを設定します。

0 : 負論理

1 : 正論理

Bit4 DACK_Level

XDACK の論理レベルを設定します。

0 : 負論理

1 : 正論理

Bit3 CS_Mode

DMA の動作モードを設定します。

0 : XDACK がアサートされているとき有効な DMA アクセスとして動作します。

1 : XCS 且つ XDACK がアサートされているとき有効な DMA アクセスとして動作します。

3. レジスタ詳細説明

Bit2 ***CPU_Endian***

CPU バスモードを参照できます。

0：偶数アドレスを上位側、奇数アドレスを下位側とします。

1：偶数アドレスを下位側、奇数アドレスを上位側とします。

このビットには、初期化期間に CPUIF_MODE レジスタに書き込んだ値が反映されます。

Bit1 ***BusMode***

CPU の動作モードを参照できます。

0: 16bit Strobe mode

1: 16bit BE mode

このビットには、初期化期間に CPUIF_MODE レジスタに書き込んだ値が反映されます。

Bit0 ***Initialized***

初期化期間が終了していることを示します。通常“1”が読めます。

3.1.81 07E-07Fh CPUIF_MODE (CPUIF Mode)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	07Eh-07Fh	CPUIF_MODE		15:	0:	1:	XXh
				14:	0:	1:	
				13:	0:	1:	
				12:	0:	1:	
				11:	0:	1:	
			W	10: CPU_Endian	0: Big Endian	1: Little Endian	
			W	9: BusMode	0: Strobe mode	1: BE mode	
				8:	0:	1:	
				7:	0:	1:	
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			W	2: CPU_Endian	0: Big Endian	1: Little Endian	
			W	1: BusMode	0: Strobe mode	1: BE mode	
				0:	0:	1:	

このレジスタは、未初期化期間にのみアクセス可能です。

本 LSI の CPUIF の動作モードを設定します。

本 LSI のハードリセット後に、必ず本レジスタの設定を行って下さい。このレジスタの設定方法は、「1.6.2 CPUIF のモード設定」を参照して下さい。

偶数バイトと奇数バイトで、同じレジスタビットが定義されていますが、どちらか一方、または、両方に“1”が書き込まれると、セットされます。

Bit10,2 CPU_Endian

CPUIF のエンディアンを設定します。

Bit9,1 BusMode

CPUIF のライトアクセスのモードを設定します。

Bit15-11,8-3,0 Reserved

3. レジスタ詳細説明

3.1.82 080h-081h AREA0StartAdrs_H,L (AREA 0 Start Address High, Low)

3.1.83 084h-085h AREA1StartAdrs_H,L (AREA 1 Start Address High, Low)

3.1.84 088h-089h AREA2StartAdrs_H,L (AREA 2 Start Address High, Low)

3.1.85 08Ch-08Dh AREA3StartAdrs_H,L (AREA 3 Start Address High, Low)

3.1.86 090h-091h AREA4StartAdrs_H,L (AREA 4 Start Address High, Low)

3.1.87 094h-095h AREA5StartAdrs_H,L (AREA 5 Start Address High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device / Host	080h-081h	AREA0StartAdrs_H,L		15:	0:	1:
	084h-085h	AREA1StartAdrs_H,L		14:	0:	1:
	088h-089h	AREA2StartAdrs_H,L		13:	0:	1:
	08Ch-08Dh	AREA3StartAdrs_H,L	R/W	12: StartAdrs[12]	AREAx{x=0-5} Start Address	0000h
	090h-091h	AREA4StartAdrs_H,L		11: StartAdrs[11]		
	094h-095h	AREA5StartAdrs_H,L		10: StartAdrs[10]		
				9: StartAdrs[9]		
				8: StartAdrs[8]		
				7: StartAdrs[7]		
				6: StartAdrs[6]		
				5: StartAdrs[5]		
				4: StartAdrs[4]		
				3: StartAdrs[3]		
				2: StartAdrs[2]		
				1:		
				0:		

AREAx{x=0-5}で使用する FIFO の領域設定を行います。

Bit15-13 Reserved

Bit12 -2 StartAdrs[12:2]

FIFO 領域 AREAx{x=0-5}に割り当てる FIFO の先頭アドレスを設定します。

アドレス値は、上位 12 ビット～2 ビットでの設定のため 4 バイト単位での指定になります。

FIFO 領域 AREAx{x=0-5}に割り当てられる領域は、AREAx{x=0-5}EndAdrs にて設定されたアドレスの 1 バイト前までとなります。

AREAx{x=0-5}StartAdrs、AREAx{x=0-5}EndAdrs を設定した後は、必ず AREAnFIFO_Clr レジスタの ClrAREAx{x=0-5}ビットに“1”とセットして FIFO 領域 AREAx{x=0-5}の FIFO をクリアして下さい。

なお、ここで設定した領域より、ジョインした USB のデバイス/ホストの MaxSize が大きい場合には、正常に動作しません。また、FIFO 領域が他の FIFO 領域と重なった場合も、正常に動作しません。

本 LSI の内蔵 RAM は 4.5kB ですので、最大 0x1200 番地のアドレスをサポートしています。

Bit1-0 Reserved

3.1.88 082h-083h AREA0EndAdrs_H,L (AREA 0 End Address High, Low)

3.1.89 086h-087h AREA1EndAdrs_H,L (AREA 1 End Address High, Low)

3.1.90 08Ah-08Bh AREA2EndAdrs_H,L (AREA 2 End Address High, Low)

3.1.91 08Eh-08Fh AREA3EndAdrs_H,L (AREA 3 End Address High, Low)

3.1.92 092h-093h AREA4EndAdrs_H,L (AREA 4 End Address High, Low)

3.1.93 096h-097h AREA5EndAdrs_H,L (AREA 5 End Address High,Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	082h-083h	AREA0EndAdrs_H,L		15:	0:	1:	0000h
	086h-087h	AREA1EndAdrs_H,L		14:	0:	1:	
	08Ah-08Bh	AREA2EndAdrs_H,L		13:	0:	1:	
	08Eh-08Fh	AREA3EndAdrs_H,L	R/W	12: EndAdrs[12]	AREAx{x=0-5} End Address		
	092h-093h	AREA4EndAdrs_H,L		11: EndAdrs[11]			
	096h-097h	AREA5EndAdrs_H,L		10: EndAdrs[10]			
				9: EndAdrs[9]			
				8: EndAdrs[8]			
				7: EndAdrs[7]			
				6: EndAdrs[6]			
				5: EndAdrs[5]			
				4: EndAdrs[4]			
				3: EndAdrs[3]			
				2: EndAdrs[2]			
			1:				
			0:				

AREAx{x=0-5}で使用する FIFO の領域設定を行います。

Bit15-13 Reserved

Bit12 -2 EndAdrs[12:2]

FIFO 領域 AREAx{x=0-5}に割り当てる FIFO の最終アドレスの次のバイトを設定します。

アドレス値は、上位 12 ビット～2 ビットでの設定のため 4 バイト単位での指定になります。

FIFO 領域 AREAx{x=0-5}に割り当てられる領域は、AREAx{x=0-5}EndAdrs にて設定されたアドレスの 1 バイト前までとなります。

AREAx{x=0-5}StartAdrs、AREAx{x=0-5}EndAdrs を設定した後は、必ず AREAnFIFO_Clr レジスタの ClrAREAx{x=0-5}ビットに“1”とセットして FIFO 領域 AREAx{x=0-5}の FIFO をクリアして下さい。

なお、ここで設定した領域より、ジョインした USB のデバイス/ホストの MaxSize が大きい場合には、正常に動作しません。また、FIFO 領域が他の FIFO 領域と重なった場合も、正常に動作しません。

本 LSI の内蔵 RAM は 4.5kB です。最大 0x1200 番地のアドレスをサポートしています。

Bit1-0 Reserved

3. レジスタ詳細説明

3.1.94 09Fh AREAnFIFO_Clr (AREA n FIFO Clear)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	09Fh	AREAnFIFO_Clr		7:	0:	1:	XXh
				6:	0:	1:	
			W	5: AREA5FIFO_Clr	0: Do nothing	1: Clear AREA5 FIFO	
			W	4: AREA4FIFO_Clr	0: Do nothing	1: Clear AREA4 FIFO	
			W	3: AREA3FIFO_Clr	0: Do nothing	1: Clear AREA3 FIFO	
			W	2: AREA2FIFO_Clr	0: Do nothing	1: Clear AREA2 FIFO	
			W	1: AREA1FIFO_Clr	0: Do nothing	1: Clear AREA1 FIFO	
			W	0: AREA0FIFO_Clr	0: Do nothing	1: Clear AREA0FIFO	

該当する FIFO 領域 AREAn{n=0-5}の FIFO をクリアします。ライトオンリーのレジスタです。

このレジスタの各ビットは、“1”をセットされると FIFO をクリアする動作のみ行い、セットされた値は保持しません。

FIFO 領域 AREAn{n=0-5}に DMA がジョインされ、かつ、該当する DMA が起動中(DMA_Running ビットが“1”の間)に、該当するエンドポイントのビットを“1”にセットしないで下さい。

また、このレジスタは、データ保持情報の初期化動作のみを行い、データ自身を書き込みまたはクリアをしません。従って、このビットにより RAM 上のデータがクリアされることはありません。

3.1.95 0A0h AREA0Join_0 (AREA 0 Join 0)**3.1.96 0A2h AREA1Join_0 (AREA 1 Join 0)****3.1.97 0A4h AREA2Join_0 (AREA 2 Join 0)****3.1.98 0A6h AREA3Join_0 (AREA 3 Join 0)****3.1.99 0A8h AREA4Join_0 (AREA 4 Join 0)****3.1.100 0AAh AREA5Join_0 (AREA 5 Join 0)**

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	0A0h	AREA0Join_0	R/W	7: JoinFIFO_Stat	0: Do nothing	1: Join to FIFO Status	00h
	0A2h	AREA1Join_0		6:	0:	1:	
	0A4h	AREA2Join_0		5:	0:	1:	
	0A6h	AREA3Join_0		4:	0:	1:	
	0A8h	AREA4Join_0		3:	0:	1:	
	0AAh	AREA5Join_0	R/W	2: JoinDMA	0: Do nothing	1: Join to DMA	
			R/W	1: JoinCPU_Rd	0: Do nothing	1: Join to CPU Read	
			R/W	0: JoinCPU_Wr	0: Do nothing	1: Join to CPU Write	

FIFO 領域 AREA_x{x=0-5}に接続するポートを設定します。

Bit7 JoinFIFO_Stat

FIFO 領域 AREA_x{x=0-5}の FIFO の Full 及び Empty の状態を、FIFO_IntStat.FIFO_NotEmpty、FIFO_IntStat.FIFO_Full 及び FIFO_IntStat.FIFO_Empty でモニタできるようにします。

Bit6-3 Reserved**Bit2 JoinDMA**

FIFO 領域 AREA_x{x=0-5}の FIFO で DMA の転送を行います。転送の方向は、DMA_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

FIFO 領域 AREA_x{x=0-5}の FIFO で CPU レジスタアクセスのリード転送を行います。即ち FIFO_Rd_0,1 レジスタ、または、FIFO_ByteRd レジスタのリードが行われると、この FIFO 領域からデータが読み出されます。

Bit0 JoinCPU_Wr

FIFO 領域 AREA_x{x=0-5}の FIFO で CPU レジスタアクセスのライト転送を行います。即ち FIFO_Wr_0,1 レジスタへのライトが行われると、この FIFO 領域にデータが書き込まれます。

JoinDMA ビットを設定した場合は、DMA_Control.Dir ビットが 1 のときは残りデータ数、0 のときは空き容量が、DMA_Remain_H,L レジスタにより、それぞれ参照できます。

JoinCPU_Rd, JoinCPU_Wr ビットを設定した場合は、FIFO_RdRemain_H,L、FIFO_WrRemain_H,L を参照し、FIFO_Rd_0,1、FIFO_ByteRd、FIFO_Wr_0,1 レジスタからデータを読み出し、または書き込みできます。

JoinDMA ビット、JoinCPU_Rd ビット、JoinCPU_Wr ビットは、同時に 1 ビットのみを“1”にセットしてください。複数のビットに対して、同時に“1”を書きこんだ場合、動作が不安定になる恐れがあります。

3. レジスタ詳細説明

3.1.101 0A1h AREA0Join_1 (AREA 0 Join 1)

3.1.102 0A3h AREA1Join_1 (AREA 1 Join 1)

3.1.103 0A5h AREA2Join_1 (AREA 2 Join 1)

3.1.104 0A7h AREA3Join_1 (AREA 3 Join 1)

3.1.105 0A9h AREA4Join_1 (AREA 4 Join 1)

3.1.106 0ABh AREA5Join_1 (AREA 5 Join 1)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	0A1h	AREA0Join_1		7:	0:	1:	00h
	0A3h	AREA1Join_1		6:	0:	1:	
	0A5h	AREA2Join_1	R/W	5: JoinEPeCHe	0: Do nothing	1: Join to EPe / CHe	
	0A7h	AREA3Join_1	R/W	4: JoinEPdCHd	0: Do nothing	1: Join to EPd / CHd	
	0A9h	AREA4Join_1	R/W	3: JoinEPcCHc	0: Do nothing	1: Join to EPc / CHc	
	0ABh	AREA5Join_1	R/W	2: JoinEPbCHb	0: Do nothing	1: Join to EPb / CHb	
			R/W	1: JoinEPaCHa	0: Do nothing	1: Join to EPa / CHa	
			R/W	0: JoinEP0CH0	0: Do nothing	1: Join to EP0 / CH0	

FIFO 領域 AREA_x{x=0-5}に接続するエンドポイント及びチャネルを設定します。

Bit7-6 **Reserved**

Bit5 **JoinEPeCHe**

FIFO 領域 AREA_x{x=0-5}にエンドポイント EPe もしくはチャネル CHe を接続します。接続することで、エンドポイント EPe、チャネル EPe を使用して、データ転送を伴うトランザクションを実行可能となります。

Bit4 **JoinEPdCHd**

FIFO 領域 AREA_x{x=0-5}にエンドポイント EPd もしくはチャネル CHd を接続します。接続することで、エンドポイント EPd、チャネル EPd を使用して、データ転送を伴うトランザクションを実行可能となります。

Bit3 **JoinEPcCHc**

FIFO 領域 AREA_x{x=0-5}にエンドポイント EPc もしくはチャネル CHc を接続します。接続することで、エンドポイント EPc、チャネル EPc を使用して、データ転送を伴うトランザクションを実行可能となります。

Bit2 **JoinEPbCHb**

FIFO 領域 AREA_x{x=0-5}にエンドポイント EPb もしくはチャネル CHb を接続します。接続することで、エンドポイント EPb、チャネル EPb を使用して、データ転送を伴うトランザクションを実行可能となります。

Bit1 **JoinEPaCHa**

FIFO 領域 AREA_x{x=0-5}にエンドポイント EPa もしくはチャネル CHa を接続します。接続することで、エンドポイント EPa、チャネル EPa を使用して、データ転送を伴うトランザクションを実行可能となります。

Bit0 **JoinEP0CH0**

FIFO 領域 AREA_x{x=0-5}にエンドポイント EP0 もしくはチャネル CH0 を接続します。接続することで、エンドポイント EP0、チャネル EP0 を使用して、データ転送を伴うトランザクションを実行可能となります。

同一の FIFO 領域に JoinEPxCHx{x=0,a-e} ビットを同時に複数設定した場合には、トランザクション順によっては、想定外の動作を行うことがありますので、ご注意願います。基本的には、同一の FIFO 領域に JoinEPxCHx{x=0,a-e} ビットを設定行わないことを推奨致します。

3. レジスタ詳細説明

3.1.107 0AEh ClrAREAnJoin_0 (Clear AREA n Join 0)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	0AEh	ClrAREAnJoin_0	W	7: ClrJoinFIFO_Stat	0: Do nothing	1: Clear JoinFIFO_Stat	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			W	2: ClrJoinDMA	0: Do nothing	1: Clear JoinDMA	
			W	1: ClrJoinCPU_Rd	0: Do nothing	1: Clear JoinCPU_Rd	
			W	0: ClrJoinCPU_Wr	0: Do nothing	1: Clear JoinCPU_Wr	

各 FIFO 領域と該当するポートの接続をクリアします。ライトオンリーのレジスタです。

このレジスタのビットは、接続クリア後、自動的に“0”にクリアされます。

FIFO 領域がポートに接続（AREAn{n=0-5}Join_0 レジスタの該当するビットが“1”にセット）され、且つ各ポートの起動中に、このレジスタのビットを“1”にセットしないで下さい。誤動作の原因となります。

3.1.108 0AFh ClrAREAnJoin_1 (Clear AREA n Join 1)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device / Host	0AFh	ClrAREAnJoin_1		7:	0:	1:	00h
				6:	0:	1:	
			W	5: Clr JoinEPeCHe	0: Do nothing	1: Clear JoinEPeCHe	
			W	4: ClrJoinEPdCHd	0: Do nothing	1: Clear JoinEPdCHd	
			W	3: ClrJoinEPcCHc	0: Do nothing	1: Clear JoinEPcCHc	
			W	2: ClrJoinEPbCHb	0: Do nothing	1: Clear JoinEPbCHb	
			W	1: ClrJoinEPaCHa	0: Do nothing	1: Clear JoinEPaCHa	
			W	0: ClrJoinEP0CH0	0: Do nothing	1: Clear JoinEP0CH0	

各 FIFO 領域と該当するエンドポイント及びチャネルの接続をクリアします。ライトオンリーのレジスタです。

このレジスタのビットは、接続クリア後、自動的に“0”にクリアされます。

FIFO 領域がエンドポイント及びチャネルに接続 (AREAn{n=0-5}Join_1 レジスタの該当するビットが“1”にセット) され、且つ各エンドポイント及びチャネルのトランザクションが実行されている時に、このレジスタのビットを“1”にセットしないで下さい。誤動作の原因となります。

3. レジスタ詳細説明

3.2 デバイスレジスタ詳細説明

3.2.1 0B0h D_SIE_IntStat (Device SIE Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device	0B0h	D_SIE_IntStat		7:	0:	1:
			R (W)	6: NonJ	0: None	1: Detect Non J state
			R (W)	5: RcvSOF	0: None	1: Received SOF
			R (W)	4: DetectReset	0: None	1: Detect USB Reset
			R (W)	3: DetectSuspend	0: None	1: Detect USB Suspend
			R (W)	2: ChirpCmp	0: None	1: Chirp Complete
			R (W)	1: RestoreCmp	0: None	1: Restore Complete
			R (W)	0: SetAddressCmp	0: None	1: AutoSetAddress Complete

デバイス SIE 関連の割り込みを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 **Reserved**

Bit6 **NonJ**

割り込み要因を直接指示します。

USB バス上で J ステート以外の状態を検出すると“1”にセットされます。このビットは、本 LSI が SNOOZE 状態 (PM_Control レジスタの InSnooze ビットが“1”) の時、及び AutoNegotiation 機能使用時に USB_Control レジスタの InSUSPEND ビットが“1”にセットされている時に有効です。

Bit5 **RcvSOF**

割り込み要因を直接指示します。

SOF トークンを受信すると“1”にセットされます。

Bit4 **DetectReset**

割り込み要因を直接指示します。

USB のリセットステートを検出すると“1”にセットされます。このビットがセットされている間は USB のサスペンドステートの検出ができません (DetectSUSPEND がセットされません)。

このリセット検出は、D_NegoControl レジスタの ActiveUSB ビットが “1” にセットされている時に有効です。

"HS"動作モードの場合は、バス・アクティビティが一定時間無くなると、USB のリセット／サスペンド検出のために FS ターミネーションを自動的に設定し、SE0 が検出されるとリセットと判断して、このビットが“1”にセットされます。

AutoNegotiation 機能を使用しない場合には、このビットが“1”にセットされた場合、継続するリセットを誤検出しないよう、D_NegoControl レジスタの DisBusDetect ビットを“1”にセットして USB のリセット／サスペンドステートの検出を無効にしてください。リセットに対する処理終了後に DisBusDetect ビットを“0”にクリアして USB のリセット／サスペンドステートの検出を有効にしてください。

リセット検出時、D_NegoControl レジスタの GoChirp ビットにより、“HS Detection Handshake”を開始することができます。

AutoNegotiation 機能については、D_NegoControl レジスタの EnAutoNego ビットの項を参照してください。

Bit3 DetectSuspend

割り込み要因を直接指示します。

USB のサスペンドステートを検出すると “1” にセットされます。このビットがセットされている間は USB のリセットステートの検出ができません (DetectRESET がセットされません)。

“HS” 動作モード の場合は、バス・アクティビティが一定時間無くなると、USB のリセット／サスペンド検出のために“FS”動作モードに自動的に設定されます。USB のサスペンドステートの検出後は、PM_Control.GoSLEEP ビットを “1” にセットすることにより、本 LSI をスリープモード(内蔵 PLL 及びオシレータの発振を停止)にすることが出来ます。

Bit2 ChirpCmp

割り込み要因を直接指示します。

D_NegoControl レジスタの GoChirp ビットにより開始された“HS Detection Handshake”が完了すると“1” にセットされます。

割り込み発生後に D_USB_Status レジスタの FSxHS ビットをリードすることで、現在の動作モード(FS or HS)の判定をすることができます。

Bit1 RestoreCmp

割り込み要因を直接指示します。

D_NegoControl レジスタの RestoreUSB ビットにより開始された Restore 処理が終了すると“1”にセットされます。このビットが“1” にセットされると動作モード(FS or HS)が Suspend する前の状態に戻ります。

Bit0 SetAddressCmp

割り込み要因を直接指示します。

SetAddress() リクエストを受信すると、AutoSetAddress 機能(USB_Address レジスタ参照)が、そのコントロール転送の処理を自動的に行います。ステータスステージを行って SetAddress() リクエストに関わるコントロール転送が完了した時に、このステータスが“1” にセットされます。また、同時に D_USB_Address レジスタにアドレスがセットされます。

同期ビット (Bit5～0) は、パワーマネージメントが ACTIVE ステートであっても、HostDeviceSel.HOST x DEVICE ビットが“0”、すなわち DEVICE モードでなければ、読み出し、書き込み (割り込み要因クリア) ができません。したがって、この状態から移行する場合には、これらの割り込みステータスにより割り込み信号 XINT がアサートされないよう、F/W にて以下の処理を行って下さい。

＜ACTIVE で DEVICE モードから移行する時＞

- 1) 割り込みステータスを処理し、クリアする (D_SIE_IntStat.Bit5～0)
- 2) 割り込みステータスをディスエーブルにする (D_SIE_IntEnb.Bit5～0)

＜ACTIVE で DEVICE モードに移行する時＞

- 3) 割り込みステータスをクリアする (D_SIE_IntStat.Bit5～0)
- 4) 割り込みステータスをイネーブルにする (D_SIE_IntEnb.Bit5～0)

3. レジスタ詳細説明

3.2.2 0B3h D_BulkIntStat (Device Bulk Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0B3h	D_BulkIntStat	R (W)	7: CBW_Cmp	0: None	1: CBW Complete	00h
			R (W)	6: CBW_LengthErr	0: None	1: CBW Length Error	
			R (W)	5: CBW_Err	0: None	1: CBW Transaction Error	
			R (W)	4:	0:	1:	
			R (W)	3: CSW_Cmp	0: None	1: CSW Complete	
			R (W)	2: CSW_Err	0: None	1: CSW Error	
				1:	0:	1:	
				0:	0:	1:	

Bulk 転送機能関連の割り込みステータスを表示します。全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 CBW_Comp

割り込み要因を直接指示します。

CBW の 31 バイトを正常に受信できた時に“1”にセットされます。

Bit6 CBW_LengthErr

割り込み要因を直接指示します。

受信した CBW のパケット長が 31 バイト以外であった時に“1”にセットされます。

Bit5 CBW_Err

割り込み要因を直接指示します。

受信した CBW に CRC エラー等のトランザクションエラーを検出した時に“1”にセットされます。

Bit4 Reserved

Bit3 CSW_Cmp

割り込み要因を直接指示します。

CSW の 13 バイトを正常に送信できた時に“1”にセットされます。

Bit2 CSW_Err

割り込み要因を直接指示します。

CSW の送信にエラーがあった時（ACK が返ってこなかった時）に“1”にセットされます。

Bit1-0 Reserved

3.2.3 0B4h D_EPrIntStat (Device EPr Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0B4h	D_EPrIntStat	R	7: D_AlarmIN_IntStat	0: None	1: Alarm IN Interrupt	00h
			R	6: D_AlarmOUT_IntStat	0: None	1: Alarm OUT Interrupt	
				5:	0:	1:	
			R	4: D_EPeIntStat	0: None	1: EPe Interrupt	
			R	3: D_EPdIntStat	0: None	1: EPd Interrupt	
			R	2: D_EPcIntStat	0: None	1: EPc Interrupt	
			R	1: D_EPbIntStat	0: None	1: EPb Interrupt	
			R	0: D_EPaIntStat	0: None	1: EPa Interrupt	

エンドポイント EPr{r=a-e}、及び AlarmIN/AlarmOUT の割り込みを表示します。

Bit7 D_AlarmIN_IntStat

割り込み要因を間接指示します。

D_AlarmIN_IntStat_H,L レジスタに割り込み要因があり、かつその割り込み要因に対応する D_AlarmIN_IntEnb_H,L レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit6 D_AlarmOUT_IntStat

割り込み要因を間接指示します。

D_AlarmOUT_IntStat_H,L レジスタに割り込み要因があり、かつその割り込み要因に対応する D_AlarmOUT_IntEnb_H,L レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit5 Reserved**Bit4 D_EPeIntStat**

割り込み要因を間接指示します。

D_EPeIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する D_EPeIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit3 D_EPdIntStat

割り込み要因を間接指示します。

D_EPdIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する D_EPdIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit2 D_EPcIntStat

割り込み要因を間接指示します。

D_EPcIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する D_EPcIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit1 D_EPbIntStat

割り込み要因を間接指示します。

D_EPbIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する D_EPbIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

3. レジスタ詳細説明

Bit0

D_EPaIntStat

割り込み要因を間接指示します。

D_EPaIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する

D_EPaIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

3.2.4 0B5h D_EP0IntStat (Device EP0 Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0B5h	D_EP0IntStat	R (W)	7: DescriptorCmp	0: None	1: Descriptor Complete	00h
			R (W)	6: OUT_ShortACK	0: None	1: OUT Short-Packet ACK	
			R (W)	5: IN_TranACK	0: None	1: IN Transaction ACK	
			R (W)	4: OUT_TranACK	0: None	1: OUT Transaction ACK	
			R (W)	3: IN_TranNAK	0: None	1: IN Transaction NAK	
			R (W)	2: OUT_TranNAK	0: None	1: OUT Transaction NAK	
			R (W)	1: IN_TranErr	0: None	1: IN Transaction Error	
			R (W)	0: OUT_TranErr	0: None	1: OUT Transaction Error	

エンドポイント EP0 の割り込みステータスを表示します。全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 DescriptorCmp

割り込み要因を直接指示します。

Descriptor 返信機能において、DescriptorSize レジスタの設定数のデータを返信し終わると、“1”にセットされます。

また、DescriptorSize レジスタの設定数まで送信する前にステータスステージへ移行（OUT トークンを受信）した場合には OUT_TranNAK ビットと共に、“1”にセットされます。

Bit6 OUT_ShortACK

割り込み要因を直接指示します。

OUT トランザクションでショートパケットを受信し、ACK を返信した時、OUT_TranACK と同時に“1”にセットされます。

Bit5 IN_TranACK

割り込み要因を直接指示します。

IN トランザクションで ACK を受信した時、“1”にセットされます。

Bit4 OUT_TranACK

割り込み要因を直接指示します。

OUT トランザクションで ACK を返信した時、“1”にセットされます。

Bit3 IN_TranNAK

割り込み要因を直接指示します。

IN トランザクションで NAK を返信した時、“1”にセットされます。

Bit2 OUT_TranNAK

割り込み要因を直接指示します。

OUT トランザクション及びPING トランザクションに対して NAK を返信した時、“1”にセットされます。

Bit1 IN_TranErr

割り込み要因を直接指示します。

IN トランザクションにおいて STALL を返した場合、パケットにエラーがあった場合、及びハンドシェイクがタイムアウトになった場合に、“1”にセットされます。

3. レジスタ詳細説明

Bit0

OUT_TrnErr

割り込み要因を直接指示します。

OUT トランザクションにおいて **STALL** を返信した場合、及び、パケットにエラーがあった場合に、“1”にセットされます。

3.2.5 0B6h D_EPaIntStat (Device EPa Interrupt Status)**3.2.6 0B7h D_EPbIntStat (Device EPb Interrupt Status)****3.2.7 0B8h D_EPCIntStat (Device EPc Interrupt Status)****3.2.8 0B9h D_EPdIntStat (Device EPd Interrupt Status)****3.2.9 0BAh D_EPeIntStat (Device EPe Interrupt Status)**

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0B6h	D_EPaIntStat		7:	0:	1:	00h
	0B7h	D_EPbIntStat	R (W)	6: OUT_ShortACK	0: None	1: OUT Short Packet ACK	
	0B8h	D_EPCIntStat	R (W)	5: IN_TranACK	0: None	1: IN Transaction ACK	
	0B9h	D_EPdIntStat	R (W)	4: OUT_TranACK	0: None	1: OUT Transaction ACK	
	0BAh	D_EPeIntStat	R (W)	3: IN_TranNAK	0: None	1: IN Transaction NAK	
			R (W)	2: OUT_TranNAK	0: None	1: OUT Transaction NAK	
			R (W)	1: IN_TranErr	0: None	1: IN Transaction Error	
			R (W)	0: OUT_TranErr	0: None	1: OUT Transaction Error	

エンドポイント EPx{x=a-e}の割り込みステータスを表示します。全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 **Reserved**

Bit6 **OUT_ShortACK**

割り込み要因を直接指示します。

OUT トランザクションでショートパケットを受信し、ACK を返信した時、OUT_TranACK と同時に“1”にセットされます。

Bit5 **IN_TranACK**

割り込み要因を直接指示します。

IN トランザクションで ACK を受信した時、“1”にセットされます。

Bit4 **OUT_TranACK**

割り込み要因を直接指示します。

OUT トランザクションで ACK を返信した時、“1”にセットされます。

Bit3 **IN_TranNAK**

割り込み要因を直接指示します。

IN トランザクションで NAK を返信した時、“1”にセットされます。

Bit2 **OUT_TranNAK**

割り込み要因を直接指示します。

OUT トランザクション及びPING トランザクションに対して NAK を返信した時、“1”にセットされます。

Bit1 **IN_TranErr**

割り込み要因を直接指示します。

IN トランザクションにおいて STALL を返した場合、パケットにエラーがあった場合、及びハンドシェイクがタイムアウトになった場合に、“1”にセットされます。

3. レジスタ詳細説明

Bit0

OUT_TrnErr

割り込み要因を直接指示します。

OUT トランザクションにおいて **STALL** を返信した場合、及び、パケットにエラーがあった場合に、“1”にセットされます。

3.2.10 0BCh-0BDh D_AlarmIN_IntStat_H,L (Device AlarmIN Interrupt Status High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0BCh-0BDh	D_AlarmIN _IntStat_H,L	R (W)	15: AlarmEP15IN	0: Do nothing	1: EP15 received IN token	0000h
			R (W)	14: AlarmEP14IN	0: Do nothing	1: EP14 received IN token	
			R (W)	13: AlarmEP13IN	0: Do nothing	1: EP13 received IN token	
			R (W)	12: AlarmEP12IN	0: Do nothing	1: EP12 received IN token	
			R (W)	11: AlarmEP11IN	0: Do nothing	1: EP11 received IN token	
			R (W)	10: AlarmEP10IN	0: Do nothing	1: EP10 received IN token	
			R (W)	9: AlarmEP9IN	0: Do nothing	1: EP9 received IN token	
			R (W)	8: AlarmEP8IN	0: Do nothing	1: EP8 received IN token	
			R (W)	7: AlarmEP7IN	0: Do nothing	1: EP7 received IN token	
			R (W)	6: AlarmEP6IN	0: Do nothing	1: EP6 received IN token	
			R (W)	5: AlarmEP5IN	0: Do nothing	1: EP5 received IN token	
			R (W)	4: AlarmEP4IN	0: Do nothing	1: EP4 received IN token	
			R (W)	3: AlarmEP3IN	0: Do nothing	1: EP3 received IN token	
			R (W)	2: AlarmEP2IN	0: Do nothing	1: EP2 received IN token	
			R (W)	1: AlarmEP1IN	0: Do nothing	1: EP1 received IN token	
				0:	0:	1:	

アラーム IN の割り込みステータスを表示します。全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

全てのビットは、割り込み要因を直接指示しています。

D_EnEP_IN_H,L にて有効にしたエンドポイントのうち、D_EPx{x=0,a-e}関連レジスタに適切な設定が行われ、且つ AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットにてジョイン設定しているエンドポイントを除いたエンドポイントに対し、ホストからトランザクションが発行 (IN トークンを受信) された場合、以下に示す応答を行い、該当するビットに“1”をセットします。なお、エンドポイント EP0 は、常に有効であるため、D_EP0 関連レジスタが適切に設定されていなかったり、AREAn{n=0-5}Join.JoinEP0CH0 ビットがどこの FIFO 領域に対しても設定されていない場合においても、同様の動作を行います。

この際、IN トークンに対して行われる応答は、D_EnEP_IN_ISO_H,L の設定によって選択され、該当するビットが“1”にセットされているエンドポイントの場合には、ホストに対しゼロ長パケット応答し、“0”にクリアされているエンドポイントの場合には、ホストに対し NAK 応答します。

本レジスタの該当ビットがセットされた場合には、D_EPx{x=0,a-e}関連レジスタを適切に設定し、AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットを用いてエンドポイントを FIFO 領域にジョインし、トランザクションを実行可能にしてください。

3. レジスタ詳細説明

3.2.11 0BEh-0BFh D_AlarmOUT_IntStat_H,L (Device AlarmOUT Interrupt Status High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0BEh-0BFh	D_AlarmOUT _IntStat_H,L	R (W)	15: AlarmEP15OUT	0: Do nothing	1: EP15 received OUT token	0000h
			R (W)	14: AlarmEP14OUT	0: Do nothing	1: EP14 received OUT token	
			R (W)	13: AlarmEP13OUT	0: Do nothing	1: EP13 received OUT token	
			R (W)	12: AlarmEP12OUT	0: Do nothing	1: EP12 received OUT token	
			R (W)	11: AlarmEP11OUT	0: Do nothing	1: EP11 received OUT token	
			R (W)	10: AlarmEP10OUT	0: Do nothing	1: EP10 received OUT token	
			R (W)	9: AlarmEP9OUT	0: Do nothing	1: EP9 received OUT token	
			R (W)	8: AlarmEP8OUT	0: Do nothing	1: EP8 received OUT token	
			R (W)	7: AlarmEP7OUT	0: Do nothing	1: EP7 received OUT token	
			R (W)	6: AlarmEP6OUT	0: Do nothing	1: EP6 received OUT token	
			R (W)	5: AlarmEP5OUT	0: Do nothing	1: EP5 received OUT token	
			R (W)	4: AlarmEP4OUT	0: Do nothing	1: EP4 received OUT token	
			R (W)	3: AlarmEP3OUT	0: Do nothing	1: EP3 received OUT token	
			R (W)	2: AlarmEP2OUT	0: Do nothing	1: EP2 received OUT token	
			R (W)	1: AlarmEP1OUT	0: Do nothing	1: EP1 received OUT token	
				0:	0:	1:	

アラーム OUT の割り込みステータスを表示します。全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

全てのビットは、割り込み要因を直接指示しています。

D_EnEP_OUT_H,L にて有効にしたエンドポイントのうち、D_EPx{x=0,a-e}関連レジスタに適切な設定が行われ、且つ AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットにてジョイン設定しているエンドポイントを除いたエンドポイントに対し、ホストからトランザクションが発行（OUT トークンを受信）された場合、以下に示す応答を行い、該当するビットに“1”をセットします。なお、エンドポイント EP0 は、常に有効であるため、D_EP0 関連レジスタが適切に設定されていなかったり、AREAn{n=0-5}Join.JoinEP0CH0 ビットがどこの FIFO 領域に対しても設定されていない場合においても、同様の動作を行います。

この際、OUT トークンに対して行われる応答は、D_EnEP_OUT_ISO_H,L の設定によって選択され、該当するビットが“1”にセットされているエンドポイントの場合には、ホストから送出されたデータを受け取らず且つハンドシェイクを応答しません。“0”にクリアされているエンドポイントの場合には、ホストに対し NAK 応答します。デバイスが HS に設定され、ホストから PING トークンが発行された場合には、NAK 応答します。

本レジスタの該当ビットがセットされた場合には、D_EPx{x=0,a-e}関連レジスタを適切に設定し、AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットを用いてエンドポイントを FIFO 領域にジョインし、トランザクションを実行可能にしてください。

3.2.12 0C0h D_SIE_IntEnb (Device SIE Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0C0h	D_SIE_IntEnb		7:	0:	1:	00h
			R/W	6: EnNonJ	0: Disable	1: Enable	
			R/W	5: EnRcvSOF	0: Disable	1: Enable	
			R/W	4: EnDetectRESET	0: Disable	1: Enable	
			R/W	3: EnDetectSUSPEND	0: Disable	1: Enable	
			R/W	2: EnChirpCmp	0: Disable	1: Enable	
			R/W	1: EnRestoreCmp	0: Disable	1: Enable	
			R/W	0: EnSetAddressCmp	0: Disable	1: Enable	

D_SIE_IntStat レジスタの割り込み要因による、USB_DeviceIntStat レジスタの D_SIE_IntStat ビットのアサートを許可／禁止します。

EnNonJ ビットは SLEEP 中も有効です。

3. レジスタ詳細説明

3.2.13 0C3h D_BulkIntEnb (Device Bulk Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0C3h	D_BulkIntEnb	R/W	7: EnCBW_Cmp	0: Disable	1: Enable	00h
			R/W	6: EnCBW_LengthErr	0: Disable	1: Enable	
			R/W	5: EnCBW_Err	0: Disable	1: Enable	
				4:	0:	1:	
			R/W	3: EnCSW_Cmp	0: Disable	1: Enable	
			R/W	2: EnCSW_Err	0: Disable	1: Enable	
				1:	0:	1:	
				0:	0:	1:	

D_BulkIntStat レジスタの割り込み要因による、USB_DeviceIntStat レジスタの D_BulkIntStat ビットのアサートを許可／禁止します。

3.2.14 0C4h D_EPrIntEnb (Device EPr Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0C4h	D_EPrIntEnb	R/W	7: EnD_AlarmIN_IntStat	0:	1:	00h
			R/W	6: EnD_AlarmOUT_IntStat	0:	1:	
				5:	0:	1:	
			R/W	4: EnD_EPeIntStat	0: Disable	1: Enable	
			R/W	3: EnD_EPdIntStat	0: Disable	1: Enable	
			R/W	2: EnD_EPcIntStat	0: Disable	1: Enable	
			R/W	1: EnD_EPbIntStat	0: Disable	1: Enable	
			R/W	0: EnD_EPaIntStat	0: Disable	1: Enable	

D_EPrIntStat レジスタの割り込み要因による、USB_DeviceIntStat レジスタの D_EPrIntStat ビットのアサートを許可／禁止します。

3. レジスタ詳細説明

3.2.15 0C5h D_EP0IntEnb (Device EP0 Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0C5h	D_EP0IntEnb	R/W	7: EnDescriptorCmp	0: Disable	1: Enable	00h
			R/W	6: EnOUT_ShortACK	0: Disable	1: Enable	
			R/W	5: EnIN_TranACK	0: Disable	1: Enable	
			R/W	4: EnOUT_TranACK	0: Disable	1: Enable	
			R/W	3: EnIN_TranNAK	0: Disable	1: Enable	
			R/W	2: EnOUT_TranNAK	0: Disable	1: Enable	
			R/W	1: EnIN_TranErr	0: Disable	1: Enable	
			R/W	0: EnOUT_TranErr	0: Disable	1: Enable	

D_EP0IntStat レジスタの割り込み要因による、USB_DeviceIntStat レジスタの D_EP0IntStat ビットのアサートを許可／禁止します。

3.2.16 0C6h D_EPaIntEnb (Device EPa Interrupt Enable)**3.2.17 0C7h D_EPbIntEnb (Device EPb Interrupt Enable)****3.2.18 0C8h D_EPcIntEnb (Device EPc Interrupt Enable)****3.2.19 0C9h D_EPdIntEnb (Device EPd Interrupt Enable)****3.2.20 0CAh D_EPeIntEnb (Device EPe Interrupt Enable)**

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0C6h	D_EPaIntEnb		7:	0:	1:	00h
	0C7h	D_EPbIntEnb	R/W	6: EnOUT_ShortACK	0: Disable	1: Enable	
	0C8h	D_EPcIntEnb	R/W	5: EnIN_TranACK	0: Disable	1: Enable	
	0C9h	D_EPdIntEnb	R/W	4: EnOUT_TranACK	0: Disable	1: Enable	
	0CAh	D_EPeIntEnb	R/W	3: EnIN_TranNAK	0: Disable	1: Enable	
			R/W	2: EnOUT_TranNAK	0: Disable	1: Enable	
			R/W	1: EnIN_TranErr	0: Disable	1: Enable	
			R/W	0: EnOUT_TranErr	0: Disable	1: Enable	

D_EPx{x=a-e}IntStat レジスタの割り込み要因による、D_EPrIntStat レジスタの EPx{x=a-e}IntStat ビットのアサートを許可／禁止します。

3. レジスタ詳細説明

3.2.21 0CCh-0CDh D_AlarmIN_IntEnb_H,L (Device AlarmIN Interrupt Enable High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0CCh-0CDh	D_AlarmIN _IntEnb,L	R/W	15: EnAlarmEP15IN	0: Disable	1: Enable	0000h
			R/W	14: EnAlarmEP14IN	0: Disable	1: Enable	
			R/W	13: EnAlarmEP13IN	0: Disable	1: Enable	
			R/W	12: EnAlarmEP12IN	0: Disable	1: Enable	
			R/W	11: EnAlarmEP11IN	0: Disable	1: Enable	
			R/W	10: EnAlarmEP10IN	0: Disable	1: Enable	
			R/W	9: EnAlarmEP9IN	0: Disable	1: Enable	
			R/W	8: EnAlarmEP8IN	0: Disable	1: Enable	
			R/W	7: EnAlarmEP7IN	0: Disable	1: Enable	
			R/W	6: EnAlarmEP6IN	0: Disable	1: Enable	
			R/W	5: EnAlarmEP5IN	0: Disable	1: Enable	
			R/W	4: EnAlarmEP4IN	0: Disable	1: Enable	
			R/W	3: EnAlarmEP3IN	0: Disable	1: Enable	
			R/W	2: EnAlarmEP2IN	0: Disable	1: Enable	
			R/W	1: EnAlarmEP1IN	0: Disable	1: Enable	
				0:	0:	1:	

D_AlarmIN_IntStat レジスタの割り込み要因による、D_EPrIntStat レジスタの AlarmIN_IntStat ビットの
アサートを許可／禁止します。

3.2.22 0CEh-0CFh D_AlarmOUT_IntEnb_H,L (Device AlarmOUT Interrupt Enable High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0CEh-0CFh	D_AlarmOUT _IntEnb_H,L	R/W	15: EnAlarmEP15OUT	0: Disable	1: Enable	0000h
			R/W	14: EnAlarmEP14OUT	0: Disable	1: Enable	
			R/W	13: EnAlarmEP13OUT	0: Disable	1: Enable	
			R/W	12: EnAlarmEP12OUT	0: Disable	1: Enable	
			R/W	11: EnAlarmEP11OUT	0: Disable	1: Enable	
			R/W	10: EnAlarmEP10OUT	0: Disable	1: Enable	
			R/W	9: EnAlarmEP9OUT	0: Disable	1: Enable	
			R/W	8: EnAlarmEP8OUT	0: Disable	1: Enable	
			R/W	7: EnAlarmEP7OUT	0: Disable	1: Enable	
			R/W	6: EnAlarmEP6OUT	0: Disable	1: Enable	
			R/W	5: EnAlarmEP5OUT	0: Disable	1: Enable	
			R/W	4: EnAlarmEP4OUT	0: Disable	1: Enable	
			R/W	3: EnAlarmEP3OUT	0: Disable	1: Enable	
			R/W	2: EnAlarmEP2OUT	0: Disable	1: Enable	
			R/W	1: EnAlarmEP1OUT	0: Disable	1: Enable	
				0:	0:	1:	

D_AlarmOUT_IntStat レジスタの割り込み要因による、D_EPrIntStat レジスタの AlarmOUT_IntStat ビットのアサートを許可／禁止します。

3. レジスタ詳細説明

3.2.23 0D0h D_NegoControl (Device Negotiation Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0D0h	D_NegoControl	R/W	7: DisBusDetect	0: Enable BusDetect	1: Disable BusDetect	00h
			R/W	6: EnAutoNego	0: Disable AutoNegotiation	1: Enable AutoNegotiation	
			R/W	5: InSUSPEND	0: Do nothing	1: Monitor NonJ	
			R/W	4: DisableHS	0: HS mode	1: Disable HS mode	
			R/W	3: SendWakeup	0: Do nothing	1: Send Remotewakeup Signal	
			R/W	2: RestoreUSB	0: Do nothing	1: Restore operation mode	
			R/W	1: GoChirp	0: Do nothing	1: Do Chirp sequence	
			R/W	0: ActiveUSB	0: Disactivate USB	1: Activate USB	

デバイスのネゴシエーション に関する動作設定を行います。

Bit7 DisBusDetect

このビットを“1”にセットすると、USB のリセット／サスペンドステートの自動検出を無効にします。このビットが“0”にクリアされている場合、USB のリセット／サスペンドステートの検出のため、USB バス上のバス・アクティビティを監視します。

“HS”モード時は、バス・アクティビティが 3ms の期間検出されない場合、自動的に“FS”モードに切り替えたのち、USB のリセットあるいはサスペンドステートの判定を行い、その後該当する割り込み要因 (DetectReset、DetectSuspend) をセットします。“FS”モード時はバス・アクティビティが 3ms の期間検出されない USB のサスペンドステートと判定し、また、2.5 μ s 以上の“SE0”を検出するとリセットと判断し、該当する割り込み要因をセットします。

DetectReset、DetectSuspend のビットが“1”にセットされたら DisBusDetect ビットを“1”にセットして USB のリセット／サスペンドステートが継続している間、検出を無効にして下さい。AutoNegotiation 機能を使用する場合、このビットに“1”をセットしないようにして下さい。

Bit6 EnAutoNego

AutoNegotiation 機能を有効にします。AutoNegotiation 機能は、リセット検出時に、スピードネゴシエーションが終了してスピードモードが決定するまでのシーケンスを自動化します。AutoNegotiation 機能の詳細は、動作説明の章を参照して下さい。

Bit5 InSUSPEND

AutoNegotiation 機能使用時に、USB のサスペンドステートを検出すると自動的に“1”にセットされ NonJ ステートの検出機能を有効にします。USB のサスペンドステートから復帰する場合には、このビットを“0”にクリアして下さい。

AutoNegotiation 機能を使用する場合の説明は、“機能説明 オート・ネゴシエーション機能”をご参照下さい。

Bit4 DisableHS

GoChirp が“1”にセットされた時に、このビットが“1”にセットされている時には、DeviceChirp を送出せずに強制的に FS モードとなり、ChirpCmp 割り込みを発生します。

Bit3 SendWakeup

このビットを“1”にセットすると、USB ポートに RemoteWakeup 信号(K)を出力します。

RemoteWakeup 信号の送出開始から 1ms 以上 15ms 以内経過後、このビットを“0”にクリアして送出を停止して下さい。

Bit2 RestoreUSB

USB のサスペンドステートからリジュームする際に、このビットを“1”にセットすると、USB のサスペンド前に保存された動作モード (FS or HS) に自動的に切り替えられ、該当する割り込み要因 (RestoreCmp) がセットされます。

このビットは、動作終了後自動的に“0”にクリアされます。

AutoNegotiation 機能を使用する場合、このビットの機能は自動的に制御されますので、このビットをセット／クリアしないで下さい。

Bit1 GoChirp

USB バスがリセット状態である場合に、このビットに“1”をセットすると、ホスト／ハブとの間で“HS Detection Handshake”を行い、XcvrControl レジスタの TermSelect ビット、XcvrSelect ビット及び USB_Status レジスタの FSxHS ビットが自動的に設定されます。動作終了と同時に割り込み要因 (ChirpCmp) がセットされます。

このビットは、動作終了後自動的に“0”にクリアされます。動作終了後 USBStauts レジスタの FSxHS ビットを参照することで、“HS Detection Handshake”の結果が確認できます。

AutoNegotiation 機能を使用する場合、このビットの機能は自動的に制御されますので、このビットをセット／クリアしないで下さい。

Bit0 ActiveUSB

本 LSI では、このビットがハードリセット後“0”にクリアされているため、USB デバイスの全機能を停止しています。本 LSI の設定終了後に、本ビットを“1”にセットすることで、USB デバイスとしての動作が可能となります。

3. レジスタ詳細説明

3.2.24 0D3h D_XcvtControl (Device Xcvt Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0D3h	D_XcvrControl	R/W	7: TermSelect	0: HS Termination	1: FS Termination	41h
			R/W	6: XcvrSelect	0: HS Transceiver	1: FS Transceiver	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: OpMode [1]	OpMode [1:0]		
				0: OpMode [0]			

デバイスのトランシーバマクロに関する設定を行います。

Bit7 TermSelect

FS または HS いずれかのターミネーションを選択して有効にします。USB_Control レジスタの GoChirp ビットによって“HS detection handshake”を実行した場合、または、D_NegoControl レジスタの EnAutoNego ビットがセットされ、AutoNegotiation 機能が実行された場合、このビットは自動的に設定されます。

Bit6 XcvtSelect

FS または HS いずれかのトランシーバを選択して有効にします。D_NegoControl レジスタの GoChirp ビットによって“HS detection handshake”を実行した場合、または、D_NegoControl レジスタの EnAutoNego ビットがセットされ、AutoNegotiation 機能が実行された場合、このビットは自動的に設定されます。

Bit5-2 Reserved

Bit1-0 OpMode

MTM のオペレーションモードを設定します。

USB ケーブルが抜かれている時（※）、USB のサスペンド状態になる時、またはテストモード時以外には、通常設定する必要がありません。

OpMode		
00	“Normal Operation”	通常使用状態
01	“Non-Driving”	USB ケーブルが抜かれている時にはこの状態にして下さい。
10	“Disable Bitstuffing and NRZI encoding”	USB テストモード時にはこの状態にしてください。
11	“Power-Down”	USB のサスペンド時にはこの状態にしてください。

※USB ケーブルが抜けている時には、このレジスタを“41h”にセットすることを推奨します。

3.2.25 0D4h D_USB_Test (Device USB_Test)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0D4h	D_USB_Test	R/W	7: EnHS_Test	0: Do nothing	1: EnHS_Test	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R/W	3: Test_SE0_NAK	0: Do nothing	1: Test_SE0_NAK	
			R/W	2: Test_J	0: Do nothing	1: Test_J	
			R/W	1: Test_K	0: Do nothing	1: Test_K	
			R/W	0: Test_Packet	0: Do nothing	1: Test_Packett	

デバイスの USB 2.0 のテストモードに関する動作設定を行います。SetFeature リクエストで指定されたテストモードに対応するビットを設定し、ステータスステージ終了後に EnHS_Test ビットに“1”をセットすることにより、USB2.0 の規格で定義されたテストモードの動作を行うようにして下さい。

Bit7 EnHS_Test

このビットに“1”をセットすると、D_USB_Test レジスタの下位 4 ビットのいずれかのビットに“1”が設定されている場合、そのビットに対応するテストモードに入ります。テストモードを行う際には、D_NegoControl レジスタの DisBusDetect ビットを“1”にして USB のサスペンドとリセットの検出を行わないようにする必要があります。また、D_NegoControl レジスタの EnAutoNego ビットを“0”にクリアして、AutoNegotiation 機能を無効にして下さい。

また、テストモードへの移行は、SetFeature リクエストにおけるステータスステージの終了後に行うように、ご注意ください。

Bit6-4 Reserved**Bit3 Test_SE0_NAK**

このビットを“1”に設定し、EnHS_Test ビットに“1”をセットすることにより、Test_SE0_NAK テストモードに入ることができます。

Bit2 TEST_J

このビットを“1”に設定し、EnHS_Test ビットに“1”をセットすることにより、Test_J テストモードに入ることができます。なお、このテストモードでは、EnHS_Test ビットを“1”にセットする前に、XcvtControl レジスタの、TermSelect 及び XcvtSelect をスピードに従って設定し、また、OpMode を“10” (Disable Bitstuffing and NRZI encoding) にセットして下さい。

Bit1 TEST_K

このビットを“1”に設定し、EnHS_Test ビットに“1”をセットすることにより、Test_K テストモードに入ることができます。なお、このテストモードでは、EnHS_Test ビットを“1”にする前に、XcvtControl レジスタの、TermSelect 及び XcvtSelect をスピードに従って設定し、また、OpMode を“10” (Disable Bitstuffing and NRZI encoding) にセットして下さい。

3. レジスタ詳細説明

Bit0 Test_Packet

このビットを“1”に設定し、EnHS_Test ビットに “1” をセットすることにより、Test_Packet テストモードに入ることができます。

このテストモードは EP0 以外の任意のエンドポイントで使用できますので、下記の設定を行って下さい。

- 1) エンドポイント EPx{x=a-e}の MaxPacketSize を 64 以上、転送方向を IN に設定し、EndpointNumber を"0xF"に設定して、使用可能として下さい。また、エンドポイント EPx{x=a-e}の FIFO を 64Byte 以上、割り当てて下さい。
- 2) エンドポイントの設定を、上記 EPx{x=a-e}の設定と重複しないようにして下さい。
または、AREAx{x=1-5}Join_1.JoinEPxCHx{x=a-e}ビットをクリアして下さい。
- 3) EPx{x=a-e}の FIFO をクリアし、下記のテストパケット用のデータをこの FIFO に書き込んで下さい。
D_EPx{x=a-e}IntStat レジスタの IN_TranErr ビットを“0”にクリアして下さい。
- 4) Test Packet の送信完了の毎に、IN_TranErr ステータスが“1”にセットされます。
パケット送信テストモード時に FIFO に書き込むデータは以下の 53 バイトです。

00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,
00h, AAh, AAh, AAh, AAh, AAh, AAh, AAh,
AAh, EEh, EEh, EEh, EEh, EEh, EEh, EEh,
EEh, FEh, FFh, FFh, FFh, FFh, FFh, FFh,
FFh, FFh, FFh, FFh, FFh, 7Fh, BFh, DFh,
EFh, F7h, FBh, FDh, FCh, 7Eh, BFh, DFh,
EFh, F7h, FBh, FDh, 7Eh

テストパケット送出時に、SIE が PID と CRC を付加しますので、FIFO に書き込むデータは、USB 規格 Rev.2.0 に記載されているテストパケットデータのうち、DATA0 PID の次のデータから、CRC16 以外のデータまでとなります。

3.2.26 0D6h D_EPnControl (Device Endpoint Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0D6h	D_EPnControl	W	7: AllForceNAK	0: Do nothing	1: Set All ForceNAK	XXh
			W	6: EPrForceSTALL	0: Do nothing	1: Set EP's ForceSTALL	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

エンドポイントの動作設定を行います。ライトオンリーのレジスタです。

Bit7 AllForceNAK

全てのエンドポイントの ForceNAK ビットを“1”にセットします。

Bit6 EPrForceSTALL

エンドポイント EPa,EPb,EPc,EPd,EPe の ForceSTALL ビットを“1”にセットします。

Bit5-0 Reserved

3. レジスタ詳細説明

3.2.27 0D8h D_BulkOnlyControl (Device BulkOnly Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0D8h	D_BulkOnlyControl	R/W	7:AutoForceNAK_CBW	0: None	1: AutoForceNAK after CBW	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R/W	2: GoCBW_Mode	0: None	1: Begin CBW Mode	
			R/W	1: GoCSW_Mode	0: None	1: Begin CSW Mode	
				0:	0:	1:	

バルクオンリーサポート機能を制御します。

Bit7 **AutoForceNAK_CBW**

このビットを“1”にセットすると、CBW サポートによって CBW の受信する OUT トランザクションが完了すると、該当するエンドポイントの ForceNAK ビットを“1”にセットします。

Bit6-3 **Reserved**

Bit2 **GoCBW_Mode**

このビットを“1”にセットすると、該当するエンドポイントで CBW サポートを実行します。CBW サポートを実行するエンドポイントについては、BulkOnlyConfig レジスタの項を参照して下さい。

Bit1 **GoCSW_Mode**

このビットを“1”にセットすると、該当するエンドポイントで CSW サポートを実行します。CSW サポートを実行するエンドポイントについては、BulkOnlyConfig レジスタの項を参照して下さい。

Bit0 **Reserved**

3.2.28 0D9h D_BulkOnlyConfig (Device BulkOnly Configuration)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0D9h	D_BulkOnlyConfig		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R/W	4: EPeBulkOnly	0: None	1: Enable BulkOnly on EPe	
			R/W	3: EPdBulkOnly	0: None	1: Enable BulkOnly on EPd	
			R/W	2: EPcBulkOnly	0: None	1: Enable BulkOnly on EPc	
			R/W	1: EPbBulkOnly	0: None	1: Enable BulkOnly on EPb	
			R/W	0: EPaBulkOnly	0: None	1: Enable BulkOnly on EPa	

バルクオンリーサポート機能を有効にします。

Bit7-5 Reserved**Bit4 EPeBulkOnly**

このビットを“1”にセットすると、エンドポイント **EPe** でバルクオンリーサポート機能が有効になります。バルクオンリーサポートが有効にされると、エンドポイント **EPe** が **OUT** のエンドポイントである場合、**BulkOnlyControl.GoCBW_Mode** ビットをセットすることによって、**CBW** サポートを行います。また、エンドポイント **EPe** が **IN** のエンドポイントである場合、**BulkOnlyControl.GoCSW_Mode** ビットをセットすることによって、**CSW** サポートを行います。

同時に2つ以上の **OUT** のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。同様に、同時に2つ以上の **IN** のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。

Bit3 EPdBulkOnly

このビットを“1”にセットすると、エンドポイント **EPd** でバルクオンリーサポート機能が有効になります。バルクオンリーサポートが有効にされると、エンドポイント **EPd** が **OUT** のエンドポイントである場合、**BulkOnlyControl.GoCBW_Mode** ビットをセットすることによって、**CBW** サポートを行います。また、エンドポイント **EPd** が **IN** のエンドポイントである場合、**BulkOnlyControl.GoCSW_Mode** ビットをセットすることによって、**CSW** サポートを行います。

同時に2つ以上の **OUT** のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。同様に、同時に2つ以上の **IN** のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。

Bit2 EPcBulkOnly

このビットを“1”にセットすると、エンドポイント **EPc** でバルクオンリーサポート機能が有効になります。バルクオンリーサポートが有効にされると、エンドポイント **EPc** が **OUT** のエンドポイントである場合、**BulkOnlyControl.GoCBW_Mode** ビットをセットすることによって、**CBW** サポートを行います。また、エンドポイント **EPc** が **IN** のエンドポイントである場合、**BulkOnlyControl.GoCSW_Mode** ビットをセットすることによって、**CSW** サポートを行います。

同時に2つ以上の **OUT** のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。同様に、同時に2つ以上の **IN** のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。

3. レジスタ詳細説明

Bit1 EPbBulkOnly

このビットを“1”にセットすると、エンドポイント EPb でバルクオンリーサポート機能が有効になります。バルクオンリーサポートが有効にされると、エンドポイント EPb が OUT のエンドポイントである場合、BulkOnlyControl.GoCBW_Mode ビットをセットすることによって、CBW サポートを行います。また、エンドポイント EPb が IN のエンドポイントである場合、BulkOnlyControl.GoCSW_Mode ビットをセットすることによって、CSW サポートを行います。

同時に2つ以上の OUT のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。同様に、同時に2つ以上の IN のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。

Bit0 EPaBulkOnly

このビットを“1”にセットすると、エンドポイント EPa でバルクオンリーサポート機能が有効になります。バルクオンリーサポートが有効にされると、エンドポイント EPa が OUT のエンドポイントである場合、BulkOnlyControl.GoCBW_Mode ビットをセットすることによって、CBW サポートを行います。また、エンドポイント EPa が IN のエンドポイントである場合、BulkOnlyControl.GoCSW_Mode ビットをセットすることによって、CSW サポートを行います。

同時に2つ以上の OUT のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。同様に、同時に2つ以上の IN のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。

3.2.29 0E0h D_EP0SETUP_0 (Device EP0 SETUP 0)**3.2.30 0E1h D_EP0SETUP_1 (Device EP0 SETUP 1)****3.2.31 0E2h D_EP0SETUP_2 (Device EP0 SETUP 2)****3.2.32 0E3h D_EP0SETUP_3 (Device EP0 SETUP 3)****3.2.33 0E4h D_EP0SETUP_4 (Device EP0 SETUP 4)****3.2.34 0E5h D_EP0SETUP_5 (Device EP0 SETUP 5)****3.2.35 0E6h D_EP0SETUP_6 (Device EP0 SETUP 6)****3.2.36 0E7h D_EP0SETUP_7 (Device EP0 SETUP 7)**

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Device	0E0h	D_EP0SETUP_0	R	7: EP0SETUP_n [7]	Endpoint 0 SETUP Data 0 -Endpoint 0 SETUP Data 7	00h
	~0E7h	~D_EP0SETUP_7		6: EP0SETUP_n [6]		
				5: EP0SETUP_n [5]		
				4: EP0SETUP_n [4]		
				3: EP0SETUP_n [3]		
				2: EP0SETUP_n [2]		
				1: EP0SETUP_n [1]		
				0: EP0SETUP_n [0]		

エンドポイント EP0 のセットアップステージで受信した 8 バイトのデータが、EP0SETUP_0 から順に格納されます。

EP0SETUP_0

BmRequestType がセットされます。

EP0SETUP_1

BRequest がセットされます。

EP0SETUP_2

Wvalue の下位 8 ビットがセットされます。

EP0SETUP_3

Wvalue の上位 8 ビットがセットされます。

EP0SETUP_4

WIndex の下位 8 ビットがセットされます。

EP0SETUP_5

WIndex の上位 8 ビットがセットされます。

EP0SETUP_6

WLength の下位 8 ビットがセットされます。

EP0SETUP_7

WLength の上位 8 ビットがセットされます。

3. レジスタ詳細説明

3.2.37 0E8h D_USB_Address (Device USB Address)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0E8h	D_USB_Address		7:	0:	1:	00h
			R (W)	6: USB_Address [6]	USB Address		
				5: USB_Address [5]			
				4: USB_Address [4]			
				3: USB_Address [3]			
				2: USB_Address [2]			
				1: USB_Address [1]			
				0: USB_Address [0]			

AutoSetAddress 機能により、USB アドレスが設定されます。

SetAddress() リクエストを受信すると、AutoSetAddress 機能はそのコントロール転送を自動的行います。AutoSetAddress 機能は、SetAddress() リクエストに関わるコントロール転送のステータスステージが完了し、USB_Address をセットした後に、SetAddressCmp ステータスを発行します。

Bit7 **Reserved**

Bit6-0 **USB_Address**

USB アドレスが設定されます。

AutoSetAddress 機能によって自動的に書き込まれます。

また、書き込みが可能ですが、SetAddress() リクエストを受信すると、再度自動的に書き換えます。

3.2.38 0EAh D_SETUP_Control(Device SETUP Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0EAh	D_SETUP_Control		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R/W	0: ProtectEP0	0: None	1: Protect EP0	

コントロール転送関係の設定をします。

Bit7-1 Reserved**Bit0 ProtectEP0**

コントロール転送のセットアップステージが終了し、

受信したデータが D_EP0SETUP_0～D_EP0SETUP_7 レジスタに格納されると、“1”にセットされます。

同時に D_EP0ControlIN,D_EP0ControlOUT レジスタの ForceSTALL ビットが“0”に、ForceNAK ビットが“1”に、ToggleStat ビットが“1”に、自動的に設定されます。

ProtectEP0 ビットは SETUP トランザクションが行われるとセットされます。従って、SetAddress()リクエストに対してもセットされます。

このビットが“1”にセットされていると、EP0 の ForceNAK ビット、ForceSTALL ビットの設定変更ができません。

3. レジスタ詳細説明

3.2.39 0EEh-0EFh D_FrameNumber_H,L (Device FrameNumber High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0EEh-0EFh	D_FrameNumber _H,L	R	15: FnInvalid	0: Frame number is valid	1: Frame number is not valid	8000h
				14:	0:	1:	
				13:	0:	1:	
				12:	0:	1:	
				11:	0:	1:	
			R	10: FrameNumber [10]	Frame Number		
				9: FrameNumber [9]			
				8: FrameNumber [8]			
				7: FrameNumber [7]			
				6: FrameNumber [6]			
				5: FrameNumber [5]			
				4: FrameNumber [4]			
				3: FrameNumber [3]			
				2: FrameNumber [2]			
				1: FrameNumber [1]			
				0: FrameNumber [0]			

SOF トークンを受信する毎に更新される、USB のフレームナンバーが表示されます。

Bit15 FnInvalid

受信した SOF パケットにエラーが発生した時に、このビットが “1” にセットされます。

Bit14-11 Reserved

Bit10-0 FrameNumber [10:0]

受信した SOF パケットの FrameNumber が表示されます。

3.2.40 0F0h D_EP0MaxSize (Device EP0 Max Packet Size)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0F0h	D_EP0MaxSize		7:	0:	1:	40h
			R/W	6: EP0MaxSize [6]	Endpoint [0] Max Packet Size		
				5: EP0MaxSize [5]			
				4: EP0MaxSize [4]			
				3: EP0MaxSize [3]			
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

エンドポイント EP0 の設定を行います。

Bit7 **Reserved**

Bit6-3 **EP0MaxSize [6:3]**

エンドポイント EP0 の MaxPacketSize を設定します。

このエンドポイントは、以下のサイズから任意のサイズを選択して使用可能です。

FS 時 8, 16, 32, 64 バイト

HS 時 64 バイト

Bit2-0 **Reserved**

3. レジスタ詳細説明

3.2.41 0F1h D_EP0Control (Device EP0 Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0F1h	D_EP0Control	R/W	7: INxOUT	0: OUT	1: IN	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R/W	0: ReplyDescriptor	0: Do nothing	1: Reply Descriptor	

エンドポイント EP0 の設定を行います。

Bit7 INxOUT

エンドポイント EP0 の転送方向を設定します。

セットアップステージで受信したリクエストを判断して、このビットに値を設定して下さい。

データステージがある場合は、このビットにデータステージにおける転送方向をセットして下さい。セットアップステージが完了することにより、D_EP0ControlIN 及び D_EP0ControlOUT レジスタの ForceNAK ビットがセットされるので、データステージ及びステータスステージの実行時にクリアして下さい。

データステージが終了したら、ステータスステージの方向に合わせて、このビットを設定しなおして下さい。データステージの転送方向が IN の場合は、ステータスステージは OUT 方向となりますので、このビットに“0”を設定して下さい。また、データステージの転送方向が OUT、またはデータステージがない場合は、ステータスステージは IN 方向となりますので、エンドポイント EP0 の FIFO をクリアして、このビットに“1”を設定して下さい。

このビットの設定値と異なる方向の IN または OUT トランザクションに対しては、NAK 応答します。但し、そのトランザクション方向に対応する D_EP0ControlIN または D_EP0ControlOUT レジスタの ForceSTALL ビットがセットされていると STALL 応答します。

Bit6-1 Reserved

Bit0 ReplyDescriptor

Descriptor 返信機能を実行します。

このビットが“1”にセットされると、エンドポイント EP0 の IN トランザクションに応答して、FIFO から Descriptor データを、MaxPacketSize 分返信します。Descriptor データは、D_DescAdrs_H,L レジスタの設定値のアドレスを先頭にする、D_DescSize_H,L レジスタの設定サイズのデータを指します。これらの設定値は、Descriptor 返信機能の実行中に更新されますので、ReplyDescriptor ビットをセットする毎に設定して下さい。

1つのトランザクション毎に、D_DescAdrs_H,L レジスタは、送信したデータ数だけインクリメントされ、また、D_DescSize_H,L レジスタは、送信したデータ数だけデクリメントされます。

D_DescSize_H,L の設定数のデータを送信して終了した場合、及び、IN トランザクション以外のトランザクションが行われた場合には、Descriptor 返信機能は終了し、ReplyDescriptor ビットは“0”にクリアされ、D_EP0IntStat レジスタの DescriptorCmp ビットと D_EP0IntStat レジスタの IN_TranACK ビットに“1”がセットされます。

さらに詳細な説明は、動作説明の章を参照して下さい。

3.2.42 0F2h D_EP0ControlIN (Device EP0 Control IN)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0F2h	D_EP0ControlIN		7:	0:	1:	00h
			R/W	6: EnShortPkt	0: Do nothing	1: Enable short Packet	
				5:	0:	1:	
			R	4: ToggleStat	Toggle sequence bit		
			W	3: ToggleSet	0: Do nothing	1: Set Toggle sequence bit	
			W	2: ToggleClr	0: Do nothing	1: Clear Toggle sequence bit	
			R/W	1: ForceNAK	0: Do nothing	1: Force NAK	
			R/W	0: ForceSTALL	0: Do nothing	1: Force STALL	

エンドポイント EP0 の IN トランザクションに関する動作設定及び状態表示を行います。

Bit7 **Reserved**

Bit6 **EnShortPkt**

このビットを“1”にセットすることで、エンドポイント EP0 の IN トランザクションに対して、MaxPacketSize に満たない FIFO 内のデータをショートパケットとして送信することができます。ショートパケットを送信した IN トランザクションが完了すると、自動的にこのビットが“0”にクリアされます。MaxPacketSize のパケットを送信した場合は、このビットはクリアされません。

FIFO 内にデータが無い場合にこのビットを“1”にセットすると、ホストからの IN トークンに対して Zero 長パケットを送信することができます。このビットをセットしてパケットを送信している最中に、該当 FIFO にデータを書き込むと、タイミングによりそのデータも含めて送信されることがあります。パケットの送信が終了し、このビットがクリアされるまで、FIFO へのデータ書き込みは行わないで下さい。

Bit5 **Reserved**

Bit4 **ToggleStat**

エンドポイント EP0 の、IN トランザクションのトグルシーケンスビットの状態を示します。

Bit3 **ToggleSet**

エンドポイント EP0 の、IN トランザクションのトグルシーケンスビットを“1”にセットします。ToggleClr ビットと同時にセットした場合、ToggleClr ビットの機能が優先されます。

Bit2 **ToggleClr**

エンドポイント EP0 の、IN トランザクションのトグルシーケンスビットを“0”にクリアします。ToggleSet ビットと同時にセットした場合、このビットの機能が優先されます。

3. レジスタ詳細説明

Bit1 ForceNAK

このビットを“1”にセットすると、FIFO のデータ数に関わらずエンドポイント EP0 の IN トランザクションに対して NAK 応答します。

セットアップステージが完了することによって USB_DeviceIntStat レジスタの RcvEP0SETUP ビットに“1”がセットされると、このビットは“1”にセットされ、D_SETUP_Control.ProtectEP0 ビットが“1”である間、このビットは“0”にクリアできません。また、ショートパケットを送信した IN トランザクションが完了した時、このビットは“1”にセットされます。

このビットを“1”にセットする際に、既にトランザクションが実行中である場合においては、そのトランザクションが終了するまでビットはセットされず、終了と同時にこのビットは“1”にセットされます。トランザクションが実行中で無い場合においては、即座に“1”にセットされます。

Bit0 ForceSTALL

このビットを“1”にセットすると、エンドポイント EP0 の IN トランザクションに対して STALL 応答します。このビットは、ForceNAK ビットの設定より優先されます。

セットアップステージが完了することによって、USB_DeviceIntStat レジスタの RcvEP0SETUP ビットに“1”がセットされると、このビットは“0”にクリアされ、D_SETUP_Control.ProtectEP0 ビットが“1”である間はこのビットを“1”にセットできません。

現在実行中のトランザクションがある場合、トランザクション開始から一定時間後のこのビットの設定は、次のトランザクションから有効になります。

3.2.43 0F3h D_EP0ControlOUT (Device EP0 Control OUT)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0F3h	D_EP0ControlOUT	R/W	7: AutoForceNAK	0: Do nothing	1: Auto Force NAK	00h
				6:	0:	1:	
				5:	0:	1	
			R	4: ToggleStat	Toggle sequence bit		
			W	3: ToggleSet	0: Do nothing	1: Set Toggle sequence bit	
			W	2: ToggleClr	0: Do nothing	1: Clear Toggle sequence bit	
			R/W	1: ForceNAK	0: Do nothing	1: Force NAK	
			R/W	0: ForceSTALL	0: Do nothing	1: Force STALL	

エンドポイント EP0 の OUT トランザクションに関する動作設定及び状態表示を行います。

Bit7 AutoForceNAK

エンドポイント EP0 の OUT トランザクションが正常に完結すると、このレジスタの ForceNAK ビットを“1”にセットします。

Bit6-5 Reserved**Bit4 ToggleStat**

エンドポイント EP0 の、OUT トランザクションのトグルシーケンスビットの状態を示します。

Bit3 ToggleSet

エンドポイント EP0 の、OUT トランザクションのトグルシーケンスビットを“1”にセットします。ToggleClr ビットと同時にセットした場合、ToggleClr ビットの機能が優先されます。

Bit2 ToggleClr

エンドポイント EP0 の、OUT トランザクションのトグルシーケンスビットを“0”にクリアします。ToggleSet ビットと同時にセットした場合、このビットの機能が優先されます。

Bit1 ForceNAK

このビットを“1”にセットすると、FIFO の空き容量に関わらずエンドポイント EP0 の OUT トランザクションに対して NAK 応答します。

セットアップステージが完了することによって USB_DeviceIntStat レジスタの RcvEP0SETUP ビットに“1”がセットされると、このビットは“1”にセットされ、D_SETUP_Control.ProtectEP0 ビットが“1”である間はこのビットを“0”にクリアすることはできません。

このビットを“1”にセットする際に、既にトランザクションが実行中である場合においては、そのトランザクションが終了するまでビットはセットされず、終了と同時にこのビットは“1”にセットされます。トランザクションが実行中で無い場合においては、即座に“1”にセットされます。

3. レジスタ詳細説明

Bit0 ForceSTALL

このビットを“1”にセットすると、エンドポイント EP0 の OUT トランザクションに対して STALL 応答します。このビットは、ForceNAK ビットの設定より優先されます。

セットアップステージが完了することによって、USB_DeviceIntStat レジスタの RcvEP0SETUP ビットに“1”がセットされると、このビットは“0”にクリアされ、D_SETUP_Control.ProtectEP0 ビットが“1”である間はこのビットを“1”にセットすることはできません。

現在実行中のトランザクションがある場合、トランザクション開始から一定時間後のこのビットの設定は、次のトランザクションから有効になります。

3.2.44 0F8h-0F9h D_EPaMaxSize_H,L (Device EPa Max Packet Size High, Low)**3.2.45 100h-101h D_EPbMaxSize_H,L (Device EPb Max Packet Size High, Low)****3.2.46 108h-109h D_EPcMaxSize_H,L (Device EPc Max Packet Size High, Low)****3.2.47 110h-111h D_EPdMaxSize_H,L (Device EPd Max Packet Size High, Low)****3.2.48 118h-119h D_EPeMaxSize_H,L (Device EPe Max Packet Size High), Low**

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0F8h-0F9h	D_EPaMaxSize_H,L		15:	0:	1:	0000h
	100h-101h	D_EPbMaxSize_H,L		14:	0:	1:	
	108h-109h	D_EPcMaxSize_H,L		13:	0:	1:	
	110h-111h	D_EPdMaxSize_H,L		12:	0:	1:	
	118h-119h	D_EPeMaxSize_H,L		11:	0:	1:	
			R/W	10: MaxSize[10]	Endpoint [x] Max Packet Size		
				9: MaxSize [9]			
				8: MaxSize [8]			
				7: MaxSize [7]			
				6: MaxSize [6]			
				5: MaxSize [5]			
				4: MaxSize [4]			
				3: MaxSize [3]			
				2: MaxSize [2]			
				1: MaxSize [1]			
				0: MaxSize [0]			

MaxPacketSize を設定します。

Bit15-11 Reserved**Bit10-0 MaxSize [10:0]**

エンドポイント EPx{x=a-e}の MaxPacketSize を設定します。

このエンドポイントをバルク転送用として使用する場合には、

FS 時 8, 16, 32, 64 バイト

HS 時 512 バイト

のいずれかに設定して下さい。

このエンドポイントをインタラプト転送用として使用する場合は、

FS 時 64 バイトまで

HS 時 1024 バイトまで

の任意の転送数が設定可能です。

このエンドポイントをアイソクロナス転送用として使用する場合は、

FS 時 1～1023 バイトまで

HS 時 1～1024 バイトまで

の任意の転送数が設定可能です。

3. レジスタ詳細説明

3.2.49 0FAh D_EPaConfig (Device EPa Configuration)

3.2.50 102h D_EPbConfig (Device EPb Configuration)

3.2.51 10Ah D_EPcConfig (Device EPc Configuration)

3.2.52 112h D_EPdConfig (Device EPd Configuration)

3.2.53 11Ah D_EPeConfig (Device EPe Configuration)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0FAh	D_EPaConfig	R/W	7: INxOUT	0: OUT	1: IN	00h
	102h	D_EPbConfig	R/W	6: IntEP_Mode	0: Normal Toggle (IN)	1: Always Toggle (IN)	
	10Ah	D_EPcConfig			0: Bulk OUT (OUT)	1: Interrupt OUT (OUT)	
	112h	D_EPdConfig	R/W	5: ISO	0: Not Isochronous	1: Isochronous	
	11Ah	D_EPeConfig		4:	0:	1:	
			R/W	3: EndpointNumber [3]	Endpoint Number		
				2: EndpointNumber [2]			
				1: EndpointNumber [1]			
				0: EndpointNumber [0]			

エンドポイント EPx{x=a-e}の設定を行います。

EndpointNumber と INxOUT の組み合わせが、他のエンドポイントと重複しないように設定して下さい。

Bit7 INxOUT

エンドポイントの転送方向を設定します。

Bit6 IntEP_Mode

Interrupt 転送に関する設定を行います。

Bulk のエンドポイントでは、このビットに“1”を設定しないで下さい。

このビットの設定は、エンドポイントの方向 (IN/OUT) によって異なります (エンドポイントの方向は Bit7”INxOUT”によって設定されます)。

IN 方向 (INxOUT = 1) の場合、トグルシーケンスビットの動作モードを設定します。トグルシーケンスの動作モードは、アプリケーションに依存します。Interrupt IN のエンドポイントに対し、どちらかの動作モードを選択して下さい。

0: Normal toggle — 通常のトグルシーケンスを行います。

1: Always toggle — トランザクション毎に常にトグルします。

このモードについては、USB2.0 規格書 5.7.5 項をご参照下さい。

OUT 方向 (INxOUT = 0) の場合、このエンドポイントにおいて PING フローコントロールを行うか否かを設定します。Interrupt OUT のエンドポイントでは、このビットを“1”にセットして下さい。

0: Bulk OUT — Bulk OUT のエンドポイントはこの設定にして下さい。

1: Interrupt OUT — Interrupt OUT のエンドポイントはこの設定にして下さい。

Bit5 ISO

アイソクロナス転送を行う際に“1”に設定します。バルク転送、インタラプト転送を行うエンドポイントでは、“0”に設定してください。

Bit4 Reserved

Bit3-0 EndpointNumber

0x1～0xF の任意のエンドポイントナンバーを設定します。

3. レジスタ詳細説明

3.2.54 0FCh D_EPaControl (Device EPa Control)

3.2.55 104h D_EPbControl (Device EPb Control)

3.2.56 10Ch D_EPcControl (Device EPc Control)

3.2.57 114h D_EPdControl (Device EPd Control)

3.2.58 11Ch D_EPeControl (Device EPe Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	0FCh	D_EPaControl	R/W	7: AutoForceNAK	0: Do nothing	1: Auto Force NAK	00h
	104h	D_EPbControl	R/W	6: EnShortPkt	0: Do nothing	1: Enable Short Packet	
	10Ch	D_EPcControl	R/W	5: DisAF_NAK_Short	0: Auto Force NAK Short	1 Disable Auto Force	
	114h	D_EPdControl	R	4: ToggleStat	Toggle sequence bit		
	11Ch	D_EPeControl	W	3: ToggleSet	0: Do nothing	1: Set Toggle sequence bit	
			W	2: ToggleClr	0: Do nothing	1: Clear Toggle sequence bit	
			R/W	1: ForceNAK	0: Do nothing	1: Force NAK	
			R/W	0: ForceSTALL	0: Do nothing	1: Force STALL	

エンドポイント EPx{x=a-e}の動作設定を行います。

Bit7 AutoForceNAK

エンドポイント EPx{x=a-e}のトランザクションが正常に完結すると、このレジスタの ForceNAK ビットを“1”にセットします。

Bit6 EnShortPkt

このビットを“1”にセットすることで、エンドポイント EPx{x=a-e}の IN トランザクションに対して、MaxPacketSize に満たない FIFO 内のデータをショートパケットとして送信することができます。ショートパケットを送信した IN トランザクションが完了すると、自動的にこのビットが“0”にクリアされます。マックスパケットサイズのパケットを送信した場合は、このビットはクリアされません。

FIFO 内にデータが無い場合にこのビットを“1”にセットすると、ホストからの IN トークンに対して Zero 長パケットを送信することができます。このビットをセットしてパケットを送信している最中に、該当 FIFO にデータを書き込むと、タイミングによりそのデータも含めて送信されることがあります。パケットの送信が終了し、このビットがクリアされるまで、FIFO へのデータ書き込みは行わないで下さい。

Bit5 DisAF_NAK_Short

Auto Force NAK Short（以下、AF_NAK_Short※）機能の有効／無効を設定します。

※ 正常な OUT トランザクション完結時に受信したパケットがショートパケットの場合、自動的に ForceNAK ビットを“1”にセットする。

デフォルトの設定は AF_NAK_Short 機能が有効です。

このビットを“1”にセットすると、AF_NAK_Short 機能が無効になります。

AutoForceNAK ビットが“1”にセットされている場合は、AutoForceNAK ビットが優先されます。

Bit4 ToggleStat

エンドポイント EPx{x=a-e}のトグルシーケンスビットの状態を示します。

Bit3 ToggleSet

エンドポイント $EPx\{x=a-e\}$ トグルシーケンスビットを“1”にセットします。ToggleClr ビットと同時にセットした場合、ToggleClr ビットの機能が優先されます。

Bit2 ToggleClr

エンドポイント $EPx\{x=a-e\}$ のトグルシーケンスビットを“0”にクリアします。ToggleSet ビットと同時にセットした場合、このビットの機能が優先されます。

Bit1 ForceNAK

このビットを“1”にセットすると、FIFO のデータ数または空き容量に関わらずエンドポイント $EPx\{x=a-e\}$ のトランザクションに対して NAK 応答します。

このビットを“1”にセットする際に、既にトランザクションが実行中である場合においては、そのトランザクションが終了するまでビットはセットされず、終了と同時にこのビットは“1”にセットされます。トランザクションが実行中で無い場合においては、即座に“1”にセットされます。

Bit0 ForceSTALL

このビットを“1”にセットすると、エンドポイント $EPx\{x=a-e\}$ のトランザクションに対して STALL 応答します。このビットは、ForceNAK ビットの設定より優先されます。

現在実行中のトランザクションがある場合、トランザクション開始から一定時間後のこのビットの設定は、次のトランザクションから有効になります。

3. レジスタ詳細説明

3.2.59 120h-121h D_DescAdrs_H,L (Device Descriptor Address High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	120h-121h	D_DescAdrs_H,L		15:	0:	1:	0000h
				14:	0:	1:	
				13:	0:	1:	
			R/W	12: DescAdrs[12]	Descriptor Address		
				11: DescAdrs [11]			
				10: DescAdrs [10]			
				9: DescAdrs [9]			
				8: DescAdrs [8]			
				7: DescAdrs [7]			
				6: DescAdrs [6]			
				5: DescAdrs [5]			
				4: DescAdrs [4]			
				3: DescAdrs [3]			
				2: DescAdrs [2]			
				1: DescAdrs [1]			
				0: DescAdrs [0]			

Descriptor Address を指定します。

Bit15-13 Reserved

Bit12- -0 DescAdrs [12:0]

Descriptor 返信機能における、Descriptor 返信動作開始時の FIFO の先頭アドレスを指定します。

Descriptor Address は、Descriptor 返信機能に対して FIFO 領域を割り当てるものではありません。Descriptor Address は、FIFO の領域設定に関わらず、0x0000 から 0x11FF(4.5kByte)までの FIFO の全領域を指定することが出来ます。

Descriptor 返信時には、エンドポイント EP0 における IN トランザクション完了毎に、送信データ数の分だけ DescAdrs は更新されます。Descriptor 返信機能については、D_EP0Control レジスタの ReplyDescriptor の項を参照して下さい。

Descriptor 返信機能用の FIFO 領域は、明示的には割り当てませんので、D_DescAdrs_H,L レジスタと D_DescSize_H,L レジスタの指定によって、他のエンドポイントの FIFO との重複を避けて下さい。CSW 領域の予約された終了アドレス(0x0030)から、AREA0～5 で確保する領域の先頭アドレスまでの間が適切です。

3.2.60 122h-123h D_DescSize_H,L (Device Descriptor Size High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	122h-123h	D_DescSize_H,L		15:	0:	1:	0000h
				14:	0:	1:	
				13:	0:	1:	
				12:	0:	1:	
				11:	0:	1:	
				10:	0:	1:	
			R/W	9: DescSize [9]	Descriptor Size		
				8: DescSize [8]			
				7: DescSize [7]			
				6: DescSize [6]			
				5: DescSize [5]			
				4: DescSize [4]			
				3: DescSize [3]			
				2: DescSize [2]			
				1: DescSize [1]			
				0: DescSize [0]			

Descriptor Size を指定します。

Bit15-10 Reserved**Bit9-0 DescSize [9:0]**

Descriptor Size には、Descriptor 返信機能において、返信する総データ数を指定します。Descriptor 返信機能については、D_EP0Control レジスタの ReplyDescriptor ビットの項を参照して下さい。

Descriptor Size には、FIFO のサイズ及び領域設定に関わらず、0x000 から 0x3FF までの値を指定することが出来ます。Descriptor 返信時には、エンドポイント EP0 における IN トランザクション完了毎に、送信データ数の分だけ DescSize は更新されます。

Descriptor 返信機能用の FIFO 領域は、明示的には割り当てませんので、D_DescAdrs_H,L レジスタと D_DescSize_H,L レジスタの指定によって、他のエンドポイントの FIFO との重複を避けて下さい。CSW 領域の予約された終了アドレス(0x0030)から、AREA0～5 で確保する領域の先頭アドレスまでの間が適切です。

3. レジスタ詳細説明

3.2.61 126h D_EP_DMA_Ctrl (Device EP DMA Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	126h	D_EP_DMA_Ctrl	R	7: FIFO_Running	0: FIFO is not running	1: FIFO is running	00h
			R/W	6: AutoEnShort	0: Do nothing	1: Auto Enable Short Packet	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

DMA 転送時の、FIFO の状態の表示及び設定を行います。

Bit7 FIFO_Running

DMA に接続されたエンドポイントの FIFO が動作中であることを示します。DMA を起動すると“1”にセットされ、DMA が終了した後、FIFO が空になると“0”にクリアされます。

Bit6 AutoEnShort

DMA の終了時に、マックスパケットサイズに満たないデータ数が FIFO に残る場合に、そのエンドポイントの EnShortPkt ビットを“1”にセットします。

DMA に接続されたエンドポイントが IN 方向である場合に有効です。

Bit5-0 Reserved

3.2.62 128h-129h D_EnEP_IN_H,L (Device Enable Endpoint-IN High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	128h-129h	D_EnEP_IN_H,L	R/W	15: EnEP15IN	0: Disable	1: Enable	0000h
			R/W	14: EnEP14IN	0: Disable	1: Enable	
			R/W	13: EnEP13IN	0: Disable	1: Enable	
			R/W	12: EnEP12IN	0: Disable	1: Enable	
			R/W	11: EnEP11IN	0: Disable	1: Enable	
			R/W	10: EnEP10IN	0: Disable	1: Enable	
			R/W	9: EnEP9IN	0: Disable	1: Enable	
			R/W	8: EnEP8IN	0: Disable	1: Enable	
			R/W	7: EnEP7IN	0: Disable	1: Enable	
			R/W	6: EnEP6IN	0: Disable	1: Enable	
			R/W	5: EnEP5IN	0: Disable	1: Enable	
			R/W	4: EnEP4IN	0: Disable	1: Enable	
			R/W	3: EnEP3IN	0: Disable	1: Enable	
			R/W	2: EnEP2IN	0: Disable	1: Enable	
			R/W	1: EnEP1IN	0: Disable	1: Enable	
				0:	0:	1:	

デバイス時に IN として有効にするエンドポイントを設定します。

D_EnEP_IN_H,L にて有効にしたエンドポイントのうち、D_EPx{x=0,a-e}関連レジスタに適切な設定が行われ、且つ AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットにてジョイン設定しているエンドポイントを除いたエンドポイントに対し、ホストからトランザクションが発行（IN トークンを受信）された場合、以下に示す応答を行います。

この際、IN トークンに対して行われる応答は、D_EnEP_IN_ISO_H,L の設定によって選択され、該当するビットが“1”にセットされているエンドポイントの場合には、ホストに対しゼロ長パケット応答し、“0”にクリアされているエンドポイントの場合には、ホストに対し NAK 応答します。

3. レジスタ詳細説明

3.2.63 12Ah-12Bh D_EnEP_OUT_H,L (Device Enable Endpoint-OUT High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	12Ah-12Bh	D_EnEP_OUT_H,L	R/W	15: EnEP15OUT	0: Disable	1: Enable	0000h
			R/W	14: EnEP14OUT	0: Disable	1: Enable	
			R/W	13: EnEP13OUT	0: Disable	1: Enable	
			R/W	12: EnEP12OUT	0: Disable	1: Enable	
			R/W	11: EnEP11OUT	0: Disable	1: Enable	
			R/W	10: EnEP10OUT	0: Disable	1: Enable	
			R/W	9: EnEP9OUT	0: Disable	1: Enable	
			R/W	8: EnEP8OUT	0: Disable	1: Enable	
			R/W	7: EnEP7OUT	0: Disable	1: Enable	
			R/W	6: EnEP6OUT	0: Disable	1: Enable	
			R/W	5: EnEP5OUT	0: Disable	1: Enable	
			R/W	4: EnEP4OUT	0: Disable	1: Enable	
			R/W	3: EnEP3OUT	0: Disable	1: Enable	
			R/W	2: EnEP2OUT	0: Disable	1: Enable	
			R/W	1: EnEP1OUT	0: Disable	1: Enable	
				0:	0:	1:	

デバイス時に OUT として有効にするエンドポイントを設定します。

D_EnEP_OUT_H,L にて有効にしたエンドポイントのうち、D_EPx{x=0,a-e}関連レジスタに適切な設定が行われ、且つ AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットにてジョイン設定しているエンドポイントを除いたエンドポイントに対し、ホストからトランザクションが発行（OUT トークンを受信）された場合、以下に示す応答を行い、該当するビットに“1”をセットします。

この際、OUT トークンに対して行われる応答は、D_EnEP_OUT_ISO_H,L の設定によって選択され、該当するビットが“1”にセットされているエンドポイントの場合には、ホストから送出されたデータを受け取らず且つハンドシェイクを応答しません。“0”にクリアされているエンドポイントの場合には、ホストに対し NAK 応答します。デバイスが HS に設定され、ホストから PING トークンが発行された場合には、NAK 応答します。

本レジスタの該当ビットがセットされた場合には、D_EPx{x=0,a-e}関連レジスタを適切に設定し、AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットを用いてエンドポイントを FIFO 領域にジョインし、トランザクションを実行可能にしてください。

3.2.64 12Ch-12Dh D_EnEP_IN_ISO_H,L (Device Enable Endpoint-IN Isocronouse High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	12Ch-12Dh	D_EnEP_IN _ISO_H,L	R/W	15: EnEP15IN_ISO	0: Disable	1: Enable	0000h
			R/W	14: EnEP14IN_ISO	0: Disable	1: Enable	
			R/W	13: EnEP13IN_ISO	0: Disable	1: Enable	
			R/W	12: EnEP12IN_ISO	0: Disable	1: Enable	
			R/W	11: EnEP11IN_ISO	0: Disable	1: Enable	
			R/W	10: EnEP10IN_ISO	0: Disable	1: Enable	
			R/W	9: EnEP9IN_ISO	0: Disable	1: Enable	
			R/W	8: EnEP8IN_ISO	0: Disable	1: Enable	
			R/W	7: EnEP7IN_ISO	0: Disable	1: Enable	
			R/W	6: EnEP6IN_ISO	0: Disable	1: Enable	
			R/W	5: EnEP5IN_ISO	0: Disable	1: Enable	
			R/W	4: EnEP4IN_ISO	0: Disable	1: Enable	
			R/W	3: EnEP3IN_ISO	0: Disable	1: Enable	
			R/W	2: EnEP2IN_ISO	0: Disable	1: Enable	
			R/W	1: EnEP1IN_ISO	0: Disable	1: Enable	
				0:	0:	1:	

デバイス時に IN として有効にするエンドポイントを、ISO モードとして設定します。

D_EnEP_IN_H,L にて有効にしたエンドポイントのうち、D_EPx{x=0,a-e}関連レジスタに適切な設定が行われ、且つ AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットにてジョイン設定しているエンドポイントを除いたエンドポイントに対し、ホストからトランザクションが発行 (IN トークンを受信) された場合、以下に示す応答を行います。

この際、IN トークンに対して行われる応答は、D_EnEP_IN_ISO_H,L の設定によって選択され、該当するビットが“1”にセットされているエンドポイントの場合には、ホストに対しゼロ長パケット応答し、“0”にクリアされているエンドポイントの場合には、ホストに対し NAK 応答します。

3. レジスタ詳細説明

3.2.65 12Eh-12Fh D_EnEP_OUT_ISO_H,L (Device Enable Endpoint-OUT Isocronouse High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Device	12Eh-12Fh	D_EnEP_OUT _ISO_H,L	R/W	15: EnEP15OUT_ISO	0: Disable	1: Enable	0000h
			R/W	14: EnEP14OUT_ISO	0: Disable	1: Enable	
			R/W	13: EnEP13OUT_ISO	0: Disable	1: Enable	
			R/W	12: EnEP12OUT_ISO	0: Disable	1: Enable	
			R/W	11: EnEP11OUT_ISO	0: Disable	1: Enable	
			R/W	10: EnEP10OUT_ISO	0: Disable	1: Enable	
			R/W	9: EnEP9OUT_ISO	0: Disable	1: Enable	
			R/W	8: EnEP8OUT_ISO	0: Disable	1: Enable	
			R/W	7: EnEP7OUT_ISO	0: Disable	1: Enable	
			R/W	6: EnEP6OUT_ISO	0: Disable	1: Enable	
			R/W	5: EnEP5OUT_ISO	0: Disable	1: Enable	
			R/W	4: EnEP4OUT_ISO	0: Disable	1: Enable	
			R/W	3: EnEP3OUT_ISO	0: Disable	1: Enable	
			R/W	2: EnEP2OUT_ISO	0: Disable	1: Enable	
			R/W	1: EnEP1OUT_ISO	0: Disable	1: Enable	
				0:	0:	1:	

デバイス時に OUT として有効にするエンドポイントを設定します。

D_EnEP_OUT_H,L にて有効にしたエンドポイントのうち、D_EPx{x=0,a-e}関連レジスタに適切な設定が行われ、且つ AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットにてジョイン設定しているエンドポイントを除いたエンドポイントに対し、ホストからトランザクションが発行（OUT トークンを受信）された場合、以下に示す応答を行い、該当するビットに“1”をセットします。

この際、OUT トークンに対して行われる応答は、D_EnEP_OUT_ISO_H,L の設定によって選択され、該当するビットが“1”にセットされているエンドポイントの場合には、ホストから送出されたデータを受け取らず且つハンドシェイクを応答しません。“0”にクリアされているエンドポイントの場合には、ホストに対し NAK 応答します。デバイスが HS に設定され、ホストから PING トークンが発行された場合には、NAK 応答します。

本レジスタの該当ビットがセットされた場合には、D_EPx{x=0,a-e}関連レジスタを適切に設定し、AREAn{n=0-5}Join.JoinEPxCHx{x=0,a-e}ビットを用いてエンドポイントを FIFO 領域にジョインし、トランザクションを実行可能にしてください。

3.3 ホストレジスタ詳細説明

3.3.1 140h H_SIE_IntStat_0 (Host SIE Interrupt Status 0)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	140h	H_SIE_IntStat_0		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R (W)	4: DetectCon	0: None	1: Detect Connect	
			R (W)	3: DetectDiscon	0: None	1: Detect Disconnect	
			R (W)	2: DetectRmtWkup	0: None	1: Detect Remote WakeUp	
			R (W)	1: DetectDevChirpOK	0: None	1: Detect Device Chirp OK	
			R (W)	0: DetectDevChirpNG	0: None	1: Detect Device Chirp NG	

ホスト SIE 関連の割り込みを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7-5 **Reserved**

Bit4 **DetectCon**

割り込み要因を直接指示します。

USB ケーブルの接続が検出されると“1” にセットされます。

Bit3 **DetectDiscon**

割り込み要因を直接指示します。

USB ケーブルの切断が検出されると“1” にセットされます。

Bit2 **DetectRmtWkup**

割り込み要因を直接指示します。

Suspend 状態で、デバイスからの Remote WakeUp 信号を検出すると“1” にセットされます。

Bit1 **DetectDevChirpOK**

割り込み要因を直接指示します。

デバイスからのチャープ信号が正常であった場合に“1”にセットされます。

また、ホストステートが RESET の状態において、H/W はこのビットによりデバイスからのチャープ信号の有無を判断します。このビットが“1”にセットされている場合は H_NegoControl_1.PortSpeed を“00b(HS)”に設定し、以降 HS デバイスに対する処理を行います。この為、デバイスが切断された時には必ずこのビットを“0”にクリアしてください。

Bit0 **DetectDevChirpNG**

割り込み要因を直接指示します。

デバイスからのチャープ信号が異常であった場合に“1”にセットされます。

3. レジスタ詳細説明

同期ビット (Bit4～0) は、パワーマネージメントが **ACTIVE** ステートであっても、**HostDeviceSel.HOSTxDEVICE** ビットが“1”、すなわち **HOST** モードでなければ、読み出し、書き込み (割り込み要因クリア) できません。したがって、この状態から移行する場合には、これらの割り込みステータスにより割り込み信号 **XINT** がアサートされないよう、F/W にて以下の処理を行って下さい。

＜ACTIVE で HOST モードから移行する時＞

- 1) 割り込みステータスを処理し、クリアする (**H_SIE_IntStat_0.Bit4～0**)
- 2) 割り込みステータスをディスエーブルにする (**H_SIE_IntEnb_0.Bit4～0**)

＜ACTIVE で HOST モードに移行する時＞

- 3) 割り込みステータスをクリアする (**H_SIE_IntStat_0.Bit4～0**)
- 4) 割り込みステータスをイネーブルにする (**H_SIE_IntEnb_0.Bit4～0**)

3.3.2 141h H_SIE_IntStat_1 (SIE Host Interrupt Status 1)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	141h	H_SIE_IntStat_1		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R (W)	3: DisabledCmp	0: None	1: Disabled Complete	
			R (W)	2: ResumeCmp	0: None	1: Resume Complete	
			R (W)	1: SuspendCmp	0: None	1: Suspend Complete	
			R (W)	0: ResetCmp	0: None	1: Reset Complete	

ホスト SIE 関連の割り込みを表示します。全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7-4 Reserved**Bit3 DisabledCmp**

割り込み要因を直接指示します。

H_NegoControl_0.AutoMode[3:0]に GoDISABLED を設定してステート管理機能を実行させた場合に、DISABLED ステートへの遷移が完了した時に“1” にセットされます。

Bit2 ResumeCmp

割り込み要因を直接指示します。

H_NegoControl_0.AutoMode[3:0]に GoRESUME を設定してステート管理機能を実行させた場合に、レジュームが正常に完了した時に“1” にセットされます。

Bit1 SuspendCmp

割り込み要因を直接指示します。

H_NegoControl_0.AutoMode[3:0]に GoSUSPEND を設定してステート管理機能を実行させた場合に、サスペンドへの遷移が完了した時に“1” にセットされます。

Bit0 ResetCmp

割り込み要因を直接指示します。

H_NegoControl_0.AutoMode[3:0]に GoRESET を設定してステート管理機能を実行させた場合に、USB リセットが正常に完了した時に“1” にセットされます。

同期ビット（Bit3～0）は、パワーマネジメントが ACTIVE ステートであっても、HostDeviceSel.HOSTxDEVICE ビットが“1”、すなわち HOST モードでなければ、読み出し、書き込み(割り込み要因クリア)が出来ません。

従って、この状態から以降する場合には、これらの割り込みステータスにより割り込み信号 XINT がアサートされないよう、F/W にて、以下の処理を行って下さい。

＜ACTIVE で HOST モードから移行するとき＞

- 1) 割り込みステータスを処理し、クリアする（H_SIE_IntStat_1.Bit3～0）
- 2) 割り込みステータスをディスエーブルにする（H_SIE_IntEnb_1.Bit3～0）

＜ACTIVE で HOST モードに移行するとき＞

- 3) 割り込みステータスをクリアする（H_SIE_IntStat_1.Bit3～0）
- 4) 割り込みステータスをイネーブルにする（H_SIE_IntEnb_1.Bit3～0）

3. レジスタ詳細説明

3.3.3 143h H_FrameIntStat (Host Frame Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	143h	H_FrameIntStat		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R (W)	2: PortErr	0: None	1: Port Error	
			R (W)	1: FrameNumOver	0: None	1: Frame Number Over	
			R (W)	0: SOF	0: None	1: SOF	

ホストのフレーム関連の割り込みを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7-3 Reserved

Bit2 PortErr

割り込み要因を直接指示します。

ホスト動作時にポートエラーが検出された場合に“1”にセットされます。

Bit2 FrameNumOver

割り込み要因を直接指示します。

フレームナンバーカウンタがオーバーフローした (FrameNumber_H レジスタの MSb (ビット 2) が“1”から“0”に変化した) 時に“1”にセットされます。FrameNumber_H、L レジスタではカウンタ桁数が不足する場合に、この割り込みをカウントすることによりそれを補うことができます。

Bit0 SOF

割り込み要因を直接指示します。

転送スピードに応じて、下記の場合に“1”にセットされます。

HS: ホストコントローラがマイクロフレーム 0 の SOF トークンを送信する時

FS: ホストコントローラが SOF トークンを送信する時

LS: ホストコントローラが keepalive を送信する時

3.3.4 144h H_CHrIntStat (Host CHr Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	144h	H_CHrIntStat		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R	4: H_CHeIntStat	0: None	1: CHe Interrupt	
			R	3: H_CHdIntStat	0: None	1: CHd Interrupt	
			R	2: H_CHcIntStat	0: None	1: CHc Interrupt	
			R	1: H_CHbIntStat	0: None	1: CHb Interrupt	
			R	0: H_CHaIntStat	0: None	1: CHa Interrupt	

チャンネル CHr の割り込みを表示します。

Bit7-5 Reserved**Bit4 H_CHeIntStat**

割り込み要因を間接指示します。

H_CHeIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する H_CHeIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit3 H_CHdIntStat

割り込み要因を間接指示します。

H_CHdIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する H_CHdIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit2 H_CHcIntStat

割り込み要因を間接指示します。

H_CHcIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する H_CHcIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit1 H_CHbIntStat

割り込み要因を間接指示します。

H_CHbIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する H_CHbIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

Bit0 H_CHaIntStat

割り込み要因を間接指示します。

H_CHaIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する H_CHaIntEnb レジスタのビットがイネーブルにされている時、“1”にセットされます。

3. レジスタ詳細説明

3.3.5 145h H_CH0IntStat (Host CH0 Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	145h	H_CH0IntStat	R (W)	7: TotalSizeCmp	0: None	1: TotalSize Complete	00h
			R (W)	6: TranACK	0: None	1: Tran ACK	
			R (W)	5: TranErr	0: None	1: Tran Error	
			R (W)	4: ChangeCondition	0: None	1: Change Condition	
				3:	0:	1:	
				2:	0:	1:	
			R (W)	1: CTL_SupportCmp	0: None	1: CTL_Support Complete	
			R (W)	0: CTL_SupportStop	0: None	1: CTL_Support Stop	

チャンネル CH0 の割り込みステータスを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 TotalSizeCmp

割り込み要因を直接指示します。

IRP 単位の packets 転送を正常に転送完了した時に、このビットが“1”にセットされます。

コントロール転送サポート機能動作時はセットアップステージ、データステージ、ステータスステージの各ステージが正常終了した場合にこのビットが“1”にセットされます。

Bit6 TranACK

割り込み要因を直接指示します。

H_CH0Config_0.ACK_Cnt ビットに設定した回数の個別トランザクションを正常に転送完了した時に、このビットが“1”にセットされます。

コントロール転送サポート機能動作時はデータステージでのみこのビットは“1”にセットされます。

Bit5 TranErr

割り込み要因を直接指示します。

個別トランザクションがリトライエラー、すなわちタイムアウトエラー、CRC エラー、ビットスタッフィングエラー、PID エラー（予期せぬ PID 含む）、トグルミスマッチエラーのいずれかで完了した時に、このビットが“1”にセットされます。

Bit4 ChangeCondition

割り込み要因を直接指示します。

トランザクションにおいてコンディションコードのストール、データ・オーバーラン、データ・アンダーラン、3 回連続のリトライエラーが発生した時に、このビットが“1”にセットされます。

また、F/W により H_CH0Config_0.TranGo ビットを引き上げた際もこのビットが“1”にセットされます。そのときの ConditionCode は最終トランザクションの結果を示します。

Code	Meaning	Description
000	NoERROR ノーエラー	トランザクションがエラーなく完了した
001	STALL ストール	エンドポイントがストール PID を返した
010	DATAOVERRUN データ・オーバーラン	- 最大パケットサイズを超えるデータパケットを受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します - IRP(TotalSize)を超えるデータ数を受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します * データパケットが最大パケットサイズ以下で、データパケットに含まれたデータトグルが予期した値と一致していない場合、データ・オーバーランではなくトグルミスマッチとして処理します
011	DATAUNDERRUN データ・アンダーラン	- 最大パケットサイズ未満のデータパケットを受信し、且つデータ数が IRP(TotalSize)に満たない * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します
100	RETRYERROR リトライエラー	- デバイスがトークンに規定時間以内に応答しない(IN)、または規定時間以内にハンドシェークを発信しない(OUT) - エンドポイントからのデータパケットに CRC エラーが含まれている - エンドポイントからのデータパケットにビット・スタッフィングエラーが含まれている - エンドポイントからの PID 検査ビットが、データ PID(IN)またはハンドシェーク(OUT)に失敗した - 受信 PID が無効または PID 値が定義されていない - エンドポイントからのデータパケットに含まれたデータトグルが、予期した値と一致していない（トグルミスマッチ）
その他	Reserved	

Bit3-2 Reserved

Bit1 CTL_SupportCmp

割り込み要因を直接指示します。

コントロール転送サポート機能によるコントロール転送の全ステージが正常に完了した時に、このビットが“1”にセットされます。

また、H_CTL_SupportControl レジスタの CTL_SupportGo ビットをクリアすることによるコントロール転送サポート機能の停止処理において、ステータスステージが正常終了して停止処理が終了した場合、このビットが“1”にセットされます。

Bit0 CTL_SupportStop

割り込み要因を直接指示します。

コントロール転送サポート機能によるコントロール転送の途中で異常終了した時に、このビットが“1”にセットされます。

また、H_CTL_SupportControl レジスタの CTL_SupportGo ビットをクリアすることによるコントロール転送サポート機能の停止処理において、ステータスステージ以外のステージで停止処理が終了するか、ステータスステージでトランザクションがエラー終了した場合、このビットが“1”にセットされます。

3. レジスタ詳細説明

3.3.6 146h H_CHaIntStat (Host CHa Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	146h	H_CHaIntStat	R (W)	7: TotalSizeCmp	0: None	1: TotalSize Complete	00h
			R (W)	6: TranACK	0: None	1: Tran ACK	
			R (W)	5: TranErr	0: None	1: Tran Error	
			R (W)	4: ChangeCondition	0: None	1: Change Condition	
				3:	0:	1:	
				2:	0:	1:	
			R (W)	1: BO_SupportCmp	0: None	1: BO Support Complete	
			R (W)	0: BO_SupportStop	0: None	1: BO Support Stop	

チャンネル CHa の割り込みステータスを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 TotalSizeCmp

割り込み要因を直接指示します。

IRP 単位の packets 転送で正常に転送完了した時に、このビットが“1”にセットされます。

バルクオンリーサポート機能動作時は CBW トランスポート、データトランスポート、CSW トランスポートの各トランスポートが正常終了した場合にこのビットが“1”にセットされます。

Bit6 TranACK

割り込み要因を直接指示します。

H_CHaConfig_0.ACK_Cnt ビットに設定した回数の個別トランザクションで正常に転送完了した時に、このビットが“1”にセットされます。

バルクオンリーサポート機能動作時はデータトランスポートでのみこのビットは“1”にセットされます。

Bit5 TranErr

割り込み要因を直接指示します。個別トランザクションがリトライエラー、すなわちタイムアウトエラー、CRC エラー、ビットスタッフィングエラー、PID エラー（予期せぬ PID 含む）、トグルミスマッチエラーのいずれかで完了した時に、このビットが“1”にセットされます。

Bit4 ChangeCondition

割り込み要因を直接指示します。

トランザクションにおいてコンディションコードのストール、データ・オーバーラン、データ・アンダーラン、3 回連続のリトライエラーが発生した時に、このビットが“1”にセットされます。

また、F/W により H_CHaConfig_0.TranGo ビットを引き上げた際もこのビットが“1”にセットされます。そのときの ConditionCode は最終トランザクションの結果を示します。

Code	Meaning	Description
000	NOERROR ノーエラー	トランザクションがエラーなく完了した
001	STALL ストール	エンドポイントがストール PID を返した
010	DATAOVERRUN データ・オーバーラン	- 最大パケットサイズを越えるデータパケットを受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します - IRP(TotalSize)を越えるデータ数を受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します * データパケットが最大パケットサイズ以下で、データパケットに含まれたデータトグルが予期した値と一致していない場合、データ・オーバーランではなくトグルミスマッチとして処理します
011	DATAUNDERRUN データ・アンダーラン	- 最大パケットサイズ未満のデータパケットを受信し、且つデータ数が IRP(TotalSize)に満たない * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します
100	RETRYERROR リトライエラー	- デバイスがトークンに規定時間以内に応答しない(IN)、または規定時間以内にハンドシェークを発信しない(OUT) - エンドポイントからのデータパケットに CRC エラーが含まれている - エンドポイントからのデータパケットにビット・スタッフィングエラーが含まれている - エンドポイントからの PID 検査ビットが、データ PID(IN)またはハンドシェーク(OUT)に失敗した - 受信 PID が無効または PID 値が定義されていない - エンドポイントからのデータパケットに含まれたデータトグルが、予期した値と一致していない（トグルミスマッチ）
その他	Reserved	

Bit3-2 Reserved**Bit1 BO_SupportCmp**

割り込み要因を直接指示します。

バルクオンリーサポート機能による転送で、ステータstransポートが正常に完了した時に、このビットが“1”にセットされます。

また、H_CHaBO_SupportCtl レジスタの BO_SupportGo ビットをクリアすることによるバルクオンリーサポート機能の停止処理において、CSW トランスポートが正常終了して停止処理が終了した場合、このビットが“1”にセットされます。

Bit0 BO_SupportStop

割り込み要因を直接指示します。

バルクオンリーサポート機能による転送で、いずれかの転送が異常終了した時に、このビットが“1”にセットされます。

また、H_CHaBO_SupportCtl レジスタの BO_SupportGo ビットをクリアすることによるバルクオンリーサポート機能の停止処理において、CSW トランスポート以外のトランスポートで停止処理が終了するか、CSW トランスポートでエラーを検出した場合、このビットが“1”にセットされます。

3. レジスタ詳細説明

3.3.7 147h H_CHbIntStat (Host CHb Interrupt Status)

3.3.8 148h H_CHcIntStat (Host CHc Interrupt Status)

3.3.9 149h H_CHdIntStat (Host CHd Interrupt Status)

3.3.10 14Ah H_CHeIntStat (Host CHe Interrupt Status)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	147h	H_CHbIntStat	R (W)	7: TotalSizeCmp	0: None	1: TotalSize Complete	00h
	148h	H_CHcIntStat	R (W)	6: TranACK	0: None	1: Tran ACK	
	149h	H_CHdIntStat	R (W)	5: TranErr	0: None	1: Tran Error	
	14Ah	H_CHeIntStat	R (W)	4: ChangeCondition	0: None	1: Change Condition	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

チャンネル CHx{x=b-e}の割り込みステータスを表示します。

全てのビットは“1”を書き込む事で割り込み要因をクリアできます。

Bit7 TotalSizeCmp

割り込み要因を直接指示します。

IRP 単位の packets 転送で正常に転送完了した時に、このビットが“1”にセットされます。

Bit6 TranACK

割り込み要因を直接指示します。

H_CHx{x=b-e}Config_0.ACK_Cnt ビットに設定した回数の個別トランザクションで正常に転送完了した時に、このビットが“1”にセットされます。

Bit5 TranErr

割り込み要因を直接指示します。個別トランザクションがリトライエラー、すなわちタイムアウトエラー、CRC エラー、ビットスタッフィングエラー、PID エラー（予期せぬ PID 含む）、トグルミスマッチエラーのいずれかで完了した時に、このビットが“1”にセットされます。

Bit4 ChangeCondition

割り込み要因を直接指示します。

トランザクションにおいてコンディションコードのストール、データ・オーバーラン、データ・アンダーラン、3 回連続のリトライエラーが発生した時に、このビットが“1”にセットされます。

また、F/W により H_Chx{x=b-e}Config_0.TranGo ビットを引き上げた際もこのビットが“1”にセットされます。そのときの ConditionCode は最終トランザクションの結果を示します。

Code	Meaning	Description
000	NoERROR ノーエラー	トランザクションがエラーなく完了した
001	STALL ストール	エンドポイントがストール PID を返した
010	DATAOVERRUN データ・オーバーラン	- 最大パケットサイズを越えるデータパケットを受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します - IRP(TotalSize)を越えるデータ数を受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します * データパケットが最大パケットサイズ以下で、データパケットに含まれたデータトグルが予想した値と一致していない場合、データ・オーバーランではなくトグルミスマッチとして処理します
011	DATAUNDERRUN データ・アンダーラン	- 最大パケットサイズ未満のデータパケットを受信し、且つデータ数が IRP(TotalSize)に満たない * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します
100	RETRYERROR リトライエラー	- デバイスがトークンに規定時間以内に応答しない(IN)、または規定時間以内にハンドシェークを発信しない(OUT) - エンドポイントからのデータパケットに CRC エラーが含まれている - エンドポイントからのデータパケットにビット・スタッフィングエラーが含まれている - エンドポイントからの PID 検査ビットが、データ PID(IN)またはハンドシェーク(OUT)に失敗した - 受信 PID が無効または PID 値が定義されていない - インタラプト転送のスプリットトランザクションにおいて ERR ハンドシェークを受信した - インタラプト転送のスプリットトランザクションにおいて NYET ハンドシェークを 3 回連続して受信した - エンドポイントからのデータパケットに含まれたデータトグルが、予想した値と一致していない (トグルミスマッチ)
110	BUFFEROVERRUN バッファ・オーバーラン	アイソクロナス転送において FIFO の空き容量が最大パケットサイズ未満であるためトランザクションが行われなかった。
111	BUFFERUNDERRUN バッファ・アンダーラン	アイソクロナス転送において FIFO の有効データ数が不足のためトランザクションが行われなかった。
その他	Reserved	

Bit3-0 Reserved

3. レジスタ詳細説明

3.3.11 150h H_SIE_IntEnb_0 (Host SIE Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	150h	H_SIE_IntEnb_0		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R/W	4: EnDetectCon	0: Disable	1: Enable	
			R/W	3: EnDetectDiscon	0: Disable	1: Enable	
			R/W	2: EnDetectRmtWkup	0: Disable	1: Enable	
			R/W	1: EnDetectDevChirpOK	0: Disable	1: Enable	
			R/W	0: EnDetectDevChirpNG	0: Disable	1: Enable	

H_SIE_IntStat_0 レジスタの割り込み要因による、MainIntStat レジスタの H_SIE_IntStat_0 ビットのアサート許可／禁止します。

3.3.12 151h H_SIE_IntEnb_1(SIE Host Interrupt Enable 1)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	151h	H_SIE_IntEnb_1		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1	
				4:	0:	1: F	
			R/W	3: EnDisabledCmp	0: Disable	1: Enable	
			R/W	2: EnResumeCmp	0: Disable	1: Enable	
			R/W	1: EnSuspendCmp	0: Disable	1: Enable	
			R/W	0: EnResetCmp	0: Disable	1: Enable	

H_SIE_IntStat_1 レジスタの割り込み要因による、MainIntStat レジスタの H_SIE_IntStat_1 ビットのアサートを許可／禁止します。

3. レジスタ詳細説明

3.3.13 153h H_FrameIntEnb(Host Frame Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	153h	H_FrameIntEnb		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R/W	2: EnPortErr	0: Disable	1: Enable	
			R/W	1: EnFrameNumOver	0: Disable	1: Enable	
			R/W	0: EnSOF	0: Disable	1: Enable	

H_FrameIntStat レジスタの割り込み要因による、MainIntStat レジスタの H_FrameIntStat ビットのアサート許可／禁止します。

3.3.14 154h H_CHrIntEnb(Host CHr Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	154h	H_CHrIntEnb		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R/W	4: EnH_CHeIntStat	0: Disable	1: Enable	
			R/W	3: EnH_CHdIntStat	0: Disable	1: Enable	
			R/W	2: EnH_CHcIntStat	0: Disable	1: Enable	
			R/W	1: EnH_CHbIntStat	0: Disable	1: Enable	
			R/W	0: EnH_CHaIntStat	0: Disable	1: Enable	

H_CHrIntStat レジスタの割り込み要因による、MainIntStat レジスタの H_CHrIntStat ビットのアサートを許可／禁止します。

3. レジスタ詳細説明

3.3.15 155h H_CH0IntEnb(Host CH0 Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	155h	H_CH0IntEnb	R/W	7: EnTotalSizeCmp	0: Disable	1: Enable	00h
			R/W	6: EnTranACK	0: Disable	1: Enable	
			R/W	5: EnTranErr	0: Disable	1: Enable	
			R/W	4: EnChangeCondition	0: Disable	1: Enable	
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: EnCTL_SupportCmp	0: Disable	1: Enable	
			R/W	0: EnCTL_SupportStop	0: Disable	1: Enable	

H_CH0IntStat レジスタの割り込み要因による、MainIntStat レジスタの H_CH0IntStat ビットのアサート
を許可／禁止します。

3.3.16 156h H_CHaIntEnb (Host CHa Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	156h	H_CHaIntEnb	R/W	7: EnTotalSizeCmp	0: Disable	1: Enable	00h
			R/W	6: EnTranACK	0: Disable	1: Enable	
			R/W	5: EnTranErr	0: Disable	1: Enable	
			R/W	4: EnChangeCondition	0: Disable	1: Enable	
				3:	0:	1:	
				2:	0: Disable	1:	
			R/W	1: EnBO_Support_Cmp	0: Disable	1: Enable	
			R/W	0: EnBO_Support_Stop	0: Disable	1: Enable	

H_CHaIntStat レジスタの割り込み要因による、H_CHrIntStat レジスタの CHaIntStat ビットのアサートを許可／禁止します。

3. レジスタ詳細説明

3.3.17 157h H_CHbIntEnb (Host CHb Interrupt Enable)

3.3.18 158h H_CHcIntEnb (Host CHc Interrupt Enable)

3.3.19 159h H_CHdIntEnb (Host CHd Interrupt Enable)

3.3.20 15Ah H_CHeIntEnb (Host CHe Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	157h	H_CHbIntEnb	R/W	7: EnTotalSizeCmp	0: Disable	1: Enable	00h
	158h	H_CHcIntEnb	R/W	6: EnTranACK	0: Disable	1: Enable	
	159h	H_CHdIntEnb	R/W	5: EnTranErr	0: Disable	1: Enable	
	15Ah	H_CHeIntEnb	R/W	4: EnChangeCondition	0: Disable	1: Enable	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

H_CHx{x=b-e}IntStat レジスタの割り込み要因による、H_CHrIntStat レジスタの CHx{x=b-e}IntStat ビットのアサートを許可／禁止します。

3.3.21 160h H_NegoControl_0 (Host NegoControl 0)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	160h	H_NegoControl_0	R/W	7: AutoModeCancel	0: None	1: Cancel	1Xh
			R	6: HostState[2]	HostState[2:0]		
			R	5: HostState[1]			
			R	4: HostState[0]			
			W	3: AutoMode[3]	AutoMode[3:0]		
			W	2: AutoMode[2]			
			W	1: AutoMode[1]			
			W	0: AutoMode[0]			

ホストのネゴシエーション に関する動作設定を行います。

Bit7 AutoModeCancel

このビットを“1”にセットすると、ホストステート管理サポート機能の実行を停止して、そのステートに留まります(H_NegoControl_0. AutoMode および H_XcvrControl の設定を保持、信号ラインの状態を保持、内蔵タイマを停止、接続・切断・デバイス Chirp・リモートウェークアップの各検出機能を OFF します)。

以下の操作を行うとき、事前にこのビットによりホストステート管理サポート機能の実行を停止する必要があります。

- ホストステートを IDLE ステートに遷移させるとき、
- デバイスからの Chirp に異常を検出しリセット完了ステータス (H_SIE_IntStat_1.ResetCmp) の発行を待たずにホストステートを DISABLED ステートに遷移させるとき、
- AutoMode を上書きするとき、
- H_USB_Test.EnHS_Test によりテストモードを実行するとき、

このビットに“1”をセットすると、ホストステート管理サポート機能の停止処理を行い、停止処理が完了することで本ビットは“0”になります (60MHz クロックで 6 サイクル程度必要)。上記の場合においては、本ビットが“0”になったことを確認してから、H_NegoControl_0. AutoMode の GoIDLE 設定または GoDISABLED 設定、あるいは H_USB_Test.EnHS_Test セットを行ってください。

Bit6-4 HostState[2:0]

ホストステート管理サポート機能実行時の現在のホストステートを示します。ステートは以下のいずれかの項目になります。

- 000: Reserved
- 001: IDLE
- 010: WAIT_CONNECT
- 011: DISABLED
- 100: USB_RESET
- 101: USB_OPERATIONAL
- 110: USB_SUSPEND
- 111: USB_RESUME

3. レジスタ詳細説明

Bit3-0 **AutoMode[3:0]**

ホストステート管理サポート機能の実行で、遷移させるホストステートを設定します。
本レジスタはライトオンリーレジスタであり、以下のいずれかを設定します。

- 0001: GoIDLE (IDLE ステートに遷移させる)
- 0010: GoWAIT_CONNECT (WAIT_CONNECT ステートに遷移させる)
- 0011: GoDISABLED (DISABLED ステートに遷移させる)
- 0100: GoRESET (RESET ステートに遷移させる)
- 0101: GoOPERATIONAL (OPERATIONAL ステートに遷移させる)
- 0110: GoSUSPEND (SUSPEND ステートに遷移させる)
- 0111: GoRESUME (RESUME ステートに遷移させる)
- 1001: GoWAIT_CONNECTtoDIS (WAIT_CONNECT ステートから DISABLED ステートまで連続に遷移させる)
- 1010: GoWAIT_CONNECTtoOP (WAIT_CONNECT ステートから OPERATIONAL ステートまで連続に遷移させる)
- 1100: GoRESETtoOP (RESET ステートから OPERATIONAL ステートまで連続に遷移させる)
- 1110: GoSUSPENDtoOP (SUSPEND ステートから OPERATIONAL ステートまで連続に遷移させる)
- 1111: GoRESUMEtoOP (RESUME ステートから OPERATIONAL ステートまで連続に遷移させる)

上記以外: Reserved

あるステートから IDLE ステートに遷移させる (GoIDLE を実行する) 場合には、以下の手順を実行してください。

- H_NegoControl_0 レジスタに 0x80 をライトします。
- H_NegoControl_0.AutoModeCancel ビットが 0 になることを確認します。
- H_NegoControl_0 レジスタに 0x01 をライトします。

3.3.22 162h H_NegoControl_1 (Host NegoControl 1)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	162h	H_NegoControl__1		7:	0:	1:	10h
				6:	0:	1:	
			R/W	5: PortSpeed[1]	PortSpeed[1:0]		
			R/W	4: PortSpeed[0]			
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: DisChirpFinish	0: Normal	1: DisableChirpFinish	
			R/W	0: RmtWkupDetEnb	0: Disable	1: Enable	

ホストのネゴシエーション に関する動作設定を行います。

(注) 本レジスタの **Reset** 値は パワーマネージメントステートが **ACTIVE** 時に読める値です。それ以外のステートでは、**Reset** 値は **00h** が読めます。

Bit7-6 **Reserved**

Bit5-4 **PortSpeed[1:0]**

転送スピードを表示・設定します。

00: High Speed

01: Full Speed

10: Reserved

11: Low Speed

Bit3-2 **Reserved**

Bit1 **DisChirpFinish**

デバイス Chirp が規定時間に完了しないときの動作モードを設定します。

0: デバイス Chirp 異常のステータスを上げた後、規定時間の USB Reset 送出により USB Reset を完了します。

1: デバイス Chirp 異常のステータスを上げた後、デバイス Chirp の完了を待ち、デバイス Chirp が完了するとホスト Chirp を

実行した後、USB Reset を完了します。

Bit0 **RmtWkupDetEnb**

リモートウェイクアップ検出機能の有効／無効を設定します。

3. レジスタ詳細説明

3.3.23 164h H_USB_Test (Host USB_Test)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	164h	H_USB_Test	R/W	7: EnHS_Test	0: Do nothing	1: EnHS_Test	00h
				6:	0:	1:	
				5:	0:	1:	
			R/W	4: Test_Force_Enable	0: Do nothing	1: Test_Force_Enable	
			R/W	3: Test_SE0_NAK	0: Do nothing	1: Test_SE0_NAK	
			R/W	2: Test_J	0: Do nothing	1: Test_J	
			R/W	1: Test_K	0: Do nothing	1: Test_K	
			R/W	0: Test_Packet	0: Do nothing	1: Test_Packett	

ホストの USB 2.0 のテストモードに関する動作設定を行います。

テストモードの実行は WAIT_CONNECT、DISABLED、SUSPEND のいずれかのステートで有効です。

これらのステートからテストモードに入る前には、各ステートの処理を停止する必要があります。テストモードに入る際には以下の手順を実行してください。

- 全チャネルの TranGo ビット(H_CHx{x=0,a-e}Config_0.TranGo)、H_CTL_SupportControl.CTL_SupportGo、H_CHaBO_SupportCtl.BOSupportGo を“0”に設定します。
- H_NegoControl_0 レジスタに 0x80 をライトします。
- H_NegoControl_0.AutoModeCancel ビットが“0”になるのを確認します。
- 本レジスタの下位 5 ビットのいずれかのビットと、EnHS_Test ビットとに、同時に“1”をセットします。

また、あるテストモードから他のテストモードに切り替えるとき、あるいはテストモードを終了させるときは、本レジスタに 0x00 をライトしてください。テストモードを終了し、ホストステートは IDLE となります。

Bit7 EnHS_Test

このビットと、H_USB_Test レジスタの下位 5 ビットのいずれかのビットとに、同時に“1”をセットすると、そのビットに対応するテストモードに入ります。

Bit6-5 Reserved

Bit4 Test_Force_Enable

このビットと、EnHS_Test ビットとに、同時に“1”をセットすることにより、TestForceEnable テストモードに入ることができます。このテストモードでは、ホストポートは HS モードにて SOF を送出し、切断検出が可能となります。

Bit3 Test_SE0_NAK

このビットと、EnHS_Test ビットとに、同時に“1”をセットすることにより、Test_SE0_NAK テストモードに入ることができます。このテストモードでは、ホストポートは HS モードにて受信状態となります。

Bit2 TEST_J

このビットと、EnHS_Test ビットとに、同時に“1”をセットすることにより、Test_J テストモードに入ることができます。このテストモードでは、ホストポートは HS モードにて“J”を送出します。

Bit1 TEST_K

このビットと、EnHS_Test ビットとに、同時に“1”をセットすることにより、Test_K テストモードに入ることができます。このテストモードでは、ホストポートは HS モードにて“K”を送出します。

Bit0 Test_Packet

このビットと、EnHS_Test ビットとに、同時に“1”をセットすることにより、Test_Packet テストモードに入ることができます。このテストモードは CH0 のみで使用できます。テストモードに入る前に CH0 の FIFO 領域を 64 バイトに設定し、CH0 の FIFO をクリアし、下記のテストパケット用のデータをこの FIFO に書き込んで下さい。また、H_CH0Config_1.TID = 00b と設定してください。

パケット送信テストモード時に FIFO に書き込むデータは以下の 53 バイトです。

00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h,
00h, AAh, AAh, AAh, AAh, AAh, AAh, AAh,
AAh, EEh, EEh, EEh, EEh, EEh, EEh, EEh,
EEh, FEh, FFh, FFh, FFh, FFh, FFh, FFh,
FFh, FFh, FFh, FFh, FFh, 7Fh, BFh, DFh,
EFh, F7h, FBh, FDh, FCh, 7Eh, BFh, DFh,
EFh, F7h, FBh, FDh, 7Eh

テストパケット送出時に、SIE が PID と CRC を付加しますので、FIFO に書き込むデータは、USB 規格 Rev.2.0 に記載されているテストパケットデータのうち、DATA0 PID の次のデータから、CRC16 以外のデータまでとなります。

3. レジスタ詳細説明

3.3.24 170h H_CH0SETUP_0 (Host CH0 SETUP 0)

3.3.25 171h H_CH0SETUP_1 (Host CH0 SETUP 1)

3.3.26 172h H_CH0SETUP_2 (Host CH0 SETUP 2)

3.3.27 173h H_CH0SETUP_3 (Host CH0 SETUP 3)

3.3.28 174h H_CH0SETUP_4 (Host CH0 SETUP 4)

3.3.29 175h H_CH0SETUP_5 (Host CH0 SETUP 5)

3.3.30 176h H_CH0SETUP_6 (Host CH0 SETUP 6)

3.3.31 177h H_CH0SETUP_7 (Host CH0 SETUP 7)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	170h -177h	H_CH0SETUP_0 -H_CH0SETUP_7	R/W	7: CH0SETUP_n [7]	Channel 0 SETUP Data 0 - Channel 0 SETUP Data 7	00h
				6: CH0SETUP_n [6]		
				5: CH0SETUP_n [5]		
				4: CH0SETUP_n [4]		
				3: CH0SETUP_n [3]		
				2: CH0SETUP_n [2]		
				1: CH0SETUP_n [1]		
				0: CH0SETUP_n [0]		

チャンネル CH0 のセットアップステージで送信する 8 バイトのデータを順にセットするレジスタです。

CH0SETUP_0

BmRequestType をセットします。

CH0SETUP_1

Brequest をセットします。

CH0SETUP_2

Wvalue の下位 8 ビットをセットします。

CH0SETUP_3

Wvalue の上位 8 ビットをセットします。

CH0SETUP_4

WIndex の下位 8 ビットをセットします。

CH0SETUP_5

WIndex の上位 8 ビットをセットします。

CH0SETUP_6

WLength の下位 8 ビットをセットします。

CH0SETUP_7

WLength の上位 8 ビットをセットします。

3.3.32 17Eh-17Fh H_FrameNumber_H,L (Host FrameNumber High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	17Eh-17Fh	H_FrameNumber_H,L		15:	0:	1:	07FFh
				14:	0:	1:	
				13:	0:	1:	
				12:	0:	1:	
				11:	0:	1:	
			R/W	10: FrameNumber [10]		Frame Number	
				9: FrameNumber [9]			
				8: FrameNumber [8]			
				7: FrameNumber [7]			
				6: FrameNumber [6]			
				5: FrameNumber [5]			
				4: FrameNumber [4]			
				3: FrameNumber [3]			
				2: FrameNumber [2]			
				1: FrameNumber [1]			
				0: FrameNumber [0]			

SOF トークンを送信する毎に更新される、USB のフレームナンバーが表示されます。

(注) 本レジスタの Reset 値は パワーマネージメントステートが ACTIVE 時に読める値です。それ以外のステートでは、Reset 値は 0000h が読めます。

Bit15-11 Reserved

Bit10-0 FrameNumber [10:0]

送信する SOF パケットの FrameNumber が表示されます。

3. レジスタ詳細説明

3.3.33 180h H_CH0Config_0(Host Channel 0 Configuration0)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	180h	H_CH0Config_0	R/W	7: ACK_Cnt[3]	ACK Count		00h
				6: ACK_Cnt[2]			
				5: ACK_Cnt[1]			
				4: ACK_Cnt[0]			
			R/W	3: SpeedMode[1]	Speed Mode		
				2: SpeedMode[0]			
			R/W	1: Toggle	0: Toggle0	1: Toggle1	
			R/W	0: TranGo	0: Stand by	1: Transaction Start	

ホスト動作時にチャンネル CH0 の基本設定を行います。

Bit7-4 ACK_Cnt [3:0]

チャンネル CH0 で行われる転送において、ACK をカウントする数を設定します。

設定された値だけ ACK をカウントすると、H_CH0IntStat レジスタの TranACK ビットがセットされます。

0000: 16 回の ACK をカウントします。

0001~1111: 1 回~15 回の ACK をカウントします。

コントロール転送サポート機能実行中においてはデータステージのトランザクションのみカウントされ、セットアップステージ、ステータスステージのトランザクションはカウントされません。

Bit3-2 SpeedMode [1:0]

チャンネル CH0 で転送を行うデバイスの動作モードを設定します。

00: HS モード — HS デバイスの時はこの設定にしてください。

01: FS モード — FS デバイスの時はこの設定にしてください。

10: Reserved — 本値の使用を禁止します。

11: LS モード — LS デバイスの時はこの設定にしてください。

Bit1 Toggle

トランザクションを開始する時のトグルシーケンスビットの初期値を設定します。また、トランザクション実行後、およびトランザクション完了後はトグルシーケンスビットの状態を示します。

0: トグル 0

1: トグル 1

Bit0**TranGo**

このビットを“1”に設定するとチャンネル **CH0** のトランザクションを開始します。トランザクション開始後このビットを“0”にクリアするとトランザクション処理を停止させることができます。また、このビットはチャンネル **CH0** がトランザクション実行中か否かのステータスの意味も有します。

0: トランザクションを停止します（トランザクション停止中です）

1: トランザクションを開始します（トランザクション実行中です）

転送が **H_CH0TotalSize_H～L** レジスタ で設定したバイト数を完了した時点で **H_CH0IntStat** レジスタの **TotalSizeCmp** ビットが“1”にセットされ、本ビットは自動的に“0”に戻ります。**H_CH0IntStat** レジスタの **ChangeCondition** ビットがセットされたような場合は“0”にリセットされます。この時は、**H_CH0ConditionCode** レジスタにその原因がセットされていますので参照して下さい。

また、このビットのクリアによる停止を行った時は処理最中のトランザクションが終了した時点で、**H_CH0IntStat** レジスタの **ChangeCondition** ビットがセットされます。トランザクションが停止されても、FIFO 中のデータ、（残りの）トータルサイズ、チャンネルに関する設定はそのままの状態となります。従って再びこのビットを“1”にセットすることによって、トランザクションを停止された時の続きから再開させることが可能です。（新たなトランザクションを行う場合は FIFO をクリアし、チャンネル情報の設定をし直してください。）

コントロール転送サポート機能を使用する際はこのビットを設定する必要はありません。

3. レジスタ詳細説明

3.3.34 181h H_CH0Config_1(Host Channel 0 Configuration1)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	181h	H_CH0Config_1	R/W	7: TID[1]	Transaction ID	00h
				6: TD[0]		
				5:	0:	
				4:	0:	
				3:	0:	
				2:	0:	
				1:	0:	
				0:	0:	

ホスト動作時にチャンネル CH0 の基本設定を行います。

Bit7-6 TID[1:0]

チャンネル CH0 で発行するトランザクションの種類（SETUP、OUT、IN）を設定します。
このビットの設定は CTL_SupportControl レジスタの CTL_SupportGo ビットを“1”に設定してトランザクションを開始した時には無効になります。

- 00: SETUP — SETUP トークンを発行します。
- 01: OUT — OUT トークンを発行します。
- 10: IN — IN トークンを発行します。
- 11: Reserved — 本値の使用を禁止します。

コントロール転送サポート機能を使用する際はこのビットを設定する必要はありません。

Bit5-0 Reserved

3.3.35 183h H_CH0MaxPktSize (Host Channel 0 Max Packet Size)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	183h	H_CH0MaxPktSize		7:	0:	1:	00h
			R/W	6: MaxPktSize[6]	Max Packet Size		
				5: MaxPktSize[5]			
				4: MaxPktSize[4]			
				3: MaxPktSize[3]			
				2: MaxPktSize[2]			
				1: MaxPktSize[1]			
				0: MaxPktSize[0]			

ホスト動作時にチャンネル CH0 の MaxPacketSize の設定を行います。

Bit7 **Reserved**

Bit6-0 **MaxPktSize[6:0]**

チャンネル CH0 の MaxPacketSize を設定します。

LS 時 8 バイト

FS 時 8, 16, 32, 64 バイト

HS 時 64 バイト

のいずれかに設定してください。

それ以外の値の設定は禁止します。

3. レジスタ詳細説明

3.3.36 186h-187h H_CH0TotalSize_H,L (Host Channel 0 Total Size High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	186h-187h	H_CH0TotalSize_H,L	R/W	15: TotalSize[15]	Total Size	0000h
				14: TotalSize[14]		
				13: TotalSize[13]		
				12: TotalSize[12]		
				11: TotalSize[11]		
				10: TotalSize[10]		
				9: TotalSize[9]		
				8: TotalSize[8]		
				7: TotalSize[7]		
				6: TotalSize[6]		
				5: TotalSize[5]		
				4: TotalSize[4]		
				3: TotalSize[3]		
				2: TotalSize[2]		
				1: TotalSize[1]		
				0: TotalSize[0]		

ホスト動作時にチャンネル CH0 で転送を行うデータの Total Size を設定します。

Bit15-0 TotalSize[15:0]

チャンネル CH0 における転送データの全バイト数(最大 65,535byte:約 64Kbyte)を設定します。

H_CH0Config_0 レジスタの TranGo ビットによりトランザクションが開始された後は、トランザクションが正常終了するたびにこのレジスタは更新されます。

また、TotalSize = 0 の時、H_CH0Config_0 レジスタの TranGo ビットにより OUT トランザクションを開始するとゼロ長パケットが送信されます。

SETUP トランザクションを行う場合、およびコントロール転送サポート機能を使用する場合は、このレジスタを設定する必要はありません。

3.3.37 188h H_CH0HubAdrs (Host Channel 0 Hub Address)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	188h	H_CH0HubAdrs	R/W	7: HubAdrs[3]	Hub Address		00h
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:	1:	
			R/W	2: Port[2]	Port Number		
				1: Port[1]			
				0: Port[0]			

ホスト動作時にチャンネル CH0 に接続するハブの設定を行います。

Bit7 HubAdrs[3:0]

チャンネル CH0 で転送を行うファンクションが接続されているハブの USB アドレスを設定します。

0～15 までの任意の値が設定できます。

Bit3 Reserved**Bit2-0 Port[2:0]**

チャンネル CH0 で転送を行うファンクションが接続されているハブのポートナンバーを設定します。

0～7 までの任意の値が設定できます。

3. レジスタ詳細説明

3.3.38 189h H_CH0FuncAdrs (Host Channel 0 Function Address)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	189h	H_CH0FuncAdrs	R/W	7: FuncAdrs[3]	Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R/W	3: EP_Number[3]	Endpoint Number	
				2: EP_Number[2]		
				1: EP_Number[1]		
				0: EP_Number[0]		

ホスト動作時にチャンネル CH0 で転送を行うファンクションのアドレス設定を行います。

Bit7-4 FuncAdrs[3:0]

チャンネル CH0 が管理するエンドポイントを含むファンクションの USB アドレスを設定します。

0～15 までの任意の値が設定できます。

Bit3-0 EP_Number[3:0]

チャンネル CH0 で転送を行うエンドポイント番号を設定します。

0～15 までの任意の値が設定できます。

3.3.39 18Bh H_CTL_SupportControl (Host ControlTransfer Support Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	18Bh	H_CTL_Support-Control		7:	0:	1:	00h
				6:	0:	1:	
			R/W	5: CTL_SupportState[1]	ControlTransfer Support State		
				4: CTL_SupportState[0]			
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R/W	0: CTL_SupportGo	0: Stand by	1: Control Transfer Go	

ホスト動作時にチャンネル CH0 でコントロール転送を行う時の支援機能の設定を行います。

Bit7-6 **Reserved**

Bit5-4 **CTL_SupportState[1:0]**

CTL_SupportGo ビットを“1”に設定し、コントロール支援機能を使用して転送を行っている時にどのステージを実行しているかを示します。

- 00: Idle — 転送未実行、または転送が正常に終了したことを示します。
- 01: Setup Stage — セットアップステージを実行していることを示します。
- 10: Data Stage — データステージを実行していることを示します。
- 11: Status Stage — ステータスステージを実行していることを示します。

Bit3-1 **Reserved**

Bit0 **CTL_SupportGo**

このビットを“1”に設定するとチャンネル CH0 でコントロール転送支援機能により、セットアップステージ～（データステージ）～ステータスステージが自動的に行われます。

セットアップステージでは、SETUP トークンを自動的に送出し H_CH0SETUP_0～7 に設定したリクエストが送信されます。

次にデータステージがある場合は、指定された方向およびサイズでトランザクションが自動的に実行されます。

最後にステータスステージでは、データステージの有無および方向により、自動的に適当な PID のトークンを発行しゼロ長パケットの送受信が行われます。

以上のトランザクションおよびステージシーケンスが正常完了すると、H_CH0IntStat レジスタの CTL_SupportCmp ビットがセットされます。シーケンスの途中でパケットにエラーを検出した場合は、H_CH0IntStat レジスタの CTL_SupportStop ビットがセットされ、トランザクションは停止します。この時は、ConditionCode レジスタにその原因がセットされていますので参照して下さい。

コントロール転送が終了した場合(正常終了、エラー終了いずれの場合も)、このビットは自動的にクリアされます。

コントロール転送サポート機能を実行中にこのビットをクリアすることにより、コントロール転送を停止することが出来ます。コントロール転送がステータスステージで正常終了した場合は CTL_SupportCmp ビットがセットされます。これ以外の場合は CTL_SupportStop ビットがセットされます。コントロール転送が停止したステージは CTL_SupportState を参照して下さい。

3. レジスタ詳細説明

3.3.40 18Eh H_CH0ConditionCode (Host Channel 0 Condition Code)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	18Eh	H_CH0ConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Condition Code		
				5: ConditionCode[1]			
				4: ConditionCode[0]			
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

ホスト動作時にチャネル CH0 の転送完了結果を示します。

Bit7 **Reserved**

Bit6-4 **ConditionCode[2:0]**

チャネル CH0 で転送が完了した場合の結果を示します。

Code	Meaning	Description
000	NOERROR ノーエラー	トランザクションがエラーなく完了した
001	STALL ストール	エンドポイントがストール PID を返した
010	DATAOVERRUN データ・オーバーラン	- 最大パケットサイズを越えるデータパケットを受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します - IRP(TotalSize)を越えるデータ数を受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します * データパケットが最大パケットサイズ以下で、データパケットに含まれたデータトグルが予期した値と一致していない場合、データ・オーバーランではなくトグルミスマッチとして処理します
011	DATAUNDERRUN データ・アンダーラン	- 最大パケットサイズ未満のデータパケットを受信し、且つデータ数が IRP(TotalSize)に満たない * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します
100	RETRYERROR リトライエラー	- デバイスがトークンに規定時間以内に応答しない(IN)、または規定時間以内にハンドシェークを発信しない(OUT) - エンドポイントからのデータパケットに CRC エラーが含まれている - エンドポイントからのデータパケットにビット・スタッフィングエラーが含まれている - エンドポイントからの PID 検査ビットが、データ PID(IN)またはハンドシェーク(OUT)に失敗した - 受信 PID が無効または PID 値が定義されていない - エンドポイントからのデータパケットに含まれたデータトグルが、予期した値と一致していない（トグルミスマッチ）
その他	Reserved	

Bit3-0 **Reserved**

3.3.41 190h H_CHaConfig_0(Host Channel a Configuration0)

Mode	Address	Register Name	R/W	Bit Symbol	Description			Reset
Host	190h	H_CHaConfig_0	R/W	7: ACK_Cnt[3]	ACK Count			00h
				6: ACK_Cnt[2]				
				5: ACK_Cnt[1]				
				4: ACK_Cnt[0]				
			R/W	3: SpeedMode[1]	Speed Mode			
				2: SpeedMode[0]				
			R/W	1: Toggle	0: Toggle0	1: Toggle1		
			R/W	0: TranGo	0: Stand by	1: Transaction Start		

ホスト動作時にチャンネル CHa の基本設定を行います。

Bit7-4 ACK_Cnt [3:0]

チャンネル CHa で行われる転送において、ACK をカウントする数を設定します。

設定された値だけ ACK をカウントすると、H_CHaIntStat レジスタの TranACK ビットがセットされます。

0000: 16 回の ACK をカウントします。

0001~1111: 1 回~15 回の ACK をカウントします。

バルクオンリーサポート機能実行中においてはデータトランスポートのトランザクションのみカウントされ、CBW トランスポート、CSW トランスポートのトランザクションはカウントされません。

Bit3-2 SpeedMode [1:0]

チャンネル CHa で転送を行うデバイスの動作モードを設定します。

00: HS モード — HS デバイスの時はこの設定にしてください。

01: FS モード — FS デバイスの時はこの設定にしてください。

10-11: Reserved — 本値の使用を禁止します。

Bit1 Toggle

トランザクションを開始する時のトグルシーケンスビットの初期値を設定します。また、トランザクション実行後、およびトランザクション完了後はトグルシーケンスビットの状態を示します。

0: トグル 0

1: トグル 1

バルクオンリーサポート機能を使用する際はこのビットを設定する必要はありません。

3. レジスタ詳細説明

Bit0 TranGo

このビットを“1”に設定するとチャネル CHa のトランザクションを開始します。トランザクション開始後このビットを“0”にクリアするとトランザクション処理を停止させることができます。また、このビットはチャネル CHa がトランザクション実行中か否かのステータスの意味も有します。

0: トランザクションを停止します（トランザクション停止中です）

1: トランザクションを開始します（トランザクション実行中です）

転送が H_CHaTotalSize_HH～LL レジスタ で設定したバイト数完了した時点で H_CHaIntStat レジスタの TranCmp ビットが“1”にセットされ、本ビットは自動的に“0”に戻ります。H_CHaIntStat レジスタの ChangeCondition ビットがセットされたような場合は“0”にリセットされます。この時は、H_CHaConditionCode レジスタにその原因がセットされていますので参照して下さい。

また、このビットのクリアによる停止を行った時は処理最中のトランザクションが終了した時点で、H_CHaIntStat レジスタの ChangeCondition ビットがセットされます。トランザクションが停止されても、FIFO 中のデータ、（残りの）トータルサイズ、チャネルに関する設定はそのままの状態となります。従って再びこのビットを“1”にセットすることによって、トランザクションを停止された時の続きから再開させることが可能です。（新たなトランザクションを行う場合は FIFO をクリアし、チャネル情報の設定をし直してください。）

バルクオンリーサポート機能を使用する際はこのビットを設定する必要はありません。

3.3.42 191h H_CHaConfig_1(Host Channel a Configuration1)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	191h	H_CHaConfig_1	R/W	7: TID[1]	Transaction ID	00h
				6: TID[0]		
				5:	0:	1:
				4:	0:	1:
			R/W	3: AutoZerolen	0: Do nothing	1: Add Zerolen
				2:	0:	1:
				1:	0:	1:
			R/W	0: TotalSizeFree	0: Do nothing	1: Total Size Free

ホスト動作時にチャネル CHa の基本設定を行います。

Bit7-6 TID[1:0]

チャネル CHa で発行するトランザクションの種類（OUT、IN）を設定します。このビットの設定は H_CHaBO_SupportCtl レジスタの BO_SupportGo ビットを“1”に設定してトランザクションを開始した時には無効になります。

00: Reserved — 本値の使用を禁止します。

01: OUT — OUT トークンを発行します。

10: IN — IN トークンを発行します。

11: Reserved — 本値の使用を禁止します。

バルクオンリーサポート機能を使用する際はこのビットを設定する必要はありません。

Bit5-4 Reserved**Bit3 AutoZerolen**

このビットに“1”を設定すると H_CHaTotalSizeHH～LL レジスタで設定したサイズの転送がちょうど Max Packet Size で完了した際に、ゼロ長パケットを最後に自動付与します。OUT 転送の場合のみこのビットは有効となります。

Bit2-1 Reserved**Bit0 TotalSizeFree**

このビットに“1”を設定すると H_CHaTotalSizeHH～LL レジスタの設定値に関わりなく転送サイズが無限になります。

3. レジスタ詳細説明

3.3.43 192h-193h H_CHaMaxPktSize_H,L (Host Channel a Max Packet Size High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	192h-193h	H_CHaMaxPktSize_H,L		15:	0:	1:	0000h
				14:	0:	1:	
				13:	0:	1:	
				12:	0:	1:	
				11:	0:	1:	
				10:	0:	1:	
			R/W	9: MaxPktSize[9]	Max Packet Size		
				8: MaxPktSize[8]			
				7: MaxPktSize[7]			
				6: MaxPktSize[6]			
				5: MaxPktSize[5]			
				4: MaxPktSize[4]			
				3: MaxPktSize[3]			
				2: MaxPktSize[2]			
				1: MaxPktSize[1]			
				0: MaxPktSize[0]			

ホスト動作時にチャンネル CHa の MaxPacketSize の設定を行います。

Bit15-10 Reserved

リザーブ・ビットには“1”を書き込まないで下さい。

Bit9-0 MaxPktSize[9:0]

チャンネル CHa の MaxPacketSize を設定します。

FS 時 8, 16, 32, 64 バイト (バルクオンリーサポート機能を使用する場合は 32, 64 バイトのいずれか)

HS 時 512 バイト

のいずれかに設定してください。

それ以外の値の設定は禁止します。

3.3.44 194h-195h H_CHaTotalSize_HH,HL (Host Channel a Total Size High-High, High-Low)**3.3.45 196h-197h H_CHaTotalSize_LH,LL (Host Channel a Total Size Low-High, Low-Low)**

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	194h-195h	H_CHaTotalSize_HH,HL	R/W	15: TotalSize[31]	Total Size High	0000h
				14: TotalSize[30]		
				13: TotalSize[29]		
				12: TotalSize[28]		
				11: TotalSize[27]		
				10: TotalSize[26]		
				9: TotalSize[25]		
				8: TotalSize[24]		
				7: TotalSize[23]		
				6: TotalSize[22]		
				5: TotalSize[21]		
				4: TotalSize[20]		
				3: TotalSize[19]		
				2: TotalSize[18]		
				1: TotalSize[17]		
				0: TotalSize[16]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	196h-197h	H_CHaTotalSize_LH,LL	R/W	15: TotalSize[15]	Total Size Low	0000h
				14: TotalSize[14]		
				13: TotalSize[13]		
				12: TotalSize[12]		
				11: TotalSize[11]		
				10: TotalSize[10]		
				9: TotalSize[9]		
				8: TotalSize[8]		
				7: TotalSize[7]		
				6: TotalSize[6]		
				5: TotalSize[5]		
				4: TotalSize[4]		
				3: TotalSize[3]		
				2: TotalSize[2]		
				1: TotalSize[1]		
				0: TotalSize[0]		

ホスト動作時にチャンネル CHa で転送を行うデータの Total Size を設定します。

チャンネル CHa における転送データの全バイト数(最大 4,294,967,295byte : 約 4Gbyte)を設定します。

H_CHaConfig_0 レジスタの TranGo ビットによりトランザクションが開始された後は、トランザクションが正常に終了するたびにこのレジスタは更新されます。

3. レジスタ詳細説明

また、TotalSize = 0 の時、H_CHaConfig_0 レジスタの TranGo ビットにより OUT トランザクションを開始するとゼロ長パケットが送信されます。

バルクオンリーサポート機能を使用する際はこのレジスタを設定する必要はありません。

3.3.46 198h H_CHaHubAdrs (Host Channel a Hub Address)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	198h	H_CHaHubAdrs	R/W	7: HubAdrs[3]	Hub Address		00h
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:	1:	
			R/W	2: Port[2]	Port Number		
				1: Port[1]			
				0: Port[0]			

ホスト動作時にチャンネル CHa に接続するハブの設定を行います。

Bit7 HubAdrs[3:0]

チャンネル CHa で転送を行うファンクションが接続されているハブの USB アドレスを設定します。

0～15 までの任意の値が設定できます。

Bit3 Reserved**Bit2-0 Port[2:0]**

チャンネル CHa で転送を行うファンクションが接続されているハブのポートナンバーを設定します。

0～7 までの任意の値が設定できます。

3. レジスタ詳細説明

3.3.47 199h H_CHaFuncAdrs (Host Channel a Function Address)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	199h	H_CHaFuncAdrs	R/W	7: FuncAdrs[3]	Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R/W	3: EP_Number[3]	Endpoint Number	
				2: EP_Number[2]		
				1: EP_Number[1]		
				0: EP_Number[0]		

ホスト動作時にチャンネル CHa で転送を行うファンクションのアドレス設定を行います。

Bit7-4 FuncAdrs[3:0]

チャンネル CHa が管理するエンドポイントを含むファンクションの USB アドレスを設定します。

0～15 までの任意の値が設定できます。

Bit3-0 EP_Number[3:0]

チャンネル CHa で転送を行うエンドポイント番号を設定します。

0～15 までの任意の値が設定できます。

バルクオンリーサポート機能を使用する際はこのビットを設定する必要はありません。

3.3.48 19Ah H_CHaBO_SupportCtl (Host CHa Bulk Only Transfer Support Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	19Ah	H_CHaBO_Support- Ctl		7:	0:	1:	00h
				6:	0:	1:	
			R/W	5: BO_TransportState[1]	Bulk Only Transfer Transport State		
				4: BO_TransportState[0]			
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R/W	0: BO_SupportGo	0: Stand by	1: BO Transfer Go	

ホスト動作時にチャネル CHa のバルクオンリーサポート機能の設定を行います。

Bit7-6 Reserved

Bit5-4 BO_TransportState[1:0]

BO_SupportGo ビットを“1”に設定し、バルクオンリーサポート機能を使用して転送を行っている時にどのトランスポートを実行しているかを示します。

- 00: Idle — 転送未実行、または転送が正常に終了したことを示します。
- 01: CBW Transport — CBW トランスポートを実行していることを示します。
- 10: Data Transport — データトランスポートを実行していることを示します。
- 11: CSW Transport — CSW トランスポートを実行していることを示します。

Bit3-1 Reserved

Bit0 BO_SupportGo

このビットを“1”に設定するとチャネル CHa でバルクオンリーサポート機能により、CBW トランスポート～(データトランスポート)～CSW トランスポートが自動的に行われます。

CBW トランスポートでは、OUT トークンを自動的に送出し FIFO の CBW エリアに設定したデータが送信されます。

次にデータステージがある場合は、指定された方向およびサイズでデータトランスポートが自動的に実行されます。

最後に CSW トランスポートでは、IN トークンを自動的に送出し FIFO の CSW エリアにデータを受信します。

以上のトランスポートが正常完了すると、H_CHaIntStat レジスタの BO_SupportCmp ビットがセットされます。トランスポートの途中でパケットにエラーを検出した場合、および CSW の値が適切でない場合は H_CHaIntStat レジスタの BO_SupportStop ビットがセットされ、トランザクションは停止します。この時は、H_CHaConditionCode レジスタにその原因がセットされていますので参照して下さい。H_CHaIntStat レジスタの BO_SupportStop ビットが“1”に設定された時に ConditionCode の値が“000”である場合は、CSW の値が適切でないことを表しています。

一連のトランスポートが終了した場合(正常終了、エラー終了いずれの場合も)、このビットは自動的にクリアされます。

バルクオンリーサポート機能を実行中にこのビットをクリアすることにより、トランスポートの処理を停止することが出来ます。この場合、CSW トランスポートが正常終了した場合は BO_SupportCmp ビットが、それ以外は BO_SupportStop ビットがセットされます。停止したトランスポートは BO_TransportState にて確認してください。

3. レジスタ詳細説明

3.3.49 19Bh H_CHaBO_CSW_RcvSize(Host CHa Bulk Only Transfer Support CSW Receive Data Size)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	19Bh	H_CHaBO _CSW_RcvDataSize		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R	3: CSW_RcvDataSize[3]	CSW Resceive Data Size		
				2: CSW_RcvDataSize[2]			
				1: CSW_RcvDataSize[1]			
				0: CSW_RcvDataSize[0]			

ホスト動作時にチャネル CHa のバルクオンリーサポート機能において、CSW トランスポート実行時の受信データ数を示します。

Bit7-4 **Reserved**

Bit3-0 **CSW_RcvDataSize[3:0]**

CSW の受信データ数を示します。

CSW トランスポートで 13 バイト未満のデータを受信した場合、このレジスタにより受信データ数を確認出来ます。

CSW トランスポートでハンドシェークを受信した場合、および CSW トランスポート以外の場合においては、このレジスタの値は意味を持ちません。

3.3.50 19Ch H_CHaBO_OUT_EP_Ctl(Host CHa Bulk Only Transfer Support OUT Endpoint Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	19Ch	H_CHaBO _OUT_EP_Ctl		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R/W	4: OUT_Toggle	0: Toggle0	1: Toggle1	
			R/W	3: OUT_EP_Number[3]	OUT EP Number		
				2: OUT_EP_Number[2]			
				1: OUT_EP_Number[1]			
				0: OUT_EP_Number[0]			

ホスト動作時にチャネル CHa のバルクオンリーサポート機能の設定を行います。

Bit7-5 Reserved

Bit4 OUT_Toggle

H_CBW_Control レジスタの BO_SupportGo ビットを“1”に設定し、バルクオンリーサポート機能を使用して OUT 方向の転送（CBW トランスポート、Data OUT トランスポート）を行う時のトグルシーケンスビットの初期値を設定します。

0: トグル 0

1: トグル 1

また、OUT 方向のトランスポートが正常終了した場合、トグルシーケンスビットをこのビットに自動的に保持します。

Bit3-0 OUT_EP_Number[3:0]

H_CBW_Control レジスタの BO_SupportGo ビットを“1”に設定し、バルクオンリーサポート機能を使用して OUT 方向の転送（CBW トランスポート、Data OUT トランスポート）を行う時の転送先デバイスのエンドポイント番号を設定します。

0～15 までの任意の値が設定できます。

3. レジスタ詳細説明

3.3.51 19Dh H_CHaBO_IN_EP_Ctl(Host CHa Bulk Only Transfer Support IN Endpoint Control)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	19Dh	H_CHaBO_IN_EP_Ctl		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R/W	4: IN_Toggle	0: Toggle0	1: Toggle1	
			R/W	3: IN_EP_Number[3]	IN EP Number		
				2: IN_EP_Number[2]			
				1: IN_EP_Number[1]			
				0: IN_EP_Number[0]			

ホスト動作時にチャネル CHa のバルクオンリーサポート機能の設定を行います。

Bit7-5 **Reserved**

Bit4 **IN_Toggle**

H_CBW_Control レジスタの BO_SupportGo ビットを“1”に設定し、バルクオンリーサポート機能を使用して IN 方向の転送（CSW トランスポート、Data IN トランスポート）を行う時のトグルシーケンスビットの初期値を設定します。

0: トグル 0

1: トグル 1

また、IN 方向のトランスポートが正常終了した場合、トグルシーケンスビットをこのビットに自動的に保持します。

Bit3-0 **IN_EP_Number[3:0]**

H_CBW_Control レジスタの BO_SupportGo ビットを“1”に設定し、バルクオンリーサポート機能を使用して IN 方向の転送（CSW トランスポート、Data IN トランスポート）を行う時の転送先デバイスのエンドポイント番号を設定します。

0～15 までの任意の値が設定できます。

3.3.52 19Eh H_CHaConditionCode (Host Channel a Condition Code)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	19Eh	H_CHaConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Condition Code		
				5: ConditionCode[1]			
				4: ConditionCode[0]			
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

ホスト動作時にチャネル CHa の転送完了結果を示します。

Bit7-4 ConditionCode[2:0]

チャネル CHa で転送が完了した場合の結果を示します。

Code	Meaning	Description
000	NoERROR ノーエラー	トランザクションがエラーなく完了した
001	STALL ストール	エンドポイントがストール PID を返した
010	DATAOVERRUN データ・オーバーラン	- 最大パッケージサイズを越えるデータパケットを受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します - IRP(TotalSize)を越えるデータ数を受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します * データパケットが最大パッケージサイズ以下で、データパケットに含まれたデータトグルが予期した値と一致していない場合、データ・オーバーランではなくトグルミスマッチとして処理します
011	DATAUNDERRUN データ・アンダーラン	- 最大パッケージサイズ未満のデータパケットを受信し、且つデータ数が IRP(TotalSize)に満たない * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します
100	RETRYERROR リトライエラー	- デバイスがトークンに規定時間以内に応答しない(IN)、または規定時間以内にハンドシェークを発信しない(OUT) - エンドポイントからのデータパケットに CRC エラーが含まれている - エンドポイントからのデータパケットにビット・スタッフィングエラーが含まれている - エンドポイントからの PID 検査ビットが、データ PID(IN)またはハンドシェーク(OUT)に失敗した - 受信 PID が無効または PID 値が定義されていない - エンドポイントからのデータパケットに含まれたデータトグルが、予期した値と一致していない (トグルミスマッチ)
その他	Reserved	

Bit3-0 Reserved

3. レジスタ詳細説明

3.3.53 1A0h H_CHbConfig_0(Host Channel b Configuration0)

3.3.54 1B0h H_CHcConfig_0(Host Channel c Configuration0)

3.3.55 1C0h H_CHdConfig_0(Host Channel d Configuration0)

3.3.56 1D0h H_CHeConfig_0(Host Channel e Configuration0)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	1A0h	CHbConfig_0	R/W	7: ACK_Cnt[3]	ACK Count		00h
	1B0h	CHcConfig_0		6: ACK_Cnt[2]			
	1C0h	CHdConfig_0		5: ACK_Cnt[1]			
	1D0h	CHeConfig_0		4: ACK_Cnt[0]			
			R/W	3: SpeedMode[1]	Speed Mode		
				2: SpeedMode[0]			
			R/W	1: Toggle	0: Toggle0	1: Toggle1	
			R/W	0: TranGo	0: Stand by	1: Transaction Start	

ホスト動作時にチャネル CHx{x=b-e}の基本設定を行います。

Bit7-4 ACK_Cnt [3:0]

チャネル CHx{x=b-e}で行われる転送において、ACK をカウントする数を設定します。

設定された値だけ ACK をカウントすると、H_CHx{x=b-e}IntStat レジスタの TranACK ビットがセットされます。

0000: 16 回の ACK をカウントします。

0001~1111: 1 回~15 回の ACK をカウントします。

Bit3-2 SpeedMode [1:0]

チャネル CHx{x=b-e}で転送を行うデバイスの動作モードを設定します。

00: HS モード — HS デバイスの時はこの設定にしてください。

01: FS モード — FS デバイスの時はこの設定にしてください。

10: Reserved — 本値の使用を禁止します。

11: LS モード — LS デバイスの時はこの設定にしてください。

Bit1 Toggle

トランザクションを開始する時のトグルシーケンスビットの初期値を設定します。また、トランザクション実行後、およびトランザクション完了後はトグルシーケンスビットの状態を示します。

0: トグル 0

1: トグル 1

Bit0

TranGo

このビットを“1”に設定するとチャネル CHx{x=b-e}のトランザクションを開始します。トランザクション開始後このビットを“0”にクリアするとトランザクション処理を停止させることができます。また、このビットはチャネル CHx{x=b-e}がトランザクション実行中か否かのステータスの意味も有します。

0: トランザクションを停止します（トランザクション停止中です）

1: トランザクションを開始します（トランザクション実行中です）

転送が H_CHx{x=b-e}TotalSize_HH~LL レジスタ で設定したバイト数完了した時点で H_CHx{x=b-e}IntStat レジスタの TotalSizeCmp ビットが “1” にセットされ、本ビットは自動的に“0”に戻ります。バルク転送、インタラプト転送では H_CHx{x=b-e}IntStat レジスタの ChangeCondition ビットがセットされたような場合は“0”にリセットされます。この時は、H_CHx{x=b-e}ConditionCode レジスタにその原因がセットされていますので参照して下さい。

また、このビットのクリアによる停止を行った時は処理最中のトランザクションが終了した時点で、H_CHx{x=b-e}IntStat レジスタの ChangeCondition ビットがセットされます。トランザクションが停止されても、FIFO 中のデータ、（残りの）トータルサイズ、チャネルに関する設定はそのままの状態となります。従って再びこのビットを“1”にセットすることによって、トランザクションを停止された時の続きから再開させることが可能です。（新たなトランザクションを行う場合は FIFO をクリアし、チャネル情報の設定をし直してください。）

3. レジスタ詳細説明

3.3.57 1A1h H_CHbConfig_1(Host Channel b Configuration1)

3.3.58 1B1h H_CHcConfig_1(Host Channel c Configuration1)

3.3.59 1C1h H_CHdConfig_1(Host Channel d Configuration1)

3.3.60 1D1h H_CHeConfig_1(Host Channel e Configuration1)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	1A1h	H_CHbConfig_1	R/W	7: TID[1]	Transaction ID		00h
	1B1h	H_CHcConfig_1		6: TID[0]			
	1C1h	H_CHdConfig_1	R/W	5: TranType[1]	Transfer Type		
	1D1h	H_CHeConfig_1		4: TranType[0]			
			R/W	3: AutoZerolen	0: Do nothing	1: Add Zerolen	
			R/W	2: Audio441	0: Disable	1: Enable	
			R/W	1: TotalSizeFree[1]	TotalSizeFree[1:0]		
				0: TotalSizeFree[0]			

ホスト動作時にチャネル CHx{x=b-e}の基本設定を行います。

Bit7-6 TID[1:0]

チャネル CHx{x=b-e}で発行するトランザクションの種類（OUT、IN）を設定します。

00: Reserved — 本値の使用を禁止します。

01: OUT — OUT トークンを発行します。

10: IN — IN トークンを発行します。

11: Reserved — 本値の使用を禁止します。

Bit5-4 TranType[1:0]

チャネル CHx{x=b-e}で行う転送の種別を設定します。

00: Reserved — 本値の使用を禁止します。

01: Isochronous — アイソクロナス転送を行います。

10: Bulk — バルク転送を行います。

11: Interrupt — インタラプト転送を行います。

Bit3 AutoZerolen

このビットに“1”を設定すると H_CHx{x=b-e}TotalSizeHH~LL レジスタで設定したサイズの転送がちょうど Max Packet Size で完了した際に、ゼロ長パケットを最後に自動付与します。OUT 転送の場合のみこのビットは有効となります。

Bit2 Audio441

このビットに“1”を設定するとオーディオクラス・アシスト機能が有効になります。オーディオクラス・アシスト機能はアイソクロナス転送でサンプリング周波数 44.1kHz の 16bit、2 チャンネル PCM データを 1ms 周期で送信する時に使用します。オーディオクラス・アシスト機能が有効の時は送信するデータパケットのサイズを自動で調整します。このときデータパケットのサイズを 176Byte として 9 回連続トランザクションを行った後、データパケットのサイズを 180Byte として 1 回トランザクションを行います。

Bit1-0 TotalSizeFree[1:0]

H_CHx{x=b-e}TotalSize_HH~LL レジスタによる転送の制御を設定します。

- 00: H_CHx{x=b-e}TotalSize_HH~LL レジスタで設定したサイズの転送が完了すると TranGo ビットをクリアし転送を終了します。そして、この時 TotalSizeCmp 割り込みを発生します。OUT 転送の場合、最後のデータパケットのサイズは MaxPktSize と TotalSize の小さい方の値となります。IN 転送の場合、期待する最後のデータパケットのサイズは MaxPktSize と TotalSize の小さい方の値となります。
- 01: H_CHx{x=b-e}TotalSize_HH~LL レジスタの値に関係なく転送が行われます。OUT 転送の場合、データパケットのサイズは MaxPktSize の値となります。IN 転送の場合、期待するデータパケットのサイズは MaxPktSize の値となります。この設定では H_CHx{x=b-e}TotalSize_HH~LL レジスタで設定したサイズの転送が完了しても TranGo ビットのクリアは行わず、TotalSizeCmp 割り込みも発生しません。
- 10: H_CHx{x=b-e}TotalSize_HH~LL レジスタで設定したサイズの転送が完了すると時 TotalSizeCmp 割り込みを発生します。この時 TranGo ビットはクリアされませんが TotalSize の値が “0”の間中このチャネルのトランザクションの発行は停止されます。OUT 転送の場合、データパケットのサイズは MaxPktSize と TotalSize の小さい方の値となります。IN 転送の場合、期待するデータパケットのサイズは MaxPktSize と TotalSize の小さい方の値となります。
- 11: Reserved

3. レジスタ詳細説明

3.3.61 1A2h-1A3h H_CHbMaxPktSize_H,L (Host Channel b Max Packet Size High, Low)

3.3.62 1B2h-1B3h H_CHcMaxPktSize_H,L (Host Channel c Max Packet Size High, Low)

3.3.63 1C2h-1C3h H_CHdMaxPktSize_H,L (Host Channel d Max Packet Size High, Low)

3.3.64 1D2h-1D3h H_CHeMaxPktSize_H,L (Host Channel e Max Packet Size High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	1A2h-1A3h	H_CHbMaxPktSize_H,L		15:	0:	1:	0000h
	1B2h-1B3h	H_CHcMaxPktSize_H,L		14:	0:	1:	
	1C2h-1C3h	H_CHdMaxPktSize_H,L		13:	0:	1:	
	1D2h-1D3h	H_CHeMaxPktSize_H,L		12:	0:	1:	
				11:	0:	1:	
				10:	0:	1:	
			R/W	9: MaxPktSize[9]	Max Packet Size		
				8: MaxPktSize[8]			
				7: MaxPktSize[7]			
				6: MaxPktSize[6]			
				5: MaxPktSize[5]			
				4: MaxPktSize[4]			
				3: MaxPktSize[3]			
				2: MaxPktSize[2]			
				1: MaxPktSize[1]			
				0: MaxPktSize[0]			

ホスト動作時にチャンネル CHx{x=b-e}の MaxPacketSize の設定を行います。

Bit15-10 Reserved

リザーブ・ビットには“1”を書き込まないで下さい。

Bit9-0 MaxPktSize[9:0]

チャンネル CHx{x=b-e}の MaxPacketSize を設定します。

このチャンネルをバルク転送用として使用する場合には、

FS 時 8, 16, 32, 64 バイト

HS 時 512 バイト

のいずれかに設定して下さい。

このチャンネルをインタラプト転送用として使用する場合は、

LS 時 8 バイトまで

FS 時 64 バイトまで

HS 時 1024 バイトまで

の任意の転送数が設定可能です。

このチャネルをアイソクロナス転送用として使用する場合は、

FS 時 1023 バイトまで

HS 時 1024 バイトまで

の任意の転送数が設定可能です。

それ以外の値の設定は禁止します。

3. レジスタ詳細説明

3.3.65 1A4h-1A5h H_CHbTotalSize_HH,HL (Host Channel b Total Size High-High, High-Low)

3.3.66 1A6h-1A7h H_CHbTotalSize_LH,LL (Host Channel b Total Size Low-High, Low-Low)

3.3.67 1B4h-1B5h H_CHcTotalSize_HH,HL (Host Channel c Total Size High-High, High-Low)

3.3.68 1B6h-1B7h H_CHcTotalSize_LH,LL (Host Channel c Total Size Low-High, Low-Low)

3.3.69 1C4h-1C5h H_CHdTotalSize_HH,HL (Host Channel d Total Size High-High, High-Low)

3.3.70 1C6h-1C7h H_CHdTotalSize_LH,LL (Host Channel d Total Size Low-High, Low-Low)

3.3.71 1D4h-1D5h H_CHeTotalSize_HH,HL (Host Channel e Total Size High-High, High-Low)

3.3.72 1D6h-1D7h H_CHeTotalSize_LH,LL (Host Channel e Total Size Low-High, Low-Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	1A4h-1A5h	H_CHbTotalSize_HH,HL	R/W	15: TotalSize[31]	Total Size High	0000h
	1B4h-1B5h	H_CHcTotalSize_HH,HL		14: TotalSize[30]		
	1C4h-1C5h	H_CHdTotalSize_HH,HL		13: TotalSize[29]		
	1D4h-1D5h	H_CHeTotalSize_HH,HL		12: TotalSize[28]		
				11: TotalSize[27]		
				10: TotalSize[26]		
				9: TotalSize[25]		
				8: TotalSize[24]		
				7: TotalSize[23]		
				6: TotalSize[22]		
				5: TotalSize[21]		
				4: TotalSize[20]		
				3: TotalSize[19]		
				2: TotalSize[18]		
				1: TotalSize[17]		
				0: TotalSize[16]		

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	1A6h-1A7h	H_CHbTotalSize_LH,LL	R/W	15: TotalSize[15]	Total Size Low	0000h
	1B6h-1B7h	H_CHcTotalSize_LH,LL		14: TotalSize[14]		
	1C6h-1C7h	H_CHdTotalSize_LH,LL		13: TotalSize[13]		
	1D6h-1D7h	H_CHeTotalSize_LH,LL		12: TotalSize[12]		
				11: TotalSize[11]		
				10: TotalSize[10]		
				9: TotalSize[9]		
				8: TotalSize[8]		
				7: TotalSize[7]		
				6: TotalSize[6]		
				5: TotalSize[5]		
				4: TotalSize[4]		
				3: TotalSize[3]		
				2: TotalSize[2]		
				1: TotalSize[1]		
				0: TotalSize[0]		

ホスト動作時にチャンネル CHx{x=b-e} で転送を行うデータのバイト数(最大 4,294,967,295byte:約 4Gbyte)を設定します。

H_CHx{x=b-e}Config_0 レジスタの TranGo ビットによりトランザクションが開始された後は、このレジスタをリードすることにより残りの転送数を読み出すことができます。

また、TotalSize = 0 で OUT トランザクションを実行するとゼロ長パケットが送信されます。

3. レジスタ詳細説明

3.3.73 1A8h H_CHbHubAdrs (Host Channel b Hub Address)

3.3.74 1B8h H_CHcHubAdrs (Host Channel c Hub Address)

3.3.75 1C8h H_CHdHubAdrs (Host Channel d Hub Address)

3.3.76 1D8h H_CHeHubAdrs (Host Channel e Hub Address)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	1A8h	H_CHbHubAdrs	R/W	7: HubAdrs[3]	Hub Address		00h
	1B8h	H_CHcHubAdrs		6: HubAdrs[2]			
	1C8h	H_CHdHubAdrs		5: HubAdrs[1]			
	1D8h	H_CHeHubAdrs		4: HubAdrs[0]			
				3:	0:	1:	
			R/W	2: Port[2]	Port Number		
				1: Port[1]			
				0: Port[0]			

ホスト動作時にチャンネル CHx{x=b-e}に接続するハブの設定を行います。

Bit7 HubAdrs[3:0]

チャンネル CHx{x=b-e}で転送を行うファンクションが接続されているハブの USB アドレスを設定します。

0～15 までの任意の値が設定できます。

Bit3 Reserved

Bit2-0 Port[2:0]

チャンネル CHx{x=b-e}で転送を行うファンクションが接続されているハブのポートナンバーを設定します。

0～7 までの任意の値が設定できます。

3.3.77 1A9h H_CHbFuncAdrs (Host Channel b Function Address)**3.3.78 1B9h H_CHcFuncAdrs (Host Channel c Function Address)****3.3.79 1C9h H_CHdFuncAdrs (Host Channel d Function Address)****3.3.80 1D9h H_CHeFuncAdrs (Host Channel e Function Address)**

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	1A9h	H_CHbFuncAdrs	R/W	7: FuncAdrs[3]	Function Address	00h
	1B9h	H_CHcFuncAdrs		6: FuncAdrs[2]		
	1C9h	H_CHdFuncAdrs		5: FuncAdrs[1]		
	1D9h	H_CHeFuncAdrs		4: FuncAdrs[0]		
			R/W	3: EP_Number[3]	Endpoint Number	
				2: EP_Number[2]		
				1: EP_Number[1]		
				0: EP_Number[0]		

ホスト動作時にチャネル CHx{x=b-e}で転送を行うファンクションのアドレス設定を行います。

Bit7-4 FuncAdrs[3:0]

チャネル CHx{x=b-e}が管理するエンドポイントを含むファンクションの USB アドレスを設定します。

0～15 までの任意の値が設定できます。

Bit3-0 EP_Number[3:0]

チャネル CHx{x=b-e}で転送を行うエンドポイント番号を設定します。

0～15 までの任意の値が設定できます。

3. レジスタ詳細説明

3.3.81 1AAh-1ABh H_CHbInterval_H,L(Host Channel b Interval High, Low)

3.3.82 1BAh-1BBh H_CHcInterval_H,L(Host Channel c Interval High, Low)

3.3.83 1CAh-1CBh H_CHdInterval_H,L(Host Channel d Interval High, Low)

3.3.84 1DAh-1DBh H_CHeInterval_H,L(Host Channel e Interval High, Low)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	1AAh-1ABh	H_CHbInterval_H,L		15:	0:	1:
	1BAh-1BBh	H_CHcInterval_H,L		14:	0:	1:
	1CAh-1CBh	H_CHdInterval_H,L		13:	0:	1:
	1DAh-1DBh	H_CHeInterval_H,L		12:	0:	1:
				11:	0:	1:
			R/W	10: Interval[10]	Transfer Interval	0000h
				9: Interval[9]		
				8: Interval[8]		
				7: Interval[7]		
				6: Interval[6]		
				5: Interval[5]		
				4: Interval[4]		
				3: Interval[3]		
				2: Interval[2]		
				1: Interval[1]		
				0: Interval[0]		

ホスト動作時にチャネル CHx{x=b-e}でインタラプト転送及びアイソクロナス転送を行う時のインターバル値を設定します。

Bit15-11 Reserved

Bit10-0 Interval[10:0]

インタラプト転送及びアイソクロナス転送のトークン発行間隔（周期）をこのレジスタにより指定します。下位 3 ビットはマイクロフレーム（125 μ s）単位で指定し、上位 7 ビットはフレーム（ms）単位で指定します。このレジスタの設定は H_CHx{x=b-e}Config1 レジスタの TranType ビットが“11”（インタラプト転送）の場合または TranType ビットが“01”（アイソクロナス転送）の場合のみ有効です。また、このレジスタの“0d0”の設定は無効となります。

トランザクションの再送時もこのレジスタの設定間隔で行います。

Interval[2:0] μ Frame — インターバルを 125 μ s 単位で指定します。1、2、4 マイクロフレームのいずれかに設定してください。それ以外の値の設定は禁止します。また、本ビットを設定する際は、Interval[10:3]はすべて“0”に設定してください。

Interval[10:3] Frame — インターバルを ms 単位で指定します。1～255 フレームの任意の値を設定できます。また、本ビットを設定する際は、Interval[2:0]はすべて“0”に設定してください。

3.3.85 1ACh H_CHbTranPause(Host Channel b Transaction Pause)**3.3.86 1BCh H_CHcTranPause(Host Channel c Transaction Pause)****3.3.87 1CCh H_CHdTranPause(Host Channel d Transaction Pause)****3.3.88 1DCh H_CHeTranPause(Host Channel e Transaction Pause)**

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	1ACh	H_CHbTranPause		7:	0:	1:	00h
	1BCh	H_CHcTranPause		6:	0:	1:	
	1CCh	H_CHdTranPause		5:	0:	1:	
	1DCh	H_CHeTranPause		4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: EnTranPause	0: Disable	1: Enable	
			R/W	0: TranPause	0: Do nothing	1: Transaction Pause	

Bit7-2 Reserved**Bit1 EnTranPause**

このビットに“1”を設定すると、H_CHx{x=b-e}IntStat レジスタの TranACK ビットがセットされる時、同時に H_CHx{x=b-e}TranPause レジスタの TranPause ビットが“1”にセットされます。

Bit0 TranPause

このビットが“1”の時、H_CHx{x=b-e}Config_0 の TranGo ビットが“1”に設定されていてもこのチャンネルではトランザクションは実行されません。このチャンネルがインタラプト転送あるいはアイソクロナス転送に設定されている場合、このビットが“1”の為にトランザクションが実行されなくても転送の周期は維持されます。

3. レジスタ詳細説明

3.3.89 1AEh H_CHbConditionCode (Host Channel b Condition Code)

3.3.90 1BEh H_CHcConditionCode (Host Channel c Condition Code)

3.3.91 1CEh H_CHdConditionCode (Host Channel d Condition Code)

3.3.92 1DEh H_CHeConditionCode (Host Channel e Condition Code)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	1AEh	H_CHbConditionCode		7:	0:	1:	00h
	1BEh	H_CHcConditionCode	R	6: ConditionCode[2]	Condition Code		
	1CEh	H_CHdConditionCode		5: ConditionCode[1]			
	1DEh	H_CHeConditionCode		4: ConditionCode[0]			
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

ホスト動作時にチャネル CHx{x=b-e}の転送完了結果を示します。

Bit7-4 ConditionCode[2:0]

チャネル CHx{x=b-e}で転送が完了した場合の結果を示します。

Code	Meaning	Description
000	NOERROR ノーエラー	トランザクションがエラーなく完了した
001	STALL ストール	エンドポイントがストール PID を返した
010	DATAOVERRUN データ・オーバーラン	- 最大パケットサイズを越えるデータパケットを受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します - RP(TotalSize)を越えるデータ数を受信した * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します * データパケットが最大パケットサイズ以下で、データパケットに含まれたデータトグルが予期した値と一致していない場合、データ・オーバーランではなくトグルミスマッチとして処理します
011	DATAUNDERRUN データ・アンダーラン	- 最大パケットサイズ未満のデータパケットを受信し、且つデータ数が IRP(TotalSize)に満たない * CRC エラーやビット・スタッフィングエラーが同時に検出された場合はリトライエラーとして処理します

Code	Meaning	Description
100	RETRYERROR リトライエラー	• デバイスがトークンに規定時間以内に応答しない(IN)、または規定時間以内にハンドシェークを発信しない(OUT) • エンドポイントからのデータパケットに CRC エラーが含まれている • エンドポイントからのデータパケットにビット・スタッフィングエラーが含まれている • エンドポイントからの PID 検査ビットが、データ PID(IN)またはハンドシェーク(OUT)に失敗した • 受信 PID が無効または PID 値が定義されていない • インタラプト転送のスプリットトランザクションにおいて ERR ハンドシェークを受信した • インタラプト転送のスプリットトランザクションにおいて NYET ハンドシェークを 3 回連続して受信した • エンドポイントからのデータパケットに含まれたデータトグルが、予期した値と一致していない (トグルミスマッチ)
110	BUFFEROVERRUN バッファ・オーバーラン	• アイソクロナス転送において FIFO の空き容量が最大パケットサイズ未満であるためトランザクションが行われなかった。
111	BUFFERUNDERRUN バッファ・アンダーラン	• アイソクロナス転送において FIFO の有効データ数が不足のためトランザクションが行われなかった。
その他	Reserved	

Bit3-0 Reserved

Appendix A リトルエンディアンの CPU との接続

本 LSI の内部バスはビッグエンディアンで構成されており、偶数アドレスが上位バイト、奇数アドレスが下位バイトとなっています。これに対して、リトルエンディアンの CPU に接続しての使用方法を説明致します。

<基板>

リトルエンディアンの CPU と本 LSI の端子は、データ・バスとライト制御信号につきましては端子名称のまま接続して下さい。すなわち CPU のデータバスのビット 15 からビット 8(上位バイト)には、本 LSI の CD15 から CD8 を接続し、CPU のデータ・バスのビット 7 からビット 0(即ち下位バイト)には本 LSI の CD7 から CD0 を接続してください。また、ライト信号につきましても、ハイ、ローをそのまま接続してください。

なお、ご使用になられる CPU によってはライト信号自身の仕様が異なりますので、注意してください。

<F/W>

リトルエンディアンの CPU にて、本 LSI を動作させる時には、まず以下の手順を行ってください。

- (1) CPUIF_MODE.CPU_Endian ビットを“1”にセット
このレジスタは 0x07E 番地にワードで割り当てられ、未初期化期間にのみ書き込み可能となっています。ハードリセット後、最初のアクセスで設定してください。CPU_Endian ビットは、bit10 及び bit2 のどちらか、または両方に“1”をセットすると、リトルエンディアン設定となります。
- (2) ダミーリードまたは CPUIF_MODE レジスタをダミーライト
このダミーアクセスによって、初期化期間が終了し、レジスタが下記レジスタマップに示す通りのアドレス定義となります。

なお、この設定を行った後、すべての内部レジスタが、Char もしくは Short にてアクセス可能となります。また CPU の DMAC を使用してのアクセスに対しても問題は御座いません（以下の表を参照）。

例 USB から 01_02_03_04_05_06 と順にデータを受け取った場合の FIFO_Rd_0/1 レジスタへのアクセス

Short にてアクセス	CPU のアクセス方法			
	ビッグエンディアン		リトルエンディアン	
	CD[15:8]	CD[7:0]	CD[15:8]	CD[7:0]
1st	01	02	02	01
2nd	03	04	04	03
3rd	05	06	06	05

ただし、Short より大きなサイズのレジスタにつきましては、Short にて分割アクセスして頂き、CPU のメモリ上にてキャストしてご使用ください。CPU_Endian=“1”(Little Endian)とした時に、偶数番地と奇数番地が入れ替わるレジスタを下記の表に示します。これらのレジスタでは、CD[15:0]に、16 ビットレジスタがそのままの並びで読み書きできます。

BigEndian

0x014	WakeupTim_H
0x015	WakeupTim_L
0x024	FIFO_RdRemain_H
0x025	FIFO_RdRemain_L
0x026	FIFO_WrRemain_H
0x027	FIFO_WrRemain_L
0x030	RAM_RdAdrs_H
0x031	RAM_RdAdrs_L
0x038	RAM_WrAdrs_H
0x039	RAM_WrAdrs_L
0x064	DMA_Remain_H
0x065	DMA_Remain_L
0x068	DMA_Count_HH
0x069	DMA_Count_HL
0x06A	DMA_Count_LH
0x06B	DMA_Count_LL
0x080	AREA0StartAdrs_H
0x081	AREA0StartAdrs_L
0x082	AREA0EndAdrs_H
0x083	AREA0EndAdrs_L
0x084	AREA1StartAdrs_H
0x085	AREA1StartAdrs_L
0x086	AREA1EndAdrs_H
0x087	AREA1EndAdrs_L
0x088	AREA2StartAdrs_H
0x089	AREA2StartAdrs_L
0x08A	AREA2EndAdrs_H
0x08B	AREA2EndAdrs_L
0x08C	AREA3StartAdrs_H
0x08D	AREA3StartAdrs_L
0x08E	AREA3EndAdrs_H
0x08F	AREA3EndAdrs_L
0x090	AREA4StartAdrs_H
0x091	AREA4StartAdrs_L
0x092	AREA4EndAdrs_H
0x093	AREA4EndAdrs_L
0x094	AREA5StartAdrs_H
0x095	AREA5StartAdrs_L
0x096	AREA5EndAdrs_H
0x097	AREA5EndAdrs_L
0x0BC	D_AlarmIN_IntStat_H
0x0BD	D_AlarmIN_IntStat_L
0x0BE	D_AlarmOUT_IntStat_H

LittleEndian

0x014	WakeupTim_L
0x015	WakeupTim_H
0x024	FIFO_RdRemain_L
0x025	FIFO_RdRemain_H
0x026	FIFO_WrRemain_L
0x027	FIFO_WrRemain_H
0x030	RAM_RdAdrs_L
0x031	RAM_RdAdrs_H
0x038	RAM_WrAdrs_L
0x039	RAM_WrAdrs_H
0x064	DMA_Remain_L
0x065	DMA_Remain_H
0x068	DMA_Count_HL
0x069	DMA_Count_HH
0x06A	DMA_Count_LL
0x06B	DMA_Count_LH
0x080	AREA0StartAdrs_L
0x081	AREA0StartAdrs_H
0x082	AREA0EndAdrs_L
0x083	AREA0EndAdrs_H
0x084	AREA1StartAdrs_L
0x085	AREA1StartAdrs_H
0x086	AREA1EndAdrs_L
0x087	AREA1EndAdrs_H
0x088	AREA2StartAdrs_L
0x089	AREA2StartAdrs_H
0x08A	AREA2EndAdrs_L
0x08B	AREA2EndAdrs_H
0x08C	AREA3StartAdrs_L
0x08D	AREA3StartAdrs_H
0x08E	AREA3EndAdrs_L
0x08F	AREA3EndAdrs_H
0x090	AREA4StartAdrs_L
0x091	AREA4StartAdrs_H
0x092	AREA4EndAdrs_L
0x093	AREA4EndAdrs_H
0x094	AREA5StartAdrs_L
0x095	AREA5StartAdrs_H
0x096	AREA5EndAdrs_L
0x097	AREA5EndAdrs_H
0x0BC	D_AlarmIN_IntStat_L
0x0BD	D_AlarmIN_IntStat_H
0x0BE	D_AlarmOUT_IntStat_L

Appendix B S1R72V17 と S1R72V27 の相違点

		S1R72V17	S1R72V27
CPU-IF	AC スペック	(※1)	
	Bus モード設定手順	(※2)	
	Bus 8bit モード	有り	無し
	DMA シングルワード転送	無し	有り
	DMA ライト FIFO ライト	ライトキャッシュ無し	ライトキャッシュ有り (※3)
	FIFO_ByteWr レジスタ	無し	有り
	WrRAM_Door 使用時の制限	無し	有り (※4)
USB	USB Host Isochronous 転送サポート	無し	有り
その他	RevisionNum	0x10	0x08

- ※1 AC スペックの違いについては、『S1R72V17 仕様書』および、『S1R72V27 データシート』をご覧ください。
- ※2 Bus モード設定手順の違いについての詳細は『S1R72V17 仕様書』6.6 章、および『S1R72V27 テクニカルマニュアル』「1.6.2 CPUIF のモード設定」をご覧ください。
- ※3 S1R72V27 はライトキャッシュが実装されているため、FIFO エリア切り替え時に制限事項があります。詳細は『S1R72V27 テクニカルマニュアル』「1.6.3.1.3 FIFO アクセス(ライト)」をご覧ください。
- ※4 WrRAM_Door 使用時の制限については、『S1R72V27 テクニカルマニュアル』「1.5.5.2 RAM アクセスの方法(RAM_WrDoor)」をご覧ください。

改訂履歴 (Rev.1.10)

年月日	改訂内容			
	Rev.	項 (旧版)	種別	内 容
07/09/07	0.79	全項	新規	新規制定
07/09/21	0.80		改訂	修正内容 修正前 修正後
		P28		誤記修正 (1.2.7.2.7) NonJ を検知したら、f/w により InSUSPEND をクリアして、IN_SUSPEND スレートを抜けて下さい。 NonJ を検知したら、f/w により InSUSPEND をクリアして、IN_SUSPEND ステートを抜けて下さい。
		P147		追記 (2.1.3) - LineStateChanged の Description を追記
		P149		誤記修正 (2.1.5) IDE の転送 DMA の転送
		P149		誤記修正 (2.1.5) 該当するチャネルの FIFO 領域 該当するエリアの FIFO 領域
		P152		誤記修正 (2.1.8) LineStateChanged EnLineStateChanged
		P152		追記 (2.1.8) - EnLineStateChanged の Description を追記
		P152		追記 (2.1.8) EnVBUS_Err ビットは SLEEP 中も有効です。 EnVBUS_Err ビット、 <i>EnLineStateChanged</i> ビットは SLEEP 中も有効です。
		P161		誤記修正 (2.1.16) DP_A および DM_A DP および DM
		P161		誤記修正 (2.1.16) HTM のオペレーションモードを設定します。 MTM のオペレーションモードを設定します。
		P168		誤記修正 (2.1.25) その際に、FIFO_RdRemain_H レジスタを先にアクセスして下さい。 (記述削除)
		P169		誤記修正 (2.1.26) その際に、FFIO_WrRemain_H レジスタを先にアクセスして下さい。 (記述削除)
		P176		誤記修正 (2.1.33) RAM_WrDoor_H,L RAM_WrDoor_0,1
		P177		誤記修正 (2.1.34~ 2.1.35) CSW0 エリア CSW エリア
		P177		誤記修正 (2.1.34~ 2.1.35) CBW0,1 エリア CBW エリア
		P236		誤記修正 (2.3.1) H_NegoControl_1.PortSpeed= "HS"であり、かつ、 H_NegoControl_0.HostState== "SUSPEND"の時は、本検出機能は使用できません。 (記述削除)
		P286		章追加 (2.3.85 ~ 2.3.88) - 1ACh~1DCh H_ChxTranPause (x=b-e)
		P290		追加 - Appendix B S1R72V17 と S1R72V27 の相違点

改訂履歴

07/11/20	1.00	P4	改訂	修正	エンドポイント EP0 を FIFO の AREA0 領域に関連付けて使用して下さい。	エンドポイント EP0 を FIFO の任意の AREA $\{x=0-5\}$ 領域に関連付けて使用して下さい。	
		P4			AREA0 領域に割り当てる FIFO 領域を、アドレスで指定します。	AREA $\{x=0-5\}$ 領域に割り当てる FIFO 領域を、アドレスで指定します。	
		P4		修正	5 本までの独立したエンドポイントを使用できます。	5 本までの独立したエンドポイントを同時にアクティブにして使用できます。	
		P5		修正	「0.レジスタ」を参照して下さい。	「2.レジスタ詳細説明」を参照して下さい。	
		P114		修正	本 LSI の各ポートの FIFO には、	本 LSI の FIFO には、	
		P114		削除	なお、各ポートの FIFO は独立していますので、ポートをまたがっての制限はありません。		
		P115		追加	-	なお、書き込み開始アドレスに以下の制限があります。 ①書き込み開始アドレスを、RAM_WrAdrs_H,L == 0x0100 より下位とする。 ②書き込み開始アドレスを、RAM_WrAdrs_H,L[1:0] == 0b00（すなわち 4Byte 境界）とする。	
		P119		追加	-	AREAnFIFO_Clr にて、FIFO クリアする。	
		P128		修正	-	初期レジスタマップ 0x27E を削除	
		P156		削除	※参照したポートが SLEEP ステートであって、他のポートが ACTIVE ステートにある場合、PM_State は 0b01 となります。		
		P161		修正	改行部分の体裁修正（それに伴い、以降のページ番号も変更。）		
		P172		追加	-	AREAnFIFO_Clr にて、FIFO クリアする。	
		P176		追加	-	なお、書き込み開始アドレスに以下の制限があります。 ①書き込み開始アドレスを、RAM_WrAdrs_H,L == 0x0100 より下位とする。 ②書き込み開始アドレスを、RAM_WrAdrs_H,L[1:0] == 0b00（すなわち 4Byte 境界）とする。	
		P182		修正	-	Bit7,3,2,1 の R/W 記載漏れを追記。	
		P187		修正	-	ClkSelect レジスタ定義を修正。	
		P290		追加	-	WrRAM_Door 使用時の制限	
				修正	および『S1R72V27 テクニカルマニュアル』1.6 章をご覧ください。	および『S1R72V27 テクニカルマニュアル』「1.6.2 CPUIF のモード設定」をご覧ください。	
				修正	詳細は『S1R72V27 テクニカルマニュアル』1.6 章を参照してください。	詳細は『S1R72V27 テクニカルマニュアル』「1.6.3.1.3 FIFO アクセス(ライト)」をご覧ください。	

				追加	-	※4 WrRAM_Door 使用時の制限については、『S1R72V27 テクニカルマニュアル』「1.5.5.2 RAM アクセスの方法(RAM_WrDoor)」をご覧ください。
08/10/25	1.10		改訂	修正内容	修正前	修正後
		P67		削除	この機能を使用する場合はチャンネル CH0 にジョインする FIFO 領域として AREA0 を選択してください。	-
				修正	FIFO 領域 (AREA0StartAdrs_H,L,,AREA0EndAdrs_H,L)、 FIFO 領域ジョイン (AREA0Join_1.JoinEP0CH0)	FIFO 領域 (AREAn{n=0-5}StartAdrs_H,L, AREAn{n=0-5}EndAdrs_H,L)、 FIFO 領域ジョイン (AREAn{n=0-5}Join_1.JoinEP0CH0)
		P73		削除	この機能を使用する場合はチャンネル CHa にジョインする FIFO 領域として AREA1 を選択してください。	-
				修正	FIFO 領域 (AREA1StartAdrs_H,L,AREA1EndAdrs_H,L)、FIFO 領域ジョイン (AREA1Join_1.JoinEPaCHa)	FIFO 領域 (AREAn{n=0-5}StartAdrs_H,L, AREAn{n=0-5}EndAdrs_H,L)、 FIFO 領域ジョイン (AREAn{n=0-5}Join_1.JoinEPaCHa)
		P85		修正	3 回連続で切断を検出した場合に切断検出と判断します	3 つの連続した uSOF で切断を検出した場合に切断状態と判断します
		P164		修正	レジスタマップ 0x075 CPU_Config Reset=0x00	レジスタマップ 0x075 CPU_Config Reset=0x01
		P263		追加	-	(bit4-bit0) 同様に、同時に 2 つ以上の IN のエンドポイントでバルクオンリーサポート機能を有効にしないで下さい。
		P268		削除	フレームナンバーを取得する場合は、FrameNumber_H と FrameNumber_L レジスタを対でアクセスする必要があります。その際に FrameNumber_H レジスタを先にアクセスして下さい。	-
		P272		修正	RcvEP0SETUP ビットが“1”である間、このビットは“0”にクリアできません。	D_SETUP_Control.ProtectEP0 ビットが“1”である間、このビットは“0”にクリアできません。
				修正	RcvEP0SETUP ビットが“1”である間はこのビットを“1”にセットできません。	D_SETUP_Control.ProtectEP0 ビットが“1”である間はこのビットを“1”にセットできません。
		P273		修正	RcvEP0SETUP ビットが“1”である間はこのビットを“0”にクリアすることはできません。	D_SETUP_Control.ProtectEP0 ビットが“1”である間はこのビットを“0”にクリアすることはできません。

改訂履歴

		P274		修正	RcvEP0SETUP ビットが“1”である間はこのビットを“1”にセットすることはできません。	D_SETUP_Control.ProtectEP0 ビットが“1”である間はこのビットを“1”にセットすることはできません。
		P283 P284 P285 P286		修正	(Bit Symbol) 7-0	(Bit Symbol) 15-8
		P287		削除	(Bit3) H_NegoControl_1.PortSpeed= =“HS”であり、かつ、 H_NegoControl_0.HostState== “SUSPEND”の時は、本検出機能は使用できません。	-
				追加	-	(Bit1) また、ホストステートがRESET の状態において、H/W はこのビットによりデバイスからのチャープ信号の有無を判断します。このビットが“1”にセットされている場合は H_NegoControl_1.PortSpeed を “00b(HS)”に設定し、以降 HS デバイスに対する処理を行います。この為、デバイスが切断された時には必ずこのビットを“0”にクリアしてください。
		P295, P308, P323		修正	H_BO_SupportControl	H_CHaBO_SupportCtl
		P297, P345		追加	-	表“BufferOverrun”、“BufferUnderrun”を追加
		P305		追加	-	(Bit7) 以下の操作を行うとき、事前にこのビットによりホストステート管理サポート機能の実行を停止する必要があります。
				削除	(Bit7) 事前にこのビットによりホストステート管理サポート機能の実行を停止する必要があります。	-
				追加	-	(Bit7) “AutoMode を上書きするとき”を AutoModeCancel を実行する必要がある場合に追加。
		P309		修正	それ以外は CH0 に関する設定は必要ありません。	また、H_CH0Config_1.TID = 00b と設定してください。
		P311		削除	フレームナンバーを取得する場合は、H_FrameNumber_H と H_FrameNumber_L レジスタを対でアクセスする必要があります。その際に H_FrameNumber_H レジスタを先にアクセスして下さい。	-

		P314, P323, P329, P330, P331, P332	修正	Supportot	Support
		P329	修正	H_BO_SupportIntStat レジスタ	H_CHaIntStat レジスタ
		P335	修正	TranCmp ビットが “1” にセットされ、本ビットは自動的に“0”に戻ります。 H_CHx{x=b-e}IntStat レジスタの ChangeCondition ビットがセットされたような場合は“0”にリセットされます。	TotalSizeCmp ビットが “1” にセットされ、本ビットは自動的に“0”に戻ります。 バルク転送、インタラプト転送では H_CHx{x=b-e}IntStat レジスタの ChangeCondition ビットがセットされたような場合は“0”にリセットされます。
		P336	追加	-	Bit2 Audio441 (ビット記述追加)
			追加	-	Bit1 TotalSizeFree[1] (ビット記述追加)
			追加	-	Bit5-4 TranType[1:0]=01 Isochronous を追加
		P341	削除	読み出す場合は、 H_CHbTotalSize_HH,HL を先に、 H_CHbTotalSize_HL,LL を後の順番にアクセスしてください。	-
		P344	修正	(Ofst Adrs) 1AAh-1BAh	(Ofst Adrs) 1AAh-1ABh
			修正	(Description) Interrupt Transfer Interval	(Description) Transfer Interval
			修正	インタラプト転送 (2 箇所)	インタラプト転送及びアイソクロナス転送 (2 箇所)
			修正	H_CHx{x=b-e}Config1 レジスタの TranType ビットが“11” (インタラプト転送)の場合のみ有効です。	H_CHx{x=b-e}Config1 レジスタの TranType ビットが“11” (インタラプト転送)の場合または TranType ビットが“01” (アイソクロナス転送)の場合のみ有効です。
			以下余白		

セイコーエプソン株式会社
半導体事業部 IC 営業部

<IC 国内営業グループ>

東京 〒191-8501 東京都日野市日野 421-8
TEL (042) 587-5313 (直通) FAX (042) 587-5116

大阪 〒541-0059 大阪市中央区博労町 3-5-1 エプソン大阪ビル 15F
TEL (06) 6120-6000 (代表) FAX (06) 6120-6100

ドキュメントコード : 411340101
2008 年 10 月作成