

S1R72V05

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件の許諾を行うものではありません。
3. 特性値のの数値の大小は、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、一般民生用です。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。

総則

適用範囲

本仕様書は、セイコーエプソン株式会社 半導体事業部製 USB2.0 コントローラ「S1R720V05**」に適用します。

目次

1.	概要	1
2.	特長	1
3.	ブロック図	2
3.1.	Device Transceiver Macro (DTM)	3
3.2.	Host Transceiver Macro (HTM)	3
3.3.	Oscillator & PLL60	3
3.4.	Device Serial Interface Engine (Device SIE)	3
3.5.	Host Serial Interface Engine (Host SIE)	3
3.6.	Port Selector	3
3.7.	USB FIFO / USB FIFO Controller	3
3.8.	Media FIFO / Media FIFO Controller	3
3.9.	CPU I/F Controller	3
3.10.	DMA Controller	3
3.11.	IDE Master Controller	3
3.12.	TestMUX	3
4.	端子配置図	4
5.	端子機能説明	6
6.	機能説明	9
6.1.	デバイス/ホスト・レジスタマップ選択	10
6.1.1.	レジスタマップ選択方法	10
6.1.2.	ポート状態変化検出ステータス	10
6.1.2.1.	ポート状態変化検出ステータスの使用例	10
6.1.2.1.1.	VBUS_B端子変化ステータス	10
6.1.2.1.2.	信号ライン変化ステータス	12
6.2.	USBデバイス制御	13
6.2.1.	エンドポイント	13
6.2.2.	トランザクション	14
6.2.2.1.	SETUPトランザクション	16
6.2.2.2.	バルク/インタラプトOUTトランザクション	17
6.2.2.3.	バルク/インタラプトINトランザクション	19
6.2.2.4.	PINGトランザクション	19
6.2.3.	コントロール転送	20
6.2.3.1.	セットアップステージ	21
6.2.3.2.	データステージ/ステータスステージ	21
6.2.3.3.	自動アドレス設定機能	21
6.2.3.4.	デスク립タ返信機能	21
6.2.4.	バルク転送/インタラプト転送	22
6.2.5.	データフロー	22
6.2.5.1.	OUT転送	22
6.2.5.2.	IN転送	22
6.2.6.	バルクオンリーサポート	23
6.2.6.1.	CSWサポート	23
6.2.6.2.	CSWサポート	23
6.2.7.	オート・ネゴシエーション機能	25
6.2.7.1.	DISABLE	25
6.2.7.2.	IDLE	26
6.2.7.3.	WAIT_TIM3US	26
6.2.7.4.	WAIT_CHIRP	26

6.2.7.5.	WAIT_RSTEND	26
6.2.7.6.	DET_SUSPEND	26
6.2.7.7.	IN_SUSPEND	26
6.2.7.8.	CHK_EVENT	26
6.2.7.9.	WAIT_RESTORE	27
6.2.7.10.	ERR	27
6.2.7.11.	各ネゴシエーション機能の単体説明	28
6.2.7.11.1.	サスペンド検出 (HSモード)	28
6.2.7.11.2.	サスペンド検出 (FSモード)	30
6.2.7.11.3.	リセット検出 (HSモード)	31
6.2.7.11.4.	リセット検出 (FSモード)	32
6.2.7.11.5.	HS Detection Handshake.....	33
6.2.7.11.5.1.	FSのダウンストリームポートに繋がれた場合.....	34
6.2.7.11.5.2.	HSのダウンストリームポートに繋がれた場合.....	36
6.2.7.11.5.3.	スヌーズ中にリセットされた場合.....	39
6.2.7.11.6.	レジュームの発行.....	41
6.2.7.11.7.	レジュームの検出.....	43
6.2.7.11.8.	ケーブル挿入.....	45
6.3.	USBホスト制御.....	47
6.3.1.	チャンネル	47
6.3.1.1.	チャンネル概要	47
6.3.1.2.	コントロール専用チャンネル	48
6.3.1.3.	汎用チャンネル	49
6.3.1.4.	チャンネル使用例	50
6.3.1.4.1.	ストレージデバイスを1台接続した場合	50
6.3.1.4.2.	コミュニケーションデバイスを1台接続した場合	51
6.3.1.4.3.	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	52
6.3.1.4.4.	ハブを介してストレージデバイスを2台接続した場合	53
6.3.2.	スケジューリング	54
6.3.3.	トランザクション	55
6.3.3.1.	SETUPトランザクション	56
6.3.3.2.	バルクOUTトランザクション	57
6.3.3.3.	インタラプトOUTトランザクション	58
6.3.3.4.	バルクINトランザクション	59
6.3.3.5.	インタラプトINトランザクション	61
6.3.3.6.	PINGトランザクション	61
6.3.3.7.	low-speed (LS) トランザクション	63
6.3.3.8.	スプリットトランザクション	64
6.3.4.	コントロール転送	65
6.3.4.1.	セットアップステージ	66
6.3.4.2.	データステージ/ステータスステージ	66
6.3.4.3.	コントロール転送サポート機能	66
6.3.5.	バルク転送/インタラプト転送	70
6.3.6.	データフロー	70
6.3.6.1.	OUT転送	70
6.3.6.2.	IN転送	70
6.3.7.	ゼロ長パケット自動発行機能	71
6.3.7.1.	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	71
6.3.8.	バルクオンリーサポート機能	72
6.3.9.	ホストステート管理サポート機能	76

6.3.9.1.	ホストステート	76
6.3.9.1.1.	IDLE	79
6.3.9.1.2.	WAIT_CONNECT	79
6.3.9.1.3.	DISABLED	79
6.3.9.1.4.	RESET	79
6.3.9.1.5.	OPERATIONAL	80
6.3.9.1.6.	SUSPEND	81
6.3.9.1.7.	RESUME	81
6.3.9.2.	検出機能	82
6.3.9.2.1.	VBUS異常検出	82
6.3.9.2.2.	切断検出	83
6.3.9.2.2.1.	HSデバイスが切断された場合	83
6.3.9.2.2.2.	FSまたはLSデバイスが切断された場合	84
6.3.9.2.3.	リモート・ウェークアップ検出	85
6.3.9.2.3.1.	HSデバイスが接続されている場合	85
6.3.9.2.3.2.	FSデバイスが接続されている場合	86
6.3.9.2.3.3.	LSデバイスが接続されている場合	87
6.3.9.2.4.	デバイスチャープ検出機能	88
6.3.9.2.4.1.	正しいデバイスのChirpを検出した場合	88
6.3.9.2.4.2.	異常なデバイスのChirpを検出した場合	89
6.3.9.2.5.	ポートエラー検出	90
6.3.9.3.	ホストステート管理サポート機能の単体説明	91
6.3.9.3.1.	GoIDLE	91
6.3.9.3.2.	GoWAIT_CONNECT	92
6.3.9.3.2.1.	FSデバイスが接続された場合	92
6.3.9.3.2.2.	LSデバイスが接続された場合	94
6.3.9.3.3.	GoDISABLED	95
6.3.9.3.3.1.	HSデバイスが接続されている場合	95
6.3.9.3.3.2.	FSデバイスが接続されている場合	96
6.3.9.3.3.3.	LSデバイスが接続されている場合	97
6.3.9.3.4.	GoRESET	98
6.3.9.3.4.1.	HSデバイスに対するリセット	98
6.3.9.3.4.2.	異常なデバイスのChirp検出	101
6.3.9.3.4.2.1.	チャープ完了ディセーブル (H_NegoControl_1.DisChirpFinish) 設定が0の場合	101
6.3.9.3.4.2.2.	チャープ完了ディセーブル (H_NegoControl_1.DisChirpFinish) 設定が1の場合	103
6.3.9.3.4.3.	FSデバイスに対するリセット	106
6.3.9.3.4.4.	LSデバイスに対するリセット	107
6.3.9.3.5.	GoOPERATIONAL	108
6.3.9.3.6.	GoSUSPEND	109
6.3.9.3.6.1.	HSデバイスが接続されている場合	109
6.3.9.3.6.2.	FSデバイスが接続されている場合	110
6.3.9.3.6.3.	LSデバイスが接続されている場合	111
6.3.9.3.7.	GoRESUME	112
6.3.9.3.7.1.	HSデバイスが接続されている場合	112
6.3.9.3.7.2.	FSデバイスが接続されている場合	114
6.3.9.3.7.3.	LSデバイスが接続されている場合	115
6.3.9.3.8.	GoWAIT_CONNECTtoDIS	116
6.3.9.3.9.	GoWAIT_CONNECTtoOP	117
6.3.9.3.9.1.	HSデバイスが接続されている場合	117
6.3.9.3.9.2.	FSまたはLSデバイスが接続されている場合	118

6.3.9.3.10.	GoRESETtoOP.....	119
6.3.9.3.10.1.	HSデバイスが接続されている場合.....	119
6.3.9.3.10.2.	FSまたはLSデバイスが接続されている場合.....	120
6.3.9.3.11.	GoSUSPENDtoOP.....	121
6.3.9.3.12.	GoRESUMEtoOP.....	122
6.4.	メディアデータ転送機能.....	123
6.4.1.	メディアデータ.....	123
6.4.2.	メディアデータ転送.....	123
6.4.3.	消費電力の削減.....	124
6.5.	パワーマネージメント機能.....	125
6.5.1.	SLEEP (スリープ).....	126
6.5.2.	SNOOZE (スヌーズ).....	126
6.5.3.	ACTIVE60 (アクティブ60).....	127
6.5.4.	ACT_DEVICE (アクトデバイス).....	127
6.5.5.	ACT_HOST (アクトホスト).....	127
6.6.	FIFO管理.....	128
6.6.1.	デバイスFIFO管理.....	128
6.6.1.1.	FIFOメモリマップ.....	128
6.6.1.2.	デスクリプタエリアの使用方法.....	129
6.6.1.2.1.	デスクリプタエリアへのデータの書き込み.....	129
6.6.1.2.2.	デスクリプタエリアでのデータステージ (IN) の実行.....	129
6.6.1.3.	CBWエリアの使用方法.....	129
6.6.1.3.1.	CBWエリアへの受信.....	129
6.6.1.3.2.	CBWエリアからのデータの読み出し.....	129
6.6.1.4.	CSWエリアの使用方法.....	129
6.6.1.4.1.	CSWエリアからの送信.....	129
6.6.1.4.2.	CSWエリアへのデータの書き込み.....	130
6.6.1.5.	FIFOへのアクセス方法.....	131
6.6.1.5.1.	FIFOへのアクセス方法 (RAM_Rd).....	131
6.6.1.5.2.	FIFOへのアクセス方法 (RAM_WrDoor).....	131
6.6.1.5.3.	FIFOへのアクセス方法 (レジスタアクセス).....	131
6.6.1.5.4.	FIFOへのアクセス方法 (DMA).....	131
6.6.1.5.5.	FIFOへのアクセス方法 (IDE).....	131
6.6.1.5.6.	FIFOへのアクセス制限.....	131
6.6.2.	ホストFIFO管理.....	133
6.6.2.1.	FIFOメモリマップ.....	133
6.6.2.2.	CBWエリアの使用方法.....	133
6.6.2.2.1.	CBWエリアからの送信.....	134
6.6.2.2.2.	CBWエリアへのデータの書き込み.....	134
6.6.2.3.	CSWエリアの使用方法.....	134
6.6.2.3.1.	CSWエリアへの受信.....	134
6.6.2.3.2.	CSWエリアからのデータの読み出し.....	134
6.6.2.4.	FIFOへのアクセス方法.....	135
6.6.2.4.1.	FIFOへのアクセス方法 (RAM_Rd).....	135
6.6.2.4.2.	FIFOへのアクセス方法 (RAM_WrDoor).....	135
6.6.2.4.3.	FIFOへのアクセス方法 (レジスタアクセス).....	135
6.6.2.4.4.	FIFOへのアクセス方法 (DMA).....	135
6.6.2.4.5.	FIFOへのアクセス方法 (IDE).....	135
6.6.2.4.6.	FIFOへのアクセス制限.....	136
6.6.3.	メディアFIFO管理.....	137

6.6.3.1.	メディアFIFO	137
6.6.3.2.	FIFOへのアクセス方法	137
6.6.3.2.1.	FIFOへのアクセス方法 (レジスタアクセス)	137
6.6.3.2.2.	FIFOへのアクセス方法 (DMA)	137
6.6.3.2.3.	FIFOへのアクセス方法 (IDE)	137
6.6.3.2.4.	FIFOへのアクセス制限	137
6.7.	CPUIF	138
6.7.1.	モード切り替え	138
6.7.2.	モード切り替えに際して	138
6.7.2.1.	16bit BE modeをご使用の場合	138
6.7.2.2.	8bit modeをご使用の場合	140
6.7.3.	ブロック構成	140
6.7.3.1.	REG (S1R72V05 Registers)	140
6.7.3.1.1.	同期レジスタアクセス (ライト)	140
6.7.3.1.2.	同期レジスタアクセス (リード)	140
6.7.3.1.3.	FIFOアクセス (ライト)	140
6.7.3.1.4.	FIFOアクセス (リード)	141
6.7.3.1.5.	FIFOアクセスの端数処理	142
6.7.3.1.6.	RAM_Rdアクセス	144
6.7.3.1.7.	非同期レジスタアクセス (ライト)	144
6.7.3.1.8.	非同期レジスタアクセス (リード)	144
6.7.3.2.	DMA0/DMA1 (DMA ch.0 / ch.1)	144
6.7.3.2.1.	基本機能	144
6.7.3.2.2.	端子設定	146
6.7.3.2.3.	カウントモード (ライト)	147
6.7.3.2.4.	カウントモード (リード)	148
6.7.3.2.5.	フリーランモード (ライト)	150
6.7.3.2.6.	フリーランモード (リード)	150
6.7.3.2.7.	REQアサートカウントオプション (ライト)	150
6.7.3.2.8.	REQアサートカウントオプション (リード)	151
6.7.3.2.9.	DMAのFIFOアクセス端数処理	151
6.8.	IDE I/F	152
6.8.1.	IDEタスクファイルレジスタへのアクセス	152
6.8.1.1.	IDEタスクファイルレジスタからのリード	152
6.8.1.2.	IDEタスクファイルレジスタへのライト	152
6.8.1.3.	IDEタスクファイルレジスタへのシーケンシャルライト	152
6.8.1.4.	IDEタスクファイルレジスタからのオートステータスレジスタリード	152
6.8.2.	PIOアクセス	153
6.8.2.1.	PIOリードDMA	153
6.8.2.2.	PIOライトDMA	153
6.8.3.	Multi-Word DMA	153
6.8.3.1.	Multi-Word DMAリード	153
6.8.3.2.	Multi-Word DMAライト	153
6.8.4.	Ultra DMA	155
6.8.4.1.	Ultra DMAリード	155
6.8.4.2.	Ultra DMAライト	155
6.8.5.	IDE転送モードの設定について	156
6.9.	バウンダリスキャン (JTAG)	157
6.9.1.	対応インストラクション	157
6.9.2.	DEVICE_CODEIに関して	157

6.9.3.	バウンダリスキャン除外端子	157
7.	レジスタ.....	158
7.1.	デバイス/ホスト共通レジスタマップ.....	158
7.2.	デバイス・レジスタマップ.....	164
7.3.	ホスト・レジスタマップ	170
7.4.	デバイス/ホスト共通レジスタ詳細説明.....	176
7.4.1.	00h <i>MainIntStat</i> (<i>Main Interrupt Status</i>)	176
7.4.2.	01h <i>DeviceIntStat</i> (<i>Device Interrupt Status</i>).....	177
7.4.3.	02h <i>HostIntStat</i> (<i>Host Interrupt Status</i>)	179
7.4.4.	03h <i>CPU_IntStat</i> (<i>CPU Interrupt Status</i>)	181
7.4.5.	04h <i>IDE_IntStat</i> (<i>IDE Interrupt Status</i>)	182
7.4.6.	05h <i>MediaFIFO_IntStat</i> (<i>Media FIFO Interrupt Status</i>).....	183
7.4.7.	06h~0Fh Reserved ()	184
7.4.8.	10h <i>MainIntEnb</i> (<i>Main Interrupt Enable</i>)	185
7.4.9.	11h <i>DeviceIntEnb</i> (<i>Device Interrupt Enable</i>)	186
7.4.10.	12h <i>HostIntEnb</i> (<i>Host Interrupt Enable</i>)	187
7.4.11.	13h <i>CPU_IntEnb</i> (<i>CPU Interrupt Enable</i>)	188
7.4.12.	14h <i>IDE_IntEnb</i> (<i>IDE Interrupt Enable</i>)	189
7.4.13.	15h <i>MediaFIFO_IntEnb</i> (<i>Media FIFO Interrupt Enable</i>).....	190
7.4.14.	16h~1Fh Reserved ()	191
7.4.15.	20h <i>RevisionNum</i> (<i>Revision Number</i>)	192
7.4.16.	21h <i>ChipReset</i> (<i>Chip Reset</i>)	193
7.4.17.	22h <i>PM_Control_0</i> (<i>Power Management Control 0</i>).....	194
7.4.18.	23h <i>PM_Control_1</i> (<i>Power Management Control 1</i>).....	196
7.4.19.	24h <i>WakeupTim_H</i> (<i>Wakeup Time High</i>)	197
7.4.20.	25h <i>WakeupTim_L</i> (<i>Wakeup Time Low</i>)	197
7.4.21.	26h <i>H_USB_Control</i> (<i>Host USB Control</i>)	198
7.4.22.	27h <i>H_XcvrControl</i> (<i>Host Xcvr Control</i>)	199
7.4.23.	28h <i>D_USB_Status</i> (<i>Device USB Status</i>)	200
7.4.24.	29h <i>H_USB_Status</i> (<i>Host USB Status</i>)	201
7.4.25.	2A~2Fh Reserved ()	202
7.4.26.	30h <i>FIFO_Rd_0</i> (<i>FIFO Read 0</i>)	203
7.4.27.	31h <i>FIFO_Rd_1</i> (<i>FIFO Read 1</i>)	203
7.4.28.	32h <i>FIFO_Wr_0</i> (<i>FIFO Write 0</i>)	204
7.4.29.	33h <i>FIFO_Wr_1</i> (<i>FIFO Write 1</i>)	204
7.4.30.	34h <i>FIFO_RdRemain_H</i> (<i>FIFO Read Remain High</i>).....	205
7.4.31.	35h <i>FIFO_RdRemain_L</i> (<i>FIFO Read Remain Low</i>)	205
7.4.32.	36h <i>FIFO_WrRemain_H</i> (<i>FIFO Write Remain High</i>).....	206
7.4.33.	37h <i>FIFO_WrRemain_L</i> (<i>FIFO Write Remain Low</i>).....	206
7.4.34.	38h <i>FIFO_ByteRd</i> (<i>FIFO Byte Read</i>)	207
7.4.35.	39~3Fh Reserved ()	208
7.4.36.	40h <i>RAM_RdAdrs_H</i> (<i>RAM Read Address High</i>)	209
7.4.37.	41h <i>RAM_RdAdrs_L</i> (<i>RAM Read Address Low</i>)	209
7.4.38.	42h <i>RAM_RdControl</i> (<i>RAM Read Control</i>)	210
7.4.39.	43h <i>RAM_RdCount</i> (<i>RAM Read Counter</i>)	211
7.4.40.	44h <i>RAM_WrAdrs_H</i> (<i>RAM Write Address High</i>)	212
7.4.41.	45h <i>RAM_WrAdrs_L</i> (<i>RAM Write Address Low</i>)	212
7.4.42.	46h <i>RAM_WrDoor_0</i> (<i>RAM Write Door 0</i>)	213
7.4.43.	47h <i>RAM_WrDoor_1</i> (<i>RAM Write Door 1</i>)	213
7.4.44.	48h <i>MediaFIFO_Control</i> (<i>Media FIFO Control</i>).....	214

7. 4. 45.	49h ClrAllMediaFIFO_Join (Clear All Media FIFO Join).....	215
7. 4. 46.	4Ah MediaFIFO_Join (Media FIFO Join)	216
7. 4. 47.	4Bh Reserved ()	217
7. 4. 48.	4C~4Fh Reserved ()	218
7. 4. 49.	50h RAM_Rd_00 (RAM Read 00)	219
7. 4. 50.	51h RAM_Rd_01 (RAM Read 01)	219
7. 4. 51.	52h RAM_Rd_02 (RAM Read 02)	219
7. 4. 52.	53h RAM_Rd_03 (RAM Read 03)	219
7. 4. 53.	54h RAM_Rd_04 (RAM Read 04)	219
7. 4. 54.	55h RAM_Rd_05 (RAM Read 05)	219
7. 4. 55.	56h RAM_Rd_06 (RAM Read 06)	219
7. 4. 56.	57h RAM_Rd_07 (RAM Read 07)	219
7. 4. 57.	58h RAM_Rd_08 (RAM Read 08)	219
7. 4. 58.	59h RAM_Rd_09 (RAM Read 09)	219
7. 4. 59.	5Ah RAM_Rd_0A (RAM Read 0A)	219
7. 4. 60.	5Bh RAM_Rd_0B (RAM Read 0B)	219
7. 4. 61.	5Ch RAM_Rd_0C (RAM Read 0C)	219
7. 4. 62.	5Dh RAM_Rd_0D (RAM Read 0D)	219
7. 4. 63.	5Eh RAM_Rd_0E (RAM Read 0E)	219
7. 4. 64.	5Fh RAM_Rd_0F (RAM Read 0F)	219
7. 4. 65.	60h RAM_Rd_10 (RAM Read 10)	219
7. 4. 66.	61h RAM_Rd_11 (RAM Read 11)	219
7. 4. 67.	62h RAM_Rd_12 (RAM Read 12)	219
7. 4. 68.	63h RAM_Rd_13 (RAM Read 13)	219
7. 4. 69.	64h RAM_Rd_14 (RAM Read 14)	219
7. 4. 70.	65h RAM_Rd_15 (RAM Read 15)	219
7. 4. 71.	66h RAM_Rd_16 (RAM Read 16)	219
7. 4. 72.	67h RAM_Rd_17 (RAM Read 17)	219
7. 4. 73.	68h RAM_Rd_18 (RAM Read 18)	219
7. 4. 74.	69h RAM_Rd_19 (RAM Read 19)	219
7. 4. 75.	6Ah RAM_Rd_1A (RAM Read 1A)	219
7. 4. 76.	6Bh RAM_Rd_1B (RAM Read 1B)	219
7. 4. 77.	6Ch RAM_Rd_1C (RAM Read 1C)	219
7. 4. 78.	6Dh RAM_Rd_1D (RAM Read 1D)	219
7. 4. 79.	6Eh RAM_Rd_1E (RAM Read 1E)	219
7. 4. 80.	6Fh RAM_Rd_1F (RAM Read 1F)	219
7. 4. 81.	70h Reserved ()	220
7. 4. 82.	71h DMAO_Config (DMAO Config)	221
7. 4. 83.	72h DMAO_Control (DMAO Control)	222
7. 4. 84.	73h Reserved ()	223
7. 4. 85.	74h DMAO_Remain_H (DMAO FIFO Remain High)	224
7. 4. 86.	75h DMAO_Remain_L (DMAO FIFO Remain Low)	224
7. 4. 87.	76h~77h Reserved ()	225
7. 4. 88.	78h DMAO_Count_HH (DMAO Transfer Byte Counter High/High).....	226
7. 4. 89.	79h DMAO_Count_HL (DMAO Transfer Byte Counter High/Low).....	226
7. 4. 90.	7Ah DMAO_Count_LH (DMAO Transfer Byte Counter Low/High).....	226
7. 4. 91.	7Bh DMAO_Count_LL (DMAO Transfer Byte Counter Low/Low).....	226
7. 4. 92.	7Ch DMAO_RdData_0 (DMAO Read Data 0)	227
7. 4. 93.	7Dh DMAO_RdData_1 (DMAO Read Data 1)	227
7. 4. 94.	7Eh DMAO_WrData_0 (DMAO Write Data 0)	228

7. 4. 95.	7Fh DMA0_WrData_1 (DMA0 Write Data 1)	228
7. 4. 96.	80h Reserved 0	229
7. 4. 97.	81h DMA1_Config (DMA0 Config)	230
7. 4. 98.	82h DMA1_Control (DMA1 Control)	231
7. 4. 99.	83h Reserved 0	232
7. 4. 100.	84h DMA1_Remain_H (DMA1 FIFO Remain High)	233
7. 4. 101.	85h DMA1_Remain_L (DMA1 FIFO Remain Low)	233
7. 4. 102.	86h~87h Reserved 0	234
7. 4. 103.	88h DMA1_Count_HH (DMA1 Transfer Byte Counter High/High).....	235
7. 4. 104.	89h DMA1_Count_HL (DMA1 Transfer Byte Counter High/Low).....	235
7. 4. 105.	8Ah DMA1_Count_LH (DMA1 Transfer Byte Counter Low/High).....	235
7. 4. 106.	8Bh DMA1_Count_LL (DMA1 Transfer Byte Counter Low/Low).....	235
7. 4. 107.	8Ch DMA1_RdData_0 (DMA1 Read Data 0)	237
7. 4. 108.	8Dh DMA1_RdData_1 (DMA1 Read Data 1)	237
7. 4. 109.	8Eh DMA1_WrData_0 (DMA1 Write Data 0)	238
7. 4. 110.	8Fh DMA1_WrData_1 (DMA1 Write Data 1)	238
7. 4. 111.	90h IDE_Status (IDE Status)	239
7. 4. 112.	91h IDE_Control (IDE Control)	240
7. 4. 113.	92h IDE_Config_0 (IDE Configuration 0)	241
7. 4. 114.	93h IDE_Config_1 (IDE Configuration 1)	242
7. 4. 115.	94h IDE_Rmod (IDE Register Mode)	243
7. 4. 116.	95h IDE_Tmod (IDE Transfer Mode)	244
7. 4. 117.	96h IDE_Umod (IDE Ultra-DMA Transfer Mode)	245
7. 4. 118.	97h~99h Reserved 0	246
7. 4. 119.	9Ah IDE_CRC_H (IDE CRC High)	247
7. 4. 120.	9Bh IDE_CRC_L (IDE CRC Low)	247
7. 4. 121.	9Ch Reserved 0	248
7. 4. 122.	9Dh IDE_Count_H (IDE Transfer Byte Counter High).....	249
7. 4. 123.	9Eh IDE_Count_M (IDE Transfer Byte Counter Middle).....	249
7. 4. 124.	9Fh IDE_Count_L (IDE Transfer Byte Counter Low).....	249
7. 4. 125.	A0h IDE_RegAdrs (IDE Register Address)	250
7. 4. 126.	A1h Reserved 0	251
7. 4. 127.	A2h IDE_RdRegValue_0 (IDE Register Read Value 0).....	252
7. 4. 128.	A3h IDE_RdRegValue_1 (IDE Register Read Value 1).....	252
7. 4. 129.	A4h IDE_WrRegValue_0 (IDE Register Write Value 0).....	253
7. 4. 130.	A5h IDE_WrRegValue_1 (IDE Register Write Value 1).....	253
7. 4. 131.	A6h IDE_SeqWrRegControl (IDE Sequential Register Write Control).....	254
7. 4. 132.	A7h IDE_SeqWrRegCnt (IDE Sequential Register Write Counter).....	255
7. 4. 133.	A8h IDE_SeqWrRegAdrs (IDE Sequential Register Write Address FIFO).....	256
7. 4. 134.	A9h IDE_SeqWrRegValue (IDE Sequential Register Write Value FIFO).....	257
7. 4. 135.	AAh~ABh Reserved 0	258
7. 4. 136.	ACh IDE_RegConfig (IDE Register Configuration).....	259
7. 4. 137.	ADh~AFh Reserved 0	260
7. 4. 138.	B0h Reserved 0	261
7. 4. 139.	B1h <i>HostDeviceSel (Host Device Select)</i>	262
7. 4. 140.	B2h Reserved 0	263
7. 4. 141.	B3h <i>ModeProtect (Mode Protection)</i>	264
7. 4. 142.	B4h Reserved 0	265
7. 4. 143.	B5h <i>ClkSelect (Clock Select)</i>	266
7. 4. 144.	B6h Reserved 0	267

7.4.145.	B7h <i>ChipConfig (Chip Configuration)</i>	268
7.4.146.	B8h Reserved ()	269
7.4.147.	B9h <i>CPU_ChgEndian (CPU Change Endian)</i>	270
7.4.148.	BAh~DFh Reserved ()	271
7.5.	デバイス・レジスタ詳細説明	272
7.5.1.	E0h <i>D_SIE_IntStat (Device SIE Interrupt Status)</i>	272
7.5.2.	E1h Reserved ()	275
7.5.3.	E2h <i>D_FIFO_IntStat (Device FIFO Interrupt Status)</i>	276
7.5.4.	E3h <i>D_BulkIntStat (Device Bulk Interrupt Status)</i>	277
7.5.5.	E4h <i>D_EPrIntStat (Device EPr Interrupt Status)</i>	278
7.5.6.	E5h <i>D_EP0IntStat (Device EP0 Interrupt Status)</i>	279
7.5.7.	E6h <i>D_EPaIntStat (Device EPa Interrupt Status)</i>	280
7.5.8.	E7h <i>D_EPbIntStat (Device EPb Interrupt Status)</i>	281
7.5.9.	E8h <i>D_EPcIntStat (D_EPc Interrupt Status)</i>	282
7.5.10.	E9h~EFh Reserved ()	283
7.5.11.	F0h <i>D_SIE_IntEnb (Device SIE Interrupt Enable)</i>	284
7.5.12.	F1h Reserved ()	285
7.5.13.	F2h <i>D_FIFO_IntEnb (Device FIFO Interrupt Enable)</i>	286
7.5.14.	F3h <i>D_BulkIntEnb (Device Bulk Interrupt Enable)</i>	287
7.5.15.	F4h <i>D_EPrIntEnb (Device EPr Interrupt Enable)</i>	288
7.5.16.	F5h <i>D_EP0IntEnb (Device EP0 Interrupt Enable)</i>	289
7.5.17.	F6h <i>D_EPaIntEnb (Device EPa Interrupt Enable)</i>	290
7.5.18.	F7h <i>D_EPbIntEnb (Device EPb Interrupt Enable)</i>	291
7.5.19.	F8h <i>D_EPcIntEnb (Device EPc Interrupt Enable)</i>	292
7.5.20.	F9h~FFh Reserved ()	293
7.5.21.	100h <i>D_Reset (Device Reset)</i>	294
7.5.22.	101h Reserved ()	295
7.5.23.	102h <i>D_NegoControl (Device Nego Control)</i>	296
7.5.24.	103h Reserved ()	298
7.5.25.	104h <i>D_ClrAllEPnJoin (Device Clear All EPn Join)</i>	299
7.5.26.	105h <i>D_XcvrControl (Device Xcvr Control)</i>	300
7.5.27.	106h <i>D_USB_Test (Device USB_Test)</i>	301
7.5.28.	107h Reserved ()	303
7.5.29.	108h <i>D_EPnControl (Device Endpoint Control)</i>	304
7.5.30.	109h <i>D_EPrFIFO_Clr (Device Endpoint FIFO Clear)</i>	305
7.5.31.	10Ah <i>D_BulkOnlyControl (Device BulkOnly Control)</i>	306
7.5.32.	10Bh <i>D_BulkOnlyConfig (Device BulkOnly Configuration)</i>	307
7.5.33.	10C~10Fh Reserved ()	308
7.5.34.	110h <i>D_EPOSETUP_0 (Device EPO SETUP 0)</i>	309
7.5.35.	111h <i>D_EPOSETUP_1 (Device EPO SETUP 1)</i>	309
7.5.36.	112h <i>D_EPOSETUP_2 (Device EPO SETUP 2)</i>	309
7.5.37.	113h <i>D_EPOSETUP_3 (Device EPO SETUP 3)</i>	309
7.5.38.	114h <i>D_EPOSETUP_4 (Device EPO SETUP 4)</i>	309
7.5.39.	115h <i>D_EPOSETUP_5 (Device EPO SETUP 5)</i>	309
7.5.40.	116h <i>D_EPOSETUP_6 (Device EPO SETUP 6)</i>	309
7.5.41.	117h <i>D_EPOSETUP_7 (Device EPO SETUP 7)</i>	309
7.5.42.	118h <i>D_USB_Address (Device USB Address)</i>	310
7.5.43.	119h Reserved ()	311
7.5.44.	11Ah <i>D_SETUP_Control (Device SETUP Control)</i>	312
7.5.45.	11Bh~11Dh Reserved ()	313

7. 5. 46.	11Eh D_FrameNumber_H (Device FrameNumber High).....	314
7. 5. 47.	11Fh D_FrameNumber_L (Device FrameNumber Low).....	314
7. 5. 48.	120h D_EPOMaxSize (Device EPO Max Packet Size).....	315
7. 5. 49.	121h D_EPOControl (Device EPO Control)	316
7. 5. 50.	122h D_EPOControlIN (Device EPO Control IN).....	317
7. 5. 51.	123h D_EPOControlOUT (Device EPO Control OUT).....	319
7. 5. 52.	124h Reserved ()	320
7. 5. 53.	125h D_EPOJoin (Device EndPoint0 Join)	321
7. 5. 54.	126h~12Fh Reserved ()	322
7. 5. 55.	130h D_EPaMaxSize_H (Device EPa Max Packet Size High).....	323
7. 5. 56.	131h D_EPaMaxSize_L (Device EPa Max Packet Size Low).....	323
7. 5. 57.	132h D_EPaConfig_0 (Device EPa Configuration 0).....	324
7. 5. 58.	133h Reserved ()	325
7. 5. 59.	134h D_EPaControl (Device EPa Control)	326
7. 5. 60.	135h D_EPJoin (Device End Point a Join)	328
7. 5. 61.	136h~13Fh Reserved ()	329
7. 5. 62.	140h D_EPbMaxSize_H (Device EPb Max Packet Size High).....	330
7. 5. 63.	141h D_EPbMaxSize_L (Device EPb Max Packet Size Low).....	330
7. 5. 64.	142h D_EPbConfig_0 (Device EPb Configuration 0).....	331
7. 5. 65.	143h Reserved ()	332
7. 5. 66.	144h D_EPbControl (Device EPb Control)	333
7. 5. 67.	145h D_EPbJoin (Device End Point b Join)	335
7. 5. 68.	146h~14Fh Reserved ()	336
7. 5. 69.	150h D_EPcMaxSize_H (Device EPc Max Packet Size High).....	337
7. 5. 70.	151h D_EPcMaxSize_L (Device EPc Max Packet Size Low).....	337
7. 5. 71.	152h D_EPcConfig_0 (Device EPc Configuration 0).....	338
7. 5. 72.	153h Reserved ()	339
7. 5. 73.	154h D_EPcControl (Device EPc Control)	340
7. 5. 74.	155h D_EPcJoin (Device End Point c Join)	342
7. 5. 75.	156h~15Fh Reserved ()	343
7. 5. 76.	160h D_DescAdrs_H (Device Descriptor Address High).....	344
7. 5. 77.	161h D_DescAdrs_L (Device Descriptor Address Low).....	344
7. 5. 78.	162h D_DescSize_H (Device Descriptor Size High).....	345
7. 5. 79.	163h D_DescSize_L (Device Descriptor Size Low).....	345
7. 5. 80.	164h~16Fh Reserved ()	346
7. 5. 81.	170h D_DMA0_FIFO_Control (Device DMA0 FIFO Control).....	347
7. 5. 82.	171h Reserved ()	348
7. 5. 83.	172h D_DMA1_FIFO_Control (Device DMA1 FIFO Control).....	349
7. 5. 84.	173h~17Fh Reserved ()	350
7. 5. 85.	180h~183h Reserved ()	351
7. 5. 86.	184h D_EPaStartAdrs_H (Device Endpoint a Start Address High).....	352
7. 5. 87.	185h D_EPaStartAdrs_L (Device Endpoint a Start Address Low).....	352
7. 5. 88.	186h~187h Reserved ()	353
7. 5. 89.	188h D_EPbStartAdrs_H (Device Endpoint b Start Address High).....	354
7. 5. 90.	189h D_EPbStartAdrs_L (Device Endpoint b Start Address Low).....	354
7. 5. 91.	18Ah~18Bh Reserved ()	355
7. 5. 92.	18Ch D_EPcStartAdrs_H (Device Endpoint c Start Address High).....	356
7. 5. 93.	18Dh D_EPcStartAdrs_L (Device Endpoint c Start Address Low).....	356
7. 5. 94.	18Eh D_EPcEndAdrs_H (Device Endpoint c End Address High).....	357
7. 5. 95.	18Fh D_EPcEndAdrs_L (Device Endpoint c End Address Low).....	357

7.5.96.	190h~1FFh Reserved 0	358
7.6.	ホストレジスタ詳細説明	359
7.6.1.	E0h H_SIE_IntStat_0 (Host SIE Interrupt Status 0)	359
7.6.2.	E1h H_SIE_IntStat_1 (SIE Host Interrupt Status 1)	360
7.6.3.	E2h H_FIFO_IntStat (Host FIFO Interrupt Status)	361
7.6.4.	E3h H_FrameIntStat (Host Frame Interrupt Status)	362
7.6.5.	E4h H_CHrIntStat (Host CHr Interrupt Status)	363
7.6.6.	E5h H_CHOIntStat (Host CHO Interrupt Status)	364
7.6.7.	E6h H_CHaIntStat (Host CHa Interrupt Status)	367
7.6.8.	E7h H_CHbIntStat (Host CHb Interrupt Status)	370
7.6.9.	E8h H_CHcIntStat (Host CHc Interrupt Status)	372
7.6.10.	E9h H_CHdIntStat (Host CHd Interrupt Status)	374
7.6.11.	EAh H_CHeIntStat (Host CHe Interrupt Status)	376
7.6.12.	EBh~EFh Reserved 0	378
7.6.13.	F0h H_SIE_IntEnb_0 (Host SIE Interrupt Enable 0)	379
7.6.14.	F1h H_SIE_IntEnb_1 (SIE Host Interrupt Enable 1)	380
7.6.15.	F2h H_FIFO_IntEnb (Host FIFO Interrupt Enable)	381
7.6.16.	F3h H_FrameIntEnb (Host Frame Interrupt Enable)	382
7.6.17.	F4h H_CHrIntEnb (Host CHr Interrupt Enable)	383
7.6.18.	F5h H_CHOIntEnb (Host CHO Interrupt Enable)	384
7.6.19.	F6h H_CHaIntEnb (Host CHa Interrupt Enable)	385
7.6.20.	F7h H_CHbIntEnb (Host CHb Interrupt Enable)	386
7.6.21.	F8h H_CHcIntEnb (Host CHc Interrupt Enable)	387
7.6.22.	F9h H_CHdIntEnb (Host CHd Interrupt Enable)	388
7.6.23.	FAh H_CHeIntEnb (Host CHe Interrupt Enable)	389
7.6.24.	FBh~FFh Reserved 0	390
7.6.25.	100h H_Reset (Host Reset)	391
7.6.26.	101h Reserved 0	392
7.6.27.	102h H_NegoControl_0 (Host NegoControl 0)	393
7.6.28.	103h Reserved 0	394
7.6.29.	104h H_NegoControl_1 (Host NegoControl 1)	395
7.6.30.	105h Reserved 0	396
7.6.31.	106h H_USB_Test (Host USB_Test)	397
7.6.32.	107h Reserved 0	399
7.6.33.	108h H_CHnControl (Host CHn Control)	400
7.6.34.	109h H_CHrFIFO_Clr (Host CHr FIFO Clear)	401
7.6.35.	10Ah H_ClrAllCHnJoin (Host Clear All CHn Join)	402
7.6.36.	10B~10Fh Reserved 0	403
7.6.37.	110h H_CHOSETUP_0 (Host CHO SETUP 0)	404
7.6.38.	111h H_CHOSETUP_1 (Host CHO SETUP 1)	404
7.6.39.	112h H_CHOSETUP_2 (Host CHO SETUP 2)	404
7.6.40.	113h H_CHOSETUP_3 (Host CHO SETUP 3)	404
7.6.41.	114h H_CHOSETUP_4 (Host CHO SETUP 4)	404
7.6.42.	115h H_CHOSETUP_5 (Host CHO SETUP 5)	404
7.6.43.	116h H_CHOSETUP_6 (Host CHO SETUP 6)	404
7.6.44.	117h H_CHOSETUP_7 (Host CHO SETUP 7)	404
7.6.45.	118h~11Dh Reserved 0	405
7.6.46.	11Eh H_FrameNumber_H (Host FrameNumber High)	406
7.6.47.	11Fh H_FrameNumber_L (Host FrameNumber Low)	406
7.6.48.	120h H_CHOConfig_0 (Host Channel 0 Configuration0)	407

7. 6. 49.	121h H_CH0Config_1 (Host Channel 0 Configuration1).....	408
7. 6. 50.	122h Reserved 0	409
7. 6. 51.	123h H_CH0MaxPktSize (Host Channel 0 Max Packet Size).....	410
7. 6. 52.	124h~125h Reserved 0	411
7. 6. 53.	126h H_CH0TotalSize_H (Host Channel 0 Total Size High).....	412
7. 6. 54.	127h H_CH0TotalSize_L (Host Channel 0 Total Size Low).....	412
7. 6. 55.	128h H_CH0HubAdrs (Host Channel 0 Hub Address).....	413
7. 6. 56.	129h H_CH0FuncAdrs (Host Channel 0 Function Address).....	414
7. 6. 57.	12Ah Reserved 0	415
7. 6. 58.	12Bh CTL_SupportControl (Host ControlTransfer Support Control).....	416
7. 6. 59.	12Ch~12Dh Reserved 0	417
7. 6. 60.	12Eh H_CH0ConditionCode (Host Channel 0 Condition Code).....	418
7. 6. 61.	12Fh H_CH0Join (Host Channel 0 Join)	419
7. 6. 62.	130h H_CHaConfig_0 (Host Channel a Configuration0).....	420
7. 6. 63.	131h H_CHaConfig_1 (Host Channel a Configuration1).....	421
7. 6. 64.	132h H_CHaMaxPktSize_H (Host Channel a Max Packet Size High).....	422
7. 6. 65.	133h H_CHaMaxPktSize_L (Host Channel a Max Packet Size Low).....	422
7. 6. 66.	134h H_CHaTotalSize_HH (Host Channel a Total Size High-High).....	423
7. 6. 67.	135h H_CHaTotalSize_HL (Host Channel a Total Size High-Low).....	423
7. 6. 68.	136h H_CHaTotalSize_LH (Host Channel a Total Size Low-High).....	423
7. 6. 69.	137h H_CHaTotalSize_LL (Host Channel a Total Size Low-Low).....	423
7. 6. 70.	138h H_CHaHubAdrs (Host Channel a Hub Address).....	425
7. 6. 71.	139h H_CHaFuncAdrs (Host Channel a Function Address).....	426
7. 6. 72.	13Ah H_BO_SupportControl (Host Bulk Only Transfer Support Control).....	427
7. 6. 73.	13Bh H_CSW_RcvDataSize (Host CSW Receive Data Size).....	428
7. 6. 74.	13Ch H_OUT_EP_Control (Host OUT Endpoint Control).....	429
7. 6. 75.	13Dh H_IN_EP_Control (Host IN Endpoint Control).....	430
7. 6. 76.	13Eh H_CHaConditionCode (Host Channel a Condition Code).....	431
7. 6. 77.	13Fh H_CHaJoin (Host Channel a Join)	432
7. 6. 78.	140h H_CHbConfig_0 (Host Channel b Configuration0).....	433
7. 6. 79.	141h H_CHbConfig_1 (Host Channel b Configuration1).....	434
7. 6. 80.	142h H_CHbMaxPktSize_H (Host Channel b Max Packet Size High).....	435
7. 6. 81.	143h H_CHbMaxPktSize_L (Host Channel b Max Packet Size Low).....	435
7. 6. 82.	144h H_CHbTotalSize_HH (Host Channel b Total Size High-High).....	436
7. 6. 83.	145h H_CHbTotalSize_HL (Host Channel b Total Size High-Low).....	436
7. 6. 84.	146h H_CHbTotalSize_LH (Host Channel b Total Size Low-High).....	436
7. 6. 85.	147h H_CHbTotalSize_LL (Host Channel b Total Size Low-Low).....	436
7. 6. 86.	148h H_CHbHubAdrs (Host Channel b Hub Address).....	438
7. 6. 87.	149h H_CHbFuncAdrs (Host Channel b Function Address).....	439
7. 6. 88.	14Ah H_CHbInterval_H (Host Channel b Interval High).....	440
7. 6. 89.	14Bh H_CHbInterval_L (Host Channel b Interval Low).....	440
7. 6. 90.	14Ch~14Dh Reserved 0	441
7. 6. 91.	14Eh H_CHbConditionCode (Host Channel b Condition Code).....	442
7. 6. 92.	14Fh H_CHbJoin (Host Channel b Join)	444
7. 6. 93.	150h H_CHcConfig_0 (Host Channel c Configuration0).....	445
7. 6. 94.	151h H_CHcConfig_1 (Host Channel c Configuration1).....	446
7. 6. 95.	152h H_CHcMaxPktSize_H (Host Channel c Max Packet Size High).....	447
7. 6. 96.	153h H_CHcMaxPktSize_L (Host Channel c Max Packet Size Low).....	447
7. 6. 97.	154h H_CHcTotalSize_HH (Host Channel c Total Size High-High).....	448
7. 6. 98.	155h H_CHcTotalSize_HL (Host Channel c Total Size High-Low).....	448

7. 6. 99.	156h H_CHcTotalSize_LH (Host Channel c Total Size Low-High).....	448
7. 6. 100.	157h H_CHcTotalSize_LL (Host Channel c Total Size Low-Low).....	448
7. 6. 101.	158h H_CHcHubAdrs (Host Channel c Hub Address).....	450
7. 6. 102.	159h H_CHcFuncAdrs (Host Channel c Function Address).....	451
7. 6. 103.	15Ah H_CHcInterval_H (Host Channel c Interval High).....	452
7. 6. 104.	15Bh H_CHcInterval_L (Host Channel c Interval Low).....	452
7. 6. 105.	15Ch~15Dh Reserved ()	453
7. 6. 106.	15Eh H_CHcConditionCode (Host Channel c Condition Code).....	454
7. 6. 107.	15Fh H_CHcJoin (Host Channel c Join)	456
7. 6. 108.	160h H_CHdConfig_0 (Host Channel d Configuration0).....	457
7. 6. 109.	161h H_CHdConfig_1 (Host Channel d Configuration1).....	458
7. 6. 110.	162h H_CHdMaxPktSize_H (Host Channel d Max Packet Size High).....	459
7. 6. 111.	163h H_CHdMaxPktSize_L (Host Channel d Max Packet Size Low).....	459
7. 6. 112.	164h H_CHdTotalSize_HH (Host Channel d Total Size High-High).....	460
7. 6. 113.	165h H_CHdTotalSize_HL (Host Channel d Total Size High-Low).....	460
7. 6. 114.	166h H_CHdTotalSize_LH (Host Channel d Total Size Low-High).....	460
7. 6. 115.	167h H_CHdTotalSize_LL (Host Channel d Total Size Low-Low).....	460
7. 6. 116.	168h H_CHdHubAdrs (Host Channel d Hub Address).....	462
7. 6. 117.	169h H_CHdFuncAdrs (Host Channel d Function Address).....	463
7. 6. 118.	16Ah H_CHdInterval_H (Host Channel d Interval High).....	464
7. 6. 119.	16Bh H_CHdInterval_L (Host Channel d Interval Low).....	464
7. 6. 120.	16Ch~16Dh Reserved ()	465
7. 6. 121.	16Eh H_CHdConditionCode (Host Channel d Condition Code).....	466
7. 6. 122.	16Fh H_CHdJoin (Host Channel d Join)	468
7. 6. 123.	170h H_CHeConfig_0 (Host Channel e Configuration0).....	469
7. 6. 124.	171h H_CHeConfig_1 (Host Channel e Configuration1).....	470
7. 6. 125.	172h H_CHeMaxPktSize_H (Host Channel e Max Packet Size High).....	471
7. 6. 126.	173h H_CHeMaxPktSize_L (Host Channel e Max Packet Size Low).....	471
7. 6. 127.	174h H_CHeTotalSize_HH (Host Channel e Total Size High-High).....	472
7. 6. 128.	175h H_CHeTotalSize_HL (Host Channel e Total Size High-Low).....	472
7. 6. 129.	176h H_CHeTotalSize_LH (Host Channel e Total Size Low-High).....	472
7. 6. 130.	177h H_CHeTotalSize_LL (Host Channel e Total Size Low-Low).....	472
7. 6. 131.	178h H_CHeHubAdrs (Host Channel e Hub Address).....	474
7. 6. 132.	179h H_CHeFuncAdrs (Host Channel e Function Address).....	475
7. 6. 133.	17Ah H_CHeInterval_H (Host Channel e Interval High).....	476
7. 6. 134.	17Bh H_CHeInterval_L (Host Channel e Interval Low).....	476
7. 6. 135.	17Ch~17Dh Reserved ()	477
7. 6. 136.	17Eh H_CHeConditionCode (Host Channel e Condition Code).....	478
7. 6. 137.	17Fh H_CHeJoin (Host Channel b Join)	480
7. 6. 138.	180h H_CHOStartAdrs_H (Host Channel 0 Start Address High).....	481
7. 6. 139.	181h H_CHOStartAdrs_L (Host Channel 0 Start Address Low).....	481
7. 6. 140.	182h H_CHOEndAdrs_H (Host Channel 0 End Address High).....	482
7. 6. 141.	183h H_CHOEndAdrs_L (Host Channel 0 End Address Low).....	482
7. 6. 142.	184h H_CHaStartAdrs_H (Host Channel a Start Address High).....	483
7. 6. 143.	185h H_CHaStartAdrs_L (Host Channel a Start Address Low).....	483
7. 6. 144.	186h H_CHaEndAdrs_H (Host Channel a End Address High).....	484
7. 6. 145.	187h H_CHaEndAdrs_L (Host Channel a End Address Low).....	484
7. 6. 146.	188h H_CHbStartAdrs_H (Host Channel b Start Address High).....	485
7. 6. 147.	189h H_CHbStartAdrs_L (Host Channel b Start Address Low).....	485
7. 6. 148.	18Ah H_CHbEndAdrs_H (Host Channel b End Address High).....	486

7. 6. 149.	18Bh H_CHbEndAdrs_L (Host Channel b End Address Low).....	486
7. 6. 150.	18Ch H_CHcStartAdrs_H (Host Channel c Start Address High).....	487
7. 6. 151.	18Dh H_CHcStartAdrs_L (Host Channel c Start Address Low).....	487
7. 6. 152.	18Eh H_CHcEndAdrs_H (Host Channel c End Address High).....	488
7. 6. 153.	18Fh H_CHcEndAdrs_L (Host Channel c End Address Low).....	488
7. 6. 154.	190h H_CHdStartAdrs_H (Host Channel d Start Address High).....	489
7. 6. 155.	191h H_CHdStartAdrs_L (Host Channel d Start Address Low).....	489
7. 6. 156.	192h H_CHdEndAdrs_H (Host Channel d End Address High).....	490
7. 6. 157.	193h H_CHdEndAdrs_L (Host Channel d End Address Low).....	490
7. 6. 158.	194h H_CHeStartAdrs_H (Host Channel e Start Address High).....	491
7. 6. 159.	195h H_CHeStartAdrs_L (Host Channel e Start Address Low).....	491
7. 6. 160.	196h H_CHeEndAdrs_H (Host Channel e End Address High).....	492
7. 6. 161.	197h H_CHeEndAdrs_L (Host Channel e End Address Low).....	492
7. 6. 162.	198h~1FFh Reserved 0.....	493
8.	電気的特性.....	494
8. 1.	絶対最大定格.....	494
8. 2.	推奨動作条件.....	494
8. 3.	DC特性.....	495
8. 4.	AC特性.....	500
8. 4. 1.	RESETタイミング.....	500
8. 4. 2.	クロックタイミング.....	500
8. 4. 3.	CPU/DMA I/Fアクセスタイミング.....	501
8. 4. 3. 1.	CVDD (typ) = 1.65V~3.6Vの場合.....	501
8. 4. 3. 2.	CVDD (typ) = 3.0~3.6Vに制限した場合(一部規格を緩和).....	502
8. 4. 4.	IDE I/Fタイミング.....	503
8. 4. 4. 1.	PIO Read Timing.....	503
8. 4. 4. 2.	PIO Write Timing.....	504
8. 4. 4. 3.	DMA Read Timing.....	505
8. 4. 4. 4.	DMA Write Timing.....	506
8. 4. 4. 5.	Ultra DMA Read Timing.....	507
8. 4. 4. 6.	Ultra DMA Write Timing.....	509
8. 4. 5.	USB I/Fタイミング.....	510
9.	接続例.....	511
9. 1.	CPU I/F接続例.....	511
9. 2.	USB I/F接続例.....	512
9. 2. 1.	QFP15~128 の場合 (デバイス部周辺).....	512
9. 2. 2.	QFP15~128 の場合 (ホスト部周辺).....	513
9. 2. 3.	PFBGA8UX121/PFBGA10UX121 の場合 (デバイス部周辺).....	514
9. 2. 4.	PFBGA8UX121/PFBGA10UX121 の場合 (ホスト部周辺).....	515
10.	外形寸法図.....	516
10. 1.	QFPパッケージ.....	516
10. 2.	BGAパッケージ(PFBGA8UX121).....	517
10. 3.	BGAパッケージ(PFBGA10UX121).....	518
Appendix A	IDE_Config_1.Swapビット設定.....	519
Appendix B	リトルエンディアンのCPUへの接続.....	520
Appendix C	1ポートモード.....	530
Appendix D	HOST High-Speed動作時のSUSPENDIについて.....	531
Appendix E	SetAddressリクエストの応答について.....	535
改訂履歴 (Rev. 0.79~Rev. 0.90).....		537
改訂履歴 (Rev. 0.91~).....		542

1. 概要

S1R72V05**はUSB2.0準拠のハイスピードモードに対応するUSBホスト / デバイスコントローラLSIです。ホストポートとデバイスポートを独立に備え、制御を切り替えることによりUSBホストまたはUSBデバイスとして動作することが出来ます。

また、IDE I/Fを備え、HDDを内蔵するポータブル機器に適した特長を持ちます。

2. 特長

《 USB2.0 デバイス機能 》

- HS(480Mbps)及びFS(12Mbps)転送サポート
- FS/HSターミネーション内蔵(外付け回路不要)
- VBUS 5V I/F(外付け保護回路要)
- コントロール、バルク及びインタラプト転送をサポート
- Bulk転送用2本、Interrupt転送用1本、及び、Endpoint 0をサポート

《 USB2.0 ホスト機能 》

- HS(480Mbps), FS(12Mbps) 及びLS(1.5Mbps)転送サポート
- ダウンストリームポート用プルダウン抵抗内蔵(外付け回路不要)
- HSターミネーション内蔵(外付け回路不要)
- コントロール、バルク及びインタラプト転送をサポート

Channel方式

Control転送専用Channel 1本

Bulk転送専用Channel 1本

BulkおよびInterrupt転送用Channel 4本

- USBパワースイッチインタフェース

《 Mediaデータ転送機能 》

- USB機能を停止した状態でのIDE⇄CPUデータ転送が可能
- 専用FIFOを内蔵(USB用FIFOとは独立)

《 CPU I/F 》

- 16bit又は8bit幅の、汎用CPU I/Fに対応
- DMA 2ch. 搭載 (Multi-word手順)
- Big Endian(Little EndianのCPUに対応するためのバススワップ機能を搭載)
- I/F電圧変更可能(3.3V or 1.8V)

《 IDE I/F 》

- ATA/ATAPI6に対応
- PIOモード0~4, Multi word DMA, UDMAモード0~5

《 その他 》

- クロック入力 12MHz, 24MHz、水晶発振子対応。(発振回路及び帰還抵抗1MΩ内蔵)
- 電源電圧 3.3V と 1.8V、及び、CPU I/F電源の3電圧系統
- パッケージタイプ QFP15-128, PFBGA8UX121, PFBGA10UX121
- 動作保証温度範囲 -40°C~85°C

※耐放射設計はされていません

3. ブロック図

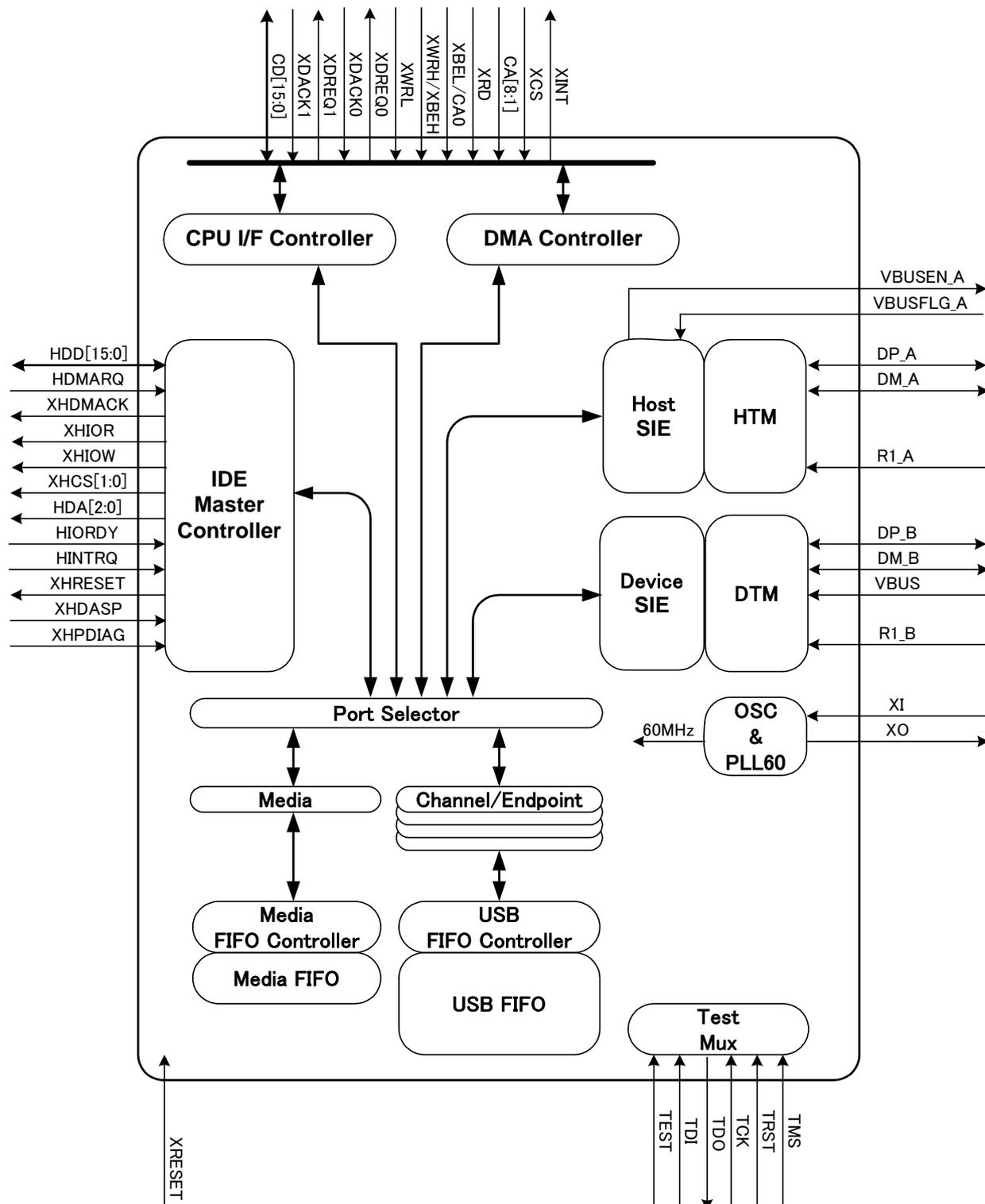


図 3-1 全体ブロック図

3.1. Device Transceiver Macro (DTM)

UTMI1.03仕様準拠のUSB2.0トランシーバマクロです。アナログ回路及び高速ロジック回路を持ち、HSモード(480Mbps)、FSモード(12Mbps)をサポートします。

HS/FSに対応するトランスミッタ、レシーバ、ターミネーション、スケルチ回路や、エラステシティバッファ、シリパラ/パラシリ変換回路、ビットスタッフ/アンスタッフ回路、SYNC/EOPの付加/除去回路等を内蔵し、USBインタフェースを実現します。

また、内蔵するPLLでHS転送に必要な480MHzクロックを生成します。PLLは、Oscillatorを発振源とします。

3.2. Host Transceiver Macro (HTM)

Host用USB2.0トランシーバマクロです。アナログ回路及び高速ロジック回路を持ち、HSモード(480Mbps)、FSモード(12Mbps)、LSモード(1.5Mbps)をサポートします。

HS/FS/LSに対応するトランスミッタ、レシーバ、ターミネーション、スケルチ回路、切断エンベロープ検出回路や、エラステシティバッファ、シリパラ/パラシリ変換回路、ビットスタッフ/アンスタッフ回路、SYNC/EOPの付加/除去回路等を内蔵し、USBインタフェースを実現します。

また、内蔵するPLLでHS転送に必要な480MHzクロックを生成します。

3.3. Oscillator & PLL60

発振回路を内蔵し、内部ロジック動作に必要な60MHzクロックを生成します。発振回路の入力クロックは12MHz/24MHzの水晶発振子に対応可能です。

3.4. Device Serial Interface Engine (Device SIE)

トランザクションの管理と、パケット生成を行います。

また、サスペンド・リジューム・リセット等の、バスイベントを制御します。

3.5. Host Serial Interface Engine (Host SIE)

トランザクションのスケジューリング、トランザクションの管理およびパケット生成を行います。

また、サスペンド・リジューム・リセット等の、バスイベントを生成します。

さらに、接続切断状態の検出、およびVBUSの制御(外付けUSBパワースイッチとの連携による)を行います。

3.6. Port Selector

CPU(DMA)、IDE、DeviceSIE、HostSIEの各ポートと、USB FIFO、Media FIFOの各FIFOとの接続切替を行います。

3.7. USB FIFO / USB FIFO Controller

チャンネル / エンドポイント用バッファです。

3.8. Media FIFO / Media FIFO Controller

Media用バッファです。

3.9. CPU I/F Controller

CPU I/Fのタイミングを制御し、レジスタアクセスを可能にします。

3.10. DMA Controller

CPU I/FのDMAタイミングを制御し、FIFOアクセスを可能にします。2チャンネルのDMAを搭載します。

3.11. IDE Master Controller

UATA100に対応するIDEインタフェースです。

3.12. TestMUX

テスト回路です。

4. 端子配置図

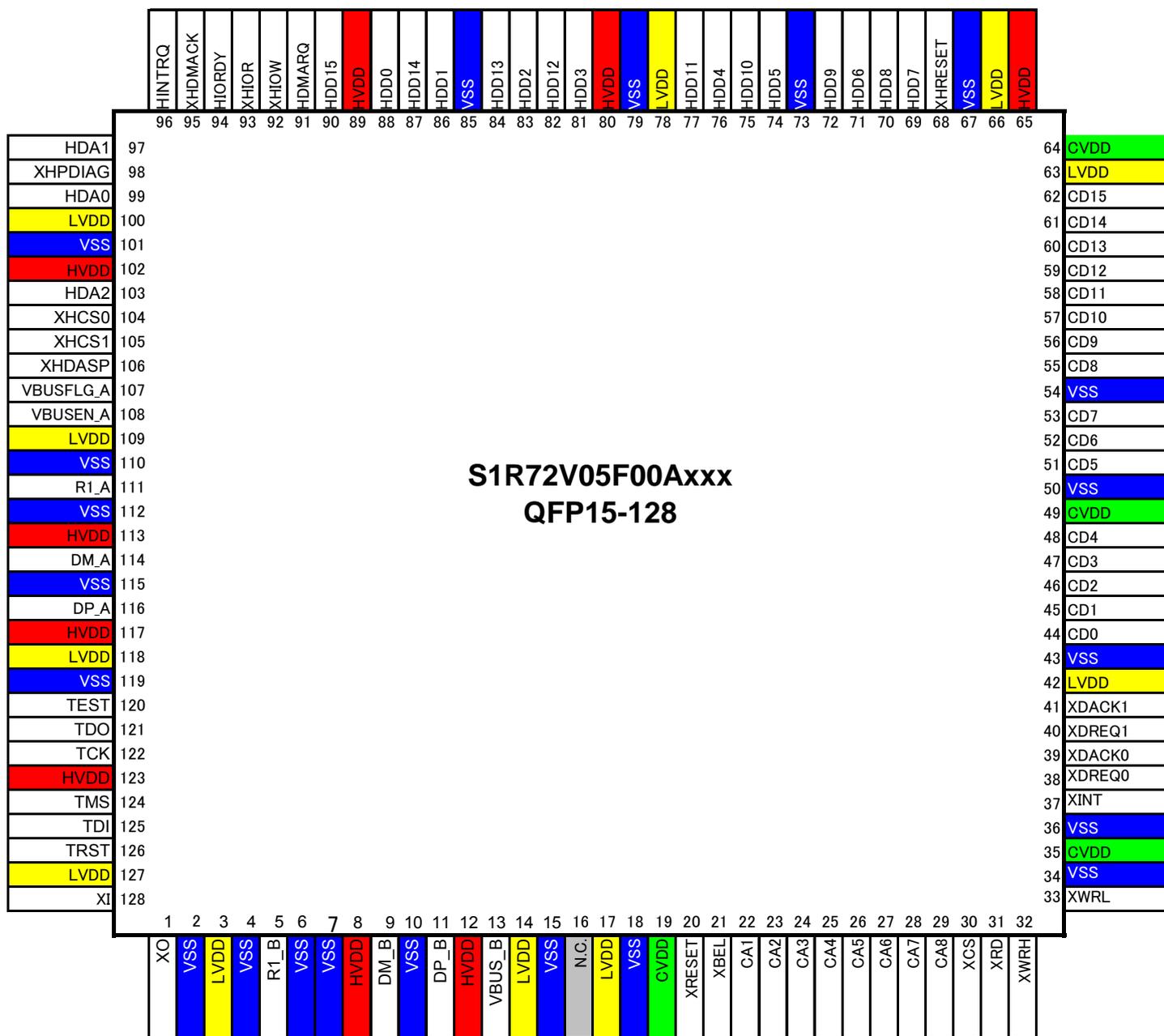


図 4-1 QFP パッケージ端子配置図

4. 端子配置図

S1R72V05B00A2xx/PFBGA8UX121
S1R72V05B00A3xx/PFBGA10UX121
TOP View

1	2	3	4	5	6	7	8	9	10	11	
NC	XI	LVDD	LVDD	DP_A	DM_A	HVDD	R1_A	LVDD	HDA0	NC	A
XO	VSS	TRST	VSS	HVDD	VSS	VBUSEN_A	VSS	VSS	HDA2	XHPDIAG	B
LVDD	VSS	TDI	TCK	TEST	XHCS0	VBUSFLG_A	VSS	XHCS1	HDA1	HINTRQ	C
R1_B	VSS	TDO	XHDASP	HVDD	XHDMACK	HIORDY	XHIOW	XHIOR	HDD0	HDMARQ	D
HVDD	TMS	VSS	LVDD	VSS	HDD14	HDD15	HDD12	VSS	HDD2	HDD13	E
DM_B	VSS	VSS	CA2	VSS	LVDD	HDD3	VSS	HDD1	VSS	HVDD	F
DP_B	HVDD	VBUS_B	CA3	XINT	XDACK1	HVDD	HDD11	HDD5	HDD10	HDD4	G
LVDD	VSS	CVDD	CA4	XDACK0	CD3	CD6	CVDD	CD13	HDD8	HDD9	H
LVDD	XRESET	CA1	XBEL	XDREQ1	CD0	CD4	CD7	CD10	HDD6	HDD7	J
CA8	XCS	CA5	CA6	CA7	CD1	CD5	CD9	CD12	CD14	XHRESET	K
NC	XRD	XWRH	XWRL	XDREQ0	CD2	CVDD	CD8	CD11	CD15	NC	L
1	2	3	4	5	6	7	8	9	10	11	

図 4-2 BGA パッケージ端子配置図

5. 端子機能説明

OSC						
Pin	Ball	名称	I/O	RESET	端子タイプ	端子説明
128	A2	XI	IN	-	Analog	内部発振回路用入力 12MHz/24MHz
1	B1	XO	OUT	-	Analog	内部発振回路用出力

TEST						
Pin	Ball	名称	I/O	RESET	端子タイプ	端子説明
120	C5	TEST	IN	-		テスト端子(Low 固定)
121	D3	TDO	OUT	Hi-Z	2mA	JTAG TDO 端子
122	C4	TCK	IN	-		JTAG TCK 端子
124	E2	TMS	IN	-		JTAG TMS 端子
125	C3	TDI	IN	-		JTAG TDI 端子
126	B3	TRST	IN	-		JTAG TRST 端子

JTAG 機能を使用しない場合は、TEST, TCK, TMS, TDI, TRST の各端子は Low 固定、TDO 端子はオープンとして処理してください。

PD: Pull Down

PU: Pull Up

USB						
Pin	Ball	名称	I/O	RESET	端子タイプ	端子説明
111	A8	R1_A	IN	-	Analog	内部動作設定端子 6.2kΩ ± 1%の抵抗を VSS 間に接続
116	A5	DP_A	BI	Hi-Z	Analog	USB ホスト・データライン Data+
114	A6	DM_A	BI	Hi-Z	Analog	USB ホスト・データライン Data-
107	C7	VBUSFLG_A	IN	-	PU Schmitt	USB パワースイッチ・フォールト検出信号 1: 正常 0: 異常
108	B7	VBUSEN_A	OUT	Lo	2mA	USB パワースイッチ制御信号
5	D1	R1_B	IN	-	Analog	内部動作設定端子 6.2kΩ ± 1%の抵抗を VSS 間に接続
11	G1	DP_B	BI	Hi-Z	Analog	USB デバイス・データライン Data+
9	F1	DM_B	BI	Hi-Z	Analog	USB デバイス・データライン Data-
13	G3	VBUS_B	IN	(PD)	(PD)	USB デバイス・バス検出信号

PD: Pull Down

PU: Pull Up

5. 端子機能説明

CPU I/F								
Pin	Ball	名称	I/O	RESET	端子タイプ	端子説明		
Bus Mode ⇒						16bit Strobe mode	16bit BE mode	8bit mode
20	J2	XRESET	IN	-	-	リセット信号		
31	L2	XRD	IN	-	-	リード・ストローブ		
33	L4	XWRL (XWR)	IN	-	-	ライト・ストローブ 下位	ライト・ストローブ	
32	L3	XWRH (XBEH)	IN	-	-	ライト・ストローブ 上位	ハイ・バイト イネーブル	High 固定
30	K2	XCS	IN	-	-	チップ選択信号		
37	G5	XINT	OUT	High	2mA (Tri-state)	割り込み出力信号		
38	L5	XDREQ0	OUT	High	2mA	DMA0 リクエスト		
39	H5	XDACK0	IN	-	-	DMA0 アクノリッジ		
40	J5	XDREQ1	OUT	High	2mA	DMA1 リクエスト		
41	G6	XDACK1	IN	-	-	DMA1 アクノリッジ		
21	J4	XBEL (CA0)	IN	-	-	High or Low 固定	ロー・バイト イネーブル	アドレス 0
22	J3	CA1	IN	-	-	CPU バスアドレス		
23	F4	CA2	IN	-	-			
24	G4	CA3	IN	-	-			
25	H4	CA4	IN	-	-			
26	K3	CA5	IN	-	-			
27	K4	CA6	IN	-	-			
28	K5	CA7	IN	-	-			
29	K1	CA8	IN	-	-			
44	J6	CD0	BI	Hi-Z	2mA	CPU データバス		
45	K6	CD1	BI	Hi-Z	2mA			
46	L6	CD2	BI	Hi-Z	2mA			
47	H6	CD3	BI	Hi-Z	2mA			
48	J7	CD4	BI	Hi-Z	2mA			
51	K7	CD5	BI	Hi-Z	2mA			
52	H7	CD6	BI	Hi-Z	2mA			
53	J8	CD7	BI	Hi-Z	2mA			
55	L8	CD8	BI	Hi-Z	2mA			
56	K8	CD9	BI	Hi-Z	2mA			
57	J9	CD10	BI	Hi-Z	2mA			
58	L9	CD11	BI	Hi-Z	2mA			
59	K9	CD12	BI	Hi-Z	2mA			
60	H9	CD13	BI	Hi-Z	2mA			
61	K10	CD14	BI	Hi-Z	2mA			
62	L10	CD15	BI	Hi-Z	2mA			

XINT 端子は、レジスタ設定により、1/0 モードと Hi-Z/0 モードを選択できます。

PD: Pull Down

PU: Pull Up

IDE I/F

5. 端子機能説明

Pin	Ball	名称	I/O	RESET	端子タイプ	端子説明
103	B10	HDA2	OUT	Hi-Z	4mA	IDE レジスタ・アドレス
97	C10	HDA1	OUT	Hi-Z	4mA	
99	A10	HDA0	OUT	Hi-Z	4mA	
105	C9	XHCS1	OUT	Hi-Z	4mA	コントロールレジスタアクセス用チップセレクト
104	C6	XHCS0	OUT	Hi-Z	4mA	コマンドブロックレジスタアクセス用チップセレクト
93	D9	XHIOR	OUT	Hi-Z	4mA	IDE リードストローブ
92	D8	XHIOW	OUT	Hi-Z	4mA	IDE ライトストローブ
91	D11	HDMAHQ	IN	(PD)	(PD)	DMA 転送要求
95	D6	XHDMACK	OUT	Hi-Z	4mA	DMA 転送許可
94	D7	HIORDY	IN	(PU)	(PU)	IDE レジスタレディ信号
96	C11	HINTRQ	IN	(PD)	(PD)	IDE 割り込み要求
68	K11	XHRESET	OUT	Hi-Z	4mA	IDE バスリセット
106	D4	XHDASP	IN	(PU)	(PU)	ドライブ有効/スレーブドライブ有り
98	B11	XHPDIAG	IN	(PU)	(PU)	診断シーケンス終了信号
90	E7	HDD15	BI	Hi-Z	4mA(PU)	IDE データ・バス
87	E6	HDD14	BI	Hi-Z	4mA(PU)	
84	E11	HDD13	BI	Hi-Z	4mA(PU)	
82	E8	HDD12	BI	Hi-Z	4mA(PU)	
77	G8	HDD11	BI	Hi-Z	4mA(PU)	
75	G10	HDD10	BI	Hi-Z	4mA(PU)	
72	H11	HDD9	BI	Hi-Z	4mA(PU)	
70	H10	HDD8	BI	Hi-Z	4mA(PU)	
69	J11	HDD7	BI	(PD)	4mA(PD)	
71	J10	HDD6	BI	Hi-Z	4mA(PU)	
74	G9	HDD5	BI	Hi-Z	4mA(PU)	
76	G11	HDD4	BI	Hi-Z	4mA(PU)	
81	F7	HDD3	BI	Hi-Z	4mA(PU)	
83	E10	HDD2	BI	Hi-Z	4mA(PU)	
86	F9	HDD1	BI	Hi-Z	4mA(PU)	
88	D10	HDD0	BI	Hi-Z	4mA(PU)	

PU および PD はレジスタ設定により ON/OFF できます。

PD: Pull Down

PU: Pull Up

(注)IDE I/F 端子は全て 5V トレラントになっております。

POWER				
Pin	Ball	名称	電圧	端子説明
8, 12, 65, 80, 89, 102, 113, 117, 123	G7, D5, F11, E1, G2, B5, A7	HVDD	3.3V	IDE I/F I/O, USB I/O, TEST I/O 用電源
19, 35, 49, 64	H3, L7, H8	CVDD	1.8~3.3 v	CPU I/F I/O 用電源
14, 17, 42, 63, 66, 78, 100, 118, 127, 3, 109	J1, E4, F6, H1, A3, A4, C1, A9	LVDD	1.8V	内部電源, OSC I/O 用電源
2, 6, 7, 10, 15, 18, 34, 36, 43, 50, 54, 67, 73, 79, 85, 101, 112, 115, 119, 4, 110	F3, E3, E5, F5, C8, F8, E9, F10, H2, F2, B2, B4, B6, B8, D2, C2, B9	VSS	0V	GND
16	A1, L1, A11, L11	N.C.	0V	N.C.端子(GND に接続してください)

6. 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表記します。

- ・1 アドレス分のレジスタを指す場合の名称
レジスタ名+レジスタと表記します。
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+ビットと表記します。
例:「MainIntStat.CPU_IntStat ビット」
- ・エンドポイント毎にあるレジスタ
D_EPx{x=0,a-c}~レジスタ等と表記します。
例:「D_EPx{x=0,a-c}IntStat レジスタ」
- ・チャンネル毎にあるレジスタ
H_CHx{x=0,a-e}~レジスタ等と表記します。
例:「H_CHx{x=0,a-c}IntStat レジスタ」
- ・DMA チャンネル毎にあるレジスタ
DMAx{x=0,1}~レジスタ等と表記します。
例:「DMAx{x=0,1}Config レジスタ」

6.1. デバイス／ホスト・レジスタマップ選択

本 LSI では、USB を使用する際に、デバイス・レジスタとホスト・レジスタについてはいずれか一方のレジスタマップを選択する必要があります。

デバイス・レジスタマップを選択した場合（以下デバイス・モード）、デバイス／ホスト・共通レジスタおよびデバイス・レジスタのレジスタ・ビットが有効となります。

ホスト・レジスタマップを選択した場合（以下ホスト・モード）、デバイス／ホスト・共通レジスタおよびホスト・レジスタのレジスタ・ビットが有効となります。

6.1.1. レジスタマップ選択方法

デバイス・レジスタとホスト・レジスタとのレジスタマップ選択は、レジスタ・ビットの設定にて行います。

レジスタマップの選択は、SLEEP / SNOOZE / ACTIVE60 / ACT_DEVICE / ACT_HOST（パワー・マネージメント項を参照）の全ステートで設定をすることができます。デバイス／ホスト・レジスタマップ選択の設定項目を表 6-1 に示します。

表 6-1 デバイス・モード／ホスト・モード選択の設定項目

項目	レジスタ／ビット	説明
デバイス／ホスト レジスタマップ選択	HostDeviceSel.HOSTxDEVICE	デバイスとホストのいずれかのレジスタマップを選択し、 選択した方のレジスタが使用できます。

6.1.2. ポート状態変化検出ステータス

本 LSI には、ポート状態の検出機能があります。

これらは SLEEP / SNOOZE / ACTIVE60 / ACT_DEVICE / ACT_HOST（パワー・マネージメント項を参照）の全ステートで使用できます。また、デバイス／ホストのモードにかかわらず使用できますので、デバイス・モード時に Downstream port（ホストポート）の状態変化を、ホスト・モード時に Upstream port（デバイスポート）の状態変化を検出することができます。

ホストポートおよびデバイスポートの状態変化は、モード選択のきっかけとして使用することができます。

6.1.2.1. ポート状態変化検出ステータスの使用例

VBUS_B 端子変化ステータスおよび信号ライン変化ステータスの使用例を示します。

6.1.2.1.1. VBUS_B 端子変化ステータス

本ステータスは、デバイスポートの **VBUS_B** 端子の状態が変化したことを示します。

VBUS_B 端子変化ステータス関連レジスタを表 6-2 に示します。

表 6-2 **VBUS_B** 端子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_B 端子変化 ステータス	DeviceIntStat.VBUS_Changed	デバイスポートの VBUS_B 端子の状態が変化したことを 示します。
VBUS_B 端子変化 ステータスイネーブル	DeviceIntEnb.EnVBUS_Changed	DeviceIntStat.VBUS_Changed による MainIntStat.DeviceIntStat ビットのアサートを許可／禁止 します。
デバイスポート VBUS_B 端子状態	D_USB_Status.VBUS	デバイスポートの VBUS_B 端子の状態を示します。

VBUS_B 端子変化ステータスを使用する場合、ファームウェアは(1)、(2)、(4)～(7)の処理を行います。

- (1) **VBUS_B** 端子変化ステータスをクリアします。
- (2) **VBUS_B** 端子変化ステータスイネーブルをセットします。
- (3) デバイスポートに VBUS の供給があると、**VBUS_B** 端子変化ステータスが発行されます。
- (4) **VBUS_B** 端子変化ステータスを確認します。

- (5) **VBUS_B** 端子変化ステータスをクリアします。
- (6) **VBUS_B** 端子変化ステータスイネーブルをクリアします。
- (7) デバイスポート **VBUS_B** 端子状態を確認します。デバイスポート **VBUS_B** 端子状態ビットが“1”であれば、VBUSが供給されている(デバイスポートにホストまたはハブが接続されている)状態と判断します。

6.1.2.1.2. 信号ライン変化ステータス

本ステータスは、ホストポートの DP 端子および DM 端子の状態が SE0 から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 6-3 に示します。

表 6-3 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	HostIntStat.LineStateChanged	ホストポートのDP端子およびDM端子の状態がSE0から変化したことを示します。
信号ライン変化ステータスイネーブル	HostIntEnb.EnLineStateChanged	HostIntStat.LineStateChanged による MainIntStat.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	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	HostIntEnb.EnVBUS_Err	HostIntStat.VBUS_Err による MainIntStat.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]	HTM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。
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_1.AutoMode[3:0]	遷移させるホストステートを設定します。

本ステータスは、USB ホストとして動作中の本来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 / SNOOZE / ACTIVE60 / ACT_DEVICE (パワーマネジメント項を参照) のいずれかのステートにおいて、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 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、本ステータスの使用はここでの信号ライン変化の検出に留め、USB ホスト動作時は HostIntEnb.EnLineStateChanged をディセーブルにしてください。

また、USB ホスト動作時以外で 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。
- (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 イネーブルビットは、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、ホストステート管理サポート機能の項を参照してください。

6.2. USB デバイス制御

ここでは、USB デバイス機能について説明します。

6.2.1. 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、3本の汎用エンドポイント(EPa, EPb, EPc)を持ちます。エンドポイント EPa, EPb, EPc は、それぞれに、バルクまたはインタラプト転送用のエンドポイントとして使用できます。

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 6-4 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。

表 6-4 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 には、FIFO の 0 番地から 64 バイトの領域が割り当てられます。

表 6-5 に汎用エンドポイント (EPa, EPb, EPc) の基本設定項目を示します。エンドポイント EPa, EPb, EPc 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、3本までの独立したエンドポイントを使用できます。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。

エンドポイント EPa, EPb, EPc の FIFO 領域はスタートアドレスおよびエンドアドレスにより設定します。

表 6-5 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-c}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-c}MaxSize_H, D_EPx{x=a-c}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。
エンドポイントナンバー	D_EPx{x=a-c}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-c}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、"0" にセットして下さい。
エンドポイント有効	D_EPx{x=a-c}Config_0.EnEndpoint	各エンドポイントを有効にします。 そのエンドポイントを使用する USB 定義インタフェイスが有効にされた時に設定して下さい。
FIFO 領域	D_EPx{x=a-c}StartAdrs_H, D_EPx{x=a-c}StartAdrs_L, D_EPcEndAdrs_H, D_EPcEndAdrs_L	各エンドポイント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、機能説明の FIFO の項を参照して下さい。

6.2.2. トランザクション

LSI は HW 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、レジスタ説明の章を参照して下さい。

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)、または、IDE インタフェイス (IDE ライト) 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) または、IDE インタフェイス (IDE リード) 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 6-6 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 6-6 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。
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	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_FIFO_IntStat.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 6-7 に、汎用エンドポイント EPa, EPb, EPc 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 6-7 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-c}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-c}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-c}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-c}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-c}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-c}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-c}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-c}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-c}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-c}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-c}IntStat.OUT_ShortACK, D_EPx{x=a-c}IntStat.IN_TranACK, D_EPx{x=a-c}IntStat.OUT_TranACK, D_EPx{x=a-c}IntStat.IN_TranNAK, D_EPx{x=a-c}IntStat.OUT_TranNAK, D_EPx{x=a-c}IntStat.IN_TranErr, D_EPx{x=a-c}IntStat.OUT_TranErr	トランザクションの結果を示します。

6.2.2.1. SETUP トランザクション

自ノードのエンドポイント EP0 宛ての SETUP トランザクションは、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。

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。

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 6-1 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。

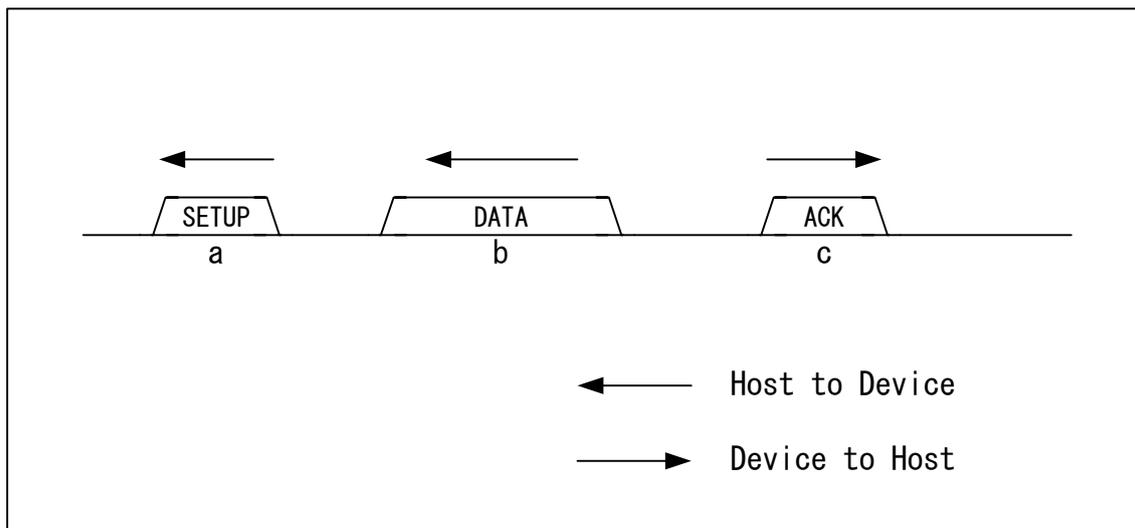


図 6-1 SETUP トランザクション

6.2.2.2. 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。

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_TransACK ステータス (D_EPx{x=0,a-c}IntStat.OUT_TransACK 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-c}IntStat.OUT_ShortACK ビット) を発行します。また、D_EPx{x=a-c}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-c}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_TransErr ステータス (D_EPx{x=0,a-c}IntStat.OUT_TransErr 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_TransNAK ステータス (D_EPx{x=0,a-c}IntStat.OUT_TransNAK ビット) を発行します。FIFO は更新されません。

図 6-2 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。LSII 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) LSII 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。

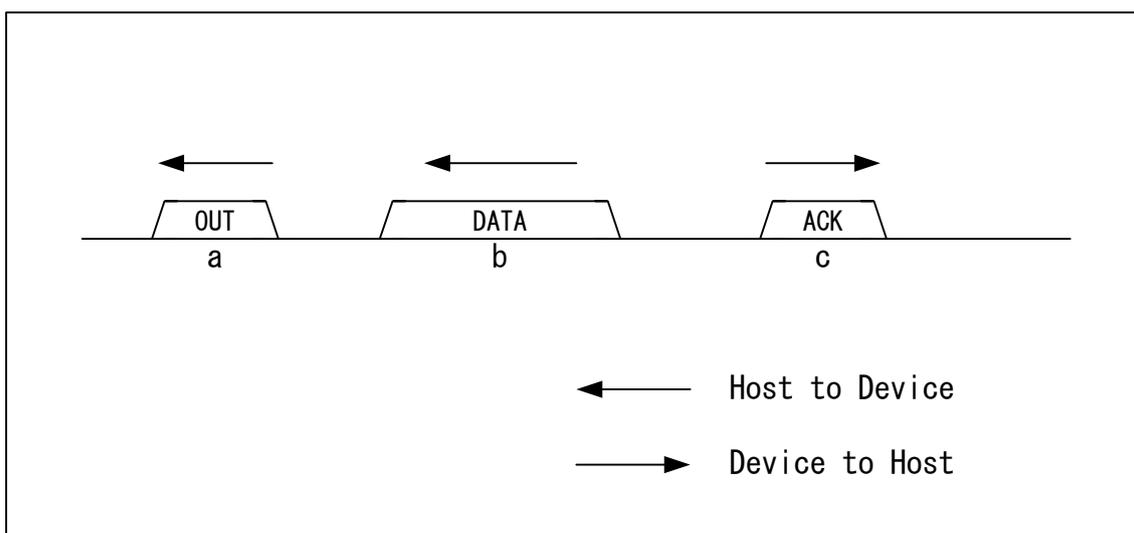


図 6-2 OUT トランザクション

6.2.2.3. 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あるか、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ตอบสนอง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_EP0Control.IN.EnShortPkt ビットまたは D_EPx{x=a-c}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_EP0Control.IN.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_TranACK ステータス (D_EPx{x=0,a-c}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。

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_TranErr ステータス (D_EPx{x=0,a-c}IntStat.IN_TranErr 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_TranNAK ステータス (D_EPx{x=0,a-c}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。

図 6-3 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) LSII は、この IN トランザクションにตอบสนอง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。LSII は ACK を受信すると、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。

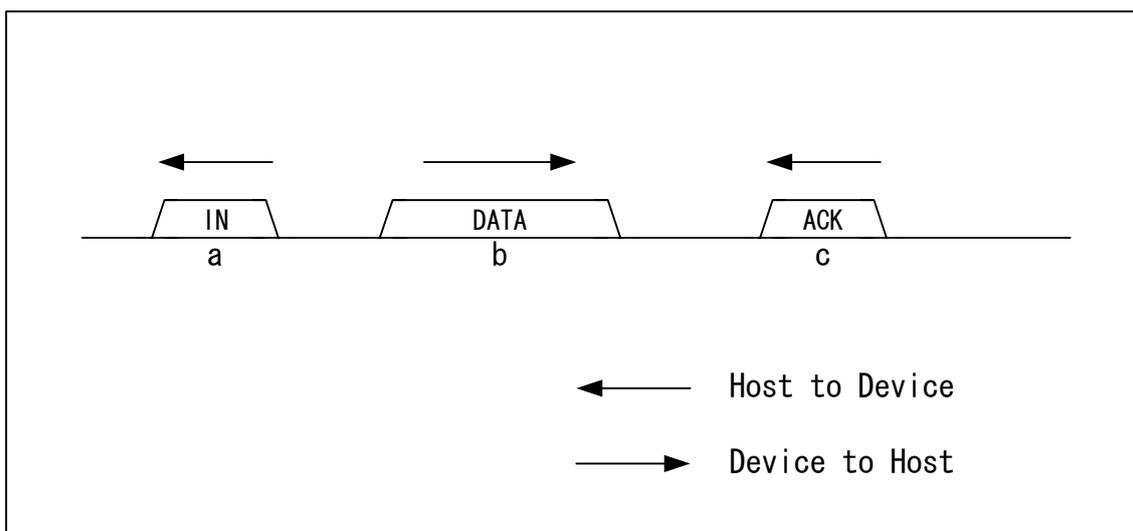


図 6-3 IN トランザクション

6.2.2.4. 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_TranNAK ステータス (D_EPx{x=0,a-c}IntStat.OUT_TranNAK 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 6-4 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) LSII 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て 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。

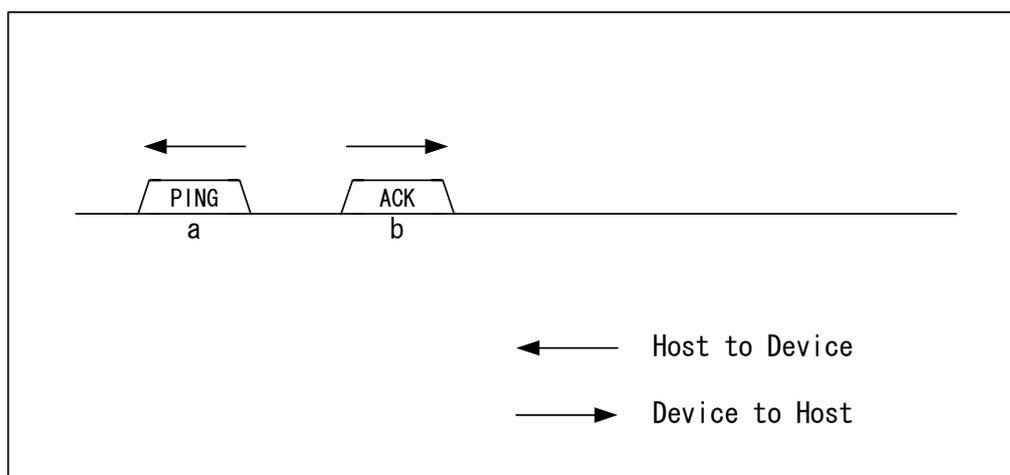


図 6-4 PING トランザクション

6.2.3. 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 6-5 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、SETUPT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INT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。

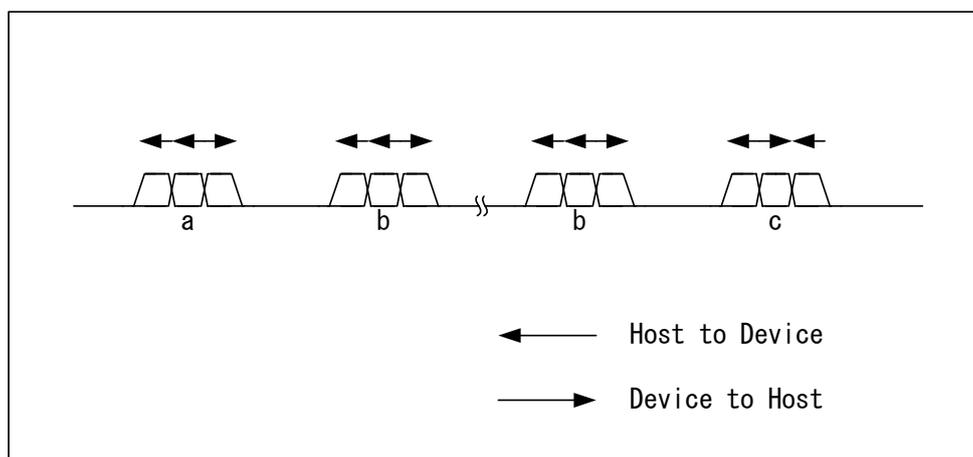


図 6-5 データステージが OUT 方向のコントロール転送

図 6-6 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、SETUPT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INT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。

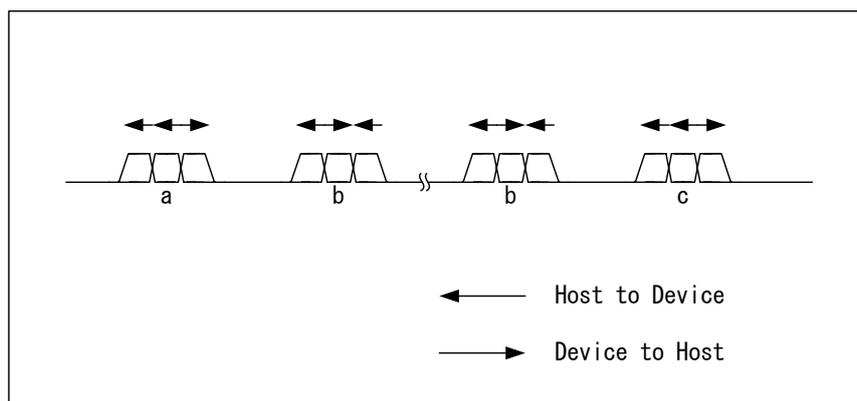


図 6-6 データステージが IN 方向のコントロール転送

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。

6.2.3.1. 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。

6.2.3.2. 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。

6.2.3.3. 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()リクエストであった場合には、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、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 USB_Address レジスタにより、アドレスを確認できます。

6.2.3.4. 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回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_EP0Control.In.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応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_FIFO_IntStat.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、機能説明の FIFO の項を参照して下さい。

6.2.4. バルク転送／インタラプト転送

汎用エンドポイント EPa, EPb, EPc におけるバルク転送、インタラプト転送は、データフロー(6.2.5 参照)としても、連続する個々のトランザクション(6.2.2 参照)としても制御できます。

6.2.5. 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。

6.2.5.1. OUT 転送

OUT 転送によって受信したデータは、各エンドポイントの FIFO 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、及び IDE へのライト転送の方法があります。

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、D_EPx{x=0,a-c}Join.JoinCPU_Rd ビットにより、ただ一つのエンドポイントを選択して下さい。選択したエンドポイントの FIFO 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、D_EPx{x=0,a-c}Join.JoinDMAx{x=0,1} ビットにより、DMA のチャンネル毎にただ一つのエンドポイントを選択し、DMAx{x=0,1}_Control.Dir ビットに 1 を設定して下さい。選択したエンドポイントの FIFO 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、DMAx{x=0,1}_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。

IDE インタフェイスのライト転送により、FIFO のデータを読み出すには、D_EPx{x=a-c}Join.JoinIDE ビットにより、ただ一つのエンドポイントを選択し、IDE.Control.Dir ビットに 1 を設定して下さい。選択したエンドポイントの FIFO は、IDE_Control.IDE_Go による IDE 転送を実行することにより、受信順に読み出されます。FIFO が空になると、IDE インタフェイスは自動的にライト転送を一時停止して、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有れば、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-c}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-c}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-c}Control.ForceNAK ビットをクリアして下さい。

6.2.5.2. IN 転送

IN 転送により送信するデータを、各エンドポイントの FIFO 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、及び IDE からのリード転送の方法があります。

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、D_EPx{x=0,a-c}Join.JoinCPU_Wr ビットにより、ただ一つのエンドポイントを選択して下さい。選択したエンドポイントの FIFO には、FIFO_Wr 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、D_EPx{x=0,a-c}Join.JoinDMAx{x=0,1} ビットにより、DMA のチャンネル毎にただ一つのエンドポイントを選択し、DMAx{x=0,1}_Control.Dir ビットに 0 を設定して下さい。選択したエンドポイントの FIFO 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。

IDE インタフェイスのリード転送により、FIFO へデータを書き込むには、D_EPx{x=a-c}Join.JoinIDE ビットにより、ただ一つのエンドポイントを選択し、IDE_Control.Dir ビットに 0 を設定して下さい。選択したエンドポイントの FIFO は、IDE_Control.IDE_Go による IDE 転送

を実行することにより、IDE からリードした順に書き込まれ、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、IDE インタフェイスは自動的にリード転送を一時停止して、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有れば、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、DMAx{x=0,1}_FIFO_Control.AutoEnShort ビットがセットされていると、CPU インタフェイスの 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。

6.2.6. バルクオンリーサポート

本 LSI には、エンドポイント EPb, EPc におけるバルク転送において、USB Mass Storage Class(Bulk Only 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。

BulkOnlyConfig.EPx{x=b,c}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の CBW サポート、または CSW サポートが実行されている間、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。

6.2.6.1. CBW サポート

ファームウェアはBulk Only 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。BulkOnlyConfig.EPx{x=b,c}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制御して下さい。CBWサポートが有効であるときに、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、Bulk Only 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=b,c}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。

6.2.6.2. CSW サポート

ファームウェアはBulk Only 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=b,c}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トランザクションにおいて送信するデータをCSWとします。

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了した場合には、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。

6.2.7. オート・ネゴシエーション機能

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、リストア実行を USB バスの状態を逐次チェックしながら自動的にを行います。実際に何が行われたかは、各割り込み (DetectRESET、DetectSUSPEND、ChirpCmp、RestoreCmp) をチェックすることにより確認することが出来ます。

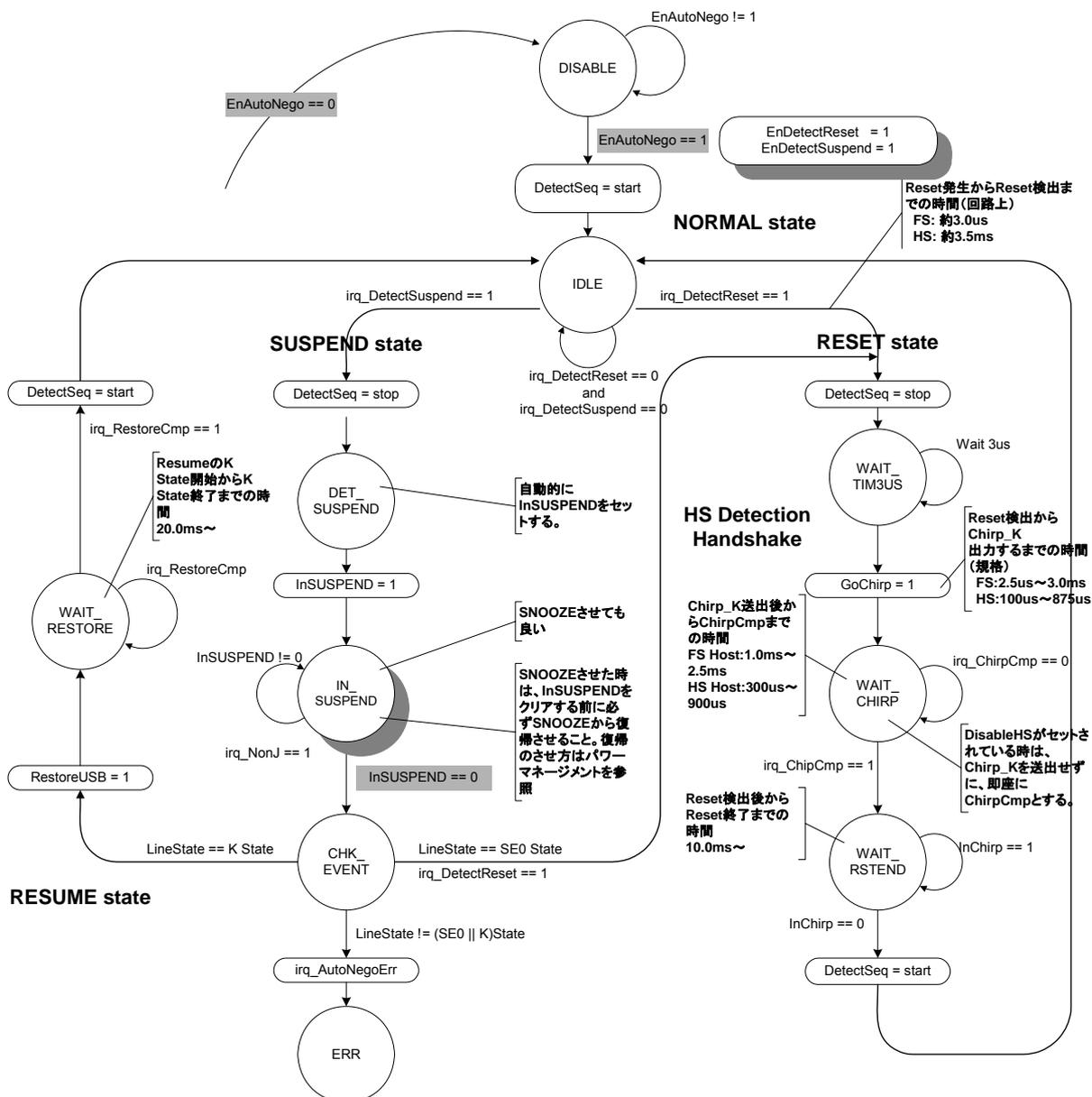


図 6-7 オート・ネゴシエータ

6.2.7.1. DISABLE

D_NegoControl.EnAutoNego ビットをクリアしている時に、この状態に入ります。

オート・ネゴシエーション機能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・ネゴシエーション機能を有効にすると、内部イベント検出機能を有効にします。オート・ネゴシエーション機能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。

6.2.7.2. 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。

6.2.7.3. 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。

6.2.7.4. 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、6.2.7.11.5 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。

6.2.7.5. 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。

6.2.7.6. 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、2段階の消費電流軽減策(スヌーズ、スリープ)を持っています。詳しい内容、さらには制御方法につきましては、パワーマネジメント機能(6.5)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、F/WにてD_SIE_IntEnb.EnNonJ ビットをセットし、NonJ 割り込みを許可してください。

6.2.7.7. IN_SUSPEND

NonJ 割り込みステータス(D_SIE_IntStat.NonJ)がセットされた場合サスペンドからの復帰指示であると判断し、D_NegoControl.InSUSPEND ビットを F/W にてクリアすると、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。

6.2.7.8. 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。

6.2.7.9. 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。

6.2.7.10. ERR

一旦このステートに突入すると、オート・ネゴシエーション機能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・ネゴシエーション機能を停止してください。

6.2.7.11. 各ネゴシエーション機能の単体説明

6.2.7.11.1. サスペンド検出 (HS モード)

本 LSI が HS モードで動作している時に、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。

この時、D_SIE_IntEnb.EnDetectSUSPEND ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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ので、USB のサスペンド状態であると判断します。以下の図では、スヌーズを行った時の動作を表しています。

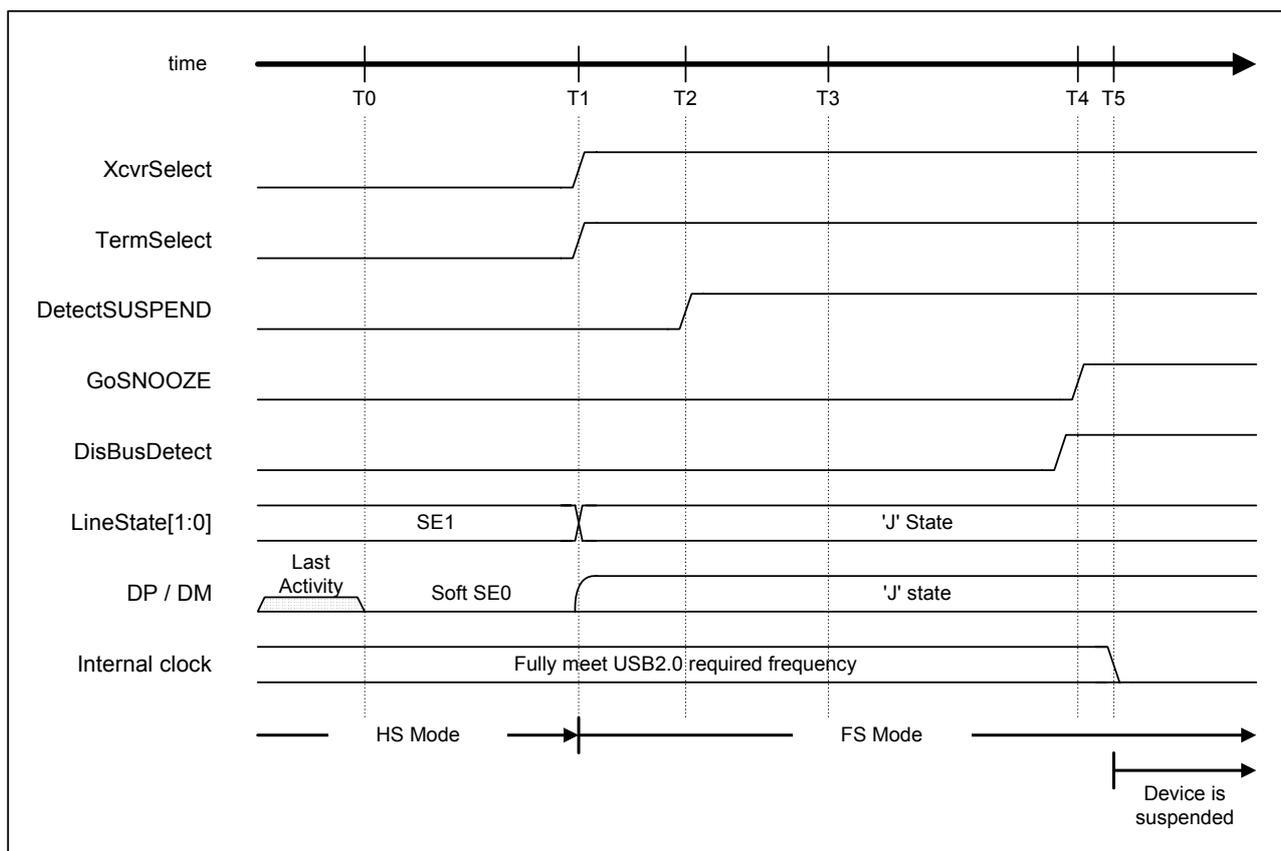


図 6-8 Suspend Timing (HS mode)

表 6-8 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、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 '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 + 100us < T2 {T _{WTWRSTHS} } < T1 + 875us
T3	これより前では、RESUME を発行してはいけけない。	HS Reset T0 + 5ms {T _{WTRSM} }
T4	SnnoozeUTM を '1' にセットし、完全にスヌーズに移行。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引っ張ってはいけけない。 (スヌーズ移行前に、DisBusDetect を '1' にセット)	HS Reset T0 + 10ms {T _{2SUSP} }
T5	内部クロックが完全停止。(スヌーズ電流 8mA (typ))	T5 < T4 + 10us

注: {} は、USB2.0 規格書で規格されている名称である。

6.2.7.11.2. サスペンド検出 (FS モード)

本 LSI が FS モードで動作している時に、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 のサスペンド状態であると判断し、SIE_IntStat.DetectSUSPEND ビットがセットされます。

この時、D_SIE_IntEnb.EnDetectSUSPEND ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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。以下の図では、スヌーズを行った時の動作を表しています。

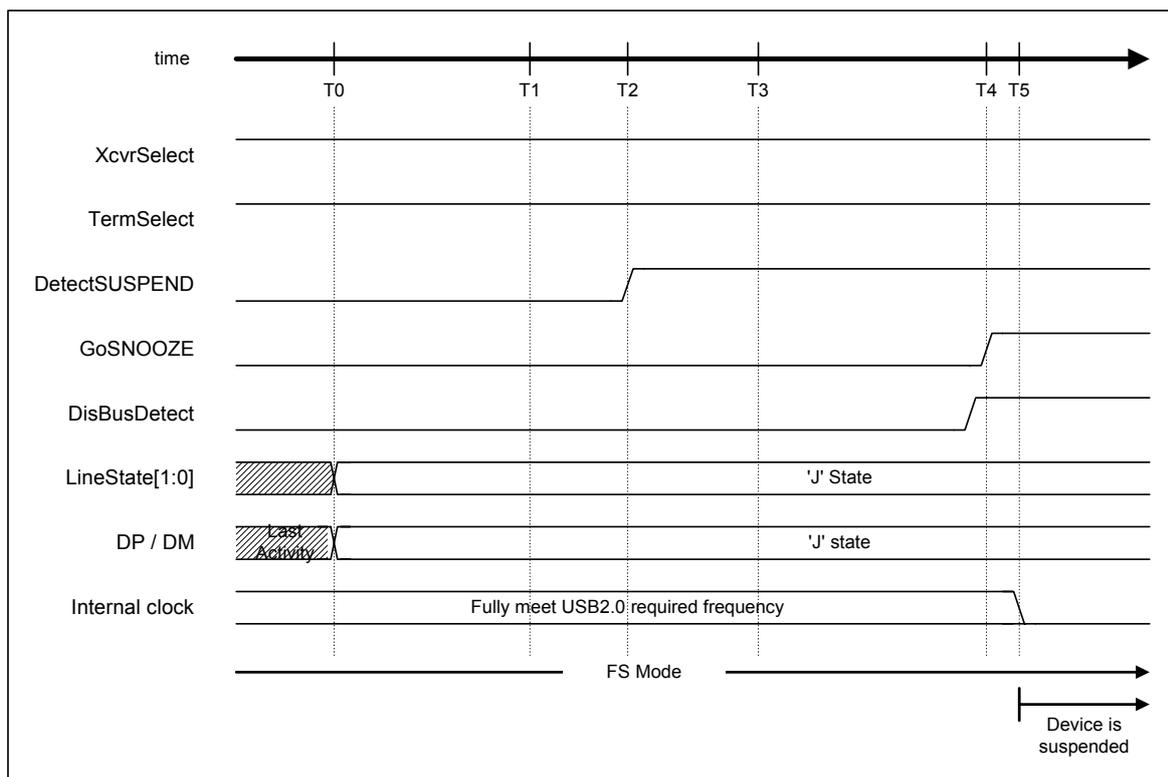


図 6-9 Suspend Timing (FS mode)

表 6-9 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 のサスペンド状態と判断する。	$T1 + 100\mu\text{s} < T2 \{T_{WTWRSTHS}\} < T1 + 875\mu\text{s}$
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	SnoozeUTMを'1'にセットし、完全にスヌーズに移行。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引っ張ってはいけない。(スヌーズ移行前に、DisBusDetectを'1'にセット)	$T0 + 10\text{ms} \{T_{2SUSP}\}$
T5	内部クロックが完全停止。(スヌーズ電流 8mA (typ))	$T5 < T4 + 10\mu\text{s}$

注: {}は、USB2.0 規格書で規格されている名称である。

6.2.7.11.3. リセット検出 (HS モード)

本 LSI が HS モードで動作している時に、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)。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、D_SIE_IntStat.DetectRESET ビットがセットされます。

この時、D_SIE_IntEnb.EnDetectRESET ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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ので、リセットの指示であると判断し、以降は、D_NegoControl.DisBusDetect ビットをセットした後に、HS Detection Handshake (後述) を行ってください。

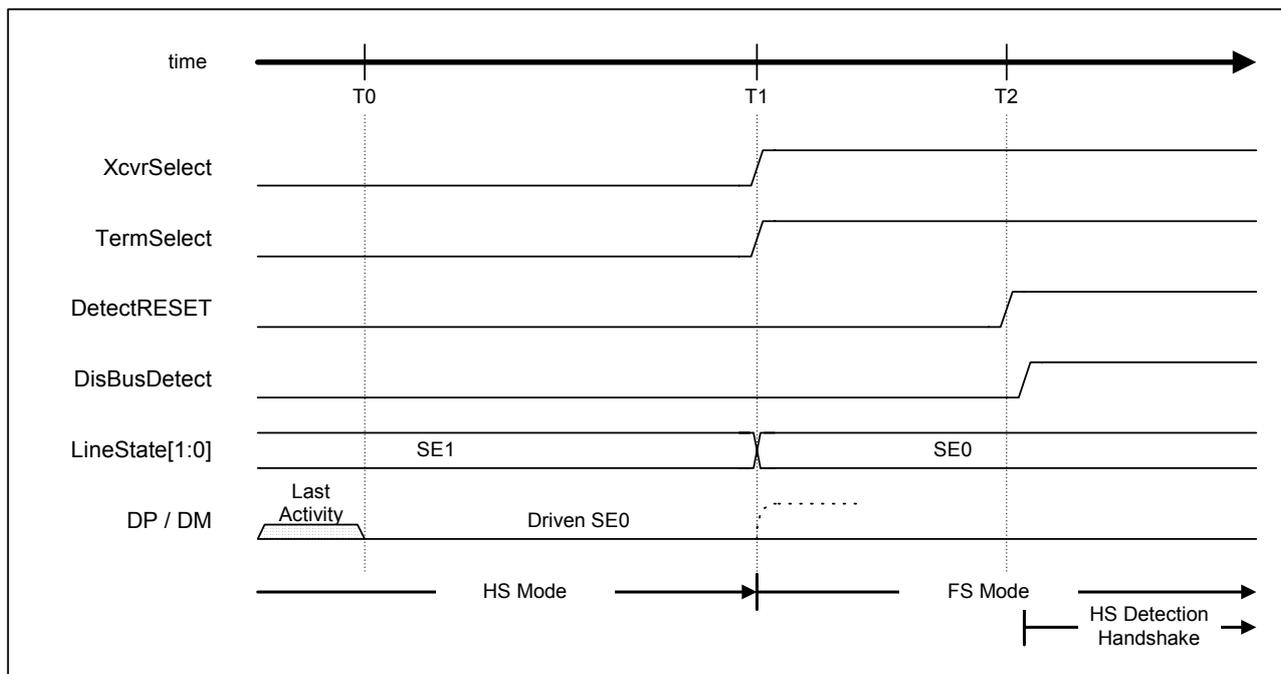


図 6-10 Reset Timing (HS mode)

表 6-10 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。	$T1 + 100\mu s < T2 \{T_{WTWRSTHS}\} < T1 + 875\mu s$

注: {} は、USB2.0 規格書で規格されている名称である。

6.2.7.11.4. リセット検出 (FS モード)

本 LSI が FS モードで動作している時に、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)、D_SIE_IntStat.DetectRESET ビットがセットされます。

この時、D_SIE_IntEnb.EnDetectRESET ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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ので、リセットの指示であると判断し、以降は UD_NegoControl.DisBusDetect ビットをセットした後に、HS Detection Handshake (後述)を行ってください。

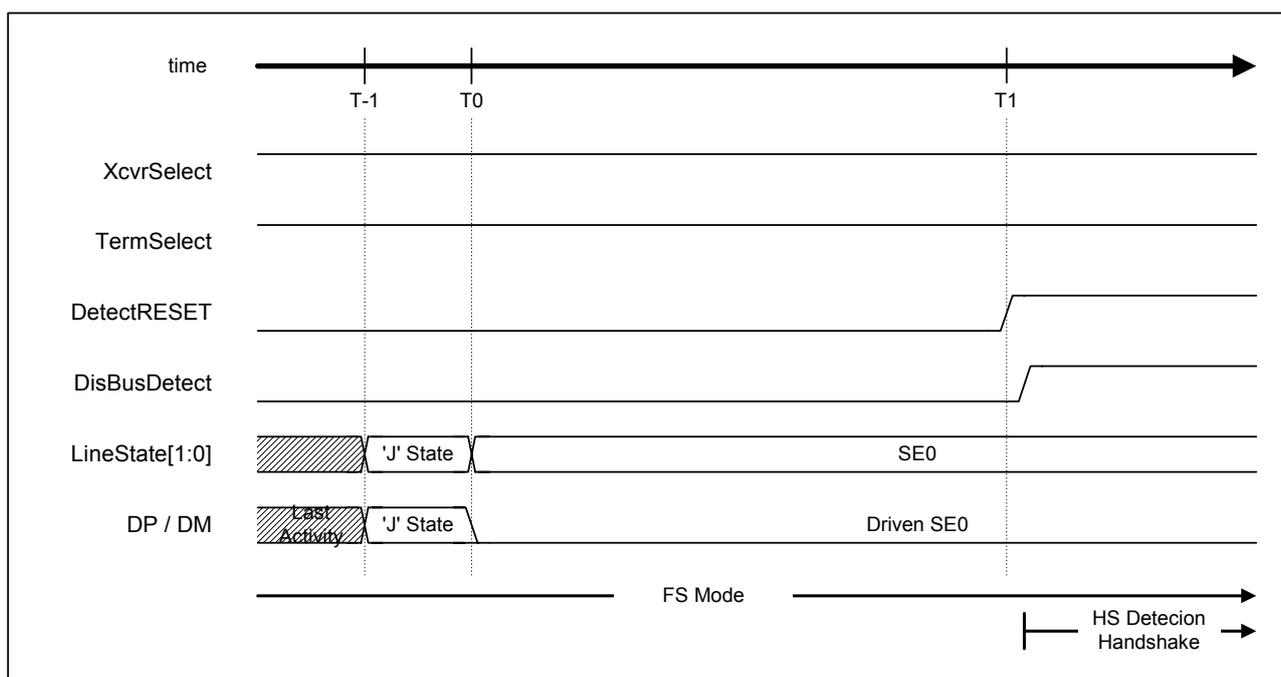


図 6-11 Reset Timing (FS mode)

表 6-11 Reset Timing Values (FS mode)

Timing Parameter	Description	Value
T-1	最後のバス・アクティビティ。	
T0	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。

6.2.7.11.5. HS Detection Handshake

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、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 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" を検出後直ちに HS Detection Handshake に移行してください。

本 LSI が FS モードで動作している状態では、2.5us 以上の "SE0" を検出後、HS Detection Handshake に移行してください。

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 のサスペンド状態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_XcvtControl.XcvtSelect、D_XcvtControl.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、3.125ms 以内に行われなければなりません。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 のサスペンド状態として判断し、"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 に移行してください。

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充分静定しており問題ありませんが、サスペンド中にスリープ/スヌーズ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 が行うために、必ず PM_Control_0.GoActDevice ビットを "1" にセットし、内部クロックを動作させてください。この動作の詳細は、パワーマネージメント機能 (6.5) を参照してください。

6.2.7.11.5.1. FS の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 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_XcvtControl.XcvtSelect と D_XcvtControl.TermSelect は両ビットともに FS モードでなければいけません (FS ターミネーション、即ち DP のプルアップ抵抗 (Rpu) を有効にし、HS ターミネーションを無効にします)。

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、"0" で埋め尽くされたデータが準備されます (T1)。これは、バス上に "HS K" (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、downstream port に "HS K" (chirp) が送出されます。送出終了後、downstream port からの chirp を待ちます (T2)。通常、HS をサポートしている 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)、downstream port が HS をサポートしていない場合 (今回の場合) は、T4 の時点でも 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。

この時、D_SIE_IntEnb.EnChirpCmp ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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ので、HS Detection Handshake が終了したと判断してください。

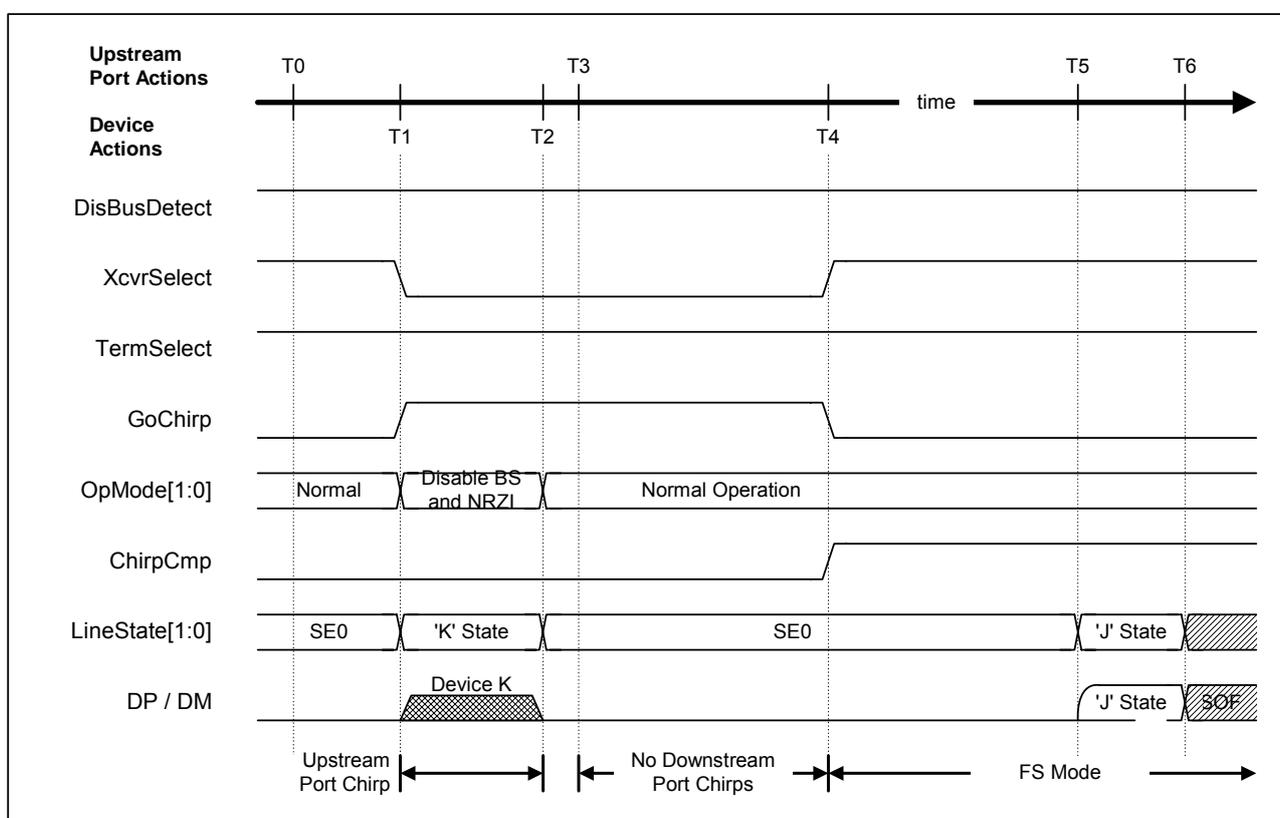


図 6-12 HS Detection Handshake Timing (FS mode)

表 6-12 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。	$T1 + 1.0\text{ms} \{T_{UCH}\} < T2 < \text{HS Reset } T0 + 7.0\text{ms} \{T_{UCHEND}\}$
T3	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。	$T2 < T3 < T2 + 100\mu\text{s} \{T_{WTDCH}\}$
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。	$T2 + 1.0\text{ms} < T4 \{T_{WTF5}\} < T2 + 2.5\text{ms}$
T5	リセットシーケンスの終了。	$\text{HS Reset } T0 + 10\text{ms} \{T_{DRST} (\text{Min})\}$
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。

6.2.7.11.5.2. HS の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モードでなければなりません(FSターミネーション、即ちDPのプルアップ抵抗(Rpu)を有効にし、HSターミネーションを無効にします)。

まずは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、“0”で埋め尽くされたデータが準備されます(T1)。これは、バス上に“HS K”(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”(chirp)が送出されます。送出終了後、downstream portからの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。

この時、D_SIE_IntEnb.EnChirpCmpビットおよびDeviceIntEnb.EnD_SIE_IntStatビットがセットされ、MainIntEnb.EnDeviceIntStat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ので、HS Detection Handshakeが終了したと判断してください。

この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ージーな為、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”を出力します。

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。

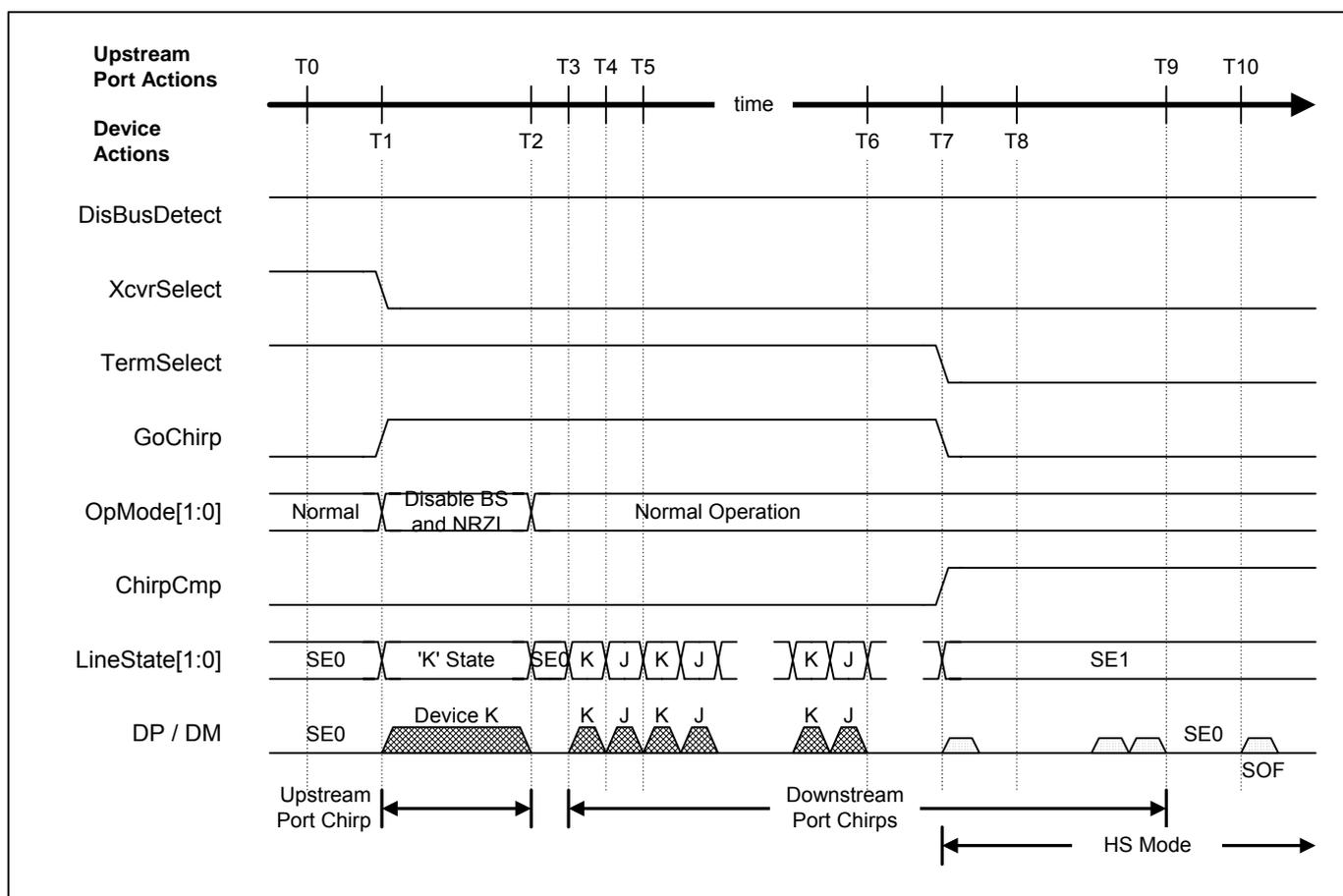


図 6-13 HS Detection Handshake Timing (HS mode)

表 6-13 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	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	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	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、パケット受信中と認識されることは無い。	T8
T9	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。

6.2.7.11.5.3. スヌーズ中にリセットされた場合

本LSIは、スヌーズ状態では、内部クロックは出力されていません。ここでは、発振回路は動作しているものとして(スリープ状態ではなく、スヌーズ状態)、その時の動作を説明します。

スヌーズ状態で、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 DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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、PM_Control_0.GoActDevice 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_1.PM_State[2:0]が"ACT_DEVICE"になり、内部クロックが出力され始めます。この後 HS Detection Handshake (前述)を行ってください。

この時、発振回路を停めていなければ(スリープ状態からの復帰でなければ)、内部クロックは USB2.0 規格に則った周波数精度で出力されています。

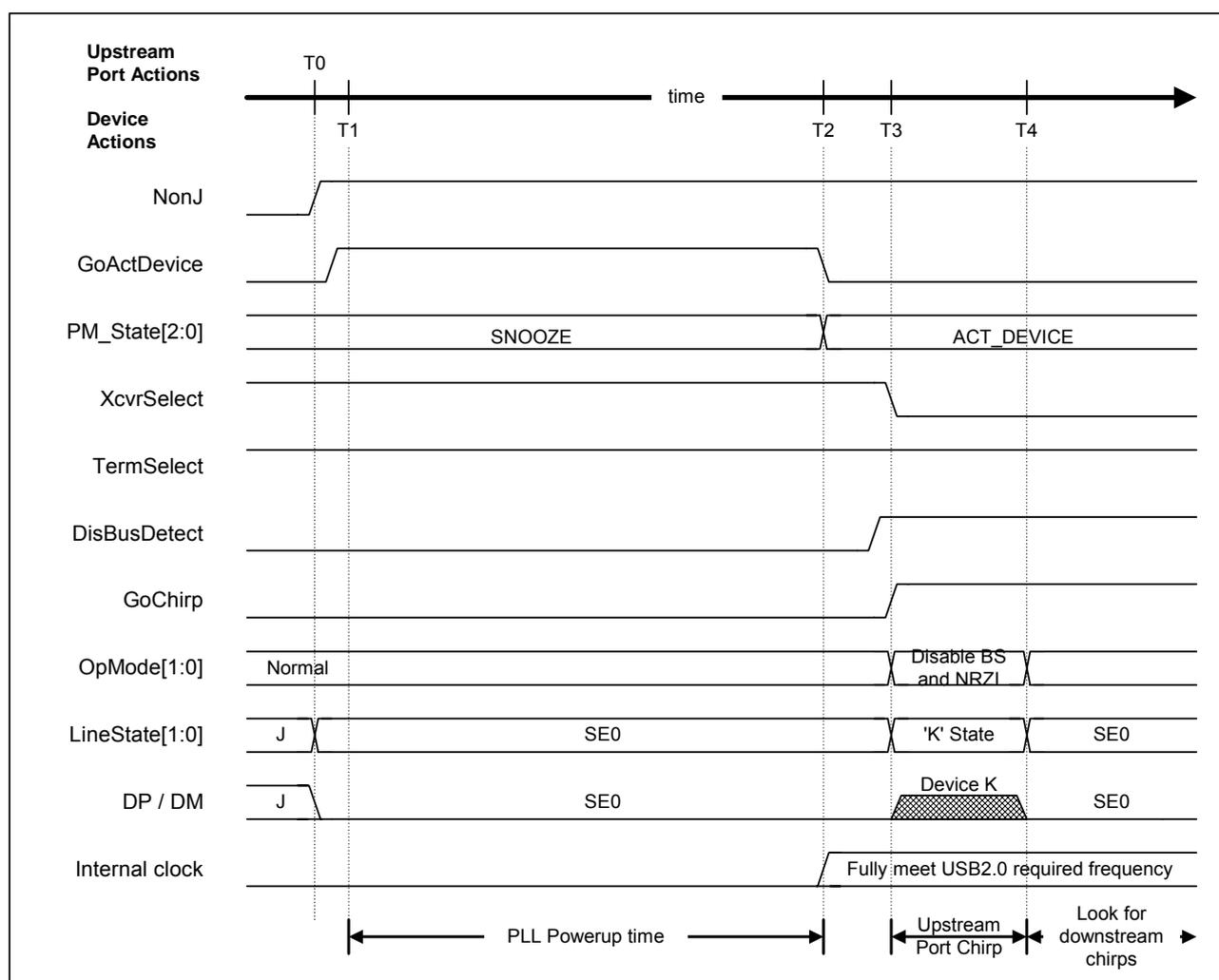


図 6-14 HS Detection Handshake Timing from Suspend

表 6-14 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)。

6.2.7.11.6. 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 から引っ張ることはできません。

デバイスは、リモート・ウェイクアップする為に、まずスリープ/スヌーズから復帰します。D_SIE_IntEnb.EnNonJ ビットをクリアし、PM_Control_0.GoActDevice ビットをセットし (T0)、PLL パワーアップ時間経過後 (T1)、PM_Control_1.PM_State[2:0] ビットが "ACT_DEVICE" になると同時に内部クロックが出力され始めます。この時、発振回路を停めていなければ、この内部クロックは、USB2.0 規格に則った周波数精度で出力されています。

その後、D_NegoControl.SendWakeup ビットをセットし、レジューム信号を送出します (T2)。この時、内部では、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) を送ります。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)。

レジューム信号を送出し始めてから約 1ms 後、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 (T4)、この時点では 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。

そこで、D_NegoControl.RestoreUSB ビットをセットします。一定時間経過後、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 ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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 の必要はありません。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。実際、FS モードの時には、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変化はありません。

本 LSI が、スヌーズ状態 (PM_Control_1.PM_State[2:0] ビットが "SNOOZE") では、内部クロックは出力されていません。ここでは、発振回路は動作しているものとして (スリープ状態ではなく、スヌーズ状態)、その時の動作を説明しています。

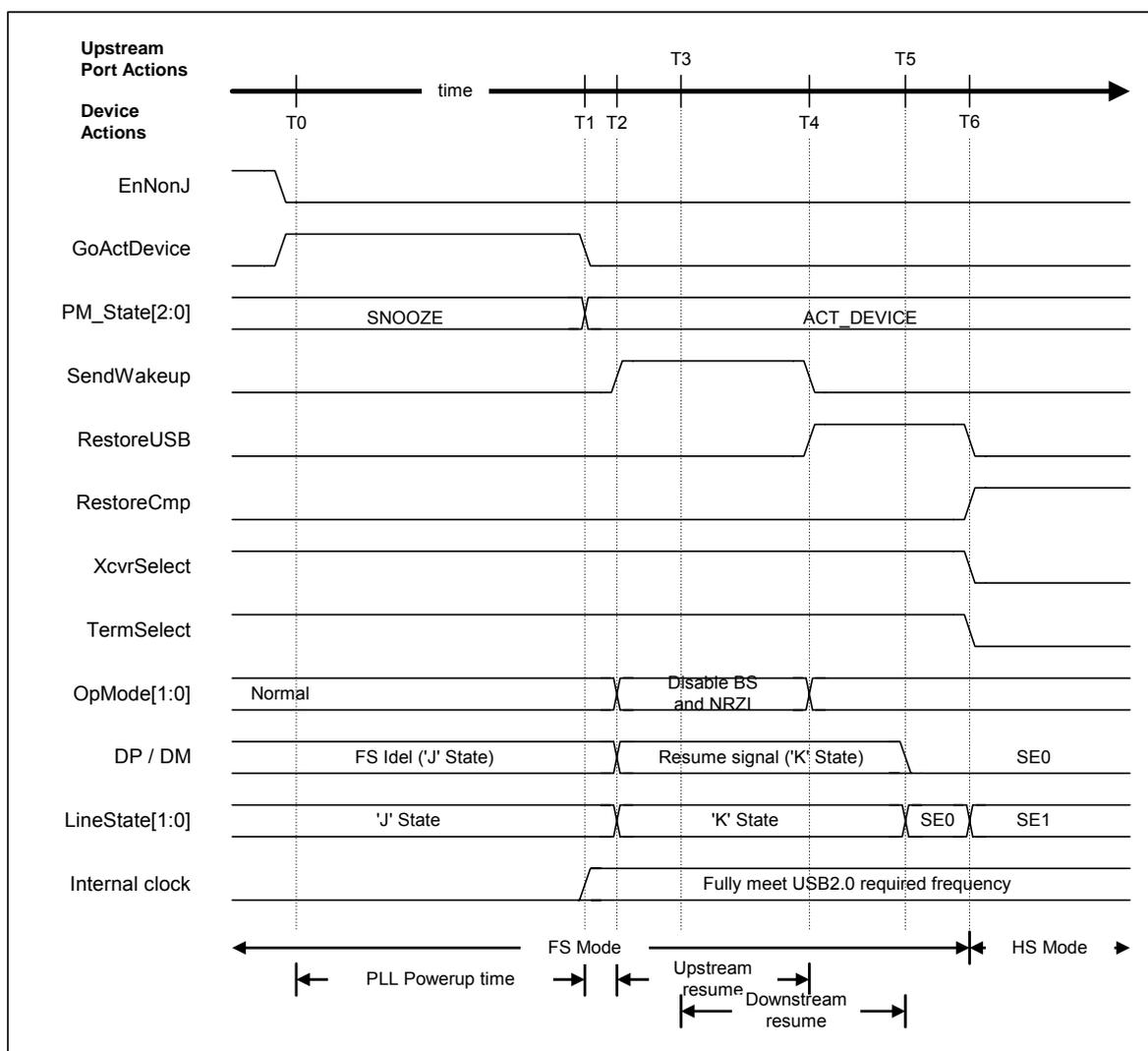


図 6-15 Assert Resume Timing (HS mode)

表 6-15 Assert Resume Timing Values (HS mode)

Timing Parameter	Description	Value
T0	レジューム開始。GoActDevice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 ACT_DEVICE 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	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	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。

6.2.7.11.7. レジュームの検出

本 LSI がスヌーズしている時、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、downstream port からのウェイクアップの指示(レジューム指示)を受け取ったことになります(T0)。この時、発振回路が動作を停止していなければ(スリープ状態でなければ)、SIE_IntStat.NonJ ビットがセットされます。この時、D_SIE_IntEnb.EnNonJ ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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。

まず、PM_Control_0.GoActDevice 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_1.PM_State[2:0] が"ACT_DEVICE"になると同時に内部クロックが出力され始めます。この時、発振回路を停めていなければ、この内部クロックは USB2.0 規格に則った周波数精度で出力されています。

そこで、D_NegoControl.RestoreUSB をセットします。一定時間経過後、downstream port はレジューム信号の送出を停止し(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_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 ビットおよび DeviceIntEnb.EnD_SIE_IntStat ビットがセットされ、MainIntEnb.EnDeviceIntStat 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。

ここでは、発振回路は動作しているものとして(スリープ状態ではなく、スヌーズ状態)、その時の動作を説明しています。

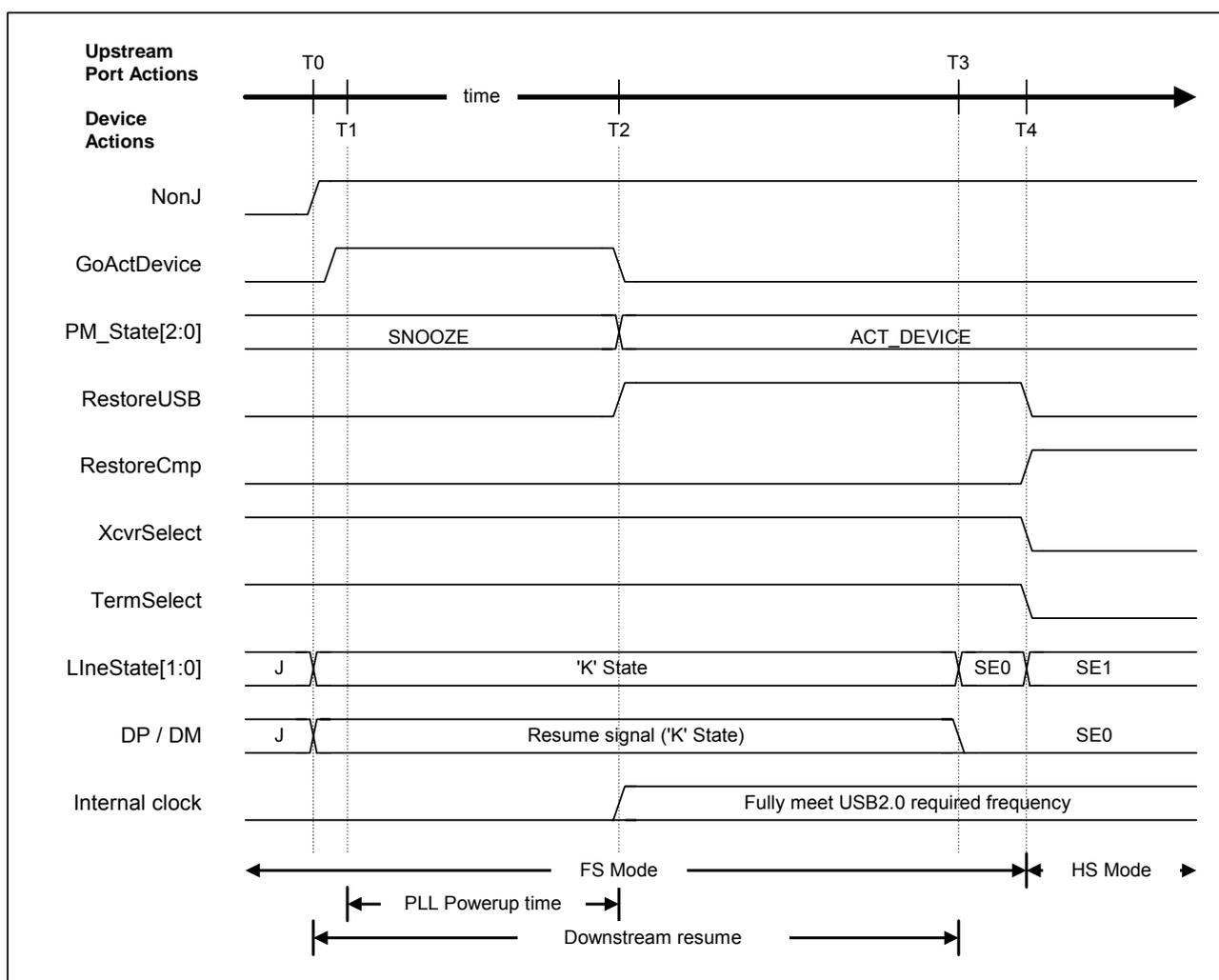


図 6-16 Detect Resume Timing (HS mode)

表 6-16 Detect Resume Timing Values (HS mode)

Timing Parameter	Description	Value
T0	downstream port が FS の "K" を送出。NonJ が '1' になる。	0 (reference)
T1	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。 LineState[1:0] で "K" を確認後、RestoreUSB を '1' にセットする。	$T1 + 250\mu s < T2$
T3	downstream port が FS の "K" 送出を終了。同時に downstream port は USB のサスペンド以前の HS モードに移行。	$T2 + 20\text{ms} \{T_{\text{DRSM DN}}\}$
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。

6.2.7.11.8. 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。

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、 V_{BUS} が "H" になり、同時に D_USB_Status.VBUS ビットがセットされます(T1)。この時スヌーズ状態にしていた場合は、PM_Control_0.GoActDevice 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_1.PM_State[2:0] が "ACT_DEVICE" 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、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。

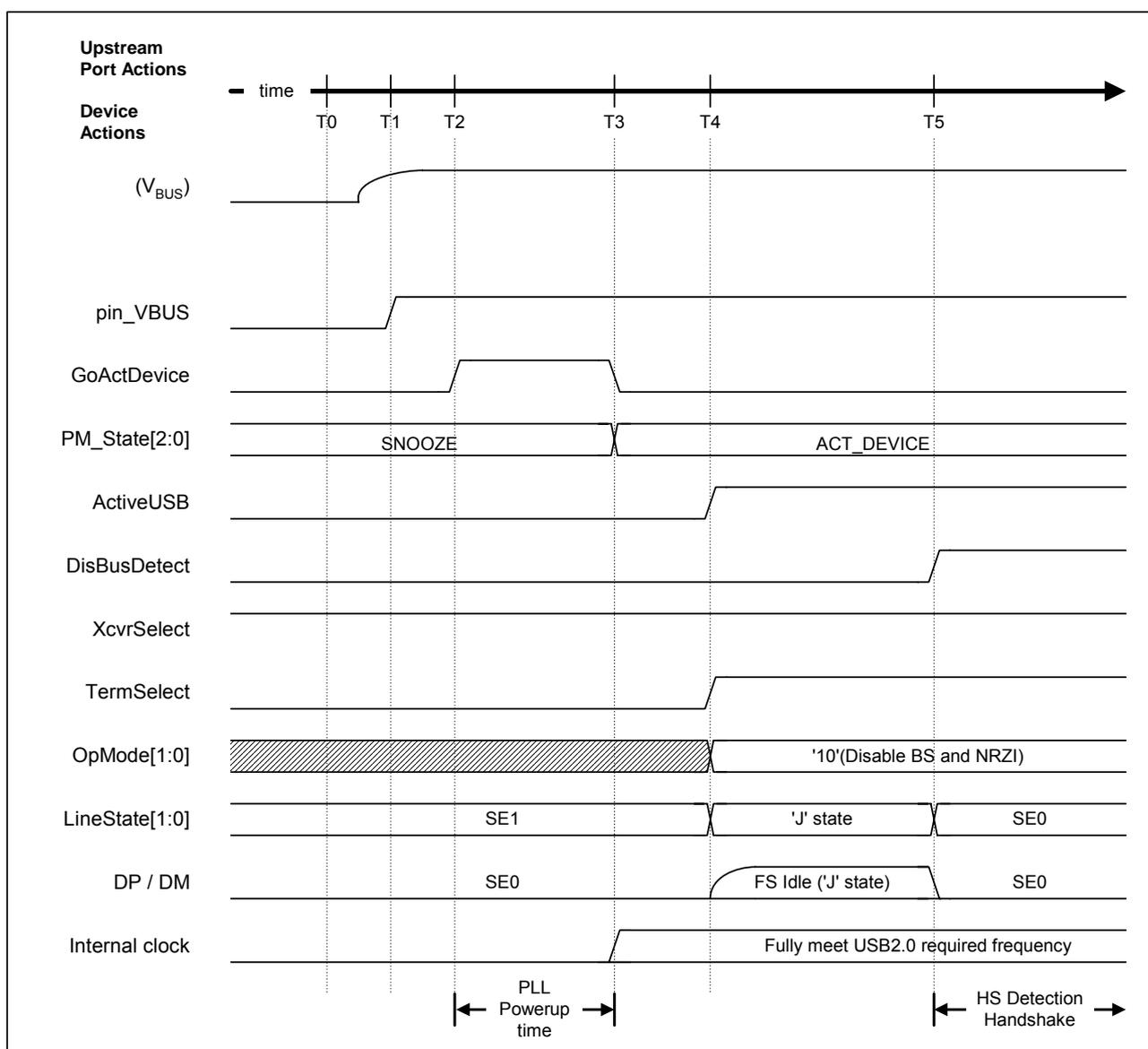


図 6-17 Device Attach Timing

表 6-17 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	GoActDevice を '1' にセット。	T2
T3	PM_State が ACT_DEVICE'1' 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	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。

6.3. USB ホスト制御

6.3.1. チャンネル

6.3.1.1. 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に各種設定レジスタを、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。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 6-18 にチャンネルの概念を示します。

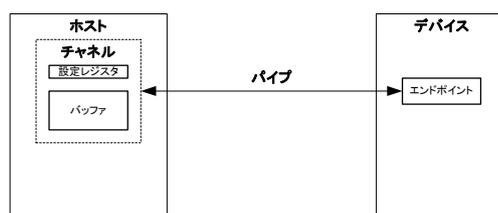


図 6-18 チャンネル概念図

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をセットします。転送実行をセットした後は、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。

各チャンネルのバッファは、ファームウェアにより LSI に内蔵された RAM の任意の領域に割り当てることができ、それぞれ独立した FIFO として動作します。

図 6-19 に転送を行う際の基本的な流れを示します。

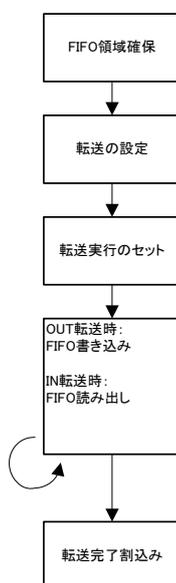


図 6-19 チャンネルの基本的な転送の手順

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。

なお、同時に設定できるインタラプト転送の最大数は4本です。

各チャンネルが対応する転送タイプを表 6-18 に示します。

表 6-18 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(後述)使用可
CHa	バルク転送	バルクオンリーサポート機能(後述)使用可
CHb, CHc CHd, CHe	バルク転送 インタラプト転送	

6.3.1.2. コントロール専用チャンネル

本 LSI では、コントロール専用チャンネル(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、チャンネル CH0 を時分割して使用します。

表 6-19 にコントロール専用チャンネル(CH0)の基本設定項目を示します。

表 6-19 コントロール専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_CH0Config_0.SpeedMode	チャンネル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	チャンネル 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0x0~0xFの間の任意の値に設定します。
エンドポイントナンバー	H_CH0FuncAdrs.EP_Number	チャンネル CH0 に対応する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。
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	チャンネルCH0のIRPのデータ数をバイト単位で設定します。
FIFO 領域	H_CH0StartAdrs_H, H_CH0StartAdrs_L, H_CH0EndAdrs_H, H_CH0EndAdrs_L	チャンネル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、チャンネル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、機能説明のFIFOの項を参照して下さい。
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。

6.3.1.3. 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 6-20 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。

表 6-20 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(パルク、インタラプト)を設定します。
トグルモード	H_CHx{x=b-e}Config_1.ToggleMode	インタラプト転送におけるトグルビットの動作モードを設定します。
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	各チャンネルのマックスパケットサイズを、1Byte～512Byte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	各チャンネルが管理するエンドポイントを含む機能のUSB アドレスを0x0～0xFの間の任意の値に設定します。
エンドポイントナンバー	H_CHx{x=a-e}FuncAdrs.EP_Number	各チャンネルに対応する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。
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	各チャンネルのIRPのデータ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	インタラプト転送におけるトークン発行間隔(周期)を設定します。
FIFO 領域	H_CHx{x=a-e}StartAdrs_H, H_CHx{x=a-e}StartAdrs_L, H_CHx{x=a-e}EndAdrs_H, H_CHx{x=a-e}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、機能説明の FIFO の項を参照して下さい。

6.3.1.4. チャンネル使用例

6.3.1.4.1. 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 6-20 に示します。

このクラスで使用するバルク IN 転送およびバルク OUT 転送は逐次処理が可能です。

コントロール転送の IRP は CH0 を使用します。

一方、バルク IN 転送の IRP およびバルク OUT 転送の IRP は、CHa を逐次的に使用します。

CHa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(6.3.8 参照)があり、転送におけるCPU負荷の低減と転送効率の向上を図ることができます。

なお、バルクオンリーサポート機能を使用する必要がない場合には、バルク IN 転送の IRP およびバルク OUT 転送の IRP 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。

ハードウェアは、チャンネルに設定されているIRPIに対し転送のスケジューリング(6.3.2 参照)を行い、トランザクションを実行します。

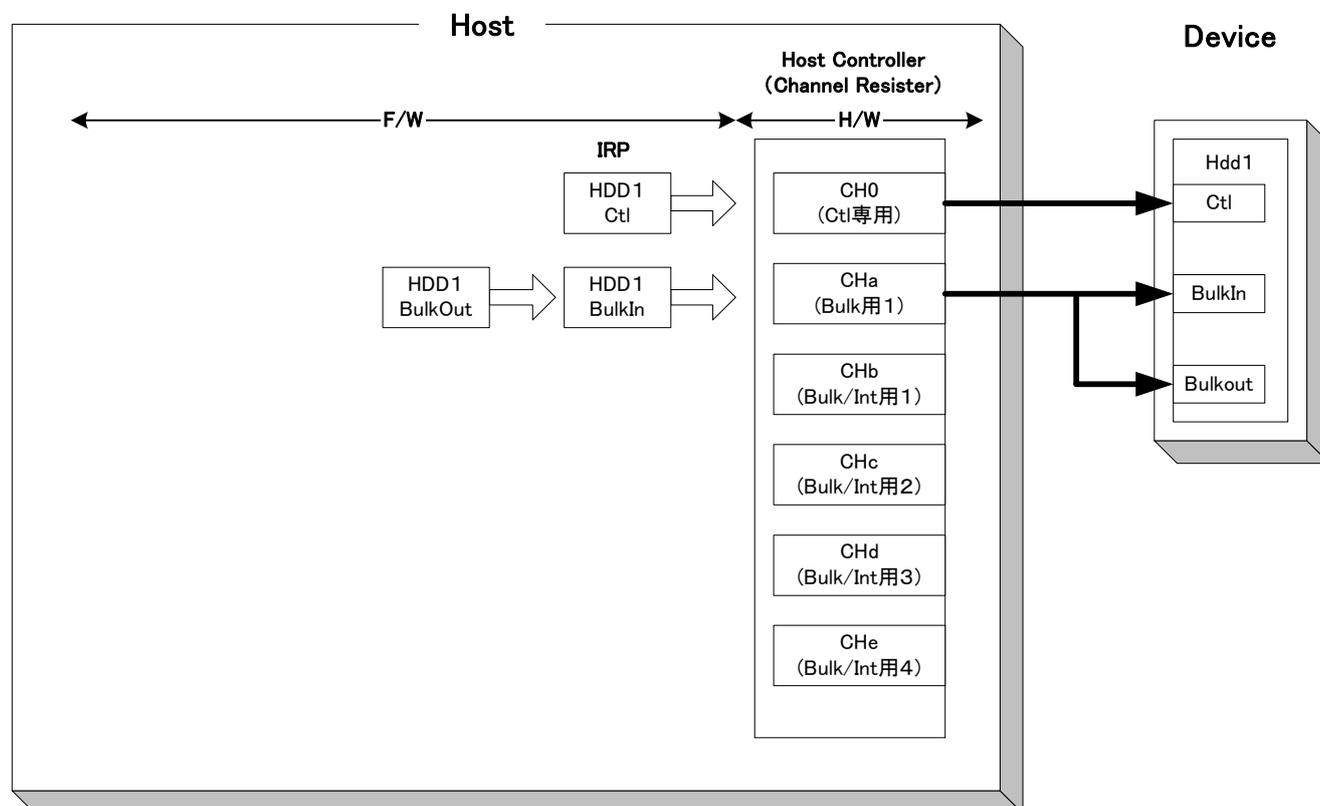


図 6-20 チャンネル使用例(ストレージデバイスを1台接続した場合)

6.3.1.4.2. 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 6-21 に示します。

このクラスで使用するバルク IN 転送およびバルク OUT 転送は並列処理が必要です。

コントロール転送の IRP は CH0 を使用します。

バルク IN 転送の IRP およびバルク OUT 転送の IRP、およびインタラプト IN 転送の IRP は汎用チャンネルに個別(例えば CHb, CHc, CHd)に割り当てて使用します。

ハードウェアは、チャンネルに設定されているIRPIに対し転送のスケジューリング(6.3.2 参照)を行い、トランザクションを実行します。

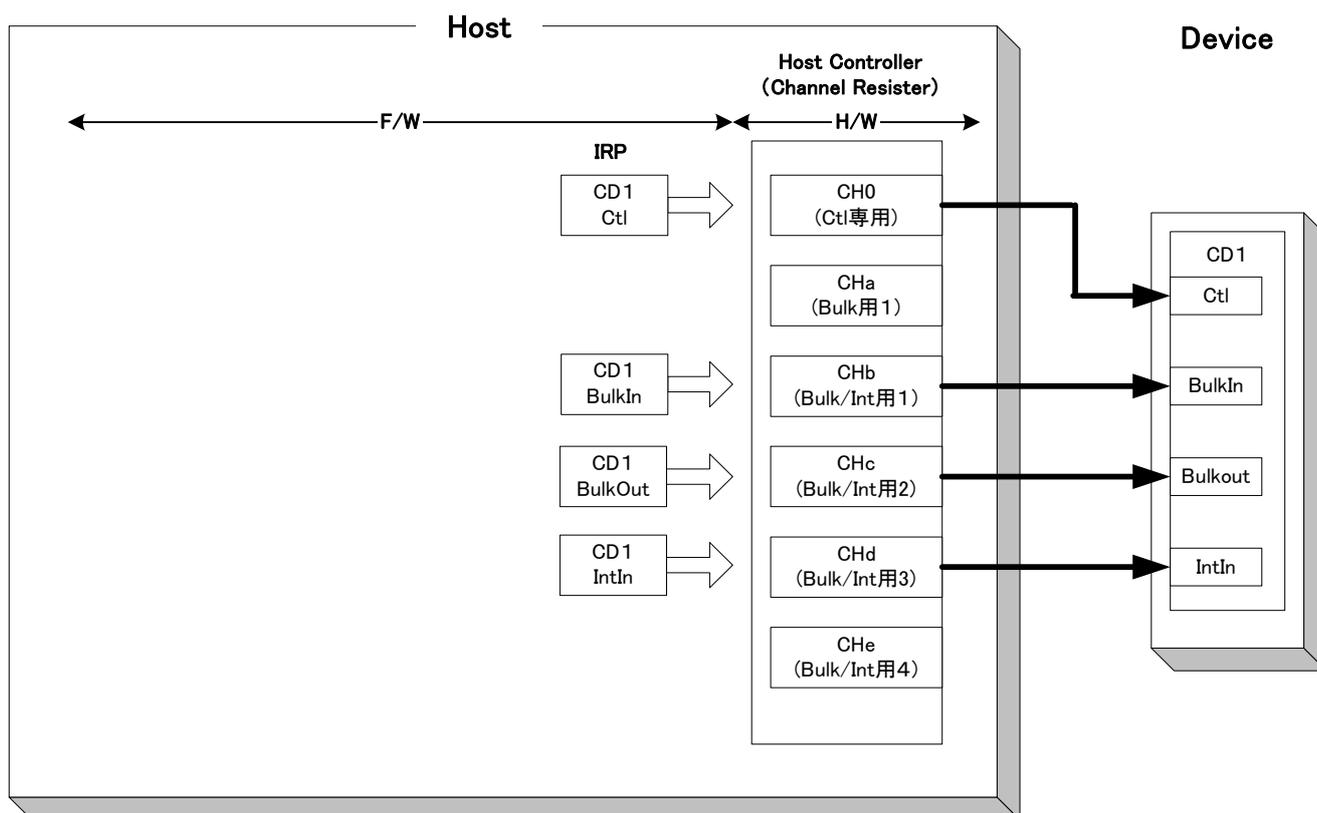


図 6-21 チャンネル使用例(コミュニケーションデバイスを1台接続した場合)

6.3.1.4.3. ヒューマンインタフェースデバイスを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 6-22 に示します。

コントロール転送の IRP は CH0 を使用します。

インタラプト IN 転送の IRP は、汎用チャンネル(例えば CHd)に割り当てて使用します。

ハードウェアは、チャンネルに設定されているIRPIに対し転送のスケジューリング(6.3.2 参照)を行い、トランザクションを実行します。

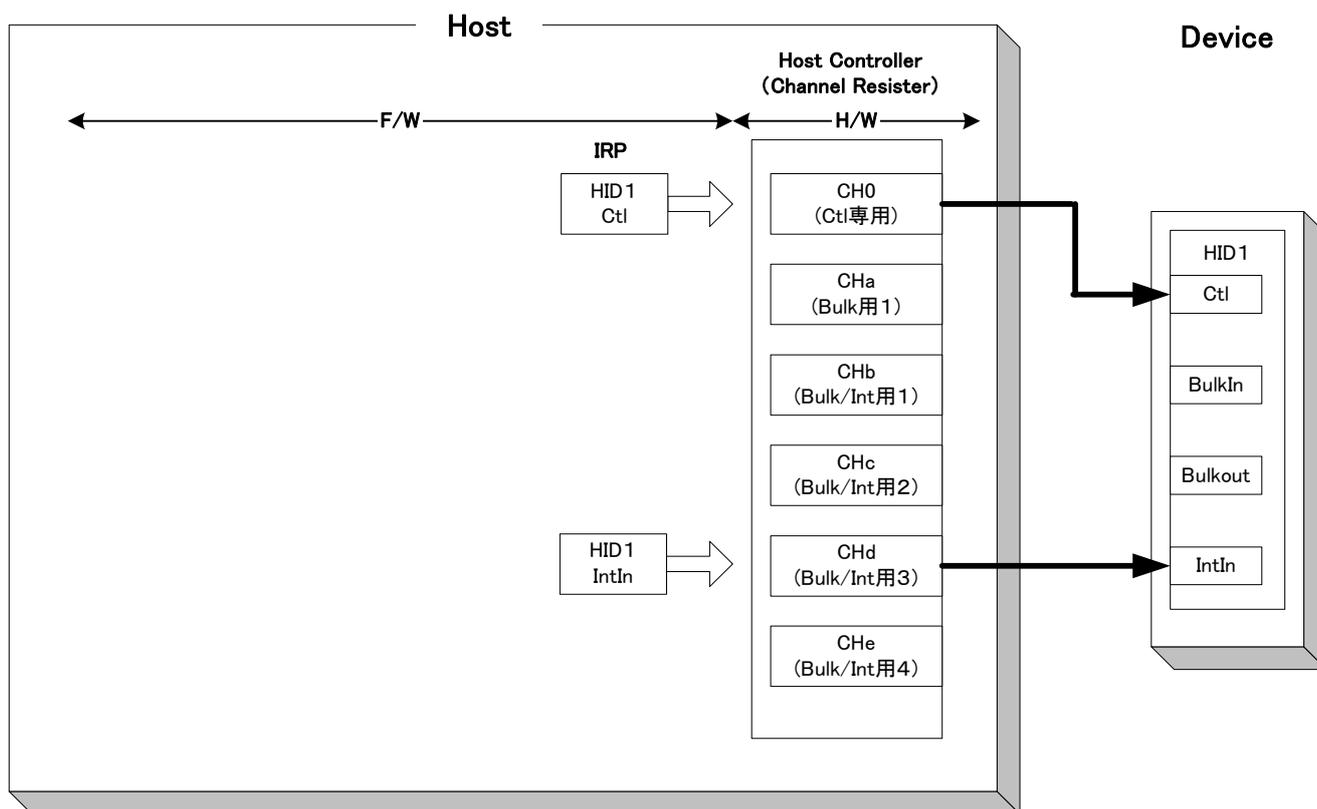


図 6-22 チャンネル使用例(ヒューマンインタフェースデバイスを1台接続した場合)

6.3.1.4.4. 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 6-23 に示します。

このクラスで使用するバルク IN 転送およびバルク OUT 転送は逐次処理が可能です。

コントロール転送の IRP は CH0 を使用します。

インタラプト IN 転送の IRP は、汎用チャンネル(例えば CHd, CHe)に割り当てて使用します。

一方、バルク IN 転送の IRP およびバルク OUT 転送の IRP は、CHa を逐次的に使用します。

CHa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(6.3.8 参照)があり、転送におけるCPU負荷の低減と転送効率の向上を図ることができます。

なお、バルクオンリーサポート機能を使用する必要がない場合には、バルク IN 転送の IRP およびバルク OUT 転送の IRP を汎用チャンネルに個別に割り当てることも可能です。

ハードウェアは、チャンネルに設定されているIRPに対し転送のスケジューリング(6.3.2 参照)を行い、トランザクションを実行します。

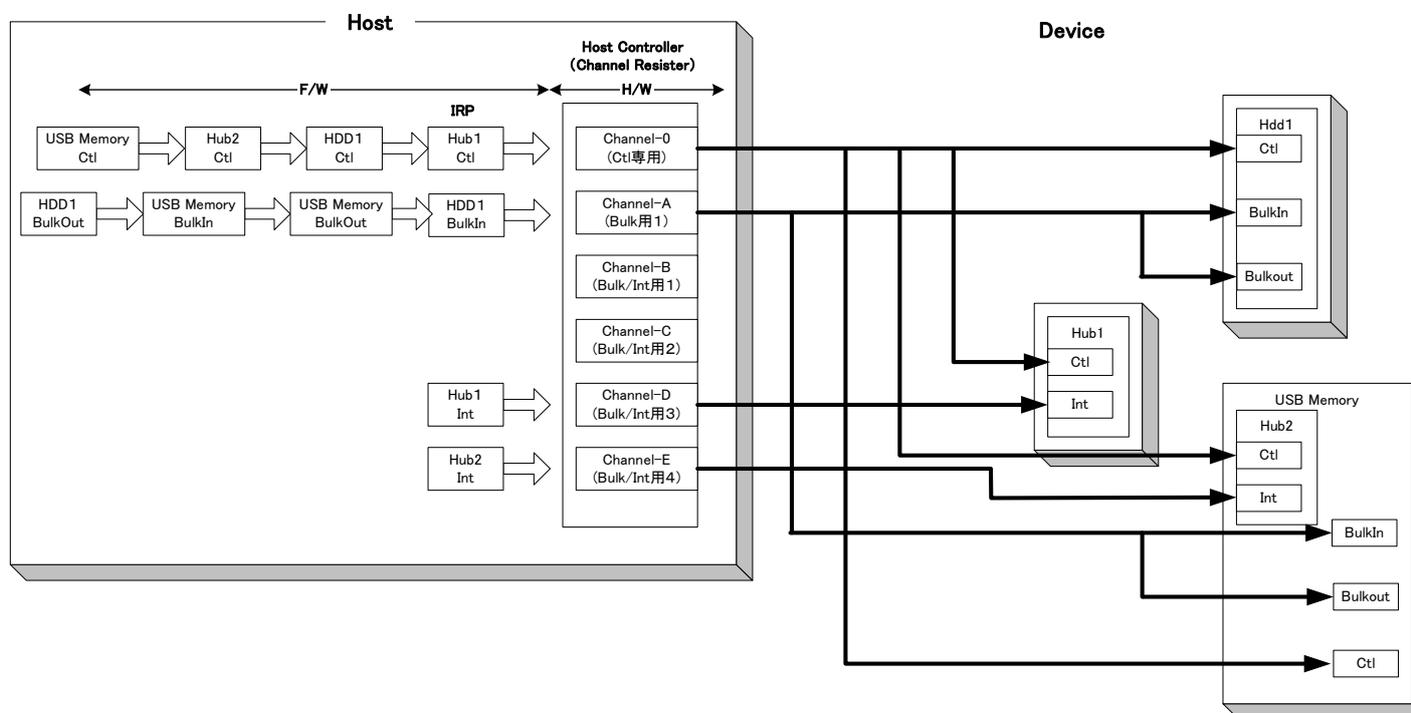


図 6-23 チャンネル使用例(ハブを介してストレージデバイスを2台接続した場合)

6.3.2. 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。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エンドポイントに対して転送を行います。

表 6-21 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 6-21 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 6-22 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 6-22 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。

6.3.3. トランザクション

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、レジスタ説明の章を参照して下さい。

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。

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できます。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できます。

表 6-23 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 6-23 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 6-24 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 6-24 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。

6.3.3.1. 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、該当チャンネルが選択された場合、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、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_CH0Control.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_CH0ConnditionCode)を適宜設定します。そして、ファームウェアに対し、ChangeCondition ステータス(H_CH0IntStat.ChangeCondition ビット)を発行します。

図 6-1 にSETUP トランザクションの様子を図示します。(a) LSIが、この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。

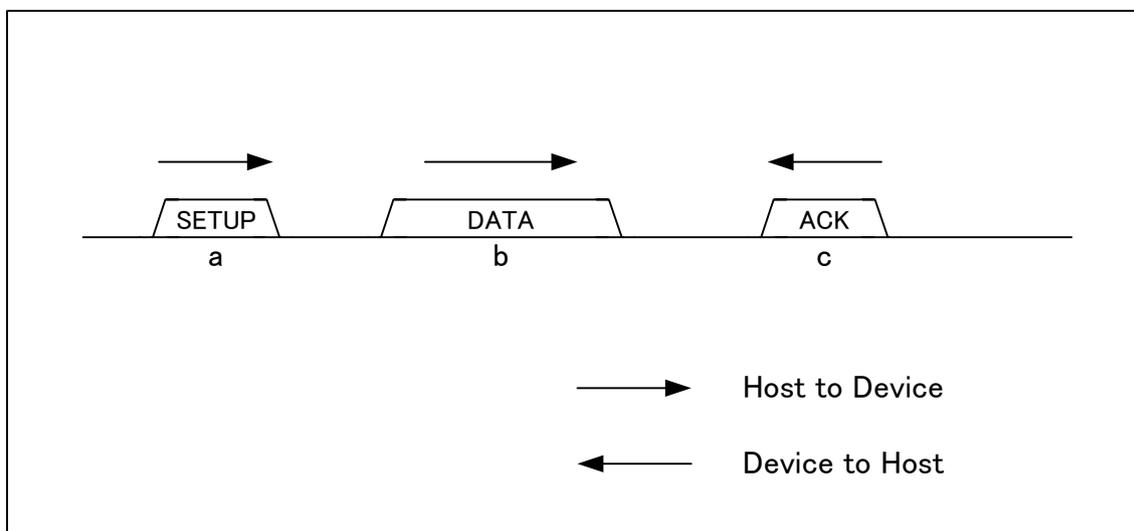


図 6-24 SETUP トランザクション

6.3.3.2. バルク OUT トランザクション

CHx 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、該当チャンネルが選択された場合、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。

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=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=a-e}ConnditionCode) 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、領域も開放せず、コンディションコード (H_CHx{x=a-e}ConnditionCode) 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、コンディションコード (H_CHx{x=a-e}ConnditionCode) を適宜設定します。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。

図 6-2 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、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) 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。

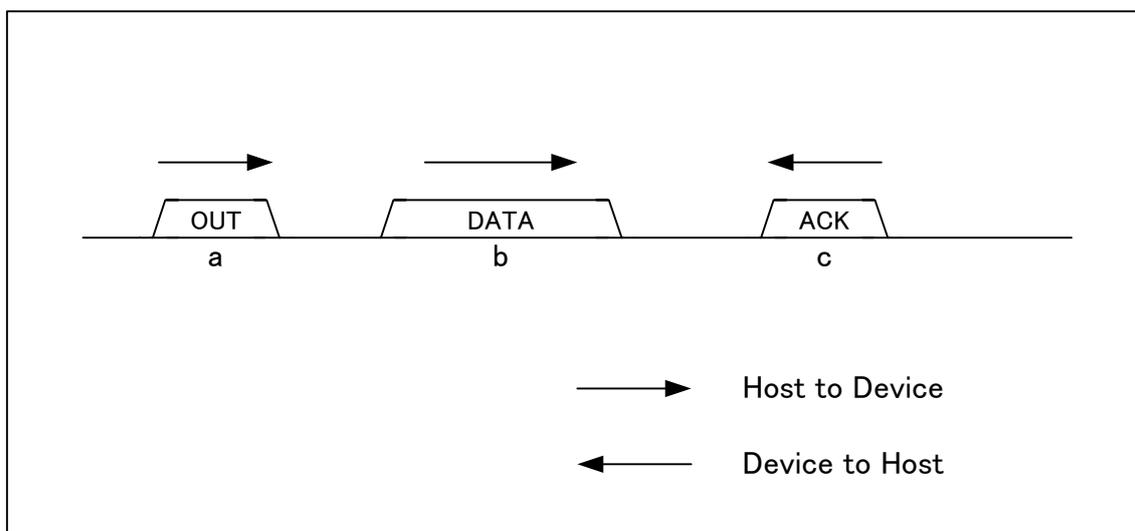


図 6-25 OUT トランザクション

6.3.3.3. インタラプト OUT トランザクション

CHx 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、該当チャ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、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}ConnditionCode) 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、領域も開放せず、コンディションコード (H_CHx{x=b-e}ConnditionCode) 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。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、コンディションコード (H_CHx{x=b-e}ConnditionCode) を適宜設定します。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。

6.3.3.4. バルク IN トランザクション

CHx の基本設定レジスタにおいて、転送種別 (H_CHx{x=a-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、該当チャ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。

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}ConnditionCode) 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}ConnditionCode) 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}ConnditionCode) 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}ConnditionCode) 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}ConnditionCode) 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}ConnditionCode) 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 6-3 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、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) 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。

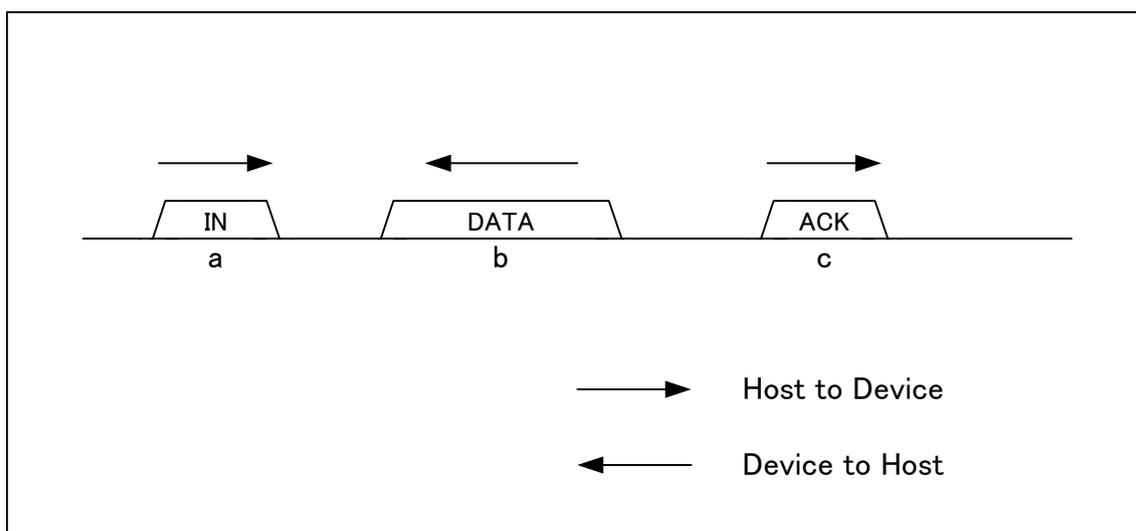


図 6-26 INトランザクション

6.3.3.5. インタラプト IN トランザクション

CHx 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、該当チャ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了します。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}ConnditionCode) 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}ConnditionCode) 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}ConnditionCode) 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}ConnditionCode) 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、ビットスタッピングエラー、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}ConnditionCode) 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}ConnditionCode) 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。

6.3.3.6. 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、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、ハードウェアは 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 ACK を受信した場合、ハードウェアは OUT トランザクションを行う状態に戻ります。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 NAK を受信した場合、ハードウェアは引き続き 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 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}ConnditionCode) 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}ConnditionCode) 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。そしてエラーが 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 6-4 に、PINGトランザクションに対してACK応答する様子を图示します。(a)LSII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。

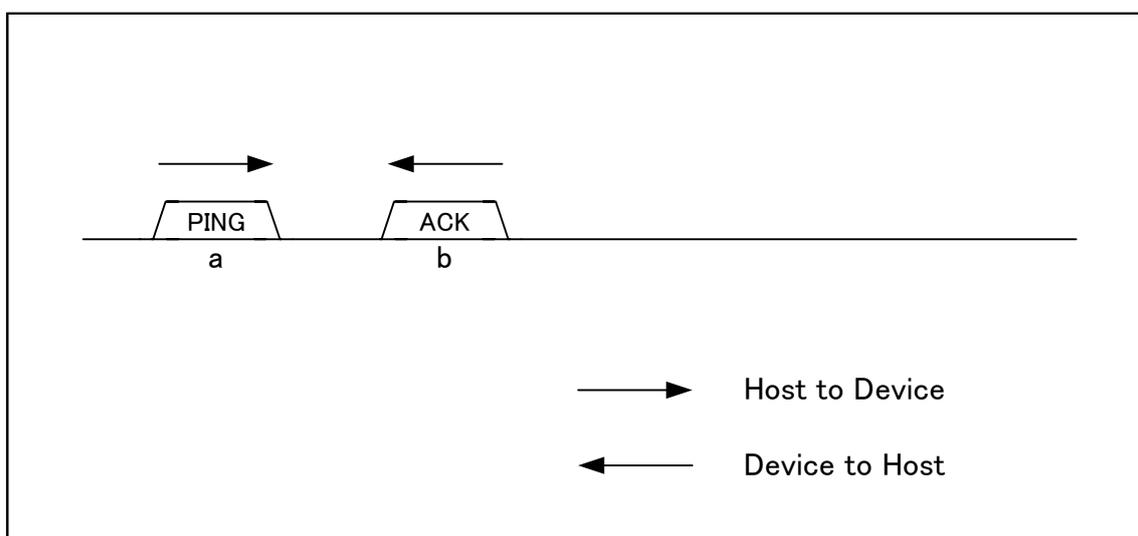


図 6-27 PINGトランザクション

6.3.3.7. 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。使用するチャンネルの転送速度 (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。使用するチャンネルの転送速度 (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ブルを伴って送信します。プリアンブルは FS bit time にて送信し、プリアンブ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 6-28 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)LSII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OUTトークンを先頭にプリアンブルを付与して発行します。(b)LSII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ブルを付与して送信します。(c)LSII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。

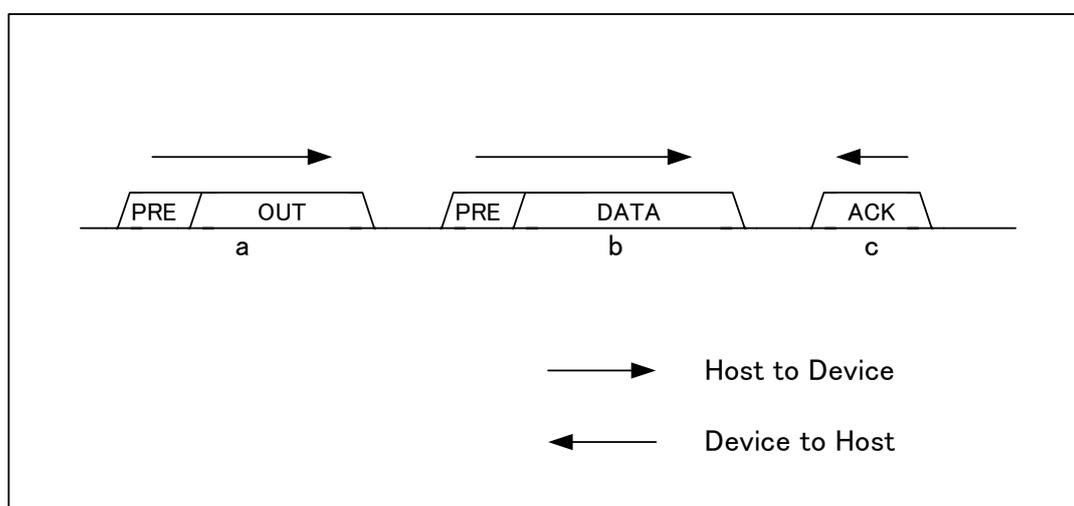


図 6-28 Preamble が付与された OUT トランザクション

図 6-29 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)LSIIは、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INTトークンを先頭にプリアンブ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できます。LSIIはこのデータを該当するチャンネルのFIFOに書き込みます。(c)LSIIはデータを受信出来るとACK応答を先頭にプリアンブ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。

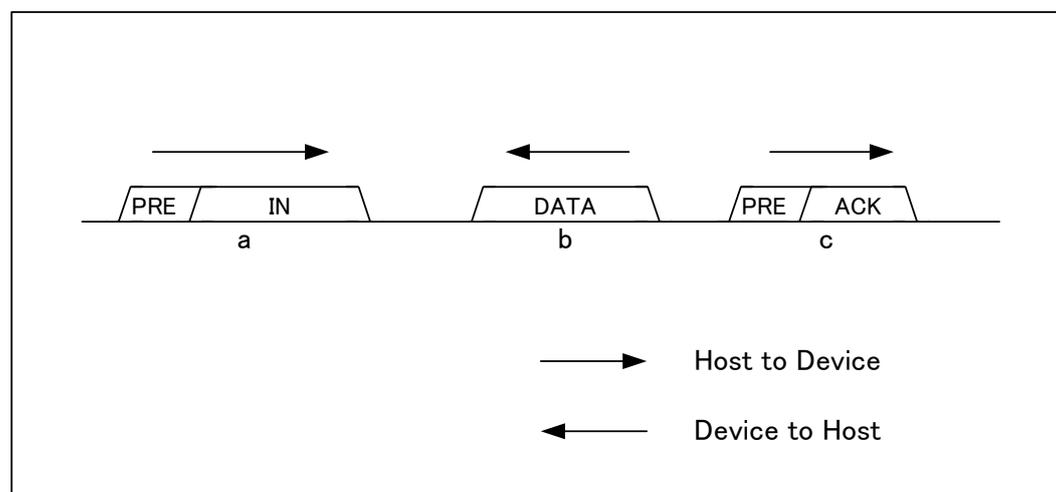


図 6-29 Preamble が付与された IN トランザクション

6.3.3.8. 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。使用するチャンネルの転送速度(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。

該当チャン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～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。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。

スタートスプリットトランザクション～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、最終コンプリートスプリットトランザクション以外の、個々のトランザクションの正常な完結による、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～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}ConnditionCode)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。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。

6.3.4. 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します。

図 6-30 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。

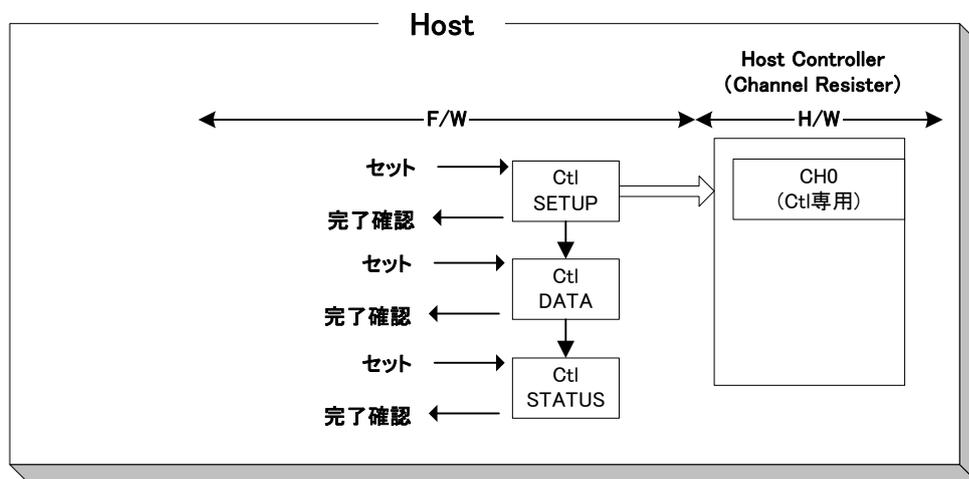


図 6-30 コントロール転送の制御

図 6-5 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。(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INT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。

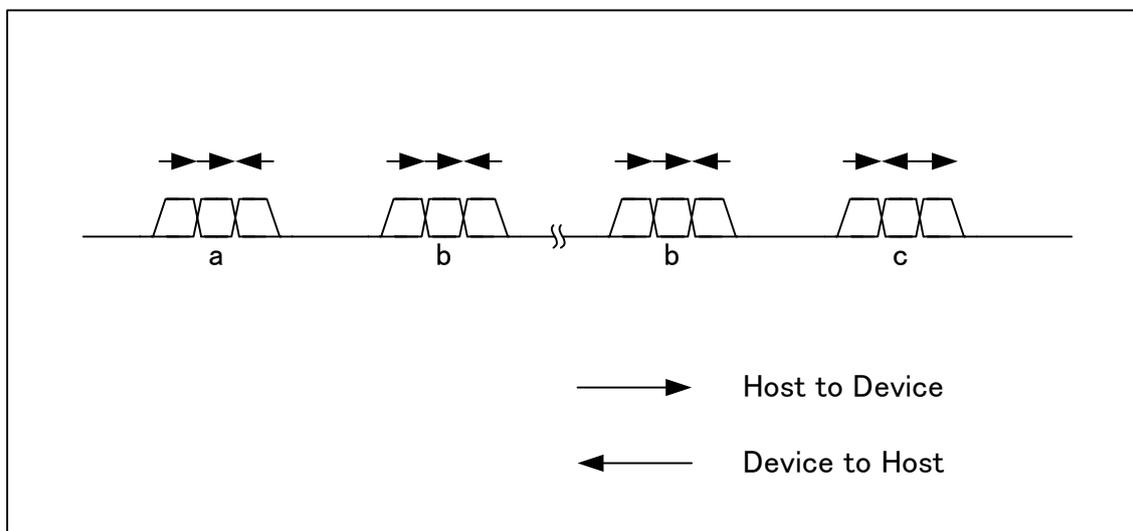


図 6-31 データステージが OUT 方向のコントロール転送

図 6-6 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。(b)ホストはINT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。

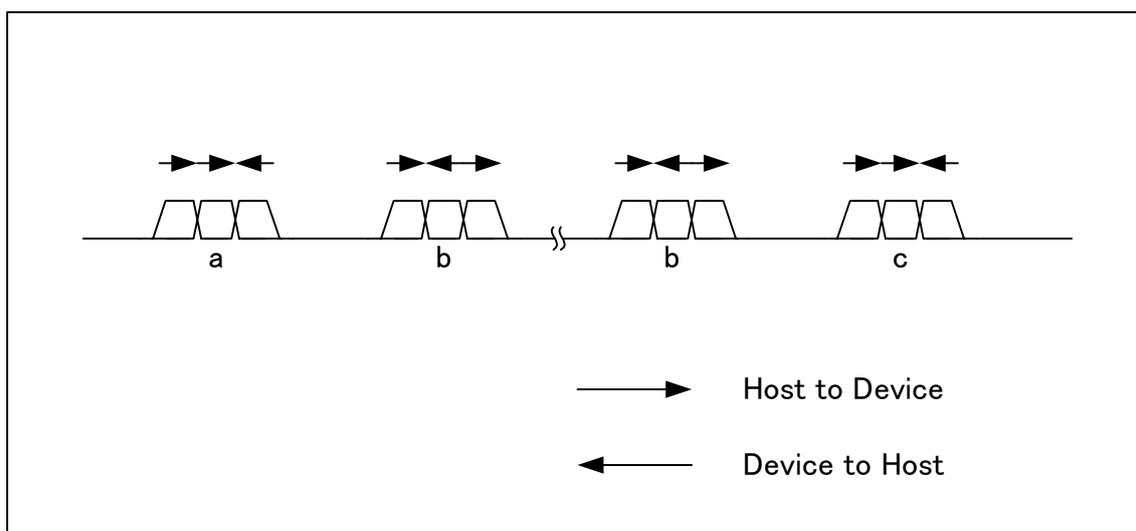


図 6-32 データステージが IN 方向のコントロール転送

6.3.4.1. 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。詳細は機能説明のセットアップトランザクションの項を参照して下さい。

6.3.4.2. 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 0 に設定してトランザクションを実行してください。

6.3.4.3. 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。

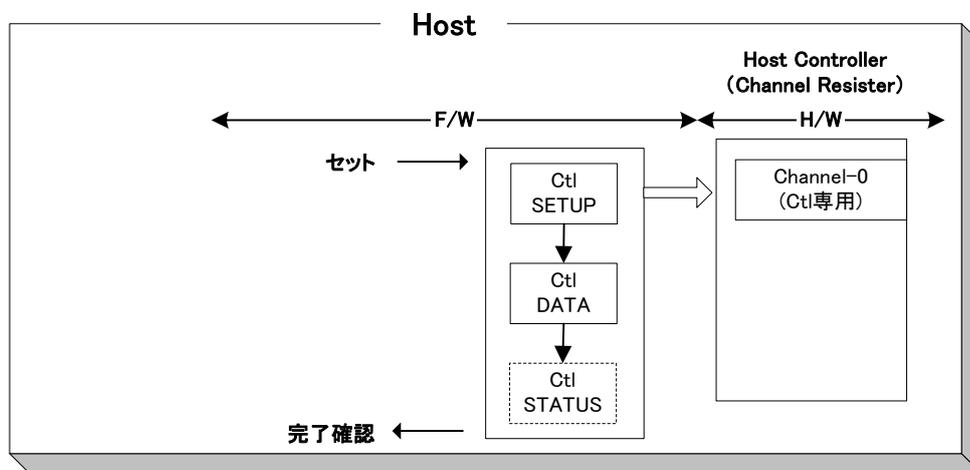


図 6-33 コントロール転送サポート機能の制御

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のようになります。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 (H_CH0StartAdrs,H_CH0EndAdrs)
- (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 FIFO の CH0 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 FIFO の CH0 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、FIFO のチャンネル CH0 エリアにあるデータを 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)。
 - ・ 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 FIFO のチャンネル CH0 エリアに書き込みます (IN 方向のデータステージ)。
 - ・ データステージは SETUP データの wLength で示されるデータ数に対するトランザクションを行います。
 - ・ データステージが IN 方向の場合は、ショートパケットを受信すると受信したデータのデータサイズが SETUP データの wLength で指定されたサイズ未満であった場合も IN トランザクションの発行を停止します。
 - ・ 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) FIFO の CH0 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、FIFO 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、FIFO 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。
 - ・ データステージが OUT 方向だった場合、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)。
 - ・ データステージが IN 方向だった場合、SETUP データの wLength で示される分のデータを受信するかショートパケットを受信した為にトランザクションが発行されなくなった状態で、FIFO に存在する受信データが全て読み出され 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_CH0ConnditionCode)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。

コントロール転送が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、機能説明の各トランザクションの項を参照して下さい。

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 6-25 に示します。

表 6-25 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、機能説明のコントロール転送サポート 機能の項を参照して下さい。
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。

6.3.5. バルク転送／インタラプト転送

CHaにおけるバルク転送、CHb, CHc, CHd, CHeにおけるバルク転送及びインタラプト転送はデータフロー(6.3.6 参照)としても、連続する個々のトランザクション(6.3.3 参照)としても制御できます。

6.3.6. 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。

6.3.6.1. OUT 転送

H_CH0TotalSize_H,L または 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、各チャンネルの FIFO 上書き込んで下さい。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、及び IDE からのリード転送の方法があります。

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、H_CHx{x=0,a-e}Join.JoinCPU_Wr ビットにより、ただ一つのチャンネルを選択して下さい。選択したチャンネルの FIFO には、FIFO_Wr 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_Remain_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 H_Remain_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、CHx{x=0,a-e}Join.JoinDMAx{x=0,1} ビットにより、DMA のチャンネル毎にただ一つのチャンネルを選択し、DMAx{x=0,1}_Control.Dir ビットに 0 を設定して下さい。選択したチャンネルの FIFO 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。

IDE インタフェイスのリード転送により、FIFO へデータを書き込むには、CHx{x=a-e}Join.JoinIDE ビットにより、ただ一つのチャンネルを選択し、IDE_Control.Dir ビットに 0 を設定して下さい。選択したチャンネルの FIFO は、IDE_Control.IDE_Go による IDE 転送を実行することにより、IDE からリードした順に書き込まれ、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、IDE インタフェイスは自動的にリード転送を一時停止して、フロー制御を行います。

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、H_CH0TotalSize_H,L または H_CHx{x=a-e}TotalSize_HH,HL,LH,LL と H_CHx{x=0,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。

6.3.6.2. IN 転送

H_CH0TotalSize_H,L または 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 H_CH0TotalSize_H,L または H_CHx{x=a-e}TotalSize_HH,HL,LH,LL と H_CHx{x=0,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有れば、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、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=0,a-e}ConnditionCode 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。

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、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}ConnditionCode)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。

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 FIFO 上書き込まれます。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、及び IDE へのライト転送の方法があります。

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、H_CHx{x=0,a-e}Join.JoinCPU_Rd ビットにより、ただ一

つのチャンネルを選択して下さい。選択したチャンネルの FIFO 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できます。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、H_CHx{x=0,a-e}Join.JoinDMAx{x=0,1} ビットにより、DMA のチャンネル毎にただ一つのチャンネルを選択し、DMAx{x=0,1}_Control.Dir ビットに 1 を設定して下さい。選択したチャンネルの FIFO 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、DMAx{x=0,1}_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。

IDE インタフェイスのライト転送により、FIFO のデータを読み出すには、CHx{x=a-e}Join.IDE ビットにより、ただ一つのチャンネルを選択し IDE_Control.Dir ビットに 1 を設定して下さい。選択したチャンネルの FIFO は、IDE_Control.IDE_Go による IDE 転送を実行することにより、受信順に読み出されます。FIFO が空になると、IDE インタフェイスは自動的にライト転送を一時停止して、フロー制御を行います。

6.3.7. 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 6-26 に示します。

表 6-26 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。

6.3.7.1. 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。

6.3.8. バルクオンリーサポート機能

本 LSI には、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、ファームウェアは各トランスポートの制御を個別に行う必要がなくなります。バルクオンリーサポート機能を用いた場合の制御の例、使用せずに各トランスポートを個別のトランザクションとして制御した場合の例を以下に図示します。

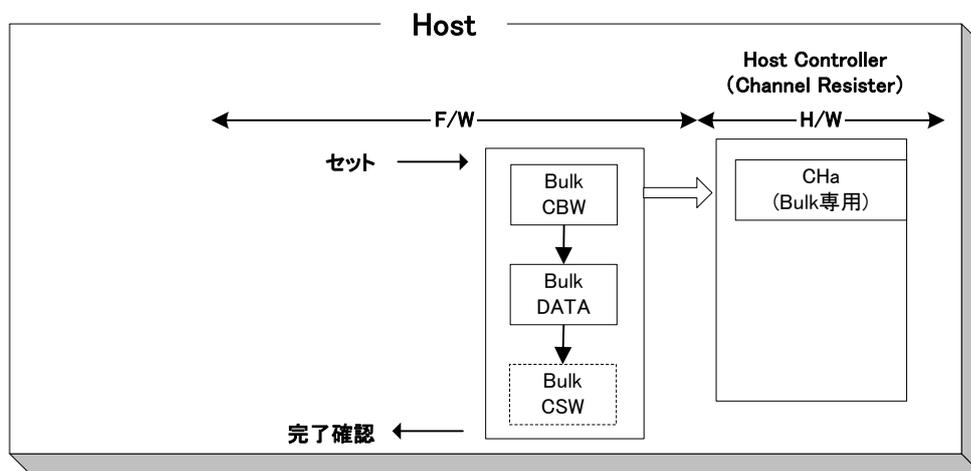


図 6-34 バルクオンリーサポート機能の制御

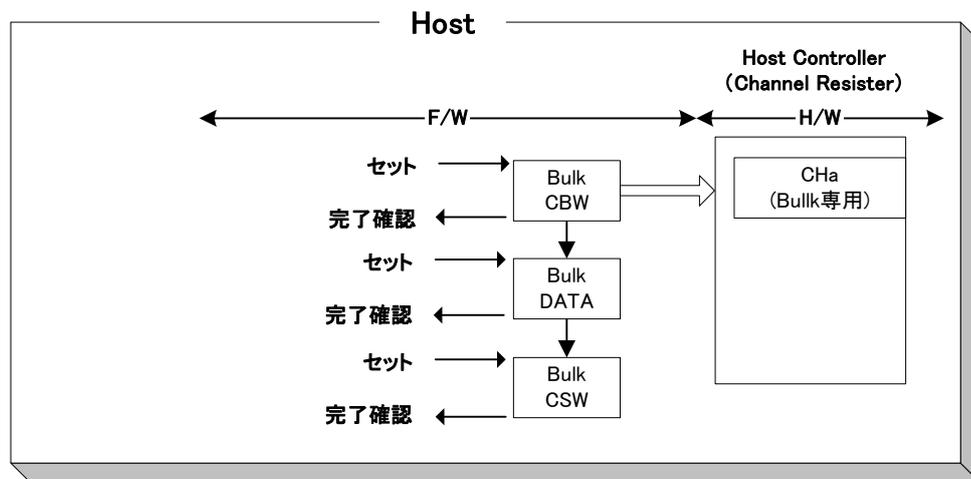


図 6-35 バルクオンリーサポート機能を使用しない場合の制御(参考)

バルクオンリーサポート機能はチャンネル CHa にて有効です。この機能を使用した場合のトランスポートの処理は以下のようになります。ファームウェアは(1)～(5)の処理を行います。

- (1) チャンネル CHa 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_CHaConfig_0.SpeedMode)、マックスパケットサイズ (H_CHaMaxPktSize)、USB アドレス (H_CHaFuncAdrs.FuncAdrs)、FIFO 領域 (H_CHaStartAdrs, CHaEndAdrs)
- (2) バルクオンリーサポート機能の以下の制御レジスタに適宜設定を行います。
OUT エンドポイントトグルシーケンス (H_OUT_EP_Control.OUT_Toggle)、OUT エンドポイントナンバー (H_OUT_EP_Control.OUT_EP_Number)、IN エンドポイントトグルシーケンス (H_IN_EP_Control.OUT_Toggle)、IN エンドポイントナンバー (H_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) FIFO の CHa エリアに DMA、または IDE の設定を行います。

- (5) バルクオンリーサポート実行(H_BO_SupportControl.BO_SupportGo)をセットします。
この時、トランスポート状態(H_BO_SupportControl.BO_TransportState)の値は Idle (00b)として H_BO_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_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、FIFO のチャネル CHa エリアにあるデータを、OUT エンドポイントナンバー(H_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)。
 - 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_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 のチャネル CHa エリアに書き込みます(IN 方向のデータトランスポート)。
 - 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に対してトランザクションを行います。
 - データトランスポートが IN 方向の場合は、ショートパケットを受信すると受信したデータのデータサイズが CBW データの dCBWDataTransferLength で指定されたサイズ未満であった場合も IN トランザクションの発行を終了します。
 - 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) IN エンドポイントナンバー(H_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_CS_W_RcvDataSize.CSW_RcvDataSize)に反映します。
- データトランスポートが OUT 方向の場合、CBW データの dCBWDataTransferLength で示されるデータ数のトランザクションが終了すると、ステータストランスポートを実行する状態になります。
 - データトランスポートが IN 方向の場合、CBW データの dCBWDataTransferLength で示されるデータ数のトランザクションが終了するかショートパケットを受信した為に IN トランザクションが発行されなくなった状態で、FIFO に存在する受信データが全て読み出され 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。
- (9) 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_BO_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出来ます。
- (10) 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_BO_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。
- (11) 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_BO_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_BO_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_CHaConnditionCode)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_BO_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。

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_BO_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_BO_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_BO_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_BO_SupportControl.BO_TransportState)にIdle(00b)を設定し、バルクオンリーサポート実行(H_BO_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、機能説明の各トランザクションの項を参照して下さい。

FIFOのCBWエリア、CSWエリアについては、機能説明のFIFOの項を参照して下さい。

DMAについては、機能説明のDMAの項を参照して下さい。

IDEについては、機能説明のIDEの項を参照して下さい。

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 6-27 に示します。

表 6-27 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_BO_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、機能説明のバルクオンリーサポート機能の項を参照して下さい。
OUT エンドポイントトグルシーケンス	H_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_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_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_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_BO_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。

6.3.9. ホストステート管理サポート機能

6.3.9.1. 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 6-36 示します。

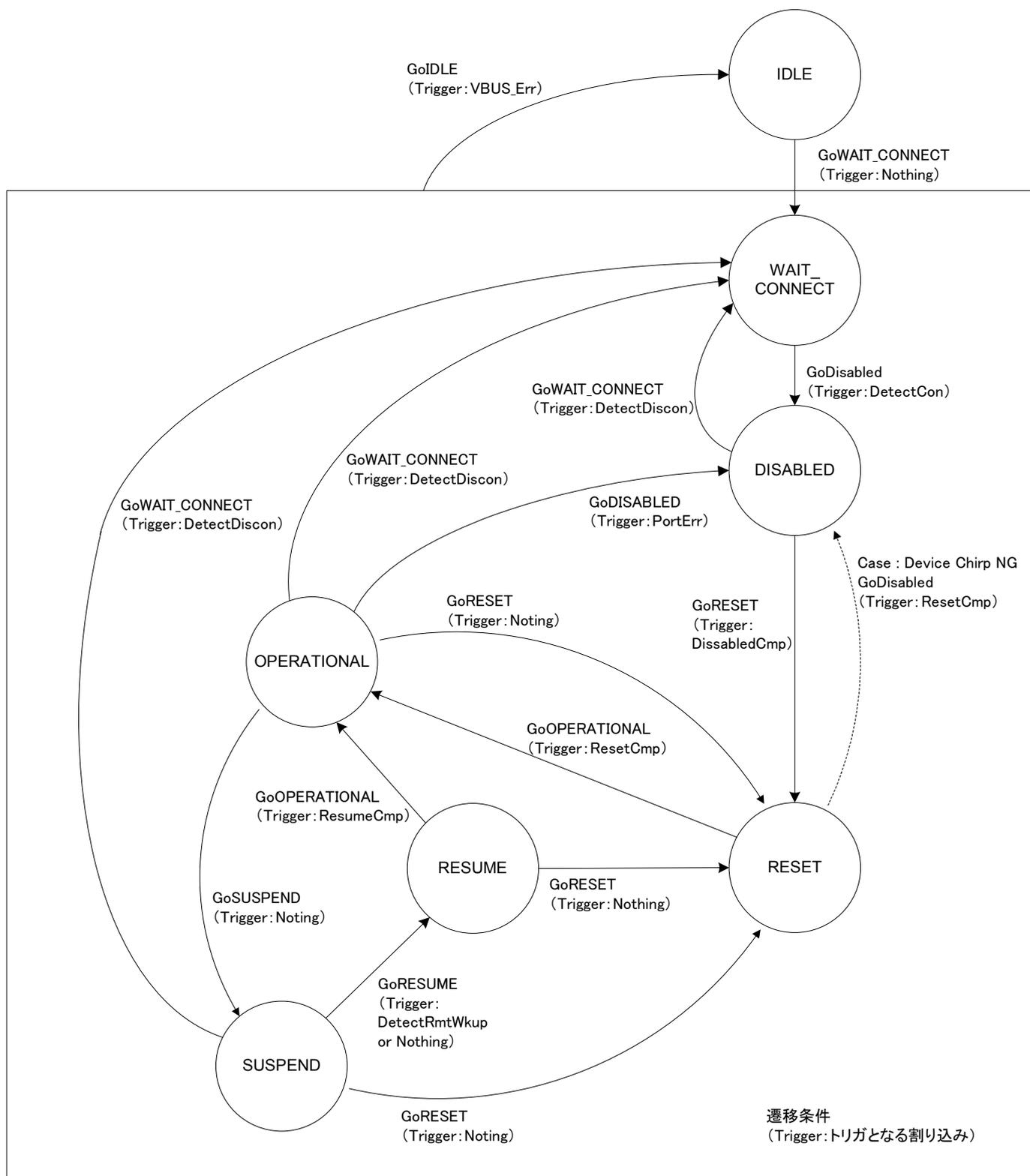


図 6-36 ホストステート遷移図

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 6-28 に示します。

表 6-28 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。
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。

6.3.9.1.1. IDLE

USB ホスト機能を初期化するステートで、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・ウェイクアップ検出、デバイスチャープ検出の全検出機能をオフする。

6.3.9.1.2. 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により、OPERATIONAL および RESET の各ステートでデバイスの切断を検出した時は、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・ウェイクアップ検出、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。

6.3.9.1.3. 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、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。
- 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。

6.3.9.1.4. 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。
- ・ 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 upstream port での "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。詳細は異常なデバイスのチャープ検出項 (6.3.9.3.4.2) 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。

6.3.9.1.5. 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。

(注) RESET から OPERATIONAL、または RESUME から OPERATIONAL への遷移を正しく行うためには、最初の uSOF (HS)、SOF (FS)、KeepAlive (LS) が発行されるまでの時間において、USB 2.0 規格の規定時間を遵守する必要があります。この規定時間を意識することなく遷移を行うためには、GoRESETtoOP、GoSUSPENDtoOP、GoRESUMEtoOP の連続遷移用コマンドを使用することを推奨します。規定時間については、6.3.9.3.4.1 の T11、6.3.9.3.4.3 の T5、6.3.9.3.4.4 の T5、6.3.9.3.7.1 の T5、6.3.9.3.7.2 の T4、6.3.9.3.7.3 の T4 を参照してください。

6.3.9.1.6. 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 PpowerDown 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。

6.3.9.1.7. 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。

6.3.9.2. 検出機能

6.3.9.2.1. 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(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タス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タス移行実行(H_NegoControl_0.AutoMode)にGoIDLEを設定)してください。それによりIDLEステータスに遷移し、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。

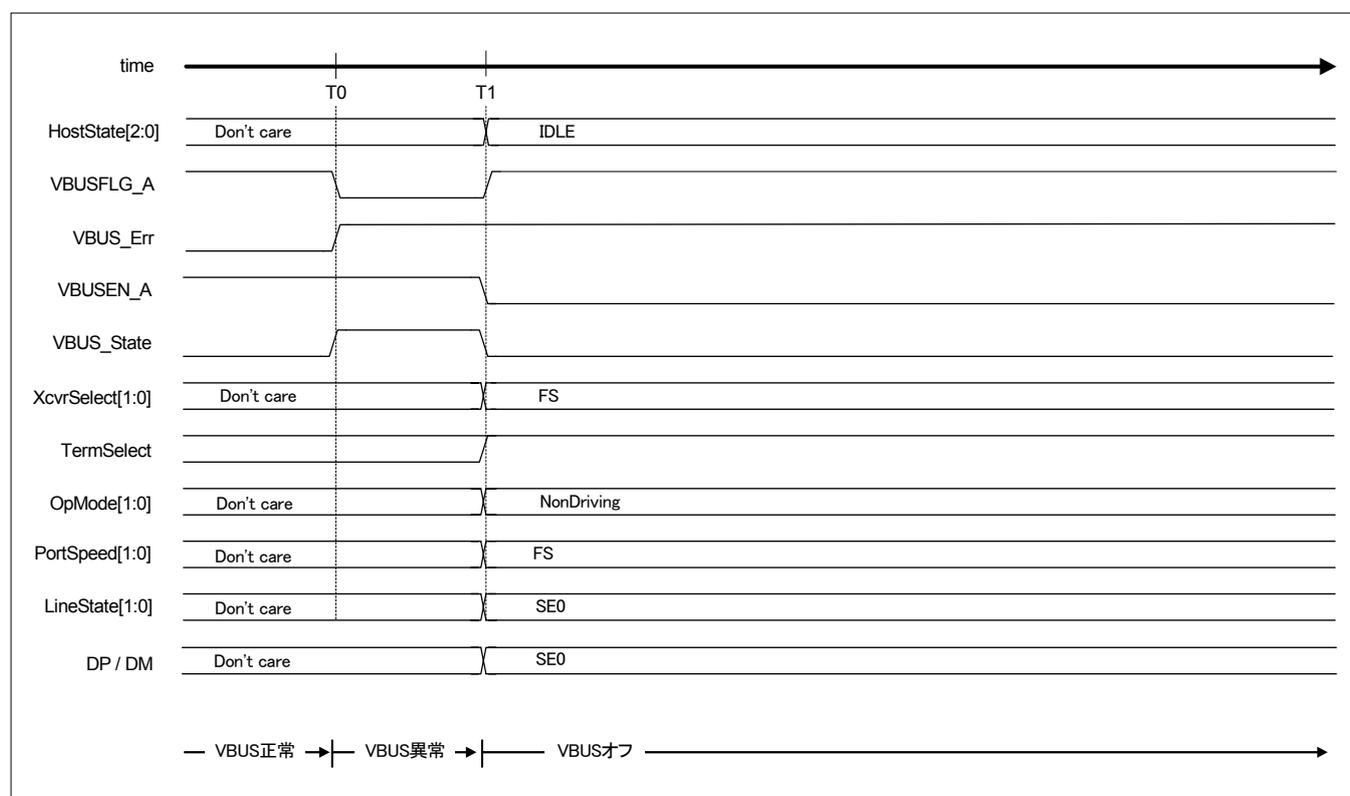


図 6-37 VBUS 異常検出 Timing

表 6-29 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(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ステータスに遷移させる。(F/W)	T1

6.3.9.2.2. 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。

6.3.9.2.2.1. 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 回連続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)。

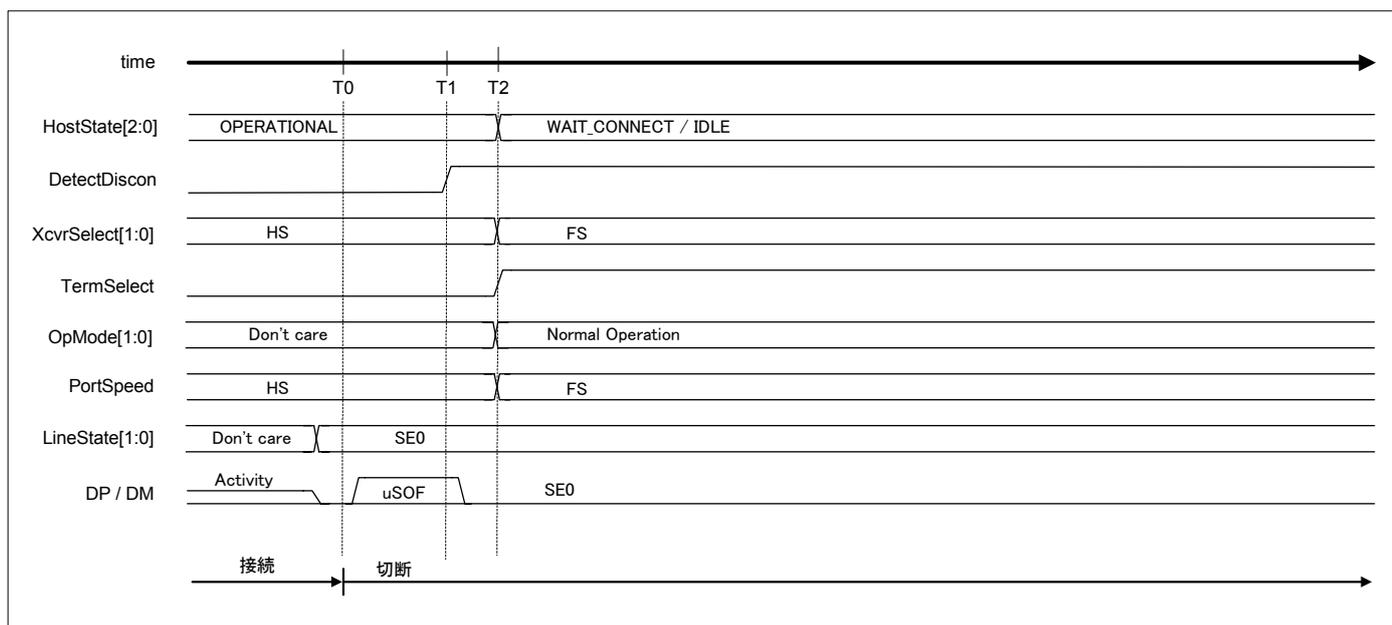


図 6-38 切断検出 Timing (HS mode)

表 6-30 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。(FW)	T2

6.3.9.2.2. 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)。

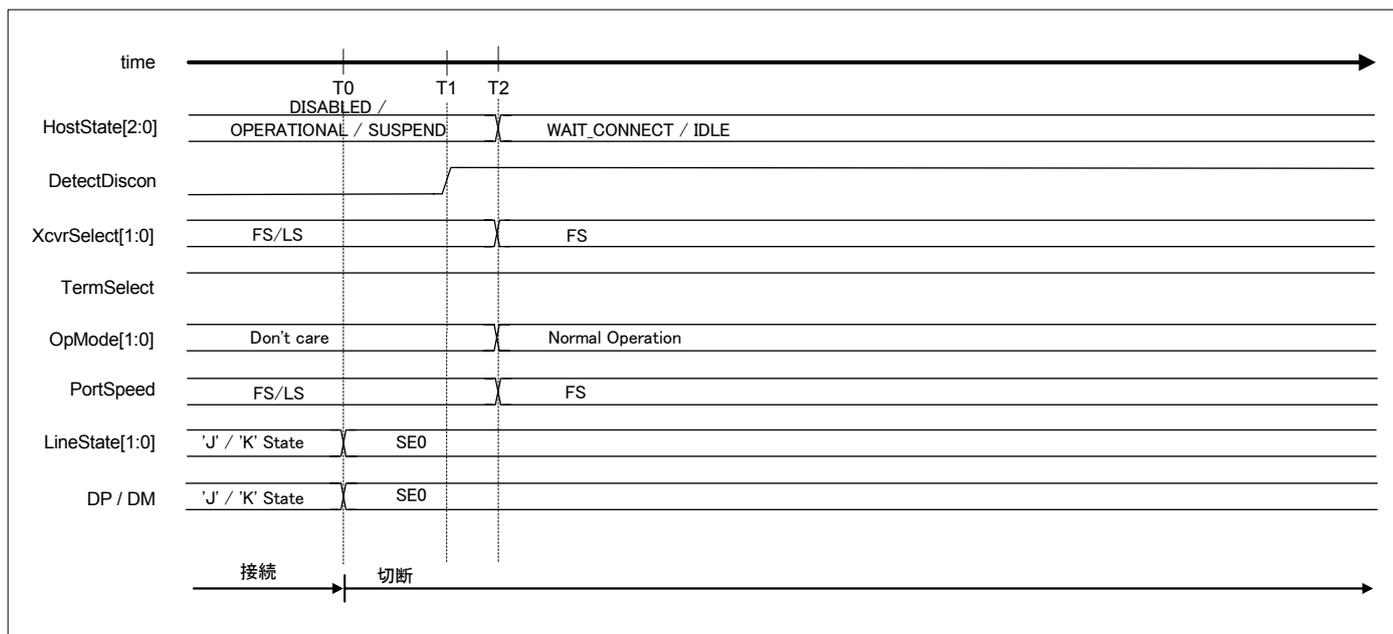


図 6-39 切断検出 Timing (FS または LS mode)

表 6-31 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。(FW)	規定なし

6.3.9.2.3. 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。

6.3.9.2.3.1. 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。

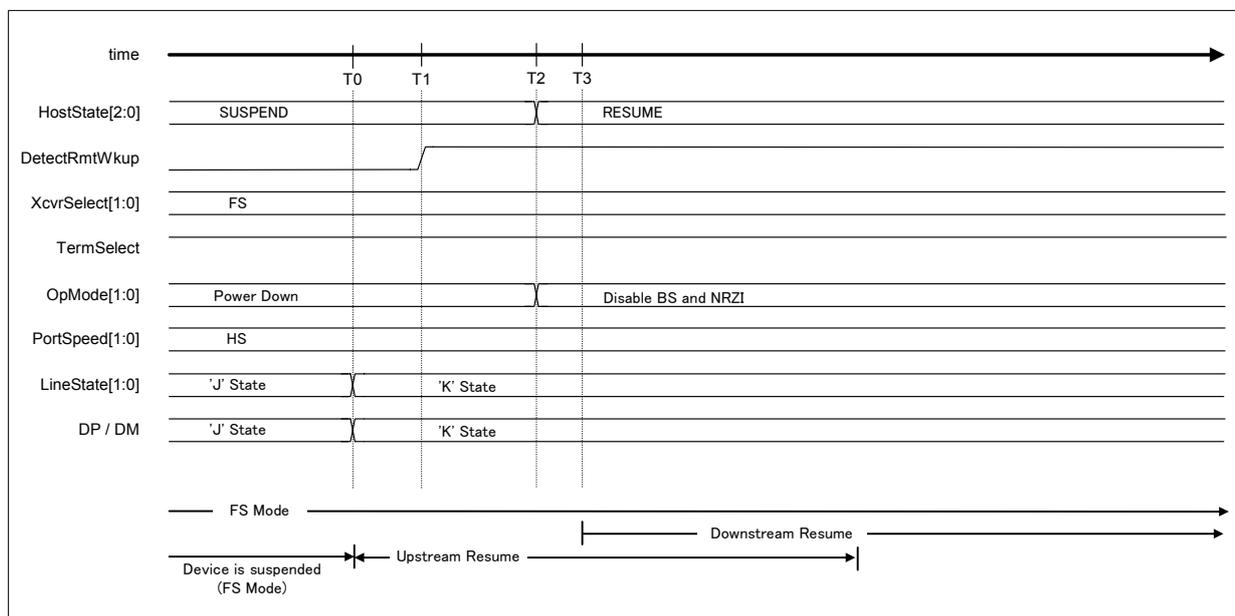


図 6-40 Remote Wakeup Timing (HS mode)

表 6-32 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。

6.3.9.2.3.2. 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、HS デバイスが接続されている場合を参照してください。

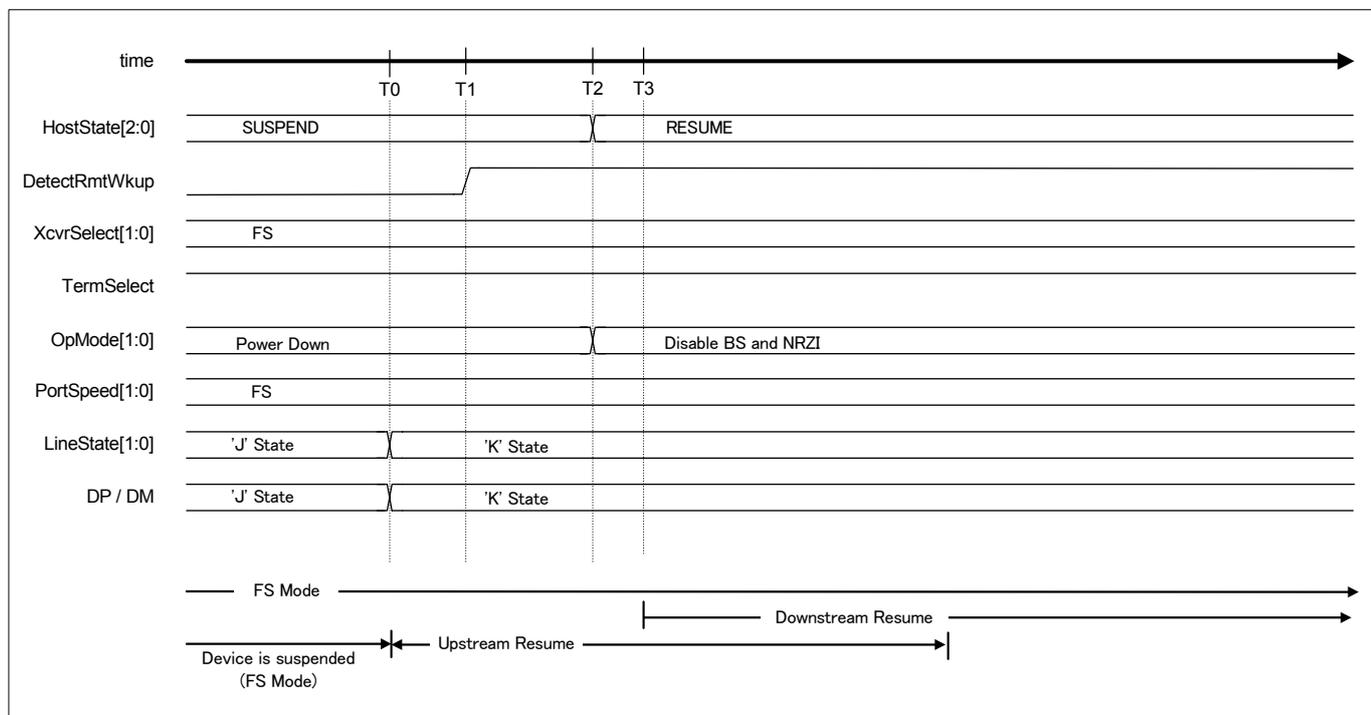


図 6-41 Remote Wakeup Timing (FS mode)

表 6-33 Remote Wakeup Timing Values (FS mode)

Timing Parameter	Description	Value
T0	デバイスがリモート・ウェイクアップ信号(K)を送出開始する。	0 (reference)
T1	リモート・ウェイクアップ信号(K)を検出する。 リモート・ウェイクアップ検出ステータスを発行する。(HW)	$T0 + 2.5\mu s < T1 \{T_{URLK}\}$
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。(FW)	$T2 < T1 + 900\mu s$
T3(参考)	ホストがレジューム信号(K)の発行を開始する。(HW)	$T3 < T0 + 1ms \{T_{URSM}\}$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.2.3.3. 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、HS デバイスが接続されている場合を参照してください。

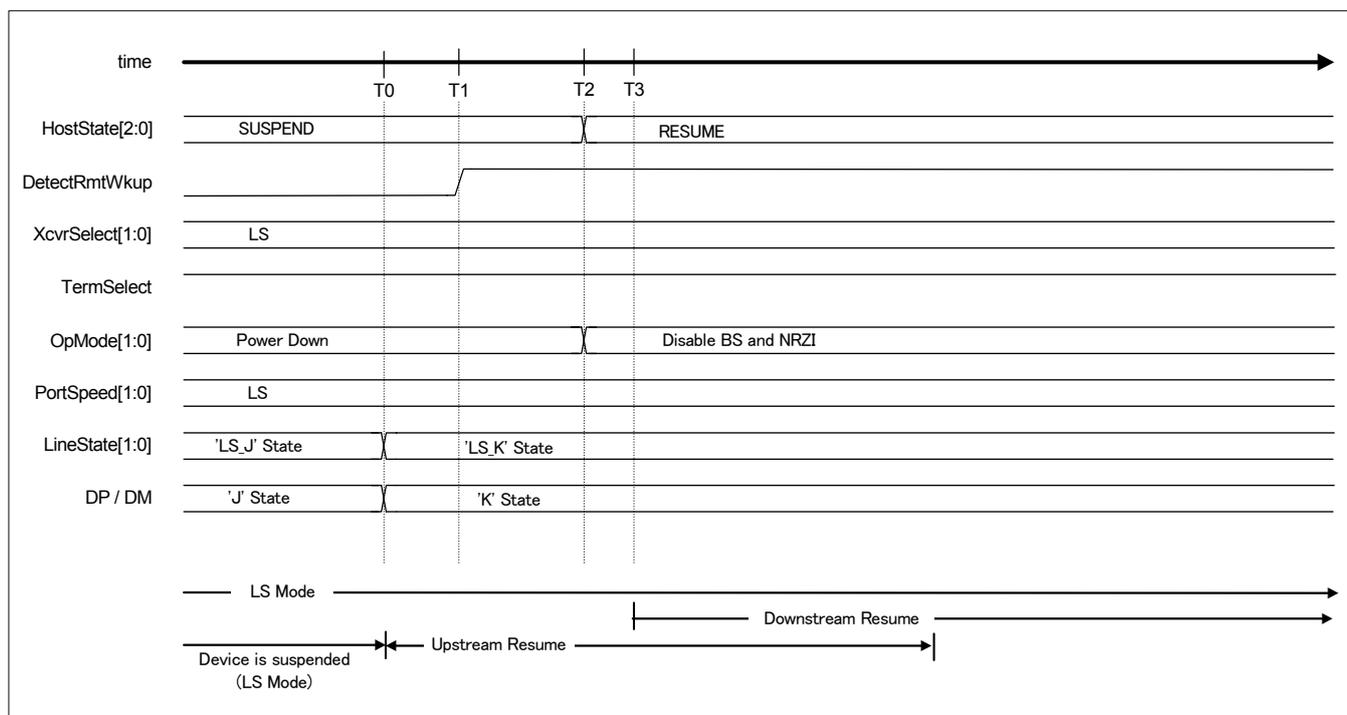


図 6-42 Remote Wakeup Timing (LS mode)

表 6-34 Remote Wakeup Timing Values (LS mode)

Timing Parameter	Description	Value
T0	デバイスがリモート・ウェイクアップ信号(K)を送出開始する。	0 (reference)
T1	リモート・ウェイクアップ信号(K)を検出する。 リモート・ウェイ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。

6.3.9.2.4. 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。

6.3.9.2.4.1. 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)。

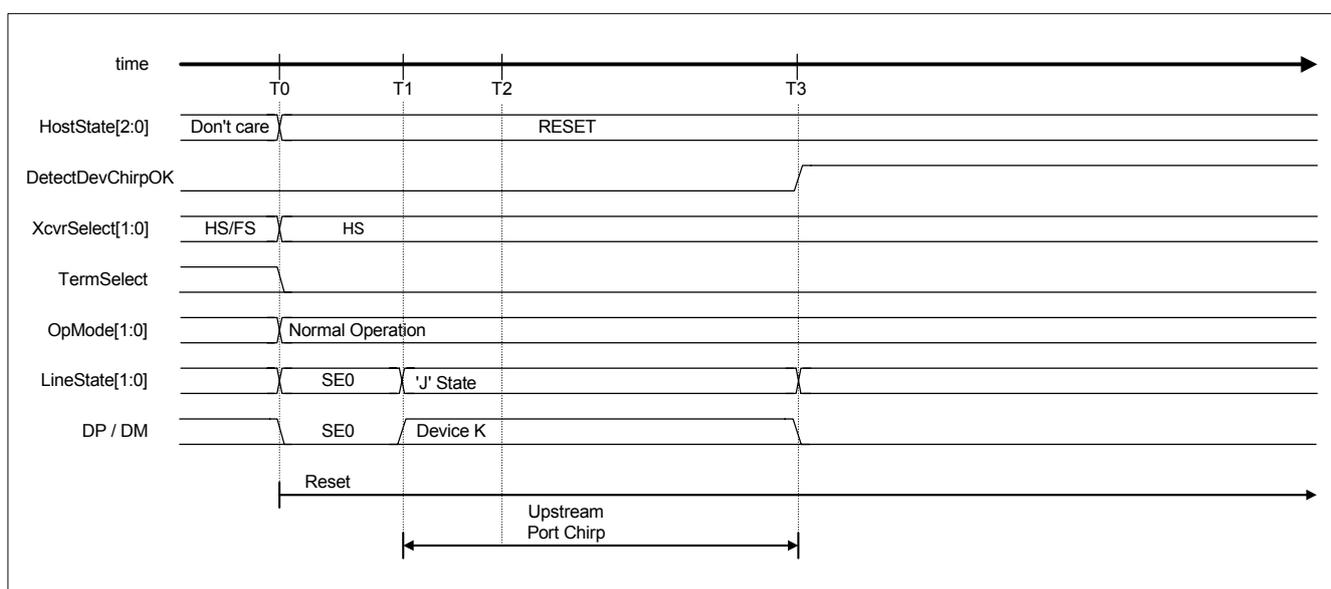


図 6-43 Device Chirp Timing

表 6-35 Device Chirp Timing Values

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に GoRESET を設定する。(FW)	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。(HW)	$T1 + 2.5\mu\text{s} \{T_{\text{FILT}}\} < T2$
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。(HW)	$T1 + 1.0\text{ms} \{T_{\text{UCH}}\} < T3 < T0 + 7.0\text{ms} \{T_{\text{UCHEND}}\}$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.2.4.2. 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)。

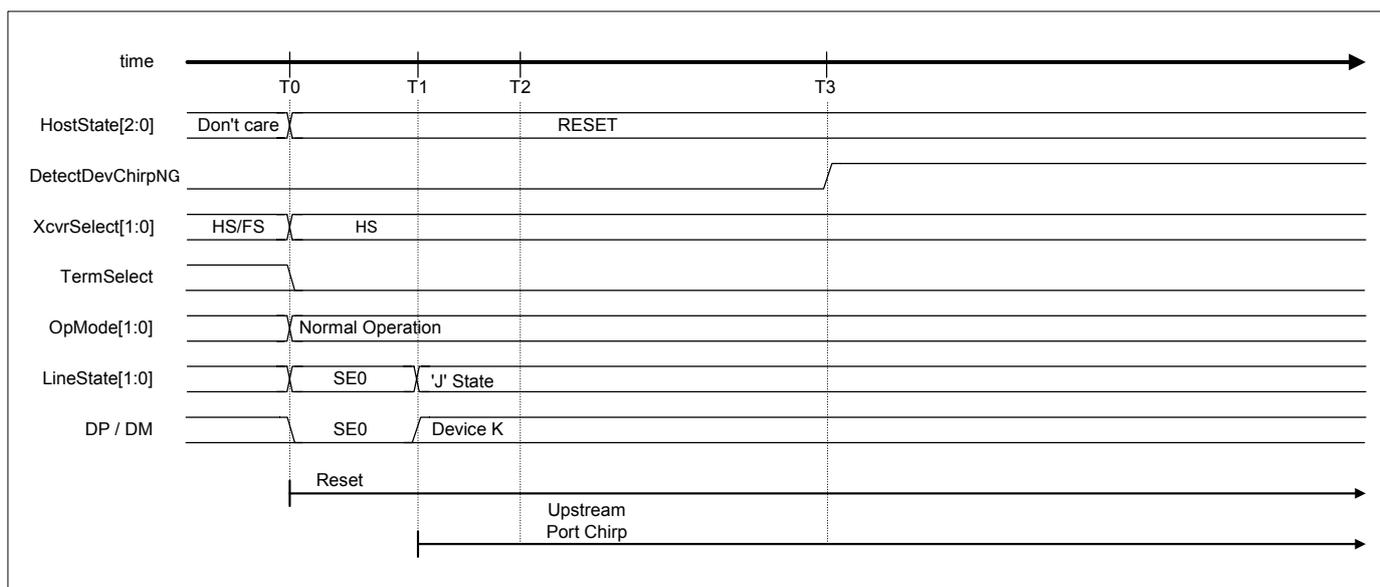


図 6-44 Device Chirp Timing(NG)

表 6-36 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。

6.3.9.2.5. 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) H_NegoControl_0.ResetHTM に 1 を設定し、ホス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、H_NegoControl_0.ResetHTM に 0 を設定し、ホストランシーバマクロのリセットを解除します。

6.3.9.3. ホストステート管理サポート機能の単体説明

6.3.9.3.1. 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)。

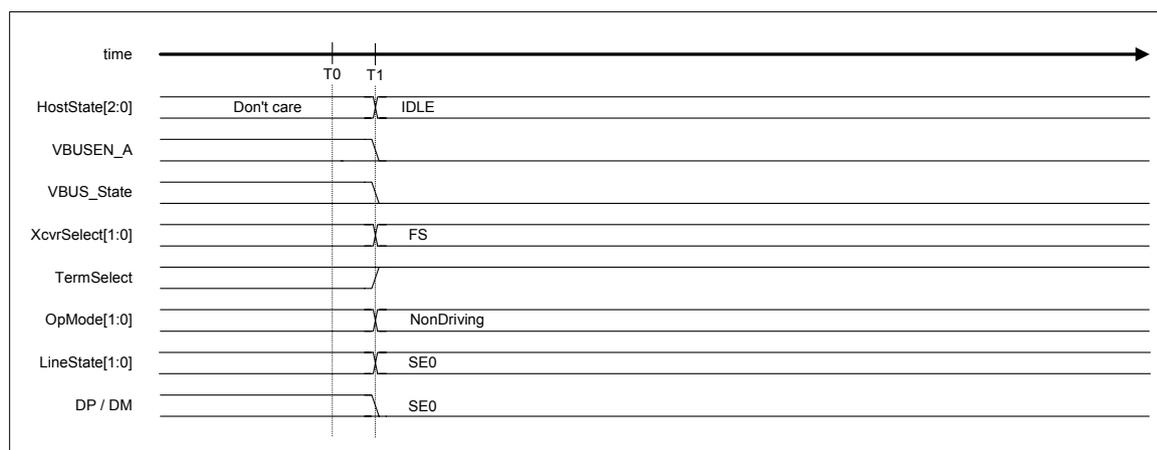


図 6-45 GoIDLE Timing

表 6-37 GoIDLE Timing Value

Timing Parameter	Description	Value
T0	実行中ステートの動作を停止する。(FW)	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。(FW) 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"NonDriving"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、デバイスチャープ検出をオフする。(HW)	$T0 + 5\text{cycle}(60\text{MHz}) < T1$

6.3.9.3.2. GoWAIT_CONNECT

ホスト状態遷移実行(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。

6.3.9.3.2.1. 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状態遷移実行(H_NegoControl_0.AutoMode)に GoWAIT_CONNECT を設定します(T0)。
- (2) ホスト状態モニタ(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)。

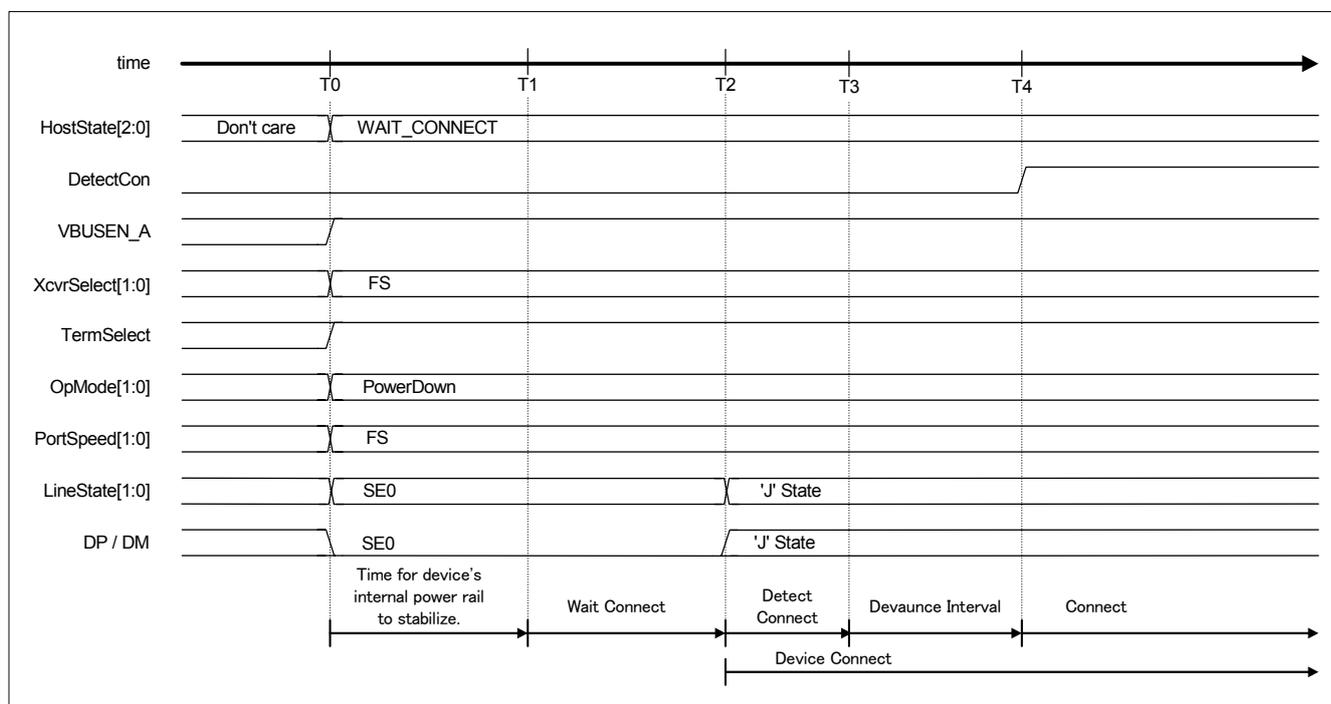


図 6-46 Device Attach Timing(FS mode)

表 6-38 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)	$T2 + 2.5\mu\text{s}\{T_{\text{DCNN}}\} < T3$
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)	$T3 + 100\text{ms}\{T_{\text{ATTDB}}\} < T4$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.2.2. 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_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_NegoControl1.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_XcvrControl.XcvrSelect[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_NegoControl1.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(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_XcvrControl.XcvrSelect)とポートスピード(H_NegoControl1.PortSpeed[1:0])をととも 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)。

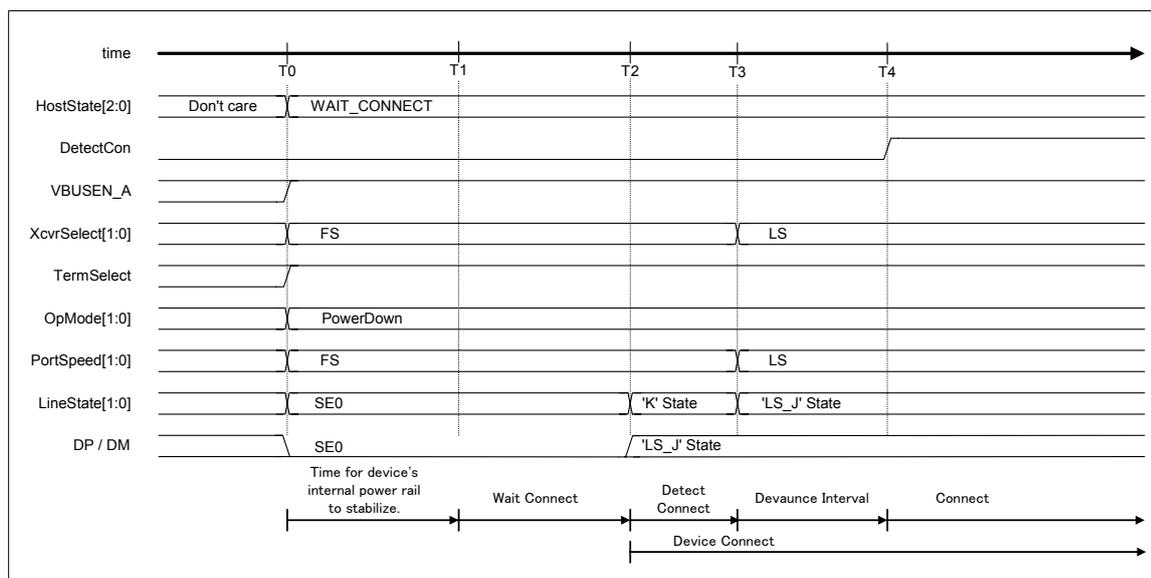


図 6-47 Device Attach Timing(LS mode)

表 6-39 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)	$T3 + 100\text{ms}\{T_{\text{ATTDB}}\} < T4$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.3. 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。

6.3.9.3.3.1. 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 Go DISABLED 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、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)。

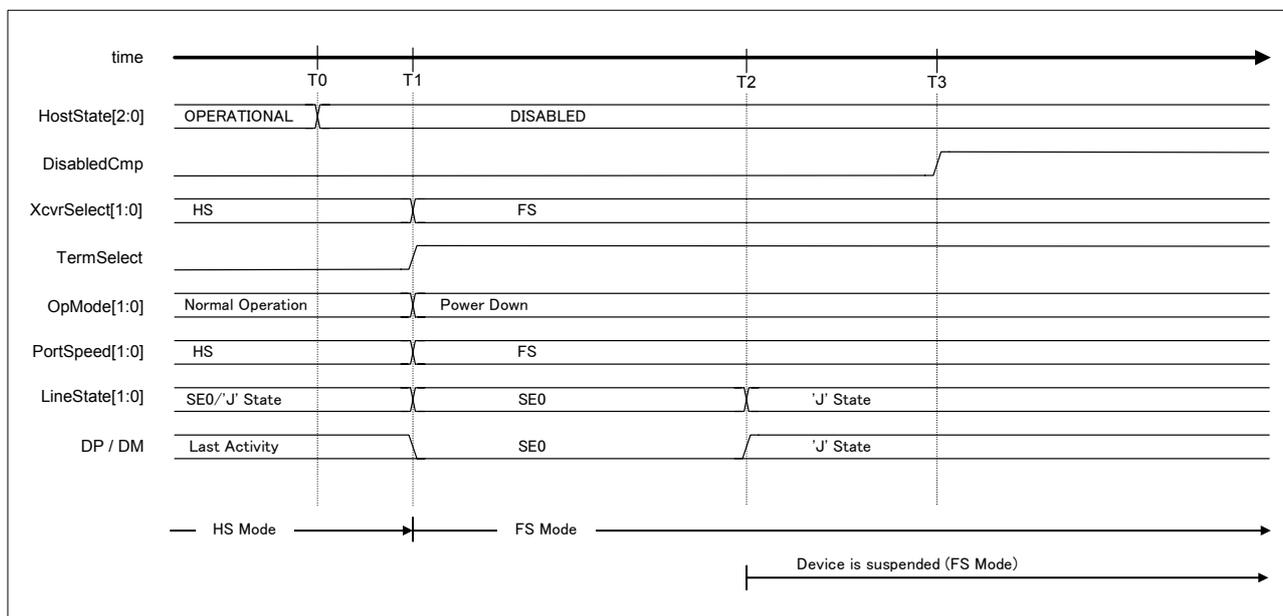


図 6-48 Disabled Timing (HS mode)

表 6-40 Disabled Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に Go DISABLED を設定する。(FW) 切断検出機能をオフする。(HW)	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、オペレーションモード (XcvrControl.OpMode[1:0]) を "PowerDown" に設定する。(HW)	T1
T2	デバイスがサスペンドを検出し、FS mode になる。	$T1 + 3.0ms < T2 \{T_{WTREV}\} < T1 + 3.125ms$
T3	切断検出機能をオンする。(HW) ディセーブルド遷移完了ステータスを発行する。(HW)	$T1 + 4ms < T3$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.3.2. 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_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)。

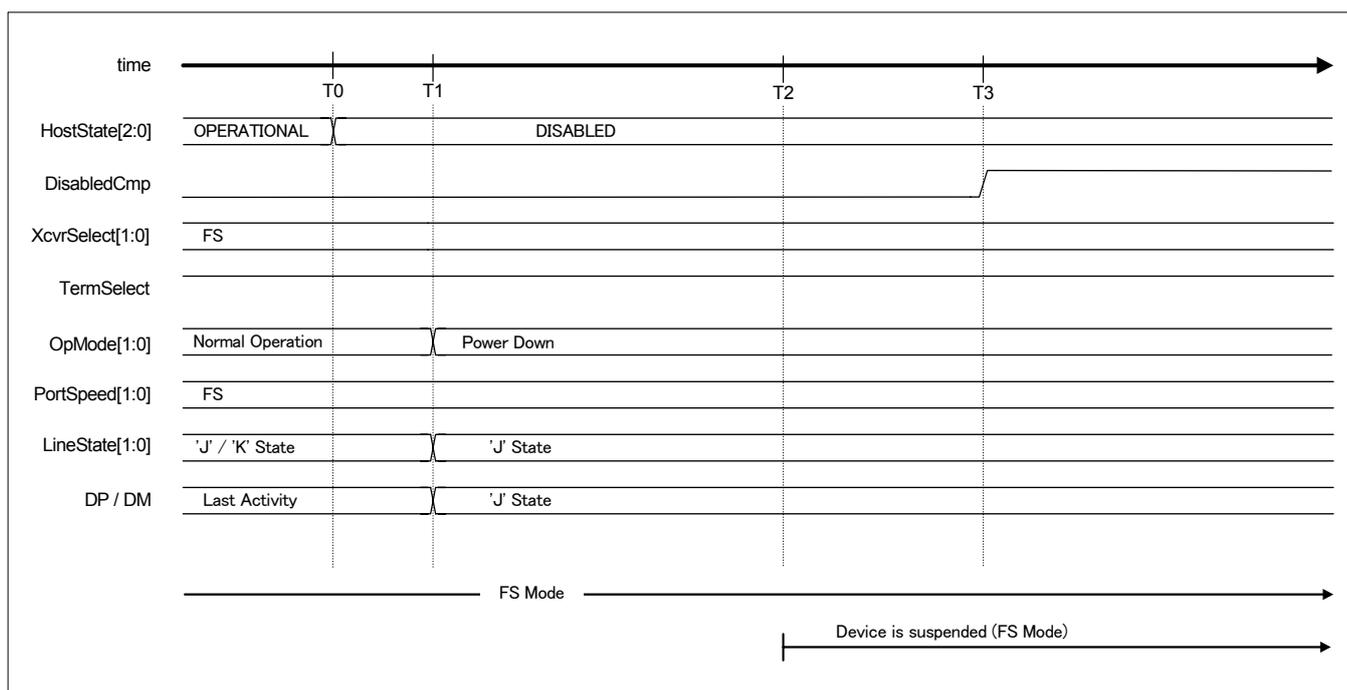


図 6-49 Disabled Timing (FS mode)

表 6-41 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_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する。	$T1 + 3.0\text{ms} < T2 \{T_{\text{WTREV}}\} < T1 + 3.125\text{ms}$
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。

6.3.9.3.3.3. 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_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)。

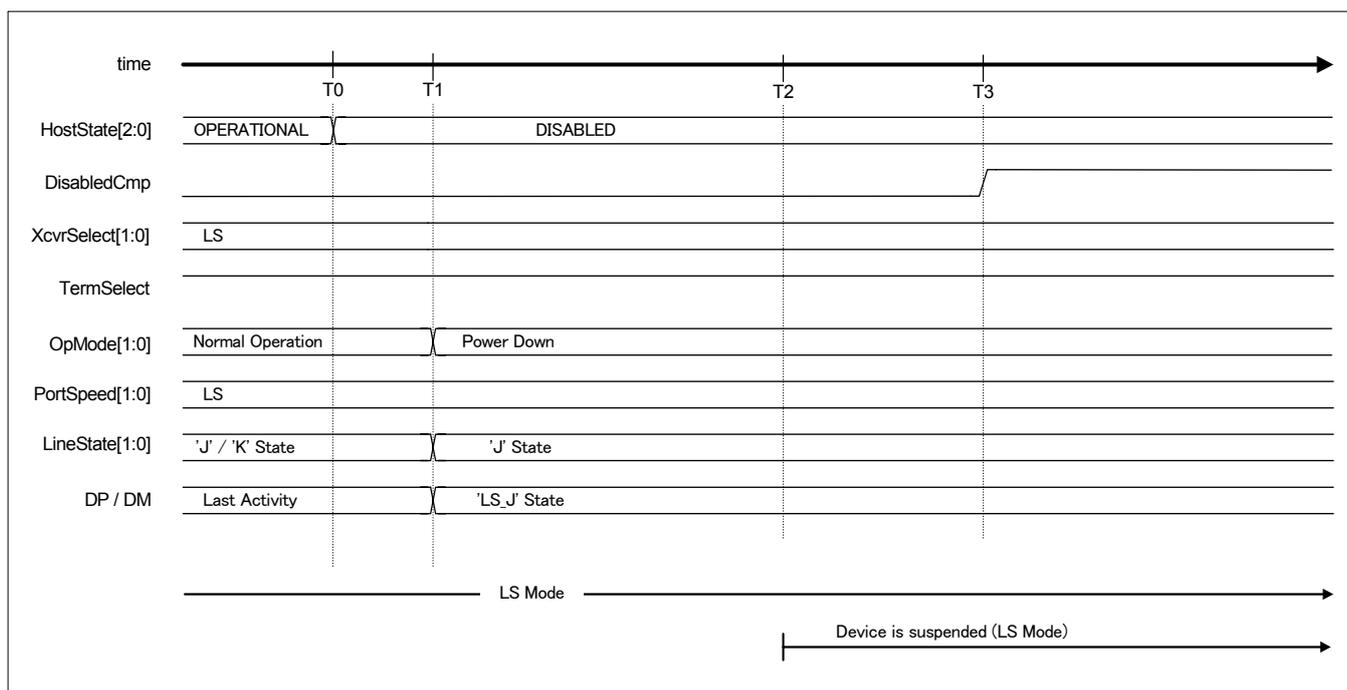


図 6-50 Disabled Timing (LS mode)

表 6-42 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_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する。	$T1 + 3.0\text{ms} < T2 \{T_{\text{WTREV}}\} < T1 + 3.125\text{ms}$
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。

6.3.9.3.4. 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。

6.3.9.3.4.1. 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(ジ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)。
- (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)。

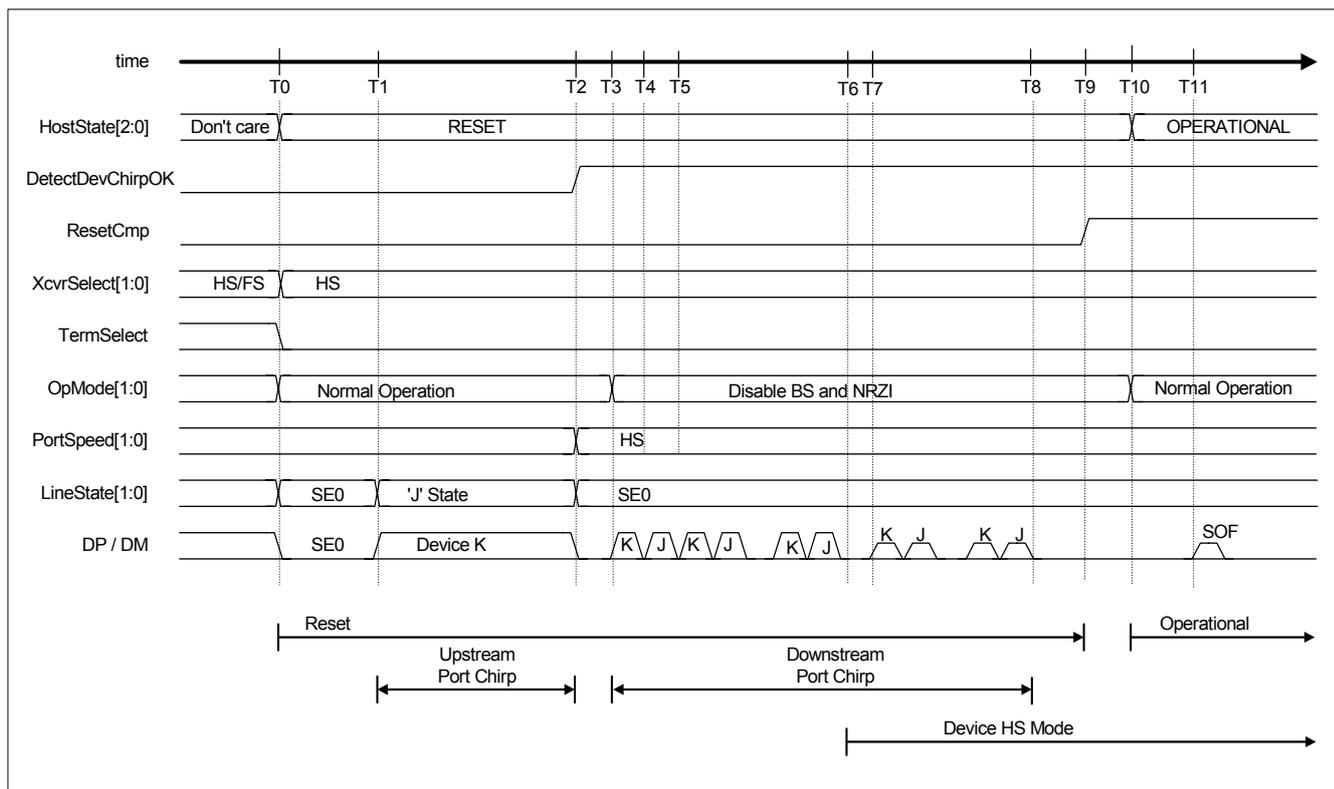


図 6-51 Reset Timing (HS mode)

表 6-43 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)	$T2 < T3 < T2 + 100\text{us} \{T_{WTDCH}\}$
T4	ホストが Chirp K から Chirp J に切り替えて出力する。(H/W)	$T3 + 40\text{us} \{T_{DCHBIT}\} < T4 < T3 + 60\text{us} \{T_{DCHBIT}\}$
T5	ホストが Chirp J から Chirp K に切り替えて出力する。(H/W)	$T4 + 40\text{us} \{T_{DCHBIT}\} < T5 < T4 + 60\text{us} \{T_{DCHBIT}\}$
T6	デバイスがホストの Chirp 検出する。	T6
T7	デバイスが HS モードに遷移。	$T6 < T7 < T6 + 500\text{us}$
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)	$T8 < T9 < T8 + 150\text{us}$
T10(参考)	H_NegoControl_0.AutoMode に GoOPERATIONAL を設定する。(F/W)	$T10 < T9 + 200\text{us}$
T11(参考)	最初の SOF が送出。(H/W)	$T10 + 120\text{us} < T11 < T10 + 130\text{us}$ $T8 + 100\text{us} \{T_{DCHSE0}\} < T11 < T8 + 500\text{us} \{T_{DCHSE0}\}$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.4.2. 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。

6.3.9.3.4.2.1. 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)。

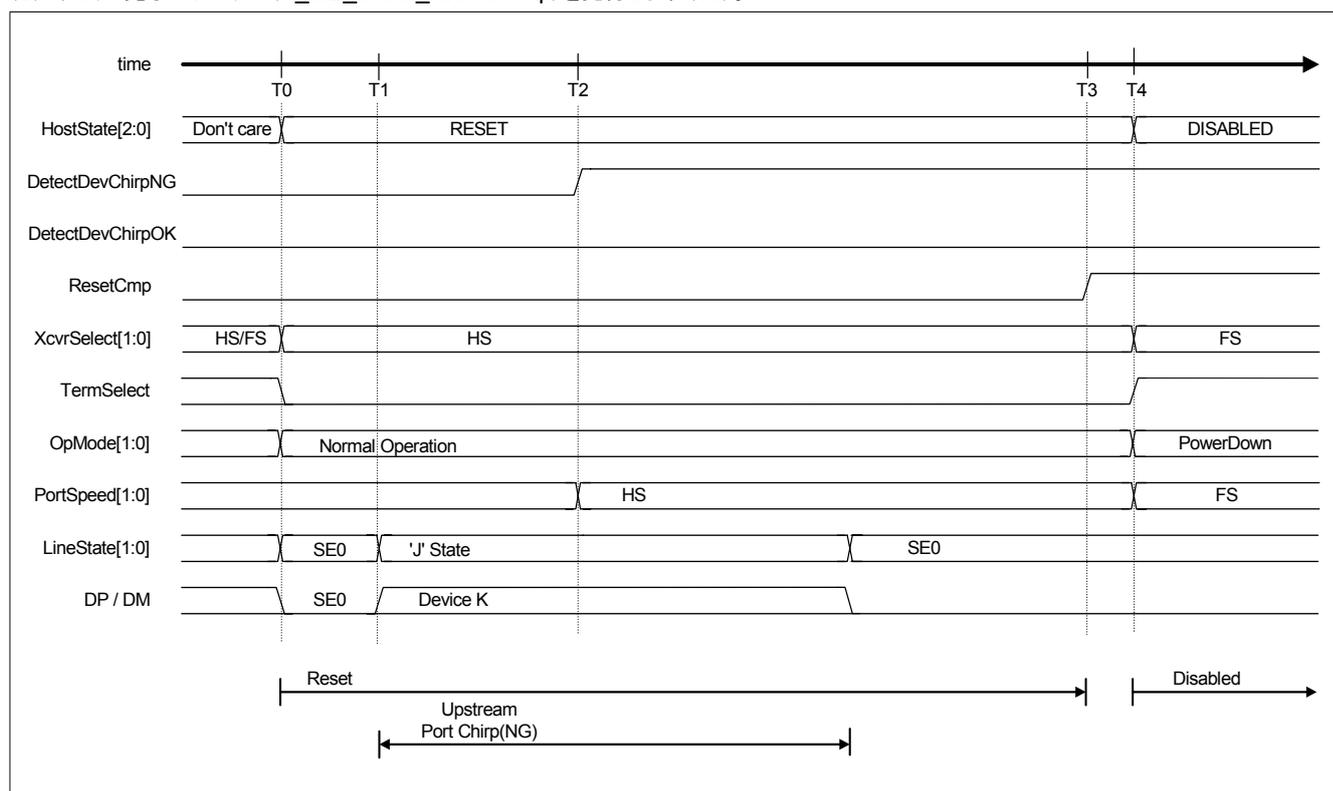


図 6-52 Detect Device Chirp NG Timing (チャープ完了ディセーブル設定 0)

表 6-44 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)	0 (reference)

	デバイスチャープ検出機能をオンする。(H/W)	
T1	デバイスが Chirp を開始。	$T_0 < T_1 < T_0 + 6.0\text{ms}$
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)	$T_0 + 7\text{ms}\{T_{\text{UCHEND}}\} < T_2$
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)	$T_2 + 50\text{ms}\{T_{\text{DRSTR}}\} < T_3$
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。

6.3.9.3.4.2.2. 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 完了を待つホストの 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 ステータス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タス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タス遷移実行(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タス遷移実行(H_NegoControl_0.AutoMode)に GoRESET を設定します(T0)。
- (2) ホストステータスモニタ(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) ラインステータス(USB_Host_Status.LineState[1:0])にアクティビティ有('J'ステータスとして見える)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(ラインステータス(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)。

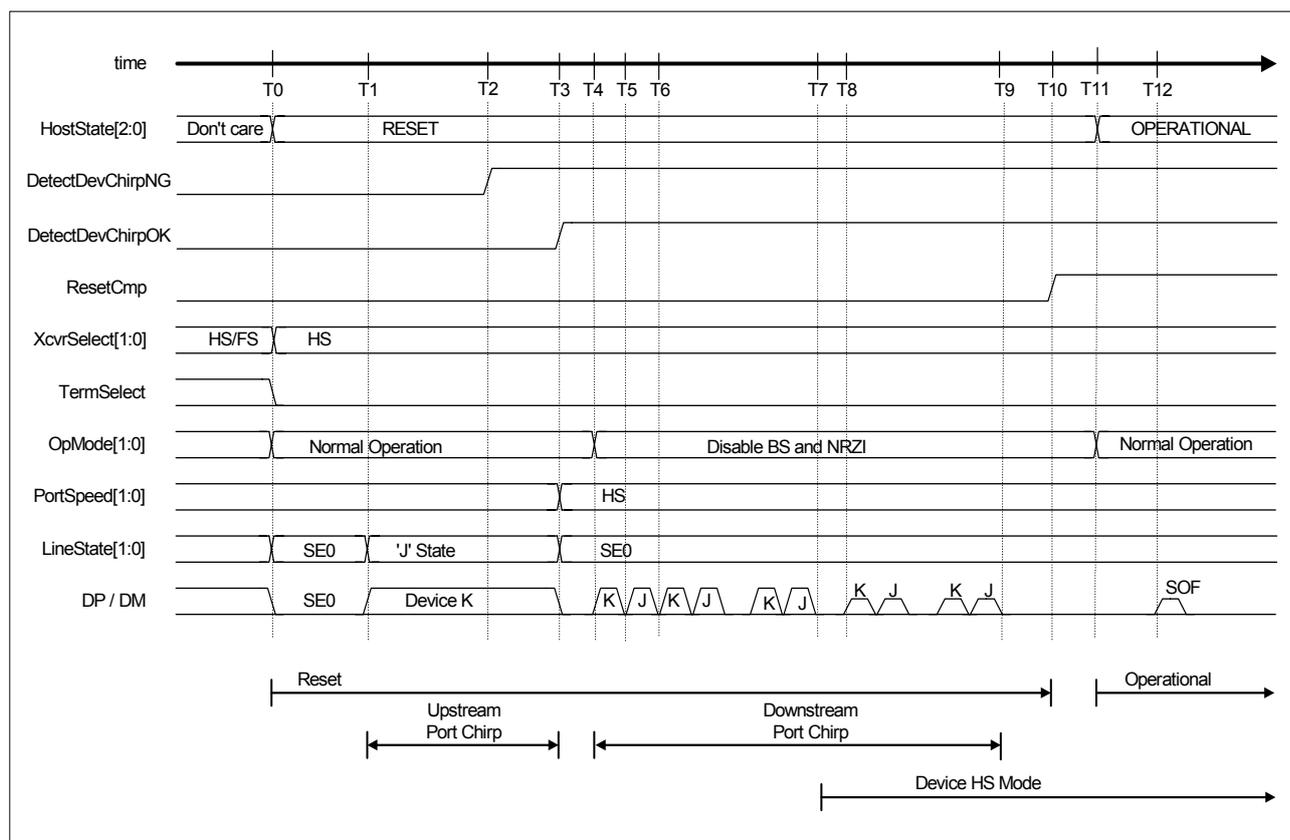


図 6-53 Detect Device Chirp NG Timing(チャープ完了ディセーブル設定 1)

表 6-45 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。	$T0 < T1 < T0 + 6.0ms$
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)	$T0 + 7ms\{T_{UCHEND}\} < T2$
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)	$T3 < T4 < T3 + 100\mu s\{T_{WTDCH}\}$
T5	ホストが Chirp K から Chirp J に切り替えて出力する。(H/W)	$T4 + 40\mu s\{T_{DCHBIT}\} < T4 < T4 + 60\mu s\{T_{DCHBIT}\}$
T6	ホストが Chirp J から Chirp K に切り替えて出力する。(H/W)	$T5 + 40\mu s\{T_{DCHBIT}\} < T6 < T5 + 60\mu s\{T_{DCHBIT}\}$
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。	$T7 < T8 < T6 + 500\mu s$
T9	ホストが Chirp を終了。(H/W)	$T4 + 50ms\{T_{DRSTR}\} < T9$

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)	$T9 < T10 < T9 + 150\mu s$
T11(参考)	H_NegoControl0.AutoMode に GoOPERATIONAL を設定する。(F/W)	$T11 < T10 + 200\mu s$
T12(参考)	最初の SOF が送出。(H/W)	$T11 + 120\mu s < T12 < T11 + 130\mu s$ $T9 + 100\mu s \{T_{DCHSE0}\} < T12 <$ $T9 + 500\mu s \{T_{DCHSE0}\}$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.4.3. 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_XcvrControl.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)。

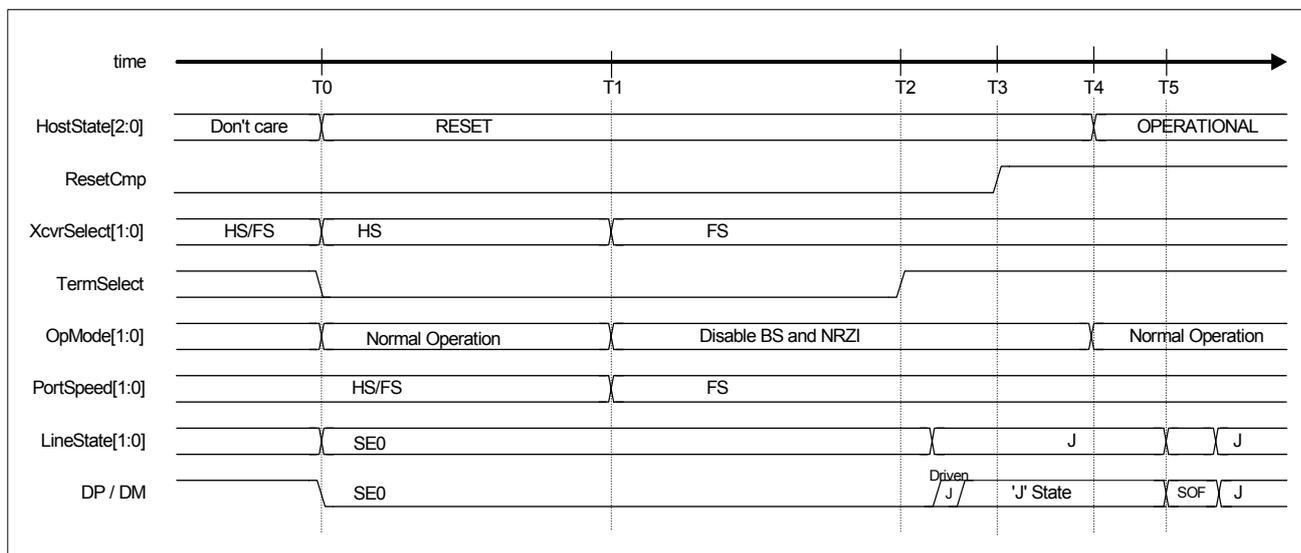


図 6-54 Reset Timing (FS mode)

表 6-46 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 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)	$T0 + 7.0ms\{T_{UCHEND}\} < T1$
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 SOF 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。

6.3.9.3.4.4. 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)。

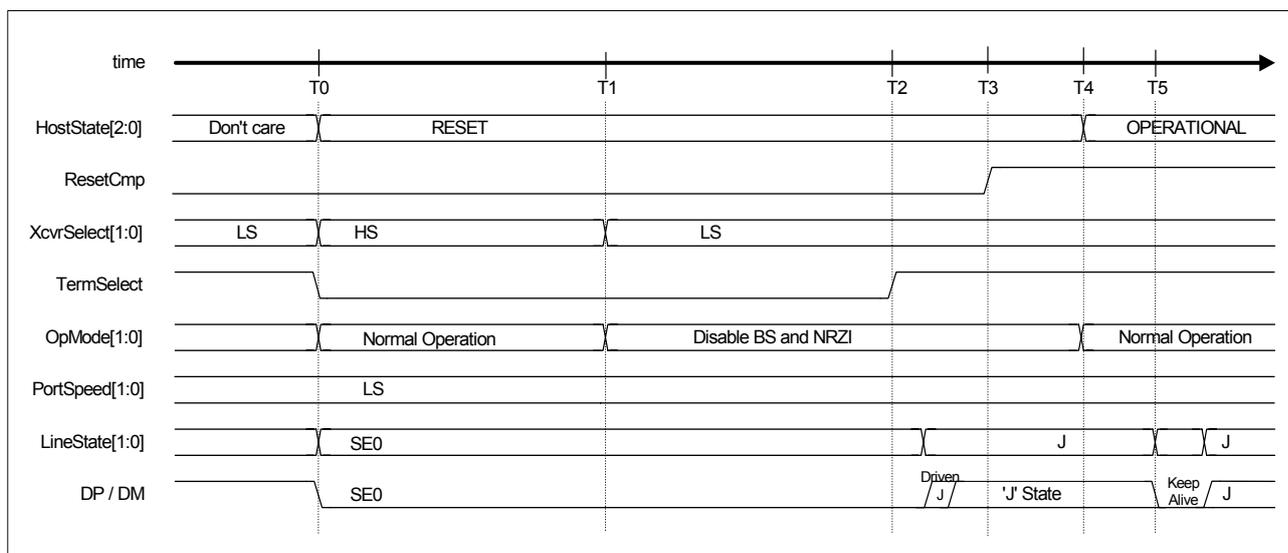


図 6-55 Reset Timing (LS mode)

表 6-47 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 LS に設定する。(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)	$T2 + 150\mu\text{s} < T3$
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)	$T4 + 0.9\text{ms} < T5 < T4 + 1.1\text{ms}$ ($T5 < T2 + 3\text{ms}$)

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.5. 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_XcvrControl.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、LSに設定されている場合は最初のKeepAliveが発行され(T1)、以降チャネルの設定に従って転送が行われます。

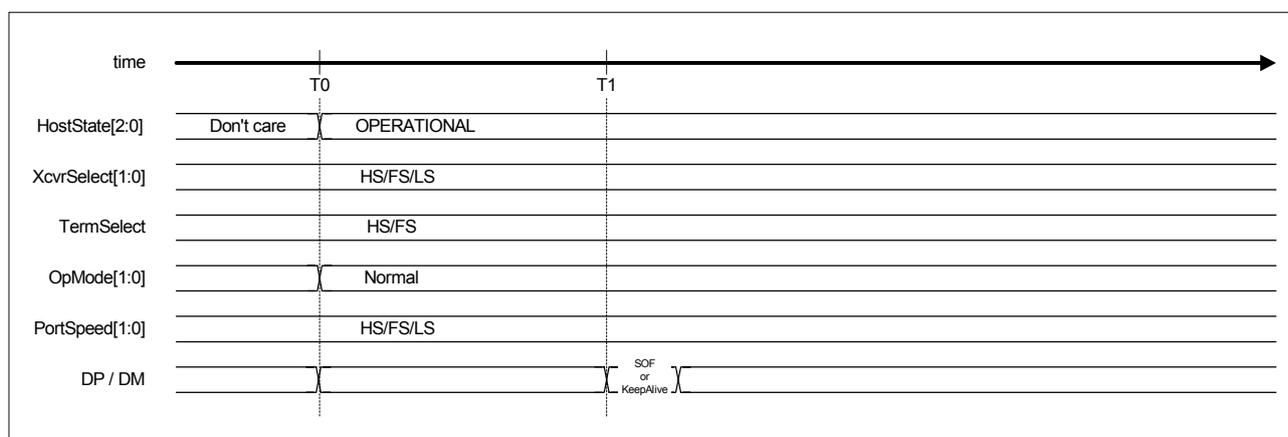


図 6-56 GoOPERATIONAL Timing

表 6-48 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。	$T0+120\mu s < T1(HS) < T0 + 130\mu s$ $T0+0.9ms < T1(FS,LS) < T0 + 1.1ms$

6.3.9.3.6. 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。

6.3.9.3.6.1. 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、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)。

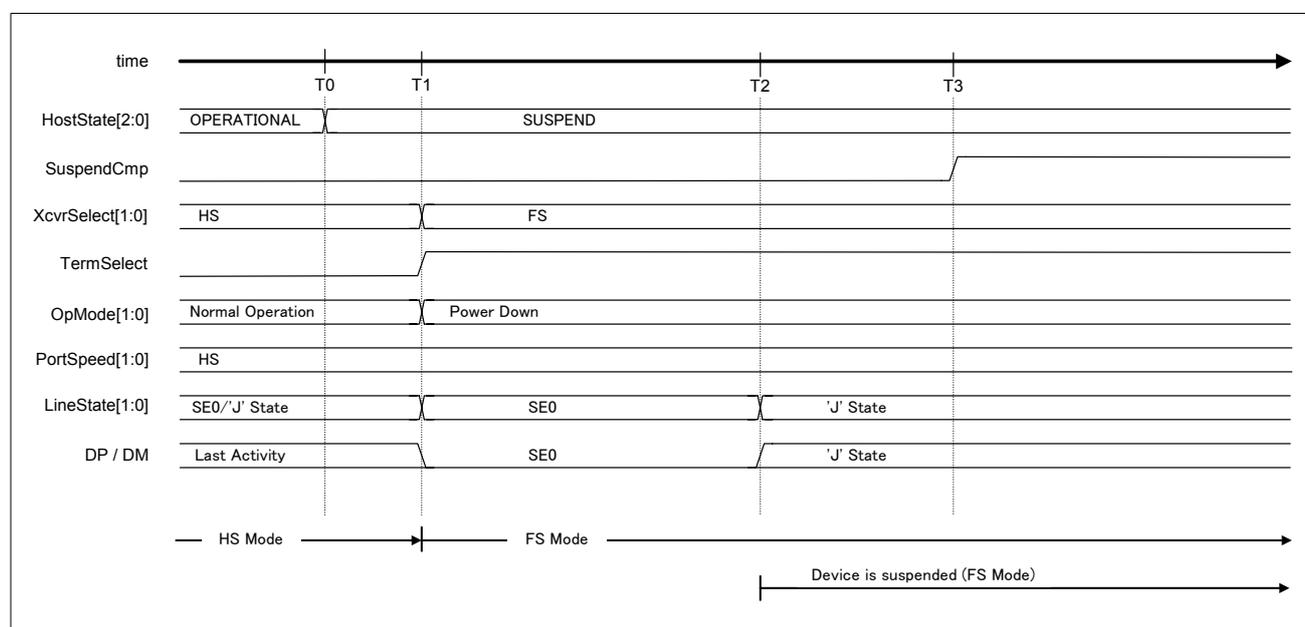


図 6-57 Suspend Timing (HS mode)

表 6-49 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、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。	$T1 + 3.0\text{ms} < T2 \{T_{WTRV}\} < T1 + 3.125\text{ms}$
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。

6.3.9.3.6.2. 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_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)。

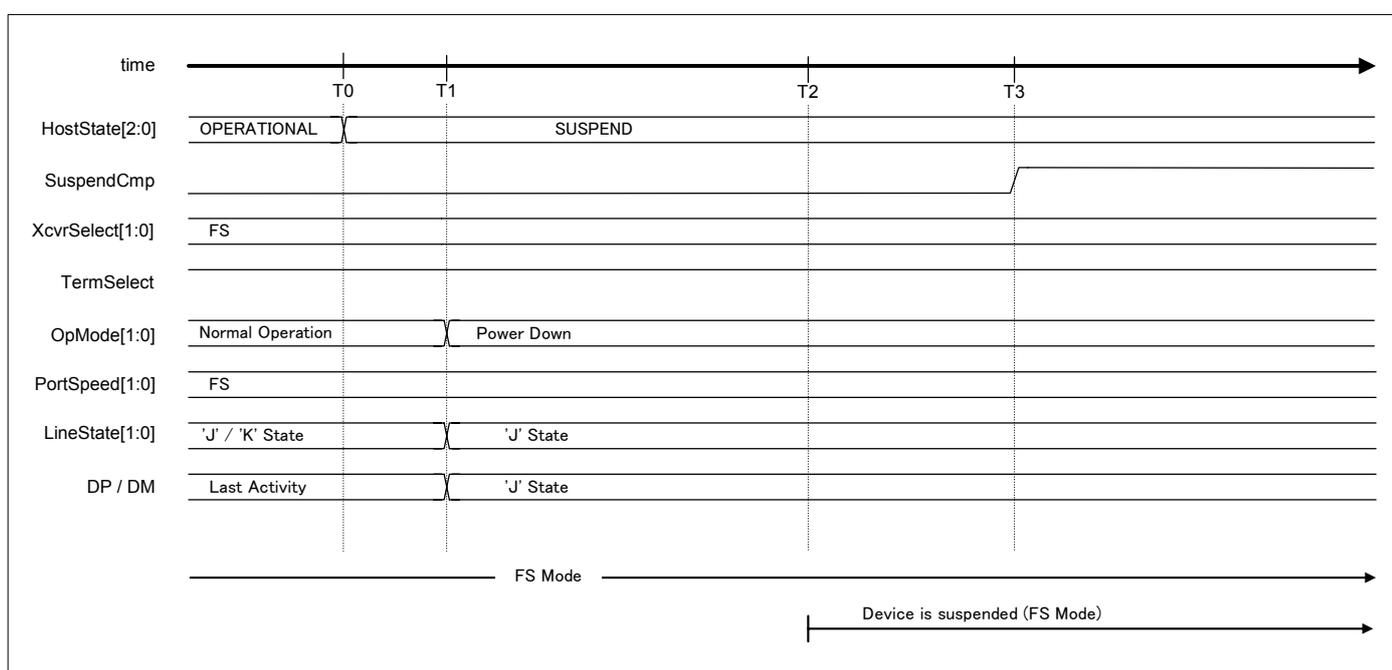


図 6-58 Suspend Timing (FS mode)

表 6-50 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。(HW)	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_XcvrControl.OpMode[1:0])を”PowerDown”に設定する。(HW)	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。(HW)	$T1 + 5\text{ms}\{T_{WTRSM}\} < T3$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.6.3. 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_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_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)。

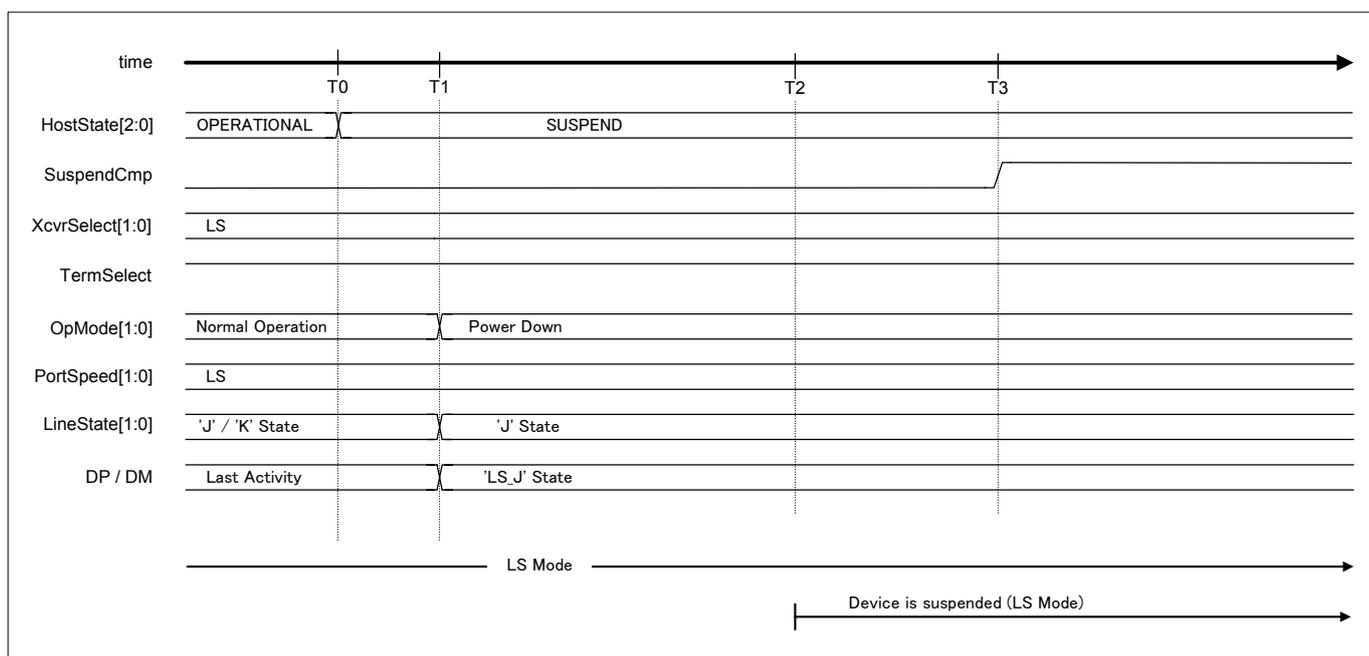


図 6-59 Suspend Timing (LS mode)

表 6-51 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。(HW)	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_XcvrControl.OpMode[1:0])を”PowerDown”に設定する。(HW)	T1
T2	デバイスがサスペンドを検出する。	$T1 + 3.0ms < T2 \{T_{WTREV}\} < T1 + 3.125ms$
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。(HW)	$T1 + 5ms\{T_{WTRSM}\} < T3$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.7. 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。

6.3.9.3.7.1. 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 Go RESUME 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_XcvtControl.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(XcvtControl.TermSelect)を HS に設定します(T2)。
- (7) トランシーバ選択(XcvtControl.XcvtSelect)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(SIE_IntStat_1.ResumeCmp)を発行します(T3)。

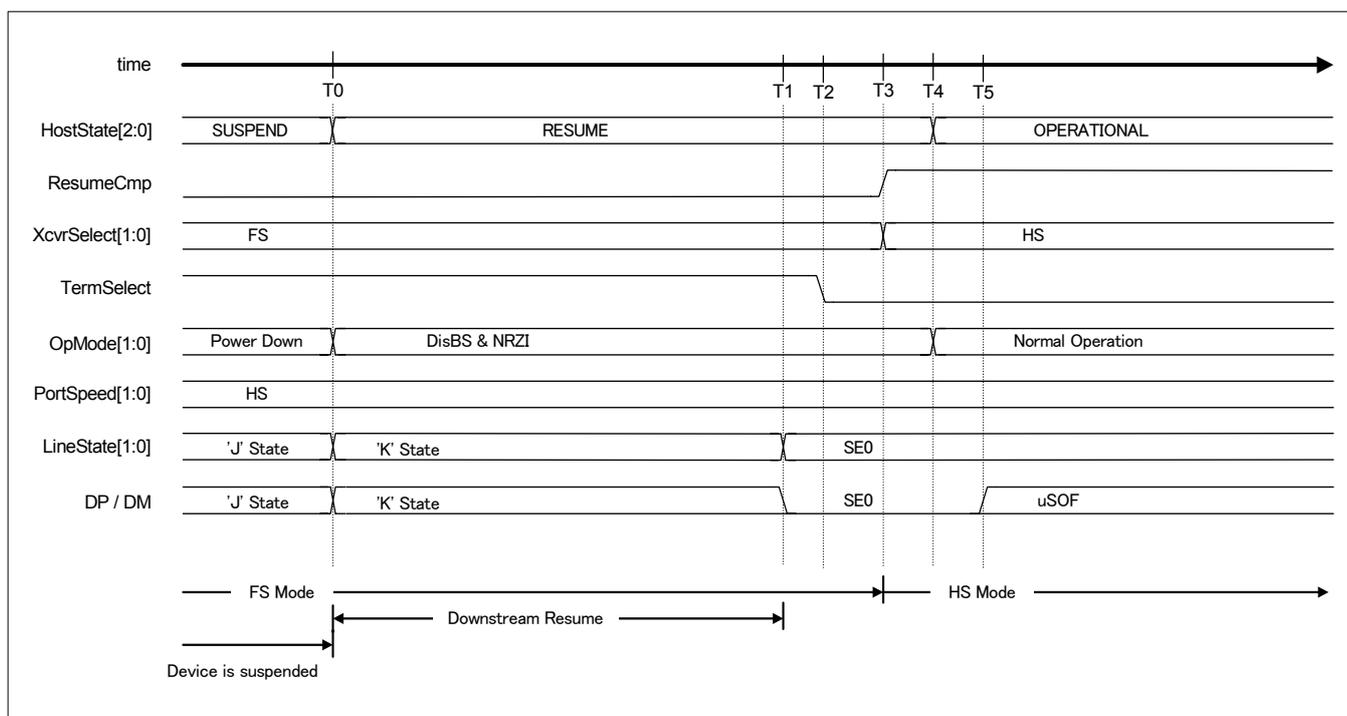


図 6-60 Resume Timing (HS mode)

表 6-52 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)	$T0 + 20\text{ms}\{T_{\text{DRSM DN}}\} < T1$
T2	トランシーバ選択を HS に設定する。(H/W)	$T1 + 100\text{ns} < T2 < T1 + 2.0\text{us}$
T3	レジューム完了ステータス (SIE_IntStat_1.ResumeCmp) を発行する。 (H/W)	$T1 + 90\text{us} < T3 < T1 + 110\text{us}$
T4(参考)	GoOPERATIONAL を設定する。(F/W) オペレーションモードを”NormalOperation”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)	$T5 < T1 + 3\text{ms}$ $T4 + 120\text{us} < T5 < T4 + 130\text{us}$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.7.2. 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 Go RESUME 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(SIE_IntStat_1.ResumeCmp)を発行します(T2)。

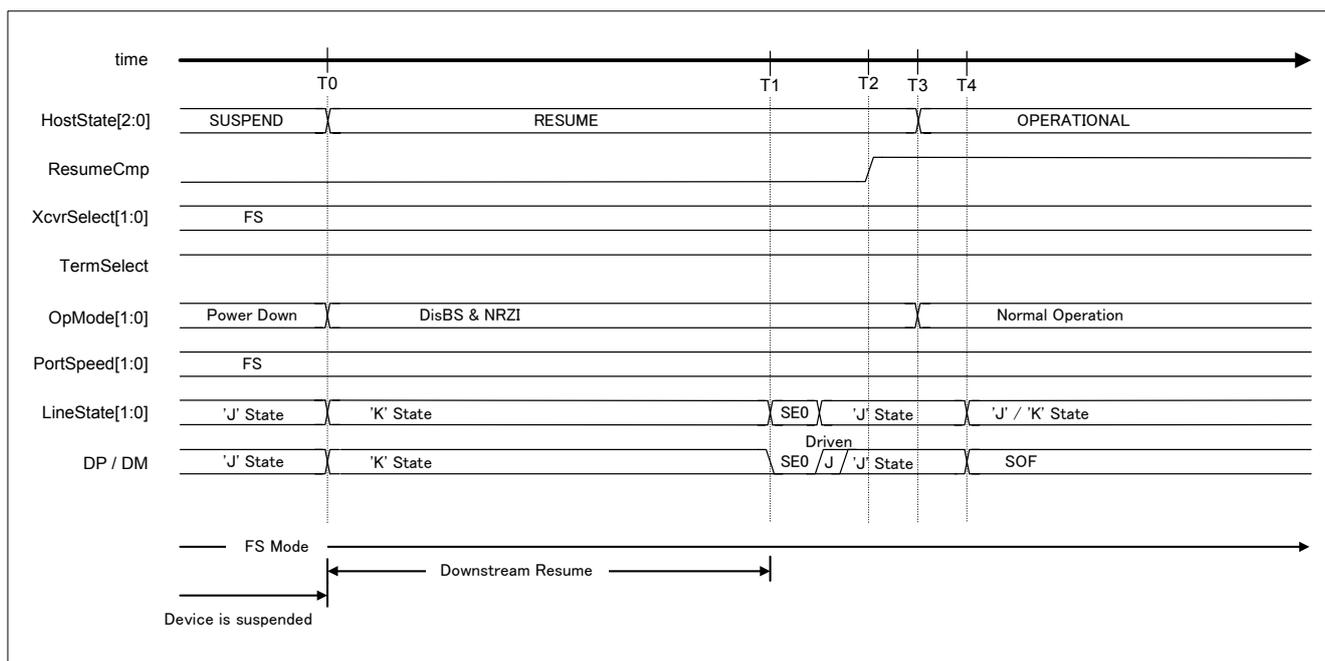


図 6-61 Resume Timing (FS mode)

表 6-53 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)	$T0 + 20ms\{T_{DRSMDN}\} < T1$
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)	$T1 + 90\mu s < T2 < T1 + 110\mu s$
T3(参考)	GoOPERATIONAL を設定する。(F/W) オペレーションモードを”Normal Operation”に設定する。(H/W)	T3
T4(参考)	最初の SOF を発行する。(H/W)	$T4 < T1 + 3ms$ $T3 + 0.9ms < T4 < T3 + 1.1ms$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.7.3. 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 Go RESUME 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(SIE_IntStat_1.ResumeCmp)を発行します(T2)。

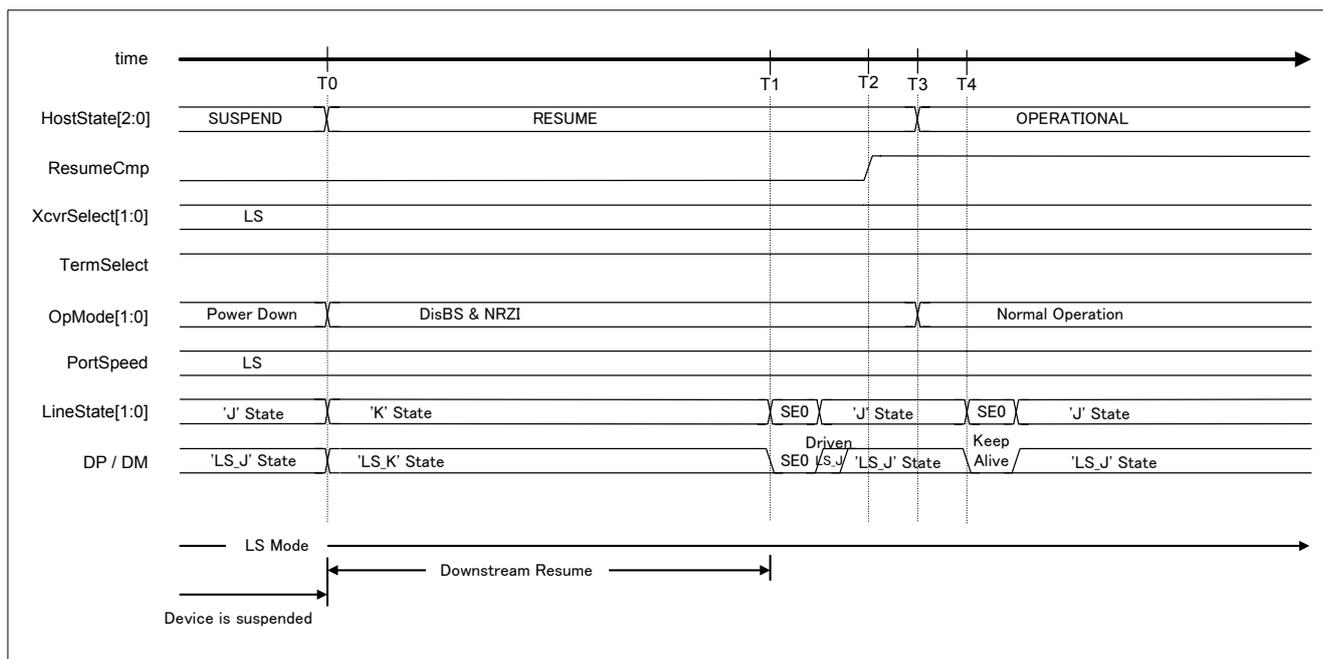


図 6-62 Resume Timing (LS mode)

表 6-54 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)	$T0 + 20ms\{T_{DRSMDN}\} < T1$
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)	$T1 + 90\mu s < T2 < T1 + 110\mu s$
T3(参考)	GoOPERATIONAL を設定する。(F/W) オペレーションモードを”NormalOperation”に設定する。(H/W)	T3
T4(参考)	最初の Keep Alive を発行する。(H/W)	$T4 < T1 + 3ms$ $T3 + 0.9ms < T4 < T3 + 1.1ms$

注: {}は、USB2.0 規格書で規格されている名称である。

6.3.9.3.8. 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、GoWAIT_CONNECT、GoDISABLEDの項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、切断検出、VBUS異常の項を参照してください。

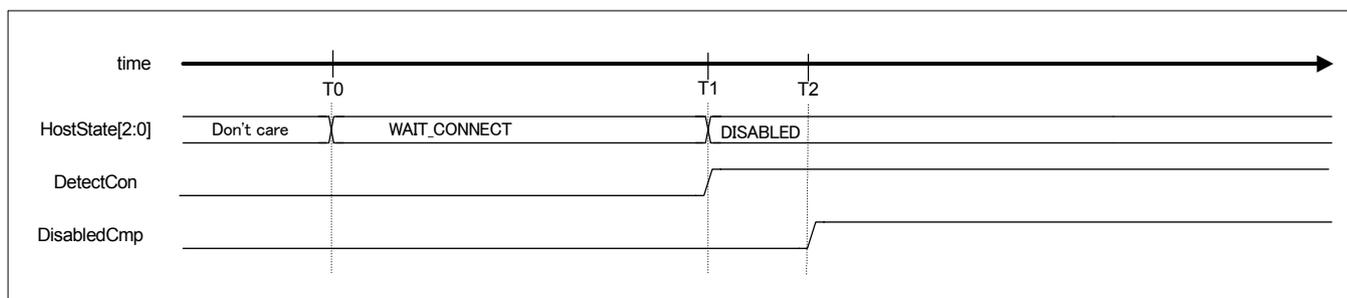


図 6-63 GoWAIT_CONNECTtoDIS Timing (HS mode)

表 6-55 GoWAIT_CONNECTtoDIS Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に GoWAIT_CONNECTtoOP を設定する。(FW) GoWAIT_CONNECT と同等の処理を実行する。(HW)	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。(HW)	T1
T2	ディセーブルド完了ステータスを発行する。(HW)	T2

6.3.9.3.9. 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。

6.3.9.3.9.1. 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、GoWAIT_CONNECT、GoDISABLED、GoRESET、GoOPERATIONAL の項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、切断検出、VBUS 異常、GoRESET の項を参照してください。

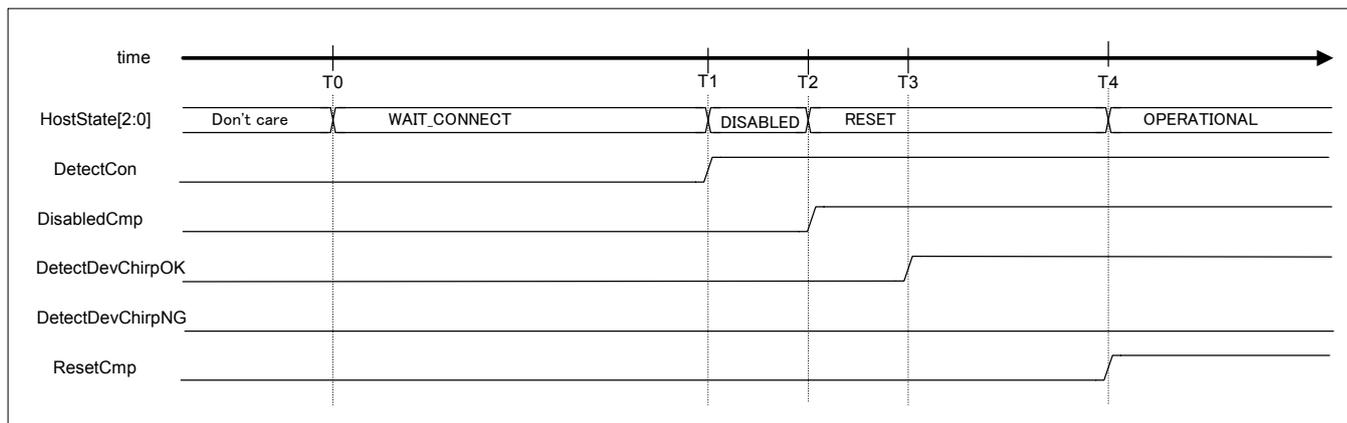


図 6-64 GoWAIT_CONNECTtoOP Timing (HS mode)

表 6-56 GoWAIT_CONNECTtoOP Timing Values (HS mode)

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に GoWAIT_CONNECTtoOP を設定する。(FW) GoWAIT_CONNECT と同等の処理を実行する。(HW)	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。(HW)	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。(HW)	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。(HW)	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。(HW)	T4

6.3.9.3.9.2. 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、GoWAIT_CONNECT、GoDISABLED、GoRESET、GoOPERATIONAL の項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、切断検出、VBUS 異常の項を参照してください。

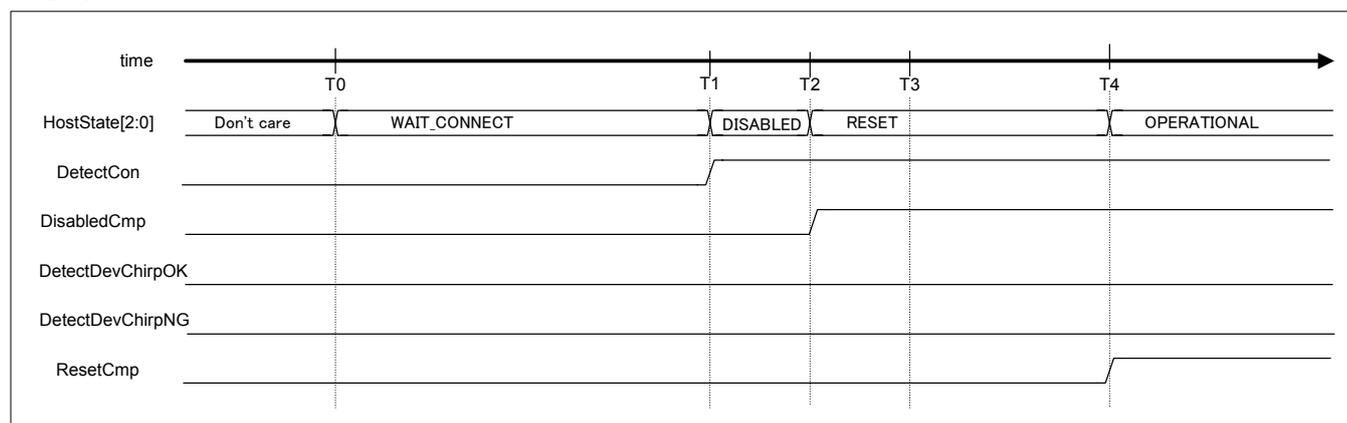


図 6-65 GoWAIT_CONNECTtoOP Timing (FS または LS mode)

表 6-57 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。(FW) GoWAIT_CONNECT と同等の処理を実行する。(HW)	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。(HW)	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。(HW)	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。(HW)	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。(HW)	T4

6.3.9.3.10. GoRESETtoOP

ホスト状態遷移実行(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。

6.3.9.3.10.1. 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状態遷移実行(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、GoRESET、GoOPERATIONAL の項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、VBUS 異常および GoRESET の項を参照してください。

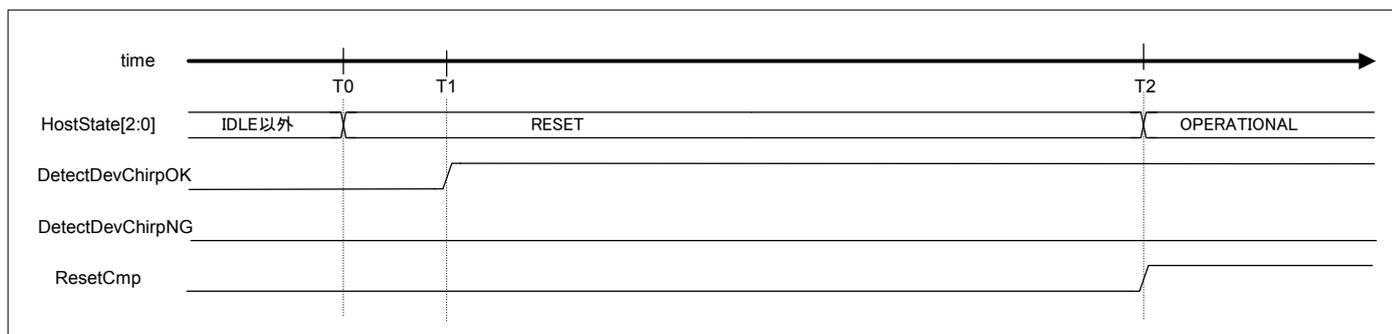


図 6-66 GoRESETtoOP Timing (HS mode)

表 6-58 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

6.3.9.3.10.2. 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、GoRESET、GoOPERATIONAL の項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、VBUS 異常の項を参照してください。

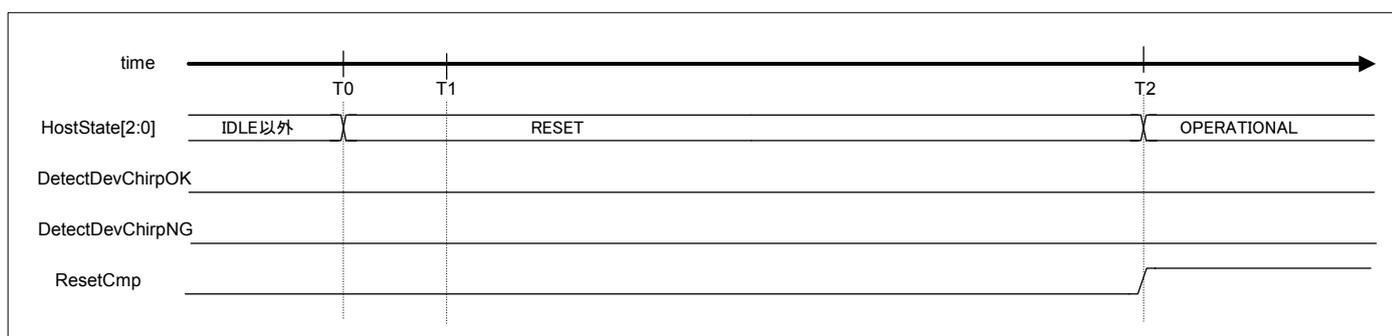


図 6-67 GoRESETtoOP Timing (FS または LS mode)

表 6-59 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

6.3.9.3.11. 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)。

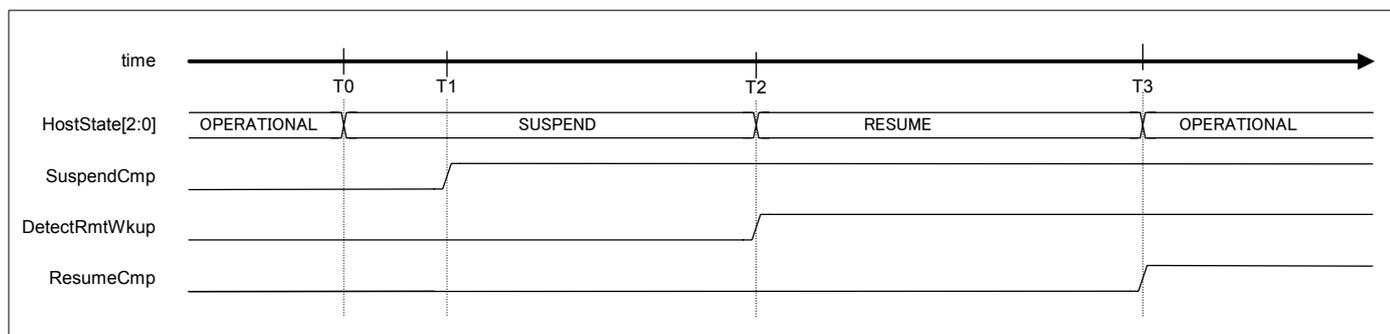


図 6-68 GoSUSPENDtoOP Timing

表 6-60 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

6.3.9.3.12. 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)。

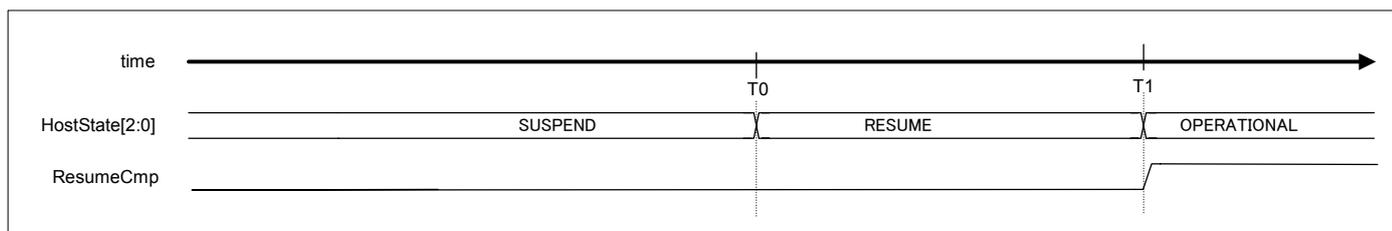


図 6-69 GoRESUMEtoOP

表 6-61 GoRESUMEtoOP Timing Value

Timing Parameter	Description	Value
T0	H_NegoControl_0.AutoMode に GoRESUMEtoOP を設定する。(F/W) Go RESUME 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

6.4. メディアデータ転送機能

6.4.1. メディアデータ

機器が扱うデータはアプリケーションにより様々であり、例えば音楽データや映像データなどがあります。これらのデータを総称して、本書ではメディアデータと呼びます。

6.4.2. メディアデータ転送

本 LSI は、USB 転送に使用する FIFO とは別に、メディアデータの転送に使用する 64 バイトの FIFO (メディア FIFO) を有します。この FIFO を使用して、HDD とメモリ間でのメディアデータの転送が容易に実現できます。

このように、メディア FIFO を介して HDD とメモリ間の転送を行う本 LSI の機能を、本書ではメディアデータ転送機能と呼びます。メディアデータ転送の様子を再生時 (HDD → メモリ) を例に示します。

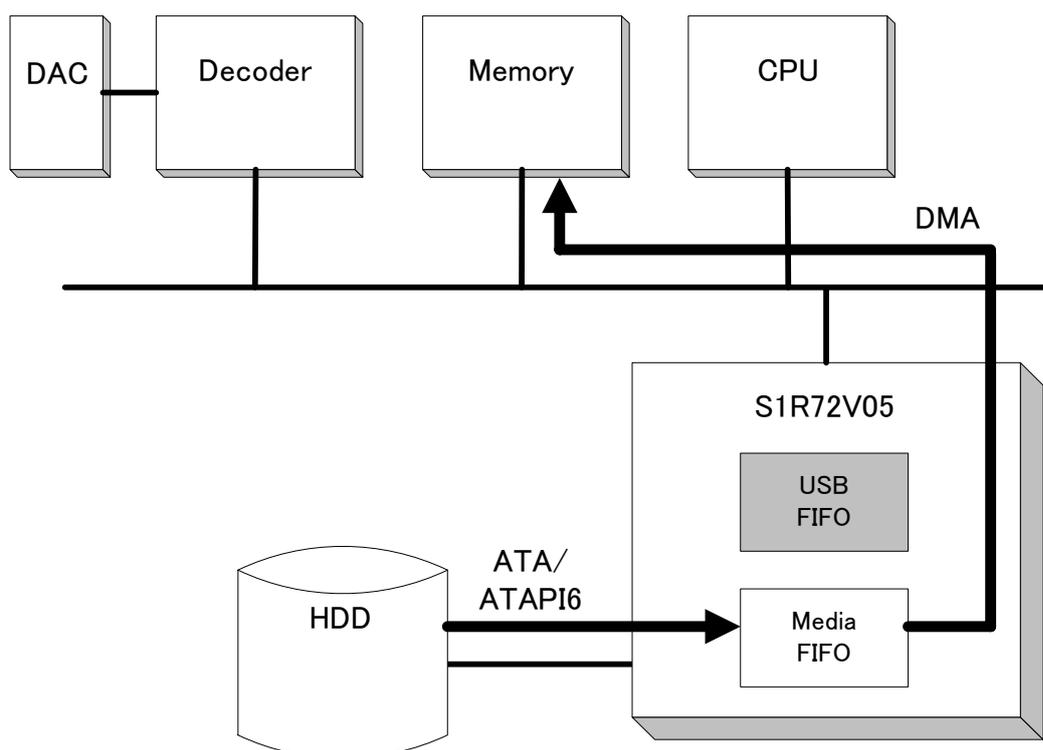


図 6-70 メディアデータ転送例(再生時)

メディア FIFO へのアクセスは、MediaFIFO_Join レジスタにより、IDE ポートと、もう一方のポート (DMA0/1, CPU_Rd, CPU_Wr のいずれか) を指定することで行います。

メディア FIFO へのアクセス方法の詳細については、メディア FIFO 管理 (6.6.3) を参照してください。

6.4.3. 消費電力の削減

メディアデータ転送機能は、本 LSI の電力モードステートのうち、ACTIVE60、ACT_DEVICE、ACT_HOST のいずれかのステートで使用することができます。このうち、ACTIVE60 ステートでは、USB デバイス部および USB ホスト部の機能を停止しますので、消費電力を大きく削減することができます。

パワーマネジメント機能の詳細は、パワーマネジメント機能(6.5)を参照してください。

USB 転送とメディアデータ転送を行う場合、およびメディアデータ転送のみを行う場合のデータフローの様子をそれぞれ下図に示します。

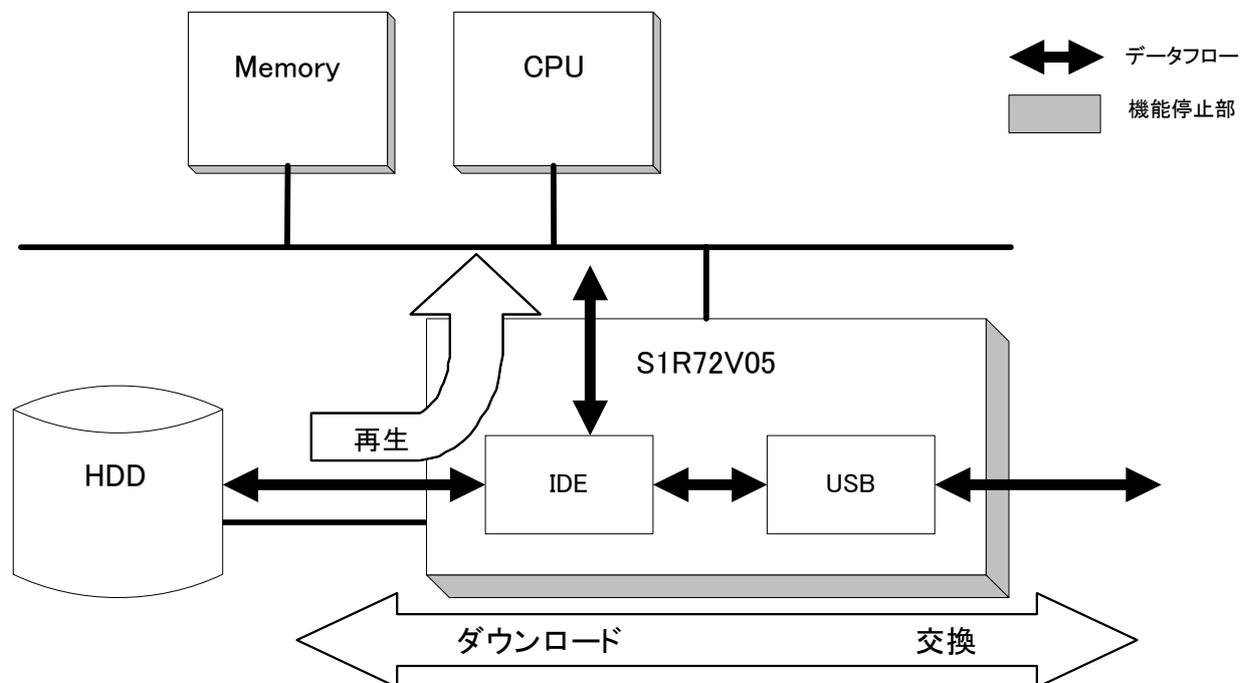


図 6-71 USB 転送とメディアデータ転送(例:再生時)

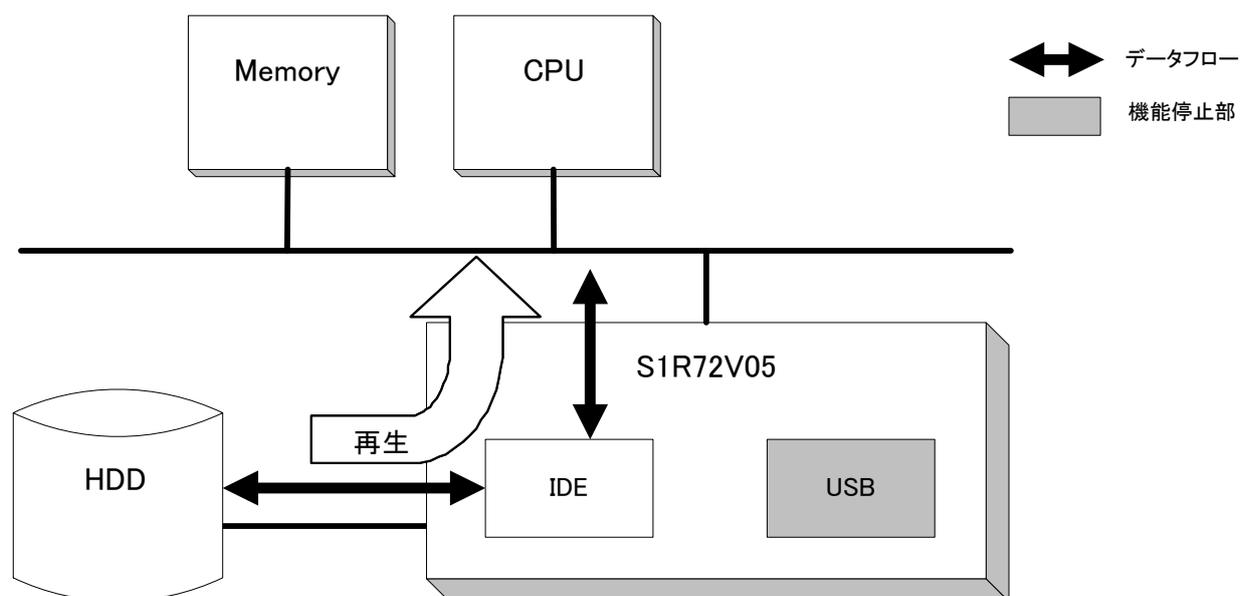
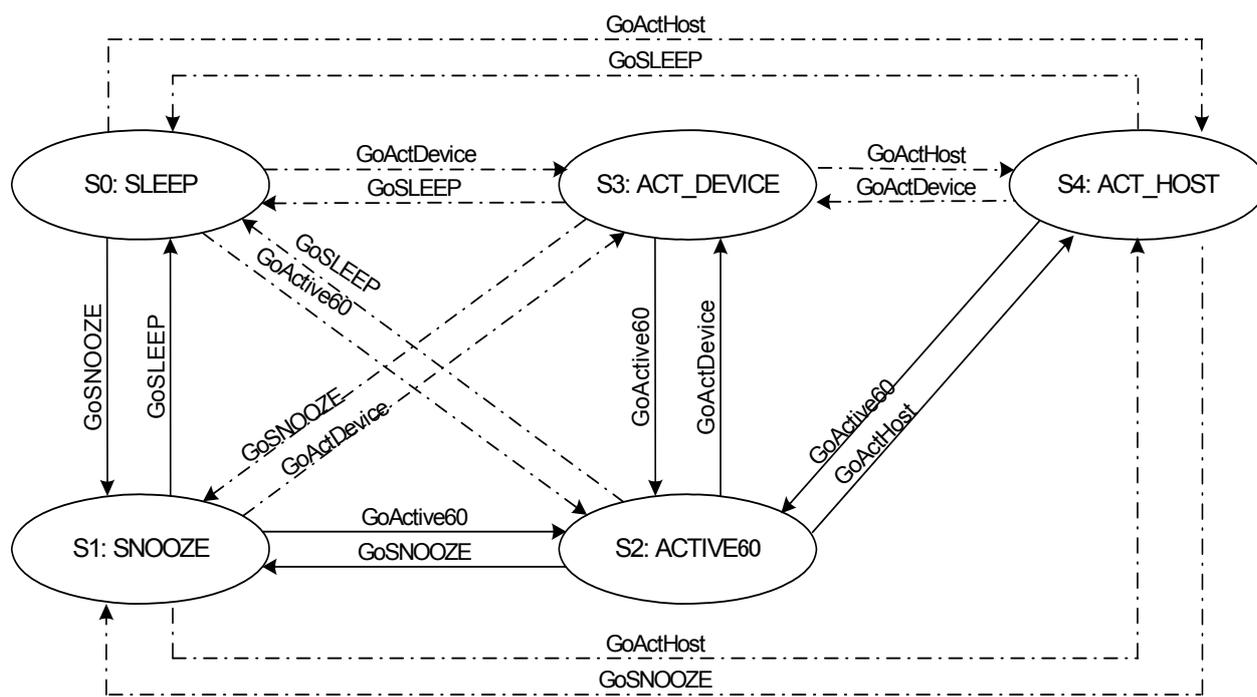


図 6-72 メディアデータ転送(例:再生時)のみ

6.5. パワーマネージメント機能

オシレータ、PLL (DevicePLL480, HostPLL480, PLL60) の動作を制御し、スリープ、スヌーズ、アクティブ 60、アクトデバイス、アクトホストの5ステートを遷移します。他のステートに遷移する場合には、PM_Control_0.GoSLEEP、PM_Control_0.GoSNOOZE、PM_Control_0.GoActive60、PM_Control_0.GoActDevice、PM_Control_0.GoActHost 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_1.PM_State[3:0]を確認してください。また遷移終了時、MainIntStat.FinishedPM イベントを発生します。この時、MainIntEnb.EnFinishedPM ビットをセットし、かつ MainIntEnb.EnSIE_IntStat ビットをセットしていれば、割り込み XINT が発生します。

全てのステートから他のステートへ遷移することが可能で、アクトデバイス・ステートまたはアクトホスト・ステート時に PM_Control.GoSLEEP ビットをセットした場合は、アクティブ 60・ステート、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 SIE_IntStat.FinishedPM イベントを発生します。またスリープ・ステート時に PM_Control.GoActDevice ビットまたは PM_Control.GoActHost ビットをセットした場合も、スヌーズ・ステート、アクティブ 60・ステートを経由し、アクトデバイス・ステートまたはアクトホスト・ステートに遷移し、完全に遷移終了した時点で MainIntStat.FinishedPM イベントを発生します。同様に、アクティブ 60・ステート時に PM_Control.GoSLEEP ビットをセットした場合も、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 MainIntStat.FinishedPM イベントを発生します。またスリープ・ステート時に PM_Control_0.GoActive60 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 60・ステートに遷移し、完全に遷移終了した時点で MainIntStat.FinishedPM イベントを発生します。また、アクトデバイス・ステート時に PM_Control_0.GoActHost ビットをセットした場合は、アクティブ 60・ステートを経由してアクトホスト・ステートに遷移し、完全に遷移終了した時点で MainIntStat.FinishedPM イベントを発生します。同様に、アクトホスト・ステート時に PM_Control_0.GoActDevice ビットをセットした場合は、アクティブ 60・ステートを経由してアクトデバイス・ステートに遷移し、完全に遷移終了した時点で MainIntStat.FinishedPM イベントを発生します。



※一点鎖線の遷移は、実際には実線を通して遷移する。

図 6-73 パワーマネージメント

6.5.1. SLEEP (スリープ)

オシレータが発振していない状態です。したがって、この状態では、PLL も発振していません。

スヌーズ・ステート、アクティブ 60・ステート、アクトデバイス・ステート及びアクトホスト・ステート中に PM_Control_0.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 PLL、OSC を DevicePLL480 または HostPLL480、PLL60 の順で停止し、最終的に、OSCCLK 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_0.GoSNOOZE、PM_Control_0.GoActive60 ビット、PM_Control.GoActDevice ビット及び PM_Control_0.GoActHost 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。

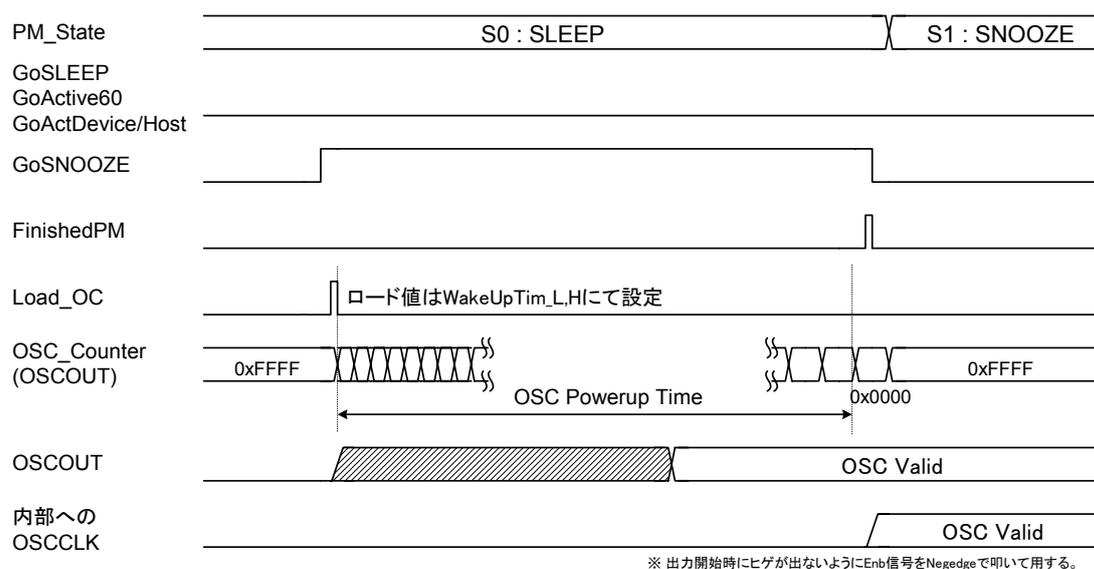


図 6-74 SLEEP ステートからの離脱(GoSNOOZE 時)

6.5.2. SNOOZE (スヌーズ)

オシレータは発振している状態で、PLL が発振していない状態です。

アクティブ 60・ステート、アクトデバイス・ステート及びアクトホスト・ステート中に PM_Control_0.GoSNOOZE ビットをセットし、スヌーズに遷移する場合は、出力しているクロックを停止した後、DevicePLL480 または HostPLL480、PLL60 の順で停止します。

しかし、スヌーズ・ステート中に PM_Control_0.GoActDevice ビット及び PM_Control.GoActHost ビット及び PM_Control_0.GoActive60 ビットをセットし、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。

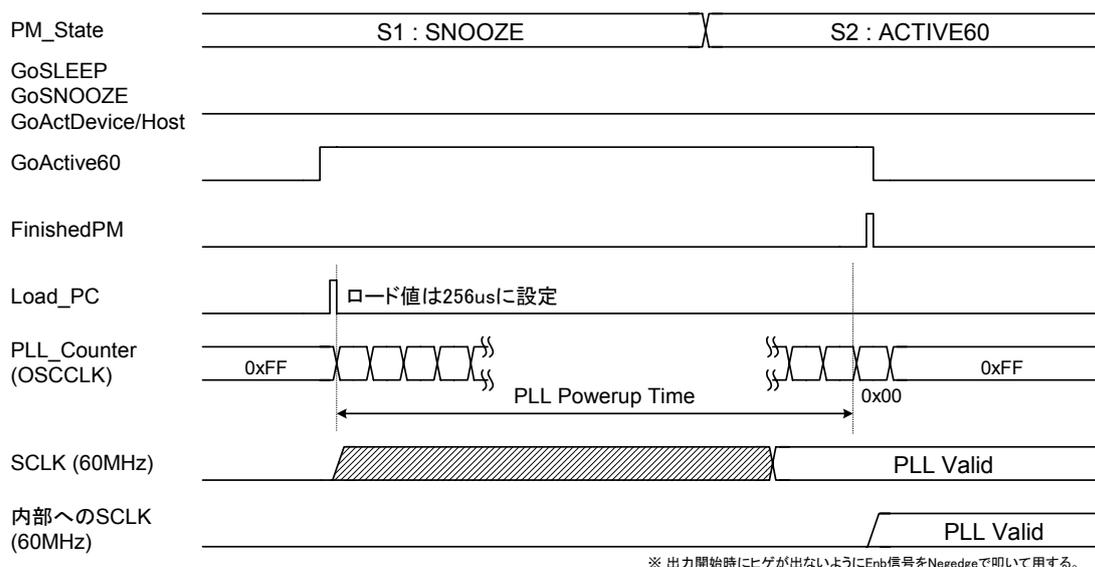


図 6-75 SNOOZE ステートからの離脱(GoActive60 時)

6.5.3. ACTIVE60 (アクティブ 60)

オシレータ、PLL60 が動作し、DevicePLL480 およびHostPLL480 が停止しているステートです。レジスタマップの**斜体太字**のレジスタ及びビットはスヌーズ及びスリープ時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いては、アクティブ 60・ステートでは、デバイス/ホスト共通レジスタマップ(7.1)のレジスタは読み書きでき(注)、デバイス・レジスタマップ(7.2)およびホスト・レジスタマップ(7.3)のレジスタは読むことができます。また、USB回路は、SCLK480 を必要としているため、USBデバイス機能はアクティブ・ステートでのみ、USBホスト機能はアクティブ・ステートでのみ動作します。IDE並びにCPU回路は、SCLK480 が不必要なため、アクティブ 60・ステート、アクティブ・ステート及びアクティブ・ステートのいずれかで動作します。480MHzのPLLの消費電流は 60MHzの消費電流に比較して大きいので、USBを必要としない場合にはアクティブ 60・ステートに遷移することで、低消費電力化が図れます。

(注)ただし、0x40～0x47, 0x50～0x6F のレジスタは、アクティブ 60・ステート時は読み書きできません。

6.5.4. ACT_DEVICE (アクティブ デバイス)

オシレータ、PLL60、DevicePLL480 が動作しているステートです。レジスタマップの**斜体太字**のレジスタ及びビットはスヌーズ及びスリープ時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いては、アクティブ・ステートでは、デバイス/ホスト共通レジスタマップ(7.1)およびデバイス・レジスタマップ(7.2)のレジスタは読み書きでき、ホスト・レジスタマップ(7.3)のレジスタは読むことができます。

また、USB デバイス回路は、SCLK480 を必要としているためアクティブ・ステートでのみ動作します。IDE 並びに CPU 回路は、SCLK480 が不必要なため、アクティブ 60・ステートもしくはアクティブ・ステート及びアクティブ・ステートのいずれかで動作します。480MHzのPLLの消費電流は60MHzの消費電流に比較して大きいので、USBを必要としない場合にはアクティブ 60・ステートに遷移することで、低消費電力化が図れます。

6.5.5. ACT_HOST (アクティブ ホスト)

オシレータ、PLL60、HostPLL480 が動作しているステートです。レジスタマップの**斜体太字**のレジスタ及びビットはスヌーズ及びスリープ時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いては、アクティブ・ステートでは、デバイス/ホスト共通レジスタマップ(7.1)およびホスト・レジスタマップ(7.3)のレジスタは読み書きでき、デバイス・レジスタマップ(7.2)のレジスタは読むことができます。また、USBホスト回路は、SCLK480 を必要としているためアクティブ・ステートでのみ動作します。IDE並びにCPU回路は、SCLK480 が不必要なため、アクティブ 60・ステートもしくはアクティブ・ステート及びアクティブ・ステートのいずれかで動作します。480MHzのPLLの消費電流は 60MHzの消費電流に比較して大きいので、USBを必要としない場合にはアクティブ 60・ステートに遷移することで、低消費電力化が図れます。

6.6. FIFO 管理

6.6.1. デバイス FIFO 管理

FIFO 管理について説明します。

6.6.1.1. FIFO メモリマップ

FIFO のメモリマップを以下に示します。

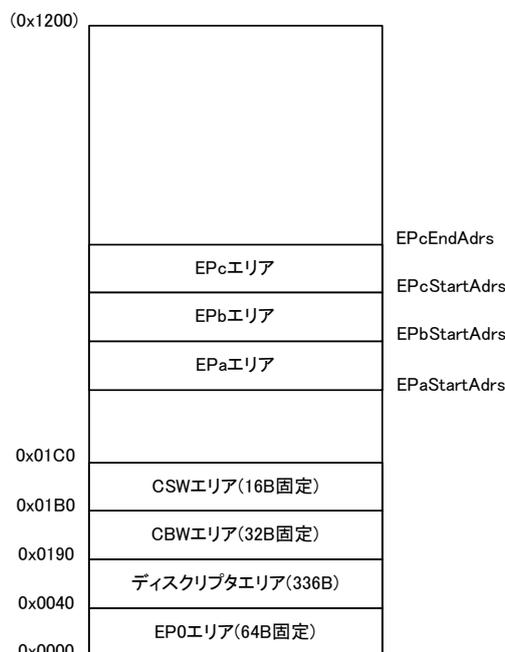


図 6-76 デバイス FIFO メモリマップ例

FIFOのメモリは、EP0 エリア、ディスクリプタエリア、CBWエリア、CSWエリア、EPaエリア、EPbエリア、EPcエリアの最大7エリアに分割して使用できます。EP0 エリア、ディスクリプタエリア、CBWエリア、CSWエリアについては 図 6-76 のように固定領域が割り当てられます。一方その他のEPx{x=a-c}エリアについてはFIFOエリア設定レジスタ(D_EPx{x=a-c}StartAdrs_H,L, D_EPcEndAdrs_H,L)で領域を柔軟に設定することが可能です。

EP0 エリアは、USB で必須のエンドポイント 0 に使用するエリアで、IN 及び OUT の両方向に使用されます。このエリアには、64 バイトが確保されていますが、そのうち使用できるのは、アドレス 0x000 から始まる、エンドポイント 0 のマックスパケットサイズ分の領域です。したがって、エンドポイント 0 は常にシングルバッファとなります。

ディスクリプタエリアは、ディスクリプタ返信機能で使用される領域です。336 バイトが確保されており、そ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、6.6.1.2 に後述します。実際には全てのFIFO領域をディ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をディスクリプタエリアと定め、使用を推奨します。

CBWエリアは、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 0x190 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、6.6.1.3 に後述します。

CSWエリアは、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 0x1B0 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、6.6.1.4 に後述します。

EPa エリア、EPb エリア、EPc エリアは、エンドポイント番号と、IN または OUT の方向を任意に設定できる、汎用エンドポイントのエリアです。EPa エリアは FS モードのバルク転送や HS/FS モードのインタラプト転送に使用することが出来ます。EPb エリア、EPc エリアは加えて HS モードのバルク転送にも使用することが出来ます。

EP0 エリア、EPa エリア、EPb エリア、EPc 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、D_EPnControl.AllFIFO_Clr ビットか D_EPnControl.EP0FIFO_Clr ビット、または、各領域に対応した D_EPnFIFO_Clr.EPx{x=a-c}FIFO_Clr 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、ディ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。

6.6.1.2. デスクリプタエリアの使用方法

デスクリプタエリアは、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 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。

6.6.1.2.1. 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り行います。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となっています。

6.6.1.2.2. 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 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_FIFO_IntStat.DescriptorCmp がセットされます。ファームウェアはステータスステージの処理に移行して下さい。

6.6.1.3. CBW エリアの使用方法

CBW エリアは、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 EPb, EPc)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 にはデータだけを受信することが出来、IDE や DMA 等の転送の制御を容易にします。

6.6.1.3.1. CBW エリアへの受信

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 バイト未満で無かった場合は、エラーステータスを発行し、データは破棄します。

6.6.1.3.2. CBW エリアからのデータの読み出し

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。

6.6.1.4. CSW エリアの使用方法

CSW エリアは、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 EPb, EPc)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 からはデータだけを送信することが出来、IDE や DMA 等の転送の制御を容易にします。

6.6.1.4.1. CSW エリアからの送信

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。

6.6.1.4.2. CSW エリアへのデータの書き込み

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 (0x1B0) 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なく問題ありません。

6.6.1.5. FIFO へのアクセス方法

FIFO へのアクセス要因には、CPU(レジスタ)、CPU(DMA)、IDE、USB が有ります。

6.6.1.5.1. FIFO へのアクセス方法 (RAM_Rd)

FIFO に CPUIF の 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 レジスタからデータを読み出してください。RAM_Rd レジスタのデータは 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、チャンネルの 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。

6.6.1.5.2. FIFO へのアクセス方法 (RAM_WrDoor)

FIFO に CPUIF の 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、チャンネルの FIFO 領域の設定に関係なく、任意に行う事が出来ます。

6.6.1.5.3. FIFO へのアクセス方法 (レジスタアクセス)

FIFO に CPU の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 つのチャンネルに対して、D_EPx{x=0,a-c}Join.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 CPU の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 つのチャンネルに対して、D_EPx{x=0,a-c}Join.JoinCPU_Wr に1をセットし、FIFO_Wr_0,1 レジスタに書き込みを行います。

FIFO_RdRemain_H,L レジスタは、D_EPx{x=0,a-c}Join.JoinCPU_Rd にてただ 1 つ設定されたチャンネルにおいて、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、D_EPx{x=0,a-c}Join.JoinCPU_Wr にてただ 1 つ設定されたチャンネルにおいて、FIFO に書き込み可能なエリアの残り数を表しています。

ICE 等を使用してファームウェア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、D_EPx{x=0,a-c}Join.JoinCPU_Rd レジスタ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。

6.6.1.5.4. FIFO へのアクセス方法 (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、DMA の各チャンネルにつき、D_EPx{x=0,a-c}Join.JoinDMAx{x=0,1} ビットによりただ一つのエンドポイントを選択し、DMAx{x=0,1}_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 の各チャンネルに付き、D_EPx{x=0,a-c}Join.JoinDMAx{x=0,1} ビットによりただ一つのエンドポイントを選択し、DMAx{x=0,1}_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。

DMAx{x=0,1}_Remain_H,L レジスタは、DMA の各チャンネルについて、D_EPx{x=0,a-c}Join.JoinDMAx{x=0,1} ビットによりただ一つ選択されたエンドポイントにおいて、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 の各チャンネルについて、D_EPx{x=0,a-c}Join.JoinDMAx{x=0,1} ビットによりただ一つ選択されたエンドポイントにおいて、FIFO に書き込み可能なエリアの残り数を表しています。

6.6.1.5.5. FIFO へのアクセス方法(IDE)

FIFO に IDE がアクセスする場合には、いずれか 1 つのチャンネルに対して、D_EPx{x=a-c}Join.JoinIDE ビットによりただ一つのエンドポイントを選択し、IDE 手順を実行してデータ転送を行います。IDE 転送の方向は IDE_Control.Dir ビットによります。

6.6.1.5.6. FIFO へのアクセス制限

本 LSI の FIFO には、USB との送受信、CPU バスからのレジスタまたは DMA 書き込み、読み出し、IDE との送受信が同時に行われ

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、それぞれの**エンドポイント**における 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ません。
- 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ません。
- 一つのエンドポイントには、JoinCPU_Wr,JoinCPU_Rd,JoinDMAx{x=0,1}のうちの一つしか設定できません。
- JoinIDE, JoinCPU_Wr, JoinCPU_Rd, JoinDMAx{x=0,1}は、それぞれ、同時に一つのエンドポイントにしか設定できません。
- JoinIDE,JoinCPU_Wr,JoinCPU_Rd,JoinDMAx{x=0,1}は、FIFO とメディア FIFO に同時には Join できません。

同一の FIFO エリアに対する書き込み／読み出し要因の重複に関し、USB 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 を設定して書き込むことができます。**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、JoinCPU_Wr 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 を設定して読み出すことができます。**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 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、各エンドポイントの EnEndpoint がクリアされている場合や、ForceNAK がセットされている場合などがあります。

6.6.2. ホスト FIFO 管理

USB ホスト機能における FIFO 管理について説明します。

6.6.2.1. FIFO メモリマップ

FIFO のメモリマップを以下に示します。

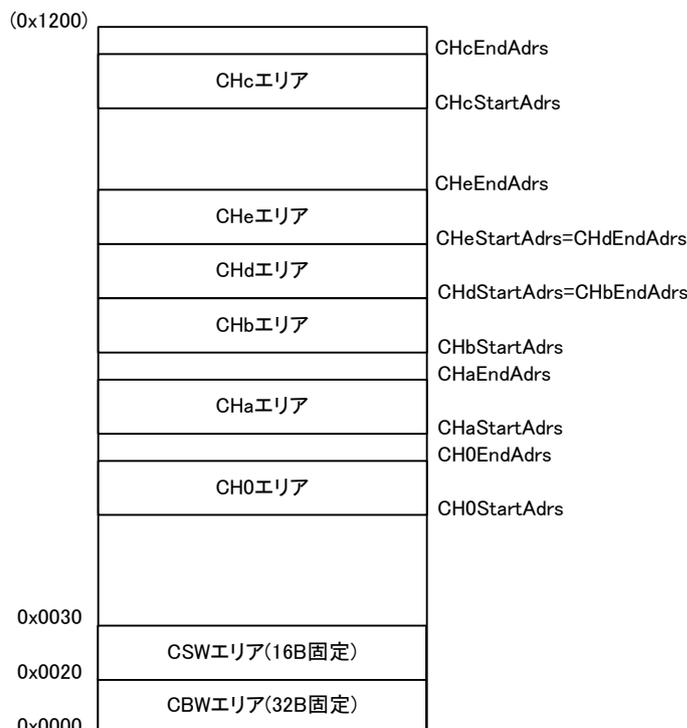


図 6-77 ホスト FIFO メモリマップ例

FIFO のメモリは、CBW エリア、CSW エリア、CH0 エリア、CHa エリア、CHb エリア、CHc エリア、CHd エリア、CHe エリア、の最大 8 エリア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 6-77 のように固定領域が割り当てられます。一方その他の CHx{x=0,a-e} エリアについては FIFO エリア設定レジスタ(H_CHx{x=0,a-e}StartAdrs_H,L, CHx{x=0,a-e}EndAdrs_H,L) で領域を柔軟に設定することが可能です。

CBW エリアは、0x0000 から 32 バイトの領域です。実際の使用方法については、6.6.2.2 に後述します。

CSW エリアは、0x0020 から 16 バイトの領域です。実際の使用方法については、6.6.2.3 に後述します。

CH0 エリアは、コントロール転送のみに使用するエリアです。

CHa エリアはバルク転送に使用するエリアです。

CHb エリア、CHc エリア、CHd エリア、CHe エリアはバルク転送およびインタラプト転送に使用するエリアです。

CH0 エリア、CHa エリア、CHb エリア、CHc エリア、CHd エリア、CHe 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、H_CHnControl.AllFIFO_Clr ビットか H_CHnControl.CH0FIFO_Clr ビット、または、各領域に対応した、H_CHrFIFO_Clr.CHx{x=a-e}FIFO_Clr 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。

6.6.2.2. CBW エリアの使用方法

CBW エリアは、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 のコマンドトランスポートを行う際に、このエリアから CBW データをデータパケットとして送信します。

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。

6.6.2.2.1. CBW エリアからの送信

バルクオンリーサポートを実行している時に、チャンネル CHa でコマンドトランスポートを行う際に、このエリアから CBW をデータパケットとして送信します。

6.6.2.2.2. CBW エリアへのデータの書き込み

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なく問題ありません。

6.6.2.3. CSW エリアの使用方法

CSW エリアは、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。

6.6.2.3.1. CSW エリアへの受信

バルクオンリーサポートを実行している時に、チャンネル CHa でステータストランスポートが行われ、CSW データを受信します。

6.6.2.3.2. CSW エリアからのデータの読み出し

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 (13 バイト) が読み出されて 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。

6.6.2.4. FIFO へのアクセス方法

FIFO へのアクセス要因には、CPU(レジスタ)、CPU(DMA)、IDE、USB が有ります。

6.6.2.4.1. FIFO へのアクセス方法 (RAM_Rd)

FIFO に CPUIF の 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 レジスタからデータを読み出してください。RAM_Rd レジスタのデータは 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、チャンネルの 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。

6.6.2.4.2. FIFO へのアクセス方法 (RAM_WrDoor)

FIFO に CPUIF の 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、チャンネルの FIFO 領域の設定に関係なく、任意に行う事が出来ます。

6.6.2.4.3. FIFO へのアクセス方法 (レジスタアクセス)

FIFO に CPU の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 つのチャンネルに対して、H_CHx{x=0,a-c}Join.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 CPU の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 つのチャンネルに対して、H_CHx{x=0,a-c}Join.JoinCPU_Wr に1をセットし、FIFO_Wr_0,1 レジスタに書き込みを行います。

FIFO_RdRemain_H,L レジスタは、H_CHx{x=0,a-c}Join.JoinCPU_Rd にてただ 1 つ設定されたチャンネルにおいて、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、H_CHx{x=0,a-c}Join.JoinCPU_Wr にてただ 1 つ設定されたチャンネルにおいて、FIFO に書き込み可能なエリアの残り数を表しています。

ICE 等を使用してファームウェアのデバッグを行う際に、レジスタのダンプ等を行う場合に、H_CHx{x=0,a-c}Join.JoinCPU_Rd レジスタ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。

6.6.2.4.4. FIFO へのアクセス方法 (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、DMA の各チャンネルにつき、H_CHx{x=0,a-h}Join.JoinDMAx{x=0,1} ビットによりただ一つのチャンネルを選択し、DMAx{x=0,1}_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 の各チャンネルに付き、H_CHx{x=0,a-h}Join.JoinDMAx{x=0,1} ビットによりただ一つのチャンネルを選択し、DMAx{x=0,1}_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。

DMAx{x=0,1}_Remain_H,L レジスタは、DMA の各チャンネルについて、H_CHx{x=0,a-h}Join.JoinDMAx{x=0,1} ビットによりただ一つ選択されたチャンネルにおいて、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 の各チャンネルについて、H_CHx{x=0,a-h}Join.JoinDMAx{x=0,1} ビットによりただ一つ選択されたチャンネルにおいて、FIFO に書き込み可能なエリアの残り数を表しています。

6.6.2.4.5. FIFO へのアクセス方法(IDE)

FIFO に IDE がアクセスする場合には、いずれか 1 つのチャンネルに対して、H_CHx{x=a-h}Join.JoinIDE ビットによりただ一つのチャンネルを選択し、IDE 手順を実行してデータ転送を行います。IDE 転送の方向は IDE_Control.Dir ビットによります。

6.6.2.4.6. FIFO へのアクセス制限

本 LSI の FIFO には、USB との送受信、CPU バスからのレジスタまたは DMA 書き込み、読み出し、IDE との送受信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、それぞれのチャンネルにおける 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ません。
- 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ません。
- 一つのチャンネルには、JoinCPU_Wr,JoinCPU_Rd,JoinDMAx{x=0,1}のうちの一つしか設定できない。
- JoinIDE, JoinCPU_Wr, JoinCPU_Rd, JoinDMAx{x=0,1}は、それぞれ、同時に一つのチャンネルにしか設定できない。
- JoinIDE,JoinCPU_Wr,JoinCPU_Rd,JoinDMAx{x=0,1}は、FIFO とメディア FIFO に同時には Join できない。

同一の FIFO エリアに対する書き込み／読み出し要因の重複に関し、USB 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、IN のチャンネルの FIFO エリアに対し、JoinCPU_Wr を設定して書き込むことができます。**この場合、必ず IN トランザクションが行われなかった状況にした後、JoinCPU_Wr を設定し、CPU からの書き込みを行う必要があります。**また、OUT のチャンネルの FIFO エリアに対し、JoinCPU_Rd を設定して読み出すことができます。**この場合、必ず OUT トランザクションが行われなかった状況にした後、JoinCPU_Rd を設定し、CPU からの読み出しを行う必要があります。**

6.6.3. メディア FIFO 管理

メディアデータ転送機能における FIFO 管理について説明します。

6.6.3.1. メディア FIFO

メディア FIFO は IDE およびメモリ間でのメディアデータの転送用に使用します。

メディア FIFO は、USB FIFO とは独立しており、自動的に 64 バイトが使用エリアの固定領域として割り当てられます。そのため、FW による領域設定は必要ありません。

使用エリアは、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、MediaFIFO_Control.MediaFIFO_Clr をセットして下さい。

6.6.3.2. FIFO へのアクセス方法

メディア FIFO へのアクセス要因には、CPU(レジスタ)、CPU(DMA)、IDE が有ります。

6.6.3.2.1. FIFO へのアクセス方法 (レジスタアクセス)

FIFO に CPU の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、MediaFIFOJoin.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 CPU の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 つのチャンネルに対して、MediaFIFOJoin.JoinCPU_Wr に1をセットし、FIFO_Wr_0,1 レジスタに書き込みを行います。

FIFO_RdRemain_H,Lレジスタは、MediaFIFOJoin.JoinCPU_Rdにて設定された場合に、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、MediaFIFOJoin.JoinCPU_Wr にて設定された場合に、FIFO に書き込み可能なエリアの残り数を表しています。

ICE 等を使用してファームウェア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、MediaFIFOJoin.JoinCPU_Rd レジスタ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。

6.6.3.2.2. FIFO へのアクセス方法 (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、DMA の各チャンネルにつき、MediaFIFOJoin.JoinDMAx{x=0,1} ビットによりただ一つのチャンネルを選択し、DMAx{x=0,1}_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 の各チャンネルに付き、MediaFIFOJoin.JoinDMAx{x=0,1}ビットによりただ一つのチャンネルを選択し、DMAx{x=0,1}_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。

DMAx{x=0,1}_Remain_H,L レジスタは、DMA の各チャンネルについて、MediaFIFOJoin.JoinDMAx{x=0,1}ビットによりただ一つ選択されたチャンネルにおいて、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 の各チャンネルについて、MediaFIFOJoin.JoinDMAx{x=0,1}ビットによりただ一つ選択されたチャンネルにおいて、FIFO に書き込み可能なエリアの残り数を表しています。

6.6.3.2.3. FIFO へのアクセス方法(IDE)

FIFO に IDE がアクセスする場合には、MediaFIFOJoin.JoinIDE ビットに 1 をセットし、IDE 手順を実行してデータ転送を行います。IDE 転送の方向は IDE_Control.Dir ビットによります。

6.6.3.2.4. FIFO へのアクセス制限

本 LSI の FIFO には、CPU バスからのレジスタまたは DMA 書き込み、読み出し、IDE との送受信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、**メディア FIFO へのアクセス設定方法(Join)**に対し、下記の排他ルールがあります。

- ・ 複数の書き込み要因を同時に設定してはいけません。
- ・ 複数の読み出し要因を同時に設定してはいけません。
- ・ メディア FIFO には、JoinCPU_Wr,JoinCPU_Rd,JoinDMAx{x=0,1}のうちの一つしか設定できない。
- ・ JoinIDE,JoinCPU_Wr,JoinCPU_Rd,JoinDMAx{x=0,1}は、FIFO とメディア FIFO に同時には Join できない。

6.7. CPUIF

6.7.1. モード切り替え

S1R72V05 の CPUIF は非同期 CPU に対応し、以下の 3 つの動作モードを持っております。

表 6-62 CPUIF 動作モード設定

動作モード	BusMode	Bus8x16	備考
16bit Strobe mode	0	0	デフォルト
16bit BE mode	1	*	BusMode ビットの設定優先
8bit mode	0	1	

各動作モードの切り替えは ChipConfig レジスタの BusMode ビット、Bus8x16 ビットの設定によって行います。ChipConfig レジスタの値は ModeProtect レジスタの設定によって誤書き込みに対するプロテクトをかけることができます。

実使用時には、電源投入直後、第一に ChipConfig レジスタの設定を行い動作モードを決定して下さい。その後、ModeProtect レジスタにより設定を保護して下さい。

また、S1R72V05 の CPUIF はバスの Swap 機能を持ちます。この機能を使用する場合は ChipConfig レジスタの初期設定時に ChipConfig.CPU_Endian ビットを設定して下さい。Swap 機能は CPU_Endian ビットを設定した後、B9h 番地をリードする事で有効になります。ChipConfig レジスタの設定では上記に加え、XINT の論理レベル及び出力モードの設定、XDREQ0,1、XDACK0,1 の論理レベル設定、DMA0,1 の CS_Mode の設定を行う事が出来ます。

以下の説明においては、特に断りの無い限りデフォルト設定の 16bit Strobe mode、Bus Swap なしを基本として説明致します。

6.7.2. モード切り替えに際して

S1R72V05 は ChipConfig レジスタにより、ご使用の CPU に適合する CPU バスの動作モードに設定することが可能です。チップの初期状態は 16bit Strobe mode で動作しておりますので、16bit BE mode、または 8bit mode に切り替える際には以下の点にご注意下さい。

6.7.2.1. 16bit BE mode をご使用の場合

ご使用の CPU に合わせて 16bit BE mode を使用する場合、6.7.1 に示す通り、ChipConfig レジスタへの設定を第一に行って下さい。また、その際には図 6-78 に示す通り、必ず B7h 番地に対するバイトライトを行って下さい。この時、S1R72V05 は初期状態の 16bit Strobe mode で動作しており、CPU の chip select 信号と byte mask high 信号 (XCS と XWRH) に下図に示すようなスキューがある場合、有効なライト期間と見なして内部的に動作してしまう可能性があります。S1R72V05 ではこの様なスキューを除去する為のフィルタ回路 (min:1ns) を搭載しておりますが、ご使用の CPU の AC 特性をご確認の上、基板上での処理等により、この様なスキューが生じないようにして下さい。

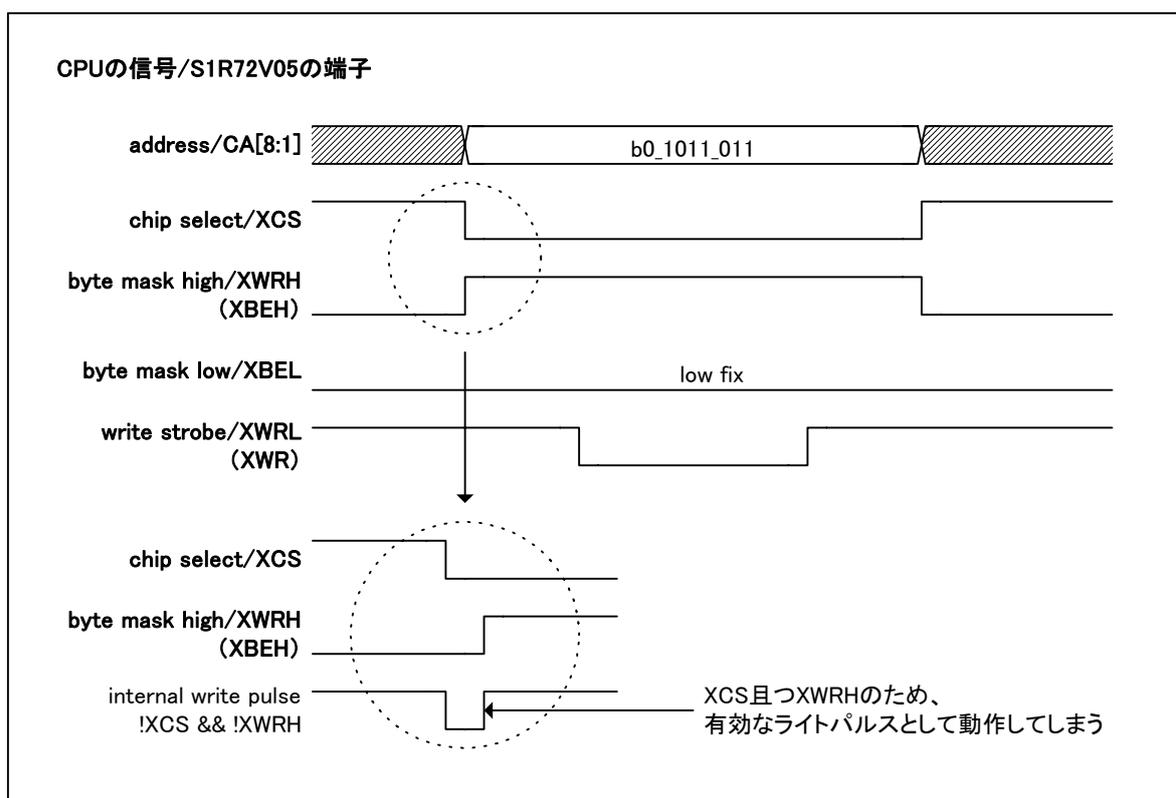


図 6-78 ChipConfig レジスタの初期設定

動作モードの設定完了後は内部的なライトパルス (internal write pulse) の生成条件が更新されますので、この様な制限はございません。

また、ChipConfigレジスタの設定前に、S1R72V05 にリードアクセスを行った場合、図 6-79 に示す通り、チップの内部ではリード動作とライト動作を同時に行ってしまいます。この場合の動作は保証されませんので、必ずChipConfigレジスタの設定を第一に行ってください。

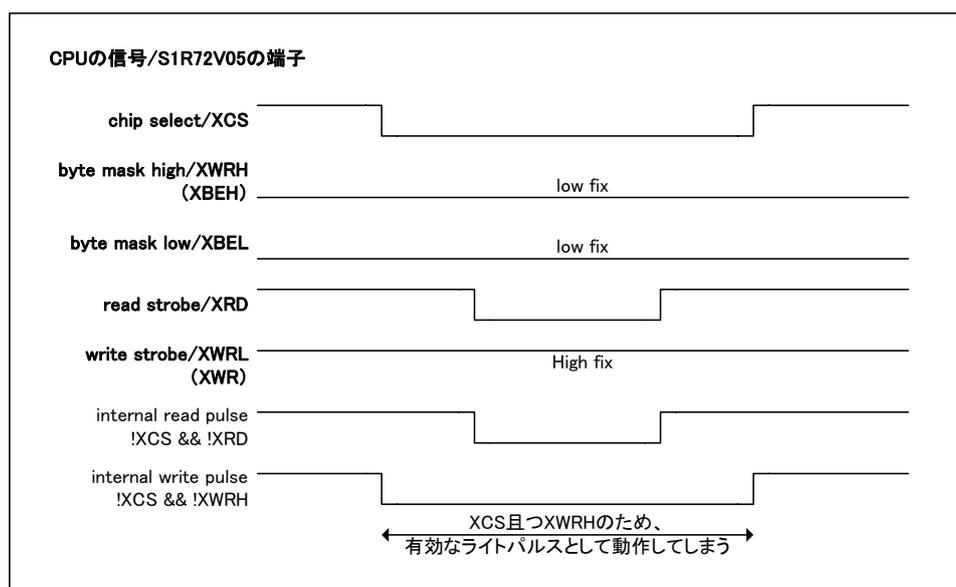


図 6-79 ChipConfig レジスタの初期設定前のリードアクセス

6.7.2.2. 8bit mode をご使用の場合

ご使用のCPUに合わせて 8bit modeを使用する場合、6.7.1 に示す通り、ChipConfigレジスタへの設定を第一に行ってください。ChipConfigレジスタの設定の前に、S1R72V05 にリードアクセスを行った場合、S1R72V05 は初期状態の 16bit Strobe modeで動作しておりますので、CD[15:0]全端子が出力状態になります。CD[15:8]をPull Up/Pull Down処理されている場合は特に問題ありませんが、これらの端子をVDD/GNDに直結している場合、電流消費が大幅に増大します。これを回避する為、必ずChipConfigレジスタの設定を第一に行ってください。

6.7.3. ブロック構成

S1R72V05 CPUIF(以下、CPUIF)のブロック構成を図 6-80 に示します。

REG/DMA0/DMA1 の 3 ブロックから構成されます。

- ・ REG:S1R72V05 レジスタ領域へのアクセス制御
- ・ DMA0:DMA チャンネル 0
- ・ DMA1:DMA チャンネル 1

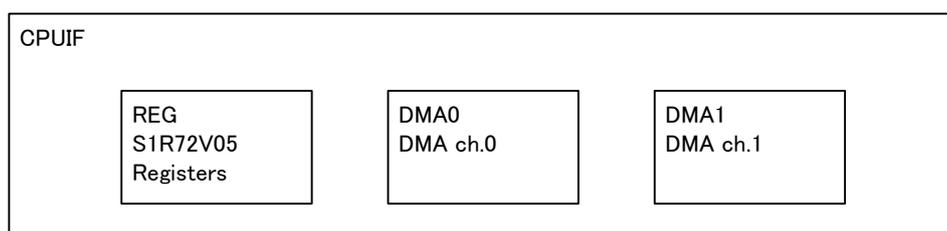


図 6-80 ブロック構成

6.7.3.1. REG (S1R72V05 Registers)

S1R72V05 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

6.7.3.1.1. 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。

6.7.3.1.2. 同期レジスタアクセス (リード)

リード(XCS、XRD 共にアサート)期間を出カイン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 (8bit mode の場合 2 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します。

6.7.3.1.3. FIFO アクセス (ライト)

FIFO ライトアクセスは FIFO_Wr_0,1、RAM_WrDoor_0,1 レジスタへの書き込みを意味します。8bit mode 動作の場合は FIFO_Wr_0,1レジスタのどちらにアクセスを行ってもFIFO への書き込みを行うことができます。RAM_WrDoor_0,1 に関しても同様です。

FIFO アクセス(ライト)には以下の制限事項があります。

- ・ D_EPx(x=0,a-c)Join.JoinCPU_Wr ビット、H_CHx(x=0, a-e)Join.JoinCPU_Wr ビット、MediaFIFO_Join.JoinCPU_Wr のいずれかを設定した後、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ください。RAM_WrDoor_0,1 レジスタにはこの制限はございません。
- ・ 16bit CPUを使用の場合は、基本的にワード(2バイト)単位でアクセスを行ってください。端数(奇数)バイトの書き込みを行う場合は FIFO のバイト境界を意識してストロブ信号を制御してください。詳細は”FIFO アクセスの端数処理”をご参照下さい。

- FIFO_Wr_0,1 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 1CPU サイクル以上の間隔を空けて確認してください。
- 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 1CPU サイクル以上の間隔を空けて確認してください。

6.7.3.1.4. 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。8bit mode 動作の場合は FIFO_Rd_0,1 レジスタ、FIFO_ByteRd レジスタのどのレジスタにアクセスを行っても同様に FIFO からの読み出しを行うことができます。

FIFO リードアクセスには、以下の制限事項があります。

- D_EPx{x=0,a-c}Join.JoinCPU_Wr ビット、H_CHx{x=0, a-e}Join.Join_Wr ビット、MediaFIFO_Join.JoinCPU_Wr のいずれかを設定した後、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。
- 16bit mode で使用の場合、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 FIFO アクセスの端数処理をご参照下さい。

6.7.3.1.5. 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。実際のFIFOは4byte幅ですが、この章の説明では簡易化のため2byte幅で表記致します。4byte/2byteによる動作の相違はありません。

【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。

D_EPnControl.EP0FIFO_ClrビットやD_EPrFIFO_Clr.EPx{x=a-c}FIFO_Clrビット(デバイス時)、又はH_CHnControl.CH0FIFO_ClrビットやH_CHrFIFO_Clr.CHx{x=a-e}FIFO_Clrビット(ホスト時)、又はMediaFIFO_Control.MediaFIFO_Clrビット(メディアデータ転送時)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6-81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6-81の(2)に示します。

以上は、正常な書き込み動作です。

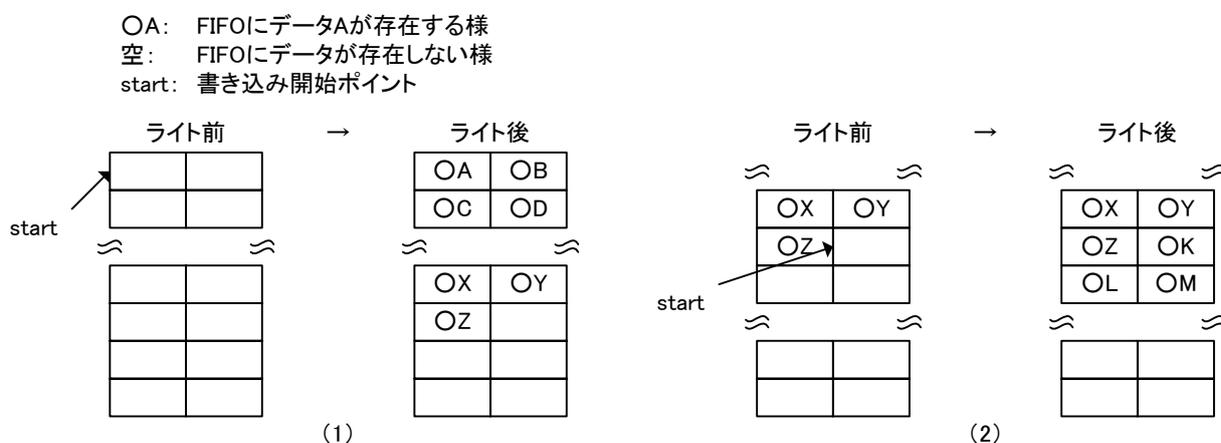


図 6-81 FIFO ライト処理(正常動作)

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6-82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6-82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6-82の(5))。また、FIFOにバイト境界が無く、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6-82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。

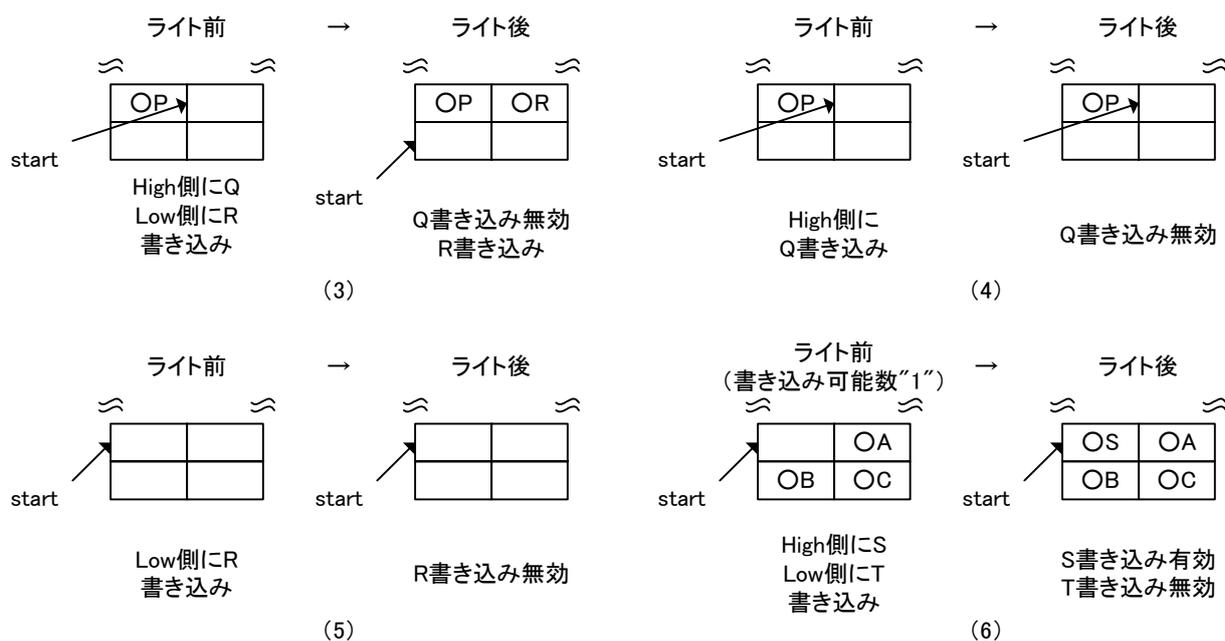


図 6-82 FIFO ライト処理(注意が必要な動作)

【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 6-83 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 6-83 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。

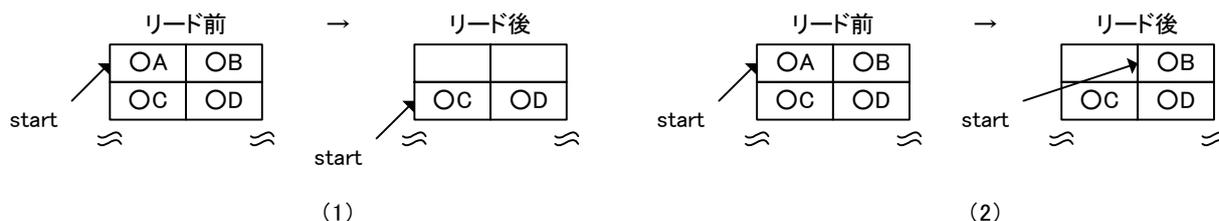


図 6-83 FIFO リード処理(正常動作)

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 6-84 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 6-84 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。

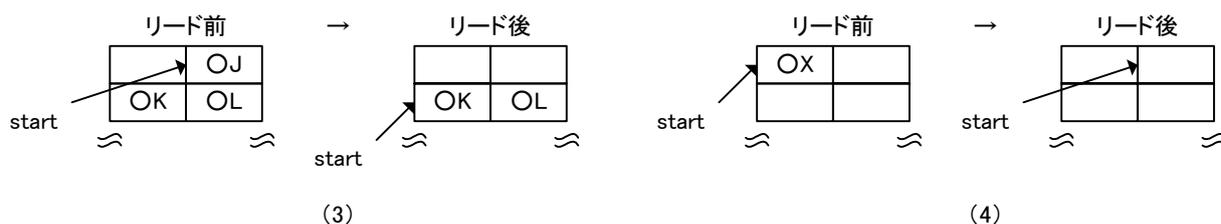


図 6-84 FIFO リード処理(注意が必要な動作)

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。

6.7.3.1.6. 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)期間を出力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 FIFO へのアクセス方法(RAM_Rd)をご参照下さい。

6.7.3.1.7. 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。

6.7.3.1.8. 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。

6.7.3.2. DMA0/DMA1 (DMA ch.0 / ch.1)

6.7.3.2.1. 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。

【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 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 DMAx{x=0,1}_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_x{x=0,1}_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。

16bit mode の場合、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出来ます。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 6-63 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(16bit/8bitモード動作時共)

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(16bit/8bitモード動作時共)

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(16bitモード動作時)

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(16bitモード動作時)

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)	-

カウントモード・ReqAssertCountオプション未使用時(8bitモード動作時)

条件	カウントモード(Count>0)		
	Count ≥ Ready		Count < Ready
	Ready ≥ 1	Ready < 1	Ready ≥ Count
XDREQ	アサート	ネゲート	アサート
転送可能数	Ready	-	Count

フリーランモード・ReqAssertCountオプション未使用時(8bitモード動作時)

条件	フリーランモード	
	-	
	Ready ≥ 1	Ready < 1
XDREQ	アサート	ネゲート
転送可能数	Ready	-

表中の Req は DMAx{x=0,1}Config.ReqAssertCount の設定値を、Ready は FIFO の空き領域/データ数を、Count は DMAx{x=0,1} Count_HH,HL,LH,LL の値を指しています。

6.7.3.2.2. 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。

6.7.3.2.3. カウントモード (ライト)

【動作開始】

DMAx{x=0,1}_Count_HH,HL,LH,LL レジスタにカウント値を設定した後、DMAx{x=0,1}_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(8bit mode の場合 1 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 DMAx{x=0,1}_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。

- ・ DMAx{x=0,1}_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
- ・ DMAx{x=0,1}_Control.DMA_Stop ビットに"1"をライト

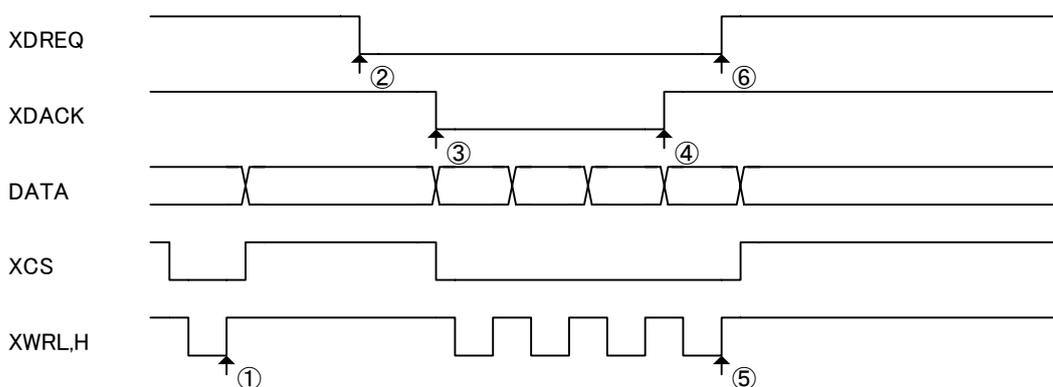
DMA 動作停止時に CPU_IntStat.DMAx{x=0,1}_Cmp ビットがセットされます。

DMAx{x=0,1}_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 DMAx{x=0, 1}_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 6-85 に示します。

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 6-85 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 6-86 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

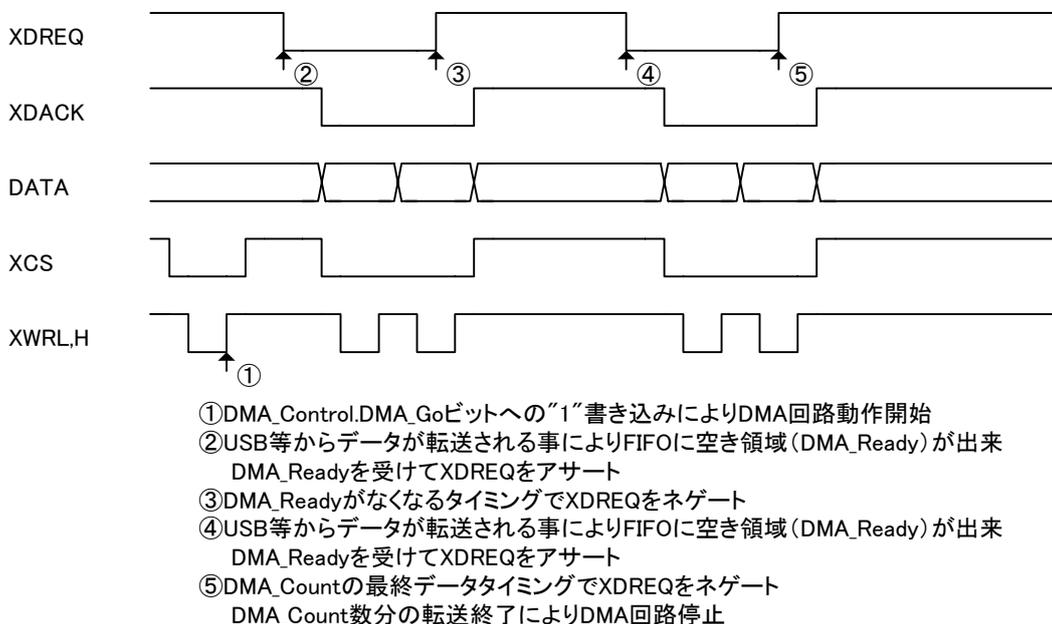


図 6-86 カウントモードライトタイミング 2

6.7.3.2.4. カウントモード (リード)

【動作開始】

DMAx{x=0,1}_Count_HH,HL,LH,LL レジスタにカウント値を設定した後、DMAx{x=0,1}_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(8bit mode動作時は1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DMAx{x=0,1}_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行って下さい。

動作停止となるまでDMAx{x=0,1}_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。

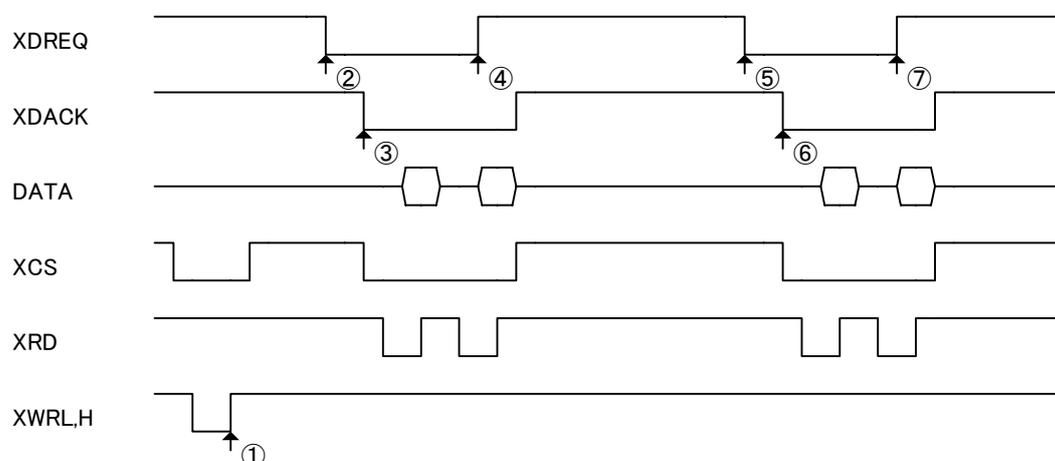
- DMAx{x=0,1}_Count_HH,HL,LH,LLレジスタに設定したカウント数分のDMA転送が完了
- DMAx{x=0,1}_Control.DMA_Stopビットに“1”をライト

DMAx{x=0,1}_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 6-87 に示します。

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



- 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FIFOにデータが書き込まれ、外部からデータ読み出し可能になることでXDREQをアサート
- ③ XDACKがアサートされDMA転送開始
- ④ FIFOデータが空になるタイミングでXDREQネゲート
- ⑤ USB等からFIFOにデータが書き込まれ、外部からデータ読み出し可能になることでXDREQをアサート
- ⑥ XDACKがアサートされDMA転送開始
- ⑦ DMA_Countの最終データタイミングでXDREQネゲート

図 6-87 カウントモードリードタイミング

6.7.3.2.5. フリーランモード（ライト）

【動作開始】

DMAx{x=0,1}_Config.FreeRun ビットをセットした後、DMAx{x=0,1}_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 (8bit mode 動作時は 1 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。転送を行う場合はカウントモードの説明をご参照ください。

動作停止となるまで DMAx{x=0,1}_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。

- ・ DMAx{x=0,1}_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 DMAx{x=0,1}_Count_HH,HL,LH,LL レジスタの値がオーバーフローすると、CPU_IntStat.DMAx{x=0,1}_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、DMAx{x=0,1}_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 DMAx{x=0,1}_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。

6.7.3.2.6. フリーランモード（リード）

【動作開始】

DMAx{x=0,1}_Config.FreeRun ビットをセットした後、DMAx{x=0,1}_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 (8bit mode 動作時は 1 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。転送を行う場合はカウントモードの説明をご参照ください。

動作停止となるまで DMAx{x=0,1}_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。

- ・ DMAx{x=0,1}_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 DMAx{x=0,1}_Count_HH,HL,LH,LL レジスタの値がオーバーフローすると、CPU_IntStat.DMAx{x=0,1}_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、DMAx{x=0,1}_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 DMAx{x=0,1}_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。

6.7.3.2.7. REQ アサートカウントオプション（ライト）

【動作開始】

DMAx{x=0,1}_Config.ReqAssertCount[1:0] ビットでアサートカウント数を設定した後、DMAx{x=0,1}_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。

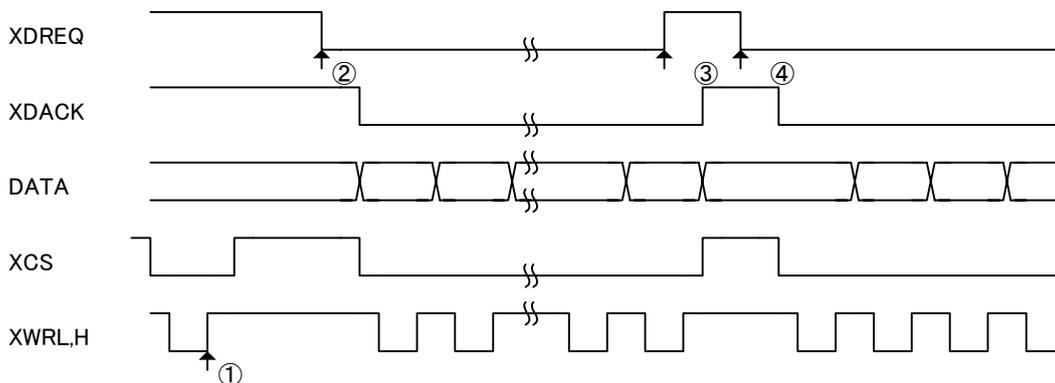
このモードでは、ReqAssertCount[1:0] ビットに設定された転送数毎に一旦 XDREQ がネゲートされます。

動作停止となるまで DMAx{x=0,1}_Control.DMA_Running ビットに"1"が読めます。

【動作停止】

動作停止条件はカウントモード、フリーランモードの説明をご参照ください。

ex.【転送開始】REQアサートカウント(8beat:16byte)



- ①DMA_Control.DMA_Goビットへの“1”書き込みによりDMA回路動作開始
DMA_Readyの値が連続転送数に満たないためXDREQはアサートしない
- ②USB等からデータが転送され、FIFOに連続転送数以上の有効な空き領域(DMA_Ready)が出来
DMA_Readyを受けてXDREQをアサート
- ③連続転送数(REQアサートカウント)終了のタイミングでXDREQネゲート
- ④1回目の連続転送終了時点で、次の連続転送分の空き領域(DMA_Ready)がある
DMA_Readyを受けてXDREQをアサート

図 6-88 REQ アサートカウントオプションライトタイミング

6.7.3.2.8. REQ アサートカウントオプション (リード)

【動作開始】

DMAx{x=0,1}_Config.ReqAssertCount[1:0]ビットでアサートカウント数を設定した後、DMAx{x=0,1}_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となります。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DMAx{x=0,1}_Control.DMA_Runningビットに“1”が読めます。

【動作停止】

動作停止条件はカウントモード、フリーランモードの説明をご参照ください。

動作タイミングは図 6-87、図 6-88をご参照下さい。

6.7.3.2.9. DMA の FIFO アクセス端数処理

6.7.3.1.5“FIFOアクセスの端数処理”をご参照ください。DMAIにはバイト読み出しの口はございませんのでご注意ください。

6.8. IDE I/F

ここでは、IDE I/F 機能について説明します。

IDE I/F 機能を使用する際は、IDE_Config_1.ActiveIDE ビットにあらかじめ"1"を書き込む必要があります。

6.8.1. IDE タスクファイルレジスタへのアクセス

IDE タスクファイルレジスタへのアクセス方法について説明します。

ファームウェアは IDE_RegAdrs レジスタから IDE_RegConfig レジスタまでのレジスタ群を介して、IDE タスクファイルレジスタにアクセスすることが出来ます。IDE バス上のリードライトシーケンスは、動作完了まで LSI のハードウェアが行うため、ファームウェアはコマンドをセットした後ポーリングまたは割り込みでシーケンス終了が通知されるまで、この IDE タスクファイルレジスタへのアクセス動作から解放されません。

IDE タスクファイルレジスタへアクセスする際には、IDE バスの XHCS0=0/HDA[2:0]=0 の時は IDE_Tmod レジスタの設定値が、それ以外のアドレスでは IDE_Rmod レジスタの設定値が使われて転送が行われるため、IDE の転送モードに応じて IDE_Rmod/IDE_Tmod レジスタにあらかじめ適切な値を設定する必要があります。

6.8.1.1. IDE タスクファイルレジスタからのリード

IDE_RegAdrs.IDE_RegAddress[3:0]ビットにアクセスしたいアドレスをあらかじめまたは同時にセットし、IDE_RegAdrs.IDE_RdReg ビットに"1"をライトすることにより、IDE タスクファイルレジスタからのリード動作が行われます。

IDE_RegAdrs.IDE_RdReg ビットが"0"になり、かつ IDE_IntStat.IDE_RegCmp ビットが"1"になりリード動作が終了します。

IDE タスクファイルレジスタからリードされたデータは IDE_RdRegValue レジスタに格納されます。

6.8.1.2. IDE タスクファイルレジスタへのライト

IDE_WrRegValue レジスタにあらかじめライトしたいデータをセットし、IDE_RegAdrs.IDE_RegAddress[3:0]ビットにアクセスしたいアドレスをあらかじめまたは同時にセットし、IDE_RegAdrs.IDE_WrReg ビットに"1"をライトすることにより、IDE タスクファイルレジスタへのライト動作が行われます。

IDE_RegAdrs.IDE_WrReg ビットが"0"になり、かつ IDE_IntStat.IDE_RegCmp ビットが"1"になりライト動作が終了します。

6.8.1.3. IDE タスクファイルレジスタへのシーケンシャルライト

IDE_SeqWrRegControl レジスタを使うことにより、あらかじめセットしておいた最大 16 組のアドレス・データで IDE タスクファイルレジスタへシーケンシャルライトすることが出来ます。

あらかじめ IDE_SeqWrRegAdrs.IDE_SeqWrRegAddress[3:0]ビットにライトしたいアドレスを、IDE_SeqWrRegValue レジスタにライトしたいバイトデータを対で最大 16 組ライトします。XHCS0=0/HDA[2:0]=0 のアドレスの時はワードアクセスになるため、下位/上位の順にデータを 2 回書き込む必要があり、アドレス・データ対は 2 組使用されます。その後 IDE_SeqWrRegControl.IDE_SeqWrReg ビットに"1"を書き込むとシーケンシャルライト動作が行われます。IDE_SeqWrRegControl.IDE_SeqWrReg ビットが"0"になり、かつ IDE_IntStat.IDE_SeqWrRegCmp ビットが"1"になりシーケンシャルライト動作が終了します。

シーケンシャルライト動作の起動前に、あらかじめセットしておいたアドレス・データが不要になった場合は、IDE_SeqWrRegControl.IDE_SeqWrRegClr ビットに"1"をライトしてアドレス・データを破棄することが出来ます。

シーケンシャルライト動作中にファームウェアが通常の IDE タスクファイルレジスタのリードライト動作を行うと、IDE_IntStat.IDE_RegErr ビットが"1"になり、そのリードライトは無視されます。

6.8.1.4. IDE タスクファイルレジスタからのオートステータスレジスタリード

IDE バスの HINTRQ がセットされた場合、IDE_RegConfig.EnAutoStsRd ビットに"1"が書き込まれていると、LSI は自動で IDE ステータスレジスタ (XHCS0=0,HDA[2:0]=7) をリードし、リードしたデータを IDE_RdRegValue レジスタに格納した後、IDE_IntStat.IDE_CompleteINTRQ ビットが"1"になります。

オートステータスレジスタリード動作からファームウェアが IDE_RdRegValue_1 レジスタを読み出すまでの期間に、ファームウェアが通常の IDE タスクファイルレジスタのリードライト動作を行うと、IDE_IntStat.IDE_RegErr ビットが"1"になり、そのリードライトは無視されます。

6.8.2. PIO アクセス

PIO モードによる DMA 機能について説明します。

PIO モードによる DMA では、IDE_Tmod レジスタの設定値が使われて DMA 転送が行われるため、IDE の転送モードに応じて IDE_Tmod レジスタにあらかじめ適切な値を設定する必要があります。

6.8.2.1. PIO リード DMA

PIO リード DMA は以下の手順で動作します。

IDE_Config_0.DMA ビット及び IDE_Config_0.Ultra ビット両方に"0"を書き込みます。

IDE_CountH/M/L レジスタに転送バイト数を設定します。

IDE_Control.IDE_Clr ビットに"1"を書き込み、IDE 回路を初期状態にします。(必ずしも必要ではありません)

IDE_Control.IDE_Go ビットに"1"を書き込むと DMA 動作が開始され、転送とともに IDE_CountH/M/L レジスタの内容が減っていき、カウントが 0 になって IDE の DMA 動作と IDE からリードしたデータの内部 FIFO への書き込みが全て終了すると、IDE_Control.IDE_Go ビットが"0"になり、かつ IDE_IntStat.IDE_Cmp ビットが"1"になり DMA 動作が終了します。

DMA 転送中に IDE_Control.IDE_Go ビットに"0"を書き込むと DMA 動作が中断されて DMA が終了します。その時内部 FIFO まで転送したデータは有効ですが、回路の中間バッファに溜まったデータは捨てられるため、IDE の転送バイト数を正確に管理することは出来ません。

6.8.2.2. PIO ライト DMA

PIO ライト DMA は以下の手順で動作します。

IDE_Config_0.DMA ビット及び IDE_Config_0.Ultra ビット両方に"0"を書き込みます。

IDE_CountH/M/L レジスタに転送バイト数を設定します。

IDE_Control.IDE_Clr ビットに"1"を書き込み、IDE 回路を初期状態にします。(必ずしも必要ではありません)

IDE_Control.IDE_Go ビットに"1"を書き込むと DMA 動作が開始され、転送とともに IDE_CountH/M/L レジスタの内容が減っていき、カウントが 0 になって IDE への DMA ライトが全て終了すると、IDE_Control.IDE_Go ビットが"0"になり、かつ IDE_IntStat.IDE_Cmp ビットが"1"になり DMA 動作が終了します。

DMA 転送中に IDE_Control.IDE_Go ビットに"0"を書き込むと DMA 動作が中断されて DMA が終了します。その時まで IDE にライトしたデータは有効ですが、回路の中間バッファに溜まったデータは捨てられるため、IDE の転送バイト数を正確に管理することは出来ません。

6.8.3. Multi-Word DMA

Multi-Word DMA モードによる DMA 機能について説明します。

Multi-Word DMA モードによる DMA では、IDE_Tmod レジスタの設定値が使われて DMA 転送が行われるため、IDE の転送モードに応じて IDE_Tmod レジスタにあらかじめ適切な値を設定する必要があります。

6.8.3.1. Multi-Word DMA リード

Multi-Word DMA リードは以下の手順で動作します。

IDE_Config_0.DMA ビットに"1"を、IDE_Config_0.Ultra ビットに"0"を書き込みます。

IDE_Config_1.DelayStrobe ビット、IDE_Config_1.InterLock ビットに適切な値を書き込みます。

IDE_CountH/M/L レジスタに転送バイト数を設定します。

IDE_Control.IDE_Clr ビットに"1"を書き込み、IDE 回路を初期状態にします。(必ずしも必要ではありません)

IDE_Control.IDE_Go ビットに"1"を書き込むと DMA 動作が開始され、転送とともに IDE_CountH/M/L レジスタの内容が減っていき、カウントが 0 になって IDE の DMA 動作と IDE からリードしたデータの内部 FIFO への書き込みが全て終了すると、IDE_Control.IDE_Go ビットが"0"になり、かつ IDE_IntStat.IDE_Cmp ビットが"1"になり DMA 動作が終了します。

DMA 転送中に IDE_Control.IDE_Go ビットに"0"を書き込むと DMA 動作が中断されて DMA が終了します。その時内部 FIFO まで転送したデータは有効ですが、回路の中間バッファに溜まったデータは捨てられるため、IDE の転送バイト数を正確に管理することは出来ません。

6.8.3.2. Multi-Word DMA ライト

Multi-Word DMA ライトは以下の手順で動作します。

IDE_Config_0.DMA ビットに"1"を、IDE_Config_0.Ultra ビットに"0"を書き込みます。

IDE_Config_1.DelayStrobe ビット、IDE_Config_1.InterLock ビットに適切な値を書き込みます。

IDE_CountH/M/L レジスタに転送バイト数を設定します。

IDE_Control.IDE_Clr ビットに"1"を書き込み、IDE 回路を初期状態にします。(必ずしも必要ではありません)

IDE_Control.IDE_Go ビットに"1"を書き込むと DMA 動作が開始され、転送とともに IDE_CountH/M/L レジスタの内容が減っていき、カウントが 0 になって IDE への DMA ライトが全て終了すると、IDE_Control.IDE_Go ビットが"0"になり、かつ IDE_IntStat.IDE_Cmp ビットが"1"になり DMA 動作が終了します。

DMA 転送中に IDE_Control.IDE_Go ビットに"0"を書き込むと DMA 動作が中断されて DMA が終了します。その時まで IDE にライトしたデータは有効ですが、回路の中間バッファに溜まったデータは捨てられるため、IDE の転送バイト数を正確に管理することは出来ません。

6.8.4. Ultra DMA

Ultra DMA モードによる DMA 機能について説明します。

Ultra DMA モードによる DMA では、IDE_Umod レジスタの設定値が使われて DMA 転送が行われるため、IDE の転送モードに応じて IDE_Umod レジスタにあらかじめ適切な値を設定する必要があります。

6.8.4.1. Ultra DMA リード

Ultra DMA リードは以下の手順で動作します。

IDE_Config_0.DMA ビット及び IDE_Config_0.Ultra ビットに"1"を書き込みます。

IDE_CountH/ML レジスタに転送バイト数を設定します。

IDE_Control.IDE_Clr ビットに"1"を書き込み、IDE 回路を初期状態にします。(必ずしも必要ではありません)

IDE_Control.IDE_Go ビットに"1"を書き込むと DMA 動作が開始され、転送とともに IDE_CountH/ML レジスタの内容が減っていき、カウンタが 0 になって IDE の DMA 動作と IDE からリードしたデータの内部 FIFO への書き込みが全て終了すると、IDE_Control.IDE_Go ビットが"0"になり、かつ IDE_IntStat.IDE_Cmp ビットが"1"になり DMA 動作が終了します。

DMA 転送中に IDE_Control.IDE_Go ビットに"0"を書き込むと DMA 動作が中断されて DMA が終了します。その時内部 FIFO まで転送したデータは有効ですが、回路の中間バッファに溜まったデータは捨てられるため、IDE の転送バイト数を正確に管理することは出来ません。また、この場合接続された IDE デバイスにはホスターミネートを通知します。

DMA 転送中にデバイスターミネートが発生した場合、IDE_IntStat.DetectTerm ビットが"1"になり接続された IDE デバイスは転送を止めてしまいますが、起動した DMA は終了しませんので IDE_Control.IDE_Go ビットに"0"を書き込んで DMA を終了させる必要があります。

6.8.4.2. Ultra DMA ライト

Ultra DMA ライトは以下の手順で動作します。

IDE_Config_0.DMA ビット及び IDE_Config_0.Ultra ビットに"1"を書き込みます。

IDE_CountH/ML レジスタに転送バイト数を設定します。

IDE_Control.IDE_Clr ビットに"1"を書き込み、IDE 回路を初期状態にします。(必ずしも必要ではありません)

IDE_Control.IDE_Go ビットに"1"を書き込むと DMA 動作が開始され、転送とともに IDE_CountH/ML レジスタの内容が減っていき、カウンタが 0 になって IDE への DMA ライトが全て終了すると、IDE_Control.IDE_Go ビットが"0"になり、かつ IDE_IntStat.IDE_Cmp ビットが"1"になり DMA 動作が終了します。

DMA 転送中に IDE_Control.IDE_Go ビットに"0"を書き込むと DMA 動作が中断されて DMA が終了します。その時まで IDE にライトしたデータは有効ですが、回路の中間バッファに溜まったデータは捨てられるため、IDE の転送バイト数を正確に管理することは出来ません。また、この場合接続された IDE デバイスにはホスターミネートを通知します。

DMA 転送中にデバイスターミネートが発生した場合、IDE_IntStat.DetectTerm ビットが"1"になり接続された IDE デバイスは転送を止めてしまいますが、起動した DMA は終了しませんので IDE_Control.IDE_Go ビットに"0"を書き込んで DMA を終了させる必要があります。

6.8.5. IDE 転送モードの設定について

IDE 転送の各モードにする設定値を下表に示します。

- Register Mode (XHCS0=0,HDA=0 以外のファームウェアによる IDE タスクファイルレジスタへのアクセス時)

Mode	レジスタ設定値(必須)		レジスタ設定値(推奨)
	IDE_Config_0.Ultra	IDE_Config_0.DMA	IDE_Rmod
0	影響しない	影響しない	FFh
1	影響しない	影響しない	F1h
2	影響しない	影響しない	F0h
3	影響しない	影響しない	22h
4	影響しない	影響しない	10h

- PIO Mode (XHCS0=0,HDA=0 のファームウェアによる IDE タスクファイルレジスタへのアクセス時)

Mode	レジスタ設定値(必須)		レジスタ設定値(推奨)
	IDE_Config_0.Ultra	IDE_Config_0.DMA	IDE_Tmod
0	影響しない	影響しない	FFh
1	影響しない	影響しない	88h
2	影響しない	影響しない	44h
3	影響しない	影響しない	22h
4	影響しない	影響しない	10h

- PIO Mode (DMA 転送時)

Mode	レジスタ設定値(必須)		レジスタ設定値(推奨)
	IDE_Config_0.Ultra	IDE_Config_0.DMA	IDE_Tmod
0	0	0	FFh
1	0	0	88h
2	0	0	44h
3	0	0	22h
4	0	0	10h

- Multi-word DMA Mode (DMA 転送時)

Mode	レジスタ設定値(必須)		レジスタ設定値(推奨)
	IDE_Config_0.Ultra	IDE_Config_0.DMA	IDE_Tmod
0	0	1	BBh
1	0	1	20h
2	0	1	10h

- Ultra Mode (DMA 転送時、DATA-OUT 時)

Mode	レジスタ設定値(必須)		レジスタ設定値(推奨)
	IDE_Config_0.Ultra	IDE_Config_0.DMA	IDE_Umod
0	1	1	06h
1	1	1	04h
2	1	1	03h
3	1	1	02h
4	1	1	01h
5	1	1	00h

注:Ultra Mode において IDE バスからデータを入力する DATA-IN 時は、IDE_Umod レジスタの設定によらず全てのモードでデータ受信可能です。

6.9. バウンダリスキャン (JTAG)

バウンダリスキャン(JTAG)は、TEST 端子をL(デフォルト)の時使用可能です。バウンダリスキャンは、JTAG(IEEE 1149.1)仕様に準拠した、BSR(Boundary Scan Register)、それを繋ぐスキャンパスおよび TAP コントローラで構成されます。バウンダリスキャンの接続情報は、BSDL フォーマットにて提供可能です。

6.9.1. 対応インストラクション

本 LSI の JTAG インストラクションビット幅は 4 ビットとなっており、以下の JTAG インストラクションに対応致します。

表 6-64 JTAG インストラクションコード

インストラクション	説明	コード
SAMPLE/PRELOAD	LSI 内部状態の BSR への取り込み及びデータ設定。	0010
BYPASS	BSR によるスキャンパスをバイパスする。	1111
EXTEST	デバイスの物理的な接続チェック。	0000
CLAMP	出力値を保持しながら、スキャンパスをバイパスする。	0011
HIGHZ	出力を全て Hi-Z に固定する。	0100
IDCODE	定められた DEVICE_CODE の出力。	0001

6.9.2. DEVICE_CODE に関して

IDCODE インストラクションに対する DEVICE_CODE の構成要素は以下のとおりとなります。

表 6-65 DEVICE_CODE

Version	1
Part Number	000F
Manufacturer	0x0BE

従って、IDCODE インストラクションに対する DEVICE_CODE 応答は、

0001_0000000000001111_00010111110_1

となります。

6.9.3. バウンダリスキャン除外端子

本 LSI の端子のうち、DP_A、DM_A、DP_B、DM_B、XI、XO、VBUS_B、R1_A、R1_B、および、TEST にはバウンダリスキャンセルが挿入されていないため、スキャン対象外となります。

7. レジスタ

レジスタは、デバイス/ホスト共通レジスタ、デバイス・レジスタ、ホスト・レジスタに分類され、デバイス・レジスタとホスト・レジスタとは HostDeviceSel.HOSTxDEVICE ビットの設定で、マップの切替を行います。同ビットが“0”のときにデバイス・レジスタが選択され、“1”のときにホスト・レジスタが選択されます。同ビットの設定を変えても、レジスタの設定値はクリアされません。

リザーブレジスタ・ビットには“1”を書き込まないでください。

7.1. デバイス/ホスト共通レジスタマップ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読み書きできます(注)。

(注)ただし、0x40~0x47, 0x50~0x6F のレジスタは、ACTIVE60 時は読み書きできません。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x00	MainIntStat	R(W)	0x00	DeviceIntStat	HostIntStat	CPU_IntStat	IDE_IntStat	MediaFIFO_IntStat			FinishedPM
0x01	DeviceIntStat	R(W)	0x00	VBUS_Changed		D_SIE_IntStat	D_BulkIntStat	RcvEP0SETUP	D_FIFO_IntStat	D_EP0IntStat	D_EP1IntStat
0x02	HostIntStat	R(W)	0x00	VBUS_Err	LineStateChanged	H_SIE_IntStat_1	H_SIE_IntStat_0	H_FrameIntStat	H_FIFO_IntStat	H_CH0IntStat	H_CH1IntStat
0x03	CPU_IntStat	R(W)	0x00	RAM_RdCmp				DMA1_Countup	DMA1_Cmp	DMA0_Countup	DMA0_Cmp
0x04	IDE_IntStat	R(W)	0x00	IDE_RegCmp	IDE_RegErr	IDE_SeqWrRegCmp	CompleteINTRQ		IDE_Cmp	DetectINTRQ	DetectTerm
0x05	MediaFIFO_IntStat	R(W)	0x00		MediaIDE_Cmp				FIFO_NotEmpty	FIFO_Full	FIFO_Empty
0x06			0xFF								
0x07			0xFF								
0x08			0xFF								
0x09			0xFF								
0x0A			0xFF								
0x0B			0xFF								
0x0C			0xFF								
0x0D			0xFF								
0x0E			0xFF								
0x0F			0xFF								

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x10	MainIntEnb	RW	0x00	EnDeviceIntStat	EnHostIntStat	EnCPU_IntStat	EnIDE_IntStat	EnMediaFIFO_IntStat			EnFinishedPM
0x11	DeviceIntEnb	RW	0x00	EnVBUS_Changed		EnD_SIE_IntStat	EnD_BulkIntStat	EnRcvEP0SETUP	EnD_FIFO_IntStat	EnD_EP0IntStat	EnD_EP1IntStat
0x12	HostIntEnb	RW	0x00	EnVBUS_Err	EnLineStateChanged	EnH_SIE_IntStat_1	EnH_SIE_IntStat_0	EnH_FrameIntStat	EnH_FIFO_IntStat	EnH_CH0IntStat	EnH_CH1IntStat
0x13	CPU_IntEnb	RW	0x00	EnRAM_RdCmp				EnDMA1_Countup	EnDMA1_Cmp	EnDMA0_Countup	EnDMA0_Cmp
0x14	IDE_IntEnb	RW	0x00	EnIDE_RegCmp	EnIDE_RegErr	EnIDE_SeqWrRegCmp	EnCompleteINTRQ		EnIDE_Cmp	EnDetectINTRQ	EnDetectTerm
0x15	MediaFIFO_IntEnb	RW	0x00		EnMediaIDE_Cmp				EnFIFO_NotEmpty	EnFIFO_Full	EnFIFO_Empty
0x16			0xFF								
0x17			0xFF								
0x18			0xFF								
0x19			0xFF								
0x1A			0xFF								
0x1B			0xFF								
0x1C			0xFF								
0x1D			0xFF								
0x1E			0xFF								
0x1F			0xFF								

7. レジスタ

SLEEP / SNOOZE 時に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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読み書きできます(注)。

(注)ただし、0x40~0x47, 0x50~0x6F のレジスタは、ACTIVE60 時は読み書きできません。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x20	RevisionNum	R	0x50	RevisionNum								
0x21	ChipReset	W	0xFF	AllReset								
0x22	PM_Control_0	RW	0x00	GoSLEEP	GoSNOOZE	GoActive60	GoActDevice	GoActHost				
0x23	PM_Control_1	R	0x00								PM_State[3:0]	
0x24	WakeupTim_H	RW	0x00	WakeupTim[15:8]								
0x25	WakeupTim_L	RW	0x00	WakeupTim[7:0]								
0x26	H_USB_Control	RW	0x00	VBUS_Enb								
0x27	H_XcvtControl	RW	0x91	TermSelect	RemoveRPD	XcvtSelect[1:0]			OpMode[1:0]			
0x28	D_USB_Status	RW	0xFF	VBUS	FSxHS						LineState[1:0]	
0x29	H_USB_Status	R	0xFF	VBUS_State						LineState[1:0]		
0x2A			0xFF									
0x2B			0xFF									
0x2C			0xFF									
0x2D			0xFF									
0x2E			0xFF									
0x2F			0xFF									

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x30	FIFO_Rd_0	R	0xFF	FIFO_Rd_0[7:0]							
0x31	FIFO_Rd_1	R	0xFF	FIFO_Rd_1[7:0]							
0x32	FIFO_Wr_0	W	0xFF	FIFO_Wr_0[7:0]							
0x33	FIFO_Wr_1	W	0xFF	FIFO_Wr_1[7:0]							
0x34	FIFO_RdRemain_H	R	0x00	RdRemainValid						RdRemain[12:8]	
0x35	FIFO_RdRemain_L	R	0x00	RdRemain[7:0]							
0x36	FIFO_WrRemain_H	R	0x00						WrRemain[12:8]		
0x37	FIFO_WrRemain_L	R	0x00	WrRemain[7:0]							
0x38	FIFO_ByteRd	R	0xFF	FIFO_ByteRd[7:0]							
0x39			0xFF								
0x3A			0xFF								
0x3B			0xFF								
0x3C			0xFF								
0x3D			0xFF								
0x3E			0xFF								
0x3F			0xFF								

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読み書きできます(注)。

(注)ただし、0x40~0x47, 0x50~0x6F のレジスタは、ACTIVE60 時は読み書きできません。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x40	RAM_RdAdrs_H	RW	0x00					RAM_RdAdrs[12:8]				
0x41	RAM_RdAdrs_L	RW	0x00	RAM_RdAdrs[7:2]								
0x42	RAM_RdControl	RW	0x00	RAM_GoRdCBW _CSW	RAM_GoRd							
0x43	RAM_RdCount	RW	0x00	RAM_RdCount[5:2]								
0x44	RAM_WrAdrs_H	RW	0x00	RAM_WrAdrs[12:8]								
0x45	RAM_WrAdrs_L	RW	0x00	RAM_WrAdrs[7:0]								
0x46	RAM_WrDoor_0	W	0xFF	RAM_WrDoor_0[7:0]								
0x47	RAM_WrDoor_1	W	0xFF	RAM_WrDoor_1[7:0]								
0x48	MediaFIFO_Control	W	0xFF								MediaFIFO_Clr	
0x49	ClrAllMediaFIFO_Join	W	0xFF	ClrJoinIDE				ClrJoinDMA1	ClrJoinDMA0	ClrJoinCPU_Rd	ClrJoinCPU_Wr	
0x4A	MediaFIFO_Join	RW	0x00	JoinIDE				JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	
0x4B			0xFF									
0x4C			0xFF									
0x4D			0xFF									
0x4E			0xFF									
0x4F			0xFF									

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x50	RAM_Rd_00	R	0x00	RAM_Rd_00[7:0]								
0x51	RAM_Rd_01	R	0x00	RAM_Rd_01[7:0]								
0x52	RAM_Rd_02	R	0x00	RAM_Rd_02[7:0]								
0x53	RAM_Rd_03	R	0x00	RAM_Rd_03[7:0]								
0x54	RAM_Rd_04	R	0x00	RAM_Rd_04[7:0]								
0x55	RAM_Rd_05	R	0x00	RAM_Rd_05[7:0]								
0x56	RAM_Rd_06	R	0x00	RAM_Rd_06[7:0]								
0x57	RAM_Rd_07	R	0x00	RAM_Rd_07[7:0]								
0x58	RAM_Rd_08	R	0x00	RAM_Rd_08[7:0]								
0x59	RAM_Rd_09	R	0x00	RAM_Rd_09[7:0]								
0x5A	RAM_Rd_0A	R	0x00	RAM_Rd_0A[7:0]								
0x5B	RAM_Rd_0B	R	0x00	RAM_Rd_0B[7:0]								
0x5C	RAM_Rd_0C	R	0x00	RAM_Rd_0C[7:0]								
0x5D	RAM_Rd_0D	R	0x00	RAM_Rd_0D[7:0]								
0x5E	RAM_Rd_0E	R	0x00	RAM_Rd_0E[7:0]								
0x5F	RAM_Rd_0F	R	0x00	RAM_Rd_0F[7:0]								

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読み書きできます(注)。

(注)ただし、0x40~0x47, 0x50~0x6F のレジスタは、ACTIVE60 時は読み書きできません。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x60	RAM_Rd_10	R	0x00	RAM_Rd_10[7:0]							
0x61	RAM_Rd_11	R	0x00	RAM_Rd_11[7:0]							
0x62	RAM_Rd_12	R	0x00	RAM_Rd_12[7:0]							
0x63	RAM_Rd_13	R	0x00	RAM_Rd_13[7:0]							
0x64	RAM_Rd_14	R	0x00	RAM_Rd_14[7:0]							
0x65	RAM_Rd_15	R	0x00	RAM_Rd_15[7:0]							
0x66	RAM_Rd_16	R	0x00	RAM_Rd_16[7:0]							
0x67	RAM_Rd_17	R	0x00	RAM_Rd_17[7:0]							
0x68	RAM_Rd_18	R	0x00	RAM_Rd_18[7:0]							
0x69	RAM_Rd_19	R	0x00	RAM_Rd_19[7:0]							
0x6A	RAM_Rd_1A	R	0x00	RAM_Rd_1A[7:0]							
0x6B	RAM_Rd_1B	R	0x00	RAM_Rd_1B[7:0]							
0x6C	RAM_Rd_1C	R	0x00	RAM_Rd_1C[7:0]							
0x6D	RAM_Rd_1D	R	0x00	RAM_Rd_1D[7:0]							
0x6E	RAM_Rd_1E	R	0x00	RAM_Rd_1E[7:0]							
0x6F	RAM_Rd_1F	R	0x00	RAM_Rd_1F[7:0]							

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x70			0xXX								
0x71	DMA0_Config	RW	0x00	FreeRun	DMA_Mode			ActiveDMA		ReqAssertCount[1:0]	
0x72	DMA0_Control	RW	0x00	DMA_Running			CounterClr	Dir		DMA_Stop	DMA_Go
0x73			0xXX								
0x74	DMA0_Remain_H	R	0x00				DMA_Remain[12:8]				
0x75	DMA0_Remain_L	R	0x00	DMA_Remain[7:0]							
0x76			0xXX								
0x77			0xXX								
0x78	DMA0_Count_HH	RW	0x00	DMA_Count[31:24]							
0x79	DMA0_Count_HL	RW	0x00	DMA_Count[23:16]							
0x7A	DMA0_Count_LH	RW	0x00	DMA_Count[15:8]							
0x7B	DMA0_Count_LL	RW	0x00	DMA_Count[7:0]							
0x7C	DMA0_RdData_0	R	0xXX	DMA_RdData_0[7:0]							
0x7D	DMA0_RdData_1	R	0xXX	DMA_RdData_1[7:0]							
0x7E	DMA0_WrData_0	W	0xXX	DMA_WrData_0[7:0]							
0x7F	DMA0_WrData_1	W	0xXX	DMA_WrData_1[7:0]							

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読み書きできます(注)。

(注)ただし、0x40～0x47, 0x50～0x6F のレジスタは、ACTIVE60 時は読み書きできません。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8		
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0		
0x80			0xXX										
0x81	DMA1_Config	RW	0x00	FreeRun	DMA_Mode			ActiveDMA		ReqAssertCount[1:0]			
0x82	DMA1_Control	RW	0x00	DMA_Running			CounterClr	Dir		DMA_Stop	DMA_Go		
0x83			0xXX										
0x84	DMA1_Remain_H	R	0x00					DMA_Remain[12:8]					
0x85	DMA1_Remain_L	R	0x00	DMA_Remain[7:0]									
0x86			0xXX										
0x87			0xXX										
0x88	DMA1_Count_HH	RW	0x00	DMA_Count[31:24]									
0x89	DMA1_Count_HL	RW	0x00	DMA_Count[23:16]									
0x8A	DMA1_Count_LH	RW	0x00	DMA_Count[15:8]									
0x8B	DMA1_Count_LL	RW	0x00	DMA_Count[7:0]									
0x8C	DMA1_RdData_0	R	0xXX	DMA_RdData_0[7:0]									
0x8D	DMA1_RdData_1	R	0xXX	DMA_RdData_1[7:0]									
0x8E	DMA1_WrData_0	W	0xXX	DMA_WrData_0[7:0]									
0x8F	DMA1_WrData_1	W	0xXX	DMA_WrData_1[7:0]									

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x90	IDE_Status	R	0x00	DMARQ	DMACK	INTRQ	IORDY			PDIAG	DASP
0x91	IDE_Control	RW	0x00		IDE_Clr			Dir			IDE_Go
0x92	IDE_Config_0	RW	0x00	IDE_BusReset	IDE_LongBusReset					Ultra	DMA
0x93	IDE_Config_1	RW	0x04	ActiveIDE	DelayStrobe		InterLock		Swap		
0x94	IDE_Rmod	RW	0x00	RegisterAssertPulseWidth[3:0]				RegisterNegatePulseWidth[3:0]			
0x95	IDE_Tmod	RW	0x00	TransferAssertPulseWidth[3:0]				TransferNegatePulseWidth[3:0]			
0x96	IDE_Umod	RW	0x00					UltraDMA_Cycle[3:0]			
0x97			0xXX								
0x98			0xXX								
0x99			0xXX								
0x9A	IDE_CRC_H	R	0x00	IDE_CRC[15:8]							
0x9B	IDE_CRC_L	R	0x00	IDE_CRC[7:0]							
0x9C			0xXX								
0x9D	IDE_Count_H	RW	0x00	IDE_Count[23:16]							
0x9E	IDE_Count_M	RW	0x00	IDE_Count[15:8]							
0x9F	IDE_Count_L	RW	0x00	IDE_Count[7:1]							

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読み書きできます(注)。

(注)ただし、0x40~0x47, 0x50~0x6F のレジスタは、ACTIVE60 時は読み書きできません。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0xA0	IDE_RegAdrs	RW	0x00	IDE_WrReg	IDE_RdReg				IDE_RegAddress[3:0]		
0xA1			0xFF								
0xA2	IDE_RdRegValue_0	R	0x00	IDE_RdRegValue_0[7:0]							
0xA3	IDE_RdRegValue_1	R	0x00	IDE_RdRegValue_1[7:0]							
0xA4	IDE_WrRegValue_0	RW	0x00	IDE_WrRegValue_0[7:0]							
0xA5	IDE_WrRegValue_1	RW	0x00	IDE_WrRegValue_1[7:0]							
0xA6	IDE_SeqWrRegControl	RW	0x00	IDE_SeqWrReg	IDE_SeqWrRegCt						
0xA7	IDE_SeqWrRegCnt	R	0x00	IDE_SeqWrRegCnt[4:0]							
0xA8	IDE_SeqWrRegAdrs	W	0xFF					IDE_SeqWrRegAdrs[3:0]			
0xA9	IDE_SeqWrRegValue	W	0xFF	IDE_SeqWrRegValue[7:0]							
0xAA			0xFF								
0xAB			0xFF								
0xAC	IDE_RegConfig	RW	0x00	EnAutoStsRd							
0xAD			0xFF								
0xAE			0xFF								
0xAF			0xFF								

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0xB0			0xFF								
0xB1	<i>HostDeviceSel</i>	RW	0x00								<i>HOSTxDEVICE</i>
0xB2			0xFF								
0xB3	<i>ModeProtect</i>	RW	0x56	<i>ModeProtect[7:0](56 以外を書くとプロテクト、0x56 で解除)</i>							
0xB4			0xFF								
0xB5	<i>ClkSelect</i>	RW	0x41	<i>xActIDE_Term</i>	<i>xActIDE_DD_Term</i>					<i>PORT1x2</i>	<i>ClkSelect</i>
0xB6			0xFF								
0xB7	<i>ChipConfig</i>	RW	0x00	<i>IntLevel</i>	<i>IntMode</i>	<i>DREQ_Level</i>	<i>DACK_Level</i>	<i>CS_Mode</i>	<i>CPU_Endian</i>	<i>BusMode</i>	<i>Bus8x16</i>
0xB8			0xFF								
0xB9	<i>CPU_ChgEndian</i>	R	0xFF	<i>このレジスタをダミーリードすることで、ChipConfig.CPU_Endian で設定した Endian が有効になる</i>							
0xBA			0xFF								
0xBB			0xFF								
0xBC			0xFF								
0xBD			0xFF								
0xBE			0xFF								
0xBF			0xFF								

0xC0~0xDF は Rreserved です。

7.2. デバイス・レジスタマップ

SLEEP / SNOOZE 時に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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_DEVICE 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0xE0	<i>D_SIE_IntStat</i>	R(W)	0x00		<i>NonJ</i>	RcvSOF	DetectRESET	DetectSUSPEND	ChirpCmp	RestoreCmp	SetAddressCmp
0xE1			0xFF								
0xE2	D_FIFO_IntStat	R(W)	0x00	DescriptorCmp	FIFO_IDE_Cmp	FIFO1_Cmp	FIFO0_Cmp		FIFO_NotEmpty	FIFO_Full	FIFO_Empty
0xE3	D_BulkIntStat	R(W)	0x00	CBW_Cmp	CBW_LengthErr	CBW_Err		CSW_Cmp	CSW_Err		
0xE4	D_EPrintStat	R	0x00						EPintStat	EPIntStat	EPIntStat
0xE5	D_EP0IntStat	R(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0xE6	D_EPalntStat	R(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0xE7	D_EPblntStat	R(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0xE8	D_EPcIntStat	R(W)	0x00		OUT_ShortACK	IN_TranACK	OUT_TranACK	IN_TranNAK	OUT_TranNAK	IN_TranErr	OUT_TranErr
0xE9			0xFF								
0xEA			0xFF								
0xEB			0xFF								
0xEC			0xFF								
0xED			0xFF								
0xEE			0xFF								
0xEF			0xFF								

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0xF0	<i>D_SIE_IntEnb</i>	RW	0x00		<i>EnNonJ</i>	EnRcvSOF	EnDetectRESET	EnDetectSUSPEND	EnChirpCmp	EnRestoreCmp	EnSetAddressCmp
0xF1			0xFF								
0xF2	D_FIFO_IntEnb	RW	0x00	EnDescriptorCmp	EnFIFO_IDE_Cmp	EnFIFO1_Cmp	EnFIFO0_Cmp		EnFIFO_NotEmpty	EnFIFO_Full	EnFIFO_Empty
0xF3	D_BulkIntEnb	RW	0x00	EnCBW_Cmp	EnCBW_LengthErr	EnCBW_Err		EnCSW_Cmp	EnCSW_Err		
0xF4	D_EPrintEnb	RW	0x00						EnEPintStat	EnEPIntStat	EnEPIntStat
0xF5	D_EP0IntEnb	RW	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0xF6	D_EPalntEnb	RW	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0xF7	D_EPblntEnb	RW	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0xF8	D_EPcIntEnb	RW	0x00		EnOUT_ShortACK	EnIN_TranACK	EnOUT_TranACK	EnIN_TranNAK	EnOUT_TranNAK	EnIN_TranErr	EnOUT_TranErr
0xF9			0xFF								
0xFA			0xFF								
0xFB			0xFF								
0xFC			0xFF								
0xFD			0xFF								
0xFE			0xFF								
0xFF			0xFF								

7. レジスタ

SLEEP / SNOOZE 時に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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_DEVICE 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x100	<i>D_Reset</i>	RW	0x01								<i>ResetDTM</i>
0x101			0xFF								
0x102	D_NegoControl	RW	0x00	DisBusDetect	EnAutoNego	InSUSPEND	DisableHS	SendWakeup	RestoreUSB	GoChirp	ActiveUSB
0x103			0xFF								
0x104	D_ClrAllEPhJoin	W	0xFF	ClrJoinIDE	ClrJoinFIFO_Stat			ClrJoinDMA1	ClrJoinDMA0	ClrJoinCPU_Rd	ClrJoinCPU_Wr
0x105	D_XcvtControl	RW	0x41	TermSelect	XcvtSelect					OpMode[1:0]	
0x106	D_USB_Test	RW	0x00	EnHS_Test				Test_SE0_NAK	Test_J	Test_K	Test_Packet
0x107			0xFF								
0x108	D_EPhControl	W	0xFF	AllForceNAK	EPrForceSTALL	AllFIFO_Clr					EP0FIFO_Clr
0x109	D_EPrFIFO_Clr	W	0xFF						EPcFIFO_Clr	EPbFIFO_Clr	EPaFIFO_Clr
0x10A	D_BulkOnlyControl	RW	0x00	AutoForceNAK_CBW					GoCBW_Mode	GoCSW_Mode	
0x10B	D_BulkOnlyConfig	RW	0x00						EPcBulkOnly	EPbBulkOnly	
0x10C			0xFF								
0x10D			0xFF								
0x10E			0xFF								
0x10F			0xFF								

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8		
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0		
0x110	D_EPOSETUP_0	R	0x00	SETUP 0[7:0]									
0x111	D_EPOSETUP_1	R	0x00	SETUP 1[7:0]									
0x112	D_EPOSETUP_2	R	0x00	SETUP 2[7:0]									
0x113	D_EPOSETUP_3	R	0x00	SETUP 3[7:0]									
0x114	D_EPOSETUP_4	R	0x00	SETUP 4[7:0]									
0x115	D_EPOSETUP_5	R	0x00	SETUP 5[7:0]									
0x116	D_EPOSETUP_6	R	0x00	SETUP 6[7:0]									
0x117	D_EPOSETUP_7	R	0x00	SETUP 7[7:0]									
0x118	D_USB_Address	R(W)	0x00	<i>SetAddress</i>	USB_Address[6:0]								
0x119			0xFF										
0x11A	D_SETUP_Control	RW	0x00								ProtectEP0		
0x11B			0xFF										
0x11C			0xFF										
0x11D			0xFF										
0x11E	D_FrameNumber_H	R	0x80	Fn_Invalid						FrameNumber[10:8]			
0x11F	D_FrameNumber_L	R	0x00	FrameNumber[7:0]									

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_DEVICE 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x120	D_EP0MaxSize	RW	0x00	EP0MaxSize[6:3]								
0x121	D_EP0Control	RW	0x00	InxOUT							ReplyDescriptor	
0x122	D_EP0ControlIN	RW	0x00		EnShortPkt		ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL	
0x123	D_EP0ControlOUT	RW	0x00	AutoForceNAK			ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL	
0x124			0xFF									
0x125	D_EP0Join	RW	0x00		JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	
0x126			0xFF									
0x127			0xFF									
0x128			0xFF									
0x129			0xFF									
0x12A			0xFF									
0x12B			0xFF									
0x12C			0xFF									
0x12D			0xFF									
0x12E			0xFF									
0x12F			0xFF									

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x130	D_EPaMaxSize_H	RW	0x00	EPaMaxSize[9:8]							
0x131	D_EPaMaxSize_L	RW	0x00	EPaMaxSize[7:3]							
0x132	D_EPaConfig_0	RW	0x00	InxOUT	IntEP_Mode	EnEndpoint	EndpointNumber[3:0]				
0x133			0xFF								
0x134	D_EPaControl	RW	0x00	AutoForceNAK	EnShortPkt	DisAF_NAK_Short	ToggleStat	ToggleSet	ToggleClr	ForceNAK	ForceSTALL
0x135	D_EPaJoin	RW	0x00	JoinIDE	JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr
0x136			0xFF								
0x137			0xFF								
0x138			0xFF								
0x139			0xFF								
0x13A			0xFF								
0x13B			0xFF								
0x13C			0xFF								
0x13D			0xFF								
0x13E			0xFF								
0x13F			0xFF								

7. レジスタ

SLEEP / SNOOZE 時に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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_DEVICE 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x140	D_EPbMaxSize_H	RW	0x00							EPbMaxSize[9:8]		
0x141	D_EPbMaxSize_L	RW	0x00	EPbMaxSize[7:3]								
0x142	D_EPbConfig_0	RW	0x00	<i>INxOUT</i>	<i>IntEP_Mode</i>	<i>EnEndpoint</i>		<i>EndpointNumber[3:0]</i>				
0x143			0xFF									
0x144	D_EPbControl	RW	0x00	<i>AutoForceNAK</i>	<i>EnShortPkt</i>	<i>DisAF_NAK_Short</i>	<i>ToggleStat</i>	<i>ToggleSet</i>	<i>ToggleClr</i>	<i>ForceNAK</i>	<i>ForceSTALL</i>	
0x145	D_EPbJoin	RW	0x00	<i>JoinIDE</i>	<i>JoinFIFO_Stat</i>			<i>JoinDMA1</i>	<i>JoinDMA0</i>	<i>JoinCPU_Rd</i>	<i>JoinCPU_Wr</i>	
0x146			0xFF									
0x147			0xFF									
0x148			0xFF									
0x149			0xFF									
0x14A			0xFF									
0x14B			0xFF									
0x14C			0xFF									
0x14D			0xFF									
0x14E			0xFF									
0x14F			0xFF									

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x150	D_EPcMaxSize_H	RW	0x00							EPcMaxSize[9:8]		
0x151	D_EPcMaxSize_L	RW	0x00	EPcMaxSize[7:3]								
0x152	D_EPcConfig_0	RW	0x00	<i>INxOUT</i>	<i>IntEP_Mode</i>	<i>EnEndpoint</i>		<i>EndpointNumber[3:0]</i>				
0x153			0xFF									
0x154	D_EPcControl	RW	0x00	<i>AutoForceNAK</i>	<i>EnShortPkt</i>	<i>DisAF_NAK_Short</i>	<i>ToggleStat</i>	<i>ToggleSet</i>	<i>ToggleClr</i>	<i>ForceNAK</i>	<i>ForceSTALL</i>	
0x155	D_EPcJoin	RW	0x00	<i>JoinIDE</i>	<i>JoinFIFO_Stat</i>			<i>JoinDMA1</i>	<i>JoinDMA0</i>	<i>JoinCPU_Rd</i>	<i>JoinCPU_Wr</i>	
0x156			0xFF									
0x157			0xFF									
0x158			0xFF									
0x159			0xFF									
0x15A			0xFF									
0x15B			0xFF									
0x15C			0xFF									
0x15D			0xFF									
0x15E			0xFF									
0x15F			0xFF									

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_DEVICE 時に書くことができます。

Byte Addr.	Register Name	R/W	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x160	D_DescAdrs_H	R/W	0x00						DescAdrs[11:8]			
0x161	D_DescAdrs_L	R/W	0x00	DescAdrs[7:0]								
0x162	D_DescSize_H	R/W	0x00						DescSize[9:8]			
0x163	D_DescSize_L	R/W	0x00	DescSize[7:0]								
0x164			0xXX									
0x165			0xXX									
0x166			0xXX									
0x167			0xXX									
0x168			0xXX									
0x169			0xXX									
0x16A			0xXX									
0x16B			0xXX									
0x16C			0xXX									
0x16D			0xXX									
0x16E			0xXX									
0x16F			0xXX									

Byte Addr.	Register Name	R/W	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x170	D_DMA0_FIFO_Control	R/W	0x00	FIFO_Running	AutoEnShort						
0x171			0xXX								
0x172	D_DMA1_FIFO_Control	R/W	0x00	FIFO_Running	AutoEnShort						
0x173			0xXX								
0x174			0xXX								
0x175			0xXX								
0x176			0xXX								
0x177			0xXX								
0x178			0xXX								
0x179			0xXX								
0x17A			0xXX								
0x17B			0xXX								
0x17C			0xXX								
0x17D			0xXX								
0x17E			0xXX								
0x17F			0xXX								

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_DEVICE 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x180			0xFF									
0x181			0xFF									
0x182			0xFF									
0x183			0xFF									
0x184	D_EPaStartAdrs_H	RW	0x00						StartAdrs[12:8]			
0x185	D_EPaStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x186			0xFF									
0x187			0xFF									
0x188	D_EPbStartAdrs_H	RW	0x00						StartAdrs[12:8]			
0x189	D_EPbStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x18A			0xFF									
0x18B			0xFF									
0x18C	D_EPcStartAdrs_H	RW	0x00						StartAdrs[12:8]			
0x18D	D_EPcStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x18E	D_EPcEndAdrs_H	RW	0x00						EndAdrs[12:8]			
0x18F	D_EPcEndAdrs_L	RW	0x00	EndAdrs[7:2]								

0x190~0x1DF は Reserved です。

下記レジスタの詳細は、「Appendix E」を参照して下さい。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x1E0	<i>(Reserved)</i>		0xFF								
0x1E1	D_ModeControl	W	0xFF	<i>(Reserved)</i>	<i>(Reserved)</i>	<i>(Reserved)</i>	SetAddressMode	<i>(Reserved)</i>	<i>(Reserved)</i>	<i>(Reserved)</i>	<i>(Reserved)</i>

0x1E2~0x1FF は Reserved です。

7.3. ホスト・レジスタマップ

SLEEP / SNOOZE 時に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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_HOST 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0xE0	H_SIE_IntStat_0	R(W)	0x00				DetectedCon	DetectedDiscon	DetectedRmtWkup	DetectedDevChirpOK	DetectedDevChirpNG
0xE1	H_SIE_IntStat_1	R(W)	0x00					DisabledCmp	ResumeCmp	SuspendCmp	ResetCmp
0xE2	H_FIFO_IntStat	R(W)	0x00		FIFO_IDE_Cmp	FIFO1_Cmp	FIFO0_Cmp		FIFO_NotEmpty	FIFO_Full	FIFO_Empty
0xE3	H_FrameIntStat	R(W)	0x00						PortErr	FrameNumOver	SOF
0xE4	H_CHrIntStat	R	0x00				H_CHeIntStat	H_CHdIntStat	H_CHcIntStat	H_CHbIntStat	H_CHaIntStat
0xE5	H_CH0IntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition			CTL_SupportCmp	CTL_SupportStop
0xE6	H_CHaIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition			BO_SupportCmp	BO_SupportStop
0xE7	H_CHbIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0xE8	H_CHcIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0xE9	H_CHdIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0xEA	H_CHeIntStat	R(W)	0x00	TotalSizeCmp	TranACK	TranErr	ChangeCondition				
0xEB			0xFF								
0xEC			0xFF								
0xED			0xFF								
0xEE			0xFF								
0xEF			0xFF								

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0xF0	H_SIE_IntEnb_0	RW	0x00				EnDetectedCon	EnDetectedDiscon	EnDetectedRmtWkup	EnDetectedDevChirpOK	EnDetectedDevChirpNG
0xF1	H_SIE_IntEnb_1	RW	0x00					EnDisabledCmp	EnResumeCmp	EnSuspendCmp	EnResetCmp
0xF2	H_FIFO_IntEnb	RW	0x00		EnFIFO_IDE_Cmp	EnFIFO1_Cmp	EnFIFO0_Cmp		EnFIFO_NotEmpty	EnFIFO_Full	EnFIFO_Empty
0xF3	H_FrameIntEnb	RW	0x00						EnPortErr	EnFrameNumOver	EnSOF
0xF4	H_CHrIntEnb	RW	0x00				EnH_CHeIntStat	EnH_CHdIntStat	EnH_CHcIntStat	EnH_CHbIntStat	EnH_CHaIntStat
0xF5	H_CH0IntEnb	RW	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition			EnCTL_SupportCmp	EnCTL_SupportStop
0xF6	H_CHaIntEnb	RW	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition			EnBO_SupportCmp	EnBO_SupportStop
0xF7	H_CHbIntEnb	RW	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0xF8	H_CHcIntEnb	RW	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0xF9	H_CHdIntEnb	RW	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0xFA	H_CHeIntEnb	RW	0x00	EnTotalSizeCmp	EnTranACK	EnTranErr	EnChangeCondition				
0xFB			0xFF								
0xFC			0xFF								
0xFD			0xFF								
0xFE			0xFF								
0xFF			0xFF								

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_HOST 時に書くことができます。

(注) H_NegoControl_1、H_FrameNumber_H、H_FrameNumber_L の Reset 値は ACT_HOST 時に読める値です。それ以外のステータスでは、

Reset 値は 0x00 が読めます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x100	H_Reset	RW	0x01								ResetHTM
0x101			0xFF								
0x102	H_NegoControl_0	RW	0x1X	AutoModeCancel	HostState[2:0]			AutoMode[3:0]			
0x103											
0x104	H_NegoControl_1	RW	0x10		PortSpeed[1:0]					DisChirpFinish	RmtWkupDetEnb
0x105			0xFF								
0x106	H_USB_Test	RW	0x00	EnHS_Test			Test_Force_Enable	Test_SE0_NAK	Test_J	Test_K	Test_Packet
0x107			0xFF								
0x108	H_ChnControl	W	0xFF			AllFIFO_Clr					CH0FIFO_Clr
0x109	H_ChFIFO_Clr	W	0xFF				ChEFIFO_Clr	ChdFIFO_Clr	ChHFIFO_Clr	ChbFIFO_Clr	ChAFIFO_Clr
0x10A	H_ClrAllChnJoin	W	0xFF	ClrJoinIDE	ClrJoinFIFO_Stat			ClrJoinDMA1	ClrJoinDMA0	ClrJoinCPU_Rd	ClrJoinCPU_Wr
0x10B			0xFF								
0x10C			0xFF								
0x10D			0xFF								
0x10E			0xFF								
0x10F			0xFF								

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x110	H_CH0SETUP_0	RW	0x00	SETUP_0[7:0]							
0x111	H_CH0SETUP_1	RW	0x00	SETUP_1[7:0]							
0x112	H_CH0SETUP_2	RW	0x00	SETUP_2[7:0]							
0x113	H_CH0SETUP_3	RW	0x00	SETUP_3[7:0]							
0x114	H_CH0SETUP_4	RW	0x00	SETUP_4[7:0]							
0x115	H_CH0SETUP_5	RW	0x00	SETUP_5[7:0]							
0x116	H_CH0SETUP_6	RW	0x00	SETUP_6[7:0]							
0x117	H_CH0SETUP_7	RW	0x00	SETUP_7[7:0]							
0x118			0xFF								
0x119			0xFF								
0x11A			0xFF								
0x11B			0xFF								
0x11C			0xFF								
0x11D			0xFF								
0x11E	H_FrameNumber_H	R	0x07						FrameNumber[10:8]		
0x11F	H_FrameNumber_L	R	0xFF	FrameNumber[7:0]							

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_HOST 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x120	H_CH0Config_0	RW	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo	
0x121	H_CH0Config_1	RW	0x00	TID[1:0]								
0x122			0xFF									
0x123	H_CH0MaxPktSize	RW	0x00	MaxPktSize[6:0]								
0x124			0xFF									
0x125			0xFF									
0x126	H_CH0TotalSize_H	RW	0x00	TotalSize[15:8]								
0x127	H_CH0TotalSize_L	RW	0x00	TotalSize[7:0]								
0x128	H_CH0HubAdrs	RW	0x00	HubAdrs[3:0]					Port[2:0]			
0x129	H_CH0FuncAdrs	RW	0x00	FuncAdrs[3:0]				EP_Number[3:0]				
0x12A			0xFF									
0x12B	H_CTL_SupportControl	RW	0x00	CTL_SupportState[1:0]							CTL_SupportGo	
0x12C			0xFF									
0x12D			0xFF									
0x12E	H_CH0ConditionCode	R	0x00	ConditionCode[2:0]								
0x12F	H_CH0Join	RW	0x00		JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x130	H_CHaConfig_0	RW	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo	
0x131	H_CHaConfig_1	RW	0x00	TID[1:0]				AutoZerolen			TotalSizeFree	
0x132	H_CHaMaxPktSize_H	RW	0x00							MaxPktSize[9:8]		
0x133	H_CHaMaxPktSize_L	RW	0x00	MaxPktSize[7:0]								
0x134	H_CHaTotalSize_HH	RW	0x00	TotalSize[31:24]								
0x135	H_CHaTotalSize_HL	RW	0x00	TotalSize[23:16]								
0x136	H_CHaTotalSize_LH	RW	0x00	TotalSize[15:8]								
0x137	H_CHaTotalSize_LL	RW	0x00	TotalSize[7:0]								
0x138	H_CHaHubAdrs	RW	0x00	HubAdrs[3:0]					Port[2:0]			
0x139	H_CHaFuncAdrs	RW	0x00	FuncAdrs[3:0]				EP_Number[3:0]				
0x13A	H_BO_SupportControl	RW	0x00			BO_TransportState[1:0]					BO_SupportGo	
0x13B	H_CSW_RcvDataSize	R	0x00					CSW_RcvDataSize[3:0]				
0x13C	H_OUT_EP_Control	RW	0x00				OUT_Toggle	OUT_EP_Number[3:0]				
0x13D	H_IN_EP_Control	RW	0x00				IN_Toggle	IN_EP_Number[3:0]				
0x13E	H_CHaConditionCode	R	0x00	ConditionCode[2:0]								
0x13F	H_CHaJoin	RW	0x00	JoinIDE	JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	

7. レジスタ

SLEEP / SNOOZE 時に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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_HOST 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x140	H_CHbConfig_0	RW	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo	
0x141	H_CHbConfig_1	RW	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen			TotalSizeFree	
0x142	H_CHbMaxPktSize_H	RW	0x00							MaxPktSize[9:8]		
0x143	H_CHbMaxPktSize_L	RW	0x00	MaxPktSize[7:0]								
0x144	H_CHbTotalSize_HH	RW	0x00	TotalSize[31:24]								
0x145	H_CHbTotalSize_HL	RW	0x00	TotalSize[23:16]								
0x146	H_CHbTotalSize_LH	RW	0x00	TotalSize[15:8]								
0x147	H_CHbTotalSize_LL	RW	0x00	TotalSize[7:0]								
0x148	H_CHbHubAdrs	RW	0x00	HubAdrs[3:0]					Port[2:0]			
0x149	H_CHbFuncAdrs	RW	0x00	FuncAdrs[3:0]				EP_Number[3:0]				
0x14A	H_CHbInterval_H	RW	0x00						Interval[10:8]			
0x14B	H_CHbInterval_L	RW	0x00	Interval[7:0]								
0x14C			0xFF									
0x14D			0xFF									
0x14E	H_CHbConditionCode	R	0x00	ConditionCode[2:0]								
0x14F	H_CHbJoin	RW	0x00	JoinIDE	JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x150	H_CHcConfig_0	RW	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo	
0x151	H_CHcConfig_1	RW	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen			TotalSizeFree	
0x152	H_CHcMaxPktSize_H	RW	0x00							MaxPktSize[9:8]		
0x153	H_CHcMaxPktSize_L	RW	0x00	MaxPktSize[7:0]								
0x154	H_CHcTotalSize_HH	RW	0x00	TotalSize[31:24]								
0x155	H_CHcTotalSize_HL	RW	0x00	TotalSize[23:16]								
0x156	H_CHcTotalSize_LH	RW	0x00	TotalSize[15:8]								
0x157	H_CHcTotalSize_LL	RW	0x00	TotalSize[7:0]								
0x158	H_CHcHubAdrs	RW	0x00	HubAdrs[3:0]					Port[2:0]			
0x159	H_CHcFuncAdrs	RW	0x00	FuncAdrs[3:0]				EP_Number[3:0]				
0x15A	H_CHcInterval_H	RW	0x00						Interval[10:8]			
0x15B	H_CHcInterval_L	RW	0x00	Interval[7:0]								
0x15C			0xFF									
0x15D			0xFF									
0x15E	H_CHcConditionCode	R	0x00	ConditionCode[2:0]								
0x15F	H_CHcJoin	RW	0x00	JoinIDE	JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	

7. レジスタ

SLEEP / SNOOZE 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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_HOST 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x160	H_CHdConfig_0	RW	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo	
0x161	H_CHdConfig_1	RW	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen			TotalSizeFree	
0x162	H_CHdMaxPktSize_H	RW	0x00							MaxPktSize[9:8]		
0x163	H_CHdMaxPktSize_L	RW	0x00	MaxPktSize[7:0]								
0x164	H_CHdTotalSize_HH	RW	0x00	TotalSize[31:24]								
0x165	H_CHdTotalSize_HL	RW	0x00	TotalSize[23:16]								
0x166	H_CHdTotalSize_LH	RW	0x00	TotalSize[15:8]								
0x167	H_CHdTotalSize_LL	RW	0x00	TotalSize[7:0]								
0x168	H_CHdHubAdrs	RW	0x00	HubAdrs[3:0]					Port[2:0]			
0x169	H_CHdFuncAdrs	RW	0x00	FuncAdrs[3:0]				EP_Number[3:0]				
0x16A	H_CHdInterval_H	RW	0x00						Interval[10:8]			
0x16B	H_CHdInterval_L	RW	0x00	Interval[7:0]								
0x16C			0xFF									
0x16D			0xFF									
0x16E	H_CHdConditionCode	R	0x00	ConditonCode[2:0]								
0x16F	H_HdJoin	RW	0x00	JoinIDE	JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x170	H_CHeConfig_0	RW	0x00	ACK_Cnt[3:0]				SpeedMode[1:0]		Toggle	TranGo	
0x171	H_CHeConfig_1	RW	0x00	TID[1:0]		TranType[1:0]		AutoZeroLen			TotalSizeFree	
0x172	H_CHeMaxPktSize_H	RW	0x00							MaxPktSize[9:8]		
0x173	H_CHeMaxPktSize_L	RW	0x00	MaxPktSize[7:0]								
0x174	H_CHeHubAdrs	RW	0x00	TotalSize[31:24]								
0x175	H_CHeFuncAdrs	RW	0x00	TotalSize[23:16]								
0x176	H_CHeTotalSize_HH	RW	0x00	TotalSize[15:8]								
0x177	H_CHeTotalSize_HL	RW	0x00	TotalSize[7:0]								
0x178	H_CHeTotalSize_LH	RW	0x00	HubAdrs[3:0]					Port[2:0]			
0x179	H_CHeTotalSize_LL	RW	0x00	FuncAdrs[3:0]				EP_Number[3:0]				
0x17A	H_CHeInterval_H	RW	0x00						Interval[10:8]			
0x17B	H_CHeInterval_L	RW	0x00	Interval[7:0]								
0x17C			0xFF									
0x17D			0xFF									
0x17E	H_CHeConditionCode	R	0x00	ConditonCode[2:0]								
0x17F	H_CHeJoin	RW	0x00	JoinIDE	JoinFIFO_Stat			JoinDMA1	JoinDMA0	JoinCPU_Rd	JoinCPU_Wr	

7. レジスタ

SLEEP / SNOOZE 時に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、ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_HOST 時に書くことができます。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x180	H_CH0StartAdrs_H	RW	0x00				StartAdrs[12:8]					
0x181	H_CH0StartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x182	H_CH0EndAdrs_H	RW	0x00				EndAdrs[12:8]					
0x183	H_CH0EndAdrs_L	RW	0x00	EndAdrs[7:2]								
0x184	H_CHaStartAdrs_H	RW	0x00				StartAdrs[12:8]					
0x185	H_CHaStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x186	H_CHaEndAdrs_H	RW	0x00				EndAdrs[12:8]					
0x187	H_CHaEndAdrs_L	RW	0x00	EndAdrs[7:2]								
0x188	H_CHbStartAdrs_H	RW	0x00				StartAdrs[12:8]					
0x189	H_CHbStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x18A	H_CHbEndAdrs_H	RW	0x00				EndAdrs[12:8]					
0x18B	H_CHbEndAdrs_L	RW	0x00	EndAdrs[7:2]								
0x18C	H_CHcStartAdrs_H	RW	0x00				StartAdrs[12:8]					
0x18D	H_CHcStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x18E	H_CHcEndAdrs_H	RW	0x00				EndAdrs[12:8]					
0x18F	H_CHcEndAdrs_L	RW	0x00	EndAdrs[7:2]								

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit12	bit12	bit11	bit10	bit9	bit8	
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
0x190	H_CHdStartAdrs_H	RW	0x00				StartAdrs[12:8]					
0x191	H_CHdStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x192	H_CHdEndAdrs_H	RW	0x00				EndAdrs[12:8]					
0x193	H_CHdEndAdrs_L	RW	0x00	EndAdrs[7:2]								
0x194	H_CHeStartAdrs_H	RW	0x00				StartAdrs[12:8]					
0x195	H_CHeStartAdrs_L	RW	0x00	StartAdrs[7:2]								
0x196	H_CHeEndAdrs_H	RW	0x00				EndAdrs[12:8]					
0x197	H_CHeEndAdrs_L	RW	0x00	EndAdrs[7:2]								
0x198			0xFF									
0x199			0xFF									
0x19A			0xFF									
0x19B			0xFF									
0x19C			0xFF									
0x19D			0xFF									
0x19E			0xFF									
0x19F			0xFF									

0x1A0~0x1FF は Reserved です。(0x1F5,0x1F6 を除く)

下記レジスタの詳細は、「Appendix D」を参照して下さい。

Byte Addr.	Register Name	RW	Reset	bit15	bit14	bit12	bit12	bit11	bit10	bit9	bit8
				bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x1F4			0xFF								
0x1F5	H_Protect	RW	0x00					PortSpeedWrEnb		TranEnb[1:0]	
0x1F6	H_Monitor	R	0x00								TranRunning
0x1F7			0xFF								

7.4. デバイス/ホスト共通レジスタ詳細説明

7.4.1. 00h *MainIntStat (Main Interrupt Status)*

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	00h	<i>MainIntStat</i>	R	7: <i>DeviceIntStat</i>	0: None	1: Device Interrupts	00h
			R	6: <i>HostIntStat</i>	0: None	1: Host Interrupts	
			R	5: CPU_IntStat	0: None	1: CPU Interrupts	
			R	4: IDE_IntStat	0: None	1: IDE Interrupts	
			R	3: MediaFIFO_IntStat	0: None	1: MediaFIFO Interrupts	
				2:	0:	1:	
				1:	0:	1:	
			R (W)	0: <i>FinishedPM</i>	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 DeviceIntStat

割り込み要因を間接指示します。

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 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 / SNOOZE 中もリード有効です。

Bit6 HostIntStat

割り込み要因を間接指示します。

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 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 / SNOOZE 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 IDE_IntStat

割り込み要因を間接指示します。

IDE_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 IDE_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 MediaFIFO_IntStat

割り込み要因を間接指示します。

MediaFIFO_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 MediaFIFO_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。

Bit2-1 Reserved**Bit0 FinishedPM**

割り込み要因を直接指示します。

PM_Control_0 レジスタで、GoSLEEP, GoSNOOZE, GoActive60, GoActDevice, GoActHost 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 / SNOOZE 中も有効です。

7.4.2. 01h *DeviceIntStat (Device Interrupt Status)*

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	01h	<i>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	
			R	2: <i>D_FIFO_IntStat</i>	0: None	1: FIFO Interrupts	
			R	1: <i>D_EP0IntStat</i>	0: None	1: EP0 Interrupts	
	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_B 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_B**端子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 / SNOOZE 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 / SNOOZE 中もリード有効です。

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 D_FIFO_IntStat

割り込み要因を間接指示します。

D_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 D_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。

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_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。

7.4.3. 02h *HostIntStat (Host Interrupt Status)*

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	02h	<i>HostIntStat</i>	R/(W)	7: <i>VBUS_Err</i>	0: None 1: VBUS Error	00h
			R/(W)	6: <i>LineStatusChanged</i>	0: None 1: Line Status Changed	
			R	5: <i>H_SIE_IntStat_1</i>	0: None 1: SIE Interrupts1	
			R	4: <i>H_SIE_IntStat_0</i>	0: None 1: SIE Interrupts0	
			R	3: <i>H_FrameIntStat</i>	0: None 1: Frame Interrupts	
			R	2: <i>H_FIFOIntStat</i>	0: None 1: FIFO Interrupts	
			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 / SNOOZE 中も有効です。

VBUSFLG_A 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_A 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 / SNOOZE 中も有効です。

ホストポートの DP 端子および DM 端子の状態が SE0 から変化したことを示します。

このビットは、USB ホスト機能を使用していないときにホストポートの信号ライン変化を検出するために使用します。USB ホスト動作時に HostIntEnb.EnLineStateChanged がイネーブルとなっていると、このビットは頻繁にアサートされますので、USB ホスト動作時は HostIntEnb.EnLineStateChanged をディセーブルに設定してください。

Bit5 *H_SIE_IntStat_1*

割り込み要因を間接指示します。

H_SIE_IntStat_1 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_IntEnb_1 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_IntStat_0*

割り込み要因を間接指示します。

H_SIE_IntStat_0 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_IntEnb_0 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 *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 *H_FIFO_IntStat*

割り込み要因を間接指示します。

H_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 H_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。

Bit1 *H_CH0_IntStat*

割り込み要因を間接指示します。

H_CH0_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 H_CH0_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。

Bit0 H_CHr_IntStat

割り込み要因を間接指示します。

H_CHr_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 H_CHr_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。

7.4.4. 03h CPU_IntStat (CPU Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	03h	CPU_IntStat	R (W)	7: RAM_RdCmp	0: None	1: RAM Read Complete	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R (W)	3: DMA1_Countup	0: None	1: DMA1 Counter Overflow	
			R (W)	2: DMA1_Cmp	0: None	1: DMA1 Complete	
			R (W)	1: DMA0_CountUp	0: None	1: DMA0 Counter Overflow	
			R (W)	0: DMA0_Cmp	0: None	1: DMA0 Complete	

CPU インタフェイ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。

Bit6-4 Reserved**Bit3 DMA1_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、DMA1_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。DMA1_Count_HH,HL,LH,LL の値は 0 に戻り、DMA 動作は継続します。

Bit2 DMA1_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。

Bit1 DMA0_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、DMA0_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。DMA0_Count_HH,HL,LH,LL の値は 0 に戻り、DMA 動作は継続します。

Bit0 DMA0_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。

7.4.5. 04h IDE_IntStat (IDE Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	04h	IDE_IntStat	R (W)	7: IDE_RegCmp	0: None	1: Register Access Complete	00h
			R (W)	6: IDE_RegErr	0: None	1: Register Access Error	
			R (W)	5: IDE_SeqWrRegCmp	0: None	1: Sequence Write Complete	
			R (W)	4: CompleteINTRQ	0: None	1: Auto Status Read Compl.	
				3:	0:	1:	
			R (W)	2: IDE_Cmp	0: None	1: DMA Complete	
			R (W)	1: DetectINTRQ	0: None	1: Detected Interrupt	
			R (W)	0: DetectTerm	0: None	1: Detected Device terminate	

IDE 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 IDE_RegCmp

割り込み要因を直接指示します。
IDE_RegAdrs レジスタによる IDE レジスタへのリードライトアクセスが完了した時に"1"にセットされます。

Bit6 IDE_RegErr

割り込み要因を直接指示します。
以下の場合に"1"にセットされます。

- 1) IDE_RegConfig レジスタによる自動ステータスリードの動作シーケンス中に IDE_RegAdrs レジスタによる IDE レジスタへのリードライトアクセスを行った。
- 2) IDE_SeqWrRegControl レジスタによるシーケンスライトの動作シーケンス中に IDE_RegAdrs レジスタによる IDE レジスタへのリードライトアクセスを行った。

Bit5 IDE_SeqWrRegCmp

割り込み要因を直接指示します。
IDE_SeqWrRegControl レジスタによるシーケンスライトの動作が完了した時に"1"にセットされます。

Bit4 CompleteINTRQ

割り込み要因を直接指示します。
IDE_RegConfig レジスタによる自動ステータスリードの動作が完了した時に"1"にセットされます。

Bit3 Reserved**Bit2 IDE_Cmp**

割り込み要因を直接指示します。
IDE_Control レジスタによる DMA 動作が完了した時に"1"にセットされます。

Bit1 DetectINTRQ

割り込み要因を直接指示します。
IDE_RegConfig レジスタの EnAutoStsRd ビットがセットされていないとき、IDE の HINTRQ 信号の立ち上がりを検出された時に"1"にセットされます。

Bit0 DetectTerm

割り込み要因を直接指示します。
IDE_Control レジスタによる IDE の DMA 動作中で転送モードが Ultra-DMA の場合、デバイスターミネートを検出した時に"1"にセットされます。

7.4.6. 05h MediaFIFO_IntStat (Media FIFO Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	05h	MediaFIFO_IntStat		7:	0:	1:
			R (W)	6: MediaIDE_Cmp	0: None	1: Media FIFO IDE Complete
				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

Media 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 **Reserved**

Bit6 **MediaIDE_Cmp**

割り込み要因を直接指示します。

IDE が Join され Dir が"0"の場合、IDE の転送が終了した後に MediaFIFO が空になると、"1"にセットされます。
 Dir が"1"の場合、IDE の転送が終了すると、"1"にセットされます。

Bit5-3 **Reserved**

Bit2 **FIFO_NotEmpty**

割り込み要因を直接指示します。

Media FIFO にデータが存在する状態(NotEmpty)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。

Media FIFO が Full 時に"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。

Media FIFO が Empty 時に"1"にセットされます。

7.4.7. 06h~0Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	06h -0Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.8. 10h *MainIntEnb (Main Interrupt Enable)*

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	10h	MainIntEnb	R / W	7: EnDeviceIntStat	0: Disable	1: Enable	00h
			R / W	6: EnHostIntStat	0: Disable	1: Enable	
			R / W	5: EnCPU_IntStat	0: Disable	1: Enable	
			R / W	4: En IDE_IntStat	0: Disable	1: Enable	
			R / W	3: EnMediaFIFO_IntStat	0: Disable	1: Enable	
				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。

EnDeviceIntStat, EnHostIntStat, EnFinishedPM ビットは SLEEP / SNOOZE 中も有効です。

7.4.9. 11h *DeviceIntEnb (Device Interrupt Enable)*

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	11h	DeviceIntEnb	R / W	7: EnVBUS_Changed	0: Disable	1: Enable	00h
				6:	0:	1:	
			R / W	5: EnD_SIE_IntStat	0: Disable	1: Enable	
			R / W	4: EnD_BulkIntStat	0: Disable	1: Enable	
			R / W	3: EnRcvEP0SETUP	0: Disable	1: Enable	
			R / W	2: EnD_FIFO_IntStat	0: Disable	1: Enable	
			R / W	1: EnD_EP0IntStat	0: Disable	1: Enable	
			R / W	0: EnD_EPrntStat	0: Disable	1: Enable	

DeviceIntStat レジスタの割り込み要因による、MainIntStat レジスタの 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 / SNOOZE 中も有効です。

7.4.10. 12h *HostIntEnb (Host Interrupt Enable)*

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	12h	<i>HostIntEnb</i>	R / W	7: <i>EnVBUS_Err</i>	0: Disable	1: Enable	00h
			R / W	6: <i>EnLineStatusChanged</i>	0: Disable	1: Enable	
			R / W	5: EnH_SIE_IntStat_1	0: Disable	1: Enable	
			R / W	4: EnH_SIE_IntStat_0	0: Disable	1: Enable	
			R / W	3: EnH_FrameIntStat	0: Disable	1: Enable	
			R / W	2: EnH_FIFOIntStat	0: Disable	1: Enable	
			R / W	1: EnH_CH0IntStat	0: Disable	1: Enable	
			R / W	0: EnH_CHrIntStat	0: Disable	1: Enable	

HostIntStat レジスタの割り込み要因による、MainIntStat レジスタの HostIntStat ビットのアサートを許可／禁止します。
EnVBUS_Err, EnLineStatusChanged ビットは SLEEP / SNOOZE 中も有効です。

7.4.11. 13h CPU_IntEnb (CPU Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	13h	CPU_IntEnb	R / W	7: EnRAM_RdCmp	0: Disable	1: Enable	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R / W	3: EnDMA1_Countup	0: Disable	1: Enable	
			R / W	2: EnDMA1_Cmp	0: Disable	1: Enable	
			R / W	1: EnDMA0_CountUp	0: Disable	1: Enable	
			R / W	0: EnDMA0_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。

7.4.12. 14h IDE_IntEnb (IDE Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	14h	IDE_IntEnb	R / W	7: EnIDE_RegCmp	0: Disable	1: Enable	00h
			R / W	6: EnIDE_RegErr	0: Disable	1: Enable	
			R / W	5: En_SeqWrRegCmp	0: Disable	1: Enable	
			R / W	4: EnCompleteINTRQ	0: Disable	1: Enable	
				3:	0:	1:	
			R / W	2: EnIDE_Cmp	0: Disable	1: Enable	
			R / W	1: EnDetectINTRQ	0: Disable	1: Enable	
			R / W	0: EnDetectTerm	0: Disable	1: Enable	

IDE_IntStat レジスタの割り込み要因による、MainIntStat レジスタの IDE_IntStat ビットのアサートを許可／禁止します。

7.4.13. 15h MediaFIFO_IntEnb (Media FIFO Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	15h	MediaFIFO_IntEnb		7:	0:	1:	00h
			R / W	6: EnMedialDE_Cmp	0: Disable	1: Enable	
				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	

MediaFIFO_IntStat レジスタの割り込み要因による、MainIntStat レジスタの MediaFIFO_IntStat ビットのアサートを許可／禁止します。

7.4.14. 16h~1Fh Reserved ()

Address	Register Name	R / W	Bit Symbol	Description	Reset
16h -1Fh	Reserved		7:	0:	1:
			6:	0:	1:
			5:	0:	1:
			4:	0:	1:
			3:	0:	1:
			2:	0:	1:
			1:	0:	1:
			0:	0:	1:
					XXh

7.4.15. 20h **RevisionNum** (Revision Number)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	20h	RevisionNum	R	7: RevisionNum[7]	Revision Number	50h
				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 / SNOOZE 中でもアクセス有効です。
現仕様に関するリビジョンナンバーは、0x50 となっています。

7.4.16. 21h *ChipReset* (*Chip Reset*)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	21h	ChipReset		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			W	0: AllReset	0: None	1: Reset	

本 LSI をリセットします。
SLEEP / SNOOZE 時でもアクセス有効です。

Bit7-1 Reserved**Bit0 AllReset**

本 LSI の全回路をリセットします。外部リセット端子(XRST)と同義です。

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下さい。

7.4.17. 22h PM_Control_0 (Power Management Control 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	22h	PM_Control_0	R / W	7: GoSLEEP	0: Do nothing	1: Go to SLEEP	00h
			R / W	6: GoSNOOZE	0: Do nothing	1: Go to SNOOZE	
			R / W	5: GoActive60	0: Do nothing	1: Go to ACTIVE60	
			R / W	4: GoActDevice	0: Do nothing	1: Go to ACT_DEVICE	
			R / W	3: GoActHost	0: Do nothing	1: Go to ACT_HOST	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

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 / SNOOZE 中も有効です。

Bit7 GoSLEEP

SLEEP ステート以外のステートから、SLEEP ステートへの移行を開始します。
SNOOZE 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停止し SLEEP ステートに移行します。
ACTIVE60 ステート時に本ビットを"1"にセットすると、まず PLL60 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。
ACT_DEVICE ステート時に本ビットを"1"にセットすると、最初に DevicePLL480 の発振を停止し、次に PLL60 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。
ACT_HOST ステート時に本ビットを"1"にセットすると、最初に HostPLL480 の発振を停止し、次に PLL60 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 GoSNOOZE

SNOOZE ステート以外のステートから、SNOOZE 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、SNOOZE ステートに移行します。
ACTIVE60 ステート時に本ビットを"1"にセットすると、PLL60 の発振を停止し、SNOOZE ステートに移行します。
ACT_DEVICE ステート時に本ビットを"1"にセットすると、最初に DevicePLL480 の発振を停止し、次に PLL60 の発振を停止し、SNOOZE ステートに移行します。
ACT_HOST ステート時に本ビットを"1"にセットすると、最初に HostPLL480 の発振を停止し、次に PLL60 の発振を停止し、SNOOZE 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 GoActive60

ACTIVE60 ステート以外のステートから、ACTIVE60 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、PLL60 の発振を開始し、PLL60 発振安定時間 (約 250us) 経過後、ACTIVE60 ステートに移行します。
SNOOZE ステート時に本ビットを"1"にセットすると、PLL60 の発振を開始し、PLL 発振安定時間 (約 250us) 経過後、ACTIVE60 ステートに移行します。
ACT_DEVICE ステート時に本ビットを"1"にセットすると、DevicePLL480 の発振を停止し、ACTIVE60 ステートに移行します。
ACT_HOST ステート時に本ビットを"1"にセットすると、HostPLL480 の発振を停止し、ACTIVE60 ステートに移行

します。

どの状態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。

Bit4 GoActDevice

ACT_DEVICE ステート以外のステートから、ACT_DEVICE 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、PLL60 の発振を開始し、PLL60 発振安定時間(約 250us) 経過後、DevicePLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_DEVICE ステートに移行します。

SNOOZE ステート時に本ビットを"1"にセットすると、PLL60 の発振を開始し、PLL 発振安定時間(約 250us) 経過後、DevicePLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_DEVICE・ステートに移行します。

ACTIVE60 ステート時に本ビットを"1"にセットすると、DevicePLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_DEVICE ステートに移行します。

ACT_HOST ステート時に本ビットを"1"にセットすると、HostPLL480 の発振を停止し、ACTIVE60 ステートに移行します。その後 DevicePLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_DEVICE ステートに移行します。

どの状態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。

Bit3 GoActHost

ACT_HOST ステート以外のステートから、ACT_HOST 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、PLL60 の発振を開始し、PLL60 発振安定時間(約 250us) 経過後、HostPLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_HOST ステートに移行します。

SNOOZE・ステート時に本ビットを"1"にセットすると、PLL60 の発振を開始し、PLL 発振安定時間(約 250us) 経過後、HostPLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_HOST ステートに移行します。

ACTIVE60・ステート時に本ビットを"1"にセットすると、HostPLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_HOST ステートに移行します。

ACT_DEVICE ステート時に本ビットを"1"にセットすると、DevicePLL480 の発振を停止し、ACTIVE60 ステートに移行します。その後 HostPLL480 の発振を開始し、PLL480 発振安定時間(約 250us) 経過後、ACT_HOST ステートに移行します。

どの状態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。

※本 LSI は SLEEP / SNOOZE 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、SNOOZE 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、SNOOZE 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、FW により以下の処理を行って下さい。

<SLEEP / SNOOZE 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 / SNOOZE 解除後>

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)。

Bit2-0 Reserved

7.4.18. 23h *PM_Control_1* (Power Management Control 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	23h	<i>PM_Control_1</i>		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R	3: <i>PM_State[3]</i>	PM_State[3:0]		
	2: <i>PM_State[2]</i>						
	1: <i>PM_State[1]</i>						
	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 / SNOOZE 中も有効です。

Bit7-4 Reserved**Bit3-0** *PM_State[3:0]*

電力モードのステートを表します。

0000: SLEEP ステート	(OSC オフ、PLL60 オフ、DevicePLL480 オフ、HostPLL480 オフ)
0001: SNOOZE ステート	(OSC オン、PLL60 オフ、DevicePLL480 オフ、HostPLL480 オフ)
0011: ACTIVE60 ステート	(OSC オン、PLL60 オン、DevicePLL480 オフ、HostPLL480 オフ)
0111: ACT_DEVICE ステート	(OSC オン、PLL60 オン、DevicePLL480 オン、HostPLL480 オフ)
1011: ACT_HOST ステート	(OSC オン、PLL60 オン、DevicePLL480 オフ、HostPLL480 オン)
その他: 未使用	

なお本ステートは、*PM_Control_0.GoXXXX* 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、
PM_Control_0.GoXXXX ビットがクリアされるまでは、
該当ステートに向かい逐次変化致しますので、参照しないでください。

7.4.19. 24h *WakeupTim_H* (*Wakeup Time High*)7.4.20. 25h *WakeupTim_L* (*Wakeup Time Low*)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	24h -25h	<i>WakeupTim_H</i> <i>WakeupTim_L</i>	R / W	<i>WakeupTim[15:0]</i>	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_0.GoActDevice、PM_Control_0.GoActHost、PM_Control_0.GoActive60、

PM_Control_0.GoSNOOZE 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 + PLL60 安定時間(250us 未満) + PLL480 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。

7.4.21. 26h *H_USB_Control (Host USB Control)*

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	26h	<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 / SNOOZE 中も有効です。

Bit7 *VBUS_Enb*

VBUSEN_A 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*

7.4.22. 27h H_XcvtControl (Host Xcvt Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	27h	H_XcvtControl	R / W	7: TermSelect	0: HS Termination	1: FS Termination	91h
			R / W	6: RemovedRPD	0: RPD ON	1: RPD OFF	
			R / W	5: XcvtSelect[1]	XcvtSelect[1:0]		
			R / W	4: XcvtSelect[0]			
				3:	0:	1:	
				2:	0:	1:	
			R / W	1: OpMode[1] 0: OpMode[0]	OpMode[1:0]		

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 / SNOOZE 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 RemovedRPD

ホストデータラインである DP_A および DM_A 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、通常は"0"(ON)の設定のままで使用してください。

特にホスト動作時(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 XcvtSelect[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: Reserved

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**

HTM 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、ACT_HOSTステート以外の状態でホストポートの信号ライン変化ステータスを検出する場合は、信号ライン変化ステータス項(6.1.2.1.2)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

7.4.23. 28h *D_USB_Status* (Device USB Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	28h	<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[1]</i> 0: <i>LineState[0]</i>	Line State[1:0]	

デバイスに関するステータスを表示します。

Bit7 *VBUS*

VBUS_B 端子の状態が表示されます。このビットは SLEEP / SNOOZE 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 ACTIVE60 / ACT_DEVICE / ACT_HOST 時に読むことができ、ACT_DEVICE 時に書くことができます。

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 / SNOOZE 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

7.4.24. 29h H_USB_Status (Host USB Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	29h	H_USB_Status	R	7: VBUS_State	0: VBUSFLG_A = High 1: VBUSFLG_A = Low	XXh
				6:	0: 1:	
				5:	0: 1:	
				4:	0: 1:	
				3:	0: 1:	
				2:	0: 1:	
			R	1: LineState[1]	Line State[1:0]	
				0: LineState[0]		

ホストに関するステータスを表示します。

Bit7 VBUS_State

VBUSFLG_A 端子の状態が表示されます。このビットは SLEEP / SNOOZE 中も有効です。

Bit6-2 Reserved**Bit1-0 LineState[1:0]**

USB ケーブル上の信号状態を示します。このビットは SLEEP / SNOOZE も有効です。

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		
XcvrSelect[1:0]	DP / DM	LineState[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。

7.4.25. 2A~2Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	2Ah -2Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.26. 30h FIFO_Rd_0 (FIFO Read 0)

7.4.27. 31h FIFO_Rd_1 (FIFO Read 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	30h	FIFO_Rd_0	R	7: FIFO_Rd_0[7]	Endpoint n / Channel n / Media 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	31h	FIFO_Rd_1	R	7: FIFO_Rd_1[7]	Endpoint n / Channel n / Media 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]		

30h.Bit7-0, 31h.Bit7-0 FIFO_Rd_0[7:0], FIFO_Rd_1[7:0]

D_EPx{x=0,a-c}Join.JoinCPU_Rd ビットがセットされているエンドポイント FIFO、または H_CHx{x=0,a-e}Join.JoinCPU_Rd ビットがセットされているチャンネル FIFO、または MediaFIFO_Join.JoinCPU_Rd ビットがセットされているメディア FIFO のデータを読み出せます。

8bit mode 時には FIFO_Rd_0,1 いずれのレジスタにアクセスしても FIFO のデータを読み出せます。

16bit mode 時の 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。

7.4.28. 32h FIFO_Wr_0(FIFO Write 0)

7.4.29. 33h FIFO_Wr_1(FIFO Write 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	32h	FIFO_Wr_0	W	7: FIFO_Wr_0[7]	Endpoint n / Channel n / Media 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[2]		
				1: FIFO_Wr_0[1]		
				0: FIFO_Wr_0[0]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	33h	FIFO_Wr_1	W	7: FIFO_Wr_1[7]	Endpoint n / Channel n / Media 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]		

32h.Bit7-0, 33h.Bit7-0 FIFO_Wr_0[7:0], FIFO_Wr_1[7:0]

D_EPx{x=0,a-c}Join.JoinCPU_Wr ビットがセットされているエンドポイント FIFO、または H_CHx{x=0,a-e}Join.JoinCPU_Wr ビットがセットされているチャンネル FIFO、または MediaFIFO_Join.JoinCPU_Wr ビットがセットされているメディア FIFO ヘデータを書き込みます。

8bit mode 時には FIFO_Wr_0,1 いずれのレジスタにアクセスしても FIFO ヘデータを書き込みます。

16bit mode 時の 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。

7.4.30. 34h FIFO_RdRemain_H (FIFO Read Remain High)

7.4.31. 35h FIFO_RdRemain_L (FIFO Read Remain Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Device / Host	34h	FIFO_RdRemain_H	R	7: RdRemainValid	0: None	1: Read Remain Valid	00h	
				6:	0:	1:		
				5:	0:	1:		
			R	4: RdRemain[12]	Endpoint n / Channel n / Media FIFO Read Remain High			
				3: RdRemain[11]				
				2: RdRemain[10]				
				1: RdRemain[9]				
0: RdRemain[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	35h	FIFO_RdRemain_L	R	7: RdRemain[7]	Endpoint n / Channel n / Media FIFO Read Remain Low		00h
				6: RdRemain[6]			
				5: RdRemain[5]			
				4: RdRemain[4]			
				3: RdRemain[3]			
				2: RdRemain[2]			
				1: RdRemain[1]			
				0: RdRemain[0]			

34h.Bit7 **RdRemainValid**

D_EPx{x=0,a-c}Join.JoinCPU_Rd ビット、または H_CHx{x=0,a-e}Join.JoinCPU_Rd ビット、または MediaFIFO_Join.JoinCPU_Rd ビットによって、CPU I/F にエンドポイント／チャンネル／メディア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。

34h.Bit6-5 **Reserved**34h.Bit4-0, 35h.Bit7-0 **RdRemain[12:0]**

D_EPx{x=0,a-c}Join.JoinCPU_Rd ビット、または H_CHx{x=0,a-e}Join.JoinCPU_Rd ビット、または MediaFIFO_Join.JoinCPU_Rd ビットによって CPU I/F に接続しているエンドポイント／チャンネルの 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。その際に、FIFO_RdRemain_H レジスタを先にアクセスして下さい。

7.4.32. 36h FIFO_WrRemain_H (FIFO Write Remain High)

7.4.33. 37h FIFO_WrRemain_L (FIFO Write Remain Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	36h	WrRemain_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R	4: WrRemain[12]	Endpoint n / Channel n / Media FIFO Write Remain High		
				3: WrRemain[11]			
				2: WrRemain[10]			
				1: WrRemain[9]			
0: WrRemain[8]							

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	37h	WrRemain_L	R	7: WrRemain[7]	Endpoint n / Channel n / Media FIFO Write Remain Low		00h
				6: WrRemain[6]			
				5: WrRemain[5]			
				4: WrRemain[4]			
				3: WrRemain[3]			
				2: WrRemain[2]			
				1: WrRemain[1]			
				0: WrRemain[0]			

36h.Bit7-5

Reserved

36h.Bit4-0, 37h.Bit7-0

WrRemain[12:0]

D_EPx{x=0,a-c}Join.JoinCPU_Wr ビットにより CPU I/F に接続しているエンドポイント FIFO、または H_CHx{x=0,a-e}Join.JoinCPU_Wr ビット、または MediaFIFO_Join.JoinCPU_Wr ビットにより CPU I/F に接続しているチャンネル 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。その際に、FIFO_WrRemain_H レジスタを先にアクセスして下さい。

7.4.34. 38h FIFO_ByteRd(FIFO Byte Read)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	38h	FIFO_ByteRd	R	7: FIFO_ByteRd[7]	Endpoint n / Channel n / Media 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]

D_EPx{x=0,a-c}Join.JoinCPU_Rd ビットがセットされているエンドポイント FIFO、または H_CHx{x=0,a-e}Join.JoinCPU_Rd ビット、または MediaFIFO_Join.JoinCPU_Rd ビットがセットされているチャンネル 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。

7.4.35. 39~3Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	39h -3Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.36. 40h RAM_RdAdrs_H (RAM Read Address High)

7.4.37. 41h RAM_RdAdrs_L (RAM Read Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Device / Host	40h	RAM_RdAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: RAM_RdAdrs[12]	RAM Read Address		
					3: RAM_RdAdrs[11]			
					2: RAM_RdAdrs[10]			
1: RAM_RdAdrs[9]								
0: RAM_RdAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Device / Host	41h	RAM_RdAdrs_L	R / W	7: RAM_RdAdrs[7]	RAM Read Address	00h		
				6: RAM_RdAdrs[6]				
				5: RAM_RdAdrs[5]				
				4: RAM_RdAdrs[4]				
				3: RAM_RdAdrs[3]				
				2: RAM_RdAdrs[2]				
				1:				
				0:				

40h.Bit7-5 **Reserved**

40h.Bit4-0, 41h.Bit7-2 **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。

なお、デバイス・モードでは RAM_RdAdrs[12]に"1"を書き込まないで下さい。RAM_RdAdrs[11:2]が有効なビットです。

ホスト・モードは RAM_RdAdrs[12:2]が有効なビットです。

41h.Bit1-0 **Reserved**

7.4.38. 42h RAM_RdControl (RAM Read Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	42h	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 (ACT_DEVICE ステート) 時は CBW エリア、ホスト動作 (ACT_HOST ステート) 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

7.4.39. 43h RAM_RdCount (RAM Read Counter)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	43h	RAM_RdCount	R / W	7:	RAM Read Counter	00h
				6:		
				5: RAM_RdCount[5]		
				4: RAM_RdCount[4]		
				3: RAM_RdCount[3]		
				2: RAM_RdCount[2]		
				1:		
				0:		

Bit7-0 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。

7.4.40. 44h RAM_WrAdrs_H (RAM Write Address High)

7.4.41. 45h RAM_WrAdrs_L (RAM Write Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	44h	RAM_WrAdrs_H	R / W	7:	RAM Write Address High	00h
				6:		
				5:		
				4: RAM_WrAdrs[12]		
				3: RAM_WrAdrs[11]		
				2: RAM_WrAdrs[10]		
				1: RAM_WrAdrs[9]		
0: RAM_WrAdrs[8]						

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	45h	RAM_WrAdrs_L	R / W	7: RAM_WrAdrs[7]	RAM Write Address Low	00h
				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。

44h.Bit7-5

Reserved

44h.Bit4-0, 45h.Bit7-0

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。

RAM_WrAdrs を参照する場合は、RAM_WrAdrs_H, RAM_WrAdrs_L の順に読み出してください。

7.4.42. 46h RAM_WrDoor_0 (RAM Write Door 0)

7.4.43. 47h RAM_WrDoor_1 (RAM Write Door 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	46h	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	47h	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]		

46h.Bit7-0, 47h.Bit7-0 RAM_WrDoor_0[7:0], 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。

デバイス・モードの場合には、RAM_WrDoor_0,1 レジスタにより、デスクリプタエリアおよび 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リプ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有りません。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。

ホスト・モードの場合には、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。

7.4.44. 48h MediaFIFO_Control (Media FIFO Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	48h	MediaFIFO_Control		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			W	0: MediaFIFO_Clr	0: Do nothing	1: Clear Media FIFO	

MediaFIFO の動作設定を行います。ライトオンリーのレジスタです。

Bit7-1 Reserved**Bit0 MediaFIFO_Clr**

MediaFIFO をクリアします。

このビットは、"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。

MediaFIFO に DMA_x{x=0,1}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、このビットを"1"にセットしないで下さい。

7.4.45. 49h ClrAllMediaFIFO_Join (Clear All Media FIFO Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	49h	ClrAllMediaFIFO_Join	W	7:ClrJoinIDE	0: Do nothing	1: Clear Join IDE	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			W	3:ClrJoinDMA1	0: Do nothing	1: Clear Join DMA1	
				2:ClrJoinDMA0	0: Do nothing	1: Clear Join DMA0	
				1:ClrJoinCPU_Rd	0: Do nothing	1: Clear Join CPU_Rd	
				0:ClrJoinCPU_Wr	0: Do nothing	1: Clear Join CPU_Wr	

該当するポートと MediaFIFO 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。

MediaFIFO がポートに接続 (MediaFIFO_Join 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。

7.4.46. 4Ah MediaFIFO_Join (Media FIFO Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	4Ah	MediaFIFO_Join	R / W	7:JoinIDE	0: Do nothing	1: Join MediaFIFO to IDE	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join MediaFIFO to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join MediaFIFO to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join MediaFIFO to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join MediaFIFO to CPU_Wr	

MediaFIFO とデータ転送を行うポートを指定します。

Bit7 JoinIDE

MediaFIFO で DMA1 の転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6-4 Reserved**Bit3 JoinDMA1**

MediaFIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

MediaFIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

MediaFIFO 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、このチャンネルの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

MediaFIFO 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、このエンドポイントの FIFO にデータが書き込まれます。

JoinDMAx{x=0,1}ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.4.47. 4Bh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	4Bh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.48. 4C~4Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	4Ch -4Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.49.	50h RAM_Rd_00 (RAM Read 00)
7.4.50.	51h RAM_Rd_01 (RAM Read 01)
7.4.51.	52h RAM_Rd_02 (RAM Read 02)
7.4.52.	53h RAM_Rd_03 (RAM Read 03)
7.4.53.	54h RAM_Rd_04 (RAM Read 04)
7.4.54.	55h RAM_Rd_05 (RAM Read 05)
7.4.55.	56h RAM_Rd_06 (RAM Read 06)
7.4.56.	57h RAM_Rd_07 (RAM Read 07)
7.4.57.	58h RAM_Rd_08 (RAM Read 08)
7.4.58.	59h RAM_Rd_09 (RAM Read 09)
7.4.59.	5Ah RAM_Rd_0A (RAM Read 0A)
7.4.60.	5Bh RAM_Rd_0B (RAM Read 0B)
7.4.61.	5Ch RAM_Rd_0C (RAM Read 0C)
7.4.62.	5Dh RAM_Rd_0D (RAM Read 0D)
7.4.63.	5Eh RAM_Rd_0E (RAM Read 0E)
7.4.64.	5Fh RAM_Rd_0F (RAM Read 0F)
7.4.65.	60h RAM_Rd_10 (RAM Read 10)
7.4.66.	61h RAM_Rd_11 (RAM Read 11)
7.4.67.	62h RAM_Rd_12 (RAM Read 12)
7.4.68.	63h RAM_Rd_13 (RAM Read 13)
7.4.69.	64h RAM_Rd_14 (RAM Read 14)
7.4.70.	65h RAM_Rd_15 (RAM Read 15)
7.4.71.	66h RAM_Rd_16 (RAM Read 16)
7.4.72.	67h RAM_Rd_17 (RAM Read 17)
7.4.73.	68h RAM_Rd_18 (RAM Read 18)
7.4.74.	69h RAM_Rd_19 (RAM Read 19)
7.4.75.	6Ah RAM_Rd_1A (RAM Read 1A)
7.4.76.	6Bh RAM_Rd_1B (RAM Read 1B)
7.4.77.	6Ch RAM_Rd_1C (RAM Read 1C)
7.4.78.	6Dh RAM_Rd_1D (RAM Read 1D)
7.4.79.	6Eh RAM_Rd_1E (RAM Read 1E)
7.4.80.	6Fh RAM_Rd_1F (RAM Read 1F)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	50h -6Fh	RAM_Rd_00	R	7: RAM_Rd_xx[7]	RAM Read	00h
		~		6: RAM_Rd_xx[6]		
		RAM_Rd_1F		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]		

50h-6Fh.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。

7.4.81. 70h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	70h	Reserved		7:	0:	1	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.82. 71h DMA0_Config (DMA0 Config)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	71h	DMA0_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: DMA0 Inactive	1: DMA0 Active	
				2:	0:	1:	
			R / W	1: ReqAssertCount[1] 0: ReqAssertCount[0]	Request Assert Count		

DMA0 の動作モードを設定します。

Bit7 FreeRun

DMA0 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

DMA0 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 DMA0_RdData/DMA0_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**

DMA0 の DACK0 を有効にします。

- 0: DMA0(DACK0)無効
- 1: DMA0(DACK0)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。

XDREQ0 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 XDREQ0 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 XDREQ0 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 XDREQ0 をアサートします。

つまり、一回の XDREQ0 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 DMA0_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 DMA0_Count_HH,HL,LH,LL のカウント数が優先され、DMA0_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 XDREQ0 をアサートします。

下表に DMA0_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 XDREQ0 信号及び転送可能数の関係を示します。

DMA0_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
XDREQ0	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	8bit mode
0b00	Normal	Normal
0b01	16Byte(8Count)	16Byte(16Count)
0b10	32Byte(16Count)	32Byte(32Count)
0b11	64Byte(32Count)	64Byte(64Count)

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。

7.4.83. 72h DMA0_Control (DMA0 Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	72h	DMA0_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	

DMA0 の制御、及び状態を表示します。

Bit7 DMA_Running

DMA0 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、EPx{x=0,a-c}Join.JoinDMA0、CHx{x=0,a-e}Join.JoinDMA0、MediaFIFO_Join.JoinDMA0 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、DMA0_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

DMA0 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、DMA0 の転送を終了します。DMA0 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 DMA0_Cmp ビットに"1"をセットします。DMA0 の転送を再開する場合、DMA_Running ビットまたは DMA0_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、DMA0 の転送を開始します。

7.4.84. 73h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	73h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.85. 74h DMA0_Remain_H (DMA0 FIFO Remain High)

7.4.86. 75h DMA0_Remain_L (DMA0 FIFO Remain Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	74h	DMA0_Remain_H		7:	0:	1:	
				6:	0:	1:	
				5:	0:	1:	
				R	4: DMA_Remain[12]	DMA FIFO Remain High	
					3: DMA_Remain[11]		
					2: DMA_Remain[10]		
					1: DMA_Remain[9]		
0: DMA_Remain[8]							
00h							

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	75h	DMA0_Remain_L	R	7: DMA_Remain[7]	DMA FIFO Remain Low	
				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]		
00h						

74h.Bit7-5

Reserved

74h.Bit4-0, 75h.Bit7-0

DMA_Remain[12:0]

読み出しの場合、EPx{x=0,a-c}Join.JoinDMA0 ビットによって DMA に接続しているエンドポイント FIFO、または CHx{x=0,a-e}Join.JoinDMA0 ビットによって DMA に接続しているチャンネル、または MediaFIFO_Join.JoinDMA0 ビットによって DMA に接続している MediaFIFO の FIFO 内の残りデータ数を示します。

書き込みの場合、EPx{x=0,a-c}Join.JoinDMA0 ビットによって DMA に接続しているエンドポイントの FIFO、または CHx{x=0,a-e}Join.JoinDMA0 ビットによって DMA に接続しているチャンネル FIFO、または MediaFIFO_Join.JoinDMA0 ビットによって DMA に接続している MediaFIFO 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。

このレジスタを読み出す場合は DMA0_Remain_H,L の順に読み出してください。

7.4.87. 76h~77h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	76h -77h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

- 7.4.88. 78h DMA0_Count_HH (DMA0 Transfer Byte Counter High/High)
 7.4.89. 79h DMA0_Count_HL (DMA0 Transfer Byte Counter High/Low)
 7.4.90. 7Ah DMA0_Count_LH (DMA0 Transfer Byte Counter Low/High)
 7.4.91. 7Bh DMA0_Count_LL (DMA0 Transfer Byte Counter Low/Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	78h	DMA0_Count_HH	R / W	7: DMA_Count[31]	DMA Transfer Byte Counter High-High	00h
				6: DMA_Count[30]		
				5: DMA_Count[29]		
				4: DMA_Count[28]		
				3: DMA_Count[27]		
				2: DMA_Count[26]		
				1: DMA_Count[25]		
				0: DMA_Count[24]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	79h	DMA0_Count_HL	R / W	7: DMA_Count[23]	DMA Transfer Byte Counter High-Low	00h
				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	7Ah	DMA0_Count_LH	R / W	7: DMA_Count[15]	DMA Transfer Byte Counter Low-High	00h
				6: DMA_Count[14]		
				5: DMA_Count[13]		
				4: DMA_Count[12]		
				3: DMA_Count[11]		
				2: DMA_Count[10]		
				1: DMA_Count[9]		
				0: DMA_Count[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	7Bh	DMA0_Count_LL	R / W	7: DMA_Count[7]	DMA Transfer Byte Counter Low-Low	00h
				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DMA0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、DMA0_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。DMA0_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DMA0_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せん。

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DMA0_Count_HH,HL,LH,LLの順に読み出してください。

い。

7.4.92. 7Ch DMA0_RdData_0 (DMA0 Read Data 0)

7.4.93. 7Dh DMA0_RdData_1 (DMA0 Read Data 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	7Ch	DMA0_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	7Dh	DMA0_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]		

7Ch.Bit7-0, 7Dh.Bit7-0 DMA_RdData_0[7:0], DMA_RdData_1[7:0]

DMA0_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 EPx{x=0,a-c}Join.JoinDMA0 ビット、または CHx{x=0,a-e}Join.JoinDMA0 ビット、または MediaFIFO_Join.JoinDMA0 ビットによって DMA に接続しているエンドポイント/チャンネル/MediaFIFO の FIFO データを読み出す事が出来ます。このとき、DMA0_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。

8bit Mode で動作する場合は、DMA0_RdData_0, DMA0_RdData_1 いずれにアクセスしても同様に DMA アクセスする事が出来ます。

7.4.94. 7Eh DMA0_WrData_0 (DMA0 Write Data 0)

7.4.95. 7Fh DMA0_WrData_1 (DMA0 Write Data 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	7Eh	DMA0_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	7Fh	DMA0_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]		

7Eh.Bit7-0, 7Fh.Bit7-0 DMA_WrData_0[7:0], DMA_WrData_1[7:0]

DMA0_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 EPx{x=0,a-c}Join.JoinDMA0 ビット、または CHx{x=0,a-e}Join.JoinDMA0 ビット、または MediaFIFO_Join.JoinDMA0 ビットによって DMA に接続しているエンドポイント/チャンネル/MediaFIFO の FIFO にデータを書き込む事が出来ます。このとき、DMA0_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。

8bit Mode で動作する場合は、DMA0_WrData_0, DMA0_WrData_1 いずれにアクセスしても同様に DMA アクセスする事が出来ます。

7.4.96. 80h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	80h	reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.97. 81h DMA1_Config (DMA0 Config)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	81h	DMA1_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: DMA1 Inactive	1: DMA1 Active	
			R / W	1: ReqAssertCount[1] 0: ReqAssertCount[0]	Request Assert Count		

DMA1 の動作モードを設定します。

Bit7 FreeRun

DMA1 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

DMA0 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 DMA0_RdData/DMA0_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**

DMA1 の DACK1 を有効にします。

- 0: DMA1 (DACK1) 無効
- 1: DMA1 (DACK1) 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。

XDREQ1 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 XDREQ1 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 XDREQ1 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 XDREQ1 をアサートします。

つまり、一回の XDREQ1 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 DMA1_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 DMA1_Count_HH,HL,LH,LL のカウント数が優先され、DMA1_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 XDREQ1 をアサートします。

下表に DMA1_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 XDREQ1 信号及び転送可能数の関係を示します。

DMA1_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
XDREQ1	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	8bit mode
0b00	Normal	Normal
0b01	16Byte(8Count)	16Byte(16Count)
0b10	32Byte(16Count)	32Byte(32Count)
0b11	64Byte(32Count)	64Byte(64Count)

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。

7.4.98. 82h DMA1_Control (DMA1 Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	82h	DMA1_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				

DMA1 の制御、及び状態を表示します。

Bit7 DMA_Running

DMA1 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、EPx{x=0,a-c}Join.JoinDMA1 ビット、CHx{x=0,a-e}Join.JoinDMA1 ビット、MediaFIFO_Join.JoinDMA1 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、DMA1_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

DMA1 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、DMA1 の転送を終了します。DMA1 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 DMA1_Cmp ビットに"1"をセットします。DMA1 の転送を再開する場合、DMA_Running ビットまたは DMA1_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、DMA1 の転送を開始します。

7.4.99. 83h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	83h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.100. 84h DMA1_Remain_H (DMA1 FIFO Remain High)

7.4.101. 85h DMA1_Remain_L (DMA1 FIFO Remain Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Device / Host	84h	DMA1_Remain_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R	4: DMA_Remain[12]	DMA FIFO Remain High		
					3: DMA_Remain[11]			
					2: DMA_Remain[10]			
					1: DMA_Remain[9]			
0: DMA_Remain[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	85h	DMA1_Remain_L	R	7: DMA_Remain[7]	DMA FIFO Remain Low	00h
				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]		

84h.Bit7-5

Reserved

84h.Bit4-0, 85h.Bit7-0 DMA_Remain[12:0]

読み出しの場合、EPx{x=0,a-c}Join.JoinDMA1 ビットによって DMA に接続しているエンドポイント FIFO、または CHx{x=0,a-h}Join.JoinDMA1 ビットによって DMA に接続しているチャンネル FIFO、または MediaFIFO_Join.JoinDMA1 ビットによって DMA に接続している MediaFIFO 内の残りデータ数を示します。

書き込みの場合、EPx{x=0,a-c}Join.JoinDMA1 ビットによって DMA に接続しているエンドポイントの FIFO、または CHx{x=0,a-h}Join.JoinDMA1 ビットによって DMA に接続しているチャンネル FIFO、または MediaFIFO_Join.JoinDMA1 ビットによって DMA に接続している MediaFIFO 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。

このレジスタを読み出す場合は DMA1_Remain_H,L の順に読み出してください。

7.4.102. 86h~87h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	86h -87h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

- 7.4.103. 88h DMA1_Count_HH (DMA1 Transfer Byte Counter High/High)
 7.4.104. 89h DMA1_Count_HL (DMA1 Transfer Byte Counter High/Low)
 7.4.105. 8Ah DMA1_Count_LH (DMA1 Transfer Byte Counter Low/High)
 7.4.106. 8Bh DMA1_Count_LL (DMA1 Transfer Byte Counter Low/Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	88h	DMA1_Count_HH	R / W	7: DMA_Count[31]	DMA Transfer Byte Counter High-High	00h
				6: DMA_Count[30]		
				5: DMA_Count[29]		
				4: DMA_Count[28]		
				3: DMA_Count[27]		
				2: DMA_Count[26]		
				1: DMA_Count[25]		
				0: DMA_Count[24]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	89h	DMA1_Count_HL	R / W	7: DMA_Count[23]	DMA Transfer Byte Counter High-Low	00h
				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	8Ah	DMA1_Count_LH	R / W	7: DMA_Count[15]	DMA Transfer Byte Counter Low-High	00h
				6: DMA_Count[14]		
				5: DMA_Count[13]		
				4: DMA_Count[12]		
				3: DMA_Count[11]		
				2: DMA_Count[10]		
				1: DMA_Count[9]		
				0: DMA_Count[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	8Bh	DMA1_Count_LL	R / W	7: DMA_Count[7]	DMA Transfer Byte Counter Low-Low	00h
				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]		

88h-8Bh.Bit7-0 DMA_Count[31:0]

カウントモード時に DMA1 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、DMA1_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。DMA1_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 DMA1_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。

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 DMA1_Count_HH,HL,LH,LL の順に読み出してください。

7.4.107. 8Ch DMA1_RdData_0 (DMA1 Read Data 0)

7.4.108. 8Dh DMA1_RdData_1 (DMA1 Read Data 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	8Ch	DMA1_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	8Dh	DMA1_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]		

8Ch.Bit7-0, 8Dh.Bit7-0 DMA_RdData_0[7:0], DMA_RdData_1[7:0]

DMA1_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 EPx{x=0,a-c}Join.JoinDMA1 ビット、または CHx{x=0,a-e}Join.JoinDMA1 ビット、または MediaFIFO_Join.JoinDMA1 ビットによって DMA に接続しているエンドポイント/チャンネル/MediaFIFO の FIFO データを読み出す事が出来ます。このとき、DMA1_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。

8bit Mode で動作する場合は、DMA1_RdData_0, DMA1_RdData_1 いずれにアクセスしても同様に DMA アクセスする事が出来ます。

7.4.109. 8Eh DMA1_WrData_0 (DMA1 Write Data 0)

7.4.110. 8Fh DMA1_WrData_1 (DMA1 Write Data 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	8Eh	DMA1_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	8Fh	DMA1_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]		

8Eh.Bit7-0, 8Fh.Bit7-0 DMA_WrData_0[7:0], DMA_WrData_1[7:0]

DMA1_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 EPx{x=0,a-c}Join.JoinDMA1 ビット、または CHx{x=0,a-e}Join.JoinDMA1 ビット、または MediaFIFO_Join.JoinDMA1 ビットによって DMA に接続しているエンドポイント/チャンネル/MediaFIFO の FIFO にデータを書き込む事が出来ます。このとき、DMA1_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。

8bit Mode で動作する場合は、DMA1_WrData_0, DMA1_WrData_1 いずれにアクセスしても同様に DMA アクセスする事が出来ます。

7.4.111. 90h IDE_Status (IDE Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	90h	IDE_Status	R	7: DMARQ	0:HDMARQ Not Asserted	1:HDMARQ Asserted	00h
			R	6: DMACK	0:XHDMACK Not Asserted	1:XHDMACK Asserted	
			R	5: INTRQ	0:HINTRQ Not Asserted	1:HINTRQ Asserted	
			R	4: IORDY	0:HIORDY Not Asserted	1:HIORDY Asserted	
				3:	0:	1:	
				2:	0:	1:	
			R	1: PDIAG	0:xHPDIAG Not Asserted	1:xHPDIAG Asserted	
			R	0: DASP	0:xHDASP Not Asserted	1:xHDASP Asserted	

IDE バスの信号状態を表示します。各信号がアサートされている時に"1"を読み出すことができます。

XHDMACK、XHPDIAG、XHDASP の負論理信号は、電圧レベルが"0"の時に"1"が読み出されることに注意してください。

Bit3-2 はリザーブで常に"0"が読み出されます。

7.4.112. 91h IDE_Control (IDE Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	91h	IDE_Control		7:	0:	1:	00h
			W	6: IDE_Clr	0: None	1: Clear IDE Circuit	
				5:	0:	1:	
				4:	0:	1:	
			R/W	3: Dir	0: IDE → FIFO RAM	1: IDE ← FIFO RAM	
				2:	0:	1:	
				1:	0:	1:	
			R/W	0: IDE_Go	0: None	1: IDE DMA Go	

IDE の DMA を制御します。

Bit7 **Reserved**

Bit6 **IDE_Clr**

このビットに"1"をセットすると、IDE 回路が初期状態に戻ります。設定したレジスタの内容は変わりません。IDE の DMA 中やレジスタアクセスのシーケンス中に本ビットをセットしてはいけません。

Bit5-4 **Reserved**

Bit3 **Dir**

IDE の転送方向を設定します。

0: IDE → FIFO RAM (IDE リード)

1: IDE ← FIFO RAM (IDE ライト)

Bit2-1 **Reserved**

Bit0 **IDE_Go**

このビットに"1"をセットすると IDE の DMA が起動され、終了すると IDE_IntStat レジスタの IDE_Cmpビットが"1"になります。

DMA の転送中はこのビットが"1"にセットされて、終了するとこのビットは"0"に戻ります。

このビットが"1"である間に"0"を書き込むと実行中の DMA 転送は中断されて終了しますが、IDE_IntStat レジスタの IDE_Cmpビットはセットされません。

7.4.113. 92h IDE_Config_0 (IDE Configuration 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	92h	IDE_Config_0	R/W	7: IDE_BusReset	0: None	1: XHRESET Asserted	00h
			R/W	6: IDE_LongBusReset	0: None	1: XHRESET Asserted	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R/W	1: Ultra	0: Non Ultra mode	1: Ultra mode	
			R/W	0: DMA	0: Non DMA mode	1: DMA mode	

IDE の DMA を制御します。

Bit7 IDE_BusReset

このビットに"1"をセットすると、IDE の xHRESET 信号が 50 μ s アサートされます。このビットに"1"が表示されている間に再度このビットに"1"をセットすると、そこからさらに 50 μ s 間 IDE の XHRESET 信号がアサートされます。Bit7/Bit6 どちらかの XHRESET 信号アサートでこのビットには"1"が読み出されます。

Bit6 IDE_LongBusReset

このビットに"1"をセットすると、IDE の xHRESET 信号が 400 μ s アサートされます。このビットに"1"が表示されている間に再度このビットに"1"をセットすると、そこからさらに 400 μ s 間 IDE の XHRESET 信号がアサートされます。Bit7/Bit6 どちらかの XHRESET 信号アサートでこのビットには"1"が読み出されます。

Bit5-2 Reserved**Bit1 Ultra**

このビットと Bit0:DMA に同時に"1"をセットすると、IDE_Control レジスタで起動される IDE の DMA が Ultra モードになります。

DMA の転送中はこのビットを書き換えてはいけません。下表にこのビットにより設定される IDE の DMA の転送モードを示します。

Bit0 DMA

このビットに"1"をセットすると、IDE_Control レジスタで起動される IDE の DMA がマルチワード DMA モードになります。

DMA の転送中はこのビットを書き換えてはいけません。下表にこのビットにより設定される IDE の DMA の転送モードを示します。

Bit1-0	"00"	"01"	"10"	"11"
	PIO	Multiword DMA	設定禁止	Ultra

7.4.114. 93h IDE_Config_1 (IDE Configuration 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	93h	IDE_Config_1	R/W	7: ActiveIDE	0: InActivated IDE Bus 1: Activate IDE Bus	04h
			R/W	6: DelayStrobe	0: Not Delay Strobe Signal 1: Delay Strobe Signal	
				5:	0: 1:	
			R/W	4: InterLock	0: None 1: DMA InterLock	
				3:	0: 1:	
			R/W	2: Swap	0: Data Swap 1: None	
				1:	0: 1:	
				0:	0: 1:	

IDE バスの状態を制御します。

Bit7 ActiveIDE

このビットに"1"をセットすると、IDE の出力信号が有効になります。IDE バスへのレジスタリードライトコマンド発行や、IDE-DMA を実行するにはあらかじめこのビットに"1"をセットしておく必要があります。このビットが"0"の時は、IDE 信号は全て入力状態になっています。

Bit6 DelayStrobe

このビットに"1"をセットすると、IDE-DMA のマルチワード DMA 転送時、XHDMACK アサートから XHIOR/XHIOW のストロブ信号アサートまで 2 システムクロック間(約 33ns)セットアップ時間を確保します。このビットが"0"の時は IDE-DMA のマルチワード DMA 転送時、XHDMACK アサートと XHIOR/XHIOW のストロブ信号アサートは同時(約 0ns)になります。

Bit5 Reserved**Bit4 InterLock**

このビットに"1"をセットすると、IDE-DMA のマルチワード DMA 転送時、LSI 内部でデータが転送できなくなったことによる要因では XHDMACK をネゲートせずに、IDE バスを保持したまま内部のデータが準備できるまで待ちます。このビットが"0"の時は、内部のデータが準備できない時は一旦 XHDMACK を開放します。

Bit3 Reserved**Bit2 Swap**

このビットを"0"にクリアすると、IDE バスのデータ上位 8 ビットと下位 8 ビットが反転して入出力されます。通常は、このビットが"1"にセットされているようにして下さい。Swap ビットの詳細につきまして、Appendix A を参照して下さい。

Bit1-0 Reserved

7.4.115. 94h IDE_Rmod (IDE Register Mode)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	94h	IDE_Rmod	R/W	7: RegisterAssertPulseWidth[3]	Register Assert Pulse Width	00h
				6: RegisterAssertPulseWidth[2]		
				5: RegisterAssertPulseWidth[1]		
				4: RegisterAssertPulseWidth[0]		
			R/W	3: RegisterNegatePulseWidth[3]	Register Negate Pulse Width	
				2: RegisterNegatePulseWidth[2]		
				1: RegisterNegatePulseWidth[1]		
				0: RegisterNegatePulseWidth[0]		

CPU の IDE バスへのレジスタモードでのアクセス時 XHIOR/XHIOW アサート・ネゲート時のストロブ幅を設定します。
IDE の転送モードに合わせた適切な値を選択する必要があります。

Bit7-4 RegisterAssertPulseWidth[3:0]

システムクロック(60MHz)周期の[RegisterAssertPulseWidth + 4]倍になります。

ex 0000: $4 \times 16.67\text{nS} = 67\text{nS}$

0001: $5 \times 16.67\text{nS} = 83\text{nS}$

Bit3-0 RegisterNegatePulseWidth[3:0]

システムクロック(60MHz)周期の[RegisterNegatePulseWidth + 4]倍になります。

ex 0000: $4 \times 16.67\text{nS} = 67\text{nS}$

0001: $5 \times 16.67\text{nS} = 83\text{nS}$

7.4.116. 95h IDE_Tmod (IDE Transfer Mode)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	95h	IDE_Tmod	R/W	7: TransferAssertPulseWidth[3]	Transfer Assert Pulse Width	00h
				6: TransferAssertPulseWidth[2]		
				5: TransferAssertPulseWidth[1]		
				4: TransferAssertPulseWidth[0]		
			R/W	3: TransferNegatePulseWidth[3]	Transfer Negate Pulse Width	
				2: TransferNegatePulseWidth[2]		
				1: TransferNegatePulseWidth[1]		
				0: TransferNegatePulseWidth[0]		

CPU 及び DMA 転送時の IDE バスへの PIO モードアクセス時 XHIOR/XHIOW アサート・ネゲート時のストロブ幅を設定します。

IDE の転送モードに合わせた適切な値を選択する必要があります。

Bit7-4 TransferAssertPulseWidth[3:0]

システムクロック(60MHz)周期の [TransferAssertPulseWidth + 4] 倍になります。

ex 0000: $4 \times 16.67\text{nS} = 67\text{nS}$

0001: $5 \times 16.67\text{nS} = 83\text{nS}$

Bit3-0 TransferNegatePulseWidth[3:0]

システムクロック(60MHz)周期の [TransferNegatePulseWidth + 4] 倍になります。

ex 0000: $4 \times 16.67\text{nS} = 67\text{nS}$

0001: $5 \times 16.67\text{nS} = 83\text{nS}$

7.4.117. 96h IDE_Umod (IDE Ultra-DMA Transfer Mode)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	96h	IDE_Umod		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R/W	3: UltraDMA_Cycle[3]	UltraDMA_Cycle		
				2: UltraDMA_Cycle[2]			
				1: UltraDMA_Cycle[1]			
				0: UltraDMA_Cycle[0]			

DMA 転送時の IDE バスへの Ultra モードアクセス時のアクセスサイクル幅を設定します。
IDE の転送モードに合わせた適切な値を選択する必要があります。

Bit7-4 **Reserved**

Bit3-0 **UltraDMA_Cycle[3:0]**

システムクロック(60MHz)周期の[UltraDMA_Cycle + 2]倍になります。

ex 0000: $2 \times 16.67\text{nS} = 33\text{nS}$

0001: $3 \times 16.67\text{nS} = 50\text{nS}$

7.4.118. 97h~99h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	97h -99h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.119. 9Ah IDE_CRC_H (IDE CRC High)

7.4.120. 9Bh IDE_CRC_L (IDE CRC Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	9Ah	IDE_CRC_H	R	7: IDE_CRC[15]	IDE_CRC[15:8]	00h
				6: IDE_CRC[14]		
				5: IDE_CRC[13]		
				4: IDE_CRC[12]		
				3: IDE_CRC[11]		
				2: IDE_CRC[10]		
				1: IDE_CRC[9]		
				0: IDE_CRC[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	9Bh	IDE_CRC_L	R	7: IDE_CRC[7]	IDE_CRC[7:0]	00h
				6: IDE_CRC[6]		
				5: IDE_CRC[5]		
				4: IDE_CRC[4]		
				3: IDE_CRC[3]		
				2: IDE_CRC[2]		
				1: IDE_CRC[1]		
				0: IDE_CRC[0]		

IDE の Ultra モードでの DMA 転送時の CRC 計算結果を逐次表示します。読み出す場合は、IDE_CRC_H と IDE_CRC_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 IDE_CRC_H レジスタを先にアクセスして下さい。

7.4.121. 9Ch Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	9Ch	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.122. 9Dh IDE_Count_H (IDE Transfer Byte Counter High)

7.4.123. 9Eh IDE_Count_M (IDE Transfer Byte Counter Middle)

7.4.124. 9Fh IDE_Count_L (IDE Transfer Byte Counter Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	9Dh	IDE_Count_H	R/W	7: IDE_Count[23]	IDE_Count[23:16]	00h
				6: IDE_Count[22]		
				5: IDE_Count[21]		
				4: IDE_Count[20]		
				3: IDE_Count[19]		
				2: IDE_Count[18]		
				1: IDE_Count[17]		
				0: IDE_Count[16]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	9Eh	IDE_Count_M	R/W	7: IDE_Count[15]	IDE_Count[15:8]	00h
				6: IDE_Count[14]		
				5: IDE_Count[13]		
				4: IDE_Count[12]		
				3: IDE_Count[11]		
				2: IDE_Count[10]		
				1: IDE_Count[9]		
				0: IDE_Count[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	9Fh	IDE_Count_L	R/W	7: IDE_Count[7]	IDE_Count[7:1]	00h
				6: IDE_Count[6]		
				5: IDE_Count[5]		
				4: IDE_Count[4]		
				3: IDE_Count[3]		
				2: IDE_Count[2]		
				1: IDE_Count[1]		
				0:	0:	1:

IDE の DMA 転送時の転送バイト数を設定します。このレジスタ設定が 0 バイトで DMA を起動した場合は起動が無視されます。読み出す場合は、IDE_Count_H と IDE_Count_M と IDE_Count_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 IDE_CRC_H レジスタを先にアクセスして下さい。尚、IDE_Count_L レジスタの最下位ビットは常に"0"が表示されます。

7.4.125. A0h IDE_RegAdrs (IDE Register Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	A0h	IDE_RegAdrs	R/W	7: IDE_WrReg	0: None	1: IDE Register Write Go	00h
			R/W	6: IDE_RdReg	0: None	1: IDE Register Read Go	
				5:	0:	1:	
				4:	0:	1:	
			R/W	3: IDE_RegAddress[3]	IDE_RegAddress[3:0]		
				2: IDE_RegAddress[2]			
				1: IDE_RegAddress[1]			
	0: IDE_RegAddress[0]						

CPUによるIDEバスへのレジスタアクセスを制御します。

Bit7 IDE_WrReg

このビットに"1"をセットすると、あらかじめセットしてある IDE_WrRegValue_0,1 レジスタの内容で、IDE バスに対してPIOモードまたはレジスタモードでIDEレジスタをライトします。動作中はこのビットに"1"が表示され、動作が終了するとIDE_IntStatレジスタのIDE_RegCmpビットがセットされて、このビットは"0"に戻ります。IDEバスへのアドレスはあらかじめ、または同時にIDE_RegAddressにセットする必要があります。あらかじめIDE_Rmod、IDE_Tmodレジスタを適切なモードに設定しておく必要があります。

Bit6 IDE_RdReg

このビットに"1"をセットすると、IDEバスに対してPIOモードまたはレジスタモードでIDEレジスタをリードして、リードした値をIDE_RdRegValue_0,1レジスタにセットします。動作中はこのビットに"1"が表示され、動作が終了するとIDE_IntStatレジスタのIDE_RegCmpビットがセットされて、このビットは"0"に戻ります。IDEバスへのアドレスはあらかじめ、または同時にIDE_RegAddressにセットする必要があります。あらかじめIDE_Rmod、IDE_Tmodレジスタを適切なモードに設定しておく必要があります。

Bit5-4 Reserved**Bit3-0 IDE_RegAddress[3:0]**

IDE_WrReg、IDE_RdRegビットによるCPUのIDEバスへのレジスタアクセス時のアドレスを設定します。IDEバスに出力されるアドレスの対応は下表の通りです。

IDE_RegAddress[3]	0:XHCS0=0	1:XHCS1=0
IDE_RegAddress[2]	0:HDA2=0	1:HDA2=1
IDE_RegAddress[1]	0:HDA1=0	1:HDA1=1
IDE_RegAddress[0]	0:HDA0=0	1:HDA0=1

7.4.126. A1h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	A1h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.127. A2h IDE_RdRegValue_0 (IDE Register Read Value 0)

7.4.128. A3h IDE_RdRegValue_1 (IDE Register Read Value 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	A2h	IDE_RdRegValue_0	R	7: IDE_RdRegValue_0[7]	IDE RdRegValue 0	00h
				6: IDE_RdRegValue_0[6]		
				5: IDE_RdRegValue_0[5]		
				4: IDE_RdRegValue_0[4]		
				3: IDE_RdRegValue_0[3]		
				2: IDE_RdRegValue_0[2]		
				1: IDE_RdRegValue_0[1]		
				0: IDE_RdRegValue_0[0]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	A3h	IDE_RdRegValue_1	R	7: IDE_RdRegValue_1[7]	IDE RdRegValue1	00h
				6: IDE_RdRegValue_1[6]		
				5: IDE_RdRegValue_1[5]		
				4: IDE_RdRegValue_1[4]		
				3: IDE_RdRegValue_1[3]		
				2: IDE_RdRegValue_1[2]		
				1: IDE_RdRegValue_1[1]		
				0: IDE_RdRegValue_1[0]		

IDE_RegAdrs レジスタの IDE_RdReg ビットによる CPU の IDE バスに対する IDE レジスタのリードで、リードした値がこのレジスタにセットされます。また、IDE_RegConfig レジスタの自動ステータスレジスタリードでもリードした値がこのレジスタにセットされます。読み出す場合は、IDE_RdRegValue_0 と IDE_RdRegValue_1 レジスタを対でアクセスする必要があります。その際に IDE_RdRegValue_0 レジスタを先にアクセスして下さい。

7.4.129. A4h IDE_WrRegValue_0 (IDE Register Write Value 0)

7.4.130. A5h IDE_WrRegValue_1 (IDE Register Write Value 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	A4h	IDE_WrRegValue_0	R/W	7: IDE_WrRegValue_0[7]	IDE WrRegValue 0	00h
				6: IDE_WrRegValue_0[6]		
				5: IDE_WrRegValue_0[5]		
				4: IDE_WrRegValue_0[4]		
				3: IDE_WrRegValue_0[3]		
				2: IDE_WrRegValue_0[2]		
				1: IDE_WrRegValue_0[1]		
				0: IDE_WrRegValue_0[0]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	A5h	IDE_WrRegValue_1	R/W	7: IDE_WrRegValue_1[7]	IDE WrRegValue 1	00h
				6: IDE_WrRegValue_1[6]		
				5: IDE_WrRegValue_1[5]		
				4: IDE_WrRegValue_1[4]		
				3: IDE_WrRegValue_1[3]		
				2: IDE_WrRegValue_1[2]		
				1: IDE_WrRegValue_1[1]		
				0: IDE_WrRegValue_1[0]		

IDE_RegAdrs レジスタの IDE_WrReg ビットによる CPU の IDE バスに対する IDE レジスタのライトで、ライトすべきデータをあらかじめここにセットします。

7.4.131. A6h IDE_SeqWrRegControl (IDE Sequential Register Write Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	A6h	IDE_SeqWrRegControl	R/W	7: IDE_SeqWrReg	0:	1: IDE Sequence Write Go	00h
			W	6: IDE_SeqWrRegClr	0:	1: Clear IDE Sequence Write	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

CPUによるIDEバスへのレジスタへのシーケンスライト動作を制御します。

Bit7 IDE_SeqWrReg

このビットに"1"をセットすると、あらかじめ IDE_SeqWrRegAdrs / IDE_SeqWrRegValue にセットされた最大 16 組のアドレス・データが、セットした順に IDE バスへレジスタライトされて、終了すると IDE_IntStat レジスタの IDE_SeqWrRegCmpビットが"1"になります。

シーケンス動作中はこのビットが"1"にセットされて、終了するとこのビットは"0"に戻ります。

Bit6 IDE_SeqWrRegClr

このビットに"1"をセットすると、あらかじめ IDE_SeqWrRegAdrs / IDE_SeqWrRegValue にセットされた最大 16 組のアドレス・データを破棄して初期状態に戻すことができます。シーケンス動作中に本ビットをセットしてはいけません。

Bit5-0 Reserved

7.4.132. A7h IDE_SeqWrRegCnt (IDE Sequential Register Write Counter)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	A7h	IDE_SeqWrRegCnt		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R	4: IDE_SeqWrRegCnt[4]	IDE_SeqWrRegCnt[4:0]		
				3: IDE_SeqWrRegCnt[3]			
				2: IDE_SeqWrRegCnt[2]			
				1: IDE_SeqWrRegCnt[1]			
	0: IDE_SeqWrRegCnt[0]						

IDE_SeqWrRegValue レジスタに書き込んだデータ数を表示します。最大 10h までが表示されます。IDE バスへのシーケンスライトが行われるとともに値が減り、IDE_SeqWrRegValue レジスタに書き込んだ全てのデータが IDE バスに書き込み完了すると"0"に戻ります。IDE_SeqWrRegControl レジスタの IDE_SeqWrRegClr ビットへの"1"書き込みでも"0"に戻ります。

7.4.133. A8h IDE_SeqWrRegAdrs (IDE Sequential Register Write Address FIFO)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	A8h	IDE_SeqWrRegAdrs		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			W	3: IDE_SeqRegAddress[3]	IDE_SeqRegAddress[3:0]		
				2: IDE_SeqRegAddress[2]			
				1: IDE_SeqRegAddress[1]			
				0: IDE_SeqRegAddress[0]			

IDE_SeqWrRegControl レジスタによる IDE バスへのシーケンスライト動作時、IDE バスへ出力するアドレスを IDE_SeqWrRegValue レジスタのデータと対でセットします。同じアドレスが連続する場合には、そのアドレスを一度セットしたら再セットする必要はありません。IDE バスに出力されるアドレスとビットとの関係は IDE_RegAdrs レジスタの IDE_RegAddress ビットと同じです。

7.4.134. A9h IDE_SeqWrRegValue (IDE Sequential Register Write Value FIFO)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	A9h	IDE_SeqWrRegValue	W	7: IDE_SeqWrRegValue[7]	IDE_SeqWrRegValue[7:0]	XXh
				6: IDE_SeqWrRegValue[6]		
				5: IDE_SeqWrRegValue[5]		
				4: IDE_SeqWrRegValue[4]		
				3: IDE_SeqWrRegValue[3]		
				2: IDE_SeqWrRegValue[2]		
				1: IDE_SeqWrRegValue[1]		
				0: IDE_SeqWrRegValue[0]		

IDE_SeqWrRegControl レジスタによる IDE バスへのシーケンスライト動作時、IDE バスへ出力するデータを IDE_SeqWrRegAdrs レジスタのアドレスと対で順にセットします。最大 16 組セットすることが出来、それを超える書き込みは無視されます。IDE_SeqWrRegAdrs が"0"の時(XHCS=0、HDA=0 のデータポートへのライト)は、IDE に対しては 16 ビットアクセスになりますので、このレジスタを 2 回セット(データは下位バイト・上位バイトの順です)する必要があり、その場合は 16 組中の 2 組が使われます。それ以外のアドレスでは IDE に対しては 8 ビットアクセスになりますので、IDE への 1 回のライトに対してこのレジスタを 1 回セットすることになります。

7.4.135. AAh~ABh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	AAh -ABh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.136. ACh IDE_RegConfig (IDE Register Configuration)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	ACh	IDE_RegConfig	R/W	7: EnAutoStsRd	0: None	1: Auto Status Read Enable	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

IDE バスの HINTRQ 割り込みでの自動ステータスリード動作を制御します。

Bit7 EnAutoStsRd

このビットに"1"をセットすると、IDE バスの HINTRQ 割り込みが発生した際に IDE バスのステータスレジスタ (XHCS0=0、HDA=7)を自動で読みに行き、終了すると IDE_RdRegValue レジスタにリードした値をセットして、IDE_IntStat レジスタの CompleteINTRQ ビットが"1"になります。終了してもこのビットはセットされたままです。

Bit6-0 Reserved

7.4.137. ADh~AFh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	ADh -AFh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.138. B0h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	B0h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.139. B1h *HostDeviceSel* (*Host Device Select*)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	B1h	<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	

レジスタマップのデバイス・モードまたはホスト・モードの設定を行います。

Bit7-1 Reserved

Bit0 **HOSTxDEVICE**

レジスタマップのデバイス・モードまたはホスト・モードの設定を行います。
このビットの設定を変更しても、各レジスタの設定値はクリアされません。
このビットは、SLEEP / SNOOZE 中もアクセス有効です。

7.4.140. B2h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	B2h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.141. B3h *ModeProtect*(*Mode Protection*)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	B3h	<i>ModeProtect</i>	R / W	7: <i>ModeProtect</i> [7]	Mode Protection	56h
				6: <i>ModeProtect</i> [6]		
				5: <i>ModeProtect</i> [5]		
				4: <i>ModeProtect</i> [4]		
				3: <i>ModeProtect</i> [3]		
				2: <i>ModeProtect</i> [2]		
				1: <i>ModeProtect</i> [1]		
				0: <i>ModeProtect</i> [0]		

Bit7-0 ModeProtect[7:0]

ChipConfig レジスタ及び ClkSelect.ClkSelect ビットの値を保護します。このレジスタに 56h を書き込むと ChipConfig レジスタ及び ClkSelect.ClkSelect ビット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、ChipConfig レジスタ及び ClkSelect.ClkSelect ビット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、ChipConfig レジスタ及び ClkSelect.ClkSelect ビットの設定を保護して下さい。

このビットは、SLEEP / SNOOZE 中もアクセス有効です。

7.4.142. B4h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	B4h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.143. B5h *ClkSelect* (Clock Select)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device / Host	B5h	<i>ClkSelect</i>	R / W	7: <i>xActIDE_Term</i>	0: Termination ON	1: Termination OFF	41h
			R / W	6: <i>xActIDE_DD_Term</i>	0: Termination ON	1: Termination OFF	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R / W	1: <i>PORT1x2</i>	0: 2 Port	1: 1 Port	
			R / W	0: <i>ClkSelect</i>	0: 12MHz	1: 24MHz	

Bit7 *xActIDE_Term*

IDE ポートの端子のうち、
HDMARQ、HIORDY、HINTRQ、XHDASP、XHPDIAG、HDD7 の Termination を ON/OFF します。
このビットは、SLEEP / SNOOZE 中もアクセス有効です。
0: Termination ON
1: Termination OFF

Bit6 *xActIDE_DD_Term*

IDE ポート端子のうち、
HDD15~HDD8、HDD6~HDD0 の Termination を ON/OFF します。
このビットは、SLEEP / SNOOZE 中もアクセス有効です。
0: Termination ON
1: Termination OFF

Bit6-2 **Reserved****Bit1** *PORT1x2*

本 LSI で使用する USB ポートのモードを選択します。
2 Port 設定時は、A ポートに USB Host、B ポートに USB Device が割り当てられ、いずれかの機能を選択して使
用します。
1 Port 設定時は、B ポートのみが使用でき、B ポートにて USB Host / Device のいずれかの機能を選択して使
用します。
1 ポートモードの詳細は Appendix C を参照してください。
このビットは、SLEEP / SNOOZE 中もアクセス有効です。
0: 2 Port
1: 1 Port

Bit0 *ClkSelect*

本 LSI で使用するクロックを選択します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。
0: 12MHz
1: 24MHz

7.4.144. B6h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	B6h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.145. B7h ChipConfig (Chip Configuration)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	B7h	ChipConfig	R / W	7: IntLevel	0: Low Active	1: High Active	00h
			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 / W	2: CPU_Endian	0: Do nothing	1: Bus Swap	
			R / W	1: BusMode	0: XWRH/L mode	1: XBEH/L mode	
			R / W	0: Bus8x16	0: 16bit mode	1: 8bit mode	

本 LSI の動作モードを設定します。

Bit7 IntLevel

XINT の論理レベルを設定します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。

- 0: 負論理
- 1: 正論理

Bit6 IntMode

XINT の出力モードを設定します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。

- 0: 1/0 モード
- 1: Hi-z/0 モード

Bit5 DREQ_Level

XDREQ0,1 の論理レベルを設定します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。

- 0: 負論理
- 1: 正論理

Bit4 DACK_Level

XDACK0,1 の論理レベルを設定します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。

- 0: 負論理
- 1: 正論理

Bit3 CS_Mode

DMA0,1 の動作モードを設定します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。

- 0: XDACK0,1 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 XDACK0,1 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。

Bit2 CPU_Endian

16bit mode 時の CPU バスを設定します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。8bit mode 時はこのビットをセットしないで下さい。

- 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。

このビットの設定は、レジスタ書き込み後、B9h 番地をリードする事により有効になります。ChipReset.ResetAll ビットにて、回路のリセットを行った場合、レジスタの値は初期化されますが、その設定が有効になるのは、上述同様に B9h 番地をリードした後になります。

リトルエンディアンの CPU への接続についての詳細およびレジスタマップは、Appendix B を参照してください。

Bit1-0 BusMode, Bus8x16

CPU の動作モードを設定します。このビットは、SLEEP / SNOOZE 中もアクセス有効です。

動作モード	bit1.BusMode	bit0.Bus8x16
16bit Strobe mode	0	0
16bit BE mode	1	*
8bit mode	0	1

7.4.146. B8h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	B8h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.4.147. B9h CPU_ChgEndian (CPU Change Endian)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device / Host	B9h	CPU_ChgEndian	R	7: CPU_ChgEndian[7]	CPU Change Endian	XXh
				6: CPU_ChgEndian[6]		
				5: CPU_ChgEndian[5]		
				4: CPU_ChgEndian[4]		
				3: CPU_ChgEndian[3]		
				2: CPU_ChgEndian[2]		
				1: CPU_ChgEndian[1]		
				0: CPU_ChgEndian[0]		

Bit7-0 CPU_ChgEndian

このレジスタをダミーリードすることにより、ChipConfig.CPU_Endian で設定した Endian が有効になります。
このビットは、SLEEP / SNOOZE 中もアクセス有効です。

7.4.148. BAh~DFh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device / Host	BAh -DFh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5. デバイス・レジスタ詳細説明

7.5.1. E0h *D_SIE_IntStat* (Device SIE Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	E0h	<i>D_SIE_IntStat</i>		7:	0:	1:	00h
			R (W)	6: <i>NonJ</i>	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 のリセット状態を検出すると"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下さい。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下さい。

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下さい。

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_0 レジスタの GoSnooze ビットを "1" にセットすることにより、本 LSI をスヌーズモード(内蔵 PLL 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 ACTIVE60 / ACT_HOST ステートの時、読み出しはできますが、書き込み(割り込み要因クリア)できません。

ACT_DEVICE ステートを出る際には、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、FWIにて以下の処理を行って下さい。

<ACT_DEVICE ステートを出るとき>

- 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)

<ACT_DEVICE ステートに入るとき>

- 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)

7.5.2. E1h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	E1h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.3. E2h D_FIFO_IntStat (Device FIFO Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	E2h	D_FIFO_IntStat	R (W)	7: DescriptorCmp	0: None	1: Descriptor Complete	00h
			R (W)	6: FIFO_IDE_Cmp	0: None	1: FIFO-IDE Complete	
			R (W)	5: FIFO1_Cmp	0: None	1: FIFO1 Complete	
			R (W)	4: FIFO0_Cmp	0: None	1: FIFO0 Complete	
				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	

デバイス 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 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 D_EP0IntStat レジスタの OUT_TransNAK ビットと共に、"1"にセットされます。

Bit6 FIFO_IDE_Cmp

割り込み要因を直接指示します。

IDE にジョインされているエンドポイントが IN 方向の場合、IDE 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。IDE にジョインされているエンドポイントが OUT 方向の場合、IDE の転送が終了すると、"1"にセットされます。

Bit5 FIFO1_Cmp

割り込み要因を直接指示します。

DMA1 にジョインされているエンドポイントが IN 方向の場合、DMA1 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。DMA1 にジョインされているエンドポイントが OUT 方向の場合、DMA1 の転送が終了すると、"1"にセットされます。

Bit4 FIFO0_Cmp

割り込み要因を直接指示します。

DMA0 にジョインされているエンドポイントが IN 方向の場合、DMA0 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。DMA0 にジョインされているエンドポイントが OUT 方向の場合、DMA0 の転送が終了すると、"1"にセットされます。

Bit3 Reserved**Bit2 FIFO_NotEmpty**

割り込み要因を直接指示します。

EPx{x=0,a-c}Join.JoinFIFO_Stat ビットが"1"にセットされているときに、該当するエンドポイントの FIFO 領域にデータが存在する状態(NotEmpty)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。

EPx{x=0,a-c}Join.JoinFIFO_Stat ビットが"1"にセットされているときに、該当するエンドポイントの 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。

EPx{x=0,a-c}Join.JoinFIFO_Stat ビットが"1"にセットされているときに、該当するエンドポイントの FIFO 領域が

Empty になると、"1"にセットされます。

7.5.4. E3h D_BulkIntStat (Device Bulk Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	E3h	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_Cmp

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

7.5.5. E4h D_EPIntStat (Device EPr Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	E4h	D_EPIntStat		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R	2: D_EPIntStat	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 の割り込みを表示します。

Bit7-3 Reserved**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。

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。

7.5.6. E5h D_EP0IntStat (Device EP0 Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	E5h	D_EP0IntStat		7:	0:	1:	00h
			R (W)	6: OUT_ShortACK	0: None	1: OUT Short-Packet ACK	
			R (W)	5: IN_TrانACK	0: None	1: IN Transaction ACK	
			R (W)	4: OUT_TrانACK	0: None	1: OUT Transaction ACK	
			R (W)	3: IN_TrانNAK	0: None	1: IN Transaction NAK	
			R (W)	2: OUT_TrانNAK	0: None	1: OUT Transaction NAK	
			R (W)	1: IN_TrانErr	0: None	1: IN Transaction Error	
			R (W)	0: OUT_TrانErr	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 **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_TrانACK と同時に"1" にセットされます。

Bit5 **IN_TrانACK**

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_TrانACK**

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_TrانNAK**

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_TrانNAK**

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_TrانErr**

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。

Bit0 **OUT_TrانErr**

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

7.5.7. E6h D_EPIntStat (Device EPa Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	E6h	D_EPIntStat		7:	0:	1:	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 EPa 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。

Bit0 **OUT_TranErr**

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。

7.5.8. E7h D_EPbIntStat (Device EPb Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	E7h	D_EPbIntStat		7:	0:	1:
			R (W)	6: OUT_ShortACK	0: None	1: OUT Short Packet ACK
			R (W)	5: IN_TrانACK	0: None	1: IN Transaction ACK
			R (W)	4: OUT_TrانACK	0: None	1: OUT Transaction ACK
			R (W)	3: IN_TrانNAK	0: None	1: IN Transaction NAK
			R (W)	2: OUT_TrانNAK	0: None	1: OUT Transaction NAK
			R (W)	1: IN_TrانErr	0: None	1: IN Transaction Error
			R (W)	0: OUT_TrانErr	0: None	1: OUT Transaction Error

エンドポイント EPb 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_TrانACK と同時に"1" にセットされます。

Bit5 **IN_TrانACK**

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_TrانACK**

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_TrانNAK**

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_TrانNAK**

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_TrانErr**

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。

Bit0 **OUT_TrانErr**

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

7.5.9. E8h D_EPcIntStat (D_EPc Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	E8h	D_EPcIntStat		7:	0:	1:	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 EPc 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。

Bit0 **OUT_TranErr**

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。

7.5.10. E9h~EFh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	E9h -EFh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.11. F0h *D_SIE_IntEnb* (Device SIE Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	F0h	<i>D_SIE_IntEnb</i>		7:	0:	1:	00h
			R / W	6: <i>EnNonJ</i>	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、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 / SNOOZE 中も有効です。

同期ビット(Bit5~0)は ACTIVE60 / ACT_HOST ステートの時、読み出しはできますが、書き込みできません。これら同期ビットの ACT_DEVICE ステートを出る際の処理に関しては、D_SIE_IntStat レジスタの説明をご参照下さい。

7.5.12. F1h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	F1h	Reserved		7:	0:	1:	00h
				6:	0:	1	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.13. F2h D_FIFO_IntEnb (Device FIFO Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	F2h	D_FIFO_IntEnb	R / W	7: EnDescriptorCmp	0: Disable	1: Enable	00h
			R / W	6: EnFIFO_IDE_Cmp	0: Disable	1: Enable	
			R / W	5: EnFIFO1_Cmp	0: Disable	1: Enable	
			R / W	4: EnFIFO0_Cmp	0: Disable	1: Enable	
				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	

D_FIFO_IntStat レジスタの割り込み要因による、DeviceIntStat レジスタの D_FIFO_IntStat ビットのアサートを許可／禁止します。

7.5.14. F3h D_BulkIntEnb (Device Bulk Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	F3h	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、DeviceIntStat レジスタの D_BulkIntStat ビットのアサートを許可／禁止します。

7.5.15. F4h D_EPrintEnb (Device EPr Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	F4h	D_EPrintEnb		7: (Reserved)	0:	1:	00h
				6: (Reserved)	0:	1:	
				5: (Reserved)	0:	1:	
				4: (Reserved)	0:	1:	
				3: (Reserved)	0:	1:	
			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、DeviceIntStat レジスタの D_EPrintStat ビットのアサートを許可／禁止します。

7.5.16. F5h D_EP0IntEnb (Device EP0 Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	F5h	D_EP0IntEnb		7:	0:	1:	00h
			R / W	6: EnOUT_ShortACK	0: Disable	1: Enable	
			R / W	5: EnIN_TransACK	0: Disable	1: Enable	
			R / W	4: EnOUT_TransACK	0: Disable	1: Enable	
			R / W	3: EnIN_TransNAK	0: Disable	1: Enable	
			R / W	2: EnOUT_TransNAK	0: Disable	1: Enable	
			R / W	1: EnIN_TransErr	0: Disable	1: Enable	
			R / W	0: EnOUT_TransErr	0: Disable	1: Enable	

D_EP0IntStat レジスタの割り込み要因による、DeviceIntStat レジスタの D_EP0IntStat ビットのアサートを許可／禁止します。

7.5.17. F6h D_EPaIntEnb (Device EPa Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	F6h	D_EPaIntEnb		7:	0:	1:	00h
			R / W	6: EnOUT_ShortACK	0: Disable	1: Enable	
			R / W	5: EnIN_TransACK	0: Disable	1: Enable	
			R / W	4: EnOUT_TransACK	0: Disable	1: Enable	
			R / W	3: EnIN_TransNAK	0: Disable	1: Enable	
			R / W	2: EnOUT_TransNAK	0: Disable	1: Enable	
			R / W	1: EnIN_TransErr	0: Disable	1: Enable	
			R / W	0: EnOUT_TransErr	0: Disable	1: Enable	

D_EPaIntStat レジスタの割り込み要因による、D_EPrIntStat レジスタの EPaIntStat ビットのアサートを許可／禁止します。

7.5.18. F7h D_EPbIntEnb (Device EPb Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	F7h	D_EPbIntEnb		7:	0:	1:	00h
			R / W	6: EnOUT_ShortACK	0: Disable	1: Enable	
			R / W	5: EnIN_TrانACK	0: Disable	1: Enable	
			R / W	4: EnOUT_TrانACK	0: Disable	1: Enable	
			R / W	3: EnIN_TrانNAK	0: Disable	1: Enable	
			R / W	2: EnOUT_TrانNAK	0: Disable	1: Enable	
			R / W	1: EnIN_TrانErr	0: Disable	1: Enable	
			R / W	0: EnOUT_TrانErr	0: Disable	1: Enable	

D_EPbIntStat レジスタの割り込み要因による、D_EPbIntStat レジスタの EPbIntStat ビットのアサートを許可／禁止します。

7.5.19. F8h D_EPcIntEnb (Device EPc Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	F8h	D_EPcIntEnb		7:	0:	1:	00h
			R / W	6: EnOUT_ShortACK	0: Disable	1: Enable	
			R / W	5: EnIN_TrانACK	0: Disable	1: Enable	
			R / W	4: EnOUT_TrانACK	0: Disable	1: Enable	
			R / W	3: EnIN_TrانNAK	0: Disable	1: Enable	
			R / W	2: EnOUT_TrانNAK	0: Disable	1: Enable	
			R / W	1: EnIN_TrانErr	0: Disable	1: Enable	
			R / W	0: EnOUT_TrانErr	0: Disable	1: Enable	

D_EPcIntStat レジスタの割り込み要因による、D_EPcIntStat レジスタの EPcIntStat ビットのアサートを許可／禁止します。

7.5.20. F9h~FFh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	F9h -FFh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.21. 100h *D_Reset* (Device Reset)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	100h	<i>D_Reset</i>		7:	0:	1:	01h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R / W	0: <i>ResetDTM</i>	0: Do nothing	1: Reset DTM	

デバイスのトランシーバマクロをリセットします。
SLEEP / SNOOZE 時でもアクセス有効です。

Bit7-1 **Reserved**

Bit0 **ResetDTM**

このビットに"1"をセットすると、本 LSI のデバイストランシーバマクロを初期化します。
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。

7.5.22. 101h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	101h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.23. 102h D_NegoControl (Device Nego Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	102h	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。

7.5.24. 103h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	103h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.25. 104h D_ClrAllEPnJoin (Device Clear All EPn Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	104h	D_ClrAllEPnJoin	W	7:ClrJoinIDE	0: Do nothing	1: Clear Join IDE	XXh
			W	6:ClrJoinFIFO_Stat	0: Do nothing	1: Clear Join FIFO_Stat	
				5:	0:	1:	
				4:	0:	1:	
			W	3:ClrJoinDMA1	0: Do nothing	1: Clear Join DMA1	
			W	2:ClrJoinDMA0	0: Do nothing	1: Clear Join DMA0	
			W	1:ClrJoinCPU_Rd	0: Do nothing	1: Clear Join CPU_Rd	
			W	0:ClrJoinCPU_Wr	0: Do nothing	1: Clear Join CPU_Wr	

該当するポートと各エンドポイン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。

エンドポイントがポートに接続(EPx{x=0,a-c}Join 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。

7.5.26. 105h D_XcvrControl (Device Xcvr Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	105h	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] 0: OpMode[0]	OpMode[1:0]		

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 XcvrSelect

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**

UTM 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。

7.5.27. 106h D_USB_Test (Device USB_Test)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	106h	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。

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 EP_x{x=a-c}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 EP_x{x=a-c}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 EP_x{x=a-c}の設定と重複しないようにして下さい。
または、EP_x{x=a-c}Config.EnEndpoint ビットをクリアして下さい。
- 3) EP_x{x=a-c}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。
EP_x{x=a-c}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_TransErr 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。

7.5.28. 107h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	107h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.29. 108h D_EPnControl (Device Endpoint Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	108h	D_EPnControl	W	7: AllForceNAK	0: Do nothing	1: Set All ForceNAK	XXh
			W	6: EPrForceSTALL	0: Do nothing	1: Set EP's ForceSTALL	
			W	5: AllFIFO_Clr	0: Do nothing	1: Clear All FIFO	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			W	0: EP0FIFO_Clr	0: Do nothing	1: Clear EP0 FIFO	

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 の ForceSTALL ビットを"1"にセットします。

Bit5 AllFIFO_Clr

全てのエンドポイントの FIFO がクリアされます。各エンドポイントの領域設定を行った時は、設定終了後に一度必ずこのビットに"1"をセットして、全てのエンドポイントの FIFO をクリアして下さい。このビットは、FIFO クリア完了後自動的に"0"にクリアされます。

エンドポイントに DMA_x{x=0,1}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。

Bit4-1 Reserved**Bit0 EP0FIFO_Clr**

エンドポイント EP0 の FIFO をクリアします。

このビットは、"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。

エンドポイント EP0 に DMA_x{x=0,1}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、このビットを"1"にセットしないで下さい。

7.5.30. 109h D_EPrFIFO_Clr (Device Endpoint FIFO Clear)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	109h	EPrFIFO_Clr		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			W	2: EPcFIFO_Clr	0: Do nothing	1: Clear EPc FIFO	
			W	1: EPbFIFO_Clr	0: Do nothing	1: Clear EPb FIFO	
			W	0: EPaFIFO_Clr	0: Do nothing	1: Clear EPa FIFO	

該当するエンドポイントの 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。

エンドポイントに DMA_x{x=0,1}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。

7.5.31. 10Ah D_BulkOnlyControl (Device BulkOnly Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	10Ah	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

7.5.32. 10Bh D_BulkOnlyConfig (Device BulkOnly Configuration)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	10Bh	D_BulkOnlyConfig		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R / W	2: EPcBulkOnly	0: None	1: Enable BulkOnly on EPc	
			R / W	1: EPbBulkOnly	0: None	1: Enable BulkOnly on EPb	
				0:	0:	1:	

バルクオンリーサポート機能を有効にします。

Bit7-3 Reserved**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。

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。

Bit0 Reserved

7.5.33. 10C~10Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	10Ch -10Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.34.	110h	D_EP0SETUP_0 (Device EP0 SETUP 0)
7.5.35.	111h	D_EP0SETUP_1 (Device EP0 SETUP 1)
7.5.36.	112h	D_EP0SETUP_2 (Device EP0 SETUP 2)
7.5.37.	113h	D_EP0SETUP_3 (Device EP0 SETUP 3)
7.5.38.	114h	D_EP0SETUP_4 (Device EP0 SETUP 4)
7.5.39.	115h	D_EP0SETUP_5 (Device EP0 SETUP 5)
7.5.40.	116h	D_EP0SETUP_6 (Device EP0 SETUP 6)
7.5.41.	117h	D_EP0SETUP_7 (Device EP0 SETUP 7)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	110h	D_EP0SETUP_0	R	7: EPOSETUP_n[7]	Endpoint 0 SETUP Data 0 -Endpoint 0 SETUP Data 7	00h
	-117h	-D_EP0SETUP_7		6: EPOSETUP_n[6]		
				5: EPOSETUP_n[5]		
				4: EPOSETUP_n[4]		
				3: EPOSETUP_n[3]		
				2: EPOSETUP_n[2]		
				1: EPOSETUP_n[1]		
				0: EPOSETUP_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、EPOSETUP_0 から順に格納されます。

EPOSETUP_0

BmRequestType がセットされます。

EPOSETUP_1

BRequest がセットされます。

EPOSETUP_2

Wvalue の下位 8 ビットがセットされます。

EPOSETUP_3

Wvalue の上位 8 ビットがセットされます。

EPOSETUP_4

WIndex の下位 8 ビットがセットされます。

EPOSETUP_5

WIndex の上位 8 ビットがセットされます。

EPOSETUP_6

WLength の下位 8 ビットがセットされます。

EPOSETUP_7

WLength の上位 8 ビットがセットされます。

7.5.42. 118h D_USB_Address (Device USB Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	118h	D_USB_Address	R / W	7: SetAddress	0: none	1: Set USB Address	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に行います。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 SetAddress

SetAddress リクエストの受信時にセットすると、同リクエストのステータスステージが完了した時に、USB_Address が自動的にセットされます。自動アドレス設定モードが無効の場合に、このビットの設定が有効になります。

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。

7.5.43. 119h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	119h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.44. 11Ah D_SETUP_Control(Device SETUP Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	11Ah	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	

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。

7.5.45. 11Bh~11Dh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	11Bh -11Dh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.46. 11Eh D_FrameNumber_H (Device FrameNumber High)

7.5.47. 11Fh D_FrameNumber_L (Device FrameNumber Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	11Eh	D_FrameNumber_H	R	7: FnInvalid	0: Frame number is valid	1: Frame number is not valid	80h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				R	2: FrameNumber[10] 1: FrameNumber[9] 0: FrameNumber[8]	Frame Number High	

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	11Fh	D_FrameNumber_L	R	7: FrameNumber[7]	Frame Number Low	00h
				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。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。

11Eh.Bit7 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。

11Eh.Bit6-3 Reserved**11Eh.Bit2-0, 11Fh.Bit7-0 FrameNumber[10:0]**

受信した SOF パケットの FrameNumber が表示されます。

7.5.48. 120h D_EP0MaxSize (Device EP0 Max Packet Size)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Device	120h	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**

7.5.49. 121h D_EP0Control (Device EP0 Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	121h	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	

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回答して、FIFO から Descriptor データを、MaxPacketSize 分返信します。Descriptor データは、DescAdrs_H,L レジスタの設定値のアドレスを先頭にする、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、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、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。

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_FIFO_IntStat 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。

7.5.50. 122h D_EP0ControlIN (Device EP0 Control IN)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	122h	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。

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 MainIntStat 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、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。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、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、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。

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。

7.5.51. 123h D_EP0ControlOUT (Device EP0 Control OUT)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	123h	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 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、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。

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 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、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、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。

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。

7.5.52. 124h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	124h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.53. 125h D_EP0Join (Device EndPoint0 Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	125h	D_EP0Join		7:	0:	1:	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join EP0 to show Status	
				5:	0:	1:	
				4:	0:	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join EP0 to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join EP0 to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join EP0 to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join EP0 to CPU_Wr	

エンドポイント 0 とデータ転送を行うポートを指定します。

Bit7 **Reserved**

Bit6 **JoinFIFO_Stat**

エンドポイント EP0 の FIFO の Full、Empty 及び NotEmpty の状態を、D_FIFO_IntStat.FIFO_Full、D_FIFO_IntStat.FIFO_Empty 及び D_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 **Reserved**

Bit3 **JoinDMA1**

エンドポイント EP0 の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 **JoinDMA0**

エンドポイント EP0 の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 **JoinCPU_Rd**

エンドポイント EP0 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、このエンドポイントの FIFO からデータが読み出されます。

Bit0 **JoinCPU_Wr**

エンドポイント EP0 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、このエンドポイントの FIFO にデータが書き込まれます。

JoinDMAx {x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.5.54. 126h~12Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	126h -12Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.55. 130h D_EPaMaxSize_H (Device EPa Max Packet Size High)

7.5.56. 131h D_EPaMaxSize_L (Device EPa Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	130h	D_EPaMaxSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				R / W	1: MaxSize[9] 0: MaxSize[8]	Endpoint[a] Max Packet Size	

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	131h	D_EPaMaxSize_L	R / W	7: MaxSize[7]	Endpoint[a] Max Packet Size	00h	
				6: MaxSize[6]			
				5: MaxSize[5]			
				4: MaxSize[4]			
				3: MaxSize[3]			
				2:	0:		1:
				1:	0:		1:
				0:	0:		1:

MaxPacketSize を設定します。

130h.Bit7-2 Reserved**131h.Bit1-0, 131h.Bit7-3 EPaMaxSize[9:3]**

エンドポイント EPa 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 512 バイトまで

の任意の転送数が設定可能です。

131h.Bit2-0 Reserved

7.5.57. 132h D_EPaConfig_0 (Device EPa Configuration 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	132h	D_EPaConfig_0	R / W	7: INxOUT	0: OUT 1: IN	00h
			R / W	6: IntEP_Mode	0: Normal Toggle (IN) 0: Bulk OUT (OUT) 1: Interrupt OUT (OUT)	
			R / W	5: EnEndpoint	0: Disable Endpoint 1: Enable Endpoint	
				4:	0: 1:	
			R / W	3: EndpointNumber[3] 2: EndpointNumber[2] 1: EndpointNumber[1] 0: EndpointNumber[0]	Endpoint Number	

エンドポイント EPa 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 EnEndpoint

このビットを"1"にセットすることで、このエンドポイントを有効にします。

このビットが"0"の時は、エンドポイントへのアクセスを無視します。

ホストからの SetConfiguration リクエストに従って設定して下さい。

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。

7.5.58. 133h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	133h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.59. 134h D_EPaControl (Device EPa Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	134h	D_EPaControl	R / W	7: AutoForceNAK	0: Do nothing	1: Auto Force NAK	00h
			R / W	6: EnShortPkt	0: Do nothing	1: Enable Short Packet	
			R / W	5: DisAF_NAK_Short	0: Auto Force NAK Short	1: Disable Auto Force	
			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 EPa の動作設定を行います。

Bit7 AutoForceNAK

エンドポイント EPa 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 EPa 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 EPa 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 EPa 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 EPa 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 EPa 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 EPa 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。

7.5.60. 135h D_EPaJoin (Device End Point a Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	135h	D_EPaJoin	R / W	7: JoinIDE	0: Do nothing	1: Join EPa to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join EPa to show Status	
				5:	0:	1:	
				4:	0:	1	
			R / W	3: JoinDMA1	0: Do nothing	1: Join EPa to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join EPa to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join EPa to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join EPa to CPU_Wr	

エンドポイント EPa とデータ転送を行うポートを指定します。

Bit7 JoinIDE

エンドポイント EPa の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

エンドポイント EPa の FIFO の Full、Empty 及び NotEmpty の状態を、D_FIFO_IntStat.FIFO_Full、D_FIFO_IntStat.FIFO_Empty 及び D_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

エンドポイント EPa の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

エンドポイント EPa の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

エンドポイント EPa 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、このエンドポイントの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

エンドポイント EPa 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、このエンドポイントの FIFO にデータが書き込まれます。

JoinDMAx {x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.5.61. 136h~13Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	136h -13Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.62. 140h D_EPbMaxSize_H (Device EPb Max Packet Size High)

7.5.63. 141h D_EPbMaxSize_L (Device EPb Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	140h	D_EPbMaxSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				R / W	1: MaxSize[9] 0: MaxSize[8]	Endpoint[b] Max Packet Size	

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	141h	D_EPbMaxSize_L	R / W	7: MaxSize[7]	Endpoint[b] Max Packet Size		00h
				6: MaxSize[6]			
				5: MaxSize[5]			
				4: MaxSize[4]			
				3: MaxSize[3]			
				2:	0:	1:	
				1:	0:	1:	
0:	0:	1:					

MaxPacketSize を設定します。

140h.Bit7-2 **Reserved**

141h.Bit1-0, 141h.Bit7-3 **EPbMaxSize[9:3]**

エンドポイント EPb 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 512 バイトまで

の任意の転送数が設定可能です。

141h.Bit2-0 **Reserved**

7.5.64. 142h D_EPbConfig_0 (Devie EPb Configuration 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	142h	D_EPbConfig_0	R / W	7: INxOUT	0: OUT	1: IN	00h
			R / W	6: IntEP_Mode	0: Normal Toggle (IN) 0: Bulk OUT (OUT)	1: Always Toggle (IN) 1: Interrupt OUT (OUT)	
			R / W	5: EnEndpoint	0: Disable Endpoint	1: Enable Endpoint	
				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 EPb 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 EnEndpoint

このビットを"1"にセットすることで、このエンドポイントを有効にします。

このビットが"0"の時は、エンドポイントへのアクセスを無視します。

ホストからの SetConfiguration リクエストに従って設定して下さい。

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。

7.5.65. 143h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	143h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.66. 144h D_EPbControl (Device EPb Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	144h	D_EPbControl	R / W	7: AutoForceNAK	0: Do nothing	1: Auto Force NAK	00h
			R / W	6: EnShortPkt	0: Do nothing	1: Enable Short Packet	
			R / W	5: DisAF_NAK_Short	0: Auto Force NAK Short	1 Disable Auto Force	
			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 EPb の動作設定を行います。

Bit7 AutoForceNAK

エンドポイント EPb 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 EPb 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 EPb 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 EPb 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 EPb 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 EPb 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 EPb 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。

7.5.67. 145h D_EPbJoin (Device End Point b Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	145h	D_EPbJoin	R / W	7: JoinIDE	0: Do nothing	1: Join EPb to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join EPb to show Status	
				5:	0:	1:	
				4:	0:	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join EPb to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join EPb to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join EPb to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join EPb to CPU_Wr	

エンドポイント EPb とデータ転送を行うポートを指定します。

Bit7 JoinIDE

エンドポイント EPb の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

エンドポイント EPb の FIFO の Full、Empty 及び NotEmpty の状態を、D_FIFO_IntStat.FIFO_Full、D_FIFO_IntStat.FIFO_Empty 及び D_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

エンドポイント EPb の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

エンドポイント EPb の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

エンドポイント EPb 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、このエンドポイントの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

エンドポイント EPb 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、このエンドポイントの FIFO にデータが書き込まれます。

JoinDMAx {x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.5.68. 146h~14Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	146h -14Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.69. 150h D_EPcMaxSize_H (Device EPc Max Packet Size High)

7.5.70. 151h D_EPcMaxSize_L (Device EPc Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	150h	D_EPcMaxSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				R / W	1: MaxSize[9] 0: MaxSize[8]	Endpoint[c] Max Packet Size	

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	151h	D_EPcMaxSize_L	R / W	7: MaxSize[7]	Endpoint[c] Max Packet Size	00h	
				6: MaxSize[6]			
				5: MaxSize[5]			
				4: MaxSize[4]			
				3: MaxSize[3]			
				2:	0:		1:
				1:	0:		1:
				0:	0:		1:

MaxPacketSize を設定します。

150h.Bit7-2 Reserved

150h.Bit1-0, 151h.Bit7-3 EPcMaxSize[9:3]

エンドポイント EPc 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 512 バイトまで

の任意の転送数が設定可能です。

151h.Bit2-0 Reserved

7.5.71. 152h D_EPcConfig_0 (Device EPc Configuration 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	152h	D_EPcConfig_0	R / W	7: INxOUT	0: OUT 1: IN	00h
			R / W	6: IntEP_Mode	0: Normal Toggle (IN) 0: Bulk OUT (OUT) 1: Interrupt OUT (OUT)	
			R / W	5: EnEndpoint	0: Disable Endpoint 1: Enable Endpoint	
				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 EPc 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 EnEndpoint

このビットを"1"にセットすることで、このエンドポイントを有効にします。

このビットが"0"の時は、エンドポイントへのアクセスを無視します。

ホストからの SetConfiguration リクエストに従って設定して下さい。

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。

7.5.72. 153h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	153h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.73. 154h D_EPcControl (Device EPc Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	154h	D_EPcControl	R / W	7: AutoForceNAK	0: Do nothing	1: Auto Force NAK	00h
			R / W	6: EnShortPkt	0: Do nothing	1: Enable Short Packet	
			R / W	5: DisAF_NAK_Short	0: Auto Force NAK Short	1 Disable Auto Force	
			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 EPc の動作設定を行います。

Bit7 AutoForceNAK

エンドポイント EPc 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 EPc 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 EPc 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 EPc 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 EPc 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 EPc 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 EPc 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。

7.5.74. 155h D_EPcJoin (Device End Point c Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	155h	D_EPcJoin	R / W	7: JoinIDE	0: Do nothing	1: Join EPc to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join EPc to show Status	
				5:	0:	1:	
				4:	0:	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join EPc to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join EPc to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join EPc to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join EPc to CPU_Wr	

エンドポイント EPc とデータ転送を行うポートを指定します。

Bit7 JoinIDE

エンドポイント EPc の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

エンドポイント EPc の FIFO の Full、Empty 及び NotEmpty の状態を、D_FIFO_IntStat.FIFO_Full、D_FIFO_IntStat.FIFO_Empty 及び D_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

エンドポイント EPc の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

エンドポイント EPc の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

エンドポイント EPc 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、このエンドポイントの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

エンドポイント EPc 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、このエンドポイントの FIFO にデータが書き込まれます。

JoinDMAx {x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.5.75. 156h~15Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	156h -15Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.76. 160h D_DescAdrs_H (Device Descriptor Address High)

7.5.77. 161h D_DescAdrs_L (Device Descriptor Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Device	160h	D_DescAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				4:	0:	1:		
			R / W	3: DescAdrs[11]	Descriptor Address			
				2: DescAdrs[10]				
				1: DescAdrs[9]				
				0: DescAdrs[8]				

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	161h	D_DescAdrs_L	R / W	7: DescAdrs[7]	Descriptor Address	00h
				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。

160h.Bit7-4 **Reserved**160h.Bit3-0, 161h.Bit7-0 **DescAdrs[11:0]**

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 領域を割り当ててものではありません。Descriptor Address は、FIFO の領域設定に関わらず、0x000 から 0x9FF(2.5kByte)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。エンドポイント EP0 の予約された領域の終了アドレス(0x040)から CBW 領域の先頭アドレス(0x190)までの間が適切です。

Descriptor Address を参照する場合は、D_DescAdrs_H,D_DescAdrs_L の順に読み出して下さい。

7.5.78. 162h D_DescSize_H (Device Descriptor Size High)

7.5.79. 163h D_DescSize_L (Device Descriptor Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	162h	D_DescSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				R / W	1: DescSize[9] 0: DescSize[8]	Descriptor Size	

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	163h	D_DescSize_L	R / W	7: DescSize[7]	DescriptorSize	00h
				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。

162h.Bit7-2 Reserved

162h.Bit1-0, 163h.Bit7-0 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、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できます。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、DescAdrs_H,L レジスタと 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。エンドポイント EP0 の予約された領域の終了アドレス(0x040)から CBW 領域の先頭アドレス(0x190)までの間を使用するようにして下さい。

Descriptor Size を参照する場合は、DescSize_H,DescSize_L の順に読み出して下さい。

7.5.80. 164h~16Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	164h -16Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.81. 170h D_DMA0_FIFO_Control (Device DMA0 FIFO Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	170h	D_ DMA0_FIFO_Control	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:	

DMA0 転送時の、FIFO の状態の表示及び設定を行います。

Bit7 FIFO_Running

DMA0 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。DMA0 を起動すると"1"にセットされ、DMA0 が終了した後、FIFO が空になると"0"にクリアされます。

Bit6 AutoEnShort

DMA0 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。

DMA0 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

7.5.82. 171h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	171h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.83. 172h D_DMA1_FIFO_Control (Device DMA1 FIFO Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	172h	D_ DMA1_FIFO_Control	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:	

DMA1 転送時の、FIFO の状態の表示及び設定を行います。

Bit7 FIFO_Running

DMA1 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。DMA1 を起動すると"1"にセットされ、DMA1 が終了した後、FIFO が空になると"0"にクリアされます。

Bit6 AutoEnShort

DMA1 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。

DMA1 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

7.5.84. 173h~17Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	173h -17Fh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.85. 180h~183h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	180h -183h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.86. 184h D_EPaStartAdrs_H (Device Endpoint a Start Address High)

7.5.87. 185h D_EPaStartAdrs_L (Device Endpoint a Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset				
Device	184h	D_EPaStartAdrs_H		7:	0:	1:	00h			
				6:	0:	1:				
				5:	0:	1:				
			R / W	4: StartAdrs[12]	EPa Start Address High					
				3: StartAdrs[11]						
				2: StartAdrs[10]						
				1: StartAdrs[9]						
	0: StartAdrs[8]									

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset			
Device	185h	D_EPaStartAdrs_L	R / W	7: StartAdrs[7]	EPa Start Address Low		00h		
				6: StartAdrs[6]					
				5: StartAdrs[5]					
				4: StartAdrs[4]					
				3: StartAdrs[3]					
				2: StartAdrs[2]					
				1:					
				0:					

デバイス動作時にチャンネル EPa で使用する FIFO の領域設定を行います。

184h.Bit7-5

Reserved

184h.Bit4-0, 185h.Bit7-2

StartAdrs[12:2]

エンドポイント EPa 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。

チャンネル EPa に割り当てられる領域は、EPbStartAdrs にて設定されたアドレスの 1 バイト前までとなります。

EPa StartAdrs、EPb StartAdrs を設定した後は、必ず D_EPbFIFO_Clr レジスタの EPaFIFO_Clr ビットを"1"にしてエンドポイント EPa の FIFO をクリアして下さい。

なお、ここで設定した領域より、EPa の MaxSize が大きい場合には、正常に動作しません。

全エンドポイントで確保する FIFO 領域、EP0 エリア、デスクリプタエリア、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

185h.Bit1-0

Reserved

7.5.88. 186h~187h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	186h -187h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.89. 188h D_EPbStartAdrs_H (Device Endpoint b Start Address High)**7.5.90. 189h D_EPbStartAdrs_L (Device Endpoint b Start Address Low)**

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Device	188h	D_EPbStartAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
			R / W	4: StartAdrs[12]	EPb Start Address High			
				3: StartAdrs[11]				
				2: StartAdrs[10]				
				1: StartAdrs[9]				
	0: StartAdrs[8]							

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	189h	D_EPbStartAdrs_L	R / W	7: StartAdrs[7]	EPb Start Address Low		00h
				6: StartAdrs[6]			
				5: StartAdrs[5]			
				4: StartAdrs[4]			
				3: StartAdrs[3]			
				2: StartAdrs[2]			
				1:			
	0:						

デバイス動作時にチャンネル EPb で使用する FIFO の領域設定を行います。

188h.Bit7-5 Reserved**188h.Bit4-0, 189h.Bit7-2 StartAdrs[12:2]**

エンドポイント EPb 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。

チャンネル EPb に割り当てられる領域は、EPcStartAdrs にて設定されたアドレスの 1 バイト前までとなります。

EPb StartAdrs、EPc StartAdrs を設定した後は、必ず D_EPbFIFO_Clr レジスタの EPbFIFO_Clr ビットを"1"にしてエンドポイント EPb の FIFO をクリアして下さい。

なお、ここで設定した領域より、EPb の MaxSize が大きい場合には、正常に動作しません。

全エンドポイントで確保する FIFO 領域、EP0 エリア、デスクリプタエリア、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

189h.Bit1-0 Reserved

7.5.91. 18Ah~18Bh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	18Ah -18Bh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.5.92. 18Ch D_EPcStartAdrs_H (Device Endpoint c Start Address High)

7.5.93. 18Dh D_EPcStartAdrs_L (Device Endpoint c Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	18Ch	D_EPcStartAdrs_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R / W	4: StartAdrs[12]	EPc Start Address High		
				3: StartAdrs[11]			
				2: StartAdrs[10]			
				1: StartAdrs[9]			
0: StartAdrs[8]							

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Device	18Dh	D_EPcStartAdrs_L	R / W	7: StartAdrs[7]	EPc Start Address Low	00h
				6: StartAdrs[6]		
				5: StartAdrs[5]		
				4: StartAdrs[4]		
				3: StartAdrs[3]		
				2: StartAdrs[2]		
				1:		
	0:					

デバイス動作時にチャンネル EPc で使用する FIFO の領域設定を行います。

18Ch.Bit7-5 **Reserved**

18Ch.Bit4-0, 18Dh.Bit7-2 **StartAdrs[12:2]**

エンドポイント EPc 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。

18Dh.Bit1-0 **Reserved**

7.5.94. 18Eh D_EPcEndAdrs_H (Device Endpoint c End Address High)

7.5.95. 18Fh D_EPcEndAdrs_L (Device Endpoint c End Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	18Ch	D_EPcEndAdrs_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R / W	4: EndAdrs[12]	EPc End Address High		
				3: EndAdrs[11]			
				2: EndAdrs[10]			
1: EndAdrs[9]							
	0: EndAdrs[8]						

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	18Dh	D_EPcEndAdrs_L	R / W	7: EndAdrs[7]	EPc End Address Low		00h
				6: EndAdrs[6]			
				5: EndAdrs[5]			
				4: EndAdrs[4]			
				3: EndAdrs[3]			
				2: EndAdrs[2]			
				1:			
				0:			

デバイス動作時にチャンネル EPc で使用する FIFO の領域設定を行います。

18Ch.Bit7-5 **Reserved**18Ch.Bit4-0, 18Dh.Bit7-2 **EndAdrs[12:2]**

エンドポイント EPc に割り当てる FIFO の終端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 64 バイト単位での指定になります。

チャンネル EPc に割り当てられる領域は、EPcStartAdrs にて設定されたアドレスから、EPcEndAdrs にて設定されたアドレスの 1 バイト前までとなります。

EPc StartAdrs、EPc EndAdrs を設定した後は、必ず D_EPcFIFO_Clr レジスタの EPcFIFO_Clr ビットを"1"にしてエンドポイント EPc の FIFO をクリアして下さい。

なお、ここで設定した領域より、EPc の MaxSize が大きい場合には、正常に動作しません。

全エンドポイントで確保する FIFO 領域、EP0 エリア、デスク립タエリア、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

18Dh.Bit1-0 **Reserved**

7.5.96. 190h~1FFh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	190h -1FFh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6. ホスト・レジスタ詳細説明

7.6.1. E0h H_SIE_IntStat_0 (Host SIE Interrupt Status 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	E0h	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。

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。

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。

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。

同期ビット(Bit4~0)は ACTIVE60 / ACT_DEVICE ステートの時、読み出しはできますが、書き込み(割り込み要因クリア)できません。

ACT_HOST ステートを出る際には、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。

<ACT_HOST ステートを出るとき>

- 1) 割り込みステータスを処理し、クリアする(H_SIE_IntStat_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)

<ACT_HOST ステートに入るとき>

- 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)

7.6.2. E1h H_SIE_IntStat_1 (SIE Host Interrupt Status 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	E1h	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 ACTIVE60 / ACT_DEVICE ステートの時、読み出しはできますが、書き込み(割り込み要因クリア)できません。

ACT_HOST ステートを出る際には、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。

<ACT_HOST ステートを出るとき>

- 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)

<ACT_HOST ステートに入るとき>

- 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)

7.6.3. E2h H_FIFO_IntStat (Host FIFO Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	E2h	H_FIFO_IntStat		7:	0:	1:	00h
			R (W)	6: FIFO_IDE_Cmp	0: None	1: FIFO-IDE Complete	
			R (W)	5: FIFO1_Cmp	0: None	1: FIFO1 Complete	
			R (W)	4: FIFO0_Cmp	0: None	1: FIFO0 Complete	
				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	

ホストの FIFO 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 **FIFO_IDE_Cmp**

割り込み要因を直接指示します。

IDE にジョインされているチャンネルが OUT 方向の場合、IDE 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。IDE にジョインされているチャンネルが IN 方向の場合、IDE の転送が終了すると、"1"にセットされます。

Bit5 **FIFO1_Cmp**

割り込み要因を直接指示します。

DMA1 にジョインされているチャンネルが OUT 方向の場合、DMA1 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。DMA1 にジョインされているチャンネルが IN 方向の場合、DMA1 の転送が終了すると、"1"にセットされます。

Bit4 **FIFO0_Cmp**

割り込み要因を直接指示します。

DMA0 にジョインされているチャンネルが OUT 方向の場合、DMA0 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。DMA0 にジョインされているチャンネルが IN 方向の場合、DMA0 の転送が終了すると、"1"にセットされます。

Bit3 **Reserved**

Bit2 **FIFO_NotEmpty**

割り込み要因を直接指示します。

H_CHX{x=0,a-e}Join.JoinFIFO_Stat ビットが"1"にセットされているときに、該当するチャンネルの FIFO 領域にデータが存在する状態(NotEmpty)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。

H_CHX{x=0,a-e}Join.JoinFIFO_Stat ビットが"1"にセットされているときに、該当するチャンネルの 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。

H_CHX{x=0,a-e}Join.JoinFIFO_Stat ビットが"1"にセットされているときに、該当するチャンネルの FIFO 領域が Empty になると、"1"にセットされます。

7.6.4. E3h H_FrameIntStat (Host Frame Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	E3h	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。

Bit1 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ト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

7.6.5. E4h H_CHrIntStat (Host CHr Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	E4h	H_CHrIntStat		7:	0:	1:
				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。

7.6.6. E5h H_CH0IntStat (Host CH0 Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	E5h	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 単位のデータ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が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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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。

7.6.7. E6h H_CHaIntStat (Host CHa Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	E6h	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が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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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イクを発信しない(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)またはハンドシェイク(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_BO_SupportControl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_BO_SupportControl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。

7.6.8. E7h H_CHbIntStat (Host CHb Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	E7h	H_ChbIntStat	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:	
				1:	0:	1:	
				0:	0:	1:	

チャンネル CHb 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_CHbConfig_0.ACK_Cnt ビットに設定した回数の個別トランザクションが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_ChbConfig_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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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)
その他	Reserved	

Bit3-0 Reserved

7.6.9. E8h H_CHcIntStat (Host CHc Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	E8h	H_CHcIntStat	R (W)	7: TotalSizeCmp	0: None	1: TotalSize Complete	00h
			R (W)	6: TranACK	0: None	1: TranACK	
			R (W)	5: TranErr	0: None	1: Tran Error	
			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 CHc 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_CHcConfig_0.ACK_Cnt ビットに設定した回数の個別トランザクションが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_CHcConfig_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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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

7.6.10. E9h H_CHdIntStat (Host CHd Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	E9h	H_CHdIntStat	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:	
				1:	0:	1:	
				0:	0:	1:	

チャンネル CHd 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_CHdConfig_0.ACK_Cnt ビットに設定した回数の個別トランザクションが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_ChdConfig_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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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

7.6.11. EAh H_CHeIntStat (Host CHe Interrupt Status)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	EAh	H_CHeIntStat	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:	
				1:	0:	1:	
				0:	0:	1:	

チャンネル CHe 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_CHeConfig_0.ACK_Cnt ビットに設定した回数の個別トランザクションが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_CheConfig_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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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

7.6.12. EBh~EFh Reserved()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	EBh -EFh	Reserved		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.13. F0h H_SIE_IntEnb_0 (Host SIE Interrupt Enable 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	F0h	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、HostIntStat レジスタの H_SIE_IntStat_0 ビットのアサートを許可／禁止します。

同期ビット (Bit4~0) は ACTIVE60 / ACT_DEVICE ステートの時、読み出しはできますが、書き込みできません。これら同期ビットの ACT_HOST ステートを出す際の処理に関しては、H_SIE_IntStat_0 レジスタの説明をご参照下さい。

7.6.14. F1h H_SIE_IntEnb_1 (SIE Host Interrupt Enable 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	F1h	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、HostIntStat 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。
同期ビット (Bit3~0) は ACTIVE60 / ACT_DEVICE ステートの時、読み出しはできますが、書き込みできません。これら同期ビットの ACT_HOST ステートを出る際の処理に関しては、H_SIE_IntStat_1 レジスタの説明をご参照下さい。

7.6.15. F2h H_FIFO_IntEnb (Host FIFO Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	F2h	H_FIFO_IntEnb		7:	0:	1:	00h
			R/W	6: EnFIFO_IDE_Cmp	0: Disable	1: Enable	
			R/W	5: EnFIFO1_Cmp	0: Disable	1: Enable	
			R/W	4: EnFIFO0_Cmp	0: Disable	1: Enable	
				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	

H_FIFO_IntStat レジスタの割り込み要因による、HostIntStat レジスタの H_FIFO_IntStat ビットのアサートを許可／禁止します。

7.6.16. F3h H_FrameIntEnb (Host Frame Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	F3h	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、HostIntStat レジスタの H_FrameIntStat ビットのアサートを許可／禁止します。

7.6.17. F4h H_CHrIntEnb (Host CHr Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset	
Host	F4h	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、HostIntStat レジスタの H_CHrIntStat ビットのアサートを許可／禁止します。

7.6.18. F5h H_CH0IntEnb (Host CH0 Interrupt Enable)

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	F5h	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、HostIntStat レジスタの H_CH0IntStat ビットのアサートを許可／禁止します。

7.6.19. F6h H_CHalntEnb (Host CHa Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	F6h	H_CHalntEnb	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_CHalntStat レジスタの割り込み要因による、H_CHRIntStat レジスタの CHalntStat ビットのアサートを許可／禁止します。

7.6.20. F7h H_CHbIntEnb (Host CHb Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	F7h	H_CHbIntEnb	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:	
				1:	0:	1:	
				0:	0:	1:	

H_CHbIntStat レジスタの割り込み要因による、H_CHrIntStat レジスタの CHbIntStat ビットのアサートを許可／禁止します。

7.6.21. F8h H_CHcIntEnb (Host CHc Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	F8h	H_CHcIntEnb	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:	
				1:	0:	1:	
				0:	0:	1:	

H_CHcIntStat レジスタの割り込み要因による、H_CHrIntStat レジスタの CHcIntStat ビットのアサートを許可／禁止します。

7.6.22. F9h H_CHdIntEnb (Host CHd Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	F9h	H_CHdIntEnb	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:	
				1:	0:	1:	
				0:	0:	1:	

H_CHdIntStat レジスタの割り込み要因による、H_CHrIntStat レジスタの CHdIntStat ビットのアサートを許可／禁止します。

7.6.23. FAh H_CHeIntEnb (Host CHe Interrupt Enable)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	FAh	H_CHeIntEnb	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:	
				1:	0:	1:	
				0:	0:	1:	

H_CHeIntStat レジスタの割り込み要因による、H_CHrIntStat レジスタの CHeIntStat ビットのアサートを許可／禁止します。

7.6.24. FB h ~FFh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	FBh -FFh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.25. 100h *H_Reset (Host Reset)*

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	100h	<i>H_Reset</i>		7:	0:	1:	01h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			R / W	0: <i>ResetHTM</i>	0: Do nothing	1: Reset HTM	

ホストのトランシーバマクロをリセットします。
SLEEP / SNOOZE 時でもアクセス有効です。

Bit7-1 **Reserved**

Bit0 **ResetHTM**

このビットに"1"をセットすると、本 LSI のホストトランシーバマクロを初期化します。
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。

7.6.26. 101h Reserved ()

Mode	Address	Register Name	R/W	Bit Symbol	Description		Reset
Host	101h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.27. 102h H_NegoControl_0 (Host NegoControl 0)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	102h	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_XcvtControl 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・リモートウェイクアップの各検出機能を OFF します)。

- ・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、
 - ・H_USB_Test.EnHS_Test によりテストモードを実行するとき、
- 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。

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

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。

7.6.28. 103h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	103h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.29. 104h H_NegoControl_1 (Host NegoControl 1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	104h	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 ACT_HOST 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

リモートウェークアップ検出機能の有効／無効を設定します。

7.6.30. 105h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	105h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.31. 106h H_USB_Test (Host USB_Test)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset	
Host	106h	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_Packet	

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 のいずれかの状態で有効です。

これらの状態からテストモードに入る前には、各状態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_BO_SupportControl.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 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、ホスト状態は 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"を設定すると、下位 5 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。それ以外は CH0 に関する設定は必要ありません。

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。

7.6.32. 107h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	107h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.33. 108h H_CHnControl (Host CHn Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	108h	H_CHnControl		7:	0:	1:	XXh
				6:	0:	1:	
			W	5: AllFIFO_Clr	0: Do nothing	1: Clear All FIFO	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
			W	0: CH0FIFO_Clr	0: Do nothing	1: Clear EP0 FIFO	

チャンネルの動作設定を行います。ライトオンリーのレジスタです。

Bit7-6 Reserved

Bit5 AllFIFO_Clr

全てのチャンネルのFIFOがクリアされます。各チャンネルの領域設定を行った時は、設定終了後に一度必ずこのビットに"1"をセットして、全てのチャンネルのFIFOをクリアして下さい。このビットは、FIFOクリア完了後自動的に"0"にクリアされます。

チャンネルにDMA_x{x=0,1}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、該当するチャンネルのビットを"1"にセットしないで下さい。

Bit4-1 Reserved

Bit0 CH0FIFO_Clr

チャンネルCH0のFIFOをクリアします。

このビットは、"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。

チャンネルCH0にDMA_x{x=0,1}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、このビットを"1"にセットしないで下さい。

7.6.34. 109h H_CHrFIFO_Clr (Host CHr FIFO Clear)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	109h	H_CHrFIFO_Clr		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
			W	4: CHeFIFO_Clr	0: Do nothing	1: Clear CHe FIFO	
			W	3: CHdFIFO_Clr	0: Do nothing	1: Clear CHd FIFO	
			W	2: CHcFIFO_Clr	0: Do nothing	1: Clear CHc FIFO	
			W	1: CHbFIFO_Clr	0: Do nothing	1: Clear CHb FIFO	
W	0: CHaFIFO_Clr	0: Do nothing	1: Clear CHa FIFO				

該当するチャンネルの 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。

エンドポイントに DMA_x{x=0,1}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。

7.6.35. 10Ah H_ClrAllCHnJoin (Host Clear All CHn Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	10Ah	H_ClrAllCHnJoin	W	7:ClrJoinIDE	0: Do nothing	1: Clear Join IDE	XXh
				6:ClrJoinFIFO_Stat	0: Do nothing	1: Clear Join FIFO_Stat	
				5:	0:	1:	
				4:	0:	1:	
				3:ClrJoinDMA1	0: Do nothing	1: Clear Join DMA1	
				2:ClrJoinDMA0	0: Do nothing	1: Clear Join DMA0	
				1:ClrJoinCPU_Rd	0: Do nothing	1: Clear Join CPU_Rd	
				0:ClrJoinCPU_Wr	0: Do nothing	1: Clear Join CPU_Wr	

該当するポートと各チャン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。

チャンネルがポートに接続(CHx{x=0,a-e}Join 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。

7.6.36. 10B~10Fh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Device	10Bh -10F	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.37.	110h H_CH0SETUP_0 (Host CH0 SETUP 0)
7.6.38.	111h H_CH0SETUP_1 (Host CH0 SETUP 1)
7.6.39.	112h H_CH0SETUP_2 (Host CH0 SETUP 2)
7.6.40.	113h H_CH0SETUP_3 (Host CH0 SETUP 3)
7.6.41.	114h H_CH0SETUP_4 (Host CH0 SETUP 4)
7.6.42.	115h H_CH0SETUP_5 (Host CH0 SETUP 5)
7.6.43.	116h H_CH0SETUP_6 (Host CH0 SETUP 6)
7.6.44.	117h H_CH0SETUP_7 (Host CH0 SETUP 7)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset
Host	110h -117h	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。

7.6.45. 118h~11Dh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	118h -11Dh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.46. 11Eh H_FrameNumber_H (Host FrameNumber High)

7.6.47. 11Fh H_FrameNumber_L (Host FrameNumber Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	11Eh	H_FrameNumber_H		7:	0:	1:	07h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				R/W	2: FrameNumber[10] 1: FrameNumber[9] 0: FrameNumber[8]	Frame Number High	

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	11Fh	H_FrameNumber_L	R/W	7: FrameNumber[7]	Frame Number Low	FFh
				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。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。

(注)本レジスタの Reset 値は ACT_HOST 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。

11Eh.Bit7-3 Reserved

11Eh.Bit2-0, 11Fh.Bit7-0 FrameNumber[10:0]

送信する SOF パケットの FrameNumber が表示されます。

7.6.48. 120h H_CH0Config_0 (Host Channel 0 Configuration0)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	120h	H_CH0Config_0	R / W	7: ACK_Cnt[3]	Channel 0 ACK Count	00h
				6: ACK_Cnt[2]		
				5: ACK_Cnt[1]		
				4: ACK_Cnt[0]		
			R / W	3: SpeedMode[1]	Channel 0 Speed Mode	
				2: SpeedMode[0]		
			R / W	1: Toggle	0: Toggle0	
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。

7.6.49. 121h H_CH0Config_1 (Host Channel 0 Configuration1)

Mode	Address	Register Name	R/W	Bit Symbol	Description	Reset	
Host	121h	H_CH0Config_1	R/W	7: TID[1]	Channel 0 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

7.6.50. 122h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	122h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.51. 123h H_CH0MaxPktSize (Host Channel 0 Max Packet Size)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	123h	H_CH0MaxPktSize		7:	0:	1:	00h	
			R / W	6: MaxPktSize[6]	Channel 0 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。

7.6.52. 124h~125h Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	124h -125h	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.53. 126h H_CH0TotalSize_H (Host Channel 0 Total Size High)

7.6.54. 127h H_CH0TotalSize_L (Host Channel 0 Total Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	126h	H_CH0TotalSize_H	R / W	7: TotalSize[15]	Channel 0 Total Size High	00h
				6: TotalSize[14]		
				5: TotalSize[13]		
				4: TotalSize[12]		
				3: TotalSize[11]		
				2: TotalSize[10]		
				1: TotalSize[9]		
				0: TotalSize[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	127h	H_CH0TotalSize_L	R / W	7: TotalSize[7]	Channel 0 Total Size Low	00h
				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。

118h.Bit7-0, 119h.Bit7-0 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、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。

bit 15~8 (H_CH0TotalSize_H レジスタ)をリードすると bit 7~0 (H_CH0TotalSize_L レジスタ)の値が固定されます。(リード値が固定されても IC 内部ではカウントは継続されます)

よって、レジスタ 8bit アクセス時で残り転送数をリードする場合は、H_CH0TotalSize_H、H_CH0TotalSize_L レジスタの順番にアクセスしてください。

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。

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。

7.6.55. 128h H_CH0HubAdrs (Host Channel 0 Hub Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	128h	H_CH0HubAdrs	R / W	7: HubAdrs[3]	Channel 0 Hub Address	00h	
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:		1:
			R / W	2: Port[2]	Channel 0 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。

Bit7-4 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。

7.6.56. 129h H_CH0FuncAdrs (Host Channel 0 Function Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	129h	H_CH0FuncAdrs	R / W	7: FuncAdrs[3]	Channel 0 Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R / W	3: EP_Number[3]	Channel 0 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。

7.6.57. 12Ah Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	12Ah	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.58. 12Bh CTL_SupportControl (Host ControlTransfer Support Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	12Bh	H_CTL_SupportControl		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。

7.6.59. 12Ch~12Dh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	12Ch -12Dh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.60. 12Eh H_CH0ConditionCode (Host Channel 0 Condition Code)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	11Eh	H_CH0ConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Channel 0 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ンネル CH0 の転送完了結果を示します。

Bit7 Reserved

Bit6-4 ConditionCode[2:0]

チャンネル 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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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

7.6.61. 12Fh H_CH0Join (Host Channel 0 Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	12Fh	H_CH0Join		7:	0:	1:
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join CH0 to show Status
				5:	0:	1:
				4:	0	1:
			R / W	3: JoinDMA0	0: Do nothing	1: Join CH0 to DMA0
			R / W	2: JoinDMA1	0: Do nothing	1: Join CH0 to DMA1
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join CH0 to CPU_Rd
	0: JoinCPU_Wr	0: Do nothing	1: Join CH0 to CPU_Wr	00h		

ホスト動作時にチャンネル CH0 とデータ転送を行うポートを指定します。

Bit7 Reserved

Bit6 JoinFIFO_Stat

チャンネル CH0 の FIFO の Full、Empty 及び NotEmpty の状態を、H_FIFO_IntStat.FIFO_Full、H_FIFO_IntStat.FIFO_Empty 及び H_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved

Bit3 JoinDMA1

チャンネル CH0 の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

チャンネル CH0 の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

チャンネル CH0 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、このチャンネルの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

チャンネル CH0 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、このチャンネルの FIFO にデータが書き込まれます。

JoinDMAx{x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.6.62. 130h H_CHaConfig_0 (Host Channel a Configuration0)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	130h	H_CHaConfig_0	R / W	7: ACK_Cnt[3]	Channel[a] ACK Count	00h
				6: ACK_Cnt[2]		
				5: ACK_Cnt[1]		
				4: ACK_Cnt[0]		
			R / W	3: SpeedMode[1]	Channel[a] 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。

Bit0 TranGo

このビットを"1"に設定するとチャンネル 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、このビットはチャンネル 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。)

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。

7.6.63. 131h H_CHaConfig_1 (Host Channel a Configuration1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	131h	H_CHaConfig_1	R / W	7: TID[1]	Channel a 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ンネル CHa の基本設定を行います。

Bit7-6 TID[1:0]

チャンネル 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_BO_SupportControl 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。

7.6.64. 132h H_CHaMaxPktSize_H (Host Channel a Max Packet Size High)

7.6.65. 133h H_CHaMaxPktSize_L (Host Channel a Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	132h	H_CHaMaxPktSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R / W	1: MaxPktSize[9] 0: MaxPktSize[8]	Channel a Max Packet Size High		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	133h	H_CHaMaxPktSize_L	R / W	7: MaxPktSize[7]	Channel a Max Packet Size Low	00h
				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。

132h.Bit7-2 Reserved

リザーブ・ビットには"1"を書き込まないで下さい。

132h.Bit1-0, 133h.Bit7-0 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。

- 7.6.66. 134h H_CHaTotalSize_HH (Host Channel a Total Size High-High)
 7.6.67. 135h H_CHaTotalSize_HL (Host Channel a Total Size High-Low)
 7.6.68. 136h H_CHaTotalSize_LH (Host Channel a Total Size Low-High)
 7.6.69. 137h H_CHaTotalSize_LL (Host Channel a Total Size Low-Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	134h	H_CHaTotalSize_HH	R / W	7: TotalSize[31]	Channel a Total Size High-High	00h
				6: TotalSize[30]		
				5: TotalSize[29]		
				4: TotalSize[28]		
				3: TotalSize[27]		
				2: TotalSize[26]		
				1: TotalSize[25]		
				0: TotalSize[24]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	135h	H_CHaTotalSize_HL	R / W	7: TotalSize[23]	Channel a Total Size High-Low	00h
				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	136h	H_CHaTotalSize_LH	R / W	7: TotalSize[15]	Channel a Total Size Low-High	00h
				6: TotalSize[14]		
				5: TotalSize[13]		
				4: TotalSize[12]		
				3: TotalSize[11]		
				2: TotalSize[10]		
				1: TotalSize[9]		
				0: TotalSize[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	137h	H_CHaTotalSize_LL	R / W	7: TotalSize[7]	Channel a Total Size Low-Low	00h
				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。

134h.Bit7-0, 135h.Bit7-0, 136h.Bit7-0, 137h.Bit7-0 TotalSize[31:0]

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、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。

bit 31~24 (H_CHaTotalSize_HH レジスタ)をリードすると bit 23~16 (H_CHaTotalSize_HL レジスタ)の値、bit 15~8 (H_CHaTotalSize_LH レジスタ)の値、bit 7~0 (H_CHaTotalSize_LL レジスタ)の値が固定されます。(リード値が固定されても IC 内部ではカウントは継続されます)

よって、レジスタ 8bit アクセス時で残り転送数をリードする場合は、H_CHaTotalSize_HH、H_CHaTotalSize_HL、H_CHaTotalSize_LH、H_CHaTotalSize_LL レジスタの順番にアクセスしてください。

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。

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。

7.6.70. 138h H_CHaHubAdrs (Host Channel a Hub Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	138h	H_CHaHubAdrs	R / W	7: HubAdrs[3]	Channel a Hub Address	00h	
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:		1:
			R / W	2: Port[2]	Channel a 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。

Bit7-4 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。

7.6.71. 139h H_CHaFuncAdrs (Host Channel a Function Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	139h	H_CHaFuncAdrs	R / W	7: FuncAdrs[3]	Channel a Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R / W	3: EP_Number[3]	Channel a 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。

7.6.72. 13Ah H_BO_SupportControl (Host Bulk Only Transfer Support Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	13Ah	H_BO_SupportControl		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ンネル 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"に設定するとチャンネル 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_BO_SupportIntStat 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。

7.6.73. 13Bh H_CSW_RcvDataSize (Host CSW Receive Data Size)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	13Bh	H_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ンネル CHa のパ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。

7.6.74. 13Ch H_OUT_EP_Control (Host OUT Endpoint Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	13Ch	H_OUT_EP_Control		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ンネル 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。

7.6.75. 13Dh H_IN_EP_Control (Host IN Endpoint Control)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	13Dh	H_IN_EP_Control		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ンネル 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。

7.6.76. 13Eh H_CHaConditionCode (Host Channel a Condition Code)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	13Eh	H_CHaConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Channel a 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ンネル CHa の転送完了結果を示します。

Bit7 Reserved

Bit6-4 ConditionCode[2:0]

チャンネル 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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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

7.6.77. 13Fh H_CHaJoin (Host Channel a Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	13Fh	H_CHaJoin	R / W	7: JoinIDE	0: Do nothing	1: Join CHa to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join CHa to show Status	
				5:	0:	1:	
				4:	0	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join CHa to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join CHa to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join CHa to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join CHa to CPU_Wr	

ホスト動作時にチャンネル CHa とデータ転送を行うポートを指定します。

Bit7 JoinIDE

チャンネル CHa の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

チャンネル CHa の FIFO の Full、Empty 及び NotEmpty の状態を、H_FIFO_IntStat.FIFO_Full、H_FIFO_IntStat.FIFO_Empty 及び H_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

チャンネル CHa の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

チャンネル CHa の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

チャンネル CHa 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、このチャンネルの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

チャンネル CHa 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、このチャンネルの FIFO にデータが書き込まれます。

JoinDMAx{x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.6.78. 140h H_CHbConfig_0 (Host Channel b Configuration0)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	140h	CHbConfig_0	R / W	7: ACK_Cnt[3]	Channel b ACK Count		00h
				6: ACK_Cnt[2]			
				5: ACK_Cnt[1]			
				4: ACK_Cnt[0]			
			R / W	3: SpeedMode[1]	Channel b 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 CHb の基本設定を行います。

Bit7-4 ACK_Cnt[3:0]

チャンネル CHb 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_CHbIntStat 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]

チャンネル CHb 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 CHb 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 CHb 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_CHbTotalSize_HH~LL レジスタ で設定したバイト数完了した時点で H_CHbIntStat 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_CHbIntStat 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_CHbConditionCode 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ください。

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_CHbIntStat 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。)

7.6.79. 141h H_CHbConfig_1 (Host Channel b Configuration1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	141h	H_CHbConfig_1	R / W	7: TID[1]	Channel b Transaction ID	00h	
				6: TID[0]			
			R / W	5: TranType[1]	Channel b Transfer Type		
				4: TranType[0]			
			R / W	3: AutoZerolen	0: Do nothing		1: Add Zerolen
				2:	0:		1:
			R / W	1:	0:		1:
0: TotalSizeFree	0: Do nothing	1: Total Size Free					

ホスト動作時にチャンネル CHb の基本設定を行います。

Bit7-6 TID[1:0]

チャンネル CHb 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]

チャンネル CHb 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: Reserved — 本値の使用を禁止します。
- 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_CHbTotalSizeHH~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_CHbTotalSizeHH~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。

7.6.80. 142h H_CHbMaxPktSize_H (Host Channel b Max Packet Size High)

7.6.81. 143h H_CHbMaxPktSize_L (Host Channel b Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	142h	H_CHbMaxPktSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R / W	1: MaxPktSize[9] 0: MaxPktSize[8]	Channel b Max Packet Size High		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	143h	H_CHbMaxPktSize_L	R / W	7: MaxPktSize[7]	Channel b Max Packet Size Low	00h
				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 CHb の MaxPacketSize の設定を行います。

142h.Bit7-2 Reserved

リザーブ・ビットには"1"を書き込まないで下さい。

142h.Bit1-0, 143h.Bit7-0 MaxPktSize[9:0]

チャンネル CHb 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 512 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。

- 7.6.82. 144h H_CHbTotalSize_HH (Host Channel b Total Size High-High)
 7.6.83. 145h H_CHbTotalSize_HL (Host Channel b Total Size High-Low)
 7.6.84. 146h H_CHbTotalSize_LH (Host Channel b Total Size Low-High)
 7.6.85. 147h H_CHbTotalSize_LL (Host Channel b Total Size Low-Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	144h	H_CHbTotalSize_HH	R / W	7: TotalSize[31]	Channel b Total Size High-High	00h
				6: TotalSize[30]		
				5: TotalSize[29]		
				4: TotalSize[28]		
				3: TotalSize[27]		
				2: TotalSize[26]		
				1: TotalSize[25]		
				0: TotalSize[24]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	145h	H_CHbTotalSize_HL	R / W	7: TotalSize[23]	Channel b Total Size High-Low	00h
				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	146h	H_CHbTotalSize_LH	R / W	7: TotalSize[15]	Channel b Total Size Low-High	00h
				6: TotalSize[14]		
				5: TotalSize[13]		
				4: TotalSize[12]		
				3: TotalSize[11]		
				2: TotalSize[10]		
				1: TotalSize[9]		
				0: TotalSize[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	147h	H_CHbTotalSize_LL	R / W	7: TotalSize[7]	Channel b Total Size Low-Low	00h
				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 CHb で転送を行うデータの Total Size を設定します。

144h.Bit7-0, 145h.Bit7-0, 146h.Bit7-0, 147h.Bit7-0 TotalSize[31:0]

チャンネル CHb 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_CHbConfig_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。

bit 31~24 (H_CHbTotalSize_HH レジスタ)をリードすると bit 23~16 (H_CHbTotalSize_HL レジスタ)の値、bit 15~8 (H_CHbTotalSize_LH レジスタ)の値、bit 7~0 (H_CHbTotalSize_LL レジスタ)の値が固定されます。(リード値が固定されても IC 内部ではカウントは継続されます)

よって、レジスタ 8bit アクセス時で残り転送数をリードする場合は、H_CHbTotalSize_HH、H_CHbTotalSize_HL、H_CHbTotalSize_LH、H_CHbTotalSize_LL レジスタの順番にアクセスしてください。

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。

7.6.86. 148h H_CHbHubAdrs (Host Channel b Hub Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	148h	H_CHbHubAdrs	R / W	7: HubAdrs[3]	Channel b Hub Address	00h	
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:		1:
			R / W	2: Port[2]	Channel b Port Number		
				1: Port[1]			
0: Port[0]							

ホスト動作時にチャンネル CHb に接続するハブの設定を行います。

Bit7-4 HubAdrs[3:0]

チャンネル CHb 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 CHb 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。

7.6.87. 149h H_CHbFuncAdrs (Host Channel b Function Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	149h	H_CHbFuncAdrs	R / W	7: FuncAdrs[3]	Channel b Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R / W	3: EP_Number[3]	Channel b 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 CHb 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 CHb 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 CHb 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。

7.6.88. 14Ah H_CHbInterval_H (Host Channel b Interval High)

7.6.89. 14Bh H_CHbInterval_L (Host Channel b Interval Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	14Ah	H_CHbInterval_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R / W	2: Interval[10]	Channel b Interrupt Transfer Interval High		
	1: Interval[9]						
			0: Interval[8]				

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	14Bh	H_CHdInterval_L	R / W	7: Interval[7]	Channel b Interrupt Transfer Interval Low	00h
				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ンネル CHb でインターラプト転送を行う時のインターバル値を設定します。

14Ah.Bit7-3

Reserved

14Ah.Bit2-0, 14Bh.Bit7-0

Interval[10:0]

インターラプト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_CHbConfig1 レジスタの TranType ビットが"11"(インターラプト転送)の場合のみ有効です。また、このレジスタの"0d"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。

7.6.90. 14Ch~14Dh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	14Ch -14Dh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.91. 14Eh H_CHbConditionCode (Host Channel b Condition Code)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	14Eh	H_CHbConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Channel b 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ンネル CHb の転送完了結果を示します。

Bit7 **Reserved**

Bit6-4 **ConditionCode[2:0]**

チャンネル CHb 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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

7.6.92. 14Fh H_CHbJoin (Host Channel b Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	14Fh	H_CHbJoin	R / W	7: JoinIDE	0: Do nothing	1: Join CHb to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join CHb to show Status	
				5:	0:	1:	
				4:	0	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join CHb to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join CHb to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join CHb to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join CHb to CPU_Wr	

ホスト動作時にチャンネル CHb とデータ転送を行うポートを指定します。

Bit7 JoinIDE

チャンネル CHb の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

チャンネル CHb の FIFO の Full、Empty 及び NotEmpty の状態を、H_FIFO_IntStat.FIFO_Full、H_FIFO_IntStat.FIFO_Empty 及び H_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

チャンネル CHb の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

チャンネル CHb の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

チャンネル CHb 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、このチャンネルの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

チャンネル CHb 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、このチャンネルの FIFO にデータが書き込まれます。

JoinDMAx{x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.6.93. 150h H_CHcConfig_0 (Host Channel c Configuration0)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	150h	CHcConfig_0	R / W	7: ACK_Cnt[3]	Channel c ACK Count		00h
				6: ACK_Cnt[2]			
				5: ACK_Cnt[1]			
				4: ACK_Cnt[0]			
			R / W	3: SpeedMode[1]	Channel c 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 CHc の基本設定を行います。

Bit7-4 ACK_Cnt[3:0]

チャンネル CHc 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_CHcIntStat 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]

チャンネル CHc 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 CHc 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 CHc 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_CHcTotalSize_HH~LL レジスタ で設定したバイト数完了した時点で H_CHcIntStat 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_CHcIntStat 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_CHcConditionCode 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ください。

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_CHcIntStat 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。)

7.6.94. 151h H_CHcConfig_1 (Host Channel c Configuration1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	151h	H_CHcConfig_1	R / W	7: TID[1]	Channel c Transaction ID	00h	
				6: TID[0]			
			R / W	5: TranType[1]	Channel c Transfer Type		
				4: TranType[0]			
			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ンネル CHc の基本設定を行います。

Bit7-6 TID[1:0]

チャンネル CHb 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]

チャンネル CHb 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: Reserved — 本値の使用を禁止します。
- 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_CHcTotalSizeHH~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_CHbTotalSizeHH~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。

7.6.95. 152h H_CHcMaxPktSize_H (Host Channel c Max Packet Size High)

7.6.96. 153h H_CHcMaxPktSize_L (Host Channel c Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	152h	H_CHcMaxPktSize_H		7:	0:	1:
				6:	0:	1:
				5:	0:	1:
				4:	0:	1:
				3:	0:	1:
				2:	0:	1:
			R / W	1: MaxPktSize[9] 0: MaxPktSize[8]	Channel c Max Packet Size High	00h

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	153h	H_CHcMaxPktSize_L	R / W	7: MaxPktSize[7]	Channel c Max Packet Size Low	00h
				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 CHc の MaxPacketSize の設定を行います。

152h.Bit7-2 Reserved

リザーブ・ビットには"1"を書き込まないで下さい。

152h.Bit1-0, 153h.Bit7-0 MaxPktSize[9:0]

チャンネル CHc 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 512 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。

- 7.6.97. 154h H_CHcTotalSize_HH (Host Channel c Total Size High-High)
 7.6.98. 155h H_CHcTotalSize_HL (Host Channel c Total Size High-Low)
 7.6.99. 156h H_CHcTotalSize_LH (Host Channel c Total Size Low-High)
 7.6.100. 157h H_CHcTotalSize_LL (Host Channel c Total Size Low-Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	154h	H_CHcTotalSize_HH	R / W	7: TotalSize[31]	Channel c Total Size High-High	00h
				6: TotalSize[30]		
				5: TotalSize[29]		
				4: TotalSize[28]		
				3: TotalSize[27]		
				2: TotalSize[26]		
				1: TotalSize[25]		
				0: TotalSize[24]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	155h	H_CHcTotalSize_HL	R / W	7: TotalSize[23]	Channel c Total Size High-Low	00h
				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	156h	H_CHcTotalSize_LH	R / W	7: TotalSize[15]	Channel c Total Size Low-High	00h
				6: TotalSize[14]		
				5: TotalSize[13]		
				4: TotalSize[12]		
				3: TotalSize[11]		
				2: TotalSize[10]		
				1: TotalSize[9]		
				0: TotalSize[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	157h	H_CHcTotalSize_LL	R / W	7: TotalSize[7]	Channel c Total Size Low-Low	00h
				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 CHc で転送を行うデータの Total Size を設定します。

154h.Bit7-0, 155h.Bit7-0, 156h.Bit7-0, 157h.Bit7-0 TotalSize[31:0]

チャンネル CHc 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_CHcConfig_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。

bit 31~24 (H_CHcTotalSize_HH レジスタ)をリードすると bit 23~16 (H_CHcTotalSize_HL レジスタ)の値、bit 15~8 (H_CHcTotalSize_LH レジスタ)の値、bit 7~0 (H_CHcTotalSize_LL レジスタ)の値が固定されます。(リード値が固定されても IC 内部ではカウントは継続されます)

よって、レジスタ 8bit アクセス時で残り転送数をリードする場合は、H_CHcTotalSize_HH、H_CHcTotalSize_HL、H_CHcTotalSize_LH、H_CHcTotalSize_LL レジスタの順番にアクセスしてください。

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。

7.6.101. 158h H_CHcHubAdrs (Host Channel c Hub Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	158h	H_CHcHubAdrs	R / W	7: HubAdrs[3]	Channel c Hub Address	00h	
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:		1:
			R / W	2: Port[2]	Channel c Port Number		
				1: Port[1]			
0: Port[0]							

ホスト動作時にチャンネル CHc に接続するハブの設定を行います。

Bit7-4 HubAdrs[3:0]

チャンネル CHc 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 CHc 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。

7.6.102. 159h H_CHcFuncAdrs (Host Channel c Function Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	159h	H_CHcFuncAdrs	R / W	7: FuncAdrs[3]	Channel c Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R / W	3: EP_Number[3]	Channel c 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 CHc 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 CHc 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 CHc 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。

7.6.103. 15Ah H_CHcInterval_H (Host Channel c Interval High)

7.6.104. 15Bh H_CHcInterval_L (Host Channel c Interval Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	15Ah	H_CHcInterval_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R / W	2: Interval[10]	Channel c Interrupt Transfer Interval High		
	1: Interval[9]						
		0: Interval[8]					

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	15Bh	H_CHcInterval_L	R / W	7: Interval[7]	Channel c Interrupt Transfer Interval Low	00h
				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ンネル CHc でインタラプト転送を行う時のインターバル値を設定します。

15Ah.Bit7-3

Reserved

15Ah.Bit2-0, 15Bh.Bit7-0

Interval[10:0]

インタラプト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_CHcConfig1 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。また、このレジスタの"0d"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。

7.6.105. 15Ch~15Dh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	15Ch -15Dh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.106. 15Eh H_CHcConditionCode (Host Channel c Condition Code)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	15Eh	H_CHcConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Channel c 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ンネル CHc の転送完了結果を示します。

Bit7 **Reserved**

Bit6-4 **ConditionCode[2:0]**

チャンネル CHc 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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

7.6.107. 15Fh H_CHcJoin (Host Channel c Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	15Fh	H_CHcJoin	R / W	7: JoinIDE	0: Do nothing	1: Join CHc to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join CHc to show Status	
				5:	0:	1:	
				4:	0	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join CHc to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join CHc to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join CHc to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join CHc to CPU_Wr	

ホスト動作時にチャンネル CHc とデータ転送を行うポートを指定します。

Bit7 JoinIDE

チャンネル CHc の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

チャンネル CHc の FIFO の Full、Empty 及び NotEmpty の状態を、H_FIFO_IntStat.FIFO_Full、H_FIFO_IntStat.FIFO_Empty 及び H_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

チャンネル CHc の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

チャンネル CHc の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

チャンネル CHc 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、このチャンネルの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

チャンネル CHc 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、このチャンネルの FIFO にデータが書き込まれます。

JoinDMAx {x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.6.108. 160h H_CHdConfig_0 (Host Channel d Configuration0)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	160h	CHdConfig_0	R / W	7: ACK_Cnt[3]	Channel d ACK Count		00h
				6: ACK_Cnt[2]			
				5: ACK_Cnt[1]			
				4: ACK_Cnt[0]			
			R / W	3: SpeedMode[1]	Channel d 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 CHd の基本設定を行います。

Bit7-4 ACK_Cnt[3:0]

チャンネル CHd 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_CHdIntStat 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]

チャンネル CHd 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 CHd 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 CHd 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_CHdTotalSize_HH~LL レジスタ で設定したバイト数完了した時点で H_CHdIntStat 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_CHdIntStat 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_CHdConditionCode 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ください。

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_CHdIntStat 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。)

7.6.109. 161h H_CHdConfig_1 (Host Channel d Configuration1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	161h	H_CHdConfig_1	R / W	7: TID[1]	Channel d Transaction ID	00h	
				6: TID[0]			
			R / W	5: TranType[1]	Channel d Transfer Type		
				4: TranType[0]			
			R / W	3: AutoZerolen	0: Do nothing		1: Add Zerolen
				2:	0:		1:
			R / W	1:	0:		1:
0: TotalSizeFree	0: Do nothing	1: Total Size Free					

ホスト動作時にチャンネル CHd の基本設定を行います。

Bit7-6 TID[1:0]

チャンネル CHd 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]

チャンネル CHd 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: Reserved — 本値の使用を禁止します。
- 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 CHdTotalSizeHH~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_CHdTotalSizeHH~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。

7.6.110. 162h H_CHdMaxPktSize_H (Host Channel d Max Packet Size High)

7.6.111. 163h H_CHdMaxPktSize_L (Host Channel d Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	162h	H_CHdMaxPktSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R / W	1: MaxPktSize[9] 0: MaxPktSize[8]	Channel d Max Packet Size High		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	163h	H_CHdMaxPktSize_L	R / W	7: MaxPktSize[7]	Channel d Max Packet Size Low	00h
				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 CHd の MaxPacketSize の設定を行います。

162h.Bit7-2 Reserved

リザーブ・ビットには"1"を書き込まないで下さい。

162h.Bit1-0, 163h.Bit7-0 MaxPktSize[9:0]

チャンネル CHd 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 512 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。

- 7.6.112. 164h H_CHdTotalSize_HH (Host Channel d Total Size High-High)
 7.6.113. 165h H_CHdTotalSize_HL (Host Channel d Total Size High-Low)
 7.6.114. 166h H_CHdTotalSize_LH (Host Channel d Total Size Low-High)
 7.6.115. 167h H_CHdTotalSize_LL (Host Channel d Total Size Low-Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	164h	H_CHdTotalSize_HH	R / W	7: TotalSize[31]	Channel d Total Size High-High	00h
				6: TotalSize[30]		
				5: TotalSize[29]		
				4: TotalSize[28]		
				3: TotalSize[27]		
				2: TotalSize[26]		
				1: TotalSize[25]		
				0: TotalSize[24]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	165h	H_CHdTotalSize_HL	R / W	7: TotalSize[23]	Channel d Total Size High-Low	00h
				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	166h	H_CHdTotalSize_LH	R / W	7: TotalSize[15]	Channel d Total Size Low-High	00h
				6: TotalSize[14]		
				5: TotalSize[13]		
				4: TotalSize[12]		
				3: TotalSize[11]		
				2: TotalSize[10]		
				1: TotalSize[9]		
				0: TotalSize[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	167h	H_CHdTotalSize_LL	R / W	7: TotalSize[7]	Channel d Total Size Low-Low	00h
				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 CHd で転送を行うデータの Total Size を設定します。

166h.Bit7-0, 167h.Bit7-0, 168h.Bit7-0, 169h.Bit7-0 TotalSize[31:0]

チャンネル CHd 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_CHdConfig_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。

bit 31~24 (H_CHdTotalSize_HH レジスタ)をリードすると bit 23~16 (H_CHdTotalSize_HL レジスタ)の値、bit 15~8 (H_CHdTotalSize_LH レジスタ)の値、bit 7~0 (H_CHdTotalSize_LL レジスタ)の値が固定されます。(リード値が固定されても IC 内部ではカウントは継続されます)

よって、レジスタ 8bit アクセス時で残り転送数をリードする場合は、H_CHdTotalSize_HH、H_CHdTotalSize_HL、H_CHdTotalSize_LH、H_CHdTotalSize_LL レジスタの順番にアクセスしてください。

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。

7.6.116. 168h H_CHdHubAdrs (Host Channel d Hub Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	168h	H_CHdHubAdrs	R / W	7: HubAdrs[3]	Channel d Hub Address	00h	
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:		1:
			R / W	2: Port[2]	Channel d Port Number		
				1: Port[1]			
0: Port[0]							

ホスト動作時にチャンネル CHd に接続するハブの設定を行います。

Bit7-4 HubAdrs[3:0]

チャンネル CHd 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 CHd 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。

7.6.117. 169h H_CHdFuncAdrs (Host Channel d Function Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	169h	H_CHdFuncAdrs	R / W	7: FuncAdrs[3]	Channel d Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R / W	3: EP_Number[3]	Channel d 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 CHd 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 CHd 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 CHd 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。

7.6.118. 16Ah H_CHdInterval_H (Host Channel d Interval High)

7.6.119. 16Bh H_CHdInterval_L (Host Channel d Interval Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	16Ah	H_CHdInterval_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R / W	2: Interval[10]	Channel d Interrupt Transfer Interval High		
	1: Interval[9]						
			0: Interval[8]				

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	16Bh	H_CHdInterval_L	R / W	7: Interval[7]	Channel d Interrupt Transfer Interval Low	00h
				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ンネル CHd でインタラプト転送を行う時のインターバル値を設定します。

16Ah.Bit7-3

Reserved

16Ah.Bit2-0, 16Bh.Bit7-0

Interval[10:0]

インタラプト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_CHdConfig1 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。また、このレジスタの"0d"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。

7.6.120. 16Ch~16Dh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	16Ch -16Dh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.121. 16Eh H_CHdConditionCode (Host Channel d Condition Code)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	16Eh	H_CHdConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Channel d 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ンネル CHd の転送完了結果を示します。

Bit7 **Reserved**

Bit6-4 **ConditionCode[2:0]**

チャンネル CHd 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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

7.6.122. 16Fh H_CHdJoin (Host Channel d Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	16Fh	H_CHdJoin	R / W	7: JoinIDE	0: Do nothing	1: Join CHd to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join CHd to show Status	
				5:	0:	1:	
				4:	0	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join CHd to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join CHd to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join CHd to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join CHd to CPU_Wr	

ホスト動作時にチャンネル CHd とデータ転送を行うポートを指定します。

Bit7 JoinIDE

チャンネル CHd の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

チャンネル CHd の FIFO の Full、Empty 及び NotEmpty の状態を、H_FIFO_IntStat.FIFO_Full、H_FIFO_IntStat.FIFO_Empty 及び H_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

チャンネル CHd の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

チャンネル CHd の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

チャンネル CHd 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、このチャンネルの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

チャンネル CHd 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、このチャンネルの FIFO にデータが書き込まれます。

JoinDMAx{x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.6.123. 170h H_CHeConfig_0 (Host Channel e Configuration0)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	170h	CHeConfig_0	R / W	7: ACK_Cnt[3]	Channel e ACK Count		00h
				6: ACK_Cnt[2]			
				5: ACK_Cnt[1]			
				4: ACK_Cnt[0]			
			R / W	3: SpeedMode[1]	Channel e 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 CHe の基本設定を行います。

Bit7-4 ACK_Cnt[3:0]

チャンネル CHe 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_CHbIntStat 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]

チャンネル CHe 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 CHe 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 CHe 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_CHeTotalSize_HH~LL レジスタ で設定したバイト数完了した時点で H_CHeIntStat 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_CHeIntStat 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_CHeConditionCode 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ください。

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_CHeIntStat 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。)

7.6.124. 171h H_CHeConfig_1 (Host Channel e Configuration1)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	171h	H_CHeConfig_1	R / W	7: TID[1]	Channel e Transaction ID	00h	
				6: TID[0]			
			R / W	5: TranType[1]	Channel e Transfer Type		
				4: TranType[0]			
			R / W	3: AutoZerolen	0: Do nothing		1: Add Zerolen
				2:	0:		1:
			R / W	1:	0:		1:
0: TotalSizeFree	0: Do nothing	1: Total Size Free					

ホスト動作時にチャンネル CHe の基本設定を行います。

Bit7-6 TID[1:0]

チャンネル CHe 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]

チャンネル CHb 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: Reserved — 本値の使用を禁止します。
- 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 CHeTotalSizeHH~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_CHeTotalSizeHH~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。

7.6.125. 172h H_CHeMaxPktSize_H (Host Channel e Max Packet Size High)

7.6.126. 173h H_CHeMaxPktSize_L (Host Channel e Max Packet Size Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	172h	H_CHeMaxPktSize_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
			R / W	1: MaxPktSize[9] 0: MaxPktSize[8]	Channel e Max Packet Size High		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	173h	H_CHeMaxPktSize_L	R / W	7: MaxPktSize[7]	Channel e Max Packet Size Low	00h
				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 CHe の MaxPacketSize の設定を行います。

172h.Bit7-2 Reserved

リザーブ・ビットには"1"を書き込まないで下さい。

172h.Bit1-0, 173h.Bit7-0 MaxPktSize[9:0]

チャンネル CHe 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 512 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。

- 7.6.127. 174h H_CHeTotalSize_HH (Host Channel e Total Size High-High)
 7.6.128. 175h H_CHeTotalSize_HL (Host Channel e Total Size High-Low)
 7.6.129. 176h H_CHeTotalSize_LH (Host Channel e Total Size Low-High)
 7.6.130. 177h H_CHeTotalSize_LL (Host Channel e Total Size Low-Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	174h	H_CHeTotalSize_HH	R / W	7: TotalSize[31]	Channel e Total Size High-High	00h
				6: TotalSize[30]		
				5: TotalSize[29]		
				4: TotalSize[28]		
				3: TotalSize[27]		
				2: TotalSize[26]		
				1: TotalSize[25]		
				0: TotalSize[24]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	175h	H_CHeTotalSize_HL	R / W	7: TotalSize[23]	Channel e Total Size High-Low	00h
				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	176h	H_CHeTotalSize_LH	R / W	7: TotalSize[15]	Channel e Total Size Low-High	00h
				6: TotalSize[14]		
				5: TotalSize[13]		
				4: TotalSize[12]		
				3: TotalSize[11]		
				2: TotalSize[10]		
				1: TotalSize[9]		
				0: TotalSize[8]		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	177h	H_CHeTotalSize_LL	R / W	7: TotalSize[7]	Channel e Total Size Low-Low	00h
				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 CHe で転送を行うデータの Total Size を設定します。

174h.Bit7-0, 175h.Bit7-0, 176h.Bit7-0, 177h.Bit7-0 TotalSize[31:0]

チャンネル CHe 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_CHeConfig_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。

bit 31~24 (H_CHeTotalSize_HH レジスタ)をリードすると bit 23~16 (H_CHeTotalSize_HL レジスタ)の値、bit 15~8 (H_CHeTotalSize_LH レジスタ)の値、bit 7~0 (H_CHeTotalSize_LL レジスタ)の値が固定されます。(リード値が固定されても IC 内部ではカウントは継続されます)

よって、レジスタ 8bit アクセス時で残り転送数をリードする場合は、H_CHeTotalSize_HH、H_CHeTotalSize_HL、H_CHeTotalSize_LH、H_CHeTotalSize_LL レジスタの順番にアクセスしてください。

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。

7.6.131. 178h H_CHeHubAdrs (Host Channel e Hub Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	178h	H_CHeHubAdrs	R / W	7: HubAdrs[3]	Channel e Hub Address	00h	
				6: HubAdrs[2]			
				5: HubAdrs[1]			
				4: HubAdrs[0]			
				3:	0:		1:
			R / W	2: Port[2]	Channel e Port Number		
				1: Port[1]			
0: Port[0]							

ホスト動作時にチャンネル CHe に接続するハブの設定を行います。

Bit7-4 HubAdrs[3:0]

チャンネル CHe 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 CHe 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。

7.6.132. 179h H_CHeFuncAdrs (Host Channel e Function Address)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	179h	H_CHeFuncAdrs	R / W	7: FuncAdrs[3]	Channel e Function Address	00h
				6: FuncAdrs[2]		
				5: FuncAdrs[1]		
				4: FuncAdrs[0]		
			R / W	3: EP_Number[3]	Channel e 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 CHe 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 CHe 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 CHe 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。

7.6.133. 17Ah H_CHeInterval_H (Host Channel e Interval High)

7.6.134. 17Bh H_CHeInterval_L (Host Channel e Interval Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	17Ah	H_CHeInterval_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
			R / W	2: Interval[10]	Channel e Interrupt Transfer Interval High		
	1: Interval[9]						
		0: Interval[8]					

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	17Bh	H_CHeInterval_L	R / W	7: Interval[7]	Channel e Interrupt Transfer Interval Low	00h
				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ンネル CHe でインターラプト転送を行う時のインターバル値を設定します。

17Ah.Bit7-3

Reserved

17Ah.Bit2-0, 17Bh.Bit7-0

Interval[10:0]

インターラプト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_CHeConfig1 レジスタの TranType ビットが"11"(インターラプト転送)の場合のみ有効です。また、このレジスタの"0d"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。

7.6.135. 17Ch~17Dh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	17Ch -17Dh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

7.6.136. 17Eh H_CHeConditionCode (Host Channel e Condition Code)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	17Eh	H_CHeConditionCode		7:	0:	1:	00h
			R	6: ConditionCode[2]	Channel e 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ンネル CHe の転送完了結果を示します。

Bit7 **Reserved**

Bit6-4 **ConditionCode[2:0]**

チャンネル CHe 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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	<ul style="list-style-type: none"> ・ 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

7.6.137. 17Fh H_CHeJoin (Host Channel b Join)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	17Fh	H_CHeJoin	R / W	7: JoinIDE	0: Do nothing	1: Join CHe to IDE	00h
			R / W	6: JoinFIFO_Stat	0: Do nothing	1: Join CHe to show Status	
				5:	0:	1:	
				4:	0	1:	
			R / W	3: JoinDMA1	0: Do nothing	1: Join CHe to DMA1	
			R / W	2: JoinDMA0	0: Do nothing	1: Join CHe to DMA0	
			R / W	1: JoinCPU_Rd	0: Do nothing	1: Join CHe to CPU_Rd	
			R / W	0: JoinCPU_Wr	0: Do nothing	1: Join CHe to CPU_Wr	

ホスト動作時にチャンネル CHe とデータ転送を行うポートを指定します。

Bit7 JoinIDE

チャンネル CHe の FIFO で IDE のデータ転送を行います。転送の方向は、IDE_Control.Dir ビットの設定によります。

Bit6 JoinFIFO_Stat

チャンネル CHe の FIFO の Full、Empty 及び NotEmpty の状態を、H_FIFO_IntStat.FIFO_Full、H_FIFO_IntStat.FIFO_Empty 及び H_FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit5-4 Reserved**Bit3 JoinDMA1**

チャンネル CHe の FIFO で DMA1 の転送を行います。転送の方向は、DMA1_Control.Dir ビットの設定によります。

Bit2 JoinDMA0

チャンネル CHe の FIFO で DMA0 の転送を行います。転送の方向は、DMA0_Control.Dir ビットの設定によります。

Bit1 JoinCPU_Rd

チャンネル CHe 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、このチャンネルの FIFO からデータが読み出されます。

Bit0 JoinCPU_Wr

チャンネル CHe 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、このチャンネルの FIFO にデータが書き込まれます。

JoinDMAx{x=0,1} ビットを設定した場合は、DMA0_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、DMAx{x=0,1}_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。

JoinDMA1 ビット、JoinDMA0 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"にセットすることが可能です。複数のビットに対して、同時に"1"を書きこんだ場合、上位ビットが有効とされます。

7.6.138. 180h H_CH0StartAdrs_H (Host Channel 0 Start Address High)

7.6.139. 181h H_CH0StartAdrs_L (Host Channel 0 Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset	
Host	180h	H_CH0StartAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: StartAdrs[12]	Channel 0 Start Address High		
					3: StartAdrs[11]			
					2: StartAdrs[10]			
1: StartAdrs[9]								
0: StartAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	181h	H_CH0StartAdrs_L	R / W	7: StartAdrs[7]	Channel 0 Start Address Low		00h
				6: StartAdrs[6]			
				5: StartAdrs[5]			
				4: StartAdrs[4]			
				3: StartAdrs[3]			
				2: StartAdrs[2]			
				1:			
				0:			

ホスト動作時にチャンネル CH0 で使用する FIFO の領域設定を行います。

180h.Bit7-5

Reserved

180h.Bit4-0, 181h.Bit7-2

StartAdrs[12:2]

チャンネル CH0 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。

チャンネル CH0 に割り当てられる領域は、H_CH0StartAdrs にて設定されたアドレスから、H_CH0EndAdrs にて設定されたアドレスの 1 バイト前までとなります。

H_CH0StartAdrs、H_CH0EndAdrs を設定した後は、必ず H_CHnControl レジスタの CH0FIFO_Clr ビットを"1"にしてチャンネル CH0 の FIFO をクリアして下さい。

なお、ここで設定した領域より、チャンネル CH0 の MaxPktSize が大きい場合には、正常に動作しません。

全チャンネルで確保する FIFO 領域、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

181h.Bit1-0

Reserved

7.6.140. 182h H_CH0EndAdrs_H (Host Channel 0 End Address High)

7.6.141. 183h H_CH0EndAdrs_L (Host Channel 0 End Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	182h	H_CH0EndAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: EndAdrs[12]	Channel 0 EndAddress High		
					3: EndAdrs[11]			
					2: EndAdrs[10]			
1: EndAdrs[9]								
0: EndAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	183h	H_CH0EndAdrs_L	R / W	7: EndAdrs[7]	Channel 0 End Address Low	00h
				6: EndAdrs[6]		
				5: EndAdrs[5]		
				4: EndAdrs[4]		
				3: EndAdrs[3]		
				2: EndAdrs[2]		
				1:		
				0:		

ホスト動作時にチャンネル CH0 で使用する FIFO の領域設定を行います。

182h.Bit7-5 **Reserved**

182h.Bit4-0, 183h.Bit7-2 **EndAdrs[12:2]**

チャンネル CH0 が使用する FIFO の領域の終端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。

1A3h.Bit1-0 **Reserved**

7.6.142. 184h H_CHaStartAdrs_H (Host Channel a Start Address High)

7.6.143. 185h H_CHaStartAdrs_L (Host Channel a Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset	
Host	184h	H_CHaStartAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: StartAdrs[12]	Channel a Start Address High		
					3: StartAdrs[11]			
					2: StartAdrs[10]			
1: StartAdrs[9]								
0: StartAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	185h	H_CHaStartAdrs_L	R / W	7: StartAdrs[7]	Channel a Start Address Low		00h
				6: StartAdrs[6]			
				5: StartAdrs[5]			
				4: StartAdrs[4]			
				3: StartAdrs[3]			
				2: StartAdrs[2]			
				1:			
				0:			

ホスト動作時にチャンネル CHa で使用する FIFO の領域設定を行います。

184h.Bit7-5

Reserved

184h.Bit4-0, 185h.Bit7-2

StartAdrs[12:2]

チャンネル CHa 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。

チャンネル CHa に割り当てられる領域は、H_CHaStartAdrs にて設定されたアドレスから、H_CHaEndAdrs にて設定されたアドレスの 1 バイト前までとなります。

H_CHaStartAdrs、H_CHaEndAdrs を設定した後は、必ず H_CHnControl レジスタの CHaFIFO_Clr ビットを"1"にしてチャンネル CHa の FIFO をクリアして下さい。

なお、ここで設定した領域より、チャンネル CHa の MaxPktSize が大きい場合には、正常に動作しません。

全チャンネルで確保する FIFO 領域、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

185h.Bit1-0

Reserved

7.6.144. 186h H_CHaEndAdrs_H (Host Channel a End Address High)

7.6.145. 187h H_CHaEndAdrs_L (Host Channel a End Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	186h	H_CHaEndAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: EndAdrs[12]	Channel a EndAddress High		
					3: EndAdrs[11]			
					2: EndAdrs[10]			
1: EndAdrs[9]								
0: EndAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	187h	H_CHaEndAdrs_L	R / W	7: EndAdrs[7]	Channel a End Address Low		00h
				6: EndAdrs[6]			
				5: EndAdrs[5]			
				4: EndAdrs[4]			
				3: EndAdrs[3]			
				2: EndAdrs[2]			
				1:			
				0:			

ホスト動作時にチャンネル CHa で使用する FIFO の領域設定を行います。

186h.Bit7-5 **Reserved**

186h.Bit4-0, 187h.Bit7-2 **EndAdrs[12:2]**

チャンネル CHa が使用する FIFO の領域の終端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。

187h.Bit1-0 **Reserved**

7.6.146. 188h H_CHbStartAdrs_H (Host Channel b Start Address High)

7.6.147. 189h H_CHbStartAdrs_L (Host Channel b Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	188h	H_CHbStartAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: StartAdrs[12]	Channel b Start Address High		
					3: StartAdrs[11]			
					2: StartAdrs[10]			
1: StartAdrs[9]								
0: StartAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	189h	H_CHbStartAdrs_L	R / W	7: StartAdrs[7]	Channel b Start Address Low	00h
				6: StartAdrs[6]		
				5: StartAdrs[5]		
				4: StartAdrs[4]		
				3: StartAdrs[3]		
				2: StartAdrs[2]		
				1:		
				0:		

ホスト動作時にチャンネル CHb で使用する FIFO の領域設定を行います。

188h.Bit7-5

Reserved

188h.Bit4-0, 189h.Bit7-2

StartAdrs[12:2]

チャンネル CHb 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。

チャンネル CHb に割り当てられる領域は、H_CHbStartAdrs にて設定されたアドレスから、H_CHbEndAdrs にて設定されたアドレスの 1 バイト前までとなります。

H_CHbStartAdrs、H_CHbEndAdrs を設定した後は、必ず H_CHrFIFO_Clr レジスタの CHbFIFO_Clr ビットを"1"にしてチャンネル CHb の FIFO をクリアして下さい。

なお、ここで設定した領域より、チャンネル CHb の MaxPktSize が大きい場合には、正常に動作しません。

全チャンネルで確保する FIFO 領域、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

189h.Bit1-0

Reserved

7.6.148. 18Ah H_CHbEndAdrs_H (Host Channel b End Address High)

7.6.149. 18Bh H_CHbEndAdrs_L (Host Channel b End Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	18Ah	H_CHbEndAdrs_H		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
			R / W	4: EndAdrs[12]	Channel b EndAddress High		
				3: EndAdrs[11]			
				2: EndAdrs[10]			
				1: EndAdrs[9]			
0: EndAdrs[8]							

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	18Bh	H_CHbEndAdrs_L	R / W	7: EndAdrs[7]	Channel b End Address Low	00h
				6: EndAdrs[6]		
				5: EndAdrs[5]		
				4: EndAdrs[4]		
				3: EndAdrs[3]		
				2: EndAdrs[2]		
				1:		
				0:		

ホスト動作時にチャンネル CHb で使用する FIFO の領域設定を行います。

18Ah.Bit7-5 Reserved

18Ah.Bit4-0, 18Bh.Bit7-2 EndAdrs[12:2]

チャンネル CHb が使用する FIFO の領域の終端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。

18Bh.Bit1-0 Reserved

7.6.150. 18Ch H_CHcStartAdrs_H (Host Channel c Start Address High)

7.6.151. 18Dh H_CHcStartAdrs_L (Host Channel c Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset	
Host	18Ch	H_CHcStartAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: StartAdrs[12]	Channel c Start Address High		
					3: StartAdrs[11]			
					2: StartAdrs[10]			
1: StartAdrs[9]								
0: StartAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description		Reset
Host	18Dh	H_CHcStartAdrs_L	R / W	7: StartAdrs[7]	Channel c Start Address Low		00h
				6: StartAdrs[6]			
				5: StartAdrs[5]			
				4: StartAdrs[4]			
				3: StartAdrs[3]			
				2: StartAdrs[2]			
				1:			
				0:			

ホスト動作時にチャンネル CHc で使用する FIFO の領域設定を行います。

18Ch.Bit7-5 **Reserved**18Ch.Bit4-0, 18Dh.Bit7-2 **StartAdrs[12:2]**

チャンネル CHc 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 64 バイト単位での指定になります。

チャンネル CHc に割り当てられる領域は、H_CHcStartAdrs にて設定されたアドレスから、H_CHcEndAdrs にて設定されたアドレスの 1 バイト前までとなります。

H_CHcStartAdrs、H_CHbEndAdrs を設定した後は、必ず H_CHrFIFO_Clr レジスタの CHcFIFO_Clr ビットを"1"にしてチャンネル CHc の FIFO をクリアして下さい。

なお、ここで設定した領域より、チャンネル CHc の MaxPktSize が大きい場合には、正常に動作しません。

全チャンネルで確保する FIFO 領域、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

18Dh.Bit1-0 **Reserved**

7.6.152. 18Eh H_CHcEndAdrs_H (Host Channel c End Address High)

7.6.153. 18Fh H_CHcEndAdrs_L (Host Channel c End Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	18Eh	H_CHcEndAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: EndAdrs[12]	Channel c EndAddress High		
					3: EndAdrs[11]			
					2: EndAdrs[10]			
1: EndAdrs[9]								
0: EndAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	18Fh	H_CHcEndAdrs_L	R / W	7: EndAdrs[7]	Channel c End Address Low		00h
				6: EndAdrs[6]			
				5: EndAdrs[5]			
				4: EndAdrs[4]			
				3: EndAdrs[3]			
				2: EndAdrs[2]			
				1:			
				0:			

ホスト動作時にチャンネル CHc で使用する FIFO の領域設定を行います。

18Eh.Bit7-5 **Reserved**

18Eh.Bit4-0, 18Fh.Bit7-2 **EndAdrs[12:2]**

チャンネル CHc が使用する FIFO の領域の終端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。

18Fh.Bit1-0 **Reserved**

7.6.154. 190h H_CHdStartAdrs_H (Host Channel d Start Address High)

7.6.155. 191h H_CHdStartAdrs_L (Host Channel d Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	190h	H_CHdStartAdrs_H		7:	0:	1:	
				6:	0:	1:	
				5:	0:	1:	
				R / W	4: StartAdrs[12]	Channel d Start Address High	
					3: StartAdrs[11]		
					2: StartAdrs[10]		
1: StartAdrs[9]							
0: StartAdrs[8]							
					00h		

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	191h	H_CHdStartAdrs_L	R / W	7: StartAdrs[7]	Channel d Start Address Low	
				6: StartAdrs[6]		
				5: StartAdrs[5]		
				4: StartAdrs[4]		
				3: StartAdrs[3]		
				2: StartAdrs[2]		
				1:		
				0:		
					00h	

ホスト動作時にチャンネル CHd で使用する FIFO の領域設定を行います。

190h.Bit7-5

Reserved

190h.Bit4-0, 191h.Bit7-2

StartAdrs[12:2]

チャンネル CHd 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。

チャンネル CHd に割り当てられる領域は、H_CHdStartAdrs にて設定されたアドレスから、H_CHdEndAdrs にて設定されたアドレスの 1 バイト前までとなります。

H_CHdStartAdrs、H_CHbEndAdrs を設定した後は、必ず H_CHrFIFO_Clr レジスタの CHdFIFO_Clr ビットを"1"にしてチャンネル CHd の FIFO をクリアして下さい。

なお、ここで設定した領域より、チャンネル CHd の MaxPktSize が大きい場合には、正常に動作しません。

全チャンネルで確保する FIFO 領域、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

191h.Bit1-0

Reserved

7.6.156. 192h H_CHdEndAdrs_H (Host Channel d End Address High)

7.6.157. 193h H_CHdEndAdrs_L (Host Channel d End Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	192h	H_CHdEndAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: EndAdrs[12]	Channel d EndAddress High		
					3: EndAdrs[11]			
					2: EndAdrs[10]			
1: EndAdrs[9]								
0: EndAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	193h	H_CHdEndAdrs_L	R / W	7: EndAdrs[7]	Channel d End Address Low		00h
				6: EndAdrs[6]			
				5: EndAdrs[5]			
				4: EndAdrs[4]			
				3: EndAdrs[3]			
				2: EndAdrs[2]			
				1:			
				0:			

ホスト動作時にチャンネル CHd で使用する FIFO の領域設定を行います。

192h.Bit7-2 Reserved

192h.Bit4-0, 193h.Bit7-2 EndAdrs[12:2]

チャンネル CHd が使用する FIFO の領域の終端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。

193h.Bit1-0 Reserved

7.6.158. 194h H_CHeStartAdrs_H (Host Channel e Start Address High)

7.6.159. 195h H_CHeStartAdrs_L (Host Channel e Start Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	194h	H_CHeStartAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
				R / W	4: StartAdrs[12]	Channel e Start Address High		
					3: StartAdrs[11]			
					2: StartAdrs[10]			
1: StartAdrs[9]								
0: StartAdrs[8]								

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset
Host	195h	H_CHeStartAdrs_L	R / W	7: StartAdrs[7]	Channel e Start Address Low	00h
				6: StartAdrs[6]		
				5: StartAdrs[5]		
				4: StartAdrs[4]		
				3: StartAdrs[3]		
				2: StartAdrs[2]		
				1:		
				0:		

ホスト動作時にチャンネル CHe で使用する FIFO の領域設定を行います。

194h.Bit7-5

Reserved

194h.Bit4-0, 195h.Bit7-2

StartAdrs[12:2]

チャンネル CHe 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。

チャンネル CHe に割り当てられる領域は、H_CHeStartAdrs にて設定されたアドレスから、H_CHeEndAdrs にて設定されたアドレスの 1 バイト前までとなります。

H_CHeStartAdrs、H_CHeEndAdrs を設定した後は、必ず H_CHrFIFO_Clr レジスタの CHeFIFO_Clr ビットを"1"にしてチャンネル CHe の FIFO をクリアして下さい。

なお、ここで設定した領域より、チャンネル CHe の MaxPktSize が大きい場合には、正常に動作しません。

全チャンネルで確保する FIFO 領域、CBW エリア、CSW エリアの合計が内蔵の RAM の合計を超えないように設定して下さい。

195h.Bit1-0

Reserved

7.6.160. 196h H_CHeEndAdrs_H (Host Channel e End Address High)

7.6.161. 197h H_CHeEndAdrs_L (Host Channel e End Address Low)

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset		
Host	196h	H_CHeEndAdrs_H		7:	0:	1:	00h	
				6:	0:	1:		
				5:	0:	1:		
			R / W	4: EndAdrs[12]	Channel e EndAddress High			
				3: EndAdrs[11]				
				2: EndAdrs[10]				
1: EndAdrs[9]								
	0: EndAdrs[8]							

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	197h	H_CHeEndAdrs_L	R / W	7: EndAdrs [7]	Channel e Start Address Low		00h
				6: EndAdrs [6]			
				5: EndAdrs [5]			
				4: EndAdrs [4]			
				3: EndAdrs [3]			
				2: EndAdrs [2]			
				1:			
				0:			

ホスト動作時にチャンネル CHe で使用する FIFO の領域設定を行います。

196h.Bit7-5 **Reserved**

196h.Bit4-0, 197h.Bit7-2 **EndAdrs[12:2]**

チャンネル CHe が使用する FIFO の領域の終端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。

197h.Bit1-0 **Reserved**

7.6.162. 198h~1FFh Reserved ()

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	198h -1FFh	Reserved		7:	0:	1:	XXh
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
				3:	0:	1:	
				2:	0:	1:	
				1:	0:	1:	
				0:	0:	1:	

0x1F5, 0x1F6 にレジスタが追加されています。詳細は、「Appendix D」を参照して下さい。

8. 電気的特性

8.1. 絶対最大定格

(V_{SS}=0V)

項目	記号	定格値	単位
電源電圧	HVDD	V _{SS} -0.3 ~ 4.0	V
	CVDD	V _{SS} -0.3 ~ 4.0	V
	LVDD	V _{SS} -0.3 ~ 2.5	V
入力電圧	HVI	V _{SS} -0.3 ~ HVDD+0.5	V
	CVI ※1	V _{SS} -0.3 ~ CVDD+0.5	V
	IVI ※2	V _{SS} -0.3 ~ 5.5	V
	VVI ※3	V _{SS} -0.3 ~ 6.0	V
	LVI ※4	V _{SS} -0.3 ~ LVDD+0.5	V
出力電圧	HVO	V _{SS} -0.3 ~ HVDD+0.5	V
	CVO ※1	V _{SS} -0.3 ~ CVDD+0.5	V
出力電流/端子	IOUT	±10	mA
保存温度	T _{stg}	-65 ~ 150	°C

※1 CPU I/F
 ※2 IDE I/F
 ※3 VBUS_B
 ※4 XI

8.2. 推奨動作条件

項目	記号	MIN	TYP	MAX	単位
電源電圧	HVDD	3.00	3.30	3.60	V
	CVDD	1.65	-	3.60	V
	LVDD	1.65	1.80	1.95	V
入力電圧	HVI	-0.3	-	HVDD+0.3	V
	CVI ※1	-0.3	-	CVDD+0.3	V
	IVI ※2	-0.3	-	5.5	V
	VVI ※3	-0.3	-	6.0	V
	LVI ※4	-0.3	-	LVDD+0.3	V
周囲温度	T _a	-40	25	110	°C

※1 CPU I/F
 ※2 IDE I/F
 ※3 VBUS_B
 ※4 XI

本ICは下記順序で電源投入を行ってください。

LVDD (内部) →HVDD, CVDD (IO 部)

また、本ICは下記順序で電源オフを行ってください。

HVDD, CVDD (IO 部) →LVDD (内部)

注)

LVDD が切断されている状態で HVDD, CVDD のみを継続的(1Sec 以上)に印可することは Chip の信頼性上問題がありますので避けてください。

8.3. DC 特性

DC 状態における入力特性(推奨動作条件による)

項目	記号	条件	MIN	TYP	MAX	単位
電源供給電流 ※1						
電源電流	IDDH	HVDD = 3.3V(typ), 3.6V(max)	-	41	65	mA
	IDCH	CVDD = 3.3V(typ), 3.6V(max)	-	1	4	mA
	IDCL	CVDD = 1.8V(typ), 1.95V(max)	-	0.7	2	mA
	IDDL	LVDD = 1.8V(typ), 1.95V(max)	-	75	120	mA
静止電流 ※2						
電源電流	IDDS	VIN = HVDD, CVDD, LVDD or VSS HVDD = 3.6V CVDD = 3.6V LVDD = 1.95V	-	-	80	μA
入カリーク						
入カリーク電流	IL	HVDD=3.6V CVDD=3.6V LVDD=1.95V HVIH=HVDD CVIH=CVDD LVIH=LVDD VIL=VSS	-5	-	5	μA
入カリーク						
入カリーク電流 (5Vトレラント)	ILIF	HVDD = 3.0V CVDD = 3.0V LVDD = 1.65V HVIH=5.5V	-10	-	10	μA

※1: typ は USB ホストとして USB-HDD を接続し、IDE-HDD と USB-HDD 間でデータを送受信している状態(実転送レート 30MB/s)での測定値。max は同値からの見積値。

※2: Ta=25°C、双方向端子が入力状態である場合の静止電流値。

弊社動作環境下における、各パワーマネージメントでの消費電力実測値 (Ta = 25°C)

項目	条件	MIN	TYP	MAX	単位
SLEEP (CPU バス動作時 ※1) ※2					
電源電力	HVDD = 3.3V CVDD = 3.3V LVDD = 1.8V	-	0.23	-	mW
SNOOZE (CPU バス動作時 ※1) ※2					
電源電力	HVDD = 3.3V CVDD = 3.3V LVDD = 1.8V	-	1.8	-	mW
ACTIVE60 (IDE⇔CPU) ※3					
電源電力	HVDD = 3.3V CVDD = 3.3V LVDD = 1.8V	-	41	-	mW
ACT_DEVICE (IDE⇔USB) ※4					
電源電力	HVDD = 3.3V CVDD = 3.3V LVDD = 1.8V		131	-	mW
ACT_HOST (IDE⇔USB) Copy ※5					
電源電力	HVDD = 3.3V CVDD = 3.3V LVDD = 1.8V		134	-	mW
ACT_HOST (IDE⇔USB) Direct Copy ※6					
電源電力	HVDD = 3.3V CVDD = 3.3V LVDD = 1.8V		273	-	mW

※1: CPU が CPU バス上に接続されているメモリ (SRAM や ROM 等) にアクセスしている状態。

※2: S1R72V05 が内蔵している USB デバイス時の DP プルアップ抵抗による消費電流値 (約 200 μ A) を除く。

※3: IDE-HDD と CPU 間でデータを送受信している状態 (実転送レート 4MB/s)。

※4: USB デバイスとして PC に接続し、IDE-HDD と USB 間でデータを送受信している状態 (実転送レート 25MB/s)。

※5: USB ホストとして USB-HDD を接続し、IDE-HDD と USB-HDD 間でデータを送受信している状態 (実転送レート 5.3MB/s)。

※6: USB ホストとして USB-HDD を接続し、IDE-HDD と USB-HDD 間でデータを送受信している状態 (実転送レート 30MB/s)。

8. 電気的特性

DC 状態における入力特性(推奨動作条件による)(つづき)

項目	記号	条件	MIN	TYP	MAX	単位
入力特性(LVCMOS)	端子名:	TEST, TDI, TCK, TRST, TMS				
"H"レベル入力電圧	VIH1	HVDD = 3.6V	2.2	-	-	V
"L"レベル入力電圧	VIL1	HVDD = 3.0V	-	-	0.8	V
入力特性(LVCMOS)	端子名:	CA[8:1], CD[15:0], XCS, XRD, XWRL, XWRH, XBEL, XDACK0, XDACK1, XRESET				
"H"レベル入力電圧	VIH2	CVDD = 3.6V	2.2	-	-	V
"L"レベル入力電圧	VIL2	CVDD = 3.0V	-	-	0.8	V
"H"レベル入力電圧	VIH3	CVDD = 1.95V	1.27	-	-	V
"L"レベル入力電圧	VIL3	CVDD = 1.65V	-	-	0.57	V
入力特性(LVCMOS)	端子名:	HDD[15:0], HDMARQ, HIORDY, HINTRQ, XHDASP, XHPDIAG				
"H"レベル入力電圧	VIH4	HVDD = 3.6V	2.2	-	-	V
"L"レベル入力電圧	VIL4	HVDD = 3.0V	-	-	0.8	V
シュミット入力特性	端子名:	VBUSFLG_A				
"H"レベルトリガ電圧	VT+	HVDD = 3.6V	1.4	-	2.7	V
"L"レベルトリガ電圧	VT-	HVDD = 3.0V	0.6	-	1.8	V
ヒステリシス電圧	ΔV	HVDD = 3.0V	0.3	-	-	V
シュミット入力特性(USB:FS)	端子名:	DP_A, DM_A, DP_B, DM_B				
"H"レベルトリガ電圧	VT+(USB)	HVDD = 3.6V	1.1	-	1.8	V
"L"レベルトリガ電圧	VT-(USB)	HVDD = 3.0V	1.0	-	1.5	V
ヒステリシス電圧	ΔV (USB)	HVDD = 3.0V	0.1	-	-	V
入力特性(USB:FS差動入力)	端子名:	DP_A, DM_Aのペア、DP_B, DM_Bのペア				
差動入力の感度	VDS(USB)	HVDD = 3.0V 差動入力電圧 0.8V ~ 2.5V	-	-	0.2	V
入力特性(VBUS)	端子名:	VBUS_B				
"H"レベルトリガ電圧	VT+(VBUS)	HVDD = 3.6V	1.86	-	2.85	V
"L"レベルトリガ電圧	VT-(VBUS)	HVDD = 3.0V	1.48	-	2.23	V
ヒステリシス電圧	ΔV (VBUS)	HVDD = 3.0V	0.31	-	0.64	V
プルダウン抵抗	RPLDV	VI=5.0V	110	125	150	k Ω
入力特性	端子名:	HDD[15:8], HDD[6:0], HIORDY, XHDASP, XHPDIAG, VBUSFLG_A				
プルアップ抵抗	RPLU	VI = VSS	50	100	240	k Ω
入力特性	端子名:	HDD7, HDMARQ, HINTRQ				
プルダウン抵抗	RPLD	VI = HVDD	50	100	240	k Ω

8. 電気的特性

DC 状態における出力特性(推奨動作条件による)

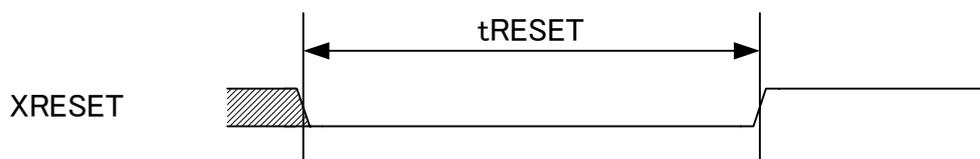
項目	記号	条件	MIN	TYP	MAX	単位
出力特性 端子名: CD[15:0], XDREQ0, XDREQ1, XINT						
"H"レベル出力電圧	VOH1	CVDD = 3.0V IOH = -2mA	CVDD-0.4	-	-	V
"L"レベル出力電圧	VOL1	CVDD = 3.0V IOL = 2mA	-	-	VSS+0.4	V
"H"レベル出力電圧	VOH2	CVDD = 1.65V IOH = -1mA	CVDD-0.4	-	-	V
"L"レベル出力電圧	VOL2	CVDD = 1.65V IOL = 1mA	-	-	VSS+0.4	V
出力特性 端子名: HDD[15:0], HDA[2:0], XHCS1, XHCS0, XHIOR, XHIOW, XHDMACK, XHRESET						
"H"レベル出力電圧	VOH3	HVDD = 3.0V IOH = -4mA	HVDD-1.0	-	-	V
"L"レベル出力電圧	VOL3	HVDD = 3.0V IOL = 4mA	-	-	VSS+0.4	V
出力特性 端子名: TDO, VBUSEN_A						
"H"レベル出力電圧	VOH4	HVDD = 3.0V IOH = -2mA	HVDD-0.4	-	-	V
"L"レベル出力電圧	VOL4	HVDD = 3.0V IOL = 2mA	-	-	VSS+0.4	V
出力特性(USB:FS) 端子名: DP_A, DM_A, DP_B, DM_B						
"H"レベル出力電圧	VOH(USB)	HVDD = 3.0V	2.8	-	-	V
"L"レベル出力電圧	VOL(USB)	HVDD = 3.6V	-	-	0.3	V
出力特性(USB:HS) 端子名: DP_A, DM_A, DP_B, DM_B						
"H"レベル出力電圧	VHSOH (USB)	HVDD = 3.0V	360	-	-	mV
"L"レベル出力電圧	VHSOL (USB)	HVDD = 3.6V	-	-	10.0	mV
出力特性 端子名: CD[15:0], XINT						
OFF-STATEリーク電流	IOZ	CVDD = 3.6 LVDD = 1.95V CVOH = CVDD LVOH = LVDD VOL = VSS	-5	-	5	μ A
出力特性 端子名: HDD[15:0], HDA[2:0], XHCS1, XHCS0, XHIOR, XHIOW, XHDMACK, XHRESET						
OFF-STATEリーク電流 (5Vトレラント)	IOZHF	HVDD = 3.0 HVOH = 5.5V	-10	-	10	μ A

端子容量

項目	記号	条件	MIN	TYP	MAX	単位
端子容量						
端子名: 全入力端子						
入力端子容量	CI	f = 10MHz HVDD=CVDD=LVD D=VSS	-	-	10	pF
端子容量						
端子名: 全出力端子						
出力端子容量	CO	f = 10MHz HVDD=CVDD=LVD D=VSS	-	-	10	pF
端子容量						
端子名: 全入出力端子 (DP_A, DM_A, DP_B, DM_Bを除く)						
入出力端子容量1	CIO1	f = 10MHz HVDD=CVDD=LVD D=VSS	-	-	10	pF
端子容量						
端子名: DP_A, DM_A, DP_B, DM_B						
入出力端子容量2	CIO2	f = 10MHz HVDD=CVDD=LVD D=VSS	-	-	11	pF

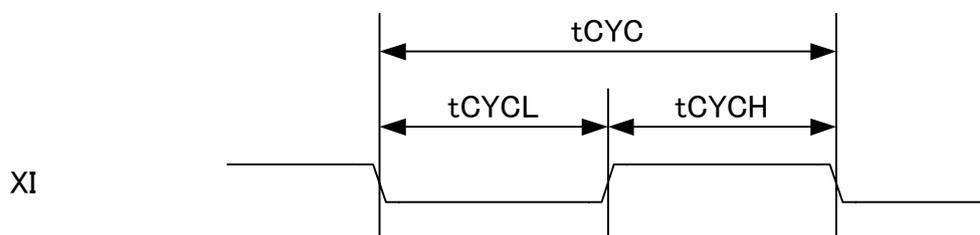
8.4. AC 特性

8.4.1. RESET タイミング

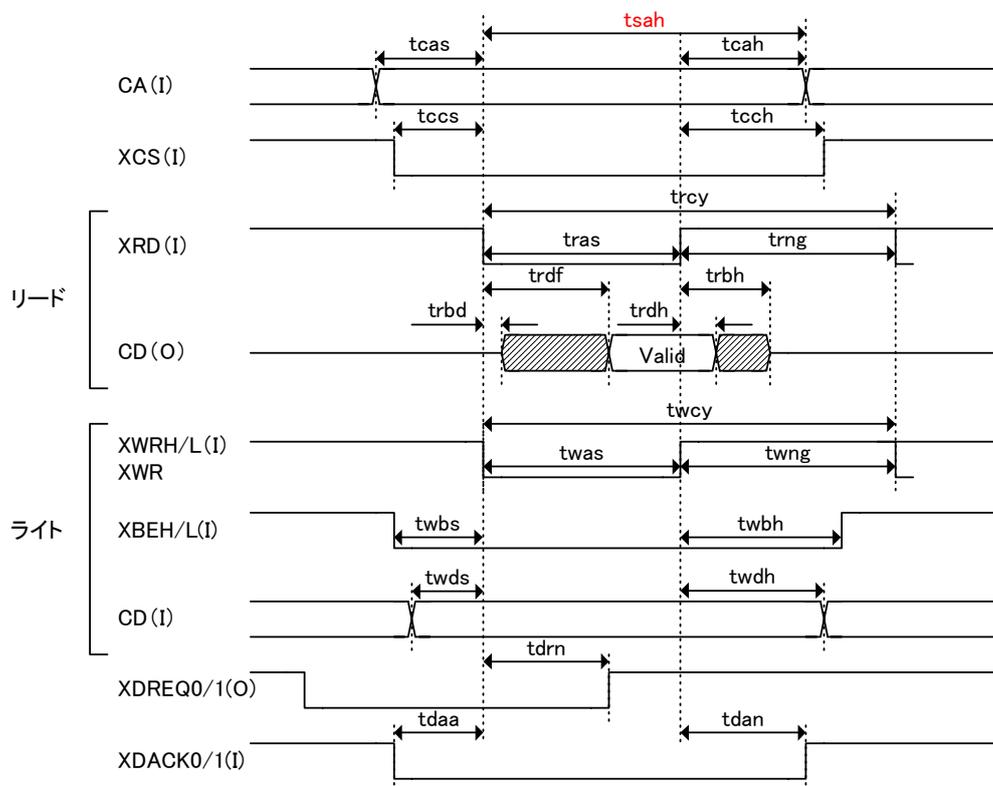


記号	説明	min	typ	max	単位
t_{RESET}	リセットパルス幅	40	-	-	ns

8.4.2. クロックタイミング

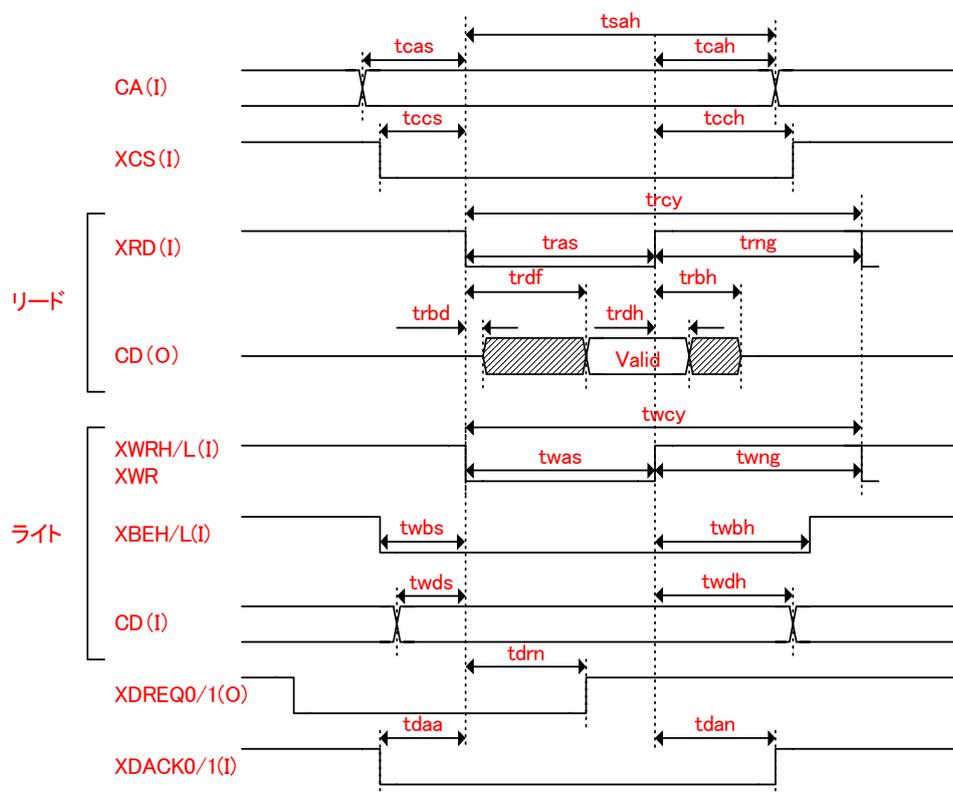


記号	説明	min	typ	max	単位
$t_{CYC}(\ast)$	クロックサイクル(ClkSelect=0)	11.999	12	12.001	MHz
$t_{CYC}(\ast)$	クロックサイクル(ClkSelect=1)	23.998	24	24.002	MHz
t_{CYCH} t_{CYCL}	クロックデューティ	45	-	55	%

8.4.3.
8.4.3.1.CPU/DMA I/F アクセスタイミング
CVDD(typ) = 1.65V~3.6V の場合(C_L=30pF)

記号	項目	min	typ	max	unit
tcas	アドレスセットアップ時間	6	-	-	ns
tcah	アドレスホールド時間(ストローブネゲーションから)	6	-	-	ns
tsah	アドレスホールド時間(ストローブアサーションから)	55	-	-	ns
tccs	XCSセットアップ時間	6	-	-	ns
tcch	XCSホールド時間	6	-	-	ns
trcy	リードサイクル	80	-	-	ns
tras	リードストローブアサート時間	40	-	-	ns
trng	リードストローブネゲート時間	25	-	-	ns
trbd	リードデータ出力開始時間	1	-	-	ns
trdf	リードデータ確定時間	-	-	35	ns
trdh	リードデータホールド時間	3	-	-	ns
trbh	リードデータ出力遅延時間	-	-	10	ns
twcy	ライトサイクル	80	-	-	ns
twas	ライトストローブアサート時間	40	-	-	ns
twng	ライトストローブネゲート時間	25	-	-	ns
twbs	ライトバイトイネーブルセットアップ時間	6	-	-	ns
twbh	ライトバイトイネーブルホールド時間	6	-	-	ns
twds	ライトデータセットアップ時間	0	-	-	ns
twdh	ライトデータホールド時間	0	-	-	ns
tdrn	XDREQ0/1ネゲート遅延時間	-	-	35	ns
tdaa	XDACK0/1セットアップ時間	6	-	-	ns
tdan	XDACK0/1ホールド時間	6	-	-	ns

8.4.3.2. CVDD(typ) = 3.0~3.6V に制限した場合(一部規格を緩和)

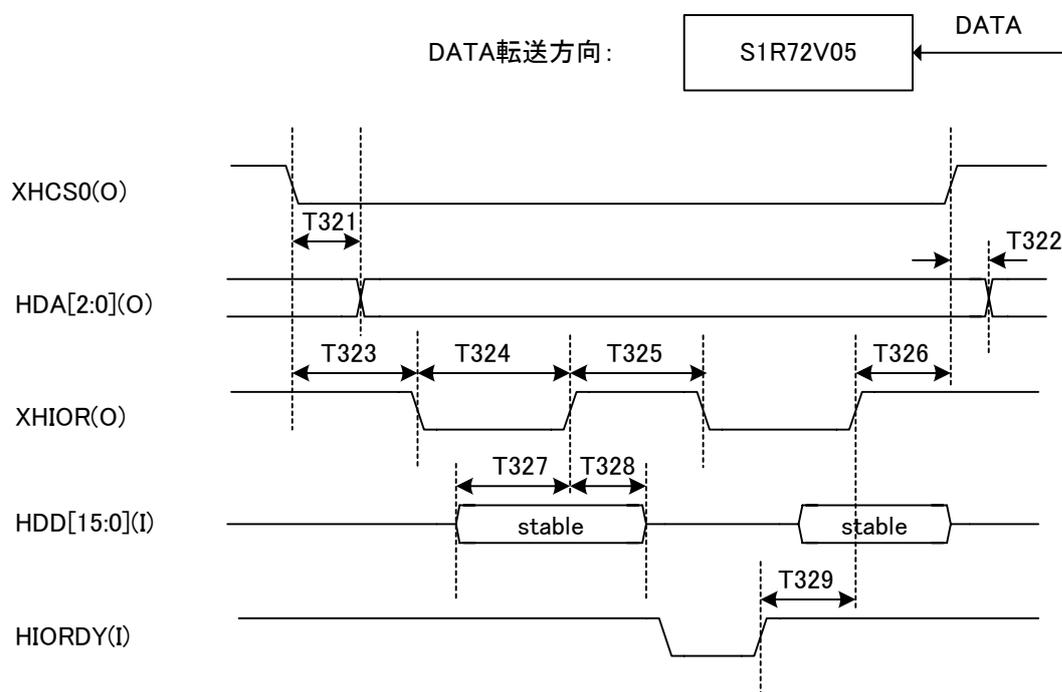


(CL=30pF)

記号	項目	min	typ	max	unit
tcas	アドレスセットアップ時間	6	-	-	ns
tcah	アドレスホールド時間(ストロブネゲーションから)	6	-	-	ns
tsah	アドレスホールド時間(ストロブアサーションから)	55	-	-	ns
tccs	XCSセットアップ時間	6	-	-	ns
tcch	XCSホールド時間	6	-	-	ns
trcy	リードサイクル	75	-	-	ns
tras	リードストロブアサート時間	37	-	-	ns
trng	リードストロブネゲート時間	25	-	-	ns
trbd	リードデータ出力開始時間	1	-	-	ns
trdf	リードデータ確定時間	-	-	30	ns
trdh	リードデータホールド時間	3	-	-	ns
trbh	リードデータ出力遅延時間	-	-	10	ns
twcy	ライトサイクル	75	-	-	ns
twas	ライトストロブアサート時間	37	-	-	ns
twng	ライトストロブネゲート時間	25	-	-	ns
twbs	ライトバイトイネーブルセットアップ時間	6	-	-	ns
twbh	ライトバイトイネーブルホールド時間	6	-	-	ns
twds	ライトデータセットアップ時間	0	-	-	ns
twdh	ライトデータホールド時間	0	-	-	ns
tdrn	XDREQ0/1ネゲート遅延時間	-	-	30	ns
tdaa	XDACK0/1セットアップ時間	6	-	-	ns
tdan	XDACK0/1ホールド時間	6	-	-	ns

8.4.4. IDE I/F タイミング

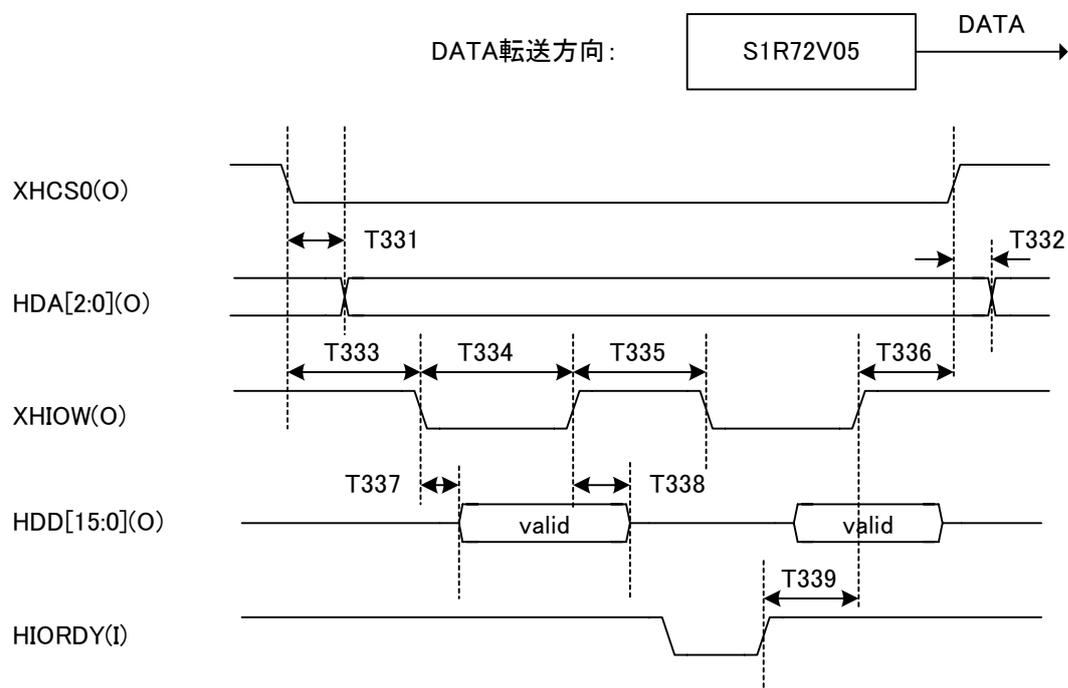
8.4.4.1. PIO Read Timing



記号	説明	min	typ	max	単位
T321	XHCS0 ↓ → HDA HDA出力遅延時間	-	0	-	ns
T322	XHCS0 ↑ → HDA HDAホールド時間	-	0	-	ns
T323	XHCS0 ↓ → XHIOR ↓ XHCS0セットアップ時間	80	-	-	ns
T324	XHIOR ↓ → XHIOR ↑ XHIORアサートパルス幅	-	$(AP+4) * 16.7 - 3$	-	ns
T325	XHIOR ↑ → XHIOR ↓ XHIORネゲートパルス幅	-	$(NP+4) * 16.7 + 3$	-	ns
T326	XHIOR ↑ → XHCS0 ↑ XHCS0ホールド時間	50	-	-	ns
T327	HDD → XHIOR ↑ データセットアップ時間	10	-	-	ns
T328	XHIOR ↑ → HDD データホールド時間	0	-	-	ns
T329	HIORDYアサート → XHIOR ↑ XHIOR出力遅延時間	-	-	25	ns

*1:AP=IDE_Tmod.AssertPulseWidth, NP=IDE_Tmod.NegatePulseWidth
詳細はレジスタ説明、「IDE Transfer Mode」を参照のこと

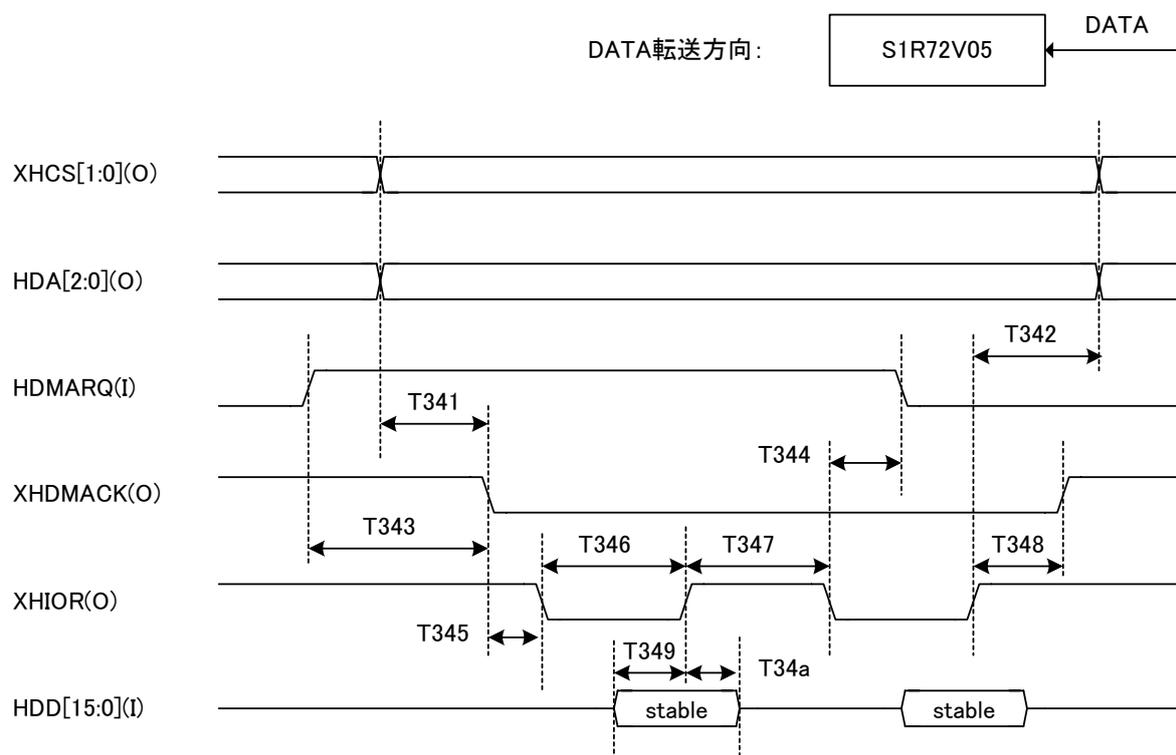
8.4.4.2. PIO Write Timing



記号	説明	min	typ	max	単位
T331	XHCS0 ↓ → HDA HDA出力遅延時間	-	0	-	ns
T332	XHCS0 ↑ → HDA HDAホールド時間	-	0	-	ns
T333	XHCS0 ↓ → XHIOW ↓ XHCS0セットアップ時間	80	-	-	ns
T334	XHIOW ↓ → XHIOW ↑ XHIOWアサートパルス幅	-	(AP+4) * 16.7 - 3	-	ns
T335	XHIOW ↑ → XHIOW ↓ XHIOWネゲートパルス幅	-	(NP+4) * 16.7 + 3	-	ns
T336	XHIOW ↑ → XHCS0 ↑ XHCS0ホールド時間	50	-	-	ns
T337	XHIOW ↓ → HDD データ出力遅延時間	0	-	10	ns
T338	XHIOW ↑ → HDD データバスネゲート時間	33	-	45	ns
T339	HIORDYアサート → XHIOW ↑ XHIOW出力遅延時間	-	-	25	ns

*1: AP=IDE_Tmod.AssertPulseWidth, NP=IDE_Tmod.NegatePulseWidth
詳細はレジスタ説明、“IDE Transfer Mode”を参照のこと

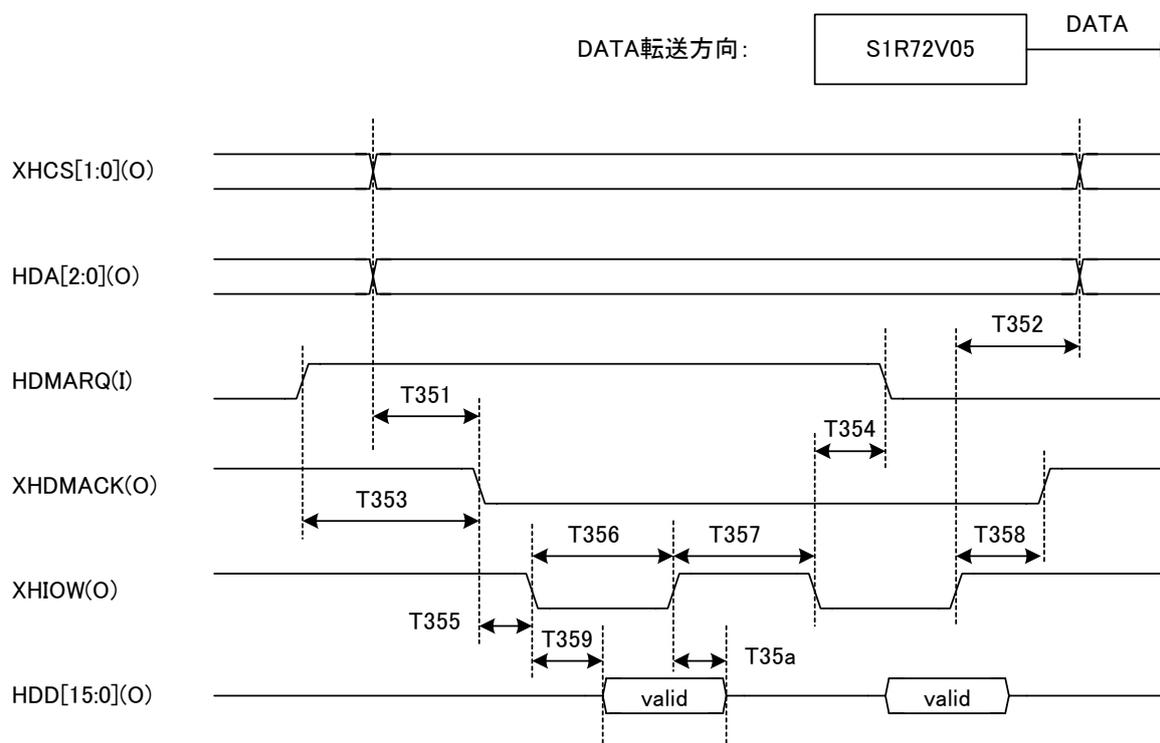
8.4.4.3. DMA Read Timing



記号	説明	min	typ	max	単位
T341	XHCS ↑、HDA → XHDMACK ↓ アドレスセットアップ時間	70	-	-	ns
T342	XHIOR ↑ → XHCS ↑、HDA アドレスホールド時間	50	-	-	ns
T343	HDMARQ ↑ → XHDMACK ↓ XHDMACK応答時間	17	-	-	ns
T344	XHIOR ↓ → HDMARQネゲート HDMARQホールド時間	0	-	-	ns
T345	XHDMACK ↓ → XHIOR ↓ XHDMACKセットアップ時間	0	-	-	ns
T346	XHIOR ↓ → XHIOR ↑ XHIORアサートパルス幅	-	(AP+4) * 16.7 - 3	-	ns
T347	XHIOR ↑ → XHIOR ↓ XHIORネゲートパルス幅	-	(NP+4) * 16.7 + 3	-	ns
T348	XHIOR ↑ → XHDMACK ↑ XHDMACKホールド時間	30	-	90	ns
T349	HDD → XHIOR ↑ データセットアップ時間	10	-	-	ns
T34a	XHIOR ↑ → HDD データバスホールド時間	0	-	-	ns

*1: AP=IDE_Tmod.AssertPulseWidth, NP=IDE_Tmod.NegatePulseWidth
詳細はレジスタ説明、“IDE Transfer Mode”を参照のこと

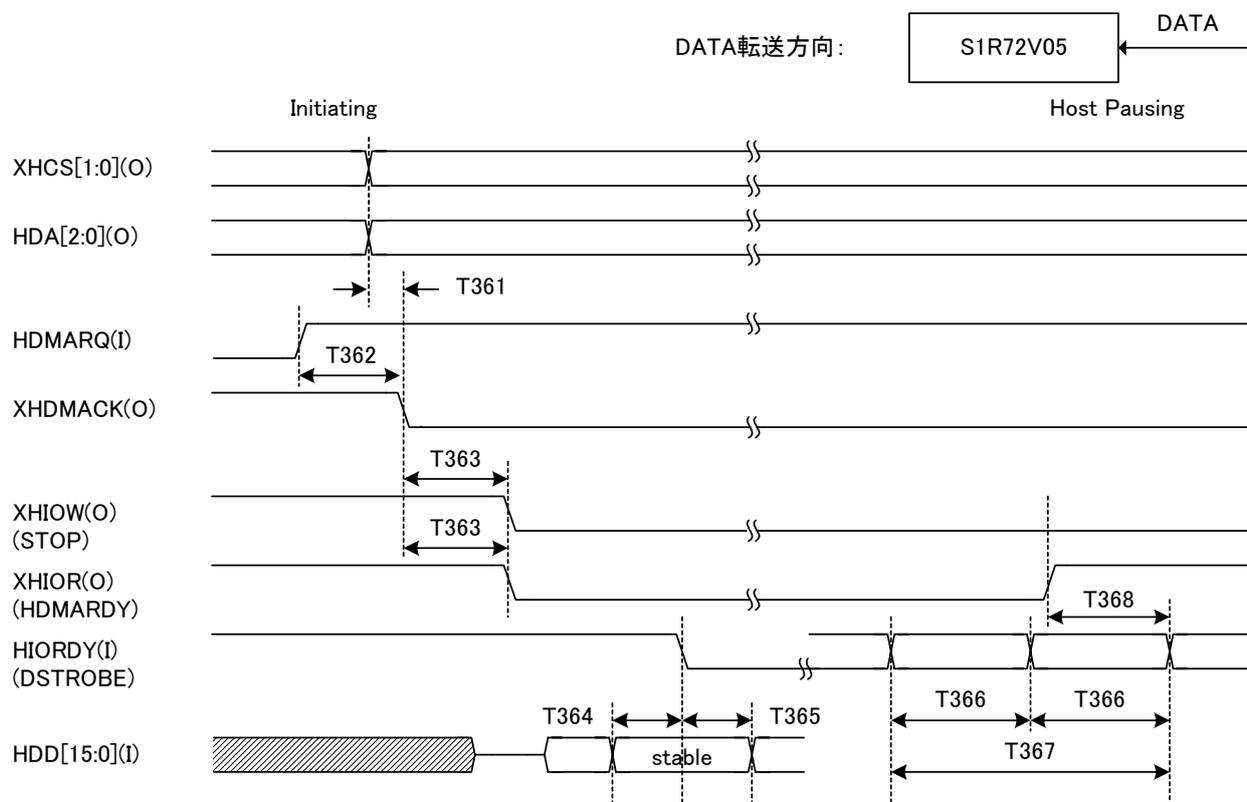
8.4.4.4. DMA Write Timing



記号	説明	min	typ	max	単位
T351	XHCS ↑、HDA → XHDMACK ↓ アドレスセットアップ時間	70	-	-	ns
T352	XHIOW ↑ → XHCS ↑、HDA アドレスホールド時間	50	-	-	ns
T353	HDMARQ ↑ → XHDMACK ↓ XHDMACK応答時間	17	-	-	ns
T354	XHIOW ↓ → HDMARQネゲート HDMARQホールド時間	0	-	-	ns
T355	XHDMACK ↓ → XHIOW ↓ XHDMACKセットアップ時間	0	-	-	ns
T356	XHIOW ↓ → XHIOW ↑ XHIOWアサートパルス幅	-	$(AP+4) * 16.7 - 3$	-	ns
T357	XHIOW ↑ → XHIOW ↓ XHIOWネゲートパルス幅	-	$(NP+4) * 16.7 + 3$	-	ns
T358	XHIOW ↑ → XHDMACK ↑ XHDMACKホールド時間	30	-	90	ns
T359	XHIOW ↓ → HDD データ出力遅延時間	0	-	10	ns
T35a	XHIOW ↑ → HDD データバスネゲート時間	33	-	45	ns

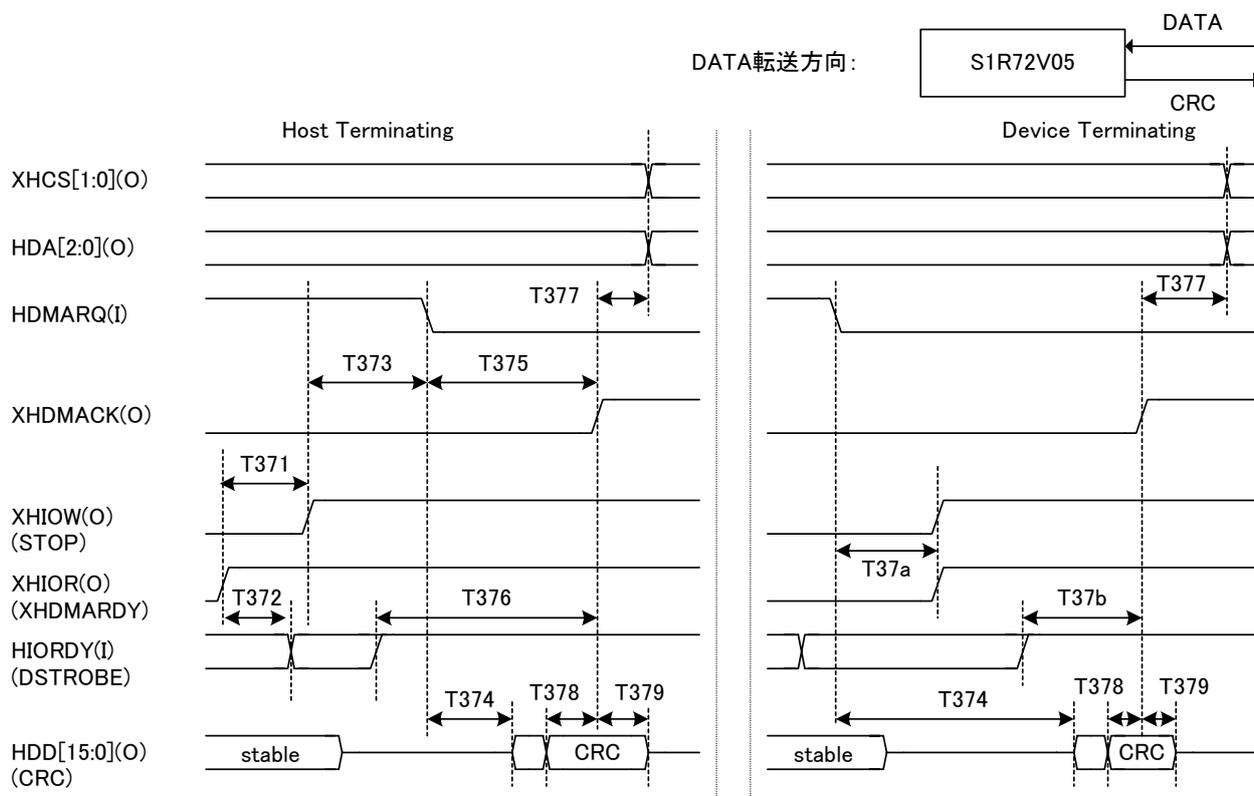
*1:AP=IDE_Tmod.AssertPulseWidth, NP=IDE_Tmod.NegatePulseWidth
詳細はレジスタ説明、“IDE Transfer Mode”を参照のこと

8.4.4.5. Ultra DMA Read Timing



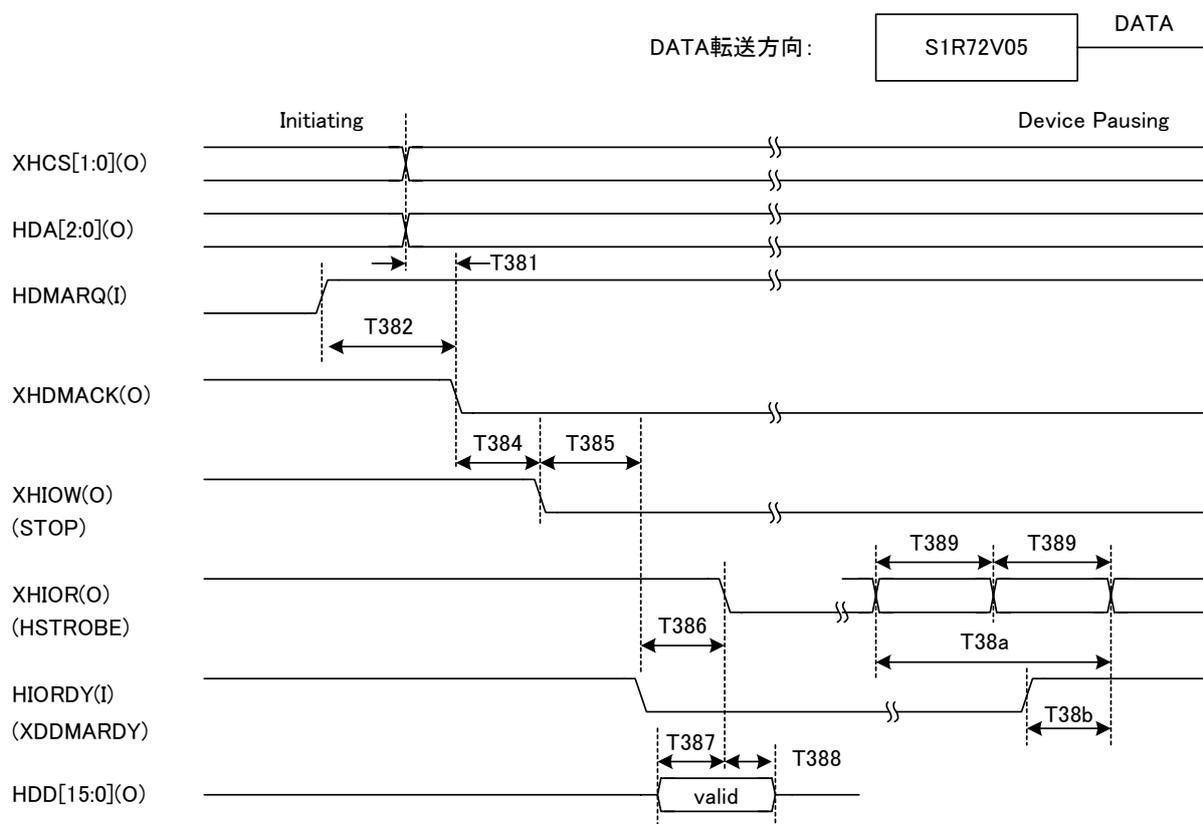
記号	説明	min	typ	max	単位
T361	XHCS ↑、HDA → XHDMACK ↓ アドレスセットアップ時間	80	-	-	ns
T362	HDMARQ ↑ → XHDMACK ↓ XHDMACK応答時間	65	-	-	ns
T363	XHDMACK ↓ → XHIOR(W) ↓ エンベロープ時間	28	-	40	ns
T364	HDD → HIORDY データセットアップ時間	4	-	-	ns
T365	HIORDY → HDD データホールド時間	4	-	-	ns
T366	HIORDY → HIORDY HIORDYサイクル時間	15	-	-	ns
T367	HIORDY → HIORDY HIORDYサイクル時間 × 2	30	-	-	ns
T368	XHIOR ↑ → HIORDY 最終のSTROBE時間	-	-	IDE規格 t_{RFS}	ns

Ultra DMA Read Timing (つづき)



記号	説明	min	typ	max	単位
T371	XHIOR ↑ → XHIOW ↑ STOPアサートまでの時間	180	-	-	ns
T372	XHIOR ↑ → HIORDY 最終のSTROBE時間	-	-	IDE規格 t_{RFS}	ns
T373	XHIOW ↑ → HDMARQ ↓ 制限付きインターロック時間	-	-	IDE規格 t_{LI}	ns
T374	HDMARQ ↓ → HDD 出力遅延時間	70	-	-	ns
T375	HDMARQ ↓ → XHDMACK ↑ 最小インターロック時間	160	-	-	ns
T376	HIORDY → XHDMACK ↑ 最小インターロック時間	110	-	-	ns
T377	XHDMACK ↑ → XHCS0,1 XHCS0,1ホールド時間	35	-	-	ns
T378	HDD(CRC) → XHDMACK ↑ CRCデータセットアップ時間	75	-	-	ns
T379	XHDMACK ↑ → HDD(CRC) CRCデータホールド時間	12	-	-	ns
T37a	HDMARQ ↓ → XHIOR ↑ 制限付きインターロック時間	20	-	38	ns
T37b	HIORDY → XHDMACK ↑ 最小インターロック時間	110	-	-	ns

8.4.4.6. Ultra DMA Write Timing



記号	説明	min	typ	max	単位
T381	XHCS ↑、HDA → XHDMACK ↓ アドレスセットアップ時間	80	-	-	ns
T382	HDMARQ ↑ → XHDMACK ↓ XHDMACK応答時間	65	-	-	ns
T384	XHDMACK ↓ → XHIOW ↓ エンベロープ時間	28	-	40	ns
T385	XHIOW ↓ → HIORDY ↓ 制限付きインターロック時間	IDE規格 t_{LI}	-	IDE規格 t_{LI}	ns
T386	HIORDY ↓ → XHIOR ↓ 制限無しインターロック時間	20	-	-	ns
T387	HDD → XHIOR ↓ データセットアップ時間	-	(cyc+1) * 16.7	-	ns
T388	XHIOR ↓ → HDD データホールド時間	-	(cyc+1) * 16.7	-	ns
T389	XHIOR → XHIOR XHIORサイクル時間	-	(cyc+2) * 16.7	-	ns
T38a	XHIOR → XHIOR XHIORサイクル時間 × 2	-	T389 * 2	-	ns
T38b	HIORDY ↑ → XHIOR 最終のSTROBE時間	20	-	38	ns

*1:cyc=UltraDMAcycle

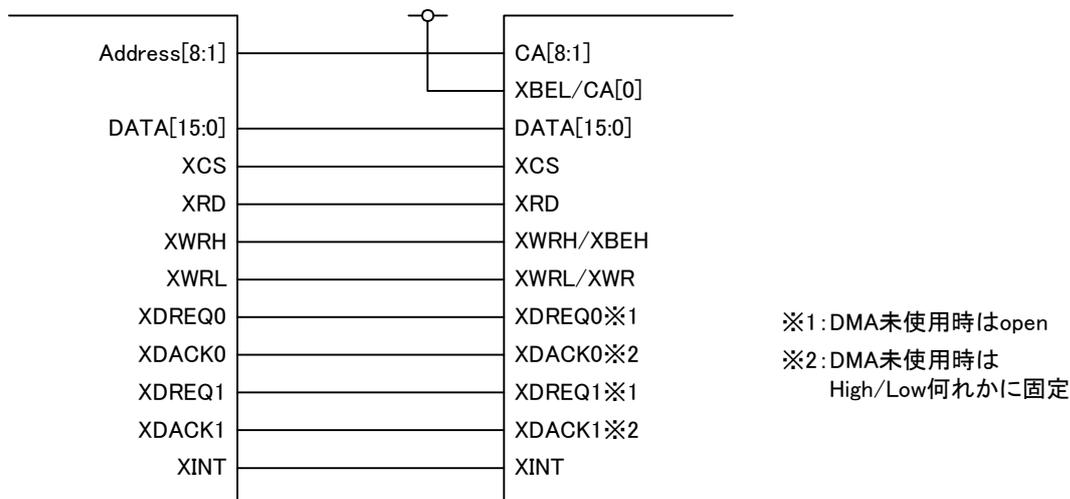
詳細はレジスタ説明、“IDE Ultra-DMA Transfer Mode”を参照のこと

8.4.5. USB I/F タイミング

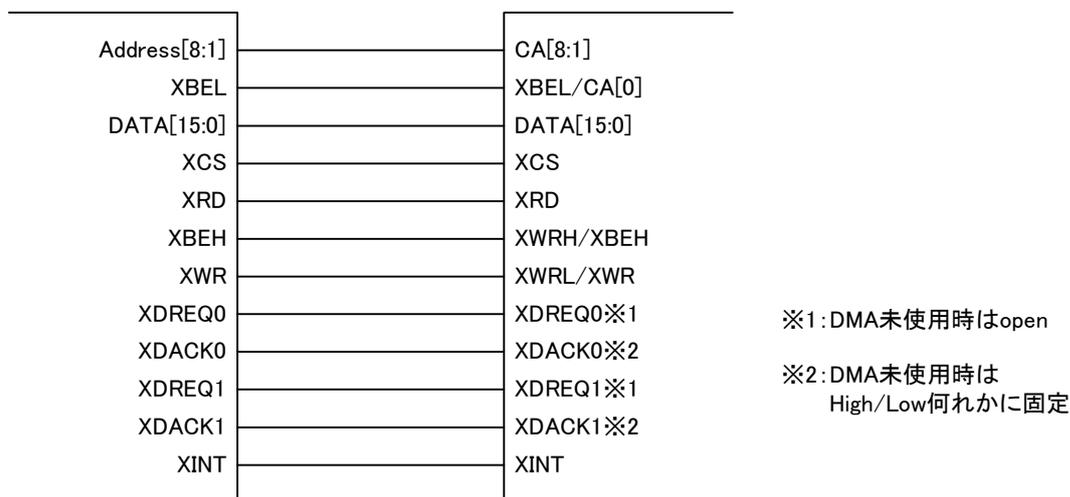
USB2.0 規格に準拠します。

9. 接続例

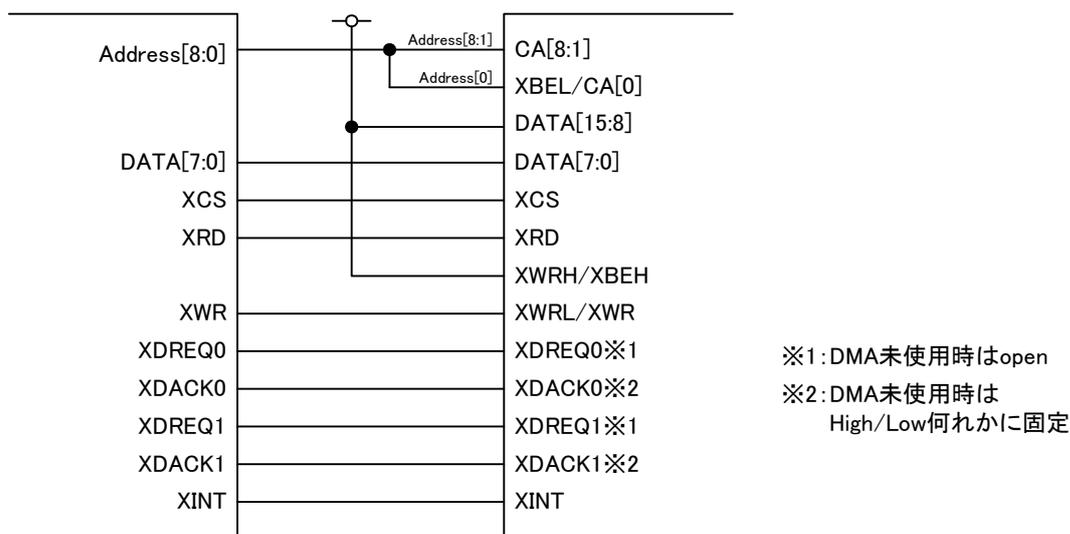
9.1. CPU I/F 接続例



16bit CPU (XWRH/XWRL) の接続例



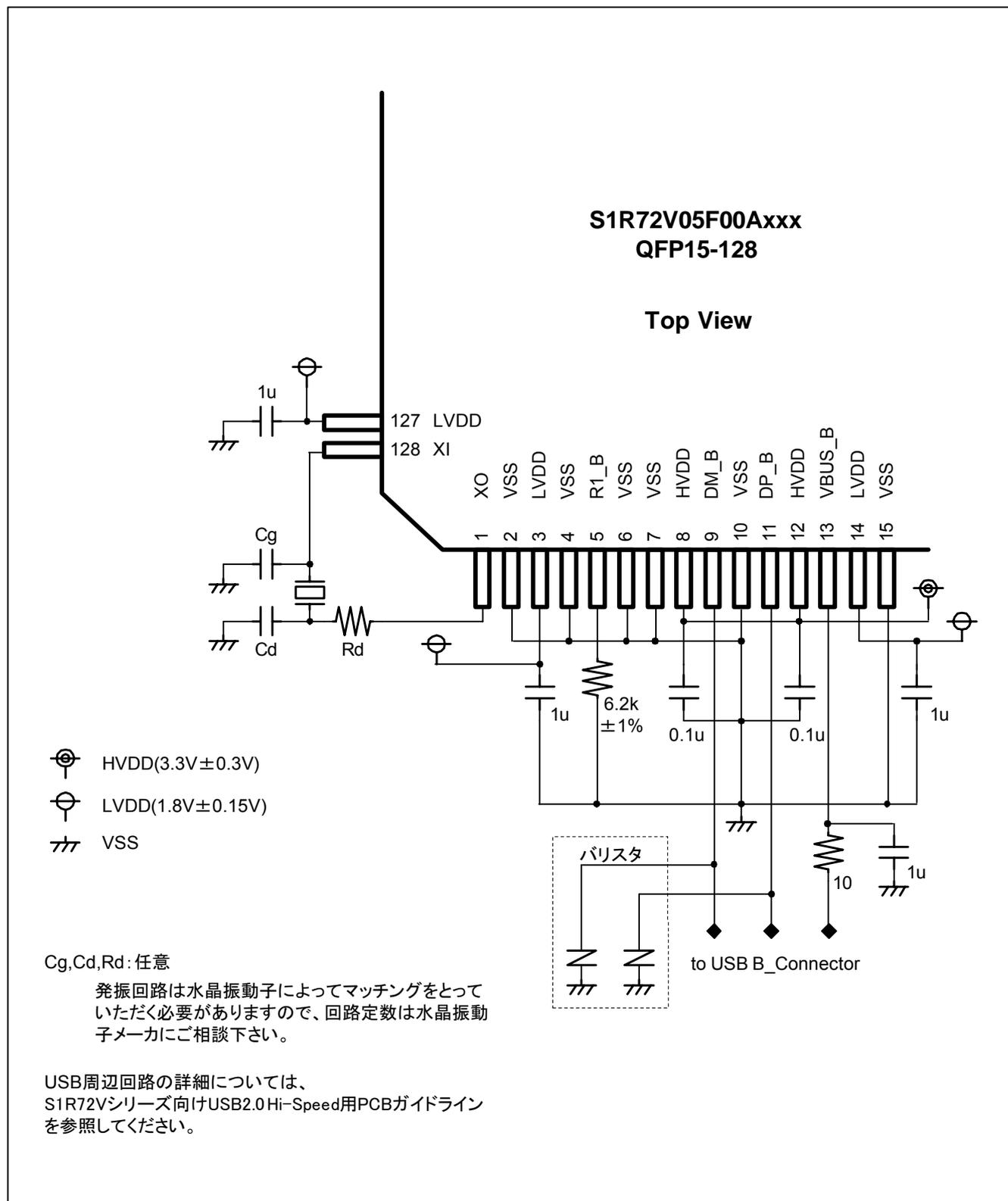
16bit CPU (XBEH/XBEL) の接続例



8bit CPU の接続例

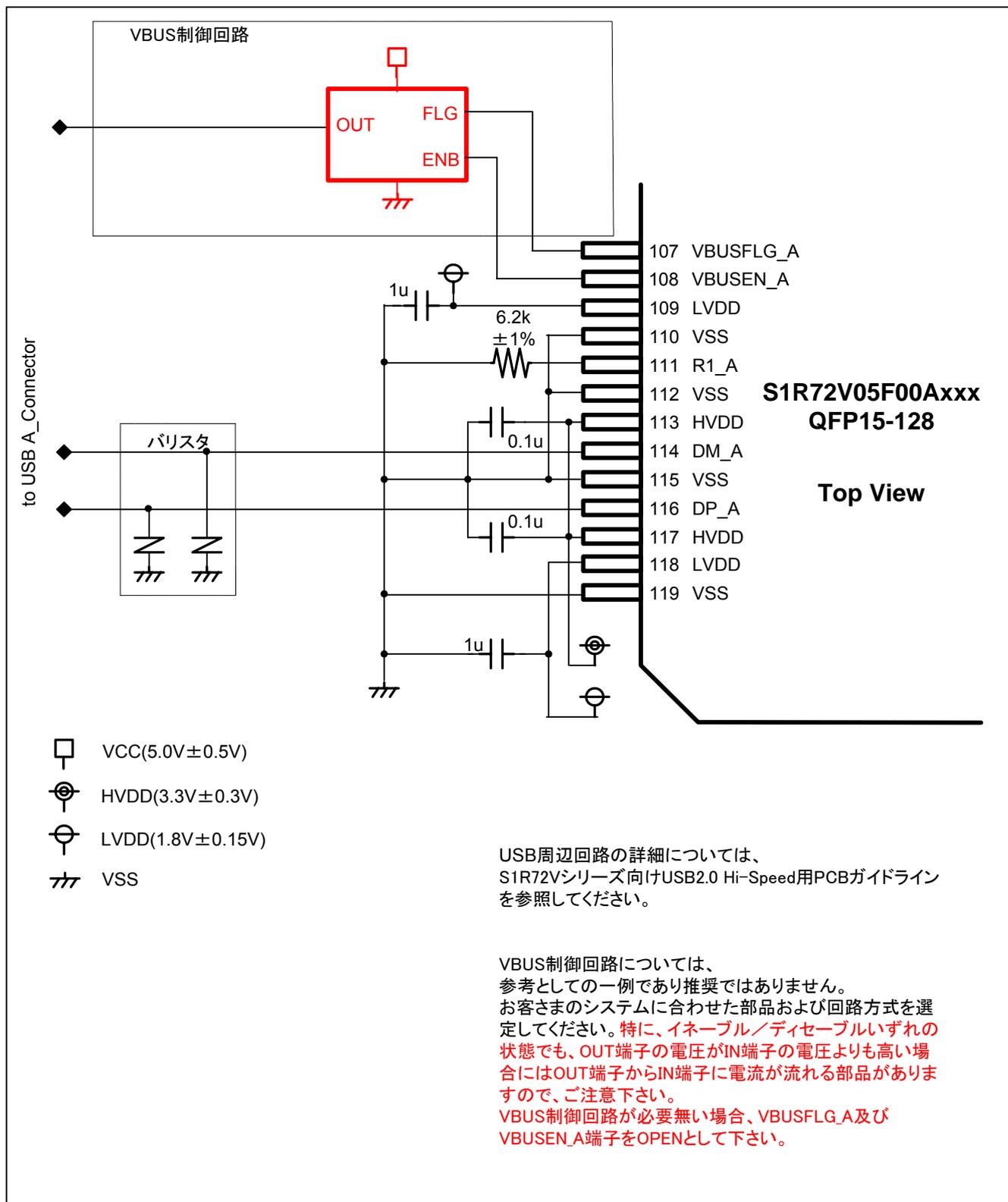
9.2. USB I/F 接続例

9.2.1. QFP15-128 の場合 (デバイス部周辺)



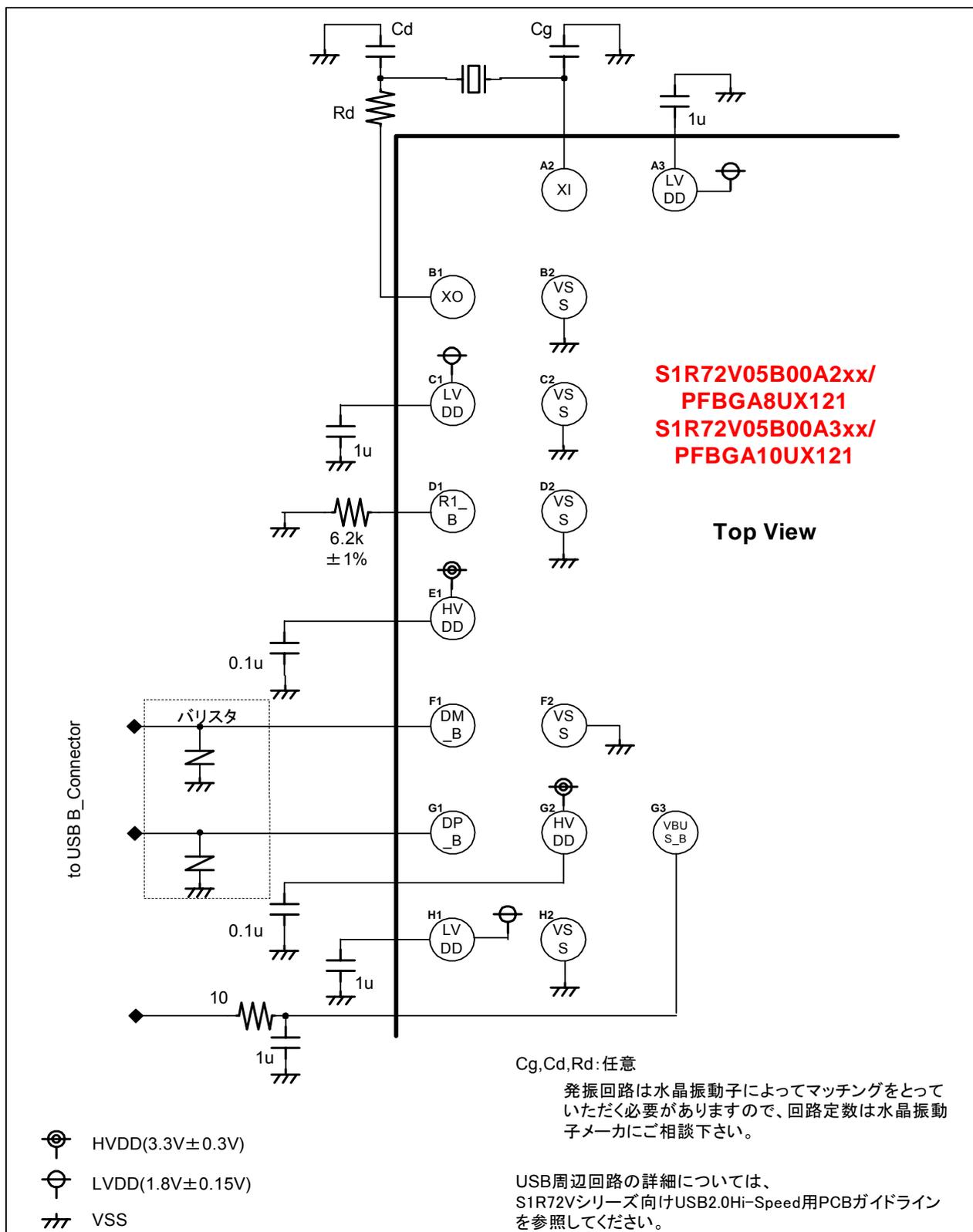
電源素子の性能は USB 信号波形品質影響を与える為、その選定には注意してください。

9.2.2. QFP15-128 の場合 (ホスト部周辺)



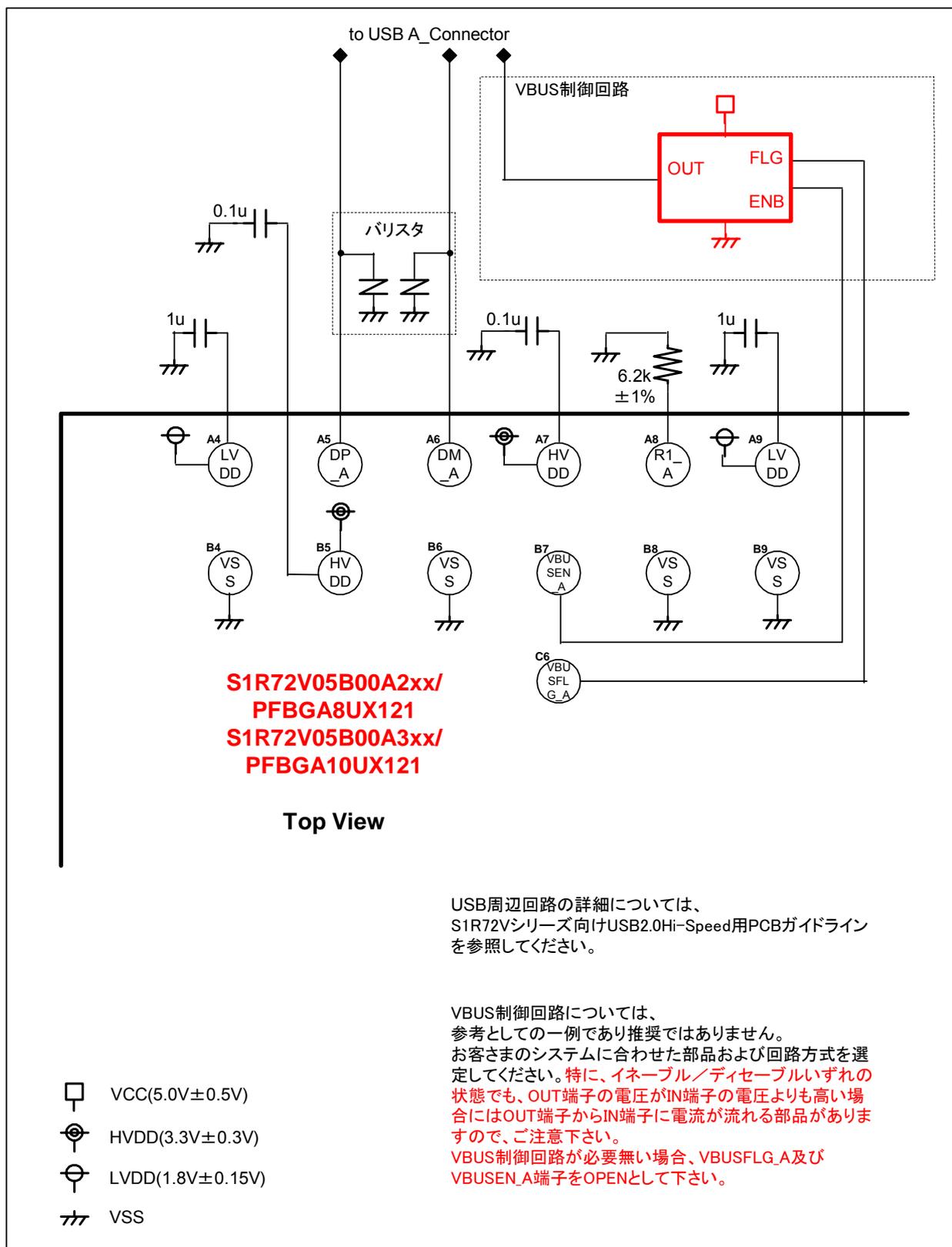
電源素子の性能は USB 信号波形品質影響を与える為、その選定には注意してください。

9.2.3. PFBGA8UX121/PFBGA10UX121 の場合 (デバイス部周辺)



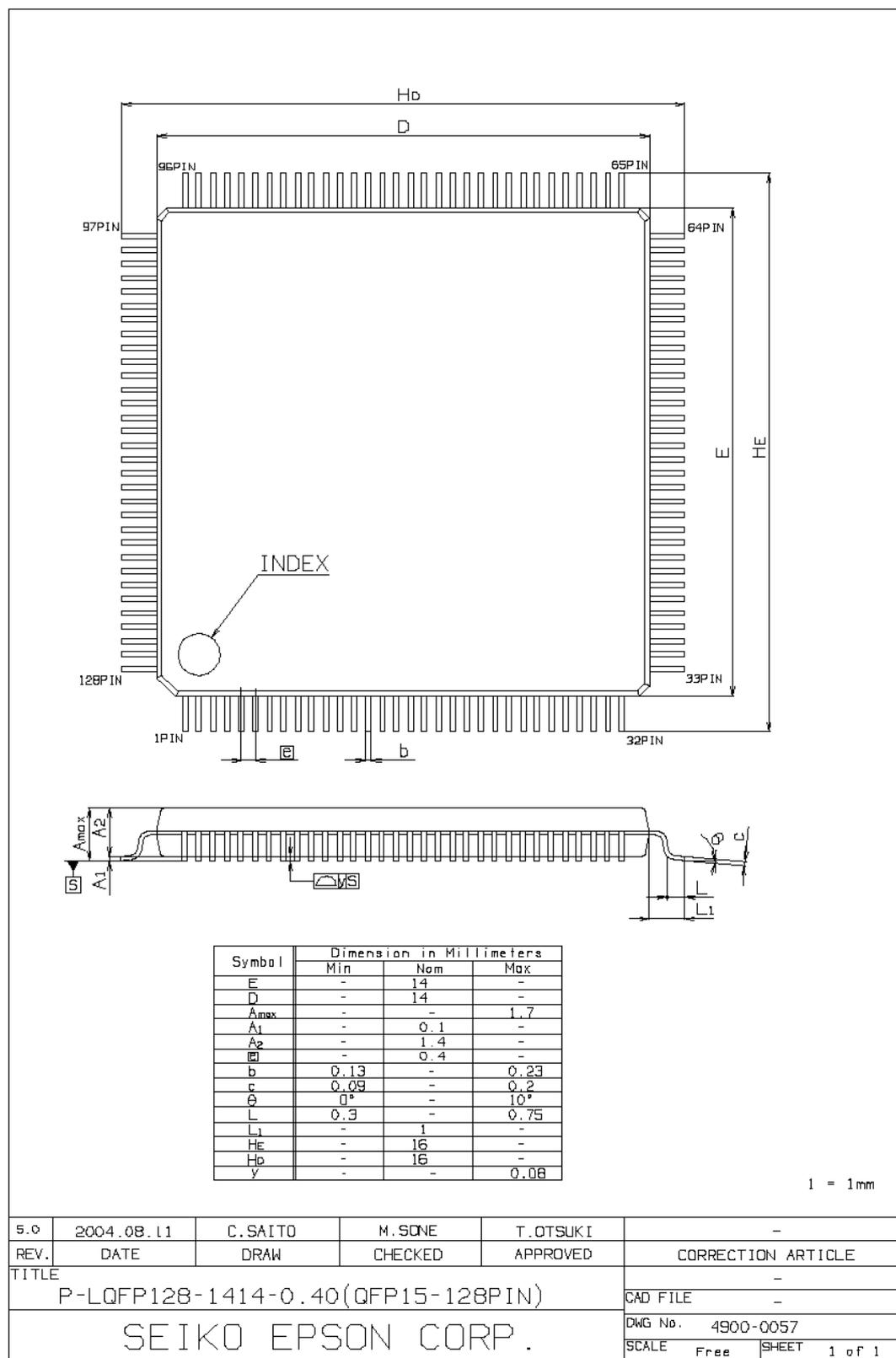
電源素子の性能は USB 信号波形品質影響を与える為、その選定には注意してください。

9.2.4. PFBGA8UX121/PFBGA10UX121 の場合 (ホスト部周辺)



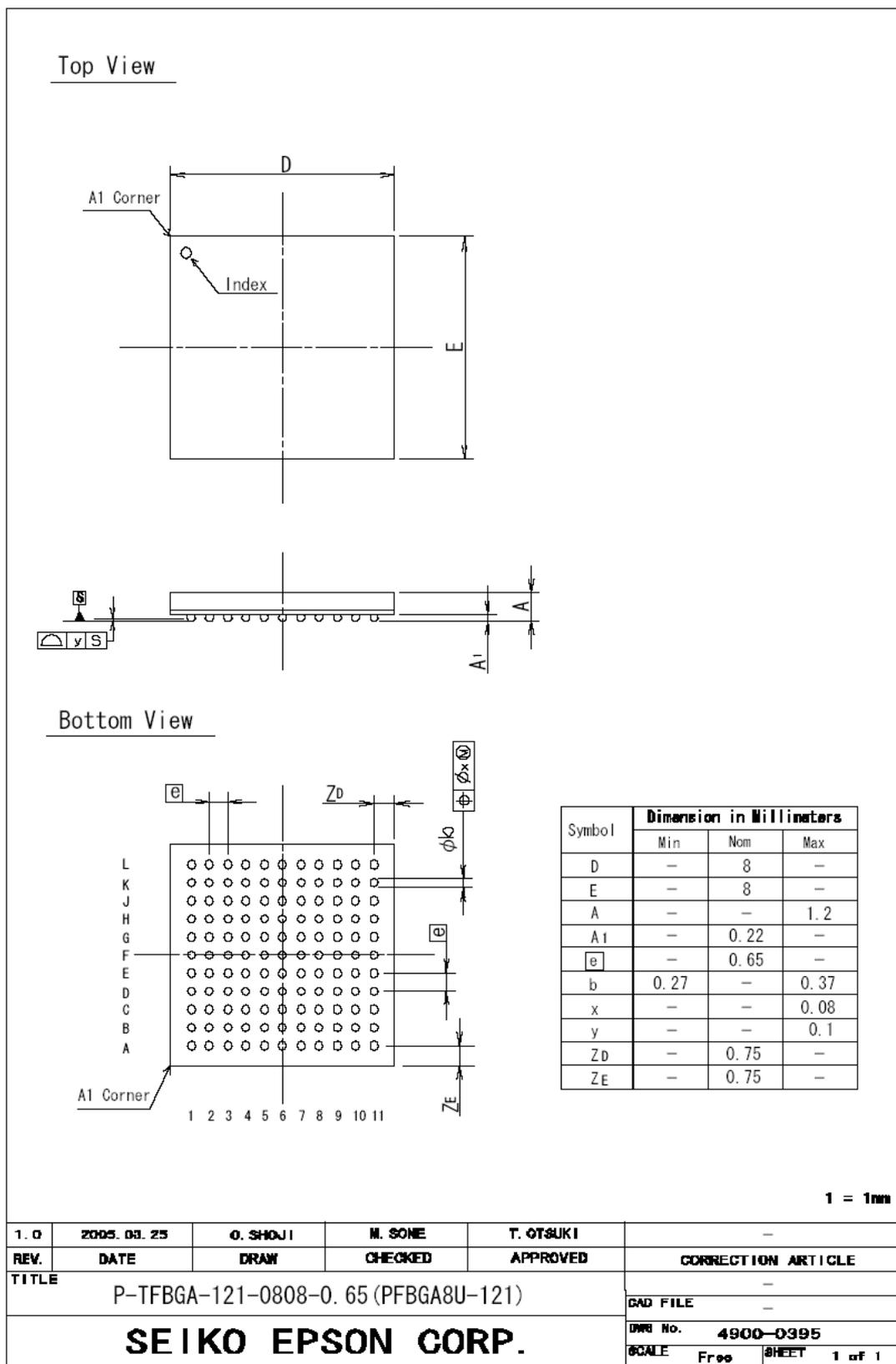
電源素子の性能は USB 信号波形品質影響を与える為、その選定には注意してください。

10. 外形寸法図
10.1. QFP パッケージ



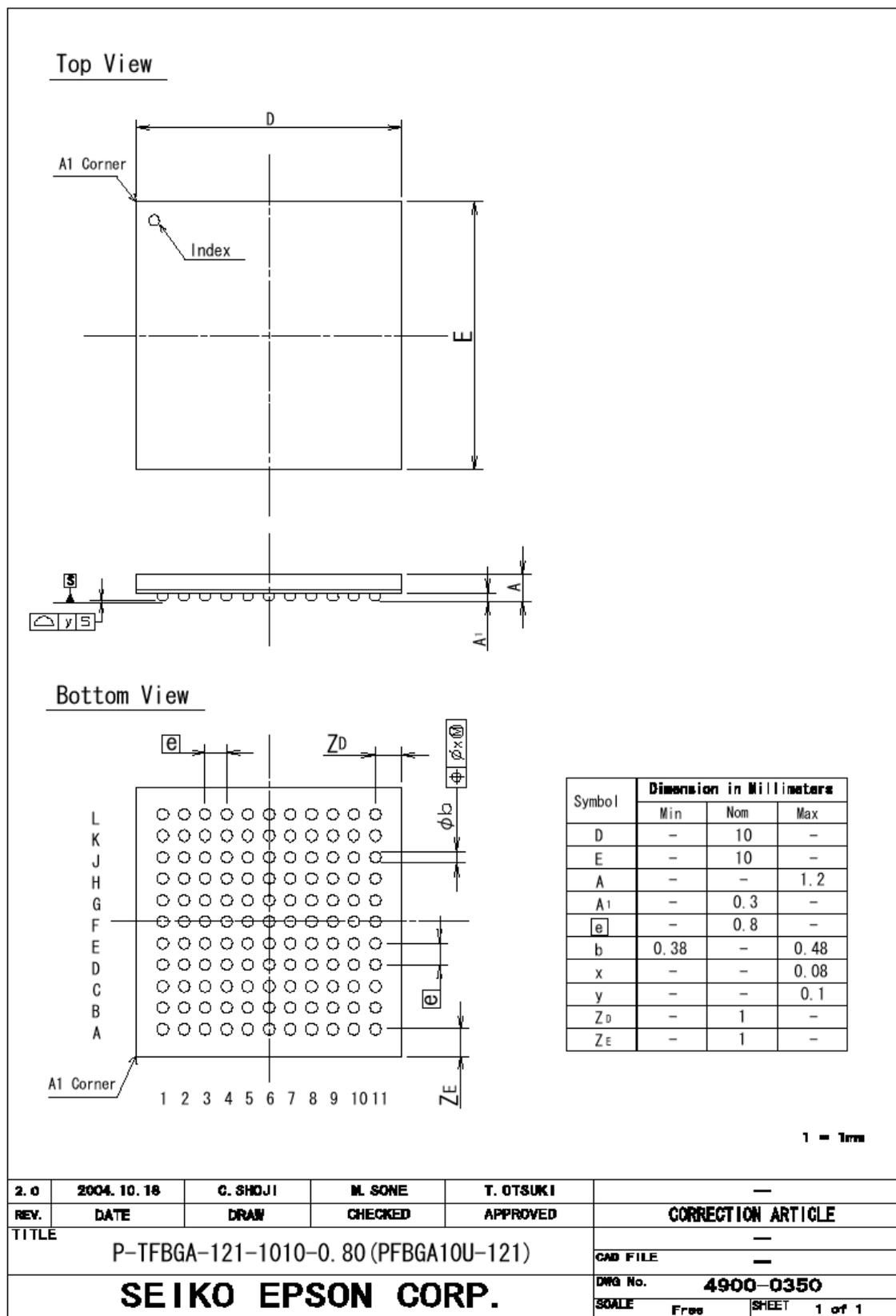
2990-002-01(Rev. 1.1)

10.2. BGA パッケージ(PFBGA8UX121)



2900-0002-01 (Rev. 1. 1)

10.3. BGA パッケージ(PFBGA10UX121)



2900-0002-01 (Rev. 1. 1)

Appendix A IDE_Config_1.Swap ビット設定

S1R72V05 の内部バスはビッグエンディアンで構成されており、[15:8]側がファーストバイトとなっています。これに対し IDE I/F はリトルエンディアンであり、[7:0]側がファーストバイトとなります。S1R72V05 では、IDE_Config_1.Swap ビットを使用して、S1R72V05 の内部バスと IDE I/F とのデータバスの結線を切り替えることが出来ます。

以下では、IDE_Config_1.Swap ビットの設定による H/W の動作に関し説明致します。

なお、表中の A⇒B は、A の値が B に反映される、となります。

●IDE_Control.IDE_Go ビットによるデータの DMA 転送

Swap	HDD[15:0]	
	IDE リード	IDE ライト
0	HDD[15:0] ⇒ 内部バス[15:0]	内部バス[15:0] ⇒ HDD[15:0]
1	HDD[15:0] ⇒ {内部バス[7:0], 内部バス[15:8]}	内部バス [15:0] ⇒ {HDD[7:0], HDD [15:8]}

●IDE データレジスタアクセス

Swap	IDE データレジスタ HDD[15:0]			
	IDE リード		IDE ライト	
	IDE_RegAdrs. IDE_RdReg	IDE_RegConfig. EnAutoStsRd	IDE_RegAdrs. IDE_WrReg	IDE_SeqWrRegControl. IDE_SeqWrReg
0	HDD[15:8] ⇒ IDE_RdRegValue_0 HDD[7:0] ⇒ IDE_RdRegValue_1	なし	IDE_WrRegValue_0 ⇒ HDD[15:8] IDE_WrRegValue_1 ⇒ HDD[7:0]	IDE_SeqWrRegValue(1 st) ⇒ HDD[7:0] IDE_SeqWrRegValue(2 nd) ⇒ HDD[15:8]
1	HDD[15:8] ⇒ IDE_RdRegValue_1 HDD[7:0] ⇒ IDE_RdRegValue_0	なし	IDE_WrRegValue_0 ⇒ HDD[7:0] IDE_WrRegValue_1 ⇒ HDD[15:8]	IDE_SeqWrRegValue(1 st) ⇒ HDD[15:8] IDE_SeqWrRegValue(2 nd) ⇒ HDD[7:0]

●IDE タスクファイルレジスタアクセス

Swap	IDE タスクファイルレジスタ HDD[7:0]			
	IDE リード		IDE ライト	
	IDE_RegAdrs. IDE_RdReg	IDE_RegConfig. EnAutoStsRd	IDE_RegAdrs. IDE_WrReg	IDE_SeqWrRegControl. IDE_SeqWrReg
0	HDD[7:0] ⇒ IDE_RdRegValue_0 HDD[7:0] ⇒ IDE_RdRegValue_1	同左	IDE_WrRegValue_1 ⇒ HDD[7:0]	IDE_SeqWrRegValue ⇒ HDD[7:0]
1	同上	同上	IDE_WrRegValue_0 ⇒ HDD[7:0]	同上

Appendix B リトルエンディアンの CPU への接続

S1R72V05 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 V05 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、即ち上位バイトには、V05 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 V05 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。

<FW>

リトルエンディアンの CPU にて、V05 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 ChipConfig.CPU_Endian ビットを"1"にセット
このレジスタのアドレス番地は V05 においては 0xB7 に割り当てられていますが、②を行うまでは、リトルエンディアンの CPU においては、0xB6 に割り当てられているように動作しています。これは、本 LSI の初期状態がビッグエンディアンであるために、ライト信号の上位、下位が逆転しているからです。
- ② 0xB9 番地をリード
このリード動作によって、本 LSI は CPU のバスの上位と下位を切り替えます。①を実行しただけでは切り替わっていないことにご注意願います。したがって、このリード動作の後には、レジスタが下記レジスタマップに示す通りのアドレス番地となります。

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。

Appendix B リトルエンディアンの CPU への接続

<レジスタマップ>

Big Endian

デバイス/ホスト共通レジスタ

0x00	MainIntStat
0x01	DeviceIntStat
0x02	HostIntStat
0x03	CPU_IntStat
0x04	IDE_IntStat
0x05	MediaFIFO_IntStat
0x06	
0x07	
0x08	
0x09	
0x0A	
0x0B	
0x0C	
0x0D	
0x0E	
0x0F	

0x10	MainIntEnb
0x11	DeviceIntEnb
0x12	HostIntEnb
0x13	CPU_IntEnb
0x14	IDE_IntEnb
0x15	MediaFIFO_IntEnb
0x16	
0x17	
0x18	
0x19	
0x1A	
0x1B	
0x1C	
0x1D	
0x1E	
0x1F	

0x20	RevisionNum
0x21	ChipReset
0x22	PM_Control_0
0x23	PM_Control_1
0x24	WakeupTim_H
0x25	WakeupTim_L
0x26	H_USB_Control
0x27	H_XcvrControl
0x28	D_USB_Status
0x29	H_USB_Status
0x2A	
0x2B	
0x2C	
0x2D	
0x2E	
0x2F	

0x30	FIFO_Rd_0
0x31	FIFO_Rd_1
0x32	FIFO_Wr_0
0x33	FIFO_Wr_1
0x34	FIFO_RdRemain_H
0x35	FIFO_RdRemain_L
0x36	FIFO_WrRemain_H
0x37	FIFO_WrRemain_L
0x38	FIFO_ByteRd
0x39	
0x3A	
0x3B	
0x3C	
0x3D	
0x3E	
0x3F	

Little Endian

Little Endian時に上位、下位が変わるレジスタ

デバイス/ホスト共通レジスタ

0x00	MainIntStat
0x01	DeviceIntStat
0x02	HostIntStat
0x03	CPU_IntStat
0x04	IDE_IntStat
0x05	MediaFIFO_IntStat
0x06	
0x07	
0x08	
0x09	
0x0A	
0x0B	
0x0C	
0x0D	
0x0E	
0x0F	

0x10	MainIntEnb
0x11	DeviceIntEnb
0x12	HostIntEnb
0x13	CPU_IntEnb
0x14	IDE_IntEnb
0x15	MediaFIFO_IntEnb
0x16	
0x17	
0x18	
0x19	
0x1A	
0x1B	
0x1C	
0x1D	
0x1E	
0x1F	

0x20	RevisionNum
0x21	ChipReset
0x22	PM_Control_0
0x23	PM_Control_1
0x24	WakeupTim_L
0x25	WakeupTim_H
0x26	H_USB_Control
0x27	H_XcvrControl
0x28	D_USB_Status
0x29	H_USB_Status
0x2A	
0x2B	
0x2C	
0x2D	
0x2E	
0x2F	

0x30	FIFO_Rd_0
0x31	FIFO_Rd_1
0x32	FIFO_Wr_0
0x33	FIFO_Wr_1
0x34	FIFO_RdRemain_L
0x35	FIFO_RdRemain_H
0x36	FIFO_WrRemain_L
0x37	FIFO_WrRemain_H
0x38	FIFO_ByteRd
0x39	
0x3A	
0x3B	
0x3C	
0x3D	
0x3E	
0x3F	

Appendix B リトルエンディアンの CPU への接続

Big Endian

デバイス/ホスト共通レジスタ

0x40	RAM_RdAdrs_H
0x41	RAM_RdAdrs_L
0x42	RAM_RdControl
0x43	RAM_RdCount
0x44	RAM_WrAdrs_H
0x45	RAM_WrAdrs_L
0x46	RAM_WrDoor_0
0x47	RAM_WrDoor_1
0x48	MediaFIFO_Control
0x49	ClrAllMediaFIFO_Join
0x4A	MediaFIFO_Join
0x4B	
0x4C	
0x4D	
0x4E	
0x4F	

0x50	RAM_Rd_00
0x51	RAM_Rd_01
0x52	RAM_Rd_02
0x53	RAM_Rd_03
0x54	RAM_Rd_04
0x55	RAM_Rd_05
0x56	RAM_Rd_06
0x57	RAM_Rd_07
0x58	RAM_Rd_08
0x59	RAM_Rd_09
0x510	RAM_Rd_0A
0x511	RAM_Rd_0B
0x512	RAM_Rd_0C
0x513	RAM_Rd_0D
0x514	RAM_Rd_0E
0x515	RAM_Rd_0F

0x60	RAM_Rd_10
0x61	RAM_Rd_11
0x62	RAM_Rd_12
0x63	RAM_Rd_13
0x64	RAM_Rd_14
0x65	RAM_Rd_15
0x66	RAM_Rd_16
0x67	RAM_Rd_17
0x68	RAM_Rd_18
0x69	RAM_Rd_19
0x6A	RAM_Rd_1A
0x6B	RAM_Rd_1B
0x6C	RAM_Rd_1C
0x6D	RAM_Rd_1D
0x6E	RAM_Rd_1E
0x6F	RAM_Rd_1F

0x70	
0x71	DMA0_Config
0x72	DMA0_Control
0x73	
0x74	DMA0_Remain_H
0x75	DMA0_Remain_L
0x76	
0x77	
0x78	DMA0_Count_HH
0x79	DMA0_Count_HL
0x7A	DMA0_Count_LH
0x7B	DMA0_Count_LL
0x7C	DMA0_RdData_0
0x7D	DMA0_RdData_1
0x7E	DMA0_WrData_0
0x7F	DMA0_WrData_1

Little Endian

Little Endian時に上位、下位が変わるレジスタ

デバイス/ホスト共通レジスタ

0x40	RAM_RdAdrs_L
0x41	RAM_RdAdrs_H
0x42	RAM_RdControl
0x43	RAM_RdCount
0x44	RAM_WrAdrs_L
0x45	RAM_WrAdrs_H
0x46	RAM_WrDoor_0
0x47	RAM_WrDoor_1
0x48	MediaFIFO_Control
0x49	ClrAllMediaFIFO_Join
0x4A	MediaFIFO_Join
0x4B	
0x4C	
0x4D	
0x4E	
0x4F	

0x50	RAM_Rd_00
0x51	RAM_Rd_01
0x52	RAM_Rd_02
0x53	RAM_Rd_03
0x54	RAM_Rd_04
0x55	RAM_Rd_05
0x56	RAM_Rd_06
0x57	RAM_Rd_07
0x58	RAM_Rd_08
0x59	RAM_Rd_09
0x5A	RAM_Rd_0A
0x5B	RAM_Rd_0B
0x5C	RAM_Rd_0C
0x5D	RAM_Rd_0D
0x5E	RAM_Rd_0E
0x5F	RAM_Rd_0F

0x60	RAM_Rd_10
0x61	RAM_Rd_11
0x62	RAM_Rd_12
0x63	RAM_Rd_13
0x64	RAM_Rd_14
0x65	RAM_Rd_15
0x66	RAM_Rd_16
0x67	RAM_Rd_17
0x68	RAM_Rd_18
0x69	RAM_Rd_19
0x6A	RAM_Rd_1A
0x6B	RAM_Rd_1B
0x6C	RAM_Rd_1C
0x6D	RAM_Rd_1D
0x6E	RAM_Rd_1E
0x6F	RAM_Rd_1F

0x70	
0x71	DMA0_Config
0x72	DMA0_Control
0x73	
0x74	DMA0_Remain_L
0x75	DMA0_Remain_H
0x76	
0x77	
0x78	DMA0_Count_HL
0x79	DMA0_Count_HH
0x7A	DMA0_Count_LL
0x7B	DMA0_Count_LH
0x7C	DMA0_RdData_0
0x7D	DMA0_RdData_1
0x7E	DMA0_WrData_0
0x7F	DMA0_WrData_1

Appendix B リトルエンディアンの CPU への接続

Big Endian

デバイス/ホスト共通レジスタ

0x80	
0x81	DMA1_Config
0x82	DMA1_Control
0x83	
0x84	DMA1_Remain_H
0x85	DMA1_Remain_L
0x86	
0x87	
0x88	DMA1_Count_HH
0x89	DMA1_Count_HL
0x8A	DMA1_Count_LH
0x8B	DMA1_Count_LL
0x8C	DMA1_RdData_0
0x8D	DMA1_RdData_1
0x8E	DMA1_WrData_0
0x8F	DMA1_WrData_1

0x90	IDE_Status
0x91	IDE_Control
0x92	IDE_Config_0
0x93	IDE_Config_1
0x94	IDE_Rmod
0x95	IDE_Tmod
0x96	IDE_Umod
0x97	
0x98	
0x99	
0x9A	IDE_CRC_H
0x9B	IDE_CRC_L
0x9C	
0x9D	IDE_Count_H
0x9E	IDE_Count_M
0x9F	IDE_Count_L

0xA0	IDE_RegAdrs
0xA1	
0xA2	IDE_RdRegValue_0
0xA3	IDE_RdRegValue_1
0xA4	IDE_WrRegValue_0
0xA5	IDE_WrRegValue_1
0xA6	IDE_SeqWrRegControl
0xA7	IDE_SeqWrRegCnt
0xA8	IDE_SeqWrRegAdrs
0xA9	IDE_SeqWrRegValue
0xAA	
0xAB	
0xAC	IDE_RegConfig
0xAD	
0xAE	
0xAF	

0xB0	
0xB1	<i>HostDeviceSel</i>
0xB2	
0xB3	<i>ModeProtect</i>
0xB4	
0xB5	<i>ClkSelect</i>
0xB6	
0xB7	<i>ChipConfig</i>
0xB8	
0xB9	<i>CPU_ChgEndian</i>
0xBA	
0xBB	
0xBC	
0xBD	
0xBE	
0xBF	

0xC0~0xDF:Reserved

Little Endian

Little Endian時に上位、下位が変わるレジスタ

デバイス/ホスト共通レジスタ

0x80	
0x81	DMA1_Config
0x82	DMA1_Control
0x83	
0x84	DMA1_Remain_L
0x85	DMA1_Remain_H
0x86	
0x87	
0x88	DMA1_Count_HL
0x89	DMA1_Count_HH
0x8A	DMA1_Count_LL
0x8B	DMA1_Count_LH
0x8C	DMA1_RdData_0
0x8D	DMA1_RdData_1
0x8E	DMA1_WrData_0
0x8F	DMA1_WrData_1

0x90	IDE_Status
0x91	IDE_Control
0x92	IDE_Config_0
0x93	IDE_Config_1
0x94	IDE_Rmod
0x95	IDE_Tmod
0x96	IDE_Umod
0x97	
0x98	
0x99	
0x9A	IDE_CRC_L
0x9B	IDE_CRC_H
0x9C	IDE_Count_H
0x9D	
0x9E	IDE_Count_L
0x9F	IDE_Count_M

0xA0	IDE_RegAdrs
0xA1	
0xA2	IDE_RdRegValue_0
0xA3	IDE_RdRegValue_1
0xA4	IDE_WrRegValue_0
0xA5	IDE_WrRegValue_1
0xA6	IDE_SeqWrRegControl
0xA7	IDE_SeqWrRegCnt
0xA8	IDE_SeqWrRegAdrs
0xA9	IDE_SeqWrRegValue
0xAA	
0xAB	
0xAC	IDE_RegConfig
0xAD	
0xAE	
0xAF	

0xB0	
0xB1	<i>HostDeviceSel</i>
0xB2	
0xB3	<i>ModeProtect</i>
0xB4	
0xB5	<i>ClkSelect</i>
0xB6	
0xB7	<i>ChipConfig</i>
0xB8	
0xB9	<i>CPU_ChgEndian</i>
0xBA	
0xBB	
0xBC	
0xBD	
0xBE	
0xBF	

0xC0~0xDF:Reserved

Appendix B リトルエンディアンの CPU への接続

デバイス・レジスタ

Big Endian

0xE0	D_SIE_IntStat
0xE1	
0xE2	D_FIFO_IntStat
0xE3	D_BulkIntStat
0xE4	D_EPrIntStat
0xE5	D_EP0IntStat
0xE6	D_EPaIntStat
0xE7	D_EPbIntStat
0xE8	D_EPcIntStat
0xE9	
0xEA	
0xEB	
0xEC	
0xED	
0xEE	
0xEF	

0xF0	D_SIE_IntEnb
0xF1	
0xF2	D_FIFO_IntEnb
0xF3	D_BulkIntEnb
0xF4	D_EPrIntEnb
0xF5	D_EP0IntEnb
0xF6	D_EPaIntEnb
0xF7	D_EPbIntEnb
0xF8	D_EPcIntEnb
0xF9	
0xFA	
0xFB	
0xFC	
0xFD	
0xFE	
0xFF	

0x100	D_Reset
0x101	
0x102	D_NegoControl
0x103	
0x104	D_ClrAllEPnJoin
0x105	D_XcvrControl
0x106	D_USB_Test
0x107	
0x108	D_EPnControl
0x109	D_EPrFIFO_Clr
0x10A	D_BulkOnlyControl
0x10B	D_BulkOnlyConfig
0x10C	
0x10D	
0x10E	
0x10F	

0x110	D_EP0SETUP_0
0x111	D_EP0SETUP_1
0x112	D_EP0SETUP_2
0x113	D_EP0SETUP_3
0x114	D_EP0SETUP_4
0x115	D_EP0SETUP_5
0x116	D_EP0SETUP_6
0x117	D_EP0SETUP_7
0x118	D_USB_Address
0x119	
0x11A	D_SETUP_Control
0x11B	
0x11C	
0x11D	
0x11E	D_FrameNumber_H
0x11F	D_FrameNumber_L

デバイス・レジスタ

Little Endian

Little Endian時に上位、下位が変わるレジスタ

0xE0	D_SIE_IntStat
0xE1	
0xE2	D_FIFO_IntStat
0xE3	D_BulkIntStat
0xE4	D_EPrIntStat
0xE5	D_EP0IntStat
0xE6	D_EPaIntStat
0xE7	D_EPbIntStat
0xE8	D_EPcIntStat
0xE9	
0xEA	
0xEB	
0xEC	
0xED	
0xEE	
0xEF	

0xF0	D_SIE_IntEnb
0xF1	
0xF2	D_FIFO_IntEnb
0xF3	D_BulkIntEnb
0xF4	D_EPrIntEnb
0xF5	D_EP0IntEnb
0xF6	D_EPaIntEnb
0xF7	D_EPbIntEnb
0xF8	D_EPcIntEnb
0xF9	
0xFA	
0xFB	
0xFC	
0xFD	
0xFE	
0xFF	

0x100	D_Reset
0x101	
0x102	D_NegoControl
0x103	
0x104	D_ClrAllEPnJoin
0x105	D_XcvrControl
0x106	D_USB_Test
0x107	
0x108	D_EPnControl
0x109	D_EPrFIFO_Clr
0x10A	D_BulkOnlyControl
0x10B	D_BulkOnlyConfig
0x10C	
0x10D	
0x10E	
0x10F	

0x110	D_EP0SETUP_0
0x111	D_EP0SETUP_1
0x112	D_EP0SETUP_2
0x113	D_EP0SETUP_3
0x114	D_EP0SETUP_4
0x115	D_EP0SETUP_5
0x116	D_EP0SETUP_6
0x117	D_EP0SETUP_7
0x118	D_USB_Address
0x119	
0x11A	D_SETUP_Control
0x11B	
0x11C	
0x11D	
0x11E	D_FrameNumber_L
0x11F	D_FrameNumber_H

Appendix B リトルエンディアンの CPU への接続

デバイス・レジスタ

Big Endian

0x120	D_EP0MaxSize
0x121	D_EP0Control
0x122	D_EP0ControlIN
0x123	D_EP0ControlOUT
0x124	
0x125	D_EP0Join
0x126	
0x127	
0x128	
0x129	
0x12A	
0x12B	
0x12C	
0x12D	
0x12E	
0x12F	

0x130	D_EPaMaxSize_H
0x131	D_EPaMaxSize_L
0x132	D_EPaConfig_0
0x133	
0x134	D_EPaControl
0x135	D_EPaJoin
0x136	
0x137	
0x138	
0x139	
0x13A	
0x13B	
0x13C	
0x13D	
0x13E	
0x13F	

0x140	D_EPbMaxSize_H
0x141	D_EPbMaxSize_L
0x142	D_EPbConfig_0
0x143	
0x144	D_EPbControl
0x145	D_EPbJoin
0x146	
0x147	
0x148	
0x149	
0x14A	
0x14B	
0x14C	
0x14D	
0x14E	
0x14F	

0x150	D_EPcMaxSize_H
0x151	D_EPcMaxSize_L
0x152	D_EPcConfig_0
0x153	
0x154	D_EPcControl
0x155	D_EPcJoin
0x156	
0x157	
0x158	
0x159	
0x15A	
0x15B	
0x15C	
0x15D	
0x15E	
0x15F	

デバイス・レジスタ

Little Endian

Little Endian時に上位、下位が変わるレジスタ

0x120	D_EP0MaxSize
0x121	D_EP0Control
0x122	D_EP0ControlIN
0x123	D_EP0ControlOUT
0x124	
0x125	D_EP0Join
0x126	
0x127	
0x128	
0x129	
0x12A	
0x12B	
0x12C	
0x12D	
0x12E	
0x12F	

0x130	D_EPaMaxSize_L
0x131	D_EPaMaxSize_H
0x132	D_EPaConfig_0
0x133	
0x134	D_EPaControl
0x135	D_EPaJoin
0x136	
0x137	
0x138	
0x139	
0x13A	
0x13B	
0x13C	
0x13D	
0x13E	
0x13F	

0x140	D_EPbMaxSize_L
0x141	D_EPbMaxSize_H
0x142	D_EPbConfig_0
0x143	
0x144	D_EPbControl
0x145	D_EPbJoin
0x146	
0x147	
0x148	
0x149	
0x14A	
0x14B	
0x14C	
0x14D	
0x14E	
0x14F	

0x150	D_EPcMaxSize_L
0x151	D_EPcMaxSize_H
0x152	D_EPcConfig_0
0x153	
0x154	D_EPcControl
0x155	D_EPcJoin
0x156	
0x157	
0x158	
0x159	
0x15A	
0x15B	
0x15C	
0x15D	
0x15E	
0x15F	

Appendix B リトルエンディアンの CPU への接続

デバイス・レジスタ

Big Endian

0x160	D_DescAdrs_H
0x161	D_DescAdrs_L
0x162	D_DescSize_H
0x163	D_DescSize_L
0x164	
0x165	
0x166	
0x167	
0x168	
0x169	
0x16A	
0x16B	
0x16C	
0x16D	
0x16E	
0x16F	

0x170	D_DMA0_FIFO_Control
0x171	
0x172	D_DMA1_FIFO_Control
0x173	
0x174	
0x175	
0x176	
0x177	
0x178	
0x179	
0x17A	
0x17B	
0x17C	
0x17D	
0x17E	
0x17F	

0x180	
0x181	
0x182	
0x183	
0x184	D_EPaStartAdrs_H
0x185	D_EPaStartAdrs_L
0x186	
0x187	
0x188	D_EPbStartAdrs_H
0x189	D_EPbStartAdrs_L
0x18A	
0x18B	
0x18C	D_EPcStartAdrs_H
0x18D	D_EPcStartAdrs_L
0x18E	D_EPcEndAdrs_H
0x18F	D_EPcEndAdrs_L

0x190~0x1DF:Reserved

0x1E0	(Reserved)
0x1E1	D_ModeControl

0x1E2~0x1FF:Reserved

デバイス・レジスタ

Little Endian

Little Endian時に上位、下位が変わるレジスタ

0x160	D_DescAdrs_L
0x161	D_DescAdrs_H
0x162	D_DescSize_L
0x163	D_DescSize_H
0x164	
0x165	
0x166	
0x167	
0x168	
0x169	
0x16A	
0x16B	
0x16C	
0x16D	
0x16E	
0x16F	

0x170	D_DMA0_FIFO_Control
0x171	
0x172	D_DMA1_FIFO_Control
0x173	
0x174	
0x175	
0x176	
0x177	
0x178	
0x179	
0x17A	
0x17B	
0x17C	
0x17D	
0x17E	
0x17F	

0x180	
0x181	
0x182	
0x183	
0x184	D_EPaStartAdrs_L
0x185	D_EPaStartAdrs_H
0x186	
0x187	
0x188	D_EPbStartAdrs_L
0x189	D_EPbStartAdrs_H
0x18A	
0x18B	
0x18C	D_EPcStartAdrs_L
0x18D	D_EPcStartAdrs_H
0x18E	D_EPcEndAdrs_L
0x18F	D_EPcEndAdrs_H

0x190~0x1DF:Reserved

0x1E0	(Reserved)
0x1E1	D_ModeControl

0x1E2~0x1FF:Reserved

Appendix B リトルエンディアンの CPU への接続

ホスト・レジスタ

Big Endian

0xE0	H_SIE_IntStat_0
0xE1	H_SIE_IntStat_1
0xE2	H_FIFO_IntStat
0xE3	H_FrameIntStat
0xE4	H_CHrIntStat
0xE5	H_CH0IntStat
0xE6	H_CHaIntStat
0xE7	H_CHbIntStat
0xE8	H_CHcIntStat
0xE9	H_CHdIntStat
0xEA	H_CHeIntStat
0xEB	
0xEC	
0xED	
0xEE	
0xEF	

0xF0	H_SIE_IntEnb_0
0xF1	H_SIE_IntEnb_1
0xF2	H_FIFO_IntEnb
0xF3	H_FrameIntEnb
0xF4	H_CHrIntEnb
0xF5	H_CH0IntEnb
0xF6	H_CHaIntEnb
0xF7	H_CHbIntEnb
0xF8	H_CHcIntEnb
0xF9	H_CHdIntEnb
0xFA	H_CHeIntEnb
0xFB	
0xFC	
0xFD	
0xFE	
0xFF	

0x100	H_Reset
0x101	
0x102	H_NegoControl_0
0x103	
0x104	H_NegoControl_1
0x105	
0x106	H_USB_Test
0x107	
0x108	H_CHnControl
0x109	H_CHrFIFO_Clr
0x10A	H_ClrAllCHnJoin
0x10B	
0x10C	
0x10D	
0x10E	
0x10F	

0x110	H_CH0SETUP_0
0x111	H_CH0SETUP_1
0x112	H_CH0SETUP_2
0x113	H_CH0SETUP_3
0x114	H_CH0SETUP_4
0x115	H_CH0SETUP_5
0x116	H_CH0SETUP_6
0x117	H_CH0SETUP_7
0x118	
0x119	
0x11A	
0x11B	
0x11C	
0x11D	
0x11E	H_FrameNumber_H
0x11F	H_FrameNumber_L

ホスト・レジスタ

Little Endian

Little Endian時に上位、下位が変わるレジスタ

0xE0	H_SIE_IntStat_0
0xE1	H_SIE_IntStat_1
0xE2	H_FIFO_IntStat
0xE3	H_FrameIntStat
0xE4	H_CHrIntStat
0xE5	H_CH0IntStat
0xE6	H_CHaIntStat
0xE7	H_CHbIntStat
0xE8	H_CHcIntStat
0xE9	H_CHdIntStat
0xEA	H_CHeIntStat
0xEB	
0xEC	
0xED	
0xEE	
0xEF	

0xF0	H_SIE_IntEnb_0
0xF1	H_SIE_IntEnb_1
0xF2	H_FIFO_IntEnb
0xF3	H_FrameIntEnb
0xF4	H_CHrIntEnb
0xF5	H_CH0IntEnb
0xF6	H_CHaIntEnb
0xF7	H_CHbIntEnb
0xF8	H_CHcIntEnb
0xF9	H_CHdIntEnb
0xFA	H_CHeIntEnb
0xFB	
0xFC	
0xFD	
0xFE	
0xFF	

0x100	H_Reset
0x101	
0x102	H_NegoControl_0
0x103	
0x104	H_NegoControl_1
0x105	
0x106	H_USB_Test
0x107	
0x108	H_CHnControl
0x109	H_CHrFIFO_Clr
0x10A	H_ClrAllCHnJoin
0x10B	
0x10C	
0x10D	
0x10E	
0x10F	

0x110	H_CH0SETUP_0
0x111	H_CH0SETUP_1
0x112	H_CH0SETUP_2
0x113	H_CH0SETUP_3
0x114	H_CH0SETUP_4
0x115	H_CH0SETUP_5
0x116	H_CH0SETUP_6
0x117	H_CH0SETUP_7
0x118	
0x119	
0x11A	
0x11B	
0x11C	
0x11D	
0x11E	H_FrameNumber_L
0x11F	H_FrameNumber_H

Appendix B リトルエンディアンの CPU への接続

ホスト・レジスタ

Big Endian

0x120	H_CH0Config_0
0x121	H_CH0Config_1
0x122	
0x123	H_CH0MaxPktSize
0x124	
0x125	
0x126	H_CH0TotalSize_H
0x127	H_CH0TotalSize_L
0x128	H_CH0HubAdrs
0x129	H_CH0FuncAdrs
0x12A	
0x12B	H_CTL_SupportControl
0x12C	
0x12D	
0x12E	H_CH0ConditionCode
0x12F	H_CH0Join

0x130	H_CHaConfig_0
0x131	H_CHaConfig_1
0x132	H_CHaMaxPktSize_H
0x133	H_CHaMaxPktSize_L
0x134	H_CHaTotalSize_HH
0x135	H_CHaTotalSize_HL
0x136	H_CHaTotalSize_LH
0x137	H_CHaTotalSize_LL
0x138	H_CHaHubAdrs
0x139	H_CHaFuncAdrs
0x13A	H_BO_SupportControl
0x13B	H_CSW_RcvDataSize
0x13C	H_OUT_EP_Control
0x13D	H_IN_EP_Control
0x13E	H_CHaConditionCode
0x13F	H_CHaJoin

0x140	H_CHbConfig_0
0x141	H_CHbConfig_1
0x142	H_CHbMaxPktSize_H
0x143	H_CHbMaxPktSize_L
0x144	H_CHbTotalSize_HH
0x145	H_CHbTotalSize_HL
0x146	H_CHbTotalSize_LH
0x147	H_CHbTotalSize_LL
0x148	H_CHbHubAdrs
0x149	H_CHbFuncAdrs
0x14A	H_CHbInterval_H
0x14B	H_CHbInterval_L
0x14C	
0x14D	
0x14E	H_CHbConditionCode
0x14F	H_CHbJoin

0x150	H_CHcConfig_0
0x151	H_CHcConfig_1
0x152	H_CHcMaxPktSize_H
0x153	H_CHcMaxPktSize_L
0x154	H_CHcTotalSize_HH
0x155	H_CHcTotalSize_HL
0x156	H_CHcTotalSize_LH
0x157	H_CHcTotalSize_LL
0x158	H_CHcHubAdrs
0x159	H_CHcFuncAdrs
0x15A	H_CHcInterval_H
0x15B	H_CHcInterval_L
0x15C	
0x15D	
0x15E	H_CHcConditionCode
0x15F	H_CHcJoin

ホスト・レジスタ

Little Endian

Little Endian時に上位、下位が変わるレジスタ

0x120	H_CH0Config_0
0x121	H_CH0Config_1
0x122	H_CH0MaxPktSize
0x123	
0x124	
0x125	
0x126	H_CH0TotalSize_L
0x127	H_CH0TotalSize_H
0x128	H_CH0HubAdrs
0x129	H_CH0FuncAdrs
0x12A	
0x12B	H_CTL_SupportControl
0x12C	
0x12D	
0x12E	H_CH0ConditionCode
0x12F	H_CH0Join

0x130	H_CHaConfig_0
0x131	H_CHaConfig_1
0x132	H_CHaMaxPktSize_L
0x133	H_CHaMaxPktSize_H
0x134	H_CHaTotalSize_HL
0x135	H_CHaTotalSize_HH
0x136	H_CHaTotalSize_LL
0x137	H_CHaTotalSize_LH
0x138	H_CHaHubAdrs
0x139	H_CHaFuncAdrs
0x13A	H_BO_SupportControl
0x13B	H_CSW_RcvDataSize
0x13C	H_OUT_EP_Control
0x13D	H_IN_EP_Control
0x13E	H_CHaConditionCode
0x13F	H_CHaJoin

0x140	H_CHbConfig_0
0x141	H_CHbConfig_1
0x142	H_CHbMaxPktSize_L
0x143	H_CHbMaxPktSize_H
0x144	H_CHbTotalSize_HL
0x145	H_CHbTotalSize_HH
0x146	H_CHbTotalSize_LL
0x147	H_CHbTotalSize_LH
0x148	H_CHbHubAdrs
0x149	H_CHbFuncAdrs
0x14A	H_CHbInterval_L
0x14B	H_CHbInterval_H
0x14C	
0x14D	
0x14E	H_CHbConditionCode
0x14F	H_CHbJoin

0x150	H_CHcConfig_0
0x151	H_CHcConfig_1
0x152	H_CHcMaxPktSize_L
0x153	H_CHcMaxPktSize_H
0x154	H_CHcTotalSize_HL
0x155	H_CHcTotalSize_HH
0x156	H_CHcTotalSize_LL
0x157	H_CHcTotalSize_LH
0x158	H_CHcHubAdrs
0x159	H_CHcFuncAdrs
0x15A	H_CHcInterval_L
0x15B	H_CHcInterval_H
0x15C	
0x15D	
0x15E	H_CHcConditionCode
0x15F	H_CHcJoin

Appendix B リトルエンディアンの CPU への接続

ホスト・レジスタ

Big Endian

0x160	H_CHdConfig_0
0x161	H_CHdConfig_1
0x162	H_CHdMaxPktSize_H
0x163	H_CHdMaxPktSize_L
0x164	H_CHdTotalSize_HH
0x165	H_CHdTotalSize_HL
0x166	H_CHdTotalSize_LH
0x167	H_CHdTotalSize_LL
0x168	H_CHdHubAdrs
0x169	H_CHdFuncAdrs
0x16A	H_CHdInterval_H
0x16B	H_CHdInterval_L
0x16C	
0x16D	
0x16E	H_CHdConditionCode
0x16F	H_CHdJoin

0x170	H_CHeConfig_0
0x171	H_CHeConfig_1
0x172	H_CHeMaxPktSize_H
0x173	H_CHeMaxPktSize_L
0x174	H_CHeTotalSize_HH
0x175	H_CHeTotalSize_HL
0x176	H_CHeTotalSize_LH
0x177	H_CHeTotalSize_LL
0x178	H_CHeHubAdrs
0x179	H_CHeFuncAdrs
0x17A	H_CHeInterval_H
0x17B	H_CHeInterval_L
0x17C	
0x17D	
0x17E	H_CHeConditionCode
0x17F	H_CHeJoin

0x180	H_CH0StartAdrs_H
0x181	H_CH0StartAdrs_L
0x182	H_CH0EndAdrs_H
0x183	H_CH0EndAdrs_L
0x184	H_CHaStartAdrs_H
0x185	H_CHaStartAdrs_L
0x186	H_CHaEndAdrs_H
0x187	H_CHaEndAdrs_L
0x188	H_CHbStartAdrs_H
0x189	H_CHbStartAdrs_L
0x18A	H_CHbEndAdrs_H
0x18B	H_CHbEndAdrs_L
0x18C	H_CHcStartAdrs_H
0x18D	H_CHcStartAdrs_L
0x18E	H_CHcEndAdrs_H
0x18F	H_CHcEndAdrs_L

0x190	H_CHdStartAdrs_H
0x191	H_CHdStartAdrs_L
0x192	H_CHdEndAdrs_H
0x193	H_CHdEndAdrs_L
0x194	H_CHeStartAdrs_H
0x195	H_CHeStartAdrs_L
0x196	H_CHeEndAdrs_H
0x197	H_CHeEndAdrs_L

0x198~0x1F3:Reserved

0x1F4	(Reserved)
0x1F5	H_Protect
0x1F6	H_Monitor
0x1F7	(Reserved)

0x1A0~0x1FF:Reserved

ホスト・レジスタ

Little Endian

Little Endian時に上位、下位が変わるレジスタ

0x160	H_CHdConfig_0
0x161	H_CHdConfig_1
0x162	H_CHdMaxPktSize_L
0x163	H_CHdMaxPktSize_H
0x164	H_CHdTotalSize_HL
0x165	H_CHdTotalSize_HH
0x166	H_CHdTotalSize_LL
0x167	H_CHdTotalSize_LH
0x168	H_CHdHubAdrs
0x169	H_CHdFuncAdrs
0x16A	H_CHdInterval_L
0x16B	H_CHdInterval_H
0x16C	
0x16D	
0x16E	H_CHdConditionCode
0x16F	H_CHdJoin

0x170	H_CHeConfig_0
0x171	H_CHeConfig_1
0x172	H_CHeMaxPktSize_L
0x173	H_CHeMaxPktSize_H
0x174	H_CHeTotalSize_HL
0x175	H_CHeTotalSize_HH
0x176	H_CHeTotalSize_LL
0x177	H_CHeTotalSize_LH
0x178	H_CHeHubAdrs
0x179	H_CHeFuncAdrs
0x17A	H_CHeInterval_L
0x17B	H_CHeInterval_H
0x17C	
0x17D	
0x17E	H_CHeConditionCode
0x17F	H_CHeJoin

0x180	H_CH0StartAdrs_L
0x181	H_CH0StartAdrs_H
0x182	H_CH0EndAdrs_L
0x183	H_CH0EndAdrs_H
0x184	H_CHaStartAdrs_L
0x185	H_CHaStartAdrs_H
0x186	H_CHaEndAdrs_L
0x187	H_CHaEndAdrs_H
0x188	H_CHbStartAdrs_L
0x189	H_CHbStartAdrs_H
0x18A	H_CHbEndAdrs_L
0x18B	H_CHbEndAdrs_H
0x18C	H_CHcStartAdrs_L
0x18D	H_CHcStartAdrs_H
0x18E	H_CHcEndAdrs_L
0x18F	H_CHcEndAdrs_H

0x190	H_CHdStartAdrs_L
0x191	H_CHdStartAdrs_H
0x192	H_CHdEndAdrs_L
0x193	H_CHdEndAdrs_H
0x194	H_CHeStartAdrs_L
0x195	H_CHeStartAdrs_H
0x196	H_CHeEndAdrs_L
0x197	H_CHeEndAdrs_H

0x198~0x1F3:Reserved

0x1F4	(Reserved)
0x1F5	H_Protect
0x1F6	H_Monitor
0x1F7	(Reserved)

0x1A0~0x1FF:Reserved

Appendix C 1ポートモード

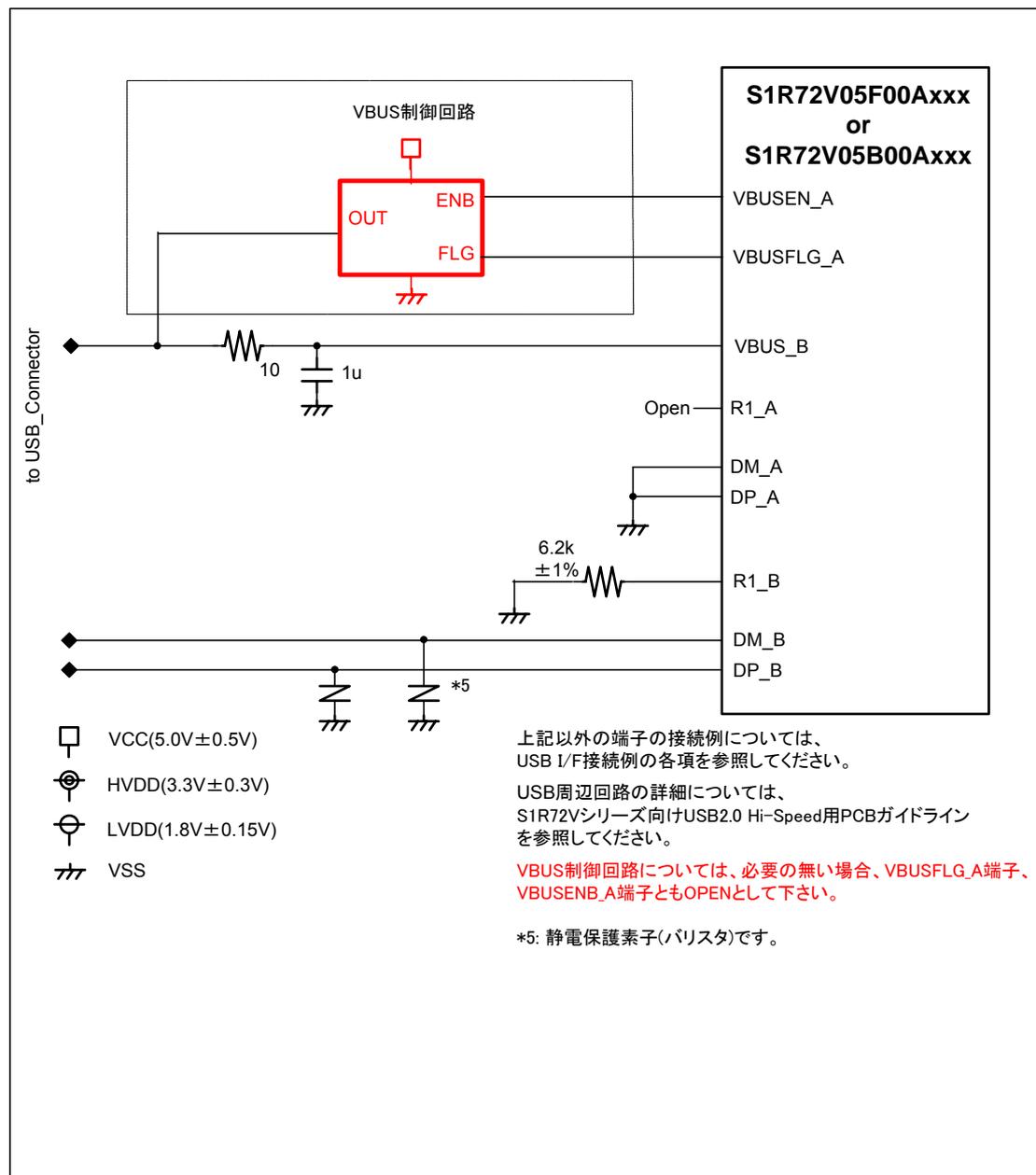
S1R72V05は、初期設定時にClkSelect.PORT1x2ビットを設定することにより、2ポートモード(ClkSelect.PORT1x2 = “0”)と1ポートモード(ClkSelect.PORT1x2 = “1”)を選択できます。2ポートモードでは、Aポートにてホスト機能を、Bポートにてデバイス機能を実現しています。対して、1ポートモードでは、Bポートのみを使用してホスト/デバイスの両機能を実現しています。

なお、1ポートモードを選択する場合には、空きポートであるAポートを下図を参考に処理してください。また、ホスト機能時のVBUS制御信号(VBUSEN_A、VBUSFLG_A)は、末尾の_Aに関わらず、VBUS制御回路にて使用してください。

さらに、1ポートモードで使用する場合には、パワーマネジメント機能のうち、ACT_HOSTステータスを使用しないよう、PM_Control_0.GoActHostビットを絶対にセットしないでください。したがって、1ポートモードを使用している場合において、ホスト機能、デバイス機能のどちらを実行する場合においても、パワーマネジメント機能としては、ACT_DEVICEステータスで使用してください。

また、D_Reset.ResetDTMおよびH_Reset.ResetHTMは、両方のビットともに“0”を設定してください。

<接続例>



Appendix D HOST High-Speed 動作時の SUSPEND について

S1R72V05 の USB HOST Port の、動作モードが HS である場合において、USB バスが SUSPEND 状態にある期間中に、ディスコネクトの検出機能が使用できません。この場合、ケーブルがディスコネクトされても、H_SIE_IntStat.DetectDiscon 割り込みステータスが発行されません。

※USB HOST Port には、下記が該当します。

ClkSelect.PORT1x2=="0"の場合の USB Port A

ClkSelect.PORT1x2=="1"かつ HostDeviceSel.HOSTxDEVICE=="1"の場合の USB Port B

※動作モードが HS である場合とは、下記の場合です。

H_NegoControl_1.PortSpeed=="HS(0b00)"

本レジスタは、オートネゴシエーション機能の実行結果として、H/W によって自動的にセットされます。

※USB バスが SUSPEND 状態にある期間とは、下記の場合です。

H_NegoControl_0.HostState=="USB_SUSPEND(0b110)"

本レジスタは、オートネゴシエーション機能の実行結果として、H/W によって自動的にセットされます。

下記、いずれかの措置をお願いします。

1. SUSPEND 状態を使用しない。

通常、SUSPEND を使用しないことは、バスをアクティブにしつづけることを意味します。しかしながら、Embedded ホストでは、SUSPEND 状態でデバイスとの接続を維持するよりも、ソフト的に強制的にディスコネクト状態にすることにより、システムの省電力化及び制御の簡易化を図ることができると考えます。なお、S1R72V05 では、ソフト的にディスコネクトした状態で、LSI を SNOOZE あるいは SLEEP 等のパワーセーブモードにすることが出来ます。

ホストがソフト的に強制的にディスコネクト状態にするには、VBUS への電源供給を停止することによって行います。H_NegoControl_0.HostState="IDLE"に遷移させて下さい。そうすることによって、VBUSEN 端子の出力をネゲートします。

IDLE に移行する手順は下記ようになります。

- ・H_NegoControl_0.AutoModeCancel="1"とセットする。
H_NegoControl_0="0x80"と書き込んで下さい。
- ・H_NegoControl_0.AutoModeCancel="0"となるまで待つ。
この変化には、最大で 100ns かかることが有ります。
- ・H_NegoControl_0.AutoMode="GoldLE(0b0001)"を設定する。

2. f/w により、強制的に FS 動作モードに設定してから SUSPEND 状態に移行する。

USB バスを SUSPEND 状態に移行する前に、あらかじめ H_NegoControl_1.PortSpeed の値を FS に書き換えることにより、SUSPEND 状態にある期間中において、ディスコネクトの検出機能を使用できます。

ただし、この方法を採用する場合、USB デバイスからの Remote Wakeup に対応するためには、f/w の処理に時間制限が御座います。Remote Wakeup を検出すると、h/w が RESUME 信号を送出します。その処理を行っている間に、PortSpeed を"HS(0b00)"に書き戻す必要が有ります。

SUSPEND に移行する手順は下記ようになります。

- ・全てのトランザクションの発行を終了する。
新たにトランザクションを発行しないで下さい。
- ・H_Protect.TranEnb="STOP(0b01)"とセットする。
SOF トランザクションの発行を停止します。H_Protect=0x01 と書き込んで下さい。
- ・H_Monitor.TranRunning="0"となるまで待つ。
SOF トランザクションの発行が停止されることを待ちます。この変化には、最大で 1us かかることが有ります。
- ・H_Protect.PortSpeedWrEnb="1"とセットする。
PortSpeed を書き換えられるようにします。他のビットを書き換えないよう、リードモディファイライトをして下さい。
- ・H_NegoControl_1.PortSpeed="FS(0b01)"を設定する。

Appendix D HOST High-Speed 動作時の SUSPEND について

- ・H_Protect.PortSpeedWrEnb="0"とクリアする。
他のビットを書き換ええないよう、リードモディファイライトをして下さい。
- ・H_NegoControl_0.AutoMode="GoSUSPENDtoOP(0b1110)"を設定する。

切断検出は下記のように行われます。

- ・デバイス切断検出割り込み(H_SIE_IntStat.DetectDiscon)が発生する。
- ・H_NegoControl_0.AutoModeCancel="1"とセットする。
GoSUSPENDtoOP の設定を解除します。H_NegoControl_0="0x80"と書き込んで下さい。
- ・H_NegoControl_0.AutoModeCancel="0"となるまで待つ。
この変化には、最大で 100ns かかることがあります。
- ~~・H_NegoControl_0.AutoModeCancel="0"となるまで待つ。
この変化には、最大で 100ns かかることがあります。~~
- ・H_NegoControl_0.AutoMode="GoWAIT_CONNECTtoDIS(0b1001)"を設定する。
接続検出待ちになります。

RESUME を行う手順は下記のようになります。

- ・H_Protect.PortSpeedWrEnb="1"とセットする。
リードモディファイライトして下さい。
- ・H_NegoControl_1.PortSpeed に動作スピードを設定する
- ・H_Protect.PortSpeedWrEnb="0"とクリアする。
リードモディファイライトして下さい。
- ・H_NegoControl_0.AutoModeCancel="1"とセットする。
GoSUSPENDtoOP の設定を解除します。H_NegoControl_0="0x80"と書き込んで下さい。
- ・H_NegoControl_0.AutoModeCancel="0"となるまで待つ。
この変化には、最大で 100ns かかることがあります。
- ・H_NegoControl_0.AutoMode="GoRESUMEtoOP(0b1111)"を設定する。

Remote Wakeup への応答は下記のように行います。

- ・Remote Wakeup 検出割り込み(H_SIE_IntStat_0.DetectRmtWkup)が発生する。
これにより、Remote Wakeup を検出します。
- ・H_Protect.PortSpeedWrEnb="1"とセットする。
リードモディファイライトして下さい。
- ・H_NegoControl_1.PortSpeed に動作スピードを設定する。
- ・H_Protect.PortSpeedWrEnb="0"とクリアする。
リードモディファイライトして下さい。

Remote Wakeup への応答時には、Remote Wakeup を検出してから 20ms 以内に上記処理を行って下さい。

RESUME 中に RESET を行う手順は下記のようになります。

- ・H_NegoControl_0.AutoModeCancel="1"とセットする。
GoRESUMEtoOP の設定を解除します。H_NegoControl_0="0x80"と書き込んで下さい。
- ・H_NegoControl_0.AutoModeCancel="0"となるまで待つ。
この変化には、最大で 100ns かかることがあります。
- ・H_NegoControl_0.AutoMode="GoRESETtoOP(0b1100)"を設定する。
従来の手順から変更はありません。

3. f/w により、LineState をモニタする。

H_USB_Status.LineState をポーリングすることにより、切断を検出することが出来ます。

SUSPEND 中、接続していれば、下記がモニタできます。

・H_USB_Status.LineState=="J(0b01)"

SUPEND 中に、切断すると、下記がモニタできます。

・H_USB_Status.LineState=="SE0(0b00)"

尚、上記手順に含まれるレジスタのうち、下記のことを、開発仕様書の Rev.1.10 にて追加定義しました。

Appendix D HOST High-Speed 動作時の SUSPEND について

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	1F5h	H_Protect		7:	0:	1:	00h
				6:	0:	1:	
				5:	0:	1:	
				4:	0:	1:	
			R / W	3: PortSpeedWrEnb	Enable to replace PortSpeed		
				2:	0:	1:	
			R / W	1: TranEnb[1]	Transaction Control		
R / W	0: TranEnb[0]						

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Host	1F6h	H_Monitor		7:			00h
				6:			
				5:			
				4:			
				3:			
				2:			
				1:			
			R	0: TranRunning	Monitor transaction		

1F5h.Bit7-4

Reserved

1F5h.Bit3

PortSpeedWrEnb

H_NegoControl_1.PortSpeed への書き込みを許可します。

1F5h.Bit2

Reserved

1F5h.Bit1-0

TranEnb

SOF の送出を停止させます。H_NegoControl_1.PortSpeed への書き込みを行う前に設定します。

1F6h.Bit7-1

Reserved

1F6h.Bit0

TranRunning

SOF の送出停止をモニタします。

Appendix E SetAddress リクエストの応答について

bmRequestType が 0(標準リクエスト)以外であり、bRequest が 0x05 であるリクエストを受信した場合に、RcvEP0SETUP 割り込みステータスが発行されません。

この問題は、“自動アドレス設定機能”が、SetAddress リクエスト(bmRequestType==0, bRequest==0x05)を自動処理することに伴い、bRequest 値を以って RcvEP0SETUP 割り込みステータスをマスクしていることに起因しています。

この問題に対処するため、下記いずれかの措置をお願いします。

1. ベンダ及びクラスリクエストを制限する

bRequest==0x05 となるベンダリクエスト、またはクラスリクエストが使用されない場合、特別な対処は必要ありません。

2. 自動アドレス設定機能を無効にする

自動アドレス設定機能を無効にすることによって、本問題を対策することが出来ます。この場合、SetAddress リクエストを受信して、自動的にステータスステージを実行する機能が無効になり、他のリクエストと同様に f/w でステータスステージを実行する必要があります。但し、自動アドレス設定機能の一部の機能を使用し、D_USB_Address レジスタの設定を自動化することが出来ます。

以下に、自動アドレス設定機能を無効にする設定と、同機能が無効である場合の制御シーケンスを説明します。また、比較のため、自動アドレス設定機能が有効な場合の制御を説明します。

<自動アドレス設定機能を無効にする処理>

イベント／処理	自動アドレス設定機能＝有効	自動アドレス設定機能＝無効
①自動アドレス設定機能を無効にする	—	f/w が D_ModeControl.SetAddressMode="1"とセットする。

①自動アドレス設定機能を無効にする

D_ModeControl.SetAddressMode="1"と設定する。

この設定処理は、リセット解除後に一度行うことで、それ以降は設定する必要はありません。

<SetAddress リクエスト処理>

イベント／処理	自動アドレス設定機能＝有効	自動アドレス設定機能＝無効
①SetAddress リクエストを受信	—	h/w が RcvEP0SETUP 割り込みステータスを発行する
②リクエストの確認	—	f/w が EP0SETUP0, EP0SETUP1 により確認する
③アドレス設定指示	—	f/w が D_USB_Address.SetAddress="1"とセットする
④ステータスステージ応答準備	—	f/w が下記設定を行う D_SETUP_Control.ProtectEP0="0" D_EP0Control.INxOUT="1" D_EP0Control.IN="0x40"※ ※ForceNAK="0", EnShortPkt="1"
⑤ステータスステージが行われる	h/w が SetAddressCmp 割り込みステータスを発行する	h/w が SetAddressCmp 割り込みステータスを発行する

①SetAddress リクエストを受信

h/w がリクエストを受信し、RcvEP0SETUP 割り込みステータスを発行します。

自動アドレス設定機能を無効化したことにより、SetAddress リクエストにおいても、他のリクエストと同様に、本ステータスによってSETUPトランザクションの受領を通知します。

②リクエストの確認

f/w が D_EP0SETUP0,1(※)レジスタの内容により、bmRequestType, bRequest を確認します。

Appendix E SetAddress リクエストの応答について

bmRequest==0, bRequest==0x05 であれば、SetAddress リクエストです。

※: S1R72V03 のレジスタ定義は、"RcvEP0SETUP"となります。

③アドレス設定指示

f/w が USB_Address.SetAddress="1"とセットします。

この設定により、この後にステータスステージが行われ完了すると、SetAddress リクエストで指示されたアドレスを、h/w が D_USB_Address レジスタに上書きします。また、その処理が完了したことを、SetAddressCmp 割り込みステータスで通知します。

④ステータスステージ応答準備

他のリクエストにおける、IN 方向のステータスステージと同じく、ゼロ長パケットを返信する処理をします。

- D_SETUP_Control.ProtectEP0="0"

- D_EP0Control.INxOUT="1"

- D_EP0Control.IN="0x40"(ForceNAK="0", EnShortPkt="1")

⑤ステータスステージが行われる

ステータスステージ(INトランザクション)が行われると、h/w が SetAddressCmp 割り込みステータスを発行します。

尚、上記手順に含まれるレジスタのうち、下記"D_ModeContrl"を、開発仕様書の Rev.1.20 にて追加定義しました。また、D_USB_Address レジスタの bit7 を、開発仕様書の Rev.1.20 にて追加定義しました。

Mode	Address	Register Name	R / W	Bit Symbol	Description	Reset	
Device	1E1h	D_ModeControl	W	7: (Reserved)	Don't set "1"		XXh
			W	6: (Reserved)	Don't set "1"		
			W	5: (Reserved)	Don't set "1"		
			W	4: SetAddressMode	0: Auto mode	1: Manual mode	
			W	3: (Reserved)	Don't set "1"		
			W	2: (Reserved)	Don't set "1"		
			W	1: (Reserved)	Don't set "1"		
			W	0: (Reserved)	Don't set "1"		

Bit7-5 **Reserved**

Bit4 **SetAddressMode**

 自動アドレス設定機能を無効にします。

Bit3-0 **Reserved**

改訂履歴 (Rev.0.79~Rev.0.90)

年月日	改訂内容			
	Rev.	項(旧版)	種別	内容
04/11/30	0.79	全項	新規	新規規定
05/02/28	0.80	全項	修正	レジスタ名称変更
		2	修正	USB ホスト機能のチャンネル本数削減(6 本)と転送タイプ変更(アイソクロナス非対応)
			追加	Media データ転送機能追加 パッケージタイプ記載
		3	修正	Port Selector 機能変更
			追加	Media FIFO/Media FIFO Controller 追加
		4	追加	QFP パッケージ端子配置図記載
		5	追加	Pin 番号記載 5Vトレラント注記追加
		6.3	修正	USB ホスト制御 全般 アイソクロナス転送関連項目削除
		6.3.1.1	修正	チャンネル本数削減(6 本)と転送タイプ変更(アイソクロナス転送非対応)
		6.3.1.4	修正	チャンネル使用例変更
		6.3.3	修正	トランザクションステータス変更
		6.3.4.3	修正	コントロール転送サポート機能仕様変更
		6.3.8	修正	バルクオンリーサポート機能仕様変更
		6.3.9.1	修正	ホストステート遷移図に DISABLED ステート追加
		6.3.9.1.1	修正	IDLE ステート遷移実行手順変更
		6.3.9.3.1		
		6.3.9.1.3	追加	DISABLED ステート追加
		6.3.9.3.3		
		6.3.9.1.4	追加	異常なデバイスからの Chirp を検出した場合において、動作モードを追加
		6.3.9.3.4.2		
		6.3.9.3.11	修正	GoSUSPENDtoOP のときの RmtWkupDetEnb ビットの挙動および処理を明記
		6.3.9.3.2.1	修正	PortSpeed 変化点誤記訂正
		6.3.9.3.2.2		
		6.3.9.3.8	追加	GoWAIT_CONNECTtoDIS 追加
		6.4	追加	メディアデータ転送機能追加
		6.6.1.1	修正	デバイス FIFO 管理 EPa-c の FIFO 領域設定を可変に変更
		6.6.2.1	修正	ホスト FIFO 管理 ホスト FIFO メモリマップ変更(固定領域は CBW および CSW のみ)と PingPong モード削除
6.6.3	追加	メディア FIFO 管理追加		
7	修正	レジスタ 名称および配置全面変更		
8.3	追加	DC 特性記載(一部 TBD あり)		
10.1	追加	QFP パッケージ外形寸法記載		
05/04/28	0.90	4	修正	QFP 端子配置図を修正
			追加	BGA 端子配置図を記載
		5	修正	QFP および BGA の端子名を修正 JTAG 機能未使用時の端子処理の説明を記載
		6.1	修正	説明を修正
		6.1.1		表 6-2 の項目を表 6-3 および表 6-4 に移動
6.1.2				
6.1.2.1.1				
6.1.2.1.2				

改訂履歴 (Rev.0.79~Rev.0.90)

	6.3 全項	修正	トークン種別をトランザクション種別に変更 レジスタビット PID を TID に変更
	6.3.4	修正	図 6-30 を修正
	6.3.4.3	修正	図 6-33 を修正 表 6-26 の TranRetry を TranErr に修正
	6.3.8	修正	図 6-34、図 6-35 を修正 (10)の説明の誤記を修正
	6.3.9.1	修正	状態図変更
	6.3.9.1.2 6.3.9.3.2.1 6.3.9.3.2.2	修正	WAIT_CONNECT ステートの接続検出手順を変更
	6.3.9.1.4	修正	RESET ステートに入る際のトランザクション停止方法を変更
	6.3.9.2.1	修正	T1 の Description を変更
	6.3.9.2.2 6.3.9.2.2.2	修正	WAIT_CONNECT ステートを切断検出対象ステータスから除外
	6.3.9.2.3.1 6.3.9.2.3.2 6.3.9.2.3.3	修正	T4 削除
	6.3.9.3.4	修正	OPERATIONAL から RESET に遷移させる際の処理に関する記述を追加
	6.3.9.3.4.2.2	修正	ResetCmp ステータス発行前に DISABLED ステートに遷移させる方法の記述を追加
	6.3.9.3.4.3.2 6.3.9.3.4.3.3	修正	表の Value 誤記を修正
	6.3.9.3.6.1 6.3.9.3.6.2 6.3.9.3.6.3	修正	表の Value 誤記を修正
	6.3.9.3.7.1 6.3.9.3.7.2 6.3.9.3.7.3	修正	表の Value 誤記を修正
	6.3.9.2.5	挿入	ポートエラー検出の項を追記
	6.3.9.1.1 6.3.9.2.1 6.3.9.3.1 6.3.9.3.4.2.2	修正	AutoModeCance による停止処理にかかる時間の記述を修正
	6.8.4.1 6.8.4.2	修正	IDE_CRC_L/H レジスタに初期値を書くと言う記述を削除
	6.9.1	追加	CLAMP コード値 0011 を記載 HIGHZ コード値 0100 を記載
	6.9.2	修正	PartNumber 値を 000E から 000F に修正
	6.9.3	修正	バウンダリスキャン除外端子の DP、DM、VBUS を DP_A、DM_A、DP_B、DM_B、VBUS_B に修正
	7	修正	説明記述を修正
	7.1 7.2	修正	7.1. デバイス・レジスタマップ、7.2. ホストレ・ジスタマップを 7.1. デバイス/ホスト共通レジスタマップ、7.2. デバイス・レジスタマップ、7.3. ホストレ・ジスタマップに項目変更 レジスタ名およびレジスタビット名の修正 レジスタおよびレジスタビットの追加、削除、移動に伴う修正
	7.3.2 7.3.3 7.3.9 7.3.10	修正	01h, 02h, 11h, 12h の各レジスタにおいて、 Device モード, Host モードを Device/Host モードに統一
	7.3.3	修正	02h レジスタの LineStatusChanged において、使用上の注意点を追加
	7.3.6	追加	05h レジスタに MedialDE_Cmp 追加
	7.3.13	追加	15h レジスタに EnMedialDE_Cmp 追加

		7.3.18	修正	23hレジスタにおいて PM_State[3:0]参照に関する注意事項を追加
		7.3.22	追加	27hレジスタの Bit6 に RemoveRPD を追加
			修正	27hレジスタの Reset 値を 11h から 91h に修正
		7.3.23	修正	28hレジスタの FSxHS において ケーブルアタッチ時の記述を追加 リードできるステートおよびライトできるステートの記述を追加
		7.3.30~ 7.3.33 7.3.36 7.3.37	修正	34h~37h, 40h, 41hレジスタのビット名誤記を修正
		7.3.34	修正	38hレジスタのレジスタ名誤記を修正
		7.3.39	修正	43hレジスタのビット説明において、設定数の制限を追記
		7.3.45 7.3.47	修正	4Bh より 49h に CtrAllMediaFIFO_Join レジスタを移動し、4Bh を Reserved に変更
		7.3.49~ 7.3.80	修正	50h~6Fhレジスタのビット名誤記を修正
		7.3.82 7.3.97	修正	71h および 81hレジスタの ActiveDMA の論理と説明を修正
		7.3.114	修正	93hレジスタにおいて Bit2 の SWAP を削除
		7.3.141	修正	B3h レジスタにおいて、ModeProtect で 保護されるレジスタとして ClkSelect.ClkSelect を追加
		7.3.143	修正	B5hレジスタの xActIDE_Term を非同期レジスタに修正し、説明を修正
		7.3.145	修正	B7hレジスタの CPU_Swap ビットの説明を修正
		7.4.1	修正	ACT_DEVICE を出る際の割り込みステータスの処理説明
		7.4.11	修正	ACT_DEVICE を出る際の割り込みステータスの処理説明
		7.4.12	修正	100hレジスタの ResetDTM の論理変更(1: ResetDTM)
		7.4.5 7.4.15	修正	E4h および F4hレジスタのビット名を修正
		7.4.46 7.4.47	修正	11Eh および 11Fhレジスタの Reset 値を修正
		7.4.55	修正	130hレジスタの Reset 値を修正
		7.5.1	修正	E0hレジスタのレジスタ名を修正 ACT_HOST を出る際の割り込みステータスの処理説明
		7.5.13	修正	F0hレジスタの ACT_HOST を出る際の割り込みステータスの処理説明
		7.5.2	修正	E1hレジスタのレジスタ名を修正 ACT_HOST を出る際の割り込みステータスの処理説明
			追加	Bit3 に DisabledCmp を追加
		7.5.14	修正	F1hレジスタのレジスタ名を修正 Bit0 EnWaitConCmp を削除 EnResetCmp を Bit1 から Bit0 に移動 EnSuspendCmp を Bit2 から Bit1 に移動 EnResumeCmp を Bit3 から Bit2 に移動 ACT_HOST を出る際の割り込みステータスの処理説明
			追加	Bit3 に EnDisabledCmp を追加
		7.5.5	修正	E4hレジスタのビット名を修正
		7.5.6	修正	E5hレジスタにおいて TotalSizeCmp の説明を修正 TranAck の説明を修正 ChangeCondition の ConditionCode 表中説明を修正 CTL_SupportCmp の説明を修正 CTL_SupportStop の説明を修正
		7.5.7~ 7.5.11	修正	E6h~EAhレジスタの Bit7 のビット名を TranCmp から TotalSizeCmp に修正

改訂履歴 (Rev.0.79~Rev.0.90)

		7.5.7	修正	E6h レジスタにおいて TotalSizeCmp の説明を修正 TranAck の説明を修正 BO_SupportCmp の説明を修正 BO_SupportStop の説明を修正
			追加	ChangeCondition の説明に ConditionCode 表を追加
		7.5.8~ 7.5.11	追加	E7h~EAh レジスタにおいて ChangeCondition の説明に ConditionCode 表を追加
		7.5.17	修正	F4h レジスタのビット名を修正
		7.5.19~ 7.5.23	修正	F6h~EAh レジスタの Bit7 のビット名を EnTranCmp から EnTotalSizeCmp に修正
		7.5.25	修正	100h レジスタの ResetDTM を ResetHTM に誤記修正 ResetHTM の論理変更(1: ResetHTM) デバイストランシーバマクロをホストランシーバマクロに誤記修正
		7.5.27	修正	102h レジスタにおいて レジスタ名を修正 AutoModeCancel の説明を修正 AutoMode の説明を修正 Reset 値を修正
		7.5.28	修正	103h レジスタにおいて レジスタ名を修正 DisChirpFinish ビットの説明を修正
		7.5.31	修正	106h レジスタの Bit4 のビット名を TestForceEnable から Test_Force_enable に修正
		7.5.33	修正	108h レジスタのレジスタ名を修正
		7.5.46 7.5.60	修正	120h および 130h レジスタにおいて Reset 値を修正 ACK_Cnt[3:0]の説明を修正 Toggle の説明を修正 TranGo の説明を修正
		7.5.76 7.5.91 7.5.106 7.5.121	修正	140h, 150h, 160h, 170h レジスタにおいて Reset 値を修正 ACK_Cnt ビットの説明を修正
		7.5.47 7.5.61 7.5.77 7.5.92 7.5.107 7.5.122	修正	121h, 131h, 141h, 151h, 161h, 171h レジスタにおいて Bit7-6 のビット名を PID ビットから TID ビットに修正 同ビットの説明を修正
		7.5.53 7.5.54	修正	128h および 129h レジスタにおいて TotalSize[15:0]の説明を修正
		7.5.50~ 7.5.54	修正	124h~129h においてレジスタのアドレス変更
		7.5.64~ 7.5.69	修正	134h~139h においてレジスタのアドレス変更
		7.5.80~ 7.5.84	修正	144h~149h においてレジスタのアドレス変更
		7.5.95~ 7.5.100	修正	154h~159h においてレジスタのアドレス変更
		7.5.110~ 7.5.115	修正	164h~169h においてレジスタのアドレス変更
		7.5.125~ 7.5.130	修正	174h~179h においてレジスタのアドレス変更

	7.5.56	修正	12Bhレジスタにおいて Reset 値修正 CTL_SupportState を Rレジスタから R/W レジスタに修正 CTL_SupportGo の説明を修正
	7.5.58 7.5.74 7.5.89 7.5.104 7.5.119 7.5.134	修正	12Eh, 13Eh, 14Eh, 15Eh, 16Eh, 17Eh レジスタにおいて ConditionCode[2:0]の表中説明を修正
	7.5.62 7.5.63	修正	132h, 133h レジスタにおいて MaxPktSize[9:0]の説明を修正
	7.5.65	修正	135h レジスタにおいて EP_Number の説明を修正
	7.5.66~ 7.5.69	修正	136h~139h レジスタにおいて TotalSize[31:0]の説明を修正
	7.5.70	修正	13Ah レジスタにおいて レジスタ説明を修正 BO_TransportState[1:0]の説明を修正 BO_SupportGo の説明を修正
	7.5.71	修正	13Bh レジスタにおいて レジスタ説明を修正 CSW_RcvDataSize[3:0]の説明を修正
	7.5.72	修正	13Ch レジスタにおいて レジスタ説明を修正 OUT_Toggle の説明を修正 OUT_EP_Number レジスタのビット幅および説明を修正
	7.5.73	修正	13Dh レジスタにおいて レジスタ説明を修正 IN_Toggle の説明を修正 IN_EP_Number レジスタのビット幅および説明を修正
	7.5.77 7.5.92 7.5.107 7.5.171	修正	141h, 151h, 161h, 171h レジスタにおいて ToggleMode を削除
	7.6.86	修正	14Ah レジスタのレジスタ名修正
	7.6.87	修正	14Bh レジスタのレジスタ名修正およびアドレスの誤記訂正
	8.2	修正	電源投入順を記入
	8.3	修正	入力リーク記入
	8.4.1	修正	CPU/DMA I/F アクセスタイミングに数値を記載(Preliminary)
	8.4.1 8.4.2	挿入	RESET タイミング、クロックタイミング追記 項挿入のため、以降の項番号を修正
	9.2	追加	USB I/F 接続例を 9.2.1~9.2.4 として記載(Preliminary)
	10.2	追加	BGA パッケージ外形寸法図記載

改訂履歴 (Rev.0.91～)

改訂履歴 (Rev.0.91～)

年月日	改訂内容			内容
	Rev.	項	種別	
05/12/02	0.91	3	修正	TDOの方向:入力 → 出力
		4	修正	端子名称変更:PVDD → LVDD、PVSS → VSS
		5		
		8.1		
		8.2		
		8.3		
		9.2		
		5	修正	XRESET 端子のカテゴリを GENERAL から CPU I/F に変更
			追加	XINT 端子の Tri-state に関する注記を記載
			修正	HINTRQ 端子の Termination を変更:(PU) → (PD)
			修正	HDD7 端子の Termination を変更:(PU) → (PD)
			修正	IDE I/F の IN 端子および HDD7 端子の RESET 値を記載
			追加	IDE I/F 端子の PU および PD に関する注記を記載
			修正	5Vトレラント対応端子を IDE I/F に限定
			修正	I/O の欄にあった PU、PD の記述を端子タイプの欄へ移動
		6.3～ 6.3.7	修正	全般において、CHx で指定するチャンネル(0,a～e)の誤記訂正
		6.3.4.3	修正	コントロール転送サポート機能の制御方法変更による修正
		6.3.8	修正	バルクオンリーサポート機能の制御方法変更による修正
		6.5	修正	PM_Control1 → PM_Control_1
		6.5 6.5.1 6.5.2 7.5.1	修正	PM_Control0 → PM_Control_0
		6.6.1.2.1 6.6.1.4.2 6.6.1.5.2 6.6.2.2.2 6.6.2.4.2 6.7.2.1.3 7.4.40 7.4.41	修正	RAM_WrDoor_H,L → RAM_WrDoor_0,1
		6.6.1.5.3 6.6.2.4.3 6.6.3.2.1 6.7.2.1.4 6.7.2.1.5 7.4.46 7.5.53 7.5.60 7.5.67 7.5.74 7.6.61 7.6.77 7.6.92 7.6.107 7.6.122 7.6.137	修正	FIFO_Rd_H,L → FIFO_Rd_0,1

05/12/02	0.91	6.6.1.5.3 6.6.2.4.3 6.6.3.2.1 6.7.2.1.3 7.4.46 7.5.53 7.5.60 7.5.67 7.5.74 7.6.61 7.6.77 7.6.92 7.6.107 7.6.122 7.6.137	修正	FIFO_Wr_H,L → FIFO_Wr_0,1
		6.8.1.4	修正	IDE_RdRegValue_L → IDE_RdRegValue_1
		7.1	修正	H_FIFOI_ntStat → H_FIFO_IntStat
		7.1 7.4.6	修正	MediaFIFO_IntStat レジスタのビット位置とビット名変更: bit5 Full → bit1FIFO_Full, bit4 Empty → bit0 FIFO_Empty
			追加	MediaFIFO_IntSta レジスタの FIFO_NotEmpty ビット追加
		7.1 7.4.13	修正	MediaFIFO_IntEnb レジスタのビット位置とビット名変更: bit5 EnFull → bit1 EnFIFO_Full, bit4 EnEmpty → bit0 EnFIFO_Empty
			追加	MediaFIFO_IntStat レジスタの EnFIFO_NotEmpty ビット追加
		7.1 7.4.26 7.4.27	修正	レジスタ名変更: FIFO_Rd_H → FIFO_Rd_0 FIFO_Rd_L → FIFO_Rd_1
		7.1 7.4.28 7.4.29	修正	レジスタ名変更: FIFO_Wr_H → FIFO_Wr_0 FIFO_Wr_L → FIFO_Wr_1
		7.1 7.4.42 7.4.43	修正	レジスタ名変更: RAM_WrDoor_H → RAM_WrDoor_0 RAM_WrDoor_L → RAM_WrDoor_1
		7.1 7.4.92 7.4.93	修正	レジスタ名変更: DMA0_RdData_H → DMA0_RdData_0 DMA0_RdData_L → DMA0_RdData_1
		7.1 7.4.94 7.4.95	修正	DMA0_WrData_H → DMA0_WrData_0 DMA0_WrData_L → DMA0_WrData_1
		7.1 7.4.107 7.4.108	修正	レジスタ名変更: DMA1_RdData_H → DMA1_RdData_0 DMA1_RdData_L → DMA1_RdData_1
		7.1 7.4.109 7.4.110	修正	レジスタ名変更: DMA1_WrData_H → DMA1_WrData_0 DMA1_WrData_L → DMA1_WrData_1
		7.1 7.4.127 7.4.128	修正	レジスタ名変更: IDE_RdRegValue_H → IDE_RdRegValue_0 IDE_RdRegValue_L → IDE_RdRegValue_1
		7.1 7.4.129 7.4.130	修正	レジスタ名変更: IDE_WrRegValue_H → IDE_WrRegValue_0 IDE_WrRegValue_L → IDE_WrRegValue_1
		7.1 7.4.114	追加	IDE_Config_1 レジスタの Swap ビット追加
			修正	IDE_Config_1 の Reset 値変更: 0x00 → 0x04
		7.1 7.4.143	追加	ClkSelect レジスタの xActIDE_DD_Term ビットおよび PORT1x2 ビット追加
			修正	ClkSelect レジスタの Reset 値変更: 0x01 → 0x41

改訂履歴 (Rev.0.91～)

		7.1 7.4.145	修正	ChipConfig レジスタのビット名変更: CPU_Swap → CPU_Endian
		7.2 7.5.3	追加	D_FIFO_IntStat レジスタの FIFO_NotEmpty ビット追加
		7.2 7.5.13	追加	D_FIFO_IntEnb レジスタの EnFIFO_NotEmpty ビット追加
		7.2 7.5.53	追加	D_EP0Join レジスタの JoinFIFO_Stat ビット追加
05/12/02	0.91	7.3	修正	H_NegoControl_0 レジスタの初期値: 0x00→0x1X
		7.3 7.6.3	追加	H_FIFO_IntStat レジスタの FIFO_NotEmpty ビット追加
		7.3 7.6.15	追加	H_FIFO_IntEnb レジスタの EnFIFO_NotEmpty ビット追加
		7.3 7.6.28	修正	H_NegoControl_1 レジスタのアドレス変更により 0x103 を Reserved に変更
		7.3 7.6.29	修正	H_NegoControl_1 レジスタのアドレス変更: 0x103→0x104
		7.3 7.6.29 7.6.46 7.6.47	追加	H_NegoControl_1, H_FrameNumber_H, H_FrameNumber_L レジスタの Reset 値に関する注記を追加
		7.3 7.6.35	修正	H_ClrAllChnJoin レジスタのアドレス変更: 0x104→0x10A
		7.3 7.6.61	追加	H_CH0Join レジスタの JoinFIFO_Stat ビット追加
		7.4.1	修正	SIE Interrupts → Device Interrupts
			修正	EPr Interrupts → Host Interrupts
			修正	Device_IntStat → DeviceIntStat
			修正	PM_Control レジスタ → PM_Control_0 レジスタ
		7.4.1 7.4.3	修正	Host_IntStat → HostIntStat
		7.4.125	修正	IDE_RdRegValue_H,L → IDE_RdRegValue_0,1 IDE_WrRegValue_H,L → IDE_WrRegValue_0,1
		7.5.53	修正	Bit5 JoinCPU_Rd → Bit1 JoinCPU_Rd、Bit4 JoinCPU_Wr → Bit0 JoinCPU_Wr
		7.5.60	修正	EPnChnFIFO_Wr_H,L → FIFO_Wr_0,1
		7.5.60 7.5.67	修正	EPnChnFIFO_ByteRd → FIFO_ByteRd
		7.5.60 7.5.67 7.5.74	追加	JoinFIFO_Stat ビットの説明に NotEmpty を追加。
			修正	Bit5 JoinCPU_Rd → Bit1 JoinCPU_Rd、Bit4 JoinCPU_Wr → Bit0 JoinCPU_Wr
		7.6.3	修正	FIFO_Full、FIFO_Empty ビットの説明に CH0 追加。
		7.6.4	修正	Bit2 FrameNumOver → Bit1 FrameNumOver
		7.6.15	追加	EnFIFO_NotEmpty ビット
		7.6.36	追加	Reserved(10Bh~10Fh)項を追加
		7.6.46 7.6.47	修正	Reset 値: 0800h → 07FFh
		7.6.48 7.6.62 7.6.93 7.6.108 7.6.123	修正	4:ACK_Cnt[1] → 4:ACK_Cnt[0]

		7.6.55 7.6.70 7.6.86 7.6.101 7.6.116 7.6.131	修正	Bit7 HubAdrs[3:0] → Bit7-4 HubAdrs[3:0]
		7.6.56 7.6.71	修正	6:FuncAdrs[3], 5:FuncAdrs[3], 4:FuncAdrs[3] → 6:FuncAdrs[2], 5:FuncAdrs[1], 4:FuncAdrs[0]
		7.6.87 7.6.102 7.6.117 7.6.132	修正	Bit5 EP_Number[3:0] → Bit3-0 EP_Number[3:0]
		7.6.75 7.6.91 7.6.106 7.6.121 7.6.136	修正	Bit7-4 ConditionCode[2:0] → Bit7 Reserved Bit6-4 ConditionCode[2:0]
05/12/02	0.91	7.6.77 7.6.92 7.6.107 7.6.122 7.6.137	追加	JoinFIFO_Stat ビットの説明に NotEmpty を追加。
		7.6.143	修正	185h.Bit2-0 → 185h.Bit1-0
		7.6.149	修正	18Bh.Bit2-0 → 18Bh.Bit1-0
		7.6.153	修正	18Fh.Bit2-0 → 18Fh.Bit1-0
		7.6.157	修正	193h.Bit2-0 → 193h.Bit1-0
		7.6.159	修正	195h.Bit2-0 → 195h.Bit1-0
		7.6.161	修正	197h.Bit2-0 → 197h.Bit1-0
			修正	表に EndAdrs[5], :EndAdrs[4], EndAdrs[3], EndAdrs[2]を記載
		8.1	削除	UVI, UVO
			追加	CVI, IMI, CVO
		8.2	修正	CVDD
			削除	PVDD
			追加	CVI, UVI
		8.3	修正	表を全般的に修正
		8.4.3	修正	AC スペックを記載(trcy, tras, trdf, trdh, trbh, twcy, twas, tdm の各値を修正)
		9.2.1～ 9.2.4	修正	図を全般的に修正
		Appendix A	追加	IDE_Config_1.Swap ビット設定
		Appendix B	追加	リトルエンディアンの CPU への接続
		Appendix C	追加	1ポートモード
06/02/24	1.0	5	修正	GENERAL→OSC
			修正	CA1:I/O、RESET、端子タイプの各項目を記載
			修正	LVDD: 端子説明の項目において、OSC I/O 用電源を追加記載
		6.2.7.11.5 6.2.7.11.5.3 6.2.7.11.6 6.2.7.11.7 6.2.7.11.8	修正	誤記訂正: PM_Control.GoActiveALL → PM_Control_0.GoActDevice
		6.2.7.11.5.3 6.2.7.11.6 6.2.7.11.7 6.2.7.11.8	修正	誤記訂正: ACTIVEALL → ACT_DEVICE GoActiveALL → GoActDevice PM_Control.PM_State[2:0] → PM_Control_1.PM_State[2:0]
		6.3.9.1.5	追加	連続遷移用コマンド使用推奨に関する(注)を追加

		6.5.3 6.5.4 6.5.5	修正	斜体文字以外のレジスタについて、アクセスに関する誤記を訂正
		6.5.3 7.1	追加	0x40~0x47, 0x50~0x6F のレジスタに関する(注)を追加
		6.7.1	修正 追加	誤記訂正: CPU_BusSwap → CPU_Endian 「Swap 機能は CPU_Endian ビットを設定した後、B9h 番地をリードする事で有効になります。」を記載
		6.7.2	追加	「モード切り替えに際して」の項を追加
		6.7.2.1	追加	「16bit BE mode をご使用の場合」の項を追加
		6.7.2.2	追加	「8bit mode をご使用の場合」の項を追加
		7.1	修正	誤記訂正: 0x90 IDE_Status の R/W 項において、R/W → R
		7.1 7.4.26 7.4.27	修正	Bit Symbol 名称変更: FIFO_Rd[15:8] → FIFO_Rd_0[7:0] FIFO_Rd[7:0] → FIFO_Rd_1[7:0]
		7.1 7.4.28 7.4.29	修正	Bit Symbol 名称変更: FIFO_Wr[15:8] → FIFO_Wr_0[7:0] FIFO_Wr[7:0] → FIFO_Wr_1[7:0]
		7.1 7.4.42 7.4.43	修正	Bit Symbol 名称変更: RAM_WrDoor[15:8] → RAM_WrDoor_0[7:0] RAM_WrDoor[7:0] → RAM_WrDoor_1[7:0]
06/02/24	1.0	7.1 7.4.92 7.4.93 7.4.107 7.4.108	修正	Bit Symbol 名称変更: DMA_RdData[15:8] → DMA_RdData_0[7:0] DMA_RdData[7:0] → DMA_RdData_1[7:0]
		7.1 7.4.94 7.4.95 7.4.109 7.4.110	修正	Bit Symbol 名称変更: DMA_WrData[15:8] → DMA_WrData_0[7:0] DMA_WrData[7:0] → DMA_WrData_1[7:0]
		7.1 7.4.127 7.4.128	修正	Bit Symbol 名称変更: IDE_RdRegValue[15:8] → IDE_RdRegValue_0[7:0] IDE_RdRegValue[7:0] → IDE_RdRegValue_1[7:0]
		7.1 7.4.129 7.4.130	修正	Bit Symbol 名称変更: IDE_WrRegValue[15:8] → IDE_WrRegValue_0[7:0] IDE_WrRegValue[7:0] → IDE_WrRegValue_1[7:0]
		7.1 7.4.147 Appendix B	追加	B9h を CPU_ChgEndian レジスタとして定義
		7.3	修正	誤記訂正: 0x12B H_CTL_SupportControl の Reset 項において、0xXX → 0x00
			修正	誤記訂正: 0x13A H_CTL_SupportControl.BO_TransportState[1:0]のビット位置において、bit13 → bit13-12
			修正	誤記訂正: 0x13A H_CTL_SupportControl.BO_BO_SupportGo のビット位置において、bit9 → bit8
			修正	誤記訂正: 0x13B H_CSW_RcvDataSize の R/W 項において、R/W → R
			修正	誤記訂正: 0x171 H_CheConfig_1 において、ToggleMode を削除
		7.4.3	修正	誤記訂正: H_SIE_IntStat1 → H_SIE_IntStat_1, H_SIE_IntStat0 → H_SIE_IntStat_0 H_SIE_IntEnb1 → H_SIE_IntEnb_1, H_SIE_IntEnb0 → H_SIE_IntEnb_0

		7.4.10	修正	誤記訂正: EnH_SIE_IntStat1 → EnH_SIE_IntStat_1, EnH_SIE_IntStat0 → EnH_SIE_IntStat_0
		7.4.139	修正	誤記訂正:B0h → B1h
		7.4.143	修正	xActIDE_Termの制御対象端子において、XHPDIAGの記載もれを修正
		7.4.146 7.4.148	修正	B9hをレジスタ定義することにもない、 B8h ~ DFh Reserved()をB8h Reserved()とBAh~DFh Reserved()に項を分割
		7.5.11 7.5.13 7.5.14 7.5.15 7.5.16	修正	誤記訂正:MainIntStat → DeviceIntStat
		7.6.13 7.6.14 7.6.15 7.6.16 7.6.17 7.6.18	修正	誤記訂正:MainIntStat → HostIntStat
		7.6.48	修正	誤記訂正:Bit3-2 SpeedMode[1:0]の説明において「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。」を削除
		8.1 8.2	追加	入力電圧の項目においてLVIを追加 表外のコメントに※4を追加
			修正	誤記訂正:※3 VBUS → ※3 VBUS_B
		8.2	修正	周囲温度のMax値:85 → 110
		8.3	修正	電源供給電流の項目において、 IDDCをIDDCHとIDDCLに分割 TYP値を記載
			修正	静止電流の項目において、MAX値を記載
			修正	入力リーク電流の項目において、条件にLVIH=LVDDを追加記載
			修正	入力リーク電流(5Vトレラント)の項目において誤記訂正:HVOH → HVIH
			修正	入力特性(LVCMOS)の項目において、端子名の文字欠落を訂正:XHPDIAG
06/02/24	1.0	8.3	修正	入力特性の項目において、端子名の文字欠落を訂正:VBUSFLG_A
			修正	誤記訂正:HDD[15:0]~XHRESETの出力特性において、 CVDD → HVDD
			修正	誤記訂正:TOD,VBUSEN_Aの出力特性において、 VOH2 → VOH4, VOL2 → VOL4
			修正	誤記訂正:CD[15:0], XINTのOFF-STATEリーク電流において、 HVDD → CVDD, HVOH → CVOH
			修正	誤記訂正:HDD[15:0]~XHRESETのOFF-STATEリーク電流において、 LVDD=1.65Vを削除
			修正	端子容量の項目において、MAX値を記載
		9.1	修正	8bit CPUの接続例において、DATA[15:8]とDATA[7:0]の接続に対する誤記を修正
		9.2.1 9.2.2 9.2.3 9.2.4	修正	10u コンデンサを削除
		Appendix C	追加	1ポートモード時のD_Reset.ResetDTMおよびH_Reset.ResetHTMの設定を記載
06/4/10	1.1	3.9	削除	記述の重複につき、「また、DMA 2ch. を搭載します。」を削除。
		6.3.9.3.4.3 6.3.9.3.4.4 6.3.9.3.5	訂正	誤)H_USB_Status.PortSpeed[1:0] 正)H_NegoControl_1.PortSpeed[1:0]

		7.1	訂正	0x12 誤) EnH_SIE_IntEnb_1 誤) En_FrameIntStat 0x14 誤) EncompleteINTRQ	正) EnH_SIE_IntStat_1 正) EnH_FrameIntStat 正) EnCompleteINTRQ
		7.4.30 7.4.31	削除	誤記に付き、以下の説明を削除。 「 D_EPx{x=0,a-c}Join.JoinCPU_Rd ビット、または H_CHx{x=0,a-e}Join.JoinCPU_Rd ビット、または MediaFIFO_Join.JoinCPU_Rd ビットがセットされたときに該当するエンドポイント/チャンネル/メディアにデータが無い場合、そのエンドポイント/チャンネル/メディアに USB 等からデータが書き込まれるまでは、このビットはセットされません(RdRemainValid な"0"という状態にはなりません)。」	
		7.6.1	追記	Bit3 DetectDiscon の説明に、制限事項を追記。	
		7.6.31	訂正	誤) 本レジスタの下位 5 ビットのいずれかのビットに"1"をセットし、EnHS_Test に"1"をセットします。 正) 本レジスタの下位 5 ビットのいずれかのビットと、EnHS_Test とに、同時に"1"をセットします。	
				EnHS_Test ビットの説明 誤) このビットに"1"をセットすると、H_USB_Test レジスタの下位 5 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。 正) 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を、同時に"1"に設定すると、下位 5 ビットに対応するテストモードに入ります。	
				Test_ForceEnable, Test_SE0_NAK, TEST_J, TEST_K, Test_Packet ビットの説明 誤) 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、~ 正) 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、~	
		7.6.6 7.6.7 7.6.8 7.6.9 7.6.10 7.6.11	追記	Bit6 TranACK ビットの説明に、H_CHx{x=0,a-e}Config_0.ACK_Cnt ビットとの関連を追記。	
		9.2.1	訂正	DP/DM ライン保護用、バリスタを付加。	
		9.2.2	訂正	DP/DM ライン保護用、バリスタを付加。	
		9.2.3	訂正	DP/DM ライン保護用、バリスタを付加。	
		9.2.4	訂正	Ball A9 の名称 誤) PVDD Ball B9 の名称 誤) PVSS	正) LVDD 正) VSS
		Appendix C	修正	接続例において、VBUS 制御回路を、Reverse Current Protection 機能を有する USB パワースイッチ IC を使用した回路に変更(注記の変更を含む)。 DP/DM ライン保護用、バリスタを付加。	
		7.3 7.6.162 Appendix D	追記	0x1F5 H_Protect レジスタを追加 0x1F6 H_Monitor レジスタを追加	
06/7/14	1.2	2	修正	電源システムを2系統から3系統に修正	
		2	追記	パッケージ種に PFBGA10UX121 を追加	
		3.1	追記	PLL のクロックソースを明記	
		4	修正	図 4-2 機種名表記を修正し、パッケージタイプを追加	
		5	修正	VBUS_B の端子タイプ及び RESET 値を"PD"に修正	
		6.2.2.2	追加	OUT トランザクションの正常完了時の応答に NYET を追加	

6.6.1.5.6 6.6.2.4.6 6.6.3.2.4	修正	FIFO のアクセス制限、アクセス要因の重複に関する説明を修正
6.6.2.4.6	追加	Media FIFO に関する排他条件を追加
6.9.2	修正	DEVICE_CODE の誤記を修正
6.9.3	追加	バウンダリスキャン除外端子を追加
6.1.2.1 6.1.2.1.1 6.2.7.11.8 7.4.2 7.4.23 8.3	修正	端子名を VBUS から VBUS_B に修正
7.2 7.5.42	追加	D_USB_Address.SetAddress ビットを追加
7.2 Appendix B Appendix E	追加	Appendix E を追加 D_ModeControl レジスタを追加
7.4.1, 7.4.2, 7.4.3, 7.4.9, 7.4.10, 7.4.15, 7.4.141, 7.4.145, 7.4.147	修正	レジスタ名称/ビット名称を太字斜体に修正
7.4.45	修正	アドレス誤記を修正(4Ah⇒49h)
7.6.29	修正	PortSpeed[1:0]のアクセスタイプを RW に修正
8.2	追加	電源オフ手順を追記
8.3	修正	電源供給電流の規格値を修正、説明を追記 静止電流の規格値を修正
8.3	追加	VBUS_B 端子のプルダウン抵抗特性を追加
8.4.2	削除	不要なマーカ" *"を削除
8.4.3	追加	tsah 規定を追加 CVDD=3.3V と制限した場合の緩和規定を追加(8.4.3.2) (既存の規定を 8.4.3.1 とした)
9.2.2 9.2.4	修正	VBUS 給電部品をシンボル化
9.2.3 9.2.4	修正	機種名表記を修正し、パッケージタイプを追加
10.2	追加	タイトルにパッケージタイプを表記
10.3	追加	パッケージタイプ追加
Appendix B	追加	H_Protect, H_Monitor レジスタを追加
Appendix C	修正	接続例図の VBUS 給電部品をシンボル化
Appendix D	削除	重複する記載を削除

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 : 410667301

2006年 05月 作成

2006年 10月 改訂