EPSON
EXCEED YOUR VISION

S2S65A00
テクニカルマニュアル

セイコーエプソン株式会社

Rev.1.3

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、これらに起因する第三者の知的財産権およびその他の権利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知的財産権およびその他の権利の実施権の許諾を行うものではありません。

3. 特性値の数値の大小は、数直線上の大小関係で表しています。

4. 本資料に掲載されている製品のうち「外国為替及び外国貿易法」に定める戦略物資に該当するものについては、輸出する場合、同法に基づく輸出許可が必要です。

5. 本資料に掲載されている製品は、生命維持装置その他、きわめて高い信頼性が要求される用途を前提としていません。よって、弊社は本（当該）製品をこれらの用途に用いた場合のいかなる責任についても負いかねます。

6. 資料に掲載されている会社名、商品名は、各社の商標または登録商標です。

は、ARM 社の登録商標です。

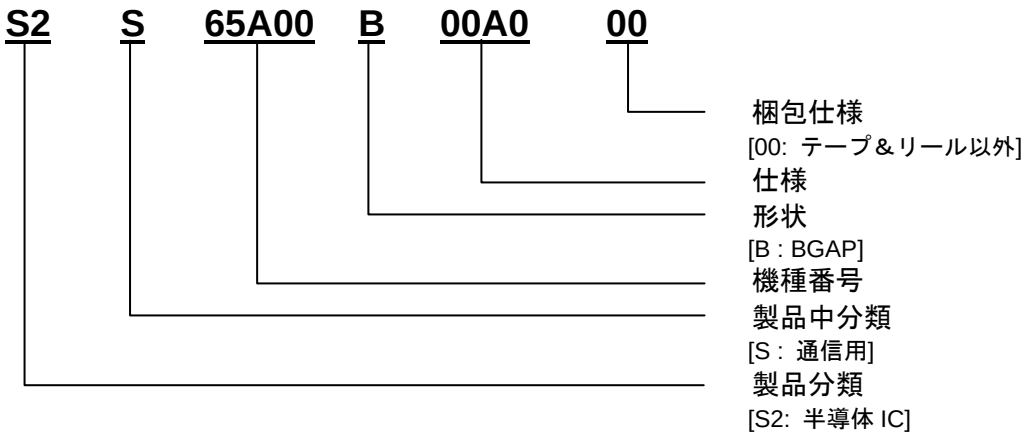
CompactFlash は Sandisk 社の登録商標です。

その他記載の会社名、製品名は、それぞれの会社の商標もしくは登録商標です。

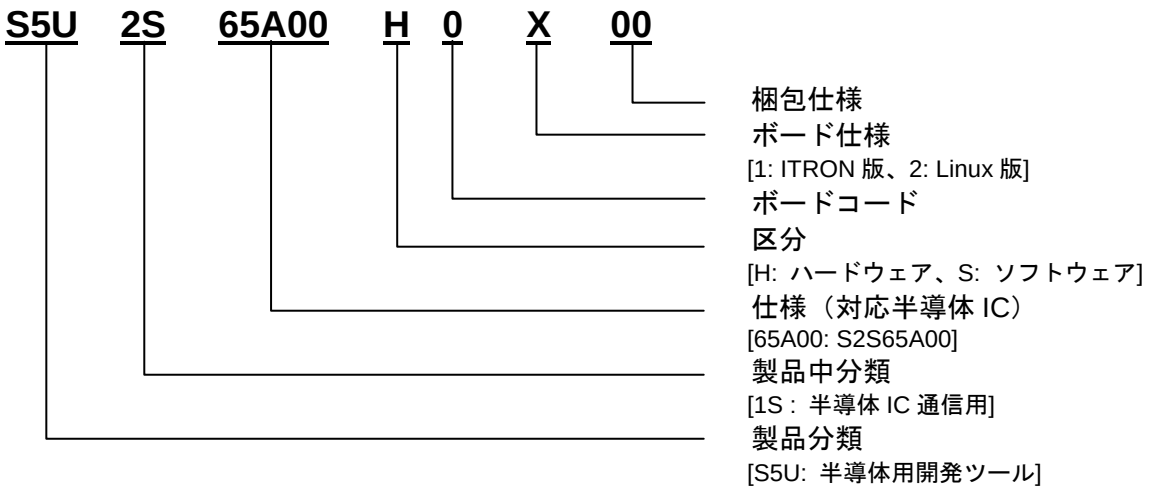
©SEIKO EPSON CORPORATION 2011, All rights reserved.

製品型番体系

●デバイス



●開発ツール



使用上の注意事項

本書のレジスタの記述に関しては、以下のことにご注意ください。

本書のレジスタに関する記述については、以下のような省略を用いることがあります。

R/W：	リードおよびライト
RO：	リードオンリ
WO：	ライトオンリ
RSV：	予約ビット／レジスタ（特に指定のない場合は“0”を書き込んでください。）
n/a：	not available（特に指定のない場合は“0”を書き込んでください。）

特に指定のない場合、レジスタの予約ビットには“0”をセットしてください。予約ビットに書き込みを行うと予想できない結果になることがあります。“n/a”と記載してあるビットはハードウェアに影響をあたえません。

あるレジスタは特定の条件のときのみアクセスできるようになっています。アクセス不可のレジスタへのリード／ライトは無効です。

目　次

1. 概要 1
1.1 特長 1
1.2 内蔵機能 1
2. ブロック図 4
3. 端子 5
3.1 端子配置図 5
3.2 端子の説明 7
3.3 GPIO端子のマルチプレクス端子機能、リセット直後の端子機能 18
3.4 リセット中の端子の状態 20
4. 機能説明 21
4.1 システムコンフィギュレーション 21
4.2 メモリマップ 22
4.2.1 メモリマップ（AHB1） 22
4.2.2 メモリマップ (AHB2) 23
4.3 I/Oマップ 24
4.4 割り込みコントローラ 25
4.5 S2S65A00 の内蔵機能 26
5. CPU 27
5.1 概要 27
5.2 ARM720Tブロック図 27
6. DMAコントローラ 1 (DMAC1) 28
6.1 概要 28
6.2 ブロック図 28
6.3 外部端子 29
6.4 レジスタ 29
6.4.1 レジスター覧 29
6.4.2 レジスタ詳細 30
7. カメラインタフェース[2:1] (CAM[2:1]) 36
7.1 概要 36
7.2 ブロック図 36
7.3 外部端子 37
7.4 レジスタ 37
7.4.1 レジスター覧 37
7.4.2 レジスタ詳細説明 38
7.5 動作説明 45
7.5.1 フレームキャプチャ割り込み 46
8. JPEGコントローラ (JPG[2:1]) 48
8.1 概要 48
8.2 ブロック図 48
8.3 外部端子 48
8.4 レジスタ 49
8.4.1 レジスター覧 49

8.4.2 Resizer Operation Registers (RSZ[2:1]) 51
8.4.3 JPEG Module Registers (JCTL[2:1]) 56
8.4.4 JPEG FIFO Setting Register (JFIFO[2:1]) 64
8.4.5 JPEG ラインバッファ設定レジスタ (JLB[2:1]) 68
8.4.6 JPEG Codec Registers (JCODEC[2:1]) 72
8.5 動作説明 81
8.5.1 キャプチャ制御機能 81
8.5.1.1 カメラ画像JPEGエンコード用ステートマシン 81
8.5.1.2 YUVキャプチャ用ステートマシン 83
8.5.2 リサイズ機能 85
8.5.2.1 切り取り機能 85
8.5.2.2 縮小機能 86
8.5.2.3 リサイズ回路利用制限 89
8.5.3 画像処理データフロー 90
8.5.3.1 カメラ画像JPEGエンコード 91
8.5.3.2 YUVデータキャプチャ 92
8.5.4 JPEGコーデック機能 93
8.5.4.1 JPEGコーデックレジスタに関する制限 94
8.5.5 JPEGコーデック以外の機能 95
8.5.5.1 JPEG FIFO 95
8.5.5.2 JPEGラインバッファ 95
8.5.5.3 YUVフォーマットコンバータ 96
8.5.5.4 JPEGモジュール割り込み 97
8.5.5.5 JPEG180°回転エンコード 97
8.5.5.6 ソフトウェアリセット処理について 98
8.5.5.7 マーカ高速出力モード 98
8.5.6 シーケンス例 99
8.5.6.1 カメラ画像JPEGエンコード処理（単フレーム） 99
8.5.6.2 終了処理 101
9. JPEG_DMAC (JDMA) 102
9.1 概要 102
9.2 ブロック図 102
9.3 外部端子 102
9.4 レジスタ 103
9.4.1 レジスター覧 103
9.4.2 レジスタ詳細説明 104
10. DMAコントローラ 2 (DMAC2) 108
10.1 概要 108
10.2 ブロック図 108
10.3 外部端子 108
10.4 レジスタ 109
10.4.1 レジスター覧 109
10.4.2 レジスタ詳細 110
11. USB HS-Device 115
11.1 概要 115
11.2 ブロック図 115
11.3 特長 116
12. APBブリッジ（APB） 117
12.1 概要 117
12.2 ブロック図 117
12.3 外部端子 118

12.4 レジスタ 118
12.4.1 レジスター覧 118
12.4.2 レジスタ詳細説明 118
13. システムコントローラ (SYS) 121
13.1 概要 121
13.2 動作モード 121
13.2.1 パワーオンステート 122
13.2.2 ロースピードモード 122
13.2.3 ロースピードHALTモード 122
13.2.4 ハイスピードモード 122
13.2.5 ハイスピードHALTモード 122
13.3 外部端子 123
13.4 レジスタ 123
13.4.1 レジスター覧 123
13.4.2 レジスタ詳細説明 124
13.5 Appendix A: PLL Setting Example 137
13.6 Appendix B: PLL Parameter table 138
13.7 Appendix C: Remap後のメモリマップ 139
13.7.1 Remap後のメモリマップ (AHB1) 139
13.7.2 Remap後のメモリマップ (AHB2) 141
13.8 Appendix D：クロック・コントロールブロック図 143
13.9 Appendix E：UART Clock 設定例: 144
14. メモリコントローラ (MEMC) 145
14.1 概要 145
14.1.1 SRAMコントローラ 145
14.1.2 SDRAM コントローラ 145
14.1.3 外部バスインタフェースモジュール 145
14.2 ブロック図 146
14.3 外部端子 146
14.4 Memory controller 147
14.4.1 デバイス数 147
14.4.2 外部メモリ幅 147
14.4.3 デバイスのセグメント設定 147
14.5 SRAM control 147
14.5.1 タイミング設定 147
14.5.2 ライトプロテクト 148
14.6 SDRAM control 148
14.6.1 モードレジスタ設定 148
14.6.2 バースト対応 148
14.6.3 オートプリチャージ設定 148
14.6.4 低消費電力化 148
14.6.5 メモリクロック停止 148
14.6.6 省電力モードサポート 149
14.6.7 オートリフレッシュ制御 149
14.6.8 セルフリフレッシュ制御 149
14.6.9 ステータスレジスタ 149
14.7 レジスタ 150
14.7.1 レジスター覧 150
14.7.2 レジスタ詳細 151
15. 割り込みコントローラ (INT) 158
15.1 概要 158
15.2 ブロック図 160

15.3 FIQ **160**
15.4 IRQ **160**
15.5 外部端子 **161**
15.6 レジスタ **161**
15.6.1 レジスター覧 161
15.6.2 レジスタ詳細説明 162
16. UART1/2/3 ..... **173**
16.1 概要 **173**
16.2 ブロック図 **173**
16.3 外部端子 **174**
16.4 レジスタ説明 **175**
16.4.1 レジスター覧 175
16.4.2 レジスタアクセスにおける注意事項 175
16.4.3 レジスタ詳細説明 176
16.4.4 ボーレート設定例 189
16.5 本UART1/2/3 の利用制限事項 **190**
17. I²C Single Master Core Module (I2C) **191**
17.1 概要 **191**
17.1.1 マスタモード 191
17.1.2 スレーブモード 191
17.2 ブロック図 **192**
17.3 外部端子 **192**
17.4 レジスタ **193**
17.4.1 レジスター覧 193
17.4.2 レジスタ詳細説明 194
17.5 動作説明（バス・コントロール・コマンド使用例） **202**
17.5.1 スタート（S）フロー例 202
17.5.2 ストップ（P）フロー例 202
17.5.3 レシーブ（R）フロー例 202
17.5.4 トランスファ（T）フロー例 203
17.5.5 スレーブ機器への書込みシーケンス例 203
17.5.6 スレーブからの読出しシーケンス例 204
17.6 本I2C Single Master Core Module (I2C)の利用制限事項 **204**
18. I²S (I2S) **205**
18.1 概要 **205**
18.1.1 機能 205
18.2 ブロック図 **205**
18.3 外部端子 **206**
18.4 レジスタ説明 **206**
18.4.1 レジスター覧 206
18.4.2 レジスタ詳細説明 207
18.5 機能説明 **214**
18.5.1 I2Sタイミングチャート (32fs) 214
18.5.2 データ幅とFIFO段数 214
18.5.3 DMA転送 215
18.5.4 INTCへの割り込み 215
18.5.5 クロック選択（クロック共有） 215
18.5.6 モノラルーステレオ変換機能 216
18.6 設定例 **216**
19. シリアル周辺機器インタフェース (SPI) **217**

19.1 概要 217
19.1.1 マスタモード 217
19.1.2 スレーブモード 218
19.2 ブロック図 219
19.3 外部端子 219
19.4 クロックとデータ転送タイミング 220
19.5 レジスタ説明 222
19.5.1 レジスター覧 222
19.5.2 レジスタ詳細説明 222
20. コンパクトフラッシュカードインタフェース (CF) 228
20.1 概要 228
20.2 ブロック図 228
20.3 CF Card I/Fの各空間の配分 229
20.4 外部端子 230
20.5 レジスタ 231
20.5.1 レジスター覧 231
20.5.2 レジスタ詳細説明 231
20.6 本コンパクトフラッシュカードインタフェースの利用制限事項 235
21. SDメモリカードインタフェース 236
21.1 概要 236
22. タイマA (TIMA) 237
22.1 概要 237
22.2 ブロック図 237
22.3 外部端子 238
22.4 レジスタ 238
22.4.1 レジスター覧 238
22.4.2 レジスタ詳細説明 239
22.5 各モードでのロード値の設定方法 247
22.5.1 タイマカウンタのモード 247
22.6 タイマ内部クロック設定例（1KHz、1MHz） 248
22.6.1 ディバイダとプリスケーラの設定方法 248
22.7 タイミング図 249
22.7.1 サイクリックモード時のイミディエイトロードリクエスト 249
22.7.2 サイクリックモード時の通常リロード 249
22.7.3 シングルモード時の通常リロード 250
22.7.4 ポート出力 250
23. タイマB (TIMB) 252
23.1 概要 252
23.2 特徴 252
23.3 ブロック図 252
23.4 外部端子 253
23.5 レジスタ 253
23.5.1 レジスター覧 253
23.5.2 レジスタ詳細説明 254
23.6 動作説明 262
23.6.1 カウント動作 262
23.6.2 タイマ入出力機能 263
24. リアルタイムクロック (RTC) 266

24.1 概要 266
24.2 ブロック図 266
24.3 外部端子 266
24.4 レジスタ 267
24.4.1 レジスター覧 267
24.4.2 レジスタ詳細説明 268
24.5 RTCのレジスタ設定方法 275
24.5.1 パワーオン後の初期設定 275
24.5.2 動作停止、動作再開始 276
24.5.3 動作中の動作停止以外の再設定 276
24.5.4 動作中のシステムリセット後の再設定 276
24.5.5 プログラミング上の注意事項 276
25. ウォッチドッグタイマ (WDT) 277
25.1 概要 277
25.2 ブロック図 277
25.3 外部端子 278
25.4 レジスタ 278
25.4.1 レジスター覧 278
25.4.2 レジスタ詳細説明 278
26. GPIO 281
26.1 概要 281
26.2 外部端子 282
26.3 レジスタ 284
26.3.1 レジスター覧 284
26.3.2 レジスタ詳細説明 285
26.3.2.1 GPIOAレジスタ 285
26.3.2.2 GPIOBレジスタ 286
26.3.2.3 GPIOCレジスタ 287
26.3.2.4 GPIODレジスタ 288
26.3.2.5 GPIOEレジスタ 289
26.3.2.6 GPIOFレジスタ 290
26.3.2.7 GPIOGレジスタ 291
26.3.2.8 GPIOHレジスタ 292
26.3.2.9 GPIOIレジスタ 293
26.3.2.10 GPIOJレジスタ 294
26.3.2.11 GPIOKレジスタ 295
26.4 GPIOAおよびGPIOBの割り込みロジック 298
27. A/D変換コントローラ(ADC) 299
27.1 概要 299
27.2 ブロック図 299
27.3 外部端子 299
27.4 動作ステート 300
27.5 レジスタ 301
27.5.1 レジスター覧 301
27.5.2 レジスタ詳細説明 301
27.6 使用例 306
27.7 入力電圧と変換データ値 307
28. エリアセンサ(ARS) 308
28.1 概要 308
28.1.1 機能 308

28.2 ブロック図 309
28.3 外部端子 309
28.4 レジスタ説明 309
28.4.1 レジスター覧 309
28.4.2 レジスタ詳細説明 310
29. 絶対最大定格 313
29.1 絶対最大定格 313
29.2 推奨動作条件（2 電源、3.3V対応入出力バッファ） 313
29.3 電源投入タイミング 314
29.4 電源切断タイミング 314
30. 電気的特性 315
30.1 DC特性 315
30.2 AC特性 316
30.2.1 AC特性測定条件 316
30.2.2 AC特性タイミング一覧表 316
30.2.2.1 クロックタイミング 316
30.2.2.2 CPUコントロール信号タイミング 316
30.2.2.3 バッテリーバックアップモードタイミング 316
30.2.2.4 カメラインタフェース (CAM) タイミング 317
30.2.2.5 メモリコントローラ (MEMC) タイミング 317
30.2.2.6 I^2C Single Master Core Module (I2C) タイミング 319
30.2.2.7 I^2Sタイミング 319
30.2.2.8 シリアル周辺機器インタフェース (SPI) タイミング 320
30.2.2.9 コンパクトフラッシュインタフェース (CF) タイミング 320
30.2.3 タイミングチャート 322
30.2.3.1 クロックタイミング 322
30.2.3.2 CPUコントロール信号タイミング 323
30.2.3.3 バッテリーバックアップモードタイミング 325
30.2.3.4 カメラインタフェースタイミング 326
30.2.3.5 メモリインタフェースコントローラ 327
30.2.3.6 I2C Single Master Core Module タイミング 338
30.2.3.7 I2Sタイミング 338
30.2.3.8 シリアル周辺機器インタフェースタイミング 339
30.2.3.9 コンパクトフラッシュインタフェース (CF) タイミング 340
31. 参考外部接続例 343
31.1 メモリ接続例 343
31.2 コンパクトフラッシュ接続例（16 ビットバス対応） 345
31.3 シリアル周辺機器インタフェース（SPI）接続例 346
31.3.1 マスタ時 346
31.3.2 スレーブ時 346
31.4 I^2S接続例 347
31.4.1 マスタ時 347
31.4.2 スレーブ時 347
32. 外形寸法図 348
32.1 Plastic TFBGA 280pin Body size 16x16x1.2mm (PFBGA16U-280) 348
33. 改訂履歴表 349
Appendix 1　S2S65A00 内部レジスター覧 358

Appendix 2　USB Device コントローラ

< Table of Contents >

A2.1　Scope 376

A2.2 Overview 376

A2.3 Block Diagram 376

A2.4 Pin List 377

A2.5 Register 378

A2.5.1　Register一覧 378

A2.5.2　Register詳細説明 382

A2.5.2.1　Register Access上の注意事項 382
A2.5.2.2　割込み制御レジスタ 382
A2.5.2.3　Power Management制御レジスタ 385
A2.5.2.4　MISCレジスタ 387
A2.5.2.5　DMA制御レジスタ 390
A2.5.2.6　USB制御レジスタ 394
A2.5.2.7　USB FIFO設定レジスタ 423

A2.6 機能説明 432

A2.6.1　初期設定 432

A2.6.1.1　USBコントローラへのアクセス設定 432
A2.6.1.2　割り込みの設定 432
A2.6.1.3　Macro Config1[0x37]に関する注意事項 432

A2.6.2　USBデバイス制御 433

A2.6.2.1　エンドポイント 433
A2.6.2.2　トランザクション 434
A2.6.2.2.1　SETUPトランザクション 436
A2.6.2.2.2　バルク/インタラプトOUTトランザクション 437
A2.6.2.2.3　バルク/インタラプトINトランザクション 438
A2.6.2.2.4　PINGトランザクション 439
A2.6.2.3　コントロール転送 440
A2.6.2.3.1　セットアップステージ 441
A2.6.2.3.2　データステージ／ステータスステージ 441
A2.6.2.3.3　自動アドレス設定機能 441
A2.6.2.3.4　デスクリプタ返信機能 442
A2.6.2.4　バルク転送/インタラプト転送 442
A2.6.2.5　データフロー 442
A2.6.2.5.1　OUT転送 442
A2.6.2.5.2　IN転送 443
A2.6.2.6　バルクオンリーサポート 443
A2.6.2.6.1　CBWサポート 443
A2.6.2.6.2　CSWサポート 444
A2.6.2.7　オート・ネゴシエーション機能 445
A2.6.2.7.1　DISABLE 445
A2.6.2.7.2　IDLE 446
A2.6.2.7.3　WAIT_TIM3US 446
A2.6.2.7.4　WAIT_CHIRP 446
A2.6.2.7.5　WAIT_RSTEND 446
A2.6.2.7.6　DET_SUSPEND 446
A2.6.2.7.7　IN_SUSPEND 447
A2.6.2.7.8　CHK_EVENT 447
A2.6.2.7.9　WAIT_RESTORE 447

A2.6.2.7.10 ERR ... 447
A2.6.2.7.11 各ネゴシエーション機能の単体説明 ... 447
A2.6.2.7.11.1 サスペンド検出（HSモード） ... 447
A2.6.2.7.11.2 サスペンド検出（FSモード） ... 449
A2.6.2.7.11.3 リセット検出（HSモード） ... 450
A2.6.2.7.11.4 HS Detection Handshake ... 452
A2.6.2.7.11.5 FSのダウンストリームポートに繋がれた場合 ... 452
A2.6.2.7.11.6 HSのダウンストリームポートに繋がれた場合 ... 454
A2.6.2.7.11.7 スヌーズ中にリセットされた場合 ... 456
A2.6.2.7.11.8 レジュームの発行 ... 457
A2.6.2.7.11.9 レジュームの検出 ... 459
A2.6.2.7.11.10 ケーブル挿入 ... 460
A2.6.3 パワーマネージメント機能 ... 462
A2.6.3.1 SLEEP（スリープ） ... 463
A2.6.3.2 SNOOZE（スヌーズ） ... 464
A2.6.3.3 ACTIVE60（アクティブ 60） ... 464
A2.6.3.4 ACT_DEVICE（アクトデバイス） ... 464
A2.6.4 FIFO管理 ... 465
A2.6.4.1 FIFOメモリマップ ... 465
A2.6.4.2 デスクリプタエリアの使用方法 ... 466
A2.6.4.2.1 デスクリプタエリアへのデータの書込み ... 466
A2.6.4.2.2 デスクリプタエリアでのデータステージ（IN）の実行 ... 466
A2.6.4.3 CBWエリアの使用方法 ... 466
A2.6.4.3.1 CBWエリアへの受信 ... 467
A2.6.4.3.2 CBWエリアからのデータの読み出し ... 467
A2.6.4.4 CSWエリアの使用方法 ... 467
A2.6.4.4.1 CSWエリアからの送信 ... 467
A2.6.4.4.2 CSWエリアへのデータの書き込み ... 467
A2.6.4.5 FIFOへのアクセス方法 ... 467
A2.6.4.5.1 FIFOへのアクセス方法（RAM_Rd） ... 467
A2.6.4.5.2 FIFOへのアクセス方法（RAM_WrDoor） ... 468
A2.6.4.5.3 FIFOへのアクセス方法（レジスタアクセス） ... 468
A2.6.4.5.3.1 FIFOアクセス（ライト） ... 468
A2.6.4.5.3.2 FIFOアクセス（リード） ... 468
A2.6.4.5.3.3 FIFOアクセスの端数処理 ... 469
A2.6.4.5.4 FIFOへのアクセス方法（DMA） ... 471
A2.6.4.5.5 FIFOへのアクセス制限 ... 471
A2.6.5 DMA ... 472
A2.6.5.1 概要 ... 472
A2.6.5.2 基本機能 ... 472
A2.6.5.2.1 カウントモード ... 472
A2.6.5.2.2 フリーランモード ... 473
A2.6.5.3 動作手順 ... 473
A2.6.5.4 強制終了 ... 474

<List of Figures>

Fig. A2.3.1 S2S65A00 USB Controller Block Diagram 366
Fig. A2.4.2 USB 外部端子 367
Fig. A2.6.1 SETUP トランザクション 427
Fig. A2.6.2 OUT トランザクション 428
Fig. A2.6.3 IN トランザクション 429
Fig. A2.6.4 PING トランザクション 429
Fig. A2.6.5 データステージが OUT 方向のコントロール転送 430
Fig. A2.6.6 データステージが IN 方向のコントロール転送 431
Fig. A2.6.7 オート・ネゴシエータ 435
Fig. A2.6.8 Suspend Timing (HS mode) 438
Fig. A2.6.9 Suspend Timing (FS mode) 439
Fig. A2.6.10 Reset Timing (HS mode) 440
Fig. A2.6.11 Reset Timing (FS mode) 441
Fig. A2.6.12 HS Detection Handshake (FS mode) 443
Fig. A2.6.13 HS Detection Handshake Timing (HS mode) 445
Fig. A2.6.14 HS Detection Handshake Timing from Suspend 446
Fig. A2.6.15 Assert Resume Timing (HS mode) 448
Fig. A2.6.16 Detect Resume Timing (HS mode) 449
Fig. A2.6.17 Device Attach Timing 451
Fig. A2.6.18 パワーマネージメント 452
Fig. A2.6.19 SLEEP ステートからの離脱（GoSNOOZE 時） 453
Fig. A2.6.20 SNOOZE ステートからの離脱（GoActive60 時） 454
Fig. A2.6.21 FIFO メモリマップ例 455
Fig. A2.6.22 FIFO ライト処理（正常動作） 459
Fig. A2.6.23 FIFO ライト処理（注意が必要な動作） 460
Fig. A2.6.24 FIFO リード処理（正常動作） 460
Fig. A2.6.25 FIFO リード処理（注意が必要な動作） 461
Fig. A2.6.26 DMA 転送手順例 464

<List of Tables>

Tbl. A2.5.1 USB レジスター覧（1/4） 368
Tbl. A2.5.2 USB レジスター覧（2/4） 369
Tbl. A2.5.3 USB レジスター覧（3/4） 370
Tbl. A2.5.4 USB レジスター覧（4/4） 371
Tbl. A2.6.1 エンドポイント EP0 の基本設定項目 423
Tbl. A2.6.2 汎用エンドポイントの基本設定項目 424
Tbl. A2.6.3 エンドポイント EP0 の制御項目及びステータス 425
Tbl. A2.6.4 汎用エンドポイントの制御項目とステータス 426
Tbl. A2.6.5 Suspend Timing Values (HS mode) 438
Tbl. A2.6.6 Suspend Timing Values (FS mode) 439
Tbl. A2.6.7 Reset Timing Values (HS mode) 440
Tbl. A2.6.8 Reset Timing Values (FS mode) 441
Tbl. A2.6.9 HS Detection Handshake Timing Values (FS mode) 443
Tbl. A2.6.10 HS Detection Handshake Timing Values (HS mode) 445
Tbl. A2.6.11 HS Detection Handshake Timing Values from Suspend 447
Tbl. A2.6.12 Assert Resume Timing Values (HS mode) 448
Tbl. A2.6.13 Detect Resume Timing Values (HS mode) 450
Tbl. A2.6.14 Device Attach Timing Values 451

1. 概要

本製品 S2S65A00 はドライブレコーダ専用 IC です。カメラインタフェース機能、JPEG エンコーダ機能に加え、CF、SD Memory、USB2.0 Device(High-Speed 対応) 各種インタフェース及び 8ch の ADC を内蔵しています。S2S65A00 にカメラモジュール、SDRAM、外付けのストレージ(CF Memory Card または SD Memory Card)とファームウェアを格納した Flash ROM を接続することにより、ドライブレコーダが簡単に構成できます。一例として、内蔵エンコーダで生成した JPEG データを SDRAM に常時蓄積しておきます。加速度センサ IC による急な速度変化の検出など外部トリガを受けて、蓄積した画像データを外付けのストレージ（CF Memory Card または SD Memory Card）に転送、保存することが可能です。また、S2S65A00 は GPIO および I^2C バスを搭載していますので、カメラの設定や外部機器の制御を行うことも可能です。

1.1 特長

●1 Chip ソリューション：低システムコストを実現可能
●30 fps @VGA のハードウェアによる JPEG エンコードを実現（ISO 10918 準拠）
●2 台までカメラモジュール接続対応
●カメラモジュールごとにハードウェア JPEG エンコーダを 2 個搭載
●動体検知機能による動き検出をサポート
●I^2S による音声データのサポート
●CompactFlash インタフェースによる CF Memory Card 又は無線 LAN インタフェース（802.11b/g）対応可能
●SD Memory インタフェースによる SD Memory Card の接続対応
●ARM720T　50MHz 動作
●USB 2.0 Device(High-Speed) 機能サポートとによる PC との接続が可能
●8ch の ADC サポートによる各種アナログセンサとの接続が可能
●イベントカウンタ用のタイマを搭載
●メモリバス：2 ポート (16bit-Bus: FROM/SRAM, 16/32bit-Bus: SDRAM)

1.2 内蔵機能

CPU：
・32 ビット RISC ARM720T（最大 50MHz）
・32 ビット長命令と効率の良い Thumb コードと呼ばれる 16 ビット長命令コードを切り替えて利用可能
・32 ビット汎用レジスタ（31 本）
・乗算器を内蔵

RAM：
・56 KB の CPU/JPEG Work 用内蔵 RAM（CPU Work: 32KB max.）

スタンバイ機能
・CPU の動作が必要ない場合には CPU のクロックをとめることができる HALT 機能
・主要な I/O ブロックごとにクロックをとめられる I/O クロックストップ機能

カメラ入力／JPEG エンコーダ：
・8 ビット パラレルインタフェース×2 ポート
・2 台のカメラモジュールを接続可能
・解像度 640×480 程度まで（VGA,QVGA,CIF, QCIF）
・ハードウェア JPEG エンコーダ×2
・Max 30 fps @VGA 以上の処理能力（カメラモジュール 1 台接続時）
・YUV4-2-2 プログレッシブ（2 ポート共）
・カメラデータ入力用ピクセルクロック周波数は CPU クロック周波数の 2/3 未満

JPEG：
- ハードウェア JPEG エンコーダ
- Resize 機能 （画面切り取り可）
- 専用の Line Buffer
- JPEG エンコーダ出力に容量可変 FIFO 内蔵
- Enhanced DMA 内蔵

USB2.0 Device:
- HS(480Mbps)および FS(12Mbps)の転送をサポート
- FS/HS Termination 機能内蔵（外付け回路不要）
- VBUS 5V Interface（ただし、外付け保護回路が必要です）
- Control Bulk および Interrupt 転送をサポート
- Control(End Point 0)および Bulk/Interrupt 共用 End Point（8 本）をサポート
- 16-bit または 8-bit 幅の汎用 CPU Interface 搭載
- Littler Endian 対応
- Register Table については S1R72V05 の HS-Device 部を基本として追加／削除
- USB 用 Clock 入力として 12MHz または 24MHz 水晶発振子入力に対応
- 内部 Clock 用途としては入力 USB 用 Clock を元にして、以下の周波数が使用可能
- 入力 USB 用 Clock：12MHz または 24MHz
- 内部 USB 用 Clock：60MHz（内蔵 USB 用 PLL 経由）

メモリコントローラ：
- AHB バスインタフェースのメモリコントローラ
- SRAM タイミングデバイスを最大 4 つまでサポート可能
- SDRAM を最大 8 つまでサポート可能
- SDRAM オートリフレッシュのリフレッシュ間隔をデバイスにあわせて調整可能
- SDRAM バーストリフレッシュサポート
- SDRAM のセルフリフレッシュをサポート

CF カードインタフェース：
- CF+ 仕様 Rev1.4 準拠
- 無線 LAN および PHS カードなどの interface として利用可能
- True IDE モードのサポート

SD メモリインタフェース：
- SD Memory Card Physical Layer Spec. ver.1.0 準拠
- 1bit/4bits インタフェースサポート

割り込みコントローラ：
- 32 本の IRQ と 2 本の FIQ のサポート

シリアルインタフェース：
- UART：16550 ソフトウェア互換 × 3 チャンネル
- SPI：クロック同期式 ×1 チャンネル
- I^2C マスタインタフェース （カメラ I/F および汎用用途）
- I^2S インタフェース（音声／オーディオデータ対応、I^2S 規格に準拠）

タイマ A：
- 16 ビットタイマ ×3 チャンネルのタイマ
- リロード/Cyclic または One Shot 動作モード
- アンダーフロー出力によるトグル出力、または Port 出力をサポート

タイマ B：イベントカウンタ用
- 16 ビット・アップ・カウント・タイマ
- COMMON レジスタを 4 つ実装し、それぞれアウトプット・レジスタ、インプットキャプチャレジスタとして設定可能

ウォッチドッグタイマ
・割り込み出力またはリセット可能なウォッチドッグタイマ

リアルタイムクロック
・年、月、日、時間、分、秒のサポート
・1/128 – 1/2　までの内部タイマタップを割り込みソースとしても利用可
・アラーム機能および割り込みをサポート

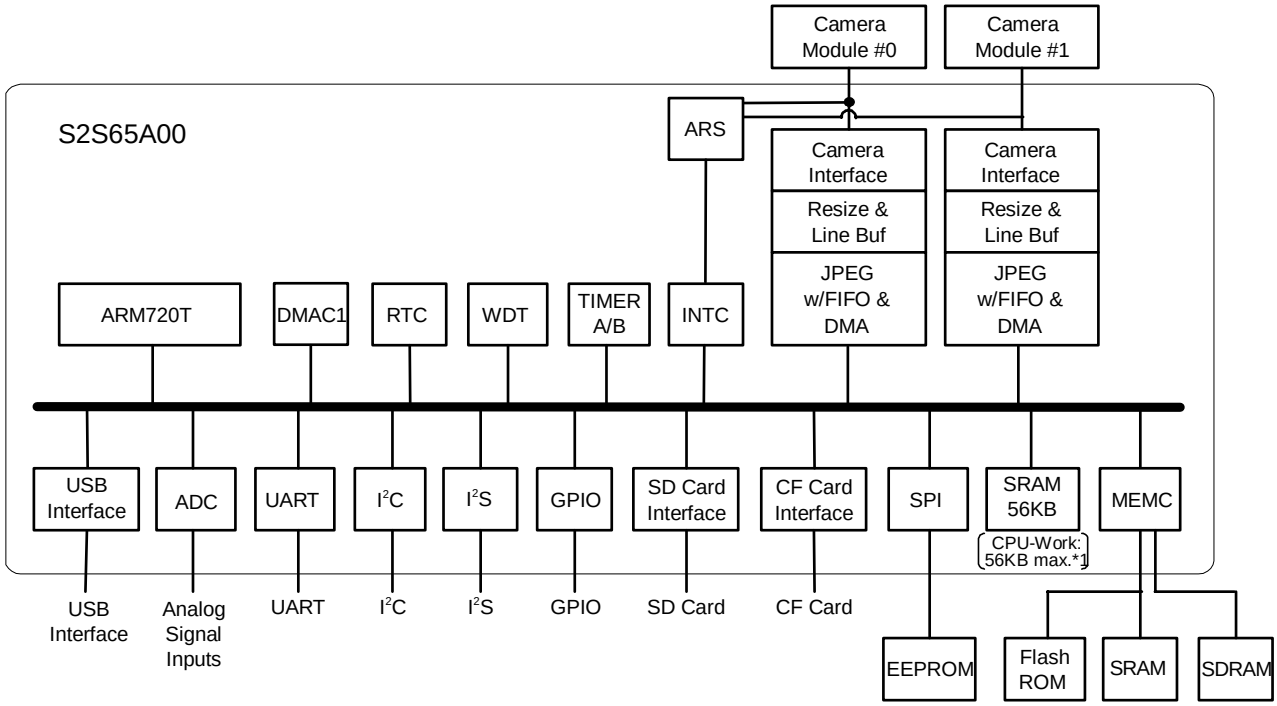
GPIO：
・汎用 I/O Port（最大 82 本）
・すべてのポートに対して方向がプログラマブル設定可能
・一部は他の I/O 機能と選択

A/D コンバータ：
・8-Channel Input(コントローラによるセレクト)
・10-bit 分解能
・1 回あたりの変換処理時間は 20µs 以下

電源：
・3.3V（I/O 電源）
・3.3V（USB）
・1.8V（コア電源）
・2.4～3.6V（カメラ 1/2 I/O 電源）
・2.7～3.6V（SDRAM I/O 電源）
・3.3V（A/D 電源）
・1.8V（USB/PLL/RTC）

パッケージ：
・PFBGA 280 Pin（PFBGA16UX280）　16 x 16 x 1.2 mm, 0.8mm ボールピッチ

2. ブロック図



*1 ：内蔵 SRAM は JPG[2:1]のラインバッファと共有しています。JPG[2:1]を使用する場合は CPU-Work として使用することは出来ません。

図 2.1 S2S65A00 内部ブロック図

3. 端子

3.1 端子配置図

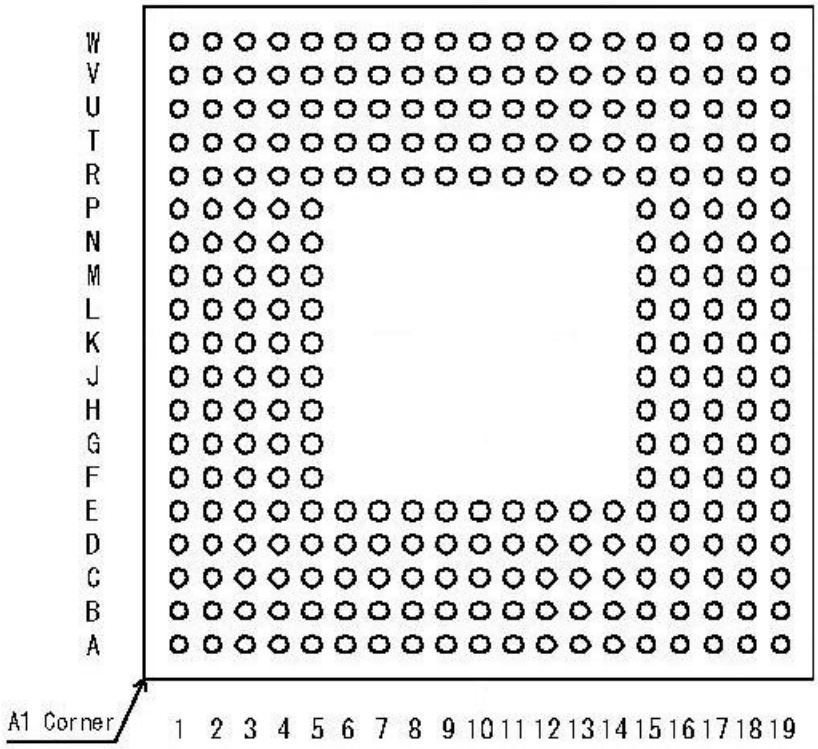


図 3.1　端子配置図 (BOTTOM VIEW)

ピン No.	端子名称	ピン No.	端子名称	ピン No.	端子名称	ピン No.	端子名称
A1	NC	D14	HVDD	K15	LVDD	T7	CM2DATA2
A2	CM1CLKIN	D15	GPIOA2	K16	CFDEN#	T8	GPIOD2
A3	CM1CLKOUT	D16	GPIOA4	K17	CFDDIR	T9	GPIOD1
A4	C1VDD	D17	GPIOA0	K18	LVDD	T10	MA17
A5	SYS_OSCO	D18	LVDD	K19	CFSTSCHG#	T11	MA15
A6	SYS_OSCI	D19	GPIOA1	L1	VSS	T12	MA11
A7	RTCVDD	E1	LVDD	L2	GPIOJ6	T13	MA7
A8	PLLVSS	E2	VSS	L3	GPIOJ3	T14	VSS
A9	SYSVCP	E3	SDA6	L4	GPIOJ5	T15	MA2
A10	SYSCLKI	E4	VSS	L5	LVDD	T16	MCS1#
A11	LVDD	E5	SDA0	L15	HVDD	T17	MD13
A12	VSS	E6	GPIOK5	L16	CFWAIT#	T18	MD12
A13	UXVDD	E7	GPIOK4	L17	CFRST	T19	MD10
A14	DP	E8	VSS	L18	VSS	U1	SDD0
A15	DM	E9	CM1DATA0	L19	CFIRQ	U2	SDDQM1#
A16	UVDD3	E10	TESTEN1	M1	GPIOJ0	U3	CM2HREF
A17	R1	E11	VSS	M2	GPIOJ2	U4	VSS
A18	UPVDD	E12	LVDD	M3	GPIOJ1	U5	CM2DATA4
A19	NC	E13	TDI	M4	SDD15	U6	AVSS
B1	SDA13	E14	GPIOB3	M5	VSS	U7	CM2DATA5
B2	SDA14	E15	GPIOB1	M15	CFCE2#	U8	CM2DATA7
B3	CM1HREF	E16	GPIOB0	M16	MD3	U9	GPIOD3

3. 端子

ピン No.	端子名称	ピン No.	端子名称	ピン No.	端子名称	ピン No.	端子名称
B4	VSS	E17	GPIOA3	M17	CFCE1#	U10	GPIOD0
B5	CM1DATA3	E18	HVDD	M18	CFIORD#	U11	MA16
B6	CM1DATA6	E19	GPIOA5	M19	CFIOWR#	U12	MA12
B7	VSS	F1	SDA1	N1	SDD11	U13	MA8
B8	BUP#	F2	SDVDD	N2	VSS	U14	MA4
B9	PLLVDD	F3	SDA5	N3	SDD13	U15	MA1
B10	SYSCKSEL	F4	SDA2	N4	SDD14	U16	VSS
B11	TRST#	F5	GPIOK1	N5	SDD12	U17	MBEL#
B12	TESTCK	F15	LVDD	N15	MD5	U18	MD14
B13	XVSS	F16	GPIOB2	N16	MD1	U19	MD15
B14	UVDD3	F17	GPIOA6	N17	VSS	V1	GPIOD6
B15	UVSS	F18	LVDD	N18	MD2	V2	VSS
B16	VSS	F19	GPIOA7	N19	MD0	V3	CM2VREF
B17	UVSS	G1	GPIOK7	P1	SDVDD	V4	CM2DATA0
B18	PVSS	G2	GPIOK6	P2	SDD10	V5	AVDD
B19	USBCK_OSCO	G3	GPIOK3	P3	SDD9	V6	ADIN7
C1	SDA11	G4	GPIOK0	P4	SDD8	V7	ADIN5
C2	VSS	G5	VSS	P5	VSS	V8	ADIN3
C3	SDVDD	G15	GPIOC1	P15	MD9	V9	ADIN1
C4	SDA12	G16	HVDD	P16	MD8	V10	AVSS
C5	CM1VREF	G17	GPIOB5	P17	MD6	V11	MA19
C6	CM1DATA5	G18	VSS	P18	MD4	V12	MA14
C7	CM1DATA7	G19	GPIOB4	P19	LVDD	V13	VSS
C8	VSS	H1	LVDD	R1	SDD6	V14	MA3
C9	HVDD	H2	VSS	R2	SDD5	V15	GPIOD5
C10	VSS	H3	GPIOK2	R3	SDD4	V16	MCS0#
C11	TESTEN0	H4	SDWE#	R4	SDD7	V17	MWE#
C12	TDO	H5	SDCAS#	R5	SDD3	V18	LVDD
C13	VSS	H15	GPIOC6	R6	SDD2	V19	VSS
C14	VBUS	H16	GPIOC0	R7	CM2DATA1	W1	NC
C15	VSS	H17	GPIOB6	R8	CM2DATA6	W2	CM2CLKIN
C16	HVDD	H18	HVDD	R9	VSS	W3	CM2CLKOUT
C17	VSS	H19	GPIOB7	R10	HVDD	W4	C2VDD
C18	VSS	J1	SDCS1#	R11	MA13	W5	AVDD
C19	USBCK_OSCI	J2	SDCS0#	R12	MA9	W6	ADIN6
D1	SDA8	J3	SDRAS#	R13	MA6	W7	ADIN4
D2	SDA9	J4	GPIOJ7	R14	MA5	W8	ADIN2
D3	SDA7	J5	GPIOJ4	R15	HVDD	W9	ADIN0
D4	SDA10	J15	GPIOC7	R16	GPIOD4	W10	AVDD
D5	SDA4	J16	GPIOC5	R17	MD11	W11	MA18
D6	SDA3	J17	GPIOC4	R18	VSS	W12	HVDD
D7	CM1DATA1	J18	GPIOC3	R19	MD7	W13	MA10
D8	CM1DATA2	J19	GPIOC2	T1	VSS	W14	LVDD
D9	CM1DATA4	K1	SDCLK	T2	LVDD	W15	HVDD
D10	LVDD	K2	SDCLKEN	T3	SDD1	W16	MA0
D11	TCK	K3	SDVDD	T4	GPIOD7	W17	MOE#
D12	TMS	K4	SDVDD	T5	SDDQM0#	W18	MBEH#
D13	RESET#	K5	VSS	T6	CM2DATA3	W19	NC

注意：端子名称右端の＃は、ローアクティブ信号であることを示しています。

3.2 端子の説明

＃　：端子名称右端の＃は、そのローアクティブ信号であることを示しています。
I　：入力ピン
O　：出力ピン
IO　：双方向ピン
P　：電源

表 3.1　Cell Type の説明

Cell Type	説明	使用端子例	
		端子名	使用電源電圧
ILS	Low Voltage LVCMOS Schmitt input	BUP#	RTCVDD
ICD1	LVCMOS input with pull-down resistor (50kΩ@3.3V)	TESTEN[1:0],TESTCK	HVDD
ICU1	LVCMOS input with pull-up resistor (50kΩ@3.3V)	TMS, TDI	HVDD
ICSU1	LVCMOS Schmitt input with pull-up resistor (50kΩ@3.3V)	TRST#,TCK,RESET#	HVDD
ICSD1	LVCMOS Schmitt input with pull-down resistor (50kΩ@3.3V)	SYSCKSEL	HVDD
ILTR	Low Voltage Transparent Input	SYS_OSCI	RTCVDD
		USBCK_OSCI	UPVDD
		R1	UVDD3
IHTR	High Voltage Transparent Input	ADIN[7:0]	ADVDD
OLTR	Low Voltage Transpatent Output	SYS_OSCO,SYSVCP	PLLVDD
		USBCK_OSCO	UPVDD
BLNC4U1	Low noise LVCMOS IO buffer with pull-up resistor (50kΩ@3.3V) (±4mA)	CF I/F	HVDD
BLNC4D2	Low noise LVCMOS IO buffer with pull-down resistor (100kΩ@3.3V) (±4mA)	MD [15:0]	HVDD
		SDD[15:0],GPIOJ,GPIOK	SDVDD
BLNS4	Low noise LVCMOS Schmitt IO buffer (±4mA)	GPIOA, GPIOB, GPIOC,GPIOD, SYSCLKI	HVDD
BLNS4D1	Low noise LVCMOS Schmitt IO buffer with pull-down resistor (50kΩ@3.3V) (±4mA)	Camera I/F	C1VDD, C2VDD
OLN4	Low noise output buffer (±4mA)	SRAM デバイス I/F (MD を除く)	HVDD
		SDRAM I/F(SDD[15:0]を除く)	SDVDD
OTLN4	Low noise Tri-state output buffer (±4mA)	TDO	HVDD
USBDM	USB DM buffer	DM	UVDD3
USBDP	USB DP buffer	DP	UVDD3
USBVBUS	USB VBUS output buffer	VBUS	UVDD3

表 3.2　端子の説明

端子名称	Type	Cell Type	ピン No.	説明
(MA [23:20])	(I/O)	(BLNS4)		これらの端子については **GPIOD[3:0]**の説明を参照してください。
MA [19:12]	O	OLN4	V11,W11,T10, U11,T11,V12, R11,U12	**Flash-ROM/SRAM 用アドレス出力信号[19:12]**
MA 11	O	OLN4	T12	この端子は以下の機能を持っています。 **・MA11：Flash-ROM/SRAM 用アドレス出力信号 11** （リセット直後の端子機能） **・CFREG#出力信号** コンパクトフラッシュ(CF)インタフェース動作時は、CF インタフェースのアトリビュートおよび I/O 空間を選択する REG 信号として動作します。
MA [10:0]	O	OLN4	W13,R12,U13, T13,R13,R14, U14,V14,T15, U15,W16	これらの端子は以下の機能を持っています。 **・MA[10:0]：Flash-ROM/SRAM 用アドレス出力信号[10:0]**（リセット直後の端子機能） **・CFADDR[10:0]出力信号** CF インタフェース動作時は、CF インタフェースのアドレス信号[10:0]となります。
MBEL#	I/O	OLN4	U17	**Flash-ROM/SRAM** 用データバス下位バイト・イネーブル出力
MBEH#	I/O	OLN4	W18	**Flash-ROM/SRAM** 用データバス上位バイト・イネーブル出力
MD [15:0]	I/O	BLNC4D2	U19,U18,T17, T18,R17,T19, P15,P16,R19, P17,N15,P18, M16,N18,N16, N19	これらの端子は以下の機能を持っています。 **・16Bit の Flash-ROM/SRAM 用 Data Bus**（リセット直後の端子機能） **・CF インタフェース動作時は 16Bit データ**となります。 **・MODESEL[15:0]** パワーオンリセット時（RESET#の Low から High への遷移時）に内部の動作モードを決めるためにサンプリングされます。“4.1 システムコンフィギュレーション”の項を参照してください。 このとき、動作モードを決めるために外部でプルアップ抵抗が必要になる場合があります。（4.7～10kΩ程度の抵抗）
MCS [3:2]#	O	(BLNS4)	V15,R16	これらの端子については **GPIOD[5:4]**の説明を参照してください。
MCS [1:0]#	O	OLN4	T16,V16	**Flash-ROM/SRAM 用チップセレクト信号[1:0]** （ローアクティブ信号）
MOE#	O	OLN4	W17	この端子は以下の機能を持っています。 （ローアクティブ信号） **・MOE#：Flash-ROM/SRAM 用ストローブ信号**（リセット直後の端子機能） **・CFOE#出力信号** CF インタフェース動作時は、CF インタフェースのメモリおよびアトリビュート空間の出力イネーブル信号となります。

端子名称	Type	Cell Type	ピン No.	説明
MWE#	O	OLN4	V17	この端子は以下の機能を持っています。 （ローアクティブ信号） ・**MWE# : Flash-ROM/SRAM 用 Write Enable 信号**（スタティックメモリ用）（リセット直後の端子機能） ・**CFWE#出力信号** CF インタフェース動作時は、CF インタフェースのメモリおよびアトリビュート空間のライトイネーブル信号となります。
SDA [14:0]	O	OLN4	B2,B1,C4, C1,D4,D2, D1,D3,E3, F3,D5,D6, F4,F1,E5	**SDRAM 用アドレス出力[14:0]**
SDD[31:16]	(I/O)	(BLNC4D2)		これらの端子については **GPIOK[7:0],GPIOJ[7:0]**の説明を参照してください。
SDD [15:0]	I/O	BLNC4D2	M4,N4,N3, N5,N1,P2, P3,P4,R4, R1,R2,R3, R5,R6,T3, U1	**SDRAM 用データ入出力[15:0]**
SDWE#	O	OLN4	H4	**SDRAM 用 Write Enable 信号**
SDCLK	O	OLN4	K1	**SDRAM 用クロック出力** 内部動作周波数（CPUCLK）と同じ周波数が出力されます。
SDCLKEN	O	OLN4	K2	**SDRAM 用クロック・イネーブル信号**
SDRAS#	O	OLN4	J3	**SDRAM 用 RAS 信号**（ローアクティブ信号）
SDCAS#	O	OLN4	H5	**SDRAM 用 CAS 信号**（ローアクティブ信号）
SDCS[1:0]#	O	OLN4	J1,J2	**SDRAM 用チップセレクト信号**（ローアクティブ信号）
SDDQM[3:2]#	(I/O)	(BLNS4)		これらの端子については **GPIOD[7:6]**の説明を参照してください。
SDDQM[1:0]#	O	OLN4	U2,T5	**SDRAM 用 DQM 信号**（ローアクティブ信号） SDDQM0#が下位バイトに対応し、SDDQM1#が上位バイトに対応します。
CM1DATA[7:0]	I/O	BLNS4D1	C7,B6,C6, D9,B5,D8, D7,E9	これらの端子は以下の機能を持っています。 ・**CM1DATA[7:0]：カメラ 1YUV データ入力** これらの端子は、リセット時は GPIOE[7:0]入力になっています。CM1DATA[7:0]端子として使用するためには GPIOE 端子機能レジスタのビット[15:0]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOE[7:0]入出力**（リセット直後の端子機能）
CM1VREF	I/O	BLNS4D1	C5	この端子は以下の機能を持っています。 ・**CM1VREF：カメラ 1 データ入力時の垂直同期入力** この端子は、リセット時は GPIOF0 入力になっています。CM1VREF 端子として使用するためには GPIOF 端子機能レジスタのビット[1:0]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOF0 入出力**（リセット直後の端子機能）
CM1HREF	I/O	BLNS4D1	B3	この端子は以下の機能を持っています。 ・**CM1HREF：カメラ 1 データ入力時の水平同期入力** この端子は、リセット時は GPIOF1 入力になっています。CM1HREF 端子として使用するためには GPIOF 端子機能レジスタのビット[3:2]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOF1 入出力**（リセット直後の端子機能）

端子名称	Type	Cell Type	ピン No.	説明
CM1CLKOUT	I/O	BLNS4D1	A3	この端子は以下の機能を持っています。 **・CM1CLKOUT：カメラ 1 用基本クロック出力** この端子は、リセット時は GPIOF2 入力になっています。CM1CLKOUT 端子として使用するためには GPIOF 端子機能レジスタのビット[5:4]を“GPIO 以外の機能 1”に設定してください。 **・GPIOF2 入出力**（リセット直後の端子機能）
CM1CLKIN	I/O	BLNS4D1	A2	この端子は以下の機能を持っています。 **・CM1CLKIN：カメラ 1 データ入力用ピクセルクロック** この端子は、リセット時は GPIOF3 入力になっています。CM1CLKIN 端子として使用するためには GPIOF 端子機能レジスタのビット[7:6]を“GPIO 以外の機能 1”に設定してください。 **・GPIOF3 入出力**（リセット直後の端子機能）
CM2DATA[7:0]	I/O	BLNS4D1	U8,R8,U7, U5,T6,T7, R7,V4	これらの端子は以下の機能を持っています。 **・CM2DATA[7:0]：カメラ 2YUV データ入力** これらの端子は、リセット時は GPIOG[7:0]入力になっています。CM2DATA[7:0]端子として使用するためには GPIOG 端子機能レジスタのビット[15:0]を“GPIO 以外の機能 1”に設定してください。 **・GPIOG[7:0]入出力**（リセット直後の端子機能）
CM2VREF	I/O	BLNS4D1	V3	この端子は以下の機能を持っています。 **・CM2VREF：カメラ 2 データ入力時の垂直同期入力** この端子は、リセット時は GPIOF4 入力になっています。CM2VREF 端子として使用するためには GPIOF 端子機能レジスタのビット[9:8]を“GPIO 以外の機能 1”に設定してください。 **・GPIOF4 入出力**（リセット直後の端子機能）
CM2HREF	I/O	BLNS4D1	U3	この端子は以下の機能を持っています。 **・CM2HREF：カメラ 2 データ入力時の水平同期入力** この端子は、リセット時は GPIOF5 入力になっています。CM2HREF 端子として使用するためには GPIOF 端子機能レジスタのビット[11:10]を“GPIO 以外の機能 1”に設定してください。 **・GPIOF5 入出力**（リセット直後の端子機能）
CM2CLKOUT	I/O	BLNS4D1	W3	この端子は以下の機能を持っています。 **・CM2CLKOUT：カメラ 2 用基本クロック出力** この端子は、リセット時は GPIOF6 入力になっています。CM2CLKOUT 端子として使用するためには GPIOF 端子機能レジスタのビット[13:12]を“GPIO 以外の機能 1”に設定してください。 **・GPIOF6 入出力**（リセット直後の端子機能）
CM2CLKIN	I/O	BLNS4D1	W2	この端子は以下の機能を持っています。 **・CM2CLKIN：カメラ 2 データ入力用ピクセルクロック** この端子は、リセット時は GPIOF7 入力になっています。CM2CLKIN 端子として使用するためには GPIOF 端子機能レジスタのビット[15:14]を“GPIO 以外の機能 1”に設定してください。 **・GPIOF7 入出力**（リセット直後の端子機能）

端子名称	Type	Cell Type	ピン No.	説明
CFCE2#	I/O	BLNC4U1	M15	この端子は以下の機能を持っています。 ・**CFCE2#：Compact Flash Memory Interface 用（以下 CF 用）カードイネーブル 2(CE2#)出力**（ローアクティブ信号） この端子は、リセット時は GPIOH0 入力になっています。CFCE2#端子として使用するためには GPIOH 端子機能レジスタのビット[1:0]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH0 入出力**（リセット直後の端子機能） ・**SDMDATA0：SD カード用データ入出力 0**（GPIO 以外の端子機能 2）
CFCE1#	I/O	BLNC4U1	M17	この端子は以下の機能を持っています。 ・**CFCE1#：CF 用カードイネーブル 1（CE1#）出力**（ローアクティブ信号） この端子は、リセット時は GPIOH1 入力になっています。CFCE1#端子として使用するためには GPIOH 端子機能レジスタのビット[3:2]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH1 入出力**（リセット直後の端子機能） ・**SDMDATA1：SD カード用データ入出力 1**（GPIO 以外の端子機能 2）
CFIORD#	I/O	BLNC4U1	M18	この端子は以下の機能を持っています。 ・**CFIORD#：CF 用 IO Read ストローブ出力**（ローアクティブ信号） この端子は、リセット時は GPIOH2 入力になっています。CFIORD#端子として使用するためには GPIOH 端子機能レジスタのビット[5:4]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH2 入出力**（リセット直後の端子機能） ・**SDMDATA2：SD カード用データ入出力 2**（GPIO 以外の端子機能 2）
CFIOWR#	I/O	BLNC4U1	M19	この端子は以下の機能を持っています。 （ローアクティブ信号） ・**CFIOWR#：CF 用 IO Write ストローブ出力** この端子は、リセット時は GPIOH3 入力になっています。CFIOWR#端子として使用するためには GPIOH 端子機能レジスタのビット[7:6]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH3 入出力**（リセット直後の端子機能） ・**SDMDATA3：SD カード用データ入出力 3**（GPIO 以外の端子機能 2）
CFWAIT#	I/O	BLNC4U1	L16	この端子は以下の機能を持っています。 ・**CFWAIT#：CF 用ウエイト要求力**（ローアクティブ信号） この端子は、リセット時は GPIOH4 入力になっています。CFWAIT#端子として使用するためには GPIOH 端子機能レジスタのビット[9:8]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH4 入出力**（リセット直後の端子機能） ・**SDMCMD：SD カード用コマンド入出力**（GPIO 以外の機能 2）

端子名称	Type	Cell Type	ピン No.	説明
CFRST	I/O	BLNC4U1	L17	この端子は以下の機能を持っています。 ・**CFRST**：**CF カードへのリセット信号** カードリセット時に HIGH、カードの通常動作時に LOW となります。 この端子は、リセット時は GPIOH5 入力になっています。CFRST 端子として使用するためには GPIOH 端子機能レジスタのビット[11:10]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH5 入出力**（リセット直後の端子機能） ・**SDMCLK**：**SD カード用クロック出力**（GPIO 以外の機能 2）
CFIRQ	I/O	BLNC4U1	L19	この端子は以下の機能を持っています。 ・**CFIRQ**：**CF カードからの割り込み要求信号** この端子は、リセット時は GPIOH6 入力になっています。CFIRQ 端子として使用するためには GPIOH 端子機能レジスタのビット[13:12]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH6 入出力**（リセット直後の端子機能） ・**SDMCD#**：**SD カード用カードディテクト入力**（GPIO 以外の機能 2）
CFSTSCHG#	I/O	BLNC4U1	K19	この端子は以下の機能を持っています。 ・**CFSTSCHG#**：**CF カードからのステータスチェンジ信号**（ローアクティブ信号） この端子は、リセット時は GPIOH7 入力になっています。CFSTSCHG#端子として使用するためには GPIOH 端子機能レジスタのビット[15:14]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOH7 入出力**（リセット直後の端子機能） ・**SDMWP**：**SD カード用ライトプロテクト入力**（GPIO 以外の機能 2）
CFDEN#	I/O	BLNC4U1	K16	この端子は以下の機能を持っています。 ・**CFDEN#**：**CF カードの外部バッファ用データバスイネーブル信号**（ローアクティブ信号） この端子は、リセット時は GPIOI0 入力になっています。CFDEN#端子として使用するためには GPIOI 端子機能レジスタのビット[1:0]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOI0 入出力端子**（リセット直後の端子機能） ・**SDMGPO**：**SD カード用汎用出力**（GPIO 以外の機能 2）
CFDDIR	I/O	BLNC4U1	K17	この端子は以下の機能を持っています。 ・**CFDDIR**：**CF 用データバス方向指示出力** この端子は、CF 用データのリード時に LOW となります。また、この端子はリセット時には GPIOI1 入力になっています。CFDDIR 端子として使用するためには GPIOI 端子機能レジスタのビット[3:2]を“GPIO 以外の機能 1”に設定してください。 ・**GPIOI1 入出力**（リセット直後の端子機能）
R1	I	ILTR	A17	USB Device 内部動作設定端子
DM	I/O	USBDM	A15	USB Device D- 入出力
DP	I/O	USBDP	A14	USB Device D+ 入出力
VBUS	I	USBVBUS	C14	USB Device VBUS 入力

端子名称	Type	Cell Type	ピン No.	説明
ADIN[7:0]	I	IHTR	V6,W6, V7,W7, V8,W8, V9,W9	アナログ信号入力
GPIOA0	I/O	BLNS4	D17	この端子は以下の機能を持っています。 ・**GPIOA0 入出力**（リセット直後の端子機能） ・**TXD1 : UART1 送信データ出力**（GPIO 以外の機能 1）
GPIOA1	I/O	BLNS4	D19	この端子は以下の機能を持っています。 ・**GPIOA1 入出力**（リセット直後の端子機能） ・**RXD1 : UART1 受信データ入力**（GPIO 以外の機能 1）
GPIOA2	I/O	BLNS4	D15	この端子は以下の機能を持っています。 ・**GPIOA2 入出力**（リセット直後の端子機能） ・**RTS1 : UART1 送信要求出力**（GPIO 以外の機能 1） ・**I2S1_WS : I2S1 用ワードセレクト**（GPIO 以外の機能 2）
GPIOA3	I/O	BLNS4	E17	この端子は以下の機能を持っています。 ・**GPIOA3 入出力**（リセット直後の端子機能） ・**CTS1 : UART1 送信可能入力**（GPIO 以外の機能 1） ・**I2S1_SCK : I2S1 用シリアルクロック**（GPIO 以外の機能 2）
GPIOA4	I/O	BLNS4	D16	この端子は以下の機能を持っています。 ・**GPIOA4 入出力**（リセット直後の端子機能） ・**TXD2 : UART2 送信データ出力**（GPIO 以外の機能 1）
GPIOA5	I/O	BLNS4	E19	この端子は以下の機能を持っています。 ・**GPIOA5 入出力**（リセット直後の端子機能） ・**RXD2 : UART2 受信データ入力**（GPIO 以外の機能 1）
GPIOA6	I/O	BLNS4	F17	この端子は以下の機能を持っています。 ・**GPIOA6 入出力**（リセット直後の端子機能） ・**RTS2 : UART2 送信要求出力**（GPIO 以外の機能 1） ・**SCL : I2C クロック入出力**（GPIO 以外の機能２）
GPIOA7	I/O	BLNS4	F19	この端子は以下の機能を持っています。 ・**GPIOA7 入出力**（リセット直後の端子機能） ・**CTS2 : UART2 受信可能入力**（GPIO 以外の機能 1） ・**SDA : I2C データ入出力**（GPIO 以外の機能２）
GPIOB0	I/O	BLNS4	E16	この端子は以下の機能を持っています。 ・**GPIOB0 入出力**（リセット直後の端子機能） ・**INT0 入力** ・**I2S0_WS : I2S0 用ワードセレクト**（GPIO 以外の機能 1）
GPIOB1	I/O	BLNS4	E15	この端子は以下の機能を持っています。 ・**GPIOB1 入出力**（リセット直後の端子機能） ・**INT1 入力** ・**I2S0_SCK : I2S0 用シリアルクロック**（GPIO 以外の機能 1）
GPIOB2	I/O	BLNS4	F16	この端子は以下の機能を持っています。 ・**GPIOB2 入出力**（リセット直後の端子機能） ・**INT2 入力** ・**I2S0_SD : I2S0 用シリアルデータ**（GPIO 以外の機能 1）
GPIOB3	I/O	BLNS4	E14	この端子は以下の機能を持っています。 ・**GPIOB3 入出力**（リセット直後の端子機能） ・**INT3 入力** ・**I2S1_SD : I2S1 用シリアルデータ**（GPIO 以外の機能 1）

端子名称	Type	Cell Type	ピン No.	説明
GPIOB4	I/O	BLNS4	G19	この端子は以下の機能を持っています。 ・**GPIOB4 入出力**（リセット直後の端子機能） ・**INT4 入力** ・**TimerA0Out**（GPIO 以外の機能 1）
GPIOB5	I/O	BLNS4	G17	この端子は以下の機能を持っています。 ・**GPIOB5 入出力**（リセット直後の端子機能） ・**INT5 入力** ・**TimerA1Out**（GPIO 以外の機能 1）
GPIOB6	I/O	BLNS4	H17	この端子は以下の機能を持っています。 ・**GPIOB6 入出力**（リセット直後の端子機能） ・**INT6 入力** ・**TimerA2Out**（GPIO 以外の機能 1）
GPIOB7	I/O	BLNS4	H19	この端子は以下の機能を持っています。 ・**GPIOB7 入出力**（リセット直後の端子機能） ・**INT7 入力** ・**TimerBIn**（GPIO 以外の機能 1）
GPIOC0	I/O	BLNS4	H16	この端子は以下の機能を持っています。 ・**GPIOC0 入出力**（リセット直後の端子機能） ・**TimerB0IO**（GPIO 以外の機能 1）
GPIOC1	I/O	BLNS4	G15	この端子は以下の機能を持っています。 ・**GPIOC1 入出力**（リセット直後の端子機能） ・**TimerB1IO**（GPIO 以外の機能 1）
GPIOC2	I/O	BLNS4	J19	この端子は以下の機能を持っています。 ・**GPIOC2 入出力**（リセット直後の端子機能） ・**TimerB2IO**（GPIO 以外の機能 1）
GPIOC3	I/O	BLNS4	J18	この端子は以下の機能を持っています。 ・**GPIOC3 入出力**（リセット直後の端子機能） ・**TimerB3IO**（GPIO 以外の機能 1） ・**UART3_CLK**（GPIO 以外の機能 2）
GPIOC4	I/O	BLNS4	J17	この端子は以下の機能を持っています。 ・**GPIOC4 入出力**（リセット直後の端子機能） ・**SPI_SS : SPI 用チップセレクト**（GPIO 以外の機能 1） ・**TXD3 : UART3 送信データ出力**（GPIO 以外の機能 2）
GPIOC5	I/O	BLNS4	J16	この端子は以下の機能を持っています。 ・**GPIOC5 入出力**（リセット直後の端子機能） ・**SPI_SCLK : SPI 用シリアルクロック**（GPIO 以外の機能 1） ・**RXD3 : UART3 受信データ入力**（GPIO 以外の機能 2）
GPIOC6	I/O	BLNS4	H15	この端子は以下の機能を持っています。 ・**GPIOC6 入出力**（リセット直後の端子機能） ・**SPI_MISO:SPI 用シリアルデータマスタ入力／スレーブ出力**（GPIO 以外の機能 1） ・**RTS3 : UART3 送信要求出力**（GPIO 以外の機能 2）
GPIOC7	I/O	BLNS4	J15	この端子は以下の機能を持っています。 ・**GPIOC6 入出力**（リセット直後の端子機能） ・**SPI_MOSI:SPI 用シリアルデータマスタ出力／スレーブ入力**（GPIO 以外の機能 1） ・**CTS3 : UART3 受信可能入力**（GPIO 以外の機能 2）
GPIOD[3:0]	I/O	BLNS4	U9,T8, T9,U10	この端子は以下の機能を持っています。 ・**GPIOD [3:0]入出力**（リセット直後の端子機能） ・**MA [23:20] : アドレス出力信号 [23:20]**（GPIO 以外の機能 1）

端子名称	Type	Cell Type	ピン No.	説明
GPIOD[5:4]	I/O	BLNS4	V15,R16	この端子は以下の機能を持っています。 ・**GPIOD[5:4]入出力**（リセット直後の端子機能） ・**MCS[3:2]# : メモリ用チップセレクト出力信号**（GPIO 以外の機能 1）
GPIOD6	I/O	BLNS4	V1	この端子は以下の機能を持っています。 ・**GPIOD6 入出力**（リセット直後の端子機能） ・**SDRAM 用 DQM2#信号**（ローアクティブ信号） ・**SDDQM2# : SDRAM32 ビットデータ幅の上位 16 ビットの下位バイトに対応**（GPIO 以外の機能 1）
GPIOD7	I/O	BLNS4	T4	この端子は以下の機能を持っています。 ・**GPIOD7 入出力**（リセット直後の端子機能） ・**SDRAM 用 DQM3#信号**（ローアクティブ信号） ・**SDDQM3# : SDRAM32 ビットデータ幅の上位 16 ビットの上位バイトに対応**（GPIO 以外の機能 1）
GPIOJ[7:0]	I/O	BLNC4D2	J4,L2, L4,J5, L3,M2, M3,M1	この端子は以下の機能を持っています。 ・**GPIOJ[7:0]入出力**（リセット直後の端子機能） ・**SDD[23:16] : SDRAM 用データ入出力**（GPIO 以外の機能 1）
GPIOK[7:0]	I/O	BLNC4D2	G1,G2, E6,E7, G3,H3, F5,G4	この端子は以下の機能を持っています。 ・**GPIOK[7:0]入出力**（リセット直後の端子機能） ・**SDD[31:24] : SDRAM 用データ入出力**（GPIO 以外の機能 1）
SYSCLKI	I/O	BLNS4	A10	**32KHz システムクロック入力** SYSCKSEL="High"の時、基本クロック入力。 SYSCKSEL="Low"の時、設定により 32KHz またはシステムクロックの 1/4 が出力。
SYS_OSCI	I	ILTR	A6	**水晶振動子用接続端子** 動作クロック発振端子です。32KHz の水晶振動子を接続します。
SYS_OSCO	O	OLTR	A5	**水晶振動子用接続端子** 動作クロック発振端子です。32KHz の水晶振動子を接続します。
SYSVCP	O	OLTR	A9	**内蔵 PLL 用システムテスト端子** システムのテスト時に PLL の出力をモニタするために使用します。 通常使用時はオープンにしてください。
SYSCKSEL	I	ICSD1	B10	**32KHz システムクロック入力の水晶振動子/発振器の選択信号** SYSCKSEL="High"の時、水晶発振器を使用。 SYSCKSEL="Low"の時、水晶振動子を使用。
USBCK_OSCI	I	ILTR	C19	**水晶振動子用接続端子** USB 専用の動作クロック発振端子です。12/24MHz の水晶振動子を接続します。
USBCK_OSCO	O	OLTR	B19	**水晶振動子用接続端子** USB 専用の動作クロック発振端子です。12/24MHz の水晶振動子を接続します。
TRST#	I	ICSU1	B11	**JTAG Interface 用リセット**（ローアクティブ信号） プルアップ抵抗付シュミットトリガ入力になっています。
TCK	I	ICSU1	D11	**JTAG Interface 用クロック入力ピン** シュミットトリガ入力になっています。
TMS	I	ICU1	D12	**JTAG Interface 用 TMS ピン** この端子はプルアップ抵抗を内蔵しています。
TDI	I	ICU1	E13	**JTAG Interface 用シリアルデータ入力ピン** この端子はプルアップ抵抗を内蔵しています。

端子名称	Type	Cell Type	ピン No.	説明
TDO	O	OTLN4	C12	**JTAG Interface 用シリアルデータ出力ピン**
TESTEN0	I	ICD1	C11	**テストイネーブル 0**（ハイアクティブ信号） この端子はプルダウン抵抗を内蔵しています。 通常使用時は VSS へ接続するか、またはオープンにしてください。
TESTEN1	I	ICD1	E10	**テストイネーブル 1**（ハイアクティブ信号） この端子はプルダウン抵抗を内蔵しています。 通常使用時は VSS へ接続するか、またはオープンにしてください。
TESTCK	I	ICD1	B12	**テストクロック端子** この端子はプルダウン抵抗を内蔵しています。 通常使用時は VSS へ接続するか、またはオープンにしてください。
RESET#	I	ICSU1	D13	**システムリセット信号**（ローアクティブ信号） HVDD1 および LVDD が安定した後も 100ms の間 Reset#をアクティブ(Low) に保ってください。
BUP#	I	ILS	B8	**バックアップ信号**（ローアクティブ信号）
HVDD	P	P	C9,C16,D14, E18,G16,H18, L15,R10,R15, W12,W15	**I/O セル用電源**：3.3V(Typical) 3.0V (Min) – 3.6V (Max)
C1VDD	P	P	A4	**カメラ 1 インタフェース用電源**：3.0 (Typical) 2.4V (Min) – 3.6V (Max)
C2VDD	P	P	W4	**カメラ 2 インタフェース用電源**：3.0 (Typical) 2.4V (Min) – 3.6V (Max)
SDVDD	P	P	C3,F2, K3,K4, P1	**SDRAM 用電源** 3.3V(Typical) 2.7V (Min) – 3.6V (Max)
AVDD	P	P	V5,W5, W10	**A/D C 用電源**：3.3V(Typical) 3.0V (Min) – 3.6V (Max)
UVDD3	P	P	A16,B14	**USB 用電源**：3.3V(Typical) 3.0V (Min) – 3.6V (Max)
LVDD	P	P	A11,D10,D18, E1,E12,F15, F18,H1,K15, K18,L5,P19, T2,V18,W14	**コア（内部）用電源**：1.8V(Typical) 1.65V (Min) – 1.95V (Max)
UPVDD	P	P	A18	**USB 用電源**：1.8V(Typical) 1.65V (Min) – 1.95V (Max)
UXVDD	P	P	A13	**USB 用電源**：1.8V(Typical) 1.65V (Min) – 1.95V (Max)
PLLVDD	P	P	B9	**アナログ(PLL)用電源**：1.8V(Typical) 1.65V (Min) – 1.95V (Max) アナログ電源としての取り扱いが必要です。ノイズの少ない安定した電源を供給してください。
RTCVDD	P	P	A7	**RTC 用電源**：1.8V(Typical) 1.65V (Min) – 1.95V (Max)
UVSS	P	P	B15,B17	**USB 用グラウンド**
PVSS	P	P	B18	**USB 用グラウンド**
XVSS	P	P	B13	**USB 用グラウンド**
AVSS	P	P	U6,V10	**A/D C 用グラウンド**
PLLVSS	P	P	A8	**アナログ(PLL)用グラウンド** アナログ電源としての取り扱いが必要です。ノイズの少ない安定したグラウンドを供給してください。

端子名称	Type	Cell Type	ピン No.	説明
VSS	P	P	A12,B4,B7, B16,C2,C8, C10,C13,C15, C17,C18,E2, E4,E8,E11, G5,G18,H2, K5,L1,L18, M5,N2,N17, P5,R9,R18, T1,T14,U4, U16,V2,V13, V19	**I/O セル、カメラインタフェースおよびコア電源共通グラウンド**

3.3 GPIO 端子のマルチプレクス端子機能、リセット直後の端子機能

S2S65A00 端子名称	リセット直後の端子機能	GPIO	INT	FlashROM/ SAM 拡張	SDRAM 拡張	UART	I2C/ I2S/ SPI	タイマ	カメラ I/F	CF カード	SD カード
GPIOA0	GPIOA0	GPIOA0				TXD1					
GPIOA1	GPIOA1	GPIOA1				RXD1					
GPIOA2	GPIOA2	GPIOA2				RTS1	I2S1_WS				
GPIOA3	GPIOA3	GPIOA3				CTS1	I2S1_SCK				
GPIOA4	GPIOA4	GPIOA4				TXD2					
GPIOA5	GPIOA5	GPIOA5				RXD2					
GPIOA6	GPIOA6	GPIOA6				RTS2	SCL				
GPIOA7	GPIOA7	GPIOA7				CTS2	SDA				
GPIOB0	GPIOB0	GPIOB0	INT0				I2S0_WS				
GPIOB1	GPIOB1	GPIOB1	INT1				I2S0_SCK				
GPIOB2	GPIOB2	GPIOB2	INT2				I2S0_SD				
GPIOB3	GPIOB3	GPIOB3	INT3				I2S1_SD				
GPIOB4	GPIOB4	GPIOB4	INT4					TimerA0out			
GPIOB5	GPIOB5	GPIOB5	INT5					TimerA1out			
GPIOB6	GPIOB6	GPIOB6	INT6					TimerA2out			
GPIOB7	GPIOB7	GPIOB7	INT7					TimerBIN			
GPIOC0	GPIOC0	GPIOC0						TimerB0IO			
GPIOC1	GPIOC1	GPIOC1						TimerB1IO			
GPIOC2	GPIOC2	GPIOC2						TimerB2IO			
GPIOC3	GPIOC3	GPIOC3				UART3_CLK		TimerB3IO			
GPIOC4	GPIOC4	GPIOC4				TXD3	SPI_SS				
GPIOC5	GPIOC5	GPIOC5				RXD3	SPI_SCLK				
GPIOC6	GPIOC6	GPIOC6				RTS3	SPI_MISO				
GPIOC7	GPIOC7	GPIOC7				CTS3	SPI_MOSI				
GPIOD0	GPIOD0	GPIOD0		MA20							
GPIOD1	GPIOD1	GPIOD1		MA21							
GPIOD2	GPIOD2	GPIOD2		MA22							
GPIOD3	GPIOD3	GPIOD3		MA23							
GPIOD4	GPIOD4	GPIOD4		MCS2#							
GPIOD5	GPIOD5	GPIOD5		MCS3#							
GPIOD6	GPIOD6	GPIOD6			SDDQM2#						
GPIOD7	GPIOD7	GPIOD7			SDDQM3#						
CM1DATA0	GPIOE0	GPIOE0							CM1DATA0		
CM1DATA1	GPIOE1	GPIOE1							CM1DATA1		
CM1DATA2	GPIOE2	GPIOE2							CM1DATA2		
CM1DATA3	GPIOE3	GPIOE3							CM1DATA3		
CM1DATA4	GPIOE4	GPIOE4							CM1DATA4		
CM1DATA5	GPIOE5	GPIOE5							CM1DATA5		
CM1DATA6	GPIOE6	GPIOE6							CM1DATA6		
CM1DATA7	GPIOE7	GPIOE7							CM1DATA7		
CM1VREF	GPIOF0	GPIOF0							CM1VREF		
CM1HREF	GPIOF1	GPIOF1							CM1HREF		
CM1CLKOUT	GPIOF2	GPIOF2							CM1CLKOUT		
CM1CLKIN	GPIOF3	GPIOF3							CM1CLKIN		
CM2VREF	GPIOF4	GPIOF4							CM2VREF		
CM2HREF	GPIOF5	GPIOF5							CM2HREF		
CM2CLKOUT	GPIOF6	GPIOF6							CM2CLKOUT		
CM2CLKIN	GPIOF7	GPIOF7							CM2CLKIN		
CM2DATA0	GPIOG0	GPIOG0							CM2DATA0		
CM2DATA1	GPIOG1	GPIOG1							CM2DATA1		
CM2DATA2	GPIOG2	GPIOG2							CM2DATA2		
CM2DATA3	GPIOG3	GPIOG3							CM2DATA3		
CM2DATA4	GPIOG4	GPIOG4							CM2DATA4		
CM2DATA5	GPIOG5	GPIOG5							CM2DATA5		
CM2DATA6	GPIOG6	GPIOG6							CM2DATA6		
CM2DATA7	GPIOG7	GPIOG7							CM2DATA7		

S2S65A00 端子名称	リセット 直後の 端子機能	GPIO	INT	FlashROM/ SAM 拡張	SDRAM 拡張	UART/ SPI	I2C/ I2S	タイマ	カメラ I/F	CF カード	SD カード
CFCE2#	GPIOH0	GPIOH0								CFCE2#	SDMDATA0
CFCE1#	GPIOH1	GPIOH1								CFCE1#	SDMDATA1
CFIORD#	GPIOH2	GPIOH2								CFIORD#	SDMDATA2
CFIOWR#	GPIOH3	GPIOH3								CFIOWR#	SDMDATA3
CFWAIT#	GPIOH4	GPIOH4								CFWAIT#	SDMCMD
CFRST	GPIOH5	GPIOH5								CFRST	SDMCLK
CFIRQ	GPIOH6	GPIOH6								CFIRQ	SDMCD#
CFSTSCHG#	GPIOH7	GPIOH7								CFSTSCHG#	SDMWP
CFDEN#	GPIOI0	GPIOI0								CFDEN#	SDMGPO
CFDDIR	GPIOI1	GPIOI1								CFDDIR	
GPIOJ0	GPIOJ0	GPIOJ0			SDD16						
GPIOJ1	GPIOJ1	GPIOJ1			SDD17						
GPIOJ2	GPIOJ2	GPIOJ2			SDD18						
GPIOJ3	GPIOJ3	GPIOJ3			SDD19						
GPIOJ4	GPIOJ4	GPIOJ4			SDD20						
GPIOJ5	GPIOJ5	GPIOJ5			SDD21						
GPIOJ6	GPIOJ6	GPIOJ6			SDD22						
GPIOJ7	GPIOJ7	GPIOJ7			SDD23						
GPIOK0	GPIOK0	GPIOK0			SDD24						
GPIOK1	GPIOK1	GPIOK1			SDD25						
GPIOK2	GPIOK2	GPIOK2			SDD26						
GPIOK3	GPIOK3	GPIOK3			SDD27						
GPIOK4	GPIOK4	GPIOK4			SDD28						
GPIOK5	GPIOK5	GPIOK5			SDD29						
GPIOK6	GPIOK6	GPIOK6			SDD30						
GPIOK7	GPIOK7	GPIOK7			SDD31						

機能 1 ：機能 1　　機能 2 ：機能 2

3.4 リセット中の端子の状態

Pin Name	RESET 中の方向	RESET 中の値	内蔵抵抗の有無	説明
MA[19:0]	出力	Low (但し、bit11 のみ High)	無し	
MD[15:0]	入力	High-Z(Pull-down)	Pull Down 抵抗有り	100kΩ
MCS[1:0]#	出力	High	無し	
MCS[3:2]#	入力	High-Z	無し	外部回路依存(GPIOD[5:4])
MBEL#	出力	Low	無し	
MBEH#	出力	Low	無し	
MOE#	出力	High	無し	
MWE#	出力	High	無し	
SDA[14:0]	出力	Low	無し	
SDD[15:0]	入力	High-Z(Pull-down)	Pull Down 抵抗有り	100kΩ
SDCS[1:0]#	出力	High	無し	
SDWE#	出力	High	無し	
SDCLK	出力	SDCLK(32KHz)	無し	
SDLKEN	出力	Low	無し	
SDRAS#	出力	High	無し	
SDCAS#	出力	High	無し	
SDDQM[1:0]#	出力	Low	無し	
CM1DATA[7:0]	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM1VREF	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM1HREF	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM1CLKOUT	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM1CLKIN	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM2DATA[7:0]	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM2VREF	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM2HREF	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM2CLKOUT	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CM2CLKIN	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
CFCE2#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFCE1#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFIORD#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFIOWR#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFWAIT#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFRST	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFIRQ	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFSTSCHG#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFDEN#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
CFDDIR	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
GPIOA[7:0]	入力	High-Z	無し	外部回路依存
GPIOB[7:0]	入力	High-Z	無し	外部回路依存
GPIOC[7:0]	入力	High-Z	無し	外部回路依存
GPIOD[7:0]	入力	High-Z	無し	外部回路依存
GPIOJ[7:0]	入力	High-Z(Pull-down)	Pull Down 抵抗有り	100kΩ
GPIOK[7:0]	入力	High-Z(Pull-down)	Pull Down 抵抗有り	100kΩ
SYSCLKI	−	High-Z	無し	SYSCKSEL に依存
SYSVCP	出力	High-Z	無し	Open のまま使用する
SYSCKSEL	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
TRST#	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
TCK	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
TMS	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
TDI	入力	High-Z(Pull-up)	Pull Up 抵抗有り	50kΩ
TDO	出力	High-Z	無し	
TESTEN0	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
TESTEN1	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
TESTCK	入力	High-Z(Pull-down)	Pull Down 抵抗有り	50kΩ
RESET#	入力	Low	無し	

4. 機能説明

4.1 システムコンフィギュレーション

S2S65A00 ではチップの内部動作を、MD バスから決めるようになっています。
具体的にはリセット中の MD[15:0]バスの値を RESET#がアクティブの期間中及びその立ち上がりに内部でサンプルし、確定させることにより、IC の動作モードが確定します。また、MD バス内部にはプルダウン抵抗（Typ: 100KΩ）が内蔵されているため、通常は何もする必要はありません。システムの都合によっては外部にプルアップ抵抗をつけ、動作モードの変更を行うことができます。
外部にプルアップ抵抗をつけた場合には、外部のプルアップ抵抗と内部のプルダウン抵抗による定常電流が存在することになります。しかし S2S65A00 では、このチップ内部のプルダウン抵抗をソフトウェアにより切り離すことによりこの定常電流を切ることができます。
詳細は“13. システムコントローラ”を参照してください。

表 4.1 システムコンフィギュレーション端子 MODESEL[15:0]

端子名	端子機能	リセット時の値	
		Low	High
MD0	MODESEL0	32KHz Mode（通常動作）	予約（テスト用）*
MD1	MODESEL1	通常動作	予約（テスト用）*
MD2	MODESEL2	通常動作	予約（テスト用）*
MD3	MODESEL3	USER 設定用	USER 設定用
MD4	MODESEL4	USER 設定用	USER 設定用
MD5	MODESEL5	USER 設定用	USER 設定用
MD6	MODESEL6	USER 設定用	USER 設定用
MD7	MODESEL7	USER 設定用	USER 設定用
MD8	MODESEL8	USER 設定用	USER 設定用
MD9	MODESEL9	USER 設定用	USER 設定用
MD10	MODESEL10	USER 設定用	USER 設定用
MD11	MODESEL11	USER 設定用	USER 設定用
MD12	MODESEL12	USER 設定用	USER 設定用
MD13	MODESEL13	USER 設定用	USER 設定用
MD14	MODESEL14	USER 設定用	USER 設定用
MD15	MODESEL15	USER 設定用	USER 設定用

注意 *：予約（テスト用）と指定されている設定は使用しないでください。予約（テスト用）の設定を使用した場合には IC の破壊に至ることがあります。

MD0： クロック入力の設定
Low： 32KHz （PLL 入力：通常動作）
High：予約（テスト用：ユーザは使用できません。）

MD[2:1]： クロック関連設定
Low： 通常動作指定
High：予約（テスト用：ユーザは使用できません）

MD [15:3]： ユーザ用に 12Bits 分を開放しています。
設定された値はシステムコントローラ内の Chip Configuration Register に反映されます。ユーザ側で用途を決めて利用できます。

4.2 メモリマップ

内部では 2 つの AHB Bus が存在しますが、ARM720T が接続された AHB Bus 側を以下 AHB1 Bus と呼びます。

4.2.1 メモリマップ（AHB1）

表 4.2 AHB1 メモリマップ

Start Address	End Address	Size (Mega Byte)	Device	外部 Chip Select	Device Bus size (bit)
0x0000_0000	0x01FF_FFFF	32 MB	外部 ROM	MCS0#	16
0x0200_0000	0x03FF_FFFF	32 MB	Reverved		
0x0400_0000	0x05FF_FFFF	32 MB	外部 SRAM	MCS1#	16
0x0600_0000	0x07FF_FFFF	32 MB	Reverved		
0x0800_0000	0x09FF_FFFF	32 MB	外部 SRAM	MCS2#	16
0x0A00_0000	0x0BFF_FFFF	32 MB	Reverved		
0x0C00_0000	0x0DFF_FFFF	32 MB	外部 SRAM	MCS3#	16
0x0E00_0000	0x0FFF_FFFF	32 MB	Reverved		
0x1000_0000	0x1FFF_FFFF	256 MB	Reserved		
0x2000_0000	0x2FFF_FFFF	256 MB	内蔵 SRAM		32
0x3000_0000	0x3FFF_FFFF	256 MB	Reserved		
0x4000_0000	0x4FFF_FFFF	256 MB	外部 SDRAM	SDCS0#	16/32
0x5000_0000	0x5FFF_FFFF	256 MB	外部 SDRAM	SDCS1#	16/32
0x6000_0000	0x6FFF_FFFF	256 MB	Reserved		
0x7000_0000	0x7FFF_FFFF	256 MB	Reserved		
0x8000_0000	0x8FFF_FFFF	256 MB	Reserved		
0x9000_0000	0x9FFF_FFFF	256 MB	Reserved		
0xA000_0000	0xAFFF_FFFF	256 MB	Reserved		
0xB000_0000	0xBFFF_FFFF	256 MB	Reserved		
0xC000_0000	0xC1FF_FFFF	32 MB	外部 ROM	MCS0#	16
0xC200_0000	0xC3FF_FFFF	32 MB	Reserved		
0xC400_0000	0xC5FF_FFFF	32 MB	外部 SRAM	MCS1#	16
0xC600_0000	0xC7FF_FFFF	32 MB	Reserved		
0xC800_0000	0xC9FF_FFFF	32 MB	外部 SRAM	MCS2#	16
0xCA00_0000	0xCBFF_FFFF	32 MB	Reserved		
0xCC00_0000	0xCDFF_FFFF	32 MB	外部 SRAM	MCS3#	16
0xCE00_0000	0xCFFF_FFFF	32 MB	Reserved		
0xD000_0000	0xDFFF_FFFF	256 MB	Reserved		
0xE000_0000	0xEFFF_FFFF	256 MB	USB		8
0xF000_0000	0xFFFF_FFFF	256 MB	内蔵 I/O エリア		32/16/8

CS0 につながるデバイスがブートデバイスとなります。

0xC000_0000～0xCFFF_FFFF のエリアは 0x0000_0000 に配置された Device の別名空間エリアになります。すなわち、こちらの空間をアクセスしても同じ内容が見えます。

メモリマップの最下位に位置する部分は S2S65A00 の内蔵 I/O デバイスによって使用されます。この内蔵 I/O の配置は、“仕様書：IO マップ”に示します。

4.2.2 メモリマップ (AHB2)

S2S65A00 の 2 つめの AHB-Bus は AHB2 とよばれます。以下にメモリマップを示します。

表 4.3 メモリマップ (AHB2)

Start Address	End Address	Size (Mega Byte)	Device	外部 Chip Select	Device Bus size (bit)
0x0000_0000	0x01FF_FFFF	32 MB	外部 ROM	MCS0#	16
0x0200_0000	0x03FF_FFFF	32 MB	Reverved		
0x0400_0000	0x05FF_FFFF	32 MB	外部 SRAM	MCS1#	16
0x0600_0000	0x07FF_FFFF	32 MB	Reverved		
0x0800_0000	0x09FF_FFFF	32 MB	外部 SRAM	MCS2#	16
0x0A00_0000	0x0BFF_FFFF	32 MB	Reverved		
0x0C00_0000	0x0DFF_FFFF	32 MB	外部 SRAM	MCS3#	16
0x0E00_0000	0x0FFF_FFFF	32 MB	Reverved		
0x1000_0000	0x1FFF_FFFF	256 MB	Reserved		
0x2000_0000	0x2FFF_FFFF	256 MB	内蔵 SRAM		32
0x3000_0000	0x3FFF_FFFF	256 MB	Reserved		
0x4000_0000	0x4FFF_FFFF	256 MB	外部 SDRAM	SDCS0#	16/32
0x5000_0000	0x5FFF_FFFF	256 MB	外部 SDRAM	SDCS1#	16/32
0x6000_0000	0x6FFF_FFFF	256 MB	Reserved		
0x7000_0000	0x7FFF_FFFF	256 MB	Reserved		
0x8000_0000	0x8FFF_FFFF	256 MB	Reserved		
0x9000_0000	0x9FFF_FFFF	256 MB	Reserved		
0xA000_0000	0xAFFF_FFFF	256 MB	Reserved		
0xB000_0000	0xBFFF_FFFF	256 MB	Reserved		
0xC000_0000	0xC1FF_FFFF	32 MB	外部 ROM	MCS0#	16
0xC200_0000	0xC3FF_FFFF	32 MB	Reserved		
0xC400_0000	0xC5FF_FFFF	32 MB	外部 SRAM	MCS1#	16
0xC600_0000	0xC7FF_FFFF	32 MB	Reserved		
0xC800_0000	0xC9FF_FFFF	32 MB	外部 SRAM	MCS2#	16
0xCA00_0000	0xCBFF_FFFF	32 MB	Reserved		
0xCC00_0000	0xCDFF_FFFF	32 MB	外部 SRAM	MCS3#	16
0xCE00_0000	0xCFFF_FFFF	32 MB	Reserved		
0xD000_0000	0xDFFF_FFFF	256 MB	JPEG2 DMA Port		32
0xE000_0000	0xEFFF_FFFF	256 MB	JPEG1 DMA Port		32
0xF000_0000	0xFFFF_FFFF	256 MB	Reserved		

AHB1 および AHB2 のメモリマップの通り、S2S65A00 の（外部）メモリコントローラにつながったメモリ（外部 ROM、外部 SRAM、外部 SDRAM)、および内蔵 SRAM に関しては、共有リソースとしてどちらの AHB-Bus (Master) からもアクセスすることができます。

4.3 I/O マップ

内蔵 I/O 領域は 0xF000_0000～0xFFFF_FFFF の 256 MB 領域の内、以下の部分を実際に使用しています。空欄部分（予約）に関しては Device は存在せず、アクセスしても不定データが読み出されるだけです。

表 4.4 内蔵 I/O マップ

Base Address	Space Size	S2S65A00
0xFFFD_0000	4KB	予約
0xFFFD_1000	4KB	予約
0xFFFD_2000	4KB	予約
0xFFFD_3000	4KB	Timer B
0xFFFD_4000	4KB	UART2
0xFFFD_5000	4KB	UART3
0xFFFD_6000	4KB	予約
0xFFFD_7000	4KB	予約
0xFFFD_8000	4KB	カメラ I/F_2
0xFFFD_9000	4KB	JPEG resize_2
0xFFFD_A000	4KB	JPEG module/FIFO_2
0xFFFD_B000	4KB	JPEG codec_2
0xFFFD_C000	4KB	ADC
0xFFFD_D000	4KB	SD メモリ
0xFFFD_E000	4KB	予約
0xFFFD_F000	4KB	USB_2.0_Device
0xFFFE_0000	4KB	APB ブリッジ
0xFFFE_1000	4KB	予約
0xFFFE_2000	4KB	予約
0xFFFE_3000	4KB	DMAC1
0xFFFE_4000	2KB	CF attribute
0xFFFE_4800	2KB	CF common
0xFFFE_5000	2KB	CF I/O
0xFFFE_5800	1KB	CF ture IDE CS1#
0xFFFE_5C00	1KB	CF ture IDE CS2#
0xFFFE_6000	4KB	CF　制御
0xFFFE_7000	4KB	ARS
0xFFFE_8000	4KB	カメラ I/F_1
0xFFFE_9000	4KB	JPEG resize_1
0xFFFE_A000	4KB	JPEG module/FIFO_1
0xFFFE_B000	4KB	JPEG codec_1
0xFFFE_C000	4KB	JPEG DMAC
0xFFFE_D000	4KB	I2C
0xFFFE_E000	4KB	I2S
0xFFFE_F000	4KB	(割込みコントローラ)
0xFFFF_0000	4KB	予約
0xFFFF_1000	4KB	GPIO/端子機能
0xFFFF_2000	4KB	SPI
0xFFFF_3000	4KB	予約
0xFFFF_4000	4KB	予約
0xFFFF_5000	4KB	UART1
0xFFFF_6000	1KB	予約
0xFFFF_7000	4KB	予約
0xFFFF_8000	4KB	RTC
0xFFFF_9000	4KB	DMAC2
0xFFFF_A000	4KB	メモリコントローラ
0xFFFF_B000	4KB	Timer A
0xFFFF_C000	4KB	WDT
0xFFFF_D000	4KB	システムコントローラ
0xFFFF_E000	4KB	予約
0xFFFF_F000	4KB	割込みコントローラ

4.4 割り込みコントローラ

S2S65A00 では FIQ については 2 つのソース、IRQ については最大 32 個の割り込みソースを取り扱うことができます。

IRQ のマッピングは以下のように内部の割り込み要求が割り込みコントローラに接続されています。詳細は“15. 割り込みコントローラ”を参照してください。

種類	レベル	S2S65A00
FIQ	FIQ0	ウオッチドックタイマ
	FIQ1	GPIOB0
IRQ	IRQ0	ウオッチドックタイマ
	IRQ1	割込みコントローラ
	IRQ2	ARM720T COMMRx
	IRQ3	ARM721T COMMTx
	IRQ4	Timer A Ch.0
	IRQ5	Timer A Ch.1
	IRQ6	Timer A Ch.2
	IRQ7	予約
	IRQ8	JPEG 制御_1
	IRQ9	DMA 1 C
	IRQ10	JPEG　DMAC_1
	IRQ11	カメラ　I/F_1
	IRQ12	ARS
	IRQ13	DMA2C
	IRQ14	GPIOA or GPIOB
	IRQ15	SPI
	IRQ16	I2C
	IRQ17	UART_1
	IRQ18	RTC
	IRQ19	CF カード I/F
	IRQ20	INT_GPIOB (GPIOB)
	IRQ21	Timer B
	IRQ22	予約
	IRQ23	JPEG 制御_2
	IRQ24	JPEG　DMAC_2
	IRQ25	カメラ　I/F_2
	IRQ26	UART2
	IRQ27	UART3
	IRQ28	SD メモリ
	IRQ29	USB_DEV
	IRQ30	ADC
	IRQ31	I2S

4.5 S2S65A00 の内蔵機能

S2S65A00 は、ドライブカメラコントローラ機能を実現するために多くの機能ブロックを内蔵しています。
以下に一覧を示します。

章番号	機能ブロック名	機能名略号
5.	CPU	CPU
6.	DMA コントローラ 1	DMAC1
7.	カメラインタフェース[1:0]	CAM[1:0]
8.	JPEG コントローラ[1:0]	JPG[1:0]
9.	JPEG_DMAC	JDMA
10.	DMA コントローラ 2	DMAC2
11.	USB2.0 デバイス	USB2d
12.	APB ブリッジ	APB
13.	システムコントローラ	SYS
14.	メモリコントローラ	MEMC
15.	割り込みコントローラ	INT
16.	UART[2:0]	UART[2:0]
17.	I2C Single Master Core Module	I2C
18.	I2S インタフェース	I2S
19.	シリアル周辺機器インタフェース	SPI
20.	コンパクトフラッシュカードインタフェース	CF
21.	SD メモリコントロールインタフェース	SD
22.	タイマ A	TIMA
23.	タイマ B	TIMB
24.	リアルタイムクロック	RTC
25.	ウォッチドッグタイマ	WDT
26.	GPIO	GPIO
27.	AD コンバータ	ADC
28.	エリアセンサ	ARS

5. CPU

5.1 概要

S2S65A00 の CPU モジュールとしては、ARM720T コアが使用されています。ARM720T は ARM7TDMI をコアとして、ユニファイド 8k バイトキャッシュ、メモリ管理ユニット（MMU）、拡張書き込みバッファを搭載しています。

ARM720T コアについての詳細は“*ARM720T Revision 4 (AMBA AHB Bus Interface Version) コア CPU マニュアル*”を参照してください。

5.2 ARM720T ブロック図

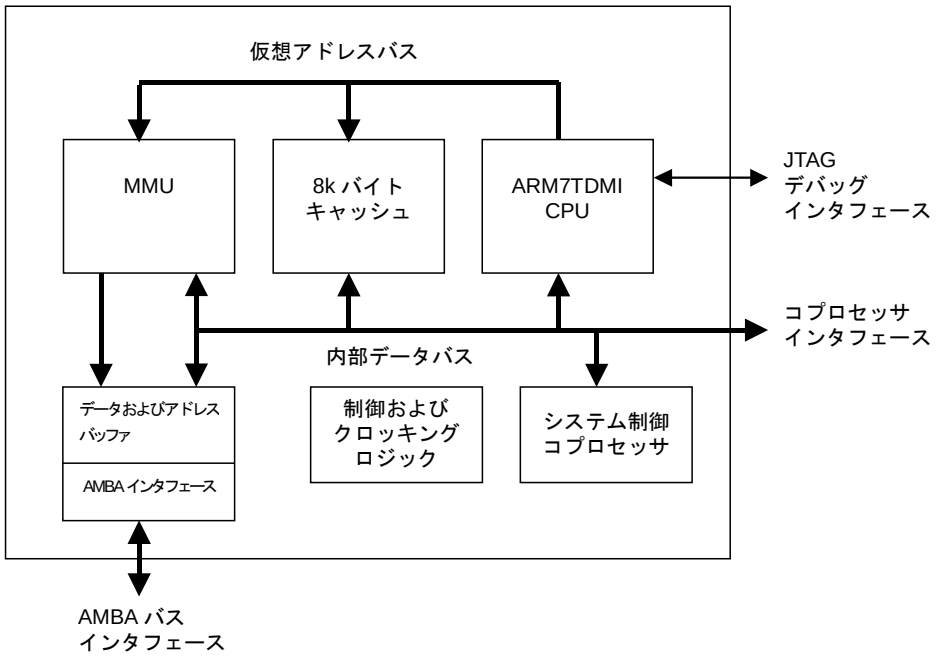


図 5.1　ARM720T ブロック図

6. DMA コントローラ 1 (DMAC1)

6.1 概要

DMAC1 は、AHB1 上のバスマスタとして置かれ、APB デバイスとメモリ間（内蔵および外部メモリ）、またはメモリ間（内蔵および外部メモリ）で、CPU を介することなく転送を行う DMA コントローラです。

本 DMA コントローラは 2 つのアドレス・フェーズを使うデュアルアドレス転送方式を採用しています。この方式は DMA 要求がある度ごとに、転送元アドレスからデータを DMAC1 内のテンポラリーレジスタに読み込み、次に転送先アドレスへテンポラリーレジスタの内容を書き込む方式で、転送回数が“0”になるまで、この動作を繰り返します。転送サイズは 8、16、32 ビットを指定可能です。

6.2 ブロック図

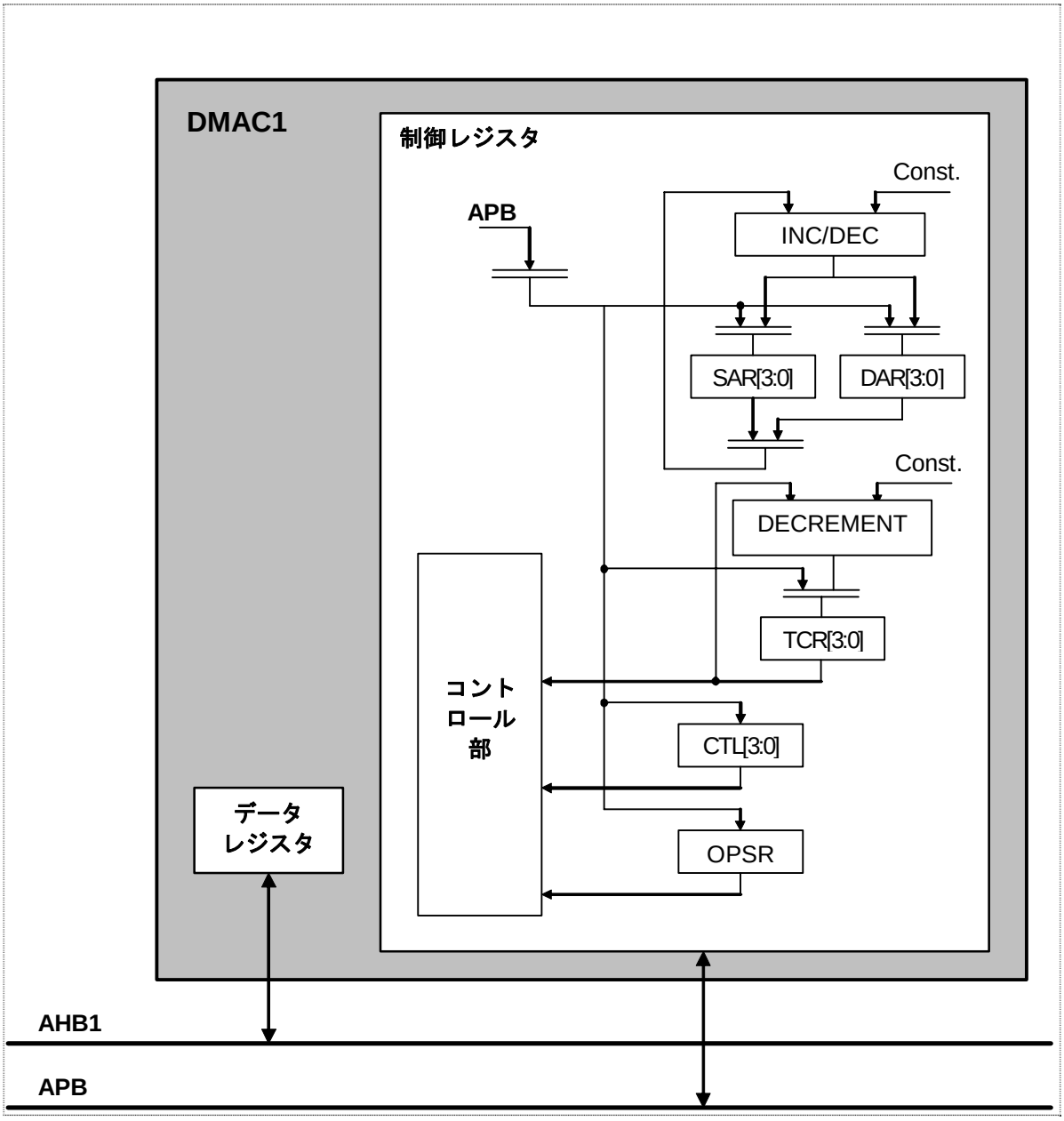


図 6.1　DMA コントローラ 1（DMAC1）ブロック図

6.3 外部端子

DMA コントローラ 1 に関連する外部端子はありません。

6.4 レジスタ

6.4.1 レジスター覧

DMAC1 関連レジスタのベースアドレスは 0xFFFE_3000 です。
この章以降のレジスタに関しては、以下のような省略を用いることがあります。

R/W： リードおよびライト
RO： リードオンリ
WO： ライトオンリ
RSV： 予約ビット／レジスタ（特に指定のない場合は“0”を書き込んでください。）
n/a： not available（特に指定のない場合は“0”を書き込んでください。）

表 6.1 レジスター覧（**ベースアドレス：0xFFFE_3000**）

アドレス オフセット	レジスタ名称	レジスタ 略称	初期値	R/W	データ アクセス サイズ
0x00	DMA チャネル 0 ソースアドレスレジスタ	SAR0	0xXXXX_XXXX	R/W	32
0x04	DMA チャネル 0 デスティネーションアドレスレジスタ	DAR0	0xXXXX_XXXX	R/W	32
0x08	DMA チャネル 0 転送カウントレジスタ	TCR0	0x00XX_XXXX	R/W	32
0x0C	DMA チャネル 0 コントロールレジスタ	CTL0	0x00XX_XXXX	R/W	32
0x10	DMA チャネル 1 ソースアドレスレジスタ	SAR1	0xXXXX_XXXX	R/W	32
0x14	DMA チャネル 1 デスティネーションアドレスレジスタ	DAR1	0xXXXX_XXXX	R/W	32
0x18	DMA チャネル 1 転送カウントレジスタ	TCR1	0x00XX_XXXX	R/W	32
0x1C	DMA チャネル 1 コントロールレジスタ	CTL1	0x00XX_XXXX	R/W	32
0x20	DMA チャネル 2 ソースアドレスレジスタ	SAR2	0xXXXX_XXXX	R/W	32
0x24	DMA チャネル 2 デスティネーションアドレスレジスタ	DAR2	0xXXXX_XXXX	R/W	32
0x28	DMA チャネル 2 転送カウントレジスタ	TCR2	0x00XX_XXXX	R/W	32
0x2C	DMA チャネル 2 コントロールレジスタ	CTL2	0x00XX_XXXX	R/W	32
0x30	DMA チャネル 3 ソースアドレスレジスタ	SAR3	0xXXXX_XXXX	R/W	32
0x34	DMA チャネル 3 デスティネーションアドレスレジスタ	DAR3	0xXXXX_XXXX	R/W	32
0x38	DMA チャネル 3 転送カウントレジスタ	TCR3	0x00XX_XXXX	R/W	32
0x3C	DMA チャネル 3 コントロールレジスタ	CTL3	0x00XX_XXXX	R/W	32
0x60	DMA チャネル オペレーティング選択レジスタ	OPSR	0x0000_0000	R/W	32

6.4.2 レジスタ詳細

特に指定のない場合、予約ビットには“0”をセットしてください。予約ビットに書き込みを行うと予想できない結果になることがあります。“n/a”と記載してあるビットはハードウェアに影響をあたえません。

あるレジスタは特定の条件のときのみアクセスできるようになっています。アクセス不可のレジスタへのリード／ライトは無効です。

DMA チャネル[3:0] ソースアドレスレジスタ (SAR[3:0])

DMAC1[0x00] [0x10] [0x20] [0x30]　　初期値 = 0xXXXX_XXXX　　Read/Write

DMA チャネル[3:0] ソースアドレス [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル[3:0] ソースアドレス [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**DMA チャネル[3:0] ソースアドレス [31:0]**

チャネル[3:0]で DMA 転送を行う場合の転送元アドレスをソフトウェアにより設定します。転送元アドレスは転送データサイズに合った境界アドレスを設定する必要があります。例えば 32 ビット転送の場合、このレジスタのビット[1:0]は 00b でなければなりません。DMA 転送開始後は、1 回の転送終了ごとに転送データサイズ（TS：チャネル[3:0]コントロールレジスタ ビット[4:3]）とソースアドレスモード（SAM：チャネル[3:0]コントロールレジスタ ビット[13:12]）に従った次の転送元アドレスに自動的に更新されます。

DMA チャネル[3:0] デスティネーションアドレスレジスタ (DAR[3:0])

DMAC1[0x04] [0x14] [0x24] [0x34]　　初期値 = 0xXXXX_XXXX　　Read/Write

DMA チャネル[3:0] デスティネーションアドレス [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル[3:0] デスティネーションアドレス [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**DMA チャネル[3:0] デスティネーションアドレス [31:0]**

チャネル[3:0]で DMA 転送を行う場合の転送先アドレスをソフトウェアにより設定します。転送先アドレスは転送データサイズに合った境界アドレスを設定する必要があります。例えば 32 ビット転送の場合、このレジスタのビット[1:0]は 00b でなければなりません。DMA 転送開始後は、1 回の転送終了ごとに転送データサイズ（TS：チャネル[3:0]コントロールレジスタ ビット[4:3]）とデスティネーションアドレスモード（DAM：チャネル[3:0]コントロールレジスタ ビット[15:14]）に従った次の転送先アドレスに自動的に更新されます。

DMA チャネル[3:0] 転送カウントレジスタ (TCR[3:0])

DMAC1[0x08] [0x18] [0x28] [0x38]　　初期値 = 0x00XX_XXXX　　Read/Write

n/a								DMA チャネル[3:0] 転送カウント [23:16]							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル[3:0] 転送カウント [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:0]：　**DMA チャネル[3:0] 転送カウント [23:0]**

これらのビットには DMA 転送時の転送回数をソフトウェアにより設定します。DMA 転送を開始すると、1 回の転送終了ごとにデクリメントされます。ここに“0”を設定した場合、転送回数は 2^{24}=16777216 回になります。DMA 割り込みはこのカウンタがカウントダウンによって“0”になった時点で発生します。

このレジスタをリードすると bits[31:24]には“0”が付加されます。

DMA チャネル[1:0] コントロールレジスタ (CTL[1:0])

DMAC1[0x0C] [0x1C]　　初期値 = 0x00XX_XXXX　　Read/Write

n/a								RSV				IDLE	RSV	AM	RSV
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DAM		SAM		RS				RSV	RIM	TM	TS		IE	TE	DE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:20] :　**RSV　予約（0）**

Bit 19 :　**IDLE　アイドル遅延イネーブル**

DMA ターゲット・デバイスによっては、イネーブルの設定が必要です。
デバイスへのライト転送（メモリから IO デバイスへの転送）時には“1”に設定することを推奨します。

0： 通常動作
1： デバイスからの次のリクエストの受け付けタイミングを遅らせます。

Bit 18 :　**RSV　予約**

Bit 17 :　**AM　アクノリッジモード**

DACK 信号の出力タイミングを選択します。

0： DMA リードサイクルの間アクティブ
1： DMA ライトサイクルの間アクティブ

Bit 16 :　**RSV　予約（0）**

Bits [15:14] :　**DAM　デスティネーションアドレスモード [1:0]**

1 回転送終了後のデスティネーションアドレスレジスタの更新モードを選択します。

00：転送先アドレス固定（更新しない）
01：転送先アドレスを転送データサイズに合わせてインクリメント
　（8 ビットは+1、16 ビットは+2、32 ビットは+4）
10：転送先アドレスを転送データサイズに合わせてデクリメント
　（8 ビットは-1、16 ビットは-2、32 ビットは-4）
11：予約

Bits [13:12] :　**SAM　ソースアドレスモード [1:0]**

1 回転送終了後のソースアドレスレジスタの更新モードを選択します。

00：転送元アドレス固定（更新しない）
01：転送元アドレスを転送データサイズに合わせてインクリメント
　（8 ビットは+1、16 ビットは+2、32 ビットは+4）
10：転送元アドレスを転送データサイズに合わせてデクリメント
　（8 ビットは-1、16 ビットは-2、32 ビットは-4）
11：予約

Bits [11:8] :　**RS　リソース選択 [3:0]**

DMA 転送を開始させる要因を選択します。

0000： USB_Device 入出力(USB-REQ0)
0001： USB_Device 入出力(USB-REQ1)
0010： Reserved
0011： Reserved
0100： UART#1 出力(TX)
0101： UART#1 入力(RX)
0110： UART#2 出力(TX)
0111： UART#2 入力(RX)
1000： UART#3 出力(TX)
1001： UART#3 入力(RX)
1010： SPI 入出力(SPI-TX/RX)
1011： Reserved

1100： SD_MEM 入出力(SDM-REQ)
1101： Reserved
1110： Reserved
1111： SW-Request　ソフトウェアリクエスト
Bits[11:8]を“1111”にセットすることでソフトウェアの DMA 転送を選択します。

Bit 7： **RSV　予約（0）**

Bit 6： **RIM　リクエスト入力モード**
DMA 要求信号の入力モードを選択します。
0： アクティブ LOW（レベルトリガ）
1： 立ち下がりエッジ（エッジトリガ）

Bit 5： **TM　転送モード**
DMA 転送の転送モードを選択します。
0： シングル転送　（1 回の DMA 要求にたいして 1 回の転送）
1： デマンド転送　（DMA 要求がある間は転送を継続）

Bits [4:3]： **TS　転送サイズ [1:0]**
1 回で転送するデータサイズを選択
00：8 ビット
01：16 ビット
10：32 ビット
11：予約

Bit 2： **IE　割り込みイネーブル**
DMA チャネル[1:0]の転送終了割り込みを許可／禁止します。
0： 割り込み禁止
1： 割り込み許可

Bit 1： **TE　DMA 転送終了**
0(リード時)：転送中または待機中
1(リード時)：DMA 転送終了
0(ライト時)：本ビットをクリア
1(ライト時)：無効
このビットは DMA 転送の結果、DMA チャネル 0 転送カウントレジスタの値が “0” になるとセットされます。一度セットされるとこのビットに “0” をライトして本ビットをクリアするまで “1” を保持します。本ビットがクリアされるまでは、このチャネルの DMA 転送が禁止されます。このビットは割り込みソースフラグとしても機能します。

Bit 0： **DE　DMA イネーブル**
このビットによりチャネル[1:0]の DMA 転送を許可します。
0： DMA 転送禁止
1： DMA 転送許可

DMA チャネル[3:2] コントロールレジスタ (CTL[3:2])

DMAC1[0x2C] [0x3C]　　初期値 = 0x00XX_XXXX　　Read/Write

n/a								RSV				IDLE	RSV	AM	RSV
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DAM		SAM		RS				RSV	RIM	TM	TS		IE	TE	DE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:20] :　**RSV　予約（0）**

Bit 19 :　**IDLE　アイドル遅延イネーブル**

DMA ターゲット・デバイスによっては、イネーブルの設定が必要です。
デバイスへのライト転送（メモリから IO デバイスへの転送）時には“1”に設定することを推奨します。

0： 通常動作
1： デバイスからの次のリクエストの受け付けタイミングを遅らせます。

Bit 18 :　**RSV　予約（0）**

Bit 17 :　**AM　アクノリッジモード**

DACK 信号の出力タイミングを選択します。

0： DMA リードサイクルの間アクティブ
1： DMA ライトサイクルの間アクティブ

Bit 16 :　**RSV　予約（0）**

Bits [15:14] :　**DAM　デスティネーションアドレスモード [1:0]**

1 回転送終了後のデスティネーションアドレスレジスタの更新モードを選択します。

00：転送先アドレス固定（更新しない）
01：転送先アドレスを転送データサイズに合わせてインクリメント
　（8 ビットは+1、16 ビットは+2、32 ビットは+4）
10：転送先アドレスを転送データサイズに合わせてデクリメント
　（8 ビットは-1、16 ビットは-2、32 ビットは-4）
11：予約

Bits [13:12] :　**SAM　ソースアドレスモード [1:0]**

1 回転送終了後のソースアドレスレジスタの更新モードを選択します。

00：転送元アドレス固定（更新しない）
01：転送元アドレスを転送データサイズに合わせてインクリメント
　（8 ビットは+1、16 ビットは+2、32 ビットは+4）
10：転送元アドレスを転送データサイズに合わせてデクリメント
　（8 ビットは-1、16 ビットは-2、32 ビットは-4）
11：予約

Bits [11:8] :　**RS　リソース選択 [3:0]**

DMA 転送を開始させる要因を選択します。

0000： USB_Device#0 入出力(USB-REQ0)
0001： USB_Device#1 入出力(USB-REQ1)
0010： Reserved
0011： Reserved
0100： I2C 出力(I2C-TX)
0101： I2C 入力(I2C-RX)
0110： I2S 入出力(I2S-CH0)
0111： I2S 入出力(I2S-CH1)
1000： Reserved
1001： Reserved
1010： SPI 入出力(SPI-TX/RX)
1011： Reserved

1100： SD_MEM 入出力(SDM-REQ)
1101： Reserved
1110： Reserved
1111： SW-Request　ソフトウェアリクエスト
Bits[11:8]を“1111”にセットすることでソフトウェアの DMA 転送を選択します。

Bit 7： **RSV　予約（0）**

Bit 6： **RIM　リクエスト入力モード**
DMA 要求信号の入力モードを選択します。
0： アクティブ LOW（レベルトリガ）
1： 立ち下がりエッジ（エッジトリガ）

Bit 5： **TM　転送モード**
DMA 転送の転送モードを選択します。
0： シングル転送　（1 回の DMA 要求にたいして 1 回の転送）
1： デマンド転送　（DMA 要求がある間は転送を継続）

Bits [4:3]： **TS　転送サイズ [1:0]**
1 回で転送するデータサイズを選択
00：8 ビット
01：16 ビット
10：32 ビット
11：予約

Bit 2： **IE　割り込みイネーブル**
DMA チャネル[3:2]の転送終了割り込みを許可／禁止します。
0： 割り込み禁止
1： 割り込み許可

Bit 1： **TE　DMA 転送終了**
0(リード時)：転送中または待機中
1(リード時)：DMA 転送終了
0(ライト時)：本ビットをクリア
1(ライト時)：無効
このビットは DMA 転送の結果、DMA チャネル 3 転送カウントレジスタの値が “0” になるとセットされます。一度セットされるとこのビットに “0” をライトして本ビットをクリアするまで “1” を保持します。本ビットがクリアされるまでは、このチャネルの DMA 転送が禁止されます。このビットは割り込みソースフラグとしても機能します。

Bit 0： **DE　DMA イネーブル**
このビットによりチャネル[3:2]の DMA 転送を許可します。
0： DMA 転送禁止
1： DMA 転送許可

DMA チャネル オペレーティング選択レジスタ (OPSR)

DMAC1[0x60]　　初期値 = 0x0000_0000　　Read/Write

na															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a						DPM		n/a							DGE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit [9:8]：　**DPM　プライオリティモード**

同時に複数のチャネルに対して転送要求があった場合に優先順位を決めます。

00：CH0>CH1>CH2>CH3
01：CH1>CH0>CH3>CH2
10：CH2>CH3>CH0>CH1
11：簡易ラウンドロビンモード

Bit 0：　**DGE　DMA グローバルイネーブル**

DMA の全チャネルをすべてイネーブル／ディセーブルにします。

0： ディセーブル
1： イネーブル

7. カメラインタフェース[2:1] (CAM[2:1])

7.1 概要

カメラインタフェースは以下のような特長を持っています。

- 8 ビットパラレルインタフェース：2 ポート
- 2 台のカメラモジュール接続可能
- 最大 VGA サイズ（640×480）までのカメラに対応可能（AC 特性に依存）
- 8 ビットデータバスインタフェース（YUV4:2:2 形式）
- ITU-R BT.656 形式に対応
- キャプチャフレームの設定可能
- カメラデータ入力用ピクセルクロック周波数は CPU クロックの 2/3 未満

7.2 ブロック図

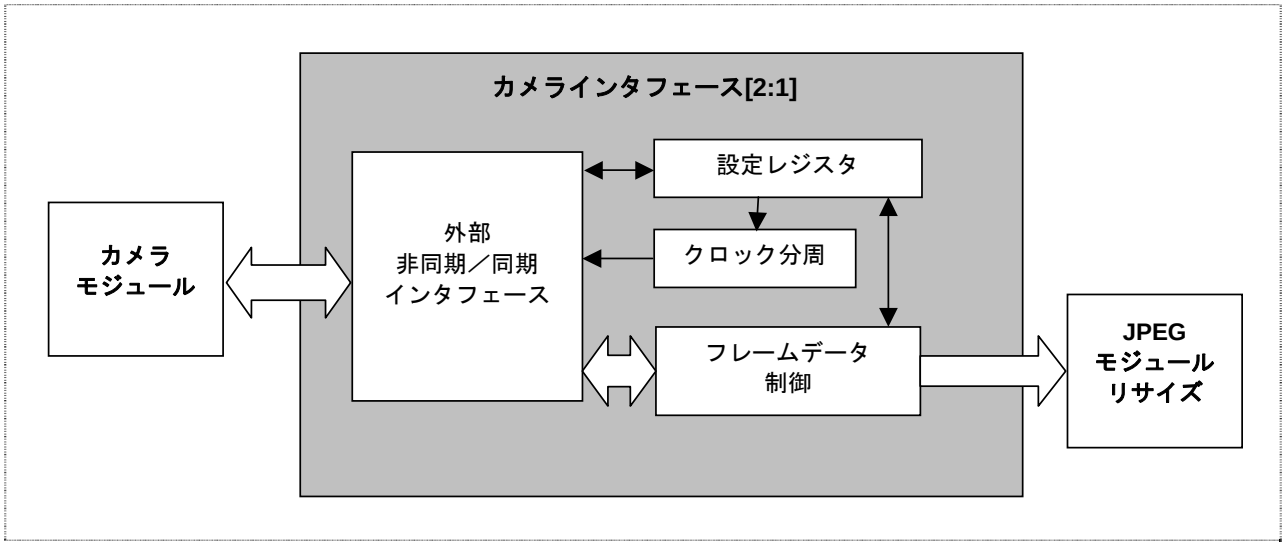


図 7.1　カメラインタフェース[2:1] ブロック図

7.3 外部端子

カメラインタフェース[2:1] 関連の外部端子は以下のようになっています。

表 7.1　外部端子

端子名	入出力	端子機能	マルチプレクス端子*／備考
CM1DATA [7:0]	I	カメラ 1 データ(YUV)入力	GPIOE [7:0]
CM1VREF	I	カメラ 1 データ入力時の垂直同期入力	GPIOF0
CM1HREF	I	カメラ 1 データ入力時の水平同期入力	GPIOF1
CM1CLKOUT	O	カメラ 1 用基本クロック出力	GPIOF2
CM1CLKIN	I	カメラ 1 データ入力用ピクセルクロック	GPIOF3
CM2DATA [7:0]	I	カメラ 2 データ(YUV)入力	GPIOG [7:0]
CM2VREF	I	カメラ 2 データ入力時の垂直同期入力	GPIOF4
CM2HREF	I	カメラ 2 データ入力時の水平同期入力	GPIOF5
CM2CLKOUT	O	カメラ 2 用基本クロック出力	GPIOF6
CM2CLKIN	I	カメラ 2 データ入力用ピクセルクロック	GPIOF7

注意(*): カメラインタフェース(CAM)用の外部端子は GPIO 端子とマルチプレクスしています。CAM 用外部端子として使用する場合は、GPIO 端子機能レジスタにより“GPIO 以外の機能 1” に設定してください。

7.4 レジスタ

7.4.1 レジスター覧

以下にカメラインタフェース[2:1]のレジスター覧を示します。これらのレジスタのベースアドレスは次のようになっています。
カメラインタフェース 1：0xFFFE_8000
カメラインタフェース 2：0xFFFD_8000

表 7.2　レジスター覧

アドレス オフセット	レジスタ名称	初期値	R/W	データ アクセス サイズ
カメラインタフェース　：Base Address　0xFFFE_8000(CAM1) 0xFFFD_8000(CAM2)				
0x00	カメラ 1/2 クロック周期設定レジスタ	0x0000	R/W	16bit
0x04	カメラ 1/2 信号設定レジスタ	0x0000	R/W	16bit
0x08 – 0x1C	予約	-	-	-
0x20	カメラ 1/2 モード設定レジスタ	0x0000	R/W	16bit
0x24	カメラ 1/2 フレーム制御レジスタ	0x0000	R/W	16bit
0x28	カメラ 1/2 コントロールレジスタ	0x0000	WO	16bit
0x2C	カメラ 1/2 ステータスレジスタ	0x0004	RO	16bit
0x30 – 0x5C	予約	-	-	-

7.4.2 レジスタ詳細説明

カメラ[2:1]クロック周期設定レジスタ

CAM2 [0x00], CAM1 [0x00]　　初期値 = 0x0000　　Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			Clock Frequency Select bits [4:0]				
7	6	5	4	3	2	1	0

Bits [4:0] :　**Clock Frequency Select bits [4:0]**

出力クロック（CMCLKOUT）の周期を設定します。

表 7.3　カメラクロック周期設定

Bits [4:0]	Clock Frequency
00000	内部クロック 1/1
00001	内部クロック 1/2（推奨値）
00010	内部クロック 1/3
00011	内部クロック 1/4
00100	内部クロック 1/5
00101	内部クロック 1/6
00110	内部クロック 1/7
00111	内部クロック 1/8
01000	内部クロック 1/9
01001	内部クロック 1/10
01010	内部クロック 1/11
01011	内部クロック 1/12
01100	内部クロック 1/13
01101	内部クロック 1/14
01110	内部クロック 1/15
01111	内部クロック 1/16
10000	内部クロック 1/17
10001	内部クロック 1/18
10010	内部クロック 1/19
10011	内部クロック 1/20
10100	内部クロック 1/21
10101	内部クロック 1/22
10110	内部クロック 1/23
10111	内部クロック 1/24
11000	内部クロック 1/25
11001	内部クロック 1/26
11010	内部クロック 1/27
11011	内部クロック 1/28
11100	内部クロック 1/29
11101	内部クロック 1/30
11110	内部クロック 1/31
11111	内部クロック 1/32

カメラ[2:1]信号設定レジスタ CAM2 [0x04], CAM1 [0x04]　　初期値 = 0x0000							Read/Write
n/a							
15	14	13	12	11	10	9	8
n/a	Reserved (0)	Clock Mode Select	YUV Data Format Select bits [1:0]		HREF Active Select	VREF Active Select	Valid Input Clock Edge
7	6	5	4	3	2	1	0

Bit 6：　**予約ビット（0）**
書き込む場合は“0”を書き込んでください。

Bit 5：　**Clock Mode Select**
内部動作で使用するクロックを選択します。内部分周クロック使用時は 1/2 分周以上に設定する必要があります。また外部クロック入力と異なり、内部分周クロックを使用する場合は、ボード上遅延およびカメラ（イメージセンサ）内部遅延によってカメラ（イメージセンサ）の出力データと内部分周クロックとの同期が取りにくくなります。安定動作させるためにはそれらの遅延が無視できる程度まで分周比を大きく設定する必要があります。
0： 外部クロック入力（CMCLKIN 入力を使用）
1： 内部分周クロック（カメラクロック周期設定レジスタで設定したクロック（＝CMCLKOUT と同じクロック）を内部で使用。）

Bits [4:3]：　**YUV Data Format Select bits[1:0]**
バイト単位で入力する YUV データの並び順を選択します。

表 7.4　YUV Format Select

Bits [4:3]	YUV Format (8bit)
00	(1st) CbYCrY (last)
01	(1st) CrYCbY (last)
10	(1st) YCbYCr (last)
11	(1st) YCrYCb (last)

Bit 2：　**HREF Active Select**
HREF のデータアクティブレベルを設定します。
0： HREF のデータ有効期間は High レベルで認識します。
1： HREF のデータ有効期間は Low レベルで認識します。

Bit 1：　**VREF Active Select**
VREF のデータアクティブレベルを設定します。
0： VREF のデータ有効期間時は High レベルで認識します。
1： VREF のデータ有効期間時は Low レベルで認識します。

Bit 0：　**Valid Input Clock Edge**
入力クロックの有効エッジを選択します。
Bit 5 で選択した外部／内部のどちらのクロックに対しても有効です。
0： クロックが Low から High に変化したときデータを取り込みます。
1： クロックが High から Low に変化したときデータを取り込みます。

カメラ[2:1]モード設定レジスタ

CAM2 [0x20], CAM1 [0x20]　　初期値 = 0x0000　　Read/Write

15	14	13	12	11	10	9	8
RSV (0)			RSV Camera Active Pull-down Disable	n/a	Fast Sampling Mode	RSV (0)	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
ITU-R BT656 Enable	RSV (0)			Clock Output Disable	RSV (0)		Camera Module Enable

Bit [15:13]：　**RSV　予約（0）**

Bit 12：　**RSV　予約（Camera Active Pull-down Disable）**

カメラ端子のアクティブプルダウンを無効にします。
0：アクティブプルダウン有効
1：アクティブプルダウン無効

注意：S2S65A00 ではこのビット機能は無効です。システムコントローラ内の GPIOE/GPIOF/GPIOG Resistor Control Register によって同様の制御が可能になっています。

Bit 10：　**Fast Sampling Mode**

カメラからの入力データを通常の倍速でサンプリングする動作を設定します。
0： 通常サンプリング
1： 高速サンプリング

Bit [9:8]：　**RSV　予約（0）**

Bit 7：　**ITU-R BT656 Enable**

ITU-R BT656 対応カメラ入力の選択を行います。
このモードは YUV422-8bit インタフェースモード時のみ有効です。
0： 通常カメラモード
1： ITU-R BT656 対応カメラモード

Bits [6:4]：　**RSV　予約（0）**

Bit 3：　**Clock Output Disable**

クロック出力（CMCLKOUT）の動作を設定します。
0： クロック出力（CMCLKOUT 出力を使用。周期はカメラ周期設定レジスタで設定。）
1： クロック出力禁止（CMCLKOUT を出力せず、Low レベル固定になります。）

Bits [2:1]：　**RSV　予約（0）**

Bit 0：　**Camera Module Enable**

カメラモジュールの動作を設定します。
このビットの設定が有効のとき、外部クロック出力等が有効になります。
このビットが“0”のとき、カメラモジュールへのクロック供給が停止するため、低消費電力対策に有効です。このビットが“0”であってもカメラインタフェースのレジスタ群へのリード/ライトは可能です。
0： カメラモジュール無効
1： カメラモジュール有効

カメラ[2:1]フレーム制御レジスタ

CAM2 [0x24], CAM1 [0x24]　　初期値 = 0x0000　　Read/Write

n/a							Jpeg Raw Data Capture Mode
15	14	13	12	11	10	9	8
Frame Capture Interrupt Control	Single Frame Capture Enable	Shutter Sync. Disable	Frame Sample Control bits [2:0]			Frame Capture Interrupt Polarity	Frame Capture Interrupt Enable
7	6	5	4	3	2	1	0

Bit 8：　**Jpeg Raw Data Capture Mode**

JPEG データ取り込みモードを設定します。

0： YUV データ取り込み設定

1： JPEG データ取り込み設定

Bit 7：　**Frame Capture Interrupt Control**

フレーム割り込みの制御を行います。

フレーム取り込み終了割り込み設定時は bit5 の設定に影響されません。

フレーム取り込み終了割り込み設定時は bit0 の設定には影響されず割り込みが有効になります。

表 7.5　フレーム割り込み制御

Image Capture Interrupt Polarity Bit	Interrupt
0	有効フレーム取り込み時割り込み発生
1	フレーム取り込み終了時割り込み発生

Bit 6：　**Single Frame Capture Enable**

フレーム取り込みのモードを設定します。

0： 繰り返し取り込みを行います。

1： 画像フレーム取り込み動作設定（CAM[0x28] bit 2 = 1）後の 1 フレーム取り込み後停止します。

注意：このビットは Camera Module Enable（CAM[0x20] bit 0）が“1”のときは変更禁止です。

Bit 5：　**Shutter Synchronization Disable**

フレーム割り込みフラグのシャッター同期を禁止します。

0： シャッターを押した後有効フレーム毎にフレーム割り込みステータスを出力します。

1： 有効フレーム毎に常時フレーム割り込みステータスを出力します。

Bits [4:2]：　**Frame Sample Control Bits [2:0]**

カメラ入力のフレーム間引きの制御を行います。

表 7.6　フレーム間引き制御

Frame Sample Control Bits[2:0]	Mode
000	間引きなし
001	1/2 に間引き
010	1/3 に間引き
011	1/4 に間引き
100	1/5 に間引き
101	1/6 に間引き
110	Reserved
111	Reserved（全フレーム間引き）

Bit 1 : **Image Capture Interrupt Polarity**
画像取り込み時の割り込み発生の制御を行います。

表 7.7　フレーム割り込み制御

Image Capture Interrupt Polarity Bit	Interrupt
0	VREF データアクティブレベルから VREF データインアクティブレベルへの変化時
1	VREF データインアクティブレベルから VREF データアクティブレベルへの変化時

Bit 0 : **Image Capture Interrupt Enable**
画像取り込み時の割り込みを有効にします。
0： 無効
1： 有効

カメラ[2:1]コントロールレジスタ

CAM2 [0x28], CAM1 [0x28]　　初期値 = 0x0000　　Write Only

15	14	13	12	11	10	9	8
n/a						ITU-R BT656 Error Flag 1 Clear	ITU-R BT656 Error Flag 0 Clear
7	6	5	4	3	2	1	0
n/a				Frame Capture Stop	Frame Capture Start	Frame Interrupt Status Clear	Camera Module Soft Reset

Bit 9：　**ITU-R BT656 Error Flag 1 Clear**

このビットに“1”を書き込むことで状態をクリアできます。

0： 何も動作しません。

1： エラーフラグ 1 の値をクリアします。

Bit 8：　**ITU-R BT656 Error Flag 0 Clear**

このビットに“1”を書き込むことで状態をクリアできます。

0： 何も動作しません。

1： エラーフラグ 0 の値をクリアします。

Bit 3：　**Frame Capture Stop**

画像フレーム取り込みの停止設定をします。

なお、リセット後のデフォルトは取り込み動作状態ですが、bit 0 カメラモジュールソフトリセットと同時に本ビットを設定することで、カメラモジュールを取り込み停止状態から開始することができます.

0： 何も動作しません

1： 取り込み動作の停止設定をします

Bit 2：　**Frame Capture Start**

画像フレーム取り込みの動作設定をします。

シングルフレーム取り込み設定時は、1 フレーム取り込み後停止します。

0： 何も動作しません。

1： 取り込み動作の動作設定をします。

Bit 1：　**Frame Capture Interrupt Status Clear**

このビットに 1 を書き込むことでカメラからのフレーム割り込みステータスをクリアします。

0： フレーム割り込みステータスをクリアしません。

1： フレーム割り込みステータスをクリアします。

Bit 0：　**Camera Module Soft Reset**

カメラ回路部を初期化します。

0： カメラ回路部を初期化しません。

1： カメラ回路部を初期化します。

注意： このビットが“1”のとき、レジスタは初期化されませんが、CAM [0x20] bit0 (Camera Module Enable) だけは初期化され“0”になります。

カメラ[2:1]ステータスレジスタ
CAM2 [0x2C], CAM1 [0x2C]　　初期値 = 0x0004　　Read Only

n/a						ITU-R BT656 Error Flag 1	ITU-R BT656 Error Flag 0
15	14	13	12	11	10	9	8
n/a	Camera VSYNC	RSV (1)	Effective Frame Status	Frame Capture Busy Status	Frame Capture Start/Stop Flag	Frame Interrupt Status	n/a
7	6	5	4	3	2	1	0

Bit 9 : **ITU-R BT656 Error Flag 1**
ITU-R BT656 モード時、リファレンスコマンドの状態を確認できます。
0： 正常動作
1： リファレンスデコード時 2 bit のエラーを検出

Bit 8 : **ITU-R BT656 Error Flag 0**
ITU-R BT656 モード時、リファレンスコマンドの状態を確認できます。
0： 正常動作
1： リファレンスデコード時 1 bit のエラー訂正を検出

Bit 6 : **Camera VSYNC**
カメラモジュールからの VSYNC の値を確認できます。
ITU-R BT656 モード時はリファレンスコードデコード後の値を確認できます。
また、VSYNC の値はポラリティ設定によらず一定です。
0： 垂直ブランク期間中
1： 有効データ期間中

Bit 5 : **RSV　予約（1）**

Bit 4 : **Effective Frame Status**
このビットを読み込むことでフレーム間引き時の有効/無効フレームを確認できます。
0： フレームは無効（間引き）フレーム
1： フレームは有効フレーム

Bit 3 : **Frame Capture Busy Status**
画像フレーム取り込み状態を確認できます。
0： フレーム取り込み停止中
1： フレーム取り込み動作中

Bit 2 : **Frame Capture Start/Stop Flag**
画像フレーム取り込みの設定状態を確認できます。
シングルフレーム取り込み時は動作設定後、フレーム取り込み終了で自動的に停止設定になります。
0： フレーム取り込み停止設定
1： フレーム取り込み動作設定

Bit 1 : **Frame Capture Interrupt Status**
このビットを読み込むことでフレーム割り込みステータスを確認できます。
0： フレーム割り込みが発生していません
1： フレーム割り込みが発生しています
注意： **フレーム割り込みはカメラフレーム制御レジスタ (CAM[0x24]) の bit7 = 1 または bit0 = 1 のときに発生します。**

7.5 動作説明

本デバイスでは、最大 UXGA サイズのカメラモジュール（イメージセンサ）を接続することができます。カメラインタフェースは 8 ビットのデータバスを使用して YUV4:2:2 形式の画像データをクロックで受信します。

カメラの使用可否については必ず AC 特性を確認してください。

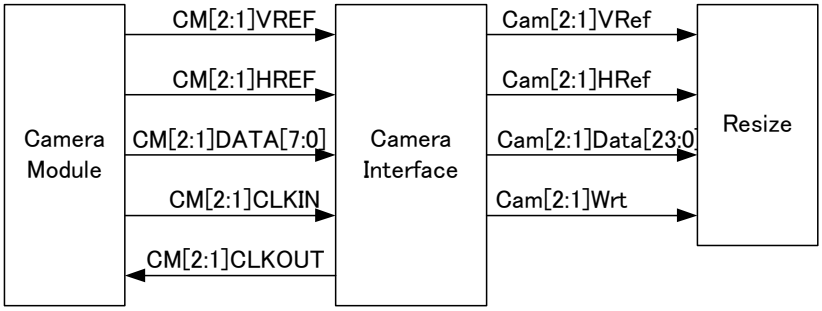


図 7.2　カメラインタフェース接続図

表 7.8　内部信号

内部信号名	説明
CM[2:1]VREF	カメラモジュールから来る垂直データ有効信号
CM[2:1]HREF	カメラモジュールから来る水平データ有効信号
CM[2:1]DATA[7:0]	カメラモジュールから来る 8 ビットデータ信号 ITU-R BT.601 フォーマットにも対応
CM[2:1]CLKIN	カメラモジュールから来るピクセルクロック
CM[2:1]CLKOUT	カメラモジュールを動作させるクロック
Cam[2:1]VRef	リサイズ回路へ送る垂直データ有効信号
Cam[2:1]HRef	リサイズ回路へ送る水平データ有効信号
Cam[2:1]Data[23:0]	YUV4:2:2 を YUV4:4:4 に変換した上で 24 ビット化した信号
Cam[2:1]Wrt	リサイズ回路へ送るデータ有効信号

カメラインタフェースはカメラモジュールから来る内部画像処理用クロックと非同期な信号を内部画像処理用クロックに同期させ、リサイズ回路にカメラ画像データを出力します。下図は CM[2:1]DATA、CM[2:1]HREF、CM[2:1]VREF を使用してカメラモジュールより出力された画像データを CM[2:1]CLKIN の立ち上がりエッジでデータサンプルする回路を表したものです。

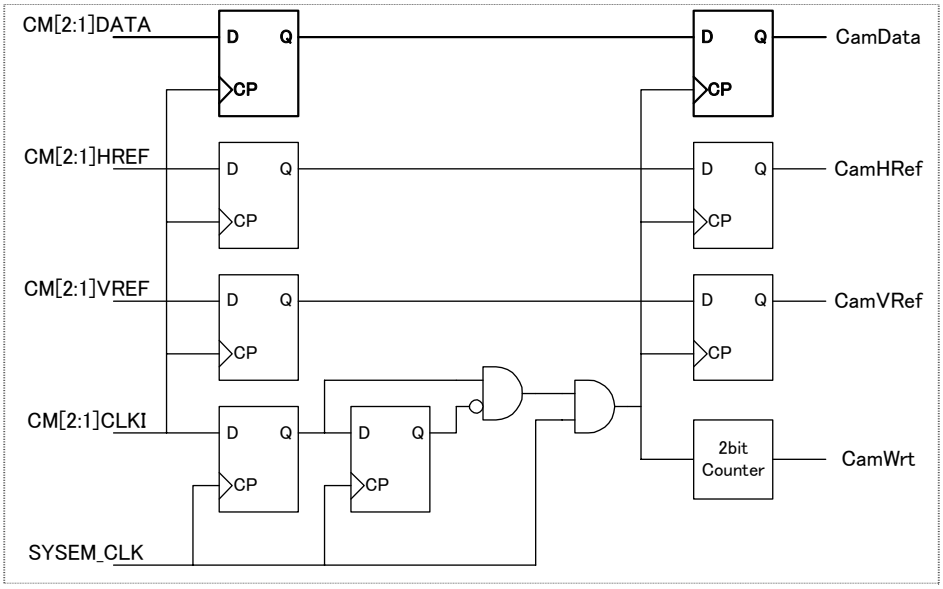


図 7.3　データサンプル回路

通常サンプリングモード（カメラモード設定レジスタ CAMx[0x20] bit10=0）では、カメラクロックモジュールから来るクロックを内部システムクロックでサンプリングし、クロックエッジを検出している機構から、理論上内部システムクロックはカメラモジュールから来るピクセルクロックの 2 倍以上の周波数で無ければなりません。現実的な問題としてクロックデューティ比などの要因により、2 倍の周波数で動作するかどうかはシステム依存となり、確実に動作させるためには 3 倍以上の周波数である必要があります。

高速サンプリングモード（カメラモード設定レジスタ CAMx[0x20] bit10=1）では、通常サンプリングモードの回路を 2 系統使用して 1 サイクルごとにトグルさせて使用するため、サンプリング性能が 2 倍になります。

7.5.1 フレームキャプチャ割り込み

カメラ画像データの VREF 信号から割り込みを発生させることができます。この割り込みは JPEG エンコードなどのカメラ画像を使用した画像処理ルーチンで必要な割り込みです。

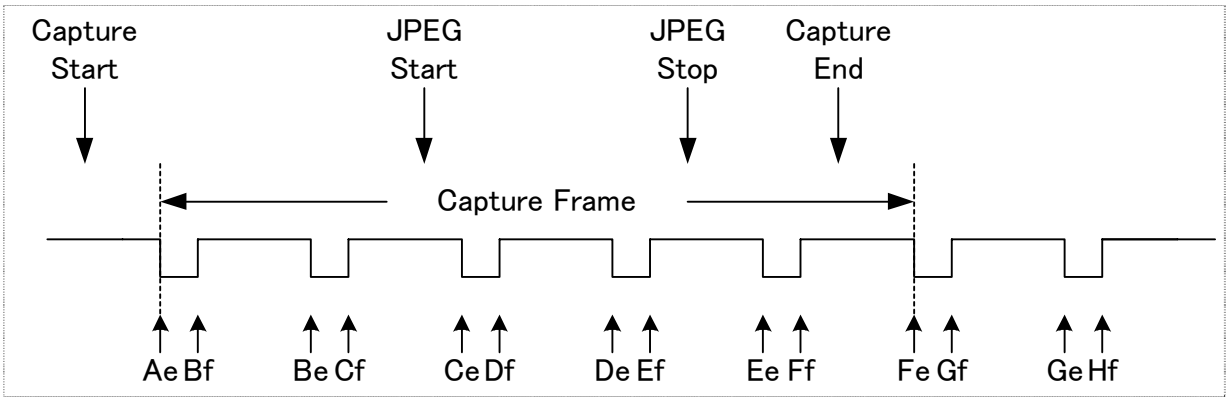


図 7.4　割り込み発生タイミング

図 7.4 はカメラインタフェースのキャプチャを有効にしてから無効にするまでの一連のタイミングチャートで、カメラフレーム割り込みに関係するアクションが示してあります。Capture Start はカメラインタフェースのカメラモード設定レジスタビット 0 を“1”にしたことを意味します。同様に Capture End はカメラモード設定レジスタビット 0 を“0”にしたことを意味します。JPEG Start は JPEG

スタートストップ制御レジスタに“1”を書いたことを意味し、JPEG End は JPEG スタートストップ制御レジスタに“0”を書いたことを意味します。Capture Frame はカメラモジュールから来たデータをリサイズ回路に流すフレームを表しています。

フレームキャプチャの割り込みタイミングを決定するレジスタ設定は全部で 4 つあります。表 7.9 は 4 つのレジスタビットの設定により割り込みが上の図のどのタイミングで発生するかを示しています。通常使用するレジスタ設定は上から 4 番目の設定（網掛け部分）になります。

表 7.9　割り込み発生タイミング

カメラフレーム制御レジスタ CAM[0x24]				カメラフレーム割り込みが発生するタイミング
ビット 7	ビット 5	ビット 1	ビット 0	
0	x	x	0	割り込み発生なし
0	0	0	1	Ce, De
0	0	1	1	Df, Ef
0	1	0	1	Be, Ce, De, Ee, Fe
0	1	1	1	Cf, Df, Ef, Ff, Gf
1	x	0	x	Fe
1	x	1	x	Gf

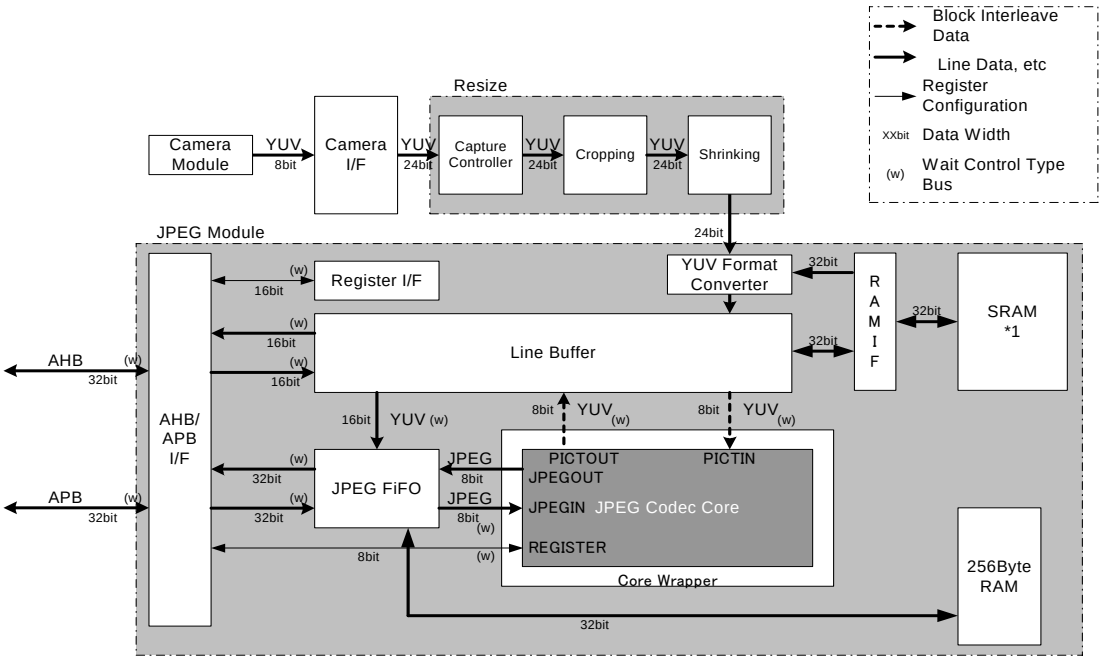
8. JPEG コントローラ (JPG[2:1])

8.1 概要

カメラ入力画像を JPEG エンコードする機能、および YUV データとしてキャプチャする機能を実現しています。JPEG エンコード方式は JPEG の Baseline 方式にほぼ準拠しています。演算精度は JPEG Part2 (ISO/IEC10918-2) に準拠しており十分な演算精度を持っています。最大画像サイズは VGA です。JPEG エンコードできる画像は MCU サイズ以上の YUV フォーマットの最小単位画像となっています。量子化テーブルは圧縮時 2 テーブル設定可能で、伸長時は 4 テーブルまで対応可能になっています。ハフマンテーブル DC/AC はそれぞれ 2 つのテーブルを扱うことができます。任意のマーカをエンコード時最大 3 6 byte まで挿入可能です。処理対象マーカは SOI、SOF0、SOS、DQT、DHT、DRI、RSTm、EOI で伸長時には自動解読されます。DNL マーカには対応していません。対応 YUV フォーマットはカメラ画像のエンコード時が YUV4:4:4、YUV4:2:2、YUV4:1:1、YUV4:2:0、YUV データキャプチャ時が YUV4:2:2、YUV:4:2:0 です。グレー画像や RGB 画像には対応していません。処理速度は VGA サイズで最小 1/30 秒ですが、画像データやハフマンテーブル、量子化テーブルに大きく影響を受けますので常にその処理速度が出ることは保証できません。

2 つのカメラモジュールに対応するために、全く同等の機能を持つ 2 チャネルの JPEG コントローラ[2:1]を持っています。

8.2 ブロック図



*1：SRAM の割り当ては、SYS[0x68]の Bits[5:4] EMBRAMSEL[1:0]を参照。

図 8.1 JPEG コントローラブロック図

8.3 外部端子

JPEG コントローラに関連する外部端子はありません。

8.4 レジスタ

以下にレジスター覧を示します。全く同じ JPEG コントローラが 2 チャネルあります（JPEG1, JPEG2）ので、ベースアドレスが異なるだけで、内容は全く同じ機能を持つレジスタになります。

8.4.1 レジスター覧

表 8.1 レジスター覧

Address Offset	Register Name	Default Value	R/W	Data Access Size
Resizer Operation Registers (RSZ[2:1]) : Base Address =				**0xFFFE_9000(JPG1) 0xFFFD_9000(JPG2)**
0x60	グローバルリサイザ制御レジスタ	0x0000	WO	16bit
0x64	キャプチャ制御ステートレジスタ	0x0000	RO	16bit
0x68	キャプチャデータ設定レジスタ	0x0000	R/W	16bit
0x70-0x7C	予約レジスタ	-	-	-
0xC0	取り込みリサイズ制御レジスタ	0x0000	R/W	16bit
0xC8	取り込みリサイズスタート X 座標レジスタ	0x0000	R/W	16bit
0xCC	取り込みリサイズスタート Y 座標レジスタ	0x0000	R/W	16bit
0xD0	取り込みリサイズエンド X 座標レジスタ	0x027F	R/W	16bit
0xD4	取り込みリサイズエンド Y 座標レジスタ	0x01DF	R/W	16bit
0xD8	取り込みリサイズ縮小率レジスタ	0x8080	R/W	16bit
0xDC	取り込みリサイズ縮小モードレジスタ	0x0000	R/W	16bit
JPEG Module Registers (JCTL[2:1]) : Base Address =				**0xFFFE_A000(JPG1) 0xFFFD_A000(JPG2)**
0x00	JPEG 制御レジスタ	0x0000	R/W	16bit
0x04	JPEG ステータスフラグレジスタ	0x8080	R/W	16bit
0x08	JPEG ロウステータスフラグレジスタ	0x8080	RO	16bit
0x0C	JPEG 割り込み制御レジスタ	0x0000	R/W	16bit
0x10	予約レジスタ	-	-	-
0x14	JPEG コーデックスタートストップ制御レジスタ	0x0000	WO	16bit
0x18 – 0x1C	予約レジスタ	-	-	-
0x20	ハフマンテーブル自動設定レジスタ	0x0000	R/W	16bit
JPEG FIFO Setting Registers (JFIFO[2:1]) : Base Address =				**0xFFFE_A000(JPG1) 0xFFFD_A000(JPG2)**
0x40	JPEG FIFO 制御レジスタ	0x0000	R/W	16bit
0x44	JPEG FIFO ステータスレジスタ	0x8001	RO	16bit
0x48	JPEG FIFO サイズレジスタ	0x003F	R/W	16bit
0x4C	JPEG FIFO リード/ライトポートレジスタ	0x0000 0000	R/W	32bit
0x50 – 0x58	予約レジスタ	-	-	-
0x60	エンコードサイズリミットレジスタ 0	0x0000	R/W	16bit
0x64	エンコードサイズリミットレジスタ 1	0x0000	R/W	16bit
0x68	エンコードサイズ結果レジスタ 0	0x0000	RO	16bit
0x6C	エンコードサイズ結果レジスタ 1	0x0000	RO	16bit
0x70 – 0x78	予約レジスタ	-	-	-
JPEG Line Buffer Setting Registers (JLB[2:1]) : Base Address =				**0xFFFE_A000(JPG1) 0xFFFD_A000(JPG2)**
0x80	JPEG ラインバッファステータスフラグレジスタ	0x0000	R/W	16bit
0x84	JPEG ラインバッファロウステータスフラグレジスタ	0x0000	RO	16bit
0x88	JPEG ラインバッファカレントステータスフラグレジスタ	0xX009	RO	16bit
0x8C	JPEG ラインバッファ割り込み制御レジスタ	0x0000	R/W	16bit
0x90 – 0x9C	予約レジスタ	-	-	-

8. JPEG コントローラ (JPG[2:1])

Address Offset	Register Name	Default Value	R/W	Data Access Size
0xA0	JPEG ラインバッファ水平ピクセル許容サイズレジスタ	0x2800	R/W	16bit
0xA4	JPEG ラインバッファメモリアドレスオフセットレジスタ	0x0020	R/W	16bit
0xA8 – 0xBC	予約レジスタ	-	-	-
0xC0	JPEG ラインバッファリード/ライトポートレジスタ	0x0000 0000	R/W	32bit
JPEG Codec Registers (JCODEC[2:1]) : Base Address = 0xFFFE_B000(JPG0) 0xFFFD_B000(JPG1)				
0x00	動作モード設定レジスタ	0x0000	R/W	16bit
0x04	コマンド設定レジスタ	Not applicable	WO	16bit
0x08	JPEG 動作ステータスレジスタ	0x0000	RO	16bit
0x0C	量子化テーブル番号レジスタ	0x0000	R/W	16bit
0x10	ハフマンテーブル番号レジスタ	0x0000	R/W	16bit
0x14	DRI 設定レジスタ 0	0x0000	R/W	16bit
0x18	DRI 設定レジスタ 1	0x0000	R/W	16bit
0x1C	垂直ピクセルサイズレジスタ 0	0x0000	R/W	16bit
0x20	垂直ピクセルサイズレジスタ 1	0x0000	R/W	16bit
0x24	水平ピクセルサイズレジスタ 0	0x0000	R/W	16bit
0x28	水平ピクセルサイズレジスタ 1	0x0000	R/W	16bit
0x2C - 0x34	予約レジスタ	-	-	-
0x38	RST マーカ動作設定レジスタ	0x0000	R/W	16bit
0x3C	RST マーカ動作ステータスレジスタ	0x0000	RO	16bit
0x40 – 0xCC	挿入マーカデータレジスタ	0x00FF	R/W	16bit
0x400 – 0x4FC	量子化テーブル No.0 レジスタ	Not applicable	R/W	16bit
0x500 – 0x5FC	量子化テーブル No.1 レジスタ	Not applicable	R/W	16bit
0x800 – 0x83C	DC ハフマンテーブル No.0 レジスタ 0	Not applicable	WO	16bit
0x840 – 0x86C	DC ハフマンテーブル No.0 レジスタ 1	Not applicable	WO	16bit
0x880 – 0x8BC	AC ハフマンテーブル No.0 レジスタ 0	Not applicable	WO	16bit
0x8C0 – 0xB44	AC ハフマンテーブル No.0 レジスタ 1	Not applicable	WO	16bit
0xC00 – 0xC3C	DC ハフマンテーブル No.1 レジスタ 0	Not applicable	WO	16bit
0xC40 – 0xC6C	DC ハフマンテーブル No.1 レジスタ 1	Not applicable	WO	16bit
0xC80 – 0xCBC	AC ハフマンテーブル No.1 レジスタ 0	Not applicable	WO	16bit
0xCC0 – 0xF44	AC ハフマンテーブル No.1 レジスタ 1	Not applicable	WO	16bit

以下に各レジスタの詳細について説明します。

8.4.2 Resizer Operation Registers (RSZ[2:1])

注意：リサイズのレジスタは一部を除き、カメラインタフェースからのデータを受け取っている間は変更できません。

グローバルリサイザ制御レジスタ
RSZ[0x60]　初期値 = 0x0000　Write Only

15	14	13	12	11	10	9	8
n/a					Reserved		ACTAGAIN
7	6	5	4	3	2	1	0
n/a			Reserved		n/a	Reserved	

Bit [10:9]：　**予約**
必ず各ビットに“0”を設定してください。

Bit 8：　**ACTAGAIN（ライトオンリ）**
このビットは連続エンコードを実行するときに使用します。このビットに 1 を書くと現フレームに引き続き次のフレームも JPEG コーデック回路に画像が送られます。
連続エンコードの実現可否はシステムおよびソフトウェア仕様に依存します。詳細は動作説明を参照してください。

Bit [4:3]：　**予約**
必ず、各ビットに“0”を設定してください。

Bits [1:0]：　**予約**
必ず、各ビットに“0”を設定してください。

キャプチャ制御ステートレジスタ
RSZ[0x64]　初期値 = 0x0000　Read Only

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
n/a				State Value			

Bit [3:0]：　**ステート値**
キャプチャ制御シーケンスステートマシンの現在のステート値を示しています。ステート値の意味は、動作説明「8.5.1 キャプチャ制御機能」を参照してください。

キャプチャデータ設定レジスタ
RSZ[0x68]　初期値 = 0x0000　Read/Write

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
n/a							Data Format Select

Bit 0：　**取り込み画像データフォーマット選択**
取り込み画像のデータフォーマットを選択します。このビットを“1”に設定した場合、RSZ[0xC0]以外の RSZ[*]レジスタの設定は全て無効となります。（ReadOnly のレジスタの中には状態が変化するものもありますが、有効値ではありません。）
0： YUV データ
1： JPEG データ

予約レジスタ

RSZ[0x70-7C]　初期値 = 0x0000　Read/Write

15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							

Bit [15:0]：　**予約**

必ず、各ビットに“0”を設定してください。

取り込みリサイズ制御レジスタ

RSZ[0xC0]　初期値 = 0x0000　Read/Write

15	14	13	12	11	10	9	8
n/a							
Capture Resizer Software Reset (WO)	n/a			Reserved (0)			Capture Resizer Enable
7	6	5	4	3	2	1	0

Bit 7：　**取り込みリサイズソフトウェアリセット（ライトオンリ）**

このビットに“1”を書くことで取り込みリサイズはソフトウェアリセットされます。“0”を書いたときには何も起きません。

Bits [3:1]：　**予約**

必ず各ビットに“0”を設定してください。

Bit 0：　**取り込みリサイズイネーブル**

このビットが“0”のとき、リサイズモジュールへのクロック供給が停止するため、低消費電力対策に有効です。このビットが“0”であっても、リサイズモジュールのレジスタ群へのリード/ライトは可能です。

ライト時

0： 取り込みリサイズはディセーブルになります。

1： 取り込みリサイズはイネーブルになります。

リード時

0： 取り込みリサイズはディセーブルです。

1： 取り込みリサイズはイネーブルです。

取り込みリサイズスタート X 座標レジスタ

RSZ[0xC8]　初期値= 0x0000　Read/Write

15	14	13	12	11	10	9	8
n/a					Capture Resizer Start X Position bits [10:8]		
Capture Resizer Start X Position bits [7:0]							
7	6	5	4	3	2	1	0

Bits [10:0]：　**取り込みリサイズスタート X 座標 [10:0]**

これらのビットは取り込みリサイズの取り込み開始 X 座標を設定します。

取り込みリサイズスタート Y 座標レジスタ

RSZ[0xCC]　初期値= 0x0000　Read/Write

n/a					Capture Resizer Start Y Position bits [10:8]		
15	14	13	12	11	10	9	8
Capture Resizer Start Y Position bits [7:0]							
7	6	5	4	3	2	1	0

Bits [10:0]：　**取り込みリサイズスタート Y 座標 [10:0]**

これらのビットは取り込みリサイズの取り込み開始 Y 座標を決定します。

取り込みリサイズエンド X 座標レジスタ

RSZ[0xD0]　初期値 = 0x027F　Read/Write

n/a					Capture Resizer End X Position bits [10:8]		
15	14	13	12	11	10	9	8
Capture Resizer End X Position bits [7:0]							
7	6	5	4	3	2	1	0

Bits [10:0]：　**取り込みリサイズエンド X 座標 [10:0]**

これらのビットは取り込みリサイズの取り込み終了 X 座標を決定します。

取り込みリサイズエンド Y 座標レジスタ

RSZ[0xD4]　初期値 = 0x01DF　Read/Write

n/a					Capture Resizer End Y Position bits [10:8]		
15	14	13	12	11	10	9	8
Capture Resizer End Y Position bits [7:0]							
7	6	5	4	3	2	1	0

Bits [10:0]：　**取り込みリサイズエンド Y 座標 [10:0]**

これらのビットは取り込みリサイズの取り込み終了 Y 座標を決定します。

取り込みリサイズ縮小率レジスタ
RSZ[0xD8]　　初期値 = 0x8080　　Read/Write

15	14	13	12	11	10	9	8
Reserved (0)							

7	6	5	4	3	2	1	0
Reserved (0)				Capture Resizer Scaling Rate bits [3:0]			

Bits [15:4] :　**予約**
必ず各ビットに“0”を設定してください。

Bits [3:0] :　**取り込みリサイズ縮小率 [3:0]**
これらのビットは取り込みリサイズの縮小率を設定します。取り込みリサイズ縮小モードレジスタの設定により利用できない縮小率があります。詳しくは取り込みリサイズ縮小モードレジスタの説明にある *“表 8.4　取り込みリサイズ縮小率/モード選択”* を参照してください。

表 8.2　取り込みリサイズ縮小率設定

bits [3:0]	取り込みリサイズ縮小率設定
0000	予約
0001	1/1
0010	1/2
0011	1/3（縦横方向間引きモードのみ、取り込みリサイズ縮小モードレジスタビット 1-0=01）
0100	1/4
0101	1/5（縦横方向間引きモードのみ、取り込みリサイズ縮小モードレジスタビット 1-0=01）
0110	1/6（縦横方向間引きモードのみ、取り込みリサイズ縮小モードレジスタビット 1-0=01）
0111	1/7（縦横方向間引きモードのみ、取り込みリサイズ縮小モードレジスタビット 1-0=01）
1000	1/8
1001-1111	予約

取り込みリサイズ縮小モードレジスタ

RSZ[0xDC]　　初期値 = 0x0000　　Read/Write

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
n/a				Reserved (0)		Capture Resizer Scaling Mode bits [1:0]	

Bits [3:2] :　**予約**

必ず各ビットに“0”を設定してください。

Bits [1:0] :　**取り込みリサイズ縮小モード [1:0]**

これらのビットは取り込みリサイズ縮小モードを設定します。縮小モードによっては取り込みリサイズ縮小率レジスタで設定できない縮小率があります。詳しくは *“表 8.4　取り込みリサイズ縮小率/モード選択”* を参照してください。

表 8.3　取り込みリサイズ縮小モード選択

Bits [1:0]	Capture Resizer Scaling Mode
00	縮小なし
01	縦方向、横方向ともに間引き
10	縦方向間引き、横方向加算平均
11	予約

表 8.4　取り込みリサイズ縮小率/モード選択

		RSZ[0xDC] Bits [1:0]			
		00	**01**	**10**	**11**
RSZ[0xD8] Bits [3:0]	**0000**	1/1	予約	予約	予約
	0001	1/1	予約	予約	予約
	0010	1/1	1/2	1/2	予約
	0011	1/1	1/3	予約	予約
	0100	1/1	1/4	1/4	予約
	0101	1/1	1/5	予約	予約
	0110	1/1	1/6	予約	予約
	0111	1/1	1/7	予約	予約
	1000	1/1	1/8	1/8	予約
	other	予約	予約	予約	予約

8.4.3 JPEG Module Registers (JCTL[2:1])

JPEG 制御レジスタ
JCTL[0x00] 初期値 = 0x0000 Read/Write

15	14	13	12	11	10	9	8
JPEG Encode Fast Mode	JPEG Marker Fast Output Mode	Reserved (0)					JPEG 180° Rotation Enable

7	6	5	4	3	2	1	0
JPEG Module SW Reset (WO)	Reserved (0)		UV Data Type Conversion	Operation Mode bits [2:0]			JPEG Module Enable

Bit15： **JPEG 高速エンコードモード**
0： 影響なし。
1： 固定されたハフマンテーブルを使用して JPEG 圧縮処理を促進します。
注意： このビットを“1”にして JPEG 圧縮を行う際には、JPEG ハフマンテーブル値（JCODEC[0x800]-[0xF44]の有効アドレス）を「ISO/IEC 10918-1 Annex K」に記載されている値に設定するか、もしくは JCTL[0x20]ハフマン自動設定レジスタを“1”に設定する必要があります。

Bit14： **JPEG マーカ高速出力モード**
0： 影響なし。
1： 固定されたハフマンテーブルを使用して JPEG マーカ出力を促進します。
注意： このビットの設定は高速エンコードモード (Bit15) が“1”のときのみ有効です。このビットを“1”にして JPEG 圧縮を行う際には、JPEG ハフマンテーブル値（JCODEC[0x800]-[0xF44]の有効アドレス）を書き込む必要はありません。（「ISO/IEC 10918-1 Annex K」に記載されている値が使用されます。）

Bits [13:9]： **予約**
必ず各ビットに“0”を設定してください。

Bit 8： **JPEG 180° 回転イネーブル**
JPEGエンコードデータを回転させるかどうか選択します。ハードウェアが回転するのは特定のライン単位になりますので、ソフトウェアでJPEGファイル中のデータの並べ替えが必要となります。詳しくは“***8.5.5.5 JPEG180° 回転エンコード***”を参照してください。
0： 回転しない
1： 180° 回転させる

Bit 7： **JPEG モジュールソフトウェアリセット（ライトオンリ）**
JPEG コーデックを除く JPEG モジュールをリセットします。レジスタはリセットされません。JPEG エンコード開始前には必ずリセットしてください。このリセットを実行するためには JPEG モジュールイネーブルに“1”が設定されている必要があります。JPEG モジュールがイネーブルになっていない状態では、このビットに“1”を書いてもソフトウェアリセットが実行されない場合があります。
0： 影響なし
1： リセット

Bit [6:5] **予約**
必ず各ビットに“0”を設定してください。

Bit 4： **UV データタイプ変換**
カメラから入力される UV タイプを変換します。

表 8.5 UV Data Type

Bit 4	Camera Data Type	Internal Data Type
0 (変換)	0 ≦ U ≦ 255 0 ≦ V ≦ 255	-128 ≦ U ≦ 127 -128 ≦ V ≦ 127
	16 ≦ Cb ≦ 240 16 ≦ Cr ≦ 240	-112 ≦ Cb ≦ 112 -112 ≦ Cr ≦ 112
	-128 ≦ U ≦ 127 -128 ≦ V ≦ 127	0 ≦ U ≦ 255 0 ≦ V ≦ 255
	-112 ≦ Cb ≦ 112 -112 ≦ Cr ≦ 112	16 ≦ Cb ≦ 240 16 ≦ Cr ≦ 240
1 (非変換)	0 ≦ U ≦ 255 0 ≦ V ≦ 255	0 ≦ U ≦ 255 0 ≦ V ≦ 255
	16 ≦ Cb ≦ 240 16 ≦ Cr ≦ 240	16 ≦ Cb ≦ 240 16 ≦ Cr ≦ 240
	-128 ≦ U ≦ 127 -128 ≦ V ≦ 127	-128 ≦ U ≦ 127 -128 ≦ V ≦ 127
	-112 ≦ Cb ≦ 112 -112 ≦ Cr ≦ 112	-112 ≦ Cb ≦ 112 -112 ≦ Cr ≦ 112

Bits [3:1]： **動作モード選択**

JPEG 動作モードを設定します。YUV データキャプチャが選択されると JPEG コーデックへのクロック供給が停止され JPEG コーデックレジスタへアクセスできません。そのため JPEG コーデックレジスタの設定変更をする場合はカメラ画像 JPEG エンコード（YUV データキャプチャ以外）に設定してください。

カメラ画像 JPEG エンコード YUV フォーマットは JCODEC[0x00]bit1-0YUV フォーマット選択で設定しますが、YUV データキャプチャの YUV フォーマットはこのレジスタビットで設定します。

表 8.6 JPEG 動作モード

bits [3:1]	JPEG 動作モード
000	カメラ画像 JPEG エンコード（YUV4:4:4、YUV4:2:2、YUV4:1:1、YUV4:2:0 ただし YUV4:4:4 はリサイズにて 1/2 以下に縮小した画像のみ対応）（初期値）
001	予約
010	予約
011	YUV データキャプチャ (YUV 4:2:2)
100	予約
101	予約
110	予約
111	YUV データキャプチャ (YUV 4:2:0)

Bit 0： **JPEG モジュールイネーブル**

JPEG モジュールをイネーブルにします。ディセーブル時は JPEG モジュールへのクロック供給が停止され JPEG コーデックレジスタへアクセスできません。また JPEG モジュールのディセーブルは必ずリサイザをディセーブルする前に行ってください。

0： ディセーブル（初期値）

1： イネーブル

JPEG ステータスフラグレジスタ

JCTL[0x04] 初期値 = 0x8080 Read/Write

15	14	13	12	11	10	9	8
Reserved (1) -	JPEG Codec File Out Status (RO)	JPEG FIFO Threshold Status bits [1:0] (RO)		Encode Size Limit Violation Flag (R/W)	JPEG FIFO Threshold Trigger Flag (R/W)	JPEG FIFO Full Flag (R/W)	JPEG FIFO Empty Flag (R/W)
Reserved (1) -			Reserved (R/W)	Reserved -	JPEG Line Buffer Overflow Flag (RO)	JPEG Codec Interrupt Flag (RO)	JPEG Line Buffer Interrupt Flag (RO)
7	6	5	4	3	2	1	0

Bit 15 : **予約**

必ず“1”を設定してください。

Bit 14 : **JPEG コーデックファイルアウトステータス（リードオンリ）**

JPEG エンコード時の JPEG コーデック出力状態を示します。

0： JPEG ファイル出力停止中

1： JPEG エンコード中または JPEG ファイル出力中

ライト時には必ず“1”を設定してください。

Bits [13:12] : **JPEG FIFO しきい値ステータス（リードオンリ）**

JPEG FIFO の現在のデータ状態を示します。

ライト時には必ず“1”を設定してください。

表 8.7 JPEG FIFO Threshold Status

bits [13:12]	JPEG FIFO Threshold Status
00	エンプティ
01	4 バイト以上、FIFO サイズの 1/4 未満
10	FIFO サイズの 1/4 以上、1/2 未満
11	FIFO サイズの 1/2 以上

Bit 11 : **エンコードサイズリミット違反フラグ**

JPEG エンコード時に JPEG ファイルのサイズがエンコードサイズリミットレジスタに設定されるエンコードサイズリミットを超えたことを示す割り込みフラグです。このフラグは JPEG 割込み制御レジスタ (JCTL[0x0C] bit 11) によりディセーブルにできます。この機能はファイルサイズを超えたことを通知するだけで、JPEG エンコード動作は継続しています。

リード時

0： 割り込みなし

1： 割り込みあり（エンコードサイズオーバー発生）

ライト時

0： 影響なし

1： クリア

Bit 10 : **JPEG FIFO しきい値トリガフラグ**

JPEG FIFO のデータサイズが JPEG FIFO しきい値（JPEG FIFO 制御レジスタ bits 5-4）を少なくとも一回は超えたことを示す割り込みフラグです。このフラグは JPEG 割込み制御レジスタ (bit 10) によりディセーブルにできます。

リード時

0： 割り込みなし

1： 割り込みあり（JPEG FIFO しきい値オーバー発生）

ライト時

0： 影響なし

1： クリア

Bit 9 : **JPEG FIFO フルフラグ**

JPEG FIFO が少なくとも一回はフルになったことを示す割り込みフラグです。このフラグは JPEG 割込み制御レジスタ (bit 9) によりディセーブルにできます。

リード時

0 : 割り込みなし

1 : 割り込みあり（JPEG FIFO フル発生）

ライト時

0 : 影響なし

1 : クリア

Bit 8 : **JPEG FIFO エンプティフラグ**

JPEG FIFO が少なくとも一回はエンプティになったことを示す割り込みフラグです。このフラグは JPEG 割込み制御レジスタ (bit 8) によりディセーブルにできます。

リード時

0 : 割り込みなし

1 : 割り込みあり（JPEG FIFO エンプティ発生）

ライト時

0 : 影響なし

1 : クリア

Bit [7:5] : **予約**

各ビットに、必ず“1”を設定してください。

Bit 4 : **予約**

必ず、“0”を設定してください。

Bit 3 : **予約**

必ず、“1”を設定してください。

Bit 2 : **JPEG ラインバッファオーバーフローフラグ（リードオンリ）**

JPEG ラインバッファがオーバーフローしたことを示す割り込みフラグです。このフラグは JPEG 割込み制御レジスタ (bit 2) によりディセーブルにできます。

このフラグは JPEG モジュールソフトウェアリセット (bit 7) によりクリアされます。

0 : 割り込みなし

1 : 割り込みあり (JPEG ラインバッファオーバーフロー発生)

ライト時には必ず“1”を設定してください。

この機能はラインバッファがオーバーフローしたことを通知するだけで、JPEG モジュールは動作を継続しています。ただしデータはオーバーフローにより破壊されています。

Bit 1 : **JPEG コーデック割り込みフラグ（リードオンリ）**

JPEG コーデックが割り込みを発生したことを示す割り込みフラグです。このフラグは JPEG 割込み制御レジスタ (bit 1) によりディセーブルにできます。このフラグは JPEG 動作ステータス (bit 0) をリードすることでクリアされます。

0 : 割り込みなし

1 : 割り込みあり (JPEG コーデック割り込み発生)

Bit 0 : **予約（リードオンリ）**

JPEG ロウステータスフラグレジスタ

JCTL[0x08] 初期値 = 0x8080 Read Only

15	14	13	12	11	10	9	8
Reserved	Raw JPEG Codec File Out Status	Raw JPEG FIFO Threshold Status bits [1:0]		Raw Encode Size Limit Violation Flag	Raw JPEG FIFO Threshold Trigger Flag	Raw JPEG FIFO Full Flag	Raw JPEG FIFO Empty Flag
7	6	5	4	3	2	1	0
Reserved			Reserved	Reserved	Raw JPEG Line Buffer Overflow Flag	Raw JPEG Codec Interrupt Flag	Raw JPEG Line Buffer Interrupt Flag

Bit 15 : **予約**

Bit 14 : **JPEG コーデックファイルアウトロウステータス**

JPEG エンコード時の JPEG コーデック出力状態を示します。

0： JPEG ファイル出力停止中

1： JPEG エンコード中または JPEG ファイル出力中

Bits [13:12] : **JPEG FIFO しきい値ロウステータス**

JPEG FIFO の現在のデータ状態を示します。

表 8.8 JPEG FIFO Threshold Status

Bits [13:12]	JPEG FIFO Threshold Status
00	エンプティ
01	4 バイト以上、FIFO サイズ未満
10	FIFO サイズの 1/4 以上、1/2 未満
11	FIFO サイズの 1/2 以上

Bit 11 : **エンコードサイズリミット違反ロウフラグ**

JPEG エンコード時に JPEG ファイルのサイズがエンコードサイズリミット（エンコードサイズリミットレジスタ 0,1）を超えたことを示すフラグです。このフラグは JPEG 割込み制御レジスタ (bit 11) には影響されません。このフラグのクリアは JPEG ステータスフラグレジスタ bit 11 に “1” をライトすることで行われます。この機能はファイルサイズを超えたことを通知するだけで、JPEG エンコード動作は継続しています。

0： エンコードサイズオーバー発生なし

1： エンコードサイズオーバー発生あり

Bit 10 : **JPEG FIFO しきい値トリガロウフラグ**

JPEG FIFO のデータサイズが JPEG FIFO しきい値（JPEG FIFO 制御レジスタ bit5-4）を少なくとも一回は超えたことを示すフラグです。このフラグは JPEG 割込み制御レジスタ (bit 10) には影響されません。このフラグのクリアは JPEG ステータスフラグレジスタ bit 10 に“1” をライトすることで行われます。

0： JPEG FIFO しきい値オーバー発生なし

1： JPEG FIFO しきい値オーバー発生あり

Bit 9 : **JPEG FIFO フルロウフラグ**

JPEG FIFO が少なくとも一回はフルになったことを示すフラグです。このフラグは JPEG 割込み制御レジスタ (bit 9) には影響されません。このフラグのクリアは JPEG ステータスフラグレジスタ bit 9 に“1” をライトすることで行われます。

0： JPEG FIFO フル発生なし

1： JPEG FIFO フル発生あり

Bit 8 : **JPEG FIFO エンプティロウフラグ**
JPEG FIFO が少なくとも一回はエンプティになったことを示すフラグです。このフラグは JPEG 割込み制御レジスタ (bit 8) には影響されません。このフラグのクリアは JPEG ステータスフラグレジスタ bit 8 に“1” をライトすることで行われます。
0： JPEG FIFO エンプティ発生なし
1： JPEG FIFO エンプティ発生あり

Bit [7:5] : **予約**

Bit 4 : **予約**

Bit 3 : **予約**

Bit 2 : **JPEG ラインバッファオーバーフローロウフラグ**
JPEG ラインバッファがオーバーフローしたことを示すフラグです。このフラグは JPEG 割込み制御レジスタ (JCTL[0x0C] bit 2) には影響されません。このフラグは JPEG モジュールソフトウェアリセット (JCTL[0x00] bit 7) によりクリアされます。
0： JPEG ラインバッファオーバーフロー発生なし
1： JPEG ラインバッファオーバーフロー発生あり

Bit 1 : **JPEG コーデック割り込みロウフラグ**
JPEG コーデックが割り込みを発生したことを示すフラグです。このフラグは JPEG 割込み制御レジスタ (bit 1) には影響されません。このフラグは JPEG 動作ステータスレジスタ (bit 0) をリードすることでクリアされます。
0： JPEG コーデック割り込み発生なし
1： JPEG コーデック割り込み発生あり

Bit 0 : **JPEG ラインバッファ割り込みロウフラグ**
YUV データキャプチャで使用し、JPEG ラインバッファ割り込みが発生したことを示すフラグです。このフラグは JPEG 割込み制御レジスタ (bit 0) には影響されません。このフラグは JPEG ラインバッファ割り込みフラグレジスタをクリアすることでクリアされます。
0： JPEG ラインバッファ割り込み発生なし
1： JPEG ラインバッファ割り込み発生あり

JPEG 割り込み制御レジスタ

JCTL[0x0C]　　初期値 = 0x0000　　Read/Write

15	14	13	12	11	10	9	8
Reserved (0)				Encode Size Limit Violation Interrupt Enable	JPEG FIFO Threshold Trigger Interrupt Enable	JPEG FIFO Full Interrupt Enable	JPEG FIFO Empty Interrupt Enable
7	6	5	4	3	2	1	0
Reserved (0)			Reserved	Reserved (0)	JPEG Line Buffer Overflow Interrupt Enable	JPEG Codec Interrupt Enable	JPEG Line Buffer Interrupt Enable

Bits [15:12]：　**予約**
必ず、各ビットに“0”を設定してください。

Bit 11：　**エンコードサイズリミット違反割り込みイネーブル**
エンコードサイズリミット違反割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 10：　**JPEG FIFO しきい値トリガ割り込みイネーブル**
JPEG FIFO しきい値トリガ割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 9：　**JPEG FIFO フル割り込みイネーブル**
JPEG FIFO フル割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 8：　**JPEG FIFO エンプティ割り込みイネーブル**
JPEG FIFO エンプティ割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bits [7:5]：　**予約**
必ず、各ビットに“0”を設定してください。

Bit 4：　**予約**
必ず、“0”を設定してください。

Bit 3：　**予約**
必ず、“0”を設定してください。

Bit 2：　**JPEG ラインバッファオーバーフロー割り込みイネーブル**
JPEG ラインバッファオーバーフロー割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 1：　**JPEG コーデック割り込みイネーブル**
JPEG コーデック割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 0： **JPEG ラインバッファ割り込みイネーブル**

JPEG ラインバッファ割り込みをイネーブルにします。このビットは JPEG ラインバッファ割り込み制御レジスタのイネーブル制御の根元となるイネーブルです。このビットがディセーブルである場合は JPEG ラインバッファ割り込み制御レジスタのどのビットをイネーブルにしても JPEG ラインバッファ割り込み制御レジスタで制御される JPEG ラインバッファの割り込みがイネーブルになることはありません。JPEG ラインバッファオーバーフロー割り込みはこのイネーブル制御とは関係ありません。

0： ディセーブル（初期値）
1： イネーブル

JPEG コーデックスタートストップ制御レジスタ

JCTL[0x14] 初期値 = 0x0000 Write Only

n/a							
15	14	13	12	11	10	9	8
n/a							JPEG Start/Stop Control
7	6	5	4	3	2	1	0

Bit 0： **JPEG スタート/ストップ制御**

JPEG モジュール動作（YUV データキャプチャ含む）の制御を行います。

JPEG エンコード時

0： エンコード キャンセル（エンコード前の場合エンコード開始をキャンセル）
1： エンコード開始（次フレームをエンコード）

YUV データキャプチャ時

0： キャプチャ停止（現フレーム後停止）
1： キャプチャ開始（次フレームから開始）

ハフマンテーブル自動設定レジスタ

JCTL[0x20] 初期値 = 0x0000 Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a						ハフマンテーブル自動設定非待機モード	ハフマンテーブル自動設定
7	6	5	4	3	2	1	0

Bit 1： **ハフマンテーブル自動設定時の非待機モード**

JPEG コーデックコアをエンコードに設定時に、Bit0 に“1”を書き込むと同時にこのビットに“1”を書き込むと、JCODEC レジスタ以外のレジスタへのアクセスが可能になります。
Bit 0 に“1”を書き込むのと同時に“0”を書き込んだ場合、ハフマンテーブルの設定終了後に他のレジスタアドレスへのアクセスが可能になります。
Bit0 に対する有効な“1”書き込みと同時に書き込んだとき以外は無効です。
読み出し時は“0”です。

Bit 0： **ハフマンテーブル自動設定**

JPEG コーデックコアをエンコードに設定時に“1”を書き込むと JPEG ハフマンテーブル値（JCODEC[0x800]-[0xF44]の有効アドレス）に「ISO/IEC 10918-1 Annex K」記載の値を自動的に書き込みます。
エンコード設定時以外や“0”を書き込んだときには何も起こりません。
このビットが“1”のときは、JCODEC レジスタにはアクセス出来ません。（ダミーライト扱いになります。）
このビットが“1”のときは、本レジスタアドレスへの書き込みアクセスは禁止です。
このビットは“1”を書き込まれた後、ハフマンテーブル設定終了後に自動的にクリア（“0”）されます。

8.4.4 JPEG FIFO Setting Register (JFIFO[2:1])

JPEG FIFO 制御レジスタ
JFIFO[0x40]　　初期値 = 0x0000　　Read/Write

Reserved (0)							
15	14	13	12	11	10	9	8
Reserved (0)		JPEG FIFO Trigger Threshold bits [1:0]		Reserved (0)	JPEG FIFO Clear	JPEG FIFO Direction (RO)	Reserved (0)
7	6	5	4	3	2	1	0

Bits [15:6]：　**予約**
必ず、各ビットに“0”を設定してください。

Bits [5:4]：　**JPEG FIFO トリガしきい値**
JPEG FIFO のトリガのしきい値を設定します。

表 8.9　JPEG FIFO Trigger Threshold Selection

Bits [5:4]	JPEG FIFO Trigger Threshold
00	トリガなし
01	4 バイト以上でトリガ
10	FIFO サイズの 1/4 以上でトリガ
11	FIFO サイズの 1/2 以上でトリガ

Bit 3：　**予約**
必ず、“0”を設定してください。

Bit 2：　**JPEG FIFO クリア**
JPEG FIFO をクリアします。JPEG FIFO をクリアしたときは必ず JPEG モジュールをリセット（JPEG 制御レジスタ bit 7）してください。
0： 影響なし
1： JPEG FIFO クリア

Bit 1　　**JPEG FIFO コーデック方向（リードオンリ）**
JPEG FIFO の方向を示します。
0： 受信（JPEG エンコード設定）
1： Reserved

Bit 0：　**予約**
必ず、“0”を設定してください。

JPEG FIFO ステータスレジスタ

JFIFO[0x44]　初期値 = 0x8001　Read Only

Reserved							
15	14	13	12	11	10	9	8
Reserved				JPEG FIFO Threshold Status bits [1:0]		JPEG FIFO Full Status	JPEG FIFO Empty Status
7	6	5	4	3	2	1	0

Bits [15:4]：　**予約**

Bits [3:2]：　**JPEG FIFO しきい値ステータス**

JPEG FIFO の現在のデータ状態を示します。

表 8.10　JPEG FIFO Threshold Status

Bits [3:2]	JPEG FIFO Threshold Status
00	エンプティ
01	4 バイト以上、FIFO サイズの 1/4 未満
10	FIFO サイズの 1/4 以上、1/2 未満
11	FIFO サイズの 1/2 以上

Bit 1：　**JPEG FIFO フルステータス**

JPEG FIFO の現在のフル状態を示します。

0： フルでない

1： フル

Bit 0：　**JPEG FIFO エンプティステータス**

JPEG FIFO の現在のエンプティ状態を示します。

0： エンプティでない

1： エンプティ

JPEG FIFO サイズレジスタ

JFIFO[0x48]　初期値 = 0x003F　Read/Write

Reserved (0)	JPEG FIFO Size bits [14:8]						
15	14	13	12	11	10	9	8
JPEG FIFO Size bits [7:0]							
7	6	5	4	3	2	1	0

Bit 15：　**予約**

必ず、“0”に設定してください。

Bits [14:0]：　**JPEG FIFO サイズ**

JPEG FIFO サイズをワード単位で設定します。JPEG FIFO の最大値は 64 ワードです。JPEG FIFO は専用 RAM を使用していますので、通常は最大値である 64 ワードを設定することになります。JPEG FIFO サイズ（ワード）= JPEG FIFO サイズ bits[14:0]+1

注意：　本レジスタは最大“0x003F”まで設定可能です。“0x003F”（初期値）でお使いになることを推奨します。その他“0x001F”，“0x000F”，“0x0007”，“0x0003”，“0x0001”，“0x0000”が設定可能です。それ以外の数値は設定しないでください。

JPEG FIFO リード／ライトポートレジスタ

JFIFO[0x4C]　　初期値 = 0x0000 0000　　Read/Write

JPEG FIFO Read/Write Port bits [31:24]							
31	30	29	28	27	26	25	24
JPEG FIFO Read/Write Port bits [23:16]							
23	22	21	20	19	18	17	16
JPEG FIFO Read/Write Port bits [15:8]							
15	14	13	12	11	10	9	8
JPEG FIFO Read/Write Port bits [7:0]							
7	6	5	4	3	2	1	0

Bits [31:0] :　**JPEG FIFO リード/ライトポート**

JPEG エンコード時および YUV データキャプチャ時の JPEG FIFO リードポートです。

予約レジスタ

JFIFO[0x50, 0x54, 0x58]　　初期値 = -　　-/-

Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0

エンコードサイズリミットレジスタ 0

JFIFO[0x60]　初期値 = 0x0000　Read/Write

Encode Size Limit bits [15:8]							
15	14	13	12	11	10	9	8
Encode Size Limit bits [7:0]							
7	6	5	4	3	2	1	0

エンコードサイズリミットレジスタ 1

JFIFO[0x64]　初期値 = 0x0000　Read/Write

n/a							
15	14	13	12	11	10	9	8
Encode Size Limit bits [23:16]							
7	6	5	4	3	2	1	0

エンコードサイズリミットレジスタ 1　bits [7:0]
エンコードサイズリミットレジスタ 0　bits [15:0]：

エンコードサイズリミット bits[23:0]

JPEG エンコード時の JPEG ファイルのデータサイズリミットをバイト数で設定します。このレジスタ設定値以上の JPEG ファイルがエンコードされた場合はエンコードサイズリミット違反の割り込みが発生します。この値を超えても JPEG エンコード動作自体には影響はありません。

エンコードサイズ結果レジスタ 0

JFIFO[0x68]　初期値 = 0x0000　Read Only

Encode Size Result bits [15:8]							
15	14	13	12	11	10	9	8
Encode Size Result bits [7:0]							
7	6	5	4	3	2	1	0

エンコードサイズ結果レジスタ 1

JFIFO[0x6C]　初期値 = 0x0000　Read Only

n/a							
15	14	13	12	11	10	9	8
Encode Size Result bits [23:16]							
7	6	5	4	3	2	1	0

エンコードサイズ結果レジスタ 1　bits [7:0]
エンコードサイズ結果レジスタ 0　bits [15:0]：

エンコードサイズ結果 bits[23:0]

JPEG エンコード時の JPEG ファイルのデータサイズを示します。JPEG エンコード終了時のみ正しいサイズを表示します。

予約レジスタ

JFIFO[0x70, 0x74, 0x78]　初期値 = -　-/-

予約							
15	14	13	12	11	10	9	8
予約							
7	6	5	4	3	2	1	0

8.4.5 JPEG ラインバッファ設定レジスタ (JLB[2:1])

JPEG ラインバッファステータスフラグレジスタ

JLB[0x80] 初期値 = 0x0000 Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			予約	JPEG Line Buffer Empty Flag	JPEG Line Buffer Full Flag	JPEG Line Buffer Half Flag	予約
7	6	5	4	3	2	1	0

Bit 4 : **予約**
必ず、“1”に設定してください。

Bit 3 : **JPEG ラインバッファエンプティフラグ**
JPEG ラインバッファが少なくとも一回はエンプティになったことを示す割り込みフラグです。このフラグは JPEG ラインバッファ割込み制御レジスタ (bit 3) によりディセーブルにできます。
リード時
0： 割り込みなし
1： 割り込みあり（JPEG ラインバッファエンプティ発生）
ライト時
0： 影響なし
1： クリア

Bit 2 : **JPEG ラインバッファフルフラグ**
JPEG ラインバッファが少なくとも一回はフルになったことを示す割り込みフラグです。このフラグは JPEG ラインバッファ割込み制御レジスタ (bit 2) によりディセーブルにできます。
リード時
0： 割り込みなし
1： 割り込みあり（JPEG ラインバッファフル発生）
ライト時
0： 影響なし
1： クリア

Bit 1 : **JPEG ラインバッファハーフフルフラグ**
JPEG ラインバッファが少なくとも一回はハーフフルになったことを示す割り込みフラグです。このフラグは JPEG ラインバッファ割込み制御レジスタ (bit 1) によりディセーブルにできます。
リード時
0： 割り込みなし
1： 割り込みあり（JPEG ラインバッファハーフフル発生）
ライト時
0： 影響なし
1： クリア

Bit 0 : **予約**
必ず、“1”に設定してください。

JPEG ラインバッファロウステータスフラグレジスタ

JLB[0x84]　　初期値 = 0x0000　　Read Only

n/a							
15	14	13	12	11	10	9	8
n/a			Reserved	Raw JPEG Line Buffer Empty Flag	Raw JPEG Line Buffer Full Flag	Raw JPEG Line Buffer Half Flag	Reserved
7	6	5	4	3	2	1	0

Bit 4 :　**予約**

Bit 3 :　**JPEG ラインバッファエンプティロウフラグ**

JPEG ラインバッファが少なくとも一回はエンプティになったことを示すフラグです。このフラグは JPEG ラインバッファ割込み制御レジスタ (bit 3) には影響されません。

0 : JPEG ラインバッファエンプティ発生なし

1 : JPEG ラインバッファエンプティ発生あり

Bit 2 :　**JPEG ラインバッファフルロウフラグ**

JPEG ラインバッファが少なくとも一回はフルになったことを示すフラグです。このフラグは JPEG ラインバッファ割込み制御レジスタ (bit 2) には影響されません。

0 : JPEG ラインバッファフル発生なし

1 : JPEG ラインバッファフル発生あり

Bit 1 :　**JPEG ラインバッファハーフフルロウフラグ**

JPEG ラインバッファが少なくとも一回はハーフフルになったことを示すフラグです。このフラグは JPEG ラインバッファ割込み制御レジスタ (bit 1) には影響されません。

0 : JPEG ラインバッファハーフフル発生なし

1 : PEG ラインバッファハーフフル発生あり

Bit 0 :　**予約**

JPEG ラインバッファカレントステータスフラグレジスタ

JLB[0x88]　　初期値 = 0xX009　　Read Only

n/a							
15	14	13	12	11	10	9	8
n/a			Reserved	JPEG Line Buffer Empty Current Status	JPEG Line Buffer Full Current Status	JPEG Line Buffer Half Full Current Status	Reserved
7	6	5	4	3	2	1	0

Bit 4 :　**予約**

Bit 3 :　**JPEG ラインバッファエンプティカレントステータス**

JPEG ラインバッファの現在のエンプティ状態を示します。

0 : エンプティでない

1 : エンプティ

Bit 2 :　**JPEG ラインバッファフルカレントステータス**

JPEG ラインバッファの現在のフル状態を示します。

0 : フルでない

1 : フル

Bit 1 :　**JPEG ラインバッファハーフフルステータス**

JPEG ラインバッファの現在のハーフフル状態を示します。

0 : ハーフフルでない

1： ハーフフル

Bit 0： **予約**

JPEG ラインバッファ割り込み制御レジスタ

JLB[0x8C]　初期値 = 0x0000　Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			Reserved	JPEG Line Buffer Empty Interrupt Enable	JPEG Line Buffer Full Interrupt Enable	JPEG Line Buffer Half Full Interrupt Enable	Reserved
7	6	5	4	3	2	1	0

Bit4： **予約**
必ず、“0”に設定してください。

Bit 3： **JPEG ラインバッファエンプティ割り込みイネーブル**
JPEG ラインバッファエンプティ割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 2： **JPEG ラインバッファフル割り込みイネーブル**
JPEG ラインバッファフル割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 1： **JPEG ラインバッファハーフフル割り込みイネーブル**
JPEG ラインバッファハーフフル割り込みをイネーブルにします。
0： ディセーブル（初期値）
1： イネーブル

Bit 0： **予約**
必ず、“0”に設定してください。

JPEG ラインバッファ水平ピクセル許容サイズレジスタ

JLB[0xA0]　初期値 = 0x2800　Read/Write

Horizontal Support Size bits [10:4]							
15	14	13	12	11	10	9	8
Horizontal Support Size bits [3:0]				n/a	Horizontal Support Size Setting bits [2:0]		
7	6	5	4	3	2	1	0

Bits [15:4]： **JPEG ラインバッファ水平ピクセル許容サイズ（リードオンリ）**
Bits[2:0]によって設定された水平ピクセル許容サイズです。

Bits [2:0]： **JPEG ラインバッファ水平ピクセル許容サイズ設定**
JPEG ラインバッファが許容する水平ピクセルサイズを設定します。このレジスタを初期値以外で使用する場合は、JLB[0xA4]及びシステムコントローラのメモリ配分レジスタの設定と合わせて変更する必要があります。
000：水平ピクセルサイズ＝640 まで許容します。（初期値）
001：水平ピクセルサイズ＝800 まで許容します。
010：水平ピクセルサイズ＝1024 まで許容します。
011：水平ピクセルサイズ＝1280 まで許容します。
100：水平ピクセルサイズ＝1600 まで許容します。
101-111：設定不可

JPEG ラインバッファメモリアドレスオフセットレジスタ

JLB[0xA4]　初期値 = 0x0020　Read/Write

15	14	13	12	11	10	9	8
Reserved (0)							

7	6	5	4	3	2	1	0
Reserved (0)	JPG-LB Memory Address Offset bits [6:0]						

Bits [15:7] : **予約**

必ず、“0”に設定してください。

Bits [6:0] : **JPEG ラインバッファメモリアドレスオフセット**

JPEG ラインバッファ使用する内部メモリのアドレスオフセット値を 1KB アドレス単位で設定します。このレジスタを初期値以外で使用する場合は、システムコントローラのメモリ配分レジスタの設定と合わせて変更する必要があります。

JPEG ラインバッファリード/ライトポートレジスタ

JLB[0xC0]　初期値 = 0x0000_0000　Read/Write

31	30	29	28	27	26	25	24
JPEG Line Buffer Read/Write Port bits [31:24]							

23	22	21	20	19	18	17	16
JPEG Line Buffer Read/Write Port bits [23:16]							

15	14	13	12	11	10	9	8
JPEG Line Buffer Read/Write Port bits [15:8]							

7	6	5	4	3	2	1	0
JPEG Line Buffer Read/Write Port bits [7:0]							

Bits [31:0] : **JPEG ラインバッファリード/ライトポート**

Reserved

8.4.6 JPEG Codec Registers (JCODEC[2:1])

動作モード設定レジスタ
JCODEC[0x00] 初期値 = 0x0000 Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a			Reserved (0)	Marker Insert Enable	JPEG Operation Select	YUV Format Select bits [1:0]	
7	6	5	4	3	2	1	0

Bit 4 : **予約**
必ず、“0”を設定してください。

Bit 3 : **マーカ挿入イネーブル**
JPEG エンコード時に挿入マーカデータレジスタに書かれたデータの JPEG ファイルへの挿入をイネーブルにします。
0： ディセーブル（マーカ挿入なし）（初期値）
1： イネーブル（マーカ挿入あり）

Bit 2 : **JPEG 動作モード選択**
JPEG 動作モードを設定します。

表 8.11 JPEG Operation Selection

Bit 2	JPEG Operation
0（初期値）	エンコード
1	Reserved

Bits [1:0] : **YUV フォーマット選択**
JPEG エンコード時の YUV データ形式を設定します。リサイズで縮小処理を行っていない場合は YUV 4:4:4 形式の JPEG エンコードはできません。

表 8.12 YUV Format Selection

Bits [1:0]	YUV Format
00（初期値）	4:4:4
01	4:2:2
10	4:2:0
11	4:1:1

コマンド設定レジスタ

JCODEC[0x04]　初期値 = not applicable　Write Only

15	14	13	12	11	10	9	8
n/a							

7	6	5	4	3	2	1	0
JPEG Codec SW Reset	n/a						JPEG Operation Start

このレジスタはリードしないでください。また JPEG 動作中にライトしないでください。（リセットは可能）

Bit 7 : **JPEG コーデックソフトウェアリセット**

JPEG コーデックをソフトウェアリセットします。JPEG コーデックレジスタはリセットされません。

0： 影響なし

1： リセット

Bit 0 : **JPEG 動作スタート**

JPEG 動作（YUV データキャプチャ含む）を開始します。

0： 影響なし

1： JPEG 動作開始

JPEG 動作ステータスレジスタ

JCODEC[0x08]　初期値 = 0x0000　Read Only

15	14	13	12	11	10	9	8
n/a							

7	6	5	4	3	2	1	0
n/a							JPEG Operation Status (RO)

このレジスタをリードすることで JPEG ステータスフラグレジスタ bit 1 / JPEG ロウステータスフラグレジスタ bit 1 がクリアされます。

Bit 0 : **JPEG 動作ステータス**

JPEG コーデックの動作状態を示します。

0： 停止中

1： JPEG エンコード

量子化テーブル番号レジスタ

JCODEC[0x0C]　初期値 = 0x0000　Read/Write

15	14	13	12	11	10	9	8
n/a							

7	6	5	4	3	2	1	0
n/a					V Table Select	U Table Select	Y Table Select

Bit 2 : **V 成分テーブル選択**

JPEG エンコード時の V 成分の量子化テーブル番号を設定します。

0： 量子化テーブル 0 使用（初期値）

1： 量子化テーブル 1 使用

Bit 1 : **U 成分テーブル選択**
JPEG エンコード時の U 成分の量子化テーブル番号を設定します。
0： 量子化テーブル 0 を使用（初期値）
1： 量子化テーブル 1 を使用

Bit 0 : **Y 成分テーブル選択**
JPEG エンコード時の Y 成分の量子化テーブル番号を設定します。
0： 量子化テーブル 0 を使用（初期値）
1： 量子化テーブル 1 を使用

ハフマンテーブル番号レジスタ
JCODEC[0x10] 初期値 = 0x0000 Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a		V ACTable Select	V DCTable Select	U ACTable Select	U DCTable Select	Y ACTable Select	Y DCTable Select
7	6	5	4	3	2	1	0

Bit 5 : **V 成分 AC テーブル選択**
JPEG エンコード時の V 成分の AC 成分用ハフマンテーブル番号を設定します。JPEG 高速エンコードモード時は“1”を設定してください。
0： AC ハフマンテーブル 0 使用（初期値）
1： AC ハフマンテーブル 1 使用

Bit 4 : **V 成分 DC テーブル選択**
JPEG エンコード時の V 成分の DC 成分用ハフマンテーブル番号を設定します。JPEG 高速エンコードモード時は“1”を設定してください。
0： DC ハフマンテーブル 0 使用（初期値）
1： DC ハフマンテーブル 1 使用

Bit 3 : **U 成分 AC テーブル選択**
JPEG エンコード時の U 成分の AC 成分用ハフマンテーブル番号を設定します。JPEG 高速エンコードモード時は“1”を設定してください。
0： AC ハフマンテーブル 0 使用（初期値）
1： AC ハフマンテーブル 1 使用

Bit 2 : **U 成分 DC テーブル選択**
JPEG エンコード時の U 成分の DC 成分用ハフマンテーブル番号を設定します。JPEG 高速エンコードモード時は“1”を設定してください。
0： DC ハフマンテーブル 0 使用（初期値）
1： DC ハフマンテーブル 1 使用

Bit 1 : **Y 成分 AC テーブル選択**
JPEG エンコード時の Y 成分の AC 成分用ハフマンテーブル番号を設定します。JPEG 高速エンコードモード時は“0”を設定してください。
0： AC ハフマンテーブル 0 使用（初期値）
1： AC ハフマンテーブル 1 使用

Bit 0 : **Y 成分 DC テーブル選択**
JPEG エンコード時の Y 成分の DC 成分用ハフマンテーブル番号を設定します。JPEG 高速エンコードモード時は“0”を設定してください。
0： DC ハフマンテーブル 0 使用（初期値）
1： DC ハフマンテーブル 1 使用

DRI 設定レジスタ 0

JCODEC[0x14]　　初期値 = 0x0000　　Read/Write

n/a							
15	14	13	12	11	10	9	8
DRI Value bits [15:8]							
7	6	5	4	3	2	1	0

DRI 設定レジスタ 1

JCODEC[0x18]　　初期値 = 0x0000　　Read/Write

n/a							
15	14	13	12	11	10	9	8
DRI Value bits [7:0]							
7	6	5	4	3	2	1	0

DRI 設定レジスタ 0 bits [7:0]
DRI 設定レジスタ 1 bits [7:0]：

DRI 値 bits [15:0]

JPEG エンコード時の RST マーカを挿入する MCU 数を設定します。
（設定値が“0”の場合、RST マーカは挿入されませんが、RST インターバル定義マーカはインターバル定義値として“0”が挿入されます。）

垂直ピクセルサイズレジスタ 0

JCODEC[0x1C]　初期値 = 0x0000　Read/Write

n/a							
15	14	13	12	11	10	9	8
Y Pixel Size bits [15:8]							
7	6	5	4	3	2	1	0

垂直ピクセルサイズレジスタ 1

JCODEC[0x20]　初期値 = 0x0000　Read/Write

n/a							
15	14	13	12	11	10	9	8
Y Pixel Size bits [7:0]							
7	6	5	4	3	2	1	0

垂直ピクセルサイズレジスタ 0　bits [7:0]
垂直ピクセルサイズレジスタ 1　bits [7:0]：

Y ピクセルサイズ bits[15:0]
JPEG エンコード時および YUV データキャプチャ時の垂直方向の画像サイズを設定します。

注意：YUV データキャプチャモード設定時（JCTL[0x00] bit 3-1 = 011 もしくは 111）のときは、ライトオンリとなります。そのときにリードした値は不定になります。

水平ピクセルサイズレジスタ 0

JCODEC[0x24]　初期値 = 0x0000　Read/Write

n/a							
15	14	13	12	11	10	9	8
X Pixel Size bits [15:8]							
7	6	5	4	3	2	1	0

水平ピクセルサイズレジスタ 1

JCODEC[0x28]　初期値 = 0x0000　Read/Write

n/a							
15	14	13	12	11	10	9	8
X Pixel Size bits [7:0]							
7	6	5	4	3	2	1	0

水平ピクセルサイズレジスタ 0　bits [7:0]
水平ピクセルサイズレジスタ 1　bits [7:0]：

X ピクセルサイズ bits[15:0]
JPEG エンコード時および YUV データキャプチャ時の水平方向の画像サイズを設定します。

注意：YUV データキャプチャモード設定時（JCTL[0x00] bit 3-1 = 011 もしくは 111）のときは、ライトオンリとなります。そのときにリードした値は不定になります。

予約レジスタ
JCODEC[0x2C-34]

15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							

RST マーカ動作設定レジスタ
JCODEC[0x38]　初期値 = 0x0000　Read/Write

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
n/a						RST Marker Operation Select bits [1:0]	

Bits [1:0]：　**RST マーカ動作選択**
Reserved

RST マーカ動作ステータスレジスタ
JCODEC[0x3C]　初期値 = 0x0000　Read Only

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
Revise Code	JPEG Error Status bits [3:0]				n/a		

Bit 7：　**補正コード**
Reserved

Bits [6:3]：　**JPEG デコードエラーステータス**
Reserved

挿入マーカデータレジスタ
JCODEC[0x40-0xCC]　初期値 = 0x00FF　Read/Write

15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
Insert marker Data bits [7:0]							

JPEG エンコード時のマーカ挿入するレジスタ（36 バイト）です。マーカ長に関係なく 36 バイトすべてが挿入されます。

Address Offset [40h-44h]：　挿入するマーカコードを設定します。

Address Offset [48h-4Ch]：　マーカ長を設定します。設定値は 0002h ～ 0022h です。

Address Offset [50h-CCh]：　マーカデータを設定します。最大 32 バイトです。マーカ長を超えるデータ部分は必ず FFh で埋めてください。

量子化テーブル No.0 レジスタ

JCODEC[0x400-0x4FC] 初期値 = not applicable Read/Write

n/a							
15	14	13	12	11	10	9	8
Quantization Table No. 0 bits [7:0]							
7	6	5	4	3	2	1	0

量子化テーブル No.0

JPEG エンコード時のテーブル番号 0 の量子化テーブル値を設定します。

量子化テーブル No.1 レジスタ

JCODEC[0x500-0x5FC] 初期値 = not applicable Read/Write

n/a							
15	14	13	12	11	10	9	8
Quantization Table No. 1 bits [7:0]							
7	6	5	4	3	2	1	0

量子化テーブル No.1

JPEG エンコード時のテーブル番号 1 の量子化テーブル値を設定します。

DC ハフマンテーブル No.0 レジスタ 0

JCODEC[0x800-0x83C] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
DC Huffman Table No. 0 Register 0 bits [7:0]							
7	6	5	4	3	2	1	0

DC ハフマンテーブル No.0

JPEG エンコード時のテーブル番号 0 の DC 成分用ハフマンテーブル値です。符号長毎の符号数を設定します。ハフマンテーブル自動設定時は使用しません。

DC ハフマンテーブル No.0 レジスタ 1

JCODEC[0x840-0x86C] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
Reserved (must be all 0)				DC Huffman Table No. 0 Register 1 bits [3:0]			
7	6	5	4	3	2	1	0

DC ハフマンテーブル No.0

JPEG エンコード時のテーブル番号 0 の DC 成分用ハフマンテーブル値です。発生頻度順のグループ番号を設定します。設定は下位４ ビットで行い、上位４ ビットは必ず“0000” に設定してください。ハフマンテーブル自動設定時は使用しません。

AC ハフマンテーブル No.0 レジスタ 0

JCODEC[0x880-0x8BC] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
AC Huffman Table No. 0 Register 0 bits [7:0]							
7	6	5	4	3	2	1	0

AC ハフマンテーブル No.0

JPEG エンコード時のテーブル番号 0 の AC 成分用ハフマンテーブル値です。符号長毎の符号数を設定します。ハフマンテーブル自動設定時は使用しません。

AC ハフマンテーブル No.0 レジスタ 1

JCODEC[0x8C0-0xB44] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
AC Huffman Table No. 0 Register 0 bits [7:0]							
7	6	5	4	3	2	1	0

AC ハフマンテーブル No.0

JPEG エンコード時のテーブル番号 0 の AC 成分用ハフマンテーブル値です。発生頻度順のゼロラン長/ グループ番号を設定します。ハフマンテーブル自動設定時は使用しません。

DC ハフマンテーブル No.1 レジスタ 0

JCODEC[0xC00-0xC3C] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
DC Huffman Table 1 Register No. 0 bits [7:0]							
7	6	5	4	3	2	1	0

DC ハフマンテーブル No.1

JPEG エンコード時のテーブル番号 1 の DC 成分用ハフマンテーブル値です。符号長毎の符号数を設定します。ハフマンテーブル自動設定時は使用しません。

DC ハフマンテーブル No.1 レジスタ 1

JCODEC[0xC40-0xC6C] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
Reserved (must be all 0)				DC Huffman Table No. 1 Register 1 bits [3:0]			
7	6	5	4	3	2	1	0

DC ハフマンテーブル No.1

JPEG エンコード時のテーブル番号 1 の DC 成分用ハフマンテーブル値です。発生頻度順のグループ番号を設定します。設定は下位 4 ビットで行い、上位 4 ビットは必ず“0000” に設定してください。ハフマンテーブル自動設定時は使用しません。

AC ハフマンテーブル No.1 レジスタ 0

JCODEC[0xC80-0xCBC] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
AC Huffman Table No. 1 Register 0 bits [7:0]							
7	6	5	4	3	2	1	0

AC ハフマンテーブル No.1

JPEG エンコード時のテーブル番号 1 の AC 成分用ハフマンテーブル値です。符号長毎の符号数を設定します。ハフマンテーブル自動設定時は使用しません。

AC ハフマンテーブル No.1 レジスタ 1

JCODEC[0xCC0-0xF44] 初期値 = not applicable Write Only

n/a							
15	14	13	12	11	10	9	8
AC Huffman Table No. 1 Register 0 bits [7:0]							
7	6	5	4	3	2	1	0

AC ハフマンテーブル No.1

JPEG エンコード時のテーブル番号 1 の AC 成分用ハフマンテーブル値です。発生頻度順のゼロラン長／グループ番号を設定します。ハフマンテーブル自動設定時は使用しません。

8.5 動作説明

8.5.1 キャプチャ制御機能

本デバイスでは、ステートマシンによってデータキャプチャを制御しています。これはカメラ画像 JPEG エンコードや YUV キャプチャではフレーム単位でのデータキャプチャが必要であり、その制御をハードウェアに行わせることによって時間制限の緩和が期待されるためです。
ステートマシンはカメラ画像 JPEG エンコードと YUV データキャプチャでステート遷移条件およびステートの意味が異なっています。
ステートマシンの現在のステートはレジスタからリードできますが、リアルタイムに変化する信号であるため、リードする目的はソフトウェアデバッグであることが望ましいです。

8.5.1.1 カメラ画像 JPEG エンコード用ステートマシン

カメラ画像 JPEG エンコードモード時のキャプチャ制御ステートマシンは以下のようになります。

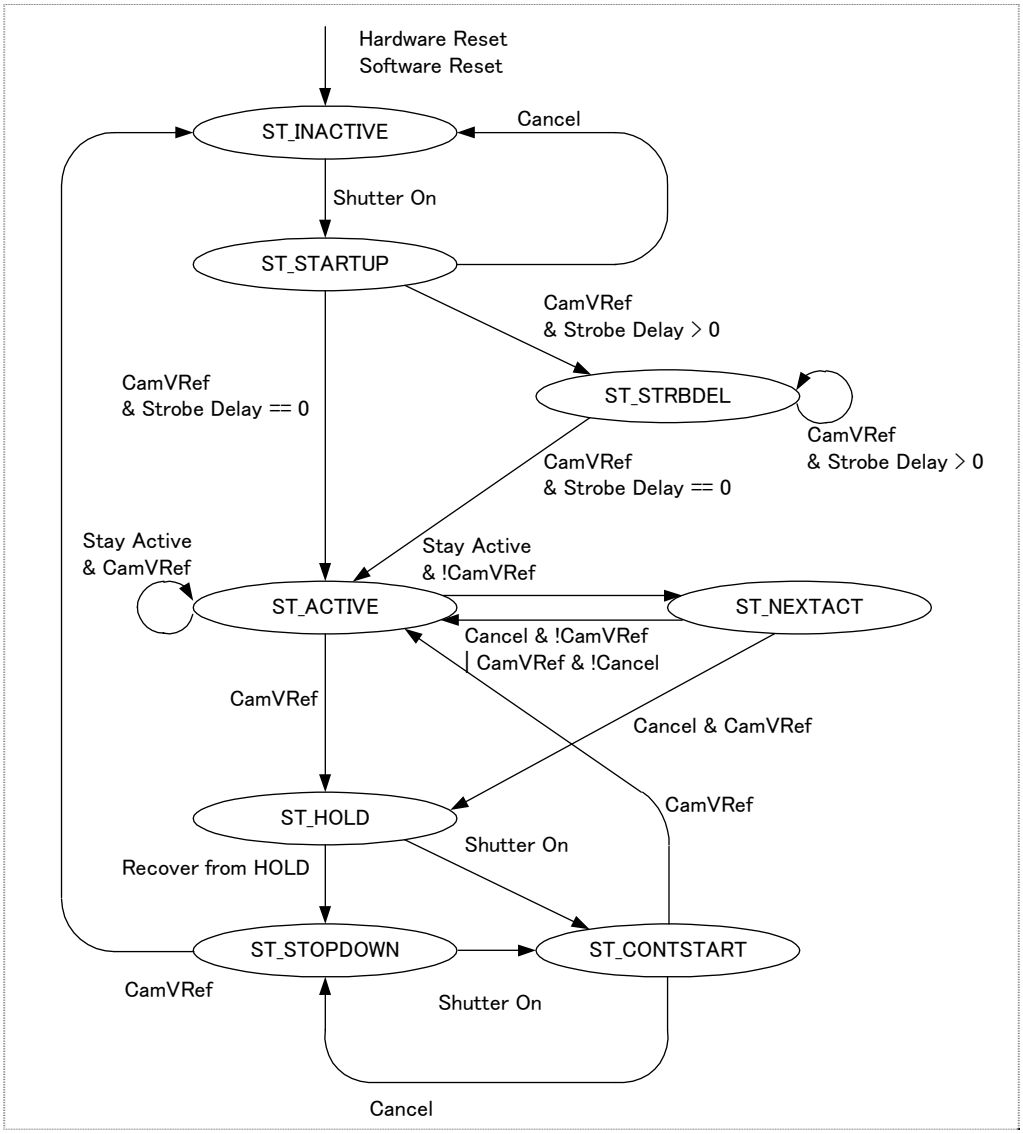


図 8.2　State Machine for Camera Image JPEG Encode

各ステートの意味は以下の表のとおりです。ステート ID はレジスタリードによって読み出される値になります。

表 8.13 State Description in Camera Image JPEG Encode Mode

ステート	説明
ST_INACTIVE ステート ID（0x0）	初期状態です。このステートはカメラデータを取り込んでいません。
ST_STARTUP ステート ID（0x1）	ST_INACTIVE から、シャッターが押された状態です。このステートはカメラデータを取り込んでいません。
ST_STRBDEL ステート ID（0x3）	ストロボフレーム遅延を待っている状態です。ストロボ遅延がある場合には ST_ACTIVE に行く前にこのステートで指定フレーム数分取り込みを遅らせます。このステートはカメラデータを取り込んでいません。
ST_ACTIVE ステート ID（0x7）	カメラデータを取り込むステートです。このステートではカメラからの VREF 以外でステート遷移を起こすことはないため、切り取り、縮小設定が正しく設定されている限り、フレームデータが取り込まれることを保証しています。
ST_NEXTACT ステート ID（0xB）	ST_ACTIVE 中に次フレームキャプチャリクエストを“1”にすることでこのステートに入ります。このステートでは、カメラからの VREF が入り次第、ST_ACTIVE に戻ります。このステートは連続フレームエンコードを実現するために用意されています。このステートはカメラデータを取り込んでいます。
ST_HOLD ステート ID（0x6）	取り込みが終了した次のフレームからこのステートに入ります。このステートはカメラデータの取り込みを行っていません。
ST_STOPDOWN ステート ID（0x4）	ST_HOLD からホールド解除信号が来た状態です。次のフレームから ST_INACTIVE に遷移し、ホールドが解除されます。このステートはカメラデータの取り込みを行っていません。
ST_CONTSTART ステート ID（0x5）	ホールド状態のときにシャッターが押されたことで次のフレームからまた取り込みを行います。このステートはカメラデータの取り込みを行っていません。

またイベントとアクションとの対応は以下の表の通りになります。

表 8.14 Event Description in Camera Image JPEG Encode Mode

イベント	アクション
Software Reset	リサイズ回路をソフトウェアリセット
Shutter On	JPEG スタートストップレジスタに“1”をライト
Cancel, Recover from HOLD	JPEG スタートストップレジスタに“0”をライト
CamVRef	カメラ画像データの VREF データアクティブレベルからデータインアクティブレベルへ変化するタイミング。本節の説明において VREF とだけ書いてあるときはこのタイミングを表しています。
Stay Active	JPEG 制御レジスタの次フレーム取り込みリクエストビットに“1”をライト
Strobe Delay	ストロボフレーム遅延カウント

カメラ画像 JPEG エンコードを一回行ったときのタイミングチャートは以下のようになります。

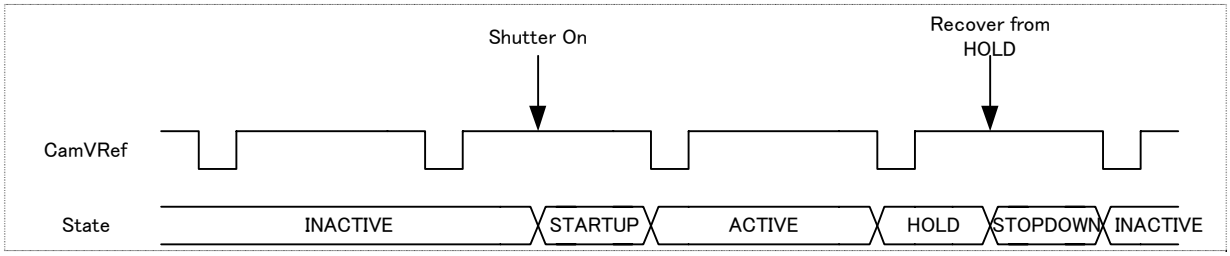


図 8.3 Timing Chart for Camera Image JPEG Encode (Single)

8.5.1.2 YUV キャプチャ用ステートマシン

YUV キャプチャモードのキャプチャ制御ステートマシンは以下のようになります。

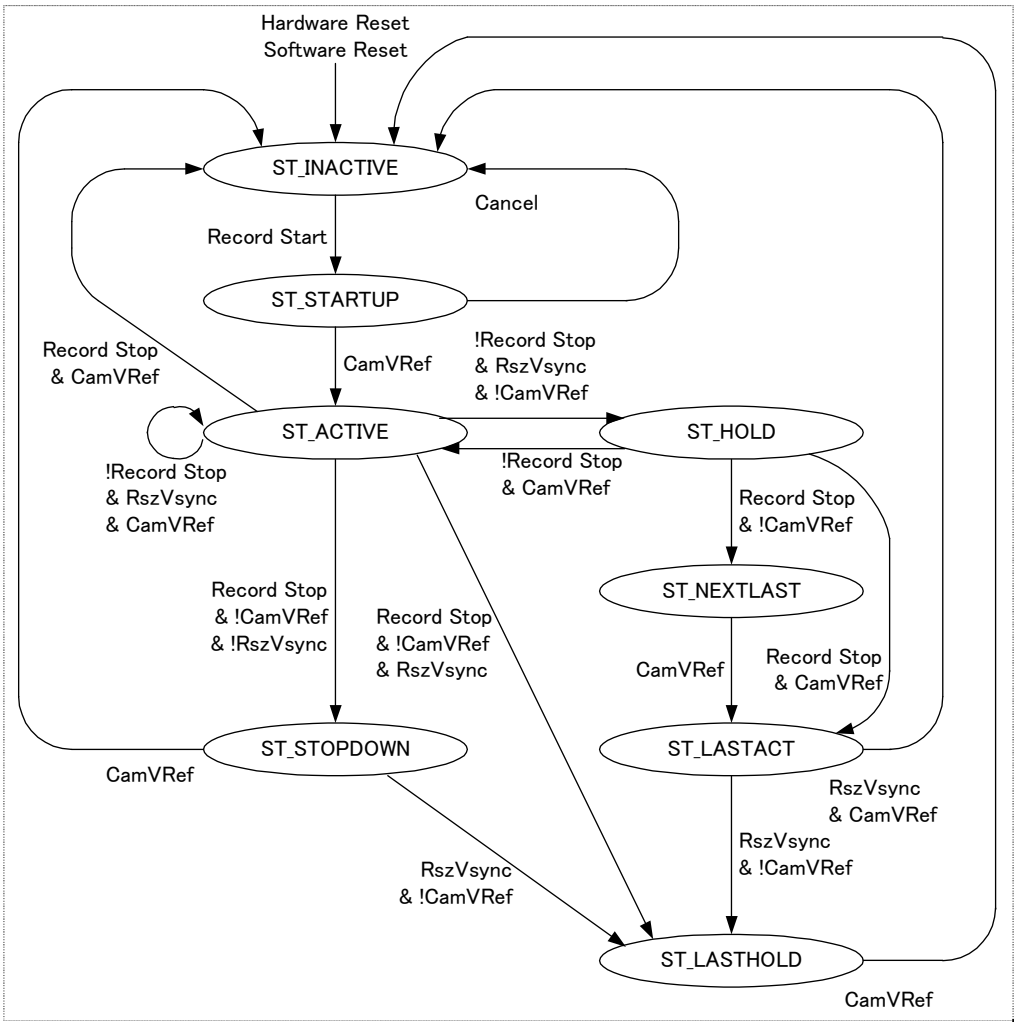


図 8.4 State Machine for YUV Capture

表 8.15　State Description in YUV Capture Mode

ステート	説明
ST_INACTIVE ステート ID（0x0）	初期状態です。このステートはカメラデータの取り込みを行っていません。
ST_STARTUP ステート ID（0x1）	ST_INACTIVE から、録画スタート信号が入った状態です。このステートはカメラデータの取り込みを行っていません。
ST_ACTIVE ステート ID（0x7）	カメラデータを取り込むステートです。このステートはカメラデータの取り込みを行っています。
ST_HOLD ステート ID（0x6）	フレーム中に取り込みデータの VREF が発生したときに、ST_ACTIVE からこのステートに移ります。このステートはカメラデータの取り込みを行っていません。
ST_STOPDOWN ステート ID（0x4）	ST_ACTIVE から、録画ストップ信号が入った状態です。現フレームで取り込みが終了します。このステートはカメラデータの取り込みを行っています。
ST_NEXTLAST ステート ID（0xE）	ST_HOLD 中に録画ストップ信号が入るとこのステートに入ります。このステートは取り込みデータの VREF 後からカメラ VREF までの間は JPEG モジュールが動画取り込みの終わりを認識できないという制限のために、取り込みデータの VREF 後の録画ストップ信号に対してもう“1”フレーム取り込ませるためのステートです。このステートはカメラデータの取り込みを行っていません。
ST_LASTACT ステート ID（0xF）	上記の制限で説明されているもう 1 フレーム取り込むステートになります。このステートはカメラデータの取り込みを行っています。
ST_LASTHOLD ステート ID（0xD）	録画の最終フレームで取り込みデータの VREF が発生した後にこのステートに移ります。このステートはカメラデータの取り込みを行っていません。

またイベントとアクションとの対応は以下の表の通りになります。

表 8.16　Event Description in YUV Capture Mode

イベント	アクション
Software Reset	リサイズ回路をソフトウェアリセット
Record Start	JPEG スタートストップレジスタに“1”をライト
Cancel, Record Stop	JPEG スタートストップレジスタに“0”をライト
CamVRef	カメラ画像データの VREF データアクティブレベルからデータインアクティブレベルへ変化するタイミング。本節の説明において VREF とだけ書いてあるときはこのタイミングを表しています。
RszVsync	リサイズ回路の出力するフレームエンド信号

8.5.2 リサイズ機能

リサイズ回路はカメラインタフェースからくる画像データに対して切り取りと縮小を行います。リサイズ回路は切り取り処理と縮小処理の 2 つのステージを持っています。YUV4:2:2 でカメラモジュールから来た画像データはカメラインタフェースにおいて YUV4:4:4 フォーマットに変換されますので、リサイズ回路では 1x1 画素単位で切り取り処理を行うことができます。

8.5.2.1 切り取り機能

切り取り処理は縮小処理の前段階として、縮小処理に必要な画像サイズになるように画像の周辺部を切り捨てます。切り取り処理は X/Y スタート/エンド座標レジスタを使用します。元画像の左上を座標(0,0)とし、スタート座標は切り取り残す画像の左上の座標、エンド座標は残す画像の右下の座標を表します。これらの設定値は 1x1 画素単位で指定できますが、それぞれ入力カメラ画像の最大座標より大きい座標を指定することはできません。エンド座標をスタート座標より小さい値に設定した場合、リサイズ回路は正常に動作せず、何らかのデータがリサイズ回路から出力されますので、エンド座標は必ずスタート座標以上になるように設定してください。リサイズ回路からのデータ出力を強制停止させたい場合には、RSZ[0xC0]bit0 に“0”を書き、リサイズ回路をディセーブルにしてください。

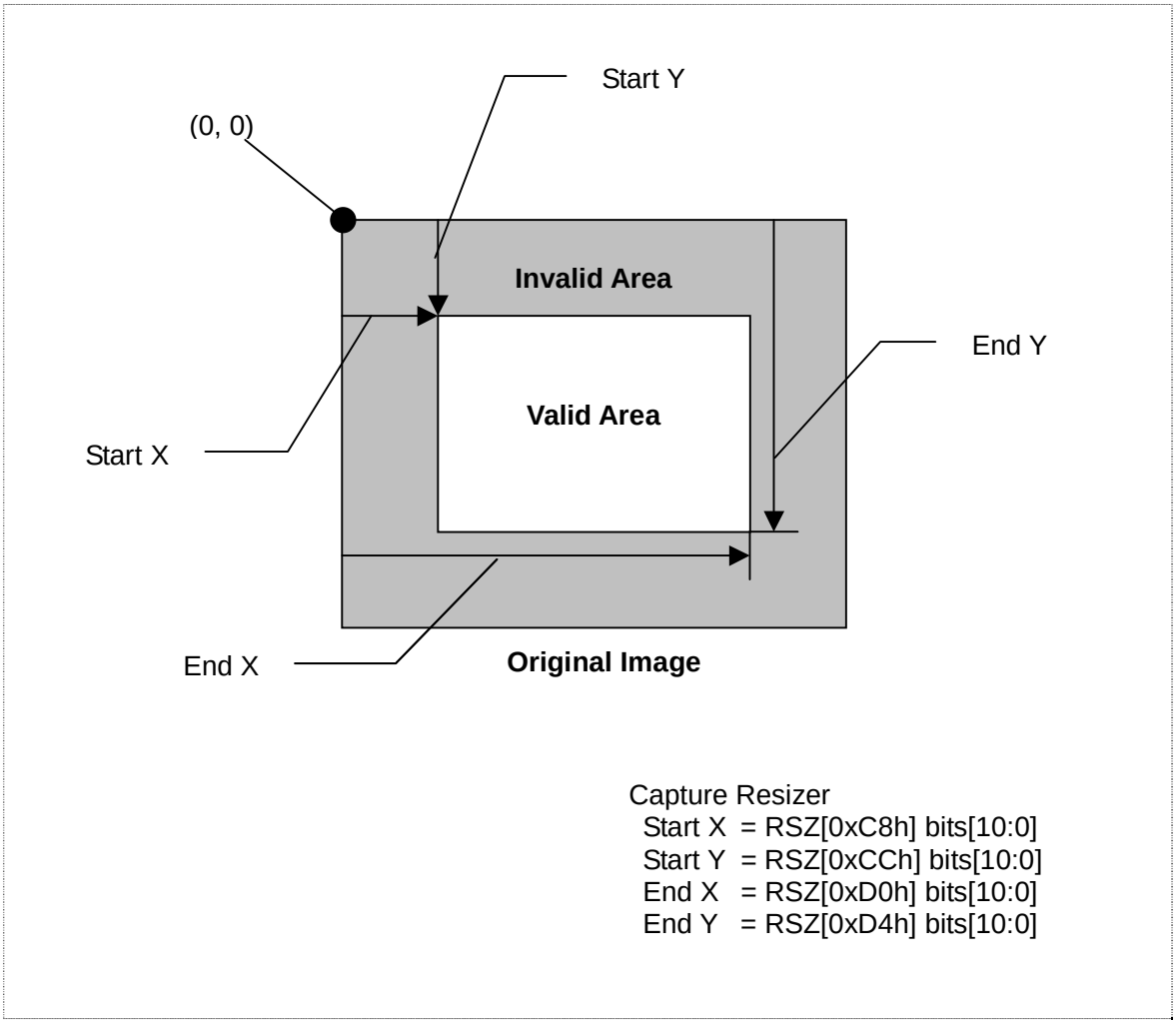


図 8.5 Trimming Function

8.5.2.2 縮小機能

縮小処理は切り取り処理の後に行われ、切り取られた画像に対して設定された縮小率で画像を縮小します。縮小後の画像サイズは、JPEG エンコードで設定する画像サイズと同じでなければなりません。

縮小処理には、間引き縮小モード、加算平均縮小モード、縮小無しモードと 3 種類の縮小モードがあります。間引き縮小モードでは画像ブロックから代表点を選び、それを縮小後画像の 1 画素とします。1/2～1/8 の任意の縮小率で設定が可能です。加算平均縮小モードでは、縦方向が間引き縮小で横方向が加算平均による縮小になります。間引き縮小選択された画像ブロック列のすべての画素を加算し、その平均値が画像ブロックの縮小後の 1 画素となります。1/2、1/4、1/8 の 3 種類の縮小率を選ぶことができます。縮小無しモードは、縮小しないときに設定します。縮小無しモードでは、縮小率設定レジスタの値は無効になります。また縮小無しモードを用意していますので、それ以外のモードで縮小率設定レジスタに“1”を書いたときの動作は保証していません。縮小しない場合は縮小無しモードを選択してください。

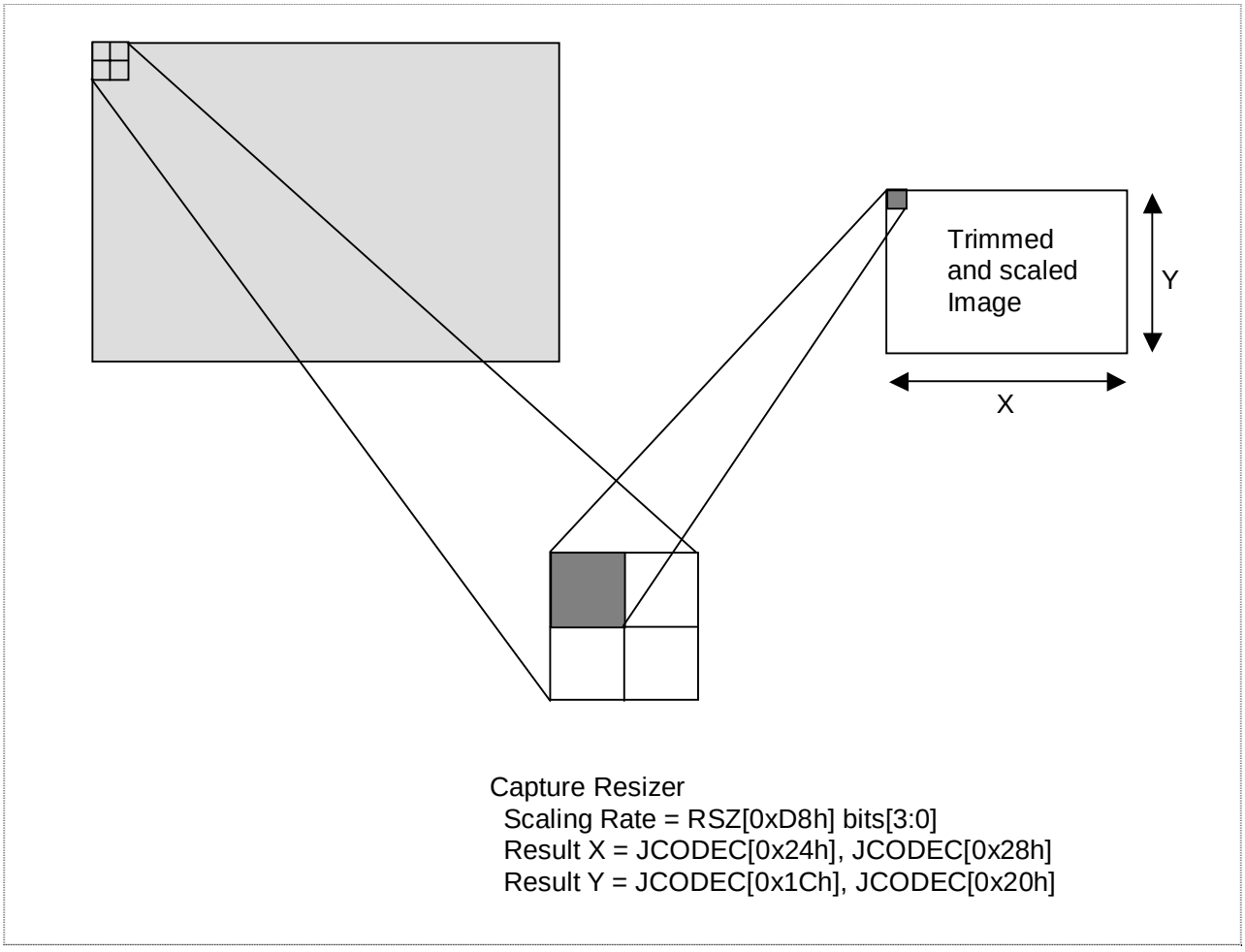


図 8.6 Scaling Example (1/2 Scaling)

1/2 縮小

1/2 縮小では、2x2 画素ブロック毎に 1 画素に縮小されます。この縮小率では間引き、加算平均の両方の縮小モードが選択できます。ただし縦方向はどちらの縮小モードでも同じく間引き縮小となります。

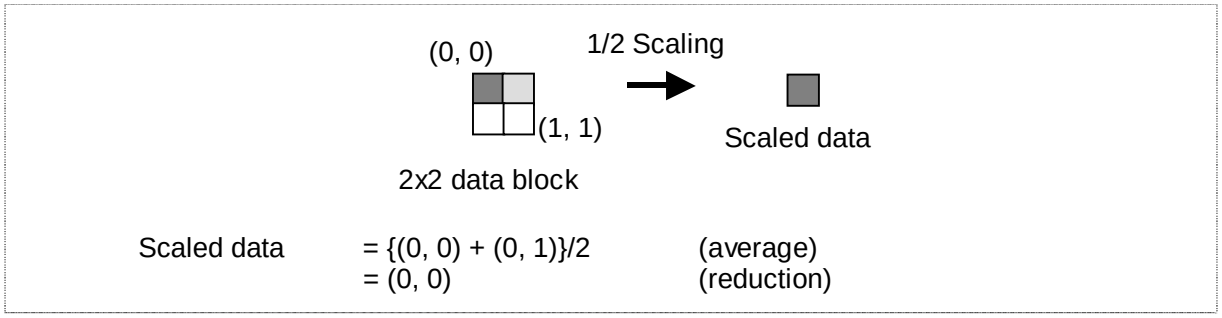


図 8.7 1/2 Compression

1/3 縮小

1/3 縮小では、3x3 画素ブロック毎に 1 画素に縮小されます。この縮小率では縮小モードは間引き縮小のみとなります。

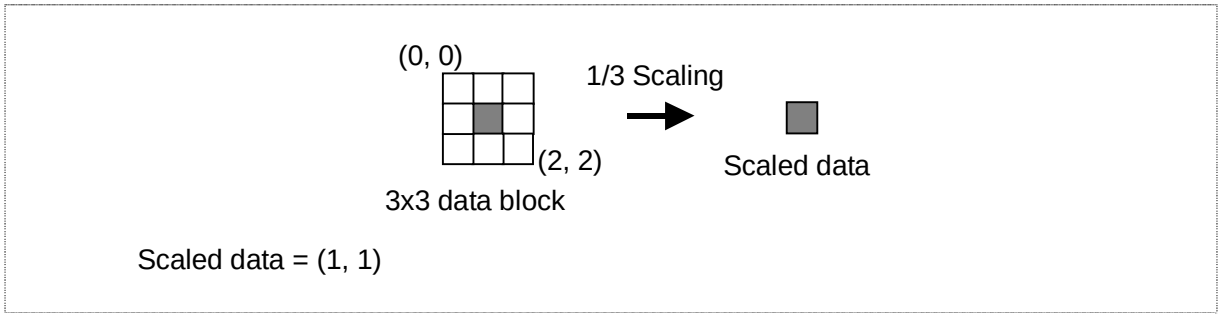


図 8.8 1/3 Scaling

1/4 縮小

1/4 縮小では、4x4 画素ブロック毎に 1 画素に縮小されます。この縮小率では間引き、加算平均の両方の縮小モードが選択できます。ただし縦方向はどちらの縮小モードでも同じく間引き縮小となります。

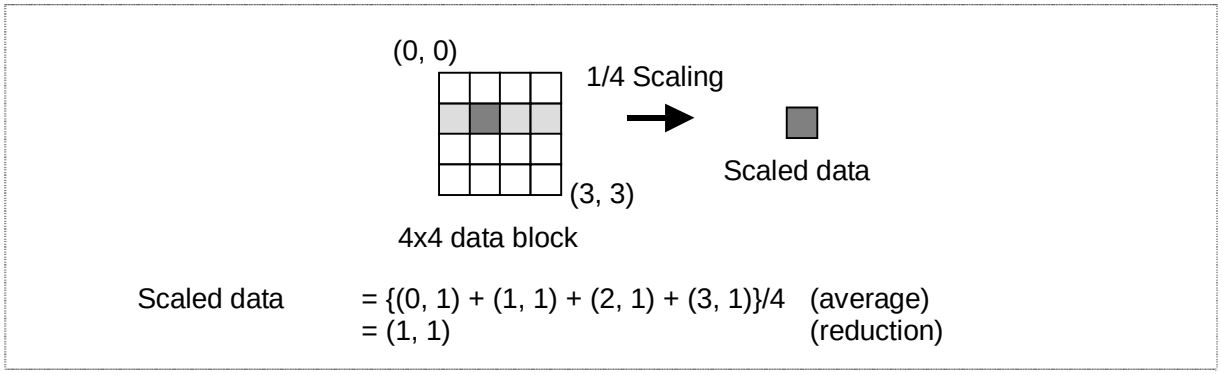


図 8.9 1/4 Scaling

1/5 縮小

1/5 縮小では、5x5 画素ブロック毎に 1 画素に縮小されます。この縮小率では縮小モードは間引き縮小のみとなります。

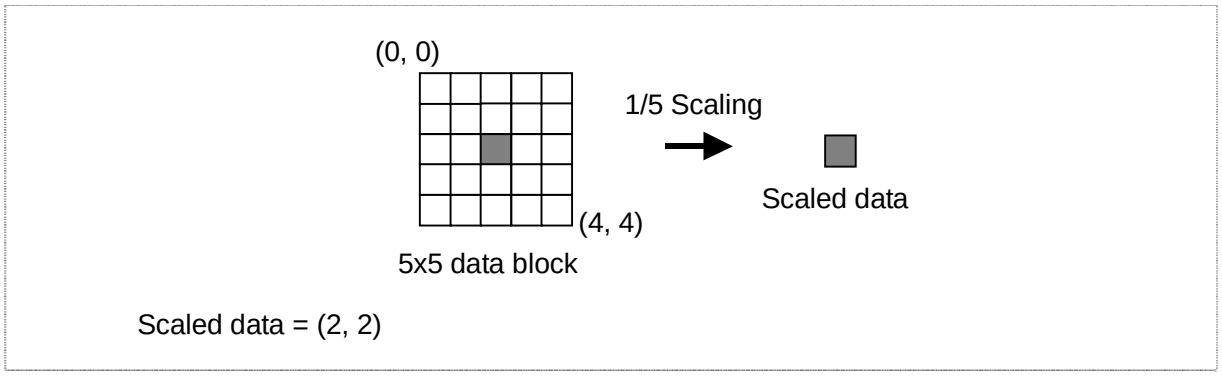


図 8.10 1/5 Scaling

1/6 縮小

1/6 縮小では、6x6 画素ブロック毎に 1 画素に縮小されます。この縮小率では縮小モードは間引き縮小のみとなります。

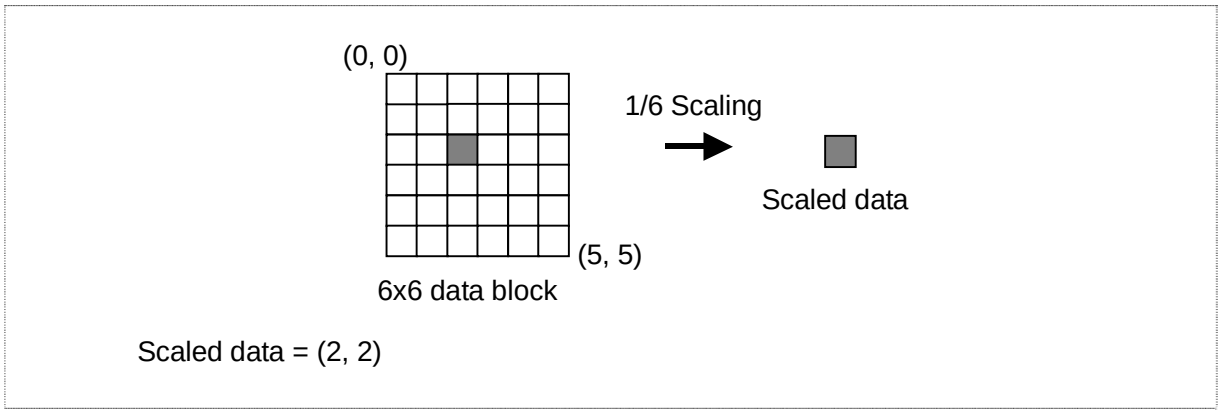


図 8.11 1/6 Scaling

1/7 縮小

1/7 縮小では、7x7 画素ブロック毎に 1 画素に縮小されます。この縮小率では縮小モードは間引き縮小のみとなります。

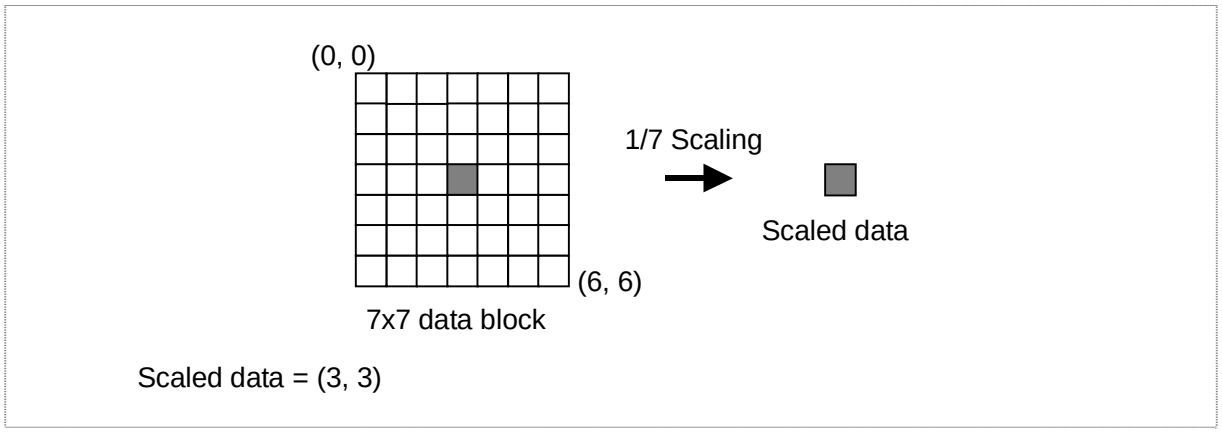


図 8.12 1/7 Scaling

1/8 縮小

1/8 縮小では、8x8 画素ブロック毎に 1 画素に縮小されます。この縮小率では間引き、加算平均の両方の縮小モードが選択できます。ただし縦方向はどちらの縮小モードでも同じく間引き縮小となります。

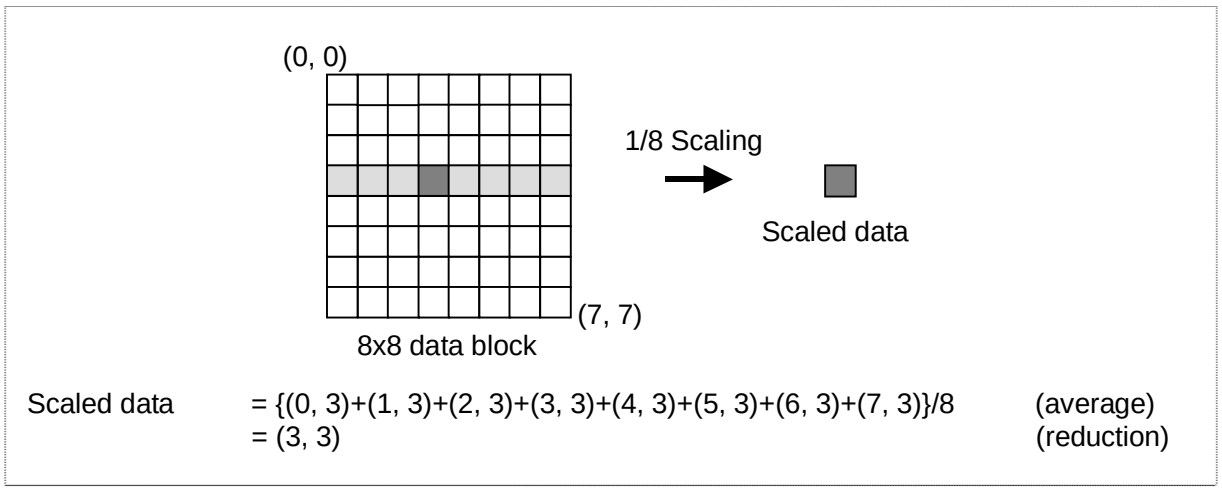


図 8.13 1/8 Scaling

8.5.2.3 リサイズ回路利用制限

リサイズ回路のレジスタは、ソフトウェアリセットやイネーブルを除き、カメラインタフェースからデータがきている間に変更できません。カメラインタフェースからデータがきているときにレジスタを変更した場合、データが破壊されます。リサイズ回路のレジスタ設定をカメラ動作中に変更する場合は、カメラインタフェースの VREF 割り込みを使用して、VREF データブランク期間（フレームの間のカメラインタフェースがイメージセンサーからデータを受け取っていない期間）中に変更する方法が一般的です。

リサイズ X/Y スタート、ストップ座標レジスタは、カメラ画像よりも大きい座標を指定することはできません。指定した場合は、正常に動作しなくなります。

リサイズ X/Y スタート、ストップ座標レジスタを指定することで切り取られる画像は、縮小率設定レジスタで設定した値で割り切れなければなりません。

8.5.3 画像処理データフロー

ここでは、各画像処理モードでのデータフローを図示します。

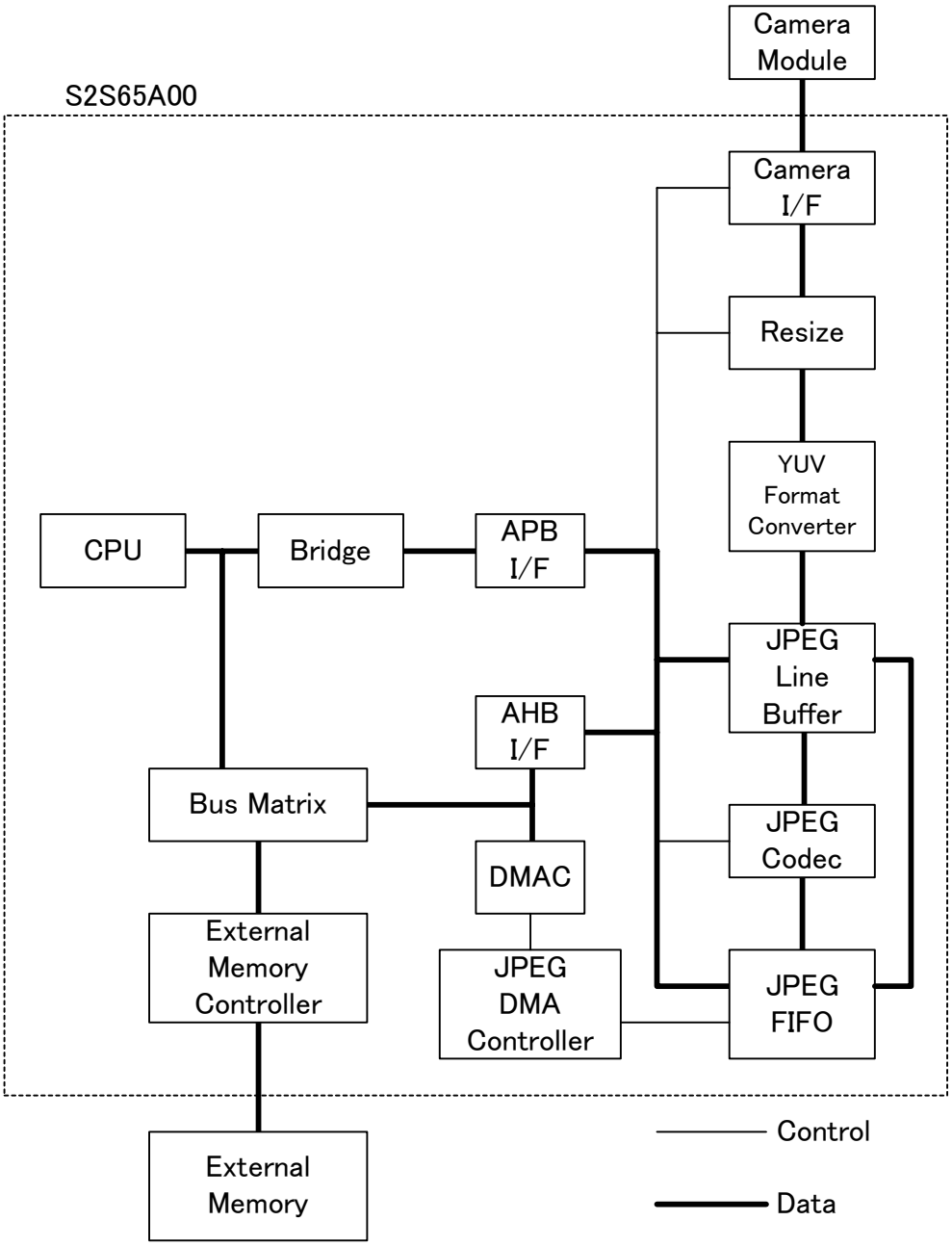


図 8.14　画像処理データフロー

8.5.3.1 カメラ画像 JPEG エンコード

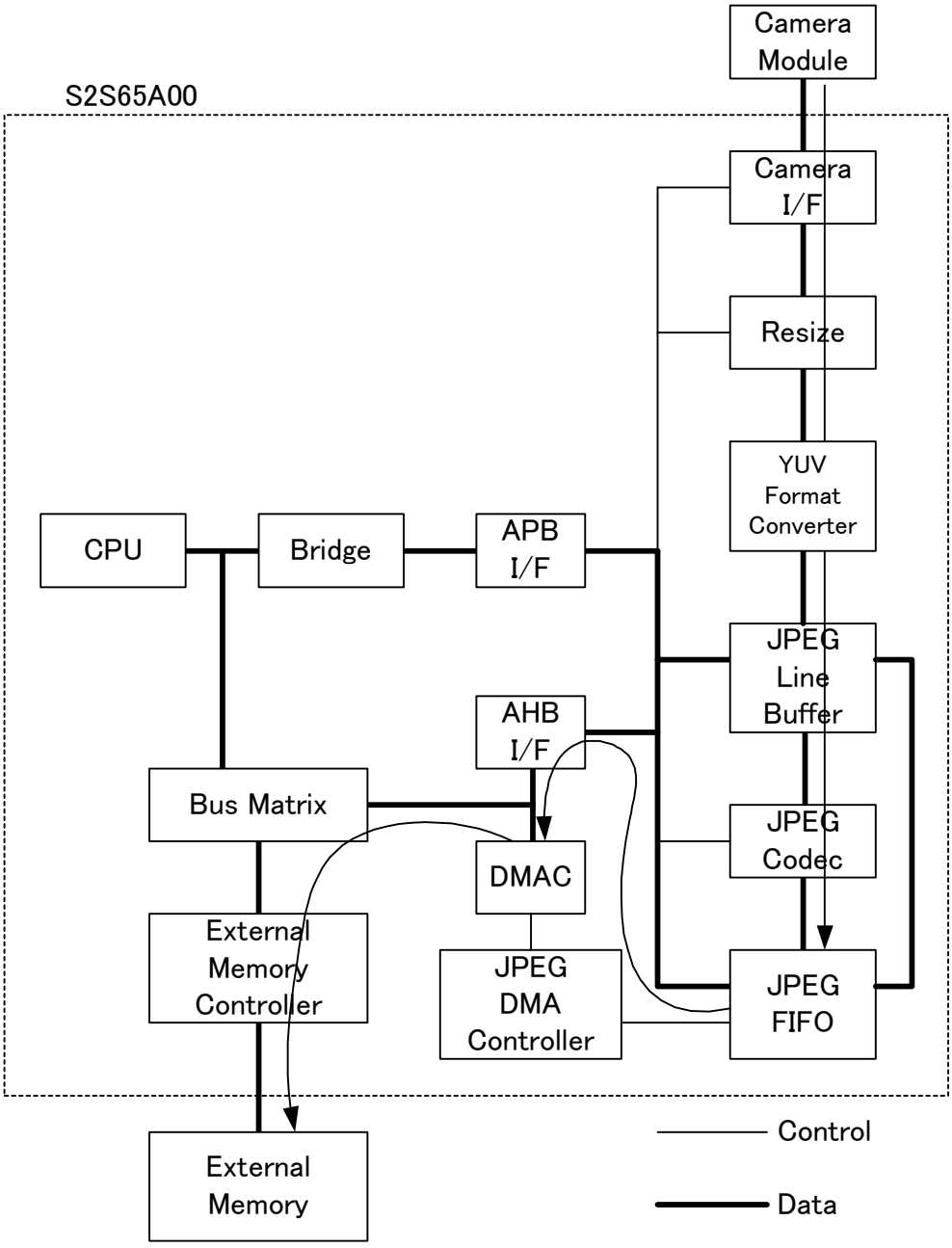


図 8.15　カメラ画像 JPEG エンコードデータフロー

8.5.3.2 YUV データキャプチャ

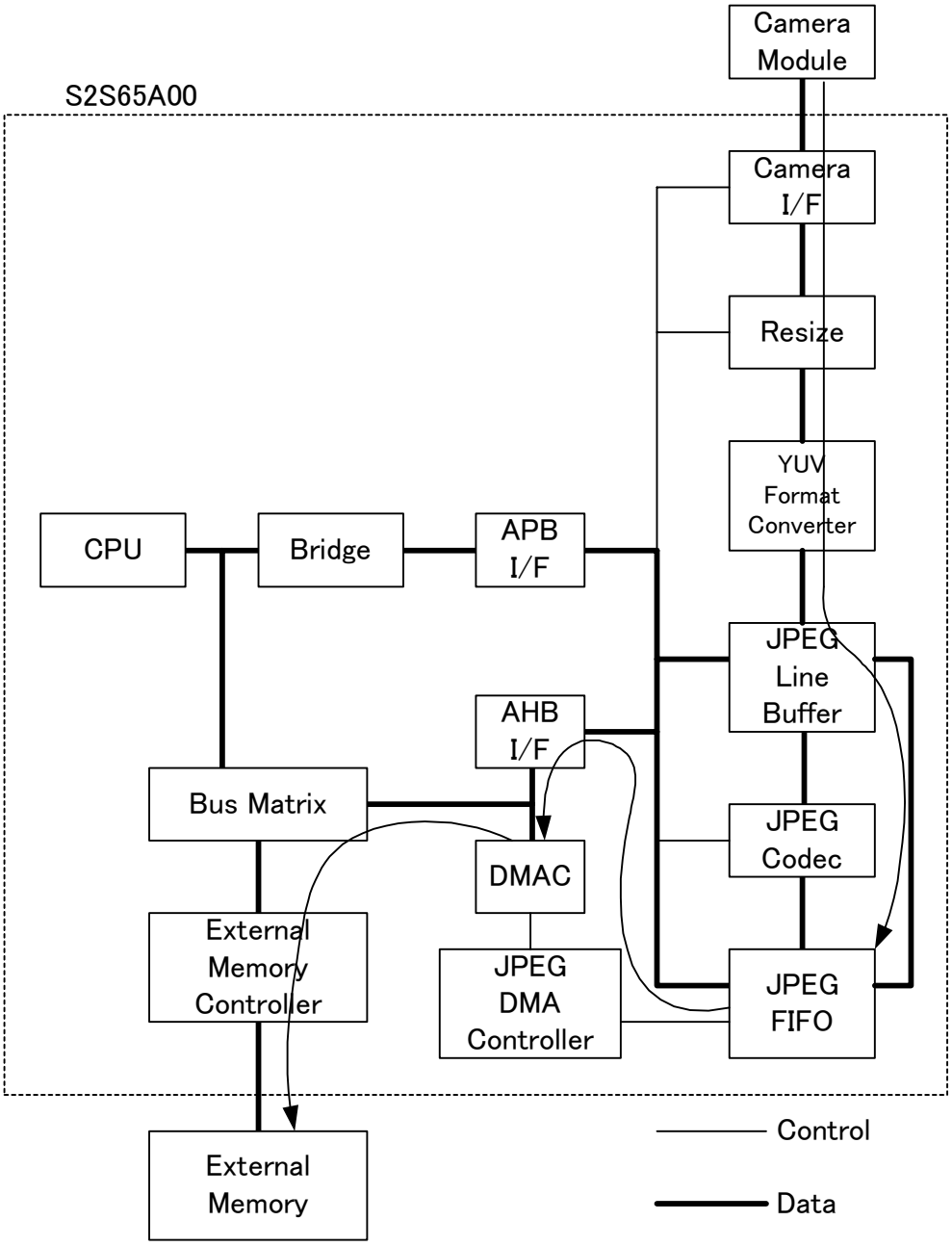


図 8.16　YUV データキャプチャデータフロー

8.5.4 JPEG コーデック機能

本デバイスに搭載されている JPEG コーデック回路は JPEG ベースライン方式にほぼ準拠しており、JPEG Part-2 （ISO/IEC10918-2）で要求される適合性試験の結果を満足しています。JPEG コーデック回路は最大横幅 640 画素、縦幅は 480 画素まで対応しています。VGA サイズまでの JPEG エンコードが可能です。

リサイズ回路から来た YUV4:4:4 フォーマットのデータは、YUV フォーマットコンバータによって JCODEC[0x00] bits[1:0] で設定された YUV フォーマットに変換されます。ただしカメラ画像 JPEG エンコード、および YUV キャプチャでは処理される画像サイズが以下の解像度でなければならないという制限があります。

表 8.17 Minimum Resolution Restrictions

YUV Format	Minimum Resolution
4:4:4	1x1
4:2:2	2x1
4:2:0	2x2
4:1:1	4x1

さらに画像の最小サイズにも制限があり、このサイズより小さいサイズの画像は動作保証されません。

表 8.18 Minimum Size

YUV Format	MCU Size (Horizontal x Vertical)
4:4:4	(8 x 8)
4:2:2	16 x 8
4:2:0	16 x 16
4:1:1	32 x 8

量子化テーブルはエンコード用に 2 つ持つことができます。ハフマンテーブルは AC、DC それぞれ 2 つづつ持つことができます。また JPEG エンコードでは、マーカ識別子を含め最大 36 バイトのマーカを挿入することができます。カメラ画像の JPEG エンコード処理は YUV4:4:4、YUV4:2:2、YUV4:2:0、および YUV4:1:1 をサポートしています。ただし YUV4:4:4 はカメラ入力画像よりデータ量が大きくなりますので、カメラ画像を縮小無しでエンコードする場合はサポート対象外となります。RAM 上の YUV データから JPEG データへのエンコードは、YUV4:2:2 および YUV4:2:0 のみサポートします。画像データ処理能力は 640x480 画像サイズで 1/30 秒以下となりますが、量子化テーブル、ハフマンテーブルの値、カメラ入力画像の細かさなどの要因によりこの処理能力を満たせない場合があります。

JPEG を連続してエンコードする場合、安定して動作する方式は 1 フレームエンコードしたら 1 フレームエンコードしない処理になります。毎フレーム連続エンコード機能は持っていますが、実現できるかどうかはシステムで採用されるカメラモジュールの種類、ソフトウェア処理速度に依存します。具体的には以下の不等式が成り立つ必要があります。

カメラの VREF インアクティブ期間
> 割り込み応答時間
＋ プライオリティの高いタスクによって処理が中断される合計時間
＋ 次フレームエンコードのための設定処理時間
＋ JPEG コーデック回路のマーカ出力時間

JPEG コーデック回路のマーカ出力時間は、S2S65A00 で追加されたマーカ高速出力モードを使用するとシステムクロック 50MHz 時で 36μs、使用しない場合はおよそ 2ms になります。マーカ高速出力モードを使用しないで VGA サイズの毎フレーム連続エンコードを行う場合は、VREF のインアクティブ期間が少なくとも 15 ライン以上無いとオーバーフローが発生する可能性が極めて高くなります。

8.5.4.1 JPEG コーデックレジスタに関する制限

JPEG コーデックレジスタは JPEG コーデック動作中（JPEG コーデックスタートから JPEG コーデック終了まで）は変更禁止です。該当期間中のアクセスに対して応答があった場合、誤動作を引き起こすか、もしくは無効になります。

また、JPEG コーデックレジスタは JPEG コーデック回路にクロックが供給されていないときにアクセスしても無効になります。JPEG コーデック回路のクロックが停止する条件は以下の通りです。

- JPEG モジュールがディセーブルになっているとき
- JPEG モジュールがイネーブルで JPEG 処理モードが 000 または 100 ではないとき

ただし YUV キャプチャで設定が必要となるレジスタに限っては JPEG コーデック回路にクロックが供給されていなくても設定可能なようになっています。

JPEG コーデック回路で予約となっているレジスタや書き込みのみのレジスタにリードアクセスした場合、リード値を保証しないだけでなく JPEG コーデック回路そのものの動作を保証しないという制限があります。

また JPEG コーデックステータスレジスタ及び JPEG コーデックマーカステータスレジスタはリード自体で内部状態が変化するため、必要なときにのみリードするように注意してください。**また JPEG エンコード処理が終わったときには JPEG コーデックステータスレジスタをリードする前に JPEG コーデックマーカステータスレジスタをリードしてください。このリードを行うことによって JPEG コーデック回路は処理が完了します。リードを行わなかった場合、次の処理が正常に動作しなくなります。**

量子化テーブルおよびハフマンテーブルは、連続して同じ処理を行うときのみテーブル書き込みを省略できます。エンコード処理でも設定を変更した場合は、テーブルは書き直しになります。

8.5.5 JPEG コーデック以外の機能

8.5.5.1 JPEG FIFO

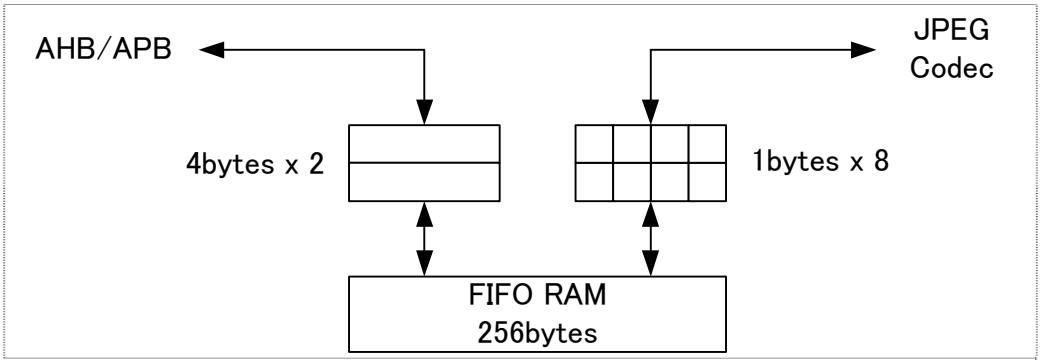


図 8.17　JPEG FIFO Overview

JPEG FIFO は 256 バイトの RAM、4 バイト x2 のバッファが RAM のリードライトで 2 つの計 272 バイトのデータを保持することができます。JPEG FIFO のステータスを見て FIFO リードライトを CPU で行う場合は、256 バイトの RAM をリードライトできる最大値とすることが望ましいです。

JPEG FIFO のステータスは JPEG FIFO ステータスレジスタでチェックすることができます。さらにそのステータスの大部分が JPEG 割り込みステータスレジスタでもチェックできます。JPEG FIFO ステータスレジスタはリードした時点のステータスを表しており、JPEG 割り込みステータスレジスタの Empty と Full ステータスは、一度でもその状態になったことを表しています。JPEG 割り込み制御レジスタではこの一度でもその状態になったという情報から割り込みを発生させていますので、割り込みがクリアされるまでは、情報を保持しつづけています。

カメラ画像 JPEG エンコード、YUV データ JPEG エンコードにおいて JPEG FIFO からデータを取り出す方法は 2 つあります。1 つは CPU が JPEG FIFO リードライトポートレジスタをリードすることです。もう 1 つは JPEG DMA コントローラを使用することです。

1. ローパフォーマンス – JPEG FIFO エンプティフラグでデータがあることを確認して、CPU が JPEG FIFO リードライトポートレジスタをリードする方法です。FIFO サイズが小さいため、FIFO フルやしきい値での割り込みは効率が悪く使用できません。CPU で JPEG コーデック終了割り込みフラグが立つまで、JPEG FIFO エンプティフラグをポーリングしてデータを読み出すため、CPU 占有率が高く、マルチタスクやリアルタイム処理に向きません。

2. ハイパフォーマンス – JPEG DMA コントローラによってデータをメモリ領域に転送する方法です。JPEG DMA コントローラは JPEG FIFO からフレーム終了の情報をもらって自動的に転送を終了させるため、DMA 設定においてメモリ領域に確保したメモリサイズ回路を最大転送量に設定するだけで、必要な転送は実現できます。1 フレームにつき少しのレジスタ設定だけでよく、FIFO からの転送にかかる CPU 負荷はほとんどありません。

8.5.5.2 JPEG ラインバッファ

JPEG ラインバッファは、30KB の RAM を持ち、ラインベース方式データとブロックインターリーブ方式データへの相互変換を行っています。処理できるデータサイズは横幅 640 画素までです。JPEG ファイルを作成するときには MCU 単位サイズの画像データが必要であるため、足りないデータの補完を行っています。たとえば 100x100 画像を YUV4:2:2 でエンコードする場合、112x104 に補完して JPEG コーデック回路にデータを送出します。この機能は MCU 単位サイズとして足りない部分だけを補完しますので、たとえばリサイズ回路から 60x60 でしかきていないデータを 112x104 などに大きく補完するわけではありません。

JPEG ラインバッファは、自分自身がオーバーフローしたときを検出し、割り込みを発生させることができます。JPEG エンコードにおいては、JPEG FIFO や JPEG コーデック回路は処理が続けられない状態になったときに入力データを待たせる機能がついているため、リサイズ回路から止められないデータが来たときに JPEG コーデック回路が入力を頻繁に待たせていると JPEG ラインバッファがオーバーフローを起こします。したがって JPEG FIFO リードや JPEG エンコード処理負荷などの理由

によりリアルタイムでエンコードできない場合は、すべてラインバッファオーバーフロー割り込みが検出されます。

ラインバッファはラインベース方式データとブロックインターリーブ方式データへの相互変換のため MCU 単位横幅のライン数分のデータを 1 バンクという単位として扱っており、30KB RAM を有効利用するように入力画像サイズの横幅によって RAM に貯めておけるバンク数が変化するようになっています。

YUV フォーマットが 4:2:0 の場合 16 ラインで 1 バンク、4:2:2 の場合 8 ラインで 1 バンクとなります。

表 8.19 入力画像サイズによる RAM データ量（バンク数）

入力画像横サイズ	データ量
≦32	入力画像横サイズ x 32banks
≦64	入力画像横サイズ x 16banks
≦128	入力画像横サイズ x 8banks
≦256	入力画像横サイズ x 4banks
>256	入力画像横サイズ x 2banks

8.5.5.3 YUV フォーマットコンバータ

YUV フォーマットコンバータは、リサイズ回路から来た YUV4:4:4 フォーマットのカメラ画像を 4 種類の YUV フォーマットに変換します。YUV フォーマット変換は下図のようにすべて加算平均を使用しています。下図では U 成分を例にしていますが、V 成分の加算平均式は同じになります。Y 成分は YUV フォーマットでも元画像と同じになります。

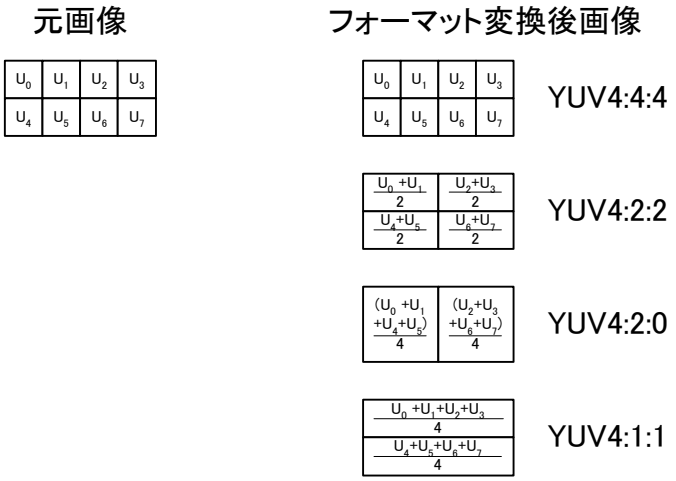


図 8.18 YUV フォーマット変換

8.5.5.4 JPEG モジュール割り込み

ここでは割り込みフラグがどのようなときに使用されるのかを説明しています。

1. JPEG コーデック割り込みフラグ
   カメラ画像 JPEG エンコードでは、CPU による FIFO アクセスを行うときに使用します。このフラグが立った段階で JPEG ファイルサイズが確定しますので、すでにリードしたデータ量をエンコード結果サイズから引いた値が最後にリードすべきデータ残数となります。YUV キャプチャではこのフラグが立つことはありません。

2. JPEG ラインバッファオーバーフロー割り込みフラグ
   カメラ画像 JPEG エンコードのときに使用します。JPEG コーデック回路は JPEG エンコードがリアルタイムで常に成功することを保証していないため、カメラ画像 JPEG エンコードを行うときには、この割り込みをイネーブルにすることが必須となります。

3. JPEG FIFO エンプティフラグ
   JPEG エンコードでは FIFO のデータがなくなったことを確認するという使用方法がありますが、念のためにという意味合いが強く、このフラグを見てエンコード処理を行う必要性はあまりありません。

4. JPEG FIFO フルフラグ
   カメラ画像 JPEG エンコードで CPU による JPEG FIFO リードを行うときに使用することがあります。その場合フルになったら FIFO のサイズ分だけリードするという使用方法になります。ただしフル中は JPEG コーデック回路のエンコード処理が中断しており、ラインバッファにデータが溜まりつづける状態になるため、実際には JPEG FIFO しきい値ステータスレジスタで 1/2 以上や 1/4 以上を見て、FIFO 有効サイズレジスタの値分をリードする方法が採用されることになります。

5. JPEG FIFO しきい値トリガフラグ
   しきい値をトリガとした割り込みによる FIFO アクセスは FIFO サイズが小さいため、割り込み応答時間のオーバーヘッドが大きくなり、実質的に使用できません。したがってこのフラグの使い道は特にありません。

6. エンコードサイズリミットオーバーフラグ
   カメラ画像 JPEG エンコードで使用します。JPEG DMA コントローラを使用する場合には、DMA 転送の最大転送数による割り込みを使用する方がメモリ領域の破壊が発生しませんので、JPEG ファイルの最大サイズを制限する直接的な方法としては使用しません。かわりにもう少しで超えそうであるという状態を検出することに使用し、たとえば量子化テーブルを大きい値に変更するなどの処理を行うことができます。

8.5.5.5 JPEG180°回転エンコード

JPEG180°回転エンコードは、1 フレームを回転させるのではなく、フレームの MCU 単位幅ライン数分のライン単位で回転します。したがってソフトウェアがライン単位で回転させたデータの並べ替えを行う必要があります。たとえば最初のライン単位のデータは JPEG ファイルの画像データ領域の最初に書かれていますが、1 フレーム回転では画像データ領域の最後になりますので、ソフトウェアで最後に移動させる必要があります。

図 8.19　JPEG180°回転エンコード

ソフトウェアがエンコードされた画像データだけからライン単位の終わりを認識することは困難ですので、RST マーカをライン単位が終わるごとに挿入するようにしてライン単位の終わりを認識させます。したがって RST マーカ挿入は必須です。RST マーカの挿入間隔は画像の横幅を MCU 単位横サイズで割った値の小数点以下繰り上げとなります。たとえば横幅 100 画素の画像を YUV4:2:0 フォーマットで JEPG180° 回転エンコードしたときに挿入される RST マーカの挿入間隔は、100/16=6.25 → 7 となります。

8.5.5.6 ソフトウェアリセット処理について

JPEG コーデック回路は 1 フレームがエンコードおける基本単位になっており、連続エンコードはそれら単発の処理を繰り返し行うことで実現します。JPEG コーデック回路が処理を行う前には、ソフトウェアリセットを行うことを推奨します。これは JPEG コーデック回路が初期状態に戻ることで安定した動作を期待できるためです。ソフトウェアリセットは機能ブロックのみをリセットするため、レジスタ設定した値はそのまま残っています。

JPEG コーデック回路のソフトウェアリセット以外に YUV フォーマットコンバータ、JPEG ラインバッファ、JPEG FIFO のコンポーネントをリセットする JPEG モジュールソフトウェアリセットがあり、カメラ画像を JPEG エンコード処理する場合に、推奨するソフトウェアリセット処理が異なります。これは、各コンポーネントでの処理時間によるタイムラグがあるためです。

毎フレーム連続エンコードする場合は、JPEG モジュールソフトウェアリセットを実行しないことを推奨します。現フレームの処理の終わりと次フレームの処理の始まりが平行して動作し、ソフトウェアリセットを実行するタイミングが存在しない可能性が高いためです。

一方で 1 フレームエンコードしたら 1 フレーム以上のインターバルを設ける場合には、現フレームのエンコード処理と次フレームのエンコード処理が重なることはありえませんので、JPEG エンコードを開始する前にソフトウェアリセットを実行することを推奨します。

また JPEG モジュールソフトウェアリセットは MCU 単位ではない画像をエンコード処理した後に必ず実行する必要があります。これは毎フレーム連続エンコードをする場合には、MCU 単位の画像になる必要があることも意味しています。

8.5.5.7 マーカ高速出力モード

JPEG データでは JPEG マーカと呼ばれるいくつかのマーカでデータの種類を区別しています。JPEG コーデック回路はほとんどのマーカを圧縮データ部より前に配置しており、以前までの製品ではそのマーカを出力するのにシステムクロック 50MHz でおよそ 2ms かかっていました。S2S65A00 では新たにマーカ高速出力モードを追加し、JPEG 高速エンコードモードと同時に使用することにより、システムクロック 50MHz でおよそ 36μs でマーカ出力を終えることができるようになりました。

ハフマンテーブルを固定値にすることで高速化を図っていますので、JPEG 高速エンコードモードが無効になっているとレジスタ設定に関わらずマーカ高速出力モードも無効になります。

8.5.6 シーケンス例

8.5.6.1 カメラ画像 JPEG エンコード処理（単フレーム）

ここでは、カメラ画像を 1 枚だけ JPEG エンコードするシーケンスを説明します。FIFO からのデータリードには DMA を使用します。

1. カメラインタフェースの設定を行います。詳細はカメラインタフェース機能説明を参照してください。
2. JPEG モジュールをイネーブルし（JCTL[0x00] bit 0 = 1）、JPEG 動作モード（同レジスタ bits [3:1]）を“000”に設定します。
3. JPEG モジュールをソフトウェアリセットします（JCTL[0x00] bit 7 = 1）。
4. JPEG コーデックレジスタを初期化します。基本的にレジスタの若い順番から設定しますが、コマンドレジスタを除いて、順番に依存性はありません。
   (ア) JPEG コーデック回路をソフトウェアリセットします（JCODEC[0x04] bit 7 = 1）。
   (イ) JPEG エンコードの処理モードをエンコードにします（JCODEC[0x00] bit 2 = 0）。
   (ウ) 任意のマーカを挿入する場合は、マーカ挿入をイネーブルにします（JCODEC[0x00] bit 3 = 1）。
   (エ) 量子化テーブル番号およびハフマンテーブル番号を設定します（JCODEC[0x0C], [0x10]）。
   (オ) RST マーカを挿入する場合は、挿入する間隔を設定します（JCODEC[0x14], [0x18]）。
   (カ) 画像サイズを入力します（JCODEC[0x1C], [0x20], [0x24], [0x28]）。
   (キ) 挿入マーカを入力します（JCODEC[0x40 - 0xCC]）。
   (ク) 量子化テーブルを設定します。設定する順番は以下のとおりです（JCODEC[0x400 - 0x4FC], [0x500 - 0x5FC]）。

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

   (ケ) ハフマンテーブルを設定します。設定するデータは勧告書 ISO/IEC 10918 付属書 K を例にすると以下のようになります。

   DC ハフマンテーブル No. 0 レジスタ 0 には A を設定する（JCODEC[0x800 - 0x83C]）
   DC ハフマンテーブル No. 0 レジスタ 1 には B を設定する（JCODEC[0x840 - 0x86C]）
   AC ハフマンテーブル No. 0 レジスタ 0 には E を設定する（JCODEC[0x880 - 0x8BC]）
   AC ハフマンテーブル No. 0 レジスタ 1 には F を設定する（JCODEC[0x8C0 - 0xB44]）
   DC ハフマンテーブル No. 1 レジスタ 0 には C を設定する（JCODEC[0xC00 - 0xC3C]）
   DC ハフマンテーブル No. 1 レジスタ 1 には D を設定する（JCODEC[0xC40 - 0xC6C]）
   AC ハフマンテーブル No. 1 レジスタ 0 には G を設定する（JCODEC[0xC80 - 0xCBC]）
   AC ハフマンテーブル No. 1 レジスタ 1 には H を設定する（JCODEC[0xCC0 - 0xF44]）

A:	00h, 01h, 05h, ..., 00h, 00h	16 byte
B:	00h, 01h, 02h, ..., 0Ah, 0Bh	12 byte
C:	00h, 03h, 01h, ..., 00h, 00h	16 byte
D:	00h, 01h, 02h, ..., 0Ah, 0Bh	12 byte
E:	00h, 02h, 01h, 03h, ...1h, 7Dh	16 byte
F:	01h, 02h, 03h, ..., F9h, FAh	162 byte
G:	00h, 02h, 01h, 02h, ..., 02h, 77h	16 byte
H:	00h, 01h, 02h, ..., F9h, FAh	162 byte

5. JPEG モジュールを初期化します。
   (ア) JPEG FIFO サイズを設定します。FIFO は専用 RAM ですので 0x3F を入力します。（JFIFO[0x48] = 0x3F）
   (イ) エンコードサイズリミットをバイト単位で設定します（JFIFO[0x60], [0x64]）。
   (ウ) JPEG FIFO をクリアします（JFIFO[0x40] bit 2 = 1）。

6. リサイズ回路を初期化します。（RSZ[0xC0] bit 7 = 1）リサイズ後の画像サイズがステップ 4 (カ) で設定した値と同じになるように設定します。（RSZ[0xC8], [0xCC], [0xD0], [0xD4]）
7. 割り込みを設定します。JPEG ステータスフラグレジスタ（JCTL[0x04]）に 0x0000FFFF を書いて割り込みをクリアし、その後でラインバッファオーバーフロー割り込み、エンコードサイズリミット違反割り込みを有効にします（JCTL[0x0C] bit 2 = bit11 = 1）。JPEG ステータスフラグレジスタは予約になっているビットに“1”を書いても問題ありません。さらに割り込みコントローラの JPEG モジュールの割り込みイネーブルをイネーブルに設定します（INT[0x008]）。
8. JPEG DMA コントローラを設定します（JDMA[0x00]～[0x40]）。
9. JPEG エンコードをスタートさせます。
   (ア) JPEG コーデック回路をスタートさせます（JCODEC[0x04] bit 0 = 1）。
   (イ) JPEG モジュールをスタートさせます（JCTL[0x14] bit 0 = 1）。
   JPEG コーデック回路をスタートさせた後、JPEG マーカを出力するためにシステムクロック 50MHz でおよそ 2ms 必要となります。2ms 経過する前に JPEG モジュールをスタートさせた場合は、実際には 2ms 経過してからキャプチャ処理がスタートすることになります。
10. JPEG DMA コントローラのフレーム終了割り込みを待ちます。途中ラインバッファオーバーフロー割り込みが発生した場合やしばらくたっても JPEG エンコード処理が終了しない場合は終了処理を実行します。

8.5.6.2 終了処理

終了処理はすべての処理モードで共通にすることができます。処理に異常が起きた場合でも終了処理を実行することで初期状態に戻ります。JPEG コーデック回路レジスタにライトアクセスした場合、その後に別のレジスタへのダミーリードが必要ですが、このシーケンスでは省略しています。

1. リサイズ回路のグローバル制御レジスタ (RSZ[0x60]) に“0x0000”を書き込みます。これは予約レジスタに間違った値が入ってしまっている場合を想定しています。
2. リサイズ回路をイネーブル (RSZ[0xC0] bit 0 = 1) にし、さらにソフトウェアリセット (RSZ[0xC0] bit 7 = 1) します。
3. JPEG モジュール回路をイネーブルにし (JCTL[0x00] bit 0 = 1)、JPEG 動作モードを 000 にします (JCTL[0x00] bits 3-1 = 000)。
4. JPEG コーデック回路をソフトウェアリセットします (JCODEC[0x04] bit 7 = 1)。
5. JPEG コーデック回路の RST マーカ動作ステータスレジスタ (JCODEC[0x3C]) をダミーリードします。
6. JPEG コーデック回路の JPEG 動作ステータスレジスタ (JCODEC[0x08]) をダミーリードします。
7. JPEG コーデック回路の動作モード設定レジスタ (JCODEC[0x00]) に 0x00 を書きます。
8. JPEG モジュール回路をソフトウェアリセットします (JCTL[0x00] bit 7 = 1)。
9. JPEG ラインバッファ割り込み制御レジスタ (JLB[0x8C]) に 0x0000 を書き、すべてのラインバッファ割り込みをディセーブルにします。
10. JPEG ラインバッファステータスフラグレジスタ (JLB[0x80]) に 0xFFFF を書きます。
11. JPEG 割り込み制御レジスタ (JCTL[0x0C]) に 0x0000 を書き、すべての JPEG 割り込みをディセーブルにします。
12. JPEG ステータスフラグレジスタ (JCTL[0x04]) に 0xFFFF を書きます。
13. JPEG DMA 回路をソフトウェアリセットします (JDMA[0x20] bit 15 = 1)。
14. JPEG DMA 回路の DMA イネーブルおよび JPEG 割り込みイネーブルをディセーブルにします (JDMA[0x0C] bit 0 = 0、JDMA[0x0C] bit 21 = 0)。
15. JPEG DMA 回路の割り込みフラグをクリアします (JDMA[0x0C] bit 1 = 0)。
16. 割り込みコントローラの JPEG コントローラ割り込み、JPEG DMA 割り込みをディセーブルにします。
17. JPEG モジュールをディセーブルにします (JCTL[0x00] bit 0 = 0)。
18. リサイズ回路をディセーブルにします (RSZ[0xC0] bit 0 = 0)。

注意：JPEG モジュールはリサイズ回路よりも先にディセーブルにする必要があります。JPEG モジュールが最後にディセーブルになると、JPEG コーデック回路へのクロック供給が停止しない場合があります。

9. JPEG_DMAC (JDMA)

9.1 概要

JPEG_DMAC はカメラインタフェースから送られてくる画像データを専用に処理するための DMA コントローラです。

JPEG_DMAC は DMAC2 の機能を使用し、DMA 転送を行う DMAC2 に対して必要なレジスタデータを転送します。DMAC2 に対するリクエスト／アクノリッジは JPEG コントローラ内蔵の FIFO が行います。

9.2 ブロック図

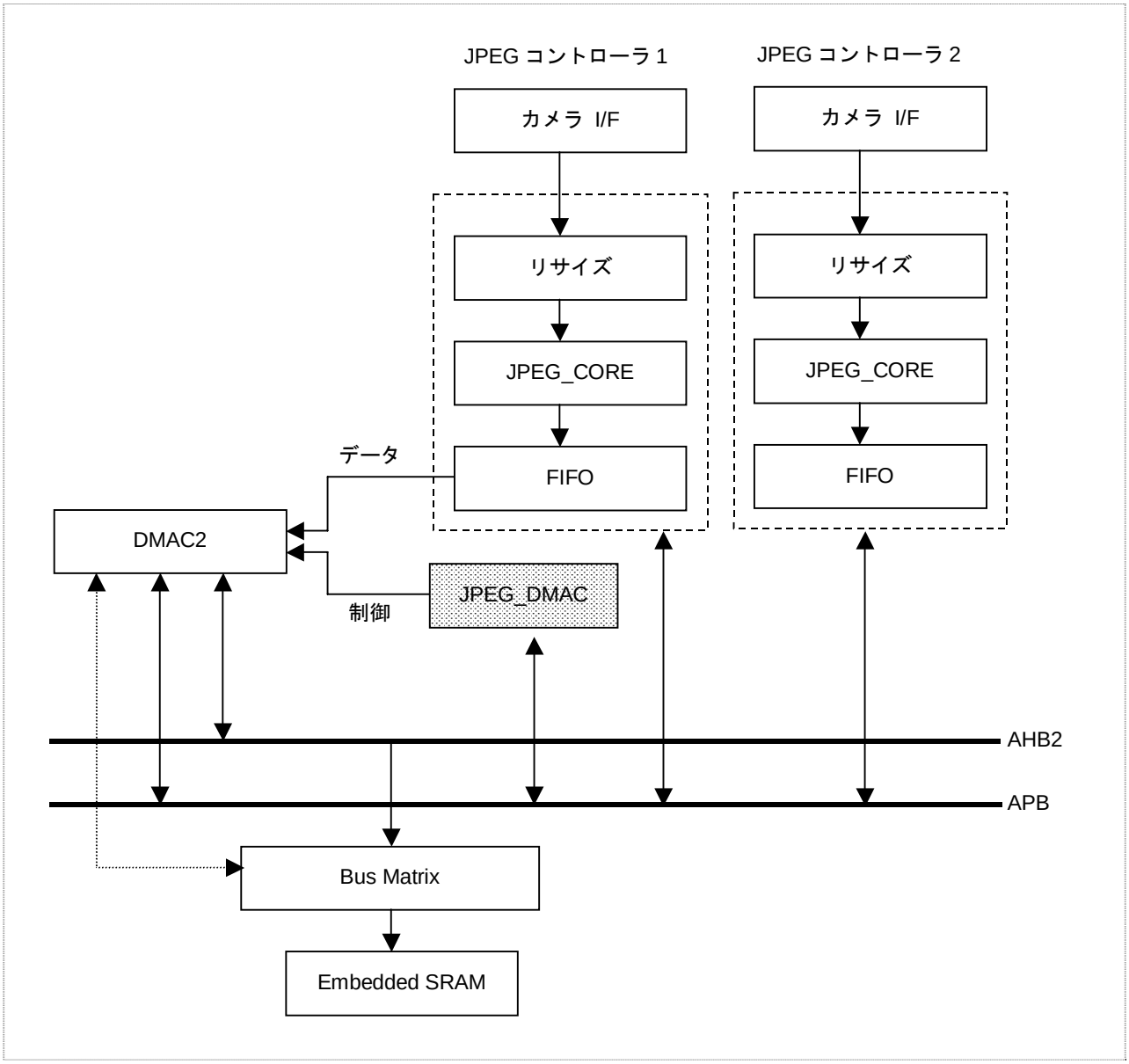


図 9.1　JPEG_DMAC と JPEG コントローラ、DMAC2 等の関係ブロック図

9.3 外部端子

JPEG_DMAC に関連する外部端子はありません。

9.4 レジスタ

9.4.1 レジスター覧

JPEG_DMAC のレジスタのベースアドレスは 0xFFFE_C000 です。

表 9.1 レジスター覧（**ベースアドレス：0xFFFE_C000**）

アドレス オフセット	レジスタ名称	レジスタ 略称	初期値*	R/W	データ アクセス サイズ
0x00	DMA チャネル 0 JPEG ソースアドレスレジスタ	JSAR0	0xXXXX_XXXX	R/W	32
0x04	DMA チャネル 0 JPEG デスティネーションアドレスレジスタ	JDAR0	0xXXXX_XXXX	R/W	32
0x08	DMA チャネル 0 JPEG 転送カウントレジスタ	JTCR0	0x0000_0000	R/W	32
0x0C	DMA チャネル 0 JPEG コントロールレジスタ	JCTL0	0x0000_0000	R/W	32
0x10	DMA チャネル 0 JPEG ブロックカウントレジスタ	JBCR0	0x00XX_XXXX	R/W	32
0x14	DMA チャネル 0 JPEG デスティネーションオフセットアドレスレジスタ	JOFR0	0x0000_0000	R/W	32
0x18	DMA チャネル 0 JPEG ブロックエンドカウントレジスタ	JBER0	0x00XX_XXXX	R/W	32
0x1C	-	-	-	-	-
0x20	DMA チャネル 1 JPEG ソースアドレスレジスタ	JSAR1	0xXXXX_XXXX	R/W	32
0x24	DMA チャネル 1 JPEG デスティネーションアドレスレジスタ	JDAR1	0xXXXX_XXXX	R/W	32
0x28	DMA チャネル 1 JPEG 転送カウントレジスタ	JTCR1	0x0000_0000	R/W	32
0x2C	DMA チャネル 1 JPEG コントロールレジスタ	JCTL1	0x0000_0000	R/W	32
0x30	DMA チャネル 1 JPEG ブロックカウントレジスタ	JBCR1	0x00XX_XXXX	R/W	32
0x34	DMA チャネル 1 JPEG デスティネーションオフセットアドレスレジスタ	JOFR1	0x0000_0000	R/W	32
0x38	DMA チャネル 1 JPEG ブロックエンドカウントレジスタ	JBER1	0x00XX_XXXX	R/W	32
0x3C	-	-	-	-	-
0x40	DMA チャネル JPEG FIFO データ選択モードレジスタ	JFSM	0x0000_0000	R/W	32
0x48	DMA チャネル JPEG 拡張レジスタ	JHID	0x0000_0000	R/W	32

*注意： X：不定(h)
補足）DMA チャネル 0 は JPEG コントローラ 1 で、DMA チャネル 1 は JPEG コントローラ 2 で使用することを推奨します。

9.4.2 レジスタ詳細説明

DMA チャネル 0/1 JPEG ソースアドレスレジスタ (JSAR0/1)

JDMA[0x00], [0x20] 初期値 = 0xXXXX_XXXX Read/Write

DMA チャネル 0/1 JPEG ソースアドレス [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル 0/1 JPEG ソースアドレス [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] : **DMA チャネル 0/1 JPEG ソースアドレス [31:0]**

これらのビットには JPEG_DMA 転送のソースアドレスをソフトウェアにより設定します。ハードウェアにより更新することはありません。
JPEG1 を使用する場合は、0xE000_004C を設定してください。
JPEG2 を使用する場合は、0xD000_004C を設定してください。

DMA チャネル 0/1 JPEG デスティネーションアドレスレジスタ (JDAR0/1)

JDMA[0x04], [0x24] 初期値 = 0xXXXX_XXXX Read/Write

DMA チャネル 0/1 JPEG デスティネーションアドレス [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル 0/1 JPEG デスティネーションアドレス [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] : **DMA チャネル 0/1 JPEG デスティネーションアドレス [31:0]**

これらのビットには JPEG_DMA 転送のデスティネーションアドレスをソフトウェアにより設定します。DMA 転送の 1 ブロックが終了すると現在設定されている値に DMA チャネル JPEG デスティネーションオフセットアドレスレジスタに設定されているオフセット値が自動的に加算されます。

DMA チャネル 0/1 JPEG 転送カウントレジスタ (JTCR0/1)

JDMA[0x08], [0x28] 初期値 = 0x0000_0000 Read/Write

n/a								DMA チャネル 0/1 JPEG 転送カウント [23:16]							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル 0/1 JPEG 転送カウント [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:0] : **DMA チャネル 0/1 JPEG 転送カウント [23:0]**

これらのビットには JPEG_DMA 転送時の転送回数をソフトウェアにより設定します。この値をハードウェアで更新することはありません。
このレジスタをリードすると bits[31:24]には“0”が付加されます。

DMA チャネル 0/1 JPEG コントロールレジスタ (JCTL0/1)

JDMA[0x0C], [0x2C] 初期値 = 0x0000_0000 Read/Write

n/a								RSV	JS	JIE	JCS1	JCS0	RSV	AM	AL
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DAM		SAM		RS				RSV		TM	TS		IE	JTE	DE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 23 : **RSV 予約（0）**

Bit 22 : **JS JPEG_DMA 転送スタート**

0： 1 ブロック転送終了または FIFO_END がアサートされるとこのビットは自動的に“0”にクリアされます。
1： JPEG_DMA 転送を開始するときに“1”にセットされます。

Bit 21 : **JIE　JPEG 割り込みイネーブル**

このビットが“1”のときに JTE（Bit 1）がセットされると、割り込みをアサートします。

0： 割り込みディセーブル

1： 割り込みイネーブル

Bit [20:19] : **JCS[1:0]　DMA チャネル選択**

使用する DMAC2 の DMA チャネルを選択します。

00：DMA チャネル 0

01：DMA チャネル 1

10：DMA チャネル 2

11：DMA チャネル 3

Bit 18 : **RSV　予約（0）**

これらのビットには“0”を設定してください。

Bit 17 : **AM　アクノリッジモード**

DACK 信号の出力タイミングを選択します。

0： DMA リードサイクルの間アクティブ

1： DMA ライトサイクルの間アクティブ

Bit 16 : **AL　アクノリッジレベル**

DACK 信号の出力極性を選択します。

0： LOW アクティブ

1： HIGH アクティブ

Bits [15:14] : **DAM　デスティネーションアドレスモード [1:0]**

1 回転送終了後のデスティネーションアドレスレジスタの更新モードを選択します。

00：転送先アドレス固定（更新しない）

01：転送先アドレスを転送データサイズに合わせてインクリメント
（8 ビットは+1、16 ビットは+2、32 ビットは+4）

10：転送先アドレスを転送データサイズに合わせてデクリメント
（8 ビットは-1、16 ビットは-2、32 ビットは-4）

11：予約

Bits [13:12] : **SAM　ソースアドレスモード [1:0]**

1 回転送終了後のソースアドレスレジスタの更新モードを選択します。

00：転送元アドレス固定（更新しない）

01：転送元アドレスを転送データサイズに合わせてインクリメント
（8 ビットは+1、16 ビットは+2、32 ビットは+4）

10：転送元アドレスを転送データサイズに合わせてデクリメント
（8 ビットは-1、16 ビットは-2、32 ビットは-4）

11：予約

Bits [11:8] : **RS　リソース選択 [3:0]**

DMA 転送を開始させる要因を選択します。

0010：JPEG1

0011：JPEG2

Other：予約

Bits [7:6] : **RSV　予約（0）**

Bit 5 : **TM　送信モード**

DMA 転送の送信モードを選択します。

0： シングル転送

1： デマンド転送

Bits [4:3] : **TS　転送データサイズ**

1 回で転送するデータサイズを選択

00：8 ビット
01：16 ビット
10：32 ビット
11：予約

Bit 2 : **IE　割り込みイネーブル**

0： 1 回のブロック転送終了後に割り込みを発生しない
1： 1 回のブロック転送終了後に割り込みを発生

Bit 1 : **JTE　JPEG_DMA 転送終了**

0 (リード時)：転送中または待機中
1 (リード時)：JPEG_DMA 転送終了
0 (ライト時)：本ビットをクリア
1 (ライト時)：無効

このビットは DMA チャネル JPEG 転送ブロックカウントレジスタの値が“0”になるとセットされます。一度セットされるとこのビットに“0”をライトして本ビットをクリアするまで“1”を保持します。本ビットがクリアされるまでは、このチャネルの DMA 転送が禁止されます。このビットは割り込みフラグとしても機能します。

Bit 0 : **DE　DMA イネーブル**

このビットにより JPEG_DMA 転送を許可します。

0： JPEG_DMA 転送禁止
1： JPEG_DMA 転送許可

DMA チャネル 0/1 JPEG ブロックカウントレジスタ　(JBCR0/1)

JDMA[0x10], [0x30]　　初期値 = 0x00XX_XXXX　　Read/Write

n/a								DMA チャネル 0/1 JPEG ブロックカウント [23:16]							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル 0/1 JPEG ブロックカウント [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:0] : **DMA チャネル 0/1 JPEG ブロックカウント [23:0]**

これらのビットには JPEG_DMA 転送時のブロック転送回数をソフトウェアにより設定します。JPEG_DMA 転送の 1 ブロックが終了するとこのレジスタの値から自動的に“1”デクリメントします。

このレジスタをリードすると bits[31:24]には“0”が付加されます。

DMA チャネル 0/1 JPEG デスティネーションオフセットアドレスレジスタ　(JOFR0/1)

JDMA[0x14], [0x34]　　初期値 = 0x0000_0000　　Read/Write

n/a								DMA チャネル0/1 JPEG デスティネーションオフセットアドレス [23:16]							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル 0/1 JPEG デスティネーションオフセットアドレス [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:0] : **DMA チャネル 0/1 JPEG デスティネーションオフセットアドレス [23:0]**

これらのビットには JPEG_DMA 転送時のオフセット値をソフトウェアにより設定します。この値をハードウェアで更新することはありません。

このレジスタをリードすると bits[31:24]には“0”が付加されます。

DMA チャネル 0/1 JPEG ブロックエンドカウントレジスタ　(JBER0/1)

JDMA[0x18], [0x38]　　初期値 = 0x00XX_XXXX　　Read/Write

n/a								DMA チャネル JPEG ブロックエンドカウント [23:16]							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル JPEG ブロックエンドカウント [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:0]：　**DMA チャネル 0/1 JPEG 転送ブロックエンドカウント [23:0]**

これらのビットには JPEG_DMA 転送時のブロック転送回数をソフトウェアにより設定します。
JPEG_DMA 転送の 1 ブロック転送が終了するとこのレジスタの値に“1”インクリメントします。
通常は現在のブロック転送回数を表示します。
このレジスタをリードすると bits[31:24]には“0”が付加されます。

DMA チャネル JPEG FIFO データ選択モードレジスタ　(JFSM)

JDMA[0x40]　　初期値 = 0x0000_0000　　Read/Write

n/a		RSV						n/a		RSV		JF1B	RSV		
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a		RSV						n/a		RSV		JF0B	RSV		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [29:24]：　**RSV　予約（0）**
Bits [21:20]：　**RSV　予約（0）**
Bits [18:16]：　**RSV　予約（0）**
Bits [13:8]：　**RSV　予約（0）**
Bits [5:4]：　**RSV　予約（0）**

Bit 19, 3：　**JF[1:0]B　FIFO モード**

JPEG FIFO[1:0]のデータ出力バスの設定を行います。
0： APB バス側からのアクセスの対応
1： AHB バス側からのアクセスの対応
JF1B は JPEG2、JF0B は JPEG1 にそれぞれ対応します。

Bits [2:0]：　**RSV　予約（0）**

DMA チャネル JPEG 拡張レジスタ　(JHID)

JDMA[0x48]　　初期値 = 0x0000_0000　　Read/Write

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SW	n/a														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 15：　**SW　ソフトウェアリセット**

JDMA 内のすべてのレジスタをソフトウェアリセットします。本ビットに“1”を書き込むことでソフトウェアリセットを実行します。ソフトウェアリセットの完了後に本ビットは自動的に“0”に戻ります。

10. DMA コントローラ 2 (DMAC2)

10.1 概要

DMAC2 は、JPEG_DMAC から送られてくる制御情報もしくはソフトウェア設定による制御情報をもとに DMA 転送を行う DMA コントローラです。

10.2 ブロック図

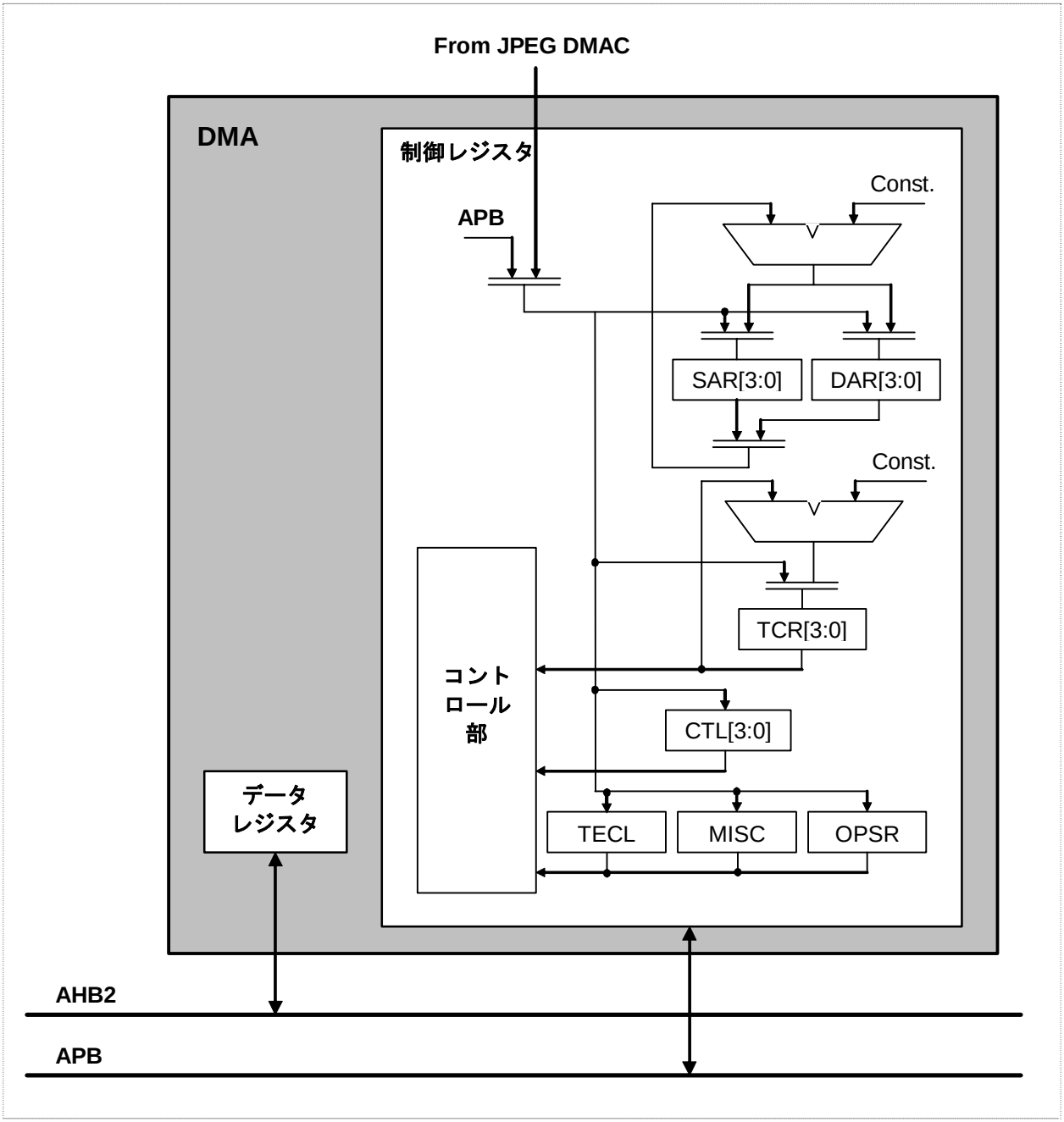


図 10.1　DMA コントローラ 2 ブロック図

10.3 外部端子

DMA コントローラ 2 に関連する外部端子はありません。

10.4 レジスタ

10.4.1 レジスター覧

DMAC2 関連レジスタのベースアドレスは 0xFFFF_9000 です。

表 10.1 レジスター覧（**ベースアドレス：0xFFFF_9000**）

アドレス オフセット	レジスタ名称	レジスタ 略称	初期値	R/W	データ アクセス サイズ
0x00	DMA チャネル 0 ソースアドレスレジスタ	SAR0	0xXXXX_XXXX	R/W	32
0x04	DMA チャネル 0 デスティネーションアドレスレジスタ	DAR0	0xXXXX_XXXX	R/W	32
0x08	DMA チャネル 0 転送カウントレジスタ	TCR0	0x00XX_XXXX	R/W	32
0x0C	DMA チャネル 0 コントロールレジスタ	CTL0	0x0000_0000	R/W	32
0x10	DMA チャネル 1 ソースアドレスレジスタ	SAR1	0xXXXX_XXXX	R/W	32
0x14	DMA チャネル 1 デスティネーションアドレスレジスタ	DAR1	0xXXXX_XXXX	R/W	32
0x18	DMA チャネル 1 転送カウントレジスタ	TCR1	0x00XX_XXXX	R/W	32
0x1C	DMA チャネル 1 コントロールレジスタ	CTL1	0x0000_0000	R/W	32
0x20	DMA チャネル 2 ソースアドレスレジスタ	SAR2	0xXXXX_XXXX	R/W	32
0x24	DMA チャネル 2 デスティネーションアドレスレジスタ	DAR2	0xXXXX_XXXX	R/W	32
0x28	DMA チャネル 2 転送カウントレジスタ	TCR2	0x00XX_XXXX	R/W	32
0x2C	DMA チャネル 2 コントロールレジスタ	CTL2	0x0000_0000	R/W	32
0x30	DMA チャネル 3 ソースアドレスレジスタ	SAR3	0xXXXX_XXXX	R/W	32
0x34	DMA チャネル 3 デスティネーションアドレスレジスタ	DAR3	0xXXXX_XXXX	R/W	32
0x38	DMA チャネル 3 転送カウントレジスタ	TCR3	0x00XX_XXXX	R/W	32
0x3C	DMA チャネル 3 コントロールレジスタ	CTL3	0x0000_0000	R/W	32
0x40 – 0x5C	予約	-	-	-	-
0x60	DMA チャネル オペレーティング選択レジスタ	OPSR	0x0000_0000	R/W	32
0x64	DMA チャネル MISC レジスタ	MISC	0x0000_0000	R/W	32
0x70	DMA チャネル 転送終了コントロールレジスタ	TECL	0x0000_0000	R/W	32

10.4.2 レジスタ詳細

DMA チャネル[3:0] ソースアドレスレジスタ (SAR[3:0])

DMAC2[0x00], [0x10], [0x20], [0x30] 初期値 = 0xXXXX_XXXX Read/Write

DMA チャネル[3:0] ソースアドレス [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル[3:0] ソースアドレス [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] : **DMA チャネル[3:0] ソースアドレス [31:0]**

チャネル[3:0]で DMA 転送を行う場合の転送元アドレスをソフトウェアにより設定します。転送元アドレスは転送データサイズに合った境界アドレスを設定する必要があります。例えば 32 ビット転送の場合、このレジスタのビット[1:0]は 00b でなければなりません。DMA 転送開始後は、1 回の転送終了ごとに転送データサイズ（TS : チャネル[3:0]コントロールレジスタ ビット[4:3]）とソースアドレスモード（SAM : チャネル[3:0]コントロールレジスタ ビット[13:12]）に従った次の転送元アドレスに自動的に更新されます。

DMA チャネル[3:0] デスティネーションアドレスレジスタ (DAR[3:0])

DMAC2[0x04], [0x14], [0x24], [0x34] 初期値 = 0xXXXX_XXXX Read/Write

DMA チャネル[3:0] デスティネーションアドレス [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル[3:0] デスティネーションアドレス [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] : **DMA チャネル[3:0] デスティネーションアドレス [31:0]**

チャネル[3:0]で DMA 転送を行う場合の転送先アドレスをソフトウェアにより設定します。転送先アドレスは転送データサイズに合った境界アドレスを設定する必要があります。例えば 32 ビット転送の場合、このレジスタのビット[1:0]は 00b でなければなりません。DMA 転送開始後は、1 回の転送終了ごとに転送データサイズ（TS : チャネル[3:0]コントロールレジスタ ビット[4:3]）とデスティネーションアドレスモード（DAM : チャネル[3:0]コントロールレジスタ ビット[15:14]）に従った次の転送先アドレスに自動的に更新されます。

DMA チャネル[3:0] 転送カウントレジスタ (TCR[3:0])

DMAC2[0x08], [0x18], [0x28], [0x38] 初期値 = 0x00XX_XXXX Read/Write

n/a								DMA チャネル[3:0] 転送カウント [23:16]							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA チャネル[3:0] 転送カウント [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:0] : **DMA チャネル[3:0] 転送カウント [23:0]**

これらのビットには DMA 転送時の転送回数をソフトウェアにより設定します。DMA 転送を開始すると、1 回の転送終了ごとにデクリメントされます。ここに“0”を設定した場合、転送回数は 2^{24}=16777216 回になります。DMA 割り込みはこのカウンタがカウントダウンによって“0”になった時点で発生します。

このレジスタをリードすると bits[31:24]には“0”が付加されます。

DMA チャネル[3:0] コントロールレジスタ (CTL[3:0])

DMAC2[0x0C], [0x1C], [0x2C], [0x3C]　　初期値 = 0x0000_0000　　Read/Write

n/a								RSV					IB4	AM	AL
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DAM		SAM		RS				RSV	RIM	TM	TS		IE	TE	DE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [23:19]：　**RSV　予約（0）**

Bit 18：　**IB4　インクリメントバースト 4**

このビットが“1”のときにインクリメントバースト 4 転送を実行します。ただしインクリメントバースト 4 転送で転送するデータはデータサイズ 4 個分で割り切れなければなりません。データサイズ 4 個分で割り切れないサイズのデータを転送する場合には、少なくとも端数は別の DMA 転送としてこのビットを“0”に設定して転送する必要があります。
この機能はメモリからメモリへのブロック転送にのみ使用します。

Bit 17：　**AM　アクノリッジモード**

DACK 信号の出力タイミングを選択します。
0： DMA リードサイクルの間アクティブ
1： DMA ライトサイクルの間アクティブ

Bit 16：　**AL　アクノリッジレベル**

DACK 信号の出力極性を選択します。
0： LOW アクティブ
1： HIGH アクティブ

Bits [15:14]：　**DAM　デスティネーションアドレスモード [1:0]**

1 回転送終了後のデスティネーションアドレスレジスタの更新モードを選択します。
00：転送先アドレス固定（更新しない）
01：転送先アドレスを転送データサイズに合わせてインクリメント
（8 ビットは+1、16 ビットは+2、32 ビットは+4）
10：転送先アドレスを転送データサイズに合わせてデクリメント
（8 ビットは-1、16 ビットは-2、32 ビットは-4）
11：予約

Bits [13:12]：　**SAM　ソースアドレスモード [1:0]**

1 回転送終了後のソースアドレスレジスタの更新モードを選択します。
00：転送元アドレス固定（更新しない）
01：転送元アドレスを転送データサイズに合わせてインクリメント
（8 ビットは+1、16 ビットは+2、32 ビットは+4）
10：転送元アドレスを転送データサイズに合わせてデクリメント
（8 ビットは-1、16 ビットは-2、32 ビットは-4）
11：予約

Bits [11:8]：　**RS　リソース選択 [3:0]**

DMA 転送を開始させる要因を選択します。
1111：**SW-Request**　ソフトウェアリクエスト
Bits[11:8]を“1111”にセットすることでソフトウェアの DMA 転送を開始します。
なお、設定可能なアドレスは、4.2.2 メモリマップ (AHB2) でマッピングされているアドレスのみとなります。
Other：予約

Bit 7：　**RSV　予約（0）**

Bit 6 : **RIM　リクエスト入力モード**

DMA 要求信号の入力モードを選択します。

0： LOW アクティブ（レベルトリガ）

1： 立ち下がりエッジ（エッジトリガ）

Bit 5 : **TM　転送モード**

DMA 転送の転送モードを選択します。

0： シングル転送

1： デマンド転送

Bits [4:3] : **TS　転送サイズ [1:0]**

1 回で転送するデータサイズを選択

00：8 ビット

01：16 ビット

10：32 ビット

11：予約

Bit 2 : **IE　割り込みイネーブル**

DMA チャネル[3:0]の転送終了割り込みを許可／禁止します。

0： 割り込み禁止

1： 割り込み許可

Bit 1 : **TE　DMA 転送終了**

0 (リード時)：転送中または待機中

1 (リード時)：JPEG_DMA 転送終了

0 (ライト時)：本ビットをクリア

1 (ライト時)：無効

このビットは DMA チャネル[3:0]転送カウントレジスタの値が“0”になるとセットされます。一度セットされるとこのビットに“0”をライトして本ビットをクリアするまで“1”を保持します。本ビットがクリアされるまでは、このチャネルの DMA 転送が禁止されます。このビットは割り込みフラグとしても機能します。

Bit 0 : **DE　DMA イネーブル**

このビットによりチャネル[3:0]の DMA 転送を許可します。

0： DMA 転送禁止

1： DMA 転送許可

DMA チャネル オペレーティング選択レジスタ (OPSR)

DMAC2[0x60] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
n/a						DPE	DPM	n/a							DGE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 9 : **DPE DMA プライオリティ変更イネーブル**

0 : プライオリティの変更を行わない。

1 : プライオリティの変更を行う。

本ビットが“1”の場合、以下の条件で Bit 8 の DPM の値を変更します。

シングル転送時 : 1 回の転送ごとに DPM を変更します。

デマンド転送時 : リクエストがネゲートされ、転送が中断したとき、もしくは転送カウンタが“0”となったときに変更します。

Bit 8 : **DPM DMA プライオリティモード**

0 : CH0>CH1>CH2>CH3 の優先

1 : CH1>CH0>CH2>CH3 の優先

Bit 0 : **DGE DMA グローバルイネーブル**

DMA の全チャネルをすべてイネーブル／ディセーブルにします。

0 : ディセーブル

1 : イネーブル

DMA チャネル MISC レジスタ (MISC)

DMAC2[0x64] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
SR	n/a											DPL			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 15 : **SR ソフトウェアリセット**

この Bit に“1”を書き込むことで、DMAC2 内のレジスタを初期化することができます。

DMAC2 内のすべてのレジスタが初期化されますので、必要なレジスタは再設定を行ってください。

Bits [3:0] : **DPL DMA 極性選択[1:0]**

DPL3 DMA チャネル 3 の極性を選択します。

0 : 正

1 : 負

DPL2 DMA チャネル 2 の極性を選択します。

0 : 正

1 : 負

DPL1 DMA チャネル 1 の極性を選択します。

0 : 正

1 : 負

DPL0 DMA チャネル 0 の極性を選択します。

0 : 正

1 : 負

DMA チャネル 転送終了コントロールレジスタ (TECL)

DMAC2[0x70]　　初期値 = 0x0000_0000　　Read/Write

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a		STTE	ENTE	n/a											
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 13 :　**STTE　転送終了受付による TE セット**

この Bit に“1”を書き込むことで、リクエスト発行要因からの転送終了信号を受け付け、TE をセットします。

この Bit は Bit12 が“1”にセットされたときに有効となります。

Bits 12 :　**ENTE　転送終了受け付けイネーブル**

この Bit に“1”を書き込むことで、リクエスト発行要因からの転送終了信号を受け付けします。但し、TE のセットは行わず TE をセットする場合は Bit13 に“1”を書き込んでください。

11. USB HS-Device

11.1 概要

本 USB HS-Device Macro は SEIKO EPSON USB2.0 準拠 High Speed USB-Host/Device Combination Controller (S1R72V05)に搭載されている USB HS-Device Controller 部分を SoC (System On Chip) 搭載用途向けに切り出し Macro 化したものです。

本 USB HS-Device Macro の詳細は“Appendix2 USB Device コントローラ”を参照してください。

11.2 ブロック図

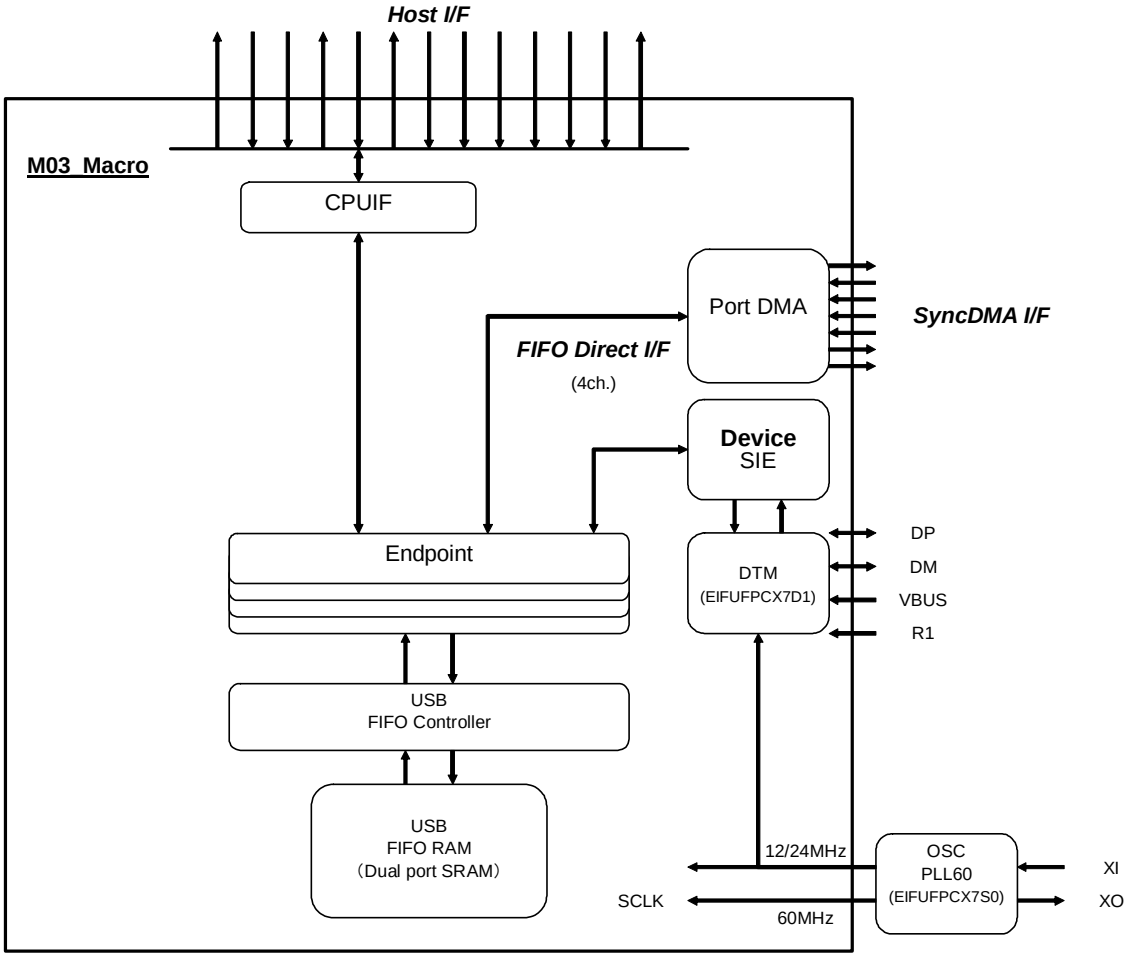


図 11.1 USB HS-Device ブロック図

11.3 特長

本 USB HS-Device Macro は以下の基本機能を有します。

<USB2.0 Device Function>
- ・HS(480Mbps)および FS(12Mbps)の転送 Support。
- ・FS/HS Termination 機能内蔵(外付け回路不要)。
- ・VBUS 5V Interface (但し外付け保護回路が必要となります。)
- ・Control Bulk および Interrupt 転送を Support。
- ・Control(End Point 0)および Bulk/Interrupt 共用 End Point（8 本）を Support。

<CPU Interface>
- ・16-bit または 8-bit 幅の汎用 CPU Interface 搭載。
- ・Littler Endian 対応。
- ・Register Table については S1R72V05 の HS-Device 部を基本として追加／削除。

<発振回路>
- ・USB 用 Clock 入力として 12MHz または 24MHz 水晶発振子入力に対応。
  (発振回路および帰還抵抗として 1MΩ)内蔵。
- ・内部 Clock 用途としては入力 USB 用 Clock を元にして、以下の周波数が使用可能です。
  入力 USB 用 Clock：12MHz または 24MHz
  内部 USB 用 Clock：60MHz(内蔵 USB 用 PLL 経由)

12. APB ブリッジ（APB）

12.1 概要

本モジュールは、内部の高速バスーAHB1 と低速の APB 機能デバイス(以下 APB デバイス)が接続されている APB バス間におかれ、AHB バスへの制御を各 APB デバイスに代わって行うブリッジ機能を持った AHB バス上のスレーブ・デバイスです。この APB ブリッジを経由することにより、個々の APB デバイスは AHB バスの制御から開放され、その代わりシンプルな APB バスに対する制御のみを行えば済むことになります。ソフトウエアのプログラマにとっては、この APB ブリッジは本来気にすることのないデバイスですが、場合によっては APB デバイスに対するウェイト機能(0～3 ウェイト)を設定する必要があります。通常はリセット時のままお使いください。

12.2 ブロック図

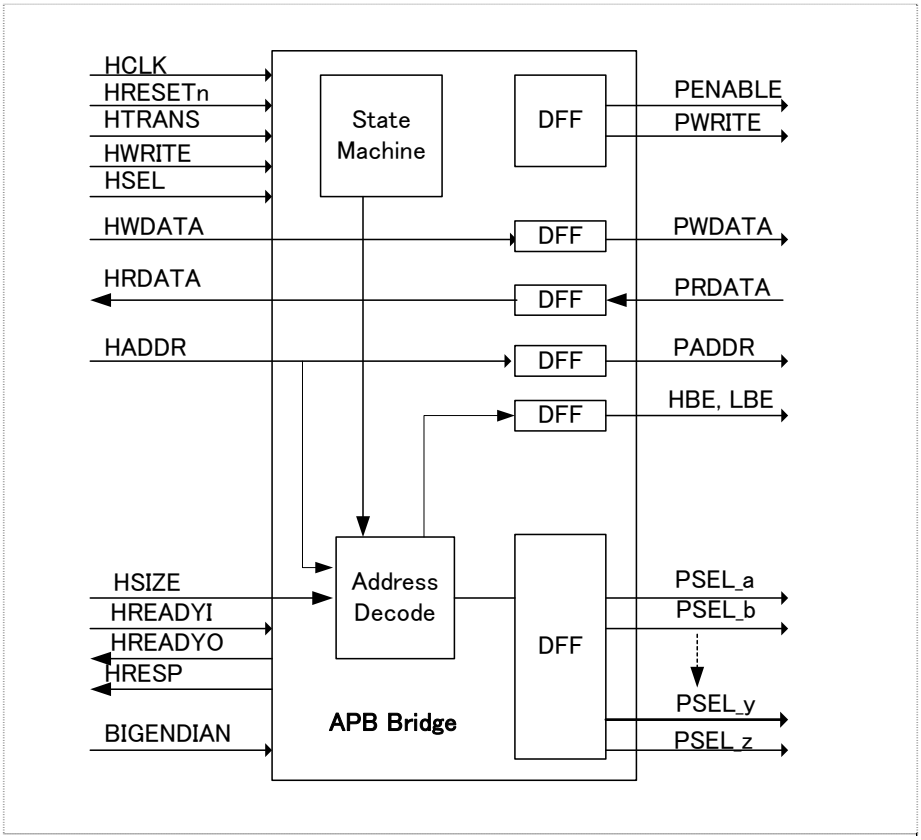


図 12.1　APB ブリッジ プロック図

12.3 外部端子

APB ブリッジに関連する外部端子はありません。

12.4 レジスタ

12.4.1 レジスター覧

APB 関連レジスタのベースアドレスは 0xFFFE_0000 です。

表 12.1　レジスター覧（ベースアドレス：0xFFFE_0000）

アドレス オフセット	レジスタ名称	レジスタ 略称	初期値	R/W	データ アクセス サイズ
0x00	APB WAIT0 Register	APBWAIT0	0x0050_0500	R/W	32bit
0x04	APB WAIT1 Register	APBWAIT1	0x0000_0000	R/W	32bit
0x08	APB WAIT2 Register	APBWAIT2	0x0050_0000	R/W	32bit

12.4.2 レジスタ詳細説明

APB WAIT0 レジスタ　(APBWAIT0)

APB[0x00]　　初期値 = 0x0050_0500　　Read/Write

PW0FCNF [1:0]		PW0ECNF [1:0]		PW0DCNF [1:0]		PW0CCNF [1:0]		PW0BCNF [1:0]		PW0ACNF [1:0]		PW09CNF [1:0]		PW08CNF [1:0]	
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PW07CNF [1:0]		PW06CNF [1:0]		PW05CNF [1:0]		PW04CNF [1:0]		PW03CNF [1:0]		PW02CNF [1:0]		PW01CNF [1:0]		PW00CNF [1:0]	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] :　**PWxCNF[1:0] (x=00 to 0F)**

00：0 ウェイト　基本の 2-APB サイクル（デフォルト）
01：1 ウェイト　2-APB サイクル＋1-ウェイトサイクル＝3-APB サイクル
10：2 ウェイト　2-APB サイクル＋2-ウェイトサイクル＝4-APB サイクル
11：3 ウェイト　2-APB サイクル＋3-ウェイトサイクル＝5-APB サイクル

APB WAIT1 レジスタ　(APBWAIT1)

APB[0x04]　　初期値 = 0x0000_0000　　Read/Write

PW1FCNF [1:0]		PW1ECNF [1:0]		PW1DCNF [1:0]		PW1CCNF [1:0]		PW1BCNF [1:0]		PW1ACNF [1:0]		PW19CNF [1:0]		PW18CNF [1:0]	
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PW17CNF [1:0]		PW16CNF [1:0]		PW15CNF [1:0]		PW14CNF [1:0]		PW13CNF [1:0]		PW12CNF [1:0]		PW11CNF [1:0]		PW10CNF [1:0]	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] :　**PWxCNF[1:0] (x=10 to 1F)**

00：0 ウェイト　基本の 2-APB サイクル（デフォルト）
01：1 ウェイト　2-APB サイクル＋1-ウェイトサイクル＝3-APB サイクル
10：2 ウェイト　2-APB サイクル＋2-ウェイトサイクル＝4-APB サイクル
11：3 ウェイト　2-APB サイクル＋3-ウェイトサイクル＝5-APB サイクル

APB WAIT2 レジスタ　(APBWAIT2)

APB[0x08]　　初期値 = 0x0050_0000　　Read/Write

PW2FCNF [1:0]		PW2ECNF [1:0]		PW2DCNF [1:0]		PW2CCNF [1:0]		PW2BCNF [1:0]		PW2ACNF [1:0]		PW29CNF [1:0]		PW28CNF [1:0]	
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PW27CNF [1:0]		PW26CNF [1:0]		PW25CNF [1:0]		PW24CNF [1:0]		PW23CNF [1:0]		PW22CNF [1:0]		PW21CNF [1:0]		PW20CNF [1:0]	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**PWxCNF[1:0] (x=20 to 2F)**

00：0 ウェイト　基本の 2-APB サイクル（デフォルト）
01：1 ウェイト　2-APB サイクル＋1-ウェイトサイクル＝3-APB サイクル
10：2 ウェイト　2-APB サイクル＋2-ウェイトサイクル＝4-APB サイクル
11：3 ウェイト　2-APB サイクル＋3-ウェイトサイクル＝5-APB サイクル

表 12.2　APBWAIT[2:0]レジスタと APB Device の対応一覧

APBWAIT0		APBWAIT1		APBWAIT2	
PW0xCNF	APB Device	PW1xCNF	APB Device	PW2xCNF	APB Device
PW00CNF	APB ブリッジ	PW10CNF	Reserved	PW20CNF	Reserved
PW01CNF	Reserved	PW11CNF	GPIO	PW21CNF	Reserved
PW02CNF	Reserved	PW12CNF	SPI	PW22CNF	Reserved
PW03CNF	DMA1C	PW13CNF	Reserved	PW23CNF	Timer B
PW04CNF	CF Attribute CF Common	PW14CNF	Reserved	PW24CNF	UART2
PW05CNF	CF I/O	PW15CNF	UART1	PW25CNF	UART3
PW06CNF	CF CTL	PW16CNF	Reserved	PW26CNF	Reserved
PW07CNF	ARS	PW17CNF	Reserved	PW27CNF	Reserved
PW08CNF	Camera I/F-1	PW18CNF	RTC	PW28CNF	Camera I/F-2
PW09CNF	JPEG Resize-1	PW19CNF	DMA2C	PW29CNF	JPEG Resize-2
PW0ACNF	JPEG module/FIFO-1	PW1ACNF	Memory CNTL	PW2ACNF	JPEG module/FIFO-2
PW0BCNF	JPEG Codec-1	PW1BCNF	Timer A	PW2BCNF	JPEG Codec-2
PW0CCNF	JPEG-DMAC	PW1CCNF	WDT	PW2CCNF	ADC
PW0DCNF	I2C	PW1DCNF	SYS-CNTL	PW2DCNF	SD-Memory
PW0ECNF	I2S	PW1ECNF	Reserved	PW2ECNF	Reserved
PW0FCNF	(INT-CNTL)	PW1FCNF	INT-CNTL	PW2FCNF	USB_2.0 Device

表 12.3　内蔵 I/O マップ

Base Address	Space Size	S2S65A00
0xFFFD_0000	4KB	Reserved
0xFFFD_1000	4KB	Reserved
0xFFFD_2000	4KB	Reserved
0xFFFD_3000	4KB	**Timer B**
0xFFFD_4000	4KB	**UART2**
0xFFFD_5000	4KB	**UART3**
0xFFFD_6000	4KB	Reserved
0xFFFD_7000	4KB	Reserved
0xFFFD_8000	4KB	**Camera I/F-2**
0xFFFD_9000	4KB	**JPEG Resize-2**
0xFFFD_A000	4KB	**JPEG module/FIFO-2**
0xFFFD_B000	4KB	**JPEG Codec-2**
0xFFFD_C000	4KB	**ADC**
0xFFFD_D000	4KB	**SD-Memory**
0xFFFD_E000	4KB	Reserved
0xFFFD_F000	4KB	**USB_2.0 Device**
0xFFFE_0000	4KB	**APB ブリッジ**
0xFFFE_1000	4KB	Reserved
0xFFFE_2000	4KB	Reserved
0xFFFE_3000	4KB	**DMA1C**
0xFFFE_4000	2KB	CF attribute
0xFFFE_4800	2KB	CF common
0xFFFE_5000	2KB	CF I/O
0xFFFE_5800	1KB	CF ture IDE CS1#
0xFFFE_5C00	1KB	CF ture IDE CS2#
0xFFFE_6000	4KB	CF　制御
0xFFFE_7000	4KB	**ARS**
0xFFFE_8000	4KB	**Camera I/F-1**
0xFFFE_9000	4KB	**JPEG Resize-1**
0xFFFE_A000	4KB	**JPEG module/FIFO-1**
0xFFFE_B000	4KB	**JPEG Codec-1**
0xFFFE_C000	4KB	**JPEG-DMAC**
0xFFFE_D000	4KB	**I2C**
0xFFFE_E000	4KB	**I2S**
0xFFFE_F000	4KB	(INT-CNTL)
0xFFFF_0000	4KB	Reserved
0xFFFF_1000	4KB	**GPIO**
0xFFFF_2000	4KB	**SPI**
0xFFFF_3000	4KB	Reserved
0xFFFF_4000	4KB	Reserved
0xFFFF_5000	4KB	**UART1**
0xFFFF_6000	1KB	Reserved
0xFFFF_7000	4KB	Reserved
0xFFFF_8000	4KB	**RTC**
0xFFFF_9000	4KB	**DMA2C**
0xFFFF_A000	4KB	**Memory CNTL**
0xFFFF_B000	4KB	**Timer A**
0xFFFF_C000	4KB	**WDT**
0xFFFF_D000	4KB	**SYS-CNTL**
0xFFFF_E000	4KB	Reserved
0xFFFF_F000	4KB	**INT-CNTL**

13. システムコントローラ (SYS)

13.1 概要

このブロックは主にクロックコントロール、ローパワー化手段、およびチップ・システム全体にかかわる事柄－メモリマップなどを制御するブロックです。

主な特徴は、

－HALT によるローパワーモード（IDLE）への移行サポート
－HALT 中の CPU、Bus クロックの ON/OFF の選択可能
－CPU/AHB/APB クロックのダイナミックな周波数切り替え
－内蔵 I/O 毎のクロック ON/OFF
－32KHz 動作モードのサポート
－PLL クロック切り替え手段のサポート
－ソフトウェアリセットのサポート
－Programmable Clock generator for UART

13.2 動作モード

S2S65A00 はロースピードモード、ロースピード HALT モード、ハイスピードモード、ハイスピード HALT モードの４つの動作状態をもっています。

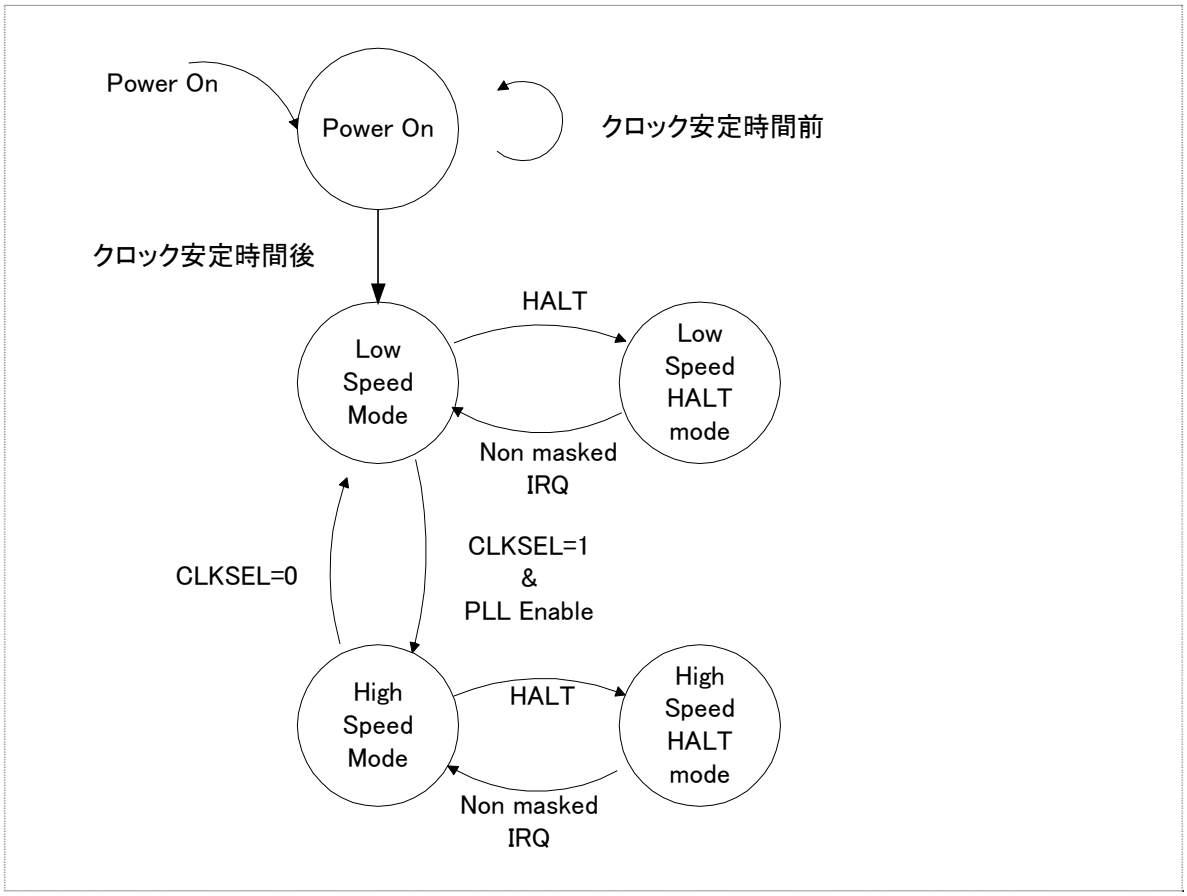


図 13.1　System の内部動作ステート

注意：　Low Speed Mode から High Speed Mode に移る際には、ソフトウエアにより PLL 安定時間（Max 100ms）を待った後、移行することができます。

以下、各動作状態（モード）について説明します。

13.2.1 パワーオンステート

このモードは特別な動作モードではありませんが、パワーオン時のリセット信号によりこの状態に入ります。クロック入力には 32KHz を使用します。外部の 32KHz を想定した場合、立ち上がり時間を Max 3 秒ほど確保する時間が必要であり、その間この状態にとどまります。リセット解除後、約 3 秒後に次の動作モードであるロースピードモードへ遷移します。

13.2.2 ロースピードモード

リセット直後のこのモードでは、PLL は停止しています。32KHz による動作のみが可能です。パワーオンリセット直後は、この状態から CPU がスタートすることになります。

PLL の設定をこのモードで行うことにより、(UART やタイマでの基本クロックに制約があるものの)任意の逓倍率を設定することが可能です。PLL の動作は PLL の安定に時間がかかるため(Max:100ms)、ソフトウエアでこの時間を待った後、設定した周波数にソフトウエアにより切り替えること（即ちハイスピードモード）になります。ハイスピードモード移行時に PLL の設定を変更したい場合は、必ずこのロースピードモードに遷移した後で、変更を行う必要があります。

13.2.3 ロースピード HALT モード

このモードは、ロースピードから HALT 動作を行うコマンドを発行することにより遷移することができ、もっとも消費電力の小さいモードとなります。内蔵 I/O バスや内蔵 I/O へのクロックの供給は、設定レジスタの値に依存しますが、すべてとめてしまうことも可能です。このモードからの復帰（常にロースピードモードへ復帰）にはマスクされていない割り込みによります。例としては、割り込みイネーブルされた GPIO 入力の変化、外部割込み端子、もしくは 32KHz で動作しているタイマからの割り込みなどによってこのモードを抜け出すことができます。

13.2.4 ハイスピードモード

このモードは、ロースピードモードで PLL を起動した後、ソフトウエアにより移行され、PLL に設定された逓倍率の基本クロックが CPU や内部のバスに供給され、動作します。

13.2.5 ハイスピード HALT モード

ハイスピードモードの状態で、HALT 動作を行うコマンドを発行することにより、このモードへ遷移します。このモードでは、このモードへ入る前に設定された各 I/O バスへのクロックストップ設定、CPU へのクロックストップ設定が同時に有効になります。したがってローパワーを実現するためには、このモードの多用、即ち CPU が動作する必要のない場合には、HALT を頻繁に発行することにより、実現できます。このモードはマスクされていない割り込みにより、ハイスピードモードへ戻ることができます。例としてタイマ割り込みや UART 受信割り込みをイネーブルにしておけば、割り込みが発生し、その後すぐにハイスピードモードに移ることが可能です。

13.3 外部端子

システムコントローラ関連の外部端子は以下の通りです。

端子名	入出力	端子機能	マルチプレクス端子／備考
RESET#	I	ハードウェアリセット入力	なし
SYSCLKI	I/O	32KHz 水晶発振器クロック入力または クロック出力(水晶振動子選択時)	SYSCKSEL に依存
SYSCKSEL	I	32KHz クロック入力選択信号 Low : 水晶振動子選択 High : 水晶発信器選択	なし

13.4 レジスタ

13.4.1 レジスター覧

以下に、システムコントロールの制御レジスタを示します。これらのレジスタのベースアドレスは、0xFFFF_D000 です。

表 13.1 レジスター覧（**ベースアドレス：0xFFFF_D000**）

Address Offset	Register Name	Abbreviation Name	Initial Value	R/W	Data Access Size
0x00	Chip ID Register	CHIPID	0x065A_000X	RO	32bit
0x04	Chip Configuration Register	CHIPCFG	0x0000_XXXX	RO	16/32bit
0x08	PLL Setting Register 1	PLLSET1	0x0421_84AE	R/W	32bit
0x0C	PLL Setting Register 2	PLLSET2	0x0000_0000	(R/W)	16/32bit
0x10	HALT Mode Clock Control Register	HALTMODE	0x0000_0000	R/W	16/32bit
0x14	IO Clock Control Register	IOCLKCTL	0x0000_0000	R/W	16/32bit
0x18	Clock Select Register	CLK_SEL	0x0000_0000	R/W	16/32bit
0x1C	HALT Control Register	HALTCTL	-	WO	16/32bit
0x20	Memory Remap Register	REMAP	0x0000_0000	R/W	16/32bit
0x24	Software Reset Register	SOFTRST	-	WO	32bit
0x28	UART1 Clock Divider Register	UART1DIV	0x0000_0000	R/W	16/32bit
0x2C	UART2 Clock Divider Register	UART2DIV	0x0000_0000	R/W	16/32bit
0x30	UART3 Clock Divider Register	UART3DIV	0x0000_0000	R/W	16/32bit
0x34	Timer-B Clock Select Register	TIMBCKSEL	0x0000_0000	R/W	16/32bit
0x40	MD Bus Pull-down Control Register	MDPLDCTL	0x0000_0000	R/W	16/32bit
0x44	SDD Bus Pull-down Control Register	SDDPLDCTL	0x0000_0000	R/W	16/32bit
0x48	GPIOE Resistor Control Register	PORTERCTL	0x0000_0000	R/W	16/32bit
0x4C	GPIOF Resistor Control Register	PORTFRCTL	0x0000_0000	R/W	16/32bit
0x50	GPIOG Resistor Control Register	PORTGRCTL	0x0000_0000	R/W	16/32bit
0x54	GPIOH Resistor Control Register	PORTHRCTL	0x0000_0000	R/W	16/32bit
0x58	GPIOI Resistor Control Register	PORTIRCTL	0x0000_0000	R/W	16/32bit
0x5C	GPIOJ Resistor Control Register	PORTJRCTL	0x0000_0000	R/W	16/32bit
0x60	GPIOK Resistor Control Register	PORTKRCTL	0x0000_0000	R/W	16/32bit
0x64	Internal TEST Mode Register	ITESTM	0x0000_0000	R/W	32bit
0x68	Embedded Memory Control Register	EMBMEMCTL	0x0000_0000	R/W	32bit
0x6C	Misc Register	MISC	0x0000_0000	R/W	32bit

13.4.2 レジスタ詳細説明

Chip ID Register (CHIPID)

SYS[0x00] 初期値 = 0x065A_000X — Read Only

PRODUCT ID [23:16]							
31	30	29	28	27	26	25	24
PRODUCT ID [15:8]							
23	22	21	20	19	18	17	16
PRODUCT ID [7:0]							
15	14	13	12	11	10	9	8
Reserved					REVISION CODE		
7	6	5	4	3	2	1	0

Bits[31:8]： **Product ID Code [23:0]**
このチップでは 16 進表記で 065A00h が埋め込まれています。

Bits[7:3]： **予約**

Bits[2:0]： **Revision Code [2:0]**
この IC のレビジョンを示します。最初のチップが REV1 で 01h になります。以下、版を変更するごとに 1 ずつ増えてゆきます。

Chip Configuration Register (CHIPCFG)

SYS[0x04] 初期値 = 0x0000_XXXX — Read Only

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
CONF [15:8]							
15	14	13	12	11	10	9	8
CONF [7:0]							
7	6	5	4	3	2	1	0

Bits[15:0]： **CONF [15:0]**
CNF[15:0]は MD[15:0]データバスにつながったプルアップ／プルダウン抵抗の状態を RESET#の立ち上がりでサンプルし、内部の動作を決めるために利用されます。
詳細な機能は 4.1 システムコンフィギュレーションの項を参照してください。

PLL Setting Register 1 (PLLSET1)
SYS[0x08] 初期値 = 0x0421_D46A Read/Write

n/a	CS [1:0]		CP [4:0]				
31	30	29	28	27	26	25	24
RS [3:0]				VC[3:0]			
23	22	21	20	19	18	17	16
N-Counter [3:0]				W-Divider [1:0]]		L-Counter [9:8]	
15	14	13	12	11	10	9	8
L-Counter [7:0]							
7	6	5	4	3	2	1	0

このレジスタは内蔵 PLL の動作周波数を決定するレジスタです。設定値を変更する場合には、必ず PLL Setting Register 2 (SYS[0x0C]) の Bit 1 (PLLEN) に“0”を設定し、PLL をパワーダウン状態にしてから行ってください。具体的な推奨設定値については Appendix A, B を参照願います。

Bits [30:29] : **CS [1:0]**
内蔵 PLL の LPF の容量値を調整します。 推奨値にしたがってください。

Bits [28:24]: **CP [4:0]**
内蔵 PLL の CP の電流値を調整します。推奨値にしたがってください。

Bits [23:20]: **RS [3:0]**
内蔵 PLL の LPF の抵抗値を調整します。 推奨値にしたがってください。

Bits [19:16] : **VC[3:0]**
内蔵 PLL の VCO 動作のためのパラメータ指定です。推奨値にしたがってください。

Bits [15:12] : **NN value N-Counter [3:0]**
この N-Counter は下記の L-Counter と共に、PLL の逓倍率を決定するために使用されます。詳細は LL Value の説明、および PLL の設定例を参照してください。

Bits [11:10] : **V-Divider [1:0]**
内蔵 PLL の内部で PLL-Out を分周するときの値を指定します。
00：使用禁止
01：1/2 分周
10：1/4 分周
11：1/8 分周
推奨値にしたがってください。

Bits [9:0] : **LL Value L-Counter [9:0]**
これらの値は内蔵 PLL の逓倍率を決定します。推奨値にしたがってください。
PLL Output = (N-Counter+1) x (L-Counter+1) x CLKI_L
= NN x LL x CLKI_L
NN = N-Counter, LL = L-Counter, CLKI_L = 外部クロック入力(32.768KHz)

PLL Setting Register 2 (PLLSET2)
SYS[0x0C] 初期値 = 0x0000_0000 (Read/Write)

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a							PLLEN
7	6	5	4	3	2	1	0

Bits [7:4]： **予約**

Bit 0： **PLLEN**
内蔵 PLL の Enable/Disable を指定します。PLLEN = 0 のときにのみ、PLL Setting Register1 の値を変更できます。
0： PLL は Disable になります。（PLL はパワーダウン状態を保ちます。）
1： PLL は Enable になります。

HALT Mode Clock Control Register (HALTMODE)
SYS[0x10] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved (0)							
23	22	21	20	19	18	17	16
Reserved (0)							
15	14	13	12	11	10	9	8
Reserved (0)							
7	6	5	4	3	2	1	0
CPUCKSEL [1:0]		n/a		HALT_MDCLK [4:0]			

CPU や内部バス（AHB1、AHB2、APB Bus）のクロック周波数および HALT モードの間にどのクロックを停止するかどうかを決めるレジスタです。

Bits [31:8]： **予約（0）**

Bits [7:6]： **CPUCKSEL [1:0]**
CPU/AHB1/AHB2/APB 用クロック分周の指定をします。
CPU クロック＝AHB1 クロック＝AHB2 クロック＝APB クロックとなります。すなわち、CPU クロックを分周した場合でも AHB1 クロック、AHB2 クロック、APB クロックは全て同じ周波数になります。このクロック変更指定は、書き込むと即座にグリッジ無しに変更後の周波数に反映されます。
00：PLL 出力の 1/1 が CPU に供給されます。
01：PLL 出力の 1/2 が CPU に供給されます。
10：PLL 出力の 1/4 が CPU に供給されます。
11：PLL 出力の 1/8 が CPU に供給されます。

Bits [3:0]： **HALT_MDCLK [3:0]**
本 Bits [4:0]は、該当 Bit に“0”を書くことにより HALT モードの間、指定された内部 Bus（CPU、AHB1-Bus、AHB2-Bus、APB-bus のいずれか、またはすべて）のクロック供給を停止します。HALT モードからの復帰はマスクされていない割り込みにより可能です。そのため割り込みを起こす必要のある内部デバイス（UART、Timer、Ethernet、GPIO など）へはクロックを供給しておく必要があります。
0： クロックは供給され続けます (Clock On)
1： クロックは停止状態になります (Clock Off)
Bit3：ARM720T 用クロックの On/Off を指定します。
Bit2：AHB1 用クロックの On/Off を指定します。*
Bit1：AHB2 用クロックの On/Off を指定します。
Bit0：APB 用クロックの On/Off を指定します。

Bits [3:0]のクロック On/Off 指定は、“1”を書いてもすぐにクロックが停止するわけではなく、HALT Control Register を介して HALT モードに移行した間のみ有効になります。 例えば、実行するべき Job が無い場合、または何かの割り込みイベントを待っている場合にこの機能を利用して、何もしないで CPU が待っている間の低電力化が図れます。

＊使用上の制限： AHB1 バスクロックを HALT で停止しようとするときには、必ず CPUCLK も HALT で停止する指定をしてください。 CPUCLK の停止については、AHB1 バスクロックと独立に制限なく HALT による停止指定ができます。

IO Clock Control Register (IOCLKCTL)

SYS[0x14] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
Reserved (0)	IOCLKCTL [14:8]						
15	14	13	12	11	10	9	8
IOCLKCTL [7:0]							
7	6	5	4	3	2	1	0

このレジスタの各 Bit は対応する I/O（Timer, UART, SPI. . . など）へクロックを供給するか供給を停止するかを指定します。“1”を書くことによりクロックが供給され、“0”を書くことによりクロック供給が停止します。使用しない I/O 機能に対して“0”を書くことにより低消費電力化が図れます。

Bit 15： **予約（0）**

Bits [6:0]： **IOCLKCTL [14:0]**

Bit14（ADC_CLKEN） ：ADC 用クロックの On/Off を指定します。
Bit13（UART1_CLKEN） ：UART1 用クロックの On/Off を指定します。
Bit12（UART2_CLKEN） ：UART2 用クロックの On/Off を指定します。
Bit11（UART3_CLKEN） ：UART3 用クロックの On/Off を指定します。
Bit10（DMA1C_CLKEN） ：DMA1C 用クロックの On/Off を指定します。
Bit9（DMA2C_CLKEN） ：DMA2C 用クロックの On/Off を指定します。
Bit8（SPI_CLKEN） ：SPI 用クロックの On/Off を指定します。
Bit7（I2C_CLKEN） ：I2C 用クロックの On/Off を指定します。
Bit6（I2S_CLKEN） ：I2S 用クロックの On/Off を指定します。
Bit5（TIMERA_CLKEN） ：Timer Ch0/1/2 用クロックの On/Off を指定します。
Bit4（TIMERB_CLKEN） ：Timer Ch0/1/2 用クロックの On/Off を指定します。
Bit3（CF_CLKEN） ：CF Card I/F 用クロックの On/Off を指定します。
Bit2（SD-MEM） ：SD-Memory I/F 用クロックの On/Off を指定します。
Bit1（USB_CLKEN） ：USB I/F 用クロックの On/Off を指定します。
Bit0（Reserved） ：予約

Clock Select Register (CLK_SEL)

SYS[0x18] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a		Reserved		N/a			CLKSEL
7	6	5	4	3	2	1	0

Bit [5:4]： **Reserved[1:0]**

Bit 0： **CLKSEL**

32KHz または PLL 出力のどちらをシステムクロックとして使用するかを決めます。“PLL 出力を使用”に指定する場合には、PLL の諸パラメータをあらかじめ設定しておき、更に PLL をイネーブルにし、適切な PLL 安定時間 (100ms) を保持した後、この PLL 出力変更 Bit を 1 に設定する必要があります。

0： 32KHz
1： PLL 出力

HALT Control Register (HALTCTL)

SYS[0x1C] 初期値 = - Write Only

31	30	29	28	27	26	25	24
Halt Control [31:24]							
23	22	21	20	19	18	17	16
Halt Control [23:16]							
15	14	13	12	11	10	9	8
Halt Control [15:8]							
7	6	5	4	3	2	1	0
Halt Control [7:0]							

Bits [31:0] : **Halt Control [31:0]**

このレジスタに任意の値を書くことにより、チップを HALT モードへ移行させることができます。

Memory Remap Register (REMAP)

SYS[0x20] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
n/a						REMAP2	REMAP1

リセット後のメモリマップを変更できます。Remap 機能により、リセット後に 0x0 番地に見えていた空間に SDRAM の空間がアサインされます。 詳細はシステムコントローラの Appendix を参照してください。 通常は使用する必要はありません。 RAM ベースで走行する OS によっては使い勝手がよくなります。

Bit1 : **REMAP2**

AHB2 Bus 側のメモリマップを変更できます。

0 : リセット後のメモリマップになります。

1 : リセット後のメモリマップを変更します。

Bit0 : **REMAP1**

AHB1 Bus 側のメモリマップを変更できます。

0 : リセット後のメモリマップになります。

1 : リセット後のメモリマップを変更します。

注意：メモリマップ変更の操作は、メモリマップの影響を受けない領域で変更コードを実行するなどの注意が必要です。
また、AHB1/AHB2 間でのメモリマップの矛盾を避けるため、REMAP1 と REMAP2 は同時にセットすることを推奨いたします。

Software Reset Register (SOFTRST)

SYS[0x24] 初期値 = - Write Only

Software Reset [31:24]							
31	30	29	28	27	26	25	24
Software Reset [23:16]							
23	22	21	20	19	18	17	16
Software Reset [15:8]							
15	14	13	12	11	10	9	8
Software Reset [7:0]							
7	6	5	4	3	2	1	0

Bits [31:0] : **Software Reset [31:0]**

このレジスタに AA5555AAh を書くことにより本 IC（S2S65A00）内のすべてのレジスタが初期化され、CPU はリセットされます。

UART1/2 Clock Divider Register (UART1/2DIV)

SYS[0x28/2C] 初期値 = 0x0000_0000 Read/Write

N/a							
31	30	29	28	27	26	25	24
N/a							
23	22	21	20	19	18	17	16
N/a							
15	14	13	12	11	10	9	8
UART 1/2 CLKDIV [7:0]							
7	6	5	4	3	2	1	0

このレジスタは UART1/2 用のボーレート TIME ベース用分周器として使用されます。本 IC では、基本的に PCLK クロックが、この分周比×1/2 分周によって分周されて UART の SCLK（＝転送レート生成クロック）に供給されます。

Bits [7:0] : **UART1/2CLKDIV [7:0]**

分周比 1/N を生成します。 （N = Bits[7:0]＋１）

0： 1/1

1： 1/2 分周

・

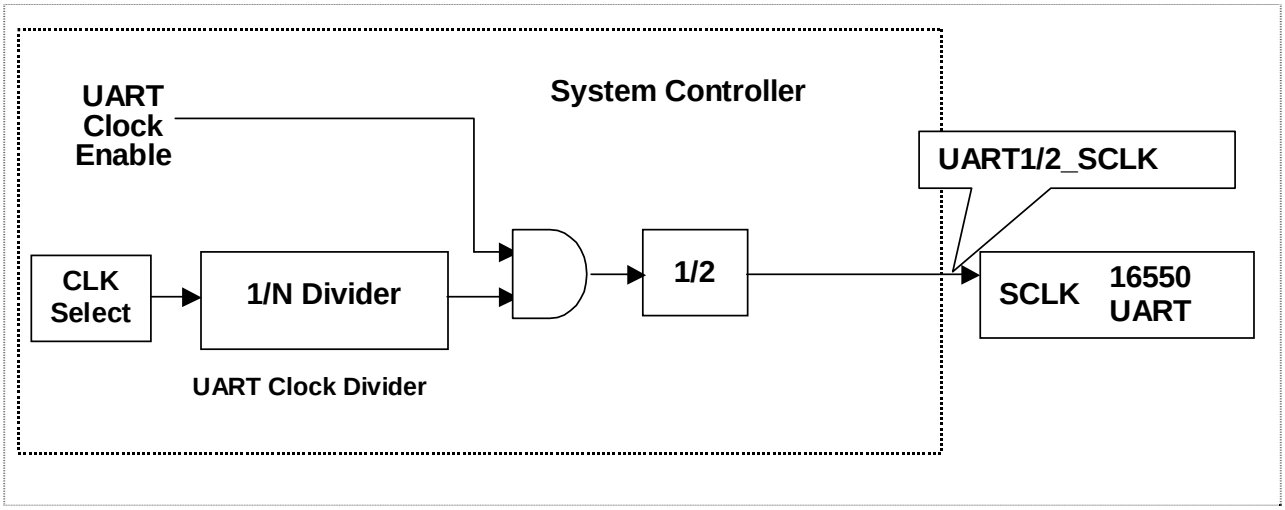
・

255：1/256 分周

UART1/2 への UART1/2_SCLK は、UART1/2CLKDIV の分周後、さらに 1/2 分周されたクロックで供給されます。

即ち、UART1/2_SCLK = (PCLK の周波数) * 1/N * 1/2 となります。（下図参照。）

注意：この UART1/2_SCLK は SPI I/F 用クロックの SCLK とは全く異なります。



UART3 Clock Divider Register (UART3DIV)

SYS[0x30] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
N/a							
23	22	21	20	19	18	17	16
N/a							
15	14	13	12	11	10	9	8
N/a							UART3_CKSEL
7	6	5	4	3	2	1	0
UART 3 CLKDIV [7:0]							

このレジスタは UART3 用のボーレート TIME ベース用分周器として使用されます。本 IC では、基本的に PCLK クロックが、この分周比×1/2 分周によって分周されて UART の SCLK（＝転送レート生成クロック）に供給されます。

Bits8： **UART3CKSEL**

UART3 のみ外部入力のクロックを選択可能とします。

0： 内部クロックを使用（分周比については Bit[7:0]を参照）

1： 外部クロックを使用

Bits [7:0]： **UART3CLKDIV [7:0]**

分周比 1/N を生成します。 （N = Bits[7:0]＋１）

0： 1/1

1： 1/2 分周

・

・

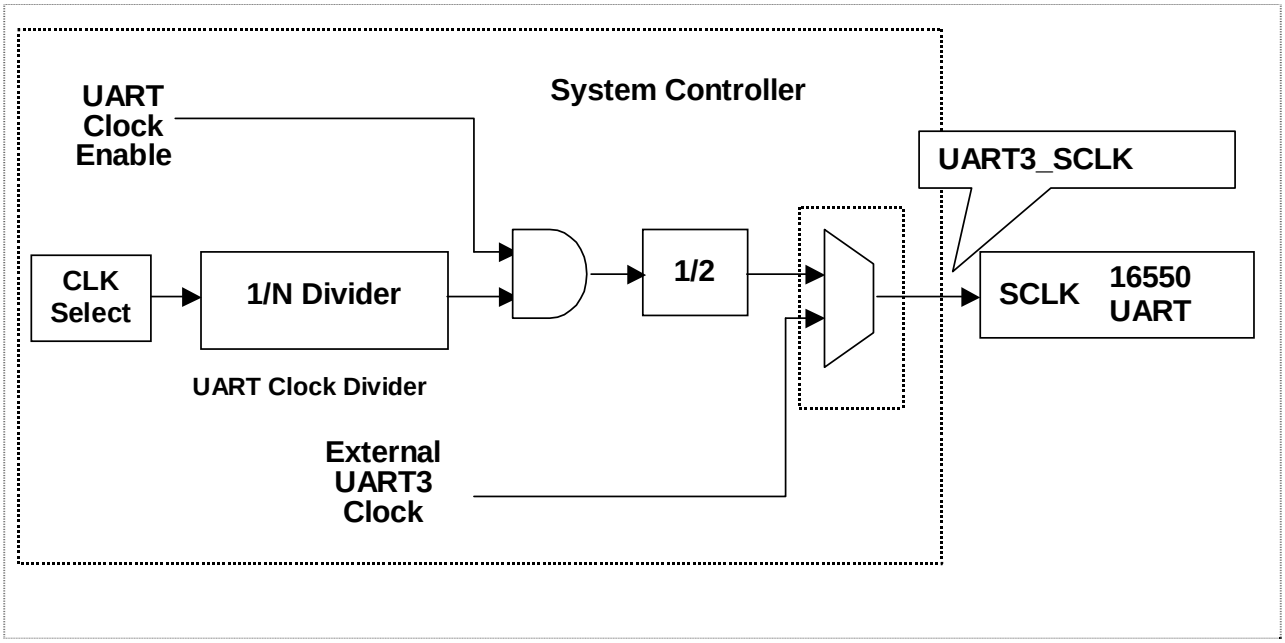
255：1/256 分周

UART3 へ内部クロックを選択した場合、UART3_SCLK は、UART3CLKDIV の分周後、さらに 1/2 分周されたクロックを供給します。

即ち、UART3_SCLK = (PCLK の周波数) * 1/N * 1/2 となります。

外部クロックを選択した場合は、分周は行わず、入力されるクロックの周波数がそのまま供給されます。（下図参照。）

注意：この UART3_SCLK は SPI I/F 用クロックの SCLK とは全く異なります。



Timer B Clock Select Register (TIMBCKSEL)

SYS[0x34] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
N/a							CLKSEL

Bit 0： **CLKSEL**

このレジスタは Timer-B に供給するクロックを選択します。

0： PLL 出力の 1/8

1： 32KHz

MD Bus Pulldown Control Register (MDPLDCTL)

SYS[0x40] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
MDPLDNDIS [15:8]							
7	6	5	4	3	2	1	0
MDPLDNDIS [7:0]							

Bits [15:0]： **MDPLDNDIS [15:0]**

MD[15:0] Bus に内蔵されているプルダウン抵抗の接続／切り離しを制御します。リセット後、必要に応じて（特にローパワーアプリケーション時には外部でのプルアップ抵抗の有無に応じて）切り離すことが可能です。

各ビットはそれぞれ MD[15:0]の各端子に対応しています。

0： プルダウン抵抗イネーブル （リセット後はこの状態です。）

1： プルダウン抵抗ディセーブル

SDD Bus Pulldown Control Register (SDDPLDCTL)

SYS[0x44] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
SDDPLDNDIS [15:8]							
7	6	5	4	3	2	1	0
SDDPLDNDIS [7:0]							

Bits [15:0]： **SDDPLDNDIS [15:0]**

SDD[15:0] Bus に内蔵されているプルダウン抵抗の接続／切り離しを制御します。リセット後、必要に応じて（特にローパワーアプリケーション時には外部でのプルアップ抵抗の有無に応じて）切り離すことが可能です。

各ビットはそれぞれ SDD[15:0]の各端子に対応しています。

0： プルダウン抵抗イネーブル （リセット後はこの状態です。）

1： プルダウン抵抗ディセーブル

GPIOE Resistor Control Register (PORTERCTL)

SYS[0x48] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
PORTEPDDIS [7:0]							

Bits [7:0]： **PORTEPDDIS [7:0]**

このレジスタは GPIOE[7:0]端子内蔵のプルダウン抵抗の接続／切り離しを制御します。

各ビットはそれぞれ GPIOE[7:0]の各端子に対応しています。

0： プルダウン抵抗イネーブル（リセット後はこの状態です。）

1： プルダウン抵抗ディセーブル

GPIOF Resistor Control Register (PORTFRCTL)

SYS[0x4C] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
PORTFPDDIS [7:0]							

Bits [7:0]： **PORTFPDDIS [7:0]**

これらのビットは GPIOF[7:0]端子内蔵のプルダウン抵抗の接続／切り離しを制御します。

各ビットはそれぞれ GPIOF[7:0]の各端子に対応しています。

0： プルダウン抵抗イネーブル（リセット後はこの状態です。）

1： プルダウン抵抗ディセーブル

GPIOG Resistor Control Register (PORTGRCTL)

SYS[0x50] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
PORTGPDDIS [7:0]							

Bits [7:0] : **PORTGPDDIS [7:0]**

このレジスタは GPIOG[7:0]端子内蔵のプルダウン抵抗の接続／切り離しを制御します。
各ビットはそれぞれ GPIOG[7:0]の各端子に対応しています。
0： プルダウン抵抗イネーブル（リセット後はこの状態です。）
1： プルダウン抵抗ディセーブル

GPIOH Resistor Control Register (PORTHRCTL)

SYS[0x54] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
PORTHPUDIS [7:0]							

Bits [7:0] : **PORTHPUDIS [7:0]**

このレジスタは GPIOH[7:0]端子内蔵のプルアップ抵抗の接続／切り離しを制御します。
各ビットはそれぞれ GPIOH[7:0]の各端子に対応しています。
0： プルアップ抵抗イネーブル（リセット後はこの状態です。）
1： プルアップ抵抗ディセーブル

GPIOI Resistor Control Register (PORTIRCTL)

SYS[0x58] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved						PORTIPUDIS [1:0]	

Bits [1:0] : **PORTIPUDIS [1:0]**

このレジスタは GPIOI[1:0]端子内蔵のプルアップ抵抗の接続／切り離しを制御します。
各ビットはそれぞれ GPIOI[1:0]の各端子に対応しています。
0： プルアップ抵抗イネーブル（リセット後はこの状態です。）
1： プルアップ抵抗ディセーブル

GPIOJ Resistor Control Register (PORTJRCTL)

SYS[0x5C] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
PORTJPDDIS [7:0]							

Bits [7:0] : **PORTJPDDIS [7:0]**

このレジスタは GPIOJ[7:0]端子内蔵のプルダウン抵抗の接続／切り離しを制御します。
各ビットはそれぞれ GPIOJ[7:0]の各端子に対応しています。

0： プルダウン抵抗イネーブル（リセット後はこの状態です。）
1： プルダウン抵抗ディセーブル

GPIOK Resistor Control Register (PORTKRCTL)

SYS[0x60] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
PORTKPDDIS [7:0]							

Bits [7:0] : **PORTKPDDIS [7:0]**

このレジスタは GPIOK[7:0]端子内蔵のプルダウン抵抗の接続／切り離しを制御します。
各ビットはそれぞれ GPIOK[7:0]の各端子に対応しています。

0： プルダウン抵抗イネーブル（リセット後はこの状態です。）
1： プルダウン抵抗ディセーブル

Internal TEST Mode Register (ITESTM)

SYS[0x64] 初期値 = 0x0000_0000 Read/Write

31	30	29	28	27	26	25	24
23	22	21	20	19	18	17	16
15	14	13	12	11	10	9	8
7	6	5	4	3	2	1	0

この LSI の内部テストに使用されるレジスタです。ユーザは触ることはできません。
リセット時のまま、何もしないでください。

Embedded Memory Control Register (EMBMEMCTL)

SYS[0x68] 初期値 = 0x0000_0000 Read/Write

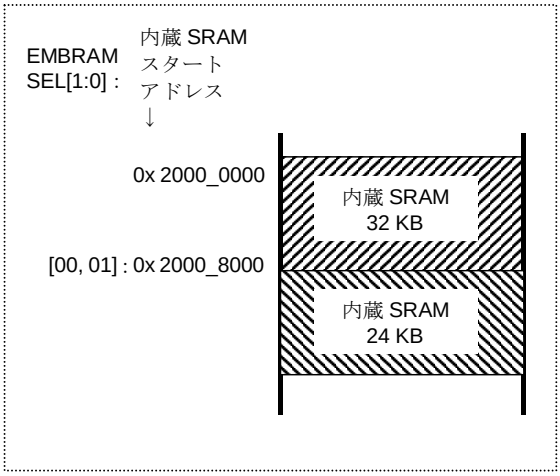
n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
Reserved		EMBRAMSEL[1:0]		Reserved		EMBWAITEN[1:0]	
7	6	5	4	3	2	1	0

Bits [31:6] : **予約（0）**

Bits [5:4] : **EMBRAMSEL[1:0] Embedded SRAM Select**

00 : JPEG1 ラインバッファとして、内蔵 SRAM の 24KB が割り当てられます。（JPEG コントローラ 1 の JPEG Line Buffer の[0xA4]をデフォルト値(0x0020)の設定で使用してください。）
内蔵 SRAM として 32KB が使用可能です。使用する場合のスタートアドレスは 0x 2000_0000 となります。

01 : JPEG1 ラインバッファとして 32KB、JPEG2 ラインバッファとして 24KB が割り当てられます。
（この場合、JPEG コントローラ 1 の JPEG Line Buffer の[0xA4]を 0x0000、JPEG コントローラ 2 の JPEG Line Buffer の[0xA4]をデフォルト値(0x0020)の設定で使用してください。）



10 : JPEG1 ラインバッファとして 56KB が割り当てられます。JPEG コントローラ 1 の JPEG Line Buffer の[0xA4]を 0x0000 の設定で使用してください。）

11 : 内蔵 SRAM として全てのサイズ（56KB）が使用可能です。使用する場合のスタートアドレスは 0x 2000_0000 となります。

Bits [3:2] : **予約（0）**

Bits [1:0] : **EMBWAITEN[1:0] Embedded SRAM Wait Control**

00 : ノーウエイト
01 : Read Access Wait ON （Read： 1 ウエイト， Write： ノーウエイト）
10 : Read Access Wait ON、Read Data Wait ON、
（Read： 2 ウエイト， Write： ノーウエイト）
11 : Read Access Wait ON、Read Data Wait ON、Write Access Wait On
（Read： 2 ウエイト， Write： 1 ウエイト）

MISC Register (MISC)

SYS[0x6C] 初期値 = 0x0000_0000 (Read/Write)

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a					Reserved	USB_CLK_CTL	Reserved
7	6	5	4	3	2	1	0

Bits [31:3] : **未使用**

Bit 2 : **予約**

Bit 1 : **USB_CLK_CTL**

USB HS-Device に供給するクロックの指定をします。USB HS-Device を使用するときは本ビットを“1”に設定してください。

0： USB HS-Device にクロックを供給しません。

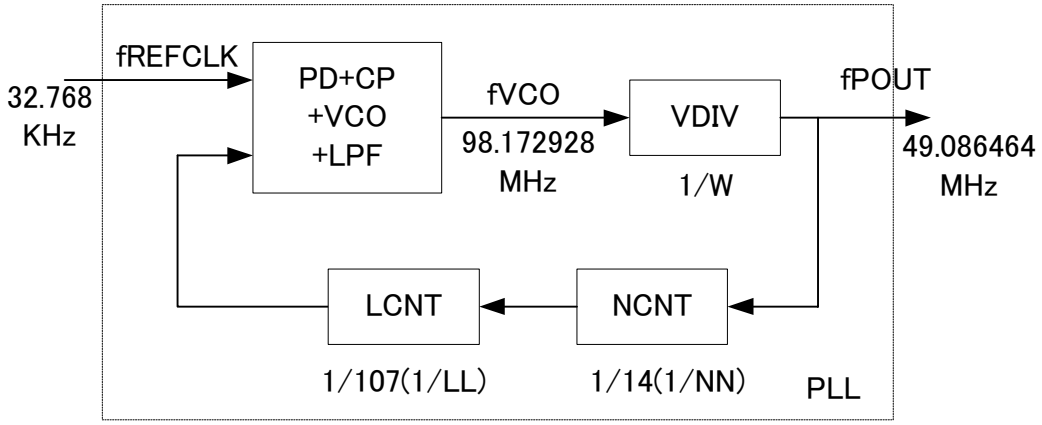
1： USB HS-Device に 60MHz のクロックを供給します。

Bit 0 : **予約**

注意）USB HS-Device を使用する場合は、0x0000_0002 の設定を行ってください。

13.5 Appendix A: PLL Setting Example

以下に PLL の設定例を示します。
CPU Clock = 49.086464 MHz の場合：



fPOUT＝ 49.086464 MH z を選択した場合：
fREFCLK = 32.768KHz
fPOUT / fREFCLK = 49.086464 MHz / 32.768KHz = 1498 （分周比）
分周比は 1498 = 2 x 7 x 207 = 107 x 14
したがって分周の値として、NN = 4, LL = 107 を選ぶことができる。
NN = 14、つまり N-Counter [3:0] = 13 = 1101 (Binary)
LL = 107、つまり L-Counter [9:0] = 106 = 00_0110_1010 (Binary)

VCO 周波数： fVCO = fPOUT × W = 49.086464 MHz × 2 = 98.172928 MHz
この場合 W = 2 で、約 100MHz であるので、
W = 2、つまり W-Divider [1:0] = 01 (Binary) でよい。

13.6 Appendix B: PLL Parameter table

PLL Seeting Register1 の設定は使用する周波数のレンジに応じて以下の値を使用してください。PLL の出力安定時間は Max100ms 必要ですので，設定後 PLL を ON にしても、実際に 32KHz から切り換える際には、必ずこの時間を待ったあと、CLKSEL=1 にして所望のクロックにスイッチしてください。なお PLL の周波数の変更は 32KHz で走行時で、かつ PLLEN=0 のとき（= PLL Disable）にのみ変更可能です。32KHz 以外の周波数で既に走行時に、別の周波数にダイレクトに移行することはできません。必ず 32KHz を経由してください。実際の PLL Setting Register1 の設定例を以下に示します。

ターゲット周波数例	逓倍率 from 32.768KHz	PLL_Setting Register1 (Hex)	NN N-Counter	LL L-Counter
49.086464MHz	1498 = 14 x 107	0x0421_D46A	1101b	00_0110_1010b

W-Divider、VC、RS、CP、CS の各値は、VCO 周波数が 90-100MHz である限り、以下の値をご使用ください。

VV_Divider	VC[3:0]	RS[3:0]	CP[4:0]	CS[1:0]
01b	0001b	0010b	0_0100b	00b

13.7 Appendix C: Remap 後のメモリマップ

13.7.1 Remap 後のメモリマップ (AHB1)

Remap 機能により、リセット後のメモリマップを変更することができます。
ROM ベースで走行する OS では必要ありませんが、RAM ベースで走行する OS にとっては Remap 後のメモリマップの方が使いやすい場合もあります。

表 13.2 Remap 後の AHB1 メモリマップ

Start Address	End Address	Size	Device	外部 Chip Select	Device Bus size
0x0000_0000	0x0FFF_FFFF	256MByte	外部 SDRAM	SDCS0#	32
0x1000_0000	0x1FFF_FFFF	256MByte	Reserved		
0x2000_0000	0x2FFF_FFFF	128MByte	内蔵 SRAM		32
0x3000_0000	0x3FFF_FFFF	256MByte	Reserved		
0x4000_0000	0x4FFF_FFFF	256MByte	外部 SDRAM	SDCS0#	32
0x5000_0000	0x5FFF_FFFF	256MByte	外部 SDRAM	SDCS1#	32
0x5000_0000	0x5FFF_FFFF	256Mbyte	Reserved		
0x6000_0000	0x6FFF_FFFF	256Mbyte	Reserved		
0x7000_0000	0x7FFF_FFFF	256Mbyte	Reserved		
0x8000_0000	0x8FFF_FFFF	256Mbyte	Reserved		
0x9000_0000	0x9FFF_FFFF	256Mbyte	Reserved		
0xA000_0000	0xAFFF_FFFF	256Mbyte	Reserved		
0xB000_0000	0xBFFF_FFFF	256Mbyte	Reserved		
0xC000_0000	0xC1FF_FFFF	32MByte	外部 ROM	MCS0#	16
0xC200_0000	0xC3FF_FFFF	32MByte	Reserved		
0xC400_0000	0xC5FF_FFFF	32MByte	外部 SRAM	MCS1#	16
0xC600_0000	0xC7FF_FFFF	32MByte	Reserved		
0xC800_0000	0xC9FF_FFFF	32MByte	外部 SRAM	MCS2#	16
0xCA00_0000	0xCBFF_FFFF	32MByte	Reserved		
0xCC00_0000	0xCDFF_FFFF	32MByte	外部 SRAM	MCS3#	16
0xCE00_0000	0xCFFF_FFFF	32MByte	Reserved		
0xD000_0000	0xDFFF_FFFF	256Mbyte	Reserved		
0xE000_0000	0xEFFF_FFFF	256Mbyte	USB DMA Port		
0xF000_0000	0xFFFF_FFFF	256Mbyte	内蔵 I/O エリア		32/16/8

表 13.3　Remap 前の AHB1 メモリマップ

Start Address	End Address	Size	Device	外部 Chip Select	Device Bus size
0x0000_0000	0x01FF_FFFF	32MByte	外部 ROM	MCS0#	16
0x0200_0000	0x03FF_FFFF	32MByte	Reserved		
0x0400_0000	0x05FF_FFFF	32MByte	外部 SRAM	MCS1#	16
0x0600_0000	0x07FF_FFFF	32MByte	Reserved		
0x0800_0000	0x09FF_FFFF	32MByte	外部 SRAM	MCS2#	16
0x0A00_0000	0x0BFF_FFFF	32MByte	Reserved		
0x0C00_0000	0x0DFF_FFFF	32MByte	外部 SRAM	MCS3#	16
0x0E00_0000	0x0FFF_FFFF	32MByte	Reserved		
0x1000_0000	0x1FFF_FFFF	256MByte	Reserved		
0x2000_0000	0x2FFF_FFFF	256Mbyte	内蔵 SRAM		32
0x3000_0000	0x3FFF_FFFF	256MByte	Reserved		
0x4000_0000	0x4FFF_FFFF	256MByte	外部 SDRAM	SDCS0#	16/32
0x5000_0000	0x5FFF_FFFF	256MByte	外部 SDRAM	SDCS1#	16/32
0x6000_0000	0x6FFF_FFFF	256Mbyte	Reserved		
0x7000_0000	0x7FFF_FFFF	256Mbyte	Reserved		
0x8000_0000	0x8FFF_FFFF	256Mbyte	Reserved		
0x9000_0000	0x9FFF_FFFF	256Mbyte	Reserved		
0xA000_0000	0xAFFF_FFFF	256Mbyte	Reserved		
0xB000_0000	0xBFFF_FFFF	256Mbyte	Reserved		
0xC000_0000	0xC1FF_FFFF	32MByte	外部 ROM	MCS0#	16
0xC200_0000	0xC3FF_FFFF	32MByte	Reserved		
0xC400_0000	0xC5FF_FFFF	32MByte	外部 SRAM	MCS1#	16
0xC600_0000	0xC7FF_FFFF	32MByte	Reserved		
0xC800_0000	0xC9FF_FFFF	32MByte	外部 SRAM	MCS2#	16
0xCA00_0000	0xCBFF_FFFF	32MByte	Reserved		
0xCC00_0000	0xCDFF_FFFF	32MByte	外部 SRAM	MCS3#	16
0xCE00_0000	0xCFFF_FFFF	32MByte	Reserved		
0xD000_0000	0xDFFF_FFFF	256Mbyte	Reserved		
0xE000_0000	0xEFFF_FFFF	256Mbyte	USB DMA Port		
0xF000_0000	0xFFFF_FFFF	256Mbyte	内蔵 I/O エリア		32/16/8

13.7.2 Remap 後のメモリマップ (AHB2)

表 13.4 remap 後のメモリマップ (AHB2)

Start Address	End Address	Size	Device	外部 Chip Select	Device Bus size
0x0000_0000	0x0FFF_FFFF	256MByte	外部 SDRAM	SDCS0#	32
0x1000_0000	0x1FFF_FFFF	256MByte	Reserved		
0x2000_0000	0x2FFF_FFFF	128MByte	内蔵 SRAM		32
0x3000_0000	0x3FFF_FFFF	256MByte	Reserved		
0x4000_0000	0x4FFF_FFFF	256MByte	外部 SDRAM	SDCS0#	32
0x5000_0000	0x5FFF_FFFF	256MByte	外部 SDRAM	SDCS1#	32
0x5000_0000	0x5FFF_FFFF	256Mbyte	Reserved		
0x6000_0000	0x6FFF_FFFF	256Mbyte	Reserved		
0x7000_0000	0x7FFF_FFFF	256Mbyte	Reserved		
0x8000_0000	0x8FFF_FFFF	256Mbyte	Reserved		
0x9000_0000	0x9FFF_FFFF	256Mbyte	Reserved		
0xA000_0000	0xAFFF_FFFF	256Mbyte	Reserved		
0xB000_0000	0xBFFF_FFFF	256Mbyte	Reserved		
0xC000_0000	0xC1FF_FFFF	32MByte	外部 ROM	MCS0#	16
0xC200_0000	0xC3FF_FFFF	32MByte	Reserved		
0xC400_0000	0xC5FF_FFFF	32MByte	外部 SRAM	MCS1#	16
0xC600_0000	0xC7FF_FFFF	32MByte	Reserved		
0xC800_0000	0xC9FF_FFFF	32MByte	外部 SRAM	MCS2#	16
0xCA00_0000	0xCBFF_FFFF	32MByte	Reserved		
0xCC00_0000	0xCDFF_FFFF	32MByte	外部 SRAM	MCS3#	16
0xCE00_0000	0xCFFF_FFFF	32MByte	Reserved		
0xD000_0000	0xDFFF_FFFF	256Mbyte	JPEG2 DMA Port		
0xE000_0000	0xEFFF_FFFF	256Mbyte	JPEG1 DMA Port		32
0xF000_0000	0xFFFF_FFFF	256Mbyte	Reserved		

表 13.5　Remap 前の AHB2 メモリマップ

Start Address	End Address	Size	Device	外部 Chip Select	Device Bus size
0x0000_0000	0x01FF_FFFF	32MByte	外部 ROM	MCS0#	16
0x0200_0000	0x03FF_FFFF	32MByte	Reserved		
0x0400_0000	0x05FF_FFFF	32MByte	外部 SRAM	MCS1#	16
0x0600_0000	0x07FF_FFFF	32MByte	Reserved		
0x0800_0000	0x09FF_FFFF	32MByte	外部 SRAM	MCS2#	16
0x0A00_0000	0x0BFF_FFFF	32MByte	Reserved		
0x0C00_0000	0x0DFF_FFFF	32MByte	外部 SRAM	MCS3#	16
0x0E00_0000	0x0FFF_FFFF	32MByte	Reserved		
0x1000_0000	0x1FFF_FFFF	256MByte	Reserved		
0x2000_0000	0x2FFF_FFFF	256Mbyte	内蔵 SRAM		32
0x3000_0000	0x3FFF_FFFF	256MByte	Reserved		
0x4000_0000	0x4FFF_FFFF	256MByte	外部 SDRAM	SDCS0#	16/32
0x5000_0000	0x5FFF_FFFF	256MByte	外部 SDRAM	SDCS1#	16/32
0x6000_0000	0x6FFF_FFFF	256Mbyte	Reserved		
0x7000_0000	0x7FFF_FFFF	256Mbyte	Reserved		
0x8000_0000	0x8FFF_FFFF	256Mbyte	Reserved		
0x9000_0000	0x9FFF_FFFF	256Mbyte	Reserved		
0xA000_0000	0xAFFF_FFFF	256Mbyte	Reserved		
0xB000_0000	0xBFFF_FFFF	256Mbyte	Reserved		
0xC000_0000	0xC1FF_FFFF	32MByte	外部 ROM	MCS0#	16
0xC200_0000	0xC3FF_FFFF	32MByte	Reserved		
0xC400_0000	0xC5FF_FFFF	32MByte	外部 SRAM	MCS1#	16
0xC600_0000	0xC7FF_FFFF	32MByte	Reserved		
0xC800_0000	0xC9FF_FFFF	32MByte	外部 SRAM	MCS2#	16
0xCA00_0000	0xCBFF_FFFF	32MByte	Reserved		
0xCC00_0000	0xCDFF_FFFF	32MByte	外部 SRAM	MCS3#	16
0xCE00_0000	0xCFFF_FFFF	32MByte	Reserved		
0xD000_0000	0xDFFF_FFFF	256Mbyte	JPEG2 DMA Port		
0xE000_0000	0xEFFF_FFFF	256Mbyte	JPEG1 DMA Port		32
0xF000_0000	0xFFFF_FFFF	256Mbyte	Reserved		

13.8 Appendix D：クロック・コントロールブロック図

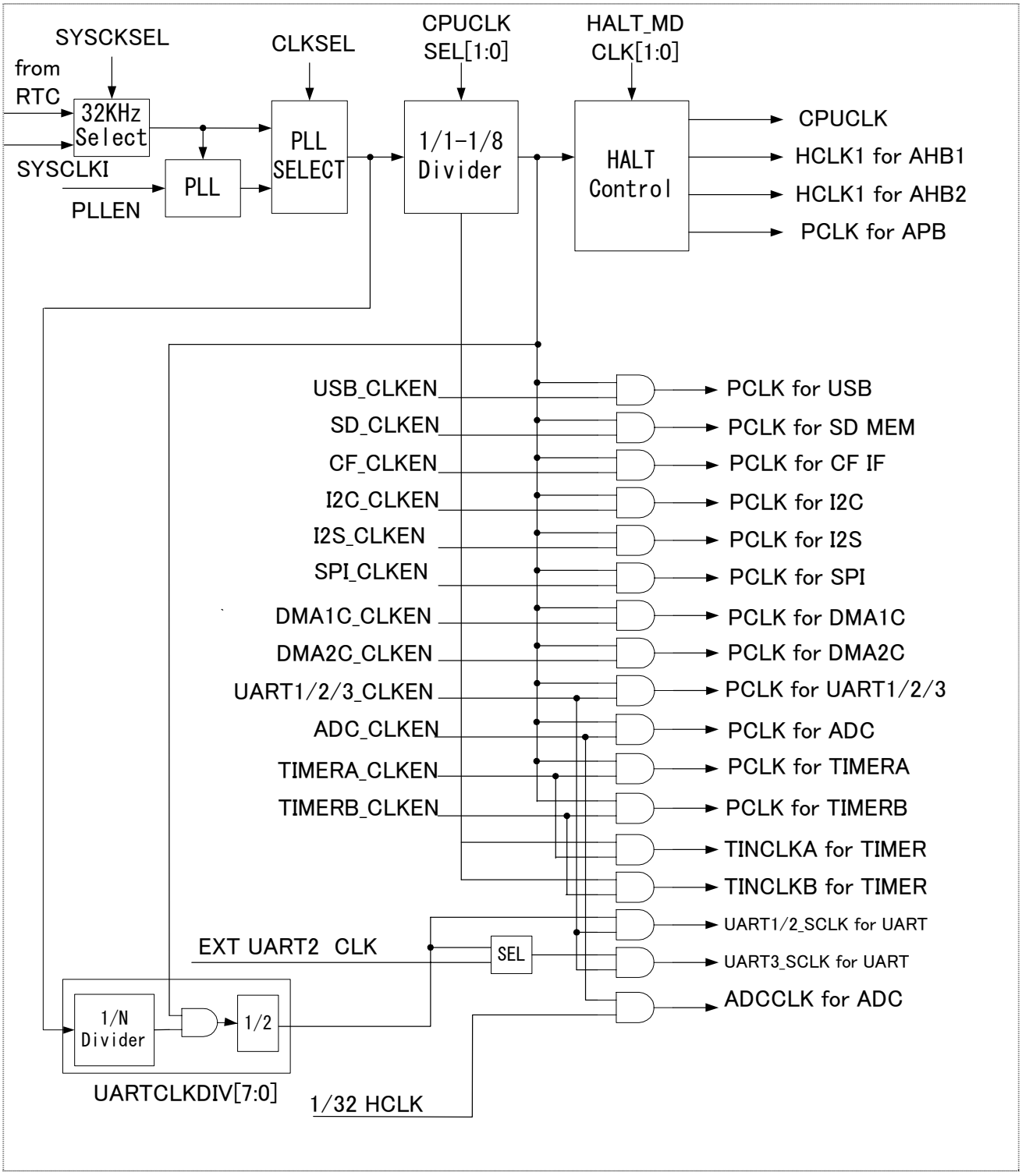


図 13.2　クロック・コントロールブロック図

図でクロック信号名“PCLK_for_ブロック名”は、各ブロックでのバス制御およびレジスタ制御用に使用されるクロックを意味しています。また UART1/2/3_SCLK はそれぞれ UART1/2/3 での送受信レート生成用のクロック入力、TINCLKA/B はそれぞれ TimerA/B ブロックでのカウント用基本クロック入力にそれぞれ使用されます。

13.9 Appendix E：UART Clock 設定例:

CPU Clock = 49.086464 MHz の場合：

Baudrate	Ideal x16 Clock (Hz)	SYS[0x28]	16550 Divisor Value (DEC)	16550 Divisor Value (HEX)	Percent Error %	Actual x16 Clock (Hz)	UART_SCLK
110	1760	0	13945	3679	0.00	1760.0	24543232
300	4800	0	5113	13F9	0.00	4800.2	24543232
600	9600	0	2557	09FD	0.02	9598.4	24543232
1200	19200	0	1278	04FE	0.02	19204.4	24543232
2400	38400	0	639	027F	0.02	38408.8	24543232
4800	76800	0	320	0140	0.13	76697.6	24543232
9600	153600	0	160	00A0	0.13	153395.2	24543232
14400	230400	0	107	006B	0.44	229376.0	24543232
19200	307200	0	80	0050	0.13	306790.4	24543232
38400	614400	0	40	0028	0.13	613580.8	24543232
57600	921600	0	27	001B	1.37	909008.6	24543232
115200	1843200	0	13	000D	2.43	1887940.9	24543232

14. メモリコントローラ (MEMC)

14.1 概要

メモリコントローラは非同期 SRAM タイプデバイスおよび SDRAM をコントロールする AHB バスインタフェースのメモリコントローラです。非同期 SRAM は最大 4 つまで、SDRAM は最大 2 つまでのデバイスをサポートすることができます。

メモリコントローラの特長を以下に示します。

- SRAM タイミングデバイスに対応します。
- SDRAM に対応します。
- SDRAM オートリフレッシュのリフレッシュ間隔をデバイスにあわせて調整することができます。メモリコントローラが他のメモリアクセス等で使用できない場合は、複数回のリフレッシュを貯めこみ、メモリコントローラが空いた時点でバーストリフレッシュをすることができます。リフレッシュはバックグラウンドで実行します。
- SDRAM のセルフリフレッシュをサポートします。またセルフリフレッシュに入っている SDRAM に読み書きを行うことができます。この場合、セルフリフレッシュから抜けたあと、アイドル状態のままでいるか、再びセルフリフレッシュモードに入るかを選択できます。

14.1.1 SRAM コントローラ

- SRAM コントローラは次のデバイスをサポートします。
  非同期 SRAM, ROM, FLASH, EEPROM
- 外部 16 ビットバス
- バースト転送をサポート
- wait state をプログラム可能
- WE#, OE#の挿入タイミングがプログラム可能。
- WE#共通、バイトイネーブルが独立の x16 タイプのスタティックメモリに外部ロジックなしで接続できます。

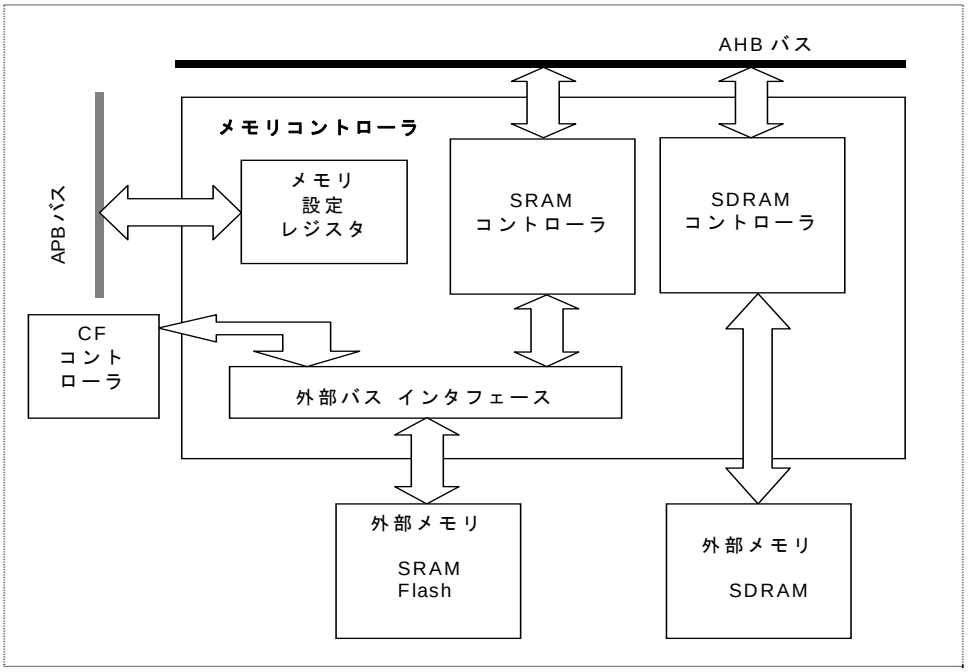
14.1.2 SDRAM コントローラ

- SDRAM コントローラは次のデバイスをサポートします。
  SDRAM
- 16/32 ビットバス
- バースト転送をサポート
- オートプリチャージの ON/OFF をプログラム可能
- セルフリフレッシュモードデバイスからの IN/OUT 自動設定モードサポート
- SDRAM イニシャライズ手動モードサポート

14.1.3 外部バスインタフェースモジュール

- SRAM コントローラと CF コントローラモジュールをアドレス/データバスを共通で使用したい場合に必要なモジュールです。外部メモリとのアクセスに対して、SRAM コントローラと CF コントローラモジュールの外部バスアクセス要求の調停を行います。バスの要求はハンドシェイク方式です。

14.2 ブロック図



14.3 外部端子

メモリコントローラ関連の外部端子は以下の通りです。

表 14.1 メモリコントローラ関連外部端子一覧

端子名	入出力	端子機能	マルチプレクス端子／備考
MA[23:20]	O	SRAM アドレス出力信号[23:20]	GPIOD[3:0]*
MA [19:12]	O	SRAM アドレス出力信号[19:12]	
MA11	O	SRAM アドレス出力信号 11	CFREG#**
MA [10:0]	O	SRAM アドレス出力信号[10:0]	CFADDR [10:0]**
MD [15:0]	I/O	SRAM データ入出力信号[15:0]	MODESEL[15:0]***
MCS [3:2]#	O	SRAM 用チップセレクト信号[3:2]	GPIOD[5:4]*
MCS [1:0]#	O	SRAM 用チップセレクト信号[1:0]	
MOE#	O	SRAM 出力用ストローブ信号	CFOE#**
MWE#	O	SRAM 用 Write イネーブル信号	CFWE#**
MBEH#	O	SRAM 用上位バイトイネーブル信号	
MBEL#	O	SRAM 用下位バイトイネーブル信号	
MWAIT	I	メモリコントローラ用ウェイト信号	CFWAIT#と端子を共有

注意(*)：これらのメモリコントローラ用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1” に設定することにより使用できます。
注意(**)：コンパクトフラッシュインタフェース(CF)動作時には、メモリコントローラ用端子が CF の外部端子として動作します。
注意(***)：パワーオンリセット時には、内部動作モードを決定するための MODESEL 端子として動作します。

表 14.2 メモリコントローラ関連外部端子一覧

端子名	入出力	端子機能	マルチプレクス端子／備考
SDA [14:13]	O	SDRAM バンクドレス出力信号[1:0]	
SDA [12:0]	O	SDRAM アドレス出力信号[12:0]	
SDD [31:16]	I/O	SDRAM データ入出力信号[31:16]	GPIOJ[7:0]*,GPIOK[7:0]*
SDD [15:0]	I/O	SDRAM データ入出力信号[15:0]	
SDCS[1:0]#	O	SDRAM 用チップセレクト信号[1:0]	
SDWE#	O	SDRAM 用 Write イネーブル信号	
SDCLK	O	SDRAM 用クロック出力信号	
SDCLKEN	O	SDRAM 用クロックイネーブル出力信号	
SDRAS#	O	SDRAM 用 RAS 信号	
SDCAS#	O	SDRAM 用 CAS 信号	
SDDQM3#	I/O	SDRAM 用最上位バイト用 DQM 信号	GPIOD7*
SDDQM2#	I/O	SDRAM 用中上位バイト用 DQM 信号	GPIOD6*
SDDQM1#	O	SDRAM 用中下位バイト用 DQM 信号	
SDDQM0#	O	SDRAM 用最下位バイト用 DQM 信号	

注意(*)：これらのメモリコントローラ用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1” に設定することにより使用できます。

14.4 Memory controller

14.4.1 デバイス数

最大 4 つまでの SRAM タイミングデバイス（デバイス 0-3）を接続できます。
SDRAM デバイスはチップセレクト信号を２本使用してデータ８ビット幅の SDRAM を最大８つ(Max:512MB)接続可能です。

14.4.2 外部メモリ幅

外部メモリのバス幅は、SRAM タイミングデバイスにおいて 16 ビット、SDRAM デバイスは 16 ビット幅、又は 32 ビット幅を設定により切り替え可能です。

14.4.3 デバイスのセグメント設定

SRAM インタフェースは最大 32MB の空間がチップセレクト信号毎に割り当てられています。SRAM のタイミング空間としては 128MB(32MBx4)利用可能です。また、SDRAM インタフェースは最大 256MB の空間がチップセレクト毎に割り当てられ、最大で 512MB（各 256MB） がアサインされます。

14.5 SRAM control

14.5.1 タイミング設定

各デバイスコントローラのタイミング設定は、メモリコントローラの供給されるクロック周期に係数をかけるという形で設定します。クロック周期にあわせて、最適な値に設定してください。
調整できる項目としては、読み出し、書き込みのタイミング、OE,WE 信号の挿入タイミングなどがあります。

14.5.2 ライトプロテクト

該当する制御レジスタの WPROTECT ビットを“1”に設定することで対象となるデバイスを書き込み禁止にすることができます。禁止された領域へ書き込み動作を行うとバスアクセスがアボートし、CPU へエラーが通知されますので、ソフトウエアで適切に処理して下さい。

14.6 SDRAM control

14.6.1 モードレジスタ設定

「SDRAM モードレジスタ」を設定した後、初期化制御レジスタにてデバイスに LMR 命令を発行します。通常は INIT_SD（MEMC[0x80]bit15）を使用してください。このとき命令を発行するターゲットのデバイスの選択を同時に行ってください。各デバイスコントローラのタイミング設定は、クロック数で設定します。クロック周期にあわせて、最適な値に設定してください。

14.6.2 バースト対応

SDRAM はバースト長 1,2,4,8 に対応します。また、アドレスはインクリメントバーストに限定されています。SDRAM では同一 ROW の中ではバースト長を越える場合でもアクセスのペナルティはありません。あるデバイスへの R/W 中に、異なるデバイスへのアクセス要求によって割込みが起きた場合、SDRAM コントローラは自動的にプリチャージまたはバーストターミネートコマンドを発行してバスの引き渡しを迅速に行います。

14.6.3 オートプリチャージ設定

読み出し、書き込み動作を完了後、プリチャージを発行するかどうかを設定します。「プリチャージしない」を選択すると、ROW 系が選択されたままになり、同じページ内のデータの読み出し、書き込みが早くなります。アクティブな ROW アドレスはオートリフレッシュ時にプリチャージが行われ、その後アイドルになります。また手動でプリチャージをすることもできます。 「SDRAM 設定レジスタ（MEMC[0x70]）」を参照してください。

14.6.4 低消費電力化

データのアクセスが完了してアイドル状態となっている（セルフリフレッシュ状態や ROW アクティブなままの状態を含む）デバイスに対して、MCLKEN をディアサートすることができます。SDRAM 設定レジスタ（MEMC[0x70]）の CKECTRL ビットを“1”に設定することで有効になります。

CKECTRL ビットを“1”に設定した場合に限り、更なる低消費電力化を目的として、メモリに供給するクロック自身を止めることもできます。CLKCTRL ビットを“1”に設定することで有効になります。

オートリフレッシュの蓄積値が設定値を超えた場合や､デバイスに対して読み書き要求がある場合、CKECTRL ビットが“1”であっても CKE を HIGH にアサートしてデータのやり取りが再開されます。

14.6.5 メモリクロック停止

CKECTRL にて MCLKEN を LOW に保つ以外に、SDRAM へのクロック供給を停止することで更なる低消費電力を実現することができます。このモードは CLKCTRL ビットを“1”にすることで有効になります。CLKCTRL ビットを有効にするときは、必ず CKECTRL ビットを有効にしてください。メモリコントローラは接続されている全ての SDRAM が次のいずれかの状態である場合にメモリクロック MCLK を停止します。

メモリ未初期化の状態、セルフリフレッシュモード中または ROW アクティブでないチップ IDLE 状態。

メモリクロック停止状態においても、メモリの読み書き要求やオートリフレッシュ要求がある場合はメモリクロックは再び供給を開始します。

なお例外的に、SDRAM 詳細設定レジスタ（MEMC[0x74]）中の CLKFORCE ビットを“1”にすると、SDRAM メモリの状態にかかわらず、MCLK の出力が継続されます。

14.6.6 省電力モードサポート

システムが省電力モードに入るときに、メモリコントローラ自体へのクロックが停止される場合があります。SDRAM 内のデータを保持するためには、オートリフレッシュかセルフリフレッシュが必要ですが、メモリコントローラのクロック供給がなくなるので、全ての SDRAM をセルフリフレッシュモードに入れる必要があります。
ソフトウェアによって全ての SDRAM をセルフリフレッシュモードに入れてから、システムコントローラを省電力モードに設定することで実現してください。

14.6.7 オートリフレッシュ制御

オートリフレッシュは少なくとも一つの SDRAM が初期化され、かつ IDLE（セルフリフレッシュは除く）状態にある場合に実行されます。周期は SDRAM リフレッシュタイマレジスタに設定された HCLK 数です。メモリコントローラは接続されている SDRAM を同時にリフレッシュしますので、リフレッシュ周期は一番頻繁にリフレッシュが必要なデバイスにあわせてください。
分散リフレッシュを基本としますが、オートリフレッシュによってデータ転送が分断されることを防ぐために、複数回のリフレッシュ要求を蓄積することができます。蓄積回数は SDRAM 詳細設定レジスタ（MEMC[0x74]）の AREFWAIT ビット[3:0]で指定します。蓄積リフレッシュカウントが指定した回数を超えた場合に、SDRAM コントローラはリフレッシュ要求を認識して、次のメモリコントローラアイドルの際にオートリフレッシュを開始します。
リフレッシュタイミング要求タイミングにおいてデバイスがバスアクセスをしている場合は、バスアクセスを優先し、バスリクエストがなくなった時点でオートリフレッシュが行われます。

14.6.8 セルフリフレッシュ制御

メモリコントローラは SDRAM のセルフリフレッシュをサポートします。セルフリフレッシュモードには SDRAM 詳細設定レジスタ（MEMC[0x74]）の該当デバイスの SELF ビットを“1”にすることで移行します。セルフリフレッシュモードから抜ける場合は、対応するレジスタビットに“0”を設定してください。
セルフリフレッシュモードに入っている SDRAM へのアクセス (Read/Write) があった場合は、自動的にセルフリフレッシュモードから抜け出して、必要なコマンドを実行します。この後デバイスは基本的にアイドル状態になりますが、設定により再度セルフリフレッシュモードに入るようにすることができます。このモードを使用するには SDRAM 詳細設定レジスタ（MEMC[0x74]）の RESELF ビットを“1”に設定してください。SREFCNT ビットに設定したカウント分、対象とする SDRAM へのアクセスが無くなると、自動的にセルフリフレッシュモードに入ります。

14.6.9 ステータスレジスタ

SDRAM コントローラおよび接続されるデバイスに関するステータスを知ることができます。

14.7 レジスタ

14.7.1 レジスター覧

以下にメモリコントローラのレジスター覧を示します。これらのレジスタのベースアドレスは、0xFFFF_A000 です。

表 14.2 レジスター覧（**ベースアドレス：0xFFFF_A000**）

アドレス オフセット	レジスタ名称	レジスタ 略称	初期値	R/W	データ アクセス サイズ
SRAM controller registers					
0x20	SRAM デバイス 0 タイミングレジスタ	RAMTMG0	0x0000_1C7F	R/W	32
0x24	SRAM デバイス 0 制御レジスタ	RAMCNTL0	0x0000_0001	R/W	32
0x30	SRAM デバイス 1 タイミングレジスタ	RAMTMG1	0x0000_1C7F	R/W	32
0x34	SRAM デバイス 1 制御レジスタ	RAMCNTL1	0x0000_0001	R/W	32
0x40	SRAM デバイス 2 タイミングレジスタ	RAMTMG2	0x0000_1C7F	R/W	32
0x44	SRAM デバイス 2 制御レジスタ	RAMCNTL2	0x0000_0001	R/W	32
0x50	SRAM デバイス 3 タイミングレジスタ	RAMTMG3	0x0000_1C7F	R/W	32
0x54	SRAM デバイス 3 制御レジスタ	RAMCNTL3	0x0000_0001	R/W	32
SDRAM controller registers					
0x60	SDRAM モードレジスタ	SDMR	0x0000_0032	R/W	16/32
0x64	予約	-	-	-/-	-
0x68	予約	-	-	-/-	-
0x70	SDRAM 設定レジスタ	SDCNFG	0x0600_C700	R/W	32
0x74	SDRAM 詳細設定レジスタ	SDADVCNFG	0x000F_0300	R/W	32
0x80	初期化制御レジスタ	SDINIT	0x0000_0000	R/W	16/32
0x90	SDRAM リフレッシュタイマレジスタ	SDREF	0x0000_00A0	R/W	16/32
0xA0	SDRAM ステータスレジスタ	SDSTAT	0x0000_0202	RO	32

14.7.2 レジスタ詳細

SRAM デバイス[3:0] タイミングレジスタ (RAMTMG[3:0])
MEMC[0x20, 0x30, 0x40, 0x50] 初期値 = 0x0000_1C7F Read/Write

RSV (0)	WAITWE [4:0]					RSV (0)	WAITOE [4:0]					RSV (0)			
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV (0)	WAITWR [4:0]					RSV (0)	WAITRD [4:0]					RSV (0)			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 31 **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [30:26] **WAITWE [4:0] ライトイネーブル信号挿入ディレイ制御**
WE#信号挿入タイミング設定（初期値 0x0）

Bit 25 **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [24:20] **WAITOE [4:0] 出力イネーブル信号挿入ディレイ制御**
OE#信号挿入タイミング設定（初期値 0x0）

Bits [19:15] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [14:10] **WAITWR [4:0] ライトサイクルウェイト制御**
書き込み時のウエイトサイクル設定。SRAM/ROM デバイスに対し、書き込み時のウエイトサイクル数を設定します。（初期値 0x07）

Bit 9 **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [8:4] **WAITRD [4:0] リードサイクルウェイト制御**
読み出し時のウエイトサイクル設定。SRAM/ROM デバイスに対し、読み出し時のウエイトサイクル数を設定します。（初期値 0x07）

Bits [3:0] : **RSV 予約**
予約ビットです。必ず“0”を設定してください。

SRAM デバイス[3:0] 制御レジスタ (RAMCNTL[3:0])
MEMC[0x24, 0x34, 0x44, 0x54] 初期値 = 0x0000_0001 Read/Write

RSV (0)															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV (0)												WPROTECT	MWAITPOL[1:0]		RBLE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:4] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bit 3 **WPROTECT ライトプロテクト**
ライトプロテクトビットを設定したデバイスに対して、書き込みを実行しません。
0： ライトプロテクトなし（初期値）
1： ライトプロテクトあり

Bits [2:1] **MWAITPOL [1:0]**

MWAIT 信号の極性を設定します。

00：ディスエーブル（初期値）
01：ローアクティブでイネーブル
10：ハイアクティブでイネーブル
11：予約

Bit 0 **RBLE　バイトレーンコントロール制御設定**

バイトレーンコントロールのないデバイスで書き込み時のバイトコントロールを実現するには、このビットを“0”に設定し、対象となるデバイスの WE#と本デバイスの DQM を接続してください。

0： 読み出し時に DQM[1:0]が HIGH になり、書き込みを防ぎます。
1： 読み出し時に DQM[1:0]が LOW になって全バイトレーンから読み出しを行います。（初期値）

SDRAM モードレジスタ (SDMR)

MEMC[0x60]　初期値 = 0x0000_0032　Read/Write

RSV (0)															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV (0)						WBM	OP Mode [1:0]		CL [2:0]			BT	BL [2:0]		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:10]： **RSV　予約**

予約ビットです。必ず“0”を設定してください。

Bit 9： **WBM　ライトバーストモード**

0： 設定されたバースト長での書き込み（初期値）
1： シングル ロケーション アクセス

Bits [8:7]： **OP Mode [1:0]　オペレーションモード**

00：通常オペレーション
xx：他の組み合わせは全て予約

Bits [6:4]： **CL [2:0]　CAS レイテンシ**

000：予約
001：CL=1
010：CL=2
011：CL=3（初期値）
1xx：予約

Bit 3： **BT　バーストタイプ**

0： シーケンシャル（初期値）
1： 予約

Bits [2:0]： **BL [2:0]　バースト長**

000：BL=1
001：BL=2
010：BL=4（初期値）
011：BL=8
100：予約
101：予約
110：予約
111：予約

予約レジスタ

MEMC[0x64, 0x68]　　初期値 = 0x xxxx_xxxx　　- / -

RSV															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

予約レジスタです。アクセスしないでください。

SDRAM 設定レジスタ (SDCNFG)

MEMC[0x70]　　初期値 = 0x0600_C700　　Read/Write

RSV (0)					CLK-CTRL	CKE-CTRL	XBW (W)	COLW [7:4]				RSV (0)			
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV (0)					TRCD [1:0]		APCG	RSV (0)		RSV (0)		BNUM [1:0]		RSV (0)	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:27]　**RSV　予約**

予約ビットです。必ず“0”を設定してください。

Bit 26　**CLKCTRL　MCLK コントロール**

このビットはダイナミック MCLKEN コントロール (CKECTRL) をイネーブルにした場合にのみ、“1”に設定可能です。

0： MCLK が継続的に出力されます。ただし SDRAM の初期化は必要です。

1： SDRAM がアイドル状態（セルフリフレッシュ状態含む）にあるとき、MCLK を停止します。（初期値）

注意： SDRAM の初期化（MEMC [0x80] bit15 : INIT_SD = 1）を行った後、本ビット（CLKCTRL）の値を変更するには、いったん SDRAM をセルフリフレッシュモードに入れてください。すなわち、セルフリフレッシュモード IN → CLKCTRL 変更 → セルフリフレッシュモード OUT としてください。

Bit 25　**CKECTRL　ダイナミック MCLKEN コントロール**

0： MCLKEN＝H が継続的に出力されます（セルフリフレッシュ状態を除く）。

1： SDRAM がアイドル状態（セルフリフレッシュ状態か、バンクアクティブだがアクセスがない状態を含む）にあるとき、MCLKEN を LOW にします。（初期値）

注意： SDRAM の初期化（MEMC [0x80] bit15 : INIT_SD = 1）を行った後、本ビット（CKECTRL）の値を変更するには、いったん SDRAM をセルフリフレッシュモードに入れてください。すなわち、セルフリフレッシュモード IN → CKECTRL 変更 → セルフリフレッシュモード OUT としてください。

Bit 24　**XBW　外部バス幅 (Write Only)**

外部バス幅を選択します。

0： 外部バス幅＝16 ビット（初期値）

1： 外部バス幅＝32 ビット

Bits [23:20]　**COLW [7:4]　コラムアドレス幅**

SDRAM のコラムアドレス幅を設定します。

COLW [5:4]: SDRAM デバイス 0 設定用

COLW [7:6]: SDRAM デバイス 1 設定用

00：コラムアドレス A0 – A7（初期値）

01：コラムアドレス A0 – A8

10：コラムアドレス A0 – A9

11：コラムアドレス A0 – A9, A11

Bits [19:16] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [15:11] **RSV 予約**
予約ビットです。初期値=0b11000 ですが必ず“0”を設定してください。

Bits [10:9] **TRCD [1:0] RAS-CAS ディレイ**
MRAS＃から MCAS#までの遅延時間を設定（サイクル数）します。
00：予約
01：1 サイクル
10：2 サイクル
11：3 サイクル（初期値）

Bit 8 **APCG オートプリチャージコントロール**
オートプリチャージの設定をします。
0： オートプリチャージなし（オートリフレッシュ実行のタイミングで全バンクプリチャージされます）
1： オートプリチャージ有り（初期値）

Bits [7:6] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [5:4] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [3:2] **BNUM [1:0] バンク数**
接続されている SDRAM のバンク数を設定します。
BNUM0：SDRAM デバイス 0 設定用
BNUM1：SDRAM デバイス 1 設定用
0： ４バンクデバイス（初期値）
1： 予約（２バンクデバイス）

Bits [1:0] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

SDRAM 詳細設定レジスタ (SDADVCNFG)
MEMC[0x74] 初期値 = 0x000F_0300 Read/Write

RSV (0)												SREFCNT [3:0]			
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
				AREFWAIT [3:0]				CLK-FORCE	RESELF	RSV (0)		SELF [1:0]		RSV (0)	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:20] : **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [19:16] : **SREFCNT [3:0] セルフリフレッシュ再導入前サイクル数**
SDRAM へのアクセスがなくなった後、セルフリフレッシュモードに再び入るまでのサイクル数を設定します。設定値は RESELF ビット（MEMC[0x74] bit 6）を“1”に設定した場合のみ有効です。（初期値 0xF）

Bits [15:12] : **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [11:8] : **AREFWAIT [3:0] オートリフレッシュ ホールド数**
分散リフレッシュの実行タイミングにおいて、メモリコントローラインタフェイスが占有されている場合、リフレッシュを一旦保持し、後に複数回のリフレッシュを行うことができます。（初期値 0x03）

Bit 7 : **CLKFORCE**
0： MCLK クロックの出力は、メモリコントローラの状態に依存します。（初期値）
1： メモリコントローラの状態によらず、MCLK クロックを出力します。

Bit 6 : **RESELF セルフリフレッシュ再導入モード**
このビットはセルフリフレッシュモードに入っている SDRAM を読み出し/書き込みアクセスのために一旦起動させた後、再びセルフリフレッシュモードに再導入するかどうかをコントロールします。
0： セルフリフレッシュモードに再導入しません。（初期値）
1： SREFCNT[3:0]サイクル分アクセスがない場合再びセルフリフレッシュモードに SDRAM を導入します。

Bits [5:4] : **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [3:2] : **SELF [1:0] セルフリフレッシュモード ON/OFF**
このビットはセルフリフレッシュモードに入っている SDRAM が読み書きのために起きた場合、ゼロに自動的にクリアされます（RESELF モード設定時は除く）。
SELF0：SDRAM デバイス 0 設定用
SELF1：SDRAM デバイス 1 設定用
0： セルフリフレッシュモードから抜けます。（初期値）
1： SDRAM をセルフリフレッシュモードに設定します。

Bits [1:0] : **RSV 予約**
予約ビットです。必ず“0”を設定してください。

初期化制御レジスタ (SDINIT)
MEMC[0x80] 初期値 = 0x0000_0000 Read/Write

								RSV (0)							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INIT_SD	RSV (0)							LMR	AREF	PCG-ALL	RSV (0)	DEVSEL [1:0]		RSV (0)	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

注意：このレジスタの複数の命令を同時に設定しないでください。
通常 SDRAM の初期化においては INIT_SD をご使用ください。
その他の命令はマニュアルによる初期化が必要な場合に使用してください。
いずれの命令も対象となるデバイスを DEVSEL[1:0]（MEMC[0x80] bits[3:2]）で選択してください。

Bits [31:16] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bit 15 **INIT_SD SDRAM 初期化**
SDRAM の初期化シーケンスを自動的に行います。DEVSEL（本レジスタの bit[3:0]）にて対象となるデバイスを選択してください。
0： 何もなし（初期値）
1： SDRAM の初期化を行います（初期化完了後、自動的に“0”に戻ります）。

Bits [14:8] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bit 7 **LMR　モードレジスタセット**

SDRAM へモードレジスタセット命令を発行します。
0： 何もなし（初期値）
1： モードレジスタセット命令を発行します（命令発行後、自動的に 0 に戻ります）

Bit 6 **AREF　オートリフレッシュ**

SDRAM へオートリフレッシュ命令を発行します。
0： 何もなし（初期値）
1： オートリフレッシュ命令を発行します（命令発行後、自動的に 0 に戻ります）

Bit 5 **PCGALL　プリチャージオール**

SDRAM に全バンクプリチャージ命令を発行します。
0： 何もなし（初期値）
1： 全バンクプリチャージ命令を発行します（命令発行後、自動的に 0 に戻ります）

Bit 4 **RSV　予約**

予約ビットです。必ず“0”を設定してください。

Bits [3:2] **DEVSEL [1:0]　デバイス選択**

命令を発行するデバイスを選択します。
DEVSEL0：SDRAM デバイス 0 設定用
DEVSEL1：SDRAM デバイス 1 設定用
0： 選択しない（初期値）
1： 選択した

Bits [1:0] **RSV　予約**

予約ビットです。必ず“0”を設定してください。

SDRAM リフレッシュタイマレジスタ　(SDREF)

MEMC[0x90]　　初期値 = 0x0000_00A0　　Read/Write

RSV (0)															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV (0)				REFTIME [11:0]											
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:12] **RSV　予約**

予約ビットです。必ず“0”を設定してください。

Bits [11:0] **REFTIME [11:0]　リフレッシュ間隔**

分散リフレッシュの発行間隔を設定します。
分散オートリフレッシュ間隔に見合った HCLK サイクル数を設定してください。
例えばリフレッシュ間隔が 16µs で、システムクロック HCLK の周波数が 10MHz であったとすると、
16µs x 10MHz = 160 サイクル（=1010_0000b）を設定してください。

SDRAM ステータスレジスタ (SDSTAT)
MEMC[0xA0] 初期値 = 0x0000_0202 Read Only

RSV															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RSV				DEVST1 [3:0]				RSV				DEVST0 [3:0]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:12] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [11:8] **DEVST1 [3:0] SDRAM デバイス 1 ステータス**
SDRAM デバイス 1 の現在の状態を示します
DEVST1 [0] 1 : SDRAM デバイス 1 が使用可能（イネーブルで、初期設定が完了している状態）
0：SDRAM デバイス 1 が使用できない
DEVST1 [1] 1：SDRAM デバイス 1 がアイドル状態かサスペンド状態
0：SDRAM デバイス 1 がその他の状態
DEVST1 [2] 1：SDRAM デバイス 1 のいずれかのバンクの ROW が活性化されている
0：SDRAM デバイス 1 の ROW 不活性状態
DEVST1 [3] 1：SDRAM デバイス 1 がセルフリフレッシュモード中
0：SDRAM デバイス 1 がセルフリフレッシュモードではない

Bits [7:4] **RSV 予約**
予約ビットです。必ず“0”を設定してください。

Bits [3:0] **DEVST0 [3:0] SDRAM デバイス 0 ステータス**
SDRAM デバイス 0 の現在の状態を示します。
DEVST0 [0] 1 : SDRAM デバイス 0 が使用可能（イネーブルで、初期設定が完了している状態）。
0：SDRAM デバイス 0 が使用できない。
DEVST0 [1] 1：SDRAM デバイス 0 がアイドル状態かサスペンド状態。
0：SDRAM デバイス 0 がその他の状態。
DEVST0 [2] 1：SDRAM デバイス 0 のいずれかのバンクの ROW が活性化されている。
0：SDRAM デバイス 0 の ROW 不活性状態。
DEVST0 [3] 1：SDRAM デバイス 0 がセルフリフレッシュモード中。
0：SDRAM デバイス 0 がセルフリフレッシュモードではない。

15. 割り込みコントローラ (INT)

15.1 概要

割り込みコントローラ (INTC) は 2 つの高速割り込み要求 (FIQ) と 32 の通常の割り込み要求 (IRQ)を処理することができます。各割り込み要求入力への周辺回路／機能の割り当てを 表 15.1 に示します。リセット後は、割り込み要求はすべてHIGHアクティブのレベルトリガ入力です。 INT[7:0] については、リセット直後はローレベルトリガ設定になっていますが、必要に応じて、本割り込みコントローラにより入力信号の極性とレベル /エッジトリガの選択が行えるようになっています。

割り込みコントローラは、FIQ および IRQ 入力を処理し、ARM720T コアに対して nFIQ と nIRQ の 2 つの割り込み要求信号を出力します。

ハードウェア上、割り込みの優先順位は設定されていません。

表 15.1 割り込みソース一覧

割り込みの種類	割り込みレベル	割り込みソース	説明
高速割り込み要求 FIQ	FIQ0	ウォッチドッグタイマ	
	FIQ1(***)	GPIOB0 端子	Ex. Battery Low
通常割り込み要求 IRQ	IRQ0	ウォッチドッグタイマ	
	IRQ1	割り込みコントローラ	レジスタによるソフトウェアリクエスト
	IRQ2	ARM720T COMMRx	Debug Communication Port
	IRQ3	ARM720T COMMTx	Debug Communication Port
	IRQ4	タイマ A	16 ビットタイマチャネル 0
	IRQ5	タイマ A	16 ビットタイマチャネル 1
	IRQ6	タイマ A	16 ビットタイマチャネル 2
	IRQ7	予約	
	IRQ8	JPEG1 制御	
	IRQ9	DMAC1	AHB1 上の DMAC
	IRQ10	JPEG1 DMAC	
	IRQ11	カメラ 1 I/F	
	IRQ12	ARS	
	IRQ13	DMAC2	DMA INT（AHB2 上の JPEG 用の DMAC）
	IRQ14 (*)	GPIOA または GPIOB	GPIOA/B より割り込み入力端子選択
	IRQ15	SPI	SPI TXRDY/RXRDY
	IRQ16	I2C	Transfer Complete
	IRQ17	UART1	UART TXRDY/RXRDY
	IRQ18	RTC	Alarm or Timer tick
	IRQ19	CF カード I/F	
	IRQ20 (**)	INTGB	GPIOB7～GPIOB0 直接入力代表 bit
	IRQ21	タイマ B	
	IRQ22	予約	
	IRQ23	JPEG2 制御	
	IRQ24	JPEG2 DMAC	
	IRQ25	カメラ 2 I/F	
	IRQ26	UART2	
	IRQ27	UART3	
	IRQ28	SD メモリ	
	IRQ29	USB DEV	
	IRQ30	ADC	
	IRQ31	I2S	

注意(*) ：GPIOA[7:0]または GPIOB[7:0]から割り込み要求を設定します。詳細は“26. GPIO”のレジスタ詳細説明 GPIO[0x60]～GPIO[0x6C]を参照してください。

注意(**) ：GPIOB[7:0]から割り込み要求を設定します。詳細は IRQxx レジスタ詳細説明を参照してください。

注意(***) ：GPIOB0 端子から直接入力（デフォルトはローレベルアクティブ割込み要求。但し、内部で反転していますので、内部での割り込みレベルはハイレベルアクティブになります）。

表 15.2　割り込みソース一覧

割り込みの種類	割り込みレベル	割り込みソース	説明
GPIOB 割り込み要求 IRQxx	IRQGB0(*)	INT0	GBIOB0 直接入力
	IRQGB1(*)	INT1	GBIOB1 直接入力
	IRQGB2(*)	INT2	GBIOB2 直接入力
	IRQGB3(*)	INT3	GBIOB3 直接入力
	IRQGB4(*)	INT4	GBIOB4 直接入力
	IRQGB5(*)	INT5	GBIOB5 直接入力
	IRQGB6(*)	INT6	GBIOB6 直接入力
	IRQGB7(*)	INT7	GBIOB7 直接入力

注意(*) : GPIOB[7:0]端子から直接入力（デフォルトはローレベルアクティブ割込み要求。但し、内部で反転していますので、内部での割り込みレベルはハイレベルアクティブになります)。これらの割込み設定（イネーブル、極性、レベル等）は割込みコントローラの制御レジスタによってのみ変更が可能です。IRQ14(*)とは異なり、”26.GPIO”のレジスタ詳細説明 GPIO[0x60]～GPIO[0x6C]では変更可能できません。

15.2 ブロック図

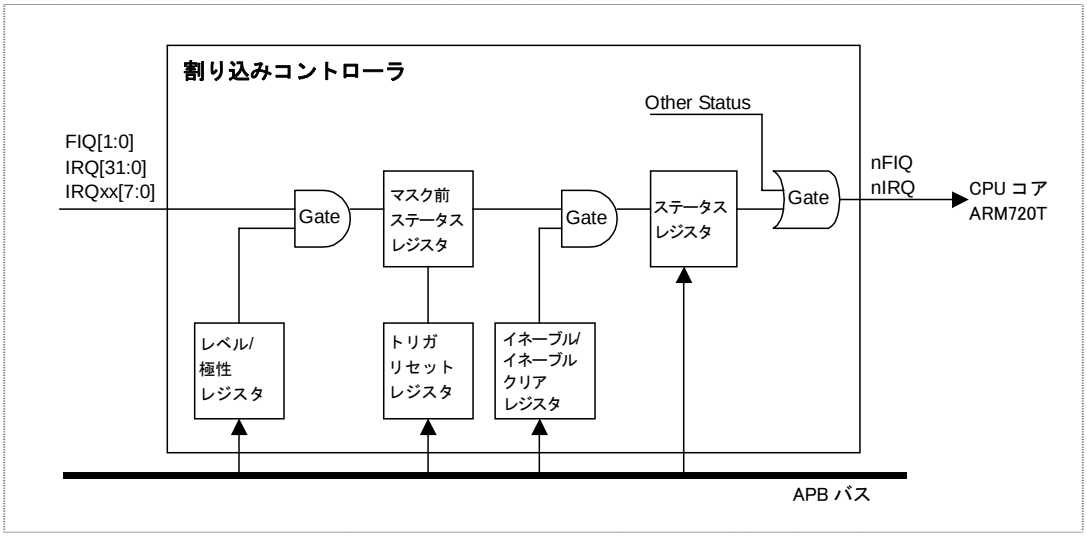


図 15.1 割り込みコントローラ ブロック図

15.3 FIQ

FIQ0、FIQ1 にはそれぞれ割り込み出力としてのウォッチドッグタイマ、および GPIO ポート B[0] が割り当てられています。この 2 本の FIQ 信号は ARM720T に入力する nFIQ を生成するために使用されます。

15.4 IRQ

IRQ には内部デバイスの割り込み要求が割り当てられています。すべてのIRQ割り込み要求は、ARM720Tに入力するnIRQ信号を生成するためにORされます。割り込みコントローラは全部で 32 本の割り込み要求入力IRQ31～IRQ0 を持ち、それぞれ、割り込み制御レジスタのビットに 1 対 1 で対応しています。たとえば、IRQ0 はビット 0 に割り付けられ、IRQ1 はビット 1 に割り付けられています。対応については 表 15.1 を参照してください。

外部割込みの INT0～INT7 において、IRQGB0～IRQGB7 で割込み要求を受けた後、外部代表割込みとして IRQ20 に割当てられます。

15.5 外部端子

割り込みコントローラ (INT) に関連する外部端子は以下の通りです。

端子名	入出力	端子機能	マルチプレクス端子*／備考
FIQ1	I	高速割り込み端子 1	GPIOB0/I2S_WS
INT0	I	外部割り込み端子 0	GPIOB0/I2S_WS
INT1	I	外部割り込み端子 1	GPIOB1/I2S_SCK
INT2	I	外部割り込み端子 2	GPIOB2/I2S_SDO
INT3	I	外部割り込み端子 3	GPIOB3/I2S_SDI
INT4	I	外部割り込み端子 4	GPIOB4/TimerA0out
INT5	I	外部割り込み端子 5	GPIOB5/TimerA1out
INT6	I	外部割り込み端子 6	GPIOB6/TimerA2out
INT7	I	外部割り込み端子 7	GPIOB7/TimerBIn

注意(*)：INT 関連の外部割り込み端子は、GPIO 端子入力により行います。GPIO 端子を割り込み入力として使用する場合は割り込み制御用の内部レジスタの設定を行います。その他の端子機能として使用する場合は、GPIO 端子機能レジスタにより“GPIO 以外の機能 1 または 2”に設定してください。

15.6 レジスタ

15.6.1 レジスター覧

割り込みコントローラの制御レジスタが配置されているベースアドレスは 0xFFFF_F000 です。

表 15.3 レジスター覧（**ベースアドレス：0xFFFF_F000**）

アドレス オフセット	レジスタ名称	初期値	R/W	データアクセス サイズ
0x000	IRQ ステータスレジスタ	0x0000_0000	RO	32
0x004	IRQ マスク前ステータスレジスタ	0x0000_0000(*1)	RO	32
0x008	IRQ イネーブルレジスタ	0x0000_0000	R/W	32
0x00C	IRQ イネーブルクリアレジスタ	0x0000_0000	WO	32
0x010	ソフトウェア IRQ レジスタ	0x0000_0000	WO	32
0x020	IRQxx ステータスレジスタ	0x0000_0000	RO	32
0x024	IRQxx マスク前ステータスレジスタ	0x0000_0000(*1)	RO	32
0x028	IRQxx イネーブルレジスタ	0x0000_0000	R/W	32
0x02C	IRQxx イネーブルクリアレジスタ	0x0000_0000	WO	32
0x080	IRQ レベルレジスタ	0x0000_0000	R/W	32
0x084	IRQ 極性レジスタ	0xFFFF_FFFF	R/W	32
0x088	IRQ トリガリセットレジスタ	0x0000_0000	WO	32
0x0A0	IRQxx レベルレジスタ	0x0000_0000	R/W	32
0x0A4	IRQxx 極性レジスタ	0x0000_0FFF	R/W	32
0x0A8	IRQxx トリガリセットレジスタ	0x0000_0000	WO	32
0x100	FIQ ステータスレジスタ	0x0000_0000	RO	32
0x104	FIQ マスク前ステータスレジスタ	0x0000_0000(*1)	RO	32
0x108	FIQ イネーブルレジスタ	0x0000_0000	R/W	32
0x10C	FIQ イネーブルクリアレジスタ	0x0000_0000	WO	32
0x180	FIQ レベルレジスタ	0x0000_0000	R/W	32
0x184	FIQ 極性レジスタ	0x0000_0003	R/W	32
0x188	FIQ トリガリセットレジスタ	0x0000_0000	WO	32

注意(*1)：IRQ/IRQxx/FIQ マスク前ステータスレジスタの初期値は、システムの構成条件により変化します。

15.6.2 レジスタ詳細説明

特に指定のない場合、予約されていないレジスタビットの初期値はすべて“0”です。

IRQ ステータスレジスタ

INT[0x000]　　初期値 = 0x0000_0000　　Read Only

IRQ[31:16] ステータス															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQ[15:0] ステータス															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] : **IRQ[31:0]ステータス**

各ユニットからの割り込み要求の発生状態（マスク後）を示します。

0 (r)：割り込み要求なし

1 (r)：割り込み要求あり

これらのステータスビットは、IRQ イネーブルレジスタ (INT[0x008]) で割り込みを許可されている IRQ の状態を示します。割り込みを禁止したユニットからの割り込み要求が発生していても、そのビットは“1”にはなりません。“1”にセットされている割り込み要求は、CPU に送られます。これらのビットは、IRQ マスク前ステータスレジスタの対応するビットをクリアすることによって“0”に戻ります。

ビット [31:0]は同じ番号のIRQ[31:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.1 を参照してください。

IRQ マスク前ステータスレジスタ

INT[0x004]　　初期値 = 0x0000_0000　　Read Only

IRQ[31:16] マスク前ステータス															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQ[15:0] マスク前ステータス															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] : **IRQ[31:0]マスク前ステータス**

各ユニットの割り込み要因の発生状態（マスク前）を示します。

0 (r)：割り込み要因なし

1 (r)：割り込み要因あり

これらのステータスビットは、IRQ イネーブルレジスタ (INT[0x008]) で割り込みをマスクする前の IRQ の状態を示します。割り込みを禁止したユニットで割り込み要因が発生している場合も、ビットが“1”になります。レベルトリガ割り込みの場合、これらのビットは各ユニットの割り込みフラグをクリアすることによって“0”に戻ります。エッジトリガ割り込みの場合は、IRQ トリガリセットレジスタ (INT[0x088]) の対応するビットに“1”を書き込むことによって“0”に戻ります。

ビット[31:0]は同じ番号のIRQ[31:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.1 を参照してください。

IRQ イネーブルレジスタ

INT[0x008]　　初期値 = 0x0000_0000　　Read/Write

IRQ[31:16] イネーブル															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQ[15:0] イネーブル															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQ[31:0]イネーブル**

各ユニットからの割り込み要求 (IRQ) 入力を許可します。

0 (r)： 割り込み要求禁止状態

1 (r)： 割り込み要求許可状態

0 (w)： 無効

1 (w)： 割り込み要求入力を許可

このレジスタを読み出すと、各割り込み要求 (IRQ) 入力が現在許可（ビットが“1”）されているか禁止（ビットが“0”）されているかが分かります。

レジスタへの書き込みでは、“1”を書き込むことによって割り込み要求 (IRQ) 入力が許可されます。割り込みコントローラはそのビットに対応する IRQ の入力を受け付け、 CPU に割り込み要求を出力します。“0”の書き込みは無効で、このレジスタへのアクセスによって割り込み要求 (IRQ) 入力が禁止されることはありません。ビットのクリアは IRQ イネーブルクリアレジスタ (INT[0x00C]) で行います。

リセット時は、すべての割り込みが禁止状態に設定されます。

ビット[31:0]は同じ番号のIRQ[31:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.1 を参照してください。

IRQ イネーブルクリアレジスタ

INT[0x00C]　　初期値 = 0x0000_0000　　Write Only

IRQ[31:16] イネーブルクリア															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQ[15:0] イネーブルクリア															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQ[31:0]イネーブルクリア**

各ユニットからの割り込み要求 (IRQ) 入力を禁止（マスク）します。

0 (w)： 無効

1 (w)： 割り込み要求入力を禁止

“1”を書き込むことによって、そのビットに対応する IRQ イネーブルレジスタ (INT[0x008]) の IRQ イネーブルビットがクリアされ、割り込み要求 (IRQ) 入力が禁止されます。“0”の書き込みは無効です。

ビット[31:0]は同じ番号のIRQ[31:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.1 を参照してください。

ソフトウェア IRQ レジスタ

INT[0x010]　　初期値 = 0x0000_0000　　Write Only

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a														ソフトウェア IRQ	RSV

Bit 1 :　　**ソフトウェア IRQ**

IRQ1 のソフトウェア割り込みを制御します。

0 (w)　ソフトウェア割り込み要求をクリア

1 (w)　ソフトウェア割り込みを発生

現在設定されているソフトウェア割り込み要因の状態は IRQ マスク前ステータスレジスタ (INT[0x004]) のビット 1 から読み出すことができます。

Bit 0 :　　**RSV　予約**

IRQxx ステータスレジスタ

INT[0x020]　　初期値 = 0x0000_0000　　Read Only

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQxx[31:16] ステータス															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRQxx[15:0] ステータス															

Bits [31:0] :　　**IRQxx[31:0]ステータス**

IRQxx[31:12]　　: n/a

IRQxx[11:8]　　: Reserved

IRQxx[7:0]　　: IRQxx(IRQGB[7:0])ステータス

各ユニットからの割り込み要求の発生状態（マスク後）を示します。

0 (r) : 割り込み要求なし

1 (r) : 割り込み要求あり

これらのステータスビットは、IRQxx イネーブルレジスタ (INT[0x028]) で割り込みを許可されている IRQxx の状態を示します。割り込みを禁止したユニットからの割り込み要求が発生していても、そのビットは“1”にはなりません。“1”にセットされている割り込み要求は、CPU に送られます。これらのビットは、IRQ マスク前ステータスレジスタの対応するビットをクリアすることによって“0”に戻ります。

ビット [7:0]は同じ番号のIRQxx[7:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.2 を参照してください。

IRQxx マスク前ステータスレジスタ

INT[0x024]　初期値 = 0x0000_0000　Read Only

IRQxx[31:16] マスク前ステータス															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQxx[15:0] マスク前ステータス															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQxx[31:0]マスク前ステータス**

IRQxx[31:12]　：　n/a
IRQxx[11:8]　：　Reserved
IRQxx[7:0]　：　IRQxx(IRQGB[7:0])マスク前ステータス

各ユニットの割り込み要因の発生状態（マスク前）を示します。
　0 (r)：　割り込み要因なし
　1 (r)：　割り込み要因あり

これらのステータスビットは、IRQxx イネーブルレジスタ (INT[0x028]) で割り込みをマスクする前の IRQxx の状態を示します。割り込みを禁止したユニットで割り込み要因が発生している場合も、ビットが“1”になります。レベルトリガ割り込みの場合、これらのビットは各ユニットの割り込みフラグをクリアすることによって“0”に戻ります。エッジトリガ割り込みの場合は、IRQxx トリガリセットレジスタ (INT[0x0A8]) の対応するビットに“1”を書き込むことによって“0”に戻ります。

ビット[7:0]は同じ番号のIRQxx[7:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.2 を参照してください。

IRQxx イネーブルレジスタ

INT[0x028]　初期値 = 0x0000_0000　Read/Write

IRQxx[31:16] イネーブル															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQxx[15:0] イネーブル															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQxx[31:0]イネーブル**

IRQxx[31:12]　：　n/a
IRQxx[11:8]　：　Reserved
IRQxx[7:0]　：　IRQxx(IRQGB[7:0])イネーブルレジスタ

各ユニットからの割り込み要求 (IRQ) 入力を許可します。
　0 (r)：　割り込み要求禁止状態
　1 (r)：　割り込み要求許可状態
　0 (w)：　無効
　1 (w)：　割り込み要求入力を許可

このレジスタを読み出すと、各割り込み要求 (IRQxx) 入力が現在許可（ビットが“1”）されているか禁止（ビットが“0”）されているかが分かります。

レジスタへの書き込みでは、“1”を書き込むことによって割り込み要求 (IRQxx) 入力が許可されます。割り込みコントローラはそのビットに対応する IRQxx の入力を受け付け、 CPU に割り込み要求を出力します。“0”の書き込みは無効で、このレジスタへのアクセスによって割り込み要求 (IRQxx) 入力が禁止されることはありません。ビットのクリアは IRQxx イネーブルクリアレジスタ (INT[0x02C]) で行います。

リセット時は、すべての割り込みが禁止状態に設定されます。

ビット[7:0]は同じ番号のIRQxx[7:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.2 を参照してください。

IRQxx イネーブルクリアレジスタ

INT[0x02C]　　初期値 = 0x0000_0000　　Write Only

IRQxx[31:16] イネーブルクリア															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQxx[15:0] イネーブルクリア															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQxx[31:0]イネーブルクリア**

IRQxx[31:12]　：　n/a
IRQxx[11:8]　：　Reserved
IRQxx[7:0]　：　IRQxx(IRQGB[7:0])イネーブルクリア

各ユニットからの割り込み要求 (IRQxx) 入力を禁止（マスク）します。
0 (w)： 無効
1 (w)： 割り込み要求入力を禁止
“1”を書き込むことによって､そのビットに対応する IRQxx イネーブルレジスタ (INT[0x028]) の IRQxx イネーブルビットがクリアされ、割り込み要求 (IRQxx) 入力が禁止されます。“0”の書き込みは無効です。
ビット[7:0]は同じ番号のIRQxx[7:0]に 1 対 1 に対応しています｡各ビットが示す割り込みの内容については、表 15.2 を参照してください。

IRQ レベルレジスタ

INT[0x080]　　初期値 = 0x0000_0000　　Read/Write

IRQ[31:16] レベル															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQ[15:0] レベル															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQ[31:0]レベル**

各 IRQ 入力のトリガモードを設定します。
0 (r/w)：レベルトリガモード
1 (r/w)：エッジトリガモード
この選択により、割り込み要求 (IRQ) 信号をレベルでサンプリングするか、信号の立ち下がり / 立ち上がりエッジでサンプリングするかが決まります｡信号の極性（LOWレベル/立ち下がりエッジまたはHIGHレベル/立ち上がりエッジ）はIRQ極性レジスタ (INT[0x084]) で設定します。ビット[31:0]は同じ番号のIRQ[31:0]に 1 対 1 に対応しています｡各ビットが示す割り込みの内容については、表 15.1 を参照してください。

注意：通常はこのレジスタの値を変更することなくリセット時のままお使いください。

IRQ 極性レジスタ

INT[0x084]　　初期値 = 0xFFFF_FFFF　　Read/Write

IRQ[31:16] 極性															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQ[15:0] 極性															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQ[31:0]極性**

各 IRQ 信号の極性を設定します。

0 (r/w)：LOW レベル/立ち下がりエッジ

1 (r/w)：HIGH レベル/立ち上がりエッジ

この選択に従って割り込み要求 (IRQ)信号がサンプリングされます。レベル/エッジの選択はIRQ レベルレジスタ (INT[0x080]) で行います。ビット[31:0]は同じ番号のIRQ[31:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.1 を参照してください。

注意：通常はこのレジスタの値を変更することなくリセット時のままお使いください。
SD カード検出の割り込みを要因に使用する場合は、bit28 を 0 に設定してください。

IRQ トリガリセットレジスタ

INT[0x088]　　初期値 = 0x0000_0000　　Write Only

IRQ[31:16] トリガリセット															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQ[15:0] トリガリセット															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQ[31:0]トリガリセット**

エッジトリガモードに設定されている割り込みのマスク前ステータスをクリアします。

0 (w)： 無効

1 (w)： 割り込みステータスをクリア

“1”を書き込むと、IRQマスク前ステータスレジスタ (INT[0x004]) 内の対応するステータスビットの中で、エッジトリガモードに設定されているビットをクリアします。ビット[31:0]は同じ番号のIRQ[31:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.1 を参照してください。

IRQxx レベルレジスタ

INT[0x0A0]　　初期値 = 0x0000_0000　　Read/Write

IRQxx[31:16] レベル															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQxx[15:0] レベル															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] :　**IRQxx[31:0]レベル**

IRQxx[31:12] : n/a
IRQxx[11:8] : Reserved
IRQxx[7:0] : IRQxx(IRQGB[7:0])レベル

各 IRQxx 入力のトリガモードを設定します。
0 (r/w)：レベルトリガモード
1 (r/w)：エッジトリガモード

この選択により、割り込み要求 (IRQxx) 信号をレベルでサンプリングするか、信号の立ち下がり/立ち上がりエッジでサンプリングするかが決まります。信号の極性（LOW レベル/立ち下がりエッジまたは HIGH レベル/立ち上がりエッジ）は IRQxx 極性レジスタ (INT[0x0A4]) で設定します。
ビット[7:0]は同じ番号のIRQxx[7:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.2 を参照してください。

IRQxx 極性レジスタ

INT[0x0A4]　　初期値 = 0x0000_0FFF　　Read/Write

IRQxx[31:16] 極性															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQxx[15:0] 極性															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0] :　**IRQxx[31:0]極性**

IRQxx[31:12] : n/a
IRQxx[11:8] : Reserved
IRQxx[7:0] : IRQxx(IRQGB[7:0])極性

各 IRQxx 信号の極性を設定します。
0 (r/w)：LOW レベル/立ち下がりエッジ
1 (r/w)：HIGH レベル/立ち上がりエッジ

この選択に従って割り込み要求 (IRQxx)信号がサンプリングされます。レベル/エッジの選択は IRQxx レベルレジスタ (INT[0x0A0]) で行います。
ビット[7:0]は同じ番号のIRQxx[7:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.2 を参照してください。
但し、GPIOB[7:0]は内部で反転しています。ピンでローレベルアクティブの場合は、内部ではハイレベルアクティブとなりますので、”1”を設定してください。

IRQxx トリガリセットレジスタ

INT[0x0A8]　　初期値 = 0x0000_0000　　Write Only

IRQxx[31:16] トリガリセット															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRQxx[15:0] トリガリセット															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]：　**IRQxx[31:0]トリガリセット**

IRQxx[31:12]　：　n/a
IRQxx[11:8]　：　Reserved
IRQxx[7:0]　：　IRQxx(IRQGB[7:0])トリガリセット

エッジトリガモードに設定されている割り込みのマスク前ステータスをクリアします。
0 (w)：　無効
1 (w)：　割り込みステータスをクリア
“1”を書き込むと、IRQxx マスク前ステータスレジスタ (INT[0x024]) 内の対応するステータスビットの中で、エッジトリガモードに設定されているビットをクリアします。
ビット[7:0]は同じ番号のIRQxx[7:0]に 1 対 1 に対応しています。各ビットが示す割り込みの内容については、表 15.2 を参照してください。

FIQ ステータスレジスタ

INT[0x100]　　初期値 = 0x0000_0000　　Read Only

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a														FIQ [1:0] ステータス	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [1:0]：　**FIQ[1:0]ステータス**

FIQ 割り込み要求の発生状態（マスク後）を示します。
0 (r)：割り込み要求なし
1 (r)：割り込み要求あり
これらのステータスビットは、FIQ イネーブルレジスタ (INT[0x108]) で割り込みを許可されている FIQ の状態を示します。割り込みを禁止したユニットからの割り込み要求が発生していても、そのビットは“1”にはなりません。“1”にセットされている割り込み要求は、CPU に送られます。これらのビットは、FIQ マスク前ステータスレジスタの対応するビットをクリアすることによって“0”に戻ります。
ビット[1:0]は同じ番号の FIQ[1:0]に 1 対 1 に対応しています。

FIQ マスク前ステータスレジスタ

INT[0x104]　　初期値 = 0x0000_0000　　Read Only

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a														FIQ [1:0]マスク前ステータス	

Bits [1:0]：　**FIQ[1:0]マスク前ステータス**

FIQ 割り込み要因の発生状態（マスク前）を示します。

0 (r)：割り込み要因なし

1 (r)：割り込み要因あり

これらのステータスビットは、FIQ イネーブルレジスタ (INT[0x108]) で割り込みをマスクする前の FIQ の状態を示します。割り込みを禁止したユニットで割り込み要因が発生している場合も、ビットが“1”になります。レベルトリガ割り込みの場合、これらのビットは各ユニットの割り込みフラグをクリアすることによって“0”に戻ります。エッジトリガ割り込みの場合は、FIQ トリガリセットレジスタ (INT[0x188]) の対応するビットに“1”を書き込むことによって“0”に戻ります。

ビット[1:0]は同じ番号の FIQ[1:0]に 1 対 1 に対応しています。

FIQ イネーブルレジスタ

INT[0x108]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a														FIQ [1:0] イネーブル	

Bits [1:0]：　**FIQ[1:0]イネーブル**

高速割り込み要求 (FIQ) 入力を許可します。

0 (r)： 割り込み要求禁止状態

1 (r)： 割り込み要求許可状態

0 (w)： 無効

1 (w)： 割り込み要求入力を許可

このレジスタを読み出すと、各割り込み要求 (FIQ) 入力が現在許可（ビットが“1”）されているか禁止（ビットが“0”）されているかが分かります。

レジスタへの書き込みでは、“1”を書き込むことによって割り込み要求 (FIQ) 入力が許可されます。割り込みコントローラはそのビットに対応する FIQ の入力を受け付け、CPU に割り込み要求を出力します。“0”の書き込みは無効で、このレジスタへのアクセスによって割り込み要求 (FIQ) 入力が禁止されることはありません。ビットのクリアは FIQ イネーブルクリアレジスタ (INT[0x10C]) で行います。

リセット時は、すべての割り込みが禁止状態に設定されます。

ビット[1:0]は同じ番号の FIQ[1:0]に 1 対 1 に対応しています。

FIQ イネーブルクリアレジスタ

INT[0x10C]　　初期値 = 0x0000_0000　　Write Only

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a														FIQ [1:0]イネーブルクリア	

Bits [1:0]：　**FIQ[1:0]イネーブルクリア**

高速割り込み要求 (FIQ) 入力を禁止（マスク）します。

0 (w)： 無効

1 (w)： 割り込み要求入力を禁止

“1”を書き込むことによって、そのビットに対応する FIQ イネーブルレジスタ (INT[0x108]) の FIQ イネーブルビットがクリアされ、割り込み要求 (FIQ) 入力が禁止されます。“0”の書き込みは無効です。ビット[1:0]は同じ番号の FIQ[1:0]に 1 対 1 に対応しています。

FIQ レベルレジスタ

INT[0x180]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a														FIQ [1:0] レベル	

Bits [1:0]：　**FIQ[1:0]レベル**

各 FIQ 入力のトリガモードを設定します。

0 (r/w)：レベルトリガモード

1 (r/w)：エッジトリガモード

この選択により、割り込み要求 (FIQ) 信号をレベルでサンプリングするか、信号の立ち下がり/立ち上がりエッジでサンプリングするかが決まります。信号の極性（LOW レベル/立ち下がりエッジ、または HIGH レベル/立ち上がりエッジ）は FIQ 極性レジスタ (INT[0x184]) で設定します。ビット[1:0]は同じ番号の FIQ[1:0]に 1 対 1 に対応しています。

FIQ 極性レジスタ

INT[0x184]　　初期値 = 0x0000_0003　　Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a														FIQ [1:0] 極性	

Bits [1:0]：　**FIQ[1:0]極性**

各 FIQ 信号の極性を設定します。

0 (r/w)：LOW レベル/立ち下がりエッジ

1 (r/w)：HIGH レベル/立ち上がりエッジ

この選択にしたがって割り込み要求 (FIQ) 信号がサンプリングされます。レベル/エッジの選択は FIQ レベルレジスタ (INT[0x180]) で行います。ビット[1:0]は同じ番号の FIQ[1:0]に 1 対 1 に対応しています。

但し、GPIOB0 は内部で反転しています。ピンでローレベルアクティブの場合は、内部ではハイレベルアクティブとなりますので、”1”を設定してください。

FIQ トリガリセットレジスタ

INT[0x188]　　初期値 = 0x0000_0000　　Write Only

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a														FIQ [1:0] トリガリセット	

Bits [1:0] :　**FIQ[1:0]トリガリセット**

エッジトリガモードに設定されている割り込みのマスク前ステータスをクリアします。

0 (w)： 無効

1 (w)： 割り込みステータスをクリア

“1”を書き込むと、FIQ マスク前ステータスレジスタ (INT[0x104]) 内の対応するステータスビットの中で、エッジトリガモードに設定されているビットをクリアします。ビット[1:0]は同じ番号の FIQ[1:0]に 1 対 1 に対応しています。

16. UART1/2/3

16.1 概要

UART1/2/3 は、業界標準の 16550 と互換性を持つ非同期データ転送用インタフェースです。CPU のパラレルデータをシリアル変換して周辺デバイスに転送するとともに、周辺デバイスから送られたシリアルデータを入力しパラレルデータに変換します。

S2S65A00 は、まったく同じ機能を持つ UART モジュールを 3 個(UART1/2/3)持っています。

16.2 ブロック図

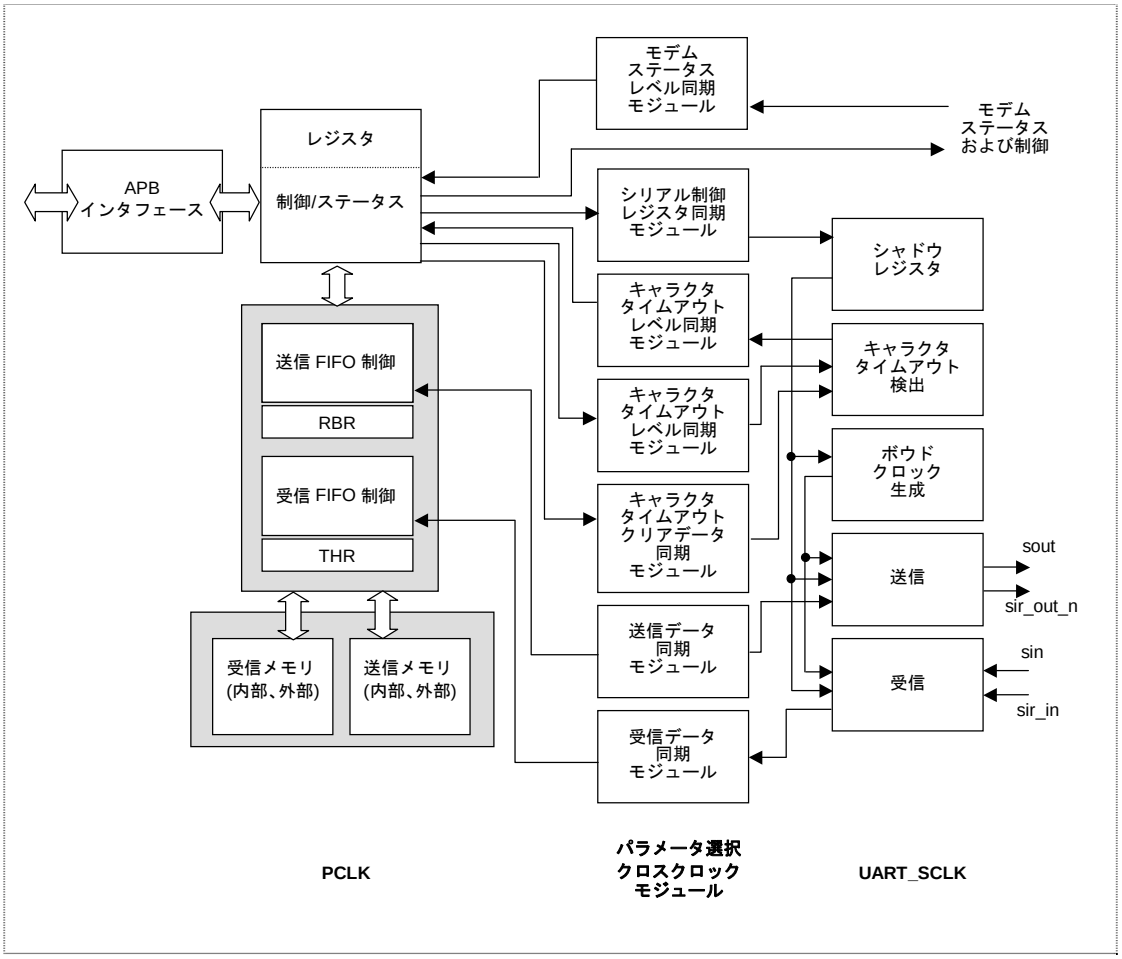


図 16.1　UART1/2/3 ブロック図

16.3 外部端子

UART1/2/3 関連の外部端子は以下の通りです。

端子名	入出力	端子機能	マルチプレクス端子
TXD1	出力	UART1 用送信データ出力	GPIOA0(*)
RXD1	入力	UART1 用受信データ入力	GPIOA1(*)
RTS1	出力	UART1 用送信要求出力	GPIOA2(*)
CTS1	入力	UART1 用送信可能入力	GPIOA3(*)
TXD2	出力	UART2 用送信データ出力	GPIOA4(*)
RXD2	入力	UART2 用受信データ入力	GPIOA5(*)
RTS2	出力	UART2 用送信要求出力	GPIOA6(*)
CTS2	入力	UART2 用送信可能入力	GPIOA7(*)
UART3_CLK	入力	UART3 用クロック入力	GPIOC3(**)
TXD3	出力	UART3 用送信データ出力	GPIOC4(**)
RXD3	入力	UART3 用受信データ入力	GPIOC5(**)
RTS3	出力	UART3 用送信要求出力	GPIOC6(**)
CTS3	入力	UART3 用送信可能入力	GPIOC7(**)

注意(*)：UART1/2 用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1” に設定することにより使用できます。
注意(**)：UART3 用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 2” に設定することにより使用できます。

16.4 レジスタ説明

UART1/2/3 の制御レジスタが配置されているデフォルトのベースアドレスは、それぞれ以下のようになっています。

UART1：0x FFFF_5000
UART2：0x FFFD_4000
UART3：0x FFFD_5000

特に指定のない場合、予約されていないレジスタビットのデフォルト値はすべて“0”です。

注意：UART1/2/3 に対するアクセスは 32 ビット境界のオフセットでアクセスする限り、8bit, 16bit, 32bit のいずれのアクセスサイズでもリード／ライトが可能です。

16.4.1 レジスター覧

表 16.1　レジスター覧
ベースアドレス：0xFFFF_5000　(UART1)
0xFFFD_4000　(UART2)
0xFFFD_5000　(UART3)

アドレスオフセット	DLAB	レジスタ名称	レジスタ略称	初期値	R/W	データアクセスサイズ
0x00	0	受信バッファレジスタ	RBR	0x00	RO	8/16/32bit
0x00	0	送信ホールディングレジスタ	THR	-	WO	8/16/32bit
0x00	1	デバイザラッチ LSB レジスタ	DLL	0x00	R/W	8/16/32bit
0x04	0	割り込みイネーブルレジスタ	IER	0x00	R/W	8/16/32bit
0x04	1	デバイザラッチ MSB レジスタ	DLM	0x00	R/W	8/16/32bit
0x08	-	割り込み識別レジスタ	IIR	0x01	RO	8/16/32bit
0x08	-	FIFO 制御レジスタ	FCR	-	WO	8/16/32bit
0x0C	-	ライン制御レジスタ	LCR	0x00	R/W	8/16/32bit
0x10	-	モデム制御レジスタ	MCR	0x00	R/W	8/16/32bit
0x14	-	ラインステータスレジスタ	LSR	0x60	RO	8/16/32bit
0x18	-	モデムステータスレジスタ	MSR	0xEX	RO	8/16/32bit
0x1C	-	スクラッチレジスタ	SCR	0x00	R/W	8/16/32bit
0x20	-	テスト 0 レジスタ	T0	0x00	R/W	8/16/32bit
0x24	-	テスト 1 レジスタ	T1	0x00	R/W	8/16/32bit
0x28	-	テストステータス 0 レジスタ	TS0	-	RO	8/16/32bit
0x2C	-	テストステータス 1 レジスタ	TS1	0x01	RO	8/16/32bit
0x30	-	テストステータス 2 レジスタ	TS2	0x0F	RO	8/16/32bit
0x3C	-	テストステータス 3 レジスタ	TS3	0x02	RO	8/16/32bit

16.4.2 レジスタアクセスにおける注意事項

UART1/2/3 の制御レジスタは、レジスタ間のアドレスにアクセスしたときの動作は保証されません。例えば、アドレスオフセット 01h にバイトアクセスしたときの動作は保証されません。必ず指定されたアドレスオフセットによるアクセスを行ってください。

20h~3Ch までのレジスタは、UART 自身をデバッグする目的で用意されたレジスタです。デバッグ以外で使用することはできません。またこれらのレジスタは将来において仕様が変更される可能性があります。

16.4.3 レジスタ詳細説明

受信バッファレジスタ (RBR)

UART1/2/3[0x00] DLAB [0] 初期値 = 0x00 Read Only

シリアル受信データ (RBR[7:0])							
7	6	5	4	3	2	1	0

Bits [7:0] : **RBR[7:0] シリアル受信データビット[7:0]**

ライン制御レジスタの DLAB ビット (UART1/2/3[0x0C] Bit 7) が“0”の場合の読み出し時は、このレジスタは受信バッファとして機能します。Bit [7:0]からはシリアルポートが受信したバイトデータを読み出すことができます。なお、読み出されたデータは、ラインステータスレジスタのデータレディビット (UART1/2/3[0x14] Bit 0) が “1”の場合にのみ有効です。このレジスタをアクセスすると、受信 FIFO の先頭データを読み出します。受信 FIFO が満杯の場合、FIFO 内にすでに受信したデータは保持されますが、以降に送られるデータは失われます。

送信ホールディングレジスタ (THR)

UART1/2/3[0x00] DLAB [0] 初期値 = - Write Only

シリアル送信データ (THR [7:0])							
7	6	5	4	3	2	1	0

Bits [7:0] : **THR[7:0] シリアル送信データビット[7:0]**

ライン制御レジスタの DLAB ビット (UART1/2/3[0x0C] Bit 7) が“0”の場合の書き込み時、このレジスタは送信バッファとして機能します。送信 FIFO が満杯になるまで、最大 16 バイトのデータを書き込むことができます。送信 FIFO が満杯になると、それ以降にこのレジスタに書き込んだデータは失われます。

デバイザラッチ LSB レジスタ (DLL)

UART1/2/3[0x00] DLAB [1] 初期値 = 0x00 Read/Write

デバイザラッチ LSB (DL[7:0])							
7	6	5	4	3	2	1	0

Bits [7:0] : **DL[7:0] デバイザラッチ LSB ビット**

ライン制御レジスタの DLAB ビット (UART1/2/3[0x0C] Bit 7) が“1”の場合、このレジスタはボーレートを決定するソースクロックの分周比を設定するために使用されます。設定値は 16 ビットで、このレジスタは下位 8 ビット用です。上位 8 ビットはデバイザラッチ MSB レジスタ (UART1/2/3[0x04]) に設定します。ボーレートは次のように設定されます。

ボーレート = 入力クロック / (16×DL[15:0])

表 16.4 にボーレートと分周値の対応を示します。

割り込みイネーブルレジスタ (IER)

UART1/2/3[0x04] DLAB [0] 初期値= 0x00 Read/Write

7	6	5	4	3	2	1	0
プログラマブル送信ホールディングエンプティ割り込み許可 (EPTBEI)	Reserved (0)			モデムステータス割り込み許可 (EDSSI)	受信ラインステータス割り込み許可 (ELSI)	送信ホールディングエンプティ割り込み許可 (ETBEI)	受信データ可能割り込み許可 (ERBFI)

ライン制御レジスタの DLAB ビット (UART1/2/3[0x0C] Bit 7) が 0 の場合、このレジスタは UART1/2/3 の割り込みイネーブルレジスタとして機能し、5 つの UART1/2/3 割り込み許可/禁止します。Bit 7 は 16550 にはない機能です。16550 と互換性を保つファームウェアを作成する場合は、Bit 7 を使用しないで下さい。

Bit 7 : **EPTBEI プログラマブル送信ホールディングレジスタエンプティ割り込み許可**

FIFO モード時 （UART1/2/3[0x08] Bit 0 = 1）かつ送信ホールディングレジスタエンプティ割り込みが許可されているとき （UART1/2/3[0x04] Bit 1 = 1）にのみ有効になります。

0 (r/w)：割り込み禁止

1 (r/w)：割り込み許可

注意：このビットは 16550 には存在しません。（16550 非互換）

Bit 3 : **EDSSI モデムステータス割り込み許可**

0 (r/w)：割り込み禁止

1 (r/w)：割り込み許可

ただしオート CTS 制御モード（モデム制御レジスタ Bit 5 が“1”で、かつ FIFO 制御レジスタ Bit 0 が“1”に設定されているとき）では、CTS#入力変化によって割り込みが発生しなくなります。

Bit 2 : **ELSI 受信ラインステータス割り込み許可**

0 (r/w)：割り込み禁止

1 (r/w)：割り込み許可

Bit 1 : **ETBEI 送信ホールディングレジスタエンプティ割り込み許可**

0 (r/w)：割り込み禁止

1 (r/w)：割り込み許可

Bit 0 : **ERBFI 受信データ可能割り込み許可**

0 (r/w)：割り込み禁止

1 (r/w)：割り込み許可

各割り込み要因の内容については、表 16.2 を参照してください。

デバイザラッチ MSB レジスタ (DLM)

UART1/2/3[0x04] DLAB [1] 初期値 = 0x00 Read/Write

7	6	5	4	3	2	1	0
デバイザラッチ MSB (DL[15:8])							

Bits [7:0] : **DL[15:8] デバイザラッチ MSB ビット**

ライン制御レジスタの DLAB ビット (UART1/2/3[0x0C] Bit 7) が“1”の場合、このレジスタはボーレートを決定するソースクロックの分周比を設定するために使用されます。設定値は 16 ビットで、このレジスタは上位 8 ビット用です。下位 8 ビットはデバイザラッチ LSB レジスタ (UART1/2/3[0x00]) に設定します。ボーレートは次のように設定されます。

ボーレート = 入力クロック / (16×DL[15:0])

ボーレートと分周値の対応については、

表 16.4 を参照してください。

割り込み識別レジスタ (IIR)

UART1/2/3[0x08]　　初期値 = 0x01　　Read Only

FIFO イネーブル (FFEN [1:0])		Reserved		割り込み ID (IID [3:0])			
7	6	5	4	3	2	1	0

Bits [7:6] : **FFEN [1:0]　FIFO イネーブルステータスビット**

送信用および受信用の FIFO が有効になっているか、無効になっているかを示します。

00：FIFO は無効

11：FIFO は有効

Bits [3:0] : **IID [3:0]　割り込み ID ビット[3:0]**

UART1/2/3 で発生している割り込みソースを識別します。

表 16.2　UART1/2/3 割り込み

IID [3:0]	割り込みの種類	割り込みソース	割り込み ID のリセット方法	優先順位
0001	なし	なし	n/a	n/a
0110	受信ラインステータス割り込み	・オーバーランエラー ・パリティエラー ・フレーミングエラー ・ブレーク受信	・ラインステータスレジスタの読み出し	1 (最高)
0100	受信データ可能割り込み	・受信データ可能	・ラインステータスレジスタの読み出し	2
	受信データトリガレベル到達割り込み	・受信 FIFO 内に受信データをトリガレベルまで受信	・受信 FIFO 内のデータがトリガレベル未満に減少	
1100	キャラクタタイムアウト割り込み	・受信 FIFO に 1 キャラクタ以上のデータがある状態で、4 キャラクタ相当の期間、受信 FIFO からデータが読み出されないか、入力されない	・受信バッファレジスタの読み出し	2
0010	送信ホールディングレジスタエンプティ割り込み	・送信ホールディングレジスタが空	・割り込み識別レジスタの読み出し ・送信ホールディングレジスタへの書き込み	3
	送信データトリガレベル到達割り込み	・送信 FIFO 内のデータをトリガレベルまで送信	・割り込み識別レジスタの読み出し ・送信 FIFO へトリガレベルよりも多いデータの書き込み	
0000	モデムステータス割り込み	・CTS#入力、DSR 入力、RI 入力、DCD 入力のいずれかが変化	・モデムステータスレジスタの読み出し	4 (最低)

FIFO 制御レジスタ (FCR)							
UART1/2/3[0x08] 初期値 = -							Write Only
受信データトリガレベル (RCVRT[1:0])		送信データトリガレベル (XMITT[1:0])		DMA モード選択 (DMAMS)	送信FIFOリセット (XMITFR)	受信FIFOリセット (RCVRFR)	FIFO イネーブル (EFIFO)
7	6	5	4	3	2	1	0

Bits [7:6] : **RCVRT[1:0] 受信データトリガレベル設定**

FIFO を有効に設定している場合（FIFO イネーブル（Bit 0）= 1）に受信 FIFO に何バイトのデータを受信したら受信データトリガレベル到達割り込みを発生させるかを設定します。

00 (w) : 1 バイト
01 (w) : 4 バイト
10 (w) : 8 バイト
11 (w) : 14 バイト

受信 FIFO 内のデータ数がこの設定値以上になると、受信データトリガレベル到達割り込みを発生します。受信バッファレジスタから受信データを読み出し、FIFO 内のデータ数が設定値未満になると、データ受信割り込み要因はクリアされます。

Bits [5:4] : **XMITT[1:0] 送信データトリガレベル設定**

プログラマブル送信ホールディングレジスタエンプティ割り込みが許可されていて (UART1/2/3[0x04] Bit 7=1) 、かつ FIFO を有効に設定している場合（FIFO イネーブル（Bit 0）= 1）に送信 FIFO に残っているデータが何バイト以下のときに送信データトリガレベル到達割り込みを発生させるかを設定します。**このビットは 16550 にはない機能です。16550 と互換性を保つファームウェアを作成する場合は、このビットを使用しないで下さい。**

00 (w) : 0 バイト
01 (w) : 2 バイト
10 (w) : 4 バイト
11 (w) : 8 バイト

送信 FIFO 内のデータ数がこの設定値以下になると、送信データトリガレベル到達割り込みを発生します。送信バッファレジスタに送信データを書き込み、FIFO 内のデータ数が設定値より多くなると、データ送信割り込み要因はクリアされます。

注意 : このビットは 16550 には存在しません。（16550 非互換）

Bit 3 : **DMAMS DMA モード選択ビット**

DMA コントローラ用のステータス信号（内部信号）の動作モードを設定します。

0 (w) : シングルワードモード
1 (w) : マルチワードモード

Bit 2 : **XMITFR 送信 FIFO リセット**

FIFO を有効に設定している場合（FIFO イネーブル（Bit 0）= 1）に送信 FIFO 内のデータをクリアします。シフトレジスタはリセットされません。このビットにライトされた“1”は自動的にリセットされます。

0 (w) : 無効
1 (w) : クリア

注意 : FIFO に残っているデータの中で、2 キャラクタまではクリア前に送信されます。

Bit 1 : **RCVRFR 受信 FIFO リセット**

FIFO を有効に設定している場合（FIFO イネーブル（Bit 0）= 1）に受信 FIFO 内のデータをクリアします。シフトレジスタはリセットされません。このビットにライトされた“1”は自動的にリセットされます。

0 (w) : 無効
1 (w) : クリア

注意 : 受信 FIFO リセットによっては、受信データがあることを示すデータレディビットはクリアされません。受信 FIFO をクリアした後は、必ず受信バッファレジスタ (UART1/2/3[0x00]) を読み出してデータレディビットをクリアしてください。

Bit 0 : **EFIFO　FIFO イネーブル**

送信および受信 FIFO を有効/無効に設定します。

0 (w) : 無効。送受信に FIFO は使用されません。

1 (w) : 有効。FIFO を通して送受信が行われます。

ライン制御レジスタ　(LCR)

UART1/2/3[0x0C]　　初期値 = 0x00　　Read/Write

DLAB	ブレーク制御 (SBRK)	Reserved (0)	偶数パリティ (EPS)	パリティイネーブル (PEN)	ストップビット数 (STB)	ワード長 (WLS[1:0])	
7	6	5	4	3	2	1	0

Bit 7 : **DLAB　デバイザラッチアクセスビット**

UART1/2/3[0x00]、UART1/2/3[0x04]からデバイザラッチ LSB/MSB レジスタにアクセスするか、受信バッファレジスタ、送信ホールディングレジスタ、割り込みイネーブルレジスタにアクセスするか設定します。

0 (r/w) : 受信バッファレジスタ、送信ホールディングレジスタ、割り込みイネーブルレジスタにアクセス

1 (r/w) : デバイザラッチ LSB/MSB レジスタにアクセス

Bit 6 : **SBRK　ブレーク制御**

ブレーク信号の出力を制御します。

0 (r/w) : 通常出力

1 (r/w) : ブレーク信号出力。このビットが“1”に設定されている間（“0”を書き込むまで）、シリアル出力が LOW になります。

Bit 4 : **EPS　偶数パリティ**

偶数/奇数パリティを選択します。

0 (r/w) : 奇数パリティ

1 (r/w) : 偶数パリティ

この設定は、パリティイネーブル (Bit 3) が“1”の場合にのみ有効です。

Bit 3 : **PEN　パリティイネーブル**

パリティチェックとパリティビットの付加を有効/無効に設定します。

0 (r/w) : パリティ無効。受信データはパリティなしとして扱われます。送信データにはパリティビットが付加されません。

1 (r/w) : パリティ有効。パリティビット付きとしてデータを受信し、パリティチェックを行います。送信データにはパリティビットが付加されます。

Bit 2 : **STB　ストップビット数**

ストップビット数を選択します。

0 (r/w) : 1 ビット

1 (r/w) : 2 ビット（データ長が 6、7 または 8 ビットの場合）
1.5 ビット（データ長が 5 ビットの場合）

選択されたストップビットが送信データに付加されます。受信時は選択したストップビット数にかかわらず、最初の 1 ビットのみがチェックされます。

Bits [1:0] : **WLS[1:0]　ワード長ビット[1:0]**

送受信キャラクタ（パリティ、ストップビットを除く）のビット数を選択します。

00 (r/w) : 5 ビット

01 (r/w) : 6 ビット

10 (r/w) : 7 ビット

11 (r/w) : 8 ビット

モデム制御レジスタ (MCR)
UART1/2/3[0x10] 初期値 = 0x00 Read/Write

n/a		オートフロー制御イネーブル(AFCE)	ループバック (LOOP)	アウトプット 2 制御 (OUT2)	アウトプット 1 制御 (OUT1)	RTS 制御 (RTS)	DTR 制御 (DTR)
7	6	5	4	3	2	1	0

Bit 5 : **AFCE オートフロー制御イネーブル**

モデムの CTS#、RTS#信号を使用した、オートフロー制御をイネーブルします。
0 (r/w)：マニュアルフロー制御モード
1 (r/w)：オートフロー制御モード
このビットを“1”に設定し、かつ FIFO 制御レジスタの Bit 0 を“1”にすると、オート CTS 制御モードになり、送信 FIFO にデータが入っているときに CTS#入力が HIGH になると自動的にデータ送信を中断します。CTS#入力が LOW になると自動的にデータ送信を再開します。さらにモデム制御レジスタ Bit 1 を“1”に設定すると、オート RTS 制御モードになり、受信 FIFO が受信データトリガレベルに到達すると、自動的に RTS#出力を HIGH にします。受信 FIFO がエンプティになると、自動的に RTS#出力を LOW にします。
注意： このビットは 16550 には存在しません。（16550 非互換）

Bit 4 : **LOOP ループバック**

モデムのローカルループテストを行います。
0 (r/w)：通常動作
1 (r/w)：ローカルループテスト
このビットを“1”に設定すると、シリアル出力ラインがシリアル入力ラインに接続され、送信データを入力して診断が行えます。
以下にループ･バック･モードと通常モードの相違点を示します。
表に示すように、ループ･バック･モードでは、シリアル･データとモデム制御の出力系が入力系にフィードバックされますので簡単な自己診断が単体で行えます。また、この場合でも割り込み機能やブレーク機能はそのまま動作します。

表 16.3 ループバックモードと通常モードの相違点

相違箇所		ループ･バック･モード時	通常モード時
DTR#端子		常時ハイ･レベル（インアクティブ）を出力	DTR ビットの反転値を出力
RTS#端子		常時ハイ･レベル（インアクティブ）を出力	RTS ビットの反転値を出力
TXD 端子		常時ハイ･レベル（マーク状態）を出力	送信シフトレジスタ（TSR）からシリアルデータを順次出力
モデムステータスレジスタ	CTS ビット	RTS ビットの設定値がリードされます。	CTS#端子の反転値がリードされます。
	DSR ビット	DTR ビットの設定値がリードされます。	DSR#端子の反転値がリードされます。
	RI ビット	OUT1 ビットの設定値がリードされます。	RI#端子の反転値がリードされます。
	DCD ビット	OUT2 ビットの設定値がリードされます。	DCD#端子の反転値がリードされます。
	DCTS ビット	RTS ビットの変化を捕らえます。	CTS#端子の変化を捕らえます。
	DDSR ビット	DTR ビットの変化を捕らえます。	DSR#端子の変化を捕らえます。
	TERI ビット	OUT1 ビットの立ち下がり変化を捕らえます。	RI#端子の変化を捕らえます。
	DDCD ビット	OUT1 ビットの変化を捕らえます。	DCD#端子の変化を捕らえます。
受信シフトレジスタ		送信シフトレジスタ（TSR）からシリアルデータを取り込みます。	RXD 端子からシリアルデータを取り込みます。

Bit 3 : **OUT2　アウトプット 2#制御**

OUT2#出力を直接制御します。ただしループバックテストモード時は常時ハイレベル出力となり、DCD#に相当する内部信号に接続されます。

0 (r/w)：OUT2# = HIGH
1 (r/w)：OUT2# = LOW

注意：このビットはループバックモード時のみ使用可能です。

Bit 2 : **OUT1　アウトプット 1#制御**

OUT1#出力を直接制御します。ただしループバックテストモード時は常時ハイレベル出力となり、RI#に相当する内部信号に接続されます。

0 (r/w)：OUT1# = HIGH
1 (r/w)：OUT1# = LOW

注意：このビットはループバックモード時のみ使用可能です。

Bit 1 : **RTS　RTS#制御**

マニュアルフロー制御モード（モデム制御レジスタ Bit 5 を 0 に設定）時は RTS#出力を直接制御します。

0 (r/w)：RTS# = HIGH
1 (r/w)：RTS# = LOW

オートフロー制御モード（モデム制御レジスタ Bit 5 を 1 に設定）時はオート RTS 制御モードになります。

0 (r/w)：RTS 非制御
1 (r/w)：オート RTS 制御モード

ただしループバックテストモード時は常時ハイレベル出力となり、CTS#に相当する内部信号に接続されます。

Bit 0 : **DTR　DTR#制御**

DTR#出力を直接制御します。ただしループバックテストモード時は常時ハイレベル出力となり、DSR#に相当する内部信号に接続されます。

0 (r/w)：DTR# = HIGH
1 (r/w)：DTR# = LOW

注意：このビットはループバックモード時のみ使用可能です。

ラインステータスレジスタ　(LSR)

UART1/2/3[0x14]　　初期値 = 0x60　　Read Only

受信 FIFO エラー	送信エンプティ (TEMT)	送信ホールディングレジスタエンプティ (THRE)	ブレーク割り込み (BI)	フレーミングエラー (FE)	パリティエラー (PE)	オーバーランエラー (OE)	データレディ (DR)
7	6	5	4	3	2	1	0

Bit 7 : **RCVRE　受信 FIFO エラー**

受信 FIFO でエラー（パリティエラー、フレーミングエラー、ブレーク表示）が発生しているかどうかを示します。このビットは FIFO を有効に設定している場合（FIFO 制御レジスタ UART1/2/3[08h] Bit 0（FIFO イネーブルビット）= 1）にのみ有効で、FIFO を無効に設定している場合（FIFO イネーブルビット = 0）は常に 0 になります。

0： エラーなし
1： エラー発生

次に受信バッファから読み出されるデータ以外に FIFO 内にエラーデータがなければ、このフラグは本レジスタの読み出しにより“0”にリセットされます。

Bit 6 : **TEMT　送信エンプティ**

送信シフトレジスタと送信 FIFO/送信ホールディングレジスタが空であることを示します。

0： 送信データあり
1： 送信データなし

FIFO を有効に設定している場合（FIFO イネーブルビット = 1）、送信 FIFO と送信シフトレジスタの両方が空かどうかを示します。FIFO を無効に設定している場合（FIFO イネーブルビット = 0）

は、送信ホールディングレジスタと送信シフトレジスタの両方が空かどうかを示します。送信データが書き込まれると 0 に戻ります。

Bit 5 : **THRE　送信ホールディングレジスタエンプティ**

このビットは割り込みイネーブルレジスタ UART1/2/3[0x04]Bit 7（EPTBEI ビット）によって動作が変わります。

プログラマブル送信ホールディングレジスタエンプティ割り込みが禁止されている場合（EPTBEI ビット= 0）、送信 FIFO/送信ホールディングレジスタが空であることを示します。
0： 送信データあり
1： 送信データなし
FIFO を有効に設定している場合（FIFO イネーブルビット = 1）、送信 FIFO が空かどうかを示します。FIFO を無効に設定している場合（FIFO イネーブルビット = 0）は、送信ホールディングレジスタが空かどうかを示します。送信ホールディングレジスタエンプティ割り込みを有効に設定している場合は、このフラグがセットされると割り込み要求が発生します。なお、このビットは送信動作によって送信 FIFO/送信ホールディングレジスタが空になった場合にセットされます。リセットや FIFO のクリア等によって空になった場合はセットされません。送信データが書き込まれると 0 に戻ります。

プログラマブル送信ホールディングレジスタエンプティ割り込みが許可されている場合（EPTBEI ビット = 1）、送信 FIFO がフル状態であることを示しています。
0 (r)：送信 FIFO にデータが書き込める状態
1 (r)：送信 FIFO がフル状態
プログラマブル送信ホールディングレジスタエンプティ割り込みが発生したときに、送信 FIFO にデータを書く前にこのビットを読むことにより、送信 FIFO がフルになるまで送信データを書き込めます。このビットとプログラマブル送信ホールディングレジスタエンプティ割り込みを使用することで送信 FIFO は常にデータが入った状態になり、割り込みにすぐに応答できないシステムにおいても効率的なデータ転送が可能となります。

Bit 4 : **BI　ブレーク割り込みフラグ**

ブレーク割り込みを示します。
0： ブレーク割り込みなし
1： ブレーク割り込みあり
1 つのキャラクタに当る受信期間中、入力ラインが 0 になっていると、このフラグがセットされます。
FIFO を有効に設定している場合（FIFO イネーブルビット = 1）、このエラーは FIFO の先頭に置かれているキャラクタで発生していることを示します。受信ラインステータス割り込みを有効に設定している場合は、このフラグがセットされると割り込み要求が発生します。このフラグは本レジスタの読み出しにより“0”にリセットされます。

Bit 3 : **FE　フレーミングエラー**

フレーミングエラーが発生したことを示します。
0： エラーなし
1： エラーあり
受信データに有効なストップビットがない場合に、このフラグがセットされます。FIFO を有効に設定している場合（FIFO イネーブルビット = 1）、このエラーは FIFO の先頭に置かれているキャラクタで発生していることを示します。受信ラインステータス割り込みを有効に設定している場合は、このフラグがセットされると割り込み要求が発生します。このフラグは本レジスタの読み出しにより“0”にリセットされます。

Bit 2 : **PE　パリティエラー**

パリティエラーが発生したことを示します。
0： エラーなし
1： エラーあり
パリティイネーブルビットがセットされている場合に行う受信データのパリティチェックでエラーが発生すると、このフラグがセットされます。FIFO を有効に設定している場合（FIFO イネーブルビット = 1）、このエラーは FIFO の先頭に置かれているキャラクタで発生していることを示します。受信ラインステータス割り込みを有効に設定している場合は、このフラグがセットされ

ると割り込み要求が発生します。このフラグは本レジスタの読み出しにより“0”にリセットされます。

Bit 1 : **OE　オーバーランエラー**

オーバーランエラーが発生したことを示します。
0： エラーなし
1： エラーあり

FIFO を無効に設定している場合（FIFO イネーブルビット = 0）は、受信バッファレジスタ内のカレントキャラクタが読み出される前に次のキャラクタが送られてくるとオーバーランエラーになります。FIFO を有効に設定している場合（FIFO イネーブルビット = 1）は、FIFO が満杯の状態で新しいキャラクタが送られてくるとオーバーランエラーになります。受信ラインステータス割り込みを有効に設定している場合は、このフラグがセットされると割り込み要求が発生します。このフラグは本レジスタの読み出しにより“0”にリセットされます。

Bit 0 : **DR　データレディ**

受信データが存在することを示します。
0： 受信データなし
1： 受信データあり

受信バッファレジスタまたは FIFO 内に 1 キャラクタ以上の有効なデータがある場合、“1”にセットされます。受信データ可能割り込みを有効に設定している場合は、このフラグがセットされると割り込み要求が発生します。このフラグは受信 FIFO から受信データがすべて読み出されると“0”にリセットされます。

モデムステータスレジスタ　(MSR)

UART1/2/3[0x18]　　初期値 = 0xEX　　Read Only

DCD ステータス (DCD) 7	RI ステータス (RI) 6	DSR ステータス (DSR) 5	CTS ステータス (CTS) 4	DCD 変化 (DDCD) 3	RI 立下り変化 (TERI) 2	DSR 変化 (DDSR) 1	CTS 変化 (DCTS) 0

Bit 7 : **DCD　DCD ステータス**

DCD 端子の入力状態を示します。
0： DCD 入力 = HIGH
1： DCD 入力 = LOW

注意： このビットはループバックモード時のみ使用可能です。

Bit 6 : **RI　RI ステータス**

RI 端子の入力状態を示します。
0： RI 入力 = HIGH
1： RI 入力 = LOW

注意： このビットはループバックモード時のみ使用可能です。

Bit 5 : **DSR　DSR ステータス**

DSR 端子の入力状態を示します。
0： DSR 入力 = HIGH
1： DSR 入力 = LOW

注意： このビットはループバックモード時のみ使用可能です。

Bit 4 : **CTS　CTS ステータス**

CTS#端子の入力状態を示します。
0： CTS1#入力 = HIGH
1： CTS#1 入力 = LOW

Bit 3 : **DDCD　DCD 変化**

DCD 端子の入力が、前回の読み出し以降に変化したかどうかを示します。ただしループバックモード設定時 (UART1/2/3[0x10] Bit 4 = 1) では、DCD 端子にかかわらずアウトプット 2 制御ビット (UART1/2/3[0x10] Bit 3) の変化が反映されます。

0： 変化なし
1： 変化あり
このビットは、本レジスタの読み出しによりクリアされます。
注意： このビットはループバックモード時のみ使用可能です。

Bit 2： **TERI　RI 立下り変化**
RI 端子の入力が、前回の読み出し以降に立下り変化したかどうかを示します。ただしループバックモード設定時 (UART1/2/3[0x10] Bit 4 = 1) では、RI 端子にかかわらずアウトプット 1 制御ビット (UART1/2/3[0x10] Bit 2) の立下り変化が反映されます。
0： 立下り変化なし
1： 立下り変化あり
このビットは、本レジスタの読み出しによりクリアされます。
注意： このビットはループバックモード時のみ使用可能です。

Bit 1： **DDSR　DSR 変化**
DSR 端子の入力が、前回の読み出し以降に変化したかどうかを示します。ただしループバックモード設定時 (UART1/2/3[0x10] Bit 4 = 1) では、DSR 端子にかかわらず DTR ビット (UART1/2/3[0x10] Bit 0) の変化が反映されます。
0： 変化なし
1： 変化あり
このビットは、本レジスタの読み出しによりクリアされます。
注意： このビットはループバックモード時のみ使用可能です。

Bit 0： **DCTS　CTS 変化**
CTS#端子の入力が、前回の読み出し以降に変化したかどうかを示します。ただしループバックモード設定時 (UART1/2/3[0x10] Bit 4 = 1) では、CTS#端子にかかわらず RTS 制御ビット (UART1/2/3[0x10] Bit 1) の変化が反映されます。
0： 変化なし
1： 変化あり
このビットは、本レジスタの読み出しによりクリアされます。

スクラッチレジスタ　(SCR)
UART1/2/3[0x1C]　　初期値 = 0x00　　Read/Write

スクラッチビット (SCR [7:0])							
7	6	5	4	3	2	1	0

Bits [7:0]： **SCR [7:0]　スクラッチビット**
ハードウェア動作には影響を与えない汎用レジスタとして使用可能です。

テスト 0 レジスタ　(T0)
UART1/2/3[0x20]　　初期値 = 0x00　　Read/Write

n/a							テストモード
7	6	5	4	3	2	1	0

Bit 0： **テストモード**
テストモードを設定します。テストモードでは、ループバックテストと同等のテストが行えます。
0 (r/w)：通常モード
1 (r/w)：テストモード

テスト 1 レジスタ　(T1)

UART1/2/3[0x24]　　初期値 = 0x00　　Read/Write

n/a				DCD テスト	RI テスト	DSR テスト	CTS テスト
7	6	5	4	3	2	1	0

Bit 3 : **DCD テスト**

テストモード時 (UART1/2/3[0x20] Bit 0=1) に DCD 入力を制御します。

0 (r/w) : ローレベル（アクティブ）を入力

1 (r/w) : ハイレベル（インアクティブ）を入力

Bit 2 : **RI テスト**

テストモード時 (UART1/2/3[0x20] Bit 0=1) に RI 入力を制御します。

0 (r/w) : ローレベル（アクティブ）を入力

1 (r/w) : ハイレベル（インアクティブ）を入力

Bit 1 : **DSR テスト**

テストモード時 (UART1/2/3[0x20] Bit 0=1) に DSR 入力を制御します。

0 (r/w) : ローレベル（アクティブ）を入力

1 (r/w) : ハイレベル（インアクティブ）を入力

Bit 0 : **CTS テスト**

テストモード時 (UART1/2/3[0x20] Bit 0=1) に CTS#入力を制御します。

0 (r/w) : ローレベル（アクティブ）を入力

1 (r/w) : ハイレベル（インアクティブ）を入力

テストステータス 0 レジスタ　(TS0)

UART1/2/3[0x28]　　初期値 = -　　Read Only

n/a				DCD ロウ ステータス	RI ロウ ステータス	DSR ロウ ステータス	CTS ロウ ステータス
7	6	5	4	3	2	1	0

Bit 3 : **DCD ロウステータス**

常に DCD 端子の入力状態を示します。

0 : ローレベル（アクティブ）が入力

1 : ハイレベル（インアクティブ）が入力

Bit 2 : **RI ロウステータス**

常に RI 端子の入力状態を示します。

0 : ローレベル（アクティブ）が入力

1 : ハイレベル（インアクティブ）が入力

Bit 1 : **DSR ロウステータス**

常に DSR 端子の入力状態を示します。

0 : ローレベル（アクティブ）が入力

1 : ハイレベル（インアクティブ）が入力

Bit 0 : **CTS ロウステータス**

常に CTS#端子の入力状態を示します。

0 : ローレベル（アクティブ）が入力

1 : ハイレベル（インアクティブ）が入力

テストステータス 1 レジスタ (TS1)
UART1/2/3[0x2C] 初期値 = 0x01 Read Only

n/a				DCD ステータス	RI ステータス	DSR ステータス	CTS ステータス
7	6	5	4	3	2	1	0

Bit 3 : **DCD ステータス**
UART1/2/3 回路が認識する DCD 信号の入力状態を示します。
0 : ローレベル（アクティブ）が入力
1 : ハイレベル（インアクティブ）が入力

Bit 2 : **RI ステータス**
UART1/2/3 回路が認識する RI 信号の入力状態を示します。
0 : ローレベル（アクティブ）が入力
1 : ハイレベル（インアクティブ）が入力

Bit 1 : **DSR ステータス**
UART1/2/3 回路が認識する DSR 信号の入力状態を示します。
0 : ローレベル（アクティブ）が入力
1 : ハイレベル（インアクティブ）が入力

Bit 0 : **CTS ステータス**
UART1/2/3 回路が認識する CTS#信号の入力状態を示します。
0 : ローレベル（アクティブ）が入力
1 : ハイレベル（インアクティブ）が入力

テストステータス 2 レジスタ (TS2)
UART1/2/3[0x30] 初期値 = 0x0F Read Only

n/a			BAUDOUT ステータス	OUT2 ステータス	OUT1 ステータス	RTS ステータス	DTR ステータス
7	6	5	4	3	2	1	0

Bit 4 : **BAUDOUT ステータス**
UART1/2/3 回路が出力する BAUDOUT 信号の状態を示します。
0 : ローレベルを出力
1 : ハイレベルを出力

Bit 3 : **OUT2 ステータス**
UART1/2/3 回路が出力する OUT2 信号の状態を示します。
0 : ローレベルを出力
1 : ハイレベルを出力

Bit 2 : **OUT1 ステータス**
UART1/2/3 回路が出力する OUT1 信号の状態を示します。
0 : ローレベルを出力
1 : ハイレベルを出力

Bit 1 : **RTS ステータス**
UART1/2/3 回路が出力する RTS#信号の状態を示します。
0 : ローレベル（アクティブ）を出力
1 : ハイレベル（インアクティブ）を出力

Bit 0 : **DTR ステータス**
UART1/2/3 回路が出力する DTR 信号の状態を示します。
0 : ローレベル（アクティブ）を出力
1 : ハイレベル（インアクティブ）を出力

テストステータス 3 レジスタ (TS3)

UART1/2/3[0x3C] 初期値 = 0x02 Read Only

7	6	5	4	3	2	1	0
n/a					TXRDY ステータス	RXRDY ステータス	INTR ステータス

Bit 2 : **TXRDY ステータス**

UART1/2/3 回路が出力する TXRDY 信号の状態を示します。

0 : ローレベル（アクティブ）を出力

1 : ハイレベル（インアクティブ）を出力

Bit 1 : **RXRDY ステータス**

UART1/2/3 回路が出力する RXRDY 信号の状態を示します。

0 : ローレベル（アクティブ）を出力

1 : ハイレベル（インアクティブ）を出力

Bit 0 : **INTR ステータス**

UART1/2/3 回路が出力する INTR 信号の状態を示します。

0 : ローレベル（インアクティブ）を出力

1 : ハイレベル（アクティブ）を出力

16.4.4 ボーレート設定例

UART1/2/3 内でのボーレート設定のためのクロック分周値は、以下の計算式で求められます。

分周値 ＝ UART1/2/3_SCLK 入力クロック周波数(Hz) ÷ ボーレート(bps) ÷ 16

例えば、UART1/2/3 への入力クロック（＝UART1/2/3_SCLK）が 24.543232MHz のときのクロック分周値は下記のようになります。

表 16.4　ボーレートと分周値

ボーレート	×16 クロック 理想値	24.543232MHz UART1/2/3 ソースクロック		
		×16 クロック の分周値	誤差パーセント %	実際の ×16 クロック
300	4800	5113	0.00	4800.2
600	9600	2557	0.02	9598.4
1200	19200	1278	0.02	19204.4
2400	38400	639	0.02	38408.8
4800	76800	320	0.13	76697.6
9600	153600	160	0.13	153395.2
14400	230400	107	0.44	229376.0
19200	307200	80	0.13	306790.4
28800	460800	53	0.49	463079.8
38400	614400	40	0.13	613580.8
57600	921600	27	1.37	909008.6
115200	1843200	13	2.43	1887940.9
125000	2000000	12	2.26	2045269.3
250000	4000000	6	2.26	4090538.7
500000	8000000	3	2.26	8181077.3
750000	12000000	2	2.26	12271616.0
1500000	24000000	1	2.26	24002560.0

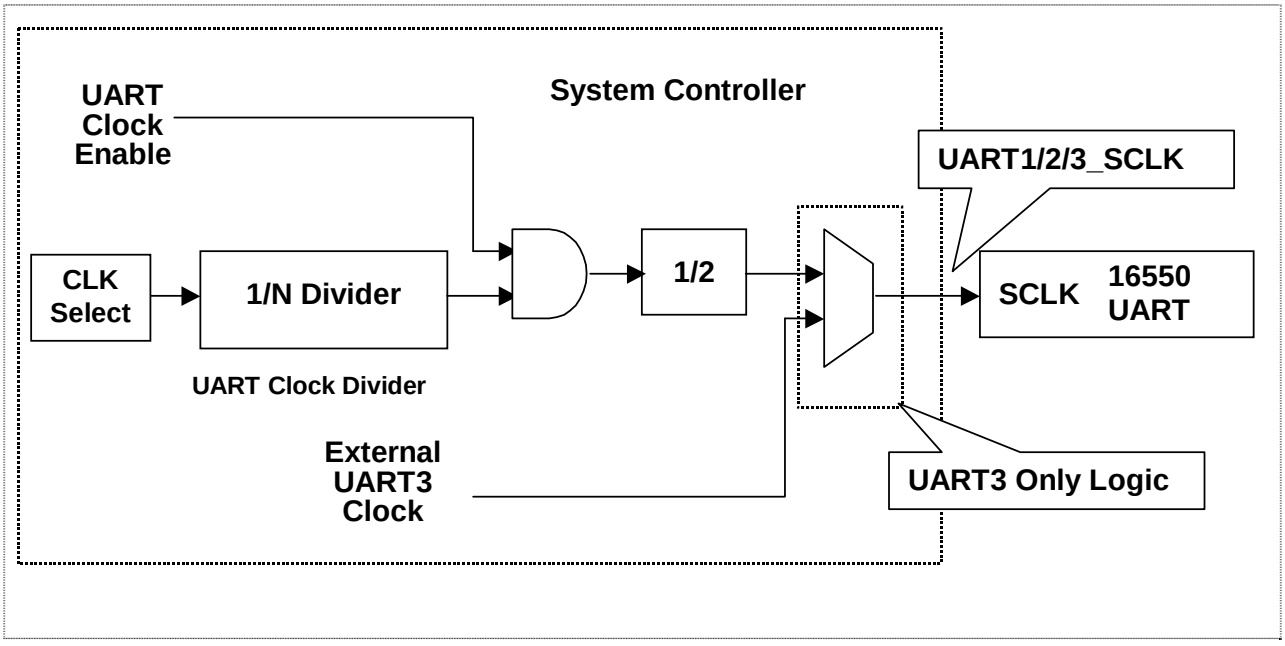


図 16.2　UART1/2/3 クロック概念図

16.5 本 UART1/2/3 の利用制限事項

本チップでは、16550 と互換を保つすべての信号を I/O 端子として用意していないため、回路自体は 16550 の機能のほとんどを有していますが、一部のレジスタに対して利用制限がかかります。以下は本チップで利用制限がかかるレジスタの一覧です。

オフセットアドレス	レジスタビット名称	制限
UART1/2/3[0x10]Bit 0	DTR：DTR#制御	ループバックモードのみ使用可能
UART1/2/3[0x10]Bit 2	OUT1：アウトプット 1#制御	ループバックモードのみ使用可能
UART1/2/3[0x10]Bit 3	OUT2：アウトプット 2#制御	ループバックモードのみ使用可能
UART1/2/3[0x18]Bit 1	DDSR：DSR 変化	ループバックモードのみ使用可能
UART1/2/3[0x18]Bit 2	TERI：RI 立下り変化	ループバックモードのみ使用可能
UART1/2/3[0x18]Bit 3	DDCD：DCD 変化	ループバックモードのみ使用可能
UART1/2/3[0x18]Bit 5	DSR：DSR ステータス	ループバックモードのみ使用可能
UART1/2/3[0x18]Bit 6	RI：RI ステータス	ループバックモードのみ使用可能
UART1/2/3[0x18]Bit 7	DCD：DCD ステータス	ループバックモードのみ使用可能

以下のレジスタ設定は、16550 に存在し、本 UART1/2/3 には存在しません。

オフセットアドレス	レジスタビット名称	制限
UART1/2/3[0x0C]Bit 5	スティックパリティ	常に使用不可

以下のレジスタ設定は、16550 には存在せず、本 UART1/2/3 に存在します。16550 互換のファームウェアを作成する場合は、これらのレジスタを使用しないようにする必要があります。

オフセットアドレス	レジスタビット名称	制限
UART1/2/3[0x04]Bit 7	EPTBEI：プログラマブル送信ホールディングレジスタエンプティ割り込み許可	16550 非互換
UART1/2/3[0x08]Bit [5:4]	XMITT[1:0]：送信データトリガレベル設定	16550 非互換
UART1/2/3[0x10]Bit 5	AFCE：オートフロー制御イネーブル	16550 非互換

17. I²C Single Master Core Module (I2C)

17.1 概要

17.1.1 マスタモード

- I²C バスの I²C シングルマスタモードをサポート
- I²C バスの I²C マルチマスタモードはサポートしていません。
- I²C バス上に複数のスレーブデバイスを接続可能。
- I²C データ転送に関するシフトレジスタのほかに、ソフトウェアによるリード／ライトタイミングを緩和するために、送信および受信用のバッファ（TBUF および RBUF）を持っています。
- バスエラーステータスを検出し、ステータスレジスタにより通知可能
- I²C のクロックウェイト機能をサポート

17.1.2 スレーブモード

- スレーブモードはサポートしていません。

17.2 ブロック図

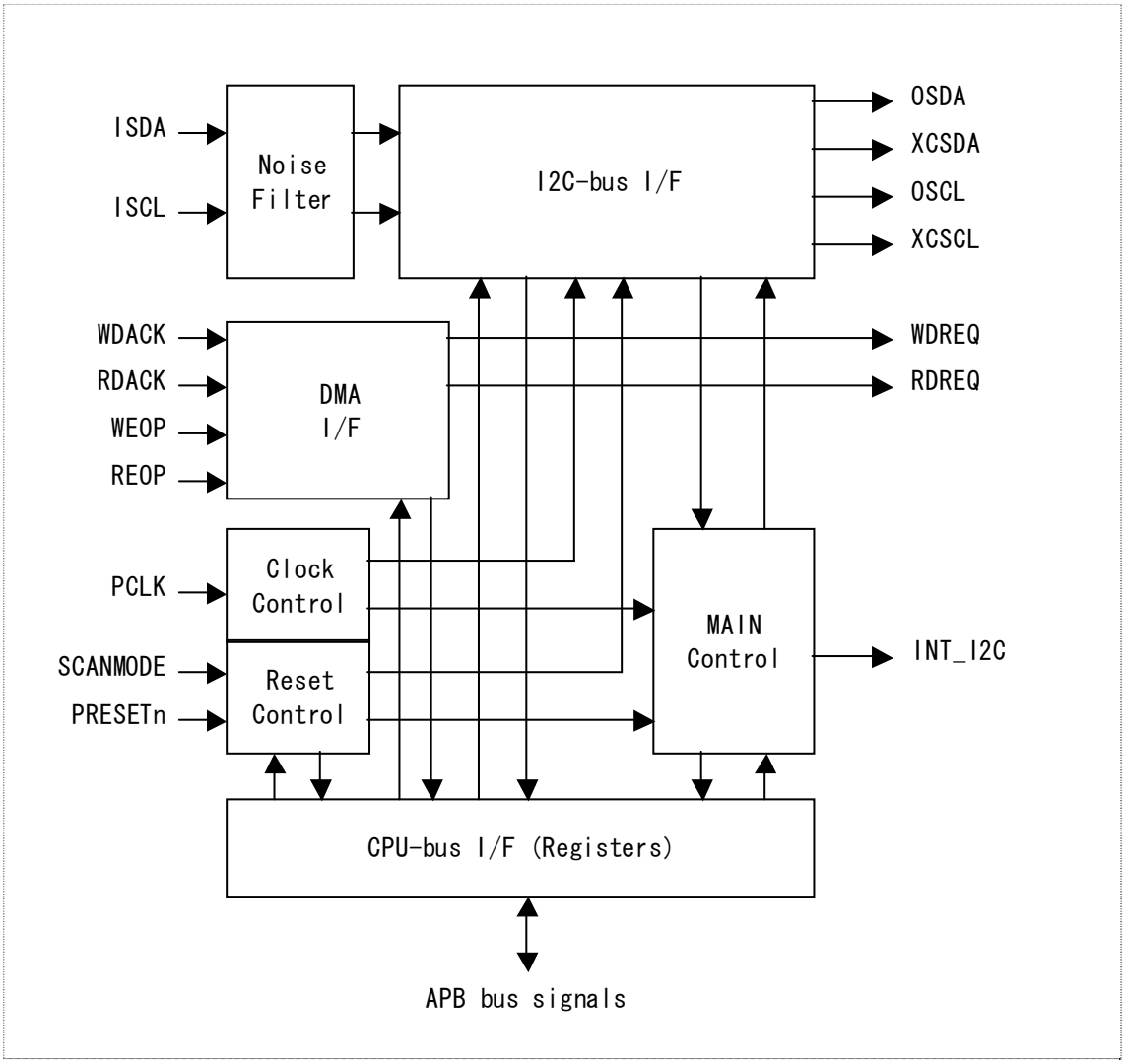


図 17.1 I^2C 内部ブロック図

17.3 外部端子

端子名	入出力	端子機能	マルチプレクス端子*／備考
SCL	入出力	I2C 用クロック入出力	GPIOA6
SDA	入出力	I2C 用データ入出力	GPIOA7

注意(*) : I2C 用の外部端子は GPIO 端子とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 2”に設定することにより使用できます。

17.4 レジスタ

17.4.1 レジスター覧

次の表に掲載されているアドレスオフセット値は I^2C ロジックモジュールに割り当てられているベースアドレスからのオフセットです。これらのレジスタのベースアドレスは、0xFFFE_D000 です。

表 17.1 レジスター覧（ベースアドレス：0xFFFE_D000）

アドレス オフセット	レジスタ名称	初期値	R/W	アクセス サイズ*1
0x00	I^2C 送信データレジスタ	0000 0000b	R/W	8 (16/32)
0x04	I^2C 受信データレジスタ	0000 0000b	RO	8 (16/32)
0x08	I^2C コントロールレジスタ	0000 0000b	R/W	8 (16/32)
0x0C	I^2C バスステータスレジスタ	00xx 0000b *2	RO	8 (16/32)
0x10	I^2C エラーステータスレジスタ	0000 0000b	RO	8 (16/32)
0x14	I^2C 割り込みコントロール／ステータスレジスタ	0000 0000b	R/W	8 (16/32)
0x18	I^2C–BUS サンプルクロック分周設定レジスタ	0000 0000b	R/W	8 (16/32)
0x1C	I^2C SCL クロック分周設定レジスタ	0000 0000b	R/W	8 (16/32)
0x20	I^2C I/O コントロールレジスタ	0000 0000b	R/W	8 (16/32)
0x24	I^2C DMA モードレジスタ	0000 0000b	R/W	8 (16/32)
0x28	I^2C DMA カウンタ値(LSB)レジスタ	0000 0000b	R/W	8 (16/32)
0x2C	I^2C DMA カウンタ値(MSB)レジスタ	0000 0000b	R/W	8 (16/32)
0x30	I^2C DMA ステータスレジスタ	0000 1000b	RO	8 (16/32)
0x34 – 0x38	Reserved	-	-	-

*1：レジスター覧にあるレジスタは 8 ビットで定義されており、通常ファームウェアではこれらのレジスタを 8 ビットアクセスで使用します。しかし、ファームウェアが 16 ビットまたは 32 ビットアクセス命令を使用している場合には 16 ビットまたは 32 ビットアクセスできますが、下位 8 ビットのみ有効な値として使用してください。

*2：このレジスタの Bit 5 (SDA) および Bit 4 (SCL) は外部端子である SDA 端子および SCL 端子をモニタしています。これらの端子の状態により初期値が決まります。通常これらの端子は外部でプルアップされているため、SDA=1、SCL=1 となりますが、外部端子の設定に対応した値となります。

17.4.2 レジスタ詳細説明

以下に各レジスタの詳細を説明します。

I²C 送信データレジスタ

I2C[0x00] 初期値 = 0000 0000b							Read/write
I²C 送信データ TD [7:0]							
7	6	5	4	3	2	1	0

Bits [7:0] : **TD[7:0] I²C 送信データ**

このレジスタは、I²C-BUS 転送の送信データを格納する 8 ビットのバッファです。
マスタからデータを送信するときには、MSB ビットから LSB ビットまでを順に送信します。（下図参照）

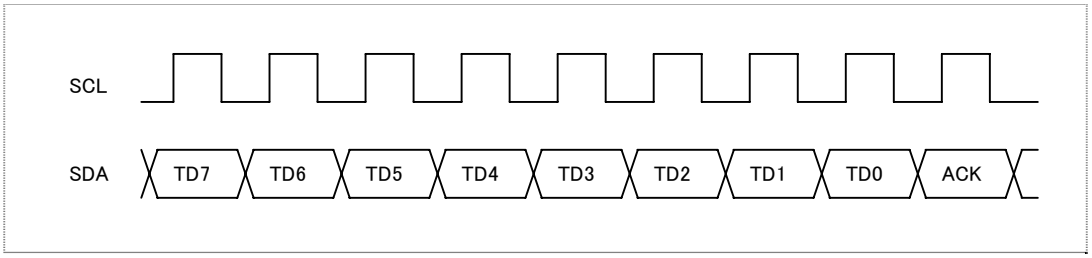


図 17.2 送信時のビット配置

I²C 受信データレジスタ

I2C[0x04] 初期値 = 0000 0000b							Read Only
I²C 受信データ RD [7:0]*							
7	6	5	4	3	2	1	0

Bits [7:0] : **RD[7:0] I²C 受信データ**

このレジスタは、I²C-BUS 転送の受信データを格納する 8 ビットのバッファです。
スレーブデバイスからの送信データを、MSB ビットから LSB ビットまでを順に受信します。（下図参照）

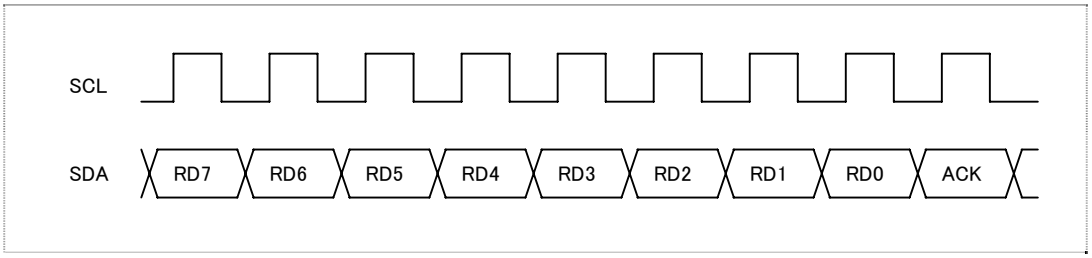


図 17.3 受信時のビット配置

I²C コントロールレジスタ
I2C[0x08]　　初期値 = 0000 0000b　　Read/write

予約 (0)		SR	CLKW*	TACK*	TRNS［2:0］*		
7	6	5	4	3	2	1	0

Note：　(*)はソフトウェアリセットでも初期化されます。

このレジスタは、スタート、ストップコンディションの発生や、データの送受信の開始などを制御します。
TRNS[2:0]がライトされるとその指定された動作を行います。
I²C バスステータスレジスタ I2C[0x0C]ビット 7 の RUN＝1 のときに TRNS[2:0]がライトされても、そのコマンドは実行されません。

Bits [7:6]：　**予約（0）**

Bit 5：　**SR　ソフトウェアリセット**
ソフトウェアで強制的にリセットをかけ、初期化動作を行います。
0：ソフトウェアリセットを解除
1：ソフトウェアリセットをセット

Bit 4：　**CLKW　クロックウェイトモードイネーブル**
クロックウェイト機能を使用するか／しないかを選択します。
0：クロックウェイトモードディセーブル
1：クロックウェイトモードイネーブル

Bit3：　**TACK　データ受信時アクノリッジ設定（受信[Read]モードのみ）**
データ受信時にスレーブデバイスに送信するアクノリッジを設定します。
0：アクノリッジをスレーブデバイスに送信しません。
1：アクノリッジをスレーブデバイスに送信します。
注意：　DMA 転送モード時は、最終バイト転送時のみ本ビットの値（TACK）が使用され、その他は常に“0”が送信されます。

Bits [2:0]：　**TRNS[2:0]　送信コントロールコマンド**
動作指定および開始の指示をします。
001：　I²C スタートコンディション
010：　I²C ストップコンディション
011：　I²C データレシーブ
100：　I²C データトランスファ
101, 110：使用不可
000, 111：エラーステータスレジスタのエラーフラグをクリアします。

I²C バスステータスレジスタ

I2C[0x0C]　初期値 = 00xx 0000b　Read Only

RUN*	予約	SDA	SCL	Using*	Busy*	Error*	Finish*
7	6	5	4	3	2	1	0

Note: (*)：ソフトウェアリセットでも初期化されます。
x：unknown（不定）

このレジスタは、 I²C-BUS の状態を表すレジスタです。

Bit 7 : **RUN　コマンド実行ステータス**
コマンドの動作の有無を示します。
0： I²C-BUS（コマンド）待ち状態
1： I²C-BUS（コマンド）実行中

Bit 6 : **予約**

Bit [5:4] : **SDA, SCL　I²C-BUS モニタ**
現在の SDA, SCL の状態を示します。通常 SDA および SCL は外部でプルアップされているため HIGH になっており、それに対応した値を示します。

Bit 3 : **Using　I²C-BUS 使用**
I²C シングルマスタが I2C-BUS を使用しているかどうかを示します。
0： I²C シングルマスタは I²C-BUS を使用していない状態
1： I²C シングルマスタが I²C-BUS を使用中

Bit 2 : **Busy　I²C-BUS ビジー**
I²C-BUS の使用状態を示します。
0： I²C-BUS BusFree 状態
1： I²C-BUS 使用中

Bit 1 : **Error　エラー発生**
エラー発生の有無を示します。
0： エラーなし
1： エラー状態の発生
このビットは I²C コントロールレジスタのライトでクリアされます。

Bit 0 : **Finish　コマンド実行終了**
コマンドの終了を示します。
0： コマンド未実行、または実行中
1： コマンド実行終了
このビットは、リセットが有効なとき（RESET# = LOW）、I²C コントロールレジスタのビット 5（ソフトウェアリセット）= 1 のとき、または I²C コントロールレジスタのビット[2:0]（TRNS）対する有効なコマンドの書き込みによってクリアされます。

I²C エラーステータスレジスタ I2C[0x10]　　初期値 = 0000 0000b							Read Only
予約			受信 アクノリッジ *	SCL ミスマッチ*	SDA ミスマッチ*	ストップ コンディション *	スタート コンディション *
7	6	5	4	3	2	1	0

Note: (*)：ソフトウェアリセットでも初期化されます。

このレジスタは、エラー状態を表すレジスタです。

Bits [7:5]：　**予約**

Bit 4：　**受信アクノリッジメントエラー**
受信アクノリッジによるエラーの有無を示します。
0： エラーなし
1： エラーあり

Bit 3：　**SCL ミスマッチエラー**
ISCL と OSCL の値不一致によるエラーの有無を示します。
0： 正常動作中
1： クロックの不一致検出あり

Bit 2：　**SDA ミスマッチエラー**
ISDA と OSDA の値不一致によるエラーの有無を示します。
0： 正常動作中
1： データの不一致検出あり

Bit 1：　**ストップコンディション検出**
ストップコンディション発生によるエラーの有無を示します。
0： 正常動作中
1： コマンドによるストップコンディション発生時以外のストップコンディションの検出あり

Bit 0：　**スタートコンディション検出**
スタートコンディション発生によるエラーの有無を示します。
0： 正常動作中
1： コマンドによるスタートコンディション発生時以外での、スタートコンディションの検出あり

注意：これらのエラーステータスは、リセットが有効なとき（RESET# = LOW）、I²C コントロールレジスタのビット５（ソフトウェアリセット）= 1 のとき、または I²C コントロールレジスタのビット[2:0]（TRNS）対する有効なコマンドの書き込みによってクリアされます。

I²C 割り込みコントロール／ステータスレジスタ

I2C[0x14]　　初期値 = 0000 0000b　　Read/Write

予約 (0)				エラーINT ステータス フラグ*	終了 INT ステータス フラグ*	エラーINT イネーブル*	終了 INT イネーブル*
7	6	5	4	3	2	1	0

Note:　(*)：ソフトウェアリセットでも初期化されます。

Bits [7:4]：　**予約（0）**

Bit 3：　**エラー発生による割り込みステータスフラグ**

このビットは Bit 1 = 1 のときのみ有効です。

0：エラー未発生

1：エラー発生による割り込み要求発生

このビットに“1”をライトすることにより、このフラグはクリアされます。

Bit 2：　**コマンド終了割り込みステータスフラグ**

このビットは Bit 0 = 1 のときのみ有効です。

0：コマンド終了割り込みイネーブルビット（Bit 0）= 0、またはコマンドが未終了

1：コマンド終了による割り込み要求発生

このビットに“1”をライトすることにより、このフラグはクリアされます。

Bit 1：　**エラー発生割り込みイネーブル**

エラー発生による割り込みのコントロールをします。

0：エラー発生による割り込み発生ディセーブル

1：エラー発生による割り込み発生イネーブル

Bit 0：　**コマンド終了割り込みイネーブル**

コマンド終了による割り込み発生のコントロールをします。

0：コマンド終了による割り込み発生ディセーブル

1：コマンド終了による割り込み発生イネーブル

I²C-BUS サンプルクロック分周設定レジスタ

I2C[0x18]　　初期値 = 0000 0000b　　Read/Write

予約 (0)				分周比（I²C サンプル）[3:0]			
7	6	5	4	3	2	1	0

マスタクロックを分周することによって I2C-BUS サンプルクロックを生成します。

Bits [7:4]：　**予約ビット（0）**

Bits [3:0]：　**分周比（I²C サンプル） [3:0]**

マスタクロックから I²C-BUS サンプルクロックを生成するための分周比を設定します。

I²C-BUS サンプルクロックの周波数（$f_{I2C\ サンプル}$）は 以下の式で求められます。

Bits[3:0] > 0 のとき：

$f_{I2C\ サンプル} = f_{PCLK} / (4 * m)$［Hz］

Bits[3:0] = 0 のとき：

$f_{I2C\ サンプル} = f_{PCLK} / 2$［Hz］

注意：　上記式において、m = 分周比（I²C サンプル）[3:0]

I²C SCL クロック分周設定レジスタ

I2C[0x1C]　初期値 = 0000 0000b　Read/Write

予約 (0)					分周比(SCL) [2:0]		
7	6	5	4	3	2	1	0

SCL を生成するための分周比を設定するレジスタです。

Bits [7:3]：　**予約（0）**

Bit [2:0]：　**分周比(SCL) [2:0]**

I²C-BUS サンプルクロックから SCL を生成するための分周比を設定します。

I²C-BUS 転送時の SCL の周波数（f_{SCL}）は以下の式で決定します。

分周比（I²C サンプル）[3:0] > 0 のとき：

$f_{SCL} = f_{I2C\ サンプル} / (2^n * 4)$

$= f_{PCLK} / \{(4 * m) * (2^n * 4)\}$

$= f_{PCLK} / (16 * m * 2^n)$ ［Hz］

分周比（I²C サンプル）[3:0] = 0 のとき：

$f_{SCL} = f_{PCLK} / \{2 * (2^n * 4)\}$

$= f_{PCLK} / (8 * 2^n)$ ［Hz］

注意：上記式において、“m”および“n”を以下のように指定：

m = 分周比（I²C サンプル）[3:0]

（I²C-BUS サンプルクロック分周設定レジスタ Bits[3:0] 参照）

n = 分周比(SCL) [2:0]

I²C I/O コントロールレジスタ

I2C[0x20]　初期値 = 0000 0000b　Read/Write

予約 (0)		High Drive SDA	Sample SDA	予約 (0)		High Drive SCL	Sample SCL
7	6	5	4	3	2	1	0

出力モード選択と、ノイズフィルタの有効、無効を決定するレジスタです。

Bits [7:6]：　**予約（0）**

Bit 5：　**SDA の HIGH ドライブイネーブル**

0：“SDA=1”は IC 外部のプルアップ抵抗により制御

1：“SDA=1”は本 IC 自身が“High”をドライブすることにより制御

Bit 4：　**SDA のサンプリングイネーブル**

0：SDA 入力として、I²C-BUS サンプルクロックにより 1 データサンプリング

1：SDA 入力として、I²C-BUS サンプルクロックにより 2 データサンプリング

注意：本ビットは“0”でご使用ください。

Bits [3:2]：　**予約**（0）

Bit 1：　**SCL の HIGH ドライブイネーブル**

0：“SCL=1”は IC 外部のプルアップ抵抗により制御

1：“SCL=1”は本 IC 自身が“High”をドライブすることにより制御

Bit 0 : **SCL のサンプリングイネーブル**

0： SCL 入力として、I²C-BUS サンプルクロックにより 1 データサンプリング

1： SCL 入力として、I²C-BUS サンプルクロックにより 2 データサンプリング

注意： 本ビットは“0”でご使用ください。

I²C DMA モードレジスタ

I2C[0x24] 初期値 = 0000 0000b Read/Write

予約 (0)						DMA_MODE [1:0]	
7	6	5	4	3	2	1	0

Bits[7:2] : **予約（0）**

Bits [1:0] : **DMA_MODE[1:0] DMA モード設定**

00：DMA 転送を使用しない。

01：シングルアドレスモード

10：デュアルアドレスモード（EOP なし）。ＤＭＡカウンタ使用。

11：デュアルアドレスモード（EOP あり）。

転送中（Start(restart)から Stop まで）の DMA モードの ON/OFF の切り替えは可能ですが、使用する DMA モード（Single、Dual（EOP あり）、Dual（EOP なし））の切り替えはしないでください。

バースト転送はサポートしておりません。

I²C DMA カウンタ値（LSB）レジスタ

I2C[0x28] 初期値 = 0000 0000b Read/Write

DMA カウンタ値(LSB)							
7	6	5	4	3	2	1	0

ライト時：DMA カウンタ値の下位バイト[7:0]をセットするレジスタです。
リード時：DMA カウンタ値の下位バイト[7:0]を返します。

I²C DMA カウンタ値（MSB）レジスタ

I2C[0x2C] 初期値 = 0000 0000b Read/Write

DMA カウンタ値(MSB)							
7	6	5	4	3	2	1	0

ライト時：DMA カウンタ値の上位バイト[15:8]をセットするレジスタです。
リード時：DMA カウンタ値の上位バイト[15:8]を返します。

I²C DMA ステータスレジスタ

I2C[0x30]　　初期値 = 0000 1000b　　Read Only

予約				TBUF Empty*	RBUF Update*	RDREQ Monitor*	WDREQ Monitor*
7	6	5	4	3	2	1	0

Note:　(*)：ソフトウェアリセットでも初期化されます。

Bits[7:4] :　**予約**

Bit 3 :　**TBUF_Empty　送信バッファエンプティ**

I²C シングルマスタコアの送信データバッファ (TBUF) の書き込みが可能かどうかを示します。

0： 送信バッファに未送信のデータあり

1： 送信バッファエンプティ。書き込み可能

Bit 2 :　**RBUF_Update　受信バッファ更新**

I²C 受信データバッファ (RBUF) のデータが更新されたものかどうかを示します。

0： 受信バッファが未更新

1： 受信バッファのデータが更新済み

Bit 1 :　**Reserved（RDREQ Monitor　RDREQ 信号モニタ）**

RDREQ 信号の状態を示します。

0： RDREQ 端子の信号が Low であることを示します。

1： RDREQ 端子の信号が High であることを示します。

注意： このビットは状態が変化しますが、使用はできません。

Bit 0 :　**Reserved（WDREQ Monitor　WDREQ 信号モニタ）**

WDREQ 信号の状態を示します。

0： WDREQ 端子の信号が Low であることを示します。

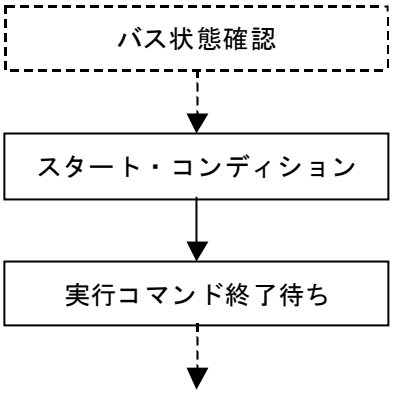
1： WDREQ 端子の信号が High であることを示します。

注意： このビットは状態が変化しますが、使用はできません。

17.5 動作説明（バス・コントロール・コマンド使用例）

以下は、本モジュールによる I2C バスをコントロールする例です。実際は、状態確認やエラー発生時の対応等が必要となる場合があります。また、具体的なコントロール方法は、スレーブ機器の仕様に依存します。

17.5.1 スタート（S）フロー例

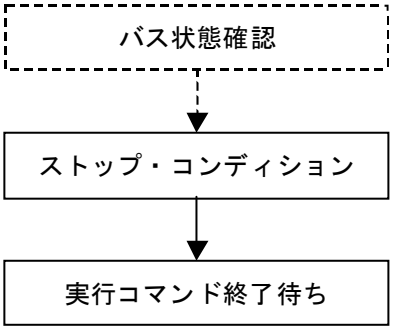


I2C[0x0C] bit2が“0”であることを確認

スタートコンディション設定：
I2C[0x08] bits[2:0] に “001”をセット

I2C[0x0C] bit 0 が“1”になるまで待つ

17.5.2 ストップ（P）フロー例

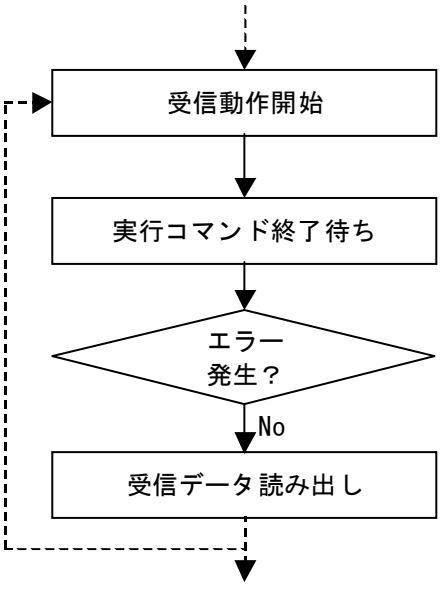


I2C[0x0C] bit3が“1”であることを確認

ストップコンディション設定：
I2C[0x08] bits[2:0] に“010”をセット

I2C[0x0C] bit 0 が“1”になるまで待つ

17.5.3 レシーブ（R）フロー例



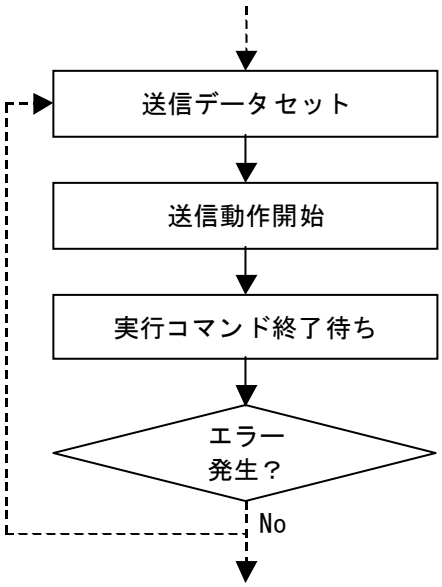
レシーブ設定：
I2C[0x08] bits[2:0] に“011”をセット

I2C[0x0C] bit 0 が“1”になるまで待つ

I2C[0x0C] bit1をチェック
“0”はエラーなし、“1”はエラー発生

I2C[0x04] の受信データ読み出し

17.5.4 トランスファ（T）フロー例



I2C[0x00] に送信データをセット

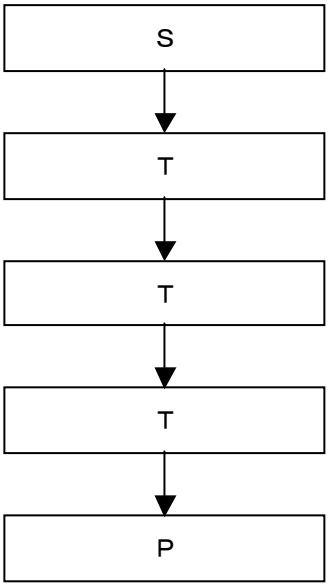
トランスファ設定：
I2C[0x08] bits[2:0] に“100”をセット

I2C[0x0C] bit 0 が“1”になるまで待つ

I2C[0x0C] bit1をチェック
“0”はエラーなし、“1”はエラー発生

17.5.5 スレーブ機器への書込みシーケンス例

図中の S／P／R／T は、前述のスタート／ストップ／レシーブ／トランスファの各フローに対応します。



スタートコンディション発生

スレーブアドレスとWriteモードを送信

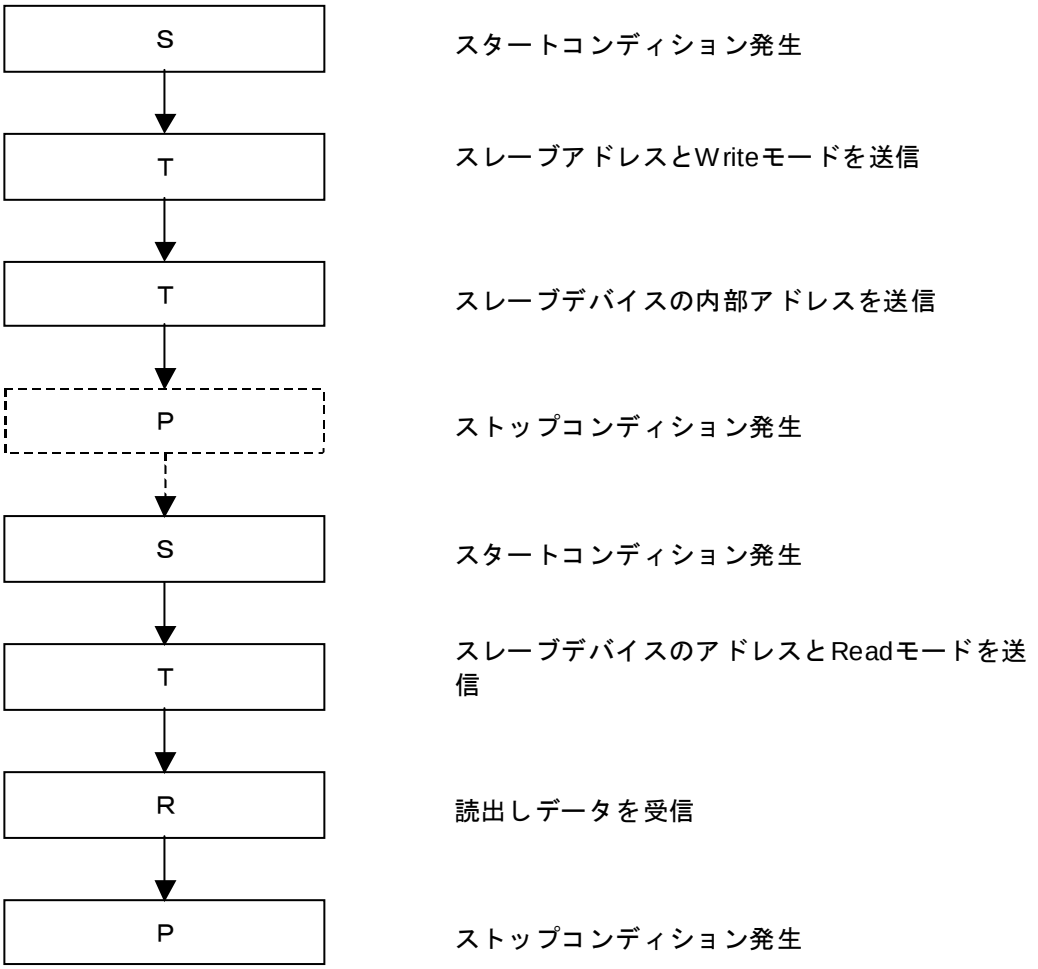
スレーブデバイスの内部アドレスを送信

書込みデータを送信

ストップコンディション発生

17.5.6 スレーブからの読出しシーケンス例

図中の S／P／R／T は、前述のスタート／ストップ／レシーブ／トランスファの各フローに対応します。



17.6 本 I2C Single Master Core Module (I2C)の利用制限事項

以下は本チップで利用制限がかかるレジスタの一覧です。

オフセットアドレス	レジスタビット名称	制限
I2C[0x30] Bit 0	WDREQ 信号モニタ	使用不可
I2C[0x30] Bit 1	RDREQ 信号モニタ	使用不可

18. I²S (I2S)

18.1 概要

I²S は Philips 社が策定した I²S 規格に準拠しています。このモジュールは主に音声／オーディオデータの通信に使用します。I²S は 2Channel の通信ができるようになっています。Channel 毎に送信か受信かを選択できますので、例えば Audio デバイスからの音声／オーディオデータの受信と Audio デバイスへの音声／オーディオデータの送信を同時に行えるだけでなく、異なる 2 つの Audio デバイスから同時に受信することもできます。

18.1.1 機能

I²S が実現している機能は以下のとおりです。

- マスターモード（SCK と WS を出力）とスレーブモード（SCK と WS を入力）が選択可能
- 送信モード（SD を出力）と受信モード（SD を入力）を選択可能
- 16 ビット、14 ビット、8 ビットのデータ幅に対応
- ステレオ、モノラルを選択可能
- 32fs、64fs、128fs、256fs のフレームサイクルに対応
- ソースクロックの 1/2~1/512 まで 256 段階でクロック分周選択可能（マスターモードのみ）
- DMA 対応
- クロック共有機能（2Channel で使用するクロックを共通にする）
- FIFO オーバーフロー、アンダーフロー検出
- FIFO の 6 種類の状態による割り込み発生
- 0 出力機能（送信モード、FIFO アンダーフロー時）
- モノラルステレオ変換（送信モードのみ）

18.2 ブロック図

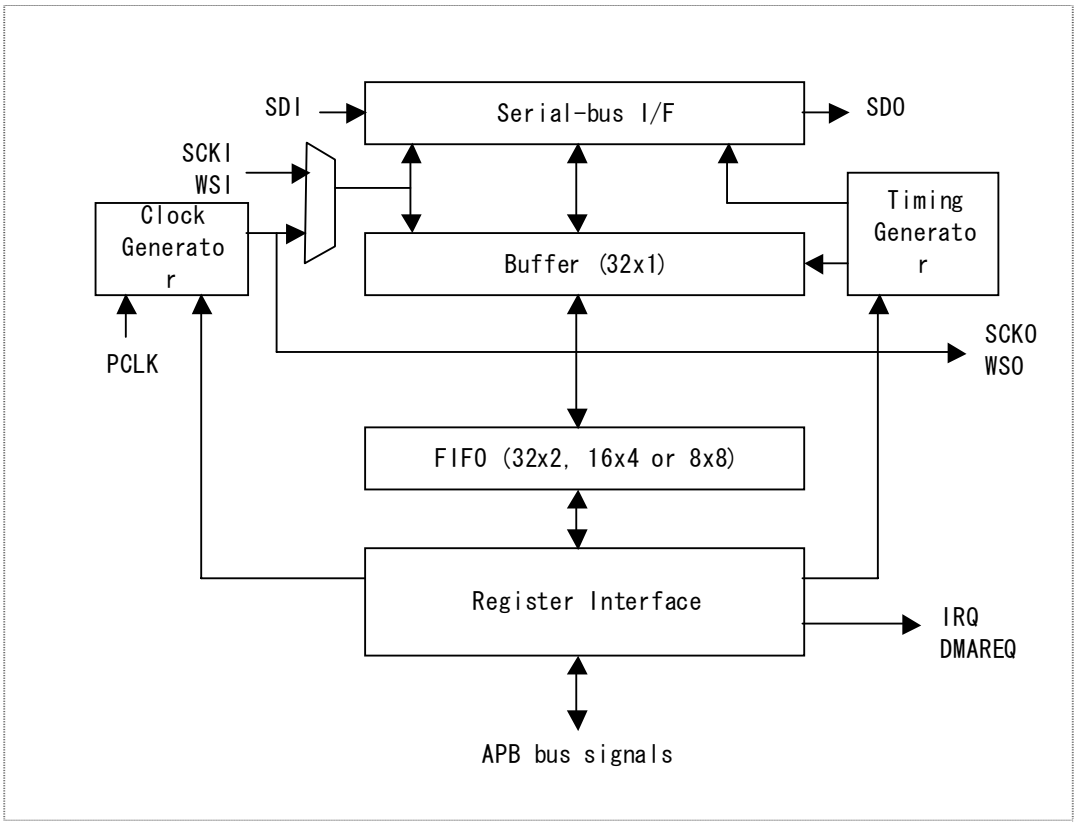


図 18.1　I2S ブロック図

18.3 外部端子

I^2S に関連する外部端子は以下の通りです。

端子名	入出力	端子機能	マルチプレクス端子／備考
I2S_SCK (I2S0_SCK)	入出力	I2S0 用シリアルクロック	GPIOB1 (*)
I2S_WS (I2S0_WS	入出力	I2S0 用ワードセレクト	GPIOB0 (*)
I2S_SDO (I2S0_SD)	入出力	I2S0 用シリアルデータ	GPIOB2 (*)
(I2S1_SCK)	入出力	I2S1 用シリアルクロック	GPIOA3 (**)
(I2S1_WS)	入出力	I2S1 用ワードセレクト	GPIOA2 (**)
I2S_SDI (I2S1_SD)	入出力	I2S1 用シリアルデータ	GPIOB3 (*)

注意(*)： I2S 用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1”に設定することにより使用できます。

注意(**)： I2S1 用の WS/SCK 外部端子は、I2S0 を送信モード/I2S1 を受信モードに設定することで外部接続は必要ありません。接続を行う場合は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 2”に設定することにより使用できます。

18.4 レジスタ説明

18.4.1 レジスター覧

I^2S の制御レジスタが配置されているデフォルトのベースアドレスは、0xFFFE_E000 です。
特に指定のない場合、予約されていないレジスタビットのデフォルト値はすべて“0”です。

表 18.1 I2S[1:0]レジスター覧

アドレス オフセット	レジスタ名称	初期値	R/W	データ アクセスサイズ
I2S0 制御レジスタ　ベースアドレス：0xFFFE_E000				
0x00	I2S0 制御レジスタ	0x0000	R/W	16/32
0x04	I2S0 クロック分周レジスタ	0x0000	R/W	16/32
0x08	I2S0 送受信ポートレジスタ	-	R/W	8/16/32
0x10	I2S0 割り込みステータスレジスタ	0x0000	R/W	16/32
0x14	I2S0 割り込みロウステータスレジスタ	0x0009	RO	16/32
0x18	I2S0 割り込みイネーブルレジスタ	0x0000	R/W	16/32
0x1C	I2S0 カレントステータスレジスタ	0x0009	RO	16/32
I2S1 制御レジスタ　ベースアドレス：0xFFFE_E000				
0x40	I2S1 制御レジスタ	0x0000	R/W	16/32
0x44	I2S1 クロック分周レジスタ	0x0000	R/W	16/32
0x48	I2S1 送受信ポートレジスタ	-	R/W	8/16/32
0x50	I2S1 割り込みステータスレジスタ	0x0000	R/W	16/32
0x54	I2S1 割り込みロウステータスレジスタ	0x0009	RO	16/32
0x58	I2S1 割り込みイネーブルレジスタ	0x0000	R/W	16/32
0x5C	I2S1 カレントステータスレジスタ	0x0009	RO	16/32

18.4.2 レジスタ詳細説明

I2S[1:0]制御レジスタ

I2S0[0x00], I2S1[0x40]　　初期値 = 0x0000　　Read/Write

n/a			CNVM2S	FRAMECYC [1:0]		CLKOUTEN	CLKSEL
15	14	13	12	11	10	9	8
SFTRST (WO)	DATAWIDTH [1:0]		MONO/ STEREO	DMAEN	TX/RX	MST/SLV	I2SEN
7	6	5	4	3	2	1	0

Bit 12 :　**CNVM2S　モノラル・ステレオ変換（送信モードのみ）**

モノラルデータをステレオデータに変換して送信します。具体的にはLチャネルで送信されるモノラルデータをRチャネルでも送信します。

0： モノラル・ステレオ変換無効
1： モノラル・ステレオ変換有効

Bits [11:10] :　**FRAMECYC [1:0]　フレームサイクル選択（マスターモードのみ）**

フレームサイクル数を制御します。このレジスタビットはマスターモードのみ有効です。
スレーブモードでは、データ幅 x 2 以上 256fs 以下の偶数となるフレームサイクル数が使用できます。

00：32fs
01：64fs
10：128fs
11：256fs

Bit 9 :　**CLKOUTEN　クロック出力イネーブル（マスターモードのみ）**

クロック出力を制御します。

0： クロック出力ディゼーブル
1： クロック出力イネーブル

Bit 8 :　**CLKSEL　クロック選択**

クロックソースを選択します。

0： 外部から来る SDI/WSI（スレーブモード）または自生成クロック／ワードセレクト（マスターモード）を使用します
1： もう一つの Channel が使用するクロック及びワードセレクトを共用して使用します

このレジスタビットは、I2S クロックと I2S ワードセレクト信号を他の Channel の I2S を共有するときに使用します。

注意： このレジスタビットを“1”にするときは、同レジスタの Bit 1 を“0”にし、スレーブモードにしてください。クロック共有機能はマスターモードで使用できません。

Bit 7 :　**SFTRST　ソフトウェアリセット (Write Only)**

FIFO データ、シフトレジスタおよび内部制御回路をリセットします。マスターモード時に使用するクロック生成回路はリセットされません。同じくマスターモード時に使用するワードセレクト生成回路は、受信モード時はリセットされませんが、送信モード時は出力が“1”になるようにリセットされます。

0： n/a
1： ソフトウェアリセット

Bits [6:5] :　**DATAWIDTH [1:0]　データビット幅**

データのビット幅を選択します。

00：16 ビット
01：14 ビット
10：8 ビット
11：Reserved

Bit 4 : **MONO/ STEREO　モノラル／ステレオ選択**

データがステレオタイプかモノラルタイプかを選択します。

0： ステレオタイプ

1： モノラルタイプ

Bit 3 : **DMAEN　DMA イネーブル**

DMA を使用するか、使用しないかを選択します。

0： DMA ディゼーブル（DMA リクエストを発行しない）

1： DMA イネーブル（DMA リクエストを DMA コントローラ 1 に発行する）

Bit 2 : **TX/RX　送信／受信選択**

I2S を送信モードで使用するか、受信モードで使用するかを選択します。

0： 受信モード（データを入力）

1： 送信モード（データを出力）

Bit 1 : **MST/SLV　マスター／スレーブ選択**

I2S をマスターモードで使用するか、スレーブモードで使用するかを選択します。

0： スレーブモード（クロックとワードセレクトを入力）

1： マスターモード（クロックとワードセレクトを出力）

Bit 0 : **I2SEN　I2S イネーブル**

I2S モジュールのイネーブル制御を行います。

0： I2S ディゼーブル

1： I2S イネーブル

I2S[1:0]クロック分周レジスタ

I2S0[0x04], I2S1[0x44]　　初期値 = 0x0000　　Read/Write

n/a							
15	14	13	12	11	10	9	8
CLKDIV [7:0]							
7	6	5	4	3	2	1	0

Bit[7:0] : **CLKDIV [7:0]　クロック分周（マスターモードのみ）**

マスターモード時に出力クロックのソースクロックからの分周数を設定します。

クロック分周数 ＝ (CLKDIV + 1) x 2

この設定によりサンプリング周波数は以下のようになります。

サンプリング周波数 ＝

ソースクロック周波数 / (クロック分周数 x フレームサイクル)

S2S65A00 ではソースクロック周波数は、システムクロック周波数と同じになります。

I2S[1:0]送受信ポートレジスタ

I2S0[0x08], I2S1[0x48]　　初期値 = -　　Read/Write

TXD/RXD [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXD/RXD [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit[31:0]：　**TXD/RXD [31:0]　送受信ポート[31:0]**

受信モード時

リード：FIFO に溜まった I2S のデータを読み出します。

ライト：n/a

送信モード時

リード：n/a

ライト：FIFO に I2S のデータを書きます。

1 回のアクセスで読み書きできるデータは、データ幅及びステレオ／モノラル設定によって異なります。

16 ビットステレオ：
- TXD/RXD[31:16] 右データ
- TXD/RXD[15:0] 左データ

14 ビットステレオ：
- TXD/RXD[31:30]　0 データ
- TXD/RXD[29:16] 右データ
- TXD/RXD[15:14]　0 データ
- TXD/RXD[13:0] 左データ

8 ビットステレオ
- TXD/RXD[31:16] 無効データ
- TXD/RXD[15:8] 右データ
- TXD/RXD[7:0] 左データ

16 ビットモノラル：
- TXD/RXD[31:16] 無効データ
- TXD/RXD[15:0] モノラルデータ

14 ビットモノラル：
- TXD/RXD[31:16] 無効データ
- TXD/RXD[15:14]　0 データ
- TXD/RXD[13:0] モノラルデータ

8 ビットモノラル：
- TXD/RXD[31:8] 無効データ
- TXD/RXD[7:0] モノラルデータ

注意：送受信ポートレジスタはサンプリング単位でデータを扱っていますので、1 回のレジスタアクセスで 1 つのサンプリングデータがすべて読み書きされなければなりません。例えば 16 ビットステレオの場合、左データのみを 16 ビットアクセスで読み出したとしても、右データは次のデータに更新されてしまいますので、32 ビットアクセスで左右両方のデータを一度で読み出すようにしてください。

I2S[1:0]割り込みステータスレジスタ

I2S0[0x10], I2S1[0x50]　　初期値 = 0x0000　　Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a		OVERFLOWFLG	UNDERFLOWFLG	NOTFULLFLG	NOTEMPTYFLG	FULLFLG	EMPTYFLG
7	6	5	4	3	2	1	0

このレジスタは、ロウ割り込みステータスレジスタ (I2S0[0x14]、I2S1[0x54]) と割り込みイネーブルレジスタ (I2S0[0x18]、I2S1[0x58]) とをビット AND した結果を示しています。

Bit 5 :　**OVERFLOWFLG　FIFO オーバーフロー割り込みフラグ**

FIFO が 1 度でもオーバーフローになったかどうかを示します。

ライト 0 :　n/a
ライト 1 :　フラグのクリアを試みます
リード 0 :　このフラグがイネーブルになっていないか、FIFO が 1 度もオーバーフローになっていません
リード 1 :　FIFO が 1 度はオーバーフローになりました

このフラグが“1”を書いたことでクリアされるかどうかはモードによって異なります。

送信モード : 必ずクリアされます。オーバーフローした分のデータはオーバーフローを起こした時点で無くなっています。すでに FIFO になったデータはそのままです。
受信モード : FIFO がフルではない状態が I2S クロック 1 クロック分に期間続いた後に試みた場合はクリアされます。またソフトウェアリセットを実行した後に試みた場合もクリアされます。

Bit 4 :　**UNDERFLOWFLG　FIFO アンダーフロー割り込みフラグ**

FIFO が 1 度でもアンダーフローになったかどうかを示します。

ライト 0 :　n/a
ライト 1 :　フラグのクリアを試みます
リード 0 :　このフラグがイネーブルになっていないか、FIFO が 1 度もアンダーフローになっていません
リード 1 :　FIFO が 1 度はアンダーフローになりました

このフラグが“1”を書いたことでクリアされるかどうかはモードによって異なります。

送信モード : FIFO にデータを書いた後 I2S クロック 1 クロック分後に試みた場合はクリアされます。またソフトウェアリセットを実行した後に試みた場合もクリアされます。
受信モード : 必ずクリアされます。アンダーフローで取得したデータの値は保証されません。

Bit 3 :　**NOTFULLFLG　FIFO ノットフル割り込みフラグ**

FIFO が 1 度でもフルではない状態になったかどうかを示します。

ライト 0 :　n/a
ライト 1 :　フラグのクリアを試みます
リード 0 :　このフラグがイネーブルになっていないか、FIFO が 1 度もフルではない状態になっていません
リード 1 :　FIFO が 1 度はフルではない状態になりました

このフラグが“1”を書いたことでクリアされるかどうかは FIFO の状態によって異なります。

FIFO がフル状態：フラグはクリアされます。
FIFO がフルではない状態：フラグはクリアされません。

Bit 2 : **NOTEMPTYFLG　FIFO ノットエンプティ割り込みフラグ**

FIFO が 1 度でもエンプティではない状態になったかどうかを示します。

ライト 0： n/a
ライト 1： フラグのクリアを試みます
リード 0： このフラグがイネーブルになっていないか、FIFO が 1 度もエンプティではない状態になっていません
リード 1： FIFO が 1 度はエンプティではない状態になりました

このフラグが“1”を書いたことでクリアされるかどうかは FIFO の状態によって異なります。

FIFO がエンプティ状態：フラグはクリアされます。
FIFO がエンプティではない状態：フラグはクリアされません。

Bit 1 : **FULLFLG　FIFO フル割り込みフラグ**

FIFO が 1 度でもフル状態になったかどうかを示します。

ライト 0： n/a
ライト 1： フラグのクリアを試みます
リード 0： このフラグがイネーブルになっていないか、FIFO が 1 度もフル状態になっていません
リード 1： FIFO が 1 度はフル状態になりました

このフラグが“1”を書いたことでクリアされるかどうかは FIFO の状態によって異なります。

FIFO がフル状態：フラグはクリアされません。
FIFO がフルではない状態：フラグはクリアされます。

Bit 0 : **EMPTYFLG　FIFO エンプティ割り込みフラグ**

FIFO が 1 度でもエンプティ状態になったかどうかを示します。

ライト 0： n/a
ライト 1： フラグのクリアを試みます
リード 0： このフラグがイネーブルになっていないか、FIFO が 1 度もエンプティ状態になっていません
リード 1： FIFO が 1 度はエンプティ状態になりました

このフラグが“1”を書いたことでクリアされるかどうかは FIFO の状態によって異なります。

FIFO がエンプティ状態：フラグはクリアされません。
FIFO がエンプティではない状態：フラグはクリアされます。

I2S[1:0]ロウ割り込みステータスレジスタ

I2S0[0x14], I2S1[0x54]　　初期値 = 0x0009　　Read Only

n/a							
15	14	13	12	11	10	9	8
n/a		RAW OVERFLOWFLG	RAW UNDERFLOWFLG	RAW NOTFULLFLG	RAW NOTEMPTYFLG	RAWFULLFLG	RAW EMPTYFLG
7	6	5	4	3	2	1	0

Bit 5 : **RAWOVERFLOWFLG　ロウ FIFO オーバーフロー割り込みフラグ**

FIFO が 1 度でもオーバーフローになったかどうかを示します。

0： FIFO が 1 度もオーバーフローになっていません
1： FIFO が 1 度はオーバーフローになりました

Bit 4 : **RAWUNDERFLOWFLG　ロウ FIFO アンダーフロー割り込みフラグ**

FIFO が 1 度でもアンダーフローになったかどうかを示します。

0： FIFO が 1 度もアンダーフローになっていません
1： FIFO が 1 度はアンダーフローになりました

Bit 3 : **RAWNOTFULLFLG　ロウ FIFO ノットフル割り込みフラグ**
FIFO が 1 度でもフルではない状態になったかどうかを示します。
0： FIFO が 1 度もフルではない状態になっていません
1： FIFO が 1 度はフルではない状態になりました

Bit 2 : **RAWNOTEMPTYFLG　ロウ FIFO ノットエンプティ割り込みフラグ**
FIFO が 1 度でもエンプティではない状態になったかどうかを示します。
0： FIFO が 1 度もエンプティではない状態になっていません
1： FIFO が 1 度はエンプティではない状態になりました

Bit 1 : **RAWFULLFLG　ロウ FIFO フル割り込みフラグ**
FIFO が 1 度でもフル状態になったかどうかを示します。
0： FIFO が 1 度もフル状態になっていません
1： FIFO が 1 度はフル状態になりました

Bit 0 : **RAWEMPTYFLG　ロウ FIFO エンプティ割り込みフラグ**
FIFO が 1 度でもエンプティ状態になったかどうかを示します。
0： FIFO が 1 度もエンプティ状態になっていません
1： FIFO が 1 度はエンプティ状態になりました

I2S[1:0]割り込みイネーブルレジスタ
I2S0[0x18], I2S1[0x58]　　初期値 = 0x0000　　Read/Write

n/a							
15	14	13	12	11	10	9	8
n/a		OVERFLOW IRQEN	UNDERFLOW IRQEN	NOTFULL IRQEN	NOTEMPTY IRQEN	FULL IRQEN	EMPTY IRQEN
7	6	5	4	3	2	1	0

Bit 5 : **OVERFLOWIRQEN　FIFO オーバーフロー割り込みイネーブル**
FIFO が 1 度でもオーバーフローになったときに割り込みを発生させるかどうかを選択します。
0： 割り込みを発生させません
1： 割り込みを発生させます

Bit 4 : **UNDERFLOWIRQEN　FIFO アンダーフロー割り込みイネーブル**
FIFO が 1 度でもアンダーフローになったときに割り込みを発生させるかどうかを選択します。
0： 割り込みを発生させません
1： 割り込みを発生させます

Bit 3 : **NOTFULLIRQEN　FIFO ノットフル割り込みイネーブル**
FIFO が 1 度でもフルではない状態になったときに割り込みを発生させるかどうかを選択します。
0： 割り込みを発生させません
1： 割り込みを発生させます

Bit 2 : **NOTEMPTYIRQEN　FIFO ノットエンプティ割り込みイネーブル**
FIFO が 1 度でもエンプティではない状態になったときに割り込みを発生させるかどうかを選択します。
0： 割り込みを発生させません
1： 割り込みを発生させます

Bit 1 : **FULLIRQEN　FIFO フル割り込みイネーブル**
FIFO が 1 度でもフル状態になったときに割り込みを発生させるかどうかを選択します。
0： 割り込みを発生させません
1： 割り込みを発生させます

Bit 0 : **EMPTYIRQEN　FIFO エンプティ割り込みイネーブル**
FIFO が 1 度でもエンプティ状態になったときに割り込みを発生させるかどうかを選択します。
0： 割り込みを発生させません
1： 割り込みを発生させます

I2S[1:0]カレントステータスレジスタ
I2S0[0x1C], I2S1[0x5C]　　初期値 = 0x0009　　Read Only

FIFOWPNTR [3:0]				FIFORPNTR [3:0]			
15	14	13	12	11	10	9	8
DMASTS	n/a			NOTFULLSTS	NOTEMPTYSTS	FULLSTS	EMPTYSTS
7	6	5	4	3	2	1	0

Bit [15:12] : **FIFOWPNTR [3:0]　FIFO ライトポインタ**
FIFO の現在のライトポインタを示します。(0x0 ～ 0xF)

Bit [11:8] : **FIFORPNTR [3:0]　FIFO リードポインタ**
FIFO の現在のリードポインタを示します。(0x0 ～ 0xF)

FIFO の実アドレスは、データサイズ及びステレオタイプかモノラルタイプかによって変わります。
ポインタの下位 1 ビット：16 ビットステレオ、14 ビットステレオ
ポインタの下位 2 ビット：16 ビットモノラル、14 ビットモノラル、8 ビットモノラル
ポインタの下位 3 ビット：8 ビットモノラル

Bit 7 : **DMASTS　DMA ステータス**
DMA リクエストを現在発行しているかどうかを示します。
0： DMA リクエストは発行していません
1： DMA リクエストを発行しています

Bit 3 : **NOTFULLSTS　FIFO ノットフルカレントステータス**
FIFO が現在フルではない状態であるかどうかを示します。
0： FIFO は現在フル状態です
1： FIFO は現在フル状態ではありません

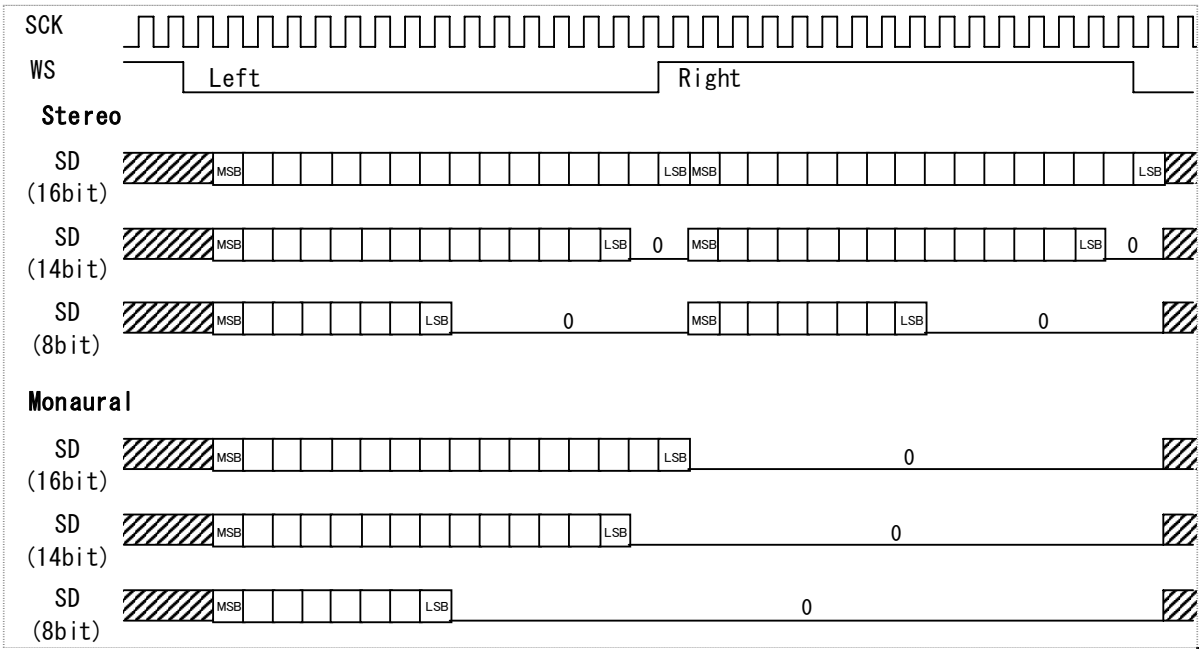
Bit 2 : **NOTEMPTYSTS　FIFO ノットエンプティカレントステータス**
FIFO が現在エンプティではない状態であるかどうかを示します。
0： FIFO は現在エンプティ状態です。
1： FIFO は現在エンプティ状態ではありません

Bit 1 : **FULLSTS　FIFO フルカレントステータス**
FIFO が現在フル状態であるかどうかを示します。
0： FIFO は現在フル状態ではありません
1： FIFO は現在フル状態です

Bit 0 : **EMPTYSTS　FIFO エンプティカレントステータス**
FIFO が現在エンプティ状態であるかどうかを示します。
0： FIFO はエンプティ状態ではありません
1： FIFO はエンプティ状態です

18.5 機能説明

18.5.1 I2S タイミングチャート (32fs)



18.5.2 データ幅と FIFO 段数

1 回のアクセスで読み書きできるデータは、データ幅及びステレオ／モノラル設定によって異なります。

16 ビットステレオ：
- TXD/RXD[31:16] 右データ
- TXD/RXD[15:0] 左データ

14 ビットステレオ：
- TXD/RXD[31:30] 0 データ
- TXD/RXD[29:16] 右データ
- TXD/RXD[15:14] 0 データ
- TXD/RXD[13:0] 左データ

8 ビットステレオ
- TXD/RXD[31:16] 無効データ
- TXD/RXD[15:8] 右データ
- TXD/RXD[7:0] 左データ

16 ビットモノラル：
- TXD/RXD[31:16] 無効データ
- TXD/RXD[15:0] モノラルデータ

14 ビットモノラル：
- TXD/RXD[31:16] 無効データ
- TXD/RXD[15:14] 0 データ
- TXD/RXD[13:0] モノラルデータ

8 ビットモノラル：
- TXD/RXD[31:8] 無効データ
- TXD/RXD[7:0] モノラルデータ

注意：送受信ポートレジスタはサンプリング単位でデータを扱っていますので、1 回のレジスタアクセスで 1 つのサンプリングデータがすべて読み書きされなければなりません。例えば 16 ビットステレオの場合、左データのみを 16 ビットアクセスで読み出したとしても、右データは次のデータに更新されてしまいますので、32 ビットアクセスで左右両方のデータを一度で読み出すようにしてください。

FIFO 段数は、データ幅及びステレオ／モノラル設定によって変化します。
FIFO2 段：16 ビットステレオ、14 ビットステレオ
FIFO4 段：16 ビットモノラル、14 ビットモノラル、8 ビットステレオ
FIFO8 段：8 ビットモノラル

18.5.3 DMA 転送

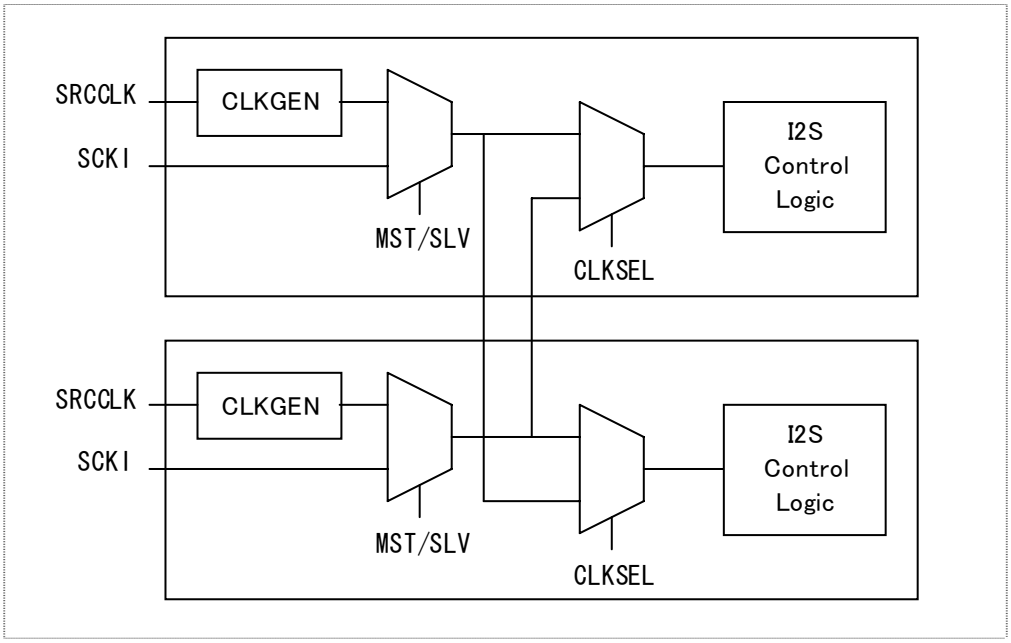
I2S は DMA コントローラ 1 を使用した DMA 転送が可能です。但し、使用可能なチャネルは 2/3 になります。送信モード時は FIFO がフル状態ではないときに DMA リクエストがアサートされ、受信モード時は FIFO がエンプティ状態ではないときに DMA リクエストがアサートされます。また FIFO がオーバーフローになった時点で DMA リクエストはネゲートされます。これは FIFO オーバーフローという異常状態のときに I2S に余計な動作をさせないための処置です。
I2S0[0x00], I2S1[0x40] Bit 3 を“1”にすることで DMA 転送が有効になります。

18.5.4 INTC への割り込み

I2S は 2Channel 構成となっていますが、INTC への割り込み(IRQ31)は 2Channel を“OR”して送信しています。割り込み要因を確認する際は、2Channel 両方の I2S[1:0]割り込みステータスレジスタ(I2S0[0x10], I2S1[0x50])を確認する必要があります。

18.5.5 クロック選択（クロック共有）

I2S0[0x00], I2S1[0x40]の Bit 8 のクロック選択機能によって、使用するクロックを共有化することができます。共有化する回路は以下の構成になっています。



他の Channel からクロックを入力してクロック共有状態となった Channel はマスターモードでクロック出力ができなくなります。したがって I2S0[0x00], I2S1[0x40]の Bit 1 を“0”にし、スレーブモードで使用します。クロックを提供する側の Channel は、マスターモード、スレーブモードのどちらでも使用できます。
同一クロックを使用して I2S の送受信を同時に行う場合、このクロック選択機能を使用することで 4 つの信号ピンで実現できるようになります。

18.5.6 モノラルーステレオ変換機能

モノラルデータをステレオタイプのオーディオに出力する場合に、モノラルデータをステレオデータに変換して出力することができます。L データしかないモノラルデータに対して L データを R データにコピーし、ステレオ化します。
送信モード時に I2S0[0x00], I2S1[0x40] Bit 12 を“1”にすることでステレオデータに変換されてデータが出力されます。
受信したステレオデータをモノラルデータに変換することはできません。

18.6 設定例

送受信をもつオーディオチップと SCK、WS、SDI、SDO の 4 本で接続し、オーディオチップのクロックとワードセレクト（LR クロック）を受信する場合の I2S レジスタ設定例を示します。I2S の I/O が他の I/O と共有している場合は、この設定の前に I2S の I/O が有効になるように GPIO のレジスタ設定を行う必要があります。

Channel0 の設定条件
- スレーブモード
- 送信モード
- DMAイネーブル
- 16ビットステレオ
- 32fs

Channel1 の設定条件
- スレーブモード
- 受信モード
- DMAイネーブル
- 16ビットステレオ
- 32fs
- クロック共有

設定手順

```
I2S0[0x00] = 0x00000005   # スレーブモード、送信モード、16 ビットステレオ、32fs
I2S0[0x18] = 0x00000030   # オーバーフロー、アンダーフロー割り込み有効
I2S0[0x00] = 0x00000085   # ソフトウェアリセット
I2S0[0x00] = 0x0000000D   # DMA イネーブル
I2S1[0x40] = 0x00000101   # スレーブモード、受信モード、16 ビットステレオ、32fs、クロック共有
I2S1[0x58] = 0x00000030   # オーバーフロー、アンダーフロー割り込み有効
I2S1[0x40] = 0x00000181   # ソフトウェアリセット
I2S1[0x40] = 0x00000109   # DMA イネーブル
```

19. シリアル周辺機器インタフェース (SPI)

19.1 概要

シリアル周辺機器インタフェース（SPI）は 1 チャネルです。
SPI はマスタまたはスレーブモードの両方の動作をサポートし、1 ～ 32 ビットのデータ転送を行います。個々のデータ転送の間に 0 ～ 65535 クロック分の遅延時間を挿入することができ、内部割り込みも発生可能です。送信、受信ともにデータバッファが組み込まれています。SPI には 4 本の端子が設けられております。SRDY#信号に関しては内部で LOW に固定されており、外部端子として使用することはできません。

19.1.1 マスタモード

マスタモードに設定された SPI は、 SPI バスに接続されたスレーブデバイスとのデータ転送を制御します。シリアルクロックは SPI が SCLK 端子からスレーブデバイスに供給し、シリアルデータは MOSI 端子から出力、 MISO 端子から入力します。 SPI には SS（スレーブセレクト）端子も設けられています。この端子はデータ転送には必要ではありませんが、モード違反エラーの検出に使用することができます。モード違反エラーはマルチマスタ SPI システムにおいて、 2 つ以上のデバイスが同時にマスタに設定されると発生します。 SPI がマスタモード時に SS 端子が アクティブレベルになったことを検出するとモード違反割り込みが発生し、SPI は信号の競合を避けるため自動的にスレーブモードに再設定されます。モード違反エラーの検出が不要な場合は SS 端子を汎用入出力ポートとして使用することができます。
SPI をイネーブル（動作可能状態）にした後、送信データレジスタ (TXD) に送信データを書き込むと、データ転送が始まります。

図 19.1 にマスタモードの制御 /動作フローを示します。

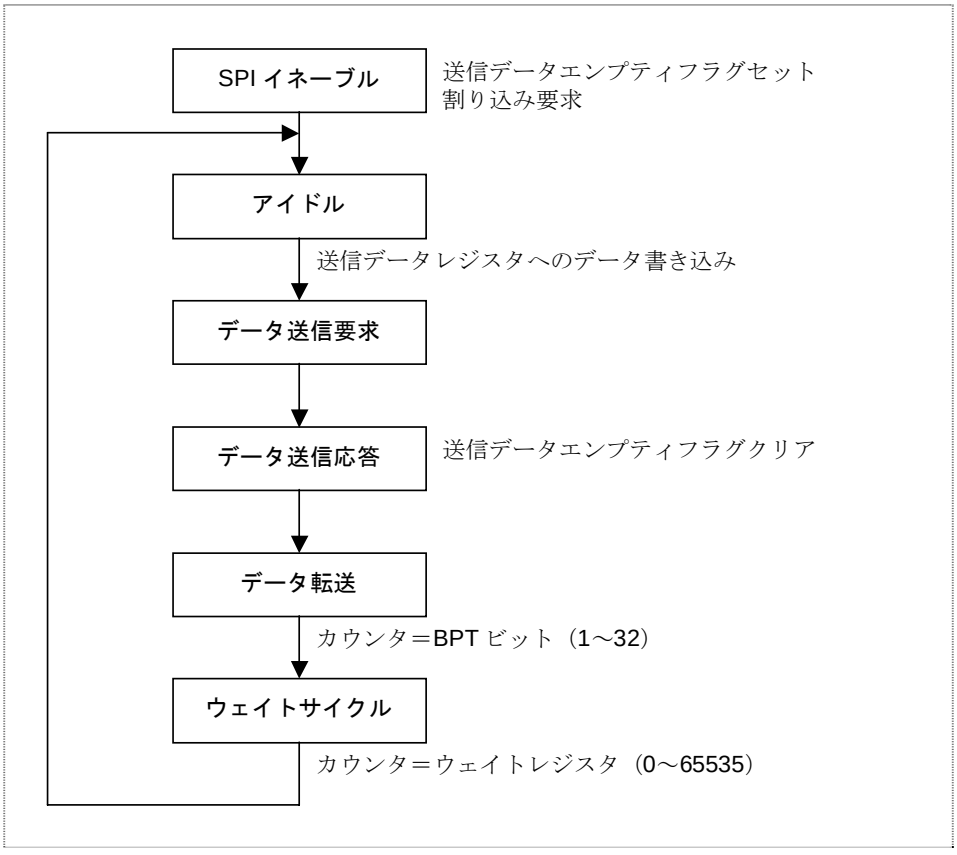


図 19.1　SPI マスタモード時の送信フロー

19.1.2 スレーブモード

SPI をスレーブモードに設定した場合は、外部 SPI マスタが SPI とのデータ転送を制御します。動作クロックは、外部マスタが出力したものを SCLK 端子から入力して使用します。シリアルデータは MOSI 端子から入力、 MISO 端子から出力します。 SS（スレーブセレクト）端子は入力となります。

SS 端子が アクティブレベル になることによってシリアルクロック入力と転送動作がイネーブルとなります。

SPI をイネーブル（動作可能状態）にした後、外部 SPI マスタによって転送が開始します。 SCLK クロックで動作するカウンタが内蔵されており、設定した転送ビット数分の送受信を制御します。

設定した転送ビット数を超えて SCLK クロックが入力された場合、設定した転送ビット数のみ転送データが保証されます。

図 19.2 にスレーブモードの制御 /動作フローを示します。

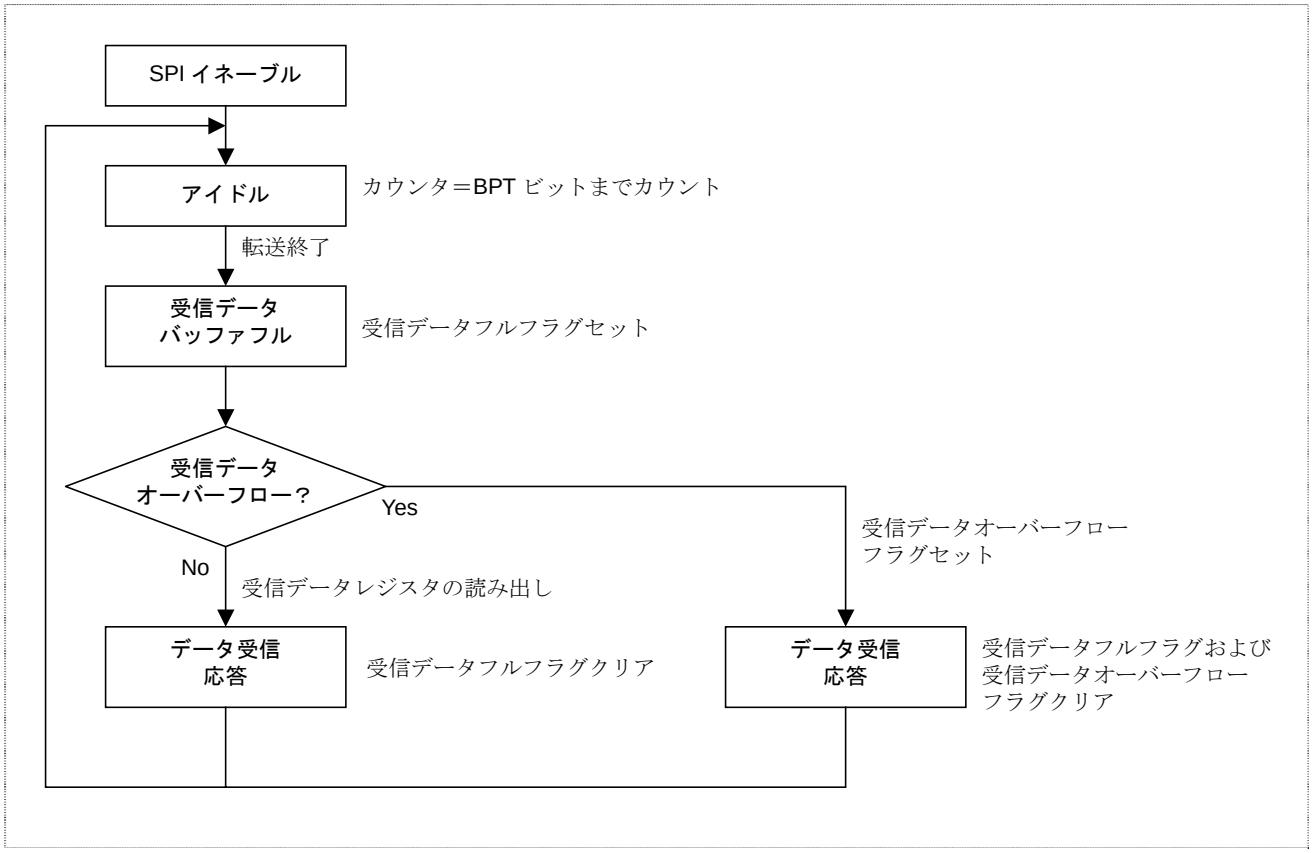


図 19.2　SPI スレーブモード時の受信フロー

19.2 ブロック図

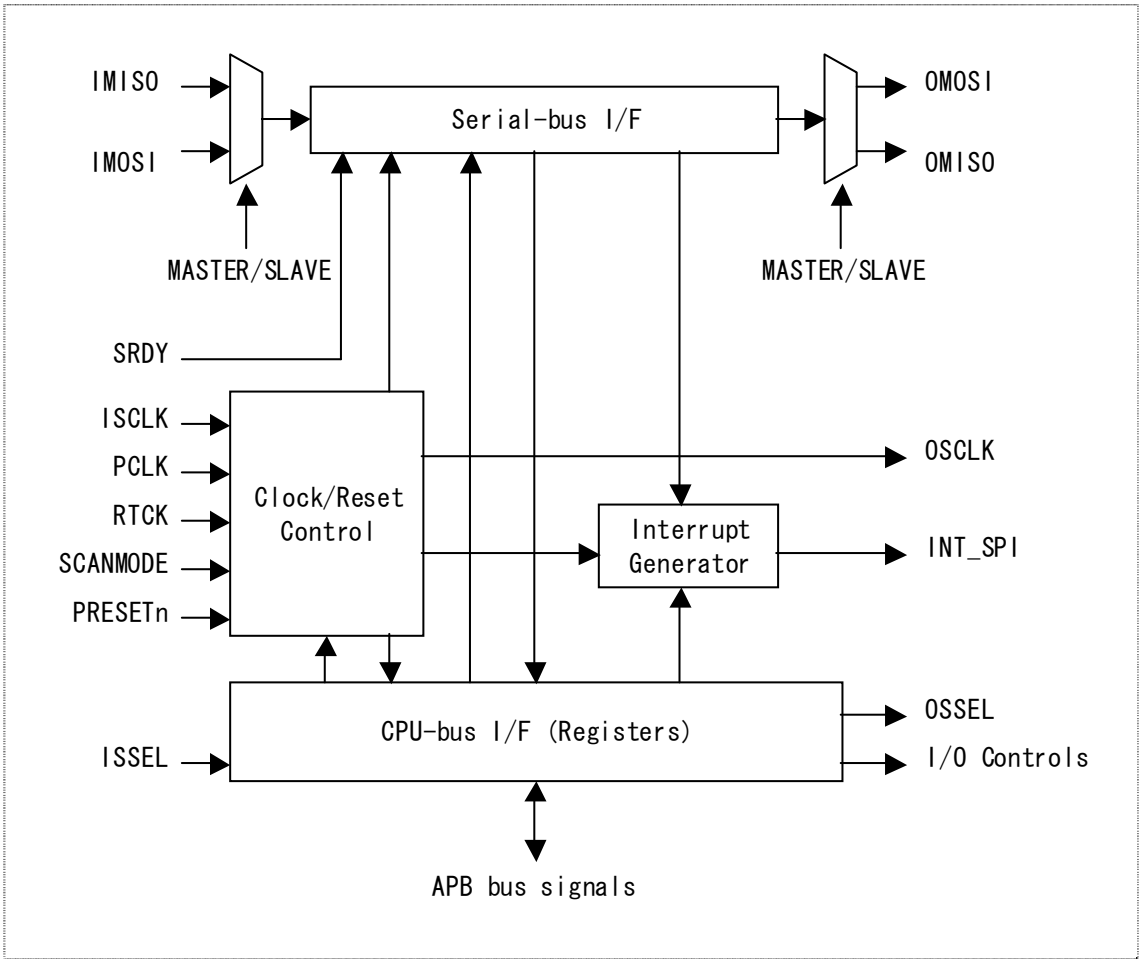


図 19.3　SPI ブロック図

19.3 外部端子

シリアル周辺機器インタフェースに関連する外部端子は以下の通りです。

端子名	入出力	端子機能	マルチプレクス端子／備考
SPI_SCLK	入出力	SPI 用シリアルクロック	GPIOC5*
SPI_SS	入出力	SPI 用チップセレクト	GPIOC4*
SPI_MISO	入出力	SPI 用シリアルデータマスタ入力/スレーブ出力	GPIOC6*
SPI_MOSI	入出力	SPI 用シリアルデータマスタ出力/スレーブ入力	GPIOC7*

注意(*) : SPI 用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1”に設定することにより使用できます。

19.4 クロックとデータ転送タイミング

SPI をマスタモードで使用する場合、内部の SCLK クロックを使用して、転送データの入出力を行うシフトレジスタを動作させます。 SCLK はクロックの位相と極性の組み合わせにより 4 種類から選択することができます。

クロックの位相は CPHA ビット（SPI 制御レジスタ 1 の Bit 9）で選択します。CPHA を“0”に設定した場合、出力データはクロックの立ち下がりエッジで変化し（シフトレジスタから出力）、入力データはクロックの立ち上がりエッジでシフトレジスタに取り込まれます（シフトレジスタ内のビットは順次シフト）。データを送信データレジスタに書き込むと、MSB が出力されます。 CPHA を“1”に設定した場合、出力は立ち上がりエッジで変化し、入力は立ち下がりエッジで取り込まれます。データの MSB は SCLK の最初の立ち上がりで出力されます。

クロックの極性は CPOL ビット（SPI制御レジスタ 1 のBit 8）で選択します。CPOLが“0”の場合はHIGHアクティブ、“1”の場合はLOWアクティブです。上記のCPHAの説明はクロックがHIGHアクティブの場合の入出力タイミングで、CPOLを“1”に設定した場合はエッジの立ち上がりと立ち下がりが逆になります。ただし、SPI内部のエッジトリガイベントのタイミングは逆転しません。

図 19.4 にこれらの選択による SCLK クロック波形を示します。この柔軟性により、市場にあるほとんどのシリアル周辺デバイスに対応します。

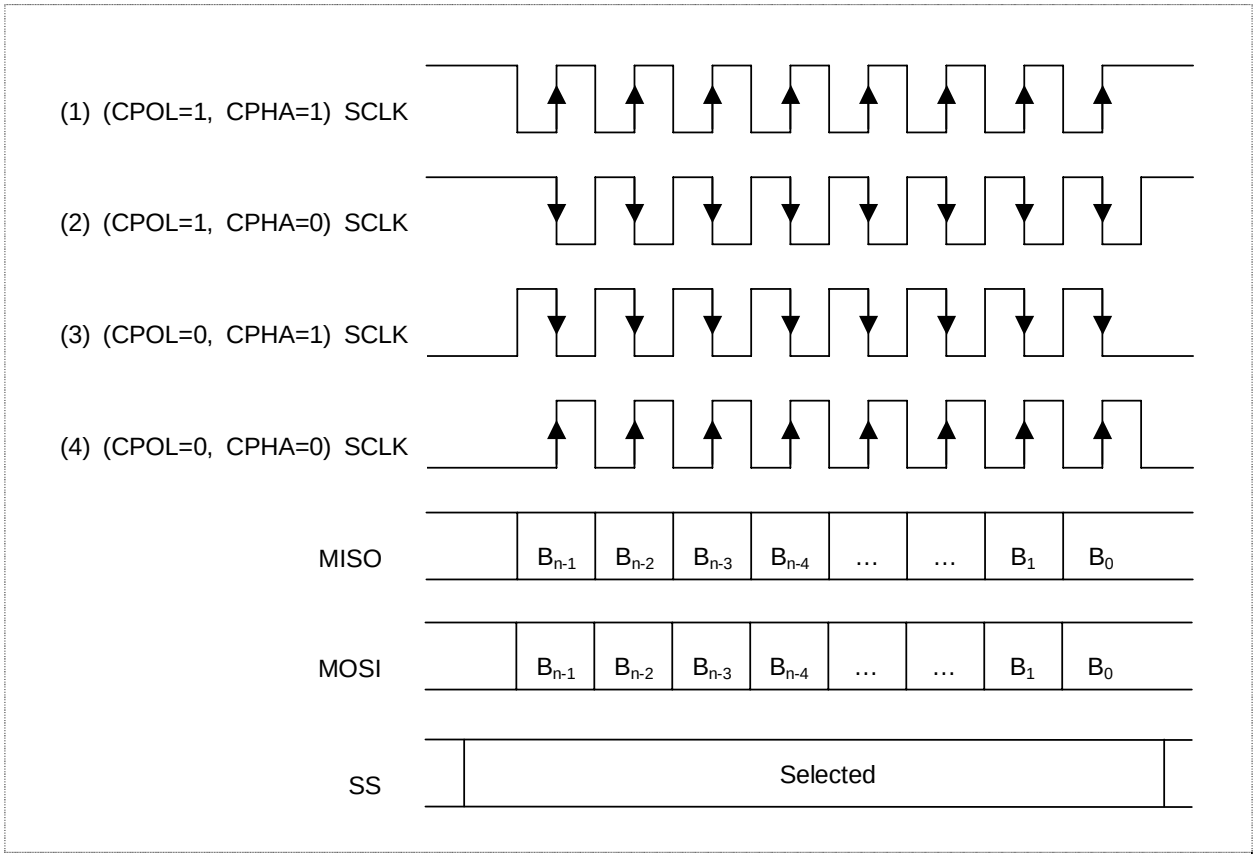


図 19.4 SPI マスタモード時のクロックの設定（転送データのビット数が n の場合）

図 19.5 にスレーブモード時のSCLKクロック波形を示します。

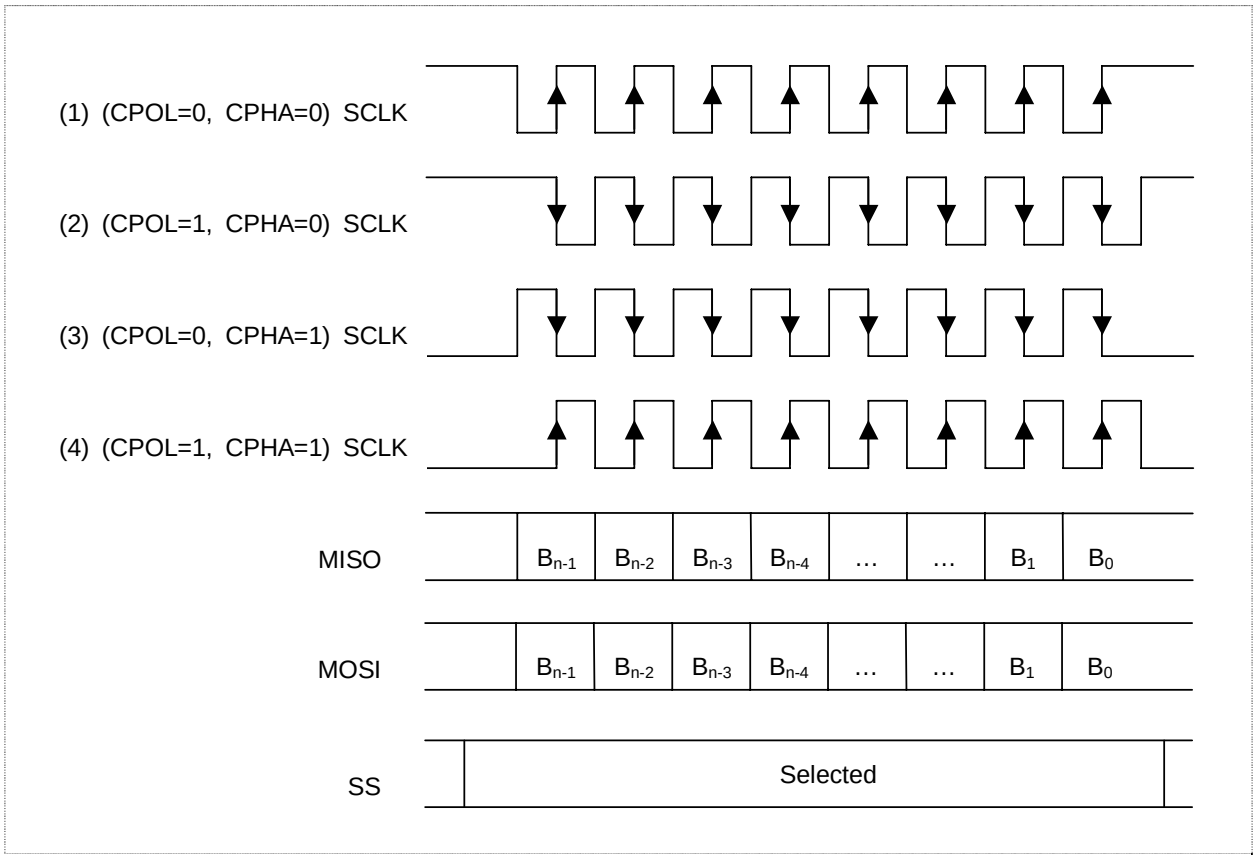


図 19.5　SPI スレーブモード時のクロックの設定（転送データのビット数が n の場合）

19.5 レジスタ説明

19.5.1 レジスター覧

SPI の制御レジスタが配置されているデフォルトのベースアドレスは、0xFFFF_2000 です。
特に指定のない場合、予約されていないレジスタビットのデフォルト値はすべて“0”です。

表 19.1 SPI レジスター覧（ベースアドレス：0xFFFF_2000）

アドレス オフセット	レジスタ名称	初期値	R/W	データ アクセスサイズ
SPI 制御レジスタ				
0x00	SPI 受信データレジスタ	0x0000_0000	RO	32
0x04	SPI 送信データレジスタ	0x0000_0000	R/W	32
0x08	SPI 制御レジスタ 1	0x0000_0000	R/W	32
0x0C	SPI 制御レジスタ 2	0x0000_0000	R/W	32
0x10	SPI ウェイトレジスタ	0x0000_0000	R/W	32
0x14	SPI ステータスレジスタ	0x0000_0010	RO	32
0x18	SPI 割り込み制御レジスタ	0x0000_0000	R/W	32

19.5.2 レジスタ詳細説明

SPI 受信データレジスタ

SPI[0x00] 初期値 = 0x0000_0000 Read Only

受信データ [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
受信データ [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]： **受信データビット[31:0]**
外部シリアル周辺デバイスから受信したデータが読み出せます。

SPI 送信データレジスタ

SPI[0x04] 初期値 = 0x0000_0000 Read/Write

送信データ [31:16]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
送信データ [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [31:0]： **送信データビット[31:0]**
送信データを書き込むデータバッファです。このレジスタが空であることを示す TDEF ビット（SPI ステータスレジスタの Bit 4）が“1”の場合にデータを書き込むことができます。

SPI 制御レジスタ 1

SPI[0x08]　初期値 = 0x0000_0000　Read/Write

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
N/a	BPT [4:0]					CPHA	CPOL	n/a	MCBR [2:0]			CLKS	RX RAW	Mode	ENA
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [14:10]：　**BPT[4:0]　転送ビット数**

1 回の転送で送受信するデータのビット長を指定します。

00000：1 ビット
00001：2 ビット
：
11110：31 ビット
11111：32 ビット

Bit 9：　**CPHA　シリアルクロック位相選択**

SCLK クロックの位相を選択します。

0： データサイクルの後半にクロックパルスを生成（図 20.4. (2)(4)参照）
1： データサイクルの前半にクロックパルスを生成（図 20.4. (1)(3)参照）

Bit 8：　**CPOL　シリアルクロック極性選択**

SCLK クロックの極性を選択します。

0： アクティブ HIGH（HIGH パルスをクロックとして生成）（図 20.4. (3)(4)参照）
1： アクティブ LOW（LOW パルスをクロックとして生成）（図 20.4. (1)(2)参照）

Bits [6:4]：　**MCBR [2:0]　マスタクロックビットレート選択**

マスタモード時の SCLK クロック速度を設定します。このビットの設定によりソースクロック（バスクロック）の分周比が次のように設定されます。

分周比 ＝ 4 ＊ $2^{MCBR[2:0]}$

したがって、SPI のマスタクロックは以下のようになります。

マスタクロック周波数 (f_{SCLK}) ＝ バスクロック周波数 / (4 ＊ $2^{MCBR[2:0]}$)

注意：　スレーブモード時、またはマスタモードでもソースクロックにリアルタイムクロック (32.768KHz) を選択した場合（本レジスタの Bit 3 = 1）、このビットは無効になります。

Bit 3：　**CLKS　ソースクロック選択**

マスタモード時の SCLK クロックを生成するソースクロックを選択します。

0： バスクロック
1： リアルタイムクロック (32.768KHz)

Bit 2：　**RXDATA RAW**

0： RXDATA は BPT 幅によりマスクされます。
1： RXDATA はシフトレジスタのマスク前データです。

Bit 1：　**Mode　SPI モード選択**

本インタフェースをマスタモードで使用するか、スレーブモードで使用するか選択します。

0： スレーブモード
1： マスタモード

Bit 0：　**ENA　SPI イネーブル**

SPI の送受信回路をイネーブルにします。

0： ディセーブル
1： イネーブル

SPI 制御レジスタ 2

SPI[0x0C]　初期値 = 0x0000_0000　Read/Write

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a				SSA	SS	SSP	SSC	n/a					予約 (0)		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 11 : **SSA　スレーブ選択端子(SS)自動制御**

このビットはマスタモード時には以下のような設定を行います。

0： SS 端子が出力に設定されている場合、SS 端子は SS（Bit10）に制御されます。

1： SS 端子が出力に設定されている場合、SS 端子は内部の転送タイミングにより制御されます。

このビットはスレーブモード時には無効です。

Bit 10 : **SS　スレーブ選択端子(SS)制御**

マスタモード時に SS 端子が出力に設定されているときに、SS 端子の出力を制御します。

0： SS 端子はインアクティブレベル出力

1： SS 端子はアクティブレベル出力

スレーブモード時に SS 端子が無効入力として設定されているとき (SSC = 0) には、

0： この SPI は選択されていない

1： この SPI は選択されている

その他の場合にはこのビットは無効です。

Bit 9 : **SSP　スレーブ選択端子(SS)極性選択**

0： LOW アクティブ

1： HIGH アクティブ

Bit 8 : **SSC スレーブ選択端子(SS)設定**

マスタモード時に、SS 端子の入出力方向を切り換えます。

0： 入力（モード違反の検出）

1： 出力（スレーブ選択出力）
HIGH/LOW 出力を SS ビット（本レジスタの Bit 10）で設定できます。
モード違反の検出は行えません。

スレーブモードの場合、

0：SS 端子を無効入力として設定。SS ビット（Bit10）の選択が有効となります。

1：SS 端子は有効入力として設定。

Bits [10:8]の設定をまとめると以下のようになります。

表 19.2　SS 端子の状態設定

モード選択*	Bit 8：SSC SS 設定	Bit 11：SSA SS 自動選択	Bit 9：SSP SS 極性選択	Bit 10：SS SS 制御	SS 端子の状態（アクティブレベル）
マスタモード	0：SS 端子入力 (モード違反検出)	無効	0：LOW アクティブ	無効	SS 入力（LOW）
			1：HIGH アクティブ		SS 入力（HIGH）
	1：SS 端子出力	0：SS ビットによる制御	0：LOW アクティブ	0：インアクティブ	SS 出力＝HIGH（LOW）
				1：アクティブ	SS 出力＝LOW（LOW）
			1：HIGH アクティブ	0：インアクティブ	SS 出力＝LOW（HIGH）
				1：アクティブ	SS 出力＝HIGH（HIGH）
		1：自動制御	0：LOW アクティブ	無効	SS 出力＝自動制御 (LOW)
			1：HIGH アクティブ		SS 出力＝自動制御 (HIGH)
スレーブモード	0：SS 端子無効入力	無効	無効	0：非選択	SS 入力 (HIGH) <非選択>
				1：選択	SS 入力 (HIGH) <選択>
	1：SS 端子有効入力		0：LOW アクティブ	無効	SS 入力（LOW）
			1：HIGH アクティブ		SS 入力（HIGH）

＊：モード選択は Mode ビット（SPI 制御レジスタ 1 の Bit 1）で行います。

Bits [2:0] :　**予約（0）**

SPI ウェイトレジスタ

SPI[0x10]　初期値 = 0x0000_0000　Read/Write

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WAIT Cycles [15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [15:0] :　**WAIT Cycles [15:0]　ウェイトサイクル**

各データ送受信動作の間に挿入する待ち時間を SCLK クロック数で設定します。（待ち時間 = WAIT Cycles [15:0] × SCLK 周期）

0000h (w)：0 クロック
0001h (w)：1 クロック
0002h (w)：2 クロック
：
FFFFh (w)：65535 クロック

このレジスタの設定は、マスタモード時のみ有効となります。

SPI ステータスレジスタ

SPI[0x14]　初期値 = 0x0000_0010　Read Only

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a									BSYF	MFEF	TDEF	RDOF	RDFF	n/a	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 6 : **BSYF　転送ビジーフラグ**

SPI が送信/受信動作中であることを示します。

0： 待機中

1： 送信または受信動作中

このフラグは送信または受信開始により自動的にセットされます。セットされたフラグは、転送を終了して待機中になると自動的にクリアされます。

このフラグはマスタモード時のみ有効で、スレーブモード時は常に“0”になります。

Bit 5 : **MFEF　モード違反エラーフラグ**

モード違反エラーが発生したことを示します。

0： エラーなし

1： エラー発生

SPI がマスタモードの場合に、SS 端子が外部シリアルデバイスによってアクティブレベルにされると、このフラグがセットされます。このフラグはエラーを解除すると自動的にクリアされます。エラーを解除するために、SPI はこのフラグおよび MFIE ビット（SPI 割り込み制御レジスタの Bit5）がセットされている間スレーブモードになり、データ転送を開始せずにすべての出力をディセーブルにします。

Bit 4 : **TDEF　送信データエンプティフラグ**

送信データレジスタが空であることを示します。

0： 送信データあり

1： 送信データなし（デフォルト）

このフラグは、送信データレジスタに書き込まれたデータがシリアルインタフェースに送られるとき（またはリセット時）にセットされます。これにより、次の送信データを送信データレジスタに書き込むことができ、セットされたフラグは送信データレジスタへの書き込みによってクリアされます。

Bit 3 : **RDOF　受信データオーバーフローフラグ**

受信データオーバーフローが発生したことを示します。

0： オーバーフローなし

1： オーバーフロー発生

受信データフルフラグがセットされている状態（受信データが読み出されていない状態）で、次の受信データがシリアルインタフェースから受信データレジスタに送られると、このフラグがセットされます。セットされたフラグは受信データレジスタを読み出すことでクリアされます。

Bit 2 : **RDFF　受信データフルフラグ**

受信データレジスタに受信データがあることを示します。

0： 受信データなし

1： 受信データあり

受信データがシリアルインタフェースから受信データレジスタに送られると、このフラグがセットされます。セットされたフラグは受信データレジスタを読み出すことでクリアされます。

注意：このレジスタは、SPI イネーブルビット（SPI 制御レジスタ 1 の Bit 0）が“0”にセットされ、SPI ディセーブルになったときもすべてクリアされます。

SPI 割り込み制御レジスタ

SPI[0x18]　初期値 = 0x0000_0000　Read/Write

n/a															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a										MFIE	TEIE	ROIE	RFIE	MIRQ	IRQE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit 5 : **MFIE　モード違反割り込みイネーブル**

モード違反割り込みをイネーブル（許可）/ディセーブル（禁止）します。

0： 割り込みディセーブル

1： 割り込みイネーブル

この設定は、SPI がマスタモードで、SS 端子がモード違反検出用に設定されている場合に有効です。

Bit 4 : **TEIE　送信データレジスタエンプティ割り込みイネーブル**

送信データレジスタエンプティ割り込みをイネーブル/ディセーブルします。

0： 割り込みディセーブル

1： 割り込みイネーブル

Bit 3 : **ROIE　受信データオーバーフローエラー割り込みイネーブル**

受信データオーバーフローエラー割り込みをイネーブル/ディセーブルします。

0： 割り込みディセーブル

1： 割り込みイネーブル

Bit 2 : **RFIE　受信データレジスタフル割り込みイネーブル**

受信データフル割り込みをイネーブル/ディセーブルします。

0： 割り込みディセーブル

1： 割り込みイネーブル

Bit 1 : **MIRQ　マニュアル割り込み要求セット／クリア**

SPI のマニュアル割り込み要求をセット/クリアします。

0： 割り込み要求をクリア

1： 割り込み要求をセット

このビットにより、ソフトウェアで SPI 割り込みを発生させることができます。IRQE (Bit 0) が“0（割り込み禁止）”に設定されている場合、このビットによる制御は無効です。

Bit 0 : **IRQE　割り込み要求イネーブル**

SPI の割り込み要求をイネーブル/ディセーブルします。

0： 割り込み要求ディセーブル

1： 割り込み要求イネーブル

20. コンパクトフラッシュカードインタフェース (CF)

20.1 概要

コンパクトフラッシュカードインタフェースには以下のような特長があります。

- CF Card Attribute memory Space (2KB space)
- CF Card Common memory Space (2KB space)
- CF Card IO Space (2KB space)
- 割り込み出力対応 (STSCHG#, IRQ)
- 内部 PCLK クロックの各種レンジ (50MHz-6MHz) に対応したコマンドストローブタイミング出力
- CFIORD#,CFIOWR#についてはプログラマブルなアイドルサイクル挿入とプログラマブルなコマンドサイクル挿入をサポート
- CF I/F での True IDE モードをサポート （CFOE#のプルダウン回路と CSSEL 信号、およびローアクティブ・リセットへの対応が外部で必要です。）

注意：S2S65A00 では、以下の信号線については端子数の制約から実装しておりません。内部ですべて“LOW”に固定されています。

- **CD [2:1]#**
- **VS [2:1]#**
- **BVD2#**
- **WP/IOIS16#**

20.2 ブロック図

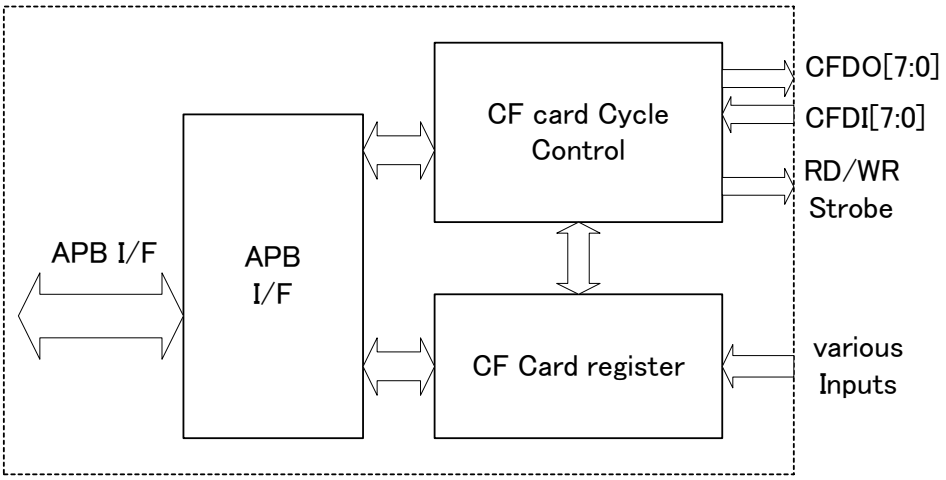


図 20.1 CF Card I/F ブロック図

20.3 CF Card I/F の各空間の配分

表 20.1　CF カード I/F の空間配分

Description	Address Range	サイズ
CF カードアトリビュート空間	0xFFFE4000 - 0xFFFE47FF	2KB
CF カードコモンメモリ空間	0xFFFE4800 - 0xFFFE4FFF	2KB
CF カード I/O 空間	0xFFFE5000 - 0xFFFE57FF	2KB
CF カード True IDE CS1# Space	0xFFFE5800 - 0xFFFE5BFF	1KB
CF カード True IDE CS2# Space	0xFFFE5C00 - 0xFFFE5FFF	1KB
CF カード インタフェース 設定	0xFFFE6000 - 0xFFFE6FFF	4KB

注意： CF カードの各空間のアクセスには 8Bit または 16Bit のデータサイズのアクセスのみを行ってください。32Bit のデータサイズによるアクセスでは正常動作いたしません。
CF カードインタフェース設定のレジスタアクセスについては 16/32Bit のいずれかでアクセスをしてください。

True IDE の実現：

S2S65A00 の CF I/F を使用して True IDE を実現するためには、指定された空間をアクセスするだけでは実現できません。システムレベル（＝ボード上）であらかじめ信号を操作することが必要です。
具体的には以下の 2 信号を正しく操作することが必須です。
－OE#（別名 ATASEL）
－CSSEL
CF Card I/F の OE#信号を Power On Reset（電源が OFF から ON に至ることが必須）時に LOW とサンプルされるようにしておきます。
もう一つは IDE デバイスのマスター/スレーブ動作を決めるために CSSEL を Pull Up または Pull Down、あるいは Open しておくことが必要になります。

また True IDE での CS1#, CS2#空間に対するアドレスは以下の様になっています。
－CS1#でアクセスできるレジスタは上記の表のアドレス空間中の下位 3Bit が 0x0 - 0x7 である空間
－CS2#でアクセスできるレジスタは上記の表のアドレス空間中の下位 3Bit が 0x6 - 0x7 である空間
になっています。たとえば“Alternate Status Register”は CS2#空間の内、下位 3Bit が 0x7 のアドレスに割り振られています。

20.4 外部端子

コンパクトフラッシュカードインタフェース (CF) 関連の外部端子は以下の通りです。

表 20.2　外部端子（CF）

端子名	入出力	端子機能	マルチプレクス端子／備考
CFCE2#	出力	CF 用カードイネーブル 2(CE2#)出力	GPIOH0*
CFCE1#	出力	CF 用カードイネーブル 1(CE1#)出力	GPIOH1*
CFIORD#	出力	CF 用 I/O Read ストローブ出力	GPIOH2*
CFIOWR#	出力	CF 用 I/O Write ストローブ出力	GPIOH3*
CFWAIT#	入力	CF カードからのウェイト要求入力	GPIOH4*
CFRST	出力	CF カードへのリセット出力	GPIOH5*
CFIRQ	入力	CF カードからの割り込み要求入力	GPIOH6*
CFSTSCHG#	入力	CF カードからのステータスチェンジ入力	GPIOH7*
CFDEN#	出力	CF カードの外部バッファ用データイネーブル出力	GPIOI0*
CFDDIR	出力	CF 用データバス方向指示出力	GPIOI1*
CFREG#	出力	CF 用アトリビュート空間および I/O 空間選択 REG 信号	MA11**
CFADDR [10:0]	出力	CF 用アドレス信号	MA [10:0]**
CFDATA [15:0]	入出力	CF 用 16 ビットデータ信号	MD [15:0]**
CFOE#	出力	CF インタフェースのメモリおよびアトリビュート空間の出力イネーブル信号	MOE#**
CFWE#	出力	CF インタフェースのメモリおよびアトリビュート空間のライトイネーブル信号	MWE0#**

注意(*)： CF 用の外部端子は GPIO 端子とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1”に設定することにより使用できます。
注意(**)： CF 動作時には、メモリコントローラ用の端子が CF の外部端子として動作します。

20.5 レジスタ

20.5.1 レジスター覧

コンパクトフラッシュカードインタフェース レジスタのベースアドレスは、0xFFFE_6000 です。

表 20.3 レジスター覧（**ベースアドレス：0xFFFE_6000**）

Address Offset	Register Name	Abbreviation Name	Default Value	R/W	Data Access Size
0x00	CF Card Interface Control Register	CFCTL	0x1000	(R/W)	16 (/32)
0x04	CF Card Pin Status Register	CFPINSTS	0x0XXX	RO	16 (/32)
0x08	CF Card IRQ Source & Clear Register	CFINTRSTS	0x0XXX	R/W	16 (/32)
0x0C	CF Card IRQ Enable Register	CFINTMSTS	0x0000	R/W	16 (/32)
0x10	CF Card IRQ Status Register	CFINTSTS	0x0000	RO	16 (/32)
0x14	CF Card MISC Register	CFMISC	0x0000	R/W	16 (/32)

20.5.2 レジスタ詳細説明

CF Card Interface Control Register (CFCTL)

CF[0x00] 初期値 = 0x1000 (Read/Write)

15	14	13	12	11	10	9	8
PROG CYCEN	PROG IDLE [2:0]			PROG CYC [3:0]			
R/W							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved (0)	IOIS8_IO	IOIS8_MEM	PROG IDLE EN	CFRST	CFCARDEN	PCKMD[1:0]	
RO				R/W			

Bit 15 : **PROG_CYCEN**
CFIORD#/CFIOWR#信号についてプログラマブルなコマンドサイクル機能をイネーブルにします。
通常はディスエーブルのまま使ってください。
0： Disable (After RESET/ Default)
1： Enable

Bits [14:12] : **PROG_IDLE[2:0]**
CF Card I/F での Idle Cycle 数を指定します。（Default 値は“1”）

Bits [11:8] : **PROG_CYC[3:0]**
CFIORD#/CFIOWR#信号についてコマンド・アクティブ・サイクル数を指定します。
PROG_CYCEN =1 のときのみ有効。

Bit 7 : **Reserved**

Bit 6 : **IOIS8_IO**
CF カードの IO 空間のデバイスサイズを指定します。
0： 16Bit CF カード・デバイスとして動作
1： 8Bit CF カード・デバイスとして動作

Bit 5 : **IOIS8_MEM**
CF カードのコモンメモリ空間のデバイスサイズを指定します。
0： 16Bit CF カード・デバイスとして動作
1： 8Bit CF カード・デバイスとして動作

Bit 4 : **PROG_IDLE_EN**
CFIORD#/CFIOWR#信号について CFCE1#・CFCE2#からコマンドがアクティブになるまでのプログラマブルなアイドルサイクル機能をイネーブルにします。通常はディスエーブルのまま使用してください。
0： Disable
1： Enable

Bit 3 : **CFRST**
CF I/F 機能が選択された場合、CFRST Pin をダイレクトにコントロールします。
0： CFRST Pin は“Low”になります。
1： CFRST Pin は“High”になります。

Bit 2 : **CFCARDEN**
0： CF Card Interface は無効です。
1： CF Card Interface が有効になります。

Bits [1:0] : **PCKMD[1:0]**
クロックの周波数に応じて CF Card I/F が適切に働くよう値を変更できます。
00：PCLK が 50MHz (-25MHz) 程度のとき、この値を使用してください。
01：PCLK が 24MHz 程度のときこの値を使用してください。
10：PCLK が 12MHz 程度のときこの値を使用してください。
11：PCLK が 6MHz 程度のときこの値を使用してください。

CF Card Pin Status Register (CFPINSTS)
CF[0x04] 初期値 = 0x0XXX Read Only

15	14	13	12	11	10	9	8
0							IREQ#2
WP	IREQ#1	BVD2#	BVD1#/ STSCHG	VS2#	VS1#	CD2#	CD1#
7	6	5	4	3	2	1	0

Bit [1:0]を除き Pin の状態そのものの値が各 Bit に反映されます。
Bit [1:0]（CD2#,CD1#）Pin はノイズフィルタを通った pin の状態がみえます。

Bit 8 : **IREQ#2 入力**
IREQ# Pin が High アクティブの割り込み入力のときには、この Bit 参照が便利です。
CF Card IREQ# の Pin 状態そのものを知ることができます。

Bit 7 : **WP Pin 入力**
CF Card インタフェース WP Pin の状態を知ることができます。

Bit 6 : **RDY/BSY, IREQ pin 入力（CF Card ではモードにより名前が変わります）**
IREQ# Pin が Low アクティブの割り込み入力のときには、この Bit 参照が便利です。
CF Card インタフェース RDY/BSY または IREQ Pin の反転入力状態を知ることができます。

Bit 5 : **BVD2# pin 入力**
CF Card インタフェース BVD2# pin の状態を知ることができます。

Bit 4 : **BVD1#/STSCHG# pin 入力**
CF Card インタフェース BVD1#/STSCHG# pin の状態を知ることができます。

Bit 3 : **VS2# pin 入力**
CF Card インタフェース VS2# pin の状態を知ることができます。

Bit 2 : **VS1# pin 入力**
CF Card インタフェース VS2# pin の状態を知ることができます。

Bit 1 : **CD2# pin 入力**
ノイズ除去後の CF Card インタフェース CD2# pin の状態を知ることができます。

Bit 0 : **CD1# pin 入力**
ノイズ除去後の CF Card インタフェース CD1# pin の状態を知ることができます。

注意：Bit7, Bit5, Bits [3:0]は S2S65A00 ではサポートしておりません。

CF Card IRQ Source & Clear Register (CFINTRSTS)
CF[0x08] 初期値 = 0x0XXX Read/Write

Reserved							IRQ#2
15	14	13	12	11	10	9	8
Reserved	IRQ#1	Reserved	BVD1/ STSCHG	Reserved		CD2	CD1
7	6	5	4	3	2	1	0

このレジスタは割り込み要求のソース(マスクされていない)を示すと同時に“1”を書くことにより、それぞれ該当 Bit がクリアされます。

Bits [15:7] : **予約ビット**
読み出し時は“0”が読み出されます。書き込み時は必ず“0”を書くこと。

Bit 8 : **IRQ#2 入力**
CFIRQ Pin が High アクティブの割り込み入力のときには、この Bit 参照が便利です。
CF Card が I/O モードにプログラムされたあと、この Bit が“1”を示すときは CFIRQ による割り込み要求が発生していることを示します。“0”を示すときは CFIRQ pin による割り込み要求は発生していません。

Bit 6 : **IRQ#1（反転入力）**
CFIRQ Pin が Low アクティブの割り込み入力のときには、この Bit 参照が便利です。
CF Card が I/O モードにプログラムされたあと、この Bit が“1”を示すときは CFIRQ による割り込み要求が発生していることを示します。“0”を示すときは CFIRQ pin による割り込み要求は発生していません。

Bit 4 : **BVD1/STSCHG#**
CF Card が I/O モードにプログラムされたあと、この Bit が“0”を示すときは、STSCHG#が“0”、すなわち RDY/BSY#または WP 信号に変化があったことを示します。“1”は特に RDY/BSY#または WP 信号に変化がないことを示します。

Bit 1 : **CD2 pin status change**
この信号（ノイズ除去後の信号）が Low から High,または High から Low に変化したときに“1”にセットされます。割り込み信号のソースのひとつであり、“1”を書くことによりクリアされます。

Bit 0 : **CD1 pin status change**
この信号（ノイズ除去後の信号）が Low から High,または High から Low に変化したときに“1”にセットされます。割り込み信号のソースのひとつであり、“1”を書くことによりクリアされます。

注意：Bits [1:0] は S2S65A00 ではサポートしておりません。

CF Card IRQ Enable Register (CFINTMSTS)

CF[0x0C] 初期値 = 0x0000 Read/Write

15	14	13	12	11	10	9	8
Reserved							IRQEN#EN2
7	6	5	4	3	2	1	0
Reserved	IREQ#EN1	Reserved	BVD1EN/ STSCHGEN	Reserved		CD2EN	CD1EN

各 Bit は CF Card IRQ Source & Clear Register の Bit に対応しています。
各 Bit とも
0：割り込みはマスクされます。
1：割り込みはアンマスク、すなわちイネーブルの状態になります。

注意：Bits [1:0] は S2S65A00 ではサポートしておりません。

CF Card IRQ Status Register (CFINTSTS)

CF[0x10] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Reserved							IREQ#2
7	6	5	4	3	2	1	0
Reserved	IREQ#1	Reserved	BVD1/ STSCHG	Reserved		CD2	CD1

このレジスタには IRQ Source Register と IRQ Enable Register の各 Bit の“AND”をとった値が反映されます。
0：割り込みそのものがないか、割り込みがマスクされている状態。
1：割り込みがイネーブルでかつ割り込み要因が存在することを示す。

注意：Bits [1:0] は S2S65A00 ではサポートしておりません。

CF Card MISC Register (CFMISC)

CF[0x14] 初期値 = 0x0000 Read/Write

15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							CSRDEN

このレジスタは一部のハードウエア用に用意されており、通常は何も書かないでください。

Bits [15:2]： **予約ビット**

Bit 0： **CSRDEN**
リード時に CFCE1#と CFCE2#の両方をアクティブにするかどうかを決定します。
0： ノーマルオペレーション。
1： リード操作時に CFCE1#と CFCE2#の両方をアクティブにします。

20.6 本コンパクトフラッシュカードインタフェースの利用制限事項

本チップでは、端子数の制約のために以下の端子を実装しておらず、内部で“LOW”に固定しているために、一部のレジスタに対して利用制限がかかります。

- CD [2:1]#
- VS [2:1]#
- BVD2#
- WP/IOIS16#

以下は本チップで利用制限がかかるレジスタの一覧です。

オフセットアドレス	レジスタビット名称	制限
CF[0x04] Bit 7	WP Pin 入力	使用不可
CF[0x04] Bit 5	BVD2# Pin 入力	使用不可
CF[0x04] Bit 3	VS2# Pin 入力	使用不可
CF[0x04] Bit 2	VS1# Pin 入力	使用不可
CF[0x04] Bit 1	CD2# Pin 入力	使用不可
CF[0x04] Bit 0	CD1# Pin 入力	使用不可
CF[0x08] Bit 1	CD2 pin status change	使用不可
CF[0x08] Bit 0	CD1 pin status change	使用不可
CF[0x0C] Bit 1	CD2EN	使用不可
CF[0x0C] Bit 0	CD1EN	使用不可
CF[0x10] Bit 1	CD2	使用不可
CF[0x10] Bit 0	CD1	使用不可

21. SD メモリカードインタフェース

21.1 概要

SD メモリカードインタフェースは SD Memory Card Physical Layer Specification Version 1.01 に準拠しています。

「SDA または MMCA のいずれに加盟していること」を確認願います。確認後、「SD メモリカードインタフェース」の仕様を提示します。加盟されていない場合は、仕様公開がきません。

(SDA＝SD Card Association、MMCA＝MultiMedia Card Association)

22. タイマ A (TIMA)

22.1 概要

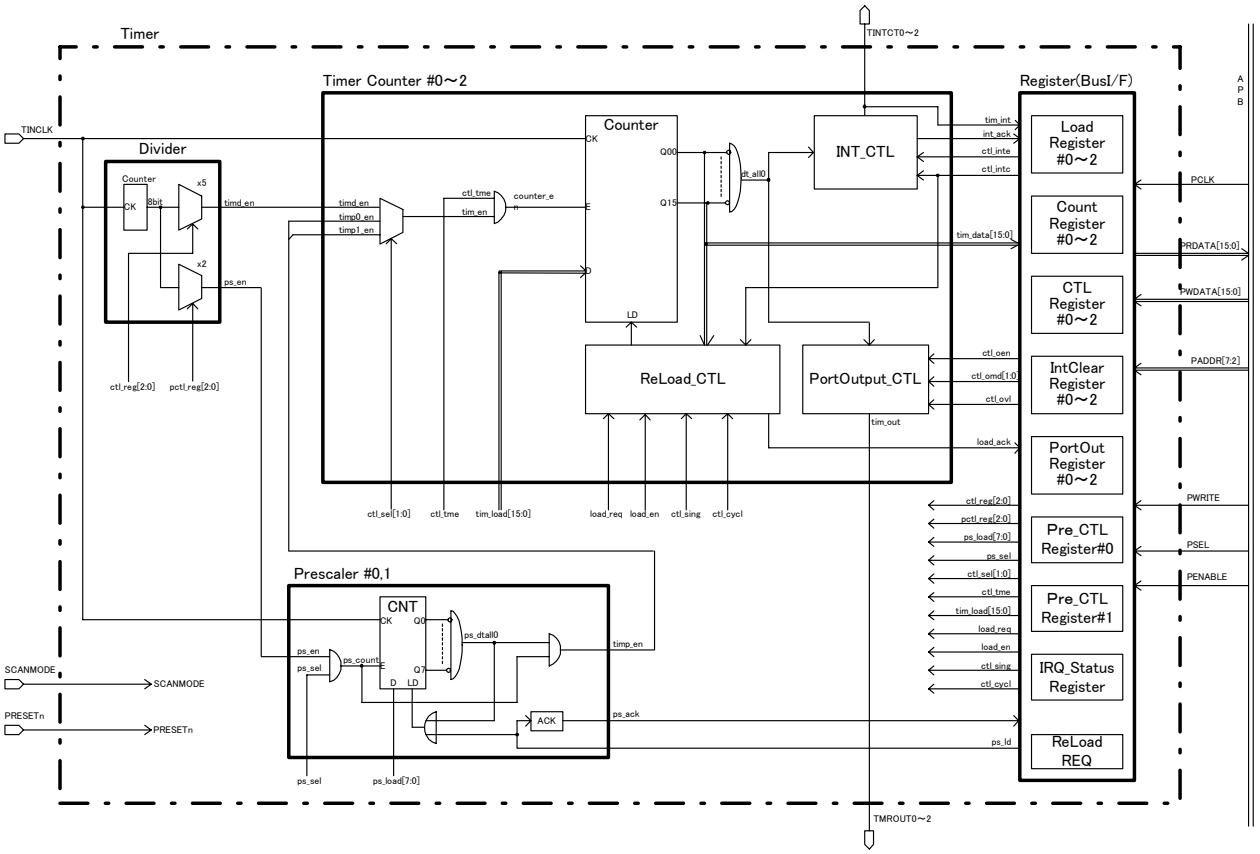
タイマ A には以下のような特長があります。

- 3ch の 16bit ダウンカウントタイマを内蔵（3ch のタイマはすべて同一構造）
- 2 種類のタイマモードをサポート（サイクリックモード／シングルモード）
- タイマがカウント“0”になるたび、IRQ がマスクされていない限り割り込みを発生
- ディバイダ用 8bit カウンタ内蔵（1/1～1/256）
- 2ch のプリスケーラ用 8bit カウンタにより、ディバイダで分周されたクロックを任意の値でカウント設定が可能
- アンダーフロー時のタイマ出力モード機能（3 種類）：
  アンダーフロー信号、任意の 0/1 値出力、アンダーフロー周波数のトグル出力

22.2 ブロック図

以下にタイマ A のブロック図を示します。タイマブロックはレジスタブロック (Register (BusI/F))、ディバイダブロック (Divider)、2ch のプリスケーラブロック (Prescaler #0,1) 及び 3ch のタイマカウンタブロック (Timer Counter #0-2) から構成されています。

プリスケーラ及びタイマカウンタのそれぞれのチャネルは同一のものであり、下記の詳細ブロックにはそれぞれの 0ch を代表して記します。ディバイダとレジスタはそれぞれ全てのチャネルに対応して制御します。タイマカウント用のクロック (TINCLK) はシステムコントローラより供給され、PCLK の 1/8 クロックが使用されます。詳細はシステムコントローラの項を参照願います。



＜16bit TIMER BLOCK＞

図 22.1 ブロックダイアグラム

22.3 外部端子

タイマ A 関連の外部端子は以下の通りです。

端子名	入出力	端子機能	マルチプレクス端子*／備考
TimerA0out	出力	タイマ A0 出力	GPIOB4/INT4 (*)
TimerA1out	出力	タイマ A1 出力	GPIOB4/INT5 (*)
TimerA2out	出力	タイマ A2 出力	GPIOB5/INT6 (*)

注意(*)： タイマ A 用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1” に設定することにより使用できます。

22.4 レジスタ

22.4.1 レジスター覧

これらのレジスタのベースアドレスは、0xFFFF_B000 です。

表 22.1 レジスター覧（ベースアドレス：0xFFFF_B000）

アドレス オフセット	レジスタ名称	レジスタ 略称	初期値	R/W	データアクセス サイズ
0x00	タイマ A0 ロードレジスタ	TMA0LD	0x0000	R/W	16 (/32) *1
0x04	タイマ A0 カウントレジスタ	TMA0CNT	0x0000	RO	16 (/32) *1
0x08	タイマ A0 制御レジスタ	TMA0CTRL	0x0000	(R/W)	16 (/32) *1
0x0C	タイマ A0 IRQ フラグクリアレジスタ	TMA0IRQ	-	WO	8 (/16/32) *2
0x10	タイマ A0 ポート出力制御レジスタ	TMA0POUT	0x0000	(R/W)	8 (/16/32) *2
0x20	タイマ A1 ロードレジスタ	TMA1LD	0x0000	R/W	16 (/32) *1
0x24	タイマ A1 カウントレジスタ	TMA1CNT	0x0000	RO	16 (/32) *1
0x28	タイマ A1 制御レジスタ	TMA1CTRL	0x0000	(R/W)	16 (/32) *1
0x2C	タイマ A1 IRQ フラグクリアレジスタ	TMA1IRQ	-	WO	8 (/16/32) *2
0x30	タイマ A1 ポート出力制御レジスタ	TMA1POUT	0x0000	(R/W)	8 (/16/32) *2
0x40	タイマ A2 ロードレジスタ	TMA2LD	0x0000	R/W	16 (/32) *1
0x44	タイマ A2 カウントレジスタ	TMA2CNT	0x0000	RO	16 (/32) *1
0x48	タイマ A2 制御レジスタ	TMA2CTRL	0x0000	(R/W)	16 (/32) *1
0x4C	タイマ A2 IRQ フラグクリアレジスタ	TMA2IRQ	-	WO	8 (/16/32) *2
0x50	タイマ A2 ポート出力制御レジスタ	TMA2POUT	0x0000	(R/W)	8 (/16/32) *2
0x60-0x9C	予約	-	-	-	-
0xA0	プリスケーラ 0 制御レジスタ	PS0CTRL	0x0000	(R/W)	16 (/32) *1
0xA4	プリスケーラ 1 制御レジスタ	PS1CTRL	0x0000	(R/W)	16 (/32) *1
0xB0	タイマ A IRQ ステータスレジスタ	TMAIRQSTS	0x0000	RO	8 (/16/32) *2

*1: 16 ビットまたは 32 ビットデータアクセスが可能。
*2: 各種サイズ（8/16/32 ビット）のデータアクセスが可能。

22.4.2 レジスタ詳細説明

タイマ A0　ロードレジスタ　(TMA0LD)

TIM[0x00]　　初期値 = 0x0000　　Read/Write

タイマ A0 ロード値[15:8]							
15	14	13	12	11	10	9	8
タイマ A0 ロード値[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**TMA0LD [15:0]　タイマ A0 ロード値ビット[15:0]**

タイマ A0 にロードするカウント初期値（16 ビット値）を設定するレジスタです。

タイマ A0　カウントレジスタ　(TMA0CNT)

TIM[0x04]　　初期値 = 0x0000　　Read only

タイマ A0 カレントカウント値[15:8]							
15	14	13	12	11	10	9	8
タイマ A0 カレントカウント値[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**TMA0CNT [15:0]　タイマ A0 カレントカウント値ビット[15:0]**

タイマ A0 の現在のカウント値が読み出せます。

タイマ A0　制御レジスタ　(TMA0CTRL)

TIM[0x08]　　初期値 = 0x0000　　(Read/Write)

n/a RO						ディバイダ／プリスケーラ [1:0] R/W	
15	14	13	12	11	10	9	8
タイマ A0 イネーブル	n/a	モード選択	ディバイダ分周比 [2:0]			イミディエイト ロード リクエスト	IRQ リクエスト
R/W	RO	R/W	R/W			R/W	R/W
7	6	5	4	3	2	1	0

Bits [9:8]：　**ディバイダ／プリスケーラ選択 [1:0]**

タイマ A0 で使用するクロックの分周元を指定します。

0x：ディバイダ
10：プリスケーラ#0
11：プリスケーラ#1

Bit 7：　**タイマ A0 イネーブル**

このビットがイネーブルのとき、このレジスタのモード選択ビットを反映したモードでダウンカウントを開始します。

0： タイマ A0 ディセーブル
1： タイマ A0 イネーブル（サイクリックモード/シングルモード）

Bit 5：　**モード選択**

0： サイクリックモード
1： シングルモード

Bits [4:2]： **ディバイダ分周比 [2:0]**

000：分周なし →1/1
001：4 分周 →1/4
010：8 分周 →1/8
011：16 分周 →1/16
100：32 分周 →1/32
101：64 分周 →1/64
110：128 分周 →1/128
111：256 分周 →1/256

Bit 1： **イミディエイトロードリクエスト（サイクリックモード時）**

サイクリックモード時にタイマ A0 のロード値をリロードするタイミングを要求します。
0： カウンタ値が“0x0000”になったときに、書き込まれたロード値をリロード
1： レジスタにロード値を書き込むと即時リロード

注意： シングルモード時は、このビットに関係なくロード値を書き込むとその値が即時反映されます。

Bit 0： **タイマ A0 割り込み要求イネーブル**

0： 割り込み要求ディセーブル（IRQ マスク）
1： 割り込み要求イネーブル

タイマ A0 IRQ フラグクリアレジスタ (TMA0IRQ)

TIM[0x0C] 初期値 = - Write Only

15	14	13	12	11	10	9	8
任意のデータ							
7	6	5	4	3	2	1	0
任意のデータ							

このレジスタはタイマ A0 の割り込み要求（IRQ）フラグをクリアするためのライトオンリのポートです。IRQ フラグをクリアするためには、このレジスタに任意のデータを書き込みます。

タイマ A0 ポート出力制御レジスタ (TMA0POUT)

TIM[0x10] 初期値 = 0x0000 (Read/Write)

15	14	13	12	11	10	9	8
n/a RO							
7	6	5	4	3	2	1	0
n/a RO				出力モード選択 R/W		出力イネーブル R/W	出力値 R/W

Bits [3:2]： **出力モード選択**

00：アンダーフロー発生時に出力値（Bit 0）を出力
01：アンダーフロー出力
10：アンダーフロー発生毎にトグル出力
11：予約

Bit 1： **出力イネーブル**

0： 出力モードディセーブル
1： 出力モードイネーブル
該当するポートが GPIO として設定されているときに、このビットを“1”にするとポートは出力モード（Bits [3:2]）で設定したように動作します。

Bit 0： **出力値**

出力モード（Bits [3:2]）が“00”のときに、このビットの値が出力されます。

タイマ A1　ロードレジスタ　(TMA1LD)

TIM[0x20]　　初期値 = 0x0000　　Read/Write

タイマ A1 ロード値[15:8]							
15	14	13	12	11	10	9	8
タイマ A1 ロード値[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**TMA1LD [15:0]　タイマ A1 ロード値ビット[15:0]**

タイマ A1 にロードするカウント初期値（16 ビット値）を設定するレジスタです。

タイマ A1　カウントレジスタ　(TMA1CNT)

TIM[0x24]　　初期値 = 0x0000　　Read Only

タイマ A1 カレントカウント値[15:8]							
15	14	13	12	11	10	9	8
タイマ A1 カレントカウント値[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**TMA1CNT [15:0]　タイマ A1 カレントカウント値ビット[15:0]**

タイマ A1 の現在のカウント値が読み出せます。

タイマ A1　制御レジスタ　(TMA1CTRL)

TIM[0x28]　　初期値 = 0x0000　　(Read/Write)

n/a RO						ディバイダ／プリスケーラ [1:0] R/W	
15	14	13	12	11	10	9	8
タイマ A1 イネーブル	n/a	モード選択	ディバイダ分周比 [2:0]			イミディエイト ロード リクエスト	IRQ リクエスト
R/W	RO	R/W	R/W			R/W	R/W
7	6	5	4	3	2	1	0

Bits [9:8]：　**ディバイダ／プリスケーラ選択 [1:0]**

タイマ A1 で使用するクロックの分周元を指定します。

0x：ディバイダ
10：プリスケーラ#0
11：プリスケーラ#1

Bit 7：　**タイマ A1 イネーブル**

このビットがイネーブルのとき、このレジスタのモード選択ビットを反映したモードでダウンカウントを開始します。

0： タイマ A1 ディセーブル
1： タイマ A1 イネーブル（サイクリックモード/シングルモード）

Bit 5：　**モード選択**

0： サイクリックモード
1： シングルモード

Bits [4:2]： **ディバイダ分周比 [2:0]**

000：分周なし →1/1
001：4 分周 →1/4
010：8 分周 →1/8
011：16 分周 →1/16
100：32 分周 →1/32
101：64 分周 →1/64
110：128 分周 →1/128
111：256 分周 →1/256

Bit 1： **イミディエイトロードリクエスト（サイクリックモード時）**

サイクリックモード時にタイマ A1 のロード値をリロードするタイミングを設定します。
0： カウンタ値が“0x0000”になったときに、書き込まれたロード値をリロード
1： レジスタにロード値を書き込むと即時リロード

注意： シングルモード時は、このビットに関係なくロード値を書き込むとその値が即時反映されます。

Bit 0： **タイマ A1 割り込み要求イネーブル**

0： 割り込み要求ディセーブル（IRQ マスク）
1： 割り込み要求イネーブル

タイマ A1 IRQ フラグクリアレジスタ (TMA1IRQ)

TIM[0x2C] 初期値 = - Write Only

15	14	13	12	11	10	9	8
任意のデータ							
7	6	5	4	3	2	1	0
任意のデータ							

このレジスタはタイマ A1 の割り込み要求（IRQ）フラグをクリアするためのライトオンリのポートです。IRQ フラグをクリアするためには、このレジスタに任意のデータを書き込みます。

タイマ A1 ポート出力制御レジスタ (TMA1POUT)

TIM[0x30] 初期値 = 0x0000 (Read/Write)

15	14	13	12	11	10	9	8
n/a RO							
7	6	5	4	3	2	1	0
n/a RO				出力モード選択 R/W		出力イネーブル R/W	出力値 R/W

Bits [3:2]： **出力モード選択**

00：アンダーフロー発生時に出力値（Bit 0）を出力
01：アンダーフロー出力
10：アンダーフロー発生毎にトグル出力
11：予約

Bit 1： **出力イネーブル**

0： 出力モードディセーブル
1： 出力モードイネーブル
該当するポートが GPIO として設定されているときに、このビットを“1”にするとポートは出力モード（Bits [3:2]）で設定したように動作します。

Bit 0： **出力値**

出力モード（Bits [3:2]）が“00”のときに、このビットの値が出力されます。

タイマ A2　ロードレジスタ　(TMA2LD)

TIM[0x40]　　初期値 = 0x0000　　Read/Write

タイマ A2 ロード値[15:8]							
15	14	13	12	11	10	9	8
タイマ A2 ロード値[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**TMA2LD [15:0]　タイマ A2 ロード値ビット[15:0]**

タイマ A2 にロードするカウント初期値（16 ビット値）を設定するレジスタです。

タイマ A2　カウントレジスタ　(TMA2CNT)

TIM[0x44]　　初期値 = 0x0000　　Read only

タイマ A2 カレントカウント値[15:8]							
15	14	13	12	11	10	9	8
タイマ A2 カレントカウント値[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**TMA2CNT [15:0]　タイマ A2 カレントカウント値ビット[15:0]**

タイマ A2 の現在のカウント値が読み出せます。

タイマ A2　制御レジスタ　(TMA2CTRL)

TIM[0x48]　　初期値 = 0x0000　　(Read/Write)

n/a RO						ディバイダ／プリスケーラ [1:0] R/W	
15	14	13	12	11	10	9	8
タイマ A2 イネーブル	n/a	モード選択	ディバイダ分周比 [2:0]			イミディエイト ロード リクエスト	IRQ リクエスト
R/W	RO	R/W	R/W			R/W	R/W
7	6	5	4	3	2	1	0

Bits [9:8]：　**ディバイダ／プリスケーラ選択 [1:0]**

タイマ A2 で使用するクロックの分周元を指定します。

0x：ディバイダ
10：プリスケーラ#0
11：プリスケーラ#1

Bit 7：　**タイマ A2 イネーブル**

このビットがイネーブルのとき、このレジスタのモード選択ビットを反映したモードでダウンカウントを開始します。

0： タイマ A2 ディセーブル
1： タイマ A2 イネーブル（サイクリックモード/シングルモード）

Bit 5：　**モード選択**

0： サイクリックモード
1： シングルモード

Bits [4:2]： **ディバイダ分周比 [2:0]**

000 ：分周なし →1/1
001 ：4 分周 →1/4
010 ：8 分周 →1/8
011 ：16 分周 →1/16
100 ：32 分周 →1/32
101 ：64 分周 →1/64
110 ：128 分周 →1/128
111 ：256 分周 →1/256

Bit 1： **イミディエイトロードリクエスト（サイクリックモード時）**

サイクリックモード時にタイマ A2 のロード値をリロードするタイミングを設定します。
0： カウンタ値が“0x0000”になったときに、書き込まれたロード値をリロード
1： レジスタにロード値を書き込むと即時リロード

注意： シングルモード時は、このビットに関係なくロード値を書き込むとその値が即時反映されます。

Bit 0： **タイマ A2 割り込み要求イネーブル**

0： 割り込み要求ディセーブル（IRQ マスク）
1： 割り込み要求イネーブル

タイマ A2 IRQ フラグクリアレジスタ (TMA2IRQ)

TIM[0x4C] 初期値 = - Write Only

15	14	13	12	11	10	9	8
任意のデータ							
7	6	5	4	3	2	1	0
任意のデータ							

このレジスタはタイマ A2 の割り込み要求（IRQ）フラグをクリアするためのライトオンリのポートです。IRQ フラグをクリアするためには、このレジスタに任意のデータを書き込みます。

タイマ A2 ポート出力制御レジスタ (TMA2POUT)

TIM[0x50] 初期値 = 0x0000 (Read/Write)

15	14	13	12	11	10	9	8
				n/a RO			
7	6	5	4	3	2	1	0
n/a RO				出力モード選択 R/W		出力イネーブル R/W	出力値 R/W

Bits [3:2]： **出力モード選択**

00：アンダーフロー発生時に出力値（Bit 0）が有効
01：アンダーフロー出力
10：アンダーフロー発生毎にトグル出力
11：予約

Bit 1： **出力イネーブル**

0： 出力モードディセーブル
1： 出力モードイネーブル
該当するポートが GPIO として設定されているときに、このビットを“1”にするとポートは出力モード（Bits [3:2]）で設定したように動作します。

Bit 0： **出力値**

出力モード（Bits [3:2]）が“00”のときには、このビットの値が出力されます。

プリスケーラ 0　制御レジスタ　(PS0CTRL)

TIM[0xA0]　　初期値 = 0x0000　　(Read/Write)

ディバイダ分周比 [2:0]			n/a				
R/W			RO				
15	14	13	12	11	10	9	8
プリスケーラ 0 ロード値 [7:0]							
R/W							
7	6	5	4	3	2	1	0

Bits [15:13] :　**ディバイダ分周比 [2:0]**

000：分周なし　→1/1
001：4 分周　→1/4
010：8 分周　→1/8
011：16 分周　→1/16
100：32 分周　→1/32
101：64 分周　→1/64
110：128 分周　→1/128
111：256 分周　→1/256

Bits [7:0] :　**プリスケーラ 0 ロード値ビット[7:0]**

プリスケーラ 0 においてディバイダで分周されたクロックをカウントする値を設定するレジスタです。

プリスケーラ 1　制御レジスタ　(PS1CTRL)

TIM[0xA4]　　初期値 = 0x0000　　(Read/Write)

ディバイダ分周比			n/a				
R/W			RO				
15	14	13	12	11	10	9	8
プリスケーラ 1 ロード値[7:0]							
R/W							
7	6	5	4	3	2	1	0

Bits [15:13] :　**ディバイダ分周比 [2:0]**

000：分周なし　→1/1
001：4 分周　→1/4
010：8 分周　→1/8
011：16 分周　→1/16
100：32 分周　→1/32
101：64 分周　→1/64
110：128 分周　→1/128
111：256 分周　→1/256

Bits [7:0] :　**プリスケーラ 1 ロード値ビット [7:0]**

プリスケーラ 1 においてディバイダで分周されたクロックをカウントする値を設定するレジスタです。

タイマ A　IRQ ステータスレジスタ　(TMAIRQSTS)

TIM[0xB0]　　初期値 = 0x0000　　Read Only

n/a							
15	14	5	4	3	2	1	0
n/a			予約		Timer2 IRQ	Timer1 IRQ	Timer0 IRQ
7	6	5	4	3	2	1	0

Bits [4:3]：　**予約**

Bit 2：　**タイマ A2 IRQ ステータス**

タイマ A2 の割り込みステータスを示します。
　0： 割り込み要求なし（または、IRQ マスク）
　1： 割り込み要求あり
この割り込みステータスはタイマ A2 IRQ フラグクリアレジスタでクリアします。

Bit 1：　**タイマ A1 IRQ ステータス**

タイマ A1 の割り込みステータスを示します。
　0： 割り込み要求なし（または、IRQ マスク）
　1： 割り込み要求あり
この割り込みステータスはタイマ A1 IRQ フラグクリアレジスタでクリアします。

Bit 0：　**タイマ A0 IRQ ステータス**

タイマ A0 の割り込みステータスを示します。
　0： 割り込み要求なし（または、IRQ マスク）
　1： 割り込み要求あり
この割り込みステータスはタイマ A0 IRQ フラグクリアレジスタでクリアします。

22.5 各モードでのロード値の設定方法

22.5.1 タイマカウンタのモード

3ch のタイマは、それぞれ 2 種類のモードが設定できます。
各モードについてのロード値の設定方法を説明します。

（1）サイクリックモード
ロードレジスタにより設定したロード値から“0”までを繰り返しカウントダウンします。ロードのタイミングには 2 通りあります。

a) イミディエイトロードリクエストをした場合
タイマ A 制御レジスタの Bit 1（イミディエイトロードリクエストビット）の値を“1”にすることで即時にロード値をカウンタに反映することができます。
手順は、即時反映させたい場合にイミディエイトロードリクエストを“1”にし、次に反映させたいロード値をロードレジスタで設定します。タイマ A 制御レジスタの 7bit 目（タイマ A イネーブル）の状態に関わらず強制的にロード値を書き込むことができます。

b) イミディエイトロードリクエストをしない場合
タイマ A 制御レジスタの Bit 1（イミディエイトロードリクエストビット）の値が“0”であれば、ダウンカウンタの各ビットがすべて“0”になったときに割り込みを発生し、その時点でロードレジスタに設定してある値をリロードして再度ダウンカウントし始めます。
フリーランモードで使用したい場合は、ロード値に’FFFFh’の値を設定します。

（2）シングルモード
ロードレジスタにより設定したロード値からカウントダウンし、カウンタの出力がすべて“0”になったら割り込みを発生してストップします。IRQ フラグクリアレジスタで割り込みをクリアしても、カウンタはストップしたままです。

リロードのタイミングはタイマ A 制御レジスタの Bit 1（イミディエイトロードリクエストビット）の値に関係なく、ロードレジスタによりロード値を設定した時点で即時反映されます。タイマ A 制御レジスタの Bit 7（タイマ A イネーブルビット）の状態に関わらず強制的にロード値を書き込むことができます。

22.6 タイマ内部クロック設定例（1KHz、1MHz）

22.6.1 ディバイダとプリスケーラの設定方法

3ch のタイマは、それぞれディバイダと 2ch のプリスケーラで分周比とカウント数を設定して任意の 2 種類の周期でカウントダウンすることができます。

プリスケーラを使用せずディバイダの分周比でタイマを動作させる場合は 3ch のタイマそれぞれにディバイダ分周比を設定することができます。

タイマの入力クロック（TINCLK）は APB バスクロック（PCLK）の 1/8 クロックが入力されます。表 22.2 に 6MHzのTINCLK（PCLK=48MHzのとき）が入力された場合の 1ms (1KHz) および 1μs (1MHz) 周期の設定例を示します。該当するプリスケーラ制御レジスタを下記のように設定して下さい。なお、プリスケーラのロード値はカウントを 1/1 にする場合には“0x00”、1/2 にする場合には“0x01”、1/3 にする場合は“0x02”・・・として設定します。

表 22.2　ミリ＆マイクロ周期設定例*

周期	ディバイダ分周比	プリスケーラ ロード値	周波数
1 ms	Bits [15:13] = 100b (1/32 選択)	Bits [7:0] = 1011 1010b (0xBA) (1/187 選択)	1.002673KHz
	Bits [15:13] = 101b (1/64 選択)	Bits [7:0] = 0101 1101b (0x5D) (1/94 選択)	0.997340KHz
1 μs	Bits [15:13] = 000b (1/1 選択)	Bits [7:0] = 0000 0101b (0x05) (1/6 選択)	1.000000MHz

注意(*)：f_{TINCLK}=6MHz（f_{PCLK}=48MHz）が入力された場合の設定例です。

作成したい周波数を PCLK の周波数から算出する計算式は以下のようになります。

作成したい周波数に近くなるようにディバイダ分周比とプリスケーラロード値をプリスケーラ[1:0]制御レジスタに設定してください。

$$\text{作成したい周波数 } f = \underbrace{\frac{f_{PCLK} \div 8}{f_{TINCLK}}} \div \underbrace{\text{bit [15:13]*による設定値}}_{\text{ディバイダ分周比}} \div \underbrace{\text{bit [7:0]*による設定値}}_{\text{プリスケーラロード値}}$$

*：プリスケーラ[1:0]制御レジスタ（TIM[0xA0], TIM[0xA4]）

表 22.2 による計算例

例 1） 1KHz ÷ 48MHz ÷ 8 ÷ 32 ÷ 187 ＝ 1.002673KHz

例 2） 1KHz ÷ 48MHz ÷ 8 ÷ 64 ÷ 94 ＝ 0.997340KHz

例 3） 1MHz = 48MHz ÷ 8 ÷ 1 ÷ 6 ＝ 1.000000MHz

22.7 タイミング図

22.7.1 サイクリックモード時のイミディエイトロードリクエスト

サイクリックモード時にイミディエイトロードリクエストを行い、即時にロード値を反映した場合のタイミングチャートを下記に示します。カウンタのイネーブルサイクルをプリスケーラにて 1/3 に設定しています。

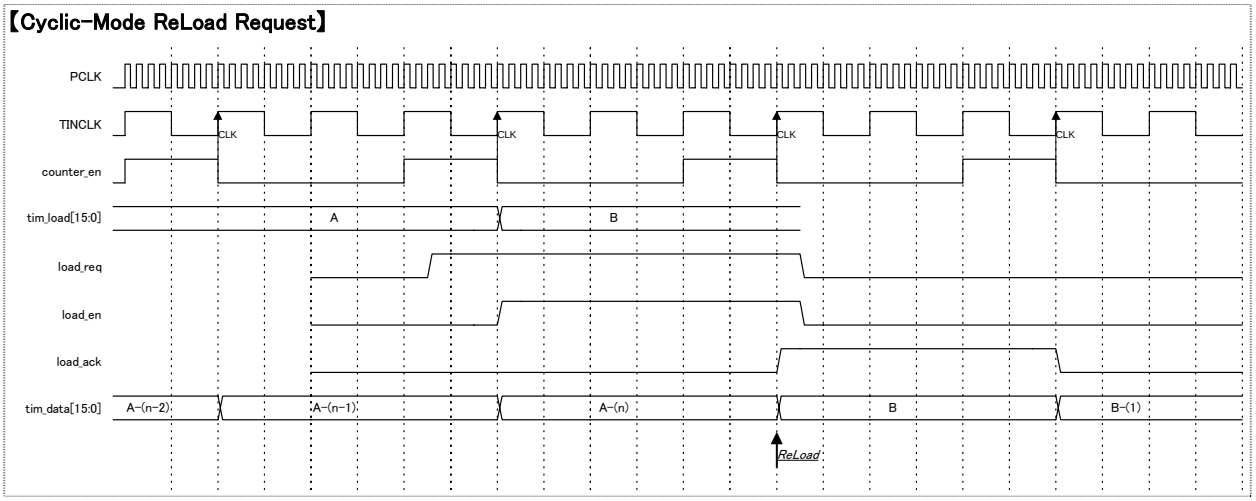


図 22.2　サイクリックモード時のイミディエイトロードリクエスト

22.7.2 サイクリックモード時の通常リロード

サイクリックモード時にイミディエイトロードリクエストを行わず、カウンタ出力がすべて“0”になったときにリロードする通常の循環カウントダウンのタイミングチャートと割り込みの発生を、分周なしの場合のタイミングで下記に記します。

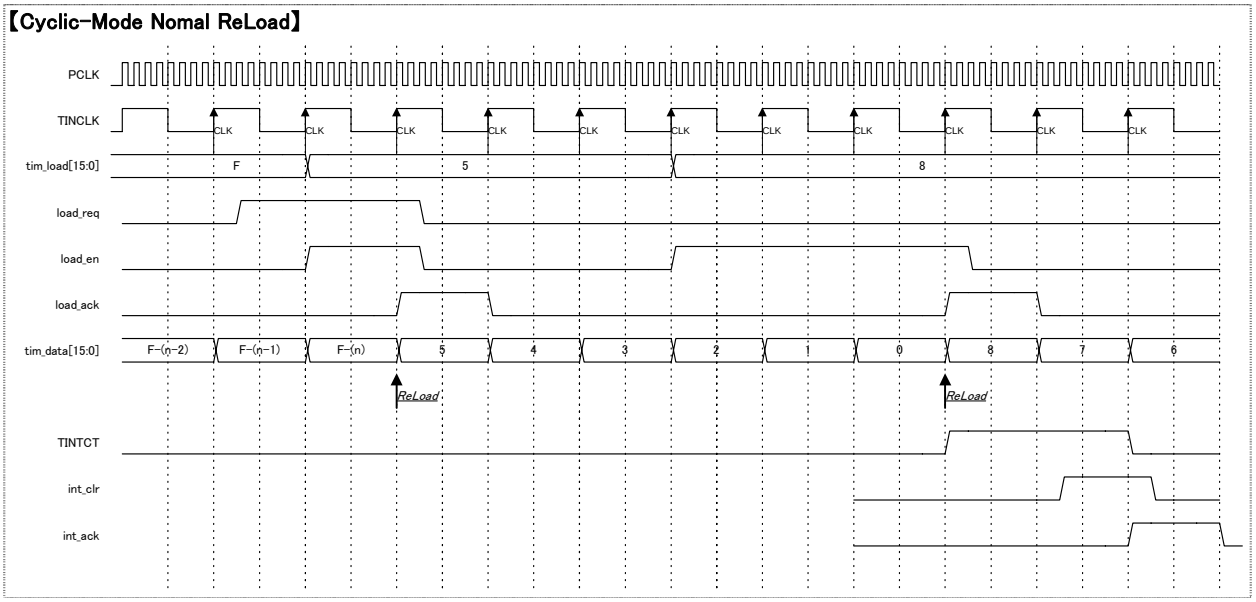


図 22.3　サイクリックモード時の通常リロード

22.7.3 シングルモード時の通常リロード

シングルモード時にカウンタ出力がすべて“0”になった場合の割り込みの発生と、ロード値を書き込んだ場合のタイミングチャートを、分周なしの場合のタイミングで下記に記します。

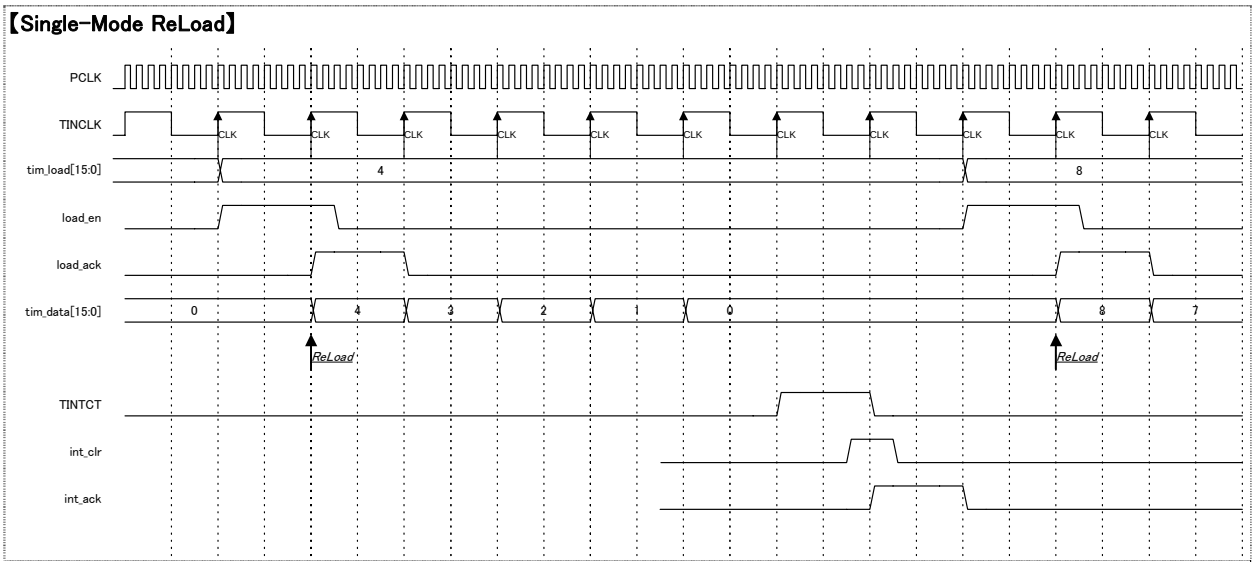


図 22.4　シングルモード時の通常リロード

22.7.4 ポート出力

アンダーフロー発生時にポート出力する場合のタイミングチャートをモードごとに記します。
以下に記載している波形はすべてディバイダおよびプリスケーラの設定値をデフォルト（1/1）にしたものです。

（1）アンダーフロー発生時に出力値（bit0）を出力（出力モード＝“00”）
アンダーフローが発生すると、ポート出力制御レジスタの Bit 0（出力値ビット）の値を保持しながら出力します。

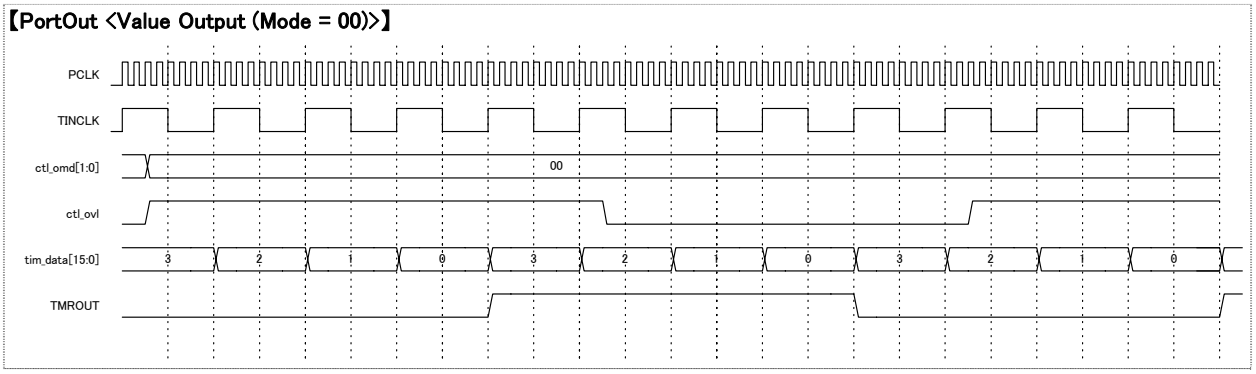


図 22.5　アンダーフロー発生時の出力（出力値）

（２）アンダーフローを出力（出力モード＝“01”）

アンダーフローが発生するたびにアンダーフロー自身を出力します。

TMROUT のアンダーフローのパルス幅はダウンカウントのデータ幅と同じです。カウント値が“0”から Load 値（図中では“3”）になるタイミングでアンダーフローが出力されます。

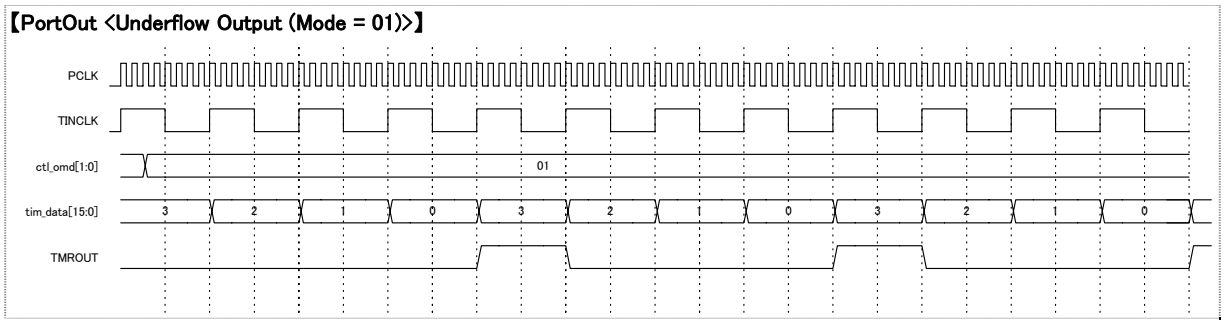


図 22.6　アンダーフロー発生時の出力（アンダーフロー出力）

（３）アンダーフロー発生時にトグル出力（出力モード＝“10”）

アンダーフローが発生するたびに、信号を反転させながら出力します。

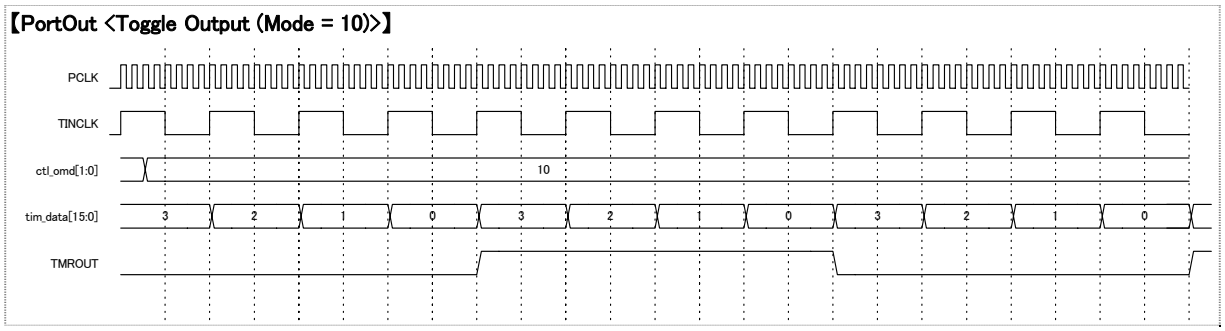


図 22.7　アンダーフロー発生時の出力（トグル出力）

23. タイマＢ (TIMB)

23.1 概要

タイマＢは、アウトプットコンペア機能、インプットキャプチャ機能を内蔵した 16 ビットのタイマです。また、外部入力パルスによりイベントカウンタとして動作させることが可能なほか、タイマカウンタと 4 本の common レジスタのコンペア信号により任意のデューティ比パルス出力を実現できる多機能タイマです。

23.2 特徴

- カウンタ入力クロック：
  外部クロックまたは内部システムクロックの 1/16～1/2048
- 最大 4 本のパルス出力処理が可能
- common レジスタ：4 本
  それぞれをアウトプットレジスタまたはインプットキャプチャレジスタとして設定可能
- タイマ入出力機能：
  アウトプットコンペア：0 出力/1 出力/トグル出力が可能
  インプットキャプチャ：立ち上がりエッジ/立ち下がりエッジ/両エッジを検出可能
  PWM 出力が可能
- インプットキャプチャ動作時は、現在のカウント値と一つ前のカウント値を保持
- 外部入力パルスによるイベントカウント動作が可能
- 5 種類の割り込み要因：
- コンペアマッチ/インプットキャプチャ兼用割り込み x4 要因、オーバーフロー割り込み

23.3 ブロック図

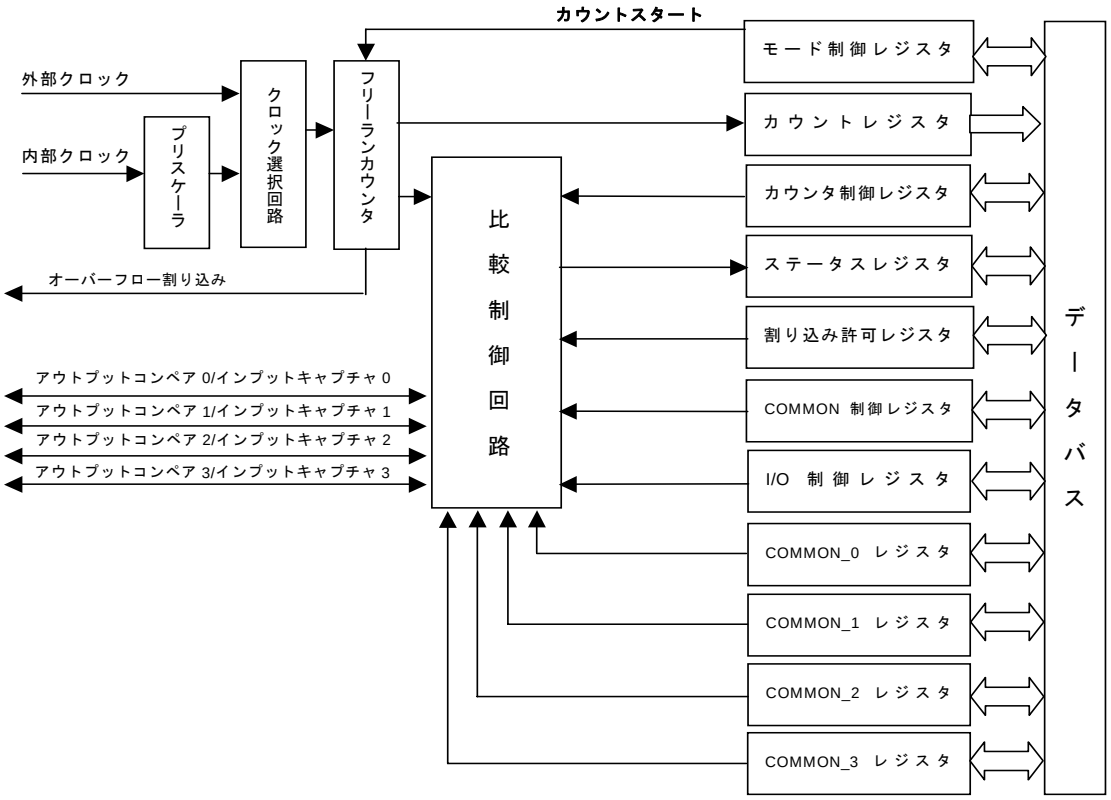


図 23.1 タイマ B ブロック図

23.4 外部端子

タイマ B 関連の外部端子は以下の通りです。

端子名	入出力	端子機能	マルチプレクス端子*／備考
TimerBIn	入力	外部クロック入力端子	GPIOB7
TimerB0IO	入出力	common_0 レジスタ： アウトプットコンペア出力/インプットキャプチャ入力端子	GPIOC0
TimerB1IO	入出力	common_1 レジスタ： アウトプットコンペア出力/インプットキャプチャ入力端子	GPIOC1
TimerB2IO	入出力	common_2 レジスタ： アウトプットコンペア出力/インプットキャプチャ入力端子	GPIOC2
TimerB3io	入出力	common_3 レジスタ： アウトプットコンペア出力/インプットキャプチャ入力端子	GPIOC3/ UART3_CLK

注意(*)： タイマ B 用の外部端子は GPIO 端子等とマルチプレクスされていますので、GPIO 端子機能レジスタにより“GPIO 以外の機能 1”に設定することにより使用できます。

23.5 レジスタ

23.5.1 レジスター覧

これらのレジスタのベースアドレスは、0xFFFD_3000 です。

表 23.1　レジスター覧（ベースアドレス：0xFFFD_3000）

アドレスオフセット	レジスタ名称	レジスタ略称	初期値	R/W	データアクセスサイズ
0x00	タイマ B　モード制御レジスタ	TIMBMDC	0x0000	R/W	16 (/32)
0x04	タイマ B　カウンタ制御レジスタ	TIMBCNTC	0x0000	R/W	16 (/32)
0x08	タイマ B　ステータスレジスタ	TIMBSTS	0x0000	R/W	16 (/32)
0x0C	タイマ B　割り込み許可レジスタ	TIMBINTEN	0x0000	R/W	16 (/32)
0x10	タイマ B　カウントレジスタ	TIMBCNT	0x0000	R/W	16 (/32)
0x14	タイマ B　COMMON 制御レジスタ	TIMBCOMC	0x0000	R/W	16 (/32)
0x18	タイマ B　I/O 制御レジスタ	TIMBIO	0x0000	R/W	16 (/32)
0x1C	タイマ B　周期レジスタ	TIMBDUTY	0x0000	R/W	16 (/32)
0x20	タイマ B　COMMON_0 レジスタ	TIMBCOM0	0x0000	R/W	16 (/32)
0x24	タイマ B　COMMON_1 レジスタ	TIMBCOM1	0x0000	R/W	16 (/32)
0x28	タイマ B　COMMON_2 レジスタ	TIMBCOM2	0x0000	R/W	16 (/32)
0x2C	タイマ B　COMMON_3 レジスタ	TIMBCOM3	0x0000	R/W	16 (/32)
0x30	タイマ B　COMINP_0 レジスタ	TIMBINP0	0x0000	RO	16 (/32)
0x34	タイマ B　COMINP_1 レジスタ	TIMBINP1	0x0000	RO	16 (/32)
0x38	タイマ B　COMINP_2 レジスタ	TIMBINP2	0x0000	RO	16 (/32)
0x3C	タイマ B　COMINP_3 レジスタ	TIMBINP3	0x0000	RO	16 (/32)

23.5.2 レジスタ詳細説明

タイマＢ　モード制御レジスタ　(TIMBMDC)							
TIMB[0x00]　初期値 = 0x0000							Read/Write
clr_B3o	clr_B2o	clr_B1o	clr_B0o	set_B3o	set_B2o	set_B1o	set_B0o
15	14	13	12	11	10	9	8
Reserved				B3pwm	B2pwm	B1pwm	B0pwm
7	6	5	4	3	2	1	0

Bit 15 : **clr_B3o　タイマ出力３強制クリア**
(W) 1 : TimerB3io 出力を強制的に Low にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB3io の出力状態を表します（インプットキャプチャ時は無効）

Bit 14 : **clr_B2o　タイマ出力２強制クリア**
(W) 1 : TimerB2io 出力を強制的に Low にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB2io の出力状態を表します（インプットキャプチャ時は無効）

Bit 13 : **clr_B1o　タイマ出力１強制クリア**
(W) 1 : TimerB1io 出力を強制的に Low にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB1io の出力状態を表します（インプットキャプチャ時は無効）

Bit 12 : **clr_B0o　タイマ出力０強制クリア**
(W) 1 : TimerB0io 出力を強制的に Low にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB0io の出力状態を表します（インプットキャプチャ時は無効）

Bit 11 : **set_B3o　タイマ出力３強制セット**
(W) 1 : TimerB3io 出力を強制的に High にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB3io の出力状態を表します（インプットキャプチャ時は無効）

Bit 10 : **set_B2o　タイマ出力２強制セット**
(W) 1 : TimerB2io 出力を強制的に High にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB2io の出力状態を表します（インプットキャプチャ時は無効）

Bit 9 : **set_B1o　タイマ出力１強制セット**
(W) 1 : TimerB1io 出力を強制的に High にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB1io の出力状態を表します（インプットキャプチャ時は無効）

Bit 8 : **set_B0o　タイマ出力０強制セット**
(W) 1 : TimerB0io 出力を強制的に High にします（インプットキャプチャ時は無効）
0 : 無効
(R) : 現在の TimerB0io の出力状態を表します（インプットキャプチャ時は無効）

Bits [7 : 4] : **Reserved**

Bits 3 : **B3pwm　PWM モード３（TimerB3io 出力）**
0 : アウトプットコンペア出力
1 : PWM 出力

Bits 2 : **B2pwm　PWM モード２（TimerB2io 出力）**
0 : アウトプットコンペア出力
1 : PWM 出力

Bits 1： **B1pwm　PWM モード 1（TimerB1io 出力）**
0： アウトプットコンペア出力
1： PWM 出力

Bits 0： **B0pwm　PWM モード 0（TimerB0io 出力）**
0： アウトプットコンペア出力
1： PWM 出力

タイマ B　カウンタ制御レジスタ　(TIMBCNTC)
TIMB[0x04]　　初期値 = 0x0000　　Read/Write

BcntTST	Reserved			Bclksel			
15	14	13	12	11	10	9	8
BcntCLR	Reserved						BcntST
7	6	5	4	3	2	1	0

Bits 15： **BcntTST　カウンタテストモード（Reserved）**
1 を書き込まないで下さい。

Bits [14:12]： **Reserved**

Bits [11：8]： **Bclksel　クロックセレクト※**
0000：内部システムクロックの 16 分周
0001：内部システムクロックの 32 分周
0010：内部システムクロックの 64 分周
0011：内部システムクロックの 128 分周
0100：内部システムクロックの 256 分周
0101：内部システムクロックの 512 分周
0110：内部システムクロックの 1024 分周
0111：内部システムクロックの 2048 分周
1xxx：外部クロック

※ 内部システムクロックからの分周比はシステムコントローラで行われる 8 分周を含みます

Bits 7： **BcntCLR　カウントモード設定**
0： フリーランカウンタとして動作（0xFFFF までカウントアップし、0x0000 に戻る）
1： 周期カウンタとして動作（周期レジスタ TIMBDUTY までカウントアップし、0x0000 に戻る

Bits [6:1]： **Reserved**

Bits 0： **BcntST　カウンタスタート**
0： タイマカウント停止
1： タイマカウント動作

タイマ B　ステータスレジスタ　(TIMBSTS)

TIMB[0x08]　　初期値 = 0x0000　　Read/Write

Reserved						Bmatch	Bovf
15	14	13	12	11	10	9	8
Reserved				B3sts	B2sts	B1sts	B0sts
7	6	5	4	3	2	1	0

Bits [15：10]：　**Reserved**

Bits 9：　**Bmatch　周期レジスタコンペアマッチステータス**

(R) 1：周期レジスタの値とタイマカウンタ値が一致したとき。
(R) 0：カウンタの一致はありません。

(W) 1：ステータスレジスタのクリア
(W) 0：無効

Bits 8：　**Bovf　オーバーフロー**

(R) 1：タイマカウンタが H'FFFF から H'0000 にオーバーフローしたとき
(R) 0：オーバーフローはありません。

(W) 1：ステータスレジスタのクリア
(W) 0：無効

Bits [7：4]：　**Reserved**

Bits 3：　**B3sts　TimerB3io インプットキャプチャ/コンペアマッチステータス**

(R) 1：アウトプットコンペア設定でカウンタ値と一致したき。
　　インプットキャプチャ設定でカウンタ値が TIMBCOM3 に転送されたとき。
(R) 0：ステータスフラグの割り込みはありません。

(W) 1：ステータスレジスタのクリア
(W) 0：無効

Bits 2：　**B2sts　TimerB2io インプットキャプチャ/コンペアマッチステータス**

(R) 1：アウトプットコンペア設定でカウンタ値と一致したき。
　　インプットキャプチャ設定でカウンタ値が TIMBCOM2 に転送されたとき。
(R) 0：ステータスフラグの割り込みはありません。

(W) 1：ステータスレジスタのクリア
(W) 0：無効

Bits 1：　**B1sts　TimerB1io インプットキャプチャ/コンペアマッチステータス**

(R) 1：アウトプットコンペア設定でカウンタ値と一致したき。
　　インプットキャプチャ設定でカウンタ値が TIMBCOM1 に転送されたとき。
(R) 0：ステータスフラグの割り込みはありません。

(W) 1：ステータスレジスタのクリア
(W) 0：無効

Bits 0：　**B0sts　TimerB0io インプットキャプチャ/コンペアマッチステータス**

(R) 1：アウトプットコンペア設定でカウンタ値と一致したき。
　　インプットキャプチャ設定でカウンタ値が TIMBCOM0 に転送されたとき。
(R) 0：ステータスフラグの割り込みはありません。

(W) 1：ステータスレジスタのクリア
(W) 0：無効

タイマ B　割り込み許可レジスタ　(TMBINTEN)

TIMB[0x0C]　　初期値 = 0x0000　　Read/Write

15	14	13	12	11	10	9	8
Reserved						BmatchEN	BovfEN
7	6	5	4	3	2	1	0
reserved				B3intEN	B2intEN	B1intEN	B0intEN

Bits [15：10]：　**Reserved**

Bits 9：　**BmatchEN　周期レジスタコンペアマッチ割り込み制御フラグ**
1： 割り込み許可
0： 割り込み禁止

Bits 8：　**BovfEN　TimerBovf オーバーフロー割り込み制御フラグ**
1： 割り込み許可
0： 割り込み禁止

Bits [7：4]：　**Reserved**

Bits 3：　**B3intEN　TimerB3io インプットキャプチャ/コンペアマッチ割り込み制御フラグ**
1： 割り込み許可
0： 割り込み禁止

Bits 2：　**B2intEN　TimerB2io インプットキャプチャ/コンペアマッチ割り込み制御フラグ**
1： 割り込み許可
0： 割り込み禁止

Bits 1：　**B1intEN　TimerB1io インプットキャプチャ/コンペアマッチ割り込み制御フラグ**
1： 割り込み許可
0： 割り込み禁止

Bits 0：　**B0intEN　TimerB0io インプットキャプチャ/コンペアマッチ割り込み制御フラグ**
1： 割り込み許可
0： 割り込み禁止

タイマ B　カウントレジスタ　(TIMBCNT)

TIMB[0x10]　　初期値 = 0x0000　　Read/Write

15	14	13	12	11	10	9	8
Bcount [15:8]							
7	6	5	4	3	2	1	0
Bcount [7:0]							

Bits [15:0]：　**Bcount　カレントカウント値ビット[15:0]**
タイマ B の現在のカウント値が読み出せます。
タイマカウントが停止中（BcntST が 0）のときは、カウント値を任意の値に設定できます。

タイマ B　COMMON 制御レジスタ　(TIMBCOMC)

TIMB[0x14]　　初期値 = 0x0000　　Read/Write

Reserved							
15	14	13	12	11	10	9	8
Reserved				B3comFUNC	B2comFUNC	B1comFUNC	B0comFUNC
7	6	5	4	3	2	1	0

Bits [15:4]：　**Reserved**

Bits 3：　**B3comFUNC　TIMBCOM3 の機能選択**
0：アウトプットコンペアレジスタとして機能
1：インプットキャプチャレジスタとして機能

Bits 2：　**B2comFUNC　TIMBCOM2 の機能選択**
0：アウトプットコンペアレジスタとして機能
1：インプットキャプチャレジスタとして機能

Bits 1：　**B1comFUNC　TIMBCOM1 の機能選択**
0：アウトプットコンペアレジスタとして機能
1：インプットキャプチャレジスタとして機能

Bits 0：　**B0comFUNC　TIMBCOM0 の機能選択**
0：アウトプットコンペアレジスタとして機能
1：インプットキャプチャレジスタとして機能

タイマ B　I/O 制御レジスタ　(TIMBIO)

TIMB[0x18]　　初期値 = 0x0000　　Read/Write

Reserved[15:8]							
15	14	13	12	11	10	9	8
B3comIO		B2comIO		B1comIO		B0comIO	
7	6	5	4	3	2	1	0

Bits [15:8]：　**Reserved**

Bits [7:6]：　**B3comIO**
TIMBCOM3 がアウトプットコンペアレジスタ機能の場合
00：コンペアマッチによる端子出力禁止
01：コンペアマッチで TimerB3io に 0 出力
10：コンペアマッチで TimerB3io に 1 出力
11：コンペアマッチで TimerB3io にトグル出力
TIMBCOM3 がインプットキャプチャレジスタ機能の場合
00：TimerB3io の立ち上がりエッジで TIMBCOM3 にインプットキャプチャ
01：TimerB3io の立ち下がりエッジで TIMBCOM3 にインプットキャプチャ
10：TimerB3io の両エッジで TIMBCOM3 にインプットキャプチャ
11：Reserved

Bits [5:4] : **B2comIO**

TIMBCOM2 がアウトプットコンペアレジスタ機能の場合
- 00：コンペアマッチによる端子出力禁止
- 01：コンペアマッチで TimerB2io に 0 出力
- 10：コンペアマッチで TimerB2io に 1 出力
- 11：コンペアマッチで TimerB2io にトグル出力

TIMBCOM2 がインプットキャプチャレジスタ機能の場合
- 00：TimerB2io の立ち上がりエッジで TIMBCOM2 にインプットキャプチャ
- 01：TimerB2io の立ち下がりエッジで TIMBCOM2 にインプットキャプチャ
- 10：TimerB2io の両エッジで TIMBCOM2 にインプットキャプチャ
- 11：Reserved

Bits [3:2] : **B1comIO**

TIMBCOM1 がアウトプットコンペアレジスタ機能の場合
- 00：コンペアマッチによる端子出力禁止
- 01：コンペアマッチで TimerB1io に 0 出力
- 10：コンペアマッチで TimerB1io に 1 出力
- 11：コンペアマッチで TimerB1io にトグル出力

TIMBCOM1 がインプットキャプチャレジスタ機能の場合
- 00：TimerB1io の立ち上がりエッジで TIMBCOM1 にインプットキャプチャ
- 01：TimerB1io の立ち下がりエッジで TIMBCOM1 にインプットキャプチャ
- 10：TimerB1io の両エッジで TIMBCOM1 にインプットキャプチャ
- 11：Reserved

Bits [1:0] : **B0comIO**

TIMBCOM0 がアウトプットコンペアレジスタ機能の場合
- 00：コンペアマッチによる端子出力禁止
- 01：コンペアマッチで TimerB0io に 0 出力
- 10：コンペアマッチで TimerB0io に 1 出力
- 11：コンペアマッチで TimerB0io にトグル出力

TIMBCOM0 がインプットキャプチャレジスタ機能の場合
- 00：TimerB0io の立ち上がりエッジで TIMBCOM0 にインプットキャプチャ
- 01：TimerB0io の立ち下がりエッジで TIMBCOM0 にインプットキャプチャ
- 10：TimerB0io の両エッジで TIMBCOM0 にインプットキャプチャ
- 11：Reserved

タイマ B　周期レジスタ　(TIMBDUTY)

TIMB[0x1C]　　初期値 = 0x0000　　Read/Write

15	14	13	12	11	10	9	8
Bduty[15:8]							
7	6	5	4	3	2	1	0
Bduty [7:0]							

Bits [15:0] : **Bduty　デューティレジスタ**

周期カウントモードにおける周期カウント値を設定します。

タイマ B　COMMON_0 レジスタ　(TIMBCOM0)

TIMB[0x20]　　初期値 = 0x0000　　Read/Write

B0com[15:8]							
15	14	13	12	11	10	9	8
B0com [7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B0com　COMMON_0 レジスタ**

アウトプットコンペアモード時のカウンタ値を設定します。
インプットキャプチャモード時は、一つ前の B0inp の内容が保持されます。

タイマ B　COMMON_1 レジスタ　(TIMBCOM1)

TIMB[0x24]　　初期値 = 0x0000　　Read/Write

B1com[15:8]							
15	14	13	12	11	10	9	8
B1com [7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B1com　COMMON_1 レジスタ**

アウトプットコンペアモード時のカウンタ値を設定します。
インプットキャプチャモード時は、一つ前の B1inp の内容が保持されます。

タイマ B　COMMON_2 レジスタ　(TIMBCOM2)

TIMB[0x28]　　初期値 = 0x0000　　Read/Write

B2com[15:8]							
15	14	13	12	11	10	9	8
B2com[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B2com　COMMON_2 レジスタ**

アウトプットコンペアモード時のカウンタ値を設定します。
インプットキャプチャモード時は、一つ前の B2inp の内容が保持されます。

タイマ B　COMMON_3 レジスタ　(TIMBCOM3)

TIMB[0x2C]　　初期値 = 0x0000　　Read/Write

B3com[15:8]							
15	14	13	12	11	10	9	8
B3com[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B3com　COMMON_3 レジスタ**

アウトプットコンペアモード時のカウンタ値を設定します。
インプットキャプチャモード時は、一つ前の B3inp の内容が保持されます。

タイマ B　COMINP_0 レジスタ　(TIMBINP0)

TIMB[0x30]　　初期値 = 0x0000　　Read Only

B0inp[15:8]							
15	14	13	12	11	10	9	8
B0inp [7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B0inp　COMINP_0 レジスタ**

インプットキャプチャモード時にカウンタ値が入力されます。

タイマ B　COMINP_1 レジスタ　(TIMBINP1)

TIMB[0x34]　　初期値 = 0x0000　　Read Only

B1inp[15:8]							
15	14	13	12	11	10	9	8
B1inp [7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B1inp　COMINP_1 レジスタ**

インプットキャプチャモード時にカウンタ値が入力されます。

タイマ B　COMINP_2 レジスタ　(TIMBINP2)

TIMB[0x38]　　初期値 = 0x0000　　Read Only

B2inp[15:8]							
15	14	13	12	11	10	9	8
B2inp[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B2inp　COMINP_2 レジスタ**

インプットキャプチャモード時にカウンタ値が入力されます。

タイマ B　COMINP_3 レジスタ　(TIMBINP3)

TIMB[0x3C]　　初期値 = 0x0000　　Read Only

B3inp[15:8]							
15	14	13	12	11	10	9	8
B3inp[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]：　**B3inp　COMINP_3 レジスタ**

インプットキャプチャモード時にカウンタ値が入力されます。

23.6 動作説明

23.6.1 カウント動作

タイマカウンタの動作は、16 ビットアップカウンタを使ったフリーランニングカウント動作または周期カウント動作の 2 種類になります。

（1）フリーランニングカウント動作

カウンタ制御レジスタ(TIMBCNTC)のカウントモード設定ビット（Bits[7]）を 0 に、カウンタスタートビット(Bits[0])を 1 にセットするとフリーランニングカウント動作を開始します。カウントクロックは、カウンタ制御レジスタ（TIMBCNTC）のクロックセレクトビット(Bits[3:1])の値により内蔵プリスケーラで分周されます。カウンタが H`FFFF から H`0000 にオーバーフローすると、ステータスレジスタ(TIMBSTS)のオーバーフロービット（Bovf Bits[8]）に 1 がセットされ、割り込み許可レジスタ(TIMBINTEN)のオーバーフロー制御フラグ(Bits[8])が割り込み許可設定になっていれば割り込み要求(TimerBovf)を発生します。

割り込み要求(Bovf)クリアはソフトウェアによりおこないます。フリーランニングカウント動作を図 23.2 に示します。

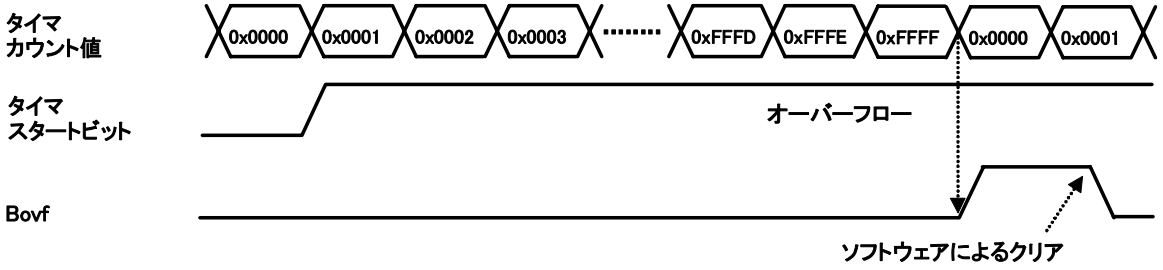


図 23.2 フリーランニングカウント動作

（2）周期カウント動作

カウンタ制御レジスタ(TIMBCNTC)のカウントモード設定ビット(Bits[7])に 1 をセットすると、周期レジスタの値で周期カウント動作をおこないます。カウンタの値が周期レジスタの値と一致するとタイマカウンタはH`0000 にクリアされ、ステータスレジスタ(TIMBSTS)の周期レジスタコンペアマッチ(Bits[9])に 1 がセットされ、割り込み許可レジスタ(TIMBINTEN)の周期レジスタコンペアマッチ割り込み制御フラグ(Bits[9])が割り込み許可設定になっていれば割り込み要求(TimerBovf)を発生します。割り込み要求(Bmatch)クリアはソフトウェアによりおこないます。周期レジスタの値がH`12FFの場合の周期カウント動作を 図 23.3 に示します。

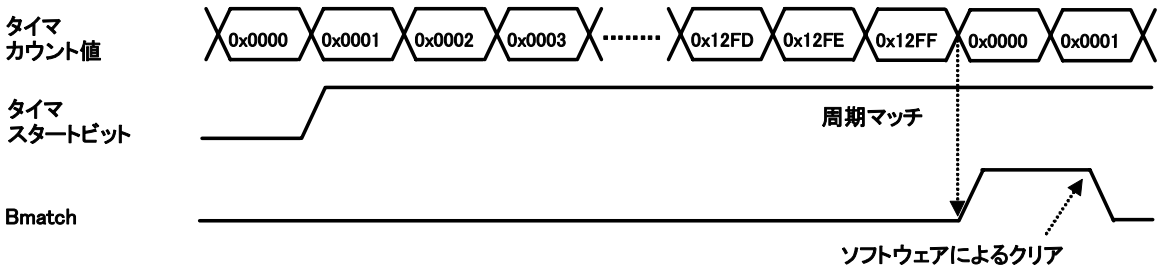


図 23.3 周期カウント動作（周期レジスタの値が H`12FF の場合）

23.6.2 タイマ入出力機能

タイマ B は、入出力機能として以下の 3 つのモードを持っています。各モードは端子（TimerB0/1/2/3io）毎に設定することが可能です。

・アウトプットコンペア
・PWM 出力
・インプットキャプチャ

（1）アウトプットコンペア動作

COMMON 制御レジスタ(TIMBCOM)の TimerB0io、TimerB1io、TimerB2io、TimerB3io の機能選択をアウトプットコンペアレジスタに設定し、I/O 制御レジスタ(TIMBIO)の機能選択をおこなうことにより、TimerB0io、TimerB1io、TimerB3io 端子から 0 出力/1 出力/トグル出力をおこなうことができます。

タイマカウンタをフリーランニング動作させ、COMMON_0 レジスタの値を H`AAAA、COMMON_1 レジスタの値を H`BBBB、COMMON_2 レジスタの値を H`AAAA に設定し、TimerB0io を 1 出力、TimerB1io を 0 出力、TimerB2io をトグル出力にした場合の動作を図 23.4 に示します。

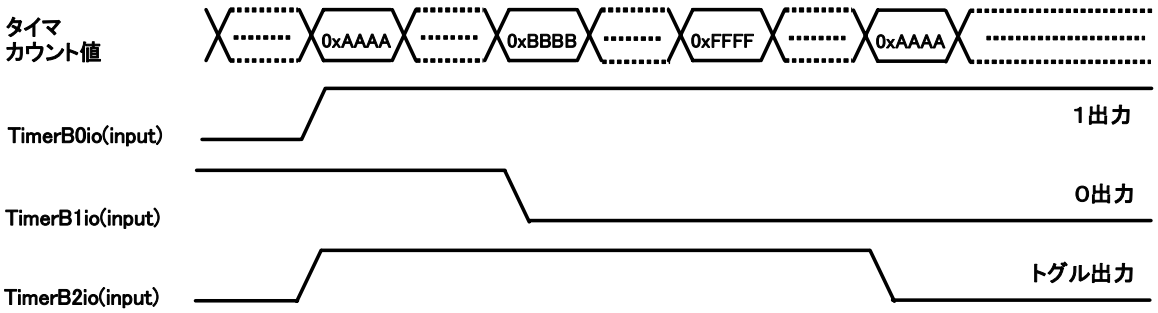


図 23.4 出力の設定

TimerB*io の出力変化に関わらず、COMMON レジスタとカウンタ値が一致すると B*sts の割込み要因が発生します。

（２）PWM 動作

PWM モードは、周期レジスタ(TIMBDUTY)に対する COMMON レジスタ（TIMBCOM0、TIMBCOM1、TIMBCOM2,TIMBCOM3）の値により TimerB0io、TimerB1io、TimerB2io、TimerB3io 出力端子よりそれぞれの PWM 波形を出力します。最大 4 相の PWM 出力が可能です。モード制御レジスタ(TIMBMDC)で PWM モード出力機能(Bits[3:0])を選択した場合、選択された COMMON レジスタは PWM のコンペアレジスタとして機能します。周期レジスタ(TIMBDUTY)と COMMON レジスタの値が同一の場合、コンペアマッチが発生しても出力値は変化しません。

周期レジスタを H`FF00、TIMBCOM0 を H`AA00、TIMBCOM1 を H`BB00 に設定し、TimerB0io、TimerB1io の出力レベルが“High”だった場合の動作を図 23.5 に示します。

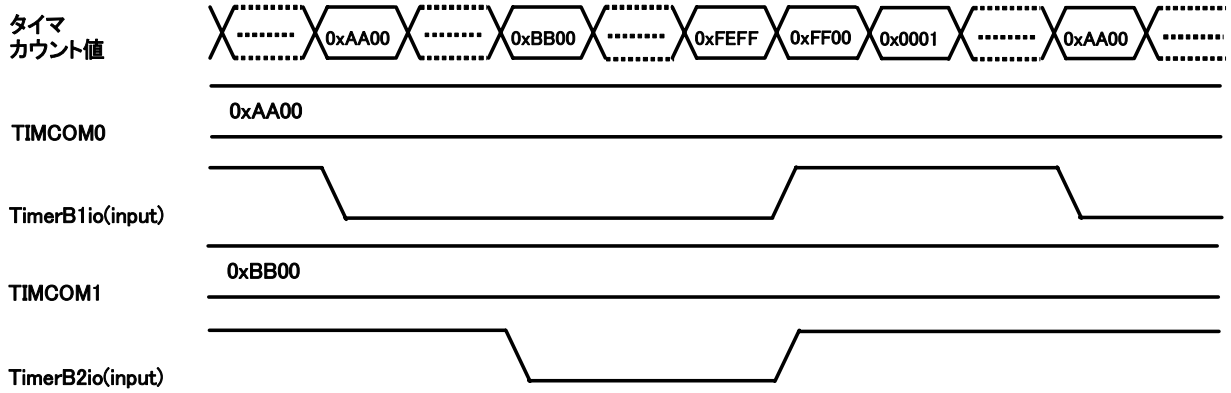


図 23.5　PWM 動作

（3）インプットキャプチャ動作

COMMON 制御レジスタ(TIMBCOMC)の TimerB0io、TimerB1io、TimerB2io、TimerB3io の機能選択をインプットキャプチャレジスタに設定することにより、TimerB0io、TimerB1io、TimerB2io、TimerB3io 端子の入力エッジを検出してタイマカウンタの値を COMINP_0 レジスタ、COMINP_1 レジスタ、COMINP_2 レジスタ、COMINP_3 レジスタに転送できます。また一つ前の COMINP_*レジスタの内容が、COMMON_*レジスタに退避されます。従って両レジスタの内容と、オーバーフロー割込みが発生した回数より、入力信号のエッジ間隔を正確に計算することができます。検出エッジは、I/O 制御レジスタ(TIMBIO)の機能を設定することにより、立ち上がり/立下り/両エッジを選択できます。これらの機能を使って、パルス幅や周期の測定をおこなうことができます。

タイマカウンタをフリーランニング動作に設定し、TimerB0ioのインプットキャプチャ機能を両エッジ、TimerB1ioのインプットキャプチャ機能を立ち上がりエッジに設定した場合の動作を 図 23.6 に示します。

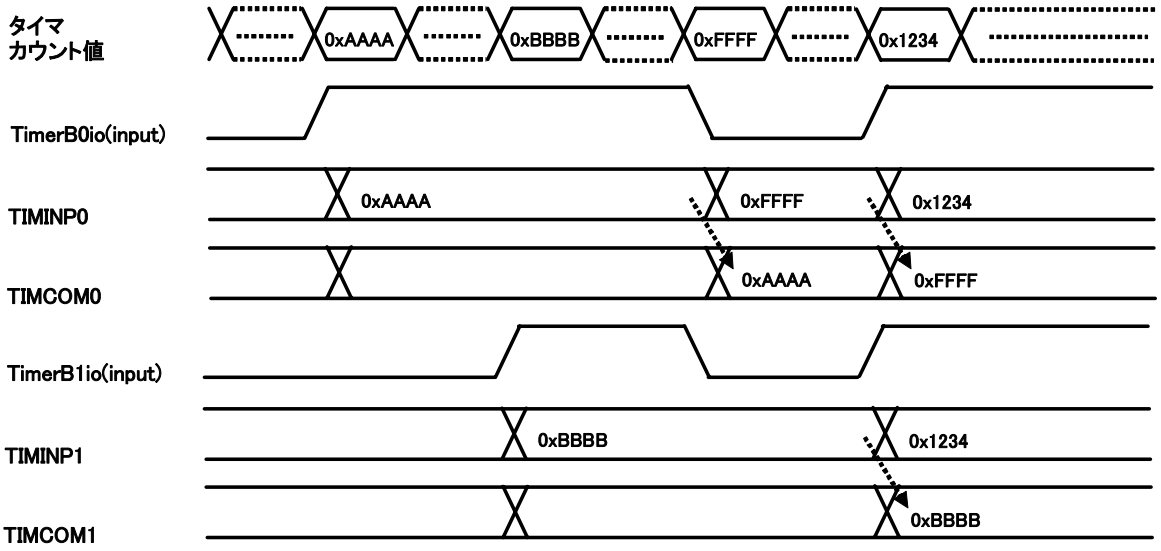


図 23.6　動作設定

24. リアルタイムクロック (RTC)

24.1 概要

リアルタイムクロック(以降 RTC)は 32,768Hz を入力クロックとして､256Hz を出力するプリスケーラ、8bit の分周タイマ、秒、分、時、日、月、年カウンタにより時刻を計時し、時計やストップウォッチなどの各種の計時機能を実現します。各データはソフトウェアによって読み出すことができます。また、32Hz、8Hz、2Hz、1 秒、1 分、1 時間、1 日のカウントアップによる割り込みを発生させることができ、周期割り込みやウェイクアップソースとして使用できます。さらに、分、時、日、月、年指定によるアラームを発生させることができ、ウェイクアップソースやアラーム機能として使用できます。RTC は 32,768Hz のクロックが動作していれば CPU および他の内蔵周辺回路がスタンバイ状態でも動作可能です。またシステムリセットによって計時に影響がでませんので、外部からリセット動作が入っても計時をつづけることができます。

メモリとして 8Byte のバックアップワーク用 RAM を内蔵しています。

RTC は消費電力低減を目的としたバックアップモードを用意しております。すべての電源が供給されている状態から、BUP#端子を“0”に、RTCVDD 以外の電源供給を止めることで実現します。このとき RTC にアクセスすることができなくなります。

24.2 ブロック図

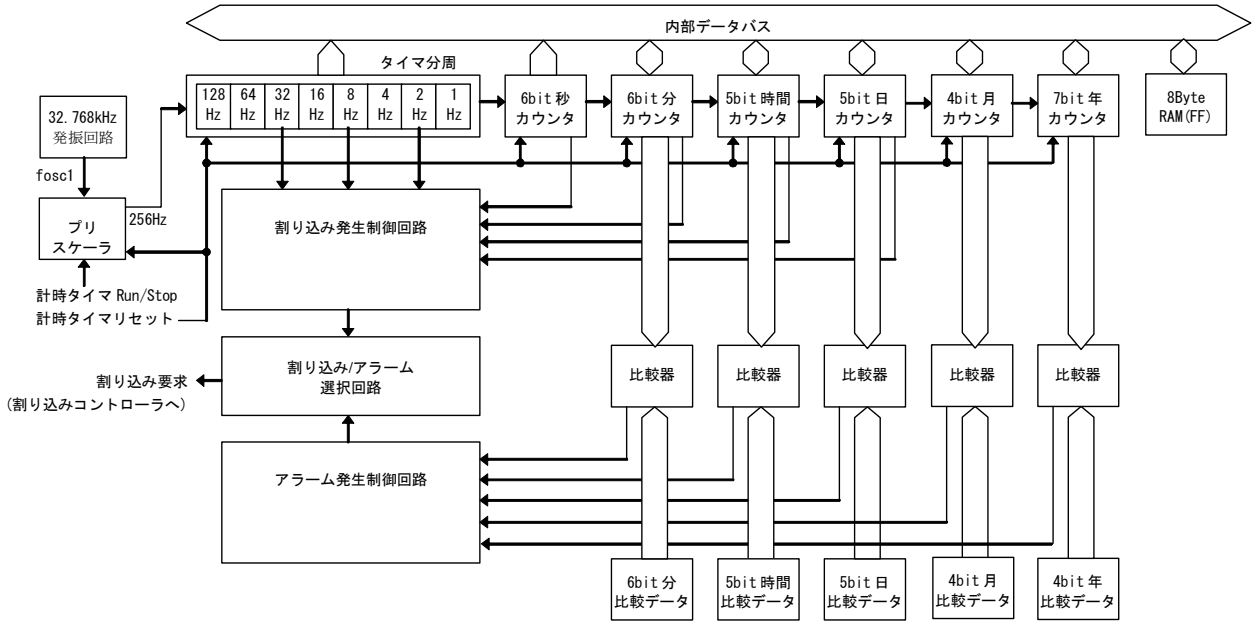


図 24.1　ブロックダイアグラム

24.3 外部端子

RTC に関連する外部端子は以下の通りです。

端子名	入出力	バス幅	端子機能	マルチプレクス端子／備考
BUP#	入力	1	バックアップモード切替	
SYS_OSCI	入力	1	水晶振動子用接続端子	
SYS_OSCO	出力	1	水晶振動子用接続端子	

24.4 レジスタ

24.4.1 レジスター覧

これらのレジスタのベースアドレスは、0x FFFF_8000 です。

表 24.1 レジスター覧（ベースアドレス：0x FFFF_8000）

Address Offset	Register Name	Default Value	R/W	Data Access Size
0x00	RTC ラン／ストップ制御レジスタ	xxx- --xxb	R/W	8 (/16/32)
0x04	RTC 割り込みレジスタ	0x xxxx	R/W	16 (/32)
0x08	RTC タイマ分周レジスタ	xxxx xxxxb	R/(W)	8 (/16/32)
0x0C	RTC 秒カウンタレジスタ	--xx xxxxb	R/W	8 (/16/32)
0x10	RTC 分カウンタレジスタ	--xx xxxxb	R/W	8 (/16/32)
0x14	RTC 時カウンタレジスタ	---x xxxxb	R/W	8 (/16/32)
0x18	RTC 日カウンタレジスタ	---x xxxxb	R/W	8 (/16/32)
0x1C	RTC 月カウンタレジスタ	---- xxxxb	R/W	8 (/16/32)
0x20	RTC 年カウンタレジスタ	-xxx xxxxb	R/W	8 (/16/32)
0x24	RTC アラーム分コンペアレジスタ	--xx xxxxb	R/W	8 (/16/32)
0x28	RTC アラーム時コンペアレジスタ	---x xxxxb	R/W	8 (/16/32)
0x2C	RTC アラーム日コンペアレジスタ	---x xxxxb	R/W	8 (/16/32)
0x30	RTC アラーム月コンペアレジスタ	---- xxxxb	R/W	8 (/16/32)
0x34	RTC アラーム年コンペアレジスタ	-xxx xxxxb	R/W	8 (/16/32)
0x38-0x3C	N/a			
0x40	RTC テストレジスタ	---x xxxxb	R/W	8 (/16/32)
0x44	RTC プリスケーラレジスタ	-xxx xxxxb	R/(W)	8 (/16/32)
0x48	RTC テストクロックレジスタ	xxxx xxxxb	WO	8 (/16/32)
0x4C-0x5C	N/a			
0x60	RTC RAM0	xxxx xxxxb	R/W	8 (/16/32)
0x64	RTC RAM1	xxxx xxxxb	R/W	8 (/16/32)
0x68	RTC RAM2	xxxx xxxxb	R/W	8 (/16/32)
0x6c	RTC RAM3	xxxx xxxxb	R/W	8 (/16/32)
0x70	RTC RAM4	xxxx xxxxb	R/W	8 (/16/32)
0x74	RTC RAM5	xxxx xxxxb	R/W	8 (/16/32)
0x78	RTC RAM6	xxxx xxxxb	R/W	8 (/16/32)
0x7C	RTC RAM7	xxxx xxxxb	R/W	8 (/16/32)

24.4.2 レジスタ詳細説明

すべての予約ビットは、特に指定のないかぎり“0”にしてください。ソフトウェアがこれらの予約ビットを読み出しても、その値は信頼すべきではありません。

RTC ラン／ストップ制御レジスタ (8 bit)

RTC[0x00] 初期値 = xxx- --xx b Read/Write

BUSY	BUSYWIDTH[1:0]		Reserved			TCADJ	TCRUN
x	xx		-			-	0
RO	R/W					WO	R/W
7	6	5	4	3	2	1	Bit0

x：不定ビット

Bit 7 : **BUSY ビジー（リードオンリ）**

このビットが“0”のとき、ソフトウェアは RTC レジスタにリード／ライト可能です。このビットが“1”のとき、RTC 内部のアップデートサイクルが動作中のため、ソフトウェアが仮にレジスタ・アクセスを行った場合、データの保証はできません。このアップデートサイクルは 1 秒に一回発生し、RTC ラン／ストップ制御レジスタ Bits [6:5] (BUSYWIDTH) で設定された期間の間、“1”となります。したがってこのビットが“1”の間は設定期間待ってから再度アクセスする必要があります。

Bits [6:5] : **BUSYWIDTH ビジー間隔**

ビジー期間の間隔を設定します。

00：約 244µs
01：約 122µs
10：約 61µs
11：予約

このビットはシステムリセットによって初期化されません。
正常動作を保証するため、RTC 停止中にリセット動作を行ってください。

Bits [4:2] : **Reserved 予約**

Bit 1 : **TCADJ RTC アジャスト（ライトオンリ）**

0： ノーマルモード
このビットを読み出すと、常に“0”になります。
1： アジャスト RTC カウンタ
このビットに 1 を、RTC ラン／ストップ制御レジスタのビット 0 に“0”を同時に書くことではプリスケーラカウンタと 128-1Hz カウンタのみリセットします。

RTC が動作中の場合、アジャスト動作は無効となります。

Bit 0 : **TCRUN**

読み出し
0： RTC 停止中
1： RTC 動作中

書き込み
0： RTC を停止
1： RTC を動作開始

32KHz クロックとの同期を取るために、約 30～61µs 遅延した後に動作／停止します。

RTC 割り込みレジスタ (16 bit)

RTC[0x04]　　初期値 = 0xXXXX　　Read/Write

Reserved					TCISE[2:0]		
-	-	-	-	-	x	x	x
					R/W	R/W	R/W
15	14	13	12	11	10	9	8
Reserved	TCASE[4:0]					TCIF	TCAF
-	x	x	x	x	x	x	x
	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [15:11]：　**Reserved　予約**

Bits [10:8]：　**TCISE[2:0]　割り込みソースイネーブルビットの選択ビット**

000：“32Hz”からのキャリー（1/32 秒に 1 回）
001：“8Hz”からのキャリー（1/8 秒に 1 回）
010：“2Hz”からのキャリー（1/2 秒に 1 回）
011：“1Hz”からのキャリー（1 秒に 1 回）
100：“1 分”からのキャリー（1 分に 1 回）
101：“1 時間”からのキャリー（1 時間に 1 回）
110：“日”からのキャリー（1 日に 1 回）
111：割り込みソースなし（初期値）

キャリーの変化を割り込みソースとしているため、割り込みが発生するのは指定された間隔に対して 1 回です。

Bits [6:2]：　**TCASE[4:0]　RTC アラームソース選択ビット**

00000：アラームなし（初期値）
xxxx1：分アラームイネーブル
xxx1x：時アラームイネーブル
xx1xx：日アラームイネーブル
x1xxx：月アラームイネーブル
1xxxx：年アラームイネーブル

“1”に設定されたアラームソースのカウンタがアラームコンペアレジスタと一致したときに割り込みが発生します。複数のソースをイネーブルにした場合は、そのいずれもがアラームコンペアレジスタと一致しているときに割り込みが発生します。
この割り込みはアラームコンペアレジスタと一致する間は割り込み要求が発生しつづけます。これは TCAF ビットに“1”を書き込み割り込み要求をクリアしたとしても同じです。したがって同じアラーム設定で次の割り込みを待ちたいときは、指定したアラームソースの中で一番小さい時間単位のカウンタが変化するまでは、すべてのソースをディゼーブルにする必要があります。たとえば時間と日のアラームをイネーブルにした場合は、割り込みが発生してから 1 時間の間は同じアラームソース設定を選択しないようにします。

Bit 1：　**TCIF　RTC タイマからの割り込み要求フラグ**

読み出し
0： ペンディング中の割り込みなし
1： RTC タイマからの未処理の割り込み要求あり
書き込み
0： N/A
1： RTC タイマからの割り込み要求をクリア

Bit 0：　**TCAF　アラームからの割り込み要求フラグ**

読み出し
0： ペンディング中の割り込みなし
1： アラームからの未処理の割り込み要求あり
書き込み
0： N/A

1： アラームからの割り込み要求をクリア

RTC タイマ分周レジスタ (8 bit)							
RTC[0x008] 初期値 = xxxx xxxx b							Read/(Write)
TCD[7:0]							
x	x	x	x	x	x	x	x
R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)
7	6	5	4	3	2	1	0

x：不定ビット

Bit 7： **TCD7: 1Hz インジケータ** 1: High, 0: Low
Bit 6： **TCD6: 2Hz インジケータ** 1: High, 0: Low
Bit 5： **TCD5: 4Hz インジケータ** 1: High, 0: Low
Bit 4： **TCD4: 8Hz インジケータ** 1: High, 0: Low
Bit 3： **TCD3: 16Hz インジケータ** 1: High, 0: Low
Bit 2： **TCD2: 32Hz インジケータ** 1: High, 0: Low
Bit 1： **TCD1: 64Hz インジケータ** 1: High, 0: Low
Bit 0： **TCD0: 128Hz インジケータ** 1: High, 0: Low

このレジスタは RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 1 (TCADJ)を“1”に書くことで“0”にリセットされます。ソフトウェアはこのレジスタをアップカウンタとしてみなすことができます。このレジスタには 32KHz からシステムクロックに同期させる回路を内蔵していないため、複数回リードを行い同じ値になったときの値を正しい値とする必要があります。
RTC テストレジスタ（RTC[0x40]）で分周カウンタ書き込み許可モードになっているときにのみ書き込み可能になります。ハードウェアテスト用の機能であるため分周カウンタ書き込み許可モードによるこのレジスタへの書き込み動作は保証外となります。

RTC 秒カウンタレジスタ (8 bit)							
RTC[0x0C] 初期値 = --xx xxxx b							Read/Write
Reserved		TCMD[5:0]					
-	-	x	x	x	x	x	x
		R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:6]： **Reserved 予約**

Bits [5:0]： **TCMD[5:0]**
これらの 6 ビットは 0-59 秒をバイナリデータとして示します。
TCMD5=MSB, TCMD0=LSB.
59 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC 分カウンタレジスタ (8 bit)							
RTC[0x10] 初期値 = --xx xxxx							Read/Write
Reserved		TCHD[5:0]					
-	-	x	x	x	x	x	x
		R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:6]： **Reserved 予約**

Bits [5:0] : **TCHD[5:0]**
これらの 6 ビットは 0-59 分をバイナリデータとして示します。
TCHD5=MSB, TCHD0=LSB.
59 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC 時間カウンタレジスタ (8 bit)
RTC[0x14] 初期値 = ---x xxxx b Read/Write

Reserved			TCDD[4:0]				
-	-	-	x	x	x	x	x
			R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:5] : **Reserved 予約**

Bits [4:0] : **TCDD[4:0]**
これらの 5 ビットは 0-23 時間をバイナリデータとして示します。
TCDD4=MSB, TCDD0=LSB.
23 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC 日カウンタレジスタ (8 bit)
RTC[0x18] 初期値 = ---x xxxx b Read/Write

Reserved			TCND [4:0]				
-	-	-	x	x	x	x	x
			R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:5] : **Reserved 予約**

Bits [4:0] : **TCND[4:0]**
この 5 ビットの日カウンタは、1−31 日をバイナリデータで示します。
TCND4 = MSB, TCND0 = LSB.
0 及び 31 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC 月カウンタレジスタ (8 bit)
RTC[0x1C] 初期値 = ---- xxxx b Read/Write

Reserved				TCDD[3:0]			
-	-	-	-	x	x	x	x
				R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:4] : **Reserved 予約**

Bits [3:0] : **TCTD[3:0]**
これらの 4 ビットは 1-12 月をバイナリデータとして示します。
TCTD3=MSB, TCTD0=LSB.
0 及び 12 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC 年カウンタレジスタ (8 bit)

RTC[0x20] 初期値 = -xxx xxxx b Read/Write

Reserved	TCYD[6:0]						
-	x	x	x	x	x	x	x
	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bit 7 : **Reserved 予約**

Bits [6:0] : **TCYD[6:0]**
これらの 7 ビットは 1-99 年をバイナリデータとして示します。
TCYD6=MSB, TCTD0=LSB.
0 及び 99 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC アラーム分コンペアレジスタ (8 bit)

RTC[0x24] 初期値 = --xx xxxx b Read/Write

Reserved		TCCH[5:0]					
-	-	x	x	x	x	x	x
		R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:6] : **Reserved 予約**

Bits [5:0] : **TCCH[5:0]**
これらの 6 ビットは 0-59 分をバイナリデータとして示します。
TCCH5=MSB, TCCH0=LSB.
59 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC アラーム時間コンペアレジスタ (8 bit)

RTC[0x28] 初期値 = ---x xxxx b Read/Write

Reserved			TCCD[4:0]				
-	-	-	x	x	x	x	x
			R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:5] : **Reserved 予約**

Bits [4:0] : **TCCD[4:0]**
これらの 5 ビットは 0-23 時間をバイナリデータとして示します。
TCCD4=MSB, TCCD0=LSB.
23 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC アラーム日コンペアレジスタ (8 bit)

RTC[0x2C]　初期値 = ---- ---x xxxx xxxx b　Read/Write

Reserved			TCCN[4:0]				
-	-	-	X	x	x	x	x
			R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [4:0]：　**TCCN[4:0]**

これらの 4 ビットは 0-31 日をバイナリデータとして示します。
TCCN4=MSB, TCCN0 LSB.
0 及び 31 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC アラーム月コンペアレジスタ (8 bit)

RTC[0x30]　初期値 = ---- xxxx b　Read/Write

Reserved				TCCT[3:0]			
-	-	-	-	x	x	x	x
				R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット (b)

Bits [7:4]：　**Reserved　予約**

Bits [3:0]：　**TCCT[3:0]**

これらの 4 ビットは 0-15 月をバイナリデータとして示します。
TCCT3=MSB, TCCT0=LSB.
０及び 12 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC アラーム年コンペアレジスタ (8 bit)

RTC[0x34]　初期値 = -xxx xxxx b　Read/Write

Reserved	TCCY[6:0]						
-	x	x	x	x	x	x	x
	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bit　7：　**Reserved　予約**

Bits [3:0]：　**TCCY[6:0]**

これらの 7 ビットは 0-127 年をバイナリデータとして示します。
TCCY6=MSB, TCCY0=LSB.
０及び 99 より大きい値を書き込んだときは、その値がそのまま反映されます。

RTC テストレジスタ (8 bit)

RTC[0x40] 初期値 = ---x xxxx b Read/Write

Reserved			RTST4	RTST3	RTST2	RTST1	RTST0
-			x	x	x	x	x
-			R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:5] : **Reserved 予約**

Bits [4:1] : **RTST[4:1] テストモード設定**

0000：分周カウンタ書き込み許可モード
このモード時に分周カウンタは書き込み可能になります。

xx10：テストクロックモード
このモード時は 32KHz クロック入力の代わりに RTC テストクロックレジスタへの書き込みによって発生するパルスを使用します。

x1xx：秒、分、時、日、年、月カウンタキャリーバイパスモード
秒、分、時、日、月、年カウンタのキャリーがクロックになります。

1xxx：分周カウンタキャリーバイパスモード
分周カウンタのキャリーがクロックになります。

Bit 0 **RTST0 テストモード許可**

前回“1”を書き、次に“0”を書いたときにテストモードになります。

RTC プリスケーラレジスタ (8 bit)

RTC[0x44] 初期値 = -xxx xxxx b Read/(Write)

Reserved	TCP [6:0]						
-	x	x	x	x	x	x	x
	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)
7	6	5	4	3	2	1	0

x：不定ビット

Bits 7 : **Reserved 予約**

Bits [6:0] : **TCP[6:0]**

これらの 7 ビットはプリスケーラ値を示します。
TCP8 = MSB, TCP0 = LSB.
このレジスタはシステムリセットによって初期化されません。
RTC テストレジスタで分周カウンタ書き込み許可モードになっているときにのみ書き込み可能になります。ハードウェアテスト用の機能であるため分周カウンタ書き込み許可モードによるこのレジスタへの書き込み動作は保証外となります。

RTC テストクロックレジスタ (8 bit)

RTC[0x48] 初期値 = -xxx xxxx b Write Only

TSTCLK[7:0]							
x	x	x	x	x	x	x	x
WO	WO	WO	WO	WO	WO	WO	WO
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:0] : **TSTCLK[7:0] テストクロック**

RTC テストレジスタでテストクロックモードに設定して、このレジスタに任意の値を書くとテストクロックにパルスが 1 回発生します。ハードウェアテスト用の機能であるため、このレジスタの使用は動作保証外となります。

RTC RAM0 - 7 レジスタ (8 bit)							
RTC[0x60 - 0x7C]　**初期値** = xxxx xxxx b							Read/Write
TCRAM0[7:0]							
x	x	x	x	x	x	x	X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

x：不定ビット

Bits [7:0] : **TCRAM0 - 7[7:0]**
これらの 8 ビットは RAM として使用可能です。
TCRAM0－7[7]=MSB, TCRAM0－7 [0]=LSB.
またこのレジスタはシステムリセット/ADJ ビットによって初期化されません。

24.5 RTC のレジスタ設定方法

テストレジスタ、テストクロックレジスタ及びテストモードにかかわる動作は、動作保証対象外となります。また、すべてのレジスタはシステムリセットによる初期化は行われません。

24.5.1 パワーオン後の初期設定

パワーオン直後、RTC 内の全てのレジスタは不定状態です。以下の手順での初期化が必要です。

A) RTC テストレジスタ(RTC[0x40])の全てのビットに 2 度“0x00”を書きます。
B) RTC が停止状態で各種設定を行いますので、RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 に“0b0”を書き停止状態にします。
C) RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 が“0b0”であることを確認します。（このビットが“0b0”のとき RTC は停止状態、“0b1”のとき動作状態であることを示します。RTC は各種レジスタ設定を 32KHz 系の回路と同期させるために 30~61µs の遅延が発生しますので、確実に停止していることを確認する必要があります。RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 は 32KHz 系回路側のランストップ制御信号を再度同期化していますので、同期による遅延が反映された結果になっています。）
D) RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 1-0 に“0b10”を書き、プリスケーラおよび分周タイマをリセットします。
（パワーオン時は全てのレジスタが不定状態ですので、この動作によりプリスケーラおよび分周タイマが“0”に初期化されます。ビット 1 に“0b1”を書くだけでなく、ビット 0 に“0b0”を書く必要があることに注意してください。）
E) 日時の設定をする前に割り込みをディセーブルにし、フラグをクリアする必要があります。RTC 割り込みレジスタ(RTC[0x04])に“0x0703”を書き込み、割り込みコントローラに影響を与えないようにします。
F) RTC 割り込みレジスタ(RTC[0x04])設定後、日時の設定を行います。秒、分、時、日、月、年カウンタを設定してください。
（存在しない時刻が書き込まれた状態で下位より桁上げがあると、カウンタが誤動作しますので、正しい値を入力して下さい。）
G) アラームを設定する場合は、分、時、日、月、年アラームコンペアレジスタを設定します。また、割り込みを設定する場合は RTC 割り込みレジスタ(RTC[0x04])の周期割り込みとアラーム割り込みをそれぞれイネーブルにします。
H) ワークとして実装しています 8byteRAM も初期値不定のため割込みフラグクリア後初期化することを推奨します。
I) 最後に RTC ラン／ストップ制御レジスタ（RTC[0x00]）BUSY の幅を設定します。BUSY の幅はソフトウェア処理が確実に終了できる時間より大きい値の設定を推奨します。たとえばソフトウェアが他の処理によって設定処理が停止することも含めて 100µs 以内に設定処理が終了する場合は、BUSYWIDTH を“01”(122µs) に設定します。RTC が停止状態ですので秒カウンタ

から BUSY の幅設定までは順不同となります。どれから設定を行っても問題ありません。

J) 以上の設定が終了した後、RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 に“0b1”を書くことで RTC をスタートさせます。30~61µs 後に RTC は動作を開始します。

24.5.2 動作停止、動作再開始

RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 に“0b0”を書き、RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 が“0b0”になるまで待ちます（およそ 30~61µs）。ビット 0 が“0b0”になることで、動作停止状態となります。各カウンタの値は停止中でも保持されています。動作停止状態から動作を再開始する場合、RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 に“0b1”を書き込むことで動作状態となります。この時保持している値からカウントを再開します。割り込み処理後、動作を再開する場合は、割り込みフラグをクリアした後で、RTC ラン／ストップ制御レジスタのビット 1-0 に“10”を書き、プリスケーラ及び分周タイマをリセット後、RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 0 に“0b1”を書き込み動作状態にします。

24.5.3 動作中の動作停止以外の再設定

RTC ラン／ストップ制御レジスタ（RTC[0x000]）のビット 0 以外、動作中にアクセスを行うと動作保証範囲外となります。特にプリスケーラ及び分周タイマのリセットを動作中に行うと誤動作を行う可能性がありますので注意願います。

RTC 割り込みレジスタおよび分、時、日、月、年アラームコンペアレジスタは BUSY ビットが“0”のときに設定変更可能ですが、変更前に RTC の割り込みを割り込みコントローラでディゼーブルにした後で設定してください。変更後は割り込みコントローラ及び RTC の割り込みフラグをクリアした上で RTC の割り込みを割り込みコントローラでイネーブルにする必要があります。これは変更中に発生する可能性のある不要な割り込みを禁止するためです。

秒、分、時、日、月、年カウンタレジスタは BUSY ビットが“0”のときに変更可能です。秒、分、時、日、月、年カウンタレジスタへの再設定は一瞬で終わりますので、その期間中は他の周辺回路の割り込みをすべてディゼーブルにして他の処理が割り込まないようにすることで 1 回の BUSY の確認だけで再設定を済ますことができます。

24.5.4 動作中のシステムリセット後の再設定

RTC が動作状態のときにシステムリセットが発生した場合、RTC に影響はなく動作し続けます。（RTC 内の全てのレジスタはシステムリセットの影響を受けません。）

RTC は RTCVDD（1.8V：Typ.）を落とさない限り動作し続けます。

24.5.5 プログラミング上の注意事項

- パワーオン直後、RTC内の全てのレジスタは不定状態です。「24.5.1　パワーオン後の初期設定」にしたがって初期設定を行ってください。
- プリスケーラおよび分周タイマをリセットする場合は、RTC を停止させた後で、RTC ラン／ストップ制御レジスタ（RTC[0x00]）のビット 1-0 に“0b10”を書いてください。ビット 1 に“1”を書くだけでなく、ビット 0 に“0”を書く必要があります。RTC 動作中にリセット動作を行った場合、無効となります。
- 不要な割り込み発生を防止するため、割り込み要因およびアラーム要因の設定変更は、RTC の割り込みを割り込みコントローラ側でディゼーブルにした状態で行ってください。また割り込みをイネーブルに戻す前に割り込み要因フラグおよびアラーム要因フラグをクリアしてください。

25. ウォッチドッグタイマ (WDT)

25.1 概要

ウォッチドッグタイマ (WDT) はシステムの暴走を監視するためのユニットで、ソフトウェアプリセットが可能な 16 ビットダウンカウンタで構成されています。初期設定値からカウントダウンし、カウンタが“0”になると、ウォッチドッグタイマ動作選択ビット (WDT[0x08] Bit 4) の設定に従い割り込み要求またはリセット要求を発生します。ソフトウェアは周期的にプリセットデータをカウンタにロードし、カウンタが“0”にならないようにします。これで割り込みまたはリセット要求が発生した場合、プログラムが正常に実行できなかったと判断できます。

“0”になった後のカウントで、カウンタは 0xFFFF になります。カウンタの値はソフトウェアでいつでも読み出し可能です。なお、カウンタはシステムリセットあるいはウォッチドッグタイマイネーブルビット (WDT[0x08]Bit 5) への“0”書き込みによって 0xFFFF に設定されます。カウントはウォッチドッグタイマイネーブルビットを“0”に設定することで停止させることもできます。ウォッチドッグタイマは HALT モード時も動作します。

ウォッチドッグタイマのソースクロックは APB クロックです。ウォッチドッグタイマはプリスケーラを内蔵しているため、ソースクロックの分周比を制御してカウントクロックの周波数を設定することができます。

25.2 ブロック図

ウォッチドッグタイマのブロック図を以下に示します。

ウォッチドッグタイマは、レジスタブロック（read, load, control レジスタ/APB バス I/F）と 16 ビットダウンカウンタ、11 ビットダウンカウンタ内蔵のプリスケーラブロックから構成されています。

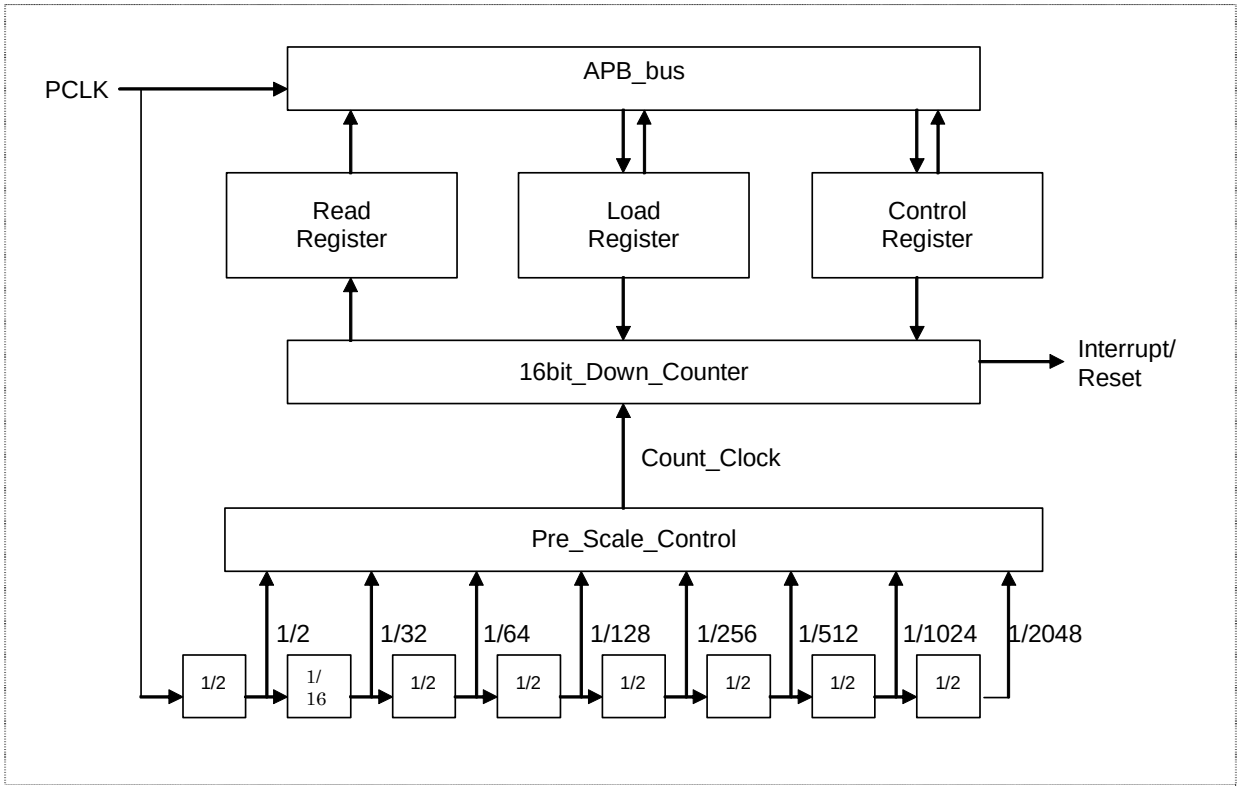


図 25.1 WDT ブロック図

25.3 外部端子

ウォッチドッグタイマに関連する外部端子はありません。

25.4 レジスタ

25.4.1 レジスター覧

これらのレジスタのベースアドレスは、0xFFFF_C000 です。

表 25.1 レジスター覧（ベースアドレス：0xFFFF_C000）

アドレス オフセット	レジスタ名称	初期値	R/W	データ アクセスサイズ
0x00	ウォッチドッグタイマロードレジスタ	0x0000_FFFF	R/W	16 (/32)
0x04	ウォッチドッグタイマカウントレジスタ	0x0000_FFFF	RO	16 (/32)
0x08	ウォッチドッグタイマ制御レジスタ	0x0000_0000	R/W	16 (/32)

25.4.2 レジスタ詳細説明

ウォッチドッグタイマの制御レジスタのベースアドレスは 0xFFFF_C000 です。特に指定のない場合、予約されていないレジスタビットのデフォルト値はすべて“0”です。

ウォッチドッグタイマロードレジスタ

WDT[0x00] 初期値 = 0x0000_FFFF Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
タイマロード値															

Bits [15:0] : **タイマロード値ビット[15:0]**
ここにデータを書き込むと、その値がカウンタにロードされます。

ウォッチドッグタイマカウントレジスタ

WDT[0x04] 初期値 = 0x0000_FFFF Read Only

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
n/a															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
カレントタイマカウント値															

Bits [15:0] : **カレントタイマカウント値ビット[15:0]**
現在のカウンタの値が読み出せます。

ウォッチドッグタイマ制御レジスタ

WDT[0x08]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
								n/a							
予約 (A5h に固定)								n/a	WDT ステータス (RO)	WDT イネーブル	WDT 動作選択	n/a	プリスケーラ分周比		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [15:8] :　**予約**

書き込み時は必ず 0xA5 にしてください。

Bit 6 :　**ウォッチドッグタイマステータス（リードオンリ）**

カウンタが“0”になったかどうかを示します。

0 : カウンタ ＝ “0”以外

1 : カウンタ ＝ “0”

このビットはカウンタが“0”になると“1”にセットされ、その後その状態を保持します。クリアするには、ウォッチドッグタイマイネーブル（本レジスタのビット 5）を一旦“0”に設定し、ウォッチドッグタイマをリセットする必要があります。

Bit 5 :　**ウォッチドッグタイマイネーブル**

ウォッチドッグタイマのカウントを許可/禁止します。

0 (r/w) : タイマディセーブル。

“0”を書き込むと、カウント中であればカウントを停止します。同時にカウンタが 0xFFFF にリセットされ、ウォッチドッグタイマステータス（本レジスタのビット 6）も“0”にリセットされます。

1 (r/w) : タイマイネーブル。

“1”を書き込んだ時点でウォッチドッグタイマはダウンカウントを開始します。

Bit 4 :　**ウォッチドッグタイマ動作**

カウンタが“0”になったときの動作を指定します。

0 (r/w) : リセット要求信号を出力。

カウンタが“0”になると、ウォッチドッグタイマはリセット要求信号（WRST=HIGH）をシステムコントローラに出力します。リセット要求信号はカウンタが“0”以外になると LOW に戻ります。

1 (r/w) : 割り込み要求信号を出力。

カウンタが“0”になると、ウォッチドッグタイマは割り込み要求（信号 WINT = HIGH）を出力します。割り込み要求信号はウォッチドッグタイマイネーブル（本レジスタのビット 5）をディセーブルにすると LOW に戻ります。

Bits [2:0] :　**プリスケーラ分周比ビット[2:0]**

カウントクロックを生成するためのプリスケーラの分周比を選択します。プリスケーラのソースクロックは APB クロックです。

Bits [2:0]	プリスケーラ分周比 n*
000	2
001	32
010	64
011	128
100	256
101	512
110	1024
111	2048

＊：プリスケーラ分周比　n ＝ 2　　・・・Bits[2:0] = 0 のとき

＝ $2^{(4 + Bits[2:0])}$　・・・Bits[2:0] ≠ 0 のとき

このとき、カウントクロック周波数（$f_{カウントクロック}$）は APB システムクロック (PCLK) 周波数（f_{PCLK}）とプリスケーラ分周比（n）により以下の式で表されます。

$$f_{カウントクロック} = f_{PCLK} / n$$

26. GPIO

26.1 概要

この機能ブロックは GPIO の機能と、GPIO 端子にマルチプレクスした端子機能を選択するためのレジスタで構成されています。
本 GPIO には以下のような特長があります。

- 10 の 8Bit GPIO ポートをサポート
  GPIOA/GPIOB/GPIOC/GPIOD/GPIOE/GPIOF/GPIOG/GPIOH/ GPIOJ/GPIOK
- 1 つの 2Bit GPIO ポートをサポート
  GPIOI
- 全ての GPIO の端子毎に方向（入力／出力）が変更可能
- 一部の端子は他の機能とマルチプレクス仕様
- マルチ機能の I/O 端子は端子機能レジスタで機能を選択
- GPIOA ＆ GPIOB については割り込み入力の機能もサポート
- GPIOA および GPIOB からの割り込みはエッジまたはレベルおよび High/Low の極性の選択が可能

26.2 外部端子

GPIO に関連する外部端子は以下の通りです。

ポート	端子名	入出力	端子機能	マルチプレクス端子／備考		
				共用	機能 1	機能 2
A	GPIOA0	入出力	汎用 I/O ポート A0 入出力		TXD1	
	GPIOA1	入出力	汎用 I/O ポート A1 入出力		RXD1	
	GPIOA2	入出力	汎用 I/O ポート A2 入出力		RTS1	I2S1_WS
	GPIOA3	入出力	汎用 I/O ポート A3 入出力		CTS1	I2S1_SCK
	GPIOA4	入出力	汎用 I/O ポート A4 入出力		TXD2	
	GPIOA5	入出力	汎用 I/O ポート A5 入出力		RXD2	
	GPIOA6	入出力	汎用 I/O ポート A6 入出力		RTS2	SCL
	GPIOA7	入出力	汎用 I/O ポート A7 入出力		CTS2	SDA
B	GPIOB0	入出力	汎用 I/O ポート B0 入出力	INT0	I2S0_WS	
	GPIOB1	入出力	汎用 I/O ポート B1 入出力	INT1	I2S0_SCK	
	GPIOB2	入出力	汎用 I/O ポート B2 入出力	INT2	I2S0_SD	
	GPIOB3	入出力	汎用 I/O ポート B3 入出力	INT3	I2S1_SD	
	GPIOB4	入出力	汎用 I/O ポート B4 入出力	INT4	TimerA0out	
	GPIOB5	入出力	汎用 I/O ポート B5 入出力	INT5	TimerA1out	
	GPIOB6	入出力	汎用 I/O ポート B6 入出力	INT6	TimerA2out	
	GPIOB7	入出力	汎用 I/O ポート B7 入出力	INT7	TimerBIn	
C	GPIOC0	入出力	汎用 I/O ポート C0 入出力		TimerB0IO	
	GPIOC1	入出力	汎用 I/O ポート C1 入出力		TimerB1IO	
	GPIOC2	入出力	汎用 I/O ポート C2 入出力		TimerB2IO	
	GPIOC3	入出力	汎用 I/O ポート C3 入出力		TimerB3IO	UART3_CLK
	GPIOC4	入出力	汎用 I/O ポート C4 入出力		SPI_SS	TXD3
	GPIOC5	入出力	汎用 I/O ポート C5 入出力		SPI_SCLK	RXD3
	GPIOC6	入出力	汎用 I/O ポート C6 入出力		SPI_MISO	RTS3
	GPIOC7	入出力	汎用 I/O ポート C7 入出力		SPI_MOSI	CTS3
D	GPIOD0	入出力	汎用 I/O ポート D0 入出力		MA20	
	GPIOD1	入出力	汎用 I/O ポート D1 入出力		MA21	
	GPIOD2	入出力	汎用 I/O ポート D2 入出力		MA22	
	GPIOD3	入出力	汎用 I/O ポート D3 入出力		MA23	
	GPIOD4	入出力	汎用 I/O ポート D4 入出力		MCS2#	
	GPIOD5	入出力	汎用 I/O ポート D5 入出力		MCS3#	
	GPIOD6	入出力	汎用 I/O ポート D6 入出力		SDDQM2	
	GPIOD7	入出力	汎用 I/O ポート D7 入出力		SDDQM3	
E	GPIOE0	入出力	汎用 I/O ポート E0 入出力		CM1DATA0	
	GPIOE1	入出力	汎用 I/O ポート E1 入出力		CM1DATA1	
	GPIOE2	入出力	汎用 I/O ポート E2 入出力		CM1DATA2	
	GPIOE3	入出力	汎用 I/O ポート E3 入出力		CM1DATA3	
	GPIOE4	入出力	汎用 I/O ポート E4 入出力		CM1DATA4	
	GPIOE5	入出力	汎用 I/O ポート E5 入出力		CM1DATA5	
	GPIOE6	入出力	汎用 I/O ポート E6 入出力		CM1DATA6	
	GPIOE7	入出力	汎用 I/O ポート E7 入出力		CM1DATA7	
F	GPIOF0	入出力	汎用 I/O ポート F0 入出力		CM1VREF	
	GPIOF1	入出力	汎用 I/O ポート F1 入出力		CM1HREF	
	GPIOF2	入出力	汎用 I/O ポート F2 入出力		CM1CLKOUT	
	GPIOF3	入出力	汎用 I/O ポート F3 入出力		CM1CLKIN	
	GPIOF4	入出力	汎用 I/O ポート F4 入出力		CM2VREF	
	GPIOF5	入出力	汎用 I/O ポート F5 入出力		CM2HREF	
	GPIOF6	入出力	汎用 I/O ポート F6 入出力		CM2CLKOUT	
	GPIOF7	入出力	汎用 I/O ポート F7 入出力		CM2CLKIN	

ポート	端子名	入出力	端子機能	マルチプレクス端子／備考		
				共用	機能 1	機能 2
G	GPIOG0	入出力	汎用 I/O ポート G0 入出力		CM2DATA0	
	GPIOG1	入出力	汎用 I/O ポート G1 入出力		CM2DATA1	
	GPIOG2	入出力	汎用 I/O ポート G2 入出力		CM2DATA2	
	GPIOG3	入出力	汎用 I/O ポート G3 入出力		CM2DATA3	
	GPIOG4	入出力	汎用 I/O ポート G4 入出力		CM2DATA4	
	GPIOG5	入出力	汎用 I/O ポート G5 入出力		CM2DATA5	
	GPIOG6	入出力	汎用 I/O ポート G6 入出力		CM2DATA6	
	GPIOG7	入出力	汎用 I/O ポート G7 入出力		CM2DATA7	
H	GPIOH0	入出力	汎用 I/O ポート H0 入出力		CFCE2#	SDMDATA0
	GPIOH1	入出力	汎用 I/O ポート H1 入出力		CFCE1#	SDMDATA1
	GPIOH2	入出力	汎用 I/O ポート H2 入出力		CFIORD#	SDMDATA2
	GPIOH3	入出力	汎用 I/O ポート H3 入出力		CFIOWR#	SDMDATA3
	GPIOH4	入出力	汎用 I/O ポート H4 入出力		CFWAIT#	SDMCMD
	GPIOH5	入出力	汎用 I/O ポート H5 入出力		CFRST	SDMCLK
	GPIOH6	入出力	汎用 I/O ポート H6 入出力		CFIRQ	SDMCD#
	GPIOH7	入出力	汎用 I/O ポート H7 入出力		CFSTSCHG#	SDMWP
I	GPIOI0	入出力	汎用 I/O ポート I0 入出力		CFDEN#	SDMGPO
	GPIOI1	入出力	汎用 I/O ポート I1 入出力		CFDDIR	
	~~GPIOI2~~	~~入出力~~	~~汎用 I/O ポート I2 入出力~~			
	~~GPIOI3~~	~~入出力~~	~~汎用 I/O ポート I3 入出力~~			
	~~GPIOI4~~	~~入出力~~	~~汎用 I/O ポート I4 入出力~~			
	~~GPIOI5~~	~~入出力~~	~~汎用 I/O ポート I5 入出力~~			
	~~GPIOI6~~	~~入出力~~	~~汎用 I/O ポート I6 入出力~~			
	~~GPIOI7~~	~~入出力~~	~~汎用 I/O ポート I7 入出力~~			
J	GPIOJ0	入出力	汎用 I/O ポート J0 入出力		SDD16	
	GPIOJ1	入出力	汎用 I/O ポート J1 入出力		SDD17	
	GPIOJ2	入出力	汎用 I/O ポート J2 入出力		SDD18	
	GPIOJ3	入出力	汎用 I/O ポート J3 入出力		SDD19	
	GPIOJ4	入出力	汎用 I/O ポート J4 入出力		SDD20	
	GPIOJ5	入出力	汎用 I/O ポート J5 入出力		SDD21	
	GPIOJ6	入出力	汎用 I/O ポート J6 入出力		SDD22	
	GPIOJ7	入出力	汎用 I/O ポート J7 入出力		SDD23	
K	GPIOK0	入出力	汎用 I/O ポート K0 入出力		SDD24	
	GPIOK1	入出力	汎用 I/O ポート K1 入出力		SDD25	
	GPIOK2	入出力	汎用 I/O ポート K2 入出力		SDD26	
	GPIOK3	入出力	汎用 I/O ポート K3 入出力		SDD27	
	GPIOK4	入出力	汎用 I/O ポート K4 入出力		SDD28	
	GPIOK5	入出力	汎用 I/O ポート K5 入出力		SDD29	
	GPIOK6	入出力	汎用 I/O ポート K6 入出力		SDD30	
	GPIOK7	入出力	汎用 I/O ポート K7 入出力		SDD31	

26.3 レジスタ

26.3.1 レジスター覧

これらのレジスタのベースアドレスは、0xFFFF_1000 です。

表 26.1　レジスター覧（ベースアドレス：0xFFFF_1000）

Address Offset	Register Name	Abbreviation Name	Default Value	R/W	Data Access Size
0x00	GPIOA データレジスタ	GPIOA_DATA	0x0000	R/W	8 (/16/32) *1
0x04	GPIOA 端子機能レジスタ	GPIOA_FNC	0x0000	R/W	16 (/32) *2
0x08	GPIOB データレジスタ	GPIOB_DATA	0x0000	R/W	8 (/16/32) *1
0x0C	GPIOB 端子機能レジスタ	GPIOB_FNC	0x0000	R/W	16 (/32) *2
0x10	GPIOC データレジスタ	GPIOC_DATA	0x0000	R/W	8 (/16/32) *1
0x14	GPIOC 端子機能レジスタ	GPIOC_FNC	0x0000	R/W	16 (/32) *2
0x18	GPIOD データレジスタ	GPIOD_DATA	0x0000	R/W	8 (/16/32) *1
0x1C	GPIOD 端子機能レジスタ	GPIOD_FNC	0x0000	R/W	16 (/32) *2
0x20	GPIOE データレジスタ	GPIOE_DATA	0x0000	R/W	8 (/16/32) *1
0x24	GPIOE 端子機能レジスタ	GPIOE_FNC	0x0000	R/W	16 (/32) *2
0x28	GPIOF データレジスタ	GPIOF_DATA	0x0000	R/W	8 (/16/32) *1
0x2C	GPIOF 端子機能レジスタ	GPIOF_FNC	0x0000	R/W	16 (/32) *2
0x30	GPIOG データレジスタ	GPIOG_DATA	0x0000	R/W	8 (/16/32) *1
0x34	GPIOG 端子機能レジスタ	GPIOG_FNC	0x0000	R/W	16 (/32) *2
0x38	GPIOH データレジスタ	GPIOH_DATA	0x0000	R/W	8 (/16/32) *1
0x3C	GPIOH 端子機能レジスタ	GPIOH_FNC	0x0000	R/W	16 (/32) *2
0x40	GPIOI データレジスタ	GPIOI_DATA	0x0000	R/W	8 (/16/32) *1
0x44	GPIOI 端子機能レジスタ	GPIOI_FNC	0x0000	R/W	16 (/32) *2
0x48	GPIOJ データレジスタ	GPIOJ_DATA	0x0000	R/W	8 (/16/32) *1
0x4C	GPIOJ 端子機能レジスタ	GPIOJ_FNC	0x0000	R/W	16 (/32) *2
0x50	GPIOK データレジスタ	GPIOK_DATA	0x0000	R/W	8 (/16/32) *1
0x54	GPIOK 端子機能レジスタ	GPIOK_FNC	0x0000	R/W	16 (/32) *2
0x58-5C	N/a				
0x60	GPIOA&B IRQ タイプレジスタ	GPIOAB_ITYP	0x0000	R/W	16 (/32) *2
0x64	GPIOA&B IRQ 極性レジスタ	GPIOAB_IPOL	0x0000	R/W	16 (/32) *2
0x68	GPIOA&B IRQ イネーブルレジスタ	GPIOAB_IEN	0x0000	R/W	16 (/32) *2
0x6C	GPIOA&B IRQ ステータス＆クリアレジスタ	GPIOAB_ISTS	0x0000	R/W	16 (/32) *2

*1: 8、16 または 32 ビットデータがアクセス可能です。
*2: 16 または 32 ビットデータがアクセス可能です。

26.3.2 レジスタ詳細説明

26.3.2.1 GPIOA レジスタ

GPIOA データレジスタ　(GPIOA_DATA)

GPIO[0x00]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
GPIOADATA [7:0]							

GPIOA のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOA が出力設定時にはこのレジスタの内容が読み出され、GPIOA が入力設定時には端子の状態を読むことができます。

GPIOA 端子機能レジスタ　(GPIOA_FNC)

GPIO[0x04]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPA7MD [1:0]		GPA6MD [1:0]		GPA5MD [1:0]		GPA4MD [1:0]	
7	6	5	4	3	2	1	0
GPA3MD [1:0]		GPA2MD [1:0]		GPA1MD [1:0]		GPA0MD [1:0]	

GPIOA の端子機能を選択します。GPIOA の各ポートに対して 2Bit が端子機能選択に使われます。

表 26.2　PortA の端子選択機能

GPAxMD1	GPAxMD0	端子機能（x は PORT 中の Bit 位置を示す）
0	0	GPIOAx ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIOAx ポート出力
1	1	GPIO 以外の機能 2（GPIOA[7:6],[3:2]のみ設定可能）

26.3.2.2 GPIOB レジスタ

GPIOB データレジスタ　(GPIOB_DATA)							
GPIO[0x08]　　初期値 = 0x0000_0000							Read/Write
n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOBDATA [7:0]							
7	6	5	4	3	2	1	0

GPIOB のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOB が出力設定時にはこのレジスタの内容が読み出され、GPIOB が入力設定時には端子の状態を読むことができます。

GPIOB 端子機能レジスタ　(GPIOB_FNC)							
GPIO[0x0C]　　初期値 = 0x0000_0000							Read/Write
n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPB7MD [1:0]		GPB6MD [1:0]		GPB5MD [1:0]		GPB4MD [1:0]	
15	14	13	12	11	10	9	8
GPB3MD [1:0]		GPB2MD [1:0]		GPB1MD [1:0]		GPB0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOB の端子機能を選択します。GPIOB の各ポートに対して 2Bit が端子機能選択に使われます。

表 26.3　PortB の端子選択機能

GPBxMD1	GPBxMD0	端子機能（x は PORT 中の Bit 位置を示す）
0	0	GPIOBx ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIOBx ポート出力
1	1	予約

26.3.2.3 GPIOC レジスタ

GPIOC データレジスタ　(GPIOC_DATA)

GPIO[0x10]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOCDATA [7:0]							
7	6	5	4	3	2	1	0

GPIOC のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOC が出力設定時にはこのレジスタの内容が読み出され、GPIOC が入力設定時には端子の状態を読むことができます。

GPIOC 端子機能レジスタ　(GPIOC_FNC)

GPIO[0x14]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPC7MD [1:0]		GPC6MD [1:0]		GPC5MD [1:0]		GPC4MD [1:0]	
15	14	13	12	11	10	9	8
GPC3MD [1:0]		GPC2MD [1:0]		GPC1MD [1:0]		GPC0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOC の端子機能を選択します。GPIOC の各ポートに対して 2Bit が端子機能選択に使われます。

表 26.4　PortC の端子選択機能

GPCxMD1	GPCxMD0	端子機能（x は PORT 中の Bit 位置を示す）
0	0	GPIOCx ポート入力（デフォルト）
0	1	GPIO 以外の機能１
1	0	GPIOCx ポート出力
1	1	GPIO 以外の機能２（GPIOC[7:3]のみ設定可能）

26.3.2.4 GPIOD レジスタ

GPIOD データレジスタ　(GPIOD_DATA)
GPIO[0x18]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIODDATA [7:0]							
7	6	5	4	3	2	1	0

GPIOD のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOD が出力設定時にはこのレジスタの内容が読み出され、GPIOD が入力設定時には端子の状態を読むことができます。

GPIOD 端子機能レジスタ　(GPIOD_FNC)
GPIO[0x1C]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPD7MD [1:0]		GPD6MD [1:0]		GPD5MD [1:0]		GPD4MD [1:0]	
15	14	13	12	11	10	9	8
GPD3MD [1:0]		GPD2MD [1:0]		GPD1MD [1:0]		GPD0MD[1:0]	
7	6	5	4	3	2	1	0

GPIOD の端子機能を選択します。GPIOD の各ポートに対して 2Bit が端子機能選択に使われます。

表 26.5　PortD の端子選択機能

GPDxMD1	GPDxMD0	端子機能（x は PORT 中の Bit 位置を示す）
0	0	GPIODx ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIODx ポート出力
1	1	予約

26.3.2.5 GPIOE レジスタ

GPIOE データレジスタ　(GPIOE_DATA)							
GPIO[0x20]　　初期値 = 0x0000_0000							Read/Write
n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOEDATA [7:0]							
7	6	5	4	3	2	1	0

GPIOE のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOE が出力設定時にはこのレジスタの内容が読み出され、GPIOE が入力設定時には端子の状態を読むことができます。

GPIOE 端子機能レジスタ　(GPIOE_FNC)							
GPIO[0x24]　　初期値 = 0x0000_0000							Read/Write
n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPE7MD [1:0]		GPE6MD [1:0]		GPE5MD [1:0]		GPE4MD [1:0]	
15	14	13	12	11	10	9	8
GPE3MD [1:0]		GPE2MD [1:0]		GPE1MD [1:0]		GPE0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOE の端子機能を選択します。GPIOE の各ポートに対して 2Bit が端子機能選択に使われます。

表 26.6　PortE の端子選択機能

GPExMD1	GPExMD0	端子機能（x は PORT 中の Bit 位置を示す）
0	0	GPIOEx ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIOEx ポート出力
1	1	予約

26.3.2.6 GPIOF レジスタ

GPIOF データレジスタ　(GPIOF_DATA)

GPIO[0x28]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOFDATA [7:0]							
7	6	5	4	3	2	1	0

GPIOF のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOF が出力設定時にはこのレジスタの内容が読み出され、GPIOF が入力設定時には端子の状態を読むことができます。

GPIOF 端子機能レジスタ　(GPIOF_FNC)

GPIO[0x2C]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPF7MD [1:0]		GPF6MD [1:0]		GPF5MD [1:0]		GPF4MD [1:0]	
15	14	13	12	11	10	9	8
GPF3MD [1:0]		GPF2MD [1:0]		GPF1MD [1:0]		GPF0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOF の端子機能を選択します。GPIOF の各ポートに対して 2Bit が端子機能選択に使われます。

表 26.7　PortF の端子選択機能

GPFxMD1	GPFxMD0	端子機能（x は PORT 中の Bit 位置を示す）
0	0	GPIOFx ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIOFx ポート出力
1	1	予約

26.3.2.7 GPIOG レジスタ

GPIOG データレジスタ　(GPIOG_DATA)

GPIO[0x30]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOGDATA [7:0]							
7	6	5	4	3	2	1	0

GPIOG のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOG が出力設定時にはこのレジスタの内容が読み出され、GPIOG が入力設定時には端子の状態を読むことができます。

GPIOG 端子機能レジスタ　(GPIOG_FNC)

GPIO[0x34]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPG7MD [1:0]		GPG6MD [1:0]		GPG5MD [1:0]		GPG4MD [1:0]	
15	14	13	12	11	10	9	8
GPG3MD [1:0]		GPG2MD [1:0]		GPG1MD [1:0]		GPG0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOG の端子機能を選択します。GPIOG の各ポートに対して 2Bit が端子機能選択に使われます。

表 26.8　PortG の端子選択機能

GPGxMD1	GPGxMD0	端子機能（x は PORT 中の Bit 位置を示す）
0	0	GPIOGx ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIOGx ポート出力
1	1	予約

26.3.2.8 GPIOH レジスタ

GPIOH データレジスタ　(GPIOH_DATA)

GPIO[0x38]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOHDATA[7:0]							
7	6	5	4	3	2	1	0

GPIOH のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOH が出力設定時にはこのレジスタの内容が読み出され、GPIOH が入力設定時には端子の状態を読むことができます。

GPIOH 端子機能レジスタ　(GPIOH_FNC)

GPIO[0x3C]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPH7MD [1:0]		GPH6MD [1:0]		GPH5MD [1:0]		GPH4MD [1:0]	
15	14	13	12	11	10	9	8
GPH3MD [1:0]		GPH2MD [1:0]		GPH1MD [1:0]		GPH0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOH の端子機能を選択します。GPIOH のポートに対して 2Bit が端子機能選択に使われます。

表 26.9　PortH の端子選択機能

GPHxMD1	GPHxMD0	端子機能
0	0	GPIOHx ポート入力（デフォルト）
0	1	GPIO 以外の機能１
1	0	GPIOHx ポート出力
1	1	GPIO 以外の機能２

26.3.2.9 GPIOI レジスタ

GPIOI データレジスタ　(GPIOI_DATA)

GPIO[0x40]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
						GPIOIDATA[1:0]	

GPIOI のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOI が出力設定時にはこのレジスタの内容が読み出され、GPIOI が入力設定時には端子の状態を読むことができます。

GPIOI 端子機能レジスタ　(GPIOI_FNC)

GPIO[0x44]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
n/a							
7	6	5	4	3	2	1	0
				GPI1MD [1:0]		GPI0MD [1:0]	

GPIOI の端子機能を選択します。GPIOI のポートに対して 2Bit が端子機能選択に使われます。

表 26.10　PortI の端子選択機能

GPIxMD1	GPIxMD0	端子機能
0	0	GPIOIx ポート入力（デフォルト）
0	1	GPIO 以外の機能 1（GPIOI[1:0]のみ設定可能）
1	0	GPIOIx ポート出力
1	1	GPIO 以外の機能 2（GPIOI[0]のみ設定可能）

26.3.2.10 GPIOJ レジスタ

GPIOJ データレジスタ　(GPIOJ_DATA)

GPIO[0x48]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOJDATA[7:0]							
7	6	5	4	3	2	1	0

GPIOJ のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOJ が出力設定時にはこのレジスタの内容が読み出され、GPIOJ が入力設定時には端子の状態を読むことができます。

GPIOJ 端子機能レジスタ　(GPIOJ_FNC)

GPIO[0x4C]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPJ7MD [1:0]		GPJ6MD [1:0]		GPJ5MD [1:0]		GPJ4MD [1:0]	
15	14	13	12	11	10	9	8
GPJ3MD [1:0]		GPJ2MD [1:0]		GPJ1MD [1:0]		GPJ0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOJ の端子機能を選択します。GPIOJ のポートに対して 2Bit が端子機能選択に使われます。

表 26.11　PortJ の端子選択機能

GPJxMD1	GPJxMD0	端子機能
0	0	GPIOJ0 ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIOJ0 ポート出力
1	1	予約

26.3.2.11 GPIOK レジスタ

GPIOK データレジスタ　(GPIOK_DATA)

GPIO[0x50]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
GPIOKDATA[7:0]							
7	6	5	4	3	2	1	0

GPIOK のデータレジスタです。書き込みと読出しが可能です。このレジスタのリードについては GPIOK が出力設定時にはこのレジスタの内容が読み出され、GPIOK が入力設定時には端子の状態を読むことができます。

GPIOK 端子機能レジスタ　(GPIOK_FNC)

GPIO[0x54]　　初期値 = 0x0000_0000　　Read/Write

n/a							
31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
GPK7MD [1:0]		GPK6MD [1:0]		GPK5MD [1:0]		GPK4MD [1:0]	
15	14	13	12	11	10	9	8
GPK3MD [1:0]		GPK2MD [1:0]		GPK1MD [1:0]		GPK0MD [1:0]	
7	6	5	4	3	2	1	0

GPIOK の端子機能を選択します。GPIOK のポートに対して 2Bit が端子機能選択に使われます。

表 26.12　PortK の端子選択機能

GPKxMD1	GPKxMD0	端子機能
0	0	GPIOK0 ポート入力（デフォルト）
0	1	GPIO 以外の機能 1
1	0	GPIOK0 ポート出力
1	1	予約

GPIOA&B IRQ に関連するレジスタ

以下の GPIO[0x60]～GPIO[0x6C]は IRQ14 の割り込み要求を設定するためのレジスタです。

GPIOA&B IRQ TYPE

GPIOA&B IRQ タイプレジスタ　(GPIOAB_ITYP)

GPIO[0x60]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
PORTB_IRQ_TYPE [7:0]							
7	6	5	4	3	2	1	0
PORTA_IRQ_TYPE [7:0]							

Bits [15:8]：　**PORTB_IRQ_TYPE**

Bits [7:0]：　**PORTA_IRQ_TYPE**

各 Bit は以下のようにレベルトリガかエッジ割り込みのいずれかを選択できます。

0： 割り込み要求をレベルトリガとして取り扱います。

1： 割り込みをエッジとして取り扱います。

GPIOA&B IRQ Polarity

GPIOA&B IRQ 極性レジスタ　(GPIOAB_IPOL)

GPIO[0x64]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
PORTB_IRQ_POL [7:0]							
7	6	5	4	3	2	1	0
PORTA_IRQ_POL [7:0]							

Bits [15:8]：　**PORTB_IRQ_POL**

Bits [7:0]：　**PORTA_IRQ_POL**

0： 割り込み要求がレベルのとき High レベルが割り込み要求になります。
　　割り込み要求がエッジの場合には信号の立ち上がりが割り込み要求になります。

1： 割り込み要求がレベルのとき Low レベルが割り込み要求になります。
　　割り込み要求がエッジの場合には信号の立ち下がりが割り込み要求になります。

GPIOA&B IRQ ENABLE

GPIOA&B IRQ イネーブルレジスタ　(GPIOAB_IEN)

GPIO[0x68]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
PORTB_IEN [7:0]							
7	6	5	4	3	2	1	0
PORTA_IEN [7:0]							

Bits [15:8]：　**PORTB_IEN[7:0]**

PORTB[7:0]の割り込み許可／禁止に対応します。

0： GPIOB からの割り込み禁止

1： GPIOB からの割り込み許可

Bits [7:0]：　**PORTA_IEN[7:0]**

PORTA[7:0]の割り込み許可／禁止に対応します。

0： GPIOA からの割り込み禁止

1： GPIOA からの割り込み許可

GPIOA&B IRQ STATUS & Clear

GPIOA&B IRQ ステータス&クリアレジスタ　(GPIOAB_ISTS)

GPIO[0x6C]　　初期値 = 0x0000_0000　　Read/Write

31	30	29	28	27	26	25	24
n/a							
23	22	21	20	19	18	17	16
n/a							
15	14	13	12	11	10	9	8
PORTB_IRQ [7:0]							
7	6	5	4	3	2	1	0
PORTA_IRQ [7:0]							

Bits [15:8]：　**PORTB_IRQ[7:0]**

PORTB[7:0]の各 Bit からの割り込みステータスとクリアに対応します。

[Read]

0： 割り込み要求はありません。

1： 割り込み要求があることを示します。

[Write]

0： 何も変化しません。

1： “1”を書き込むことにより、割り込み要求要因がクリアされます。

Bits [7:0]：　**PORTA_IRQ[7:0]**

PORTA[7:0]の各 Bit からの割り込みステータスとクリアに対応します。

[Read]

0： 割り込み要求はありません。

1： 割り込み要求があることを示します。

[Write]

0： 何も変化しません。

1： “1”を書き込むことにより、割り込み要求要因がクリアされます。

Note: GPIOA と GPIOB からの各 Bit からの割り込み要求は“ロジカル OR”されて割り込みコントローラに通知されます。そのため、GPIO IRQ ハンドラ内でどのポートからの割り込み要求かをソフトウェアで判定する必要があります。

26.4 GPIOA および GPIOB の割り込みロジック

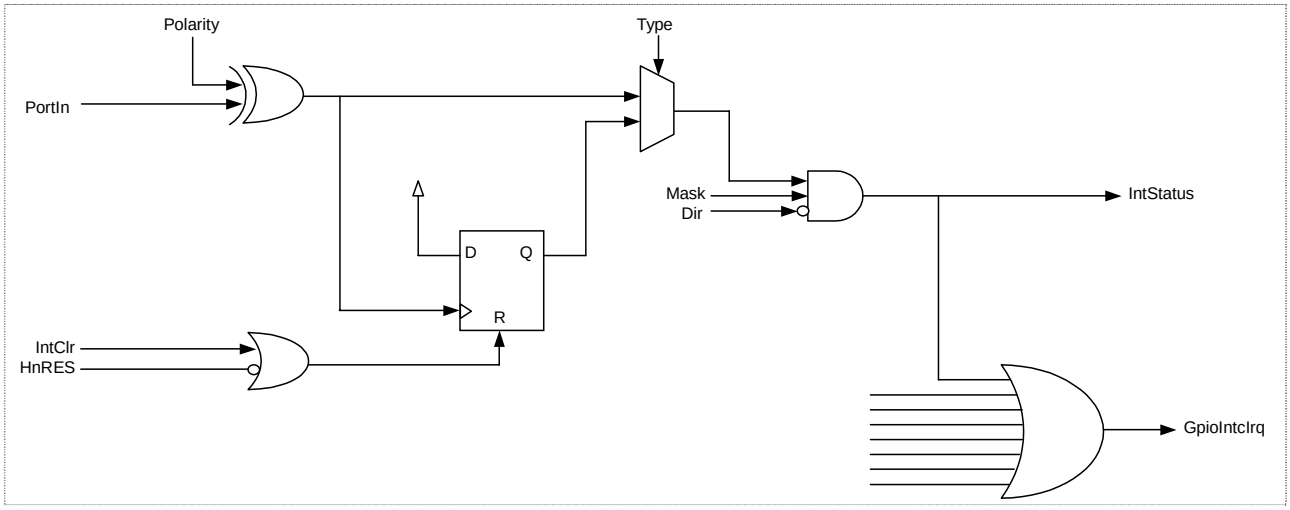


図 26.1　GPIOA and GPIOB の Interrupt Logic

注意：割り込みのタイプまたは極性を変更した際には偽の割り込みが入ることがあります。これを防ぐために、タイプや極性を変更したときには必ず割り込み要因をクリアしてから使用してください。

27. A/D 変換コントローラ(ADC)

27.1 概要

8 本のアナログ信号 ADIN[7:0]を入力し、内蔵している 10 ビット A/D 変換器(ADC)により A/D 変換を行い、デジタル信号を出力します。また、ADC をコントロールするコントローラを持っています。
アナログ入力電圧範囲は 0V～ 3.3V で、8 つのチャネルを連続的にサンプリング可能です。
変換時間は 1 回あたり 20µs 以下(システムクロック≒49MHz 設定時)で、変換が終了した時点で割り込みを発生させることができます。

27.2 ブロック図

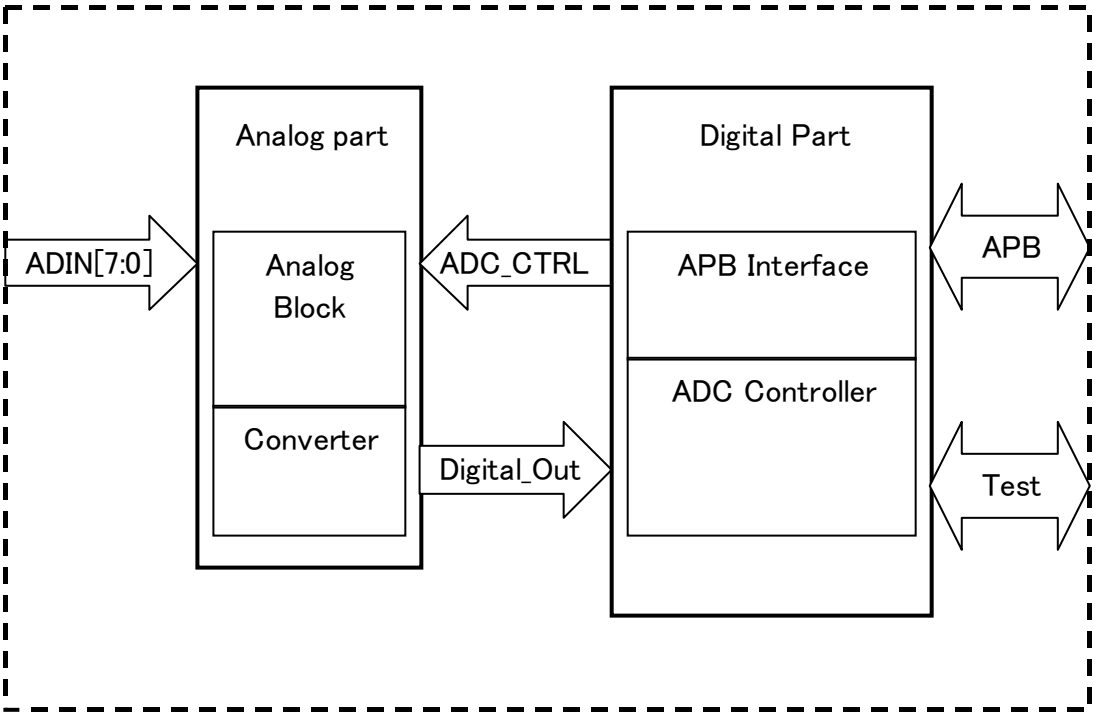


図 27.1　ADC ブロック図

27.3 外部端子

ADC 関連の外部端子は以下の通りです。

表 27.1　外部端子（CF）

端子名	入出力	端子機能	マルチプレクス端子／備考
ADIN[7:0]	入力	アナログ信号入力出力	

27.4 動作ステート

ADC は以下のステートにより動作を行います。

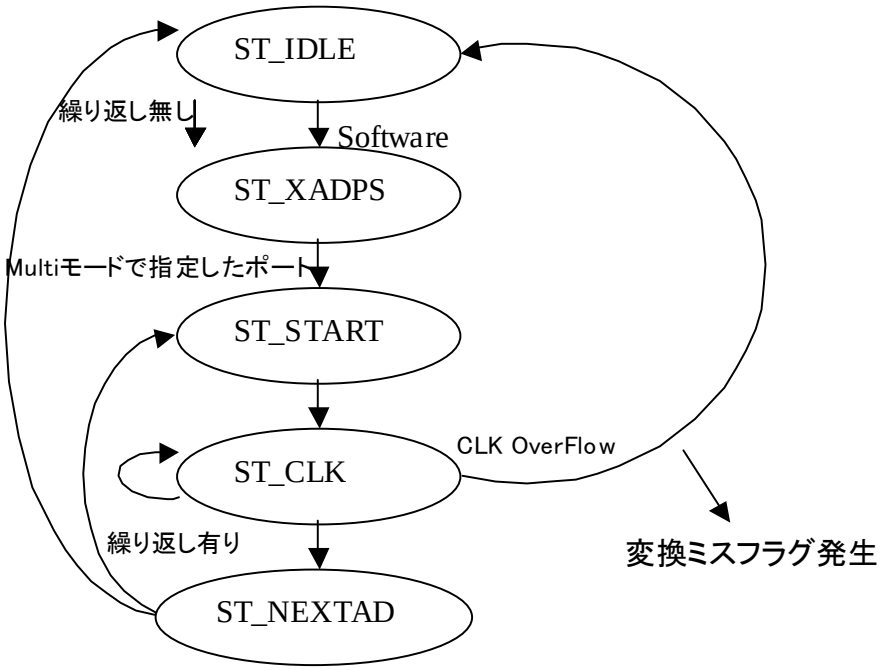


図 27.2　ADC 動作ステート

ST_IDLE　：　初期状態はこのステートです。
ソフトウェアによる入力が行なわれるまでこのステートを保持します。

ST_XADPS　：　A/D 変換を開始する信号 XADPS を起動します。
このステートで A/D 電源が ON になります。

ST_START　：　START 信号を起動します。

ST_CLK　：　A/D 変換を行うステートです。
A/D 変換には一定時間が必要ですが、何らかの要因により一定時間を超える場合はエラーとして ST_IDLS ステートに戻ることになります。このとき ADC フラグレジスタのビット 3 に“1”が書き込まれます。（このときの A/D 変換結果の値は保証されません）
シングルポートの場合は、A/D 変換後 ST_IDLE に戻ります。

ST_NEXTAD　：　A/D 変換をマルチポートで指定し、全ての変換が終了していない場合は、ST_START に戻り A/D 変換を再開します。
マルチポートで指定した全てのポートが終了した場合は ST_IDLE に戻ります。

27.5 レジスタ

27.5.1 レジスター覧

ADC レジスタのベースアドレスは、0xFFFD_C000 です。

表 27.2 レジスター覧（ベースアドレス：0xFFFD_C000）

Address Offset	Register Name	Abbreviation Name	Default Value	R/W	Data Access Size
0x00	ADC Data Register 0	ADCDT0	0x0000	RO	16 (/32)
0x04	ADC Data Register 1	ADCDT1	0x0000	RO	16 (/32)
0x08	ADC Data Register 2	ADCDT2	0x0000	RO	16 (/32)
0x0C	ADC Data Register 3	ADCDT3	0x0000	RO	16 (/32)
0x10	ADC Data Register 4	ADCDT4	0x0000	RO	16 (/32)
0x14	ADC Data Register 5	ADCDT5	0x0000	RO	16 (/32)
0x18	ADC Data Register 6	ADCDT6	0x0000	RO	16 (/32)
0x1C	ADC Data Register 7	ADCDT7	0x0000	RO	16 (/32)
0x20	ADC Control Register	ADCCTL	0x00	R/W	8 (/16/32)
0x24	ADC Flag Register	ADCFLG	0x0000	R/W	16 (/32)

27.5.2 レジスタ詳細説明

ADC Data Register 0 (ADCDT0)

ADC[0x00] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 0 [9:2] RO							

7	6	5	4	3	2	1	0
Data 0 [1:0] RO		Reserved RO					

Bit [15:6] : **Data 0 [9:0]**
チャネル 0 10bit A/D 変換データ[9:0]
チャネル 0 (ADIN0 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Data Register 1 (ADCDT1)

ADC[0x04] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 1 [9:2] RO							

7	6	5	4	3	2	1	0
Data 1 [1:0] RO		Reserved RO					

Bit [15:6] : **Data 1 [9:0]**
チャネル 1 10bit A/D 変換データ[9:0]
チャネル 1 (ADIN1 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Data Register 2 (ADCDT2)

ADC[0x08] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 2 [9:2] RO							
Data 2 [1:0] RO		Reserved RO					
7	6	5	4	3	2	1	0

Bit [15:6] : **Data 2 [9:0]**
チャネル 2 10bit A/D 変換データ[9:0]
チャネル 2 (ADIN2 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Data Register 3 (ADCDT3)

ADC[0x0C] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 3 [9:2] RO							
Data 3 [1:0] RO		Reserved RO					
7	6	5	4	3	2	1	0

Bit [15:6] : **Data 3 [9:0]**
チャネル 3 10bit A/D 変換データ[9:0]
チャネル 3 (ADIN3 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Data Register 4 (ADCDT4)

ADC[0x10] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 4 [9:2] RO							
Data 4 [1:0] RO		Reserved RO					
7	6	5	4	3	2	1	0

Bit [15:6] : **Data 4 [9:0]**
チャネル 4 10bit A/D 変換データ[9:0]
チャネル 4 (ADIN4 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Data Register 5 (ADCDT5)

ADC[0x14] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 5 [9:2] RO							
Data 5 [1:0] RO		Reserved RO					
7	6	5	4	3	2	1	0

Bit [15:6] : **Data 5 [9:0]**
チャネル 5 10bit A/D 変換データ[9:0]
チャネル 5 (ADIN5 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Data Register 6 (ADCDT6)

ADC[0x18] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 6 [9:2] RO							
Data 6 [1:0] RO		Reserved RO					
7	6	5	4	3	2	1	0

Bit [15:6] : **Data 6 [9:0]**
チャネル 6 10bit A/D 変換データ[9:0]
チャネル 6 (ADIN6 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Data Register 7 (ADCDT7)

ADC[0x1C] 初期値 = 0x0000 Read Only

15	14	13	12	11	10	9	8
Data 7 [9:2] RO							
Data 7 [1:0] RO		Reserved RO					
7	6	5	4	3	2	1	0

Bit [15:6] : **Data 7 [9:0]**
チャネル 7 10bit A/D 変換データ[9:0]
チャネル 7 (ADIN7 端子のアナログ入力)の変換結果が読み出します。

Bits [5:0] : **Reserved**

ADC Control Register (ADCCTL)							
ADC[0x20] 初期値 = 0x00							Read/Write
ADCFLG	ADCIEN	ADCSTT	ADCMLT	ADCEXT	ADCCH2	ADCCH1	ADCCH0
RW	R/W	R/W	R/W	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0

Bit 7 : **A/D End Flag**

A/D 変換が終了したことを示す。

0 (r) 変換中または動作停止中
1 (r) A/D 変換終了
0 (w) 本フラグをクリア
1 (w) 無効

このフラグは指定したチャネルすべての A/D 変換が終了すると 1 にセットされます。ADC 内のレジスタを更新（読み出し含み）することで自動的に 0 にリセットされます。また、このビットが 1 の場合に 0 を書き込んだときも 0 にリセットされます。
割り込み制御を行わない場合（bit6 を“0”とした場合）、このレジスタをモニタせずに[0x24]ビット 2 の ADCNV を使用して下さい。
このビットは AD 変換終了をモニタしているレジスタであり、データのセットを保証する値ではありません、データセットを保証するのは[0x24]ビット 2 の ADCNV になります。

Bit 6 : **A/D 割り込み許可**

A/D 変換終了割り込みを許可/禁止します。

0 (r/w) 割り込み禁止
1 (r/w) 割り込み許可

割り込み許可は必ず“1”にして下さい。割り込みを発生させたくない場合は割り込み制御モジュールを割り込み禁止にして下さい。

Bit 5 : **A/D 変換を開始します。**

0 (r) A/D 変換終了
1 (r/w) A/D 変換開始

指定したチャネルすべての A/D 変換が終了すると、このビットは自動的に 0 に戻ります。
A/D 変換中に 0 を書き込んだ場合でも変換動作は終了しません。
A/D 変換が規定のクロックで終了しない場合は 0x24[3]のビットに“1”が発生し、エラー状態を確認することができます。

Bit 4 : **マルチモード選択**

変換モード(シングルモードあるいはマルチモード)を選択します。

0 (r/w) シングルモード
1 (r/w) マルチモード

シングルモード時は 1 チャネルの入力のみを A/D 変換します。マルチモードが選択された場合は、すべてのチャネルについて、連続的に A/D 変換します。どちらの場合も、変換するチャネルはチャネル選択ビット(ビット[2:0])により切換え可能です。

Bit 3 : **外部トリガイネーブル**

0 外部トリガディセーブル
1 外部トリガイネーブル

このレジスタがイネーブルになっている場合は、Timer B からの立ち下がりエッジをトリガとして A/D Convert がスタートします。
変換実行中に Timer B の立ち下がりエッジが発生した場合、現在の変換を終了したところで A/D Convert が終了するため、再度 A/D Convert を実施して下さい。
Timer B から入力する信号は 1μs 以上になるようにして下さい。

Bit [2:0] : **チャネル選択ビット**[2:0]
A/D 変換するチャネルを選択します。シングルモードとマルチモードで、選択肢は変わります。

Bits[2:0]	機能説明	
CH[2:0]	Single Mode	Multi Mode
000	AN0	AN0
001	AN1	AN0-AN1
010	AN2	AN0AN2
011	AN3	AN0AN3
100	AN4	AN0-AN4
101	AN5	AN0-AN5
110	AN6	AN0-AN6
111	AN7	AN0-AN7

A/D 変換は外部トリガイネーブルを“1”にすることで、Timer B からのトリガで行なうことができます。このため、変換後のデータを転送する場合は A/D 変換後の割り込みを検知し、割り込み発生後に読み出すことを推奨します。
変換後のデータを読み出すことで End Flag は“0”になります。End Flag が“1”の状態で Timer B からの新たな入力を受けると End Flag を“0”として新たな A/D 変換が開始されます。
A/D 変換中にデータ転送をした場合の値は保証できません。

ADC Flag Register (ADCFLG)
ADC[0x24] 初期値 = 0x00 Read/Write

Reserved				ADCERR	ADCCNV	Reserved	
-	-	-	-			-	-
7	6	5	4	3	2	1	0

Bit [7:4] : **Reserved**

Bit 3 : **ADCERR**
A/D 変換中に異常が発生した場合に“1”が書き込まれます。
“1”が書き込まれた要因は A/D 変換クロックが 15 回以上発生しても A/D 変換が終了しない場合になります。このときデータレジスタに変換データは書き込まれないため、変換前のレジスタ値が読み出されることになります。このフラグが発生した後でも、A/D 変換が正常に行われればデータレジスタの値は正しく更新されます。
“1”が書き込まれた場合、システムリセットかこのビットに“1”を書き込むことで、このビットを“0”にします。

Bit 2 : **ADCCNV**
A/D 変換が正常に終了し、データレジスタに変換データが格納されたことを示す。
マルチモード時は、設定した全てのチャネルの A/D 変換が終了し、全てのデータがデータレジスタに格納されたことを示す。このビットを有効にするためには、ADC[0x20]-Bit6 の A/D 割り込み許可をイネーブルにする必要があります。
ADCCNV は Analog Part から出力される信号をトレースしています。AD 変換が終了した後、Analog Part から変換終了の Flag が出力されレジスタに変換データが格納されると“1”になります。再度 AD 変換を開始すると、このビットは“0”に戻ります。

Bit [1:0] : **Reserved**

27.6 使用例

・シングルモード

シングルモードは、単チャンネルのアナログ入力を変換します。

A/D 変換はソフトウェアまたは外部トリガ入力によって ADC 制御レジスタ (以下 ADC[0x20]))のビット 5 が“1”にセットされると指定したチャンネルの A/D 変換が開始されます。

以下にシングルモードで ADIN1 端子のみを使用する場合の動作を示します。

1. シングルモード、ADIN1 入力および割り込み許可の設定を行います(ADC[0x20]のビット[7:0]を 61h に設定し A/D 変換を開始します)。
2. A/D 変換が終了すると、ADC[0x20]のビット[7]に“1”がセットされ、ADC[0x20]のビット[5]が“0”に戻り、次の A/D 変換開始まで待機します。その後変換結果がデータレジスタ 1 にセットされます。
3. 割込みコントローラで ADC 変換割り込みを許可している場合は、変換終了後、割り込みが受けつけられますので､割り込み処理ルーチン内にデータレジスタ 1 の読み出しを設定することで、ADIN1 の変換結果を読み出すことが出来ます。
4. 割り込み処理ルーチン終了後、ADC[0x20]のビット[5]を“1”にセットするとで次の変換を行うことが出来ます。

・マルチモード

マルチモードは、複数チャンネル(1 チャンネルを含む)のアナログ入力を順次変換します。

マルチモードおよび ADC[0x20]のビット[2:0]に “001 が”設定された場合は､ADIN0 の変換が終了した後、続けて ADIN1 の変換が開始されます。A/D 変換は、指定した全てのチャンネルを一巡して変換が終了します｡変換後は End Flag(ADCFLG)が“1”にセットされデータ変換が終了したことを示します。

変換されたデータは各チャンネルに対応した ADC データレジスタに保持されます。

以下にマルチモードで ADIN0/1/2 端子を使用する場合の動作を示します。

1. マルチモード、ADIN0 - ADIN1 - ADIN2 入力、割り込み許可を選択します(ADC[0x20]のビット[7:0]を 72h を設定し A/D 変換を開始します)。
2. A/D 変換は ADIN0 から始まります。変換が終了すると変換結果がデータレジスタ 0 にセットされます。同様に ADIN1、ADIN2 の順に変換され変換結果が各データレジスタにセットされますが、最終のとなる ADIN2 のデータは End Flag をセットした後に格納されます。
3. ADIN2 の変換が終了すると、ADC[0x20]のビット[7]に“1”がセットされ、ADC[0x20]のビット[5]が“0”に戻り、次の A/D 変換は開始まで待機します。その後変換結果がデータレジスタ 2 にセットされます。
4. 割込みコントローラ側で ADC 変換終了割り込みを許可している場合は、変換終了後、割り込みが受けつけられますので、割り込み処理ルーチン内にデータレジスタ 1/2/3 の読み出しを設定することで、ADIN1/2/3 の変換結果を読み出すことが出来ます。
5. 割り込み処理ルーチン終了後、ADC[0x20]のビット[5]を“1”にセットするとで次の変換を行うことが出来ます。

27.7 入力電圧と変換データ値

ADC Data Register0-7[15:0]に入力電圧により、下図の値が格納されます。
本 ADC は 10bitAD 変換を行いますが、±2bit の誤差がありますので ADC Data Register[7:0]の bit15-8 を参照願います。

入力電圧 [V]	変換基準値	ADC Data Register0-7[15:0]		備考
		Min.	Max.	
0.0	0	0x00_00	0x00_C0	
0.3	0x5D	0x16_80	0x18_00	
0.6	0xB5	0x2D_C0	0x2F_40	
0.9	0x117	0x45_00	0x46_80	
1.2	0x1174	0x5C_40	0x5D_C0	
1.5	0x1D1	0x73_80	0x75_00	
1.8	0x22E	0x8A_C0	0x8C_40	
2.1	0x28B	0xA2_00	0xA3_80	
2.4	0x2E8	0xB9_40	0xBA_C0	
2.7	0x345	0xD0_80	0xD2_00	
3.0	0x3A2	0xE7_C0	0xE9_40	
3.3	0x3FF	0xFF_00	0xFF_C0	

28. エリアセンサ(ARS)

28.1 概要

カメラインタフェースからの YUV 入力信号を画面イメージで 16 分割し、それぞれの矩形エリアごとに YUV 成分（どれか 1 つを選択）を積算してレジスタに保持し、以前の積算値と比較することで、部分的な動きの検出をリアルタイムに、CPU に負担をかけずに実行することができます。

【２M ピクセルカメラ（1632*1224）の例】
積算されるデータ幅は 1632 x 1224 x 63/16 = 7865424（Dec) = 780450(Hex) = 23bit ですが、
4 ピクセルに 1 回だけ積算を行う為 21bit となります。

Area 0	Area 1	Area 2	Area 3
Area 4	Area 5	Area 6	Area 7
Area 8	Area 9	Area 10	Area 11
Area 12	Area 13	Area 14	Area 15

図 28.1　画像分割エリア分布（Area0～15）

28.1.1 機能

エリアセンサが実現している機能は以下のとおりです。

- 画像イメージを 16 分割して各々のエリアで YUV 成分から 1 つを選択し積算します
- 16 分割された矩形エリアは、データサイズ（H/V Sync）により自動的に計算します
- 積算・比較の周期を設定可能ですので速い動きや遅い動きに対応できます
- 積算値の変化率を設定し、設定された変化率よりも大きな変化を感知して割り込みを発生させます
- ステータスレジスタを読むことにより、どの矩形エリアにコントラストに変化があったのか認識出来ます
- 割り込み発生をエリアごとにイネーブルすることやクリアすることが出来ます
- 各エリアの積算値（21bit）及び前フレームとの大小関係を読み出し可能
- レジスタによりカメラ 1/カメラ 2 を切り替え可能

28.2 ブロック図

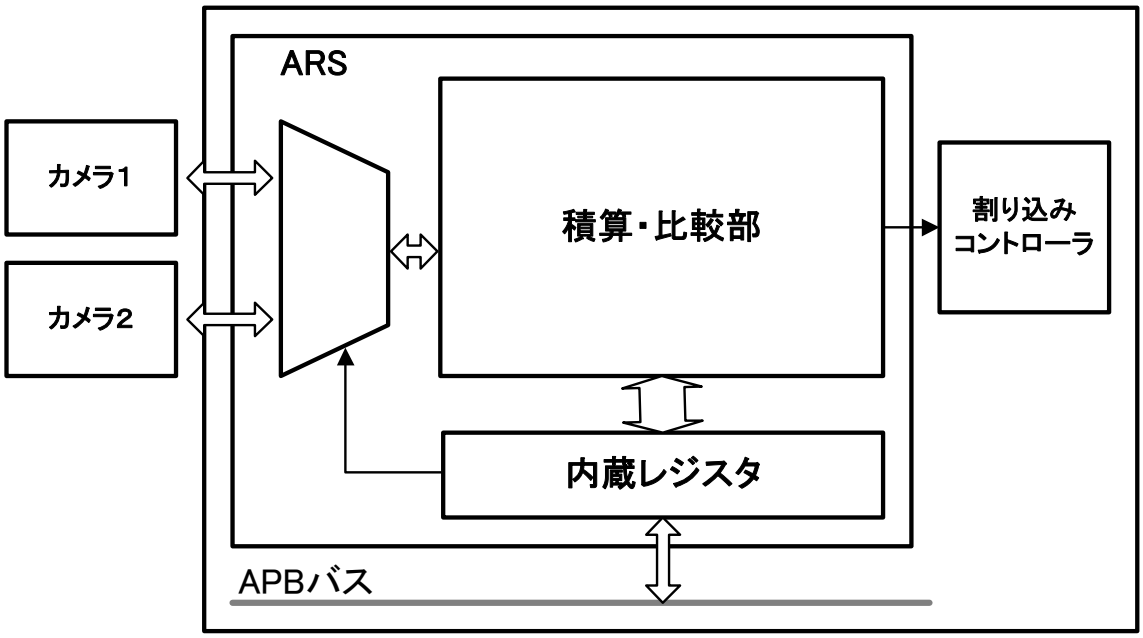


図 28.2　ARS ブロック図

28.3 外部端子

ARS に関連する外部端子はありません。なお、各カメラの入力及びクロックは内部カメラ入力用モジュールより供給されております。

28.4 レジスタ説明

28.4.1 レジスター覧

ARS の制御レジスタが配置されているデフォルトのベースアドレスは、0xFFFE_7000 です。
特に指定のない場合、予約されていないレジスタビットのデフォルト値はすべて“0”です。

表 28.1　ARS レジスター覧

アドレス オフセット	レジスタ名称	初期値	R/W	データ アクセスサイズ
ARS 制御レジスタ　ベースアドレス：0xFFFE_7000				
0x00	ARS コントロールレジスタ(ARSCTRL)	0x0000_0080	R/W	32
0x04	ARS エリア選択レジスタ(ARSASEL)	0x0000_0000	R/W	32
0x0C	ARS ステータスレジスタ(ARSSTAT)	0x0000_0000	R/W	32
0x10	ARS 割り込み要因レジスタ(ARSINT)	0x0000_0000	R/W	32
0x40	ARS 積算レジスタ 0 (ARSADD0)	0x0000_0000	RO	32
0x44	ARS 積算レジスタ 1 (ARSADD1)	0x0000_0000	RO	32
0x48	ARS 積算レジスタ 2 (ARSADD2)	0x0000_0000	RO	32
0x4C	ARS 積算レジスタ 3 (ARSADD3)	0x0000_0000	RO	32
0x50	ARS 積算レジスタ 4 (ARSADD4)	0x0000_0000	RO	32
0x54	ARS 積算レジスタ 5 (ARSADD5)	0x0000_0000	RO	32
0x58	ARS 積算レジスタ 6 (ARSADD6)	0x0000_0000	RO	32
0x5C	ARS 積算レジスタ 7 (ARSADD7)	0x0000_0000	RO	32
0x60	ARS 積算レジスタ 8 (ARSADD8)	0x0000_0000	RO	32
0x64	ARS 積算レジスタ 9 (ARSADD9)	0x0000_0000	RO	32
0x68	ARS 積算レジスタ 10 (ARSADD10)	0x0000_0000	RO	32
0x6C	ARS 積算レジスタ 11 (ARSADD11)	0x0000_0000	RO	32
0x70	ARS 積算レジスタ 12 (ARSADD12)	0x0000_0000	RO	32
0x74	ARS 積算レジスタ 13 (ARSADD13)	0x0000_0000	RO	32
0x78	ARS 積算レジスタ 14 (ARSADD14)	0x0000_0000	RO	32
0x7C	ARS 積算レジスタ 15 (ARSADD15)	0x0000_0000	RO	32

28.4.2 レジスタ詳細説明

ARS コントロールレジスタ

ARS[0x00]　　初期値 = 0x0000_0080　　Read/Write

SWRST (WO)	Reserved		ADDCYC (R/W)				
15	14	13	12	11	10	9	8
ADDOFF (RO)	CHGRATE （R/W）			CAMSEL （R/W）	YUVSEL （R/W）		ARSEN （R/W）
7	6	5	4	3	2	1	0

Bit 15：　**SWRST　ソフトウェアリセット**

エリアセンサモジュールをリセットできます。

0： 何もしない

1： ソフトウェアリセットを実行

Bits [14:13]：　**(Reserved)**

Bits [12:8]：　**ADDCYC　積算周期設定**

積算・比較を行う周期を設定します。このレジスタに設定した値でカメラインタフェースから送られてくる画像イメージの YUV 成分を積算し、以前の値との比較を行います。

00000：積算しない（初期値）

00001：毎フレーム積算

00010：2 フレーム毎に積算

:

11111：31 フレーム毎に積算

Bit 7 : **ADDOFF　積算 On/Off 状態表示　（Read Only）**

積算の On/Off の状態を表示します。積算値やステータスレジスタを読み出す場合にはこのレジスタを参照し、Off になっていることを確認してから読み出しを行ってください。

0： 積算 On（初期値）
1： 積算 Off

Bits [6:4] : **CHGRATE　変化率設定**

積算値に以下の閾値以上の変化があった場合にステータスレジスタ（ARS[0x08]）の各ステータスビットに“1”をセットし、割り込みを発生させます。

000：約 1.5%（初期値）
001：約 3%
010：約 6%
011：約 13%
100：約 25%
101：約 50%
11x：(Reserved)

Bit 3 **CAMSEL　カメラ選択**

積算するカメラの入力を選択します。

0： カメラ 0（初期値）
1： カメラ 1

Bits [2:1] **YUVSEL　積算エレメント選択**

Y/U/V の中から積算する値を選択します。

00：積算しない（初期値）
01：Y を積算
10：U を積算
11：V を積算

Bit 0 : **ARSEN　ARS イネーブル**

ARS の積算機能の On/Off を設定します。このビットに“1”を書き込みますと、次のフレームより積算を開始します。また、このビットに“0”を書き込みますと現在のフレームが終了した後に ARS の積算機能が停止します。

0： ARS Disable
1： ARS Enable

ARS エリア選択レジスタ

ARS[0x04]　　初期値 = 0x0000_0000　　Read/Write

ASEL[15:8]							
15	14	13	12	11	10	9	8
ASEL[7:0]							
7	6	5	4	3	2	1	0

Bit[15:0] : **ASEL　エリア選択レジスタ**

変化を検出するエリアを選択します。（複数選択可）各分割エリアの対応は図 28.1 分割エリアの分布を参照して下さい。

ARS ステータスレジスタ

ARS[0x0C] 初期値 = 0x0000_0000 Read/Write

ARSSTAT [15:8]							
15	14	13	12	11	10	9	8
ARSSTAT[7:0]							
7	6	5	4	3	2	1	0

Bit[15:0]： **ARSSTAT エリアステータスレジスタ**

各分割エリアに設定した変化率以上の変化があったかどうかを示します。

0： エリア内に変化なし

1： エリア内に変化あり

また、各ビットに“1”を書き込むことでクリアすることができます。

ARS 割り込みステータスレジスタ

ARS[0x10] 初期値 = 0x0000_0000 Read/Write

ARSINT [15:8]							
15	14	13	12	11	10	9	8
ARSINT[7:0]							
7	6	5	4	3	2	1	0

Bits [15:0]： **ARSINT[15:0] ARS 割り込み要因レジスタ**

各分割エリアで割り込み要因が発生したかを示します。また、各ビットに“1”を書くことで要因をクリアすることができます。

ARS 積算値レジスタ 0～15

ARS[0x40] ～ ARS[0x7C] 初期値 = 0000_0000 Read Only

GTO	（Reserved）										ADDV[20:16]				
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDV[15:0]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit [31]： **GTO 積算値増加/減少レジスタ**

現在の積算値が以前の積算値と比較して増加したか減少したかを示します。

0： 減少した

1： 増加した

Bits [20:0]： **ADDV [20:0] エリア積算値レジスタ**

各エリアの積算値を示します。

29. 絶対最大定格

29.1 絶対最大定格

(VSS = 0 [V])

項目	記号	定格値	単位
電源電圧	HVDD,UVDD3, C1VDD,C2VDD, SDVDD,AVDD	-0.3 ~ 4.0	V
	LVDD, UPVDD,UXVDD, RTCVDD,PLLVDD	-0.3 ~ 2.5	V
入力電圧	HVI	-0.3 ~ HVDD+0.5	V
	LVI	-0.3 ~ LVDD+0.5	V
	USBVBUS	-0.3 ~ 6.0	V
出力電圧	HVO	-0.3 ~ HVDD+0.5	V
	LVO	-0.3 ~ LVDD+0.5	V
出力電流／ピン	I_{OUT}	± 10	mA
保存温度	T_{stg}	-65 ~ 150	°C

29.2 推奨動作条件（2 電源、3.3V 対応入出力バッファ）

(UVSS = PVSS = XVSS = 0 [V]
VSS = PLLVSS = AVSS = 0 [V])

項目		記号	Min.	Typ.	Max.	単位
電源電圧（高電圧）	I/O セル用電源	HVDD	3.00	3.30	3.60	V
	USB 用電源	UVDD3	3.00	3.30	3.60	V
	カメラ 1 I/F 用電源	C1VDD	2.40	3.00	3.60	V
	カメラ 2 I/F 用電源	C2VDD	2.40	3.00	3.60	V
	SDRAM I/F 用電源	SDVDD	2.70	3.00	3.60	V
	A/D 用電源	AVDD	3.00	3.30	3.60	V
電源電圧（低電圧）	コア（内部）用電源	LVDD	1.65	1.80	1.95	V
	USB 用電源	UPVDD	1.65	1.80	1.95	V
	USB 用電源	UXVDD	1.65	1.80	1.95	V
	アナログ（PLL）用電源	PLLVDD	1.65	1.80	1.95	V
	RTC 用電源	RTCVDD	1.65	1.80	1.95	V
入力電圧	I/O セル用電源	HV_I	VSS	−	HVDD	V
	USB 用電源電圧	$UV3_I$	UVSS	−	UVDD3	V
	カメラ 1 I/F 用電源	$C1V_I$	VSS	−	C1VDD	V
	カメラ 2 I/F 用電源	$C2V_I$	VSS	−	C2VDD	V
	SDRAM I/F 用電源	SDV_I	VSS	−	SDVDD	V
	A/D 用電源	AV_I	AVSS	−	AVDD	V
	コア（内部）用電源	LV_I	VSS	−	LVDD	V
	USB 用電源	UPV_I	PVSS	−	UPVDD	V
	USB 用電源	UXV_I	XVSS	−	UXVDD	V
	アナログ（PLL）用電源	$PLLV_I$	PLLVSS	−	PLLVDD	V
	RTC 用電源	$RTCV_I$	VSS		RTCVDD	V
周囲温度		Ta	-40	25	85*	°C
入力立ち上がり時間（ノーマル入力）		t_{ri}	−	−	50	ns
入力立ち下がり時間（ノーマル入力）		t_{fa}	−	−	50	ns
入力立ち上がり時間（シュミット入力）		t_{ri}	−	−	5	ms
入力立ち下がり時間（シュミット入力）		t_{fa}	−	−	5	ms

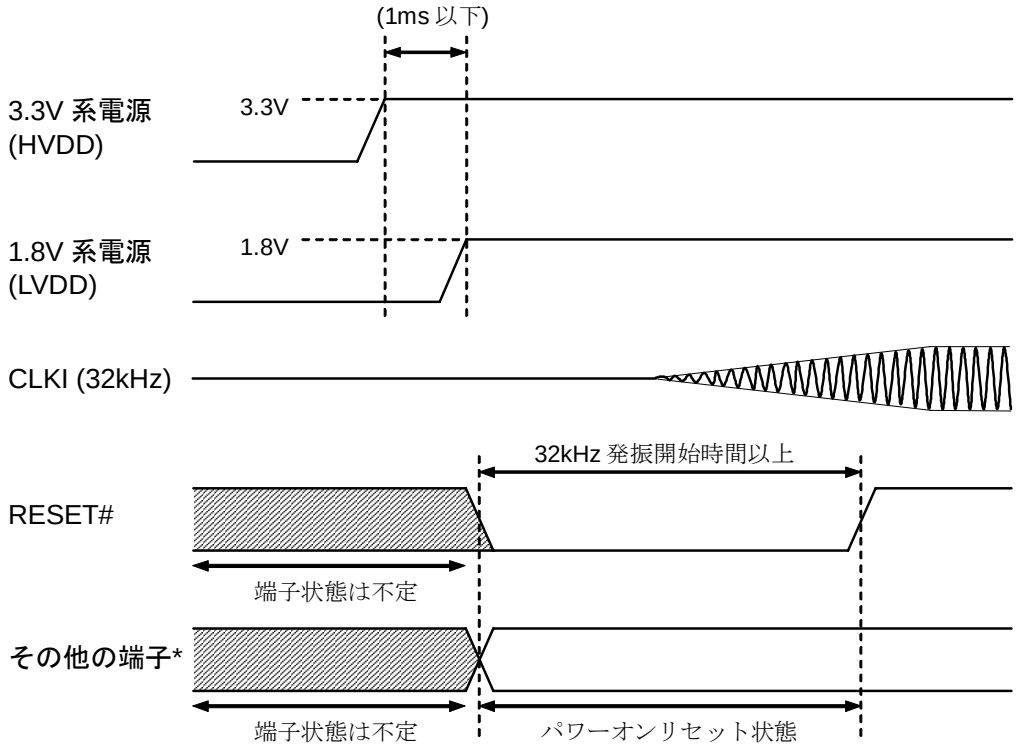
*: この温度範囲は、Tj = -40～125[℃]を想定した推奨周囲温度です。

29.3 電源投入タイミング

3.3V 系電源（HVDD）と 1.8V 系電源（LVDD）の投入順序については以下のようにお願いします。

(1) 3.3V 系電源を先に投入してください。1ms 以内に 1.8V 系電源を投入してください。この間隔をできるだけ短くすることを推奨します。

(2) HVDD1 および LVDD が安定した後も、32KHz 発振開始時間以上（例えば 100ms 以上）RESET# をローレベルに保ってください。



注意：“その他の端子”とは電源/GND、RESET#、クロック関連、アナログ端子を除く

図 29.1　電源投入タイミング

29.4 電源切断タイミング

3.3V 系電源（HVDD）と 1.8V 系電源（LVDD）の切断順序については以下のようにお願いします。

(1) 電源投入時とは逆に、1.8V 系電源を先に切断してください。この後、1ms 以内に 3.3V 系電源を切断してください。この間隔をできるだけ短くすることを推奨します。

(2) 1.8V 系電源だけを切断した状態では端子状態は不定となります。この不定状態によってシステムが誤動作を起こさないようにシステム設計を行ってください。

30. 電気的特性

30.1 DC 特性

表 30.1　DC 特性（3.3V）

(HVDD = 3.3V ± 0.3V, VSS = 0V, Ta= -40～85℃)

項目	記号	条件		Min.	Typ.	Max.	単位
入力リーク電流	I_{LI}	－		-5	－	5	μA
オフステートリーク電流	I_{OZ}	－		-5	－	5	μA
高レベル出力電圧*	V_{OH}	I_{OH} = -4mA HVDD=Min.		HVDD -0.4	－	－	V
低レベル出力電圧*	V_{OL}	I_{OL} = 4mA HVDD=Min.		－	－	0.4	V
高レベル入力電圧	V_{IH1}	LVCMOS レベル、HVDD=Max.		2.2	－	－	V
低レベル入力電圧	V_{IL1}	LVCMOS レベル、HVDD=Min.		－	－	0.8	V
高レベル入力電圧	V_{T1+}	LVCMOS シュミット		1.4	－	2.7	V
低レベル入力電圧	V_{T1-}	LVCMOS シュミット		0.6	－	1.8	V
ヒステリシス電圧	V_{H1}	LVCMOS シュミット		0.3	－	－	V
高レベル入力電圧	V_{IH2}	LVTTL レベル、HVDD=Max.		2.0	－	－	V
低レベル入力電圧	V_{IL2}	LVTTL レベル、HVDD=Min.		－	－	0.8	V
プルアップ抵抗	P_{PU}	VI=0V		25	50	120	kΩ
プルダウン抵抗	P_{PD}	VI=HVDD	その他の端子*2	25	50	120	kΩ
			MD[15:0]端子	50	100	240	
入力端子容量	C_I	f=1MHz, HVDD = 0V		－	－	8	pF
出力端子容量	C_O	f=1MHz, HVDD = 0V		－	－	8	pF
入出力端子容量	C_{IO}	f=1MHz, HVDD = 0V		－	－	8	pF
消費電流 (LVDD)	I_{LOW}	ロースピードモード (32KHz)		－	220	－	μA
	I_{FO}	ハイスピードモード*3		－	150	－	mA
消費電力(RTCVDD)	I_{RLOW}	ロースピード	BUP#=HIGH	－	170	－	μA
	I_{RLOWS}	モード(32KHz)	BUP#=LOW		2		μA
	I_{RO}	ハイスピードモード*3		－	735	－	μA
消費電流 (PLLVDD)	I_{DDPLL}	PLL 周波数 = 50MHz		－	1	－	mA

* ： すべての出力および入出力端子が対象となります。
*2： MD[15:0]端子以外のプルダウン抵抗付きの端子
*3： ２台のカメラでそれぞれ 30fps@VGA で画像転送時

表 30.2　DC 特性（1.8V）

(RTCVDD = 1.8V ± 0.15V, VSS = 0V, Ta= -40～85℃)

項目	記号	条件	Min.	Typ.	Max.	単位
入力リーク電流	I_{LI}	－	-5	－	5	μA
オフステートリーク電流	I_{OZ}	－	-5	－	5	μA
高レベル入力電圧	V_{T1+}	LVCMOS シュミット	0.6	－	1.4	V
低レベル入力電圧	V_{T1-}	LVCMOS シュミット	0.3	－	1.1	V
ヒステリシス電圧	V_{H1}	LVCMOS シュミット	0.02	－	－	V
入力端子容量	C_I	f=1MHz, HVDD = 0V	－	－	8	pF

表 30.3　USBVBUS 判定電圧

項目	記号	条件	Min.	Typ.	Max.	単位
H レベルトリガ電圧	VB_{TH}	UVDD3 = 3.6V	1.86	－	2.85	V
L レベルトリガ電圧	VB_{TL}	UVDD3 = 3.0V	1.48	－	2.23	V
ヒステリシス電圧	VB_H	UVDD3 = 3.0V	0.31	－	0.64	V

30.2 AC 特性

30.2.1 AC 特性測定条件

HVDD、AVDD、UVDD3、 = 3.3V ± 0.3V
C1VDD、C2VDD = 3.0V ± 0.6V
SDVDD = 2.7V～3.6V
LVDD、UPVDD、UXVDD、PLLVDD、RTCVDD = 1.8V ± 0.15V
T_A = -40℃～85℃
C_L = 50pF（特記なき場合）

30.2.2 AC 特性タイミング一覧表

30.2.2.1 クロックタイミング

表 30.4 クロック（CLKI）タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
SYSCLKI 周波数	f_{clkI}	-	32.768	-	KHz	-
SYSCLKI 入力サイクル時間	t_{clkI}	-	1/ f_{OSC}	-	s	-
SYSCLKI ハイレベルパルス幅	t_{CLKIH}	5	-	-	μs	-
SYSCLKI ローレベルパルス幅	t_{CLKIL}	5	-	-	μs	-
SYSCLKI 立ち上がり時間（10%→90%）	t_{CLKIR}	-	-	12	μs	-
SYSCLKI 立ち下がり時間（90%→10%）	t_{CLKIF}	-	-	12	μs	-
システムクロック周波数	fsys	-	-	50	MHz	-
システムクロック周波数周期	Ts	1/fsys -	-	-	ns	-

30.2.2.2 CPU コントロール信号タイミング

表 30.5 CPU コントロール信号 タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
RESET#パルス幅	t_{RESW}	10	-	-	T_{CLKI}	*1
IRQ/FIQ パルス幅	t_{IRQW}	10	-	-	T_S	*2
クロック再開時間	t_{WAK}	-	-	4	T_S	*2
PLL 安定時間	t_{PLLST}	-	-	100	ms	-

*1： T_{CLKI} = 32KHz 単位。スレッショルド電圧を切る振幅が必要。
*2： T_S = システムクロックサイクル時間

30.2.2.3 バッテリーバックアップモードタイミング

表 30.6 バッテリーバックアップタイミング

項目	記号	Min.	Typ.	Max.	単位	備考
電源供給開始時 BUP#端子遅延時間	t_{BUPPO}	0	-	-	ns	
バックアップ開始時電源遮断時間	t_{BUPS}	10	-	-	ns	
バックアップ復帰時電源安定時間	t_{BUPH}	10	-	-	T_{CLKI}	*1
バックアップ解除後 RESET#有効期間	t_{BUPRST}	10	-	-	T_{CLKI}	*1

*1：T_{CLKI} = 32KHz 単位。スレッショルド電圧を切る振幅が必要。

30.2.2.4 カメラインタフェース (CAM) タイミング

表 30.7　カメラインタフェース（CAM）タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
CMVREF 立ち上がりエッジ→CMHREF 立ち上がりエッジ	t_{CAM1}	0	-	-	T_C	*3
水平ブランク期間	t_{CAM2}	4	-	-	T_C	*3
CMHREF 立ち下がりエッジ→CMVREF 立ち下がりエッジ	t_{CAM3}	0	-	-	T_C	*3
垂直ブランク期間	t_{CAM4}	1	-	-	Line	-
カメラ入力クロック期間	t_{CAM5}	1.6 (3.2)	-	-	T_S	*2
カメラ入力クロックローレベルパルス幅	t_{CAM6}	0.8 (1.6)	-	-	T_S	*2
カメラ入力クロックハイレベルパルス幅	t_{CAM7}	0.8 (1.6)	-	-	T_S	*2
データセットアップ時間	t_{CAM8}	10	-	-	ns	-
データホールド時間	t_{CAM9}	10	-	-	ns	-
CMVREF, CMHREF セットアップ時間	t_{CAM10}	10	-	-	ns	-
CMVREF, CMHREF ホールド時間	t_{CAM11}	10	-	-	ns	-

*2： T_S = システムクロックサイクル時間
　　Min 値は高速サンプリング時の値、(　)内は通常サンプリング時の Min 値
*3： T_C = カメラインタフェース入力クロックサイクル時間

30.2.2.5 メモリコントローラ (MEMC) タイミング

■ スタティックメモリコントローラ タイミング

表 30.8　スタティックメモリタイミング 1 (MCS0#/MCS1#)

項目	記号	Min.	Typ.	Max.	単位	備考
アドレス信号セットアップ時間	t_{MAS}	– 3.5	-	– 1	ns	-
アドレス信号ホールド時間(対 MOE#)	t_{MAHOE}	– 0.5	-	0	ns	-
アドレス信号ホールド時間(対 MWE#)	t_{MAHWE}	1Ts – 1	-	-	ns	*2
アドレス信号ホールド時間(対 MCSx#)	t_{MAHCE}	1Ts – 1	-	-	ns	*2
データ出力許可信号遅延時間	t_{MOED}	n_1Ts – 0.5	-	-	ns	*2 *4
データ出力許可信号有効期間	t_{MOEV}	n_2Ts – 0.5	-	-	ns	*2 *5
データ出力許可信号インターバル期間	t_{MOEI}	-	n_1	-	Ts	*2 *4
チップセレクトホールド時間(対 MOE#)	t_{MCEHOE}	– 0.5	-	0.5	ns	-
読み込みデータセットアップ時間	t_{MDRS}	-	-	17	ns	-
読み込みデータホールド時間	t_{MDRH}	0	-	-	ns	-
書き込み有効信号遅延時間	t_{MWED}	n_3Ts – 0.5	-	-	ns	*2 *6
書き込み有効信号有効期間	t_{MWEV}	n_4Ts+1 -n_3Ts- 0.5	-	-	ns	*2 *6 *7
書き込み有効信号インターバル期間	t_{MWEI}	-	n_3	-	Ts	*2 *6
チップセレクト信号ホールド時間(対 MWE#)	t_{MCEHWE}	1Ts – 1	-	1Ts	ns	*2
バイトイネーブル信号遅延時間	t_{MBED}	-	-	1	ns	-
バイトイネーブル信号ホールド時間	t_{MBEH}	1Ts – 2	-	-	ns	*2
書き込みデータセットアップ時間	t_{MDWS}	-	-	n_4Ts – 6	ns	*2 *7
書き込みデータホールド時間	t_{MDWH}	1Ts – 0.5	-	-	ns	*2

表 30.9　スタティックメモリタイミング 2 (MCS2#/MCS3#)

項目	記号	Min.	Typ.	Max.	単位	備考
アドレス信号セットアップ時間	t_{MAS}	– 1	-	1	ns	-
アドレス信号ホールド時間(対 MOE#)	t_{MAHOE}	– 0.5	-	0	ns	-
アドレス信号ホールド時間(対 MWE#)	t_{MAHWE}	1Ts – 1	-	-	ns	*2
アドレス信号ホールド時間(対 MCSx#)	t_{MAHCE}	1Ts – 1	-	-	ns	*2
データ出力許可信号遅延時間	t_{MOED}	n_1Ts – 3.5	-	-	ns	*2 *4
データ出力許可信号有効期間	t_{MOEV}	n_2Ts – 0.5	-	-	ns	*2 *5
データ出力許可信号インターバル期間	t_{MOEI}	-	n_1	-	Ts	*2 *4
チップセレクトホールド時間(対 MOE#)	t_{MCEHOE}	0	-	3	ns	-
読み込みデータセットアップ時間	t_{MDRS}	-	-	18	ns	-
読み込みデータホールド時間	t_{MDRH}	0	-	-	ns	-
書き込み有効信号遅延時間	t_{MWED}	n_3Ts – 3.5	-	-	ns	*2 *6
書き込み有効信号有効期間	t_{MWEV}	n_4Ts+1 -n_3Ts- 0.5	-	-	ns	*2 *6 *7
書き込み有効信号インターバル期間	t_{MWEI}	-	n_3	-	Ts	*2 *6
チップセレクト信号ホールド時間(対 MWE#)	t_{MCEHWE}	1Ts	-	1Ts + 2	ns	*2
バイトイネーブル信号遅延時間	t_{MBED}	-	-	1	ns	-
バイトイネーブル信号ホールド時間	t_{MBEH}	1Ts – 2	-	-	ns	*2
書き込みデータセットアップ時間	t_{MDWS}	-	-	n_4Ts – 6	ns	*2 *7
書き込みデータホールド時間	t_{MDWH}	1Ts – 0.5	-	-	ns	*2

* 2　T_S = システムクロックサイクル時間
* 4　n_1 = WAITOE レジスタ(MEMC:0x20,0x30,0x40,0x50)で設定した値
* 5　n_2 = WAITRD レジスタ(MEMC:0x20,0x30,0x40,0x50)で設定した値
* 6　n_3 = WAITWE レジスタ(MEMC:0x20,0x30,0x40,0x50)で設定した値
* 7　n_4 = WAITWR レジスタ(MEMC:0x20,0x30,0x40,0x50)で設定した値

■ SDRAM コントローラ タイミング

表 30.10　SDRAM コントローラ タイミング（SDVDD＝3.3V±0.3V）

項目	記号	Min.	Typ.	Max.	単位	備考
SDCKE 遅延時間	t_{CKED}	2	-	8	ns	-
SDCS[1:0]#遅延時間	t_{CSD}	2	-	10	ns	-
SDRAS#遅延時間	t_{RASD}	2	-	8	ns	-
SDCAS#遅延時間	t_{CASD}	2	-	10	ns	-
SDWE#遅延時間	t_{WED}	2	-	8	ns	-
SDDQM[3:0]遅延時間	t_{DQMD}	1	-	10	ns	-
アドレス遅延時間	t_{ADD}	1	-	10	ns	-
ライトデータ遅延時間	t_{WDD}	1	-	8	ns	-
リードデータセットアップ時間	t_{RDS}	12	-	-	ns	-
リードデータホールド時間	t_{RDH}	0	-	-	ns	-

表 30.11　SDRAM コントローラ タイミング（SDVDD＝3.0V±0.3V）

項目	記号	Min.	Typ.	Max.	単位	備考
SDCKE 遅延時間	t_{CKED}	(TBD)	-	(TBD)	ns	-
SDCS[1:0]#遅延時間	t_{CSD}	(TBD)	-	(TBD)	ns	-
SDRAS#遅延時間	t_{RASD}	(TBD)	-	(TBD)	ns	-
SDCAS#遅延時間	t_{CASD}	(TBD)	-	(TBD)	ns	-
SDWE#遅延時間	t_{WED}	(TBD)	-	(TBD)	ns	-
SDDQM[3:0]遅延時間	t_{DQMD}	(TBD)	-	(TBD)	ns	-
アドレス遅延時間	t_{ADD}	(TBD)	-	(TBD)	ns	-
ライトデータ遅延時間	t_{WDD}	(TBD)	-	(TBD)	ns	-
リードデータセットアップ時間	t_{RDS}	(TBD)	-	-	ns	-
リードデータホールド時間	t_{RDH}	(TBD)	-	-	ns	-

30.2.2.6 I^2C Single Master Core Module (I2C) タイミング

表 30.12　I2C タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
SCL サイクル時間	$t_{C(SCL)}$	8	-	30720	T_S	*2
SCL パルス幅（High）	$t_{WH(SCL)}$	-	1/2	-	$T_{C(SCL)}$	*8
SCL パルス幅（Low）	$t_{WL(SCL)}$	-	1/2	-	$T_{C(SCL)}$	*8
SDA 出力遅延時間	$t_{D(OSDA)}$	-	1/4	-	$T_{C(SCL)}$	*8
SDA 入力セットアップ時間	$t_{SU(ISDA)}$	0	-	-	ns	*9
SDA 入力ホールド時間	$t_{HD(ISDA)}$	0	-	-	ns	*9
SDA サンプル時間	$t_{SMP(SDA)}$	-	1/4	-	$T_{C(SCL)}$	*8
スタートコンディション開始時間	$t_{S(ST)}$	1/4	-	-	$T_{C(SCL)}$	*8
スタートコンディション完了時間	$t_{E(ST)}$	1/2	-	-	$T_{C(SCL)}$	*8
ストップコンディション開始時間	$t_{S(SP)}$	1/4	-	-	$T_{C(SCL)}$	*8
ストップコンディション完了時間	$t_{E(SP)}$	1/2	-	-	$T_{C(SCL)}$	*8

*2：　T_S = システムクロックサイクル時間
*8：　$T_{C(SCL)}$ = SCL（I^2C クロック）サイクル時間
*9：　SDA サンプル時間（$T_{smp(SDA)}$）参照

30.2.2.7 I^2S タイミング

表 30.13　I2S タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
SCK サイクル時間	t_{SCKCT}	2	-	512	T_S	*2
SCK パルス幅（High）	t_{SCKWH}	1	-	-	Ts	*2
SCK パルス幅（Low）	t_{SCKWL}	1	-	-	Ts	*2
SCK デューティ比	t_{SCKDT}	-	50	-	%	*10
WS サイクル時間	t_{WSCT}	32	-	256	t_{SCKCT}	*11
WS 出力遅延時間	t_{WSOD}	-1	-	1	Ts	*2
WS 入力セットアップ時間	t_{WSISU}	1	-	-	Ts	*2
WS 入力ホールド時間	t_{WSIHD}	1	-	-	Ts	*2
SD 出力遅延時間	t_{SDOD}	-1	-	1	Ts	*2
SD 入力セットアップ時間	t_{SDISU}	1	-	-	Ts	*2
SD 入力ホールド時間	t_{SDIHD}	1	-	-	Ts	*2

*2：　T_S = システムクロックサイクル時間
*10：　$t_{SCKDT} = t_{SCKWH} / (t_{SCKWH} + t_{SCKWL})$
*11：　t_{SCKCT} = SCK サイクル時間

30.2.2.8 シリアル周辺機器インタフェース (SPI) タイミング

表 30.14　SPI タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
SCLK サイクル時間	$t_{C(SCLK)}$	4	-	512	T_S	*2
SCLK パルス幅（前半）	$t_{WH1(SCLK)}$	-	1/2	-	$T_{C(SCLK)}$	*12
SCLK パルス幅（後半）	$t_{WH2(SCLK)}$	-	1/2	-	$T_{C(SCLK)}$	*12
SS 出力開始時間（自動制御時）	$t_{S(OSS)}$	3	-	-	T_S	*2
SS 出力完了時間（自動制御時）	$t_{E(OSS)}$	1	-	-	T_S	*2
SS 入力セットアップ時間	$t_{SU(ISS)}$	3	-	-	T_S	*2
SS 入力ホールド時間	$t_{HD(ISS)}$	1	-	-	T_S	*2
MISO 入力セットアップ時間	$t_{SU(MI)}$	30	-	-	ns	
MISO 入力ホールド時間	$t_{HD(MI)}$	0	-	-	ns	
MISO 出力遅延時間	$t_{D(SO)}$	-	-	30	ns	
MOSI 入力セットアップ時間	$t_{SU(SI)}$	10	-	-	ns	
MOSI 入力ホールド時間	$t_{HD(SI)}$	10	-	-	ns	
MOSI 出力遅延時間	$t_{D(MO)}$	-	-	0	ns	

*2：　T_S = システムクロックサイクル時間
*12：　$T_{C(SCLK)}$ = SCLK（SPI クロック）サイクル時間 = $(4 \times 2^{MCBR}) T_S$

30.2.2.9 コンパクトフラッシュインタフェース (CF) タイミング

■ CF Attribute Memory タイミング

表 30.15　CF Attribute Memory Read タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
リードサイクル時間	t_{ATRC}	-	20	-	T_S	*2
アドレスセットアップ時間	t_{ADSAR}	-	4	-	T_S	*2
アドレスホールド時間（MOE#インアクティブから）	t_{ADHMOE}	-	2	-	T_S	*2
リード前 CE 有効時間	t_{CEVBR}	-	3	-	T_S	*2
リード後 CE 有効時間	t_{CEVAR}	-	2	-	T_S	*2
MOE# アクティブ時間	t_{MOEW}	-	14	-	T_S	*2
リードデータセットアップ時間	t_{RDS}	$1T_S$+16	-	-	ns	
リードデータホールド時間	t_{RDH}	0	-	-	ns	

*2：　T_S = システムクロックサイクル時間

表 30.16　CF Attribute Memory Write タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
ライトサイクル時間	t_{ATWC}	-	16	-	T_S	*2
アドレスセットアップ時間	t_{ADSAW}	-	3	-	T_S	*2
MWE0#アクティブ時間	t_{MWE0W}	-	9	-	T_S	*2
ライトリカバリー時間	t_{WREC}	-	2	-	T_S	*2
ライトデータ有効時間 1	t_{WDV1}	-	11	-	T_S	*2
ライトデータ有効時間 2	t_{WDV2}	-	2	-	T_S	*2

■ CF Common Memory タイミング

表 30.17　CF Common Memory Read タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
リードサイクル時間	t_{CMRC}	-	17	-	T_S	*2
アドレスセットアップ時間	t_{CRADS}	-	4	-	T_S	*2
アドレスホールド時間（MOE#インアクティブから）	t_{ADHMOE}	-	2	-	T_S	*2
リード前 CE 有効時間	t_{CEVBR}	-	3	-	T_S	*2
リード後 CE 有効時間	t_{CEVAR}	-	2	-	T_S	*2
リード後ウェイトアクティブ許容時間	t_{WTATAR}	-	-	6	T_S	*2
ウェイトリリース後データセットアップ時間	t_{DSAWT}	-	-	0	T_S	*2
ウェイトアクティブ時間	t_{WTW}			3000	ns	
リードデータセットアップ時間	t_{RDS}	$1T_S$+16	-	-	ns	
リードデータホールド時間	t_{RDH}	0	-	-	ns	

表 30.18　CF Common Memory Write タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
ライトサイクル時間	t_{CMWC}	-	17	-	T_S	*2
アドレスセットアップ時間	t_{ADS}	-	4	-	T_S	*2
アドレスホールド時間	t_{ADH}	-	4	-	T_S	*2
ライト前 CE 有効時間	t_{CEVBW}	-	3	-	T_S	*2
ライト後 CE 有効時間	t_{CEVAW}	-	2	-	T_S	*2
MWE0# アクティブ時間	t_{MWE0W}	-	9	-	T_S	*2
ライト前データ有効時間	t_{DVBW}	-	11	-	T_S	*2
ライト後データ有効時間	t_{DVAW}	-	2	-	T_S	*2
ライトリカバリー時間	t_{WREC}	-	2	-	T_S	*2
ライト後ウェイトアクティブ許容時間	t_{WTATAW}	-	-	6	T_S	*2
ウェイトリリース後ライトアクティブ時間	t_{WWAWT}	-	-	3	T_S	*2
ウェイトアクティブ時間	t_{WTW}	-	-	3000	ns	

■ CF I/O Space/IDE タイミング

表 30.19　CF I/O Space/IDE Read タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
リードサイクル時間	t_{IORC}	-	20	-	T_S	*2
IORD#アクティブ時間	t_{IORW}	-	10	-	T_S	*2
アドレスセットアップ時間	t_{ADSIO}	-	6	-	T_S	*2
アドレスホールド時間	t_{ADHIO}	-	4	-	T_S	*2
IO リード前 CE 有効時間	$t_{CEVBIOR}$	-	5	-	T_S	*2
IO リード後 CE 有効時間	$t_{CEVAIOR}$	-	3	-	T_S	*2
IO リード前 REG 有効時間	$t_{REGVBIOR}$	-	6	-	T_S	*2
IO リード後 REG 有効時間	$t_{REGVAIOR}$	-	4	-	T_S	*2
IO リードアクティブ後ウェイト許容時間	$t_{WTATIOR}$	-	-	6	T_S	*2
ウェイトリリース後データ遅延許容時間	t_{DATAWT}	-	-	0	T_S	*2
ウェイトアクティブ時間	t_{WTW}	-	-	3000	ns	
リードデータセットアップ時間	t_{RDS}	$1T_S$+16	-		ns	
リードデータホールド時間	t_{RDH}	0			ns	

表 30.20　CF I/O Space/IDE Write タイミング

項目	記号	Min.	Typ.	Max.	単位	備考
ライトサイクル時間	t_{IOWC}	-	20	-	T_S	*2
IOWR# アクティブ時間	t_{IOWW}	-	10	-	T_S	*2
アドレスセットアップ時間	t_{ADSIO}	-	6	-	T_S	*2
アドレスホールド時間	t_{ADHIO}	-	4	-	T_S	*2
IO ライト前 CE 有効時間	$t_{CEVBIOW}$	-	5	-	T_S	*2
IO ライト後 CE 有効時間	$t_{CEVAIOW}$	-	3	-	T_S	*2
IO ライト前 REG 有効時間	$t_{REGVBIOW}$	-	6	-	T_S	*2
IO ライト後 REG 有効時間	$t_{REGVAIOW}$	-	4	-	T_S	*2
IO ライト前データ有効時間	t_{DVBIOW}	-	14	-	T_S	*2
IO ライト後データ有効時間	t_{DVAIOW}	-	3	-	T_S	*2
IO ライト後ウェイト許容時間	$t_{WTATIOW}$	-	-	6	T_S	*2
ウェイトリリース後 IO ライトインアクティブ時間	t_{WITAWT}	-	-	2	T_S	*2
ウェイトアクティブ時間	t_{WTW}	-	-	3000	ns	

30.2.3 タイミングチャート

30.2.3.1 クロックタイミング

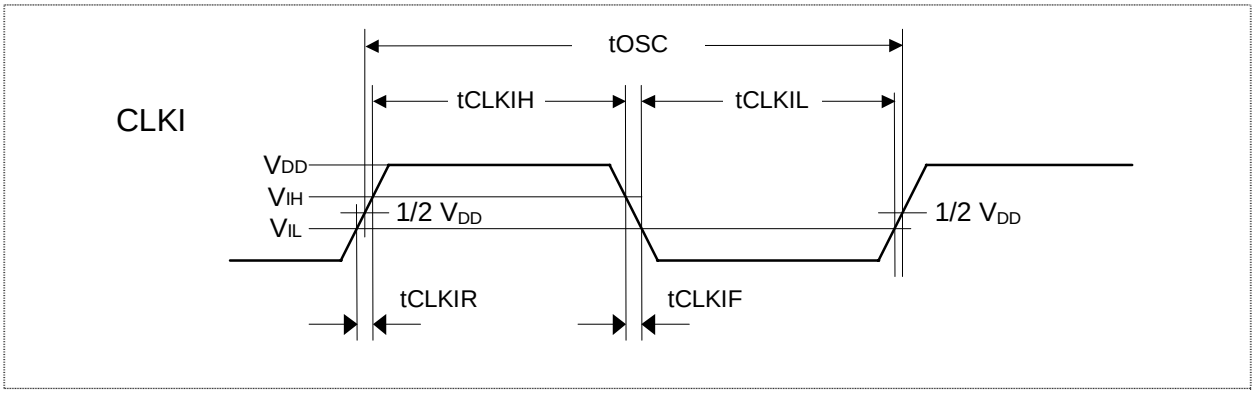


図 30.1　クロックタイミング

30.2.3.2 CPU コントロール信号タイミング

■ RESET#タイミング

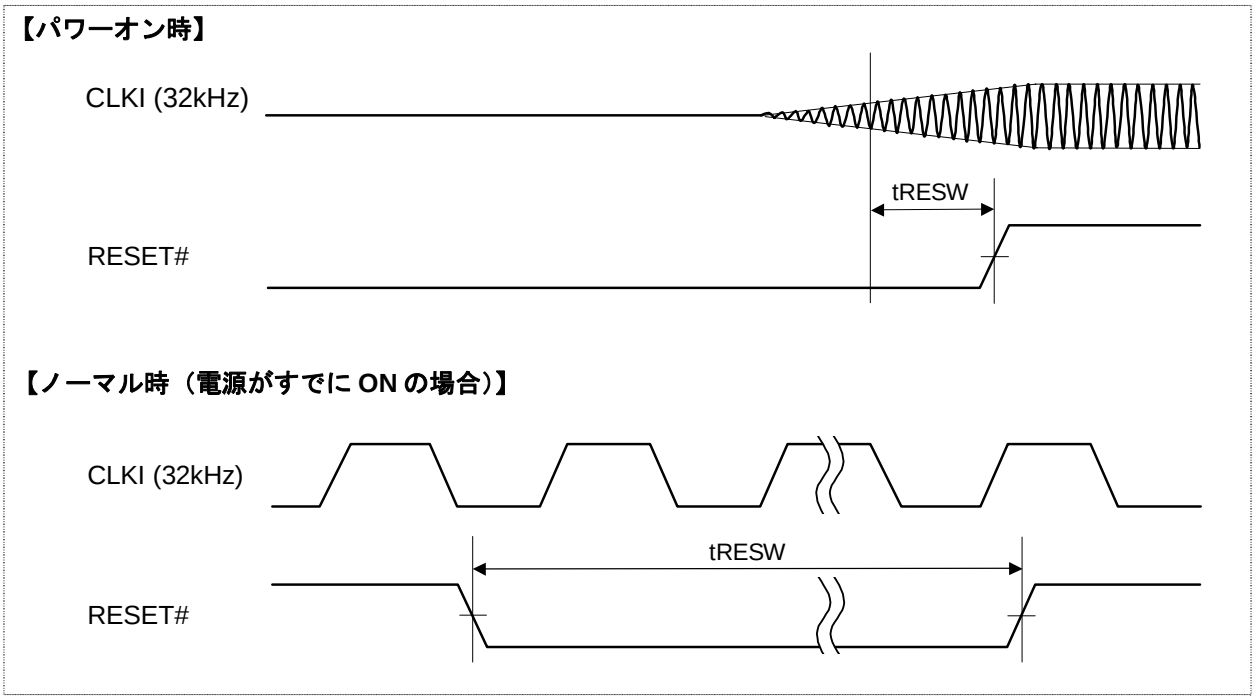


図 30.2　RESET#タイミング

■ 割り込み信号タイミング

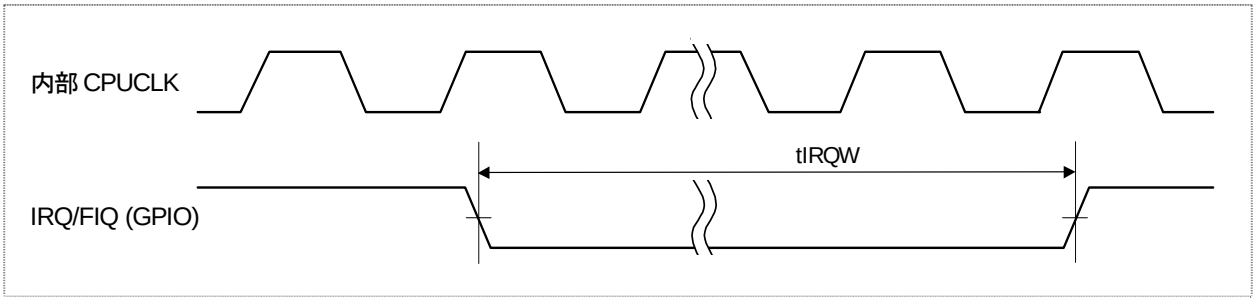
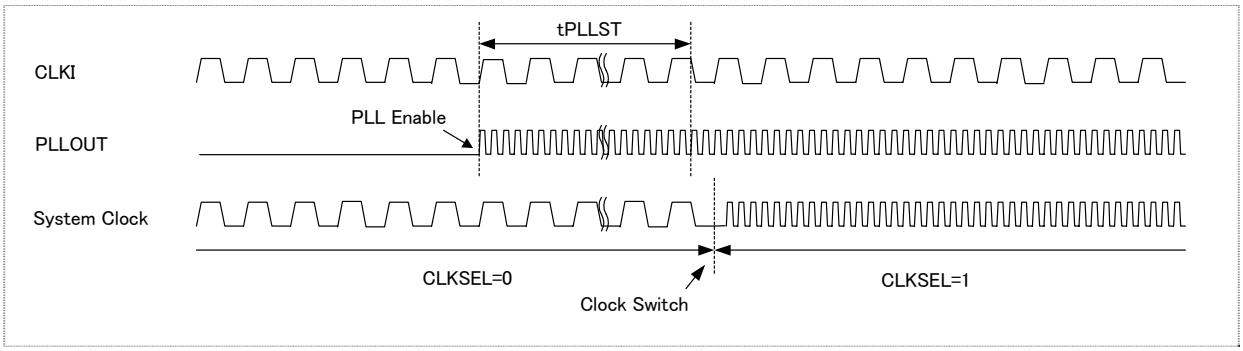


図 30.3　割り込み信号タイミング

■ PLL 関連タイミング

(1) クロックスィッチ 1（PLL Enable）

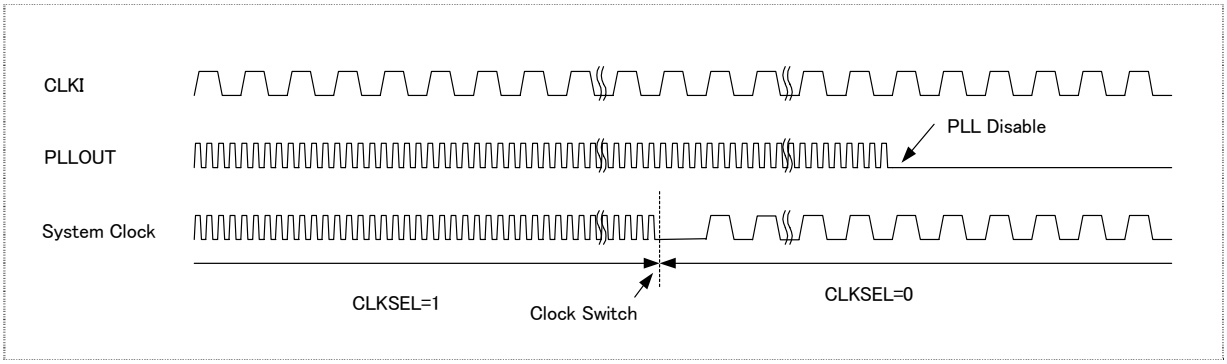


System Clock = CPUCLK / HCLK1 / HCLK2 / PCLK

図 30.4 クロックスィッチ 1（PLL Enable）

システムクロックを 32KHz から PLL 出力に変更する場合の内部クロックのタイミングです。
システムコントローラの PLL Setting Register 2 (SYS[0x0C]) ビット 0（PLLEN）を“1”にして、PLL をイネーブルにします。PLL 安定時間（t_{PLLST}）後に Clock Select Register (SYS[0x18]) ビット 0（CLKSEL）を“1”にして、システムクロックとして PLL 出力を選択します。

(2) クロックスィッチ 2（PLL Disable）

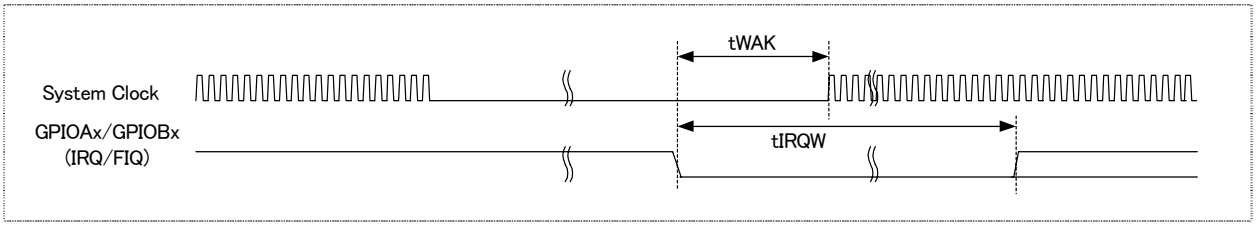


System Clock = CPUCLK / HCLK1 / HCLK2 / PCLK

図 30.5 クロックスィッチ 2（PLL Disable）

システムクロックを PLL 出力から 32KHz に変更する場合の内部クロックのタイミングです。
Clock Select Register (SYS[0x18]) ビット 0（CLKSEL）を“0”にして、システムクロックとして CLKI（32KHz）を選択します。その後、システムコントローラの PLL Setting Register 2 (SYS[0x0C]) ビット 0（PLLEN）を“0”にして、PLL をディセーブルにします。

(3) ハイスピード HALT モードから割り込みによるクロック再開

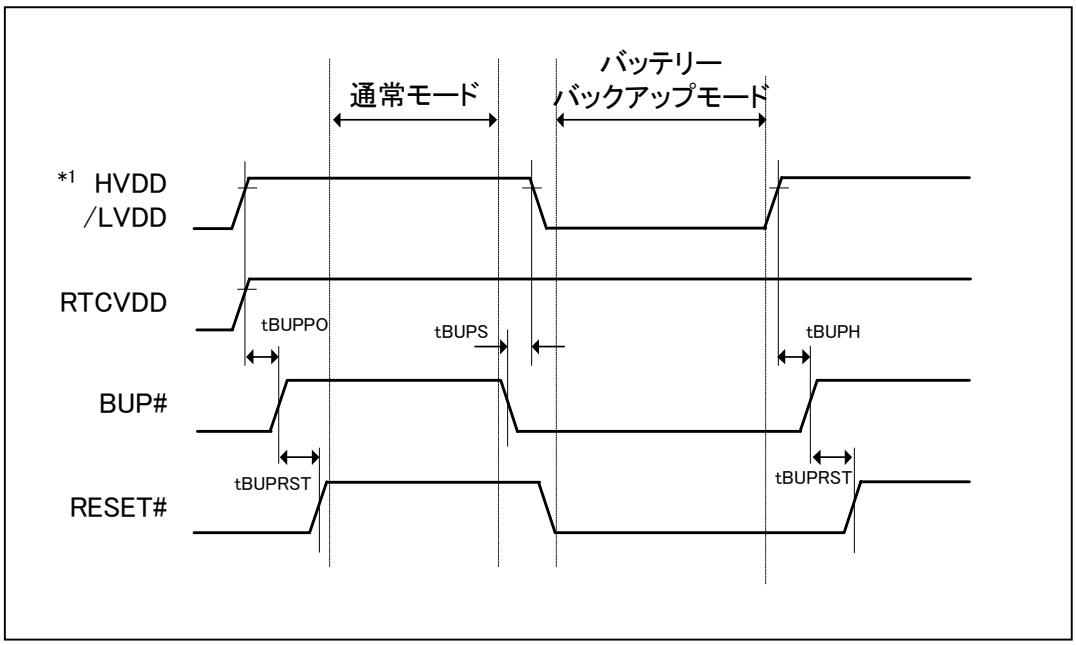


System Clock = CPUCLK / HCLK1 / HCLK2 / PCLK

図 30.6 クロック再開タイミング

ハイスピード HALT モードから割り込み発生により、ハイスピードモードへ移行するタイミングです。クロック再開時間（t_{WAK}）後にシステムクロックが再開します。割り込みパルス幅（t_{IRQW}）はそれよりも十分に長くとってください。

30.2.3.3 バッテリーバックアップモードタイミング



*1： C1VDD,C2VDD,SDVDD,AVDD,ULDD3,UPVDD,UXVDD,PLLVDD を含む

30.2.3.4 カメラインタフェースタイミング

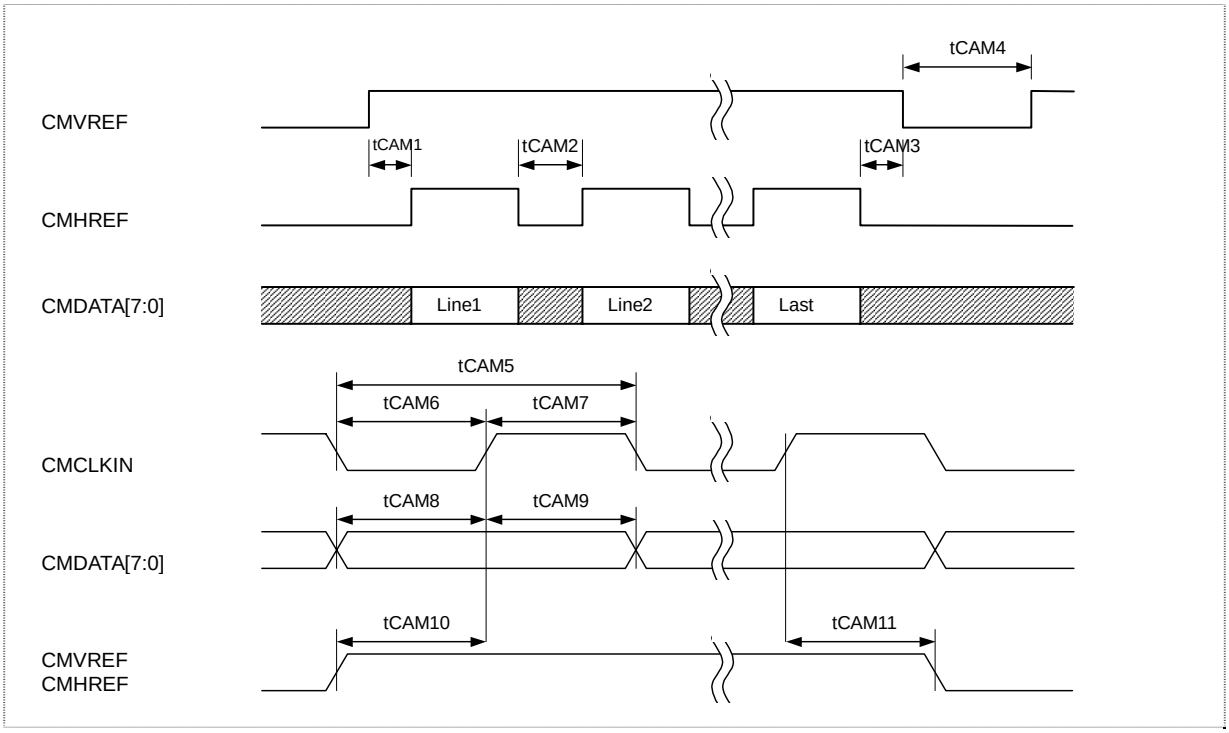


図 30.7　カメラインタフェースタイミング

CMCLKINの有効エッジはソフトウェアにより変更できます。上図はCMCLKINが“Low”から“High”に変化したときにデータを取り込むタイミングを示しています。

30.2.3.5 メモリインタフェースコントローラ

スタティックメモリコントローラタイミング（Flash EEPROM, SRAM, etc.）

■ スタティックメモリ リード タイミング

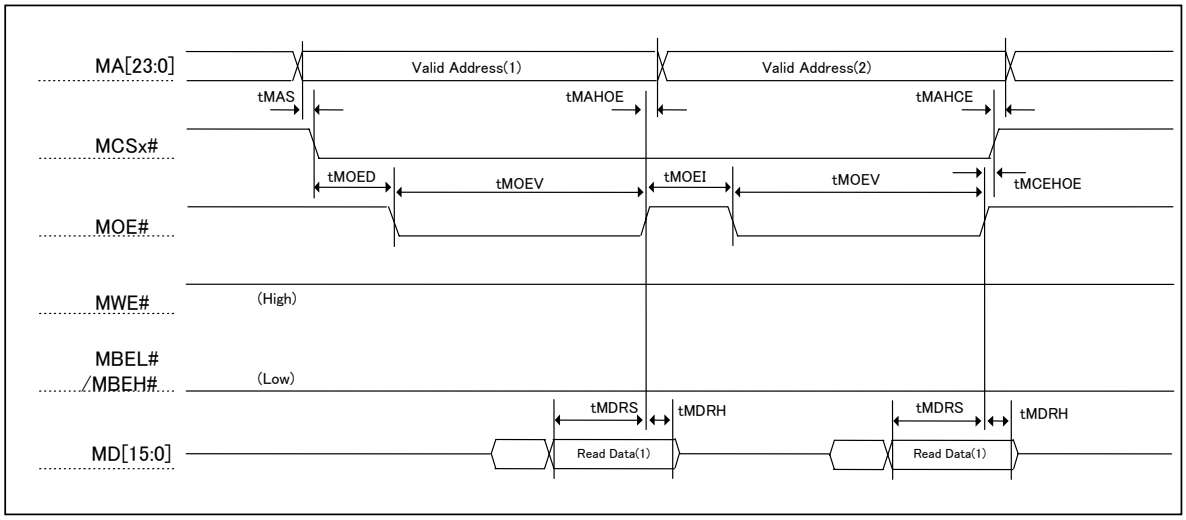


図 30.8　スタティックメモリ リードタイミング

■ スタティックメモリ ライト タイミング

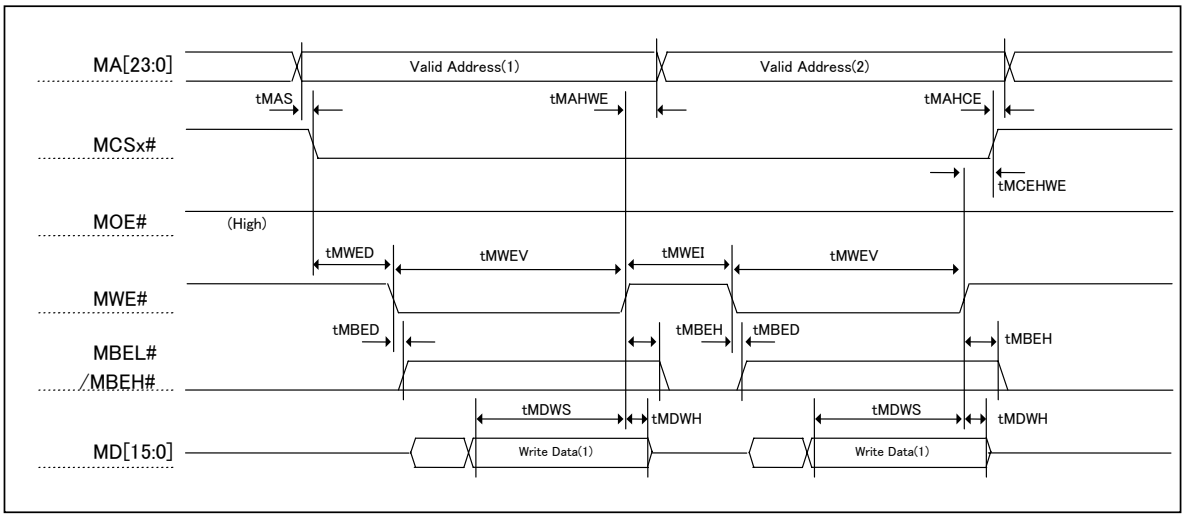


図 30.9　スタティックメモリ ライトタイミング

SDRAM コントローラ AC タイミング

以下に SDRAM コントローラの AC タイミングを示します。
図中に表記されているコマンドについて以下の表にまとめて説明します。

コマンド	機能	MCS2#	MRAS#	MCAS#	MWE1#	アドレス、その他
ACT	バンクアクティブ	L	L	H	H	Bank/Row
RD	リード	L	H	L	H	Bank/Col
WR	ライト	L	H	L	L	Bank/Col
BT	バーストターミネート	L	H	H	L	-
PCGA	プリチャージオールバンク	L	L	H	L	SDA10 = HIGH
PCG	プリチャージ	L	L	H	L	SDA10 = LOW
AREF	オートリフレッシュ	L	L	L	H	SDCLKEN = HIGH
SELF_IN	セルフリフレッシュ開始	L	L	L	H	SDCLKEN = LOW
SELF_OUT	セルフリフレッシュ終了	H	x	x	x	SDCLKEN = HIGH
LMR	モードレジスタセット	L	L	L	L	-

■ SDRAM リードサイクル

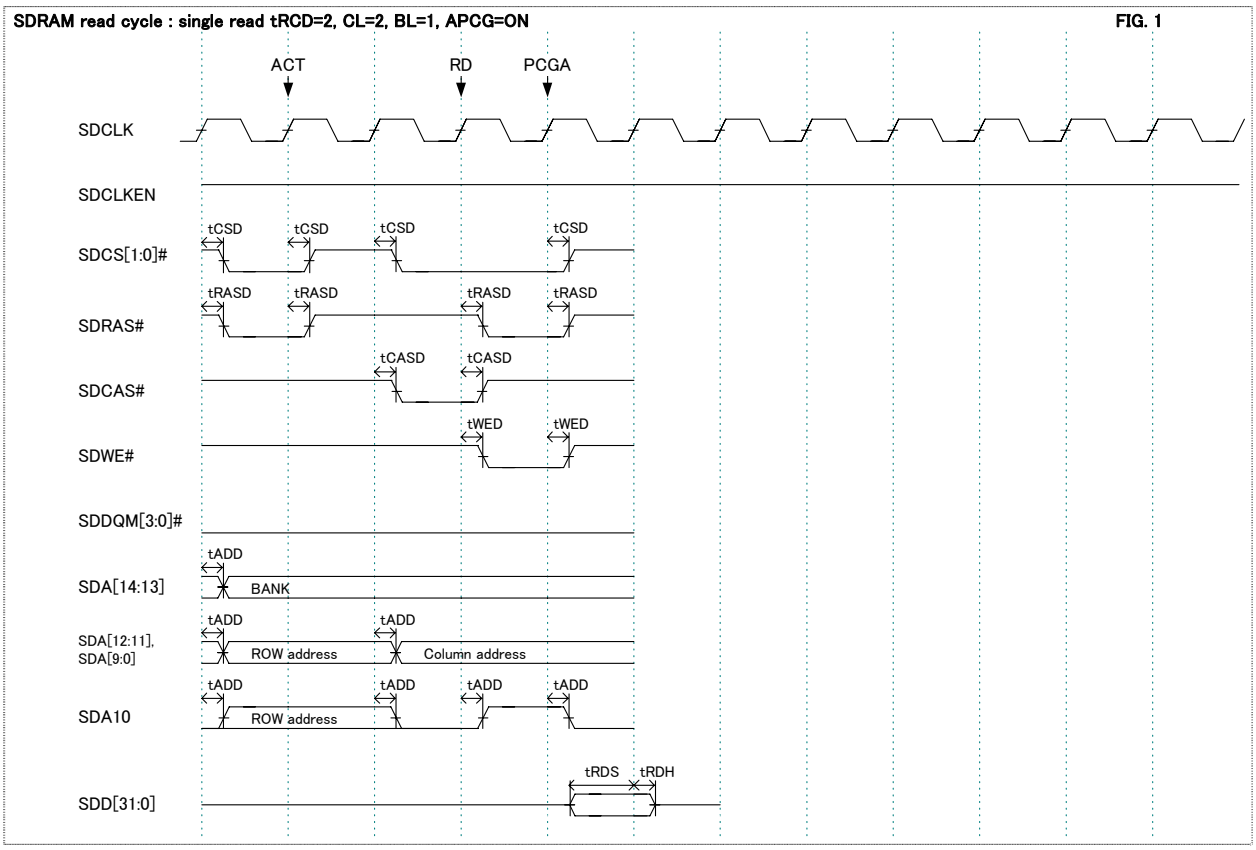


図 30.10　SDRAM リードサイクル 1：シングルリード；tRCD=2, CL=2, BL=1, APCG=ON

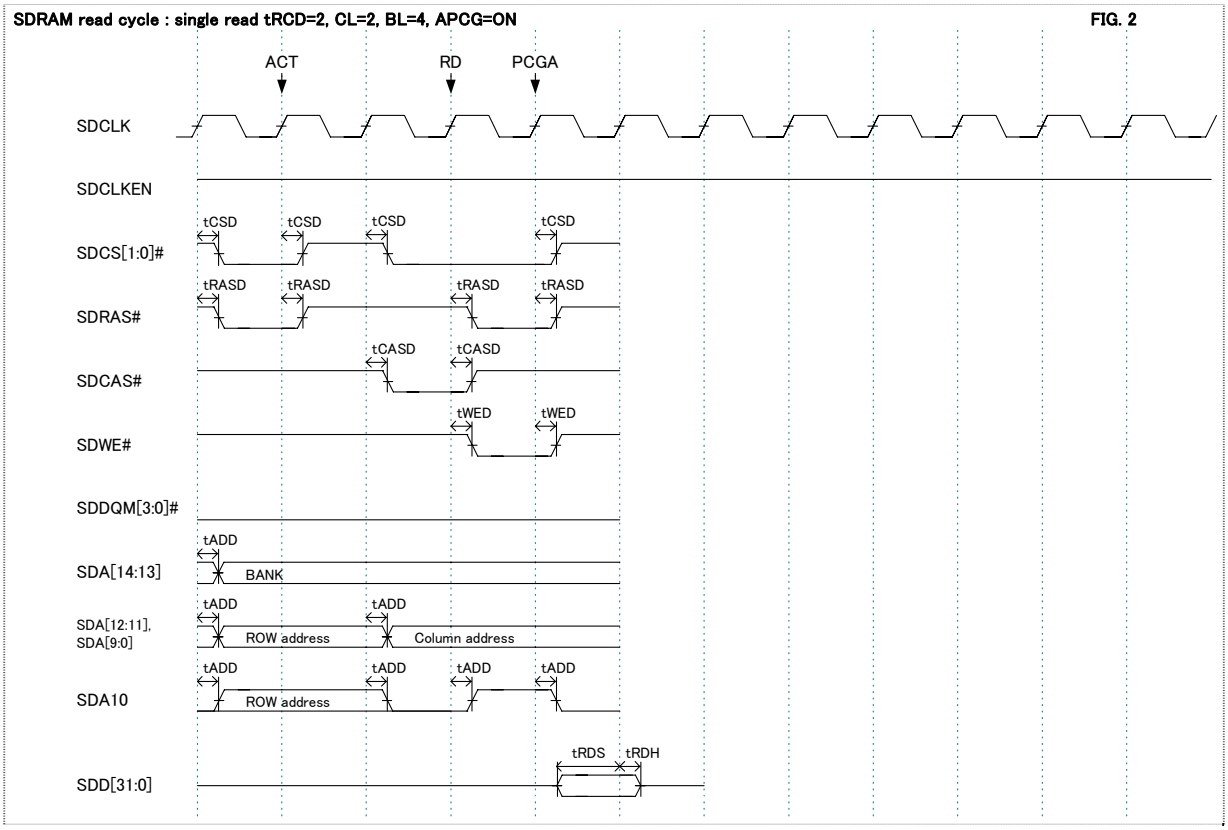


図 30.11　SDRAM リードサイクル 2：シングルリード；tRCD=2, CL=2, BL=4, APCG=ON

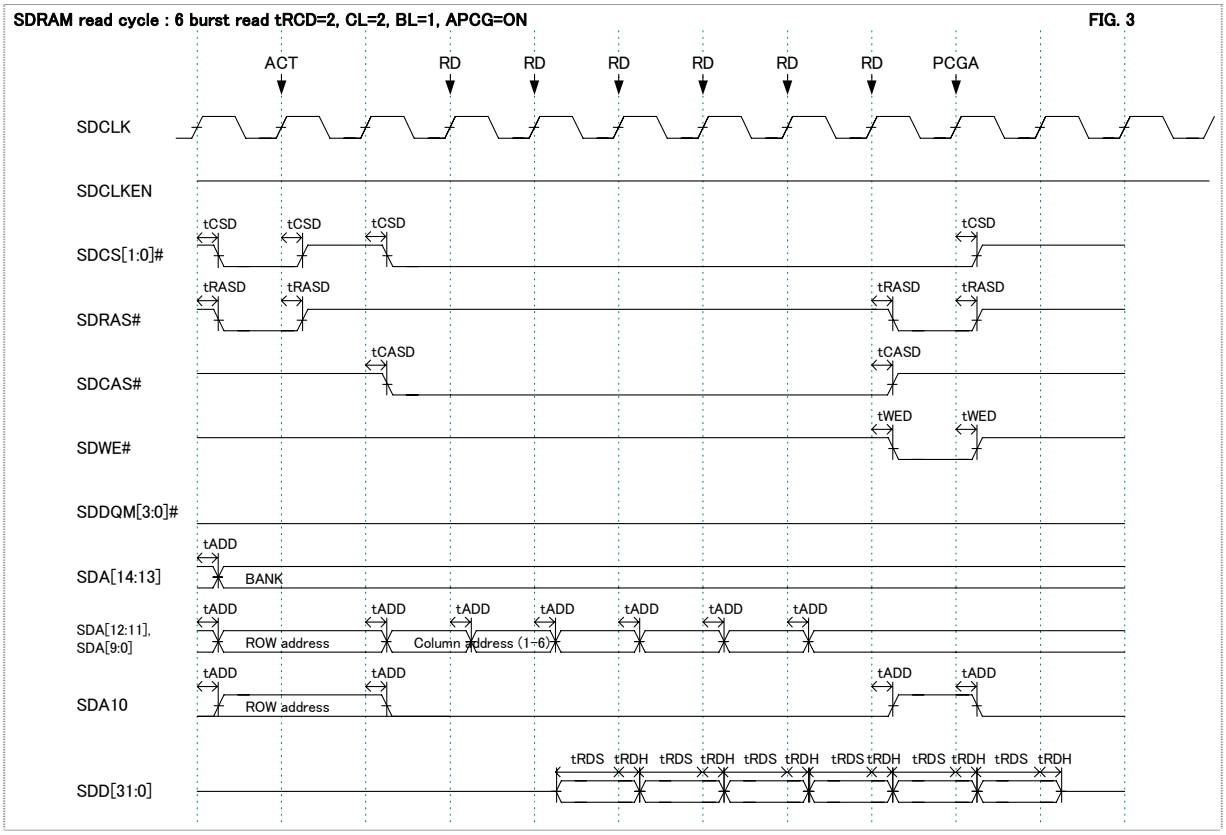


図 30.12　SDRAM リードサイクル 3：6 バーストリード；tRCD=2, CL=2, BL=1, APCG=ON

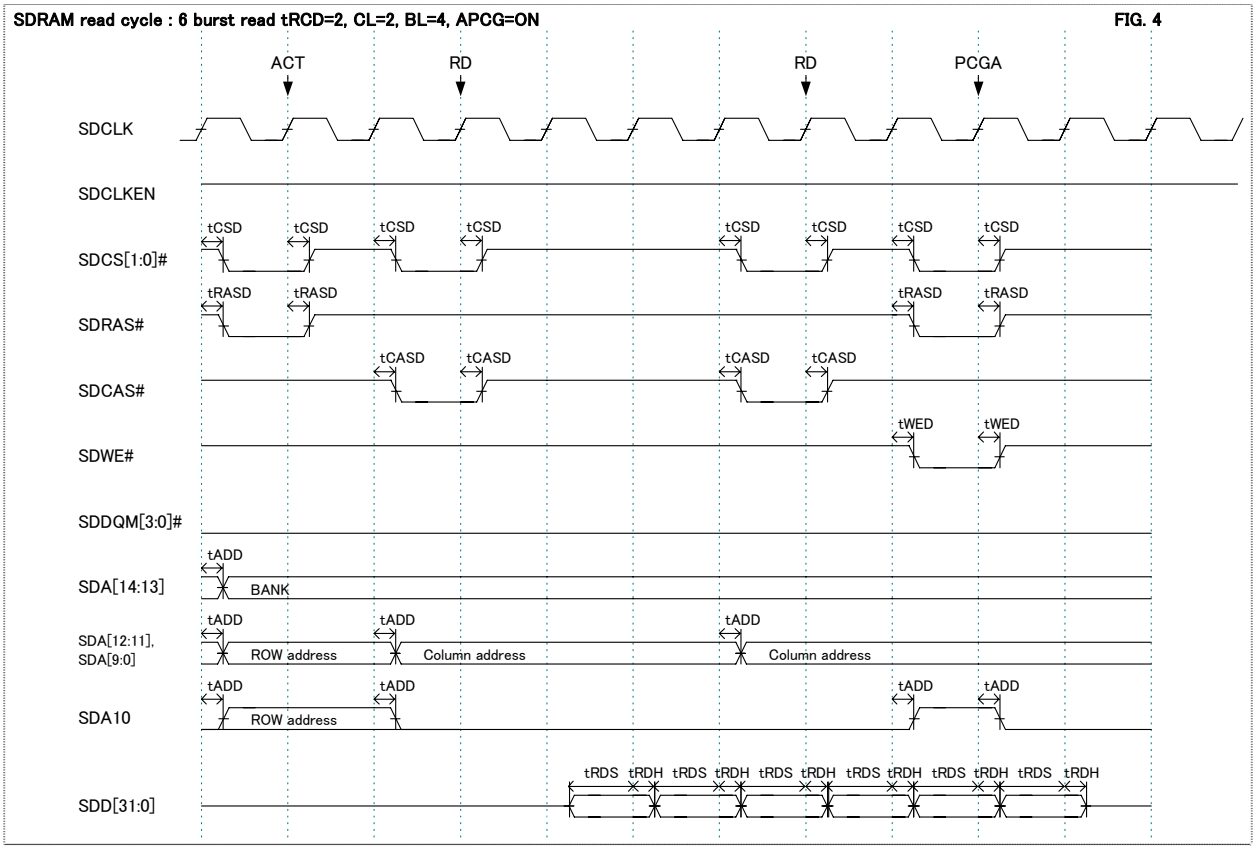


図 30.13　SDRAM リードサイクル 4：6 バーストリード；tRCD=2, CL=2, BL=4, APCG=ON

■ SDRAM ライトサイクル

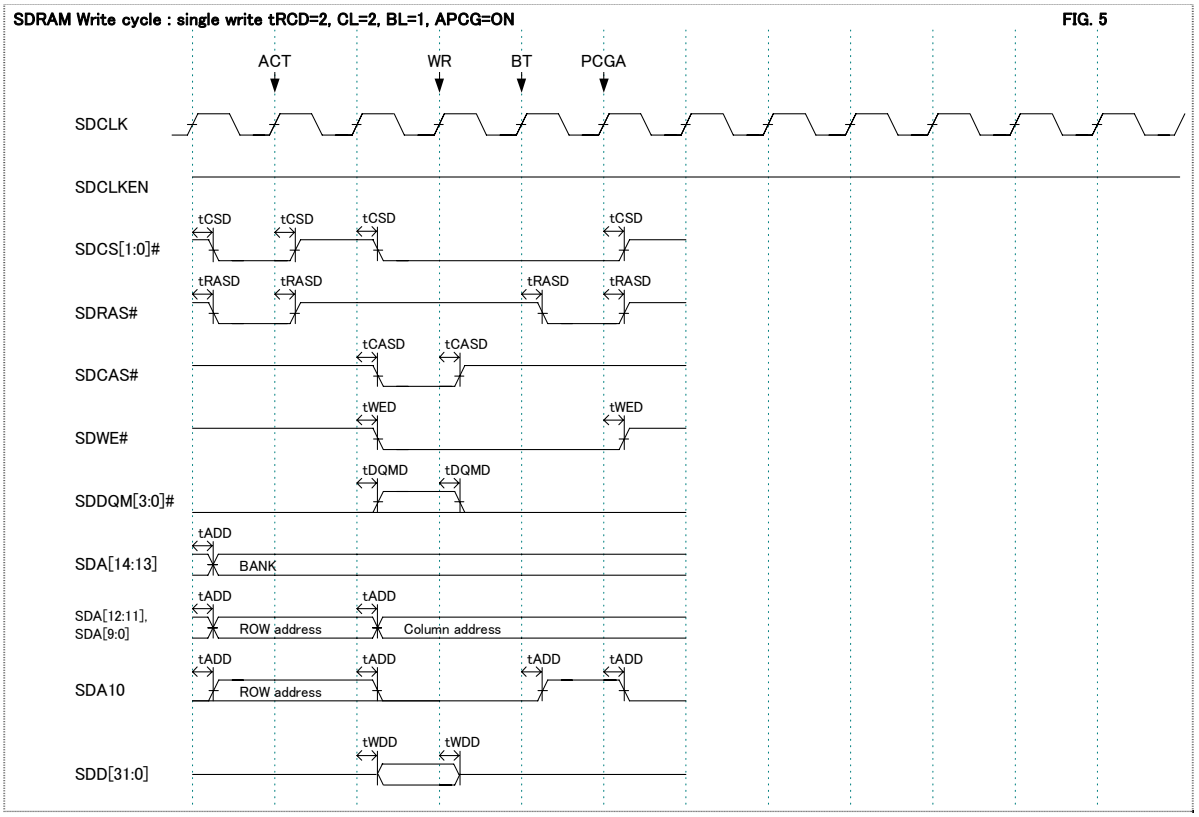


図 30.14　SDRAM ライトサイクル 1：シングルライト；tRCD=2, CL=2, BL=1, APCG=ON

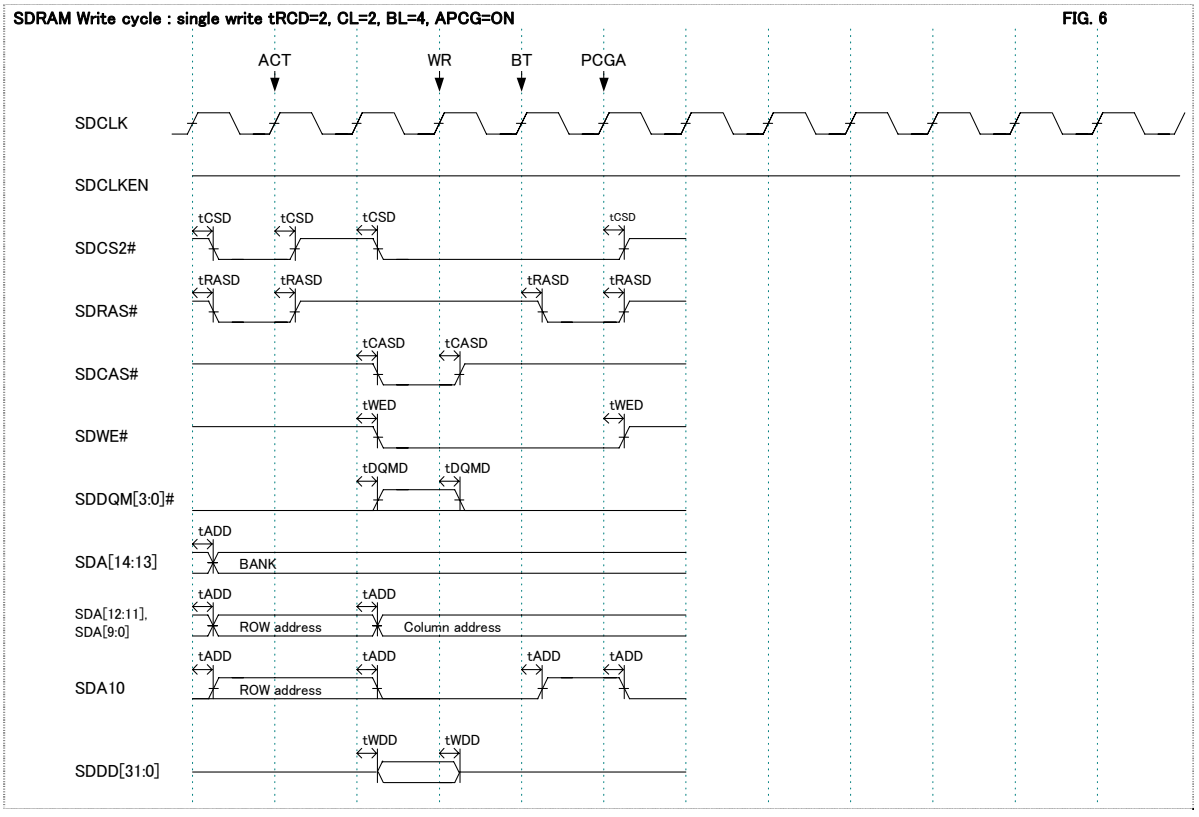


図 30.15　SDRAM ライトサイクル 2 シングルライト；tRCD=2, CL=2, BL=4, APCG=ON

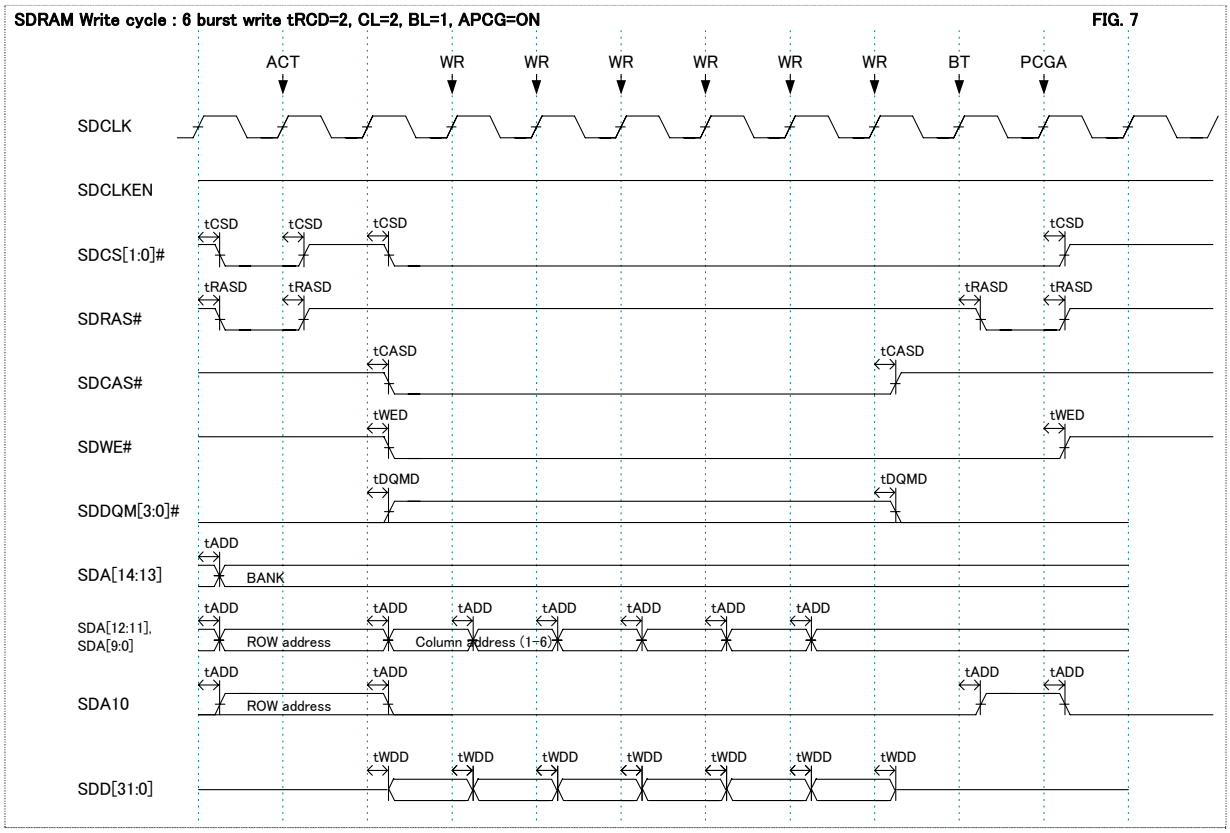


図 30.16　SDRAM ライトサイクル 3：6 バーストライト；tRCD=2, CL=2, BL=1, APCG=ON

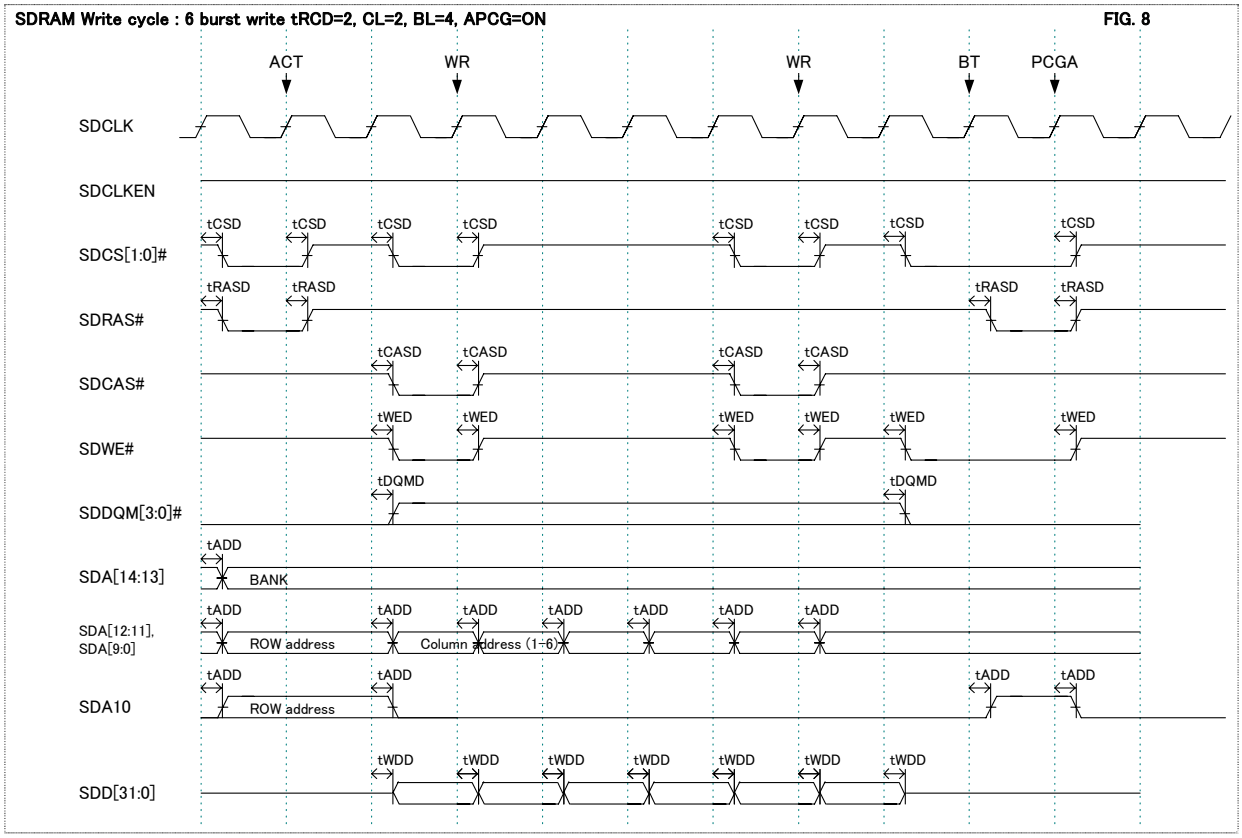


図 30.17　SDRAM ライトサイクル 4：6 バーストライト；tRCD=2, CL=2, BL=4, APCG=ON

■ SDRAM リードサイクル（Row アクティブモード）

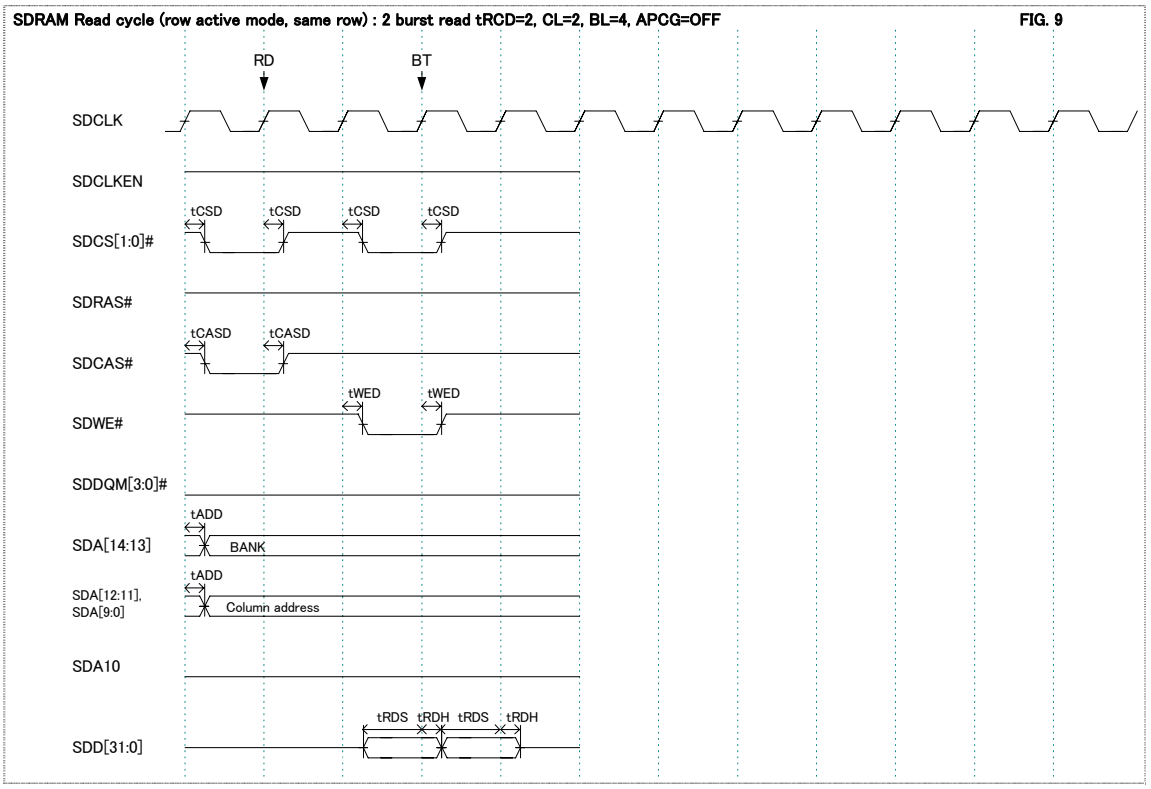


図 30.18　SDRAM リードサイクル Row アクティブモード 1 (same row)：
2 バーストリード；tRCD=2, CL=2, BL=4, APCG=OFF

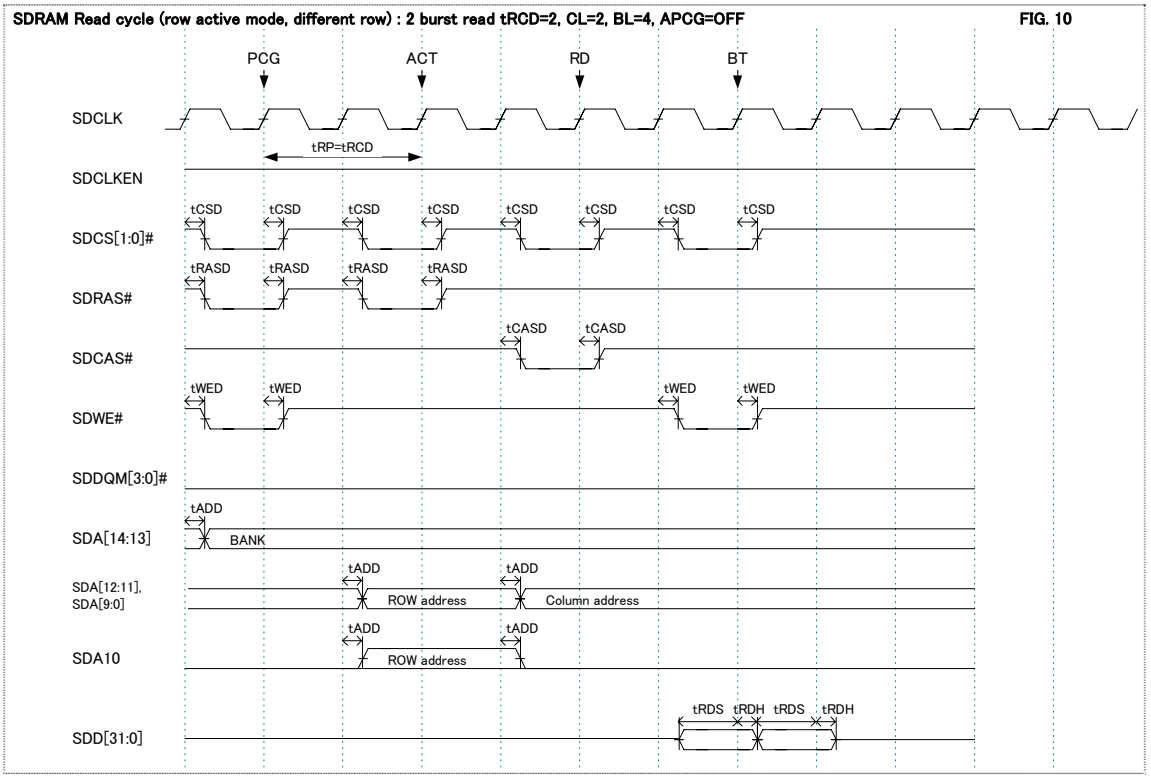


図 30.19　SDRAM リードサイクル Row アクティブモード 2 (differrent row)：
2 バーストリード；tRCD=2, CL=2, BL=4, APCG=OFF

■ SDRAM ライトサイクル（Row アクティブモード）

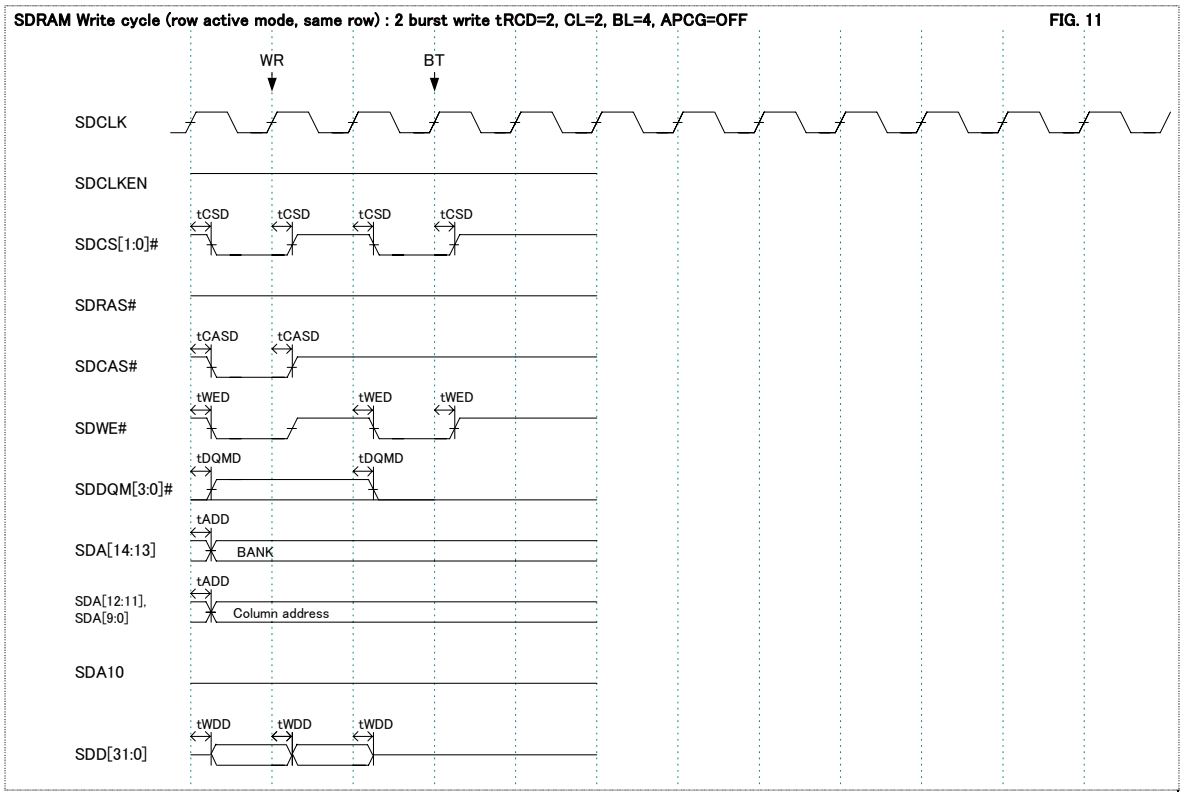


図 30.20　SDRAM ライトサイクル Row アクティブモード 1 (same row) :
2 バーストライト；tRCD=2, CL=2, BL=4, APCG=OFF

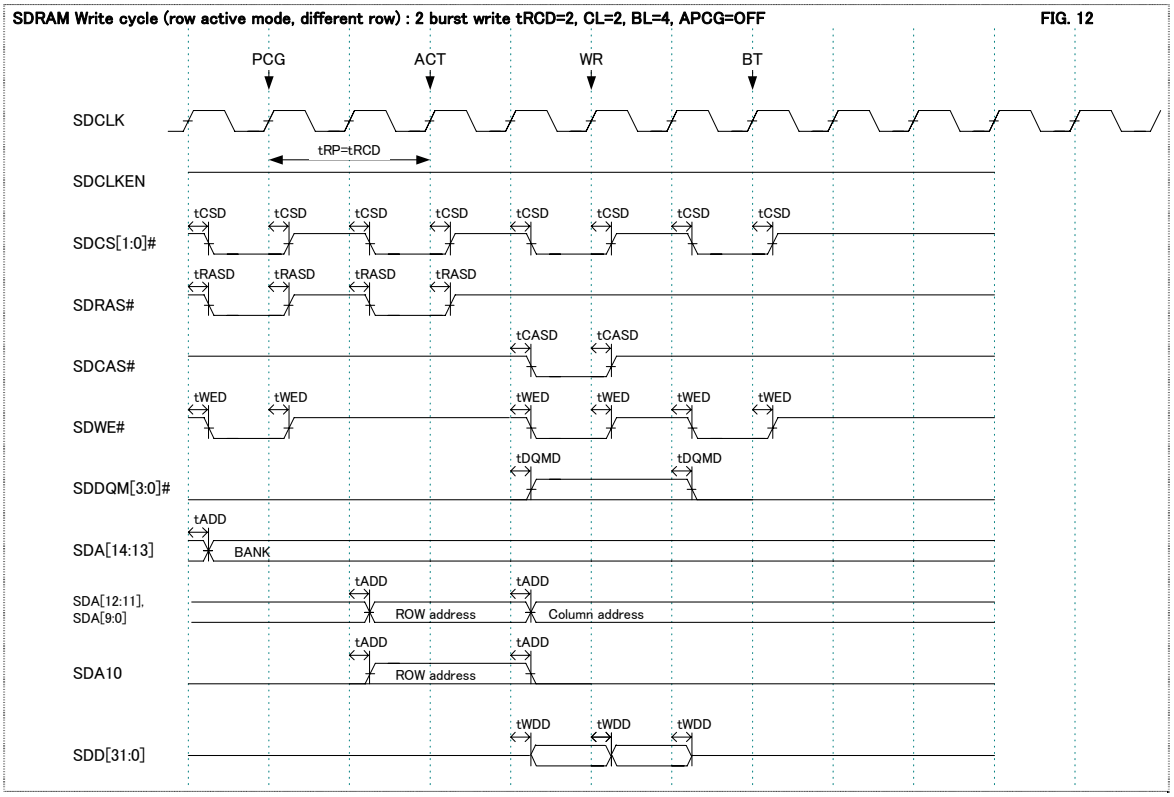


図 30.21　SDRAM ライトサイクル Row アクティブモード 2 (differrent row) :
2 バーストライト；tRCD=2, CL=2, BL=4, APCG=OFF

■ SDRAM オートリフレッシュサイクル

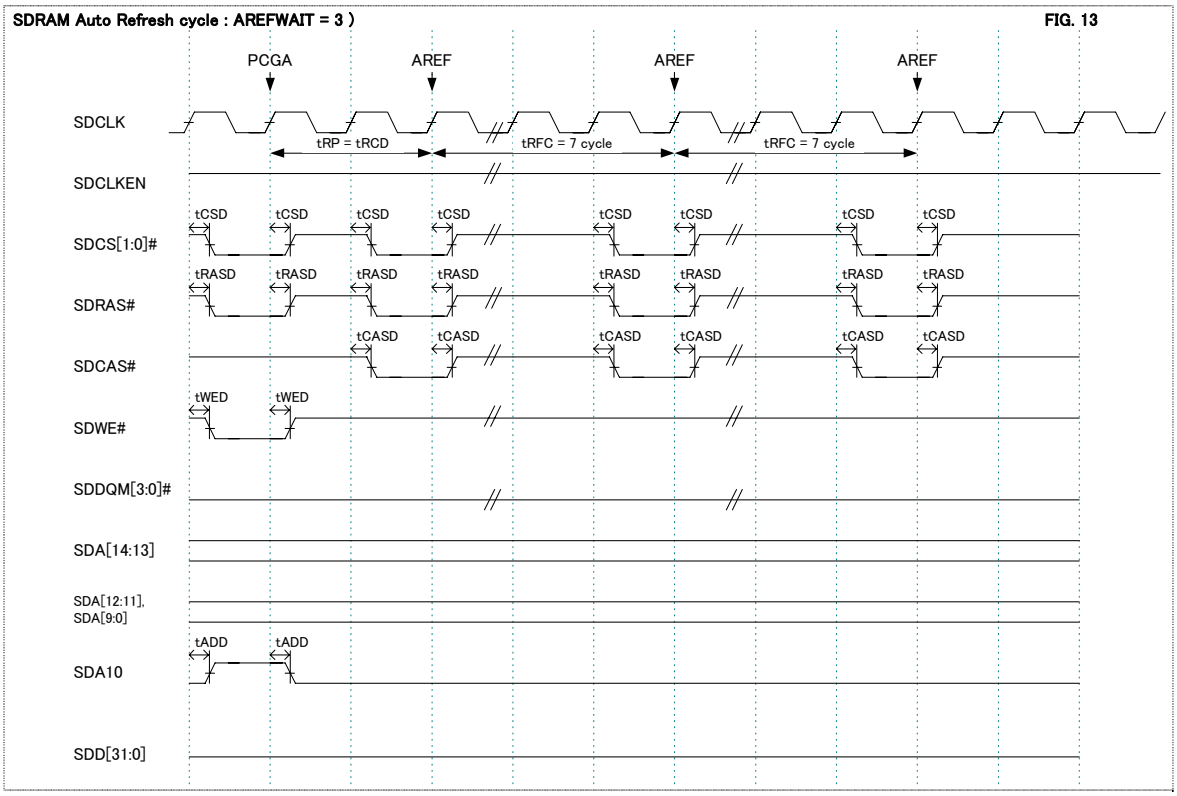


図 30.22　SDRAM オートリフレッシュサイクル：AREFWAIT=3

■ SDRAM セルフリフレッシュサイクル

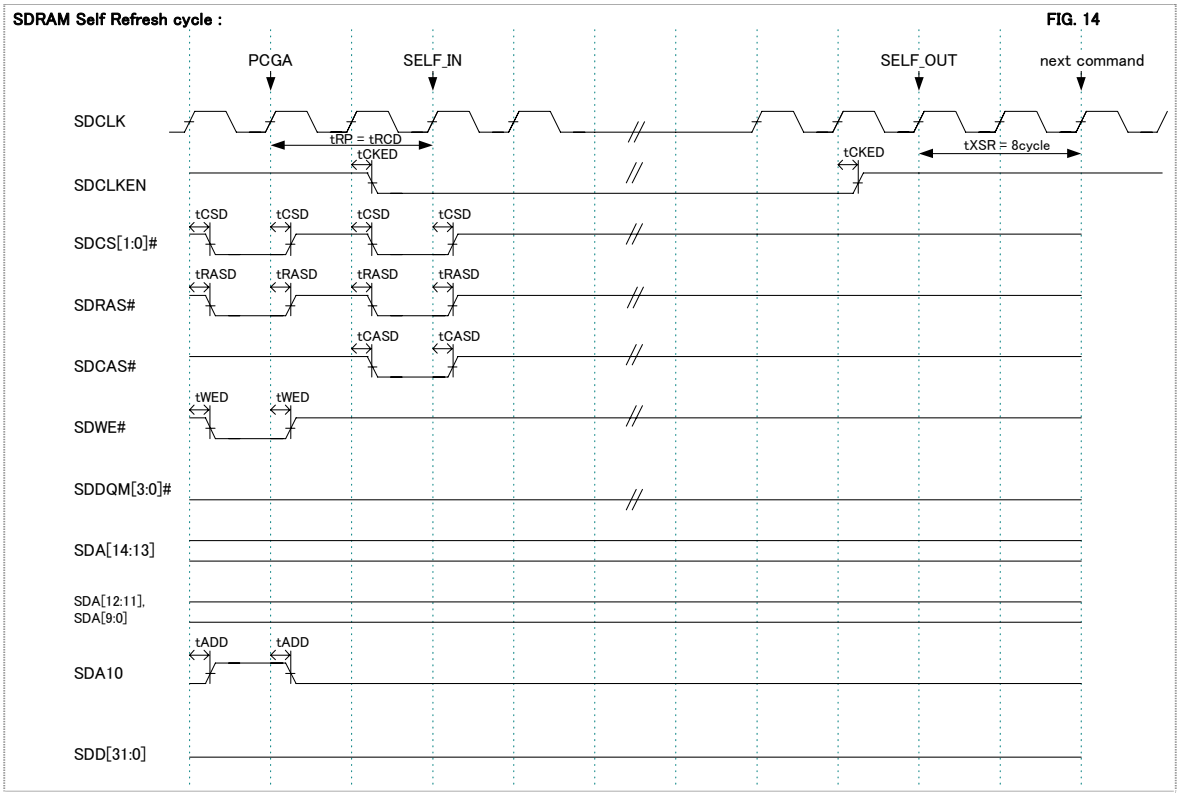


図 30.23　SDRAM セルフフレッシュサイクル

■ SDRAM 初期化サイクル

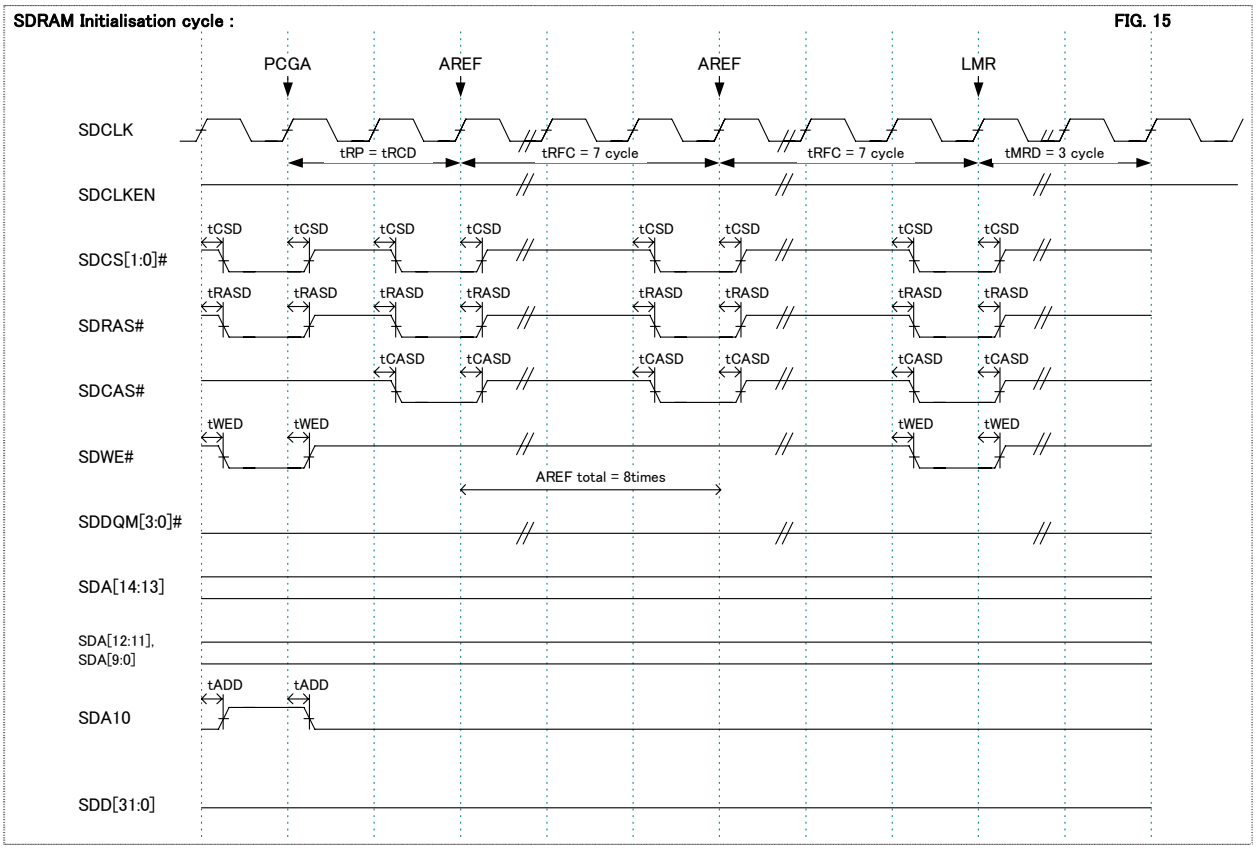


図 30.24　SDRAM 初期化サイクル

■ SDCLK/SDCLKEN コントロール

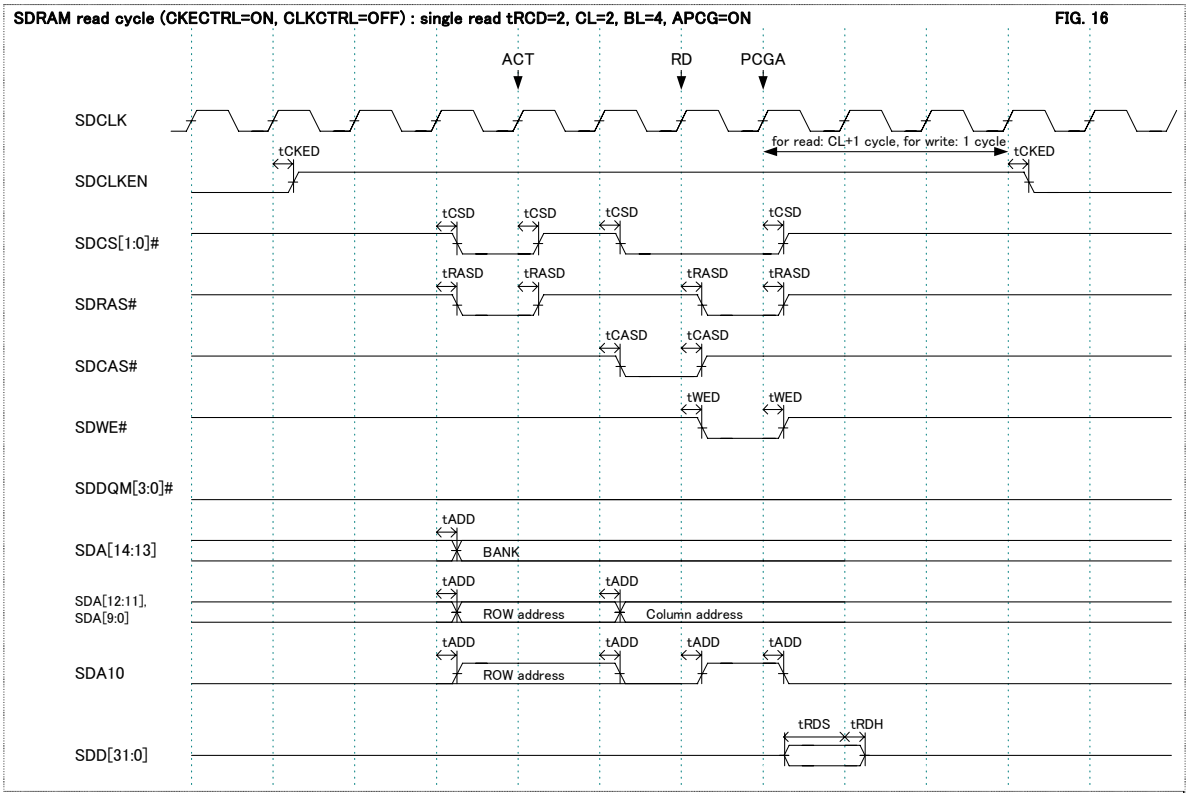


図 30.25　SDCLK/SDCLKEN コントロール 1（CKECTRL=ON, CLKCTRL=OFF）：
シングルリード；tRCD=2, CL=2, BL=4, APCG=ON

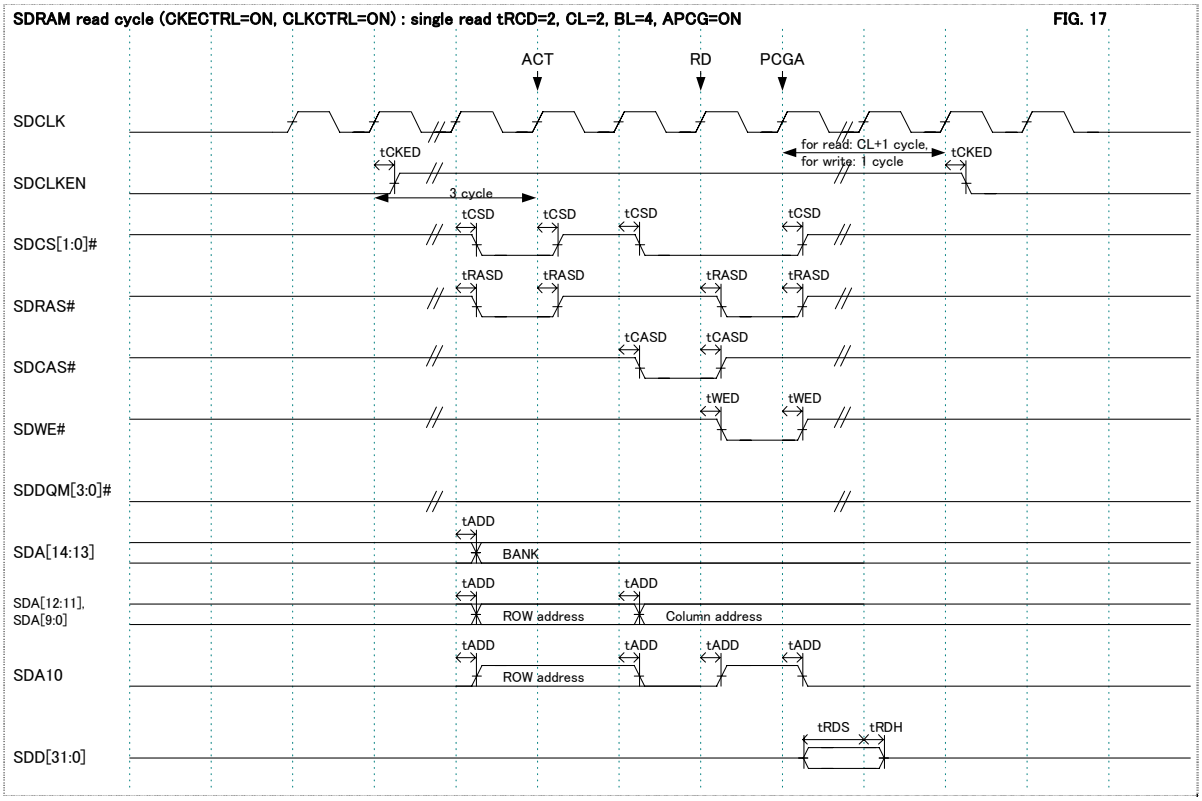


図 30.26　SDCLK/SDCLKEN コントロール 2（CKECTRL=ON, CLKCTRL=ON）：
シングルリード；tRCD=2, CL=2, BL=4, APCG=ON

30.2.3.6 I2C Single Master Core Module タイミング

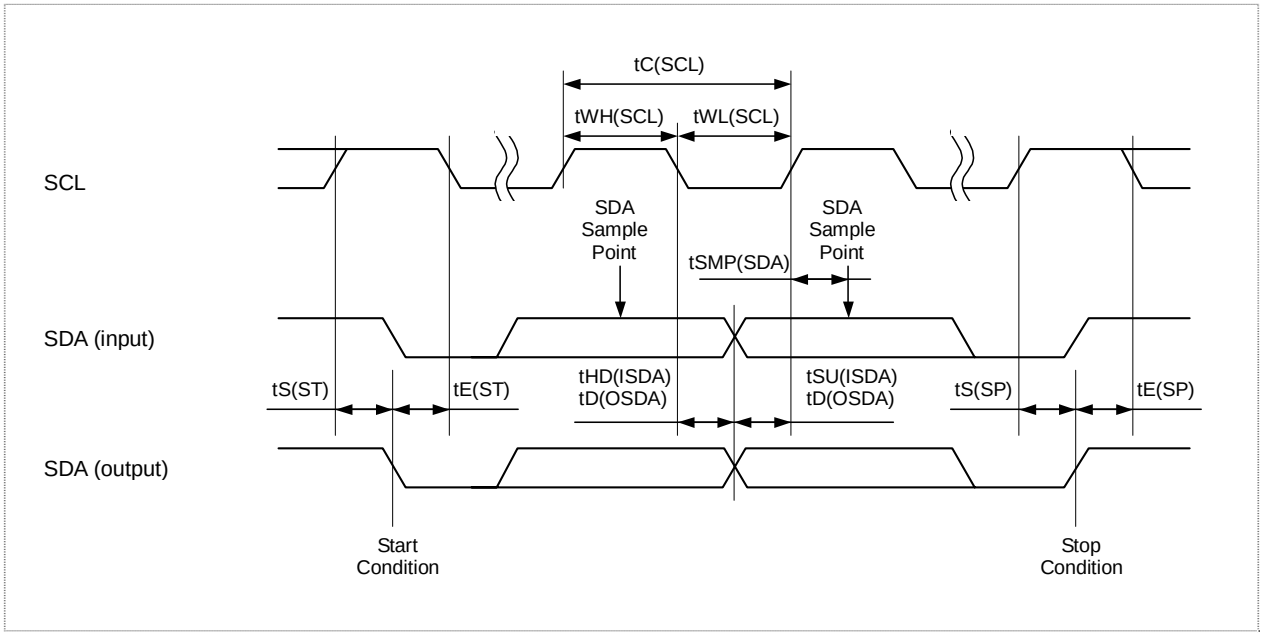


図 30.27　I2C Single Master Core Module タイミング

30.2.3.7 I2S タイミング

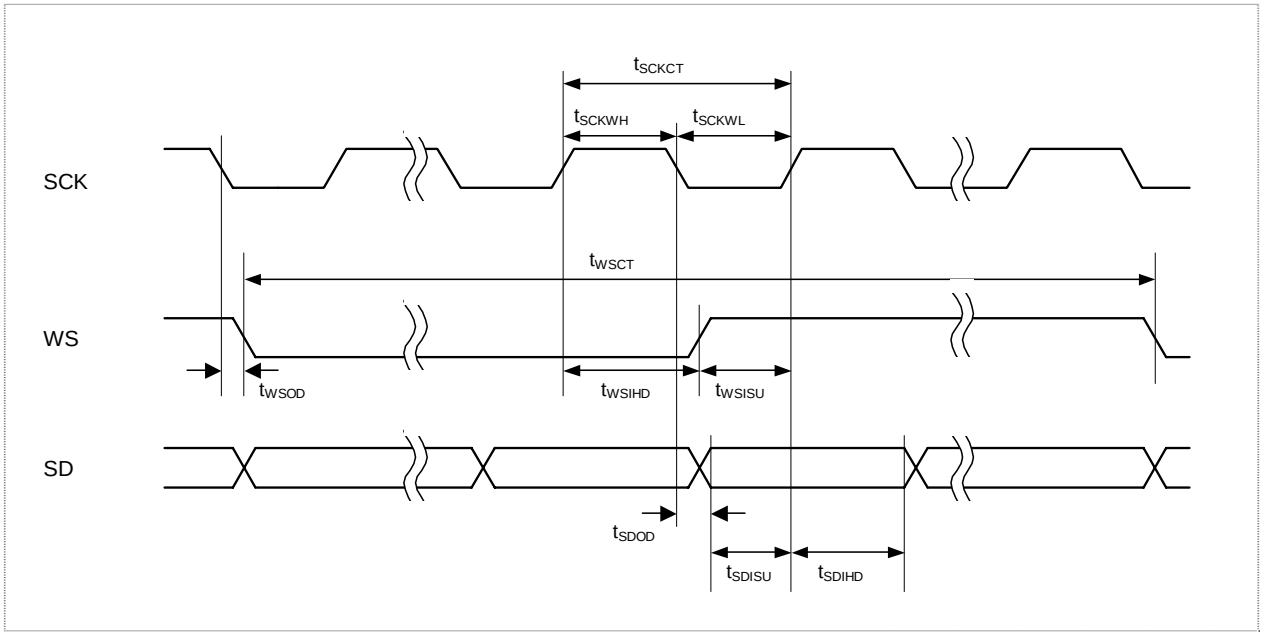


図 30.28　I2S タイミング

30.2.3.8 シリアル周辺機器インタフェースタイミング

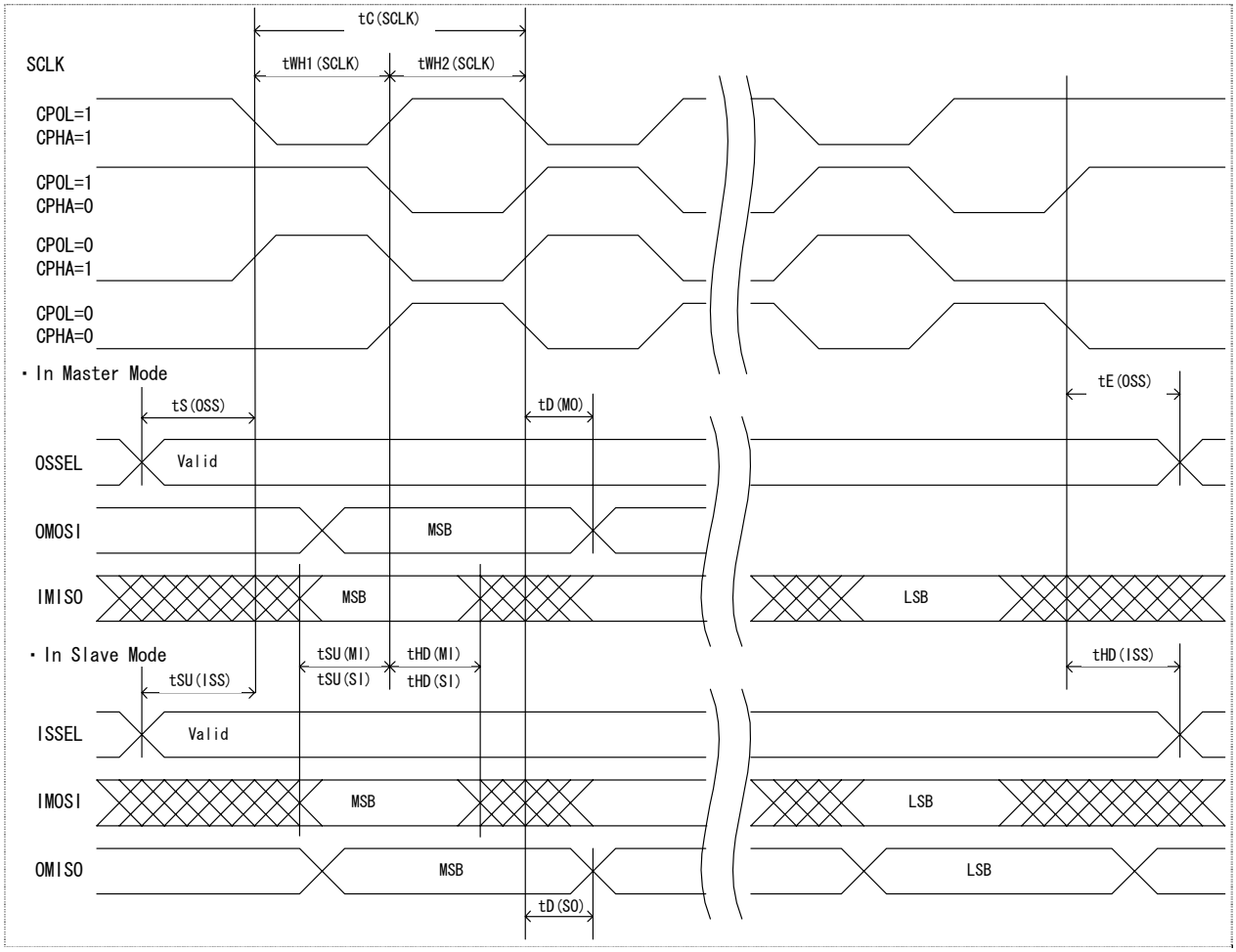


図 30.29　シリアル周辺機器インタフェースタイミング

30.2.3.9 コンパクトフラッシュインタフェース (CF) タイミング

■ CF Attribute Memory Read Cycle

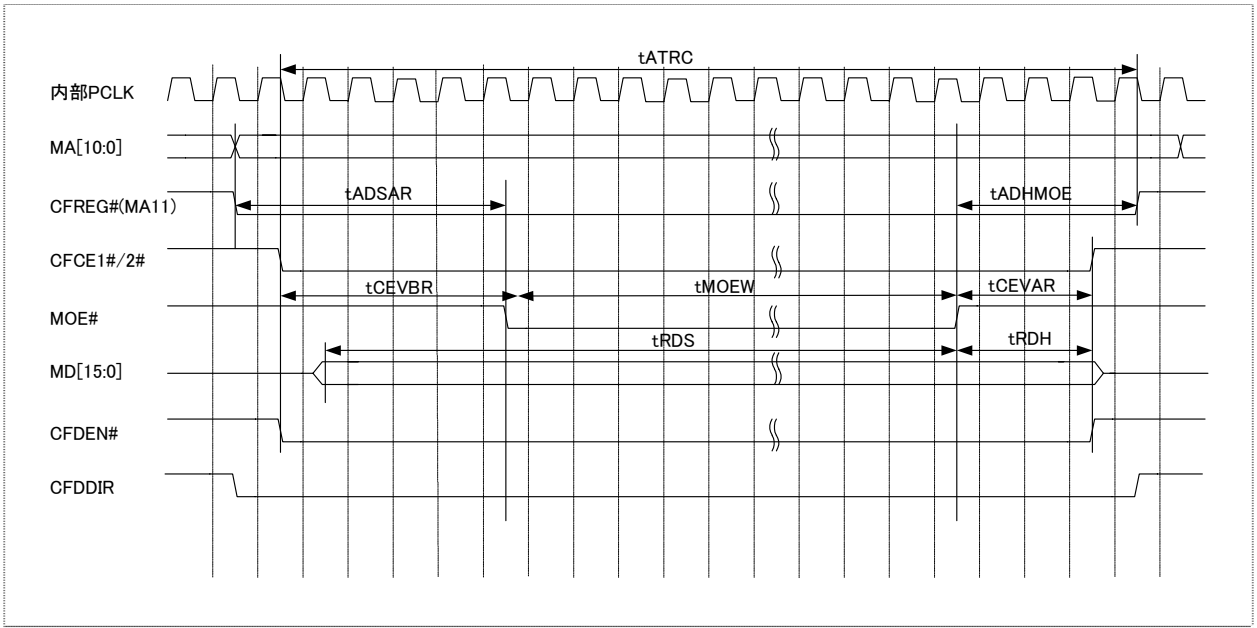


図 30.30　CF アトリビュートメモリ リードサイクル

■ CF Attribute Memory Write Cycle

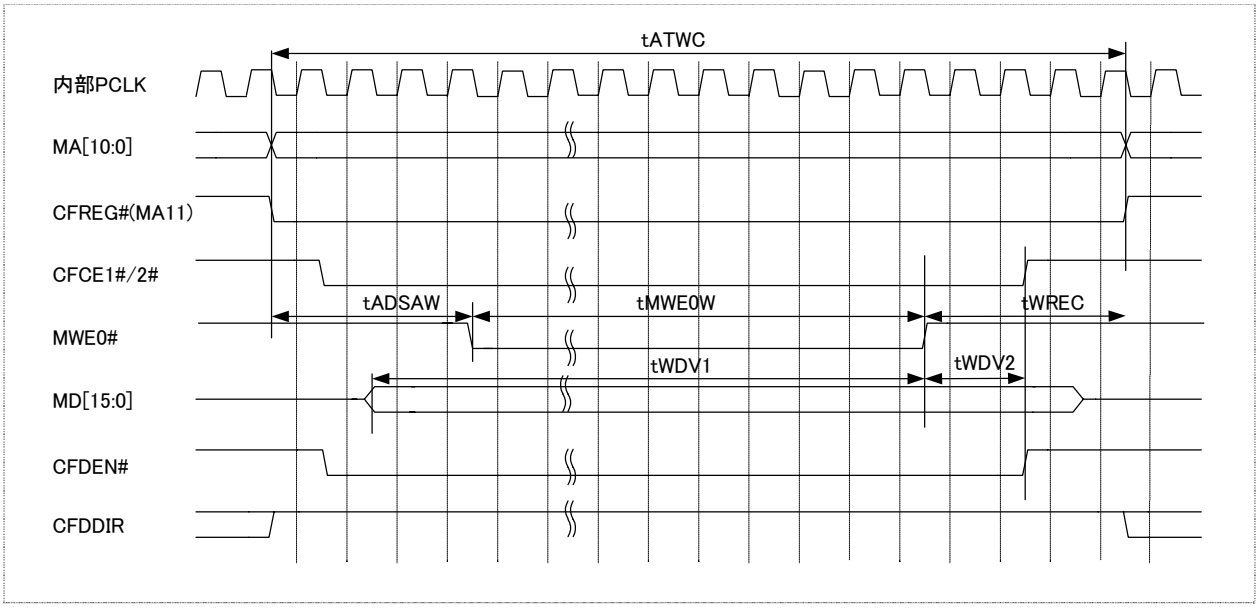


図 30.31　CF アトリビュートメモリ ライトサイクル

■ CF Common Memory Read Cycle

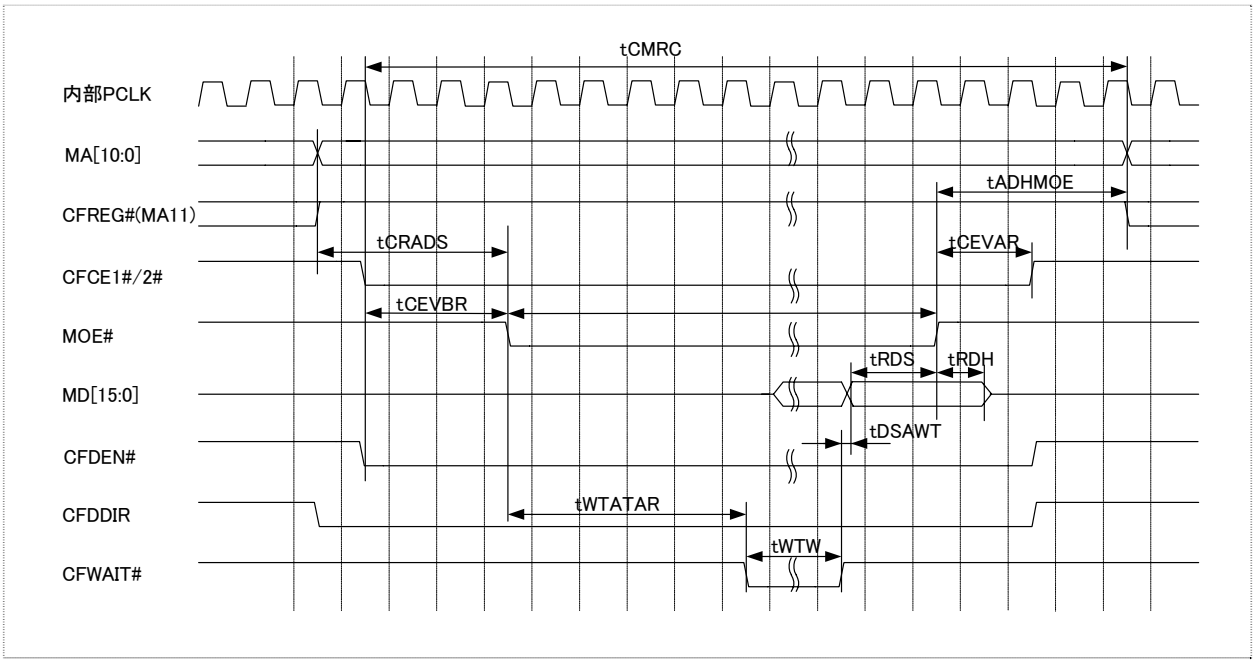


図 30.32　CF コモンメモリ リードサイクル

■ CF Common Memory Write Cycle

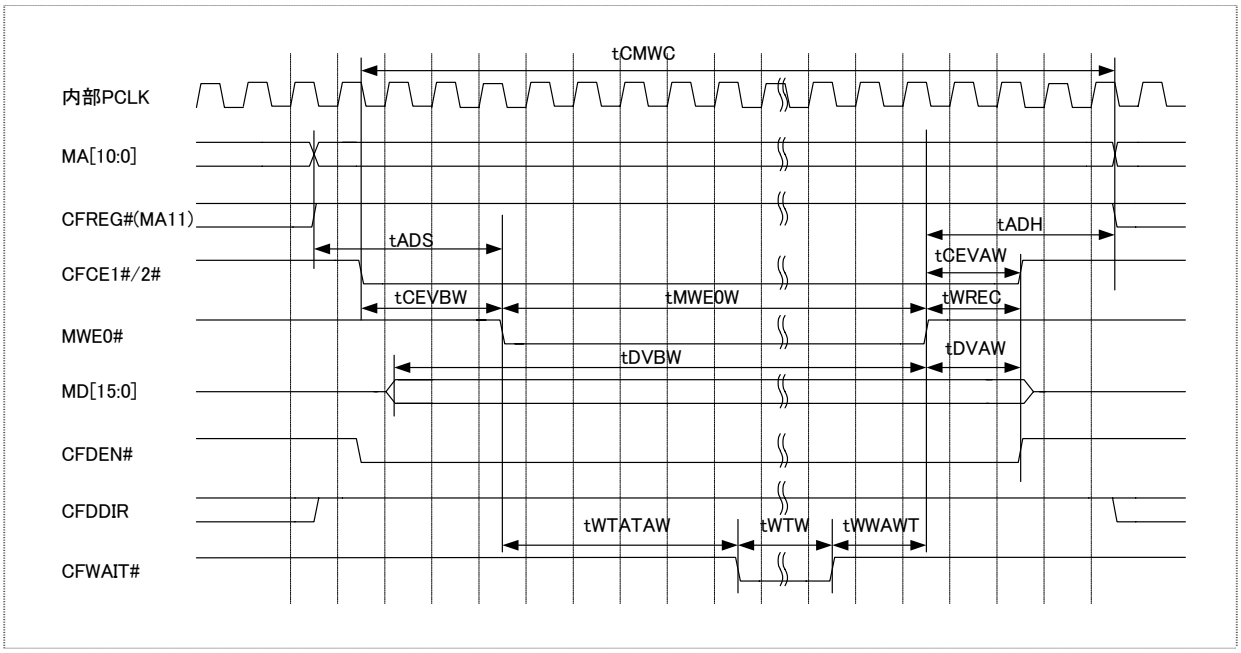


図 30.33　CF コモンメモリ ライトサイクル

■ CF I/O Space/IDE Read Cycle

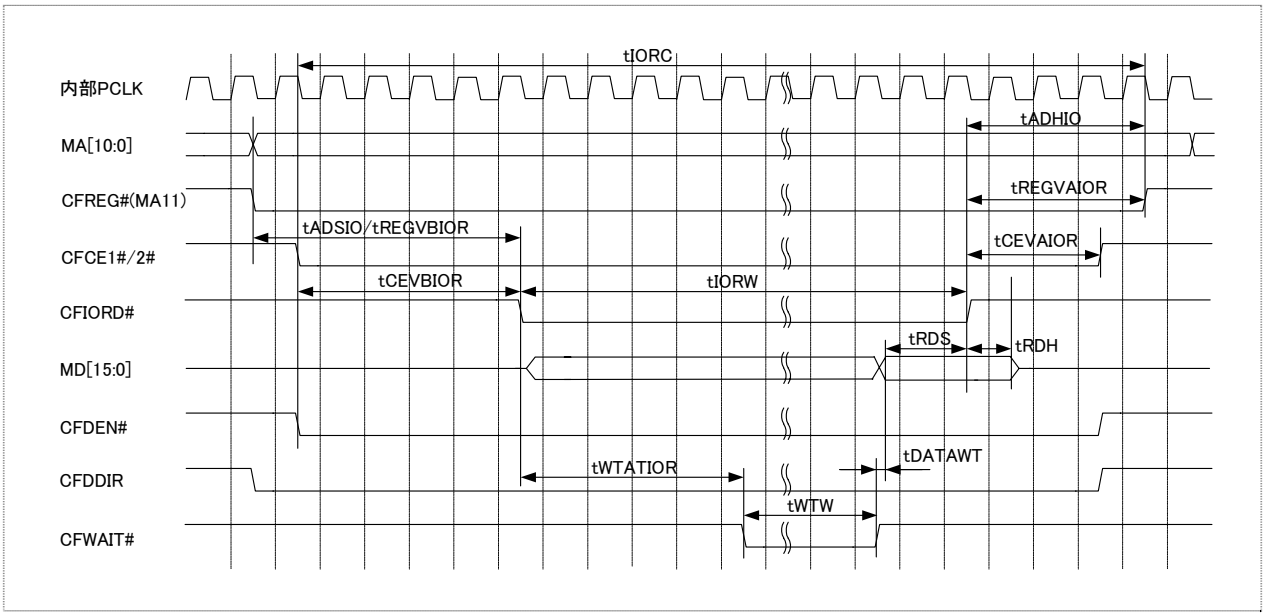


図 30.34　CF I/O 空間／IDE リードサイクル

■ CF I/O Space/IDE Write Cycle

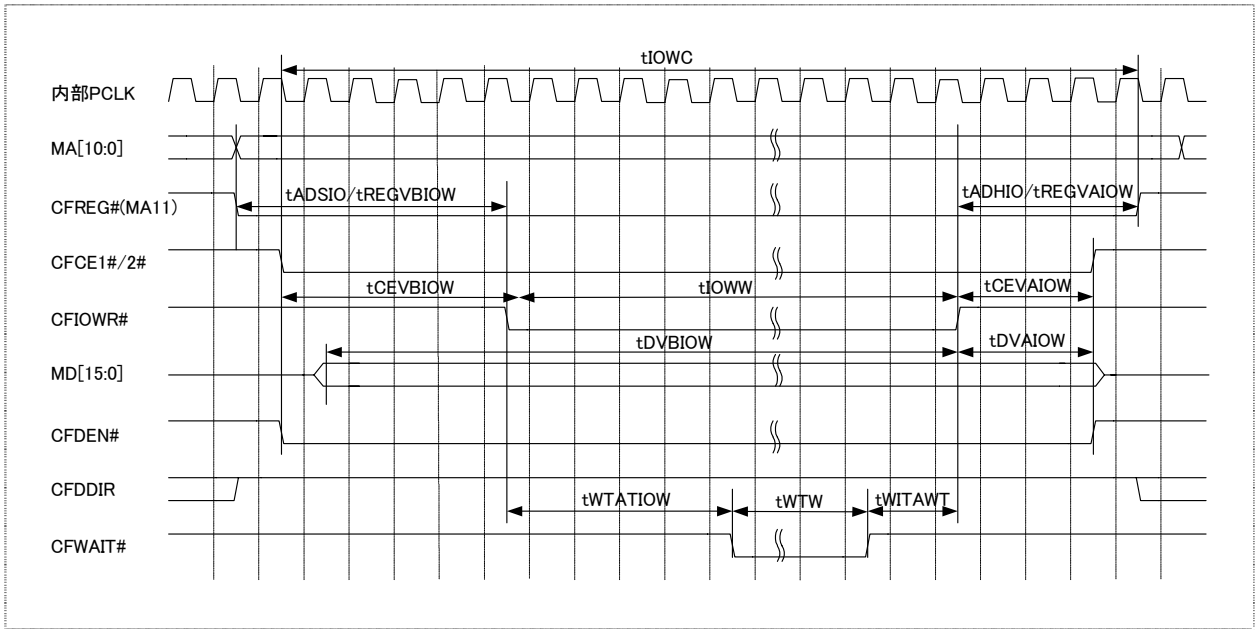


図 30.35　CF I/O 空間／IDE ライトサイクル

31. 参考外部接続例

31.1 メモリ接続例

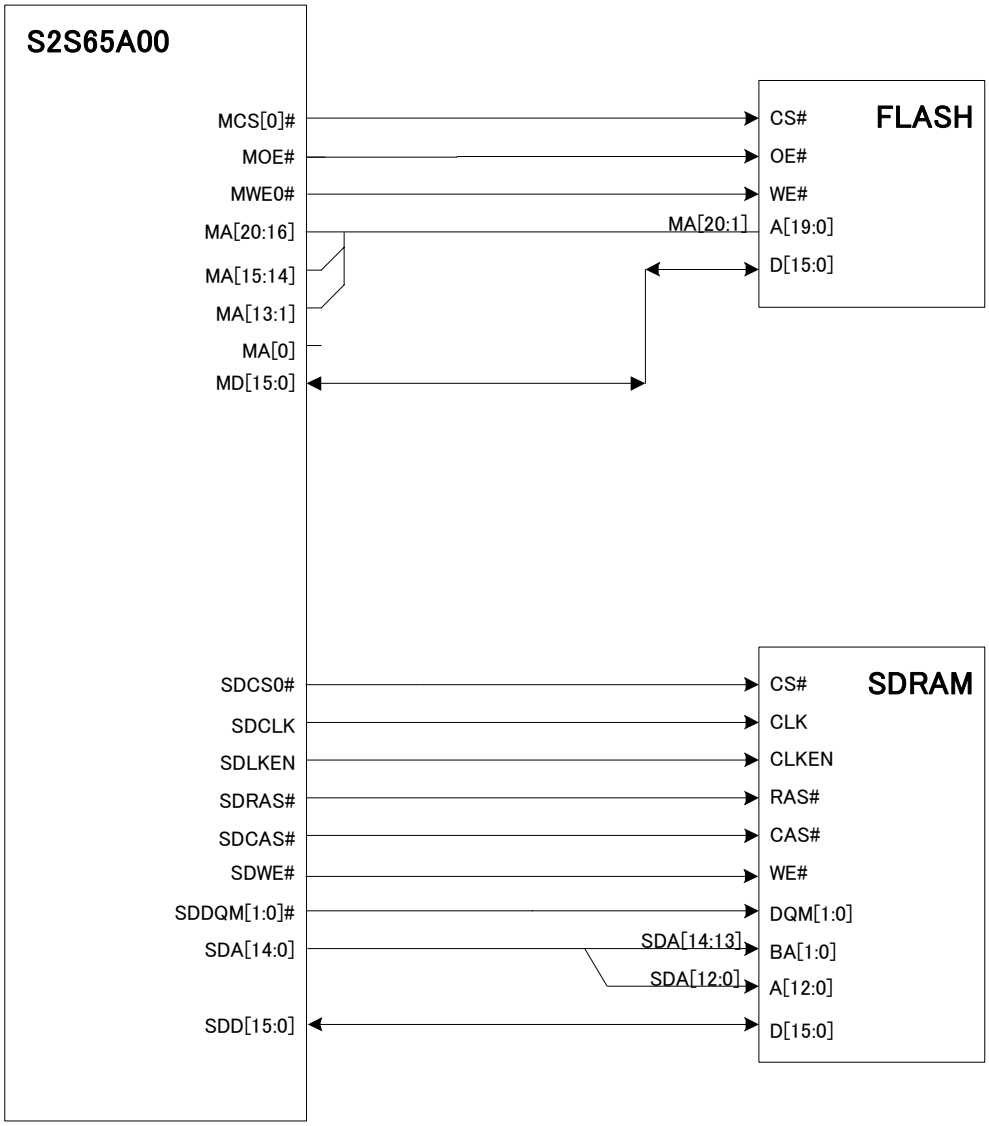


図 31.1　メモリ接続例（１）

注意：SDRAM のバンクアドレス（BA[1:0]）には SDMA[14:13]を接続してください。

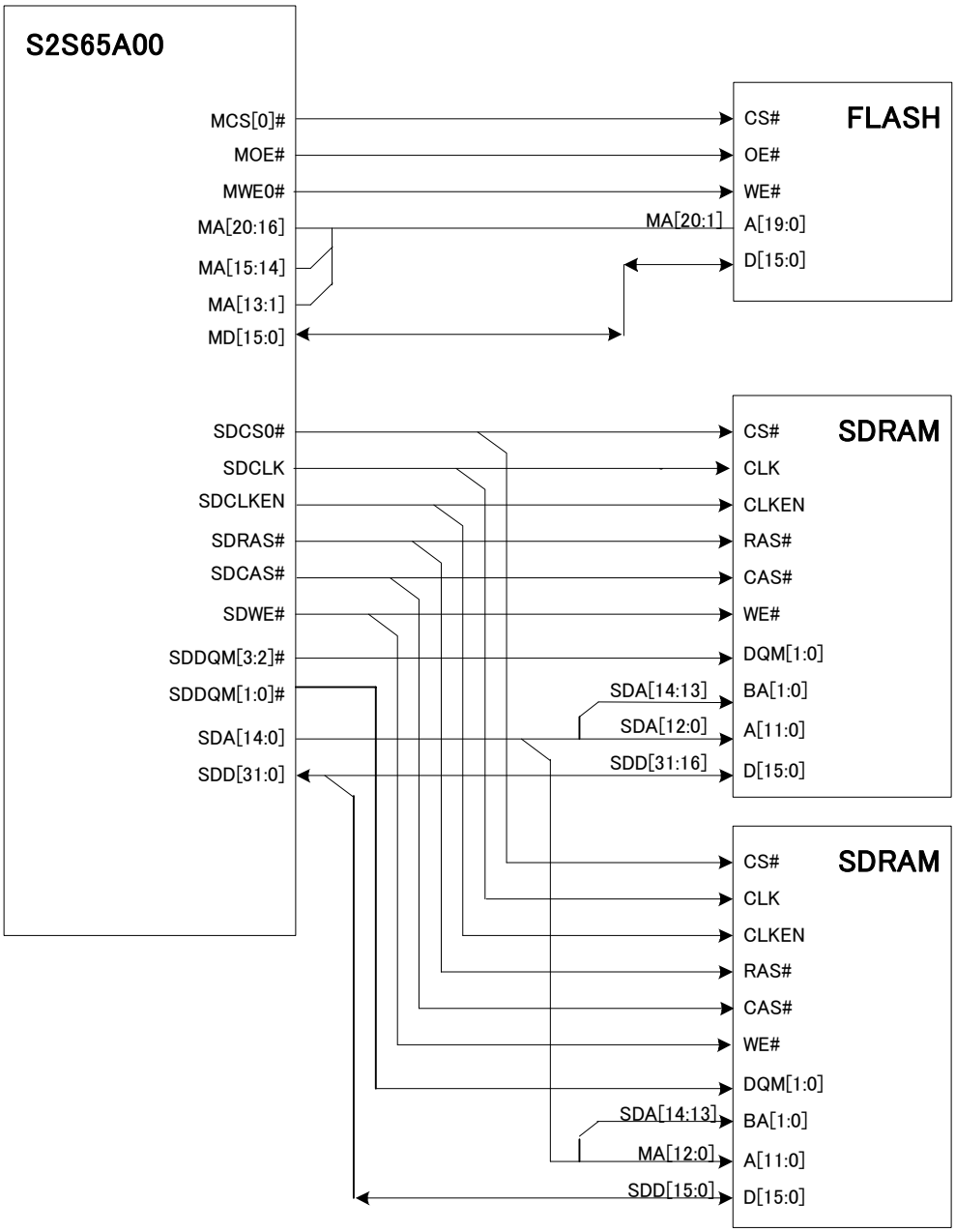


図 31.2　メモリ接続例（２）

注意：SDRAM のバンクアドレス（BA[1:0]）には SDA[14:13]を接続してください。

31.2 コンパクトフラッシュ接続例（16 ビットバス対応）

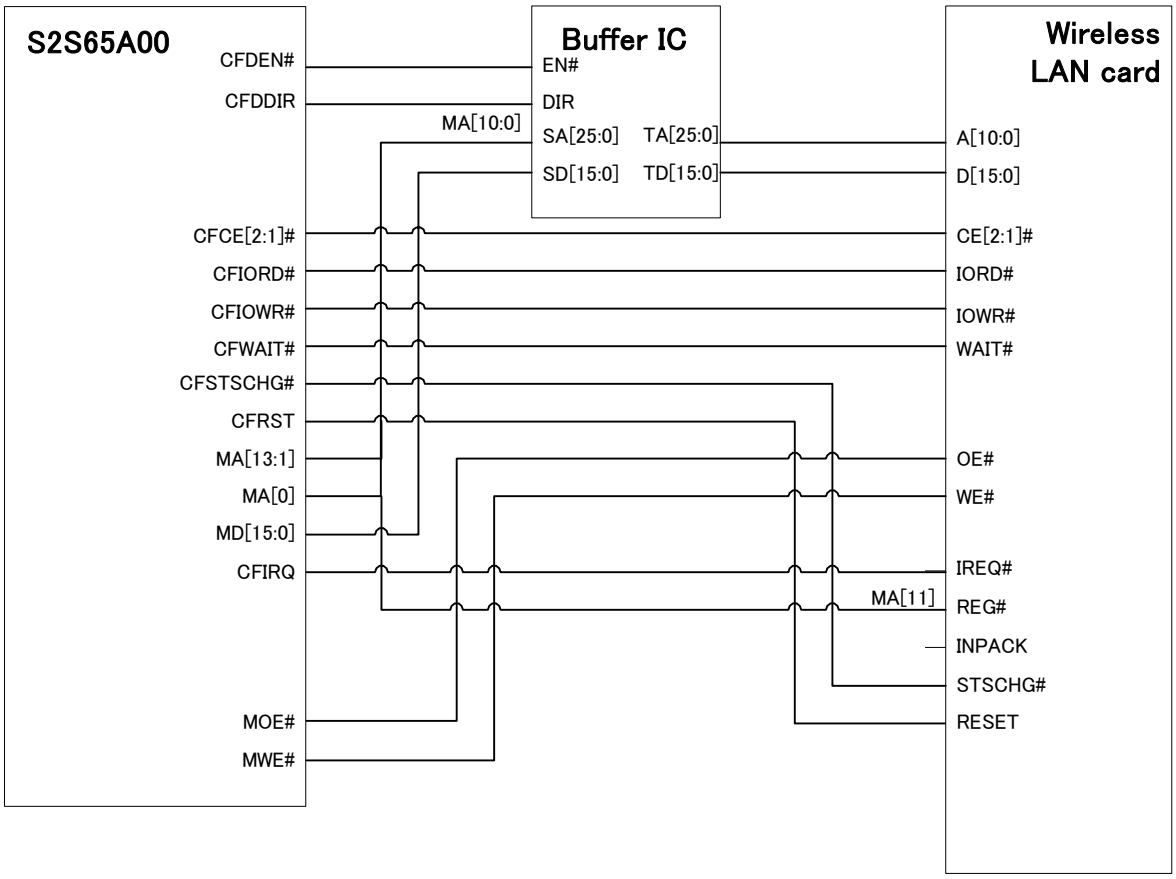
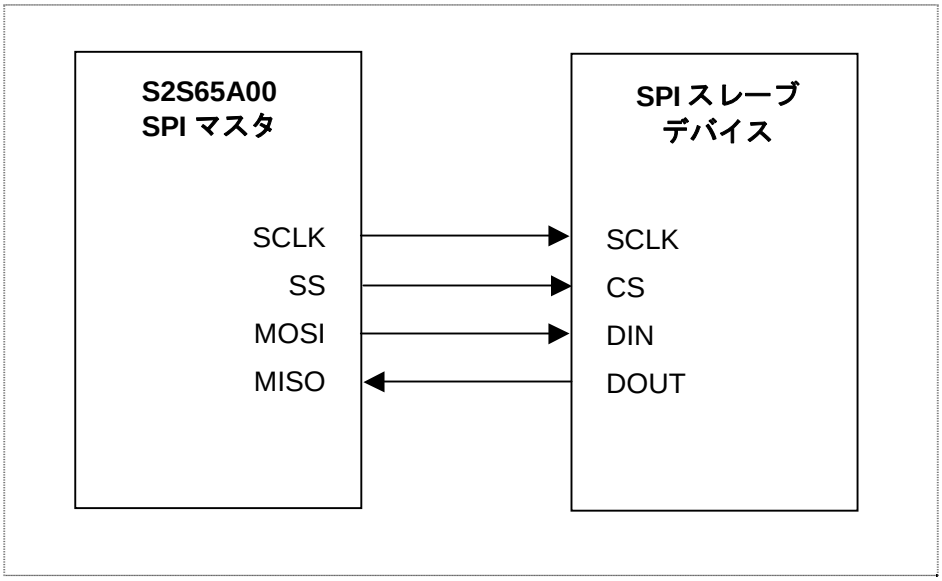


図 31.3　コンパクトフラッシュ I/F 接続例

31.3 シリアル周辺機器インタフェース（SPI）接続例

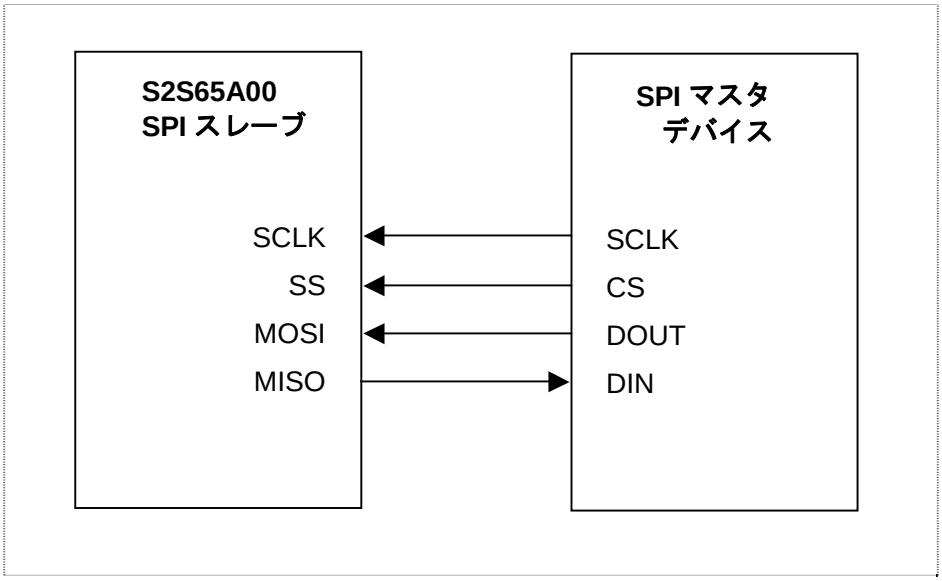
31.3.1 マスタ時

S2S65A00 がマスタ時の接続例です。



31.3.2 スレーブ時

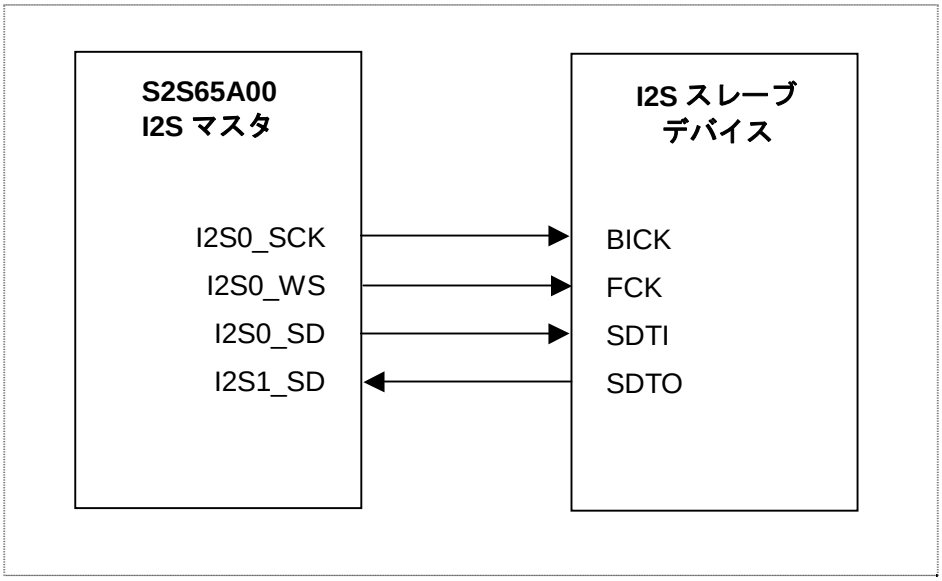
S2S65A00 がスレーブ時の接続例です。



31.4 I²S 接続例

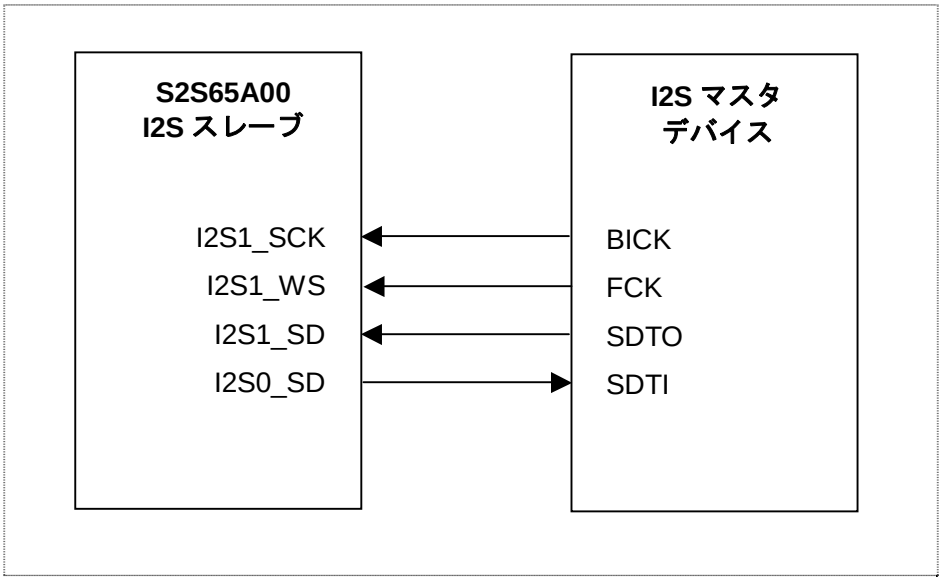
31.4.1 マスタ時

S2S65A00 がマスタ時の接続例です。



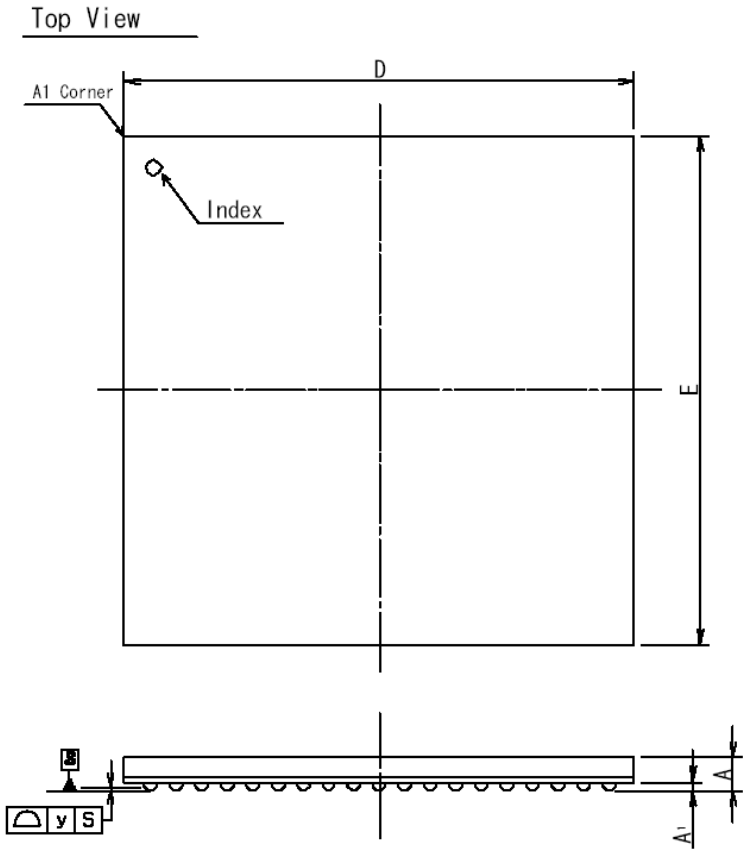
31.4.2 スレーブ時

S2S65A00 がスレーブ時の接続例です。



32. 外形寸法図

32.1 Plastic TFBGA 280pin Body size 16x16x1.2mm (PFBGA16U-280)



Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	—	16	—
E	—	16	—
A	—	—	1.2
A1	—	0.3	—
e	—	0.8	—
b	0.38	—	0.48
x	—	—	0.08
y	—	—	0.1
ZD	—	0.8	—
ZE	—	0.8	—

1 = 1mm

図 32.1　パッケージ寸法図（PFBGA16U-280PIN）

33. 改訂履歴表

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.0	2007/06/22	初版		
1.1	2007/09/13	P15 TESTCK の端子説明	テストクロック？	テストクロック端子 この端子はプルダウン抵抗を内蔵しています。 通常使用時は VSS へ接続するか、またはオープンにしてください。
		P19 リセット中の端子状態 SDCLK の RESET 中の値変更	Low	SDCLK(32KHz)
		P174 UART1/2/3 16.4 レジスタ説明の誤記	UART1：0x FFFF_5000 UART2：0x FFFD_4000 UART2：0x FFFD_5000	UART1：0x FFFF_5000 UART2：0x FFFD_4000 UART3：0x FFFD_5000
		P220 19.4 クロックとデータ転送		図 19.5 スレーブモード時の SCLK クロック波形追加
		P265 24.4 ブロック図(RTC)	4bit 月比較データ未記入	4bit 月比較データ記入
		P266 24.4 レジスタ(RTC)	表 24.1 レジスター覧未完成	表 24.1 レジスター覧作成
		P284 表 26.2 PortA の端子選択機能 GPAxMD[1:0]=11b	予約	GPIO 以外の機能２ (GPIOA[7:6],[3:2] のみ設定可能
		P285 表 26.3 PortB の端子選択機能 GPBxMD[1:0]=01b	GPIO 以外の機能１ (GPIOB[7:1] のみ設定可能	GPIO 以外の機能１
		P286 表 26.4 PortC の端子選択機能 GPCxMD[1:0]=11b	GPIO 以外の機能２ (GPIOC[6],[3:0]のみ設定可能	GPIO 以外の機能２ (GPIOC[7:3]のみ設定可能
		P287 表 26.5 PortD の端子選択機能 GPDxMD[1:0]=11b	GPIO 以外の機能２ (GPIOD[7:4] のみ設定可能	予約
		P289 表 26.7 PortF の端子選択機能 GPFxMD[1:0]=11b	GPIO 以外の機能２ (GPIOC[7:4] のみ設定可能	予約
		P290 表 26.8 PortG の端子選択機能 GPGxMD[1:0]=11b	GPIO 以外の機能２	予約
		P291 表 26.9 PortH の端子選択機能 GPHxMD[1:0]=11b	予約	GPIO 以外の機能２
		P292 表 26.10 PortI の端子選択機能 GPIxMD[1:0]=11b	予約	GPIO 以外の機能２ (GPIOI[0]のみ設定可能
		P293 表 26.11 PortJ の端子選択機能 GPJxMD[1:0]=01b	予約	GPIO 以外の機能１
		P294 表 26.12 PortK の端子選択機能 GPKxMD[1:0]=11b	予約	GPIO 以外の機能１

33. 改訂履歴表

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.1	2007/09/13	P314 表 30.1 DC 特性	消費電力：TBD	消費電力(LVDD) ロースピード：220uA ハイスピード：150mA 消費電力(RTCVDD) ロースピード： BUP#=HIGH：170uA BUP#=LOW：2uA ハイスピード：735uA 消費電力)PLLVDD) PLL 周波数 50MHz：1mA
1.2	2008/05/30	表紙・留意点	3.特性値のの数値の	3.特性値の数値の
		P1 1.1 特長	無線 LAN インタフェース(802.11b)対応可能	無線 LAN インタフェース(802.11b/g)対応可能
		P2 メモリコントローラ	SDRAM を最大 2 までサポート可能	SDRAM を最大 8 までサポート可能
		P3 A/D コンバータ	記述追加	A/D コンバータ： -8-Channel Input(コントローラによるセレクト) -10-bit 分解能 -1 回あたりの変換処理時間は 20μs 以下
		P4 2.ブロック図	CPU-Work:32KB max.	CPU-Work:54KB max. *1 *1：内蔵 SRAM は JPG[2:1]のラインバッファと共有しています。JPG[2:1]を使用する場合は CPU-Work として使用することは出来ません。
		P6 3.1 端子配置図 B19 C19 R6	 USBOSCO USBOSCI SD2	 USBCK_OSCO USBCK_OSCI SDD2
		P7 表 3.1 Cell Type の説明	Cell Type 追加 BLNC4	使用電源電圧 追加 USBDM:USB DM buffer USBDP:USB DP buffer USBVBUS:USB VBUS output buffer 削除

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.2	2008/05/30	P12,13,14,15,16 表 3.2 端子の説明 端子名称 CFIRQ の説明 R1 Cell Type ADIN[7:0] Cell Type GPIOJ[7:0] Cell Type GPIOK[7:0] Cell Type SYSCLKI Type 説明 SYS_OSCI Cell Type SYS_OSCO Cell Type SYSVCP Cell Type USBCK_OSCI Cell Type USBCK_OSCO Cell Type BUP# Cell Type GPIOA2 の説明 GPIOA3 の説明 GPIOB0 の説明 GPIOB1 の説明 GPIOB2 の説明 GPIOB3 の説明 端子名:GPIOJ[7:0]の説明 端子名:GPIOK[7:0]の説明	 CFIREQ 端子として LLIN HLIN BLNS4 BLNS4 I SYSCKSEL=”Low” の時、32KHz またはシステムクロックの 1/8 が出力 LLIN LLOT LLOT LLIN LLOT ICS I2S_WS:I2S 用ワードセレクト(GPIO 以外の機能 1) I2S_SCK:I2S 用シリアルクロック(GPIO 以外の機能 1) I2S_SDO:I2S 用シリアルデータ出力(GPIO 以外の機能 1) I2S_SDI:I2S 用シリアルデータ入力(GPIO 以外の機能 1) GPIOJ[7]0] GPIOK[7]0]	 CFIRQ 端子として ILTR IHTR BLNC4D2 BLNC4D2 I/O SYSCKSEL=”Low”のとき、設定により 32KHz またはシステムクロックの 1/4 が出力 ILTR OLTR OLTR ILTR OLTR ILS I2S1_WS:I2S1 用ワードセレクト(GPIO 以外の機能 2) 追記 I2S1_SCK:I2S1 用シリアルクロック(GPIO 以外の機能 2) 追記 I2S0_WS:I2S0 用ワードセレクト(GPIO 以外の機能 1) I2S0_SCK:I2S0 用シリアルクロック(GPIO 以外の機能 1) I2S0_SD:I2S0 用シリアルデータ(GPIO 以外の機能 1) I2S1_SD:I2S1 用シリアルデータ(GPIO 以外の機能 1) GPIOJ[7:0] GPIOK[7:0]
		P18 3.3 GPIO 端子のマルチプレクス端子機能、リセット直後の端子機能 端子名称 GPIOA2 GPIOA3 BPIOB0 GPIOB1 GPIOB2 GPIOB3	 (空白) (空白) I2S_WS I2S_SCK I2S_SDO I2S_SDI	 I2S1_WS I2S1_SCK I2S0_WS I2S0_SCK I2S0_SD I2S1_SD
		P19 3.3GPIO 端子の端子名	CFIREQ	CFIRQ

33. 改訂履歴表

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.2	2008/05/30	P20 3.4 リセット中の端子状態 Pin Name MA[19:0] RESET 中 MCS[2:0] RESET 中 MBEL# RESET 中 MBEH# RESET 中 SYSCLKI 説明 「RESET 中の値」の入力端子の値	 Low High High High (空白) High Low	 Low (但し bit11 のみ High) MCS[1:0]:High MCS[3:2]:High-Z (GPIOD[5:4]) Low Low SYSCLKI:SYSCKSEL に依存 High-Z(Pull-up) High-Z(Pull-down)
		P31 DMAC1[0x0C][0x1C]	0100： UART#1 出力 (UART#0-TX) 0101： UART#1 入力 (UART#0-RX) 0110： UART#2 出力 (UART#1-TX) 0111： UART#2 入力 (UART#1-RX) 1000： UART#3 出力 (UART#2-TX) 1001： UART#3 入力 (UART#2-RX)	0100： UART#1 出力 (TX) 0101： UART#1 入力 (RX) 0110： UART#2 出力 (TX) 0111： UART#2 入力 (RX) 1000： UART#3 出力 (TX) 1001： UART#3 入力 (RX)
		P37,44 表 7.2 レジスター覧 カメラ 1/2 ステータスレジスタ初期値	 0x0034	 0x0004
		P48 8.1 概要 8.2 ブロック図	ハフマンテーブルは DC,AC２テーブルづつを扱う、、、 30KB:SRAM	ハフマンテーブル DC,AC はそれぞれ２のテーブルを扱う、、、 SRAM *1 *1：SRAM の割り当ては、SYS[0x68] の Bits[5:4] EMBRAMSEL[1:0] を参照。
		P49 表 8.1 レジスター覧 予約レジスタ Default Value R/W	 0x0000 R/W or RO	 - -
		P98 8.5.5.6 ソフトウェアリセット処理について	各コンポーネントでの処理時間によるタイムラグがあるです。	各コンポーネントでの処理時間によるタイムラグがあるためです。
		P115 11.1 概要	S2S65A00 USB HS-Device Macro Specification	Appendix2 USB Device コントローラ

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.2	2008/05/30	P123 13.3 外部端子 端子名　SYSCLKI 入出力 追加 表 13.1 レジスター覧	I TIMBCLSEL	 I/O 端子名 SYSCKSEL 入出力:I 端子機能 32KHz システムクロック入力選択信号 “Low”:水晶振動子 “High”:水晶発振器 TIMBCKSEL
		P129 UART1/2 Clock Divider Register レジスタ説明	 このレジスタは UART1/2/3 用	 このレジスタは UART1/2 用
		P130 UART3 Clock Divider Register [30] レジスタ説明 Bit8 の記述 補足説明	 [0x30] UART1/2/3CLKDIV このレジスタは UART1/2/3 用 “0”:1/1 “1”:1/2 UART3 への UART3_SCLK は、 UART3CLKDIV の分周後、さらに 1/2 分周されたクロックで供給されます。	 UART3CLKDIV このレジスタは UART3 用 “0”:内部クロック使用 “1”:外部クロック使用 UART3 へ内部クロックを選択した場合、 UART3_SCLK は、 UART3CLKDIV の分周後、さらに 1/2 分周されたクロックで供給されます。 外部クロックを選択した場合は、分周は行わず入力されるクロックの周波数がそのまま供給されます。
		P132-134 SDD Bus Pulldown Register GPIOF Register Control Register GPIOG Register Control Register GPIOH Register Control Register GPIOI Register Control Register GPIOJ Register Control Register GPIOK Register Control Registe PORTFRCTL PORTIRCTL PORTKRCTL	 各ビットはそれぞれ MD[15:0]の各端子に対応しています。 PRTDPDDIS[7:0] GPIOG[7:0]端子内蔵プルダウン抵抗の 0:プルダウン抵抗 1:プルダウン抵抗 0:プルダウン抵抗 1:プルダウン抵抗 GPIOJ[7:0]端子内蔵プルアップ抵抗の GPIOK[7:0]端子内蔵プルアップ抵抗の Bit[7:4] Bit[7:0] 各ビットはそれぞれ GPIOG[7:0]の各端子	 各ビットはそれぞれ SDD[15:0]の各端子に対応しています。 PORTFPDDIS[7:0] GPIOG[7:0]端子内蔵プルアップ抵抗の 0:プルアップ抵抗 1:プルアップ抵抗 0:プルアップ抵抗 1:プルアップ抵抗 GPIOJ[7:0]端子内蔵プルダウン抵抗の GPIOK[7:0]端子内蔵プルダウン抵抗の Bit[7:0] Bit[1:0] 各ビットはそれぞれ GPIOK[7:0]の各端子

33. 改訂履歴表

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.2	2008/05/30	P135 EMBMEMCTL Bits[1:0]：EMBWAITEN	 11： Read Access Wait ON、 Write Access Wait ON、 Write Access Wait On	 11： Read Access Wait ON、 Read Data Wait ON、 Write Access Wait On
		P136 MISC	 USB_CTL_CTL	 USB_CLK_CTL
		P137 13.5 Appendix A	 98.1726MHz	 98.172928MHz
		P147 表 14.2 メモリコントローラ関連外部端子一覧	 SDD[31:0] SDDQMHH# SDDQMHL# SDDQMLH# SDDQMLL#	 SDD[31:16]:GPIOJ/K SDD[15:0] SDDQM3#:GPIOD7 SDDQM2#:GPIOD6 SDDQM1# SDDQM0#
		P150,151 レジスタ名称	 デバイス[3:0]タイミングレジスタ デバイス[3:0]制御レジスタ	 SRAM デバイス[3:0]タイミングレジスタ SRAM デバイス[3:0]制御レジスタ
		P153,154 SDCNFG Bit[23:20] COLW[7:4] コラムアドレス幅　COLW[5:4] COLW[7:6] Bit[7:6]	 BNUM[3:2] デバイス 2 設定用 予約 REF[3:2] リフレッシュサイクル	 BNUM[1:0] BNUM0:SDRAM デバイス 0 設定用 BNUM1:SDRAM デバイス 1 設定用 予約
		P154,155 SDADVCNFG	 SELF[3:2]	 SELF[1:0] SELF0:SDRAM デバイス 0 設定用 SELF1:SDRAM デバイス 1 設定用
		P155,156 SDINIT	 デバイスを DEVSEL (MEMC[0x60] bit[3:2]) で選択 DEVSEL[3:2]	 デバイスを DEVSEL[1:0] (MEMC[0x80] bit[3:2]) で選択 DEVSEL[1:0] DEVSEL0:SDRAM デバイス 0 設定用 DEVSEL1:SDRAM デバイス 1 設定用
		P157 SDSTAT	 DEVST3[3:0] DEVST2[3:0]	 DEVST1[3:0] SDRAM デバイス 1 ステータス DEVST0[3:0] SDRAM デバイス 0 ステータス

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.2	2008/05/30	P158 表 15.1 割り込みソース一覧	Ex. Battery Low(*)	Ex. Battery Low
		P159 表 15.2 割り込みソース一覧 注意(*)	IRQ14(**)とは異なり、GPIO の制御レジスタによって、は変更できません。	IRQ14(*)とは異なり、“26 GPIO”のレジスタ詳細説明 GPIO[0x60]～GPIO[0x6C]によって、変更できません。
		P165,166 IRQxx レジスタ[0x024] IRQxx レジスタ[0x028] IRQxx レジスタ[0x02C]	IRQxx[7:0]:GPIOB マスク前ステータス 表 15.1 を参照 IRQxx[7:0]:GPIOB マスク前ステータス 表 15.1 を参照 IRQxx[7:0]:GPIOB マスク前ステータス 表 15.1 を参照	IRQxx[7:0]:IRQxx マスク前ステータス（IRQGB[7:0]） 表 15.2 を参照 IRQxx[7:0]:IRQxx マスク前ステータス（IRQGB[7:0]） 表 15.2 を参照 IRQxx[7:0]:IRQxx マスク前ステータス（IRQGB[7:0]） 表 15.2 を参照
		P168,169 IRQxx レジスタ[0x0A0] IRQxx レジスタ[0x0A4] IRQxx レジスタ[0x0A8]	IRQxx[7:0]:GPIOB マスク前ステータス 表 15.1 を参照 IRQxx[7:0]:GPIOB マスク前ステータス 表 15.1 を参照 IRQxx[7:0]:GPIOB マスク前ステータス 表 15.1 を参照	IRQxx[7:0]:IRQxx マスク前ステータス（IRQGB[7:0]） 表 15.2 を参照 IRQxx[7:0]:IRQxx マスク前ステータス（IRQGB[7:0]） 表 15.2 を参照 IRQxx[7:0]:IRQxx マスク前ステータス（IRQGB[7:0]） 表 15.2 を参照
		P175,178 表 16.1 レジスター覧 割込み識別レジスタ初期値変更	0x00	0x01
		P207 I2S[1:0]制御レジスタ	Read Only	Read/Write
		P228 20.1 概要	割り込み出力信号(STSCHG#,IREQ)	割り込み出力信号(STSCHG#,IRQ)
		P230 表 20.2 外部端子(CF)	CFIREQ	CFIRQ
		P231,232,233 表 20.3 レジスター覧 CF Card Pin Status Register CF Card IRQ Source & Clear Register 初期値変更	 0x00XX 0x00XX	 0x0XXX 0x0XXX
		P233 CFINTRSTS	IREQ	CFIRQ

33. 改訂履歴表

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.2	2008/05/30	P236 21.1 概要	「SD メモリカードインターフェース」	「SD メモリカードインタフェース」
		P253 表 23.1 レジスター覧 タイマ B COMINP_0-3 レジスタ	R/W : R	R/W : RO
		P267 表 24.1 レジスター覧 テストクロックレジスタ R/W RTC RAM2 RAM6 R/W	RO (空白)	WO R/W
		P278,279 表 25.1 レジスター覧 ウォッチドッグタイマ制御レジスタ 初期値変更	0x0000_A500	0x000_0000
		P282 26.2 外部端子 信号名変更 端子名 GPIOA0 GPIOA1 GPIOA2 GPIOA3 GPIOA4 GPIOA5 GPIOA6 GPIOA7	 TXD0 RXD0 RTS0 CTS0 TXD1 RXD1 RTS1 CTS1	 TXD1 RXD1 RTS1 CTS1 TXD2 RXD2 RTS2 CTS2
		P283 26.2 外部端子 GPIOH6	CFIREQ	CFIRQ
		P293 GPIOI_DATA GPIOI_FUNC	GPIOIDATA[7:0] GPI[7:2]MD[1:0]	GPIOIDATA[1:0] n/a
		P298 26.4 GPIOA および GPIOB の割り込みロジック	GPIOI 及び GPIOK・・・・接続して下さい。	削除
		P299 27.1 概要 記述内容変更	変換時間は 1 回あたり 10μs 以下で、	変換時間は 1 回あたり 20μs 以下(システムクロック≒49MHz)で、
		P307 27.7 入力電圧と変換データ 1.8V	0x_8C40	0x8C_40
		P313 29.2 推奨動作条件 の VSS 記述追加	VSS=PLLVSS=0[V]	UVSS = PVSS = XVSS = 0[V] VSS = PLLVSS = AVSS = 0[V]
		P313 29.1 絶対最大定格	HVDD LVDD	HVDD,UVDD3,C1VDD, C2VDD,SDVDD,AVDD LVDD,UPVDD,UXVDD, PLLVDD,RTCVDD
		P314 29.4 電源切断タイミング	3.3V 系電源(HVDD1)	3.3V 系電源(HVDD)

Revision	年/月/日	内容		
		改訂部分	改訂前	改訂後
1.2	2008/05/30	P315		表 30.2 DC 特性 LVDD　追記 表 30.2 USBVBUS 判定電圧　追記
		P317,318 表 30.6 スタテックメモリタイミング データ出力許可信号インターバル期間:t_{MOEI} 書き込み有効信号有効期間:t_{MWEV}	 n_2 $n_4Ts-0.5$	 n_1 $n_4Ts+1 -n_3Ts-0.5$
		P327 図 30.8 スタテックメモリリードタイミング 図 30.9 スタテックメモリライトタイミング	tMAD tMAD tBEH	tMAS tMAS tMBEH
		P345 図 31.3 コンパクトフラッシュ I/F 接続例	CFIREQ	CFIRQ
		P376 Appendix2 A2.2 Onerview A2.3 Block Diagram	 4.5KB の EndPoint 用 FIFO RAM(4.5KB)	 4.0KB の EndPoint 用 FIFO RAM(4.0KB)
		P377 Appendix2 A2.4 Pin List	 USB インターフェース関連 USBOSCI USBOSCO	 USB インタフェース関連 USBCK_OSCI USBCK_OSCO
1.3	2011/12/06	P29, P31, P33 DMA コントロールレジスタ初期値 チャネル 0/1/2/3	0x0000_0000	0x00XX_XXXX
		P49, P69 JPEG ラインバッファカレントステータスフラグレジスタ初期値	0x0009	0xX009
		P167 IRQ 極性レジスタ		SD カード割り込み使用時の設定を追加
		P175, P182 ラインステータスレジスタ	0x00	0x60
		P175, P184 モデムスタータスレジスタ	0x00	0xEX
		P175, P187 テストステータスレジスタ 1	0x00	0x01
		P175, P187 テストステータスレジスタ 2	0x00	0x0F
		P175, P187 テストステータスレジスタ 3	0x00	0x02
		P193,199 I/O コントロールレジスタ初期値	0x11	0x00

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称		レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_3000	**TIMB 関連レジスタ**		**TIMB**			
0x FFFD_3000	タイマ B モード制御レジスタ		TIMBMDC	0x0000	R/W	16(/32)
0x FFFD_3004	タイマ B カウンタ制御レジスタ		TIMBCNTC	0x0000	R/W	16(/32)
0x FFFD_3008	タイマ B ステータスレジスタ		TIMBSTS	0x0000	R/W	16(/32)
0x FFFD_300C	タイマ B 割り込み許可レジスタ		TIMBINTEN	0x0000	R/W	16(/32)
0x FFFD_3010	タイマ B カウントレジスタ		TIMBCNT	0x0000	R/W	16(/32)
0x FFFD_3014	タイマ B COMMON 制御レジスタ		TIMBCOMC	0x0000	R/W	16(/32)
0x FFFD_3018	タイマ B I/O 制御レジスタ		TIMBIO	0x0000	R/W	16(/32)
0x FFFD_301C	タイマ B 周期レジスタ		TIMBDUTY	0x0000	R/W	16(/32)
0x FFFD_3020	タイマ B COMMON_0 レジスタ		TIMBCOM0	0x0000	R/W	16(/32)
0x FFFD_3024	タイマ B COMMON_1 レジスタ		TIMBCOM1	0x0000	R/W	16(/32)
0x FFFD_3028	タイマ B COMMON_2 レジスタ		TIMBCOM2	0x0000	R/W	16(/32)
0x FFFD_302C	タイマ B COMMON_3 レジスタ		TIMBCOM3	0x0000	R/W	16(/32)
0x FFFD_3030	タイマ B COMINP_0 レジスタ		TIMBINP0	0x0000	RO	16(/32)
0x FFFD_3034	タイマ B COMINP_1 レジスタ		TIMBINP1	0x0000	RO	16(/32)
0x FFFD_3038	タイマ B COMINP_2 レジスタ		TIMBINP2	0x0000	RO	16(/32)
0x FFFD_303C	タイマ B COMINP_3 レジスタ		TIMBINP3	0x0000	RO	16(/32)
0x FFFD_4000	**DLAB**	**UART2 関連レジスタ**	**UART2**			
0x FFFD_4000	0	受信バッファレジスタ	RBR	0x 00	RO	8 (/16/32)
0x FFFD_4000	0	送信ホールディングレジスタ	THR	-	WO	8 (/16/32)
0x FFFD_4000	1	デバイザラッチ LSB レジスタ	DLL	0x 00	R/W	8 (/16/32)
0x FFFD_4004	0	割り込みイネーブルレジスタ	IER	0x 00	R/W	8 (/16/32)
0x FFFD_4004	1	デバイザラッチ MSB レジスタ	DLM	0x 00	R/W	8 (/16/32)
0x FFFD_4008	割り込み識別レジスタ		IIR	0x 01	RO	8 (/16/32)
0x FFFD_4008	FIFO 制御レジスタ		FCR	-	WO	8 (/16/32)
0x FFFD_400C	ライン制御レジスタ		LCR	0x 00	R/W	8 (/16/32)
0x FFFD_4010	モデム制御レジスタ		MCR	0x 00	R/W	8 (/16/32)
0x FFFD_4014	ラインステータスレジスタ		LSR	0x 60	RO	8 (/16/32)
0x FFFD_4018	モデムステータスレジスタ		MSR	0x EX	RO	8 (/16/32)
0x FFFD_401C	スクラッチレジスタ		SCR	0x 00	R/W	8 (/16/32)
0x FFFD_4020	テスト 0 レジスタ		T0	0x 00	R/W	8 (/16/32)
0x FFFD_4024	テスト 1 レジスタ		T1	0x 00	R/W	8 (/16/32)
0x FFFD_4028	テストステータス 0 レジスタ		TS0	-	RO	8 (/16/32)
0x FFFD_402C	テストステータス 1 レジスタ		TS1	0x 01	RO	8 (/16/32)
0x FFFD_4030	テストステータス 2 レジスタ		TS2	0x 0F	RO	8 (/16/32)
0x FFFD_403C	テストステータス 3 レジスタ		TS3	0x 02	RO	8 (/16/32)
0x FFFD_5000	**DLAB**	**UART3 関連レジスタ**	**UART3**			
0x FFFD_5000	0	受信バッファレジスタ	RBR	0x 00	RO	8 (/16/32)
0x FFFD_5000	0	送信ホールディングレジスタ	THR	-	WO	8 (/16/32)
0x FFFD_5000	1	デバイザラッチ LSB レジスタ	DLL	0x 00	R/W	8 (/16/32)
0x FFFD_5004	0	割り込みイネーブルレジスタ	IER	0x 00	R/W	8 (/16/32)
0x FFFD_5004	1	デバイザラッチ MSB レジスタ	DLM	0x 00	R/W	8 (/16/32)
0x FFFD_5008	割り込み識別レジスタ		IIR	0x 01	RO	8 (/16/32)
0x FFFD_5008	FIFO 制御レジスタ		FCR	-	WO	8 (/16/32)
0x FFFD_500C	ライン制御レジスタ		LCR	0x 00	R/W	8 (/16/32)
0x FFFD_5010	モデム制御レジスタ		MCR	0x 00	R/W	8 (/16/32)
0x FFFD_5014	ラインステータスレジスタ		LSR	0x 60	RO	8 (/16/32)
0x FFFD_5018	モデムステータスレジスタ		MSR	0x EX	RO	8 (/16/32)
0x FFFD_501C	スクラッチレジスタ		SCR	0x 00	R/W	8 (/16/32)

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_5020	テスト 0 レジスタ	T0	0x 00	R/W	8 (/16/32)
0x FFFD_5024	テスト 1 レジスタ	T1	0x 00	R/W	8 (/16/32)
0x FFFD_5028	テストステータス 0 レジスタ	TS0	-	RO	8 (/16/32)
0x FFFD_502C	テストステータス 1 レジスタ	TS1	0x 01	RO	8 (/16/32)
0x FFFD_5030	テストステータス 2 レジスタ	TS2	0x 0F	RO	8 (/16/32)
0x FFFD_503C	テストステータス 3 レジスタ	TS3	0x 02	RO	8 (/16/32)
0x FFFD_8000	**カメラ 2 インタフェース関連レジスタ**	**CAM2**			
0x FFFD_8000	カメラ 2 クロック周期設定レジスタ		0x 0000	R/W	16
0x FFFD_8004	カメラ 2 信号設定レジスタ		0x 0000	R/W	16
0x FFFD_8008～ 0x FFFD_801C	予約		-	-	-
0x FFFD_8020	カメラ 2 モード設定レジスタ		0x 0000	R/W	16
0x FFFD_8024	カメラ 2 フレーム制御レジスタ		0x 0000	R/W	16
0x FFFD_8028	カメラ 2 コントロールレジスタ		0x 0000	WO	16
0x FFFD_802C	カメラ 2 ステータスレジスタ		0x 0004	RO	16
0x FFFD_8030～ 0x FFFD_805C	予約		-	-	-
0x FFFD_9000	**JPEG2 リサイズ関連レジスタ**	**RSZ2**			
0x FFFD_9060	グローバルリサイザ制御レジスタ		0x 0000	WO	16
0x FFFD_9064	キャプチャ制御ステートレジスタ		0x 0000	RO	16
0x FFFD_9068	キャプチャデータ設定レジスタ		0x 0000	R/W	16
0x FFFD_9070～ 0x FFFD_907C	予約レジスタ		-	-	-
0x FFFD_90C0	取り込みリサイズ制御レジスタ		0x 0000	R/W	16
0x FFFD_90C8	取り込みリサイズスタート X 座標レジスタ		0x 0000	R/W	16
0x FFFD_90CC	取り込みリサイズスタート Y 座標レジスタ		0x 0000	R/W	16
0x FFFD_90D0	取り込みリサイズエンド X 座標レジスタ		0x 027F	R/W	16
0x FFFD_90D4	取り込みリサイズエンド Y 座標レジスタ		0x 01DF	R/W	16
0x FFFD_90D8	取り込みリサイズ縮小率レジスタ		0x 8080	R/W	16
0x FFFD_90DC	取り込みリサイズ縮小モードレジスタ		0x 0000	R/W	16
0x FFFD_A000	**JPEG2 モジュールレジスタ**	**JCTL2**			
0x FFFD_A000	JPEG 制御レジスタ		0x 0000	R/W	16
0x FFFD_A004	JPEG ステータスフラグレジスタ		0x 8080	R/W	16
0x FFFD_A008	JPEG ロウステータスフラグレジスタ		0x 8080	RO	16
0x FFFD_A00C	JPEG 割り込み制御レジスタ		0x 0000	R/W	16
0x FFFD_A010	予約レジスタ		-	-	-
0x FFFD_A014	JPEG コーデックスタートストップ制御レジスタ		0x 0000	WO	16
0x FFFD_A018～ 0x FFFD_A01C	予約レジスタ		-	-	-
0x FFFD_A020	ハフマンテーブル自動設定レジスタ		0x 0000	R/W	16
0x FFFD_A040	**JPEG2 FIFO 設定レジスタ**	**JFIFO2**			
0x FFFD_A040	JPEG FIFO 制御レジスタ		0x 0000	R/W	16
0x FFFD_A044	JPEG FIFO ステータスレジスタ		0x 8001	RO	16
0x FFFD_A048	JPEG FIFO サイズレジスタ		0x 003F	R/W	16
0x FFFD_A04C	JPEG FIFO リード/ライトポートレジスタ		0x 0000_0000	R/W	32
0x FFFD_A050～ 0x FFFD_A058	予約レジスタ		-	-	-
0x FFFD_A060	エンコードサイズリミットレジスタ 0		0x 0000	R/W	16
0x FFFD_A064	エンコードサイズリミットレジスタ 1		0x 0000	R/W	16
0x FFFD_A068	エンコードサイズ結果レジスタ 0		0x 0000	RO	16
0x FFFD_A06C	エンコードサイズ結果レジスタ 1		0x 0000	RO	16

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_A070～ 0x FFFD_A078	予約レジスタ		-	-	16
0x FFFD_A080	**JPEG2 ラインバッファ設定レジスタ**	**JLB2**			
0x FFFD_A080	JPEG ラインバッファステータスフラグレジスタ		0x 0000	R/W	16
0x FFFD_A084	JPEG ラインバッファロウステータスフラグレジスタ		0x 0000	RO	16
0x FFFD_A088	JPEG ラインバッファカレントステータスフラグレジスタ		0x X009	RO	16
0x FFFD_A08C	JPEG ラインバッファ割り込み制御レジスタ		0x 0000	R/W	16
0x FFFD_A090～ 0x FFFD_A09C	予約レジスタ		-	-	-
0x FFFD_A0A0	JPEG ラインバッファ水平ピクセル許容サイズレジスタ		0x 2800	R/W	16
0x FFFD_A0A4	JPEG ラインバッファメモリアドレスオフセットレジスタ		0x 0020	R/W	16
0x FFFD_A0A8～ 0x FFFD_A0BC	予約レジスタ		-	-	-
0x FFFD_A0C0	JPEG ラインバッファリード/ライトポートレジスタ		0x 0000	R/W	16
0x FFFD_B000	**JPEG2 コーデックレジスタ**	**JCOCEC2**			
0x FFFD_B000	動作モード設定レジスタ		0x 0000	R/W	16
0x FFFD_B004	コマンド設定レジスタ		Not applicable	WO	16
0x FFFD_B008	JPEG 動作ステータスレジスタ		0x 0000	RO	16
0x FFFD_B00C	量子化テーブル番号レジスタ		0x 0000	R/W	16
0x FFFD_B010	ハフマンテーブル番号レジスタ		0x 0000	R/W	16
0x FFFD_B014	DRI 設定レジスタ 0		0x 0000	R/W	16
0x FFFD_B018	DRI 設定レジスタ 1		0x 0000	R/W	16
0x FFFD_B01C	垂直ピクセルサイズレジスタ 0		0x 0000	R/W	16
0x FFFD_B020	垂直ピクセルサイズレジスタ 1		0x 0000	R/W	16
0x FFFD_B024	水平ピクセルサイズレジスタ 0		0x 0000	R/W	16
0x FFFD_B028	水平ピクセルサイズレジスタ 1		0x 0000	R/W	16
0x FFFD_B02C～ 0x FFFD_B034	予約レジスタ		-	-	-
0x FFFD_B038	RST マーカ動作設定レジスタ		0x 0000	R/W	16
0x FFFD_B03C	RST マーカ動作ステータスレジスタ		0x 0000	RO	16
0x FFFD_B040～ 0x FFFD_B0CC	挿入マーカデータレジスタ		0x 00FF	R/W	16
0x FFFD_B400～ 0x FFFD_B4FC	量子化テーブル No.0 レジスタ		Not applicable	R/W	16
0x FFFD_B500～ 0x FFFD_B5FC	量子化テーブル No.1 レジスタ		Not applicable	R/W	16
0x FFFD_B800～ 0x FFFD_B83C	DC ハフマンテーブル No.0 レジスタ 0		Not applicable	WO	16
0x FFFD_B840～ 0x FFFD_B86C	DC ハフマンテーブル No.0 レジスタ 1		Not applicable	WO	16
0x FFFD_B880～ 0x FFFD_B8BC	AC ハフマンテーブル No.0 レジスタ 0		Not applicable	WO	16
0x FFFD_B8C0～ 0x FFFD_BB44	AC ハフマンテーブル No.0 レジスタ 1		Not applicable	WO	16
0x FFFD_BC00～ 0x FFFD_BC3C	DC ハフマンテーブル No.1 レジスタ 0		Not applicable	WO	16

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_BC40～ 0x FFFD_BC6C	DC ハフマンテーブル No.1 レジスタ 1		Not applicable	WO	16
0x FFFD_BC80～ 0x FFFD_BCBC	AC ハフマンテーブル No.1 レジスタ 0		Not applicable	WO	16
0x FFFD_BCC0～ 0x FFFD_BF44	AC ハフマンテーブル No.1 レジスタ 1		Not applicable	WO	16
0x FFFD_C000	**ADC 関連レジスタ**	**ADC**			
0x FFFD_C000	ADC データレジスタ 0	ADCDT0	0x 0000	RO	16(/32)
0x FFFD_C004	ADC データレジスタ 1	ADCDT1	0x 0000	RO	16(/32)
0x FFFD_C008	ADC データレジスタ 2	ADCDT2	0x 0000	RO	16(/32)
0x FFFD_C00C	ADC データレジスタ 3	ADCDT3	0x 0000	RO	16(/32)
0x FFFD_C010	ADC データレジスタ 4	ADCDT4	0x 0000	RO	16(/32)
0x FFFD_C014	ADC データレジスタ 5	ADCDT5	0x 0000	RO	16(/32)
0x FFFD_C018	ADC データレジスタ 6	ADCDT6	0x 0000	RO	16(/32)
0x FFFD_C01C	ADC データレジスタ 7	ADCDT7	0x 0000	RO	16(/32)
0x FFFD_C020	ADC 制御レジスタ	ADCCTL	0x 00	RW	8(/16/32)
0x FFFD_C024	ADC フラグレジスタ	ADCFLG	0x 0000	RW	16(/32)
0x FFFD_D000	**SD Memory Card 制御関連レジスタ**	**SDC**			
別途					
0x FFFD_F000	**USB2.0 HS Device 制御関連レジスタ**	**USB**			
0x FFFD_F000	Main Interrupt Status	MainIntStat	0x 00	R/(W)	8(/16)
0x FFFD_F001	Reserved		-	-	
0x FFFD_F002	Reserved		-	-	8(/16)
0x FFFD_F003	CPU Interrupt Status	CPU_IntStat	0x 00	R/(W)	
0x FFFD_F004～ 0x FFFD_F005	Reserved		-	-	
0x FFFD_F006	Reserved		-	-	8(/16)
0x FFFD_F007	DMA Interrupt Status	DMA_IntStat	0x 00	R/(W)	
0x FFFD_F008～ 0x FFFD_F00F	Reserved		-	-	-
0x FFFD_F010	Main Interrupt Enable	MainIntEnb	0x 00	R/W	8(/16)
0x FFFD_F011	Reserved		-	-	
0x FFFD_F012	Reserved		-	-	8(/16)
0x FFFD_F013	CPU Interrupt Enable	CPU_IntEnb	0x 00	R/W	
0x FFFD_F014～ 0x FFFD_F015	Reserved		-	-	-
0x FFFD_F016	Reserved		-	-	8(/16)
0x FFFD_F017	DMA Interrupt Enable	DMA_IntEnb	0x 00	R/W	
0x FFFD_F018～ 0x FFFD_F01F	Reserved		-	-	-
0x FFFD_F020	Power Management Control 0	PM_Control0	0x 00	R/W	8(/16)
0x FFFD_F021	Power Management Control 1	PM_Control1	0x 00	R	
0x FFFD_F022	Wakeup Time Low(BE)	WakeupTim_L	0x 00	R/W	8(/16)
0x FFFD_F023	Wakeup Time High(BE)	WakeupTim_H	0x 00	R/W	
0x FFFD_F024～ 0x FFFD_F02F	Reserved		-	-	-
0x FFFD_F030	Reserved		-	-	8(/16)
0x FFFD_F031	Macro Reset	MacroReset	0x XX	W	
0x FFFD_F032	Reserved		-	-	8(/16)
0x FFFD_F033	Mode Protection	ModeProtect	0x 56	R/W	
0x FFFD_F034	Reserved		-	-	8(/16)
0x FFFD_F035	Macro Configuration 0	MacroConfig0	0x 41	R/W	

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_F036	Reserved		-	-	8(/16)
0x FFFD_F037	Macro Configuration 1	MacroConfig1	0x 06	R/W	
0x FFFD_F038	Reserved		-	-	-
0x FFFD_F039	Reserved		-	-	-
0x FFFD_F03A	Macro Type 0	MacroType0	0x 44	R	8(/16)
0x FFFD_F03B	Macro Type 1	MacroType1	0x 08	R	
0x FFFD_F03C	Macro Type 2	MacroType2	0x 02	R	8(/16)
0x FFFD_F03D	Macro Type 3	MacroType3	0x 30	R	
0x FFFD_F03E	FIFO Capacity Low(BE)	FIFO_CapacityL	0x 00	R	8(/16)
0x FFFD_F03F	FIFO Capacity High(BE)	FIFO_CapacityH	0x 12	R	
0x FFFD_F040	Reserved		-	-	8(/16)
0x FFFD_F041	DMA0 Configuration	DMA0_Config	0x 00	R/W	
0x FFFD_F042	DMA0 Control	DMA0_Control	0x 00	R/W	8(/16)-
0x FFFD_F043	Reserved		-	-	
0x FFFD_F044	DMA0 FIFO Remain Low(BE)	DMA0_RemainL	0x 00	R	8(/16)
0x FFFD_F045	DMA0 FIFO Remain High(BE)	DMA0_RemainH	0x 00	R	
0x FFFD_F046～ 0x FFFD_F047	Reserved		-	-	-
0x FFFD_F048	DMA0 Transfer Byte Counter High/Low(BE)	DMA0_Count_HL	0x 00	R/W	8(/16)
0x FFFD_F049	DMA0 Transfer Byte Counter High/High(BE)	DMA0_Count_HH	0x 00	R/W	
0x FFFD_F04A	DMA0 Transfer Byte Counter Low/Low(BE)	DMA0_Count_LL	0x 00	R/W	8(/16)
0x FFFD_F04B	DMA0 Transfer Byte Counter Low/High(BE)	DMA0_Count_LH	0x 00	R/W	
0x FFFD_F04C～ 0x FFFD_F04F	Reserved		-	-	-
0x FFFD_F050	Reserved		-	-	8(/16)
0x FFFD_F051	DMA1 Configuration	DMA1_Config	0x 00	R/W	
0x FFFD_F052	DMA1 Control	DMA1_Control	0x 00	R/W	8(/16)
0x FFFD_F053	Reserved		-	-	
0x FFFD_F054	DMA1 FIFO Remain Low(BE)	DMA1_RemainL	0x 00	R	8(/16)
0x FFFD_F055	DMA1 FIFO Remain High(BE)	DMA1_RemainH	0x 00	R	
0x FFFD_F056～ 0x FFFD_F057	Reserved		-	-	-
0x FFFD_F058	DMA1 Transfer Byte Counter High/Low(BE)	DMA1_Count_HL	0x 00	R/W	8(/16)
0x FFFD_F059	DMA1 Transfer Byte Counter High/High(BE)	DMA1_Count_HH	0x 00	R/W	
0x FFFD_F05A	DMA1 Transfer Byte Counter Low/Low(BE)	DMA1_Count_LL	0x 00	R/W	8(/16)
0x FFFD_F05B	DMA1 Transfer Byte Counter Low/High(BE)	DMA1_Count_LH	0x 00	R/W	
0x FFFD_F05C～ 0x FFFD_F05F	Reserved		-	-	-
0x FFFD_F060～ 0x FFFD_F07F	Reserved		-	-	-
0x FFFD_F080	Device Interrupt Status	DeviceIntStat	0x 00	R/W	8(/16)
0x FFFD_F081	EPr Interrupt Status	EPrIntStat	0x 00	R	
0x FFFD_F082	SIE Interrupt Status	SIE_IntStat	0x 00	R/(W)	8(/16)
0x FFFD_F083	Reserved		-	-	
0x FFFD_F084	FIFO Interrupt Status	FIFO_IntStat	0x 00	R/(W)	8(/16)
0x FFFD_F085	Bulk Interrupt Status	BulkIntStat	0x 00	R/(W)	

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_F086	Reserved		-	-	8(/16)
0x FFFD_F087	EP0 Interrupt Status	EP0IntStat	0x 00	R/(W)	
0x FFFD_F088	EPa Interrupt Status	EPaIntStat	0x 00	R/(W)	8(/16)
0x FFFD_F089	EPb Interrupt Status	EPbIntStat	0x 00	R/(W)	
0x FFFD_F08A	EPc Interrupt Status	EPcIntStat	0x 00	R/(W)	8(/16)
0x FFFD_F08B	EPd Interrupt Status	EPdIntStat	0x 00	R/(W)	
0x FFFD_F08C	EPe Interrupt Status	EPeIntStat	0x 00	R/(W)	8(/16)
0x FFFD_F08D	EPf Interrupt Status	EPfIntStat	0x 00	R/(W)	
0x FFFD_F08E	EPg Interrupt Status	EPgIntStat	0x 00	R/(W)	8(1/6)
0x FFFD_F08F	EPh Interrupt Status	EPhIntStat	0x 00	R/(W)	
0x FFFD_F090	Device Interrupt Enable	DeviceIntEnb	0x 00	R/W	8(/16)
0x FFFD_F091	EPr Interrupt Enable	EprIntEnb	0x 00	R/W	
0x FFFD_F092	SIE Interrupt Enable	SIE_IntEnb	0x 00	R/W	8(/16)
0x FFFD_F093	Reserved		-	-	
0x FFFD_F094	FIFO Interrupt Enable	FIFO_IntEnb	0x 00	R/W	8(/16)
0x FFFD_F095	Bulk Interrupt Enable	BulkIntEnb	0x 00	R/W	
0x FFFD_F096	Reserved		-	-	8(/16)
0x FFFD_F097	EP0 Interrupt Enable	EP0IntEnb	0x 00	R/W	
0x FFFD_F098	EPa Interrupt Enable	EPaIntEnb	0x 00	R/W	8(/16)
0x FFFD_F099	EPb Interrupt Enable	EPbIntEnb	0x 00	R/W	
0x FFFD_F09A	EPc Interrupt Enable	EPcIntEnb	0x 00	R/W	8(/16)
0x FFFD_F09B	EPd Interrupt Enable	EPdIntEnb	0x 00	R/W	
0x FFFD_F09C	EPe Interrupt Enable	EPeIntEnb	0x 00	R/W	8(/16)
0x FFFD_F09D	EPf Interrupt Enable	EPfIntEnb	0x 00	R/W	
0x FFFD_F09E	EPg Interrupt Enable	EPgIntEnb	0x 00	R/W	8(/16)
0x FFFD_F09F	EPh Interrupt Enable	EPhIntEnb	0x 00	R/W	
0x FFFD_F0A0	Reset DTM	ResetDTM	0x 01	R/W	8(/16)
0x FFFD_F0A1	Reserved		-	-	
0x FFFD_F0A2	Nego Control	NegoControl	0x 00	R/W	8(/16)
0x FFFD_F0A3	Reserved		-	-	
0x FFFD_F0A4	USB Status	USB_Status	0x XX	R/W	8(/16)
0x FFFD_F0A5	Xcvr Control	XcvrControl	0x 41	R/W	
0x FFFD_F0A6	USB Test	USB_Test	0x 00	R/W	8(/16)
0x FFFD_F0A7	Reserved		-	-	
0x FFFD_F0A8	Endpoint Control	EPnControl	0x XX	W	8(/16)
0x FFFD_F0A9	Endpoint Clear	EPrFIFO_Clr	0x XX	W	
0x FFFD_F0AA	Clear All EPn Join	ClrAllEPnJoin	0x XX	W	8(/16)
0x FFFD_F0AB	Reserved		-	-	
0x FFFD_F0AC	BulkOnly Control	BulkOnlyControl	0x 00	R/W	8(/16)
0x FFFD_F0AD	BulkOnly Configuration	BulkOnlyConfig	0x 00	R/W	
0x FFFD_F0AE	Reserved		-	-	-
0x FFFD_F0AF	Reserved		-	-	-
0x FFFD_F0B0	EP0 SETUP 0	EP0SETUP_0	0x 00	R	8(/16)
0x FFFD_F0B1	EP0 SETUP 1	EP0SETUP_1	0x 00	R	
0x FFFD_F0B2	EP0 SETUP 2	EP0SETUP_2	0x 00	R	8(/16)
0x FFFD_F0B3	EP0 SETUP 3	EP0SETUP_3	0x 00	R	
0x FFFD_F0B4	EP0 SETUP 4	EP0SETUP_4	0x 00	R	8(/16)
0x FFFD_F0B5	EP0 SETUP 5	EP0SETUP_5	0x 00	R	
0x FFFD_F0B6	EP0 SETUP 6	EP0SETUP_6	0x 00	R	8(/16)
0x FFFD_F0B7	EP0 SETUP 7	EP0SETUP_7	0x 00	R	
0x FFFD_F0B8	USB Address	USB_Address	0x 00	R/(W)	8(/16)
0x FFFD_F0B9	Reserved		-	-	

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_F0BA	SETUP Control	SETUP_Control	0x 00	R/W	8(/16)
0x FFFD_F0BB	Reserved		-	-	
0x FFFD_F0BC～ 0x FFFD_F0BD	Reserved		-	-	-
0x FFFD_F0BE	FrameNumber Low	FrameNumber_L	0x 00	R	8(/16)
0x FFFD_F0BF	FrameNumber High	FrameNumber_H	0x 80	R	
0x FFFD_F0C0	EP0 Max Packet Size	EP0MaxSize	0x 40	R/W	8(/16)
0x FFFD_F0C1	EP0 Control	EP0Control	0x 00	R/W	
0x FFFD_F0C2	EP0 Control IN	EP0ControlIN	0x 00	R/W	8(/16)
0x FFFD_F0C3	EP0 Control OUT	EP0ControlOUT	0x 00	R/W	
0x FFFD_F0C4	Reserved		-	-	8(/16)
0x FFFD_F0C5	EP0 Join	EP0Join	0x 00	-	
0x FFFD_F0C6～ 0x FFFD_F0CF	Reserved		-	-	-
0x FFFD_F0D0	EPa Max Packet Size Low	EPaMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F0D1	EPa Max Packet Size High	EPaMaxSize_H	0x 00	R/W	
0x FFFD_F0D2	EPa Configuration	EPaConfig	0x 00	R/W	8(/16)
0x FFFD_F0D3	Reserved		-	-	
0x FFFD_F0D4	EPa Control	EPaControl	0x 00	R/W	8(/16)
0x FFFD_F0D5	EPa Join	EPaJoin	0x 00	R/W	
0x FFFD_F0D6	Reserved		-	-	-
0x FFFD_F0D7	Reserved		-	-	-
0x FFFD_F0D8	EPb Max Packet Size Low	EPbMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F0D9	EPb Max Packet Size High	EPbMaxSize_H	0x 00	R/W	
0x FFFD_F0DA	EPb Configuration	EPbConfig	0x 00	R/W	8(/16)
0x FFFD_F0DB	Reserved		-	-	
0x FFFD_F0DC	EPb Control	EPbControl	0x 00	R/W	8(/16)
0x FFFD_F0DD	EPb Join	EPbJoin	0x 00	R/W	
0x FFFD_F0DE	Reserved		-	-	-
0x FFFD_F0DF	Reserved		-	-	-
0x FFFD_F0E0	EPc Max Packet Size Low	EPcMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F0E1	EPc Max Packet Size High	EPcMaxSize_H	0x 00	R/W	
0x FFFD_F0E2	EPc Configuration	EPcConfig	0x 00	R/W	8(/16)
0x FFFD_F0E3	Reserved		-	-	
0x FFFD_F0E4	EPc Control	EPcControl	0x 00	R/W	8(/16)
0x FFFD_F0E5	EPc Join	EPcJoin	0x 00	R/W	
0x FFFD_F0E6	Reserved		-	-	-
0x FFFD_F0E7	Reserved		-	-	-
0x FFFD_F0E8	EPd Max Packet Size Low	EPdMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F0E9	EPd Max Packet Size High	EPdMaxSize_H	0x 00	R/W	
0x FFFD_F0EA	EPd Configuration	EPdConfig	0x 00	R/W	8(/16)
0x FFFD_F0EB	Reserved		-	-	
0x FFFD_F0EC	EPd Control	EPdControl	0x 00	R/W	8(/16)
0x FFFD_F0ED	EPd Join	EPdJoin	0x 00	R/W	
0x FFFD_F0EE	Reserved		-	-	-
0x FFFD_F0EF	Reserved		-	-	-
0x FFFD_F0F0	EPe Max Packet Size Low	EPeMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F0F1	EPe Max Packet Size High	EPeMaxSize_H	0x 00	R/W	
0x FFFD_F0F2	EPe Configuration	EPeConfig	0x 00	R/W	8(/16)
0x FFFD_F0F3	Reserved		-	-	
0x FFFD_F0F4	EPe Control	EPeControl	0x 00	R/W	8(/16)
0x FFFD_F0F5	EPe Join	EPeJoin	0x 00	R/W	

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_F0F6	Reserved		-	-	-
0x FFFD_F0F7	Reserved		-	-	-
0x FFFD_F0F8	EPf Max Packet Size Low	EPfMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F0F9	EPf Max Packet Size High	EPfMaxSize_H	0x 00	R/W	
0x FFFD_F0FA	EPf Configuration	EPfConfig	0x 00	R/W	8(/16)
0x FFFD_F0FB	Reserved		-	-	
0x FFFD_F0FC	EPf Control	EPfControl	0x 00	R/W	8(/16)
0x FFFD_F0FD	EPf Join	EPfJoin	0x 00	R/W	
0x FFFD_F0FE	Reserved		-	-	-
0x FFFD_F0FF	Reserved		-	-	-
0x FFFD_F100	EPg Max Packet Size Low	EPgMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F101	EPg Max Packet Size High	EPgMaxSize_H	0x 00	R/W	
0x FFFD_F102	EPg Configuration	EPgConfig	0x 00	R/W	8(/16)
0x FFFD_F103	Reserved		-	-	
0x FFFD_F104	EPg Control	EPgControl	0x 00	R/W	8(/16)
0x FFFD_F105	EPg Join	EPgJoin	0x 00	R/W	
0x FFFD_F106	Reserved		-	-	-
0x FFFD_F107	Reserved		-	-	-
0x FFFD_F108	EPh Max Packet Size Low	EPhMaxSize_L	0x 00	R/W	8(/16)
0x FFFD_F109	EPh Max Packet Size High	EPhMaxSize_H	0x 00	R/W	
0x FFFD_F10A	EPh Configuration	EPhConfig	0x 00	R/W	8(/16)
0x FFFD_F10B	Reserved		-	-	
0x FFFD_F10C	EPh Control	EPhControl	0x 00	R/W	8(/16)
0x FFFD_F10D	EPh Join	EPhJoin	0x 00	R/W	
0x FFFD_F10E	Reserved		-	-	-
0x FFFD_F10F	Reserved		-	-	-
0x FFFD_F110	Endpoint a Start Address Low	EPaStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F111	Endpoint a Start Address High	EPaStartAdrs_H	0x 00	R/W	
0x FFFD_F112	Endpoint b Start Address Low	EPbStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F113	Endpoint b Start Address High	EPbStartAdrs_H	0x 00	R/W	
0x FFFD_F114	Endpoint c Start Address Low	EPcStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F115	Endpoint c Start Address High	EPcStartAdrs_H	0x 00	R/W	
0x FFFD_F116	Endpoint d Start Address Low	EPdStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F117	Endpoint d Start Address High	EPdStartAdrs_H	0x 00	R/W	
0x FFFD_F118	Endpoint e Start Address Low	EPeStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F119	Endpoint e Start Address High	EPeStartAdrs_H	0x 00	R/W	
0x FFFD_F11A	Endpoint f Start Address Low	EPfStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F11B	Endpoint f Start Address High	EPfStartAdrs_H	0x 00	R/W	
0x FFFD_F11C	Endpoint g Start Address Low	EPgStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F11D	Endpoint g Start Address High	EPgStartAdrs_H	0x 00	R/W	
0x FFFD_F11E	Endpoint h Start Address Low	EPhStartAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F11F	Endpoint h Start Address High	EPhStartAdrs_H	0x 00	R/W	
0x FFFD_F120	Endpoint n End Address Low	EP_EndAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F121	Endpoint n End Address High	EP_EndAdrs_H	0x 12	R/W	
0x FFFD_F122	Reserved		-	-	-
0x FFFD_F123	Reserved		-	-	-
0x FFFD_F124	Descriptor Address Low	DescAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F125	Descriptor Address High	DescAdrs_H	0x 00	R/W	
0x FFFD_F126	Descriptor Size High	DescSize_L	0x 00	R/W	8(/16)
0x FFFD_F127	Descriptor Size Low	DescSize_H	0x 00	R/W	
0x FFFD_F128	DMA0 FIFO Control	DMA0_FIFO_Control	0x 00	R/W	8(/16)

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_F129	Reserved		-	-	
0x FFFD_F12A	DMA1 FIFO Control	DMA1_FIFO_Control	0x 00	R/W	8(/16)
0x FFFD_F12B	Reserved		-	-	
0x FFFD_F12C～0x FFFD_F12F	Reserved		-	-	-
0x FFFD_F130	FIFO Read 0	FIFO_Rd_0	0x XX	R	8(/16)
0x FFFD_F131	FIFO Read 1	FIFO_Rd_1	0x XX	R	
0x FFFD_F132	FIFO Write 0	FIFO_Wr_0	0x XX	W	8(/16)
0x FFFD_F133	FIFO Write 1	FIFO_Wr_1	0x XX	W	
0x FFFD_F134	FIFO Read Remain Low	FIFO_RdRemain_L	0x 00	R	8(/16)
0x FFFD_F135	FIFO Read Remain High	FIFO_RdRemain_H	0x 00	R	
0x FFFD_F136	FIFO Write Remain Low	FIFO_WrRemain_L	0x 00	R	8(/16)
0x FFFD_F137	FIFO Write Remain High	FIFO_WrRemain_H	0x 00	R	
0x FFFD_F138	FIFO Byte Read	FIFO_ByteRd	0x XX	R	8
0x FFFD_F139～0x FFFD_F13F	Reserved		-	-	-
0x FFFD_F140	RAM Read Address Low	RAM_RdAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F141	RAM Read Address High	RAM_RdAdrs_H	0x 00	R/W	
0x FFFD_F142	RAM Read Control	RAM_RdControl	0x 00	R/W	8(/16)
0x FFFD_F143	RAM Read Counter	RAM_RdCount	0x 00	R/W	
0x FFFD_F144	RAM Write Address Low	RAM_WrAdrs_L	0x 00	R/W	8(/16)
0x FFFD_F145	RAM Write Address High	RAM_WrAdrs_H	0x 00	R/W	
0x FFFD_F146	RAM Write Door 0	RAM_WrDoor_0	0x XX	W	8(/16)
0x FFFD_F147	RAM Write Door 1	RAM_WrDoor_1	0x XX	W	
0x FFFD_F148～0x FFFD_F14F	Reserved		-	-	-
0x FFFD_F150	RAM Read 00	RAM_Rd_00	0x XX	R	8(/16)
0x FFFD_F151	RAM Read 01	RAM_Rd_01	0x XX	R	
0x FFFD_F152	RAM Read 02	RAM_Rd_02	0x XX	R	8(/16)
0x FFFD_F153	RAM Read 03	RAM_Rd_03	0x XX	R	
0x FFFD_F154	RAM Read 04	RAM_Rd_04	0x XX	R	8(/16)
0x FFFD_F155	RAM Read 05	RAM_Rd_05	0x XX	R	
0x FFFD_F156	RAM Read 06	RAM_Rd_06	0x XX	R	8(/16)
0x FFFD_F157	RAM Read 07	RAM_Rd_07	0x XX	R	
0x FFFD_F158	RAM Read 08	RAM_Rd_08	0x XX	R	8(/16)
0x FFFD_F159	RAM Read 09	RAM_Rd_09	0x XX	R	
0x FFFD_F15A	RAM Read 0A	RAM_Rd_0A	0x XX	R	8(/16)
0x FFFD_F15B	RAM Read 0B	RAM_Rd_0B	0x XX	R	
0x FFFD_F15C	RAM Read 0C	RAM_Rd_0C	0x XX	R	8(/16)
0x FFFD_F15D	RAM Read 0D	RAM_Rd_0D	0x XX	R	
0x FFFD_F15E	RAM Read 0E	RAM_Rd_0E	0x XX	R	8(/16)
0x FFFD_F15F	RAM Read 0F	RAM_Rd_0F	0x XX	R	
0x FFFD_F160	RAM Read 10	RAM_Rd_10	0x XX	R	8(/16)
0x FFFD_F161	RAM Read 11	RAM_Rd_11	0x XX	R	
0x FFFD_F162	RAM Read 12	RAM_Rd_12	0x XX	R	8(/16)
0x FFFD_F163	RAM Read 13	RAM_Rd_13	0x XX	R	
0x FFFD_F164	RAM Read 14	RAM_Rd_14	0x XX	R	8(/16)

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFD_F165	RAM Read 15	RAM_Rd_15	0x XX	R	
0x FFFD_F166	RAM Read 16	RAM_Rd_16	0x XX	R	8(/16)
0x FFFD_F167	RAM Read 17	RAM_Rd_17	0x XX	R	
0x FFFD_F168	RAM Read 18	RAM_Rd_18	0x XX	R	8(/16)
0x FFFD_F169	RAM Read 19	RAM_Rd_19	0x XX	R	
0x FFFD_F16A	RAM Read 1A	RAM_Rd_1A	0x XX	R	8(/16)
0x FFFD_F16B	RAM Read 1B	RAM_Rd_1B	0x XX	R	
0x FFFD_F16C	RAM Read 1C	RAM_Rd_1C	0x XX	R	8(/16)
0x FFFD_F16D	RAM Read 1D	RAM_Rd_1D	0x XX	R	
0x FFFD_F16E	RAM Read 1E	RAM_Rd_1E	0x XX	R	8(/16)
0x FFFD_F16F	RAM Read 1F	RAM_Rd_1F	0x XX	R	
0x FFFD_F170～ 0x FFFD_F2FF	Reserved		-	-	-
0x FFFE_0000	**APB ブリッジ関連レジスタ**	**APB**			
0x FFFE_0000	APB WAIT0 レジスタ	APBWAIT0	0x 0050_0500	R/W	32
0x FFFE_0004	APB WAIT1 レジスタ	APBWAIT1	0x 0000_0000	R/W	32
0x FFFE_0008	APB WAIT2 レジスタ	APBWAIT2	0x 0050_0000	R/W	32
0x FFFE_3000	**DMA コントローラ 1 関連レジスタ**	**DMAC1**			
0x FFFE_3000	DMA チャネル 0 ソースアドレスレジスタ	SAR0	0x XXXX_XXXX	R/W	32
0x FFFE_3004	DMA チャネル 0 デスティネーション アドレスレジスタ	DAR0	0x XXXX_XXXX	R/W	32
0x FFFE_3008	DMA チャネル 0 転送カウントレジスタ	TCR0	0x 00XX_XXXX	R/W	32
0x FFFE_300C	DMA チャネル 0 コントロールレジスタ	CTL0	0x 00XX_XXXX	R/W	32
0x FFFE_3010	DMA チャネル 1 ソースアドレスレジスタ	SAR1	0x XXXX_XXXX	R/W	32
0x FFFE_3014	DMA チャネル 1 デスティネーション アドレスレジスタ	DAR1	0x XXXX_XXXX	R/W	32
0x FFFE_3018	DMA チャネル 1 転送カウントレジスタ	TCR1	0x 00XX_XXXX	R/W	32
0x FFFE_301C	DMA チャネル 1 コントロールレジスタ	CTL1	0x 00XX_XXXX	R/W	32
0x FFFE_3020	DMA チャネル 2 ソースアドレスレジスタ	SAR2	0x XXXX_XXXX	R/W	32
0x FFFE_3024	DMA チャネル 2 デスティネーション アドレスレジスタ	DAR2	0x XXXX_XXXX	R/W	32
0x FFFE_3028	DMA チャネル 2 転送カウントレジスタ	TCR2	0x 00XX_XXXX	R/W	32
0x FFFE_302C	DMA チャネル 2 コントロールレジスタ	CTL2	0x 00XX_XXXX	R/W	32
0x FFFE_3030	DMA チャネル 3 ソースアドレスレジスタ	SAR3	0x XXXX_XXXX	R/W	32
0x FFFE_3034	DMA チャネル 3 デスティネーション アドレスレジスタ	DAR3	0x XXXX_XXXX	R/W	32
0x FFFE_3038	DMA チャネル 3 転送カウントレジスタ	TCR3	0x 00XX_XXXX	R/W	32
0x FFFE_303C	DMA チャネル 3 コントロールレジスタ	CTL3	0x 00XX_XXXX	R/W	32
0x FFFE_3060	DMA チャネル オペレーティング選択レジスタ	OPSR	0x 0000_0000	R/W	32
0x FFFE_6000	**CF インタフェース制御関連レジスタ**	**CF**			
0x FFFE_6000	CF Card Interface Control Register	CFCTL	0x 1000	(R/W)	16(/32)
0x FFFE_6004	CF Card Pin Status Register	CFPINSTS	0x 0XXX	RO	16(/32)
0x FFFE_6008	CF Card IRQ Source & Clear Register	CFINTRSTS	0x 0XXX	R/W	16(/32)
0x FFFE_600C	CF Card IRQ Enable Register	CFINTMSTS	0x 0000	R/W	16(/32)
0x FFFE_6010	CF Card IRQ Status Register	CFINTSTS	0x 0000	RO	16(/32)
0x FFFE_6014	CF Card MISC Register	CFMISC	0x 0000	R/W	16(/32)
0x FFFE_7000	**ARS 制御レジスタ**	**ARS**			
0x FFFE_7000	ARS コントロールレジスタ	ARSCTRL	0x0000_0080	R/W	32

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFE_7004	ARS エリア選択レジスタ	ARSASEL	0x0000_0000	R/W	32
0x FFFE_700C	ARS ステータスレジスタ	ARSSTAT	0x0000_0000	R/W	32
0x FFFE_7010	ARS 割り込み要因レジスタ	ARSINT	0x0000_0000	R/W	32
0x FFFE_7040	ARS 積算レジスタ 0	ARSADD0	0x0000_0000	RO	32
0x FFFE_7044	ARS 積算レジスタ 1	ARSADD1	0x0000_0000	RO	32
0x FFFE_7048	ARS 積算レジスタ 2	ARSADD2	0x0000_0000	RO	32
0x FFFE_704C	ARS 積算レジスタ 3	ARSADD3	0x0000_0000	RO	32
0x FFFE_7050	ARS 積算レジスタ 4	ARSADD4	0x0000_0000	RO	32
0x FFFE_7054	ARS 積算レジスタ 5	ARSADD5	0x0000_0000	RO	32
0x FFFE_7058	ARS 積算レジスタ 6	ARSADD6	0x0000_0000	RO	32
0x FFFE_705C	ARS 積算レジスタ 7	ARSADD7	0x0000_0000	RO	32
0x FFFE_7060	ARS 積算レジスタ 8	ARSADD8	0x0000_0000	RO	32
0x FFFE_7064	ARS 積算レジスタ 9	ARSADD9	0x0000_0000	RO	32
0x FFFE_7068	ARS 積算レジスタ 10	ARSADD10	0x0000_0000	RO	32
0x FFFE_706C	ARS 積算レジスタ 11	ARSADD11	0x0000_0000	RO	32
0x FFFE_7070	ARS 積算レジスタ 12	ARSADD12	0x0000_0000	RO	32
0x FFFE_7074	ARS 積算レジスタ 13	ARSADD13	0x0000_0000	RO	32
0x FFFE_7078	ARS 積算レジスタ 14	ARSADD14	0x0000_0000	RO	32
0x FFFE_707C	ARS 積算レジスタ 15	ARSADD15	0x0000_0000	RO	32
0x FFFE_8000	**カメラ 1 インタフェース関連レジスタ**	**CAM1**			
0x FFFE_8000	カメラ 1 クロック周期設定レジスタ		0x 0000	R/W	16
0x FFFE_8004	カメラ 1 信号設定レジスタ		0x 0000	R/W	16
0x FFFE_8008～ 0x FFFE_801C	予約		-	-	-
0x FFFE_8020	カメラ 1 モード設定レジスタ		0x 0000	R/W	16
0x FFFE_8024	カメラ 1 フレーム制御レジスタ		0x 0000	R/W	16
0x FFFE_8028	カメラ 1 コントロールレジスタ		0x 0000	WO	16
0x FFFE_802C	カメラ 1 ステータスレジスタ		0x 0004	RO	16
0x FFFE_8030～ 0x FFFE_805C	予約		-	-	-
0x FFFE_9000	**JPEG1 リサイズ関連レジスタ**	**RSZ1**			
0x FFFE_9060	グローバルリサイザ制御レジスタ		0x 0000	WO	16
0x FFFE_9064	キャプチャ制御ステートレジスタ		0x 0000	RO	16
0x FFFE_9068	キャプチャデータ設定レジスタ		0x 0000	R/W	16
0x FFFE_9070～ 0x FFFE_907C	予約レジスタ		-	-	-
0x FFFE_90C0	取り込みリサイズ制御レジスタ		0x 0000	R/W	16
0x FFFE_90C8	取り込みリサイズスタート X 座標レジスタ		0x 0000	R/W	16
0x FFFE_90CC	取り込みリサイズスタート Y 座標レジスタ		0x 0000	R/W	16
0x FFFE_90D0	取り込みリサイズエンド X 座標レジスタ		0x 027F	R/W	16
0x FFFE_90D4	取り込みリサイズエンド Y 座標レジスタ		0x 01DF	R/W	16
0x FFFE_90D8	取り込みリサイズ縮小率レジスタ		0x 8080	R/W	16
0x FFFE_90DC	取り込みリサイズ縮小モードレジスタ		0x 0000	R/W	16
0x FFFE_A000	**JPEG1 モジュールレジスタ**	**JCTL1**			
0x FFFE_A000	JPEG 制御レジスタ		0x 0000	R/W	16
0x FFFE_A004	JPEG ステータスフラグレジスタ		0x 8080	R/W	16
0x FFFE_A008	JPEG ロウステータスフラグレジスタ		0x 8080	RO	16
0x FFFE_A00C	JPEG 割り込み制御レジスタ		0x 0000	R/W	16
0x FFFE_A010	予約レジスタ		-	-	-
0x FFFE_A014	JPEG コーデックスタートストップ制御レジスタ		0x 0000	WO	16

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFE_A018～ 0x FFFE_A01C	予約レジスタ		-	-	-
0x FFFE_A020	ハフマンテーブル自動設定レジスタ		0x 0000	R/W	16
0x FFFE_A040	**JPEG1 FIFO 設定レジスタ**	**JFIFO1**			
0x FFFE_A040	JPEG FIFO 制御レジスタ		0x 0000	R/W	16
0x FFFE_A044	JPEG FIFO ステータスレジスタ		0x 8001	RO	16
0x FFFE_A048	JPEG FIFO サイズレジスタ		0x 003F	R/W	16
0x FFFE_A04C	JPEG FIFO リード/ライトポートレジスタ		0x 0000_0000	R/W	32
0x FFFE_A050～ 0x FFFE_A058	予約レジスタ		-	-	-
0x FFFE_A060	エンコードサイズリミットレジスタ 0		0x 0000	R/W	16
0x FFFE_A064	エンコードサイズリミットレジスタ 1		0x 0000	R/W	16
0x FFFE_A068	エンコードサイズ結果レジスタ 0		0x 0000	RO	16
0x FFFE_A06C	エンコードサイズ結果レジスタ 1		0x 0000	RO	16
0x FFFE_A070～ 0x FFFE_A078	予約レジスタ		-	-	-
0x FFFE_A080	**JPEG1 ラインバッファ設定レジスタ**	**JLB1**			
0x FFFE_A080	JPEG ラインバッファステータスフラグレジスタ		0x 0000	R/W	16
0x FFFE_A084	JPEG ラインバッファロウステータスフラグレジスタ		0x 0000	RO	16
0x FFFE_A088	JPEG ラインバッファカレントステータスフラグレジスタ		0x 0009	RO	16
0x FFFE_A08C	JPEG ラインバッファ割り込み制御レジスタ		0x 0000	R/W	16
0x FFFE_A090～ 0x FFFE_A09C	予約レジスタ		-	-	-
0x FFFE_A0A0	JPEG ラインバッファ水平ピクセル許容サイズレジスタ		0x 2800	R/W	16
0x FFFE_A0A4	JPEG ラインバッファメモリアドレスオフセットレジスタ		0x 0020	R/W	16
0x FFFE_A0A8～ 0x FFFE_A0BC	予約レジスタ		-	-	-
0x FFFE_A0C0	JPEG ラインバッファリード/ライトポートレジスタ		0x 0000	R/W	16
0x FFFE_B000	**JPEG1 コーデックレジスタ**	**JCOCEC1**			
0x FFFE_B000	動作モード設定レジスタ		0x 0000	R/W	16
0x FFFE_B004	コマンド設定レジスタ		Not applicable	WO	16
0x FFFE_B008	JPEG 動作ステータスレジスタ		0x 0000	RO	16
0x FFFE_B00C	量子化テーブル番号レジスタ		0x 0000	R/W	16
0x FFFE_B010	ハフマンテーブル番号レジスタ		0x 0000	R/W	16
0x FFFE_B014	DRI 設定レジスタ 0		0x 0000	R/W	16
0x FFFE_B018	DRI 設定レジスタ 1		0x 0000	R/W	16
0x FFFE_B01C	垂直ピクセルサイズレジスタ 0		0x 0000	R/W	16
0x FFFE_B020	垂直ピクセルサイズレジスタ 1		0x 0000	R/W	16
0x FFFE_B024	水平ピクセルサイズレジスタ 0		0x 0000	R/W	16
0x FFFE_B028	水平ピクセルサイズレジスタ 1		0x 0000	R/W	16
0x FFFE_B02C～ 0x FFFE_B034	予約レジスタ		-	-	-
0x FFFE_B038	RST マーカ動作設定レジスタ		0x 0000	R/W	16
0x FFFE_B03C	RST マーカ動作ステータスレジスタ		0x 0000	RO	16
0x FFFE_B040～ 0x FFFE_B0CC	挿入マーカデータレジスタ		0x 00FF	R/W	16

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFE_B400～ 0x FFFE_B4FC	量子化テーブル No.0 レジスタ		Not applicable	R/W	16
0x FFFE_B500～ 0x FFFE_B5FC	量子化テーブル No.1 レジスタ		Not applicable	R/W	16
0x FFFE_B800～ 0x FFFE_B83C	DC ハフマンテーブル No.0 レジスタ 0		Not applicable	WO	16
0x FFFE_B840～ 0x FFFE_B86C	DC ハフマンテーブル No.0 レジスタ 1		Not applicable	WO	16
0x FFFE_B880～ 0x FFFE_B8BC	AC ハフマンテーブル No.0 レジスタ 0		Not applicable	WO	16
0x FFFE_B8C0～ 0x FFFE_BB44	AC ハフマンテーブル No.0 レジスタ 1		Not applicable	WO	16
0x FFFE_BC00～ 0x FFFE_BC3C	DC ハフマンテーブル No.1 レジスタ 0		Not applicable	WO	16
0x FFFE_BC40～ 0x FFFE_BC6C	DC ハフマンテーブル No.1 レジスタ 1		Not applicable	WO	16
0x FFFE_BC80～ 0x FFFE_BCBC	AC ハフマンテーブル No.1 レジスタ 0		Not applicable	WO	16
0x FFFE_BCC0～ 0x FFFE_BF44	AC ハフマンテーブル No.1 レジスタ 1		Not applicable	WO	16
0x FFFE_C000	**JPEG_DMAC 関連レジスタ**	**JDMA**			
0x FFFE_C000	DMA チャネル 0 JPEG ソースアドレスレジスタ	JSAR0	0x XXXX_XXXX	R/W	32
0x FFFE_C004	DMA チャネル 0 JPEG デスティネーションアドレスレジスタ	JDAR0	0x XXXX_XXXX	R/W	32
0x FFFE_C008	DMA チャネル 0 JPEG 転送カウントレジスタ	JTCR0	0x 0000_0000	R/W	32
0x FFFE_C00C	DMA チャネル 0 JPEG コントロールレジスタ	JCTL0	0x 0000_0000	R/W	32
0x FFFE_C010	DMA チャネル 0 JPEG ブロックカウントレジスタ	JBCR0	0x 00XX_XXXX	R/W	32
0x FFFE_C014	DMA チャネル 0 JPEG デスティネーションオフセット アドレスレジスタ	JOFR0	0x 0000_0000	R/W	32
0x FFFE_C018	DMA チャネル 0 JPEG ブロックエンドカウントレジスタ	JBER0	0x 00XX_XXXX	R/W	32
0x FFFE_C020	DMA チャネル 1 JPEG ソースアドレスレジスタ	JSAR1	0x XXXX_XXXX	R/W	32
0x FFFE_C024	DMA チャネル 1 JPEG デスティネーションアドレスレジスタ	JDAR1	0x XXXX_XXXX	R/W	32
0x FFFE_C028	DMA チャネル 1 JPEG 転送カウントレジスタ	JTCR1	0x 0000_0000	R/W	32
0x FFFE_C02C	DMA チャネル 1 JPEG コントロールレジスタ	JCTL1	0x 0000_0000	R/W	32
0x FFFE_C030	DMA チャネル 1 JPEG ブロックカウントレジスタ	JBCR1	0x 00XX_XXXX	R/W	32
0x FFFE_C034	DMA チャネル 1 JPEG デスティネーションオフセット アドレスレジスタ	JOFR1	0x 0000_0000	R/W	32
0x FFFE_C038	DMA チャネル 1 JPEG ブロックエンドカウントレジスタ	JBER1	0x 00XX_XXXX	R/W	32
0x FFFE_C040	DMA チャネル JPEG FIFO データ選択モードレジスタ	JFSM	0x 0000_0000	R/W	32

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFE_C048	DMA チャネル JPEG 拡張レジスタ	JHID	0x 0000_0000	R/W	32
0x FFFE_D000	**I2C 関連レジスタ**	**I2C**			
0x FFFE_D000	I2C 送信データレジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D004	I2C 受信データレジスタ		0000 0000 b	RO	8 (16/32)
0x FFFE_D008	I2C コントロールレジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D00C	I2C バスステータスレジスタ		00xx 0000 b	RO	8 (16/32)
0x FFFE_D010	I2C エラーステータスレジスタ		0000 0000 b	RO	8 (16/32)
0x FFFE_D014	I2C 割り込みコントロール/ステータスレジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D018	I2C–BUS サンプルクロック分周設定レジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D01C	I2C SCL クロック分周設定レジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D020	I2C I/O コントロールレジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D024	I2C DMA モードレジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D028	I2C DMA カウンタ値(LSB)レジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D02C	I2C DMA カウンタ値(MSB)レジスタ		0000 0000 b	R/W	8 (16/32)
0x FFFE_D030	I2C DMA ステータスレジスタ		0000 1000 b	RO	8 (16/32)
0x FFFE_D034～ 0x FFFE_D038	予約		-	-	-
0x FFFE_E000	**I2S 関連レジスタ**	**I2S**			
0x FFFE_E000	I2S0 制御レジスタ		0x 0000	R/W	16(/32)
0x FFFE_E004	I2S0 クロック分周レジスタ		0x 0000	R/W	16(/32)
0x FFFE_E008	I2S0 送受信ポートレジスタ		-	R/W	8(/16/32)
0x FFFE_E010	I2S0 割り込みステータスレジスタ		0x 0000	R/W	16(/32)
0x FFFE_E014	I2S0 割り込みロウステータスレジスタ		0x 0009	RO	16(/32)
0x FFFE_E018	I2S0 割り込みイネーブルレジスタ		0x 0000	R/W	16(/32)
0x FFFE_E01C	I2S0 カレントステータスレジスタ		0x 0009	RO	16(/32)
0x FFFE_E040	I2S1 制御レジスタ		0x 0000	R/W	16(/32)
0x FFFE_E044	I2S1 クロック分周レジスタ		0x 0000	R/W	16(/32)
0x FFFE_E048	I2S1 送受信ポートレジスタ		-	R/W	8(/16/32)
0x FFFE_E050	I2S1 割り込みステータスレジスタ		0x 0000	R/W	16(/32)
0x FFFE_E054	I2S1 割り込みロウステータスレジスタ		0x 0009	RO	16(/32)
0x FFFE_E058	I2S1 割り込みイネーブルレジスタ		0x 0000	R/W	16(/32)
0x FFFE_E05C	I2S1 カレントステータスレジスタ		0x 0009	RO	16(/32)
0x FFFF_1000	**GPIO 関連レジスタ**	**GPIO**			
0x FFFF_1000	GPIOA データレジスタ	GPIOA_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_1004	GPIOA 端子機能レジスタ	GPIOA_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1008	GPIOB データレジスタ	GPIOB_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_100C	GPIOB 端子機能レジスタ	GPIOB_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1010	GPIOC データレジスタ	GPIOC_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_1014	GPIOC 端子機能レジスタ	GPIOC_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1018	GPIOD データレジスタ	GPIOD_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_101C	GPIOD 端子機能レジスタ	GPIOD_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1020	GPIOE データレジスタ	GPIOE_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_1024	GPIOE 端子機能レジスタ	GPIOE_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1028	GPIOF データレジスタ	GPIOF_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_102C	GPIOF 端子機能レジスタ	GPIOF_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1030	GPIOG データレジスタ	GPIOG_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_1034	GPIOG 端子機能レジスタ	GPIOG_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1038	GPIOH データレジスタ	GPIOH_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_103C	GPIOH 端子機能レジスタ	GPIOH_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1040	GPIOI データレジスタ	GPIOI_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_1044	GPIOI 端子機能レジスタ	GPIOI_FNC	0x 0000_0000	R/W	16 (/32)

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称		レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFF_1048	GPIOJ データレジスタ		GPIOJ_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_104C	GPIOJ 端子機能レジスタ		GPIOJ_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1050	GPIOK データレジスタ		GPIOK_DATA	0x 0000_0000	R/W	8 (/16/32)
0x FFFF_1054	GPIOK 端子機能レジスタ		GPIOK_FNC	0x 0000_0000	R/W	16 (/32)
0x FFFF_1060	GPIOA&B IRQ タイプレジスタ		GPIOAB_ITYP	0x 0000_0000	R/W	16 (/32)
0x FFFF_1064	GPIOA&B IRQ 極性レジスタ		GPIOAB_IPOL	0x 0000_0000	R/W	16 (/32)
0x FFFF_1068	GPIOA&B IRQ イネーブルレジスタ		GPIOAB_IEN	0x 0000_0000	R/W	16 (/32)
0x FFFF_106C	GPIOA&B IRQ ステータス&クリアレジスタ		GPIOAB_ISTS	0x 0000_0000	R/W	16 (/32)
0x FFFF_2000	**SPI 関連レジスタ**		**SPI**			
0x FFFF_2000	SPI 受信データレジスタ			0x 0000_0000	RO	32
0x FFFF_2004	SPI 送信データレジスタ			0x 0000_0000	R/W	32
0x FFFF_2008	SPI 制御レジスタ 1			0x 0000_0000	R/W	32
0x FFFF_200C	SPI 制御レジスタ 2			0x 0000_0000	R/W	32
0x FFFF_2010	SPI ウェイトレジスタ			0x 0000_0000	R/W	32
0x FFFF_2014	SPI ステータスレジスタ			0x 0000_0010	RO	32
0x FFFF_2018	SPI 割り込み制御レジスタ			0x 0000_0000	R/W	32
0x FFFF_5000	**DLAB**	**UART 関連レジスタ**	**UART**			
0x FFFF_5000	0	受信バッファレジスタ	RBR	0x 00	RO	8 (/16/32)
0x FFFF_5000	0	送信ホールディングレジスタ	THR	-	WO	8 (/16/32)
0x FFFF_5000	1	デバイザラッチ LSB レジスタ	DLL	0x 00	R/W	8 (/16/32)
0x FFFF_5004	0	割り込みイネーブルレジスタ	IER	0x 00	R/W	8 (/16/32)
0x FFFF_5004	1	デバイザラッチ MSB レジスタ	DLM	0x 00	R/W	8 (/16/32)
0x FFFF_5008	割り込み識別レジスタ		IIR	0x 01	RO	8 (/16/32)
0x FFFF_5008	FIFO 制御レジスタ		FCR	-	WO	8 (/16/32)
0x FFFF_500C	ライン制御レジスタ		LCR	0x 00	R/W	8 (/16/32)
0x FFFF_5010	モデム制御レジスタ		MCR	0x 00	R/W	8 (/16/32)
0x FFFF_5014	ラインステータスレジスタ		LSR	0x 60	RO	8 (/16/32)
0x FFFF_5018	モデムステータスレジスタ		MSR	0x EX	RO	8 (/16/32)
0x FFFF_501C	スクラッチレジスタ		SCR	0x 00	R/W	8 (/16/32)
0x FFFF_5020	テスト 0 レジスタ		T0	0x 00	R/W	8 (/16/32)
0x FFFF_5024	テスト 1 レジスタ		T1	0x 00	R/W	8 (/16/32)
0x FFFF_5028	テストステータス 0 レジスタ		TS0	-	RO	8 (/16/32)
0x FFFF_502C	テストステータス 1 レジスタ		TS1	0x 01	RO	8 (/16/32)
0x FFFF_5030	テストステータス 2 レジスタ		TS2	0x 0F	RO	8 (/16/32)
0x FFFF_503C	テストステータス 3 レジスタ		TS3	0x 02	RO	8 (/16/32)
0x FFFF_8000	**RTC 関連レジスタ**		**RTC**			
0x FFFF_8000	RTC ラン／ストップ制御レジスタ			xxx- --xx b	R/W	8(/16/32)
0x FFFF_8004	RTC 割り込みレジスタ			0x xxxx	R/W	16(/32)
0x FFFF_8008	RTC タイマ分周レジスタ			xxxx xxxx b	R/(W)	8(/16/32)
0x FFFF_800C	RTC 秒カウンタレジスタ			--xx xxxx b	R/W	8(/16/32)
0x FFFF_8010	RTC 分カウンタレジスタ			--xx xxxx b	R/W	8(/16/32)
0x FFFF_8014	RTC 時間カウンタレジスタ			---x xxxx b	R/W	8(/16/32)
0x FFFF_8018	RTC 日カウンタレジスタ			---x xxxx b	R/W	8(/16/32)
0x FFFF_801C	RTC 月カウンタレジスタ			---- xxxx b	R/W	8(/16/32)
0x FFFF_8020	RTC 年カウンタレジスタ			-xxx xxxx b	R/W	8(/16/32)
0x FFFF_8024	RTC アラーム分コンペアレジスタ			--xx xxxx b	R/W	8(/16/32)
0x FFFF_8028	RTC アラーム時間コンペアレジスタ			---x xxxx b	R/W	8(/16/32)
0x FFFF_802C	RTC アラーム日コンペアレジスタ			---x xxxx b	R/W	8(/16/32)
0x FFFF_8030	RTC アラーム月コンペアレジスタ			---- xxxx b	R/W	8(/16/32)
0x FFFF_8034	RTC アラーム年コンペアレジスタ			-xxx xxxx b	R/W	8(/16/32)
0x FFFF_8040	RTC テストレジスタ			---x xxxx b	R/W	8(/16/32)
0x FFFF_8044	RTC プリスケーラレジスタ			-xxx xxxx b	R/(W)	8(/16/32)

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFF_8048	RTC テストクロックレジスタ		xxxx xxxx b	WO	8(/16/32)
0x FFFF_8060	RTC RAM0		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_8064	RTC RAM1		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_8068	RTC RAM2		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_806C	RTC RAM3		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_8070	RTC RAM4		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_8074	RTC RAM5		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_8078	RTC RAM6		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_807C	RTC RAM7		xxxx xxxx b	R/W	8(/16/32)
0x FFFF_9000	**DMA コントローラ 2 関連レジスタ**	**DMAC2**			
0x FFFF_9000	DMA チャネル 0 ソースアドレスレジスタ	SAR0	0x XXXX_XXXX	R/W	32
0x FFFF_9004	DMA チャネル 0 デスティネーション アドレスレジスタ	DAR0	0x XXXX_XXXX	R/W	32
0x FFFF_9008	DMA チャネル 0 転送カウントレジスタ	TCR0	0x 00XX_XXXX	R/W	32
0x FFFF_900C	DMA チャネル 0 コントロールレジスタ	CTL0	0x 0000_0000	R/W	32
0x FFFF_9010	DMA チャネル 1 ソースアドレスレジスタ	SAR1	0x XXXX_XXXX	R/W	32
0x FFFF_9014	DMA チャネル 1 デスティネーション アドレスレジスタ	DAR1	0x XXXX_XXXX	R/W	32
0x FFFF_9018	DMA チャネル 1 転送カウントレジスタ	TCR1	0x 00XX_XXXX	R/W	32
0x FFFF_901C	DMA チャネル 1 コントロールレジスタ	CTL1	0x 0000_0000	R/W	32
0x FFFF_9020	DMA チャネル 2 ソースアドレスレジスタ	SAR2	0x XXXX_XXXX	R/W	32
0x FFFF_9024	DMA チャネル 2 デスティネーション アドレスレジスタ	DAR2	0x XXXX_XXXX	R/W	32
0x FFFF_9028	DMA チャネル 2 転送カウントレジスタ	TCR2	0x 00XX_XXXX	R/W	32
0x FFFF_902C	DMA チャネル 2 コントロールレジスタ	CTL2	0x 0000_0000	R/W	32
0x FFFF_9030	DMA チャネル 3 ソースアドレスレジスタ	SAR3	0x XXXX_XXXX	R/W	32
0x FFFF_9034	DMA チャネル 3 デスティネーション アドレスレジスタ	DAR3	0x XXXX_XXXX	R/W	32
0x FFFF_9038	DMA チャネル 3 転送カウントレジスタ	TCR3	0x 00XX_XXXX	R/W	32
0x FFFF_903C	DMA チャネル 3 コントロールレジスタ	CTL3	0x 0000_0000	R/W	32
0x FFFF_9060	DMA チャネル オペレーティング選択レジスタ	OPSR	0x 0000_0000	R/W	32
0x FFFF_9064	DMA チャネル MISC レジスタ	MISC	0x 0000_0000	R/W	32
0x FFFF_9070	DMA チャネル 転送終了コントロールレジスタ	TECL	0x 0000_0000	R/W	32
0x FFFF_A000	**メモリコントローラ関連レジスタ**	**MEMC**			
0x FFFF_A020	SRAM デバイス 0 タイミングレジスタ	RAMTMG0	0x 0000_1C7F	R/W	32
0x FFFF_A024	SRAM デバイス 0 制御レジスタ	RAMCNTL0	0x 0000_0001	R/W	32
0x FFFF_A030	SRAM デバイス 1 タイミングレジスタ	RAMTMG1	0x 0000_1C7F	R/W	32
0x FFFF_A034	SRAM デバイス 1 制御レジスタ	RAMCNTL1	0x 0000_0001	R/W	32
0x FFFF_A040	SRAM デバイス 2 タイミングレジスタ	RAMTMG2	0x 0000_1C7F	R/W	32
0x FFFF_A044	SRAM デバイス 2 制御レジスタ	RAMCNTL2	0x 0000_0001	R/W	32
0x FFFF_A050	SRAM デバイス 3 タイミングレジスタ	RAMTMG3	0x 0000_1C7F	R/W	32
0x FFFF_A054	SRAM デバイス 3 制御レジスタ	RAMCNTL3	0x 0000_0001	R/W	32
0x FFFF_A060	SDRAM モードレジスタ	SDMR	0x 0000_0032	R/W	16/32
0x FFFF_A064	予約	-	-	-/-	-
0x FFFF_A068	予約	-	-	-/-	-
0x FFFF_A070	SDRAM 設定レジスタ	SDCNFG	0x 0600_C700	R/W	32
0x FFFF_A074	SDRAM 詳細設定レジスタ	SDADVCNFG	0x 000F_0300	R/W	32
0x FFFF_A080	初期化制御レジスタ	SDINIT	0x 0000_0000	R/W	16/32

Appendix 1　S2S65A00 内部レジスター覧

アドレス (h)	レジスタ名称	レジスタ略号	初期値[*1] (h)	R/W	データアクセスサイズ[*2] (bit)
0x FFFF_A090	SDRAM リフレッシュタイマレジスタ	SDREF	0x 0000_00A0	R/W	16/32
0x FFFF_A0A0	SDRAM ステータスレジスタ	SDSTAT	0x 0000_0202	RO	32
0x FFFF_B000	**タイマ A 関連レジスタ**	**TIM**			
0x FFFF_B000	タイマ A 0　ロードレジスタ	TMA0LD	0x 0000	R/W	16 (/32)
0x FFFF_B004	タイマ A 0　カウントレジスタ	TMA0CNT	0x 0000	RO	16 (/32)
0x FFFF_B008	タイマ A 0　制御レジスタ	TMA0CTRL	0x 0000	(R/W)	16 (/32)
0x FFFF_B00C	タイマ A 0　IRQ フラグクリアレジスタ	TMA0IRQ	-	WO	8 (/16/32)
0x FFFF_B010	タイマ A 0　ポート出力制御レジスタ	TMA0POUT	0x 0000	(R/W)	8 (/16/32)
0x FFFF_B020	タイマ A 1　ロードレジスタ	TMA1LD	0x 0000	R/W	16 (/32)
0x FFFF_B024	タイマ A 1　カウントレジスタ	TMA1CNT	0x 0000	RO	16 (/32)
0x FFFF_B028	タイマ A 1　制御レジスタ	TMA1CTRL	0x 0000	(R/W)	16 (/32)
0x FFFF_B02C	タイマ A 1　IRQ フラグクリアレジスタ	TMA1IRQ	-	WO	8 (/16/32)
0x FFFF_B030	タイマ A 1　ポート出力制御レジスタ	TMA1POUT	0x 0000	(R/W)	8 (/16/32)
0x FFFF_B040	タイマ A 2　ロードレジスタ	TMA2LD	0x 0000	R/W	16 (/32)
0x FFFF_B044	タイマ A 2　カウントレジスタ	TMA2CNT	0x 0000	RO	16 (/32)
0x FFFF_B048	タイマ A 2　制御レジスタ	TMA2CTRL	0x 0000	(R/W)	16 (/32)
0x FFFF_B04C	タイマ A 2　IRQ フラグクリアレジスタ	TMA2IRQ	-	WO	8 (/16/32)
0x FFFF_B050	タイマ A 2　ポート出力制御レジスタ	TMA2POUT	0x 0000	(R/W)	8 (/16/32)
0x FFFF_B060～ 0x FFFF_B09C	予約	-	-	-	-
0x FFFF_B0A0	プリスケーラ 0　制御レジスタ	PS0CTRL	0x 0000	(R/W)	16 (/32)
0x FFFF_B0A4	プリスケーラ 1　制御レジスタ	PS1CTRL	0x 0000	(R/W)	16 (/32)
0x FFFF_B0B0	タイマ A　IRQ ステータスレジスタ	TMAIRQSTS	0x 0000	RO	8 (/16/32)
0x FFFF_C000	**WDT 関連レジスタ**	**WDT**			
0x FFFF_C000	ウォッチドッグタイマロードレジスタ		0x 0000_FFFF	R/W	16 (/32)
0x FFFF_C004	ウォッチドッグタイマカウントレジスタ		0x 0000_FFFF	RO	16 (/32)
0x FFFF_C008	ウォッチドッグタイマ制御レジスタ		0x 0000_0000	R/W	16 (/32)
0x FFFF_D000	**システムコントローラ関連レジスタ**	**SYS**			
0x FFFF_D000	Chip ID Register	CHIPID	0x 065A_000X	RO	32
0x FFFF_D004	Chip Configuration Register	CHIPCFG	0x 0000_XXXX	RO	16 (/32)
0x FFFF_D008	PLL Setting Register 1	PLLSET1	0x 0421_84AE	R/W	32
0x FFFF_D00C	PLL Setting Register 2	PLLSET2	0x 0000_0000	(R/W)	16 (/32)
0x FFFF_D010	HALT Mode Clock Control Register	HALTMODE	0x 0000_0000	R/W	16 (/32)
0x FFFF_D014	IO Clock Control Register	IOCLKCTL	0x 0000_0000	R/W	16 (/32)
0x FFFF_D018	Clock Select Register	CLK_SEL	0x 0000_0000	R/W	16 (/32)
0x FFFF_D01C	HALT Control Register	HALTCTL	-	WO	16 (/32)
0x FFFF_D020	Memory Remap Register	REMAP	0x 0000_0000	R/W	16 (/32)
0x FFFF_D024	Software Reset Register	SOFTRST	-	WO	32
0x FFFF_D028	UART1 Clock Divider Register	UART1DIV	0x 0000_0000	R/W	16 (/32)
0x FFFF_D02C	UART2 Clock Divider Register	UART2DIV	0x 0000_0000	R/W	16 (/32)
0x FFFF_D030	UART3 Clock Divider Register	UART3DIV	0x 0000_0000	R/W	16 (/32)
0x FFFF_D034	Timer-B Clock Select Register	TIMBCKSEL	0x 0000_0000	R/W	16(/32)
0x FFFF_D040	MD Bus Pull-down Control Register	MDPLDCTL	0x 0000_0000	R/W	16 (/32)
0x FFFF_D044	SDD Bus Pull-down Control Register	SDDPLDCTL	0x 0000_0000	R/W	16(/32)
0x FFFF_D048	GPIOE Resistor Control Register	PORTERCTL	0x 0000_0000	R/W	16 (/32)
0x FFFF_D04C	GPIOF Resistor Control Register	PORTFRCTL	0x 0000_0000	R/W	16 (/32)
0x FFFF_D050	GPIOG Resistor Control Register	PORTGRCTL	0x 0000_0000	R/W	16 (/32)
0x FFFF_D054	GPIOH Resistor Control Register	PORTHRCTL	0x 0000_0000	R/W	16(/32)
0x FFFF_D058	GPIOI Resistor Control Register	PORTIRCTL	0x 0000_0000	R/W	16(/32)
0x FFFF_D05C	GPIOJ Resistor Control Register	PORTJRCTL	0x 0000_0000	R/W	16(/32)
0x FFFF_D060	GPIOK Resistor Control Register	PORTKRCTL	0x 0000_0000	R/W	16(/32)
0x FFFF_D064	Internal TEST Mode Register	ITESTM	0x 0000_0000	R/W	32

アドレス (h)	レジスタ名称	レジスタ略号	初期値*1 (h)	R/W	データアクセスサイズ*2 (bit)
0x FFFF_D068	Embedded Memory Control Register	EMBMEMCTL	0x 0000_0000	R/W	16 (/32)
0x FFFF_D06C	MISC Register	MISC	0x 0000_0000	R/W	32
0x FFFF_F000	**割り込みコントローラ関連レジスタ**	**INT**			
0x FFFF_F000	IRQ ステータスレジスタ		0x 0000_0000	RO	32
0x FFFF_F004	IRQ マスク前ステータスレジスタ		0x 0000_0000	RO	32
0x FFFF_F008	IRQ イネーブルレジスタ		0x 0000_0000	R/W	32
0x FFFF_F00C	IRQ イネーブルクリアレジスタ		0x 0000_0000	WO	32
0x FFFF_F010	ソフトウェア IRQ レジスタ		0x 0000_0000	WO	32
0x FFFF_F020	IRQxx ステータスレジスタ		0x 0000_0000	RO	32
0x FFFF_F024	IRQxx マスク前ステータスレジスタ		0x 0000_0000	RO	32
0x FFFF_F028	IRQxx イネーブルレジスタ		0x 0000_0000	R/W	32
0x FFFF_F02C	IRQxx イネーブルクリアレジスタ		0x 0000_0000	WO	32
0x FFFF_F080	IRQ レベルレジスタ		0x 0000_0000	R/W	32
0x FFFF_F084	IRQ 極性レジスタ		0x FFFF_FFFF	R/W	32
0x FFFF_F088	IRQ トリガリセットレジスタ		0x 0000_0000	WO	32
0x FFFF_F0A0	IRQxx レベルレジスタ		0x 0000_0000	R/W	32
0x FFFF_F0A4	IRQxx 極性レジスタ		0x 0000_0FFF	R/W	32
0x FFFF_F0A8	IRQxx トリガリセットレジスタ		0x 0000_0000	WO	32
0x FFFF_F100	FIQ ステータスレジスタ		0x 0000_0000	RO	32
0x FFFF_F104	FIQ マスク前ステータスレジスタ		0x 0000_0000	RO	32
0x FFFF_F108	FIQ イネーブルレジスタ		0x 0000_0000	R/W	32
0x FFFF_F10C	FIQ イネーブルクリアレジスタ		0x 0000_0000	WO	32
0x FFFF_F180	FIQ レベルレジスタ		0x 0000_0000	R/W	32
0x FFFF_F184	FIQ 極性レジスタ		0x 0000_0003	R/W	32
0x FFFF_F188	FIQ トリガリセットレジスタ		0x 0000_0000	WO	32

*1： 初期値は 16 進数表示（h：hexadecimal）になっていますが、末尾に“b”の表示がある場合は、2 進数（b：binary）を表しています。
また、“X”は不定値(h)を、“x”は不定値(b)を表しています。

*2： データアクセスサイズはレジスタにアクセスするサイズをビットで表しています。8 (/16/32) は、通常は 8 ビットアクセスで使用しますが、16 ビットまたは 32 ビットデータアクセス命令を使用している場合は 16 ビットまたは 32 ビットでもアクセス可能です。同様に 16 (/32) は、通常は 16 ビットアクセスですが、32 ビットでもアクセス可能です。これらの場合、下位ビットのみ有効な値として使用してください。

Appendix 2　USB Device コントローラ

A2.1　Scope

本仕様書は、セイコーエプソン㈱製 S2S65A00*に内蔵されている USB HS Device Controller の機能仕様について説明します。

A2.2　Overview

S2S65A00*には、セイコーエプソン製の USB コントローラ「S1R72V05*」のデバイス機能と互換性のある USB コントローラを搭載しています。

- HS（High Speed＝480Mbps）及び FS（Full Speed＝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（外付け保護回路は必要）
- コントロール/バルク/インタラプト転送をサポート
- コントロール転送用（EP0）、Bulk 転送専用 1 本、Bulk/Interrupt 転送用 7 本の Endpoint をサポート
- 4.0KB の Endpoint 用 FIFO
- 2ch. DMA I/F（DMAC1 と組み合わせて使用することが可能）
- 12MHz / 24MHz の水晶発振子によるクロック入力（帰還抵抗 1MΩ 内蔵）

A2.3　Block Diagram

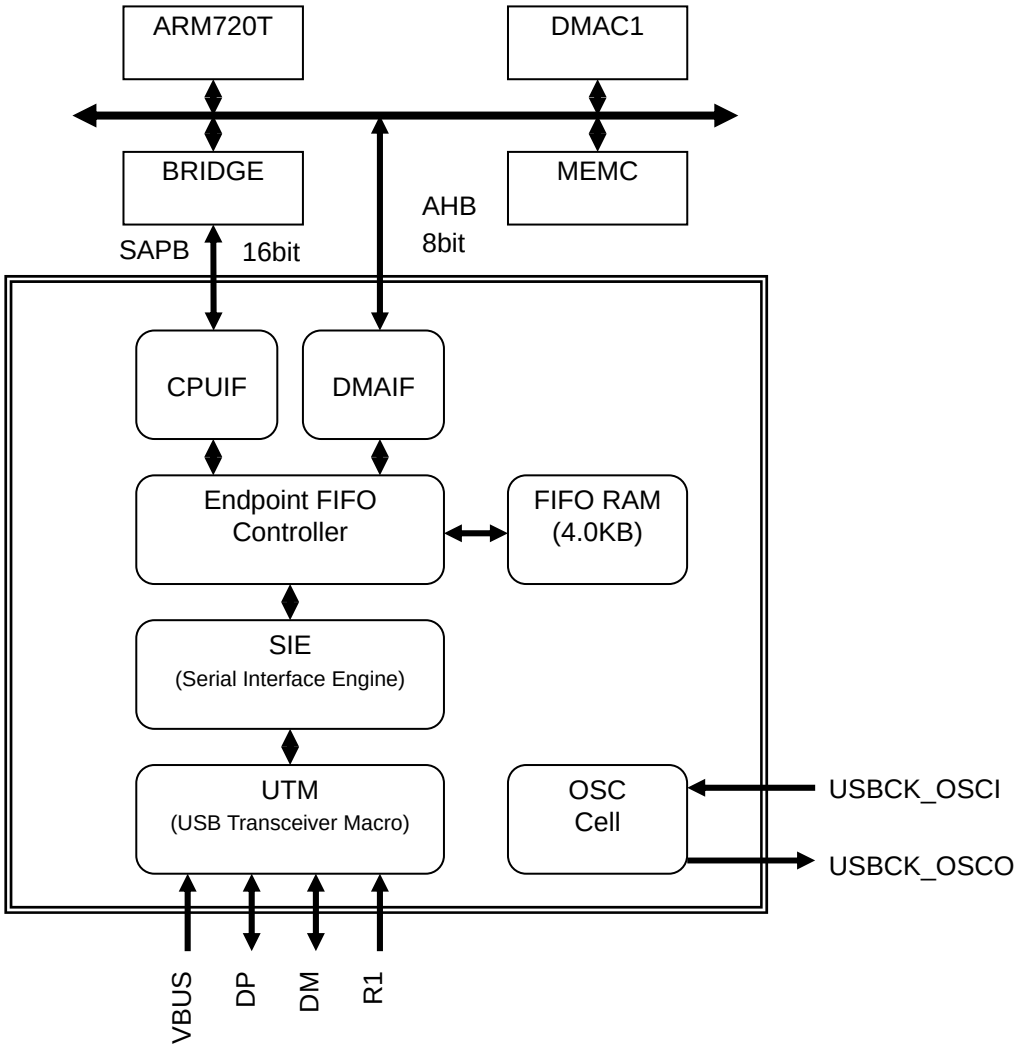


Fig. A2.3.1　S2S65A00 USB Controller Block Diagram

A2.4　Pin List

USB インタフェース関連の外部端子は以下のようになっています。

Fig. A2.4.2　USB 外部端子

端子名	入出力	端子機能
VBUS	I	USB バス検出信号
DP	I/O	USB データライン Data+
DM	I/O	USB データライン Data-
R1	I	内部動作設定 6.2KΩ±1%の抵抗を VSS 間に接続
USBCK_OSCI	I	USB 内部発振回路入力（12/24MHz）
USBCK_OSCO	O	USB 内部発振回路出力

A2.5　Register

A2.5.1　Register 一覧

本ブロック内にあるコントロール・レジスタのレジスタ・マップを Tbl. A2.5.1～Tbl. A2.5.4 に示します。表中のアドレスは APB バスにおけるアドレスとなります。これらのベースアドレスは、**0xFFFD_F000** です。

Tbl. A2.5.1　USB レジスター覧（1/4）

アドレス オフセット	レジスタ名称	説明	R/W	初期値
割込み制御レジスタ				
0x000	MainIntStat	割込みステータスレジスタ	R/(W)	0x00
0x003	CPU_IntStat	CPUIF 関連割込みステータスレジスタ	R/(W)	0x00
0x007	DMA_IntStat	DMA 関連割込みステータスレジスタ	R/(W)	0x00
0x010	MainIntEnb	割込みイネーブルレジスタ	R/W	0x00
0x013	CPU_IntEnb	CPUIF 関連割込みイネーブルレジスタ	R/W	0x00
0x017	DMA_IntEnb	DMA 関連割込みイネーブルレジスタ	R/W	0x00
Power Management レジスタ				
0x020	PM_Control0	Power Management 制御レジスタ	R/W	0x00
0x021	PM_Control1	Power Management ステータスレジスタ	R	0x00
0x022	WakeupTim_L	Wakeup Time 設定レジスタ	R/W	0x00
0x023	WakeupTim_H		R/W	0x00
MISC レジスタ				
0x031	MacroReset	ソフトウェアリセットレジスタ	R/W	0x00
0x033	ModeProtect	設定プロテクトレジスタ	R/W	0x56
0x035	MacroConfig0	クロック設定レジスタ	R/W	0x41
0x037	MacroConfig1	マクロ動作モード設定レジスタ	R/W	0x06
0x039	(fix_CPU_Swap)		R	
0x03A	MacroType0	マクロタイプレジスタ 0	R	0x44
0x03B	MacroType1	マクロタイプレジスタ 1	R	0x08
0x03C	MacroType2	マクロタイプレジスタ 2	R	0x02
0x03D	MacroType3	マクロタイプレジスタ 3	R	0x30
0x03E	FIFO_CapacityL	FIFO RAM 容量レジスタ	R	0x00
0x03F	FIFO_CapacityH		R	0x12
DMA 制御レジスタ				
0x041	DMA0_Config	DMA0 設定レジスタ	R/W	0x00
0x042	DMA0_Control	DMA0 制御レジスタ	R/W	0x00
0x044	DMA0_Remain_L	DMA0 FIFO 残量レジスタ	R	0x00
0x045	DMA0_Remain_H		R	0x00
0x048	DMA0_Count_HL	DMA0 転送カウンタレジスタ	R/W	0x00
0x049	DMA0_Count_HH		R/W	0x00
0x04A	DMA0_Count_LL		R/W	0x00
0x04B	DMA0_Count_LH		R/W	0x00
0x051	DMA1_Config	DMA1 設定レジスタ	R/W	0x00
0x052	DMA1_Control	DMA1 制御レジスタ	R/W	0x00
0x054	DMA1_Remain_L	DMA1 FIFO 残量レジスタ	R	0x00
0x055	DMA1_Remain_H		R	0x00
0x058	DMA1_Count_HL	DMA1 転送カウンタレジスタ	R/W	0x00
0x059	DMA1_Count_HH		R/W	0x00
0x05A	DMA1_Count_LL		R/W	0x00
0x05B	DMA1_Count_LH		R/W	0x00

Tbl. A2.5.2　USB レジスター覧（2/4）

アドレス オフセット	レジスタ名称	説明	R/W	初期値
USB 制御レジスタ				
0x080	DeviceIntStat	USB 割込みステータスレジスタ	R/(W)	0x00
0x081	EPrIntStat	Endpoint 割込みステータスレジスタ	R	0x00
0x082	SIE_IntStat	SIE 割込みステータスレジスタ	R/(W)	0x00
0x084	FIFO_IntStat	FIFO 割込みステータスレジスタ	R/(W)	0x00
0x085	BulkIntStat	Bulk 転送サポート割込みステータスレジスタ	R/(W)	0x00
0x087	EP0IntStat	EP0 割込みステータスレジスタ	R/(W)	0x00
0x088	EPaIntStat	EPa 割込みステータスレジスタ	R/(W)	0x00
0x089	EPbIntStat	EPb 割込みステータスレジスタ	R/(W)	0x00
0x08A	EPcIntStat	EPc 割込みステータスレジスタ	R/(W)	0x00
0x08B	EPdIntStat	EPd 割込みステータスレジスタ	R/(W)	0x00
0x08C	EPeIntStat	EPe 割込みステータスレジスタ	R/(W)	0x00
0x08D	EPfIntStat	EPf 割込みステータスレジスタ	R/(W)	0x00
0x08E	EPgIntStat	EPg 割込みステータスレジスタ	R/(W)	0x00
0x08F	EPhIntStat	EPh 割込みステータスレジスタ	R/(W)	0x00
0x090	DeviceIntEnb	USB 割込み制御レジスタ	R/W	0x00
0x091	EPrIntEnb	Endpoint 割込み制御レジスタ	R/W	0x00
0x092	SIE_IntEnb	SIE 割込み制御レジスタ	R/W	0x00
0x094	FIFO_IntEnb	FIFO 割込み制御レジスタ	R/W	0x00
0x095	BulkIntEnb	Bulk 転送サポート割込み制御レジスタ	R/W	0x00
0x097	EP0IntEnb	EP0 割込み制御レジスタ	R/W	0x00
0x098	EPaIntEnb	EPa 割込み制御レジスタ	R/W	0x00
0x099	EPbIntEnb	EPb 割込み制御レジスタ	R/W	0x00
0x09A	EPcIntEnb	EPc 割込み制御レジスタ	R/W	0x00
0x09B	EPdIntEnb	EPd 割込み制御レジスタ	R/W	0x00
0x09C	EPeIntEnb	EPe 割込み制御レジスタ	R/W	0x00
0x09D	EPfIntEnb	EPf 割込み制御レジスタ	R/W	0x00
0x09E	EPgIntEnb	EPg 割込み制御レジスタ	R/W	0x00
0x09F	EPhIntEnb	EPh 割込み制御レジスタ	R/W	0x00
0x0A0	ResetDTM	トランシーバマクロリセットレジスタ	R/W	0x01
0x0A2	NegoControl	ネゴシエーション制御レジスタ	R/W	0x00
0x0A4	USB_Status	USB バスステータスレジスタ	R/W	0xXX
0x0A5	XcvrControl	トランシーバマクロ制御レジスタ	R/W	0x41
0x0A6	USB_Test	USB テストレジスタ	R/W	0x00
0x0A8	EPnControl	EP 共通制御レジスタ	W	0xXX
0x0A9	EPrFIFO_Clr	FIFO クリアレジスタ	W	0xXX
0x0AA	ClrAllEPnJoin	EP ジョインクリアレジスタ	W	0xXX
0x0AC	BulkOnlyControl	Bulk 転送サポート制御レジスタ	R/W	0x00
0x0AD	BulkOnlyConfig	Bulk 転送サポート設定レジスタ	R/W	0x00
0x0B0	EP0SETUP_0	EP0 セットアップデータレジスタ	R	0x00
0x0B1	EP0SETUP_1		R	0x00
0x0B2	EP0SETUP_2		R	0x00
0x0B3	EP0SETUP_3		R	0x00
0x0B4	EP0SETUP_4		R	0x00
0x0B5	EP0SETUP_5		R	0x00
0x0B6	EP0SETUP_6		R	0x00
0x0B7	EP0SETUP_7		R	0x00
0x0B8	USB_Address	USB アドレスレジスタ	R/(W)	0x00
0x0BA	SETUP_Control	セットアップステージ制御レジスタ	R/W	0x00
0x0BE	FrameNumber_L	フレーム番号レジスタ	R	0x00
0x0BF	FrameNumber_H		R	0x80

Tbl. A2.5.3　USB レジスター覧（3/4）

アドレス オフセット	レジスタ名称	説明	R/W	初期値
USB 制御レジスタ				
0x0C0	EP0MaxSize	EP0 マックスパケットサイズ設定レジスタ	R/W	0x40
0x0C1	EP0Control	EP0 制御レジスタ	R/W	0x00
0x0C2	EP0ControlIN	EP0 IN 転送制御レジスタ	R/W	0x00
0x0C3	EP0ControlOUT	EP0 OUT 転送制御レジスタ	R/W	0x00
0x0C5	EP0JOin	EP0　ジョイン設定レジスタ	R/W	0x00
0x0D0	EPaMaxSize_L	EPa マックスパケットサイズ設定レジスタ	R/W	0x00
0x0D1	EPaMaxSize_H		R/W	0x00
0x0D2	EPaConfig	EPa 設定レジスタ	R/W	0x00
0x0D4	EPaControl	EPa 制御レジスタ	R/W	0x00
0x0D5	EPaJoin	EPa ジョイン設定レジスタ	R/W	0x00
0x0D8	EPbMaxSize_L	EPb マックスパケットサイズ設定レジスタ	R/W	0x00
0x0D9	EPbMaxSize_H		R/W	0x00
0x0DA	EPbConfig	EPb 設定レジスタ	R/W	0x00
0x0DC	EPbControl	EPb 制御レジスタ	R/W	0x00
0x0DD	EPbJoin	EPb ジョイン設定レジスタ	R/W	0x00
0x0E0	EPcMaxSize_L	EPc マックスパケットサイズ設定レジスタ	R/W	0x00
0x0E1	EPcMaxSize_H		R/W	0x00
0x0E2	EPcConfig	EPc 設定レジスタ	R/W	0x00
0x0E4	EPcControl	EPc 制御レジスタ	R/W	0x00
0x0E5	EPcJoin	EPc ジョイン設定レジスタ	R/W	0x00
0x0E8	EPdMaxSize_L	EPd マックスパケットサイズ設定レジスタ	R/W	0x00
0x0E9	EPdMaxSize_H		R/W	0x00
0x0EA	EPdConfig	EPd 設定レジスタ	R/W	0x00
0x0EC	EPdControl	EPd 制御レジスタ	R/W	0x00
0x0ED	EPdJoin	EPd ジョイン設定レジスタ	R/W	0x00
0x0F0	EPeMaxSize_L	EPe マックスパケットサイズ設定レジスタ	R/W	0x00
0x0F1	EPeMaxSize_H		R/W	0x00
0x0F2	EPeConfig	EPe 設定レジスタ	R/W	0x00
0x0F4	EPeControl	EPe 制御レジスタ	R/W	0x00
0x0F5	EPeJoin	EPe ジョイン設定レジスタ	R/W	0x00
0x0F8	EPfMaxSize_L	EPf マックスパケットサイズ設定レジスタ	R/W	0x00
0x0F9	EPfMaxSize_H		R/W	0x00
0x0FA	EPfConfig	EPf 設定レジスタ	R/W	0x00
0x0FC	EPfControl	EPf 制御レジスタ	R/W	0x00
0x0FD	EPfJoin	EPf ジョイン設定レジスタ	R/W	0x00
0x100	EPgMaxSize_L	EPg マックスパケットサイズ設定レジスタ	R/W	0x00
0x101	EPgMaxSize_H		R/W	0x00
0x102	EPgConfig	EPg 設定レジスタ	R/W	0x00
0x104	EPgControl	EPg 制御レジスタ	R/W	0x00
0x105	EPgJoin	EPg ジョイン設定レジスタ	R/W	0x00
0x108	EPhMaxSize_L	EPh マックスパケットサイズ設定レジスタ	R/W	0x00
0x109	EPhMaxSize_H		R/W	0x00
0x10A	EPhConfig	EPh 設定レジスタ	R/W	0x00
0x10C	EPhControl	EPh 制御レジスタ	R/W	0x00
0x10D	EPhJoin	EPh ジョイン設定レジスタ	R/W	0x00

Tbl. A2.5.4　USB レジスター覧（4/4）

アドレス オフセット	レジスタ名称	説明	R/W	初期値
USB FIFO 設定レジスタ				
0x110	EPaStartAdrs_L	EPa FIFO スタートアドレス設定レジスタ	R/W	0x00
0x111	EPaStartAdrs_H		R/W	0x00
0x112	EPbStartAdrs_L	EPb FIFO スタートアドレス設定レジスタ	R/W	0x00
0x113	EPbStartAdrs_H		R/W	0x00
0x114	EPcStartAdrs_L	EPc FIFO スタートアドレス設定レジスタ	R/W	0x00
0x115	EPcStartAdrs_H		R/W	0x00
0x116	EPdStartAdrs_L	EPd FIFO スタートアドレス設定レジスタ	R/W	0x00
0x117	EPdStartAdrs_H		R/W	0x00
0x118	EPeStartAdrs_L	EPe FIFO スタートアドレス設定レジスタ	R/W	0x00
0x119	EPeStartAdrs_H		R/W	0x00
0x11A	EPfStartAdrs_L	EPf FIFO スタートアドレス設定レジスタ	R/W	0x00
0x11B	EPfStartAdrs_H		R/W	0x00
0x11C	EPgStartAdrs_L	EPg FIFO スタートアドレス設定レジスタ	R/W	0x00
0x11D	EPgStartAdrs_H		R/W	0x00
0x11E	EPhStartAdrs_L	EPh FIFO スタートアドレス設定レジスタ	R/W	0x00
0x11F	EPhStartAdrs_H		R/W	0x00
0x120	EP_EndAdrs_L	EP FIFO エンドアドレス設定レジスタ	R/W	0x00
0x121	EP_EndAdrs_H		R/W	0x12
0x124	DescAdrs_L	デスクリプタ先頭アドレス設定レジスタ	R/W	0x00
0x125	DescAdrs_H		R/W	0x00
0x126	DescSize_L	デスクリプタサイズ設定レジスタ	R/W	0x00
0x127	DescSize_H		R/W	0x00
0x128	DMA0_FIFO_Control	DMA0-FIFO 制御レジスタ	R/W	0x00
0x12A	DMA1_FIFO_Control	DMA1-FIFO 制御レジスタ	R/W	0x00
0x130	FIFO_Rd_0	FIFO リードレジスタ	R	0xXX
0x131	FIFO_Rd_1		R	0xXX
0x132	FIFO_Wr_0	FIFO ライトレジスタ	W	0xXX
0x133	FIFO_Wr_1		W	0xXX
0x134	FIFO_RdRemain_L	FIFO リード残量レジスタ	R	0x00
0x135	FIFO_RdRemain_H		R	0x00
0x136	FIFO_WrRemain_L	FIFO ライト残量レジスタ	R	0x00
0x137	FIFO_WrRemain_H		R	0x00
0x138	FIFO_ByteRd	FIFO バイトリードレジスタ	R	0xXX
0x140	RAM_RdAdrs_L	RAM リードアドレス設定レジスタ	R/W	0x00
0x141	RAM_RdAdrs_H		R/W	0x00
0x142	RAM_RdControl	RAM リード制御レジスタ	R/W	0x00
0x143	RAM_RdCount	RAM リードカウント設定レジスタ	R/W	0x00
0x144	RAM_WrAdrs_L	RAM ライトアドレス設定レジスタ	R/W	0x00
0x145	RAM_WrAdrs_H		R/W	0x00
0x146	RAM_WrDoor_0	RAM ライトレジスタ	W	0xXX
0x147	RAM_WrDoor_1		W	0xXX
0x150 ～ 0x16F	RAM_Rd_00 ～ RAM_Rd_1F	RAM リードデータレジスタ	R	0xXX
テストレジスタ（Reserved）				
0x170 ～ 0x17F	Reserved	テスト用レジスタ	－	

A2.5.2　Register 詳細説明

A2.5.2.1　Register Access 上の注意事項

(1) Register が割り付けられていない Address に対しては書き込みを行わないでください。
(2) Register が割り付けられていない bit に対しては“0x0”の書き込みのみとしてください。
(3) SLEEP / SNOOZE 時にでも読み書きできる Register は***太字斜体***で示しています。
(4) それ以外の Register は、ACTIVE60 / ACT_DEVICE 時に読み書きができます。

A2.5.2.2　割込み制御レジスタ

Main Interrupt Status
USB[0x000]　　初期値 = 0x00

Device IntStat(R)	−	CPU_ IntStat(R)	−	−	DMA_ IntStat(R)	−	***Finished PM***
7	6	5	4	3	2	1	0

USB マクロの割り込み要因を示します。
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。MainInterruptEnb 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ときに、割り込み要因が “1” にセットされると INTC に対して割り込みが発生します。

Bit 7 :　**DeviceIntStat**
割り込み要因を間接指示します。
Device_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 SIE_DeviceIntEnb レジスタのビットがイネーブルにされているときに“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。

Bit 6 :　**Reserved**

Bit 5 :　**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ときに “1” にセットされます。

Bit [4:3] :　**Reserved**

Bit 2 :　**DMAIntStat**
割り込み要因を間接指示します。
DMAIntStat 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 DMAIntEnb レジスタのビットがイネーブルにされているときに“1”にセットされます。

Bit 1 :　**Reserved**

Bit 0 :　**FinishedPM**
割り込み要因を直接指示します。
PM_Control レジスタで、GoSLEEP,GoSNOOZE,GoActive60,GoActDevice 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。

CPU Interrupt Status USB[0x003]　　初期値 = 0x00							
RAM_ RdCmp	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0

CPU インタフェース関連の割り込みを表示します。
全てのビットは“1”を書き込むことで割り込み要因をクリアできます。

Bit 7　　**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。

Bit [6:0] :　　**Reserved**

DMA Interrupt Status USB[0x007]　　初期値 = 0x00							
–	–	–	–	DMA1_ CountUp	DMA1_ Cmp	DMA0_ CountUp	DMA0_ Cmp
7	6	5	4	3	2	1	0

DMA 関連の割り込みを表示します。
全てのビットは“1”を書き込むことで割り込み要因をクリアできます。

Bit [7:4] :　　**Reserved**

Bit 3 :　　**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ときに、DMA1_Count_HH,HL,LH,LL の値がオーバーフローしたときに、“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。

Bit 2 :　　**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ときに“1”にセットされます。

Bit 1 :　　**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ときに、DMA0_Count_HH,HL,LH,LL の値がオーバーフローしたときに、“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。

Bit 0 :　　**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ときに“1”にセットされます。

Main Interrupt Enable
USB[0x010]　　初期値 = 0x00

EnDevice IntStat	−	EnCPU IntStat	−	−	EnDMA_ IntStat	−	***EnFinished PM***
7	6	5	4	3	2	1	0

MainIntStat レジスタの割り込み要因による INTC モジュールに対する割り込み信号のアサートを許可／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, EnFinishedPM ビットは SLEEP / SNOOZE 中も有効です。

CPU Interrupt Enable
USB[0x013]　　初期値 = 0x00

EnRAM_ RdCmp	−	−	−	−	−	−	−
7	6	5	4	3	2	1	0

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。

DMA Interrupt Enable
USB[0x017]　　初期値 = 0x00

−	−	−	−	EnDMA1_ CountUp	EnDMA1_ Cmp	EnDMA0_ CountUp	EnDMA0_ Cmp
7	6	5	4	3	2	1	0

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

A2.5.2.3　Power Management 制御レジスタ

Power Management Control 0 USB[0x020]　　初期値 = 0x00							
Go SLEEP 7	***Go SNOOZE*** 6	***Go Active60*** 5	***Go ActDevice*** 4	－ 3	－ 2	－ 1	－ 0

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。

Bit 7：　**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。
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。

Bit 6：　**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。
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。

Bit 5：　**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 250µs）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 250µs）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。
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。

Bit 4 :　**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 250µs）経過後、DevicePLL480 の発振を開始し、PLL480 発振安定時間（約 250µs）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 250µs）経過後、DevicePLL480 の発振を開始し、PLL480 発振安定時間（約 250µs）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 250µs）経過後、ACT_DEVICE 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。

※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、F/W 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）

Bit [3:0] :　**Reserved**

Power Management Control 1
USB[0x021]　初期値 = 0x00

–	–	–	–	***PM_State[3:0]***(R)			
7	6	5	4	3	2	1	0

本 LSI のパワーマネージメント関連の動作状態をモニタできます。
このレジスタは SLEEP / SNOOZE 中も有効です。

Bit [7:4] :　**Reserved**

Bit [3:0] :　**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）
0001 : SNOOZE ステート　（OSC オン、PLL60 オフ、DevicePLL480 オフ）
0011 : ACTIVE60 ステート　（OSC オン、PLL60 オン、DevicePLL480 オフ）
0111 : ACT_DEVICE ステート　（OSC オン、PLL60 オン、DevicePLL480 オン）
その他 : 未使用
なお本ステートは、PM_Contorl0.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_Control0.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。

Wakeup Time[7:0]							
USB[0x022]　　初期値 = 0x00							
WakeupTim[7:0]							
7	6	5	4	3	2	1	0

Wakeup Time[15:8]							
USB[0x023]　　初期値 = 0x00							
WakeupTim[15:8]							
7	6	5	4	3	2	1	0

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_Control0.GoActDevice、PM_Control0.GoActive60、PM_Control0. 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 Wakeup_Tim 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（250µs 未満）＋ PLL480 安定時間（250µs 未満）の合計が 5.1ms 以下とならなければなりません。

A2.5.2.4　MISC レジスタ

Macro Reset							
USB[0x031]　　初期値 = 0xXX							
–	–	–	–	–	–	–	***AllReset*** (W)
7	6	5	4	3	2	1	0

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

Bit [7:1] :　**Reserved**

Bit 0 :　**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。

Mode Protect							
USB[0x033]　　初期値 = 0x56							
ModeProtect[7:0]							
7	6	5	4	3	2	1	0

Bit [7:0] :　**ModeProtect[7:0]**
MacroConfig0 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 MacroConfig0 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、MacroConfig0 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、MacroConfig0 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。

Macro Configuration 0 USB[0x035]　　初期値 = 0x41							
–	–	–	–	–	–	–	***ClkSelect***
7	6	5	4	3	2	1	0

Bit [7:1] :　**Reserved**

Bit 0 :　**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

Macro Configuration 1 USB[0x037]　　初期値 = 0x06							
IntLevel	***IntMode***	***DREQ_ Level***	***DACK_ Level***	***CS_Mode***	***CPU_Swap***	***BusMode***	***Bus8x16***
7	6	5	4	3	2	1	0

本 LSI の動作モードを設定します。本レジスタの初期値は S2S65A00 用に最適化されていますので、特に必要のない限り設定を変更しないでください。

Bit 7 :　**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

Bit 6 :　**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

Bit 5 :　**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

Bit 4 :　**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

Bit 3 :　**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。

Bit 2 :　**CPU_Swap**
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、039h 番地をリードすること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 039h 番地をリードした後になります。

Bit [1:0] :　**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

Macro Type 0
USB[0x03A]　初期値 = 0x44

Device_Host[7:0](R)							
7	6	5	4	3	2	1	0

本マクロのマクロタイプを表します。

Bit [7:0] :　**Device/Host[7:0]**
Device か Host かを表します。
Device : 0x44
Host : 0x48

Macro Type 1
USB[0x03B]　初期値 = 0x08

–	–	–	***Exist_ IDE***(R)	***EP_Num[3:0]*** (R)			
7	6	5	4	3	2	1	0

本マクロのマクロタイプを表します。

Bit [7:5] :　**Reserved**

Bit 4 :　**Exist_IDE**
IDE が実装されているかを表します。
Not mount :　0x0
Mount :　0x1

Bit [3:0] :　**EP_Num[3:0]**
End Point 数を表します。
16 進数で表します。

Macro Type 2
USB[0x03C]　初期値 = 0x02

–	–	–	–	***DMAch_Count[3:0]*** (R)			
7	6	5	4	3	2	1	0

本マクロのマクロタイプを表します。

Bit [7:4] :　**Reserved**

Bit [3:0] :　**DMAch_Count[3:0]**
DMA チャンネル数の総数を表します。
16 進数でチャンネル数を表します

Macro Type 3
USB[0x03D]　　初期値 = 0x30

PortDMA_Mount[3:0] (R)				***CPUDMA_Mount[3:0]*** (R)			
7	6	5	4	3	2	1	0

本マクロのマクロタイプを表します。

Bit [7:4]：　**PortDMA_Mount[3:0]**
実装されている Sync DMA のチャンネルを表します。
2 進数で実装されているチャンネルを表します。

Bit [3:0]：　**CPUDMA_Mount[3:0]**
実装されている CPU DMA のチャンネルを表します。
2 進数で実装されているチャンネルを表します。

FIFO Capacity[7:0]
USB[0x03E]　　初期値 = 0x00

FIFO_Capacity[7:0]							
7	6	5	4	3	2	1	0

FIFO Capacity[15:8]
USB[0x03F]　　初期値 = 0x12

FIFO_Capacity[15:8]							
7	6	5	4	3	2	1	0

実装されている FIFO RAM の容量を表します。

A2.5.2.5　DMA 制御レジスタ

DMA0 Configuration
USB[0x041]　　初期値 = 0x00

FreeRun	−	−	−	ActiveDMA	−	−	−
7	6	5	4	3	2	1	0

DMA1 Configuration
USB[0x051]　　初期値 = 0x00

FreeRun	−	−	−	ActiveDMA	−	−	−
7	6	5	4	3	2	1	0

DMA0/1 の動作モードを設定します。

Bit 7：　**FreeRun**
DMA0/1 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

Bit [6:4]：　**Reserved**

Bit 3 :　**ActiveDMA**
PortDMA0/1 を有効にします。転送中にこの bit をクリアすると即座に転送を中止します。再転送する際は USB FIFO、PortDMA の初期化を行ってください。通常の停止では DMA_Stop を使用してください。
なおこの bit は USBFIFO が Clear されたり、Join が切り替えられた場合、自動的に Clear されます。
1： PortDMA0/1 有効
0： PortDMA0/1 無効

Bit [2:0] :　**Reserved**

DMA0 Control
USB[0x042]　初期値 = 0x00

Running (R)	–	–	Counter Clr（W）	Dir	–	Stop (W)	Go (W)
7	6	5	4	3	2	1	0

DMA1 Control
USB[0x052]　初期値 = 0x00

Running (R)	–	–	Counter Clr（W）	Dir	–	Stop (W)	Go (W)
7	6	5	4	3	2	1	0

DMA0/1 の制御、及び状態を表示します。

Bit 7 :　**Running**
DMA0/1 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~h}Join.JoinDMA0/1 を書き換えることはできません。

Bit [6:5] :　**Reserved**

Bit 4 :　**CounterClr**
このビットに“1”をセットすると、Count_HH,HL,LH,LL レジスタが 0x00 にクリアされます。Running ビットが“1”であるときは、このビットへの書き込みは無視されます。

Bit 3 :　**Dir**
DMA0/1 の転送方向を設定します。
0： Port → FIFO RAM（DMA ライト）
1： Port ← FIFO RAM（DMA リード）

Bit 2 :　**Reserved**

Bit 1 :　**Stop**
このビットに“1”をセットすると、DMA0/1 の転送を終了します。DMA0/1 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 DMA0/1_Cmp ビットに“1”をセットします。DMA の転送を再開する場合、Running ビットまたは DMA0/1_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。

Bit 0 :　**Go**
このビットを“1”にセットすると、DMA0/1 の転送を開始します。

DMA0 Remain[7:0]
USB[0x044]　　初期値 = 0x00

Remain[7:0](R)							
7	6	5	4	3	2	1	0

DMA1 Remain[7:0]
USB[0x54]　　初期値 = 0x00

Remain[7:0](R)							
7	6	5	4	3	2	1	0

DMA0 Remain[12:8]
USB[0x045]　　初期値 = 0x00

–	–	–	Remain[12:8](R)				
7	6	5	4	3	2	1	0

DMA1 Remain[12:8]
USB[0x055]　　初期値 = 0x00

–	–	–	Remain[12:8](R)				
7	6	5	4	3	2	1	0

Bit [7:0] :　**Remain[7:0]**

Bit [7:5] :　**Reserved**

Bit [4:0] :　**Remain[12:8]**

読み出しの場合、EPx{x=0,a~h}Join.JoinDMA0/1 ビットによって DMA に接続しているエンドポイントの 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~h}Join.JoinDMA0/1 ビットによって DMA に接続しているエンドポイントの FIFO の空き容量を示します。DMA 書き込みを行った直後にはこのレジスタで正しい FIFO の空き容量を参照することが出来ません。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上位バイト、下位バイトの順に読み出してください。

DMA0 Count[23:16]
USB[0x048]　　初期値 = 0x00

Count[23:16]							
7	6	5	4	3	2	1	0

DMA1 Count[23:16]
USB[0x058]　　初期値 = 0x00

Count[23:16]							
7	6	5	4	3	2	1	0

DMA0 Count[31:24]
USB[0x049]　　初期値 = 0x00

Count[31:24]							
7	6	5	4	3	2	1	0

DMA1 Count[31:24]
USB[0x059]　　初期値 = 0x00

Count[31:24]							
7	6	5	4	3	2	1	0

DMA0 Count[7:0]
USB[0x04A]　　初期値 = 0x00

Count[7:0]							
7	6	5	4	3	2	1	0

DMA1 Count[7:0]
USB[0x05A]　　初期値 = 0x00

Count[7:0]							
7	6	5	4	3	2	1	0

DMA0 Count[15:8]
USB[0x04B]　　初期値 = 0x00

Count[15:8]							
7	6	5	4	3	2	1	0

DMA1 Count[15:8]
USB[0x05B]　　初期値 = 0x00

Count[15:8]							
7	6	5	4	3	2	1	0

Bit [7:0] :　　**Count[23:16]**

Bit [7:0] :　　**Count[31:24]**

Bit [7:0] :　　**Count[7:0]**

Bit [7:0] :　　**Count[15:8]**

カウントモード時に DMA0/1 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、DMA0/1_Control.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。DMA0/1_Count レジスタの値がオーバーフローすると、DMAIntStat レジスタの DMA0/1_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上位バイトから順に読み出してください。

A2.5.2.6　USB 制御レジスタ

Device Interrupt Status USB[0x080]　　初期値 = 0x00							
VBUS_Changed 7	Descriptor Cmp 6	***SIE_IntStat***(R) 5	Bulk_ IntStat(R) 4	RcvEP0 SETUP 3	FIFO_ IntStat(R) 2	EP0_ IntStat(R) 1	EPr_ IntStat(R) 0

USB 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。

Bit 7 :　**VBUS_Changed**
割り込み要因を直接指示します。
VBUS 端子の状態が変化したときに “1” にセットされます。
USB_Status レジスタの VBUS ビットによって VBUS の状態を確認して下さい。VBUS が“0”であれば、ケーブルが抜かれたことを示します。このビットは SLEEP / SNOOZE 中も有効です。

Bit 6 :　**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えると、“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 EP0IntStat レジスタの OUT_TranNAK ビットと共に、“1”にセットされます。

Bit 5 :　**SIE_IntStat**
割り込み要因を間接指示します。
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 SIE_IntEnb レジスタのビットがイネーブルにされているときに“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。

Bit 4 :　**BulkIntStat**
割り込み要因を間接指示します。
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 BulkIntEnb レジスタのビットがイネーブルにされているときに “1” にセットされます。

Bit 3 :　**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 EP0SETUP_0～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 EP0ControlIN, EP0ControlOUT レジスタの ForceSTALL ビットが“0”に EP0ControlIN, EP0ControlOUT レジスタの ForceNAK ビット、ToggleStat ビット、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。

Bit 2 :　**FIFO_IntStat**
割り込み要因を間接指示します。
FIFO_IntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する FIFO_IntEnb レジスタのビットがイネーブルにされているときに “1” にセットされます。

Bit 1：　**EP0IntStat**
割り込み要因を間接指示します。
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 EP0IntEnb レジスタのビットがイネーブルにされているときに “1” にセットされます。

Bit 0：　**EPrIntStat**
割り込み要因を間接指示します。
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 EPaIntEnb レジスタのビットがイネーブルにされているとき、“1”にセットされます。

EPr Interrupt Status
USB[0x081]　　初期値 = 0x00

EP h IntStat(R) 7	EPg IntStat(R) 6	EPf IntStat(R) 5	EPe IntStat(R) 4	EPd IntStat(R) 3	EPc IntStat(R) 2	EPb IntStat(R) 1	EPa IntStat(R) 0

エンドポイント EPr の割り込みを表示します。

Bit 7：　**EPhIntStat**
割り込み要因を間接指示します。
EPhIntStat 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 EPhIntEnb レジスタのビットがイネーブルにされているとき、“1” にセットされます。

Bit 6：　**EPgIntStat**
割り込み要因を間接指示します。
EPgIntStat 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 EPgIntEnb レジスタのビットがイネーブルにされているとき、“1” にセットされます。

Bit 5：　**EPfIntStat**
割り込み要因を間接指示します。
EPfIntStat 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 EPfIntEnb レジスタのビットがイネーブルにされているとき、“1” にセットされます。

Bit 4：　**EPeIntStat**
割り込み要因を間接指示します。
EPeIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する EPeIntEnb レジスタのビットがイネーブルにされているとき、“1” にセットされます。

Bit 3：　**EPdIntStat**
割り込み要因を間接指示します。
EPdIntStat レジスタに割り込み要因があり、かつその割り込み要因に対応する EPdIntEnb レジスタのビットがイネーブルにされているとき、“1” にセットされます。

Bit 2：　**EPcIntStat**
割り込み要因を間接指示します。
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 EPcIntEnb レジスタのビットがイネーブルにされているとき、“1” にセットされます。

Bit 1：　**EPbIntStat**
割り込み要因を間接指示します。
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 EPbIntEnb レジスタのビットがイネーブルにされているとき、“1” にセットされます。

Bit 0：　**EPaIntStat**
割り込み要因を間接指示します。
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 EPaIntEnb レジスタのビットがイネーブルにされているとき、“1”にセットされます。

SIE Interrupt Status USB[0x082]　　初期値 = 0x00							
–	***NonJ***	RcvSOF	Detect Reset	Detect Suspend	Chirp Cmp	Restore Cmp	SetAddress Cmp
7	6	5	4	3	2	1	0

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”を書き込むことで割り込み要因をクリアできます。

Bit 7 :　**Reserved**

Bit 6 :　**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”）のとき、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ときに有効です。

Bit 5 :　**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。

Bit 4 :　**DetectReset**
割り込み要因を直接指示します。
USB のリセットステートを検出すると“1”にセットされます。このビットがセットされている間は USB のサスペンドステートの検出ができません（DetectSUSPEND がセットされません）。
このリセット検出は、NegoControl レジスタの ActiveUSB ビットが “1” にセットされているときに有効です。
“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、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、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、NegoControl レジスタの EnAutoNego ビットの項を参照して下さい。

Bit 3 :　**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_Control0 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。

Bit 2 :　**ChirpCmp**
割り込み要因を直接指示します。
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 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。

Bit 1 :　**RestoreCmp**
割り込み要因を直接指示します。
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。

Bit 0 :　**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ときに、このステータスが“1”にセットされます。また、同時に 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 ステートのとき、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、F/W 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（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（SIE_IntEnb.Bit5～0）
＜ACT_DEVICE ステートに入るとき＞
1)　割り込みステータスをクリアする（SIE_IntStat.Bit5～0）
2)　割り込みステータスをイネーブルにする（SIE_IntEnb.Bit5～0）

FIFO Interrupt Status
USB[0x084]　　初期値 = 0x00

–	–	FIFO_ DMA1_Cmp	FIFO_ DMA0_Cmp	–	FIFO_ NotEmpty	FIFO_ Full	FIFO_ Empty
7	6	5	4	3	2	1	0

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”を書き込むことで割り込み要因をクリアできます。

Bit [7:6] :　**Reserved**

Bit 5 :　**FIFO_DMA1_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。

Bit 4 :　**FIFO_DMA0_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。

Bit 3 :　**Reserved**

Bit 2 :　**FIFO_NotEmpty**
割り込み要因を直接指示します。
EPx{x=0,a~h}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 でないときに、“1”にセットされます。

Bit 1：　**FIFO_Full**
割り込み要因を直接指示します。
EPx{x=0,a~h}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。

Bit 0：　**FIFO_Empty**
割り込み要因を直接指示します。
EPx{x=0,a~h}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。

Bulk Interrupt Status
USB[0x085]　　初期値 = 0x00

CBW_ Cmp 7	CBW_ LengthErr 6	CBW_ Err 5	_ 4	CSW_ Cmp 3	CSW_ Err 2	_ 1	_ 0

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”を書き込むことで割り込み要因をクリアできます。

Bit 7：　**CBW_Comp**
割り込み要因を直接指示します。
CBW の 31 バイトを正常に受信できたときに“1”にセットされます。

Bit 6：　**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ときに“1”にセットされます。

Bit 5：　**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ときに“1”にセットされます。

Bit 4：　**Reserved**

Bit 3：　**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ときに“1”にセットされます。

Bit 2：　**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とき（ACK が返ってこなかったとき）に“1”にセットされます。

Bit [1:0]：　**Reserved**

EP0 Interrupt Status USB[0x087]　　初期値 = 0x00							
_ 7	OUT_ ShortACK 6	IN_ TranACK 5	OUT_ TranACK 4	IN_ TranNAK 3	OUT_ TranNAK 2	IN_ TranErr 1	OUT_ TranErr 0

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”を書き込むことで割り込み要因をクリアできます。

Bit 7：　**Reserved**

Bit 6：　**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 を返信したとき、 OUT_TranACK と同時に“1” にセットされます。

Bit 5：　**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 を受信したとき、“1”にセットされます。

Bit 4：　**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 を返信したとき、“1”にセットされます。

Bit 3：　**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 を返信したとき、“1”にセットされます。

Bit 2：　**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 を返信したとき、“1”にセットされます。

Bit 1：　**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。

Bit 0：　**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。

EPa Interrupt Status
USB[0x088]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

EPb Interrupt Status
USB[0x089]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

EPc Interrupt Status
USB[0x08A]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

EPd Interrupt Status
USB[0x08B]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

EPe Interrupt Stauts
USB[0x08C]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

EPf Interrupt Status
USB[0x08D]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

EPg Interrupt Status
USB[0x08E]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

EPh Interrupt Status
USB[0x08F]　　初期値 = 0x00

–	OUT_ ShortACK	IN_ TranACK	OUT_ TranACK	IN_ TranNAK	OUT_ TranNAK	IN_ TranErr	OUT_ TranErr
7	6	5	4	3	2	1	0

エンドポイント EPx｛a～h｝の割り込みステータスを表示します。全てのビットは“1”を書き込むことで割り込み要因をクリアできます。

Bit 7 : **Reserved**

Bit 6 : **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 を返信したとき、 OUT_TranACK と同時に“1” にセットされます。

Bit 5 : **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 を受信したとき、“1”にセットされます。

Bit 4 : **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 を返信したとき、“1”にセットされます。

Bit 3 : **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 を返信したとき、“1”にセットされます。

Bit 2 : **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 を返信したとき、“1”にセットされます。

Bit 1 : **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。

Bit 0 : **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。

Device Interrupt Enable
USB[0x090]　　初期値 = 0x00

EnVBUS_ Changed	EnDescriptor Cmp	***EnSIE_ IntStat***	EnBulk IntStat	EnRcvEP0 SETUP	EnFIFO_ IntStat	EnEP0 IntStat	EnEPr IntStat
7	6	5	4	3	2	1	0

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, EnSIE_IntStat ビットは SLEEP / SNOOZE 中も有効です。

EPr Interrupt Enable
USB[0x091]　　初期値 = 0x00

EnEPh IntStat	EnEPg IntStat	EnEPf IntStat	EnEPe IntStat	EnEPd IntStat	EnEPc IntStat	EnEPb IntStat	EnEPa IntStat
7	6	5	4	3	2	1	0

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

SIE Interrupt Enable
USB[0x092]　　初期値 = 0x00

–	***EnNonJ***	EnRcvSOF	EnDetect RESET	EnDetect SUSPEND	EnChirp Cmp	EnRestore Cmp	EnSet AddressCmp
7	6	5	4	3	2	1	0

SIE_IntStat レジスタの割り込み要因による、DeviceIntStat レジスタの 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 ステートのとき、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、SIE_IntStat レジスタの説明をご参照下さい。

FIFO Interrupt Enable
USB[0x094]　　初期値 = 0x00

–	–	EnFIFO_ DMA1_Cmp	EnFIFO_ DMA0_Cmp	–	EnFIFO_ NotEmpty	EnFIFO_ Full	EnFIFO_ Empty
7	6	5	4	3	2	1	0

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

Bulk Interrupt Enable
USB[0x095]　　初期値 = 0x00

EnCBW_ Cmp	EnCBW_ LengthErr	EnCBW_ Err	–	EnCSW_ Cmp	EnCSW_Err	–	–
7	6	5	4	3	2	1	0

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

EP0 Interrupt Enable
USB[0x097]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EP0IntStat レジスタの割り込み要因による、DeviceIntStat レジスタの EP0IntStat ビットのアサートを許可／禁止します。

EPa Interrupt Enable
USB[0x098]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPb Interrupt Enable
USB[0x099]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPc Interrupt Enable
USB[0x09A]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPd Interrupt Enable
USB[0x09B]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPe Interrupt Enable
USB[0x09C]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPf Interrupt Enalbe
USB[0x09D]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPg Interrupt Enable
USB[0x09E]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPh Interrupt Enable
USB[0x09F]　　初期値 = 0x00

–	EnOUT_ ShortACK	EnIN_ TranACK	EnOUT_ TranACK	EnIN_ TranNAK	EnOUT_ TranNAK	EnIN_ TranErr	EnOUT_ TranErr
7	6	5	4	3	2	1	0

EPx{a-h}IntStat レジスタの割り込み要因による、EPrIntStat レジスタの EPx{a-h}IntStat ビットのアサートを許可／禁止します。

Reset DTM USB[0x0A0]　　初期値 = 0x01							
–	–	–	–	–	–	–	***ResetDTM***
7	6	5	4	3	2	1	0

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。

Bit [7:1] :　**Reserved**

Bit 0 :　**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。

Negotiation Control USB[0x0A2]　　初期値 = 0x00							
DisBus Detect	EnAuto Nego	In SUSPEND	Disable HS	Send Wakeup	Restore USB	GoChirp	Active USB
7	6	5	4	3	2	1	0

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。

Bit 7 :　**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と USB のサスペンドステートと判定し、また、2.5µs 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。

Bit 6 :　**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。

Bit 5 :　**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。

Bit 4 :　**DisableHS**
GoChirp が“1”にセットされたときに、このビットが“1”にセットされているときには、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。

Bit 3 :　**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。

Bit 2 : **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。

Bit 1 : **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。

Bit 0 : **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。

USB Status
USB[0x0A4]　初期値 = 0xXX

VBUS(R)	FSxHS	–	–	–	–	***LineState[1:0]*** (R)	
7	6	5	4	3	2	1	0

デバイスに関するステータスを表示します。

Bit 7 : **VBUS**
VBUS 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。

Bit 6 : **FSxHS**
現在の動作モードを示します。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 時に読むことができ、ACT_DEVICE 時に書くことができます。

Bit [5:2] : **Reserved**

Bit [1:0] : **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。
XcvrControl レジスタの TermSelect ビットが“1”であるとき（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” であるときは、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

Xcvr Control
USB[0x0A5]　　初期値 = 0x41

Term Select	XcvrSelect	–	–	–	–	OpMode[1:0]	
7	6	5	4	3	2	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。

Bit 7 :　**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。

Bit 6 :　**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。

Bit [5:2] :　**Reserved**

Bit [1:0] :　**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とき（※）、USB のサスペンド状態になるとき、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とき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ときには、このレジスタを"41h"にセットすることを推奨します。

USB Test
USB[0x0A6]　　初期値 = 0x00

EnHS_ Test	–	–	–	Test_ SE0_NAK	Test_J	Test_K	Test_ Packet
7	6	5	4	3	2	1	0

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。

Bit 7：　**EnHS_Test**

このビットに“1”をセットすると、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、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、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、ご注意下さい。

Bit [6:4]：　**Reserved**

Bit 3：　**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。

Bit 2：　**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、XcvrControl レジスタの、TermSelect 及び XcvrSelect 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。

Bit 1：　**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、XcvrControl レジスタの、TermSelect 及び XcvrSelect 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。

Bit 0：　**Test_Packet**

このビットを“1”に設定し、EnHS_Test ビットに “1” をセットすることにより、Test_Packet テストモードに入ることができます。
このテストモードは EP0 以外の任意のエンドポイントで使用できますので、下記の設定を行って下さい。

1) エンドポイント EPx{x=a~h}の MaxPacketSize を 64 以上、転送方向を IN に設定し、EndpointNumber を“0xF”に設定して、使用可能として下さい。また、エンドポイント EPx{x=a~h}の FIFO を 64Byte 以上、割り当てて下さい。
2) エンドポイントの設定を、上記 EPx{x=a~h}の設定と重複しないようにして下さい。
   または、EPx{x=a~h}Config.EnEndpoint ビットをクリアして下さい。
3) EPx{x=a~h}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。
   EPx{x=a~h}IntStat レジスタの IN_TranErr ビットを“0”にクリアして下さい。
4) Test Packet の送信完了の毎に、IN_TranErr ステータスが“1”にセットされます。
   パケット送信テストモード時に FIFO に書き込むデータは以下の 53 バイトです。
   00h、00h、00h、00h、00h、00h、00h、00h、
   00h、AAh、AAh、AAh、AAh、AAh、AAh、AAh、
   AAh、EEh、EEh、EEh、EEh、EEh、EEh、EEh、
   EEh、FEh、FFh、FFh、FFh、FFh、FFh、FFh、
   FFh、FFh、FFh、FFh、FFh、7Fh、BFh、DFh、
   EFh、F7h、FBh、FDh、FCh、7Eh、BFh、DFh、
   EFh、F7h、FBh、FDh、7Eh

テストパケット送出時に、SIE が PID と CRC を付加しますので、FIFO に書き込むデータは、USB 規格 Rev.2.0 に記載されているテストパケットデータのうち、DATA0 PID の次のデータから、CRC16 以外のデータまでとなります。

EPn Control USB[0x0A8]　　初期値 = 0xXX							
AllForce NAK(W) 7	EPrForce STALL(W) 6	AllFIFO_ Clr(W) 5	– 4	– 3	– 2	– 1	EP0FIFO_ Clr(W) 0

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。

Bit 7 :　**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。

Bit 6 :　**EPrForceSTALL**
エンドポイント EPx{x=a~h}の ForceSTALL ビットを“1”にセットします。

Bit 5 :　**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ときは、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 DMAx{x=0~3}がジョインされ、かつ、該当する DMA が起動中(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。

Bit [4:1] :　**Reserved**

Bit 0 :　**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 DMAx{x=0~1}がジョインされ、かつ、該当する DMA が起動中(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。

EPr FIFO Clr USB[0x0A9]　　初期値 = 0xXX							
EPh FIFO_Clr(W) 7	EPg FIFO_Clr(W) 6	EPf FIFO_Clr(W) 5	EPe FIFO_Clr(W) 4	EPd FIFO_Clr(W) 3	EPc FIFO_Clr(W) 2	EPb FIFO_Clr(W) 1	EPa FIFO_Clr(W) 0

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 DMAx{x=0~1}がジョインされ、かつ、該当する DMA が起動中(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。

Clr All EPn Join USB[0x0AA]　　初期値 = 0xXX							
– 7	ClrJoin FIFO_Stat(W) 6	– 5	– 4	ClrJoin DMA1(W) 3	ClrJoin DMA0(W) 2	ClrJoin CPU_Rd(W) 1	ClrJoin CPU_Wr(W) 0

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~h}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。

Bulk Only Control USB[0x0AC]　　初期値 = 0x00							
AutoForce NAK_CBW	–	–	–	–	GoCBW_ Mode	GoCSW_ Mode	–
7	6	5	4	3	2	1	0

バルクオンリーサポート機能を制御します。

Bit 7 :　**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。

Bit [6:3] :　**Reserved**

Bit 2 :　**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。

Bit 1 :　**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。

Bit 0 :　**Reserved**

Bulk Only Configuration USB[0x0AD]　　初期値 = 0x00							
EPh BulkOnly	EPg BulkOnly	EPf BulkOnly	EPe BulkOnly	EPd BulkOnly	EPc BulkOnly	EPb BulkOnly	EPa BulkOnly
7	6	5	4	3	2	1	0

バルクオンリーサポート機能を有効にします。

Bit [7:0] :　**EPx{a-h}BulkOnly**

このビットを“1”にセットすると、エンドポイント EPx{a-h}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 EPx{a-h}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 EPx{a-h}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。

EP0 SETUP 0
USB[0x0B0]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

EP0 SETUP 1
USB[0x0B1]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

EP0 SETUP 2
USB[0x0B2]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

EP0 SETUP 3
USB[0x0B3]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

EP0 SETUP 4
USB[0x0B4]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

EP0 SETUP 5
USB[0x0B5]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

EP0 SETUP 6
USB[0x0B6]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

EP0 SETUP 7
USB[0x0B7]　　初期値 = 0x00

EP0 SETUP_0~7[7:0](R)							
7	6	5	4	3	2	1	0

エンドポイント EP0 のセットアップステージで受信した 8 バイトのデータが、EP0SETUP_0 から順に格納されます。

EP0SETUP_0
BmRequestType がセットされます。

EP0SETUP_1
BRequest がセットされます。

EP0SETUP_2
Wvalue の下位 8 ビットがセットされます。

EP0SETUP_3
Wvalue の上位 8 ビットがセットされます。

EP0SETUP_4
WIndex の下位 8 ビットがセットされます。

EP0SETUP_5
WIndex の上位 8 ビットがセットされます。

EP0SETUP_6

WLength の下位 8 ビットがセットされます。

EP0SETUP_7

WLength の上位 8 ビットがセットされます。

USB Address USB[0x0B8]　初期値 = 0x00							
–	USB Address[6:0]						
7	6	5	4	3	2	1	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 機能が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。

Bit 7 :　**Reserved**

Bit [6:0] :　**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。

SETUP Control USB[0x0BA]　初期値 = 0x00							
–	–	–	–	–	–	–	Protect EP0
7	6	5	4	3	2	1	0

コントロール転送関係の設定をします。

Bit [7:1] :　**Reserved**

Bit 0 :　**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 EP0SETUP_0～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 EP0ControlIN,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。

Frame Number L USB[0x0BE]　初期値 = 0x00							
FrameNumber[7:0](R)							
7	6	5	4	3	2	1	0

Frame Number H USB[0x0BF]　初期値 = 0x00							
FN_ Invalid(R)	–	–	–	–	FrameNumber[10:8] (R)		
7	6	5	4	3	2	1	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。

Bit 7 :　**Fn_Invalid**
受信した SOF パケットにエラーが発生したときに、このビットが “1” にセットされます。

Bit [6:3] :　**Reserved**

Bit [2:0] :　**FrameNumber[10:8]**

Bit [7:0] :　**FrameNumber[7:0]**
受信した SOF パケットの FrameNumber が表示されます。

EP0 Max Size
USB[0x0C0]　初期値 = 0x40

–	EP0 Max Size[6:3]				–	–	–
7	6	5	4	3	2	1	0

エンドポイント EP0 の設定を行います。

Bit 7 :　**Reserved**

Bit [6:3] :　**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

Bit [2:0] :　**Reserved**

EP0 Control
USB[0x0C1]　初期値 = 0x00

INxOUT	–	–	–	–	–	–	Reply Descriptor
7	6	5	4	3	2	1	0

エンドポイント EP0 の設定を行います。

Bit 7 :　**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、EP0ControlIN 及び 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 EP0ControlIN または 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。

Bit [6:1] :　**Reserved**

Bit 0 :　**ReplyDescriptor**
Descriptor 返信機能を実行します。
このビットが“1”にセットされると、エンドポイント EP0 の IN トランザクションに応答して、FIFO から Descriptor データを、MaxPacketSize 分返信します。Descriptor データは、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、FIFO_IntStat レジスタの DescriptorCmp ビットと 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。

EP0 Control IN
USB[0x0C2]　　初期値 = 0x00

–	En ShortPkt	–	Toggle Stat(R)	Toggle Set(W)	Toggle Clr(W)	Force NAK	Force STALL
7	6	5	4	3	2	1	0

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。

Bit 7 :　**Reserved**

Bit 6 :　**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。

Bit 5 :　**Reserved**

Bit 4 :　**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。

Bit 3 :　**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。

Bit 2 :　**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。

Bit 1 :　**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とき、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。

Bit 0 :　**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。

EP0 Control OUT
USB[0x0C3]　　初期値 = 0x00

Auto ForceNAK	–	–	Toggle Stat(R)	Toggle Set(W)	Toggle Clr(W)	Force NAK	Force STALL
7	6	5	4	3	2	1	0

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。

Bit 7 :　**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。

Bit [6:5] :　**Reserved**

Bit 4 :　**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。

Bit 3 :　**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。

Bit 2 :　**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。

Bit 1 :　**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。

Bit 0 :　**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。

EP0 Join USB[0x0C5]　　初期値 = 0x00							
–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

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。

Bit 7 :　**Reserved**

Bit 6 :　**JoinFIFO_Stat**
エンドポイント EP0 の FIFO の Full 及び Empty の状態を、FIFO_IntStat.FIFO_Full、FIFO_IntStat.FIFO_Empty 及び FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit [5:4] :　**Reserved**

Bit 3 :　**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。

Bit 2 :　**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。

Bit 1 :　**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_H,L 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。

Bit 0 :　**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_H,L 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、DMAx_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_H,L、FIFO_ByteRd、FIFO_Wr_H,L 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。

EPa Max Size L
USB[0x0D0]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPb Max Size L
USB[0x0D8]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPc Max Size L
USB[0x0E0]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPd Max Size L
USB[0x0E8]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPe Max Size L
USB[0x0F0]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPf Max Size L
USB[0x0F8]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPg Max Size L
USB[0x100]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPh Max Size L
USB[0x108]　　初期値 = 0x00

MaxSize[7:3]					–	–	–
7	6	5	4	3	2	1	0

EPa Max Size H
USB[0x0D1]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

EPb Max Size H
USB[0x0D9]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

EPc Max Size H
USB[0x0E1]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

EPd Max Size H
USB[0x0E9]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

EPe Max Size H
USB[0x0F1]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

EPf Max Size H
USB[0x0F9]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

EPg Max Size H
USB[0x101]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

EPh Max Size H
USB[0x109]　　初期値 = 0x00

–	–	–	–	–	–	MaxSize[9:8]	
7	6	5	4	3	2	1	0

MaxPacketSize を設定します。

Bit [7:3] :　**MaxSize[7:3]**

Bit [2:0] :　**Reserved**

Bit [7:2] :　**Reserved**

Bit [1:0] :　**MaxSize[9:8]**
エンドポイント EPx{a-h}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。

EPa Configuration
USB[0x0D2]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

EPb Configuration
USB[0x0DA]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

EPc Configuration
USB[0x0E2]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

EPd Configuration
USB[0x0EA]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

EPe Configuration
USB[0x0F2]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

EPf Configuration
USB[0x0FA]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

EPg Configuration
USB[0x102]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

EPh Configuration
USB[0x10A]　　初期値 = 0x00

INxOUT	IntEP_ Mode	En Endpoint	–	EndpointNumber[3:0]			
7	6	5	4	3	2	1	0

エンドポイント EPx{a-h}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。

Bit 7：　**INxOUT**
エンドポイントの転送方向を設定します。

Bit 6：　**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。

Bit 5：　**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”のときは、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。

Bit 4：　**Reserved**

Bit [3:0]：　**EndpointNumber[3:0]**
0x1～0xF の任意のエンドポイントナンバーを設定します。

EPa Control
USB[0x0D4]　初期値 = 0x00

Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

EPb Control
USB[0x0DC]　初期値 = 0x00

Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

EPc Control
USB[0x0E4]　初期値 = 0x00

Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

EPd Control
USB[0x0EC]　初期値 = 0x00

Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

EPe Control							
USB[0x0F4]　初期値 = 0x00							
Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

EPf Control							
USB[0x0FC]　初期値 = 0x00							
Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

EPg Control							
USB[0x104]　初期値 = 0x00							
Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

EPh Control							
USB[0x10C]　初期値 = 0x00							
Auto ForceNAK 7	EnShort Pkt 6	DisAF_ NAK_Short 5	Toggle Stat(R) 4	Toggle Set(W) 3	Toggle Clr(W) 2	Force NAK 1	Force STALL 0

エンドポイント EPx{a-h}の動作設定を行います。

Bit 7 : **AutoForceNAK**
エンドポイント EP x{a-h}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。

Bit 6 : **EnShortPkt**
このビットを“1”にセットすることで､エンドポイント EP x{a-h}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。

Bit 5 : **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。

Bit 4 : **ToggleStat**
エンドポイント EP x{a-h}のトグルシーケンスビットの状態を示します。

Bit 3 : **ToggleSet**
エンドポイント EP x{a-h}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。

Bit 2 :　**ToggleClr**
エンドポイント EP x{a-h}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。

Bit 1 :　**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 EP x{a-h}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。

Bit 0 :　**ForceSTALL**
このビットを“1”にセットすると、エンドポイント EP x{a-h}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。

EPa Join
USB[0x0D5]　初期値 = 0x00

–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

EPb Join
USB[0x0DD]　初期値 = 0x00

–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

EPc Join
USB[0x0E5]　初期値 = 0x00

–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

EPd Join
USB[0x0ED]　初期値 = 0x00

–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

EPe Join
USB[0x0F5]　初期値 = 0x00

–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

EPf Join
USB[0x0FD]　初期値 = 0x00

–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

EPg Join USB[0x105]　初期値 = 0x00							
–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

EPh Join USB[0x10D]　初期値 = 0x00							
–	Join FIFO_Stat	–	–	Join DMA1	Join DMA0	Join CPU_Rd	Join CPU_Wr
7	6	5	4	3	2	1	0

エンドポイント EPx{a-h}とデータ転送を行うポートを指定します。

Bit 7 :　**Reserved**

Bit 6 :　**JoinFIFO_Stat**
エンドポイント EP x{a-h}の FIFO の Full 及び Empty の状態を、FIFO_IntStat.FIFO_Full、FIFO_IntStat.FIFO_Empty 及び FIFO_IntStat.FIFO_NotEmpty でモニタできるようにします。

Bit [5:4] :　**Reserved**

Bit 3 :　**JoinDMA1**
エンドポイント EP x{a-h}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。

Bit 2 :　**JoinDMA0**
エンドポイント EP x{a-h}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。

Bit 1 :　**JoinCPU_Rd**
エンドポイント EP x{a-h}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_H,L 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。

Bit 0 :　**JoinCPU_Wr**
エンドポイント EP x{a-h}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_H,L 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、DMAx_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_H,L、EPnCHnFIFO_ByteRd、FIFO_Wr_H,L 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。

A2.5.2.7　USB FIFO 設定レジスタ

EPa Start Address L
USB[0x110]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPb Start Address L
USB[0x112]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPc Start Address L
USB[0x114]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPd Start Address L
USB[0x116]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPe Start Address L
USB[0x118]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPf Start Address L
USB[0x11A]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPg Start Address L
USB[0x11C]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPh Start Address L
USB[0x11E]　　初期値 = 0x00

7	6	5	4	3	2	1	0
Start Adrs[7:2]						—	—

EPa Start Address H
USB[0x111]　　初期値 = 0x00

7	6	5	4	3	2	1	0
—	—	—	Start Adrs[12:8]				

EPb Start Address H
USB[0x113]　　初期値 = 0x00

7	6	5	4	3	2	1	0
—	—	—	Start Adrs[12:8]				

EPc Start Address H
USB[0x115]　　初期値 = 0x00

7	6	5	4	3	2	1	0
–	–	–	Start Adrs[12:8]				

EPd Start Address H
USB[0x117]　　初期値 = 0x00

7	6	5	4	3	2	1	0
–	–	–	Start Adrs[12:8]				

EPe Start Address H
USB[0x119]　　初期値 = 0x00

7	6	5	4	3	2	1	0
–	–	–	Start Adrs[12:8]				

EPf Start Address H
USB[0x11B]　　初期値 = 0x00

7	6	5	4	3	2	1	0
–	–	–	Start Adrs[12:8]				

EPg Start Address H
USB[0x11D]　　初期値 = 0x00

7	6	5	4	3	2	1	0
–	–	–	Start Adrs[12:8]				

EPh Start Address H
USB[0x11F]　　初期値 = 0x00

7	6	5	4	3	2	1	0
–	–	–	Start Adrs[12:8]				

EPx{a-h}で使用する FIFO の領域設定を行います。

Bit [7:2] :　**StartAdrs[7:2]**

Bit [1:0] :　**Reserved**

Bit [7:5] :　**Reserved**

Bit [4:0] :　**StartAdrs[12:8]**
エンドポイント EP x{a-h}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。
チャネル EP x{a-g} に割り当てられる領域は、次の EP の先頭アドレスに設定されたアドレスの 1 バイト前までとなります。EPh に割り当てられる領域は、EP_EndAdrs に設定されたアドレスの 1 バイト前までとなります。
EP x{a-h} StartAdrs を設定した後は、必ず EPrFIFO_Clr レジスタの EP x{a-h}FIFO_Clr ビットを“1”にしてエンドポイント EP x{a-h}の FIFO をクリアして下さい。
なお、ここで設定した領域より、EP x{a-h} 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。

EP End Address L
USB[0x120]　　初期値 = 0x00

End Adrs[7:2]						—	—
7	6	5	4	3	2	1	0

EP End Address H
USB[0x121]　　初期値 = 0x12

—	—	—	End Adrs[12:8]				
7	6	5	4	3	2	1	0

エンドポイント FIFO の領域設定を行います。

Bit [7:2] :　**EndAdrs[7:2]**

Bit [1:0] :　**Reserved**

Bit [7:5] :　**Reserved**

Bit [4:0] :　**EndAdrs[12:8]**
エンドポイント EPh 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 4 バイト単位での指定になります。
リセット時の初期値は搭載している FIFO RAM の最終アドレスが設定されています。
ここで設定した領域より、エンドポイントの 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。
エンドポイント数の変化による本レジスタの割り付けアドレスの変更はありません。
リセット時の初期値は FIFO 領域の大きさを示すアドレス値になりますが値の変更は可能です。

Descriptor Address L
USB[0x124]　　初期値 = 0x00

Desc Adrs[7:0]							
7	6	5	4	3	2	1	0

Descriptor Address H
USB[0x125]　　初期値 = 0x00

—	—	—	—	Desc Adrs[11:8]			
7	6	5	4	3	2	1	0

Descriptor Adress を指定します。

Bit [7:0] :　**DescAdrs[7:0]**

Bit [7:4] :　**Reserved**

Bit [3:0] :　**DescAdrs[11:8]**
Descriptor 返信機能における、Descriptor 返信動作開始時の FIFO の先頭アドレスを指定します。
Descriptor Address は、Descriptor 返信機能に対して FIFO 領域を割り当てるものではありません。
Descriptor Address は、FIFO RAM 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、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、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)までの間が適切です。
Desriptor Address を参照する場合は、DescAdrs_H,DescAdrs_L の順に読み出して下さい。

Descriptor Size L							
USB[0x126]　初期値 = 0x00							
Desc Size[7:0]							
7	6	5	4	3	2	1	0

Descriptor Size H							
USB[0x127]　初期値 = 0x00							
–	–	–	Desc Size[12:8]				
7	6	5	4	3	2	1	0

Descriptor Size を指定します。

Bit [7:0] :　**DescSize[7:0]**

Bit [7:5] :　**Reserved**

Bit [1:0] :　**DescSize[12:8]**

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 0x1FFF までの値を指定することが出来ます。Descriptor 返信時には、エンドポイント EP0 における IN トランザクション完了毎に、送信データ数の分だけ DescSize は更新されます。
Descriptor 返信機能用の FIFO 領域は、明示的には割り当てませんので、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。
Desriptor Size を参照する場合は、DescSize_H,DescSize_L の順に読み出して下さい。

DMA0 FIFO Control							
USB[0x128]　初期値 = 0x00							
FIFO_ Running(R)	AutoEn Short	–	–	–	–	–	–
7	6	5	4	3	2	1	0

DMA1 FIFO Control							
USB[0x12A]　初期値 = 0x00							
FIFO_ Running(R)	AutoEn Short	–	–	–	–	–	–
7	6	5	4	3	2	1	0

DMA0/1 転送時の、FIFO の状態の表示及び設定を行います。

Bit 7 :　**FIFO_Running**

DMA0/1 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。DMA0/1 を起動すると“1”にセットされ、DMA0/1 が終了した後、FIFO が空になると“0”にクリアされます。

Bit 6 :　**AutoEnShort**

DMA0/1 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。
DMA0/1 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。

Bit [5:0] :　**Reserved**

FIFO Rd 0
USB[0x130]　　初期値 = 0xXX

FIFO Rd0[7:0](R)							
7	6	5	4	3	2	1	0

FIFO Rd 1
USB[0x131]　　初期値 = 0xXX

FIFO Rd1[7:0] (R)							
7	6	5	4	3	2	1	0

Bit [7:0]：　**FIFO_Rd0[7:0]**

Bit [7:0]：　**FIFO_Rd1[7:0]**
EPx{x=0,a~h}Join.JoinCPU_Rd ビットがセットされているエンドポイントの FIFO データを読み出せます。
8bit mode 時には FIFO_Rd_H,L 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。

FIFO Wr 0
USB[0x132]　　初期値 = 0xXX

FIFO Wr0[7:0](W)							
7	6	5	4	3	2	1	0

FIFO Wr 1
USB[0x133]　　初期値 = 0xXX

FIFO Wr1[7:0] (W)							
7	6	5	4	3	2	1	0

Bit [7:0]：　**FIFO_Wr0[7:0]**

Bit [7:0]：　**FIFO_Wr1[7:0]**
EPx{x=0,a~h}Join.JoinCPU_Wr ビットがセットされているエンドポイントの FIFO へデータを書き込めます。
8bit mode 時には FIFO_Wr_H,L いずれのレジスタにアクセスしても FIFO へデータを書き込めます。
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 へ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。

FIFO Rd Remain L
USB[0x134]　　初期値 = 0x00

RdRemain[7:0](R)							
7	6	5	4	3	2	1	0

FIFO Rd Remain H
USB[0x135]　　初期値 = 0x00

RdRemain Valid(R)	–	–	RdRemain[12:8] (R)				
7	6	5	4	3	2	1	0

Bit [7:0]：　**RdRemain[7:0]**

Bit 7 :　**RdRemainValid**

EPx{x=0,a~h}Join.JoinCPU_Rd ビットによって、CPU I/F にエンドポイント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。

Bit [6:5] :　**Reserved**

Bit [4:0] :　**RdRemain[12:8]**

EPx{x=0,a~h}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。

FIFO Wr Remain L
USB[0x136]　初期値 = 0x00

WrRemain[7:0] (R)							
7	6	5	4	3	2	1	0

FIFO Wr Remain H
USB[0x137]　初期値 = 0x00

–	–	–	WrRemain[12:8] (R)				
7	6	5	4	3	2	1	0

Bit [7:0] :　**WrRemain[7:0]**

Bit [7:5] :　**Reserved**

Bit [4:0] :　**WrRemain[12:8]**

EPx{x=0,a~h}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 により CPU I/F に接続しているチャネル 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。

FIFO Byte Rd
USB[0x138]　初期値 = 0xXX

FIFO ByteRd[7:0] (R)							
7	6	5	4	3	2	1	0

Bit [7:0] :　**FIFO_ByteRd[7:0]**

EPx{x=0,a~h}Join.JoinCPU_Rd ビットがセットされているエンドポイント 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。

RAM Rd Adrs L							
USB[0x140]　初期値 = 0x00							
RAM Read Address[7:2]						–	–
7	6	5	4	3	2	1	0

RAM Rd Adrs H							
USB[0x141]　初期値 = 0x00							
–	–	–	RAM Read Address[12:8]				
7	6	5	4	3	2	1	0

Bit [7:2]：　**RAM_RdAdrs[8:2]**

Bit [1:0]：　**Reserved**

Bit [7:5]：　**Reserved**

Bit [4:0]：　**RAM_RdAdrs[12:8]**
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。

RAM Rd Control							
USB[0x142]　初期値 = 0x00							
RAM_GoRd CBW_CSW	RAM_GoRd	–	–	–	–	–	–
7	6	5	4	3	2	1	0

Bit 7：　**RAM_GoRdCBW_CSW**
CBW 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。
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。
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。

Bit 6：　**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。

Bit [5:0]：　**Reserved**

RAM Rd Count							
USB[0x143]　初期値 = 0x00							
–	–	RAM Rd Count[5:2]				–	–
7	6	5	4	3	2	1	0

Bit [7:6] :　**Reserved**

Bit [5:2] :　**RAM_RdCount [5:2]**
RAM_Rd 機能を用いて RAM_Rd_xx{xx=00~1F}レジスタにリードするデータ数を設定します。RAM_RdAdrs_H,L レジスタを設定した後、このレジスタをセットし、RAM_RdControl レジスタのビットをセットして RAM_Rd 機能を開始して下さい。このレジスタの値は、RAM_Rd 機能作動中は内部動作に応じて変化します。従って、一旦 RAM_RdControl レジスタのビットをセットして、RAM_Rd 機能を開始させた後は CPU_IntStat.RAM_RdCmp ビットがセットされるまでこのレジスタの値を読み出さないで下さい。RAM_Rd 機能作動中にこのレジスタを読み出した場合の値は保証されません。また、RAM_Rd 機能作動中にこのレジスタに書き込んだ場合、誤動作の原因となりますのでご注意ください。
このレジスタの最大設定数は 32 バイトです。32 バイトを超えるデータ数の設定は誤動作の原因となりますのでご注意ください。

Bit [1:0] :　**Reserved**

RAM Wr Address L							
USB[0x144]　初期値 = 0x00							
RAM WrAdrs[7:0]							
7	6	5	4	3	2	1	0

RAM Wr Address H							
USB[0x144]　初期値 = 0x00							
–	–	–	RAM WrAdrs[12:8]				
7	6	5	4	3	2	1	0

RAM_WrDoor_H,L 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。

Bit [7:0] :　**RAM_WrAdrs[7:0]**

Bit [7:5] :　**Reserved**

Bit [4:0] :　**RAM_WrAdrs[12:8]**
RAM への書き込みを行う際のアドレスを指定します。RAM_WrDoor_H,L 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_H,L レジスタ書き込み直後には正確な RAM_WrAdrs を確認することは出来ませんので、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_H,L 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。

RAM Wr Door 0
USB[0x146]　　初期値 = 0xXX

RAM WrDoor0[7:0](W)							
7	6	5	4	3	2	1	0

RAM Wr Door 1
USB[0x147]　　初期値 = 0xXX

RAM WrDoor1[7:0] (W)							
7	6	5	4	3	2	1	0

Bit [7:0]：　**RAM_WrDoor0[7:0]**

Bit [7:0]：　**RAM_WrDoor1[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_H,L 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_H,L 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。

RAM Rd 00～RAM Rd 1F
USB[0x150]~USB[0x16F]　　初期値 = 0x00

RAM Rd 00[7:0](R) ～ RAM Rd 1F[7:0] (R)							
7	6	5	4	3	2	1	0

Bit [7:0]：　**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。

A2.6　機能説明

A2.6.1　初期設定

USB コントローラを使用する際には、下記の初期設定を行ってください。

A2.6.1.1　USB コントローラへのアクセス設定

本 USB コントローラへアクセスするには以下の初期設定が必要です。

① APB バス Wait の選択
CPU から USB コントローラへのアクセスウエイトを APB ブリッジの“APB WAIT2 レジスタ （APBWAIT2）”の bit[31:30]に+2wait 以上で設定してください。
アドレス 0xFFFE_0008/bit[31:30] = 10

② I/O クロックの On （CPU⇔USB コントローラ）
システムコントローラの“IO Clock Control Register （IOCLKCTL）”の bit1 に“1”を書き込みます。
アドレス ：0xFFFF_D014/bit[1] = 1

③ USB コントローラ内部のクロック On
USB コントローラ内部で使用するクロックを有効にします。
システムコントローラの“IO Clock Control Register （MISC）”の bit1 に“1”を書き込むことで、USB コントローラ内のレジスタは USB コントローラ内蔵の PLL480 の 8 分の 1 分周の 60MHz クロックで動作します。
アドレス ：0xFFFF_D06C/bit[1] = 1
bit[1] ： 1 60MHz ON
0 60MHz OFF

A2.6.1.2　割り込みの設定

本 USB コントローラからの割り込みは、割り込みコントローラ（INTC）の通常割り込み IRQ29 に割り付け割れています。USB コントローラからの割り込み信号は Low アクティブのレベル信号で出力されますが、IC の内部において INTC の入力時に論理反転されているため、INTC の設定としては High アクティブのレベル信号として扱います。

① レベルレジスタの設定
割り込みコントローラの“IRQxx レベルレジスタ”の bit[29]に 0 を設定してください。
アドレス ：0xFFFF_F0A0/bit[29] = 0

② 極性レジスタの設定
割り込みコントローラの“IRQxx 極性レジスタ”の bit[29]に 1 を設定してください。
アドレス ：0xFFFF_F0A4/bit[29] = 1

A2.6.1.3　Macro Config1[0x37]に関する注意事項

本レジスタは本 LSI 用に最適化されています。必要のない限り、設定を変更しないでください。

bit[7]	0：負論理
bit[6]	0：1/0 モード
bit[5]	0：負論理
bit[4]	0：負論理
bit[3]	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。
bit[2]	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。
bit[1:0]	1x：16bit BE mode （固定）

本レジスタの設定（bit[2]）により、初期状態では Little Endian 形式となっています。

A2.6.2　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。

A2.6.2.1　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、8 本の汎用エンドポイント（EPa-h）を持ちます。エンドポイント EPa-h 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ース（以下、USB 定義インタフェース）の管理機能を提供しません。USB 定義インタフェースは､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 定義インタフェースを構成してください。

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

Tbl. A2.6.1 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。

Tbl. A2.6.1　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	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 バイトの領域が割当てられます。

Tbl. A2.6.2 に汎用エンドポイント（EPa-h）の基本設定項目を示します。エンドポイント EPa-h 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、8 本までの独立したエンドポイントを使用できます。USB 定義インタフェースの定義内容に合わせて適宜設定し、また有効にすることによって、USB 定義インタフェースを構成して下さい。
エンドポイント EPa-h 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。

Tbl. A2.6.2　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	EPx(x=a-h)Config.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	EPx(x=a-h)MaxSize_H EPx(x=a-h)MaxSize_L	各エンドポイントのマックスパケットサイズを 8,16,32,64,512 バイト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 8/16/32/64 バイトのいずれか、HS モード時は 512 バイトに設定してください。
エンドポイントナンバー	EPx(x=a-h)Config.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	EPx(x=a0h)Config.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、方向に関わらず“0”に設定してください。 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設定をします。OUT 転送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”にセットしてください。
エンドポイント有効	EPx(x=a-h)Config.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 定義インタフェースが有効にされたときに設定してください。
FIFO 領域	EPx(x=a-h)StartAdrs_H EPx(x=a-h)StartAdrs_L EP_EndAdrs_H EP_EndAdrs_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ください。また、FIFO 領域のサイズがデータ転送のスループットに影響します。 FIFO の領域割り当ての詳細は、機能説明の FIFO の項を参照してください。

A2.6.2.2　トランザクション

LSI は H/W でトランザクション実行機能と、ファームウェアに対するトランザクション実行のためのインタフェースを提供します。ファームウェアに対するインタフェースは、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ときに、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 インタフェース（DMA リードまたはレジスタリード)により、FIFO からデータを読み出して FIFO に空き領域を作り出すことによって、OUT トランザクションを自動的に連続して実行させることが出来ます。また、IN のエンドポイントであれば、ファームウェアは、CPU インタフェース（DMA ライトまたはレジスタライト）により、FIFO にデータを書き込んで FIFO に有効データを作り出すことによって、IN トランザクションを自動的に連続して実行させることが出来ます。

Tbl. A2.6.3 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。

Tbl. A2.6.3　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	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	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	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	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	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	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	EP0ControlIN.ToggleStat 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	EP0ControlIN.ToggleSet 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	EP0ControlIN.ToggleClr 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	EP0ControlIN.ForceNAK EP0ControlOUT.ForceNAK	FIFO のデータ数/空き数に関わらず、IN または OUT（PING を含む）トランザクションに NAK 応答します。
STALL 応答	EP0ControlIN.ForceSTALL EP0ControlOUT.ForceSTALL	IN または 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	EP0ControlOUT.AutoForceNAK	OUT トランザクションの完結ごとに、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	EP0IntStat.OUT_ShortACK EP0IntStat.IN_TranACK EP0IntStat.OUT_TranACK EP0IntStat.IN_TranNAK EP0IntStat.OUT_TranNAK EP0IntStat.IN_TranErr 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	FIFO_IntStat.DescriptorCmp	デスクリプタ j 同応答のデータステージが終了したことを示します。

Tbl. A2.6.4 に汎用エンドポイント EPa-h 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。

Tbl. A2.6.4　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	EPx{x=a-h}Control.AutoForceNAK	OUT トランザクションの完結ごとに、そのエンドポイントの EPx{x=a-h}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	EP x{x=a-h}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	EPx{x=a-h}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 EPx{a-h}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	EP x{x=a-h}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	EP x{x=a-h}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	EP x{x=a-h}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	EP x{x=a-h}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	EP x{x=a-h}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	EP x{x=a-h}IntStat.OUT_ShortACK EP x{x=a-h}IntStat.IN_TranACK EP x{x=a-h}IntStat.OUT_TranACK EP x{x=a-h}IntStat.IN_TranNAK EP x{x=a-h}IntStat.OUT_TranNAK EP x{x=a-h}IntStat.IN_TranErr EP x{x=a-h}IntStat.OUT_TranErr	トランザクションの結果を示します。

A2.6.2.2.1　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。（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 EP0SETUP_0～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、EP0ControlIN レジスタ及び 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、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、EP0ControlIN レジスタまたは 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。

Fig. A2.6.1 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 EP0SETUP_0～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。

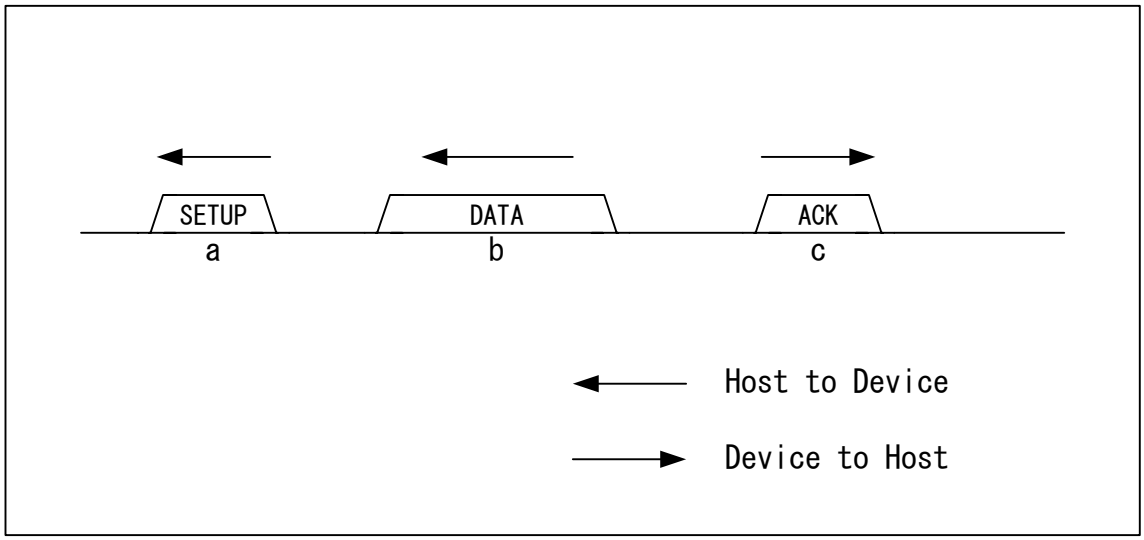


Fig. A2.6.1　SETUP トランザクション

A2.6.2.2.2　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 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_TranACK ステータス (EPx{x=0,a-h}IntStat.OUT_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。

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 (EPx{x=0,a-h}IntStat.OUT_ShortACK ビット)を発行します。また、EPx{x=a-h}Control.DisAF_NAK_Short ビットがクリアされていると、そのエンドポイントの、EPx{x=a-h}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_TranErr ステータス (EPx{x=0,a-h}IntStat.OUT_TranErr 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_TranNAK ステータス (EPx{x=0,a-h}IntStat.OUT_TranNAK ビット) を発行します。FIFO は更新されません。

Fig. A2.6.2 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。LSI はこのデータを、該当するエンドポイントの FIFO に書き込みます。(c) LSI はデータを受信できると、自動的に ACK 応答します。また、自動設定するレジスタを設定し、ファームウェアに対しステータスを発行します。

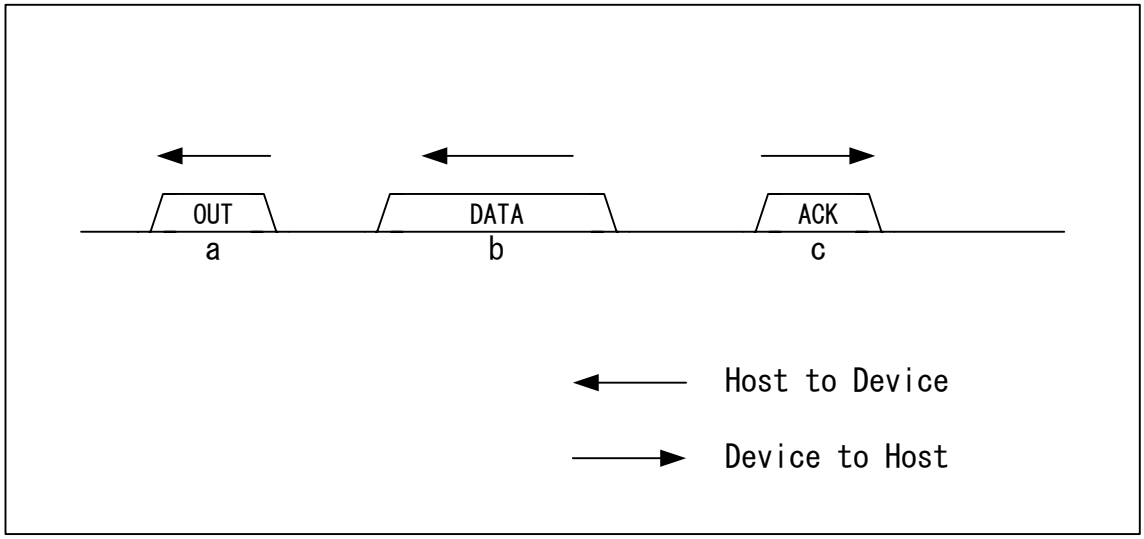


Fig. A2.6.2　OUT トランザクション

A2.6.2.2.3　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有るか、または、ファームウェアによってショートパケットの送信が許可されていると、IN トランザクションに応答して、データパケットを返信します。

ショートパケット（データ長ゼロのパケットを含む）の送信許可は、EP0ControlIN.EnShortPkt ビットまたは EPx{x=a-h}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、EP0ControlIN. ForceNAK ビットがセットされます。

データ返信した IN トランザクションで、ACK を受信すると、トランザクションを完結し、ファームウェアに対し、IN_TranACK ステータス（EPx{x=0,a-h}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（EPx{x=0,a-h}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（EPx{x=0,a-h}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。

Fig. A2.6.3 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）LSI は、この IN トランザクションに応答できる場合、マックスパケットサイズ以内のデータパケットを送信します。（c）ホストは ACK 応答します。LSI は ACK を受信すると、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。

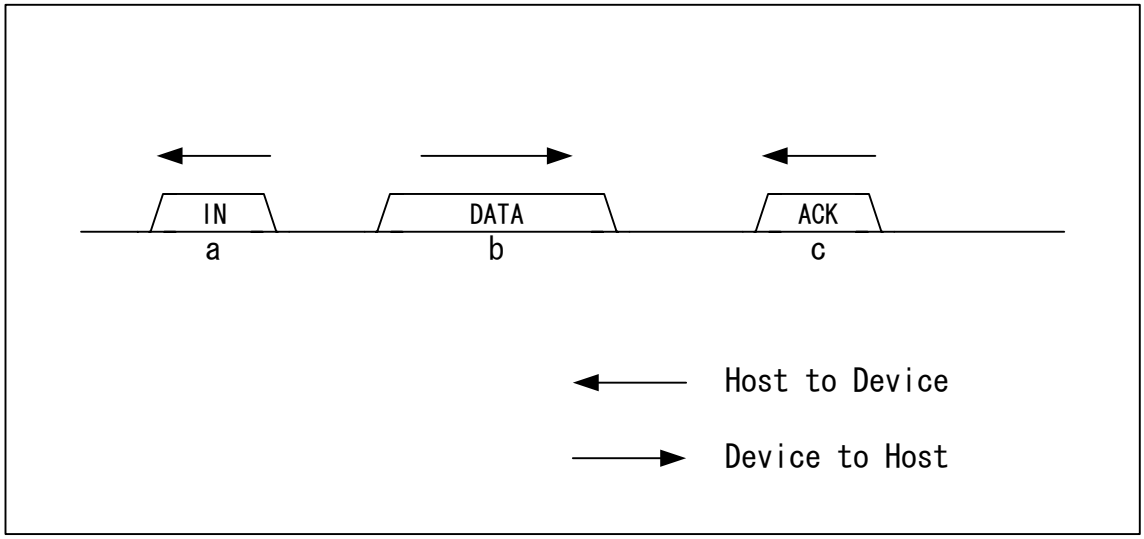


Fig. A2.6.3　IN トランザクション

A2.6.2.2.4　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（EPx{x=0,a-h}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。

Fig. A2.6.4 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) LSI は、FIFO にマックスパケットサイズ分の空きがある場合、この PING トランザクションに対し ACK 応答します。また、ファームウェアに対しステータスを発行します。

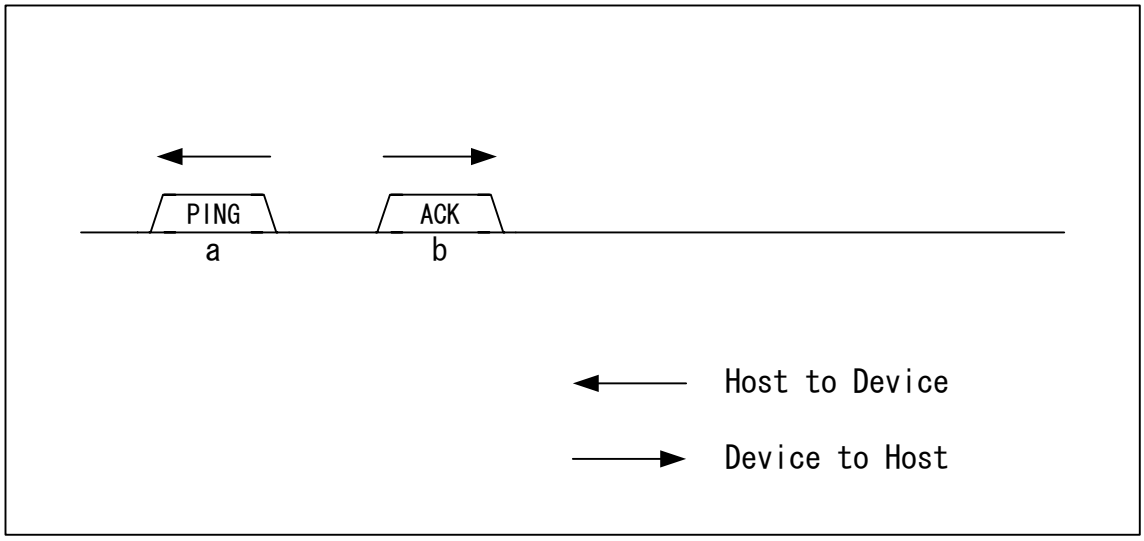


Fig. A2.6.4　PING トランザクション

A2.6.2.3　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。

Fig. A2.6.5 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。デバイスのファームウェアはリクエストの内容を解析して、データステージに応答する準備をします。(b) ホストは OUT トランザクションを発行して、データステージを行い、デバイスはデータを受信します。(c) ホストは IN トランザクションを発行して、ステータスステージを行い、デバイスはデータ長ゼロのパケットを返信します。

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（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。

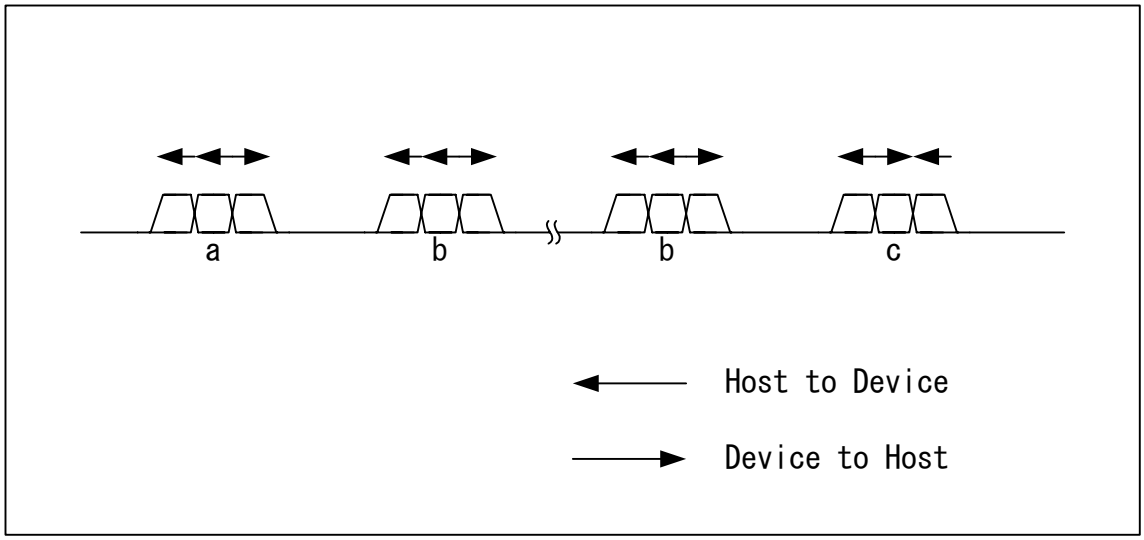


Fig. A2.6.5　データステージが OUT 方向のコントロール転送

Fig. A2.6.6 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。デバイスのファームウェアはリクエストの内容を解析して、データステージに応答する準備をします。(b) ホストは IN トランザクションを発行して、データステージを行い、デバイスはデータを送信します。(c) ホストは OUT トランザクションを発行して、ステータスステージを行い、デバイスは ACK 応答します。

ステータスステージへの移行は、ホストがデータステージと逆方向のトランザクションを発行することによってなされます。ファームウェアは、OUT_TranNAK ステータス（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。

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。

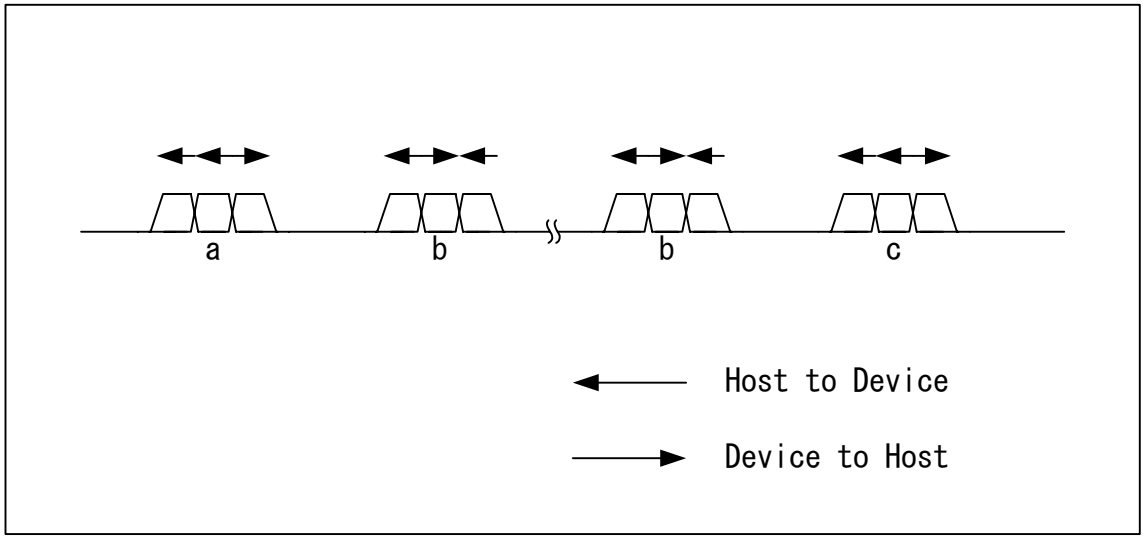


Fig. A2.6.6　データステージが IN 方向のコントロール転送

A2.6.2.3.1　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、EP0SETUP_0～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、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、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、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。

A2.6.2.3.2　データステージ／ステータスステージ

EP0SETUP_0～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、EP0Control レジスタの INxOUT をクリアして OUT 方向に設定し、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、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、EP0Control レジスタの INxOUT をセットして IN 方向に設定し、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、SETUP_Control.ProtectEP0 ビットがセットされています。

A2.6.2.3.3　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 EP0SETUP_0～EP0SETUP_7 レジスタによってリクエストの内容を確認し、有効な SetAddress()リクエストで有った場合には、ファームウェアに通知することなく、このリクエストのス

テータスステージの処理に移行します。ステータスステージが完了すると、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(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。

A2.6.2.3.4　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。
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、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、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、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応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、EP0ControlIN.EnShortPkt をセットし、全てのデータを返信するまで、IN トランザクションに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、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（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。

A2.6.2.4　バルク転送/インタラプト転送

汎用エンドポイント EPa-h におけるバルク転送、インタラプト転送は、データフロー（A2.6.2.5 参照）としても、連続する個々のトランザクション（A2.6.2.2 参照）としても制御できます。

A2.6.2.5　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。

A2.6.2.5.1　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、DMA による読み出し方法があります。
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、EPx{x=0,a-h}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て下さい。
DMA リードにより、FIFO のデータを読み出すには、EPx{x=0,a-h}Join.JoinDMAx{x=0,1}ビットにより、DMA のチャ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、外部 DMA コントローラ等により DMA 転送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、自動的に DMA を一時停止してフロー制御を行います。
FIFO にデータパケットを受信できる空き領域が有れば、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、EPx{x=a-h}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 EPx{x=a-h}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、EPx{x=a-h}Control.ForceNAK ビットをクリアして下さい。

A2.6.2.5.2　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 インタフェースによるレジスタ書き込みと、DMA による書き込み方法があります。

CPU インタフェースのレジスタライトにより、FIFO にデータを書き込むには、EPx{x=0,a-h}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。

DMA ライトにより、FIFO にデータを書き込むには、EPx{x=0,a-h}Join.JoinDMAx{x=0,1}ビットにより、DMA のチャ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、外部 DMA コントローラ等により DMA 転送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、自動的に DMA を一時停止してフロー制御を行います。

FIFO にマックスパケットサイズ以上のデータが有れば､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、DMA 書き込みが終了したときに、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。

A2.6.2.6　バルクオンリーサポート

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

BulkOnlyConfig.EPx{x=a-h}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。

A2.6.2.6.1　CBW サポート

ファームウェアは BulkOnly Transport Protocol のコマンドトランスポートを行うときに､CBW サポートを使用することが出来ます。BunlOnlyConfig.EPx{x=a-h}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、BulkOnlyContol.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(BulkIntStat.CBW_Cmp ビット)を発行します。また、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 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(BulkIntStat.CBW_LengthErr ビット)を発行します。また、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 BulkOnlyControl.GoCSW_Mode ビットがセットされていると、同時にクリアします。CBW_Err ステータスが発行された場合、BulkOnly Transport Protocol でフェーズミスマッチが発生していますので、ファームウェアはエンドポイントを STALL するなどして、通信の復旧を行ってください。

対象となるエンドポイントで EPx{x=a-h}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(BulkIntStat.CBW_Err ビット)を発行し、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 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(BulkIntStat.CBW_TranErr ビット)ステータスを発行します。この場合には、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 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。

A2.6.2.6.2　CSW サポート

ファームウェアは BulkOnly Transport Protocol のステータストランスポートを行うときに、CSW サポートを使用することが出来ます。BulkOnlyConfig.EPx{x=a-h}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結した場合には、ファームウェアに対し CSW 完了ステータス(BulkIntStat.CSW_Cmp ビット)を発行します。また、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 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(BulkIntStat.CSW_Err ビット)を発行します。このとき、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 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。

A2.6.2.7　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行います。実際に何が行われたかは、各割込み（DetectRESET、DetectSUSPEND、ChirpCmp、RestoreCmp）をチェックすることにより確認することができます。

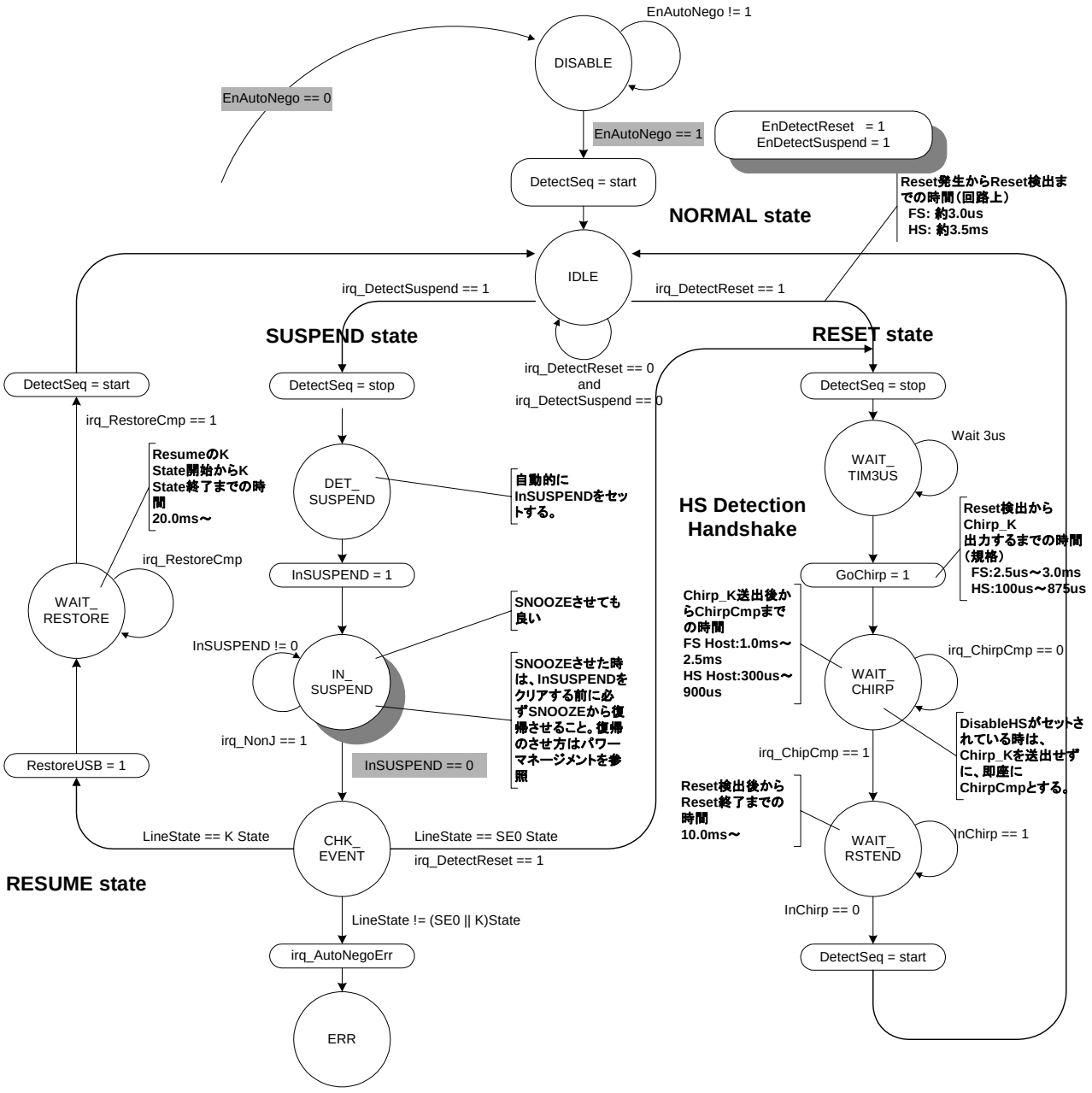


Fig. A2.6.7　オート・ネゴシエータ

A2.6.2.7.1　DISABLE

NegoControl.EnAutoNego ビットをクリアしているときに、このステートに入ります。
オート・ネゴシエーション機能を有効にするときには、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（SIE_IntEnb.EnDetectRESET）、サスペンド検出割り込み許可ビット（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、NegoControl.DisBusDetect ビットを絶対にセットしないでください。

A2.6.2.7.2　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 のときには、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 のときには、2.5μs 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、SIE_IntStat.DetectRESET ビット、または 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。

A2.6.2.7.3　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 3μs 後）、WAIT_CHIRP ステートに入ります。

A2.6.2.7.4　WAIT_CHIRP

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（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、0 を参照してください。
また、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（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、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、SIE_IntEnb.EnChirpCmp ビットをセットしてください。

A2.6.2.7.5　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。

A2.6.2.7.6　DET_SUSPEND

サスペンドと判断された場合に、自動的に NegoControl.InSUSPEND ビットがセットされ、IN_SUSPEND ステートに入ります。この 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するようになります。
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、パワーマネージメント機能（0）を参照してください。
また、このとき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 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。

A2.6.2.7.7　IN_SUSPEND

NonJ 割り込みステータス（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、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 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。

A2.6.2.7.8　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、NegoControl.RestoreUSB ビットをセットし、サスペンド前の転送スピード（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（SIE_IntStat.AutoNegoErr）ビットをセットし、ERR ステートに入ります。

A2.6.2.7.9　WAIT_RESTORE

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。

A2.6.2.7.10　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。

A2.6.2.7.11　各ネゴシエーション機能の単体説明

A2.6.2.7.11.1　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ときに、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、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、SIE_IntStat.DetectSUSPEND ビットがセットされます。

このとき、SIE_IntEnb.EnDetectSUSPEND ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み信号がアサートされますので、USB のサスペンドステートであると判断します。以下の図では、スヌーズを行ったときの動作を表しています）。

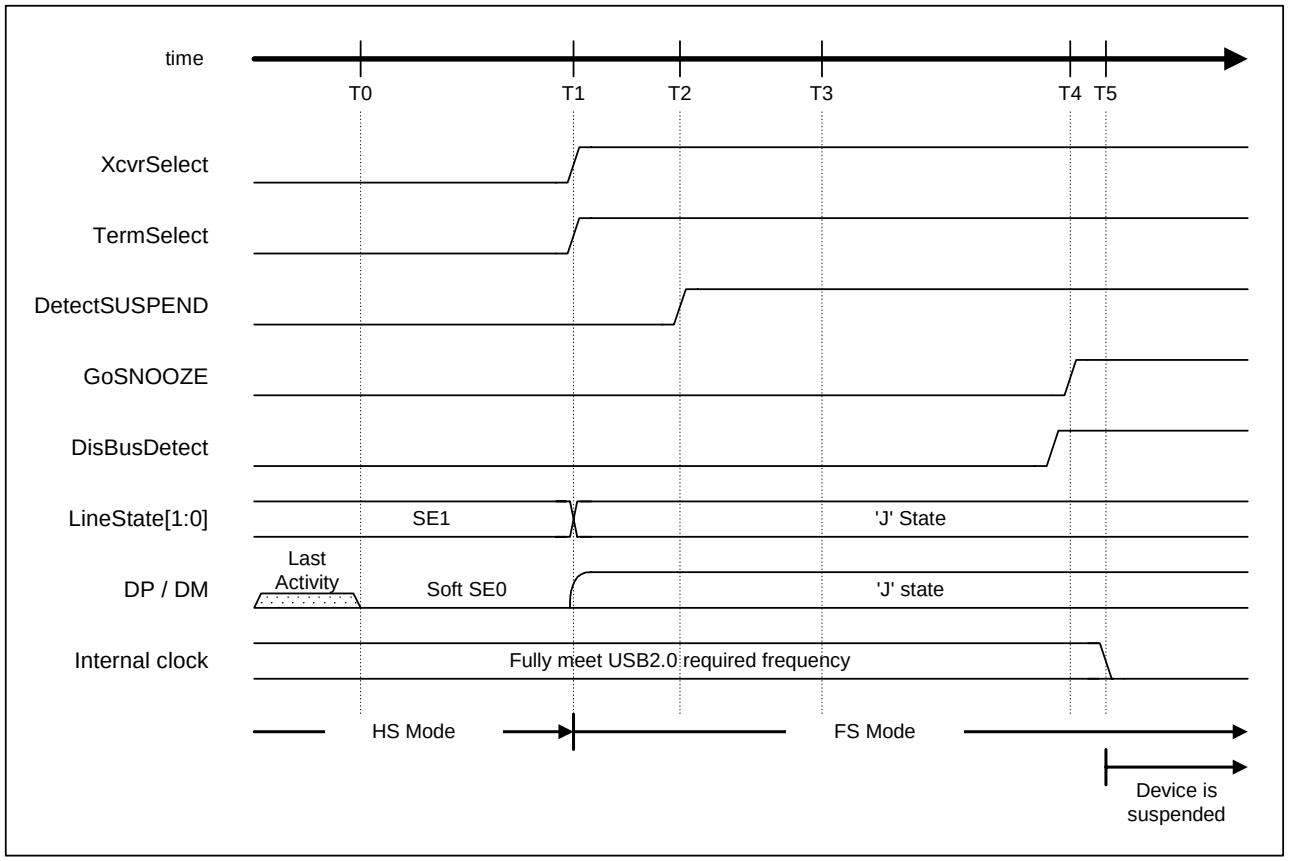


Fig. A2.6.8　Suspend Timing (HS mode)

Tbl. A2.6.5　Suspend Timing Values (HS mode)

Timing Parameter	Description	Values
T0	最後のバス・アクティビティ	0(reference)
T1	この時点で依然バス・アクティビティが無い場合、XcvrSelect、TermSelect を“1”にセットし、HS モードから FS モードに切り替える。	HS Reset T0 + 3.0ms < T1 {T_{WTREV}} < HS Reset T0 + 3.125ms
T2	LineState[1:0] をサンプリングする。このとき“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 + 100μs < T2 {$T_{WTWRSTHS}$} < T1 + 875μs
T3	これより前では、RESUME を発行してはいけない。	HS Reset T0 + 5ms {T_{WTRSM}}
T4	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）	HS Reset T0 + 10ms {T_{2SUSP}}
T5	内部クロックが完全停止	T5 < T4 + 10μs

注：　{ }は、USB2.0 規格書で規格されている名称。

A2.6.2.7.11.2　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ときに、3ms 以上送受信が何も検出されなかった場合、または USB_Status.LineState[1:0]ビットに“J”を検出し続け（T1）、さらに T2 の時点で依然“J”が検出された場合、USB のサスペンドステートであると判断し、SIE_IntStat.DetectSUSPEND ビットがセットされます。

このとき、SIE_IntEnb.EnDetectSUSPEND ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み信号がアサートされます。以下の図では、スヌーズを行ったときの動作を表しています。

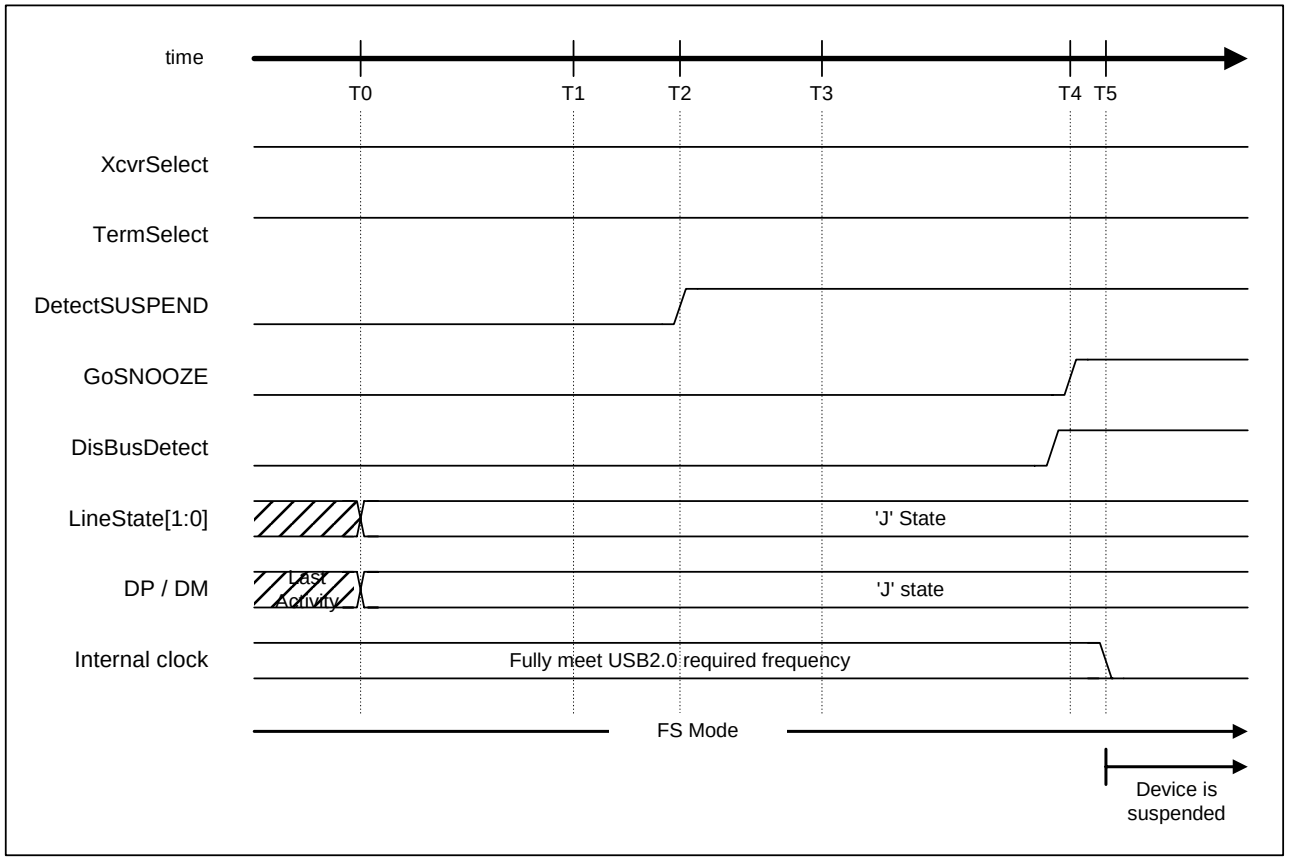


Fig. A2.6.9　Suspend Timing (FS mode)

Tbl. A2.6.6　Suspend Timing Values (FS mode)

Timing Parameter	Description	Values
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 + 3.0ms < T1 {T_{WTREV}} < T0 + 3.125ms
T2	LineState[1:0]をサンプリングする。このとき“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 + 100µs < T2 {$T_{WTWRSTHS}$} < T1 + 875µs
T3	これより前では、RESUME を発行してはいけない。	T0 + 5ms {T_{WTRSM}}
T4	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 + 10ms {T_{2SUSP}}
T5	内部クロックが完全停止。	T5 < T4 + 10µs

注：{ }は、USB2.0 規格書で規格されている名称。

A2.6.2.7.11.3　リセット検出（HS モード）

本 LSI が HS モードで動作しているときに、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 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、SIE_IntStat.DetectRESET ビットがセットされます。

このとき、SIE_IntEnb.EnDetectRESET ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み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、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。

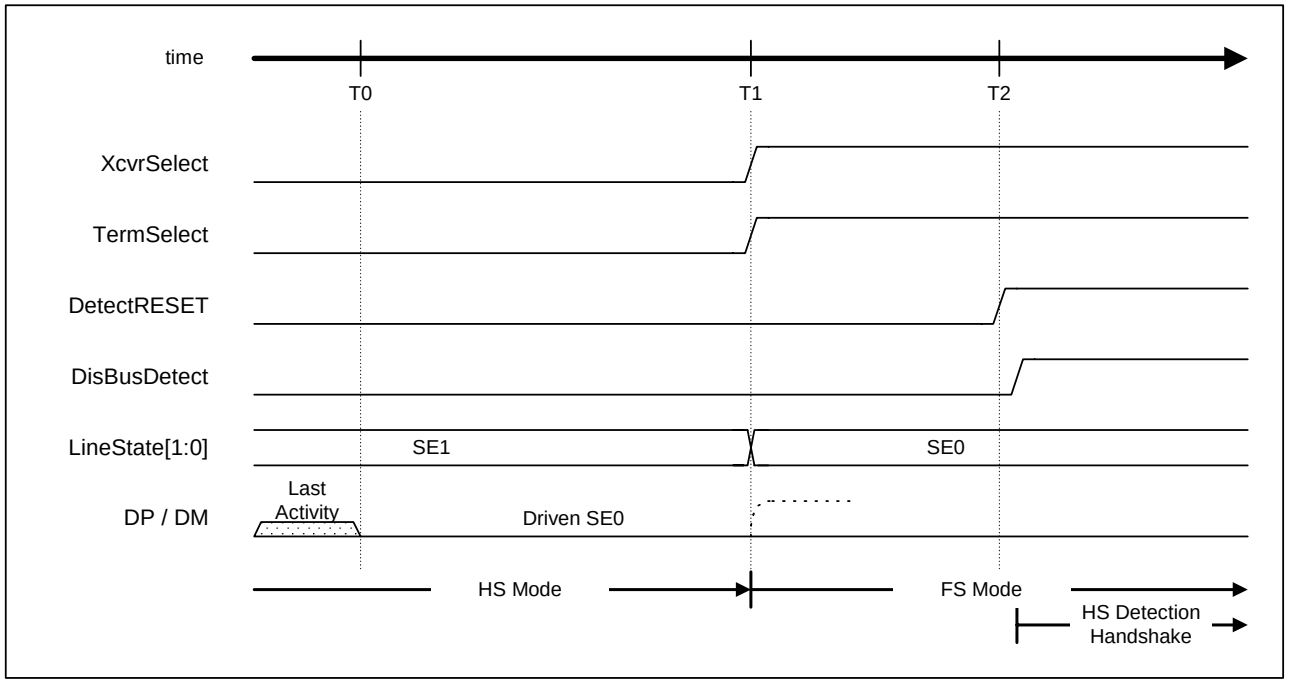


Fig. A2.6.10　Reset Timing (HS mode)

Tbl. A2.6.7　Reset Timing Values (HS mode)

Timing Parameter	Description	Values
T0	最後のバス・アクティビティ	0(reference)
T1	この時点で依然バス・アクティビティが無い場合、XcvrSelect、TermSelect を“1”にセットし、HS モードから FS モードに切り替える。	HS Reset T0 + 3.0ms < T1 {T_{WTREV}} < HS Reset T0 + 3.125ms
T2	LineState[1:0]をサンプリングする。このとき“SE0”なら、DetectRESET が“1”になり、リセットの移行と判断する。 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 + 100μs < T2 {$T_{WTWRSTHS}$} < T1 + 875μs

注：{ }は、USB2.0 規格書で規格されている名称。

リセット検出（FS モード）

本 LSI が FS モードで動作しているときに、2.5µs 以上 USB_Status.LineState[1:0]ビットに“SE0”を検出し続けた場合には（T1）、SIE_IntStat.DetectRESET ビットがセットされます。
このとき、SIE_IntEnb.EnDetectRESET ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み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 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。

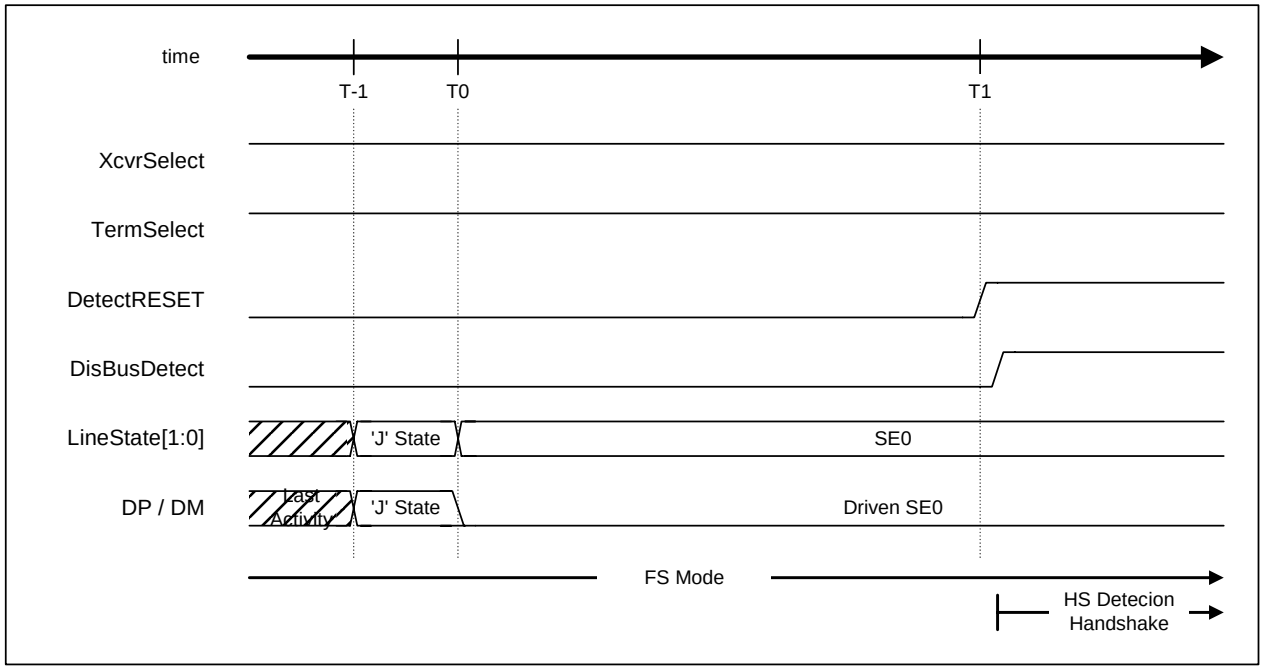


Fig. A2.6.11　Reset Timing (FS mode)

Tbl. A2.6.8　Reset Timing Values (FS mode)

Timing Parameter	Description	Values
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 + 2.5µs < T1 {T_{WTREV}}

注：{ }は、USB2.0 規格書で規格されている名称

A2.6.2.7.11.4　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とき）。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、2.5μs 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 のサスペンドステートなのかリセットなのかを判断しなければならない為、一旦 FS モードに切り替わります。このとき動作としては、XcvrControl.XcvrSelect、XcvrControl.TermSelect の両ビットを FS モードに切り替え、HS ターミネーションを無効にし、FS ターミネーションを有効にします。これらのモード切り替えは、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 100μs 以上 875μs 以内に USB_Status.LineState[1:0]ビットをチェックし、“J”なら USB のサスペンドステートとして判断し、“SE0”ならリセットと判断します。このとき、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（0）を参照してください。

A2.6.2.7.11.5　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 に接続されたときの動作を示します。HS Detection Handshake の開始時（T0）では、XcvrControl.XcvrSelect と XcvrControl.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、NegoControl.GoChirp ビットをセットします。すると 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、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）。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、XcvrControl.XcvrSelect ビットを FS モードに自動的に切り替え、NegoControl.GoChirp ビットがクリアされるとともに USB_Status.FSxHS ビットがセットされ、さらに SIE_IntStat.ChirpCmp ビットがセットされます。

このとき、SIE_IntEnb.EnChirpCmp ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み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。

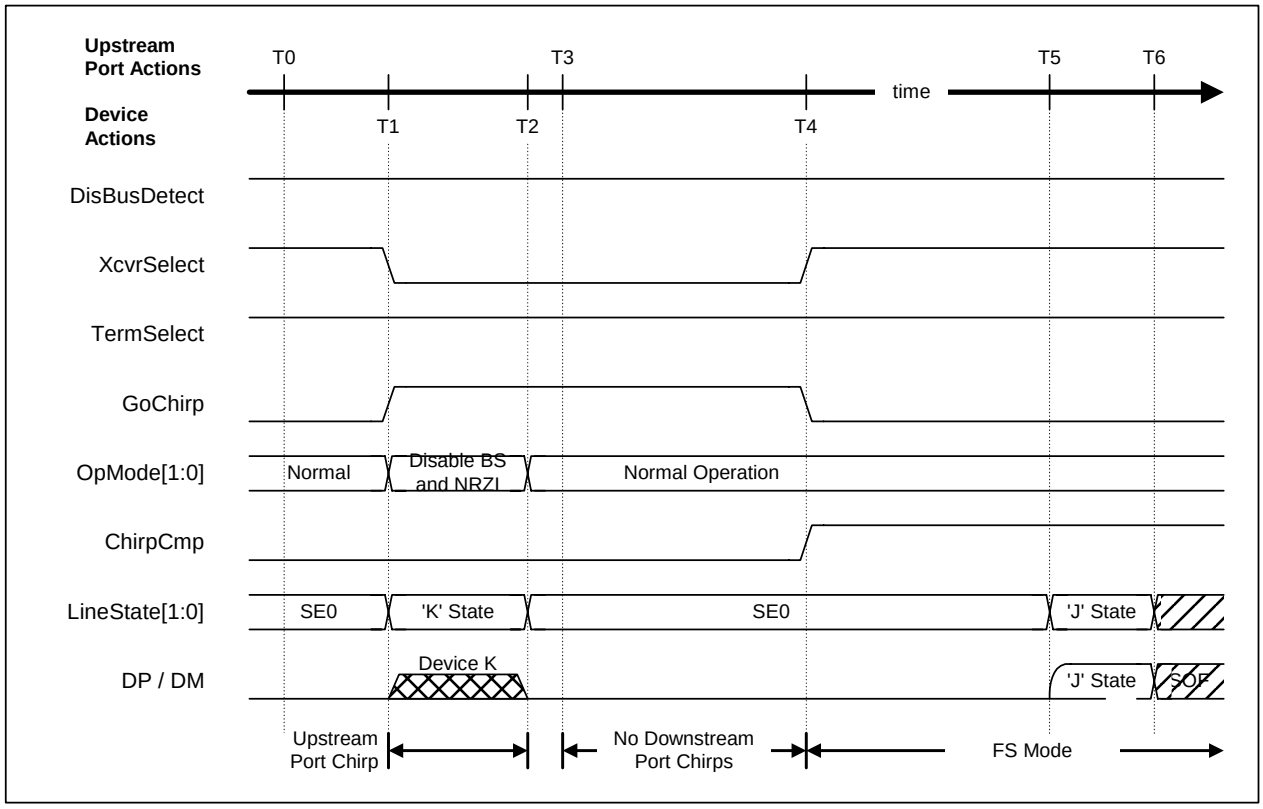


Fig. A2.6.12　HS Detection Handshake (FS mode)

Tbl. A2.6.9　HS Detection Handshake Timing Values (FS mode)

Timing Parameter	Description	Values
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 < HS Reset T0 + 6.0ms
T2	Chirp K 送出完了。最小 1ms の間は送出しなければならない。	T1 + 1.0ms {T_{UCH}} < T2 HS Reset T0+7.0ms {T_{UCHEND}}
T3	downstread 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 + 100µs {T_{WTDCH}}
T4	Chirp を検出できない場合、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 + 1.0ms < T4 {T_{WTFS}} < T2 + 2.5ms
T5	リセットシーケンスの終了	HS Rest T0 + 10ms {T_{DRST}(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。
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。

A2.6.2.7.11.6　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 に接続されたときの動作を示します。HS Detection Handshake の開始時（T0）では、XcvrControl.XcvrSelect と 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 NegoControl.GoChirp ビットをセットします。すると 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、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、NegoControl.GoChirp ビットがクリアされるとともに NegoStatus.FSxHS ビットがクリアされ、さらに SIE_IntStat.ChirpCmp ビットがセットされます。

このとき、SIE_IntEnb.EnChirpCmp ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み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、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 USB_Status.LineState[1:0]ビットにバスの信号を載せると非常にノイジーな為、XcvrControl.TermSelect ビットが HS モードのとき、バス・アクティビティがあると判断される場合には、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、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、XcvrControl.TermSelect が FS モード時の Chirp は約 800mV、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。

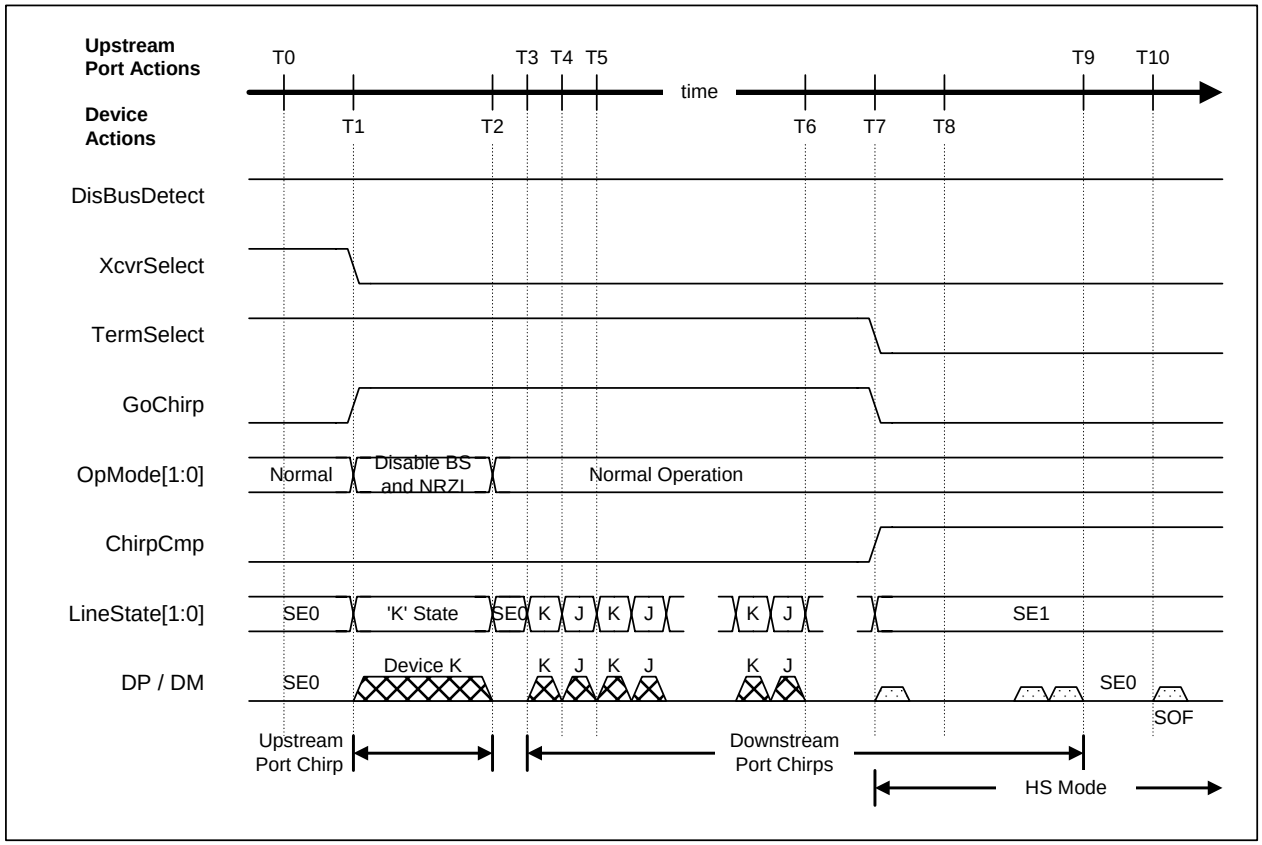


Fig. A2.6.13　HS Detection Handshake Timing (HS mode)

Tbl. A2.6.10　HS Detection Handshake Timing Values (HS mode)

Timing Parameter	Description	Values
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 < HS Reset T0 + 6.0ms
T2	Chirp K 送出完了。最小 1ms の間は送出しなければならない。	T1 + 1.0ms {T_{UCH}} < T2 HS Reset T0 + 7.0ms {T_{UCHEND}}
T3	downstream port が最初の Chirp K をバスに送出。	T2 < T3 < T2 + 100µs {T_{WTDCH}}
T4	downstream port が Chirp K から Chirp J に切り替え送出。	T3 + 40µs {T_{DCHBIT}(MIN)} < T4 < T3 + 60µs {T_{DCHBIT}(MAX)}
T5	downstream port が Chirp J から Chirp K に切り替え送出。	T4 + 40µs {T_{DCHBIT}(MIN)} < T5 < T4 + 60µs {T_{DCHBIT}(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、リセットの終了を待つ。	T6 < T7 < t6 + 500µs
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 が検出できないため、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 – 500µs {T_{DCHSE0}(MAX)} < T9 < T10 – 100µs {T_{DCHSE0}(MIN)}
T10	リセットシーケンスの終了	HS Rest T0 + 10ms {T_{DRST}(MIN)}

注：　{ }は、USB2.0 規格書で規格されている名称。
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。

A2.6.2.7.11.7　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）、そのとき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）、SIE_IntStat.NonJ ビットがセットされます。さらに、SIE_IntEnb.EnNonJ ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み信号がアサートされます。このとき、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。

このとき、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。

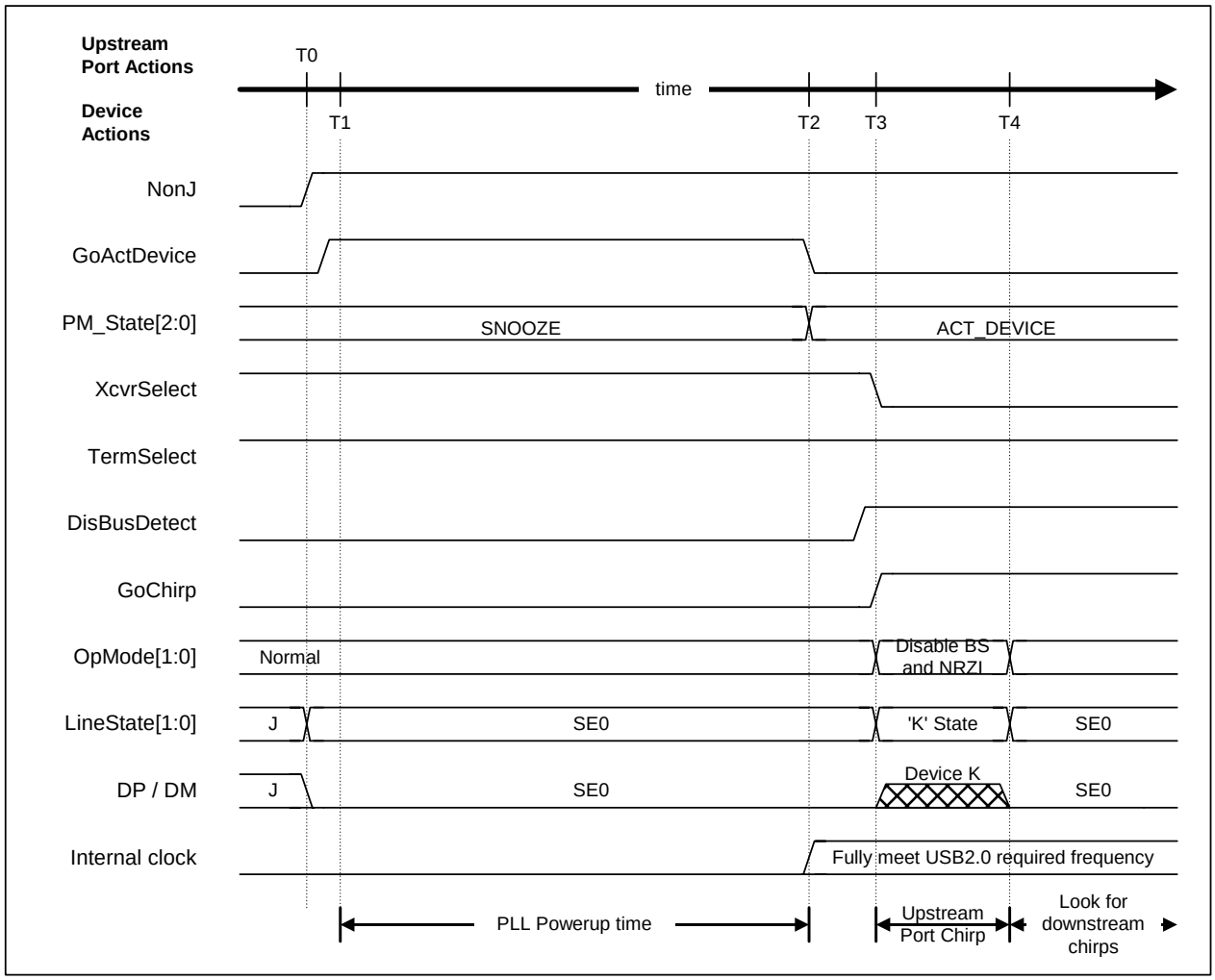


Fig. A2.6.14　HS Detection Handshake Timing from Suspend

Tbl. A2.6.11　HS Detection Handshake Timing Values from Suspend

Timing Parameter	Description	Values
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 + 250μs < 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 < HS Reset T0 + 5.8ms
T4	Chirp K 送出を終了。	T3 + 1.0ms {T_{UCH}} < T4 < HS Reset T0 + 7.0ms {T_{UCHEND}}

注： { }は、USB2.0 規格書で規格されている名称。
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）については後述する（PLL パワーアップ時間の他に、OSC パワーアップ時間が必要）。

A2.6.2.7.11.8　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ときに、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。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。このとき､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、NegoControl.SendWakeup ビットをセットし、レジューム信号を送出します（T2)。このとき､内部では､XcvrControl.OpMode[1:0]を“Disable Bit Stuffing and NRZI encoding”に設定し、送信データとして“0”を準備し､パケット送信状態にして､“K”（Resume 信号）を送出します。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、NegoControl.SendWakeup ビットをクリアすることによってバスに送出していたレジューム信号が停止されますが（T4)、この時点では 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､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、XcvrControl.XcvrSelect、XcvrControl.TermSelect の両ビットが所望のモード（今回の場合 HS モード）に切り替えられ、NegoControl.RestoreUSB ビットがクリアされるとともに SIE_IntStat.RestoreCmp ビットがセットされます｡このとき､SIE_IntEnb.EnRestoreCmp ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み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 USB_Status.FSxHS ビットが示すモードに戻ります。このとき、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 モードのときには、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)、そのときの動作を説明しています。

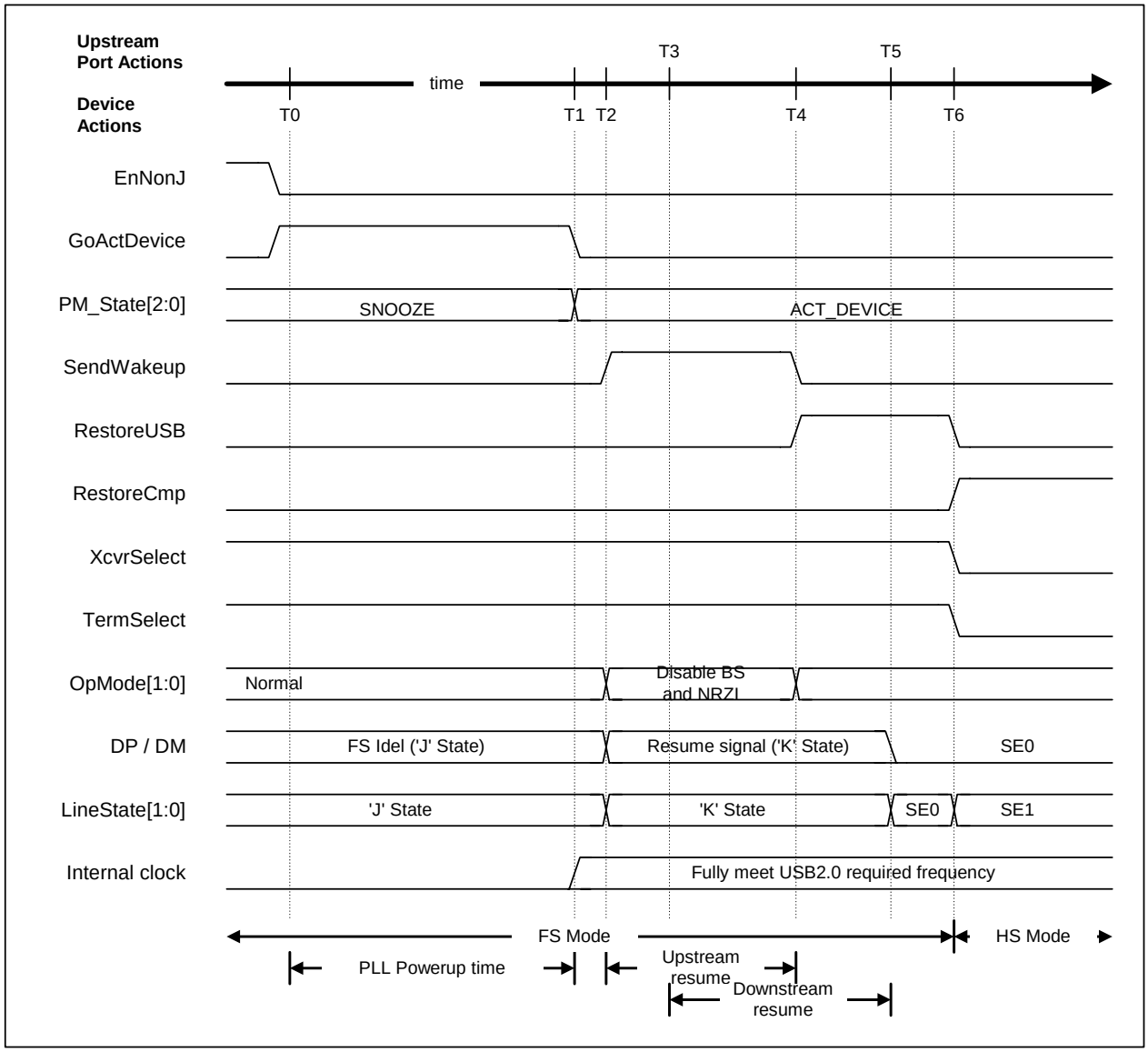


Fig. A2.6.15　Assert Resume Timing (HS mode)

Tbl. A2.6.12　Assert Resume Timing Values (HS mode)

Timing Parameter	Description	Values
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 + 250μs < 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”送出を終了。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”送出を終了。	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 + 1.33μs {2 Low-speed bit time}

注：{ }は、USB2.0 規格書で規格されている名称。

A2.6.2.7.11.9　レジュームの検出

本 LSI がスヌーズしているとき、バス上には“J”（USB_Status.LineState[1:0]は“J”）が観測されます。バス上に“K”が観測されたときは、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）。このとき、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。このとき、SIE_IntEnb.EnNonJ ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み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。このとき、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、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、XcvrControl.XcvrSelect、XcvrControl.TermSelect の両ビットが所望のモード（今回の場合 HS モード）に切り替えられ、NegoControl.RestoreUSB ビットがクリアされるとともに SIE_IntStat.RestoreCmp ビットがセットされます。このとき、SIE_IntEnb.EnRestoreCmp ビットおよび DeviceIntEnb.EnSIE_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、同時に割込み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）、そのときの動作を説明しています。

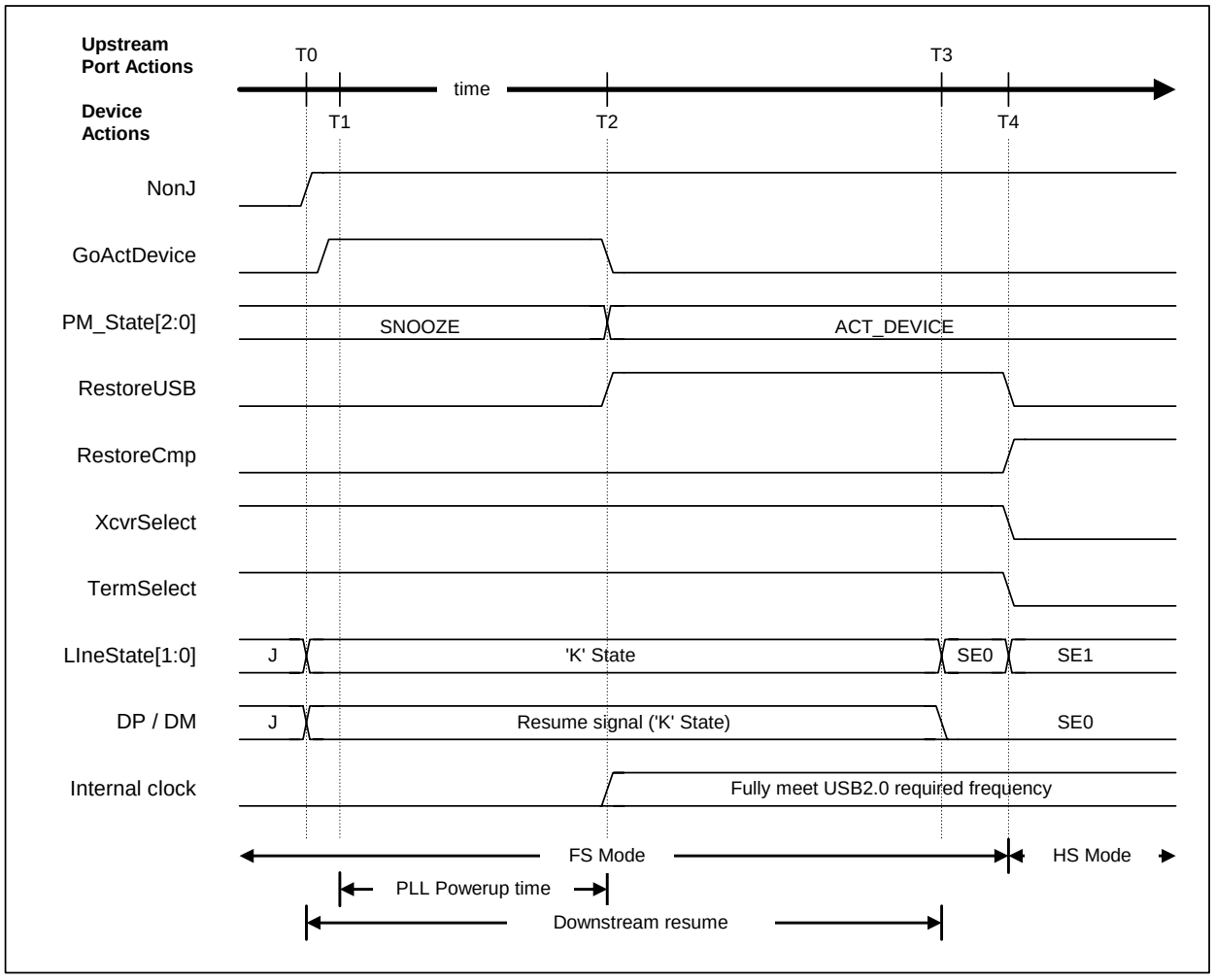


Fig. A2.6.16　Detect Resume Timing (HS mode)

Tbl. A2.6.13　Detect Resume Timing Values (HS mode)

Timing Parameter	Description	Values
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 + 250μs < 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 + 20ms {T_{DRSMDN}}
T4	USB のサスペンド以前が HS モードであった場合、自動的に HS モードに移行。	T3 + 1.33μs {2 Low-speed bit time}

注：{ }は、USB2.0 規格書で規格されている名称。

A2.6.2.7.11.10　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ときには、XcvrControl. XcvrSelect ビットは FS モード、XcvrControl.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 USB_Status.VBUS ビットがセットされます（T1）。このとき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、XcvrControl.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。

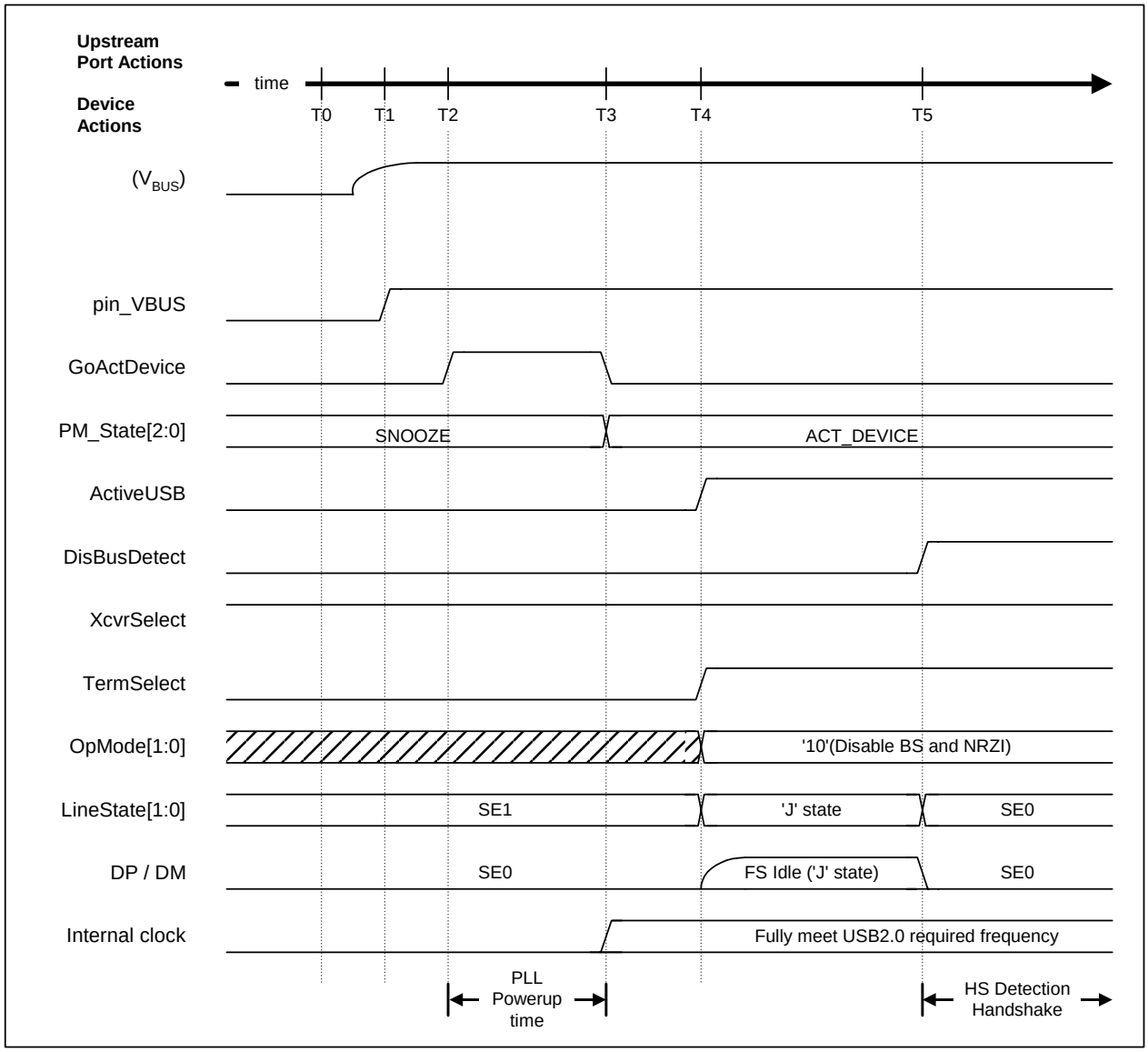


Fig. A2.6.17　Device Attach Timing

Tbl. A2.6.14　Device Attach Timing Values

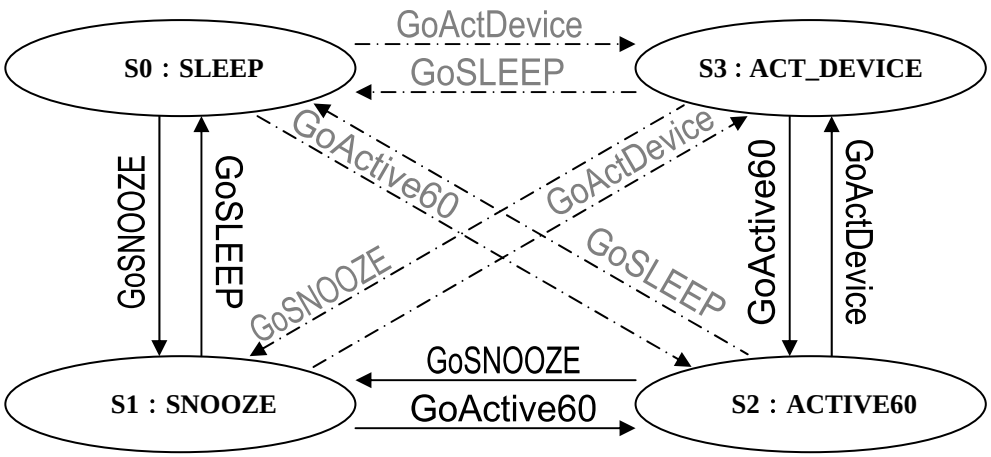
Timing Parameter	Description	Values
T0	ケーブルは挿入されていない。	0（reference）
T1	ケーブルが挿入され、入力ピン VBUS が“H”になる。	T1
T2	GoActDevice を“1”にセット。	T2
T3	PM_State が“ACT_DEVICE”になる。内部クロック出力の安定。	T2 + 250μs < T3
T4	ActiveUSB を“1”にセット。TermSelect を“1”にセット。 OpMode[1:0]を“00”に設定。 FS モードに移行。FS ターミネーションが有効。	T1 + 100ms {T_{SIGATT}} < 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_{ATTDB}} < T5

注：{ }は、USB2.0 規格書で規格されている名称。

A2.6.3　パワーマネージメント機能

オシレータ、PLL（DevicePLL480）の動作を制御し、スリープ、スヌーズ、アクティブ 60、アクトデバイスの４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 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。このとき、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。



※一点鎖線の遷移は、実際には実線を通って遷移する。

Fig. A2.6.18　パワーマネージメント

注）S2S65A00 においては、60MHz 専用の PLL を搭載しておりません。パワーマネージメントの機能として「ACTIVE60」というステートが存在しますが、内部の状態としては「ACTIVE60」ステートと「SNOOZE」ステートとは同じ状態にあります。

A2.6.3.1　SLEEP（スリープ）

オシレータが発振していないステートです。したがいまして、このステートでは、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.GoActve60 ビット、PM_Control. GoActDevice ビットをセットし、スリープからスヌーズへ離脱するときには、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。

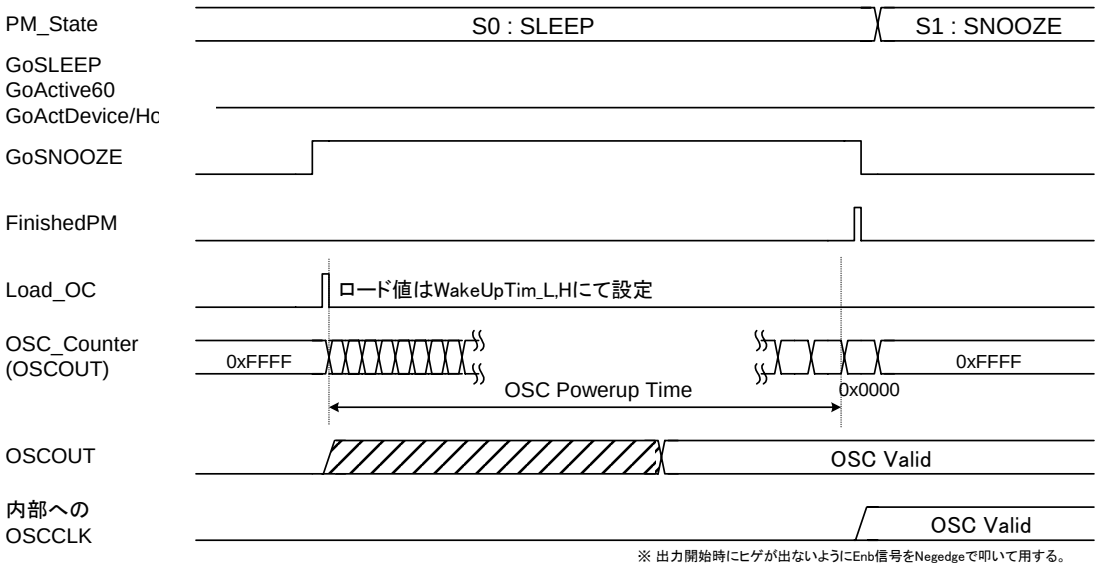


Fig. A2.6.19　SLEEP ステートからの離脱（GoSNOOZE 時）

A2.6.3.2　SNOOZE（スヌーズ）

オシレータは発振している状態で、PLL が発振していないステートです。
アクティブ 60・ステート、アクトデバイス・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 を停止します。
しかし、スヌーズ・ステート中に PM_Control_0.GoActDevice ビット及び PM_Control_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ときには、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 250μs）ゲートします。

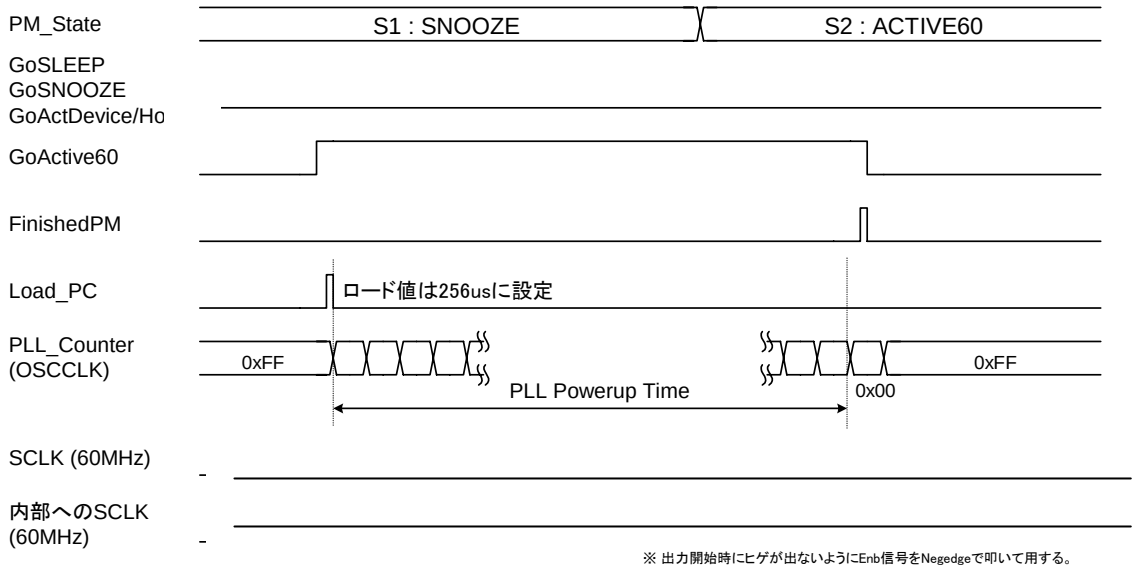


Fig. A2.6.20　SNOOZE ステートからの離脱（GoActive60 時）

A2.6.3.3　ACTIVE60（アクティブ 60）

S2S65A00 においては、60MHz 専用の PLL を搭載していないため、「SNOOZE」ステートと同じ状態にあります。ただし、スヌーズ・ステートからアクティブ 60・ステートに遷移するには、
に示すように約 256μs の時間がかかります。

A2.6.3.4　ACT_DEVICE（アクトデバイス）

オシレータ、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、本ステートにおいてのみ読み書きすることができます。
また、USB デバイス回路はアクトデバイス・ステートでのみ動作します。

A2.6.4　FIFO 管理

A2.6.4.1　FIFO メモリマップ

FIFO のメモリマップを以下に示します。

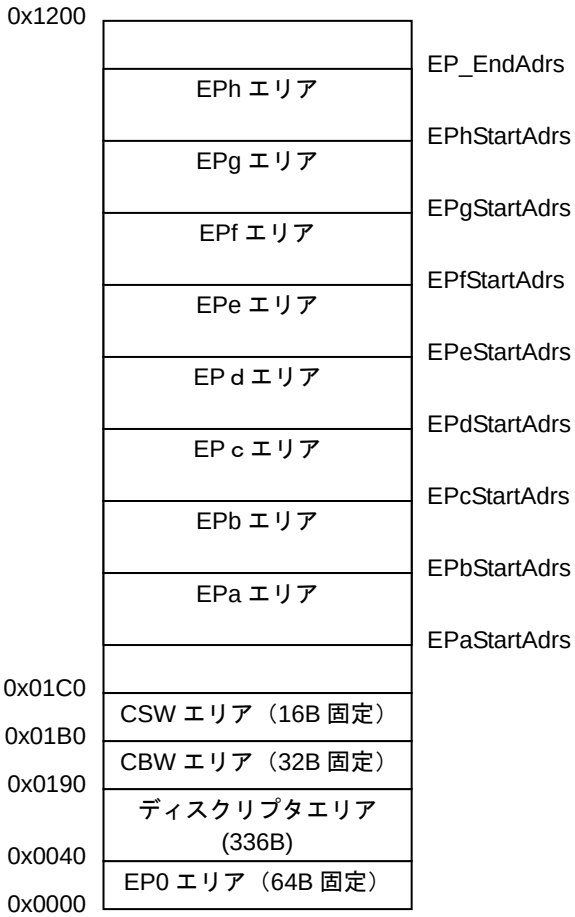


Fig. A2.6.21　FIFO メモリマップ例

FIFO のメモリは、EP0 エリア、デスクリプタエリア、CBW エリア、CSW エリア、EPa~h エリアの最大 12 エリアに分割して使用できます。EP0 エリア、デスクリプタエリア、CBW エリア、CSW エリアについては Fig. A2.6.21 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-h}エリアについては FIFO エリア設定レジスタ（EPx{x=a-h}StartAdrs_H,L, EP_EndAdrs_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。

デスクリプタエリアは、デ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、A2.6.4.2 に後述します。実際には全ての FIFO 領域をデスクリプタ返信機能に使用設定することが可能となっていますが､競合を避けるために、ここで表したエリアをデ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、A2.6.4.3 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、A2.6.4.4 に後述します。

EPa~h 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。
EP0 エリア、EPa~h 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、EPnControl.AllFIFO_Clr ビットか EPnControl.EP0FIFO_Clr ビット、または、各領域に対応した EPrFIFO_Clr.EPx{x=a-h}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。

A2.6.4.2　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できます。

A2.6.4.2.1　デスクリプタエリアへのデータの書込み

デスクリプタエリアへデータを書き込むには、RAM_WrDoor 機能を使用します。書き込み開始アドレスを RAM_WrAdrs_H,L レジスタに設定し、RAM_WrDoor_0,1 レジスタにデータを書き込むことによって行います。RAM_WrAdrs_H,L レジスタの値は、一回の書き込み毎に、書き込みデータ数づ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。

A2.6.4.2.2　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、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、EP0Control.ReplyDescriptor ビットに 1 をセットします。また、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、SETUP_Control.ProtectEP0 をクリアしてから、EP0Control_IN.ForceNAK ビットをクリアするようご注意下さい。
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（EP0MaxSize にて設定）に自動的に分割しながら、ホストに 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、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、EP0Control.ReplyDescriptor をクリアし、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。

A2.6.4.3　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、DMA 等の転送の制御を容易にします。

A2.6.4.3.1　CBW エリアへの受信

CBW サポートを実行しているときに、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。

A2.6.4.3.2　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_1F 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。

A2.6.4.4　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、DMA 等の転送の制御を容易にします。

A2.6.4.4.1　CSW エリアからの送信

CSW サポートを実行しているときに、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。

A2.6.4.4.2　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。

A2.6.4.5　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), DMA, USB が有ります。

A2.6.4.5.1　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、チャネルの FIFO 領域の設定に関係なく、任意に行うことが出来ます。

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。

A2.6.4.5.2　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、チャネルの FIFO 領域の設定に関係なく、任意に行うことが出来ます。

A2.6.4.5.3　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 つのチャネルに対して、D_EPx{x=0,a-h}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 つのチャネルに対して、D_EPx{x=0,a-h}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、EPx{x=0,a-h}Join.JoinCPU_Rd にてただ 1 つ設定されたチャネルにおいて、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-h}Join.JoinCPU_Wr にてただ 1 つ設定されたチャネルにおいて、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-h}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。

A2.6.4.5.3.1　FIFO アクセス（ライト）

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

FIFO アクセス（ライト）には以下の制限事項があります。

- EPx{x=0,a-h}Join.JoinCPU_Wr ビットを設定した後、FIFO_WrRemain_H,L レジスタで書き込み可能なデータ数を確認した上でアクセスを行って下さい。基本的にワード（2 バイト）単位でアクセスを行って下さい。端数（奇数）バイトの書き込みを行う場合は FIFO のバイト境界を意識してアクセスサイズを制御して下さい。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。

A2.6.4.5.3.2　FIFO アクセス（リード）

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

FIFO リードアクセスには、以下の制限事項があります。

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

A2.6.4.5.3.3　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ことを推奨致します。

EPnControl.EP0FIFO_Clr ビットや EPrFIFO_Clr.EPx{x=a-h}FIFO_Clr ビットをセットする等行って、バイト境界の存在しない状態からワード書き込みを行い、奇数データが存在する場合は、連続するデータの最終バイト（データ Z）のみ High 側に書き込んで下さい。この状態を Fig. A2.6.22 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 Fig. A2.6.22 の（2）に示します。

以上は、正常な書き込み動作です。

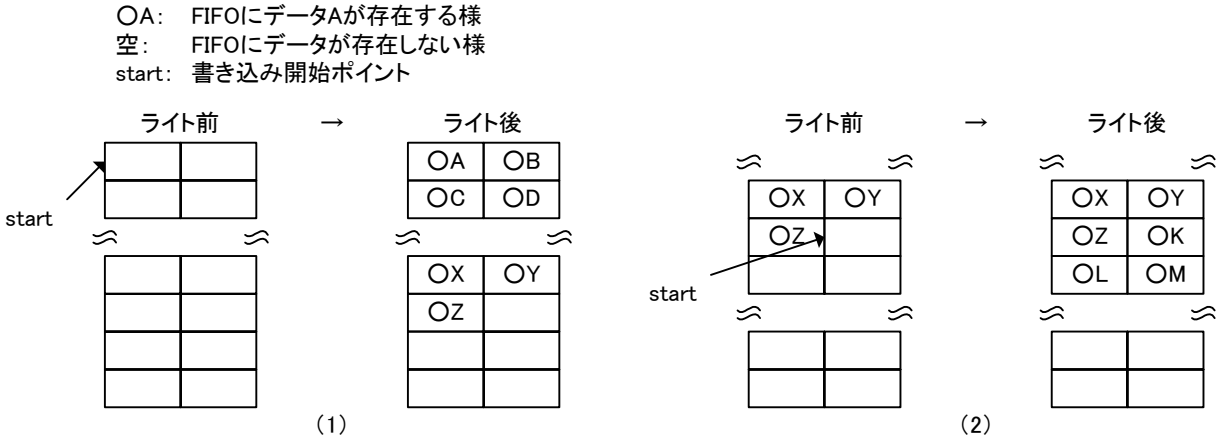


Fig. A2.6.22　FIFO ライト処理（正常動作）

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（Fig. A2.6.23 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（Fig. A2.6.23 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（Fig. A2.6.23 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(Fig. A2.6.23 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。

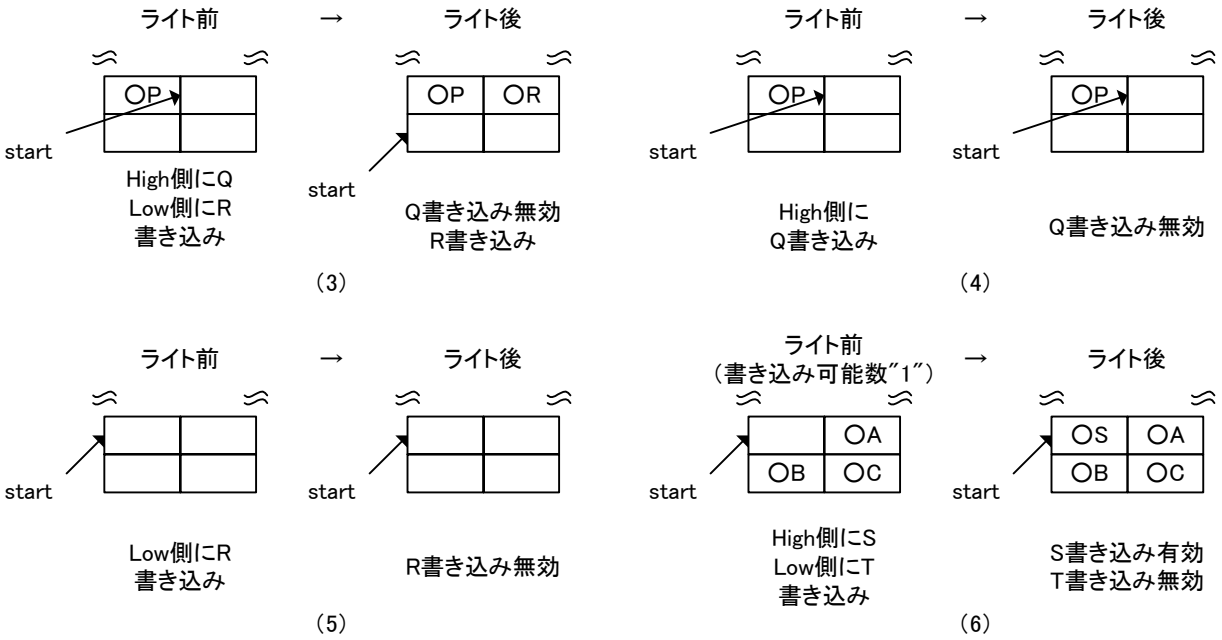


Fig. A2.6.23　FIFO ライト処理（注意が必要な動作）

【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 Fig. A2.6.24 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 Fig. A2.6.24 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。

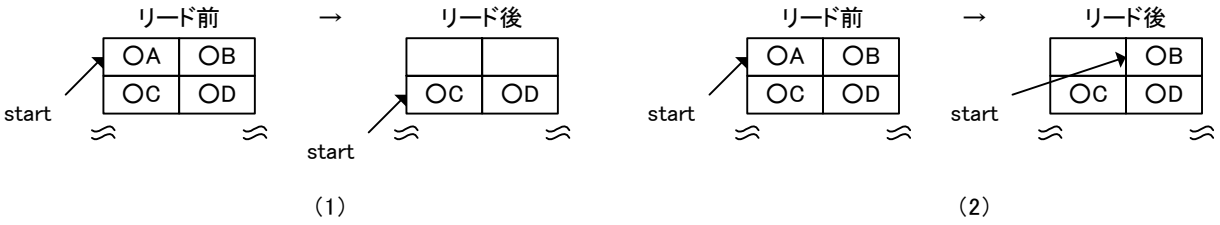


Fig. A2.6.24　FIFO リード処理（正常動作）

以下は注意が必要な読み出しを行った場合の動作です。

Fig. A2.6.25 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。Fig. A2.6.25 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。

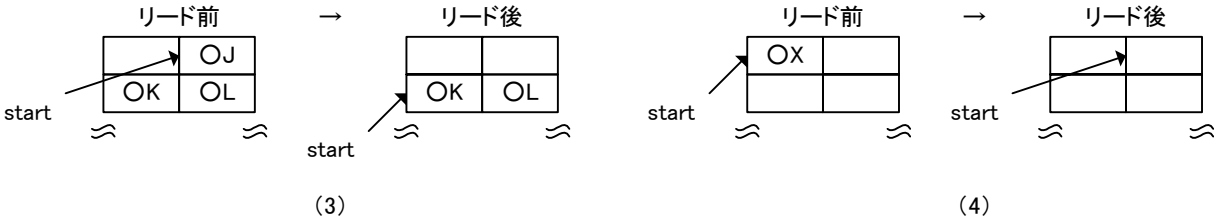


Fig. A2.6.25　FIFO リード処理（注意が必要な動作）

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 H/W が 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 H/W 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。

A2.6.4.5.4　FIFO へのアクセス方法（DMA）

FIFO に DMA アクセスによってリードアクセスする場合には、DMA の各チャネルにつき、EPx{x=0,a-h}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 DMA アクセスによってライトアクセスする場合には、DMA の各チャネルに付き、EPx{x=0,a-h} 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 の各チャネルについて、EPx{x=0,a-h}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 の各チャネルについて、EPx{x=0,a-h}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。

A2.6.4.5.5　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書込み、読出し、また DMA による書き込み、読み出しが同時に行われます。また、CPU バスからの読み出しについては、先読み処理を行っています。

これらのことから、それぞれのチャネルにおける 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。

- 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ない。

・ 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ない。

また、USB からのアクセスに関して、下記の禁止事項があります。
・ USB から書き込み中の FIFO エリアに対し、他の要因から書き込みを行ってはいけない。
・ USB への読み出し中の FIFO エリアに対し、他の要因から読み出しを行ってはいけない。

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出来ますが、必ず OUT トランザクションが行われない状況で、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出来ますが、必ず IN トランザクションが行われない状況で、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。

A2.6.5　DMA

A2.6.5.1　概要

本 USB コントローラには、AHB バス(1)に直接接続されている 8bit の DMA I/F を 2ch.搭載しております。DMA コントローラ 1 と連動させることにより、USB コントローラとメモリ間での DMA 転送を行うことができます。
DMA I/F のポートは、S2S65A00 では AHB バス(1)上の以下のアドレスにマッピングされています。

DMA0　　0xE000_0000（固定）
DMA1　　0xE000_0004（固定）
※USB コントローラの DMA ポートに対しては、アドレス固定モードで転送してください。

A2.6.5.2　基本機能

USB コントローラの DMA I/F には、2 つの動作モード（カウントモード、フリーランモード）があります。
DMA の転送方向に関しては、本仕様書では以下として説明いたします。
DMA Write（ライト）:　S2S65A00 内部/外部メモリ　⇒　USB コントローラ
DMA Read（リード）:　USB コントローラ　⇒　S2S65A00 内部/外部メモリ

A2.6.5.2.1　カウントモード

DMA0/1_Count[31:0]に設定されたカウント数（バイト数）分の DMA 転送を行います。設定されたカウント数分の DMA 転送が終了すると、DMA_IntStat.DMA0/1_Cmp の割込み要因が発生し、DMA 回路は停止します。
DMA0/1_Count レジスタにカウント数をセットした後、DMA0/1_Control.Go ビットに“1”をセットすると、USB コントローラ内部の FIFO の状態を確認し、DMA コントローラに対してデータ転送要求を発行します。データ転送を行う度に DMA0/1_Count レジスタの内容をカウントダウン（デクリメント）していき、カウント数が“0”になったところで DMA 転送を停止し、DMA_IntStat.DMA0/1_Cmp 割り込み要因を発生します。
DMA0/1_Control.Stop ビットに“1”をセットすることで DMA 転送を停止（中断）させることができます。DMA0/1_Control.Stop ビットに“1”をセットした直後でも、DMA 転送が完全に停止するまでは DMA0/1_Control.Running ビットに“1”を示します。DMA 転送が完全に停止（中断）すると、Running ビットが“0”になり、DMA_IntStat.DMA0/1_Cmp 割込み要因が発生しますので、これらのビットを確認してから次の処理を行うようにしてください。

A2.6.5.2.2　フリーランモード

このモードで DMA 転送を実行すると、USB コントローラ内部の FIFO 状態のみを確認し DMA 転送を永久的に継続します。外部の DMA コントローラ側でデータ転送数を管理するような場合、本モードを用いることで設定手順を省略することができます。

DMA0/1_Config.FreeRun ビットを“1”にセットし、DMA0/1_Control.Go ビットに“1”をセットすると、USB コントローラ内部の FIFO の状態を確認し、DMA コントローラに対してデータ転送要求を発行します。データ転送を行うたびに DMA0/1_Count レジスタの内容をカウントアップ（インクリメント）していき、カウント数が“0xFFFF_FFFF”から“0x0000_0000”になったときに DMA_IntStat.DMA0/1_CountUp 割込み要因を発生します。しかし DMA0/1_CountUp 割込み要因が発生した後も、DMA 転送は継続されます。

DMA0/1_Control.Stop ビットに“1”をセットすることで DMA 転送を停止（中断）させることができます。DMA0/1_Control.Stop ビットに“1”をセットした直後でも、DMA 転送が完全に停止するまでは DMA0/1_Control.Running ビットに“1”を示します。DMA 転送が完全に停止（中断）すると、Running ビットが“0”になり、DMA_IntStat.DMA0/1_Cmp 割込み要因が発生しますので、これらのビットを確認してから次の処理を行うようにしてください。

A2.6.5.3　動作手順

DMA コントローラ 1 を利用した DMA 転送手順の一例を示します。

条件：　SDRAM（0x4000_0000）　→　USB 間の 64byte 転送（DMA Write）
DMAC1　CH.0 使用（カウントモード）
USB DMA Port　CH.0 使用
EPa 使用（FIFO 領域、USB 通信設定等は済んでいるものとする）
DMA_Cmp ビットのポーリングにより、DMA 転送終了の判定

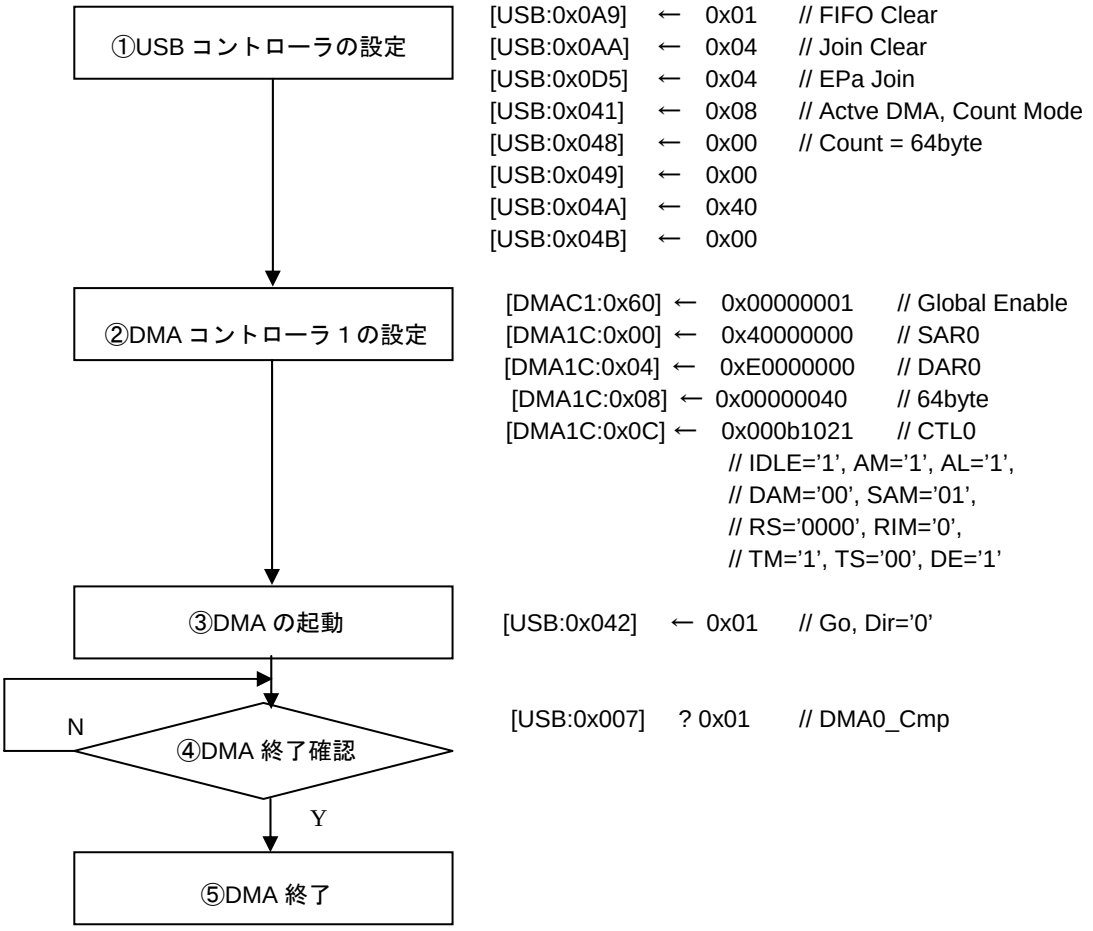


Fig. A2.6.26　DMA 転送手順例

A2.6.5.4　強制終了

DMA 転送を強制的に終了したい場合は、DMA0/1_Config.ActiveDMA ビットを“0”にクリアすることで行うことができます。ただしこの場合には、転送途中の DMA データの連続性が保証されませんので、通常の停止（中断）には DMA0/1_Control.Stop ビットにより行ってください。強制終了した後に、再度 DMA 転送を行う場合には、USB FIFO の初期化、Join の設定および DMA の設定を再度行ってください。

DMA 転送最中に USB FIFO が初期化されたり、Join の設定が変更された場合には、同様に DMA 転送が一旦強制終了され、DMA0/1_Config.ActiveDMA が自動的に“0”にクリアされます。

セイコーエプソン株式会社
マイクロデバイス事業本部　デバイス営業部

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：411076004
2007 年 4 月 作成
2011 年 12 月 改訂