
USBファンクションコントローラ (USBFC)
EIFUFAL501

ユーザーズ マニュアル

Doc #: 88-02-J01
Revision: 2.0
Date: 03/24/98

USB ファンクションコントローラ (USBFC)

1. 特長	4
1.1 特長	4
1.2 概要	4
1.3 USBFC ブロック図	5
1.4 USBFC システムブロック例	5
2. 端子説明	6
2.1 端子図	6
2.2 信号説明	7
3. 機能説明	9
3.1 USB インタフェース	9
3.2 ローカルバス	9
3.2.1 CPU制御によるUSBからローカルバスへの転送	9
3.2.2 CPU制御によるローカルバスからUSBへの転送	10
3.2.3 DMA制御によるUSBからローカルバスへの転送	10
3.2.4 DMA制御によるローカルバスからUSBへの転送	11
3.2.5 DMA 転送の終了	11
3.2.6 USB エンドポイント 1 : 受信メールボックス	12
3.3 サスPENDモード	12
3.3.1 サスPENDシーケンス	12
3.3.2 デバイスリモートウェークアップ	13
3.3.3 ホスト側からのウェークアップ	13
3.4 USBFC 電源について	13
4. ローカルレジスタ	14
4.1 レジスタ説明	14
4.2 レジスター一覧	15
4.3 (アドレス 00H; DCTL) DMA コントロールレジスタ	16
4.4 (アドレス 01H; IRQENB1) 割り込みイネーブルレジスタ 1	17
4.5 (アドレス 02H; IRQSTAT1) 割り込みステータスレジスタ 1	18
4.6 (アドレス 03H; IRQENB2) 割り込み許可レジスタ 2	18
4.7 (アドレス 04H; IRQSTAT2) 割り込みステータスレジスタ2	19
4.8 (アドレス 08H; EP1IDX) エンドポイント 1 インデックスレジスタ	19
4.9 (アドレス 09H; EP1DATA) エンドポイント 1 受信メールボックスデータ	19
4.10 (アドレス 0CH; EP2IDX) エンドポイント 2 インデックスレジスタ	20
4.11 (アドレス 0DH; EP2DATA) エンドポイント 2 送信メールボックスデータ	20
4.12 (アドレス 0EH; EP2POLL) エンドポイント 2 インタラプトポーリングインターバルレジスタ	20
4.13 (アドレス 10H; EP3DATA) エンドポイント 3 受信 FIFO データレジスタ	20
4.14 (アドレス 11H; EP3COUNT) エンドポイント 3 受信 FIFO カウントレジスタ	20
4.15 (アドレス 12H; EP3STAT) エンドポイント 3 受信 FIFO ステータスレジスタ	21
4.16 (アドレス 13H; EP3PKSZ) エンドポイント 3 最大パケットサイズレジスタ	21
4.17 (アドレス 14H; EP4DATA) エンドポイント 4 送信 FIFO データレジスタ	21
4.18 (アドレス 15H; EP4COUNT) エンドポイント 4 送信 FIFO カウントレジスタ	21
4.19 (アドレス 16H; EP4STAT) エンドポイント 4 送信 FIFO ステータスレジスタ	22
4.20 (アドレス 17H; EP4PKSZ) エンドポイント 4 最大パケットサイズレジスタ	22
4.21 (アドレス 18H; REVISION) レビジョンレジスタ	22
4.22 (アドレス 19H; USBSTAT) USB ステータスレジスタ	23

4.23 (アドレス 1AH; FRAMEMSB) フレームカウンタ MSB レジスタ	23
4.24 (アドレス 1BH; FRAMELSB) フレームカウンタ LSB レジスタ	23
4.25 (アドレス 1CH; EXTIDX) 拡張レジスタインデックス	24
4.26 (アドレス 1DH; EXTDATA) 拡張レジスタデータ	24
4.26.1 (アドレス 1Dh, インデックス 00h; VIDMSB) ベンダー ID MSB	24
4.26.2 (アドレス 1Dh, インデックス 01h; VIDLSB) ベンダー ID LSB	24
4.26.3 (アドレス 1Dh, インデックス 02h; PIDMSB) プロダクト ID MSB	24
4.26.4 (アドレス 1Dh, インデックス 03h; PIDLSB) プロダクト ID LSB	24
4.26.5 (アドレス 1Dh, インデックス 04h; RELMSB) リリース番号 MSB	24
4.26.6 (アドレス 1Dh, インデックス 05h; RELLSB) リリース番号 LSB	25
4.26.7 (アドレス 1Dh, インデックス 06h; RCVAFTH) 受信FIFOオールモストフル スレショルド	25
4.26.8 (アドレス 1Dh, インデックス 07h; XMTAETH) 送信FIFOオールモストエンプティ スレショルド	25
4.26.9 (アドレス 1Dh, インデックス 08h; USBCTL) USB コントロール	25
4.26.10 (アドレス 1Dh, インデックス 09h; MAXPWR) 最大電力消費	25
4.26.11 (アドレス 1Dh, インデックス 0Ah; PKTCTL) パケット制御	26
4.26.12 (アドレス 1Dh, インデックス 0Bh; LOCALCTL) ローカル側制御	26
4.26.13 (アドレス 1Dh, インデックス 0Ch; FIFOCTL) FIFO 制御	26
5. 標準デバイスリクエスト	27
5.1 コントロール`IN`トランザクション	27
5.1.1 Get Device Status	27
5.1.2 Get Interface Status	27
5.1.3 Get Endpoint 0, 1, 2, 3, 4	27
5.1.4 Get Device Descriptor (18 Bytes)	27
5.1.5 Get Configuration Descriptor (46 Bytes)	28
5.1.6 Get String Descriptor 0	30
5.1.7 Get String Descriptor 1	30
5.1.8 Get String Descriptor 2	30
5.1.9 Get Configuration	30
5.1.10 Get Interface	30
5.2 コントロール`OUT`トランザクション	31
5.2.1 Set Address	31
5.2.2 Set Configuration	31
5.2.3 Set Interface	31
5.2.4 Device Clear Feature	31
5.2.5 Device Set Feature	31
5.2.6 Endpoint 0, 1, 2, 3, 4 Clear Feature	31
5.2.7 Endpoint 0, 1, 2, 3, 4 Set Feature	32
5.3 エンドポイント1`OUT`トランザクション (受信メールボックス)	32
5.4 エンドポイント2`IN`トランザクション (送信メールボックス)	32
5.5 エンドポイント3`OUT`トランザクション (受信FIFO)	32
5.6 エンドポイント4`IN`トランザクション (送信FIFO)	32
7. タイミング	33
7.1 ローカルバスからレジスタへの書き込み	33
7.2 レジスタからローカルバスへの読み込み	34
7.3 FIFOへのDMA書き込み	35
7.4 FIFOからのDMA読み出し	36
8. テスト回路	37

1. 特長

1.1 特長

- USB 仕様バージョン 1.0 準拠
- 個別ローカルバスプロセッサとUSB バスとの連結
- USB デバイス帯域幅—最高12Mb/sec
- USB バルク (Bulk) 転送、アイソクロナス (Isochronous) 転送、インターラプト転送とコントローラ転送
- 送信・受信用それぞれ64 バイトのFIFOを用いてスループットを向上
- ローカルCPUまたはDMA によるデータ転送の保守機能
- 3.3 V電源動作

1.2 概要

このUSB ファンクションコントローラ (USBFC) により、一般的なローカルバスと USB 間のバルク転送または アイソクロナスデータ転送が可能になります。すなわち、USBFC は、ホストコンピュータとデジタルカメラまたはスキャナーなどのインテリジェント周辺機器との接続を可能にするものです。

USBFC の構成としては大きく分けて 3 つあり、それは USB バスインタフェース、2つの64 バイト FIFO そしてローカルバスインタフェースです。

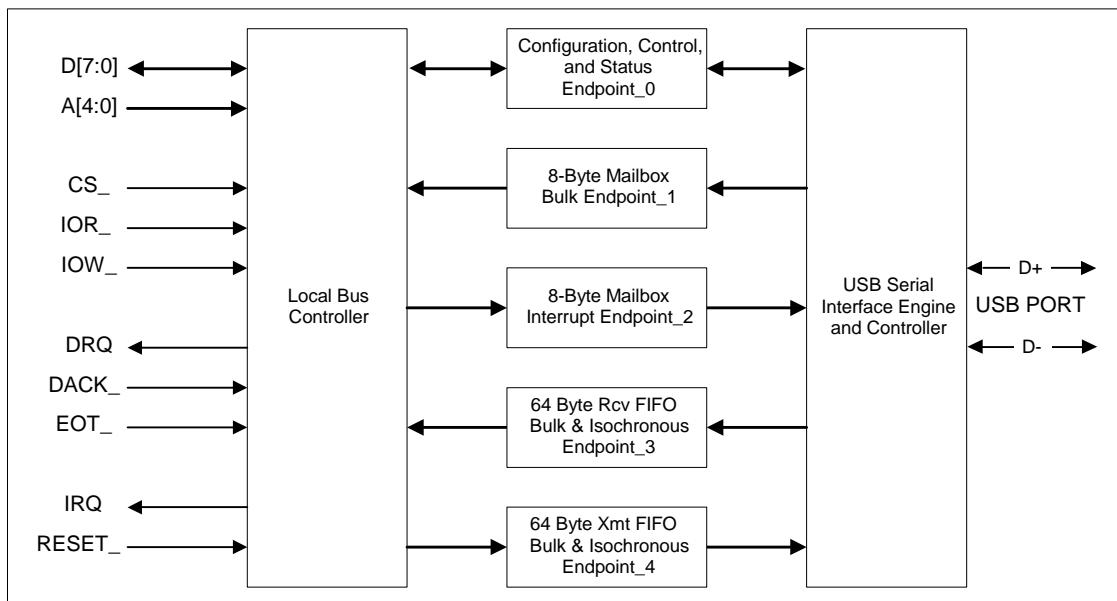
USBインタフェースの機能としては次の通りです。

- ホストとデバイス間の通信
- バルク (Bulk) またはアイソクロナス (Isochronous) のエンドポイントを通してFIFOへアクセス
- インタラプトエンドポイントを通してローカルからUSB メールボックスへアクセス
- バルクエンドポイントを通してUSB からローカルメールボックスへアクセス

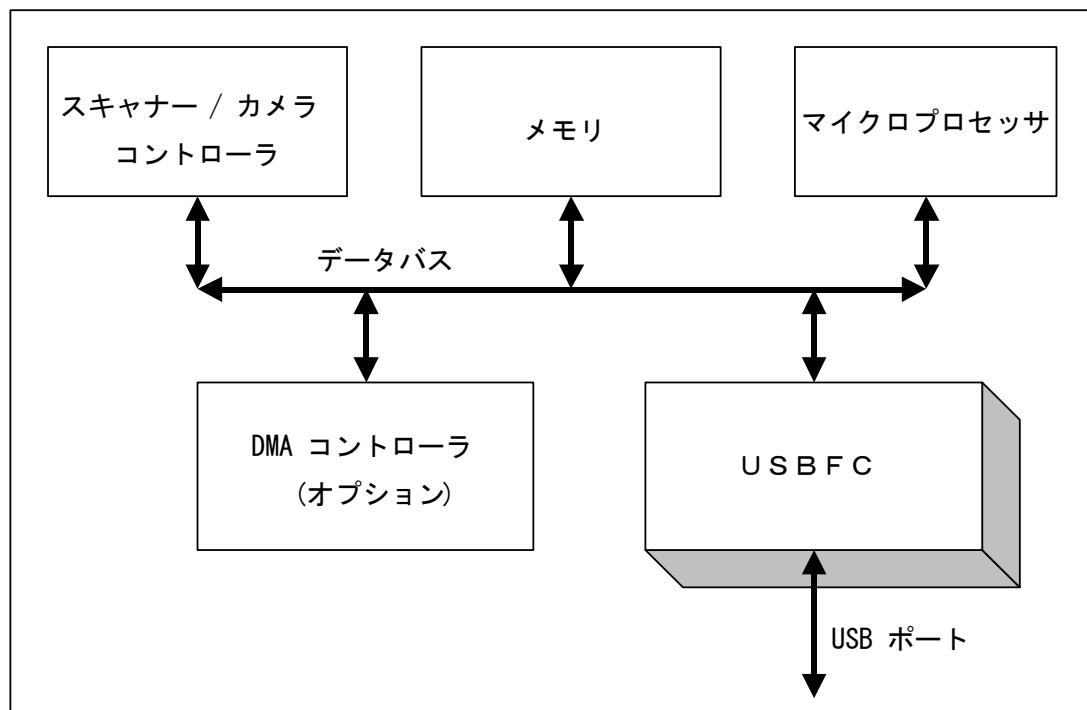
ローカルバスインタフェース機能としては次の通りです。

- FIFO コントロール
- ローカル CPU インタフェース
- ローカル DMA コントローラインタフェース
- 割り込み

1.3 USBFC ブロック図

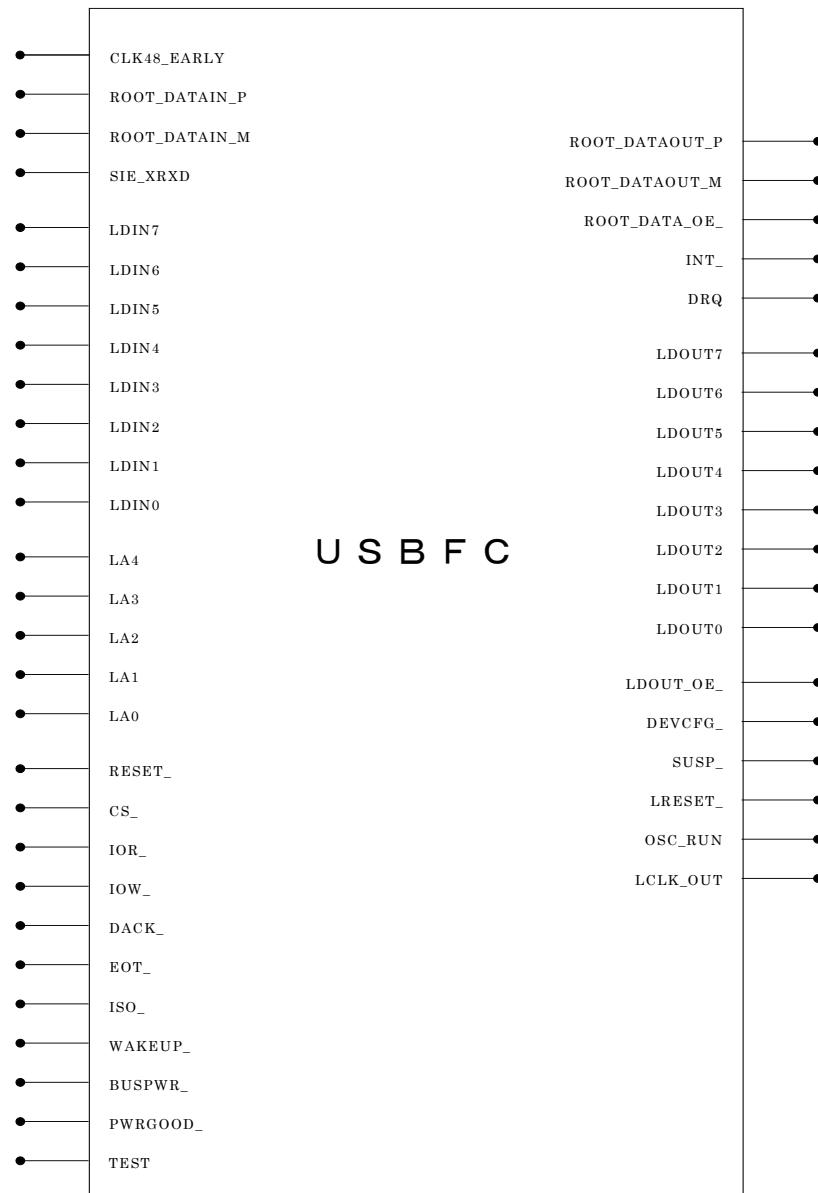


1.4 USBFC システムブロック例



2. 端子説明

2.1 端子図



2.2 信号説明

NOTE: USBFCがサスPEND状態にあるときは、入力信号はHまたはLにドライブするようにしてください。 信号名の最後にアンダースコア(_)のあるものは負論理です。

Signal Name	Type	Description
CLK48_EARLY	Input	48 MHz オシレータ入力
RESET_	Input	リセット 外部リセット。ローカルリセットまたはパワーオンリセットに接続します。オシレータ停止時（初期電源投入またはサスPEND状態）にリセットするためには、1 ms以上アサートします。オシレータ発振時は48Mhzクロックの5周期間以上アサートしてください。
ROOT_DATAIN_P, ROOT_DATAIN_M	Input	USB データ入力 USBデータポートの差動データ入力信号(DP & DM)です。
SIE_RXD	Input	USB データ差動受信器 USBデータポートの差動信号を差動受信器で受信して抽出した論理信号の入力。 DP > DM : 1, DP < DM : 0
ROOT_DATAOUT_P, ROOT_DATAOUT_M	Output	USB データ出力 USBデータポートの差動データ出力信号(DP & DM)です。
ROOT_DATA_OE_	Output	USB ポート出力カイネーブル USBFCがUSBデータ出力信号(ROOT_DATAOUT_P & ROOT_DATAOUT_M)をドライブするときTRUEになります。
LDIN[7:0]	Input	データ入力 LDIN7 がMSBです。
LDOUT[7:0]	Output	データ出力 LDOUT7 がMSBです。
LDOUT_OE_	Output	データ出力カイネーブル USBFCがデータ出力信号(LDOUT)をドライブするときTRUEになります。
LA[4:0]	Input	アドレスバス USBFCのレジスタを選択するのに使います。
CS_	Input	チップセレクト ローカルバス上のデバイスがUSBFCのレジスタにアクセスするために使います。 DACK_がアサートされているときには無視されます。
IOR_	Input	I/O リード IOR_はCS_、LA[4:0]とともに用いられ、USBFCのレジスタあるいはFIFOからローカルバスにデータを読み出します。また、DACK_がアサートされている場合、DMA転送によるFIFOからの読み出しに使います。
IOW_	Input	I/O ライト IOW_はCS_、LA[4:0]とともに用いられ、USBFCのレジスタあるいはFIFOにローカルバスからデータを書き込みます。また、DACK_がアサートされている場合、DMA転送によるFIFOへの書き込みに使います。
DRQ	Output	DMA リクエスト FIFOに対する1バイトの書き込み／読み出しをおこなうことを外付けDMAコントローラにリクエストします。データ転送中、DACK_がアサートされるまでDRQは1を出力します。

DACK_	Input	DMA アクノリッジ DRQに対してDMAコントローラから出力されるDACK_をもちいてDMA転送がおこなわれます。IOR_とIOW_によってDMA転送の方向が決まります。
EOT_	Input	DMA 転送終了 外付けDMAコントローラから出力されるEOT_は、DMA転送を終了させるために使用されます。もし、DMA転送中にEOT_がアサートされると、現在転送中のバイトは転送されますが、以降の転送はおこなわれません。EOT_信号によっては割り込みを発生するようにプログラムすることができます。
INT_	Output	割り込み要求 ローカルバス上のCPUに割り込みをかけます。この割り込みを発生させることのできる要因は複数あります。詳細は「レジスタの説明」の章を参照してください。
DEVCFG_	Output	Device Configured. USBFCがUSBホストによってコンフィギュアされると、「真」になります。
SUSP_	Output	デバイス サスペンデッド USBFCがサスPEND状態になると「真」になります。
OSC_RUN	Output	オシレータ発振 1の時は CLOCK48_EARLYにクロックを入力し、0の時は入力を停止します。48MhzのクロックをUSBFC以外が使用しない場合には発信器のイネーブルに接続することも可です。
LCLK_OUT	Output	ローカルクロック出力 CLOCK48_EARLYにバッファを介した出力です。USBFCがサスPEND状態の時はドライブされません。
LRESET_	Output	ローカル リセット RESET_がアサートされるか、もしくはUSBポートリセットがおこなわれると、この負論理出力が「真」になります。
ISO_	Input	アイソクロナス モード セレクト エンドポイント3および4をアイソクロナス転送に設定するときアサート（真）します。バルク転送の場合はネガート（偽）します。
WAKEUP_	Input	ウェークアップ これをアサートするとリモート ウェークアップをおこないます。
BUSPWR_	Input	バス パワード この負論理入力は、デバイスがUSBバスによって電源の供給を受けることを示します。この入力が1の場合、デバイスは自己電源です。
PWRGOOD_	Input	パワー グッド 自己電源の電源供給が問題なくされているとき、この負論理入力をアサートします。
TEST	Input	テスト 通常動作ではグランドに接続してください。

3. 機能説明

3.1 USB インタフェース

USBFC は、USB ファンクションデバイスであり、そのためUSB ホストのスレーブになっています。USB ホストとUSBFC ポート間相互のUSBデータ転送は必ずUSBホストによって開始されます。USBFC には5 個のUSB エンドポイントがあります。

- エンドポイント0 このコントロールエンドポイントは、デバイスを初期化するのに使用されます。そして、USBコンフィグレーション、制御、およびステータスレジスタにアクセスします。
- エンドポイント1 このエンドポイントは、USBホストからUSBFC 受信メールボックスへのバルク転送をサポートします。
- エンドポイント2 このエンドポイントは、USBFC 送信メールボックスから USB ホストへのインターラプト送信をサポートします。
- エンドポイント3 このエンドポイントは、USB ホストからUSBFC受信FIFO へのバルクまたはアイソクロナスデータ転送をサポートします。
- エンドポイント4 このエンドポイントは、USBFC 送信FIFOから USB ホストへのバルクまたはアイソクロナスデータの転送をサポートします。

3.2 ローカルバス

バルクまたはアイソクロナスデータは、送信用・受信用それぞれ64 バイトのFIFOを通してローカルバスとUSB バス間を転送されます。ローカルバス上の CPU は USBFC 内の 1対のFIFOを通して USB バスにデータを送受信させます。1対の8-バイトメールボックスレジスタは、ローカルとホストの CPU 間のメッセージ交換に使用します。受信メールボックスは、バルクデータのエンドポイントです。また送信メールボックスはインターラプトエンドポイントです。

3.2.1 CPU制御によるUSBからローカルバスへの転送

ホストからデバイスへの転送では、ローカルとホストの CPU はまず1 ブロックのデータをホストのメモリからローカルメモリへ送信するための準備をします。USB ホストは、USB バスを通してUSBFC 内の受信 FIFO へバルクまたはアイソクロナスデータの送信を実行します。

FIFO がバルク転送中にフルになると、USB NAKをホストへ返しUSBFC がデータを受け入れられないということを伝えます。この場合、FIFO 内にあるパケットデータまたはCPU すでに読み出されたパケットデータは、廃棄してください。

ローカルCPU がこのエンドポイントをストールすると、USBFC は FIFO 内に全くデータを格納せず、STALLで応答します。

ローカル CPU は、USB ホストとの転送のセッティングを行った後、直ちに有効な FIFO データのポーリングを開始するか、またはパケット終了割り込みを待ちます。FIFO がUSB サイドから書き込まれます。一方、ローカル CPU は、いつデータを取り出すことができるかを調べるため、FIFO ステータスレジスタにポーリングを行います。もしくは、パケット終了割り込みを待ち、一度に全体のパケットを読み出します。

パケットが終了したら、ローカル CPU に対して割り込みを発生することができます。ローカル CPU は、パケットが ACK, NAK または STALL で検知されたかどうかをステータスレジスタで読み出すことができます。もし、ACKビットがひとつも1になっていない場合、タイムアウトが発生して

いたことになります。バルク転送完了時の NAK またはタイムアウト状態に対して、USB ホストは再度OUT トークンを送ります。そして、USBFC は同じパケットを再び受信することになります。

3. 2. 2 CPU制御によるローカルバスからUSBへの転送

デバイスからホストへの転送では、ローカル CPU は、まずデータブロックをローカルメモリから送信 FIFO に書き込む準備をします。ローカル CPU は、USBFC にデータを書き込んでいる間、送信 FIFO カウントレジスタをモニターし、FIFO の中に空きがあるかどうか常に把握している必要があります。

データブロックが送信 FIFO にロードされた後、ローカルとホストの CPU は、データブロックを送信 FIFO からホストのメモリへ転送する準備をします。USB ホストは、IN トークンを USBFC へ送信し、USB バルクまたはアイソクロナス転送による送信 FIFO からのデータ読み出しを開始します。ローカル CPU が USB バスに合わせて十分に速いレートでデータを供給することができる場合、CPU の書き込みおよび USB の読み出し動作は、同時に起こることができます。

送信 FIFO がエンプティになった時、USBFC は、EOP (End of packet) でパケットを終了させ、もうデータがないことを知らせます。End of packet が発生すると、ローカル CPU に割り込みが入ります。ローカル CPU は、ステータスポートを読み出すことによって、パケットがホストからの ACK 信号で認識されたかどうか、または USBFC が NAK または STALL 信号で IN トークンに応答したかどうかを検出します。アックビットが 1 つも 1 になっていない場合、タイムアウトが発生していましたことになります。バルク転送の完了時の NAK 信号またはタイムアウト状態に対して、USB ホストは、再度 IN トークンを送信します。そして、USBFC は、同じパケットを再び送信することになります。

3. 2. 3 DMA制御によるUSBからローカルバスへの転送

ローカルバス上でダイレクトメモリアクセス (DMA) コントローラを使用して、データを USBFC に送受信することもできます。ホストからデバイスへの転送では、ローカルとホストの CPU はまずデータブロックをホストメモリからローカルメモリへ転送する準備をします。ローカル CPU は DMA コントローラを デマンドモード転送にプログラムします。このモードでは、USBFC が転送を要求したときのみ転送が実行されます。USBFC の受信 FIFO からのデータを読み出しと、データのローカルメモリへの書き込みは同一のバストランザクションでおこなわれます。。 DMA アドレスカウンタは、ローカルメモリ中の目的のメモリブロックを指定するようにプログラムします。また、バイトカウントに、転送されるブロックのバイト数をプログラムします。

DMA コントローラをプログラムした後で、USBFC の DMA リクエストイネーブルビットを 1 にします。USB ホストは、USB バルク転送またはアイソクロナス転送をおこない、USB バスを通して USBFC 内の受信 FIFO へ送信が実行されます。もし、FIFO がバルクデータの転送中にフルになると、USBFC は、USB NAK をホストへ返しデータを受け入れられないということを伝えます。この場合、FIFO 内にあるパケットデータまたは CPU すでに読み出されたパケットデータは、廃棄してください。

ローカル CPU がこのエンドポイントをストールした場合、USBFC は、FIFO にはデータを格納せず、STALL で応答します。

FIFO 中にデータがある限り、USBFC は DRQ をアサートしローカル DMA 転送を要求します。そして、DMA コントローラは、ローカル CPU に対してローカルバスの使用権を要求します。DMA コントローラはバスの使用権を得ると、有効メモリアドレスを出力し、DACK_、IOR_ そして MEMW_ をアサートします。これによって、1 バイトのデータが USBFC 受信 FIFO からメモリへ転送されます。この処理は、DMA バイトカウントが 0 になるまで続きます。DMA が終了した時にローカルバスの割り込みが発生するようにプログラムすることができます。

EOP (end of packet) になった時にローカル CPU に対して割り込みを発生することができます。ローカル CPU は、パケットが ACK, NAK または STALL で検知されたかどうかをステータスレジスタで読み出すことができます。もし、アックビットがひとつも1になっていない場合、タイムアウトが発生していたことになります。バルク転送完了時の NAK またはタイムアウト状態に対して、USB ホストは再度OUT トークンを送ります。そして、USBFC は同じパケットを再び受信することになります。

予定より早い EOP (end of packet) は、DMA カウントが 0 でないことによりローカルCPU によって検出することができます。この場合ローカルとホストCPUが、どのように処理を進めるかを決定します。

3. 2. 4 DMA制御によるローカルバスからUSBへの転送

デバイスからホストへの転送では、ローカル CPU は、まずデータブロックをローカルメモリから送信FIFO に書き込む準備をします。ローカルCPU は DMA コントローラをデマンドモード転送にプログラムします。このモードでは、USBFC が転送を要求したときのみ転送が実行されます。ローカルメモリからのデータの読み出しと、USBFCの送信FIFOへのデータの書き込みは同一のバストランザクションでおこなわれます。DMA アドレスカウンタは、ローカルメモリ中の目的のメモリブロックを指定するようにプログラムします。また、バイトカウントに、転送されるブロックのバイト数をプログラムします。DMA コントローラがプログラムされた後、USBFCのDMAリクエストイネーブルビットを1にします。FIFO に空きスペースがありまたバイトカウントが0 にならない限り、USBFC は DRQ をアサートし DMA 転送を要求します。そして、DMA コントローラは、ローカルCPUに対してローカルバスの使用権を要求します。DMA コントローラは、ローカルバスの使用権を得ると、有効メモリアドレスを出力し、DACK, MEMR_ そして IOW_をアサートします。これによって、1 バイトのデータがメモリからUSBFC送信 FIFO へ転送されます。

DMA転送の開始後、ローカル CPU は USB ホストにバルクのデータの読み出しを開始するように、エンドポイント 2 を使用して信号を送ることができます。アイソクロナス転送の場合パケット送信はあらかじめ定められたインターバルで起こるので、USBFCから信号を送る必要はありません。USB ホストは、IN トークンを USBFC へ送信し、送信 FIFOからUSB バルクまたはアイソクロナスデータ転送を開始させます。DMA 転送は、DMA バイトカウントが 0 になるまで続きます。DMA 転送が終了した時にローカル CPU に対する割り込みを発生させることができます。

送信 FIFO が空になると、USBFC は、EOP (end of packet) でパケットを終了させ、データがもう無いということを伝えます。パケットが終了したら、ローカル CPU に対して割り込みを発生することができます。ローカル CPU は、ステータスポートを読み出すことによって、パケットがホストからのACK 信号で認識されたかどうか、またはUSBFC が NAK または STALL 信号で IN トークンに応答したかどうかを検出します。アックビットが1 つも1になっていない場合、タイムアウトが発生していたことになります。バルク転送の完了時のNAK 信号またはタイムアウト状態に対して、USB ホストは、再度 IN トークンを送信します。そして、USBFC は、同じパケットを再び送信することになります。

3. 2. 5 DMA 転送の終了

EOT_ 信号はDMA 転送を停止するのに使用され、通常外部の DMA コントローラが出力します。DACK_ およびIOR_または IOW_が同時にアサートされている状態で、EOT_をアサートします。これにより DMA コントローラはDMAが終了したことをUSBFCに知らせます。EOT_はDMAが終了したことを表しますが、その後もUSB転送は転送中のバイトがFIFOからUSBバスに送信されるまで続きます。EOT_は USBFCのDMAリクエストイネーブルビットをリセットします。

EOT_ 信号がDMA コントローラより供給されない場合でも、USBFCのDMAリクエストイネーブルビットをリセットすることによって、DMA 転送はいつでも停止することができます。 USBFCのDMAリク

エストイネーブルビットがDMAサイクルの途中でクリアされた場合、そのサイクルが完了した後にDMAリクエストが終了します。

3.2.6 USB エンドポイント 1 : 受信メールボックス

エンドポイント 1 は、USB ホストから受信メールボックスへのバルク転送に使用されます。受信メールボックスは8個のレジスタで構成されており、ローカル CPU によって読み出されます。メールボックスへ書かれるデータのフォーマットはユーザが定義します。8-バイトのパケットの転送は以下のようにおこなわれます。ホストがまずエンドポイント1 受信メールボックスレジスタへUSB 8-バイトバルク転送を実行します。USBFC側では、受信メールボックス有効ビット(レジスタ **IRQSTAT1** の第1ビット)が1になります。このビットは、ローカルバス割り込みを発生できるように設定することができます。受信メールボックス有効ビットが1のときにUSB ホストがメールボックスレジスタに書き込もうとすると、NAKが応答します。ローカル CPU が割り込みを受信した時、8-バイトを読み出し、受信メールボックス有効ビットをクリアします。そうするとUSB ホストは別の8-バイトパケットをこのエンドポイントに送信することができます。受信メールボックスにアクセスするためにインデックスポインタを使用します。インデックスポインタは、ローカル CPU によって初期化され、ローカル CPU が受信メールボックスデータレジスタを読み出した後、自動的にインクリメントされます。

3.2.7 USB エンドポイント 2 : 送信メールボックス

エンドポイント 2 は、送信メールボックスからUSBホストへのインタラプト転送に使用されます。送信メールボックスは8個のレジスタで構成されており、ローカル CPU によってデータを書き込まれます。8-バイトのパケットの転送は以下のようにおこなわれます。ローカル CPU が送信メールボックスレジスタにデータを書き込み、送信メールボックス有効ビットを1にします。ホストは、エンドポイント2 送信メールボックスレジスタからの8バイトのUSBインタラプト転送を実行します。USBインタラプト転送が完了すると、送信メールボックス有効ビットがクリアされます。CPU から送信メールボックスレジスタへの書き込みは、送信メールボックス有効ビットが0の時のみにおこなわれるようになります。こうすることで、前回のインタラプト転送が完了する前に送信メールボックスレジスタの値が変わってしまうことがないようにします。送信メールボックス有効ビットが1でない時、USB ホストがエンドポイント 2 を読み出そうとすると、NAK が返されます。送信メールボックスにアクセスするためにインデックスポインタを使用します。インデックスポインタは、ローカル CPU によって初期化され、ローカル CPU が送信メールボックスデータレジスタを読み出した後あるいは送信メールボックスデータレジスタに書き込んだ後、自動的にインクリメントされます。

3.3 サスペンドモード

USBバスが3ミリ秒間以上の非アクティブ状態になると、デバイスは低電力サスペンド状態に入らなければならぬことがUSB仕様によって定められています。その状態の時には、デバイスは $500\ \mu A$ 以上の電流を消費することはできません。これを実現するために、USBFC にはサスペンドリクエスト割り込みの機能があります。また、USB ステータスレジスタにはサスペンドビットがあります。加えて、USBFCはリモートウェークアップの機能を備えており、ローカルCPUによって、USB のサスペンドを終了させるリクエストを送信することができます。

3.3.1 サスペンドシーケンス

サスペンドに入る場合、通常の動作の流れは次の通りです。 USBFCのデバイス初期設定時に、サスペンドリクエスト時にUSBFCが割り込みを発生するように、ローカルCPU が**IRQENB1** レジスタを設定します。USB が3 ミリ秒間アイドル状態になると、USBFC はサスペンドリクエスト割り込みを発生します。

ローカル CPU は、この割り込みを受け、**IRQSTAT1** レジスタ中の対応するビットをクリアします。そして、必要なタスクを実行し、USB電源バスから $500 \mu A$ 以上の電流が消費されないようにします。準備が整ったら、ローカルCPUはUSBステータスレジスタの7 ビット(**USBSTAT**)に1を書き込み、サスペンドを開始します。

サスペンドモードでは、**OSC_RUN** がロー(Low)になりますので、USBFC への48Mhzクロックの供給を停止します。デバイスがサスペンド状態にある間は、**SUSP_** 出力端子は、ロー(Low)になります。USBFCがサスペンド状態にあるときは、入力信号はHまたはLにドライブされている必要があります。USBFC は、USB バス上でトラフィックを検出するか、ローカルCPU からのデバイスリモートウェークアップによってサスペンドモードから開放されます。

3.3.2 デバイスリモートウェークアップ

ローカル CPU は、**WAKEUP_** 入力端子をローにすることによってデバイスリモートウェークアップの信号を送ります。USBFC は10ミリ秒間のウェークアップ信号をUSBホストに送信すると同時に、**OSC_RUN**をドライブしクロック入力を再びスタートさせます。**WAKEUP_**がアサートされて2ミリ秒後に**SUSP_**がハイになり、 USBFCがウェークアップを完了させたことを示します。

3.3.3 ホスト側からのウェークアップ

ホストはUSB上でノンアイドル状態にすることによって、USBFCをウェークアップさせることができます。USBFCは、ホストのウェークアップ要求を検出し、 **OSC_RUN**をドライブしクロック入力を再びスタートさせます。2ミリ秒後、 **SUSP_**がハイになり、 USBFCがウェークアップを完了させたことを示します。

3.4 USBFC 電源について

USB 仕様では、バス電源と自己電源デバイスが定義されています。バス電源デバイスは、上流の USB コネクターから電力の供給を受ける周辺機器です。一方、自己電力デバイスは、外部の電源によって電力の供給を受けます。USBFC は、どちらのタイプにも対応しています。

デバイスの電力消費を十分に考慮して、バス電源か自己電源か、どちらのデバイスにするかを判断します。USB 仕様は、最大電流を次のように定めています。

- ホストでコンフィグレーションされていない機器 (USBFCがコンフィグレーションされたかどうかは、**DEVCFG_** 出力端子によって知ることができます)は、USB 電源ラインから100mA のみの供給を受けることができます。
- デバイスは、USB コネクタの電源端子から最大500 mAの供給を受けることができます。
- サスペンドモードにおいては、周辺機器はUSB コネクタの電源端子から最大 $500 \mu A$ のみの供給を受けることができます。

外部電源供給を行うことなく、上記のような電源供給の制限を満たすことができれば、その周辺機器はバス電源デバイスとして実現することができます。それ以外の場合は、自己電源デバイスとして設計をおこなうことになります。

自己電源デバイスの場合、USBポートから切り離されたときにUSBFCを低パワーサスペンドモードに強制的に入らせて、電力消費を押さえることができます。**USBSTAT** レジスタのビット7 を1にするとUSBFC を低電源サスペンドモードへ入らせます。デバイスがUSB に再接続されると、USBFC は、自動的にウェークアップします。デバイスがUSBポートから切断されていない場合には、強制的にサスペンドモードに入らせないようにしてください。

4. ローカルレジスタ

4.1 レジスタ説明

USBFC は、32 ビットローカルレジスタ領域を持ち、ローカルバス上でCPU よりアクセスすることができます。エンドポイント1 受信メールボックスレジスタは、USB ホストによって書き込まれ、エンドポイント2 送信メールボックスレジスタは、USB ホストによって読み出されます。USBFC が起動またはリセットされた後、レジスタには、それぞれのデフォルト値が設定されます。

使用されていないレジスタへの書き込みは無視されます。使用されていないレジスタからの読み出しは、0 を返します。

注意: DMA コントロールレジスタ中のUSBENBビットに1を書き込むまでは、USB デバイスデスクリプタとコンフィグレーションデスクリプタは、USB ホストで読み出すことはできません。従って、Device enumeration処理を完了することはできず、そのためデバイスはUSB 上で認識されません。

4.2 レジスター観

アドレス	レジスタ名	レジスタ説明	USB エンドポイント
0	DCTL	DMA コントロール	
1	IRQENB1	割り込みイネーブル 1	
2	IRQSTAT1	割り込みステータス1	
3	IRQENB2	割り込みイネーブル 2	
4	IRQSTAT2	割り込みステータス 2	
5-7		リザーブ	
8	EP1IDX	エンドポイント 1 インデックスレジスタ	
9	EP1DATA	エンドポイント 1 受信メールボックスデータポート (USB からローカルへ)	1
A-B		リザーブ	
C	EP2IDX	エンドポイント 2 インデックスレジスタ	
D	EP2DATA	エンドポイント 2 送信メールボックスデータポート (ローカルからUSB へ)	2
E	EP2POLL	エンドポイント 2 インタラプトポーリングインターバル	
F		リザーブ	
10	EP3DATA	エンドポイント 3 受信 FIFO データ	3
11	EP3COUNT	エンドポイント 3 受信 FIFO カウント	
12	EP3STAT	エンドポイント 3 受信 FIFO ステータス	
13	EP3PKSZ	エンドポイント 3 最大パケットサイズ	
14	EP4DATA	エンドポイント 4 送信 FIFO データ	4
15	EP4COUNT	エンドポイント 4 送信 FIFO カウント	
16	EP4STAT	エンドポイント 4 送信 FIFO ステータス	
17	EP4PKSZ	エンドポイント 4 最大パケットサイズ	
18	REVISION	USBFC レビジョン	
19	USBSTAT	USB ステータス	
1A	FRAMEMSB	フレームカウンタ MSB	
1B	FRAMELSB	フレームカウンタ LSB	
1C	EXTIDX	拡張レジスティンデックス	
1D	EXTDATA	拡張レジスタデータ	
1E-1F		リザーブ	

4.3 (アドレス00h; DCTL) DMA コントロールレジスタ

ビット	説明	リード	ライト	デフォルト値
7	リザーブ	Yes	No	0
6	ソフトウェアEOT このビットは、送信 FIFO がエンプティの時、エンドポイント4に対するINリクエストへの応答を決定します。このビットが1の場合、または EOT_ 入力がアサートされている場合、FIFO がエンプティであれば、USBFCは、ACK と長さ0のパケットでエンドポイント 4 に対するINリクエストに応答します。もし、このビットが0であり、かつ EOT_ 入力がアサートされていない場合、FIFO がエンプティであれば、USBFC は、エンドポイント 4 に対するINリクエストに応答してNACKを返し、さらにデータを送信しようとしていることをホストに伝えます。このビットは、データ値 0 を書き込むことによってクリアされます。FIFO がエンプティでUSBFC が長さ0のパケットでホストに応答するとこのビットは自動的にクリアされます。	Yes	Yes	0
5	USB イネーブル このビットが0である限りは、デバイスデスクリプタおよびコンフィグレーションデスクリプタをホストがリクエストしても、NAKが返されます。これを利用して、ホストがデスクリプタを読み出す前に、ローカル CPU がインターラプトポーリングレジスタ、最大パケットサイズレジスタ、そして他のコンフィグレーションレジスタ (プロダクト ID とベンダー ID等) を設定する時間を確保することができます。	Yes	Yes	0
4	エンドポイント 4 ストール このビットが設定されると、ホストが送信FIFOのバルクリードをおこなったとき、USBFC はSTALL をホストに返します。データはUSB ホストに送信されません。	Yes	Yes	0
3	エンドポイント 3 ストール このビットが設定されると、ホストが受信FIFO へのバルクライトをおこなったとき、USBFC はSTALL をホストに返します。データは、廃棄されます。	Yes	Yes	0
2	DMA リクエスト このステータスビットは、DRQ 出力端子の状況を示します。ローカル CPU はこのビットを見てDMA転送の状況をモニターできます。	Yes	No	0
1	DMA リクエストイネーブル このビットに 1 を書き込むと、USBFCはローカルバス上の DMA コントローラに対してDMA サイクルの要求を開始します。 EOT_ 入力がアサートされると、このビットは自動的にリセットされます。ローカルバス上の CPU はこのビットをリセットしてDMA 転送を終らせることができます。また、このビットを読み出すことでDMA転送が進行中であるかどうかを知ることができます。	Yes	Yes	0
0	DMA 方向 このビットは、DMA 転送時のデータフローの方向を決定します。1 = ローカルバスから USB へ; 0 = USB からローカルバスへ	Yes	Yes	0

4.4 (アドレス01h: IRQENB1) 割り込みイネーブルレジスタ 1

ビット	説明	リード	ライト	デフォルト値
7	サスPENDリクエスト割り込みイネーブル このビットが1の場合、USB ホストがUSBFC にサスPENDモードに入ることを要求した時、ローカル割り込み信号が発生します。	Yes	Yes	0
6	SOF 割り込みイネーブル このビットが 1の場合、USBFCがstart-of-frameパケットを受信した時、ローカル割り込みが発生します。	Yes	Yes	0
5	EOT 割り込み許可 このビットが1の場合、USBFCが DMA コントローラからのEOT_信号を受信した時、ローカル割り込みが発生します。	Yes	Yes	0
4	エンドポイント 4 割り込みイネーブル このビットが 1の場合、USBFC によってエンドポイント 4 からデータパケットが送信された時、ローカル割り込みが発生します。	Yes	Yes	0
3	エンドポイント 3 割り込みイネーブル このビットが1の場合、USBFC によってエンドポイント 3 にデータパケットが受信された時、ローカル割り込みが発生します。	Yes	Yes	0
2	エンドポイント 2 割り込みイネーブル このビットが1の場合、USB エンドポイント 2 受信メールボックスレジスタがUSB ホストにより読み出された時、ローカル割り込みが発生します。	Yes	Yes	0
1	エンドポイント 1 割り込みイネーブル このビットが1の場合、USB エンドポイント 1 送信メールボックスレジスタがUSB ホストによって書き込まれた時、ローカル割り込みが発生します。	Yes	Yes	0
0	リザーブ	Yes	No	0

4.5 (アドレス 02h; IRQSTAT1) 割り込みステータスレジスタ 1

注意: これらのステータスピットは、対応する割り込みイネーブルビットとは関係なく変化します。これらのビットに 0 を書き込んでも影響はありません。

ビット	説明	リード	ライト	デフォルト値
7	サスPENDリクエスト割り込みステータス ホストからのサスPEND要求が USBFC によって受信された時、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/CLR	0
6	SOF 割り込みステータス start-of-frameパケットが USBFC によって受信された時、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/CLR	0
5	EOT 割り込みステータス EOT_ 入力が DACK_およびIOR_ またはIOW_と同時にアサートされた時、このビットが1になり、DMA 転送の完了を表します。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/CLR	0
4	エンドポイント4 割り込みステータス USB エンドポイント 4 からデータパケットが USBFC によって送信された時、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/CLR	0
3	エンドポイント3 割り込みステータス(受信FIFO有効) USBFC によってエンドポイント 3 にデータパケットが受信された時、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/CLR	0
2	エンドポイント2 割り込みステータス USB エンドポイント 2 メールボックレジスタが USB ホストによって読み出された時、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/CLR	0
1	エンドポイント1 割り込みステータス(受信メールボックス有効) USB エンドポイント 1 メールボックスレジスタが USB ホストで書き込まれた時、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/CLR	0
0	上位割り込みアクティブ IRQSTAT2 レジスタのうち少なくとも1 つの割り込みステータスピットが1になっていることを示します。	Yes	No	0

4.6 (アドレス 03h; IRQENB2) 割り込み許可レジスタ 2

ビット	説明	リード	ライト	デフォルト値
7:2	リザーブ	Yes	No	0
1	送信 FIFO オールモストエンプティ割り込みイネーブル このビットが1の場合、送信 FIFO オールモストエンプティステータスピットが1になった時、ローカル割り込みが発生します。	Yes	Yes	0
0	受信 FIFO オールモストフル割り込みイネーブル このビットが1の場合、受信 FIFO オールモストフルステータスピットが1になった時、ローカル割り込みが発生します。	Yes	Yes	0

4.7 (アドレス 04h; IRQSTAT2) 割り込みステータスレジスタ2

注意: これらのステータスピットは、対応する割り込みイネーブルビットとは関係なく変化します。これらのビットに 0 を書き込んでも影響はありません。

ビット	説明	リード	ライト	デフォルト値
7:2	リザーブ	Yes	No	0
1	送信 FIFO オールモストエンプティステータス 送信FIFO中のデータのバイト数が送信FIFOオールモストエンプティスレショルドと等しくなり、かつ次のバイトがUSBバスに送信されると、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/Clr	0
0	受信 FIFO オールモストフルステータス 受信FIFO中のデータのバイト数が受信FIFOオールモストフルスレショルドと等しくなり、かつ次のバイトがUSBバスからFIFOへ受信されると、このビットが1になります。このステータスピットは、1 を書き込むことによってクリアされます。	Yes	Yes/Clr	0

4.8 (アドレス 08h; EP1IDX) エンドポイント 1 インデックスレジスタ

ビット	説明	リード	ライト	デフォルト値
7:3	リザーブ	Yes	No	0
2:0	エンドポイント 1 インデックスレジスタ エンドポイント 1 受信メールボックスデータポートが読み出される時、このレジスタによってどのエンドポイント 1 受信メールボックスレジスタがアクセスされるかが決定されます。このレジスタは、エンドポイント 1 受信メールボックスデータポートが読み出された後、自動的にインクリメントされます。インデックスレジスタは、最大数に到達すると 0 に戻ります。	Yes	Yes	0

4.9 (アドレス 09h; EP1DATA) エンドポイント 1 受信メールボックスデータ

ビット	説明	リード	ライト	デフォルト値
7:0	エンドポイント 1 受信メールボックスデータ このポートは、受信メールボックスレジスタの 1 つからデータを読み出すのに使用されます。エンドポイント 1 インデックスレジスタで選択されたレジスタからデータが読み出されます。8 個の受信メールボックスレジスタは、エンドポイント 1 への USB バルク転送によって書きこまれます。これによってUSBホストからローカルCPUへメッセージを送ることができます。そのメッセージのフォーマットや内容は、ユーザが任意に決めます。	Yes	USB	0

4.10 (アドレス *0Ch* ; *EP2IDX*) エンドポイント 2 インデックスレジスタ

ビット	説明	リード	ライト	デフォルト値
7:3	リザーブ	Yes	No	0
2:0	エンドポイント 2 インデックスレジスタ エンドポイント 2 送信メールボックスデータポートが読み出され／書き込まれる時、このレジスタによってどのエンドポイント 2 送信メールボックスレジスタがアクセスされるかが決定されます。このレジスタは、エンドポイント 1 受信メールボックスデータポートが読み出され／書き込まれた後、自動的にインクリメントされます。インデックスレジスタは、最大数に到達すると 0 に戻ります。	Yes	Yes	0

4.11 (アドレス *0Dh* ; *EP2DATA*) エンドポイント 2 送信メールボックスデータ

ビット	説明	リード	ライト	デフォルト値
7:0	エンドポイント 2 送信メールボックスデータ このポートは、送信メールボックスレジスタの 1 つに対するデータの読み出し／書き込みに使用されます。アクセスされるレジスタは、エンドポイント 2 インデックスレジスタで選択されます。8 つの送信メールボックスレジスタは、ローカル CPU によって書き込まれます。そして、エンドポイント 2 に対する USB インタラプト転送によって読み出されます。これによってローカル CPU から USB ホストへメッセージを送ることができます。そのメッセージのフォーマットや内容は、ユーザが任意に決めます。	Yes	Yes	0

4.12 (アドレス *0Eh* ; *EP2POLL*) エンドポイント 2 インタラプトポーリングインターバルレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	インタラプトポーリングインターバルレジスタ このレジスタは、ミリ秒単位で、エンドポイント 2 インタラプトポーリングインターバルを指定します。このインターバルは、エンドポイント 2 デスクリプタを通してホストから読み出されます。	Yes	Yes	0xFF

4.13 (アドレス *10h* ; *EP3DATA*) エンドポイント 3 受信 FIFO データレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	エンドポイント 3 受信 FIFO データレジスタ ： このレジスタは、ローカル CPU によって USB 受信 FIFO からデータを読み出すために使用されます。USB ホストによるエンドポイント 3 へのバルクまたはアイソクロンバス転送によって、受信 FIFO は書き込まれます。	Yes	No	0

4.14 (アドレス *11h* ; *EP3COUNT*) エンドポイント 3 受信 FIFO カウントレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	受信 FIFO カウント ： このレジスタは、受信 FIFO のデータの有効エンタリ一数を返します。値は、0 (エンプティ) から 64 (フル) までです。	Yes	No	0

4.15 (アドレス 12h; EP3STAT) エンドポイント 3 受信 FIFO ステータスレジスタ

ビット	説明	リード	ライト	デフォルト値
7:5	リザーブ	Yes	No	0
4	受信 FIFO フラッシュ ： このビットに1を書き込むと受信 FIFO がフラッシュされます。このビットを読み出すと、常に 0 が返ります。	Yes	Yes/Clr	0
3	受信 FIFO オーバーフロー ： 受信 FIFO がフルの時、USB ホストがさらに受信 FIFO へ書き込もうとすると、このビットが1になります。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
2	受信 FIFO アンダーフロー ： 受信 FIFO がエンプティの時、受信 FIFO を読み出そうとすると、このビットが1になります。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
1	受信 FIFO フル ： このビットが1のとき、受信 FIFO がフルであることを示します。	Yes	No	0
0	受信 FIFO エンプティ ： このビットが1のとき、受信 FIFO がエンプティであることを示します。	Yes	No	1

4.16 (アドレス 13h; EP3PKSZ) エンドポイント 3 最大パケットサイズレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	エンドポイント 3 最大パケットサイズレジスタ ： このレジスタは、エンドポイント 3 の最大パケットサイズを、8 バイト単位で（デフォルト 64 バイト）指定します。最大パケットサイズは、エンドポイント 3 デスクリプタを通してホストに読み出されます。	Yes	Yes	0x08

4.17 (アドレス 14h; EP4DATA) エンドポイント 4 送信 FIFO データレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	送信 FIFO データレジスタ ： このレジスタは、ローカルCPU によってUSB 送信 FIFO にデータを書き込むために使用されます。エンドポイント 4 からのバルクまたはアイソクロナス転送によって、USB ホストが送信 FIFO からデータを読み出します。	No	Yes	0

4.18 (アドレス 15h; EP4COUNT) エンドポイント 4 送信 FIFO カウントレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	送信 FIFO カウント ： このレジスタは、送信FIFOのデータの有効エンタリ一数を返します。値は、0 (エンプティ) から 64 (フル) までです。	Yes	No	0

4.19 (アドレス 16h; EP4STAT) エンドポイント 4 送信 FIFO ステータスレジスタ

ビット	説明	リード	ライト	デフォルト値
7:6	リザーブ	Yes	No	0
5	送信 FIFO 有効 ： このビットが1の場合、ホストは次の読み出しによって、FIFO 中のデータを読み出すことができます。このビットは、ホストの読み出しによって自動的にクリアされます。このビットはレジスタ FIFOCTL のビット 0 が1である場合のみ、使用することができます。	Yes	Yes	0
4	送信 FIFO フラッシュ ： このビットに1を書き込むと送信 FIFO がフラッシュされます。このビットを読み出すると、常に0が返されます。	Yes	Yes/Clr	0
3	送信 FIFO オーバーフロー ： 送信 FIFO がフルの時、ローカル CP U がさらに送信 FIFO へ書き込もうとすると、このビットが1になります。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
2	送信 FIFO アンダーフロー ： 送信 FIFO がエンプティの時、USB ホストが受信 FIFO を読み出そうとすると、このビットが1になります。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
1	送信 FIFO フル ： このビットが1のとき、送信 FIFO がフルであることを示します。	Yes	No	0
0	送信 FIFO エンプティ ： このビットが1のとき、送信 FIFO がエンプティであることを示します。	Yes	No	1

4.20 (アドレス 17h; EP4PKSZ) エンドポイント 4 最大パケットサイズレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	エンドポイント 4 最大パケットサイズレジスタ ： このレジスタは、エンドポイント 4 の最大パケットサイズを、8 バイト単位で（デフォルト 64 バイト）指定します。最大パケットサイズは、エンドポイント 4 デスクリプタを通してホストに読み出されます。	Yes	Yes	0x08

4.21 (アドレス 18h; REVISION) レビジョンレジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	レビジョン ： このレジスタは、USBFC の現在のレビジョン番号を返します。	Yes	No	Current Revision

4.22 (アドレス 19h; *USBSTAT*) USB ステータスレジスタ

ビット	説明	リード	ライト	デフォルト値
7	サスPEND制御: USB ホストからサスPEND要求があるとこのビットが1になります。1 を書き込むと、このビットはクリアされ、USBFCはサスPENDモードに入ります。	Yes	Yes/Clr	0
6	USBエンドポイント4 STALL: エンドポイント4がストールされていた (DCTLレジスタのビット4が1のとき) ため、前回のUSB IN トーカンに対してSTALLを返したことを示します。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
5	USB エンドポイント 4 NAK: ホストから最後に送信された USB パケット (IN パケット) に対して、FIFOアンダーランのため、NAKを返したことを示します。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
4	USB エンドポイント 4 ACK: ホストから最後に送信された USB パケット (IN パケット) に対するデータ転送が成功し、USB ホストからACKが返されたことを示します。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
3	USB エンドポイント 3 STALL: エンドポイント3がストールされていた (DCTLレジスタのビット3が1のとき) ため、ホストから最後に送信された USB パケット (OUT パケット) を受信することができず、STALLを返したことを示します。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
2	USB エンドポイント 3 NAK: ホストから最後に送信された USB パケット (OUT パケット) に対するデータを受け入れることができなかつたので、NAKを返したことを示します。受信データが破損していたということを意味するのでローカル CPU は FIFO をフラッシュします。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
1	USB エンドポイント 3 ACK: ホストから最後に送信された USB パケット (OUT パケット) に対するデータ転送が成功し、USB ホストにACKを返したことを示します。1 を書き込むとこのビットはクリアされます。	Yes	Yes/Clr	0
0	エンドポイント 2 有効: 8-バイトエンドポイント 2 メールボックスレジスタが、ローカル CPU によって書き込まれましたが、USB ホストにはまだ読み出されていない場合に、このビットが1になります。このビットが1である間は、ローカル CPU は送信メールボックスレジスタに書き込みを行ってはいけません。	Yes	Yes	0

4.23 (アドレス 1Ah; *FRAMESB*) フレームカウンタ MSB レジスタ

ビット	説明	リード	ライト	デフォルト値
7:3	リザーブ	Yes	No	0
2:0	フレームカウンタMSB: このレジスタは、最新のstart-of-frameパケットのフレームカウンタの値の上位ビットを含みます。	Yes	No	0

4.24 (アドレス 1Bh; *FRAMELSB*) フレームカウンタ LSB レジスタ

ビット	説明	リード	ライト	デフォルト値
7:0	フレームカウンタLSB: このレジスタは、最新のstart-of-frameパケットのフレームカウンタの値の下位ビットを含みます。	Yes	No	0

4.25 (アドレス 1Ch; EXTIDX) 拡張レジスタインデックス

ビット	説明	リード	ライト	デフォルト値
7:0	拡張レジスタインデックス : EXDATAポートに対する読み出し／書き込みをおこなうときにアクセスされる拡張データレジスタを選択します。	Yes	Yes	0

4.26 (アドレス 1Dh; EXDATA) 拡張レジスタデータ

ビット	説明	リード	ライト	デフォルト値
7:0	拡張データ : このポートによって、拡張データレジスタの1つにアクセスすることができます。レジスタのインデックスは、EXTIDX レジスタで保持されます。	See Below	See Below	See Below

4.26.1 (アドレス 1Dh, インデックス 00h; VIDMSB) ベンダー ID MSB

ビット	説明	リード	ライト	デフォルト値
7:0	ベンダーID MSB : このレジスタは、“Get Device Descriptor” リクエストによって読み出されるベンダーIDの上位バイトです。	Yes	Yes	0x04

4.26.2 (アドレス 1Dh, インデックス 01h; VIDLSB) ベンダー ID LSB

ビット	説明	リード	ライト	デフォルト値
7:0	ベンダー ID LSB : このレジスタは、“Get Device Descriptor” リクエストによって読み出されるベンダーIDの下位バイトです。	Yes	Yes	0xB8

4.26.3 (アドレス 1Dh, インデックス 02h; PIDMSB) プロダクト ID MSB

ビット	説明	リード	ライト	デフォルト値
7:0	プロダクト ID MSB : このレジスタは、“Get Device Descriptor” リクエストによって読み出されるプロダクトIDの上位バイトです。	Yes	Yes	0x88

4.26.4 (アドレス 1Dh, インデックス 03h; PIDLSB) プロダクト ID LSB

ビット	説明	リード	ライト	デフォルト値
7:0	プロダクト ID LSB : このレジスタは、“Get Device Descriptor” リクエストによって読み出されるプロダクトIDの下位バイトです。	Yes	Yes	0x21

4.26.5 (アドレス 1Dh, インデックス 04h; RELMSB) リリース番号 MSB

ビット	説明	リード	ライト	デフォルト値
7:0	リリース番号 MSB : このレジスタは、“Get Device Descriptor” リクエストによって読み出されるリリース番号の上位バイトです。	Yes	Yes	0x01

4.26.6 (アドレス1Dh, インデックス 05h; RELLSB) リリース番号 LSB

ビット	説明	リード	ライト	デフォルト値
7:0	リリース番号 LSB ：このレジスタは、“Get Device Descriptor” リクエストによって読み出されるリリース番号の下位バイトです。	Yes	Yes	0x00

4.26.7 (アドレス 1Dh, インデックス 06h; RCVAFTH) 受信FIFOオールモストフル スレショルド

ビット	説明	リード	ライト	デフォルト値
7:6	リザーブ	Yes	No	0x00
5:0	受信FIFOオールモストフル スレショルド ：このレジスタによって、受信FIFO オールモストフル スレショルド値を決定します。	Yes	Yes	0x3C

4.26.8 (アドレス1Dh, インデックス 07h; XMTAETH) 送信FIFOオールモストエンプティ スレショルド

ビット	説明	リード	ライト	デフォルト値
7:6	リザーブ	Yes	No	0x00
5:0	送信FIFOオールモストエンプティ スレショルド ：このレジスタによって、送信FIFO オールモストエンプティ スレショルド値を決定します。	Yes	Yes	0x04

4.26.9 (アドレス 1Dh, インデックス 08h; USBCTL) USB コントロール

ビット	説明	リード	ライト	デフォルト値
7:1	リザーブ	Yes	No	0x00
0	USB ストリングイネーブル ：このビットが1の場合、ベンダーIDとプロダクトIDストリングデスクリプタをホストへ返すことができます。このビットがクリアされると、デバイスデスクリプタのストリングインデックス値が0に設定されます。また、ストリングデスクリプタの読み出しに対してはSTALLが返されます。	Yes	Yes	0x1

4.26.10 (アドレス 1Dh, インデックス 09h; MAXPWR) 最大電力消費

ビット	説明	リード	ライト	デフォルト値
7:0	最大電流 ：USBポートから周辺機器に流入する電流量を、2mAの倍数で表します。USBFCは、コンフィグレーションデスクリプタでホストコントローラにこの値を報告します。デフォルト値は、500mA(0xFA * 2mA)です。	Yes	Yes	0xFA

4.26.11 (アドレス 1Dh, インデックス 0Ah; PKTCTL) パケット制御

ビット	説明	リード	ライト	デフォルト値
7	EP4 データトグルビット : USBホストからエンドポイント4への次のINトランクに対する応答を送る場合に送信される予定のデータトグルビットの値です。	Yes	Yes	0x0 / 0x1 (Toggle)
6	EP3 データトグルビット : USBホストからエンドポイント3への次のデータパケットのデータトグルビットの予期される値です。	Yes	Yes	0x0 / 0x1 (Toggle)
5	EP2 データトグルビット : USBホストからエンドポイント2への次のINトランクに対する応答を送る場合に送信される予定のデータトグルビットの値です。	Yes	Yes	0x0 / 0x1 (Toggle)
4	EP1 データトグルビット : USBホストからエンドポイント1への次のデータパケットのデータトグルビットの予期される値です。	Yes	Yes	0x0 / 0x1 (Toggle)
3	EP4データトグルモード : このビットが1の場合、EP4からのUSB転送の終わりで、データトグルビットを0にしてリセットします。このビットが0の場合は、データトグルビットは、リセットされずに、トグルします。	Yes	Yes	0x0
2	EP3データトグルモード : このビットが1の場合、EP3へのUSB転送の終わりで、データトグルビットを0にしてリセットします。このビットが0の場合は、データトグルビットは、リセットされずに、トグルします。	Yes	Yes	0x0
1	リザーブ	Yes	No	0x0
0	EP1 データトグルモード : このビットが1の場合、EP1へのUSB転送の終わりで、データトグルビットを0にしてリセットします。このビットが0の場合は、データトグルビットは、リセットされずに、トグルします。	Yes	Yes	0x0

4.26.12 (アドレス 1Dh, インデックス 0Bh; LOCALCTL) ローカル側制御

ビット	説明	リード	ライト	デフォルト値
7:1	リザーブ	Yes	No	0x00
0	ローカルロック出力 : このビットは、LCLK端子の出力を制御します。 0 : CLKIN 信号からの 48 MHz クロック 1 : USBビットストリームから抽出された12 MHz クロック (テスト目的のため)	Yes	Yes	0x0

4.26.13 (アドレス 1Dh, インデックス 0Ch; FIFOCTL) FIFO 制御

ビット	説明	リード	ライト	デフォルト値
7:1	リザーブ	Yes	No	0x00
0	送信 FIFO 有効モード : このビットが1の場合、FIFO有効ビットが (レジスタEP4STAT) に1が書き込まれない限り、ホストの送信FIFO (EP4) からの読み出し要求にNAK応答を行います。このビットが0の場合、送信FIFO中のデータはホストの読み出し要求に直ちに応じて送信されます。FIFO有効ビットは無視されます。	Yes	Yes	0x0

5. 標準デバイスリクエスト

5.1 コントロール`IN`トランザクション

5.1.1 Get Device Status

オフセット	バイト数	説明	デフォルト値
0	2	ビット 15:2 = リザーブ ビット 1 = デバイスリモートウェークアップ可能 ビット 0 = デバイスが自己パワーモードで動作する (PWROGOOD_ 入力端子によって決まります)	0x0001

5.1.2 Get Interface Status

オフセット	バイト数	説明	デフォルト値
0	2	ビット 15:0 = リザーブ	0x0000

5.1.3 Get Endpoint 0, 1, 2, 3, 4

オフセット	バイト数	説明	デフォルト値
0	2	ビット 15:1 = リザーブ bit 0 = エンドポイントがSTALLされている	0x0000

5.1.4 Get Device Descriptor (18 Bytes)

オフセット	バイト数	説明	デフォルト値
0	1	長さ	0x12
1	1	タイプ (デバイス)	0x01
2	2	USB 仕様リリース番号	0x0100
4	1	クラスコード	0x00
5	1	サブクラスコード	0x00
6	1	プロトコル	0x00
7	1	エンドポイント0の最大パケットサイズ	0x08
8	2	ベンダーID	ローカルレジスタで決定 VIDMSB、 VIDLSB
10	2	プロダクト ID	ローカルレジスタで決定 PIDMSB、 PIDLSB
12	2	デバイスリリース番号	ローカルレジスタで決定 PIDMSB、 PIDLSB
14	1	製造者を記述するストリングデスクリプタのインデックス	0x01
15	1	プロダクトを記述するストリングデスクリプタのインデックス	0x02
16	1	シリアル番号を記述するストリングデスクリプタのインデックス	0x00
17	1	コンフィグレーションの数	0x01

5.1.5 Get Configuration Descriptor (46 Bytes)

インターフェースデスクリプタとエンドポイントデスクリプタは、Get Configuration Descriptorによって、コンフィグレーションデスクリプタと一緒に返されるということに注意してください。

オフセット	バイト数	説明	デフォルト値
コンフィグレーション デスクリプタ			
0	1	長さ	0x09
1	1	タイプ (コンフィグレーション)	0x02
2	2	本デスクリプタに関して返されるデータの長さ	0x002E
4	1	インターフェース数	0x01
5	1	本コンフィグレーションの番号	0x01
6	1	このコンフィグレーションを記述するストリングデスクリプタのインデックス	0x00
7	1	属性 bit 7 = バスパワー (BUSBWR_ 入力端子による) bit 6 = 自己パワー (BUSBWR_ 入力端子による) bit 5 = リモートウェークアップ bits 4:0 = リザーブ	0x60 (自己パワー) 0xA0 (バスパワー)
8	1	必要USB パワーの最大値 (2 mA 単位で) (BUSBWR_ 入力端子による)	0x32 (自己パワー) 0xFA (バスパワー)
インターフェース0 デスクリプタ			
0	1	このデスクリプタのサイズ (バイト)	0x09
1	1	デスクリプタ タイプ (インターフェース)	0x04
2	1	このインターフェースの番号	0x00
3	1	オルタネート セッティング	0x00
4	1	このインターフェースで使用されるエンドポイントの数 (エンドポイント0以外)	0x04
5	1	クラスコード	0x00
6	1	サブクラスコード	0x00
7	1	デバイスプロトコル	0x00
8	1	このインターフェースを記述するデスクリプタのインデックス	0x00
エンドポイント1 デスクリプタ			
0	1	このデスクリプタのサイズ	0x07
1	1	デスクリプタ タイプ (エンドポイント)	0x05
2	1	エンドポイントアドレス ビット7 = 方向 (1 = IN、 0 = OUT) ビット 6:4 = リザーブ ビット 3:0 = エンドポイント番号	0x01
3	1	エンドポイント属性 ビット 7:2 = リザーブ ビット 1:0 00 = コントロール 01 = アイソクロナス 10 = バルク 11 = インタラプト	0x02
4	2	このエンドポイントの最大パケットサイズ	0x0008
6	1	エンドポイントをポーリングするインターバル (未使用)	0x00

Get Configuration Descriptor (続き)

オフセット	バイト数	説明	デフォルト値
エンドポイント2 デスクリプタ			
0	1	このデスクリプタのサイズ	0x07
1	1	デスクリプタタイプ (エンドポイント)	0x05
2	1	エンドポイントアドレス ビット 7 = 方向 (1 = IN、 0 = OUT) ビット 6:4 = リザーブ ビット 3:0 = エンドポイント番号	0x82
3	1	エンドポイント属性 ビット 7:2 = リザーブ ビット 1:0 00 = コントロール 01 = アイソクロナス 10 = バルク 11 = インタラプト	0x03
4	2	このエンドポイントの最大パケットサイズ	0x0008
6	1	エンドポイントをポーリングするインターバル	ローカルレジスタで決定 EP2POLL
エンドポイント3 デスクリプタ			
0	1	このデスクリプタのサイズ	0x07
1	1	デスクリプタタイプ (エンドポイント)	0x05
2	1	エンドポイントアドレス ビット7 = 方向 (1 = IN、 0 = OUT) ビット 6:4 = リザーブ ビット 3:0 = エンドポイント番号	0x03
3	1	エンドポイント属性 ビット 7:2 = リザーブ ビット 1:0 00 = コントロール 01 = アイソクロナス 10 = バルク 11 = インタラプト	バルク 0x02 アイソクロナス 0x01
4	2	バルクモード：このエンドポイントの最大パケットサイズ アイソクロナスマード：バス時間	ローカルレジスタで決定 EP3PKSZ
6	1	エンドポイントをポーリングするインターバル	バルク 0x00 アイソクロナス 0x01
エンドポイント4 デスクリプタ			
0	1	このデスクリプタのサイズ	0x07
1	1	デスクリプタタイプ (エンドポイント)	0x05
2	1	エンドポイントアドレス ビット7 = 方向 (1 = IN、 0 = OUT) ビット 6:4 = リザーブ ビット 3:0 = エンドポイント番号	0x84
3	1	エンドポイント属性 ビット 7:2 = リザーブ ビット 1:0 00 = コントロール 01 = アイソクロナス 10 = バルク 11 = インタラプト	バルク 0x02 アイソクロナス 0x01
4	2	バルクモード：このエンドポイントの最大パケットサイズ アイソクロナスマード：バス時間	ローカルレジスタで決定 EP4PKSZ
6	1	エンドポイントをポーリングするインターバル	バルク 0x00 アイソクロナス 0x01

5.1.6 Get String Descriptor 0

オフセット	バイト数	説明	デフォルト値
0	2	言語 ID (英語 = 09、 U.S. = 04)	0x0403 0x0409

5.1.7 Get String Descriptor 1

オフセット	バイト数	説明	デフォルト値
0	24	マニュファクチャラ デスクリプタ	0x1803 “SEIKO EPSON”

5.1.8 Get String Descriptor 2

オフセット	バイト数	説明	デフォルト値
0	66	プロダクト デスクリプタ	0x4203 “USB Interface Controller TEST2.0”

5.1.9 Get Configuration

オフセット	バイト数	説明	デフォルト値
0	1	現在のデバイスのコンフィグレーション番号を返します。	0x00

5.1.10 Get Interface

オフセット	バイト数	説明	デフォルト値
0	1	指定したインターフェースの現在のオルタネート セッティングを返します。	0x00

5.2 コントロール ‘OUT’ トランザクション

5.2.1 Set Address

オフセット	バイト数	説明	デフォルト値
--	0	デバイスのUSBアドレスを設定します。 値 = デバイスアドレス、インデックス = 0	--

5.2.2 Set Configuration

オフセット	バイト数	説明	デフォルト値
--	0	デバイスコンフィグレーションを設定します。 値 = コンフィグレーション値 (0 または 1 がサポートされています。)	--

5.2.3 Set Interface

オフセット	バイト数	説明	デフォルト値
--	0	指定したインターフェースのオルタネート セッティングを選択します。 値 = オルタネート セッティング インデックス = 指定したインターフェース	--

5.2.4 Device Clear Feature

オフセット	バイト数	説明	デフォルト値
--	0	選択されたDevice featureをクリアします。 値 = feature selector FS = 1 --> デバイスリモートウェークアップ (デイスエーブル)	--

5.2.5 Device Set Feature

オフセット	バイト数	説明	デフォルト値
--	0	選択されたDevice featureをセットします。 値 = feature selector FS = 1 --> デバイスリモートウェークアップ (イネーブル)	--

5.2.6 Endpoint 0, 1, 2, 3, 4 Clear Feature

オフセット	バイト数	説明	デフォルト値
--	0	選択されたエンドポイントのfeatureをクリアします。 値 = feature selector インデックス = エンドポイント番号 FS = 0 --> エンドポイントストール (ストールビットのクリア)	--

5.2.7 Endpoint 0, 1, 2, 3, 4 Set Feature

Offset	Number of Bytes	Description	Default Value
--	0	選択されたエンドポイントのfeatureをセットします。 値 = feature selector インデックス = エンドポイント番号 FS = 0 --> エンドポイントストール (ストールビットのセット)	--

5.3 エンドポイント1`OUT`トランザクション (受信メールボックス)

オフセット	バイト数	説明	デフォルト値
--	8	ホストはバルクOUTトランザクションによって受信メールボックスに8バイトを書き込みます。	--

5.4 エンドポイント2`IN`トランザクション (送信メールボックス)

オフセット	バイト数	説明	デフォルト値
--	8	ホストは、インターラプトINトランザクションによって送信メールボックスから8バイトを読み出します。	--

5.5 エンドポイント3`OUT`トランザクション (受信FIFO)

オフセット	バイト数	説明	デフォルト値
--	最大 64	ホストは、バルクまたはアイソクロナスOUTトランザクションによって受信FIFOにデータを書き込みます。	--

5.6 エンドポイント4`IN`トランザクション (送信FIFO)

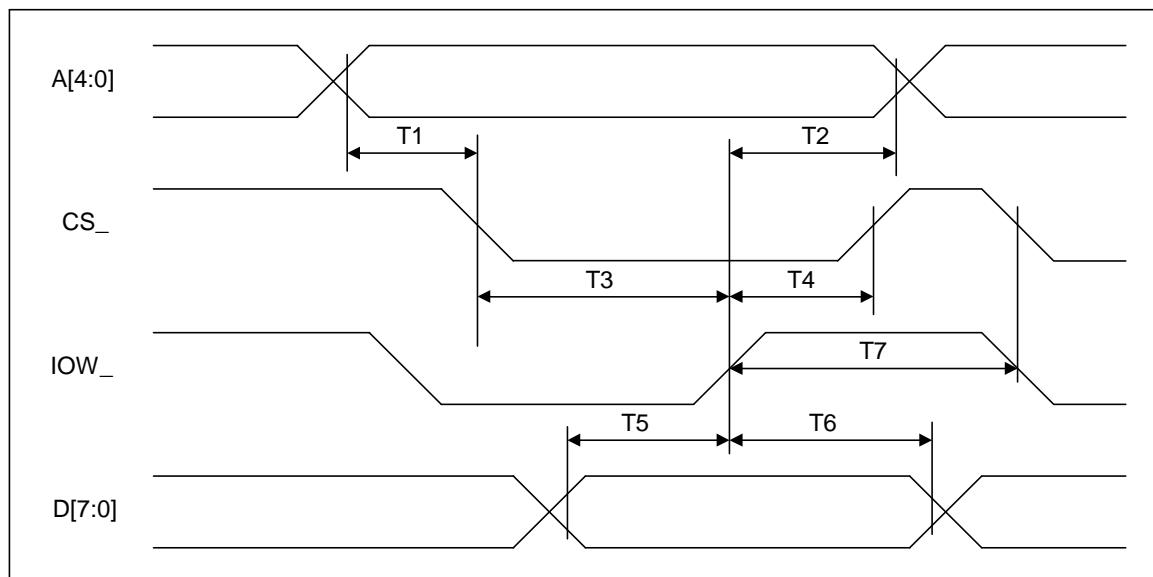
オフセット	バイト数	説明	デフォルト値
--	最大 64	ホストは、バルクまたはアイソクロナスINトランザクションによって送信FIFOからデータを読み出します。	--

7. タイミング

7.1 ローカルバスからレジスタへの書き込み

名 前	説 明	MIN	MAX	単位
T1	アドレス確定からライトイネーブルまで*	10		ns
T2	ライトイネーブル*の終端からのアドレスホールド	0		ns
T3	ライトイネーブル信号幅*	25		ns
T4	IOW_の終端からのCS_ホールド	0		ns
T5	データ確定からライトイネーブルの終端*まで	5		ns
T6	IOW_の終端からのデータホールドタイム	5		ns
T7	I/O リカバリータイム	60		ns

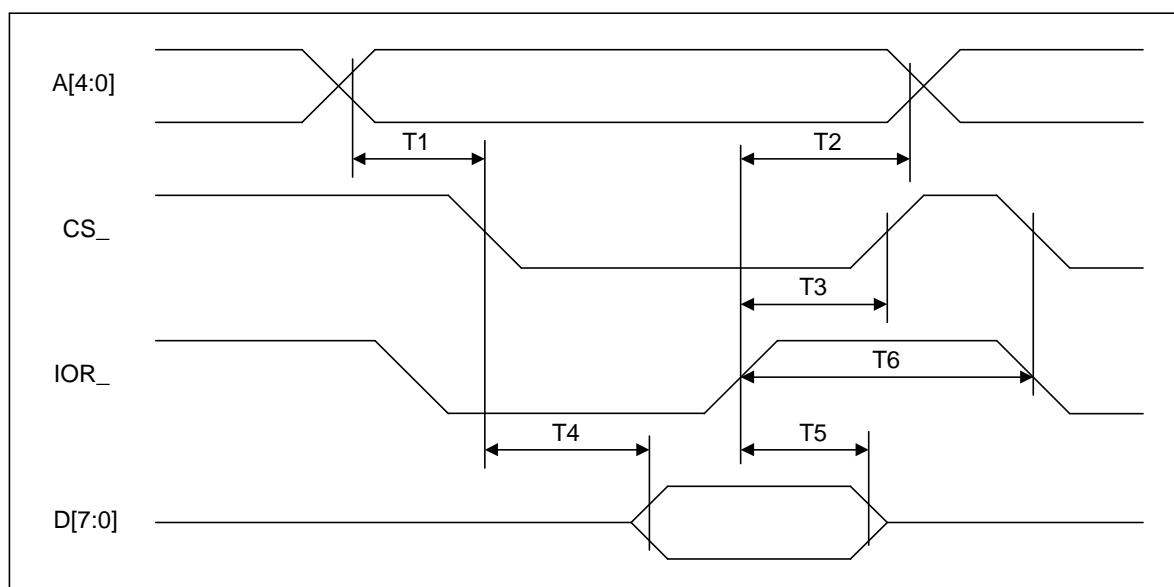
- ライトイネーブルは *IOW_* と *CS_* の両方がアサートされている期間のことを意味します。



7.2 レジスタからローカルバスへの読み込み

名前	説明	MIN	MAX	単位
T1	アドレス確定からリードイネーブルまで*	10		ns
T2	リードイネーブル*の終端からのアドレスホールド	0		ns
T3	IOR_の終端からのCS_ホールド	0		ns
T4	リードイネーブルから*からデータ確定まで		8	ns
T5	IOR_の終端からのリードデータホールドタイム	9		ns
T6	I/O リカバリータイム	60		ns

- リードイネーブルは *IOR_* と *CS_* の両方がアサートされている期間のことを意味します。

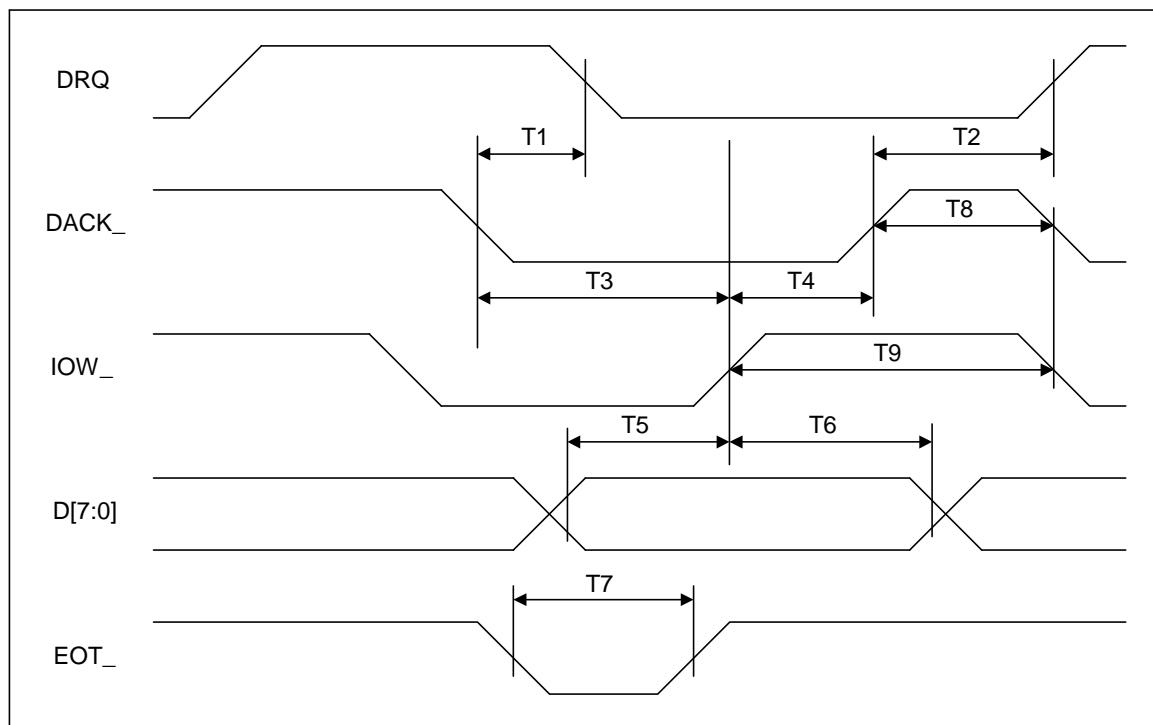


7.3 FIFOへのDMA書き込み

名前	説明	MIN	MAX	単位
T1	DACK_ 真からDRQ偽		58	ns
T2	DACK_ 偽 から DRQ 真	29		ns
T3	ライトイネーブル幅*	50		ns
T4	IOW_の終端からDACK_ ホールド	0		ns
T5	データ確定からライトイネーブルの終端*まで	50		ns
T6	IOW_の終端からのデータホールドタイム	25		ns
T7	EOT_ のパルス幅 (注意参照)	50		ns
T8	DACK_ リカバリータイム	30		ns
T9	DMA リカバリータイム	60		ns

* ライトイネーブルは IOW_ と DACK_ の両方がアサートされている期間のことを意味します。

注意: EOT_パルスが正しく認識されるためには、EOT_、IOW_、と DACK_ が、少なくとも T7 以上、同時に真である必要があります。

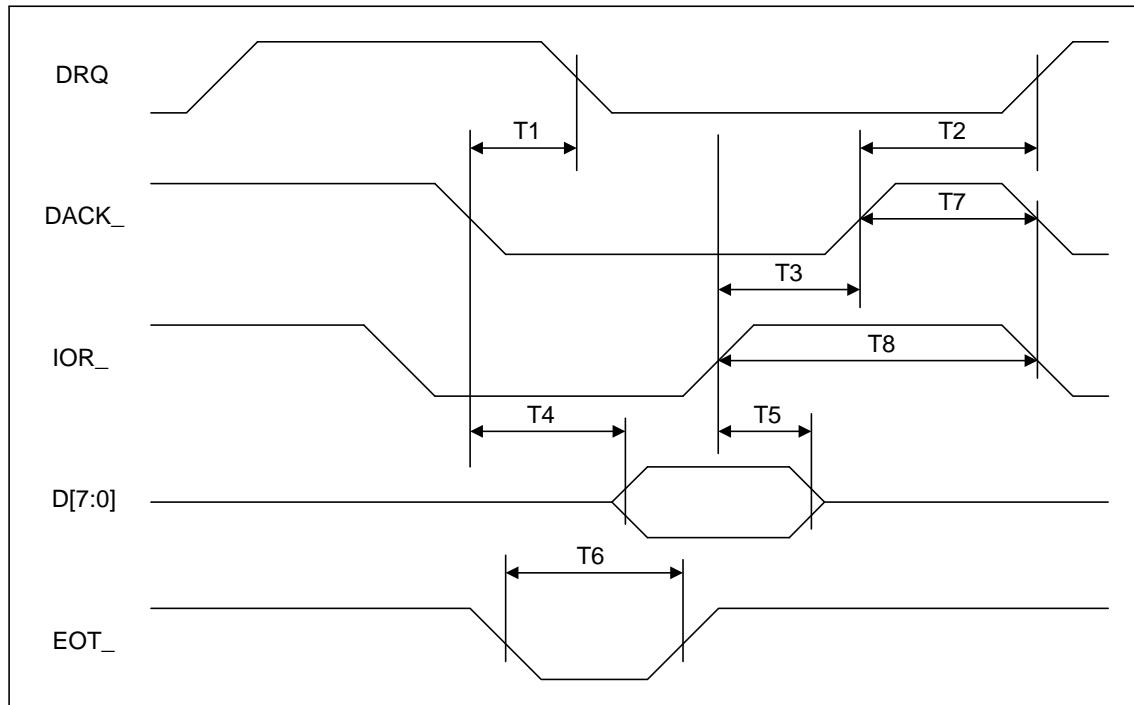


7.4 FIFOからのDMA読み出し

名前	説明	MIN	MAX	単位
T1	DACK_ 真からDRQ偽		58	ns
T2	DACK_ 偽 から DRQ 真	29		ns
T3	IOR_の終端からのDACK_ホールドタイム	0		ns
T4	リードイネーブルから*からデータ確定まで		8	ns
T5	IOW_の終端からのデータデータホールドタイム	9		ns
T6	EOT_ のパルス幅 (注意参照)	50		ns
T7	DACK_ リカバリータイム	30		ns
T8	DMA リカバリータイム	60		ns

* リードイネーブルは IOW_ と DACK_ の両方がアサートされている期間のことを意味します。

注意: EOT_パルスが正しく認識されるためには、EOT_、 IOW_、と DACK_ が、少なくとも T7 以上、同時に真である必要があります。



8. テスト回路

メガセル単独のテストモードの時は、図8.1のようなテスト回路設計をおこなってください。

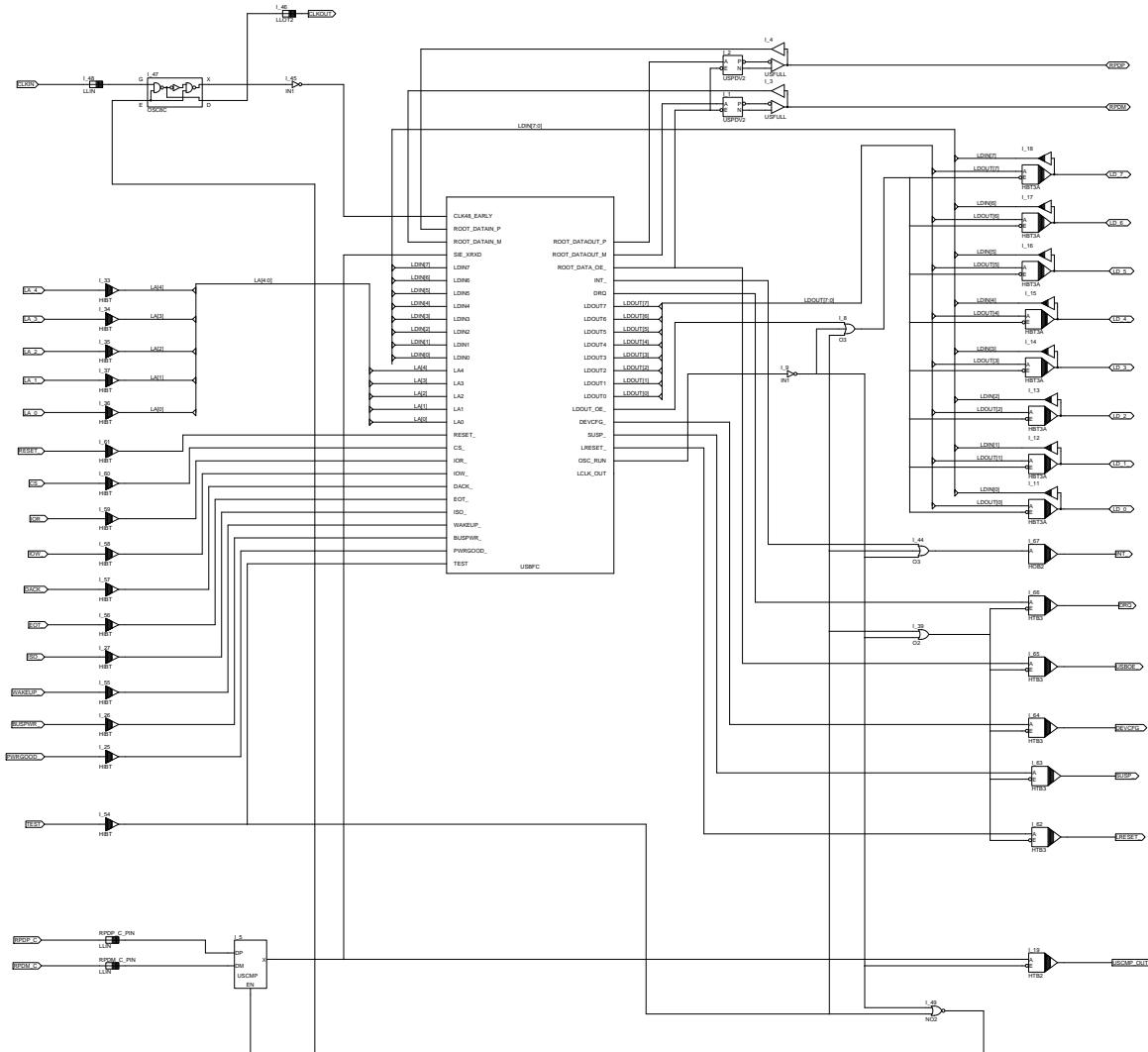


図8.1 テスト回路

図8.1において、発振セル(OSC8C)は参考までに入れてあります。ただし、OSC_RUNによって、USBFCへのクロック入力を制御できるようにしてください。

CLKOUT端子はテストパターンに含まれておりません。また、USBFCのLCLK_OUTはテスト項目の対象外です。

以上の回路図は、CPUが外付けの場合に、USBFCに最小限付加すべき回路の一例です。