



Design Gateway Co.,Ltd

本社: 〒184-0012
東京都小金井市中町 3-23-17
電話/FAX: 050-3588-7915
E-mail: sales@dgway.com
URL: www.dgway.com

特長

- USB3.0 規格 Revision1.0 の SuperSpeed (5.0Gbps) 通信機能を実装
- ホスト用コントローラを提供
- リンク・レイヤおよびプロトコル・レイヤを包含
- フィジカル・レイヤは TI 社製 PHY チップ(TUSB1310A)と直接インターフェース
- コア周波数は PIPE 接続部は 250MHz、内部は 125MHz 以上で選択可能
- USB 3.0 PIPE インタフェース 16bit サポート
- 最大 15 個の IN/OUT ルートポイントをサポート
 - コントロール 1 個
 - IN/OUT 各 7 個
- 全ての転送タイプをサポート(Control 転送、Bulk 転送、Isochronous 転送、Interrupt 転送)
- シンプルな Host プロセッサ向けトランザクション I/F および DMA I/F
- ALTERA 評価ボードによる購入前のコア実機評価が可能
- 安心の国内サポート

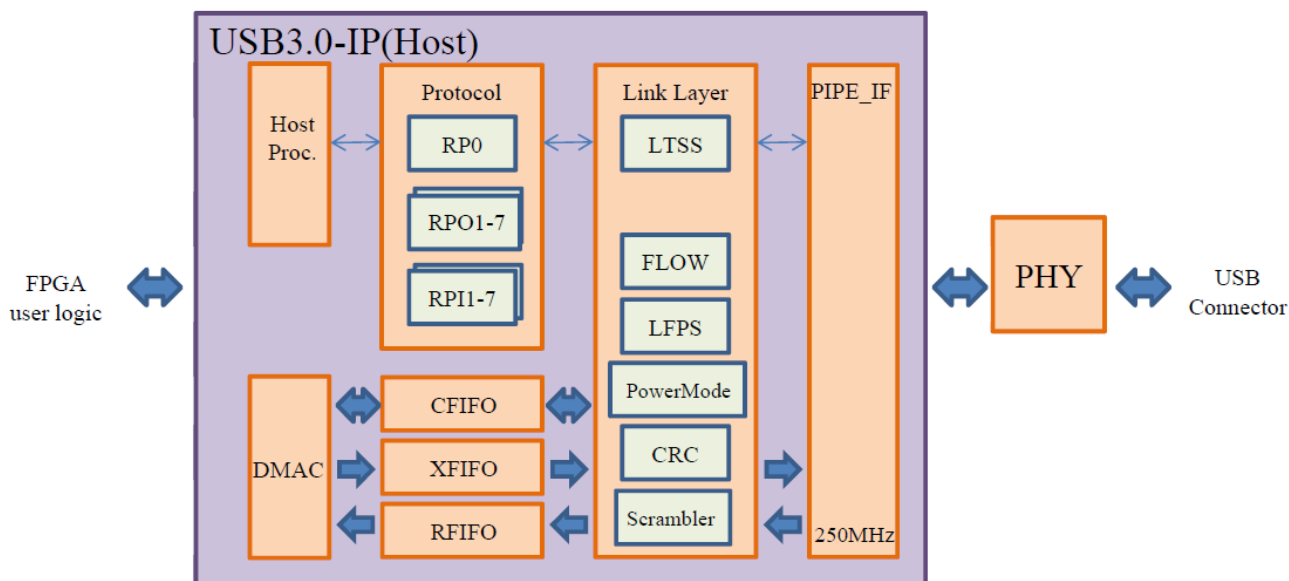
Core Facts	
コアの提供情報	
提供ドキュメント	リファレンスデザイン・マニュアル
提供形態	暗号化 HDL ファイル
サンプル	インスタンス化サンプル・プロジェクト
リファレンスデザイン アプリケーション情報	Quartus II/NIOS II プロジェクト (リファレンスデザインマニュアルに記載)
その他	ALTERA 評価ボードによる実機検証可 ArriaV GX スタータ開発キット用 Cyclone V E FPGA 開発キット用 Cyclone IV GX FPGA 開発キット用 Arria II GX FPGA 開発キット用 (Design Gateway 製 AB08-USB3HSMC カード が必要)
技術サポート	
デザインゲートウェイ・ジャパンによる国内サポート	

表1: コンパイル結果(コントロール 1 個, IN/OUT 各 2 個)

Family	Example Device	Fmax (MHz)	Combinational ALUTs ¹ / Logic Elements	Registers ¹	Pin ²	Block Memory bit	PLL	Design Tools
CycloneIV GX	EP4CGX150DF31C7	133	8,778	3,885	70	135,168	2	QuartusII 13.1
Arria II GX	EP2AGX125EF35C4	158	5,751	3,885	70	135,168	2	QuartusII 10.1
CycloneV E	5CEFA7F3117	140	5,816	4,417	70	135,168	2	QuartusII 14.0
ArriaV GX	5AGXFB3H4F35C4	142	5,816	4,305	70	135,168	2	QuartusII 14.0

備考:

- 1) 実際のロジック消費カウントはユーザロジックやフィット条件等に依存します
- 2) このサンプルはコアと TI_PHY 接続用回路の全 I/O とクロックがチップ外部と直接インターフェイスし、その他、内部信号は FF で取り囲んだケースでのコンパイル結果となります
- 3) コアのエンドポート数は可変です。Bulk 転送 FIFO 量は各 1 個へ削減可能です。



* LTSSM: Link Training and Status State Machine, LFPS: Low Frequency Periodic Signaling

図1 : USB3.0 IP ブロック図

アプリケーション情報

本コアは、低コストで 5.0Gbps の広帯域を必要とする USB3.0 対応ホスト・システムに最適です。本 IP ではほとんどの USB3.0 プロトコル(規格書の第6章の一部と第7章,第8章)をハードウェアで処理するため、ローエンド CPU での処理が可能となり、また、構成パラメータにより、最小限の制御回路(本仕様書ではルートポイントと呼ぶ)で実現可能なデバイス(マストレージクラスなど)から多数のルートポイントが必要なハイエンドなデバイスまで柔軟に対応可能です。本コアは USB2.0(480Mbps)およびそれ以前の低速規格をあえて省略し FPGA 内での占有リソースを最小限に抑えているため、5.0Gbps の広帯域通信を最も低コストで実装するソリューションを提供します。

概略

USB3.0 IP ホスト側コアはリンク・レイヤとプロトコル・レイヤを内蔵し、Host プロセッサが用意したメモリ上のデータのアドレス、転送長などをコア内蔵のレジスタへセットするだけで、パケットへの分割、CRC&スクランブルの付加、USB バス上のフロー制御などをコアのハードウェアが処理し、処理結果をレジスタへ反映します。データの受信も同様に処理されます。

IP コアの Host インタフェースはシンプルなレジスタアクセス・インタフェースで構成され、メモリへのシンプルな DMA アクセス・インタフェースを具備し、簡単に NIOS II プロセッサや ALTMEMDDR への接続が行えます。

PHY チップとの接続は、標準的な PIPE インタフェースに準拠しており、FPGA ポートとのタイミング調整用の Flip-Flop を経由して、ほぼそのまま接続が可能です。

IP コア内部のクロックは転送処理速度の関係から 125MHz 以上(125M×4 バイト=500MB/S)、PHY チップとの接続は 250MHz 固定(500MHz×2バイト)となっていますが、Host インタフェースや DMA インタフェースは必要な同期化を行った後、低速クロックでの接続が可能です。

USB3.0 IP ホスト側の内部ブロックの説明

USB3.0 IP ホスト側コアは大きく3つのブロックで構成されています。

プロトコル・レイヤ(Protocol Layer)

プロトコル・レイヤは、Host プロセッサからレジスタで指定されたメモリ上のデータの管理と、それを USB3.0 のパケットへ分割し、リンク・レイヤへ送信します。受信はその逆です。必要なデバイス(USB マスストレージなど)との End-to-End のシーケンス番号やクレジット処理はこのレイヤで行われます。Host プロセッサ上のソフトウェアにより、1つのルートポイントで複数のデバイスやエンドポイントを制御することも可能です。

(注: 本仕様書では、ホスト側の制御回路をルートポイントと呼びますが、一般的な用語ではないかもしれません。)

- **RP0(ルートポイント0)**
USB3.0 で規定されたコントロール転送を処理します。
すなわち、セットアップ・パケットの受領、コントロール用のデータ転送(IN/OUT)、ステータス転送の処理を行います。
- **RPO(ルートポイント・アウト1~7)**
BULK-OUT, INTERRUPT-OUT, ISOCHRONOUS-OUT の各転送を処理します。
- **RPI(ルートポイント・イン1~7)**
BULK-IN, INTERRUPT-IN, ISOCHRONOUS-IN の各転送を処理します。
- **MPP(マルチ・パーパスポイント)**
U0 状態への遷移直後の Port-Capabilities, Port-Configuration/Response の送受信処理の他、必要なアプリケーションでは Isochronous Time Stamp(ITS)や Link Management Packet(LMP)等の送受信処理にも使用します。
- **FIFO**
CFIFO は EP0 のデータ転送専用のデータ送受信 FIFO であり、XFIFO は RPO で共用されるデータ送信 FIFO(最大2面/自動アサイン)、RFIFO は RPI で共用されるデータ受信 FIFO(最大2面/自動アサイン)です。
- **DMAC,各種アービタ**
各エンドポイントからの DMA 要求、パケット送受信要求はプロトコル・レイヤ内部で調整され順次処理されます。

リンク・レイヤ(Link Layer)

リンク・レイヤは、プロトコル・レイヤからのパケットに各種 CRC(CRC-5/16/32)を付加し、スクランブルを掛け、8B(×4)シンボルで PIPE_IF へ送信します。リンク間の送受信、フロー管理もここで処理されます。

- **LSSSM ブロック**
USB3.0 で規定されたリンクの状態を管理し、初期化とパワー状態のシーケンス処理を行います。
- **FLOW ブロック**
リンク間のフロー制御を行います。すなわち Link Command を使って、リンク間の転送の正常・異常、リトライおよびクレジット処理を行います。
- **LFPS ブロック**
初期化時やパワーモードからの復帰時の、LFPS(Low Frequency Periodic Signaling)の送出処理、受領処理を行います。
- **Power Mode ブロック**
パワーモードへ投入時の、Link Command の送受信処理を行います。
- **送受信処理ブロック**
パケット送受信時に、CRC の付加/確認、スクランブルの付加/解除、Link Control ワードの付加/確認などを行います。

パイプ・インターフェース(PIPE Interface)

パイプ・インターフェースは、リンク・レイヤとの各種データや信号を、PIPE クロックに同期して PHY へ送受信します。受信はその逆です。

- **送信ブロック**
8B(×4)シンボルを 8B(×2)シンボルへ変換しシンボルの Elastic 処理や SKIP オーダーセットの挿入もここでを行います。(内部が 125MHz 動作の場合、PHY のクロックとの微妙な周波数差から送信側でも Elastic 処理が必要です。)
- **受信ブロック**
送信の逆の処理を行います。
- **制御ブロック**
リンク・レイヤの状態遷移などに応じて必要な信号を PIPE クロックに同期させて送ります。また、PHY からの信号を内部クロックに再同期してリンク・レイヤへ送ります。

FPGA 内部コントローラ(FPGA Controller)

FPGA 内部コントローラとしては一般的にはアプリケーション・ソフトウェアを実行する Host プロセッサが使われ、USB3.0 IP ホスト側コアをレジスタアクセスすることで、USB のデバイス・フレームワークより上位の制御(規格書の第9章以降)を構成を管理します。システム・コントローラは Host プロセッサ、DMA インタフェース、メモリなどから構成されます。

USB3.0 PHY

PHY は USB3.0 対応の外付けチップ(TI 製 TUSB1310A)を使用します。

IP コアの I/O 信号説明

コアの I/O 信号はユーザ回路と柔軟に接続できるようにするため、信号のみ定義され特定デバイスやピンには固定されておりません。下表 2 にコアの全 I/O 信号を説明します。
 なお、特に断りのない信号は正論理です。

表2: IP コアの I/O 信号

信号名	方向	説明
共通インターフェイス信号		
RST_N	In	コアのロジックをリセットするリセット信号、負論理。
CLK	In	IP コアの動作クロック。(125MHz)
PCLK	In	PIPE のクロック。(250MHz) PHY チップの生成する PIPE 用のクロックと同一の周波数(DCM 等で位相調整可)を入れる。また、PCLK 停止中は定常的に動作するクロック(CLK など)に切り替えてよい。
I_RPO_ENB[7:1]	In	ルートポイント・アウト(RPO)の実装・未実装を定義する。 最大7個実装でき、[1] RPO1, [7]RPO7 に対応する。1の時実装。
I_RPI_ENB[7:1]	In	ルートポイント・イン(RPI)の実装・未実装を定義する。 最大7個実装でき、[1] RPI1, [7]RPI7 に対応する。1の時実装。

(表 2: IP コアの I/O 信号 - 共通インターフェイス信号)

信号名	方向	説明
PIPE インターフェイス信号		
O_PIPE_READY	Out	PIPE クロック(PCLK)の動作中を示す。 P3 以外のステートに遷移後、LPHY_STATUS_ASYN の立ち上がりを検出してアサートする。
O_RX_TERMINATION	Out	受信終端の有無を制御する。1の時に終端有り。 PCLK に非同期に変化する。(PCLK 停止中でも変化する。)
O_TX_DETRX_ASYN	Out	相手の受信終端の有無の検出を制御する。 PCLK に非同期に変化し、P3 ステート時近辺(O_PIPE_READY ネグート)で使用する。
O_TX_IDLE_ASYN	Out	送信信号(SSTXP/SSTXN)を電氣的アイドル状態へ制御する。 PCLK に非同期に変化し、P3 ステート時近辺(O_PIPE_READY ネグート)で使用する。
O_PWR_DOWN_ASYN[1:0]	Out	PHY のパワーステートを制御する。00:P0, 01:P2, 10:P3, 11:P3 PCLK に非同期に変化し、P3 ステート時近辺(O_PIPE_READY ネグート)で使用する。
O_TX_DETRX_LPBK	Out	相手の受信終端の有無の検出を制御する。 PCLK に同期して変化し、P3 ステート時近辺以外(O_PIPE_READY アサート)で使用する。
O_TX_ELECIDLE	Out	送信信号(SSTXP/SSTXN)を電氣的アイドル状態へ制御する。 PCLK に同期して変化し、P3 ステート時近辺以外(O_PIPE_READY アサート)で使用する。
O_POWER_DOWN[1:0]	Out	PHY のパワーステートを制御する。00:P0, 01:P2, 10:P3, 11:P3 PCLK に同期して変化し、P3 ステート時近辺以外(O_PIPE_READY アサート)で使用する。
I_PWRPRESENT	In	VBUS に電源が供給している状態を入力する。PCLK に非同期に入力。
I_RX_ELECIDLE	In	受信信号(SSRXP/SSRXN)の電氣的アイドル状態を入力する。PCLK に非同期に入力。
I_PHY_STATUS_ASYN	In	PHY の各種状態の入力を制御する信号を入力する。PCLK に非同期に入力。 P3 ステートからの遷移後、PCLK の動作開始を検出する為に使用する。
I_RX_STATUS011_ASYN	In	相手の受信終端の有無の検出結果を入力する。PCLK に非同期に入力。 1 の時受信終端は有り。
I_PHY_STATUS	In	PHY の各種状態の入力を制御する信号を入力する。PCLK に同期して入力。 O_PIPE_READY のアサート中に、受信終端の有無の検出完了を表示する。
I_RX_STATUS011	In	相手の受信終端の有無の検出結果を入力する。PCLK に同期して入力。 LPHY_STATUS が1の時に有効で、1 の時受信終端は有り。
O_LANE_POLARITY	Out	RX のレーン極性を指定。0 は正転, 1 は反転。 オプションであり接続しない場合は、内部回路は削除。
O_TX_DATAK[1:0]	Out	送信シンボル(8B コード)が K コードか、D コードかを示す。1 の時 K コード。PCLK に同期。
O_TX_DATA[15:0]	Out	送信シンボル(8B コード)の2シンボル分。PCLK に同期。
I_RX_VALID	In	受信シンボルが有効なタイミングを示す。1 の時有効。PCLK に同期。
I_RX_DATAK[1:0]	In	受信シンボル(8B コード)が K コードか、D コードかを示す。1 の時 K コード。PCLK に同期。
I_RX_DATA[15:0]	In	受信シンボル(8B コード)の2シンボル分。PCLK に同期。

(表 2: IP コアの I/O 信号 - PIPE インターフェイス信号)

信号名	方向	説明
ホスト・レジスタ・インターフェイス信号		
I_LINK_REG_RE[15:0]	In	リンクレイヤの制御レジスタのリード信号。 16本分あるが全て実装されている訳ではない。1本は4バイト。CLKに同期し1サイクルアサートする。
I_LINK_REG_WE[15:0]	In	リンクレイヤの制御レジスタのライト信号。 16本分あるが全て実装されている訳ではない。1本は4バイト。CLKに同期し1サイクルアサートする。
I_PRTE_REG_RE[15:0]	In	プロトコルレイヤの制御レジスタのリード信号。 16本分あるが全て実装されている訳ではない。1本は4バイト。CLKに同期し1サイクルアサートする。
I_PRTE_REG_WE[15:0]	In	プロトコルレイヤの制御レジスタのライト信号。 16本分あるが全て実装されている訳ではない。1本は4バイト。CLKに同期し1サイクルアサートする。
I_XPP_REG_RE[511:0]	In	各エンドポイントの制御レジスタのリード信号。 各32本分あるが全て実装されている訳ではない。1本は4バイト。CLKに同期し1サイクルアサートする。
I_XPP_REG_WE[511:0]	In	各エンドポイントの制御レジスタのライト信号。 各32本分あるが全て実装されている訳ではない。1本は4バイト。CLKに同期し1サイクルアサートする。
O_RP_REG_RD[31:0]	Out	全制御レジスタのリードデータを示す。何れかの_REがアサートした次サイクルで有効。
I_RP_REG_WD0[31:0]	In	制御レジスタへのライトデータを入力する。何れかの_WEがアサートしたタイミングで有効。
I_RP_REG_WD1[31:0]	In	制御レジスタへのライトデータを入力する。何れかの_WEがアサートしたタイミングで有効。 RP14~7,RP04~7を使用する場合にI_RP_REG_WD0と論理的に同じ信号を接続する。負荷分散の為、異なるFFと接続することが望ましい。
O_RP_IRQ	Out	USBコアからの割り込み信号。1の時割り込み、レベル信号。
O_EXT_CNTL [3:0]	Out	USBコア外部制御用の信号。リンク・レイヤの制御レジスタでON/OFFできる。
O_LANE_POL	Out	RX Lane Pararity 制御信号 1: 反転, 0: 非反転
I_USB20_RESET	In	USB2.0からのリセットを入力。1でリセット。USB2.0コアがない場合は0に固定する。

(表 2: IP コアの I/O 信号 - ホスト・レジスタ・インターフェイス信号)

信号名	方向	説明
DMA アクセス・インターフェイス信号		
I_DMAC_IDLE	In	次の DMA アクセス開始が可能な状態を示す。
O_DMAC_REQ	Out	DMA アクセスの開始を示す。CLK に同期し 1 サイクルアサートする。
O_DMAC_ADR [31:0]	Out	DMA の開始アドレスを示す。O_DMAC_REQ アサート時に有効。
O_DMAC_U2M	Out	DMA の方向を示す。O_DMAC_REQ アサート時に有効。 0: Memory to USB, 1: USB to Memory。
O_DMAC_LEN[8:0]	Out	DMA の長さを示す。O_DMAC_REQ アサート時に有効。 0x100: 256 ワード(4K バイト)、0x001: 1ワード(4 バイト)
O_DMAC_BE [3:0]	Out	各バイトの有効/無効。
O_DMAC_DONE	Out	データ転送の完了を示す。最終データ転送の完了数サイクル後に 1 サイクルアサートする。
I_DMAC_M2U_VLD	In	DMA が Memory to USB の場合に、I_DMAC_M2U_DATA の有効なタイミングを示す。
I_DMAC_M2U_DATA[31:0] or [63:0]	In	DMA が Memory to USB の場合に、メモリからのデータを入力する。 モードはトップモジュールの verilog の PARAMETER “DMA64_MODE” で調整できる、 I_DMAC_M2U_VLD=1' b1 のタイミングで 4 バイト分(又は 8 バイト)の転送が完了する。(wait はない)
I_DMAC_U2M_WAIT	In	DMA が USB to Memory の場合に、メモリがデータを受け付けられないタイミングを示す。
O_DMAC_U2M_OUT	Out	DMA が USB to Memory の場合に、O_DMAC_U2M_DATA の有効なタイミングを示す。
O_DMAC_U2M_DATA[31:0] or [63:0]	Out	DMA が USB to Memory の場合に、メモリへのデータを出力する。 モードはトップモジュールの verilog の PARAMETER “DMA64_MODE” で調整できる、 O_DMAC_U2M_OUT=1' b1, I_DMAC_U2M_WAIT=1' b0 のタイミングで 4 バイト分(又は 8 バイト)の転送が完了する。

(表 2: IP コアの I/O 信号 - DMA アクセス・インターフェイス信号)

PIPE インターフェースの注意点

一般的な PIPE インタフェースですが、外付け PHY チップと接続する場合は信号の同期・非同期に注意が必要です。(図2参照)

入力信号には PHY からの PIPE クロック(PCLK)停止中でも参照する信号があり、その様な信号はピンと直接あるいは、コア内部のクロックで動作する FF で再同期した信号とします(ASYN 付)。PIPE クロック動作中は PIPE クロックに同期して参照する信号がある場合は2系統に分ける必要があります。

出力信号にも PIPE クロック(PCLK)停止中でも変化する信号があり、その様な信号はピンと直接あるいは、コア内部のクロックで動作する FF で再同期して出力します。(ASYN 付)。PIPE クロック動作中は同期して出力する信号がある場合は、ピンの直前で MUX するか、切り替えクロックで動作する FF で同期して出力します。

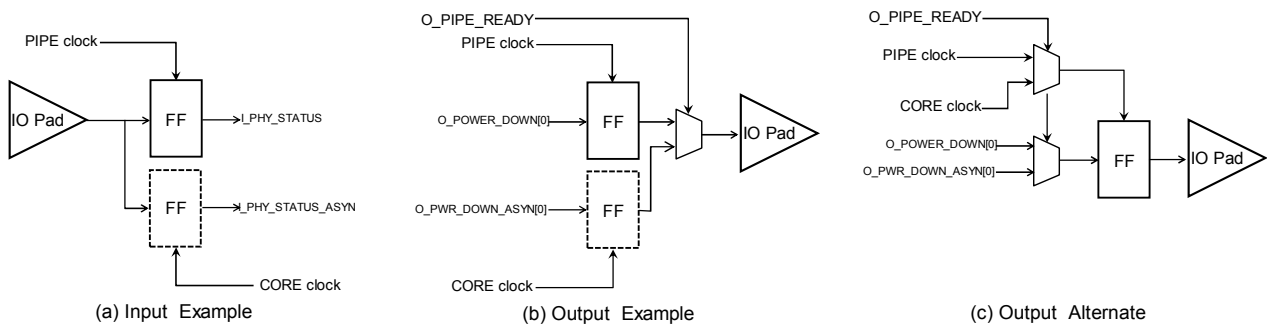


図2： PIPE インターフェース接続例

ホスト・レジスタ・インターフェースのタイミング

レジスタ・アクセスは図3されるように、アクセスしたいレジスタに相当する $I_{xxx_REG_RE}$ を1サイクルアサートするとそのレジスタのリード値が $O_{RP_REG_RD}$ に出力されます。また、 $I_{xxx_REG_WE}$ を1サイクルアサートしながらレジスタへのライト値を $I_{RP_REG_WD}$ に入力すると書き込みが行われます。

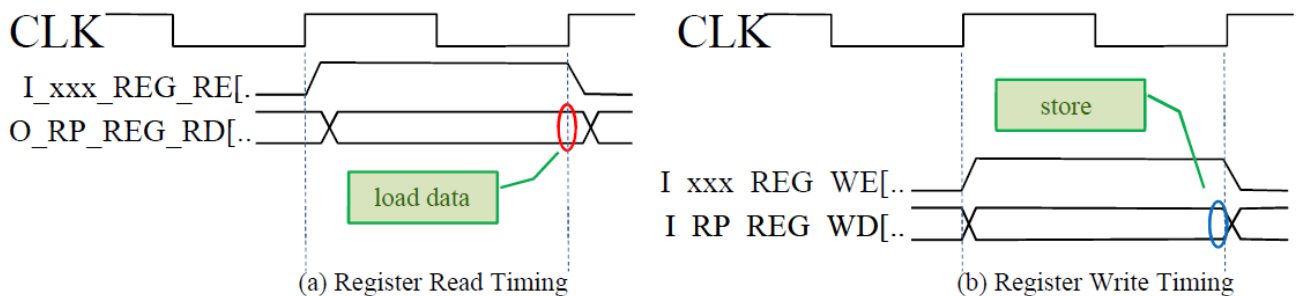


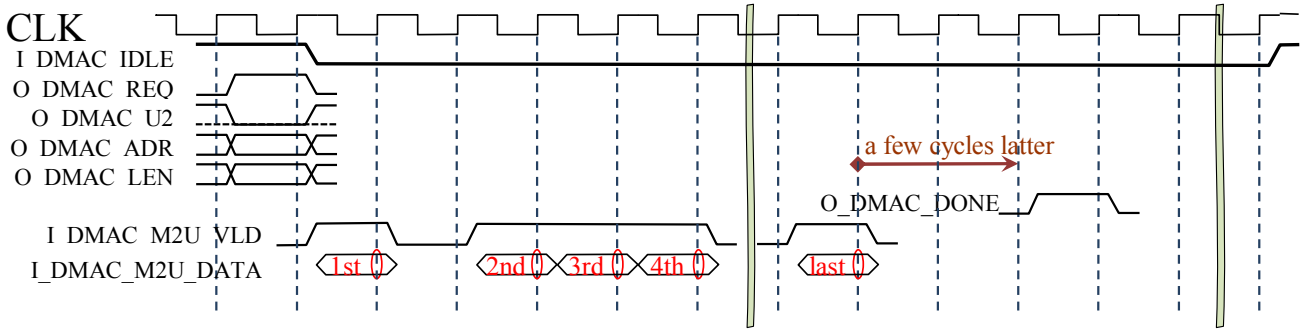
図3： ホスト・レジスタ・インターフェースの信号波形

DMA アクセス・インターフェースのタイミング

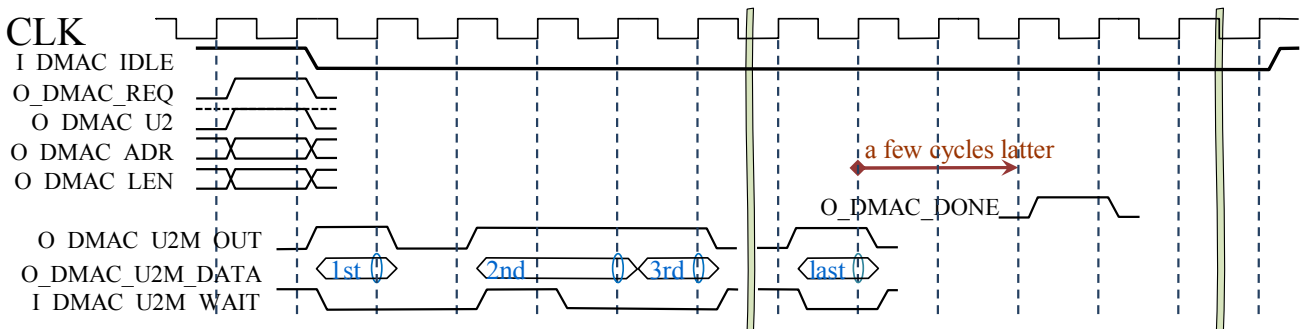
DMA アクセスは図4示されるように、I_DMAC_IDLE がアサートの時、O_DMAC_REQ が1サイクルアサートされます。この時、同時にアクセスアドレス、方向および長さが指定されます。

転送方向に応じて適宜 VLD, WAIT 信号等を使用してフロー制御が行われながらデータを転送します。

指定された長さの転送が完了すると、しばらく(数サイクル)して、O_DMAC_DONE が1サイクルアサートされ、DMA アクセスが完了します。



(a) DMA Read Timing (Memory to USB)



(b) DMA Write Timing (USB to Memory)

図4 : DMA アクセス・インターフェースの信号波形

IP コアの制御レジスタ説明

下表4に Host プロセッサからアクセス可能なコアの制御レジスタを説明します。

表4: IP コアの制御レジスタ

レジスタ名	R/W	オフセット	説明
リンク・レイヤレジスタ [ADDRESS=C_BASEADDR+LINK_BASE+OFFSET]			
LINK_CNFG	R/W	0h	リンク・レイヤ構成レジスタ
	USB_VALID	[0]	USB バス(LTSSM)の状態の SS_Disable 以外への遷移を可能とする。(動作可能状態)
	PMD_ENB	[1]	ホスト側からの Power Mode 遷移要求を受け付け可能とする。
	SCRIB_DIS	[2]	スクランブルを行わない。デバッグ用。
	EXT_CNT	[7:4]	O_EXT_CNTRL [1:0](I/O 信号参照)を ON/OFF する。
	U2_INACT_CNT	[15:8]	U1 から U2 へ遷移する時間(U2_Inactive_timer)のカウントを指定する。 Value=実時間/(USB_CLK × 0x10000)
LINK_CNTRL	W	1h	リンク・レイヤ制御レジスタ。1 ライトで指示する。未記載のビットは0とする。
	GO_RXDET	[0]	USB バス(LTSSM)の状態を SS_Disable から Rx_Detect_Active へ遷移させる。(動作開始) 事前に、LINK_CNFG[USB_VALID]を ON しておくこと。
	GO_RCOV	[3]	Active_U0 状態から Recovery 状態へ遷移させる。
	GO_BH_PORT_RESET	[7]	BH_PORT_RESET(WamReset)を発生させる。
	GO_PMD_NUM	[9:8]	GO_PMD で遷移させるパワーモードレベルを指定する。01:U1, 10:U2, 11:U3
	GO_PMD	[10]	パワーモードへの遷移を要求する。成功/非成功はホスト側の状態などによる。
	GO_U0	[11]	パワーモード(Active_U1/U2/U3)からの復帰を指示する。
LINK_IRQE	R/W	2h	リンク・レイヤ割り込みレジスタ
	IRQ	[0]	割り込み要求中。Read Only。
	ENB	[1]	IRQ が1の時、外部への割り込み信号(O_EP_IRQ。I/O 信号参照)を ON する。
LINK_LTSSM	R	3h	リンク・レイヤ状態レジスタ
	RCOV	[6:0]	リンク・レイヤがリカバリ(Recovery)状態へ遷移した要因を示す。ビット7が ON の間保持する。
	IRQ_FACTOR	[15:7]	リンク・レイヤが割り込んだ要因を表示。ただし、リンク・レイヤの状態は CPU の指示に関わらず変化する可能性がある。何れかのビットが ON の時、LINK_IRQE[IRQ]が ON となる。該当ビットへの1ライトでクリア。
	LTSSM	[28:24]	現在のリンク・レイヤ状態を表示する。状態は USB3.0 仕様に準拠。コード割り当ては別途。
	VBUS_OFF	[31]	現在の VBUS が OFF 状態である。

(表 4: IP コアの制御レジスタ - リンク・レイヤレジスタ)

レジスタ名	R/W	オフセット	説明
プロトコル・レイヤレジスタ [ADDRESS=C_BASEADDR+PRTE_BASE+OFFSET]			
PRTE_CNFG	R/W	0h	プロトコル・レイヤ構成レジスタ.
PRTE_CNTL	W	1h	プロトコル・レイヤ制御レジスタ. 1 ライトで指示する。未記載のビットは0とする。
	ARBT_RESET	[0]	プロトコル・レイヤのアービタのリセットを指示する。デバッグ用。
PRTE_IRQE	R/W	2h	プロトコル・レイヤ割り込み許可レジスタ
	RP0	[0]	RP0 からの割り込みを許可する。
	RRI	[7:1]	RPI1~7 からの割り込みを許可する。[1]: RPI1, [7]:RPI7
	MPP	[16]	MPP からの割り込みを許可する。
	RPO	[23:17]	RPO1~7 からの割り込みを許可する。[17]:RPO1, [23]RPO7
PRTE_IRQ	R	3h	プロトコル・レイヤ割り込み許可レジスタ
	RP0	[0]	RP0 から割り込みを要求中。IRQE の該当ビットが1の時、外部への割り込み信号を ON する。
	RPI	[7:1]	RPI1~7 からの割り込みを要求中。IRQE の該当ビットが1の時、外部への割り込み信号を ON する。 [1]: RPI1, [7]RPI7
	MPP	[16]	MPP からの割り込みを要求中。IRQE の該当ビットが1の時、外部への割り込み信号を ON する。
	RPO	[23:17]	RPO1~7 からの割り込みを要求中。IRQE の該当ビットが1の時、外部への割り込み信号を ON する。 [17]: RPO1, [23]RPO7
DMA_BASE	R/W	4h	DMA ベースレジスタ
	BASE	[31:28]	DMA を行うメモリアドレスの上位ビット

(表 4: IP コアの制御レジスタ - プロトコル・レイヤレジスタ)

レジスタ名	R/W	オフセット	説明
ルートポイントゼロ(RP0)レジスタ [ADDRESS=C_BASEADDR+XP_BASE+(0x80×0)+0x000+OFFSET]			
RP0_CNFG	R/W	00h	RP0 構成レジスタ。
	VALID	[0]	RP の動作を開始する。動作の開始前にはデータ送信の有無、長さ、方向を指示しておく必要がある。SetupDP パケットを送信後、デバイス側の応答状況に応じて各状態へ遷移する。
RP0_CNTL	R/W	01h	RP0 制御レジスタ。状態の表示と全ビットの更新が行える。 [13:0]の何れかのビットが ON で割り込み要求中。[23:16]は検出状態。[31:24]は過渡状態。
RP0_CLR	W	02h	RP0 制御レジスタクリア。1ライトで該当ビットをクリアする。
REP0_SET	W	03h	RP0 制御レジスタセット。1ライトで該当ビットをセットする。
	COMP	[0]	COMP 状態(転送完了)。クリアすると IDLE へ戻る。
	SETUP_ERDY	[1]	SetupDP 送信に対し、ACK(NumP=0)が応答された。もしくは、データ送受信(IN orOUT)に対し NRDY が応答された状態。デバイス側から ERDY が送信されると、WAIT_INOUT(データ送受信)もしくは WAIT_STT(STATUS 送信)へ遷移をする。クリアすると COMP 状態へ遷移する。(通常は何もしなくて良い。)
	WAIT_INOUT	[2]	データ送受信の開始前、ソフトウェアのリスタート待ち。クリアするとデータ送受信を開始する。(早く応答し過ぎると、応答できないデバイスが存在する。ソフトウェアで時間待ちが必要。)
	WAIT_STT	[3]	STATUS 送信の開始前、ソフトウェアのリスタート待ち。クリアすると STATUS 送信を開始する。(早く応答し過ぎると、応答できないデバイスが存在する。ソフトウェアで時間待ちが必要。)
	WAIT_ERDY	[10]	STATUS 送信に対し、NRDY が応答された状態。デバイス側から ERDY が送信されると、WAIT_STT (STATUS 送信)への遷移をする。クリアすると COMP 状態へ遷移する。(通常は何もしなくて良い。)
	WAIT_RECV	[14]	何らかのパケット待ち状態。BUSY が ON の時有効
	BUSY	[15]	IDLE 状態もしくは[13:0]状態)以外の時 ON。WAIT_RECV が ON の時、本ビットをクリアすると IDLE 状態へ戻る。また、どんな状態でも[14]と本ビットを一緒にクリアすると IDLE 状態へ戻る。
	ERROR	[16]	STALL パケットを受信するか、何らかの異常を検出した。
	OVER	[17]	デバイスの転送量が RP0_DLEN での指定より多い。(ホスト側が少ない)
	UNDER	[18]	デバイスの転送量が RP0_DLEN での指定より少ない。(ホスト側が多い)
	RETRY	[24]	IN 動作時に受信した DP パケットにエラーがあった。もしくは、DP 送信(OUT 動作時および SetupDP 送信)に対し Retry が返ってきた。
	RECV_STALL	[25]	STALL パケットを受信した。
	RECV_NRDY	[26]	NRDY パケットを受信した。
	RECV_ERDY	[27]	ERDY パケットを受信した。
RP0_DLEN	R/W	04h	RP0 データ長設定レジスタ
	DLEN	[20:0]	設定したデータ長。
	OUTRDY	[30]	OUT 転送(ホストからデバイス)指示。
	INRDY	[31]	IN 転送(デバイスからホスト)指示。
RP0_PLEN	R	05h	RP0 データ長結果レジスタ
	PLEN	[20:0]	実際に転送したデータ長。
	OUTRDY	[30]	OUT 転送(ホストからデバイス)指示。(DLEN レジスタのコピー)
	INRDY	[31]	IN 転送(デバイスからホスト)指示。(DLEN レジスタのコピー)
RP0_BFFR	R/W	06h	RP0 データ長バッファ(メモリアドレス)レジスタ
	BADDR	[27:8]	データを入出力するメモリの先頭アドレス
RP0_SEQN	R/W	07h	RP0 シーケンス番号レジスタ
	SEQN	[4:0]	パケットの現在のシーケンス番号。ライトもできるが通常必要ない。
RP0_DEST	R/W	08h	RP0 デスティネーション設定レジスタ
	DEV	[10:4]	デバイスアドレス
	RUT	[19:12]	ハブの Route String。下から4ビットづつ。(USB3.0 仕様書 8.9 参照)
RP0_SETUS0	R/W	09h	RP0 セットアップデータレジスタ0
RP0_SETUS1	R/W	0Ah	RP0 セットアップデータレジスタ1

USB3.0 IP ホスト側 プロトコル & リンクレイヤ・コア

SETUP0/1	[31:0]	SetupDP パケットで送信する内容。
----------	--------	----------------------

(表 4: IP コアの制御レジスタ - ルートポイントゼロ・レジスタ)

レジスタ名	R/W	オフセット	説明
ルートポイントアウト(RPO)レジスタ [ADDRESS=C_BASEADDR+XP_BASE+(0x80 × N)+0x400+OFFSET]			
RPO_CNFG	R/W	00h	RPO 構成レジスタ
	VALID	[0]	RPO の動作を開始する。動作の開始前にはデータ送信の長さを指示しておく必要がある。
	ISOCHR	[1]	Isochronous モードの転送を行う。
	BURST	[18:16]	バースト長の設定。4~1を設定する。
	FIFO_REQ	[30]	RPO の送信用 FIFO のアービトレーション要求状態。デバッグ用。リードのみ。
	FIFO_LOC	[31]	RPO が送信用 FIFO を専有するモード。デバッグ用。
RPO_CNTL	R/W	01h	RPO 制御レジスタ。状態の表示と全ビットの更新が行える。 [13:0]の何れかのビットが ON で割り込み要求中。[23:16]は検出状態。[31:24]は過渡状態。
RPO_CLR	W	02h	RPO 制御レジスタクリア。1ライトで該当ビットをクリアする。
RPO_SET	W	03h	RPO 制御レジスタセット。1ライトで該当ビットをセットする。
	COMP	[0]	COMP 状態(転送完了)。通常クリアすると IDLE へ戻る。もしくは、COMP_ERDY へ遷移する。
	COMP_ERDY	[8]	転送完了時の最後の ACK が NumP=0 であった。FLOW 制御状態となりデバイスから ERDY を受信すると IDLE へ遷移し次の転送を行える。デバイスから EDRY を受信しない場合は、ソフトウェアのタイムアウトにより IDLE へ移行させ次の送信を行う。(一部のデバイスで不要な FLOW 制御を行う場合があるため) なお、この状態では割り込みは発生しない。
	WAIT_ERDY	[10]	データ転送中にデバイスから NRDY を受信した。 デバイスから ERDY を受信した後、クリアするとデータ送信を再開する。EDRY 受信前にクリアするとこの状態に留まる。VAILD をクリアすると IDLE へ遷移する。
	WAIT_RECV	[14]	何らかの packets 待ち状態。BUSY が ON の時有効
	BUSY	[15]	(IDLE 状態もしくは[13:0]状態)以外の時 ON。WAIT_RECV が ON の時、本ビットをクリアすると IDLE 状態へ戻る。また、どんな状態でも[14]と本ビットを一緒にクリアすると IDLE 状態へ戻る。
	ERROR	[16]	STALL パケットを受信するか、何らかの異常を検出した。
	OVER	[17]	デバイスの転送量が多い。(ホスト側が少ない) Isochronous モードのみ。
	FLOW	[19]	フロー制御中である。
	RETRY	[24]	DP 送信に対し Retry が返ってきた。
	RECV_STALL	[25]	STALL パケットを受信した。
	RECV_NRDY	[26]	NRDY パケットを受信した。
	RECV_ERDY	[27]	ERDY パケットを受信した。
	RECV_ACK0	[28]	NumP=0 の ACK を受信した。
RPO_DLEN	R/W	04h	REPO データ長設定レジスタ
	DLEN	[20:0]	設定したデータ長。
	ZERO	[31]	データ長が 1KB の整数倍(0~)の場合、最後に0バイト長のデータパケットを送信する。
RPO_PLEN	R	05h	RPO データ長結果レジスタ
	PLEN	[20:0]	実際に転送したデータ長。0以外の時は何か受信した事を示す。
	ZERO	[31]	データ長が 1KB の整数倍(0~)の場合、最後に0バイト長のデータパケットを送信する。
RPO_BFFR	R/W	06h	RPO データ長バッファ(メモリアドレス)レジスタ
	BADDR	[27:8]	データを出入力するメモリの先頭アドレス
RPO_DEST	R/W	08h	RPO デスティネーション設定レジスタ
	EPN	[3:0]	エンドポイント番号(EPO)
	DEV	[10:4]	デバイスアドレス

	RUT	[19:12]	ハブの Route String. 下から4ビットづつ。(USB3.0 仕様書 8.9 参照)
RPO_SEQN	R/W	07h	RPO シーケンス番号レジスタ
	COMP_SEQN	[4:0]	転送完了(ACK 送信済み)したパケットのシーケンス番号。
	BFR_SEQN	[12:8]	バッファ(メモリ)へ格納したパケットのシーケンス番号。
	TRN_SEQN	[20:16]	送信したパケットのシーケンス番号。
	NUM_SEQN	[28:24]	ACK でクレジットを与えられたパケットのシーケンス番号。

(表 4: IP コアの制御レジスタ - ルートポイントアウト・レジスタ)

レジスタ名	R/W	オフセット	説明
ルートポイントイン(RPI)レジスタ [ADDRESS=C_BASEADDR+XP_BASE+(0x80×N)+0x000+OFFSET]			
RPI_CNFG	R/W	00h	RPI 構成レジスタ
	VALID	[0]	RP の動作を開始する。動作の開始前にはデータ送信の長さを指示しておく必要がある。
	ISOCHR	[1]	Isochronous モードの転送を行う。
	SW_WAIT_ERDY	[3]	WAIT_ERDY 状態で割り込みが発生し、ソフトウェアの介入をする。0: 介入なし。1: 介入あり。
	BURST	[18:16]	バースト長の設定。4~1を設定する。
	FIFO_REQ	[30]	RPI の受信用 FIFO のアービトレーション要求状態。デバッグ用。リードのみ。
	FIFO_LOC	[31]	RPI が受信用 FIFO を専有するモード。デバッグ用。
RPI_CNTL	R/W	01h	RPI 制御レジスタ。状態の表示と全ビットの更新が行える。 [13:0]の何れかのビットが ON で割り込み要求中。[23:16]は検出状態。[31:24]は過渡状態。
RPI_CLR	W	02h	RPI 制御レジスタクリア。1ライトで該当ビットをクリアする。
RPI_SET	W	03h	RPI 制御レジスタセット。1ライトで該当ビットをセットする。
	COMP	[0]	COMP 状態(転送完了)。通常クリアすると IDLE へ戻る。もしくは、COMP_ERDY へ遷移する。
	COMP_ERDY	[8]	転送完了時の最後の DP が EOB=1 であった。FLOW 制御状態となりデバイスから ERDY を受信すると IDLE へ遷移し次の転送を行える。デバイスから EDRY を受信しない場合は、ソフトウェアのタイムアウトにより IDLE へ移行させ次の送信を行う。(一部のデバイスで不要な FLOW 制御を行う場合があるため) なお、この状態では割り込みは発生しない。
	WAIT_ERDY	[10]	データ転送中にデバイスから NRDY を受信した。もしくは、受信した DP が EOB=1 であった。デバイスから ERDY を受信した後、クリアするとデータ送信を再開する。EDRY 受信前にクリアするとこの状態に留まる。VAILD をクリアすると IDLE へ遷移する。
	WAIT_RECV	[14]	何らかの packets 待ち状態。BUSY が ON の時有効
	BUSY	[15]	(IDLE 状態もしくは[13:0]状態)以外の時 ON。WAIT_RECV が ON の時、本ビットをクリアすると IDLE 状態へ戻る。また、どんな状態でも[14]と本ビットを一緒にクリアすると IDLE 状態へ戻る。
	ERROR	[16]	STALL パケットを受信するか、何らかの異常を検出した。
	OVER	[17]	デバイスの転送量が DLEN より多い。(ホスト側が少ない)
	UNDER	[18]	デバイスの転送量が DLEN より少ない。(ホスト側が多い)
	FLOW	[19]	フロー制御中である。
	FLOW2	[20]	フロー制御中である。
	RETRY	[24]	リトライ処理中である。
	RECV_STALL	[25]	STALL パケットを受信した。
	RECV_NRDY	[26]	NRDY パケットを受信した。
	RECV_ERDY	[27]	ERDY パケットを受信した。
RPI_DLEN	R/W	04h	RPI データ長設定レジスタ
	DLEN	[20:0]	設定したデータ長。
RPI_PLEN	R	05h	RPI データ長結果レジスタ
	PLEN	[20:0]	実際に転送したデータ長。0以外の時は何か送信した事を示す。
RPI_BFFR	R/W	06h	RPI データ長バッファ(メモリアドレス)レジスタ
	BADDR	[27:8]	データを入出力するメモリの先頭アドレス
RPI_DEST	R/W	08h	RPI デスティネーション設定レジスタ
	EPN	[3:0]	エンドポイント番号(EPI)
	DEV	[10:4]	デバイスアドレス
	RUT	[19:12]	ハブの Route String。下から4ビットづつ。(USB3.0 仕様書 8.9 参照)
RPI_SEQN	R/W	07h	RPI シーケンス番号レジスタ
	COMP_SEQN	[4:0]	転送完了(Ack 受領済み)したパケットのシーケンス番号。
	BFR_SEQN	[12:8]	バッファ(メモリ)から取り込んだパケットのシーケンス番号。
	RCV_SEQN	[20:16]	受信したパケットのシーケンス番号。

(表 4: IP コアの制御レジスタ - ルートポイントイン・レジスタ)

レジスタ名	R/W	オフセット	説明
マルチパーパスポイント(MPP)レジスタ [ADDRESS=C_BASEADDR+XP_BASE+(0x80 × 0)+0x400+OFFSET]			
MPP_CNFG	R/W	00h	MPP 構成レジスタ
	TRNS	[0]	MPP からパケットを送信する。完了するとクリアされる。Active_U0 になった時は、自動的に ON/OFF して、Port Capabilities, Port Configuration Response を送信。
MPP_THD0	R/W	04h	MPP 送信パケット0
MPP_THD1	R/W	05h	MPP 送信パケット1
	THD0/1	[31:0]	送信パケットのデータ。
MPP_RHD0	R/W	06h	MPP 受信パケット0
MPP_RHD1	R/W	07h	MPP 受信パケット1
	RHD0/1	[31:0]	受信パケットのデータ。
MPP_RCVD	R/W	08h	EPI シーケンス番号レジスタ
	MISC_RCNT	[3:0]	パケット受信で+1する。0xF になると 0x0 へ戻る。ライトも可能。
	MISC_RCVD_H	[7]	何かパケットを受信した。ON で割り込み要求。ライト可能。
	PCFG_RCVD	[30]	Active_U0 になって以降、Port Configuration を受信した。
	PCAP_RCVD	[31]	Active_U0 になって以降、Port Capabilities を受信。

(表 4: IP コアの制御レジスタ - マルチパーパスポイント・レジスタ)

IP コアの制御レジスタマップ

L_{xxx}_REG_RE/WE[...]_{...}をアドレス順接続した場合のコアのレジスタ配置の概略を図5に示します。

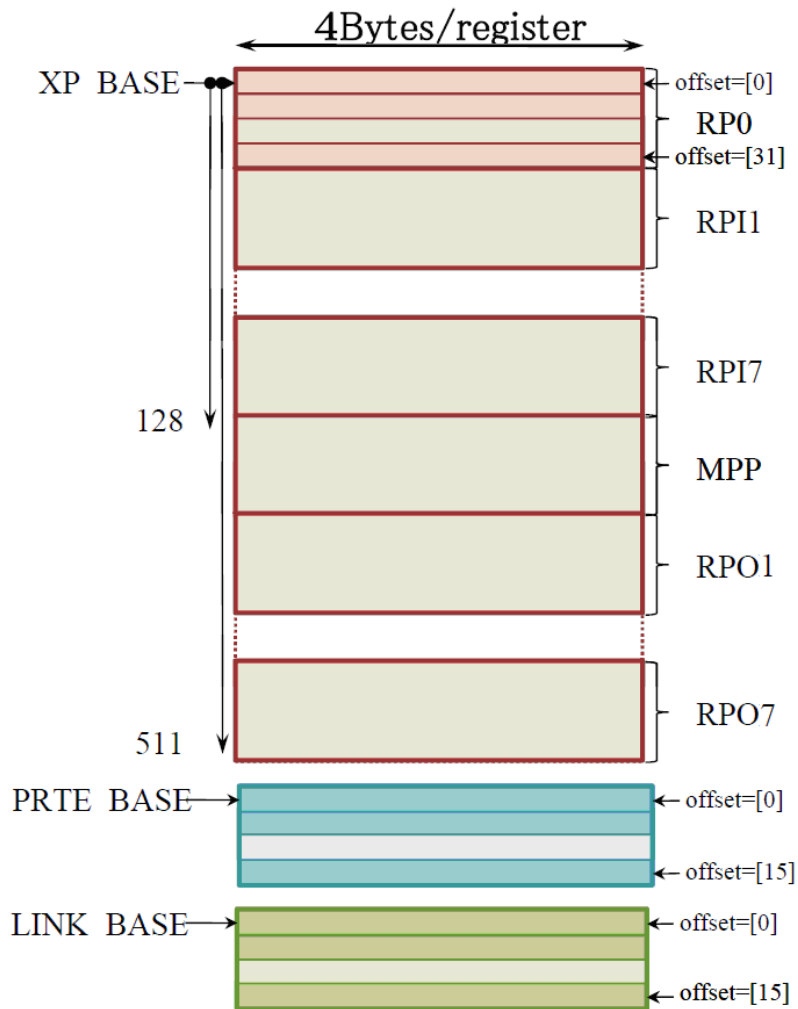


図 5 : IP コア内部レジスタのアドレス・マップ

IP コアの制御手順の概略

本コアを制御する主な手順について概略のみ説明します。

USB バス初期化

USB バスにデバイスを接続して起動する、もしくは、起動後に USB バスにデバイスを接続すると、バス動作を開始できます。その後、U0 ステートまでの概略手順を以下に示します。

- ① 制御レジスタ(たとえば、LINK_CNFG[EXT_CNT])で、VBUS を ON とする。
- ② LINK_CNFG[USB_VALID]で、動作可能状態とする。
- ③ LINK_CNTRL[GO_RXDET]で、Rx_Detect_Active 状態への遷移を指示する。
- ④ デバイス側も動作可能であれば、自動的にリンクする。
LINK_LTSSM[LTSSM]で、Active_U0 状態になればリンク成功。SS_disable に戻れば②からリトライする。

USB デバイスのコンフィグレーション(Configured)状態まで

リンク成功後、RP0 によるコントロール転送によって、コンフィグレーション(Configured)状態になるまでの概略手順を以下に示します。

- ① RP0_CNFG[VALID]で、RP0 を動作可能状態とする。
 - 最初の SetupDP は、SET_ADDRESS したがって、RP0_DEST[DEV]でアドレスは0とする。
 - SetupDP パケットを送信し ACK の受信にする。SET_ADDRESS でデータ転送なしなので、WAIT_STT 状態になる。
 - STATUS パケットを送信し、最終 ACK パケット(RESET)を送信する。
- ② 以後は、RP0_DEST[DEV]でアドレスを指定する。デバイスはアドレス(Address)状態である。
- ③ その後も、SetupDP パケット等のコントロール転送を何回か行う。
 - コントロール転送がデータ転送を伴う場合は、受信バッファをメモリに確保するか (IN 転送)、送信データをメモリに設定して (OUT 転送)、RP0_BFFR と RP0_DLEN へ指示する。
 - SetupDP パケットを送信し ACK の受信にする。データ転送(IN/OUT)を伴う場合、WAIT_INOUT 状態になる。
 - データ転送の完了後、STATUS パケットを送信する。後はデータ転送なしと同様。
- ④ 最後に、SET_CONFIGURATION を送付して、コンフィグレーション(Configured)状態となる。

USB デバイスの BULK_IN, BULK_OUT 転送

デバイスがコンフィグレーション(Configured)状態になった後、BULK_IN(RPI)、BULK_OUT(RPO)を使った転送です。転送内容自体はデバイス・クラス毎に異なります。以下は BULK_IN の場合で、BULK_OUT もほぼ同様です。

- ① RPI_CNFG で、VALID を ON, バースト長を設定する。
- ② 受信バッファをメモリに確保して、RPI_BFFR と RPI_DLEN へ指示する。
DLEN には想定されるデータ長か、それ以上を設定する。
- ③ 何か受信すると、PLEN に受信データ長が表示されるので、受信バッファ内を確認する。
COMP 状態をクリアして、②へ戻る。

コアの検証方法

USB3.0 IP デバイス側コアは DesignGateway 製 AB08-USB3HSMC ドータ基板と ALTERA 評価ボードの組み合わせによって実機での動作検証が可能です。(ただし HSMC の I/F 電圧を 2.5V に設定できる評価ボードのみ対応が可能となります。)

推奨される設計スキルに関して

本 IP をユーザ回路上に迅速・確実に実装するために、ALTERA の SOPC Builder および NIOS2EDS についての技術スキルを推奨します。また、高速インタフェース規格に関する一般的な知識を推奨します。更に、ハードウェアデバッグには USB3.0 規格のプロトコル仕様、ソフトウェア開発とデバッグには USB 規格のデバイス仕様に関する知識・理解が必要です。

注文情報

品名	コア/製品型番	実機検証ボード/備考
USB3.0 ホスト側 IP コア Stratix-IV 版	USB3H-IP-S4	DK-DEV-4SGX230N
USB3.0 ホスト側 IP コア Arria-II 版	USB3H-IP-A2	DK-DEV-2AGX125N
USB3.0 ホスト側 IP コア Arria-V 版	USB3H-IP-A5	DK-START-5AGXB3N
USB3.0 ホスト側 IP コア Cyclone-IV 版	USB3H-IP-C4	DK-DEV-4CGX150N
USB3.0 ホスト側 IP コア Cyclone-V 版	USB3H-IP-C5	DK-DEV-5CEA7N
HSMC 対応 USB3.0-IP デモ基板	AB08-USB3HSMC	Altera 標準 HSMC アダプタ基板

本 USB3.0 IP ホスト側コアの価格やライセンス条件等の注文情報については国内 ALTERA 各代理店までお問い合わせください。

履歴

リビジョン	日付	更新内容
1.0	2011/12/27	日本語版の初版作成
1.1	2012/01/05	型番情報の追加
1.2J	2015/04/22	コア内部レジスタを追加
1.3J	2015/05/14	ArriaV GX および Cyclone V E を追加
1.4J	2015/05/24	RPO/RPI で WAIT_ERDY のビット位置[2]->[10], COMP_ERDY のビット位置[1]->[8]の修正

Copyright: 2012 Design Gateway Co.,Ltd.