

SATA AHCI IP コア データ・シート

2017/01/10

Product Specification

Rev1.2



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

- AHCI(Serial ATA Advanced Host Contriller Interface)規格 1.3.1 に準拠
- 各テーブルは FPGA 内部ブロック・メモリに実装することで最小のレイテンシを実現
- 1 コマンド当たり最大 120PRD エントリをサポート
- RAM インターフェイスによるレジスタ制御
- データ・バスは 64bit の AXI4 I/F
- DMA エンジンで 4GByte までのメイン・メモリをサポート
- NCQ コマンドをサポート
- コマンドリスト・テーブル/受信 FIS テーブル/コマンドテーブル用 RAM 内蔵
- 各 FPGA 評価ボードによる AHCI IP リファレンス・デザインの提供が可能
 - CycloneV SX SoC ボード
 - ArriaV ST SoC ボード
 - Arria10 SoC ボード
- SATA AHCI IP コアを複数インスタンスすることにより RAID 等の複数ポートをサポート
- SATA-III(ArriaV ST/Arria10SoC)または SATA-II(CycloneV SX SoC)をサポート

Core Facts	
コアの提供情報	
提供ドキュメント	リファレンス・デザイン説明書 実機デモ手順書
提供形態	暗号化されたネットリスト
検証方法	評価ボードによる実機動作検証
リファレンス デザイン情報	コアのインスタンスは VHDL で記述 QuartusII プロジェクト
その他	以下各評価ボードによる実機検証 CycloneV SX SoC 開発キット ArriaV ST SoC 開発キット Arria10 SoC 開発キット
技術サポート	
デザイン・ゲートウェイによる国内サポート	

Family	Example Device	Fmax (MHz)	Logic utilization (ALMs)	Registers ¹	Block Memory bit	Design Tools
CycloneV SX	5CSXFC6D6F31C6	100	636	1096	666,112	QuartusII 15.1
ArriaV ST	5ASTFD5K3F40I3	150	629	1084	666,112	QuartusII 15.1
Arria10 SX	10AS066N3F40E2SG	150	615	1061	666,112	QuartusII 16.0

表 1: コンパイル結果 (コア単体でのコンパイル)

注:

1) 実際のスライス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

SATA AHCI IP Core

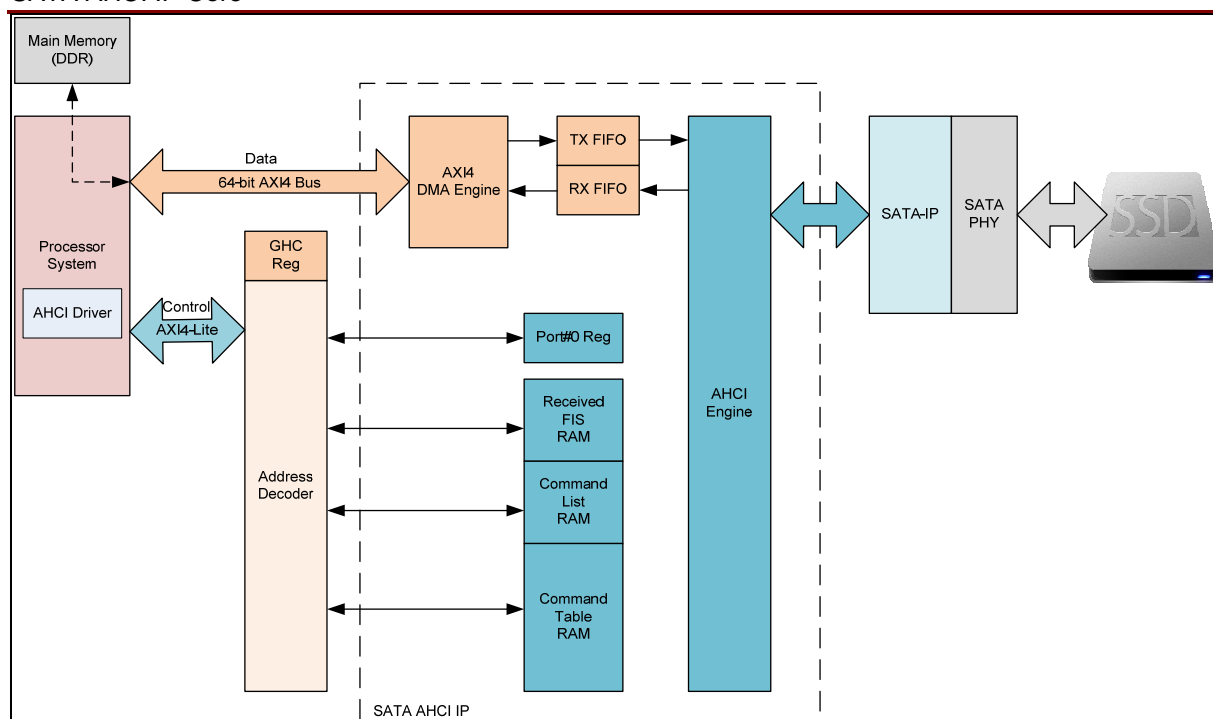


図1 : AHCI IP ブロック図

アプリケーション情報

SATA AHCI IP コアは DesignGateway 社の SATA IP コアと接続し、OS を搭載するプロセッサにて SATA デバイスをシステムのストレージとして活用するアプリケーションに最適のコアです。AHCI IP コアをアクセスする AHCI ドライバを用いることでシステムは SATA デバイスの機能と高いパフォーマンスが得られます。CycloneV/ArriaV/Arria10SoC プラットフォームの ARM コアを AHCI プロセッサとして使うことができるため、本 IP コアは組み込みストレージ・システムや RAID アプリケーションなど高速大容量のデータ収集システムに応用することができます。

概略

SATA AHCI IP コアは SATA デバイスに対してアクセスするプロセッサとして機能しシステム・メモリと SATA デバイス間のデータ転送を行う DMA エンジン機能を内蔵します。リファレンス・デザインでは単一 SATA チャンネルでの実装例となります。ユーザはリファレンス・デザインを編集することで AHCI 規格で最大 32 チャンネルの複数 SATA チャンネルを構築することができます。

AHCI 規格のレジスタは大きくは 2 種類に分類されますが、ひとつはホスト制御でもうひとつはポート制御です。ホスト制御は全てのチャンネルで共通して使える信号です。ホスト制御レジスタはリファレンス・デザインでは HDL フォーマットで提供されユーザは複数チャンネル用に編集可能です。ポート制御は各チャンネルの制御/ステータスとして SATA AHCI IP コア内部で実装されます。

簡単なデザインでは受信 FIS、コマンド・リスト、コマンド・テーブルは外部 DDR のようなメイン・メモリではなく FPGA 内部メモリのブロック RAM でデザインされます。コマンド・リストを使うことでホストは SATA デバイスへのアクセスに 32 キュー深さで NCQ コマンドが使い、非シーケンシャル・アクセスで高いパフォーマンスを維持できます。コマンド・テーブルを使うことで、ホストは一つのコマンドでのデータとして連続した空間を用意する必要がなく、多数のセグメントに分けてデータを用意することができます。この IP コアは 1 コマンドで最大 120 データ・セグメントまでサポートします。

ホストから接続 SATA デバイスへの基本的なアクセス・シーケンスは以下となります。まずホストは Port#0 レジスタをモニタし IP コアとデバイスが新たなコマンドが受け入れられることを確認します。次にコマンド FIS とデータを配置したメモリ・アドレスがコマンド・リストとテーブル RAM に書き込まれます。AHCI エンジンはコマンド FIS を RAM から SATA-IP へと転送し DMA エンジンによりメイン・メモリと SATA-IP コア間でデータが転送されます。データ転送方向はライトあるいはリード・コマンドの種類に依存します。SATA デバイスから受信したステータス・パケットは受信 FIS RAM に格納されます。

ホスト・プロセッサからの本 IP コアへのインターフェイスは 2 種類の信号グループに分類されます、一つは 64 ビット AXI4 マスタ・インターフェイスで DMA データ転送用です、もう一つはレジスタ・アクセス用の 32 ビット・レジスタ・インターフェイスです。データ・ポートはホスト・システムの AXI4 バスと直結できますが一方、レジスタ・インターフェイスは AXI4-Lite バスのスレーブ側で接続するアドレス・デコーダを介します。この回路はリファレンス・デザインにて HDL コードで提供されます。また SATA AHCI IP コアは SATA-IP コアと直結します。

LinuxOS 用の AHCI ドライバは AHCI IP リファレンス・デザインで提供可能です。このドライバは標準のドライバから受信 FIS/コマンド・リスト/コマンド・テーブル用の各メモリ、がメインメモリではなくハードウェア・レジスタ空間へマップするよう編集されています。このドライバを通して SATA デバイスにアクセスすることでアプリケーションを開発することができます。リファレンス・デザインの実行ファイル(bit ファイル等)は IP コア購入前に実機動作を評価できます。

機能ブロックの説明

図 1 に示したように、AHCI IP コアは 3 つのブロックから構成されています、すなわちデータ・インターフェイスとなる①AXI4 DMA エンジン、制御インターフェイスとなる②デコーダおよび RAM、そしてメイン・コントローラの③AHCI エンジンです。

①AXI4 DMA エンジン

AXI4 DMA エンジンは DDR3 メモリとプロセッサ・システム間のバースト・データ転送を実行するためにデザインされ、IP コア内部に送信/受信 FIFO が実装されています。SATA デバイスとの転送データは常にセクタ単位(512 バイト単位)で転送する必要があるため、AXI4 DMA エンジンはバースト・サイズを 512 バイトまたは高パフォーマンス用として 2048 バイトのいずれかにセットします。送信/受信 FIFO はデータ・バス幅を 64 ビット(AXI4 バス幅)と 32 ビット(SATA-IP コアのバス幅)で変換します。DMA エンジン内における各トランザクションにてメイン・メモリ・アドレスと総転送サイズは AHCI エンジンのコマンド・テーブル RAM からデコードされます。

②デコーダおよび RAM

AHCI 規格に準拠し 2 種類のレジスタ空間が定義されます、一つは GHC レジスタでもう一つは Port#0 レジスタです。アドレス・デコーダと GHC レジスタは HDL ソースコードで提供されるため、コアのユーザは AHCI 規格では定義されていないレジスタ空間例えば受信 FIS RAM、コマンド・リスト RAM、コマンド・テーブル RAM などを別アドレスに移植することが可能です。IP コアに添付して提供されるリファレンス・デザインにおいては、表 2 に示す 5 つのレジスタ空間がマップされます。複数チャネルをサポートするには、Port#0 と、それ以外の 3 つの RAM をそれぞれ追加した SATA チャネルにデコードするようアドレス・デコーダを編集する必要があります。更に GCH レジスタ内の値も複数チャネルが接続されたことをホスト・プロセッサに示すため編集します。

受信 FIS RAM、コマンド・リスト RAM、コマンド・テーブル RAM の詳細については図 3～図 5 を参照してください。AHCI 規格と比較すると、コマンド・リスト RAM 内のコマンド・テーブル・ベース・アドレス(CTBA)は使われませんがこれはテーブルがメイン・メモリではなく内部ブロック RAM で実装されているためです。よって本 IP コアは 32 ビットのメイン・メモリ(すなわち 4G バイト)をサポートしデータ・ベース・アドレスの上位 32 ビット (DBAU)も使われません。

Address[16:0]	説明
0x00000 - 0x0002B	ジェネリック・ホスト・コントロール・レジスタ。このエリアは AHCI1.3.1 規格の"3.1 Generic Host Control"章に準拠します。
0x0002C - 0x000FF	未使用
0x00100 - 0x0017F	ポート#0 制御レジスタ。このエリアは AHCI1.3.1 規格の"3.3 Port Register"章に準拠します。.
0x00180 - 0x00FFF	ポート#1～#31 制御レジスタ。
0x01000 - 0x07FFF	未使用
0x08000 - 0x080FF	受信 FIS エリア。このエリアは AHCI1.3.1 規格の"4.2.1 Received FIS Structure"章に準拠します。.
0x08100 - 0x08FFF	未使用
0x09000 - 0x093FF	コマンド・リスト構造体エリア。このエリアは AHCI1.3.1 規格の"4.2.2 Command List Structure"章に準拠します。.
0x09400 - 0x0FFFF	未使用
0x10000 - 0x1FFFF	コマンド・テーブル・エリア。このエリアは AHCI1.3.1 規格の"4.2.3 Command Table"章に準拠します。最大 120 エントリをサポートします。

表 2: レジスタ・マップ

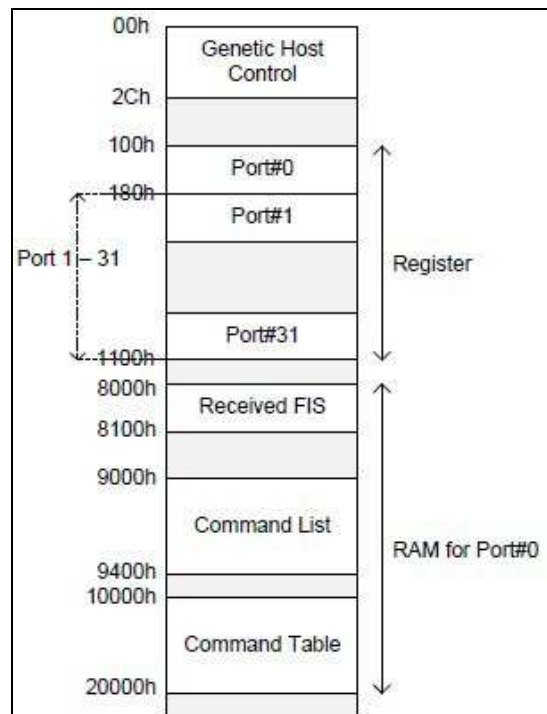


図 2: レジスタのメモリ・マップ

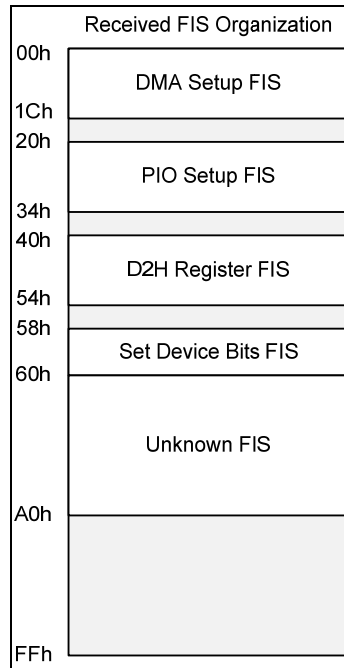


図 3: 受信 FIS のメモリ・マップ

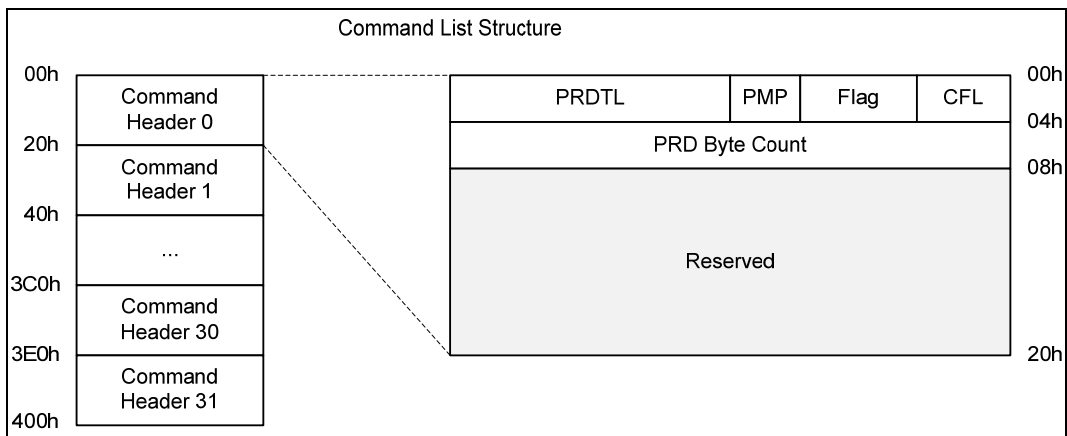


図 4: コマンド・リストのメモリ・マップ

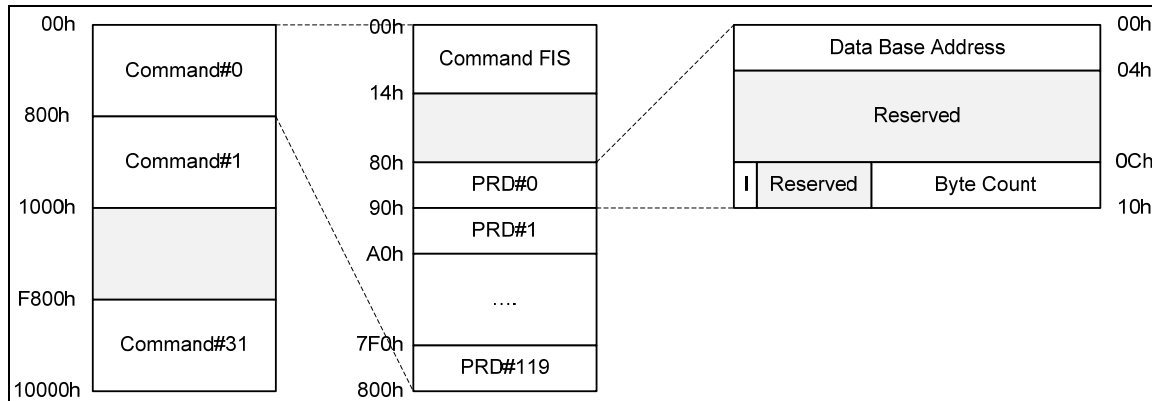


図 5: コマンド・テーブルのメモリ・マップ

③AHCI エンジン

このモジュールはメイン・コントローラであり、ポート#0 レジスタを通してプロセッサからスタート信号を受信し、各 SATA コマンドのシーケンスに沿って SATA-IP コアと共にパケットを送受信します。コマンド・キューをサポートするため、ホストは最大 32 コマンドを SATA デバイスに対して同時に発行でき、デバイス側では受信したコマンドをどの順番で実行するかを選択できます。複数コマンドに対応するため、AHCI エンジンはスロット内でコマンドを発行できるようになった時点でコマンド・テーブル・RAM から SATA-IP に対してコマンドを送信します。同時にこのエンジンは SATA-IP から送られてくる DMA セットアップ FIS を受信 FIS・RAM へ格納する機能を持ちます。タイミングによってはコマンド FIS 送信と DMA セットアップ FIS 受信が同時に起きるといったデータ衝突が発生する場合があります。このような衝突が発生した場合、SATA 規格ではホストよりデバイス側の方が優先されるため、AHCI エンジンはコマンド FIS を自動的に再送します。衝突が発生しなければ AHCI エンジンは、SATA デバイスからの DMA セットアップ FIS で選択された現在アクティブなコマンド・スロットにて、図 5 に示す各 PRD 内のベース・アドレスとバイトカウントをデコードし、AXI4 DMA エンジンに対してメイン・メモリと SATA-IP 間でのデータ転送情報として送信します。データが多数のセグメントに分割されている場合、現在の PRD の終わりに次の PRD でのアドレスと転送長情報が AXI4DMA エンジンに対してロードされます。実行したコマンドにおいて全 PRD カウントや転送長が合致しない場合、割り込みを発生するエラー・フラグがアサートされます。PRD 内の 'I' フラグがセットされていた場合、PRD 転送の最後に割り込み信号が発生します。

プロセッサ・システム

AHCI IP のリファレンス・デザインは CycloneV/ArriaV/Arria10SoC プラットフォーム上で実装されます、このためプロセッサの ARM CPU および UART、タイマー、メイン・メモリ・コントローラなどいくつかの周辺モジュールが機能します。このデザインでは AHCI 動作は CPU のファームウェアによりコントロールされます。

SATA-IP コア

AHCI IP コアと接続する SATA-IP コアは DesignGateway 社から提供されます。SATA-IP コアの詳細については DesignGateway 社の Web ページを参照してください。

コアの I/O 信号

AHCI IP コアの全 I/O 信号については表 3 を参照してください。

信号名	方向	説明
システム信号		
Reset	In	ハードウェア・リセット信号
Clk	In	クロック信号、SATA-III の場合 150MHz かそれ以上、SATA-II の場合 75MHz かそれ以上のクロックとする必要がある
AHCIBusy	Out	AHCI IP コアのビジー状態を示す信号。AHCI がアイドル状態でないとアサート
AHCIInt	Out	割り込み信号。ポート割り込みステータス・レジスタ(P0IS)のいずれかのビットがアサートされそのビットの割り込みがイネーブルであった(P0IE.bit='1')場合にアサート
レジスタおよび RAM インターフェイス		
SIAddr[6:2]	In	Port#0 レジスタ・アドレスを 32 ビット単位でライト/リードするアドレス
SIWrData[31:0]	In	Port#0 レジスタへのライト・データ・バス
SIWrEn[3:0]	In	Port#0 レジスタへのライト・データにおけるバイト・イネーブル信号、ライト動作で SIAddr および SIWrData と同じクロックで出力される
SIRdData[31:0]	Out	Port#0 レジスタからのリード・データ・バス、SIAddr の次クロックで有効値を出力
RxFisMemAddr[7:2]	In	リード/ライト・アクセスにおける受信 FIS 用 RAM アドレスを 32 ビット単位で指定
RxFisMemWrData[31:0]	In	受信 FIS 用 RAM へのライト・データ・バス
RxFisMemWrEn[3:0]	In	受信 FIS 用 RAM へのライト・データにおけるバイト・イネーブル信号、ライト動作で RxFisMemAddr および RxFisMemWrData と同じクロックで出力される
RxFisMemRdData[31:0]	Out	受信 FIS 用 RAM からのリード・データ・バス、RxFisMemAddr の次クロックで有効値を出力
CLstMemAddr[9:2]	In	リード/ライト・アクセスにおけるコマンド・リスト用 RAM アドレスを 32 ビット単位で指定
CLstMemWrData[31:0]	In	コマンド・リスト RAM へのライト・データ・バス
CLstMemWrEn[3:0]	In	コマンド・リスト用 RAM へのライト・データにおけるバイト・イネーブル信号、ライト動作で CLstMemAddr および CLstMemWrData と同じクロックで出力される
CLstMemRdData[31:0]	Out	コマンド・リスト用 RAM からのリード・データ・バス、CLstMemAddr の次クロックで有効値を出力
CTblMemAddr[15:2]	In	リード/ライト・アクセスにおけるコマンド・テーブル用 RAM アドレスを 32 ビット単位で指定
CTblMemWrData[31:0]	In	コマンド・テーブル RAM へのライト・データ・バス
CTblMemWrEn[3:0]	In	コマンド・テーブル用 RAM へのライト・データにおけるバイト・イネーブル信号、ライト動作で CTblMemAddr および CTblMemWrData と同じクロックで出力される
CTblMemRdData[31:0]	Out	コマンド・テーブル用 RAM からのリード・データ・バス、CTblMemAddr の次クロックで有効値を出力

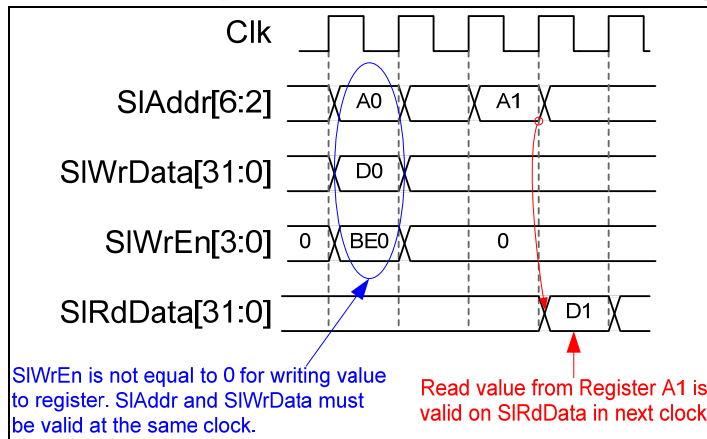
表 3: コアの I/O 信号

信号名	方向	説明
AXI4 インターフェイス(マスター側)		
M_AXI_araddr[31:0]	Out	リード・アドレス・バス、要求されたリード転送の先頭アドレスとして使われる。
M_AXI_arlen[7:0]	Out	リード・アドレス・バースト長、要求されたリード転送長を"データ・ビート長-1"で示す。
M_AXI_arready	In	リード・アドレス・レディ、ターゲットはリード・アドレス受信レディ状態であることを示す。
M_AXI_arvalid	Out	リード・アドレス有効、M_AXI_araddr が有効であることを示す。
M_AXI_awaddr[31:0]	Out	ライト・アドレス・バス、要求されたライト転送の先頭アドレスとして使われる。
M_AXI_awlen[7:0]	Out	ライト・アドレス・バースト長、要求されたリード転送長を"データ・ビート長-1"で示す。
M_AXI_awready	In	ライト・アドレス・レディ、ターゲットはライト・アドレス受信レディ状態であることを示す。
M_AXI_awvalid	Out	ライト・アドレス有効、M_AXI_awaddr が有効であることを示す。
M_AXI_bvalid	In	ライト応答有効、M_AXI_bresp(ライト応答信号)が有効であることを示す。
M_AXI_rdata[63:0]	In	リード・データ・バス、リード動作要求に対するリード・データ・バス
M_AXI_rlast	In	最終リード・データ、バースト転送の最終データ・ビートであることを示す。
M_AXI_rready	Out	リード・データ・レディ、IP コアがリード・データの受信が可能であることを示す。
M_AXI_rvalid	In	リード・データ有効、M_AXI_rdata(リード・データ・バス)が有効であることを示す。
M_AXI_wdata[63:0]	Out	ライト・データ・バス
M_AXI_wlast	Out	最終ライト・データ、バースト転送の最終データ・ビートであることを示す。
M_AXI_wready	In	ライト・データ・レディ、送信相手はライト・データの受信が可能であることを示す。
M_AXI_wvalid	Out	ライト・データ有効、M_AXI_wdata(ライト・データ・バス)が有効であることを示す。
SATA-IP インターフェイス (SATA-IP コアと直結)		
SataRstB	Out	SATA-IP コアへのリセット出力、アクティブ Low
trn_clk	Out	SATA-IP コアへのクロック出力
trn_td[31:0]	Out	32ビット送信データ・バス
trn_teof_n	Out	送信の最終フレーム、送信 SATA FIS パケットの末尾を示す、アクティブ Low
trn_tsrc_rdy_n	Out	送信元レディ、trn_td が有効であることを示す、アクティブ Low
trn_tsrc_dsc_n	Out	本 AHCI IP コアからの送信中断要求、アクティブ Low
trn_tdst_rdy_n	In	送信レディ、SATA-IP コアが送信データを受け取れることを示す、アクティブ Low
trn_tdst_dsc_n	In	SATA-IP コアからの送信中断、アクティブ Low
trn_rd[31:0]	In	32ビット受信データ・バス
trn_rsof_n	In	受信の先頭フレーム、受信 SATA FIS パケットの先頭を示す、アクティブ Low
trn_reof_n	In	受信の最終フレーム、受信 SATA FIS パケットの末尾を示す、アクティブ Low
trn_rsrc_rdy_n	In	受信元レディ、trn_rd が有効であることを示す、アクティブ Low
trn_rsrc_dsc_n	In	SATA-IP コアからの受信中断、アクティブ Low
trn_rdst_rdy_n	Out	受信レディ、本 AHCI IP コアが受信データを受け取れることを示す、アクティブ Low
trn_rdst_dsc_n	Out	本 AHCI IP コアからの受信中断要求、アクティブ Low
SATA PHY インターフェイス (SATA 物理レイヤと接続)		
GEN3	In	SATA 速度信号、'0':SATA2(3.0Gbps)、'1':SATA3(6.0Gbps)
LINKUP	In	SATA PHY のリンクアップ信号、SATA デバイスとリンクが確立したことを示す。
COMINIT	In	PHY からの COMINIT 検出信号、SATA デバイスが新たに接続されたことを示す。
COMWAKE	In	PHY からの COMWAKE 検出信号、OOB 初期化フェーズが完了したことを示す。

表 3: I/O 信号 (続き)

I/O 信号のタイミングについて

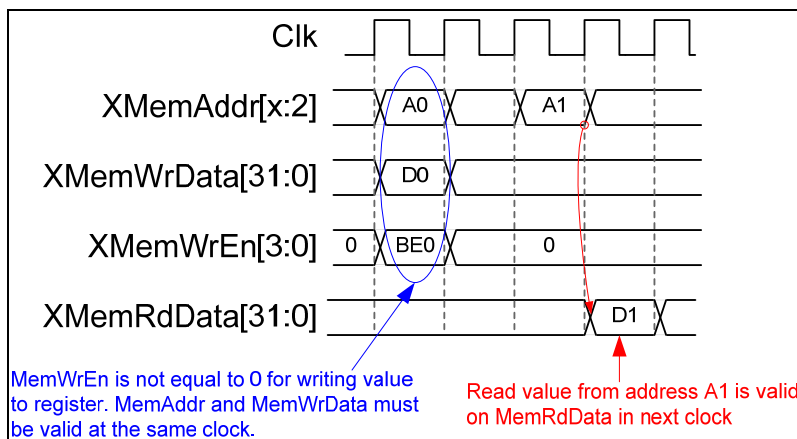
IP コア内の Port#0 レジスタやメモリに対するアクセスは図 6～図 7 のタイミングで行います。



レジスタへのライト時 SIWrEn は非ゼロ値(バイト・イネーブルがアサート)となります、このとき SIAddr と SIWrData は同じクロック期間で有効値をセットする必要があります。

レジスタ A1 の読み出し値は次クロック期間で有効値が SIRdData に出力されます。

図 6: Port#0 レジスタのアクセス・タイミング波形



メモリ(XMem)へのライト時 XMemWrEn は非ゼロ値(バイト・イネーブルがアサート)となります、このとき XMemAddr と XMemWrData は同じクロック期間で有効値をセットする必要があります。

メモリ(XMem)A1 の読み出し値は次クロック期間で有効値が XMemRdData に出力されます。

図 7: 受信 FIS/コマンド・リスト/コマンド・テーブル RAM のアクセス・タイミング波形

M_AXI ポートのタイミング詳細については ARM AXI バスの仕様書を参照してください。(本 AHCI IP コアの AXI バス・インターフェイスは AXI バス規格に準拠します。) 参照 URL: <http://www.amba.com>
SATA-IP のインターフェイス信号の詳細についても SATA-IP コアの仕様書を参照してください。

コアの検証方法

本 AHCI IP コアは機能シミュレーションにより検証可能です。また CycloneV/ArriaV/Arria10SoC 開発キットを使っての実機での動作検証を可能とするリファレンス・デザイン・プロジェクトもコア製品に同梱されます。ドキュメントで示されていない細かい信号タイミング等については、リファレンス・デザインにて SignalTAP を挿入して実機動作させることで、実波形を観測・確認することが可能です。

必要とされる環境と設計スキルに関して

本 AHCI IP コアをユーザ・アプリケーションのシステムに組み込むためには、QuartusII および QSys ツールでのデザイン知識・経験を推奨します。

注文情報

本製品の価格、ライセンス条件、カスタマイズ等についてはデザイン・ゲートウェイ (sales@dgway.com) または国内Altera(Intel) 各代理店までお問い合わせください。

更新履歴

リビジョン	日付	更新内容
1.0	2016/1/8	英語版の初期版リリース
1.1	2016/3/1	ArriaV ST SoC 開発キットのサポート開始
1.2	2016/7/27	Arria10SoC 開発キットのサポート開始
1.2J	2017/01/09	リビジョン 1.2 の日本語翻訳版