

AHCI PCIe SSD-IP (APS-IP) リファレンス・デザイン説明書

Rev1.1J 2016/03/03

本ドキュメントはAHCI PCIe SSD-IP (APS-IP)デモのリファレンス・デザインに関する説明書となります。リファレンス・デザインを実装した APS-IP の具体的なデモ手順については以下のドキュメントを参照してください。

文書名: AHCI PCIe SSD-IP (APS-IP)デモ手順書

ファイル名: dg_apsip_instruction_jp.pdf

1. 概要

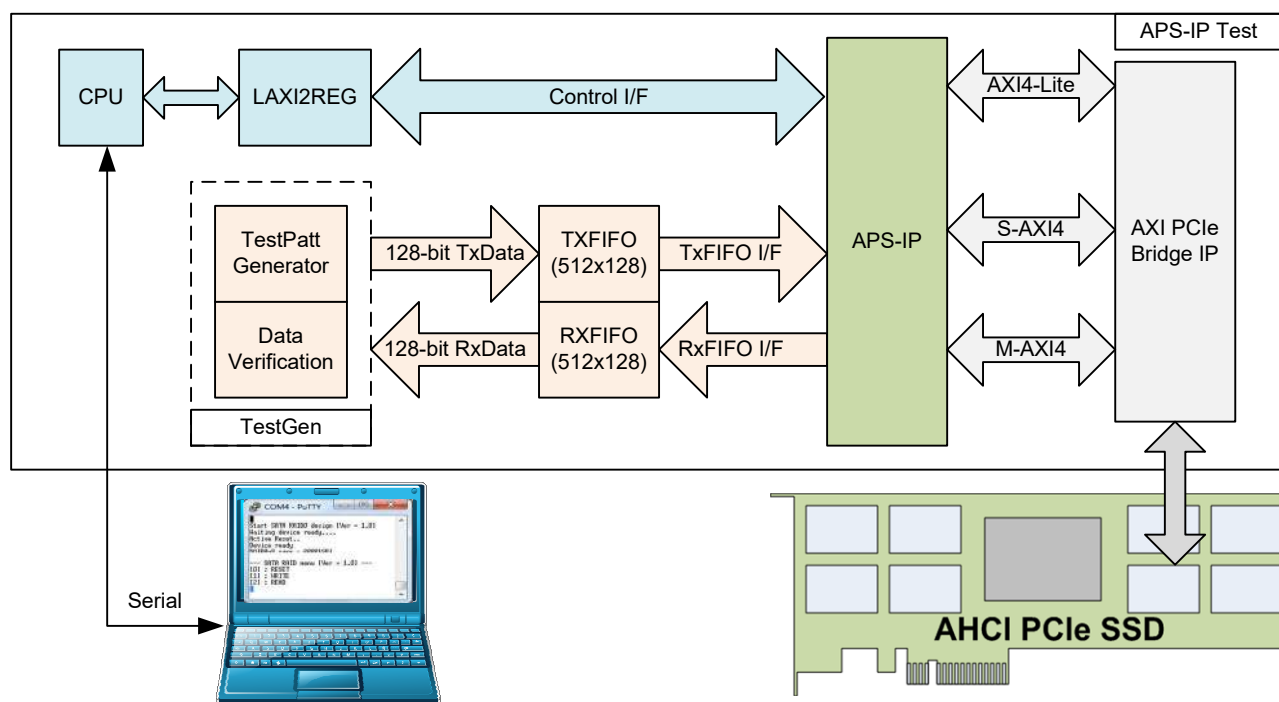


図 1: APS-IP リファレンス・デザインのブロック図

このリファレンス・デザインでは APS-IP コアにシンプルなロジックを加えて AHCI PCIe SSD に対して高速でライト・リードを実行するシステムを構築しています。デザイン内では CPU が使われていますがそれは主にシリアル・コンソールのユーザ・インターフェイスを実装するためのものです。このシンプルなテスト・システムにてユーザはリード・ライトのアクセス先セクタ・アドレス、転送長、コマンドをキーボードから入力し、その入力パラメータは APS-IP コアのインターフェイス信号に変換されます。指示動作が完了すると、CPU は計測したコマンド所要時間と総転送データ量から SSD のライト/リード・パフォーマンスを計算しコンソール上に表示します。

CPU バスと接続するため LAXI2REG モジュールによりアドレスがデコードされ CPU からのデータが APS-IP コアの制御/ステータス信号と接続されます。一方 APS-IP コアのデータ・ポートは外部 FIFO と接続します。ライトあるいはリード・ベリファイ用のテスト・データは TestGen モジュールで生成します。デザイン内の全ての回路モジュールは AXI PCIe ブリッジ IP からの内部 PLL 出力クロックをソースとした同一のクロック・ドメインで動作します。このクロックは 4 レーン GEN2 の AHCI PCIe SSD と接続する場合 125MHz です。

APS-IP コアのデータシートのダウンロードや評価用の bit ファイルのリクエストは以下の弊社 Web サイトから可能です。また、本リファレンス・デザインの実機評価でのパフォーマンスは AHCI PCIe SSD に依存します。

http://www.dgway.com/APS-IP_X.html

2. CPU

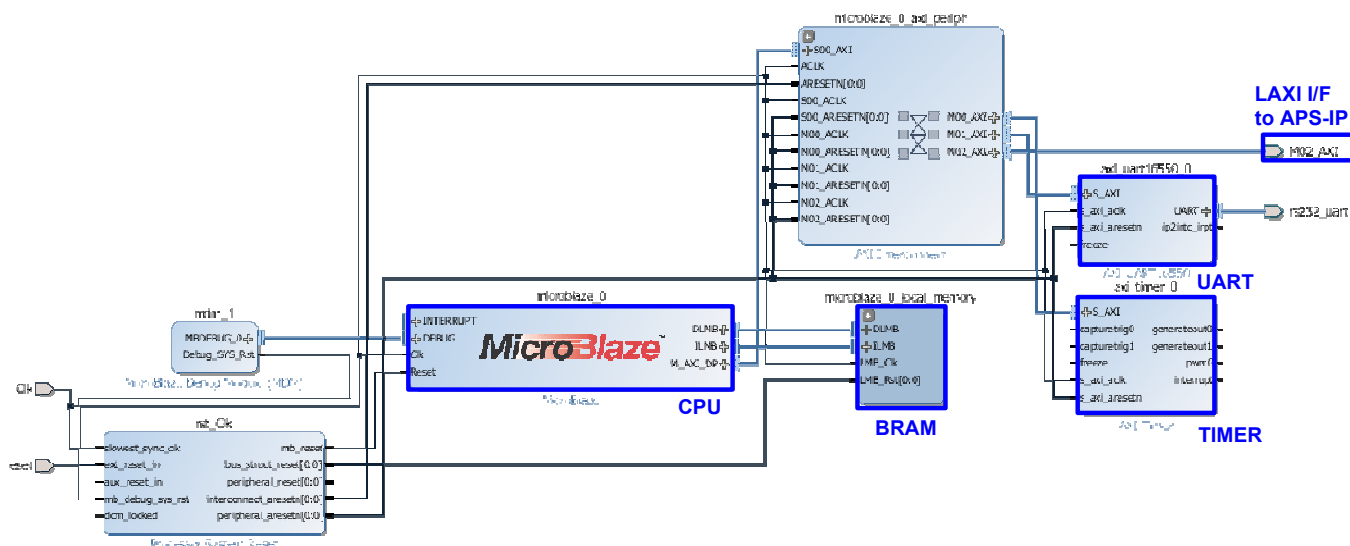


図 2: リファレンス・デザインの CPU システム

リファレンス・デザインにおいて CPU の周辺回路にはユーザ・インターフェイスとして UART、パフォーマンス計測用のタイマー、CPU ファームウェア格納用の BRAM があります。AXI インターコネクトは APS-IP コアを制御/監視するための 32 ビット AXI-Lite バスの外部接続を含みます。AXI-Lite の詳細なメモリ・マップを下表 1 に示します。

表 1: レジスタ・マップ

アドレス Rd/Wr	レジスタ名 (“apsiptest.c”内のラベル名)	説明
BA+0x00 Wr	ユーザ・アドレス(下位)レジスタ (USRADRL_REG)	[31:0]: APS-IP のアクセス先開始アドレス下位 32 ビット UserAddr[31:0]
BA+0x04 Wr	ユーザ・アドレス(上位)レジスタ (USRADRH_REG)	[15:0]: APS-IP のアクセス先開始アドレス上位 16 ビット UserAddr[47:32]
BA+0x08 Wr	ユーザ転送長(下位)レジスタ (USRLENL_REG)	[31:0]: APS-IP の転送セクタ数下位 32 ビット UserLen[31:0]
BA+0x0C Wr	ユーザ転送長(上位)レジスタ (USRLENH_REG)	[15:0]: APS-IP の転送セクタ数上位 16 ビット UserLen[47:32]
BA+0x10 Wr	ユーザ・コマンド・レジスタ (USRCMD_REG)	[1:0]: APS-IP のユーザ・コマンド UserCmd “00”-Identify device, “10”-Write SSD, “11”-Read SSD 本レジスタが書き込まれると APS-IP に対して新たなコマンド実行の要求を発生します。
BA+0x14 Wr	テスト・パターン・レジスタ (PATTSEL_REG)	[1:0]: テスト・パターン選択 “00”-インクリメンタル, “01”-デクリメンタル, “10”-オール 0, “11”-オール 1

アドレス Rd/Wr	レジスタ名 (“apsiptest.c”内のラベル名)	説明
BA+0x100 Rd	ユーザ・ステータス・レジスタ (USRSTS_REG)	[0]: APS-IP のビジー・フラグ (‘0’: アイドル, ‘1’: ビジー) [1]: APS-IP からのエラー出力 (‘0’: 通常, ‘1’: エラー) [2]: データ・ベリファイ・エラー (‘0’: 通常, ‘1’: ベリファイ・エラー発生) [4:3]: APS-IP からの PCIe リンク速度 (“00”: 未リンク状態, “01”: PCIe Gen1, “10”: PCIe Gen2, “11”: PCIe Gen3)
BA+0x104 Rd	総ドライブ容量(下位)レジスタ (LBASIZEL_REG)	[31:0]: APS-IP で報告される総ドライブ容量(単位: セクタ)下位 32 ビット LBASize[31:0]
BA+0x108 Rd	総ドライブ容量(上位)レジスタ (LBASIZEH_REG)	[15:0]: APS-IP で報告される総ドライブ容量(単位: セクタ)上位 16 ビット LBASize[47:32]
BA+0x10C Rd	ユーザ・エラー・タイプ・レジスタ (USRERRTYPE_REG)	[31:0]: APS-IP で報告されるユーザ・エラー・ステータス UserErrorType[31:0]
BA+0x110 Rd	PCIe 割込みステータス・レジスタ (PCIeINTSTS_REG)	[31:0]: APS-IP で報告される PCIe 割込みステータス PCIeIntStatus[31:0]
BA+0x114 Rd	ポート割込みステータス・レジスタ (PORTINTSTS_REG)	[31:0]: APS-IP で報告されるポート割込みステータス PortIntStatus[31:0]
BA+0x120 Rd	比較エラー・アドレス(下位)レジスタ (RDFAILNOL_REG)	[31:0]: リード(ベリファイ)にてデータ比較エラーが発生したアドレス(バイト単位)の 下位 32 ビット[31:0]
BA+0x124 Rd	比較エラー・アドレス(上位)レジスタ (RDFAILNOH_REG)	[23:0]: リード(ベリファイ)にてデータ比較エラーが発生したアドレス(バイト単位)の 上位 24 ビット[56:32]
BA+0x130 Rd	期待値ワード 0 レジスタ (EXPPATW0_REG)	[31:0]: リード(ベリファイ)での期待値データ・ワード 0 [31:0]
BA+0x134 Rd	期待値ワード 1 レジスタ (EXPPATW1_REG)	[31:0]: リード(ベリファイ)での期待値データ・ワード 1 [63:32]
BA+0x138 Rd	期待値ワード 2 レジスタ (EXPPATW2_REG)	[31:0]: リード(ベリファイ)での期待値データ・ワード 2 [95:64]
BA+0x13C Rd	期待値ワード 3 レジスタ (EXPPATW3_REG)	[31:0]: リード(ベリファイ)での期待値データ・ワード 3 [127:96]
BA+0x140 Rd	実リード値ワード 0 レジスタ (RDPATW0_REG)	[31:0]: リード(ベリファイ)での実リード値データ・ワード 0 [31:0]
BA+0x144 Rd	実リード値ワード 1 レジスタ (RDPATW1_REG)	[31:0]: リード(ベリファイ)での実リード値データ・ワード 1 [63:32]
BA+0x148 Rd	実リード値ワード 2 レジスタ (RDPATW2_REG)	[31:0]: リード(ベリファイ)での実リード値データ・ワード 2 [95:64]
BA+0x14C Rd	実リード値ワード 3 レジスタ (RDPATW3_REG)	[31:0]: リード(ベリファイ)での実リード値データ・ワード 3 [127:96]

本デザインでの CPU ファーム・ウェア動作は以下のシーケンスとなります

- シリアル・コンソールからのユーザ指示に従い、IDENTIFY DEVICE か WRITE か READ を実行します

IDENTIFY DEVICE コマンドの場合

- 1) USRCMD_REG="00"をセットします。するとテスト・ロジックはコマンドを生成し APS-IP に対してコマンドを支持します。ビジー・フラグ(USRSTS_REG[0])は'0'から'1'に変化します。
- 2) CPU は USRSTS_REG の値をモニタしコマンドが完了するか又は何らかのエラーが発生するかを確認します。コマンドが完了した場合 BIT[0]は'0'にネゲートし、何らかのエラーが検出された場合 BIT[1]が'1'にアサートします。エラー発生を検出した場合、エラー・メッセージを表示します。
- 3) 本コマンド実行結果として LBASIZEL/H_REG レジスタの SSD 全容量情報を表示します。

WRITE/READ コマンドの場合

- 1) アクセス先開始アドレス、転送セクタ数、テスト・パターンをシリアル・コンソールから受信します。ここで無効なパラメータが入力された場合、コマンド動作はキャンセルされます。
- 2) 入力された各パラメータを USRADRL/H_REG, USRLENL/H_REG, USRCMD_REG (USRCMD_REG のセット値はライトの場合"10"でリードの場合"11")にセットします。
- 3) IDENTIFY DEVICE コマンドのステップ 2) と同様にコマンド完了かエラー発生を確認します、ただしリード・コマンドの場合は USRSTS_REG[2]もチェックしベリファイ・エラーが発生していないかを確認します。
- 4) コマンド実行中はダミーの数字を每秒表示しコマンドが進行中であることを示します。コマンドが完了すると計算したパフォーマンス結果を表示します。

3. LAXI2REG

APS-IP コアの入出力信号と TestGen モジュールへのパラメータは本モジュール内にて CPU バスのレジスタ・アドレスにマッピングされます。AXI-Lite バスからのライト・データおよびアドレスはデコードされシステムの入力パラメータとして変換されます。APX-IP や TestGen モジュールからのステータス信号は本モジュール内のデータ・マルチプレクサにマップされ有効信号と合わせて AXI-Lite バスに戻されます。

4. TestGen

このモジュールの動作は2種類あります、ひとつは WrFf ポートへ出力するテスト・データの生成でユーザがライト・コマンドを指定した場合に動作します、そしてもうひとつは RdFf ポートからの受信データをベリファイしますがそれはユーザがリード・コマンドを指定した場合に動作します。このモジュールの内部ロジック・デザイン詳細を図 3 に示します。

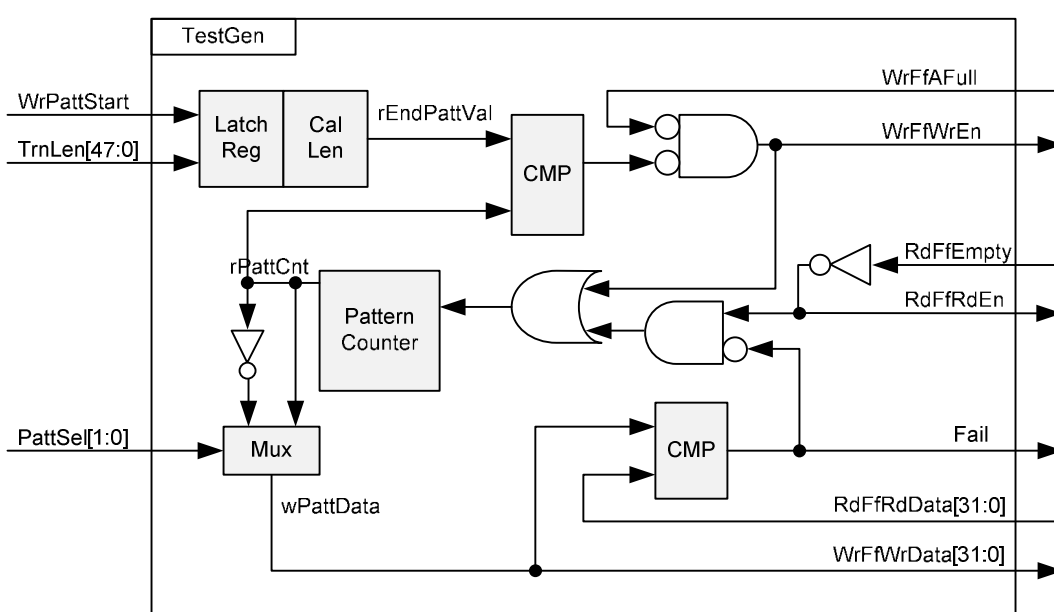


図 3: TestGen モジュール内のロジック・デザイン

ライト転送においてテスト・パターンは、WrPattStart がアサートされた後にパターン・カウンタ・モジュールで生成されます。WrFfAFull をモニタすることで WrFf に新たなテスト・データを受け入れる空きスペースがあることを確認します。テスト・データ・パターンは FIFO に空き容量がある場合に WrFf に出力され全転送サイズがユーザの指定した数量に達すると停止します。TrnLen は全転送サイズをセクタ単位で指定する入力でテスト・パターンのデータ生成を終了する値を計算するために使います。PattSel 入力でテスト・パターンを 4 種類の中から選択します。

リード転送においては RdFfEmpty をモニタして FIFO 内に有効なデータが格納されていることを確認すると RdFf のリード・イネーブルが生成されます。テスト・データ発生器は RdFfRdData 値とベリファイする期待値を生成します。データの不一致を検出した場合、Fail フラグがアサートされます。

5. 更新履歴

リビジョン	日付	履歴
1.0	23-Sep-15	Initial Release
1.0J	2015/10/22	日本語初期版作成
1.1J	2016/03/03	USRSTS_REG に PCIe リンク速度情報を追加

Copyright: 2015 Design Gateway Co,Ltd.