



pci_b PCIマスタ / ターゲット MegaCoreファンクション

1998年 9 月 ver.1

Data Sheet

機能

- pci_bは32ビット、33MHzのペリフェラル・コンポーネント・インタコネクト (PCI) マスタ / ターゲット・インタフェースを実現するパラメータ化されたMegaCore™ファンクション
- PCI SIG (Special Interest Group) のPCIローカル・バス仕様書 Revision 2.1 のタイミングおよびファンクショナル仕様に完全準拠
- 下記の広範囲なハードウェアとソフトウェアによるテストを実施済み (5 ページの「検証テスト」を参照)
 - FLEX® 10K PCI試作用ボード
 - HP E2925A PCIバス・エクスサイザ / アナライザ
 - 標準的に使用されているインテル社のホスト / PCIブリッジ、DEC社のPCI/PCIブリッジ
- ユーザ規定のロジックの集積化が容易
- ユーザ・シミュレーション用のサンプル・テスト・ベクタを添付
- 幅広い集積度とパッケージのデバイスをサポート ; FLEX 10Kのアーキテクチャに最適化
- リスク・フリーなOpenCore™評価機能により、ライセンスの購入前にMAX+PLUS® IIを使用してデザインへのインスタンス化、コンパイル、シミュレーションが可能
- FLEXデバイスの約1,050個のロジック・エレメント (LE) で構成可能
- マスタとターゲットの独立した動作が可能
- PCIマスタ機能
 - ゼロ・ウェイト・ステートによる無制限サイクル数のメモリ・リード / ライト転送 (最高132Mbytes/sec)
 - メモリ・リード / ライト、コンフィギュレーション・リード / ライト、I/Oリード / ライト、メモリ・リード・マルチプル (MRM)、メモリ・リード・ライン (MRL)、メモリ・ライト・アンド・インバリデート (MWI) を含むほとんどのPCI バス・コマンドの起動
 - ホスト・ブリッジのアプリケーションにおける、マスタによるターゲットのコンフィギュレーションを可能にするセルフ・コンフィギュレーション機能 (pci_b自身のターゲットも含む)
 - パリティ・エラー検出
 - バス・パーキング
- PCIターゲット機能
 - ゼロ・ウェイト・ステートによる無制限サイクル数のメモリ・リード / ライト転送 (最高132Mbytes/sec)
 - タイプ・ゼロのコンフィギュレーション・スペース
 - メモリ・タイプとサイズが変更可能な最大 6 個までのベース・アドレス・レジスタ (BAR) により、高い適合性と効率での実現と柔軟性の高いシステム・メモリの配分が可能

さらに多くの 特長

- コンフィギュレーション・リード/ライト、メモリ・リード/ライト、I/Oリード/ライト、MRM、MRL、MWIを含むほとんどのPCIバス・コマンドに応答；サポートされていないすべてのコマンドはPCI仕様の要求通りに無視
- パリティ・エラー検出機能
- ローカル側の生成したターゲット・アボート、リトライ、ディスコネクトのサポート
- コンフィギュレーション・レジスタ：
 - パラメータ化：デバイスID、ベンダID、クラス・コード、リビジョンID、BAR0-BAR5、サブシステムID、サブシステム・ベンダID、最大レイテンシ、最小グラント
 - 非パラメータ化：コマンド、ステータス、ヘッダ・タイプ、レイテンシ・タイマ、キャッシュ・ライン・サイズ、インタラプト・ピン、インタラプト・ライン

概要

これまで、PCIローカル・バスのアプリケーションは、ロー・エンドからハイ・エンドまでのデスクトップPCが対象となっていました。現在、PCIインタフェースはサーバ、LAN、SCSI、FDDIなどの高帯域I/Oアプリケーションに対する共通の基本ビルディング・ブロックとなっています。pci_b（注文コード：PLSM-PCI/B）はハードウェアによるテストが実施済みとなっており、32ビット、33MHzのPCIマスタ/ターゲット・インタフェースを高い性能と柔軟性で実現することができます。

pci_bはPCIの複雑なプロトコルと厳しいタイミング要求を内部処理できるようになっているため、設計者は付加価値の高いカスタム・ロジック部のデザインに注力することができ、「Time-to-Market」の期間を大幅に短縮することができます。

アルテラのFLEX 10Kデバイスに最適化されているpci_bは、コンフィギュレーション、I/O、メモリの各トランザクションをサポートしています。FLEX 10Kデバイスの提供する高い集積度により、pci_bの機能を実現した後も、残りのデバイス・リソースをローカル側のカスタム・ロジックに使用することが可能です。また、FLEX 10Kデバイスの高い性能により、pci_bはゼロ・ウェイト・ステートによる無制限サイクル数のメモリ・バースト転送をサポートすることができ、32ビット、33MHzのPCIバスにおける理論上の最高性能である132Mbytes/secのピークおよび連続動作スループットを実現します。

pci_bでは、マスタとターゲットのインタフェースがそれぞれ独立に動作することができるため、PCIバスの効率的な使用と最小のレイテンシを実現することができます。例えば、ターゲット・インタフェースがゼロ・ウェイト・ステートでのバースト・ライト・データを受信しているときに、ローカル側のロジックが同時にPCIバスへマスタの権利を要求することができます。遅延を最小に抑えることができます。また、pci_bが提供するローカル・マスタとターゲットとの分離されたデータ・パスにより、それぞれ別々にデータのプリフェッチとポスティングを行うことができます。個々のアプリケーションに応じて、FIFO（First-In First-Out）バッファのような

異なるタイプのメモリやデータ・バッファを多様な深さ、幅、構造でローカル側のロジックに実現することができます。

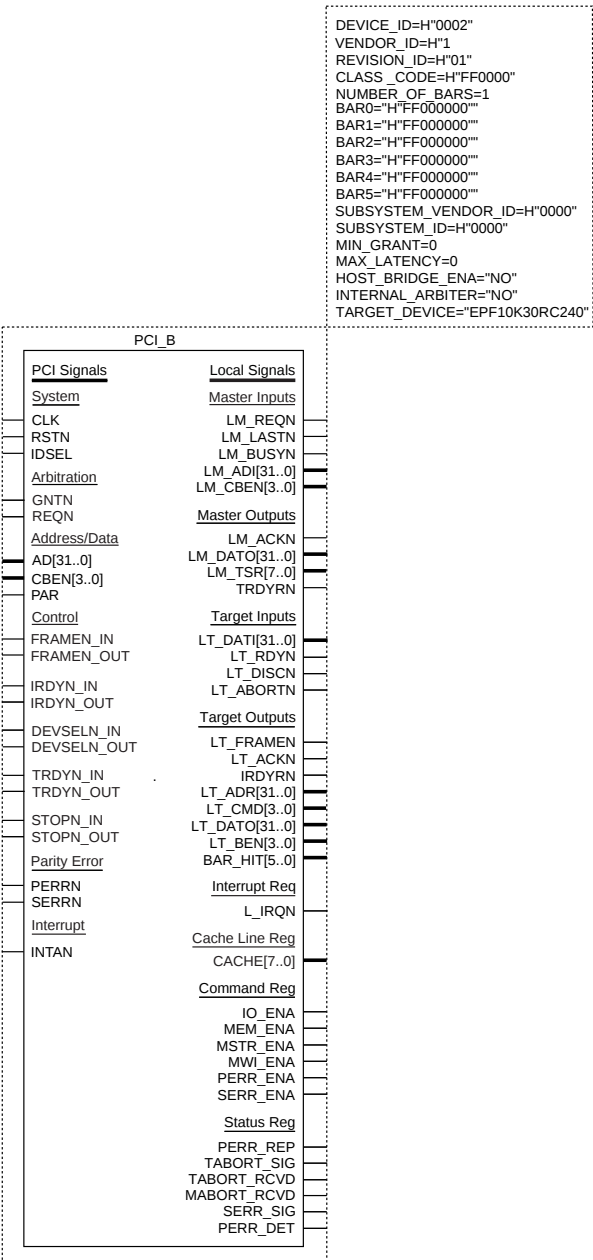
タイミングとプロトコルをPCI仕様に確実に準拠させるため、pci_bにはハードウェアによる厳格な検証テストが実施されています。実施された検証テストの詳細については、5 ページの「検証テスト」をご覧ください。

pci_bの機能はパラメータ化されており、インスタンス化を行うときにコンフィギュレーション・レジスタを変更することができます。これらの機能により、高い拡張性と幅広い適合性、そして効率的なターゲット・デバイスへの実現が提供されます。この結果、pci_bを要求内容の異なる複数のPCIプロジェクトに採用することが可能となっています。

例えば、pci_bは複数のローカル側デバイスに対して最大 6 個までのベース・アドレス・レジスタ (BAR) を提供しています。ただし、アプリケーションによっては、1 つの連続したメモリ領域だけが必要になります。このような場合は、1 個のみのBARをインスタンス化することによって、使用されるLEの数を減少させることができます。パラメータの値が規定されると、MAX+PLUS II のソフトウェアは要求されるロジックをコンパイル時に自動的にまた効率的に実現します。

次ページの図 1 は、MAX+PLUS IIのグラフィック・デザイン・ファイル (.gdf) で使用されるとき pci_bのシンボルを示したものです。

図 1 pci_bのシンボル



検証テスト

pci_bは、PCI SIGのPCIローカル・バス仕様書 *Revision 2.1*とPCI準拠チェックリスト*Revision 2.1*に準拠しています。このpci_bには出荷時にMAX+PLUS IIのシミュレータ・チャネル・ファイル(.scf)が添付されており、このファンクションのMAX+PLUS IIによるシミュレーションに使用できるようになっています。

PCI仕様で要求されるタイミングとプロトコルに対する完全な準拠を提供するため、このpci_bのファンクションには幅広いシミュレーションとハードウェアによる検証テストが実施されています。実施されたシミュレーションには、下記の要求をカバーするための数百サイクルにも及びPCI動作が含まれています。

- PCI SIGのPCI 準拠チェックリスト *Revision 2.1*で要求されるすべてのテスト・シナリオ
- PCI ローカル・バス仕様書 *Revision 2.1*のAppendix Cで規定されている追加の適合ルール
- ローカル側の動作を含むpci_b固有の機能

上記のシミュレーションをベースにした検証に加え、アルテラは、もっとも標準的なPCI / ホスト・ブリッジであるインテル社の430NX、430VX、430TXおよび430HXの各ブリッジを使用して、pci_bの機能に対する厳格なハードウェア・テストも実行しています。また、pci_bには、DEC 21052-ABを含むDEC社の複数のPCI/PCIブリッジによるテストも実行されています。各ハードウェアによるテストは下記の方法で実行されています。

- アルテラのFLEX 10K PCI 試作用ボードにメモリ・インタフェース内蔵のpci_bの機能を実現する。
- HP社のE2925A PCI バス・エクスサイザ / アナライザを使用してpci_bおよびPCIプロトコルを起動し、動作のチェック、およびランタイム・トランザクションにおけるpci_bの動作のモニタを行う。実施されるテストには以下の動作が含まれる。
 - 多様なバースト長でのメモリ・リード / ライト
 - コンフィギュレーション・リード / ライト
 - I/Oリード / ライト
 - インタラプト、ターゲット・アボート、ターゲット・リトライ、ターゲット・ディスコネクト、マスタ・アボートなどのような、アブノーマル・ターミネーション
 - PCI側およびローカル側の双方に対するランダムなウェイト・ステートの挿入

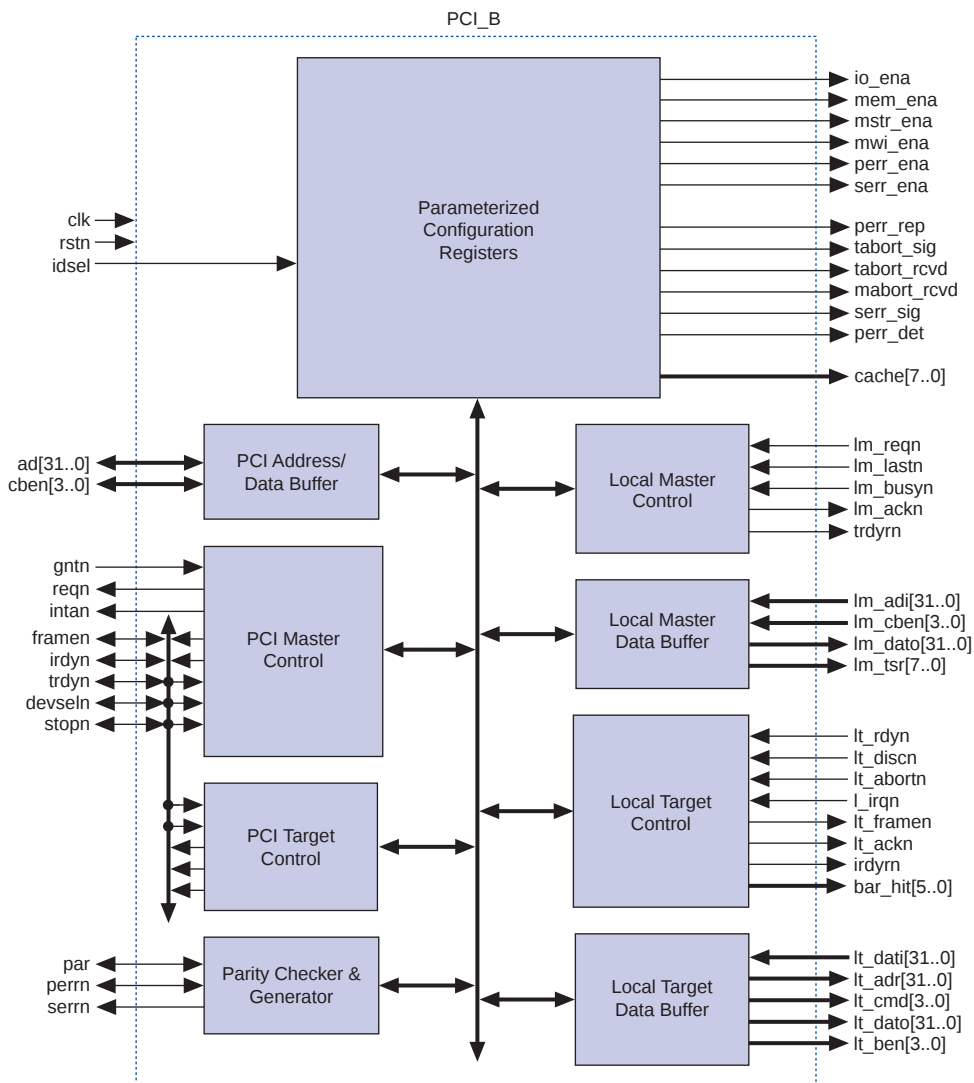
機能の説明

このpci_bは、下記の4つの主要なブロックで構成されています。

- パラメータ化されたPCIバス・コンフィギュレーション・スペース
- ターゲット・インタフェース・コントロール・ロジックとデータ・バス
- マスタ・インタフェース・コントロール・ロジックとデータ・バス
- パリティ・チェッカ / ジェネレータ

図 2 はpci_bをブロック図で示したものです。

図 2 pci_bのブロック図



バス・コマンド

pci_bは、ターゲット・モードにおいて、標準的なメモリ・リード/ライト、キャッシュ・メモリ・リード/ライト、I/Oリード/ライト、コンフィギュレーション・リード/ライトの各コマンドに応答します。また、pci_bはマスタ・モードにおいて、標準的なメモリ・リード/ライト、キャッシュ・メモリ・リード/ライト、I/Oリード/ライト、コンフィギュレーション・リード/ライトの各コマンドを生成することができます。PCIの仕様では、マスタ・インタフェースがMRM、MRLおよびMWIを開始させた場合、マスタが転送されたワード数をトラックし、キャッシュ・ラインのバウンダリ以外でキャッシュ・トランザクションを開始および終了させないことが要求されています。ローカル側のロジックは、この要求が必ず満足されるようになっている必要があります。

表 1 はpci_bが生成または応答できるPCIバス・コマンドを示したものです。PCI ローカル・バス仕様書 Revision 2.1の規定にしたがい、サポートされていないコマンドと予約されているコマンドはpci_bで無視されるようになっています。

表 1 pci_bでサポートされているPCIバス・コマンド			
cbe[n][3..0]の値	バス・コマンド・サイクル	マスタ	ターゲット
0000	インタラプト・アクノレッジ	無視	無視
0001	スペシャル・サイクル	無視	無視
0010	I/Oリード	サポート	サポート
0011	I/Oライト	サポート	サポート
0100	予約	無視	無視
0101	予約	無視	無視
0110	メモリ・リード	サポート	サポート
0111	メモリ・ライト	サポート	サポート
1000	予約	無視	無視
1001	予約	無視	無視
1010	コンフィギュレーション・リード	サポート	サポート
1011	コンフィギュレーション・ライト	サポート	サポート
1100	メモリ・リード・マルチプル (MRM)	サポート	サポート
1101	デュアル・アドレス・サイクル	無視	無視
1110	メモリ・リード・ライン (MRL)	サポート	サポート
1111	メモリ・ライト・アンド・インバリデイト (MWI)	サポート	サポート

ターゲット動作

メモリ・トランザクションに応答する場合、pci_bはゼロ・ウェイト・ステートによる無制限サイクル数のバースト・データ転送をサポートします。このため、メモリ・リードとメモリ・ライトの双方の転送で、32ビット、33MHzのPCIバス・システムで達成可能な最大帯域幅である132Mbytes/secの連続したスループットが実現されます。ただし、ローカル側のロジックが任意のデータ・サイクルでバースト・データを処理できなくなった場合には、1個または複数のウェイト・ステートを挿入し、lt_rdynポートを介してPCIバス上のデータ交換を停止させることができます。低速のアプリケーションに対しては、ローカル側のインタフェース部にデータのプリフェッチまたはポスティングを行うためのFIFOバッファのようなデータ・バッファを設けることが一般的に要求されます。このpci_bには、入力と出力のアドレスおよびデータに独立したバスが提供されています。FLEX 10Kに提供されているユニークなエンベデッド・メモリの高い集積度と柔軟性を活用することによって、ローカル側のロジックに多様なデータ幅と深さのデータ・バッファを簡単に追加することができます。

pci_bは高速のデータ転送に加え、ローカル側から生成されたアブノーマルなターミネーション動作もサポートしています。ほとんどの条件では、データがマスタとターゲットとの間で問題なく転送されますが、ローカル側のロジックがリクエストを完了できない場合には、lt_abortnまたはlt_discn信号を使用して実行中のPCIサイクルを終了させることができます。pci_bは、ローカル側のロジックとPCIバスに対する入力の種類に応じて、devseln、trdyn、stopnの各コントロール信号をドライブして、リトライ、ディスコネクト、またはアボートのいずれかのコマンドを発行して、実行中のトランザクションを終了させます。

マスタ動作

pci_bはPCIマスタとして、無制限サイクル数のゼロ・ウェイト・ステートのバースト・メモリ転送を開始させることができ、132Mbytes/secの帯域幅を連続で達成することができます。ただし、ローカル側のロジックが任意のデータ・サイクルでデータを処理できなくなった場合は、1個または複数のウェイト・ステートを生成し、lm_busrnポートを介してPCIバス上のデータ転送を停止させることができます。また、pci_bは7ページの表1に示されているコンフィギュレーション動作やI/Oトランザクションを開始させることもできます。

pci_bはホスト・ブリッジとしても動作することができます。HOST_BRIDGE_ENAのパラメータを"ON"に設定すると、pci_bはPCIバスに対するホスト・ブリッジとして動作します。pci_bをホスト・ブリッジとして動作させることによって、ローカル側のインテリジェント・ホストがPCIバス上の他のエージェントをコンフィギュレーションできるようになります。この場合、ローカル側のインテリジェント・ホストはバス上に存在するデバイスのスキャン、正確なメモリ・マップの構築、インタラプトやアブノーマル・ターミネーションの処理を行うことができ、pci_b自身のターゲットをコンフィギュレーションすることも可能です。

ほとんどのサイクルでは、データが問題なく転送されますが、エラーが発生した場合は、pci_bがターミネーションの動作を自動的に処理（リトライ、アボート、およびディスコネクト）し、ユーザ指定の動作をそのようなエラーから隔離します。また、pci_bは、ターゲットから要求されていないトランザクションを終了させることもできます。このような状態の場合は、mabort_rcvd信号がドライブされます。

低速のアプリケーションに対しては、一般的にローカル側のインタフェース部にデータ・バッファを設け、データのプリフェッチやポストイングを行えるようにすることが要求されます。pci_bは分離されたデータとアドレスのポートを提供しているため、どのようなタイプのバッファも最適化して実現できるようになっています。

マスタとターゲットのインタフェースはそれぞれ独立して動作させることができます。例えば、ターゲット・インタフェースが lt_dato[31..0]のポートを通じてメモリ・ライト・データをバースト・モードで転送しているときに、ローカル側のロジックが lm_reqnピンを通じてPCIバスのマスタの権利を要求する別の動作を行うことができます。この動作は性能を向上させ、ローカル側のレイテンシを減少させます。

コンフィギュレーション・レジスタ

PCIバスの各論理デバイスには、コンフィギュレーション・レジスタを実現するためにリザーブされている256バイトのブロックが含まれています。最初の16ワード分のDWORDSのフォーマットには、PCI SIGのPCI準拠チェックリスト *Revision 2.1*により、タイプ-1とタイプ-0の2種類のヘッダ・タイプが定義されています。ヘッダ・タイプ-1はPCI-PCIブリッジ用であり、ヘッダ・タイプ-0はpci_bを含む他のすべてのデバイスに使用されます。

表 2 は、定義されている64バイト分のコンフィギュレーション・レジスタ・スペースを示したものです。この範囲内にあるレジスタはデバイスを確定したり、PCIバスの制御、PCIバス・ステータスを表示するときに使用されます。青色で示されている部分は、pci_bがサポートしているレジスタです。

表2 PCIバスのコンフィギュレーション・レジスタ				
アドレス	バイト			
	3	2	1	0
00H	デバイスID		ベンダID	
04H	ステータス・レジスタ		コマンド・レジスタ	
08H	クラス・コード			リビジョンID
0CH	BIST	ヘッダ・タイプ	レイテンシ・タイマ	キャッシュ・ライン・サイズ
10H	ベース・アドレス・レジスタ0			
14H	ベース・アドレス・レジスタ1			
18H	ベース・アドレス・レジスタ2			
1CH	ベース・アドレス・レジスタ3			
20H	ベース・アドレス・レジスタ4			
24H	ベース・アドレス・レジスタ5			
28H	カード・バスCISポインタ			
2CH	サブシステムID		サブシステム・ベンダID	
30H	拡張ROMベース・アドレス・レジスタ			
34H	予約			
38H	予約			
3CH	最大レイテンシ	最小グラント	インタラプト・ピン	インタラプト・ライン

パラメータ

pci_bに提供されているパラメータにより、PCIバス・コンフィギュレーション・レジスタの値を設定できる高い柔軟性が実現されています。INT_ARBITER_ENA、HOST_BRIDGE_ENA、NUMBER_OF_BARS、BAR0からBAR5までのパラメータの値を変更することによって、コンパイル時に実現される機能のロジックは自動的にカスタマイズされます。他のパラメータは、リード・オンリとなっているPCIバス・コンフィギュレーション・レジスタに設定されます。各レジスタの詳細については、9ページの「コンフィギュレーション・レジスタ」を参照してください。表3は、pci_bでサポートされているパラメータをまとめたものです。

表3 pci_bのパラメータ			
パラメータ名	フォーマット	デフォルト値	説明
NUMBER_OF_BARS	10進	1	使用されるBARの数。このパラメータによって、コンパイル時にpci_bにインスタンス化されるBARの数をコントロールする。インスタンス化されるBARは、BAR0から始まるシーケンシャルな順番で設定される。
INT_ARBITER_ENA	記号	"OFF"	内部アービタをイネーブルするパラメータ。これを"ON"に設定すると、reqnとgntnが内部信号になり、pci_bで使用されるI/Oの数が減少する。また、"OFF"に設定した場合は、reqnとgntnがトライ・ステート信号となる。
HOST_BRIDGE_ENA	記号	"OFF"	ホスト・ブリッジ動作をイネーブルするパラメータ。これを"ON"に設定すると、マスタのビットがHighになる。
TARGET_DEVICE	記号	"EPF10K30RC240"	pci_bを実現するターゲット・デバイスを選択するパラメータ
BAR0	16進	H"FF000000"	BAR0
BAR1	16進	H"FF000000"	BAR1
BAR2	16進	H"FF000000"	BAR2
BAR3	16進	H"FF000000"	BAR3
BAR4	16進	H"FF000000"	BAR4
BAR5	16進	H"FF000000"	BAR5
CLASS_CODE	16進	H"FF0000"	クラス・コード・レジスタ
DEVICE_ID	16進	H"0001"	デバイスIDレジスタ
VENDOR_ID	16進	H"1172"	デバイス・ベンダIDレジスタ
REVISION_ID	16進	H"01"	リビジョンIDレジスタ
SUBSYSTEM_ID	16進	H"0000"	サブシステムIDレジスタ
SUBSYSTEM_VENDOR_ID	16進	H"0000"	サブシステム・ベンダIDレジスタ
MIN_GRANT	16進	H"0"	最小グラント・レジスタ
MAX_LATENCY	16進	H"0"	最大レイテンシ・レジスタ

NUMBER_OF_BARSのパラメータにより、コンパイル時にpci_bにインスタンス化されるBARの数が規定されます。NUMBER_OF_BARSのパラメータによっては、BAR0からBAR5までのパラメータがMAX+PLUS IIのソフトウェアによって無視されることとなります。例えば、このNUMBER_OF_BARSに1がセットされると、BAR1からBAR5までのBARが実現されなくなり、BAR1からBAR5に対応したパラメータ値が無視されることとなります。

BAR0からBAR5までのパラメータは下記に示す対応するBARの機能をコントロールします。

- 予約されているアドレス・スペースのタイプ (メモリまたはI/O)
- 予約されているメモリまたはI/Oスペースのサイズ
- メモリ・スペースがプリフェッチ可能か?
- メモリ・スペースを32ビットのアドレス空間の任意の位置に設けることができるか? 1Mバイト以下にマッピングできるか?

例えば、BAR0="HFE000008"に設定された場合は、MSB側の7個の1で7個のレジスタがインスタンス化されます。このパラメータ値から、pci_bが使用するメモリ・スペースのサイズは次のように定義されます。

$$2^{(32-7)} \text{ Bytes} = 132 \text{ Mbytes}$$

また、LSB側の4ビットはプリフェッチ可能なメモリを定義しており、さらに32ビットのアドレス・スペースの任意の位置に設けることが可能であることを示しています。

pci_bの信号

pci_bファンクションで使用されているPCIバス信号を以下に示します。

- 入力 - 標準的な入力専用信号
- 出力 - 標準的な出力専用信号
- 双方向・トライ・ステートの入出力信号
- サステインド・トライ・ステート (STS) - 1度に1つのエージェント (PCIバス上で動作するデバイスやホストなど) によってドライブされる信号。サステインド・トライ・ステート・ピンをLowにドライブしているエージェントは、トライ・ステートにする前にそのピンを1クロックの期間、Highにドライブしなければなりません。このとき、他のエージェントは、前のエージェントがリリースされた1クロック後でないと、サステインド・トライ・ステート信号をドライブすることはできません。
- オープン・ドレイン - 他のエージェントとワイヤードORされた信号。信号ソースとなるエージェントがオープン・ドレイン信号をアサートし、プルアップ・レジスタによりオープン・ドレイン信号がディアサートされます。プルアップ・レジスタがオープン・ドレイン信号をインアクティブにするまでには、2から3サイクルのPCIバス・クロックの時間が必要となります。

PCIバスとのインタフェース信号

表 4 は、pci_bがPCIバスにインタフェースされるとき PCIバス信号を示したものです。

表 4 pci_bをPCIバスに接続するための信号 (1/2)			
信号名	タイプ / 方向	極性	説 明
clk	入力	—	クロック信号。clkの入力はrstnとintan信号を除くすべてのPCIインタフェース信号の基準となる。
rstn	入力	Low	リセット信号。FLEX 10KのPCIインタフェース回路を初期化する信号。rstn入力は、PCIバスのclkエッジに対して非同期でアサートできる。この信号がアクティブになると、PCI出力信号はトライ・ステートとなり、serrnのようなオープン・ドレイン信号はフローティング状態になる。
gntn	入力	Low	グラント信号。この信号はマスタ・デバイスがPCIバスを制御していることを示す。各マスタ・デバイスは、アービタと直接接続されている2本のアービトレーション・ライン (gntnとreqn) を持つ。
reqn	出力	Low	リクエスト信号。この出力はマスタがトランザクションを実行するときに、PCIバスの制御を要求していることをアービタに対して示す信号となる。
ad[31..0]	トライ・ステート / 双方向	—	アドレス / データ・バス信号。ad[31..0] はアドレスとデータが時分割されたバスとなっている。各バス・トランザクションはアドレス・フェーズとそれに続く1つ以上のデータ・フェーズで構成される。データ・フェーズはirdynとtrdynの双方がアサートされるときに発生する。
cben[3..0]	トライ・ステート / 双方向	Low	コマンド / バイト・イネーブル信号バス。cben[3..0]はコマンドとバイト・イネーブル信号が時分割されたバスとなっている。このバスはアドレス・フェーズ期間にコマンドを示し、データ・フェーズの期間にバイト・イネーブルを示す。
par	トライ・ステート / 双方向	—	パリティ信号。parはad[31..0]とcben[3..0]の偶数パリティとなる (ad[31..0]、cben[3..0]とpar信号の1の数が偶数になる)。データ・フェーズに続くクロック・サイクルで、データのパリティがバス上に現れる。
idsel	入力	High	デバイス・セレクトの初期化信号。コンフィギュレーション動作を行うためのチップ・セレクトとして使用される。
framen 注(1)	STS / 双方向	Low	フレーム信号。動作中のバス・マスタがバス・オペレーションの開始と処理中を示す信号。 framenが最初にアサートされたときには、アドレスとコマンド信号がad[31..0]とcben[3..0]に送出される。framen信号はデータ・オペレーションの実行中にアサートされ、トランザクションの終了を示すときにデアサートされる。

表 4 pci_bをPCIバスに接続するための信号 (2/2)

信号名	タイプ / 方向	極性	説 明
irdyn 注(1)	STS / 双方向 マスタ：出力 ターゲット： 入力	Low	イニシエータ・レディ信号。バス・マスタからターゲットに出力される信号で、データ・トランザクションの実行が可能であることを示す。ライト・トランザクションでは、irdynがad[31..0]バス上に有効なデータがあることを示し、リード・トランザクションではirdynはマスタがad[31..0]バス上のデータを取り込み可能であることを示す。
devseln 注(1)	STS / 双方向	Low	デバイス・セレクト信号。ターゲットが自分自身のアドレスをデコードし、トランザクションに同意したことを示すときに、このdevseln信号を出力する。
trdyn 注(1)	STS / 双方向	Low	ターゲット・レディ信号。ターゲットがデータ・トランザクションを実行できることを示す出力。リード動作では、trdynによって、ターゲットがad[31..0]バス上にデータを出力していることが示され、ライト動作ではターゲットがad[31..0]バス上のデータを取り込み可能であることが示される。
stopn 注(1)	STS / 双方向	Low	ストップ信号。ターゲット・デバイスがバス・マスタに対して処理中のトランザクションの停止を要求する信号。このstopn信号とtrdyn、およびdevseln信号との組み合わせにより、ターゲットによって生成されたターミネーションの種類が表示される。
perrn	STS / 双方向	Low	パリティ・エラー信号。パリティ・エラーの発生を表示する信号。この信号はパリティ・エラーが発生したデータ・フェーズの2クロック後またはpar信号に続く1クロック・サイクルに挿入される。
serrn	オープン・ ドレイン / 出力	Low	システム・エラー信号。システム・エラーとアドレス・パリティ・エラーを表示する信号。pci_bは、アドレス・フェーズでパリティ・エラーを検出したときにserrn信号を挿入し、これに応じてPCIコマンド・レジスタ内で要求されるビットがセットされる。
intan	オープン・ ドレイン / 出力	Low	インタラプト-A。ホストに対するアクティブLowのインタラプト信号で、インタラプトが必要なあらゆるシングル・ファンクション・デバイスに使用される必要がある。

注：

- (1) pci_bでは、これらの信号が2つの片方向信号（入力と出力）に分割されています。例えば、framen信号はframen_inとframen_outに分割されています。これらの信号を分割することによって、これらの信号に要求されるセットアップ・タイムに適合していないデバイスの使用が可能になります。

PCIバスとFLEX 10Kデバイスでは、IEEE Std. 1149.1のJTAG(Joint Test Action Group)バウンダリ・スキャン・テスト(BST)を実行することができます。JTAG BSTを実施する場合は、PCIバスのJTAGピンをFLEX 10KデバイスのJTAGピンに接続する必要があります。

表5 は、オプションとして使用できる各JTAG信号の機能を解説したものです。

表5 オプションのIEEE Std. 1149.1-1990 JTAG信号			
信号名	タイプ	極性	説 明
TCK	入力	High	テスト・クロック信号。TCKはテスト・モードおよびテスト・データの入出力に使用されるクロック。
TMS	入力	High	テスト・モード・セレクト信号。TMSはデバイス内のテスト・アクセス・ポート(TAP)コントローラを制御するために使われる信号。
TDI	入力	High	テスト・データ入力。TDIはデバイスへの命令とテスト・データをシフト・インするときに使われる入力信号。
TDO	出力	High	テスト・データ出力。TDOはデバイスへ命令とテスト・データをシフト・アウトするときに使われる出力信号。

ターゲット・ローカル側信号

表6 は、ターゲット・トランザクションにおけるpci_bとローカル側の周辺デバイスを接続するためのpci_bのターゲット・インタフェース信号をまとめたものです。

表6 ローカル側との接続を行うpci_bのターゲット信号 (1/2)			
信号名	タイプ	極性	説 明
lt_dati[31..0]	入力	—	ローカル・データ・ターゲット・バス入力信号。ターゲット・リード・トランザクションでは、このlt_dati[31..0]バスがローカル側のペリフェラル・デバイスによってアクティブにドライブされる。
lt_rdyn	入力	Low	ローカル・ターゲット・レディ信号。ローカル側が、ターゲット・リードのときに有効なデータを出力できること、またはターゲット・ライトのときにデータの受け入れが可能であることを示すとき、このlt_rdynを挿入する。ターゲット・リードの実行中にlt_rdynがディアサートされた場合は、実行中の転送が停止される（ローカル側からウェイト・ステートが挿入される）。また、ターゲット・ライトの期間にlt_rdynをインアクティブにすることによって、PCIバス上へウェイト・ステートを挿入することがpci_bに指示される。バースト転送時にpci_bがウェイト・ステートを挿入するのは、lt_rdynによってローカル側にウェイト・ステートが挿入されるときだけである。
lt_abortn	入力	Low	ローカル・ターゲット・アボート・リクエスト信号。ローカル側のデバイスが致命的なエラーを確認したときや、実行中のトランザクションを完了できないときにこの信号をアサートする。したがって、この信号はPCIマスタに対するターゲット・アボートの発行をpci_bに要求することになる。
lt_discn	入力	Low	ローカル・ターゲット・ディスコネクト・リクエスト信号。lt_discnは、トランザクションの実行中に挿入されたタイミングに応じてリトライやディスコネクトを要求する信号として使用される。

表 6 ローカル側との接続を行うpci_bのターゲット信号 (2/2)

信号名	タイプ	極性	説 明
lt_framen	出力	Low	ローカル・ターゲット・フレーム信号。pci_bがPCIトランザクションを実行しているときに、このlt_framenがアサートされる。この信号はpci_bがdevselnをアサートする1クロック前にアクティブとなり、PCIバス上に最後の転送データが送出された後にリリースされる。
lt_ackn	出力	Low	ローカル・ターゲット・アクノレッジ信号。pci_bはこの信号をアサートして、ターゲット・ライト時に有効なデータが出力されていること、またはターゲット・リード時にデータの受け入れが可能であることを示す。ターゲット・リード実行時に、lt_acknがインアクティブになれば、pci_bがデータを受け入れられない状態であることが示され、ローカル側のロジックは動作の継続を延期させる必要がある。また、ターゲット・ライトで、lt_acknがディアサートされると、実行中の転送が停止される（PCIのマスタからウェイト・ステートが挿入される）。バースト転送時にlt_acknがインアクティブになるのは、PCIバスのマスタがウェイト・ステートを挿入したときだけである。
irdyrn	出力	Low	レジスタを通ったirdyn信号。この信号はPCIのirdyn信号のレジスタ出力で、irdynより1クロック遅れて出力される。
lt_dato[31..0]	出力	—	ローカル・ターゲット・データ・バス出力。ターゲット・ライト・トランザクションの期間に、このlt_dato[31..0]がローカル側のペリフェラル・デバイスによってアクティブにドライブされる。
lt_adr[31..0]	出力	—	ローカル・ターゲット・アドレス・バス。このバスには、ローカル側のデータ・フェーズを実行するターゲット・メモリ・アドレスが与えられる。ローカル側でのデータ転送が完了すると（lt_rdynとlt_acknが同じクロックでアクティブとなった状態）、pci_bはこのlt_adr[31..0]をインクリメントする。
lt_cmd[3..0]	出力	—	ローカル・ターゲット・コマンド・バス。このバスには、要求されているPCIトランザクションのコマンドが出力される。このバスにはcben[3..0]バスと同じエンコーディング方式が使用されている。
lt_ben[3..0]	出力	Low	ローカル・ターゲット・バイト・イネーブル・バス。データ・フェーズの期間にPCIマスタからのバイト・イネーブル・リクエストが、このバスに出力される。
bar_hit[5..0]	出力	High	ベース・アドレス・レジスタ・ヒット・バス。このバスは、PCIアドレスと一致したBARの位置、およびpci_bがトランザクションを要求したことを示すために使用される。bar_hit[5..0]の各ビットは、それぞれのBARに対応しており、bar_hit[0]はBAR0に対応している。bar_hit[5..0]バスのタイミングはlt_framenと同一である。
l_irqn	入力	Low	ローカル側インタラプト・リクエスト信号。ローカル側のペリフェラル・デバイスは、PCIバスにインタラプトを要求するときにこのl_irqnをアサートする。この信号をアサートすることによって、pci_bがこのl_irqnがアサートされている期間に、intan信号をアサートするようになる。

ローカル側とのマスタ・インタフェース信号

表7はマスタ・トランザクション実行時においてローカル側のペリフェラル・デバイスとpci_bを接続するときを使用されるpci_bマスタ・インタフェース信号をまとめたものです。

表7 ローカル側との接続を行うpci_bのマスタ信号 (1/2)			
信号名	タイプ	極性	説明
lm_reqn	入力	Low	ローカル・マスタ・リクエスト信号。ローカル側がマスタ・トランザクションを行うときに、この信号をアサートしてPCIバスのマスタの権利を要求する。ローカル側のデバイスは、このlm_reqnをHighからLowに変化させたと同じクロック・サイクルでPCIバス・アドレスとコマンドを供給する必要がある。ローカル側のデバイスによるマスタ・トランザクションの要求は、lm_reqnのアサートとアドレスとコマンドの送出を1クロック・サイクルの期間行うだけで良い。
lm_lastn	入力	Low	ローカル・マスタ・ラスト信号。ローカル側のアプリケーションがpci_bのマスタ・インタフェースに対して現在実行中のトランザクションの終了を要求するとき、この信号をドライブする。ローカル側がこの信号をアサートした場合は、pci_bのマスタ・インタフェースがすぐにframenをディアサートし、irdynをアサートして最後のデータ・フェーズが開始されたことを示す。この場合、ローカル側はマスタ・トランザクション実行中の任意のタイミングで、lm_lastnを1クロック・サイクルの期間アサートするだけで良い。
lm_busrn	入力	Low	ローカル・マスタ・ビジー信号。ローカル側がpci_bとのバースト転送にウェイト・ステートの挿入を要求するとき、この信号をアサートする。この信号がアサートされると、通常はpci_bがirdynをPCIバス上に出して、ウェイト・ステートを要求するようになる。ローカル側でのデータ転送は、lm_acknがアサートされたときのみ発生する。したがって、lm_busrnがアサートされると、lm_acknはディアサートされることになる。
lm_adi[31..0]	入力	—	ローカル・マスタ・アドレス/データ・バス。ローカル側は、lm_reqnをアサートしてマスタ・トランザクションを要求すると同時に、トランザクション・アドレスをこのバスに送出しなければならない。それ以外の場合は、ローカル側のアプリケーションからライト・トランザクションのデータがこのlm_adi [31..0]バスを通じて転送される。ローカル側のデータ転送は、lm_acknがアサートされたときに完了する。ローカル側がデータの転送を行えない場合は、lm_busrnをアサートしなければならない。この状態になると、lm_acknがディアサートされ、データ転送が行われなかったことが示される。
lm_cben[3..0]	入力	Low	ローカル・マスタ・コマンド/バイト・イネーブル・バス。このバスでは、ローカル側のコマンドとバイト・イネーブル信号が時分割で使用される。ローカル側はlm_reqnをアサートしてマスタ・トランザクションを要求すると同時に、このバスにトランザクション・コマンドを送出しなければならない。それ以外の場合、このバスにはバイト・イネーブル情報が送出される。バースト・トランザクションの実行時に、PCIバス上のデータ転送とローカル側のバイト・イネーブルとの間の同期を維持することができなくなることがある。このため、pci_bは、最初のデータ・フェーズでのみバイト・イネーブル信号を取り込むようにしている。

表7 ローカル側との接続を行うpci_bのマスタ信号 (2/2)

信号名	タイプ	極性	説 明
lm_ackn	出力	Low	ローカル・マスタ・アクノレッジ信号。pci_bのマスタ・インタフェースが、ローカル側のデータ転送が発生したときに、この信号をアサートする。ライト・トランザクションの実行時には、pci_bがローカル側からのデータを内部でラッチしたときに、この信号をアサートする。また、リード・トランザクションの実行時には、pci_bがデータをローカル側に転送したときに、この信号をアサートする。ローカル側がデータの受信/送信を行える状態になっていない場合は、lm_busrstnをアサートする必要がある。このため、マスタ・トランザクションにおいて、lm_busrstnがアサートされた場合、またはPCIのターゲットがtrdynをディアサートした場合は、pci_bがlm_acknをディアサートする。lm_acknの動作は、lt_acknの動作とは異なる。
trdynn	出力	Low	レジスタを通ったtrdyn信号。この信号はtrdyn信号のレジスタ出力で、trdynより1クロック遅れて出力される。
lm_dato[31..0]	出力	—	ローカル・マスタ・データ・バス。マスタ・リード・トランザクションの実行時に、pci_bがこのlm_dato[31..0]バスにデータを送出する。pci_bがlm_acknをアサートすることによって、データ転送が開始される。ローカル側がデータの転送を行えない場合は、lm_busrstnをアサートする必要がある。
lm_tsr[7..0]	出力	—	ローカル・マスタ・トランザクション・ステータス・レジスタ・バス。このバス上の信号でトランザクションの進行状況がローカル・インタフェースに通知される。これらの各ビットの詳細は表8に示されている。

表8は、ローカル・マスタ・トランザクション・ステータス・レジスタの各ビットが表示する内容を説明したものです。

表8 ローカル・マスタ・トランザクション・ステータス・レジスタの各ビット (1/2)

ビット番号	ビットの名称	説 明
0	tsr_req	リクエスト・ビット。このビットは、pci_bがPCIバスのコントロールを要求していることを示す。(reqnをアサートした場合など)
1	tsr_qnt	grant・ビット。pci_bがgntnがアサートされたことを検出したとき、またpci_bがトランザクション・アドレス・フェーズであるとき、このビットがアクティブになる。
2	tsr_dat_xfr	データ転送中を示すビット。pci_bがデータ転送モードになっている期間に、このビットがアクティブとなる。このビットは、アドレス・フェーズ後のアクティブとなり、次のステートが開始されるまで、アクティブのままとなる。
3	tsr_lat_exp	レイテンシ・タイマのカウンタ完了を示すビット。このビットは、レイテンシ・タイマのカウンタが完了し、pci_bがマスタ・トランザクションを終了したことを示す。
4	tsr_ret	リトライ検出ビット。このビットはターゲットがリトライを発行したため、pci_bがマスタ・トランザクションを終了させたことを示す。PCIの仕様では、リトライ中に終了したトランザクションは、その後に再度リトライ動作を行う必要があると規定されている。

表8 ローカル・マスタ・トランザクション・ステータス・レジスタの各ビット (2/2)

ビット番号	ビットの名称	説 明
5	tsr_disc_wod	データ非検出時のディスコネクトを示すビット。このビットは、データを検出できなかったターゲットがディスコネクトを発行したことにより、pci_bがマスタ・トランザクションを中止したことを示す。
6	tsr_disc_wd	データ検出時のディスコネクトを示すビット。このビットは、データを検出しているターゲットがディスコネクトを発行したために、pci_bがマスタ・トランザクションを中止したことを示す。
7	tsr_dat_phase	データ・フェーズ・ビット。このビットは、前のクロック・サイクルでPCIバス上でのデータ転送が問題なく行われたことを示す。ローカル側でPCI側へのデータが正確に何回転送されたかをトレースする場合、この信号を使用すると便利である。

コンフィギュレーション・スペース出力信号

表9 はローカル側のアプリケーションに対して便利となるpci_bのコンフィギュレーション信号とその機能を示したものです。詳細については、*PCI ローカル・バス仕様書 Revision 2.1*を参照してください。

表9 pci_bコンフィギュレーション・スペース出力信号

信号名	極性	説 明
cache[7..0]	—	PCIキャッシュ・ライン・レジスタ。ローカル側のアプリケーションは、MWI またはMRLコマンドを使用する場合、この信号を使用する必要がある。
io_ena	High	I/Oスペース・イネーブル信号。PCIコマンド・レジスタのビット-0
mem_ena	High	メモリ・スペース・イネーブル信号。PCIコマンド・レジスタのビット-1
mstr-_ena	High	マスタ・イネーブル信号。PCIコマンド・レジスタのビット-2
mwi_ena	High	メモリ・ライト・アンド・インバリデート・イネーブル信号。PCIコマンド・レジスタのビット-4
perr_ena	High	パリティ・エラー・レスポンス・イネーブル信号。PCIコマンド・レジスタのビット-6
serr_ena	High	システム・エラー・イネーブル信号。PCIコマンド・レジスタのビット-8
perr_rep	High	マスタ・ライト・トランザクションにおいてperrn信号が検出されたことを示す信号。PCIステータス・レジスタのビット-8
tabort_sig	High	pci_bがターゲット・アボートを通知したことを示す信号。PCIステータス・レジスタのビット-11
tabort_rcvd	High	pci_bがターゲット・アボートを受信したことを示す信号。PCIステータス・レジスタのビット-12
mabort_rcvd	High	pci_bがマスタ・アボートを受信したことを示す信号。PCIステータス・レジスタのビット-13
serr_sig	High	システム・エラーが通知されたことを示す信号。PCIステータス・レジスタのビット-14
perr_det	High	pci_bがデータまたはアドレスのパリティ・エラーを検出したことを示す信号。PCIステータス・レジスタのビット-15

参考資料

このpci_bファンクションに関連した参考資料は以下の通りです。

- *PCI MegaCore Function User Guide*. San Jose, California: Altera Corporation, September 1998.
- *PCI Local Bus Specification. Revision 2.1*. Portland, Oregon: PCI Special Interest Group, June 1995.
- *PCI Compliance Checklist. Revision 2.1*. Portland, Oregon.
- *Altera 1998 Data Book*. San Jose, California: Altera Corporation, January 1998.

Altera、FLEX、MAX、MAX+PLUS、MAX+PLUS II、FLEX 10K、MegaCore、OpenCoreは、Altera Corporationの米国および該当各国におけるtrademarkまたはservice markです。この資料に記載されているその他の製品名などは該当各社のtrademarkです。Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

Copyright " 1998 Altera Corporation. All rights reserved.



ALTERA®

日本アルテラ株式会社

〒163-0436
東京都新宿区西新宿2-1-1
新宿三井ビル私書箱261号
TEL. 03-3340-9480 FAX. 03-3340-9487
<http://www.altera.com/japan/>

本社 **Altera Corporation**

101 Innovation Drive,
San Jose, CA 95134
TEL : (408) 544-7000
<http://www.altera.com>

この資料に記載された内容は予告なく変更されることがあります。最新の情報は、アルテラのウェブ・サイト (<http://www.altera.com>) でご確認ください。この資料はアルテラが発行した英文のデータシートを日本語化したものであり、アルテラが保証する規格、仕様は英文オリジナルのものです。