

はじめに

Virtual Input/Output (VIO) は、内部 FPGA 信号をリアルタイムで監視および駆動できるカスタマイズ可能なコアです。2 種類の入力と 2 種類の出力が使用可能で、どちらも FPGA デザインに合わせてサイズを変更できます。

機能

- ・ 非同期および同期入力ポートを介して仮想 LED とその他のステータス インジケータを提供
- ・ 入力ポートにサンプル間の立ち上がりおよび立ち下がり遷移を検出するアクティビティ検出器あり
- ・ 非同期および同期出力ポートを介して仮想ボタンおよびその他の制御を提供
- ・ 同期出力の場合は、パルス トレイン (デザイン速度で実行される 1 と 0 の 16 サイクルトレイン) を定義可能

VIO コアの詳細は、『ChipScope Pro ソフトウェアおよびコア ユーザー ガイド』を参照してください。

LogiCORE IP に関する情報				
コアの内容				
サポートされるデバイスファミリ ⁽¹⁾	Spartan®-3、Spartan-3E、Spartan-3A、Spartan-3A DSP、Spartan-6、Virtex®-4、Virtex-5、Virtex-6			
使用リソース ⁽²⁾	I/O	LUT	フリップフロップ	ブロック RAM
	0	124	264	0
特別機能	なし			
コアに含有されるもの				
マニュアル	製品仕様			
デザイン ファイル フォーマット	なし			
制約ファイル	なし			
検証	なし			
インスタンスエーション テンプレート	Verilog および VHDL ラッパー			
リファレンス デザイン/アプリケーション ノート	なし			
その他の項目	CDC (Signal Description) ファイル			
デザイン ツール要件				
ザイリンクス インプリメンテーション ツール	ISE® 11.1			
検証	ChipScope™ Pro 11.1			
シミュレーション	シミュレーションでのサポートなし			
合成	XST で合成されたネットリスト			
サポート				
ザイリンクスによるサポートあり				

1. これらの FPGA ファミリの派生デバイスも含む
2. 1つの 8ビット幅バンクを含む Virtex-4 デバイスファミリを仮定して概算しています。

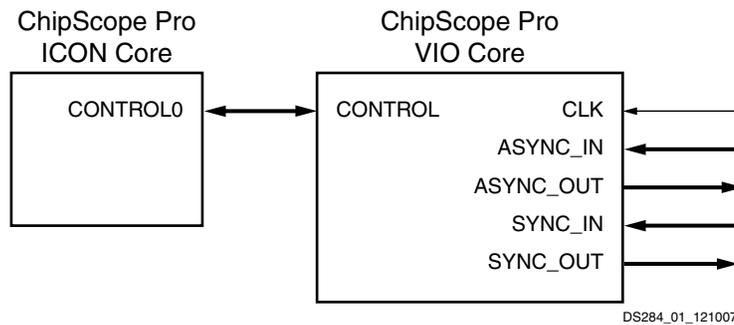
アプリケーション

VIO コアは、ChipScope Pro ソフトウェアおよびコアを使用して検証またはデバッグする必要のあるアプリケーションで使用されるように設計されています。

い、オンチップ RAM やオフチップ RAM は必要ありません。VIO との通信は、[図 1](#)のように、ICON コアを介した JTAG ポートへの接続を使用して行われます。

機能の詳細

VIO コアは、内部 FPGA 信号をリアルタイムで監視および駆動できるカスタマイズ可能なコアです。ILA および IBA コアとは違



DS284_01_121007

図 1 : VIO コアと ICON コアの接続

VIO コアには、次の 4 つの種類の信号があります。

- 非同期入力

JTAG ケーブルから駆動される JTAG クロック信号を使用してサンプリングされます。入力値は定期的に読み戻され、ChipScope Analyzer に表示されます。

- 同期入力

デザイン クロックを使用してサンプリングされます。入力値は定期的に読み戻され、ChipScope Analyzer に表示されます。

- 非同期出力

ChipScope Pro Analyzer でユーザーが定義でき、コアからその周りのデザインへ駆動されます。各非同期出力には、論理値 1 または 0 を定義できます。

- 同期出力

ChipScope Analyzer でユーザーが定義でき、デザイン クロックに同期し、コアからその周りのデザインへ駆動されます。各同期出力には、論理値 1 または 0 を定義できます。それ以外にも、同期出力には 1 と 0 の 16 クロック サイクルも定義できます。

アクティビティ検出器

VIO コア入力には、入力の遷移をキャプチャするためのセルが別にあります。デザイン クロックが ChipScope Pro Analyzer のサンプル周期よりも速いことがほとんどなので、連続するサンプル間で信号の遷移が何度も監視できます。アクティビティ検出器はこ

のビヘイビアを検出し、結果と値を ChipScope Pro Analyzer に表示します。

同期入力の場合は、非同期イベントと同期イベントを監視するアクティビティセルが使用されます。この機能では、グリッチや同期入力信号の同期遷移を検出できます。

パルス トレイン

VIO の同期出力すべてに、スタティック 1、スタティック 0、または連続する値のパルス トレインを出力する機能があります。パルス トレインは、1 と 0 の 16 クロック サイクルのシーケンスで、連続したデザイン クロック サイクルでコアから駆動されます。パルス トレイン シーケンスは、ChipScope Pro Analyzer で定義され、コアに読み込まれた後 1 度だけ実行できます。

VIO インターフェイス ポート

VIO コアの I/O 信号には、ICON への制御バスとインターフェイスポート 4 つとデザイン クロックが含まれます。どのポートも必須ではありませんが、少なくとも 1 つはイネーブルにする必要があります。

表 1 : VIO インターフェイス ポート

ポート名	方向	説明
ASYNC_IN[<m>-1:0]	入力	幅 <m> の非同期入力ポートで、オプションです (enable_asynchronous_input_port による)。このポートはベクタとして宣言する必要があります。1 ビットのポートの場合、ASYNC_IN[0:0] を使用します。
ASYNC_OUT[<m>-1:0]	出力	幅 <i>m</i> の非同期出力ポートで、オプションです (enable_asynchronous_output_port による)。このポートはベクタとして宣言する必要があります。1 ビットのポートの場合、ASYNC_OUT[0:0] を使用します。
CLK	入力	同期入力または出力ポートをレジスタに入れるために使用するデザインクロックで、オプションです (enable_synchronous_input_port または enable_synchronous_output_port による)。
CONTROL[35:0]	入出力	ICON コアの制御バスで、必須です。
SYNC_IN[<m>-1:0]	入力	幅 <i>m</i> の同期入力ポートで、オプションです (enable_synchronous_input_port による)。このポートはベクタとして宣言する必要があります。1 ビットのポートの場合、SYNC_IN[0:0] を使用します。
SYNC_OUT[<m>-1:0]	出力	幅 <i>m</i> の同期出力ポートで、オプションです (enable_synchronous_output_port による)。このポートはベクタとして宣言する必要があります。1 ビットのポートの場合、SYNC_OUT[0:0] を使用します。

VIO の XCO パラメータ

表 2 : VIO の XCO パラメータ

パラメータ名	使用可能な値	デフォルト値	説明
component_name	A-z、0-9、および_(アンダースコア)を含む文字列	vio	インスタンス化されたコンポーネントの名前
enable_asynchronous_input_port	true、false	false	ASYNC_IN ポートをイネーブルにする
enable_asynchronous_output_port	true、false	false	ASYNC_OUT ポートをイネーブルにする
enable_synchronous_input_port	true、false	false	SYNC_IN ポートをイネーブルにする
enable_synchronous_output_port	true、false	false	SYNC_OUT ポートをイネーブルにする
asynchronous_input_port_width	1-256	8	ASYNC_IN ポートの幅
asynchronous_output_port_width	1-256	8	ASYNC_OUT ポートの幅
synchronous_input_port_width	1-256	8	SYNC_IN ポートの幅
synchronous_output_port_width	1-256	8	SYNC_OUT ポートの幅
invert_clock_input	true、false	false	デザイン クロック入力を反転する

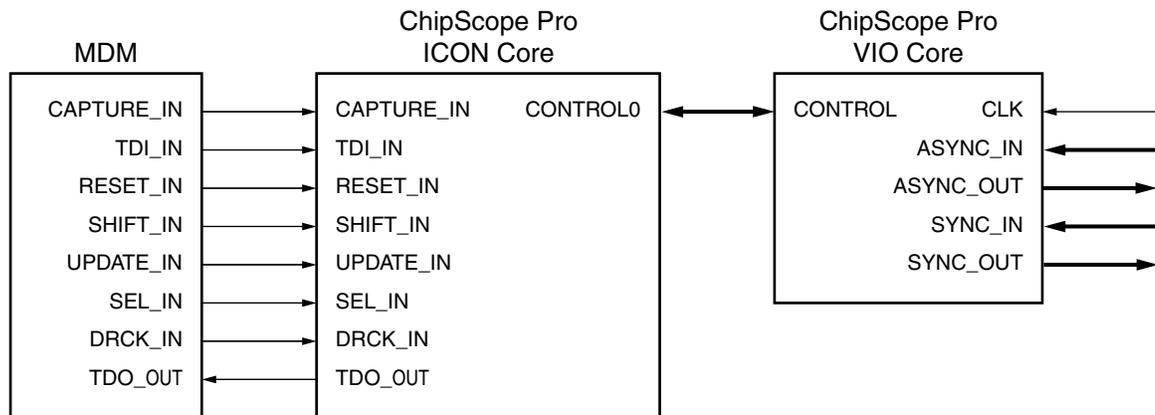
制限

1 つのデザインには、最大で 15 個までの VIO コアが使用できません。

インに含まれる ICON および OPB_MDM コンポーネント インスタンスに依存します。

EDK での VIO コアの使用方法

VIO コアは、EDK を使用してエンベデッド プロセッサ デザインに挿入できます。この場合、[図 2](#)のように、VIO コアは既にデザ



DS284_02_121007

図 2 : EDK デザインの VIO コア コンポーネント

EDK では、VIO コアが Tcl スクリプトを使用してツールに統合されています。EDK の PlatGen ツールが実行されると、Tcl スクリプトが呼び出され、このスクリプトがコマンド ライン モードで CORE Generator を呼び出します。Tcl スクリプトからは CORE Generator にパラメータ ファイル (.xco) が渡され、VIO コアのネットリストが生成されます。また、Tcl スクリプトは [表 3](#) のコア

パラメータに基づいて VIO ポートと一致するように、HDL ラッパーを生成します。

表 3: ILA の EDK 用パラメータ

パラメータ名	使用可能な値	デフォルト値	説明
c_async_input_enable	0、1	0	0 = ポートをディスエーブル、1 = ポートをイネーブル
c_async_input_width	1-256	8	使用される場合は非同期入力ポート幅
c_async_output_enable	0、1	0	0 = ポートをディスエーブル、1 = ポートをイネーブル
c_async_output_width	1-256	8	使用される場合は非同期出力ポート幅
c_family	virtex4、virtex5、 virtex6、virtex6l、 spartan3、 spartan3E、 spartan3A、 spartan3Adsp、 spartan6	なし	使用するデバイス ファミリ
c_rising_clock_edge	0、1	1	使用する入力クロックのエッジ 0 = 立ち下がりエッジ、1 = 立ち上がりエッジ
c_sync_input_enable	0、1	0	0 = ポートをディスエーブル、1 = ポートをイネーブル
c_sync_input_width	1-256	8	使用される場合は同期入力ポート幅
c_sync_output_enable	0、1	0	0 = ポートをディスエーブル、1 = ポートをイネーブル
c_sync_output_width	1-256	8	使用される場合は同期出力ポート幅
c_use_srl16s	0、1	1	0 = SRL16 を使用しない、1 = SRL16 を使用する

VIO コア用に生成された HDL ラッパー ファイルを合成するには、XST 合成ツールが使用されます。XST および ChipScope Pro CORE Generator からの NGC ネットリスト出力は、ザイリンクスの ISE Design Suite に読み込まれ、実際のデバイスのインプリメンテーションに使用されます。

検証

VIO コアは、ザイリンクス社内で開発されたバス ファンクション モデルを使用し、IP テスト環境で検証されています。

参考資料

- ChipScope Pro ソフトウェアとコアの詳細は、<http://japan.xilinx.com/documentation> から『ChipScope Pro ソフトウェアおよびコア ユーザー ガイド』を参照してください。
- EDK での ChipScope Pro を使用したハードウェア検証については、<http://japan.xilinx.com/documentation> から Platform Studio のオンライン ヘルプを参照してください。

- System Generator for DSP での ChipScope Pro を使用したハードウェア検証については、<http://japan.xilinx.com/documentation> から『System Generator for DSP ユーザー ガイド』を参照してください。

サポート

ザイリンクスでは、製品マニュアルに記述されているように、この LogiCORE 製品のテクニカル サポートを提供しています。マニュアルで定義されていないデバイスにインプリメントしたり、製品マニュアルで記述されている範囲を超えてカスタマイズしたり、「DO NOT MODIFY」と記述されているセクションに変更を加えたりした場合、タイミング、機能、製品サポートは保証されません。

注文情報

VIO コアは、ザイリンクス エンド ユーザー ライセンス契約書に基づいて提供されており、ザイリンクスの CORE Generator 11.1 またはそれ以降のバージョンを使用して生成できます。CORE

Generator は、ザイリンクスの ISE Design Suite 開発ソフトウェアに含まれています。

改訂履歴

次の表に、この文書の改訂履歴を示します。

日付	バージョン	変更内容
2008 年 3 月 24 日	1.0	リリース 10.1 用 (初期リリース)
2008 年 9 月 19 日	2.0	リリース 10.1 サービス パック 3
2009 年 4 月 7 日	3.0	リリース 11.1
2009 年 6 月 24 日	3.1	リリース 11.2
2009 年 9 月 16 日	3.1.1	文書を修正。11.3 ソフトウェア リリースで改定。

免責事項

Xilinx is providing this product documentation, hereinafter “Information,” to you “AS IS” with no warranty of any kind, express or implied. Xilinx makes no representation that the Information, or any particular implementation thereof, is free from any claims of infringement. You are responsible for obtaining any rights you may require for any implementation based on the Information. All specifications are subject to change without notice. XILINX EXPRESSLY DISCLAIMS ANY WARRANTY WHATSOEVER WITH RESPECT TO THE ADEQUACY OF THE INFORMATION OR ANY IMPLEMENTATION BASED THEREON, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OR REPRESENTATIONS THAT THIS IMPLEMENTATION IS FREE FROM CLAIMS OF INFRINGEMENT AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Except as stated herein, none of the Information may be copied, reproduced, distributed, republished, downloaded, displayed, posted, or transmitted in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx.

本資料は英語版 (v3.1.1) を翻訳したもので、内容に相違が生じる場合には原文を優先します。

資料によっては英語版の更新に対応していないものがあります。

日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。