XILINX

XAPP587 (v1.0.1) 2012 年 5 月 31 日

7 シリーズ FPGA の BPI 高速コンフィギュレーションおよび iMPACT フラッシュ プログラム

著者 : Stephanie Tapp

概要

このアプリケーション ノートでは、7 シリーズ FPGA をコンフィギュレーションする場合に同期読み出しオプションおよび EMCCLK (外部マスター コンフィギュレーション クロック) オプションを使用する BPI (Byte-wide Peripheral Interface) モードについて説明します。同期読み出しと EMCCLK を使用するこのコンフィギュレーション モードは、FPGA インターフェイスから直接コンフィギュレーションする最も高速なコンフィギュレーション方法であるため、この資料では BPI 高速コンフィギュレーションと呼びます。BPI 高速コンフィギュレーションでは大容量の不揮発性記憶装置を利用し、非同期読み出しを用いる従来の BPI コンフィギュレーションよりもコンフィギュレーション時間を 8 % 短縮できます。FPGA コンフィギュレーションの詳細は、『7 シリーズ FPGA コンフィギュレーション ユーザー ガイド』([参照 1]) を参照してください。このアプリケーション ノートの内容を理解していることが前提です。

7 シリーズ FPGA を介してパラレル NOR フラッシュを間接的にプログラムする場合のハードウェア セットアップ、ファイル生成、およびツール フローについて解説します。具体的には、VC707 評価ボードに搭載されている Virtex®-7 FPGA と 28F00AG18F パラレル NOR フラッシュを使用した ISE® Design Suite 14.1 のフローを説明します。

はじめに

図 1 に、このアプリケーション ノートで説明する 2 つのフローを示します。

- 7 シリーズの BPI 高速コンフィギュレーション プロセス
- iMPACT による FPGA 経由の間接フラッシュ プログラム

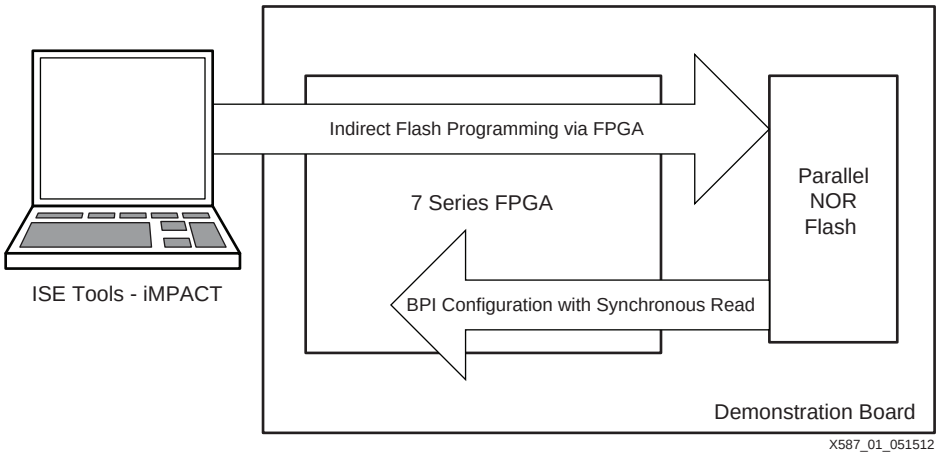


図 1 : BPI 高速コンフィギュレーションおよび iMPACT による間接フラッシュ プログラム フロー

7 シリーズ FPGA は、CCL (CMOS コンフィギャラブル ラッチ) をベースとするデバイスであり、電源投入時にコンフィギュレーション ビットストリームをロードする必要があります。パラレル NOR フラッシュは、幅の広い 16 ビット データ バスによって SPI フラッシュよりも速くコンフィギュレーションできるため、ビットストリームの格納や送信に最適です。厳密な電源投入要件がある PCI Express® などの仕様には、BPI 高速コンフィギュレーションのように短時間でコンフィギュレーションできる方法が必要です。パラレル NOR フラッシュは、高速コンフィギュレーションを可能にするだけでなく、

© Copyright 2012 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.

SPI フラッシュよりも大容量を提供します。ランダム アクセスの不揮発性アプリケーション データの格納にパラレル NOR フラッシュ メモリを使用するシステムでは、コンフィギュレーション データを 1 つのメモリ デバイスに集約できるという利点もあります。

ザイリンクスは、パラレル NOR フラッシュと FPGA 間にある既存のコンフィギュレーション接続を使用して、インシステムでパラレル NOR フラッシュを間接的にプログラムする方法も提供しています。iMPACT プログラミング ツールは、JTAG 経由であらかじめ生成されたビットストリームを FPGA にロードしてコンフィギュレーションします。このビットストリームがプログラミング ケーブルとフラッシュ間のパスを有効にします。ラボ環境では、このソリューションによって、ボードからデバイスを取り外さなくても新しいビットストリームをパラレル NOR フラッシュへ簡単にプログラムできます。

このアプリケーション ノートは、主に次のセクションで構成されています。

- 2 ページの「BPI コンフィギュレーションの基本」
- 2 ページの「パラレル NOR フラッシュ プログラミングの基本」
- 3 ページの「BPI 高速コンフィギュレーションのインプリメンテーション」
- 10 ページの「BPI 高速コンフィギュレーションのファイル生成およびフラッシュ プログラム」
- 21 ページの「チェックリストとデバッグ」

BPI コンフィギュレーションの基本

コンフィギュレーションとは、フラッシュ デバイス、CPLD、またはマイクロプロセッサなどの外部ソースを使用して FPGA へコンフィギュレーション データをダウンロードするプロセスです。BPI コンフィギュレーション モードでは、パラレル NOR フラッシュのアドレス信号、x16 データ バス信号、および制御信号に直接 FPGA を接続して格納されているビットストリームを読み出すことができます。7 シリーズ FPGA の BPI コンフィギュレーション インターフェイスは、2 つのフラッシュ読み出しオプション (非同期または同期) をサポートします。このコンフィギュレーション モードでは、FPGA CCLK (内部生成されたコンフィギュレーション クロック) または EMCCLK を使用してコンフィギュレーション データをダウンロードできます。

同期読み出しおよび EMCCLK を使用する利点

BPI 高速コンフィギュレーションでは、同期読み出しおよび EMCCLK の両オプションを使用します。同期読み出しでは、有効な開始アドレスが与えられた後に最大 100MHz でコンフィギュレーション データをバースト転送できます。したがって、非同期読み出しモードよりもコンフィギュレーション時間が大幅に短縮されます。従来の非同期読み出しの場合、FPGA はフラッシュの読み出しアクセス仕様に準拠するために 10MHz 未満でアドレスをインクリメントしなければなりません。

同期読み出しモードと EMCCLK オプションを併用することで、コンフィギュレーション時間はさらに短縮できます。EMCCLK は、FPGA の内部 CCLK よりも高速かつ正確なクロック周波数オプション可能にする外部オシレーターを活用します。このため、コンフィギュレーション時間をより正確に予測することが可能になります。たとえば、同期読み出しセットアップの最大クロック レートが 100MHz で偏差が 1% の場合は、99MHz の外部クロックを EMCCLK に使用できます。これは、同じセットアップに対して 最大周波数が 66MHz の CCLK よりも利点があるといえます。CCLK の最大周波数は、偏差 ($F_{MCCKTOL}$) を考慮しなければならないため、対象となる範囲が低くなります。たとえば、$F_{MCCKTOL}$ が ±50% の標準的な場合を例に挙げると、広範な 33 ～ 99MHz を考慮しなければなりません。9 ページの「コンフィギュレーション時間」では、コンフィギュレーション時間の予測方法について説明しています。

パラレル NOR フラッシュ プログラミングの基本

パラレル NOR フラッシュ メモリのプログラム用インターフェイスは 1 つですが、このインターフェイスにデータを受け渡す方法は主に 3 つあります。

- サードパーティ プログラマ (オフボード プログラミング)
- エンベデッド プロセッサを使用するインシステム プログラミング (ISP)

- JTAG またはカスタム ソリューションを使用する間接的な ISP

最適なプログラム ソリューションを選択するにあたって考慮すべき主な点は次のとおりです。

- プロトタイピングまたは量産プロダクションの現時段階
- ISP またはオフボード プログラミングの必要性
- 直接または間接プログラムの有効なコネクティビティ

たとえば、量産プロダクション アプリケーションで、フラッシュのプログラム時間を短縮することが最優先される場合には、BP Microsystems などオフボードのサードパーティ プログラマを使用します。オフボードでは、フラッシュへ直接接続することでオーバーヘッドを制限できるため、多くの場合にプログラム時間を短縮できます。さらにこのソリューションでは、プログラムに高い電圧オプションを利用することも可能です。

少量生産のプロトタイピング段階のアプリケーションにおいてオンボードでフラッシュをプログラムする必要がある場合は、iMPACT ツールによる間接的な ISP が最適です。このツールは、反復作業を要するデザインのテストに適しており、特にラボ環境でのデバッグ作業に有用です。ただし、量産プロダクションのプログラムには適していません。

BPI 高速コンフィギュレーションのインプリメンテーション

7 シリーズ FPGA には、サードパーティのパラレル NOR フラッシュへのインターフェイスがあります。7 シリーズの BPI コンフィギュレーション モードは、初期の FPGA ファミリと同様に非同期読み出しをサポートしますが、システム要件の変化に対応するために新しい機能が追加されています。7 シリーズ FPGA は、コンフィギュレーション用にパラレル NOR フラッシュ メモリから同期読み出しを行うビルトイン機能を備えた最初のザイリンクス FPGA です。さらに、EMCCLK オプションを有効にすると、より正確なコンフィギュレーション クロックが使用可能となります。これら 2 つの機能を併用することで、従来のデバイス ファミリと比べてコンフィギュレーション時間を大幅に削減できます。後続の各セクションでは、フラッシュ デバイス選択の基準、インターフェイス、BPI 高速コンフィギュレーション シーケンスについて説明します。

パラレル NOR フラッシュの選択

パラレル NOR フラッシュ デバイスを選択する場合、アプリケーションで必要な格納容量、ボードのペース要件に合うパッケージ タイプ、コンフィギュレーション時間に応じたデータ バス幅、フラッシュの I/O 電圧範囲など、コンフィギュレーション ソースに関するいくつかの要素を考慮する必要があります。必要最小限のフラッシュ容量を判断するためのビットストリーム サイズ情報は、『7 シリーズ FPGA コンフィギュレーション ユーザー ガイド』([参照 1]) を参照してください。

フラッシュは書き込みコマンド シーケンスで同期読み出しモードにする必要があるため、Micron 社製の G18 および P30 フラッシュ ファミリのみサポートされています。Micron G18 ファミリの中でサポートされているのは、Non-MUX インターフェイス デバイスのみです。G18 は P30 ファミリの 2 倍の周波数で動作可能であるため、このアプリケーション ノートでは G18 ファミリに焦点を当てて説明します。表 1 に、サポートされている G18 ファミリ デバイスを示します。

表 1 : BPI 高速コンフィギュレーション パラレル NOR フラッシュ

フラッシュ ファミリ	容量範囲	V_{CC}/V_{CCQ} (I/O)	iMPACT による間接プログラム
Micron G18	128Mb ～ 1Gb	1.8V/1.8V	使用方法の詳細は、iMPACT のソフトウェア マニュアル [参照 2] を参照してください。

電圧の互換性

パラレル NOR フラッシュの I/O は、接続されている FPGA ピンと互換性のある電圧をサポートする必要があります。バンク 0 のコンフィギュレーション専用ピンおよびバンク 14 とバンク 15 の多目的ピンが BPI 高速コンフィギュレーション インターフェイスで使用されます。BPI コンフィギュレーション

インターフェイスへ安定した I/O 電圧を確保するために、これらのピンには同じ V_{CCO} 値が供給される必要があります。さらに、JTAG ピンがバンク 0 に割り当てられ、このバンクの要件に従っています。コンフィギュレーション関連の I/O を 3.3V/2.5V に設定するには、コンフィギュレーション バンクの電圧セレクト ピン (CFGBVS) を High にし、1.8V に設定する場合は Low にします。このアプリケーション ノートでは、バンク 14 とバンク 15 の I/O が 1.8V 以下の規格に制限されている Virtex-7 FPGA について説明します。Virtex-7 FPGA で BPI 高速コンフィギュレーションをサポートするには、CFGBVS ピンを Low に設定します。CFGBVS の詳細は、『7 シリーズ FPGA コンフィギュレーション ユーザー ガイド』([参照 1]) を参照してください。

BPI 高速コンフィギュレーション インターフェイス

図 2 に、BPI 高速コンフィギュレーションのインターフェイスで使用される信号を示します。このモードでのコンフィギュレーション プロセスの詳細は、8 ページの「BPI 高速コンフィギュレーションのシーケンス」を参照してください。

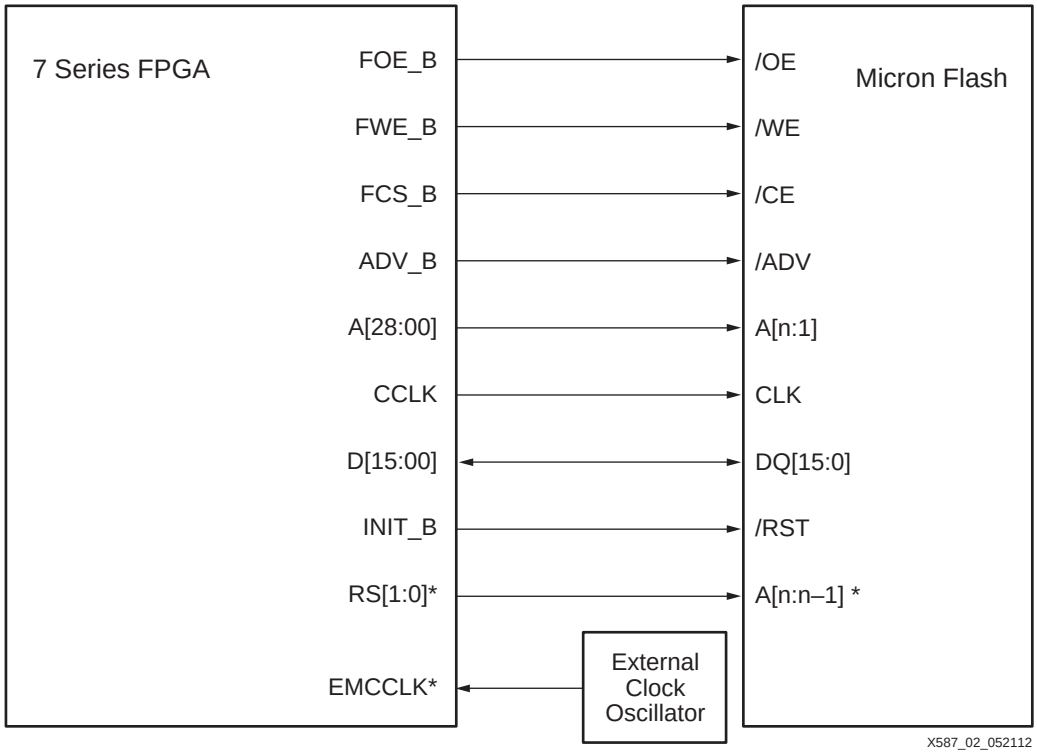


図 2 : BPI 高速コンフィギュレーション インターフェイス

注記：RS[1:0] 信号および EMCCLK 信号は、このアプリケーション ノートで説明している BPI コンフィギュレーション モードを使用する際のオプション接続です。これらについては、このアプリケーション ノートで説明しています。

表 2 に、BPI 高速コンフィギュレーションのインターフェイス信号とその説明を示します。表には、パラレル NOR フラッシュ メモリへの接続も記載されています。

表 2 : 7 シリーズ FPGA BPI コンフィギュレーション信号の説明

7 シリーズ FPGA のピン名	タイプ	専用または多目的	説明	Micron フラッシュの信号接続[1]
CCLK	入力または出力	専用	コンフィギュレーション クロック出力です。JTAG を除くすべてのコンフィギュレーション モードでは、この信号をデフォルトのコンフィギュレーション クロック ソースとして使用します。マスター BPI コンフィギュレーション モードでは、CCLK は出力となります。BPI フラッシュの非同期読み出しモード中はパラレル NOR フラッシュに直接クロックを供給するのではなく、アドレスおよびサンプル読み出しデータを生成するために FPGA 内部で使用されます。BPI フラッシュの同期読み出しモード中は、CCLK をパラレル NOR フラッシュへ直接接続し、データの逐次出力用にクロックを供給します。	CLK (クロック)
DONE	双方向、オープン ドレイン、または アクティブ	専用	コンフィギュレーションの完了を示すアクティブ High の信号です。 0 = FPGA コンフィギュレーション未完了 1 = FPGA コンフィギュレーション完了	NC
INIT_B	双方向	専用	パワーオン リセットまたは PROGRAM_B リセット後、INIT_B は Low 駆動し、FPGA がコンフィギュレーション メモリを初期化 (クリア) 中であることを示します。モード ピンのサンプル前はオープン ドレインのアクティブ Low 入力となり、Low に保持することでコンフィギュレーションの開始を遅らせることができます。 モード ピンのサンプル後は、コンフィギュレーション中の CRC エラー、またはコンフィギュレーション後のリードバック CRC エラー (リードバック CRC が有効の場合) を INIT_B 出力で示します。 0 = CRC エラー /IDCODE エラー (DONE が Low)、またはリードバック CRC エラー (DONE が High でリードバック CRC が有効) 1 = CRC エラーがなし、コンフィギュレーションの初期化完了 (VCCO_0 に外部プルアップ抵抗が必要)	/RST (リセット)
M[2:0]	入力	専用	コンフィギュレーション モードを決定するモード ピンです。 M[2:0] = 010 の場合、マスター BPI モード	NC
PROGRAM_B	入力	専用	アクティブ Low の非同期フルチップ リセットです。	NC
A[28:00]	出力	多目的	アドレス出力です。フラッシュのアドレスをインクリメントするために使用され、同期読み出しモードの場合には、同期モードでフラッシュを設定する読み出しコンフィギュレーション レジスタのコンテンツをフラッシュへ送信します。	A[n:1] (アドレス入力)
ADV_B	出力	多目的	アクティブ Low のアドレス有効出力です。マスター BPI コンフィギュレーション モードの同期読み出しオプションで使用します。同期読み出しモード時に、アドレスが有効であることを伝えるために必要な信号です。非同期読み出しモードの場合は、Low 駆動する必要があります。	/ADV (アドレス有効)

表 2 : 7 シリーズ FPGA BPI コンフィギュレーション信号の説明 (続き)

7 シリーズ FPGA のピン名	タイプ	専用または多目的	説明	Micron フラッシュの信号接続[1]
CSO_B	トライステート出力	多目的	コンフィギュレーション中に使用される、パラレル デイジーチェーンのアクティブ Low チップ セレクト出力です。それ以外の場合は、トライステート出力となります。FPGA が 1 つのアプリケーションでは使用されません。	NC
D[15:00]	双方向	多目的	この 16 ビット データ バスは、FPGA CCLK の立ち上がりエッジでサンプルされます。このバスでデータがフラッシュから読み出され、コンフィギュレーション コントローラーから同期読み出しコマンドが発行された場合には、このバスにフラッシュの読み出しコンフィギュレーション レジスタへの書き込みコマンドが送信されます。	DQ[15:0] (データ)
EMCCLK	入力	多目的	外部マスター コンフィギュレーション クロック入力です。このオプションは BPI 高速コンフィギュレーション モードで使用されます。EMCCLK コマンドがビットストリーム ヘッダーから読み出されると、この入力に外部クロックが供給され、FPGA コンフィギュレーション コントローラーが CCLK (内部コンフィギュレーション クロック) ではなく、このクロックを使用するように切り替わります。EMCCLK を使用する場合、外部オシレーターによって偏差が決定されるため、コンフィギュレーション時間がより正確に予測できます。	NC
FCS_B	出力	多目的	アクティブ Low のフラッシュ チップ セレクト出力です。コンフィギュレーション中はアクティブにトグルされます。	/CE (チップ イネーブル)
FOE_B	出力	多目的	アクティブ Low のフラッシュ出力イネーブルです。コンフィギュレーション中はアクティブにトグルされます。	/OE (出力イネーブル)
FWE_B	出力	多目的	アクティブ Low のフラッシュ書き込みイネーブルです。コンフィギュレーション中はアクティブにトグルされます。	/WE (書き込みイネーブル)
PUDC_B	出力	多目的	コンフィギュレーション中、I/O (バンク 0 の専用 I/O を除く) のプルアップ抵抗を制御します。このピンは外部で終端する必要があります。 0 = コンフィギュレーション中、プルアップ 1 = コンフィギュレーション中、トライステート	NC
RS[1:0]	出力	多目的	リビジョン セレクト ピンです。ビットストリームが複数のアプリケーションでリビジョンを簡単に選択するために使用され、フォールバック機能を提供します。コンフィギュレーション エラーを検出した場合は Low に駆動され、フォールバック ビットストリームが読み込みまれます。リビジョン管理の詳細は、『7 シリーズ FPGA コンフィギュレーション ユーザー ガイド』([参照 1]) を参照してください。	A[n:n–1] (上位アドレスビット)[2]

注記 :

1. フラッシュ信号の説明およびその他の詳細情報は、Micron G18 フラッシュのデータシート [参照 3] を参照してください。
2. リビジョン セレクト ピン (RS[1:0]) は、ビットストリームのリビジョン管理が必要な場合のオプション機能です。

図 3 に、VC707 評価ボードを使用した BPI 高速コンフィギュレーションのインプリメンテーション例を示します。パラレル NOR フラッシュとの接続に必要な信号は、この図に記載されています。iMPACT によるフラッシュの間接プログラムで使用する JTAG ポート接続信号も図中に含まれています。

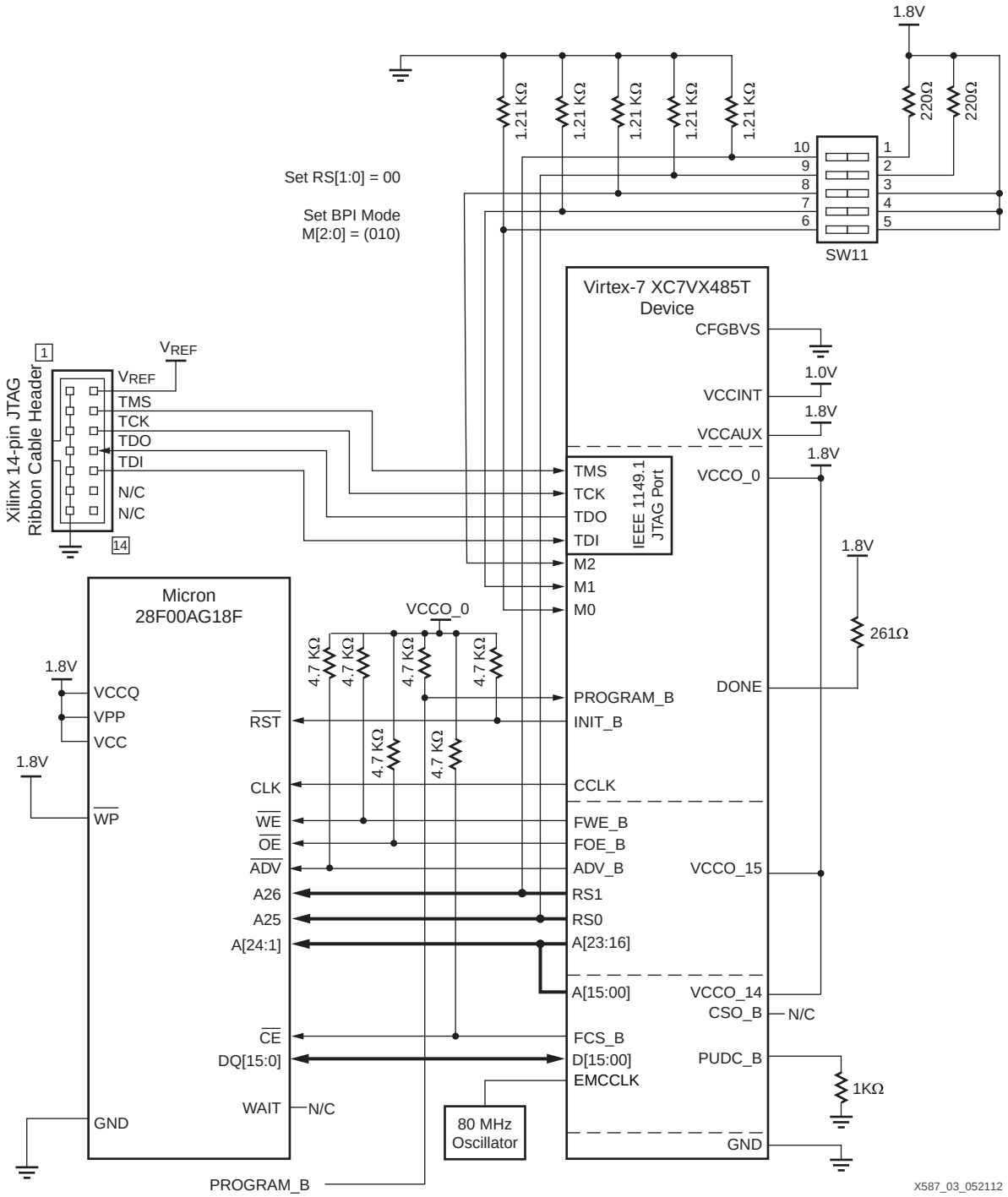


図 3 : VC707 における BPI 高速コンフィギュレーションのセットアップ例

図 3 について説明します。

1. 7 シリーズ FPGA の JTAG 信号 (TCK、TMS、TDI、TCK) は、BPI 高速コンフィギュレーション インターフェイスでは必要ありませんが、iMPACT によるフラッシュの間接プログラムを行う場合に必要です。JTAG インターフェイスは、数多くのアプリケーション セットアップで広く使用されるデバッグ インターフェイスでもあります。フラッシュを間接的にプログラムする場合の手順は、

10 ページの「BPI 高速コンフィギュレーションのファイル生成およびフラッシュ プログラム」を参照してください。

2. EMCCLK の最大周波数は、ターゲットとなるフラッシュおよび FPGA によって異なります。特定のセットアップの最大周波数を確認する場合は、9 ページの「コンフィギュレーション時間」を参照してください。VC707 ボードは、80MHz のクロック ソースを使用します。
3. RS[1:0] は、リビジョン管理用のオプション ピンです。リビジョン管理が不要なアプリケーションでは、適切なアドレス ピンが FPGA の A[28:0] からフラッシュの A[n:1] へ接続され、RS[1:0] は接続されません。
4. 7 シリーズ FPGA の VCCO_0 電源電圧は、パラレル フラッシュの V_{CCQ} と合せる必要があります。
5. フラッシュの間接プログラムおよびコンフィギュレーションを実行するには、Micron 28F00AG18F の Write Protect (/WP) 信号と Wait 信号を適切に接続する必要があります。

BPI 高速コンフィギュレーションのシーケンス

同期読み出しモード用のビットストリームが生成済みで、モード ピン M[2:0] が BPI コンフィギュレーション モードに設定されている場合、このセクションで説明する BPI 高速コンフィギュレーション シーケンスが自動的に実行されます。ビットストリームの生成手順については、10 ページの「BPI 高速コンフィギュレーションのファイル生成およびフラッシュ プログラム」を参照してください。

BPI コンフィギュレーション モードの場合、7 シリーズ FPGA は常に非同期読み出しモードでパラレル NOR フラッシュから読み出しを開始します。FPGA が指定された開始アドレスからアドレス バスをインクリメントすると、パラレル NOR フラッシュがビットストリーム データを返します。その後、7 シリーズ FPGA は、コンフィギュレーション データを読み出す際のフラッシュのモードを決定するためにビットストリーム ヘッダーを読み出します。ビットストリーム ヘッダーに同期コマンドが含まれている場合は、FPGA コンフィギュレーション コントローラーが、接続されているフラッシュの RCR (読み出しコンフィギュレーション レジスタ) に対して非同期書き込みを実行し、同期モードおよびレイテンシ ビットを設定します。フラッシュの RCR への書き込みが完了すると、FPGA コントローラーは同期読み出しを開始してビットストリームのデータ コンテンツを読み出します。コンフィギュレーション完了後、フラッシュは同期読み出しモードを維持します。

図 4 のタイミング波形は、同期読み出しモードの BPI コンフィギュレーションを開始する 3 つの段階を表しています。

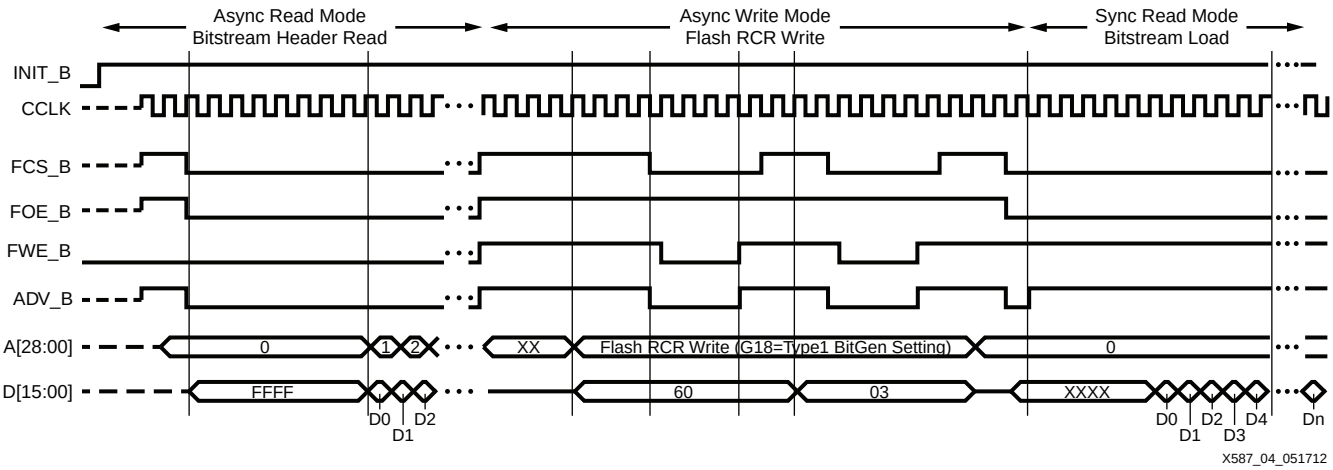


図 4 : BPI 高速コンフィギュレーションのタイミング波形

最初の段階では、非同期モードでビットストリームを読み出してターゲット モードを決定します。INIT_B 信号がリリースされ、インクリメントされた有効アドレス A[28:00] で制御信号 (FCS_B、FOE、ADV_B) がアサートされると、データ バス D[15:0] 上のフラッシュからのデータがキャプチャされま

す。その後、ビットストリーム ヘッダーが読み出されて、デザインで使用される読み出しモードが決定します。

2 番目の段階では、FPGA がフラッシュの読み出しコンフィギュレーション レジスタへ書き込み、フラッシュの同期モードを有効にします。フラッシュの同期モード ビットを書き込むために、FPGA は非同期書き込みシーケンスを発行します。FPGA は書き込み動作を実行するため、INIT_B 信号と FOE_B 信号がディアサートされている間に FCS_B 信号と FWE_B 信号をアサートします。書き込み動作では、FWE_B または FCS_B の先に到達した信号の立ち上がりエッジでアドレスおよびデータがラッチされます。FPGA は、フラッシュのコンフィギュレーション レジスタへ書き込みシーケンスを 2 サイクル間発行します。最初のサイクルでは、A[16:01] の RCR (読み出しコンフィギュレーション レジスタ) データおよびデータ バスのコマンド `0x60` が含まれます。次のサイクルでは、A[16:01] の RCR データおよびデータ バスのコマンド `0x03` が含まれます。RCR 値は、Micron G18 と P30 ファミリで異なり、BitGen の **`-g BPI_sync_mode`** オプションで決定します。オプションの詳細は、表 4 を参照してください。

最後の段階では、最終的に FPGA が非同期読み出しから同期読み出しプロトコルへ切り替わり、ビットストリームのデータ コンテンツの読み出しを再開します。このシーケンスは、FCS_B 信号および FOE_B 信号をアサートし、有効アドレスで ADV_B を 1 サイクル間アサートすると実行されます。そして、コンフィギュレーション データがフラッシュからバースト転送されて、FPGA によってリードバックされます。

ここで重要なのは、フラッシュはコンフィギュレーションで使用された読み出しモードを維持することです。たとえば、非同期読み出しモードで FPGA がコンフィギュレーションされると、フラッシュは非同期読み出しモードを維持します。逆に、同期読み出しモードで FPGA がコンフィギュレーションされた場合、フラッシュは同期読み出しモードを維持します。

コンフィギュレーション時間

このアプリケーション ノートで説明している BPI 高速コンフィギュレーションのセットアップでは、EMCCLK を使用しています。EMCCLK の最大周波数を計算するには、パラレル NOR フラッシュの Clock-to-Out 仕様とデータシートに記載の FPGA セットアップ仕様の両方が必要です。その他にボード遅延も考慮する必要があります。サポートされる最大 EMCCLK 周波数 (F_{EMCCK}) は、7 シリーズ FPGA の各データシート [参照 4] [参照 5] [参照 6] に記載されており、これらの値を超えてはいけません。BPI 高速コンフィギュレーションの最大 EMCCLK の概算値は、式 1 から求められます。

$$\text{最大周波数} = \frac{1}{\text{フラッシュの Clock-to-Out (TCHQV)} + \text{FPGA データのセットアップ (TBPIDCC)} + \text{ボード遅延}} \quad \text{式 1}$$

注記：28F00AG18F フラッシュの Clock-to-Out 値 (T_{CHQV}) が 5.5ns で、Virtex-7 XC7VX485T FPGA のデータ セットアップ値 (T_{BPIDCC}) が 4ns の場合、ボード遅延が無視できるレベルであれば、EMCCLK の最大値にはデータシートの F_{EMCCK} 値を使用できます。

表 3 では、VC707 ボードでの BPI 高速コンフィギュレーションの時間と、同じボードで非同期読み出しモードおよび CCLK を使用した BPI コンフィギュレーションの時間を比較しています。コンフィギュレーションの概算時間は、式 2 から求められます。

$$\text{コンフィギュレーション時間} = \frac{\text{ビットストリーム サイズ}}{\text{コンフィギュレーション クロック周波数} \times \text{データ バス幅}} \quad \text{式 2}$$

表 3 : VC707 ボードにおける BPI コンフィギュレーション時間の例 (28F00AG18F フラッシュメモリから XC7VX485T FPGA)

読み出しモード	コンフィギュレーション クロック ソース	フラッシュ データ幅	コンフィギュレーション時間 (162,187,488 ビット)
同期読み出し	EMCCLK (VC707 80MHz オシレーター)	x16	126ms
非同期読み出し	CCLK = 6MHz	x16	1.7s

非同期読み出しモードでは、最大クロック周波数を決定する FPGA の T_{BPICCO}、フラッシュの T_{ACC}、FPGA のデータ セットアップ T_{BPIDCC} などのパラメーターをさらに考慮する必要があります。EMCCLK ではなく内部コンフィギュレーション クロック (CCLK) を使用する場合には、最大クロック周波数が決定した後、CCLK 周波数偏差 ($F_{MCCKTOL}$) を考慮します。28F00AG18F フラッシュは、非同期読み出しモードで 9MHz を少し上回るレベルの動作が可能ですが、CCLK 周波数は $F_{MCCKTOL}$ を踏まえて通常 6MHz (3MHz ～ 9MHz の範囲) に制限されます。

BPI 高速コンフィギュレーションのファイル生成およびフラッシュ プログラム

ISE デザイン ツールを使用して 7 シリーズ FPGA のビットストリームとフラッシュのプログラム ファイルを作成し、パラレル NOR フラッシュ デバイスを間接的にプログラムします。図 5 に、ISE デザイン ツールのフロー (デザイン入力、BitGen、PROMGen、および iMPACT) を示します。BPI 高速コンフィギュレーション モード用に適切なファイルを生成してフラッシュをプログラムするには、オプション設定が必要です。このフローとオプションを使用するリファレンス デザインは、Virtex-7 FPGA VC707 評価ボードのデモンストレーション (http://japan.xilinx.com/support/documentation/vc707_14-1.htm) を参照してください。

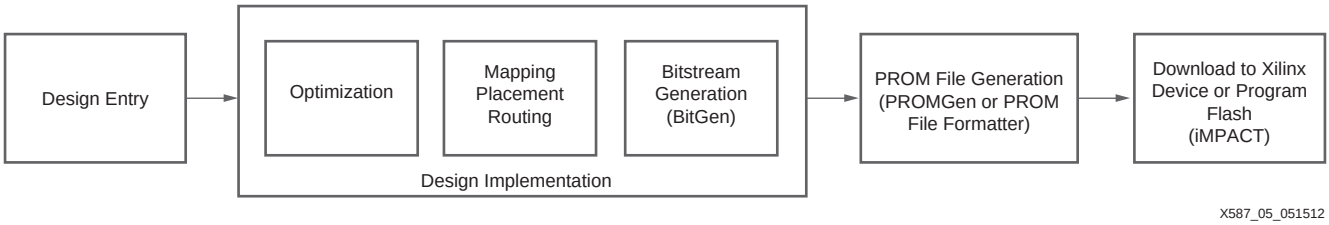


図 5 : ISE デザイン ツールの概要フロー

このセクションでは、次を使用する場合のセットアップおよびツール フローについて説明します。

- ISE ツール (v14.1)
- VC707 ボード (28F00AG18F パラレル NOR フラッシュと USB ケーブルを使用する Virtex-7 XC7VX485T FPGA ボード)

EMCCLK 用のデザイン入力

BPI コンフィギュレーションで EMCCLK オプションを使用するアプリケーションでは、EMCCLK 多目的ピンに I/O 規格が定義されている必要があります。たとえば、VC707 ボードの Virtex-7 FPGA デザインは、EMCCLK を 80MHz で使用するために次のユーザー制約オプションを適用する必要があります。

```
NET "EMCCLK"          LOC = AP37 | IOSTANDARD = LVCMOS18;
```

BPI コンフィギュレーション用のビットストリーム

このセクションでは、同期読み出しを用いる最高速の BPI コンフィギュレーション モードで使用可能なビットストリームを生成する際に必要なオプションについて説明します。表 4 では、VC707 ボード用に推奨されるオプションを示し、BPI コンフィギュレーション モードに関連する機能について説明しています。BitGen コマンドで指定されないオプションは、デフォルト設定を使用します。

表 4 : 同期読み出しを使用する BPI コンフィギュレーション モード用の BitGen コンフィギュレーション オプション

BitGen コマンド ライン	説明
-g StartupClk:Cclk	オプション : StartupClk:Cclk|UserClk|JtagClk デフォルト : Cclk FPGA のスタートアップ シーケンスを駆動するクロックとして FPGA の CCLK を指定します。BPI コンフィギュレーションには、Cclk オプションが必要です。
-g Compress	オプション : Compress デフォルト : OFF FPGA ビットストリーム ファイルの圧縮の有効/無効を指定します。任意オプションですが、フラッシュの間接プログラム時間および FPGA のコンフィギュレーション時間の両方を短縮するために推奨されるオプションです。圧縮することでファイル サイズを大幅に縮小できますが、圧縮率はユーザー デザインによって異なります。
-g ConfigRate:3	オプション : ConfigRate:3|6|9|12|16|22|26|33|40|50|66 デフォルト : 3 CCLK の標準周波数として 3MHz を指定します。FPGA の CCLK は、BPI コンフィギュレーションのタイミングを制御します。このアプリケーション ノートでは EMCCLK の使用を前提としているため、デフォルト値を推奨しており、最初のビットストリーム ヘッダーを読み出す際にのみ使用します。EMCCLK を使用しないデザインでは、$F_{MCCKTOL}$ を考慮した上で FPGA およびフラッシュでサポートされる最大クロック周波数に対応する周波数まで増加できます。
-g BPI_sync_mode:Type1	オプション : BPI_sync_mode:Disable|Enable デフォルト : Disable デフォルトは無効で、7 シリーズ FPGA フラッシュは非同期読み出しモードとなります。同期読み出しには、次のいづれかに設定する必要があります。 Type1 = G18 Type2 = P30 このアプリケーション ノートでは、Type1 を使用しています。
-g ExtMasterCclk_en:div-1	オプション : ExtMasterCclk_en:Disable|div-8|div-4|div-2|div-1 デフォルト : Disable BPI コンフィギュレーション モードで正確な CCLK が必要な場合は、外部マスター コンフィギュレーション クロックが必要です (内部マスター CCLK は、$F_{MCCKTOL}$ 周波数偏差が大きい)。div-1 と設定すると、1 分周された EMCCLK が使用される最大周波数となります。div-2 は EMCCLK を 2 分周、div-4 は 4 分周、div-8 は 8 分周します。選択した周波数がサポートされていることをフラッシュのデータシートで確認する必要があります。
-g BPI_1st_read_cycle:1	オプション : BPI_1st_read_cycle:1|2|3|4 デフォルト : 1 同期読み出しを使用する BPI コンフィギュレーション モードではサポートされていないため、デフォルトのままにしてください。
-g BPI_page_size:1	オプション : BPI_page_size:1|4|8 デフォルト : 1 同期読み出しを使用する BPI コンフィギュレーション モードではサポートされていないため、デフォルトのままにしてください。

BitGen のコマンド ライン例

ビットストリーム生成ツール (BitGen) は、コマンド ラインから簡単に実行できます。同期モードの BPI コンフィギュレーションで VC707 をターゲットとするコマンド ラインの例を次に示します。

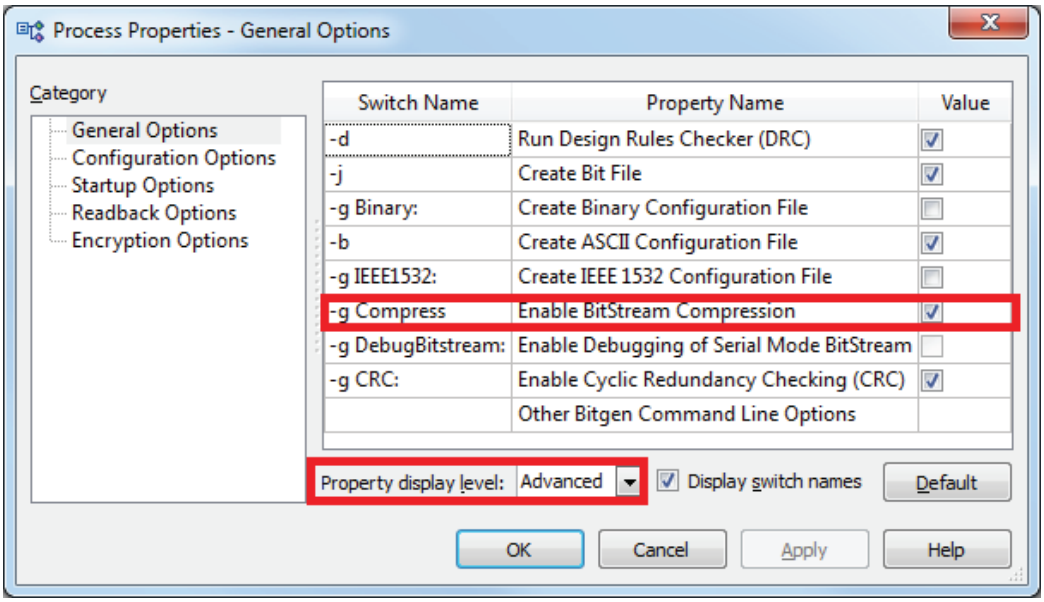
```
bitgen.exe -w -g StartUpClk:CClk -g Compress -g ConfigRate:3 -g
BPI_sync_mode:Type1 -g ExtMasterCclk_en:div-1 -g BPI_page_size:1 -g
BPI_1st_read_cycle:1 design.ncd design.bit
```

注記：VC707 ボードには、80MHz の EMCCLK FPGA I/O へ接続されたオシレーターが搭載されているため、フラッシュおよび FPGA の最大同期周波数を超えることなく div-1 オプションを使用できます。

BitGen の GUI 例

ビットストリームは、ISE ツールの Project Navigator フローからも生成できます (図 6)。Project Navigator では、オプション設定に加えてデフォルト設定が使用されます。

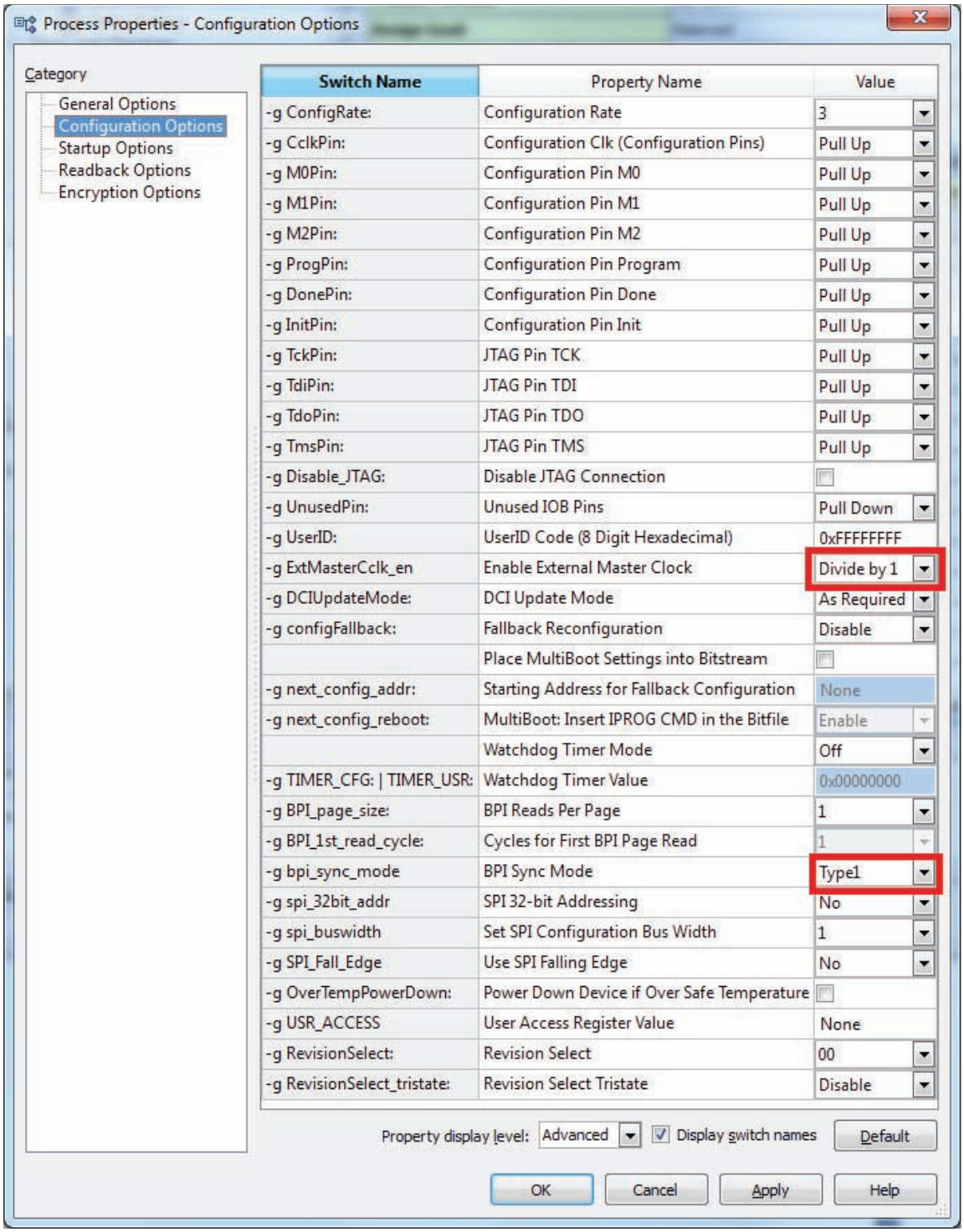
- [Generate Programming File] を右クリックし、プロセス プロパティを開きます。
  - [Property display level] で [Advanced] を選択し、すべてのオプションを表示します。
  - [General Options] にある -g Compress スイッチはオプションですが、プログラム時間やコンフィギュレーション時間が短縮されるため選択しています。



X587_06_051512

図 6 : BPI 高速コンフィギュレーション モードでの一般的なビットストリーム生成オプション

- [Configuration Options] では、外部マスター コンフィギュレーション クロックを用いる同期読み出しモードの BPI コンフィギュレーション用にビットストリームを生成するため、図 7 のようにオプションを選択してください。



X587_07_051512

図 7 : BPI 高速コンフィギュレーション モード用のビットストリームを生成するためのコンフィギュレーション オプション

パラレル NOR フラッシュのプログラム ファイルの生成

ISE ツールの PROMGen または iMPCAT の File Formatter でフラッシュ ファイルを生成します。これらの ISE ツールは、FPGA ビットストリーム (.bit) を使用してパラレル NOR フラッシュのプログラムに使用できるフラッシュ ファイル (.mcs) を生成します。表 5 では、このファイル生成に使用するオプションについて説明します。

表 5 : PROMGen オプション

PROMGen オプション	説明
-p <format>	ファイル形式です。最も一般的な形式は、インテル HEX 形式 (.mcs)です。
-c FF	フラッシュ フィルの値です。
-s <size>	キロバイト単位のフラッシュ サイズ (2 のべき乗となる) です。指定しない場合、デフォルトでファイルのサイズが定義されます。
-o <filename>	出力されるフラッシュ ファイルの名前です。
-u <hexaddr> <file[.bit]>	指定された開始アドレスから上方向に .bit ファイルをロードします。このオプションは、入力ビットストリーム ファイルの直前に指定します。
-data_width	ターゲット フラッシュのデータ幅を指定します。このビット/バイト順は、ターゲット フラッシュのデータ バス幅に応じて、x8 または x16 に変更する必要があります。x16 は、BPI コンフィギュレーション インターフェイスの同期読み出しの場合にのみサポートされます。
-w	既存の出力ファイルを上書きします。
-data_file	フラッシュ ファイルに追加するための開始アドレス、方向、およびデータ ファイルを指定します。
-bpi_dc	最初の FPGA からのデイジーチェーン出力を選択します。7 シリーズ FPGA BPI 高速コンフィギュレーション モードの場合、有効なオプションはパラレルのみです。

PROMGen のコマンド ライン例

ISE ツールの PROMGen は、コマンド ラインから簡単に実行できます。PROMGen コマンド ラインの例を次に示します。

```
Promgen -w -p mcs -c FF -bpi_dc parallel -data_width 16 -o flash_file_name
-s 131072 -u 00000000 design.bit
```

このコマンド ラインでは、次のオプションでパラレル NOR ファイルを生成します。

- format = mcs
- data bus = 16
- flash size = 1 Gb (131,072 x 1,024 bytes x 8 bits = 1,073,741,824)
- BPI from address 0
- output file = design.mcs
- fill value = FF
- input bitstream = design.bit

PROM File Formatter の GUI フロー例

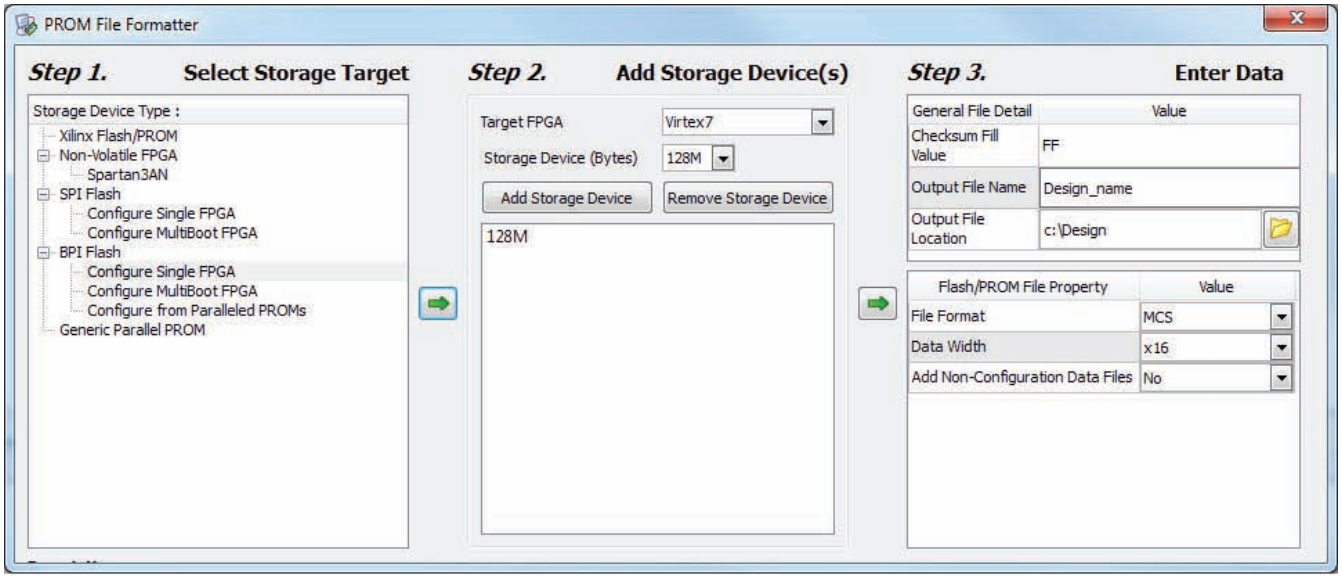
次の手順に従い、iMPACT ツールでフラッシュ ファイルを生成することも可能です。

1. iMPACT ツールを起動し、左上にある [iMPACT Flows] パネルで **[Create PROM File (PROM File Formatter)]** をダブルクリックします。
2. ウィザードが表示されるので、その指示に従って進みます (図 8)。

a. [Step 1] ([Select Storage Target]) で **[BPI Flash]** → **[Configure Single FPGA]** を選択し、緑色の矢印をクリックします。

b. [Step 2] ([Add Storage Device(s)]) で [Target FPGA] (例：**Virtex-7**) および [Storage Device (Bytes)] (28F00AG18F の場合は **128M**) を選択し、**[Add Storage Device]** をクリックします。緑色の矢印をクリックします。

c. [Step 3] ([Enter Data]) で出力ファイル名、出力先、フラッシュ プロパティを選択し、**[OK]** をクリックします。



X587_08_051512

図 8 : PROM File Formatter のウィザード

3. **[OK]** をクリックし、ターゲット ユーザー デザインのビットストリーム (`.bit`) ファイルを追加します。デザイン イメージが 1 つの場合は、次のウィンドウ (イメージが複数のデザイン用) で **[No]** をクリックします。**[OK]** をクリックしてデザイン ファイルの入力を完了し、サマリ ページでも同様に [OK] をクリックします。

4. プロジェクトの実行が完了後、**[Generate File]** をクリックしてフラッシュのプログラム イメージを生成します。

iMPACT によるフラッシュの間接プログラム用の Virtex-7 FPGA VC707 ボード セットアップ

機能およびボードの使用方法については、『Virtex-7 FPGA VC707 評価ボード ユーザー ガイド』[参照 7] を参照してください。iMPACT を使用してパラレル NOR フラッシュをプログラムし、Virtex-7 FPGA を正常にコンフィギュレーションするための基本的なボード セットアップについて簡潔に説明します。

1. モード ピン M[2:0] およびリビジョン セレクト RS[1:0] に接続された 2 つの上位アドレス ライン A[25:24] が、図 9 のようにデフォルト設定になっていることを確認します。

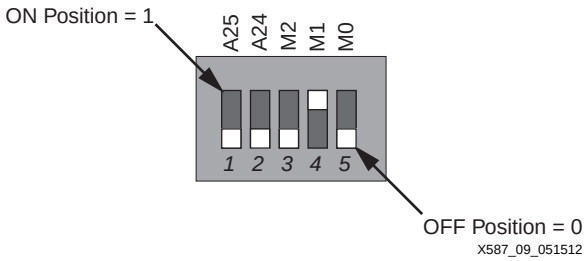


図 9 : VC707 SW1 : モード ピンおよび RS[1:0] ピンの設定

2. VC707 ボードの電源をオンにします。
3. USB (A to Micro B タイプ) ケーブルを VC707 ボード上の USB JTAG (Digilent) コネクタへ接続するか、またはザイリンクス USB II ケーブルをヘッダーへ接続します。

ケーブル オプション

iMPACT プログラミング ツールは、ザイリンクス プラットフォーム ケーブル USB II ケーブルまたは Digilent 社の JTAG SMT1 モジュール (オンボード) を使用し、Virtex-7 FPGA JTAG インターフェイス経由で VC 707 パラレル NOR フラッシュを間接的にプログラムします。表 6 に、iMPACT による間接プログラム ソリューションで利用可能なケーブルおよび最大周波数を示します。

表 6 : VC707 ケーブル サポート

ケーブル	インターフェイス	BPI 間接プログラムで利用可能な最大周波数 (MHz)
ザイリンクス プラットフォーム ケーブル USB II	USB	12
Digilent 社製 JTAG SMT1 モジュール (オンボード)	USB	30

JTAG 信号の接続およびヘッダーの詳細については、各 USB ケーブルのデータシートを参照してください。

iMPACT によるパラレル NOR フラッシュの間接プログラム

パラレル NOR フラッシュを間接的にプログラムするには、iMPACT ツールを使用します。パラレル NOR フラッシュ メモリ デバイスには JTAG インターフェイスがないため、iMPACT ツールとフラッシュ間のブリッジとしてロジックを追加する必要があります。このソリューションでは、iMPACT が 49.1 (JTAG) バスとパラレル NOR フラッシュ バス インターフェイス間のブリッジ機能を果たすビットストリームで FPGA をコンフィギュレーションします。このソリューションは、iMPACT からフラッシュが直接プログラムされるのではなく FPGA を介してプログラムされるため、間接プログラムと呼んでいます。図 10 に、iMPACT による間接プログラム ソリューションの基本セットアップを示します (Virtex-7 FPGA VC707 評価ボードを使用)。

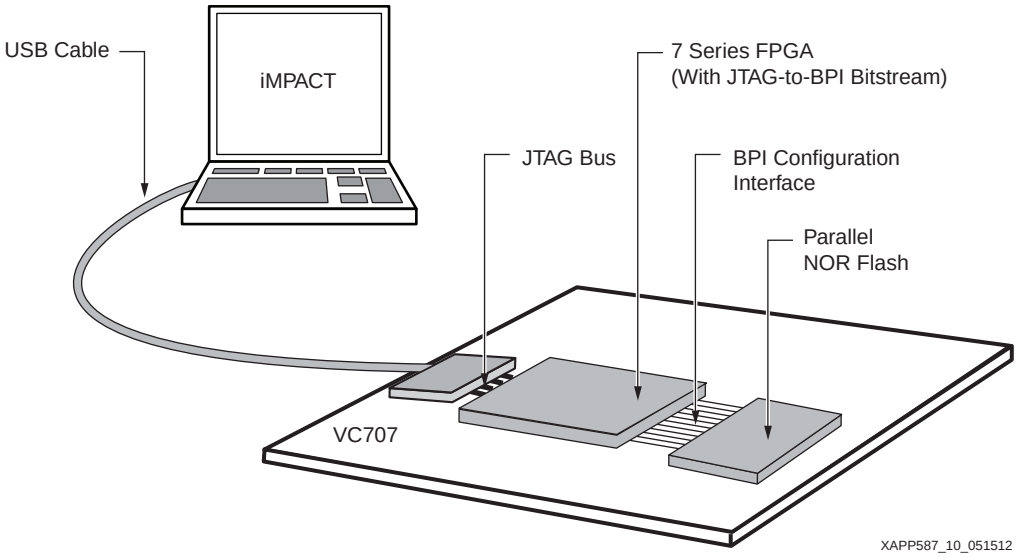


図 10 : iMPACT によるフラッシュの間接プログラムのセットアップ

VC707 評価ボードに搭載された Virtex-7 XC7VX485T デバイスおよび Micron 28F00AG18F パラレル NOR フラッシュを使用する場合の iMPACT によるプログラム フローを説明します。プログラミング プロセスを開始する前に、15 ページの「iMPACT によるフラッシュの間接プログラム用の Virtex-7 FPGA VC707 ボード セットアップ」で説明されている基本的なボードとケーブルのセットアップ要件に従う必要があります。

ボードのセットアップ完了後、ボード上のフラッシュはボードから取り外さずにプログラムできます。最初に、プログラミング ケーブルとフラッシュのブリッジとなる生成済みのビットストリーム用い、iMPACT を使用して JTAG 経由で 7 シリーズ FPGA をコンフィギュレーションします。生成済みのビットストリームをロードするために、FPGA の既存コンテンツは消去されます。このビットストリーム デザインは、パラレル NOR インターフェイスに関連しないすべての未使用 I/O を弱いプルアップ抵抗を使用してトライステートにします。設計者はこの点に注意し、アプリケーション内でこの FPGA に接続しているその他のデバイスに悪影響を及ぼさないようにする必要があります。プルアップされるべきでない未使用 I/O が存在する場合には、外部抵抗終端を使用します。ビットストリーム デザインは iMPACT プログラミング ツールに統合されており、フラッシュの間接プログラム中に自動的にバックグラウンドでロードされます。コアは iMPCAT のランタイムに使用され、デザイン統合用のソースコードはありません。ビットストリーム デザインのロードが完了すると、iMPCAT は指定したターゲット デバイスおよび動作に基づいて、フラッシュにコマンド シーケンスを発行します。

iMPCAT ツールは、バッチ モードあるいは GUI で実行できます。次に、iMPACT を使用して間接的にフラッシュをプログラムする手順を説明します。

パラレル NOR フラッシュを間接的にプログラムする場合の iMPACT のバッチ モード例

このセクションでは、コマンド ファイルを使用してバッチ モードで iMPCAT を実行する場合の一般的な動作とオプションを示します。

```
iMPACT -batch design.cmd

Sample design.cmd contents:
  setMode -bs
  setCable -port auto
  Identify -inferir
  identifyMPM
  attachflash -position 1 -bpi "28F00AG18F"
  assignfiletoattachedflash -position 1 -file "designname.mcs"
  Program -p 1 -dataWidth 16 -rs1 25 -rs0 24 -bpionly -e -loadfpga
  closeCable
```

```
  Quit

Optional instructions include flash verify and readback commands:
  BlankCheck -p 1 -dataWidth 16 -rs1 25 -rs0 24 -bpionly -loadfpga
  Verify -p 1 -dataWidth 16 -rs1 25 -rs0 24 -bpionly -loadfpga
```

パラレル NOR フラッシュを間接的にプログラムする iMPACT の GUI 例

1. iMPCAT ツールは、スタンドアロンまたは ISE Project Navigator 内で **[Configure Target Device]** プロセスをクリックして起動します。iMPCAT 画面左上の [iMPACT Flows] の下にある **[Boundary Scan]** をダブルクリックします。
2. **[Initialize Chain]** アイコンを選択し、ボード上のターゲット デバイスを確認します。
3. iMPACT ツールで、ファイルの割り当て用のプロンプトが表示されます。**[No]** をクリックします。
4. FPGA アイコン上にある **[SPI/BPI?]** と表示された破線ボックスを右クリックし、**[Add SPI/BPI Flash]** をクリックします。表示された画面で、ターゲット フラッシュ イメージを指定します。
5. VC707 ボード上の 1Gb パラレル NOR フラッシュとして、**BPI PROM** と **28F00AG18F** を選択します。バス データ幅は **16**、RS[1:0] は **25:24** に設定されていることを確認します。**[OK]** をクリックします。

   注記：VC707 ボードでは、RS0 と RS1 はそれぞれ上位フラッシュ アドレス ビット (A24 および A25) へ接続された FPGA ピンであるため、iMPCAT コアはこの設定で FPGA の RS0 と RS1 信号を駆動します。RS[1:0] ピンはオプションであり、1 つのフラッシュに複数のデザインが格納されている場合に外部からリビジョンを管理するために使用されます。リビジョン管理に RS[1:0] を使用しない場合、このオプションはプログラム中 **[Not Used]** に設定されている必要があります。

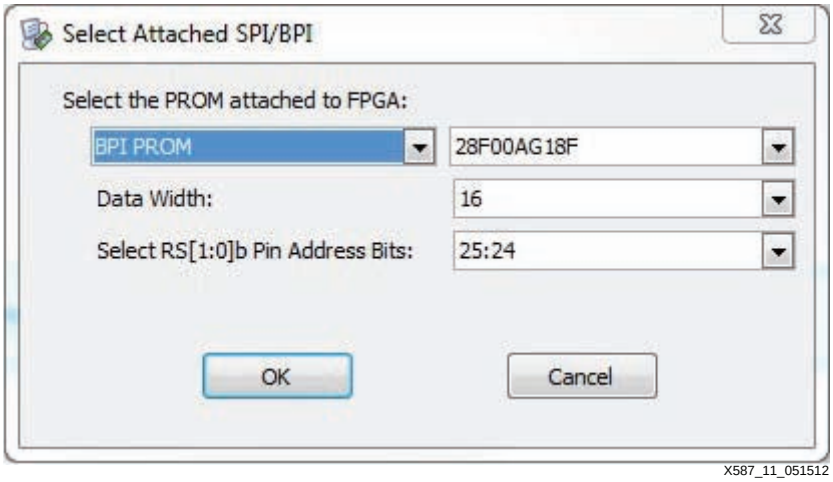


図 11：ターゲット フラッシュの選択

6. ターゲットとなるパラレル NOR フラッシュを選択した後、FPGA アイコンの上にあるフラッシュ アイコンを右クリックし、フラッシュで実行する動作あるいは動作プロパティを選択します。**[Program]** をクリックすると、デフォルトで常にデザイン指定の消去、プログラム、および検証が実行されます。

   注記：デザイン指定の消去では、ターゲット プログラム ファイルでアドレス指定されたアドレス空間内のフラッシュのコンテンツが直近のフラッシュ バウンダリまで消去されます。フラッシュ全体を消去する場合は、**[Set Erase Properties]** をクリックして **[Full Chip Erase]** に設定を変更してください。

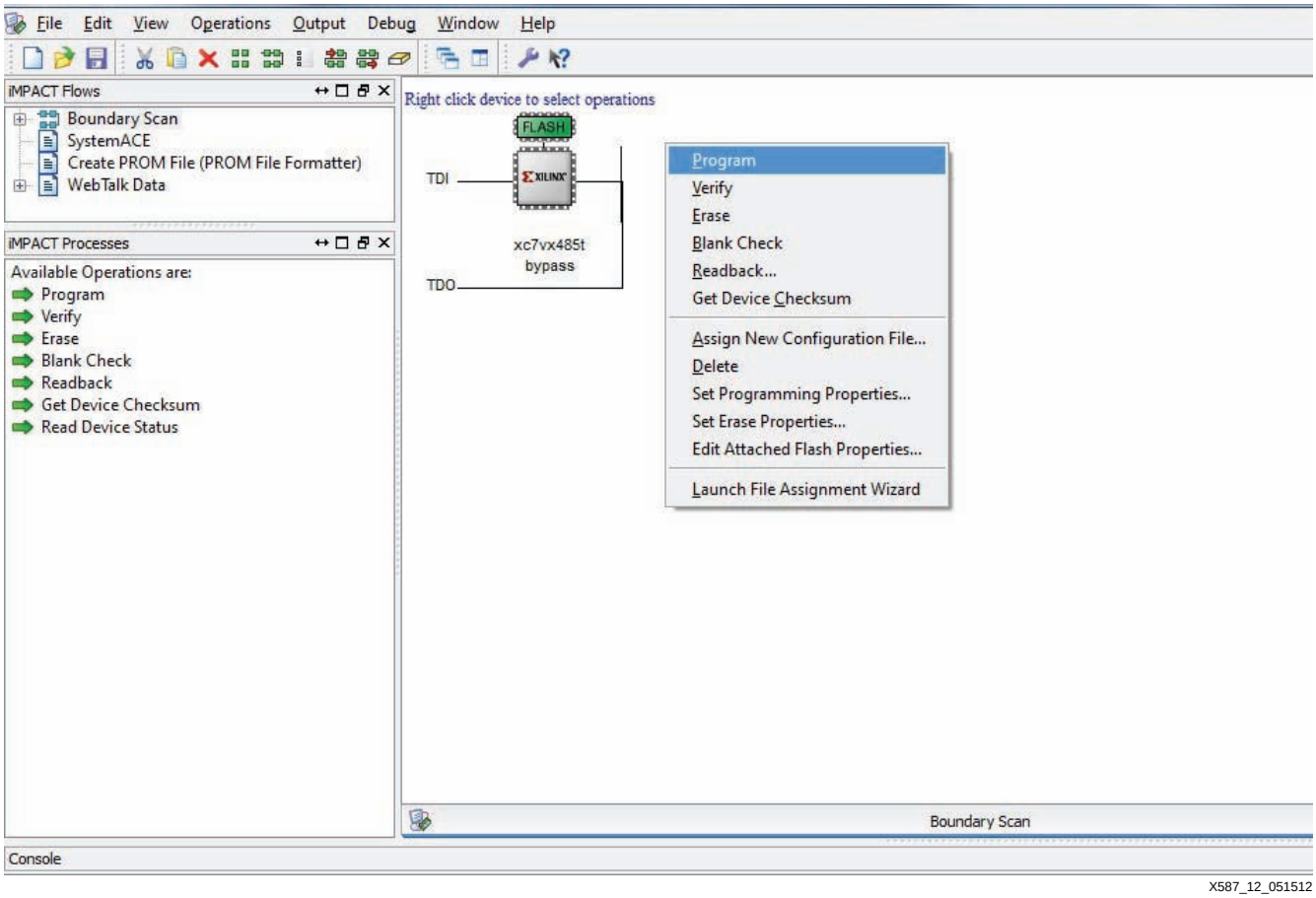


図 12 : 間接的フラッシュ動作

フラッシュ動作を設定した後、iMPCAT ツールは生成済みのビットストリームをダウンロードし、CFI (共通フラッシュ インターフェイス) のパラメーター (集積度、ファミリ) を照合します。次は、VC707 28F00AG18F における CFI クエリを示した iMPACT のログ表示の例です。

```
Populating BPI common flash interface ...
   Common Flash Interface Information Query completed successfully.
   INFO:Cse - Common Flash Interface Information from Device:
   INFO:Cse - Verification string: 51 52 59
   INFO:Cse - Manufacturer ID:     89
   INFO:Cse - Vendor ID:           00
   INFO:Cse - Device Code:         1b
   Reset Core
   Using x16 mode ...
   Set Data Width
   Using RS Pins [25,24]...
```

iMPACT によるフラッシュの間接プログラムの概算時間

VC707 ボードで iMPACT 14.1 を使用し、ボードに接続した USB ケーブルをデフォルトの 10MHz で動作させた場合の圧縮および未圧縮 XC7VX485T ビットストリームの動作時間例をいくつか紹介します (表 7)。これらの値は、参照用であり保証された値ではありません。

表 7 : フラッシュ動作の参照用概算時間

イメージ	時間 (秒)		
	消去	プログラム	検証
XC7VX485T 圧縮された 15Mb	12	90	47
XC7VX485T 未圧縮の 157.4Mb	82	918	466
1Gb フル イメージ	103	6612	3345

iMPACT の生成済みビットストリームによるその他のシステムへの影響

ターゲット フラッシュの動作を実行する前に iMPACT から JTAG-BPI の生成済みビットストリームが FPGA へロードされます。JTAG-BPI ビットストリームは、すべての未使用 I/O にプルアップ抵抗を設定します。FPGA の I/O がその他のデバイスで制御されており、このプルアップ抵抗によって競合や予期しない動作が生じる可能性がある場合には、プログラム中はそのデバイスが無効となるようにプルアップ/プルダウンを追加する必要があります。

iMPACT のビットストリームがロードされると、フラッシュの動作要求に応じて FPGA 信号を駆動します。このとき駆動される信号は、インターフェイス信号の A[28:00]、D[15:00]、FCS_B、FWE_B、FOE_B、ADV_B、およびオプションの RS[1:0] です。デザインで使用される I/O 規格は、CFGBVS 1.8V または 3.3V と互換性を持ちます。生成済みビットストリームのコントローラーがアクティブでない場合、これらのフラッシュ インターフェイス信号はトライステートとなるため、必要に応じてユーザー アプリケーションで外部終端を行う必要があります。

RS[1:0] リビジョン セレクト機能を使用するアプリケーションでは、FPGA の RS[1:0] ピンがフラッシュの上位アドレス信号へ接続されます。アドレス ピンを接続していないデザインでも、コアは RS[1:0] ピンと A[28:00] ピンの両方を駆動します。たとえば、1Gb フラッシュ デバイスで、上位アドレス ライン (A[25:24]) が RS[1:0] へ接続されている場合には、FPGA の A[25:24] 信号および RS[1:0] 信号の両方がフラッシュ動作中にコアによって駆動されます。

チェックリストとデバッグ

このセクションでは、BPI コンフィギュレーション モードおよびフラッシュの間接プログラムの重要点と一般的なデバッグ手順について簡潔に説明します。

コンフィギュレーション

- 7 シリーズ FPGA のステータス レジスタのデータは、JTAG を介して iMPACT に読み出すことができます。このレジスタは具体的なエラー状況をキャプチャするため、たとえば DONE 信号と INIT 信号が Low の場合のエラー タイプの特定に有効です。さらに、ステータス レジスタでは、モード ピン設定 M[2:0] やバス幅検出 (x16 の場合は CFG BUS [25:24]=01 に設定すべき) を確認できます。
- 非同期モード用に生成されたビットストリームおよび内部コンフィギュレーション　クロックのデフォルト設定は、同期読み出しでコンフィギュレーション エラーが生じた場合にデバッグ ツールとして利用可能です。非同期読み出しビットストリーム テストでは、ボード上のインターフェイス接続が適切であるか、また最大クロック周波数を超えていないかを判断するのに使用できます。このテスト用には、**`-g ExtMasterCclk_en:Disable`** および **`-g BPI_sync_mode:Disable`** オプションを使用した新しい非同期読み出しビットストリームが生成されます。
- パラレル NOR フラッシュへアクセスしない FPGA デザインでは、FPGA ユーザー デザインが動作中にフラッシュを安全かつ低電力状態に保つために、次のようにすることが推奨されます。
  - FPGA FCS_B ピンを High 駆動
  - FPGA FOE_B ピンを High 駆動
  - FPGA FWE_B ピンを High 駆動

ファイル生成

- RTL の記述 ：EMCCLK の I/O 規格を必ず定義してください (10 ページの「EMCCLK 用のデザイン入力」を参照)。
- BitGen の使用 ：**`-g BPI_sync_mode:Type1 -g ExtMasterCclk_en:div-1`** で同期モードと EMCCLK を有効にしてください。より高速なコンフィギュレーションおよびプログラムを実行するには、**`-g compress`** オプションを使用してください。また、ConfigRate オプションは、非同期読み出しモードでターゲット フラッシュおよびターゲット FPGA でサポートされる最大周波数を超えないようにします。
- PROMGen の使用 ：オプションの **`-data_width 16`** データ バス幅が規定どおりに選択されていることを確認してください。

iMPACT による間接プログラム

- 消去動作 ：フラッシュ デバイスは不揮発性デバイスであるため、プログラム前に必ず消去する必要があります。
- BlankCheck 動作 ：この動作は、前の消去動作を確認するために使用します。
- フラッシュ動作の最大ケーブル速度 ： 間接プログラムの最大ケーブル速度を超えないようにします。16 ページの「ケーブル オプション」を参照してください。
- リビジョン セレクト ピン RS[1:0] の使用 ：これらのピンは、オプション機能をサポートします。FPGA の上位 2 つのアドレス ビットではなく、この機能をリビジョン管理に使用する場合は、iMPACT でこのオプション機能を選択する必要があります。

まとめ

7 シリーズの BPI 高速コンフィギュレーションでは、同期読み出しモードと EMCCLK オプションを使用することで、FPGA のコンフィギュレーション時間を大幅に短縮できます。このモードは、PCIe なの電源投入要件が厳密な規格に対応できます。このソリューションでは、パラレル NOR フラッシュをインシステムでプログラムできるため、7 シリーズ FPGA システムに大きな価値をもたらします。

参考資料

次の文書は、このアプリケーション ノートの補足資料です。

1. UG470 :『7 シリーズ FPGA コンフィギュレーション ユーザー ガイド』
2. iMPACT ユーザー マニュアル
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/isehelp_start.htm#pim_c_overview.htm
3. Micron 社製のエンベデッド メモリ StrataFlash
http://www.micron.com/parts/nor-flash/parallel-nor-flash/~/media/Documents/Products/Data%20Sheet/NOR%20Flash/Parallel/G18/6702128_256_512_1Gb_65nm_g18.ashx
4. DS181 :『Artix-7 FPGA データシート : DC 特性およびスイッチ特性』
5. DS182 :『Kintex-7 FPGA データシート : DC 特性およびスイッチ特性』
6. DS183 :『Virtex-7 FPGA データシート : DC 特性およびスイッチ特性』
7. UG885 :『Virtex-7 FPGA VC707 評価ボード ユーザー ガイド』
8. Virtex-7 FPGA VC707 評価キット
http://japan.xilinx.com/products/boards-and-kits/EK-V7-VC707-G.htm
9. ISE ソフトウェア マニュアル
http://japan.xilinx.com/support/software_manuals.htm
10. iMPACT によるフラッシュ デバイスのサポート一覧表
http://japan.xilinx.com/cgi-bin/docs/rdoc?v=latest_ise;d=isehelp_start.htm;a=pim_c_introduction_indirect_programming.htm

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	内容
2012 年 5 月 25 日	1.0	初版リリース
2012 年 5 月 31 日	1.0.1	PDF 資料のプロパティを変更。

Notice of Disclaimer

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products.To the maximum extent permitted by applicable law:(1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same.Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications.You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent.Certain products are subject to the terms and conditions of the Limited Warranties which can be viewed at http://www.xilinx.com/warranty.htm; IP cores may be subject to

warranty and support terms contained in a license issued to you by Xilinx.Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in Critical Applications:http://www.xilinx.com/warranty.htm#critapps.

Automotive Applications Disclaimer

XILINX PRODUCTS ARE NOT DESIGNED OR INTENDED TO BE FAIL-SAFE, OR FOR USE IN ANY APPLICATION REQUIRING FAIL-SAFE PERFORMANCE, SUCH AS APPLICATIONS RELATED TO:(I) THE DEPLOYMENT OF AIRBAGS, (II) CONTROL OF A VEHICLE, UNLESS THERE IS A FAIL-SAFE OR REDUNDANCY FEATURE (WHICH DOES NOT INCLUDE USE OF SOFTWARE IN THE XILINX DEVICE TO IMPLEMENT THE REDUNDANCY) AND A WARNING SIGNAL UPON FAILURE TO THE OPERATOR, OR (III) USES THAT COULD LEAD TO DEATH OR PERSONAL INJURY.CUSTOMER ASSUMES THE SOLE RISK AND LIABILITY OF ANY USE OF XILINX PRODUCTS IN SUCH APPLICATIONS.

本資料は英語版 (v1.0.1) 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。

この資料に関するフィードバックおよびリンクなどの問題につきましては、jpn_trans_feedback@xilinx.com までお知らせください。いただきましたご意見を参考に早急に対応させていただきます。なお、このメールアドレスへのお問い合わせは受け付けておりません。あらかじめご了承ください。