



この翻訳版ドキュメントのメンテナンスは終了しております。

この文書には、古いコンテンツや商標が含まれている場合があります。

最新情報につきましては、次のリンクから英語版の最新資料をご確認ください。

<https://www.intel.com/content/www/us/en/programmable/documentation/lit-index.html>

Please take note that this document is no longer being maintained. It may contain legacy content and trademarks which may be outdated.

Please refer to English version for latest update at

<https://www.intel.com/content/www/us/en/programmable/documentation/lit-index.html>

この資料は英語版を翻訳したもので、内容に相違が生じる場合には原文を優先します。こちらの日本語版は参考用としてご利用ください。設計の際には、最新の英語版で内容をご確認ください。

AN-650-1.0

このアプリケーション・ノートでは、HardCopy® デバイス・マイグレーション用の UniPHY Nios® II シーケンサを初期化するために、Altera® FPGA コンフィギュレーションで一般的に使用されるファースト・パッシブ・パラレル (FPP) コンフィギュレーション・インタフェースの再利用について説明します。

HardCopy デバイスで Nios II シーケンサを初期化する別の方法があります。リソースを節約し、ボード作成のやり直しを回避する Nios II シーケンサを初期化するために、既存の FPGA コンフィギュレーション・インタフェースを再使用することを推奨します。

-  Nios II シーケンサ初期値設定方法について詳しくは、「外部メモリ・インタフェース・ハンドブック Vol 3」の「HardCopy Migration」の章を参照してください。
-  各 FPGA ファミリの FPP コンフィギュレーション手法について詳しくは、それぞれのデバイス・ハンドブックのコンフィギュレーションの章を参照してください。

このアプリケーション・ノートには以下の項が含まれています。

- 1-2 ページの「Nios II シーケンサについて」
- 1-2 ページの「機能の説明」
 - 1-3 ページの「MAX II コントローラ」
 - 1-5 ページの「シーケンサ ROM コントローラ」
- 1-7 ページの「FPGA におけるデザイン検証」
 - 1-8 ページの「ソフトウェアとハードウェア要求」
 - 1-9 ページの「Nios II シーケンスの命令コードの変換」
 - 1-9 ページの「PFL メガファンクションによるプログラミング・フラッシュ・メモリ」
 - 1-10 ページの「MAX II コントローラおよびシーケンサ ROM コントローラの検証」

Nios II シーケンサについて

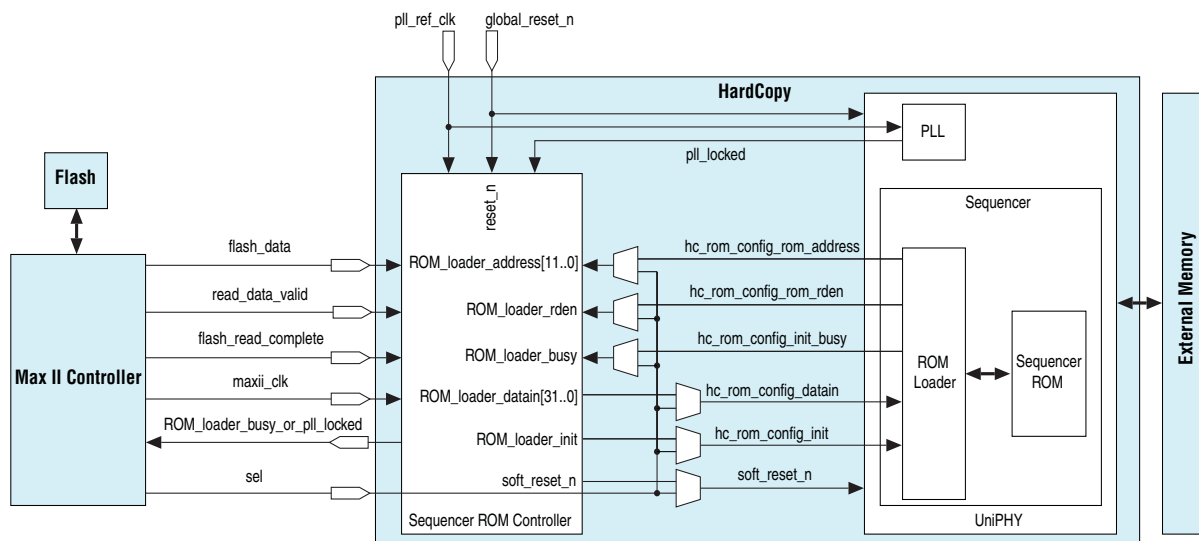
Nios II シーケンサは、UniPHY IP (Intellectual Property) の性能を向上させる、そしてキャリブレーション・プロセスがより堅牢なものにするために、RTL (レジスタ転送レベル) シーケンサを交換する UniPHY IP に統合されています。Nios II システムのブートアップは、UniPHY IP アドレスに Nios II シーケンサの統合後に必要とされています。Nios II 命令コードは、キャリブレーション・プロセスを開始するためのシステム・ブートアップ時に、Nios II システムにロードする必要があります。命令コードは FPGA 内部のメモリに保存され、FPGA のパワーアップ時に初期化されます。しかし、RAM は HardCopy デバイスの電源投入時に初期化できません。したがって、Nios II 命令コードは、HardCopy デバイスの RAM に格納することはできません。HardCopy デバイスの ROM に命令コードを格納することができますが、その後変更することはできません。将来的に命令コードを変更する柔軟性を提供するには、外部の不揮発性メモリに Nios II 命令コードを格納し、UniPHY IP の ROM ロダで Nios II シーケンサを初期化する必要があります。

機能の説明

リファレンス・デザインの 2 つの主要なブロックは、MAX II コントローラとシーケンサ ROM コントローラがあります。

図 1 には、リファレンス・デザインの MAX II コントローラおよびシーケンサ ROM コントローラの接続を示します。

図 1. MAX II コントローラおよびシーケンサ ROM コントローラの接続



MAX II コントローラ

FPGA FPP コンフィギュレーション手法では、MAX II コントローラは、フラッシュ・デバイスでのプログラミング・ファイルを読み込み、コンフィギュレーション・インタフェースのピンを介して FPGA を設定します。

このリファレンス・デザインでは、MAX II コントローラはフラッシュ・デバイスからの Nios II 命令コードを読み取るためにユーザー・モードになっています。その後、コード・リードは Nios II シーケンサを初期化するためにシーケンサ ROM コントローラへ送信されます。MAX II コントローラのリファレンス・デザインは、16 ビットのデータ・バスおよび 20 ビットのアドレス・バス付きの 512-MB の Numonyx P30 Flash とのみ適用されます。コントローラは、50 MHz で動作します。

図 2 には、MAX II コントローラのトップ・レベルを示しています。

図 2. MAX II コントローラのトップ・レベルのブロック図

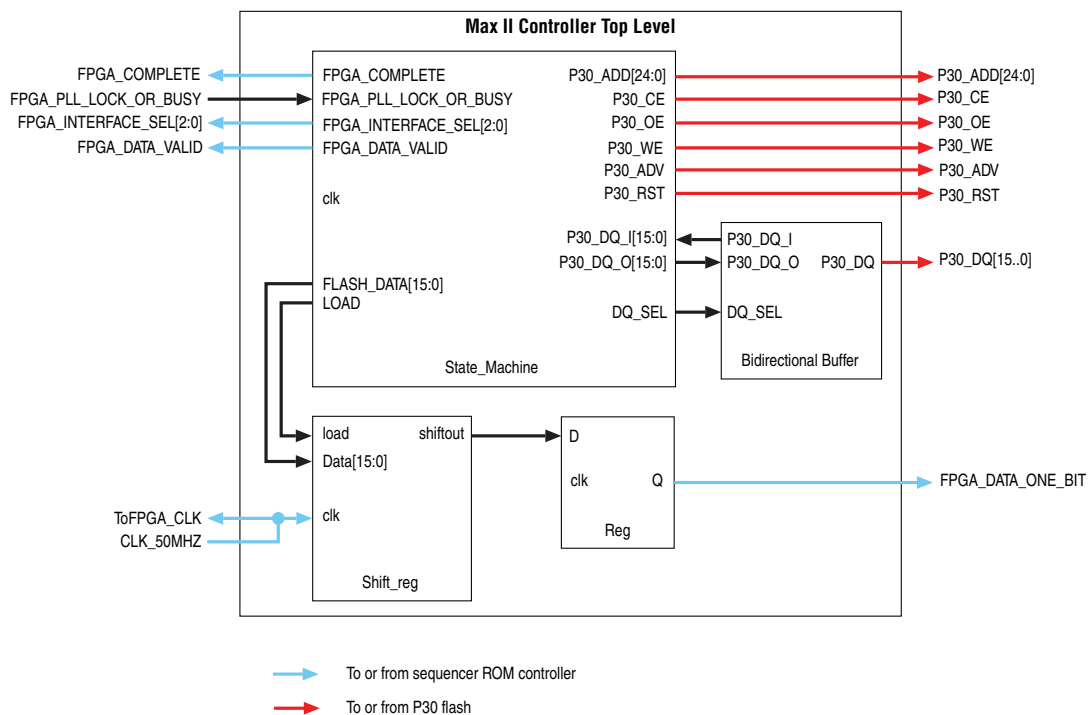


表 1 に、MAX II コントローラの入力ポートおよび出力ポートの機能について説明します。

表 1. MAX II コントローラの入力ポートと出力ポートの説明

MAX II コントローラの入力ポートと出力ポート名	説明
CLK_50MHZ	外部オシレータからクロックをソースします。
FPGA_PLL_LOCK_OR_BUSY	このポートは 2 つの機能があります。 <ul style="list-style-type: none"> ■ 初期ステージ <ul style="list-style-type: none"> ■ 1—MAX II コントローラのステート・マシンの実行を開始する。 ■ 0—アイドル状態。 ■ ステート・マシンの進行時 <ul style="list-style-type: none"> ■ 1—シーケンサ ROM コントローラはビジー状態になります。16 ビットのデータをシフト・アウトした後にアイドル状態のままになります。 ■ 0—シーケンサ ROM コントローラはアイドル状態になります。次の 16 ビット・データをシフトします。
FPGA_COMPLETE	すべての 4096 x 32 ビットのデータが完全にシフトされるシーケンサ ROM コントローラに通知します。
FPGA_DATA_ONE_BIT	1 ビット・フラッシュのデータ・パス。
FPGA_DATA_VALID	16 ビット・データがシフトし始めたことをシーケンサ ROM コントローラに通知します。 <ul style="list-style-type: none"> ■ 立ち上がりエッジ—16 ビット・データの最初のビットの送信を開始したことを示す。 ■ 立ち下がりエッジ—16 ビット・データの最後のビットが完全に送信されることを示す。
FPGA_INTERACE_SEL	複数のインターフェースから Nios II 命令コードをロードするためのインターフェースを選択します。
ToFPGA_CLK	データ同期のための FPGA のコントローラへのクロック。
P30*	P30 フラッシュ付きのインターフェースのポートです。

MAX II コントローラでは、HardCopy PLL がロックされる時、また FPGA_PLL_LOCK_OR_BUSY 信号が High のとき、フラッシュ・データの読み出しが開始されます。複数のインターフェースのデザインでは、次のサイクルで FPGA_PLL_LOCK_OR_BUSY 信号が Low のときに、コントローラはフラッシュ・データの後続のページを読み出すことを続けています。コントローラが初期化される Nios II シーケンサのインターフェースを選択する FPGA_INTERAFACE_SEL 信号を発行します。

I/O ピンの制限のため、コントローラは、HardCopy デバイスのシーケンサ ROM コントローラにデータをシフトする前に 16 ビットのフラッシュ・データをシリアル化します。FPGA_DATA_VALID 信号は、データ・リードが有効であることを示すために、データのシフトからシーケンサ ROM コントローラへの全ての単一のフラッシュ・データのリード中に、16 クロック・サイクルまでに High のままになります。フラッシュ・データ (4096 x 32 ビット) のページ全体を読み出した後、FPGA_COMPLETE 信号が High になり、シーケンサ ROM コントローラに MAX II コントローラがデータの単一のページの読み出しが完了されたことを示します。

シーケンサ ROM コントローラ

シーケンサ ROM コントローラは MAX II コントローラ付きのインタフェースに使用されるブロックです。ROM ロードが UniPHY シーケンサ ROM へのコードのロードを開始する前には、MAX II コントローラから Nios II 命令コードをデシリアライズします。シーケンサ ROM コントローラは、4 つの主要ブロックで構成されています。

■ デシリアライズ RAM

I/O 数の制限の結果としては、フラッシュ・デバイスから読み取られた Nios II 命令コードは、HardCopy デバイスに送信する前に、MAX II コントローラによってシリアライズされます。シーケンサ ROM コントローラは DeserializeRAM ブロックにシングル・ビット・データを格納します。すべてのシングル・ビット・データ（合計の 4096×32 ビット）が格納されている場合、ROM ロードが DeserializeRAM ブロックから UniPHY シーケンサ ROM に 32 ビットのフォーマットでデータをロードします。

■ ROM_loader Init ジェネレータ

MAX II コントローラは完全にコードを読み出した後、Init ジェネレータは DeserializeRAM ブロックからの命令コードをロードするために、RAM ロードを要求します。

■ soft_reset コントロール・ブロック

このコントロール・ブロックは、UniPHY シーケンサをリセットする soft_reset_n 信号を生成します。PHY がリセット状態にあるために soft_reset_n 信号は MAX II コントローラからシーケンサ ROM へのデータ・トランザクション間が Low であることを保証します。MAX II コントローラからシーケンサ ROM へのデータ・トランザクションの完了時に soft_reset_n 信号が High にプルされます。Nios II シーケンサは、すべてのデータがシーケンサ ROM において準備ができている場合、シーケンサ ROM からデータを取得するために開始されます。

■ ROM_loader_busy_or_pll_locked コントロール・ブロック

このブロックは、MAX II コントローラに ROM_loader_busy_or_pll_locked 信号を生成します。MAX II コントローラは PLL がロックされるフラッシュ・データを読み出すために開始されます。それは、ROM ロードがシーケンサ ROM に 1 つのページからデータのロードが完了したら、後続のページからデータの読み出しを開始するために、MAX II コントローラを要求します。

図 3 に、シーケンサ ROM コントローラのインターコネクトを示しています。

図 3. シーケンサ ROM コントローラ・ブロック図

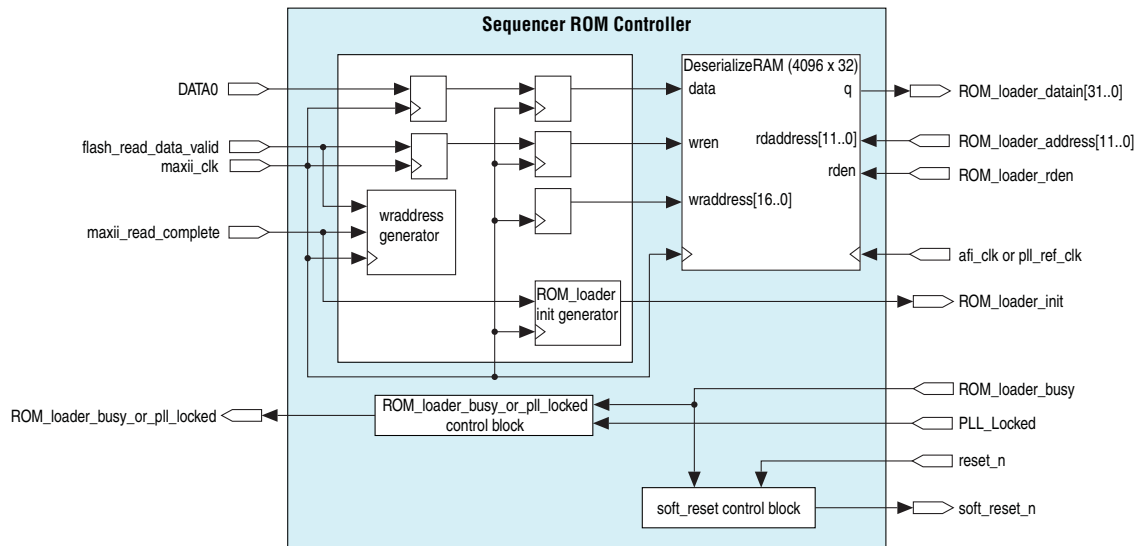


表 2 には、シーケンサ ROM コントローラ用の信号の説明を示します。

表 2. シーケンサ ROM コントローラ信号の説明

信号	説明
DATA0	MAX II コントローラからのシリアル・フラッシュ・データ。
flash_read_data_valid	信号は、読み出しデータの有効性を示します。信号は、読み出しデータが有効であることを示し、High に設定します。
maxii_clk	MAX II コントローラからのリード・クロック。
maxii_read_complete	4096x32 ビットのフラッシュ・データの 1 ページの読み出しが終了したことを示す MAX II コントローラからの信号です。
ROM_loader_busy_or_pll_locked	信号は、PLL がロックされていること、または ROM ロードがビジーであることを示すために、MAX II コントローラに送信されます。
ROM_loader_datain[31..0]	ROM ロードへのパラレル・データ。
ROM_loader_address[31..0]	ROM ロードからのリード・アドレス。
ROM_loader_rden	ROM ロードからのリード・イネーブル信号です。
ROM_loader_init	シーケンサ ROM コントローラからデータの読み出しを開始する ROM ロードを要求する ROM ロードへの信号。PLL は、PLL からの信号をロックされています。
ROM_loader_busy	信号は、ROM ロードのステータスを示します。信号は、ROM ロードがデータを読み出していることを示し、High に設定します。
PLL_locked	PLL からの PLL ロック信号です。
reset_n	グローバル・リセット信号です。

表 2. シーケンサ ROM コントローラ信号の説明

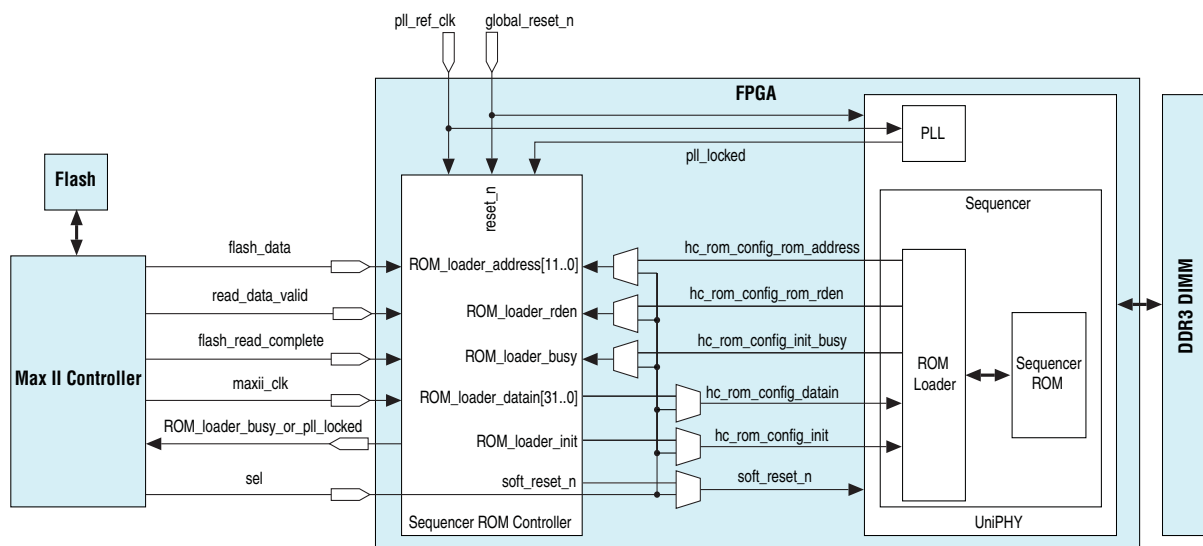
信号	説明
soft_reset_n	PLL をリセットせずに UniPHY シーケンサをリセットするリセット信号です。
afi_clk or pll_ref_clk	UniPHY または PLL 基準クロックからの信号を使用するオプションです。

FPGA におけるデザイン検証

MAX II コントローラおよびシーケンサ ROM コントローラは、Stratix IV GX チェックアウト・ボードで DDR3 SDRAM の完全なシステムで検証されます。シーケンサ ROM コントローラは DDR3 SDRAM UniPHY IP で作成されたリファレンス・デザインに統合されています。DDR3 SDRAM リファレンス・デザインは、FPGA にハーフ・レート・コントローラと 72 ビット・インタフェースで 400 MHz で動作します。

図 4 には、MAX II コントローラとシーケンサ ROM コントローラと一緒に DDR3 SDRAM UniPHY の完全なシステム接続を示します。

図 4. FPGA での UniPHY IP 付きの MAX II コントローラおよびシーケンサ ROM コントローラの接続




FPGA PLL がロックされている時、MAX II コントローラはフラッシュ・デバイスから UniPHY シーケンサ ROM に Nios II 命令コードのロードが開始されます。Nios II 命令コードが正常に外部フラッシュから UniPHY シーケンサ ROM にロードされたとき、soft_reset_n 信号は、Nios II シーケンサがシーケンサ ROM から命令コードを読み出すことができるようにディアサートされます。DDR3 SDRAM UniPHY リファレンス・デザインは、Nios II シーケンサがブートアップされると、キャリブレーションが開始されます。

MAX II デバイスと FPGA 間には、その他の接続が必要ありません。MAX II コントローラとシーケンサ ROM コントローラ・インタフェース信号は、8 つの FPGA のコンフィギュレーション・データ・ピン (DATA [0..7]) を介して接続されます。

表 3 には、コンフィギュレーション・データ・ピンを介して FPGA と MAX II デバイス間の接続を示します。

表 3. FPGA および MAX II デバイスの接続のための FPGA コンフィギュレーション・データ・ピンとインタフェース信号

FPGA コンフィギュレーション・データ・ピン	インタフェース信号	説明
DATA0	flash_read_complete	4096×32 ビットのフラッシュ・データの 1 ページの読み出しが終了したことを示す MAX II コントローラからの信号です。
DATA1	flash_data	MAX II コントローラからのシリアル・フラッシュ・データ。
DATA2	flash_data_valid	信号は、読み出しデータの有効性を示します。信号は、読み出しデータが有効であることを示し、High に設定します。
DATA[3..5]	sel[0..2]	複数のインタフェースのために MAX II コントローラからインタフェース選択の信号です。Nios II シーケンサ・インタフェースがブートアップするように選択します。
DATA6	ROM_loader_busy_or_pll_locked	PLL がロックされているか、または ROM ロダがビジーであることを示すために MAX II コントローラにシーケンサ ROM コントローラによって発行された信号です。
DATA7	maxii_clk	MAX II コントローラからクロックを読み出す信号です。

 DATA[0] 兼用ピンは、コンパイル前に、デバイスの設定で **Regular I/O** として使用するように設定されていることを確認してください。

ソフトウェアとハードウェア要求

リファレンス・デザインは、次のコンポーネントから構成されています。

- **UniPHY_HCX_Migration_DDR3_Example_Design.zip**— シーケンサ ROM が統合される DDR3 SDRAM UniPHY のリファレンス・デザイン。
- **MaxII_Controller.qar**— Nios II シーケンサに Nios II 命令コードを伝えるために使用される MAX II コントローラ。
- **MaxIIPFL.qar**— フラッシュメモリに Nios II 命令コードをプログラムするために使用される PFL (パラレル・フラッシュ・ローダ) メガファンクション。

このリファレンス・デザインは、次のソフトウェアおよびハードウェアを必要とします。

- Quartus® II ソフトウェア・バージョン 11.0
- FPP コンフィギュレーション・インタフェース付きの Stratix IV FPGA チェックアウト・ボード (F1517)
- Stratix IV デバイス : EP4SGX230KF40C3
- MAX II デバイス : EPM2210F324C3 (JTAG ポートの接続付き)
- Numonyx P30 フラッシュ・デバイス
- DDR3 SDRAM DIMM (MT9JSF12872AY-1G1BZES)

Nios II シーケンスの命令コードの変換

UniPHY Nios II シーケンスの命令コードを変換するには、以下の手順に従います。

1. リファレンス・デザインのフォルダから、**IntelFormatConversionScript.tcl** スクリプトを開きます。**.tcl** スクリプトでは、デザインに応じて、以下のコマンドで変数を変更します。

```
■ set fileid [open "<project_directory>/<variable_name>_sequencer_rom.hex" r+]
■ set new_fileid [open "<project_directory>/Converted_<variable_name>_sequencer_rom.hex" w+]
```

2. UniPHY Nios II シーケンサの命令コードが **<variable_name>_sequencer_rom.hex** ファイルとして格納されます。Intel フォーマットで **.hex** ファイルに **<variable_name>** の **<variable_name>_sequencer_rom.hex** ファイルを変換する **.tcl** スクリプトを実行します。

命令コードの 4 バイト **.hex** ファイルが各行のフラッシュに書き込まれているので、変換は 1 つのアドレス・インクリメントから 4 つのアドレス・インクリメントにアドレス指定方法の変更をイネーブルします。

3. Quartus II ソフトウェアによって、Intel フォーマットの **Converted_<variable_name>_sequencer_rom.hex** ファイルを **.pof** ファイル・フォーマットに変換します。
 - a. File メニューの **Convert Programming Files** を選択します。
 - b. **Configuration device** の **CFI_512Mb** オプションを選択します。
 - c. **Mode** には、**Fast Passive Parallel x8** を選択します。
 - d. 出力 **.pof** ファイルの名前を変更します。
 - e. **SOF Data** を削除します。Select **Add Hex Data** を選択して、**Converted_<variable_name>_sequencer_rom.hex** ファイルに追加します。
 - f. **Generate** をクリックします。

変換は、フラッシュ・デバイスをプログラムする必要があります。

PFL メガファンクションによるプログラミング・フラッシュ・メモリ

PFL メガファンクションは、JTAG インタフェースで MAX II デバイスを介してフラッシュ・メモリ・デバイスをプログラムするために使用されます。リファレンス・デザインは、PFL メガファンクションを使用して、フラッシュ・メモリ・プログラミング・モードに提供されています。リファレンス・デザインは **MaxII PFL.qar** ファイルにあります。リファレンス・デザインは、Stratix IV チェックアウト・ボードで使用される 512 M ビットのコモン・フラッシュ・インタフェース (CFI) パラレル・フラッシュに適用可能です。



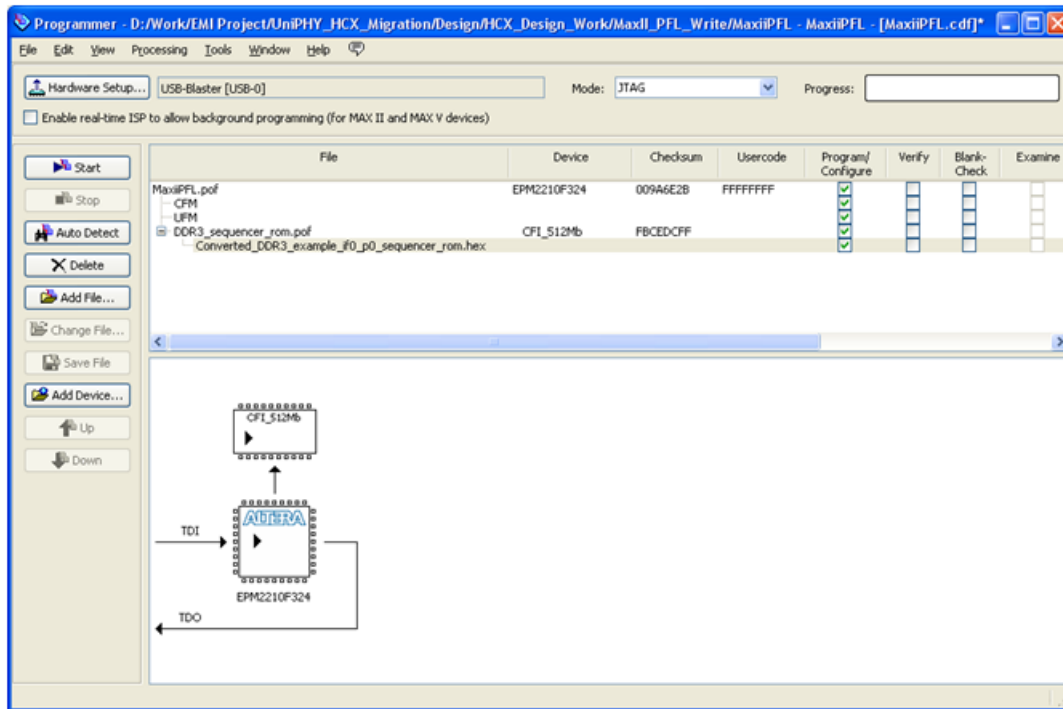
メガファンクションについて詳しくは、「[Parallel Flash Loader Megafunction User Guide](#)」を参照してください。

フラッシュ・メモリ・デバイスをプログラムするには、以下のステップに従います。

1. **MaxII_PFL.qar** ファイルを解凍して、デザインをリコンパイルします。
2. Quartus II Programmer を開き、**MaxII_PFL.pof** ファイルを追加し、**MaxII_PFL.pof** ファイルにシーケンサ ROM **.pof** を添付します。

図 5 には、デザイン・セットアップ用の Quartus II Programmer インタフェースを示します。

図 5. フラッシュ・メモリ・プログラミングのデザイン・セットアップ



3. JTAG インタフェースを介して MAX II デバイス付きの **MaxII_PFL.qar** リファレンス・デザインで提供されるフラッシュ・デバイスに変換したシーケンサ ROM **.pof** ファイルをプログラムします。

MAX II コントローラおよびシーケンサ ROM コントローラの検証

MAX II コントローラおよびシーケンサ ROM コントローラを検証するには、以下のステップに従います。

1. **MaxII_Controller.qar** ファイルを解凍して、デザインをリコンパイルします。
2. コンパイル完了後に、Quartus II Programmer で MAX II デバイスをプログラムします。
3. **UniPHY_HCX_Migration_DDR3_Example_Design.zip** file ファイルを抽出して、デザインをリコンパイルします。
4. FPGA をコンフィギュレーションします。
5. Tools メニューの **SignalTap II Logic Analyzer** を選択して、デザインを検証します。

- Tools メニューの **In-System Sources and Probes Editor** を選択して、`global_reset_n` 信号をコントロールします。デザインを開始するには、**global reset** を選択します。

改訂履歴

表 4 に、本資料の改訂履歴を示します。

表 4. 改訂履歴

日付	バージョン	変更内容
2010 年 12 月	1.0	初版。

