

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。

このアプリケーションノートでは、MAX<sup>®</sup> II および MAX V デバイスにおけるリアルタイム ISP (In-System Programmability) と ISP クランプ、そしてこれらによって Quartus II ソフトウェア、Jam<sup>™</sup> Standard Test and Programming Language (STAPL) Player と Jam STAPL Byte-Code Player での使用方法について説明します。

インシステム・プログラミング中に大部分の CPLD は、I/O ピンをトライ・ステートにして、ボード上の信号レベルの衝突を回避します。プログラミングが成功すると、デバイスはユーザー・モードに遷移し、新しいデザインが機能し始めます。

この通常のプログラミング・モードに加えて、MAX II および MAX V デバイスはリアルタイム ISP と ISP クランプ・プログラミング・モードをサポートします。

プログラミング・モードおよびその用途については、次の項で説明しています。

- 1 ページの「リアルタイム ISP」
- 4 ページの「ISP クランプ」

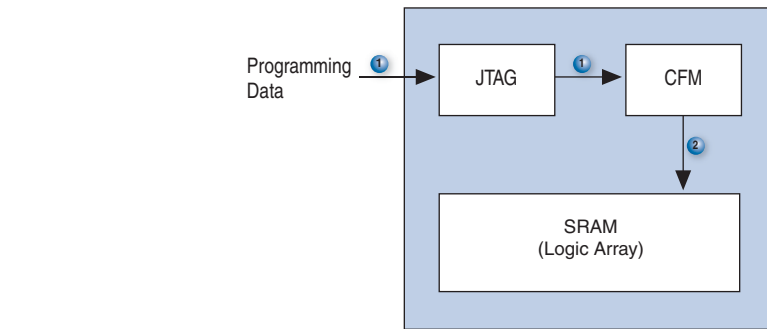
## リアルタイム ISP

リアルタイム ISP を利用すると、デバイスの動作中にも MAX II または MAX V デバイスをプログラムできます。デバイスがパワーサイクルした後の場合のみ、既存のデザインは新規のデザインに置き換わります。この機能により、システム全体の動作に影響を与えることなく、デバイスのイン・フィールド・アップデートを実行することができます。

## リアルタイム ISP の動作

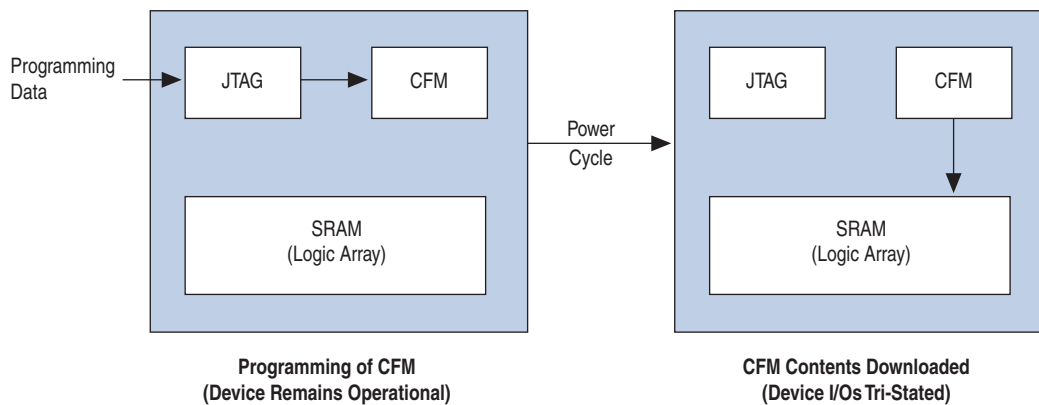
通常の ISP 動作の場合、コンフィギュレーション・フラッシュ・メモリ (CFM) から SRAM への新規デザイン・データのダウンロードは、CFM プログrammingの完了後に開始されます。プログラミングおよびダウンロードの動作中に、I/O ピンはトライ・ステートの状態に維持されます。ダウンロードが完了すると、デバイスはリセットされユーザー・モード動作に入ります。図 1 に、通常のプログラミングのフローを示します。



図 1. 通常の ISP 動作



リアルタイム ISP モードでは、CFM のプログラミング中にも、ユーザー・フラッシュ・メモリ (UFM)、プログラマブル・ロジック、および I/O ピンは動作状態のままです。t<sub>CONFIG</sub> 時間が経過した後、デバイスはユーザー・モードに入ります。図 2 に、リアルタイム ISP のフローを示します。

図 2. リアルタイム ISP の動作



- 
 MAX II デバイスの t<sub>CONFIG</sub> 値については、「MAX II デバイス・ハンドブック」の「*DC and Switching Characteristics*」の章を参照してください。
- 
 MAX V デバイスの t<sub>CONFIG</sub> 値については、「MAX V デバイス・ハンドブック」の「*DC and Switching Characteristics for MAX V Devices*」の章を参照してください。

## Quartus II ソフトウェアを使用したリアルタイム ISP

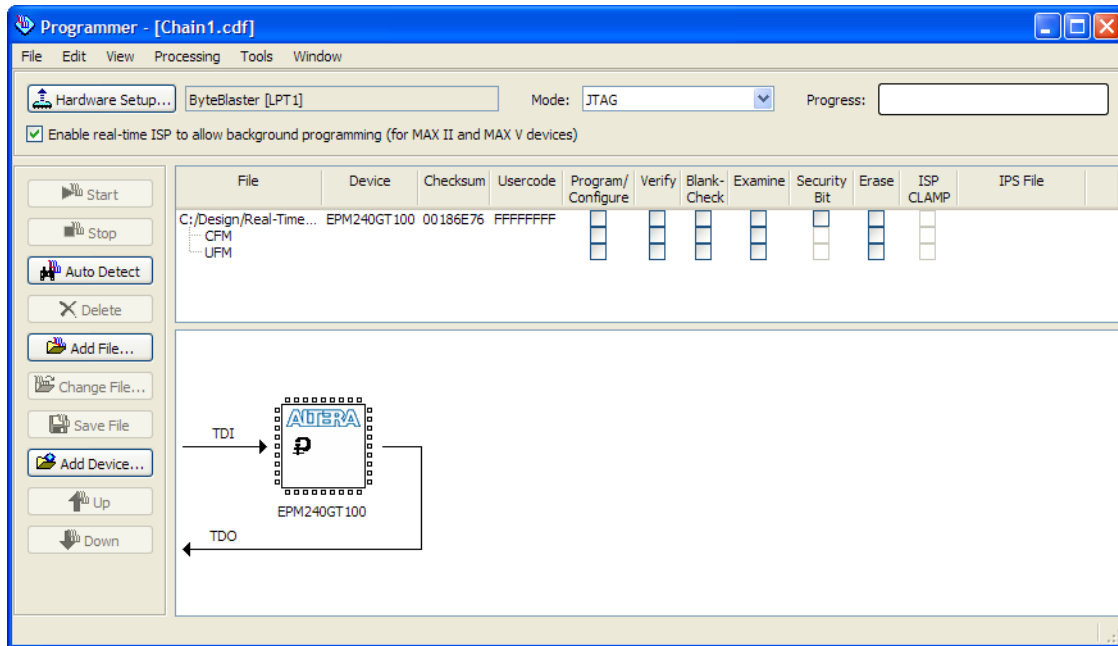
Quartus II ソフトウェアは、リアルタイム ISP および ISP クランプ機能をサポートするために、下記のプログラミング・ファイル・フォーマットを生成します。

- Programmer Object File (.pof)
- JEDEC JESD71 STAPL Format File (.jam)
- JAM Byte Code File (.jbc)

これらのプログラミング・ファイルは、Quartus II Programmer で使用できます。また、.jam および .jbc ファイルはほかのプログラミング・ツールでも使用できます。

Quartus II Programmer で MAX II または MAX V デバイスをプログラミングする前に、リアルタイム ISP 機能をイネーブルする必要があります。この機能をイネーブルするには、図 3 に示すように、Quartus II Programmer ウィンドウで **Enable real-time ISP to allow background programming (for MAX II and MAX V devices)** をオンにします。

図 3. Quartus II Programmer ウィンドウでのリアルタイム ISP オプション



Quartus II Programmer が .pof、.jam、または .jbc ファイルのプログラミングを開始すると、MAX II または MAX V デバイスはリアルタイム・モードに入ります。

## Jam STAPL および Jam STAPL Byte-Code Players を使用したリアルタイム ISP

.pof から作成された .jam または .jbc ファイルを使用すると、Jam STAPL または Jam STAPL Byte-Code Player を使用したリアルタイム ISP モードで MAX II または MAX V デバイスをプログラムできます。

.jam ファイルおよび Jam STAPL Player を使用したリアルタイム ISP の場合は、コマンドライン・プロンプトで以下のコマンドを入力します。

```
jp_23 -aprogram -ddo_real_time_isp=1 <file_name>.jam
```

.jbc ファイルおよび Jam STAPL Byte-Code Player を使用したリアルタイム ISP の場合は、コマンドライン・プロンプトで以下のコマンドを入力します。

```
jbi_22 -aprogram -ddo_real_time_isp=1 <file_name>.jbc
```

Player の実行可能ファイルの名前は、Player のバージョンによって異なります。最新バージョンの Jam STAPL および JAM STAPL Byte-Code Player は、アルテラ・ウェブサイト ([www.altera.co.jp](http://www.altera.co.jp)) からダウンロードできます。

## ISP クランプ

MAX II または MAX V デバイスが通常の ISP 動作に入ると、すべての I/O ピンがトライ・ステートとなり、内部ウィーク・プルアップ抵抗で  $V_{CCIO}$  にプルアップされます。ただし、デバイスが ISP 動作中のときには、デバイスの I/O ピンをトライ・ステートにしてはならない状況もあります。例えば、動作中のシステムで特定の信号（出力イネーブル信号またはチップイネーブル信号など）が I/O ピンの一部を使用したり、I/O ピンが High または Low ロジック・レベルになること、あるいはデバイスが ISP モードのときには現在のステートを維持するよう要求することもあります。

MAX II および MAX V デバイスの ISP クランプ機能により、デバイスをプログラムするときに、Quartus II ソフトウェアでデバイスの各 I/O ピンを指定のステートに保持することができます。デバイスを ISP クランプ・モードで正しくプログラムすると、これらの I/O ピンが解放され、デバイスが新しいデザインに従って機能します。

デバイスがプログラミングの動作中と示すため、この機能を使用することが出来ません。デバイスが ISP クランプ・モードに入ると、特定のピンをデバイスのユーザー・モード動作のステートと異なるステートに設定します。

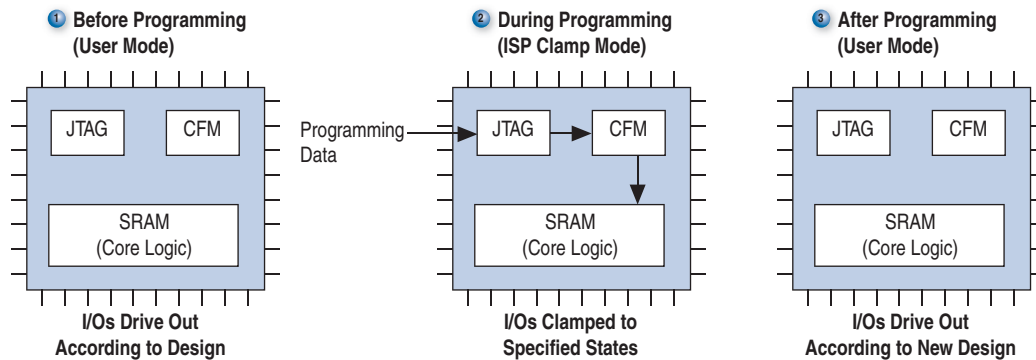
### ISP クランプの動作方式

I/O ピンをトライ・ステート（デフォルト）、High、または Low に設定でき、またはピンの既存のステートをサンプリングして、デバイスが ISP クランプ動作中にピンをそのステートに保持することもできます。Quartus II ソフトウェアはユーザーの設定に基づいて、各 I/O ピンのバウンダリ・スキャン・レジスタにスキャンする値を割り当てます。これらの値は、デバイスをプログラミングするときにクランプされるピンの状態を決定する。ISP クランプ機能を使用すると、I/O がトライ・ステート値にクランプされている場合でも、プログラミング中にウィーク・プルアップ抵抗はディセーブルされます。

I/O ピンをクランプする前に、まず SAMPLE/PRELOAD JTAG 命令を実行して、値をバウンダリ・スキャン・レジスタにロードします。そして、EXTEST 命令を実行して、I/O ピンを SAMPLE/PRELOAD 実行中にバウンダリ・スキャン・レジスタにロードされた特定の値にクランプします。


デバイスが ISP クランプ・モードに入ったときにピンの既存のステートをサンプリングし、ピンをそのステートに保持するように選択した場合は、信号が安定したステートになることを確認しなければなりません。サンプル・セットアップ・タイムは、ダウンロード・ケーブルやソフトウェアだけでなく、TCK 周波数にも依存し、制御することはできないので、安定したステート信号が必要です。トグルする信号、または長期間にわたってスタティックでない信号をサンプリングした場合、キャプチャした値は正しくないことがあります。図 4 に、ISP クランプ動作を示します。

図 4. ISP クランプ動作



## Quartus II ソフトウェアにおける ISP クランプの使用

ISP クランプ機能を使用するには、I/O ピンのステートを定義しなければなりません。Quartus II ソフトウェアでは、I/O Pin State File (.ips) または Assignment Editor を使用してピンのステートをクランプします。

 ISP クランプ機能を使用する前に、**Enable real-time ISP to allow background programming (for MAX II and MAX V devices)** をオフにします。

### .ips の使用

デバイスが ISP クランプ動作中の場合は、Assignment Editor で設定をコンフィギュレーションしてデザインを再コンパイルすることなく、.ips でピンのクランプ・ステートを指定できます。.ips ファイルは、.ips クランプ動作中のすべてのピンのステートを定義します。

新しい .ips ファイルを作成して、ファイルにピンのステートを定義するか、または既存の .ips ファイルを使用します。デザインが同じデバイスとパッケージをターゲットとする限り、作成されたファイルはどのデザインでもデバイスをプログラムするのに使用できます。.ips ファイルを .pof ファイルと共に使用します。この POF ファイルにはデバイスをプログラムするためのプログラミング・データが含まれています。

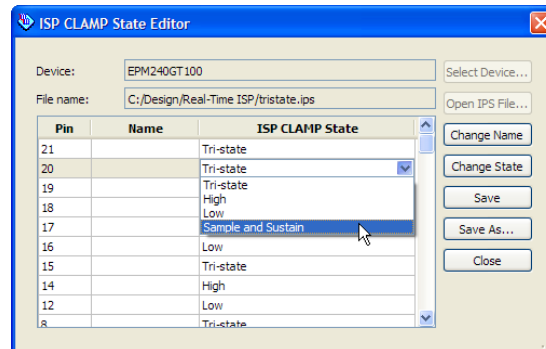
### .ips の作成

.ips を作成するには、次のステップを実行します。

1. Tools メニューで、**Programmer** をクリックして、Quartus II Programmer ウィンドウを開きます。あるいは、ツールバーでの **Programmer** をクリックします。
2. Quartus II Programmer ウィンドウで **Add File** をクリックして、プログラミング・ファイル (.pof、.jam、または .jbc) を Programmer ウィンドウに追加します。
3. リストからプログラミング・ファイルを選択します。そして、Edit メニューで、**ISP CLAMP State Editor** をクリックします。
4. ISP Clamp State Editor で、デザインのピンのステートを指定します。クランプ・ステートは、トライ・ステート、High、Low、またはサンプル/持続の 4 つから選択します。デフォルトでは、すべてのピンはトライ・ステートに設定されています。図 5 に、ISP Clamp State Editor を示します。

5. 変更後に **.ips** ファイルを保存します。

図 5. ISP Clamp State Editor



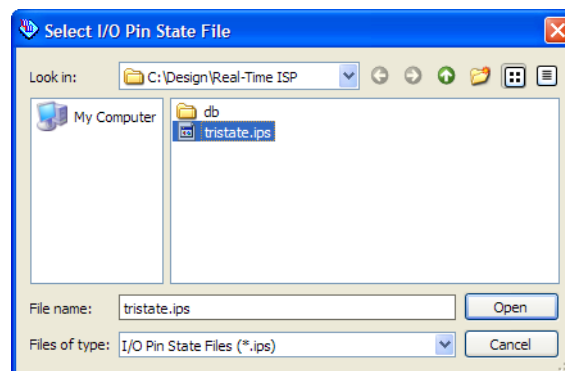
また、ISP Clamp State Editor を起動して新しい **.ips** を作成するには、**File** メニューで **Create/Update** をポイントして **Create/Update IPS File** をクリックします。

### .ips の使用

Quartus II Programmer で使用する **.ips** を指定するには、次のステップを実行してください。

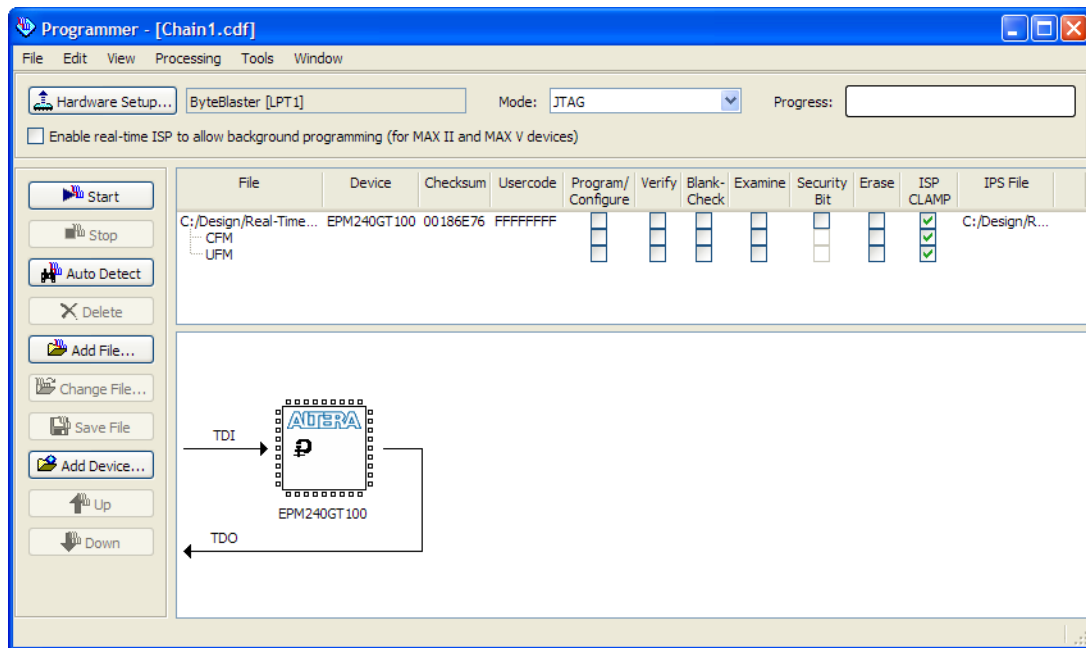
1. プログラミング・ファイルのロウを右クリックして **Add IPS File** を選択します。また、プログラミング・ファイルのロウをクリックし、そして、**Edit** メニューでの **Add IPS File** をクリックして、図 6 に示す **Select I/O Pin State File** ダイアログ・ボックスを表示させます。
2. プロジェクトの **.ips** ファイルを選択し、**Open** をクリックします。


図 6. Select I/O Pin State File メニュー



3. 図 7 に示すように、選択した **.ips** ファイルは Quartus II Programmer ウィンドウにリストされます。

図 7. 特定の .ips ファイルを表示する Quartus II Programmer ウィンドウ



 デバイスのプログラミングを開始する前に、**ISP CLAMP** をオンにしてください。

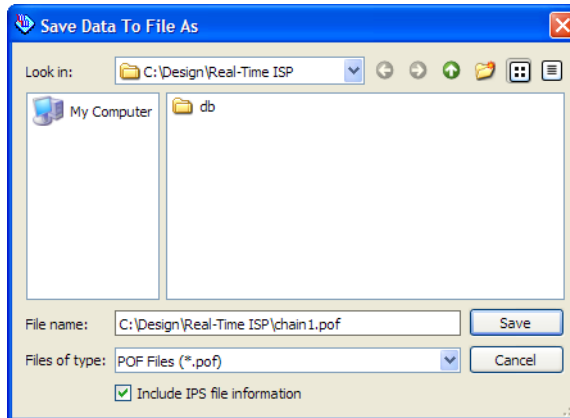
### プログラミング・ファイルへの IPS ファイル情報の保存


.ips 内のピン・ステート情報を .pof に保存することで、2つのファイルが必要となる場合が回避できます。デバイスを ISP クランプ・モードでプログラムするには、プログラミング・ファイルのみが必要です。また、このプログラミング・ファイルを使用して、ISP クランプ用のピン・ステート情報を含む .jam および .jbc ファイルを作成することができます。

.ips のピン・ステート情報をプログラミング・ファイルに保存するには、以下のステップを実行します。

1. Quartus II Programmer で、プログラミング・ファイルおよび .ips ファイルを追加します。
2. **Save File** をクリックします。図 8 に示すように、**Save Data To File As** ダイアログ・ボックスが表示されます。
3. ファイル名を指定し、**Include IPS file information** をオンにします。
4. **Save** をクリックします。

図 8. Save Data To File As ダイアログ・ボックス



 I/O ピン・ステート情報が保存された **.pof** は Quartus II ソフトウェアの ISP クランプ動作のみをサポートします。サードパーティ・ツールで ISP クランプが必要な場合は、**.jam** または **.jbc** ファイルを使用する必要があります。

**ISP CLAMP** がオンにしたデバイスをプログラムするとき、Quartus II Programmer はまず **.ips** ファイルを検索します。ソフトウェアは、**.ips** が見つからなかった場合のみ、**.pof** からピン・ステート情報を検索します。

### Assignment Editor でのピン・ステートの定義

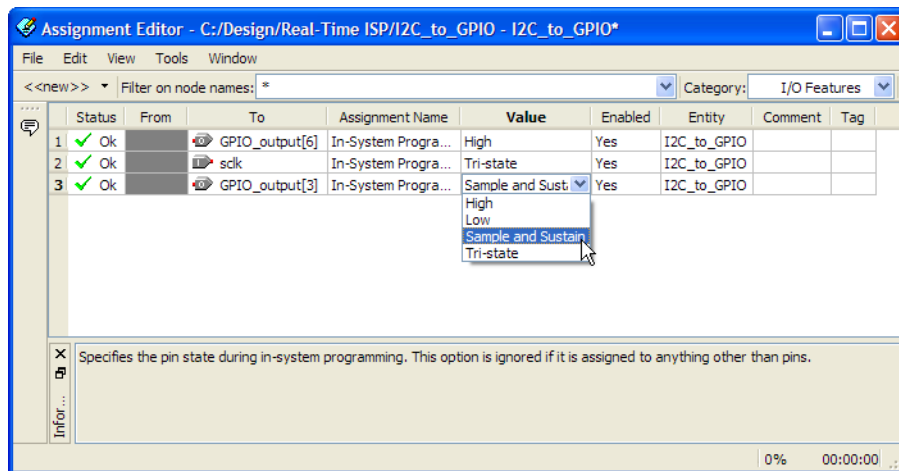
また、Assignment Editor でピン・ステートを定義してデザインをコンパイルすることもできます。生成されたプログラミング・ファイルにはすべてのピン・ステート情報が含まれます。

Assignment Editor でピン・ステートを定義するには、次のステップを実行してください。

1. ツールバーで **Start Analysis and Synthesis** をクリックします。
2. Assignments メニューで、**Assignment Editor** をクリックします。
3. **Category** リストから、**I/O Features** を選択します。
4. **To** のカラムで、デザインが ISP クランプ・モードに入るときにクランプしようとするピンを指定します。ピンの選択に **Node Finder** を使用します。
5. ステート値を設定しようとするピンを指定した後、**Assignment Name** カラムでのすべてのピンに対して **In-System Programming Clamp State** を選択します。
6. **Value** カラムで、各ピンのステートを指定します。I/O ピンを High、Low、トライ・ステートに設定でき、またはピンのステートをサンプリングして保持することもできます。デフォルトでは、デバイスが ISP クランプ・モードに入るときに、ピンはトライ・ステートになります。図 9 に、Assignment Editor でピンのステートを定義する方法を示します。
7. アサインメントを保存し、デザインを再コンパイルします。



## 図 9. Assignment Editor



デザインを再コンパイルした後、ISP クランプ・ステート情報は **.pof** に保存されます。また **.qsf** ファイルで設定を表示することができます。

### Quartus II ソフトウェアにおける ISP クランプの実行

Quartus II Programmer で、デバイスをプログラムする前に **ISP CLAMP** をオンにします。Quartus II Programmer では、Assignment Editor で設定した **.pof** より **.ips** 内の設定値が優先されるため、Quartus II Programmer に **.ips** を追加しないでください。**.pof** から作成された **.jam** および **.jbc** ファイルには、ピン・ステート情報が含まれます。

### **.jam** または **.jbc** ファイルを使用した ISP クランプ

ISP クランプ用の **.jam** および **.jbc** ファイルにはすべてのピン・ステート情報が含まれ、**.ips** ファイルを必要としません。**.jam** または **.jbc** ファイルを作成するには、常にピン・ステート情報を持つ **.pof** を使用してください。Assignment Editor を介してピン・ステート情報を **.pof** に格納するか、または **.pof** にピン・ステート情報を保存することができます。**.jam** または **.jbc** ファイルは、それぞれの Jam STAPL または Jam STAPL Byte-Code Player、あるいは Quartus II Programmer で使用できます。

## 改訂履歴

表 1 に、本資料の改訂履歴を示します。

表 1. 改訂履歴

日付	バージョン	変更内容
2010 年 12 月	1.0	初版

