

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。

MI151019-1.5

### はじめに

インシステム・プログラミング中に大部分の CPLD は、入力 / 出力 (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・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・モードとは別に、MAX® II デバイスはリアルタイム・インシステム・プログラマビリティ (ISP) と ISP クランプ・プログラミング・モードをサポートし、ISP 中の I/O およびデバイス動作制御が可能になります。この章では、これら 2 つの機能と Quartus® II ソフトウェアにおける使用方法、さらに Jam™ STAPL (Standard Test and Programming Language) および Jam STAPL Byte-Code Player について説明します。

- 12-1 ページの「リアルタイム ISP」
- 12-5 ページの「ISP クランプ」

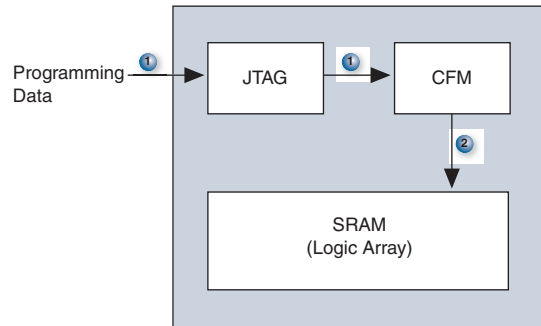
### リアルタイム ISP

リアルタイム ISP を利用すると、デバイスの動作中に MAX II デバイスをプログラムできます。デバイスのパワー・サイクル（つまり、パワーダウンして再びパワーアップする）が存在する場合、新規デザインのみが既存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、MAX II 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

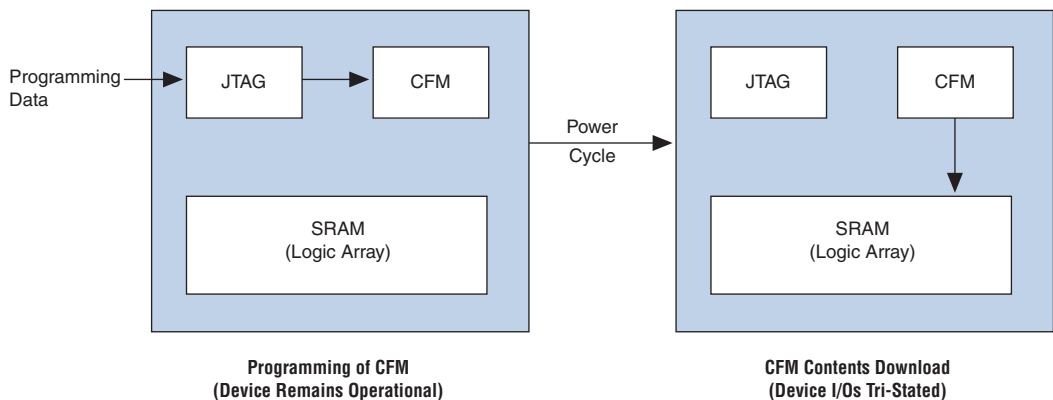
通常の ISP 動作の場合、コンフィギュレーション・フラッシュ・メモリ (CFM) から SRAM への新規デザイン・データのダウンロードは、CFM プログラミングの完了後に開始されます。CFM をプログラミングし、続いて CFM データを SRAM へダウンロードするプロセスの間、I/O ピンはトライ・ステートの状態に維持されます。SRAM への CFM ダウンロードが完了すると、デバイスはリセットされユーザ・モード動作に入ります。図 12-1 に、通常のプログラミングのフローを示します。

図 12-1. 通常 ISP 動作での MAX II デバイス



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・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。CFM のプログラミングが成功した後、CFM の内容は SRAM にダウンロードされません。その代わりに、デバイスはパワー・サイクルが発生するのを待ちます。t<sub>CONFIG</sub> 時間が経過した後、通常のパワーアップ・シーケンス (パワーアップ時に CFM が SRAM にダウンロードされる) が発生し、デバイスはユーザ・モードに入ります。図 12-2 に、リアルタイム ISP のフローを示します。

図 12-2. リアルタイム ISP 動作



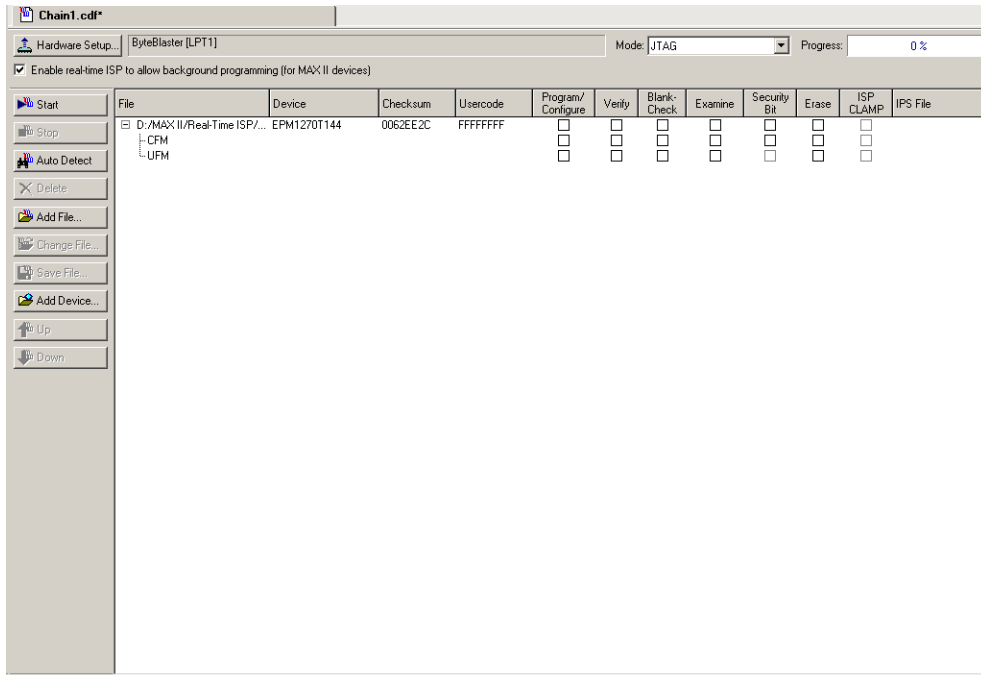
MAX II デバイスの t<sub>CONFIG</sub> 値については、「MAX II デバイス・ハンドブック」の「DC およびスイッチング特性」の章を参照してください。

## Quartus II ソフトウェアを使用したリアルタイム ISP

Quartus II ソフトウェアで生成され、これらの 2 つの機能をサポートするプログラミング・ファイルのフォーマットは、Quartus II プログラマで使用される Programmer Object File (.pof) と、Quartus II プログラマまたはその他のプログラミング・ツールのいずれかで使用される Jam ファイル (.jam) および Jam Byte-Code ファイル (.jbc) です。

Quartus II プログラマで MAX II デバイスをプログラムする前に、必ずこの機能をイネーブルにしてください。Quartus II プログラマ・ウィンドウから **Enable real-time ISP to allow background programming (for MAX II devices)** オプションを選択して、リアルタイム ISP 機能をイネーブルにできます。図 12-3 を参照してください。

図 12-3. Quartus II プログラマ・ウィンドウのリアルタイム ISP オプション



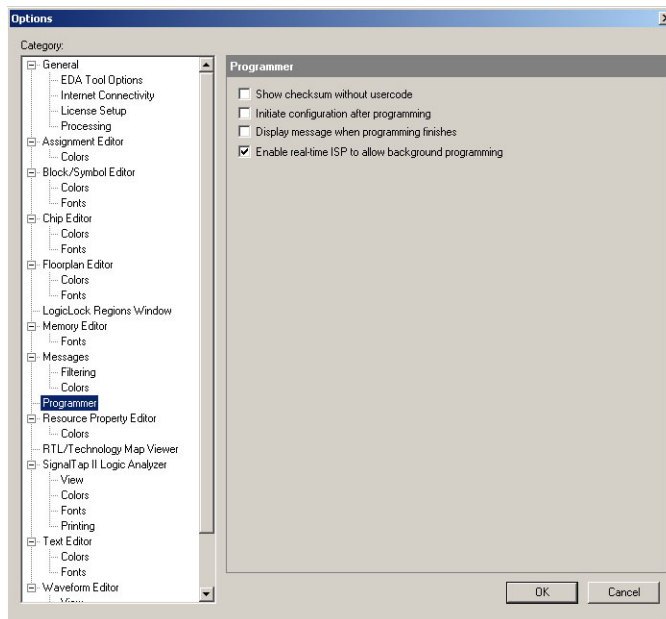
また、Quartus II ソフトウェアのリアルタイム ISP 機能は、以下の方法でもイネーブルにできます。

1. Tools メニューの **Options** をクリックします。

2. **Category** から **Programmer** を選択します。
3. **Enable real-time ISP to allow background programming (for MAX II devices)** をオンにして、**OK** をクリックします。Quartus II Programmer が3つのプログラミング・ファイルのいずれかを使用して MAX II デバイスのプログラミングを開始すると、MAX II デバイスはリアルタイム ISP モードに入ります。

図 12-4 に、Options メニューの Programmer オプションを示します。

#### 図 12-4. Options メニューの Programmer オプション



### Jam および JBC 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 または JBC Player を使用したリアルタイム ISP モードで MAX II デバイスをプログラムできます。

Jam ファイルおよび Jam 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 JBC 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。アルテラ・ウェブサイト ([www.altera.co.jp](http://www.altera.co.jp)) から最新バージョンの Jam および JBC Player をダウンロードしてください。

## ISP クランプ

MAX II 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・ステートにしなければ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 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、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。Quartus II ソフトウェアでステートを設定できます。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、新しいデザインに従って機能します。

この機能を使用して、デバイスが ISP クランプ・モードに入ったときに特定のピンを特定のステート（デバイスがユーザ・モードになっているときと異なるステート）に設定することによって、デバイスがプログラミング中であること、およびプログラミングが完了したことを示すことができます。

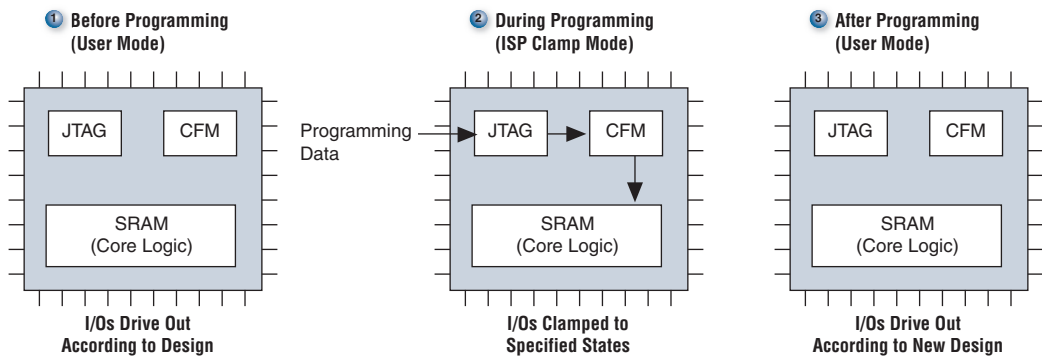
## ISP クランプの動作

ISP クランプ機能を使用すると、I/O ピンをトライ・ステート（デフォルト）、High、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タに保持することもできます。ソフトウェアはユーザの設定に基づいて、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決定します。これによって、デバイス・プログラミングの進行中にクランプされるピンのステータが決まります。ISP クランプ機能が使用されると I/O がトライ・ステータ値にクランプされても、ウィーク I/O プルアップ抵抗はプログラミング時に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、適切な値をバウンダリ・スキャン・レジスタにロードします。バウンダリ・スキャン・レジスタに適切な値をロードした後、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タをサンプリングし、ピンをそのステータ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タ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、安定したステータ信号が必要です。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 12-5 に、ISP クランプ動作を示します。

図 12-5. 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 ソフトウェアでピン・ステートを定義するには、2つの方法があります。以下のいずれかで定義できます。

- I/O ピン・ステート・ファイル (.ips) を使用するか、または
- Assignment Editor を使用して、ピンのクランプ・ステートを設定する。

### IPS ファイルの使用

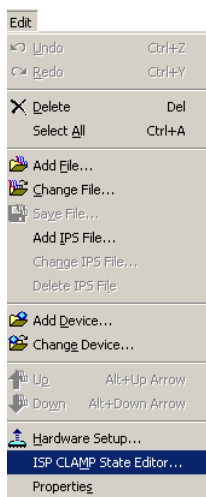
#### IPS ファイルの作成

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、ピンのクランプ・ステートを指定できます。まず、新しい I/O ピン・ステート・ファイル (.ips) ファイルを作成し、ファイルにピンのステートを定義する必要があります。IPS ファイルは、デバイスが ISP 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。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 ファイルは、POF ファイルとともに使用する必要があります、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 ファイルを作成するには、以下を実行します。

1. ツールバーで **Programmer** をクリックするか、Tools メニューの **Programmer** をクリックして、Quartus II Programmer ウィンドウを開きます。
2. プログラマで **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. プログラマでプログラミング・ファイルをクリックし (ロウ全体が強調表示される)、Edit メニューの **ISP CLAMP State Editor** をクリックします。図 12-6 を参照してください。

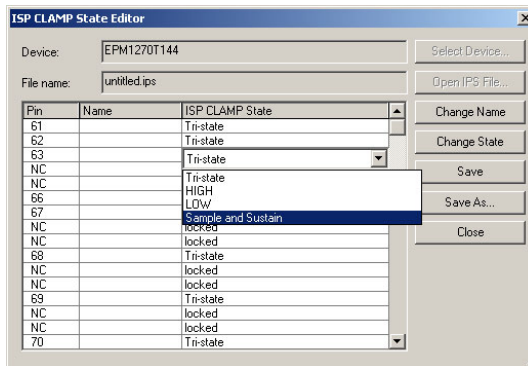
図 12-6. Edit メニュー



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。
5. 変更後に IPS ファイルを保存します。

図 12-7 に、ISP Clamp State Editor を示します。File メニューで **Create/Update > Create/Update IPS File** をクリックして、ISP Clamp State Editor を開き、新規 IPS ファイルを作成することもできます。

## 図 12-7. ISP Clamp State Editor



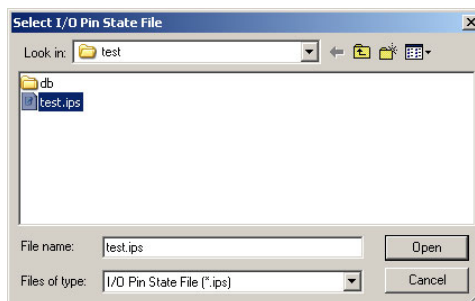
### IPS ファイルの使用

Quartus II Programmer では、以下のステップを実行して、使用する IPS ファイルを指定する必要があります。

1. IPS File カラムの下のセルをダブル・クリックすると、**Select I/O Pin State File** メニューが表示されます。
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。

また、プログラミング・ファイルをクリックし（ロウ全体が強調表示される）、Edit メニューの **Add IPS File** をクリックして、図 12-8 に示す Select I/O Pin State File ダイアログ・ボックスを開くこともできます。

## 図 12-8. Select I/O Pin State File メニュー

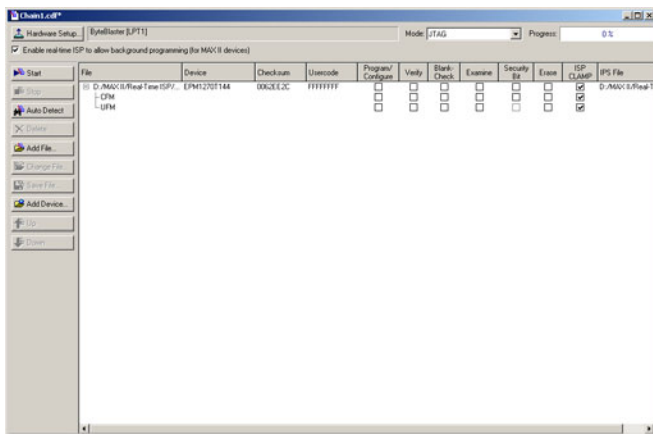


- 図 12-9 に示すように、選択した IPS ファイルは Quartus II Programmer ウィンドウにリストされます。



デバイスのプログラミングを開始する前に、ISP CLAMP チェック・ボックスがチェックされていることを確認してください。

図 12-9. 特定の IPS ファイルを表示する Quartus II Programmer ウィンドウ



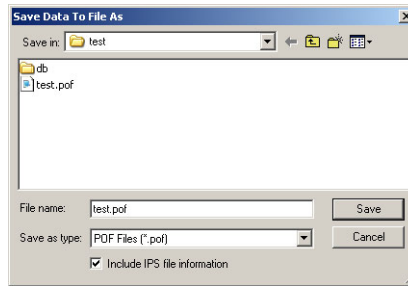
#### プログラミング・ファイルへの IPS ファイル情報の保存

IPS ファイルのピン・ステート情報は、ファイルを 2 つ用意する必要なく POF に保存する 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。このプログラミング・ファイルは、ISP クランプ用の Jam および JBC ファイルの作成にも使用されるので、Jam または JBC ファイルにはピン・ステート情報が含まれます。以下のステップを実行すると、IPS ファイルのピン・ステート情報がプログラミング・ファイルに保存されます。

- Programmer ウィンドウにプログラミング・ファイルを追加します。
- プログラマに IPS ファイルを追加します。
- Programmer ウィンドウまたは Edit メニューから **Save File** をクリックすると、**Save Data To File As** ダイアログ・ボックスが表示されます。図 12-10 を参照してください。
- ファイル名を入力し、**Include IPS file information** ボックスをチェックし、**Save** をクリックします。

IPS 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、サードパーティのプログラミング・ツールの IPS 動作はサポートしません。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。

#### 図 12-10. Save Data To File As メニュー



ISP Clamp ボックスをチェックしてデバイスをプログラムする場合、Quartus II Programmer は最初に IPS ファイルを探します。IPS ファイルが見つからなければ、その場合に限り POF 内でピン・ステート情報を調べます。

#### Assignment Editor におけるピン・ステートの定義

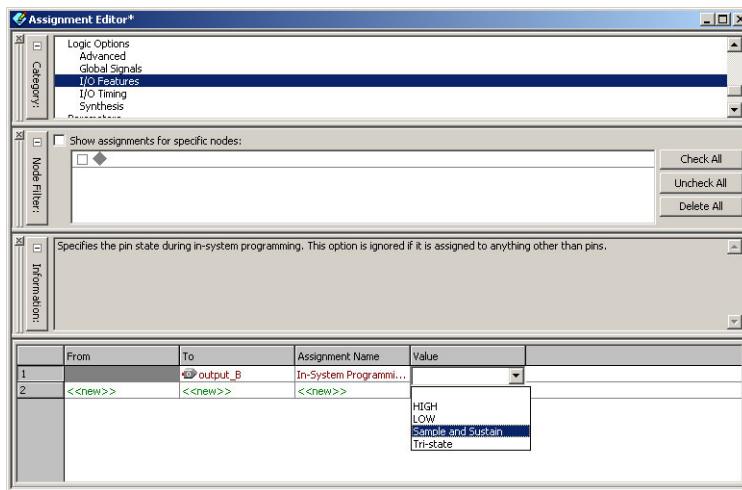
ピン・ステートを定義するもう 1 つの方法は、Assignment Editor を使用することです。Assignment Editor でピン・ステートを定義し、デザインをコンパイルすると、生成されるプログラミング・ファイルにはすべてのピン・ステート情報が格納されます。以下に、Assignment Editor のステートを示します。

1. ツールバーの **Start Analysis and Synthesis** をクリックします。
2. Assignments メニューの **Assignment Editor** をクリックします。
3. Assignment Editor の **Category** から **I/O Features** を選択します。
4. **To** カラムに、デバイスが ISP クランプ・モードのときにクランプするすべてのピンをリスト・ダウンします。Node Finder を使用すると、簡単にピンを選択できます。
5. ステート値を設定するピンをリスト・ダウンしたら、Assignment Name のすべてのピンに対して **In-System Programming Clamp State** を選択します。

6. Value で、各ピンに対するステートを定義します。ピンは、High、Low、トライ・ステート、またはサンプル / 持続にクランプ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、ピンはトライ・ステートになります。

図 12-11 は、Assignment Editor でピンのステートを定義する方法を示します。

図 12-11. Assignment Editor



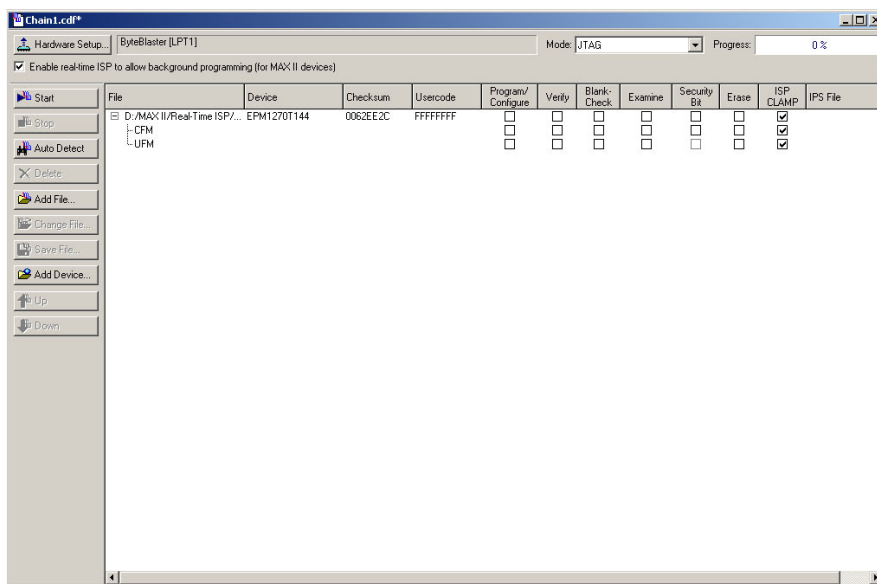
7. アサインメントを保存し、デザインを再コンパイルします。

デザインを再コンパイルすると、ISP クランプ・ステート情報が POF に格納されます。Quartus II 設定ファイル (.qsf) で、設定を確認することもできます。

### Quartus II Programmer の ISP クランプの実行

デバイスをプログラムする前に、Quartus II Programmer ウィンドウで、ISP Clamp チェック・ボックスがチェックされていることを確認します。Quartus II Programmer は、ユーザが Assignment Editor に設定した値 (POF に格納される) ではなく、IPS ファイルに指定された値を使用するため、IPS ファイルを追加しないでください。図 12-12 に、Quartus II Programmer ウィンドウと ISP Clamp チェック・ボックスを示します。POF を使用して作成された Jam および JBC ファイルには、ピン・ステート情報が格納されます。

図 12-12. Quartus II Programmer ウィンドウと ISP Clamp チェック・ボックス



## Jam/JBC ファイルを使用した ISP クランプ

ISP クランプに使用される Jam または JBC ファイルには、すべてのピン・ステート情報が格納されているため、IPS ファイルを必要としません。必ずピン・ステート情報を含む POF ファイルを使用して、Jam または JBC ファイルを作成してください。ピン・ステート情報は、Assignment Editor を使用して POF に格納できます。または、前述のとおり、ピン・ステート方法を POF に格納します。Jam または JBC ファイルは、それぞれ Quartus II Programmer、あるいは Jam または JBC Player で使用できます。

## まとめ

MAX II デバイスでリアルタイム ISP および ISP クランプ機能を利用すると、デバイスのプログラミング中にデバイスの I/O ピンを特定のステートに設定できます。リアルタイム ISP を利用すると、システムの機能に影響を与えることなく、いつでも MAX II デバイスをプログラムできます。ISP クランプ機能を利用すると、デバイスをプログラムする間にデバイスの I/O ピンを特定のステートに保持できます。

## 参考資料

この章では以下のドキュメントを参照しています。

- 「MAX II デバイス・ハンドブック」の「DC およびスイッチング特性」の章

## 改訂履歴

表 12-1 に、本資料の改訂履歴を示します。

日付 & ドキュメント・バージョン	変更内容	概要
2007 年 12 月 v1.5	「参考資料」の項を追加。	—
2006 年 12 月 v1.4	改訂履歴を追加。	—
2006 年 2 月 v1.3	<ul style="list-style-type: none"> <li>● 「Quartus II ソフトウェアを使用したリアルタイム ISP」の項を更新。</li> <li>● 図 12-3 を追加。</li> <li>● 図 12-9 および 12-12 を更新。</li> </ul>	—
2005 年 6 月 v1.2	ISP クランプの動作の項の第 1 パラグラフを更新。	—
2005 年 1 月 v1.1	13 章から変更。内容の変更はなし。	—