

# Intel Cyclone 10 GX FPGA を構成する QSPI フラッシュの有効化

## About this document

### Scope and purpose

AN229767 は、Intel Quartus Prime Pro Generic Flash Programmer ツールを使用して、アクティブシリアル (AS) モードで Intel Cyclone 10 GX FPGA を構成するために S25HS512T QSPI フラッシュを有効にする方法を説明します。同じ方法は、それぞれのフラッシュデータシートに従ってフラッシュを構成することにより、他の QSPI フラッシュファミリを有効にする場合にも適用されます。

## Table of contents

<b>About this document</b> .....	<b>1</b>
<b>Table of contents</b> .....	<b>1</b>
<b>1 はじめに</b> .....	<b>2</b>
<b>2 背景情報</b> .....	<b>3</b>
<b>3 手順</b> .....	<b>4</b>
3.1 *.sof FPGA 構成ファイルを生成.....	4
3.2 新しい Configuration Device を定義.....	4
3.2.1 新しい構成デバイスを追加.....	4
3.2.2 初期化フローテンプレートの変更.....	7
3.2.3 プログラムフローテンプレートの変更.....	12
3.2.4 消去フローテンプレートの変更.....	15
3.3 .sof ファイルを変換して.jic ファイルを生成.....	17
3.4 .jic 構成データを QSPI フラッシュデバイスにプログラム.....	20
<b>4 結論</b> .....	<b>25</b>
<b>5 参考資料</b> .....	<b>26</b>
<b>改訂履歴</b> .....	<b>27</b>

## はじめに

### 1 はじめに

このドキュメントは、読者が Intel FPGA の開発に精通していることを前提に書かれています。

Quartus Prime 19.1 以降のバージョンの Intel の Generic Flash Programmer ツールは、サードパーティの QSPI フラッシュを FPGA 構成デバイスとして構成する方法を提供します。

このアプリケーションノートでは、Generic Flash Programmer ツールを使用して FPGA 構成ビットストリームファイルを S25HS512T QSPI フラッシュにロードする方法について説明します。その後、QSPI フラッシュは、アクティブシリアル (AS) 構成を介して構成データをターゲット FPGA にロードします。

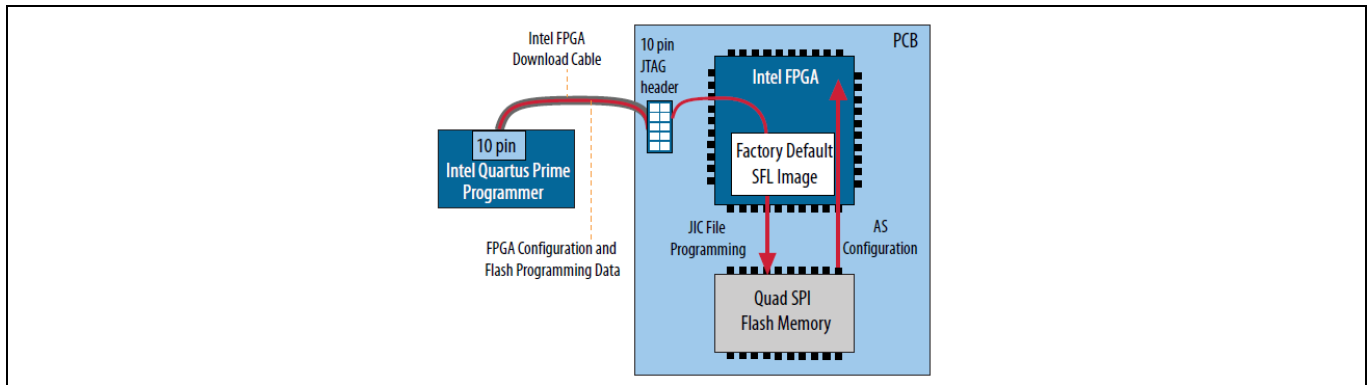
このアプリケーションノートで紹介されている手順は、Quartus Prime Pro 19.2 を使用して、クアッド SPI インターフェースを備えた Semper™ Flash を搭載した Intel Cyclone 10 GX FPGA 開発キット (特に S25HS512T) で検証されました。この手順は、他のすべての Quartus Prime バージョン 19.1 以降に適用される予定です。この手順は、それぞれのデータシートに従ってフラッシュと FPGA デバイスを構成することにより、他の QSPI フラッシュファミリと Intel Arria 10 FPGA デバイスにも適用する必要があります。

## 背景情報

### 2 背景情報

Intel Generic Flash Programmer は、Intel Stratix 10、Intel Arria 10、および Intel Cyclone 10 GX FPGA デバイスをサポートします。ただし、Intel Stratix 10 デバイスの場合は、Secure Device Manager (SDM) ファームウェアがフラッシュプログラミングフローを制御します。これらのフローは変更できません。したがって、このアプリケーションノートで紹介されている手順は、Intel Stratix 10 FPGA デバイスには適用されません。

次の図に、Generic Flash Programmer 構成のデータフローを赤で示します。



## 手順

### 3 手順

Quartus Prime **Programmer** の **Convert Programming File** ダイアログボックスから、Generic Flash Programmer の設定とコントロールにアクセスできます。

Convert Programming File ダイアログボックスを使用した Generic Flash Programming には、次の高レベルの手順が含まれます。

1. \*.sof FPGA 構成ファイルを生成します。
2. 新しい Configuration Device を定義します。
3. \*.sof ファイルを\*.jic ファイルに変換します。このファイルは、構成データを格納するために QSPI フラッシュをプログラムするために使用されます。
4. Intel Quartus Prime Programmer を使用し、Intel FPGA ダウンロードケーブルを接続して、\*.jic 構成データを QSPI フラッシュデバイスにプログラムします。

#### 3.1 \*.sof FPGA 構成ファイルを生成

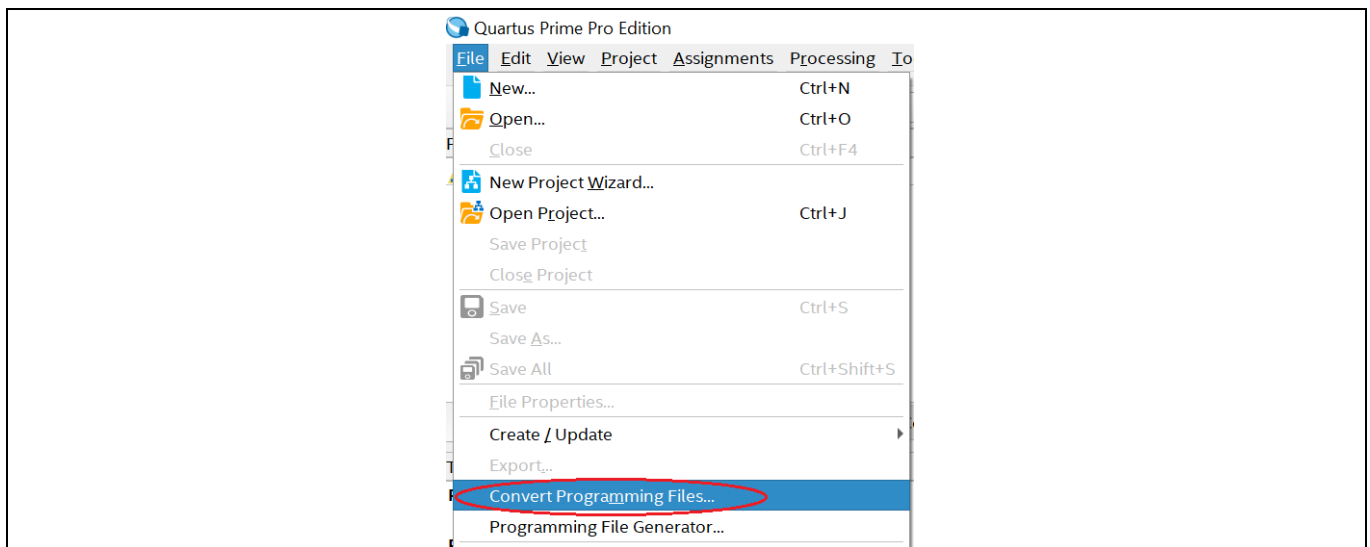
この手順は、このドキュメントではカバーされていません。\*.sof ファイルがすでに生成されていることを前提としています。\*.sof ファイルの生成については、関連する Intel ドキュメントを参照してください。

#### 3.2 新しい Configuration Device を定義

新しい Configuration Device を定義する際、Quartus Prime ソフトウェアは、設定のコレクションを、選択した **Device database directory** の場所にある\*.xml ファイルに保存します。この構成は 1 回だけ実行する必要があります。新しい Configuration Device テンプレートは、Quartus Prime にリストされているデフォルトの構成デバイステンプレートと同じように再利用できます。

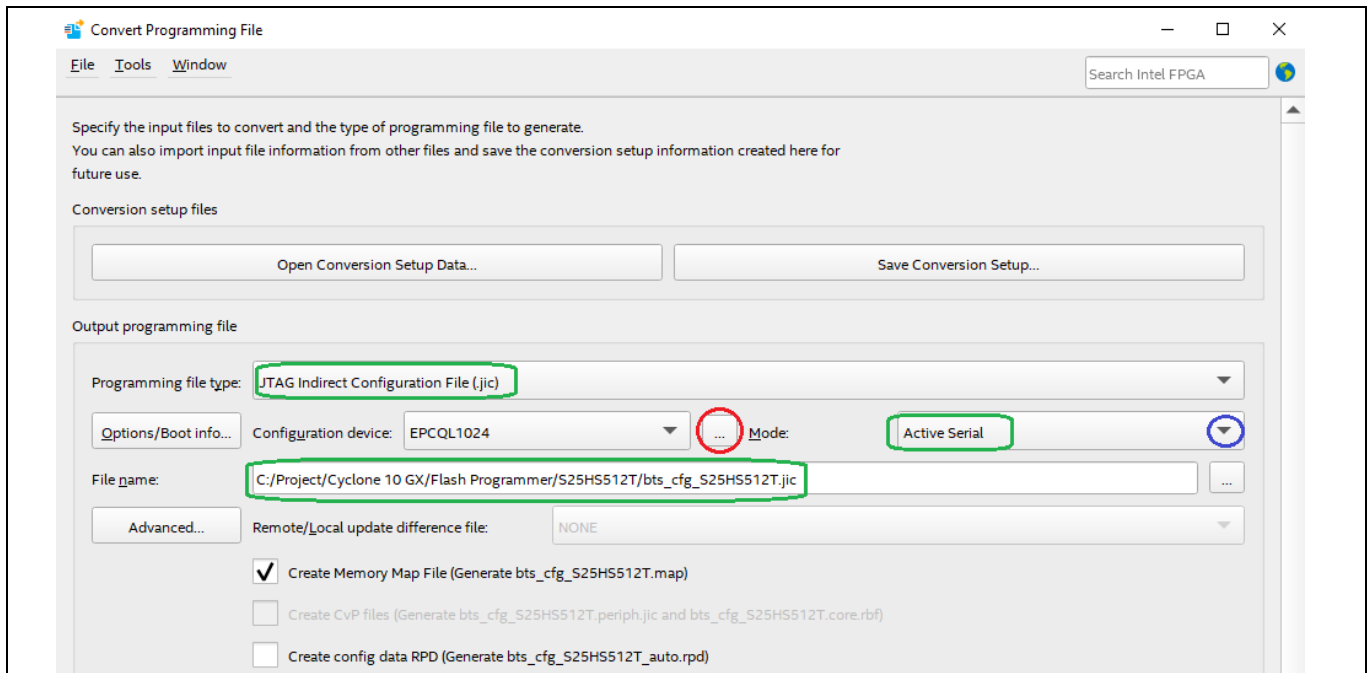
##### 3.2.1 新しい構成デバイスを追加

1. Quartus Prime Pro ソフトウェアを開き、**File > Convert Programming Files...**を選択してください。



## 手順

2. Convert Programming File 画面で以下を実行してください。



a) プログラミングファイルタイプで\*.jic ファイルタイプを選択してください。

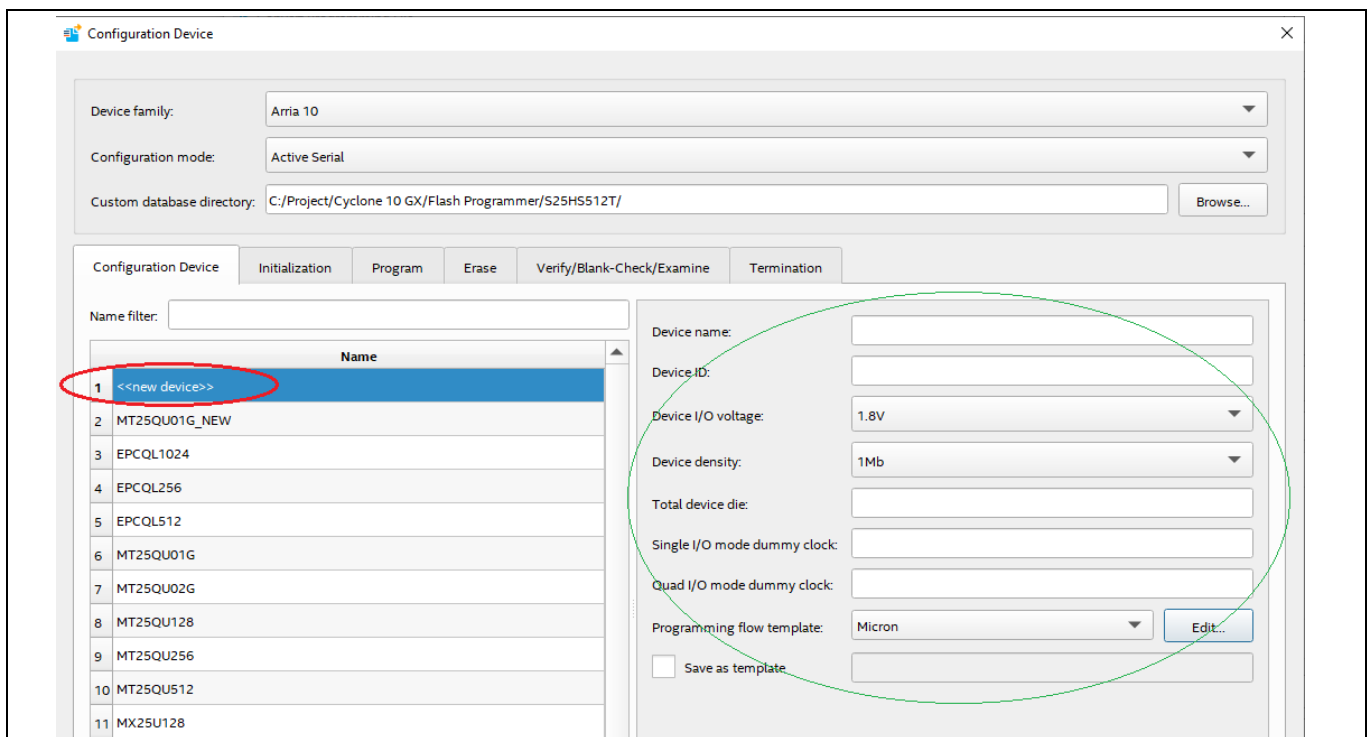
Note: Mode オプションは、\*.jic が選択されている場合にのみ選択できます。

b) File name フィールドで、変換された\*.jic ファイルを保存する場所を指定してください。

c) 青い円のプルダウンメニューをクリックし、**Active Serial** を選択してください。

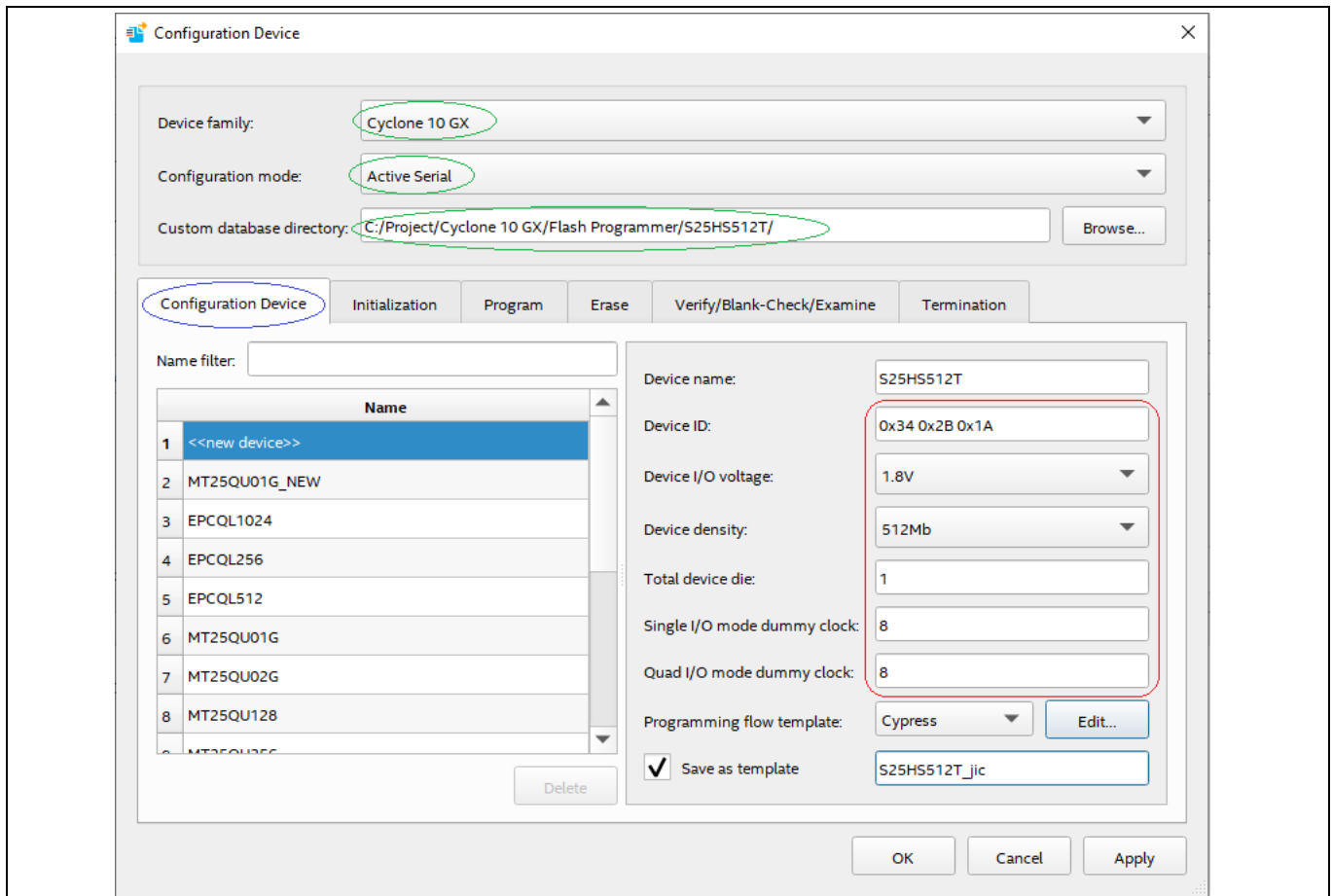
d) 赤い丸をクリックしてください。

3. << new device >>をクリックしてください。緑の円で構成可能なフィールドが表示されます。



## 手順

4. 以下の変更を行ってください。



- FPGA デバイスに応じて FPGA の Device family と Configuration mode を設定してください。
- \*.xml ファイルが保存される Custom database directory を指定してください。
- QSPI フラッシュデバイスのデータシートに従って、フラッシュデバイスのパラメータを設定してください。
  - デバイス ID
  - デバイスの I/O 電圧
  - デバイス密度
  - 総デバイスダイ
  - シングル I/O モードダミークロック
  - クワッド I/O モードダミークロック

例えば、S25HS512T デバイスの場合、S25HS512T データシートからこれらのパラメータを入力してください。

5. **Apply** をクリックしてください。新しいデバイス”S25HS512T”が左側のリストに自動的に追加されません。

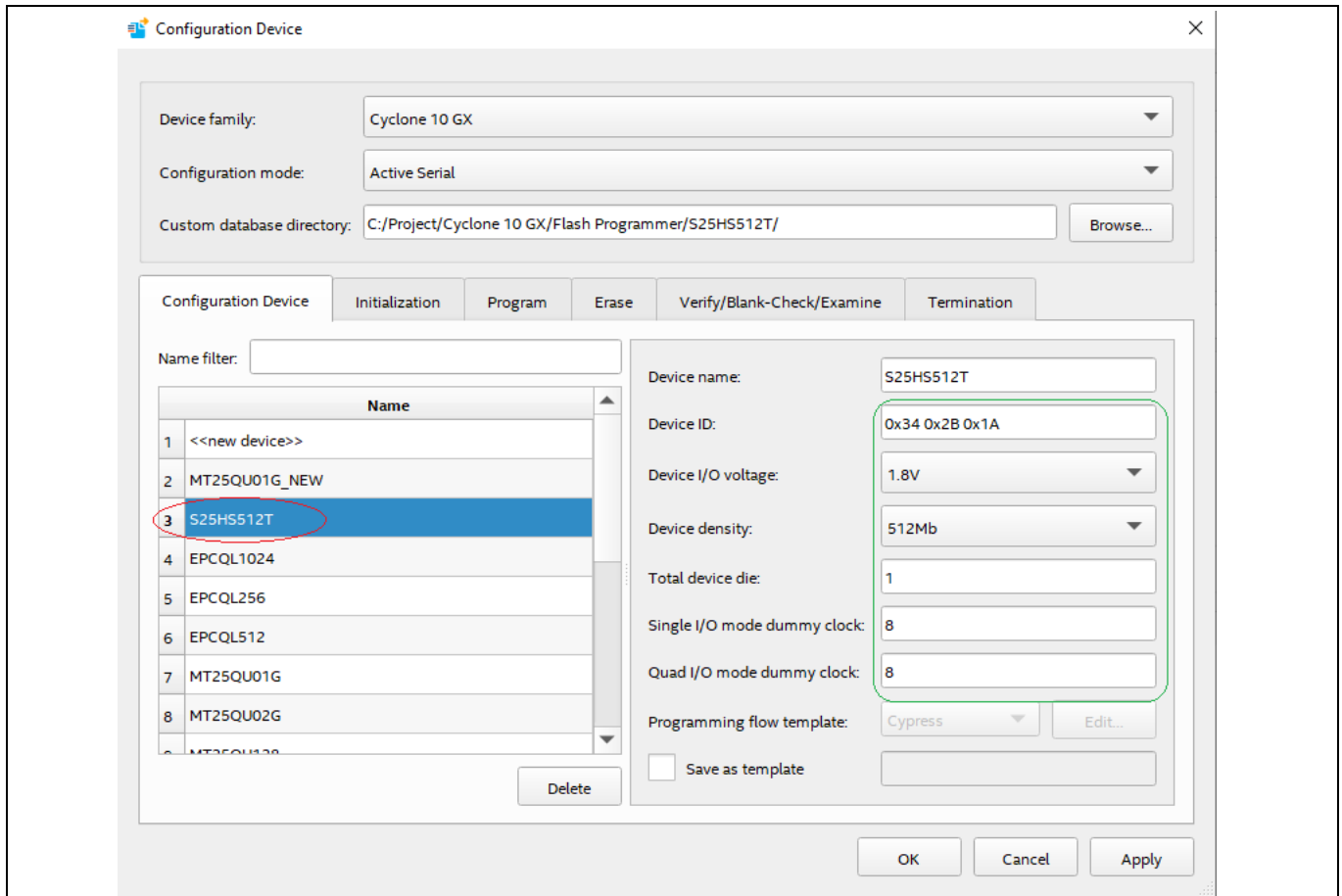
## Note:

- フラッシュパラメータの設定では、**Device name** とテンプレート名(”Save as template”)を同じにすることはできません。それ以外の場合は、**Apply** ボタンをクリックすると、”テンプレート名 xxxxxx がデバイス名 xxxxxx と競合します”というエラーメッセージが表示されます。

## 手順

- Device ID は、0xAB 0xCD 0xEF 形式の 3 バイトの 16 進データです。

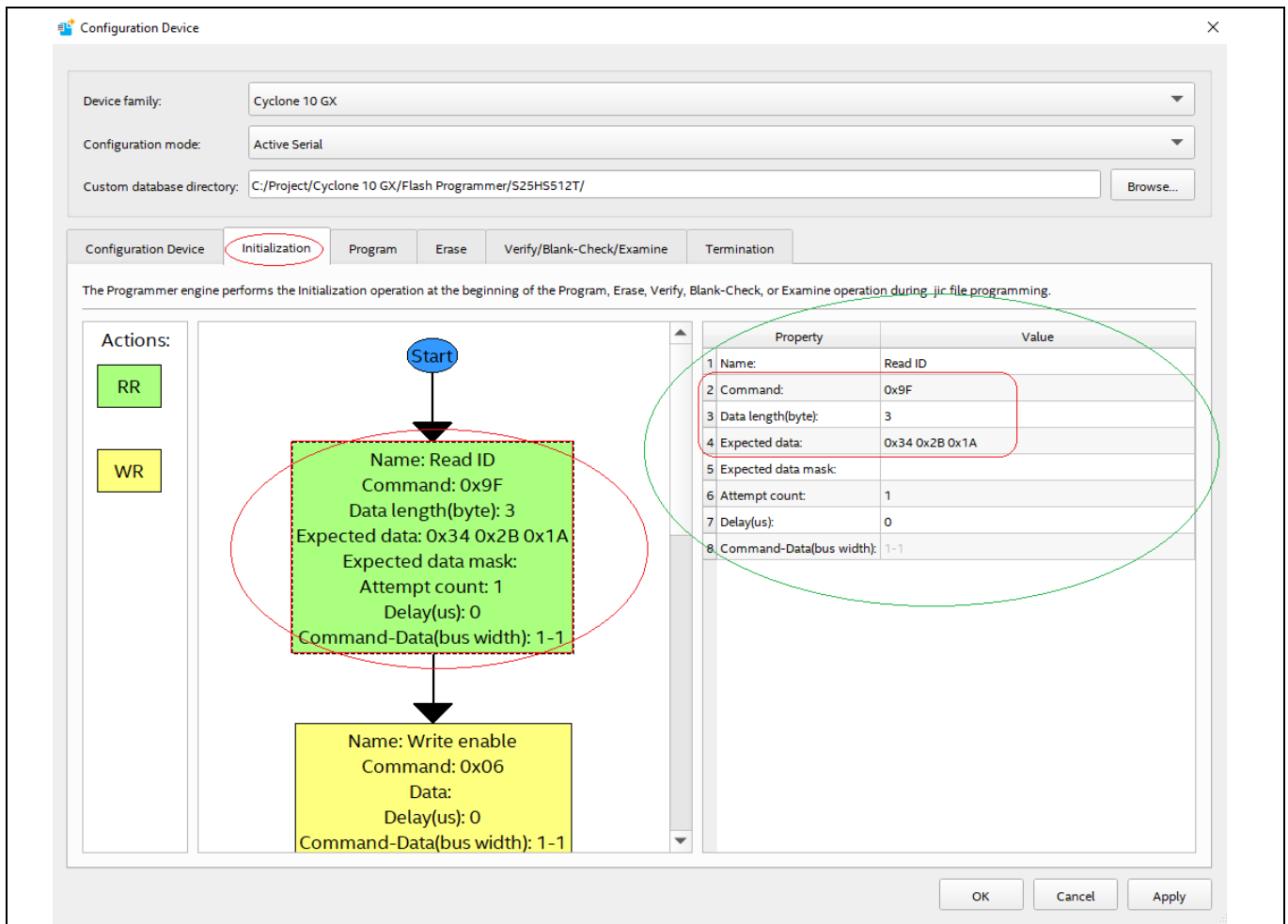
## 3.2.2 初期化フローテンプレートの変更



1. 作成した Configuration Device (例として S25HS512T) を選択し、**Initialization** タブをクリックしてください。

Initialization タブには、フラッシュ初期化フローテンプレートが表示されます。テンプレートのアクションを追加、削除、または変更できます。これは、デフォルトのフローテンプレートに基づく S25HS512T デバイスの変更を示します。必要に応じて、独自の初期化フローテンプレートを定義できます。

## 手順

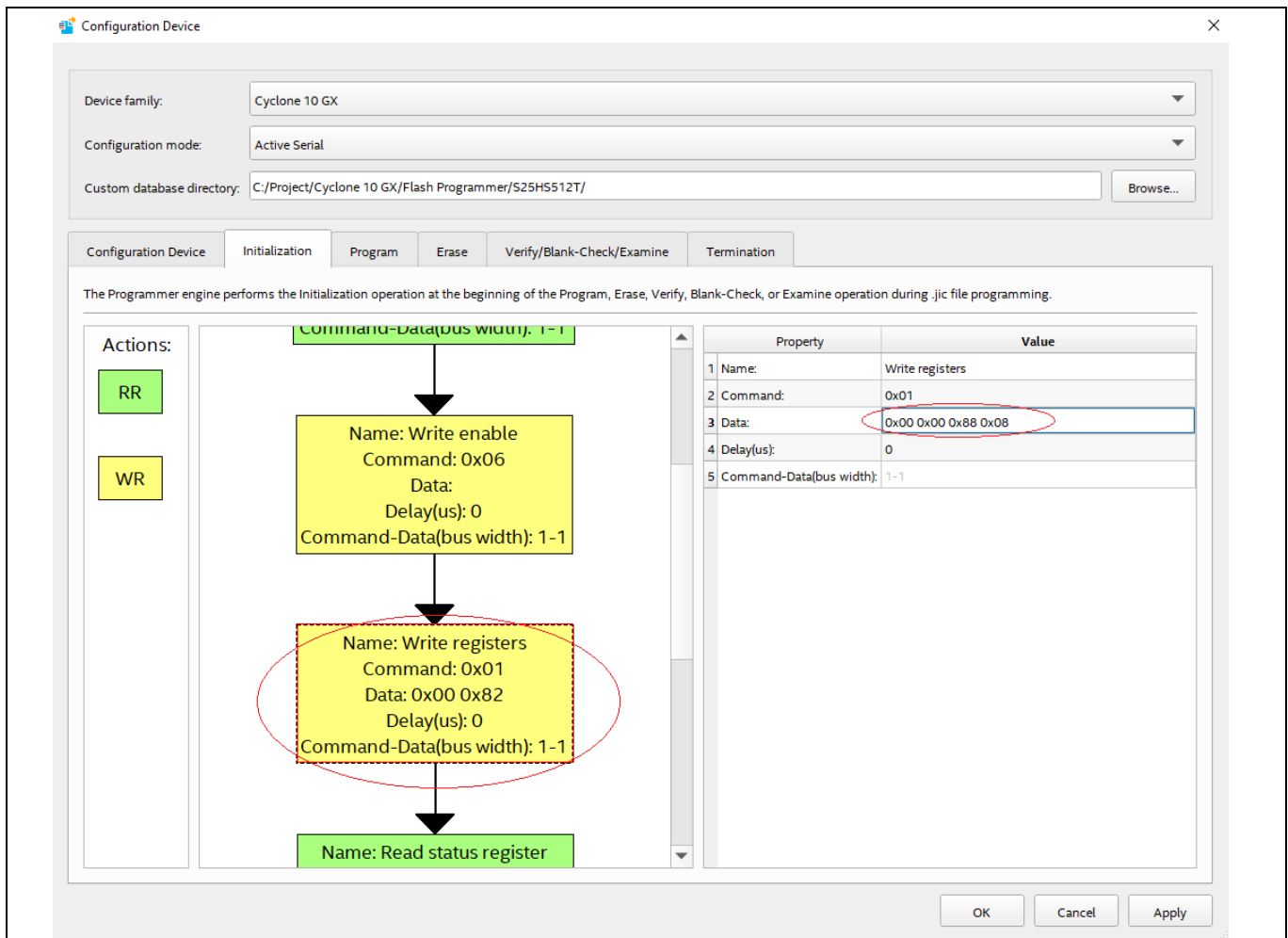


2. ウィンドウの左側で **Read ID** 動作をクリックしてください。フラッシュデバイスのデータシートに従って、右側 (緑色の円) のプロパティを構成します。
3. 動作プロパティを設定したら、**Apply** をクリックしてください。

デフォルトの”Write enable”動作は、S25HS512Tの動作と同じです。この例では、このフィールドは変更されていません。



## 手順



4. 左側のウィンドウを下にスクロールして、**Write registers** 動作をクリックしてください。プロパティを編集して、フラッシュレジスタを設定してください。

この例では、S25HS512T を 4 バイトのアドレスモードに構成し、セクターアーキテクチャを均一な 256KB に設定してください。

S25HS512T 書き込みレジスタコマンドシーケンスは次のとおりです。書き込みレジスタコマンド (0x01) + STR1 (ステータスレジスタ-1) 値 + CFR1 (コンフィギュレーションレジスタ-1) 値 + CFR2 (コンフィギュレーションレジスタ-2) 値 + CFR3 (コンフィギュレーションレジスタ-3) 値。

STR1 のデフォルト値は 0x00 です。CFR1 のデフォルトは 0x00 です。

CFR2 のデフォルト値は 0x08 です。4 バイトアドレスモードの場合は、CFR2 [7]を'1'に設定してください。CFR2 の新しい値は 0x88 です。

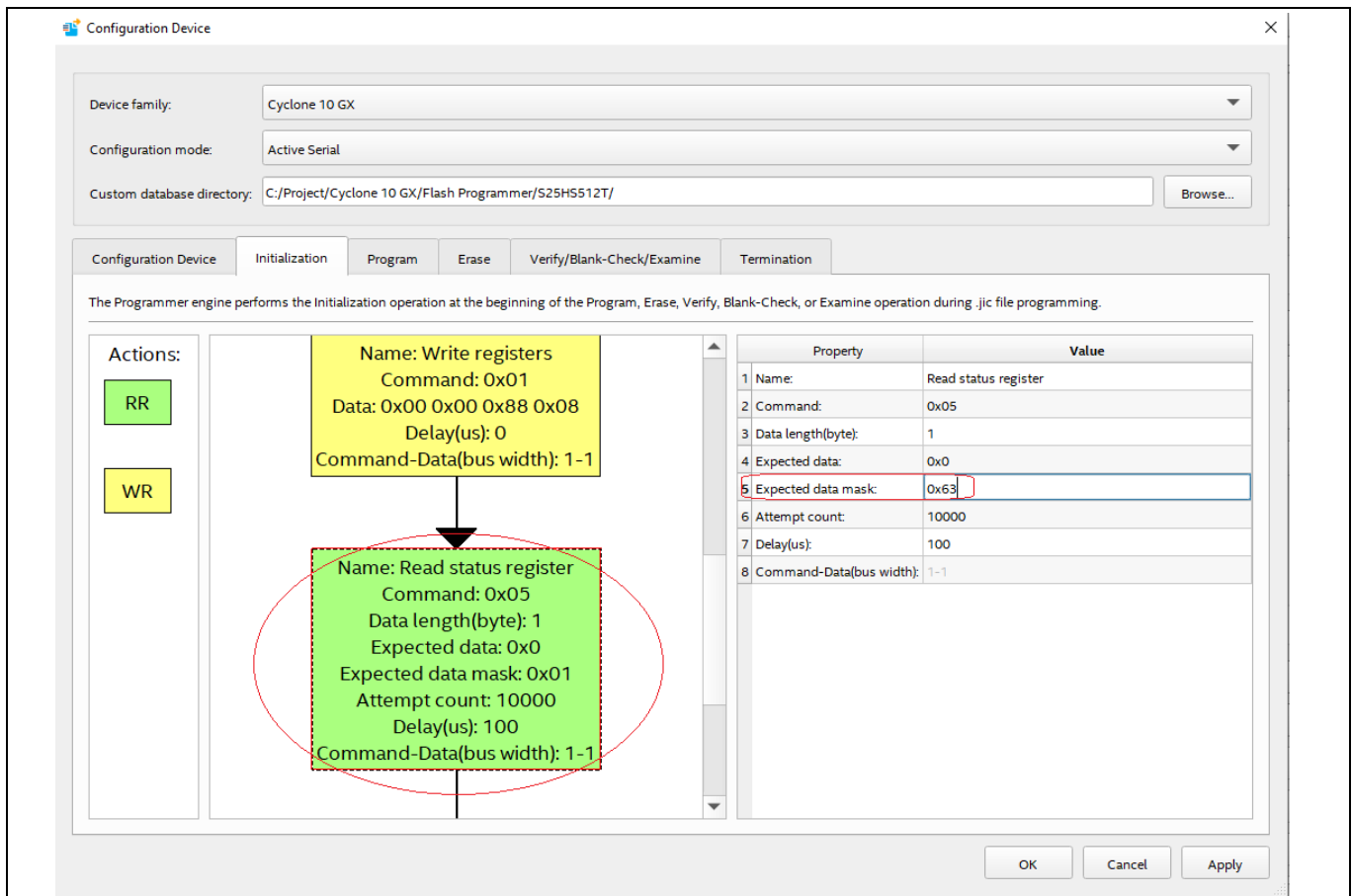
CFR3 のデフォルト値は 0x00 (ハイブリッドセクターアーキテクチャ 4KB + 256KB セクター) です。均一な 256KB セクターの場合は、CFR3 [3]を'1'に設定してください。CFR3 の新しい値は 0x08 です。

したがって、0x01 コマンドに続くデータは (右側の赤い円の上のスクリーンショットに示されているように) 0x00 0x00 0x88 0x08 です。レジスタの詳細については、S25HS512T のデータシートを参照してください。

## 手順

## Note:

- Intel Cyclone 10 GX および Arria10 デバイスには、4 バイトのアドレス指定フラッシュメモリが必要です。フラッシュは 4 バイトアドレス指定をサポートする必要があり、4 バイトアドレスモードに設定する必要があります。
- デフォルトの消去フローテンプレートは、均一な 256KB セクターアーキテクチャ用です。消去フローを単純化するためには、初期化フローでフラッシュを均一な 256KB セクターアーキテクチャとして設定します。そうでない場合、消去フローはフラッシュアレイを完全に消去することはできません。これはエラーメッセージなしで発生する可能性があることに注意してください。

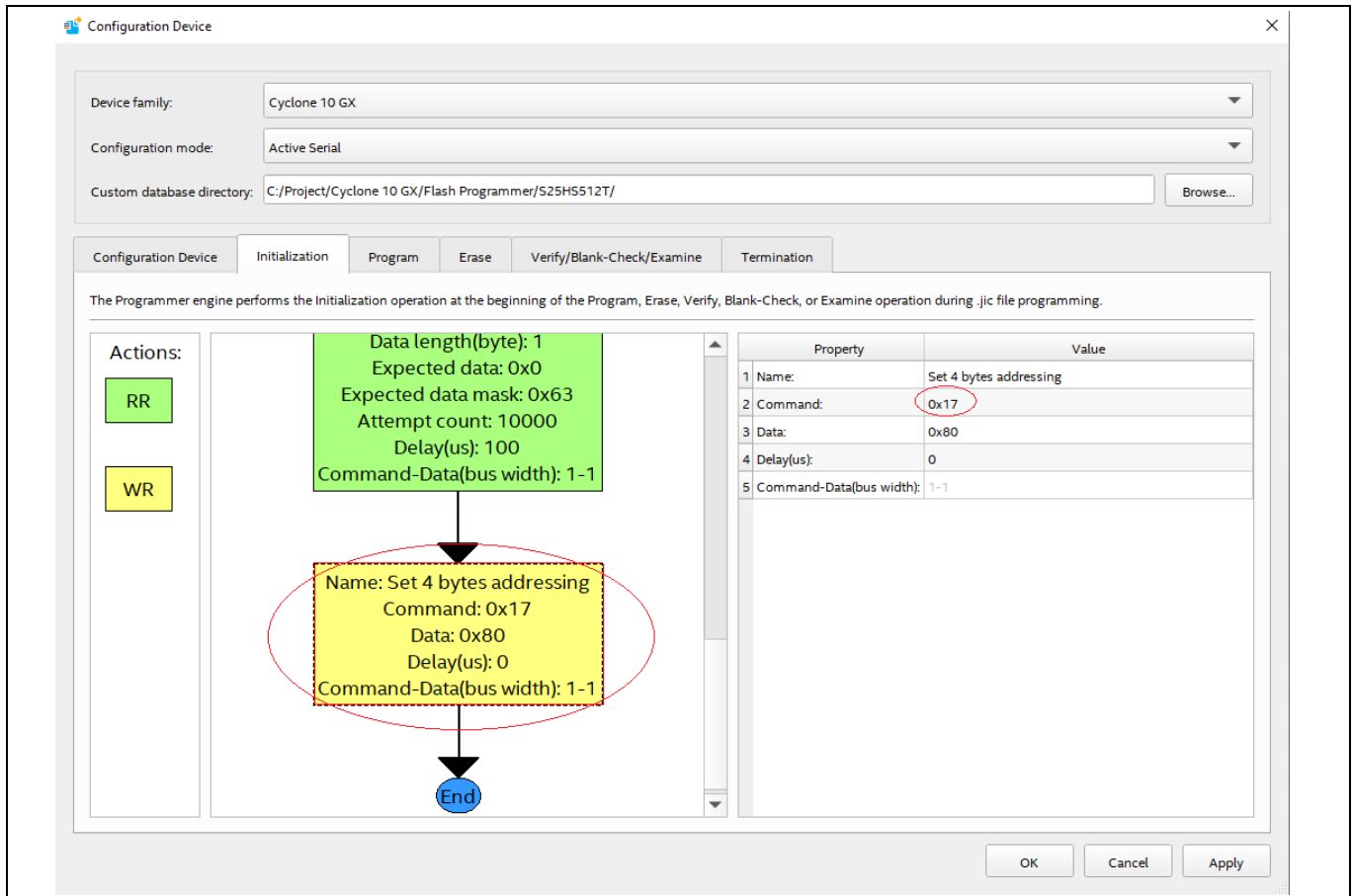


5. Write register 動作の後、書き込み操作のステータスを確認するために、ステータスレジスタを読み出してください。

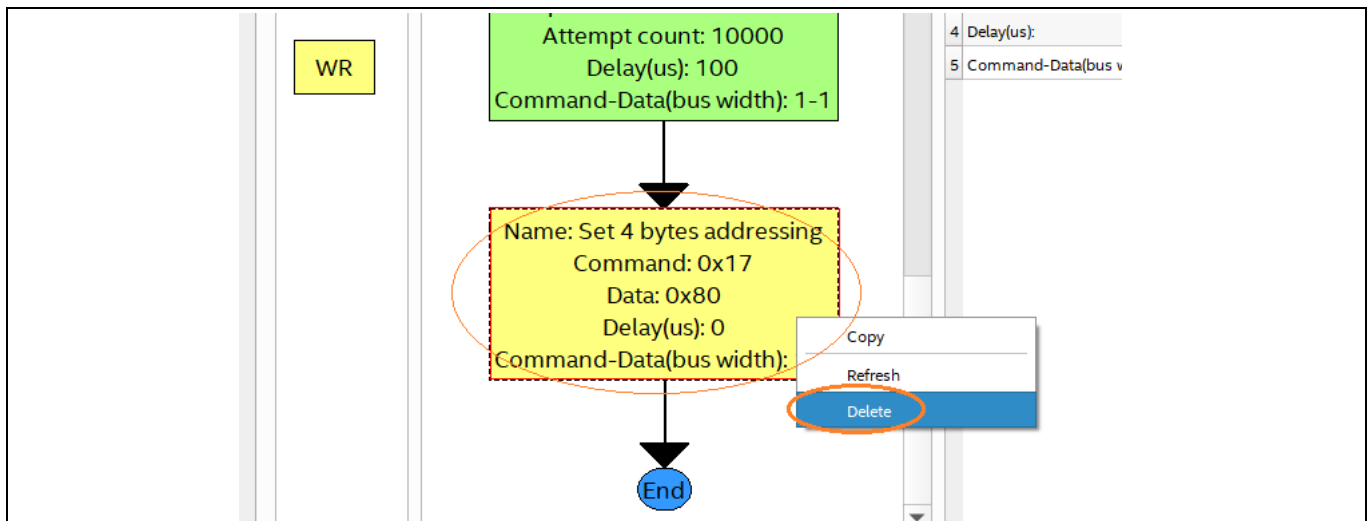
STR1 [6]は P\_ERR、STR1 [5]は E\_ERR、STR1 [1]は WEL、STR1 [0]は RDYBSY です。プログラム操作が正常に完了すると、これらの 4 ビットはすべて 0 になります。他のビットはドントケアです。したがって、期待されるデータマスクは 0110 0011b (0x63) です。マスクされた期待データは 0x0 です。ステータスレジスタビットの定義については、S25HS512T データシートを参照してください。

この動作 (Read Status Register) は、フラッシュアレイプログラム/消去操作のステータスを確認する場合と同じです。プログラムおよび消去フローテンプレートでも使用できます。

## 手順



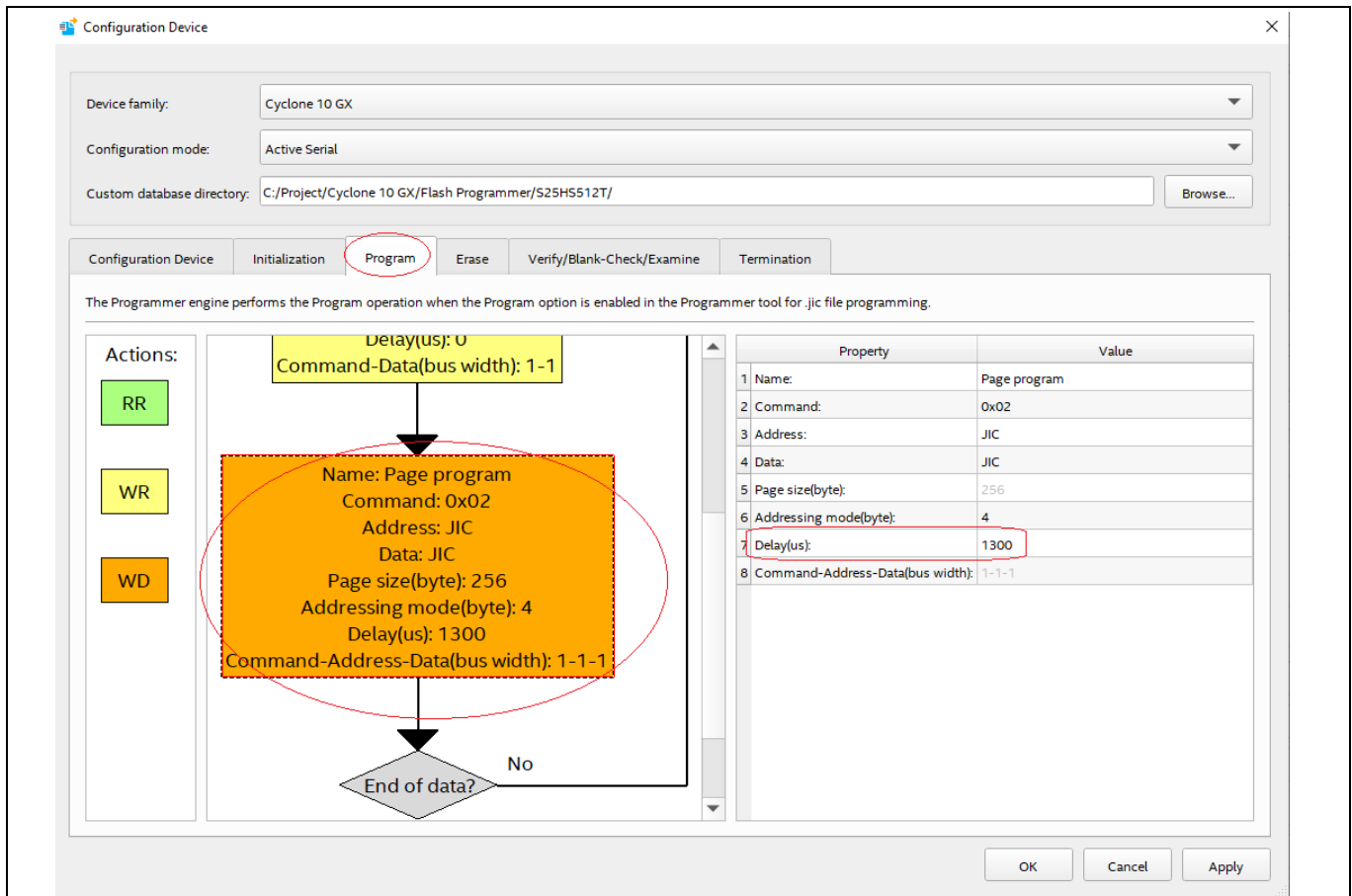
6. **Set 4 byte addressing** 動作を右クリックし、**Delete** をクリックしてから、**Apply** をクリックして変更を保存してください。これは、このコマンドがフラッシュデバイスに適用できないためです。フラッシュ (S25HS512T) の場合、前の書き込みレジスタ動作ですで行われた 4 バイトアドレス指定を設定します。



## 手順

## 3.2.3 プログラムフローテンプレートの変更

1. **Program** タブをクリックし、**Page program** 動作をクリックしてください。

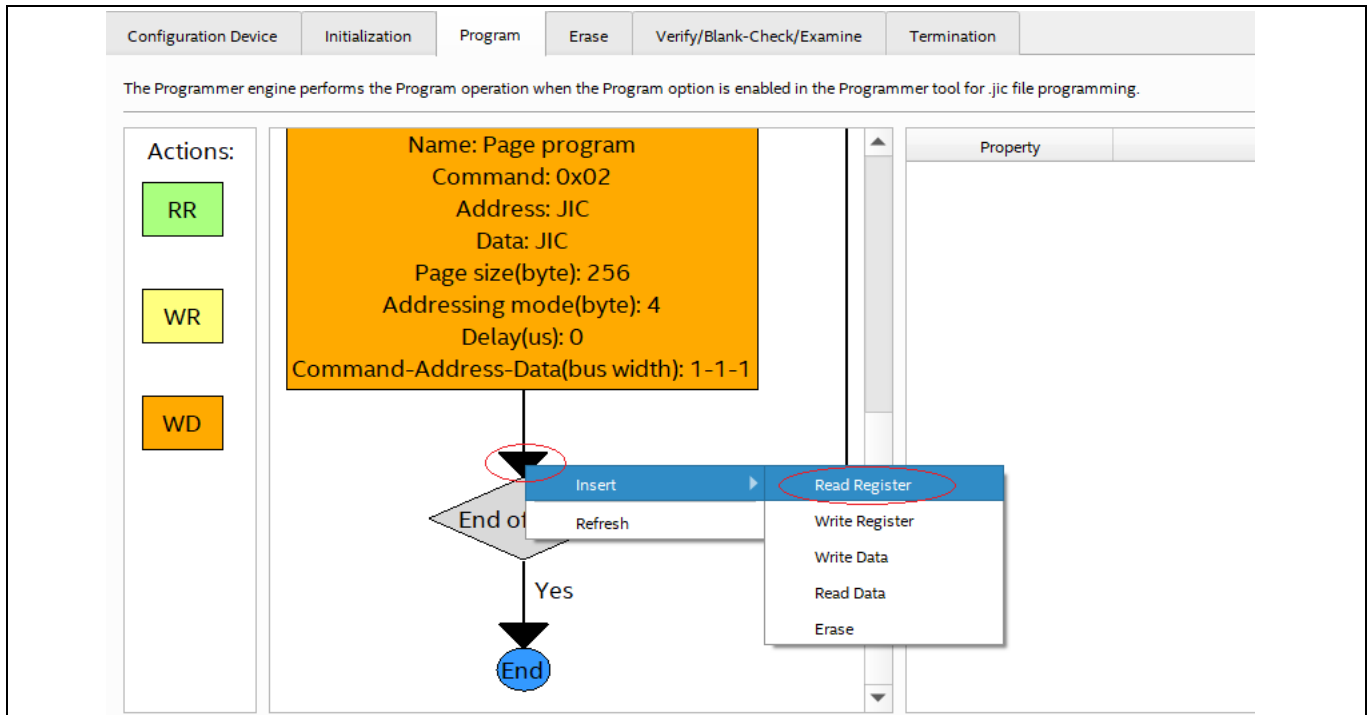


2. フラッシュデータシートで指定されている最大ページプログラム時間に従って、**Delay** の値を変更してください。例えば、S25HS512T の場合、最大ページプログラム時間は 2175  $\mu$ s です。

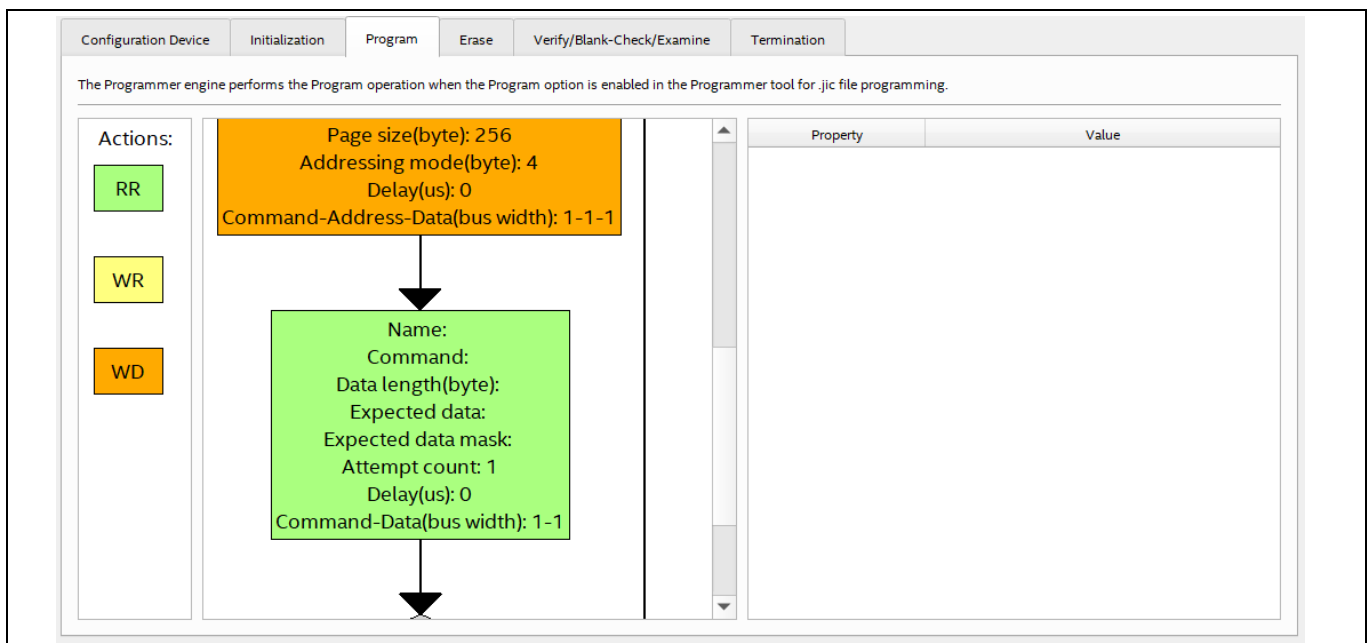
もう 1 つのオプションは、遅延値を '0' に設定し、ステータスレジスタの読み出しアクションを追加して、次のようにページプログラム操作のステータスを確認することです。

ページプログラムのサイズは、デフォルトのプログラムフローテンプレートと S25HS512T : 256 バイトで同じです。S25HS512T は、512 バイトのより大きなページプログラムサイズもサポートします。CFR3x[4]=1 を設定することにより、この値を 512 バイトに構成できます。S25HS512T に 512 バイトのページプログラムサイズを使用するためには、初期化フローテンプレートの Write register 動作で値を設定してください。

## 手順

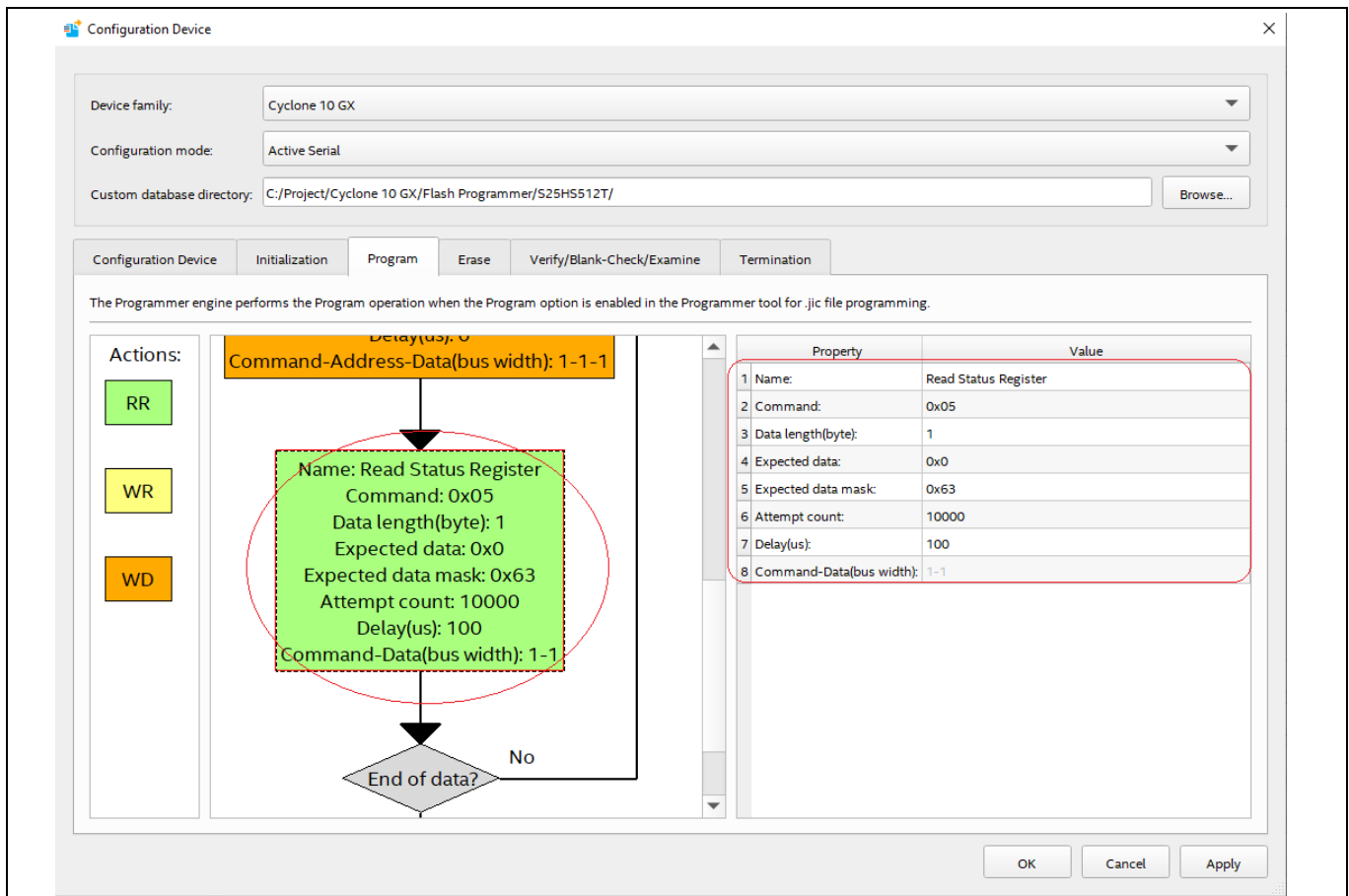


a) 矢印を右クリックし、**Insert > Read Register** を選択して、Read Register 動作をフローに追加してください。



b) 追加した動作をクリックして、そのプロパティを構成してください。

## 手順



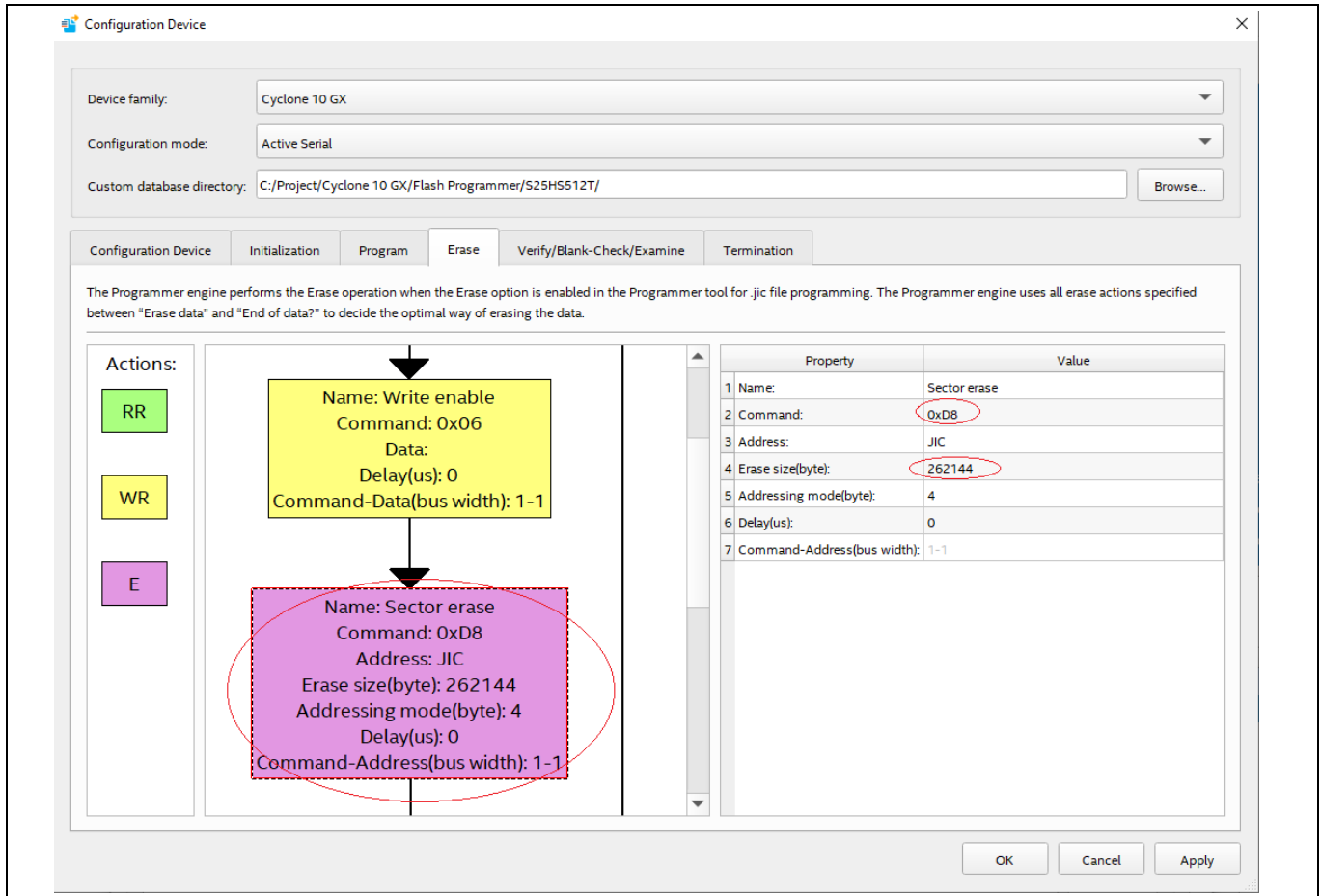
c) フラッシュデータシートに従ってプロパティを構成してください。これは、初期化テンプレートに示される構成と同じです。

時間 (Delay x Attempt count) は、フラッシュデータシートで指定されている最大ページプログラム時間よりも大きくする必要があります。例えば、S25HS512T の場合、最大ページプログラム時間は 2175 us です。したがって、Delay x Attempt count 値は 2175 us より大きくなければなりません。Read Register アクションが期待されるマスクされたデータを返した後、フローは最大試行回数に達することなく次のステップに進むため、より大きな値を入力できます。

d) プロパティを構成したら、**Apply** をクリックしてください。

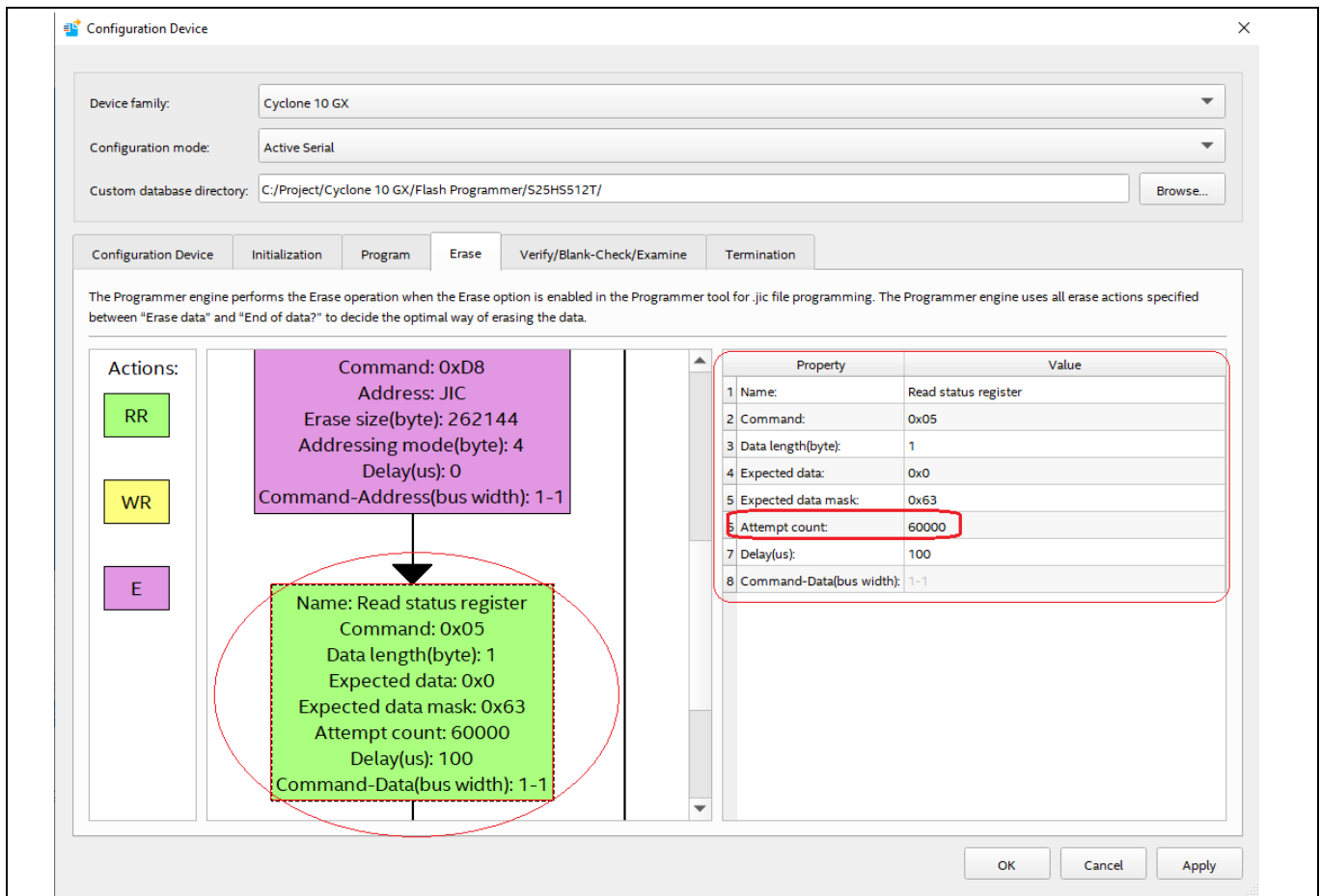
## 手順

### 3.2.4 消去フローテンプレートの変更



デフォルトの“Write enable”、および“Sector erase”動作は、S25HS512T の均一な 256KB セクター消去操作と互換性があります。S25HS512T は、初期化フローテンプレートのレジスタ Write register 動作で 256KB セクターアーキテクチャを統一するようにすでに構成されているため、これらの値は変更しないでください。

## 手順



1. **Read status register** 動作をクリックし、Command、Expected data、data mask などの S25HS512T のプロパティを更新してください。この動作は、初期化フローおよびプログラムフローの **Read status register** 動作に似ています。

Note:  $(Delay \times Attempt\ count)$  は、フラッシュデータシートで指定されている最大 256KB セクター消去時間よりも大きくする必要があります。S25HS512T の場合、256KB EnduraFlex™ アーキテクチャが有効になっていると 5869 ミリ秒になります。詳細については、S25HS512T のデータシートを参照してください。

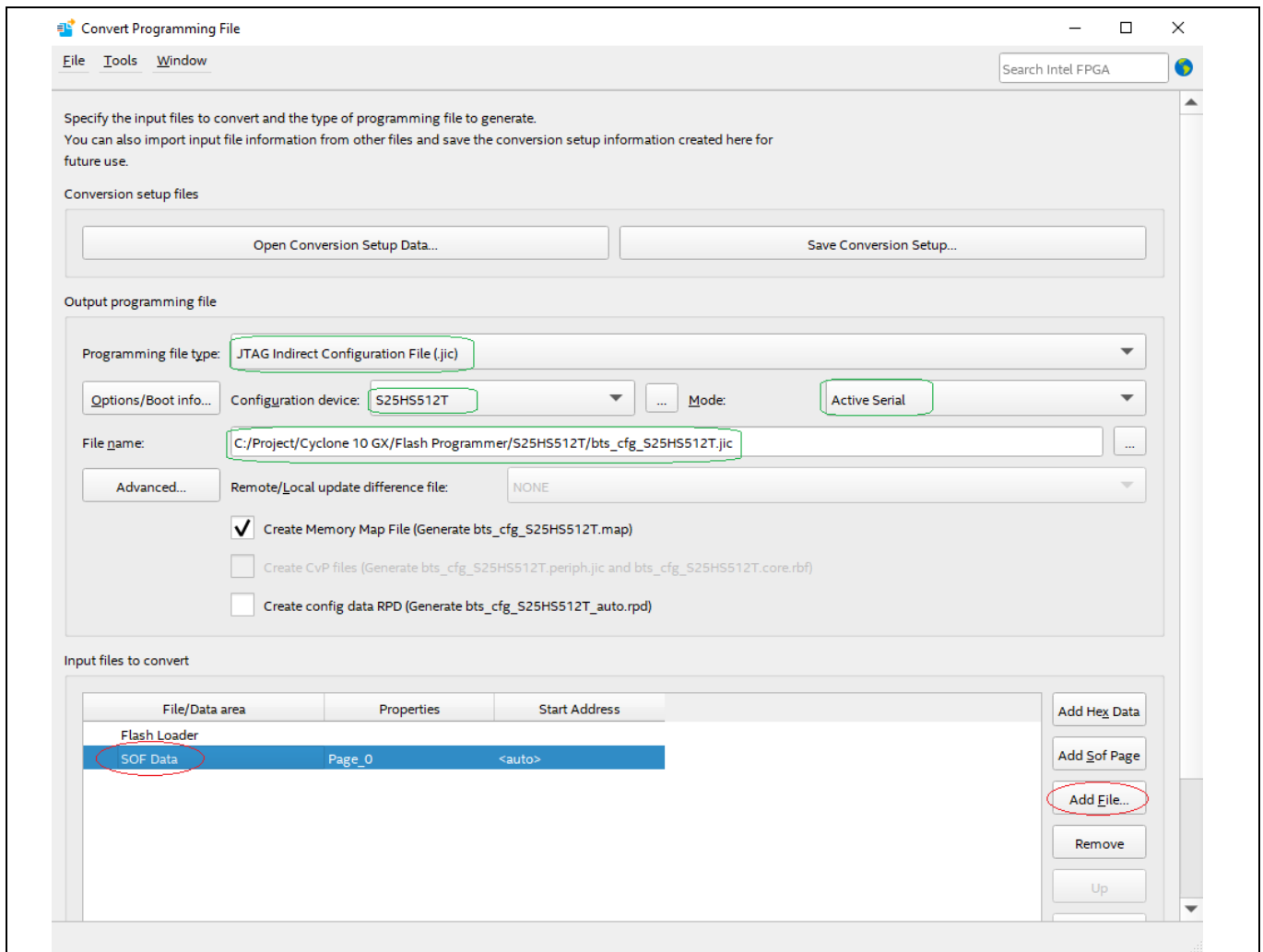
2. **OK** をクリックして変更を保存してください。

これらの動作を構成した後、.sof ファイルをフラッシュデバイスにロードできる .jic ファイルに変換するように構成デバイスを設定してください。



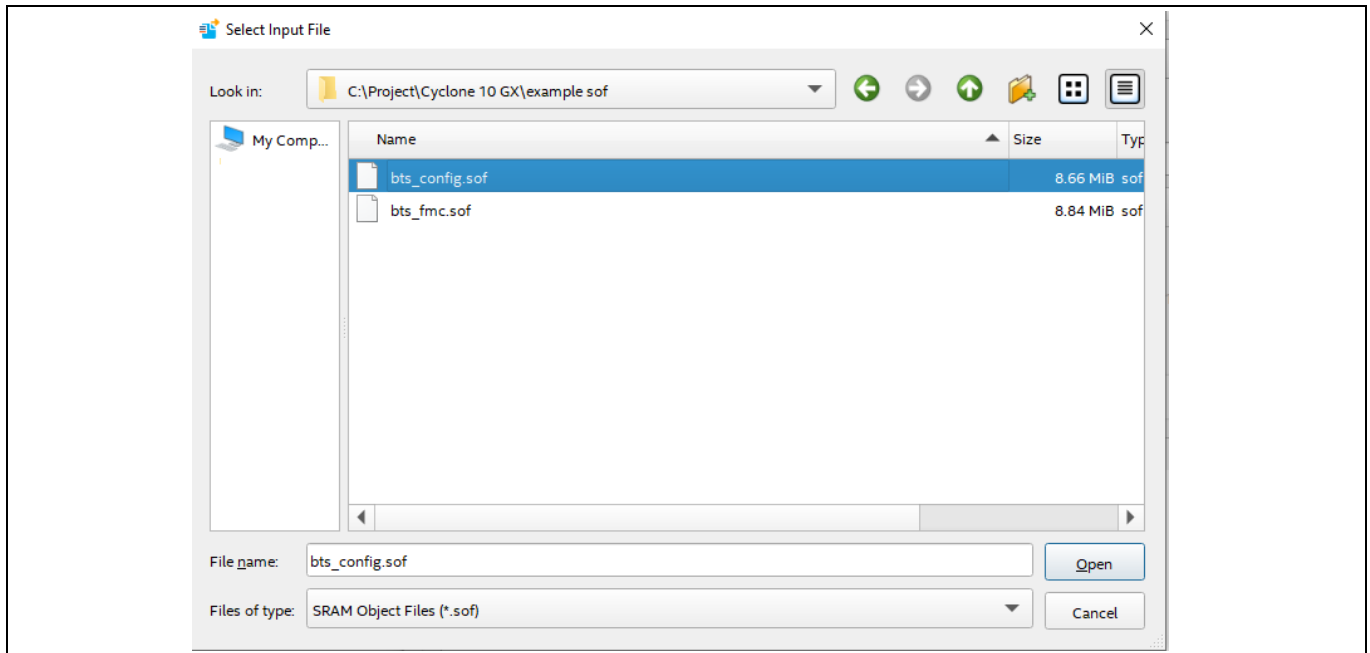
## 手順

## 3.3 .sof ファイルを変換して.jic ファイルを生成

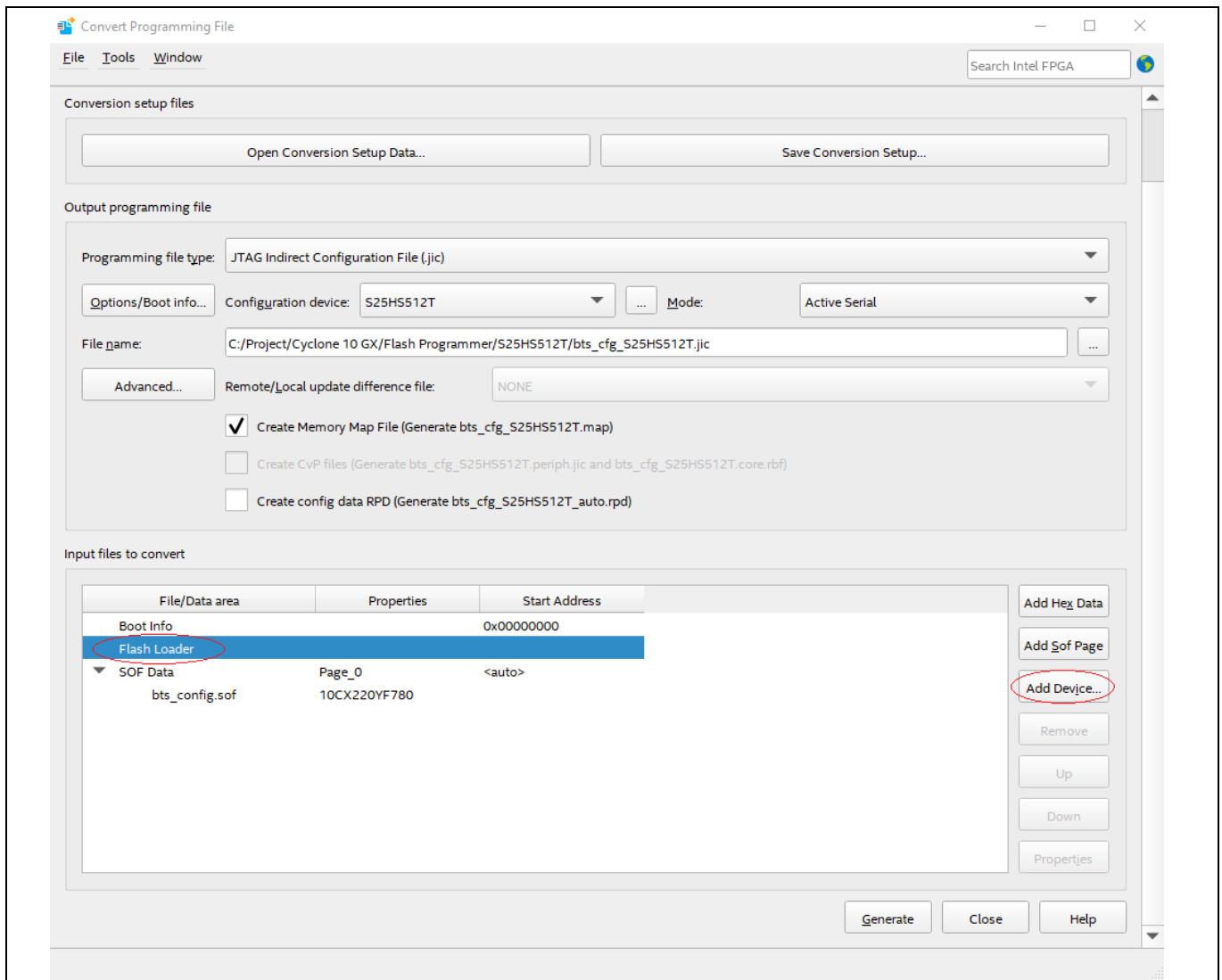


1. **Programming file type** を JTAG Indirect Configuration File (.jic) に設定してください。
2. Configuration device として S25HS512T (前の手順で作成) を選択してください。
3. Mode を **Active Serial** として選択してください。
4. 生成する.jic ファイル名と生成した.jic ファイルを保存する場所を指定してください。
5. **SOF Data** を選択し、**Add File...**をクリックしてください。
6. 変換する.sof ファイルを選択してください。

## 手順

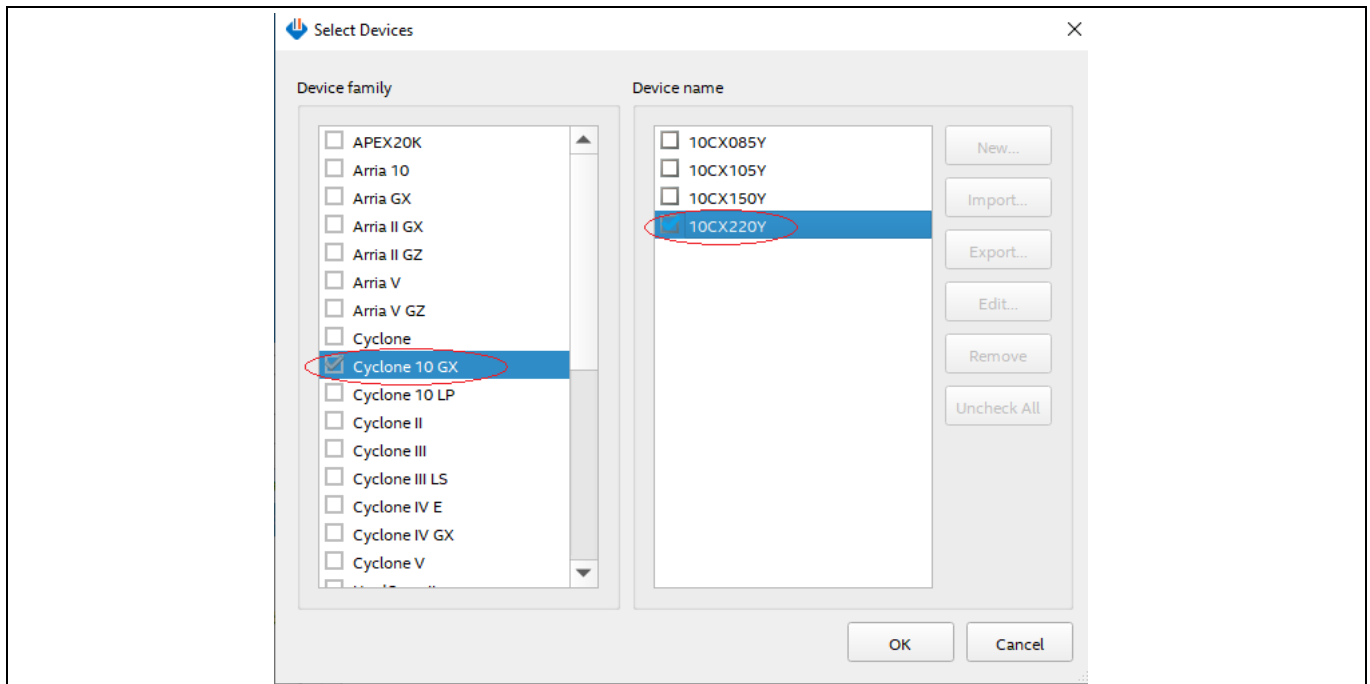


7. .sof ファイルを追加した後、**Flash Loader** を選択し、**Add Device...** をクリックしてください。



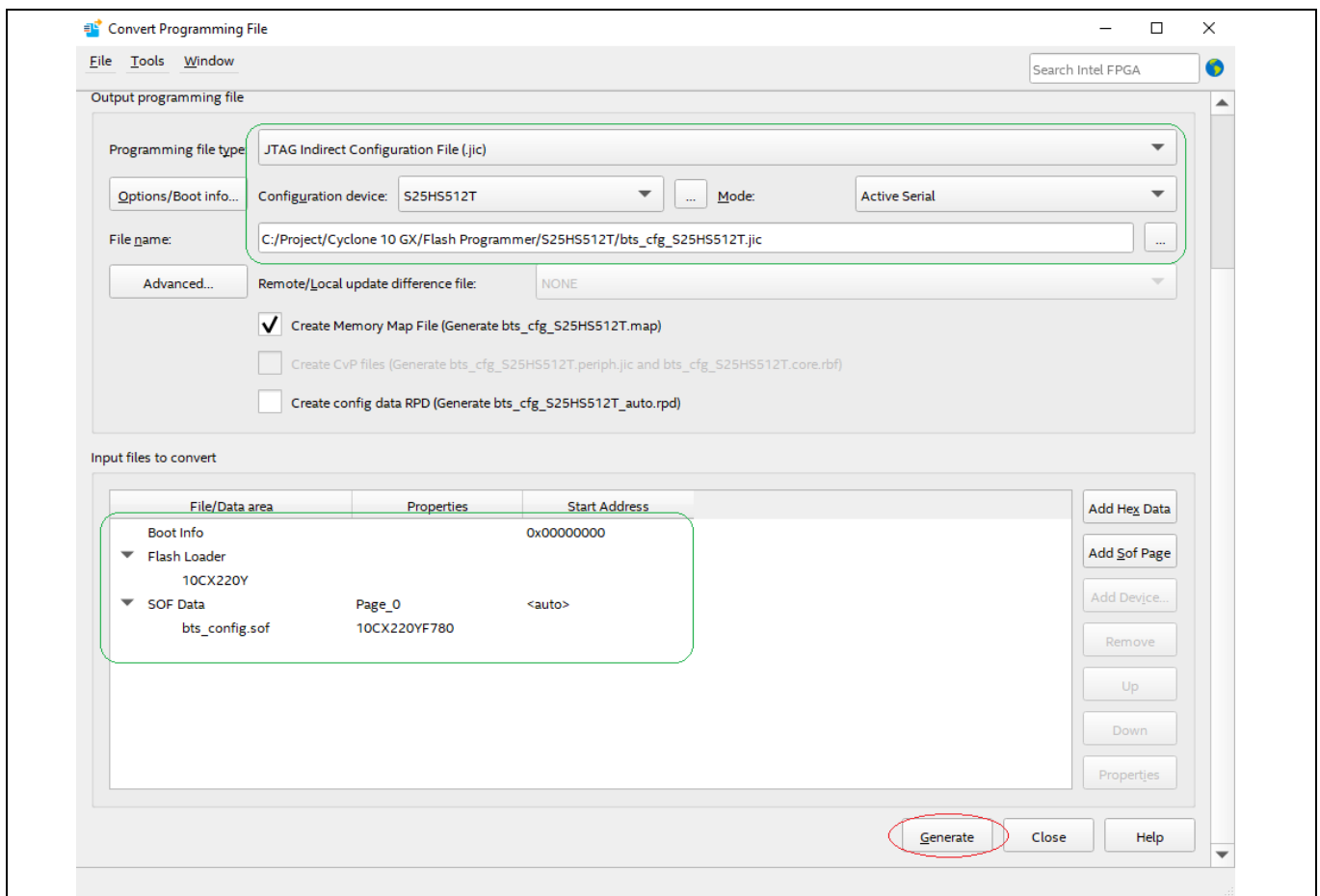
## 手順

8. 使用する FPGA デバイスに応じてデバイスを選択し、**OK** をクリックしてください。



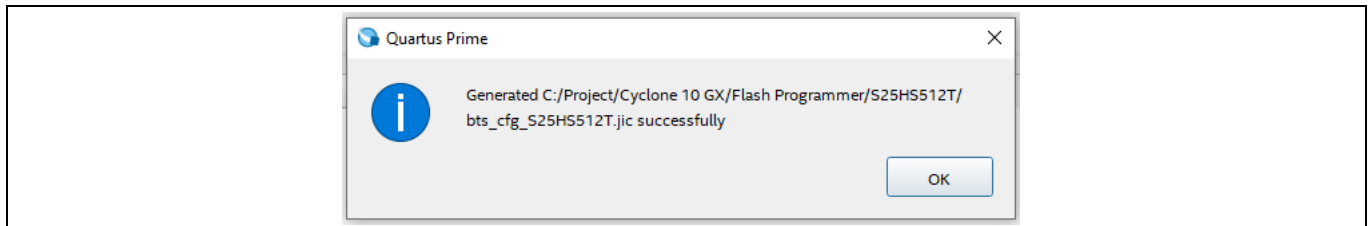
このアプリケーションノートでは、Cyclone10 GX 10CX220Y FPGA デバイスを搭載した Cyclone10 GX Dev キットで手順を確認します。

9. **Generate** をクリックしてください。



## 手順

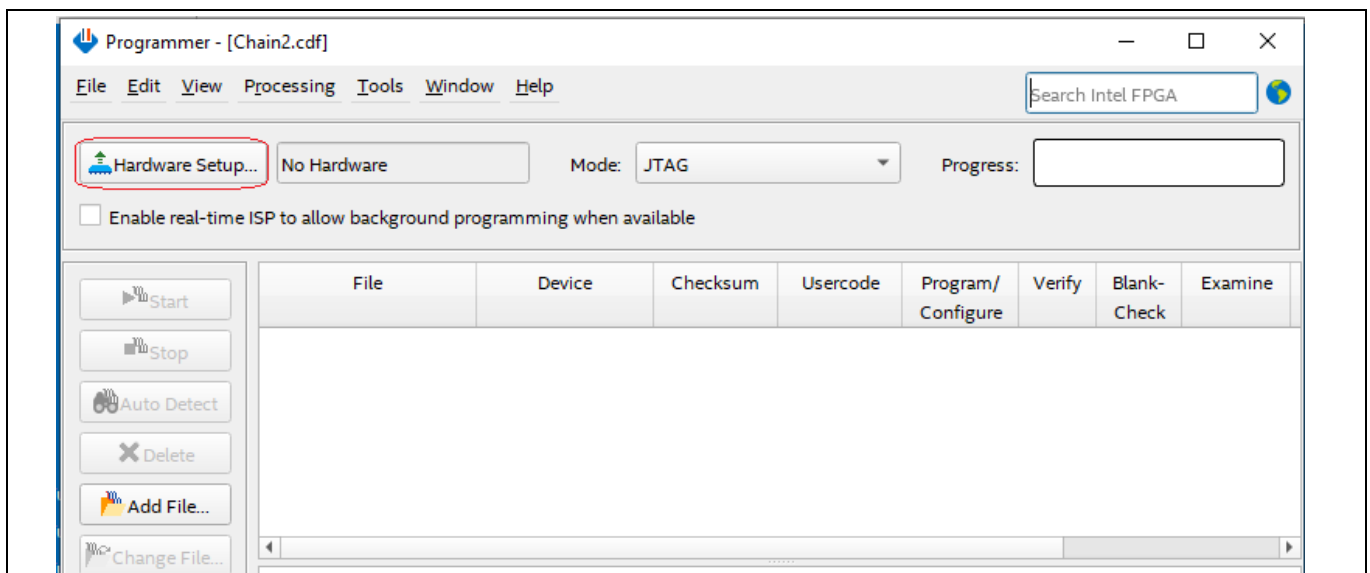
しばらくすると.jic ファイルが正常に生成されたことを示すポップアップウィンドウが表示されます。



これで、生成された.jic ファイルを構成フラッシュ (この例では S25HS512T) にプログラムできます。

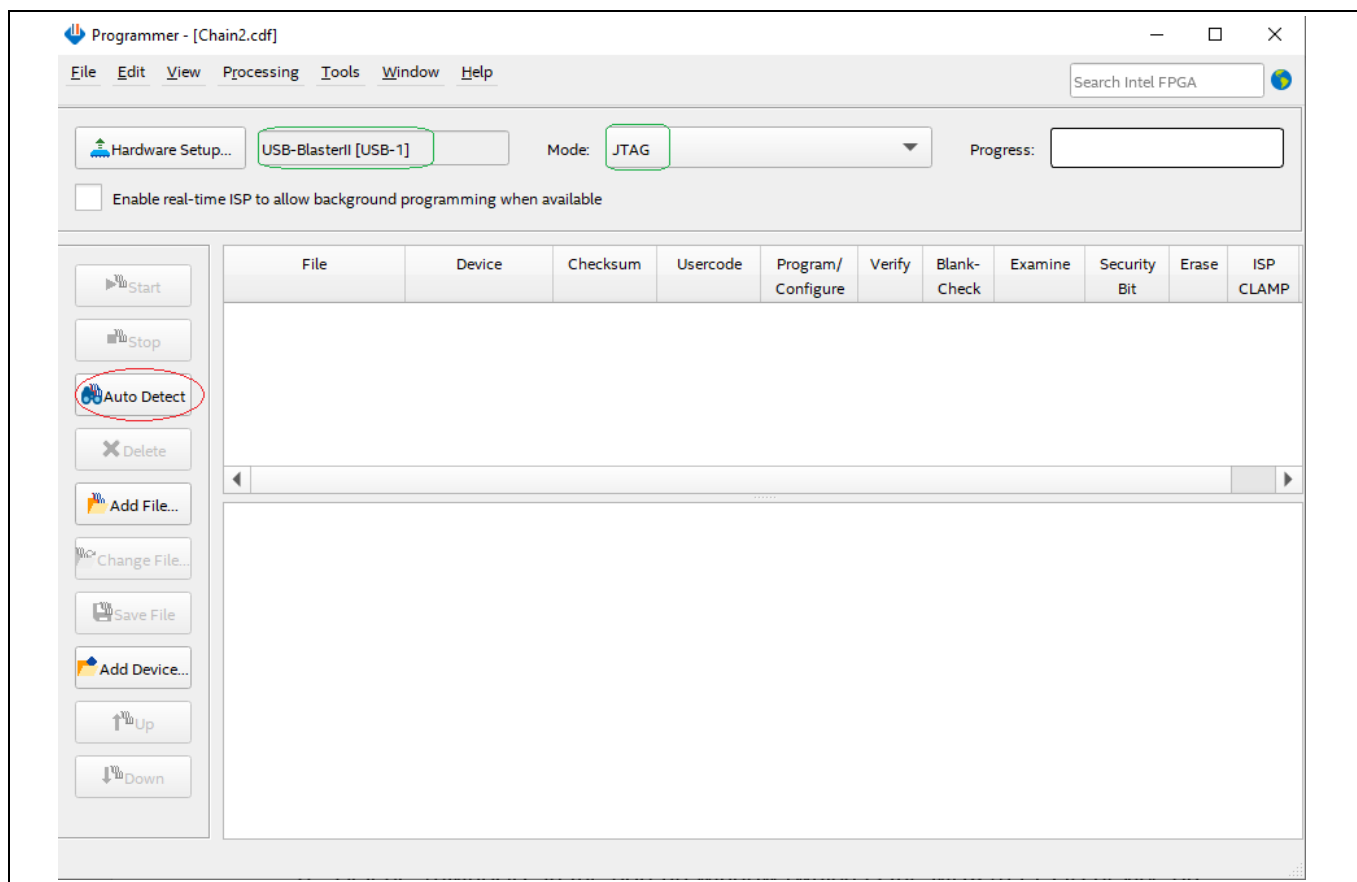
### 3.4 .jic 構成データを QSPI フラッシュデバイスにプログラム

1. Intel FPGA JTAG USB ダウンロードケーブルを介して Intel FPGA ボードを PC に接続し、FPGA ボードの電源を入れてください。
2. Quartus Prime Pro から、**Tools > Programmer** を選択してください。
3. **Hardware Setup...** をクリックしてください。

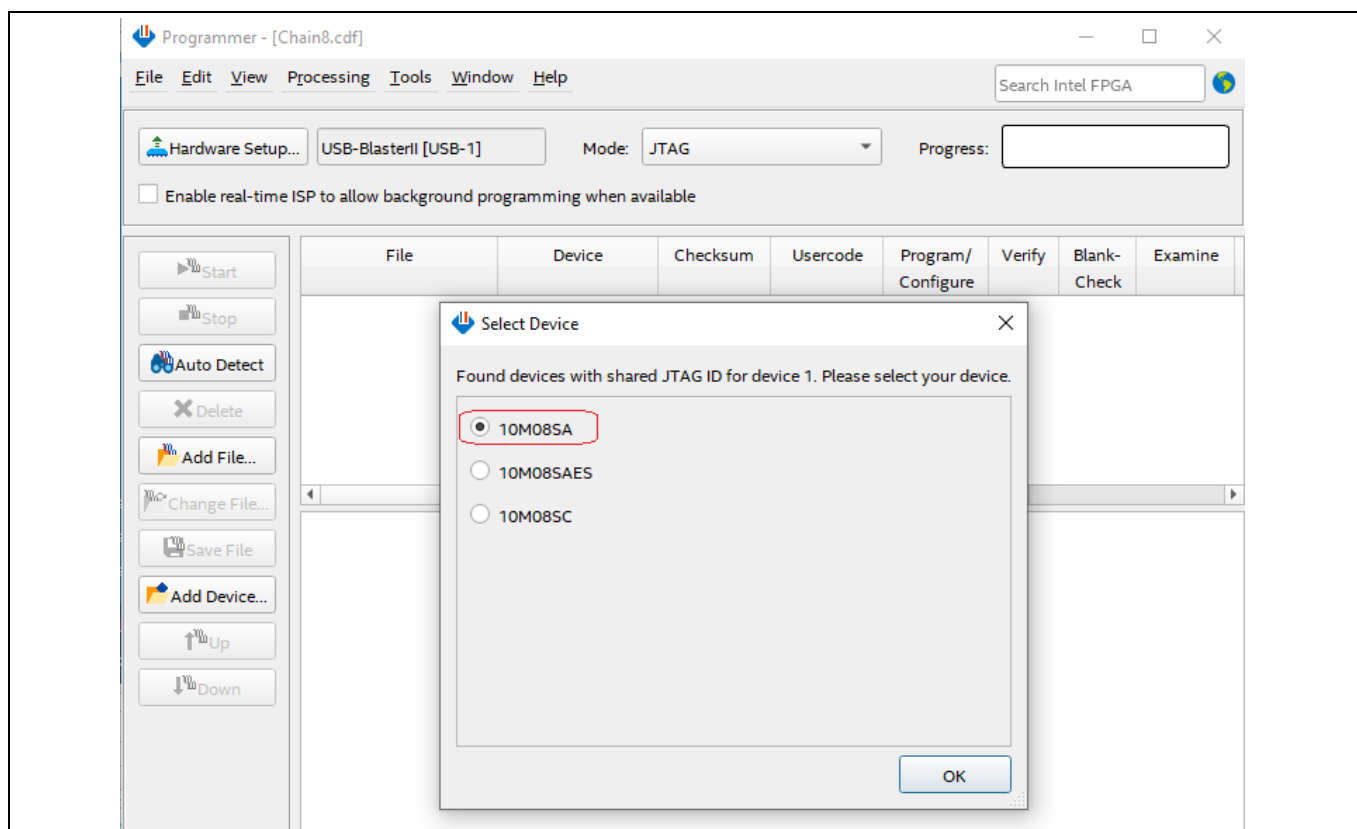


4. **USB-Blaster...** をクリックしてください。

## 手順

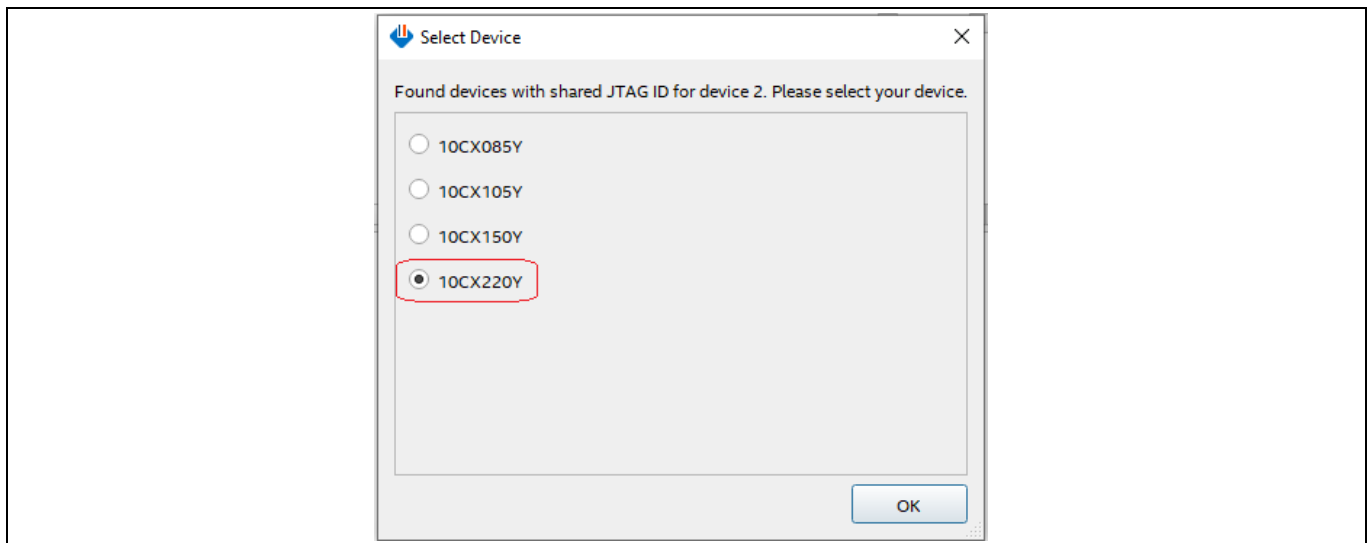


5. **Auto Detect** をクリックしてください。
6. ポップアップウィンドウで **10M08SA** を選択してください。これは、手順が検証された Cyclone10 GX 開発ボード上の MAX10 FPGA デバイスです。

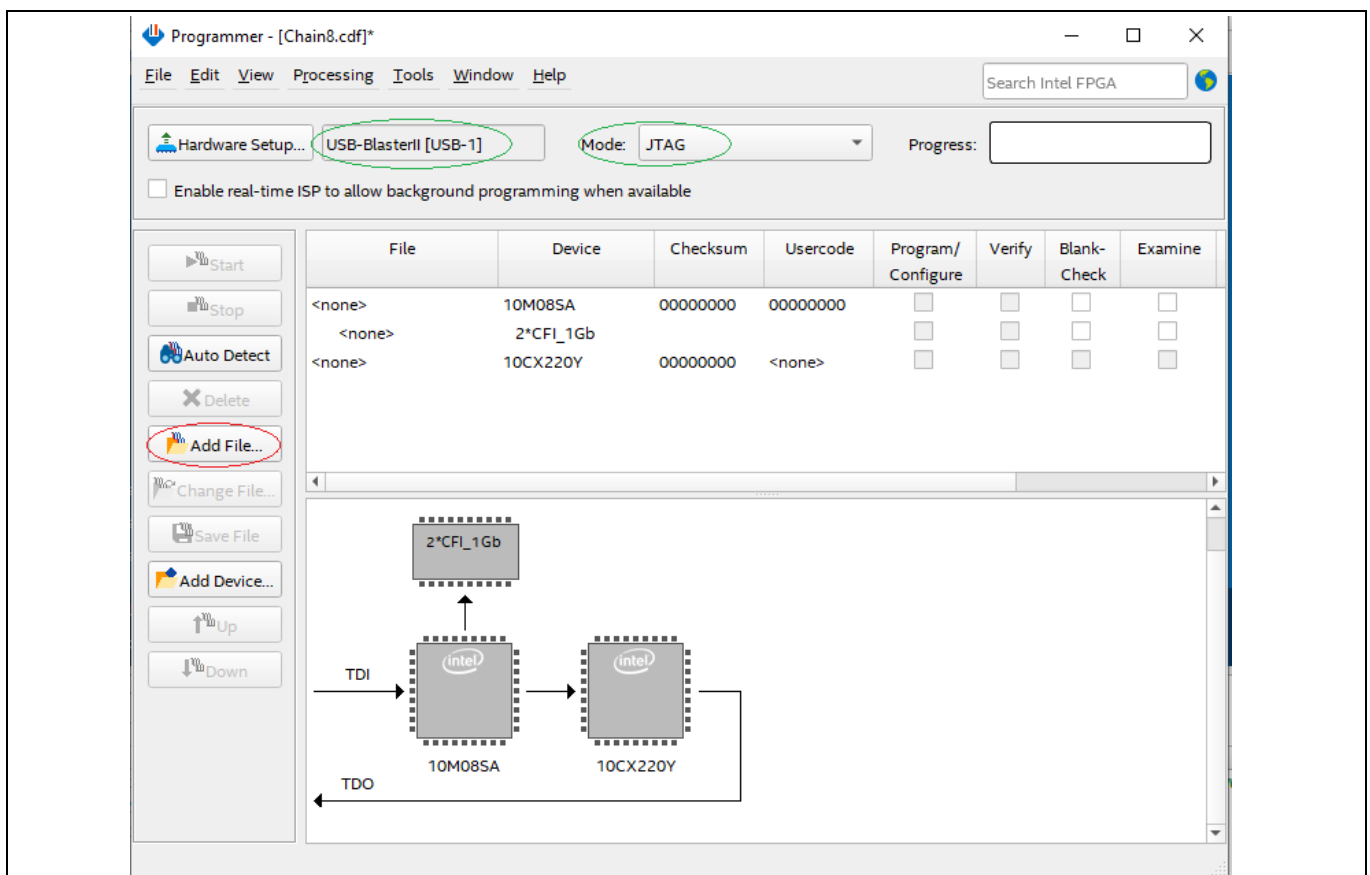


## 手順

7. ポップアップウィンドウで **10CX220Y** を選択してください。これは、開発ボード上の Cyclone 10 GX FPGA デバイスです。

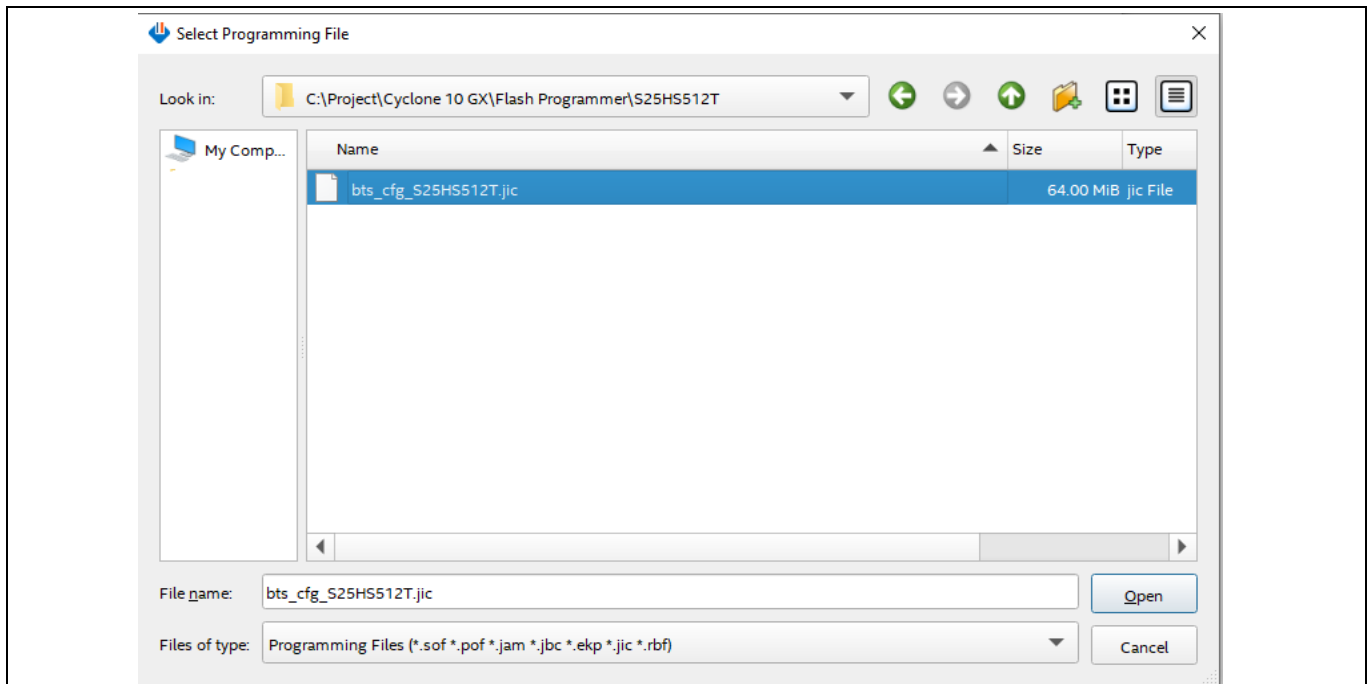


8. **Add File...** をクリックしてください。

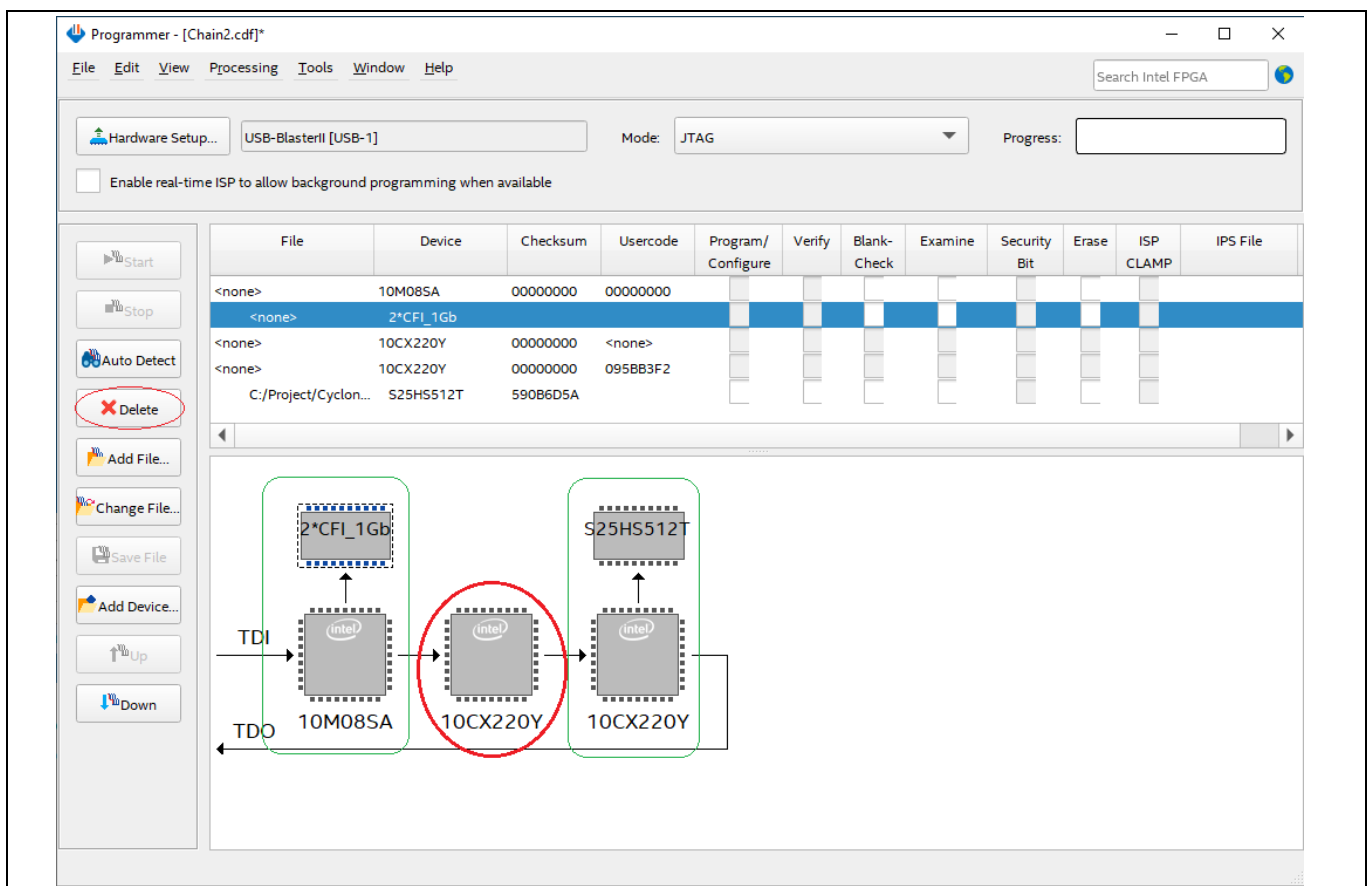


## 手順

9. 前の手順で生成された.jic ファイルを開いてください。

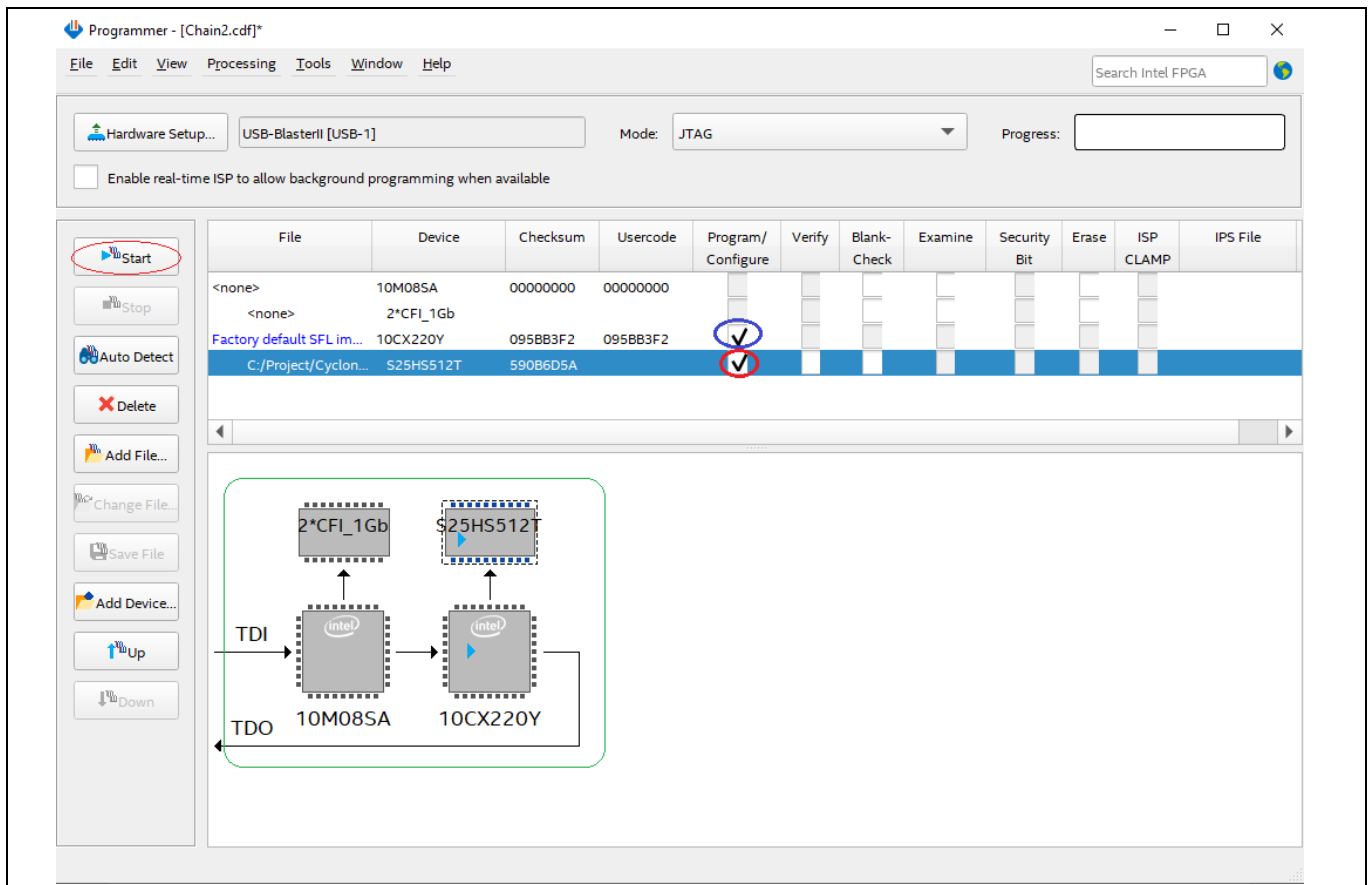


10. .jic ファイルを追加した後に追加のデバイスが表示された場合は、それを選択して **Delete** をクリックしてください。

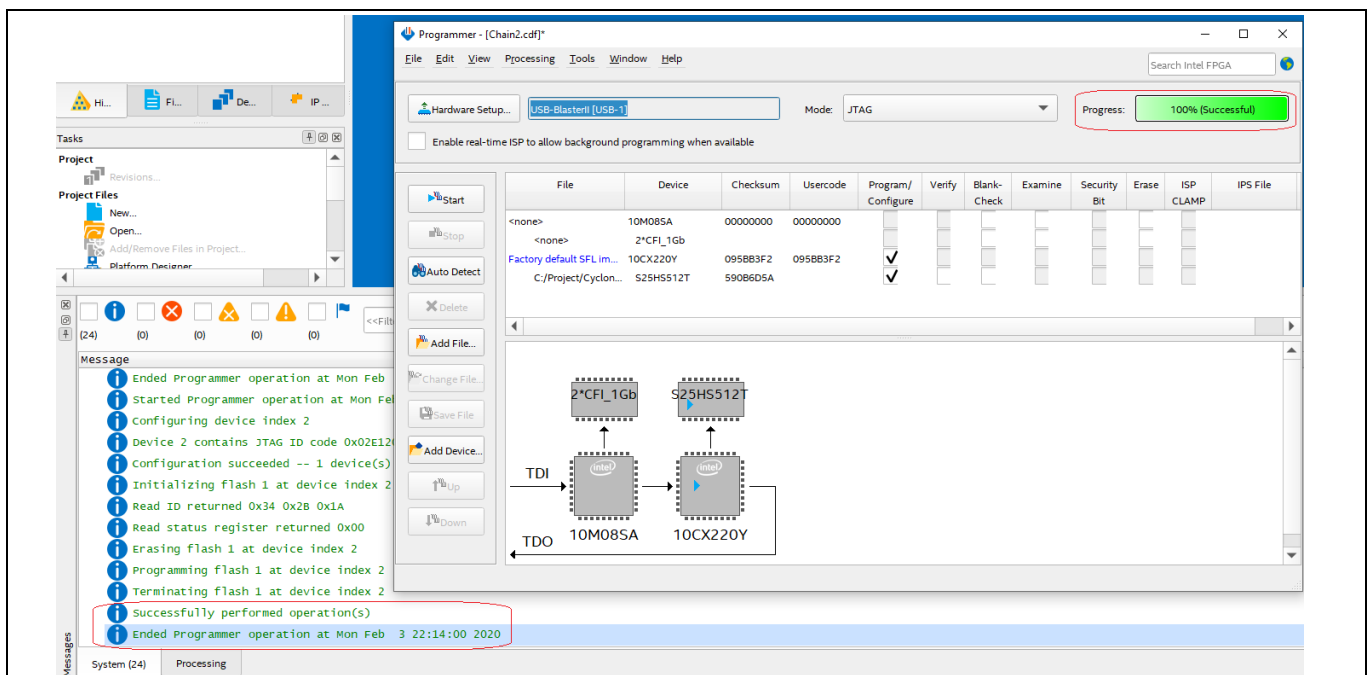


## 手順

11. **Program/Configure** オプションを有効にしてください (赤い円のチェックボックス。青い円のチェックボックスも自動的に有効になります)。



12. **Start** をクリックしてください。



13. プログレスバーに 100% (Successful) と表示され、メッセージに”Successfully performs operations”と表示されたら、ターゲットボードの電源を入れ直してください。



## 結論

### 4 結論

このアプリケーションノートでは、S25HS512T QSPI Semper Flash を有効にして Intel Cyclone 10 GX FPGA デバイスを構成する方法について説明しました。

このアプリケーションノートで紹介されている手順は、それぞれのデータシートに従ってフラッシュと FPGA デバイスを構成することにより、他の QSPI フラッシュファミリと Intel Arria 10FPGA デバイスにも適用されます。

## 参考資料

### 5 参考資料

- Intel Cyclone 10 GX FPGA 開発キットユーザーガイド

<https://www.intel.com/content/www/us/en/programmable/documentation/hvu1509010715799.html?wapkw=Intel%20Cyclone%2010%20GX%20FPGA%20Development%20Kit%20User%20Guide>

- Generic Flash Programmer ユーザーガイド：Intel Quartus Prime Pro Edition

<https://www.intel.com/content/www/us/en/programmable/documentation/pah1554479151020.html?wapkw=Generic%20Flash%20Programmer>

- S25HS-T データシート

## 改訂履歴

## 改訂履歴

Document version	Date of release	Description of changes
**	2021-03-10	本版は英語版 002-29767 Rev. **について、CYPRESS DEVELOPER COMMUNITY の参加者によって日本語に翻訳されたドキュメントです。

## Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2021-03-10

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2021 Infineon Technologies AG.

All Rights Reserved.

Do you have a question about this document?

Go to [www.cypress.com/support](http://www.cypress.com/support)

Document reference

002-32734 Rev. \*\*

## 重要事項

本文書に記載された情報は、いかなる場合も、条件または特性の保証とみなされるものではありません（「品質の保証」）。本文に記載された一切の事例、手引き、もしくは一般的価値、および/または本製品の用途に関する一切の情報に関し、インフィニオンテクノロジーズ（以下、「インフィニオン」）はここに、第三者の知的所有権の不侵害の保証を含むがこれに限らず、あらゆる種類の一切の保証および責任を否定いたします。

さらに、本文書に記載された一切の情報は、お客様の用途におけるお客様の製品およびインフィニオン製品の一切の使用に関し、本文書に記載された義務ならびに一切の関連する法的要件、規範、および基準をお客様が遵守することを条件としています。

本文書に含まれるデータは、技術的訓練を受けた従業員のみを対象としています。本製品の対象用途への適合性、およびこれら用途に関連して本文書に記載された製品情報の完全性についての評価は、お客様の技術部門の責任にて実施してください。

本製品、技術、納品条件、および価格についての詳しい情報は、インフィニオンの最寄りの営業所までお問い合わせください（[www.infineon.com](http://www.infineon.com)）。

## 警告事項

技術的要件に伴い、製品には危険物質が含まれる可能性があります。当該種別の詳細については、インフィニオンの最寄りの営業所までお問い合わせください。

インフィニオンの正式代表者が署名した書面を通じ、インフィニオンによる明示の承認が存在する場合を除き、インフィニオンの製品は、当該製品の障害またはその使用に関する一切の結果が、合理的に人的傷害を招く恐れのある一切の用途に使用することはできないこと予めご了承ください