|  |
| --- |
| AN229767 |
| ＜＜★［1］Enabling Cypress QSPI Flash to Configure Intel Cyclone 10 GX FPGA＞＞サイプレスQSPIフラッシュを有効にしてIntelCyclone 10 GXFPGAを構成する  ＜＜★In-System Programming for Cypress SPI Flash on Altera® FPGA Board＞＞アルテラ®FPGAボード上のサイプレスSPIフラッシュのインシステムプログラミング |
| ＜＜★［1］AN229767 describes how to enable Cypress S25HS512T QSPI Flash for configuring Intel Cyclone 10 GX FPGA under Active Serial (AS) mode by using the Intel Quartus Prime Pro Generic Flash Programmer tool. The same method also applies to enable other Cypress QSPI Flash families by configuring the flash according to the respective flash datasheet.＞＞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 GXFPGAを構成するためにCypressS25HS512TQSPIフラッシュを有効にする方法を説明しています。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、他のサイプレス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。 |

目次

[1 はじめに 1](#_Toc33614454)

[2 ＜＜★Background Information＞＞背景情報 1](#_Toc33614455)

[3 ＜＜★Procedures＞＞手順 2](#_Toc33614456)

[3.1 ＜＜★Generate \*.sof FPGA configuration file＞＞\* .sofFPGAコンフィギュレーションファイルを生成します 2](#_Toc33614457)

[3.2 ＜＜★Define the new Configuration Device＞＞新しい構成デバイスを定義する 2](#_Toc33614458)

[3.2.1 ＜＜★Add the new Configuration Device＞＞新しい構成デバイスを追加します 2](#_Toc33614459)

[3.2.2 ＜＜★Modify Initialization Flow Template＞＞初期化フローテンプレートの変更 5](#_Toc33614460)

[3.2.3 ＜＜★Modify Program Flow Template＞＞プログラムフローテンプレートの変更 10](#_Toc33614461)

[3.2.4 ＜＜★Modify Erase Flow Template＞＞消去フローテンプレートの変更 13](#_Toc33614462)

[3.3 ＜＜★Convert the .sof File to Generate the .jic File＞＞.sofファイルを変換して.jicファイルを生成する 15](#_Toc33614463)

[3.4 ＜＜★Program the .jic Configuration Data into the Cypress QSPI Flash Device＞＞.jic構成データをサイプレスQSPIフラッシュデバイスにプログラムします 20](#_Toc33614464)

[4 結論 25](#_Toc33614465)

[5 参考資料 25](#_Toc33614466)

[世界中にわたるセールスおよびデザイン サポート 27](#_Toc33614468)

# はじめに

＜＜★The document is written assuming the reader is familiar with Intel FPGA development.＞＞このドキュメントは、読者がIntelFPGAの開発に精通していることを前提に書かれています。

＜＜★Intel’s Generic Flash Programmer tool in Quartus Prime 19.1 and later versions provides a way to configure third-party QSPI flash as a FPGA Configuration Device.＞＞Quartus Prime19.1以降のバージョンのIntelのGenericFlash 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。

＜＜★This application note describes how to use the Generic Flash Programmer tool to load an FPGA configuration bitstream file into Cypress S25HS512T QSPI flash.＞＞このアプリケーションノートでは、Generic FlashProgrammerツールを使用してFPGAコンフィギュレーションビットストリームファイルをサイプレスS25HS512TQSPIフラッシュにロードする方法について説明します。＜＜★The Cypress QSPI flash subsequently loads the configuration data into the target FPGA via Active Serial (AS) configuration.＞＞その後、サイプレス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。

＜＜★［1］The procedures introduced in this application note were verified on the Intel Cyclone 10 GX FPGA Development Kit with Cypress Semper™ Flash with Quad SPI Interface – specifically the S25HS512T – by using Quartus Prime Pro 19.2. The procedures are expected to apply to all other Quartus Prime version 19.1 and later. The procedures should also apply to other Cypress QSPI Flash families and Intel Arria 10 FPGA devices by configuring the flash and FPGA device according to the respective datasheet.＞＞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インターフェイスを備えたCypressSemper™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QuartusPrime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、他のサイプレスQSPIフラッシュファミリとIntel Arria 10FPGAデバイスにも適用する必要があります。

# ＜＜★Background Information＞＞背景情報

＜＜★［1］The Intel Generic Flash Programmer supports Intel Stratix 10, Intel Arria 10, and Intel Cyclone 10 GX FPGA devices.＞＞Intel Generic Flash Programmerは、Intel Stratix 10、Intel Arria 10、およびIntel Cyclone 10 GXFPGAデバイスをサポートします。＜＜★However, for Intel Stratix 10 devices, the Secure Device Manager (SDM) firmware controls the flash programming flows; you cannot modify these flows. Thus, the procedure introduced in this application note does not apply to Intel Stratix 10 FPGA devices.＞＞ただし、Intel Stratix 10デバイスの場合、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。これらのフローを変更することはできません。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 10FPGAデバイスには適用されません。

＜＜★The following diagram shows the Generic Flash Programmer Configuration data flows in red.＞＞次の図は、汎用フラッシュプログラマ構成データフローを赤で示しています。



# ＜＜★Procedures＞＞手順

＜＜★You can access settings and controls for the Generic Flash Programmer from the Quartus Prime **Programmer**, **Convert** **Programming File** dialog box.＞＞Quartus Prime Programmerの[プログラミングファイルの変換]ダイアログボックスから、汎用フラッシュプログラマーの設定とコントロールにアクセスできます。

＜＜★Generic Flash Programming with the Convert Programming File dialog box includes the following high-level steps:＞＞[プログラミングファイルの変換]ダイアログボックスを使用した汎用フラッシュプログラミングには、次の高レベルの手順が含まれています。

* + - 1. ＜＜★Generate \*.sof FPGA configuration file.＞＞\* .sofFPGAコンフィギュレーションファイルを生成します。
      2. ＜＜★Define a new Configuration Device＞＞新しい構成デバイスを定義する
      3. ＜＜★Convert \*.sof file to \*.jic file which is used to program Cypress QSPI flash to store the configuration data.＞＞\* .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サイプレスQSPIフラッシュをプログラムするために使用されます。
      4. ＜＜★Use the Intel Quartus Prime Programmer and connect the Intel FPGA download cable to program the \*.jic configuration data into the Cypress QSPI flash device.＞＞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構成データをサイプレスQSPIフラッシュデバイスにプログラムします。

## ＜＜★Generate \*.sof FPGA configuration file＞＞\* .sofFPGAコンフィギュレーションファイルを生成します

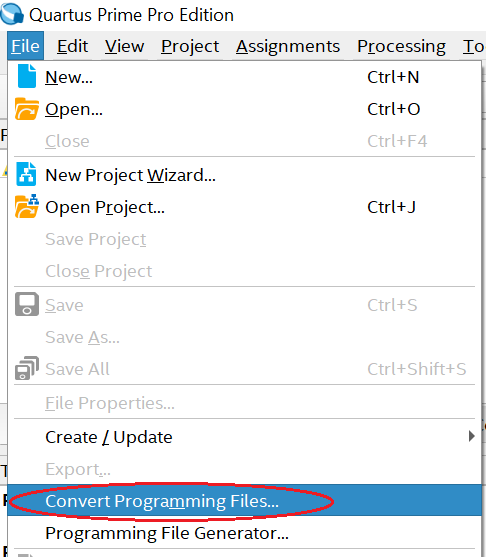
＜＜★This step is NOT covered in this document.＞＞この手順は、このドキュメントではカバーされていません。＜＜★It assumes you have a \*.sof file already generated. See relevant Intel documents for generating \*.sof files.＞＞\* .sofファイルがすでに生成されていることを前提としています。\* .sofファイルの生成については、関連するIntelドキュメントを参照してください。

## ＜＜★Define the new Configuration Device＞＞新しい構成デバイスを定義する

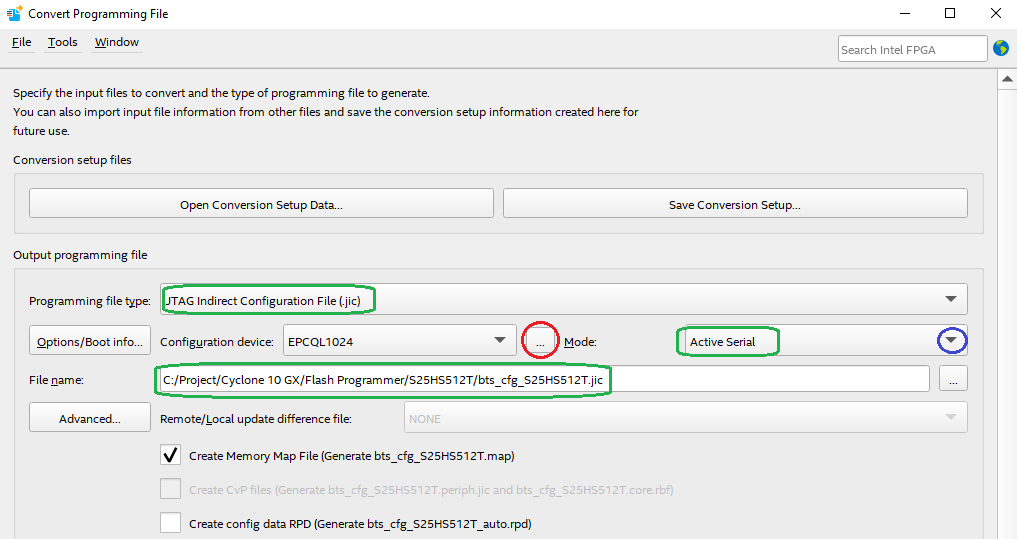
＜＜★While defining the new Configuration Device, the Quartus Prime software stores the collection of settings in an \*.xml file in the **Device database directory** location of your choice. This configuration only needs to be done once; the new Configuration Device template can be reused just as the default Configuration Device templates listed in the Quartus Prime.＞＞新しい構成デバイスを定義する際、Quartus Primeソフトウェアは、設定のコレクションを、選択したデバイスデータベースディレクトリの場所にある\* .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。新しい構成デバイステンプレートは、QuartusPrime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。

### ＜＜★Add the new Configuration Device＞＞新しい構成デバイスを追加します

* 1. ＜＜★Open Quartus Prime Pro software, and select **File** > **Convert Programming Files…**.＞＞Quartus PrimeProソフトウェアと**File** > **Convert Programming Files…**.を開きます



1. ＜＜★Do the following in the **Convert Programming File** screen:＞＞[プログラミングファイルの変換]画面で次の手順を実行します。

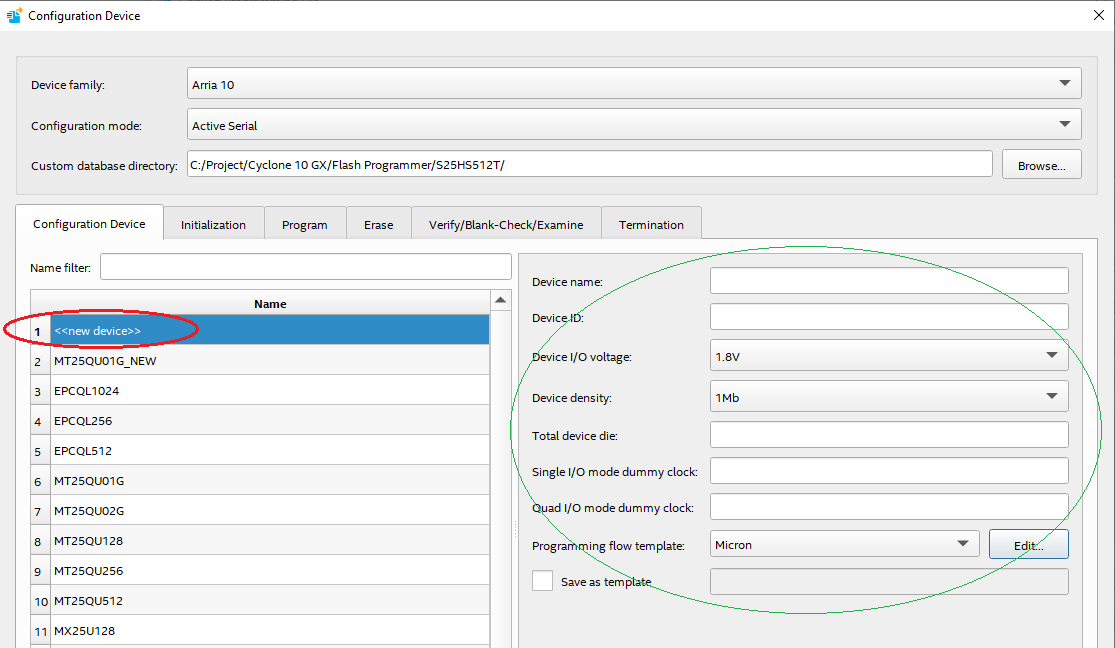


* + 1. ＜＜★Select the \*.jic file type in Programming file type.＞＞プログラミングファイルタイプで\* .jicファイルタイプを選択します。

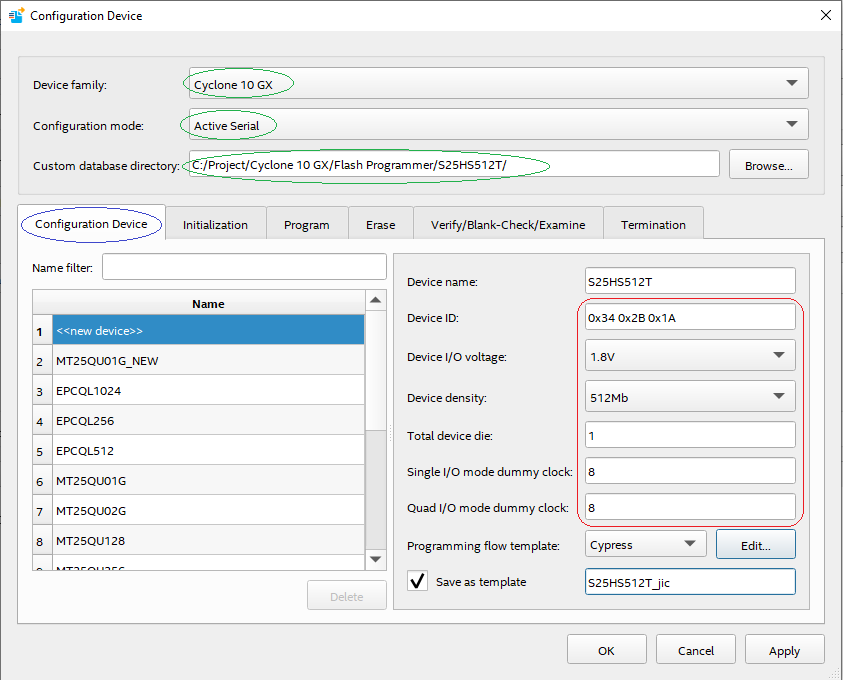
**＜＜★Note**: You can select the Mode option only if \*.jic is selected.＞＞注：[モード]オプションは、\*。jicが選択されている場合にのみ選択できます。

* + 1. ＜＜★In the File name field, specify the location to store the converted \*.jic file.＞＞[ファイル名]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ます。
    2. ＜＜★Click the pull-down menu in the blue circle, and select **Active Serial**.＞＞青い円のプルダウンメニューをクリックし、[アクティブシリアル]を選択します。
    3. ＜＜★Click in the red circle.＞＞赤い丸をクリックします。

1. ＜＜★Click **<<new device>>**.＞＞<<新しいデバイス>>をクリックします。＜＜★Configurable fields in the green circle appear.＞＞緑の円で構成可能なフィールドが表示されます。



1. ＜＜★Make the following changes:＞＞次の変更を行います。



* ＜＜★Configure the FPGA **Device family** and **Configuration mode** according to the FPGA device.＞＞FPGAデバイスに応じてFPGAデバイスファミリとコンフィギュレーションモードを設定します。
* ＜＜★Specify the **Custom database directory** where the \*.xml file will be stored.＞＞\* .xmlファイルが保存されるカスタムデータベースディレクトリを指定します。
* ＜＜★Configure the flash device parameters according to the QSPI Flash device datasheet:＞＞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ます。
  + デバイスID
  + ＜＜★Device I/O voltage＞＞デバイスのI / O電圧
  + ＜＜★Device density＞＞デバイス密度
  + ＜＜★Total device die＞＞総デバイスダイ
  + ＜＜★Single I/O mode dummy clock＞＞シングルI / Oモードダミークロック
  + ＜＜★Quad I/O mode dummy clock＞＞クワッドI / Oモードダミークロック

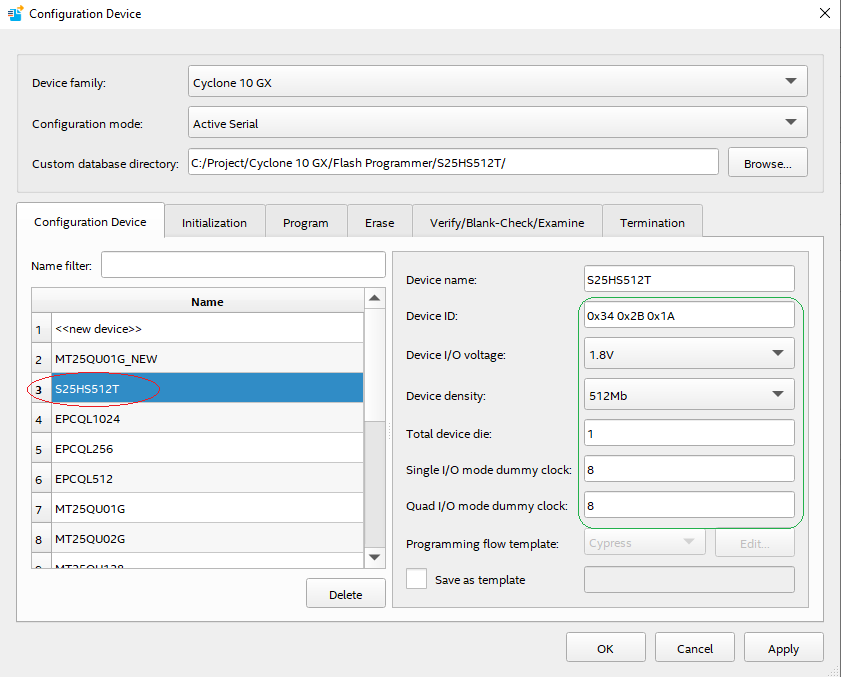
＜＜★For example, for the S25HS512T device, enter these parameters from the S25HS512T datasheet.＞＞たとえば、S25HS512Tデバイスの場合、S25HS512Tデータシートからこれらのパラメータを入力します。

1. ＜＜★Click **Apply**.＞＞[適用]をクリックします。＜＜★The new device “S25HS512T” is automatically added in the list on the left.＞＞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す。

**＜＜★Notes**:＞＞ノート：

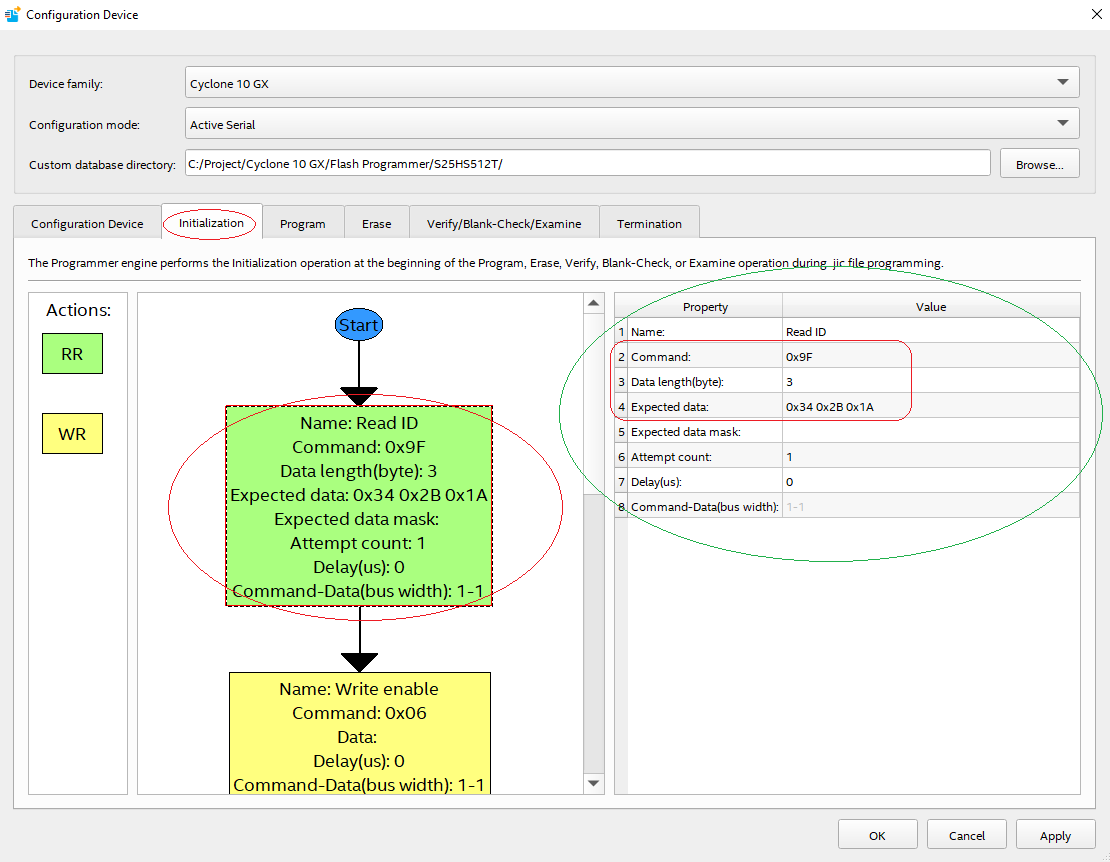
* ＜＜★In the flash parameter configuration, **Device name** and template name (“Save as template”) cannot be the same.＞＞フラッシュパラメータの設定では、デバイス名とテンプレート名（「テンプレートとして保存」）を同じにすることはできません。＜＜★Otherwise, an error message “Template name xxxxxx conflicts with the device name xxxxxx” appears when you click the **Apply** button.＞＞それ以外の場合は、[適用]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。
* ＜＜★［16］The **Device ID** is a 3-byte hex data in the format 0xAB 0xCD 0xEF.＞＞デバイスIDは、0xAB 0xCD0xEF形式の3バイトの16進データです。

### ＜＜★Modify Initialization Flow Template＞＞初期化フローテンプレートの変更



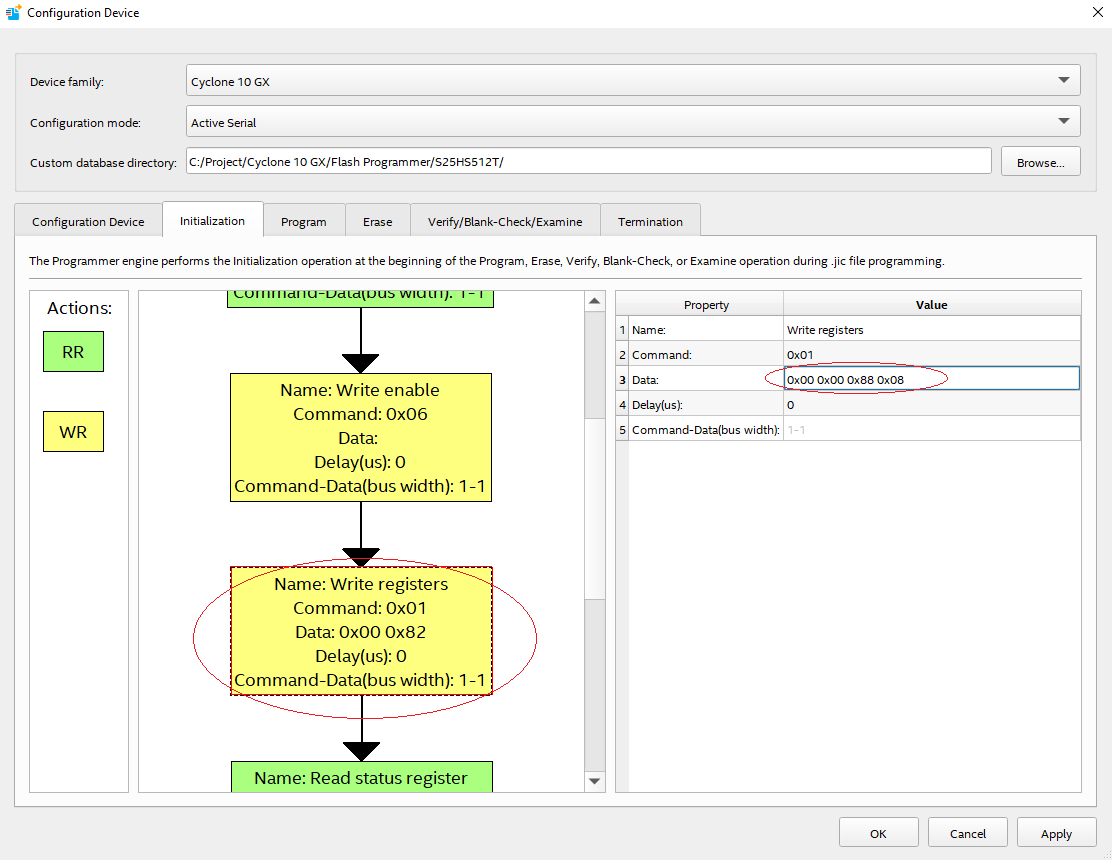
* 1. ＜＜★Select the Configuration Device that you created (i.e., S25HS512T), and then click the **Initialization** tab.＞＞作成した構成デバイス（S25HS512Tなど）を選択し、[初期化]タブをクリックします。

＜＜★The Initialization tab shows the flash initialization flow template.＞＞[初期化]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。＜＜★You can add, delete, or modify an action in the template.＞＞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。＜＜★This shows the changes for the S25HS512T device based on the default flow template.＞＞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ています。＜＜★You can define your own initialization flow template if required.＞＞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。



* 1. ＜＜★Click the **Read ID** action in the left pane.＞＞左側のペインで[IDの読み取り]アクションをクリックします。＜＜★Configure the properties on the right (in green circle) according to the flash device datasheet.＞＞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。
  2. ＜＜★Click **Apply** after action properties are configured.＞＞アクションのプロパティを設定したら、[適用]をクリックします。

＜＜★The default “Write enable” action is the same as that of S25HS512T; this field is not modified for this example.＞＞デフォルトの「書き込みイネーブル」アクションは、S25HS512Tのアクションと同じです。この例では、このフィールドは変更されていません。



1. ＜＜★Scroll down the left window, and then click the **Write registers** action.＞＞左側のウィンドウを下にスクロールして、[レジスタの書き込み]アクションをクリックします。＜＜★Edit the properties to configure the flash register.＞＞プロパティを編集して、フラッシュレジスタを設定します。

＜＜★In this example, configure the S25HS512T to 4-byte address mode, and set the sector architecture as uniform 256KB.＞＞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に設定します。

＜＜★The S25HS512T Write Register command sequence is: Write Register Command (0x01) + STR1 (Status Register-1) value + CFR1 (Configuration Register-1) value + CFR2 (Configuration Register-2) value + CFR3 (Configuration Register-3) value.＞＞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（構成レジスタ-1）値+ CFR2（構成レジスタ-2）値+ CFR3（構成レジスタ-3）値。

＜＜★The STR1 default value is 0x00; CFR1 default is 0x00.＞＞STR1のデフォルト値は0x00です。CFR1のデフォルトは0x00です。

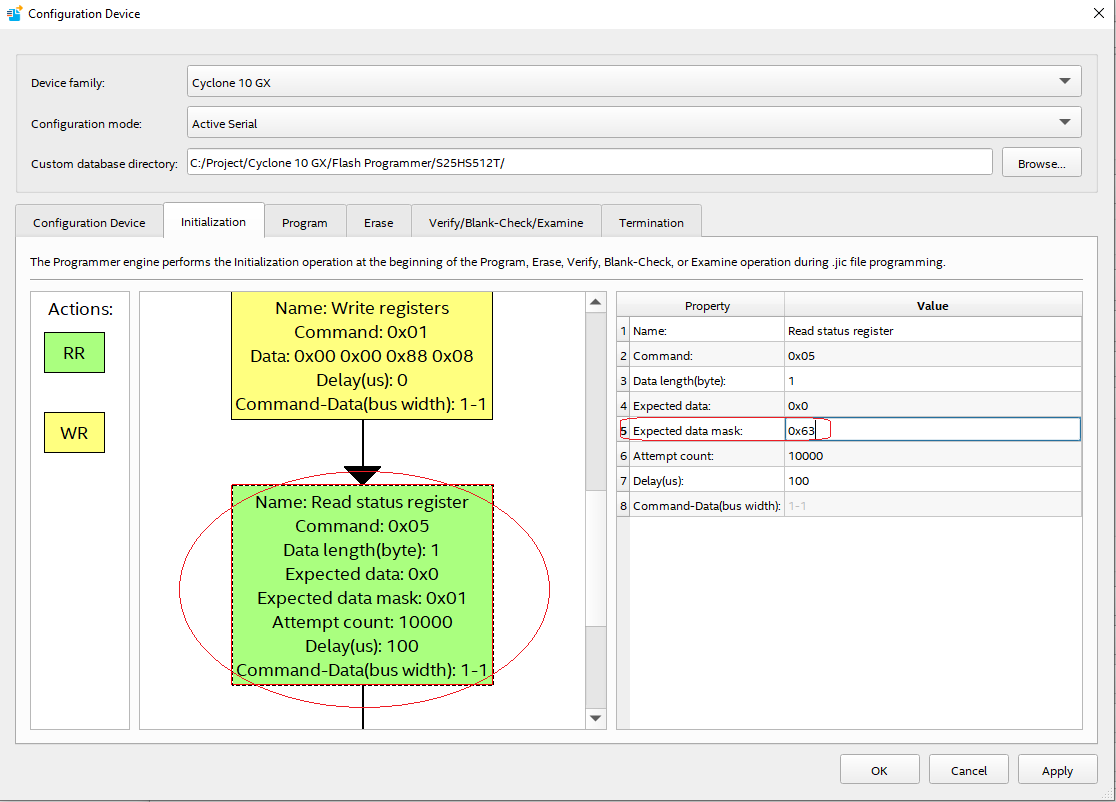
＜＜★The CFR2 default value is 0x08.＞＞CFR2のデフォルト値は0x08です。＜＜★Set CFR2[7] to ‘1’ for 4-byte address mode.＞＞4バイトアドレスモードの場合は、CFR2 [7]を「1」に設定します。＜＜★The new value for CFR2 is 0x88.＞＞CFR2の新しい値は0x88です。

＜＜★The CFR3 default value is 0x00 (Hybrid Sector Architecture 4KB + 256KB sectors).＞＞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。＜＜★Set CFR3[3] to ‘1’ for uniform 256KB sector.＞＞均一な256KBセクターの場合は、CFR3 [3]を「1」に設定します。＜＜★The new value for CFR3 is 0x08.＞＞CFR3の新しい値は0x08です。

＜＜★Thus, the data that follows the 0x01 command is 0x00 0x00 0x88 0x08 (as shown in above screenshot in the red circle on the right side).＞＞したがって、0x01コマンドに続くデータは0x00 0x00 0x88 0x08です（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）。＜＜★See the S25HS512T datasheet for register details.＞＞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。

**＜＜★Notes**:＞＞ノート：

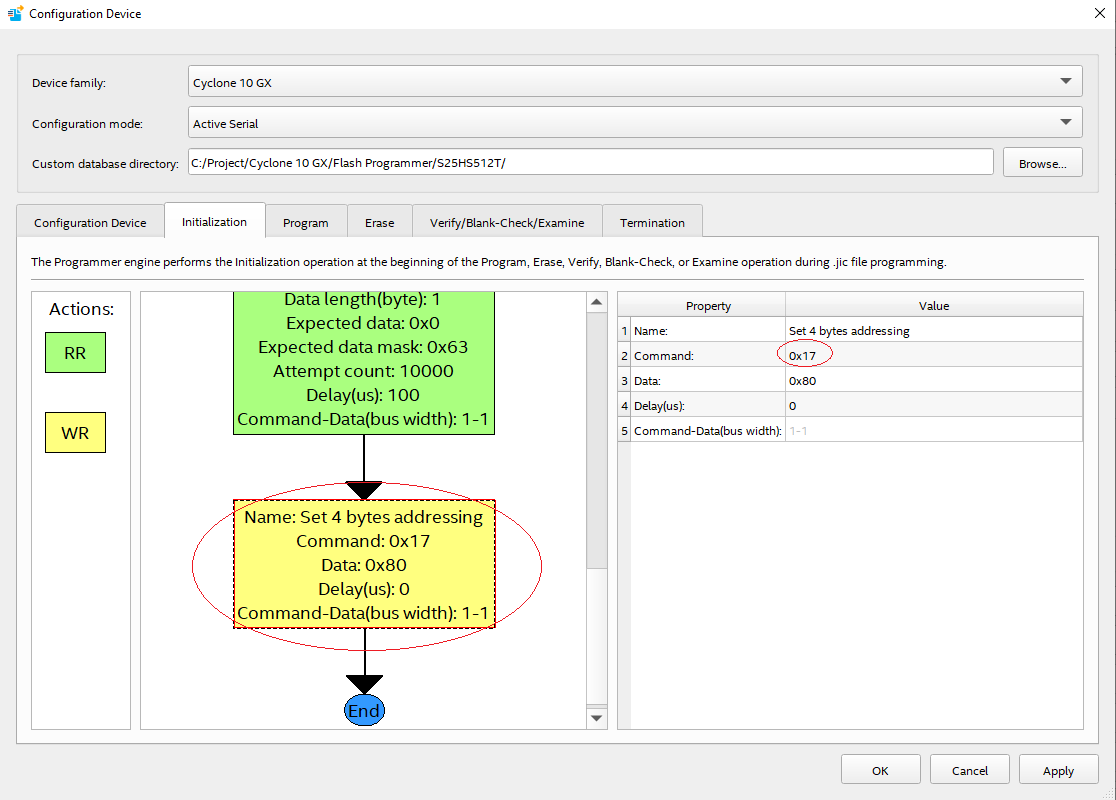
* ＜＜★［1］Intel Cyclone 10 GX and Arria 10 devices require 4-byte addressing flash memory; the flash must support 4-byte addressing and should be configured to 4-byte address mode.＞＞Intel Cyclone 10GXおよびArria10デバイスには、4バイトのアドレス指定フラッシュメモリが必要です。フラッシュは4バイトアドレス指定をサポートする必要があり、4バイトアドレスモードに設定する必要があります。
* ＜＜★The default erase flow template is for uniform 256KB sector architecture.＞＞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。＜＜★To simplify the erase flow, set the flash as uniform 256KB sector architecture in the initialization flow.＞＞消去フローを単純化するには、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。＜＜★If not, the erase flow will not be able to completely erase the flash array; note that this may happen without an error message.＞＞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。



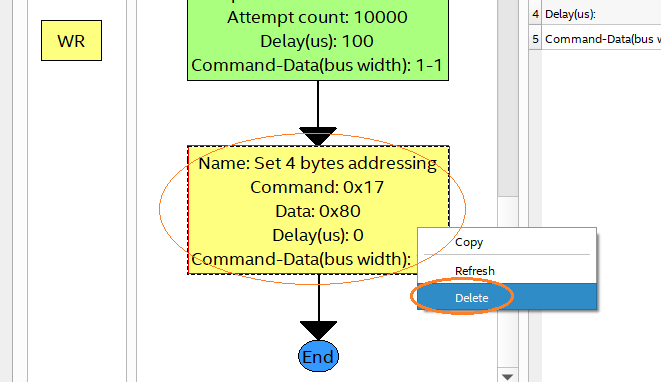
1. ＜＜★After Write Register action, read the status register to check the status of the write operation.＞＞書き込みレジスタアクションの後、ステータスレジスタを読み取り、書き込み操作のステータスを確認します。

＜＜★STR1[6] is P\_ERR, STR1[5] is E\_ERR, STR1[1] is WEL, STR1[0] is RDYBSY.＞＞STR1 [6]はP\_ERR、STR1 [5]はE\_ERR、STR1 [1]はWEL、STR1 [0]はRDYBSYです。＜＜★When the program operation completes successfully, these four bits should be all 0s; other bits are don’t care.＞＞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。他のビットは気にしません。＜＜★Therefore, the Expected data mask is 0110 0011b (0x63).＞＞したがって、期待されるデータマスクは0110 0011b（0x63）です。＜＜★The masked expected data is 0x0.＞＞マスクされた期待データは0x0です。＜＜★See the S25HS512T datasheet for definition of Status Register bits.＞＞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。

＜＜★This Action (Read Status Register) is the same for checking the status of the flash array program/erase operation.＞＞このアクション（ステータスレジスタの読み取り）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。＜＜★It can also be used in Program and Erase flow template.＞＞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。

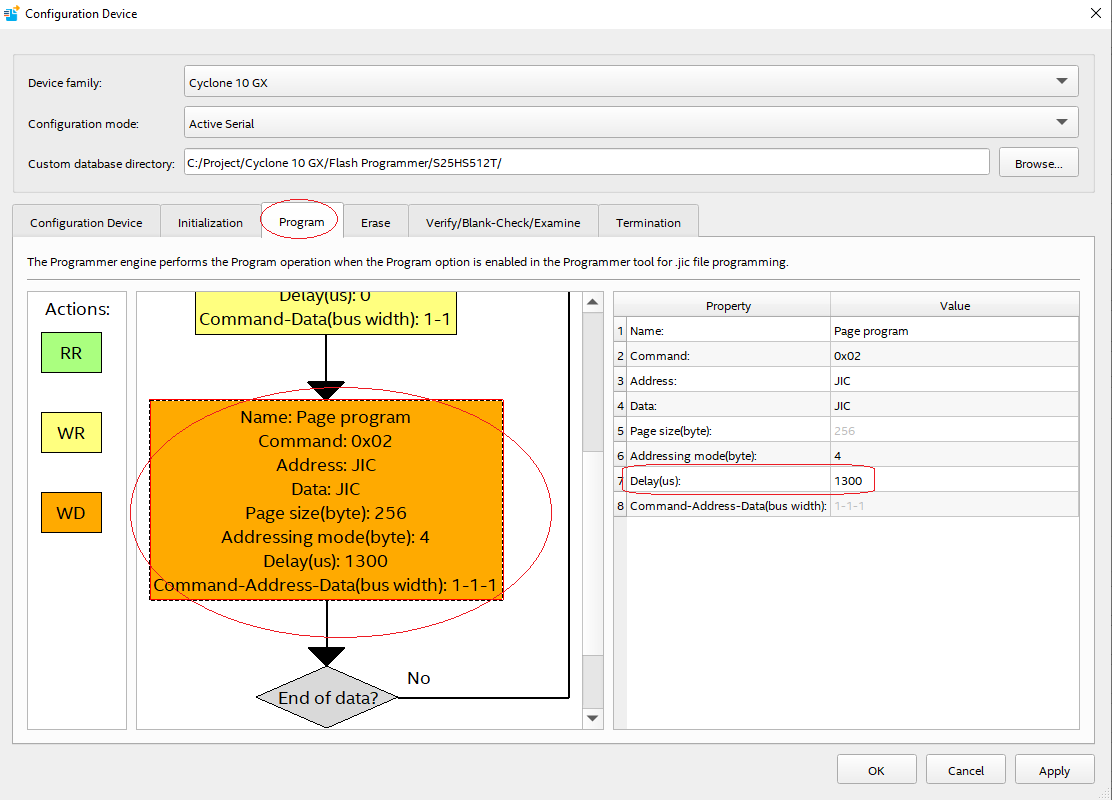


1. ＜＜★Right-click the **Set 4 bytes addressing** action and click **Delete**, and then click **Apply** to save the changes. This is because this command is not applicable to Cypress flash devices.＞＞[4バイトのアドレス指定を設定]アクションを右クリックして[削除]をクリックし、[適用]をクリックして変更を保存します。これは、このコマンドがサイプレスフラッシュデバイスに適用できないためです。＜＜★［1］For Cypress flash (S25HS512T), setting 4-byte addressing already done in the previous Write Register action.＞＞サイプレスフラッシュ（S25HS512T）の場合、前の書き込みレジスタアクションですでに行われた4バイトアドレス指定を設定します。



### ＜＜★Modify Program Flow Template＞＞プログラムフローテンプレートの変更

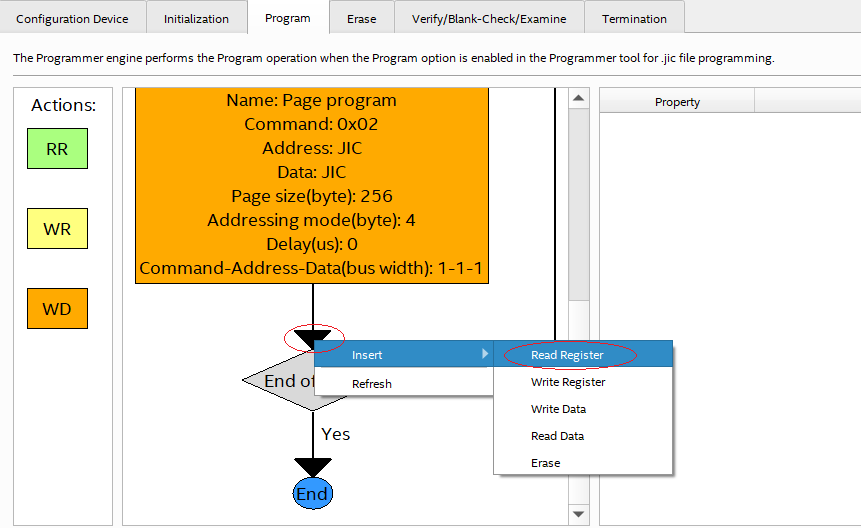
* 1. ＜＜★Click the **Program** tab, and then click the **Page Program** action.＞＞[プログラム]タブをクリックし、[ページプログラム]アクションをクリックします。



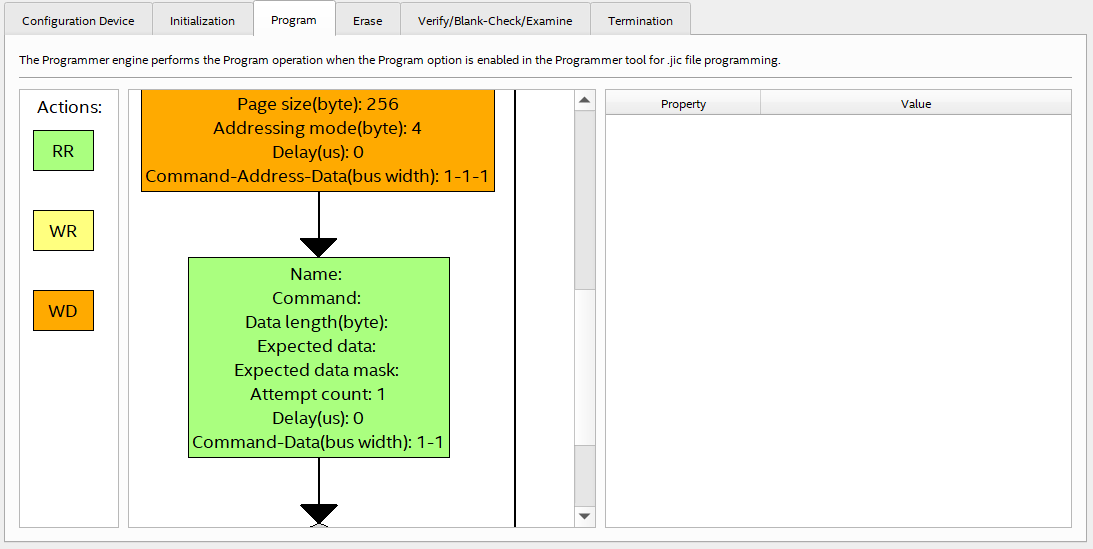
1. ＜＜★Modify the **Delay** value according to the maximum Page Program time specified in the flash datasheet.＞＞フラッシュデータシートで指定されている最大ページプログラム時間に従って、遅延値を変更します。＜＜★For example, for S25HS512T, the maximum Page Program time is 2175 µs.＞＞たとえば、S25HS512Tの場合、最大ページプログラム時間は2175 µsです。

＜＜★［1］Another option is to set the Delay value to ‘0’ and add a Read Status Register action to check the status of the Page Program operation as follows.＞＞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取り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。

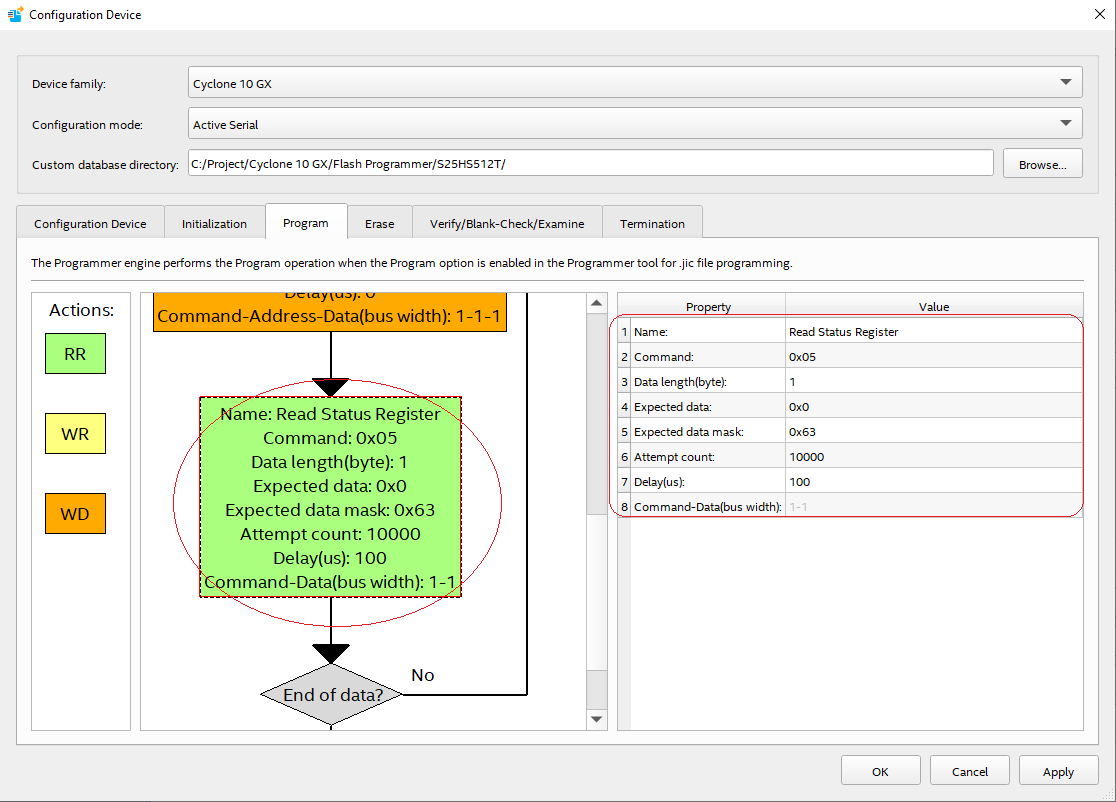
＜＜★The Page Program size is the same for the default Program flow template and S25HS512T: 256 bytes.＞＞ページプログラムのサイズは、デフォルトのプログラムフローテンプレートとS25HS512Tで同じです：256バイト。＜＜★S25HS512T also supports larger Page Program size of 512 bytes.＞＞S25HS512Tは、512バイトのより大きなページプログラムサイズもサポートします。＜＜★You can configure this value to 512 bytes by setting CFR3x[4]=1.＞＞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。＜＜★To use the Page Program size of 512 bytes for S25HS512T, configure the value in the Write Register action in the Initialization Flow template.＞＞S25HS512Tに512バイトのページプログラムサイズを使用するには、初期化フローテンプレートの書き込みレジスタアクションで値を設定します。



1. ＜＜★Right-click the arrow, and select **Insert** > **Read Register** toadd the Read Register action to the flow.＞＞矢印を右クリックし、**Insert** > **Read Register** を選択して、[レジスタの読み取り]アクションをフローに追加します。



1. ＜＜★Click the action that you added and configure its properties.＞＞追加したアクションをクリックして、そのプロパティを構成します。

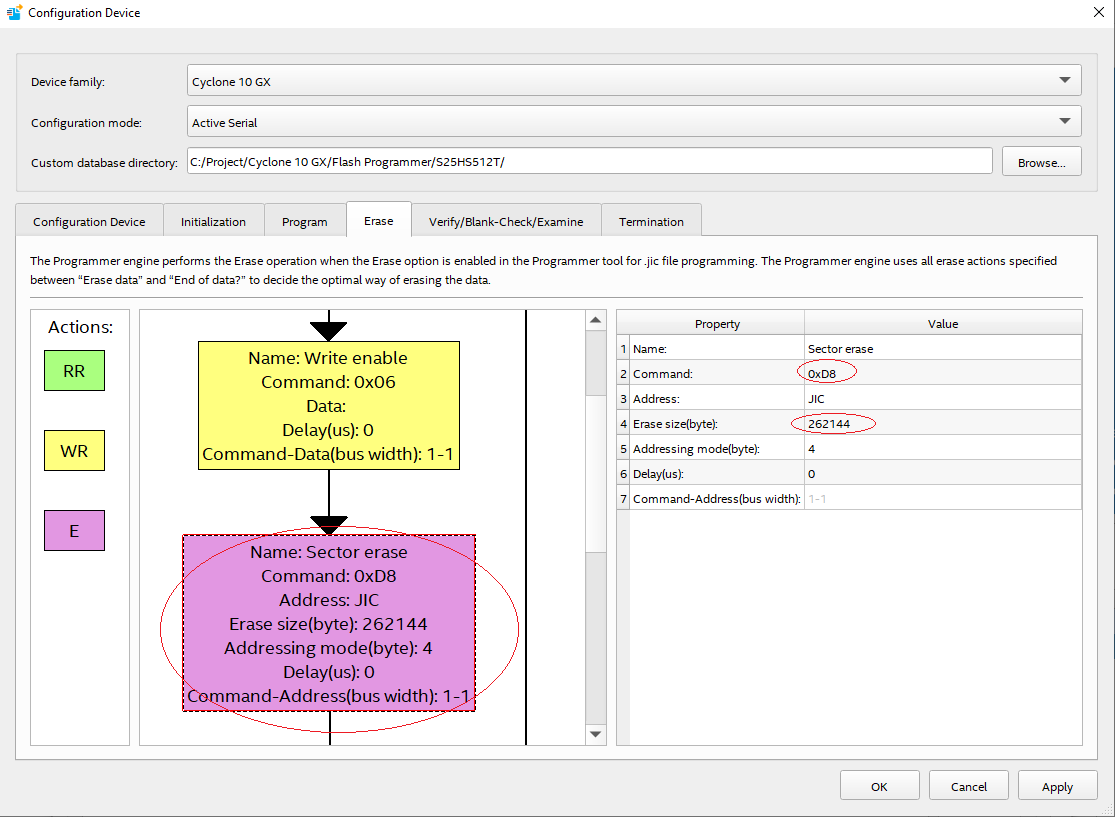


1. ＜＜★Configure the properties according to the flash datasheet. This is the same as the configuration shown in the Initialization template.＞＞フラッシュデータシートに従ってプロパティを構成します。これは、初期化テンプレートに示されている構成と同じです。

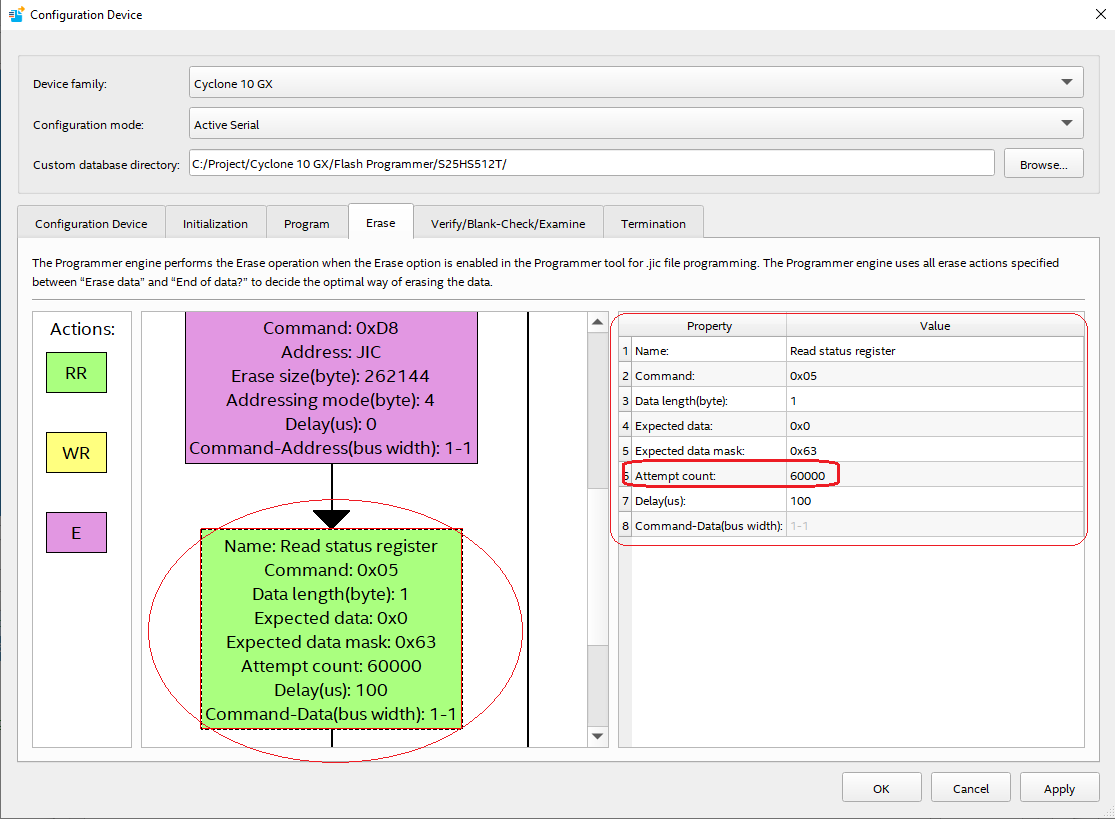
＜＜★The time (Delay x Attempt count) must be larger than the maximum Page Program time specified in the flash datasheet.＞＞時間（遅延x試行回数）は、フラッシュデータシートで指定されている最大ページプログラム時間よりも大きくする必要があります。＜＜★For example, for S25HS512T, the maximum Page Program time is 2175 µs; therefore, the Delay x Attempt count value must be larger than 2175us.＞＞たとえば、S25HS512Tの場合、最大ページプログラム時間は2175 µsです。したがって、Delay xAttemptカウント値は2175usより大きくなければなりません。＜＜★You can enter a larger value because after the Read Register Action returns the Expected masked data, the flow will go to the next step without reaching the maximum attempt count.＞＞読み取りレジスタ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。

1. ＜＜★After configuring the properties, click **Apply**.＞＞プロパティを構成したら、[適用]をクリックします。

### ＜＜★Modify Erase Flow Template＞＞消去フローテンプレートの変更



＜＜★The default “Write enable” and “Sector erase” actions are compatible with S25HS512T uniform 256KB sector erase operation.＞＞デフォルトの「書き込みイネーブル」および「セクター消去」アクションは、S25HS512Tの均一な256KBセクター消去操作と互換性があります。＜＜★You can leave these values unchanged because S25HS512T is already configured to uniform 256KB sector architecture in the Write Register action in Initialization flow template.＞＞S25HS512Tは、初期化フローテンプレートのレジスタ書き込みアクションで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おくことができます。



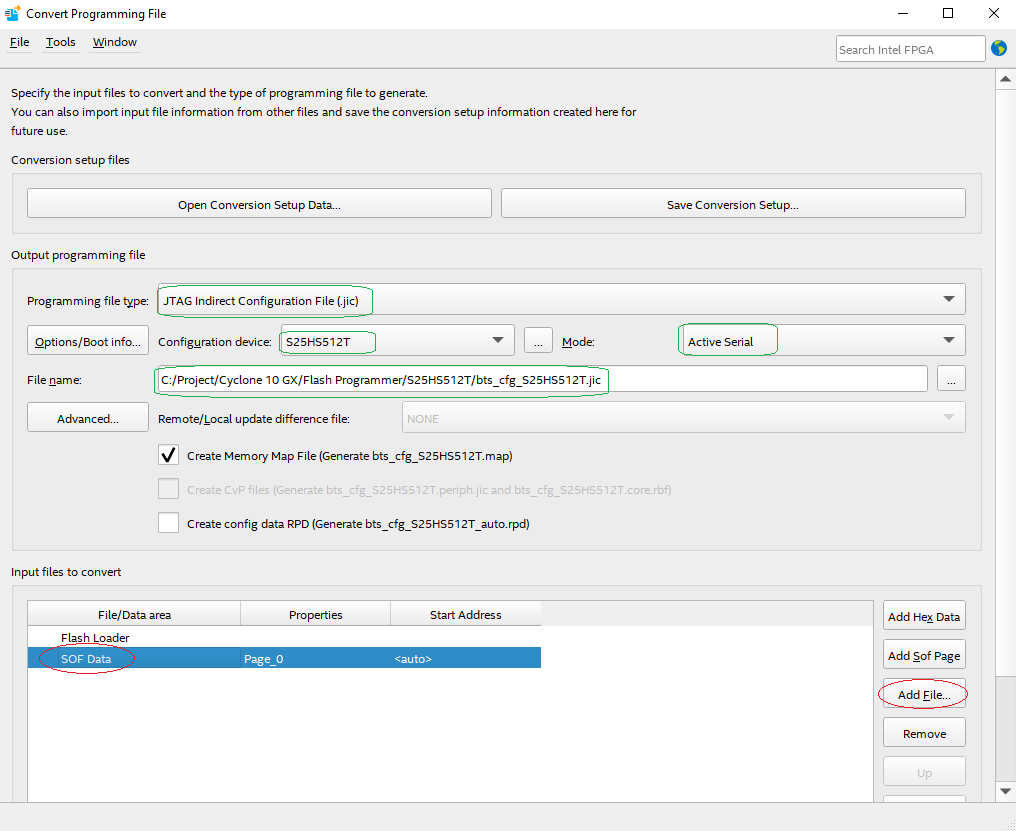
1. ＜＜★Click the **Read status register** action, and update the properties for S25HS512T such as command, expected data, and data mask.＞＞ステータスレジスタの読み取りアクションをクリックし、コマンド、期待されるデータ、データマスクなどのS25HS512Tのプロパティを更新します。＜＜★This action is similar to the **Read status register** action in the Initialization and Program flows.＞＞このアクションは、初期化フローおよびプログラムフローのステータスレジスタの読み取りアクションに似ています。

**＜＜★Note**: The (Delay x Attempt count) must be larger than the maximum 256KB sector erase time specified in the flash datasheet.＞＞注：（遅延x試行回数）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。＜＜★For S25HS512T, it is 5869 ms if the 256KB EnduraFlex™ architecture is enabled.＞＞S25HS512Tの場合、256KBEnduraFlex™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。＜＜★See S25HS512T the datasheet for details.＞＞詳細については、S25HS512Tのデータシートを参照してください。

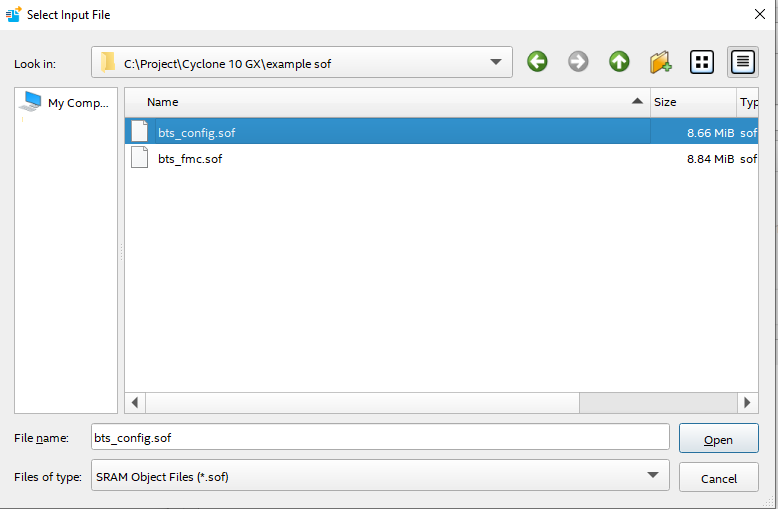
1. ＜＜★Click **OK** to save changes.＞＞[OK]をクリックして変更を保存します。

＜＜★After these actions are configured, set up the Configuration Device to convert an .sof file to a .jic file that can be loaded to the flash device.＞＞これらのアクションを構成した後、.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ます。

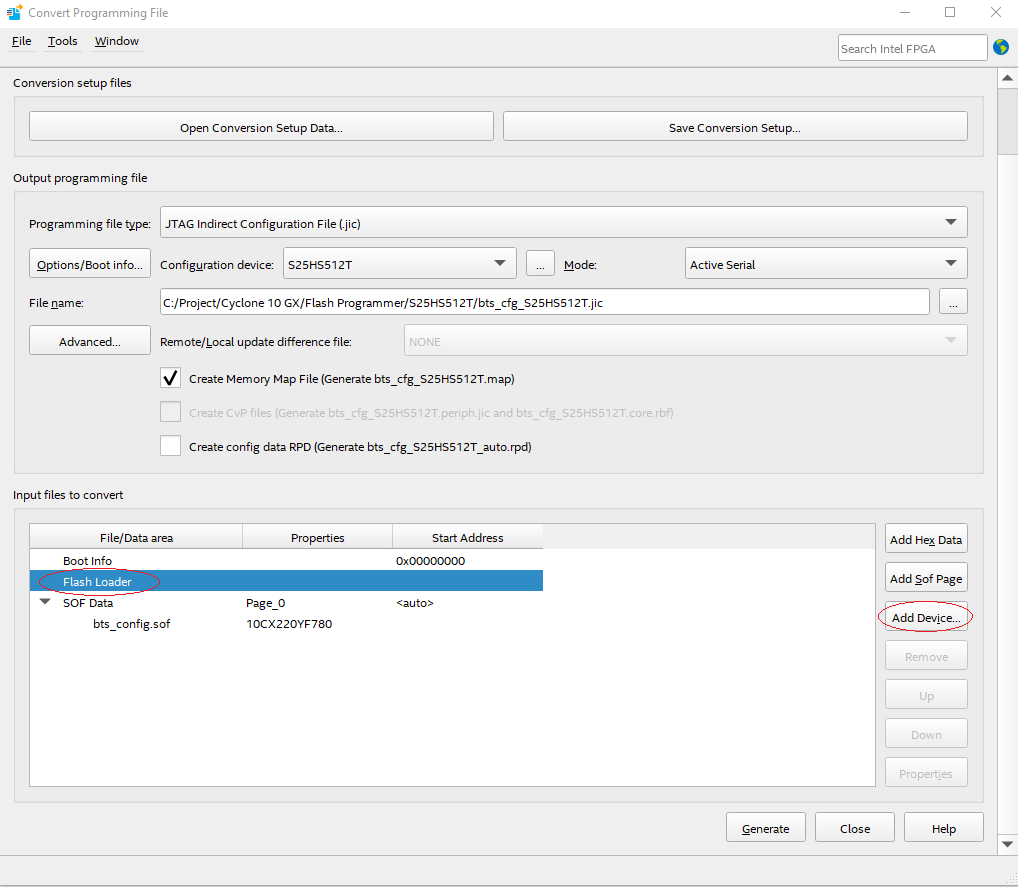
## ＜＜★Convert the .sof File to Generate the .jic File＞＞.sofファイルを変換して.jicファイルを生成する



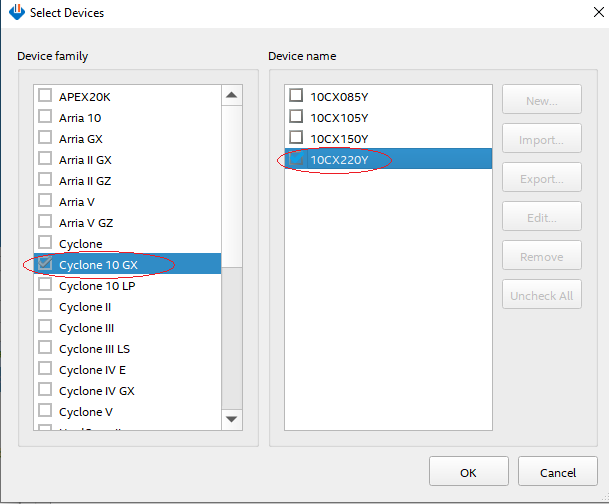
1. ＜＜★Set **Programming file type** to JTAG Indirect Configuration File (.jic).＞＞プログラミングファイルタイプをJTAG間接設定ファイル（.jic）に設定します。
2. ＜＜★Select S25HS512T (created in previous steps) as the Configuration device.＞＞構成デバイスとしてS25HS512T（前の手順で作成）を選択します。
3. ＜＜★Select mode as **Active Serial**.＞＞モードをアクティブシリアルとして選択します。
4. ＜＜★Specify the .jic file name to be generated and the location where the generated .jic file will be stored.＞＞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ます。
5. ＜＜★Select **SOF Data**, and then click **Add File…**.＞＞[SOFデータ]を選択し、[ファイルの追加...]をクリックします。
6. ＜＜★Select the .sof file you are going to convert.＞＞変換する.sofファイルを選択します。



1. ＜＜★After adding the .sof file, select **Flash Loader**, and then click **Add Device…**.＞＞.sofファイルを追加した後、[フラッシュローダー]を選択し、[デバイスの追加...]をクリックします。

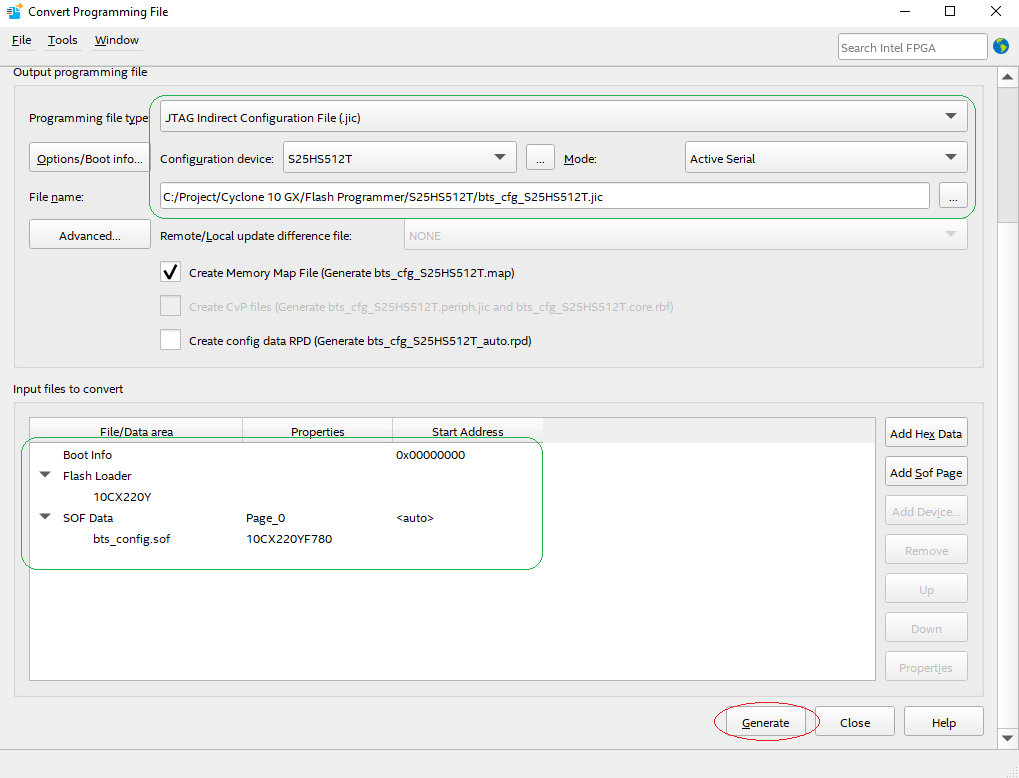


1. ＜＜★Select the device according to the FPGA device you are working on, and click **OK**.＞＞作業しているFPGAデバイスに応じてデバイスを選択し、[OK]をクリックします。

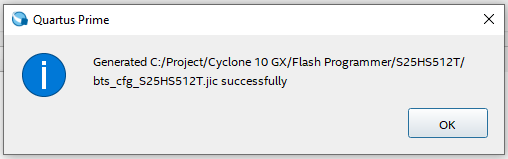


＜＜★［1］In this application note, the instructions are verified on Cyclone10 GX Dev Kit with Cyclone10 GX 10CX220Y FPGA device.＞＞このアプリケーションノートでは、Cyclone10 GX 10CX220YFPGAデバイスを搭載したCyclone10GXDevキットで手順を確認します。

1. ＜＜★Click **Generate**.＞＞[生成]をクリックします。



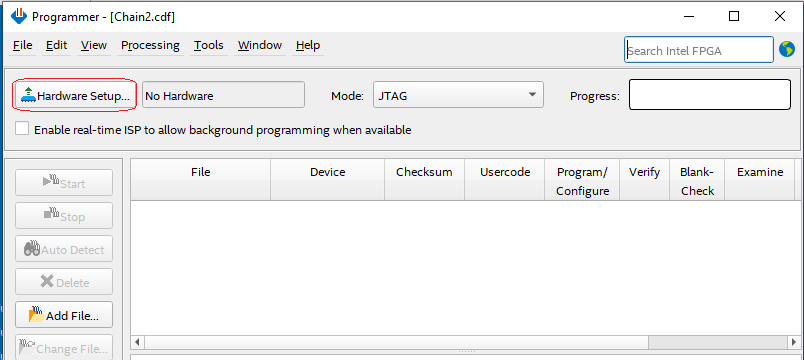
＜＜★After a little while, you will see a popup window that indicates that the .jic file is generated successfully.＞＞しばらくすると、.jicファイルが正常に生成されたことを示すポップアップウィンドウが表示されます。



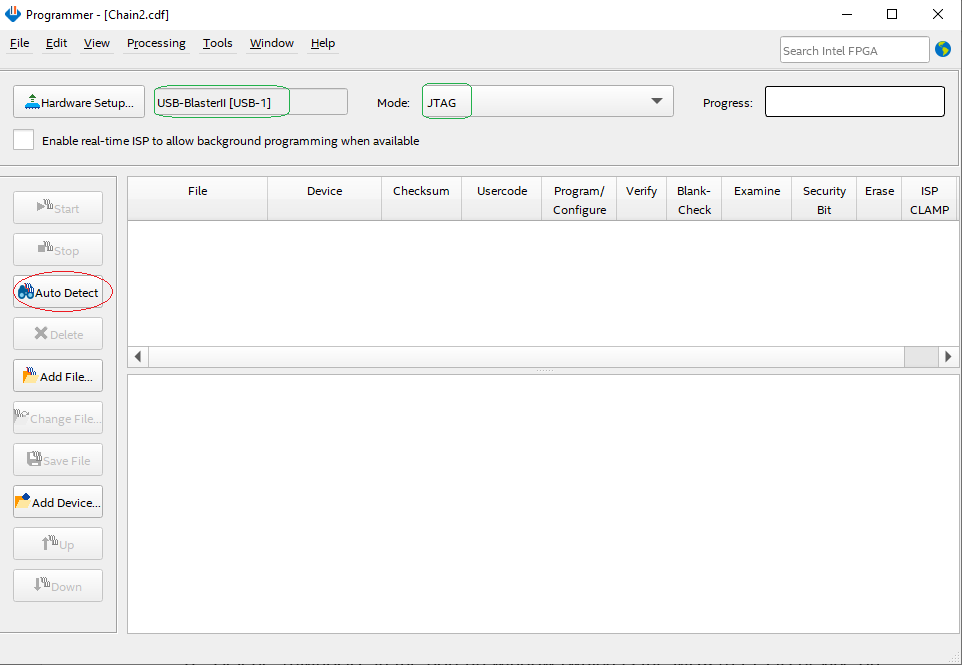
＜＜★You can now program the generated .jic file into the configuration flash (S25HS512T in this example).＞＞これで、生成された.jicファイルを構成フラッシュ（この例ではS25HS512T）にプログラムできます。

## ＜＜★Program the .jic Configuration Data into the Cypress QSPI Flash Device＞＞.jic構成データをサイプレスQSPIフラッシュデバイスにプログラムします

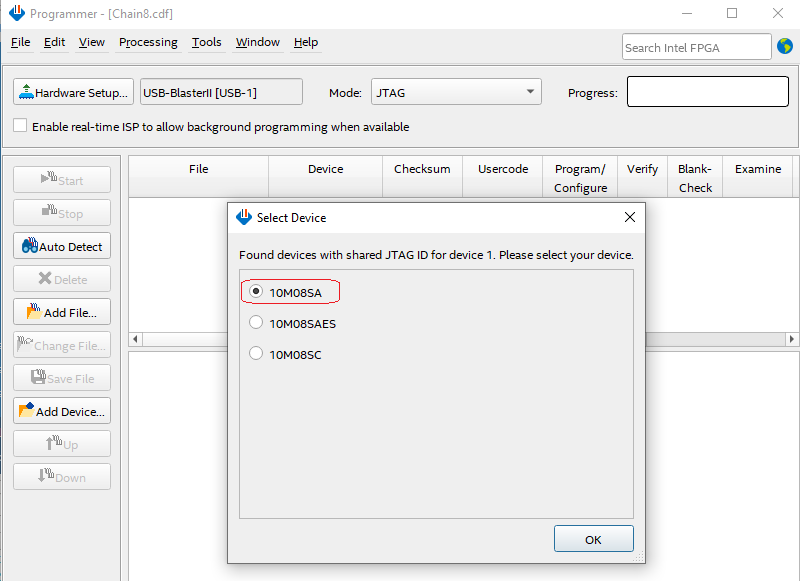
1. ＜＜★Connect Intel FPGA board to PC via Intel FPGA JTAG USB Download cable, and power on the FPGA board.＞＞Intel FPGA JTAG USBダウンロードケーブルを介してIntelFPGAボードをPCに接続し、FPGAボードの電源を入れます。
2. ＜＜★In Quartus Prime Pro, select **Tools** > **Programmer**.＞＞Quartus Prime Proで、**Tools** > **Programmer** を選択します。
3. ＜＜★Click **Hardware Setup…**.＞＞[ハードウェア設定...]をクリックします。



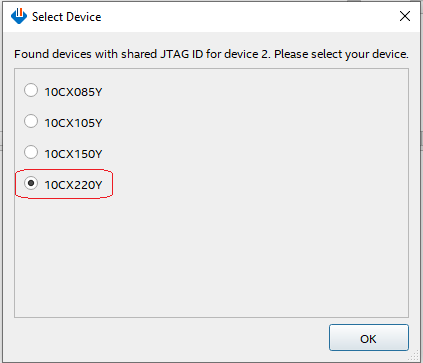
1. ＜＜★Click **USB-Blaster…**.＞＞USB-Blaster…をクリックします。



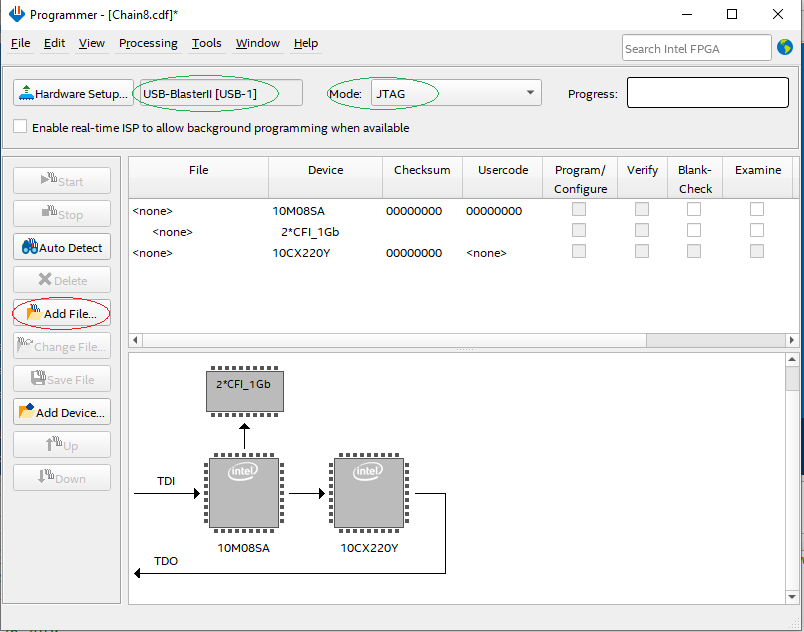
1. ＜＜★Click **Auto Detect**.＞＞[自動検出]をクリックします。
2. ＜＜★Select **10M08SA** in the pop-up window.＞＞ポップアップウィンドウで10M08SAを選択します。＜＜★［1］This is the MAX10 FPGA device on the Cyclone10 GX dev board on which the procedures were verified.＞＞これは、手順が検証されたCyclone10GX開発ボード上のMAX10FP​​GAデバイスです。



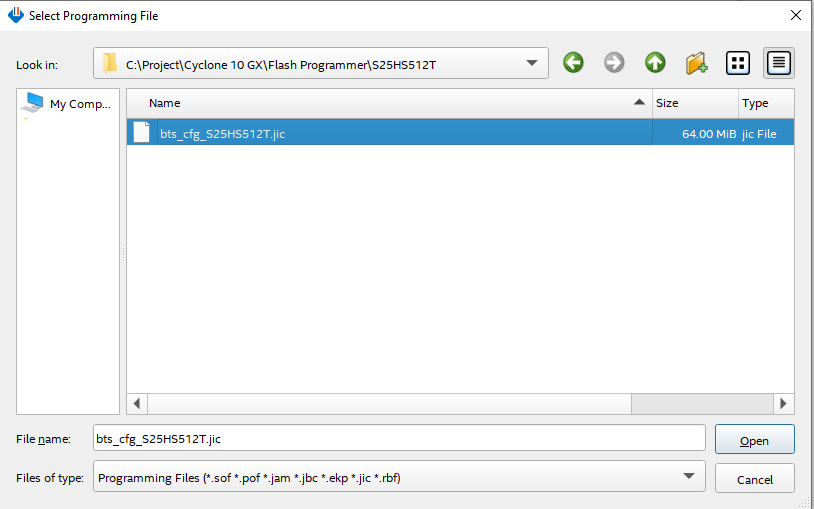
1. ＜＜★Select **10CX220Y** in the pop-up window.＞＞ポップアップウィンドウで10CX220Yを選択します。＜＜★［1］This is the Cyclone 10 GX FPGA device on the development board.＞＞これは、開発ボード上のCyclone 10 GXFPGAデバイスです。



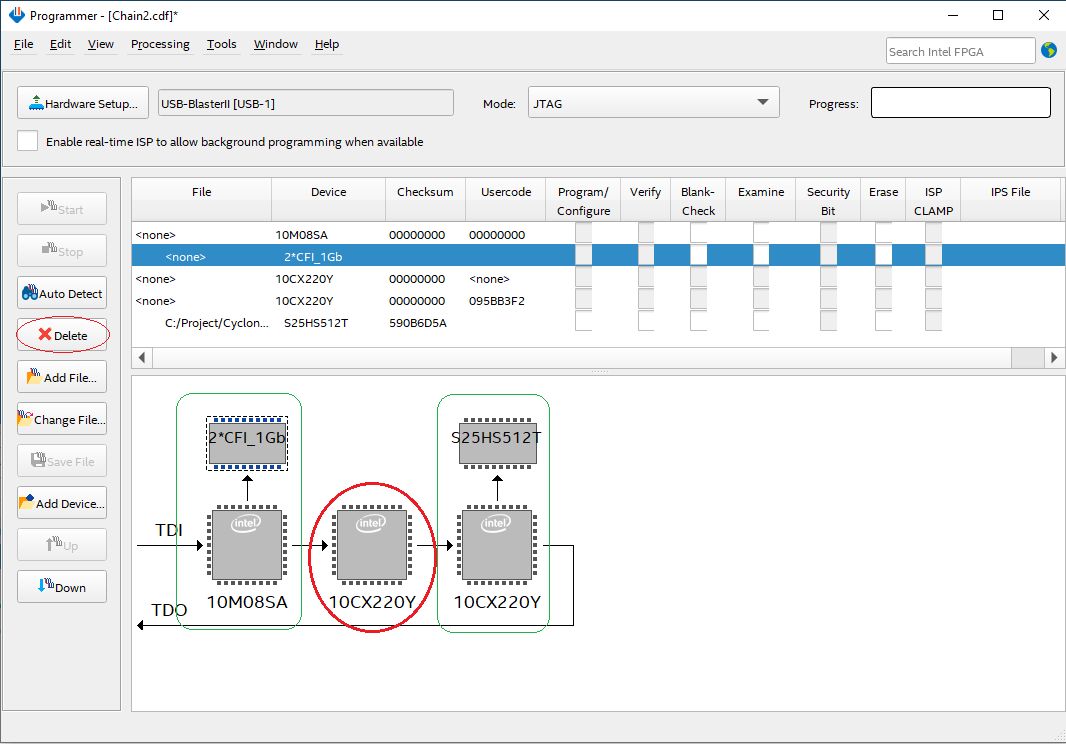
1. ＜＜★Click **Add File…**.＞＞[ファイルの追加...]をクリックします。



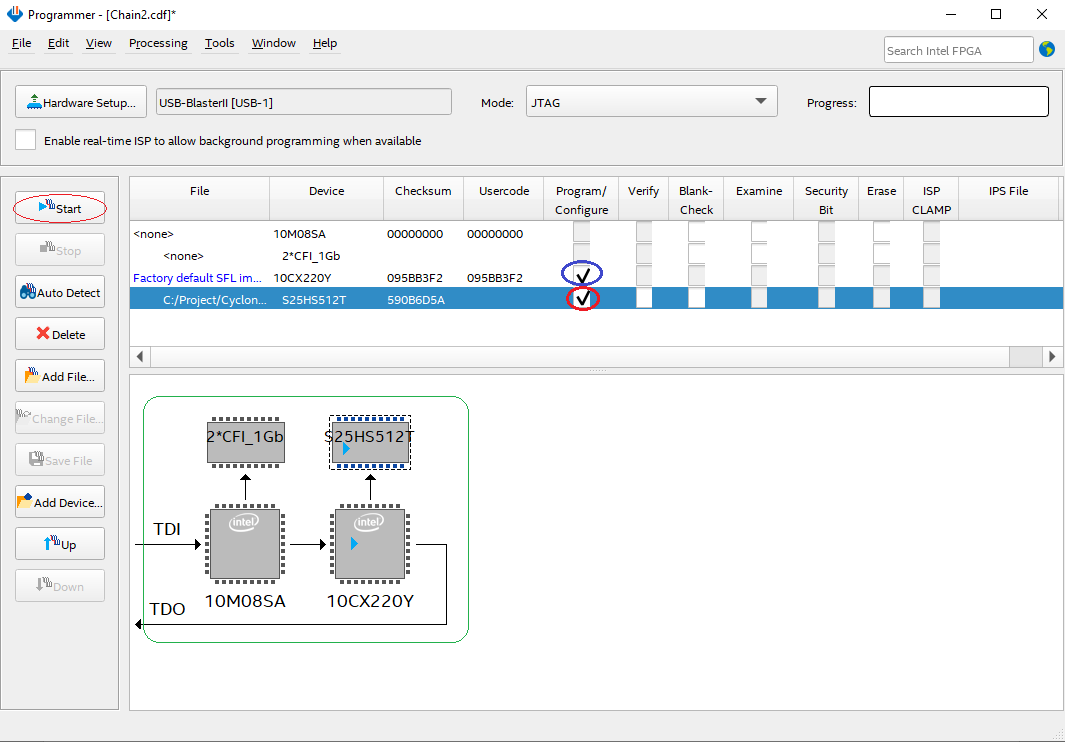
1. ＜＜★Open the .jic file generated in the earlier steps.＞＞前の手順で生成された.jicファイルを開きます。



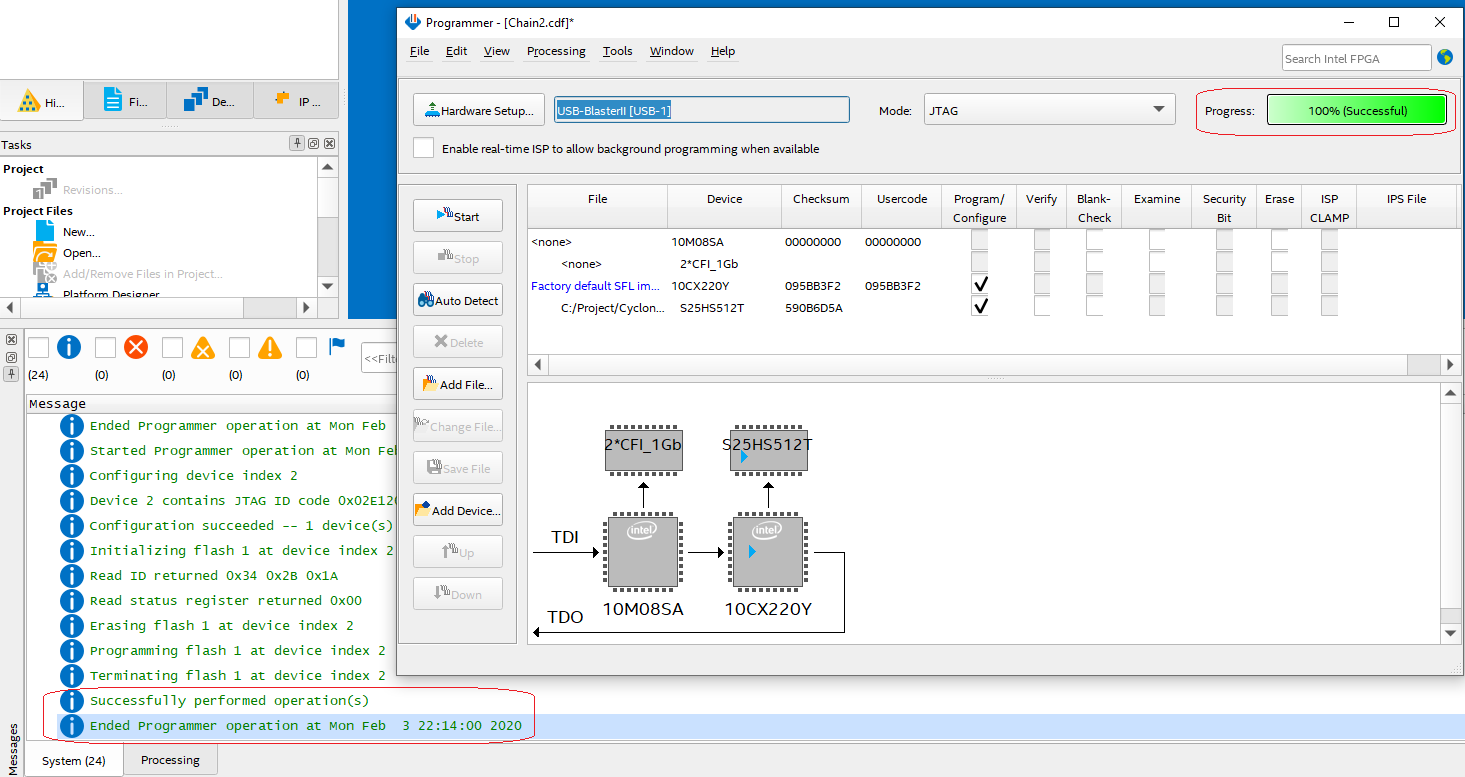
1. ＜＜★If an extra device appears after adding the .jic file, select it and then click **Delete**.＞＞.jicファイルを追加した後に追加のデバイスが表示された場合は、それを選択して[削除]をクリックします。



1. ＜＜★Enable the **Program/Configure** option (check box in the red circle; the check box in the blue circle will be also enabled automatically).＞＞[プログラム/構成]オプションを有効にします（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）。



1. ＜＜★Click **Start**.＞＞[開始]をクリックします。



1. ＜＜★When the progress bar shows 100% (Successful) and the message shows “Successfully performed operations”, power cycle the target board.＞＞プログレスバーに100％（Successful）と表示され、メッセージに「Successfully performsoperations」と表示されたら、ターゲットボードの電源を入れ直します。

# 結論

＜＜★［1］This application note describes how to enable Cypress S25HS512T QSPI Semper Flash to configure an Intel Cyclone 10 GX FPGA device.＞＞このアプリケーションノートでは、Cypress S25HS512T QSPI SemperFlashを有効にしてIntelCyclone 10 GXFPGAデバイスを構成する方法について説明します。

＜＜★The procedures introduced in this application note should also apply to other Cypress QSPI Flash families and Intel Arria 10 FPGA devices by configuring the flash and FPGA device according to respective datasheet.＞＞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、他のサイプレス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

* ＜＜★［1］Intel Cyclone 10 GX FPGA Development Kit User Guide＞＞Intel Cyclone 10 GXFPGA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 User Guide: Intel Quartus Prime Pro Edition＞＞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>

* ＜＜★Cypress S25HS-T datasheet＞＞サイプレスS25HS-Tデータシート

改訂履歴

＜＜★［1］Document Title: AN229767 – Enabling Cypress QSPI Flash to Configure Intel Cyclone 10 GX FPGA＞＞ドキュメントタイトル：AN229767 – CypressQSPIフラッシュを有効にしてIntelCyclone 10 GXFPGAを構成する

＜＜★Document Number: 002-29767

| 版 | Engineering Change Notification (技術変更届) | 発行日 | 変更内容 |
| --- | --- | --- | --- |
|  |  |  |  |