FlashPro Express User Guide

Libero SoC v2025.2

Introduction (Ask a Question)

FlashPro Express is Microchip's programming software tool designed from the ground up to address secured programming assurance in production programming house environments. FlashPro Express software supports PolarFire®, PolarFire SoC, SmartFusion® 2, IGLOO® 2, and RTG4™ in Windows® and Linux® OS environments, using the FlashPro Programmer hardware.

You can install FlashPro Express two ways:

- **Integrated with Libero**- FlashPro Express installs automatically when Libero® is installed. FlashPro Express is used by Libero to perform the programming tasks for PolarFire, PolarFire SoC, SmartFusion 2, IGLOO 2, and RTG4, as part of the design flow.
- **Stand-alone**- FlashPro Express is also available as a stand-alone installation. This installation method is primarily used for production programming or lab programming on machines that do not require a full version of Libero.

Supported Device Families

The following table lists the device families and their derivatives that FlashPro Express can program directly through Libero or by exporting a FlashPro Express job.

Table 1. Product Families and Derivatives Directly Supported by FlashPro Express

Device Family	Description
PolarFire®	Lowest power, cost optimized mid-range solution.
PolarFire SoC	Lowest power, multi-core RISC-V SoC FPGA.
SmartFusion® 2	Addresses fundamental requirements for advanced security, high reliability, and low power in critical industrial, military, aviation, communications, and medical applications.
IGLOO® 2	Low-power mixed-signal programmable solution.
RTG4™	Radiation-tolerant programmable solution.

For more information about FlashPro Express, see the Microchip Website.

Table of Contents

1. Overview (Ask a Question)

This section provides an overview of FlashPro Express.

1.1. FlashPro Hardware Programmers (Ask a Question)

The FlashPro series of hardware programmers consists of:

- FlashPro3/X
- FlashPro4
- FlashPro5
- FlashPro6

All FlashPro series hardware programmers save board space because a single JTAG chain can be used for all JTAG devices. In-system programming using the JTAG port adds the flexibility of field upgrades or post-assembly production-line characterization. Production costs are reduced significantly as a result of elimination of expensive sockets on the board.

Important: FlashPro5 and FlashPro6 support programming though a device SPI Target port.

For more information, visit the Microchip Website.

1.2. Secure Job Programming (Ask a Question)

Job programming is the concept of using a single file to program a Microchip device or chain of Microchip devices using encrypted bitstreams.

The single job file contains all of the information necessary to setup FlashPro Express as well as the encrypted bitstream images for the devices in the job. After a job file is created, it can be passed securely to production programming houses or contract engineering facilities to load the Microchip images during manufacturing. Job projects can be exported from Libero and imported into stand-alone FlashPro Express, providing a clean delineation between design flow and production programming.

1.3. Migrating FlashPro Projects to FlashPro Express (Ask a Question)

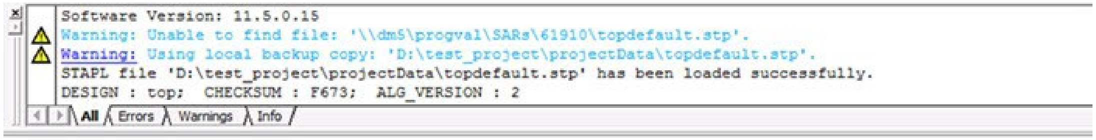
Existing FlashPro projects (`*.pro`) files are called Job Project files in FlashPro Express. These Job Projects can be opened with FlashPro Express to take advantage of Linux programming support and the simplified tool targeted for operators in a production floor environment.

Note: FlashPro projects that were created in single mode are not supported by this tool. Microchip recommends that you convert these projects to chain mode projects.

To convert the project to a chain project:

1. Open the FlashPro project (`*.pro`) in FlashPro.
2. Locate the loaded STAPL file by one of two methods:
   The log prints **STAPL file '<stapl_path>' has been loaded successfully**. **<stapl_path>** is the location of the STAPL file loaded.

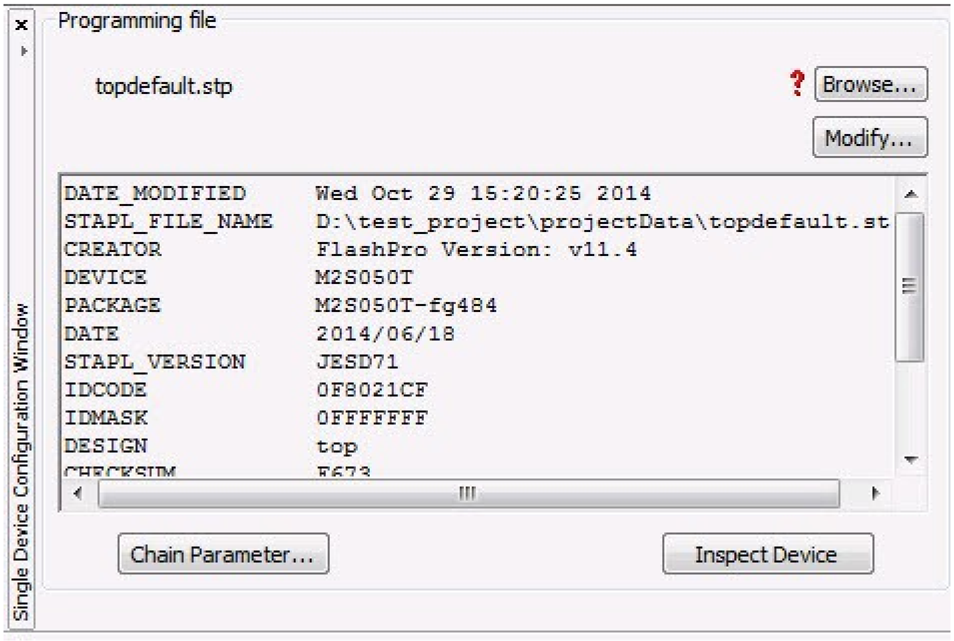
   Figure 1-1. Sample Log Message

   ```
   Software Version: 11.5.0.15
   Warning: Unable to find file: '\\dm5\progval\SARs\61910\topdefault.stp'.
   Warning: Using local backup copy: 'D:\test_project\projectData\topdefault.stp'.
   STAPL file 'D:\test_project\projectData\topdefault.stp' has been loaded successfully.
   DESIGN : top;  CHECKSUM : F673;  ALG_VERSION : 2
   ```

In the Single Device Configuration window, the field **STAPL_FILE_NAME** shows the location of the STAPL file loaded.

Figure 1-2. Location of STAPL File



```
DATE_MODIFIED      Wed Oct 29 15:20:25 2014
STAPL_FILE_NAME    D:\test_project\projectData\topdefault.st
CREATOR            FlashPro Version: v11.4
DEVICE             M2S050T
PACKAGE            M2S050T-fg484
DATE               2014/06/18
STAPL_VERSION      JESD71
IDCODE             0F8021CF
IDMASK             0FFFFFFF
DESIGN             top
CHECKSUM           F673
```

1. Switch the project to chain mode using one of the two methods:
   - Press the chain button from the toolbar:
   - From the **Tools** menu, select **Mode > Chain Programming**.
2. Load the STAPL file in chain mode by adding a Microchip device in the chain.
   a. From the **File** menu, select **Configuration > Add Microchip Devices from Files**.
   b. Browse to the location of the STAPL file and click **Open**.
3. To save the project, from the **File** menu, select **Save Project**.

You can now open the project using FlashPro Express.

When moving FlashPro project (`*.pro`) files to another machine, Microchip recommends that you archive the entire project folder, copy it to the new machine, extract it locally, and then load the job project within FlashPro Express. FlashPro Express opens a job project only when a programmer is connected to the machine, at least one Microchip device has programmed enabled, and all enabled Microchip devices have a bitstream file loaded.

2. Installing FlashPro Express Software and Hardware (Ask a Question)

For information about installing the FlashPro Express hardware and software, see the Microchip Website.

3. Getting Started (Ask a Question)

This section describes how to get started using FlashPro Express.

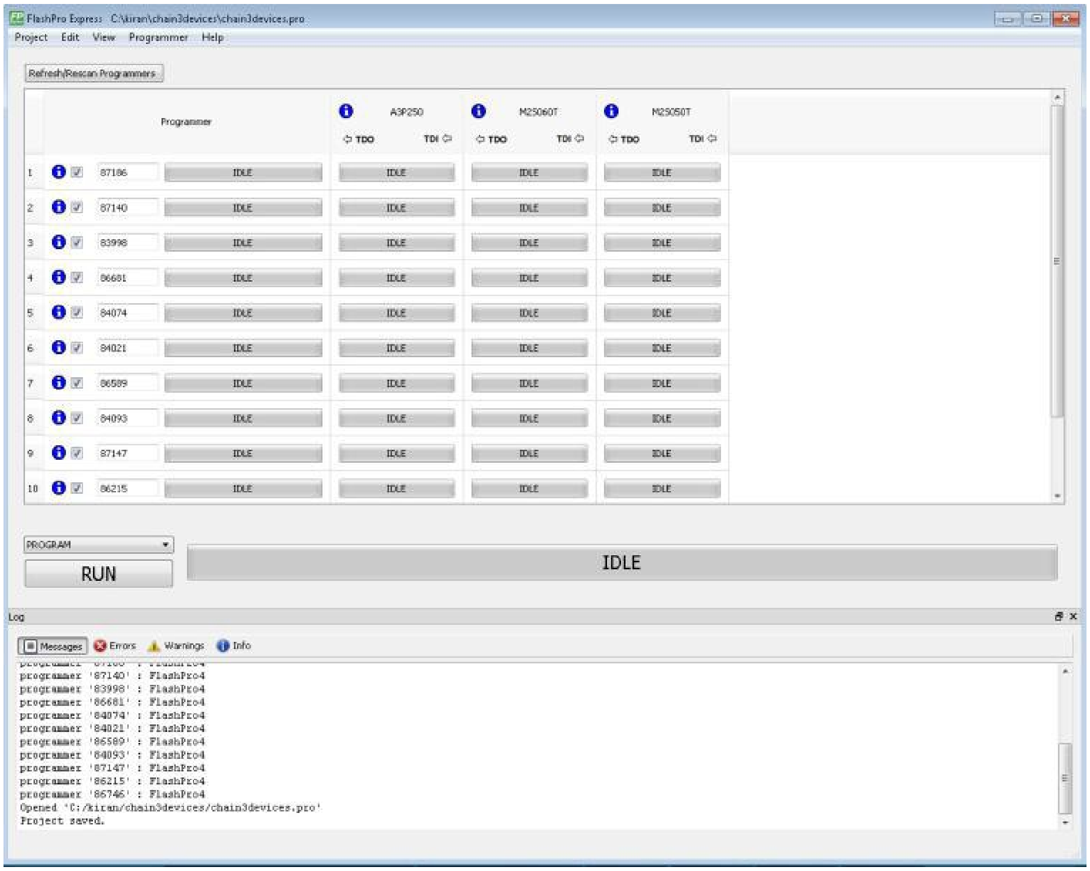
3.1. Starting FlashPro Express (Ask a Question)

FlashPro Express software is available in the `<Libero SoC installation folder>\bin` folder. Execute the `FPExpress` file to launch the FlashPro Express software.

3.2. FlashPro Express Interface (Ask a Question)

The main FlashPro Express UI consists of a list of programmers and a chain table. This view displays the programmers connected to the machine, and the devices within the JTAG chain (Figure 4-1) or a single device with SPI Target interface (Figure 4-2) specified in the job project file (PRO) file.

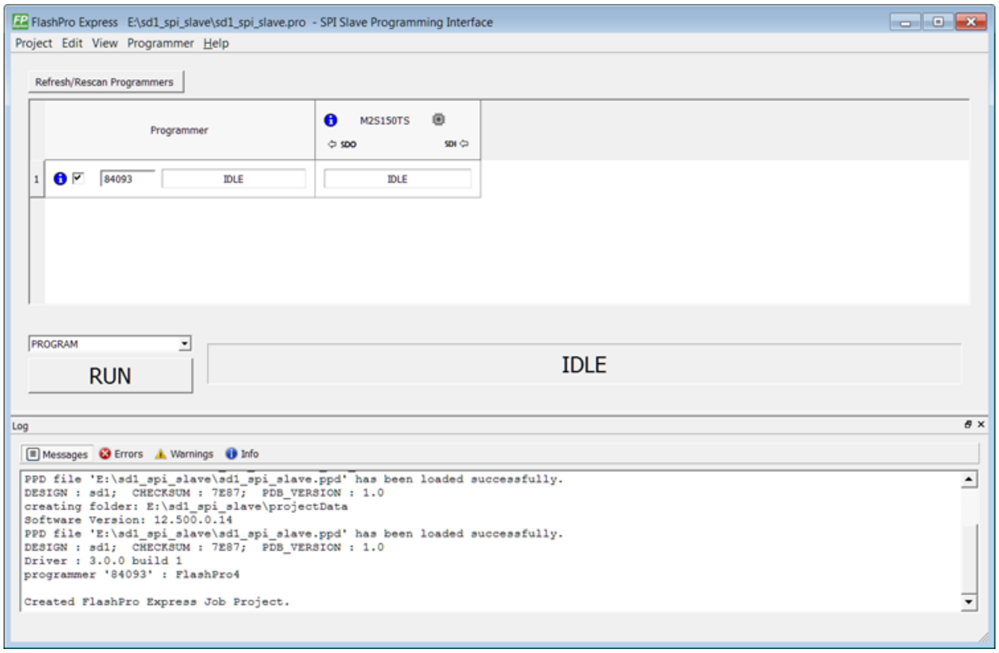
Figure 3-1. FlashPro Express Programmers and Chain Table (JTAG Example)


FlashPro Express C:\kiran\chain3devices\chain3devices.pro
Project
Edit
View
Programmer
Help
Refresh/Rescan Programmers
PROGRAM
RUN
IDLE
Log
Messages
Errors
Warnings
Info

			Programmer		A3P250 TDO TDI	M25060T TDO TDI	M25050T TDO TDI
1	☑	87186	IDLE		IDLE	IDLE	IDLE
2	☑	87140	IDLE		IDLE	IDLE	IDLE
3	☑	83998	IDLE		IDLE	IDLE	IDLE
4	☑	86681	IDLE		IDLE	IDLE	IDLE
5	☑	84074	IDLE		IDLE	IDLE	IDLE
6	☑	84021	IDLE		IDLE	IDLE	IDLE
7	☑	86589	IDLE		IDLE	IDLE	IDLE
8	☑	84093	IDLE		IDLE	IDLE	IDLE
9	☑	87147	IDLE		IDLE	IDLE	IDLE
10	☑	86215	IDLE		IDLE	IDLE	IDLE

```
programmer '87140' : FlashPro4
programmer '83998' : FlashPro4
programmer '86681' : FlashPro4
programmer '84074' : FlashPro4
programmer '84021' : FlashPro4
programmer '86589' : FlashPro4
programmer '84093' : FlashPro4
programmer '87147' : FlashPro4
programmer '86215' : FlashPro4
programmer '86746' : FlashPro4
Opened 'C:/kiran/chain3devices/chain3devices.pro'
Project saved.
```

Figure 3-2. FlashPro Express Programmers and Chain Table (SPI Target Example)


FlashPro Express E:\sd1_spi_slave\sd1_spi_slave.pro - SPI Slave Programming Interface
Project Edit View Programmer Help
Refresh/Rescan Programmers
Programmer
M2S150TS
SDO
SDI
1
84093
IDLE
IDLE
PROGRAM
RUN
IDLE
Log
Messages
Errors
Warnings
Info

```
PPD file 'E:\sd1_spi_slave\sd1_spi_slave.ppd' has been loaded successfully.
DESIGN : sd1;  CHECKSUM : 7E87;  PDB_VERSION : 1.0
creating folder: E:\sd1_spi_slave\projectData
Software Version: 12.500.0.14
PPD file 'E:\sd1_spi_slave\sd1_spi_slave.ppd' has been loaded successfully.
DESIGN : sd1;  CHECKSUM : 7E87;  PDB_VERSION : 1.0
Driver : 3.0.0 build 1
programmer '84093' : FlashPro4

Created FlashPro Express Job Project.
```

The following table describes the FlashPro actions you can perform. Devices specified as disabled in the job project (`*.pro`) file are shown disabled and their HighZ value appears in the column header.

Table 3-1. FlashPro Actions

To...	Perform This Action...
Display more information about a programmer.	Hover over the programmer Info icon.
Change a programmer name.	Click the **Name** field.
Enable or disable a programmer.	Click the check box.
Ping, Self-Test, Scan, Check Chain, or Remove it from the list.	Right-click a programmer.
View additional information about a device and programming file, if loaded.	Hover over the info icon of that device.

The following table describes the device/programmer states.

Table 3-2. Device/Programmer States

Device/Programmer State	Description
IDLE	Devices and programmers are idle and not executing any programming action.
DISABLED	Devices that are not enabled for programming.
PASSED	Last programming operation passed.
FAILED	Last programming operation failed.

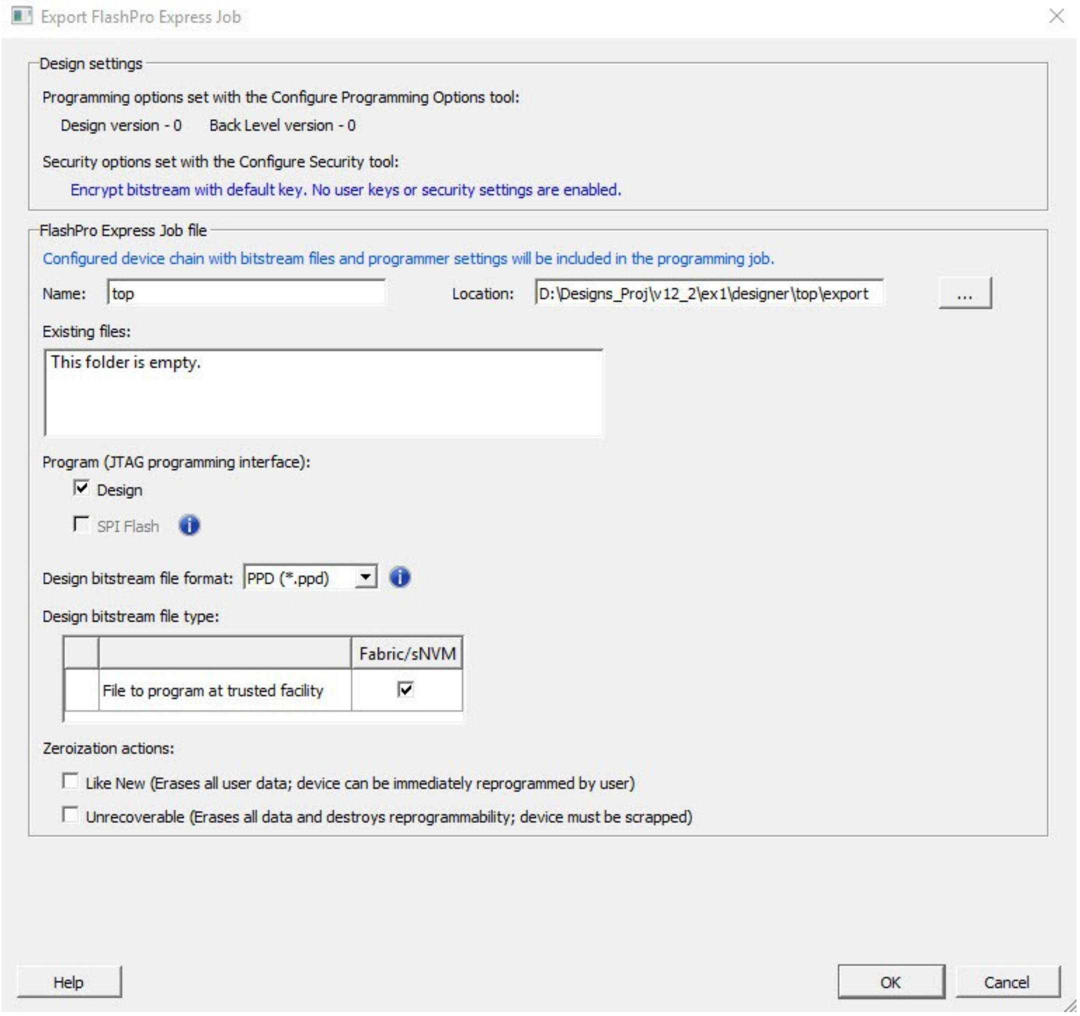
Note: SPI Target mode is supported by FlashPro5 for SmartFusion 2 and IGLOO 2 devices, and by FlashPro6 for SmartFusion 2, IGLOO 2, and PolarFire devices. JTAG is the default interface. RTG4 devices do not support SPI Target programming.

3.3. Creating a Job Project (Ask a Question)

When you are ready to hand off your design for production, create a job project.

1. In Libero, run **Export FlashPro Express Job** to create a container that will be used to transfer programming configuration information, including programming files, to the production programming tool FlashPro Express.

Figure 3-3. Export FlashPro Express Job



Export FlashPro Express Job

Design settings

Programming options set with the Configure Programming Options tool:

Design version - 0 Back Level version - 0

Security options set with the Configure Security tool:

Encrypt bitstream with default key. No user keys or security settings are enabled.

FlashPro Express Job file

Configured device chain with bitstream files and programmer settings will be included in the programming job.

Name: top Location: D:\Designs_Proj\v12_2\ex1\designer\top\export

Existing files:

This folder is empty.

Program (JTAG programming interface):

☑ Design

☐ SPI Flash

Design bitstream file format: PPD (*.ppd)

Design bitstream file type:

		Fabric/sNVM
	File to program at trusted facility	☑

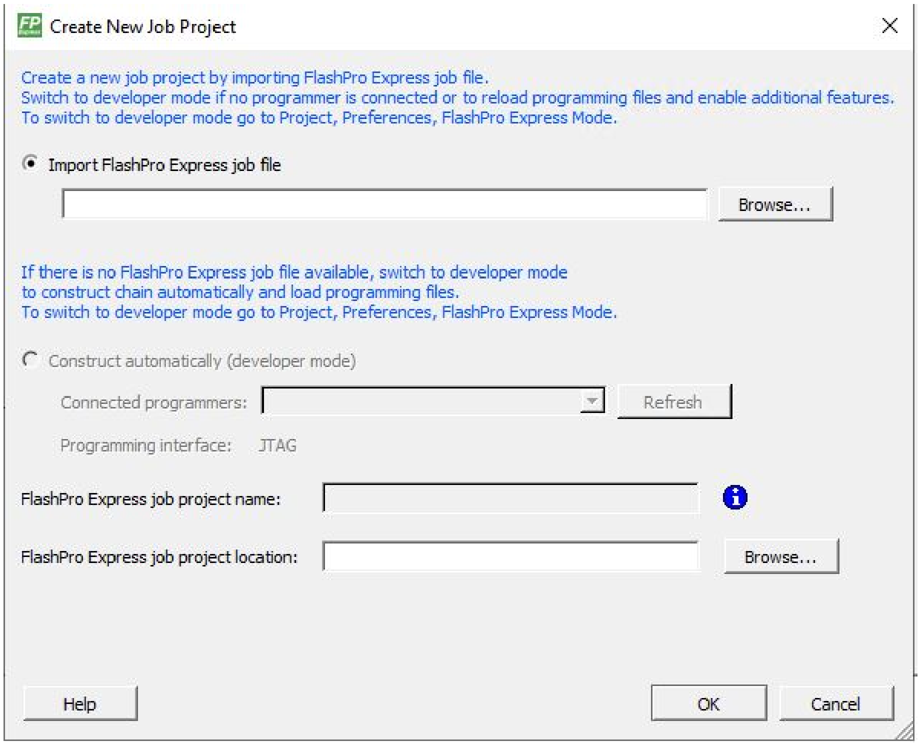
Zeroization actions:

☐ Like New (Erases all user data; device can be immediately reprogrammed by user)

☐ Unrecoverable (Erases all data and destroys reprogrammability; device must be scrapped)

2. In FlashPro Express, from the **Project** menu, choose **New Job Project**.
3. When prompted, specify the Programming Job file location that you just exported from Libero and the location to store the FlashPro Express job project. The job project name uses the programming job name and cannot be changed. Click **OK** to create and open a new job project for production programming.

Figure 3-4. New Job Project from FlashPro Express Job Dialog Box in Operator Mode



Create New Job Project

Create a new job project by importing FlashPro Express job file.
Switch to developer mode if no programmer is connected or to reload programming files and enable additional features.
To switch to developer mode go to Project, Preferences, FlashPro Express Mode.

● Import FlashPro Express job file

If there is no FlashPro Express job file available, switch to developer mode
to construct chain automatically and load programming files.
To switch to developer mode go to Project, Preferences, FlashPro Express Mode.

○ Construct automatically (developer mode)

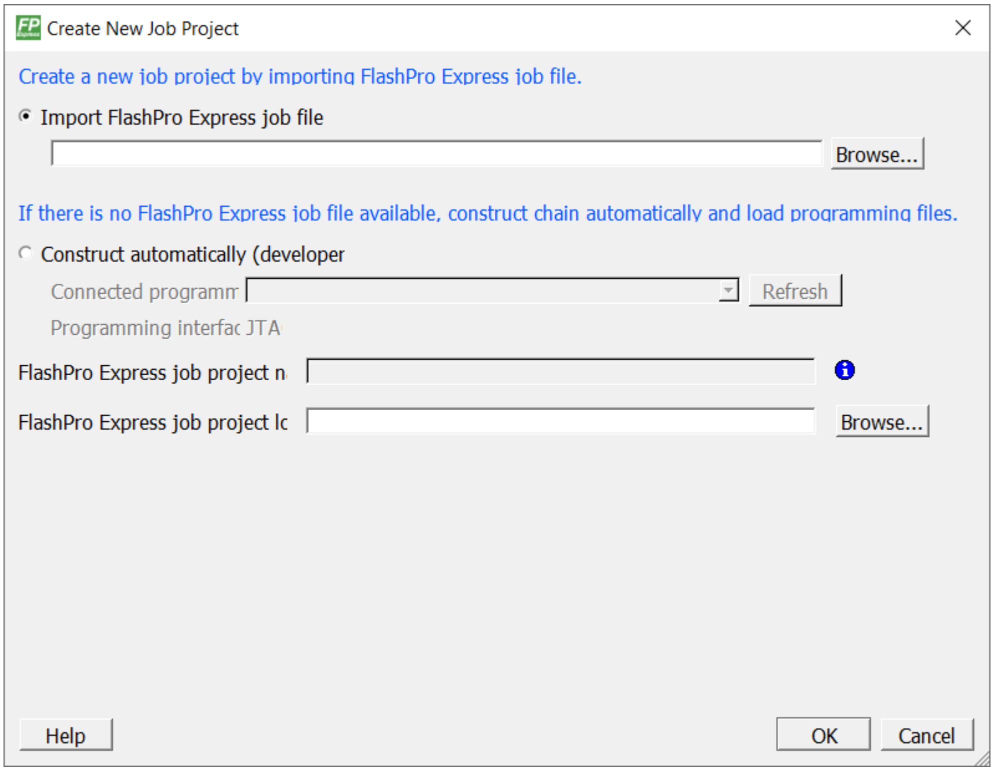
Connected programmers:

Programming interface: JTAG

FlashPro Express job project name:

FlashPro Express job project location:

Figure 3-5. New Project from FlashPro Express Job Dialog Box in Developer Mode



Create New Job Project

Create a new job project by importing FlashPro Express job file.

● Import FlashPro Express job file

If there is no FlashPro Express job file available, construct chain automatically and load programming files.

○ Construct automatically (developer

Connected programmr

Programming interfac JTA

FlashPro Express job project n

FlashPro Express job project lc

The user can create a new FlashPro Express job project without connecting hardware. After the job project is created, it can be modified by adding, removing devices, loading programming files, and so on.

> **Important:** You can use any `.job` file to create a new job when there are no programmers connected.

The user can create, a new job project using **Construct automatically** option. This option can be used to construct chain by scanning the physical chain connected to the selected programmer. This feature is available only in JTAG mode. If multiple programmers are connected to the machine, select the desired programmer by clicking on the drop down menu for connected programmers field.

3.4. Opening a Job Project (Ask a Question)

To start with FlashPro Express, load a job project (`*.pro`) file.

To open a job project:

1. From the **Project** menu, choose **Open Job Project**. The Open Project dialog box appears.
2. Find your project file or type in your project file name in the **File name** field.
3. Click **Open**.

In the Operator mode, a job project opens if the physical chain containing at least one Microchip device is connected. The Developer mode does not have any restrictions.

To update device(s), you can:

1. Use the **Device Configuration** menu to update the device.
2. Open the **Programming Connectivity and Interface** dialog to update the chain/devices.
3. Connect a programmer, and click on **Refresh/Rescan Programmers** to enable programmers/devices and run programming.

3.5. Saving a Job Project (Ask a Question)

To save a job project, either:

- Click the **Save** button on the toolbar, or
- From the **Project** menu choose **Save Job Project**.

3.6. Programming Tutorials (Ask a Question)

The following programming tutorials describe real-world examples of using FlashPro.

3.6.1. Parallel Programming with FlashPro5/4/3/3X (Ask a Question)

Parallel programming allows you to program multiple Microchip devices in parallel with multiple programmers. In parallel programming, all targeted devices are programmed with the same programming file (STAPL). The targeted device or chain configuration that is connected to each programmer must be identical.

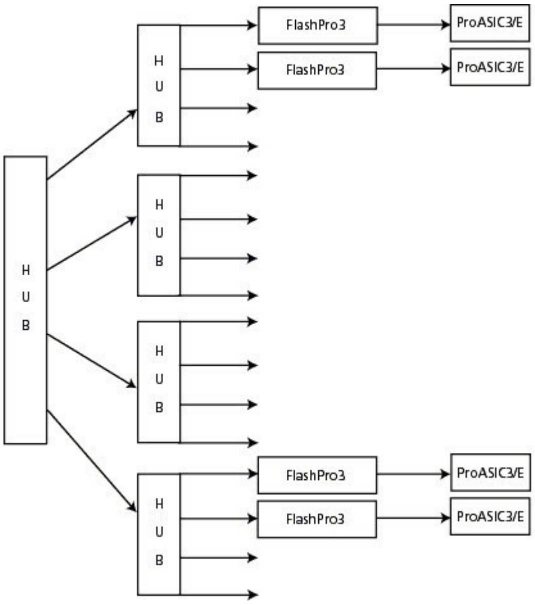
The FlashPro Express software together with the FlashPro5/4/3/3X programmers supports parallel programming via a USB port. You can connect up to sixteen FlashPro5/4/3/3X's to a PC via a USB v1.1 or a USB v2.0 port.

FlashPro5/4/3/3X requires a self-powered hub.

Connecting FlashPro5/4/3/3X (a USB v2.0 enabled programmer) to USB v1.1 port increases device programming time due to a slow data transfer rate on the USB v1.1 port in comparison to a USB v2.0 port.

The following figure shows how to connect a FlashPro5/4/3/3X programmer for parallel programming.

Figure 3-6. Connecting a FlashPro5/4/3/3X Programmer



An independent thread processes the STAPL file during parallel programming. In a Microchip test environment, parallel programming is approximately five times faster than programming 16 devices sequentially.

Note: Microchip has tested Belkin PCI-USB cards and hubs, and found that parallel programming works best when using the vendor's latest driver installed along with matching hubs.

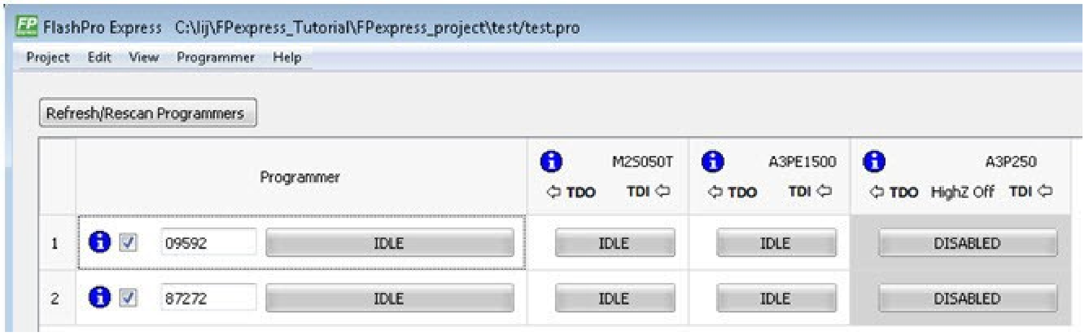
3.6.2. Chain Programming Tutorial (Ask a Question)

This tutorial describes how to use FlashPro Express to program a multi-device, multi-programmer chain. This tutorial uses the production programming flow that exports a programming job from Libero SoC, which includes chain configuration, programmer settings, and bit stream files for programming, and creates a job project from a programming job.

The following figure shows the chain used in this tutorial. M2S050T is device 1 and A3P250 is device 3.

- Device 1 is the first device to be programmed in the chain.
- Device 2 is the last device to be programmed in the chain.
- Device 3 is disabled and will not be programmed.

Figure 3-7. Chain Programming Devices

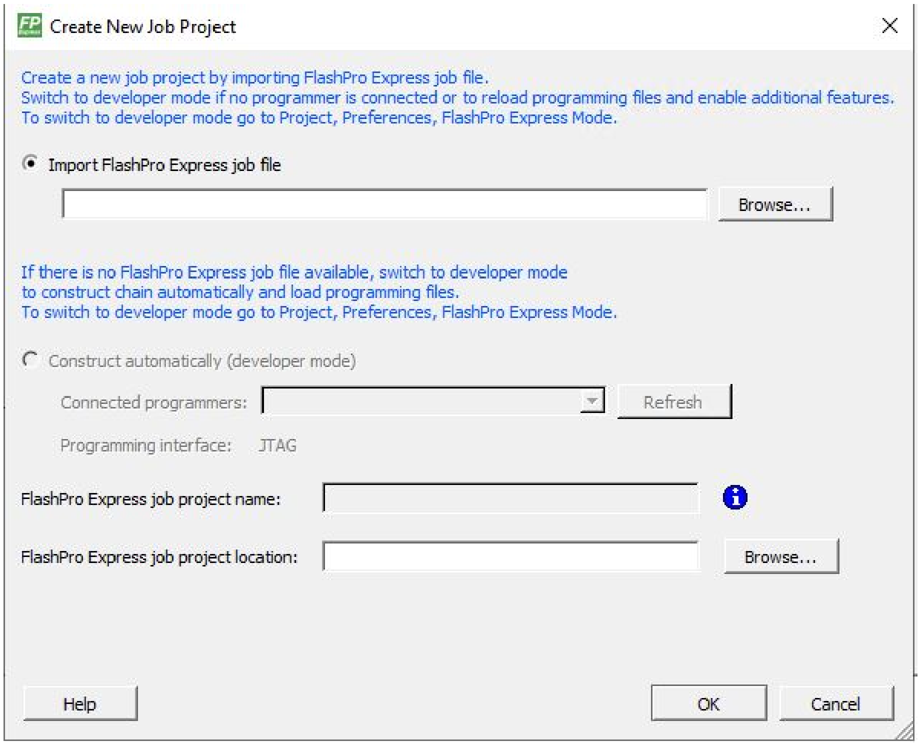

FlashPro Express C:\lij\FPexpress_Tutorial\FPexpress_project\test/test.pro
Project
Edit
View
Programmer
Help
Refresh/Rescan Programmers

	Programmer		M2S050T ⇦ TDO TDI ⇦	A3PE1500 ⇦ TDO TDI ⇦	A3P250 ⇦ TDO HighZ Off TDI ⇦
1	☑ 09592	IDLE	IDLE	IDLE	DISABLED
2	☑ 87272	IDLE	IDLE	IDLE	DISABLED

To program a chain:

1. From the **Project** menu, choose **New Job Project from FlashPro Express Job**.
2. Click **Browse** to load a Programming Job File, and specify your **FlashPro Express job project location**. Click **OK** to continue, as shown in the following figure.

Figure 3-8. New Job Project from FlashPro Express Job



Create New Job Project

Create a new job project by importing FlashPro Express job file.
Switch to developer mode if no programmer is connected or to reload programming files and enable additional features.
To switch to developer mode go to Project, Preferences, FlashPro Express Mode.

● Import FlashPro Express job file

If there is no FlashPro Express job file available, switch to developer mode
to construct chain automatically and load programming files.
To switch to developer mode go to Project, Preferences, FlashPro Express Mode.

○ Construct automatically (developer mode)

Connected programmers:

Programming interface: JTAG

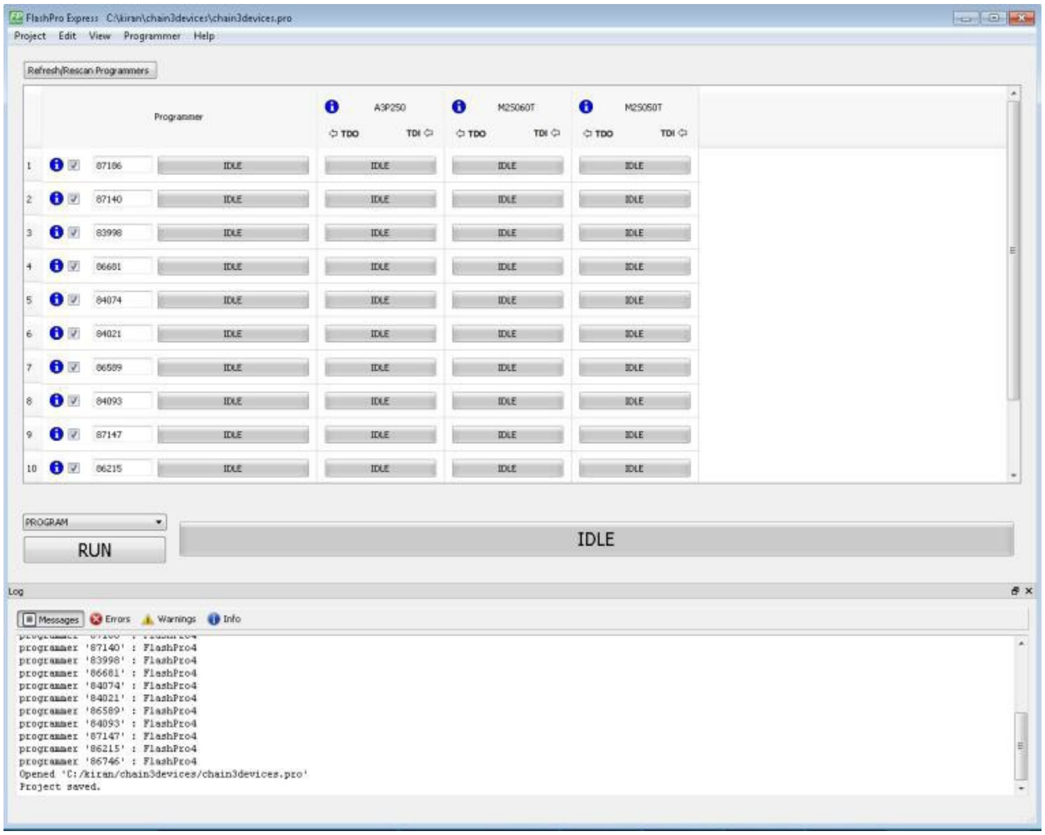
FlashPro Express job project name:

FlashPro Express job project location:

FlashPro Express displays your Job Project and programmers, as shown in the following figure. The Device/Programmer states are:

- IDLE: Devices/programmers are idle and not executing any programming action.
- DISABLED: Devices that are not enabled for programming.
- PASSED: Last programming operation passed.
- FAILED: Last programming operation failed.

Figure 3-9. FlashPro Express with Loaded Job Project (JTAG Example)

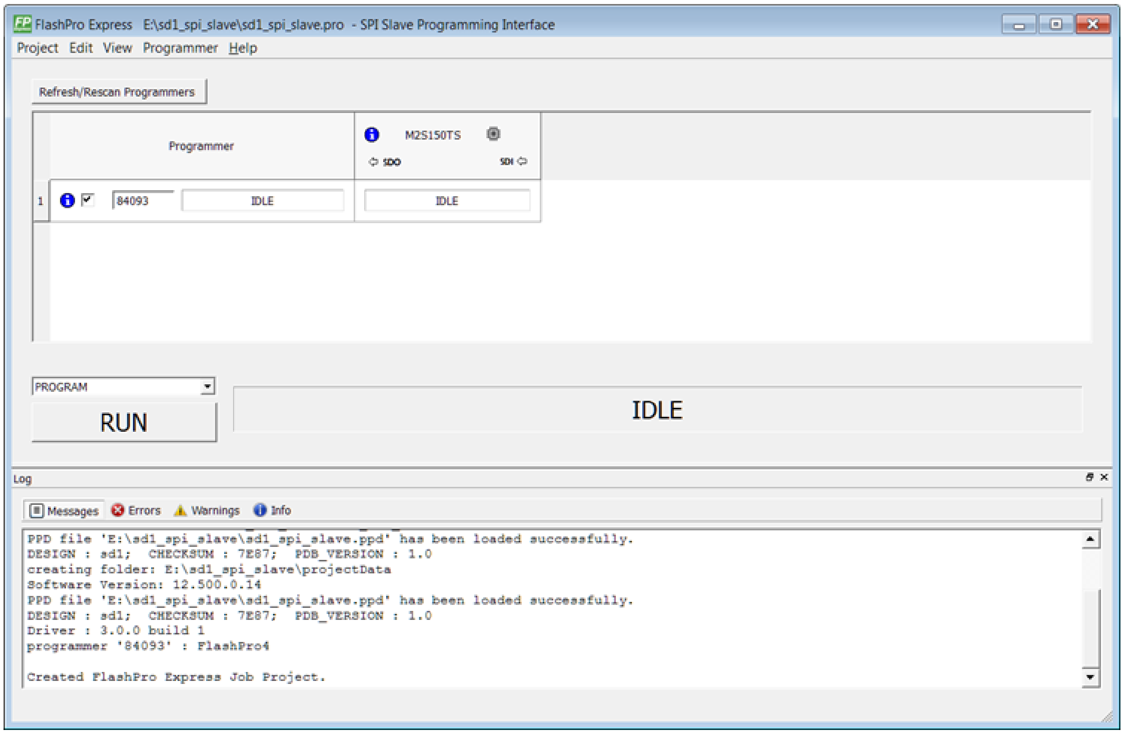


FlashPro Express C:\kiran\chain3devices\chain3devices.pro

	Programmer		A3P250 TDO TDI	M2S060T TDO TDI	M2S050T TDO TDI
1	87186	IDLE	IDLE	IDLE	IDLE
2	87140	IDLE	IDLE	IDLE	IDLE
3	83998	IDLE	IDLE	IDLE	IDLE
4	86681	IDLE	IDLE	IDLE	IDLE
5	84074	IDLE	IDLE	IDLE	IDLE
6	84021	IDLE	IDLE	IDLE	IDLE
7	86589	IDLE	IDLE	IDLE	IDLE
8	84093	IDLE	IDLE	IDLE	IDLE
9	87147	IDLE	IDLE	IDLE	IDLE
10	86215	IDLE	IDLE	IDLE	IDLE

```
programmer '87140' : FlashPro4
programmer '83998' : FlashPro4
programmer '86681' : FlashPro4
programmer '84074' : FlashPro4
programmer '84021' : FlashPro4
programmer '86589' : FlashPro4
programmer '84093' : FlashPro4
programmer '87147' : FlashPro4
programmer '86215' : FlashPro4
programmer '86746' : FlashPro4
Opened 'C:/kiran/chain3devices/chain3devices.pro'
Project saved.
```

Figure 3-10. FlashPro Express with Loaded Job Project (SPI Target Example – SmartFusion 2/IGLOO 2 Only)



FlashPro Express E:\sd1_spi_slave\sd1_spi_slave.pro - SPI Slave Programming Interface

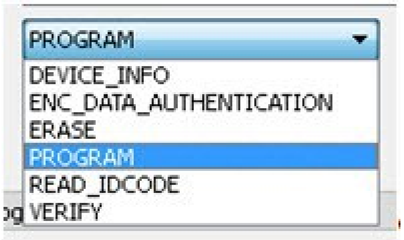
	Programmer		M2S150TS SDO SDI
1	84093	IDLE	IDLE

```
PPD file 'E:\sd1_spi_slave\sd1_spi_slave.ppd' has been loaded successfully.
DESIGN : sd1;  CHECKSUM : 7E87;  PDB_VERSION : 1.0
creating folder: E:\sd1_spi_slave\projectData
Software Version: 12.500.0.14
PPD file 'E:\sd1_spi_slave\sd1_spi_slave.ppd' has been loaded successfully.
DESIGN : sd1;  CHECKSUM : 7E87;  PDB_VERSION : 1.0
Driver : 3.0.0 build 1
programmer '84093' : FlashPro4

Created FlashPro Express Job Project.
```

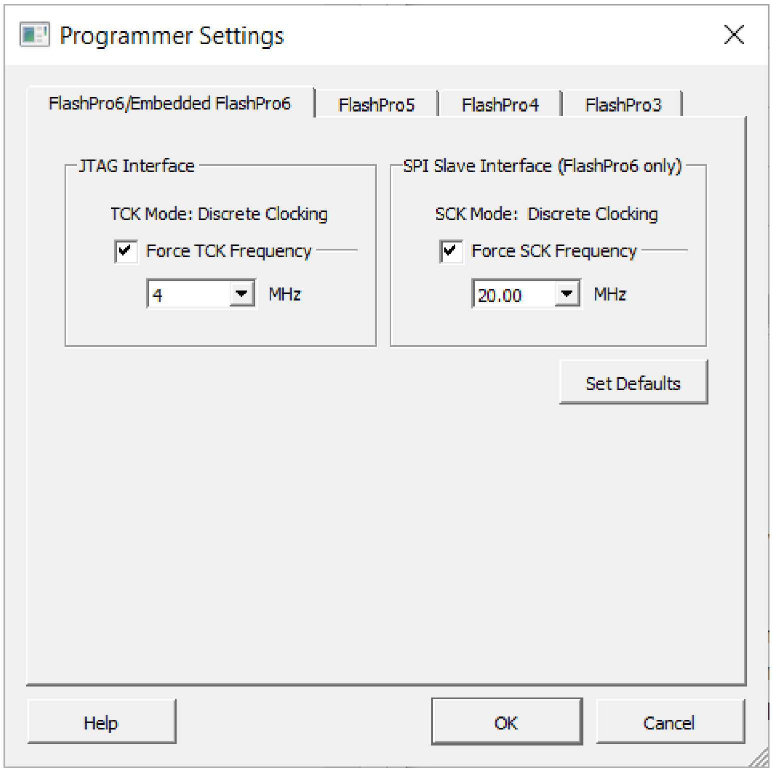
3. If your programmer is not listed, click the **Refresh/Rescan** button. To view device info, hover your mouse over the **Info** icon. If a device is Disabled for programming, the HighZ status appears in the GUI.
4. Set the Programming Action in the drop-down menu to **PROGRAM**, as shown in the following figure.

Figure 3-11. Programming Action Set to PROGRAM



5. Click **RUN**. Detailed individual programmer and device status information appears in the Programmer List. Your programmer status (PASSED or FAILED) appears in the Programmer Status Bar, as shown in the following figure.
   - Hover over the Programmer Status Bar to display information on the programmers.
   - Hover over the FAILED status to list all programmers that failed programming.
   - Hover over the PASSED status to list all the programmers that programmed successfully.

Figure 3-12. Chain Programming Complete



JTAG Interface

TCK Mode: Discrete Clocking

☑ Force TCK Frequency

4 MHz

SPI Slave Interface (FlashPro6 only)

SCK Mode: Discrete Clocking

☑ Force SCK Frequency

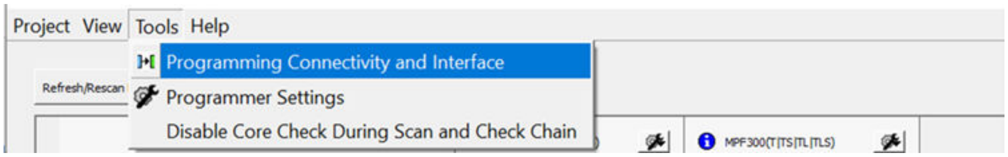
20.00 MHz

6. View the **Log** for Messages, Errors, Warnings and Info generated during programming.

4. Programming Connectivity and Interface (Ask a Question)

The **Programming Connectivity and Interface** option can be selected from the **Tools** menu.

Figure 4-1. Programming Connectivity and Interface Option in Tools Menu

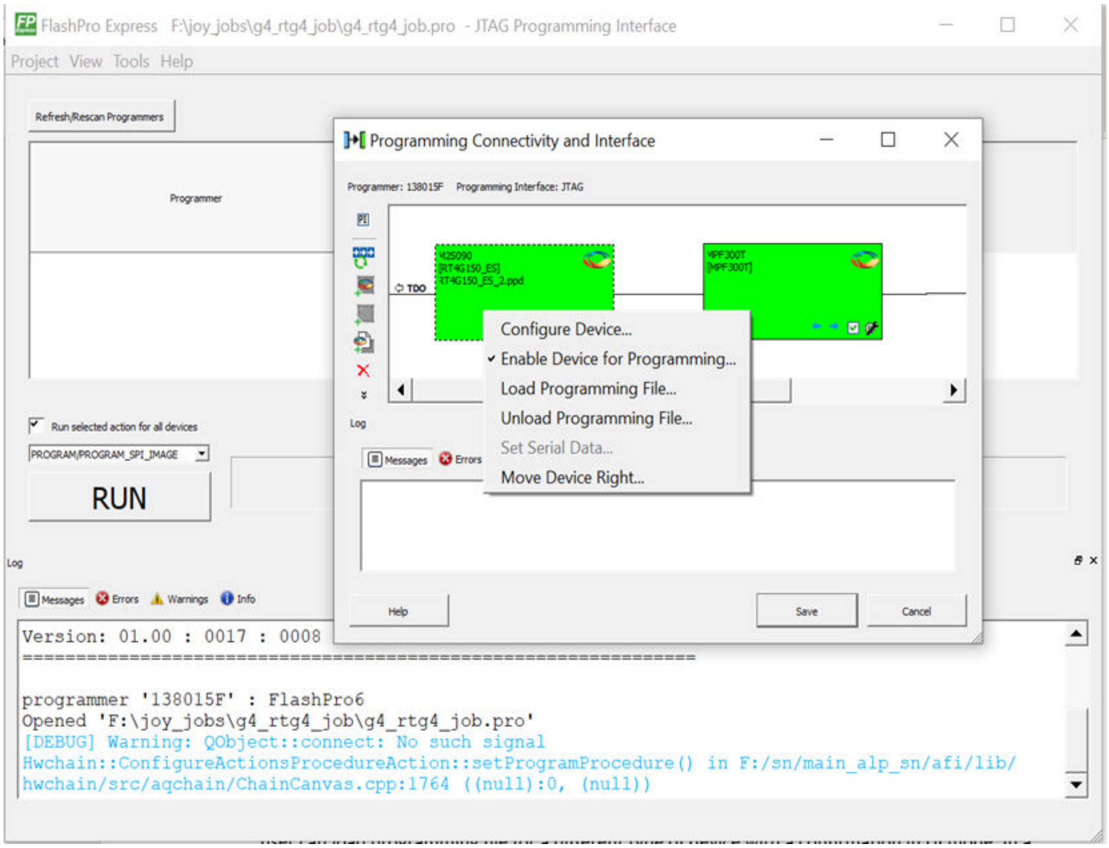

Project
View
Tools
Help
Refresh/Rescan
Programming Connectivity and Interface
Programmer Settings
Disable Core Check During Scan and Check Chain
MPF300(T|TS|TL|TLS)

By default, the tool selects the first enabled programmer and opens the Programming Connectivity and Interface dialog box with an existing JTAG chain or SPI Target device. The user can disable all programmers, except the one which is used by the FlashPro Express tool.

The selected programmer and current programming interface are shown in the dialog box for the user reference. The main window's programmer table is cleared as soon as the chain dialog box is opened and will be populated with saved changes or restored to previous state if changes are canceled when the dialog box is closed.

In developer mode, if there is no programmer connected, user can manually construct chain.

Figure 4-2. Programming Connectivity and Interface Dialog Box with Configure Chain Options


FlashPro Express F:\joy_jobs\g4_rtg4_job\g4_rtg4_job.pro - JTAG Programming Interface
Project
View
Tools
Help
Refresh/Rescan Programmers
Programmer
Run selected action for all devices
PROGRAM/PROGRAM_SPI_IMAGE
RUN
Log
Messages
Errors
Warnings
Info
Programming Connectivity and Interface
TDO
M2S090
[RT4G150_ES]
RT4G150_ES_2.ppd
MPF300T
[MPF300T]
Configure Device...
Enable Device for Programming...
Load Programming File...
Unload Programming File...
Set Serial Data...
Move Device Right...
Log
Messages
Errors
Help
Save
Cancel

Programmer: 138015F Programming Interface: JTAG

```
Version: 01.00 : 0017 : 0008
===============================================================

programmer '138015F' : FlashPro6
Opened 'F:\joy_jobs\g4_rtg4_job\g4_rtg4_job.pro'
[DEBUG] Warning: QObject::connect: No such signal
Hwchain::ConfigureActionsProcedureAction::setProgramProcedure() in F:/sn/main_alp_sn/afi/lib/
hwchain/src/aqchain/ChainCanvas.cpp:1764 ((null):0, (null))
```

For JTAG Interface, the same configure chain options are available as in Libero SoC tool.

For SPI Target interface, the following options are available:

- **Set Programming Interface**: Selects JTAG or SPI Target mode.

- **Add Microsemi Device**: Adds a Microsemi device to the chain.
- **Delete device**: Deletes selected devices in the grid.
- **Zoom In**: Zoom into the grid.
- **Zoom Out**: Zoom out of the grid.

The devices used in FlashPro Express tool have the following context menu options:

- **Configure Device**: Configure device is same as the one in Libero SoC tool. Configure device option allows to configure device by loading programming file (but not a SPI Flash file) or to set device family and die.
- **Enable/Disable Device for Programming**: Enables or disables device for programming.
- **Load Programming File**: Loads the programming file.
- **Unload Programming File**: This option is enabled if programming file loaded.
- **Load SPI Flash File**: Loads a SPI Flash file. This option is always enabled. It is available for PolarFire and PolarFire SoC devices.
- **Unload SPI Flash File**: This option is enabled if SPI Flash file is loaded. It is available for PolarFire and PolarFire SoC devices.
- **Set Serial Data**: This option is disabled; serialization is not supported.
- **Move Device**: to left/right

There is no option to configure/select programming action or SPI Flash action. This option is only available in the main table because it depends on the developer mode option to run one programming action for all devices or run an action selected for each device.

From Libero SoC v12.6 onwards, you can change a device type by loading the programming file for a different device type. This is consistent with Libero flow.

In the chain dialog box and in the main table, the user can load the programming file for a different device type with a GUI confirmation. The chain tool also allows the user to reconfigure the device by selecting a different family and a die. In this case, no confirmation is required.

In the batch mode, a programming file is loaded with a warning.

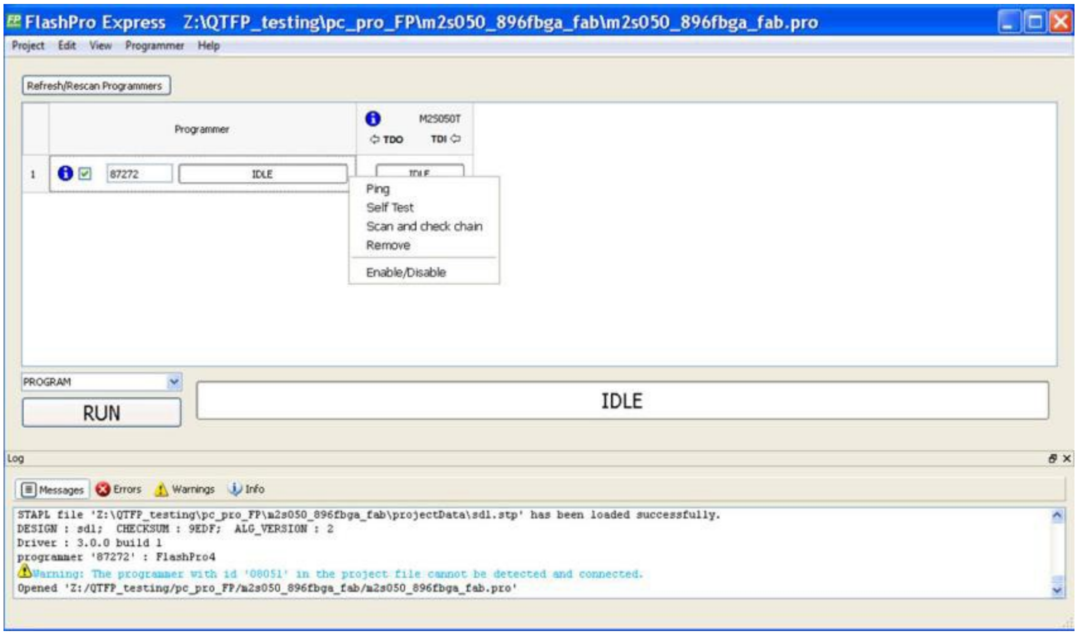
5. Programmer Settings and Operations (Ask a Question)

This section describes the FlashPro Express settings and operation.

5.1. Introduction (Ask a Question)

The FlashPro Express software allows you to connect multiple programmers to your computer. With each programmer you select, you can perform a ping, conduct a self-test, scan and check the chain, and remove, enable, or disable the JTAG chain, as shown in the following figure.

Figure 5-1. FlashPro Express Right-Click Menu



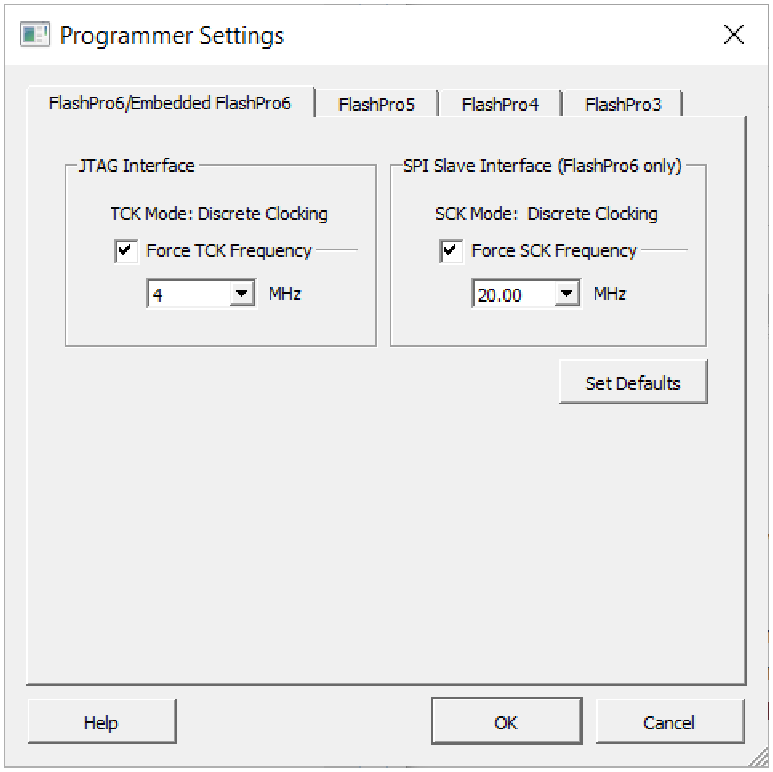
```
STAPL file 'Z:\QTFP_testing\pc_pro_FP\m2s050_896fbga_fab\projectData\sdl.stp' has been loaded successfully.
DESIGN : sdl;  CHECKSUM : 9EDF;  ALG_VERSION : 2
Driver : 3.0.0 build 1
programmer '87272' : FlashPro4
Warning: The programmer with id '08051' in the project file cannot be detected and connected.
Opened 'Z:/QTFP_testing/pc_pro_FP/m2s050_896fbga_fab/m2s050_896fbga_fab.pro'
```

5.2. Programmer Settings (Ask a Question)

To view the Programmer Settings dialog box, in the Libero SoC Design Flow window, expand **Configure Hardware**, double click **Configure Programmer** or right click **Configure Programmer** and choose **Programmer Settings**.

For the JTAG interface, you can set specific voltage and force TCK frequency values for your programmer in this dialog box. For the SPI Target interface, you can force SCK frequency values for your programmer. SPI Target mode is supported by FlashPro5 for SmartFusion 2 and IGLOO 2 devices, and by FlashPro6 for SmartFusion 2, IGLOO 2, and PolarFire devices. SPI Target mode is not supported for RTG4 devices. JTAG is the default interface.

Figure 5-2. Programmer Settings



Programmer Settings

Tabs: FlashPro6/Embedded FlashPro6 | FlashPro5 | FlashPro4 | FlashPro3

JTAG Interface

TCK Mode: Discrete Clocking

☑ Force TCK Frequency

4 MHz

SPI Slave Interface (FlashPro6 only)

SCK Mode: Discrete Clocking

☑ Force SCK Frequency

20.00 MHz

The Programmer Settings dialog box includes options for FlashPro6/5/4/3/3X. Limitations of the TCK frequency for the selected programmer are:

- FlashPro6: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20 MHz
- FlashPro5: 1, 2, 3, 4, 5, 6, 10, 15, 30 MHz
- FlashPro4: 1, 2, 3, 4, 5, 6 MHz
- FlashPro3/3X: 1, 2, 3, 4, 6 MHz TCK frequency limits by target device (refer to the target device data sheet)

During execution, the frequency set by the FREQUENCY statement in the PDB/STAPL file overrides the TCK frequency settings you select in the Programmer Settings dialog box, unless you also select the **Force TCK Frequency** check box.

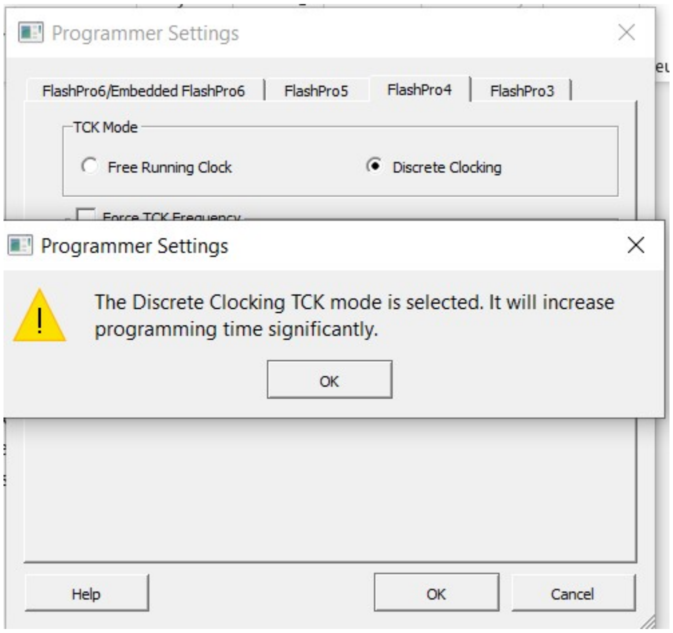
Limitation of the SCK frequency for the selected programmer are: 1.00, 2.00, 2.50, 3.33, 4.00, 5.00, 6.67, 8.00, 10.00, 13.33, and 20.00 MHz

5.2.1. FlashPro5/4/3/3X Programmer Settings (Ask a Question)

For FlashPro5/4/3/3X, if you choose the **Force TCK Frequency**, select the appropriate MHz frequency. For FlashPro4/3X settings, you can switch the TCK mode between a Free Running Clock and a Discrete Clocking. Use Discrete Clocking when there is a JTAG non-compliant device in a chain with Microchip devices. After selecting, click **OK**.

Attention: When the **Discrete Clocking** mode is selected, a warning pop-up appears, which states that using the Discrete Clocking TCK mode will increase the programming time significantly, as shown in the following figure.

Figure 5-3. Programmer Settings Warning Pop-Up



Programmer Settings

The Discrete Clocking TCK mode is selected. It will increase programming time significantly.

After the pop-up is closed by the user, a warning icon with a message tooltip appears next to the **Discrete Clocking** radio button. The icon and the tooltip appear when the programmer settings dialog is re-opened.

5.2.1.1. Default Settings (Ask a Question)

- The **Force TCK Frequency** option is unchecked to instruct the FlashPro5/4/3/3X to use the TCK frequency specified by the Frequency statement in the PDB/STAPL file(s).
- The FlashPro5/4/3/3X default TCK mode setting is **Free running clock**.

5.2.2. TCK Setting for Force TCK Frequency (Ask a Question)

If **Force TCK Frequency** is checked in the **Programmer Setting**, the selected TCK value is set for the programmer and the Frequency statement in the PDB/STAPL file is ignored.

5.2.3. Default TCK Frequency (Ask a Question)

If the IPD/STAPL file or Chain does not exist, the default TCK frequency is set to 4 MHz.

If more than one Microchip flash device is targeted in the chain, the FlashPro Express software passes through all the files and searches for the "freq" keyword and the "MAX_FREQ" **Note** field. The FlashPro Express software uses the lesser value of all the TCK frequency settings and the "MAX_FREQ" **Note** field values.

5.3. Ping Programmers (Ask a Question)

Right-click a programmer and choose Ping.

Note: To ping new programmers quickly, click the **Refresh/Rescan for Programmers** button.

5.4. Performing a Self-Test (Ask a Question)

Before performing a self-test, connect the programmer to the self-test board that came with your programmer. Then right-click the programmer you want to self-test and choose **Self Test**.

Note: Self-test is not supported with FlashPro5/4 programmers. These programmers are tested rigorously at the factory during production.

5.5. Scanning and Checking a Chain (Ask a Question)

The scan chain operation scans and analyzes the JTAG chain connected to programmers you selected and checks that a scanned chain matches the chain configured in FlashPro Express.

To scan a chain, right-click the programmer you want to scan and choose **Scan and check chain**.

5.6. Enabling and Disabling Programmers (Ask a Question)

After loading a job project, you can enable, disable, or remove a programmer, as well as ping, self-test, run scan, and check chain on any of the connected programmers. These actions are available in the right-click menu for each programmer in the programmer column.

Check the check box next to a programmer in the **Programmer** column to enable the programmer or uncheck the check box to disable the programmer.

5.7. Renaming a Programmer (Ask a Question)

Enter a new programmer name in the Programmer window to rename the programmer. By default, the programmer name is the same as the programmer ID.

5.8. Removing a Programmer (Ask a Question)

To remove a programmer, right-click the programmer and choose **Remove**.

5.9. Selecting and Running an Action (Ask a Question)

FlashPro Express supports the following programming actions:

- DEVICE_INFO - This action queries the device and gets the information about type and family of device, information about design programmed on the device (if any), the number of times the device has been programmed, information about security settings and prints out digests of various components/segments.

  Attention: If the erase operation is interrupted for any reason, the cycle count value is lost. However, the device can still be programmed.

- ENC_DATA_AUTHENTICATION - This is visible when each device in the chain contains encrypted bit stream files. Selecting this action checks each bit stream file for authentication.
- ERASE
- PROGRAM - This action programs the bitstream file on the device and the resulting programming log contains information about component digests that are generated while programming the bitstream file as well as digests that are read out from device after the bitstream has been programmed. The log also contains the entire bitstream digest that has been precalculated during bitstream file generation.

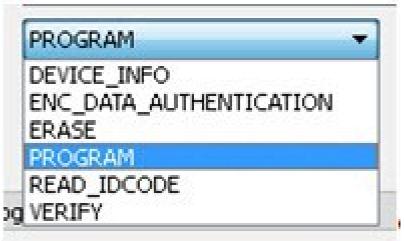
  Attention: The component digests printed during programming differ in value and count from the digests that are read out from the device.

- READ_IDCODE

- VERIFY

To select a programming action, select an action from the **Programming Action** drop-down menu in FlashPro Express, as shown in the following figure.

Figure 5-4. FlashPro Express Programming Actions



To run the selected programming action, click the **RUN** button below the **Programming Action** drop-down menu.

⚠ **CAUTION** Do not interrupt programming action while its running, else it might damage the part.

6. Chain Programming (Ask a Question)

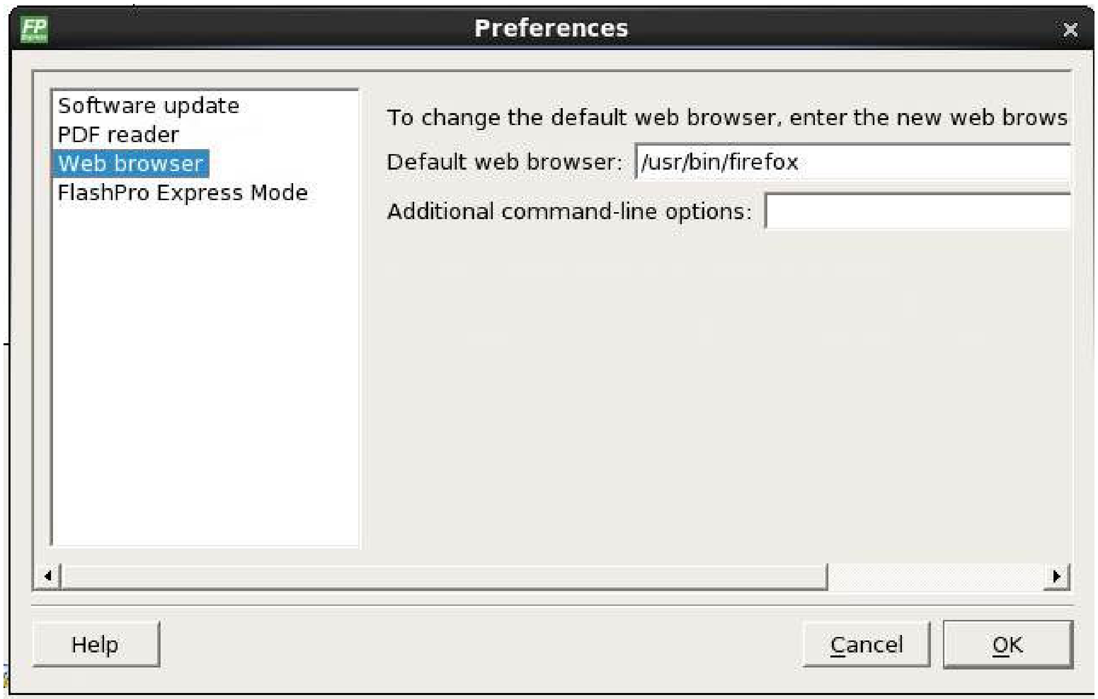
This section describes how to perform chain programming.

6.1. Chain Order (Ask a Question)

Chain Programming allows you to program several devices at one time. The order of devices in the chain imported from Job Project must match the physical chain to be programmed.

The TDO for the first device connects to the programmer, and the last device's TDI connects to the programmer. The devices in the chain go in order from a device's TDI into the next device's TDO, as shown in the following figure.

Figure 6-1. Chain Order



To change the default web browser, enter the new web brows

Default web browser: /usr/bin/firefox

Additional command-line options:

6.2. Multiple Device Chain Programming (Ask a Question)

The FlashPro Express software allows direct chain programming without generating a chain STAPL file. Each device is programmed in sequential order, starting from device 1 to device *N* (see the following example).

```
TDI > Device N > Device N-1 >… > Device 2 > Device 1 > TDO
```

6.2.1. Device Programming Compatibility (Ask a Question)

PolarFire, SmartFusion 2, IGLOO 2, and RTG4 devices can be programmed in the same chain.

6.2.2. Programmer Support (Ask a Question)

FlashPro5/4/3/3X supports PolarFire, SmartFusion 2, IGLOO 2, and RTG4 devices. The Vpump on FlashPro5/4/3/3X is designed to support the programming of only one device. Make sure that Vpump, Vcc, and Vjtag are provided on board for chain programming. Connect the Vpump to the header as the Flashpro Express software will attempt to check for all external supplies, including Vpump, to ensure successful programming. There is no limitation to the chain length; however, ensure that the JTAG signal integrity and the timing are preserved.

7. FlashPro Express Modes (Ask a Question)

Starting with Libero SoC v12.5, FlashPro Express supports two modes:

- Operator mode
- Developer mode

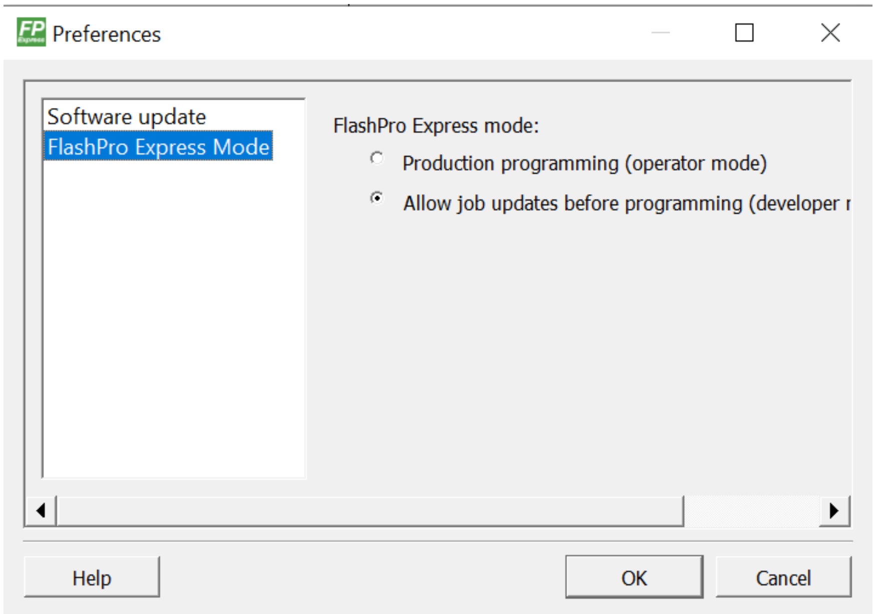
Operator mode is a current flow that provides production programming. It is the default mode and allows you to run selected actions for individual and all devices.

Developer mode allows you to:

- Update jobs before running programming.
- Enable or disable chain devices.
- Load design and SPI Flash Programming files and select different programming actions for each chain device and SPI-Flash.
- Run selected actions for individual and all devices.

You select Operator or Developer mode using the Preference dialog box (**Project** > **Preferences**). The preference is saved per user per machine on Windows and per user on Linux. The mode preference remains the same until you change it.

Figure 7-1. FlashPro Express Mode Preference Dialog (Windows)



Preferences

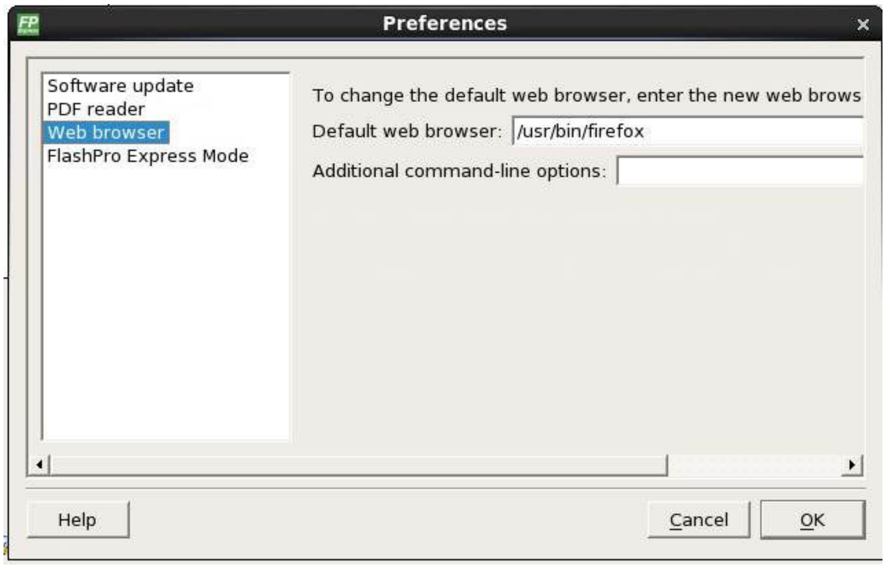
Software update
FlashPro Express Mode

FlashPro Express mode:

○ Production programming (operator mode)

● Allow job updates before programming (developer r

Figure 7-2. FlashPro Express Mode Preference Dialog (Linux)



To change the default web browser, enter the new web brows

Default web browser: /usr/bin/firefox

Additional command-line options:

The FlashPro Express mode can be switched before opening a job. If a job is opened, you are prompted to confirm closing of the job to save the mode preference after clicking the **OK** button.

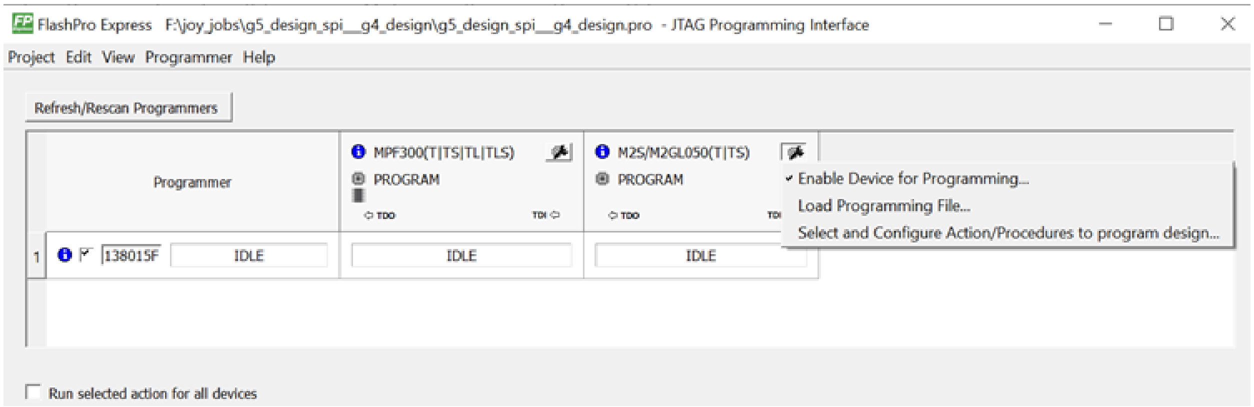
After a job is opened in Developer mode, each device displays:

- An info (ℹ) icon with device specific data.
- Design icon (▮). The per-device selected action appears next to the icon if the **Run selected action for all devices** option is unselected.
- SPI Flash icon (▮) if SPI Flash programming is available for the device. The per-device selected SPI Flash action appears next to the icon if the **Run selected action for all devices** option is unselected.
- Configure button (🔧) providing a menu of configuration options based on the device. The configure button is enabled when no programming action is running.

The following list describes Developer mode features:

- Device configuration options

Figure 7-3. Device Options to Program Design

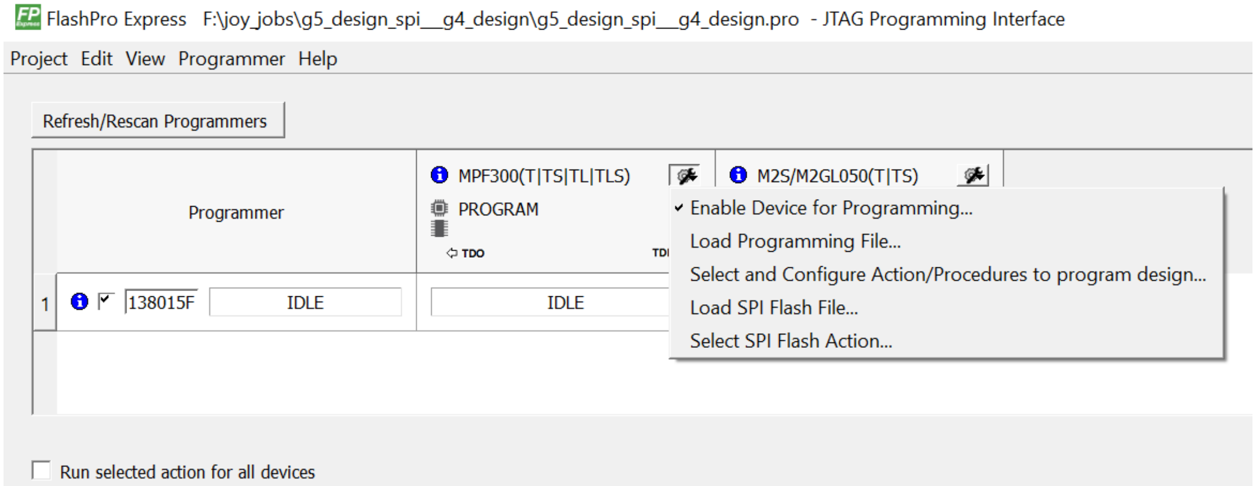

Project
Edit
View
Programmer
Help
Refresh/Rescan Programmers
Programmer
TDO
TDI
Enable Device for Programming...
Load Programming File...
Select and Configure Action/Procedures to program design...

FlashPro Express F:\joy_jobs\g5_design_spi__g4_design\g5_design_spi__g4_design.pro - JTAG Programming Interface

Programmer	MPF300(T|TS|TL|TLS) PROGRAM	M2S/M2GL050(T|TS) PROGRAM
1 ☑ 138015F IDLE	IDLE	IDLE

□ Run selected action for all devices

Figure 7-4. Device Options to Program Design and SPI Flash

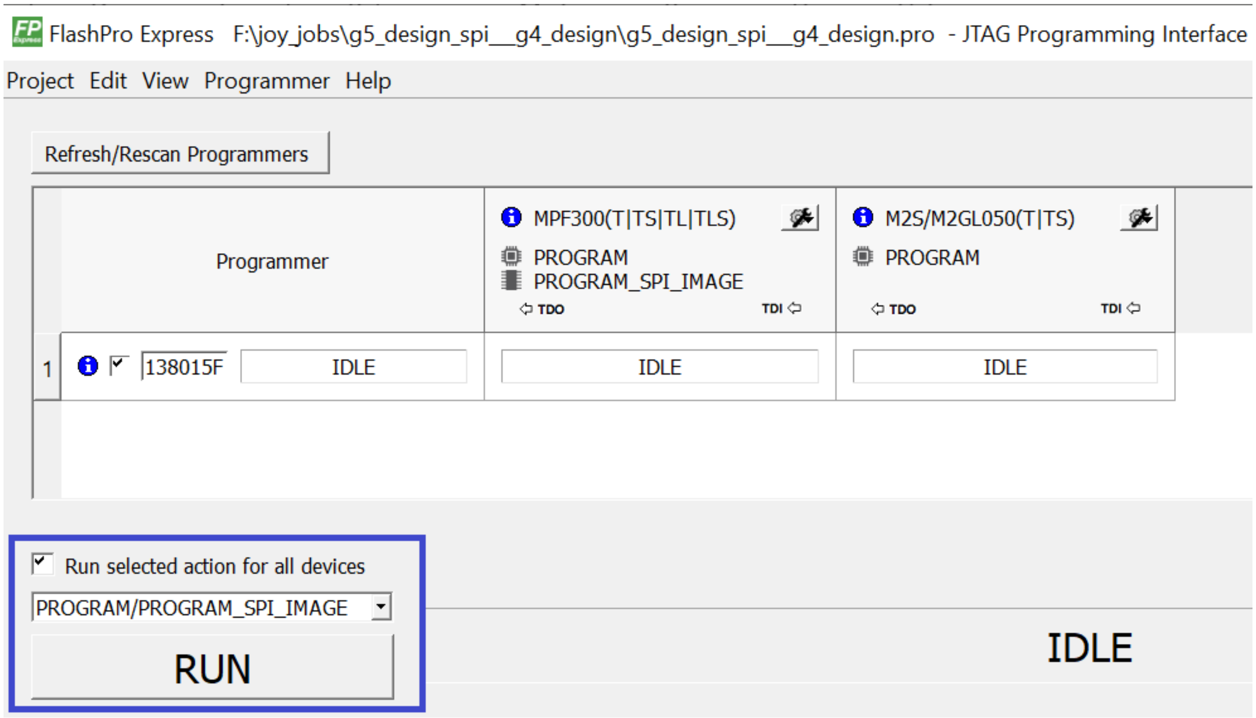


FlashPro Express F:\joy_jobs\g5_design_spi__g4_design\g5_design_spi__g4_design.pro - JTAG Programming Interface

Project Edit View Programmer Help
Refresh/Rescan Programmers
Programmer
MPF300(T|TS|TL|TLS)
PROGRAM
TDO
TDI
M2S/M2GL050(T|TS)
1
138015F
IDLE
IDLE
✓ Enable Device for Programming...
Load Programming File...
Select and Configure Action/Procedures to program design...
Load SPI Flash File...
Select SPI Flash Action...
□ Run selected action for all devices

- Enable/Disable device for programming. The option allows the device to be enabled or disabled (put in "bypass"). You must load the programming file when enabling a device that is in "bypass," with no programming file associated with the device or SPI-Flash. The device header info tooltip is updated with selected programming action if enabled or "bypass" if disabled.
  Note: The job cannot be saved if all devices are disabled or if any enabled devices do not have a loaded programming file. If all the devices are disabled, an actions combo box and a **Run** button are disabled.
- Load Programming File. The option is available for the enabled devices to load a different programming file for the target device. FlashPro Express requires all programming files to be in the local job folder. When loading a programming file from outside the job folder the file will be copied to the job folder first and then loaded for the selected device. The user must confirm copying the programming file to the job folder and overwriting the existing file.
- Select a programming action and configure actions and procedures per device. The option allows the selection of the programming action and configuration of the actions' procedures. The option is available for an enabled device that has programming file loaded, and when the **Run selected action for all devices** option is unselected.
- Load SPI Flash file. This option allows you to load a different SPI Flash programming file.
- Select SPI Flash Action. The option allows the selection of the programming action for the SPI Flash. The option is available for the enabled device that has SPI Flash Programming File loaded, and when the **Run selected action for all devices** option is unselected.
- Program device selected actions. In Developer mode, when the **Run selected action for all devices** check box is selected, FlashPro Express runs selected action from the drop-down list below the check box for all enabled devices – similar to Operator mode.

Figure 7-5. Run One Action for All Chain Devices



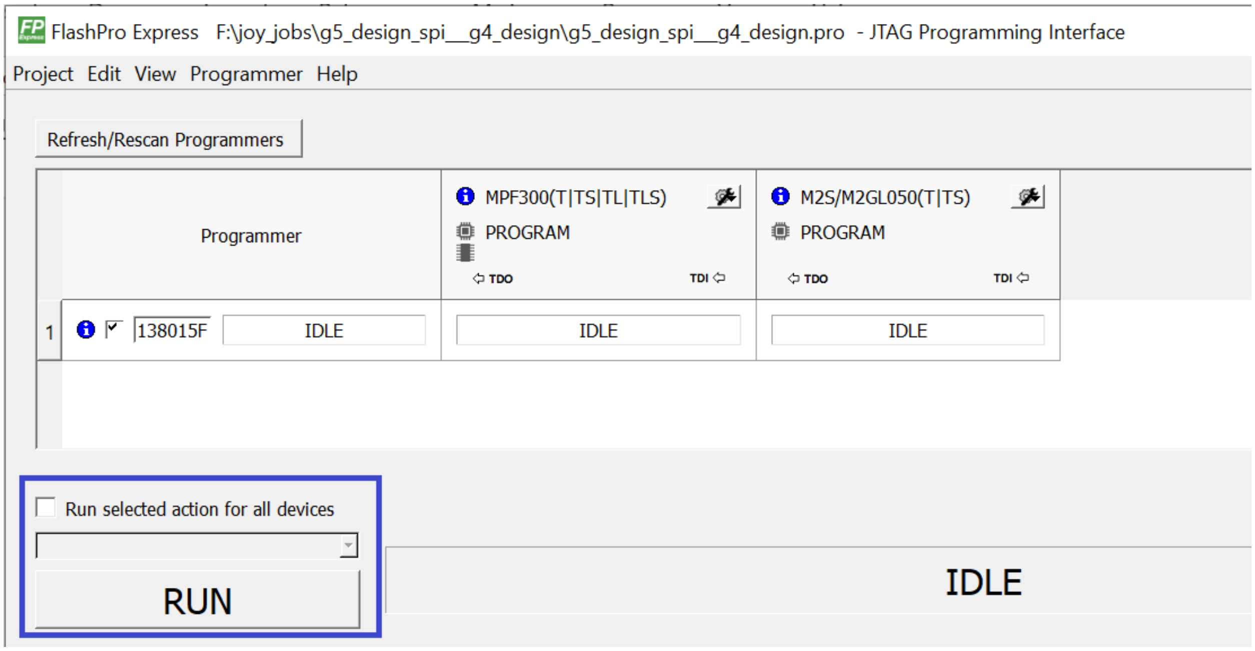
FlashPro Express F:\joy_jobs\g5_design_spi__g4_design\g5_design_spi__g4_design.pro - JTAG Programming Interface

☑ Run selected action for all devices

PROGRAM/PROGRAM_SPI_IMAGE

When the **Run selected action for all devices** check box is unselected, the actions drop-down list is disabled. FlashPro Express runs the programming actions for the enabled device and SPI Flash, as selected uniquely for each device.

Figure 7-6. Run Device Selected Actions



FlashPro Express F:\joy_jobs\g5_design_spi__g4_design\g5_design_spi__g4_design.pro - JTAG Programming Interface

☐ Run selected action for all devices

7.1. Updating FlashPro6 Internal Design (Ask a Question)

FlashPro6 programmer internal design is updated to handle a corner case of JTAG state machine traversal. This corner case is not observed using Libero or FlashPro Express tools, but it needs to be fixed for integrating with other debug tools. The FlashPro6 design update feature is added as part of the Libero or FlashPro Express software flow to detect the current design version of the attached programmer and trigger an update if you choose to. This update is possible only in Developer mode. Operator mode does not support this feature. The tool displays a warning message in the following two scenarios:

- If you choose not to proceed with the update in Developer mode.
- If the tool is in Operator mode and the programmer design is out of date.

You can update FlashPro6 with FlashPro Express tool or with Tcl Commands.

7.1.1. Updating FlashPro6 Internal Design To New Version In Developer Mode (Ask a Question)

In order to update FlashPro6 internal design from old version to new version in Developer mode, you need to follow the below steps:

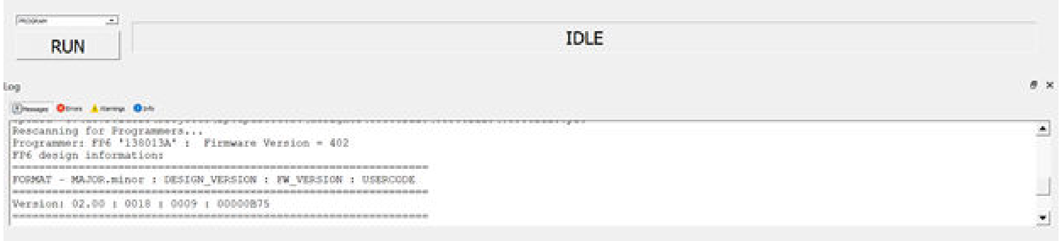
1. On connecting the FlashPro6 programmers, the **Message** window displays the design information.
2. After executing the **RUN** action, the FlashPro6 update starts.
3. A warning message pop up displays all the FlashPro6 programmers that need to be updated. You can choose whether you want the programmer to be updated with the latest design. If **Yes**, the flow starts updating all the programmers listed and then completes the action. If **No**, then the flow continues with the intended action (in this case, it runs DEVICE_INFO) along with a warning message in **Warnings** window. Whenever, you choose **RUN**, it displays all the programmers that need to be updated, regardless of selecting **No** previously.
   Note: You must not disconnect the programmers while the update is running. If you do so, the programmer does not get detected by the operating system. It is recommended to contact Microchip FPGA Support to get the defective programmer repaired.

If update passes, the programmer message shows that the update passed.

The below example shows how the design version in FlashPro6 programmer 138013A is updated from old version 0018 to new version 001A.

On connecting the programmer, the message window displays the design version 0018 for FlashPro6 programmer 138013A.

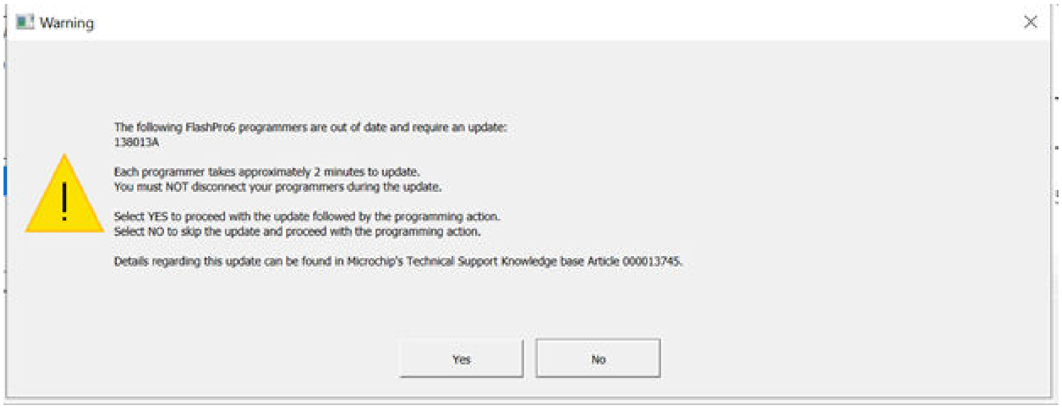
Figure 7-7. Programmer Connected With Design Version 0018



```
Rescanning for Programmers...
Programmer: FP6 '138013A' :  Firmware Version = 402
FP6 design information:
==================================================================
FORMAT - MAJOR.minor : DESIGN_VERSION : FW_VERSION : USERCODE
==================================================================
Version: 02.00 : 0018 : 0009 : 00000B75
==================================================================
```

After executing the **RUN** action, the FlashPro6 update starts. As shown in figure below, only one programmer is out of date (that is, 138013A).

Figure 7-8. Warning Message Showing The FlashPro6 Programmer 138013A To Be Out Of Date



The following FlashPro6 programmers are out of date and require an update:
138013A

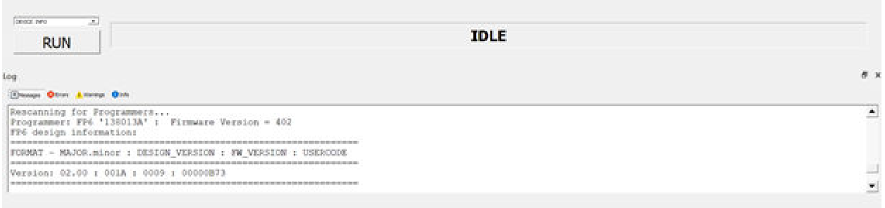
Each programmer takes approximately 2 minutes to update.
You must NOT disconnect your programmers during the update.

Select YES to proceed with the update followed by the programming action.
Select NO to skip the update and proceed with the programming action.

Details regarding this update can be found in Microchip's Technical Support Knowledge base Article 000013745.

On choosing **Yes**, the FlashPro6 programmer 138013A starts updating. If the update passes, the programmer message shows that the update passed. Once the programmer is updated, the design version of FlashPro6 programmer 138013A is updated to design version 001A.

Figure 7-9. Programmer 138013A Showing Design Version 001A On Successful Update

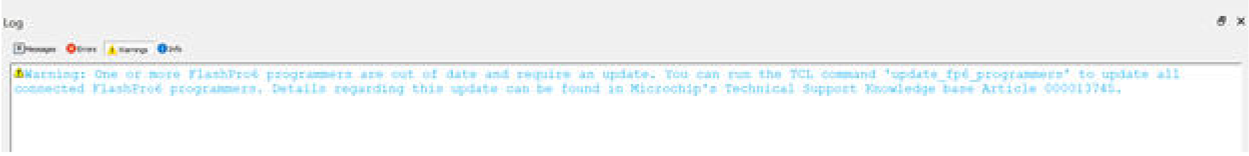


```
Rescanning for Programmers...
Programmer: FP6 '138013A' :  Firmware Version = 402
FP6 design information:
===============================================================
FORMAT - MAJOR.minor : DESIGN_VERSION : FW_VERSION : USERCODE
===============================================================
Version: 02.00 : 001A : 0009 : 00000B73
===============================================================
```

7.1.2. Updating FlashPro6 Internal Design To New Version In Operator Mode (Ask a Question)

When a FlashPro6 programmer is out of date in Operator mode, a warning message is displayed in **Warnings** window as shown in figure below.

Figure 7-10. Warning Message Displayed In Log Window Showing That Programmer Is Not Updated



```
Warning: One or more FlashPro6 programmers are out of date and require an update. You can run the TCL command 'update_fp6_programmers' to update all
connected FlashPro6 programmers. Details regarding this update can be found in Microchip's Technical Support Knowledge base Article 000013745.
```

The warning message states that one or more programmers require update. There are two options to update the programmers:

1. Switch to developer mode and run an action.
2. Use Tcl command `update_fp6_programmers` in your Tcl script.

8. Tcl Commands (Ask a Question)

This section describes the FlashPro Express Tcl commands.

8.1. About Tcl Commands (Ask a Question)

A Tool Command Language (Tcl) file contains scripts for simple or complex tasks. You can run scripts from the Windows command line, or store and run a series of Tcl commands in a `*.tcl` batch file.

You can run Tcl commands from scripts or from a Windows or Linux command line. Tcl commands are case sensitive. However, their arguments are not.

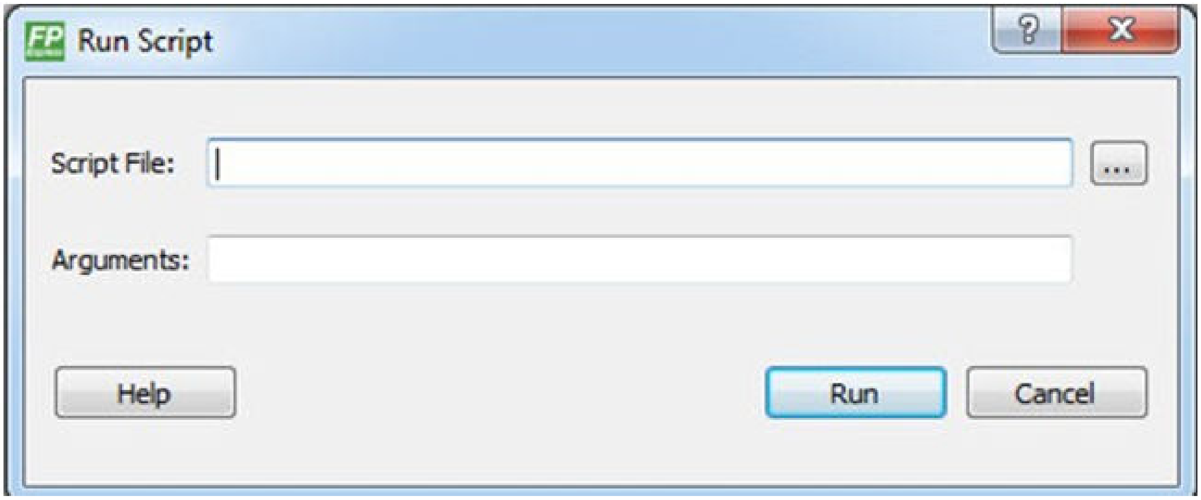
For information about all Tcl commands supported by FlashPro Express, see the Tcl Commands Reference Guide.

8.2. Executing a Tcl Script File in FlashPro Express (Ask a Question)

To execute a Tcl script in FlashPro Express:

1. From the **File** menu, choose **Execute Script** to display the Run Script dialog box.

Figure 8-1. Run Script Dialog Box



2. Click the **Browse** button to display the Open dialog box, in which you can go to the folder containing the script file to open. When you click **Open**, FlashPro Express enters the full path and script filename into the Run Script dialog box.
3. In the **Arguments** box, enter the arguments to pass to your Tcl script. Separate each argument by a space character.
4. Click **Run**.

8.3. Running Tcl Scripts from the Command Line (Ask a Question)

You can run Tcl scripts from a Windows or Linux command line.

1. At the prompt, type the path to the Microchip software followed by the word `SCRIPT`, a colon, and the name of the script file, as follows:
   ```
   <location of Microchip software>/bin/FPExpress.exe SCRIPT:<filename>
   ```
   The following example executes in batch mode the script *foo.tcl*:
   ```
   <location of Microchip software>/bin/FPExpress.exe script:foo.tcl
   ```
   The following example executes in batch mode the script foo.tcl and exports the log in the file *foo.txt*:

```
<location of Microchip software>/bin/FPExpress.exe script:foo.tcl
logfile:foo.txt
```

The following example executes in batch mode the script *foo.tcl*, creates a console where the log is displayed briefly, and exports the log in the file *foo.txt*:

```
<location of Microchip software>/bin/FPExpress.exe script:foo.tcl
console_mode:brief logfile:foo.txt
```

If you leave console_mode unspecified or set it to **hide**, FlashPro Express executes without a console window. To leave the console window open, run the script with the console_mode parameter set to **show**, as in the following example:

```
<location of Microchip software>/bin/FPExpress.exe script:foo.tcl
console_mode:show logfile:foo.txt
```

2. To pass arguments to the Tcl script from the command line, use the `SCRIPT_ARGS` variable, as follows:

```
<location of Microchip software>/bin/FPExpress.exe SCRIPT:<filename>
SCRIPT_ARGS:"param1 param2 param3"
```

Arguments passed to a Tcl script can be accessed through the Tcl variables `argc` and `argv`. The following examples show how a Tcl script accesses these arguments:

```
puts "Script name: $argv0"
puts "Number of arguments: $argc" set i 0
foreach arg $argv { puts "Arg $i : $arg" incr i
}
```

Note: If the script name is protected with double quotes, script names can contain spaces. For example:

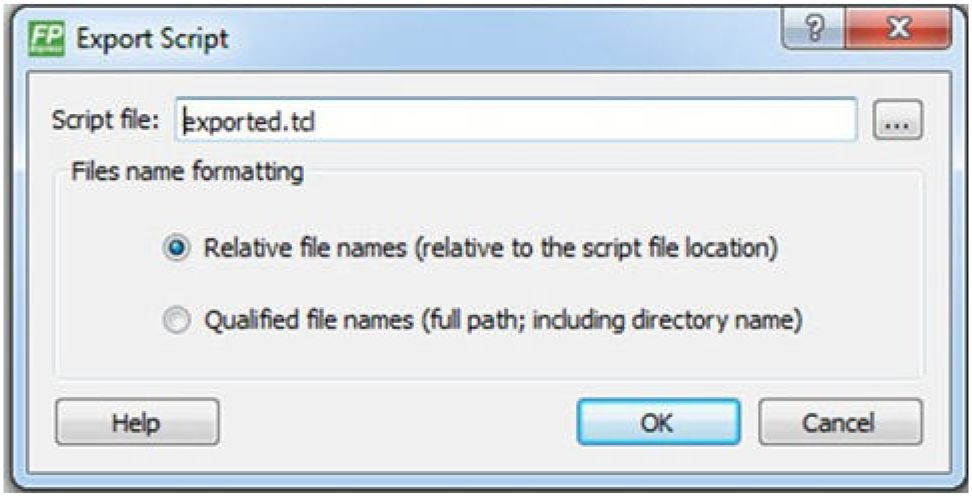
```
FPExpress script:"FPExpress tcl/foo 1.tcl"
```

8.4. Exporting Tcl Scripts from FlashPro Express (Ask a Question)

To export Tcl scripts from FlashPro Express:

1. From the **File** menu, choose **Export Script File**.
2. Enter the filename and click **Save**. The Export Script Options dialog appears, as shown in the following figure.

Figure 8-2. Script Export Options Dialog Box



Script file: exported.tcl

Files name formatting

● Relative file names (relative to the script file location)

○ Qualified file names (full path; including directory name)

3. Check the **Include commands from current project only** to export commands of the current project only. You can specify the filename formatting by selecting **Relative filenames** (relative to the current directory) or **Qualified filenames** (absolute path, including the directory name).

4. Click **OK**.

9. Troubleshooting (Ask a Question)

This section lists the exit codes for PolarFire, RTG4, and SmartFusion 2 and IGLOO 2.

9.1. Exit Codes Applicable to All Families (Ask a Question)

The following table lists the exit codes applicable to all families.

Table 9-1. Exit Codes Applicable To All Families

Programmer	Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
eFP6 (Does not apply to FlashPro6)	—	—	Failed to open eFP6 HID handle	This error could be caused by the programmer being out of sync with the software application.	Unplug the USB cable from either the programmer or the host PC and reconnect it.
FlashPro6 (Does not apply to eFP6)	—	-7	Failed to perform cyusb_bulk_transfer	This error could be caused by poor hardware USB connection to the host. **Note:** This error is specific to FP6 only. The occurrence of the error is random and cannot be predicted.	Try unplugging and plugging the programmer and rescan for new programmers to refresh the programmer list.
FlashPro5	4	—	General device I/O error	Loss of USB packets while performing an action with FlashPro5 or embedded FlashPro5 programmers.	Make sure that the USB cable is firmly connected to the programmer and PC. Avoid placing other cables on top or near the USB cable to minimize noise.

9.2. PolarFire SoC Exit Codes (Ask a Question)

The following table lists the PolarFire SoC exit codes.

Table 9-2. PolarFire SoC Exit Codes

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8020	-22	Invalid Public Key for Boot Mode 3 eNVM client	Public key may have been entered incorrectly.	Ensure that the public key is entered correctly.

9.3. PolarFire Exit Codes (Ask a Question)

The following table lists the PolarFire exit codes.

Table 9-3. PolarFire Exit Codes

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
—	0	Passed	—	—
0x8003	5	Failed to enter programming mode	Unstable/Insufficient voltage level Signal integrity issues on JTAG pins DEVRST_N is tied to LOW.	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. For more information on transient specifications, see the corresponding datasheet. Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection. Tie DEVRST_N to HIGH prior to programming the device.

Table 9-3. PolarFire Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8004	6	Failed to verify IDCODE	Incorrect programming file Incorrect device in chain Signal integrity issues on JTAG pins	Choose the correct programming file and select the correct device in the chain. Measure JTAG pins and noise for reflection. If TRST is left floating, then add pull-up to pin. Reduce the length of Ground connection.
0x8005 0x8006 0x8007 0x8008 0x8009	11	Failed to verify FPGA Array Failed to verify Fabric Configuration Failed to verify Security Failed to verify sNVM Failed to verify eNVM	Device is programmed with a different design or the component is blank. Unstable/Insufficient voltage level Signal integrity issues on JTAG pins	Verify the device is programmed with the correct data/design. Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. For more information on transient specifications, see the corresponding datasheet. Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection.
0x8013	–18	External digest check via JTAG/SPI Slave is disabled.	External Digest check via JTAG/SPI Target is disabled.	Need to use a bit stream file with a valid FlashLock/UPK1 to enable external digest check through JTAG/SPI Target.
0x801B	–20	CHECK FABRIC digest verification: FAIL sNVM digest verification: FAIL eNVM digest verification: FAIL CC digest verification: FAIL UL digest verification: FAIL UKDIGEST0 digest verification: FAIL UKDIGEST1 digest verification: FAIL UKDIGEST2 digest verification: FAIL UKDIGEST3 digest verification: FAIL UKDIGEST4 digest verification: FAIL UKDIGEST5 digest verification: FAIL UKDIGEST6 digest verification: FAIL UKDIGEST7 digest verification: FAIL UKDIGEST8 digest verification: FAIL UKDIGEST9 digest verification: FAIL UKDIGEST10 digest verification: FAIL UPERM digest verification: FAIL	Component is either erased, or the data has been corrupted or tampered with.	Reprogram the device.

Table 9-3. PolarFire Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x801C	-20	SYS digest verification: FAIL	Factory have been erased through zeroization, or the data has been corrupted or tampered with.	Switch the device (return for FA).
0x801D	–21	Device security prevented operation.	The device is protected with user pass key 1 and the bit stream file does not contain user pass key 1. User pass key 1 in the bit stream file does not match the device.	Run DEVICE_INFO to view security features that are protected. Provide a bit stream file with a user pass key 1 that matches the user pass key 1 programmed into the device.
0x801F	-22	Bitstream data is corrupted or noisy.	Bitstream file has been corrupted or was incorrectly generated.	Regenerate bitstream file. Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications.
0x8021	–23	Programming Error Invalid/Corrupted encryption key.	File contains an encrypted key that does not match the device File contains user encryption key but device has not been programmed with the user encryption key.	Provide a programming file with an encryption key that matches that on the device. First program security with initiater programming file, then program with user encryption 1/2 field update programming files.
0x8023	–24	Programming Error Back level not satisfied.	Design version is not higher than the back-level programmed device.	Generate a programming file with a design version higher than the back level version.
0x8001	–24	Failed to read DSN	Device is in System Controller Suspend Mode. Check board connections.	TRSTB should be driven High or disable System Controller Suspend Mode.
0x8027	–26	Programming Error Insufficient device capabilities.	Device does not support the capabilities specified in programming file.	Generate a programming file with the correct capabilities for the target device.
0x8029	–27	Programming Error. Incorrect DEVICEID	Incorrect programming file. Incorrect device in chain. Signal integrity issues on JTAG pins.	Choose the correct programming file and select the correct device in chain. Measure JTAG pins and noise or reflection. If TRST is left floating, then add pull-up to pin. Reduce the length of ground connection.
0x802B	–28	Programming Error Signal integrity issues on JTAG pins, please check.	Signal integrity issues on JTAG pins, please check.	Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection.
0x8030	–31	Programming Error Invalid or inaccessible Device Certificate.	FAB_RESET_N is tied to ground.	FAB_RESET_N should be tied to HIGH.
0x8032	–32	Instruction timed out	Unstable/Insufficient voltage level Signal integrity issues on JTAG pins	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection.

Table 9-3. PolarFire Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8010	–35	Failed to unlock user pass key 1	Pass key in file does not match device.	Provide a programming file with a pass key that matches pass key programmed into the device.
0x8011	–35	Failed to unlock user pass key 2	Pass key in file does not match device.	Provide a programming file with a pass key that matches pass key programmed into the device.
0x804F	–38	Bitstream programming action is disabled	Unstable/Insufficient voltage level Bitstream programming action has been disabled in Security Policy Manager.	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Use a bit stream file that has a valid FlashLock/UPK1 to enable the bit stream programming action.
0x805B	–42	Error: security must be either programmed on a blank device or with the FPGA Fabric design.	Security only bit stream programming on a programmed device.	Use this bit stream on a blank device or generate a new bit stream that contains the FPGA Fabric design along with the security.
0x8059	6	Failed to verify IDCODE MPF300(XT|T_ES|TS_ES) programming file is not compatible with MPF300 production devices. You must use a programming file for MPF300(T|TS|TL|TLS) device.	MPF300(XT|T_ES|TS_ES) programming file is not compatible with MPF300 production devices.	You must use a programming file for MPF300(T|TS|TL|TLS) device.
0x805A	6	Failed to verify IDCODE MPF300(T|TS|TL|TLS) programming file is not compatible with MPF300(XT|T_ES|TS_ES) devices. You must use a programming file for MPF300(XT|T_ES|TS_ES) device.	MPF300(T|TS|TL|TLS) programming file is not compatible with MPF300(XT|T_ES|TS_ES) devices.	You must use a programming file for MPF300(XT|T_ES|TS_ES) device.
0x80EB	7	Error reading device integrity bits.	PUF or crypto core failed Factory NVM containing signing key is corrupted.	Monitor the related power supplies causing the issue during programming; check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Switch the device (return for FA)
0x802C	14	sNVM Encryption Key (SMK) is not initialized	sNVM Encryption Key (SMK) is not initialized.	Initialized sNVM Encryption Key (SMK)
0x8014	-19	Failed while running verify digest	Factory parameters are corrupted.	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Switch the device (return for FA)

Table 9-3. PolarFire Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x80EA	-36	Device certificate could not be validated due to internal error	PUF or crypto core failed Factory NVM containing signing key is corrupted.	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Switch the device (return for FA)
0x8051	-40	Failed to zeroize the device	Zeroization through JTAG/SPI Target is disabled in user security settings.	Enable zeroization through JTAG/SPI Target in user security settings.
0x8056	-42	Failed to read zeroization certificate	Error occurred during generation of zeroization certificate due to corrupted factory parameters. Zeroization completion data is invalid.	Switch the device (return for FA). Make sure to run zeroization before reading the result.
0x8040	-32	Brownout error detected during programming	Supplied power is insufficient for the device.	Connect the device to an adequate and stable power supply.
0x805D	6	Failed to verify IDCODE. NOTE: RTPF500 {T|L|S} and RTPF500Z{T|L|S}	RT PolarFire and PolarFire programming files are not interchangeable.	Ensure that the device selection in the Libero SoC project that is used to generate the programming file matches the device part number being programmed.
0x800A	11	Failed to verify Security.	User lock settings may have changed.	Verify that the user lock settings remain consistent.
0x800C	11	Failed to verify Security.	Permanent lock settings may have changed.	Verify that the permanent lock settings remain consistent.

9.4. RTG4 Exit Codes (Ask a Question)

The following table lists the RTG4 exit codes.

Table 9-4. RTG4 Exit Codes

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
—	0	Passed	—	—
0x802B	-28	Signal integrity issues on JTAG pins, please check.	Signal integrity issues on JTAG pins, please check.	Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection.
0x8001	-24	Failed to read DSN	Device is in System Controller Suspend Mode. Check board connections.	TRSTB should be driven High on device power-up. Disable System Controller Suspend Mode in the Programming Bitstream Settings tool within Libero and reprogram the device.
0x8002	5	Device is busy	Unstable/Insufficient VDDIx voltage level.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications.

Table 9-4. RTG4 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8003	5	Failed to enter programming mode	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins. DEVRST_N is tied to LOW.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection. Tie DEVRST_N to HIGH prior to programming the device.
0x8004	6	Failed to verify IDCODE	Incorrect programming file. Incorrect device in chain. Signal integrity issues on JTAG pins.	Choose the correct programming file and select the correct device in the chain. Measure JTAG pins and noise for reflection. If TRST is left floating then add pull-up to pin. Reduce the length of Ground connection.
0x8005	6	Failed to verify IDCODE RT4G150_ES STAPL file is not compatible with RT4G150 production devices. You must use a STAPL file for RT4G150 device.	Programming file is for RT4G150_ES and device is RT4G150. Incorrect programming file Incorrect device in chain. Signal integrity issues on JTAG pins.	Generate a programming file for RT4G150 device. Choose the correct programming file and select the correct device in the chain. Measure JTAG pins and noise for reflection. If TRST is left floating then add pull-up to pin. Reduce the length of Ground connection.
0x8006	6	Failed to verify IDCODE RT4G150 STAPL file is not compatible with RT4G150_ES devices. You must use a STAPL file for RT4G150_ES device.	Programming file is for RT4G150 and device is RT4G150_ES. Incorrect programming file Incorrect device in chain. Signal integrity issues on JTAG pins.	Generate a programming file for RT4G150_ES device. Choose the correct programming file and select the correct device in the chain. Measure JTAG pins and noise for reflection. If TRST is left floating then add pull-up to pin. Reduce the length of Ground connection.
0x8008	-17	Device is blank	Attempting to verify digest of a blank device.	Program the device prior to running the **VERIFY_DIGEST** action.
0x8009	-18	Fabric digest check is disabled	Digest check has been disabled by the Programming Bitstream Settings tool within Libero.	Drive TRSTB high during device power-up. Enable digest check in the Programming Bitstream Settings tool within Libero and reprogram the device.
0x8009	-19	Failed to verify digest: Instruction timed out	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Run the **VERIFY_DIGEST** action again. Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.

Table 9-4. RTG4 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x800B	-20	FPGA Fabric digest verification: FAIL	Programming bit stream components do not match components programmed. FPGA Fabric is either erased or the data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x800C	-20	Factory row segment digest verification: FAIL	Programming bit stream components do not match components programmed. Factory row segment data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x800D	-22	Bitstream Error. Bitstream or data is corrupted or noisy.	Bitstream file has been corrupted. Bitstream was incorrectly generated. Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Regenerate bit stream file. Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x800E	15	Failed to query programming bit stream settings: Instruction timed out	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Run the **DEVICE_INFO** action again. Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x800F	-27	Bitstream Error. Incorrect DEVICEID	Incorrect programming file. Incorrect device in chain. Signal integrity issues on JTAG pins.	Choose the correct programming file and select the correct device in the chain. Measure JTAG pins and noise for reflection. If TRST is left floating then add pull-up to pin. Reduce the length of Ground connection.
0x8011	-32	Failed to check bit stream: Instruction timed out	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x8012	-32	Failed to erase device: Instruction timed out	Unstable/Insufficient voltage level Signal integrity issues on JTAG pins.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.

Table 9-4. RTG4 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8013	-32	Failed to verify device: Instruction timed out	Unstable/Insufficient voltage level Signal integrity issues on JTAG pins.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device datasheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x8014	-32	Failed to program device: Instruction timed out	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x8015	-33	Error, device is not ready.	DEVRST_N may have been driven LOW during programming.	Need to ensure that DEVRST_N is driven HIGH during programming. The reliability of the device in space cannot be ensured if this has occurred. It is the user's responsibility to ensure that DEVRST_N is driven HIGH during programming.

9.5. SmartFusion 2 and IGLOO 2 Exit Codes (Ask a Question)

Check for the following common troubleshooting steps before looking at the troubleshooting table.

1. Monitor the related power supplies that can cause issue during programming; check for transients outside of Microchip specifications.
2. Make sure to tie `DEVRST_N` to HIGH, prior to programming the device.

The following table lists the SmartFusion 2 and IGLOO 2 exit codes.

Table 9-5. SmartFusion 2 and IGLOO 2 Exit Codes

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
—	0	Passed	—	—
0x8003	5	Failed to enter programming mode	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins. DEVRST_N is tied to LOW.	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection. Tie DEVRST_N to HIGH prior to programming the device.
0x8004	6	Failed to verify IDCODE	Incorrect programming file. Incorrect device in chain. Signal integrity issues on JTAG pins.	Choose the correct programming file and select the correct device in the chain. Measure JTAG pins and noise for reflection. If TRST is left floating then add pull-up to pin. Reduce the length of Ground connection.

Table 9-5. SmartFusion 2 and IGLOO 2 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8005 0x8006 8x804A	10	Failed to program eNVM	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection.
0x8007 0x804C	11	Failed to verify FPGA Array Failed to verify Fabric Configuration Failed to verify Security	Device is programmed with a different design or the component is blank. Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Verify that the device is programmed with the correct data/design. Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x8008 0x8009 0x8049	11	Failed to verify eNVM	Device is programmed with a different design. Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Verify that the device is programmed with the correct data/design. Monitor the related power supplies causing the issue during programming; check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Monitor JTAG supply pins during programming. Measure JTAG signals for noise or reflection.
0x8013	–18	Digest request from SPI/JTAG is protected by User Pass Key 1	Digest request from SPI/JTAG is protected by user pass key 1. Lock bit has been configured in the Debug Policy within Security Policy Manager (SPM).	Provide a programming file with a pass key that matches pass key programmed into the device.
0x8014	–19	Failed while running verify digest	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Monitor the related power supplies causing the issue during programming. Check for transients outside of Microchip specifications. See your device's data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x8015	–20	FPGA Fabric digest verification: FAIL	Programming bit stream components do not match components programmed. FPGA Fabric is either erased or the data has been corrupted or tampered with.	Use the same programming file that was used to program the device.

Table 9-5. SmartFusion 2 and IGLOO 2 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8016	–20	eNVM_0 digest verification: FAIL	Programming bit stream components do not match components programmed. eNVM_0 data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x8017	–20	eNVM_1 digest verification: FAIL	Programming bit stream components do not match components programmed. eNVM_1 data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x8018	–20	User security policies digest verification: FAIL	Programming bit stream components do not match components programmed. User security policy data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x8019	–20	User key set 1 digest verification: FAIL	Programming bit stream components do not match components programmed. User key set 1 data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x801A	–20	User key set 2 digest verification: FAIL	Programming bit stream components do not match components programmed. User key set 2 data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x801B	–20	Factory row and factory key digest verification: FAIL	Programming bit stream components do not match components programmed. Factory row and factory key data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x801C	–20	Fabric configuration digest verification: FAIL	Programming bit stream components do not match components programmed. Fabric configuration data has been corrupted or tampered with.	Use the same programming file that was used to program the device.
0x801D 0x801E 0x804B	–21	Device security prevented operation	The device is protected with user pass key 1 and the bit stream file does not contain user pass key 1. User pass key 1 in the bit stream file does not match the device.	Run DEVICE_INFO to view security features that are protected. Provide a bit stream file with a user pass key 1 that matches the user pass key 1 programmed into the device.
0x801F 0x8020 0x8040	–22	Authentication Error Bitstream or data is corrupted or noisy	eNVM has been locked by a initiator in your design. Running VERIFY action on a blank device. Bitstream file has been corrupted. Bitstream was incorrectly generated.	Release the lock on the eNVM after your initiator has completed its access operations. Write 0x00 to "REQACCESS" register in eNVM Control Registers (address 0x600801FC) to release the access. Program the device prior to running VERIFY action. Regenerate the bit stream file.

Table 9-5. SmartFusion 2 and IGLOO 2 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8021 0x8022 0x8041	–23	Authentication Error Invalid/Corrupted encryption key	File contains an encrypted key that does not match the device. Attempting to erase a device with no security using initiator security file. File contains user encryption key, but device has not been programmed with the user encryption key. Device has user encryption key 1/2 enforced and you are attempting to reprogram security settings.	Provide a programming file with an encryption key that matches that on the device. Run DEVICE_INFO action to verify that the device has no security. If the device does not have security, it cannot be erased. First program security with initiator programming file, then program with user encryption 1/2 field update programming files. You must first ERASE security with the initiator security file, then you can reprogram new security settings.
0x8023 0x8024 0x8042	–24	Authentication Error Back level not satisfied.	Design version is not higher than the back-level programmed device.	Generate a programming file with a design version higher than the back level version.
0x8001	–24	Failed to read DSN	Device is in System Controller Suspend Mode. Check board connections.	TRSTB must be driven High or disable "System Controller Suspend Mode."
0x8025 0x8026 0x8043	–25	Authentication Error DSN binding mismatch	DSN specified in programming file does not match the device being programmed.	Use the correct programming file with a DSN that matches the DSN of the target device being programmed.
0x8028	–26	Authentication Error Insufficient device capabilities	Device does not support the capabilities specified in programming file.	Generate a programming file with the correct capabilities for the target device.
0x8027 0x8044	–26	Authentication Error Bitstream and device mismatch	Libero device selection does not match the target device.	Generate a programming file with the correct device selection for the target device.
0x8029 0x802A 0x8045	–27	Authentication Error Incorrect DEVICEID	Incorrect programming file. Incorrect device in chain. Signal integrity issues on JTAG pins.	Choose the correct programming file and select the correct device in chain. Measure JTAG pins and noise or reflection. If TRST is left floating, then add pull-up to pin. Reduce the length of ground connection.
0x802B	–28	Authentication Error Signal integrity issues on JTAG pins, please check.	Signal integrity issues on JTAG pins.	Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x8030 0x8031 0x8048	–31	Authentication Error Invalid or inaccessible Device Certificate.	M2S090 Rev. A or M2S150 Rev. A: Either certificate is corrupted or the user has not provided the application code in the eNVM or provided invalid application code. FAB_RESET_N is tied to ground.	User can program a valid application code. This can be done with SoftConsole. FAB_RESET_N should be tied to HIGH.

Table 9-5. SmartFusion 2 and IGLOO 2 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8032	–32	Instruction timed out	Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection.
0x8010	–35	Failed to unlock User Pass Key 1	Pass key in file does not match device. Plaintext pass key match is disabled. This occurs if HSM was used to program the device.	Provide a programming file with a pass key that matches pass key programmed into the device. Match pass key using HSM.
0x8011	–35	Failed to unlock User Pass Key 2	Pass key in file does not match device. Plaintext pass key match is disabled. This occurs if HSM was used to program the device.	Provide a programming file with a pass key that matches pass key programmed into the device. Match pass key using HSM.
0x8012	–35	Failed to unlock debug pass key	Pass key in file does not match device. Plaintext pass key match is disabled. This occurs if HSM was used to program the device.	Provide a programming file with a pass key that matches pass key programmed into the device. Match pass key using HSM.
0x804E	–37	Device already has Security programmed. Please erase the device using master file before reprogramming Security Settings.	HSM flow does not support reprogramming device directly if Security has already been programmed.	Erase security and program the device.
0x8055	–35	Please have valid Cortex®-M3 firmware code client at page 0 for Rev 0 M2S090 and M2S150 devices, else it will lead to unexpected behavior including programming lockout condition. Please see section 3.14 at ER0196: SmartFusion2 Device v1.5 Errata.	Invalid Cortex-M3 firmware code client at page 0 for Rev 0 M2S090 and M2S150 devices.	Please have valid Cortex-M3 firmware code at page 0.
0x8054	-38	Failed to read device certificate, device is busy. Failed to read device certificate (AHB HRESP error).	User has not programmed valid Cortex-M3 code in beginning page(s) of eNVM. Unstable/Insufficient voltage level. Signal integrity issues on JTAG pins.	Program valid Cortex-M3 code in beginning page(s) of eNVM. Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications. Monitor JTAG supply pins during programming; measure JTAG signals for noise or reflection."

Table 9-5. SmartFusion 2 and IGLOO 2 Exit Codes (continued)

Error Code	Exit Code	Exit Message	Possible Cause	Possible Solution
0x8052	-38	Device certificate is invalid.	Factory parameters are corrupted. Power supply issue.	Monitor related power supplies that cause the issue during programming; check for transients outside of Microchip specifications. See your device data sheet for more information on transient specifications.

10. SmartDebug (Ask a Question)

Microchip's SmartDebug tool complements design simulation by allowing verification and troubleshooting at the hardware level.

For detailed information about SmartDebug, see the SmartDebug User Guide.

11. Electrical Parameters (Ask a Question)

This section describes the FlashPro electrical parameters.

11.1. DC Characteristics for FlashPro6 (Ask a Question)

Table 11-1. DC Characteristic for FlashPro6

Parameter	Test Condition	VJTAG Voltage Range	Min.	Typ.	Max.	Unit
VIH High- level input voltage	—	1.20 V to 1.95 V	VJTAG_VSPI x 0.65	—	—	V
	—	1.95 V to 2.70 V	1.6	—	—	V
	—	2.70 V to 3.60 V	2	—	—	V
VIL Low-level input voltage	—	1.20 V to 1.95 V	—	—	VJTAG_VSPIx0.35	V
	—	1.95 V to 2.70 V	—	—	0.7	V
	—	2.70 V to 3.60 V	—	—	0.8	V
VOH	IOH = –100 µA	1.2 V to 3.6 V	VJTAG_VSPI - 0.2	—	—	V
	IOH = –3 mA	1.2 V	—	—	—	V
	IOH = –6 mA	1.4 V	1.05	—	—	V
	IOH = –8 mA	1.65 V	1.2	0.95	—	V
	IOH = –9 mA	2.3 V	1.75	—	—	V
	IOH = –12 mA	3 V	2.3	—	—	V
VOL	IOH = –100 µA	1.2 V to 3.6 V	—	—	—	V
	IOH = –3 mA	1.2 V	—	0.25	—	V
	IOH = –6 mA	1.4 V	—	—	—	V
	IOH = –8 mA	1.65 V	—	—	—	V
	IOH = –9 mA	2.3 V	—	—	—	V
	IOH = –12 mA	3 V	—	—	—	V
IOH High- level output current	—	1.1 V to 1.2 V	—	—	–3	mA
	—	1.4 V to 1.6 V	—	—	–6	mA
	—	1.65 V to 1.95 V	—	—	–8	mA
	—	2.3 V to 2.7 V	—	—	–9	mA
	—	3 V to 3.6 V	—	—	–12	mA
IOL Low-level output current	—	1.1 V to 1.2 V	—	—	3	mA
	—	1.4 V to 1.6 V	—	—	6	mA
	—	1.65 V to 1.95 V	—	8	—	mA
	—	3 V to 3.6 V	—	12	—	mA
	—	2.3 V to 2.7 V	—	9	—	mA

11.2. DC Characteristics for FlashPro5/4/3/3X (Ask a Question)

The target board must provide the VCC, VCCI, VPUMP, and VJTAG during programming.

The VJTAG signal is driven from the target/DUT board. The VJTAG pin is sensed by the FP4 to configure the internal input and output buffers to the same I/O voltage levels. The VJTAG pin is only an input pin to the programmer.

Table 11-2. DC Characteristic for FlashPro5/4/3/3X

Description	Symbol	Min.	Max.	Unit
Input low voltage, TDO	VIL	–0.5	0.35*VJTAG	V
Input high voltage, TDO	VIH	0.65*VJTAG	3.6	V
Input current, TDO	IIL, IIH	–20	+20	mA
Input capacitance, TDO			40	pF
Output voltage, VPUMP, operating	VPP	+3.0	+3.6	V
Output current, VPUMP	IPP	—	250	mA
VJTAG = 1.5 V				
Output low voltage, TCK, TMS, TDI, 100 µA load	VOL	0.0	0.2	V
Output low voltage, TCK, TMS, TDI, 4 mA load	VOL	0.0	0.30*VJTAG	V
Output high voltage, TCK, TMS, TDI, 100 µA load	V	VJTAG–0.2	VJTAG	V
Output high voltage, TCK, TMS, TDI, 4 mA load	VOH	0.70*VJTAG	VJTAG	V
Output current, TCK, TMS, TDI	IOL, IOH	–4	+4	mA
VJTAG = 1.8 V				
Output low voltage, TCK, TMS, TDI, 100 µA load	VOL	0.0	0.2	V
Output low voltage, TCK, TMS, TDI, 6 mA load	VOL	0.0	0.3	V
Output high voltage, TCK, TMS, TDI, 100 µA load	VOH	VJTAG–0.2	VJTAG	V
Output high voltage, TCK, TMS, TDI, 6 mA load	VOH	1.25	VJTAG	V
Output current, TCK, TMS, TDI	IOL, IOH	–6	+6	mA
VJTAG = 2.5 V				
Output low voltage, TCK, TMS, TDI, 100 µA load	VOL	0.0	0.2	V
Output low voltage, TCK, TMS, TDI, 8 mA load	VOL	0.0	0.6	V
Output high voltage, TCK, TMS, TDI, 100 µA load	VOH	VJTAG–0.2	VJTAG	V
Output high voltage, TCK, TMS, TDI, 8 mA load	VOH	1.8	VJTAG	V
Output current, TCK, TMS, TDI	IOL, IOH	–8	+8	mA
VJTAG = 3.3V				
Output low voltage, TCK, TMS, TDI, 100 µA load	VOL	0.0	0.2	V
Output low voltage, TCK, TMS, TDI, 8 mA load	VOL	0.0	0.4	V
Output high voltage, TCK, TMS, TDI, 100 µA load	VOH	VJTAG–0.2	VJTAG	V
Output high voltage, TCK, TMS, TDI, 8 mA load	VOH	2.4	VJTAG	V
Output current, TCK, TMS, TDI	IOL, IOH	–8	+8	mA

12. Electrical Specifications (Ask a Question)

This chapter describes the FlashPro electrical specifications.

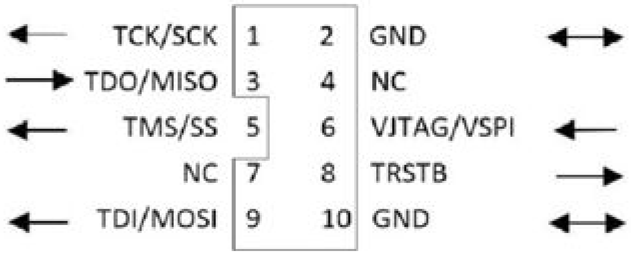
12.1. FlashPro6 (Ask a Question)

FlashPro6 is a JTAG-based programmer for Flash-based Microchip devices.

The FlashPro6 output is supplied through a connector, to which a detachable 10-pin cable is fitted. The connector on the FlashPro6 unit is a 2x5, RA male Header connector that is manufactured by 3M and has a manufacturer’s part number of N2510-5002-RB. This is a standard 2x5, 0.1-pitch keyed connector. In your board, use the 10-pin header, Samtec P/N HTST-105-01-L-DV-A, or equivalent.

The following figure shows the signals on the pins of the FlashPro6 10-pin connector.

Figure 12-1. FlashPro6 10-Pin Connector



Note: All ground pins must be connected. The rectangular shape shows connections on the programmer itself. Arrows show current flow towards or from the rectangular programmer.

The following table lists the signals.

Table 12-1. FlashPro6 Signal Description

Signal	Description
GND	Signal reference
TCK/SCK	JTAG clock; SPI clock
TDI/SDI	JTAG data input to device; SPI MOSI
TDO/SDO	JTAG data output from device; SPI MISO
TMS/SS#	JTAG mode select; SPI Chip Select
nTRST	Programmable output pin may be set to off, toggle, low, or high level.
VJTAG	Reference voltage from the target board **Note:** The current drawn by the JTAG pin for FlashPro-6 programmers is approximately 30 mA.

Some designers of high-integrity military and avionic boards may arrange their boards so that TRST is tied to ground through a weak pull-down resistor. The purpose of this is to hold the JTAG state machine in a Reset state by default, so that even with TCK oscillating, some sudden ion bombardment or other electrical event will not suddenly throw the JTAG state machine into an unknown state. If your design also uses a weak pull-down resistor on TRST on your board, enabling the “Drive TRST” flag will be required to force the JTAG state machine out of reset, to permit programming to take place. With most boards, there is no need to select this flag.

12.2. FlashPro5 (Ask a Question)

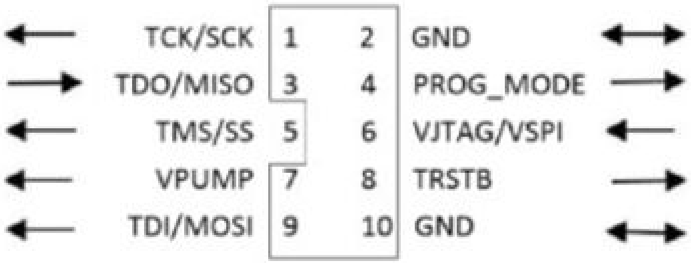
FlashPro5 is a JTAG- and a SPI-based programmer for Flash-based Microchip devices.

The FlashPro5 output is supplied via a connector to which a detachable 10-pin cable is fitted. The connector on the FlashPro5 unit is a 2x5, RA male Header connector that is manufactured by

AMP and has a manufacturer's part number of 103310-1. This is a standard 2x5, 0.1-pitch keyed connector. Use the 10-pin right-angle header, AMP P/N 103310-1 (DigiKey P/N A26285-ND) for FlashPro4 and use the 10-pin straight header, AMP P/N 103308-1 (DigiKey P/N A26267-ND) for the straight version.

The following figure shows the signals on the FlashPro5 10-pin connector pins.

Figure 12-2. FlashPro5 10-Pin Connector



Note: All ground pins must be connected. The rectangular shape shows connections on the programmer itself. Arrows show current flow towards or from the rectangular programmer.

The following table lists the signals.

Table 12-2. FlashPro5 Signal Description

Signal	Description
VPUMP	3.3V Programming voltage
GND	Signal reference
TCK/SCK	JTAG clock; SPI clock
TDI/SDI	JTAG data input to device; SPI MOSI
TDO/SDO	JTAG data output from device; SPI MISO
TMS/SS#	JTAG mode select; SPI Chip Select
nTRST	Programmable output pin may be set to off, toggle, low, or high level
VJTAG	Reference voltage from the target board **Note:** The current drawn by the JTAG pin for FlashPro 5 programmers is approximately 30 mA.
PROG_MODE	IGLOO® v2 family - used for switching from VCC 1.2V to 1.5V during programming

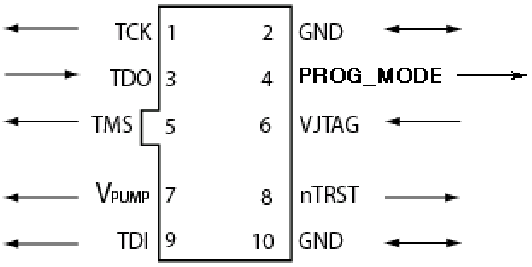
Some designers of high-integrity boards (military and avionic) may arrange their boards so that TRST is tied to ground via a weak pull-down resistor. The purpose of this is to hold the JTAG state machine in a Reset state by default, so that even with TCK oscillating, some sudden ion bombardment or other electrical even will not suddenly throw the JTAG state machine into an unknown state. If your design also uses a weak pull-down resistor on TRST on your board, enabling the "Drive TRST" flag will be required to force the JTAG state machine out of reset to permit programming to take place. With most boards, there is no need to select this flag.

12.3. FlashPro4 (Ask a Question)

The FlashPro4 output is supplied via a connector to which a detachable 10-pin cable is fitted. The connector on the FlashPro4 unit is a 2x5, RA male Header connector that is manufactured by AMP and has a manufacturer's part number of 103310-1. This is a standard 2x5, 0.1-pitch keyed connector. Use the 10-pin right-angle header, AMP P/N 103310-1 (DigiKey P/N A26285-ND) for FlashPro4 and use the 10-pin straight header, AMP P/N 103308-1 (DigiKey P/N A26267-ND) for the straight version.

The following figure shows the signals on the FlashPro4 10-pin connector pins.

Figure 12-3. FlashPro4 10-Pin Connector



Note: All ground pins must be connected. The rectangular shape shows connections on the programmer itself. Arrows show current flow towards or from the rectangular programmer.

The following table lists the signals.

Table 12-3. FlashPro4 Signal Description

Signal	Description
VPUMP	3.3V Programming voltage
GND	Signal reference
TCK	JTAG clock
TDI	JTAG data input to device
TDO	JTAG data output from device
TMS	JTAG mode select
nTRST	Programmable output pin may be set to off, toggle, low, or high level
VJTAG	Reference voltage from the target board **Note:** The current drawn by the JTAG pin for FlashPro-4 programmers is approximately 30 mA.
PROG_MODE	IGLOO® v2 family - used for switching from VCC 1.2V to 1.5V during programming

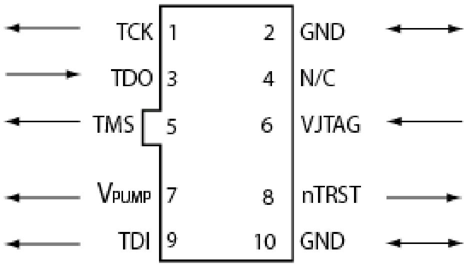
Some designers of high-integrity boards (military and avionic) may arrange their boards so that TRST is tied to ground via a weak pull-down resistor. The purpose of this is to hold the JTAG state machine in a Reset state by default, so that even with TCK oscillating, some sudden ion bombardment or other electrical event will not suddenly throw the JTAG state machine into an unknown state. If your design also uses a weak pull-down resistor on TRST on your board, enabling the "Drive TRST" flag will be required to force the JTAG state machine out of reset to permit programming to take place. With most boards, there is no need to select this flag.

12.4. FlashPro3 (Ask a Question)

The FlashPro3 output is supplied via a connector to which a detachable 10-pin cable is fitted. The connector on the FlashPro3 unit is a 2x5, RA male Header connector that is manufactured by AMP and has a manufacturer's part number of 103310-1. This is a standard 2x5, 0.1-pitch keyed connector. Use the 10-pin right-angle header, AMP P/N 103310-1 (DigiKey P/N A26285-ND) for FlashPro5/4/3/3X and use the 10-pin straight header, AMP P/N 103308-1 (DigiKey P/N A26267-ND) for the straight version.

The following figure shows the signals on the FlashPro3 10-pin connector pins.

Figure 12-4. FlashPro3 10-Pin Connector



Note: All ground pins must be connected. The rectangular shape shows connections on the programmer itself. Arrows show current flow towards or from the rectangular programmer.

The following table lists the signals.

Table 12-4. FlashPro3 Signal Description

Signal	Description
VPUMP	3.3 V Programming voltage
GND	Signal reference
TCK	JTAG clock
TDI	JTAG data input to device
TDO	JTAG data output from device
TMS	JTAG mode select
nTRST	Programmable output pin may be set to off, toggle, low, or high level
VJTAG	Reference voltage from the target board
N/C	Programmer does not connect to this pin

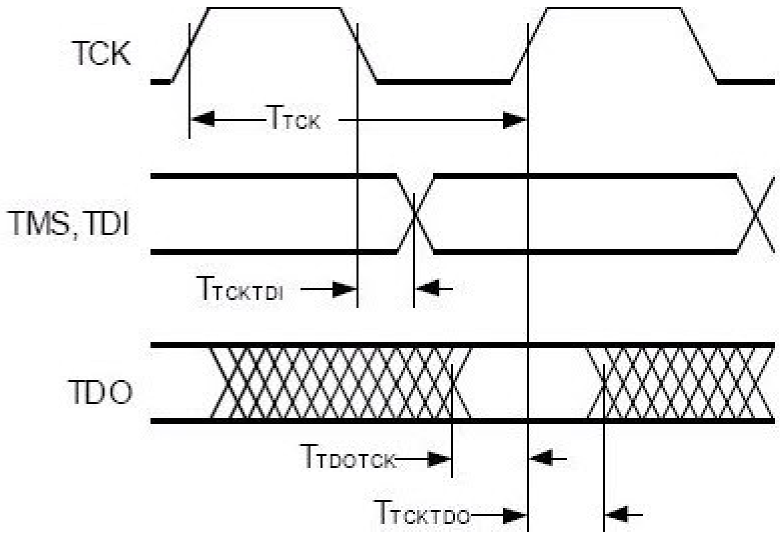
Some designers of high-integrity military and avionic boards may arrange their boards so that TRST is tied to ground via a weak pull-down resistor. The purpose of this is to hold the JTAG state-machine in a reset state by default, so that even with TCK oscillating, some sudden ion bombardment or other electrical even will not suddenly throw the JTAG state-machine into an unknown state. If your design also uses a weak pull-down resistor on TRST on your board, enabling the “Drive TRST” flag will be required to force the JTAG state-machine out of reset to permit programming to take place. With most boards, there is no need to select this flag.

12.5. JTAG Switching Characteristics (Ask a Question)

This section describes the FlashPro JTAG switching characteristics.

The following figure shows the JTAG switching characteristics.

Figure 12-5. JTAG Switching Characteristics



12.5.1. FlashPro6 Characteristics (Ask a Question)

Table 12-5. JTAG Switching Characteristics for FlashPro6

Description	Symbol	Min.	Max.	Unit
Output Delay from TCK to TDI, TMS	TTCKTDI	2	2.2	ns
TDO setup time before TCK rising, VJTAG=3.3 V	TTDOTCK	11.4	481	ns
TDO setup time before TCK rising, VJTAG=1.5 V	TTDOTCK	10.5	487	ns
TDO Hold time after TCK rising	TTCKTDO	0	—	
TCK period	TTCK	49.4	—	ns

12.5.2. FlashPro5/4/3/3X Characteristics (Ask a Question)

Table 12-6. JTAG Switching Characteristics for FlashPro5/4/3/3X

Description	Symbol	Min.	Max.	Unit
Output delay from TCK to TDI, TMS	TTCKTDI	–2	2	ns
TDO setup time before TCK rising, VJTAG=3.3	TTDOTCK	12	—	ns
TDO setup time before TCK rising, VJTAG=1.5	TTDOTCK	14.5	—	ns
TDO hold time after TCK rising	TTCKTDO	0	—	ns
TCK period	TTCK	41.7	10667	ns

13. FlashPro Express Reference (Ask a Question)

Use this chapter as a reference for the FlashPro Express user interface.

13.1. FlashPro Express Start Page (Ask a Question)

The FlashPro Express Start page is the first page that appears when the tool starts. This page provides the interface for loading a project into the tool by either navigating to the project location or clicking a recently opened project.

13.2. FlashPro Express Project Menu (Ask a Question)

The following table lists the FlashPro express project menu details.

Table 13-1. FlashPro Express Project Menu

Command	Function
New Job Project	New job project folder with programming job name will be created at the specified location.
Open Job Project	Loads a job project into the tool by reading the information in the user-specified `.pro` file.
Close Job Project	Closes the current job project.
Save Job Project	Saves the current job project.
Set Log File	Sets the location of the Log file to your specified location.
Export Log File	Exports the Log file to your specified location.
Preferences	Allows you to select FlashPro Express mode.
Execute Script	Runs your specified Tcl script.
Export Script File	Exports all commands run in this session to your specified path as a Tcl script.
Exit	Exits FlashPro Express.

13.3. FlashPro Express Edit Menu (Ask a Question)

Table 13-2. FlashPro Express Edit Menu

Command	Function
Clear Log Window	Clears the Log window.

13.4. FlashPro Express View Menu (Ask a Question)

Table 13-3. FlashPro Express View Menu

Command	Function
Log Window	Shows or hides the Log window.

13.5. FlashPro Express Tools Menu (Ask a Question)

The following table lists the FlashPro express tools menu details.

Table 13-4. FlashPro Express Tools Menu

Command	Function
Programming Connectivity and Interface	The Programming Connectivity and Interface window displays the physical chain from TDI to TDO or SPI Target configuration.
Programmer Settings	Opens the Programmer Settings dialog box in which you can set options for supported Microchip programmers.

Table 13-4. FlashPro Express Tools Menu (continued)

Command	Function
Disable Core Check During Scan and Check Chain	Disables core checking during scan and check chain operations.

13.6. FlashPro Express Help Menu (Ask a Question)

Table 13-5. FlashPro Express Help Menu

Command	Function
Help Topics	Opens the help.
Microchip Technical Support	Opens the Microchip technical support site.
Microchip Web Site	Opens the Microchip Website in your default browser.
User Guide	Opens the FlashPro Express User Guide.
Check for Software Updates	Checks for software updates if you are connected to the Internet.
About FlashPro Express	Lists the FlashPro Express release information.

13.7. FlashPro Express Log Window and Status Bar (Ask a Question)

The following sections describe the FlashPro Express Log window and status bar.

13.7.1. FlashPro Express Log Window (Ask a Question)

The FlashPro Express Log window shows status messages for user activity.

- Click the appropriate tab (**Messages**, **Errors**, **Warning**, and **Info**) to filter messages by type.
- Use the right-click menu to copy text, clear the log, and scroll the log.
- Use the def variable LOG_WINDOW_BUFFER_SIZE to set the buffer size.
- Use the **View** menu to show or hide the Log window.

13.7.2. FlashPro Express Status Bar (Ask a Question)

The Status Bar at the bottom shows the status of the load project action.

14. Appendix A: Sample Programming and SmartDebug Times Using FlashPro5 and FlashPro6 (Ask a Question)

The tables in this appendix show sample programming times using the FlashPro5 and FlashPro6 programmers.

The following table shows sample PPD programming times.

Table 14-1. Sample PPD Programming Times

Devices[1]	PPD Programming Time[2] (mm:ss)		
	FlashPro5	FlashPro6	
	TCK = 4 MHz	TCK = 4 MHz	TCK = 20 MHz[3]
	USB 2.0	USB 2.0/3.0	USB 2.0/3.0
M2S/A2GL 005	—	—	—
M2S/A2GL 010	—	—	—
M2S/A2GL 025	—	—	—
M2S/A2GL 050	2 min 9 sec	2 min 10 sec	2min 2 sec
M2S/A2GL 060	—	—	—
M2S/A2GL 090	—	—	—
M2S/A2GL 150	4 min 21 sec	4 min 19 sec	3 min 54 sec
RTG4	2 min 10 sec	1 min 56 sec	1 min 33 sec
MPF100	39 sec	28 sec	23 sec
MPF200	1 min 3 sec	43 sec	28 sec
MPF300	1 min 33 sec	1 min 4 sec	43 sec
MPF500	1 min 57 sec	1 min 34 sec	1 min

Notes:

- [1] FlashPro6 supports JTAG programming for all SmartFusion 2, IGLOO 2, RTG4 and PolarFire devices.
- [2] To benefit from the improved programming time using FlashPro6, use the PPD file format for SmartFusion 2, IGLOO 2 and PolarFire devices. Programming time speed up with PPD will be added in future releases.
- [3] To program the device at 20 MHz TCK, take appropriate steps to ensure signal integrity of the JTAG signals.

The following table shows sample SPI Flash programming times, all of which use PPD flow.

Table 14-2. SPI Flash Programming

(N25Q00AA13GSF40G / MT25QL01GBBB8ESF-0SIT TR)[1] 10 MByte data	PPD Programming Time				
	FlashPro5		FlashPro6[2]		
	TCK = 4 MHZ	TCK = 15 MHz[3]	TCK = 4 MHZ	TCK = 15 MHz[3]	TCK = 20 MHz[3]
	USB 2.0	USB 2.0	USB 2.0/3.0	USB 2.0/3.0	USB 2.0/3.0
Erase and Program SPI Flash[4]	8 min 15 sec	4 min 58sec	14 min 15 sec	5 min 45 sec	4 min 54 sec
Verify SPI Flash	1 hr 57 min 38 sec	1 hr 50 min 45 sec	16 min 33 sec	7 min 53 sec	7 min 4 sec
Read SPI Flash	2 hrs 02 min 43 sec	1 hr 55 min 30 sec	16 min 12 sec	7 min 36 sec	6 min 47 sec
Erase SPI Flash	18 sec	18 sec	1 min 52 sec	1 min 50 sec	1 min 50 sec

Important:

- [1] For more information about different SPI Flash device support, see FlashPro6 SPI Flash Support.
- [2] FlashPro6 has longer programming times for SPI Flash devices, when compared to FlashPro5. However, readback and verification times are significantly shorter. Programming time for FlashPro6 will be improved in future releases.
- [3] To program the device at a high TCK frequency, take appropriate steps to ensure signal integrity of the JTAG signals.
- [4] SPI Flash Programming time may vary from device to device, even though the part number is the same. This is due to variation in die.

15. Appendix B: Regulatory and Compliance Information (Ask a Question)

EU Declaration of Conformity:

This product complies with Directive 2011/65/EU of the European Parliament and of the Council of 8 June 2011 on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

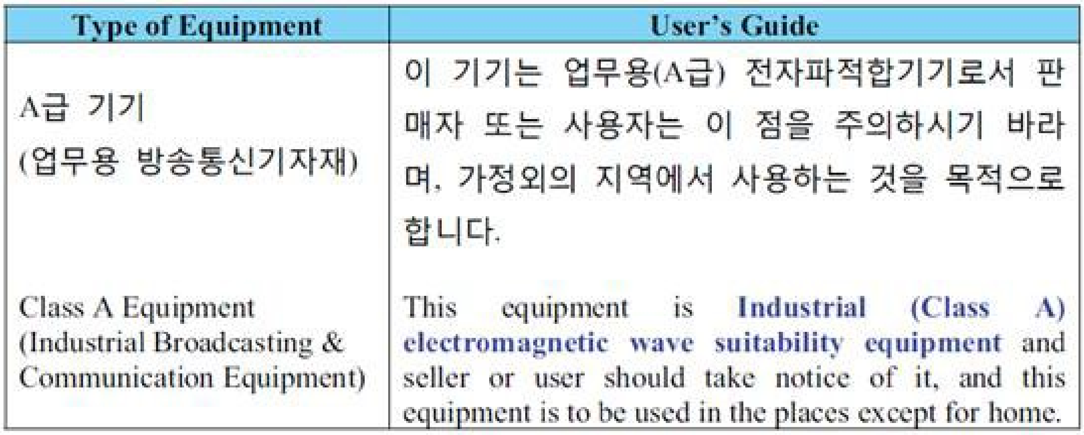
To view the Declaration of Conformity in English, see EU Declaration of Conformity.

For Non-English, see EU Declaration of Conformity (EU Languages)

Markings:

CE This product complies with 2004/108/EC, Electromagnetic Compatibility (EMC) Directive

For Korea:



Type of Equipment	User's Guide
A급 기기 (업무용 방송통신기자재)	이 기기는 업무용(A급) 전자파적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며, 가정외의 지역에서 사용하는 것을 목적으로 합니다.
Class A Equipment (Industrial Broadcasting & Communication Equipment)	This equipment is **Industrial (Class A) electromagnetic wave suitability equipment** and seller or user should take notice of it, and this equipment is to be used in the places except for home.

16. Revision History (Ask a Question)

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

Revision	Date	Description
R	12/2025	The following changes are made in this revision: • Revised the exit codes in sections Exit Codes Applicable to All Families, PolarFire SoC Exit Codes, PolarFire Exit Codes, RTG4 Exit Codes, and SmartFusion 2 and IGLOO 2 Exit Codes.
Q	05/2025	The following changes are made in this revision: • Updated the FlashPro6 topic to address some truncated text issues.
P	09/2024	The following changes are made in this revision: • Updated section Selecting and Running an Action.
N	08/2024	The following changes are made in this revision: • Updated section PolarFire Exit Codes. • Updated section RTG4 Exit Codes. • Updated section SmartFusion 2 and IGLOO 2 Exit Codes. • Updated section Appendix A: Sample Programming and SmartDebug Times Using FlashPro5 and FlashPro6.
M	02/2024	The following changes are made in this revision: • Added section PolarFire SoC Exit Codes. • Updated section PolarFire Exit Codes. • Updated section Appendix A: Sample Programming and SmartDebug Times Using FlashPro5 and FlashPro6.
L	08/2023	The following changes are made in this revision: • Updated section Creating a Job Project with information to create a job file without connecting hardware. • Updated section Opening a Job Project with information to update device(s). • Updated section Chain Programming Tutorial with screenshot of Figure 3-8. • Updated section Programming Connectivity and Interface. • Update section PolarFire Exit Codes to include information about two new error codes: 0x8040 and 0x805D.
K	04/2023	The following changes are made in this revision: • Updated section FlashPro6. • Updated section FlashPro5. • Updated section FlashPro4.
J	12/2022	The following changes are made in this revision: • Updated section SmartFusion 2 and IGLOO 2 Exit Codes.
H	12/2022	The following changes are made in this revision: • Updated section SmartFusion 2 and IGLOO 2 Exit Codes.
G	08/2022	The following changes are made in this revision: • Updated section Selecting and Running an Action. • Updated section FlashPro5/4/3/3X Programmer Settings.

Revision History (continued)

Revision	Date	Description
F	04/2022	The following changes are made in this revision: • Added section Exit Codes Applicable to All Families. • Updated section PolarFire Exit Codes. • Updated section SmartFusion 2 and IGLOO 2 Exit Codes.
E	12/2021	The following changes are made in this revision: • Added section Updating FlashPro6 Internal Design. • Added an exit code in section SmartFusion 2 and IGLOO 2 Exit Codes.
D	08/2021	The following changes are made in this revision: • Updated section Programmer Settings.
C	04/2021	Editorial updates only. No technical content updates.
B	12/2020	The following changes are made in this revision: • Added Programming Connectivity and Interface section. • Added Construct Automatically option to create new job project in developer mode, section Creating a Job Project. • Updated the following sections for minor edits: FlashPro Express Tools Menu, Programmer Settings, FlashPro Express Project Menu.
A	11/2020	Initial Revision.

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

ISBN: 979-8-3371-2491-9

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.