BPI Configuration Flash Design Guide

797 North Grove Rd, Suite 101
Richardson, TX 75081
Phone: (972) 671-9570
www.redrapids.com

Red Rapids reserves the right to alter product specifications or discontinue any product without notice. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment. This product is not designed, authorized, or warranted for use in a life-support system or other critical application.

All trademark and registered trademarks are the property of their respective owners.

Table of Contents

List of Figures

1.0 Introduction

A byte peripheral interface (BPI) flash is used to store the FPGA bitstream that will be loaded automatically at power-up. This manual is directed at the FPGA developer that will load the FPGA or configuration flash with an application specific bitstream.

The latest product documentation and software is available for download from the Red Rapids website (www.redrapids.com).

1.1 Conventions

This manual uses the following conventions:

- Hexadecimal numbers are prefixed by "0x" (e.g. 0x00058C).
- *Italic* font is used for descriptive text found on GUI displays.
- Blue font is used for GUI button and menu option names.
- Active low signals are followed by _B, For example, RST_B.

☞	Text in this format highlights useful or important information.

!	Text shown in this format is a warning. It describes a situation that could potentially damage your equipment. Please read each warning carefully.

The following are some of the acronyms used in this manual.

- **BPI** Byte Peripheral Interface
- **BIT** File extension for FPGA bitstreams
- **MCS** File extension for flash PROM bitstreams
- **XDC** File extension for Xilinx design constraints

1.2 Revision History

Version	Date	Description
R00	5/8/2014	Initial release.

2.0 Hardware Architecture

Figure 2-1 illustrates the connection between the FPGA and the BPI flash component. Maximum configuration speed is achieved using a 16-bit data interface with the synchronous read mode feature of the flash. A 100 MHz crystal oscillator is connected to the external master configuration clock (EMCCLK) pin of the FPGA. This clock source drives the internal configuration engine of the FPGA to load the bitstream at power-up.

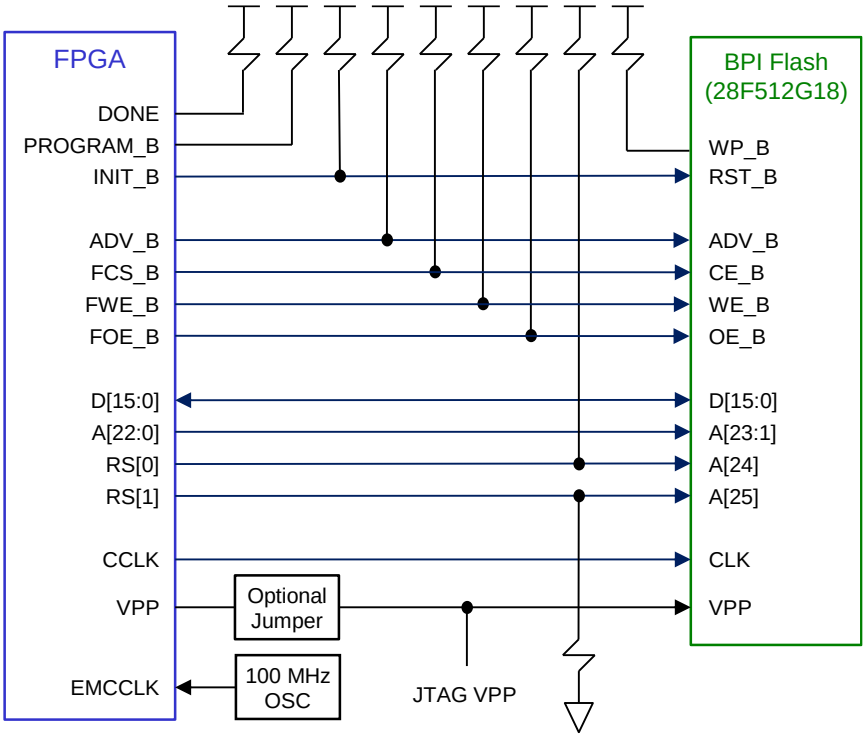

Figure 2-1 BPI Configuration Hardware Interface

The write protect (WP) pin of the flash is pulled up to disable block lock down. The program and erase operations of the flash can be enabled or disabled through the programming voltage (VPP) pin. The VPP input is always driven high when the JTAG programming cable is physically connected to the board. This allows the flash to be programmed indirectly through the FPGA using a Xilinx JTAG programming cable and the iMPACT™ configuration tool. An optional hardware jumper also allows the FPGA direct control of the programming voltage. The jumper can be eliminated as a product build option if a customer wants to disable flash programming from the FPGA application logic for security purposes.

The 512 Mbit flash is large enough to store multiple bitstreams. The exact number depends on the size of the FPGA. The Xilinx multiboot and fallback features are supported through the FPGA RS[1:0] pin connections to the upper address bits of the flash. Refer to the Xilinx Configuration User Guide for further details.

2.1 Device Configuration Bitstream Settings

There are two device configuration bitstream settings that must be changed from the default values to accommodate the BPI flash.

Setting	Default	BPI Value
BITSTREAM.CONFIG.BPI_SYNC_MODE	Disable	Type1
BITSTREAM.CONFIG.EXTMASTERCCLK_EN	Disable	div-1

These settings are included in the Xilinx Design Constraints (XDC) file supplied with each Red Rapids product. Additional settings must be changed to support multiboot and fallback configurations. Refer to the Xilinx Configuration User Guide for further details.

2.2 MCS File Generation

Before the flash can be programmed, an MCS file must be created from the BIT file produced by Vivado™. Launch the Xilinx iMPACT™ tool and double click on Create PROM File in the *iMPACT Flows* pane on the left side of the window. The *PROM File Formatter* window should display the three steps necessary to create the file.

Step 1. Select Storage Target

Single click Configure Single FPGA under the *Storage Device Type* pane as shown in Figure 2-2. Click the green arrow between Step 1 and Step 2 to proceed.

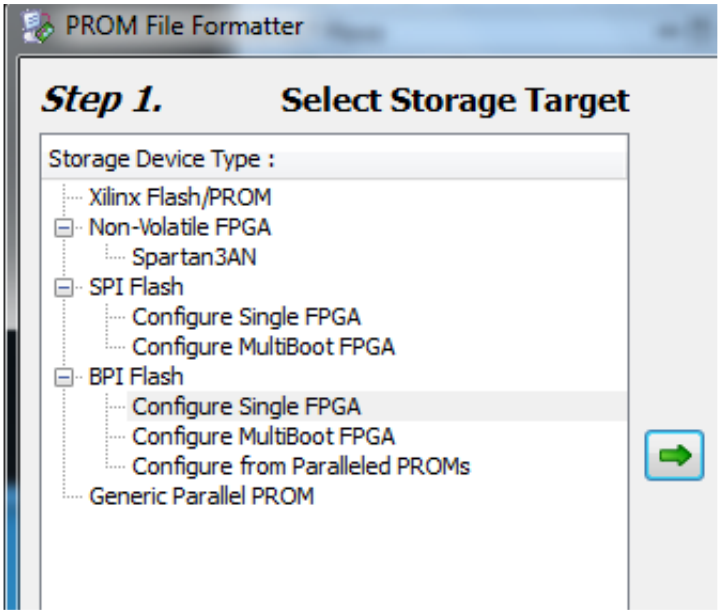

Figure 2-2 Select Storage Target

Step 2. Add Storage Device(s)

Change the *Target FPGA* to Kintex7 in the drop down menu under Step 2 and change the *Storage Device* to 64M as shown in Figure 2-3. Click the Add Storage Device button to make it appear in the device list pane. Click the green arrow between Step 1 and Step 2 to proceed.

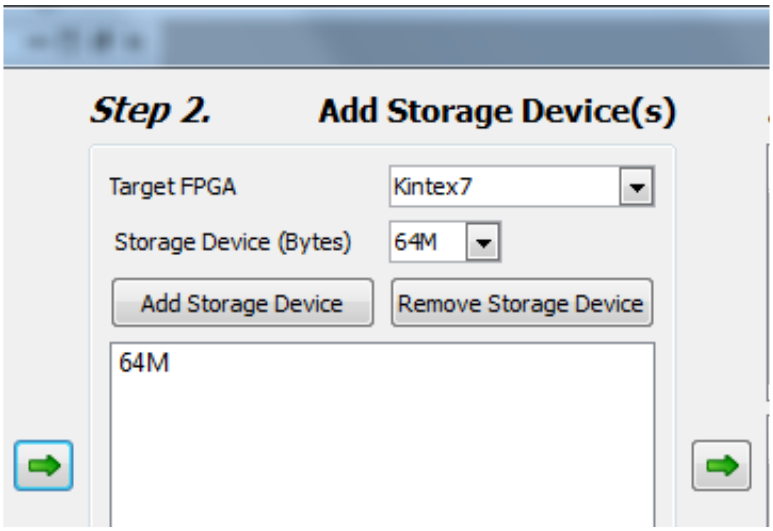

Figure 2-3 Add Storage Devices

Step 3. Enter Data

Enter your desired *Output File Name* and *Output File Location* in the *General File Detail* dialog box under Step 3 shown in Figure 2-4. Change the *Data Width* to x16 in the drop down menu under the *Flash/PROM File Property* pane. Click the OK button to proceed.

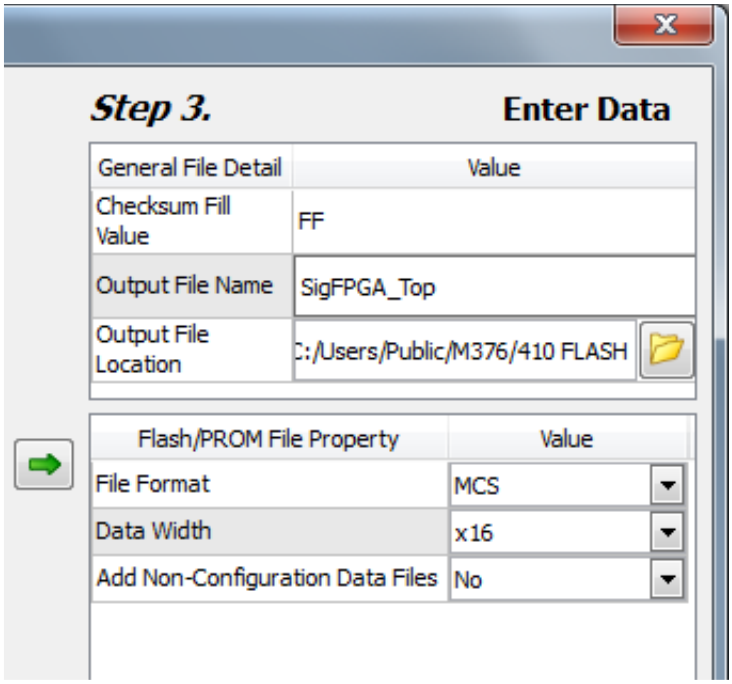

Figure 2-4 Enter Data

The window shown in Figure 2-5 should appear when the selected BIT file has been added to the flash address space. Double click Generate File… in the *iMPACT™ Processes* pane to create the MCS file in the specified directory.

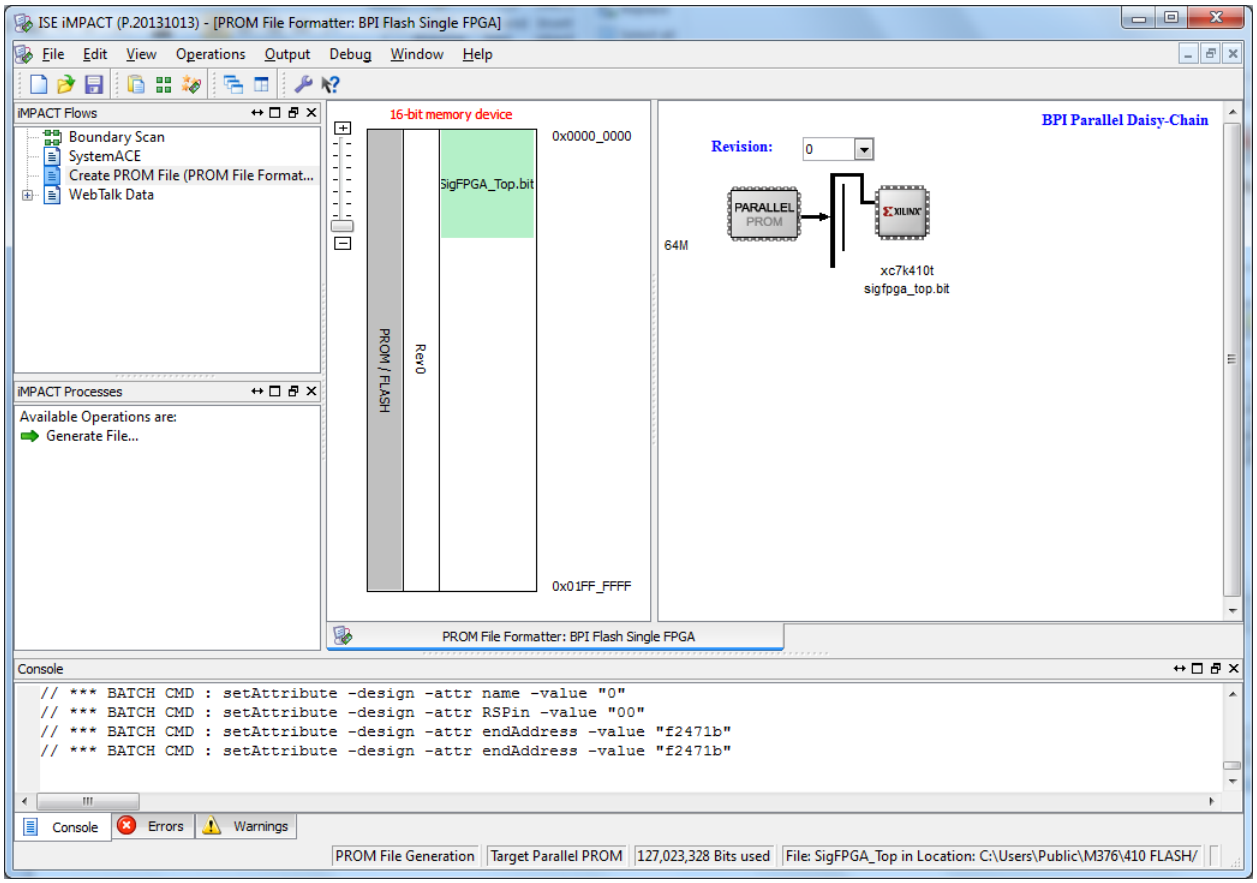

Figure 2-5 PROM File Formatter

2.1 JTAG Programming With Xilinx iMPACT™

The Xilinx iMPACT™ tool is used to directly program the FPGA from a BIT file or indirectly program the BPI flash from an MCS file. A Xilinx programming cable is needed to connect the host computer to the JTAG connector on the board. The Xilinx programmer plugs into a USB port on the host computer and a cable provided by Red Rapids connects the programmer to the JTAG port on the product.

Launch the Xilinx iMPACT™ tool with the Xilinx programmer connected to the board. Double click on Boundary Scan in the *iMPACT Flows* pane on the left side of the window. Right click on the text that reads Right click to Add Device or Initialize JTAG Chain and select the Initialize Chain option from the contextual menu. This will cause the JTAG cable to scan the chain and identify the FPGA device as shown in Figure 2-6.

Continue the process by assigning a BIT file to the FPGA and an MCS file to the BPI flash as prompted by the iMPACT™ tool. The *Select Attached SPI/BPI* window shown in Figure 2-7 will request hardware specific information. Select BPI PROM from the drop down menu below *Select the PROM Attached to the FPGA*. Select 28F512G18F from the drop down menu for the BPI PROM part number. Select 16 from the *Data Width* drop down menu and select NOT USED for *Select RS[1:0]b Pin Address Bits* to load a single configuration. Select 25:24 for multiboot or fallback loads. There is no need to change the default device programming properties. Programming of either the FPGA or BPI flash can proceed by right clicking on the device icons and selecting Program from the contextual menu.

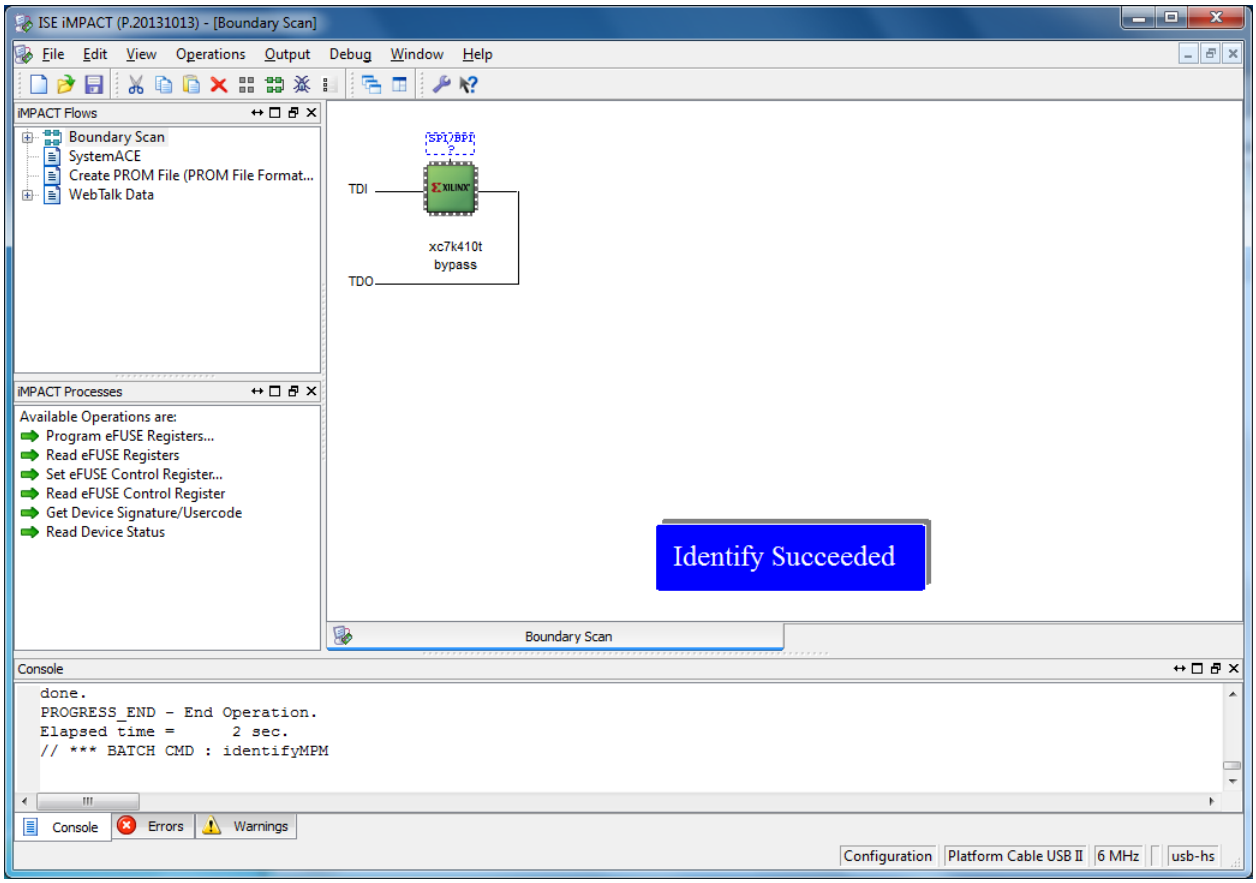

Figure 2-6 Initialize JTAG Chain

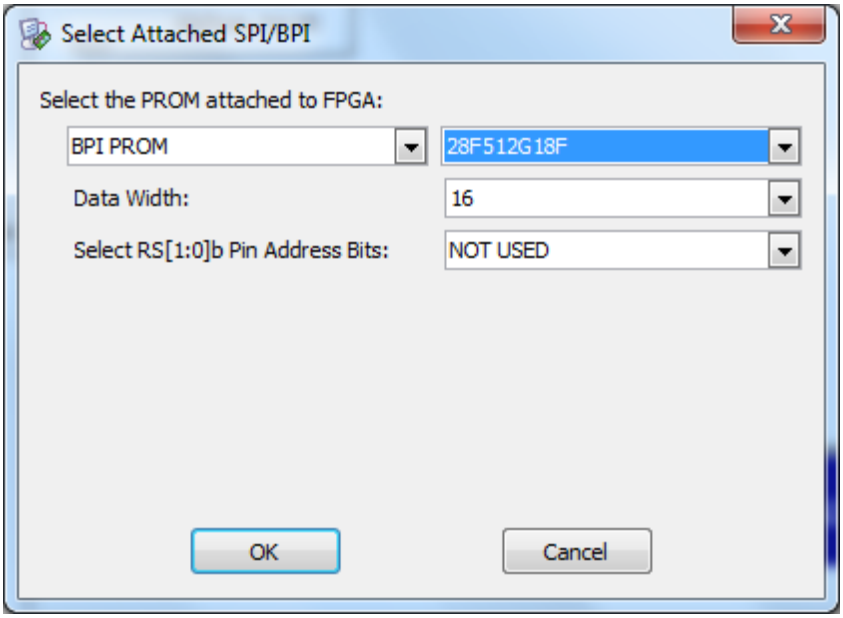

Figure 2-7 Select Attached SPI/BPI