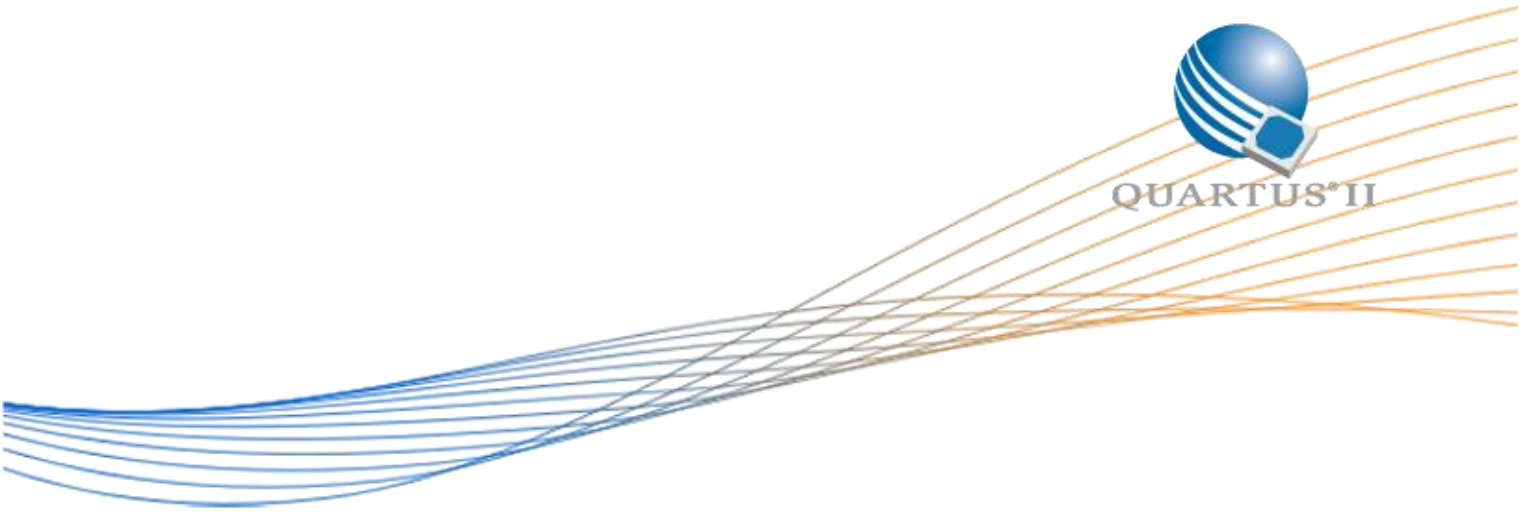
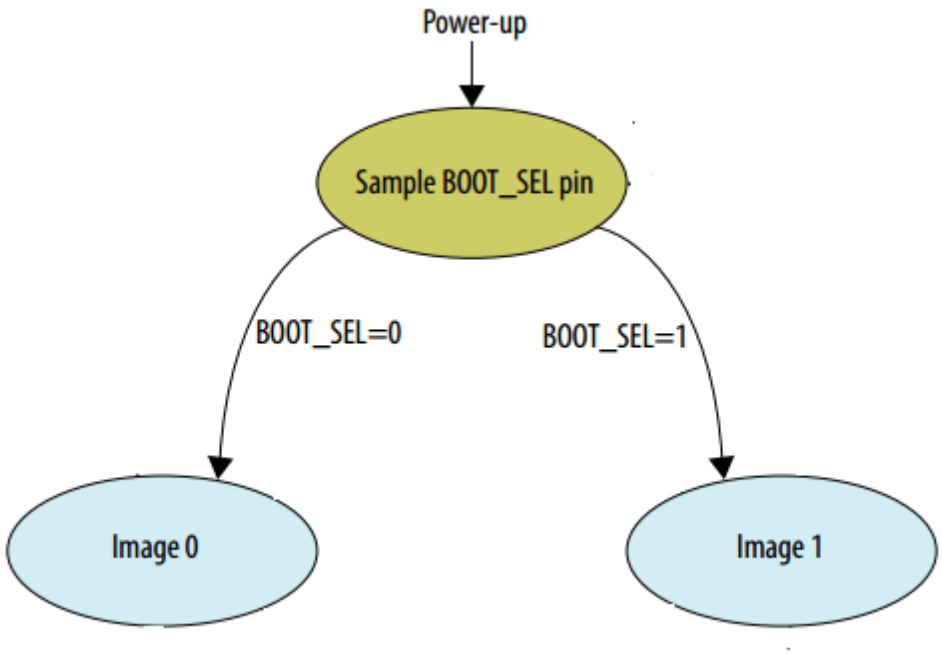



# Using Dual Boot option in MAX 10 FPGA Evaluation Kit

©2014 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at www.altera.com/common/legal.html. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

This example explains how to use the Dual Boot Feature in MAX 10 FPGA Evaluation Kit.

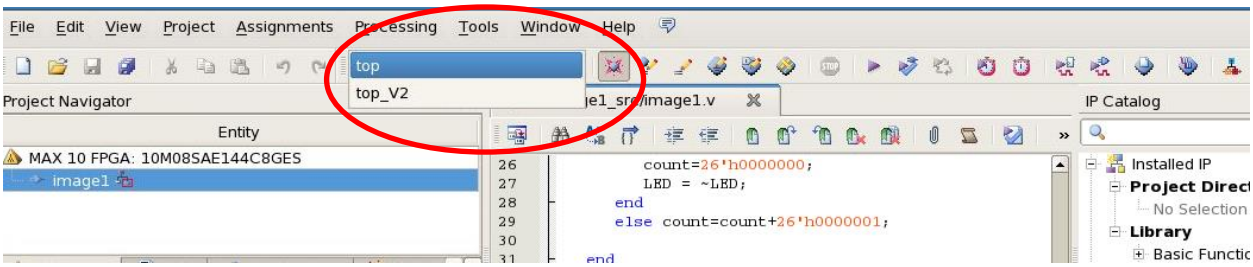



Refer to the MAX 10 FPGA Configuration User Guide for more information on the internal configuration and remote system upgrade features.

The BOOT_SEL pin on the MAX10 Evaluation Kit is Switch 6 in SW3. Based on whether the switch is in OFF position or ON position Image 0 or Image 1 will be selected.

Image0 – Blinks LED 1 and LED 2 alternatively.
Image1 – Blinks LED 3 and LED 4 alternatively.
Clk – 50MHz Oscillator in the Board
Reset – Switch 1 in SW3

In this example , we create two revisions within the same project , <revision_name>.qsf with image0_src and <revision_name>_V2.qsf with image1_src. When you open the project , you can select which revision you want to work with by selecting the corresponding revision name in drop down list box in the Standard toolbar at the top of Quartus II.




Note that selecting <revision_name> will take image 0 as the top level entity and selecting <revision_name>_V2 will take image 1 as the top level entity.

In this example, Steps 1, 2 and 3 below are already completed. We have the dual_image_boot.pof created for you in the folder , which was created from the .sof files of image0 and image1. You can directly download this dual_image_boot.pof using the Quartus II programmer to get the example working on your MAX 10 Eval Board.

If you want to get the individual .sof files, simply compile both the revisions.

Steps 1 and 2 below tells the changes to the settings that have to be completed if you want to use your own source files for Dual Boot and step 3 tells how to create the combined .pof from the two .sof files . Step 4 explains how to program the MAX 10 FPGA Board using Quartus II Programmer.

If you want to create your own source .sof files for the dual boot in MAX 10 FPGA , use the following steps:

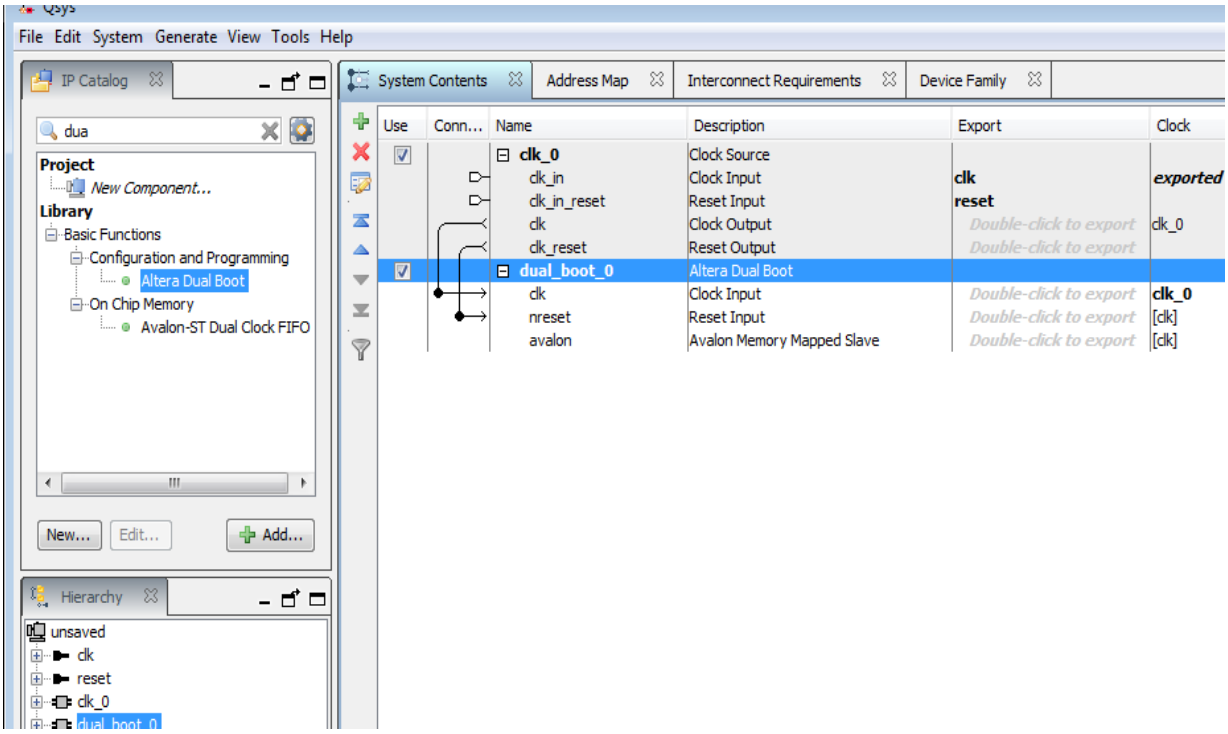
**1. Adding Dual Boot IP into the project**

(i) Open Qsys and add the Dual Boot IP. Refer to the MAX 10 FPGA Configuration User Guide for more info on the Dual Boot IP.

(ii) Double click on the clk and select the frequency as 50 MHz. You can use the 50 MHz clock source on the Evaluation Board.

(iii) Connect the clock and reset.

(iv) Save and Generate HDL.



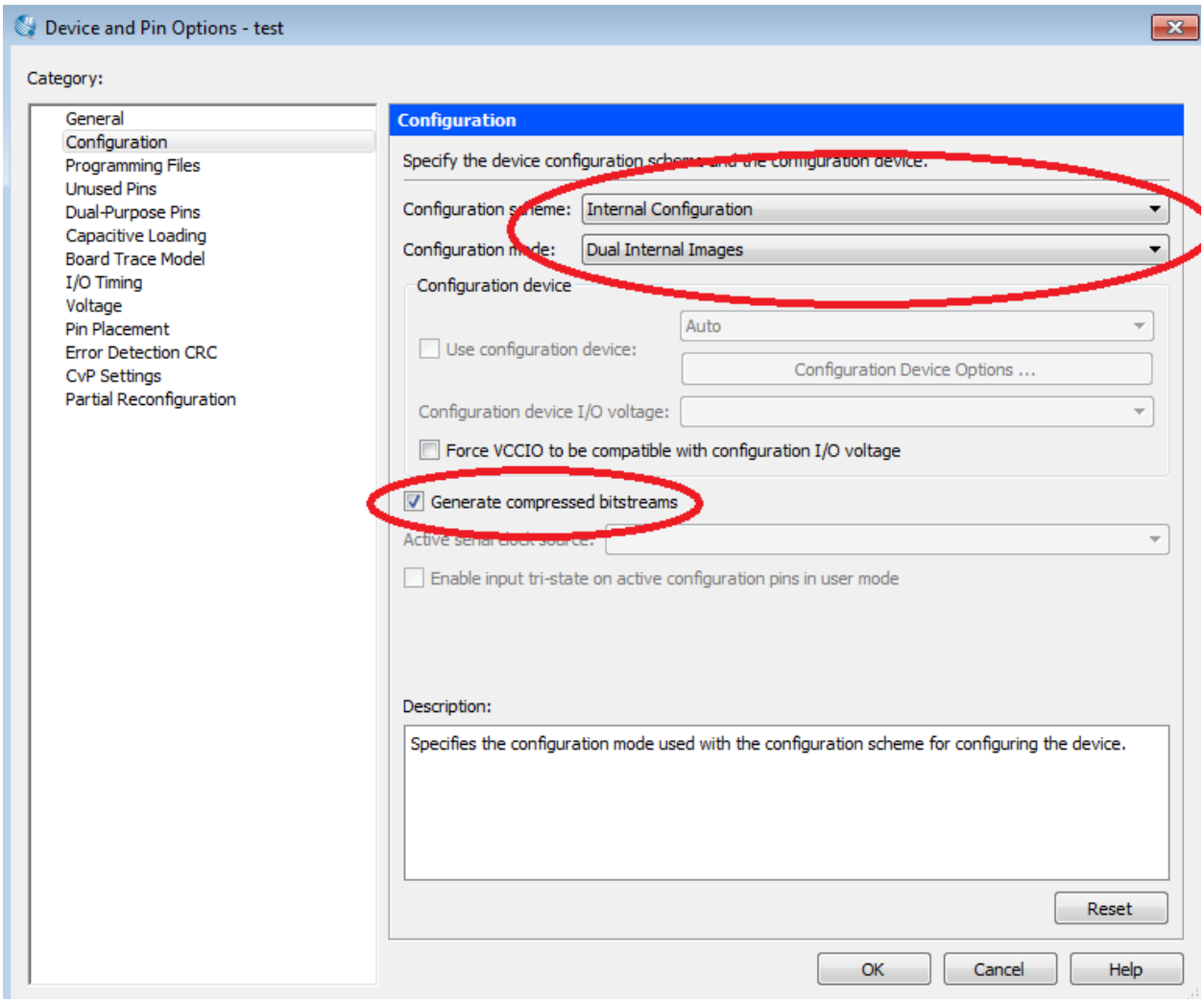

(v) Once the IP is generated, instantiate the IP is your top level RTL source code and connect the clock and reset correspondingly.
(Refer to the source codes image0.v and image1.v on how to instantiate the IP)

## 2. Selecting Internal Configuration Scheme

Max 10 devices uses Internal configuration to support Dual Boot feature. For all MAX 10 devices except the 10M02 device, there are a total of 5 different modes that can be selected when using Internal Configuration. The internal configuration scheme needs to be selected before design compilation.

To select the configuration mode, follow these steps:

(i) Open the Quartus II software and load a project using the MAX 10 device family.
(ii) On the **Assignments** menu, click **Device**. The Device page appears.
(iii) Click **Device and Pin Options**.
(iv) In the Device and Pin Options dialog box, click the **Configuration tab** on the left.
(v) In the Configuration Scheme list, select **Internal Configuration**.
(vi) In the Configuration Mode list, select **Dual Internal Images.**
(vii) Turn on **Generate compressed bitstreams** if needed.
(viii) Click OK.




Steps 1 and 2 (Adding Dual Boot IP and setting internal configuration scheme) should be completed for both the images before design compilation. Compiling each of the images gives the corresponding .sof files.

## 3. Generating the .pof file with ICB (Initialization Configuration Bit) Settings

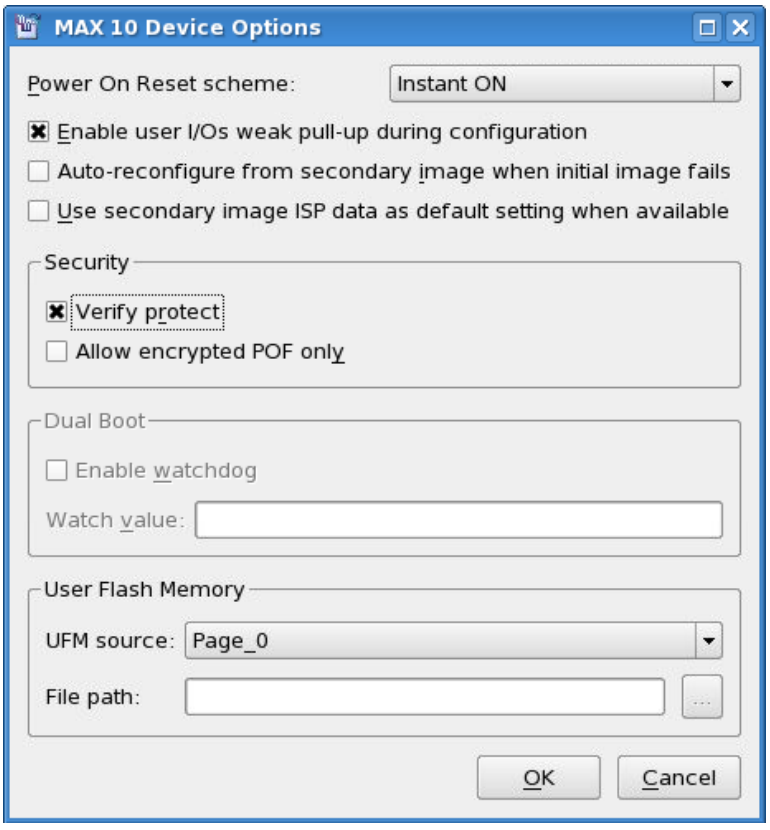
Then .pof file has to be generated with ICB Settings. To generate a .pof file from a .sof file for internal configuration, follow these steps:

(i) On the **File** menu, click **Convert Programming Files**.
(ii) Under Output programming file, select **Programmer Object File (.pof)** in the **Programming file type** list.
(iii) In the **Mode** list, select **Internal Configuration**.
(iv) To set the ICB settings, click **Option/Boot** Info button. An ICB setting dialog box will appear.

Set the following in the ICB setting dialog box:

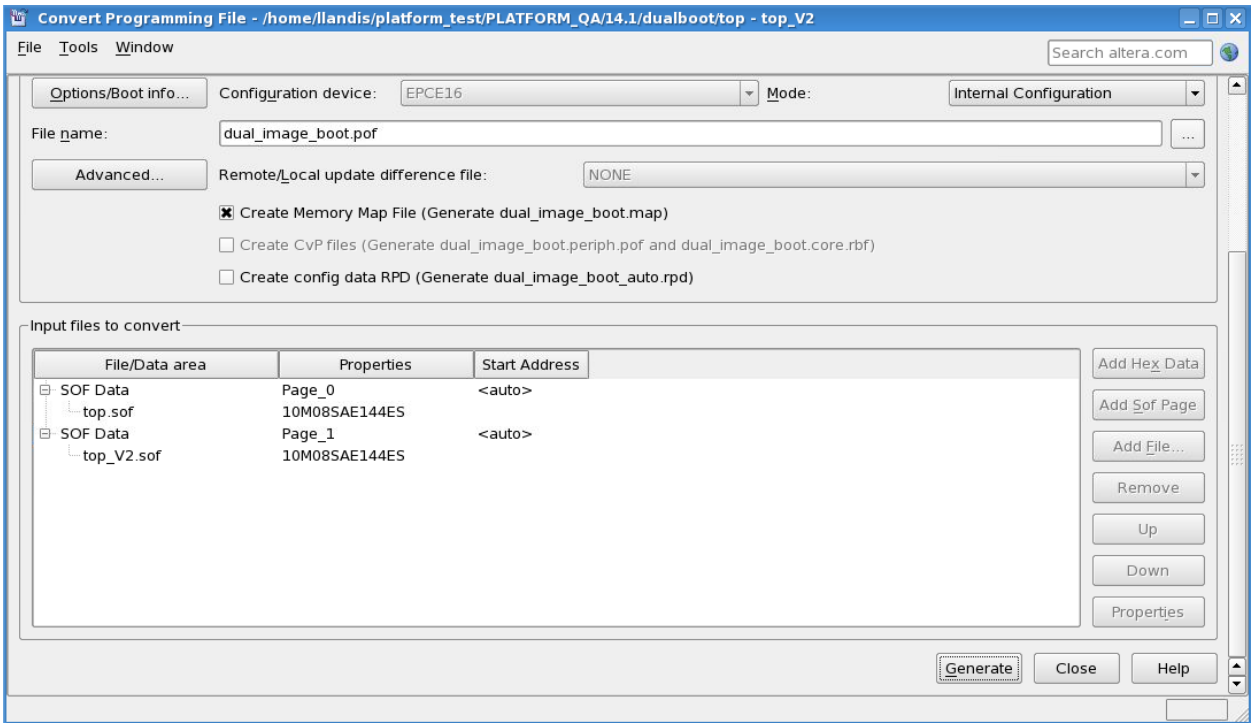
- Power on Reset Scheme: **Instant On**
- Check - Enable user IOs week pull up during configuration check box.
- Check - Enable the JTAG Security check box.
- Check - Verify Protect check box.
- In this example, **No Watchdog is used**, since the configuration reverts once the Watchdog times out.

**NOTE: It is always recommended to enable the Watch Dog Timer.**



(v) In the File name box, specify the file name for the programming file you want to create.

(vi) To generate a Memory Map File (.map), turn on Create Memory Map File (Auto generate output_file.map). In the .map file, the tool will not only will show the address of the CFM and UFM, but also will contain the information of the ICB setting that the user sets through the Option/Boot Info dialog box.




(vii) After setting all the settings as show in the figure above, click Generate to generate the dual image .pof file that is loaded into flash.

**User Watchdog Timer**

The user watchdog timer prevents a faulty application configuration from stalling the device indefinitely. You can use the timer to detect functional errors when an application configuration is successfully loaded into the device.

The counter is 29 bits wide and has a maximum count value of 229. When specifying the user watchdog timer value, specify only the most significant 12 bits. The granularity of the timer setting is 217 cycles. The cycle time is based on the frequency of the user watchdog timer internal oscillator. Depending on the counter and the internal oscillator of the device, you can set the cycle time from 17ms to 243s.

The timer begins counting as soon as the application configuration enters user mode. When the timer expires, the remote system upgrade circuitry generates a time-out signal, updates the status register, and triggers the loading of the revert configuration image. To reset the timer, pulse the RU_NRSTIMER for a minimum of 250 ns.

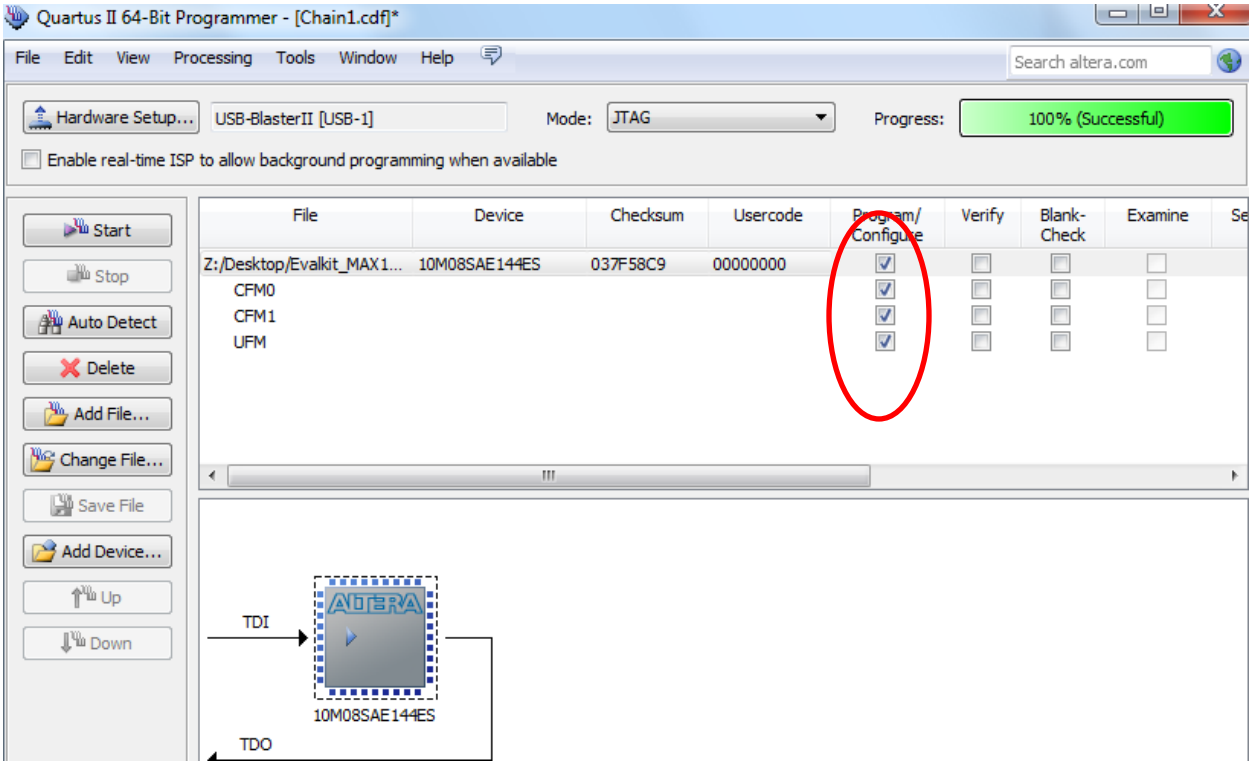
If the watchdog timer is enabled, this setting will apply to all images, all images should contain the soft logic configuration to reset the timer. Application Configuration will reset the control block registers.

Refer to the MAX 10 FPGA Evaluation User Guide and MAX 10 FPGA Configuration User Guide for more information on Watchdog and configuration modes.

## 4. Programming the Internal Flash Memory

After generating the .pof file, the Quartus II Programmer can be used to program the internal flash memory through the JTAG connection.

(i) Select the Add file and add the generated .pof file in the Quartus II programmer
(ii) Select Program/Configure checkboxes.



(iii) Click start

 Power off the device before changing the BOOT_SEL pin from ON to OFF and vice versa to see the response of the LED pins based on the two different revisions of the project. Note that either a power cycle or hitting the NCONFIG pin switch SW2 is required for the MAX 10 FPGA to reboot from the selected image.

## Acknowledgment

This design is based on Dual Boot Demo for BeMicro MAX 10 FPGA Evaluation Kit produced by Arrow Electronics.

## Document Revision History

| Date | Version | Changes |
|---|---|---|
| October 2014 | 2014.10.15 | Initial release |