KC705 MultiBoot Design

July 2012
Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>07/25/12</td>
<td>3.0</td>
<td>Recompiled for 14.2. Added AR50886.</td>
</tr>
<tr>
<td>05/08/12</td>
<td>2.0</td>
<td>Recompiled for 14.1.</td>
</tr>
<tr>
<td>01/18/12</td>
<td>1.0</td>
<td>Initial version for 13.4.</td>
</tr>
</tbody>
</table>

© Copyright 2012 Xilinx, Inc. All Rights Reserved.
XILINX, the Xilinx logo, the Brand Window and other designated brands included herein are trademarks of Xilinx, Inc. All other trademarks are the property of their respective owners.

NOTICE OF DISCLAIMER: The information disclosed to you hereunder (the “Information”) is provided “AS-IS” with no warranty of any kind, express or implied. Xilinx does not assume any liability arising from your use of the Information. You are responsible for obtaining any rights you may require for your use of this Information. Xilinx reserves the right to make changes, at any time, to the Information without notice and at its sole discretion. Xilinx assumes no obligation to correct any errors contained in the Information or to advise you of any corrections or updates. Xilinx expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE INFORMATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS.
Overview

» Kintex-7 MultiBoot Capability
» Overview of 7 Series Fallback MultiBoot
» Xilinx KC705 Board
» Software Requirements
» KC705 Setup
» Compiling the MultiBoot Design
» Run MultiBoot Design
» Kintex-7 MultiBoot Details
» References

Note: This presentation applies to the KC705
Kintex-7 MultiBoot Capability

What is MultiBoot?
- The MultiBoot feature allows the FPGA application to load two or more FPGA bitstreams under the control of the FPGA application. The FPGA application triggers a MultiBoot operation, causing the FPGA to reconfigure from a different configuration bitstream. After a MultiBoot operation is triggered, the FPGA restarts its configuration process as usual.
- More details can be found in UG470

MultiBoot Capability
- FPGA application controlled configuration
- Bitstream selection of multiple applications

Safe Update
- Golden bitstream
- Upgradeable bitstream
- Failure recovery
  - Possible Triggers (CRC error, IDCODE error, WDT timeout)

Note: Presentation applies to the KC705
Kintex-7 MultiBoot Capability

Overview of 7 Series Fallback MultiBoot

– The 7 series FPGAs MultiBoot and fallback features support updating systems in the field. Bitstream images can be upgraded dynamically in the field. The FPGA MultiBoot feature enables switching between images on the fly. When an error is detected during the MultiBoot configuration process, the FPGA can trigger a fallback feature that ensures a known good design can be loaded into the device.

– The MultiBoot feature allows the FPGA application to load two or more FPGA bitstreams under the control of the FPGA application. The FPGA application triggers a MultiBoot operation, causing the FPGA to reconfigure from a different configuration bitstream. After a MultiBoot operation is triggered, the FPGA restarts its configuration process as usual.

Reference Design Source and Application

– rdf0104.zip

– Available through http://www.xilinx.com/kc705

Note: Presentation applies to the KC705
KC705 MultiBoot Design Description

Description

- Design consists of three bitstreams, golden_iprog_spi.bit, multiboot.bit, and corrupted.bit loaded in the PROM at 0x0, 0x400000, and 0x800000
- The golden_iprog_spi.bit has software to allow a reboot to the second or third bitstreams with certain switch settings
- The multiboot.bit is compiled with “-g ConfigFallback:Enable” bitgen option
- The corrupted.bit is the multiboot.bit with one byte changed to force a CRC error when loaded

This diagram shows the general process

- Since the corrupted.bit doesn’t pass configuration, the Golden Image is booted

Note: Presentation applies to the KC705
Upgrade Example

- **MultiBoot**: Process by which the FPGA selectively reprograms and reloads its bitstream from an attached external memory.

- **Safe update**: Field updating bitstream storage with a new bitstream in such a manner to prevent any failure due to a failure in the update process. This is accomplished with the enhanced MultiBoot available in Kintex-7.

Can a multi-boot system be implemented without safe update design considerations?
Yes – If there are no potential for disruptions during flash loading

How is a system upgraded?
1. New multi-boot image is created
2. System setup to receive the new image
3. User application erases section of Flash
4. The new image is delivered into the system’s Flash
5. User application resets system

**Note:** Presentation applies to the KC705
Xilinx KC705 Board

Note: Presentation applies to the KC705
ISE Software Requirement

- Xilinx ISE 14.2 software
  - Apply AR50886
Connect a USB Type-A to Micro-B cable to the USB JTAG (Digilent) connector on the KC705 board

- Connect this cable to your PC
- Power on the KC705 board
Hardware Setup

» Set S13 to 00001 (1 = on, Position 1 → Position 5)
  - This enables Master SPI configuration from the QSPI Flash
    • Flash A25, A24 = 00
    • FPGA mode pins M[2:0] = 001

Note: Presentation applies to the KC705
Hardware Setup

Set S11 to 0000 (1 = on, Position 1 → Position 4)
- This sets the MultiBoot functions to off
Program SPI MultiBoot Design

- Unzip the KC705 Multiboot Design Files (14.2 CES)
  - Available through http://www.xilinx.com/kc705
Program SPI MultiBoot Design

Open an ISE Design Suite Command Prompt and type:

```
cd C:\kc705_multiboot\SPI_Flash
impact -batch kc705_program_spi.cmd
```

Note: Takes about ten minutes
Run MultiBoot Design

The initial design, “golden_iprog_spi.bit”, shows a cycling LED pattern on the GPIO LEDs
Run MultiBoot Design

Run iMPACT:

impact

Note: Presentation applies to the KC705
Run MultiBoot Design

Select

- Create a new project
- Configure devices using Boundary-Scan (JTAG)

Note: Presentation applies to the KC705
Run MultiBoot Design

- Select the FPGA

- Select the menu item Debug → Read Device Status

Note: Presentation applies to the KC705
Run MultiBoot Design

- Click console tab and then right click on the console panel and select "Toggle Slide Out"

Note: Presentation applies to the KC705
Run MultiBoot Design

- Right click on the Console panel and select Toggle Maximized

Note: Presentation applies to the KC705
Run MultiBoot Design

➤ Scroll to the “Reading bootsts register contents…” message
➤ Since this was a regular boot, only “VALID_0” is set

Note: Presentation applies to the KC705
Run MultiBoot Design

- Set S11 to 0001 (1 = on, Position 1 → Position 4)
  - Issues an IPROG command to re-configure from a good MultiBoot bitstream
  - Set it back to 0000 when the DONE LED goes out
Run MultiBoot Design

- The MultiBoot design, “multiboot.bit”, shows a rapid blinking LED pattern on the GPIO LEDs
Run MultiBoot Design

- Select the menu item Debug → Read Device Status
- Scroll down to the “Reading bootsts register contents…” message

Note: Presentation applies to the KC705
Run MultiBoot Design

The following status register messages are now set:

- IPROG_0
- VALID_1

Note: Presentation applies to the KC705
Run MultiBoot Design

» Press PROG to return to the golden bitstream
» Wait until you see the cycling pattern on the LEDs
Run MultiBoot Design

➢ Set S11 to 1000 (1 = on, Position 1 → Position 4)
  – Issues an IProg command to re-configure from a corrupted MultiBoot bitstream
  – Set it back to 0000 when the DONE LED goes out
Run MultiBoot Design

- The corrupted .bit generates a CRC error and cannot load the FPGA
- The FPGA falls back to the golden bitstream with a cycling LED pattern
Run MultiBoot Design

- Select the menu item Debug → Read Device Status
- Scroll down to the “Reading bootsts register contents…” message

Note: Presentation applies to the KC705
Run MultiBoot Design

The following status register messages are now seen:

- IPROG_0 is no longer active
- FALLBACK_0, IPROG_1, and CRC_ERROR_1 are now active

Note: Presentation applies to the KC705
Compile KC705 MultiBoot Design
Compile KC705 MultiBoot Design

➤ If desired, FPGA compile can be skipped by opening SDK directly:
  
  Start → All Programs → Xilinx Design Tools → ISE Design Suite 14.2 →
  EDK → Xilinx Software Development Kit

➤ Select the workspace: <design files>\SW\SDK

➤ Go to [SDK Software Compile](#)
Compile KC705 MultiBoot Design

- Open XPS project
  `<project directory>\system.xmp`

- Create the hardware design, `system.bit`, located in
  `<project directory>\implementation`
  - Click the Generate Bitstream button (1)
  - Or from the menu, select Hardware → Generate Bitstream

Note: Presentation applies to the KC705
Launch KC705 Design in SDK

Open SDK
- Click the Export Design button (1)
- Click Export & Launch SDK (2)

Note: Presentation applies to the KC705
Compile KC705 Software in SDK

SDK Software Compile - Build ELF files in SDK

- Project should build automatically; if not, select Project → Build All (1)
- Note: If by-passing the FPGA compile, the ELF files are already built; if desired, the ELF files can be re-built by selecting Clean… followed by Build All.
Generate KC705 PROM File

Start a ISE Design Suite Command Prompt and enter these commands:

```
    cd C:\kc705_multiboot\SPI_Flash
    make_prom.bat
```

Note: Presentation applies to the KC705
Generate KC705 PROM File

- The `make_prom.bat` generates compressed bitstreams with the ELF files in BRAM
- The MultiBoot bitstream uses the `-g ConfigFallBack:Enable` option
- "corrupted.bit" is a manually corrupted bitstream

Note: Presentation applies to the KC705
Kintex-7 MultiBoot Details
MultiBoot Register Setup & Command

Below are the values programmed into the ICAP via IPROG
- This table is from UG470, 7 Series FPGAs Configuration User Guide

<table>
<thead>
<tr>
<th>Configuration Data (hex)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFFFFFFF</td>
<td>Dummy Word</td>
</tr>
<tr>
<td>AA995566</td>
<td>Sync Word</td>
</tr>
<tr>
<td>20000000</td>
<td>Type 1 NO OP</td>
</tr>
<tr>
<td>30020001</td>
<td>Type 1 Write 1 Words to WBSTAR</td>
</tr>
<tr>
<td>00000000</td>
<td>Warm Boot Start Address (Load the Desired Address)</td>
</tr>
<tr>
<td>30008001</td>
<td>Type 1 Write 1 Words to CMD</td>
</tr>
<tr>
<td>00000000F</td>
<td>IPROG Command</td>
</tr>
<tr>
<td>20000000</td>
<td>Type 1 NO OP</td>
</tr>
</tbody>
</table>

Note: Presentation applies to the KC705
MultiBoot Register Setup & Command

- IPROG command issued via the software in xhwicap_low_level_example.c

```c
static u32 ReadId[HWICAP_EXAMPLE_BITSTREAM_LENGTH] = {
    0xFFFFFFFF, /* Dummy Word */
    0xAA95566,  /* Sync Word*/
    0x20000000, /* Type 1 NO OP */
    0x30020001, /* Write WBSTAR cmd */
    0x00400000, /* Addr in SPI Flash of Multiboot bitstream */
    0x30008001, /* Write CMD */
    0x0000000F, /* Write IPROG */
    0x00000000, /* Type 1 NO OP */
};
```
References

7 Series Configuration

– 7 Series FPGAs Configuration User Guide
Documentation
Documentation

Kintex-7

- Kintex-7 FPGA Family

KC705 Documentation

- Kintex-7 FPGA KC705 Evaluation Kit
- KC705 Getting Started Guide
- KC705 User Guide
- KC705 Reference Design User Guide