BIOS MCSDK 2.0

SRIO Boot Example

Applies to patch release based on 02.00.00.09
Publication Date: May 20, 2011

Texas Instruments, Incorporated
20450 Century Boulevard
Germantown, MD 20874 USA

Document License

Contributors to this document

Contents

SRIO Boot Example

1 Overview

The SRIO boot example is created to help customer quickly boot DSP through SRIO. The boot example includes two CCSv5 projects to build the HelloWorld boot image and the Host boot example application.

2 HelloWorld Boot Image

The HelloWorld project uses the BIOS MCSDK Platform Library to initialize the DDR and UART, it will print the “Hello World” information to the UART once it runs.

2.1 Procedure to build HelloWorld

- Import the project from tools\boot_loader\examples\srio\srioboot_helloworld\evmc66xxl in CCSv5
- Clean and re-build the project
- The srioboot_helloworld_evm66xxl.map and srioboot_helloworld_evm66xxl.out will be generated under tools\boot_loader\examples\srio\srioboot_helloworld\evmc66xxl\bin
- Run elf2HBin.bat under tools\boot_loader\examples\srio\srioboot_helloworld\evmc66xxl\bin, the batch file does the following file conversion:
    - Uses Code Gen utility hex6x.exe utility to convert the ELF format .out file to a ASCII hex format boot table file
    - Use Bttbl2Hfile.exe to convert the boot table file to a header text file and binary file, customers can use their own boot application to include the boot header or binary.
    - Use hfile2array.exe to convert the header text file to a header file with array of the image data

- Copy the converted header file to tools\boot_loader\examples\srio\srioboot_example\src so that the boot image can be linked into the Host boot example application

3 Host Boot Example Application

The Host Boot Example project uses the BIOS MCSDK Platform Library to initialize the DDR, it pushes the boot image data to the remote booting EVM via SRIO, and then writes the entry address of the boot image to the SRIO magic address via SRIO. The RBL running on the DSP polls the entry address and jumps to that address and starts to boot.

3.1 Procedure to build Host Boot Example

- Import the project from tools\boot_loader\examples\srio\srioboot_example\evmc66xxl in CCSv5
- Clean and re-build the project
- The srioboot_example_evm66xxl.map and srioboot_example_evm66xxl.out will be generated under tools\boot_loader\examples\srio\ srioboot_example\evmc66xxl\bin

4 Test Setup

Two breakout boards and two EVM's are required to do the test:

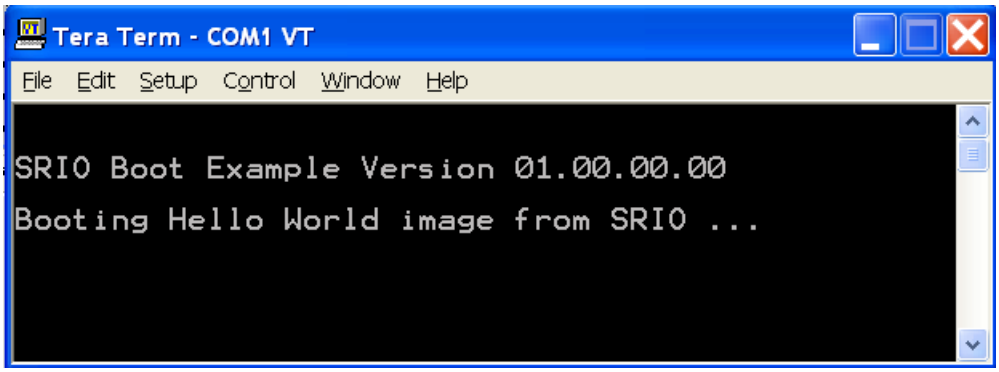
- Connect the SRIO Lane 0-3 from one breakout board to the other, be sure lane n is connected to lane n on each breakout board.
- Insert the two EVM's to the AMC slot on the two breakout boards
- Set the following boot dip switch settings on both EVM's

SW3(pin1, 2, 3, 4)	SW4(pin1, 2, 3, 4)	SW5(pin1, 2, 3, 4)	SW6(pin1, 2, 3, 4)
(off, off, on, on)	(on, on, on, off)	(on, off, on, off)	(off, on, on, on)

  This will set the board to boot from SRIO boot mode, with reference clock at 312.5 MHz, data rate at 3.125GBs, and lane setup 4-1x ports and DSP System PLL at 100 MHz. Please refer to the Technical Reference Manual (http://www.advantech.com/Support/TI-EVM/6678le_sd.aspx) for the boot dip switch settings.

- Connect the booting EVM to the PC's serial port using a RS-232 cable, connect the JTAG emulator on the Host EVM.
- Power on both EVM's

- Open a Hyper Terminal connection, set the baud rate to 115200 bps, data 8-bit, parity none, stop 1-bit, and flow control none
- Connect the Host EVM using CCSv5, load and run srioboot_example_evm66xxl.out
- The Hyper Terminal will display the following message from the booting EVM:



5 Limitations

- The boot example only supports a single core boot
- The boot example is pre-built and tested with little endian mode only

.