SPI x4 Configuration Demo on SP601 Board

Boris Qin

Agenda

- Why do we use SPI x4?
- How to use the Demo design?
- How to create the SPI x4 design?
- Measurements compared to the calculations

XILINX.

Why do we use SPI x4?

Configuration Time Definition and Calculation

- Definition

  In this application, Configuration Time is defined as the time between the rising edge of the last power rails of Vccint, Vccaux, and Vcco_2 and the rising edge of DONE pin.

- Calculation

Configuration time =

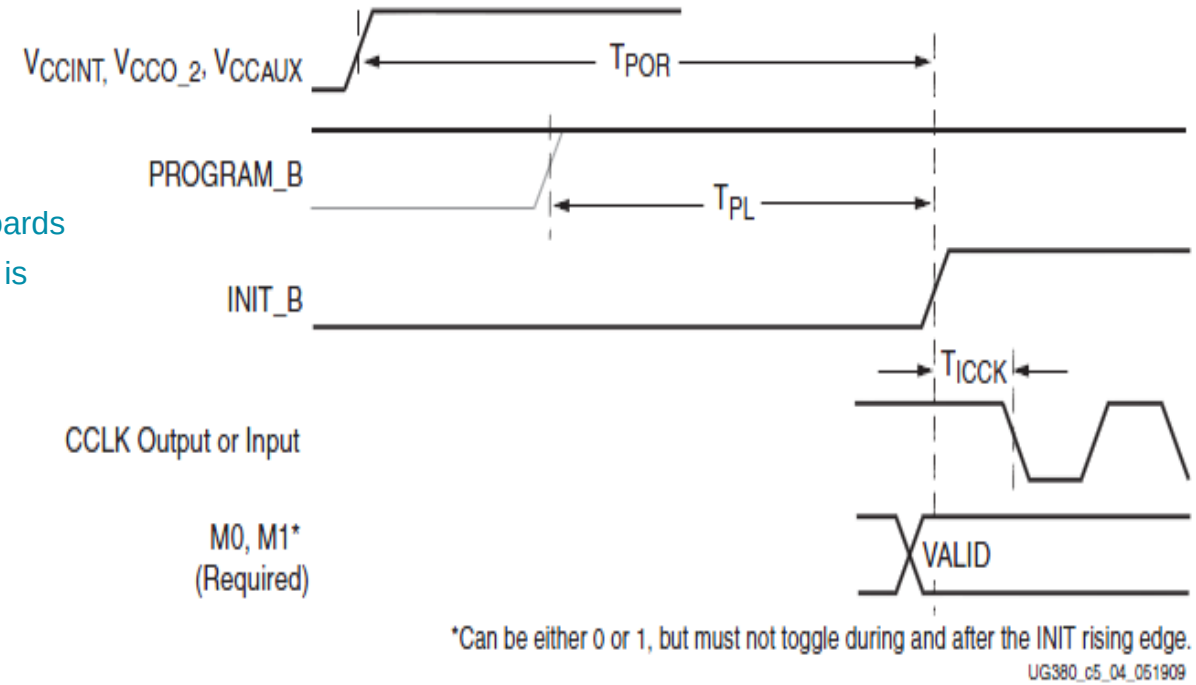
- Tpor + bitstream_size/(cclk_freq*buswidth)

Tpor = 10 ms (typical for -2 XC6VLX16)

Bitstream_size = 3,713,440

for XC6SLX16-CSG324 on SP601 Demo boards

- The few cycle for sending the SPI command is ignored
- Startup sequence time can also be ignored.



XILINX

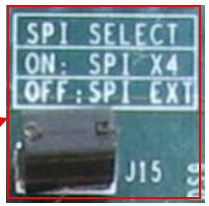
Agenda

- Why do we use SPI x4?
- How to use the Demo design?
- How to create the SPI x4 design?
- Measurements compared to the calculations

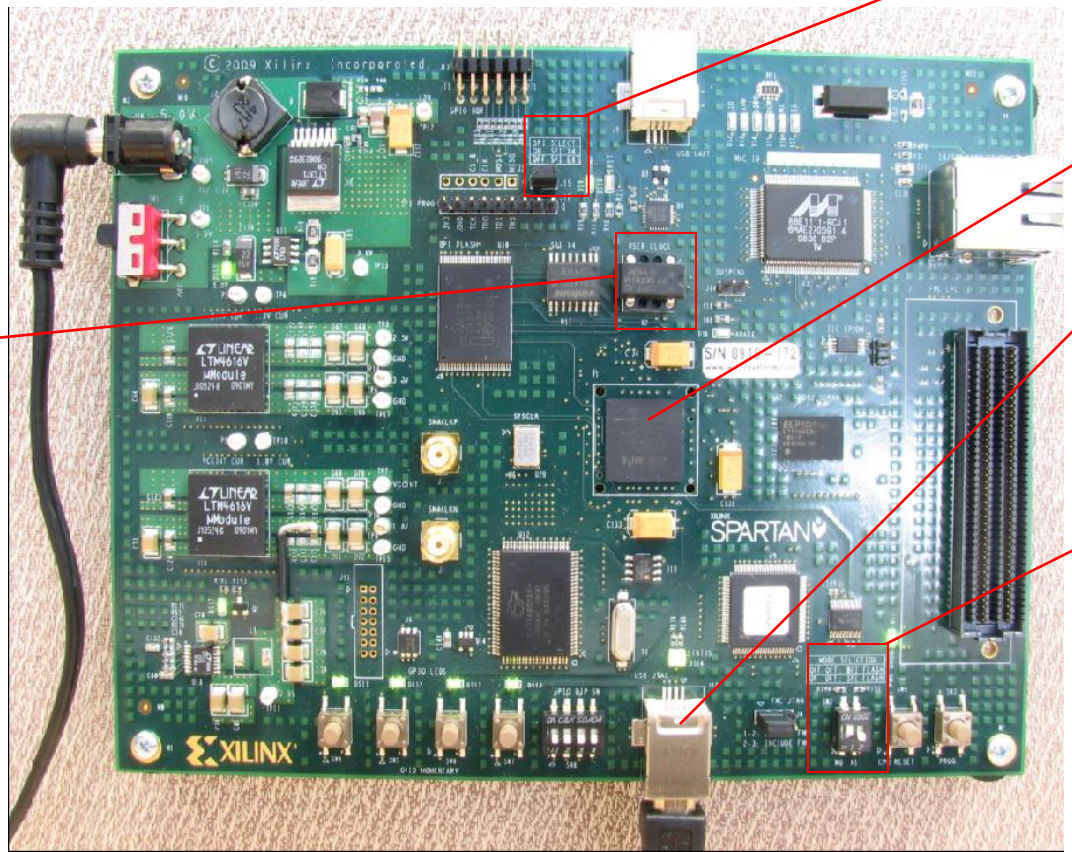
XILINX.

Design Overview and SP601 Board Setup

- This design is targeting XC6SLX16-CSG324 on SP601 Demo boards
- This design is created with ISE 11.2 (L.46) on Windows XP.
- The bitstream inside the SPI flash on SP601 will be loaded to the Spartan-6 in SPI x4 mode with external master clock.
- The design will make the four LEDs on SP601 blink one by one and then all the four LEDs will blink once again simultaneously.
- The configuration time should be within 100 ms, which can meet the specification of PCIe.



- J15 should be closed
- SW2 (M[1:0]) should be 01



XC6SLX16

USB Cable

External Master Clock Oscillator

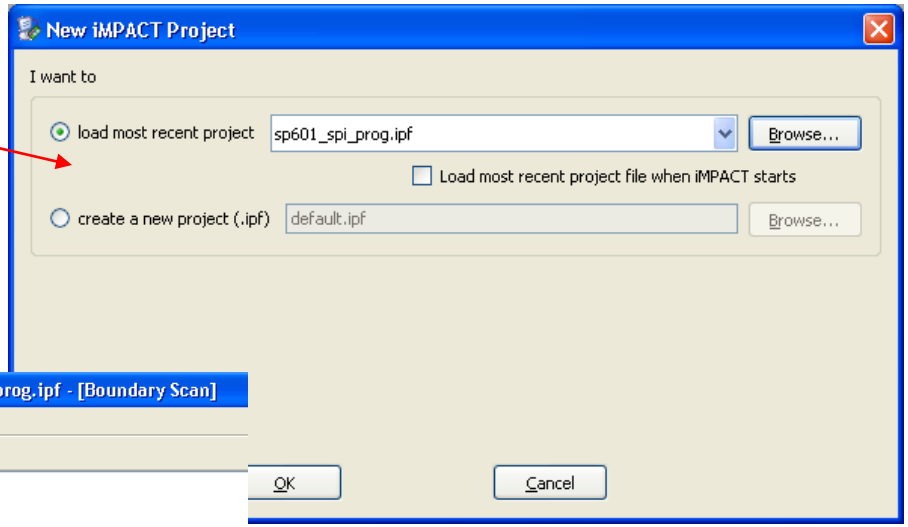
MODE pins

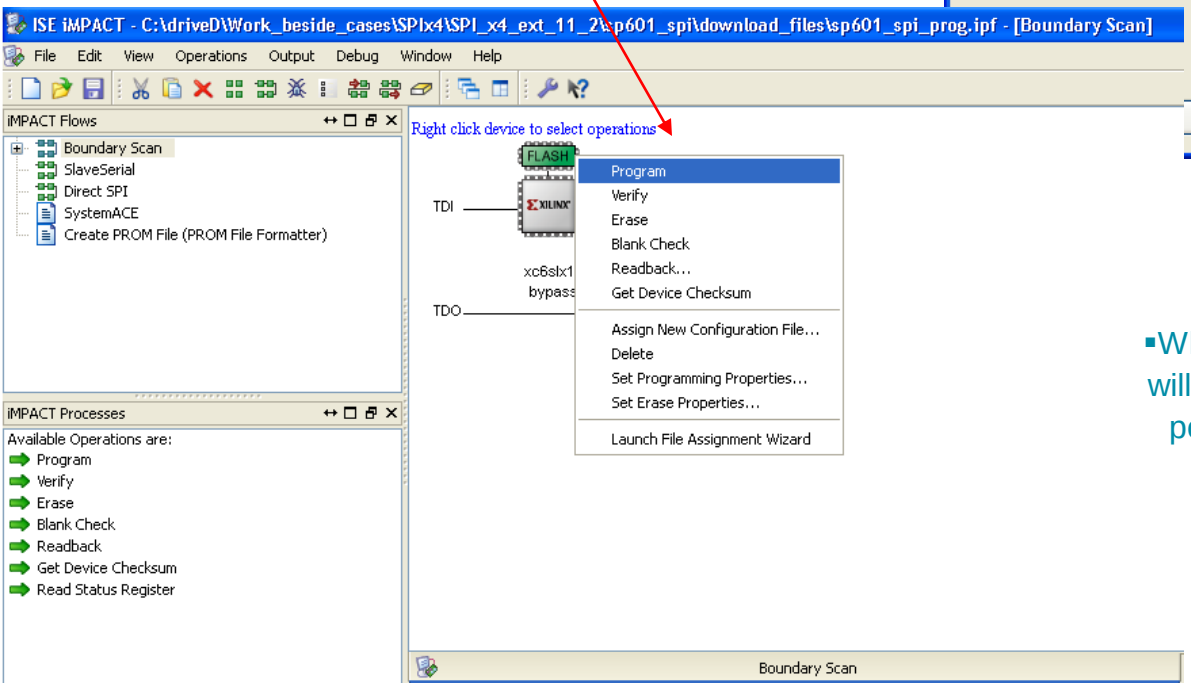
XILINX

Programming the spi_x4_ext.mcs file to the SPI Flash

- Invoke iMPACT and load the sp601_spi_prog.ipf located in sp601_spi\download_files\ folder

- Right click on FLASH and select Program



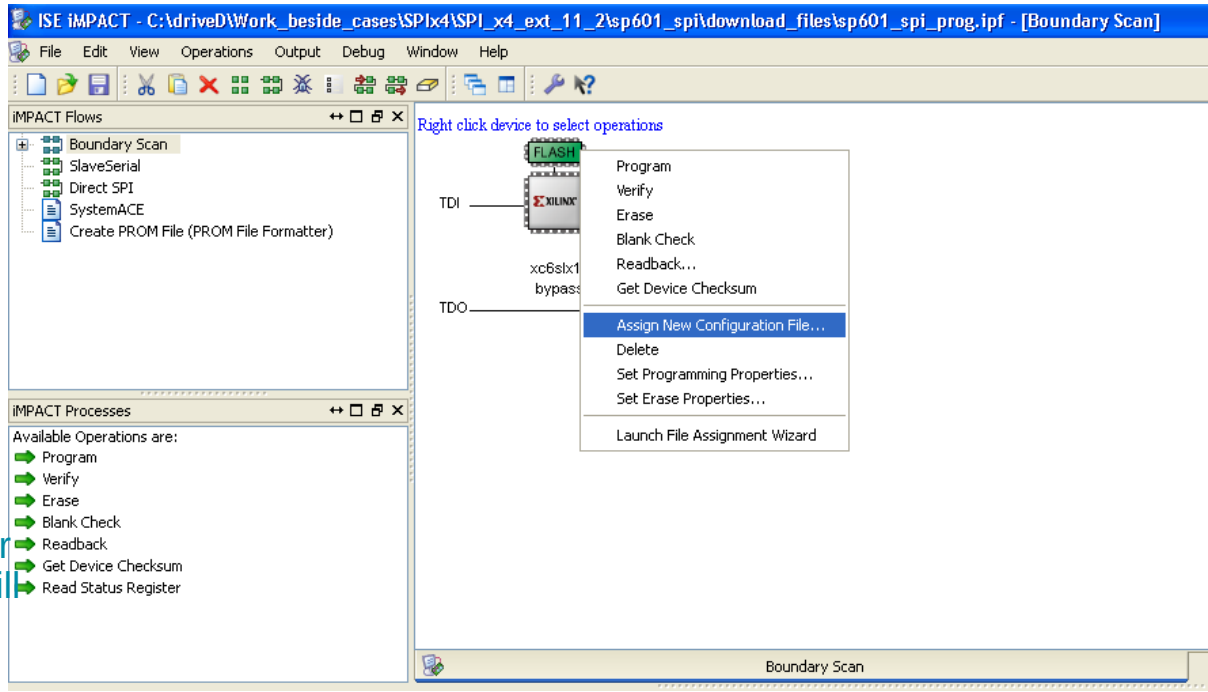


- When it finished programming, cycle the power you will see the design will be working immediately after power up. The configuration time calculation and measurement will be covered later.

XILINX

Programming the other files to the SPI Flash

- Right click on FLASH and select Assign New Configuration file…so that you can program other .mcs files located in the sp601_spi\download_files\ folder
- Make sure that the .cfi file with the same root name as the .mcs file should be placed in the folder*
- For example, you can program the spi_x1_int.mcs into the Flash. After you cycle the power supply, you will feel it needs to take a while before the FPGA starts working.



- In sp601_spi\download_files\ folder, there are 4 .mcs files, called spi_x4_ext.mcs, spi_x4_int.mcs, spi_x1_ext.mcs and spi_x1_int.mcs.
- The file names can explain the settings. For example, spi_x4_ext.mcs is generated from the .bit file with both SPI x4 and external master clock are chosen.

XILINX.

Agenda

- Why do we use SPI x4?
- How to use the Demo design?
- How to create the SPI x4 design?
- Measurements compared to the calculations

XILINX.

BitGen - .bit File Generation

- Command Line

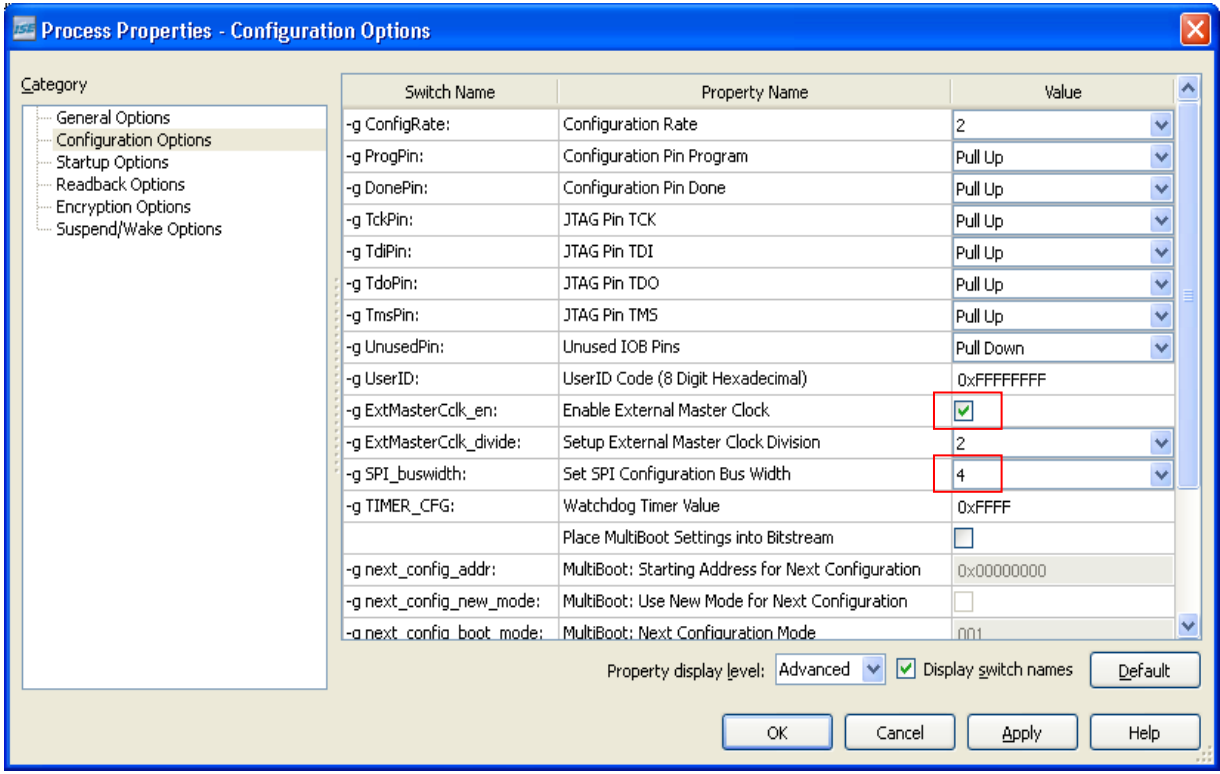
  -bitgen -w -g ExtMasterCclk_en:Yes -g ExtMasterCclk_divide:2 -g SPI_buswidth:4 top_SP601.ncd spi_x4_ext

- GUI

  1, Right click on Generating Programming File and select Process Properties… then the window on the right will pop up

  2, You can check Enable External Master Clock and Set SPI Configuration Bus Width to 4

  3, This will generate the .bit file with SPI x4 and external master clock enabled.



- With different settings combination, you can generate spi_x4_ext.bit, spi_x4_int.bit, spi_x1_ext.bit and spi_x1_int.bit. All these files can be found in the sp601_spi\download_files\ folder.
- When generating spi_x4_ext.bit and spi_x1_ext.bit, Setup External Master Clock Division is set to 2.*

XILINX.

PROM File Formatter - .mcs file generation

- Command line

  promgen -w -p mcs -c FF -o spi_x4_ext -s 8192 -u 0000 spi_x4_ext.bit -spi

- PROM File Formatter

  1, Select Storage Target: SPI Flash → Configure Single FPGA

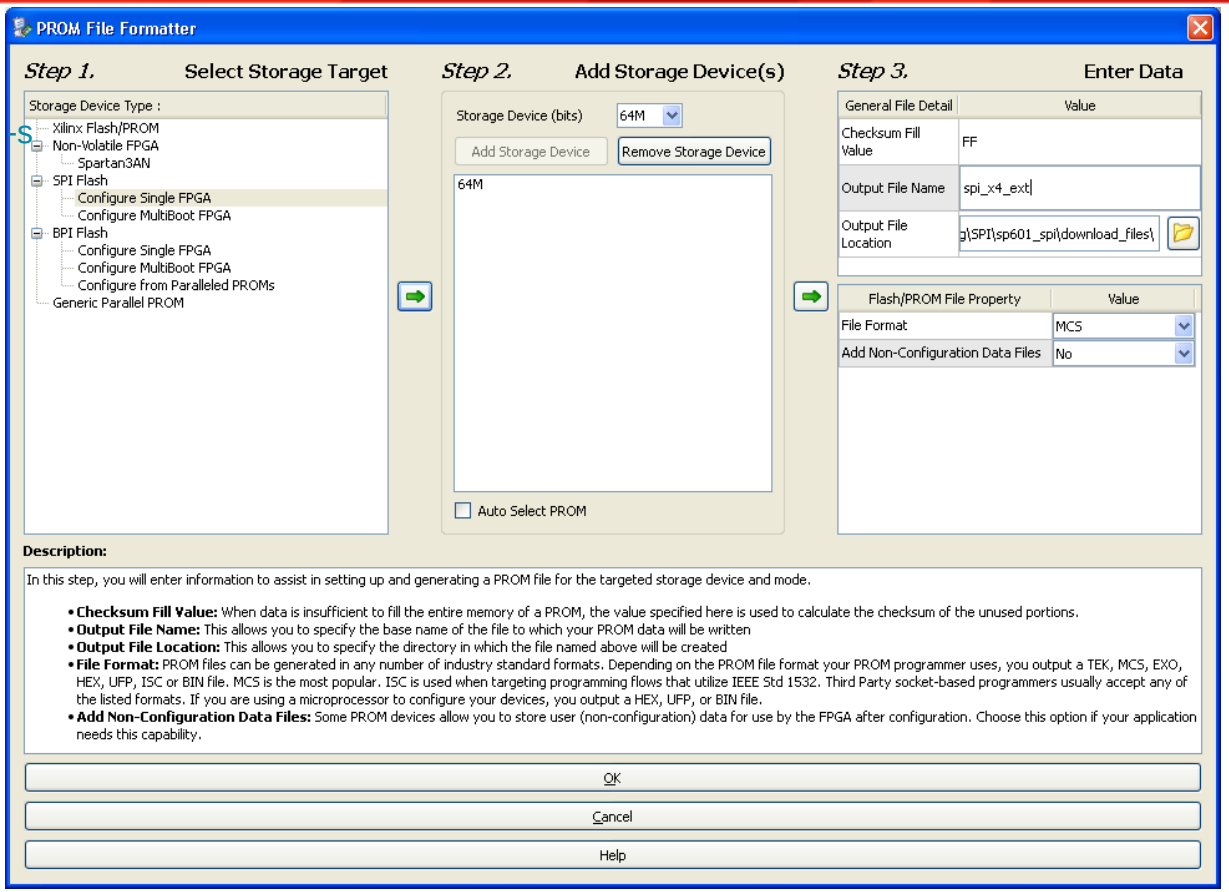
  2, Add Storage Device(s): 64M

  the SPI flash is 64M bit on SP601

  3, Enter Data:

  you can specify the following

  - Output File Name
  - Output File Location
  - File Format: MCS
  - Add Non-Configuration Data Files: no



XILINX

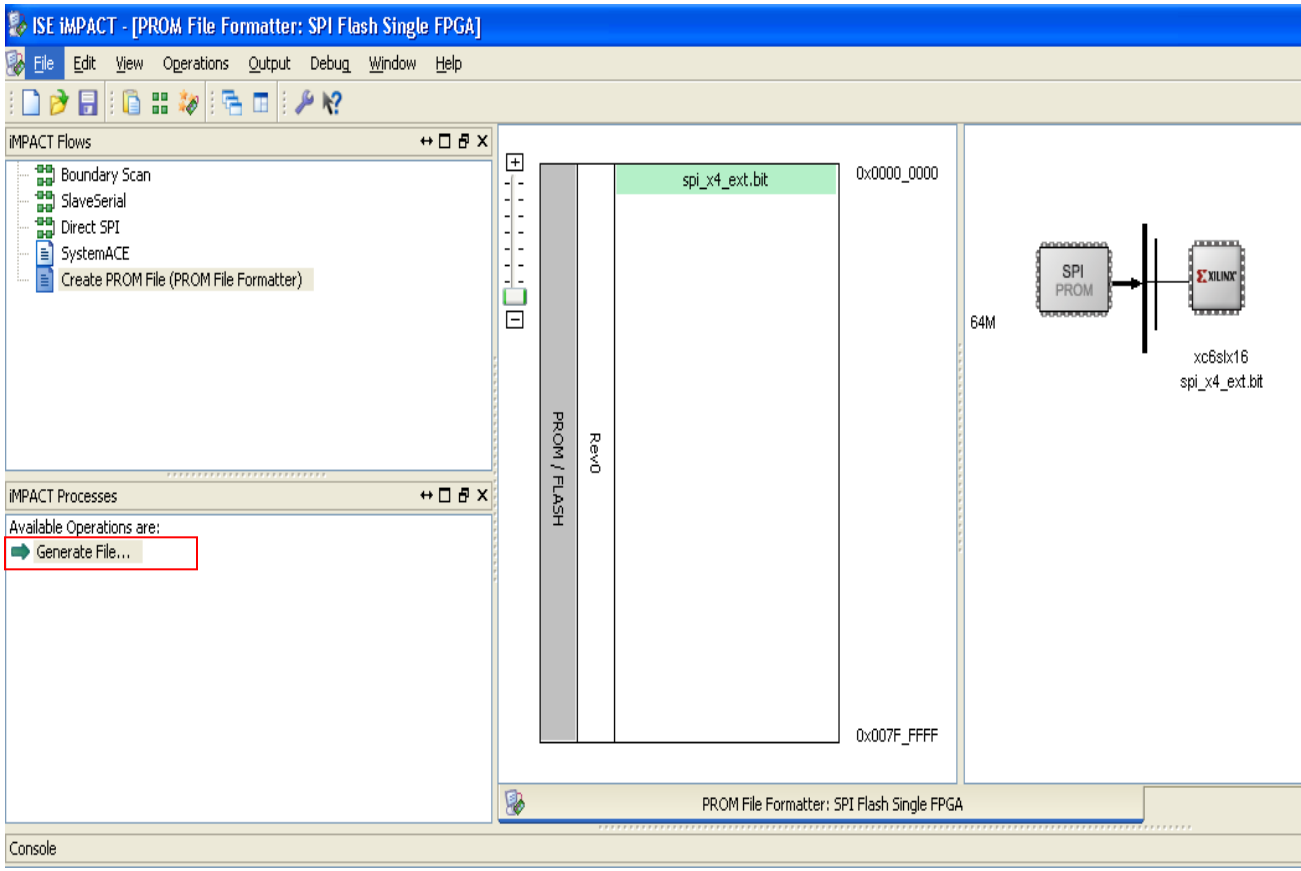
PROM File Formatter - .mcs file generation (cont.)

4, Assign the .bit file in PROM File Formatter

5, Double click on the Generating File… and you will see the spi_x4_ext.mcs file as well as the spi_x4_ext.cfi file generated in the folder specified in step3

6, Do the same to get other .mcs files generated from remaining .bit files

Note that when you move the .mcs file to other locations you should move the associated .cfi file along with it.*



XILINX

Agenda

- Why do we use SPI x4?
- How to use the Demo design?
- How to create the SPI x4 design?
- Measurements compared to the calculations

XILINX.

Configuration Time Calculation and Measurement
SPI x4 w/ External Master Clock

- Calculation

  Configuration time = Tpor + bitstream_size/(cclk_freq*buswidth)

  = 10 ms + 3,713,440/(13.5Mhz*4) = 78 ms

  - On SP601, the external oscillator is 27Mhz and it will be divided by 2 because we set -g ExtMasterCclk_divide:2 in bitgen, so the cclk_freq = 13.5Mhz.

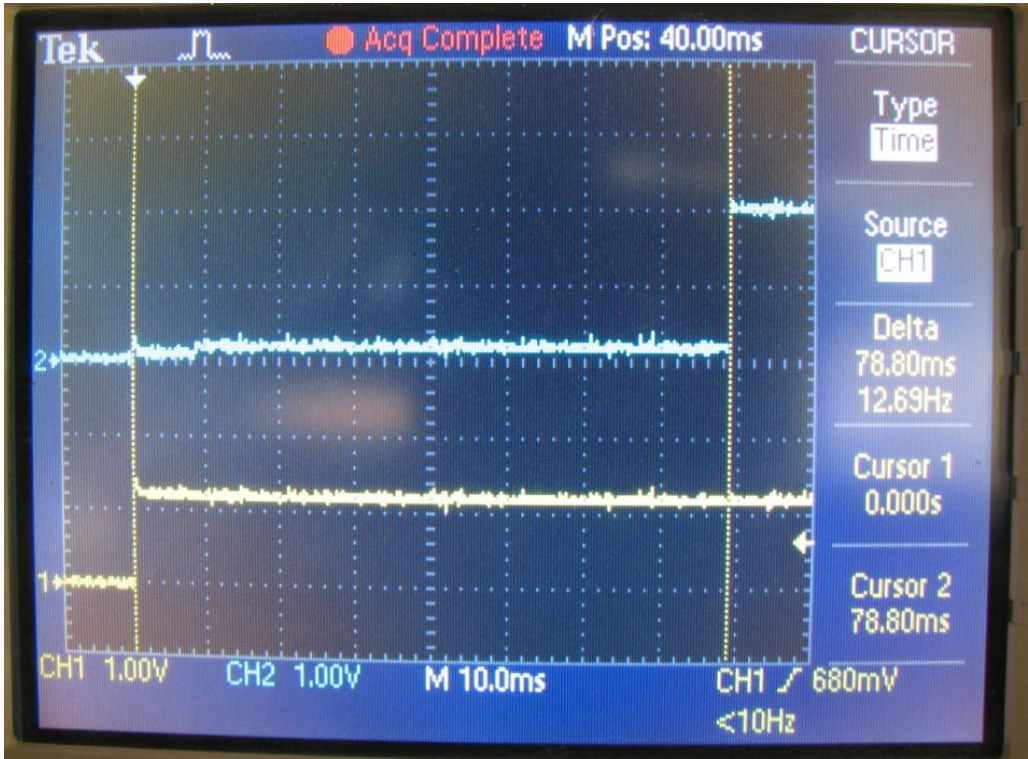
  - Buswidth = 4 because we set –g Spi_buswidth:4 in bitgen

- Measurement on SP601

  w/ an oscilloscope

  See the snapshot on the right*

  Measured configuration time = 78.80ms



Configuration Time Calculation and Measurement
SPI x4 w/ Internal Master Clock

- Calculation

  Configuration time = Tpor + bitstream_size/(cclk_freq*buswidth)

  = 10 ms + 3,713,440/(12Mhz*4)

  = 88 ms

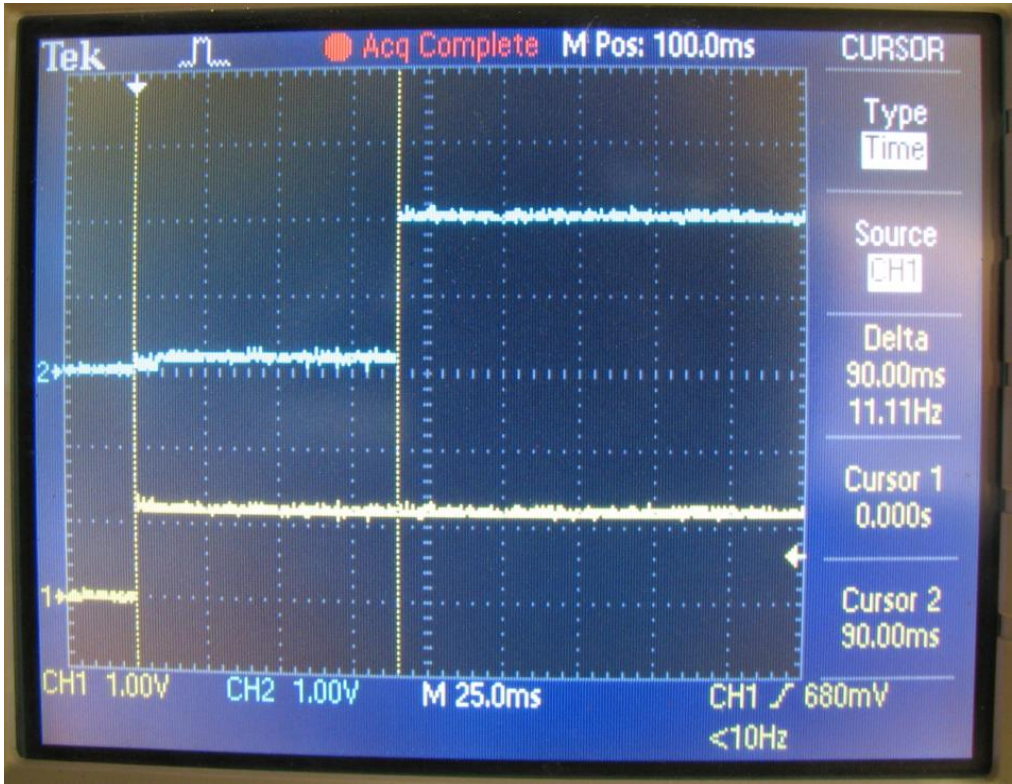
  - ConfigRate is 12 in bitgen so the cclk_freq = 12Mhz*
  - Buswidth = 4 because we set -g Spi_buswidth:4 in bitgen

- Measurement on SP601

  w/ an oscilloscope

  See the snapshot on the right **

  Measured configuration time = 90.00ms



Configuration Time Calculation and Measurement
SPI x1 w/ External Master Clock

- Calculation

  Configuration time = Tpor + bitstream_size/(cclk_freq*buswidth)

  = 10 ms + 3,713,440/(13.5Mhz*1)

  = 285 ms

- Measurement on SP601

  w/ an oscilloscope

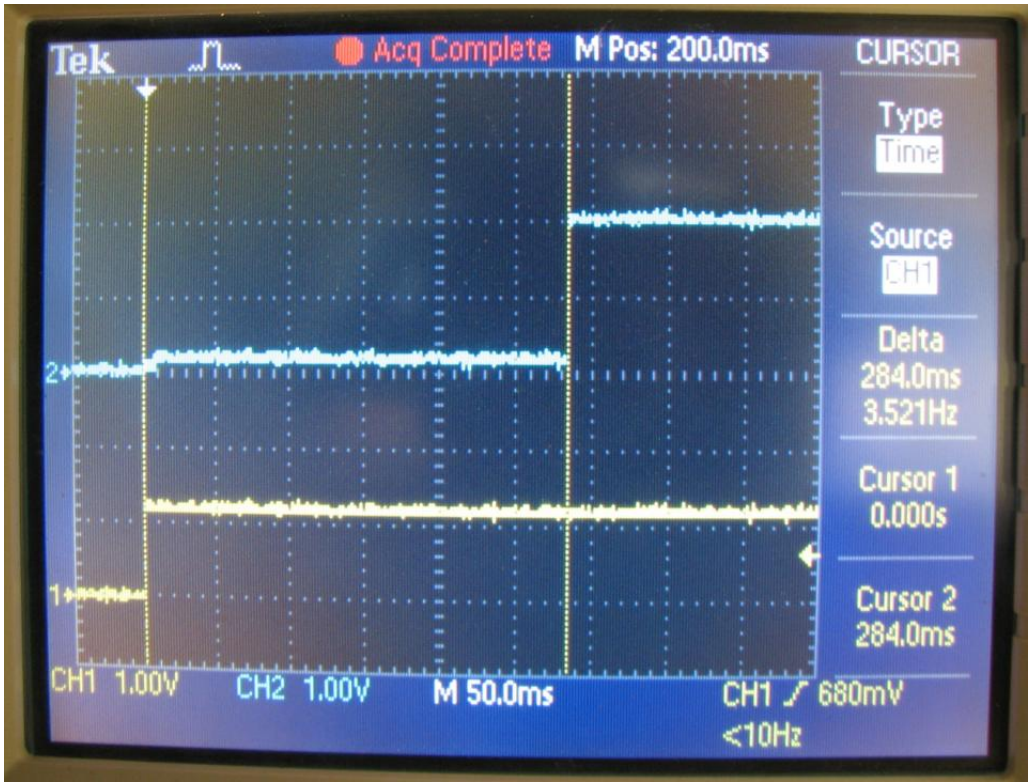
  See the snapshot on the right *

  Measured configuration time = 284.0ms

- Result

  - measurement and calculation are matched

  - the configuration time cannot meet the PCIe spec if you use SPIx1 and external master clock



Configuration Time Calculation and Measurement
SPI x1 w/ Internal Master Clock

- Calculation

  Configuration time = Tpor + bitstream_size/(cclk_freq*buswidth)

  = 10 ms + 3,713,440/(12Mhz*1)

  = 320 ms

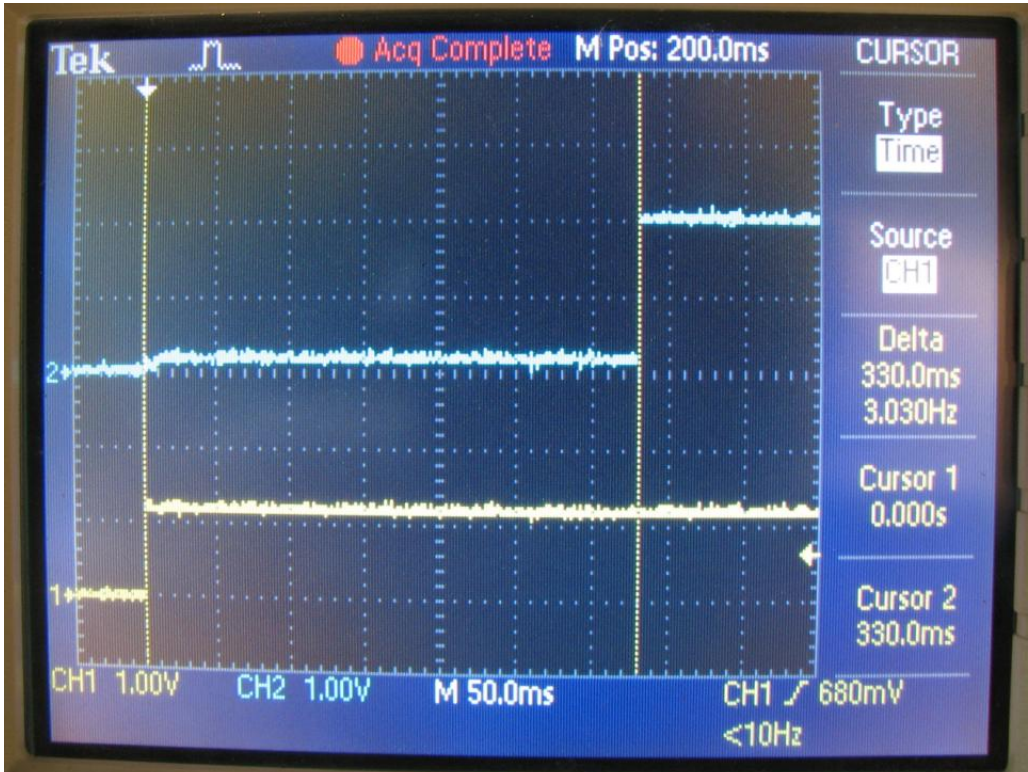
  - ConfigRate is 12 in bitgen so the cclk_freq = 12Mhz*

- Measurement on SP601

  w/ an oscilloscope

  See the snapshot on the right **

  Measured configuration time = 330.0ms



Conclusion

- The reason why we use SPI x4 is to meet fast configuration time requirements.
- According to the test results on SP601

  - SPI x4 can work well with either external master clock or internal generated clock

XILINX.

References

- PCI Express Card Electromechanical Specification, v2.0
- Spartan-6 FPGA Configuration User Guide UG380, v1.0
- Winbond W25Q64BV Datasheet, Preliminary Revision B
- LogiCORE™ IP Spartan-6 FPGA Integrated Endpoint Block for PCI Express® User Guide UG654, v1.1