Xilinx Virtex 5 Proton Accelerated Radiation Test

Melanie Berg – Principle Investigator (PI) MEI,
Test Engineers: HAK Kim, Chris Perez, Anthony Phan: MEI

Test Report Submitted: 012/10
Test Date: 08/5/09 – 08/06/09

1. INTRODUCTION

This study will be undertaken to determine:

1. Configuration memory upset cross section calculations for proton energies ranging from 89MeV to 198MeV
2. Potential Xilinx Single Event Function interrupt sensitivity to proton energies ranging from 89MeV to 198MeV
3. The effectiveness of scrubbing the Xilnx V5 configuration memory during proton irradiation
4. Potential FPGA system operation speed degradation during dose levels less than 150kRad (Si)

2. DEVICES TESTED

The devices are manufactured on an advanced 65nm copper CMOS Process Technology. The manufacturer is Xilinx. The devices were not de-lidded prior to proton testing. The Device Serial number and Lot data code:

XC5VLX30T

FFG665FGU0849

DD2608034A1

1C

Pertaining to the clock sensitivity and speed degradation investigations, there are two shift register designs with external scrubbing that are under examination. The following section is a detailed explanation of the shift register Design Under test (DUT) architecture.

2.1 DUT Architecture

All DUTS consist of windowed shift register strings see Figure 1 . The length of the strings is 300 bits. See Figure 1 for an illustration of DFFs and combinatorial logic within a string. The premise of using a windowed shift register is to reliably perform high speed testing.

In order to calculate accurate cross sections, it is mandatory that the tester have visibility and access to the DUT outputs at every shift clock cycle. Implementation is as follows:

All bits are shifted at every clock cycle. The last 4 DFFs are copied into a window at every clock period. A windowed shift register divides the system clock by-4 clock to shift the last 4 bits of the Shift register string into a DFF window (SCAN_DATA). The window is output to the tester. A data clock (SHIFT_CLK) is also output to the tester for high speed synchronous data capture.

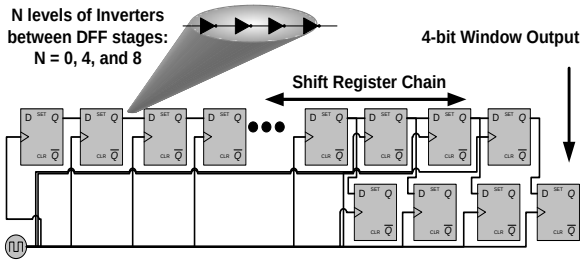
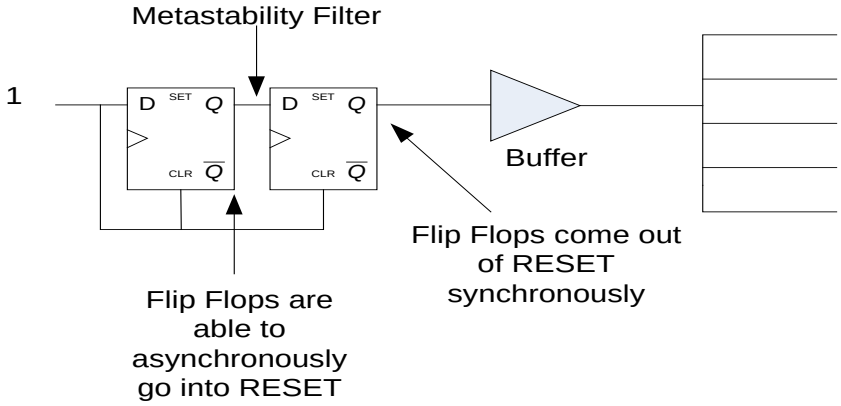


Figure 1: Windowed Shift Register

All DUT architectures contain a degree of TMR redundancy. One version has full XTMR: all data paths, I/O, and clocks are triplicated. The other version has DTMR: all data paths, I/O are triplicated. The clock domains are not. The choice of DTMR testing vs. XTMR testing is to isolate the clock tree in order to detect potential sensitivity.

2.1.1 DUT Internal Reset Circuitry

Reset passes through an asynchronous assert – synchronous de-assert circuit and is supplied to every DFF. The following is the reset circuit used within the DUT.



2.1.2 DUT Mitigation

The probability of error within an FPGA is comprised of 4 major factors:

- $P_{configuration}$: Probability that the configuration memory incurs an upset
- P_{DFFSEU}: Probability that a DFF incurs an upset
- $P_{SET \rightarrow SEU}$: Probability that a DFF captures a transient (SET)
- P_{SEFI}: Probability that a Single Event Functional Interrupt occurs

$$P(fs)_{error} \propto P_{Configuration} + P_{DFFSEU} + P(fs)_{SET \rightarrow SEU} + P_{SEFI}$$

The objective of mitigation is to reduce the probability of system error $(P(fs))_{error}$ by inserting redundancy and/or correction within the overall system design. Redundancy+correction was inserted into the functional logic using Distributed triple modular redundancy (DTMR) and Global Triple Modular Redundancy.

2.1.2.1 DTMR

DTMR: All functional paths are triplicated and voters are inserted after every DFF with a feedback path. No global routes are triplicated (e.g. clocks and resets are shared). DTMR is illustrated in Figure 2 .

With DTMR, many configuration upsets are masked by the TMR circuitry, however, due to the single points of failure, upsets in the configuration memory can disturb global routes and hence create global SEFI failures.

$$P(fs)_{error} \propto P_{SEFI} + P_{global}$$

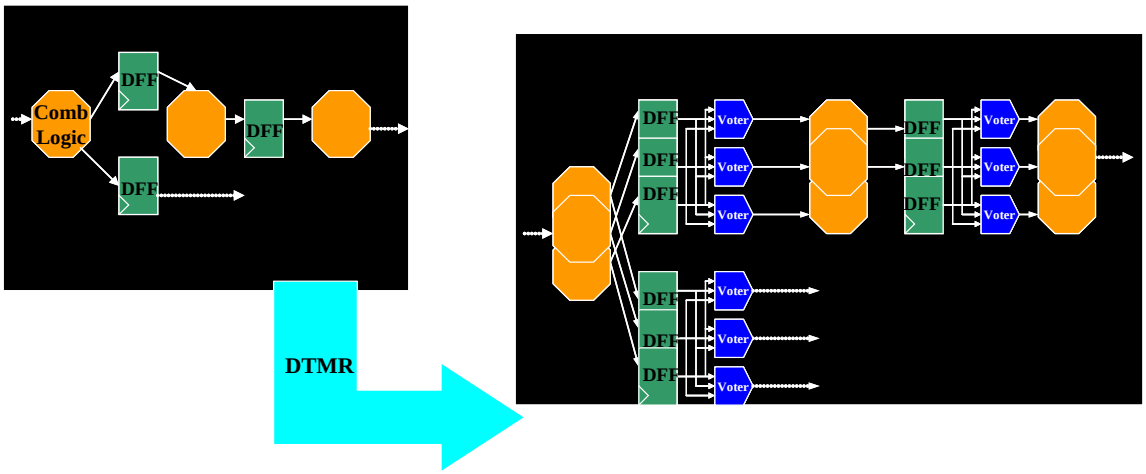


Figure 2: Distributed Triple Modular Redundancy (DTMR)

2.1.2.2 GTMR

GTMR: All functional paths are triplicated and voters are inserted after every DFF with a feedback path. All global routes are triplicated (e.g. clocks and resets are NOT shared). GTMR is illustrated in Figure 3. With GTMR, most configuration upsets are masked by the TMR circuitry. Consequently, there are very few single points of failures, hence the system error is proportional to the SEFIs that are unique to the V5 device (e.g. POR, selectMap interface, etc…).

$$P(fs)_{error} \propto P_{SEFI}$$

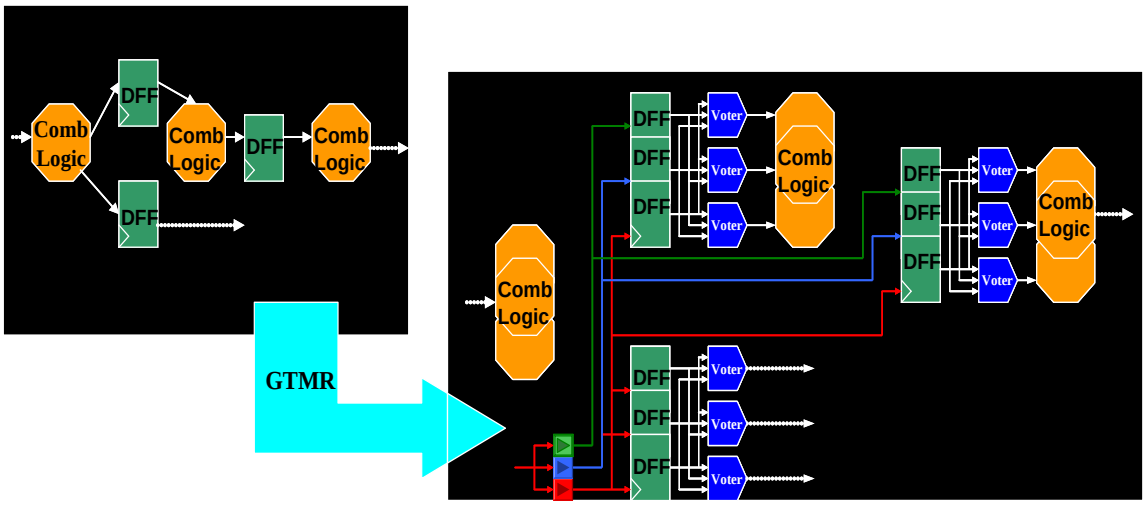


Figure 3: Global Triple Modular Redundancy (GTMR)

2.1.2.3 Scrubbing

Although the voting within the TMR logic can correct functional upsets, it will not fix upsets that occur within the configuration memory. Consequently, scrubbing is suggested so that upsets do not accumulate within the configuration memory. Two methods of scrubbing were implemented during testing: External Blind scrubbing and Internal (TMR'd) scrubbing using special V5 readback circuitry.

2.1.2.3.1 External Blind Scrubbing

This mode of scrubbing is controlled completely by the tester. During DUT operation, the tester sends a bit stream comprised of commands and data via the DUT selectMap interface. The commands control the configuration to continuously stay active while accepting and writing configuration data. While the configuration is being over-written, there is no disruption or glitching during DUT operation.

The scrub bit stream is derived from the configuration bit stream. The only difference between the two bit streams is the command area. The configuration commands take the device down as opposed to the scrubbing commands. A structural comparison is illustrated in Figure 4.

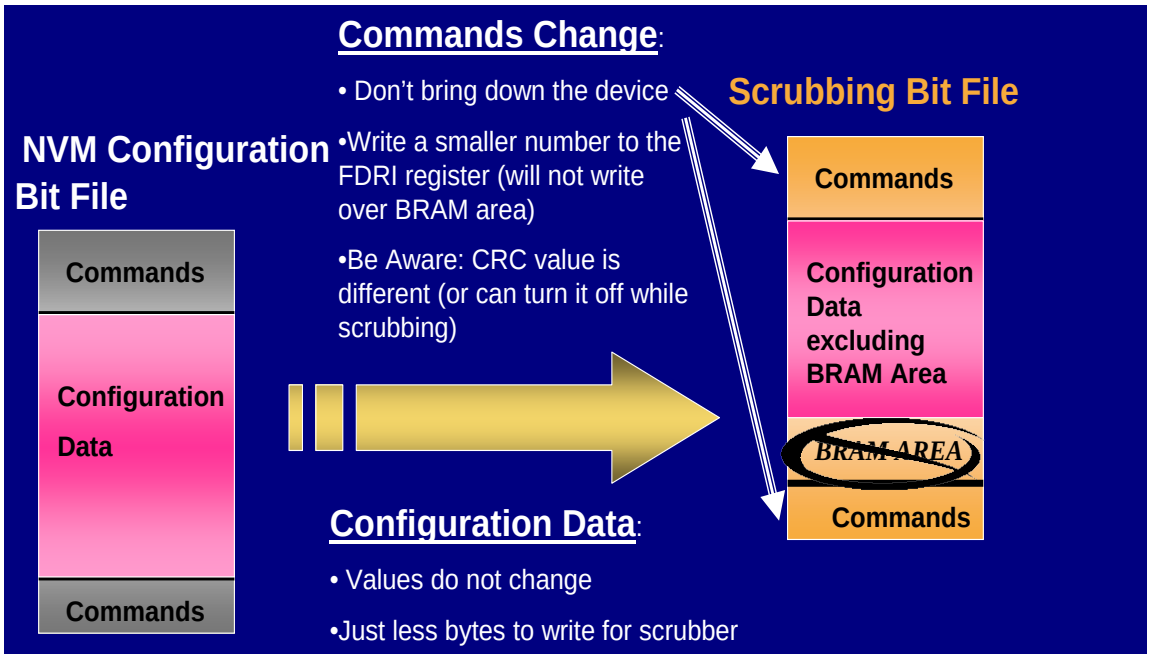


Figure 4: Configuration Bit stream vs. Scrubbing Bit Stream

2.1.2.3.2 Internal Scrubbing (Self-Scrubber)

The Self-Scrubber is contained within the DUT. It utilizes the readback circuitry that is in the V5 family of devices (not resident in previous families). The self scrubber continuously reads the configuration memory. During this process, the DUT operation is not disrupted. The readback circuitry contains a CRC checker that will signal if an error has occurred. Upon error, the tester is signaled, the tester then takes control, and the tester sends the scrub data to the DUT. The DUT has TMRd circuitry that accepts this data and sends it to the selectMap interface via the ICAP circuitry. The Frame ECC is completely bypassed because no correction is calculated. Upsets are detected and then the full configuration is scrubbed.

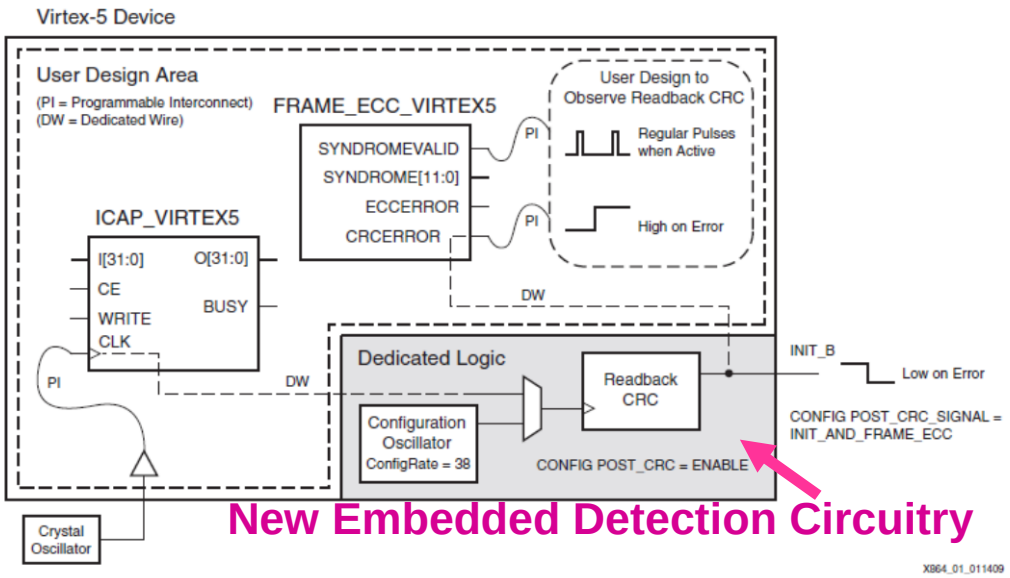


Figure 5: Embedded Readback Circutiry within V5 Family of Devices

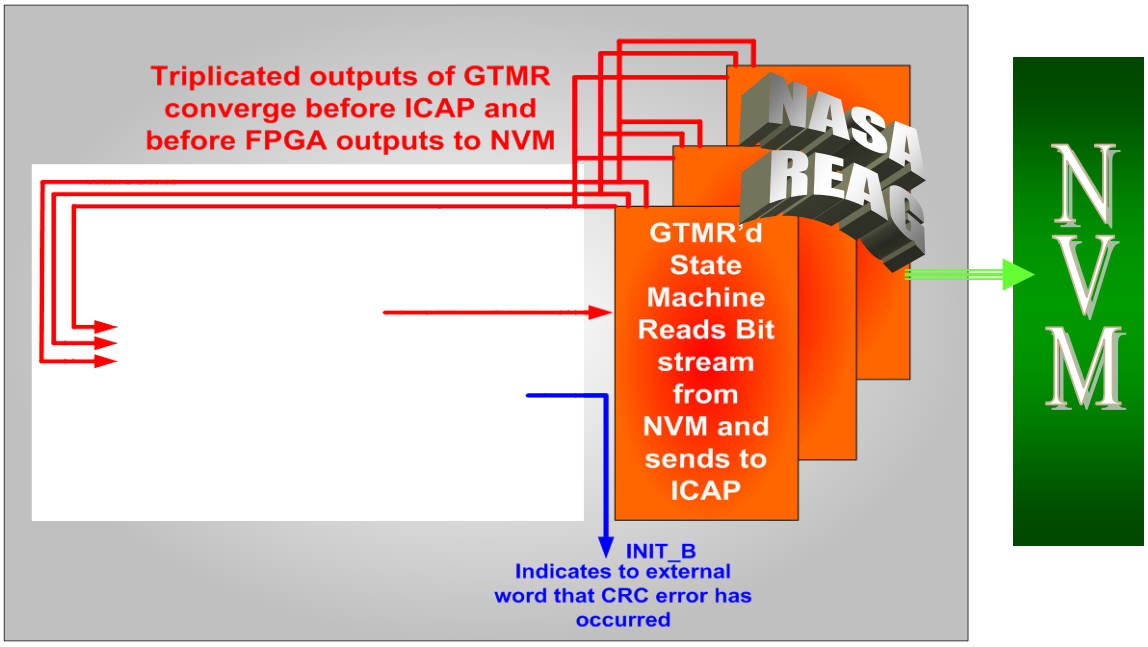


Figure 6: NASA REAG GTMR'd Internal Scrub Control Plus Embedded Readback Logic

3. PROTON TEST FACILITY

Facility: University of Indiana Cyclotron Facility
Flux: 3.0E^{9} particles/cm^{2}*s
Fluence: All tests will be run to a significant amount of errors were observed or a fluence corresponding to a selected dose between 10krad to 50Krad
Energy : 89MeV to 198 MeV

4. TEST CONDITIONS

Test Temperature: Room Temperature
Operating Frequency: 60MHz to 100MHz
Power Supply Voltage: 3.3v I/O; 2.5V Auxiliary; 1.8V PROM; 1.2V Core

5. TEST METHODS

5.1 Architectural Overview

The High Speed Digital Tester (HSDT) consists of a Mother Board (FPGA Based Controller/Processor) and a daughter board (containing DUT and its associated circuitry). The DUT is socketed onto the daughter card. The objective of the DUT Controller/processor (mother board) is to supply inputs to the Virtex-5 Device (DUT – daughter card) and perform data processing on the outputs of the Virtex-5. The HSDT communicates with a user controlled PC. The user interface is LAB-VIEW (see Figure 7). The user will send specified commands to the motherboard and receive information from the motherboard via the LabView GUI. Burst and ErrorCnt windows are available for the user to determine the state of the DUT (temporary error state vs. unrecoverable error state). ShftData_FFn is a window that contains DUT output error information. One ShftData_FF0 represents all 6 shift_registers during non-TMR circuit testing. There are 3 ShftData_FFn windows for future TMR testing. The Alive button indicates that the tester (not the DUT) is still alive. The Alive button flashes as the test is running. Please see Documents: “HSDT” and “General Tester” for further information concerning the HSDT functionality.

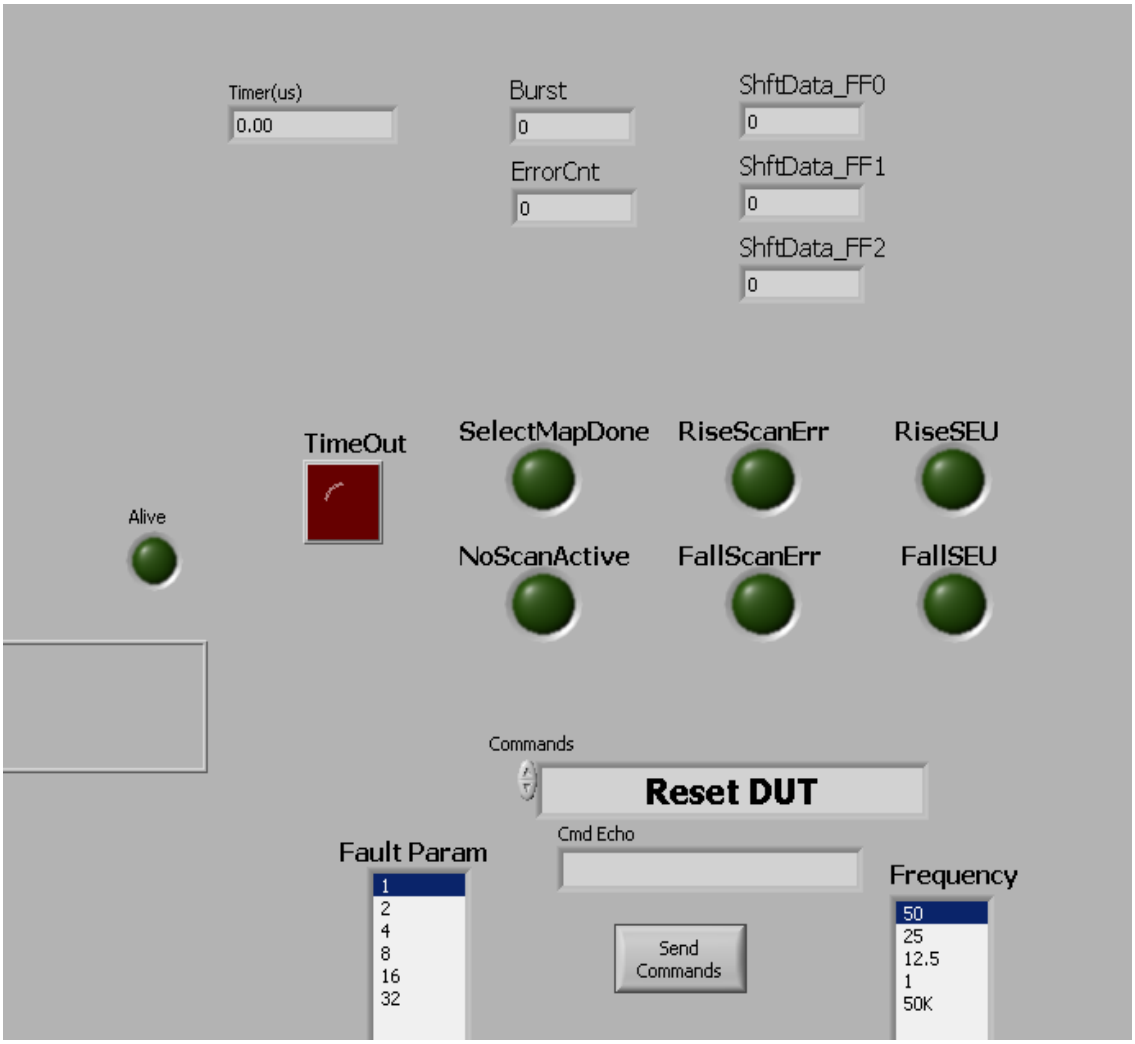


Figure 7: LABView User Interface (Resides on Host PC)

The HSDT is connected to the Virtex-5 DUT as shown in the following Block Diagram.

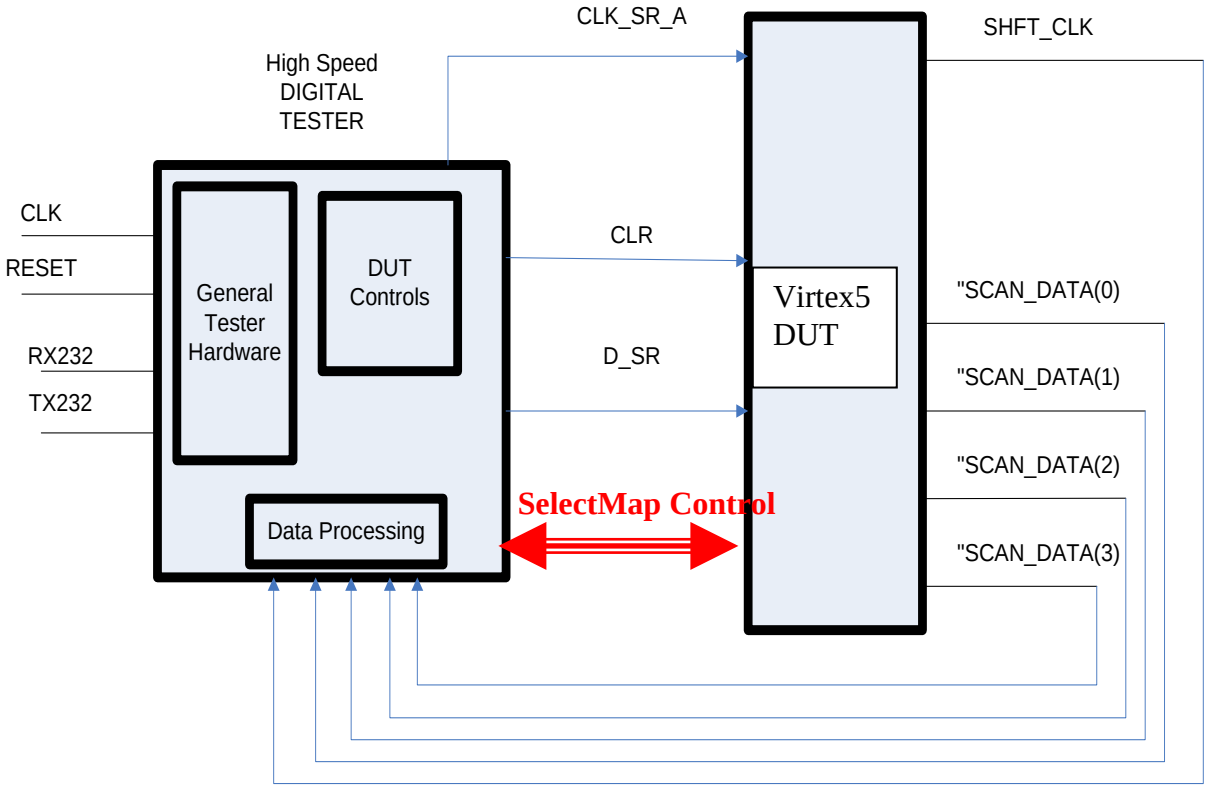


Figure 8: System Level Tester Architecture

5.1.1 Tester I/O List and Definitions

Table 1: I/O Table

Input Name	Description	Direction	Synchronous	Slew	Pullup
CLK	System clock of the HSDT	Input	Clock		N
RESET	HSDT system reset	Input	A		N
RX232	Serial receive input	Input	A		N
SCAN_DATA_TMR0(23:0)	(6 WSR chains= 24 bits) Data window of Virtex-5 . Data is processed by HSDT and compared against expected value	Input	A		N

SCAN_DATA_TMR1(23:0)	(6 WSR chains= 24 bits) Data window of Virtex-5 . Data is processed by HSDT and compared against expected value	Input	A		N
SCAN_DATA_TMR2(23:0)	(6 WSR chains= 24 bits) Data window of Virtex-5 . Data is processed by HSDT and compared against expected value	Input	A		N
SHIFT_CLK_TMR0	Output clock of VIRTEX-5 .Used to control SCAN_DATA capture. SHIFT_CLK is always ¼ the speed of CLK_SR_A. However it is not synchronous with CLK_SR_A	Input	A		N
SHIFT_CLK_TMR1	Output clock of VIRTEX-5 .Used to control SCAN_DATA capture. SHIFT_CLK is always ¼ the speed of CLK_SR_A. However it is not synchronous with CLK_SR_A	Input	A		N
SHIFT_CLK_TMR2	Output clock of VIRTEX-5 .Used to control SCAN_DATA capture. SHIFT_CLK is always ¼ the speed	Input	A		N

	of CLK_SR_A. However it is not synchronous with CLK_SR_A				
CLK_SR_A_TMR0	Input clock to VIRTEX-5 . Max speed is 150mhz	Output		FAST	N
CLK_SR_A_TMR1	Input clock to VIRTEX-5 . Max speed is 150mhz	Output		FAST	N
CLK_SR_A_TMR2	Input clock to VIRTEX-5 . Max speed is 150mhz	Output		FAST	N
CLR_TMR0	Reset to the Virtex-5	Output		FAST	N
CLR_TMR1	Reset to the Virtex-5	Output		FAST	N
CLR_TMR2	Reset to the Virtex-5	Output		FAST	N
D_SR_TMR0	Data Input to the Virtex-5	Output			N
D_SR_TMR1	Data Input to the Virtex-5	Output			N
D_SR_TMR2	Data Input to the Virtex-5	Output			N
TX232	Serial transmission line	Output			N
Selectmap_CCLK	SelectMap 30MHz clock	Output			
Selectmap_CSN	SelectMap Data valid	Output			
Selectmap_DONE	SelectMap Done (configuration) signal	Output			
Selectmap_BUSY	SelectMap Busy	Output			
Selectmap_INIT	SelectMap INIT	Output			

Selectmap_PROG_B	SelectMap Reset Signal	Output			
Selectmap_RW_B	SelectMap Read/Write	Output			
Selectmap_DATA	SelectMap 8bit Data	Output			
SCRUB_RX232	Serial input to receive configuration data via RS232 port	Input	A		
SCRUB_Tx232	Not Used				
SRAM_D	SRAM Data 16bit	INOUT	A		
SRAM_A	SRAM Address = 20 bits	Output			
SRAM_WR	SRAM read/write	Output			
SRAM_OE	SRAM output enable	Output			
SRAM_CE	SRAM chip select	Output			
SRAM_BLEN	SRAM byte enable	Output			
SRAM_BLHN	SRAM byte enable	Output			
RUN_SCAN	Turn self scrubber on	Output			
SCAN_ACTIVE	Self scrubber	Input	A		
SCAN_ERROR	Self scrubber	Input	A		
SCAN_MODE	Self scrubber	Output			
ERROR_INJECT	Self scrubber	Output			
SEU_DETECT	Self scrubber	Output			
QUICKUSB_FCLK	50MHz clock from quickusb device	Input	CLK		
QUICKUSB_REN		Output			
QUICKUSB_WEN		Output			
QUICKUSB_DATA	8 bit data	INOUT	A		

QUICKUSB_CMD_DATA	Command differentiation	Output			
TP*n*	Test points	Output			

5.2 Requirements

5.2.1 Requirement Summary

There are 3 main investigations:

1. Test shift register logic structures
2. Test external scrubber

The requirements for the Virtex-5 HSDT tester are listed in Table 3.

Table 2 – Requirements Table

Item	Requirement
1	Supply System Clock to the Virtex-5 DUT
2	Supply Reset to Virtex-5
3	Supply Data Input to the Virtex-5
4	Clock Frequency of Virtex-5 shall be variable
5	Maximum Virtex-5 input clock frequency shall be 100Mhz
6	0,1, and checker board data patterns shall be generated and placed on the VIRTEX-5 data lines
7	VIRTEX-5 reset shall be active low
8	VIRTEX-5 reset shall be active for at least 3 VIRTEX-5 system clocks
9	VIRTEX-5 Data Inputs shall be stable at the Rising Edge of the VIRTEX-5 system clock with a set-up time of 3ns and a hold time of 3ns
10	VIRTEX-5 data inputs shall be captured by the HSDT data processing module once detecting the rising edge of the data clock (SHIFT_CLK)
11	SHIFT_CLK rising edge detection must include a metastability filter because the SHIFT_CLK input is asynchronous.
12	Input Data must be registered before the data processing block implements the compares – protects against radiation induced I/O transients.
13	Data processing block shall report every error to the FIFO block

14	Tester must supply external scrubber
15	External scrubber clock shall be separate than the system (shift register) clock
16	External scrubber clock shall be able to functional at 25MHz

The tester supplies inputs as follows: Data (D_SR) changes at the falling edge of the input clock (CLK_SR) so that it is stable and can be captured at the rising edge. CLK_SR and D_SR will be at the user specified frequency.

5.3 User Commands and Control

The primary method of which the User controls the tests is via a LABVIEW interface(see Figure 7 for picture of LabView GUI) running on a host PC (noted as PC1 in Figure 9. PC 1 communicates with the HSDT with a RS232 serial link. The format of communication is a command/Data 4 byte word (see Table 3 : Summary of Commands Used in Virtex-5 Tester).

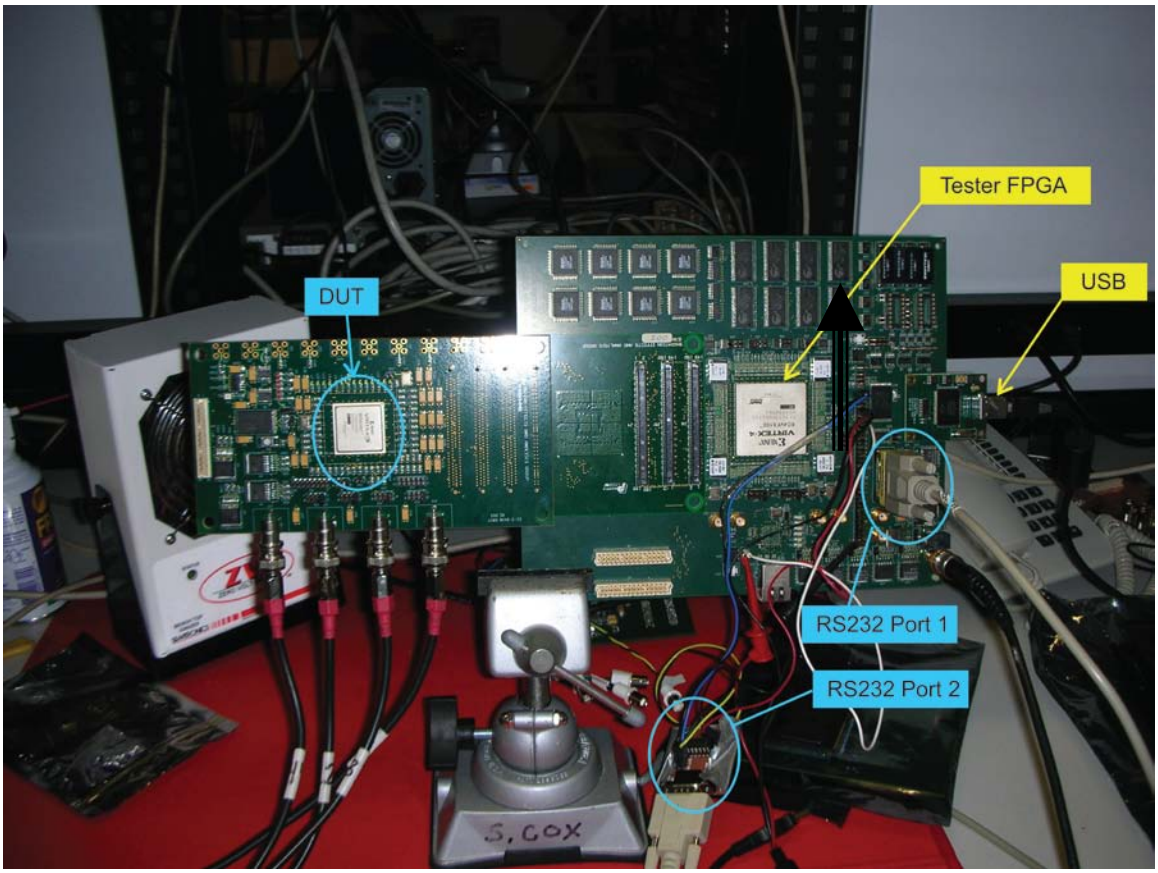


Figure 9: Tester and Xilinx DUT Board.

Table 3 : Summary of Commands Used in Virtex-5 Tester

Command #Hex	Command	D0	D1	D2	Description
01	Reset DUT	N	N	N	Resets VIRTEX-5
02	Start Test	N	N	N	Starts VIRTEX-5 clock and data generation
A0	Clock Frequency	Y	N	N	Clock frequency divider of 150mhz
81	Write Configuration Data	N	N	N	PC sends configuration data to Tester via RS232 port. Tester stores data in onboard SRAM
04	Start configuration	N	N	N	Tester configures the by sending the downloaded configuration data to the via the SelectMap parallel port.
06	Start Scrub	N	N	N	Scrubbing is turned on
0E	Inject error On	N	N	N	Scrubber will turn on error injection
7A	Scrub Error High	Y	Y	Y	Upper address bound for error inject
79	Scrub Error Low	Y	Y	Y	Lower address bound for error inject
05	Start Readback	N	N	N	Turns readback on
7B	End of Configuration	Y	Y	Y	Memory Location (relative to configuration file) to stop scrubbing – avoids BRAM
89	Set SelectMap Control Register	Y	Y	Y	24 bit value to be placed in control register
8A	Set SelectMap Mask Register	Y	Y	Y	24 bit value to be placed in Mask register

The following sections are detailed descriptions of commands and their associated functionality.

5.3.1 RESET DUT:

The RESET DUT command is decoded as x01. The following represents the command as noted in Table 3:

x01	xx	xx	xx

Figure 10: Reset Command Format – Command Number, D0, D1, and D2

Once decoded, all DUT inputs will go into reset mode (Reset, CLK_SR and D_SR are low; SelectMap Interface and QuickUSB interface are in reset)

5.3.2 START TEST:

START TEST is decoded as x02. The following represents the command as noted in Table 3:

x02	xx	xx	xx

Figure 11: Start Command Format

All other commands should be supplied before start test. I.e. the user should define the pattern and clock frequency before administering a start. This command activates the CLK_SR and D_SR DUT inputs. DUT must be configured before this command is sent.

5.3.3 CLOCK FREQUECY:

The clock frequency command is decoded as xA0. The following represents the command as noted in Figure 12:

xA0	xnn	xx	xx

Figure 12: Clock Frequency Command Format

Upon the receipt of this command, D0 is used as a clock frequency divider. This command must be sent after a RESET DUT and before a START TEST. D0 must be an even number and must be greater than or equal to 2. The associated output is CLOCK_FREQ. See the HSDT General Tester for more information concerning the processing of CLOCK_FREQ.

5.3.4 Write Configuration Data

5.3.4.1 (Old method used in 2006 tests):

The command is decoded as x81. The PC sends the configuration bit file via the RS232 link to the tester. The bit file is generated from Xilinx IMPACT and is actually the ".bin" file. Bin file must be 977488 bytes long.

x81	xx	xx	xx

Figure 13: RS232 Write Configuration Command Format

5.3.4.2 New Method – QuickUSB 2007 Tests

New method of sending data was used in the 2007 test setup. RS232 link takes roughly 85 seconds. We developed a USB controller using the QUICKUSB parallel to serial device. The full data set is now able to be sent under 1 ms.

5.3.5 Start Configuration

The command is decoded as x04. Once the configuration data has been dumped to SRAM (either by RS232 link or QuickUSB link), the tester is able to configure the DUT. Data does not need to be sent to SRAM every time the user wants to configure. Once bin file is stored in SRAM (and SRAM has not been corrupted by a data run or loss of power), then the user can reconfigure repeatedly.

x04	xx	xx	xx

Figure 14: Start Configuration Command Format

5.3.6 Start Scrub

The command is decoded as x06. Starts externally scrubbing the DUT configuration-memory. DUT must first be configured (along with the corresponding bin file dumped into onboard Tester SRAM).

x06	xx	xx	xx

Figure 15: Scrub Command Format

5.3.7 Inject Error

The command is decoded as x0E. DUT must first be configured (along with the corresponding bin file dumped into onboard Tester SRAM). Start Scrub must be turned on. This command should be used in conjunction with Scrub Error High and Scrub Error low commands. Command will inject error within a range of configuration address spaces. If the Scrub Error High and Scrub Error low commands are not used, Error injection will occur from address x800 to x70000 (relating to actual bin file byte addressing)

.

x0E	xx	xx	xx

Figure 16: Start Injecting Errors Command Format

5.3.8 Scrub Error High

The command is decoded as x7A. Designates the upper bound address of configuration memory error injection. See section 5.3.7.

x7A	MSB	NN	LSB

Figure 17: Error Injection Upper Bound Command Format

Address is 20 bits (19:0). MSB 3:0 is used; MSB 7:4 is unused. NN and LSB 7:0 is used. Address(19:0) = MSB(3:0)NN(7:0)LSB(7:0)

5.3.9 Scrub Error Low

The command is decoded as x79. Designates the lower bound address of configuration memory error injection. See section 5.3.7.

x79	MSB	NN	LSB

Figure 18: Error Injection Lower Bound Command Format

Address is 20 bits (19:0). MSB 3:0 is used; MSB 7:4 is unused. NN and LSB 7:0 is used. Address(19:0) = MSB(3:0)NN(7:0)LSB(7:0)

5.3.10 End of Configuration

The command is decoded as x7B. Designates what byte address (relative to the configuration bin file) to stop scrubbing. Address(19:0) = MSB(3:0)NN(7:0)LSB(7:0)

x7B	MSB	NN	LSB

Figure 19: Last Address Scrub Command Format

5.3.11 Set Control Register and Set Mask Register

The command is decoded as x89 (control register setting) and x8A(Mask Register setting). When scrubbing the Xilinx V5 series, the GLUT MASK bit must be set if SRL or Dynamic RAM is used. Byte 0 can not be changed and is hard-coded in the tester (byte 0 corresponds to the LSByte of the register). Care must be taken because although the bytes are LSByte first, the ordering in the byte is MSB first (7:0).

x89	Byte1	Byte2	Byte3

Figure 20: Control and Mask Register Command Format

5.4 Configuring the DUT via the Tester

Because the Xilinx device has SRAM based configuration, the user must write the configuration memory. We have decided to control the configuration from the tester. See Figure 21: Tester Controlled Configuration Flow Diagram for a flow of configuration:

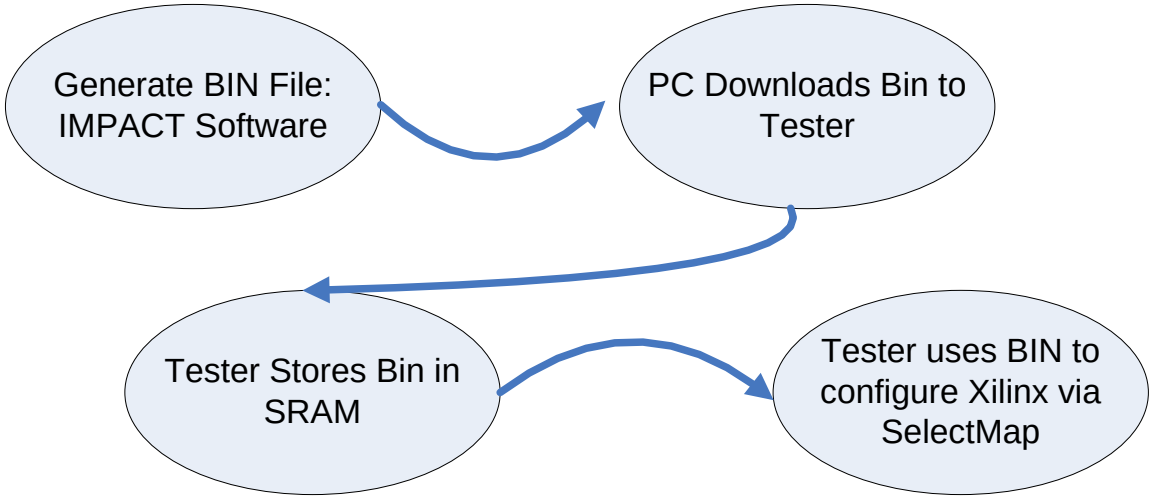


Figure 21: Tester Controlled Configuration Flow Diagram

The tester is connected to SelectMap I/O of the DUT and the mode is set so that the Tester is Master and the DUT is slave. 8 bit SelectMap is used at 30 MHz. Before configuration can be performed, the user must dump the ".bin" file to the tester SRAM. This can be accomplished in 1 of 2 ways:

1. via RS232 cable using the write configuration command: x81 (labview environment). This method uses the computer that is running labview. It is part of the regular command stream.
2. via the QuickUSB set-up including the user interface. This method requires a separate computer that is running the Launch USBmain.exe (NASA/GSFC created QuickUsb GUI). Write data command must be set pointing to the bin file. The software will determine the size of the file and send the correct number of words (16 bit transfers @ 50 MHz).

If using the QuickUSB device, the Tester I/O containing QuickUSB are utilized (connected to the parallel portion of the quickusb device). The serial end of the QuickUSB device is connected to a USB cable that is plugged into the extra computer (PC2). The user must run the Launch USBmain.exe program to invoke the user GUI.

5.5 After Configuration: Running a test

After sending the test parameters and successfully configuring the DUT, the start test command is sent. The beam is turned on after the start test command. Tests are run until a fluence of 1e5 to 1e6 is reached or unrecoverable error is detected (from LabView user interface).

5.5.1 Configuration Memory Static Testing

All Configuration memory sensitivity tests will be performed as static tests. Tests are solely performed on the configuration memory of the DUT. The procedure follows:

1. Reset Tester and DUT
2. Send DUT configuration (*bin) file to Tester from Host PC via USB
3. Send Configure Command to Tester via labview
4. Verify configuration is successful via Host PC – impact software - JTAG
5. Store verify readback file corresponding to test number
6. irradiate DUT
7. stop beam and readback configuration of DUT via Host PC – impact software – JTAG
8. store readback file and note number of bit errors
9. There are 977488 bits in the configuration memory. Therefore the static bit error cross section is calculated as:

$$\sigma_{error} = \frac{\#errors}{977488 * fluence}$$

5.5.2 Clock Sensitivity and SEFI Tests

Two mitigated DUT architectures were utilized for this testing. Clock sensitivity will use DTMR (everything is mitigated except for the clock structures). For all other SEFI sensitivity tests, GTMR mitigation will be used so that all of the functional portion of the DUT will be protected. Therefore, only SEFI type structures are vulnerable and therefore can be measured. The DUT will be configured, scrubbed, and outputs will be monitored during irradiation (observation of potential faults in the clock tree). It is imperative that the flux be maintained at low enough level such that the scrubber will operate at least 10 times bit error rate. The scrubber operates at 25 times per second. Flux rates can be determined during static bit tests by analyzing the resultant bit rates from each irradiation corresponding to a given flux. Bit rates should be maintained less than 2 per second at a given flux rate and energy. The device SEFI error cross section will be determined as follows:

$$\sigma_{error} = \frac{\#errors}{fluence}$$

Figure 22 is a flow diagram of running a clock sensitivity test.

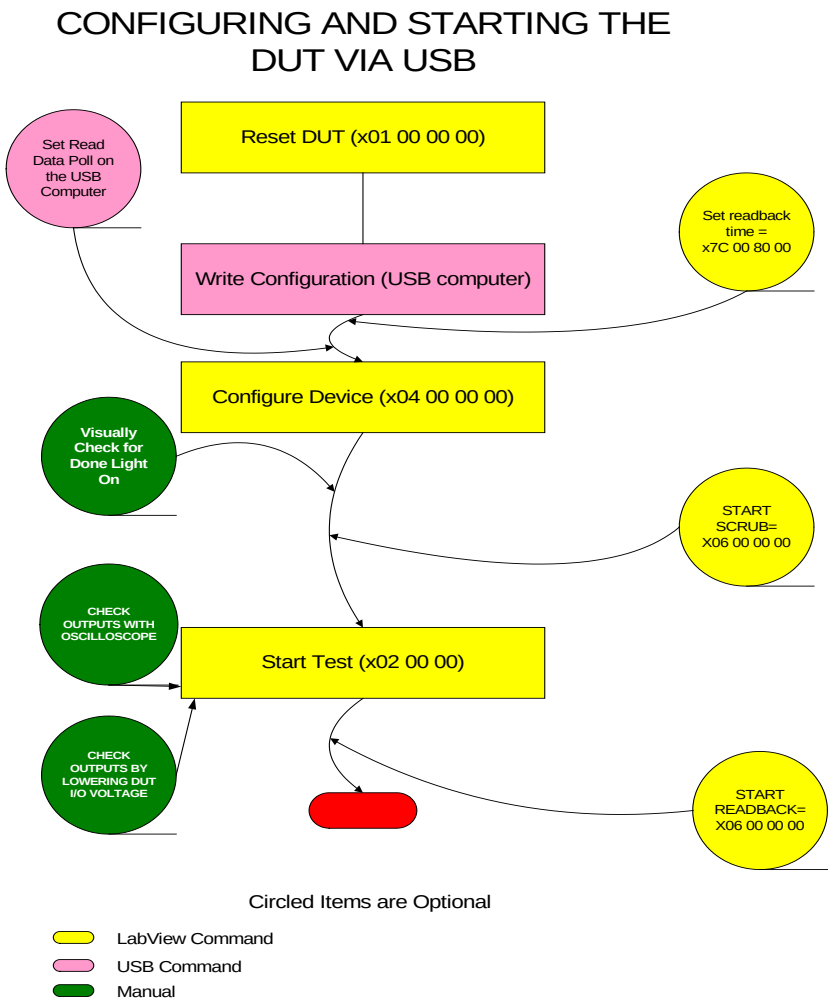


Figure 22: Example of Running a Test using the QuickUSB Device

5.5.3 Speed Degradation

No error cross section is calculated for this test. DUT outputs are monitored for potential dose failure at a given speed…i.e. device will operate at a slower speed but will no loner operate at its maximum speed grade. The procedure follows:

1. Reset Tester and DUT
2. Send DUT configuration (*bin) file to Tester from Host PC via USB
3. Send Configure Command to Tester via labview
4. Verify configuration is successful via Host PC – impact software – JTAG
5. Store verify readback file corresponding to test number
6. Start scrubber
7. Start operation (clocks and data toggle)
8. irradiate DUT and monitor outputs
9. stop beam

10. Keep testing until 150krads (Si) is reached

6. PROTON RADIATION TEST RESULTS

6.1 Configuration Memory

Static Configuration Proton Testing produced virtually constant configuration SEU cross sections across the energy spectrum of 60MeV to 200MeV. This suggests that the configuration is in saturation in this region of proton energy. Results are graphed in Figure 23.

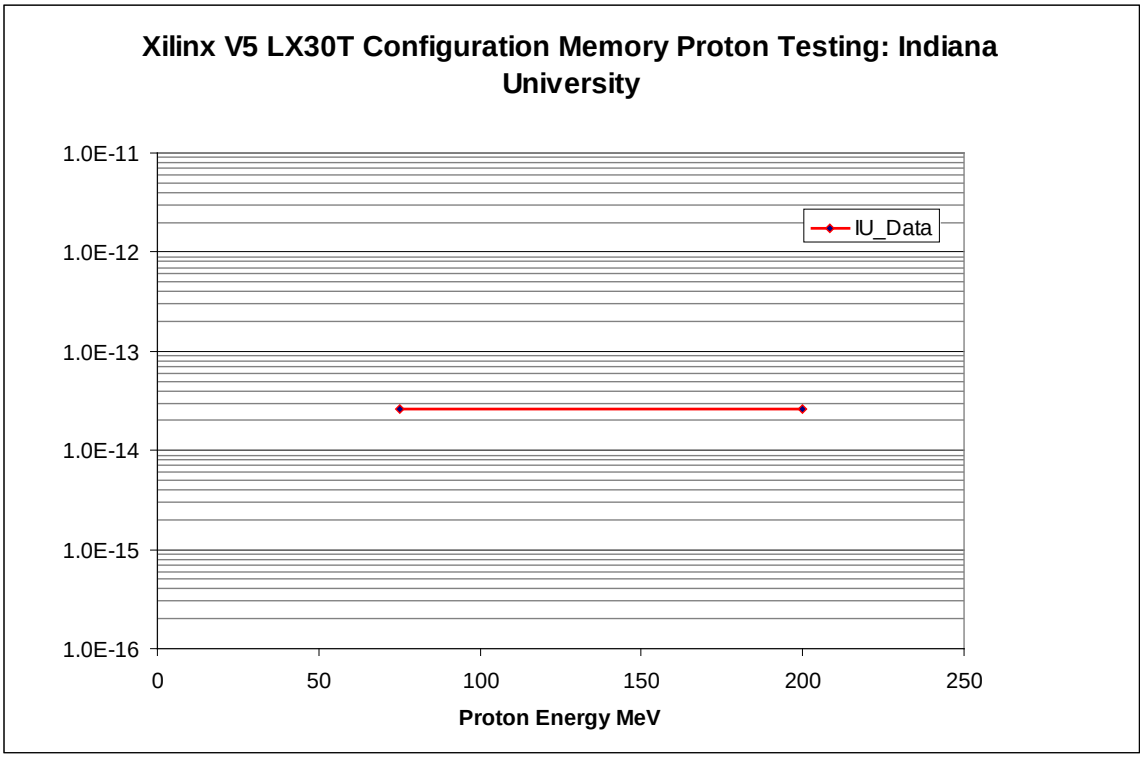


Figure 23:IU Proton Test Data. Shows Configuration SEU Cross Section is in Saturation

Additional testing was performed at UC Davis. Only static configuration testing was performed. This testing verified that the configuration SEU cross section was in saturation with the higher energy Protons. Results are graphed in Figure 24

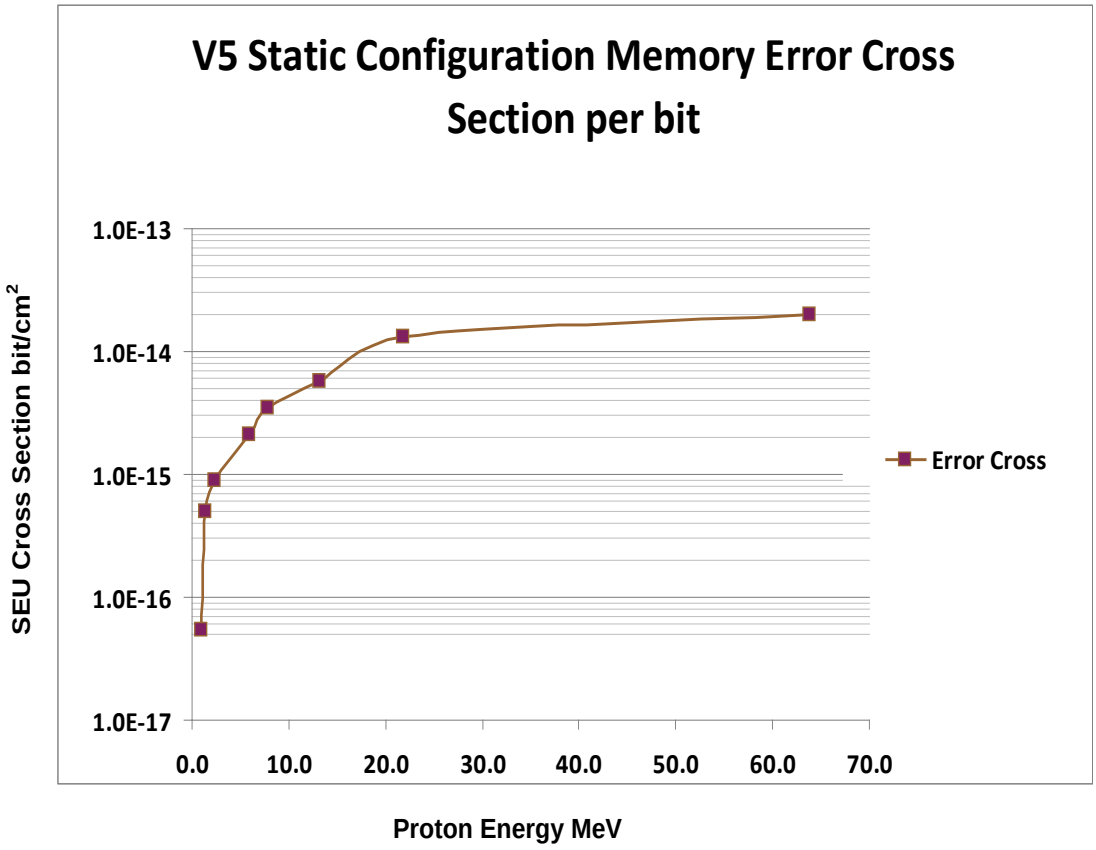


Figure 24: UC Davis Low Energy Proton Data. Configuration Cross Section is not in saturation

6.2 Scrubbing

During dynamic testing, the scrubbing was monitored to verify that the configuration memory was getting overwritten as directed. Otherwise, the scrubber would be considered, broken. Monitoring was accomplished by performing a readback post-irradiation. Significant configuration memory bit upsets indicate a broken scrubber.

Both scrubbers worked flawlessly during irradiation. No upsets within the scrubbing path were observed. Although no upsets were observed during proton testing, one must be cautious because the methods were not 100% effective during heavy ion testing.

6.3 Dynamic Testing of Functional Logic (Shift Registers – DTMR and GTMR)

As previously described, the DTMR and GTMR shift register outputs were observed during testing. No upsets were observed. It is important to note that all three domains were placed within the device such that there were no shared resources (i.e. clb's or routing matrix). If this step of placement separation were not done, then some upsets would be expected (as observed in fault injection tests). Although no upsets were

observed during proton testing, one must be cautious because the methods were not 100% effective during heavy ion testing.

6.4 Clock and SEFI Tests

No Clock upsets or SEFIs were observed during dynamic proton tests

6.5 Speed Degradation

No Speed Degradation was observed during dynamic proton tests

6.6 Appendix 1:

See www.xilinx.com and contact Melanie.D.Berg@gsfc.nasa.gov for further details: