RTAX2000-S Field Programmable Gate Array Single Event Effects (SEE) High-Speed Test - Phase I

Melanie Berg – Principle Investigator MEI
HAK Kim, Mark Friendlich, Chris Perez, Christina Seidlick: MEI
Ken Label: NASA/GSFC

Test Date: 12/8/2009 Preliminary Report Date: 2/25/2010

Test Date: 05/2010

Test Date: 08/2010

Test Date: 12/10 Report Date 1/2011

1. INTRODUCTION

This study was undertaken to determine:

1. The single event destructive and transient susceptibility of the internal memory structures embedded in the RTAXS FPGA family of devices.
2. The single event destructive and transient susceptibility of various designs implemented in the RTAXS FPGA family of devices

The DUTs were configured to have various forms of active embedded memory structures and various combinatorial plus sequential logic test structures. The test circuitry was monitored for Single Event Transient (SET) and Single Event Upset (SEU) induced faults by exposure to heavy ion beam. The purpose of the RTAX-s memory evaluation was to enhance prior testing of the device. The memory study included both static and dynamic modes of memory control. The test structures included memory configurations that utilized the embedded Error Correction and Detection (EDAC) circuitry and separate structures that performed First-In-First-Out (FIFO) operation. The combinatorial plus sequential logic evaluation was also an enhancement to prior radiation testing. The study included a more detailed analysis of transient propagation with variations in capacitance, frequency, data pattern, and Linear Energy Transfer (LET).

2. BACKGROUND

2.1 RTAXs Fabric and Embedded Mitigation

The Actel RTAX-S is a family of high performance, radiation tolerant Field Programmable Gate Arrays (FPGAs). The manufacturer has implemented Single Event Effect (SEE) Radiation Hardened by Design (RHBD) circuitry that is available to the user such as: metal-to-metal Single Event Upset

(SEU) immune configuration [1] ; localized triple mode redundancy (LTMR) at each flip-flop[1][2]; and hardened global routing networks (such as clocks)[1][3]. Because these devices have been selected to control critical circuitry within a variety of space missions subject to harsh radiation environments, they have undergone a significant amount of SEE testing. Prior testing has proven that the LTMR circuitry is beneficial because its implementation decreases SEU probability [1][2]. However, because the triplicated DFF's have shared inputs in the RHBD LTMR implementation (as illustrated in Fig 1), heavy ion-testing has revealed data path susceptibility to Single Event Transients (SETs).

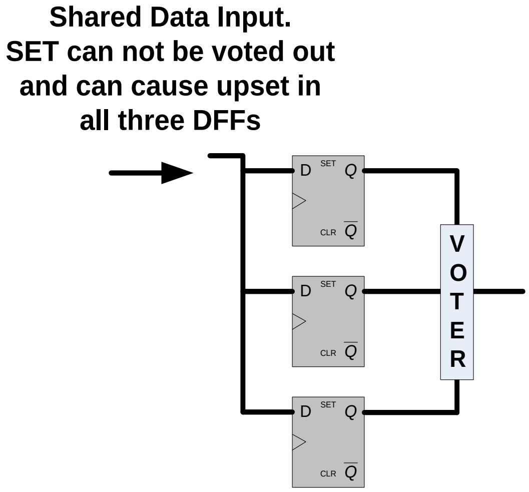

Shared Data Input.
SET can not be voted out and can cause upset in all three DFFs
D
SET
Q
CLR
Q̄
V
O
T
E
R

Fig 1: Localized Triple Mode Redundancy RTAX-S Schematic Level Implementation

Following convention, the bit-error rates (dE_{bit}/dt) published in the Actel datasheets are calculated from shift register SEU characterization. Because the most significant upsets in the RTAX-s device are due to transients captured by DFFs, the bit error rate becomes frequency dependent [2][3]. As a result, designers extrapolate shift register error rates ($dE_{bit}(fs)/dt$) to calculate error rates (dE/dt) of complex circuits as noted in equation (1).

$$\frac{dE}{dt} < \frac{dE_{bit}\left(fs\right)}{dt} * \left(\#UsedDFFs\right)$$

(1)

Shift registers (SRs) are considered to have a linear data path because each node only has one input and one output (i.e. fanout = fanin = 1). Due to the decrease in transistor geometries and capacitive node loading it has come to question if the calculated shift register dE_{bit}/dt can be applied to complex circuits. In other words, will the predicted system error rate be accurate using shift register data as parameters? As an example, counter architectures are not linear. They contain nets with fan-out and fan-in >1. The fan-out will change both the capacitive loading of cells and the utilization of routing resources within FPGA fabrics. Depending on the rise/fall time and width of a Single Event Transient (SET) the capacitive loading of a cell can filter away the SET. On the other hand, a SET that is not filtered can fan-out to multiple nodes and have the effect of a multiple bit upset. Regarding nodes that have fan-out=1 or have minimal capacitive loading, it has been proven that SETs can increase in width as they traverse a circuit. As noted, variations of SET signatures exist and will depend on design topology and operational parameters.

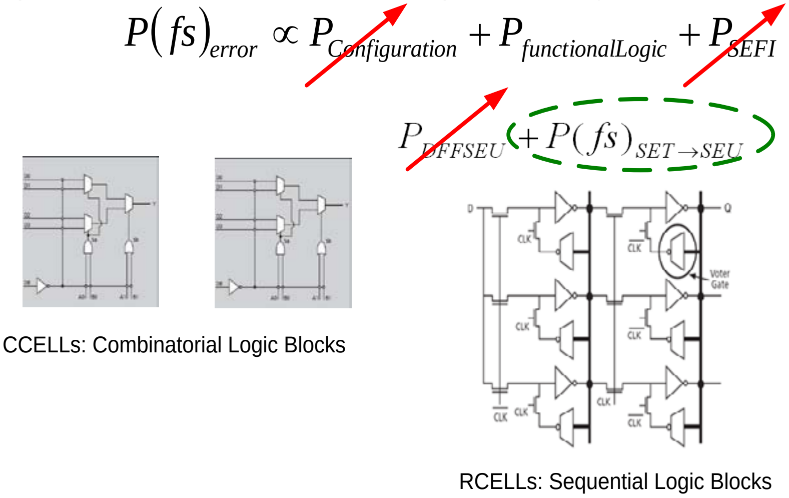
Understanding the various upset event probabilities and their effects are essential when designing critical applications and predicting error rates. As a response, a more in-depth approach to SEE characterization of complex circuits has been performed by NASA Goddard Radiation Effects and Analysis Group (REAG). The study incorporates testing shift registers, counters, and embedded memory structures.

2.2 Probability of SEUs in FPGA Devices

It is shown (2), that the probability of error due to SEEs within FPGA devices has three major factors [3]: Configuration upsets ($P_{Configuration}$), Functional Logic Upsets ($P_{functionalLogic}$), and Probability of Single Event Functional Interrupts (P_{SEFI}).

$$P(fs)_{error} \propto P_{Configuration} + P_{functionalLogic} + P_{SEFI} \qquad \textbf{(2)}$$

Because the configuration of the Actel RTAX-S device is anti-fuse technology, its $P_{Configuration}$ is essentially zero. Actel has hardened the global routes such that P_{SEFI} is low. Theoretically, $P_{functionalLogic}$ has two major components: Probability of flip-flop upsets (P_{SEUDFF}) and the probability Single Event Transients (SETs) being captured by a DFF ($P_{SET\rightarrow SEU}$).


$P(fs)_{error} \propto P_{Configuration} + P_{functionalLogic} + P_{SEFI}$
$P_{DFFSEU} + P(fs)_{SET\rightarrow SEU}$
D
Q
CLK
Voter Gate

CCELLs: Combinatorial Logic Blocks

RCELLs: Sequential Logic Blocks

Fig. 2: RTAX-S Susceptibility. Localized Triple Modular Redundancy (LTMR) will not mitigate SETs that are created by combinatorial logic because the data path to the DFF is shared [2].

2.3 SET Generation, Propagation, and Capture in RTAX-S Devices

Because the RTAX-S uses Local Triple Mode Redundancy (LTMR) [1]-[3] at each DFF, and no other circuitry is mitigated, its most prevalent upset probability factor is $P_{SET\rightarrow SEU}$ as illustrated in Fig. 2. In order for $P_{SET\rightarrow SEU}$ to exist, a transient must be generated (with a probability of $P_{generate)}$, propagated (with a probability of $P_{propagate)}$, and then captured by a destination DFF. In order for the SET to be captured, it must arrive during the destination DFF's clock edge with a probability proportional to the width of the transient multiplied by f_s. Consequently, $P_{SET\rightarrow SEU}$ is frequency dependent and has operational frequency (f_s) as a parameter ($P(f_s)_{SET\rightarrow SEU}$).

$$P(fs)_{SET\rightarrow SEU} \propto \sum_{i=1}^{NumberofCCells} P_{generate(i)} P_{propagate(i)} \tau_{width(i)} fs$$

$$P(\tau_{clk})_{SET\rightarrow SEU} \propto \sum_{i=1}^{NumberofCCells} P_{generate(i)} P_{propagate(i)} \frac{\tau_{width(i)}}{\tau_{clk}}$$

(3)

2.4 Bit Error Rates and LET Threshold

The LET_{th} is a significant parameter of error-rate calculation. Hence, it is important to determine the lowest LET at which SEEs are first observed. As noted in (3), the RTAX-S SEU cross section is frequency dependent. Subsequently, the NASA Goddard Radiation Effects and Analysis Group (REAG) has performed frequency based testing ranging from 1MHz to 160MHz on Windowed Shift Register (WSR) [2][3] strings with a variety of combinatorial logic (CCELLs[1][2]) between DFFs (RCELLs). It is important to note that WSR architectures are essentially equivalent to SR architectures. They differ by their output interfaces. SRs have a serial output based on their last bit while WSRs have a parallel output interface based on their last four bits.

3. DEVICES TESTED

The sample size per device (in this case) was not the focus since they are production- high speed parts with very little variation across the CMOS process. The emphasis was to test variations over the design state space. The devices were manufactured on an advanced 0.15um CMSOS Antifuse Process Technology with 7 layers of metal. The manufacturer is Actel. The December 2009 and May 2010 tests consisted of two RTAX-2000s devices: DUT1 and DUT2. These devices are referenced as A-Series DUT1 and DUT2. The 08/10 test comprised of two RTAX-2000s devices: DUT1 and DUT2. These devices are referenced as B-Series DUT1 and DUT2

The B-Series DUT1 and DUT2 were created due to complications during December 2009 and May 2010 heavy ion testing. The slight differences in the A-Series and B-Series DUTs are described in detail in later sections of this documented. The purpose of the August 2010 trip was to attempt to either replicate or fix the complications observed in the December 2009 and May 2010 tests.

Table 1: Test Dates and Associated Series of DUTs Tested

Date	Series Tested
December 2009	A-Series
May 2010	A-Series
August 2010	B-Series
December 2010	A-Series and B-Series

Lot Date Codes and Chip Markings –
A-Series DUT1 and DUT2:
RTAX2000S
CQ352B 0526
5962-0422101QXC

B-Series DUT1:
RTAX2000S
CQ352B 0311

B-Series DUT2:
RTAX2000SL
CQ352B 0838
5962-0 22105QXC

3.1 DUT1 and DUT2 General Architectures

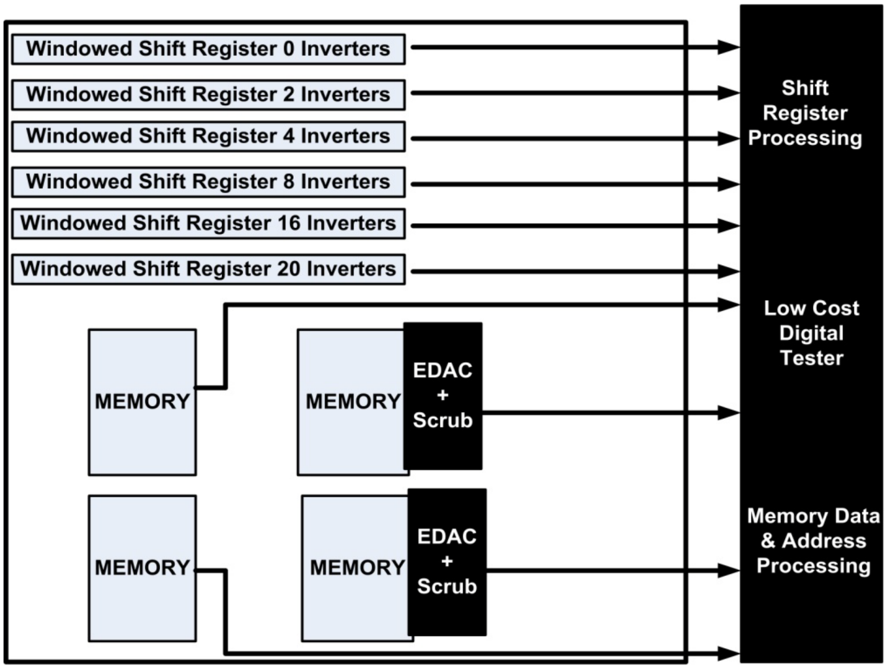

Windowed Shift Register 0 Inverters
Windowed Shift Register 2 Inverters
Windowed Shift Register 4 Inverters
Windowed Shift Register 8 Inverters
Windowed Shift Register 16 Inverters
Windowed Shift Register 20 Inverters
MEMORY
MEMORY
EDAC + Scrub
MEMORY
MEMORY
EDAC + Scrub
Shift Register Processing
Low Cost Digital Tester
Memory Data & Address Processing

Figure 3: DUT1 Designs include 4 memory structures (counted as 2 separate designs) and 6 various shift register strings (counted as 1 design)

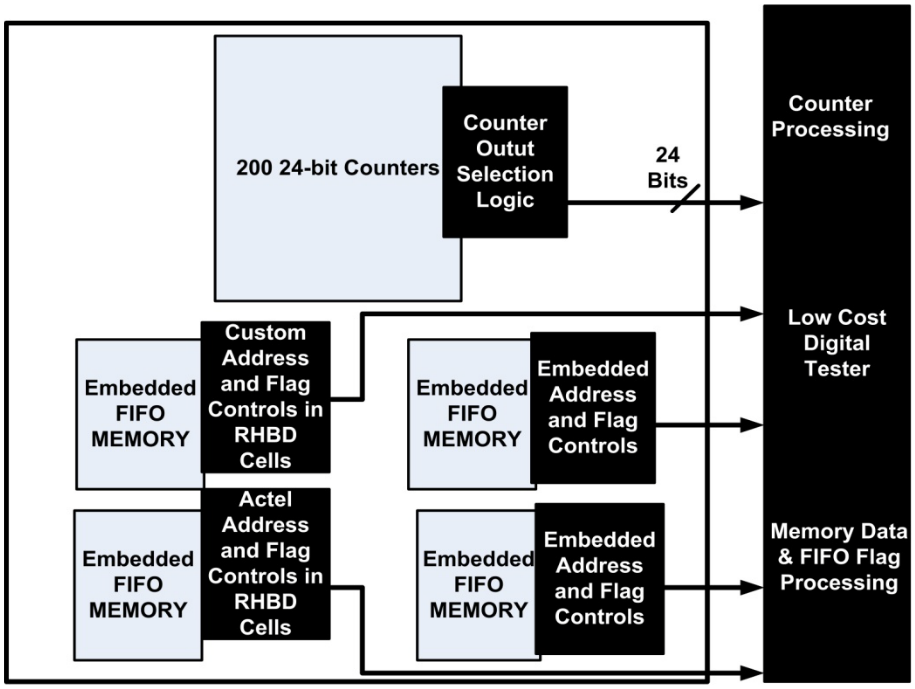

200 24-bit Counters
Counter Outut Selection Logic
24 Bits
Embedded FIFO MEMORY
Custom Address and Flag Controls in RHBD Cells
Embedded FIFO MEMORY
Embedded Address and Flag Controls
Embedded FIFO MEMORY
Actel Address and Flag Controls in RHBD Cells
Embedded FIFO MEMORY
Embedded Address and Flag Controls
Counter Processing
Low Cost Digital Tester
Memory Data & FIFO Flag Processing

Figure 4: DUT2 Designs include 4 FIFO memory structures (counted as 3 separate designs) and 200 counter strings (counted as 1 design).

Because the most significant portion of logic regarding memory access is address control, the analysis of counters was essential to the test set. Several counters were required for testing purposes in order to increase error statistics. As illustrated in Figure 3, DUT 1 contained strings of shift registers and 4 memory blocks. DUT2 contained 4 FIFO memories including strings of counters and is illustrated in Figure 4.

The following are the four designs that were radiation tested:

DUT1:

1. Random Access Memory Designs: Four 4K-bit memory blocks were instantiated. Two out of the four utilized the embedded RTAX-s EDAC and Scrub circuitry. Two out of the four had no EDAC and Scrub circuitry. Each memory block had its own address counter. Circuit monitoring consisted of address and memory bit integrity by the REAG High Speed Digital Tester (HSDT).
2. Shift Registers: 6 Windowed Shift Register (WSR) Chains with varying numbers of inverters were implemented. Original RTAX-s WSR testing was published in [2]. WSR designs and monitoring were repeated in these radiation tests to be utilized as characterization reference points.

DUT2:

3. FIFO Memories: 2 embedded asynchronous FIFO (AFIFO) memories were implemented utilizing embedded FIFO memory address, flag, and general control logic. As previously stated, the embedded control logic is not RHBD. The third embedded AFIFO had custom designed controls that were implemented using RHBD user cells. The fourth embedded FIFO was a Synchronous FIFO (SFIFO) with controls implemented in the RHBD user design space. Empty and full flags for all FIFOs were monitored. The custom AFIFO had the capability to have its read and write address monitored by the HSDT.
4. Counter Arrays: It should be emphasized that the design implemented to test the counters is a novel approach. There exist 200 counters (labeled 0 to 199) in the counter array. The HSDT can determine if an upset occurs in any of the counters and what bits were affected. A more detailed description will be provided in the following sections.

The following sections provide more detailed descriptions for each implemented design.

3.2 DUT 1 Shift Register Architectures (WSRs)

3.2.1 Functional Description

In order to examine basic gate (sequential and combinatorial logic) sensitivity, REAG has chosen a simple (yet enhanced) shift-register architecture as one of the FPGA DUT architecture for radiation testing. The basic shift-register was enhanced by two features: (1) variations of inverter logic between flip-flop stages and (2) Windowed shift-register (WSR). The top-level WSR architecture is illustrated in Figure 5. The implementation of the windowed shift-register allows for reliable high-frequency testing by increasing board level signal integrity and simplifying DUT shift-register output data capture.

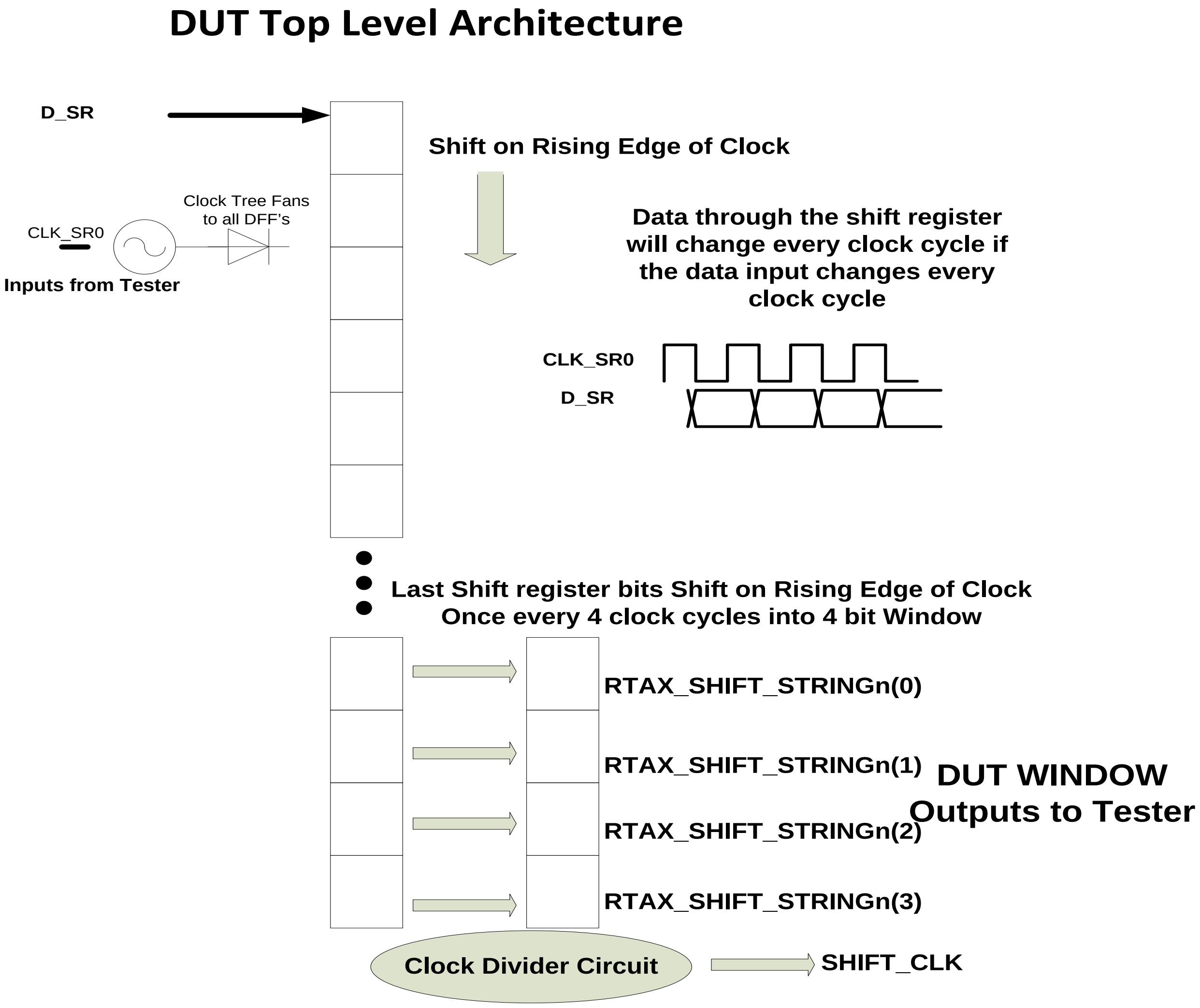

DUT Top Level Architecture
D_SR
Shift on Rising Edge of Clock
Clock Tree Fans to all DFF's
CLK_SR0
Inputs from Tester
Data through the shift register will change every clock cycle if the data input changes every clock cycle
CLK_SR0
D_SR
Last Shift register bits Shift on Rising Edge of Clock Once every 4 clock cycles into 4 bit Window
RTAX_SHIFT_STRINGn(0)
RTAX_SHIFT_STRINGn(1)
RTAX_SHIFT_STRINGn(2)
RTAX_SHIFT_STRINGn(3)
DUT WINDOW Outputs to Tester
Clock Divider Circuit
SHIFT_CLK

Figure 5: WSR Top Level Architecture

3.2.2 Combinatorial Logic and Sequential Logic Elements in the WSR

The Principle Configuration of the WSRs contains 200 to 2000 DFF's with varying levels of combinatorial logic (0, 2, 4, 8, 16, or 20 inverters) between DFF's. The number of combinatorial logic blocks is denoted by N in this document.

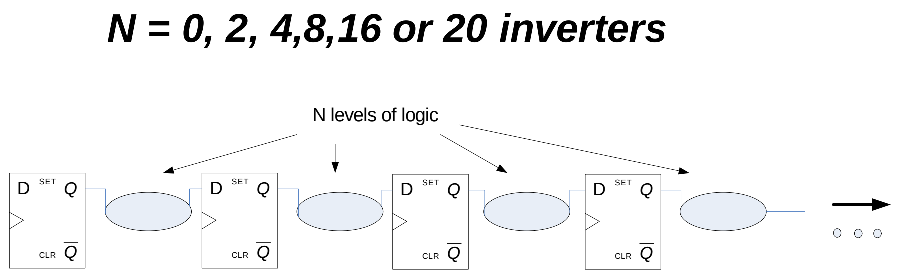


Figure 6: Shift Register String with Optional Combinatorial Logic

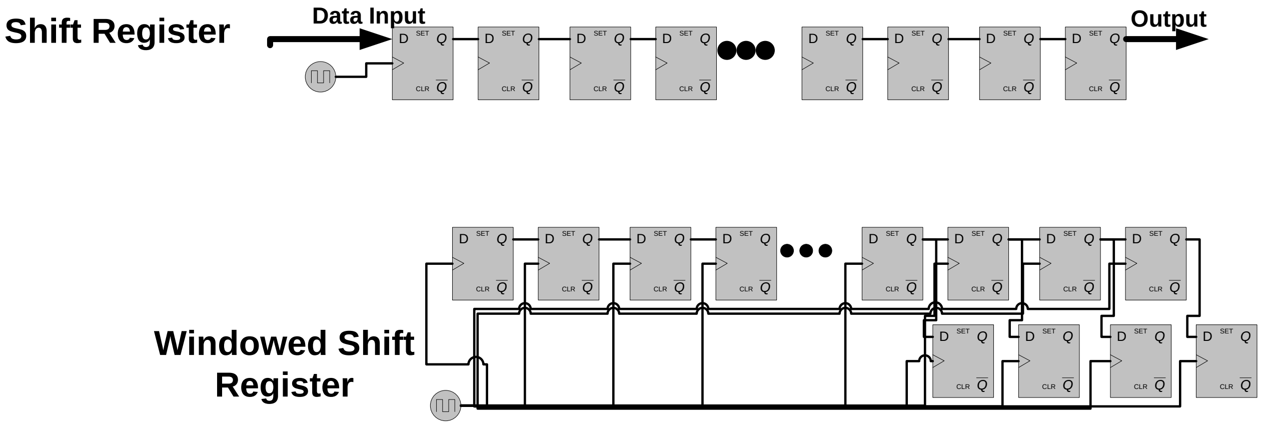


Figure 7: Comparison of a traditional Shift Register to the REAG Windowed Shift Register

Various levels of combinatorial logic are used in order to measure possible transient susceptibility ($P_{SET \rightarrow SEU}$). Table 2 describes the logic resources contained within each shift register chain for the A-Series. Although most of the A-Series WSRs contain various lengths of DFFs, the final SEU cross section is calculated per WSR chains and is normalized by dividing the number of chain-events by the number of DFFs within that chain.

Table 2: A-Seires WSR Chain Logic Elements

Chain Number	Number of CCELLS	Number of DFFs
Chain 0	0 Inverters	2000
Chain 1	2 Inverters	1000
Chain 2	4 Inverters	800
Chain 3	8 Inverters	800
Chain 4	16 Inverters	200
Chain 5	20 Inverters	200

Table 3 describes the logic resources contained within each shift register chain for the B-Series. Given the opportunity to perform additional testing, the B-Series included inverters and buffers within the WSR chains. It was also decided to have the same number of DFFs within each B-Series WSR chain.

Table 3: B-Seires WSR Chain Logic Elements

Chain Number	Number of inverters	Number of DFFs
Chain 0	0	552
Chain 1	0	552
Chain 2	8 Inverters	552
Chain 3	8 Buffers	552
Chain 4	16 Inverters	552
Chain 5	16 Buffers	552

3.2.3 WSR Data Pattern Control

In order to ensure high-frequency synchronous WSR operation, the data input to each of the chains was generated internally to the DUTs. The data pattern is selected from the 2-bit control lines listed in Table 4.

Table 4: Data Pattern Selection

2-bit Select	Data Pattern
00	Static 0
01	Static 1
10	Checker board
11	Checker board

The 2-bit control lines enter into DUT1 from the tester. There are 2 control lines per WSR. In addition, there is a data input generator per WSR. The control line and data generator circuits are illustrated in Figure 8.

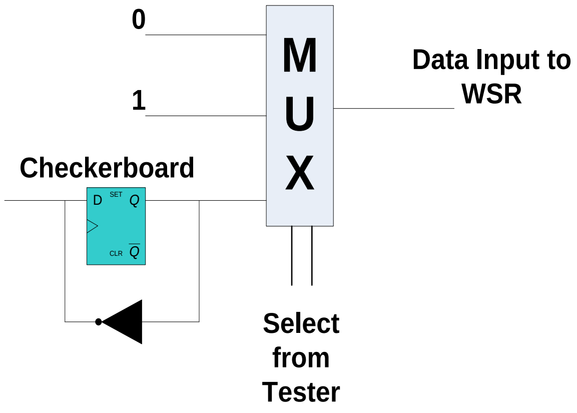


Figure 8: WSR Internal Data Input Circuit

3.2.4 WSR to Tester Interface

Pertaining to Table 5, each WSR chain is supplied a clock (CLK_SR0) and reset (CLR_SR0) from the tester. Based on a pattern select input (RTAX_PATTERN) to the DUT (output from tester), the DUT will calculate the desired WSR data input.

A By-4 clock divider circuit is implemented to shift the last 4 bits of the Shift register string into a DFF window (RTAX_SHIFT_STRINGn). The window is output to the tester. A data clock (RTAX_SHIFT_CLK) is also output to the tester for high speed synchronous data capture.

Table 5: WSR Interface. I/O direction is with respect to Tester

I/O Name	Bits	Dir wrt to DUT	Description
RTAX_SHFT_CLK0	1	OUT	Chain 0 shift Clock
RTAX_SHFT_CLK1	1	OUT	Chain 1 shift Clock
RTAX_SHFT_CLK2	1	OUT	Chain 2 shift Clock
RTAX_SHFT_CLK3	1	OUT	Chain 3 shift Clock
RTAX_SHFT_CLK4	1	OUT	Chain 4 shift Clock
RTAX_SHFT_CLK5	1	OUT	Chain 5 shift Clock
RTAX_SHIFT_STRING0	4	OUT	4-bit windowed output of chain 0
RTAX_SHIFT_STRING1	4	OUT	4-bit windowed output of chain 1
RTAX_SHIFT_STRING2	4	OUT	4-bit windowed output of chain 2
RTAX_SHIFT_STRING3	4	OUT	4-bit windowed output of chain 3
RTAX_SHIFT_STRING4	4	OUT	4-bit windowed output of chain 4
RTAX_SHIFT_STRING5	4	OUT	4-bit windowed output of chain 5
RTAX_PATTERN	12	IN	Each chain gets 2 bits from this vector: chain 0 receives bits (1:0) Chain 1 receives bits (3:2) etc… The bits select which data input pattern (all 0's, 1's, or checker board). See **Error! Reference source not found.** for selection details
RTAX_EN0	1	IN	Sent to chain enable or disable chain 3
RTAX_EN1	1	IN	Sent to chain enable or disable chain 4 and

			5
CLK_SR0	1	IN	Clock to all of the WSR circuitry
CLR_SR0	1	IN	Reset to all of the WSR circuitry

3.2.5 Data Input Interface for WSRs

The possible shift register data patterns listed in Table 4. The selection is controlled by a user command prior to each test. The pattern selection is 2 bits per DUT chain. There are a total of 6 WSR chains; therefore, RTAX_PATTERN is 12 bits wide. Chain 0 receives the 2 LSB of RTAX_PATTERN (bits 1 downto 0), chain 1 receives the next two bits (bits 3 downto 2), while chain 5 receives the 2 MSBs (bits 11 downto 10).

The RTAX_EN0 and RTAX_EN1 control signals are used to disable the shift registers that contain a significant amount of inverters between DFFs while performing high speed tests (>100MHz). Table 6 and Table 7 list the chains that are enable or disabled based on the selected test frequency.

Table 6: DUT1 Enables and Frequency Ranges. X denotes chains are enabled

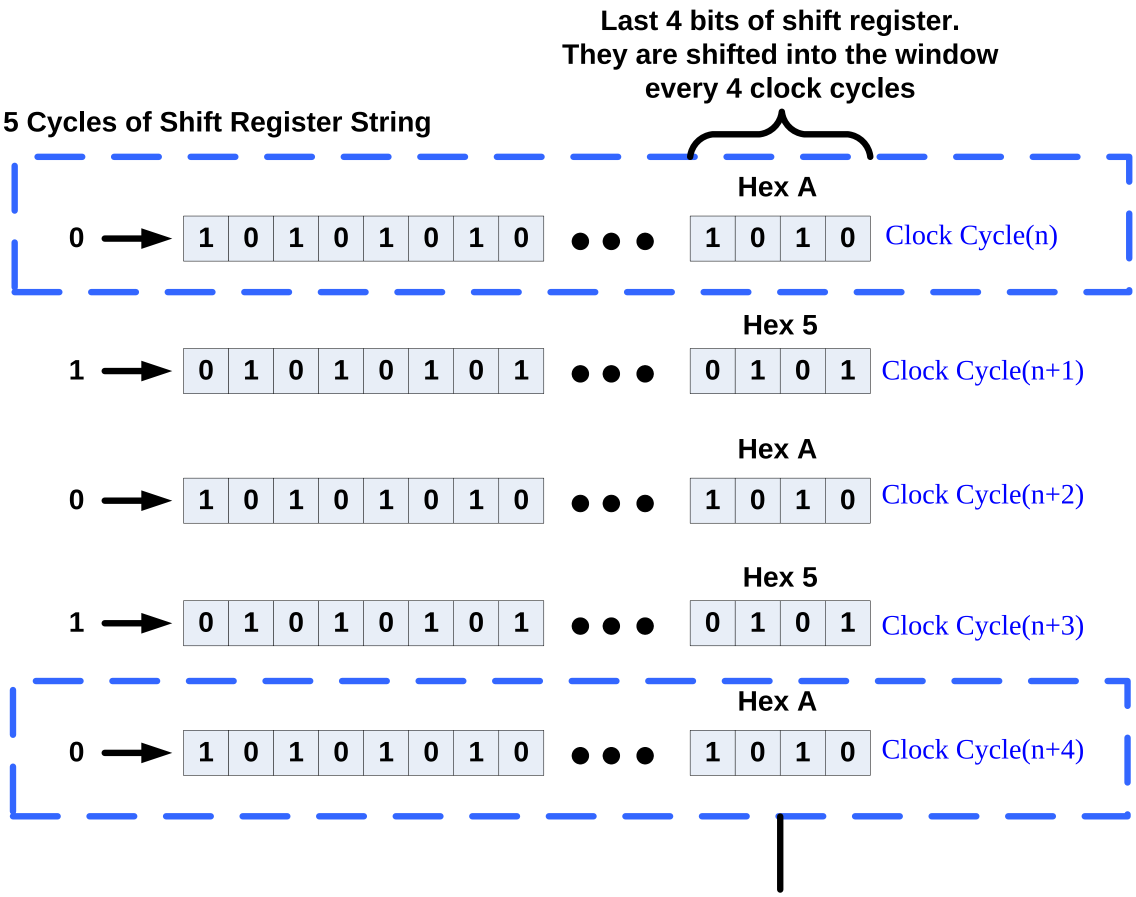
Chain	120MHz-160MHz	80MHz-100MHZ	0-50MHz
N=0	X	x	x
N=2 INV	X	X	X
N=4 INV	DISABLED	X	X
N=8 INV	DISABLED	X	X
N=16 INV	DISABLED	DISABLED	X
N=20 INV	DISABLED	DISABLED	X

Table 7: DUT1 Enables and Frequency Ranges. X denotes chains are enabled

Chain	120MHz-160MHz	80MHz-100MHZ	0-50MHz
N=0	X	X	X
N=8 INV and N=8 BUFF	DISABLED	X	X
N=16 INV and N=16 BUFF	DISABLED	DISABLED	X

3.2.6 WSR Output

Every RTAX_SHFT_CLKn cycle, RTAX_SHIFT_STRINGn is evaluated to determine if an SEE occurred. For a data pattern of all 0’s, the output will be all 0’s. For a data pattern of all 1’s, the output will be all 1’s (after the equivalent number of clock cycles as the length of the shift register). For a checkerboard pattern, the last 4 bits change every clock cycle. Because the WSR window is a snapshot of the last 4 shift register bits every 4 clock cycles, the window stays static (either a hex 5 or a hex A). The operation is illustrated in Figure 9.


Hex A
0 → 1 0 1 0 1 0 1 0 ● ● ● 1 0 1 0 Clock Cycle(n)
Hex 5
1 → 0 1 0 1 0 1 0 1 ● ● ● 0 1 0 1 Clock Cycle(n+1)
Hex A
0 → 1 0 1 0 1 0 1 0 ● ● ● 1 0 1 0 Clock Cycle(n+2)
Hex 5
1 → 0 1 0 1 0 1 0 1 ● ● ● 0 1 0 1 Clock Cycle(n+3)
Hex A
0 → 1 0 1 0 1 0 1 0 ● ● ● 1 0 1 0 Clock Cycle(n+4)

5 Cycles of Shift Register String

Last 4 bits of shift register. They are shifted into the window every 4 clock cycles

Every 4 cycles, the last for bits are equivalent. Therefore the window is static under normal operating conditions

Figure 9: WSR shift register operation for a checker board input. Every 4 clock cycles the last 4 shift register bits are equivalent. Every 4 clock cycles the window gets a snap shot of the last 4 bits of the shift register. Consequently, the window is static under normal operating conditions

3.2.7 WSR Expected Upsets

Because of the WSR structured, the string outputs are expected to be constant after the length of the string cycles following reset de-assertion. Therefore, an error is easily detected by monitoring any change within the WSR outputs as illustrated in Figure 10: Example of WSR SEE DUT output to tester.

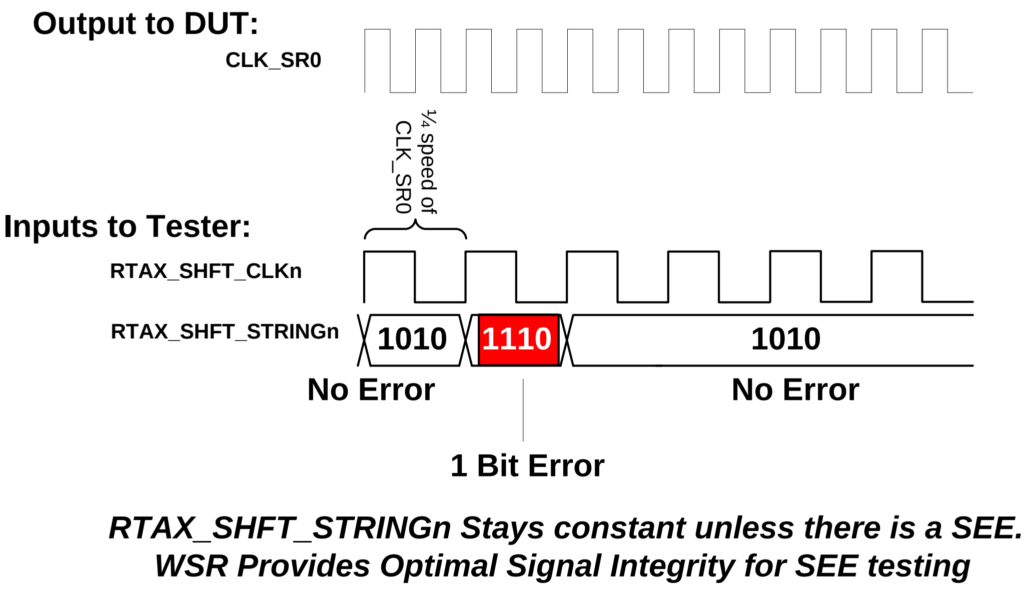


Figure 10: Example of WSR SEE DUT output to tester

Primary Upsets:

- Bit flip in shift register: Will be observed in the window for 4 cycles (because window can only change once every 4 cycles).
- Bit flip in window: Upset will be observed for less than 4 clock cycles
- Output transient: May not be able to distinguish from bit flip in window. However, the window be upset for less than one cycle.
- Global routes: An upset can occur in the clock or reset circuitry or enable circuitry (3 out of the 6 strings have enables).
- Shift_clks can get disrupted or completely stop.

3.2.8 Summary of DUT1 WSR SEE Test Evaluations

Understanding that the RCells (DFFs) of the RTAX-s user fabric have been LTMR'd, the primary source of SEE will be from transients traveling on shared (or non-redundant) paths. Conventional philosophy is that as the clock frequency increases, the probability of transients capture will increase. There are caveats to this belief regarding FPGAs. Such as: (1) FPGAs contain complex routing structures that can filter transients (2) fanout of nets can also filter transients (3) Data signal rates of change can impact LTMR sensitivity. Hence, the objectives of testing the WSR strings are to:

- Determine the impact to the error cross section when adding non-redundant cells that are known to be sensitive to SETs.
- Frequency effect evaluation
- Data Input Pattern (data path signal rate) Evaluation.
- Evaluate global route effects (such as clocks, routes, and enables)

3.3 DUT 1 Memory Structures

Four embedded RTAXs memories are instantiated. Each memory has 4K 8bit memory locations. 2 of the memories employ the embedded EDAC and scrub blocks. Due to a limitation in I/O lines, only 2 memories are output to the tester at once. The memory data output must go through a MUX that is contained within the DUT. The MUX select is controlled by the tester. Although only 2 memories are read at once, all 4 memories are written at one time.

Note that one read cycle is 16 HSDT cycles (i.e. the HSDT directs an address increment once every 16 clock cycles). Therefore, the algorithm for testing is: Take 4k read cycles (16 HSDT clock cycles each) to read all of the EDAC memories (Memory 2 and 3) followed by another 4k read cycles to read the non-EDAC memories (Memory 0 and 1). After the 8K of read cycles, write all memories. Because all memories can be written simultaneously, only 4k write cycles are necessary. See Figure 12 for algorithm flow chart.

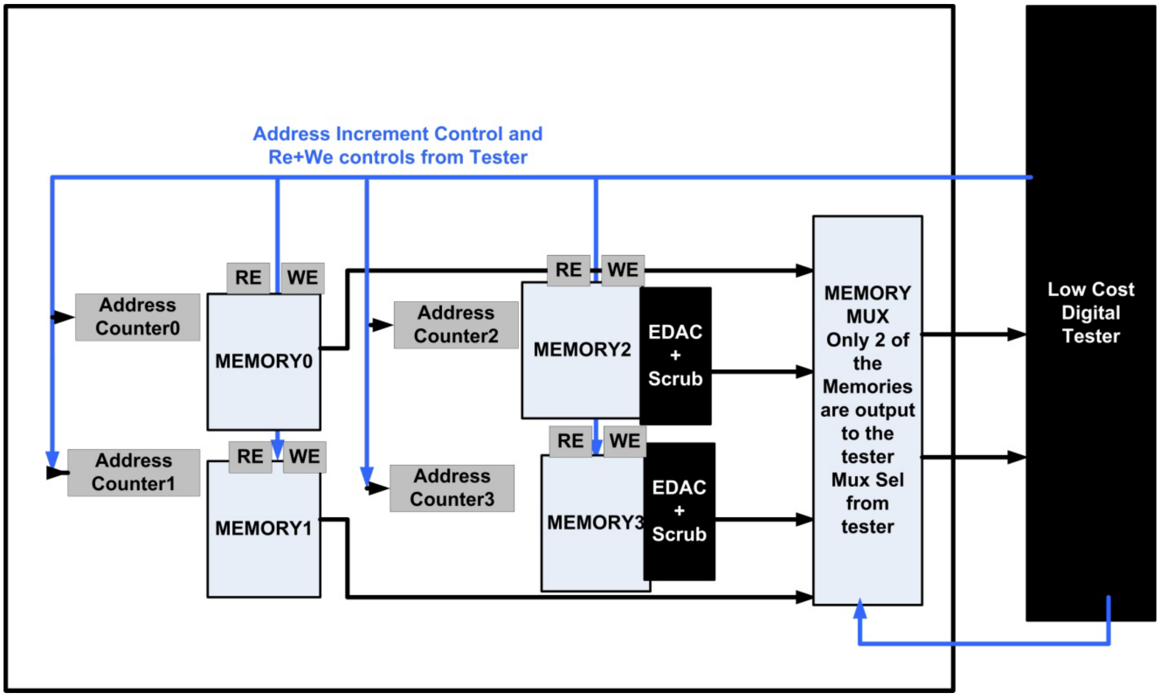


Figure 11: DUT 1 Embedded Memories- 2 with embedded EDAC and 2 without.

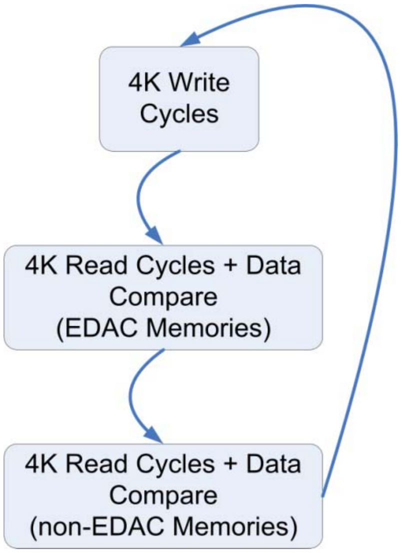

4K Write Cycles
4K Read Cycles + Data Compare (EDAC Memories)
4K Read Cycles + Data Compare (non-EDAC Memories)

Figure 12: Algorithm for Reading and Writing DUT1 Memories.

Table 8: Memory Interface. Control and Data I/O Direction are with respect to the High Speed Digital Tester (HSDT)

I/O Name	Bits	Dir wrt to DUT	Description
RTAX_MEM_WE	1	IN	Write Enable to all Memory
RTAX_MEM_WDATA	8	IN	Write Data to all Memory
RTAX_MEM_RE	1	IN	Read Enable to all Memory
RTAX_MEM_MUX	1	IN	Selects 2 of 4 memory outputs to go to the HSDT for processing
RTAX_MEM_RDATA_A	8	OUT	One of the 2 data inputs to HSDT
RTAX_MEM_ADDR_A	11	OUT	Address input for data A
RTAX_MEM_RDATA_A	8	OUT	One of the 2 data inputs to HSDT
RTAX_MEM_ADDR_A	11	OUT	Address input for data B
RTAX_MEM_Clk	1	IN	Clock used for memory circuitry in DUT
RTAX_MEM_RESET	1	IN	Reset used for memory circuitry in DUT
RTAX_MEM_stop_scrub	1	IN	Directs DUT to not do scrub cycles
RTAX_MEM_BYPASS	1	IN	Bypass EDAC circuitry (only pertinent to the 2 memories that have EDAC)

As previously stated and referring to , the MUX (RTAX_MEM_MUX) selects 2 out of the 4 memory data outputs (RTAX_MEM_RDATA_A and RTAX_MEM_RDATA_B). Each memory component has a localized address counter. The Address counters are also input to the tester 2 at a time via the RTAX_MEM_MUX control for monitoring.

3.3.1 DUT1 Memory Expected Upsets

The primary circuits subject to upset are:

- Memory block SRAM bit upsets: single or multiple SRAM cell faults results in reading corrupted data. Such faults can be fixed by utilizing the appropriate EDAC circuitry
- Address upsets: A bit in the Address generator or register gets flipped. Such an event can cause the wrong addressed memory data to either be written or read. An upset address can be fixed by either a reset or redundant circuitry.
- EDAC or scrub circuitry becoming inactive or invoking inaccurate writes: If this occurs, the device will not be operating within specifications. In addition, the circuitry can be corrupted such that bad data gets scrubbed into memory locations (all data can be lost)
- Read En or Write en faults: An upset in the Read or Write control lines can cause a data word to be missed (miss a read or write cycle). Worse case would be the read or write en circuitry being stuck high or low. A reset or special redundant circuitry can be implemented to correct a control line hit
- Global routes: An upset can occur in the clock or reset circuitry

3.3.2 Summary of DUT1 Block Memory Test Evaluations

Unprotected memory structures are known to have significant SRAM upset rates. RTAX-s provides EDAC circuitry to protect the BRAM memory structures. However, it is important to note that the EDAC structures are not protected and can get upset themselves. An upset in EDAC circuitry can be catastrophic because data can be overwritten in such a way that it is unrecoverable or upset rate sensitivities can become higher than specified with inactive EDAC. Dynamic tests are the only means for uncovering EDAC Single Event Functional Interrupts (SEFIs). This test will be the first to report potential dynamic EDAC SEFI rates.

As previously stated, RCells (DFFs) of the RTAX-s user fabric have been LTMR'd. Subsequently, the primary source of SEE will be from transients traveling on shared (or non-redundant) paths. Address counter and other control logic upsets can cause complete disruption in system activity and are consequently also under investigation. Hence, the objectives of testing the embedded Block Memories are to:

- Verify SRAM bit upset rates are as reported by manufacturer.
- Evaluate potential SEFIs due to EDAC circuitry corruption. Will the embedded EDAC structures meet project specifications or will the designer need to custom design EDAC blocks implemented in the LTMR fabric.
- Determine address and control logic upset susceptibility due to SETs during dynamic operation.
- Evaluate global route effects (such as clocks, routes, and enables) to memory structures

3.4 DUT 2 Counter Array

During SEU testing, it would be ideal to be able to monitor every element of a complex design for every cycle. This is generally not feasible because this would require an output from the DUT to the test vehicle for every observable node. Therefore, more creative designs and interfaces must be developed such that operation during irradiation is unrestricted (fast, continuous, and unobstructed) yet node observation is maximized. Just as important, the tester must be fast enough and robust enough to capture and process the data supplied by the DUT. Processing integrity is very important. Dropped or incorrectly processed data can drastically change error cross sections.

For the SEU testing of the RTAX-s Counters a simple yet effective interface was developed.

3.4.1 Counter Array Implementation

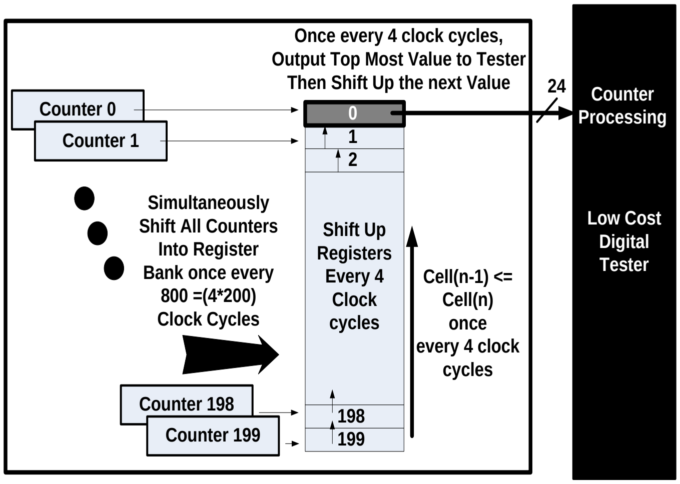

Once every 4 clock cycles,
Output Top Most Value to Tester
Then Shift Up the next Value
Counter 0
Counter 1
0
1
2
24
Counter Processing
Low Cost Digital Tester
Simultaneously Shift All Counters Into Register Bank once every 800 =(4*200) Clock Cycles
Shift Up Registers Every 4 Clock cycles
Cell(n-1) <= Cell(n) once every 4 clock cycles
Counter 198
Counter 199
198
199

Fig 13: Schematic of the 24-bit Counters and their Output Selection Logic. In this case, the output selection logic is a shift register (Shifts up counter values to the output registers every 4 cycles)

The counter array developed by REAG is illustrated in Fig 13. The array contained 200 counters that were 24 bits wide. Because it is impossible to simultaneously output 200 by 24 bits, requiring 48,000 outputs, an output scheme had to be employed that would not compromise the number or speed of the counters yet ensure that each counter is an observable node. Conventional thinking would suggest employing a multiplexer that sequences through the array and selects one of the 200 counters to be output at a time. Unfortunately, the function of the multiplexer, selecting 200 items, requires many levels of combinatorial logic which can be problematic during radiation testing and will slow down the operation of the circuit. Such a large block of logic can potentially mask the primary objective which is characterizing counter SEU susceptibility. Therefore, a novel output methodology had to be established.

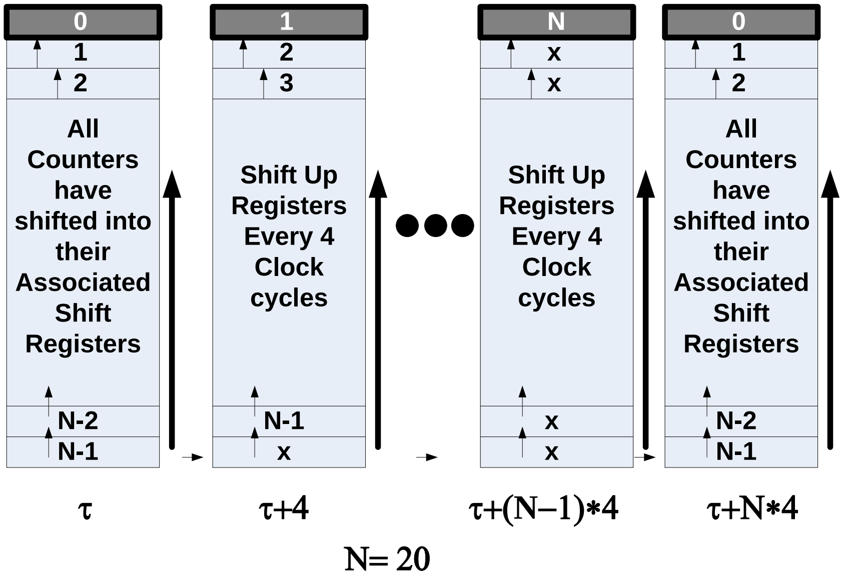

0
1
2
All Counters have shifted into their Associated Shift Registers
N-2
N-1
1
2
3
Shift Up Registers Every 4 Clock cycles
N-1
x
N
x
x
Shift Up Registers Every 4 Clock cycles
x
x
0
1
2
All Counters have shifted into their Associated Shift Registers
N-2
N-1
τ
τ+4
τ+(N−1)∗4
τ+N∗4
N= 20

Fig 14: Counter Shift Register Cycles; Numbers in shift registers represent counter labels at a given moment in time. If there is an x with the shift register, then it is considered a "don't-care" state

As an alternative, a "snap-shot" solution was implemented. With this methodology, each counter is captured simultaneously at a given time into a bank of registers. The number of registers is equivalent to the number of counters (i.e. each counter has its own snapshot register). This is illustrated in Fig 13. The top of the register bank (register 0) is the only register that is accessible by the tester and is 24 bits wide. Subsequently, the DUT to tester interface is simplified.

Fig 13 and Fig 14 illustrate the utilization of the snapshot shift register for each clock cycle. The nomenclature that will be used is as follows:

- n: counter label number
- k: snapshot cycle. First cycle out of reset all of the counter values are snapshot (shifted over) to the shift up register bank. k is 0 for this cycle. The next snapshot of counter values is 800 clock cycles later and k will increment to 1.
- $X_{n,k}$ is the counter-n value that was snapshot into the shift up register for snapshot cycle k.

As previously stated, coming out of reset, each counter has an initial value equal (X_{n0}) to the counter label number (n), e.g. Counter 0 has a reset value of 0 ($X_{0,0}$=0) and counter 199 has a value of 199 ($X_{199,0}$=199). As the circuit comes out of reset, the counter values are loaded into its corresponding register (within the shift-up register bank – see time τ in Fig 14). The counters continue to increment simultaneously as the shift registers shift counter values up every 4 clocks cycles – illustrated in Fig 14. The purpose of the shift-up is so each counter can reach shift register 0 (the output window to the tester). After all counters have been shifted up and loaded into the tester (τ+4N= once every 800 clock cycles), all of the counters are reloaded into the shift register bank.

As a summary, the key of the snapshot output scheme, is that the shift register array has now replaced a huge multiplexer. The benefits are as follows:

1. Counter upsets can easily be identifiable

2. Counters are incrementing and changing state every cycle. Hence maximum performance is able to be tested.
3. If a counter becomes upset, it will stay upset and it will eventually be captured during the snap shot period Counters are continuous and are not interrupted due an elaborate output scheme
4. Routing complexity is exclusive to just the counter array
5. The shift register architecture allows for high speed counter testing. A large multiplexer creates long paths of combinatorial logic and significantly slows down system speed.

The state space of the DUT should be deterministic and traversable. Pertaining to equation 3, for a 24 bit counter running at 25MHz and a shift up period of once every 4 clock cycles, it will take a little less than 1s for every state to be reached for all counters.

$$\frac{2^{24}}{fs} = \frac{1.67\times 10^{7}}{25MHz} = 0.67s$$

(4)

3.4.2 Counter I/O Interface and Expected Outputs

Table 9: DUT1 WSR Outputs

I/O Name	Bits	Dir wrt to DUT	Description
RTAX_Counter_CLK0	1	OUT	Chain 0 shift Clock
RTAX_COUNTER	24	OUT	Chain 1 shift Clock
CLK_SR0	1	IN	Clock to all of the WSR circuitry
CLR_SR0	1	IN	Reset to all of the WSR circuitry

The DUT receives a clock and a reset from the tester. The expected output (RTAX_COUNTER) is purely an increment by 1 starting at value 0 as illustrated in Figure 15. The first RTAX_COUNTER will pertain to counter 0, followed by counter 1, counter 2… up to counter 199. A new snap shot is performed and RTAX_COUNTER will restart by outputting counter 0.

With respect to the separate counters, Counter (n) is output every 800 cycles (800 cycles= 1 snapshot cycle). Therefore the counter values that represent each counter will increment by 800 for each snapshot cycle.

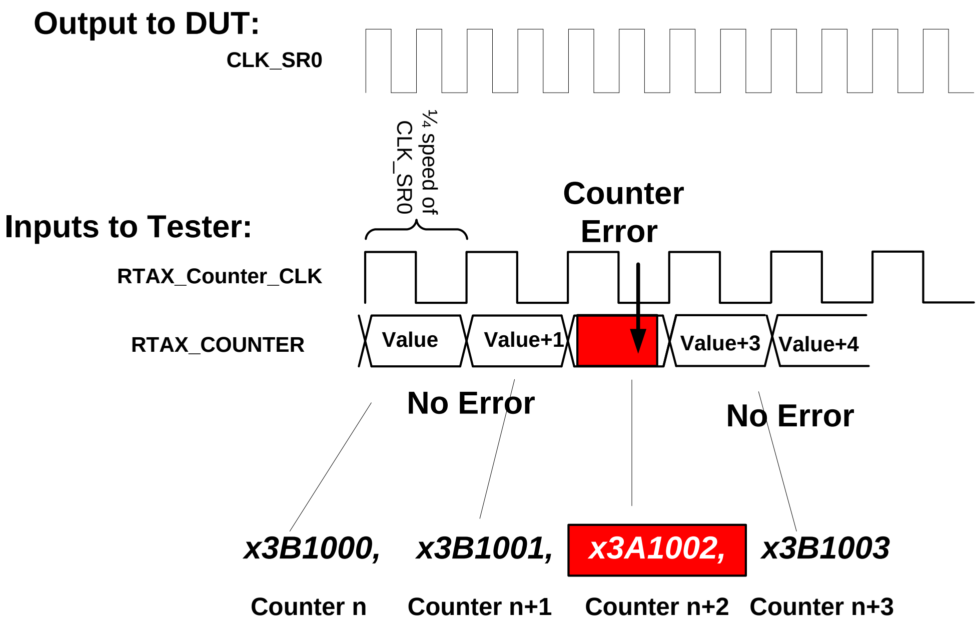


Figure 15: Typical SEE Counter Outputs. Each output represents a value from a different counter in the array. Counter selection is sequential, hence, the counter number and the counter values all increment by 1 each RTAX_Counter_Clk cycle.

3.4.3 Counter Expected Upsets

- Bit or multiple bit upset: This example is illustrated in Figure 15. If a bit flips, it will stay flipped however, the counter will still increment.
- Broken counter: Counter stops counting and its value either stays constant or becomes complete noise
- Shift register:
  - A bit can get flipped while the counter value is in the shift register. In this case, the expected counter value will only be upset for one cycle.
  - Shift register can either stop shifting- in this case the output values will remain constant
  - Shift register can skep a cycle, in this case the counter values will be off the number of skipped shift cycles from their expected values
- Global routes: An upset can occur in the clock or reset circuitry

3.4.4 Summary of DUT2 Counter Array Test Evaluations

Understanding that the RCells (DFFs) of the RTAX-s user fabric have been LTMR’d, the primary source of SEE will be from transients traveling on shared (or non-redundant) paths. Conventional philosophy is that as the clock frequency increases, the probability of transients capture will increase. There are caveats to this belief regarding FPGAs. Such as: (1) FPGAs contain complex routing structures that can filter transients (2) fanout of nets can also filter transients (3) Data signal rates of change can impact LTMR sensitivity. Hence, the objectives of testing the counter array strings are to:

- Determine the impact to the error cross section when implementing complex structures with fanout. SET filtration and propagation will be heavily analyzed
- Frequency effect evaluation

- Can conventional SEE error rates derived from shift register radiation data be applicable to complex structures?
- Because most memory address control circuitry are counters, understanding counter upset susceptibility can directly be applied to memory address control logic.

3.5 DUT 2 FIFO Memories

Figure 16 is a typical schematic of an asynchronous FIFO (AFIFO) architecture. The RTAXs contains embedded FIFO controllers that can operate as an AFIFO when connected with its embedded memory. FIFO controller logic includes the blocks listed in Figure 16. A caveat of utilizing the FIFO embedded controller logic is that the control blocks are not hardened logic (i.e. it does not utilize the RHBD LTMR logic). The sensitivity of various AFIFO implementations was required to be evaluated. A comparison of custom RHBD AFIFO control versus the embedded AFIFOs was the focus of the SEE tests.

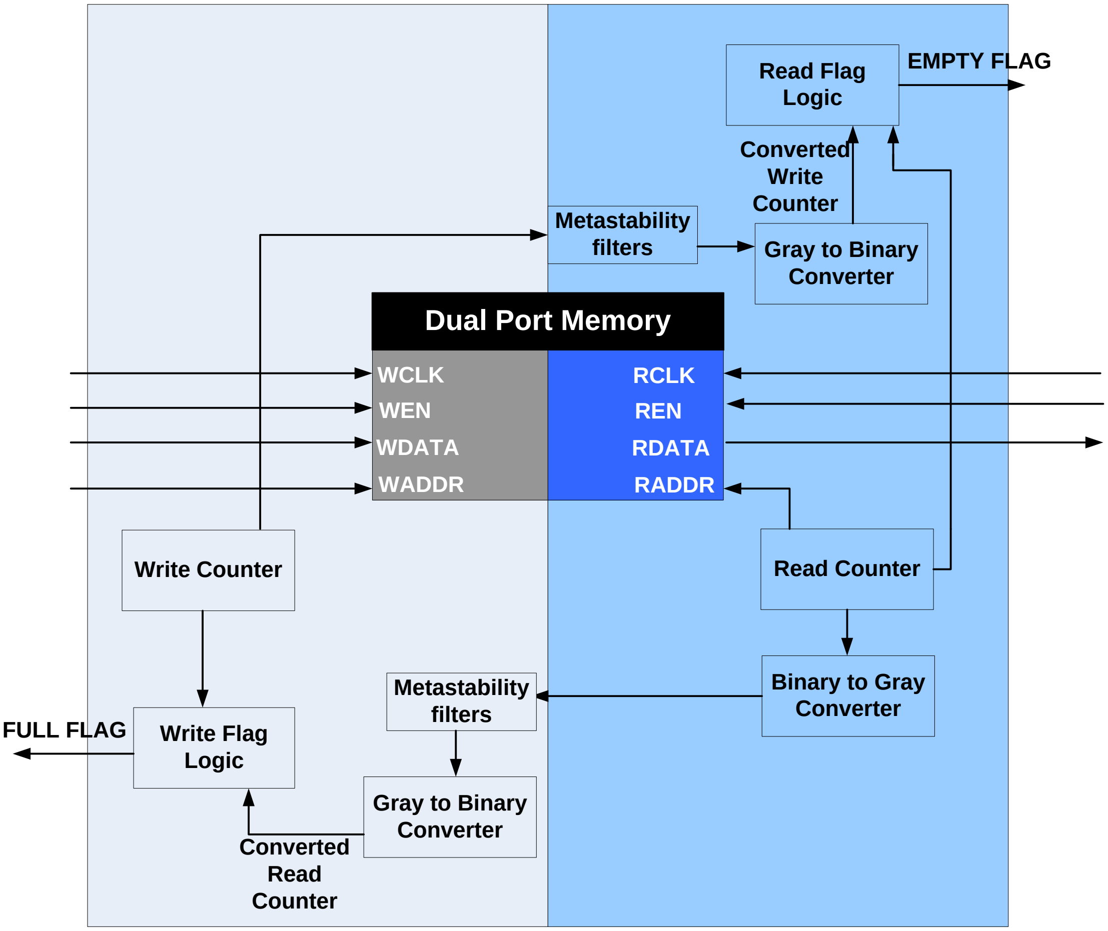


Figure 16: Typical FIFO Architecture. The gray-to-binary and binary-to-gray converters are solely used for address compares to generate EMPTY and FULL flags. The FULL or EMPTY flags can disable memory access if active.

As previously stated, DUT2 contained 2 embedded asynchronous FIFOs (AFIFO) utilizing embedded FIFO memory address, flag, and general control logic. The third AFIFO had custom designed controls that were implemented using RHBD user cells. The fourth embedded FIFO was a Synchronous FIFO (SFIFO) with controls implemented in the RHBD user design space utilizing the RTAX FIFO software. It is important to note that the RTAX FIFO generation software can only create reliable synchronize FIFOs (SFIFOs). The generated AFIFOs are unreliable and should not be used terrestrially or in critical space applications. Therefore, DUT2 has 3 AFIFOs and 1 SFIFO.

Table 10: DUT2 FIFO Implementations

FIFO Name	Implementation
FIFOA	Unhardened Embedded AFIFO
FIFOB	REAG custom FIFO implemented in RHBD cells
FIFOC	Unhardened Embedded AFIFO
FIFOD	RTAX software generated SFIFO implemented in RHBD cells

Table 11: DUT2 FIFO to Tester Interface. I/O direction is with respect to the Tester

I/O Name	Bits	Dir wrt to DUT	Description
RTAX_MEM_WE	1	IN	Write Enable to all Memory
RTAX_MEM_WDATA	8	IN	Write Data to all Memory
RTAX_MEM_RE	1	IN	Read Enable to all Memory
RTAX_MEM_RDATA_A	8	OUT	Read data from FIFOA
RTAX_MEM_RDATA_B	8	OUT	Read data from FIFOB
RTAX_MEM_RDATA_C	8	OUT	Read data from FIFOC
RTAX_MEM_RDATA_D	8	OUT	Read data from FIFOD
RTAX_MEM_RADDR	12	OUT	Read Address from FIFOB
RTAX_MEM_WADDR	12	OUT	Write Address from FIFOB
RTAX_MEM_RClk	1	IN	FIFO Read Clock
RTAX_MEM_WClk	1	IN	FIFO Write Clock
RTAX_MEM_RESET	1	IN	Reset used for memory circuitry in DUT2
RTAX_MEM_EMPTY_A	1	OUT	FIFO A Empty Flag
RTAX_MEM_EMPTY_B	1	OUT	FIFO B Empty Flag
RTAX_MEM_EMPTY_C	1	OUT	FIFO C Empty Flag
RTAX_MEM_EMPTY_D	1	OUT	FIFO D Empty Flag
RTAX_MEM_FULL_A	1	OUT	FIFO A FULL Flag
RTAX_MEM_FULL_B	1	OUT	FIFO B FULL Flag
RTAX_MEM_FULL_C	1	OUT	FIFO C FULL Flag
RTAX_MEM_FULL_D	1	OUT	FIFO D FULL Flag

3.5.1 FIFO Address Monitoring

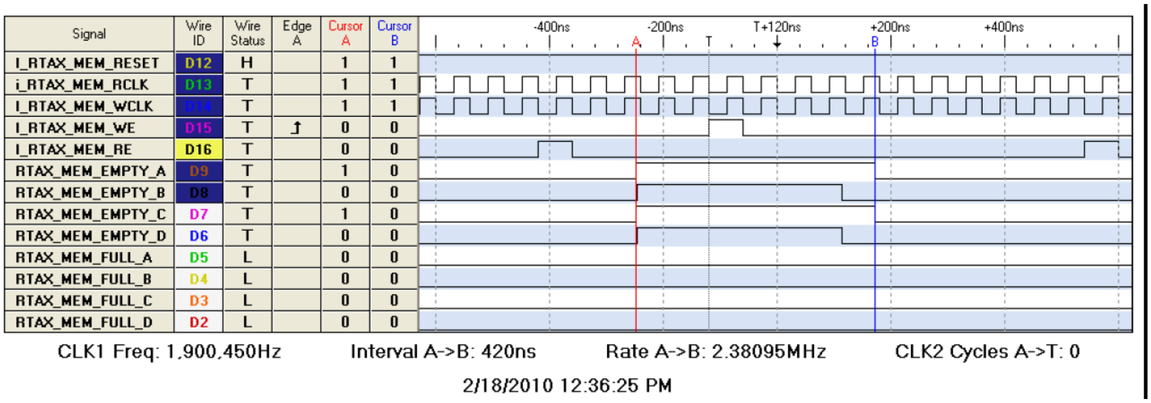
With the embedded AFIFOs and the RTAX software generated SFIFO, the address lines are not accessible – i.e. the FIFO controls are considered Black Box circuits. Because FIFOB is a custom designed AFIFO, the address lines (RTAX_MEM_RADDR and RTAX_MEM_WADDR) were accessible and were subsequently sent to the tester for SEE monitoring. The monitoring consisted of verifying the addresses were always at expected values.

3.5.2 *FIFO Clocks*

Although the AFIFOs take separate clocks (and have all the synchronizers that AFIFO require), the write and read clocks (RTAX_MEM_WClk and RTAX_MEM_RClk) were generated from the same source in the tester and were the same frequency.

3.5.3 *FIFO Empty and Full Flags*

Empty and full flags for all FIFOs were monitored to ensure were invoked when expected. Figure 17 and Figure 18 illustrate empty and full flag assertions relative to corresponding enables.

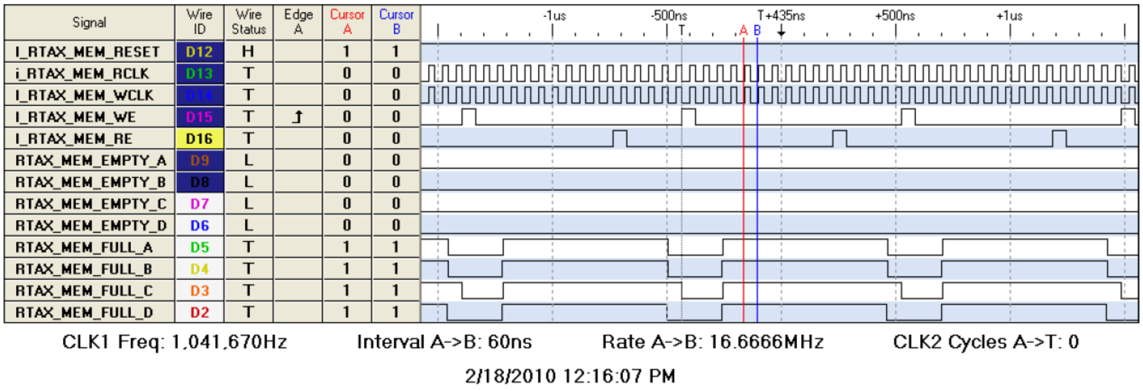


Signal	Wire ID	Wire Status	Edge A	Cursor A	Cursor B
I_RTAX_MEM_RESET	D12	H		1	1
i_RTAX_MEM_RCLK	D13	T		1	1
I_RTAX_MEM_WCLK	D14	T		1	1
I_RTAX_MEM_WE	D15	T	↑	0	0
I_RTAX_MEM_RE	D16	T		0	0
RTAX_MEM_EMPTY_A	D9	T		1	0
RTAX_MEM_EMPTY_B	D8	T		0	0
RTAX_MEM_EMPTY_C	D7	T		1	0
RTAX_MEM_EMPTY_D	D6	T		0	0
RTAX_MEM_FULL_A	D5	L		0	0
RTAX_MEM_FULL_B	D4	L		0	0
RTAX_MEM_FULL_C	D3	L		0	0
RTAX_MEM_FULL_D	D2	L		0	0

CLK1 Freq: 1,900,450Hz Interval A->B: 420ns Rate A->B: 2.38095MHz CLK2 Cycles A->T: 0

2/18/2010 12:36:25 PM

Figure 17: Empty Flags per FIFO under normal operating conditions (no SEE)



Signal	Wire ID	Wire Status	Edge A	Cursor A	Cursor B
I_RTAX_MEM_RESET	D12	H		1	1
i_RTAX_MEM_RCLK	D13	T		0	0
I_RTAX_MEM_WCLK	D14	T		0	0
I_RTAX_MEM_WE	D15	T	↑	0	0
I_RTAX_MEM_RE	D16	T		0	0
RTAX_MEM_EMPTY_A	D9	L		0	0
RTAX_MEM_EMPTY_B	D8	L		0	0
RTAX_MEM_EMPTY_C	D7	L		0	0
RTAX_MEM_EMPTY_D	D6	L		0	0
RTAX_MEM_FULL_A	D5	T		1	1
RTAX_MEM_FULL_B	D4	T		1	1
RTAX_MEM_FULL_C	D3	T		1	1
RTAX_MEM_FULL_D	D2	T		1	1

CLK1 Freq: 1,041,670Hz Interval A->B: 60ns Rate A->B: 16.6666MHz CLK2 Cycles A->T: 0

2/18/2010 12:16:07 PM

Figure 18: Full Flags per FIFO under normal operating conditions (no SEE)

3.5.4 *Expected Upsets to FIFO Hardware*

The primary areas subject to upset are:

- Memory block SRAM bit upsets

- Address upsets
- FULL and Empty flag generation disabled or incorrectly asserted (a result of converter logic upsets or stuck DFFs)
- Read En or Write en faults
- Global routes: An upset can occur in the clock or reset circuitry

3.5.5 Summary of DUT2 FIFO Block Memory Test Evaluations

The SRAM cell upset rates are considered to be evaluated with DUT1 memory tests. RTAX-s provides FIFO circuitry to the dual port BRAM memory structures as “elastic” – “First In First Out” elements. Tt is important to note that the embedded FIFO structures are not protected and can get upset themselves. An upset in the embedded circuitry can be catastrophic because full data sets can be lost due to FIFO control SEEs. Dynamic tests are the only means for uncovering FIFO SEFIs. This test will be the first to report potential dynamic FIFO SEFI rates. As a reference point, REAG has developed their own FIFO that is implemented in the LTMR’d fabric. The following are potential susceptibility that is under investigation for FIFO tests:

- Evaluate potential SEFIs due to FIFO circuitry corruption. Is the FIFO implemented in the LTMR’d fabric significantly harder than the embedded FIFO? Will the embedded FIFO upset rate meet project specifications?
- Determine address and control logic upset susceptibility due to SETs during dynamic operation.
- Evaluate global route effects (such as clocks, routes, and enables) to memory structures

3.6 Global Routes

Global routes are used for high-fan-out nets. Actel provides two types of global routes:

1. HCLK: ultra-low skew global used only for clocks. HCLK can only connect to a clock pin of a DFF. All DFF clocks in A-Series and B-Series are connected to HCLK. All DFF’s in A-Series and B-Series are synchronous and contain a rising edge-triggered clock connect.
2. CLKINT: low-skew global net. Can be used for clocks but best used for resets and other high-fanout nets.

3.6.1 High Fanout Clocks and Resets

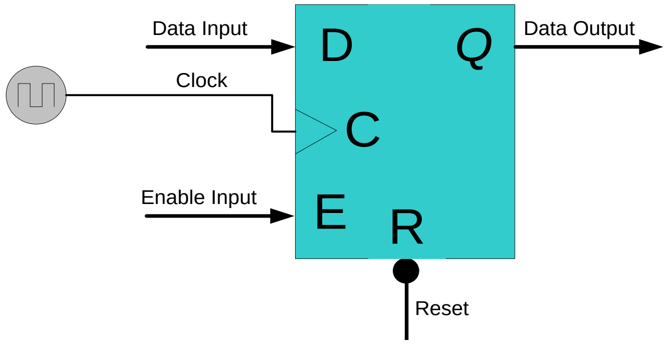


Figure 19: DFF with Data Input (D), Clock Input (C), Enable Input (E), Reset (R), and Data Output (Q)

Figure 19 illustrates the pins to the DFFs utilized in the DUT designs.

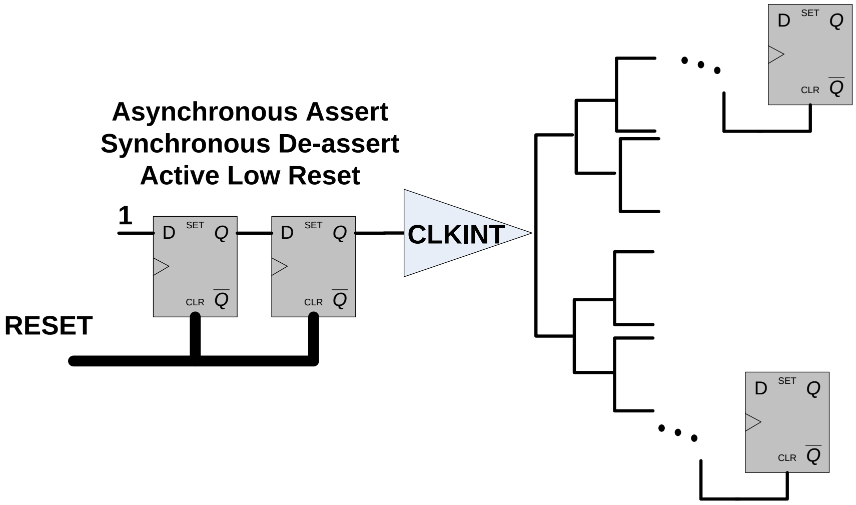


Figure 20: Asynchronous Assert Synchronous Assert Reset Connected to the high Fanout CLKINT. CLKINT is Routed to reset pins of all DFFs. Although not indicated in the figure, all DFFs clocks are connected to the HCLK.

WSRs: All WSRs contain a reset. Consequently this requires the reset to be routed to thousands of DFFs. B-Series WSR resets are connected to the CLKINT tree as illustrated in Figure 20. A-Series did not take advantage of the CLKINT routing structures. Hence the reset tree was created out of a large number of CCELL buffers in order to reach the reset pins of thousands of DFFs.

Counters Arrays: A-Series and B-Series resets to all DFFs in the counter array design (counters + snapshot bank) are connected to the CLKINT tree as illustrated in Figure 20.

3.6.2 High Fanout Enable Nets that Require Global Routes per Design

WSRs: Some WSR strings contain a large number of CCELLs between DFFs. This architecture slows down the maximum operational frequency due to the delay from a DFF through the CCELLs to the next DFF in the chain. Consequently, Enables were placed on all WSR chains with N>0. When operating at the highest frequencies, the N>0 chains were disabled. This requires an enable fanout to thousands of DFFs. If the high fanout enable is not placed on the special CLKINT tree then a large tree of buffers is created out of CCELLs to accomplish the routing. This was the case for the A-Series WSRs. The B-Series WSR enables took advantage of the CLKINT tree for its high fanout enables.

Counter Arrays: All counters are always enabled (if not in reset). Hence no high fanout enables are connected to the counters. However, the snapshot register requires high fanout enables. There are 4800 bits in the snapshot bank of registers. Consequently, the enable is required to be routed to 4800 DFFs. As with the WSRs, the A-Series created a high fanout enable tree out of CCELLs (utilized a significant number of CCELLs). The B-series took advantage of the CLKINT, hence required no additional CCELL blocks.

No Memory structures required high fanout enables.

Table 12: Global Route Utilization

	Clocks	Resets	High Fanout Enables
A-Series DUT1 WSRs	HCLK	Not on global	Not on global
A-Series DUT1 Memories	HCLK	CLKINT	CLKINT
A-Series DUT2 Counters	HCLK	CLKINT	Not on global
A-Series DUT2 Memories	HCLK	CLKINT	N/A
B-Series DUT1 WSRs	HCLK	CLKINT	CLKINT
B-Series DUT2 Counters	HCLK	CLKINT	CLKINT
B-Series DUT2 Memories	HCLK	CLKINT	CLKINT
B-Series DUT2 Memories	HCLK	CLKINT	N/A

A change was made to the B-Series tests such that all resets (DUT1 and DUT2) are placed on the CLKINT tree and all high fan-out enables are also placed on the CLKINT tree.

4. HIGH SPEED DIGITAL TESTER (HSDT) TEST VEHICLE

A-Series and B-Series share the same HSDT test vehicle. The following sections describe the construction of the HSDT including communication interfaces with the DUT and user PCs.

4.1 Architectural Overview

The RTAX-S controller/processor is instantiated as a sub component within the High Speed Digital Tester (HSDT). The HSDT consists of a Mother Board (FPGA Based Controller/Processor) and a daughter board (containing DUT and its associated necessary circuitry). The socket within the DUT Daughter board can accommodate the RTAX1000S and RTAX2000S devices. The objective of this DUT Controller/processor is to supply inputs to the RTAX-S ACTEL Device and perform data processing on the outputs of the RTAX-S. The HSDT communicates with a user controlled PC. The user interface is LAB-VIEW. It will send user specified commands to the mother board and receive information from the mother board. Please see Documents: “HSDT” and “General Tester” for further information concerning the HSDT functionality. The HSDT is connected to the RTAX DUT as shown in the following Block Diagram.

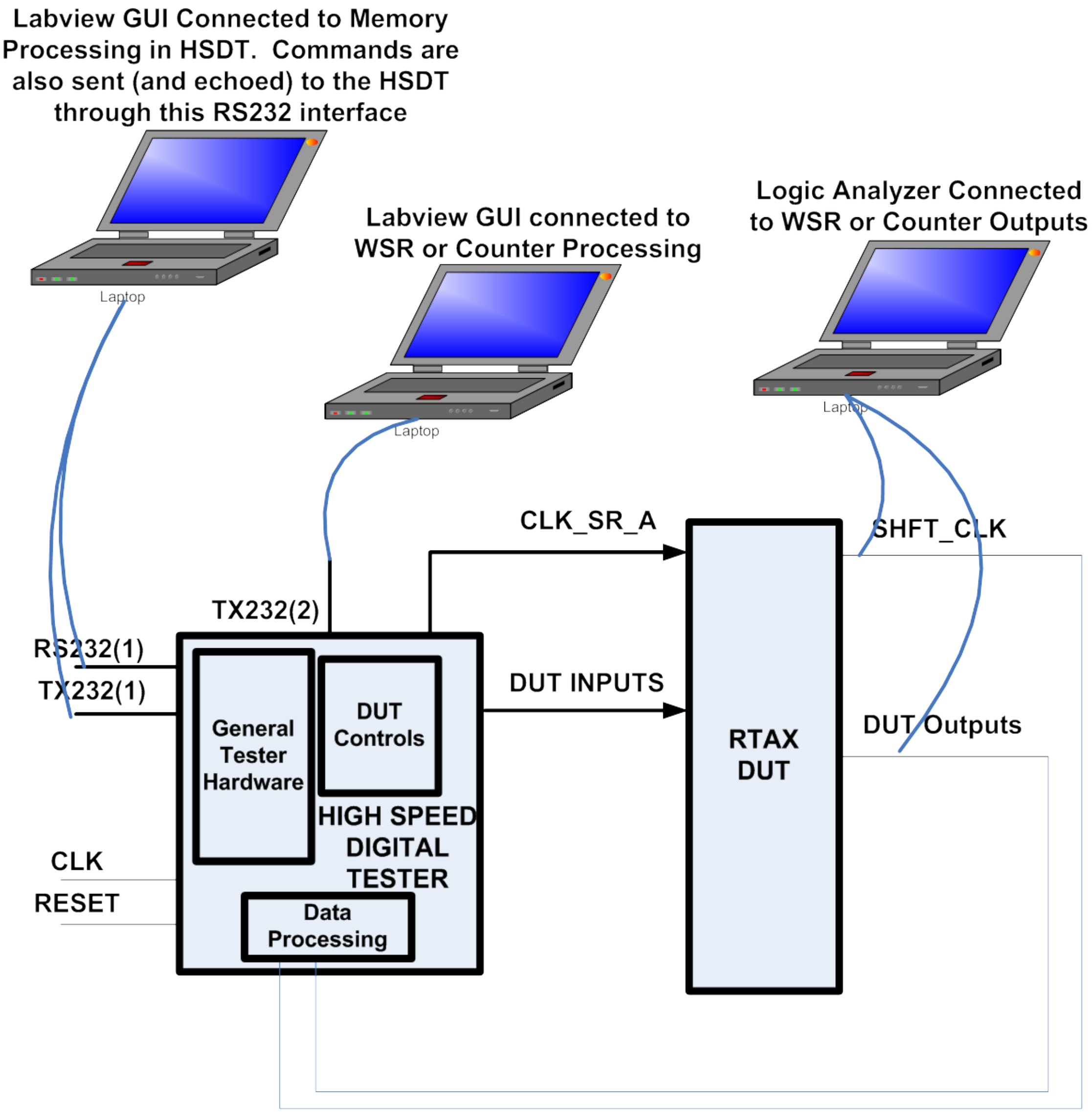

Labview GUI Connected to Memory Processing in HSDT. Commands are also sent (and echoed) to the HSDT through this RS232 interface
Laptop
Labview GUI connected to WSR or Counter Processing
Laptop
Logic Analyzer Connected to WSR or Counter Outputs
Laptop
CLK_SR_A
SHFT_CLK
TX232(2)
RS232(1)
TX232(1)
General Tester Hardware
DUT Controls
HIGH SPEED DIGITAL TESTER
Data Processing
DUT INPUTS
RTAX DUT
DUT Outputs
CLK
RESET

Figure 21: System Level Tester Architecture. Two PCs are running a Labview GUI. One PC is running a logic analyzer to have real time processing of the DUT WSR or Counter Array outputs. The PC connected to RS232(1) sends commands to the HSDT that set test parameters and starts test operations

4.1.1 I/O List and Definitions

Interface tables were supplied in the previous sections for all of the designs. The same I/O for each of the 2 DUTs are presented with respect to the HSDT in this section.

Table 13: I/O DUT1 I/O Table Shift Register, Memories, and Host PC Interface

Input Name	bits	Direction Wrt to HSDT	Description

CLK	1	IN	System clock of the HSDT from Board crystal
RESET	1	IN	HSDT system reset from Power supply
RX232(1)	1	IN	Serial receive input from Host PC. Used for PC to send commands to the HSDT
TX232(1)	1	OUT	Serial transmission line to Host PC. Used to Echo commands and to send back either Shift Register or Counter Error Data
TX232(2)	1	OUT	Serial transmission line to Host PC. Used to send back memory error data
RTAX_SHFT_CLK0	1	IN	Chain 0 shift Clock
RTAX_SHFT_CLK1	1	IN	Chain 1 shift Clock
RTAX_SHFT_CLK2	1	IN	Chain 2 shift Clock
RTAX_SHFT_CLK3	1	IN	Chain 3 shift Clock
RTAX_SHFT_CLK4	1	IN	Chain 4 shift Clock
RTAX_SHFT_CLK5	1	IN	Chain 5 shift Clock
RTAX_SHIFT_STRING0	4	IN	4-bit windowed output of chain 0
RTAX_SHIFT_STRING1	4	IN	4-bit windowed output of chain 1
RTAX_SHIFT_STRING2	4	IN	4-bit windowed output of chain 2
RTAX_SHIFT_STRING3	4	IN	4-bit windowed output of chain 3
RTAX_SHIFT_STRING4	4	IN	4-bit windowed output of chain 4
RTAX_SHIFT_STRING5	4	IN	4-bit windowed output of chain 5
RTAX_PATTERN	12	OUT	Each chain gets 2 bits from this vector: chain 0 receives bits (1:0). Chain 1 receives bits (3:2) etc… The bits select which data input pattern (all 0's, 1's, or checker board). See **Error! Reference source not found.** for selection details
RTAX_EN0	1	OUT	Sent to chain enable or disable chain 3
RTAX_EN1	1	OUT	Sent to chain enable or disable chain 4 and 5
CLK_SR0	1	OUT	Clock to all of the WSR circuitry
CLR_SR0	1	OUT	Reset to all of the WSR circuitry
RTAX_MEM_WE	1	OUT	Write Enable to all Memory

RTAX_MEM_WDATA	8	OUT	Write Data to all Memory
RTAX_MEM_RE	1	OUT	Read Enable to all Memory
RTAX_MEM_MUX	1	OUT	Selects 2 of 4 memory outputs to go to the HSDT for processing
RTAX_MEM_RDATA_A	8	IN	One of the 2 data inputs to HSDT
RTAX_MEM_ADDR_A	11	IN	Address input for data A
RTAX_MEM_RDATA_A	8	IN	One of the 2 data inputs to HSDT
RTAX_MEM_ADDR_A	11	IN	Address input for data B
RTAX_MEM_Clk	1	OUT	Clock used for memory circuitry in DUT
RTAX_MEM_RESET	1	OUT	Reset used for memory circuitry in DUT
RTAX_MEM_stop_scrub	1	OUT	Directs DUT to not do scrub cycles
RTAX_MEM_BYPASS	1	OUT	Directs DUT to bypass EDAC circuitry (only pertinent to the 2 memories that have EDAC)

4.2 RS232 communication from the HSDT to the Host PC

All RS232 communication from the HSDT to the host PC is prefaced with a header. . Information from the HSDT to the Host PC is one of the following listed in Table 14: an alive-timer, a command echo, or an Error Record.

Table 14: A list of the HSDT to Host PC RS232 Header bytes. Only the HSDT uses header information. The host PC sends pure commands to the HSDT without headers.

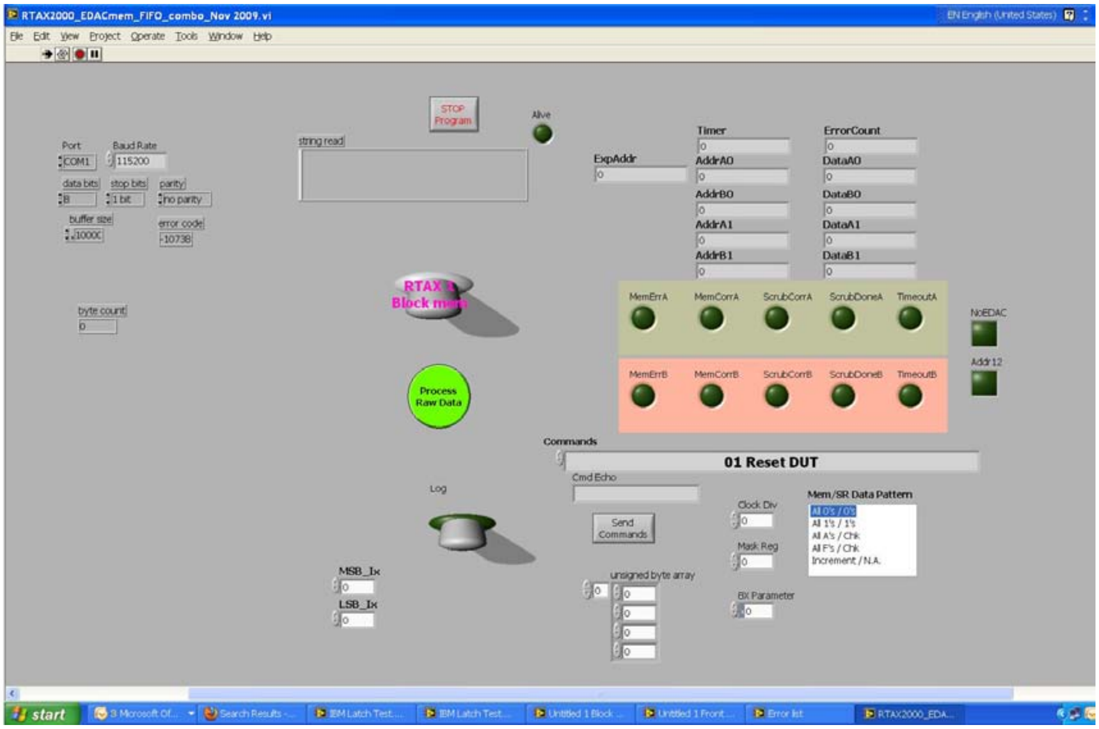
Header	Description
00 FA F3 20	Alive Header No data bytes follow (i.e. only the header is sent from the HSDT to the PC)
00 FA F3 22	Command Echo. 4 data bytes follow that represent the command that was previously sent from the Host PC to the HSDT.
00 FA F3 21	Data Error Record: 23 bytes follow.

4.3 RS232 communication From the Host PC to the HSDT

Communication from the host PC to the HSDT does not contain a header. Information sent from the host PC to the HSDT are commands and are all 4 bytes in length The interface is controlled by a user GUI designed with LabView software.

4.3.1 User GUI

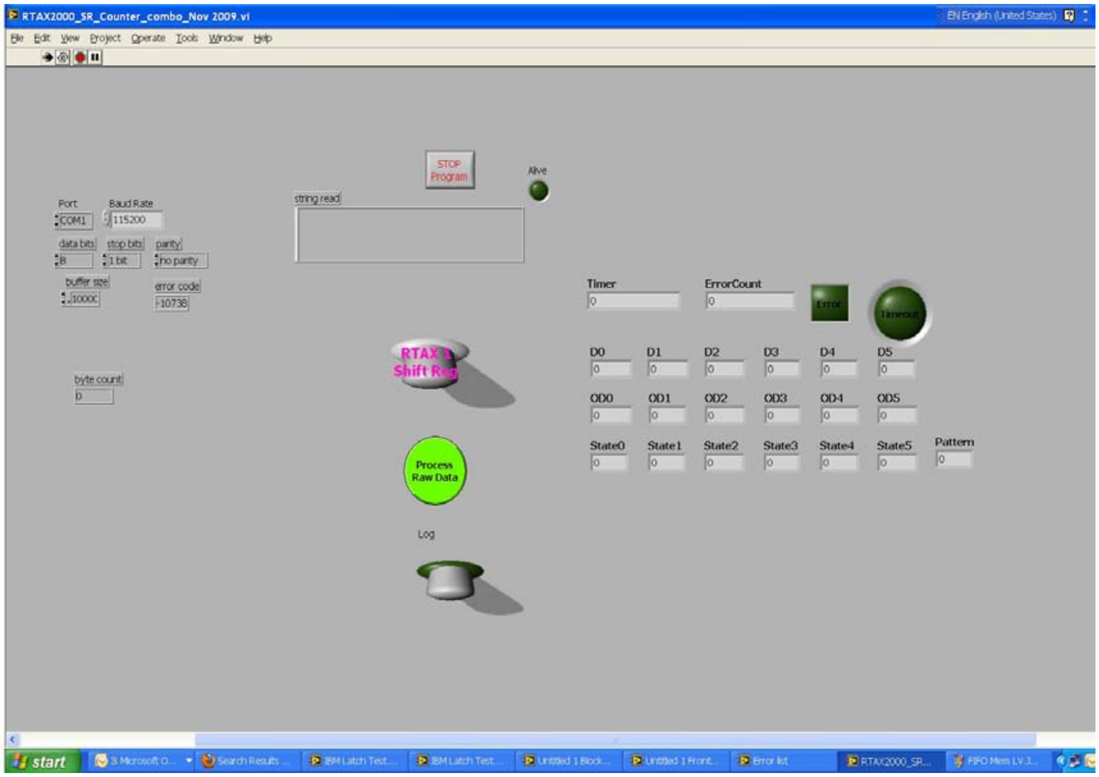
Commands are sent by typing specific values into Labview fields or controlling Labview on/off buttons listed on the screen. Figure 22 and Figure 23 are the GUI’s used when testing DUT1. Figure 24 and Figure 25 are the Labview GUIs used when testing DUT2.



RTAX2000_EDACmem_FIFO_combo_Nov 2009.vi

Port: COM1
Baud Rate: 115200
data bits: 8
stop bits: 1 bit
parity: no parity
buffer size: 10000
error code: -10738
byte count: 0
string read:
ExpAddr: 0
Timer: 0
ErrorCount: 0
AddrA0: 0
DataA0: 0
AddrB0: 0
DataB0: 0
AddrA1: 0
DataA1: 0
AddrB1: 0
DataB1: 0
Commands: 01 Reset DUT
Cmd Echo:
Clock Div: 0
Mask Reg: 0
BX Parameter: 0
MSB_Ix: 0
LSB_Ix: 0
unsigned byte array: 0 | 0, 0, 0, 0
Mem/SR Data Pattern: All 0's / 0's (selected), All 1's / 1's, All A's / Chk, All F's / Chk, Increment / N.A.

Figure 22: DUT1 Block Memory GUI. Communicates with the HSDT via the RS232(1) and TX232 (1) Interfaces. Commands are sent using this GUI. Command Echoes and Block Memory error reports are sent to this GUI from the HSDT


RTAX2000_SR_Counter_combo_Nov 2009.vi
EN English (United States)
File Edit View Project Operate Tools Window Help
STOP Program
Alive
string read
RTAX 1 Shift Reg
Process Raw Data
Log
Error
Timeout
start

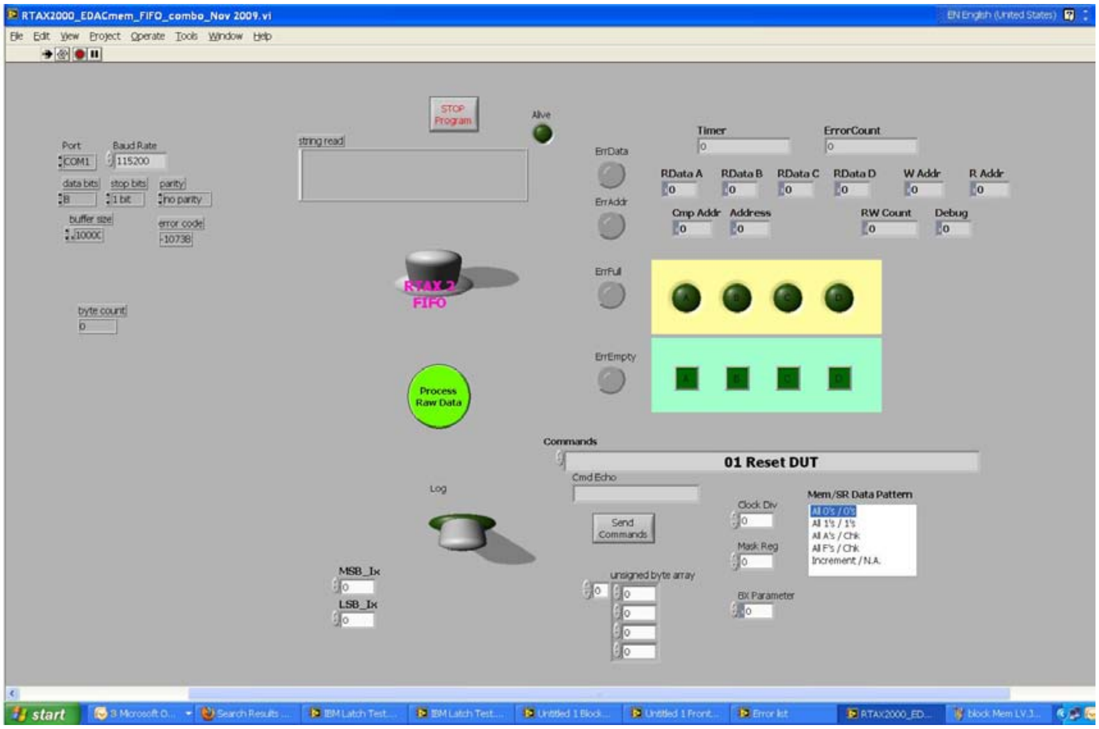
Port: COM1
Baud Rate: 115200
data bits: 8
stop bits: 1 bit
parity: no parity
buffer size: 10000
error code: -10738
byte count: 0
Timer: 0
ErrorCount: 0

D0	D1	D2	D3	D4	D5
0	0	0	0	0	0

OD0	OD1	OD2	OD3	OD4	OD5
0	0	0	0	0	0

State0	State1	State2	State3	State4	State5	Pattern
0	0	0	0	0	0	0

Figure 23: DUT 1 WSR Labview Interface. Communicates with the HSDT via the TX232 (2) Interfaces. WSR error reports are sent to this GUI from the HSDT.


RTAX2000_EDACmem_FIFO_combo_Nov 2009.vi
EN English (United States)
File Edit View Project Operate Tools Window Help
STOP Program
Alive
string read
RTAX 2 FIFO
Process Raw Data
Log
ErrData
ErrAddr
ErrFull
ErrEmpty
A
B
C
D
Send Commands
start

Port: COM1
Baud Rate: 115200
data bits: 8
stop bits: 1 bit
parity: no parity
buffer size: 10000
error code: -10738
byte count: 0
Timer: 0
ErrorCount: 0

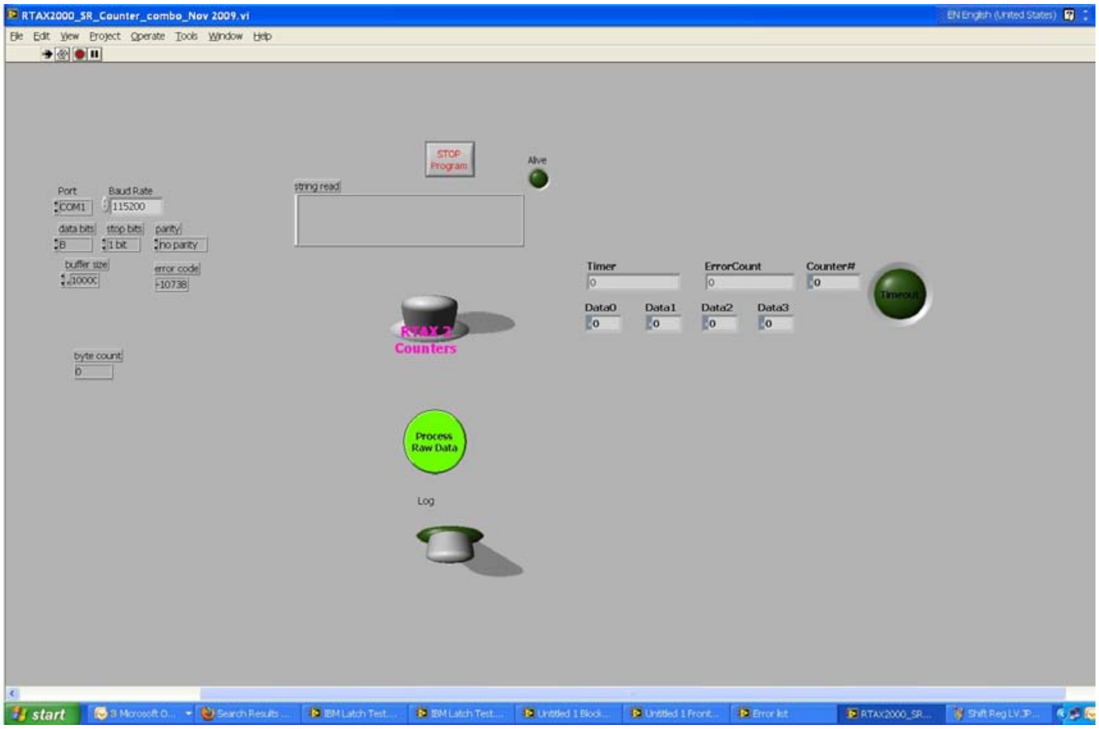
RData A	RData B	RData C	RData D	W Addr	R Addr
0	0	0	0	0	0

Cmp Addr	Address	RW Count	Debug
0	0	0	0

Commands: 01 Reset DUT
Cmd Echo:
Clock Div: 0
Mask Reg: 0
BX Parameter: 0
unsigned byte array: 0 | 0, 0, 0, 0
MSB_Ix: 0
LSB_Ix: 0

Mem/SR Data Pattern
- All 0's / 0's
- All 1's / 1's
- All A's / Chk
- All F's / Chk
- Increment / N.A.

Figure 24:DUT2 FIFO Memory GUI. Communicates with the HSDT via the RS232(1) and TX232 (1) Interfaces. Commands are sent using this GUI. Command Echoes and Memory error reports are sent to this GUI from the HSDT



Port: COM1
Baud Rate: 115200
data bits: 8
stop bits: 1 bit
parity: no parity
buffer size: 1000C
error code: -10738
byte count: 0
string read:
Timer: 0
ErrorCount: 0
Counter#: 0
Data0: 0
Data1: 0
Data2: 0
Data3: 0

Figure 25:DUT 2 Counter Array Labview Interface. Communicates with the HSDT via the TX232 (2) Interfaces. Counter Array error reports are sent to this GUI from the HSDT

The following section describes the commands sent from the Host PC to the HSDT.

4.3.2 User Interface and Command Control

The User controls the tests via a LABVIEW interface running on a PC. The PC communicates with the HSDT with a RS232 serial link. The format of communication is a command/Data 4 byte word.

Table 15 : Summary of Commands Used in RTAX-S Tester

Command #	Command	D0	D1	D2	Description
01	Reset HSDT	n	n	n	Resets RTAX-S
99	Reset DUT Memory	N	N	N	Resets RTAX-s Memory control logic (address counters, read and write enables).
03	Reset of Shift register or counter tests	N	N	n	Places either the WSRs or the counter arrays in reset mode

04	Reset + start of Shift register or counter tests	N	N	n	Sends a reset pulse and then starts the WSRs or counter array tests
02	Allow the Start of Memory Test	n	n	n	Starts RTAX-S clock and data generation and writes all memory locations with the user set data pattern. Places the tester in a wait state to receive the readback user command
90	WSR Pattern Number	y	n	n	D0 = 00… data input to WSR is a constant 0 01… data input to WSR is a constant 1 02 or 03… data input to WSR is checker board (i.e. changes every WSR clock cycle)
92	Memory Data Pattern	y	n	n	D0 = 00… data written to memory is a constant all 0 (x00) 01… data written to memory is constant all 1’s (xFF) 02 … data written to memory is constant all 1’s (x55) 03 … data written to memory is constant all 1’s (xAA) 04 … data written to memory is the lower order 8 bits of the current address
A0	Memory Clock Frequency	y	n	n	D0 is the Clock frequency divider of 100mhz. the synthesized clock will be sent to the DUT memories as their system clock.
A1	WSR or Counter Aray Clock Frequency	y	n	n	D0 is the Clock frequency divider of 100mhz. The synthesized clock will be sent to the DUT WSR or counter Arrays as their system clock.

81	Read Write Modify EDAC Command	N	N	N	
82	Write once with continuous read	N	N	N	Not used in testing
83	Write all read all	N	N	N	Write all of the memory locations then read all (do not correct the memory location upon upset – it is assumed to be overwritten during the write all phase).
84	Write Once with no read	N	N	N	Used for static testing prior to beam. It is followed by 85 command to perform a read once
85	Read Once with no write	N	N	N	Used for static testing – after a write has previously been performed
88	Bypass EDAC and Scrub Logic	N	N	n	Turns off EDAC and scrub. Necessary during static testing.

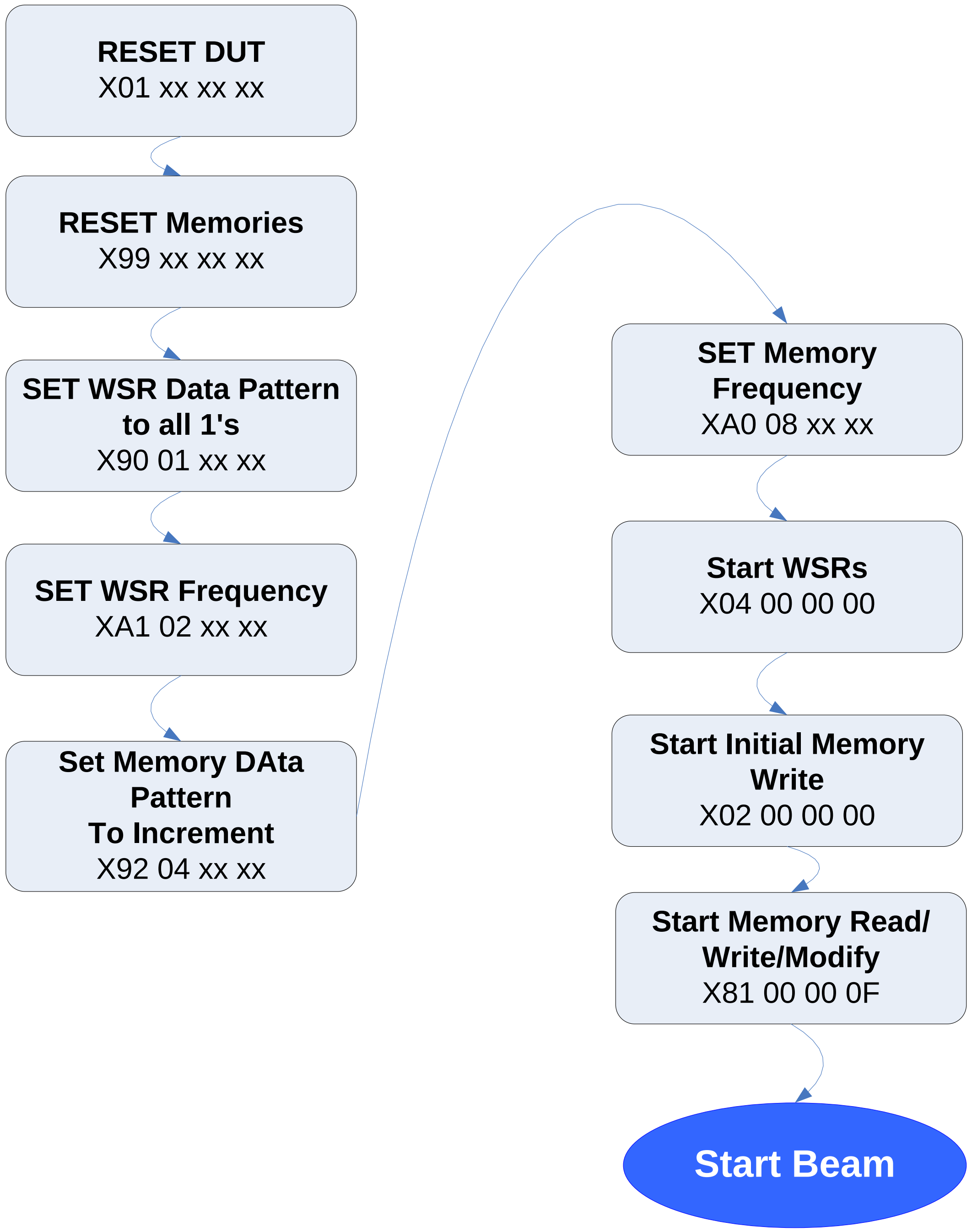


Figure 26: DUT 1 Block SRAM Memory Test Example- Read/Modify/Write at 12.5MHz with WSRs @ 50Mhz

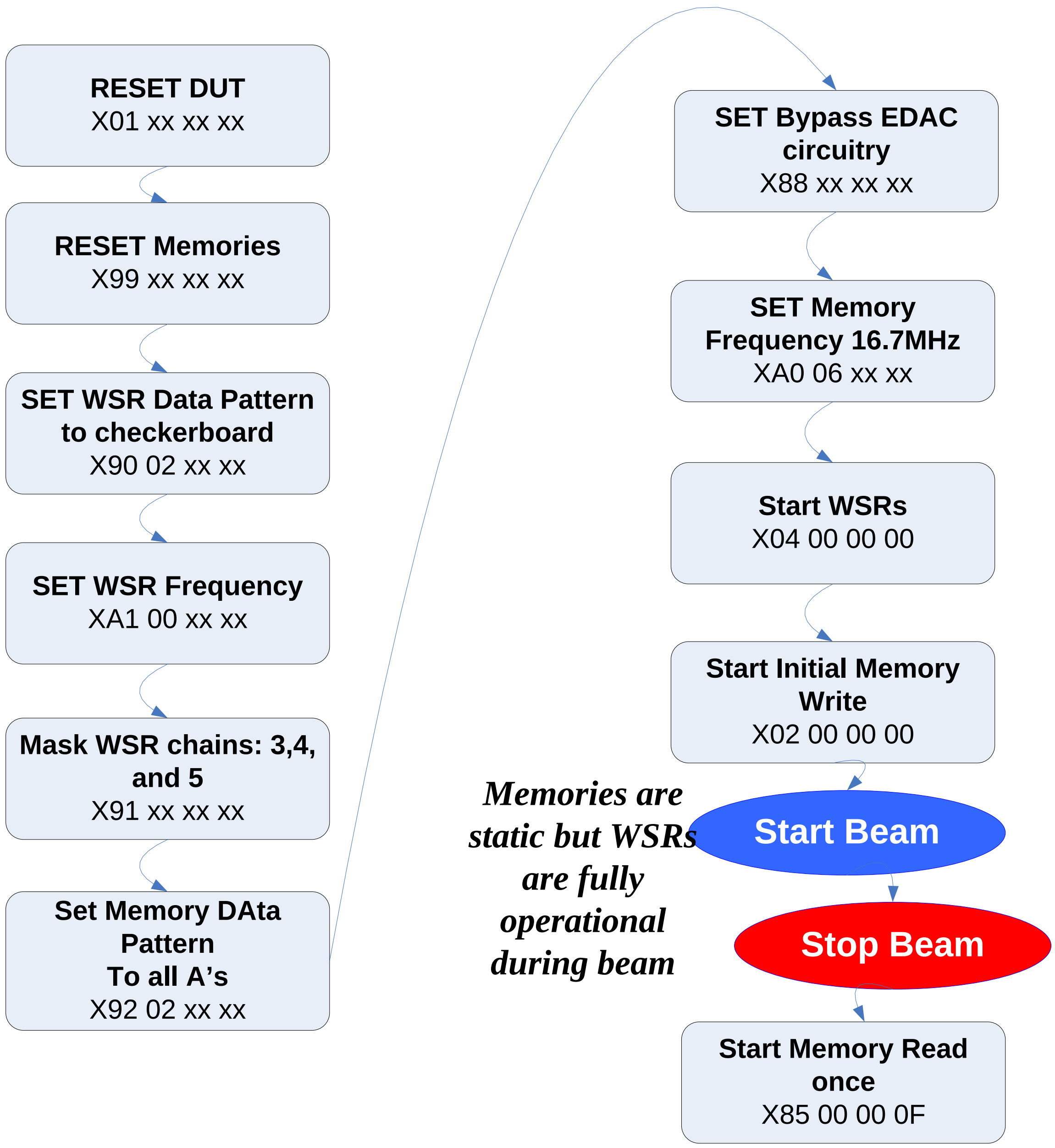

RESET DUT
X01 xx xx xx
RESET Memories
X99 xx xx xx
SET WSR Data Pattern to checkerboard
X90 02 xx xx
SET WSR Frequency
XA1 00 xx xx
Mask WSR chains: 3,4, and 5
X91 xx xx xx
Set Memory DAta Pattern To all A's
X92 02 xx xx
SET Bypass EDAC circuitry
X88 xx xx xx
SET Memory Frequency 16.7MHz
XA0 06 xx xx
Start WSRs
X04 00 00 00
Start Initial Memory Write
X02 00 00 00
Start Beam
Stop Beam
Start Memory Read once
X85 00 00 0F
Memories are static but WSRs are fully operational during beam

Figure 27: DUT1 Block Memory Static test- WSRs @ 100Mhz

5. DUT TEST PROCEDURES

5.1 DUT1 WSR Testing

The objectives of testing are to determine:

- Bit upset rates
- Frequency effects to SET capture rates
- Data Pattern effects to SET capture rates
- Global route effects

In order to obtain WSR objectives, tests were performed varying several parameters as listed in Table 16.

Table 17: WSR Test Parameter Variation

	Frequency Effects	Data Pattern Effects	Global Routes	General Bit Upsets
Architectural: Variation in combinatorial blocks between DFFs in WSR	x			x
Frequency variation from test to test	x	x		x
Data pattern variation from test to test	x	x		x
LET variation	x	x	x	x
Architectural: Some chains contain global enable routes, some do not			x	x
Static Tests				x

5.1.1 Dynamic: Evaluate susceptibility of WSR

1. Bias the device, turn on clocks and toggle reset
2. Let WSR run and compare with expected DUT output pattern (verify no errors)
3. Irradiate DUT
4. Tester reads DUT and compares to expected value
   - If error during read, then the HSDT records that an error has occurred and sends the data value with timestamp to the PC
   - Goto 4 if not done with test else goto 5
5. Stop Beam
6. Reset Tester and DUT to prepare for next test

5.2 DUT1 Block Memory Tests

5.2.1 Static: Evaluate susceptibility of SRAM Cells in biased-static state

1. Bias the device, turn on clocks and toggle reset
2. Load memory with pattern
3. Turn clock off (clock is static, reset is inactive)
4. Irradiate DUT
5. Stop Beam
6. Turn on clocks and read out memory

5.2.2 Dynamic: Evaluate susceptibility of SRAM Cells in biased-dynamic states

1. Bias the device, turn on clocks and toggle reset
2. Load initial memory pattern
3. Start reading the device (compare with expected memory pattern and verify no errors)
4. Irradiate DUT

5. Tester reads DUT and compares to expected value
    - If error during read, then the HSDT writes current location with good value and the HSDT records that an error has occurred and sends the data value with timestamp to the PC
    - If not done with testing Goto 5, else goto 6
6. Stop Beam
7. Reset Tester and DUT to prepare for next test

5.3 DUT2 Counter Array Tests

5.3.1 Dynamic: Evaluate susceptibility of Counter DFF cells in biased-dynamic states

1. Bias the device, turn on clocks and toggle reset
2. Let counter logic run and compare with expected counter pattern (verify no errors)
3. Irradiate DUT
4. Tester reads DUT and compares to expected value
    - If error during read, then the HSDT records that an error has occurred and sends the data value with timestamp to the PC
    - If not done with test Goto 4, else goto 5
5. Stop Beam
6. Reset Tester and DUT to prepare for next test

Table 18: Counter Array Test Parameter Variation

	Frequency Effects	Global Routes	General Bit Upsets
Frequency variation from test to test	x		x
LET variation	x	x	x

5.4 DUT2 FIFO Memory Tests

5.4.1 General Dynamic: Evaluate susceptibility of SRAM Cells and Address Counters in biased-dynamic states (Write All – Read all)

1. Bias the device, turn on clocks and toggle reset
2. Load initial memory pattern to all of the address locations
3. Read back every memory location (compare with expected memory pattern and verify no errors)
4. Irradiate DUT
5. Write all locations
6. Read all locations:
    - If error during read, then the HSDT records that an error has occurred and sends the data value with timestamp to the PC
    - If not done with test Goto 6 else goto 7
7. Reset Tester and DUT to prepare for next test

5.4.2 Full Flag Dynamic: Evaluate susceptibility of SRAM Cells, Address Counters, and Full Flags

in biased-dynamic states (Write All – Read one-Write one)

1. Bias the device, turn on clocks and toggle reset
2. Load initial memory pattern to all of the address locations
3. Read back one memory location (compare with expected memory pattern and verify no errors)
4. Irradiate DUT
5. Write one location (FULL flag should become active)
6. Read one locations (FULL Flag should become inactive):
    - If error during read or error with flag activation or deactivation, then the HSDT records that an error has occurred and sends the data value with timestamp to the PC
    - If not done with test Goto 6 else goto 7
7. Stop Beam
8. Reset Tester and DUT to prepare for next test

5.4.3 Empty Flag Dynamic: Evaluate susceptibility of SRAM Cells, Address Counters, and Empty Flags in biased-dynamic states (Write All Read all – Write one- Read one)

1. Bias the device, turn on clocks and toggle reset
2. Load initial memory pattern to all of the address locations
3. Read back all memory locations (compare with expected memory pattern and verify no errors) Empty flag should become active at the end of the read sequence
4. Irradiate DUT
5. Write one location (Empty flag should become inactive)
6. Read one location (Empty Flag should become active):
    - If error during read or error with flag activation or deactivation, then the Tester records that an error has occurred and sends the data value with timestamp to the PC
    - If not done with test Goto 6 else goto 7
7. Stop Beam
8. Reset Tester and DUT to prepare for next test

Table 19: FIFO Test Parameter Variation

	SRAM Cell	Global Routes	Address	Full Flag	Empty Flag
Write All Read all	x	x	x		
Write all Read one Write one	x	x	x	x	
Write all Read all Write one read one	x	x	x		x

6. PROCESSING THE DUT OUTPUTS

The outputs of the DUT are fed to the tester for data processing. The objective of the data processing is to capture data from the DUT, compare to an expected value, and report to the host PC if there is an error. The DUT system clock and reset signals are generated in the HSDT.

6.1 DUT1 and DUT2 SHIFT_CLK Processing

Regarding the SHIFT_CLK in the DUT1 and DUT2 WSR and Counter Arrays, it is used to alert the tester that the DUTs are alive. The SHIFT_CLKs are always ¼ of the speed of the DUT system clock.

Due to the interface delays and device latencies and in order to consequently decouple the DUT to tester timing restrictions, the DUT SHIFT_CLK is considered asynchronous to the tester and is sampled using the HSDT system clock (max 100 MHZ). Thus, the tester's sampling clock will always be 4 times as fast as SHIFT_CLK. The SHFT_CLK is fed into a metastability filter and an edge detect. This process takes 1 to 2 clock cycles of the sampling clock (detection will be delayed by 1 to 2 sampling clock cycles of the actual edge).

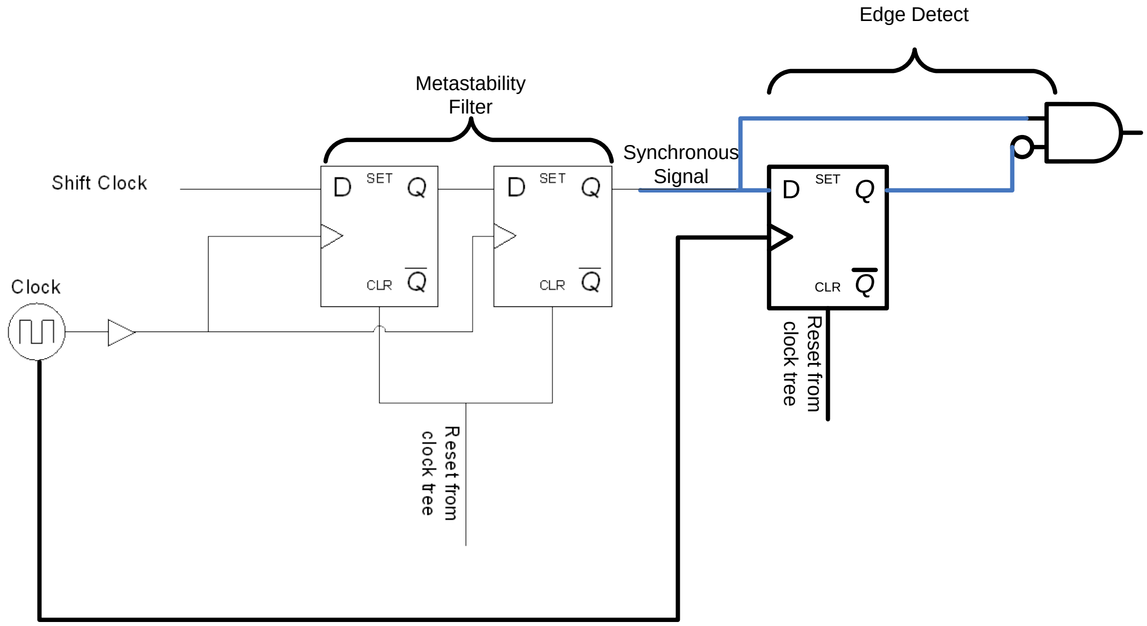


Figure 28: Shift_ClK Capture consists of a Metastability Filter and a Edge Detect

The SHIFT_CLK edge is expected to come at a frequency that is ¼ of the HSDT clock. If the edge is stopped, glitched, or missing, the event is reported to the host PC by the HSDT.

6.2 DUT1 WSR Data Processing

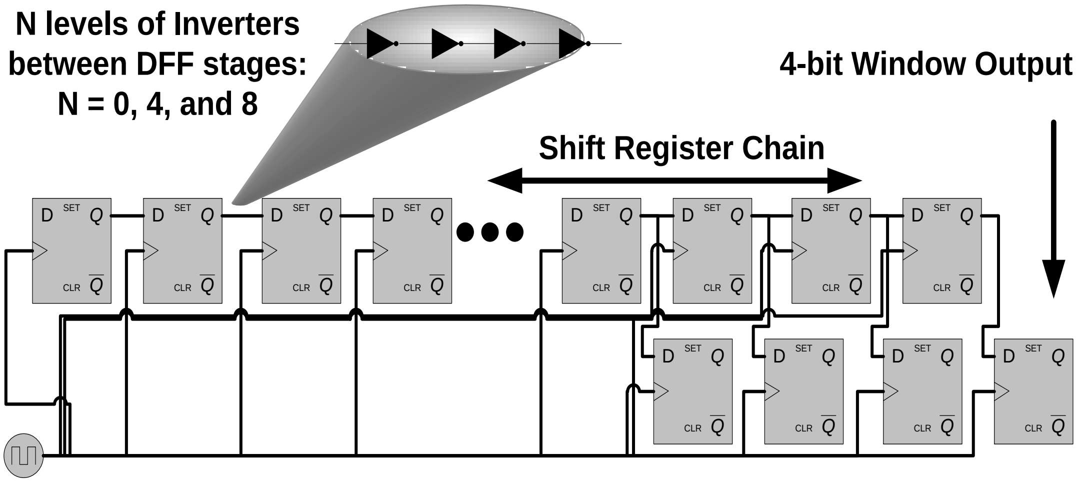


Figure 29: Another look at a WSR string. The 4-bit window is the registers in the illustration that are underneath the shift register string. The last 4-bits of the string are shifted into window, once every 4 clock cycles.

In order to avoid metastable events due to an error in the output, data is registered twice before evaluation. As illustrated in Figure 9, the four bit WSR output window stays constant for all of the tested data patterns (all 0’s, all 1’s, and checkerboard). Therefore determining bit flips are simple… it’s merely checking for a change in data input (does the data from the current cycle(n) equal the data from the last cycle(n-1)?).

One must take caution because the tester is always reporting changes in data. This requires a record report when data changes from expected to corrupted, and then from corrupted to expected. In other words, not every report from the tester suggests data is bad. On the good note is with the inclusion of the timestamp, this gives the user the ability to post process and to determine how long data is in error and if recovery is possible.

Table 20: Types of Errors with Detection and Reporting Scheme

Error	HSDT Detect
Bit flip in shift register string	Has the data changed from the previous cycle?
Global Route glitch: Clock, reset, or enable	There will be a burst of bad data in the data string. A reset has a strict error signature because of how the string comes out of reset… e.g. the string is set to all 0’s for a certain number of cycles. A clock or enable glitch can cause a burst of noise data. All cases are eventually recoverable for a WSR.
Bit flip in window or I/O	Has the data changed from the previous cycle and is only corrupt for less than the 4 clock cycle window.

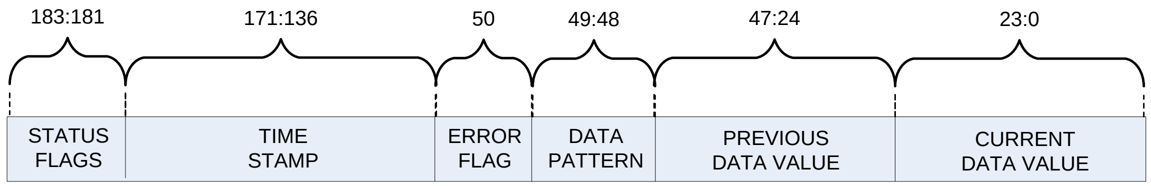


Figure 30: WSR Error Record Data Fields. Each error record is prefaced with an error header (00 FA F3 21) when being sent from the HSDT to the Host PC.

Figure 30 demonstrates an error record that is sent from the HSDT to the user PC regarding a WSR behavior. The relationship between the previously captured and currently captured WSR windowed values explains the error signatures and event occurrences within the WSR structure during testing.

Table 21 WSR Field Table. Yellow shading indicates fields that are generated from DUT intputs. White Fields are sourced from either tester settings or tester logic

Field	# of Bits	Description
Current Value	24	Currently captured data (cycle N)
Previous Value	24	Previously captured data (cycle N-1)
Data Pattern	2	2 bit data pattern set by commands (00 or 01 or 10)
Error Flag	1	Not used
Freq	8	DUT frequency set by user command
Error Count	16	Unused
Time Stamp	32	Cycle counter. Must multiply by the DUT frequency to convert to time. Used to determine error burst sequences
Status	3	Indicates type of error record: “001” is a timeout – one of the shift clocks not detected “011”Out of timeout – all shift clocks are recovered “000” Error or non-error – current value does not equal previous value “010” Debug check – command was sent to check value settings

Table 22: WSR Current Value and Previous Value Processing

Previous Value and Current Value	Indication
Previous = good and Current=bad	WSR just reported an error
Previous = bad and Current=good	WSR has recovered from error
Previous= bad and Current = bad	WSR is in a burst of error
Previous = exact inversion of Current	Either an enable was struck or there was a shift clk hit (e.g. previous = hex 5 and current = hex A).

6.2.1 WSR SEU Cross Section Calculations

Generally, calculating the SEU Cross Section for WSR chains is a simple process. Count the number of upsets and divide by the reported particle fluence. The WSR cross section is then normalized to the number of bits within the chain that is being analyzed. When there is a burst of data, the total fluence must be adjusted because upsets are not being captured during the inoperable burst period. Hence, the number of particles that the device experienced during said period is subtracted off of the total.

$$\frac{\qquad\qquad\qquad}{} \ \frac{\quad}{} \tag{5}$$

6.3 DUT1 Block Memory Tests

In order to avoid metastable events, all inputs that originate from the DUT go through two stages of HSDT DFFs prior to processing. DUT Address and data inputs are monitored after each read. As a recap, there are 4 memory structures: 2 structures that contain the embedded EDAC/Scrub circuitry and 2 that do not have any EDAC circuitry. In order to conserve I/O, only 2 out of the 4 of the memory address and data lines can be read simultaneously. This selection of address and data buses is controlled by a one bit MUX (MUX=0 passes the 2 EDAC memories buses to the HSDT and MUX=1 passes the 2 non-EDAC memory buses to the HSDT). Each read cycle is 16 HSDT cycles and each write cycle is 16 HSDT cycles. Consequently, 1 read or write is performed once every 16 cycles during testing.

6.3.1 Block Memory Address Processing

The tester has a local copy of the current address. As previously stated, each of the 4 Memory modules has its own address counter. During each read and write cycle, the available DUT addresses are captured and compared to the HSDT local copy. If the addresses are not all equivalent, then an error record will be sent to the HSDT.

6.3.2 Block Memory Data Processing

The tester has a local copy of the expected data byte. The expected value is generated by the user supplied memory pattern set by user command x92.

Table 23: User Command and data patterns

Command	Data Pattern written to Block Memory	Expected Value Compare to data read back from block memory
X92 00 00 00	00	00
X92 01 00 00	FF	FF
X92 02 00 00	AA	AA
X92 03 00 00	55	55
X92 04 00 00	Incremental pattern	Lower bits of current address

If the data pattern is all 0’s, 1’s, 5’s, or A’s, then the expected value (used for readback comparisons) stays constant and will be the selected data pattern. If the data pattern is a count, then the tester will write the lower 8 bits of the address (hence a count) into the current memory word. This simplifies the readback process during incremental-data testing by requiring the data word to always be equal to the lower bits of the address.

6.3.3 Processing and Error Detection Summary

Table 24: DUT1 Block Memory SEE detection and processing summary

Event	Detection
Bit Flip in SRAM Cell	Read back data will not equal expected. Tests will reflect the advantage of utilizing EDAC circuitry (prior to potential EDAC SEFIs)
Address Counter upset	Address readback will not equal expected. Regarding to count data pattern tests, the data readback will also be incorrect
EDAC SEFI	Data is unable to be corrected or data is getting significantly corrupted
Error Bursts	Timestamps reflect consecutive read or write cycles
Reset Upset	All addresses unexpectedly go to 0
Clock upset	Unrecoverable Chaos in outputs

6.3.4 DUT1 Block Memory Error Record

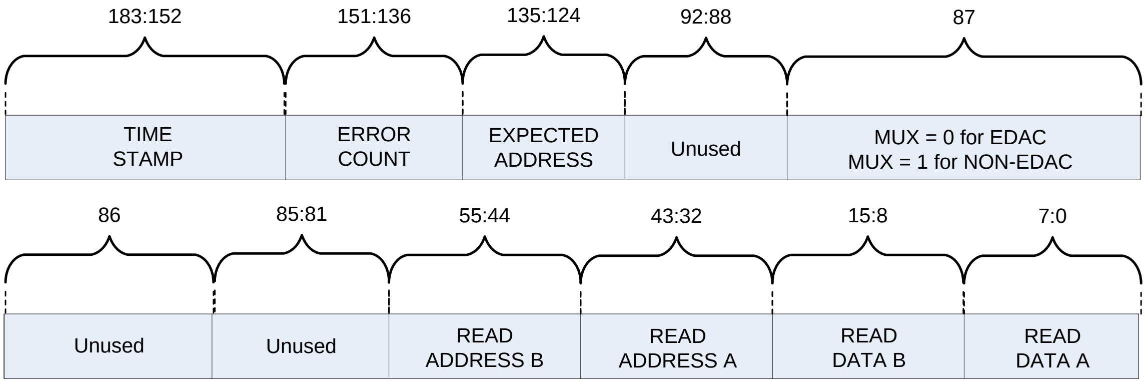


Figure 31: DUT1 Block Memory Error record:

Table 25: DUT1 Block Memory Error Record Fields: Yellow indicates Fields generated from DUT Inputs

Field	# of Bits	Description
Read DATA A	8	One of the two readback memory data words
Read DATA B	8	One of the two readback memory data words
Read Address A	12	One of the two readback memory addresses
Read Address B	12	One of the two readback memory addresses
MUX	1	MUX=0 indicates readback or write of a EDAC memory Block MUX=1 indicates readback or write of a non-EDAC memory Block
Expected Address	12	Local copy of the memory address (used as a comparison point for Read Address A and Read Address B) Lower 8 bits also represent expected the expected data if the data pattern were set to count
Error Count		unused
Time Stamp	32	Cycle counter. Must multiply by the DUT frequency to convert to time. Used to determine error burst sequences

6.4 DUT2 Counter Array

As illustrated in Figure 32, Figure 33, Figure 34, and Figure 35 the DUT output (to the HSDT) will increment once every 4 cycles. The increment is a continuous sequence of counts. It is important to note the difference between the terms counter number and counter value. The counter number is a tag (or name) given to a counter. The counter value is the actual data that is stored in the counter.

Each DUT output represents a counter value that is associated with a distinct counter. The counters are distinguished by a counter number as illustrated in Figure 32, Figure 33, Figure 34, and Figure 35. The tester keeps a local copy of the expected counter number with respect to the incoming counter value in order to keep track of the integrity of each counter.

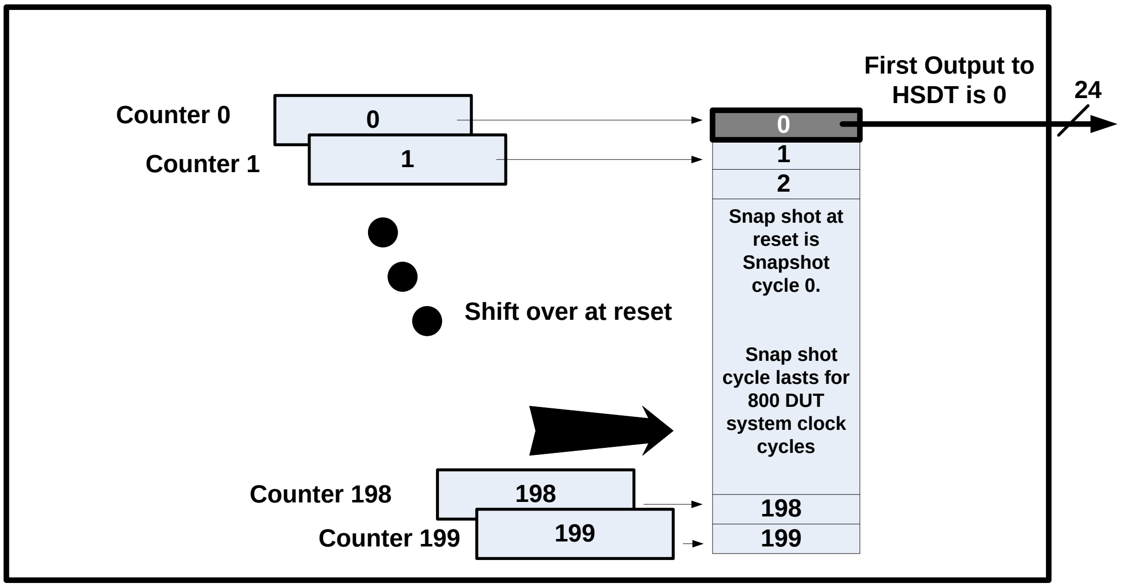

First Output to HSDT is 0
24
Counter 0
0
Counter 1
1
0
1
2
Snap shot at reset is Snapshot cycle 0.
Shift over at reset
Snap shot cycle lasts for 800 DUT system clock cycles
Counter 198
198
Counter 199
199
198
199

Figure 32: Counter Array at Reset. Coming out of reset, the shap shot register captures all counter array values

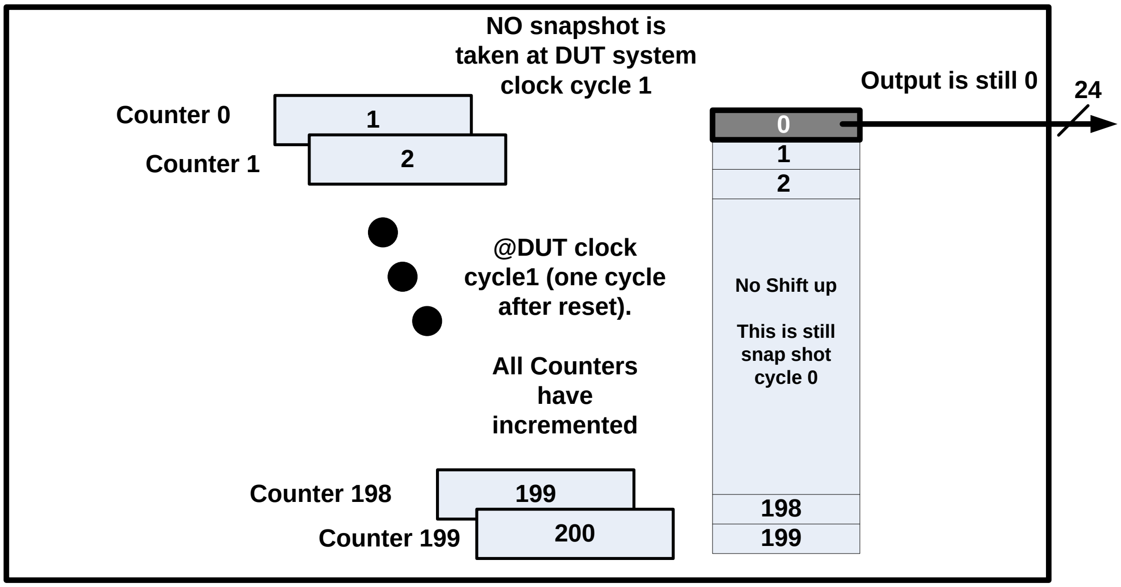

NO snapshot is taken at DUT system clock cycle 1
Output is still 0
24
Counter 0
1
Counter 1
2
0
1
2
@DUT clock cycle1 (one cycle after reset).
All Counters have incremented
No Shift up
This is still snap shot cycle 0
Counter 198
199
Counter 199
200
198
199

Figure 33: Next cycle after reset, all counter increment and there is no change in the snap shot register

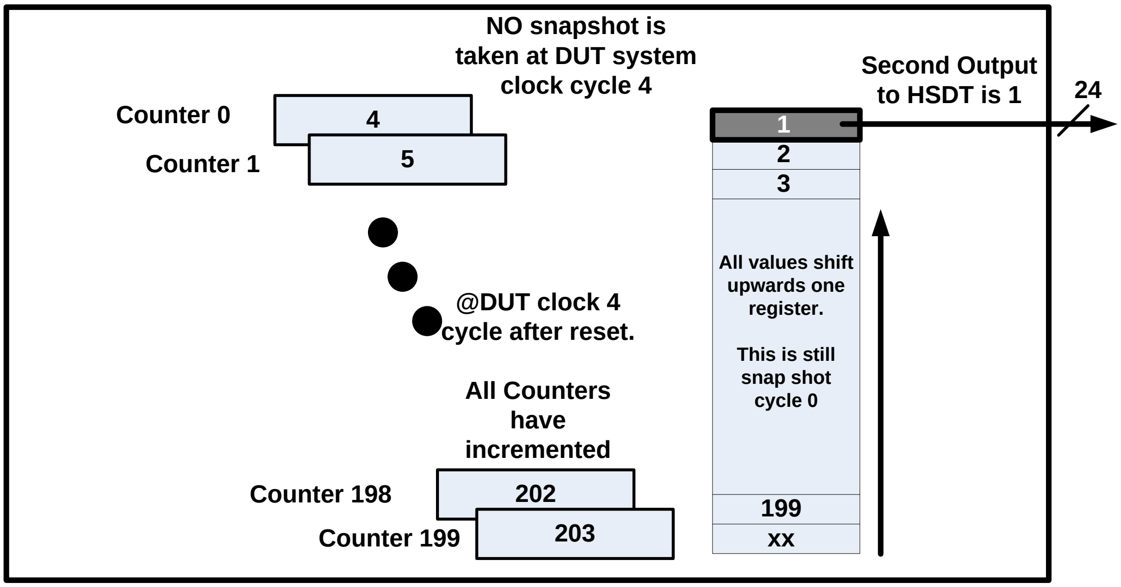

NO snapshot is taken at DUT system clock cycle 4
Counter 0
4
Counter 1
5
@DUT clock 4 cycle after reset.
All Counters have incremented
Counter 198
202
Counter 199
203
Second Output to HSDT is 1
24
1
2
3
All values shift upwards one register.
This is still snap shot cycle 0
199
xx

Figure 34: 4 cycles after reset, the output gets the next counter value because the snap shot register shifts up. Counter arrays simultaneously increment

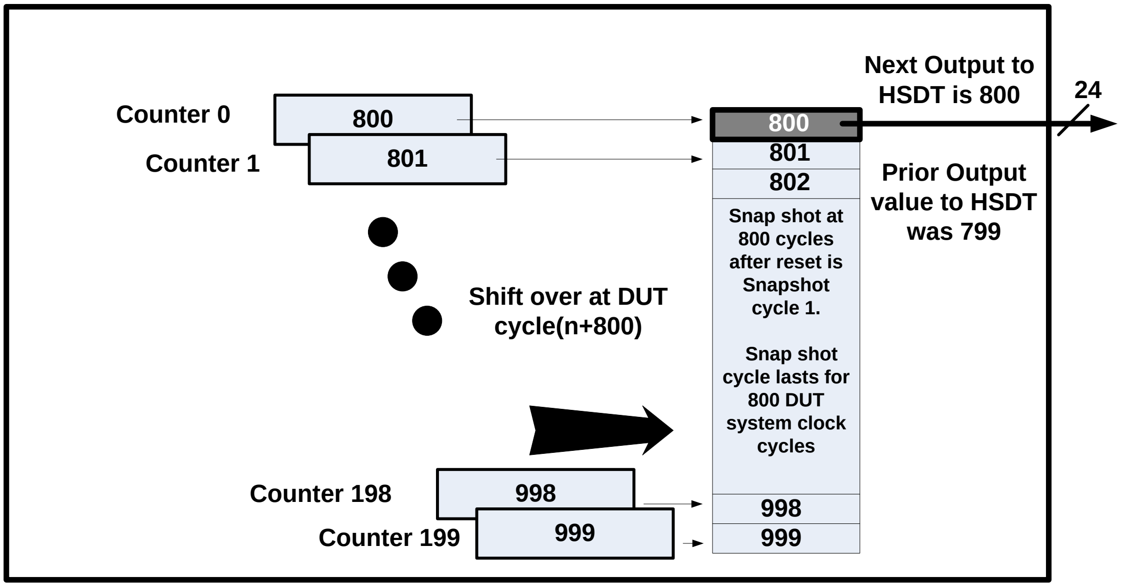

Counter 0
800
Counter 1
801
Shift over at DUT cycle(n+800)
Counter 198
998
Counter 199
999
Next Output to HSDT is 800
24
800
801
802
Snap shot at 800 cycles after reset is Snapshot cycle 1.
Snap shot cycle lasts for 800 DUT system clock cycles
998
999
Prior Output value to HSDT was 799

Figure 35: Snap shot cycle 1 (800 cycles after reset). Snap shot register gets all of the current counter values. Output to HSDT is a continuous sequence of incremented values

6.4.1 Counter Array Data Capture and compare

In order to avoid metastable events due to an error in the output, data is registered twice before evaluation. Both the data and the counter number are expected to increment every 4 cycles and will wrap around at its boundaries as listed in Table 26.

Table 26 Counter Value and Counter Number Wrap around Boundaries

	Bits	Wrap Around Value
Counter values	24	2^{24}-1 (after 2^{24}-1 next value is 0)
Counter Number	8	199 (after 199 next value is 0)

Regarding DUT value comparisons, any change in DUT output counter value must be an increment of 1 from the previous DUT counter value. If not, then an error record is sent to the HSDT. One must take caution because this will require at least 2 records per upset. The first record will be the counter that is in upset and the next record will be the following counter that is not in upset. This is because neither of the two counters will be an increment of 1 apart.

Post processing of the output records will help to determine if the upset occurred within the snap shot register or in the counter. This is done by understanding that if a counter is upset, it will stay upset. However, an upset in the snapshot register will only be upset for 1 snapshot cycle.

Upsets in the counter shift_clk value (should be a signal that is ¼ the DUT clock) was described in Section 6.1.

6.4.2 DUT2 Counter Array Error Record

A significant amount of post processing is expected to be performed on this data. Subsequently, the error record should contain enough information to comprehend and differentiate between events.

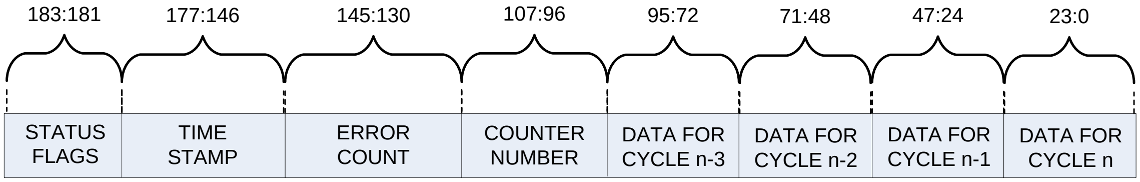


Figure 36: DUT2 Counter Array Error Record. Cycle n represents capture cycles. Capture cycles are once every 4 HSDT tester clock cycles.

Table 27: DUT2 Counter Array Error Record Fields: Yellow indicates Fields generated from DUT Inputs

Field	Bits	Description
Data Cycle N	24	Current DUT output. Error: If it is not an increment of 1 from the previous counter value and not an increment of 2 from the data value received cycle (n-2) No error (recover from error): If it is not an increment of 1 from the previous value but is an increment of the data value received cycle (n-2) Otherwise: DUT is in a burst of error
Data Cycle N-1	24	Capture cycle n-1 DUT output (capture cycle n-1 is actually 4 HSDT clock cycles from Data cycle N)
Data Cycle N-2	24	Capture cycle n-2 DUT output (capture cycle n-2 is actually 8 HSDT clock cycles from Data cycle N)
Data Cycle N-3	24	Capture cycle n-1 DUT output (capture cycle n-3 is actually 12 HSDT clock cycles from Data cycle N)
Counter Number	8	HSDT local copy of expected counter number. (0 through 199)
Error Count		
Time Stamp	32	
Status flags	3	

6.4.3 Counter Array SEU Cross Section Calculations

Because each of the 24 DFFs within each counter has different logic feeding into its data pin, all 24 bits of the counter are analyzed separately. Hence, there are 24 different cross sections. As an example, given that there are 100 counters total, the Cross section for bit0 will be made up of upsets on the 100 different bit0 per counter. Counter Array SEU cross sections in this document will be reported as binned cross sections. Each bin is an average of 4 counter bit SEU cross sections. The first bin in the average cross section for bit0, bit1, bit2, and bit3. There are six bins in total because there are 24 bits in a counter. There are 400 bits within each bin.

$$\frac{\qquad\qquad}{}\ \frac{\quad}{} \qquad \textbf{(6)}$$

$$\frac{\qquad}{}\ \frac{\quad}{} \qquad \textbf{(7)}$$

6.5 DUT2 FIFO Memories

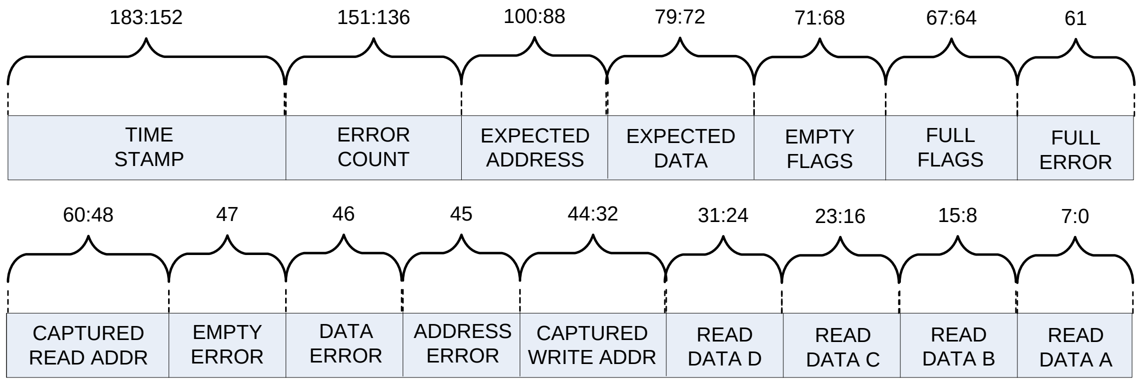


Figure 37: DUT2 FIFO Memory Error Record

7. HEAVY ION TEST FACILITY AND TEST CONDITIONS

Facility: Texas A&M University Cyclotron Single Event Effects Test Facility, 15 MeV/amu tune).

Flux: $1.0x10^4$ to $2.0x10^5$ particles/cm^2/s

Fluence: All tests were run to 1 x 10^7 p/cm^2 or until destructive or functional events occurred.

Table 28: LET Table

Ion	Energy (MEV/Nucleon)	LET (MeV*cm^2/mg) 0°	LET (MeV*cm^2/mg) 45 °	LET (MeV*cm^2/mg) 60 °	Test Dates
He	15	.016			05/10
Ne	15	2.8	3.96	5.6	05/10;08/10;12/10
Ar	15	8.5	12.6		12/09; 05/10;08/10;12/10
Cu	15	12.1	17.8		12/09; 05/10;08/10;12/10
Kr	15	28.7	40.73		12/09; 05/10;08/10
Xe	15	52.7	75.09		12/09; 05/10;08/10

Test Conditions:

Test Temperature: Room Temperature

Operating Frequency: 15 MHZ to 150MHZ

Power Supply Voltage: 3.3v I/O and 1.5V Core.

The RTAXS devices were irradiated with Argon, Krypton, and Xenon beams at normal incidence, 0, 45 and 60 degrees (yielding effective LETs values listed in Table 28: LET Table) at the Texas A&M University Cyclotron Single Event Effects Test Facility . Faults from the RTAXS devices were encountered at all LETs at 100MHZ. However, the number of SEUs was very low at Argon.

The RTAXS devices were monitored for Single Event latchup under the above conditions. Each part was placed in the beam until a Single Event latch (SEL) event occurred or 10^7 ions/cm^2 – the beam fluence was then recorded. During our experiment, no Single Event latchup events occurred, yielding a threshold SEL LET for latchup of > 74.5 MeV•cm^2/mg.

The RTAXS devices were also tested to measure the error cross section under the above conditions. Each part was placed in the beam until 10^7 ions/cm^2 was reached. An average cross section per bit was determined for a given LET as the number of fault events observed divided by the total fluence of the associated run at that LET.

8. TESTING DIFFICULTIES 12/09 HEAVY ION TESTS

During December 2009 heavy ion testing with the A-Series DUT1 and DUT2 devices, core current (Icc) was observed to increase overtime. This is not normal behavior. Icc should stay steady. The increase in core current began to occur at LET = 8.6MeV*cm^2/mg. Figure 38 shows the current increase during a test run.

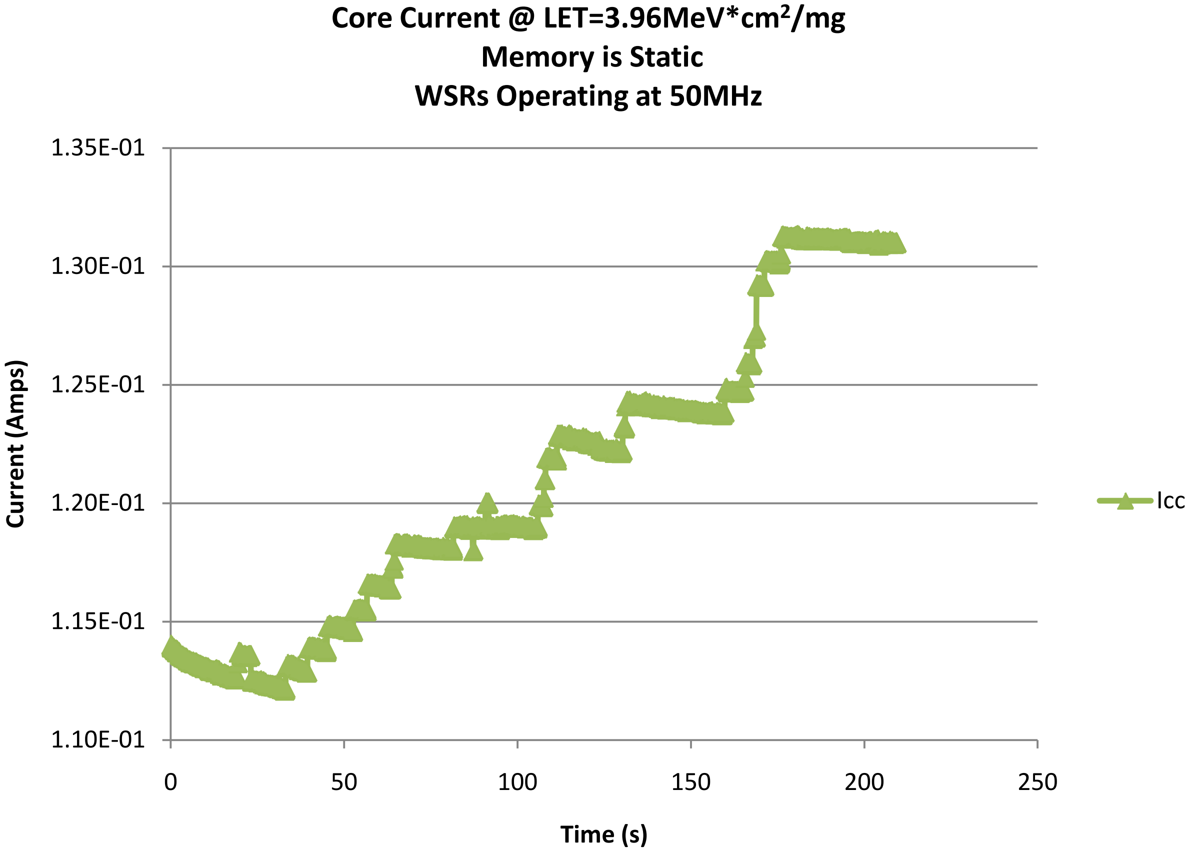


Figure 38: Example of Core Current increase during a test run. LET=3.96MeV*cm^2/mg. Memory was static hence no read, write, or scrub operations during irradiation. All WSRs were operating at 50MHz. Current increase was approximately 200mA.

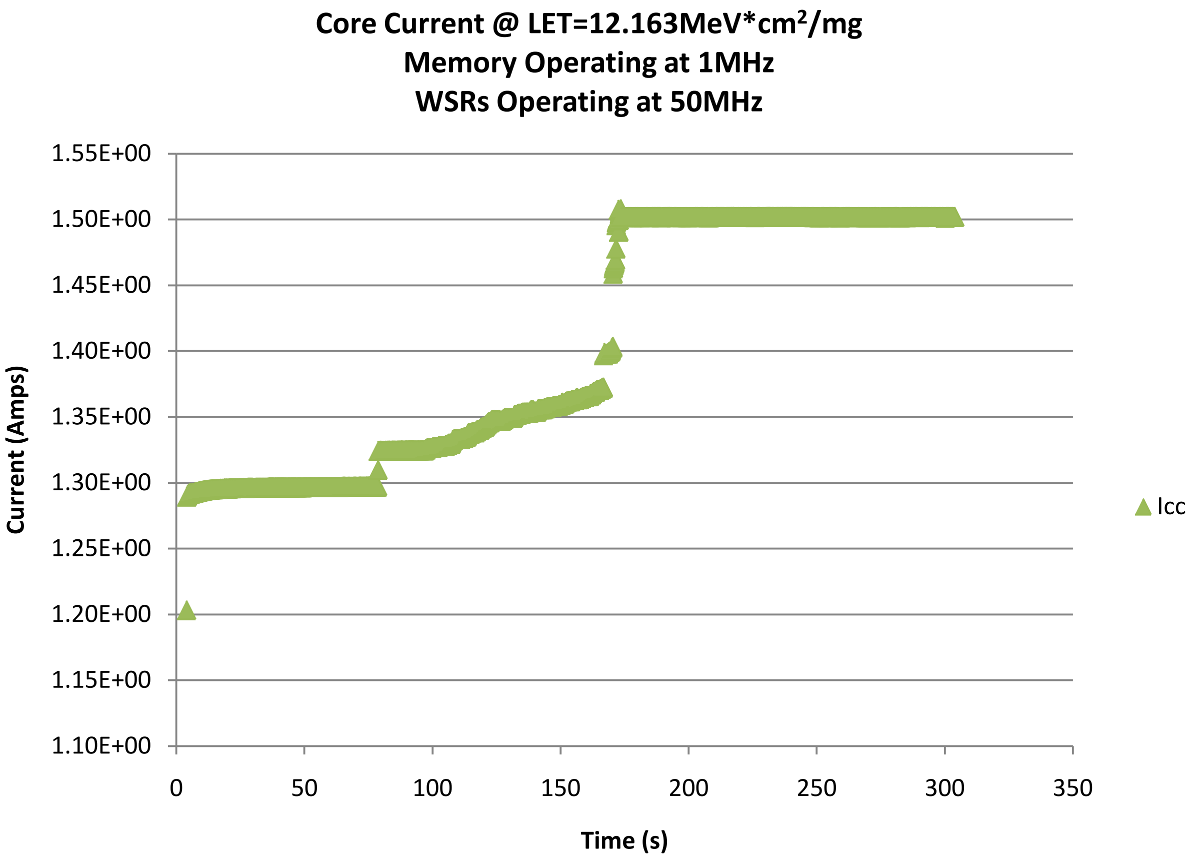


Figure 39: Example of Core Current increase during a test run. LET=12.1MeV*cm^2/mg. Memory read, write operations were occurring at 1MHz rates during irradiation. All WSRs were operating at 50MHz. Current increase was approximately 200mA.

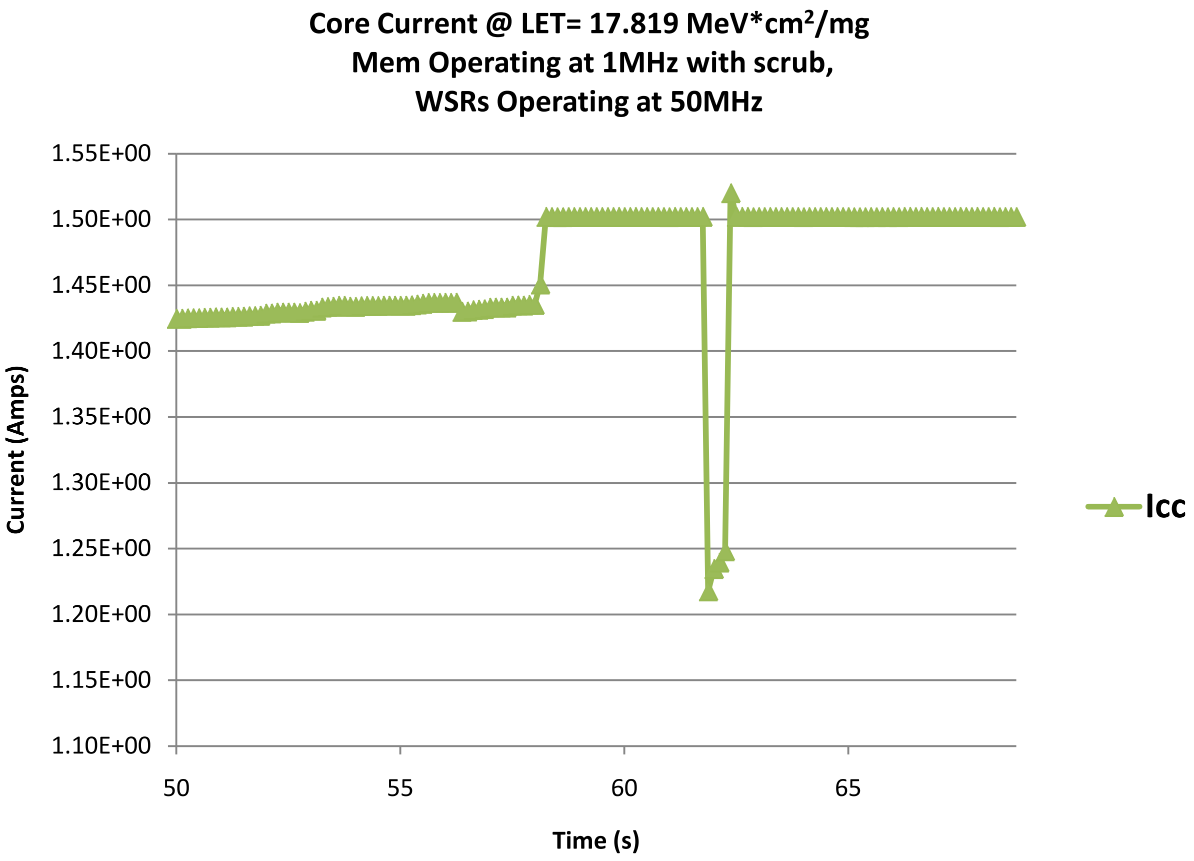


Figure 40: Example of Core Current increase during a test run. LET=17.1 MeV*cm^2/mg. Memory read, write operations were occurring at 1MHz rates during irradiation. All WSRs were operating at 50MHz. Current increase was approximately 200mA.

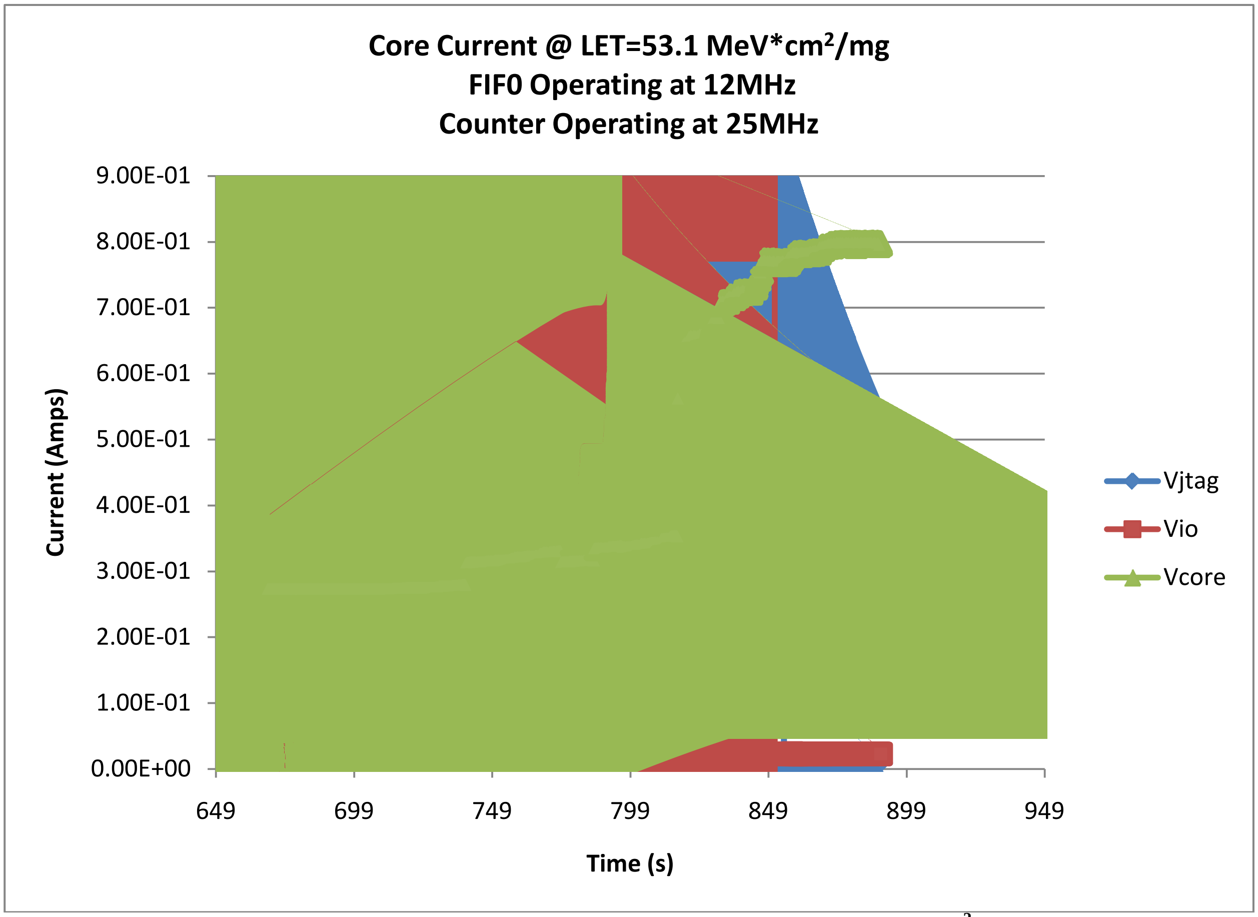


Figure 41 : Example of Core Current increase during a test run. LET=53.1 MeV*cm^2/mg. FIFO read, write operations were occurring at 12MHz rates during irradiation. All Counters were operating at 25MHz. Current increase was approximately 200mA.

Because of the unexpected response, it was decided to go back to testing with a new set of RTAX devices (B-Series). As previously described, the B-Series devices has slight changes in the global routing structures and sizing of WSRs. The B-Series DUT1 and DUT2 devices were tested August 2010. B-Series did not have a current increase response to radiation exposure. The following are bullets regarding the analysis:

- During initial REAG testing of RTAX2000s devices, a similar event was observed. The initial testing was performed in December 2005. Actel determined that the current rise was due to dangling nets created during the fusing/programming process. This was fixed in their 2006 release. DUT1 was created with a release from 2006. It was first thought that the use of the old release was the source of the problem. However, Actel confirmed that so changes to the RTAX2000s programmer were made since early 2006 and that the version used included the new fix. Additional tests were performed in 2006-2007 on a variety of RTAXs devices post fuse file creation fix, no current increases were observed. Hence, the version of the fuse file creation was not the source of the problem.
- Focus was placed on the difference between B-Series and A-Series. There were no differences in the memory structures. The differences resided in (1) WSR lengths (2) global routes. From 2005-2007 heavy ion testing, WSR strings lengths were varied with no problems in current

increase during radiation exposures. This leads REAG to deduce that the problem is most likely that the high fanout nets not placed on the global route can be the source of this issue. These nets were distributed to thousands of DFFs. As previously mentioned, if the global route is not used, a tree of buffers is created out of CCELLs. This tree helps to route high-fanout nets. If the CLKINT is used (B-Series), then no additional CCELLs are necessary for routing. The 2006-2007 DUTs had their resets equivalent to B-Series attached to the CLKINT tree. At this point it is assumed that it is the complex CCELL routing tree that is created due to misuse of the global routing in the A-Series that is causing the current rise.

- It is undetermined as to whether the current increase is due to an accumulation of particle strikes versus a SEE.
- Designers are urged to use the CLKINT for all resets and high fanout nets

9. WSR HEAVY ION TEST RESULTS

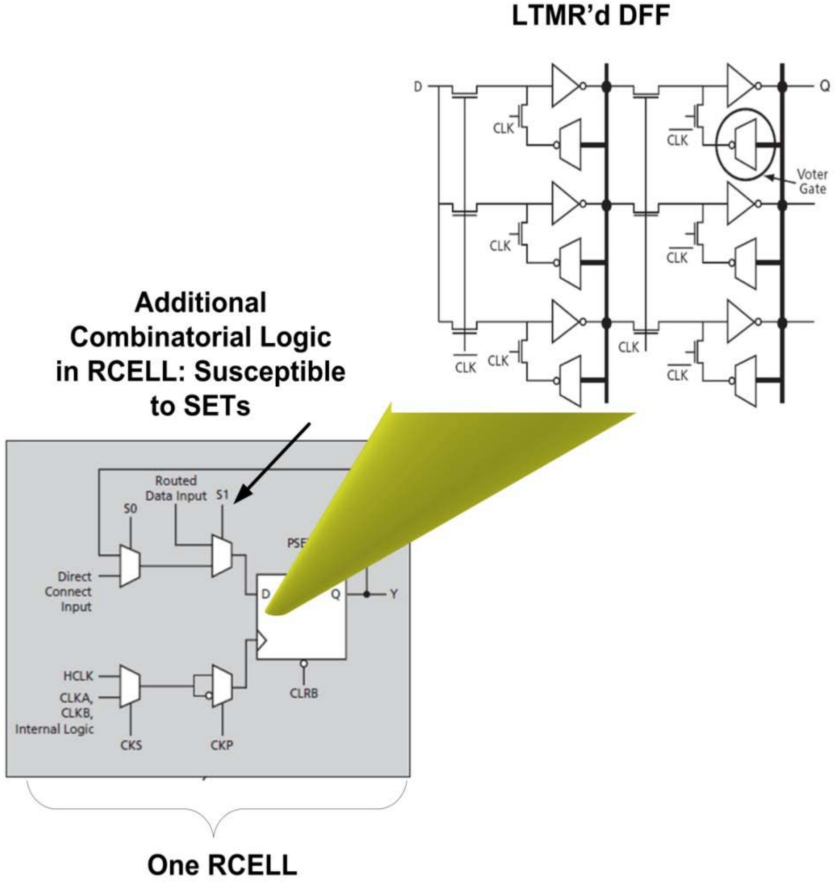


Figure 42: RTAX-s RCELL Details. RCELLS are susceptible to SETs because they contain non-redundant combinatorial logic prior to the data input of the LTMR. Subsequently, CCELLS are not necessary for SET captures, however, due to the increase in combinatorial logic, CCELLs increase SET generation and hence their probability of capture.

As previously mentioned, Actel RTAXs family of devices uses the LTMR embedded mitigation scheme. The mitigation is only effective for the DFFs – combinatorial logic is not mitigated. Figure 42 represents LTMR at a DFF. Because the data path is shared by the triplicated DFFs, a transient in the data path will not be able to be voted out. As a result, the effect of P_{DFFSEU} is null yet $P_{SET\rightarrow SEU}$ is significant. P_{SEFI} in this document will refer to SETs on global routes that upset groups of circuitry. It will be shown that P_{SEFI} is insignificant until high LET values and generally results in a burst (consecutive cycles of error). Equation (8) shows the probability of SEU given a LTMR design.

(8)

9.1 WSR SEU Cross Section with respect to LET

As the LET value increases, so does the transient width. For the N=0 WSR strings, a LET_{th}<5.6MeV*cm^2/mg has been observed. The saturation LET value is > 80MeV*cm^2/mg. LET versus SEU Cross Section is illustrated in Figure 43.

Although N=0 chains have no CCELLs between the DFFs, the chains contain combinatorial logic within each data path due to the “extra” combinatorial logic within each RCELL (see Figure 42). $P_{SET\rightarrow SEU}$ due to the RCELL combinatorial logic is the source of the errors in Figure 43.

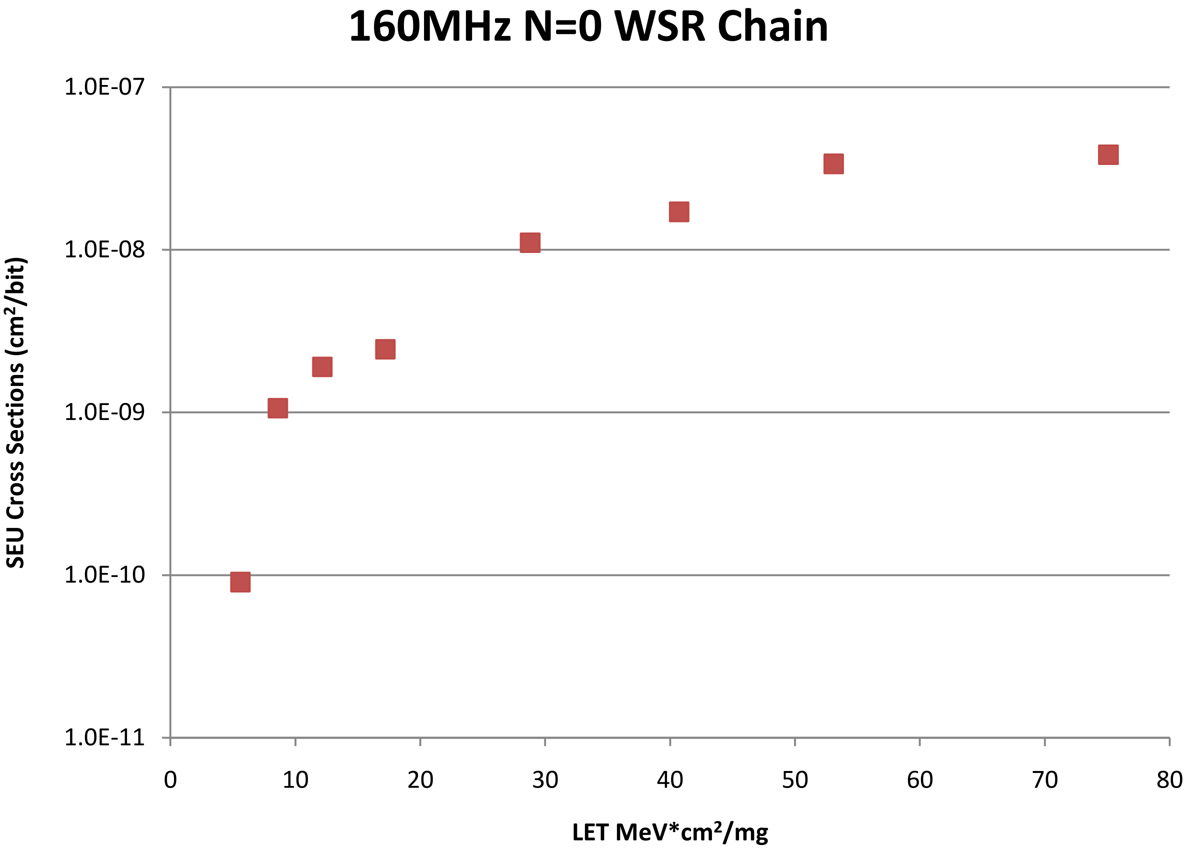


Figure 43: LET versus SEU Cross Section for 160MHz N=0 WSR

As previously mentioned, WSRs with different levels of combinatorial logic were tested at various frequencies. The following sections will describe the SEE signatures associated with SET capture, frequency effects, and architectural differences.

9.2 SET Generation, Propagation, and Capture Analysis using WSR Chains

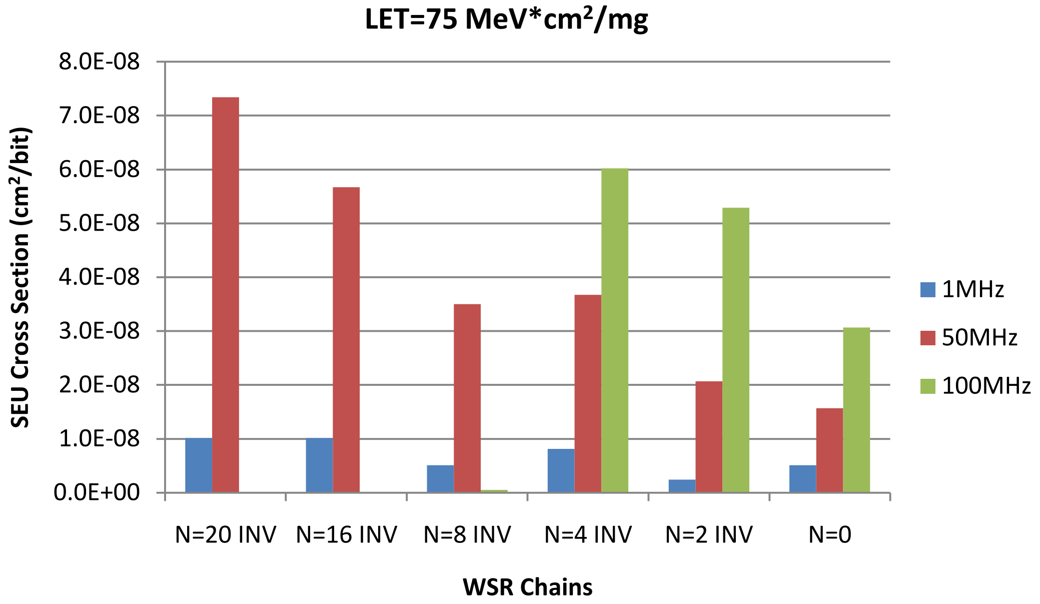


Figure 44: SEU Cross Sections at LET=75MeV*cm^2/mg per A-Series WSR Chains. Three Frequencies are shown (100MHz, 50MHz, and 1MHz). Only N=0,N=2, and N=4 can be operated at 100MHz. All WSRs can operate at 50MHz and 1MHz

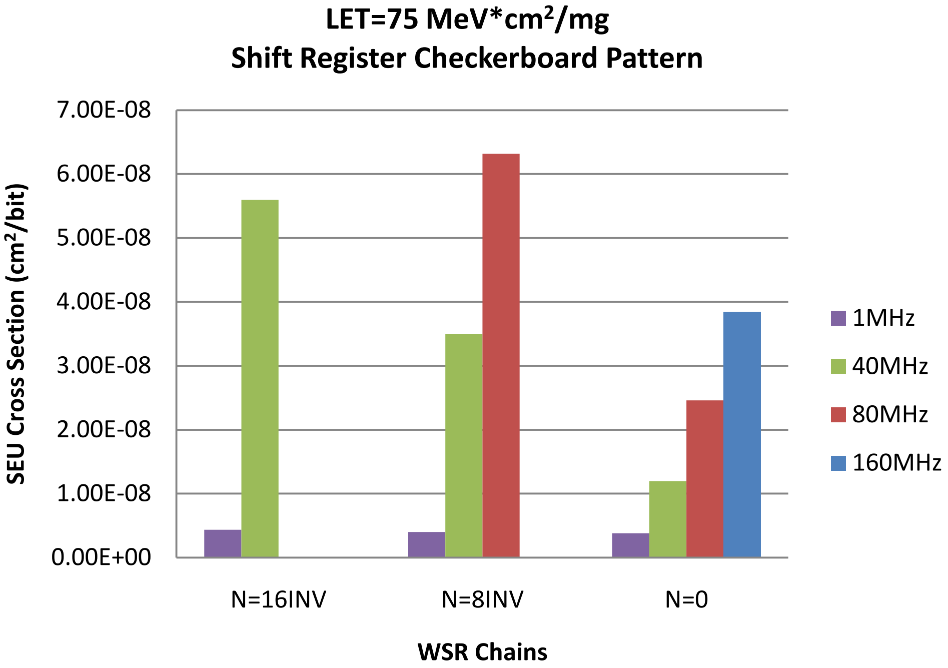


Figure 45: SEU Cross Sections at LET=75MeV*cm^2/mg per B-Series WSR Chains. Four Frequencies are shown (160MHz, 80MHz, 40MHz, and 1MHz). Only N=0 can be operated at 160MHz. Only N=0 and N=8 can be operated at 80MHz. All WSRs can operate at 40MHz and 1MHz.

DFFs capture the state of their data path at a specified clock edge (rising or falling) of its connected clock. Regarding data path SETs, capture of the SET is essential for a transient to become an upset ($P_{SET \rightarrow SEU}$). Hence, the faster the clock period, the more probable an upset that propagates to the destination DFF can get caught. In addition, increasing combinatorial logic, should (but not always due to capacitive loading) increase the potential sources of SETs and would in turn increase the SEU cross section. The following investigates such theories.

It is shown in Figure 44 and Figure 45 that for high LET values, as Frequency increases, the SEU cross sections increase. Such a frequency response is a well established concept for the RTAXs family of FPGA devices. It is also shown that as N increases so does the SEU cross section.

The summation of the number of CCELLs in Equation (3) suggests that as the number of combinatorial logic cells increase, more transients have the potential to be generated. The question is, can the transients propagate to their destination DFFs? Figure 44 and Figure 45 show:

- As N increases so does the SEU Cross Sections.
- With high LETs most SETs can propagate to their destination DFFs… i.e. SETs are not getting filtered while propagating to DFFs
- At low frequency, τ_s in Equation (3) becomes so large that any increase in SEU cross sections due to N is not obvious. Hence testing at low frequencies can mask many effects and responses.

High LET values create wider SETs with greater amplitude. Simply put, the transients have more energy. As LET decreases, $P_{generation}$ will also decrease. In addition, as LET decreases the size of the

SET will also decrease. This will affect $P_{propagate}$. SETs with less energy have a lower probability of propagating to their destination DFFs. Hence as the number of CCELLs increase and the number of routes increase, this can cause a filtration effect on SETs. The following figures demonstrate SEU response as LET is decreased.

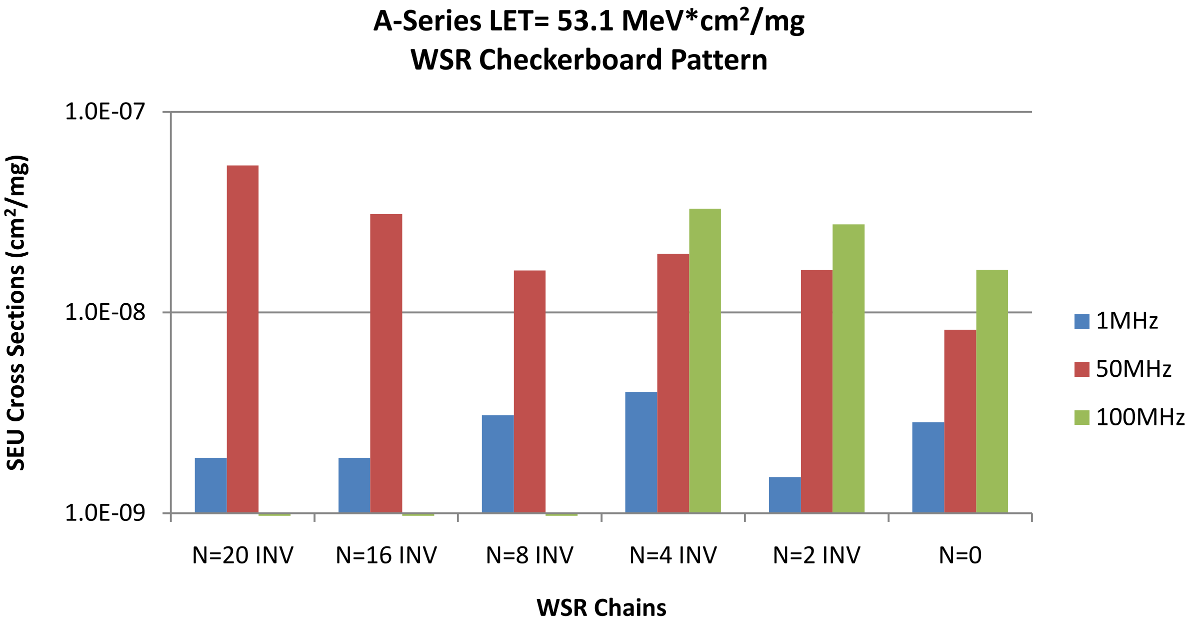


Figure 46: SEU Cross Sections at LET=53.1MeV*cm^2/mg per A-Series WSR Chains. Three Frequencies are shown (100MHz, 50MHz, and 1MHz). Only N=0,N=2, and N=4 can be operated at 100MHz. All WSRs can operate at 50MHz and 1MHz

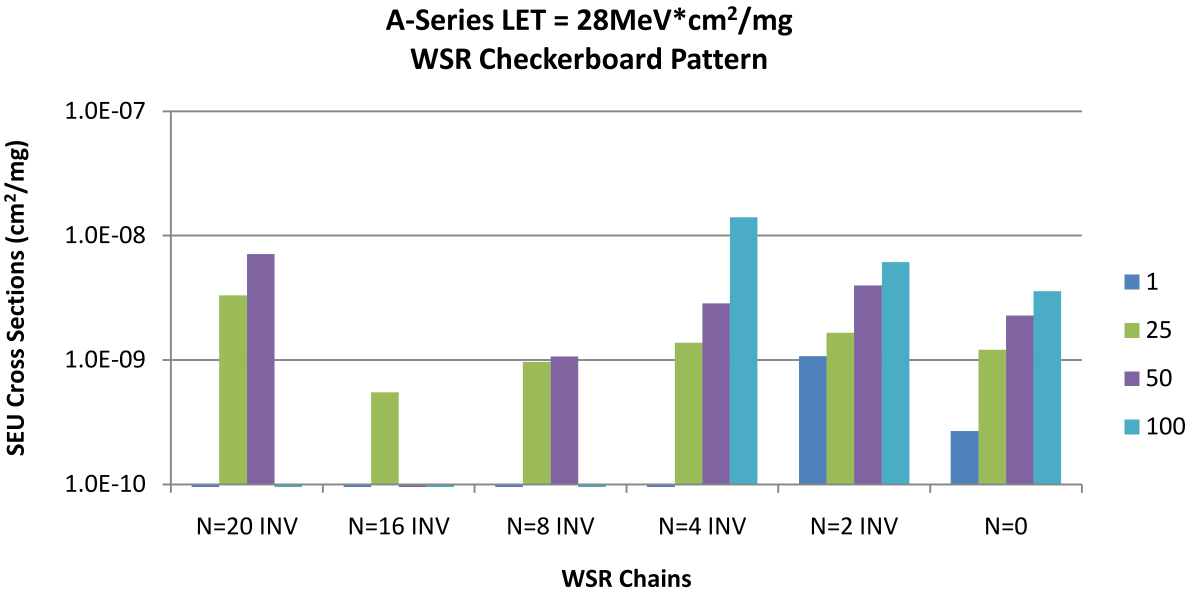


Figure 47: SEU Cross Sections at LET=28.1MeV*cm^2/mg per A-Series WSR Chains. Three Frequencies are shown (100MHz, 50MHz, and 1MHz). Only N=0,N=2, and N=4 can be operated at 100MHz. All WSRs can operate at 50MHz and 1MHz

With A-Series, N=2 and N=4 have increased SEU Cross Sections as compared to N=0. This shows that increasing the number of combinatorial logic cells can increase SEU Cross Section for frequencies >25MHz and LET≥ 28MeV*cm ^{2}/mg. It is interesting that near N=4 and N=8 INV chains, the SEU Cross Section starts to decrease. Filtration due to CCELLs and routes are taking affect.

Figure 48 illustrates B-Series WSRs. It appears that the filtration of the CCELLs and routes have more of a flattening effect of the SEU Cross Section. This is because the B-Series does not contain the N=2 and N=4 chains where the increase in SEU Cross Section is visible. In addition, A-Series and B-Series SEU Cross Sections have slightly different responses because the WSR chains have different place and routes topologies.

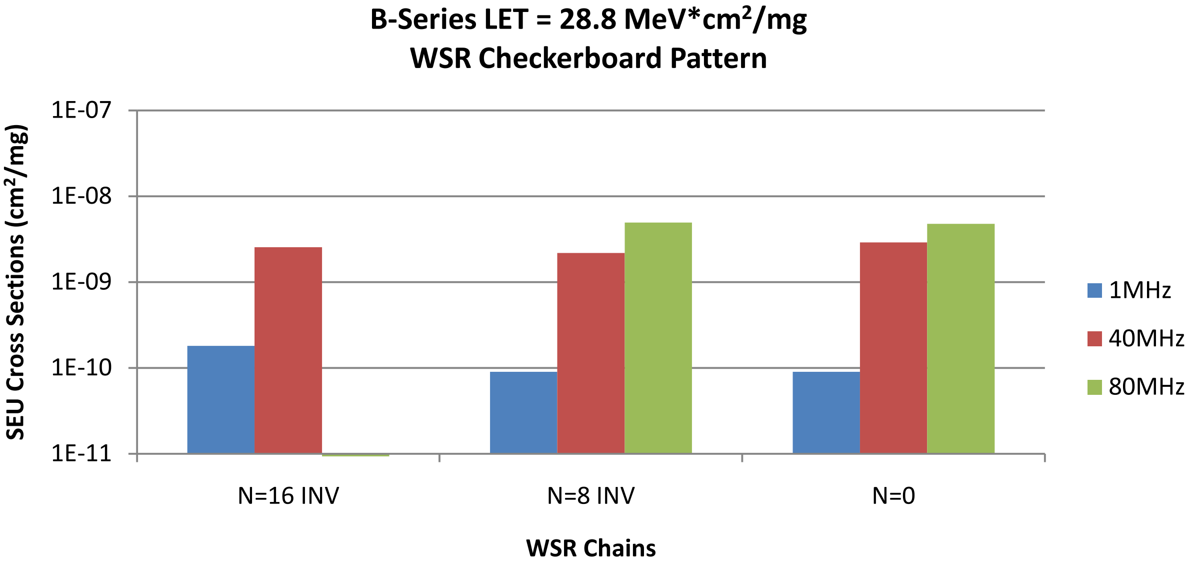


Figure 48: SEU Cross Sections at LET=28.8MeV*cm^2/mg per B-Series WSR Chains. Three Frequencies are shown (80MHz, 40MHz, and 1MHz). Only N=0 and N=8 can be operated at 80MHz. All WSRs can operate at 40MHz and 1MHz.

Fig. 49 and Fig. 50 further illustrate the SET filtration that occurs at lower LET values in the B-Series. The figures are a comparison of the N=0 and N=8 chains. It is shown that at lower LETs the N=0 chains have a higher SEU Cross section and a lower LET$_{th}$ than the N=8 WSR. Near LET=28.8MeV*cm^2/mg, the SEU cross sections statistically merge. The N=8 WSR SEU Cross Sections with LET>28.8MeV*cm^2/mg are significantly larger than the N=0 WSR. Once again, the SETs at higher LETs are not getting attenuated, hence as N increases so does the SEU Cross section as expected.

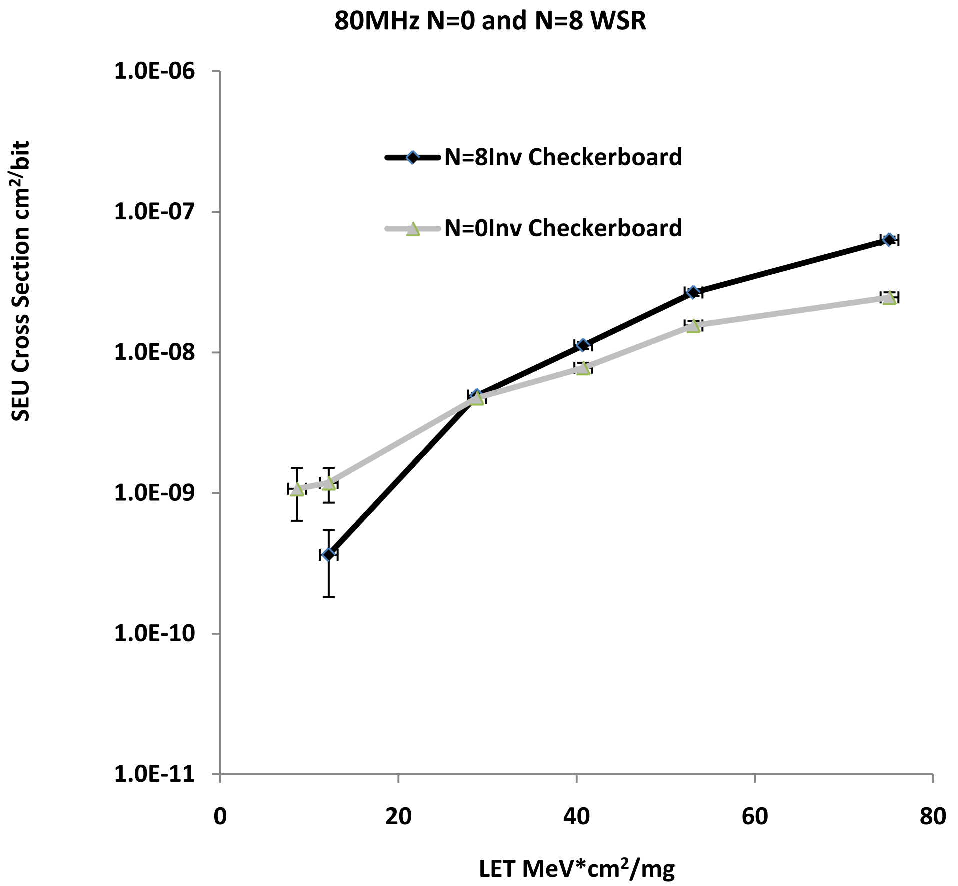


Fig. 49: 80MHz Shift Registers with 0 Inverters (N=0) and 8 Inverters (N=8). Data input is checkerboard. At lower LETs, N=0 chain has a higher SEU cross section than N=8. Suggests transients are filtered within combinatorial logic chains. Error bars are one-sigma.

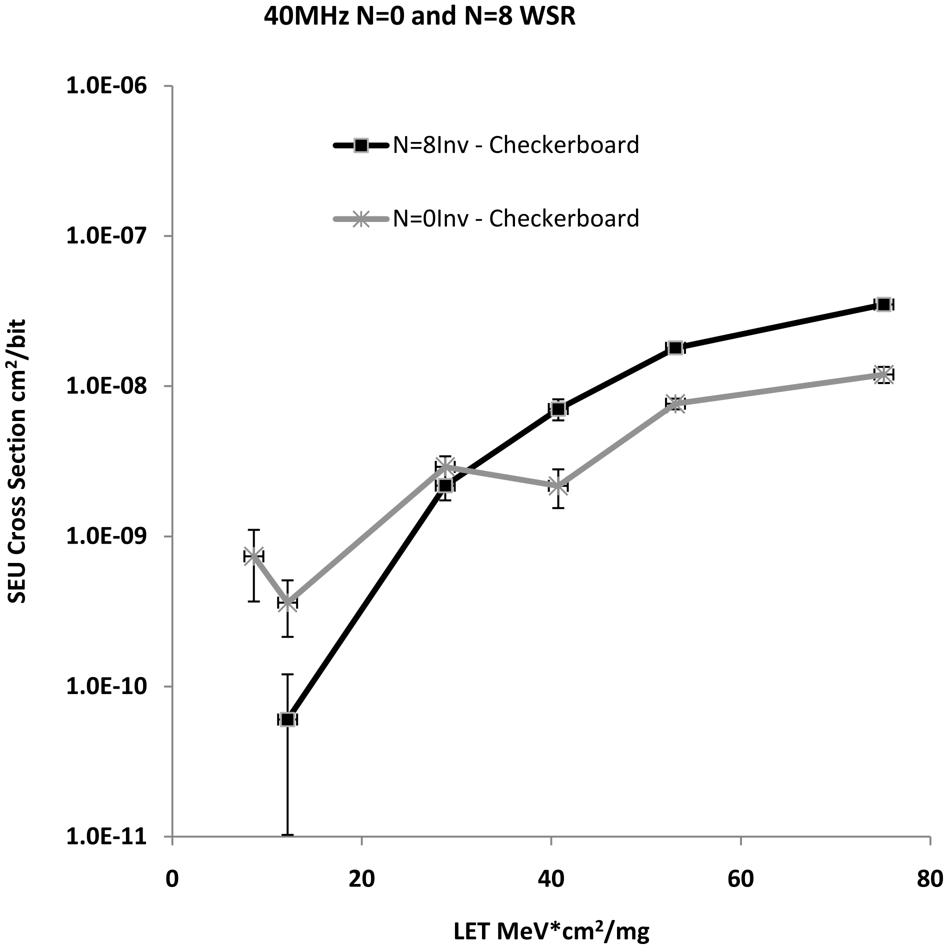


Fig. 50: 40MHz Shift Registers with 0 Inverters (N=0) and 8 Inverters (N=8). Data input Is checkerboard. At Lower LETs, N=0 chain has a higher SEU cross section than N=8. Suggests transients are filtered within combinatorial logic chains. Error bars are one-sigma.

9.3 WSRs and Routing Effects

A design is created using VHDL. The VHDL is synthesized into gate logic specific to logic cells within the target FPGA. The gate logic is then placed and routed within the target FPGA. As the length of a route increases, so does the capacitance of the route. Hence cells placed far apart from each other can have a filtering effect on SETs. It is important to note, that although long routes can filter SETs, it is not a recommended approach to SET mitigation. The main reason not to rely on separation as mitigation is that a long route will decrease operation speed, increase power, and increase area required for the target design. Circuit efficiency is considerably lost without distinct assurance that a significant number of SETs will be filtered.

Because of the difference in global routing structures within the A-Series versus B-Series WSR chains, their routing has been substantially affected. Consequently, it is assumed that their SEU Cross sections would also be affected. Figure 51 through Figure 53 show comparisons between the A-Series WSRs and the B-Series WSRs. At high LET values where filtering is minimal, it is shown that the A-

Series and B-Series are statistically equivalent. It is interesting to note that as LET is decreased, SET filtration increases, and the differences between the two series becomes more significant.

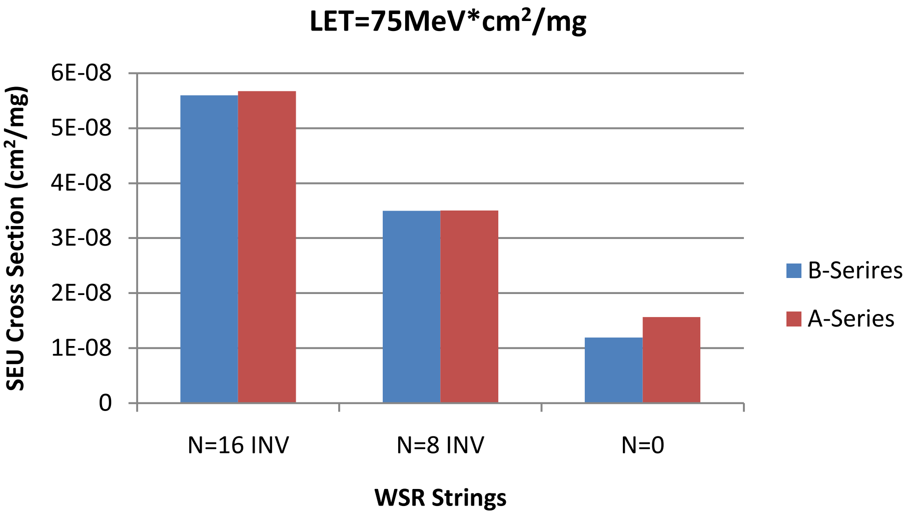


Figure 51: Comparison between A-Series and B-Series WSR Chains at 75MeV*cm^2/mg. WSR chains illustrated are N=16INV, N=8INV, and N=0

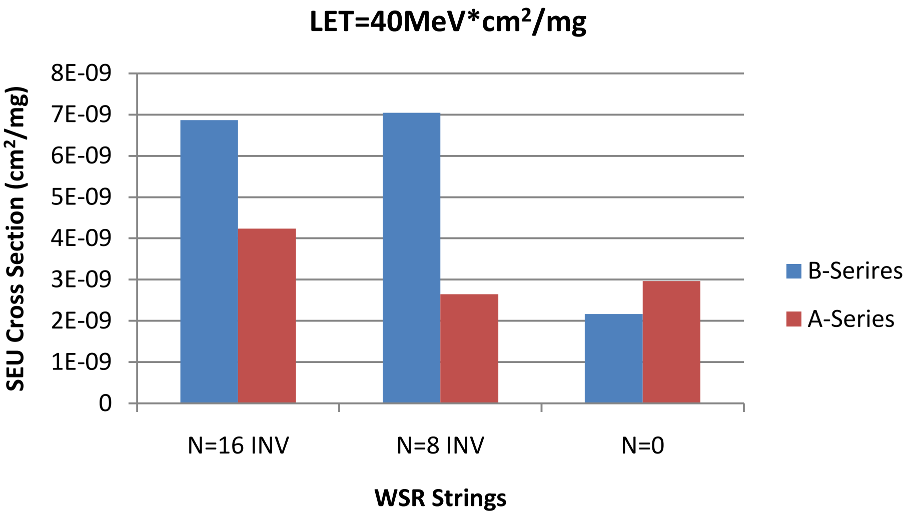


Figure 52: Comparison between A-Series and B-Series WSR Chains at 40MeV*cm^2/mg. WSR chains illustrated are N=16INV, N=8INV, and N=0

It is interesting to note that when investigating the routing effects of the A-Series and B-Series, it was observed that the A-Series N=8 and N=16 had considerably longer route lengths than the B-Series N=8 and N=16 WSRs. This corresponds to Figure 52. A-series N=8 and N=16 have longer routes; hence the A-series has lower cross sections.

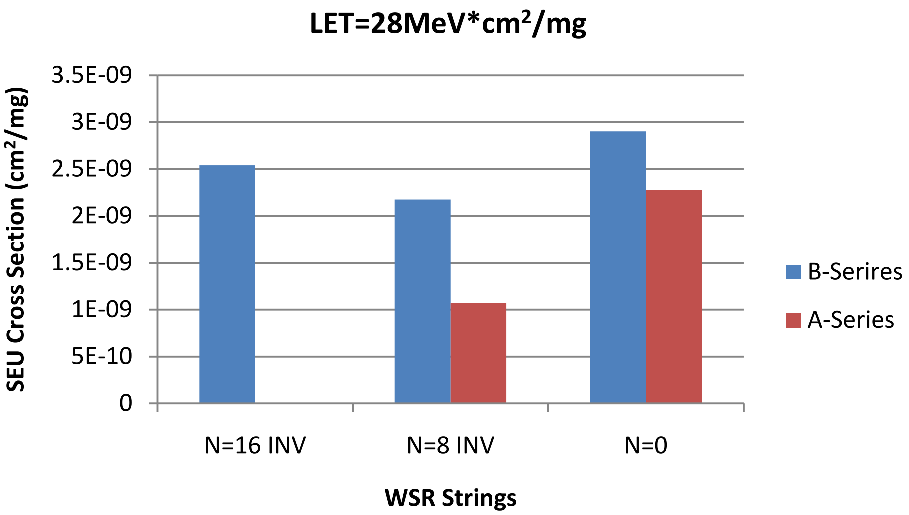


Figure 53: Comparison between A-Series and B-Series WSR Chains at 28MeV*cm^2/mg. WSR chains illustrated are N=16INV, N=8INV, and N=0

9.4 Data Pattern Effects

In order to investigate data pattern effects, comparisons are made between static data inputs (such as all 0's) and checker board (data changing every clock cycle). Static data input yielded significantly lower error cross sections than the alternating data pattern for all shift register strings at higher frequencies. However, evaluating chain 0 at lower frequencies, suggests that the difference between the static input SEE cross sections and the checkerboard SEE cross section becomes smaller. In other words the rate of data change at lower frequencies does not impact circuits with few CCELLS (chain 0 has no CCELLS). Data pattern effects also appear to become less significant as LET is increased.

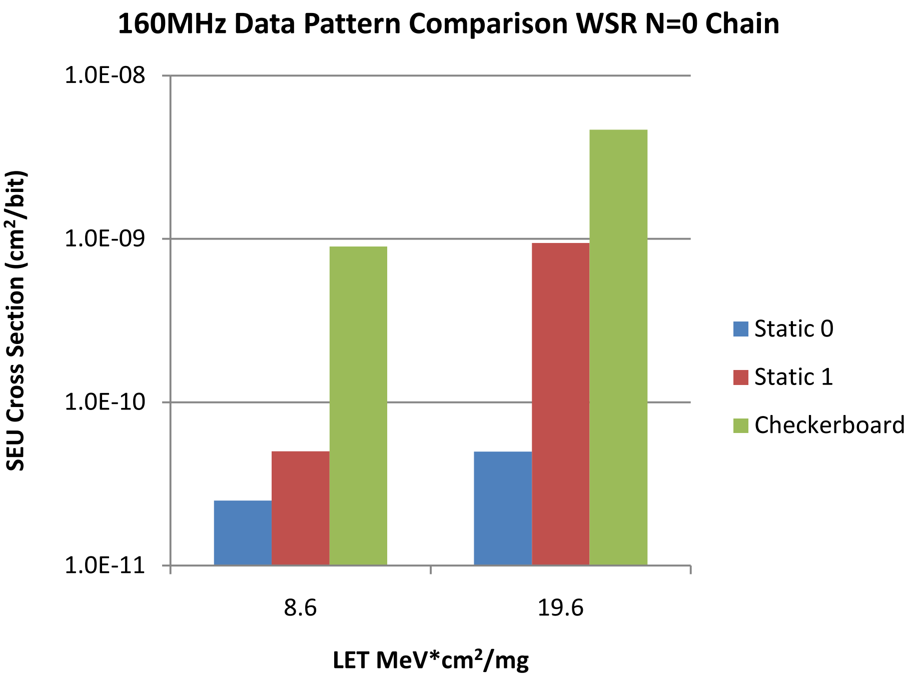


Figure 54: High Frequency Data Pattern Comparison at 160MHz. Only the N=0 Chain is shown because they are the only chains that operate at 160MHz

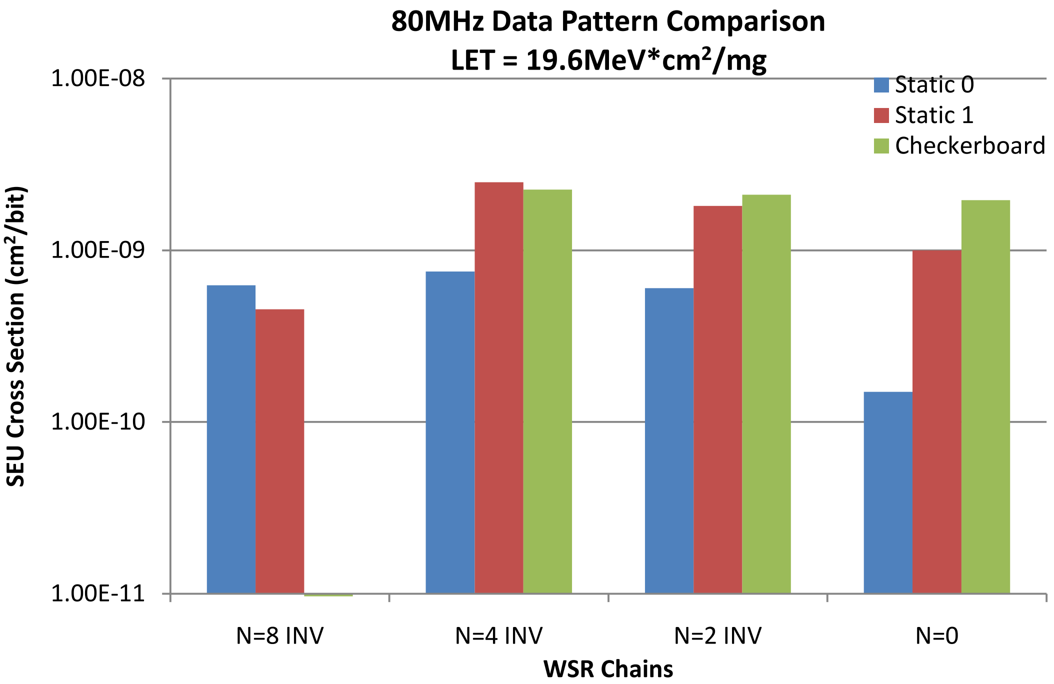


Figure 55: High Frequency Data Pattern Comparison at 80MHz. N=0 thru N=8 are illustrated. N=16 can not operate at 80MHz

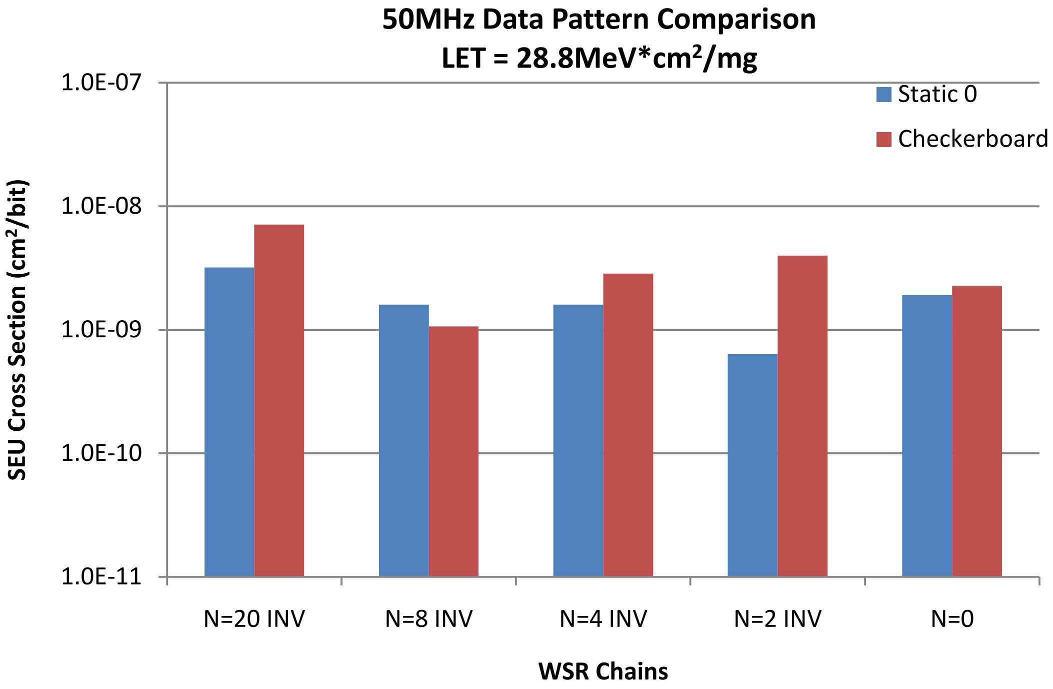


Figure 56: Lower Frequency Data Pattern Comparison at 50MHz and LET = 28.8MeV*cm^2/mg.

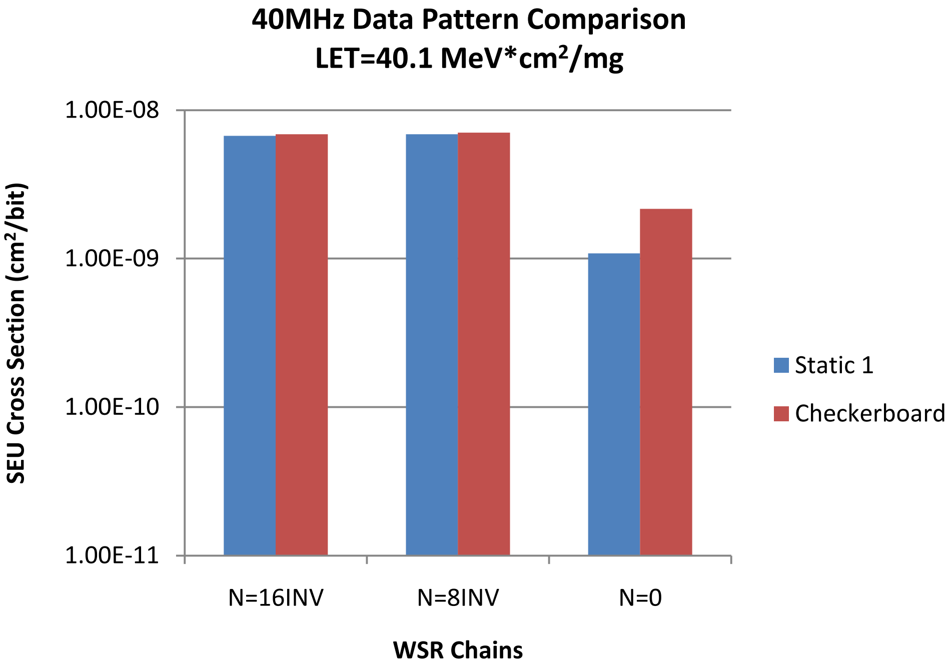


Figure 57: Lower Frequency Data Pattern Comparison at 40MHz and LET = 40.1MeV*cm^2/mg.

9.5 CCELL Effects (Inverters versus Buffers)

As shown in Figure 58 and Figure 59 inverters tend to have a higher SEU Cross Section than Buffers for LETs >28.8MeV*cm^2/mg. As LETs decrease, filtering effects exist and the CCELL cross sections tend to statistically approach each other.

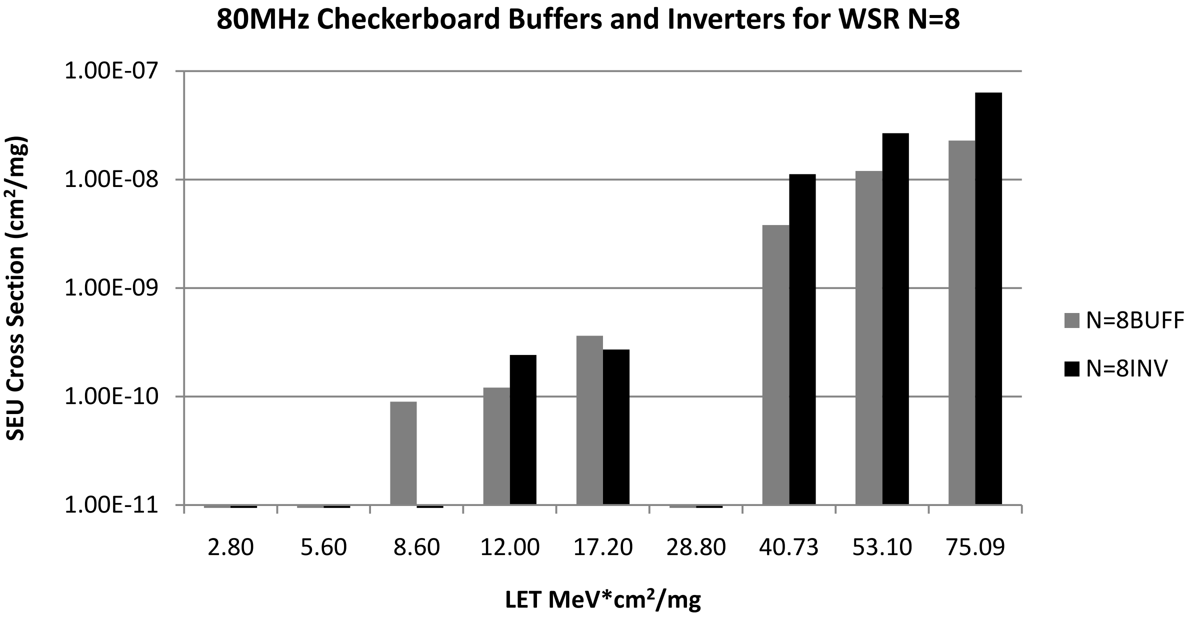


Figure 58: Comparison of Inverters and Buffers for 80MHz-Checkerboard WSR N=8

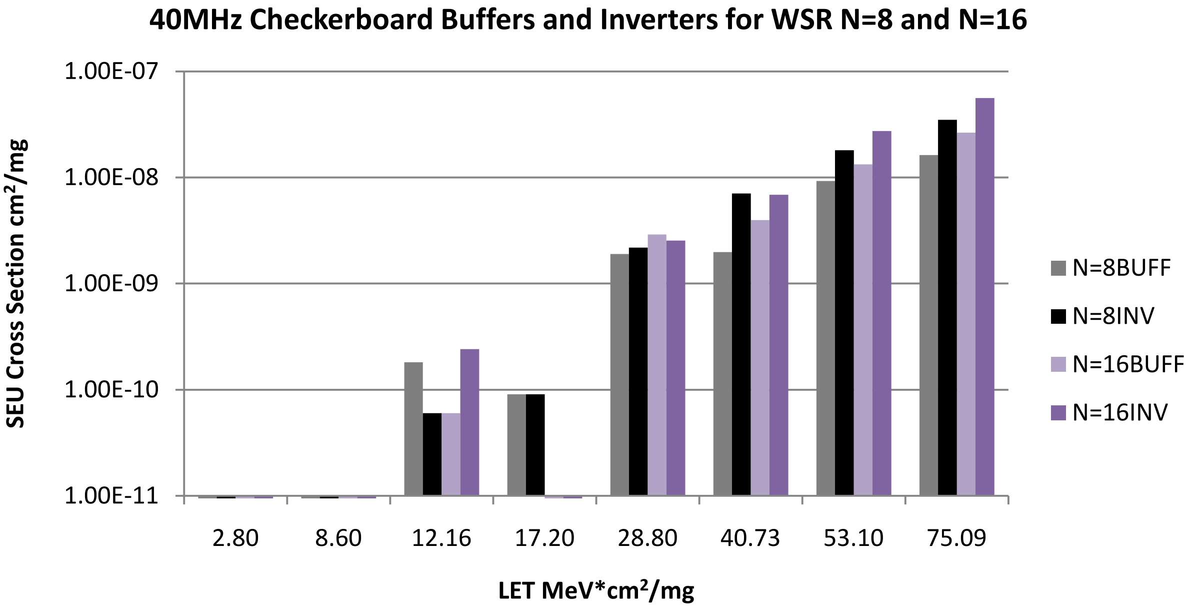


Figure 59: Comparison of Inverters and Buffers for 40MHz-Checkerboard WSR N=8 and N=16

9.6 Static WSR Tests

Static WSR tests were performed by loading the WSR strings with a given pattern. Afterwards the all WSR clocks are stopped. During irradiation all clocks are stopped. After irradiation, the clocks are turned on to read the WSRs. No upsets were observed during the static testing across all LET values.

10. COUNTER HEAVY ION TEST RESULTS

As previously mentioned, traditional FPGA SEU characterization has been performed with WSR test structures. Such test structures are a simplified representation of actual circuitry that would normally be implemented within a FPGA. A WSR is considered to be a linear design because each node within the circuit only has one input and one output (i.e. Fan-Out = Fan-In = 1). On the other hand, average architectures are not linear. They contain nets with fan-out and fan-in >1. A node's fan-out will change both the capacitive loading of cells and the utilization of routing resources within FPGA fabrics. Depending on the rise/fall time and width of a Single Event Transient (SET) the capacitive loading of a cell can filter away the SET. On the other hand, a SET that is not filtered can fan-out to multiple nodes and have the effect of a multiple bit upset. Regarding nodes that have fan-out=1 or have minimal capacitive loading, it has been proven that SETs can increase in width as they traverse a circuit.

Figure 60 is a graph of the average overall bit upset rate of the counters (Counterσ_{mean}) operating at 30MHz, 60MHz, 120MHz. Counterσ_{mean} is calculated (9) by summing all upsets across all counter bits normalized by particle fluence per LET and total number of counter bits (C_{TOT}=4800) in the array.

(9)

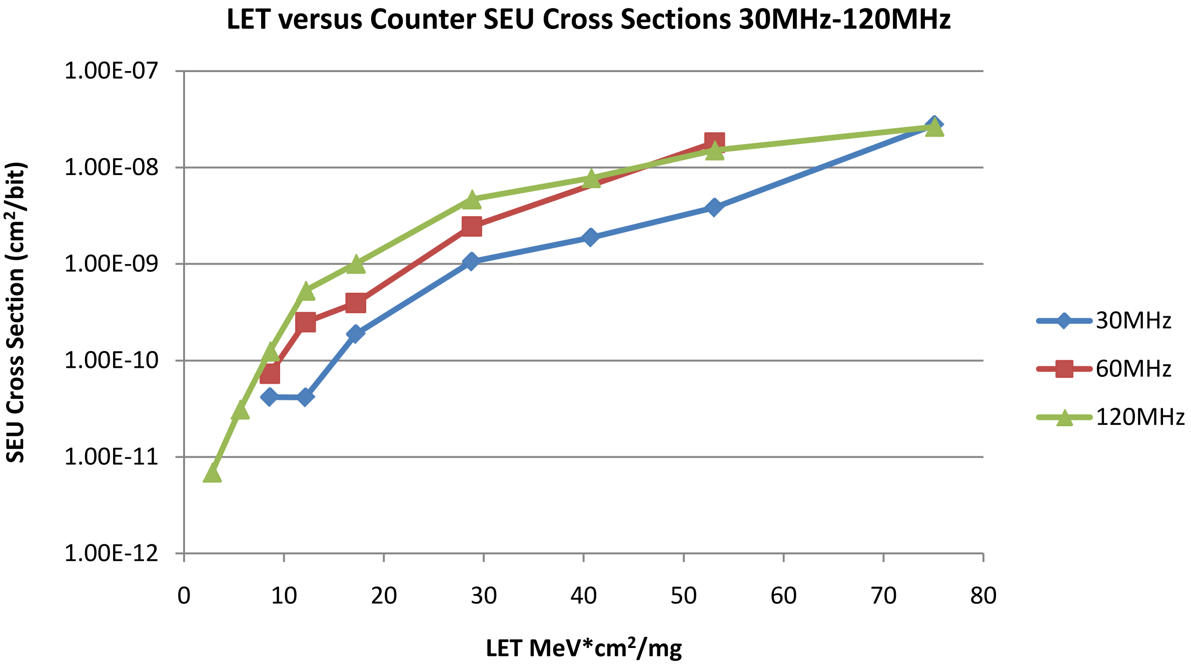


Figure 60: Counter SEU Cross Sections 30MHz-120MHz. LET$_{th}$ at 120MHz was observed to be <2.8MeV*cm^2/mg

As shown in Figure 60, as LET increases, the cross sections seem to merge. This is because the ration of SET widths to clock frequency is larger. During 120MHz heavy ion tests upsets were observed at LET=2.8MeV*cm^2/mg. This is the lowest LET$_{th}$ ever recorded. The following describes why counter tests can incur lower LET$_{th}$ than WSR tests.

10.1 Single Bit Upsets, Multiple Bit Upsets, and Fan-out

As previously mentioned, the inherent design of a counter contains fan-out. In the case of this DUT-counter study, the fanout was located at each DFF and was utilized to calculate look-ahead carry logic. Because there is fanout, it is interesting to investigate the probability of one SET propagating to multiple nodes. As a warning, this study is limited because fanout is only located at the DFFs (after a RCELL). Most of the SETs are generated within the CCELLs. Consequently, this is not a complete study of SETs and their ability to propagate to multiple paths. If more research is preferred, then an alternative counter design (or other similar type design) should be tested that contains fanout at the CCELLs.

Figure 61 thru Figure 63 illustrate the cross sections per LET based on one particle strike. As previously mentioned, it is desired to determine if this particle strike can propagate to multiple nodes in the counter. It is shown that at higher frequencies, SETs are able to propagate down multiple paths and be captured by destination DFFs. It is also shown that such events have a significantly lower probability than occurring with respect to the total cross section. As previously mentioned, this phenomenon is due to the limited location of fanout routes within the implemented counter design. Again, if the counter were implemented with fanout at the CCELLs nodes, there would probably be an increase in MBU cross sections. At the same time one must take into account that the fanout located at CCELLS can potentially attenuate the SETs.

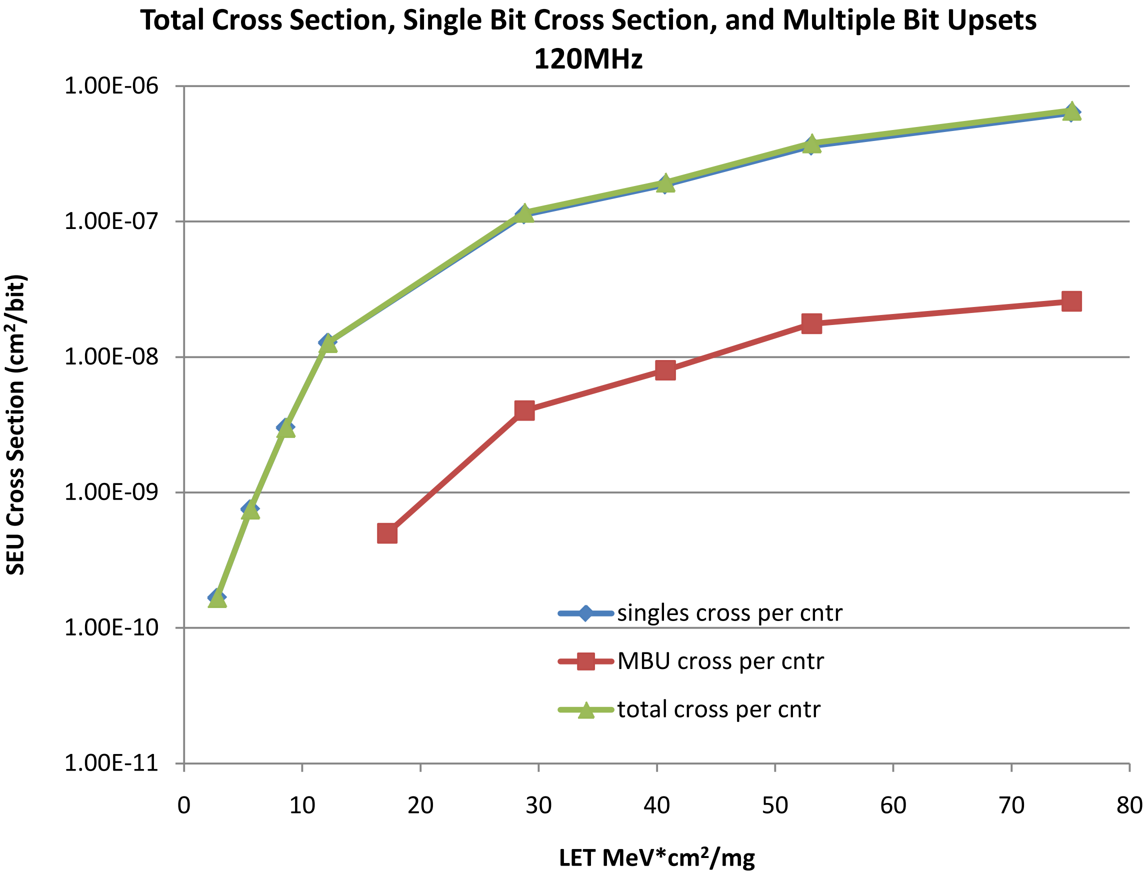


Figure 61: 120MHz Counter SEU Cross Sections. Total SEU Cross Section is broken down into Single Bit Upsets and multiple bit upsets. It is shown that multiple DFFs can upset from a single particle; however, they make up a small percentage of SEU upsets.

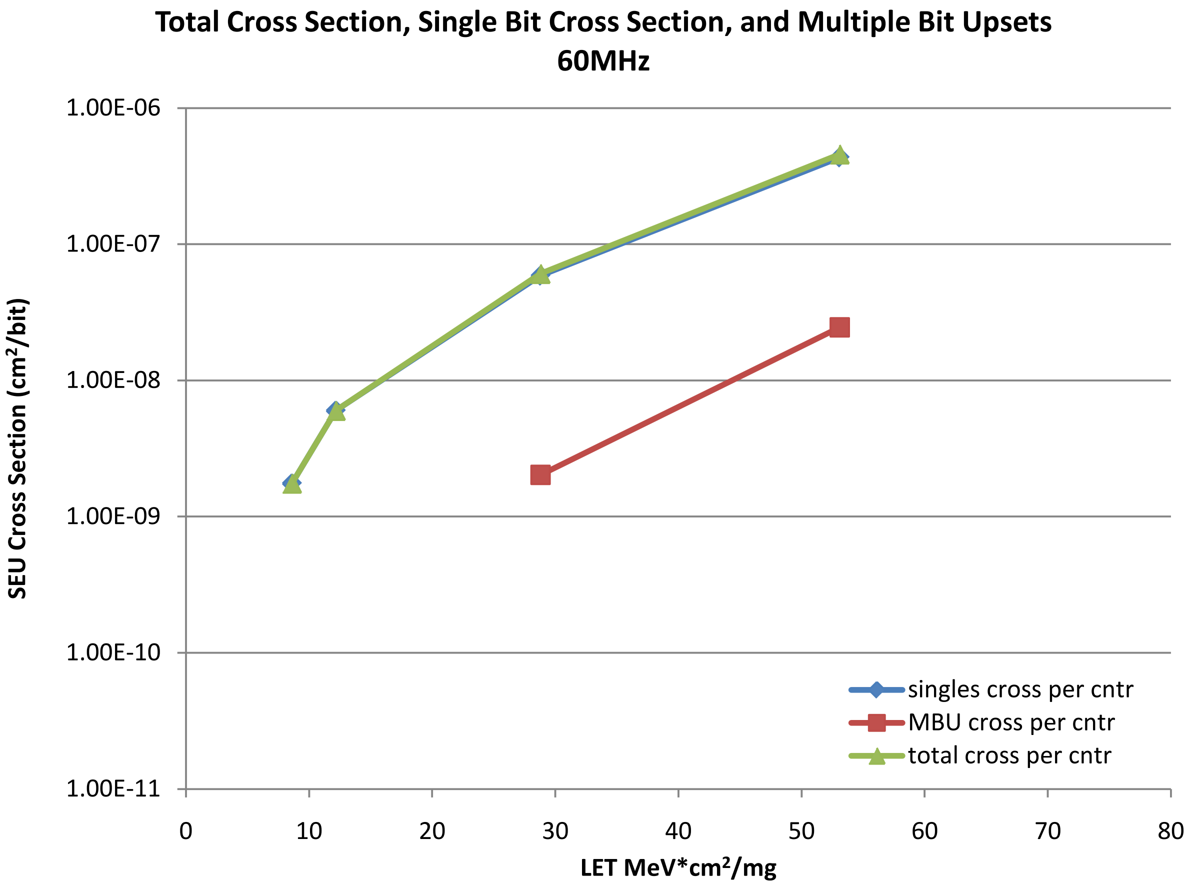


Figure 62: 60MHz Counter SEU Cross Sections. Total SEU Cross Section is broken down into Single Bit Upsets and multiple bit upsets. It is shown that multiple DFFs can upset from a single particle; however, they make up a small percentage of SEU upsets.

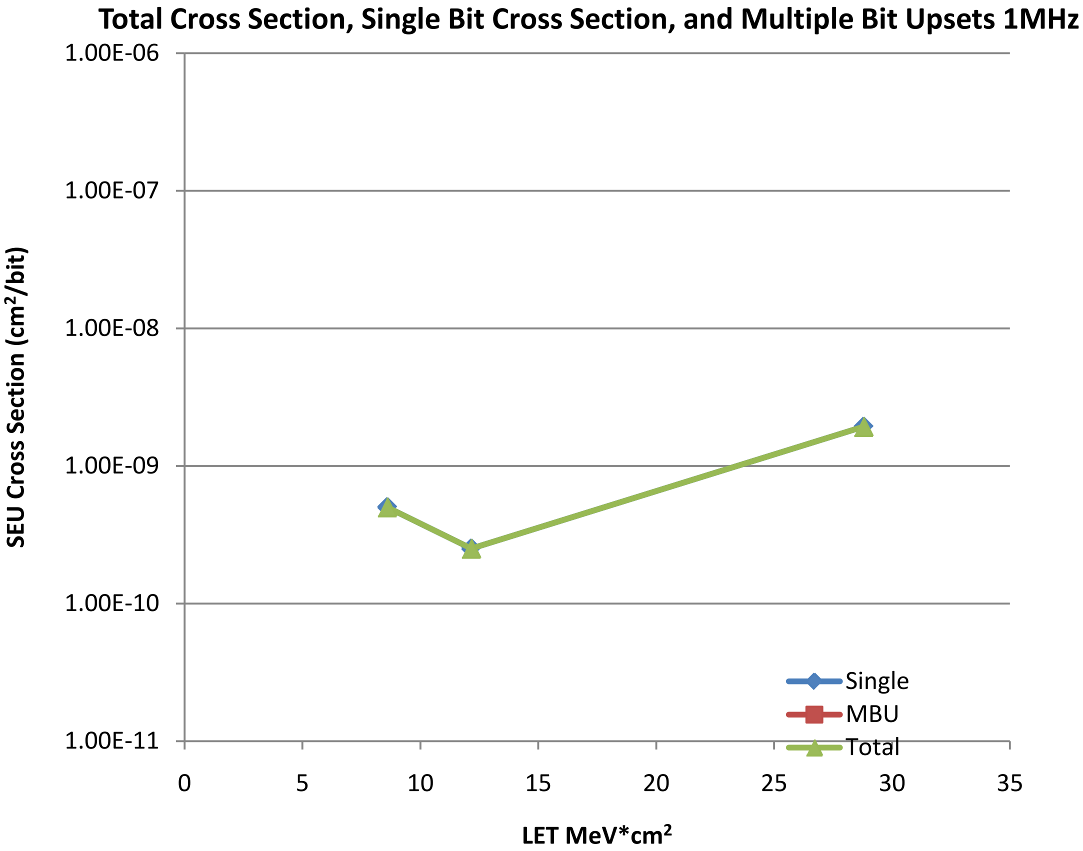


Figure 63: 120MHz Counter SEU Cross Sections. Total SEU Cross Section is broken down into Single Bit Upsets and multiple bit upsets. Multiple DFF were not observed to upset upon a SET.

10.2 SET Propagation and DFF Fan-IN

By definition, increasing the fan-in to a DFF increases the number of data paths that affect the state of the DFF. Consequently, the increase of data paths essentially increases the number of effective CCELLs that can contribute to SEU cross sections as illustrated in Fig. 64. This is a valid point for all LET values. The following sections describe the process of investigating this theory via SEU testing of complex designs.

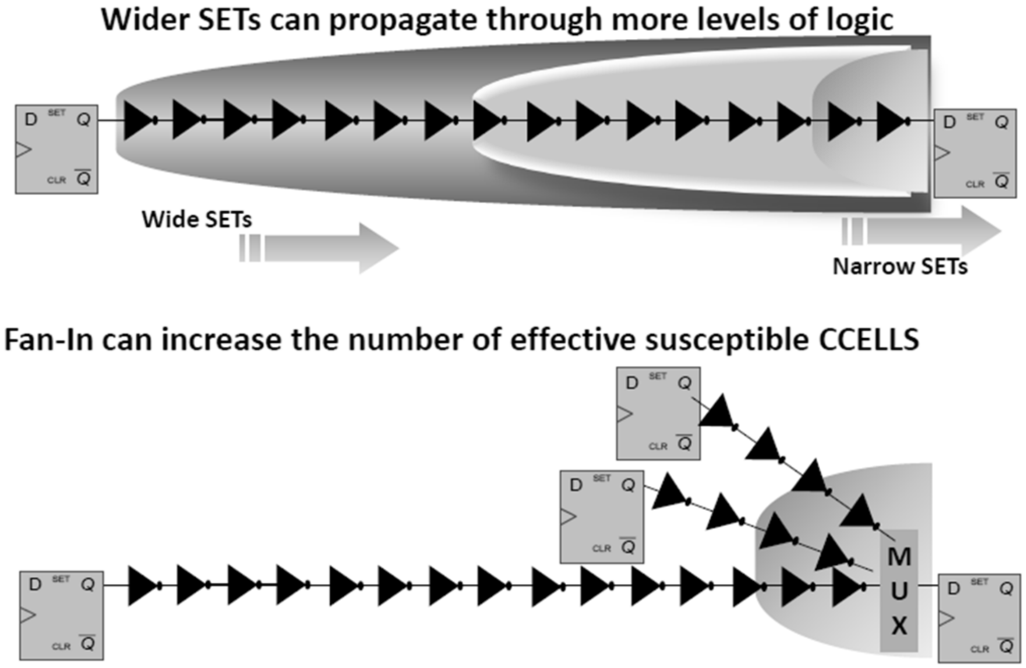

Wider SETs can propagate through more levels of logic
D SET Q
CLR Q
Wide SETs
Narrow SETs
Fan-In can increase the number of effective susceptible CCELLS
M U X

Fig. 64: Propagation paths to destination DFFs and the increase in effective $P_{SET\rightarrow SEU}$ susceptible CCELLS due to fan-in

10.3 Comparing Counters to WSR Chains

Before a comparison is performed, it is important to note that it has been shown that data pattern is a significant factor regarding SEU cross sections. The faster the data pattern changes, the higher the SEU cross section. This is valid across all LET values and was verified via testing WSR checkerboard patterns versus WSR static data-patterns. Because each WSR data path is linear and equivalent, all WSR SEU bit-cross sections are equally distributed across each DFF.

As an alternative to WSR structures, counter data paths are not equivalent to each other and are not linear. An effect of the non-linear, co-dependent counter data paths is that they produce unique data pattern rates per DFF. Due to the binary incrementing nature of a counter, the lowest order bit (LSB) has a checkerboard pattern similar to that of a checkerboard shift register. The LSB uniquely has the highest data-pattern rate in the counter. As the bit order increases, the bit data-pattern rate decreases by a factor of 2. Consequently, it is expected that the average DFF upset rate of a counter will be less than that of a WSR checkerboard.

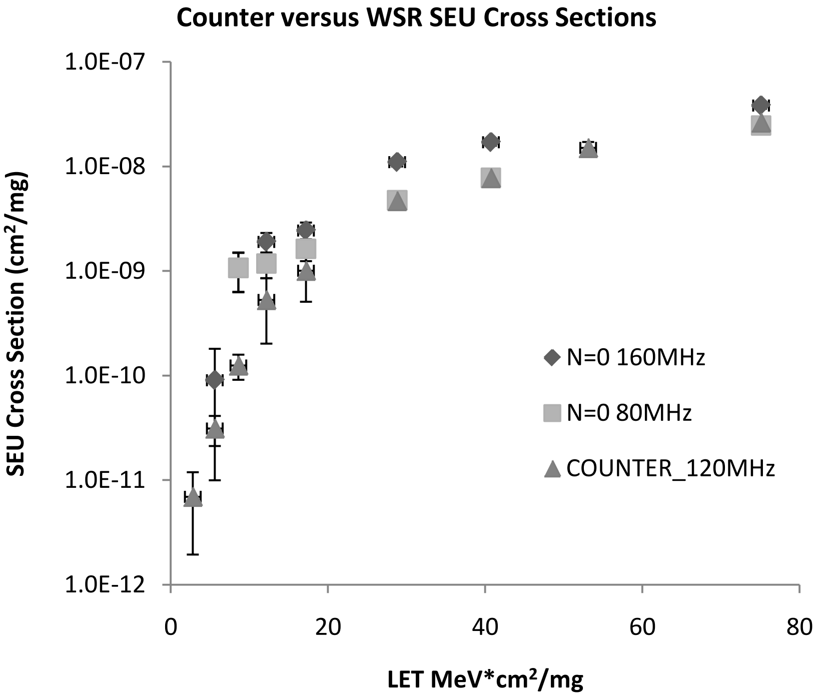


Fig. 65: A comparison of Counters and Windowed Shift Registers. LET versus SEU Cross Section. Error bars are one-sigma.

Fig. 65 is a graph of the average overall bit upset rate of the counters (Counterσ_{mean}) operating at 120MHz versus WSR checkerboard (WSR$\sigma_{checkerboard}$)[2] operating at 160MHz and 80MHz. Counterσ_{mean} is calculated (10) by summing all upsets across all counter bits normalized by particle fluence per LET and total number of counter bits (C_{TOT}=4800) in the array.

(10)

As previously mentioned, the overall data-pattern rate of the counter DFFs are lower than the WSR DFF checkerboard pattern. It follows that each Counterσ_{mean} be lower than each WSR$\sigma_{checkerboard}$ per LET running at a similar frequency. This has been verified during heavy ion testing and is illustrated in Fig. 65. The 120MHz Counterσ_{mean} seems to mirror that of an 80MHz WSR$\sigma_{checkerboard}$, yet is lower than the 160MHz WSR$\sigma_{checkerboard}$.

The SEU cross section comparisons illustrated in Fig. 65 reinforce the established RTAX-S data-pattern theory. However, this theory does not address the fact that the counters have a lower LET_{th} than the checkerboard-WSR test structures. In order to explain the low counter LET_{th}, a bit-by-bit analysis of the counters is necessary. The following section provides such analysis.

10.4 Fault Isolation and Counter Bit Analysis

The nature of the implemented counter architecture implies that as the bit order increases, the fan-in also increases. As previously mentioned, fan-in is expected to be a significant factor regarding SEU cross sections. Hence, a more detailed analysis is directed towards the individual counter bits. Instead of averaging SEU cross sections across all counter bits as in Counterσ_{mean}, DFFs are binned by counter bit order, 4 bit orders to a bin, and an average SEU cross section is calculated per bin (Counterσ_{bin}).

Counterσ_{bin} (11) is normalized by particle fluence and by the number of DFFs within a bin (C_{BIN} = 800).

(11)

Fig. 66 demonstrates binned counter cross sections at high LET values. At these LET values, most SETs are able to propagate without attenuation to their destination DFFs. The first 3 Counterσ_{bin} values in Fig. 66 pertain to Counter bit0 thru Counter bit11 and decrease as bit order increases, as expected. This is because the lower bins contain DFFs with the least number of fan-in data paths, hence it is expected that data pattern be a significant factor regarding SEU cross sections.

It is shown in Fig. 66 that as the counter bit order continues to increase, the decreasing trend of Counterσ_{bin} changes and Counterσ_{bin} begins to increase. For the test counter design, the shift in trend starts at the 4th bin (Counter bit12 thru Counter bit 23). In this region of circuitry, data-pattern is no longer the significant factor pertaining to Counterσ_{bin}. Instead, the bits in the higher bins have a significant increase in fan-in. The effective $P_{SET\rightarrow SEU}$ CCELLs increases and consequently begins to drive Counterσ_{bin}. Fig. 66 illustrates that for LET values from 75MeV*cm^2/mg to 28MeV*cm^2/mg, Counterσ_{bin} has a distinguishable and consistent pattern across bins.

Fig. 67 is an illustration of Counterσ_{bin} at lower LET values. As with WSRs, at lower LETs, it is expected that the smaller SETs will get filtered when propagating through multiple CCELLs and routes. However, as previously mentioned counters have DFFs with multiple fan-in data paths, consequently increasing the number of effective $P_{SET\rightarrow SEU}$ CCELLs. At lower LETs (2.8 MeV*cm^2/mg and 5.6 MeV*cm^2/mg), Fig. 67 emphasizes this point by demonstrating that Counterσ_{bin} becomes insignificant for the 1st bin. However, it is more likely that the bins containing fan-in have a significant Counterσ_{bin}.

The comparative SEE analysis between WSRs and counters has demonstrated that at low LETs, DFF fan-in can be a factor and consequently decrease LET_{th}.

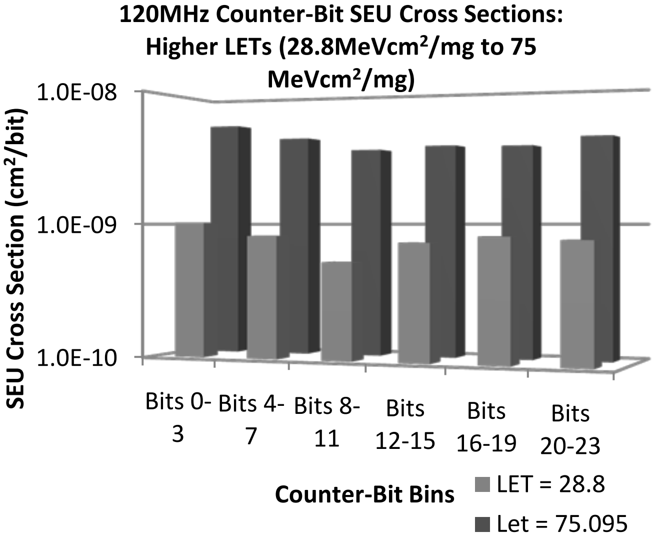


Fig. 66: SEU cross sections at high LET relative to counter bit ordering. Lower order bits have higher SEU cross sections as expected due to high frequency data patterns.

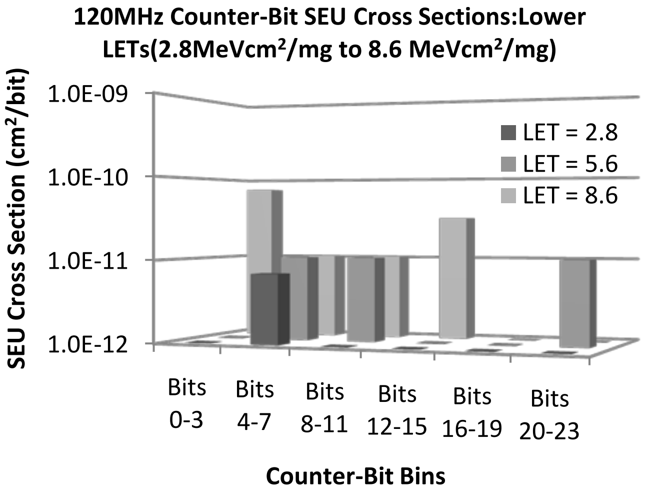


Fig. 67 SEU cross sections at low LET relative to counter bit ordering. Fan-in effects are apparent due to the upper order bit error distribution being more prevalent at LET values 2.8 to 5.6 MeV*cm^2/mg.

11. GLOBAL ROUTES

Global route upsets were observed at LET>50MeV*cm^2/mg. The probability was very low. More testing is suggested to hone in on actual rates.

12. DUT1 BLOCK MEMORY HEAVY ION TEST RESULTS

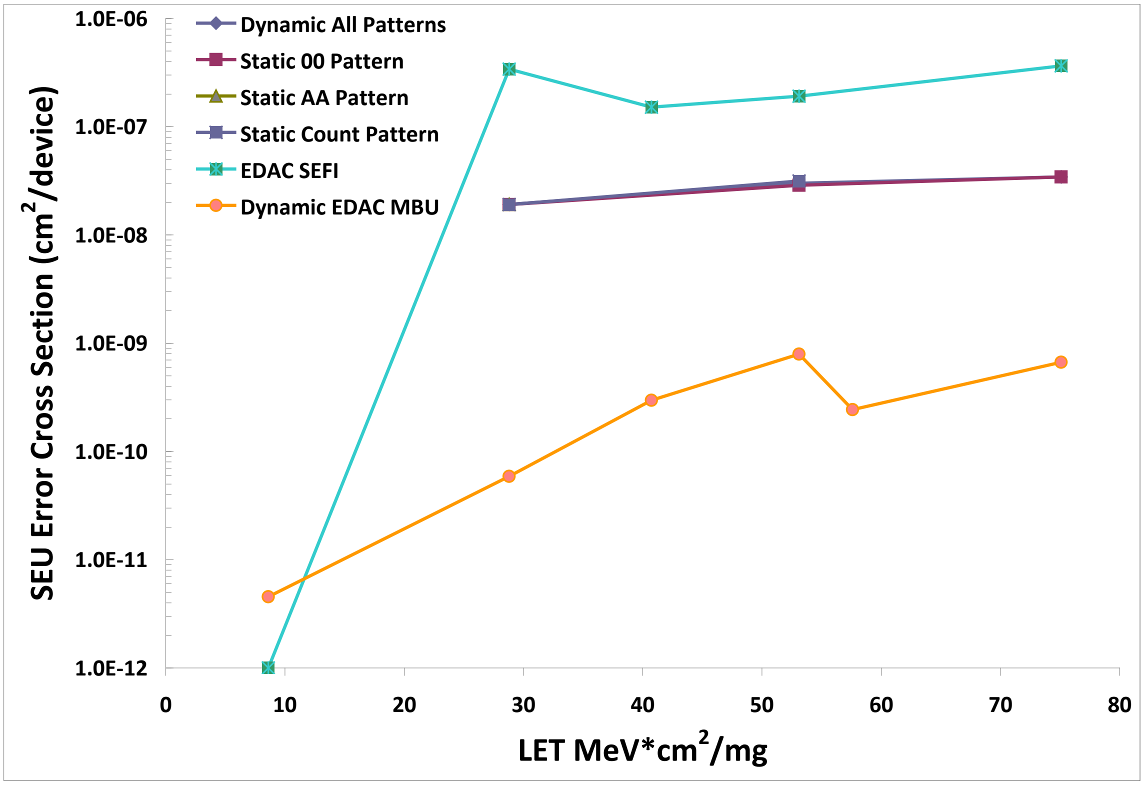


Figure 68: SEU Error Cross Sections. Static and dynamic memory reads were evaluated for several patterns for memory structures with and without EDAC controls

Figure 68 illustrates that the memory bit-cell SEU cross sections over all data patterns were statistically equivalent. Dynamic tests consisted of read-write-modify cycles. Regarding memory cell SEU-cross sections, there was no statistical difference between dynamic memory operations during DUT irradiation versus static memory operation. With dynamic testing, multiple bit failures (MBUs) per address did occur at 8.5MeVcm^2/mg. Because the EDAC structures are Single Error Correct Double Error Detect (SECDED), memory reads that utilized the embedded EDAC structures had failures. No testing was performed below 8.5MeV*cm^2/mg, therefore no on-set for memory MBU and EDAC failure has been determined. For LET > 20MeV*cm^2/mg, EDAC SEFI's also occurred. An EDAC SEFI pertains to the event of the EDAC circuitry becoming stuck in a state where it can not correct data and eventually corrupts good data. EDAC SEFIs at LETs>20MeV*cm2/mg were significant and can be avoided by the user creating a custom EDAC with the RTAX-s user fabric.

13. FIFO MEMORY HEAVY ION TEST RESULTS

The most significant observation from the current point of analysis is that the embedded FIFO memories have observed SEFIs at all LETs and tests. The following list SEFI types:

Address counter disturbed hence data is lost

FULL flag becomes stuck low and never activates as expected

Empty flag becomes stuck low and never activates as expected

Full and Empty SEFIs were not observed in the FIFOs that were implemented in the LTMR'd hardened fabric.

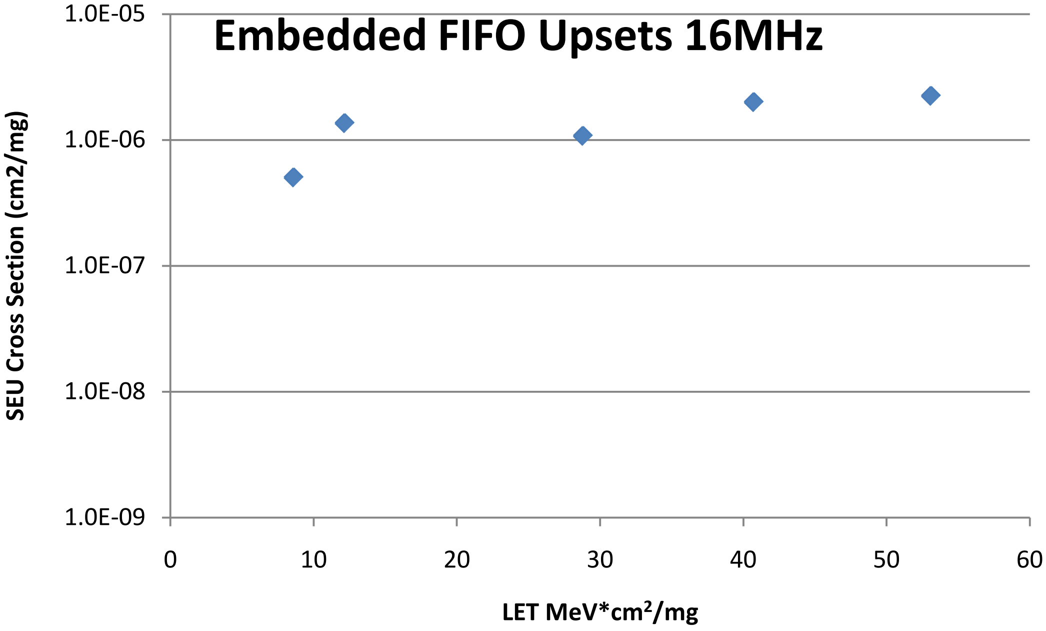


Figure 69: Embedded FIFO SEU Cross Sections. Upsets appear to be near saturation for LET>30MeV*cm^2/mg

14. CONCLUSIONS

14.1 WSR and Counters

The Actel datasheet lists a bit error rate <10^{-10}. This SEU rate is based off of shift register SEE data with LET thresholds > 37MeV*cm^2/mg. This does not correspond to the results of this study. This study has demonstrated that upset rates are frequency dependent and can be impacted by design complexity. The frequency dependency comes from the $P(fs)_{SET\rightarrow SEU}$ component of the unmitigated combinatorial logic. For each DFF, as the combinatorial logic paths increase, its susceptibility to $P(fs)_{SET\rightarrow SEU}$ will also increase.

Following the convention for error rate prediction, the user would use the bit error rate found in the manufacturer data sheet and multiply that number by the number of implemented DFFs within the design. For the case of the RTAX-s device, this may result in error rate predictions that are too low.

14.2 Block Memories and EDAC

SRAM memory cell bit error rates seem to match the manufacturer numbers within the data sheet. However, the EDAC SEFI rate is not mentioned in the data sheet. The embedded EDAC is unprotected (not hardened) and can incur a SEFI. The LET threshold for EDAC SEFI rates is greater than 30MeV*cm2/mg. This evaluation will be verified by further testing. EDAC error signatures included:

Corrupted memory words due to EDAC circuitry (data was unretreivable)

EDAC circuitry becoming inactive (increases the expected error rate)

Prior to EDAC SEFIs, the EDAC circuitry corrected data upsets at the rate predicted by the manufacturer. All EDAC SEFIs were correctable by a reset. EDAC SEFIs can be significantly reduced by the user implemented a custom EDAC within the LTMR'd hardened fabric.

14.3 *FIFO Embedded vs Custom*

Embedded FIFOs are not hardened and can SEFI just as the EDAC logic. However, it has been observed that the embedded FIFOs have a lower threshold of SEFI. The high potential to SEFI can be avoided by the user implementing custom FIFO control utilizing the LTMR'd hardened logic. The types of SEFIs that were observed were:

- Address corruption – hence loss of data
- Full flags becoming stuck
- Empty flags becoming stuck

All FIFO SEFIs were correctable by a reset.

15. APPENDIX 1:

[1] Actel Datasheet: “RTAX-S/SL RadTolerant FPGAs” http://www.actel.com/documents/RTAXS_DS.pdf, V5.2, October 2007.

[2] M. Berg “An Analysis of Single Event Upset Dependencies on High Frequency and Architectural Implementations within Actel RTAX-S Family Field Programmable Gate Arrays,” IEEE Trans. Nucl. Sci., vol. 53, n° 6, Dec. 2006.

[3] M. Berg “Trading Application Specific Integrated Circuit (ASIC) and Field Programmable Gate Array (FPGA) Considerations for System Insertion”, NSREC Short Course, Quebec City, CN, July 2009