Radiation Testing Update, SEU Mitigation, and Availability Analysis of the *Virtex* FPGA for Space Reconfigurable Computing†

Earl Fuller[2], Michael Caffrey[1], Anthony Salazar[1], Carl Carmichael[3], Joe Fabula[3]

[1] Los Alamos National Laboratory

[2] Novus Technologies, Inc.

[3] Xilinx, Inc.

Abstract

Orbital remote sensing instruments and systems can benefit from high performance, adaptable components. Field programmable SRAM-based gate arrays (FPGAs) are usually the chosen platform for real-time reconfigurable computing. This technology is driven by the commercial sector, so devices intended for the space environment must be adapted from commercial products. Total ionizing dose (TID), heavy ion and proton characterization have been performed on *Virtex* FPGAs fabricated on epitaxial silicon to evaluate the on-orbit radiation performance expected for this technology. The dominant risk is Single Event Upset (SEU), so upset detection and mitigation schemes have also been tested to demonstrate the improvement in the device upset sensitivity and the system consequence of upsets.

I. INTRODUCTION

Programmable logic has advantages over ASIC designs for the space community, including: reduced cost, faster and cheaper prototyping, and reduced lead-time before flight. Reprogrammable logic offers the additional benefit of allowing on-orbit design changes. This flexibility allows a mission to adapt systems to evolving requirements. For remote sensing applications, computing system payloads may be used for multiple sensors, multiple targets, and multiple modes (search/track). Such reuse reduces the weight, space, and power requirements. In addition, the payload can be updated as signal-processing techniques improve throughout the mission lifetime.

The economics of radiation tolerant electronics suggest that the most likely supply of high density, high performance reprogrammable logic will be adapted COTS (Commercial Off The Shelf devices). This paper discusses the radiation performance of the Xilinx *Virtex* FPGA including TID, Single Event Latch-up (SEL), and SEU. SEU characterization has been done for both static and dynamic modes of operation with results from both heavy ion and proton testing.

II. TECHNOLOGY CONSIDERATIONS

The Virtex FPGA is an SRAM based device that supports a wide range of configurable gates from 50k to 1M. The XQVR *Virtex* is fabricated on thin-epitaxial silicon wafers using the commercial mask set and the Xilinx 0.22μmm CMOS process with 5 metal layers. SEU risks dominate in the use of this technology for most applications. In particular, the reprogrammable nature of the device presents a new sensitivity due to the configuration bitstream. The function of the device is determined when the bitstream is downloaded to the device. Changing the bitstream changes the design's function. While this provides the benefits of adaptability, it also makes the device vulnerable to inadvertent SEU reconfiguration upset. A device configuration upset may result in a functional upset. User logic can also upset in the same fashion seen in fixed logic devices. These two upset domains are referred to as configuration upsets and user-logic upsets. Two features of the *Virtex* architecture can help overcome upset problems. The first is that the configuration bitstream can be read back from the part while in operation, allowing continuous monitoring for an upset in the configuration. Second, the part supports partial reconfiguration, which can speed upset recovery time.

III. RADIATION TESTING

The space radiation effects of most importance for this work are tolerance to total ionizing dose, and single event effects including latch-up and upset. The XQVR300 is the 300,000-gate device in the *Virtex* family that was used for testing and, because the technology scales just as SRAMS scale in complexity, is typical of all other parts in the family.

A. Total Ionizing Dose Tolerance

The first consideration for use of this technology in space is a survivability demonstration. To be survivable, the ionizing dose tolerance needs to be at a high enough level to be useful in many orbital scenarios. Total dose testing has demonstrated tolerance in the range of 80 to 100 krads(Si). Testing was done at both high and low dose rates using ^{60}Co sources. In-situ power supply current measurements were made throughout the course of the radiation exposure. In addition, at various cumulative dose steps, devices were temporarily removed for full functional and parametric testing

†This work, performed at Los Alamos National Laboratory, is supported by the US Department of Energy.

using the comprehensive final test program on the Xilinx test floor. The high dose rate test was done at 50 rads(Si)/sec to comply with Mil-STD-883, Method 1019 using static bias at nominal supply voltages. An accelerated anneal at 100 °C for 168 hours was used as a test for the rebound phenomenon. This anneal resulted in all degraded devices returning to pre-radiation performance without rebound indicating that trapped charge in oxides is the dominant degradation mechanism and surface state effects were not observed. To confirm that low dose rate effects were not overlooked by this test method, a low dose test at 0.0158 rads(Si)/sec was performed using a lower activity ^{60}Co source, this time with an in-situ power supply current as the principal degradation monitor. Figures 1 and 2 below show the power supply current monitor traces indicating the onset of TID degradation. Over this range of dose there were no significant changes noted in either AC (timing) or DC parametric characteristics, indicating relative stability of the surface MOS thresholds.

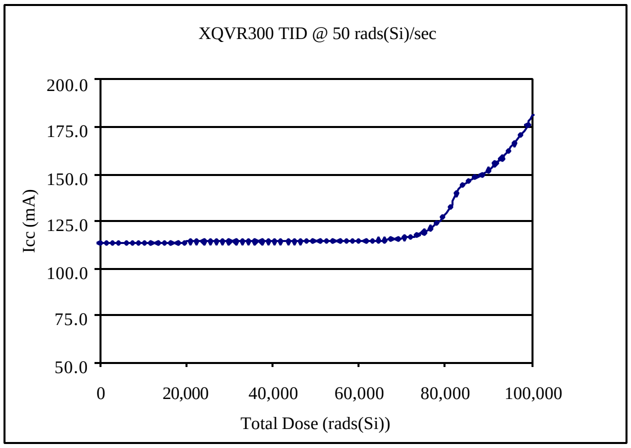


Figure 1: High dose rate performance. In-situ power supply current monitoring shows an increase in leakage above 80krads(Si).

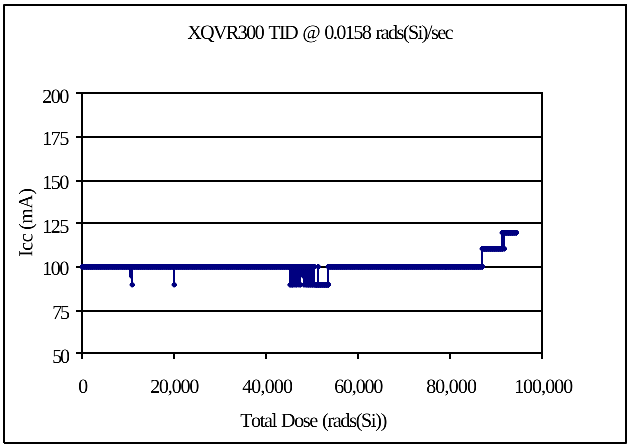


Figure 2: Low dose rate performance. In this case the increase in leakage current occurs above 90krads(Si) and does not constitute a failure below 100krads(Si). Note that the resolution of the current meter was set lower than for the measurements in figure 1. The negative-going spikes below 80,000 rads are an artifact of this setting.

These results show a somewhat higher dose degradation threshold at lower dose rates, as might be expected given the annealing response observed during the high dose rate anneal test. This performance is typical of many CMOS COTS technologies and would indicate that the on-orbit dose limit for this part is in the range of 100 krads(Si).

B. Heavy Ion Static SEU & SEL Characterization

Heavy ion characterization was conducted using the cyclotron facility at Texas A&M. Latch-up testing showed immunity to latch-up at an LET of 125MeV-cm^2/mg using gold ions with a fluence of 10^8 ions/cm^2 indicating no risk of latch up occurring on orbit. In particular, care was taken to assure that during testing the effective LET of the ions that reached the silicon surface met the number indicated. Technologies such as this, with 5 metal layers, can result in significant attenuation of heavy ion energies before reaching the sensitive region. The 2068 MeV Au beam, with a penetration range of 109µmm in silicon, and appropriate energy attenuation calculations were used to assure that the single event latch-up immunity was demonstrated. Other test parameters included maintaining ambient temperature of 23ºC and FPGA core voltage at 2.5 volts.

Testing of the FPGA was performed with the use of the AFX (Advanced FPGA Development System) system supplied by Xilinx along with test software developed by Los Alamos and the test method used is explained in detail in a previous publication [1].

Upset testing at the bit level was accomplished over a broad range of LETs with the resulting cross-section characteristic indicated in Figure 3.

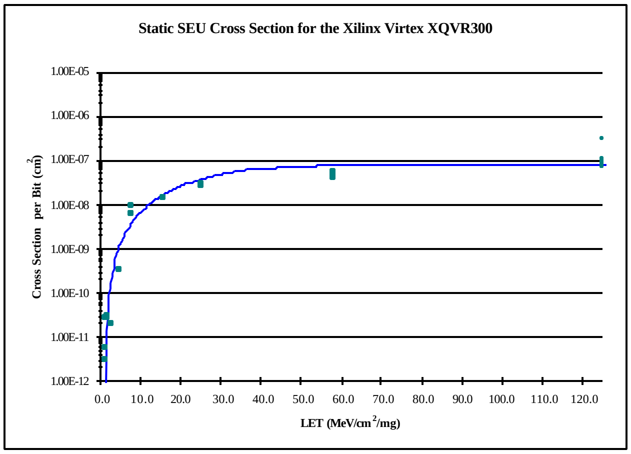


Figure 3: Static heavy ion bit upset cross-section vs. LET for the *Virtex* FPGA.

The capability to write to and read back the configuration bit stream allowed each routing bit, logic block flip-flop, memory cell, and other storage locations of the device to be individually monitored for static upset sensitivity. In this way the device could be tested as a static RAM-like part. For the case of the XQVR300 part, over 1.75M bits exist and were individually tested. Table 1 below summarizes the static bits that were accessible

Table 1: Latch types in the *Virtex* XQVR300 FPGA

Latch Type	Function	No. Bits
CLB	Configuration Logic Blocks	6,144
IOB	Programmable IO Blocks	948
LUT	Look Up Tables	98,304
BRAM	Block RAM	65,536
	Routing & Other Bits	1,579,860

Testing was allowed to proceed until 100 to 1000 bit upsets were observed in order to allow for statistical significance to the results. Accordingly there is a good Weibull fit and this formula can be used later for on-orbit upset analysis.

Weibull formula: $F(L) = s_{sat}(1- \exp\{-[(L-L_0)/W]^s\})$ [2]

where: $F(L)$ = SEU cross-section in μm^2/bit;

s_{sat} = limiting or plateau cross-section;

= $8\mu m^2$ for *Virtex*

L = effective LET in MeV-cm^2/mg;

L_0 = upset threshold LET;

= 1.2 MeV/cm^2/mg for *Virtex*

W = width parameter;

= 30 for *Virtex*

s = a dimensionless exponent.

= 2 for *Virtex*

Other authors have reported on the potential for contention to occur in the event that a configuration upset causes a bit driving high to be connected to a bit driving low.[6] This circumstance may occur and current monitoring during heavy ion testing noted small fluxuations in power supply current as upsets accumulated. The fluxuations in current were in both the positive and negative direction, indicating randomness in contention. No catastrophic upsets were observed. The *Virtex* design limits the drive current of internal bits to prevent contention from causing long-term reliability failures that might otherwise occur due to contention current.

In addition to these bit upsets, one unusual upset signature was recorded which represents an upset in the configuration control logic register. In this situation the number of bit upsets observed exceeded the total number of particles radiated on the die by as much as 10 times. This was not a multiple bit upset mode but rather an upset in the configuration control that results in the configuration memory being reinitialized. This is considered a Single Event Functional Interrupt (SEFI) type of upset and represents an apparent complete loss of configuration when it occurs. The observed LET threshold was between 8 an 16 MeV-cm^2/mg and only occurred if the fluence exceeded 10^5 ions/cm^2. Therefore the device cross-section for this upset mode is very low (<1 x10^{-5} cm^2) relative to the total cross-section for the part and there is a very small probability of occurrence on-orbit.

With this data one could multiply the number of bits times the cross-section and calculate a total cross-section for the part. Of course the device is not intended to operate statically, and dynamic cross-section needed to be measured in order to determine the significance of each type of bit upset and whether combinational logic would add to the sensitive volume. Dynamic cross-section measurements were difficult to make using heavy ions because the flux of particles was too high to allow for accurate measurements of the time to upset in a dynamic mode. Control of heavy ion fluences of as little as 10 to 100 particles/cm^2 was judged to be a too unreliable for accurate measurements. Accordingly, proton characterization was pursued. The substantially lower interaction rate from protons would allow the rate of observed device upsets to be lowered many orders of magnitude allowing accurate time-to-upset measurements.

C. Proton- Induced SEU Testing

Because of the low threshold LET, proton upsets are possible and a similar static bit characterization was performed using the proton beam at UC Davis. The bit cross-section is presented below in figure 4.

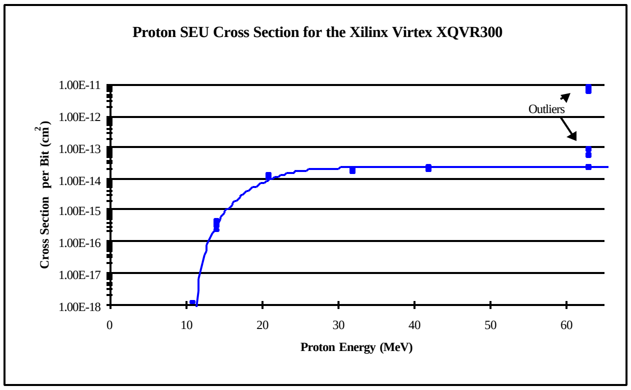


Figure 4: Static proton induced bit-upset cross-section vs. proton energy for the *Virtex* FPGA. Note the identification of outliers in the figure indicating that the configuration control circuit upset mode was observed at the highest energies tested (63MeV).

Testing was conducted using the same methods as the heavy ion case. Again, assuring that the test proceeded until a minimum of 100 bit upsets could be observed at each point provided statistical significance of the data. In this case we again can fit the data to the Weibull formula as follows:

Weibull formula: $F(x) = s_{sat}(1- \exp\{-[(x-x_0)/W]^s\})$ [3]

where:

$F(x)$ = proton-induced SEU cross-section in 10^{-12} cm^2/bit;

s_{sat} = limiting or plateau cross-section;

= 0.022 x 10^{-12} cm^2 for *Virtex*

x = proton energy in MeV;

x_0 = onset parameter;

= 10 MeV for *Virtex*

W = width parameter;

= 30 for *Virtex*

s = a dimensionless exponent.

= 2 for *Virtex*

This data is also used later for on-orbit upset rate analysis. It is also noted that from an analysis of the thin epi silicon that the thickness of the sensitive region is estimated to be 1µm.

D. Discussion of Upset Modes

Upsets in this FPGA can be grouped into three categories: configuration upsets, user logic upsets, and architectural upsets. The physics is the same for all, of course, but the observability and consequences vary.

Configuration upsets are those that occur in the configuration bitstream and can be detected by readback. The functional consequences will either be failure or no disruption of the function. The likelihood of failure depends on which bit is upset, and the specific design utilization of the device resources. Most of the static bits in the device are accessible via readback. In the case of the XQVR300 there are 1.465M bits in the readback bitstream, which represents 84% of the total. The cross-section per bit is indicated in figures 3 and 4 for heavy ions and protons. Accordingly the static bit cross-section for the part is equal to the product of the number of bits and the cross-section per bit. Of course cross-section will actually be less because not every bit upset will have a consequence in a given design.

The user logic contains elements not directly available in the bitstream for the purpose of upset detection. Actually most are in the bitstream but the contents are subject to change given the normal data manipulation functions that would be implemented. These include block RAM (BRAM), configuration logic block flip-flops (CLB-FF), and I/O block flip-flops (IOB-FF). Upset detection in these locations is not feasible because the state of each bit needs to be known a-priori, and data in these locations changes state in the normal function of the user implemented logic. In addition, any sensitivity contribution from combinational logic falls into this category. Upsets can only be mitigated while in operation with redundancy, such as triple modular redundancy (TMR), implemented by a user in the FPGA logic design. Observability is limited unless the user design can capture an event. In this case the total upset sensitivity of the user logic will be the sum of the bits included in the design and some number of bits from the combinational circuits unique to the personalization of the FPGA, all moderated by the amount of and effectiveness of redundancy. Accordingly, several designs need to be tested to develop useful metrics for these issues and this will be discussed below in section IV.C.

Architectural upsets are those upsets in the control elements of the FPGA (e.g. configuration circuit, JTAG TAP controller, reset control, etc). Measurement of SEUs in these circuit elements is often only indirectly measurable in that one needs to observe and identify an upset "signature" and associate it with a control element function. As an example, it is possible for a single bit upset in the configuration control circuit to change many of the configuration bits all at once. This upset signature is observable and it's frequency of occurrence in a heavy ion or proton test can be measured. And a small, but non-zero, cross-section can be determined in order to estimate the frequency of this type of event occurring in a given application.

There are several objectives behind understanding the upset rate and the contribution of these different categories. First, one wants to understand all the possible mechanisms for the introduction of errors in the performance of a user's function. Second, to understand the severity of the upset problem, one needs an understanding of both the upset frequency and the consequence of the upsets that occur on the system function. These determine the cost of implementing mitigation measures and where they are most effectively directed. As an example, this work is for a remote sensing application which uses the FPGA to analyze sensor data. Upsets may be more tolerable in this application than for a spacecraft control function. Implementing redundancy may come at the cost of consuming 3 times the available FPGA resources available in each device.

IV. MITIGATION OF SINGLE EVENT UPSETS

Two techniques are used to mitigate the consequences of upsets.

A. Triple Module Redundancy (TMR)

First, triple module redundancy (TMR) is used in the logic design to mitigate an upset as it occurs in the device configuration or the user's logic. Should the upset occur in the users' logic, TMR votes out the error. If the upset occurs in the device configuration, TMR eliminates the output of the discrepant logic path.

B. Bitstream Repair Techniques

Bitstream reconfiguration, complete or partial, may occur without an interruption of service in the Virtex device. In addition, the device configuration can also be read back at any time without an interruption in service. These features allow two simple techniques for maintaining coherency of the bitstream. Scrubbing simply rewrites the device bitstream, so the time to repair an error is the scrub cycle time. Cycle time can be on the order of a few milliseconds and varies with device density. Continuous readback in conjunction with a detection algorithm (bit compare, CRC, etc) provides data on upsets encountered, time, and frequency. Partial reconfiguration (PRC) repairs any section of the device where an error is detected. The repair time is comparable to scrubbing. For a full discussion of the bitstream repair techniques and how they are implemented in the Xilinx product, the reader is referred to Xilinx application literature [5].

C. Dynamic-mode SEU Testing

Dynamic testing provides an important measurement: the fluence to failure of a function. This is useful because static testing does not detect the contribution to sensitivity from combinational circuit operating at system speeds. Static testing also cannot determine the consequence of an upset; frequently an upset does not result in a functional failure. The problem with dynamic testing is that it provides poor fault isolation. It is difficult to identify the category of upset as the source of the failure. The other consideration for dynamic testing is that the test probes an FPGA design, not the device. A design will utilize some subset of the device resources and will have a unique cross-section.

1) Testing Procedure

Several different designs were used for dynamic testing. As indicated above, the AFX board was used as a test platform with an interface to a PC. Test software would then load the design configuration onto the part and dynamic operation could be initiated. The general test strategy used an on-chip compare circuit to detect upsets by a difference between two parallel processing paths or between actual and expected data in the function being implemented. Control software allowed for the device function to be halted on failure and then the bitstream could be read to determine how many bit upsets may have occurred. The proton beam fluence would accumulate until failure was detected and the resulting fluence-to-failure would indicate the cross-section of the dynamic function. Trials were repeated many times to provide statistics on the measured cross-section.

Each test design was implemented both with and without mitigation strategies to be able to measure the potential benefit. A basic design would be developed that would use about one-third of the device. A second version would be implemented in a TMR mode. Finally, each design would allow for the bitstream to be read and corrected (PRC), or continuously scrubbed to prevent individual bit upsets from accumulating. In this way each of the potential mitigation strategies could be tested individually or in combination.

The figures below show the designs used.

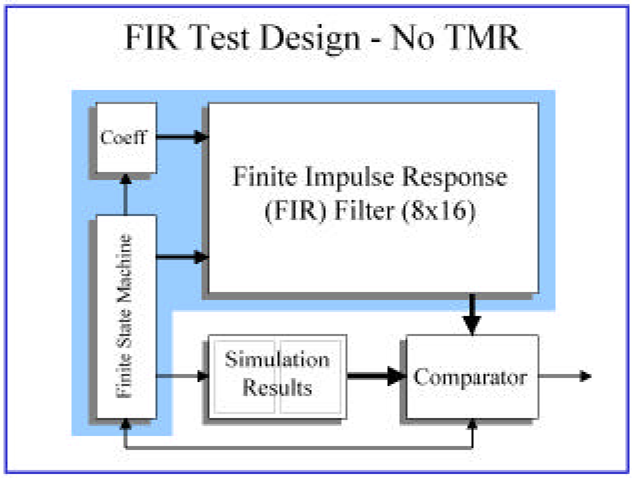


Figure 5: The FIR filter design, without triple modular redundancy (TMR), implemented in the XQVR300 for dynamic proton induced SEU testing is approximately 1/3 utilization.

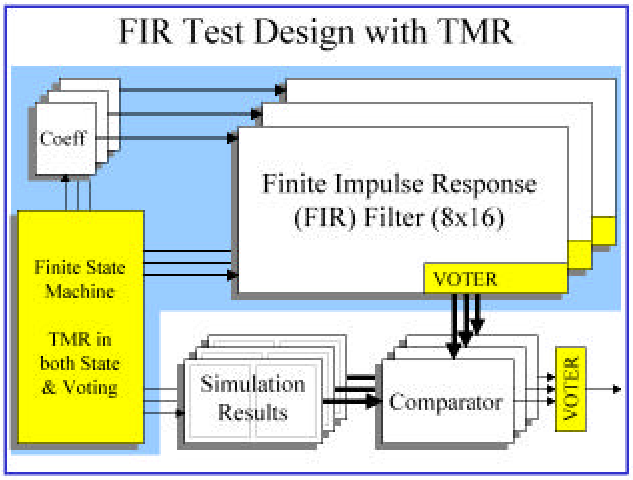


Figure 6: The same FIR filter with TMR implemented in all blocks of the function. This design uses 3 times the resources of the design above.

The first design performs a Finite Impulse Response (FIR) filter type function. One section of block RAM stored data and coefficients, and another section stored expected results. A comparator circuit detected failure when the filter output disagreed with the expected value. Both with and without redundancy, the design self checked for errors with a redundant comparator. TMR was implemented in all areas of the function; filter, block RAM, and comparator. The filter outputs flow into a redundant comparison circuit, all of which reside in the DUT. Should an error occur in one of the redundant digital filter legs, the comparator latches an error condition. Should an error occur in the redundant comparator then an error flag is also raised. A self-testing design was amenable to the test fixture and simple to test.

The goal of the second design was to develop a configuration for the device under test that utilized a large proportion of the resources in order to give the results statistical significance. The principal resources being considered are the Configuration Logic Block (CLB) flip flops, Block RAM (BRAM), Look up Tables (LUTs) and Delay Lock Loops (DLLs). The design is actually a combination of two, including one exclusively for LUTs and flip-flops and one focusing on the BRAM and is referred to in the results as the "Combo" design.

The BRAM portion (Figure 7) treated each 4 Kbit block as a 511x8 FIFO that is filled with a random number generator, implemented using a Linear Feedback Shift Register (LFSR). Once full, the output of the FIFO (First-in First-out memory) is continuously compared to an identical random number generator, the comparison providing an indication of upset. No differentiation is made between an upset that occurs in the random number generator, the FIFO, or the comparison circuit. The outputs of 16 test FIFOs were logically OR'ed together and monitored by software. For each FIFO, the input generator operated from a different DLL than the generator that sourced the comparison to determine any increase sensitivity from the clock management circuit. The BRAM test configuration utilized 100 percent of the available BRAM and 24 percent of the available logic slices.

The CLB portion (Figure 8) partitions the available CLBs into two large shift registers where the shift register used both LUTs and CLB flip-flops. Each shift register was clocked with the clock from a separate DLL. Each shift register was fed by the same oscillating flip-flop. Outputs from the two shift registers were compared to detect upset, with an output monitored by software. This design utilized 95 percent of the available slices.

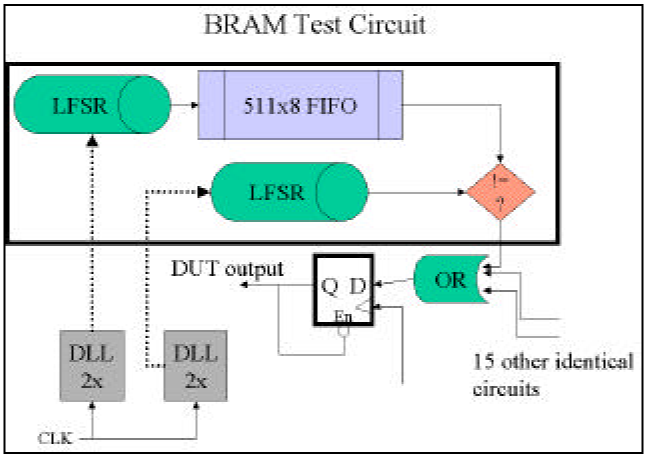


Figure 7: Block diagram of the block RAM portion of the Combo test design.

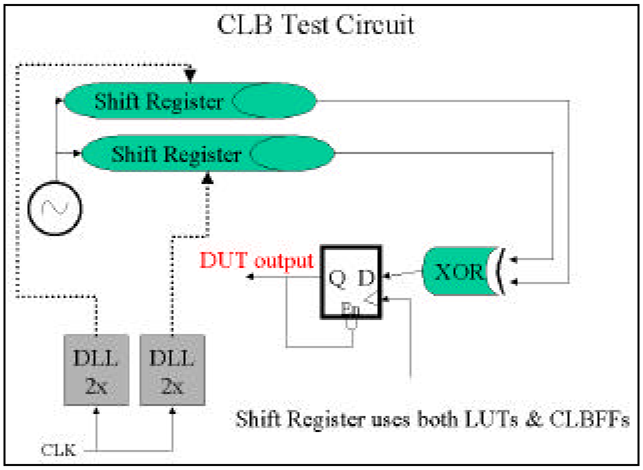


Figure 8: Block diagram of the block configuration logic block (CLB) portion of the Combo test design.

The fluence to upset was measured for the baseline, no TMR design. Improvement due to TMR was measured with fluence to failure of the TMR design. Several considerations were investigated including the potential for variation in sensitivity due to operating frequency and complexity of the design. A detailed investigation was made of the significance of bit upsets in the dynamic designs. Since the *Virtex* architecture includes a significant overhead of routing bits in order to accommodate a wide range of designs, not every bit upset will have the same consequence in potentially upsetting the device.

2) Results & Interpretation of Dynamic-mode SEU Testing

Three variations of these designs were tested without bitstream upset mitigation; FIR with no TMR, FIR with TMR, and the Combo design. Subsequently, the two FIR designs were tested with bitstream mitigation to evaluate improvement. In tests with no mitigation, two different error signatures were observed. All errors were functional failures. The first category, soft errors, recovered operation with reset. The second category, hard errors, required complete reconfiguration of the device to recover operation. Soft failures cannot be attributed to configuration upset errors, or recovery with reset would not be possible. The number of configuration bitstream upsets to cause hard failures is shown in figure 9. The significance of this data is that on average 6.5 (+- 6) configuration bitstream upsets are required to upset a design with no mitigation of any kind. Soft errors, those not due to configuration errors, accounted for 45% of the total errors in all tests with no mitigation (TMR or bitstream).

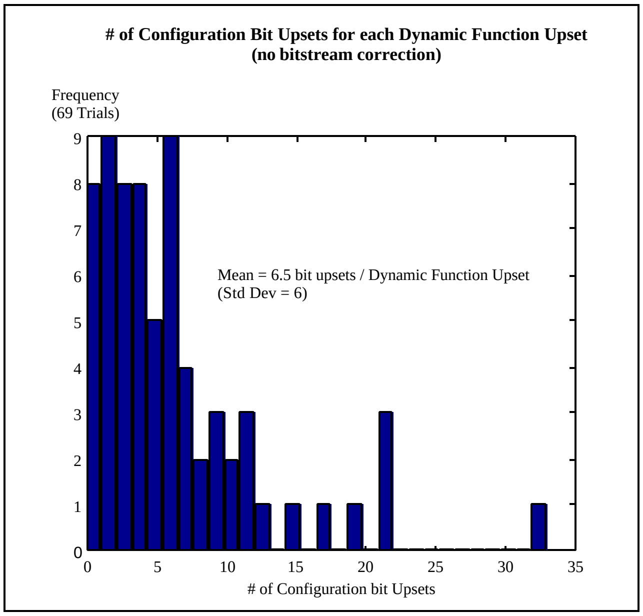


Figure 9: This histogram shows the number of bit upsets detected for each dynamic function failure in proton testing. Often, several configuration bits upset before a hard functional upset occurs.

To show the degree of benefit demonstrated by TMR and / or bitstream mitigation, data from five different tests is shown in figure 10. Each test is many trials measuring fluence to failure for a design (hard or soft). The two FIR designs with and without TMR were both tested with and without bitstream upset mitigation (PRC). The equivalent total static fluence to failure of the device is also shown on the graph for reference. This number is derived by multiplying the proton saturation cross-section for each bit, times the total number of bits in the bitstream, and then plotting the reciprocal.

Several observations can be made. First, no significant difference exists between the Combo and FIR NoTMR test designs. Given that the Combo design utilizes much more of the available resources of the XQVR300 it could have been expected that its sensitivity would have been greater. With only two designs tested, more work is required to determine how much the device sensitivity varies from one design to another. Second, the redundant FIR with TMR design performs no better with bitstream mitigation. No advantage is demonstrated by the use of scrubbing techniques alone. The FIR with TMR design showed improvement with redundancy alone and even more improvement (15x) with bitstream mitigation incorporated. It is clear that there is still a significant cross-section even with both mitigation techniques employed, suggesting another (perhaps architectural) avenue for error introduction. The dynamic cross-section is less than the static cross section as would be expected from the discussion above, i.e. not every configuration upset contributes to a failure.

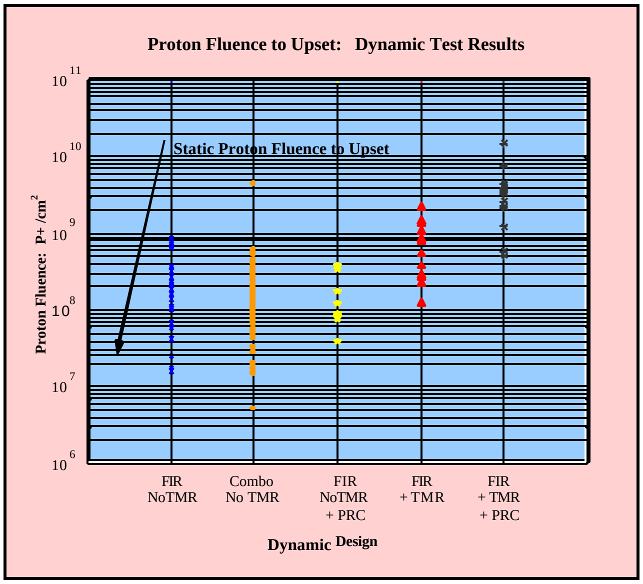


Figure 10: Scatter plot of the fluence to failure of each of the dynamic designs tested. Note that the FIR design that uses both TMR and bitstream mitigation (PRC) shows the best result at roughly 15x improvement over the basic FIR design. Also plotted for reference is the equivalent total static bit fluence to failure, which is derived by the product of bit cross-section and total bits. Clearly not every bit upset will result in a functional failure due to the architectural variables in the device.

Figure 11 shows the three test designs tested at various clock frequencies to determine any measurable transient contribution to the sensitive cross-section. Clearly, no significant effect is present.

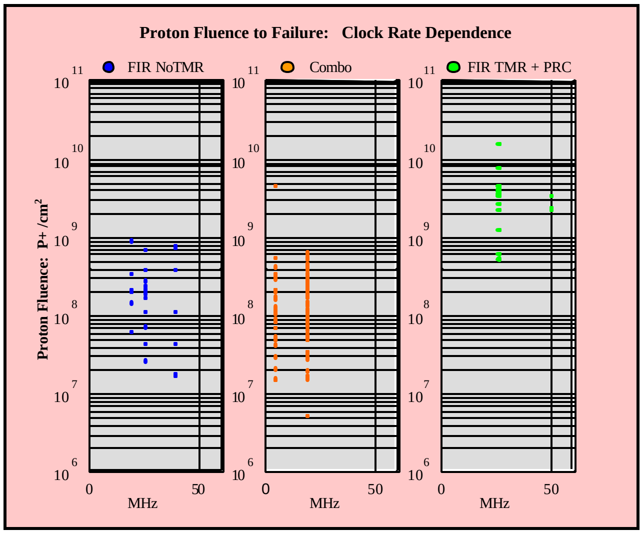


Figure 11: Scatter plot of the results of fluence to failure trials of different designs and different operating frequencies. Over the range tested, no frequency variation is evident.

The improvement in dynamic cross-section using TMR and bitstream repair is evident, but less than expected. Further analysis was therefore done to investigate other circuit elements that may contribute to this sensitivity. Inevitably, it becomes necessary to understand how the design software implements a design in a circuit to fully understand how an FPGA circuit works and therefore its potential weak points. As one can imagine, a given circuit design can be implemented in an FPGA using many different combinations of circuit resources. One of the functions of automated design software is to implement a design in its simplest form. To do so it is able to prioritize more complex resources, such as Look Up Tables, for use in more complex functions such as truth tables or memory, and use simpler resources for simple functions like driving a constant state or implementing an inverter. The analysis of the way the design software uses the simplest circuit resources first, lead to a better understanding of all of the resources available in the Virtex architecture.

Inherent in an SRAM FPGA is circuitry that initializes the array to a known good state and circuitry that holds logic resources in this desired state even if they will not be used by a design. One circuit element was identified as a potential weak point for investigation. It is referred to as a "weak-keeper" circuit whose function is to initialize and hold a logic element until and unless a routing bit connects it into a circuit. The transistor drive of the routing bit is more than sufficient to suppress the weak-keeper to prevent contention. And, since the routing bit is part of the configuration bitstream, its state can be monitored and corrected if it should become an SEU. It was determined, however, that the weak-keeper can also be upset. Again, if a routing bit is connected, it has sufficient drive to suppress the weak-keeper even if the weak-keeper changes state. But if the design software uses these weak-keeper circuits as independent low-level resources to set a state in a design, it can upset and not be evident in the bitstream. More importantly, it is not correctable except during a device reset.

A test design was therefore devised to evaluate this issue. The solution was to force the design to program resources with a controllable bit making it both detectable and correctable. Specifically, all Vcc and Ground nodes in a design were assured to be connected to a high or low level via a routing bit. In this way, the weak-keeper circuits could be suppressed and better SEU detection was provided in the bit stream. The results in proton beam testing were a significant improvement in dynamic cross-section and no occurrence of upsets that required reset. Figure 12 shows the dynamic cross-section measurements that were made.

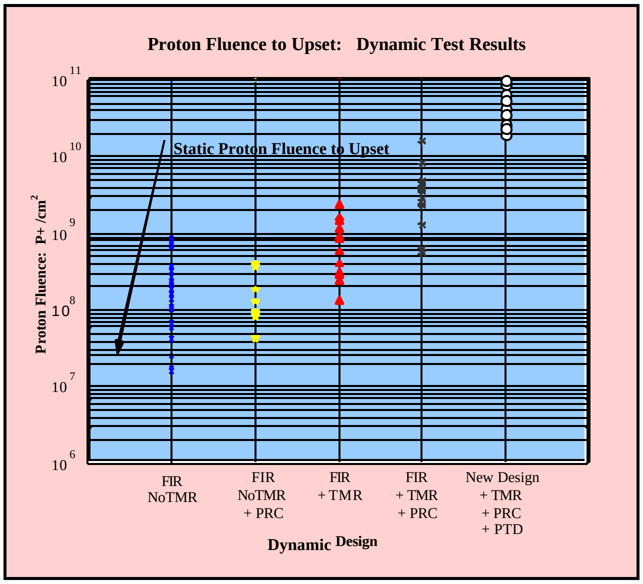


Figure 12: A revised scatter plot of the fluence to failure of each of the dynamic designs tested. Note that the new design that incorporates tying down the Vcc and Ground nodes (PTD for Power Tie Down) shows the best result of all test designs. The mean value of the dynamic fluence-to-failure measurements is 1100 times better than the static bit result previously explained in figure 10.

The results show that combining multiple design techniques, a significant mitigation of the worst case SEU rate can be achieved. The static cross-section is assumed to be the worst-case cross-section. It is calculated by multiplying the total number of static bits by the cross-section per bit. As previously suggested, it overstates the cross-section of the device because it is not possible for every single bit to result in an upset in a particular FPGA design. A lower cross-section has been measured by the dynamic mode testing procedure used. Combining all of the design techniques used together can mitigate the impact of SEUs that can occur with up to 1100 times reduction in the device upset sensitivity. This work is ongoing and more improvement is expected from recent analysis of the device architecture. Work in progress includes an SEU simulator, which can be used to test every configuration bit in the device for single bit design failures. Using partial reconfiguration, each bit can be sequentially flipped in the device and the consequences observed. This approach has several advantages including cost and schedule over accelerator test facilities. Also, every configuration bit can be tested, providing a substantial step towards reliability assurance. As design techniques mature that resist an assault on the bitstream, a return to the accelerator will determine the presence of other upset mechanisms that may be undetected in the presence of the dominating effects of user and bitstream upsets.

V. On-orbit SEU Rate Estimates

Every orbital scenario is different, however, it is useful to calculate an upset rate for sample orbits. As indicated earlier, remote sensing instruments are logical platforms for use of this technology. For the sake of example, several circular orbits were modeled assuming a 60-degree inclination angle and operation during solar maximum. Using these assumptions and the CREME96 model [4], the upset rate for the device can be estimated. With the knowledge of both the proton and heavy ion response, this modeling tool provides both an orbital average upset rate as well as the higher rate that would occur during periodic solar flare events. The graphs below indicate some of the results of the modeling effort.

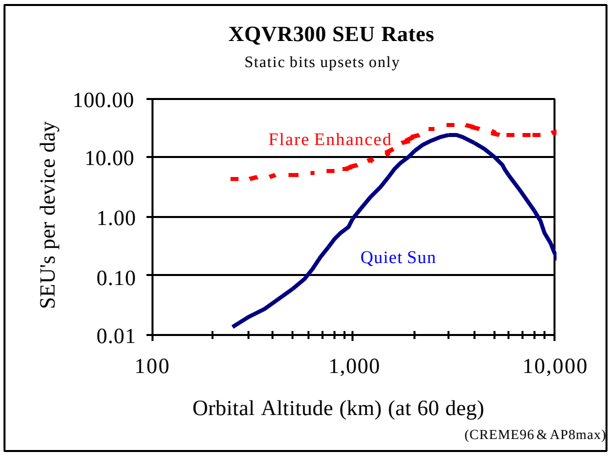
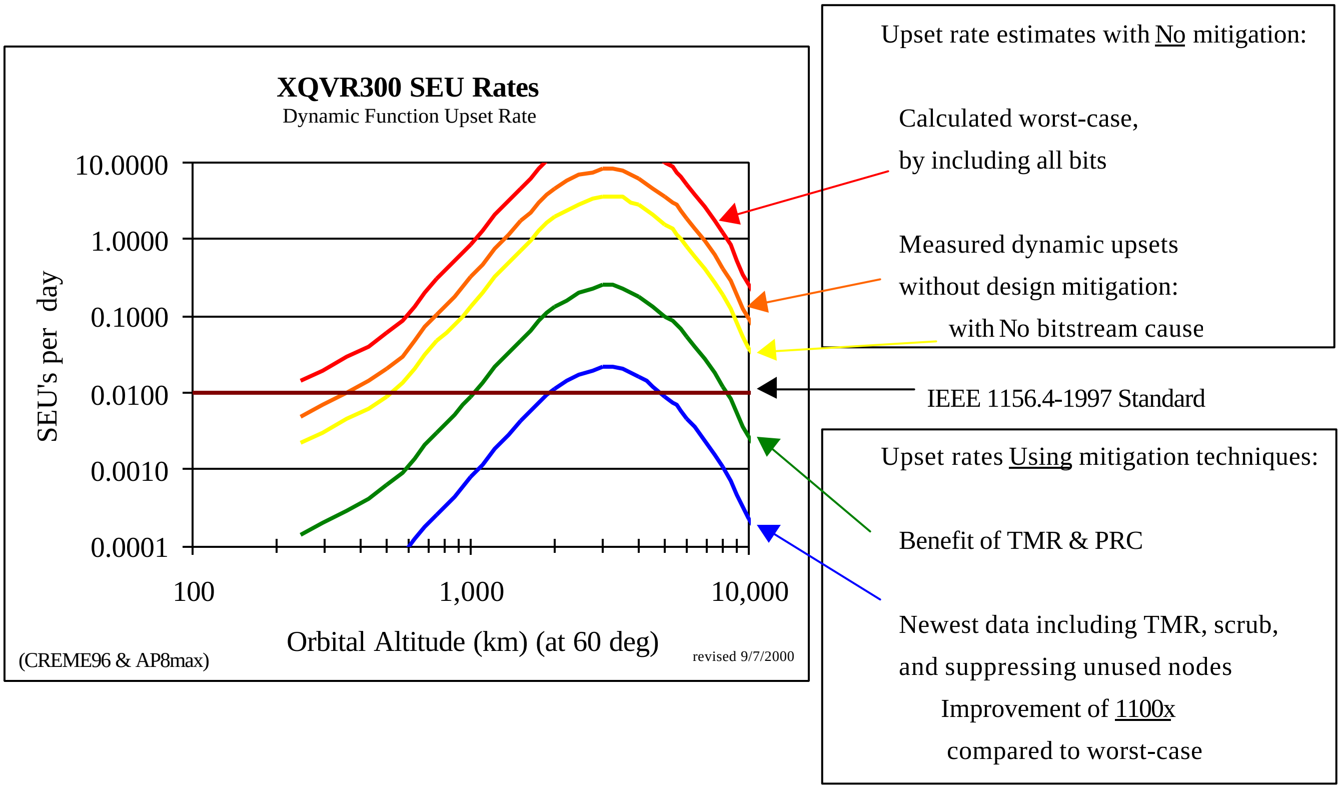


Figure 13: This plot is obtained by calculating the hypothetical sensitive volume of the XQVR300 as the product of all of the bits in the FPGA and the average cross-section from figures 3 and 4 earlier in this text. As demonstrated by the dynamic test data, the expected upset rate should be lower depending on the degree of mitigation employed.



Upset rate estimates with No mitigation:

Calculated worst-case, by including all bits

Measured dynamic upsets without design mitigation: with No bitstream cause

IEEE 1156.4-1997 Standard

Upset rates Using mitigation techniques:

Benefit of TMR & PRC

Newest data including TMR, scrub, and suppressing unused nodes Improvement of 1100x compared to worst-case

Figure 14: Applying the observed dynamic cross-sections to the upset rate calculations of the device, the maximum benefit observed so far in this work is plotted as the combined benefit of redundancy (TMR) and bitstream scrubbing along with suppression of unused nodes.

VI. Availability Considerations

Whether or not a device upsets at a given rate is of lesser concern to the system engineer than is the consequence of the upset. A device upset does not automatically have a system upset consequence. It has already shown that within the FPGA device architecture there are more routing options than are needed for a single design, so clearly not every connection upset is critical. Recovery from an upset is also a very significant system level consideration. The concept of availability is a common one for system reliability engineers. A system is defined to be "available" when it is not down or off-line. For this discussion, the Mean Up Time (MUT) is defined as the average time to upset. Then the Mean Down Time (MDT) is defined as the average time to recover from the upset. System availability is calculated as the percentage of time that the system is "up" and available to perform its intended function.

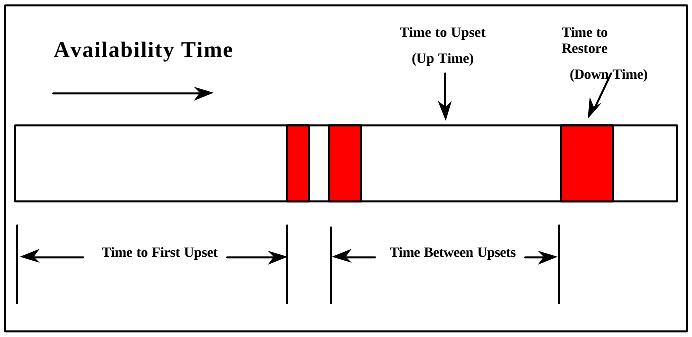


Figure 15: Availability time line.

Mathematically, Availability, A, is expressed as:

$$A = MUT / (MUT + MDT)$$

This formula indicates a couple of simple, but important, system reliability considerations. First, availability is increased if "up time" is increased by virtue of lower upset rates. And availability is also increased if "down time" is decreased by more rapid recovery from an upset. In particular, the more frequent the upset rate, the more important it is to recover rapidly to maintain system availability. Not to be over simplistic, but as an example, if an upset occurs every 100 days, a recovery time of 24 hours is sufficient to maintain 99% availability, while if the upset rate is once per day, 99% availability requires a recovery time of 36 seconds.

For the *Virtex* FPGA, upsets in the configuration are detected via readback of the configuration bitstream. Recovery can be accomplished either by partial reconfiguration or complete scrubbing of the bitstream. The SelectMAP interface in the *Virtex* architecture allows non-interfering readback and configuration. Recovery time is dependent on the number of bytes being configured and the clock frequency. The maximum frequency of the configuration clock is 66 MHz, but as a practical matter configuration is often limited by the speed of the device from which the bitstream is loaded. The XCV1800 PROM, for example, is limited to 25 MHz. The *Virtex* XQVR300 has 207,900 bytes in the bitstream. Readback and configuration at 25 MHz can therefore be done in 16.63 milliseconds. Since there will be latency in a system recover from an upset, the actual recovery time will vary depending on how the system is implemented. For an example, one can use 10x the part recovery time, or roughly 200 msec, as the system recovery time

From figure 13 and 14 above, we can use a worst-case error rate for the XCV300 in calculating the worst-case availability for this device type. The examples in table 2 are for a 1000 km circular orbit with a 60-degree inclination angle.

Table 2: Example Availability Calculations for XCV300 FPGA

	Example 1 (Quiet Sun)	Example 2 (Solar Flare)
Worst Case Upset Rate	SEU Rate = 0.88 / day	SEU Rate = 7.2 / day
Mean Up Time	MUT = 27.3 hours	MUT = 3.3 hours
Mean Down Time	MDT = 200 msec	MDT = 200 msec
Availability	A = 99.9998%	A = 99.998%

In a year of operation, the number of upsets that can occur will be less than the worst-case number of 240 (365 days * 0.88 SEUs / day). And because the recovery time is rapid, the total down time is 48 seconds due to SEUs.

The point is that a system design should include a consideration for minimizing upset recovery time as well as minimizing the rate at which upsets occur. During the process of system engineering, requirements are accommodated, and the consequence of an upset and the down time are comprehended. There are many applications where upsets are tolerable provided the downtime is small. One example is a remote sensing instrument, which collects and analyzes signals. The performance of these systems can be limited by the processing capability of the rad-hard technologies used. Using more modern COTs technologies, processing capability can be significantly enhanced. Data analysis can be done on-orbit rather than for data needing to be collected and downlinked. In these situations, system availability is limited by the relatively slow speed of the downlink. A substantial improvement in processing capability is achieved in exchanged for an occasional loss of data. And with rapid recovery a substantial improvement in on-orbit availability is easily achieved. The unique architecture of the *Virtex* FPGA enables this aspect of system engineering to be included in satellite applications.

One important consideration is the requirement for upsets to be detected to prevent fault propagation. Since upsets can occur in combinational logic, not every upset is detectable via configuration bitstream readback. Accordingly some degree of redundancy is required for maximum reliability.

VII. SUMMARY & CONCLUSIONS

The results of this radiation characterization program show that the *Virtex* FPGA meets TID and SEL requirements for many orbital applications. The static-cross section has been measured for both heavy ions and protons by testing the device as though it were an SRAM. LET threshold determined in the heavy ion test proved the part is sensitive to the proton portion of the spectrum. Dynamic testing required a proton accelerator so that the time between upsets could be increased, thereby allowing for accurate measurements

The upset risk dominates the radiation considerations for this part. The complexity of this device presents new upset modes and makes radiation testing difficult. Design approaches for upset mitigation provide significant improvement, though more work is necessary to determine the source of the remaining sensitive cross-section.

Two dynamic upset signatures have been found, soft errors where a reset is sufficient for recovery, and hard errors that require device reconfiguration. In tests without mitigation, 45% of the failures cannot be attributed to configuration bitstream upsets. It is also shown that, on average, 6.5 bitstream upsets are required for a functional failure for the test designs without mitigation. No measurable dependence on clock rate was found.

The utility of the device for orbital remote sensing data processing will depend on the mission requirements. Device processing performance and survivability are exciting, but more work is needed to find the source of the dynamic cross-section remaining after mitigation

ACKNOWLEDGMENTS

The authors wish to thank Rick Padovani of Xilinx and Mark Dunham and Steve Wallin of Los Alamos for their support of this work.

REFERENCES

[1] E. Fuller, M. Caffrey, P. Blain, C. Carmichael, N. Khalsa, A. Salazar, "Radiation Test Results of the *Virtex* FPGA and ZBT SRAM for Space Based Reconfigurable Computing," *MAPLD 1999 Proceedings*, *C_2*, September 1999.

[2] E.L. Petersen, J.C. Pickel, J.H. Adams, Jr., and E.C. Smith, "Rate Prediction for Single Event Effects -- a Critique," *IEEE Transactions on Nuclear Science NS-39*, No. 6, pp. 1577-1599, December 1992.

[3] A.J. Tylka, W.F. Dietrich, P.R. Boberg, E.C. Smith, and J.H. Adams, Jr., "Single Event Upsets Caused by Solar Energetic Heavy Ions" *IEEE Transactions on Nuclear Science*, December 1996.

[4] A.J. Tylka, J.H. Adams Jr., P.R. Boberg, B. Brownstein, W.F. Dietrich, E.O. Flueckiger, E.L. Peterson, M.A. Shea, D.F. Smart, and E.C. Smith, "CREME96: A Revision of the Cosmic Ray Effects on Micro-Electronics Code" *IEEE Transactions on Nuclear Science*, December 1997.

[5] C. Carmichael, "Correcting Single-Event Upsets through Virtex Partial Reconfiguration", Xilinx Application Note XAPP216, June, 2000.

[6] J. Wang, R. Katz, J. Sun, B. Cronquist, J. McCollum, T. Speers, W. Plants, "SRAM based Re-programmable FPGA for Space Applications" *IEEE Transactions on Nuclear Science*, Vol 46, No. 6, pp. 1728 – 1735, December 1999.

.