

RELIABILITY ANALYSIS OF RADIATION INDUCED FAULT MITIGATION  
STRATEGIES IN FIELD PROGRAMMABLE GATE ARRAYS

by

Justin Allan Hogan

A dissertation submitted in partial fulfillment  
of the requirements for the degree

of

Doctor of Philosophy

in

Engineering

MONTANA STATE UNIVERSITY  
Bozeman, Montana

April, 2014

© COPYRIGHT

by

Justin Allan Hogan

2014

All Rights Reserved

## ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Brock LaMeres, for his expectations, encouragement, advice, vision, support and understanding throughout the course of this endeavor. Thank you to my committee for being present through the entire process for anything I needed at any time. Thank you to my friends, and fellow graduate students, for continually supporting me on the path to graduation and never failing to provide company along the way. Thank you to the folks at the Bozeman Bike Kitchen for providing a grounding balance during the stressful times. Thanks, Mom and Pop, for supporting me in every way, shape and form throughout my entire academic career. Brian, Audrey, Autumn, David, Kourtney, thanks for being unwavering in your greatness. Mark, Kyle, thanks for always pushing me beyond my limits and always motivating me to find new ways of making things way more adventurous than is ever warranted.

## TABLE OF CONTENTS

1. BACKGROUND AND MOTIVATION.....	1
The Need for Space Computing .....	1
Trends in Space Computing.....	2
The Space Radiation Environment .....	4
Radiation Effects on Modern Electronics .....	8
Radiation Effects Mechanisms.....	10
Total Ionizing Dose .....	13
Single Event Effects .....	14
2. PREVIOUS WORK.....	18
Radiation Effects Mitigation .....	18
TID Mitigation .....	19
SEE Mitigation .....	20
Memory Scrubbing.....	21
Drawbacks of Current Mitigation Techniques.....	21
Radiation Effects Mitigation in Commercial FPGAs.....	23
MSU’s Approach to Reliable, High-Performance Space Computing.....	25
Reconfigurable Computing for Aerospace Applications.....	27
Radiation Sensor .....	32
New Research Hardware .....	35
1U CubeSat Stack .....	36
FPGA Board.....	37
Device Configuration .....	39
MicroSD Card Interface .....	42
System Testing.....	44
Near-Infrared Pulsed Laser.....	45
Cyclotron Testing of the Radiation Sensor.....	50
High-Altitude Scientific Balloon.....	53
Payload Thermal Design .....	56
Payload Control and Operation .....	59
Data Visualization .....	61
Payload Mechanical Design .....	62
HASP 2012 Payload Failure Analysis .....	64
HASP 2013 Flight .....	67
Drawbacks of This Approach .....	68
Current Research.....	69

## TABLE OF CONTENTS – CONTINUED

3. DESIGN OF EXPERIMENTS.....	73
Modeling Reliability.....	73
Modeling Reliability of Various Architectures.....	76
Simplex.....	76
Triple Modular Redundancy.....	79
Spare Circuitry.....	82
Repair Capability.....	83
Spare Circuitry with Repair.....	84
Method to Estimate Fault Rate.....	86
Environment Factors Affecting Fault Rate.....	86
Device Attributes Affecting Fault Rate.....	87
4. RESULTS.....	90
Analysis of System Architecture Effects on Reliability.....	90
Analysis of Single Point of Failure Area on Reliability.....	93
Analysis of Technology Partitioning on Reliability.....	96
Summary.....	97
REFERENCES CITED.....	99

## LIST OF TABLES

Table		Page
1.1	Space Radiation Environment [1–3] .....	5
3.1	Low-Earth Orbit Parameters.....	87
3.2	Virtex family SEU fault rates ( $SEU \cdot device - day^{-1}$ ). .....	89

## LIST OF FIGURES

Figure	Page
1.1 An example cosmic ray spectrum demonstrating the diversity of atomic nuclei present in the space radiation environment [4].	6
1.2 This figure graphically represents the interaction of a single cosmic ray with atoms in the atmosphere. The result is a cascade of secondary particles.	7
1.3 This rendering of the Earth and its magnetosphere show how the solar wind affects the shape of the magnetic field lines as well as the Van Allen radiation belts [5].	8
1.4 This figure shows the generation of electron-hole pairs as a cosmic ray passes through a CMOS device [6].	11
1.5 This figure shows the energy loss for $24.8 \text{ MeV} \cdot \text{AMU}^{-1}$ for Krypton, Argon and Xenon atoms [7]. It should be noted that the Bragg peaks occur at depth in the material, and in this plot the particle is incident upon the Silicon at the black diamond marker and can be visualized as traveling right-to-left.	13
1.6 This figure shows the device cross-section for a CMOS circuit along with illustrations of TID-associated charge trapping in the gate and field oxides.	15
2.1 Radiation-hardened and commercial technology performance vs. time. Radiation-hardened components generally lag their commercial counterparts by 10-years [8].	24
2.2 This figure shows a block diagram representation of the TMR+Spares architecture. The FPGA is partitioned into nine partially reconfigurable regions each containing a Microblaze processor. Three tiles are active at any time, while the remaining six are maintained as healthy spares. A fault observed by the voter triggers activation of a spare tile and repair of the damaged tile. A configuration memory scrubbing process prevents the accumulation of faults.	29
2.3 This figure shows the ML505 development board used for the implementation of many TMR+Spares w/Repair systems. The board features a Xilinx Virtex-5 FPGA.	30
2.4 This figure shows the system architecture for a 3+13 PicoBlaze TMR+Spares w/ Repair system [9].	32

## LIST OF FIGURES – CONTINUED

Figure	Page
2.5 This figure shows a graphical user interface used for monitoring the state of the many-tile system including which tiles were active, faulted and healthy spares [9]. .....	33
2.6 This figure shows the floorplan of a 64-tile counter system used for demonstrating the TMR+Spares w/Repair architecture. Purple blocks represent the reconfigurable regions of the system [10]. .....	34
2.7 This figure shows the early hardware setup used in this research. It includes the ML605 development board, a general-purpose interface board for receiving radiation sensor inputs and a parallel-to-USB adapter board for communication with a host computer [10]. .....	35
2.8 This figure shows a cross-section diagram of the stripped silicon radiation sensor. The diagram on the left is rotated 90 degrees to the one on the right to show both the top- and back-side channels. ....	36
2.9 This figure shows a custom radiation sensor designed and built at MSU mounted to its accompanying signal conditioning circuit board. This stripped silicon sensor provides 16 front-side channels and 16-back-side channels perpendicularly arranged to give 256 pixels for spatial strike information. ....	38
2.10 This figure shows the complete 1U form factor research hardware stack consisting of a battery board, power board, FPGA board and two radiation sensor boards. ....	40
2.11 This figure shows the custom FPGA board that was designed and developed for this research. It features Xilinx Virtex-6 and Spartan-6 FPGAs. External communication is available through USB and RS-232 interfaces, and local data storage is available on a MicroSD card. ....	42
2.12 This figure shows the Master Serial configuration interface used by the Spartan FPGA. ....	44
2.13 This figure shows the Slave SelectMAP x8 configuration interface used by the Virtex FPGA. ....	45
2.14 This figure shows the command sequence used for initializing a MicroSD card for SPI-mode operation. In practice, the card issues responses to each command, which must be interpreted to ensure proper initialization [11] .....	46



## LIST OF FIGURES – CONTINUED

Figure	Page
2.15 This figure shows the breadboard test setup used for early functionality testing of the radiation sensor. ....	49
2.16 This figure shows the attenuation of laser light as a function of depth in silicon material. ....	50
2.17 This figure shows a block diagram representing the test setup used to optically stimulate the radiation sensor. This testing was used to demonstrate sensor functionality prior to testing at the cyclotron facility. ....	50
2.18 This figure shows the photon energy in electron-volts ( $1eV = 1.602 \cdot 10^{-19} J$ ) for laser wavelengths between 0.8 and 1 micron. ....	52
2.19 This figure shows the custom computer stack under test at the Texas A&M Radiation Effects Facility. A custom translation stage and radiation beam chopper provide spatial and temporal isolation of the beam to allow stimulation of single channels. ....	57
2.20 This figure shows the 2012 MSU HASP payload mounted to the HASP platform, which is suspended from the launch vehicle during the early stages of flight operations. ....	58
2.21 This figure shows the HASP platform with payloads attached during the environmental testing phase of the payload integration operations at the NASA Columbia Scientific Balloon Facility in Palestine, TX. ....	61
2.22 This figure shows the predicted steady-state system temperatures for the HASP 2012 payload [12]. The predicted system temperature during the float phase of the flight was approximately 50 °C. An error of 0.23% was observed between the predicted and actual values [12]. ....	63
2.23 This figure shows the measured steady-state system temperature measured during the HASP 2012 flight. ....	63
2.24 This figure shows the graphical user interface used to parse downlink telemetry packets during HASP flight operations. ....	66
2.25 This figure shows the payload enclosure built by MSU students for flight aboard the HASP scientific balloon platform. ....	67
2.26 This figure shows the collapse of the core voltage rail on the Spartan FPGA. The collapse was triggered by stimulation of the radiation sensor, and caused the Spartan to malfunction during the 2012 flight. ....	70

## LIST OF FIGURES – CONTINUED

Figure	Page
2.27 This figure shows the circuit diagram and truth table for a combinational logic voter circuit. ....	74
3.1 This figure shows the Markov chain for a simple, two-state system. The fault rate is given by $\lambda$ . $S_1$ is a healthy state and $S_2$ is a failed state. $\lambda\delta t$ is the probability of transitioning from $S_1$ to $S_2$ . As there is no recovery process, once the system reaches $S_2$ it remains there with probability 1. ....	81
3.2 This figure shows an exponential reliability curve for a simple two-state system generated using a Markov chain. For this model, the input fault rate was $1.02E^{-08} \text{ ms}^{-1}$ , which is in the upper range of values for a Virtex-6 device in low-earth orbit. ....	83
3.3 This figure shows the structure of a basic TMR system. M0, M1, and M2 are identical hardware components. Their outputs are passed to a majority voter circuit, labeled V, which determines the final system output. A mismatch of one of the systems indicates a failure. ....	84
3.4 This figure shows the Markov chain representation of a basic TMR system. In $S_1$ all three elements are operational. In $S_2$ one of the three elements is operational. In $S_3$ any two of the three units are faulted and the system is in a failure state. ....	85
3.5 This figure shows the reliabilities of TMR and simplex systems. The MTTF of each system is the same, though the TMR system has lower reliability in long-duration applications. ....	86
3.6 This figure shows the Markov model for a TMR+Spare system with a single spare tile and no repair capability.....	87
3.7 This figure shows the Markov chain for a TMR system with repair rate $\mu$ and fault rate $\lambda$ .....	88
3.8 This figure shows the reliability curves for simplex, TMR and TMR with repair implementations. The advantage of implementing a repair process is clearly demonstrated by the substantially lower decay rate of the reliability for the TMR w/repair system compared to simplex and TMR only.....	88

## LIST OF FIGURES – CONTINUED

Figure	Page
3.9 This figure shows the Markov chain for a TMR system with repair rate $\mu$ and fault rate $\lambda$ . It also models the presence of spares in the system and the effect of SEUs occurring during the context switch process. Strikes during a context switch are assumed to result in a system failure as the ability to synchronize the newly replaced tiles is compromised. ....	90
3.10 This figure shows the reliability curves for simplex, TMR, TMR with repair and TMR+Spare(1) implementations. The advantage of implementing a spare tile in addition to a repair process is clearly demonstrated by the substantially lower decay rate of the reliability for the TMR+Spare system compared to the others. The fault rate $\lambda$ and repair rate $\mu$ are the same as in the previous models.....	90
4.1 This figure shows that adding spare tiles to a TMR system increases the MTTF of the system. The repair capability, added to any system, increases the MTTF significantly more than adding spare resources without repair. ....	95
4.2 This figure shows the Markov chain results for a variety of implementations of a Microblaze system including simplex, TMR, TMR w/Scrub, TMR+Spares. This figure shows that the MTTF for a simplex and TMR system is the same. Adding a repair process significantly increases the MTTF. MTTF is further increased by the addition of spare resources, but the improvement diminishes beyond one or two spares until exceedingly high fault rates are encountered. ....	96
4.3 This figure shows the diminishing benefit of arbitrarily increasing the number of spare resources in a TMR+Spares system. There is marked benefit to using TMR+Spares, but increasing the spares beyond one or two results in unnecessarily increased resource utilization. The family of curves presented in this figure shows that this is true regardless of fault rate. ....	98
4.4 This figure shows the effect of the susceptibility of the voter circuit as a single point of failure in a TMR system. Due to its very small size, the voter is more reliable than the TMR portion of the design. However, the reliability of the entire system is reduced when accounting for the reliability of the voter. ....	100

LIST OF FIGURES – CONTINUED

Figure	LIST OF FIGURES – CONTINUED	Page
4.5	This figure shows the effect of the susceptibility of the voter circuit as a single point of failure in a TMR+Spares system.....	100
4.6	This figure shows that scrubber functionality has sufficient impact to negate the necessity of technology partitioning small-area single-points-of-failure in a system. ....	102

## ABSTRACT

This dissertation presents the results of engineering design and analysis of a radiation tolerant, static random-access-memory-based field programmable gate array reconfigurable computer system for use in space flight applications. A custom satellite platform was designed and developed at Montana State University. This platform facilitates research into radiation tolerant computer architectures that enable the use of commercial off-the-shelf components in harsh radiation environments. The computer architectures are implemented on a Xilinx Virtex-6 field programmable gate array, the configuration of which is controlled by a Xilinx Spartan-6 field programmable gate array. These architectures build upon traditional triple modular redundancy techniques through the addition of spare processing resources. The logic fabric is partitioned into discrete, reconfigurable tiles with three tiles active in triple modular redundancy and remaining tiles maintained as spares. A voter circuit identifies design-level faults triggering rapid switch to a spare tile. Blind or readback scrubbing prevents the accumulation of configuration memory faults. The design and results from a variety of integrated system tests are presented as well as a reliability analysis of the radiation effects mitigation strategy used in the system. The research questions addressed by this dissertation are: 1) Does the inclusion of spare circuitry increase system reliability? 2) How do single-points-of-failure affect system reliability? and 3) Does migrating single-points-of-failure to an older technology node (technology partitioning) offer an improvement in reliability?

## BACKGROUND AND MOTIVATION

### The Need for Space Computing

Space science and exploration missions drive the need for increasingly powerful computer systems capable of operating in harsh radiation environments. Advances in sensor technology and space system complexity require greater processing power in order to acquire, store, process and transmit scientific data from the far reaches of the solar system to researchers back on Earth. Data intensive applications, such as image processing, generate massive amounts of data on-orbit and must transmit the information back to Earth for processing. The radio downlink is bandwidth limited, so it is desired that some data reduction processing occurs aboard the spacecraft or satellite. As an example, one commonly used radiation-hardened single-board-computers is the RAD750 by BAE Systems. At a cost of about \$200k per board, this computer delivers performance on par with an early-90s vintage Pentium processor.

The incremental increases in exploration mission ambitiousness necessitate systems that are sufficiently reliable to survive interplanetary journeys from Earth to Mars, Jupiter, Saturn, and beyond. Electronics in space are not afforded the protection of the Earth's atmosphere and magnetosphere. They are under constant bombardment by high-energy particles including an assortment of heavy-ions, electrons, and protons. Particles possessing sufficient energy are capable of inducing bit errors in computer systems through atomic-level ionization mechanisms. The frequency with which these errors are induced is dependent on a multitude of factors including the electronic manufacturing process, spacecraft shielding, spacecraft orbit, and recent solar activity. With interplanetary and orbital mission lifetimes on the order of years it is critical that the on-board computers be able to survive the harshest

radiation environments expected without performance degradation. Consequently, mitigation techniques are required for the reliable operation of electronic systems in space applications.

### Trends in Space Computing

A secondary consideration to radiation hardness is the maximization of performance and power efficiency of space computers. One promising technology for increasing the performance and power efficiency of space computers is the field programmable gate array (FPGA). An FPGA is a reconfigurable logic device which has begun to gain acceptance in aerospace applications as a result of its desirable combination of high performance, low cost, low power and design flexibility [13]. Off-the-shelf FPGAs have the potential to solve the problems of performance lag, excessive cost and inflexibility of current radiation hardened computer systems. An FPGA consists of an array of configurable logic resources, called logic blocks. Each logic block contains an assortment look-up tables, multiplexers, memory circuits, etc. Logic circuits are implemented using the look-up tables, and multiplexers are used in the routing of input/output data into and out of the logic blocks. The interconnection among logic blocks is also programmable. Hardware designs are synthesized by development tools and placed within the resources of the target FPGA. This placement describes the configuration of the device. User designs are stored in volatile static RAM cells, which must be initialized at power-up. This process is known as configuration. In the configuration process the configuration data is read from a peripheral non-volatile storage device and loaded into the FPGA. The file containing the configuration data is known as the bitstream. The bitstream describes the allocation of internal logic re-

sources and signal routing. In SRAM-based devices the configuration data is retained in volatile SRAM cells within the device.

FPGAs, particularly SRAM-based devices, provide a unique flexibility to aerospace systems. A single FPGA can be used to implement multiple system functions by loading different configuration bitstreams based on current system needs or operating mode. This allows hardware sharing by non-concurrent processes that would otherwise require independent hardware systems, resulting in an overall reduction in component count and system complexity. One of the most limited resources on space systems is electrical power. Reducing the amount of power used in computation allows increased system runtime. More advanced configuration features, such as active partial reconfiguration, allow specific portions of an FPGA to be reprogrammed without affecting the operation of the rest of the FPGA. This allows hardware peripherals to be instantiated on an as-needed basis resulting in power savings through an overall reduction in device resource utilization [18].

In space systems, configuration flexibility is highly advantageous as modifying hardware post-launch is problematic for obvious reasons. Yet another benefit of FPGAs is the ability to change the implemented hardware design at any time during a system lifecycle in response to design errors, technology advancement or evolving mission requirements, perhaps extending the useful life of a system or increasing the scientific value by incorporating post-launch advancements in data processing. FPGA design errors uncovered during a mission can be corrected by uplinking a new configuration bitstream to a system.

In high bandwidth, computationally intensive applications FPGAs stand as a compromise between custom ASICs and traditional computer processors. FPGAs enable interfacing with high data rate instruments through the use of optimized custom logic cores. They are capable of processing large data sets such as those generated



by high-resolution imaging systems, and they contribute to on-board data reduction through the use of real-time digital signal processing routines. This data reduction eases the on-board data storage requirements and reduces the amount of downlinked data. In Virtex-6 devices, for example, the logic fabric is capable of clock speeds upwards of 700-MHz and can be used to implement task-optimized hardware logic cores to perform computationally intensive tasks efficiently. The attributes mentioned here are particularly useful as payloads grow in complexity in an effort to maximize scientific value and minimize the bottleneck effect of limited RF downlink bandwidth.

In addition to all the previously mentioned benefits, there is a lower non-recurring engineering cost associated with FPGAs. Whereas ASICs require significant design time resulting in high costs, which are eventually offset by volume savings, FPGAs are more expensive per device, but the development time is significantly reduced. Together these advantages have SRAM-based FPGAs uniquely positioned to close the 10-year performance gap depicted in Figure 2.1.

### The Space Radiation Environment

Earth is under constant bombardment by ionizing radiation originating from a variety of cosmic sources. The radiation environment is best represented as a broad spectrum of particles possessing energies ranging from 10's of keV all the way up to the TeV range [4]. Particles contributing to the space radiation environment include high-energy electrons, protons, and cosmic rays. Cosmic rays are heavy ions accelerated to very high energies. Each element is represented in the cosmic ray spectrum, though some are encountered more frequently than others [14]. Particles below certain energies, and those which are not highly-penetrating do not pose a threat to electronics. Whether or not a particle does pose a threat depends on the

particle energy, the species i.e. particle type and element type, and what material the electronic device under consideration is made of. In modern electronics this material is most commonly silicon. Table 1.1 shows a high-level breakdown of the space radiation environment. Trapped particles constitute a large portion of the lower-energy spectrum while alpha particles, which are simply helium nuclei, protons from the Sun and intergalactic cosmic rays make up the higher-energy portion of the spectrum. Figure 1.1 depicts an example cosmic ray spectrum for elements between Arsenic ( $Z=33$ ) and Neodymium( $Z=60$ ) demonstrating that many particle type are present, and that they vary widely in abundance.

Table 1.1: Space Radiation Environment [1–3]

Type	Energy Range
Trapped protons/electrons	$\leq 100$ MeV
Alpha particles	5 MeV
Solar protons	$\leq 1$ GeV
Cosmic rays	TeV

Electrons, protons and neutrons ejected by the sun during solar flare and coronal mass ejection events interact with Earth’s magnetosphere, sometimes becoming trapped within it [1,2]. The altitude at which a particle becomes trapped is a function of the particle’s energy. The shape and size of the magnetosphere itself is affected by the solar wind, leaving the radiation environment in Earth orbits in a constant state of flux. The degree to which cosmic rays are able to penetrate the magnetosphere is also a function of their energy. Particles with higher energies experience less magnetic deflection than those with lower energies. Sufficiently energetic particles are capable of fully penetrating the magnetosphere and entering the atmosphere, sometimes reaching Earth’s surface. Galactic cosmic rays and charged particles with sufficient magnetic rigidity to reach the atmosphere subsequently undergo energy loss through

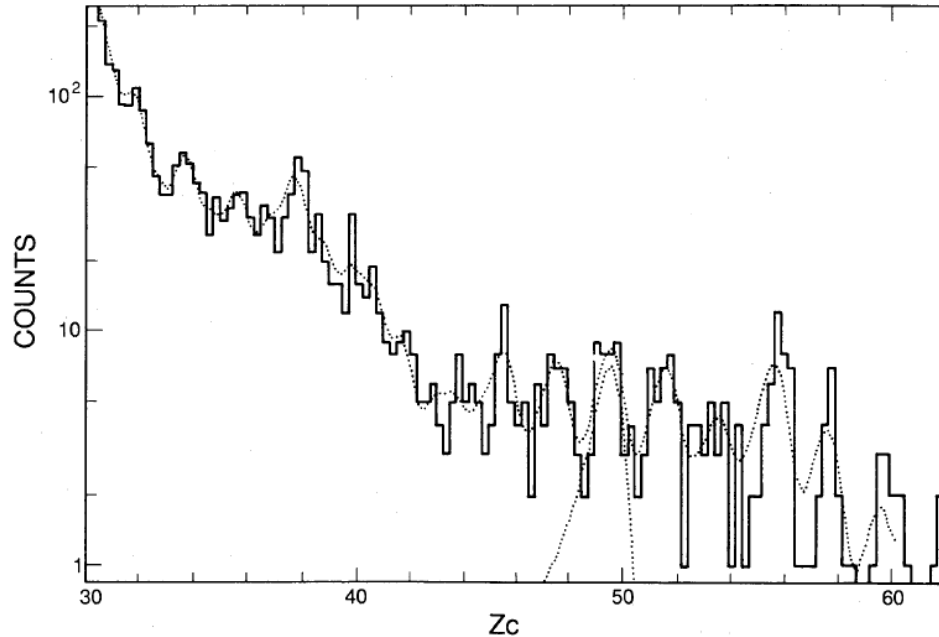


Figure 1.1: An example cosmic ray spectrum demonstrating the diversity of atomic nuclei present in the space radiation environment [4].

collisions with the atmosphere's constituent gas molecules. The result is a cascade of particles with decreasing energy as they approach the ground [2]. These cosmic rays contribute to the terrestrial radiation environment. Figure 1.2 graphically depicts the interaction between high-energy particles incident on the atmosphere and the resultant cascade of secondary radiation.

The phenomenon of particle trapping by magnetic field lines gives rise to persistent regions of elevated radiation. These regions, discovered in 1958 as a result of the Explorer 1 and Explorer 3 satellite missions, are known as the Van Allen radiation belts [15]. The Van Allen belts vary in shape, size and intensity depending on solar activity [16]. They consist of two primary bands of radiation; an inner band and an outer band. The inner band, peaking approximately one Earth radius above the equator, consists largely of high-energy protons of 10-100 MeV [2]. The weaker outer band contains energetic electrons and changes more dramatically based on solar

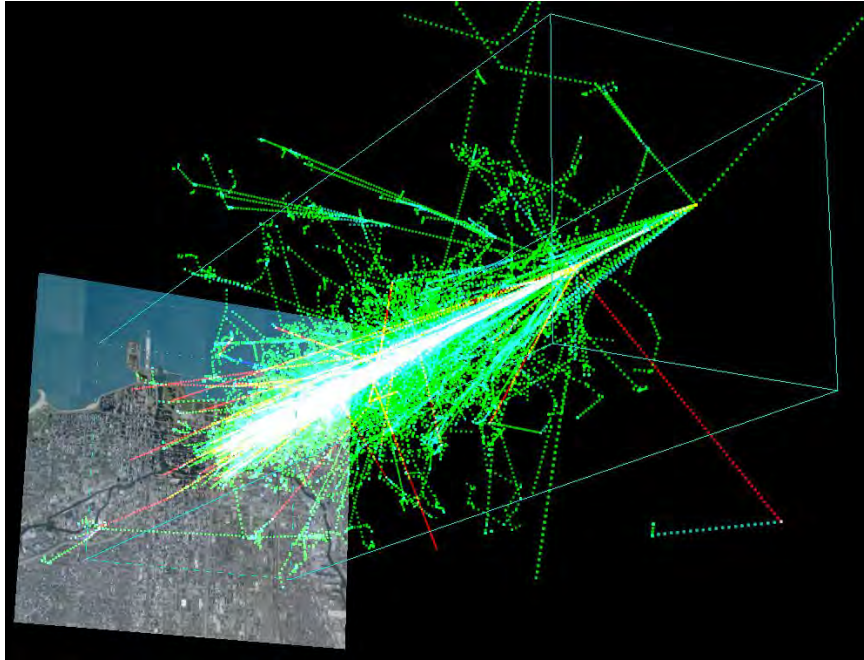


Figure 1.2: This figure graphically represents the interaction of a single cosmic ray with atoms in the atmosphere. The result is a cascade of secondary particles.

conditions [17]. Figure 1.3 depicts the solar wind interacting with Earth's magnetic field lines to affect the shape and size of the Van Allen radiation belts.

The space radiation environment consists of a wide variety of energetic particles. The severity of the environment is a function of recent solar activity and location within the magnetosphere. Each planet in the solar system, as well as the interstellar medium possesses a unique radiation environment each presenting unique challenges to the deployment of modern electronic systems. The radiation environment is a key consideration in spacecraft design. The following section expands on the effects of ionizing radiation on electronic circuits, particularly those designed using metal-oxide-semiconductor field effect transistors (MOSFET).

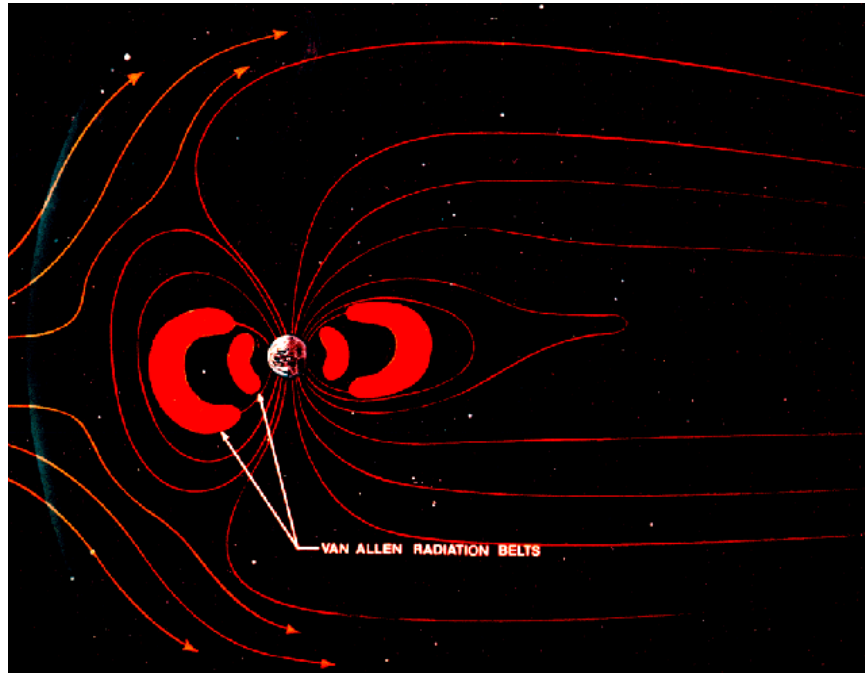


Figure 1.3: This rendering of the Earth and its magnetosphere show how the solar wind affects the shape of the magnetic field lines as well as the Van Allen radiation belts [5].

### Radiation Effects on Modern Electronics

To achieve the reliability required by space-based electronic systems, adverse effects caused by ionizing radiation must be mitigated. These effects can be divided broadly into two categories: cumulative effects and transient effects. As the name suggests, cumulative effects accumulate within a device over time. The rate at which these effects accumulate is a function of the severity of the radiation environment and the duration of exposure. Total ionizing dose (TID) is perhaps the most commonly used measure of cumulative effects in space electronics. TID represents the amount of energy deposited in a material per unit of mass and carries the unit *rad* [2]. The dose survivable by an electronic device is often specified in kilo-rad, or krad, representing

$10^3$  *rad*. Transient effects are the immediate effects of radiation interactions with electronics. Termed single event effects (SEE), these interactions are brief in duration, but their effects may linger in an affected circuit as state changes to memory elements. SEEs result from interaction with a single particle of ionizing radiation. SEEs are of primary concern in space electronics. They are induced by the transient charge generated by a high-energy particle passing through a semiconductor material. These interactions create erroneous voltages within semiconductor substrates, which may be latched into digital memory elements. The severity of SEEs, in terms of its effect on the operation of the affected system, is highly dependent upon where it occurs within the system. In synchronous digital systems the time of the strike relative to the system clock edge is also a critical factor in determining the end result of the interaction.

Many modern digital integrated circuits are designed using the MOSFET as the fundamental design building block. Complimentary MOS (CMOS) circuits have gained wide acceptance in integrated circuit technology as a direct result of their low power consumption, reliability, and ability to efficiently implement complex logic functions [18]. The performance of CMOS devices continually increases as a result of improvements in manufacturing processes, primarily the ability to manufacture transistor features on continually smaller scales in accordance with Moore's Law [19]. The MOSFET transistor is a voltage-controlled current device whose proper operation depends on well-known charge distributions within the device. These charge distributions include the doping concentrations of the source and drain, and the base semiconductor material which constitutes the bulk substrate within which a conduction channel is induced. These electron-hole carrier concentrations, along with the feature dimensions, namely channel length, width, and gate oxide thickness define

the voltage-current relationship of the device. Changes to the charge distribution adversely affect device performance.

In terrestrial applications there are few environmental factors that affect MOSFET operation. Due to their popularity, and the fact that they essentially define the state-of-the-art in modern electronic systems, designers have sought to use MOSFETs and other semiconductor devices in space applications. Unfortunately, the space environment is not as forgiving as that within Earth's atmosphere. On Earth, electronic devices are shielded from the vast majority of naturally occurring ionizing radiation thanks to the atmosphere and the magnetosphere [1,2]. On Earth there are few natural phenomena capable of randomly injecting charge into MOSFET devices aside from exceptionally high-energy, highly penetrating cosmic rays. In space the reliability of the MOSFET is diminished as a direct consequence of interactions with high-energy, ionizing radiation wherein extraneous charge is generated and sometimes trapped within the semiconductor materials. As the reliability of the basic building block decreases so too does that of the electronic system as a whole.

### Radiation Effects Mechanisms

Ionization through particle interaction is a natural process in which electrical charge is deposited in a material as a high-energy particle passes through it. Generally characterized in terms of their kinetic energy, the particles responsible for this ionization are often prefaced with the terms "energetic", or "high-energy", indicating that said particles have been accelerated to relativistic speeds and possess incredible amounts of kinetic energy. As these particles pass through a material they lose energy, imparting it to the material. The amount of energy imparted is dependent on a variety of factors including initial particle energy, the material being transited, and the

particle range in the material [20]. Figure 1.4 shows a charge track of electron-hole pairs generated as a result of ionization by a high-energy cosmic ray.

Linear energy transfer (LET) is used to quantify the rate of energy loss of the incident particle as a function of distance traveled into the material. There are two application-dependent definitions of LET. In biological dosimetry applications, LET carries units of  $MeV \cdot cm^{-2}$ , which are implied by the mathematical definition of the quantity. This quantity is also known as “linear electronic stopping power”. In electronics applications, LET carries units of  $MeV \cdot cm^2 \cdot mg^{-1}$ , which is the same quantity only scaled to account for material density. This quantity is also known as the “mass stopping power”. As these are descriptions of electronic interactions, neither of these quantities accounts for energy loss due to elastic collisions with atomic nuclei. Equation 1.1 shows the equation for LET. In this equation,  $dE$  represents the differential energy lost by the particle as it travels a differential length  $dx$  into the material. The energy lost by the particle is gained by the material and manifests as electron-hole pairs as valence electrons are excited into the conduction band.

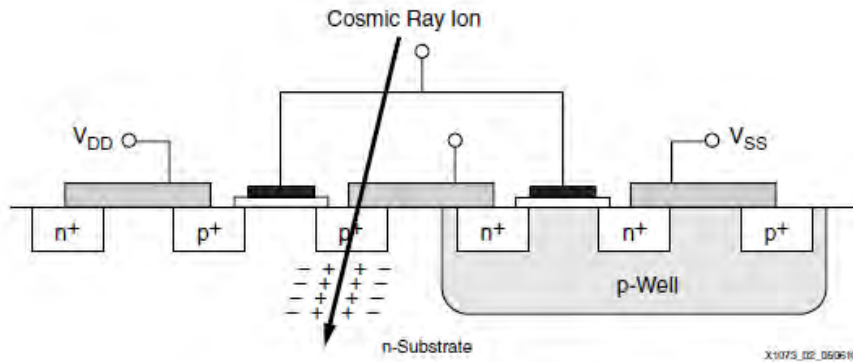


Figure 1.4: This figure shows the generation of electron-hole pairs as a cosmic ray passes through a CMOS device [6].

$$LET = \frac{dE}{dx} \quad (1.1)$$



LET provides a material and path-length dependent measure of energy transfer. After multiplying by material density, one is left with a measure of the amount of energy deposited per unit length of penetration. A final integration over the range of particle penetration results in the amount of energy in  $MeV$  deposited [20]. Figure 1.5 shows the energy loss for  $24.8 MeV \cdot AMU^{-1}$  Krypton, Argon and Xenon in silicon.

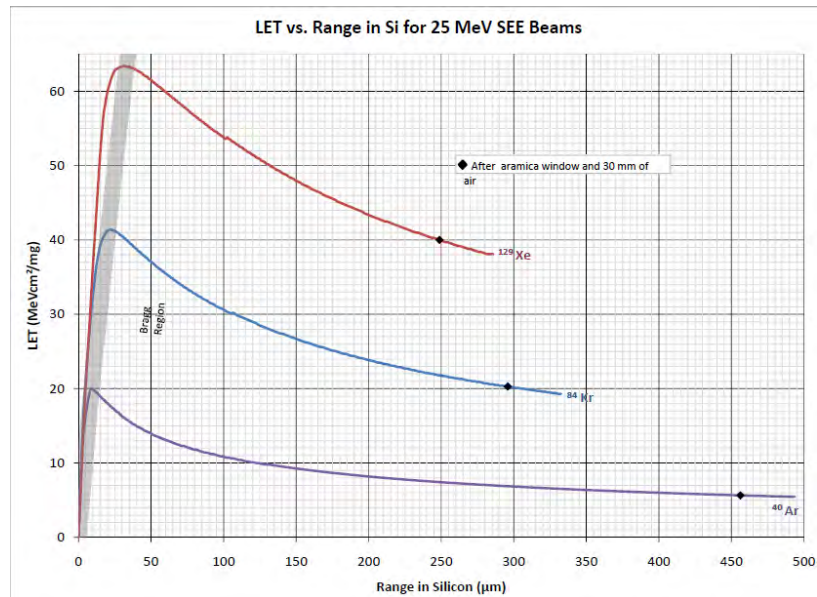


Figure 1.5: This figure shows the energy loss for  $24.8 MeV \cdot AMU^{-1}$  for Krypton, Argon and Xenon atoms [7]. It should be noted that the Bragg peaks occur at depth in the material, and in this plot the particle is incident upon the Silicon at the black diamond marker and can be visualized as traveling right-to-left.

In a semiconductor, the amount of energy required to excite a valence-band electron into the conduction band is defined as the band gap energy [21]. This is commonly taken to be  $1.1eV$  in intrinsic Silicon. Exciting a valence electron to the conduction band is synonymous with generating an electron-hole pair. This newly energized electron is free to move about the material and will do so in accordance with any locally-applied electric fields. The number of electron-hole pairs generated

can be estimated by dividing the energy deposited by the band gap energy. This generation of electron-hole pairs is the source of most radiation-induced effects in modern electronics.

### Total Ionizing Dose

Defects in the form of radiation-induced broken bonds give rise to persistent holes located in the oxide layers [22]. These holes tend to accumulate in material defects in the gate oxide near the silicon/oxide boundary, and similarly in the field oxide regions [23]. This accumulation of charge results in the performance degradation associated with TID. The charge accumulation in the gate oxide is problematic in terms of device functionality. The placement of charge at the gate interferes directly with how the MOSFET operates. Should sufficient charge accumulate in the gate oxide, a conduction channel will be induced in the same way as it is when a control voltage is applied to the gate terminal of the device. Removal of the gate control voltage no longer has the ability to eliminate the conduction channel, and the MOSFET is effectively stuck in an active state. In an NMOS transistor this equates to an inability to turn the device off. In a PMOS transistor, this equates to an inability to turn the device on. The second effect associated with charge accumulation in oxide regions is increased leakage currents both through the gate oxide as well as between adjacent transistors. Though this doesn't necessarily impact the functionality of the device, it will negatively impact the circuit from a system standpoint as the operating efficiency is reduced. In battery powered applications this is a significant effect as the system runtime will be affected. Increased leakage current may also lead to thermal problems if the system design is unable to properly dissipate the additional thermal energy. In this way, a radiation-induced electrical effect may easily become an unanticipated

thermal design issue. Figure 1.6 shows a cross section of a CMOS device along with illustrations of TID-associated charge accumulation in the gate and field oxides. This charge build-up is what contributes to the increased leakage currents associated with TID.



Figure 1.6: This figure shows the device cross-section for a CMOS circuit along with illustrations of TID-associated charge trapping in the gate and field oxides.

### Single Event Effects

Generated electrons and holes will dissipate quickly through the process of recombination. The creation and subsequent dissipation of these electron-hole pairs in the semiconductor substrate as a result of penetration by ionizing radiation is responsible for the generation of transient effects. During the process of charge recombination, electron-hole pairs flow within the semiconductor material. This flow of charge constitutes a short-lived current or voltage within the circuit. It is this transient flow of charge that gives rise to SEEs in MOSFETs.

These effects are commonly categorized by the nature in which they affect a device. The effects are transient in that the initial effect on the device is considered temporary despite the fact that some of the errors induced may persist in circuit elements. Some of these effects are nondestructive while others may cause latch-up and device failure [24]. The operational faults induced by SEEs vary in severity from temporary glitches propagating through combinational logic circuits to the corruption of system

memory contents. The nature of the fault induced in the system by this current is highly dependent on the time at which the event occurs, especially with respect to the clock edges in synchronous digital circuits, and the physical location of the event. SEEs are considered “soft errors” in that they are recoverable by rewriting data to the affected element to restore it to a correct state.

The most basic single event effect is known as a single event transient (SET). This effect is the aforementioned transient current generated as a high-energy particle passes through a device. As the name implies, this effect is short-lived with current pulse widths as short as a few hundred picoseconds [25]. These transient currents may cause voltage or current signals within the system to temporarily change value. The seriousness of such a fault is application and system dependent. A SET occurring at the input to a synchronous, bistable circuit element within the data setup and hold window has the potential to be latched into the system. Such an occurrence is known as a single event upset (SEU). Depending on where the radiation strike occurs, the digital memory contents may be corrupted regardless of when the strike occurs relative to the system clock. In CMOS random-access memory (RAM), for example, a strike occurring at the drain of a transistor in the off state can directly cause the state of the element to change [26].

Some integrated circuit devices have dedicated circuitry for performing system tasks, such as clock management, direct memory access cores, general-purpose input/output registers, communications transceivers, etc. each of which is likely to contain susceptible storage. Corruption of these memory elements would result in undesirable behavior or performance degradation. SEUs occurring in these areas which result in device malfunction are called single event functional interrupts (SEFI). A radiation tolerant system must mitigate faults occurring at both the application layer and in device control components.

Destructive SEEs include single event latchup (SEL), single event burnout (SEB), single event gate rupture and single event snapback [24]. These failure modes cause permanent damage to system components. In the CMOS fabrication process, parasitic bipolar junction transistors are created between the NMOS and PMOS transistors. The activation of these BJTs creates a low-impedance path between the system power supply and ground [24]. In space radiation environments the charge generated by ionizing radiation can be enough to activate the parasitic transistors resulting in SEL. With SEL the best-case scenario is an observed increase in current consumption. Should the parasitic transistors enter a thermal runaway condition it is possible for the current to exceed damage thresholds and permanently damage the device [24]. In either case, the system suffers degraded performance either as total system failure or increased power consumption. Increased power consumption is highly problematic in space as the power available to operate a system is an extremely limited resource. SEL is mitigated in some radiation hardened devices through the inclusion of isolation trenches between the pull-up and pull-down networks. SEB, single event gate rupture and single event snapback are more commonly associated with vertical transistor architectures rather than the typical lateral layout of CMOS transistor networks. As the focus of this discussion is on SEUs in commercial CMOS devices the reader is referred to [24] for further information on these failure mechanisms.

The balance of this paper describes the previous work that has gone into the development of this research topic. Much research and development has been accomplished over the past seven years and the work presented here owes to the success of previous students. Extensive engineering work was required to advance the research hardware to its current state. This work is detailed in a subsequent chapter. The development of research hardware led to system testing at many different levels, including bench-top optical testing, radiation effects testing at a cyclotron facility, and a handful of

flights aboard high-altitude scientific balloon platforms. These tests and outcomes are discussed. Finally, current research which adds to the understanding and estimation of system reliability is presented followed by a few comments on future work to be completed.

## PREVIOUS WORK

### Radiation Effects Mitigation

Many techniques for mitigating the various radiation effects have been devised, implemented and tested. When people think of radiation protection, generally, the first thought is shielding. Shielding from radiation involves the use of a physical barrier between the electronic device and the external radiation environment. The use of shielding is common, and required, in many terrestrial nuclear applications for the purpose of protecting the health of employees as well as preventing radioactive material from polluting the environment at nuclear sites. In terrestrial applications the shielding can be of any size and arbitrarily thick. Any type of material, regardless of weight, can be used to achieve shielding specifications. In space applications, extensive shielding to protect sensitive electronics from radiation is cost prohibitive using current launch vehicle technology. Additionally, some intergalactic cosmic rays are sufficiently energetic to penetrate any amount of shielding that could reasonably be applied to a spacecraft [2]. In fact, shielding may worsen the radiation environment for the electronics as a cosmic ray generates a stream of secondary particles as a result of its interaction with the shielding material [2]. With shielding eliminated as a means of protecting electronics from radiation in space, integrated circuit designers and electronics engineers have worked diligently to come up with radiation hardened parts and systems for use in space applications. Each type of radiation effect demands its own approach to mitigation. A summary of these approaches follows.

## TID Mitigation

TID mitigation is performed through various manufacturing techniques designed to minimize imperfections within the transistor gate oxide, at the silicon/oxide interface, and within the crystal lattice of the substrate. As a semiconductor device is exposed to ionizing radiation, small portions of the generated charge become trapped in oxide layers, at semiconductor/oxide boundaries, and within crystal lattice imperfections throughout the device. Since the gate oxide is particularly susceptible to charge accumulation, reducing its capacity to trap charge in the first place is paramount to eliminating TID. The ability of an oxide to trap charge is a function of its thickness, and is diminished in modern manufacturing process nodes [23]. The hardening of integrated circuit devices against TID currently uses local oxidation of silicon (LOCOS) [27] or shallow trench isolation (STI) [28] to isolate transistors and reduce leakage effects. These modifications in design layout fall under a category of radiation hardening known as radiation hardened by design (RHBD). Another type of radiation hardening, known as radiation hardened by process (RHBP) seeks to reduce charge trapping in the semiconductor materials by minimizing defects in the substrate and oxide layers during the manufacturing process. Polishing of wafer surfaces and careful control of oxidizing conditions helps minimize oxide interface defects thus reducing the number of possible charge traps in the device [29].

RHBP and RHBD processes have successfully achieved TID immunity up to 1 Mrad(Si). As gate oxide thicknesses have decreased with advances in manufacturing technology, TID has become less of a consideration in general as both the amount of charge buildup in an oxide as well as the resultant threshold voltage shift are directly proportional to oxide thickness [23]. Manufacturing techniques have significantly reduced total ionizing dose as a consideration when using parts designed at



45-nm or less [23]. For example, the Virtex-6 implemented in 40-nm technology has achieved 380 krad and 1 Mrad with reduced timing [30]. As a result, off-the-shelf components are becoming more inherently TID immune. Modern gate oxide thicknesses effectively eliminate TID as a design consideration when using cutting-edge off-the-shelf components.

### SEE Mitigation

The primary effect to be mitigated in modern CMOS devices is the SEU. These upsets vary in severity depending upon where they occur, but must be mitigated globally in order to achieve the reliability required by space systems. A handful of approaches exist for mitigating SEUs, many of which occur at the design level. These approaches typically focus on minimizing the amount of charge generated within the semiconductor substrate. By building the transistor structure on top of an insulating material it becomes less susceptible to SEEs as the amount of charge generated is reduced. This technology is known as silicon-on-insulator (SOI). Electrons in insulator require much more energy to be excited into the conduction band than those in a semiconductor. This property of insulators results in less charge generation under radiation exposure.

In addition to process mitigation there are many techniques for eliminating SEUs at the architectural level [29, 31–33]. A common system-level technique for SEU mitigation is triple modular redundancy (TMR) [34–37]. TMR systems run three identical components in parallel. The outputs of these concurrent operations are voted upon by a majority-rules voting circuit to determine if any of the system components are faulty and to prevent erroneous outputs from propagating through the system. TMR has been applied with varying granularity ranging from bit-level triplication of circuits

to system-level triplication of major electronic components [38,39]. This approach to mitigating radiation effects through redundancy is rooted in early theoretical work concerning how perfectly reliable computing machines could be created from inherently unreliable components, e.g. mechanical relays [40]. Though the technology has changed, this technique for mitigating component-level faults has remained relevant.

### Memory Scrubbing

Another technique for imparting radiation hardness on a system is to ensure the integrity of memory contents through the use of memory scrubbing. In this process, the contents of memory locations are periodically rewritten with known good data. This prevents errors in memory from accumulating, and reduces the likelihood of using corrupted data values in computations. Scrubbing can either be blind, or use readback technology. Blind scrubbing simply overwrites the contents of the memory regardless of its validity. A readback scrubbing process reads the contents of a memory location, compares it to the desired value, and only performs a write operation if there is a discrepancy. In FPGA systems memory scrubbing is used as a way to maintain the configuration memory contents. The presence of a scrubber significantly increases the overall reliability of a system.

### Drawbacks of Current Mitigation Techniques

The smaller geometries result in devices that are increasingly sensitive to SEEs [41]. Therefore, radiation hardening efforts for space electronics must focus on mitigating single event effects. Many of the radiation mitigation techniques discussed in the previous section rely on design-specific features or process-level modifications to

achieve adequate hardness. Though effective, these techniques add substantial costs to the manufacturing process. Once designed and manufactured, new devices must undergo extensive testing to demonstrate their radiation hardness. These tests are very expensive, yet another cost that is passed on to the customer. Since there is simply not a large market for such devices, radiation hardened parts do not benefit from cost reductions associated with volume manufacturing. The end result is radiation hardened devices that are significantly more expensive than their off-the-shelf counterparts. One of the more popular radiation hardened single board computer systems is the BAE RAD750, which has seen use in high-profile space systems such as the Deep Impact probe, the Mars Reconnaissance Orbiter, and the Mars Science Lab (MSL). The radiation hardened version of the RAD750 processor is manufactured using a 150-nm process, provides 400 Dhrystone MIPS at 200-MHz [42] and comes at a cost of about \$200,000 for the system used in the (MSL). This price far exceeds the cost of a system designed using off-the-shelf components with similar performance specifications.

In addition to being more expensive, radiation hardened components, in general, exhibit lower performance than commercial devices. The performance lag is a result of the manufacturing techniques used to protect the devices from radiation. The techniques use older process nodes resulting in larger minimum feature sizes and consequently slower switching times, greater power consumption and lower performance. Also, these design and layout techniques add area to the circuitry thus decreasing the performance further. Radiation hardened microprocessors generally lag commercial devices in performance by ten or more years. A 10-year performance lag is substantial considering Moore's law as it applies to terrestrial computing applications. Moore's law states that the number of transistors on a single wafer doubles every 1.8 years [19] and along with that comes an increase in computational power. Radiation tolerant



should the error occur in user memory. However, errors occurring in configuration memory manifest as changes in the physical circuitry implemented on the device. The way these effects manifest in FPGAs is very specific to the device. There are two conceptual layers to be considered in mitigating SEEs in FPGAs: configuration layer and application layer. The configuration layer contains hard logic cores and administrative circuits used for defining the behavior of the device. The application layer contains user-defined circuitry, application memory, and interconnect that is used to perform a desired task.

Programmable logic devices derive their functionality from data contained in configuration memory. An SEU occurring in the configuration memory region of a device results in a corresponding change to the implemented circuit. When such a change occurs in a sensitive configuration bit within a design, the system will cease proper operation. The circuit will continue operating improperly until such time as the correct data is rewritten to the corrupted configuration memory location either through a full device configuration or through the configuration memory scrubbing process. These errors can affect either the functionality of the circuit or the interconnect between logic resources. For example, a radiation induced bit flip in a look-up table represents a change to the truth table representing the logic function. Simply put, this is equivalent to changing a design-level AND gate to an OR gate. SEUs affecting the routing are analogous to opening or closing a switch; making or breaking a connection between logic resources. This is equivalent to unplugging wires in a circuit, or adding extraneous wires to a circuit. Either is likely to adversely affect circuit functionality.

SEUs may also occur in memory elements located within the application layer of a design. In this case, radiation induces an upset in user memory. Such an error may go unnoticed until the data is retrieved from memory and used in a computation producing an incorrect result, or it may be observed immediately as an unexpected

change of state in a hardware state machine. Data upsets may be tolerable and simply interpreted as system noise, but upsets to state memory contents or other system function registers may cause system failure. The consequences of system failures as a result of single event upsets occurring in user memory are very diverse and system dependent.

Much research has been performed regarding SEU mitigation in SRAM-based FPGA devices. The most widely adopted technique for fault mitigation is to use a combination of triple modular redundancy (TMR) [34, 38, 43, 44], which detects faults and prevents errors from propagating through the system, and configuration memory scrubbing, which prevents faults from accumulating in the TMR system by maintaining the integrity of the device configuration SRAM. The combination of these techniques is commonly referred to as TMR+scrubbing. The benefits of implementing TMR in terms of device reliability have been demonstrated [45, 46]. More recently, support for error detection and correction codes to protect the block RAM on Xilinx FPGAs has been included in the device architecture [47]. Additionally, configuration memory error detection and correction is implemented in configuration primitives available to system designers. These primitives enable detection of configuration faults and correction by a user design [47].

### MSU's Approach to Reliable, High-Performance Space Computing

The research vision guiding this work seeks to create a radiation tolerant, SRAM-based FPGA computer system for space flight applications. This vision stems from previous work in radiation effects mitigation using traditional TMR design techniques coupled with some form of configuration memory management. The development of advanced configuration capabilities by Xilinx, including active partial reconfiguration

and configuration memory readback, enabled hardware cores to be instantiated on an as-needed basis. This allows logic resources to remain idle, consuming less power, until they are needed in a design. With the goal of increasing computational power available on space platforms Xilinx Virtex family FPGAs were targeted for use. These devices are typically the highest performance SRAM-based FPGAs available.

Implementation of such a system requires the use of very advanced design tools. Chief among these tools is active partial reconfiguration, which allows specific regions of the FPGA to be configured independently at runtime. Use of these techniques necessitates access to the configuration interface, which has been accomplished in several different ways as the research has progressed. Configuration memory readback is another configuration tool used extensively in this research. After programming, readback allows the contents of the configuration memory to be read by an external device, the configuration controller in this case, and check for accuracy against an uncorrupted version known as the “golden copy”. The golden copy is stored in a memory technology, such as FLASH, that is less susceptible to single event effects.

The focus of this research has been to build upon the traditional fault mitigation techniques in an effort to increase the performance and reliability of SRAM FPGAs for aerospace applications. The approach to accomplishing this is to combine blind or readback scrubbing, active partial reconfiguration, and TMR in a specific way to efficiently detect and mitigate radiation induced faults while minimizing fault recovery time. To accomplish this, an FPGA is partitioned into discrete, partially reconfigurable processing resources. These are referred to as “tiles”, and they represent the granularity of the TMR implementation. Our current research system, implemented on a Xilinx Virtex-6 FPGA, consists of nine tiles each of which contains a Microblaze microprocessor. During normal operation, three tiles are active and constitute an active triad with the remaining tiles reserved as spares. The outputs of the active tiles

are routed through a multiplexer to a majority voter to form a coarse-grained TMR system. In the background, and without impacting the operation of the active triad, a scrubbing routine maintains the spare tiles using active partial reconfiguration. An external configuration controller monitors the status of the system, controlling which tiles are active, performing configuration memory scrubbing, and tracking the status of each of the tiles. The configuration controller is responsible for detecting and recovering from faults in the system. In the event of a fault in a member-tile of the active triad the affected tile is taken off-line and replaced with a healthy spare tile. After synchronization, the triad resumes operation with its new member. In the background, the faulted tile is then repaired using partial reconfiguration and reintroduced to the system as a healthy spare.

In addition to its primary research function, this system was useful for developing the requisite reconfigurable computing tools such as SelectMAP device configuration, active partial reconfiguration, configuration memory blind scrubbing, and configuration memory readback. This approach is termed TMR+Spares indicating the mitigation of errors through the use of TMR resources and recovery from faults using spare processing resources. Figure 2.2 shows this concept graphically.

### Reconfigurable Computing for Aerospace Applications

Early MSU work on this project used the Virtex-5 ML505 and Virtex-6 ML605 development boards for hardware implementation. Two systems were created which formed the foundation of later research [9]. These systems used Xilinx Microblaze and Picoblaze soft-processor cores as their primary processing resources. The Picoblaze is a smaller, lower performance version of the Microblaze allowing more tiles to be implemented on a given FPGA. Systems included a 3+1 Microblaze system and a



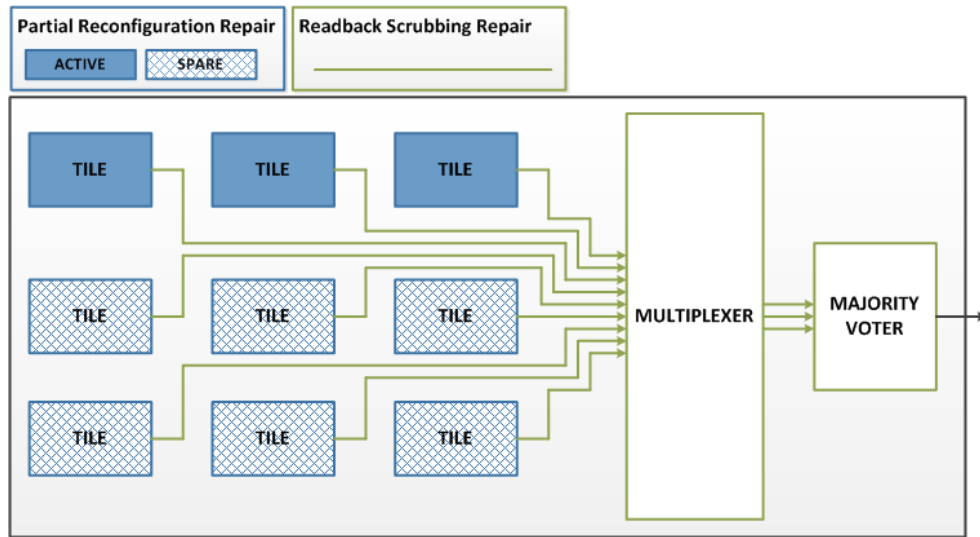


Figure 2.2: This figure shows a block diagram representation of the TMR+Spares architecture. The FPGA is partitioned into nine partially reconfigurable regions each containing a Microblaze processor. Three tiles are active at any time, while the remaining six are maintained as healthy spares. A fault observed by the voter triggers activation of a spare tile and repair of the damaged tile. A configuration memory scrubbing process prevents the accumulation of faults.

3+13 Picoblaze system with varying modes selectable via partial reconfiguration. Each of these systems implemented an active processing triad and contained spare processing resources. Figure 2.3 shows the ML505 development board upon which the early reconfigurable computer architectures were implemented.

The focus of the earliest system [9] was on providing a system featuring multiple operating modes including a low-power mode, parallel processing mode, and radiation tolerant mode. The low-power mode implemented a simplex system with no redundancy and no error detection/correction considerations. This mode would be selected in benign radiation environments to perform computationally simple tasks. The parallel processing mode increased the performance of the system by partitioning hardware tasks and assigning each task to its own dedicated hardware core. Again, this mode was not radiation tolerant, but could be used in a benign environment re-



Figure 2.3: This figure shows the ML505 development board used for the implementation of many TMR+Spares w/Repair systems. The board features a Xilinx Virtex-5 FPGA.

quiring to perform tasks requiring greater system performance. The third mode was a radiation tolerant TMR+Spares implementation which could be activated should the radiation sensor detect substantial particle fluxes. This mode ran three processing tiles in TMR, reserving spares for replacement should an active tile be faulted. As part of the recovery philosophy, tiles that had repair attempts performed unsuccessfully were marked as TID damaged and removed from consideration in future use. The main benefit of changing tiles rather than halting operation while a repair is undertaken is that the time to repair a faulted tile is substantially shorter than fully reprogramming the device. This minimization of repair time reduces the susceptibility of the system to multiple bit upsets. Figure 2.4 shows the system architecture for

the 3+13 PicoBlaze reconfigurable computing system. Figure 2.5 shows the graphical user interface used for monitoring the status of this system.

As a follow-on to the earliest work on this project, the technology was migrated to the Virtex-6 device family [10]. As before, computer systems were created which contained myriad spare processing resources. The resources were maintained in the same way as before using a scrubber routine to maintain the integrity of the design. For the first time, a radiation sensor, described in the next section, was coupled with the computer system to provide a degree of environmental awareness. In addition to the creation of more TMR+Spares systems software interfaces were developed which allowed visualization of the system state including which tiles were active, which tile were faulted, and the status of the scrubber activity. Additionally, the ability to simulate tile faults at the design level showed the response of the system to errors. Eventually, the ability to simulate these faults was linked to a radiation environment model allowing orbital fault rates to be approximated and system operation to be demonstrated for a variety of radiation environments. These analysis tools were important in gaining confidence in the overall system architecture and showed its viability in high fault rate environments. Figure 2.6 shows the FPGA floorplan for a 64-tile counter system implemented on the ML605 development board. Figure 2.7 shows the hardware configuration including the ML605 board housing the Xilinx Virtex-6 FPGA and various interface boards for interfacing with the radiation sensor. This image motivates the move to design custom research hardware, which is discussed in a subsequent section.

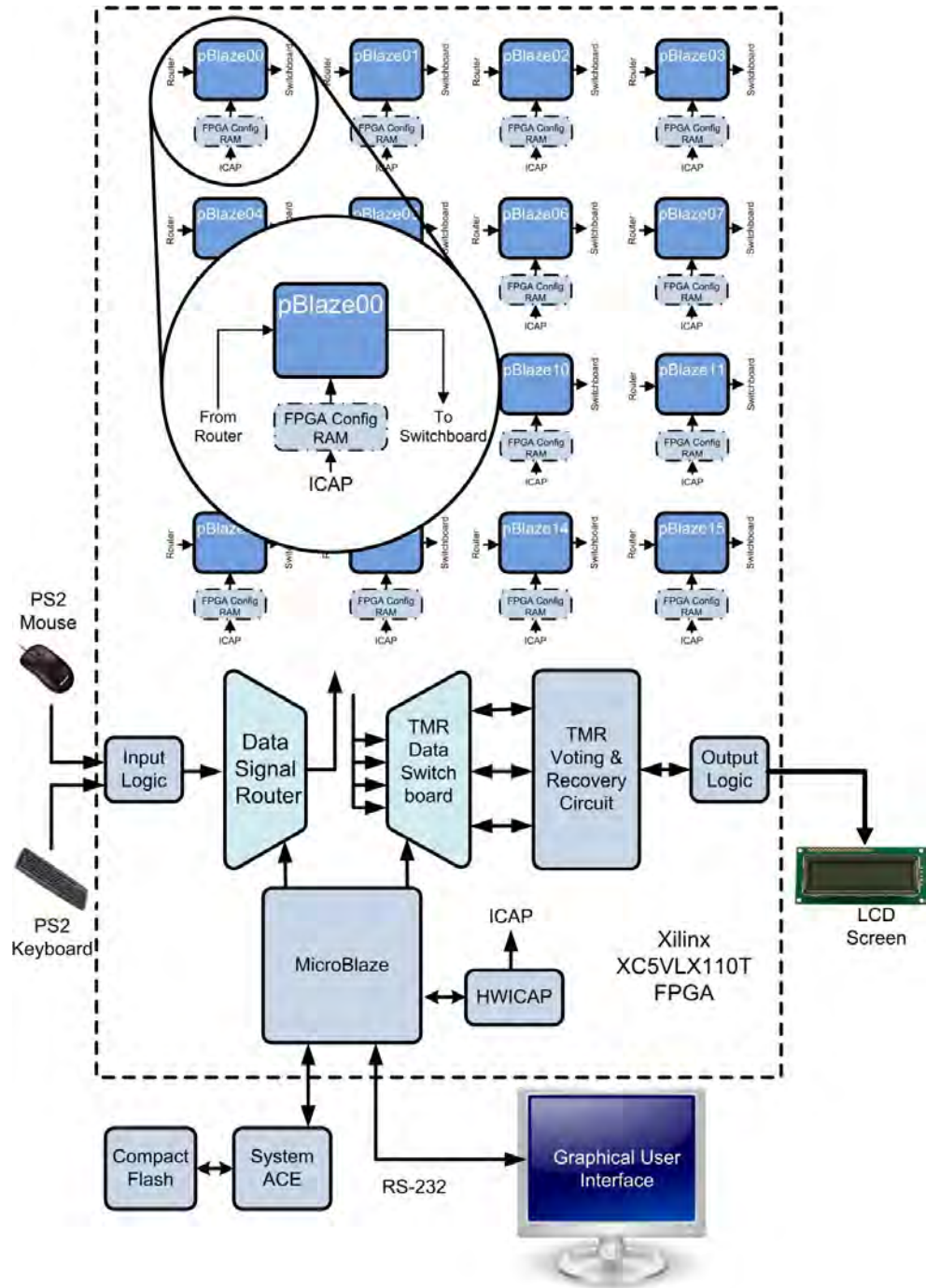


Figure 2.4: This figure shows the system architecture for a 3+13 PicoBlaze TMR+Spares w/ Repair system [9].

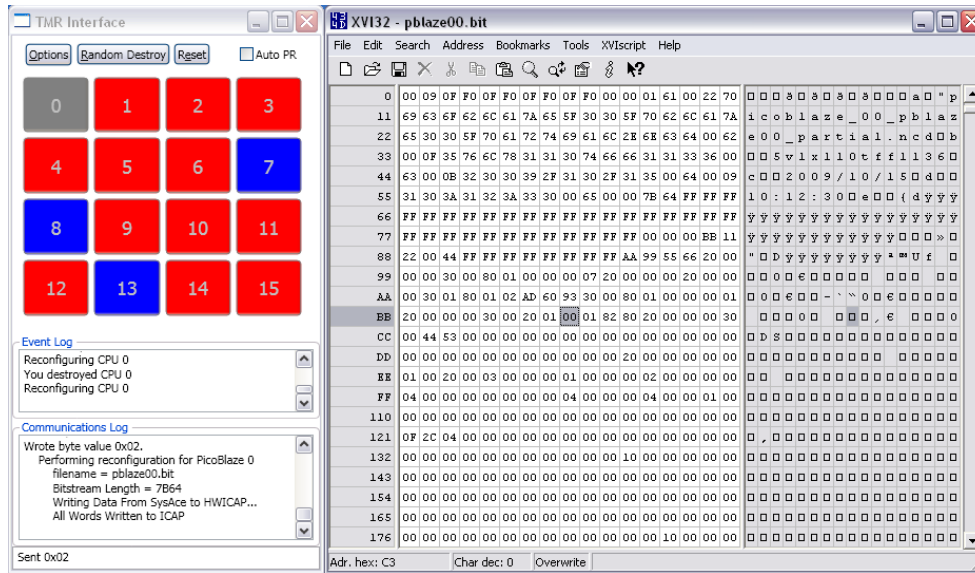


Figure 2.5: This figure shows a graphical user interface used for monitoring the state of the many-tile system including which tiles were active, faulted and healthy spares [9].

## Radiation Sensor

The radiation environment of space fluctuates wildly depending on a system's location within a given orbit, the type of orbit, solar conditions, etc. Recognition of this led to the incorporation of a radiation sensor in the radiation tolerant computer system. Designing for the worst case would require the error detection processes to run at full capacity even when the radiation environment is relatively benign. This results in unnecessary and inefficient expenditure of power. To provide an awareness of the radiation environment a custom silicon radiation sensor was designed and developed. This sensor was designed and fabricated at MSU, and provides 16 front-side channels and 16 back-side channels. The front- and back-side channels are oriented perpendicular to one another. This orientation allows spatial information to be extracted by examining which channels are stimulated simultaneously. The result is an array of 256 "pixels" that indicate the location of a strike. In addition to strike

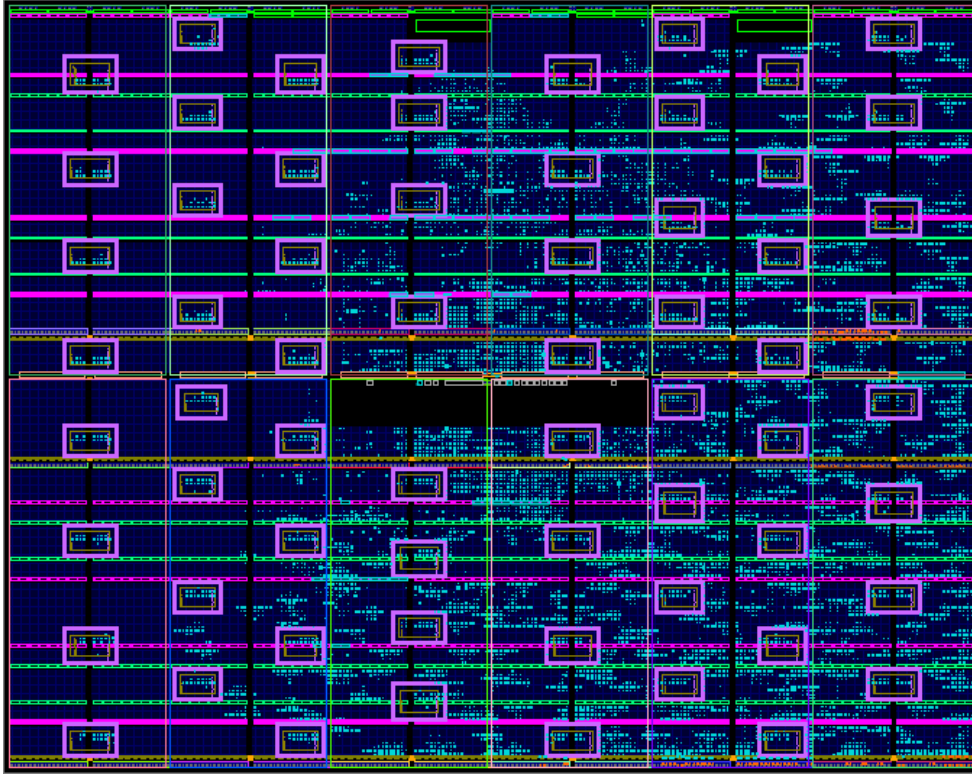


Figure 2.6: This figure shows the floorplan of a 64-tile counter system used for demonstrating the TMR+Spares w/Repair architecture. Purple blocks represent the reconfigurable regions of the system [10].

location, the sensor can be used to estimate the radiation flux rate, thereby allowing the configuration controller to throttle its activity accordingly. Figure 2.8 shows the cross-sectional view of the radiation sensor.

The radiation sensor assembly consists of a custom silicon strip sensor and a chain of amplifiers used to condition the analog sensor outputs into a square digital logic pulse. The radiation sensor is a silicon-based strip detector. The substrate consists of an intrinsic silicon wafer with a P-type (Boron doped) front surface and an N-type (Phosphorous doped) rear surface. These doped regions produce an inherent electric field inside of the silicon sensor. When a radiation particle penetrates the sensor, bonds between electrons and host atoms are broken. The breaking of these





Figure 2.7: This figure shows the early hardware setup used in this research. It includes the ML605 development board, a general-purpose interface board for receiving radiation sensor inputs and a parallel-to-USB adapter board for communication with a host computer [10].

bonds produces free electrons inside the substrate. The movement of these electrons effectively produces two types of charge carriers. The electrons themselves are the first carrier. The second carrier is represented by the void left by a traveling electron and is known as a hole. The combination of the traveling electrons and holes produces the desired signals. Once these carriers are generated, they are separated by the internal electric field inside the sensor. The electrons are pushed to the rear of the sensor while the holes move towards the front. These transient signals are then collected from the front and rear aluminum electrodes. The signals are input into a two-amplifier chain which amplifies and stretches the pulse for input into the high-speed sampler located in the Spartan-6. The high-speed sampler is a rising-edge triggered system which

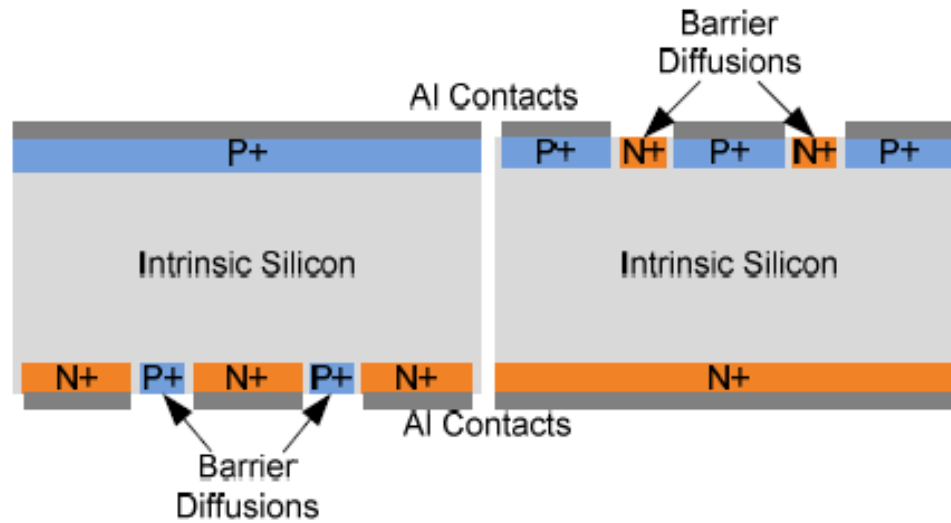


Figure 2.8: This figure shows a cross-section diagram of the stripped silicon radiation sensor. The diagram on the left is rotated 90 degrees to the one on the right to show both the top- and back-side channels.

functions as a counter for each of the radiation sensor channels. Figure 2.9 shows the radiation sensor used in this research. More information regarding the design and performance characteristics are available in [48, 49].

### New Research Hardware

The previous research systems presented were implemented on, or interfaced with, commercial FPGA development boards. FPGA development boards are designed to demonstrate the majority of features available on a target device and commonly include a wide variety of interface options e.g. general purpose input/output ports, memory card interfaces, pushbuttons, indicator LEDs, serial ports, USB ports, etc. These boards are useful because they come equipped with every feature that could possibly be necessary when designing a system, and many features that are not necessary. Early TMR+Spares systems were implemented on the Virtex-5 ML505 and



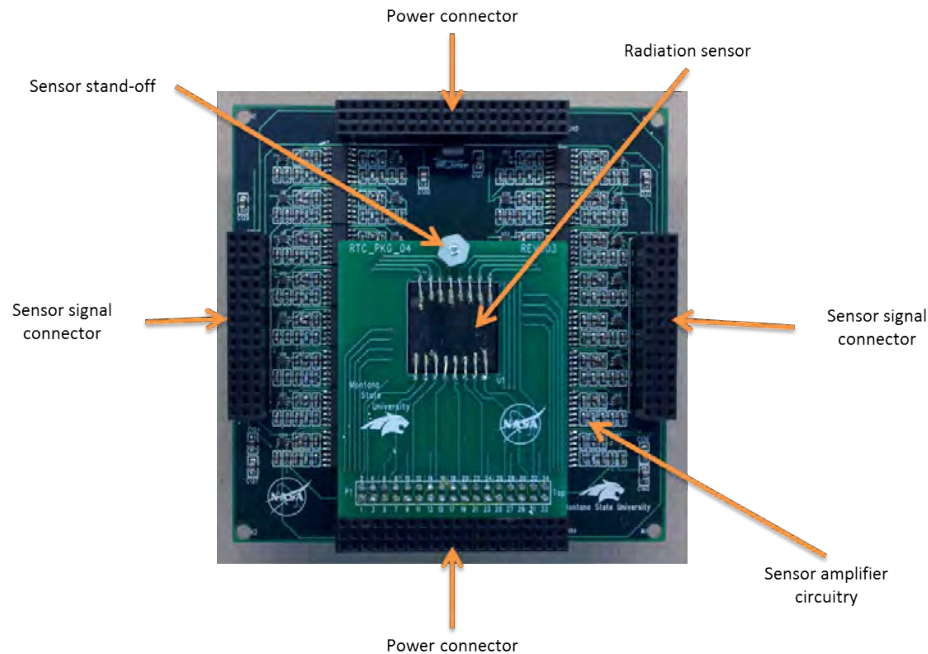


Figure 2.9: This figure shows a custom radiation sensor designed and built at MSU mounted to its accompanying signal conditioning circuit board. This stripped silicon sensor provides 16 front-side channels and 16-back-side channels perpendicularly arranged to give 256 pixels for spatial strike information.

Virtex-6 ML605 development boards. These boards were great for desktop development, but as the research advanced efforts began to flight test the systems in representative radiation environments. The desire and need to flight test necessitated the development of custom hardware in order to meet the electrical and mechanical interface requirements of available flight platforms.

1U CubeSat Stack: With space being the target environment it was natural to choose a form factor for the hardware that would position the research well for space flight consideration. Given the popularity and launch opportunities associated with CubeSat projects the 1U cube was chosen as the design goal. A vertically integrated printed circuit board stack was conceived and built. The stack contains a separate

board for each subsystem. When stacked, the structure is approximately 4"x4"x4" and it consists of a power supply board, an FPGA board, an experiment board, and up to two silicon radiation sensor amplifier boards. The power board is responsible for accepting external DC power and efficiently converting it to the many voltage rails required by the other boards in the stack. The FPGA board serves as the primary science experiment as it houses the radiation tolerant computer architectures under test. It contains two FPGAs: a high-performance Virtex-6, termed the "main FPGA", and a Spartan-6, termed the "control FPGA". The control FPGA is responsible for high-level system tasks including external communication interfacing, configuration control of the main FPGA, interfacing with the radiation sensor(s), and other general system tasks. The radiation sensor amplifier boards, known simply as "amp boards" contain the signal conditioning circuits responsible for converting miniscule current pulses generated by the radiation sensors into digital signals compatible with the control FPGA. Each sensor board is coupled with a single silicon radiation sensor through a rectangular board-to-board connector. Provisions were made to allow an experiment board to be included in the stack with the idea that the FPGA board is made available as a radiation tolerant computing resource usable by the experiment card. This structure allows custom experiments to gather data and use the main FPGA to perform any required computation.

Figure 2.10 shows a picture of the completed stack. The details of the radiation sensor amplifier board and power board are discussed in [48–50] and [51] respectively. The balance of this chapter details the design of the custom FPGA board.

FPGA Board: The FPGA board is the hardware upon which all our research systems are implemented. On development boards it was impossible to separate the

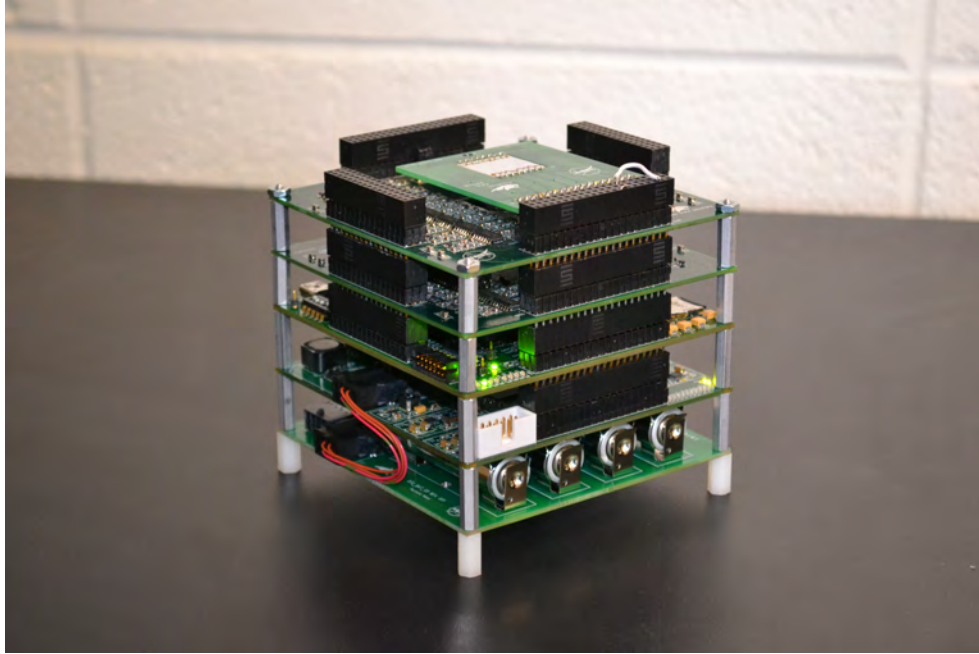


Figure 2.10: This figure shows the complete 1U form factor research hardware stack consisting of a battery board, power board, FPGA board and two radiation sensor boards.

system control functionality from the high performance computation hardware as there was only one FPGA on the board and no peripheral microprocessor devices. The ultimate goal was to provide high-performance processing capability without the use of any specifically radiation hardened components. The architecture to achieve this goal required that the FPGA configuration data be externally accessible and runtime programmable.

Much consideration went into choosing the most appropriate devices for use on the FPGA board. At the beginning of the design phase, the ML605 development board, which features a Xilinx Virtex-6 device, was being used to implement research designs. Though the Virtex-7 had recently been released, the Virtex-6 was considered to be a lower risk choice as the necessary design tools were already in place, were known to work properly, and porting designs between separate device families was not required.

The design tools also supported the requisite active partial reconfiguration that would be needed to fully implement the TMR+Spares architecture. The Spartan-6 was chosen somewhat arbitrarily. From a marketing perspective, the Spartan-6 represents a more economical FPGA solution. The Spartan device family has fewer advanced features and is generally targeted for lower performance applications than the Virtex device family. From a conceptual standpoint, the control device is envisioned as a comparatively slow component relative to the Virtex FPGA, perhaps even a simple microcontroller used only to maintain Virtex configuration integrity. The purpose for implementing system control on a slower, older technology is to attempt to reduce the susceptibility to SEUs. It is acknowledged that the Spartan-6 is neither a slow or old technology, therefore SEUs must be mitigated in the control FPGA on this particular system. Xilinx has some device features designed to help mitigate SEUs in the configuration memory, including error detection and correction capabilities. Using the Spartan-6 as the system controller allows a direct comparison of industry provided mitigation tools and radiation tolerant research architectures implemented on the Virtex. Figure 2.11 shows the custom FPGA board designed and built specifically for this research.

Device Configuration: The Spartan-6 and Virtex-6 each support a number of different configuration interfaces. Available interfaces include Master Serial, Master SPI, Master BPI-Up, Master-BPI-Down, Master SelectMAP, JTAG, Slave SelectMAP and Slave Serial. The configuration interfaces can be grouped into master and slave techniques. In master configuration interfaces, the FPGA acts as a master to a slave peripheral containing the configuration data. In slave configuration modes, the FPGA acts as a slave to a master device, which controls the configuration process. Within

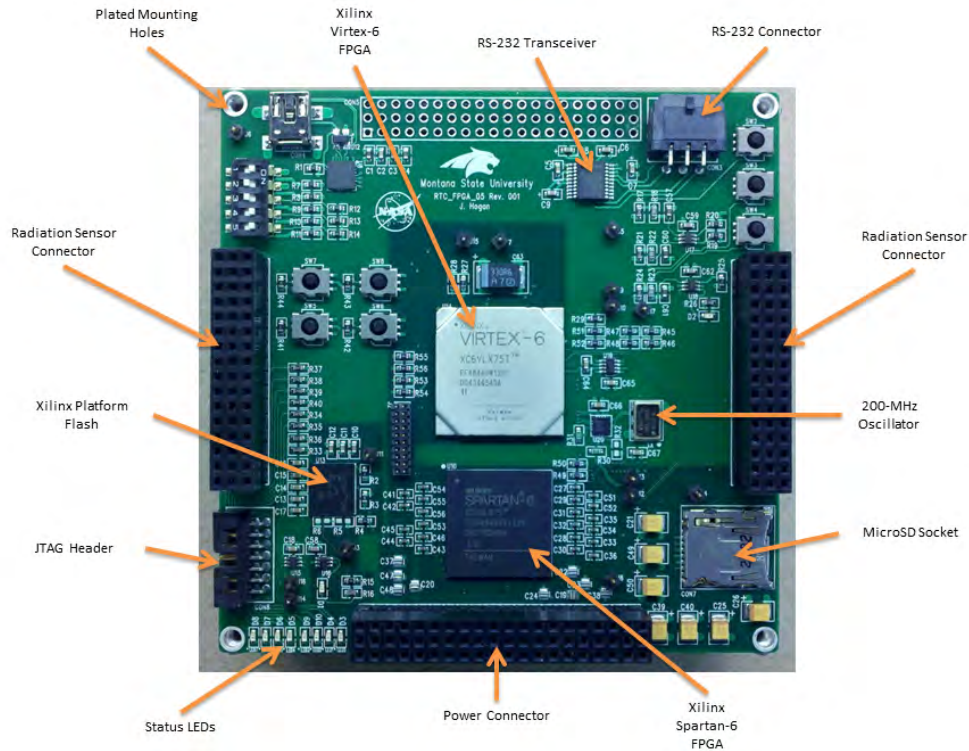


Figure 2.11: This figure shows the custom FPGA board that was designed and developed for this research. It features Xilinx Virtex-6 and Spartan-6 FPGAs. External communication is available through USB and RS-232 interfaces, and local data storage is available on a MicroSD card.

each of these groups there are serial and parallel interface options that can be used. As is commonly the case, parallel interfaces are able to complete the configuration process in fewer clock cycles than serial interfaces. However, serial interfaces require fewer signal lines between the FPGA and the configuration data source. Each type of configuration interface has distinct advantages and disadvantages.

On the FPGA board each device uses a different configuration interface. As the system controller, the Spartan-6 was designed to use a master configuration mode, which allows it to automatically configure upon application of system power. As the boot time of the system was not considered a critical design parameter, and

the simplest implementation was desired, the Master Serial interface was chosen for configuring the Spartan-6. The configuration data for the Spartan-6 is stored on a Xilinx Platform Flash device, which is specifically designed for use as configuration memory storage. The device used was the XCF32P, which has a volume of 32-Mbits. This is sufficiently large to store multiple bitstreams for the Spartan-6. In the Master Serial mode, the Spartan generates the configuration clock to the platform flash. In response to this clock, the platform flash serially transmits configuration data to the FPGA. The speed of configuration is limited by the maximum clock rate which is 30 MHz for the -1L speed grade [47]. Figure 2.12 shows the Master Serial configuration interface used by the Spartan.

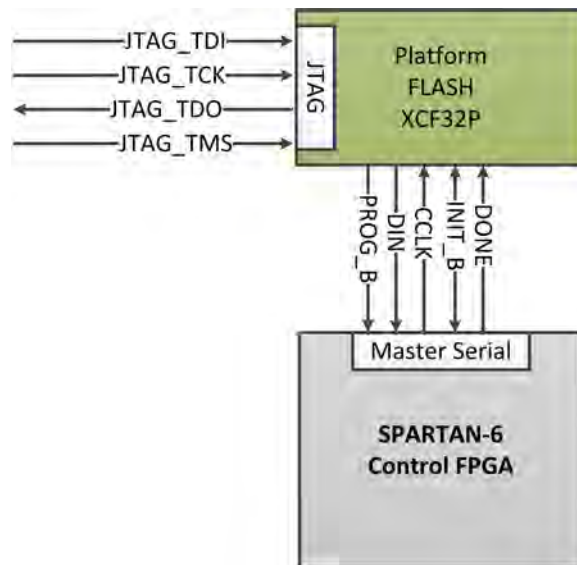


Figure 2.12: This figure shows the Master Serial configuration interface used by the Spartan FPGA.

The Virtex uses an 8-bit Slave SelectMAP interface for its configuration. In this setup, the Spartan acts as the master device as it generates the configuration clock and transmits configuration data to the Virtex. The Spartan has access to all con-

figuration port signals on the Virtex, which allows continual control over the device configuration after initial configuration is complete. This external access to the configuration interface for the Virtex is perhaps the single most important feature of the FPGA board. A MicroSD card, which is accessible to the Spartan, contains all full and partial bitstreams used by the Virtex. Separate custom hardware logic cores are used to control the retrieval of data from the SD card and its subsequent transmission to the Virtex. Figure 2.13 shows the configuration interface for the Virtex-6.

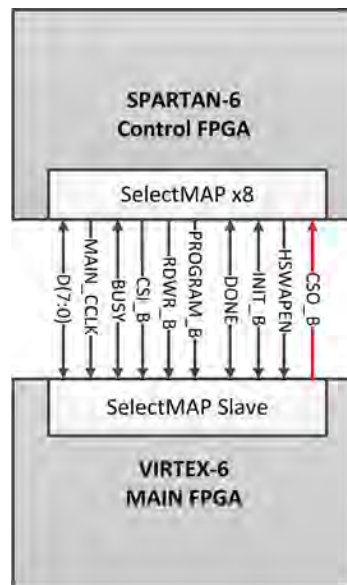


Figure 2.13: This figure shows the Slave SelectMAP x8 configuration interface used by the Virtex FPGA.

MicroSD Card Interface: A serial peripheral interface (SPI) protocol is used to communicate with the SD card. A finite-state machine (FSM) was designed to control data transfers between the SD card and the Spartan. The SD card is controlled through the issuance of command packets. A response type is associated with each command and is transmitted to the host upon completion of command processing.

The response typically indicates the status of the device, whether the command was successfully completed, or any relevant error occurrences. In order to operate the SD card in SPI mode a specific initialization command sequence must be issued. This sequence, detailed in [11], is shown in Figure 2.14. During initialization, the SD card is clocked at 400-kHz.

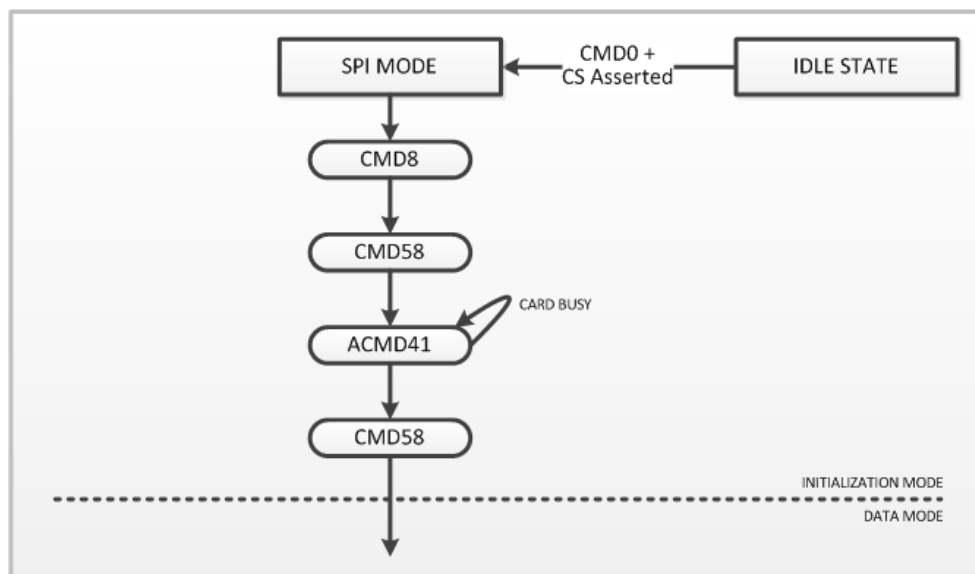


Figure 2.14: This figure shows the command sequence used for initializing a MicroSD card for SPI-mode operation. In practice, the card issues responses to each command, which must be interpreted to ensure proper initialization [11].

There are three types of SD cards: standard capacity, high capacity and extended capacity. These cards can store up to 2-GB, 32-GB and 2-TB respectively. Depending on the operating mode, data rates of up to 50-MB per second can be achieved. In SPI-mode, however, the maximum clock rate is 25-MHz, limiting the bandwidth to approximately 25-Mbits per second. The data rate is not quite 25-Mbits per second on account of a comparatively small number of overhead bits including commands, block start and stop indications, and data block checksum bits. On standard capacity



cards, the size of the data blocks is configurable between one and 512 bytes. In this application the block size was set to 512-bytes to maximize the data transfer rate.

There are two types of read operations that can be performed: single-block and multiple-block read. In both cases the block address for the data is transmitted to the SD card. In a single-block read, a response immediately follows issuance of the read command followed by the requested data block. The data block is suffixed by a 16-bit checksum, which is optionally ignored. After 512-bytes are clocked out of the SD card the card automatically returns to an idle state where it awaits subsequent commands. In a multiple-block read, a command is issued to initiate the read process. As in the case of a single-block read command, the card responds to a multiple-block read command with a status response followed by a sequence of data blocks beginning at the requested block address. The block address automatically increments and the data transfer continues until a command is issued to stop the process.

The process of writing to the SD card is similar to the read process. A command to write either single or multiple blocks to the card is followed by transmission of the data from the host to the card. Cyclic redundancy checks can be performed on the transferred data, though that is an optional feature which has been disabled in the applications described here. The ability to write data to the SD card was necessary for flights aboard vehicles lacking a telemetry stream.

### System Testing

As the technology has matured testing of the system has occurred at each incremental step. Early versions of the radiation sensor were tested in an electrical breadboard with LEDs attached to each channel to demonstrate functionality. Simple red and near-infrared laser pointers have been used to stimulate sensors via the

photoelectric effect as the photons create electron-hole pairs in the semiconductor material. As the hardware systems progressed so too did the testing. A more sophisticated pulsed-laser system was built to target individual radiation sensor channels. This provided confidence that the sensor was ready to be tested in a cyclotron to show a response to actual ionizing radiation. In the push toward eventual testing in a space environment, flight testing of the hardware began with local sounding balloon flights before progressing to participation in a long-duration, high-altitude scientific balloon flight. The following sections describe the variety of tests the hardware has undergone and some of the results that were produced.

Near-Infrared Pulsed Laser: Testing the functionality of the radiation sensors was a major priority following the fabrication, and prior to traveling to the cyclotron facility for beam testing. This testing was performed using a pulsed-laser system to stimulate radiation sensor channels individually. The goals of the tests were to both identify faulty channels, to demonstrate the basic functionality of the sensor, and to test the sensor/FPGA interface. The ability to stimulate the sensor and read the data coming into the FPGA closed the loop between the sensor and the computer system, and readied the system for subsequent cyclotron and flight testing.

Though it was designed for radiation, the sensor is also able to be stimulated by concentrated optical radiation via the photoelectric effect. As a rough functionality test, the sensor was “firehosed” with red and near-infrared laser pointers by flashing the laser across the sensor. Monitoring of a user interface displaying the sensor data showed the stimulation of the sensor. Light around 630 nanometers does not deeply penetrate the sensor, so, in general, only the front-side channels responded to the red laser. Silicon is quite transparent at near-infrared wavelengths, so the 980 nanometer

laser was able to penetrate deeper into the sensor and stimulate both front and back side channels. Figure 2.16 shows the penetration depth of a variety of wavelengths considered for use in this test system. Figure 2.15 shows the sensor installed in an electronics breadboard and stimulated by a red laser pointer to test the functionality.

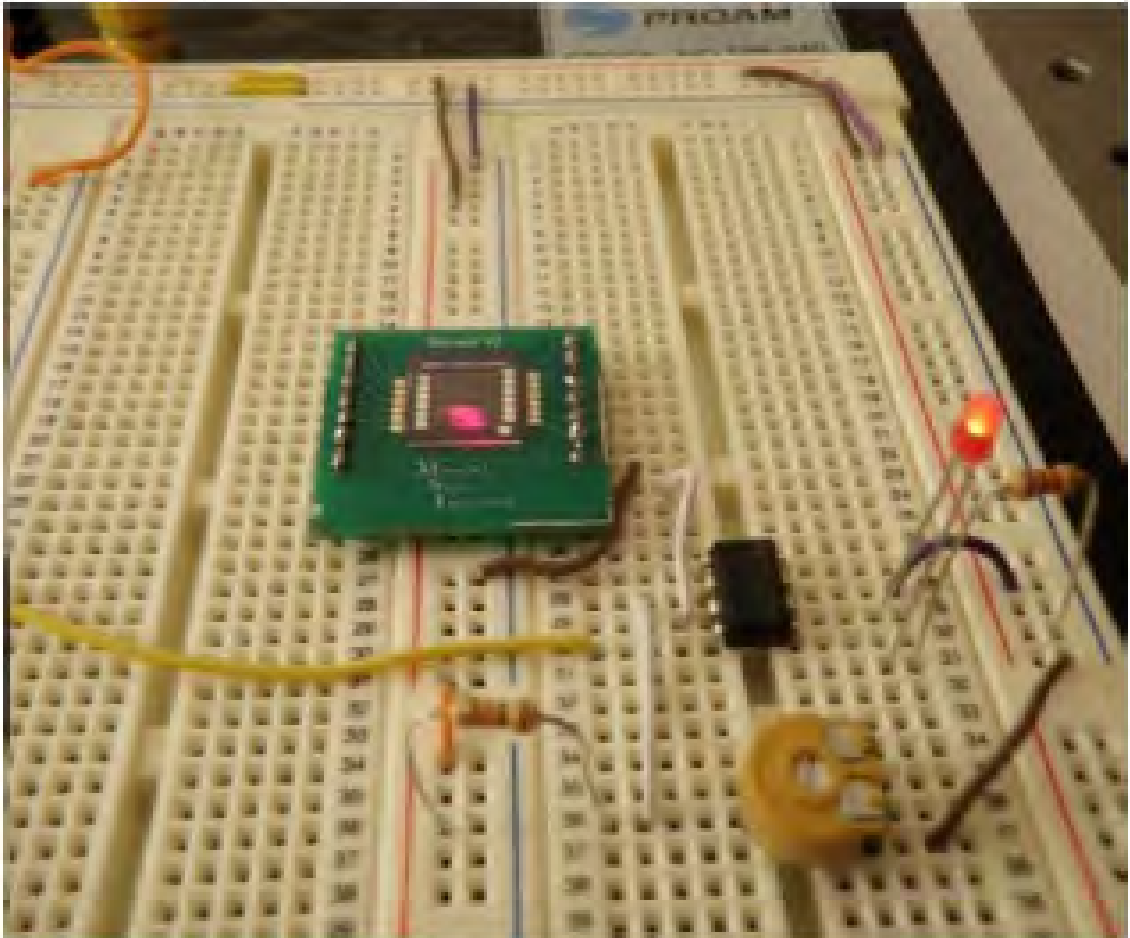


Figure 2.15: This figure shows the breadboard test setup used for early functionality testing of the radiation sensor.

The tests using laser pointers were very coarse functionality tests, and did not provide any insight into the response to lower energy deposition as would be encountered at the cyclotron. Additionally, spatial isolation of sensor channels was not possible

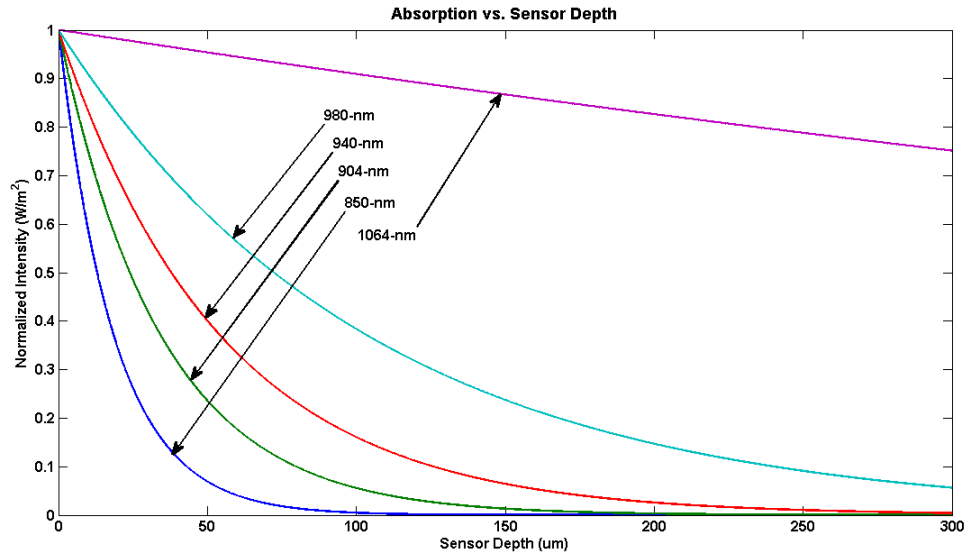


Figure 2.16: This figure shows the attenuation of laser light as a function of depth in silicon material.

using laser pointers. To better simulate the energy levels of the cyclotron, a pulsed laser test setup was constructed. This setup was used to generate low energy, short duration pulses of highly focused near-infrared laser light. Figure 2.17 shows a block diagram of the pulsed laser system.

A near-infrared semiconductor diode laser was the optical source for the experiment. The laser diode was fiber-coupled to a collimation optic, which created a small, uniform spot at the input to the optical system. The collimated laser light was focused through an acousto-optic modulator (AOM). The AOM used RF input energy to diffract the incident laser beam to a variety of angles. This diffraction resulted in multiple beams diverging from the output of the AOM at different angles. When the RF energy was disabled, no diffraction was induced by the AOM and all incident energy was present in the first-order beam passing through the device. When RF energy was present in the first-order beam passing through the device. When RF energy was enabled, a diffraction gradient was set up in the AOM causing multiple beams to diverge from the output of the AOM. The strongest of these was the second-order

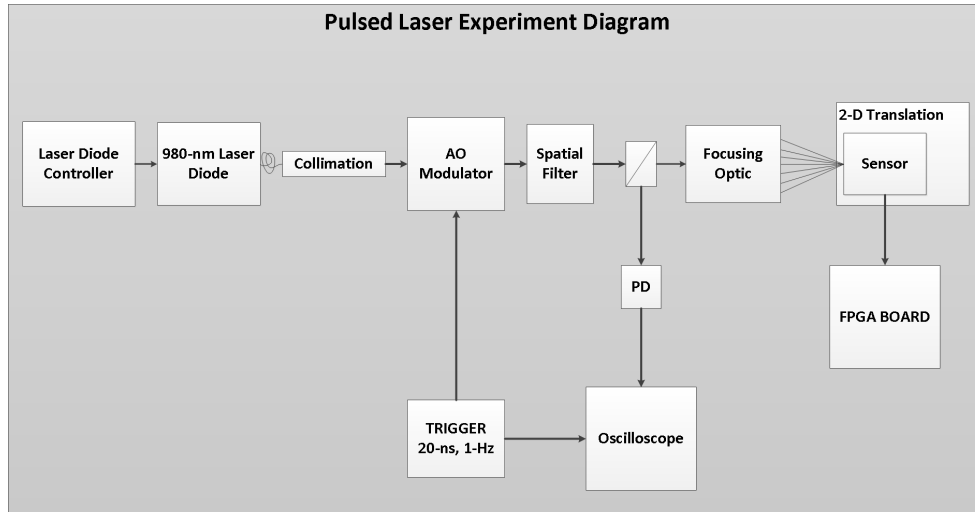


Figure 2.17: This figure shows a block diagram representing the test setup used to optically stimulate the radiation sensor. This testing was used to demonstrate sensor functionality prior to testing at the cyclotron facility.

diverging beam. This beam was spatially filtered exclusively allowing it to propagate through the remaining optics and to the radiation sensor. All other beam orders, including the primary beam, were blocked by an adjustable iris. The generation of the pulse incident on the radiation sensor was achieved by enabling/disabling the RF signal input into the AOM. When enabled, the second-order beam passed through the spatial filter and on to the sensor. When disabled, all the optical energy was in the first-order beam, which was blocked by the spatial filter. The width of the enable pulse approximately determined the width of the optical pulse. The enable pulse was a TTL signal generated by an external FPGA design, which was clocked at 50 MHz. This resulted in a 20 nanosecond laser pulse. The final optical stage used a microscope objective lens to focus the laser pulse to a very small spot size, smaller than the 100 micron gap between adjacent channels. A spot size smaller than the inter-channel gap allowed the light to penetrate the sensor rather than being reflected by the aluminum layer on each channel. The wavelength of the laser was

chosen based on a handful of considerations. Included in these considerations were the photon energy at the chosen wavelength, which was desired to be at or above the band gap energy of the sensor's doped silicon, the penetration depth in silicon, and the minimum spot size at the focal plane. Adequate photon energy ensures that enough electron/hole pairs are generated in the sensor to register a response. The bandgap energy of the sensor silicon was 1.1 eV, so the photon energy was required to be higher than that value in order to generate adequate charge. The photon energy versus wavelength was calculated using Equation 2.1. This is shown in Figure 2.18. At 980 nanometers, the photon energy is approximately 1.26 eV.

$$E = \frac{1}{1.602 \cdot 10^{-19}} \cdot \frac{h \cdot c}{\lambda} \quad (2.1)$$

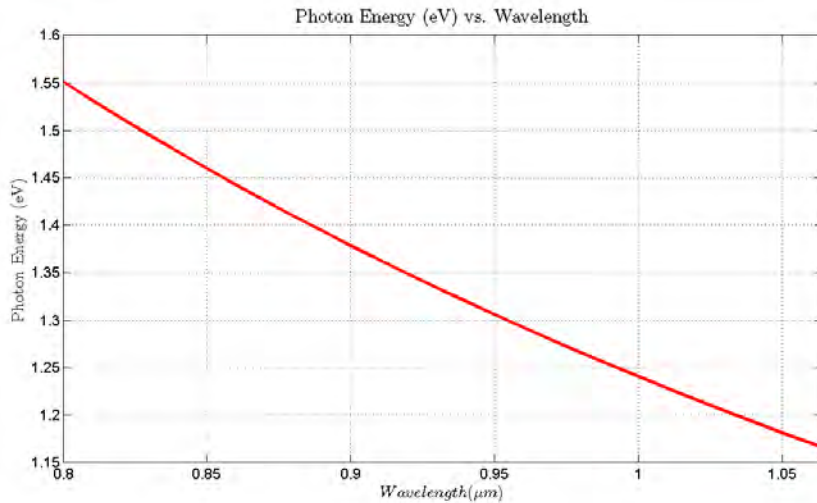


Figure 2.18: This figure shows the photon energy in electron-volts ( $1\text{eV} = 1.602 \cdot 10^{-19}\text{J}$ ) for laser wavelengths between 0.8 and 1 micron.

The primary factor in determining the minimum pulse width was the laser spot size at the AOM. The RF energy must be enabled long enough for the electromagnetic wave to propagate across the full width of the laser spot. This propagation time is

determined by the velocity of the wave in the AOM crystal, which was specified at  $3.63\text{mm}\cdot\mu\text{m}^{-1}$ . A 200 millimeter focal length lens was used to focus the laser through the AOM. This resulted in a 0.073 millimeter minimum spot size, and subsequently a 20 nanosecond minimum pulse width. This pulse width value was convenient as it matched the clock period on the FPGA board generating the AOM enable pulse.

The sensor itself was mounted to a two-axis translation stage. This stage was manually controlled using high-precision positioning micrometers. The translation stage had adequate range of motion to test each of the individual sensor channels. Movement of the sensor was equivalent to moving the laser to different parts of the sensor. This test setup was successful in demonstrating the sensitivity of the radiation sensor to relatively low pulses of input energy. This provided a high degree of confidence that the sensor would also respond to ion testing at the cyclotron facility. These tests were also able to reveal sensor channels that were defective as a result of the manufacturing process. The spatial sensitivity was demonstrated by translating the sensor across the laser focal point, stimulating each channel individually. Detailed analysis of radiation sensor testing is available in [49].

Cyclotron Testing of the Radiation Sensor: There are two techniques for testing radiation tolerant computer architectures terrestrially: software fault injection and radiation testing using a particle accelerator. As a component in the radiation tolerant computer system, the custom silicon radiation sensor warranted its own set of tests to demonstrate sensitivity to ionizing radiation, spatial sensitivity and incorporation of strike location information as feedback to the computer system. Acting as feedback in this capacity imparts a degree of awareness of the radiation environment to the computer system. This information can be used in several ways by the

configuration memory scrubbing system. It was anticipated that the spatial strike information would be used to direct the scrubber to the area of the FPGA most likely affected by a radiation strike. This would minimize the repair latency resulting in shorter fault duration and recovery times. Use of the sensor in this capacity is most effective when the sensor itself is coupled closely to the FPGA silicon substrate. As the sensor is moved vertically above the FPGA itself, up to an inch or so in the current stack configuration, there is only a narrow cone of acceptance within which incident particles will strike both the radiation sensor and the FPGA. Outside of this cone particles with greater angles-of-incidence can strike the FPGA without stimulating the sensor. This significantly reduces the effectiveness of coupling the sensor on the FPGA stack. Regardless, an important part of this research was to conduct testing of the radiation sensor and demonstrate the ability of the computer system to interpret and respond to incoming strike information.

To test the sensor, several trips were made to the cyclotron at the Texas A&M Radiation Effects Facility in College Station, TX. This facility is widely used by the aerospace electronics industry for single event effects testing of electronic components. The cyclotron offers a choice of several beams of varying energies including 15, 25 and 40 MeV. At each energy a number of particle species are available. In addition to different beam energies, the particle species can be chosen by the user to meet the experiment objectives. In the final trip to the cyclotron in April of 2013 the complete computer stack including FPGA board, power board and radiation sensor board was tested for the first time in the beam. This test was successful in demonstrating radiation sensor functionality and computer response to the radiation sensor information.

The beam selected for use in the tests was a 25 MeV Krypton beam. This beam was sufficiently energetic to penetrate deeply into the sensor's silicon substrate, al-



lowing intersection strikes to be registered. The facility provides the beam as an uninterrupted stream of particles. The fluence, or number of particles incident on the test sample, is used to determine each experiment run time. The particle flux is directly measured and integrated over time to provide an estimate of fluence. Upon reaching the specified number of particles, the experiment concludes and the beam is turned off. This capability is important in experiments seeking to measure radiation effects as function of radiation dose. In this particular experiment it was used as a convenient way to separate runs, allowing periodic access to the electronics stack during the testing.

On the FPGA, the interface to the radiation sensor is an edge-triggered counter. Each rising edge on a sensor channel increments a corresponding count value allowing the FPGA to track an approximate number of radiation strikes. For the continuous beam the flux was high enough that the activated sensor channels would stay in a steady state as long as the beam was present. In order to generate periodic sensor inputs it was required that the beam be pulsed. Though the beam itself could be briefly diverted then restored at a 1 Hertz rate it was found that doing so often caused the beam to be taken offline and subsequently re-tuned. This time consuming process used large chunks of the allotted 8 hour time slot. Rather than diverting the beam to achieve a 1 Hertz pulse rate, a simple chopper was devised and built which intermittently interrupted the beam. A microcontroller controlled a servo motor to which a thin aluminum shield was attached. Every second the servo motor rotated the shield out of the beam path, briefly allowing the radiation to strike the sensor. This resulted in edge-triggered events every second.

In order to demonstrate the spatial resolution of the sensor, an apparatus for translating a small aperture about the sensor was also designed and built. This translation stage moved two perpendicular slots in such a way as to create a rectangular opening

at specified locations on the sensor. A single servo motor controlled the location of each slot. The slots were mounted to a rail system. This allowed the X slot to move laterally independently of the Y slot. Similarly, the Y slot moved vertically along two rails. The combination of the beam chopper and the translation mask enabled stimulation of individual sensor channels. Figure 2.19 shows the computer stack, chopper and translation stage mounted to the beam test fixture.

In this series of tests, a 9-Tile Microblaze system was implemented on the Virtex FPGA. The Spartan monitored the radiation sensor outputs and controlled the set of active processors on the Virtex. In the event of a radiation strike registering on the sensor above an active tile, the affected tile was swapped for a healthy tile, repaired, and made available as a spare. Though the tiles themselves were never actually faulted, responding as if they were demonstrated the complete integration of the radiation sensor with a partially reconfigurable, radiation tolerant computer system. The tiles were running 32-bit counter applications, and rather than copying the program processor registers and program memory into the newly activated tile, synchronization was simply accomplished by resetting all of the counters.

High-Altitude Scientific Balloon: After developing the research hardware platform, the first opportunity for flight was on a high-altitude scientific balloon. Our proposal to fly a radiation effects detection system was accepted by the High Altitude Student Platforms (HASP) program in both 2012 and 2013. This program, administered by the Louisiana Space Grant Consortium at Louisiana State University, is aimed at exposing undergraduate and graduate students to scientific ballooning through direct hands-on experience. In this program, student teams selected from universities nationwide develop scientific payloads, which are flown aboard a

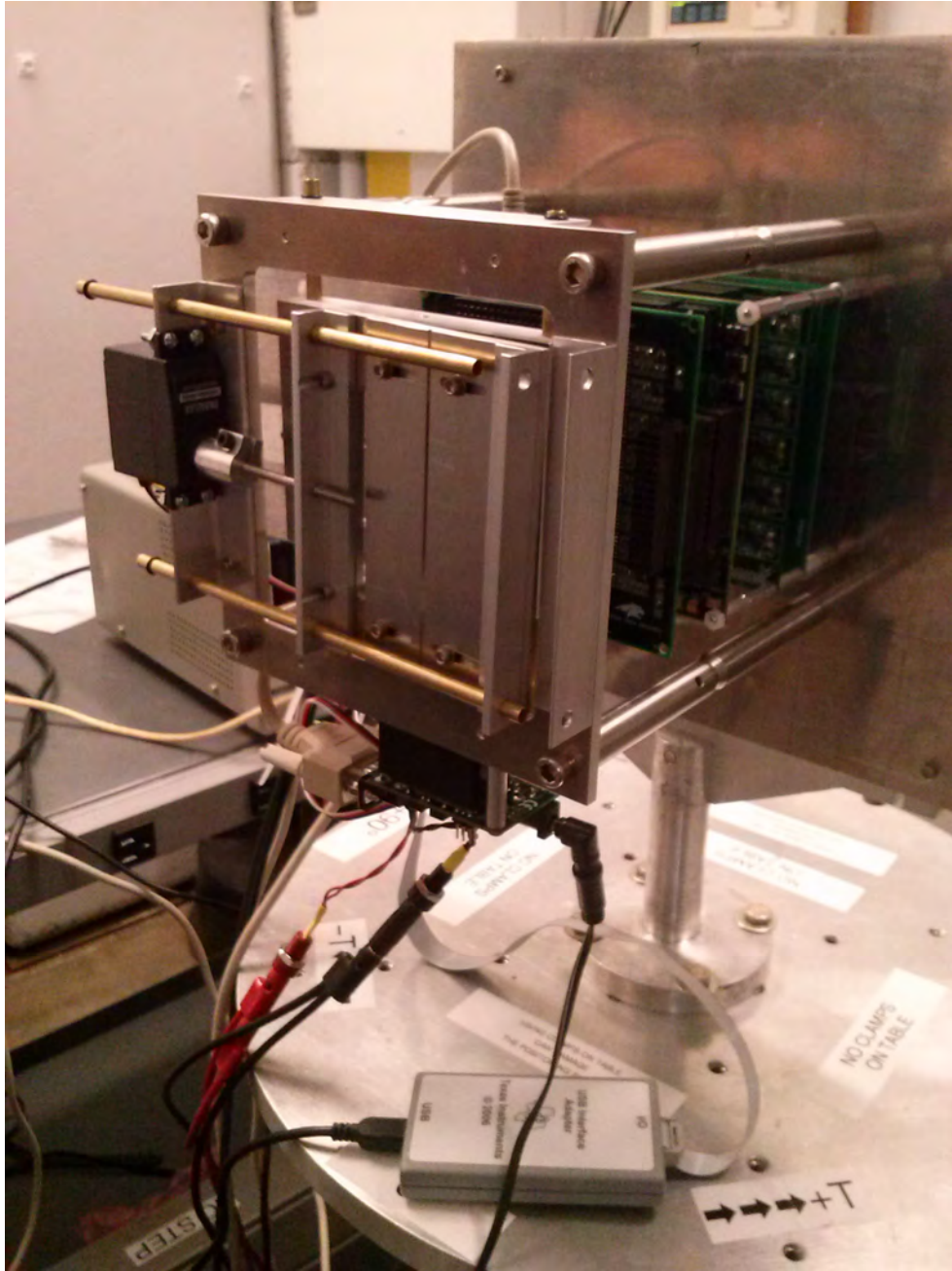


Figure 2.19: This figure shows the custom computer stack under test at the Texas A&M Radiation Effects Facility. A custom translation stage and radiation beam chopper provide spatial and temporal isolation of the beam to allow stimulation of single channels.

zero-pressure scientific balloon to approximately 120,000 feet for a duration of up to 12 hours. The HASP payload detailed here was titled “Single Event Effect Detector” (SEED) and the proposed scientific goals were to demonstrate operation of the custom MSU radiation sensor in a natural radiation environment, to measure the atmospheric neutron profile using said sensor, and to record any upsets occurring in the FPGA computer system. This program presented an opportunity not only to test our computer architectures, but also to gain experience in multiple space flight design disciplines. The mechanical design of the enclosure that would house the computer stack and attach to the HASP flight platform presented significant design challenges. These design challenges demonstrated the importance of working collaboratively in interdisciplinary teams. Through participation in the HASP program in 2012 and 2013 a couple of PCB-level design flaws were discovered, the research hardware was advanced to sub-orbital flight readiness, and extensive design experience relevant to space systems was gained. Figure 2.20 shows the MSU payload mounted to the HASP platform along with payloads from a variety of schools across the country.

Coming on the heels of the radiation sensor testing at the cyclotron, the balloon flight offered an opportunity to test the functionality of the radiation sensor in a near-space environment. The chance of observing strikes by radiation with sufficient energy to pass fully through the sensor was increased by the long flight duration and the high altitude, which carried the payload above 99.9% of the atmosphere. Particles which pass completely through the 300 micron thickness of the sensor are of particular importance because they register in the high-speed sampling circuitry as intersections of front- and back-side sensor channels. Strikes which only register on a front side channel are assumed not to have passed through the sensor, and therefore they do not pose a threat to the computer system. Penetration notwithstanding, all strikes were counted and recorded. Neutrons were expected to be the predominant particles



Figure 2.20: This figure shows the 2012 MSU HASP payload mounted to the HASP platform, which is suspended from the launch vehicle during the early stages of flight operations.

encountered as they represent the bulk of energetic particles in the atmosphere [52]. The atmospheric neutron flux is well known [53] and a major test of the sensor was to replicate the neutron flux profile during the ascent phase of the flight. The desired data products included the sensor spatial strike information, ionizing radiation strike rate, and particle flux. Due to the low expected bit upset rate on the FPGA, which was on the order of one or two upsets per day, the objective to detect single event effects within either of the two FPGAs in the payload was considered a secondary science objective.

Payload Thermal Design: On Earth computer systems are generally cooled using large heatsinks attached directly to thermal generation sources using a thermal grease, which creates a low thermal resistance path away from the sensitive compo-

ment. A fan attached to the heatsink increases the rate at which heat is moved out of the system. This convective heat transfer process is reliant on the presence of an atmosphere for effectiveness. In extremely low atmospheric pressure environments, the ability to convectively cool is lost. Therefore, conductive and radiative processes must be used to move heat away from electronics. The foremost engineering objective was to determine the thermal behavior of the electronics in a low-pressure environment. The fact that payload electronics were under development for the majority of the project life precluded early thermal testing to see if the system would overheat, or cool excessively during the ascent and float phases of the flight. Simulations were used to provide estimates of the thermal behavior of the system. These models included finite element analysis (FEA) of the combined electronics, mounting hardware, and enclosure as well as component-level models derived from the PCB layout software. Initial FEA simulations indicated that the payload would easily exceed the 100 °C maximum operating temperature of the FPGAs. These models informed the design of the payload enclosure and motivated several important PCB design decisions. The payload enclosure was designed with the main goal of maintaining FPGA core temperatures within their specified operating ranges. A major milestone required before flight was successful completion of a thermal-vacuum test to demonstrate functionality over the anticipated temperature and pressure ranges for the flight. Pressure inside the chamber was reduced to approximately 5 millibars to simulate the low pressure environment of near-space. Thermal stress tests were performed at -40 Celsius and +50 Celsius with a soak of approximately one hour at each temperature extreme. Figure 2.21 shows the HASP platform with payloads attached inside the environmental chamber prior to testing.

The enclosure was designed for thermal and mechanical protection of the payload electronics. The approach to thermal protection sought to maximize reflection of solar

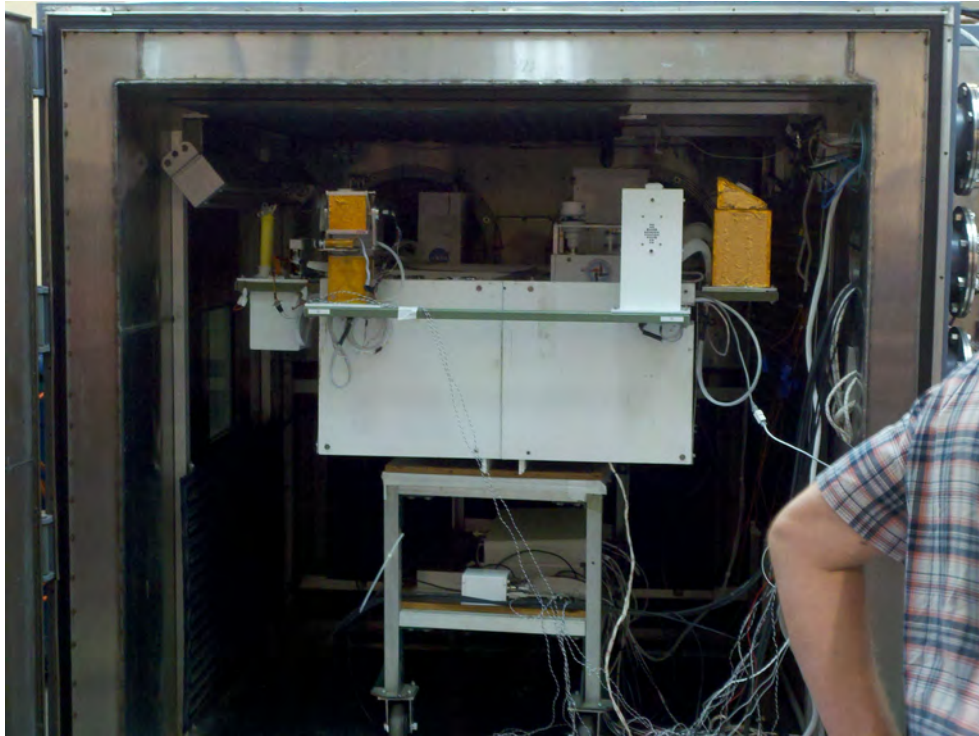


Figure 2.21: This figure shows the HASP platform with payloads attached during the environmental testing phase of the payload integration operations at the NASA Columbia Scientific Balloon Facility in Palestine, TX.

irradiation using a thin layer of aluminum and a matte finish, high-emissivity white paint. Beneath the aluminum layer was a half-inch of insulating foam material, which minimized heat transfer between the enclosure and the outside environment. This prevented excessive heat loss during the ascent phase, and heat absorption during the float phase of the flight. Within the enclosure, the heat generated by the electronics was conducted away from sensitive components through PCB ground planes, through the aluminum support stand-offs, and into a one-eighth-inch thick copper plate acting as a heatsink. The heatsink was placed inside the enclosure, beneath a piece of insulating foam to prevent internal radiative heat transfer. The bottom of the heatsink was in contact with the PVC mounting plate. Though not an excellent thermal



conductor, it was expected that this configuration would heat the mounting plate allowing a moderate amount of heat to be radiated from the payload toward the earth. Figure 2.22 shows the FEA simulation for the FPGA board inside the enclosure. It was predicted that the FPGA temperature would stabilize around  $50^{\circ}\text{C}$  during the float phase. This result was validated in the recorded telemetry data. The temperature data recorded during the 2012 flight is shown in Figure 2.23. Similar results were acquired on the 2013 flight.

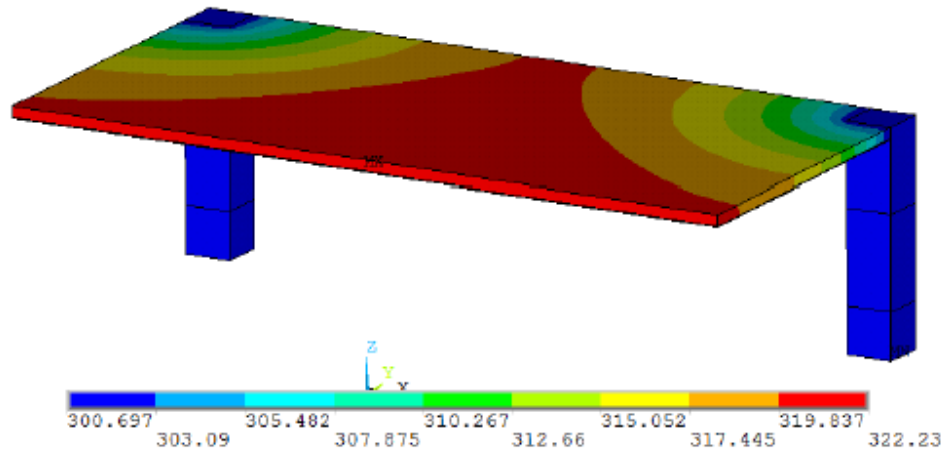


Figure 2.22: This figure shows the predicted steady-state system temperatures for the HASP 2012 payload [12]. The predicted system temperature during the float phase of the flight was approximately  $50^{\circ}\text{C}$ . An error of 0.23% was observed between the predicted and actual values [12].

Payload Control and Operation: The Spartan FPGA acted as the system controller as it housed the high-speed sampler for the radiation sensor, controlled the communication between the payload and the HASP platform, and controlled the configuration and operation of the Virtex FPGA. Since the majority of development time leading up to the 2012 flight was devoted to designing the requisite circuit boards, a complete radiation-tolerant architecture was not implemented for testing



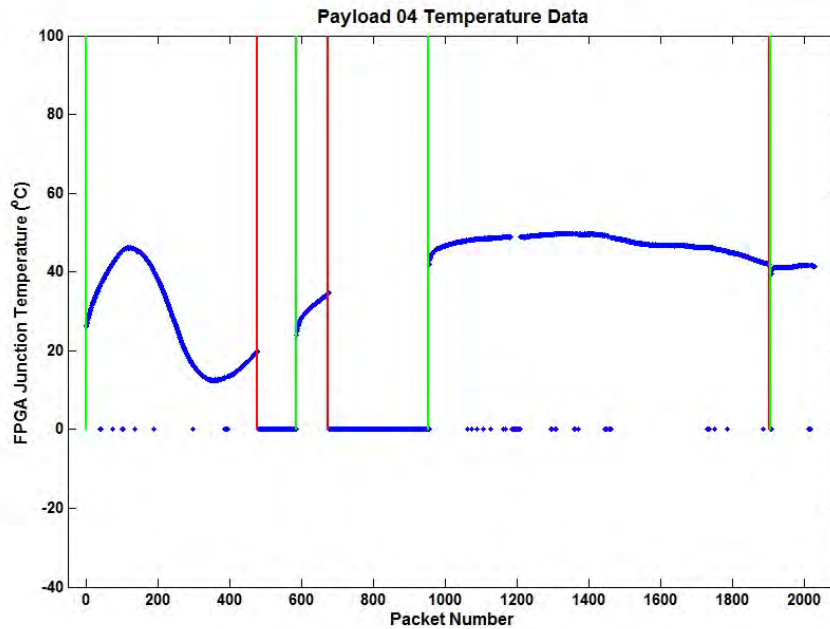


Figure 2.23: This figure shows the measured steady-state system temperature measured during the HASP 2012 flight.

on this flight. Instead, a detection system was implemented which allowed SEUs and SEFIs to be identified and avoided, but not repaired. This was viewed as an initial step toward the implementation of a fully-functional radiation-tolerant system. The Spartan device used a dedicated, internal CRC hardware component on the configuration memory, which was input into the control microprocessor as an interrupt to indicate fault occurrence. The SEE detection strategy for the Virtex used an array of 16 Microblaze processors, three of which were active in a TMR implementation. The outputs of the processors were sent to a majority voter, which determined if any of the active processors were faulted. In the event of a fault, affected processors would be replaced with one of the available spares. Signals representing the set of active processors were sent to the Spartan. Single event effects could thus be observed by the Spartan through changes in the active processor set.

The payload was designed for autonomous flight operation with minimal administrative commands. At power-on, the system performed an initialization sequence during which the FPGAs were configured, the control processor was booted, and the data storage structures were initialized. A Microblaze soft processor was used to control the system. Its operation was interrupt-driven as it handled receipt of commands and GPS data from the HASP platform, and transmitted raw telemetry data at each expiry of a 20 second fixed-interval timer. Transmitted data included a system counter, which served as a heartbeat to show that the system was running, the number of cumulative counts observed on each channel of the radiation sensor, the junction temperature of the Virtex, GPS time and position data, single event effects data, and system status flags.

Data Visualization: As the data became available during the flight it was processed and viewed in a custom MATLAB telemetry GUI. This GUI provided a graphical display of the radiation sensor including the number of cumulative strikes at each of the 256 channel intersections. In addition to the radiation sensor data, the GUI displayed the contents of the most recent telemetry packet. This included the system heartbeat counter, UTC time, latitude, longitude, altitude, GPS fix status, a payload start status word, the set of active processors and the Virtex junction temperature. The data were retrieved from the HASP website as they became available during the flight. After retrieval, the data were processed in MATLAB and the contents of each telemetry packet were displayed on the graphical user interface. This gave the team the ability to scroll through all the received packets to determine how the system was operating. Commands to reset the radiation sensor counters and to reconfigure the Virtex were available to the team during the flight. During flight operations the

team's job was to ensure that the payload was transmitting data as expected. If the data transmission ceased, as it did twice during the 2012 flight, a power cycle would be requested to re-start the payload. Figure 2.24 shows the user interface used to monitor the payload telemetry stream during the flight operations.

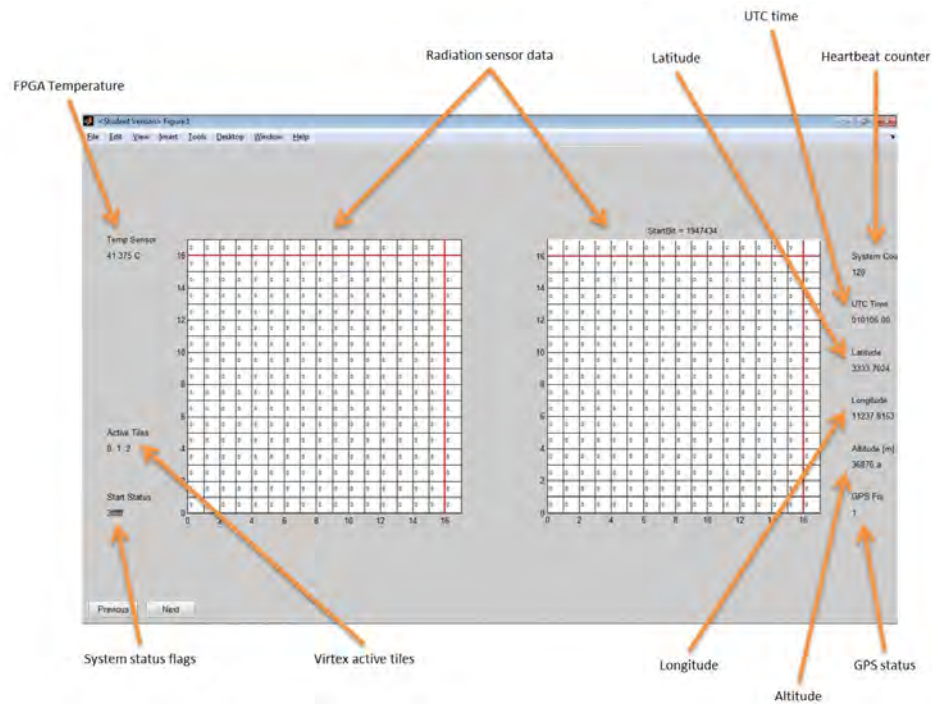


Figure 2.24: This figure shows the graphical user interface used to parse downlink telemetry packets during HASP flight operations.

Payload Mechanical Design: In addition to the science objectives there was a handful of engineering objectives to be accomplished on the flight. Engineering objectives included demonstration of the mechanical integrity of the PCB stack and thermal survival of the low-pressure environment. Mechanical restrictions placed on the payload required that it fit within a nominally 6" by 6" footprint, be 12" or less in height, and weigh less than 3 kilograms. A one-quarter-inch thick PVC mounting plate was provided upon which the payload enclosure was mounted. The enclosure

was built using a foam insulating layer externally reinforced by a carbon fiber shell. A thin layer of aluminum was placed between the foam and the carbon fiber to protect the foam from the resin used in the fiber hardening process, and to reflect long-wave solar energy from the payload. The outside of the enclosure was painted using a high-emissivity white paint, which allowed heat to efficiently radiate from the payload. Figure 2.25 shows the payload enclosure flown on the 2012 flight.

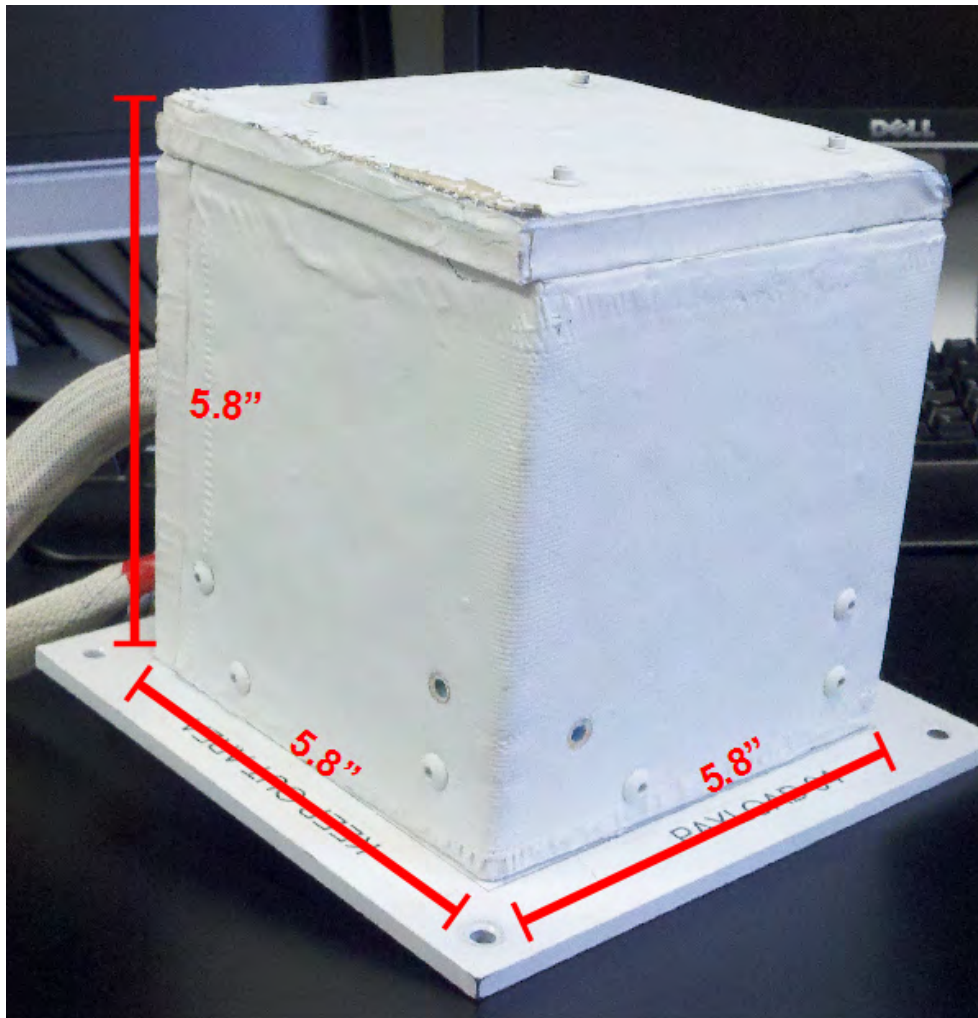


Figure 2.25: This figure shows the payload enclosure built by MSU students for flight aboard the HASP scientific balloon platform.

The payload electronics stack consisted of a power board, an FPGA board and a sensor board. The boards were vertically stacked using through-hole connectors and secured at each corner using a threaded aluminum stand-off. In addition to providing mechanical support these stand-offs were part of the payload thermal solution. The mounting holes on the FPGA and power boards were plated and connected to the internal ground planes. This allowed heat to flow through the PCB ground planes to the aluminum stand-offs. A 4"-by-4"-by-1/8" solid copper plate was placed at the bottom of the payload, secured using the stand-offs, and used as a heat sink during flight. Thermal models indicated this was a good solution for maintaining the desired operating conditions, and the payload encountered no thermal issues during payload integration or flight.

The electronics stack and enclosure were attached to the mounting plate separately. The electronics stack was secured by screws which engaged the threaded aluminum stand-offs. The enclosure was held in place using bolts and nuts which affixed internal angle brackets to the mounting plate. The enclosure survived the flight reasonably well. Damages to the payload between launch and return shipment receipt included chipped paint on the enclosure lid, a smashed corner on the lid, two missing side-panel bolts, and one sheared-off nylon stand-off, which was helping to hold the lid down. Evidence of strain was visible on the remaining 3 nylon stand-offs.

HASP 2012 Payload Failure Analysis: As the engineering of the flight hardware was not completed early enough prior to flight for extensive testing, the sensor and amplifier board were tested for functionality independently. Tests involved both the pulsed 5-mW 980-nm laser setup previously described, and exposure to high-energy ions at the Texas A&M Radiation Effects Facility. Results of these tests gave confi-

dence that the flight sensor was functioning as expected. For these initial tests, the sensor board was attached to an ML605 development board rather than the flight FPGA. These tests demonstrated that the flight sensor and amplifiers were sensitive at least down to 25 MeV. When the new flight hardware was completed it was also tested using a 630 nanometer red laser pointer and a 980 nanometer near-infrared laser pointer to stimulate the radiation sensor. These tests gave confidence that the radiation sensor was working properly with both the power board and the FPGA board integrated into a complete system.

Two operation anomalies were observed during the flight. In these events, the payload continuously streamed data beyond the typical 1224 byte packet length (up to about 8000 bytes) before becoming unresponsive. These events were investigated during post flight testing of the hardware to determine what caused the system to crash unexpectedly. The pulsed laser setup was used to perform these tests. This round of testing included the entire electronics stack used during flight. The sensor was stimulated with pulses of varying width at a 1-Hz repetition rate. The failure behavior was successfully replicated by increasing the pulse width to about  $170 - \mu s$  at which point the system began to stream data continuously prior to freezing. An audible “tick” sound was observed coming from the power supply circuitry with each incident laser pulse. This was a result of the power supplys inability to handle the current transients associated with the voltage level switches occurring in the sensor amplifier circuitry. These transients, in turn, caused instability in the voltage regulators due to insufficient phase margin in the regulator feedback path. The amplifiers drew unexpectedly high currents during pulse strikes peaking at a maximum of 1.5 amps. This peak was up from a nominal value of 0.15 amps on -3 volt and +3 volt rails. When combined with other EMI issues present on the power board (coupling between regulator circuits), these transients caused both intermediate voltage rails

(+5 volt and +15 volt) to collapse. This caused core FPGA voltage rails to also collapse. This resulting power glitch was sufficient to cause the FPGA to lose configuration, but was not long enough to trigger a full system reset. The result was erratic payload behavior followed by a freezing of the control processor. Figure 2.26 shows the Spartan core voltage dropping below the 800 millivolt limit after which FPGA configuration memory is corrupted. These problems were addressed on the second revision of the power board, which flew flawlessly on the 2013 flight. Details of the power board design can be found in [51].

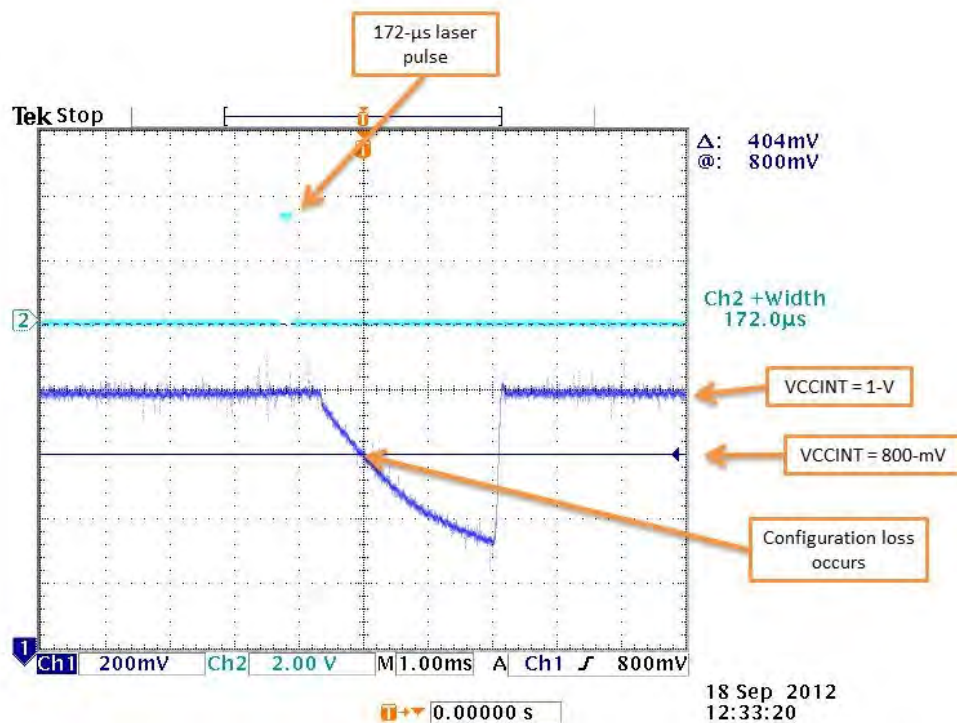


Figure 2.26: This figure shows the collapse of the core voltage rail on the Spartan FPGA. The collapse was triggered by stimulation of the radiation sensor, and caused the Spartan to malfunction during the 2012 flight.

HASP 2013 Flight: The HASP 2013 flight was uneventful compared to the 2012 flight. The hardware performed as expected for the duration of the flight. At launch, the payload was nearly stripped from the platform as it was struck by a boom on the launch vehicle immediately after launch. This impact ripped an adjacent payload off the platform, sending it plummeting to the ground. Our payload survived the impact owing to the robust enclosure construction. Aside from this there were no problems during the flight. The data collected showed no detected upsets in the FPGA. Configuration memory readback was not available during the flight, so it is possible that unused bits in the configuration memory were upset. Our detection scheme looked for upsets at the design level as indicated by the voter. Two radiation sensors were flown in this payload. The radiation sensor data was successfully logged to the MicroSD card and analyzed post flight.

Of particular interest in this data are events where the sensor count increases by a small number. In the lab the sensor has been observed entering an unstable state wherein the counter increases rapidly, eventually overflowing. During the flight, similar behavior was observed as the count on a couple channels overflowed. Looking at the data, this behavior corresponds closely to increasing internal payload temperature. It is interesting to note that when the temperature is increasing at a significant rate, the sensor channel counts increase similarly. As the payload temperature stabilizes, the counting ceases. There is an observed thermal effect that occurs when a positive temperature gradient is present in the system. The source of the count generation is hypothesized to be in the signal conditioning circuitry. It is conceivable that increasing temperature increases the thermal noise sufficiently to register as a strike event at the comparator. However, a further, in depth investigation is warranted prior to the redesign of the radiation sensor amplifier board.



### Drawbacks of This Approach

The TMR+Spares with configuration memory scrubbing system represents a big step toward using commercial SRAM FPGAs in space applications. However, there are a number of weaknesses that must be addressed in the current architecture. There have been a variety of system architectures developed based on this architecture, and a lot of engineering work has gone into creating custom hardware on which to open up further testing opportunities. However, an extensive analysis of the reliability of the system nor the effects each system element has on the overall reliability has been address. In any FPGA there are a number of resources that can be considered single-points-of-failure. These resources include administrative hardware cores that provide a unique capability and which can be instantiated as peripheral devices in a user design. Such cores include serial transceivers, clock management tiles, input/output buffers, clock distribution circuitry, configuration control hardware, etc. The components cannot necessarily be triplicated and therefore, if a system relies upon them for proper operation, a failure affecting these components may cause the system to cease proper operation. In addition to the native components portions of a user design may also act as single-points-of-failure. In the TMR+Spares architecture, and in any TMR implementation, the voter circuit is often identified as a liability as it cannot be triplicated. Intermediate voter circuits may be triplicated, but ultimately an output signal must control some singular resource, most commonly a package pin. These singular resources cannot accept multiple inputs. The impact of single-point-of-failure susceptibility on overall system reliability must be addressed.

### Current Research

The goal of this research from the start has been to increase the reliability of SRAM FPGA systems sufficiently enough that they may see wide adoption in space applications. What constitutes sufficient reliability has not been adequately discussed, nor have many of the single points of failure within the system. The research presented here takes a look at reliability in TMR systems, studies how TMR+Spares increases reliability beyond a simple TMR architecture, and proposes changes to the implementation of the TMR+Spares architecture to further enhance reliability. Reliability is the probability of a system operating properly for a specified period of time. As a probability, reliability can assume values between 0 and 1. A reliability of 1 indicates that the system is 100% reliable and will not fail. Achieving this level of reliability is practically impossible, but many applications seek reliabilities approaching this value. Mission critical functions, such as a guidance and navigation system used for autonomously landing an extraplanetary probe must have a very low risk of failure. A common metric used for determining whether a system is adequately reliable is the mean-time-to-failure (MTTF). This is the time at which the reliability of the system reaches a value of 0.5. The MTTF can be specified in any time unit (seconds, minutes, hours, etc.). The reciprocal of the MTTF is the failure rate, a measure of how often the system can be expected to encounter a failure. In space, the failure rate due to single event effects is quite variable as it depends on both environmental conditions and architectural properties. As previously mentioned, the environmental conditions include things like orbit type, location within orbit, magnetospheric charge trapping, solar conditions, etc. Architectural properties pertain more to factors under designer control such as shielding, system orientation on a spacecraft, component selection, etc.

The root of the space electronics problem is the fact that there is a trade-off between performance and reliability. It is not required that this trade-off be interpreted negatively. It must be recognized that there are different tasks required of a system, and rather than seeking a one-size-fits-all engineering solution it may be better to sort required functionality and assign it to technology with a suitable blend of performance and reliability. With regard to the research presented here, it is frequently observed that all is well with the TMR+Spares architecture, but the house of cards collapses when the voter circuit fails. Indeed, the voter is a single-point-of-failure and it must be adequately protected. In any TMR system, a failure of the voter circuit results in a system failure. The voter circuit is an exceedingly simple combinational logic circuit comprised of three ‘AND’ gates and one ‘OR’ gate. Figure 2.27 shows the circuit diagram and truth table for a voter circuit.

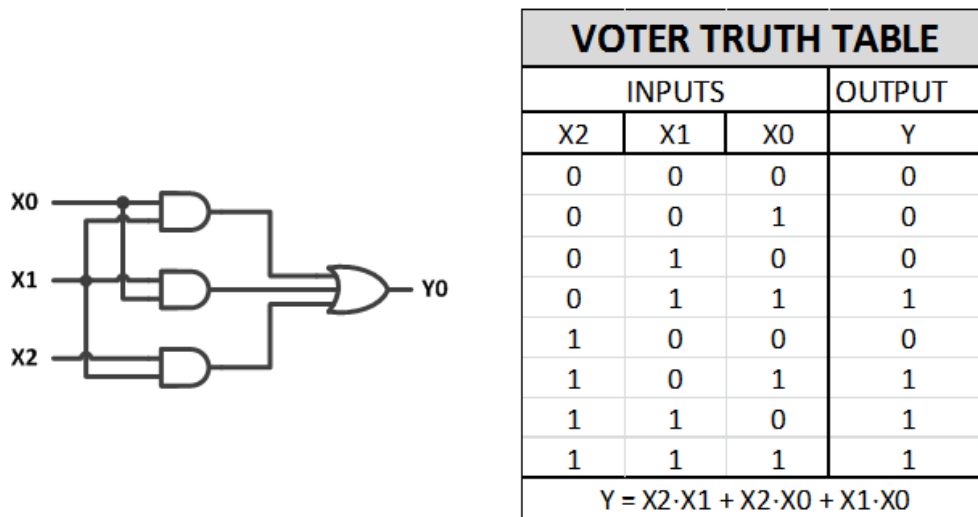


Figure 2.27: This figure shows the circuit diagram and truth table for a combinational logic voter circuit.

The voter functionality is only used to check final results within the system. Depending on the granularity with which the TMR implementation is realized, it may

not be necessary to use the voter very frequently. In the TMR+Spares system, the voter only operates on the system outputs. To the degree that the latency of the voter circuit logic is less than the function being performed by the tiles, the voter can be implemented on a device technology that is of lower performance, but of higher reliability. This reduces the impact of a single-point-of-failure on overall system reliability. The reduction is realized by taking advantage of the SEU trends with technology process scaling. Using older technology, single-points-of-failure can be partitioned onto devices with lower SEU rates than new technology. High performance circuits can be implemented in TMR on newer, more susceptible devices. In this way, the reliability of the system can be increased without decreasing performance.

The amount by which the reliability is increased depends on the susceptibility of the circuit to be migrated to a more suitable technology. A single-point-of-failure circuit occupying a substantial amount of device resources will yield a larger reliability increase compared to a similar circuit occupying minimal device resources. This research focuses on the effect of the voter circuitry on the overall reliability of a coarse-grained TMR+Spares system, and what benefits, if any, stand to be gained through implementing single-point-of-failure circuits on inherently less SEE susceptible device technology. The majority voter circuit is a single-point-of-failure in any TMR system, and the TMR+Spares architecture is no exception. A fault occurring in the voter has the potential to upset the operation of a system directly. In some designs the voter circuit is triplicated, but in a practical system signals internal to the computer must ultimately control some singular resource. The contribution presented here looks at the susceptibility of the voter to single event effects, how that susceptibility affects overall system reliability, and how technology partitioning can be used to minimize the effect of the voter on reliability. The research is presented as a generalized study

that can be applied to the partitioning of any amount of circuits between multiple technologies with the intent of increasing overall reliability.

A second research item is a general reliability analysis of the TMR+Spares architecture. One question that has yet to be answered is “how does the MTTF trend as a function of the number of spares available in the system?”. It is beneficial to understand this behavior in order to avoid diminishing returns with increased power associated with higher device resource utilization.

To answer these questions, the system reliabilities for a number of different configurations were modeled using Markov chains. The Markov chain is a statistical tool used for modeling processes which move through a sequence of states. The transitions are governed by the probability associated with moving from the current state to a neighboring state. In this application, the transition probabilities are calculated based on the SEU rate, which follows a Poisson distribution. As such, the models are generalized to allow input of any SEU rate. The probability of being in a given state at a given time is calculated using a transition probability matrix. The end result is a curve showing the reliability as a function of time. Some simple Markov models can be solved to find the closed-form solution [54] for the reliability curve, but as the number of states, repair rates and fault rates increases it becomes much simpler to solve for the reliability numerically.

## DESIGN OF EXPERIMENTS

Modeling Reliability

To begin, a simple definition of reliability is in order. According to [55], reliability is “the probability of [a system] working for a specified period of time”. In space applications, we are concerned with the probability that a system will operate as designed for a specified period of time. This period of time is highly dependent on the nature of the mission or the function of the system under study. For example, a system controlling the thrusters of a rocket-propelled probe must function properly over the duration of the mission. If this system fails it does so at the expense of the entire mission. High reliability communications links, e.g. combat satellite communications systems, cannot afford to be down during a battle on the ground. These systems must be sufficiently reliable to perform their specified task for a known period of time.

To estimate system reliability, statistical tools are put to use. In space systems, the effects due to ionizing radiation contribute to device performance interruptions and degradation. As a result, these external environmental factors must also be considered in addition to the normal device reliability statistics. In space applications, TMR has long been used as way to increase the reliability of complex systems by providing parallel paths through a system. With parallel paths, a failure in one branch does not necessarily cause a failure of the system to perform its specified function.

To calculate the reliability of a system, one starts with a probability density function representing the operational lifetime of a device. The exponential distribution is commonly used to represent the lifetime of complex systems [55]. Taking this distribution as an example, the steps for calculating system reliability are shown. The

mathematical equation for the exponential density function is shown in Equation 3.1. This equation provides the relative frequency of lifetimes of a given system.

$$f(t) = \begin{cases} \lambda e^{-\lambda t} & t \geq 0 \\ 0 & \text{else} \end{cases} \quad (3.1)$$

In Equation 3.1, the parameter  $\lambda$  represents the failure rate of the system. In the reliability analyses performed here,  $\lambda$  represents the radiation-induced fault rate. This failure rate describes a Poisson distribution, but is estimated as a constant value using available modeling tools. In fact,  $\lambda$  varies significantly with time and location within an orbit. To determine the likelihood of a system operating for a specific duration, the probability distribution function is used. This estimate is calculated by integrating the density function up to the specified value, as shown in Equation 3.2. This distribution provides the probability of failure versus time. For a system with lifetime represented by random variable  $L$ , the value of the distribution function  $F(t)$  is the probability of failure at time  $t$ .

$$F(t) = P(L \leq t) = \int_{-\infty}^t f(\tau) d\tau \quad (3.2)$$

$$F(t) = P(L \leq t) = \int_0^t \lambda e^{-\lambda \tau} d\tau \quad (3.3)$$

$$F(t) = 1 - e^{-\lambda t} \quad (3.4)$$

Previously the reliability was defined as the probability of a system functioning as specified for a given time interval. Since the distribution function gives the probability of failure, it follows that the reliability is simply  $1 - F(t)$ .

$$R(t) = 1 - F(t) \quad (3.5)$$

For the exponential distribution, Equation 3.5 becomes the expression shown in Equation 3.7.

$$R(t) = 1 - F(t) = 1 - (1 - e^{-\lambda t}) \quad (3.6)$$

$$R(t) = e^{-\lambda t} \quad (3.7)$$

In order to analyze the reliability of individual system components, such as the triplicated portion of a circuit and its corresponding voter circuit, separate reliability expressions must be used. The total system reliability is then the product of the individual reliabilities. Equation 3.10 illustrates this.

$$R_{TMR}(t) = e^{-A_1\lambda_1 t} \quad (3.8)$$

$$R(t)_{voter} = e^{-A_2\lambda_2 t} \quad (3.9)$$

$$R_{sys}(t) = R_{voter} \cdot R_{TMR} \quad (3.10)$$

This shows that as time increases, the reliability decreases exponentially. Intuitively, this makes sense as the likelihood of failure increases as time goes on. This is observed on a daily basis with computer systems. Older devices tend to fail more frequently than newer devices.



### Modeling Reliability of Various Architectures

This technique for arriving at the expression for system reliability is useful for single, simple systems. As the complexity of a system increases and multiple sources of failure are accounted for independently this approach to calculating the reliability becomes cumbersome. A related but more appropriate technique is to use Markov modeling to calculate the total system reliability. A Markov chain models a system as a state diagram and a set of probabilities describing the likelihood of transitioning among states. For information on the procedure for arriving at the same expression for reliability as Equation 3.6 using Markov modeling techniques the reader is encouraged to consult [54].

#### Simplex

The procedure outlined in [54] formed the basis for the Markov modeling performed in this research. In a Markov chain, the system is modeled as a sequence of states. The likelihood of transitioning from the current state to an adjacent state is determined by a probability of occurrence. These probabilities are related to the failure rate of the system, and are conveniently represented in matrix form. As an example consider a simple system consisting of two states: “good” and “bad”.  $S1$  denotes the good state, where the system is functioning properly, and  $S2$  denotes the bad state, where the system has failed. The state transition diagram for this system is shown in Figure 3.1.

A transition matrix  $\mathbf{T}$  is defined that contains the probabilities of transitioning between each state. The entry in position  $m, n$ , where  $m$  is the row of  $\mathbf{T}$  and  $n$  is the



Figure 3.1: This figure shows the Markov chain for a simple, two-state system. The fault rate is given by  $\lambda$ .  $S1$  is a healthy state and  $S1$  is a failed state.  $\lambda\delta t$  is the probability of transitioning from  $S1$  to  $S2$ . As there is no recovery process, once the system reaches  $S2$  it remains there with probability 1.

column of  $\mathbf{T}$  provides the probability of transitioning from state  $m$  to state  $n$ . This matrix is defined in Equation 3.11.

$$\mathbf{T}\{m, n\} = \begin{bmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,n} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,n} \\ \vdots & \vdots & \vdots & \vdots \\ t_{m,1} & t_{m,2} & \cdots & t_{m,n} \end{bmatrix} \quad (3.11)$$

For the system of Figure 3.1 the transition matrix is shown in Equation 3.12.

$$\mathbf{T} = \begin{bmatrix} 1 - \lambda \cdot \Delta t & \lambda \cdot \Delta t \\ 0 & 1 \end{bmatrix} \quad (3.12)$$

Equation 3.16 shows the probability of residing in state  $S$ ,  $S = 1 : n$  after time step  $k$ , which is calculated using Equations 3.13.

$$p_S(t = k\Delta t) = \begin{bmatrix} p_1(0) & p_2(0) & \cdots & p_n(0) \end{bmatrix} \cdot \begin{bmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,n} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,n} \\ \vdots & \vdots & \vdots & \vdots \\ t_{m,1} & t_{m,2} & \cdots & t_{m,n} \end{bmatrix}^k \quad (3.13)$$

$$p_S(t = k\Delta t) = \begin{bmatrix} p_1(0) & p_2(0) \end{bmatrix} \cdot \begin{bmatrix} t_{1,1} & t_{1,2} \\ t_{2,1} & t_{2,2} \end{bmatrix}^k \quad (3.14)$$

$$p_S(t = k\Delta t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 - \lambda \cdot \Delta t & \lambda \cdot \Delta t \\ 0 & 1 \end{bmatrix}^k \quad (3.15)$$

$$p_S(t = k\Delta t) = \begin{bmatrix} p_1(k) & p_2(k) & \cdots & p_n(k) \end{bmatrix} \quad (3.16)$$

$$R = 1 - p_2(t = k\Delta t) \quad (3.17)$$

In an absorbing Markov chain, the final state in the vector  $p_n$  is a failure state that requires system reset for recovery [56]. In this research, the  $n^{th}$  state in a system with  $n$  states is absorbing. The probability of being in this state,  $p_n(t = k\Delta t)$ , represents the probability of system failure. Thus, the reliability is  $1 - p_n(t = k\Delta t)$ . The primary metric of interest in this study is the mean time to failure (MTTF), which is defined as the point  $t$  at which the reliability  $R(t) = 0.5$ . Figure 3.2 shows a plot of the reliability for the simple, non-redundant system of described by Equation 3.12. This numerical technique for estimating system reliability is used here. In this example, the fault rate  $\lambda$  was chosen somewhat arbitrarily at the upper range of expected values

for a Virtex-6 device in low-earth orbit. The fault rate describes the frequency with which SEUs are expected to occur in a device.

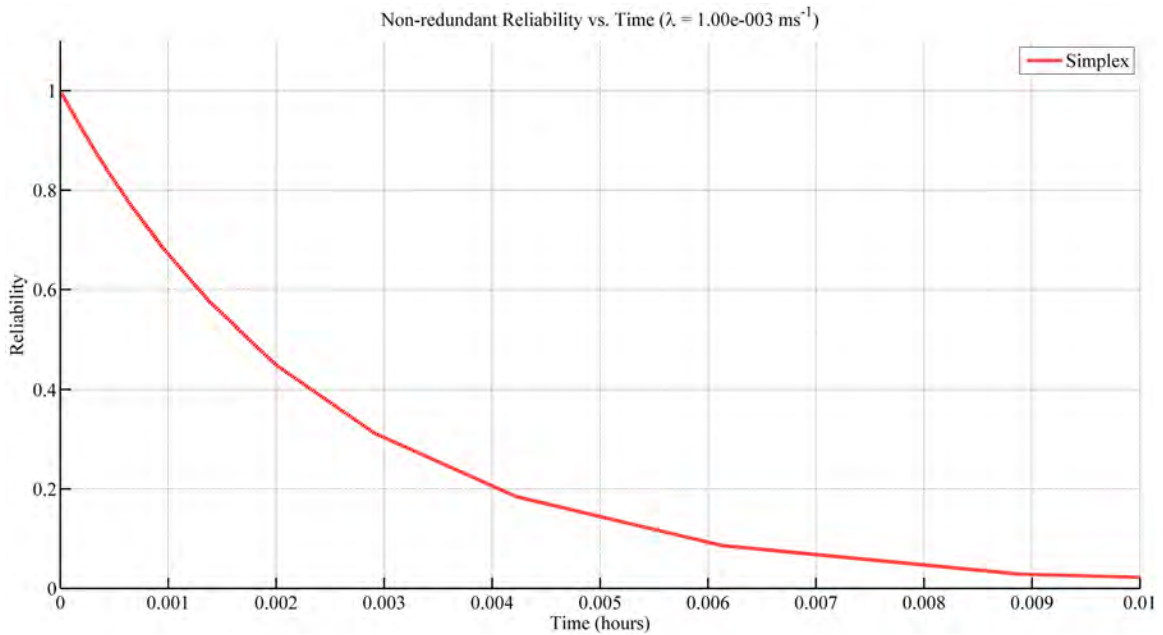


Figure 3.2: This figure shows an exponential reliability curve for a simple two-state system generated using a Markov chain. For this model, the input fault rate was  $1.02E^{-08} \text{ ms}^{-1}$ , which is in the upper range of values for a Virtex-6 device in low-earth orbit.

### Triple Modular Redundancy

Redundancy implies the use of multiple independent systems to perform an identical task. In aerospace applications this is often realized as completely separate, sometimes identical hardware components being tasked with the same work, as in the case of the space shuttle’s digital fly-by-wire system. Concerned with reliability, engineers used five identical computers to implement a control system. Four of these computers ran in parallel while the fifth computer was reserved as a spare [57]. The

basis for using redundant hardware is found in basic statistical analysis techniques. The basic structure of a TMR system is shown in Figure 3.3.

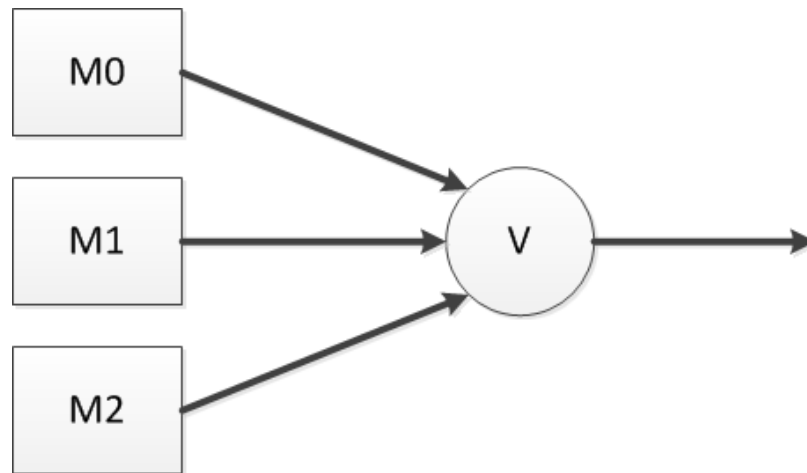


Figure 3.3: This figure shows the structure of a basic TMR system.  $M_0$ ,  $M_1$ , and  $M_2$  are identical hardware components. Their outputs are passed to a majority voter circuit, labeled  $V$ , which determines the final system output. A mismatch of one of the systems indicates a failure.

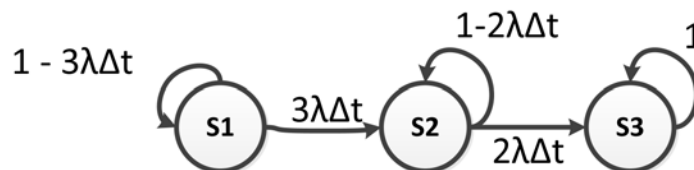


Figure 3.4: This figure shows the Markov chain representation of a basic TMR system. In  $S_1$  all three elements are operational. In  $S_2$  one of the three elements is operational. In  $S_3$  any two of the three units are faulted and the system is in a failure state.

For a TMR system, it is required that at least two of the redundant modules operate properly. Mathematically, any time two systems must operate concurrently they are considered to be in series, and their total reliability is the product of their individual reliabilities. The system is considered to be operating properly if all three modules are healthy ( $S_1$ ), or if any pair of modules is working ( $S_2$ ). The probability

of these conditions being met are shown in Equation 3.18 [34]. Figure 3.5 shows the reliability of a TMR system compared to a simplex system with the same fault rate of  $\lambda = 1E^{-3} SEU \cdot ms^{-1}$ . Of note is the fact that the reliability is higher for TMR systems before the MTTF, but the reliability is actually lower than a simplex system after the MTTF [58]. This is attributable to the fact that after one unit fails in a TMR system, it is required that both remaining systems remain operational. The added area effectively doubles the fault rate over a simplex system hence the reduced reliability [58]. The MTTF for TMR and simplex systems is the same.

$$R(t) = R^3(t) + 3R^2(t)(1 - R(t)) = 3R^2(t) - 2R^3(t) \quad (3.18)$$

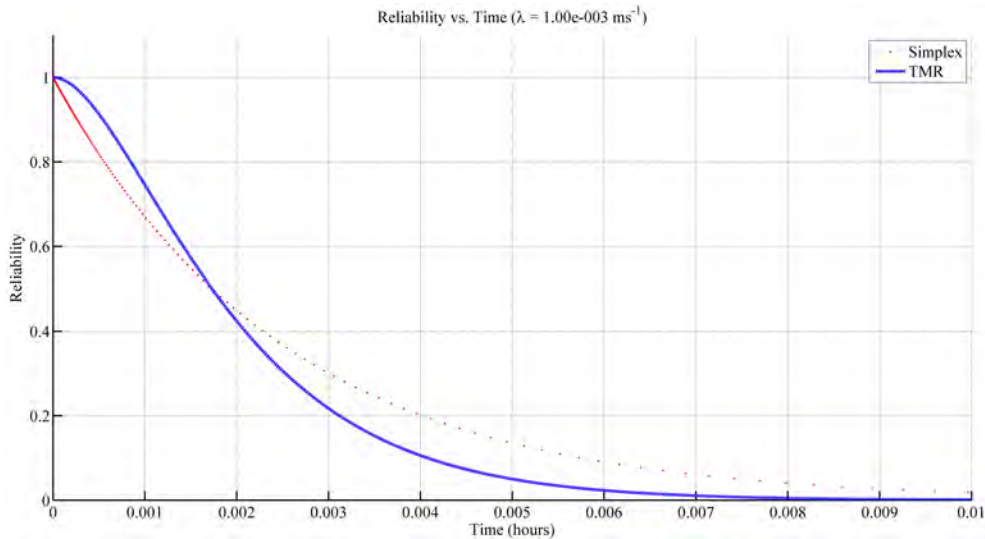


Figure 3.5: This figure shows the reliabilities of TMR and simplex systems. The MTTF of each system is the same, though the TMR system has lower reliability in long-duration applications.

### Spare Circuitry

To build on the standard TMR architecture, spare processing resources are added to the system. The thought behind adding spare processors is that more faults are required for the system to reach a failure state. The addition of spares should increase the MTTF of a system due to the fact that there are an abundance of circuits. In a high fault rate environment, as processors are faulted the system gradually degrades until there are no longer a sufficient number of tiles for the system to operate properly. In a TMR system, that number is one. When there is only one tile remaining there is no longer any way for the system to check the output thus no way for the system to determine whether or not it has experienced a design-level fault. Figure 3.6 shows the Markov model for a TMR system with a spare tile and no repair capability.

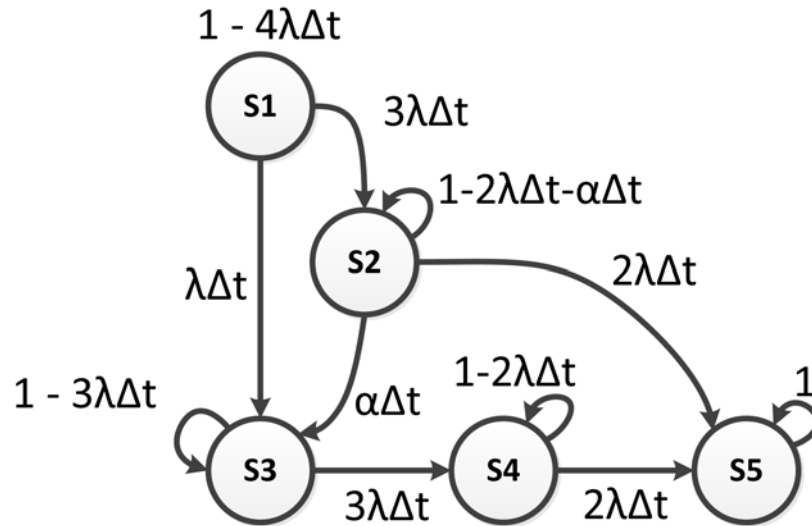


Figure 3.6: This figure shows the Markov model for a TMR+Spares system with a single spare tile and no repair capability.

### Repair Capability

In order to increase the MTTF of the system it is necessary to introduce the capability to repair faults in the system. This way the reliability is penalized for extra area to a lesser degree since faults are prevented from accumulating in the system. In many FPGA systems this repair is performed using a scrubbing process. The time it takes to fully scrub the device is known as the scrub rate, or repair rate. The repair rate, represented here by the Greek letter  $\mu$ , is the number of repairs per millisecond. The maximum repair rate is dependent on a variety of factors including device size (in terms of logic resources), the bandwidth of the configuration interface, the bandwidth of the storage device containing the golden configuration data and the latency of fault detection. These parameters vary on a per-design basis. The repair rate for these studies is taken from those measured in test systems at approximately 100-ms per repair, or  $\mu = 0.01 \text{ repairs} \cdot \text{ms}^{-1}$ . Figure 3.7 shows a Markov model for a TMR system with repair capability.

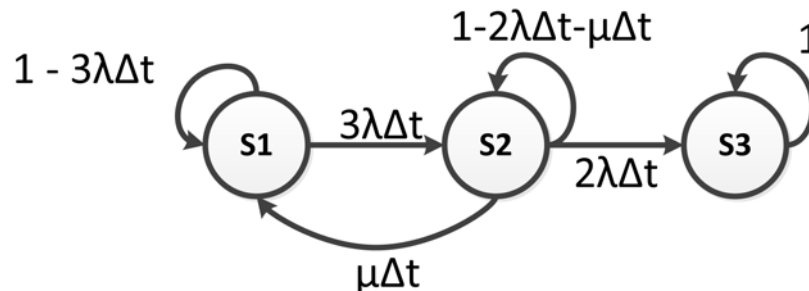


Figure 3.7: This figure shows the Markov chain for a TMR system with repair rate  $\mu$  and fault rate  $\lambda$ .

The results of a Markov model are demonstrative of the qualitative advantage of using a repair process in a TMR system. Figure 3.8 shows that the MTTF, and



reliability in general, are much greater for a TMR with repair system compared to TMR and simplex systems.

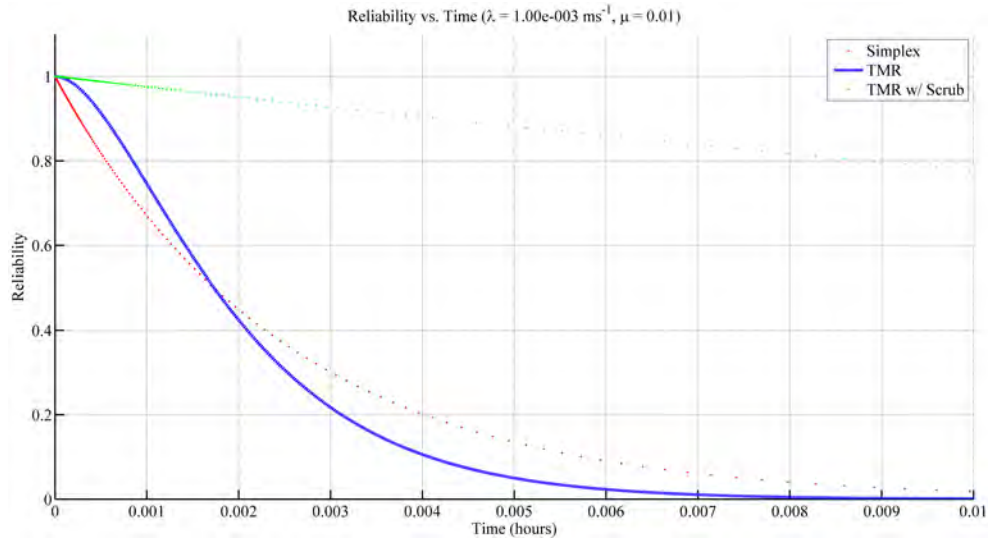


Figure 3.8: This figure shows the reliability curves for simplex, TMR and TMR with repair implementations. The advantage of implementing a repair process is clearly demonstrated by the substantially lower decay rate of the reliability for the TMR w/repair system compared to simplex and TMR only.

### Spare Circuitry with Repair

TMR with repair provides a substantial increase in reliability over a TMR system. The premise underlying the TMR+Spares w/Repair architecture is that it is faster to change from a faulted resource to a healthy resource than it is to repair a faulted resource. This leads to a further increase in reliability compared to the TMR w/repair system. Figure 3.9 shows the Markov chain for a TMR+Spares w/Repair system with a single spare. Systems with more than one spare are identical in structure to Figure 3.9 with states  $S1$  and  $S2$  repeated for each additional spare. In this architecture,  $S1$  represents the healthy state wherein no resources are faulted. There is one spare and three healthy processing tiles. There are two failure paths out of  $S1$ :

a fault in the spare tile transitions the system to  $S3$ , and a fault in one of the active tiles transitions the system to state  $S2$ . In state  $S2$  the system must undergo a context switch to replace the faulted active tile with a healthy spare. This is performed at a rate  $\alpha$  swaps per millisecond. If a fault is experienced in either of the two other active tiles then the system cannot be restored to a functional state. Thus, the system transitions to state  $S5$ , which is an absorbing state. A system reset is required for recovery. From the state  $S3$ , where the spare tile is faulted, the system can transition back to state  $S1$  through the recovery process. Recovery occurs at repair rate  $\mu$ . If an active tile is faulted before the spare tile is repaired, the system transitions to state  $S4$ . In  $S4$ , the system is technically still functional as the faulted active tile outputs are mitigated by the voter circuit. From  $S4$  the faulted active tile can be repaired to transition back to state  $S3$ . Should a fault occur in either of the remaining two active tiles in state  $S4$ , the system transitions to the failure state  $S5$  and a reset is required. Figure 3.10 shows the reliability performance compared to the other systems.

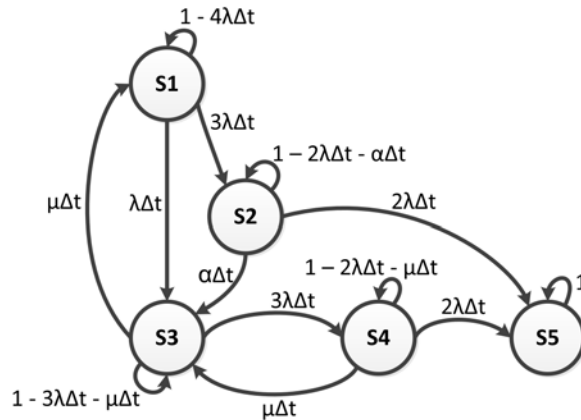


Figure 3.9: This figure shows the Markov chain for a TMR system with repair rate  $\mu$  and fault rate  $\lambda$ . It also models the presence of spares in the system and the effect of SEUs occurring during the context switch process. Strikes during a context switch are assumed to result in a system failure as the ability to synchronize the newly replaced tiles is compromised.

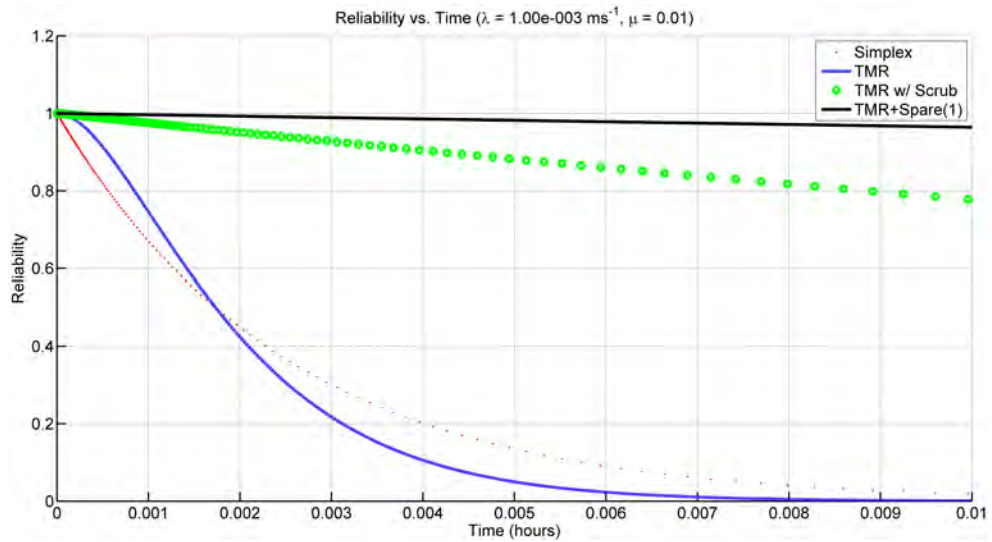


Figure 3.10: This figure shows the reliability curves for simplex, TMR, TMR with repair and TMR+Spare(1) implementations. The advantage of implementing a spare tile in addition to a repair process is clearly demonstrated by the substantially lower decay rate of the reliability for the TMR+Spare system compared to the others. The fault rate  $\lambda$  and repair rate  $\mu$  are the same as in the previous models.

### Method to Estimate Fault Rate

#### Environment Factors Affecting Fault Rate

The fault rate is perhaps the most critical parameter in the Markov chain models used to estimate the reliability of the system. It is also a very difficult parameter to estimate as it is dependent on a range of factors. The factors fall into two categories: device factors and environment factors. Environment factors include the severity of the radiation environment, and the breakdown of the radiation spectrum. The radiation environment begins with the particle flux. This describes the number of particles passing through a particular area as a function of time. The particle flux represents a spectrum of particles of varying types, energies, and linear energy transfer in silicon. The particles of greatest concern are heavy ions, and high-energy protons

trapped in the Van Allen belts. Particles below a certain energy are unlikely to induce SEUs in a device. The particle flux is highly dependent on orbit. The scope of this research is limited to low-earth orbit (LEO), which has the properties listed in Table 3.1.

Table 3.1: Low-Earth Orbit Parameters.

Min. Altitude	Max. Altitude	Min. Orbit Period	Max. Orbit Period
160 km	2000 km	90 minutes	127 minutes

#### Device Attributes Affecting Fault Rate

The SEU cross-section is a parameter used to determine the threshold particle energy required for inducing an upset. As LET increases for a given particle, the probability of inducing an upset increases until it reaches a saturation value. This saturation value is often taken as the bit cross-section, and is given in  $\frac{cm^2}{bit}$ . Thus the number of bits used in a design, often referred to as the area of a design, is the cross-sectional area of importance in determining the number of upsets to expect given a particle flux spectrum. Device factors include the manufacturing process node, the transistor architectures, isolation techniques, doping levels of the silicon substrate, critical charge values, etc. Many of these parameters are not immediately available from manufacturers as they are majorly irrelevant to designers in most applications. However, estimation of the fault rate using available tools, such as CREME96, depends on some of these parameters. Therefore, the accuracy with which the fault rate is estimated for a given orbit location under specified solar conditions is limited by the accuracy with which the device parameters can be estimated. For fault rate estimation using CREME96 in this research, the critical charge, sensitive volume and bit cross-sectional area were used. The cross-sectional area was taken to be

$5.73E^{-4}(\mu m^2 bit^{-1})$  [59] for the Virtex-5 and  $2.75E^{-04}$  for the Virtex-6, which was estimated based on process node and suggestions offered in the CREME96 documentation. The sensitive volume, which is the region of each transistor within which single event effects originate, was taken as the cube of the process node. The critical charge was calculated using Equation 3.19 [60].

$$Q_{crit} = 0.023 \frac{pC}{\mu m^2} \cdot L^2 \quad (3.19)$$

In addition to modeling the radiation environment to estimate fault rates, radiation testing is often performed at a variety of particle accelerators around the country to directly measure the SEU cross-section of many devices. Unfortunately, these tests are often performed on radiation hardened components as a means of verifying their hardness. This makes sense as system designers use radiation hardened components in space applications rather than off-the-shelf components. As such, radiation test data for off-the-shelf components is relatively scarce. In this research, models are generated for a broad range of fault rates, which is perhaps more insightful than targeting specific devices for study as the results can be applied to a broader range of technologies. To the extent that fault rate data is available or easily estimable for relevant FPGA devices it will be used in the research. Table 3.2 shows the fault rates for a variety of FPGAs in the Xilinx Virtex device family. The fault rates for the Virtex, Virtex-II and Virtex-4 are the results of CREME96 simulations performed in [61]. These values are the estimates for  $\lambda$  used in the Markov models used to calculate system reliability. For each device the fault rates are presented for five solar conditions: solar minimum, solar maximum, worst week, worst day, and peak five minutes. Solar minimum conditions model the base radiation environment without the presence of solar flares. This environment varies with the solar cycle, so it is

represented differently during solar maximum than at solar minimum. To account for this a fault rate for solar maximum is also included. For more harsh conditions the peak rates over a week, day and five minutes are included. Historical measurements form the basis for these conditions, the details of which are provided in [62].

Table 3.2: Virtex family SEU fault rates ( $SEU \cdot device - day^{-1}$ ).

	Virtex [61]	Virtex-II [61]	Virtex-4 [61]	Virtex-5	Virtex-6
Family	1000	6000	LX200	LX50T	LX75T
Process ( $nm$ )	220	150	90	65	40
Solar Min.	$8.62E^{-01}$	$4.66E^{+00}$	$5.50E^{+00}$	$8.71E^{+00}$	$1.57E^{+01}$
Solar Max.	$5.21E^{-01}$	$2.87E^{+00}$	$3.37E^{+00}$	$6.16E^{+01}$	$1.84E^{+02}$
Worst Week	$5.52E^{-01}$	$3.17E^{+00}$	$3.64E^{+00}$	$9.88E^{+01}$	$3.62E^{+02}$
Worst Day	$5.56E^{-01}$	$3.07E^{+00}$	$3.57E^{+00}$	$9.88E^{+01}$	$3.73E^{+02}$
Peak	$6.50E^{-01}$	$3.59E^{+00}$	$4.09E^{+00}$	$8.07E^{+03}$	$1.947E^{+04}$

## RESULTS

Analysis of System Architecture Effects on Reliability

The reliability for many-tile systems has been previously estimated using Markov techniques similar to those discussed in the previous chapter [10]. These estimates were for simple counter tiles numbering as many as 64. As was necessary, some simplifying assumptions were made in the model including a perfect voter circuit and an estimate of the number of essential bits in the design. The first analysis performed investigated the benefit of adding spare tiles to a system without including any capacity to repair faults. It was found that the addition of spare tiles did indeed increase the MTTF of the system above a TMR system with no spares, but the benefit was miniscule in comparison to any TMR system with a scrubbing routine. Figure 4.1 shows the MTTF versus failure rate for Simplex, TMR, TMR+Spares(1), and TMR+Spares(2). TMR+Spares(1) is a TMR system with no repair and a single spare tile. TMR+Spares(2) is a TMR system with no repair and two spare tile.

The model presented here estimates the reliability of our 9-Tile Microblaze system operating in the 51.6 degree, 500 km low-earth orbit of the International Space Station. The fault rates for this analysis were in the range of  $1E^{-06}$  to  $1E^{+00}$ , or approximately 86 to 86,000,000 SEU per device per day for the Virtex-6. The high fault rate is far in excess of the estimated peak five minute proton flux during a solar flare event, which is about 20,000 per device per day. This was done to investigate behavior of the MTTF as a function of the fault rate. The effect of variation of area from design to design is to de-rate the fault rate since the full device is not generally used. Only a subset of the configuration bits are relevant to a given design. These bits are known as essential bits. Strikes in non-essential bits are not considered to

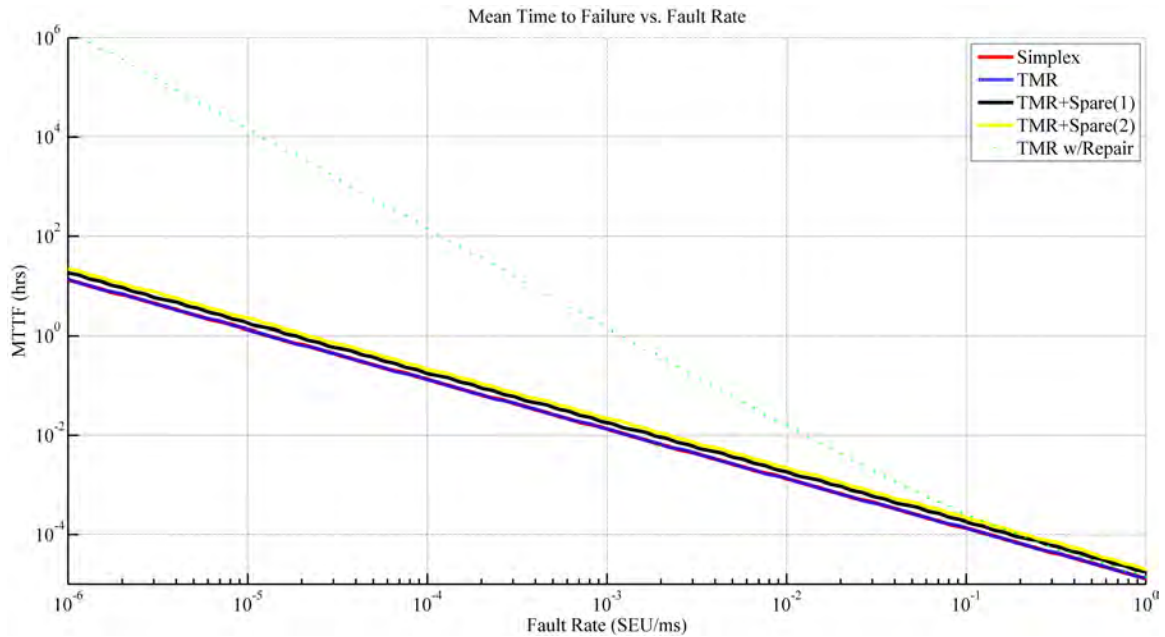


Figure 4.1: This figure shows that adding spare tiles to a TMR system increases the MTTF of the system. The repair capability, added to any system, increases the MTTF significantly more than adding spare resources without repair.

be problematic as they don't adversely impact proper operation of the design. The area for the 9-tile Microblaze system is reported as 13.72% by the Xilinx bitstream generation tool. This value was used in the Markov reliability model, which produced the family of curves shown in Figure 4.2.

From this analysis there are several important observations to be made. First, the MTTF even for the simplex system is fairly long, and it is important to consider that this analysis only pertains to upsets to the configuration memory. The design utilizes only about 13% of the configuration memory bits, so the effective fault rate is quite low. Of more importance is to examine the difference between the MTTF for the given technologies rather than the value of the MTTF itself. This shows the advantage of using spare resources in a system. Previously it had been assumed that more spares would automatically result in a more reliable system as it would be able



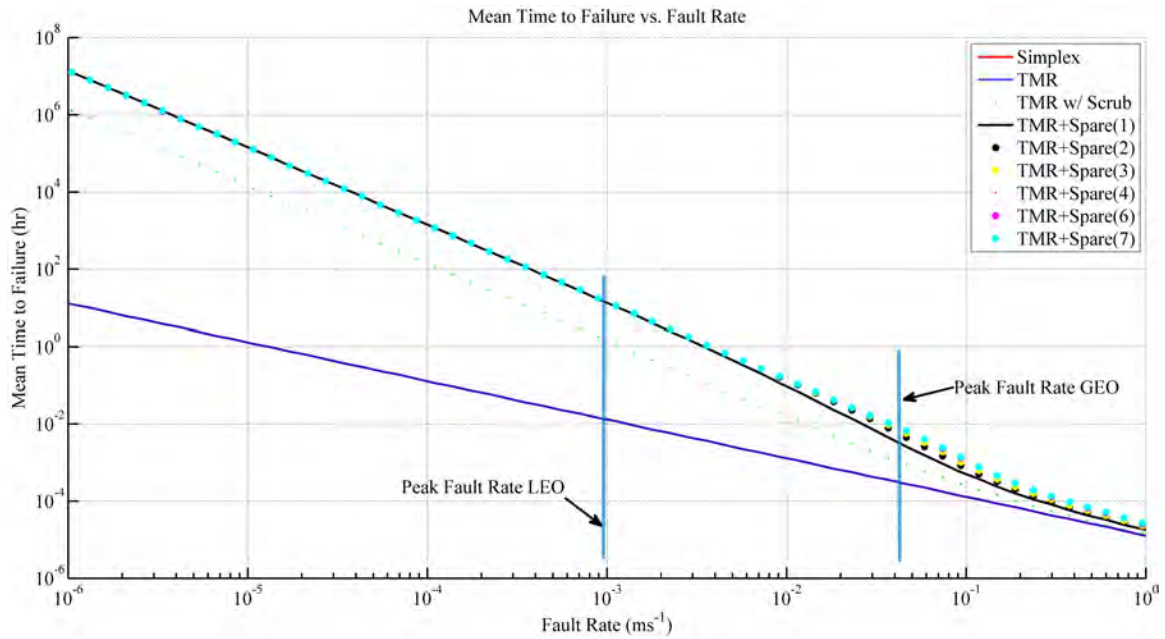


Figure 4.2: This figure shows the Markov chain results for a variety of implementations of a Microblaze system including simplex, TMR, TMR w/Scrub, TMR+Spares. This figure shows that the MTTF for a simplex and TMR system is the same. Adding a repair process significantly increases the MTTF. MTTF is further increased by the addition of spare resources, but the improvement diminishes beyond one or two spares until exceedingly high fault rates are encountered.

to survive longer times in high fault rate environments. However, this analysis shows that there is a diminishing benefit to adding more than one or two spare resources to a system. For exceedingly high fault rates there is some improvement of MTTF, but for practical radiation environments it appears that one or two spares is sufficient to realize the majority of the increase in MTTF. It is likely the case that the added area of the spare tiles offsets the benefit by proportionately increasing the observed fault rate. A larger area, therefore a larger device utilization, results in more observed faults. As the fault rate approaches the repair rate of the system, the benefit of having spares present decreases and the MTTF of TMR+Spares w/Repair, for any number of spares, begins to converge to that of a conventional TMR system. Regardless,

the presence of spares in radiation environments typical of low-earth orbit provides a substantial increase in MTTF over all other implementations. Figure 4.3 shows the MTTF as a function of the number of spares in a system demonstrating that simply adding more spares to a system does not necessarily yield improvements in reliability in the typical range of fault rates for a LEO orbit.

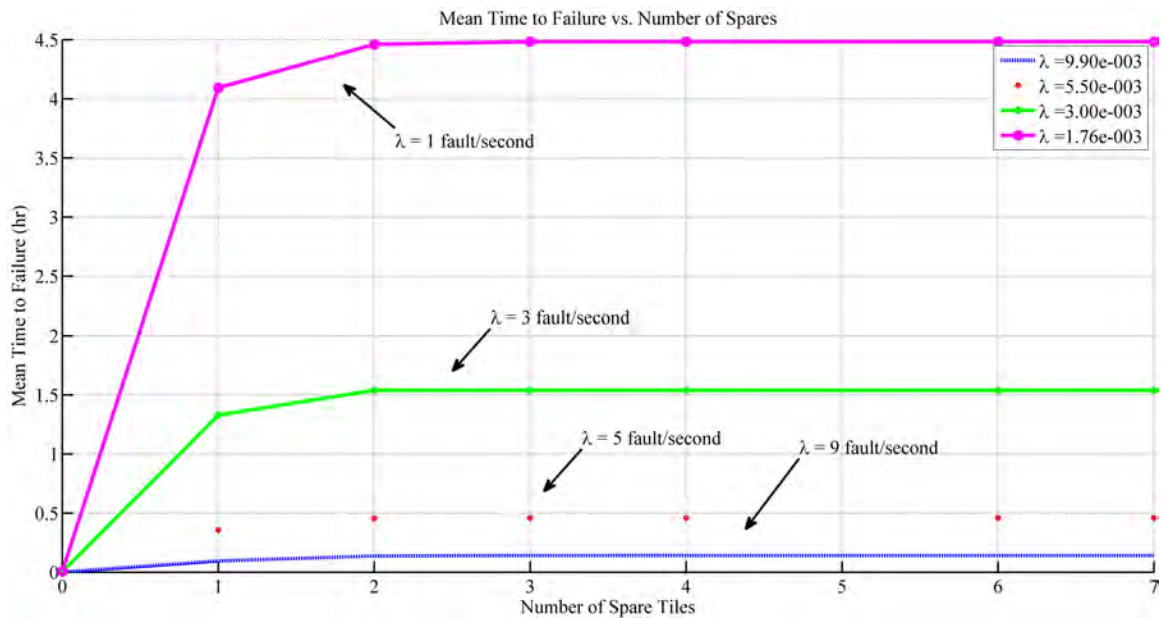


Figure 4.3: This figure shows the diminishing benefit of arbitrarily increasing the number of spare resources in a TMR+Spares system. There is marked benefit to using TMR+Spares, but increasing the spares beyond one or two results in unnecessarily increased resource utilization. The family of curves presented in this figure shows that this is true regardless of fault rate.

#### Analysis of Single Point of Failure Area on Reliability

To address the question of voter susceptibility, in particular the value of technology-partitioning the voter as a single point of failure on a device with an increased SEU tolerance, an analysis was performed to quantify the reliability of the

voter itself. The area of the voter circuit was found to be 0.34% by the bitstream generation tool. This small area is to the benefit of the system as it strongly decreases the effective fault rate.

When modeling the reliability of the system accounting for faults to the voter, the overall reliability is determined by multiplying the reliability of the TMR+Spares design with the reliability of the simplex design since both must be operating in order for the system to function. Clearly, the simplex elements may have a substantial effect on overall reliability, and therefore must be minimized in area. This analysis quantifies the susceptibility of the voter and investigates the benefit of implementing it on older Virtex FPGA technology nodes. Figure 4.4 shows the reliability versus time for a voter circuit and TMR circuit individually, and the equivalent reliability of the system as a whole.

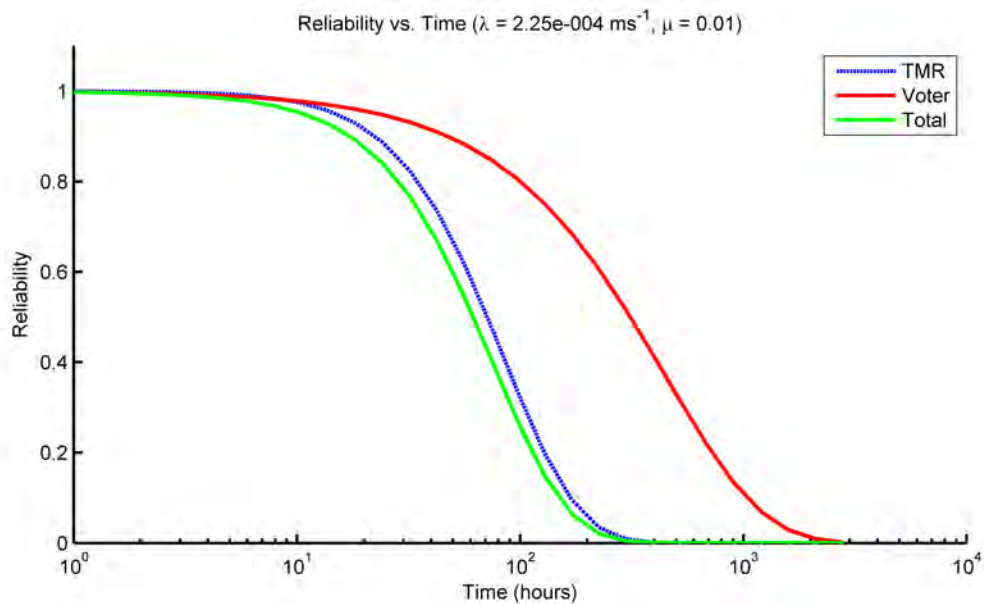


Figure 4.4: This figure shows the effect of the susceptibility of the voter circuit as a single point of failure in a TMR system. Due to its very small size, the voter is more reliable than the TMR portion of the design. However, the reliability of the entire system is reduced when accounting for the reliability of the voter.

This analysis simply demonstrates the fact that as a single-point-of-failure in a TMR system the voter diminishes the overall reliability. The amount by which the reliability is decreased is primarily dependent on the voter area. The total area of the voter must be accounted for in order to accurately estimate the reliability impact. This result only applies to a traditional TMR system. Since the reliability of a TMR+Spares system is significantly higher than TMR alone, it is instructive to evaluate the impact of voter susceptibility in such systems as well. Figure 4.5 highlights the impact the voter has on the reliability of a TMR+Spares system.

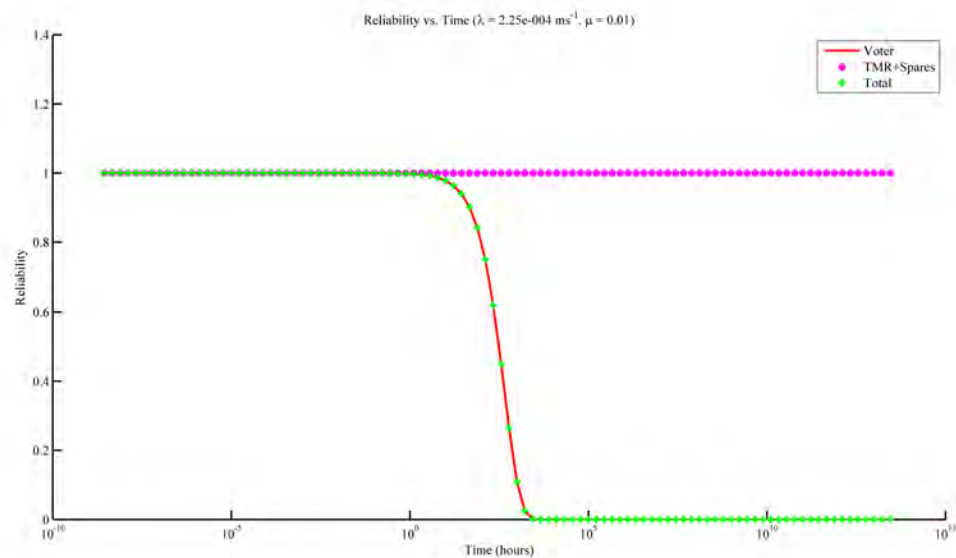


Figure 4.5: This figure shows the effect of the susceptibility of the voter circuit as a single point of failure in a TMR+Spares system.

The importance of this result cannot be understated. The reliability of a space-based FPGA system abides by the philosophy that a chain is only as strong as its weakest link. In Figure 4.5 the reliability of the TMR+Spares system is substantially greater than that of the voter circuit, remaining at a value near 1 for the duration of the simulation time. However, the reliability of the system considered as a whole is

equal to that of the voter circuit itself. This finding motivates the use of a scrubber circuit to maintain the voter and/or technology partitioning, wherein single points of failure are implemented on devices with greater inherent tolerance to SEEs.

### Analysis of Technology Partitioning on Reliability

This analysis assumes that a fault to the voter results in a system failure requiring reset. It is emphasized that this is not a permanent failure of the system, rather entry into a state in which the system temporarily cannot perform its desired function. In addition to implementing the voter on a more suitable technology, using a scrubber to periodically repair the circuit increases the reliability as well. In fact, the analysis shows that the effect of the scrubber far outweighs any potential gains to be realized by technology partitioning. The reliability of the voter, when coupled with scrubbing to repair faults, is increased beyond that of the 9-tile Microblaze system implemented on the Virtex-6. This leaves the tiled portion of the design as the weakest link and the determining factor in the overall system reliability. Figure 4.6 shows that regardless of what technology the voter is implemented on, the reliability of the system remains the same. The actual MTTF of the voter circuit was unable to be determined as the number of time steps exceeded the capabilities of the simulation. This analysis was performed with system parameters of peak five minute SEE fault rates for each device, a 100 millisecond tile repair time, a one millisecond tile swap time and a one second full scrub time.

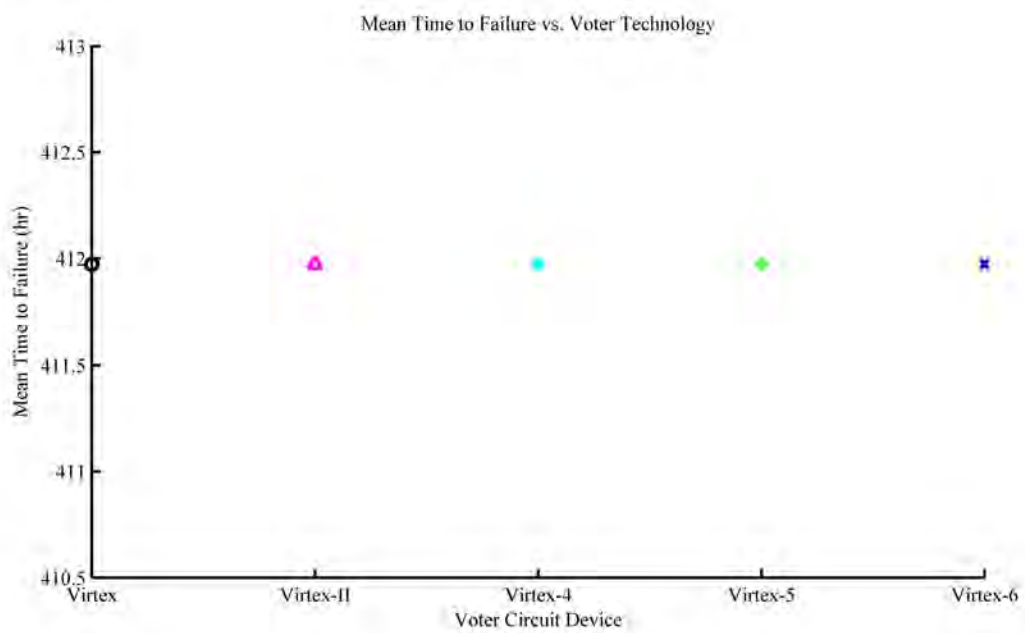


Figure 4.6: This figure shows that scrubber functionality has sufficient impact to negate the necessity of technology partitioning small-area single-points-of-failure in a system.

### Summary

The analysis presented here represents many years of research and development of the TMR+Spares architecture and provides some additional insight into the advantages and disadvantages of this type of architecture. Perhaps the most interesting finding is that the benefit of added spare resources diminishes after two spares in a system regardless of the radiation environment. This finding is important because it will reduce the complexity of systems implemented in the future eliminating the unnecessary use of precious logic resources. The study of the impact on voter reliability was the first time the reliability of a single-point-of-failure circuit component was considered separately from the redundant hardware. The impact such circuits had on system reliability was previously unquantified, and the finding that the area

of the voter in the 9-Tile Microblaze system had no negative impact on reliability was an important result. Knowing that the system is only as reliable as its least reliable component is important when considering what areas of a design require mitigation through TMR, and which areas are sufficiently small to be implemented in simplex. The scrubbing routine in a system was also demonstrated to be of paramount importance as it prevents the accumulation of faults. If a single-point-of-failure component is faulted the scrubber restores its functionality allowing the system to return to a healthy state.

REFERENCES CITED



- [1] J.L. Barth, C.S. Dyer, E.G. Stassinopoulos. Space, atmospheric, and terrestrial radiation environments. *IEEE Transactions on Nuclear Science*, 50(3):466–482, Jun 2003. doi:10.1109/TNS.2003.813131.
- [2] Andrew Holmes-Siedle, Len Adams. *Handbook of Radiation Effects*. Oxford University Press, USA, 2002.
- [3] Harry H. Heckman, Alice H. Armstrong. Energy spectrum of geomagnetically trapped protons. *Journal of Geophysical Research*, 67(4):1255–1262, 1962. doi:10.1029/JZ067i004p01255.
- [4] W. R. Binns, T. L. Garrard, P. S. Gibner, M. H. Israel, M. P. Kertzman, J. Klarman, B. J. Newport, E. C. Stone, C. J. Waddington. Abundances of ultraheavy elements in the cosmic radiation: Results from heao 3. *The Astrophysical Journal*, 346:997–1009, 1989.
- [5] NASA GSFC: The Van Allen Belts. Web. <http://image.gsfc.nasa.gov/poetry/tour/vanallen.html>.
- [6] C. Hu, S. Zain. NSEU Mitigation in Avionics Applications. Technical Report, Xilinx, Inc., 2010.
- [7] LET vs. Range in Si for 25 MeV SEE Beams. [http://cyclotron.tamu.edu/ref/LET\\_vs\\_Range\\_25.pdf](http://cyclotron.tamu.edu/ref/LET_vs_Range_25.pdf). Accessed March 4, 2014.
- [8] A. Keys, J. H. Adams, J. D. Cressler, M. C. Patrick, M. A. Johnson, R. C. Darty. Radiation hardened electronics for space environments (rhese) project overview. *Sixth International Planetary Probe Workshop*, 2008.
- [9] Clinton Francis Gauer. Radiation tolerant many-core computing system for aerospace applications. Master’s Thesis, Montana State University–Bozeman, 2010.
- [10] Jennifer Susan Hane. A fault-tolerant computer architecture for space vehicle applications [electronic resource]/by jennifer susan hane. Master’s Thesis, Montana State University-Bozeman, College of Engineering, 2012.
- [11] SD Card Association. SD Specifications Part 1 Physical Layer Simplified Specification. Technical Report, SD Group, 2010.
- [12] A. P. Lambert. Thermal-Mechanical Analysis of System-Level Electronic Packages for Space Applications. Master’s Thesis, Montana State University, 2012.
- [13] Tom Flatley. Advanced hybrid on-board science data processor-spacecube 2.0. *Earth Science Technology Forum*, 2010.

- [14] J. H. Adams Jr., R. Silberberg, C. H. Tsao. Cosmic ray effects on microelectronics, part i: The near-earth particle environment. Technical Report, Naval Research Laboratory, 1981.
- [15] J. A. Van Allen. The geomagnetically trapped corpuscular radiation. *Journal of Geophysical Research*, 64:1683–1689, 1959.
- [16] D. N. Baker, S. G. Kanekal, X. Li, S. P. Monk, J. Goldstein, J. L. Burch. An extreme distortion of the van allen belt arising from the ‘hallowe’en’ solar storm in 2003. *Nature*, 432:878–881, 2004.
- [17] D. P. Stern, M. Peredo. Radiation belts. Internet, November 2001.
- [18] Sung-Mo Kang, Yusuf Leblebici. *CMOS Digital Integrated Circuits Analysis & Design*. McGraw-Hill Science/Engineering/Math, 2002.
- [19] Gordon E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8), April 1965.
- [20] ICRU. *Linear Energy Transfer/Icru Report 16*. Intl Commission on Radiation, 1970.
- [21] A. Bar-lev. *Semiconductors and Electronic Devices*. Prentice-Hall International, Inc., 1984.
- [22] C. W. Gwyn. Model for radiation-induced charge trapping and annealing in the oxide layer of MOS devices. *Journal of Applied Physics*, 40(12):4886, 1969. doi:10.1063/1.1657309.
- [23] H. J. Barnaby. Total-ionizing-dose effects in modern CMOS technologies. *IEEE Transactions on Nuclear Science*, 53(6):3103–3121, Dec 2006. doi:10.1109/TNS.2006.885952.
- [24] F.W. Sexton. Destructive single-event effects in semiconductor devices and ICs. *IEEE Transactions on Nuclear Science*, 50(3):603–621, Jun 2003. doi:10.1109/TNS.2003.813137.
- [25] M.J. Gadlage, R.D. Schrimpf, J.M. Benedetto, P.H. Eaton, D.G. Mavis, M. Sibley, K. Avery, T.L. Turflinger. Single event transient pulse widths in digital microcircuits. *IEEE Transactions on Nuclear Science*, 51(6):3285–3290, Dec 2004. doi:10.1109/TNS.2004.839174.
- [26] S. E. Diehl, A. Ochoa, P. V. Dressendorfer, R. Koga, W. A. Kolasinski. Error analysis and prevention of cosmic ion-induced soft errors in static CMOS RAMs. *IEEE Transactions on Nuclear Science*, 29(6):2032–2039, Dec 1982. doi:10.1109/TNS.1982.4336491.

- [27] JA Appels, E Kooi, MM Paffen, JJH Schatorje, WHCG Verkuylen. Local oxidation of silicon and its application in semiconductor-device technology. *PHILIPS RESEARCH REPORTS.*, 25(2):118–132, 1970.
- [28] M.R. Shaneyfelt, P.E. Dodd, B.L. Draper, R.S. Flores. Challenges in hardening technologies using shallow-trench isolation. *IEEE Transactions on Nuclear Science*, 45(6):2584–2592, 1998. doi:10.1109/23.736501.
- [29] S. E. Kerns, B. D. Shafer, L. R. Rockett, J. S. Pridmore Jr. The design of radiation-hardened ics for space: a compendium of approaches. *Proceedings of the IEEE*, 76:1470–1509, 1988.
- [30] J. Gebelein. An approach to System-Wide Fault Tolerance for FPGAs. Kirchhoff-Institute for Physics 2009.
- [31] Akil K Sutton, Marco Bellini, John D Cressler, Jonathon A Pellish, Robert A Reed, Paul W Marshall, Guofu Niu, Gyorgy Vizkelethy, Marek Turowski, Ashok Raman. An evaluation of transistor-layout rhbd techniques for see mitigation in sigc hbts. *IEEE Transactions on Nuclear Science*, 54(6):2044–2052, 2007.
- [32] J. W. Gambles, G. K. Maki, S. R. Whitaker. Radiation hardening by design. *International Journal of Electronics*, 95(1):11–26, Jan 2008. doi:10.1080/00207210701799526.
- [33] E. H. Cannon, M. Cabanas-Holmen, J. Wert, T. Amort, R. Brees, J. Koehn, B. Meaker, E. Normand. Heavy ion, high-energy, and low-energy proton see sensitivity of 90-nm rhbd SRAMs. *IEEE Transactions on Nuclear Science*, 57:3493–3499, 2010. doi:10.1109/tns.2010.2086482.
- [34] R. E. Lyons, W. Vanderkulk. The use of triple-modular redundancy to improve computer reliability. *IBM Journal of Research and Development*, 6(2):200–209, Apr 1962. doi:10.1147/rd.62.0200.
- [35] B. Pratt, M Caffrey, P. Graham, K. Morgan. *Improving FPGA Design Robustness with Partial TMR*. IEEE Reliability Physics Symposium Proceedings, March 2007.
- [36] N. Rollins, M Caffrey, P. Graham. Evaluating tmr techniques in the presence of single event upsets. *Military and Aerospace Programmable Logic Devices International Conference*, 2003.
- [37] F. Lima, L. Carro, R. Reis. *Designing fault tolerant systems into SRAM-based FPGAs*, pages 650–655. Institute of Electrical and Electronics Engineers, 2003. doi:10.1109/DAC.2003.1219099.

- [38] Brian Pratt, Michael Caffrey, James F. Carroll, Paul Graham, Keith Morgan, Michael Wirthlin. Fine-grain SEU mitigation for FPGAs using partial tmr. *IEEE Transactions on Nuclear Science*, 55(4):2274–2280, Aug 2008. doi:10.1109/TNS.2008.2000852.
- [39] Konstantinos Kyriakoulakos, Dionisios Pnevmatikatos. *A novel SRAM-based FPGA architecture for efficient TMR fault tolerance support*, pages 193–198. Institute of Electrical and Electronics Engineers, Aug 2009. doi:10.1109/FPL.2009.5272319.
- [40] E.F. Moore, C.E. Shannon. Reliable circuits using less reliable relays. *Journal of the Franklin Institute*, 262(3):191–208, Sep 1956. doi:10.1016/0016-0032(56)90559-2.
- [41] J. M. Benedetto, P. H. Eaton, D. G. Mavis, M. Gadlage, T. Turflinger. Digital single event transient trends with technology node scaling. *IEEE Transactions on Nuclear Science*, 53(6):3462–3465, Dec 2006. doi:10.1109/TNS.2006.886044.
- [42] Processors: 100 years of space operation. Technical Report, BAE Systems Inc.
- [43] Carl Carmichael. Triple module redundancy design techniques for virtex fpgas. *Xilinx Application Note XAPP197*, 1, 2001.
- [44] F Lima Kastensmidt, Luca Sterpone, Luigi Carro, M Sonza Reorda. On the optimal design of triple modular redundancy logic for sram-based fpgas. *Proceedings of the conference on Design, Automation and Test in Europe-Volume 2*, pages 1290–1295. IEEE Computer Society, 2005.
- [45] Zhong-Ming Wang, Li-Li Ding, Zhi-Bin Yao, Hong-Xia Guo, Hui Zhou, Min Lv. The reliability and availability analysis of seu mitigation techniques in sram-based fpgas. *Radiation and Its Effects on Components and Systems (RADECS), 2009 European Conference on*, pages 497–503. IEEE, 2009.
- [46] Patrick S Ostler, Michael P Caffrey, Derrick S Gibelyou, Paul S Graham, Keith S Morgan, Brian H Pratt, Heather M Quinn, Michael J Wirthlin. Sram fpga reliability analysis for harsh radiation environments. *Nuclear Science, IEEE Transactions on*, 56(6):3519–3526, 2009.
- [47] Xilinx Inc. Spartan-6 FPGA Configuration User Guide. Technical Report UG380 v.2.3, Xilinx, July 2011.
- [48] T. Buerkle, B.J. LaMeres, T. Kaiser, E. Gowens, L. Smoot, T. Heetderks, K. Schipf, L. Clem, S. Schielke, R. Luhr. Ionizing radiation detector for environmental awareness in fpga-based flight computers. *Sensors Journal, IEEE*, 12(6):2229–2236, June 2012. doi:10.1109/JSEN.2012.2186288.

- [49] T. M. Buerkle. Ionizing Radiation Detector for Environment Awareness in FPGA-based Flight Computers. Master's Thesis, Montana State University, 2012.
- [50] B. LaMeres, T. Kaiser, E. Gowens, T. Buerkle, J. Price, K. Helsley, B. Peterson, R. Ray. Position sensitive radiation detector integrated with an fpga for radiation tolerant computing. *Sensors, 2010 IEEE*, pages 208–213, Nov 2010. doi:10.1109/ICSENS.2010.5690418.
- [51] R. J. Weber. *Reconfigurable Hardware Accelerators for High Performance Radiation Tolerant Computers*. Doctoral Dissertation, Montana State University, 2014.
- [52] CS Dyer, PR Truscott. Cosmic radiation effects on avionics. *Microprocessors and Microsystems*, 22(8):477–483, 1999.
- [53] E Normand, TJ Baker. Altitude and latitude variations in avionics seu and atmospheric neutron flux. *Nuclear Science, IEEE Transactions on*, 40(6):1484–1490, 1993.
- [54] D McMurtrey, K Morgan, B Pratt, M Wirthlin. Estimating tmr reliability on fpgas using markov models. *BYU Dept. Electr. Comput. Eng., Tech. Rep*, 2006.
- [55] Richard L. Scheaffer, James T. McClave. *Probability and Statistics for Engineers*. Duxbury Press, 1994.
- [56] Charles M. Grinstead, J. Laurie Snell. *Introduction to Probability*. American Mathematical Society, 1997.
- [57] IBM and the Space Shuttle. Web. [http : //www – 03.ibm.com/ibm/history/exhibits/space/space\\_shuttle.html](http://www-03.ibm.com/ibm/history/exhibits/space/space_shuttle.html).
- [58] Kishor Shridharbhai Trivedi. *Probability and Statistics With Reliability, Queuing and Computer Science Applications*. Prentice Hall, 1982.
- [59] H. Quinn, K. Morgan, P. Graham, J. Krone, M. Caffrey. Static proton and heavy ion testing of the xilinx virtex-5 device. *Radiation Effects Data Workshop, 2007 IEEE*, Volume 0, pages 177–184, July 2007. doi:10.1109/REDW.2007.4342561.
- [60] P Robinson, W Lee, R Aguerro, S Gabriel. Anomalies due to single event upsets. *Journal of Spacecraft and Rockets*, 31(2):166–171, 1994.
- [61] J. D. Engel, M. J. Wirthlin, K. S. Morgan, P. S. Graham. Predicting On-Orbit Static Single Event Upset Rates in Xilinx Virtex FPGAs. Technical Report, Brigham Young University and Los Alamos National Labs, 2006.

- [62] Allan J Tylka, JH Adams, Paul R Boberg, Buddy Brownstein, William F Dietrich, Erwin O Flueckiger, Edward L Petersen, Margaret A Shea, Don F Smart, Edward C Smith. Creme96: A revision of the cosmic ray effects on microelectronics code. *Nuclear Science, IEEE Transactions on*, 44(6):2150–2160, 1997.