SPATIAL AND TEMPORAL RELIABILITY FACTORS OF CMOS MEMORIES IN PULSED SINGLE-EVENT RADIATION ENVIRONMENTS

By

Daniel Scott Vibbert

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

May 13th, 2022

Nashville, Tennessee

Approved:

W. Timothy Holman, Ph.D.

Lloyd W. Massengill, Ph.D.

Jeffrey S. Kauppila, Ph.D.

Robert A. Reed, Ph.D.

Steven Csorna, Ph.D.

Kevin M. Warren, Ph.D.

© Copyright by Daniel S. Vibbert 2022

All Rights Reserved

ACKNOWLEDGEMENTS

This development of this dissertation is heavily owed to the faculty, graduate students, and staff at the Institute of Defense and Space Electronics (ISDE). This work was sponsored by the Defense Threat Reduction Agency and would not have been possible without their financial support. I would also like to acknowledge the following tool vendors for their university programs which granted access to invaluable design tools and software: Cadence Design Systems, Inc; Mentor Graphics; and Synopsis, Inc.

I would also like to thank my committee members for mentoring me throughout my graduate career and improving my professional skills. Dr. Tim Holman, my graduate advisor, broadened my horizons in electrical engineering and helped me every time I became discouraged. Dr. Lloyd Massengill kept our circuits group at ISDE focused on our academic journeys. I think that I can speak for all of my colleagues by stating that we could not have asked for a better leader. Dr. Jeff Kauppila was always a dependable source of insight and wisdom, and my colleagues and I are grateful for his technical excellence and endurance. I'd like to thank Dr. Robert Reed for helping me learn the fundamentals of radiation effects. I'd like to thank Dr. Steven Csorna for representing the Physics department on my committee and for his insightful discussions. Finally, I'd like to thank Dr. Kevin Warren for his intense technical feedback and critical evaluation of the most difficult engineering and physics topics throughout this work.

The experimental data presented in Chapter V was made possible by contributions of the graduate students and faculty who performed the necessary work to collect the heavy ion beam test data at the Texas A&M University Cyclotron. Sean Vibbert and

Joseph D’Amico collected the data at the test facility. Sean Vibbert, Chris Watkins, Joseph D’Amico, Dr. Jeff Kauppila, and Tim Haeffner assisted in the design and tape-out of the test chip. Dr. Enxia Zhang bonded the test chip, and Dr. Andrew Sternberg assisted in the design of the test board and the experiment.

Most importantly, I must thank my wonderful family for their prayers, support, and strength. Jared and Sean, my two brothers, always give me hope that better days are coming. My parents Terry and Tracey never let me give up and always did what was best for me. Finally, I’d like to thank my loving wife Lorraine for her endless faith and patience which blessed my work and made all of it worthwhile.

TABLE OF CONTENTS

Acknowledgements iii

Table of Figures ix

Acronyms xv

Chapter I Introduction 1

I.A) Motivation 1

I.B) Definition of Multiple-bit Upset 2

I.C) Dissertation Organization 3

Chapter II Single-Event Effects Background 5

II.A Overview of Radiation Effects on Microelectronics 5

II.A.1) Multi-Scale View of Microelectronic Circuits and Radiation Effects 5

II.A.2) Categories of Radiation Effects 7

II.B Single-Event Transients and Single-Event Upsets 8

II.B.1) Radiation Transport and Material Interaction with Ionizing Radiation 8

II.B.2) Single-Event Transients 12

II.B.3) Single-Event Upsets 15

II.B.4) Multiple-Node Upsets and Multiple-Bit Upsets 16

II.C Radiation-Hardening-by-Design Techniques 17

II.C.1) Bit-Level Hardening Strategies: DICE Latch 17

II.C.2) Word-Level Hardening: Error-Detection-and-Correction Codes 18

II.C.3) Error Accumulation Prevention: Scrubbing 18

II.C.4) Spatial-MBU Prevention: Bit-Interleaving 19

II.D Random Processes applied to Single-Event Effects 19

II.E Quantification of Reliability 22

II.E.1) Experiments, One-minus Reliability, and Number of Nine's Reliability 22

II.E.2) Reliability of Digital Memories 24

Chapter III Temporal-Spatial Model for Reliability Against Multiple-Bit Upset 26

III.A Problem Definition 26

III.B Related Work 28

III.C Spatial-Temporal Reliability Model 29

III.C.1) Constraints and Assumptions of the Spatial-Temporal Reliability Model .. 29

III.C.2) Derivation of the Spatial-Temporal Reliability Model 30

III.C.3) Analysis of the Spatial-Temporal Reliability Model 36

III.C.4) Reliability Against SBU, XMBU, and TMBU 37

III.C.5 Determination of Cross-Sections 41

III.D Simulation-Based Evidence for Hypotheses 43

III.E Applicability to Codes beyond SEC-DED 45

III.F Impact of Spatial-Temporal Reliability Model 49

Chapter IV Pulse Arrival-Time Invariant Reliability Model for Temporal Multi-Bit Upset 51

IV.A Problem Definition 51

IV.B Related Work: The Saleh Model 52

IV.D Derivation of the PATI Model 55

IV.F Simplified TMBU Model 59

IV.H Conclusions 60

Chapter V Simulation of Multi-Node Upset in Spatially-Redundant Circuits 62

V.A Problem Definition 63

V.A.1) Generalized Description of Single-Event Effect Vulnerability Analysis 63

V.A.2) Description of Subject Circuit: Sub-50 nm SOI DICE Latch 66

V.B Simulation Technique used in this Work 68

V.B.1) 3D Circuit Construction and Radiation Transport Setup 69

V.B.2) Monte-Carlo Radiation Transport with MRED 71

V.B.3) Dual-Node Single-Event-Transient SPICE Simulation 73

V.C Simulation Results 75

V.D Experimental Evidence 78

V.D.1) Physical Implementation of the DICE Latch 78

V.D.2) Test Setup 79

V.D.3) Results 79

V.E Similar Techniques used in Radiation Effects Literature 80

V.F Implications and Conclusions 81

Chapter VI Conclusions 82

Appendix A Monte-Carlo Simulator used in Chapter III 83

References... 86

TABLE OF FIGURES

Figure Page

1. Different levels of physical detail in the art of microelectronic circuits along with their relevant units of distance. 6

2. Summary of radiation effects versus scale. The lists of effects and consequences are not exhaustive. 7

3. Ion passing through a silicon-on-insulator MOSFET. Minority charge carriers that are produced in the different silicon regions are shown. 9

4. Neutron colliding with the high-Z material with a via in the V1 layer of an integrated circuit's back-end-of-line. The neutron trajectory is shown with the magenta arrow. The three green arrows represent the trajectories of the ionizing secondary particles. 9

5. A CMOS logic inverter shown in (a) has an input of logic-1 and a correct output of logic-0. The physical representation of a vulnerable PMOS transistor is shown in (b). The PMOS transistor is vulnerable to SET because there is a large electric potential across its source and drain terminals. In (c), an ion penetrates the PN junction at the drain and generates charge. In (d), the electric field pushes the excess carriers across the junction, resulting in single-event transient current, which is translated back onto the schematic in (e). An errant logic-1 will appear at the output for a short time. 12

6. A CMOS logic inverter shown in (a) has an input of logic-0 and a correct output of logic-1. The physical representation of a vulnerable NMOS transistor is shown in (b). The NMOS transistor is vulnerable to SET because there is a large electric potential across its source and drain terminals. In (c), an ion penetrates the PN junction at the drain and generates charge. In (d), the electric field pushes the excess carriers across the junction, resulting in single-event transient current, which is

translated back onto the schematic in (e). An errant logic-0 will appear at the output for a short time. 13

7. Schematic of a DICE latch implementation, adapted from [2]. When CLK is 1, the latch is in transparent mode. When CLK is 0, the latch is in holding mode. The four storage nodes are labeled α, β, γ and δ. In holding mode, each storage node is reinforced by the other nodes. If one node is corrupted by an SET, the other three nodes will work together to correct the one that was struck. 17

8. Relationship between flux, cross-section, and event rate. In (a), the cross-section is the physical area of the particle detector's aperture. In (b), the cross-section is a single-event upset cross-section, which describes the probability of an SEU occurring if the circuit is exposed to a fluence of one ion per square-centimeter. 21

9. Comparison of two calculations for the reliability of a memory where each individual word of memory has a reliability of 0.99999999, or 8 nines of reliability. The two calculations agree for memories with a reliability of at least 0.9. 25

10. Comparison of XMBU and TMBU. The purple particle trajectory represents neutrons while the green particle trajectories represent ions. (a) and (b) represent examples of spatial MBU while (c) and (d) represent examples of temporal MBU. 27

11. Venn diagram showing the relationship between the events S, X, and T as well as the intersection of X and T. S represents all possible outcomes and is formally known as the sample space. 34

12. Monte-Carlo simulation results for a memory word with the parameters shown. The results were calculated by performing multiple simulations, recording which ones resulted in outcomes $S, X, T, or\ M$ and calculating the ratio. 10 million simulations were performed. All three variants of the XTR model produced results that are too similar to distinguish on the logarithmic scale. 43

13. Percent difference for estimation of RM between the XTR model and Monte-Carlo simulation for each of the three model levels. The difference between the three models is only significant when no scrubbing is applied, and spatial multiple-bit upsets are relatively frequent. 44

14. Representation of blind deterministic scrubbing with period τ dividing the radiation pulse duration into smaller time intervals. The chance of not encountering a TMBU during the whole radiation pulse is the product of the chances of not encountering a TMBU during any of the constituent time intervals. 54

15. Evaluation of Equation (49) and Equation (47) plotted versus scrub interval τ for various bit-upset rate parameters λb, which have units of Hz. The red line marks the pulse duration T, which is 1 millisecond. 54

16. Revision of Figure 12 which accounts for the possibility of the radiation pulse arriving out of phase with the periodic scrubbing intervals. The size of the intervals *t0* and *t1* will be random and depend on when the radiation pulse arrives. 56

17. Four different cases which describe how the scrubbing interval interacts with the radiation pulse. The vertical black lines represent the scrubbing events. The orange rectangle represents the beginning and end of the radiation pulse. Beneath the orange rectangle, there are rectangles which represent the radiation pulse duration being broken up into smaller intervals because of periodic scrubbing. The blue intervals are the “head” of the radiation pulse, the green intervals are the “tail” of the radiation pulse, and the red intervals, which are equal to τ, are the body of the radiation pulse. Beneath each of the four cases, an equation is shown represents the overall reliability against TMBU. The colored highlights in the equation correspond to the colored intervals shown on the timeline. 57

18. Comparison of the PATI model and the Saleh model. 59

19. Comparison of the PATI and Saleh model, along with the simplified TMBU model. The simplified TMBU model consists of two straight lines which intersect at a 45° angle. 60

20. Flowchart showing a variety of physical and electrical analyses which operate at different levels of detail to calculate a single-event effect cross-section. Note that the arrow connecting charge transport simulation to transient circuit simulation is bidirectional. 63

21. Schematic of DICE latch studied in this chapter. There are four storage nodes driven by inverters which interlock when CLK = 0. 66

22. Vulnerable transistors are highlighted as **A**, **B**, **C**, **D** when the DICE latch is in the D = 1, Q = 1 state. If two or more of these transistors are struck simultaneously by a single-event, then the spatial redundancy could fail, and the latch could upset. 66

23. Three layout implementations of the sub-50nm DICE latch. The front-end-of-line is the same for all three circuits, but the interconnect gives the transistors different purposes. The locations of the vulnerable transistors from Figure 19 are highlighted. 67

24. Flowchart that summarizes the technique used in this work. The circuit's XMNU cross-section is calculated by combining the results of SPICE simulation and radiation transport simulation. 68

25. 3D model of the DICE latch studied in this work. Silicon-dioxide is used as a substitute for back-end-of-line dielectric because of its similar physical properties. 69

26. Visualization of the virtual boundary which surrounds the 3D target. All simulated particles exist within the purple bounding sphere shown. When a particle is generated during Monte-Carlo simulation, a random coordinate on the randomization area is chosen, and that point is projected onto the bounding sphere. 70

27. In (a), the virtual fluence has a tilt and roll of 0°, otherwise known as direct incidence. In (b), the virtual fluence roll angle is shown with respect to layout orientation from a top-down view. In (c), the virtual fluence is shown in 3D with

the tilt of 60° and a roll of 30°. In (d-f), an illustration of a device-under-test (DUT) is shown. The DUT is mounted on a green circuit board. (d) shows normal incidence, (e) shows a tilt of 60°, while (f) shows a tilt of 60° and a roll of 30°. In (b-f), the gray arrow points in the same direction with respect to the vertical polysilicon gates in the circuit layout. 71

28. Dual-node critical charge for four different pairs of transistors from Figure 19. Red dots indicate that the DICE latch was upset, while blue dots indicate that the DICE latch did not. Other transistor combinations such as (A, D) or (B, C) were did not induce any upsets. 73

29. Estimations of the DICE latch's SEU cross-section for each of the three layout variants from Figure 20 when exposed to unidirectional, monoenergetic neutrons at various tilt and roll angles. Estimates which used the same roll angle are grouped by the vertical black lines. The error bars represent one standard deviation from the nominal datapoint.. 75

30. Estimations of the DICE latch's SEU cross-section for each of the three layout variants, if only XMNUs on the transistor pairs CD and AC are considered. No XMNUs were detected in the columns which don't have a datapoint................. 76

31. Estimations of the DICE latch's SEU cross-section for each of the three layout variants, if only XMNUs on the transistor pairs AB and BD are considered. No XMNUs were detected in the columns which don't have a datapoint................. 77

32. Three layout variants of the DICE latch that were tested. These three layouts correspond to the layouts from Figure 20. The vulnerable transistor locations are highlighted. 79

33. Experimental data which represents that SEU cross-section of the three variants of the DICE latch. The DICE latches were exposed to unidirectional 24.8 MeV/u nitrogen ions at a variety of tilt and roll angles. Measurements which have the same roll are grouped by vertical lines. If no data points are shown, then no upsets were

detected at the tilt and roll angle. Fluence was not normalized to tilt angle. Error bars represent one standard deviation. 80

ACRONYMS

RHBD	Radiation-hardening-by-design
MBU	Multiple-bit upset
CMOS	Complementary Metal-Oxide Semiconductor
MOSFET	Metal-Oxide Semiconductor Field Effect Transistor
SEC	Single Error Correction
SEU	Single-Event Upset
JEDEC	Joint Electron Device Engineering Council
IEEE	Institute of Electrical and Electronic Engineers
XMBU	Spatial Multiple-bit upset
MNU	Multiple-node upset
XMNU	Spatial multiple-node upset
TMBU	Temporal Multiple-bit upset
MRED	Monte-Carlo Radiative Energy Deposition
TID	Total Ionizing Dose
EHP	Electron-Hole pair
SOI	Silicon-on-insulator
NFET	N-type Field Effect Transistor
PFET	P-type Field Effect Transistor
SET	Single-event transient
TCAD	Technology Computer Aided Design
SPICE	Simulation Program with Integrated Circuit Emphasis
MNU	Multiple-node upset
DICE	Dual-interlocking cell
EDAC	Error-detection-and-correction
SECDED	Single-error-correction/double-error-detection
FPGA	Field-programmable gate array
PDF	Probability density function
SRAM	Static random access memory
XTR	Spatial temporal reliability

XTOC	Spatial-temporal orthogonal causality
DECTED	Double error correction/triple error detection
SEC-DAED-DED	Single error correction/double adjacent error correction/double error detection
PATI	Pulse-arrival time-invariant
CAD	Computer assistant design
MRICE	Monte-Carlo Radiation Transport with Integrated Circuit Emphasis
DUT	Device under test
ACCRE	Advanced Computing Center for Research and Education

CHAPTER I

INTRODUCTION

The miniaturization of microelectronic circuits which has continued over the recent decades poses a continually evolving challenge to develop circuits which can operate reliably in the presence of ionizing radiation. Modern integrated circuit use transistors with feature sizes that are well below 50 nanometers in length. These sub-50nm integrated circuits play a nearly ubiquitous role in today's digitized society but are more vulnerable to energetic particles of radiation than their predecessors [1]. Effects caused by single particles of radiation are called a single-event effects (SEE). A broadly studied single-event effect, single-event upset (SEU), results in the corruption of digital information and is the underlying cause of many system-level reliability concerns. Understanding the physical mechanisms which cause SEUs has allowed for the development of a variety of design strategies and techniques which either prevent SEUs from occurring or prevent the loss of critical information when SEUs do occur.

I.A) Motivation

Redundancy is a direct approach to addressing reliability concerns in many scenarios and is especially applicable to single-event effects. Several radiation-hardening-by-design (RHBD) techniques employ some form of redundancy that is spatial, temporal, or logical to reduce the probability that an unacceptable circuit malfunction will occur during radiation exposure. Examples of these RHDB techniques include but are not limited to, triple modular redundancy (TMR), dual-interlocking cells (DICE) [2], dual-interlocking

logic (DIL) [3], and error detection and correction (EDAC) codes. While each of these techniques reduce the probability that digital information will be destroyed by SEU, quantifying their improvements to reliability is difficult. This is because the occurrence of single-event effects must be modeled as a stochastic random process which is influenced by a wide variety of parameters. Quantifying the reliability contributed by RHBD techniques is important, however, because implementing any RHBD technique will incur a performance penalty in the form of increased power consumption, reduced circuit density, or reduced operating speeds.

There is significant financial and logistical motivation to develop computational reliability estimation techniques because it reduces the need to design, fabricate, test, and redesign radiation-hardened circuits. Furthermore, measuring a hardened circuit's reliability through experiment is sometimes impractical or impossible due to the extreme or exotic nature of the radiation environment of interest. In these scenarios, circuit designers must rely exclusively on computational reliability estimation techniques which can extrapolate reliability estimations from available radiation environments to unavailable radiation environments. Facilitating the estimation of circuit reliability against single-event upsets is the goal of the work presented in this dissertation. The work includes mathematical models and simulation techniques which help estimate the reliability of memory circuits against radiation-induced multiple-bit upset (MBU) in pulsed radiation environments.

I.B) Definition of Multiple-bit Upset

Radiation-induced MBU is a well-studied single-event effect because it defeats single-error correction (SEC) codes, which only protect the contents of a data block if a single bit

corrupted. According to a published Joint Electron Device Engineering Council (JEDEC) standard [4], an MBU is defined as a single particle of radiation causing multiple bit errors in the same word or block of memory. In some rare cases, peer-reviewed journal articles associated with the Institute of Electrical and Electronic Engineers (IEEE) use the term "temporal MBU" [5] or "temporal double-bit error" [6], to refer to multiple particles inducing at least one bit error in the same block of memory. More commonly, however, this phenomenon is referred to as "SEU accumulation".

In this dissertation, both the single-particle and multiple-particle case will be jointly investigated; it will not be assumed that only one of them is a pressing concern. To facilitate the discussion of both causes of multiple-bit errors, a new terminology will be adopted. Spatial multi-bit-upset (XMBU) will refer to the case where one particle corrupts multiple bits and temporal multi-bit-upset (TMBU) will refer to the case where multiple particles each upset at least bit. MBU will refer to both XMBU and TMBU. This terminology distinguishes the two underlying mechanisms which result in multiple bit errors while acknowledging that the symptom resulting from either is the same. The mathematical models and simulation techniques presented in this dissertation focus on evaluating the probability of either XMBU or TMBU occurring.

I.C) Dissertation Organization

In chapter II of this dissertation, the relevant background information necessary to discuss flux-dependent MBU is presented. This includes an overview of radiation effects which increases in detail as the discussion approaches single-event effects, single-event upset, and finally multi-bit upset. RHBD techniques pertinent to the mitigation of XMBU

and TMBU are also discussed. Finally, essential information regarding random processes and reliability is given.

In chapter III, a mathematical model which can be used to combine spatial and temporal MBU is presented and derived. Monte-Carlo simulation results are shown which provide evidence for the model's accuracy. In chapter IV, advancements to reliability calculations with respect to TMBU are presented and derived. These advancements are primarily useful to computing the reliability of memories with SEC codes which are subjected to a high-flux, pulsed radiation environment. In chapter V, a simulation technique which can be used to extract a circuit's reliability to XMBU, or spatial multiple-node upset (XMNU) is presented. In chapter VI, conclusions will be drawn from the presented work.

CHAPTER II

SINGLE-EVENT EFFECTS BACKGROUND

This chapter contains prerequisite material for the original work presented in chapters III, IV, and V. First, an overview of radiation effects is presented, followed by a more detailed discussion of single-event effects (SEE) and single-event upset (SEU). Next, radiation-hardening-by-design (RHBD) techniques which are used to mitigate the threat of SEU are discussed. Finally, some prerequisite theory on random processes and reliability which is relevant to the study of SEU is presented.

II.A Overview of Radiation Effects on Microelectronics

The intent of this section is to offer the reader a situational awareness which shows how the work presented in the following chapters fits into the broader field of study in which it resides. To focus on the radiation effects which are prerequisite to understanding the work presented in the following chapters, proceed to section *II.B* which focuses on single-event transients and single-event upsets.

II.A.1) Multi-Scale View of Microelectronic Circuits and Radiation Effects

In the absence of radiation, microelectronic circuit design is an immensely complicated subject which requires a lifetime of practice and study to master. Like many technical subjects, it is useful to divide microelectronic circuits into a hierarchy in which each level

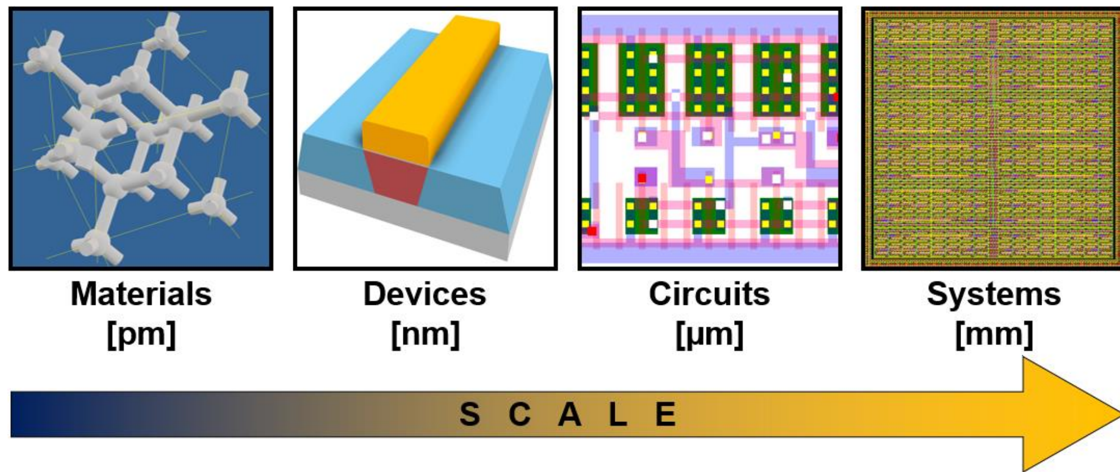
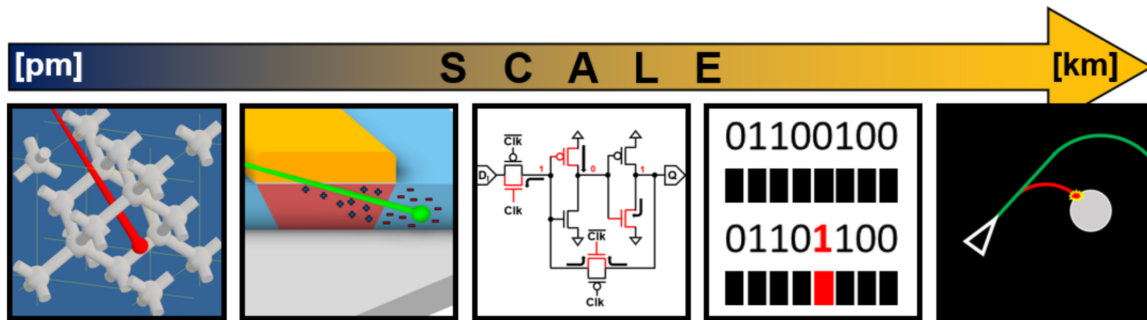


Figure 1: Different levels of physical detail in the art of microelectronic circuits along with their relevant units of distance.

can be studied independently of the other levels. In this case, these levels correspond to different physical scales, as shown in Figure 1. It is usually possible, and often required, for individual electrical engineers to specialize in one level without possessing a comprehensive understanding of the other levels. This scale-based segregation of expertise occurs naturally in engineering practice, and as a result, it is necessary for experts in a particular subdiscipline of microelectronic circuit design to know how to communicate their results, theories, and designs in ways that are useful to experts of the adjacent subdisciplines.

The study of radiation effects on microelectronic circuits is unique from other electrical engineering disciplines because radiation affects circuits at every hierarchical level, from material to system-level behavior, as depicted in Figure 2. The notional map shown in Figure 2 is not comprehensive, but it should give the reader an appreciation for the magnitude of the challenge that radiation poses to microelectronics applications.



Material Interaction	Device Effects	Circuit Effects	System Effects	Application Consequences
• Ionization • Displacement Damage • Spallation • Fission • Non-ionizing energy loss	• Single-Event Transients • Threshold Voltage Shift • Leakage Current • Single-Event Burnout • Single-Event Latch-up • Single-Event Gate Rupture	• Single-Event Upset • Performance Degradation • Increased Power Consumption • Circuit Destruction • Rail Span Collapse	• Data Corruption • System Unavailability • Incorrect Behavior	• Mission Failure • Loss of Life • Financial Loss • Downtime

Figure 2: Summary of radiation effects versus scale. The lists of effects and consequences are not exhaustive.

II.A.2) Categories of Radiation Effects

While this dissertation focuses on single-event upsets, it is useful to possess an awareness of other documented radiation effects and categorize them. One way to do this is to divide them into ionizing and non-ionizing radiation effects categories. Non-ionizing radiation effects involve the damage of a microelectronic circuit's constituent material while ionizing radiation effects involve the generation and manipulation of electrical charge in the microelectronic circuit. Another way radiation effects might be categorized is by their temporal or spatial manifestation. This results in two categories, total-dose effects, and single-event effects. Total-dose effects occur when multiple particles each contribute a small, accumulating amount of charge, energy, or damage. Total-ionizing-dose (TID) is an example of such an effect, which has traditionally been associated with device

leakage and threshold voltage shift [7]. The miniaturization of CMOS technology has lessened the threat of TID [7], [8].

Single-event effects (SEE), on the other hand, have become a greater threat as transistor dimensions continue to shrink and are the radiation effects category of interest in this dissertation. Single-event effects occur when a single particle impacts a part of the integrated circuit and creates a spatially and temporally localized effect. The ionizing single-event effect mechanisms which cause single-event upset will be elaborated upon in the next section. Some other single-event effects which constitute important reliability concerns but are not elaborated upon further in this dissertation include single-event latch-up [9] [10], single-event burnout, and single-event gate rupture [11].

II.B Single-Event Transients and Single-Event Upsets

II.B.1) Radiation Transport and Material Interaction with Ionizing Radiation

Ionizing single-event effects are caused by charged particles and neutrons. Electrically charged radiation particles gradually lose energy when they travel through solids because their electric fields interact with those of the solid's constituent atoms. Most of the energy lost by these charged particles is transferred to the surrounding electrons. In semiconductors, these energized electrons will jump from the lower-energy valence band to the higher-energy conduction band, producing an electron-hole pair (EHP). Electrons and holes are the free charge carriers which allow electronics to operate, but a sudden

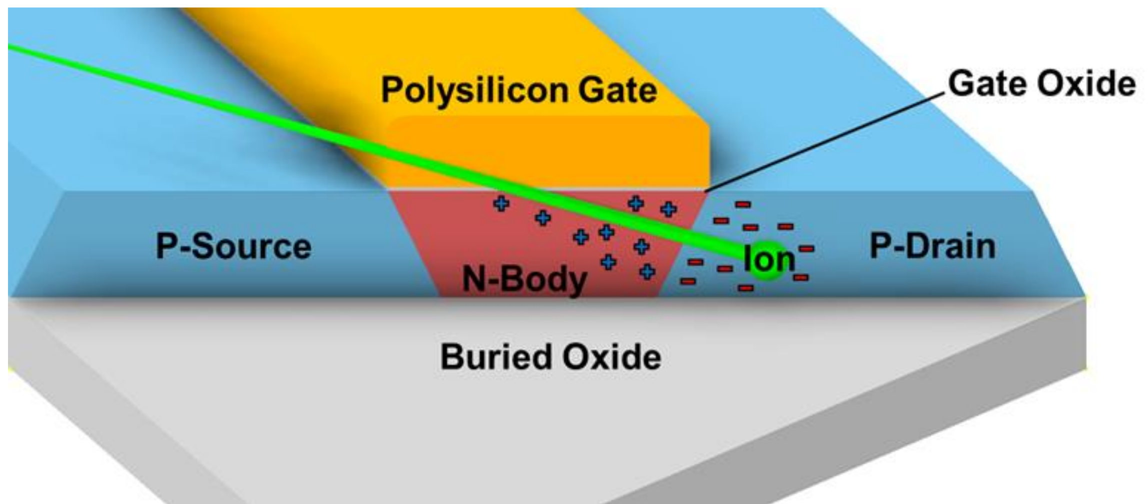


Figure 3: Ion passing through a silicon-on-insulator MOSFET. Minority charge carriers that are produced in the different silicon regions are shown.

injection of them can disturb a circuit's intended operation, as will be discussed in section II.B.2. A depiction of an ion producing a wake of EHPs is shown in Figure 3.

Despite being electrically neutral and non-ionizing, sufficiently energetic neutrons are capable of indirectly generating charge in semiconductors when they collide with material nuclei and produce secondary ionizing particles, as described in [12], [13], and [14]. The neutron's lack of coulombic interaction allows it to penetrate semiconductor material more

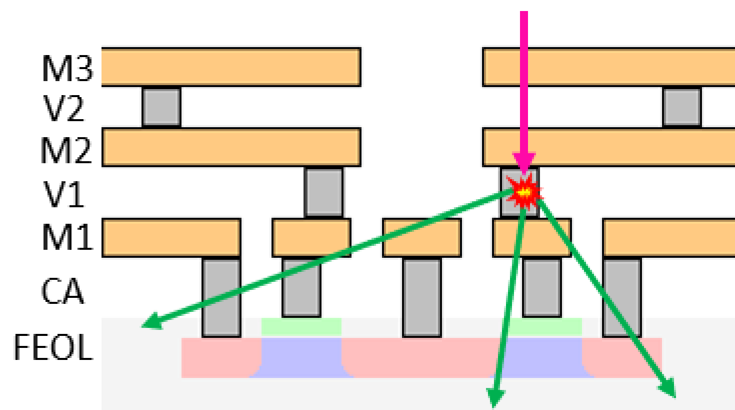


Figure 4: Neutron colliding with the high-Z material with a via in the V1 layer of an integrated circuit's back-end-of-line. The neutron trajectory is shown with the magenta arrow. The three green arrows represent the trajectories of the ionizing secondary particles.

efficiently than ions can. When a neutron does collide with a material nucleus, three different reactions can occur, depending on the mass of the nucleus and the energy of the neutron. Neutrons can split the nucleus into smaller fragments (also known as fission), it can dislodge individual nucleons from the nucleus (spallation), or it can displace the entire nucleus from the material completely. It has been shown in [15] that some neutrons have an enhanced probability of producing heavy nucleus fragments when they pass through the metal interconnection layers of the circuit. This phenomenon is depicted in Figure 4.

Because the interactions of radiation with matter in microelectronics are difficult to assess experimentally, a number of computer codes and simulators have been developed to extrapolate the available physical data to specific scenarios that microelectronic engineers require. Radiation transport simulation calculates the trajectories of individual particles of radiation as they travel through the integrated circuit material. These simulations are necessary to capture the particle-material interactions that result in single-event effects. An anthology of radiation transport tools developed before June 2013 is given in [16]. Two radiation transport tools that are relevant to this dissertation proposal are described here.

Geometry and Tracking (GEANT) is an advanced radiation transport code which allows the user to construct a 3D world which is filled with a variety of physical and hypothetical materials. The most recent version of GEANT as of this writing is GEANT4 [17]. GEANT4 is distributed as a library of C++ classes which can be used to compile a physics-based simulator tailored to the end-user's needs. A compiled GEANT4 program is like a virtual particle physics experiment in which particles of radiation are injected into the 3D world. These virtual particles can evolve over time as they travel through the world and undergo physically realistic interactions based on physical models that are either

provided by the GEANT4 collaboration and distributed with the source code or custom models developed by the end user. Detector-based calorimetry is an especially useful feature of GEANT4, which is useful for monitoring energy deposition and charge deposition of ionizing particles.

Monte-Carlo Radiative Energy Deposition (MRED) is used extensively in Chapter V and has seen much use by the radiation-effects community [18], [19], [20], [21]. MRED combines radiation transport codes of GEANT4 with the physical models which are most pertinent to the reliability of microelectronics. Like Geant4, MRED is a Monte-Carlo simulation tool, which is highly suited to radiation transport because of the stochastic nature of radiation and because of the analytically intractable number of possible outcomes that can result from a single, pseudorandom, ionizing particle's collision with an integrated circuit structure.

II.B.2) Single-Event Transients

The direct consequence of an energetic ion traveling through a semiconductor is that it generates a relatively large amount of EHPs in its wake. Electron-hole pairs consist of two free carriers which have a limited lifetime before they recombine. During their lifetime, the excess charge carriers will diffuse, resulting in a diffusion current. The presence of an electric field, like the one in a P-N junction in a transistor, will also cause the charge carriers to drift. When the charge deposited by an ion produces a current that disturbs the voltage of a circuit node, it is called a single-event transient (SET). The strongest and most severe

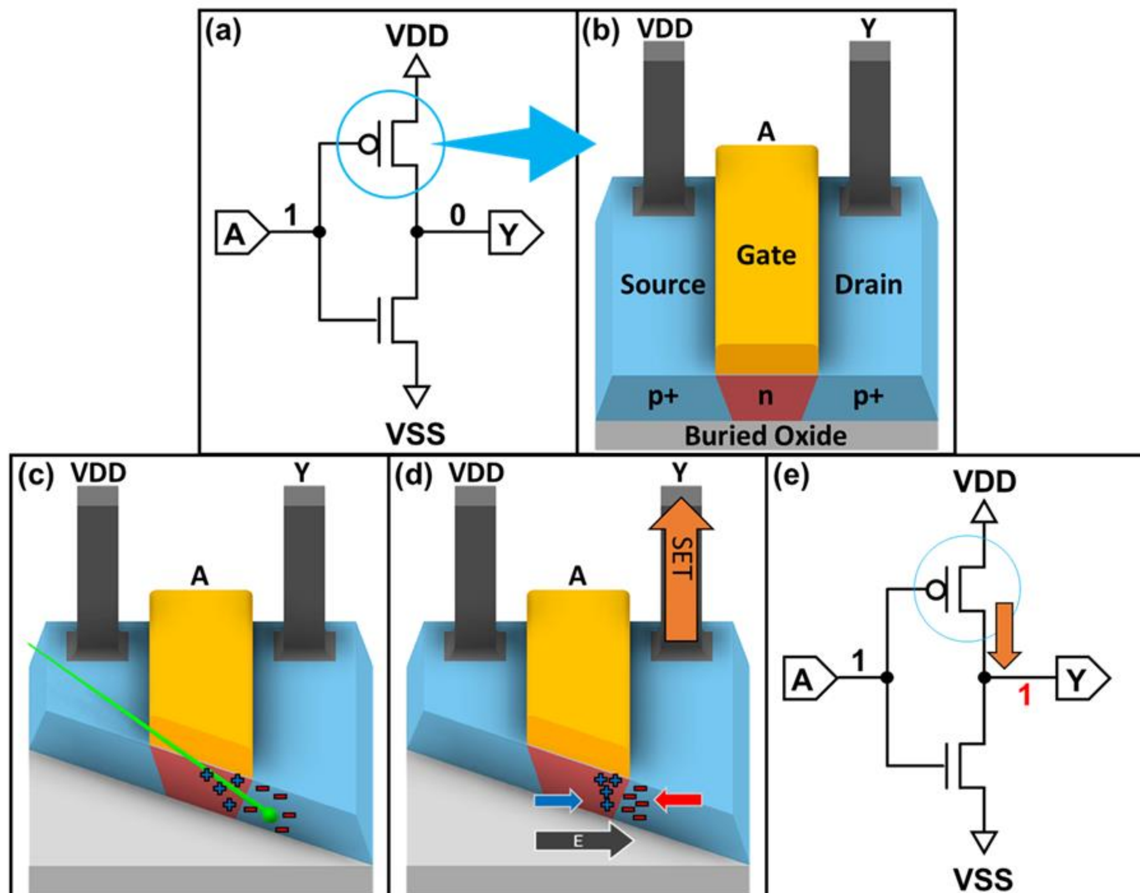


Figure 5: A CMOS logic inverter shown in (a) has an input of logic-1 and a correct output of logic-0. The physical representation of a vulnerable PMOS transistor is shown in (b). The PMOS transistor is vulnerable to SET because there is a large electric potential across its source and drain terminals. In (c), an ion penetrates the PN junction at the drain and generates charge. In (d), the electric field pushes the excess carriers across the junction, resulting in single-event transient current, which is translated back onto the schematic in (e). An errant logic-1 will appear at the output for a short time.

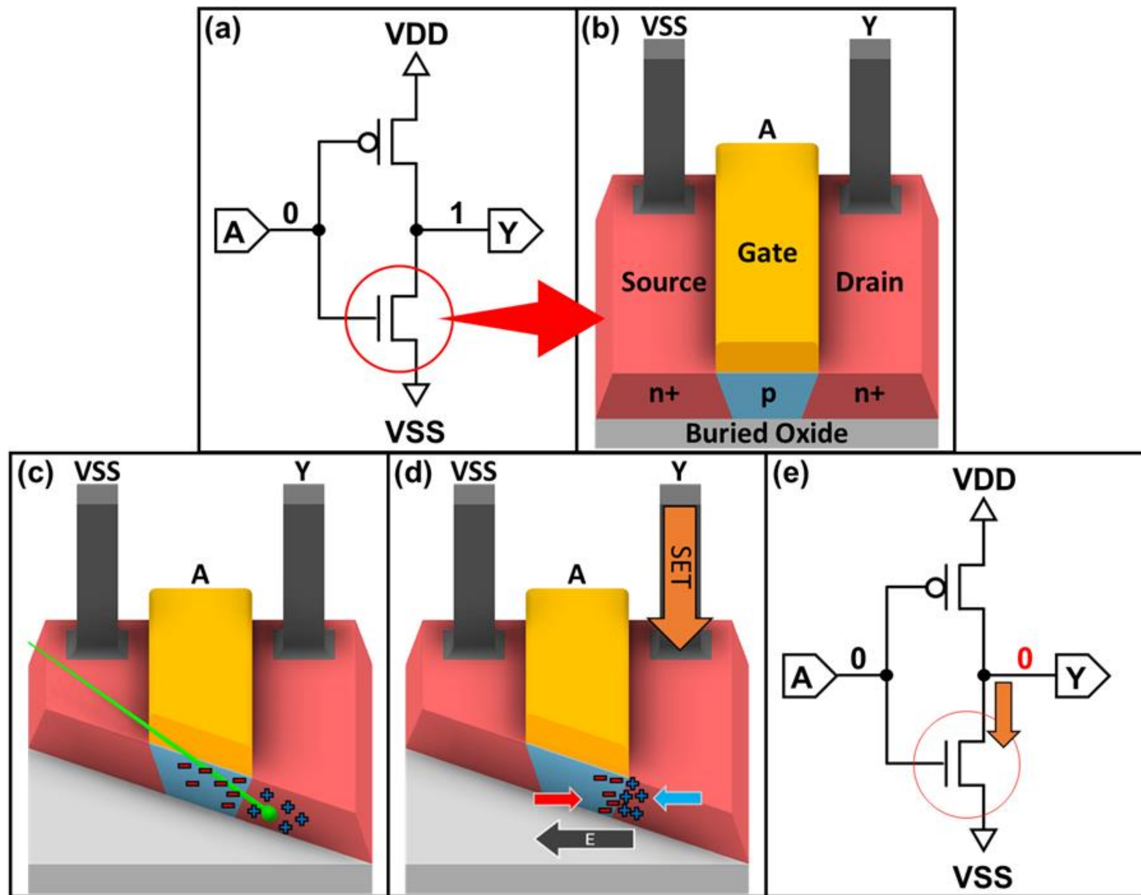


Figure 6: A CMOS logic inverter shown in (a) has an input of logic-0 and a correct output of logic-1. The physical representation of a vulnerable NMOS transistor is shown in (b). The NMOS transistor is vulnerable to SET because there is a large electric potential across its source and drain terminals. In (c), an ion penetrates the PN junction at the drain and generates charge. In (d), the electric field pushes the excess carriers across the junction, resulting in single-event transient current, which is translated back onto the schematic in (e). An errant logic-0 will appear at the output for a short time.

SETs take place in the presence of high electric fields because the peak magnitude of the current pulse is primarily made up of drift current, as opposed to diffusion current. Therefore, to understand where SETs may manifest in a circuit, one must look for places where there is a large electric potential difference and corresponding electric field. The best example is a MOSFET that is biased in cutoff with its source connected to a power supply, as exemplified in the two logic inverter circuits shown in Figure 5 and Figure 6.

So far, the descriptions of the physical mechanisms leading to single-event transients have been qualitative. This is because it is not analytically tractable to describe these mechanisms with any level of generality. The radiation effects community has instead developed mathematical models that approximate physical reality and can be incorporated into simulators which predict a circuit's behavior in a real radiation environment. In practice, experimental data is required to calibrate these models so that they can be used to analyze more complex systems.

To gain a detailed, quantitative perspective of single-event transients, charge transport simulation is performed with Technology Computer Aided Design (TCAD). Charge transport simulation fundamentally solves a system of equations for a network of points which are connected in a one-dimensional, two-dimensional, or three-dimensional mesh which represents a hypothetical or physical structure made of various materials. TCAD analysis has traditionally been used to study the design space of discrete semiconductor devices such as transistors but has also become an informative radiation effects simulation tool. This is because TCAD can be used to compute a device's response to charge that is generated in a single-event. Because charge transport depends on the external loading of a circuit cell, charge transport simulation is often combined with transient circuit simulation [22]. There are large number of TCAD/SPICE mixed-mode simulation techniques and tool-flows that have been developed over the past decades [23], [24], [25], [26].

Due to the high computational cost of mixed-mode TCAD simulation, there is motivation to simulate SETs with a circuit model which does not require TCAD, though TCAD might be used to assist with calibration of the model's parameters. The bias-dependent, radiation-aware MOSFET model developed in [27] matches this description

because it allows the end-user to capture the circuit response to an SET in a SPICE simulation without TCAD. The same model has been re-calibrated to function in multiple technology nodes [27], [28], [29], [30]. This particular model was also used in this work to evaluate the single-event response of various circuit cells and radiation-hardening techniques.

II.B.3) Single-Event Upsets

The threat of SETs pervades nearly every microelectronic application which operates in a radiation environment. In digital systems, every voltage signal is intended fall within a narrow range which represents a discrete logical value and can be unambiguously transmitted from one circuit to the next. When an SET changes the logical value of a signal such that it becomes persistent in sequential logic, it is said to have caused a single-event upset (SEU). Single-event upsets can be propagated throughout systems as part of the normal operation of the device and may ultimately lead to a variety of functional failures or erroneous outputs. If an SEU causes a functional failure or erroneous output, it is referred to as a single-event error. In memory systems, any data bits which are corrupted by SEU are called soft-errors.

Critical charge is the theoretical amount of charge that an ionizing particle must deposit in the sensitive regions of a circuit to induce an SEU [22]. Critical charge is a useful simplification because it gives a one-dimensional measure of how difficult it is to flip the logical state of a storage circuit, such as a latch or flip-flop. However, whether or not an ionizing particle causes an SEU depends not only on how much charge was deposited, but where the charge was deposited. This is due to delayed charge collection effects such as field-funneling [31]. It must be understood that the critical charge of a circuit may give an

overoptimistic estimation of a circuit's vulnerability to SEU [22]. To computationally capture all of the charge collection mechanisms that are not communicated by critical charge, it is necessary to perform a full-stack radiation simulation which includes radiation transport, charge transport, and SPICE simulation. Unfortunately, full-stack radiation simulation is computationally expensive and requires extensive calibration for every semiconductor process and circuit to which it is applied. For this reason, critical charge will continue to be used as a radiation hardness metric despite its physical limitations.

II.B.4) Multiple-Node Upsets and Multiple-Bit Upsets

Multiple-node upsets (MNU), referred to in this dissertation as spatial multi-node upsets (XMNU), occur when a single particle of radiation induces an SET on multiple nodes in a circuit, resulting in one or multiple SEU. If the affected nodes belong to different storage cells in a register or block of memory such that multiple bits were corrupted, then the MNU is also a multiple-bit upset (MBU), also referred to in this dissertation as a spatial multiple-bit upset (XMBU). XMNUs and XMBUs have proliferated in modern CMOS technologies because of technology scaling. This is true for two main reasons. The first reason is that the reduced distance between adjacent transistors in modern circuits increases the likelihood that multiple transistors will intersect any given particle's trajectory. In other words, it is becoming much more likely that a single particle of radiation can strike multiple transistors due to their small size and compact spacing. The second reason is that highly-scaled devices constitute circuits with critical charges that are smaller than their predecessors. As will be discussed in the next section, it is usually feasible to reduce the probability of XMBU by increasing the distance between critical transistors which are used to store independent bits of data.

II.C Radiation-Hardening-by-Design Techniques

In this section, a collection of radiation-hardening-by-design (RHBD) techniques will be introduced. The radiation effects community has developed a wide variety of RHBD techniques to improve reliability, but the ones discussed in this section are important to understand before discussing the original work presented in this dissertation.

II.C.1) Bit-Level Hardening Strategies: DICE Latch

The dual-interlocking cell (DICE) technique uses spatial redundancy to improve a storage cell's reliability against SEU and prevent data loss. A DICE latch implementation from [2] is shown in Figure 7. As can be seen from the figure, the DICE latch does not have a single point of failure; that is, if an SET occurs on any vulnerable node, there are enough redundant storage nodes to retain the binary state of the latch until the SET subsides. However, it is possible to upset the DICE latch with an XMNU. The likelihood

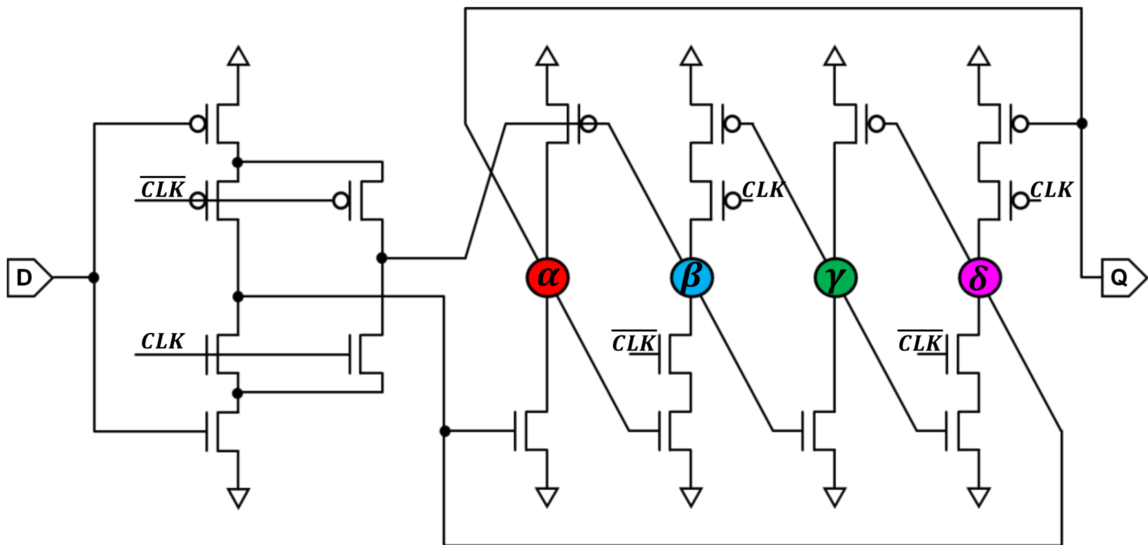


Figure 7: Schematic of a DICE latch implementation, adapted from [2]. When CLK is 1, the latch is in transparent mode. When CLK is 0, the latch is in holding mode. The four storage nodes are labeled α, β, γ and δ. In holding mode, each storage node is reinforced by the other nodes. If one node is corrupted by an SET, the other three nodes will work together to correct the one that was struck.

of an XMNU is several orders of magnitude lower than that of a single-node upset in most semiconductor processes and radiation environments, but there are some radiation environments which can jeopardize that generalization if the space between redundant transistors is not great enough. A detailed analysis of this problem will be presented in Chapter V.

II.C.2) Word-Level Hardening: Error-Detection-and-Correction Codes

Error-detection-and-correction (EDAC) codes allow a block of data to recover from one or multiple bit errors. Extended hamming codes [32] are a commonly employed single-error-correction/double-error-detection (SECDED) code. If a block of memory protected with a SECDED code has a single bit error, it is possible for a computer program or circuit to locate and rewrite the incorrect bit. If two errors are present in the block of memory, then the system reading the memory will be capable of recognizing that the block of memory is invalid but will not have enough information to recover the original data. If more than two errors occur, then it is possible that system reading the memory will not detect the error. This is referred to as silent data corruption.

II.C.3) Error Accumulation Prevention: Scrubbing

EDAC schemes can be overwhelmed if too many errors accumulate within a block of memory. To prevent errors from accumulating, the block of memory needs to be checked and “cleaned” of errors regularly. This process is called scrubbing and has been implemented in many ways in the past decade as the demand for error correction steadily increased with technology scaling. In older technologies where radiation-induced soft errors were relatively rare, computer operating systems would scrub the contents of memory words whenever those words were accessed [33]. This resulted in some words

being scrubbed more frequently than others. When technology scaled to the point where SEU became an unignorable concern, dedicated scrubbing circuits were designed and employed in memories [34]. The maximum frequency at which an individual word may be scrubbed is limited by the scrubbing system's operating frequency and the power it consumes. Due to the SEU-sensitivity of configuration memory in field programmable gate arrays (FPGA), there are a large variety of proposed scrubbing techniques developed in recent years, each of which attempt to maximize the efficiency of available hardware [35], [36], [37], [38], [39]. Of course, the simplest technique is to unconditionally scrub every accessible block of memory as quickly as possible, regardless of word's contents. This is called blind scrubbing and serves as a useful baseline for describing the reliability or efficiency of memory and error correction hardware.

II.C.4) Spatial-MBU Prevention: Bit-Interleaving

To prevent XMBU, many memory systems employ bit-interleaving. Bit-interleaving schemes increase the physical distance between the bits in a block of memory which is protected by an EDAC code. This can be done either with sophisticated layout and interconnect techniques or by using an encoder and decoder in a memory system which scrambles the contents of adjacent words. The implementation of bit-interleaving results in a tradeoff between memory access speed, memory density, and single-event immunity.

II.D Random Processes applied to Single-Event Effects

A Poisson process can be used to model the arrival of radiation particles at a detector if it is exposed to a spatially uniform particle beam which envelopes the entire detector with constant flux ϕ. The Poisson distribution shown in Equation (1) gives the probability that exactly k particles will be detected within time interval Δt:

$$P_{Poisson}(k, \Delta t) = \frac{(\lambda \Delta t)^k e^{-\lambda \Delta t}}{k!} = \frac{\mu^k e^{-\mu}}{k!} \tag{1}$$

The parameter λ is the average rate of particle detections and has units of $(1/s)$ if Δt has units of seconds. The unitless product of λ and Δt is known as the rate parameter μ. The random interarrival time t_i between any two detected particles has an exponentially distributed probability density function:

$$PDF(t_i) = \lambda e^{-\lambda t_i}\, for\ t_i > 0 \tag{2}$$

and an expected value of $1/\lambda$. Another important property of Poisson processes is that they are memoryless. This means that

$$P_{Poisson}(k, t_a - t_b) = P_{Poisson}(k, t_c - t_d)$$

for all $k \geq 0$, provided that the two time intervals $(t_a - t_b)$ and $(t_c - t_d)$ are the same length.

The rate of detected particles λ is directly proportional to the constant flux ϕ, which has units of $1\,/\,(s \cdot cm^2)$. The constant of proportionality is the cross-section of the detector σ, as shown in Equation (3).

$$\left(\text{detection rate } \lambda \left[\frac{1}{s}\right]\right) = (\text{cross section } \sigma\ [cm^2]) \times \left(\text{particle flux } \phi \left[\frac{1}{s \cdot cm^2}\right]\right) \tag{3}$$

Of course, in the context of an experiment, all three of these variables are random variables, each with their own expectation value, variance, and standard deviation.

Integrating the particle flux ϕ with respect to time over the whole duration of the radiation exposure yields the particle fluence Φ, which has units of $1/cm^2$. If it is still assumed that the particle beam completely envelopes the detector, then multiplying the

particle fluence by the detector cross-sections gives the expected number of detections during the irradiation, as stated in Equation (4).

$$(\#\text{ of detections } n) = (\text{cross section } \sigma\ [cm^2]) \times \left(\text{particle fluence } \phi \left[\frac{1}{cm^2}\right]\right) \tag{4}$$

The equations presented in this section were used to describe a specific scenario in which a particle detector with an area σ is exposed to a particle beam, as shown in Figure 8(a). In the study of radiation effects, however, it is common to use Equations (3) and (4) to describe all kinds of single-event effects, not just particle detections. Cross-section is used to express the probability that any given particle of radiation will undergo a specific reaction or cause a single-event effect. For example, if single-event upsets are the radiation effect being counted, then the cross-section is formally referred to as a single-event upset cross-section, as shown in Figure 8(b). It is important to carefully define whatever

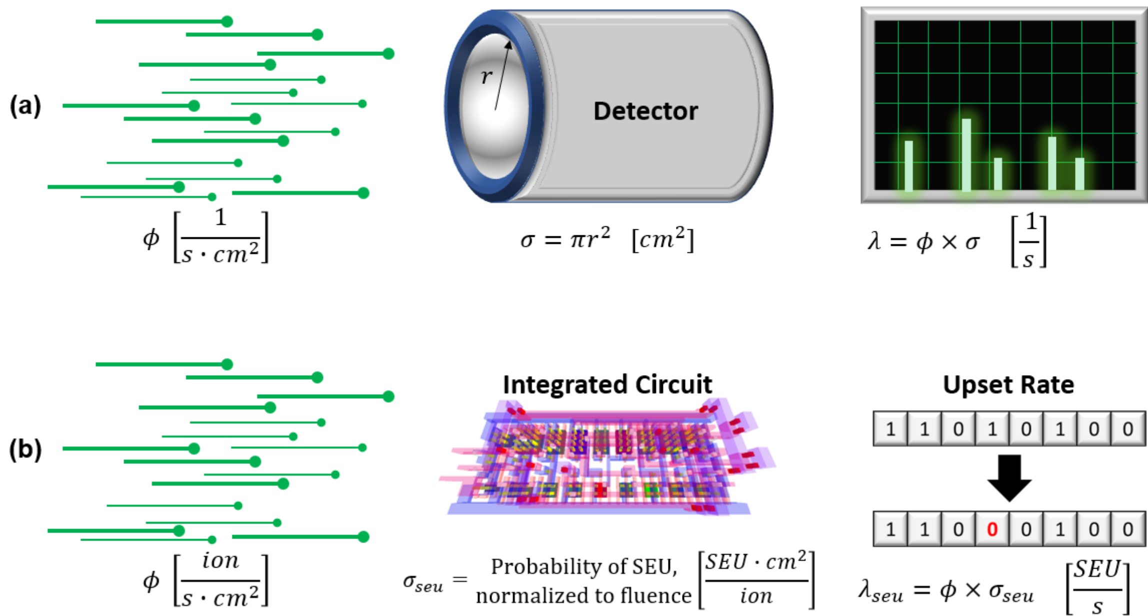


Figure 8: Relationship between flux, cross-section, and event rate. In (a), the cross-section is the physical area of the particle detector's aperture. In (b), the cross-section is a single-event upset cross-section, which describes the probability of an SEU occurring if the circuit is exposed to a fluence of one ion per square-centimeter.

phenomena is described by a cross-section, especially before using the cross-section as a metric of radiation hardness or vulnerability.

II.E Quantification of Reliability

II.E.1) Experiments, One-minus Reliability, and Number of Nine's Reliability

A hypothetical scenario in which a circuit or system is observed for a certain amount of time will be referred to as an *experiment*. Experiments must have well defined outcomes. In probability theory, these outcomes are referred to as *events* (not to be confused with single-event as in single-event effect). In this dissertation, every experiment involves a circuit which is irradiated with ions or neutrons for a specified duration. The events that are defined for these experiments all refer to single-event-effect-induced failure modes and are assigned capital letters. The probability of an event X occurring during an experiment is written with the following notation:

$$probability\ of\ event\ X\ occuring = P(X) \tag{5}$$

Often, however, we are more interested in the probably of an event not happening when the event is a particular kind of failure. This is described as the reliability against that particular failure. For example, if the event S refers to the outcome in which a single-event upset occurs during the experiment, then the reliability against SEUs would be expressed as:

$$reliability\ against\ SEU = R_S = 1 - P(S) = P(\overline{S}) \tag{6}$$

Because probabilities, and therefore reliabilities, are only defined between zero and one, is often useful to use a logarithmic scale to describe the likelihood of an event occurring or not occurring.

Systems that need to be reliable against a certain failure mechanism will often be designed to have reliabilities that are very close to one. High availability systems use a figure of merit called "number of nines" reliability, which indicates how many leading, consecutive digits are nines. For example, a reliability of 0.999<u></u>3592 would have three nines of reliability. A mathematical definition which is continuous for any defined reliability is:

$$Number\ of\ nines\ of\ reliability\ in\ R = R9 = -\log_{10}(1-R) \quad (7)$$

For systems with more than 10 nines of reliability, care must be given in the computational implementation of Equation (7) because the argument to the logarithm can become very small, resulting in numerical errors. Many computational frameworks implement a special function called "log1p" which has the following definition:

$$log1p(x) = \ln(1+x) \quad (8)$$

Log1p is used to calculate the logarithm of numbers which very close to one without numerical error. ln denotes the natural logarithm. The number of nines of reliability is then calculated with Equation (9).

$$R9 = -\frac{log1p(-R)}{\ln 10} \quad (9)$$

The number of nines of reliability is useful for plotting reliabilities, especially when different reliabilities are compared against each other. Such a plot will be used in the next subsection.

II.E.2) Reliability of Digital Memories

When a circuit designer is selecting the size of a memory which must meet a reliability requirement, it is useful to extrapolate the reliability of a memory word to calculate the reliability of the entire memory. Assuming that each word has the same reliability, and that each word is independent from others, this extrapolation can be achieved with exponentiation as shown in Equation (10), where W is the number of words in the memory.

$$R_{memory} = (R_{word})^W \tag{10}$$

What is somewhat less obvious, is how exponentiation affects a reliability which expressed as a number of nines. A rule of thumb that makes number of nines reliabilities easy to exponentiate is: One nine of reliability is lost for each power of ten that is in the exponent. This rule is presented more formally as:

$$R9_{memory} \approx R9_{word} - \log_{10} W \quad for \quad R9_{memory} \geq 1 \tag{11}$$

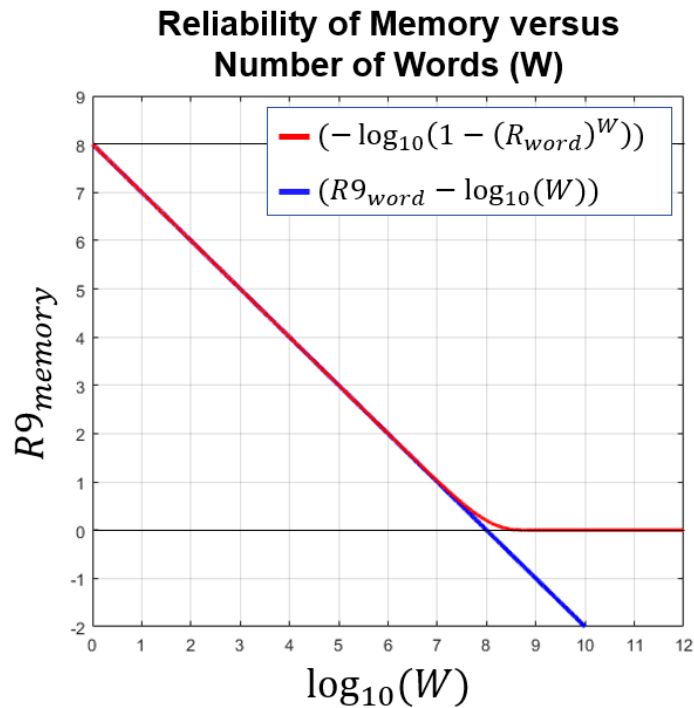


Figure 9: Comparison of two calculations for the reliability of a memory where each individual word of memory has a reliability of 0.99999999, or 8 nines of reliability. The two calculations agree for memories with a reliability of at least 0.9.

The rule holds so long as the result of the exponentiation produces a reliability that has at least one nine. Figure 9 shows that the approximation holds and there is no need to convert from number of nines, exponentiate, and convert back. This rule is useful because it allows a designer who is reviewing a memory word's reliability in number of nines notation to quickly determine how many words may be used in a memory and still meet the overall memory reliability requirement.

CHAPTER III

TEMPORAL-SPATIAL MODEL FOR RELIABILITY AGAINST MULTIPLE-BIT UPSET

The use of highly scaled memory in integrated circuits is ubiquitous, which makes the threat of SEU also ubiquitous. Because SECDED codes are common in memory systems like SRAM, the threat of multiple-bit errors is an important reliability problem to quantify. In this chapter, a new model is presented which analytically separates the two underlying mechanisms which cause multiple-bit errors and then recombines them. By modularizing the mathematical expression which predicts the probability of a multiple-bit error occurring, the new model provides insight to circuit designers and allows them to remedy the appropriate underlying cause of unreliability.

After formally defining the problem to be solved and reviewing related work performed by the radiation effects community, this chapter presents a derivation of the new model along with its underlying assumptions and constraints. Evidence for the model's accuracy is then presented in the form of simulation results. Finally, the significance of the new model is discussed.

III.A Problem Definition

As stated in section II.C.2, SECDED codes are unable to protect data if more than one bit-error has occurred. Fundamentally, SEU can do this in two ways. The first way is for a single particle to cause more than one SEU simultaneously, as shown in Figure 10(a-b).

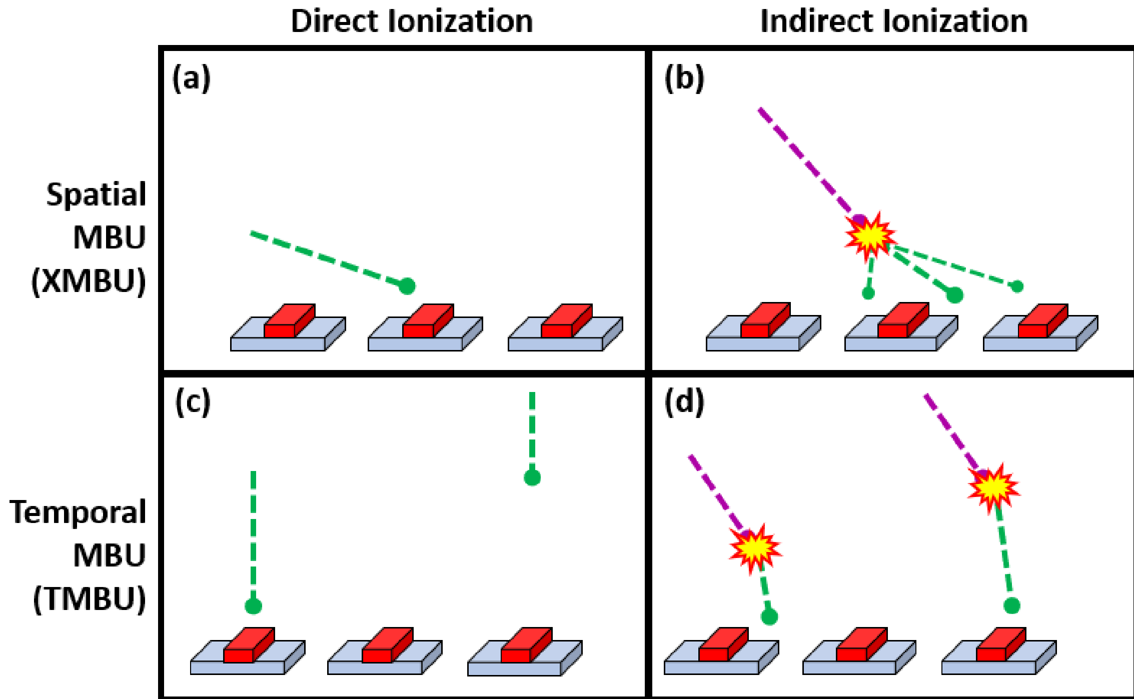


Figure 10: Comparison of XMBU and TMBU. The purple particle trajectory represents neutrons while the green particle trajectories represent ions. (a) and (b) represent examples of spatial MBU while (c) and (d) represent examples of temporal MBU.

The radiation effects community has traditionally referred to this as simply "multiple-bit upset" or "single-event multiple-bit upset," but in this dissertation, this will be referred to as spatial multiple-bit upset (XMBU). XMBU is so named because it depends entirely on particle trajectory, transistor placement, and other geometric factors. Conversely, XMBU has no dependence on how often the data is scrubbed. This is because two upsets that were caused by the same particle occur with nearly perfect simultaneity and there is no way for a scrubbing circuit to identify one bit-error before the second bit-error occurs.

The other way two or more upsets can occur is if two or more particles simply cause at least one SEU each, as shown in Figure 10(c-d). The radiation effects community has traditionally referred to this as SEU accumulation. In this dissertation, this will be referred

to as temporal multiple-bit upset (TMBU). TMBU is so named because it depends temporal factors, like the particle flux, the time-rate of upsets, and how frequently the SECDED code is scrubbed. Conversely, TMBU has no dependence on geometrical factors because two different particles of radiation will have completely independent trajectories.

It is the goal of this chapter to develop a mathematical formula which expresses the reliability of a memory word against MBU as a function of exposure time. This formula must account for both the possibility of XMBU and TMBU while also providing circuit designers insight which can help improve their designs. Additionally, the inputs and parameters to the derived mathematical function must be obtainable through experiment and simulation.

III.B Related Work

The radiation effects community has produced several works which address the problem of reliability against flux-dependent MBU. Each of these works, including the new work presented in this dissertation, settle on a varying degree of accuracy and generality for the reliability models that they derive.

In [40], the authors established analytical methods for quantifying the threat of SEU accumulation while considering the possibility that one particle might cause multiple bit upsets in different memory words. However, that model assumes that bit-interleaving has eliminated the possibility of a single particle causing two SEUs in the same word. In [5], a computational framework called MACAU was developed which uses a Markovian model to calculate the failures-in-time and mean-time-to-failure in an SRAM while including the possibility of both XMBU and TMBU. Older works such as [33], neglect the possibility of XMBUs and assume that each bit is independent of the other bits within the same word.

These assumptions and constraints were very reasonable for the available transistor dimensions and memory densities available at the time of publication of [33], but not for the submicron transistors used in modern technology.

III.C Spatial-Temporal Reliability Model

The Spatial-Temporal Reliability (XTR) Model presented in this section meets the requirements stated in section III.A. It divides the problem of accounting for both XMBU and TMBU into their respective equations which can be evaluated separately and then combined.

III.C.1) Constraints and Assumptions of the Spatial-Temporal Reliability Model

The Spatial-Temporal Reliability Model is used to describe the probability of a word of memory not failing due to multiple-bit upset. Toward this end, there are three assumptions which must be made before proceeding.

The first assumption is that once a bit inside the word is corrupted by an SEU, another SEU cannot "uncorrupt" the bit by flipping it back to its correct state. This assumption is reasonable for words with a relatively large number of bits, like 32 or 64. This assumption is also a conservative one, because the "un-corruption" of a bit would lower the chance of failure to MBU. While it is possible for bits to be set back to their original states by a second SEU, it is only likely to happen after a sizeable minority of bits in the word have already been upset, which would imply an MBU had already occurred. This assumption will be called the upset-permanence assumption.

The second assumption that needs to be made is that all bits in a data word are equally likely to be upset, regardless of their binary states. This is generally not true in most CMOS

memories because NFETs and PFETs in each storage cell will have unequal current drive strength, charge collection volume, or both. The assumption that each bit is identically vulnerable regardless of position and state will be referred to as the bit-uniformity assumption.

The third assumption is that there is an ideal scrubbing circuit which periodically checks the contents of the memory word and corrects single-bit errors instantaneously. The scrubbing circuit operates without error and performs its function with perfect periodicity. The frequency at which the scrubbing circuit checks the memory word will be called f_{scrub}. The term "scrubbing cycle" will be used to refer to the instantaneous action of clearing single-bit errors. The reciprocal of f_{scrub} will be referred to as the "scrubbing interval" and will be assigned the symbol τ in Chapter IV.

III.C.2) Derivation of the Spatial-Temporal Reliability Model

Consider an experiment or simulation where a word of memory is exposed to radiation for time t. This time is referred to the radiation pulse duration or exposure time. The particle flux ϕ is constant and uniform for the duration of the exposure. We define two discrete random variables ζ and η, as well as probabilistic outcomes (events) in Table 1 which are defined in terms of these two random variables. ζ is the greatest number of SEU-inducing particles that arrive between any two scrubbing cycles. η is the largest number of bits that were flipped by a single particle.

Table 1: Events Defined for Radiation Experiment

Event	Definition	Description
S	$\eta \geq 1$	A single-bit upset (SBU) occurred. That is, at least one SEU occurred during irradiation.
X	$\eta \geq 2$	A spatial multi-bit upset (XMBU) occurred. That is, at least one particle caused at least two bits to flip simultaneously.
T	$\zeta \geq 2$	A temporal multi-bit upset (TMBU) occurred. That is, at least two particles arrived within the same scrubbing interval, and each caused at least one bit to flip.
M	$X \cup T$	A multi-bit upset (MBU) occurred. That is, at least two bits were flipped during irradiation within the same scrubbing interval.

For now, it is asserted that there is a way to compute the probabilities of S, X, and T as functions of time. These probabilities will be discussed later in the chapter in section III.C.4. The XTR Model introduced in this section is designed to combine these probabilities to find $P(M)$ without prescribing a value for $P(S)$, $P(X)$, or $P(T)$. The probability of *M* is what is sought because to know $P(M)$ is to know R_M, the reliability of the word against MBU.

By inspecting the definitions given in Table 1, it can be seen that if either *X* or *T* occur, then *M* has occurred. That is, *M* is the union of *X* and *T* with the corresponding probability:

$$P(M) = P(X \cup T) \tag{12}$$

By using the inclusion-exclusion principle, we can rewrite (12) as the sum of the individual probability of *X* and *T* minus the probability of their intersection.

$$P(M) = P(X) + P(T) - P(X \cap T) \tag{13}$$

This change shifts the burden of finding an unknown union to an unknown intersection, however, it is feasible to approximate $P(X \cap T)$ in several ways, as will be shown here.

The first approximation is that $P(X \cap T)$ is equal to zero. This is reasonable in systems that are very reliable; if the probability of a X and T are both lower than 10^{-6}, one would expect that the probability of both occurring would be several orders of magnitude lower. This assumptions results in an expression for R_M that shown in Equation (14).

$$P(X \cap T) \approx 0 \quad \rightarrow \quad P(M) = P(X) + P(T)$$

$$1 - R_M = 1 - R_X + 1 - R_T$$

$$P(X \cap T) \approx 0 \quad \rightarrow \quad R_M = R_X + R_T - 1 \;\; ; \quad R_X + R_T > 1 \tag{14}$$

This equation is referred to as the Level-0 Spatial-Temporal Reliability (XTR) Model. Of course, one stipulation of this model is that the sum of R_X and R_T be greater than or equal to one. While there are scenarios where Level-0 XTR model suffices, it is possible to make a narrower assumption which gives improved accuracy over a wider range of values for $P(X)$ and $P(T)$.

Rather than assuming that $P(X \cap T)$ is equal to zero, one might instead assume that X and T are independent events, implying that the probability of the intersection is the product of the individual probabilities. This is not true because there are SEU mechanisms that are common to both events. However, this is a narrower assumption than setting $P(X \cap T)$ equal to zero and results in an expression for R_M that is shown in Equation (15).

$$P(X \cap T) \approx P(X)P(T) \quad \rightarrow \quad P(M) = P(X) + P(T) - P(X)P(T)$$

$$1 - R_M = 1 - R_X + 1 - R_T - (1 - R_X)(1 - R_T)$$

$$1 - R_M = 1 - R_X + 1 - R_T - 1 + R_X + R_T - R_X R_T$$

$$P(X \cap T) \approx P(X)P(T) \quad \rightarrow \quad R_M = R_X R_T \tag{15}$$

This is the Level-1 Spatial-Temporal Reliability Model. It is applicable for the entire range of R_X and R_T and gives excellent agreement with simulation results. Of all of the versions of the Spatial-Temporal Reliability Model presented in this chapter, the Level-1 model is the easiest to use and understand.

There is one more variant of the XTR model in this chapter which relies on an assumption that is even narrower than one used in the Level-1 XTR model. Suppose that the law of total probability is used to manipulate the intersection of X and T as shown in Equation (16). This introduces the probability of S.

$$P(M) = P(X) + P(T) - \left[P(X \cap T \,|S)P(S) + P\left(X \cap T|\overline{S}\right)P\left(\overline{S}\right)\right] \tag{16}$$

The relationship of S to X and T will reveal why adding it into the expression via the law of total probability is useful. By the definitions chosen in Table 1, X and T are both subsets of S. This is best illustrated in the Venn diagram in Figure 11. As a result of the subset

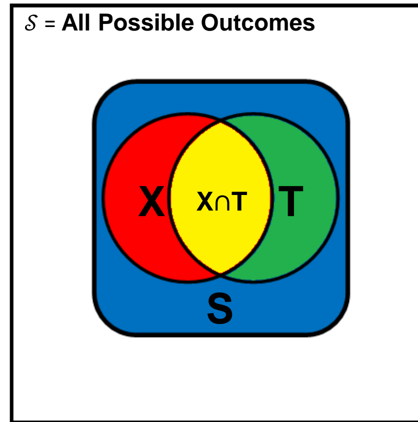


Figure 11: Venn diagram showing the relationship between the events S, X, and T as well as the intersection of X and T. $\mathcal{S}$ represents all possible outcomes and is formally known as the sample space.

relationship, Equation (16) can be simplified because $X \cap T$ and $\overline{S}$ are disjoint and the probability of either X or T occurring when S doesn't occur is zero.

$$P(M) = P(X) + P(T) - P(X \cap T|S)P(S) \tag{17}$$

This rearrangement of Equation (13) permits the use of another assumption, which is shown in Equation (18).

$$P(X \cap T|S) \approx P(X|S)P(T|S) \tag{18}$$

In probability theory, Equation (18) is known as the definition of conditional independence. That is, X and T are conditionally independent on S. It can be shown analytically that Equation (18) is not strictly true for some expressions of $P(X)$ and $P(T)$. However, all other things equal, a model which relies on a narrow assumption will give better accuracy than a model which relies upon a broad assumption.

The expression in Equation (18) can be rearranged by using the definition of conditional probability:

$$P(X \cap T|S) \approx P(X|S)P(T|S) = \left(\frac{P(X \cap S)}{P(S)}\right)\left(\frac{P(T \cap S)}{P(S)}\right) \tag{19}$$

Once again, leveraging the fact that X and T are both subsets of S allows the intersections to be reduced, resulting in:

$$P(X \cap T|S) \approx P(X|S)P(T|S) = \frac{P(X)P(T)}{[P(S)]^2} \tag{20}$$

Substituting the expression in (20) for the third term of (17) provides an expression which gives the probability of multiple-bit upset as a combination of both underling mechanisms, XMBU and TMBU.

$$P(M) = P(X) + P(T) - \frac{P(X)P(T)}{P(S)} \tag{21}$$

The expression in (21) is the probability-of-failure form of the Level-2 XTR model. It can be rewritten as a reliability against MBU in a one-minus-probability form as shown in Equation (22).

$$R_M = 1 - P(M) = \frac{R_X R_T - R_S(R_X + R_T - 1)}{1 - R_S} \tag{22}$$

Where R_M, R_S, R_T, and R_X correspond to the reliability against MBU, SBU, TMBU, and XMBU respectively. It will be shown in section *III.D* how the Level-2 XTR model gives better accuracy than the Level-0 and Level-1 models in some cases.

III.C.3) Analysis of the Spatial-Temporal Reliability Model

The XTR model describes a memory word's reliability against MBU as symmetrically dependent upon its reliability against both XMBU and TMBU. The reliability R_S is a hypothetical reliability of the memory word if it had no single-error correction. A memory system which is designed to be tolerant against single-bit upsets will have an R_M which is much greater than R_S. In this case, both R_X and R_T will be much greater than R_S and Level-2 XTR model reduces to the Level-1 XTR model. This is demonstrated by taking the limit of the Level-2 XTR model as R_S approaches zero.

$$\lim_{R_S \to 0} (R_M) = \lim_{R_S \to 0} \left(\frac{R_X R_T - R_S(R_X + R_T - 1)}{1 - R_S} \right) = R_X R_T \tag{23}$$

The Level-1 and Level-2 XTR model agree that, in most scenarios, the overall reliability against MBU is limited by whichever underling vulnerability is greater. The other limits of the Level-2 XTR model which describe its behavior are shown in Table 2.

Table 2: Limits of the Level-2 XTR Model

Limit	Value that R_M takes	Physical interpretation
$R_S \to 0$	$R_X R_T$	A single-bit upset is guaranteed. This is the same as stating that the reliabilities against XMBU and TMBU are independent of each other.
$R_S \to 1$	1	A single-bit upset is impossible. Therefore, an MBU is impossible. Note that as R_S approaches unity, R_X and R_T will approach unity faster than R_S does.
$R_X \to R_S$	R_S	The bit-level storage circuits in the word are so compact that it is impossible for a particle to upset one bit without upsetting two or more bits. The SECDED code will be defeated as soon as the first SEU occurs.
$R_T \to R_S$	R_S	The SBU rate parameter is so high that as soon as one particle upsets a bit, another particle will immediately upset another.

III.C.4) Reliability Against SBU, XMBU, and TMBU

Using the Spatial-Temporal Reliability Model requires that a method of determining R_S, R_X, and R_T is available. In this subsection, expressions based off of physical parameters for these reliabilities will be offered. The first two reliabilities are relatively straightforward to compute while the third reliability is more difficult to determine.

As discussed in section II.D , the arrival of phenomena such as SEU can be modeled as a Poisson process. For a memory word with a known SEU cross-section σ_S that is exposed to a constant and uniform particle flux ϕ, the probability that at least one SEU occurs before time t is given by Equation (24), where λ_S is the rate parameter defined in Equation (25). The reliability against single-bit upset is shown in functional form in Equation (26).

$$P(S) = \sum_{k=1}^{\infty} P_{Poisson}(k, \sigma_S t) = 1 - P_{Pois.}(0, \lambda_S t) = 1 - e^{-\lambda_S t} \tag{24}$$

$$\lambda_S = \sigma_S \phi \tag{25}$$

$$R_S(t) = 1 - P(S) = e^{-\lambda_S t} \tag{26}$$

A nearly identical set of equations exist for spatial multi-bit upset if a spatial-MBU cross-section is defined. These equations are shown in (27)-(29).

$$P(X) = 1 - e^{-\lambda_X t} \tag{27}$$

$$\lambda_X = \sigma_X \phi \tag{28}$$

$$R_X(t) = 1 - P(X) = e^{-\lambda_X t} \tag{29}$$

The meaning of the XMBU cross-section σ_X is the fluence-normalized probability that a particle will upset two or more bits. A method of determining both σ_S and σ_X will be given in subsection III.C.5.

There are a few different ways in which the reliability against TMBU might be expressed in terms of physical parameters. The model proposed in [33] will be described here because the assumptions that it leans upon are compatible with the XTR model. This model treats the occurrence of SEU in each bit in a word as an independent Poisson process which all share a common rate parameter, which we will call λ_b. The probability that any given bit will be upset by time t is given in Equation (30), where the value of λ_b is given in Equation (31).

$$P(bit\ will\ upset) = 1 - e^{-\lambda_b t} \tag{30}$$

$$\lambda_b = \sigma_b \phi \tag{31}$$

The upset-per-bit cross-section σ_b bears a causal relationship with the upset-per-word cross-section σ_S. If a word has *b* bits, the simplest connection between the two cross-sections is:

$$\sigma_S \approx \sigma_b \times b \tag{32}$$

However, the possibility of XMBU makes the relationship in (32) inaccurate. A more thorough discussion of these cross-sections will be given in the next subsection.

Given the independent probabilities of each bit being upset, it possible to compute the probability that *k* bits will be upset before time *t* by using the binomial distribution:

$$Binomial(n, k, p) = \binom{n}{k} (p)^k (1 - p)^{n-k} \tag{33}$$

The notation in Equation (33) indicates the probability that n Bernoulli trials will have k successes, where the probability of an individual success is p. According to the model in [33], we can treat each bit as an independent Bernoulli trial where a "success" counts as at least one SEU occurring on that bit. Now we can define the event B_i for all i such that $\{0 \leq i \leq b\}$ as the outcome where i independent bits were upset before time t. The probability of B_i is shown in Equations (34)-(35).

$$P(B_i) = Binomial(n = b, k = i, p = 1 - e^{-\lambda_b t}) \tag{34}$$

$$P(B_i) = \binom{b}{i}\left(1 - e^{-\lambda_b t}\right)^i \left(e^{-\lambda_b t}\right)^{b-i} \tag{35}$$

Therefore, the probability that at least two bits are upset out of a word with b bits is:

$$P(T) = \sum_{i=2}^{b} P(B_i) = 1 - \sum_{0}^{1} P(B_i) \tag{36}$$

The last simplification in Equation (36) is possible because all B_i are disjoint and their union accounts for all possible outcomes. Substituting the expression from (35) into (36) yields Equation (37). Preceding calculation steps are shown below as well.

$$P(T) = 1 - P(B_0) - P(B_1)$$

$$P(T) = 1 - \binom{b}{0}\left(1 - e^{-\lambda_b t}\right)^0 \left(e^{-\lambda_b t}\right)^{b-0} - \binom{b}{1}\left(1 - e^{-\lambda_b t}\right)^1 \left(e^{-\lambda_b t}\right)^{b-1}$$

$$P(T) = 1 - e^{-b\lambda_b t} - b\left(1 - e^{-\lambda_b t}\right)\left(e^{-\lambda_b t}\right)^{b-1}$$

$$P(T) = 1 - e^{-b\lambda_b t} - b(e^{-(b-1)\lambda_b t} - e^{-b\lambda_b t})$$

$$P(T) = 1 - be^{-(b-1)\lambda_b t} + (b - 1)e^{-b\lambda_b t}$$

$$\dot{b} = b - 1$$

$$P(T) = 1 - be^{-\dot{b}\lambda_b t} + \dot{b}e^{-b\lambda_b t} \tag{37}$$

$$R_T(t) = 1 - P(T) = be^{-\dot{b}\lambda_b t} - \dot{b}e^{-b\lambda_b t} \tag{38}$$

The expression in Equation (38) is the reliability against TMBU as a function of exposure time t, given that no periodic scrubbing is applied. This function is often referred to as the Saleh model, named after the first author of [33]. That work includes a variant of the Saleh model which includes blind periodic scrubbing. The derivation of the Saleh model which includes periodic scrubbing is somewhat involved and will be explained in Chapter IV.

In addition to the upset-permanence and bit-uniformity assumptions, the Saleh model also assumes that the occurrence of SEU in each bit is independent from that of every other bit. This is strictly untrue if spatial MBU are to be considered. Why this is strictly untrue is easy to imagine if one considers a highly scaled memory system with no bit-interleaving. In this kind of vulnerable system, most particles which cause at least one SEU will cause many more. Therefore, if a bit at index j is upset by a particle, it is likely that the bits at indices $(j-1)$ and $(j+1)$ will also be upset. Because of the possibility XMBU, bitwise SEU independence is impossible. Conversely, if XMBU were impossible, then bitwise SEU independence would be guaranteed; each particle would only induce one upset in one bit, and each bit that was upset would have no bearing on the state of any adjacent bits. This assumption will be henceforth referred to as the bitwise-SEU independence assumption.

Identifying that the expression for the reliability against TMBU relies upon the bitwise-SEU independence assumption motivates the main hypothesis in this dissertation.

> When combining $P(T)$ and $P(X)$ with the XTR model, any mathematical formula used to express $P(T)$ or $R_T(t)$ can ignore the physical possibility of XMBU and any mathematical formula used to express $P(X)$ or $R_X(t)$ can ignore the physical possibility of TMBU.

This statement will be henceforth referred to as the Spatial-Temporal Orthogonal Causality (XTOC) Hypothesis. It follows from this statement that the bitwise-SEU independent assumption is valid when applied to the calculation of $P(T)$ or $R_T(t)$. Furthermore, it implies that the approximation given in Equation (32) is accurate if the upset-per-bit cross-section σ_b is only used in the calculation of $P(T)$ or $R_T(t)$.

The greatest advantage given by this hypothesis, if found to be true, is that the mathematical effect of error scrubbing can be isolated to $R_T(t)$. This profoundly simplifies the MBU reliability problem. If $R_T(t)$ is too low, then the scrubbing frequency can be increased. If $R_X(t)$ is too low, then the spacing between vulnerable nodes can be increased. After performing these alterations, the values of $R_T(t)$ and $R_X(t)$ can be recomputed and inserted into the XTR model to see the improvement in overall reliability against MBU.

III.C.5 Determination of Cross-Sections

For the Spatial-Temporal Reliability Model or any of its constituent models to be practical, there must be away to find the average SEU rate λ_S and average XMBU rate λ_X for a constant flux, as well as the corresponding upset cross-sections σ_S and σ_X. This can be done in an experiment where a memory with SECDED code is loaded with a predetermined set of data and irradiated with a low, constant, and uniform flux of protons,

neutrons, or heavy ions. The flux would need to be low enough such that a memory scanning circuit would be able to detect the impact of a single particle and count the number of bits that were upset. If the memory scanning circuit is not fast enough, a TMBU could be miscounted as an XMBU.

If we assume that it is practical to perform such an experiment, the quantities in Table 3 would need to be known beforehand or counted during the experiment. The values of N_S and N_X are what is sought during the experiment. Many experiments focus only on counting the total number of bits which are upset, but to use this model, it is required to measure how many bits are upset within each word, and whether or not they are attributed to the same particle or different particles. If the values in Table 3 are known, then the necessary upset cross-sections may be estimated using the following equations:

$$\sigma_S = \frac{N_S}{W\phi} \tag{39}$$

$$\sigma_X = \frac{N_X}{W\phi} \tag{40}$$

Table 3: Experiment Parameters Necessary to Use Spatial-Temporal Reliability Model

Quantity	Obtained How	Definition
b	Circuit parameter	The number of bits in each word
W	Circuit parameter	The number of words in the memory
N_S	Counted During Experiment	The number of times that a particle causes at least one upset in exactly one word. If one particle causes upsets in multiple words, N_S is incremented multiple times.
N_X	Counted During Experiment	The number of times that a particle causes at least two upsets in exactly one word. If one particle causes XMBUs in multiple words, N_X is incremented multiple times.
Φ	Experiment Specification	The fluence that the memory circuit is irradiated to.

III.D Simulation-Based Evidence for Hypotheses

A Monte-Carlo simulation program was written test the XTOC hypothesis and the accuracy of the XTR model. A flowchart which describes the simulation program is given in Appendix A. Ten-million pseudorandom simulations were performed for several values of $\chi = \sigma_X / \sigma_S$, as well two different error scrubbing frequencies and one case without any

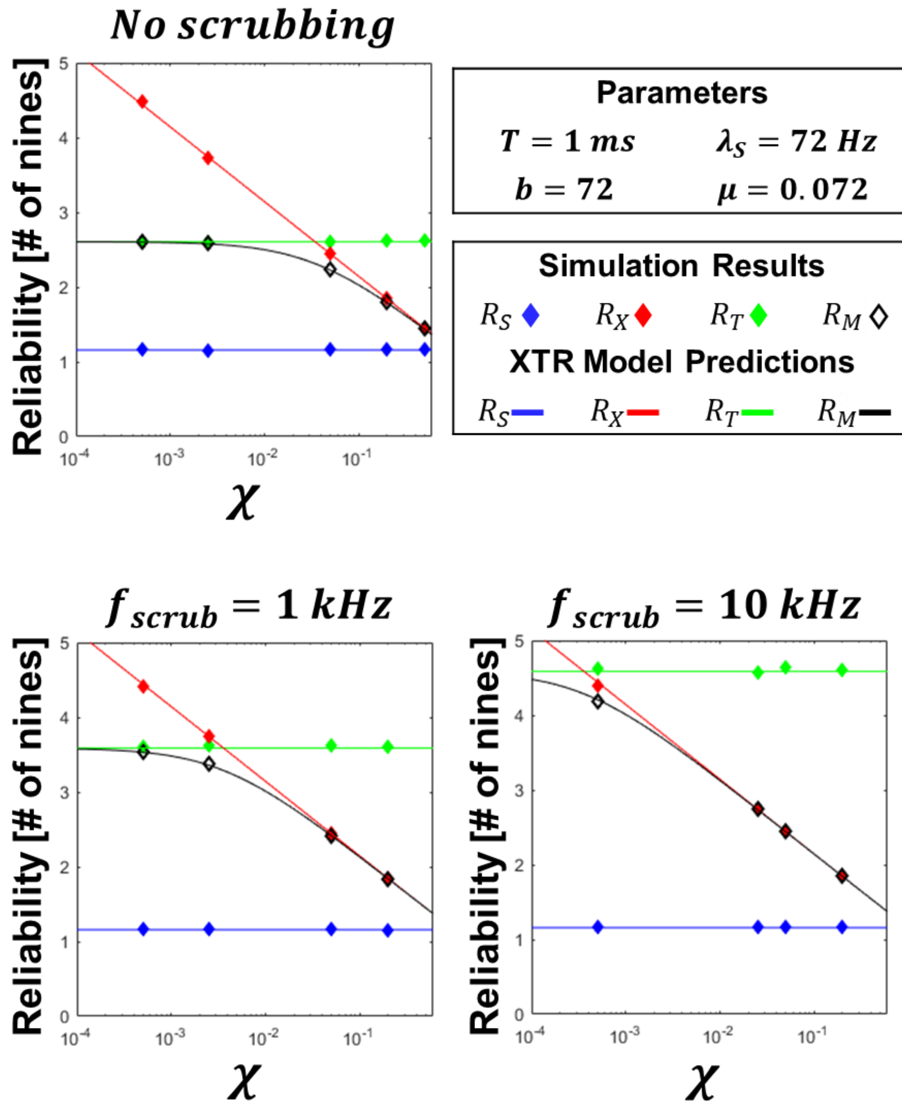


Figure 12: Monte-Carlo simulation results for a memory word with the parameters shown. The results were calculated by performing multiple simulations, recording which ones resulted in outcomes $S, X, T, or\ M$ and calculating the ratio. 10 million simulations were performed. All three variants of the XTR model produced results that are too similar to distinguish on the logarithmic scale.

error scrubbing. The results of the simulation are plotted in Figure 12. As can be seen from Figure 12 and its caption, the XTR model gives excellent agreement at levels 0, 1, and 2. The difference between the different levels of the models can only be seen if we look at the raw probabilities, as shown in Table 4. If we trust the Monte-Carlo simulation to be the "correct" value, then it can be seen that the level-2 model most closely estimates the probability of the intersection of X and T. However, for many scenarios, any of the XTR model levels will suffice. The discrepancies between the estimations provided by each of the XTR models and the results of the Monte-Carlo simulation are plotted in Figure 13 as percent differences.

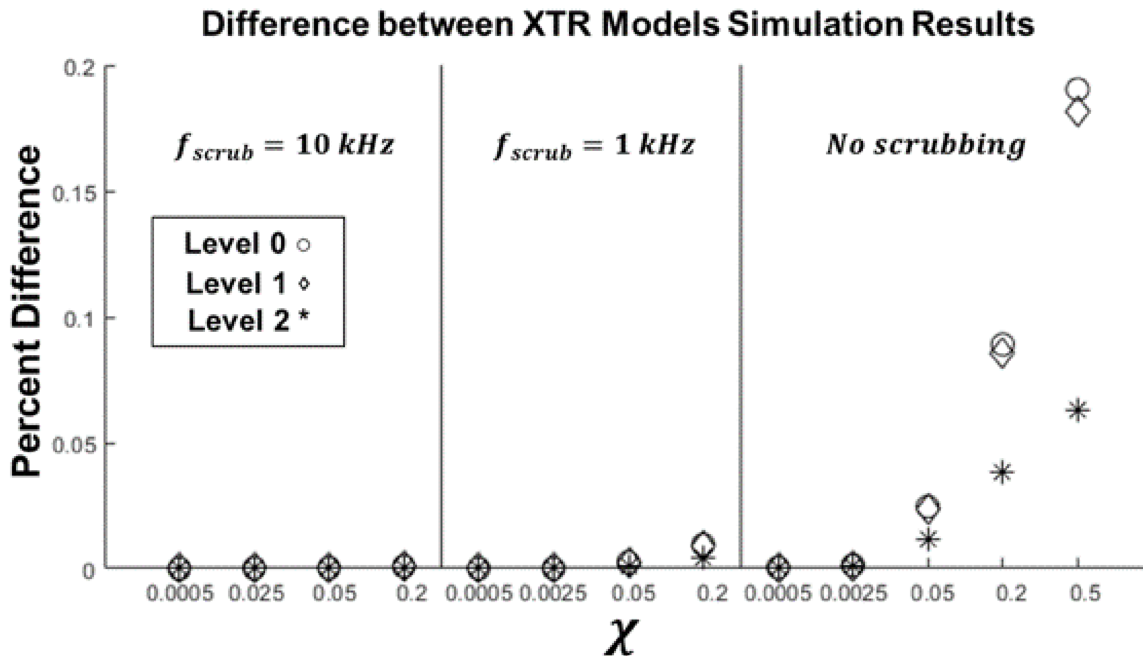


Figure 13: Percent difference for estimation of R_M between the XTR model and Monte-Carlo simulation for each of the three model levels. The difference between the three models is only significant when no scrubbing is applied, and spatial multiple-bit upsets are relatively frequent.

Table 4: Simulation Data from Figure 12

$f_{scrub}[kHz]$	χ	R_M	L0 XTR	L1 XTR	L2 XTR
1	0.0005	0.999935_**4**	0.999935_**3**	0.999935_**3**	0.999935_**3**
1	0.0025	0.99817_**75**	0.99817_**57**	0.99817_**57**	0.99817_**64**
1	0.05	0.9964_**176**	0.9964_**151**	0.9964_**152**	0.9964_**163**
1	0.2	0.9856_**925**	0.9856_**835**	0.9856_**839**	0.9856_**886**
10	0.0005	0.999711_**8**	0.999711_**7**	0.999711_**7**	0.999711_**8**
10	0.025	0.99957_**95**	0.99957_**85**	0.99957_**85**	0.99957_**91**
10	0.05	0.9961_**798**	0.9961_**548**	0.9961_**557**	0.9961_**672**
10	0.2	0.985_**4876**	0.985_**3941**	0.985_**3977**	0.985_**4454**
N/A	0.0005	0.99751_**69**	0.99751_**22**	0.99751_**23**	0.99751_**34**
N/A	0.0025	0.9973_**805**	0.9973_**669**	0.9973_**674**	0.9973_**735**
N/A	0.05	0.99_**41937**	0.99_**39496**	0.99_**39584**	0.99_**40763**
N/A	0.2	0.98_**41519**	0.98_**32756**	0.98_**33103**	0.98_**37758**
N/A	0.5	0.96_**41006**	0.96_**22667**	0.96_**23520**	0.96_**34960**

III.E Applicability to Codes beyond SEC-DED

Double-error-correction-triple-error-detection (DECTED) codes are capable of correcting double-bit errors and can relieve the demands of error scrubbing systems and bit-interleaving schemes. The Spatial-Temporal Reliability Model is still applicable to these codes because it only predicts the probability of an MBU occurring, not whether or not the MBU is correctable. If some MBUs are correctable, and the reliability designer is only interested in the probability of a non-correctable MBU occurring, then it is necessary to categorize MBUs and revise the events defined in Table 1. In this section, the Spatial-Temporal Reliability Model will be extended to calculate the reliability of a memory word with a DECTED code, where a triple-multi-bit-upset (MBU_3) is required to defeat the code. The new definitions are shown in Table 5. The accuracy of the equations presented in this section are presently under investigation.

Table 5: Revised Experiment Event Definition for DECTED Codes

Event	Definition	Description
S	$\eta \geq 1$	A single-bit upset (SBU) occurred. That is, at least one SEU occurred during irradiation.
M_2	$X \cup T$	A multi-bit upset (MBU_2) occurred. That is, at least two bits were flipped during irradiation within the same scrubbing interval.
X_2	$\eta \geq 2$	A spatial multi-bit upset ($XMBU_2$) occurred. That is, at least one particle caused at least two bits to flip simultaneously.
T_2	$\zeta \geq 2$	A temporal multi-bit upset ($TMBU_2$) occurred. That is, at least two particles arrived within the same scrubbing interval, and each caused at least one bit to flip.
M_3	$X_3 \cup T_3 \cup (X_2 \cap T_2)$	A triple-multi-bit upset occurred (MBU_3) occurred. That is, at least three bits were flipped during irradiation within the same scrubbing interval.
X_3	$\eta \geq 3$	A triple-spatial multi-bit upset ($XMBU_3$) occurred. That is, at least one particle caused at least three bits to flip simultaneously.
T_3	$\zeta \geq 3$	A triple-temporal multi-bit upset ($TMBU_3$) occurred. That is, at least three particles arrived within the same scrubbing, and each caused at least one bit to flip.

To find the word's reliability against MBU_3, the probability of M_3 must be found. Unlike MBU_2 derived in the SECDED case, there are three ways in which M_3 can occur. Either X_3 can occur, T_3 can occur, or both X_2 and T_2 occur simultaneously. This relationship is shown in (41).

$$M_3 = X_3 \cup T_3 \cup (X_2 \cap T_2) \tag{41}$$

We will assert here that there do exist equations which can be used to express the probability of X_3 and T_3, so long as the assumptions and hypotheses defined previously in the chapter still apply. It is necessary to resolve the union in Equation (41) into individual

probabilities of X_3, T_3, X_2, T_2, and S. This is partially completed in the following calculation steps which result in Equation (42).

$$P(M_3) = P\big(X_3 \cup T_3 \cup (X_2 \cap T_2)\big)$$

$$\begin{gathered} P(M_3) = P(X_3) + P(T_3) + P(X_2 \cap T_2) \dots \\ -P(X_3 \cap T_3) - P\big(X_3 \cap (X_2 \cap T_2)\big) - P\big(T_3 \cap (X_2 \cap T_2)\big) \dots \\ +P(X_3 \cap T_3 \cap (X_2 \cap T_2)) \end{gathered}$$

$$X_3 \subset X_2; \quad T_3 \subset T_2$$

$$\begin{gathered} P(M_3) = P(X_3) + P(T_3) + P(X_2 \cap T_2) \dots \\ -P(X_3 \cap T_3) - P(X_3 \cap T_2) - P(X_2 \cap T_3) \dots \\ +P(X_3 \cap T_3) \end{gathered}$$

$$P(M_3) = P(X_3) + P(T_3) + P(X_2 \cap T_2) - P(X_3 \cap T_2) - P(X_2 \cap T_3) \tag{42}$$

By using the assumptions that support the Level-2 XTR model, it is possible resolve the third term in Equation (42) as was done in Equation (18). The fourth and fifth terms, however, need another way to be resolved. To do this, a similar assumption will be used. First, the last three terms of Equation (42) will be rewritten as conditional probabilities using the law of total probability, as shown in Equation (43).

$$\begin{aligned} P(X_2 \cap T_2) &= P(X_2 \cap T_2|S)P(S) + P\big(X_2 \cap T_2|\overline{S}\big)P(\overline{S}) \\ P(X_3 \cap T_2) &= P(X_3 \cap T_2|S)P(S) + P\big(X_3 \cap T_2|\overline{S}\big)P(\overline{S}) \\ P(X_2 \cap T_3) &= P(X_2 \cap T_3|S)P(S) + P\big(X_2 \cap T_3|\overline{S}\big)P(\overline{S}) \end{aligned} \tag{43}$$

Second, it will be recognized that, due to the subset relationships shown in Equation (44), that the second term of each row in Equation (43) is zero.

$$\begin{gathered} X_2 \cap T_2 \subset S \quad \rightarrow \quad X_2 \cap T_2|\overline{S} = \emptyset \\ X_3 \cap T_2 \subset S \quad \rightarrow \quad X_3 \cap T_2|\overline{S} = \emptyset \end{gathered} \tag{44}$$

$$X_2 \cap T_3 \subset S \quad \rightarrow \quad X_2 \cap T_3 | \overline{S} = \emptyset$$

$$\begin{aligned} P(X_2 \cap T_2) &= P(X_2 \cap T_2 | S) P(S) \\ P(X_3 \cap T_2) &= P(X_3 \cap T_2 | S) P(S) \\ P(X_2 \cap T_3) &= P(X_2 \cap T_3 | S) P(S) \end{aligned} \tag{45}$$

Next, a conditional independence assumption will be made to separate the intersections in Equation (45). This is stated formally in Equation (46). It remains to be shown whether the new assumptions in Equation (46) are reasonable.

$$\begin{aligned} P(X_2 \cap T_2 | S) &\approx P(X_2 | S) P(T_2 | S) \\ P(X_3 \cap T_2 | S) &\approx P(X_3 | S) P(T_2 | S) \\ P(X_2 \cap T_3 | S) &\approx P(X_2 | S) P(T_3 | S) \end{aligned} \tag{46}$$

Finally, the definition of conditional probability can be used to simplify Equation (46) down to Equation (47).

$$\begin{aligned} P(X_2 | S) P(T_2 | S) &= \left(\frac{P(X_2 \cap S)}{P(S)} \right) \left(\frac{P(T_2 \cap S)}{P(S)} \right) = \frac{P(X_2) P(T_2)}{P(S)^2} \\ P(X_3 | S) P(T_2 | S) &= \left(\frac{P(X_3 \cap M_2)}{P(S)} \right) \left(\frac{P(T_2 \cap M_2)}{P(S)} \right) = \frac{P(X_3) P(T_2)}{P(S)^2} \\ (X_2 | S) P(T_3 | S) &= \left(\frac{P(X_2 \cap M_2)}{P(S)} \right) \left(\frac{P(T_3 \cap M_2)}{P(S)} \right) = \frac{P(X_2) P(T_3)}{P(S)^2} \end{aligned} \tag{47}$$

These three expression can then be reinserted into Equation (45) to obtain expressions for the intersection terms, which can then be inserted into Equation (42) to produce the probability of M_3 as shown in Equation (48).

$$P(M_3) = P(X_3) + P(T_3) + \frac{P(X_2)P(T_2) - P(X_3)P(T_2) - P(X_2)P(T_3)}{P(S)} \quad (48)$$

This expression is much more complicated than the corresponding expression for the probability of M_2, but it is analytically tractable and computationally feasible.

One glaring limitation of the Spatial-Temporal Reliability Model is its incompatibility with error correction codes which leverage the spatial locality of XMBU patterns. Single-error-correction-double-adjacent-error-correction-double-error-detection (SEC-DAED-DED) codes have been presented [41]. In words which are protected by these codes, MBUs can no longer be simply categorized by the number of upset bits, but instead depend on the relative position of the upset bits as well. The Spatial-Temporal Reliability Model will accurately predict the number of MBUs in any data word, but it is insufficient to determine which MBUs are correctable and which ones are not correctable.

III.F Impact of Spatial-Temporal Reliability Model

Despite its inherent constraints and underlying assumptions, the Spatial-Temporal Reliability Model produces good agreement with Monte-Carlo simulation results. The XTR model is able to provide quantitative insight into the design of reliable memory words which face the double threat of TMBU and XMBU. This is thanks to its separation of TMBU and XMBU into its constituent reliability-versus-time functions. This allows future model developers to create new expressions for $R_X(t)$ and $R_T(t)$ which account for more complicated radiation environments and combine them to form $R_M(t)$. The XTR model will be especially useful for estimating reliability in high-flux single-event environments which are difficult or impossible to replicate in laboratories. This is because a low-flux, high-fluence experiment can be performed instead to estimate flux-independent values for

$R_X(t)$, $R_S(t)$, and σ_S, the last of which can be used to calculate $R_T(t)$ for any flux and scrubbing frequency.

CHAPTER IV

PULSE ARRIVAL-TIME INVARIANT RELIABILITY MODEL FOR TEMPORAL MULTI-BIT UPSET

IV.A Problem Definition

A majority of the radiation effects literature which discusses the reliability of memories with error correction schemes focuses on the mean-time-to-failure (MTTF) metric, which is appropriate for long-duration radiation environments, but less helpful for short pulses of intense radiation. In these situations, it is more desirable to know reliability as a function of time, where reliability is defined as the probability of not failing. Memories with single-error correction codes (or even multi-error correction codes) must be periodically scrubbed of errors to prevent error accumulation, which in this work are referred to as temporal multi-bit upset (TMBU). In this chapter, a technique for calculating the reliability against TMBU is derived and presented for memories which use blind scrubbing. Two models will be presented. The first model accounts for the random arrival time of the radiation pulse, which is not accounted for in the radiation-effects literature due to its numerical insignificance in long-duration radiation exposures. After derivation, the model is analyzed, and it will be shown when its use is warranted. The second model is relatively simple and arises out of graphical analysis of the first. The second model can be used to quickly select design parameters which can be further tuned by the first model.

IV.B Related Work: The Saleh Model

To understand the implications of the Pulse-Arrival Time-Invariant (PATI) model, the Saleh model with blind deterministic scrubbing [33] will be rederived in this section. It was shown in Chapter III that the reliability of a memory word against TMBU (of two or more bits) is well approximated by Equation (49).

$$R_T(t) = be^{-\dot{b}\lambda_b t} - \dot{b}e^{b\lambda_b t} \quad (49)$$

Where λ_b is the SEU upset rate for any given bit, b is the number of bits in the word, and $\dot{b}$ is the number of bits minus one. As was noted in Chapter III, this model assumes that each bit encounters SEU independently of the other bits, and it also assumes that each upset cannot be "corrected" by another upset (see Upset Permanence Assumption from Chapter III).

While the expression in Equation (49) was referred to as the Saleh model, it is actually the most basic model developed by Saleh, et al. in [33]. That work extended the aforementioned expression to account for the effects of periodic scrubbing, both deterministic and non-deterministic. In the following discussion, an intuitive way of arriving at the Saleh model for blind periodic scrubbing will be given. Blind periodic scrubbing is focused upon because it is the most application-independent scrubbing technique and is the simplest to analytically describe. Additionally, it will be assumed that the scrubbing system operates without error and performs its read and write operations instantaneously.

If we recognize that $R_T(t)$ is the probably that a memory word with no errors will survive irradiation until time t with fewer than two errors, then the probability that two words will survive that same time period twice is $R_T(t)^2$. These two words can literally be

two different words, or it could be the same word which had its errors cleared in between trials. This is precisely what blind deterministic error scrubbing does. At this juncture we introduce the variable τ which is the time between scrubbing operations for any given word in a memory. We will also use the variable T to represent the total amount of time that word is irradiated, also known as the radiation pulse duration. If T is an integer multiple of τ, and we define $k = \frac{T}{\tau}$, it is reasonable to suppose that the probability of a word surviving TMBU is:

$$R_T(T,\tau) = [R_T(\tau)]^k \quad ; \quad T = k\tau \;\; ; \;\; k \in 0, 1, 2, 3, \ldots \tag{50}$$

Where the R_T function with two input arguments represents the reliability against TMBU for a radiation exposure of T seconds when a blind scrubbing frequency of $1/\tau$ is applied. The R_T function with one input represents the reliability against TMBU without scrubbing, as shown in Equation (49). Next, we may extend the domain of $R_T(T,\tau)$ to include values of T where it is not an integer multiple of τ. This is achieved with Equation (51).

$$R_T(T,\tau) = [R_T(\tau)]^k [R_T(T - k\tau)] \quad ; \quad k = \left\lfloor \frac{T}{\tau} \right\rfloor \tag{51}$$

A graphical representation of the reasoning behind Equation (51) is shown in Figure 14. In the next section, it will be shown why this reasoning may be insufficient in some situations, but it will be assumed that it is correct in this section. Example plots of $R_T(T,\tau)$ as a function of scrubbing interval are shown in Figure 15 for varying bit-upset rates. The red line in the plot indicates the pulse duration, which is one millisecond. To the right of the red line, k is equal to zero, which means that scrubbing has no benefit whatsoever because

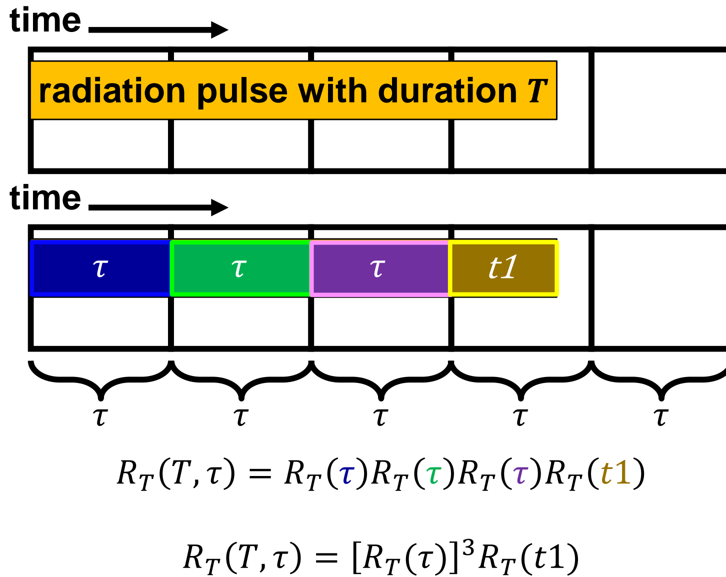


Figure 14: Representation of blind deterministic scrubbing with period τ dividing the radiation pulse duration into smaller time intervals. The chance of not encountering a TMBU during the whole radiation pulse is the product of the chances of not encountering a TMBU during any of the constituent time intervals.

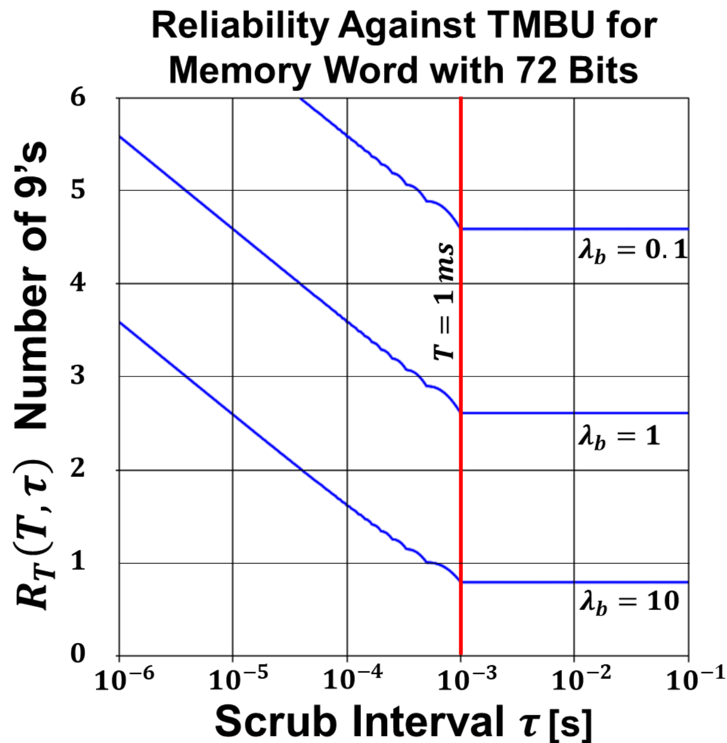


Figure 15: Evaluation of Equation (51) and Equation (49) plotted versus scrub interval τ for various bit-upset rate parameters λ_b, which have units of Hz. The red line marks the pulse duration T, which is 1 millisecond.

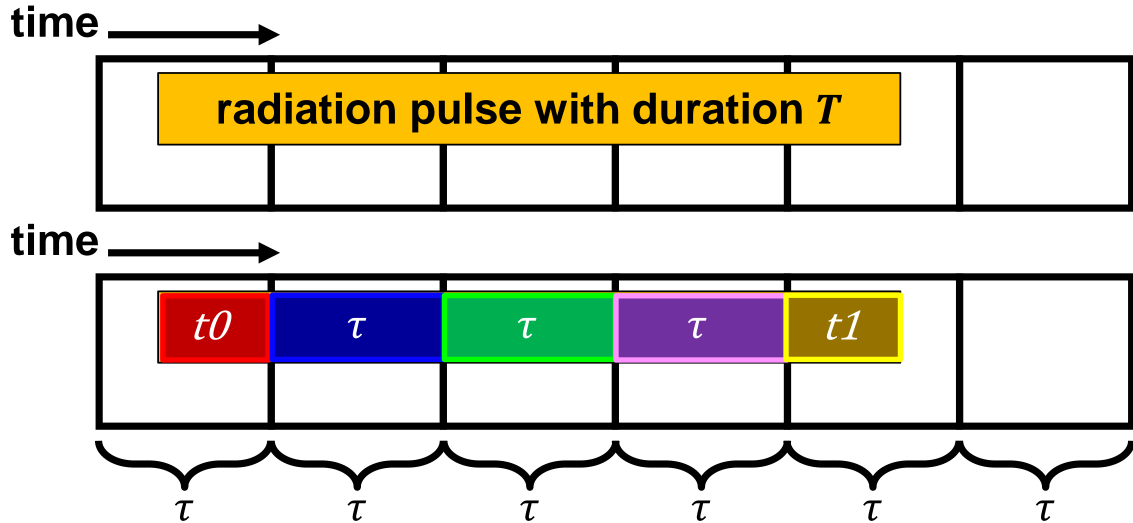
the radiation pulse duration is smaller than the scrubbing interval. To the left of the red line, the reliability increases steadily as the periodic scrubbing gets faster and faster. Though it is represented differently, this is the same result that was described in [33].

IV.D Derivation of the PATI Model

The first new model presented in this chapter is the Pulse-Arrival-Time-Invariant (PATI) Reliability Model, which calculates the probability of a temporal MBU occurring in a memory word with blind periodic scrubbing. It improves upon the reliability model given by Saleh, et al. by accounting for the randomness of the arrival time of a radiation pulse.

An implicit assumption used in Figure 14 is that the radiation pulse and first scrubbing interval begin as the same time. This is generally untrue in pulsed radiation environments but is a perfectly reasonable approximation for space environments where radiation exposure is present throughout the entire mission. This mathematically corresponds to T being many orders of magnitude greater than τ, which results in the second factor of Equation (51) becoming insignificant compared to the first.

This approximation does not hold well for brief bursts of single-event radiation, however. If the radiation pulse is not many orders of magnitude greater or smaller than the scrubbing interval, the random pulse arrival time can become important. A revision of Figure 14 which accounts for the random pulse arrival time is shown in Figure 16. As shown in the figure, there are now two time intervals which are shorter than the scrubbing interval. These are $t0$, the head of the radiation pulse, and $t1$, the tail of the radiation pulse.



$$R_T(T,\tau) = R_T(t0)R_T(\tau)R_T(\tau)R_T(\tau)R_T(t1)$$

$$R_T(T,\tau) = R_T(t0)[R_T(\tau)]^3 R_T(t1)$$

Figure 16: Revision of Figure 14 which accounts for the possibility of the radiation pulse arriving out of phase with the periodic scrubbing intervals. The size of the intervals *t0* and *t1* will be random and depend on when the radiation pulse arrives.

Because the length of the head and the tail have an inverse relationship that depends on the phase of the scrubbing cycle, it is necessary to assign a variable θ to represent that phase. θ is bounded on the interval $[0,\tau)$ because of the periodic nature of the scrubbing cycle. How θ affects the length of the head and the tail depends on the duration of the radiation pulse which is visualized in Figure 17. In Figure 17(a-b), the radiation pulse is shorter than a single scrubbing interval, and whether or not scrubbing takes effect at all depends on θ. In Figure 17(a), no scrubbing takes place, but in Figure 17(b), one scrub takes place, which improves the reliability. In Figure 17(c-d), the radiation pulse is longer than a single scrubbing interval, so it is guaranteed that at least one scrub will take place, but whether or not $(k-1)$ or k scrubbing cycles occur during the radiation pulse depends on θ. The distinction between the (c) and (d) case is very minor if k is very large but is quite

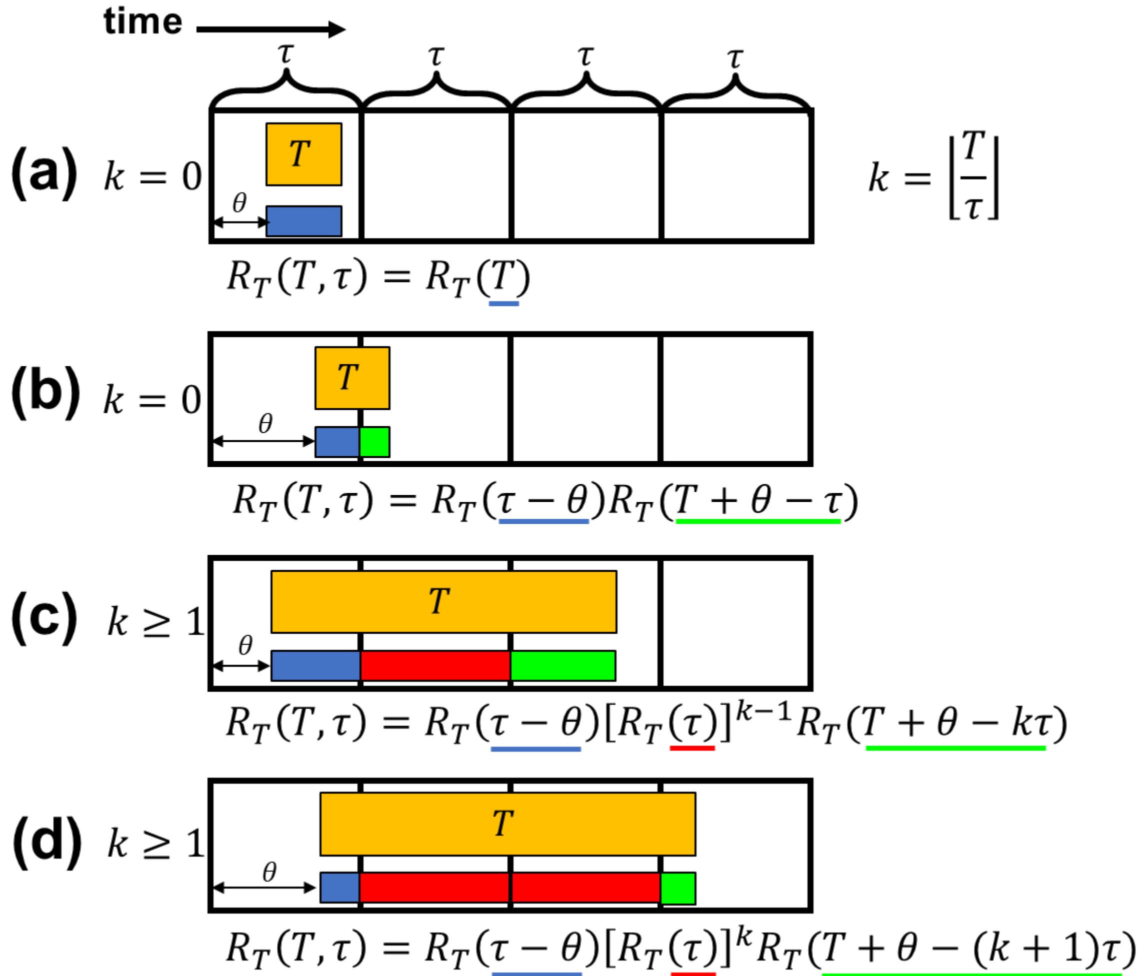


Figure 17: Four different cases which describe how the scrubbing interval interacts with the radiation pulse. The vertical black lines represent the scrubbing events. The orange rectangle represents the beginning and end of the radiation pulse. Beneath the orange rectangle, there are rectangles which represent the radiation pulse duration being broken up into smaller intervals because of periodic scrubbing. The blue intervals are the "head" of the radiation pulse, the green intervals are the "tail" of the radiation pulse, and the red intervals, which are equal to τ, are the body of the radiation pulse. Beneath each of the four cases, an equation is shown represents the overall reliability against TMBU. The colored highlights in the equation correspond to the colored intervals shown on the timeline.

significant in other situations. The equations from Figure 17 are summarized in Equation (52).

$$R_T(T,\tau,\theta) = \begin{cases} k = \left\lfloor \frac{T}{\tau} \right\rfloor = 0: & \begin{cases} \theta \leq \tau - T: & R_T(T) \\ \theta \geq \tau - T: & R_T(\tau-\theta)R_T(T+\theta-\tau) \end{cases} \\ k = \left\lfloor \frac{T}{\tau} \right\rfloor \geq 1: & \begin{cases} \theta \leq \tau + k\tau - T: & R_T(\tau-\theta)[R_T(\tau)]^{k-1}R_T(T+\theta-k\tau) \\ \theta \geq \tau + k\tau + T: & R_T(\tau-\theta)[R_T(\tau)]^{k}R_T(T+\theta-k\tau-\tau) \end{cases} \end{cases} \tag{52}$$

Because θ is truly random and it is not practical to have any information about its value, we can treat it like a random variable with the uniform probability distribution in Equation (53), where $u(t)$ denotes the unit step function.

$$P(\theta) = u(\theta)u(\tau-\theta)\left(\frac{1}{\tau}\right) \tag{53}$$

To eliminate θ from Equation (52), we must integrate the expression with respect to θ over the entire interval $[0,\tau)$, as shown in Equation (54). The result of this integration is shown in Equation (55).

$$R_T(T,\tau) = \int_{-\infty}^{\infty} P(\theta)R_T(T,\tau,\theta)d\theta = \frac{1}{\tau}\int_0^{\tau} R_T(T,\tau,\theta)d\theta \tag{54}$$

$$\boldsymbol{R_T(T,\tau)} = \begin{cases} \boldsymbol{k=0}: & \left(\boldsymbol{1-\frac{T}{\tau}}\right)\boldsymbol{R_T(T)} + \boldsymbol{\frac{T}{\tau}Q(T)} - \boldsymbol{Y(T)} \\ \boldsymbol{k \geq 1}: & \boldsymbol{R_T(\tau)^{k-1}}\left(\left(\boldsymbol{1-\frac{T_k}{\tau}}\right)\boldsymbol{Q(T_k^-)} - \boldsymbol{Y(T_k^-)e^{\lambda_b T_k}}\right) + \boldsymbol{R_T(\tau)^k}\left(\boldsymbol{\frac{T_k}{\tau}Q(T_k)} - \boldsymbol{Y(T_k)}\right) \end{cases} \tag{55}$$

$$R_T(t) = be^{-\dot{b}\lambda_b t} - \dot{b}e^{-b\lambda_b t} \qquad \dot{b} = b-1 \qquad Y(t) = \frac{2b\dot{b}}{\lambda_b \tau}\left[e^{-\dot{b}\lambda_b t} - e^{-b\lambda_b t}\right]$$

$$k = \left\lfloor \frac{T}{\tau} \right\rfloor \qquad T_k = T - k\tau \qquad T_k^- = T - k\tau + \tau \quad Q(t) = b^2e^{-\dot{b}\lambda_b t} + \dot{b}^2e^{-b\lambda_b t}$$

The formula in Equation (55) is the PATI model that was sought in the derivation. While it is more complicated than the Saleh model proposed in [33], it is necessarily more accurate in some scenarios, as shown by the plot in Figure 18. This plot uses the same

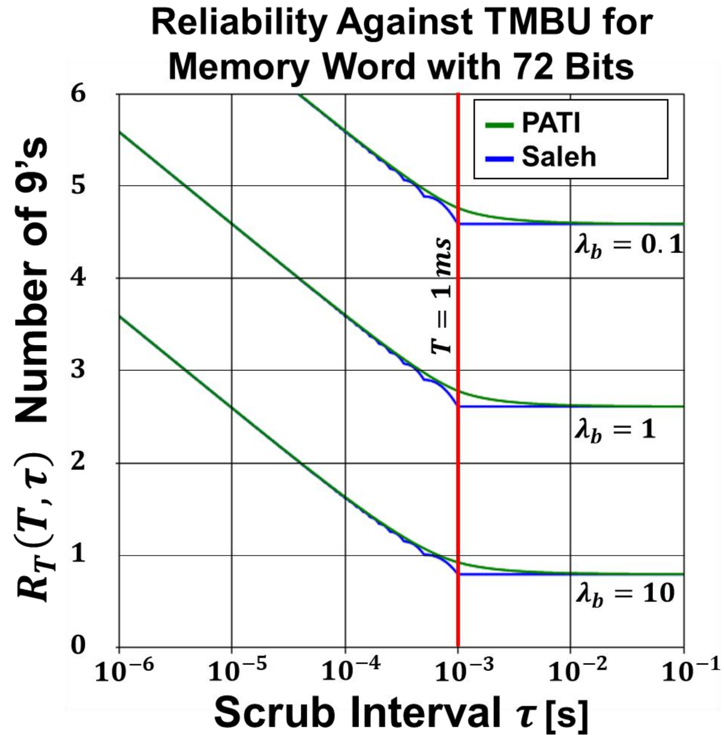


Figure 18: Comparison of the PATI model and the Saleh model.

parameters that were used in Figure 15 but it has a comparison between the PATI model and the Saleh model. As can be seen from the plot, the two models agree at the extremes, where $T \gg \tau$ or $T \ll \tau$, but disagree when T is on the same order of magnitude as τ. In this region, the PATI model predicts a gradual increase in reliability with increasing scrubbing frequency, while the Saleh model has sharp turns at every multiple of $1/T$ which are not necessarily physical.

IV.F Simplified TMBU Model

An interesting result of choosing to use a number-of-nines to express a circuit's vulnerability to TMBU is shown in Figure 19. The gold function which is plotted over the Saleh model and the PATI model consists of two straight lines which intersect at a 45° angle. The slope of the line to the left of the intersection is exactly one nine of reliability

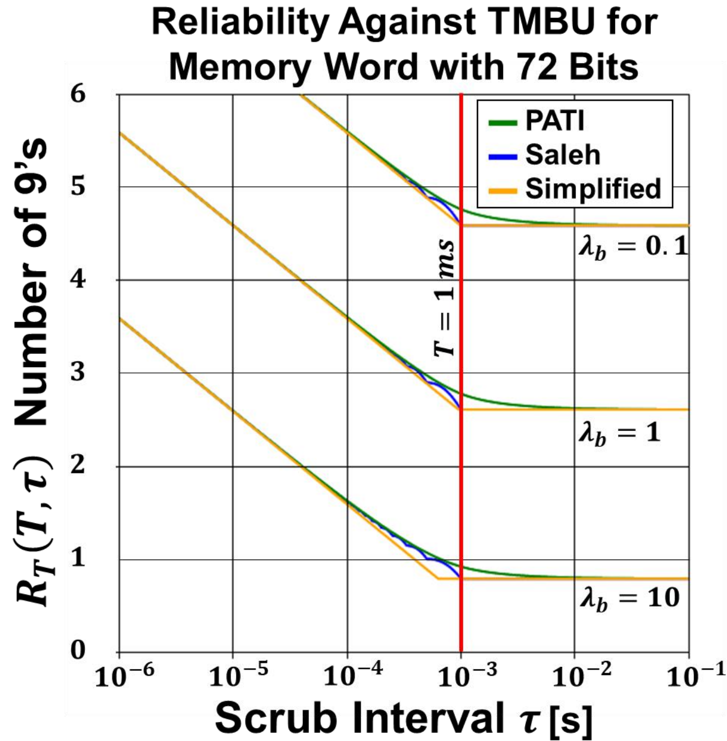


Figure 19: Comparison of the PATI and Saleh model, along with the simplified TMBU model. The simplified TMBU model consists of two straight lines which intersect at a 45° angle.

per decade of scrubbing frequency. These two lines can be found analytically by evaluating the Saleh model at any scrubbing interval which is several orders of magnitude above and below the pulse duration and drawing straight lines until they intersect. This intersection line usually takes place at the pulse duration on the τ-axis, but this might not be the case if R_T is less than one nine of reliability.

IV.H Conclusions

In this chapter, two new ways to estimate a memory word's reliability against TMBU were given, the PATI model and the simplified TMBU model, both of which build upon the Saleh model from [33]. The PATI model provides enhanced accuracy over the Saleh model in pulsed radiation environments. In addition to treating the unique case of short, intense radiation pulses, PATI model's improved accuracy will also become important to

future modeling efforts which seek to predict memory reliability for nonuniform radiation pulses, which might be approximated as an ensemble of short, uniform radiation pulses. Additionally, with the models presented and derived in this chapter, it will be possible to estimate reliabilities for high-flux environments by using the reliability measurements in low flux environments.

CHAPTER V

SIMULATION OF MULTI-NODE UPSET IN SPATIALLY-REDUNDANT CIRCUITS

In this chapter, a simulation-based technique developed in [42] to estimate a circuit's vulnerability to spatial multiple-node upset (XMNU) upset is presented. XMNU is closely related to spatial multiple-bit upset (XMBU) because they share a common physical mechanism, but XMNU is more complicated to analyze, because it involves circuit nodes that are electrically connected to each other, while bits in a memory word belong to different circuits that are independent of each other. The XMNU analysis presented in this chapter is applicable to XMBU analysis. After defining the most general way a single-event radiation vulnerability could be estimated, the specific solution developed in this work is presented along with simulation results. The developed simulation methodology is applied to three variants of a sub-50nm PDSOI DICE latch which are exposed to unidirectional, monoenergetic neutrons.

After examining the simulation results, experimental heavy ion data on similar DICE latches is presented. Both the heavy ion data and neutron simulation results shows a tight angular window of beam incidence where the spatial redundancy of the DICE latches can be defeated by simultaneous XMNU. The angular window being present in both neutron simulation and ion experimental data indicates that forward-scattered neutron secondary particles are an important mechanism in neutron-induced XMNU, which has critical implications for hardness assurance testing in those environments.

V.A Problem Definition

The vulnerability to a given single-event radiation effect is usually expressed as a cross-section. The magnitude of any kind of radiation effect cross-section σ may only be defined for a given circuit in a specific radiation environment and in a specific operating mode. If the radiation flux is directional, rather than isotropic, σ must also be defined at a variety of radiation incidence angles. More detailed studies might additionally define σ as a function of particle energy and initial linear energy transfer. In this work, each specification of σ will correspond to a specific circuit, a specific radiation effect, and a specific radiation environment. Furthermore, all of the radiation environments considered in this chapter have uniform, unidirectional fluxes consisting of one monoenergetic particle species.

V.A.1) Generalized Description of Single-Event Effect Vulnerability Analysis

A flowchart in Figure 20 shows the kinds of analyses that are required to compute an estimate of a single-event effect cross-section by tracking each particle's effect comprehensively. The first stage, radiation specification, entails specifying what kind of radiation will interact with the circuit under investigation. This includes, the particle species and their relative abundance, probability distributions for the kinetic energies of

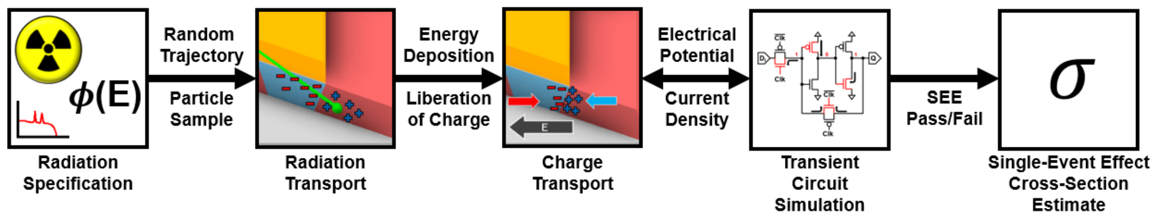


Figure 20: Flowchart showing a variety of physical and electrical analyses which operate at different levels of detail to calculate a single-event effect cross-section. Note that the arrow connecting charge transport simulation to transient circuit simulation is bidirectional.

each of the included species, and their directionality. Unless the radiation environment being studied is an extremely narrow particle beam, the first stage of the flowchart introduces an uncountable number of possible trajectories, and each trajectory that does occur will trigger a chain of events that flows from the left side of the flowchart to the right, resulting either in a radiation effect (failure) or no effect (success). Because of this, nearly all cross-section estimating techniques employ a Monte-Carlo method, where pseudorandom particle trajectories are applied to a 3D model of the circuit.

The purpose of developing a 3D model of the circuit under study is to have a 3D virtual environment for radiation transport simulation. To that end, it is necessary to define the borders of the virtual 3D space where virtual particles begin and end their trajectories. For a radiation transport simulation to accurately represent a radiation exposure, the boundaries of the virtual 3D space must surround the entire circuit under test and be large enough to include all particle trajectories which could interact with the circuit. The drawbacks to making the boundaries too large is that the radiation transport simulator will waste time and electricity calculating particle trajectories that miss the target and don't interact with the circuit. The random particle generation process which is constrained to a specific area gives rise to the concept of virtual fluence. Virtual fluence Φ_v is defined in Equation (56).

$$\Phi_v = \frac{N_{particle}}{A_{random}} \tag{56}$$

Where $N_{particle}$ is the number of random particles simulated and A_{random} is the randomization area.

In the context of single-event upset analysis, radiation transport simulation is used to determine the quantity and location of energy deposited in the circuit by each simulated particle as well as any secondary particles that are created by the primary particle.

Typically, ionizing energy deposited in active semiconductor regions of the integrated circuit are of greatest concern, because this energy is efficiently converted into electron-hole pairs which cause single-event transients. The conversion from deposited energy to liberated free carrier charge in silicon is approximately $44\ fC/MeV$.

When radiation transport simulation indicates that a particle has deposited energy (and liberated charge), it must be determined how that charge affects circuit operation and any data signals maintained by the circuit. The most thorough way to do this computationally is to perform a mixed-mode TCAD and SPICE simulation, which should predict how the deposited charge interacts with the operating voltages of the circuit with a high degree of accuracy. However, performing this analysis for every transported radiation particle is computationally expensive.

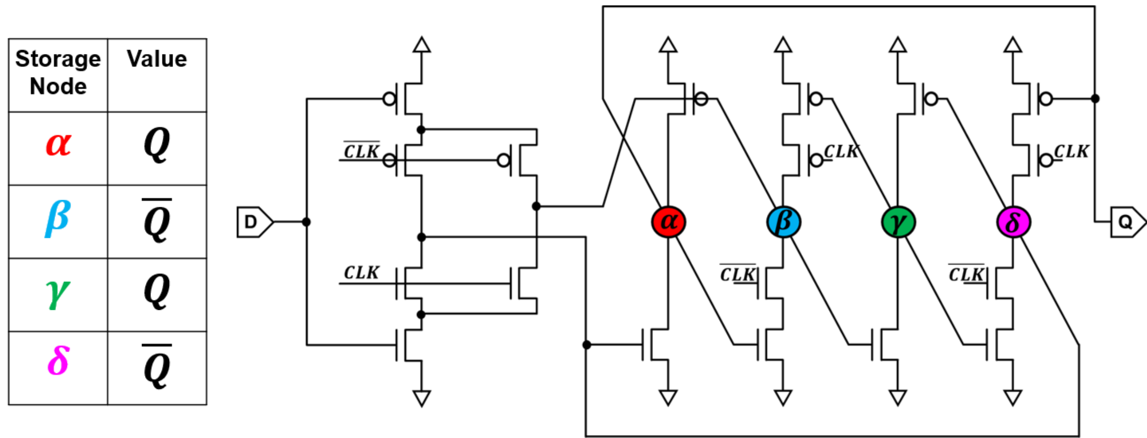
After the generated charge is collected and the circuit returns to a steady state, the logical impact of the randomly sampled radiation particle can be evaluated and tallied. After repeating this analysis for millions or billions of particles, it becomes reasonable to estimate single-event effect cross-sections from the ratio of simulated particles which did and did not cause the particular single-event effect. By counting the number of radiation transport trials that result in a certain outcome, and dividing it by the virtual fluence, it is possible to estimate the cross-section for that outcome:

$$\sigma_{outcome} = \frac{N_{outcome}}{\Phi_v} = \frac{A_{random} \times N_{outcome}}{N_{particle}} \tag{57}$$

If the failure being analyzed is XMNU or XMBU rather than SBU, then each stage of the flowchart each become much more complex. In section V.B, a technique to estimate XMNU cross-sections will be shown which can make reasonable compromises between physical accuracy and computational efficiency.

V.A.2) Description of Subject Circuit: Sub-50 nm SOI DICE Latch

The subject of the analysis in this work is a sub-50nm SOI DICE latch. The DICE latch schematic is shown in Figure 21. When the latch is in storage mode (CLK = 0), there are four transistors which are of interest. Which four these are depends on whether the latch is storing a logic 0 or a logic 1. Figure 22 shows which nodes are vulnerable for the Q = 1, D = 1 logic state. Thanks to their interlocking design, each vulnerable transistor lacks the



Storage Node	Value
α	Q
β	$\overline{Q}$
γ	Q
δ	$\overline{Q}$

Figure 21: Schematic of DICE latch studied in this chapter. There are four storage nodes driven by inverters which interlock when CLK = 0.

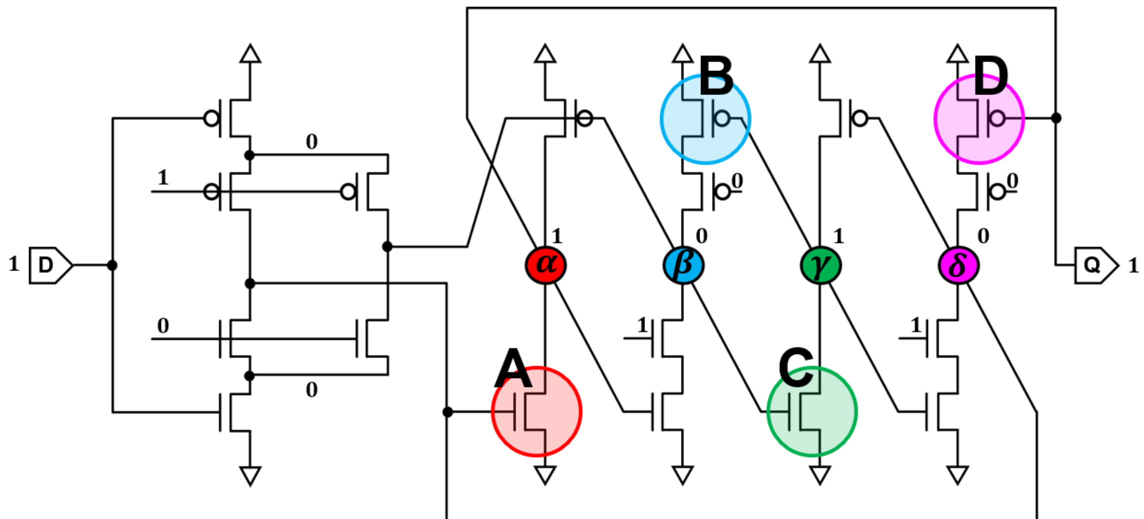


Figure 22: Vulnerable transistors are highlighted as **A**, **B**, **C**, **D** when the DICE latch is in the D = 1, Q = 1 state. If two or more of these transistors are struck simultaneously by a single-event, then the spatial redundancy could fail, and the latch could upset.

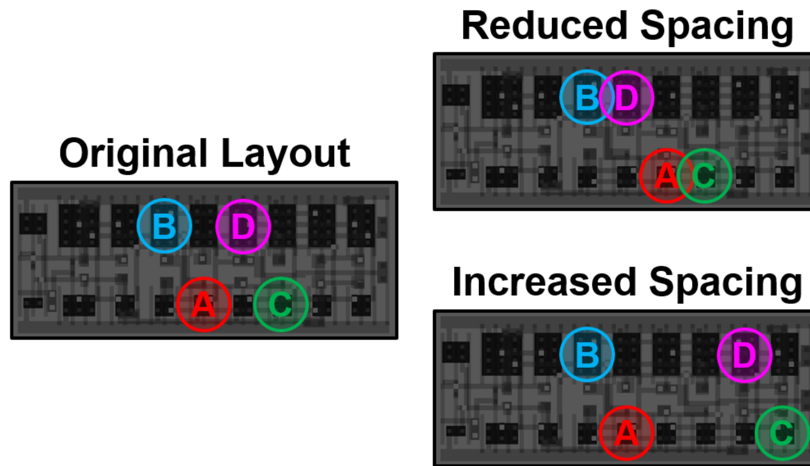


Figure 23: Three layout implementations of the sub-50nm DICE latch. The front-end-of-line is the same for all three circuits, but the interconnect gives the transistors different purposes. The locations of the vulnerable transistors from Figure 22 are highlighted.

ability to individually upset the DICE latch; two or more of them must simultaneously collect charge to flip the state of the DICE latch.

There were three layout implementations of the DICE latch used in this work. The transistor positions are identical in each layout, but the interconnect gives each transistor a different purpose in each layout. Because of the varying interconnections, the vulnerable transistors in each layout have different positions. The positions of the vulnerable transistors for storage logic state 1 are shown in Figure 23 for the three layouts. It is expected that for certain flux directions, the relative proximity of vulnerable transistor pairs will play an important role in the likelihood of multiple-node upsets.

V.B Simulation Technique used in this Work

The simulation technique used in this work is summarized in Figure 24. The advantage of this technique is that the computationally expensive mixed-mode charge transport simulation from the full-stack analysis in Figure 23 is avoided. There are three modular parts of this analysis technique which work together to make the calculation of σ_X as accurate and efficient as possible. This is done by performing radiation transport simulation with MRED and recording every event which results in multiple sensitive transistors collecting charge. These multi-device charge deposition records are then compared against two-dimensional critical charge plots to determine if each radiation event results in the DICE latch flipping or not. The two-dimensional critical charge plots are generated by performing a sweep of transient SPICE simulations with single-event enabled transistor compact models. In the following subsections, each part will be expounded upon along with relevant simulation results for the sub-50nm DICE latch test subject.

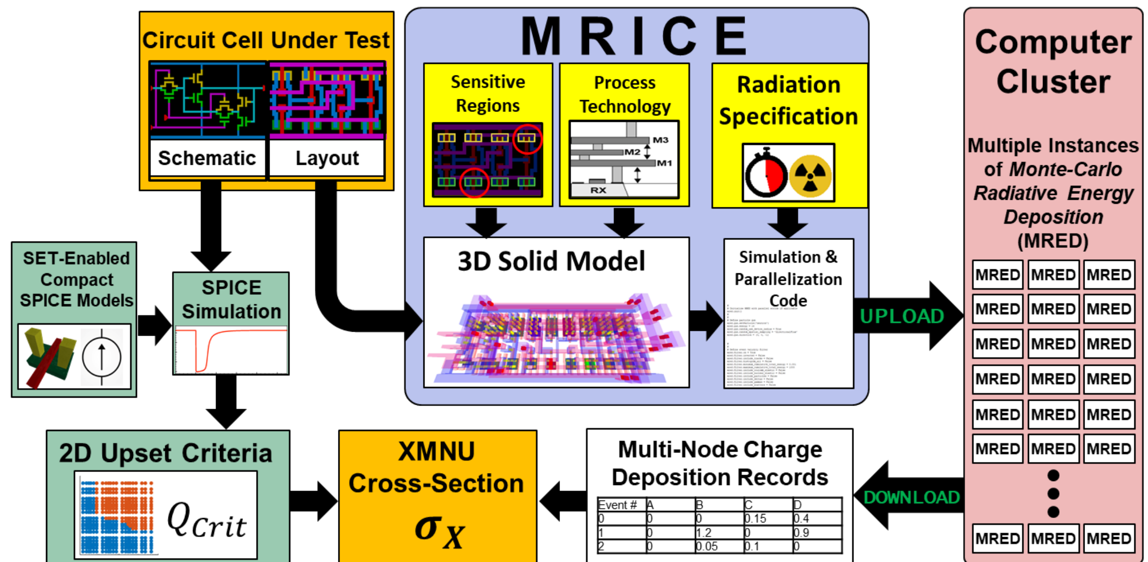


Figure 24: Flowchart that summarizes the technique used in this work. The circuit's XMNU cross-section is calculated by combining the results of SPICE simulation and radiation transport simulation.

V.B.1) 3D Circuit Construction and Radiation Transport Setup

There exist several CAD tools that can produce 3D models of integrated circuits cells. In this work, a new software tool called Monte-Carlo Radiation Transport with Integrated Circuit Emphasis (MRICE) was developed to perform this task [42]. MRICE is unique because it is specially designed to facilitate the use of MRED. MRICE does this by augmenting a circuit's 2D layout with vertical technology data to produce a 3D model which can be previewed. Within the 3D model, the user may specify which parts of the circuit are sensitive to single-event charge deposition. After the 3D model is constructed, the user may specify the type of radiation environment that is simulated. MRICE then converts the virtual radiation experiment into a self-contained script which may be uploaded to a computer cluster with MRED installed on it.

The 3D model of the DICE latch constructed by MRICE is shown in Figure 25. The virtual 3D boundary which encloses the entire circuit is a sphere, as shown in Figure 26.

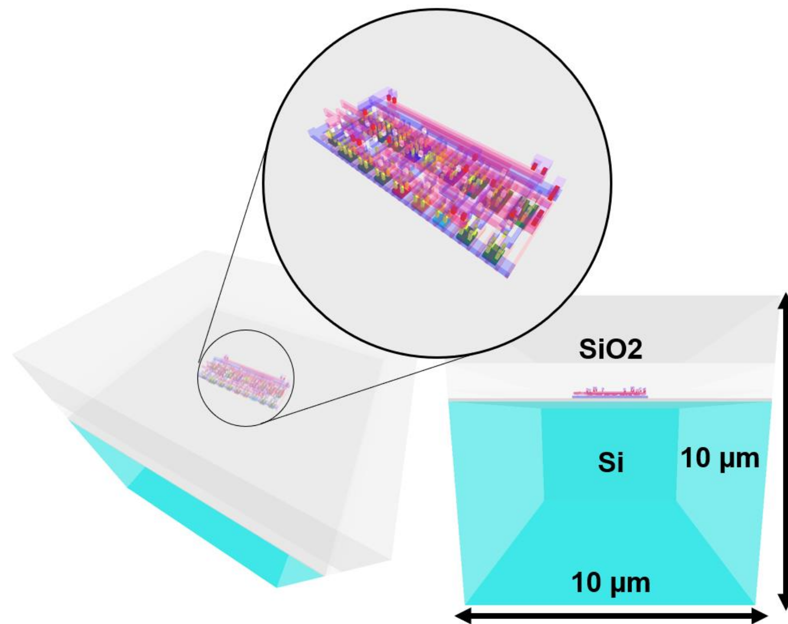


Figure 25: 3D model of the DICE latch studied in this work. Silicon-dioxide is used as a substitute for back-end-of-line dielectric because of its similar physical properties.

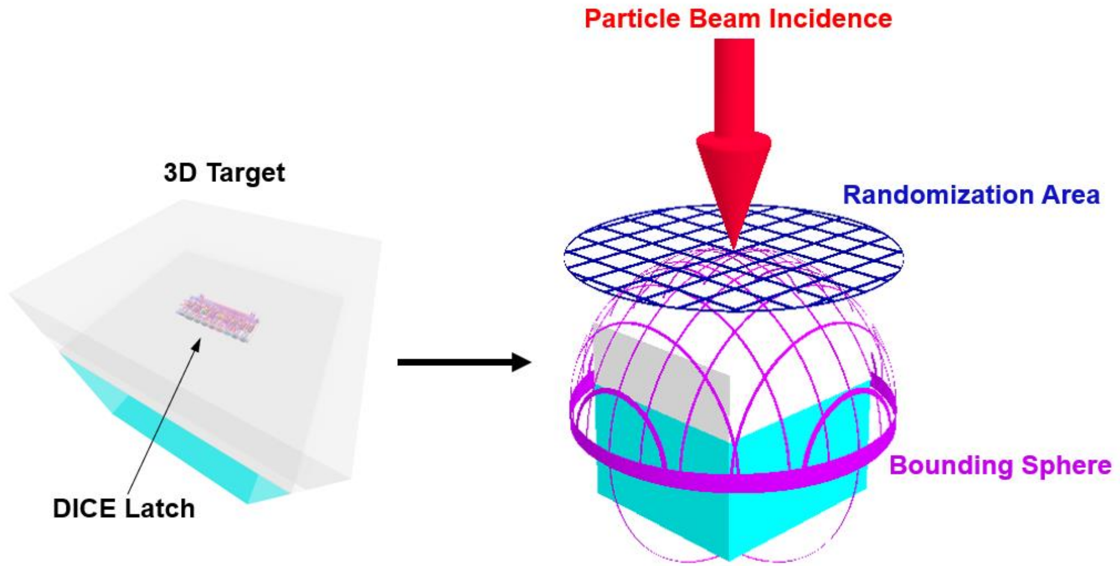


Figure 26: Visualization of the virtual boundary which surrounds the 3D target. All simulated particles exist within the purple bounding sphere shown. When a particle is generated during Monte-Carlo simulation, a random coordinate on the randomization area is chosen, and that point is projected onto the bounding sphere.

Because a unidirectional neutron flux is being simulated, the randomization area for any flux incidence is the sphere projected onto the incidence plane, which is a circle with area given by Equation (58). Because the 3D target structure is a cube, the diameter of the bounding sphere is the same as the diagonal of the cube.

$$A_{random} = \pi r^2 = \pi \left(\frac{d}{2}\right)^2 = \pi \left(\frac{\sqrt{3 \times (10\mu m)^2}}{2}\right)^2 = 75\pi\ \mu m^2 = 7.5\pi \times 10^{-7}\ cm^2 \quad (58)$$

The virtual neutron fluence in the simulation is therefore the number of simulated particles divided by $7.5\pi \times 10^{-7} cm^2$.

One last technical point which is often omitted in radiation effects literature is the translation between the virtual beam incidence in radiation transport simulation and the real beam incidence in experimental testing. In a real radiation beam experiment, the device under test is typically placed on a circuit board on an adjustable mount which can tilt and

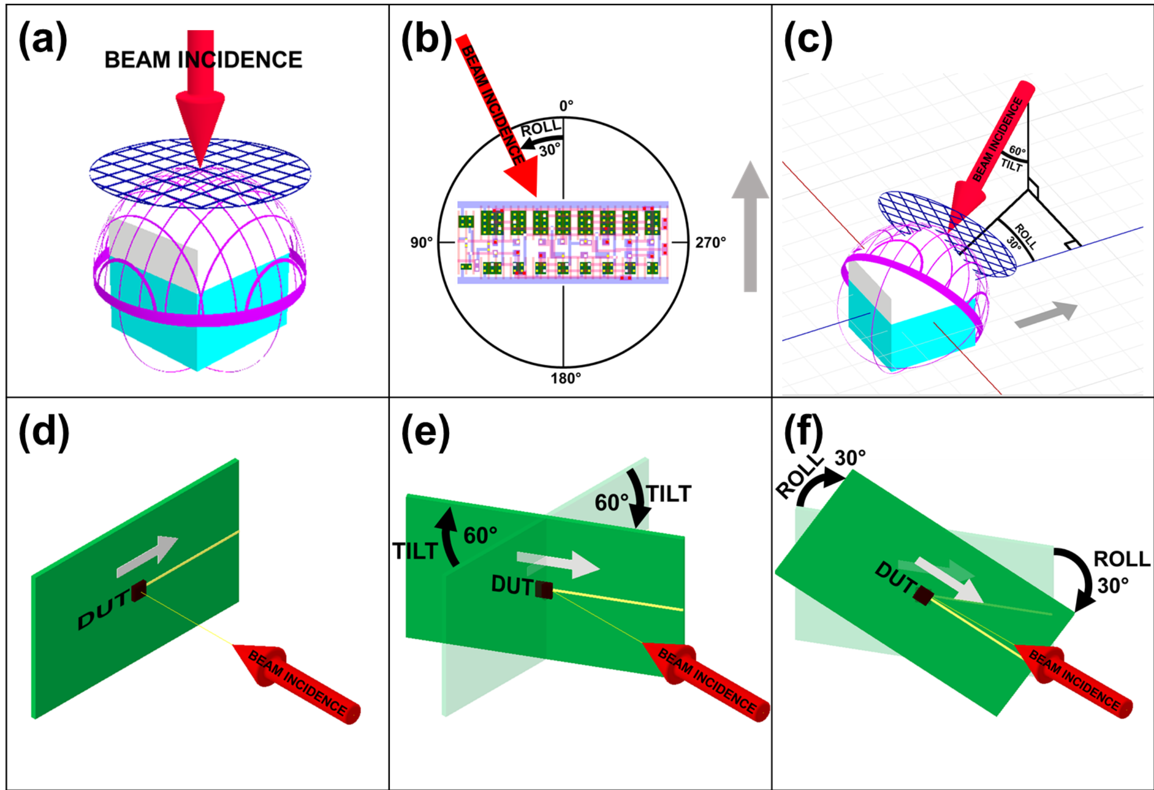


Figure 27: In (a), the virtual fluence has a tilt and roll of 0°, otherwise known as direct incidence. In (b), the virtual fluence roll angle is shown with respect to layout orientation from a top-down view. In (c), the virtual fluence is shown in 3D with the tilt of 60° and a roll of 30°. In (d-f), an illustration of a device-under-test (DUT) is shown. The DUT is mounted on a green circuit board. (d) shows normal incidence, (e) shows a tilt of 60°, while (f) shows a tilt of 60° and a roll of 30°. In (b-f), the gray arrow points in the same direction with respect to the vertical polysilicon gates in the circuit layout.

roll, while the radiation beam remains stationary. In a radiation transport simulation, the 3D circuit model remains stationary and the radiation beam incidence orbits around it. Figure 27 shows how a tilt of 60° and a roll of 30° is applied in a virtual radiation beam and in a real radiation beam.

V.B.2) Monte-Carlo Radiation Transport with MRED

In this work, MRED was programmed to perform unidirectional, monoenergetic neutron simulations, at a variety of tilt and roll angles. Because MRED was used on Vanderbilt University's ACCRE cluster [43], it was possible to run many instances of

MRED in parallel and perform approximately two billion randomly selected neutron trajectories for each tilt and roll angle examined, of which there were 22. This corresponds to a virtual fluence of approximately $8.5 \times 10^{14} \frac{neutrons}{cm^2}$. In each particle simulation trial, MRED reports the amount of energy which was deposited in each transistor of the DICE latch under test. For each simulated particle that deposited energy in two or more transistors, the corresponding amount of charge in each transistor was recorded in a list.

It was known from previous simulations that dual-node neutron-induced single-event upsets are very rare compared to single-node upsets, so a variance reduction technique known as hadronic cross-section biasing was used. MRED provides this optional functionality. Hadronic cross-section biasing causes primary particles such as neutrons to have a greater chance of interacting with an atomic nucleus and producing secondary particles [44]. This probability adjustment is accounted for in post-processing by providing a statistical weight for the event which is less than one. In this simulation, a hadronic cross-section bias of 200 was used, which means that every simulated neutron which causes an XMNU due to the enhanced cross-section will be counted as 0.005 toward the value of $N_{outcome}$ from Equation (57). If a neutron causes an XMNU because two enhanced interactions took place in its trajectory, then its statistical significance is divided by 200 twice and counted as 0.000025. Hadronic cross-section biasing is useful for Monte-Carlo radiation transport simulations with neutrons because it reduces the number of times that the simulator calculates the trajectory of a neutron which doesn't produce any secondary particles. However, it is important not to use such a high bias factor that the number of neutrons are unrealistically depleted as they pass through the target structure [44].

V.B.3) Dual-Node Single-Event-Transient SPICE Simulation

The cross-section estimation technique discussed in this work does not perform charge transport simulation for each particle. Instead, a series of SPICE simulations were performed with single-event transient enabled models from [29]. In each of these simulations, a unique pair of transistors in the DICE latch suffered simultaneous single-

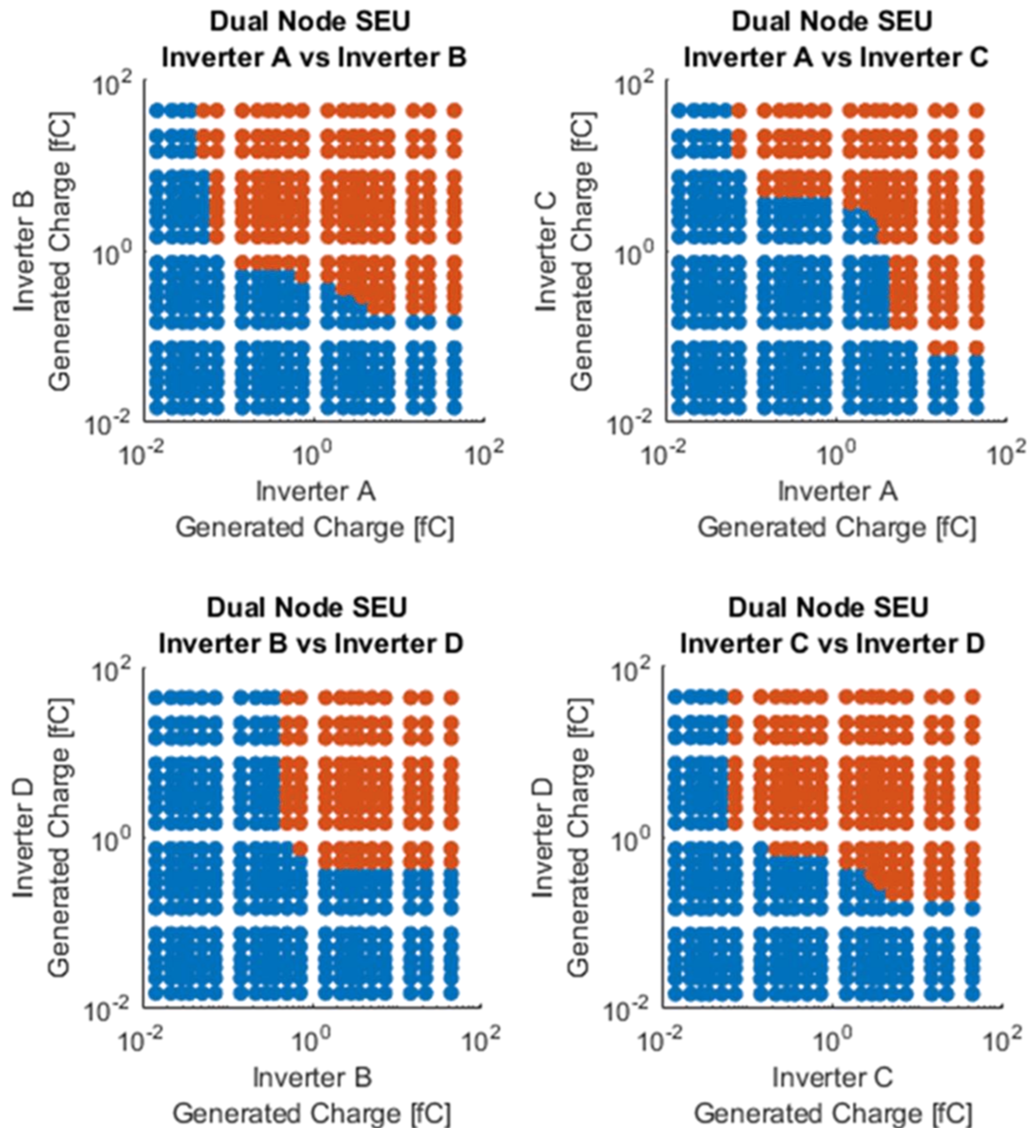


Figure 28: Dual-node critical charge for four different pairs of transistors from Figure 22. Red dots indicate that the DICE latch was upset, while blue dots indicate that the DICE latch did not. Other transistor combinations such as (A, D) or (B, C) were did not induce any upsets.

event transients, and each pair of SETs which resulted in the DICE latch were saved. Only four pairs of transistors were found to be vulnerable. These identified pairs correspond to the pairs shown in Figure 22 and Figure 23. For each of these pairs, two-dimensional sweeps were performed where the amount of deposited charge on each transistor was varied. By recording which pairs of charges caused the DICE latch to upset, it was possible to derive a “2D critical” charge. The 2D critical charge maps for each of the vulnerable pairs of transistors is shown in Figure 28.

By using the 2D critical charge maps from Figure 28, it is possible to determine which multi-node charge deposition records from MRED resulted in an XMNU-induced failure. While there are some accuracy limitations to using critical charge as an approximation, these limitations are minor because the DICE latch was constructed in an SOI technology; delayed charge collection effects such as charge sharing and field funneling do not affect SOI technology to the extent that they affect bulk technologies.

V.C Simulation Results

The technique described in the previous section was used on the described DICE latch to evaluate its sensitivity to tilt and roll angles of unidirectional neutron fluxes. For the tilt and roll angles shown in Figure 29, approximately two billion neutrons were simulated for each of the three layout variants. The corresponding XMNU-induced failure cross-section for each angle is shown in Figure 29 for each layout variant. Additionally, each failure was attributed to at least one vulnerable pair of transistors, so it is possible to see which vulnerable transistor pair caused the DICE latch to fail. The cross-sections for each sensitive pair of transistors are plotted in Figure 30 and Figure 31.

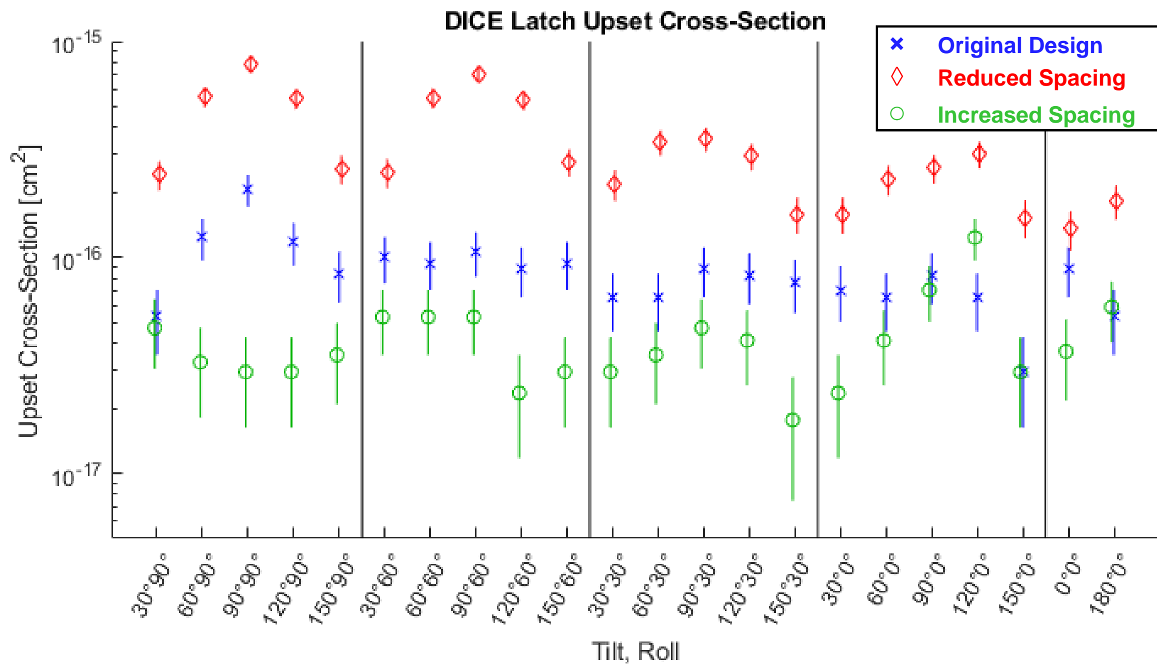


Figure 29: Estimations of the DICE latch's SEU cross-section for each of the three layout variants from Figure 23 when exposed to unidirectional, monoenergetic neutrons at various tilt and roll angles. Estimates which used the same roll angle are grouped by the vertical black lines. The error bars represent one standard deviation from the nominal datapoint.

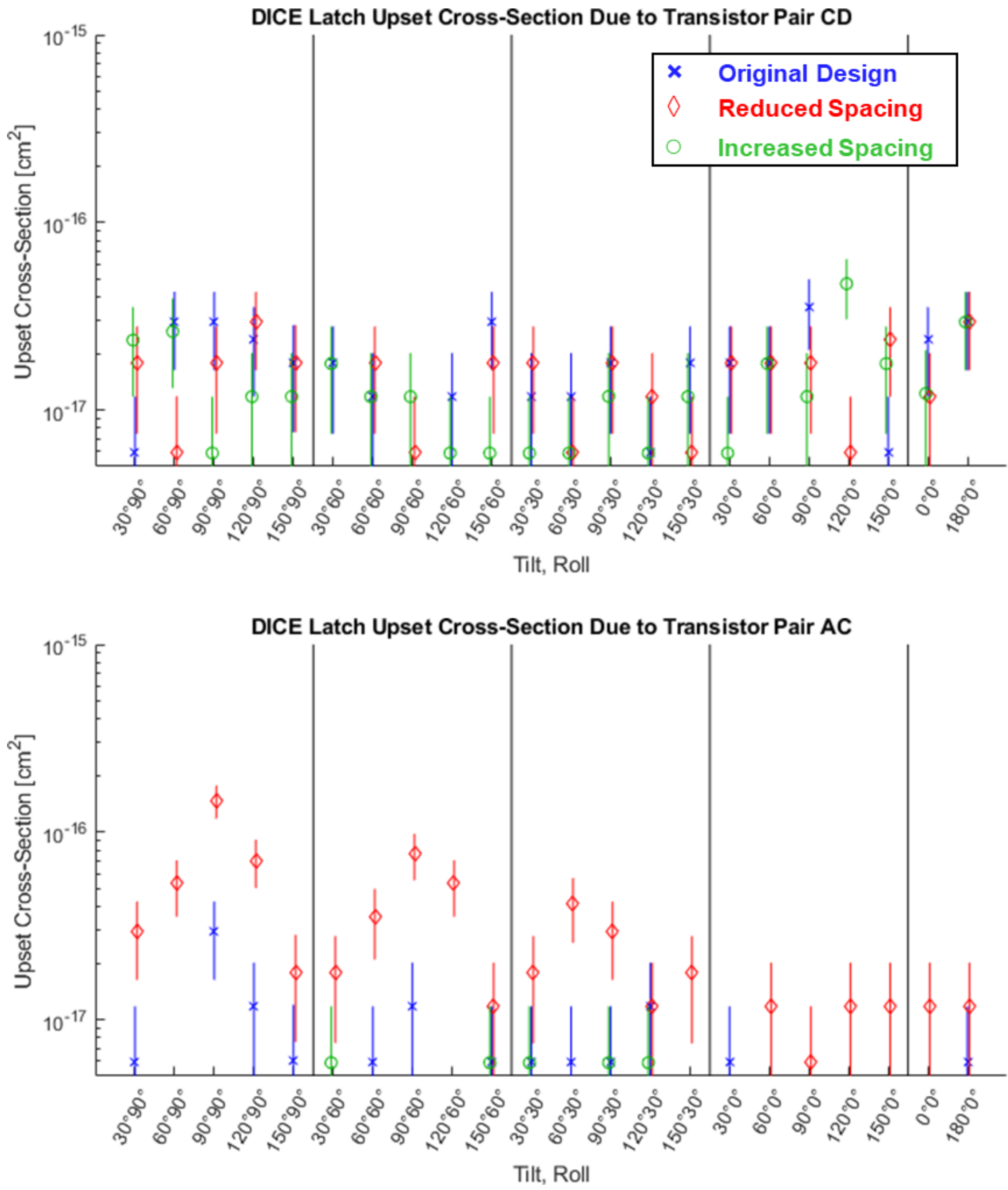


Figure 30: Estimations of the DICE latch's SEU cross-section for each of the three layout variants, if only XMNUs on the transistor pairs CD and AC are considered. No XMNUs were detected in the columns which don't have a datapoint.

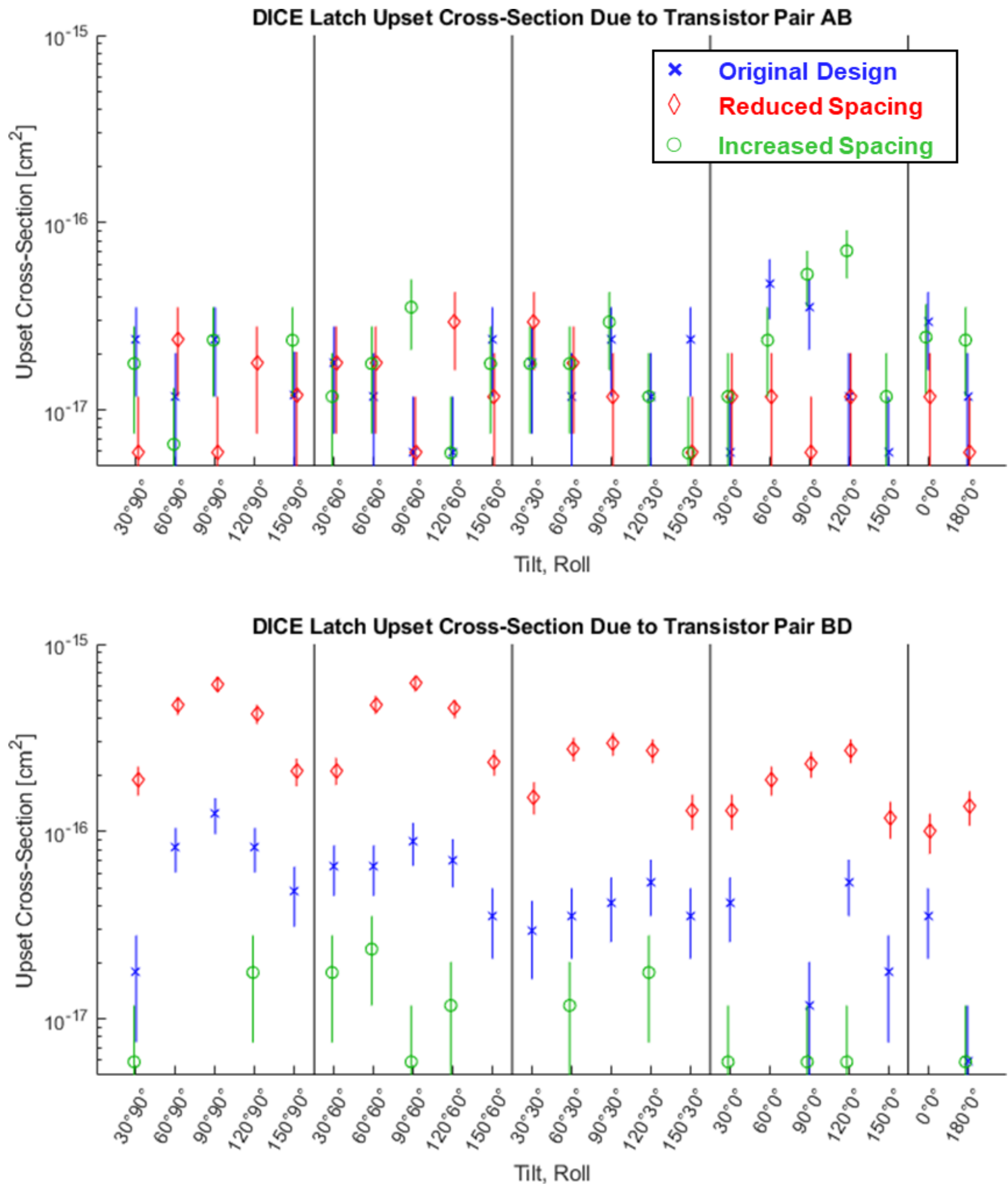


Figure 31: Estimations of the DICE latch's SEU cross-section for each of the three layout variants, if only XMNUs on the transistor pairs AB and BD are considered. No XMNUs were detected in the columns which don't have a datapoint.

As can be seen from the previous three figures, the DICE latch bears an angular vulnerability at a 90° roll and 90° tilt. This is exemplified by the contribution to the SEU cross-section by transistor pair BD and AC. This vulnerability is glaring in the layout variant with reduced spacing because these transistor pairs are adjacent in that variant. The increased spacing layout variant increases the distance between these vulnerable transistor pairs by approximately one micrometer, which lowers the tight angular vulnerability by about an order of magnitude.

V.D Experimental Evidence

In this section, experimental evidence which supports the results from the previous section will be presented. While neutron data is not yet available, single-event ion data is. Comparable DICE latches were fabricated in a comparable technology which were irradiated at the Texas A&M University cyclotron with 24.8 MeV/u nitrogen, argon, and krypton ions. Despite the differences in particle species and energies, the layout-induced multi-node vulnerability discovered in the simulation work of this chapter was also found in the experimental trial. Only the nitrogen data is shown in this section because it was created during the only beam run which used a blanket D=1, Q=1 logic state on all of the DICE latches. This state corresponds to the state of the DICE latches from the previous section and must be preserved to make a comparison between experiment and simulation.

V.D.1) Physical Implementation of the DICE Latch

The DICE latches were fabricated in a sub-50nm partially-depleted SOI technology and connected in pairs to form master-slave D-flip-flops. These flip flops were connected into three shift register chains which were each 32,768 bits long. Each shift register chain

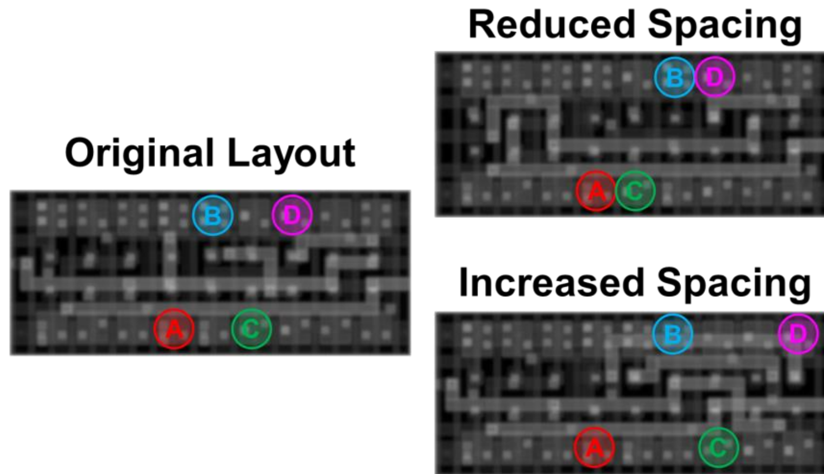


Figure 32: Three layout variants of the DICE latch that were tested. These three layouts correspond to the layouts from Figure 23. The vulnerable transistor locations are highlighted.

had one of the three layout variants with regular, reduced, and increased vulnerable transistor spacing as shown in Figure 32.

V.D.2) Test Setup

All tests were performed at the Texas A&M University Cyclotron Institute using the 24.8 MeV/u cocktail. The device under test featured a raised die fixture such that roll and tilt angles up to 90 degrees could be achieved in direct line of sight to the cyclotron beam without any obstruction to the die, including bonding wire. During testing, three different patterns were loaded into the shift registers: blanket "0s," blanket "1s," and a checkerboard pattern of alternating "0s" and "1s." The fidelity of the registers was tested before irradiation by loading a sequence of random values and comparing those values at the output of the registers.

V.D.3) Results

The experimentally measured cross-sections shown in Figure 33 were obtained for each layout variant by dividing the counted number of errors by the particle fluence which was

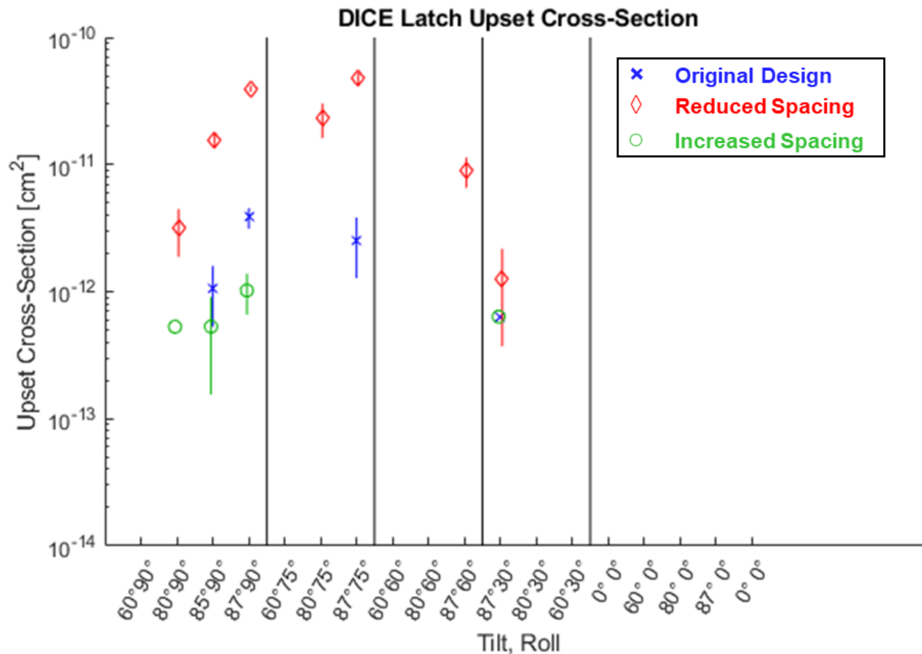


Figure 33: Experimental data which represents that SEU cross-section of the three variants of the DICE latch. The DICE latches were exposed to unidirectional 24.8 MeV/u nitrogen ions at a variety of tilt and roll angles. Measurements which have the same roll are grouped by vertical lines. If no data points are shown, then no upsets were detected at the tilt and roll angle. Fluence was not normalized to tilt angle. Error bars represent one standard deviation.

not normalized to the cosine of the tilt angle. As can be seen from the scatter plots, an angular dependence similar to that found in the simulation work is present.

V.E Similar Techniques used in Radiation Effects Literature

There were two other works of note which were very similar to the simulation technique presented in this chapter. In both of these other works, Vanderbilt University affiliated researchers used MRED in combination with the single-event enabled transistor models from [27] to evaluate the vulnerability of a circuit. In [20], a transient simulation was performed "in-the-loop" for each particle that was simulated, rather than using a 2D critical charge map like in this work. In [21], a more recent work, a 2D critical charge map was used in conjunction with MRED, which resulted in a technique which is very similar to the one developed in this chapter.

V.F Implications and Conclusions

In this chapter, it was shown how simulation can be used to estimate the threat of spatial multiple-node upset. It was revealed that the DICE latch, a circuit which depends on spatial redundancy, can be defeated by unidirectional neutrons if they arrive at an high tilt and roll angle. This possibility was reinforced by experimental ion test results. This has implications for radiation-hardness assurance testing; if high tilt and roll angles are not thoroughly tested, a fielded device could fail much sooner than expected. To avoid the cost of premature device failure and high testing fluence, it is important to leverage simulation techniques like the one used in this chapter.

CHAPTER VI

CONCLUSIONS

In this dissertation, new and familiar mathematical models and simulation techniques were presented to assist the radiation effects community in the design and evaluation of radiation-hardened memory circuits. Temporal MBU, also known as SEU pileup was investigated in chapter IV. The simplified model TMBU model helps circuit designers quickly select a base scrubbing frequency to achieve a desired reliability, while the PATI model allows the designer to evaluate cases where the radiation pulse is close to the scrubbing interval. Spatial MBU and Spatial MNU were investigated in Chapter V. This was done using Vanderbilt's MRED, MRICE, and single-event enabled compact models to construct circuit structures and evaluate their response vulnerability to simultaneous SEU on multiple circuit nodes.

The equations needed to marry these two different facets of reliability analysis were given in Chapter III. The Spatial-Temporal Reliability Model provides a means of simultaneously evaluating XMBU and TMBU and combines the process into a single equation. This equation allows radiation effects engineers to determine the necessary amount of temporal and spatial mitigation for any memory system that uses SEC-DED codes. This work will be useful to the radiation effects community because it will improve its ability to evaluate and communicate the vulnerabilities of memory circuits, both before and after fabrication. As was shown in Chapters III, IV, and V, it is possible focus on both causes of MBU and evaluate their respective impacts on reliability.

APPENDIX A

MONTE-CARLO SIMULATOR USED IN CHAPTER III

Abridged code on the next page shows how a single random trial was performed in the Monte-Carlo simulation from Chapter III. The code on the next page is written in the C# language and begins in the "Perform" method. The code shown in this appendix omits performance optimizations that allow a computer to run several of these methods in parallel and aggregate the statistics.

In the "Perform" method, the inter-arrival time of single-event upsets on a memory word is sampled from an exponential distribution because the arrival of SEUs is a Poisson process. Every time a particle arrives, a random bit is upset. There is also a random chance that the particle will upset a second, adjacent bit. After each particle arrives, the number of bits that are currently upset are tallied and that tally is stored in a list of numbers called an "Upset Sequence." These sequences represent a portion of time where multiple-bit upsets can occur. If periodic scrubbing is being employed, then multiple sequences are used, and the result of the entire trial is the logical union of all sequences. That is, if a single sequence reported a temporal multi-bit upset, then the whole trial reports that a temporal multi-bit upset occurred.

Beginning of Code

```
1           public Result Perform(double lambda_s, double pulse_duration,
2                                 double chi2, double scrub_interval, int number_of_bits)
3           {
4               // Sequences hold the transition state data for the bits
5               List<UpsetSequence> sequences = new List<UpsetSequence>() { new UpsetSequence() };
6               // bits are instantiated as false. false = ok. true = upset
7               bool[] bits = new bool[number_of_bits];
8               // A time of zero seconds corresponds to the radiation pulse beginning
9               double time = 0;
10              // First scrub time is random with respect to the pulse beginning
11              double nextScrub = UniformRandom(min: 0, max: scrub_interval);
12              // Don't use scrubbing if scrub_interval is equal to zero
13              bool scrubbingEnabled = scrub_interval > 0;
14
15              while (time < pulse_duration)
16              {
17                  time += ExponentialRandom(rate_parameter: lambda_s);
18
19                  // An error scrubbing takes place here
20                  if (scrubbingEnabled && time > nextScrub)
21                  {
22                      // Schedule the next scrub event
23                      while (time > nextScrub) nextScrub += scrub_interval;
24                      // Reset the upsets. Assume that this is done perfectly and instantly
25                      for (int i = 0; i < number_of_bits; i++) bits[i] = false;
26                      // Add a new sequence. The previous sequence remains unmodified now.
27                      sequences.Add(new UpsetSequence());
28                  }
29
30                  // Now perform a particle hit if the pulse isn't over
31                  if (time < pulse_duration)
32                  {
33                      // pick a random bit from the word
34                      int bitTarget = RandomInteger(maximumExclusive: number_of_bits);
35                      // Toggle the bit. Bit upset permanence assumption is not used here.
36                      bits[bitTarget] ^= true;
37
38                      // Text for a spatial MBU of order 2
39                      if (UniformRandom(0, 1) < chi2)
40                      {
41                          // Pick the two bits that are adjacent to this one. Use modular arithmetic
42                          int nextTargetA = ((bitTarget + 1) + number_of_bits) % number_of_bits;
43                          int nextTargetB = ((bitTarget - 1) + number_of_bits) % number_of_bits;
44                          int nextTarget = RandomBoolean() ? nextTargetA : nextTargetB;
45
46                          // Toggle the secondary struck bit
47                          bits[nextTarget] ^= true;
48                      }
49
50                      // Add up all of the bits that are upset
51                      int numberOfUpsetsRightNow = bits.Sum(b => b ? 1 : 0);
52
53                      // Add it to the current sequence
54                      sequences.Last().AddDataPoint(numberOfUpsetsRightNow);
55                  }
56              } // End of while-loop
57
58              // Finally, check each sequence
59              // If any sequence reported the events S, T2, X2, or M2 occurring,
60              // then the whole trial reports this event has occurring
61              return new Result()
62              {
63                  S = sequences.Any(seq => seq.S),
64                  T2 = sequences.Any(seq => seq.T2),
65                  X2 = sequences.Any(seq => seq.X2),
66                  M2 = sequences.Any(seq => seq.M2)
67              };
68          }
69
70
71
72
73
74          public struct Result
```

```
75          {
76              public bool S;
77              public bool X2;
78              public bool T2;
79              public bool M2;
80          }
81
82          class UpsetSequence
83          {
84              // This number goes up by 1 every time the
85              // number of upsets is higher than it was before
86              private int NumberOfIncrements = 0;
87              // This is the highest delta that was recorded
88              // in this sequence
89              private int HighestDelta = 0;
90              // After calling "AddDataPoint", this is the most
91              // recent data point
92              private int LastValue = 0;
93
94              // Event Definitions
95              public bool S => NumberOfIncrements > 0;
96              public bool T2 => NumberOfIncrements > 1;
97              public bool X2 => HighestDelta >= 2;
98              public bool M2 => LastValue >= 2;
99
100             // Adds a datapoint to the upset sequence
101             public void AddDataPoint(int datum)
102             {
103                 int delta = datum - this.LastValue;
104                 if (delta > 0)
105                     this.NumberOfIncrements++;
106                 this.HighestDelta = Math.Max(delta, this.HighestDelta);
107                 this.LastValue = datum;
108             }
109         }
```
End of Code

REFERENCES

[1] P. E. Dodd, M. R. Shaneyfelt, J. R. Schwank and J. A. Felix, "Current and future challenges in radiation effects on CMOS electronics," *IEEE Transactions on Nuclear Science,* vol. 57, no. 4, p. 1747–1763, 2010.

[2] T. Calin, M. Nicolaidis and R. Velazco, "Upset Hardened Memory Design for Submicron CMOS Technology," *IEEE Transactions on Nuclear Science,* vol. 43, no. 6, pp. 2874-2878, 1996.

[3] J. A. Maharrey, J. S. Kauppila, R. C. Harrington, P. Nsengiyumva, D. R. Ball, T. D. Haeffner, E. X. Zhang, B. L. Bhuva, W. T. Holman and L. W. Massengill, "Dual-Interlocked Logic for Single-Event Transient Mitigation," *IEEE Transactions on Nuclear Science,* vol. 65, no. 8, pp. 1872-1878, 2018.

[4] JEDEC Solid State Technology Association, Measurement and Reporting of Alpha Particle and Terrestrial Cosmic Ray-Induced Soft Errors in Semiconductor Devices (JESD89B), 2021.

[5] J. Suh, M. Annavaram and M. Dubois, "MACAU: A Markov model for reliability evaluations of caches under Single-bit and Multi-bit Upsets," in *IEEE International Symposium on High-Performance Comp Architecture*, New Orleans, LA, 2012.

[6] S. S. Mukherjee, J. Emer, T. Fossum and R. S. K, "Cache scrubbing in microprocessors: myth or necessity?," in *10th IEEE Pacific Rim International Symposium on Dependable Computing, 2004. Proceedings*, Papeete, France, 2004.

[7] J. R. Schwank, M. R. Shaneyfelt, D. M. Fleetwood, J. A. Felix, P. E. Dodd, P. Paillet and V. Ferlet-Cavrois, "Radiation Effects in MOS Oxides," *IEEE Transactions on Nuclear Science,* vol. 55, no. 4, pp. 1833-1853, 2008.

[8] D. M. Fleetwood, "Evolution of Total Ionizing Dose Effects in MOS Devices With Moore's Law Scaling," *IEEE Transactions on Nuclear Science,* vol. 65, no. 8, pp. 1465-1481, 2018.

[9] W. A. Kolasinski, J. B. Blake, J. K. Anthony, W. E. Price and E. C. Smith, "Simulation of cosmic-ray induced soft errors and latchup in integrated-circuit computer memories," *IEEE Transactions on Nuclear Science,* vol. 26, no. 6, p. 5087–5091, 1979.

[10] P. Wang, A. L. Sternberg, B. D. Sierawski, E. X. Zhang, K. M. Warren, A. M. Tonigan, R. M. Brewer, N. A. Dodds, G. Vizkelethy, S. L. Jordan and D. M. Fleetwood, "Sensitive-Volume Model of Single-Event Latchup for a 180-nm SRAM Test Structure," *IEEE Transactions on Nuclear Science,* vol. 67, no. 9, pp. 2015-2020, 2020.

[11] F. W. Sexton, "Destructive Single-Event Effects in Semiconductor Devices and ICs," *IEEE Transactions on Nuclear Science,* vol. 50, no. 3, pp. 603-621, 2003.

[12] R. C. Baumann, "Radiation-Induced Soft Errors in Advanced Semiconductor Technologies," *IEEE Transactions on Device and Materials Reliability,* vol. 5, no. 3, pp. 305-316, 2005.

[13] E. Normand, "Single-Event Effects in Avionics," *IEEE Transactions on Nuclear Science,* vol. 43, no. 2, pp. 461-474, 1996.

[14] R. Silberberg, C. H. Tsao and J. R. Letaw, "Neutron Generated Single-Event Upsets in the Atmosphere," *IEEE Transactions on Nuclear Science,* vol. 31, no. 6, pp. 1183-1185, 1984.

[15] M. A. Clemens, B. D. Sierawski, K. M. Warren, M. H. Mendenhall, N. A. Dodds, R. A. Weller, R. A. Reed, P. E. Dodd, M. R. Shaneyfelt, J. R. Schwank, S. A. Wender and R. C. Baumann, "The Effects of Neutron Energy and High-Z Materials on Single Event Upsets and Multiple Cell Upsets," *IEEE Transactions on Nuclear Science,* vol. 58, no. 6, pp. 2591-2598, 2011.

[16] R. A. Reed, R. A. Weller and e. al., "Anthology of the Development of Radiation Transport Tools as Applied to Single-Event Effects," *IEEE Transactions on Nuclear Science,* vol. 60, no. 3, pp. 1876-1911, 2013.

[17] J. Allison, et al., "Geant4 Developments and Applications," *IEEE Transactions on Nuclear Science,* vol. 53, no. 1, pp. 270-278, 2006.

[18] R. A. Reed, R. A. Weller, M. H. Mendenhall, D. M. Fleetwood, K. M. Warren, B. D. Sierawski, M. P. King, R. D. Schrimpf and a. E. C. Auden, "Physical processes and

applications of the Monte Carlo radiative energy deposition (MRED) code," *IEEE Transactions on Nuclear Science,* vol. 62, no. 4, pp. 1441-1461, 2015.

[19] R. A. Weller, M. H. Mendenhall, R. A. Reed, R. D. Schrimpf, K. M. Warren, B. D. Sierawski and L. W. Massengill, "Monte Carlo Simulation of Single Event Effects," *IEEE Transactions on Nuclear Science,* vol. 57, no. 4, pp. 1726-1746, 2010.

[20] K. Warren, J. Kauppila, R. Weller, M. Mendenhall, R. Reed, R. Schrimpf and L. Massengill, "Analyzing Single Event Effects with Monte Carlo Radiation Transport And Electronic Design Automation Tools," *Journal of Radiation Effects, Research and Engineering,* vol. 36, no. 1, pp. 2-21, 2018.

[21] J. D. Black, J. A. Dame, D. A. Black, P. E. Dodd, M. R. Shaneyfelt, J. Teifel, J. G. Salas, R. Steinbach, M. Davis, R. A. Reed, R. A. Weller, J. M. Trippe, K. M. Warren, A. M. Tonigan, R. D. Schrimpf and R. S. Marquez, "Using MRED to Screen Multiple-Node Charge-Collection Mitigated SOI Layouts," *IEEE Transactions on Nuclear Science,* vol. 66, no. 1, pp. 233-239, 2019.

[22] P. E. Dodd and F. W. Sexton, "Critical charge concepts for CMOS SRAMs," *IEEE Transactions on Nuclear Science,* vol. 42, no. 6, pp. 1764-1771, 1995.

[23] J. G. Rollins and J. Choma, "Mixed-mode PISCES-SPICE coupled circuit and device solver," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,* vol. 7, no. 8, pp. 862-867, 1988.

[24] A. Fedoseyev, R. Arslanbekov and M. Turowski, "Mixed-mode simulations of ICs with complex nuclear events from MRED/Geant4 with 3D TCAD," in *2009 MIXDES-16th International Conference Mixed Design of Integrated Circuits & Systems*, Lodz, Poland, 2009.

[25] P. Roche, G. Gasiot, J. L. Autran, R. A. Reed and R. A. Weller, "Application of the TIARA Radiation Transport Tool to Single Event Effects Simulation," *IEEE Transactions on Nuclear Science,* vol. 61, no. 3, pp. 1498-1500, 2014.

[26] T. Thery, G. Gasiot, V. Malherbe, J.-L. Autran and P. Roche, "TIARA: Industrial Platform for Monte Carlo Single-Event Simulations in Planar Bulk, FD-SOI, and FinFET," *IEEE Transactions on Nuclear Science,* vol. 68, no. 5, pp. 603-610, 2021.

[27] J. S. Kauppila, A. L. Sternberg, M. L. Alles, A. M. Francis, J. Holmes and O. A. M. L. W. Amusan, "A Bias-Dependent Single-Event Compact Model Implemented Into BSIM4 and a 90 nm CMOS Process Design Kit," *IEEE Transactions on Nuclear Science,* vol. 56, no. 6, pp. 3152-3157, 2009.

[28] J. S. Kauppila, T. D. Haeffner, D. R. Ball, K. A. V, T. D. Loveless, S. Jagannathan, A. L. Sternberg, B. L. Bhuva and L. W. Massengill, "Circuit-Level Layout-Aware Single-Event Sensitive-Area Analysis of 40-nm Bulk CMOS Flip-Flops Using Compact Modeling," *IEEE Transactions on Nuclear Science,* vol. 58, no. 6, pp. 2680-2686, 2011.

[29] J. S. Kauppila, L. W. Massengill, D. R. Ball, M. L. Alles, R. D. Schrimpf, T. D. Loveless, J. A. Maharrey, R. C. Quinn and J. D. Rowe, "Geometry-Aware Single-

Event Enabled Compact Models for Sub-50 nm Partiall Depleted Silicon-on-Insulator Technologies," *IEEE Transactions on Nuclear Science,* vol. 62, no. 4, pp. 1589-1598, 2015.

[30] J. S. Kauppila, D. R. Ball, J. A. Maharrey, R. C. Harrington, T. D. Haeffner, A. L. A. M. L. Sternberg and L. W. Massengill, "A Bias-Dependent Single-Event-Enabled Compact Model for Bulk FinFET Technologies," *IEEE Transactions on Nuclear Science,* vol. 66, no. 3, pp. 635-642, 2019.

[31] G. C. Messenger, "Collection of Charge on Junction Nodes from Ion Tracks," *IEEE Transactions on Nuclear Science,* vol. 29, no. 6, pp. 2024-2031, 1982.

[32] R. W. Hamming, "Error detecting and error correcting codes," *The Bell System Technical Journal,* vol. 29, no. 2, pp. 147-160, 1950.

[33] M. Saleh, J. Serrano and J. Patel, "Reliability of Scrubbing Recovery-Techniques for Memory Systems," *IEEE Transactions on Reliability,* vol. 39, no. 1, pp. 114-122, 1990.

[34] J. A. Abraham, E. S. Davidson and J. H. Patel, "Memory System Design for Tolerating Single Event Upsets," *IEEE Transactions on Nuclear Science,* vol. 30, no. 6, pp. 4339-4344, 1983.

[35] A. Sari and M. Psarakis, "Scrubbing-Aware Placement for Reliable FPGA Systems," *IEEE Transactions on Emerging Topics in Computing,* vol. 8, no. 3, pp. 564-576, 2020.

[36] X. Li, H. Lou and Z. Jin, "A Fault-tolerant Method of SRAM FPGA Based on Processor Scrubbing," in *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, Chongqing, China, 2021.

[37] G. He, S. Zheng and N. Jing, "A Hierarchical Scrubbing Technique for SEU Mitigation on SRAM-Based FPGAs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems,* vol. 28, no. 10, pp. 2134-2145, 2020.

[38] S. Zheng, H. You, G. He, Q. Wang, T. Si, J. Jiang, J. Jin and N. Jing, "A Rapid Scrubbing Technique for SEU Mitigation on SRAM-Based FPGAs," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, Sapporo, Japan, 2019.

[39] M. Mousavi, H. R. Pourshaghaghi, H. Corporaal and A. Kumar, "Scatter Scrubbing: A Method to Reduce SEU Repair Time in FPGA Configuration Memory," in *2019 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, Noordwijk, Netherlands, 2019.

[40] P. Reviriego, J. A. Maestro and C. Cervantes, "Reliability Analysis of Memories Suffering Multiple Bit Upsets," *IEEE Transactions on Device and Materials Reliability,* vol. 7, no. 4, pp. 595-601, 2007.

[41] A. Dutta and N. A. Touba, "Multiple Bit Upset Tolerant Memory Using a Selective Cycle Avoidance Based SEC-DED-DAEC Code," in *25th IEEE VLSI Test Symposium (VTS'07)*, Berkeley, CA, 2007.

[42] D. S. Vibbert, M. L. Breeding, J. S. Kauppila, K. M. Warren, R. A. Reed, B. D. Sierawski, W. T. Holman and L. W. Massengill, "Layout Dependence of Neutron Single-Event Vulnerabilities in Sub-50nm Hardened Latches," *Journal of Radiation Effects Research and Engineering,* vol. 39, no. 1, pp. 327-334, 2021.

[43] "ACCRE," Vanderbilt University, [Online]. Available: https://www.vanderbilt.edu/accre/.

[44] M. H. Mendenhall and R. A. Weller, "A Probability-Conserving Cross-section Biasing Mechanism for Variance Reduction in Monte-Carlo Particle Transport Calculations," *Nuclear Instruments and Methods in Physics Research A,* no. 667, pp. 38-43, 2012.