Modeling and Mitigation of Soft Errors in Nanoscale SRAMs

by

Shah M. Jahinuzzaman

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2008

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Shah M. Jahinuzzaman

Abstract

Energetic particle (alpha particle, cosmic neutron, etc.) induced single event data upsets or soft errors have emerged as key reliability concerns in SRAMs in sub-100 nanometre technologies. Low operating voltage, small node capacitance, high packing density, and lack of error masking mechanisms are primarily responsible for the soft error susceptibility of SRAMs. In addition, since SRAM occupies the majority of the die area in system-on-chips (SoCs) and microprocessors, different leakage reduction techniques, such as supply voltage reduction, gated grounding, etc., are applied to SRAMs in order to limit the overall chip leakage. These leakage reduction techniques exponentially increase the soft error rate in SRAMs. The soft error rate is further accentuated by process variations, which are prominent in scaled-down technologies. In this research, we address these concerns and propose techniques to characterize and mitigate soft errors in nanoscale SRAMs.

We develop a comprehensive analytical model of the critical charge, which is a key to assessing the soft error susceptibility of SRAMs. The model is based on the dynamic behaviour of the cell and a simple decoupling technique for the non-linearly coupled storage nodes. The model describes the critical charge in terms of NMOS and PMOS transistor parameters, cell supply voltage, and noise current parameters. Consequently, it enables characterizing the spread of critical charge due to process-induced variations in these parameters and to manufacturing defects, such as, resistive contacts or vias. In addition, the model can estimate the improvement in critical charge when MIM capacitors are added to the cell in order to improve the soft error robustness. The model is validated by SPICE simulations (90nm CMOS) and radiation tests. The critical charge calculated by the model is in good agreement with SPICE simulations with a maximum discrepancy of less than 5%. The soft error rate estimated by the model for low voltage (sub 0.8 V) operations is within 10% of the soft error rate measured in the radiation test. Therefore, the model can serve as a reliable alternative to time-consuming SPICE simulations for optimizing the critical charge and hence the soft error rate at the design stage.

In order to limit the soft error rate further, we propose an area-efficient multiword based error correction code (MECC) scheme. The MECC scheme combines four 32 bit data words to form a composite 128 bit ECC word and uses an optimized 4-input transmission-gate XOR logic. Thus MECC significantly reduces the area overhead for

check-bit storage and the delay penalty for error correction. In addition, MECC interleaves two composite words in a row to limit cosmic neutron induced multi-bit errors. The ground potentials of the composite words are controlled to minimize the leakage power without compromising the read data stability. However, use of composite words involves a unique write operation where one data word is written while other three data words in the same composite word are read to update the check-bits. A power efficient word line signaling technique is developed to facilitate the write operation. A 64 kb SRAM macro with MECC has been designed and fabricated in a commercial 90nm CMOS technology. Measurement results show that the SRAM consumes 534 μW at 100 MHz with a data latency of 3.3 ns for a single bit error correction. This translates into 82% per-bit energy saving and 8x speed improvement over recently reported multiword ECC schemes. Accelerated neutron radiation testing carried out at TRIUMF in Vancouver confirms that the proposed MECC scheme can correct up to 85% of soft errors.

Acknowledgements

I would like to express my profound gratitude to my supervisor Professor Manoj Sachdev for his insightful guidance and generous support throughout this research. I feel highly privileged to have been able to work under the supervision of a person like him whose research solves real world problems faced by the semiconductor industry. He provided me with an excellent research environment with the full freedom to develop my work. At the same time, he closely supervised my progress through regular meetings and led me in the right direction. He gave me the much needed moral support and encouragement during tough times. He made my PhD research an enjoyable learning experience indeed.

I would like to thank Professor Bruce Cockburn, Professor Ajoy Opal, Professor Andrei Sazonov, and Professor James Martin for serving on my Ph.D. Committee. Their insightful questions and comments significantly improved the quality of this thesis.

I am grateful to Dr. Ewart Blackmore of TRIUMF for his help in irradiating the test chips. I thank Phil and Fernando for keeping the lab computers up and running all times, particularly before tape-out deadlines. Special thanks to Wendy, Lisa, and Annette of the ECE Office for being so friendly and supportive, even with last minute requests.

It has been a great pleasure for me to be a part of the CMOS Design and Reliability (CDR) Group. My sincere appreciation goes to all former and present members of this group. In particular, I am grateful to Andrei, Nitin, and Mohammad Sharifkhani for their immense support at the beginning of this work. I am thankful to Hossein, Shahab, David Rennie, Tahseen, Jaspal, Sumanjit, David Li, and Pierce for their unforgettable help in laying out the test chips and designing the PCBs. Special thanks to Mohammad, Jaspal, Tahseen, and Tasreen for many cheerful moments that we shared in the lab.

I am indebted to my wife, Afrin, and our daughter, Simra, for making my life so enjoyable. In particular, Afrin's endless love, support, and patience have been invaluable for my work. She has always put higher priority to my work over her own doctoral research, enabling me to finish my work on time.

Last but surely not least, I am grateful to my parents, my sister, and my brother for their endless care and support throughout the long path of my academic endeavour.

Dedication

To my beloved parents.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter provides the basics of the soft error problem in nanoscale integrated circuits, particularly in SRAMs. The chapter describes the sources and mechanism of soft errors, identifies the key reasons for increased soft error rate in SRAMs, and discusses the resulting reliability implications, thereby outlining the motivation behind this research.

Following Moore's law, semiconductor technology scaling has enabled the remarkable advancement of integrated circuits (ICs) over the past four decades. Every technology generation, which spanned from two to three years, has doubled the transistor count per chip, increased the operating frequency by 43%, and reduced the switching energy consumption by 65% [1]. Today's ICs consist of transistors that have a gate length ($\sim$ 32 nanometre) much smaller than the size of an influenza virus ($\sim$ 100 nanometre) while being less expensive and more powerful than ever before. However, scaling in the sub-100 nanometre regime has brought in a number of quality and reliability issues that have been less of a concern so far. In particular, due to smaller device dimensions and lower operating voltages, nanoscale ICs have become highly sensitive to operational disturbances. These disturbances include board-level noise, signal coupling, supply and substrate noise, and transients caused by ionizing radiation. In a well-designed IC, however, radiation-induced transients appear to be the most troublesome. In addition, nanoscale ICs suffer from increased process-induced variations in device parameters (e.g., threshold voltage, channel length and width, etc.) and exhibit large off-state or leakage current.

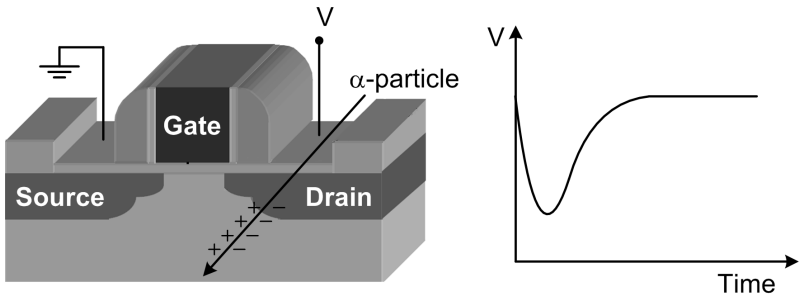


Figure 1.1: Simplistic view of alpha particle strike on a transistor and the resulting transient on the drain voltage.

Radiation-induced transients in ICs are primarily caused by energetic neutrons and alpha particles, which come from cosmic rays and chip packaging materials, respectively [2]. These particles generate a dense track of electron-hole pairs as they pass through a semiconductor device, causing a voltage transient at the node that collects the charge (see Figure 1.1) [3]. This phenomenon is referred to as a single event transient (SET). Due to their high charge collection efficiency, the reverse-biased p-n junctions in an IC are the most susceptible parts to SETs. If a sufficient amount of charge is collected by the junction, the SET results in a fault by flipping the logic state ('1' to '0' or vice versa) at the associated node. When such a fault is latched into a memory cell or a flip-flop, a single event upset (SEU) occurs. Since an SEU does not permanently damage the device, it is referred to as a "soft error".

Although a soft error does not damage the device, it poses a potential threat to the reliable operation of a circuit. If uncorrected, soft errors cause a failure rate higher than the rate of all hard failure mechanisms (gate oxide breakdown, metal electromigration, latch-up, etc.) combined. Typically, hard failure rates add up to 50∼200 FIT (Failure In Time: 1 fail per 10^9 hours of device operation). Conversely, the soft error rate (SER) can easily exceed 50,000 FIT per chip [4]. The SER can be even higher due to process variations and circuit techniques employed for leakage reduction. Thus, characterization and mitigation of soft error in nanoscale ICs become critical.

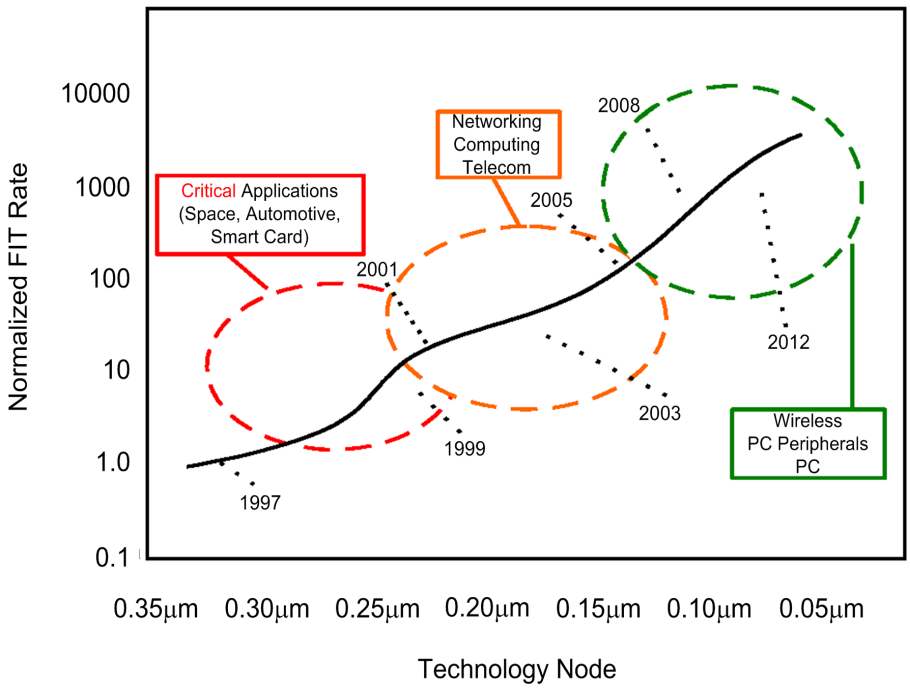


Figure 1.2: Soft error rate forecast for different digital systems. Source: iRoC Technologies and Semico Research Inc.

1.1 Soft Error Overview

The soft error phenomenon at ground level was first reported by May and Woods in dynamic random access memory (DRAM) in 1978 [5]. Cosmic ray induced malfunctions of space-borne electronics had been known even before [6]. However, the impact of soft errors was not severe at ground level due to larger node capacitance and higher noise margin that stemmed from larger geometries and higher operating voltages, respectively. With technology scaling, the operating voltage and node capacitance have both decreased approximately by 30% in every generation [7]. This has resulted in a quadratic decrease in the signal charge that represents a logic state [8]. Consequently, the minimum amount of particle-induced charge that is necessary to upset the logic has decreased. The amount of collected charge has also decreased due to shrinking of the device volume. However, the former effect dominates over the latter, resulting in an increased SER with technology scaling [9], [10].

Figure 1.2 shows the SER in a variety of digital systems as a function of technology node. Here, SER data were collected by AMD, Intel, and Compaq. As evident from

Figure 1.2, most of the digital systems in sub-100nm technologies are highly susceptible to soft errors. Accordingly, state-of-the art microprocessors are being implemented using soft error robust circuits [11], [12]. In fact, soft errors have always been a key reliability concern for mission-critical applications where a single error can lead to catastrophic failures. Examples of such applications include space-borne electronics, aircraft controllers, military electronics, microprocessors in network servers, implantable medical equipment (e.g., cardiac defibrillators), etc.

1.1.1 Soft Error Sources

Three particle sources have been identified as the major causes of soft errors in electronic systems: i) alpha particles, ii) high-energy neutrons, and iii) the interaction of thermal neutrons with boron (particularly ^{10}B) that is present in boro-phosphosilicate glass (BPSG) dielectric [2]. The third particle source is no longer a concern as the BPSG has been eliminated from the fabrication process in the 0.25μm technology onwards.

A. Alpha Particles

An alpha particle is a doubly-ionized helium atom ($^4He^2$) that is emitted when the nucleus of an unstable isotope decays to a lower energy state. Primarily, alpha particles come from residual radioactive elements in a chip's packaging material. Among such elements, Uranium (^{238}U), Thorium (^{232}Th), and Lead-210 (^{210}Pb) are the dominant sources of alpha particles for integrated circuits [13], [14]. They can be found in trace amounts in the package materials like mold compound and underfill, and predominantly in solder balls (made usually of PbSn). Thus, flip-chip packages, which use solder balls (see Figure 1.3) for the power supply and I/Os, are particularly vulnerable to soft errors. As a rule-of-thumb, 1 ppM of ^{238}U in package materials can result in an alpha flux of 1 α/cm^2-h while flux levels of the order of 0.01 α/cm^2-h can be sufficient to cause high soft error rates [13].

Alpha particles can have energies ranging from 1 to 9 MeV and can penetrate silicon to a depth of approximately 23.6 μm. The interaction of alpha particle with silicon is almost purely electronic, i. e., its energy is lost to directly generate electron hole pairs (EHPs). Typically, 3.6 eV generates 1 EHP in silicon [13]. Thus, 1 MeV of energy can

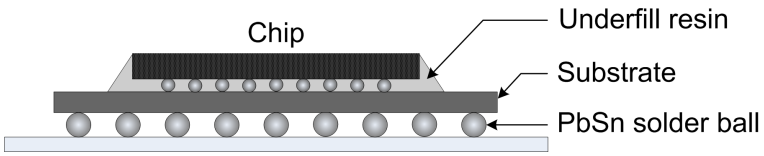


Figure 1.3: Cross-section of a flip-chip package showing the alpha emitting solder balls.

generate approximately 44.5 fC of charge, which is sufficient to flip the state of a logic circuit.

In order to reduce the alpha particle induced soft errors, three techniques are commonly employed [2]. The first technique is to use extremely pure materials that have low alpha emission rate. This technique significantly increases the packaging cost. In addition, the minimum achievable alpha emission in this technique is in the range of 0.001 α/cm^2-h, which may still not be acceptable for many high reliability applications. The second technique is to develop design rules so that sensitive circuits are kept physically separated from alpha emitting packaging components. However, this technique only works if the package has well-defined alpha emitting zones, like solder balls, and if the chip has few sensitive circuit elements. For SoCs that have memory occupying more than 50% of the chip area, the technique does not appear to be a viable solution. The last technique is to shield the high alpha emitting materials from the circuit components by employing thin polyimide (e.g., epoxy) coatings over the finished chip prior to bonding and encapsulation. While this technique is useful for lead-frame and ceramic packages, it cannot be used in flip-chip designs where the solder balls need to be electrically connected to the top metal layer of the chip. Thus, the soft error threat for flip-chip packages persists.

B. Cosmic Neutrons

The second significant source of soft errors are high-energy neutrons coming from cosmic rays, which are of galactic origin. Cosmic rays react with the Earth's atmosphere and produce complex cascades of secondary particles [15]. As the particles move deeper into the atmosphere, they generate tertiary particles (see Figure 1.4). Finally at terrestrial altitudes or sea level, the primary flux of cosmic rays is greatly reduced and only 1% or less of the primary flux remains. The predominant particles at this altitude include muons,

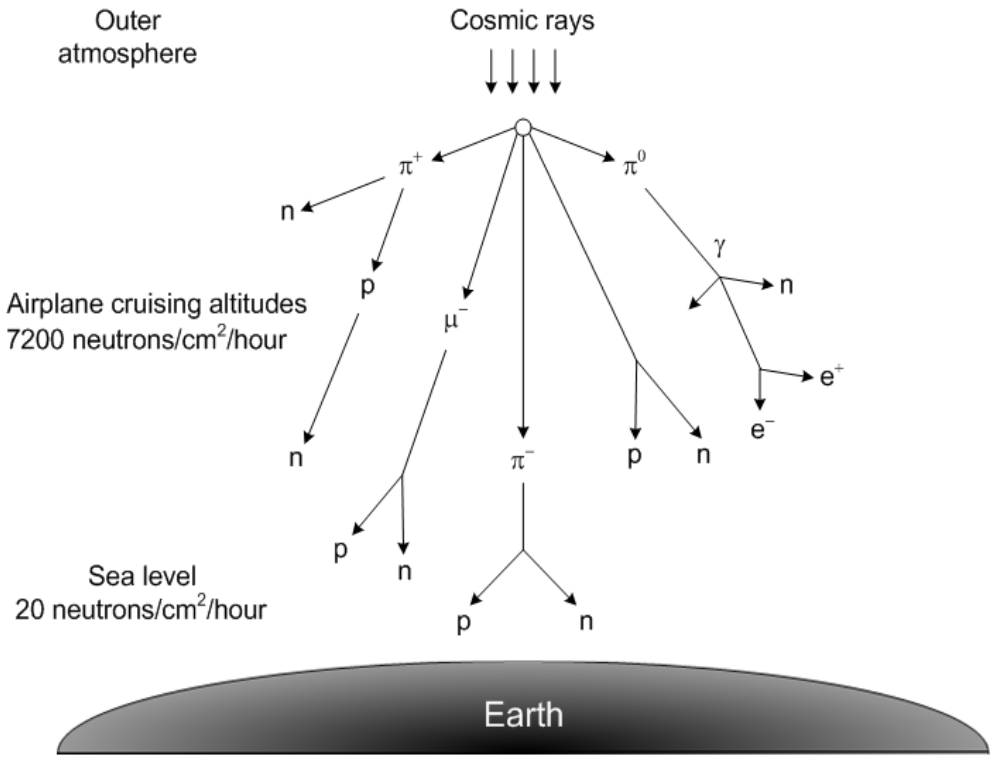


Figure 1.4: Interaction of cosmic rays with atmosphere and the resultant cascade of particles.

protons, electrons, neutrons, and pions. Muons and pions are short-lived while protons and electrons are attenuated by Coulombic interactions with the atmosphere. However, neutrons survive due to their charge neutrality and relatively high flux density. Thus, neutrons constitute the most likely cosmic radiation that causes soft errors in electronic devices at terrestrial altitudes.

The density of cosmic neutron flux is a function of neutron energy. The flux density decreases with increasing neutron energy. In addition, the flux density depends on the altitude. For example, at 10,000 feet above the sea level, the cosmic ray flux increases by 10x [15]. Thus, cosmic ray intensities vary in different cities of the world as shown in Figure 1.5. As a result, cosmic neutron-induced SER for the same device will be different in different cities.

In contrast to alpha particles, cosmic neutrons themselves do not directly generate ionization in silicon. The primary mechanism by which neutrons cause soft error is the neutron-induced silicon recoil (both elastic and non-elastic). In this mechanism, a high

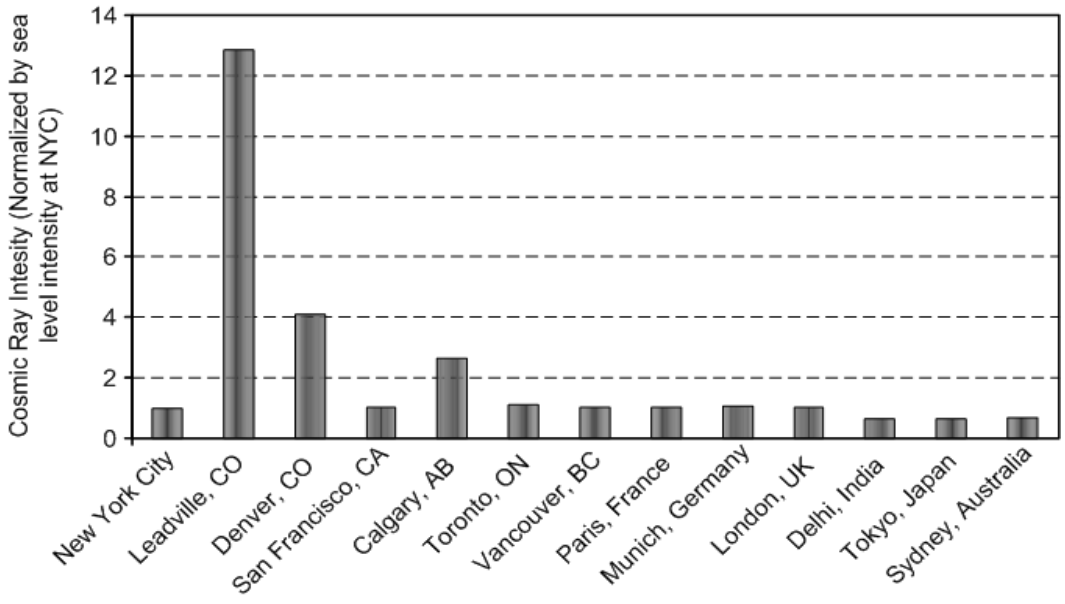


Figure 1.5: Cosmic ray intensity at different cities in the world.

energy neutron collides with a silicon nucleus and transfers most of its kinetic energy to knock the silicon from the lattice. Typically the silicon nucleus breaks into smaller fragments, each of which generates charge. The charge density per distance traveled for silicon recoils (25-150 fC/μm) is significantly higher than that for alpha particles (16 fC/μm) [4]. However, silicon recoils have smaller penetration depth in silicon ($\sim$few μm) because they lose their energy more rapidly than the alpha particles owing to being lighter. Thus, the current transient produced by neutrons has higher magnitude but shorter duration.

Reducing the cosmic neutron flux at the chip level is very difficult. Concrete has been shown to shield the cosmic radiation at a rate of approximately 1.4x per foot of concrete thickness [2]. Thus, the SER due to cosmic neutrons of a system operating in a basement surrounded by many feet of concrete could be significantly reduced. While this may be a viable option for mainframe computers, little can be done for personal desktop applications or portable electronics to reduce the neutron soft errors. Therefore, reduction of cosmic ray induced soft error requires mitigation techniques within the chip, such as improving the robustness of the circuit or using error correction techniques.

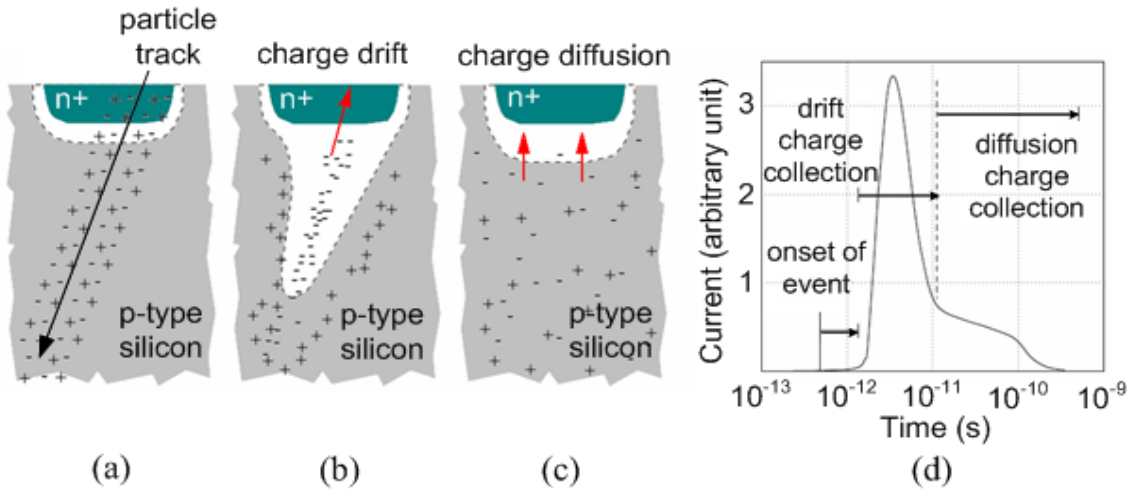


Figure 1.6: a-c) Charge deposition and collection events at a reverse-biased p-n junction after a particle strike, and d) the resulting current at the collection node. Adapted from [4].

1.1.2 Soft Error Mechanisms

The underlying mechanism of a soft error event can be divided into three main phases: (a) onset of the event, (b) drift charge collection, and (c) diffusion charge collection [4]. As shown in Figure 1.6, in phase (a), a cylindrical track of EHPs with a submicron radius and a high carrier concentration is generated in the wake of the energetic particle's passage. The amount of generated charge depends on the particle's linear energy transfer (LET), which indicates the energy loss per unit path length. Typically, LET is expressed in MeV-cm^2/mg by normalizing the energy loss per unit length (in MeV/cm) by the density of the target material (in mg/cm^3) so that LET becomes independent of the target [3]. The LET of a particle can easily be related to its charge deposition per unit path length. In silicon, an LET of 97 MeV-cm^2/mg corresponds to a charge deposition of 1 pC/μm.

When the particle-induced charge track traverses or comes close to a reverse-biased *p*-*n* junction, EHPs are rapidly collected due to the high electric field of the depletion region of the junction. Here, electrons drift to the higher potential of the *n*-diffusion and holes drift to the lower potential of the *p*-substrate. This phase of charge collection is referred to as phase (b). A notable feature of this phase is the distortion of the depletion region into a *funnel shape*. The funnel greatly enhances the efficiency of the

drift collection by extending the equipotential lines of the depletion region deeper into the substrate and thus increasing the charge collection volume. The size of the funnel is a function of substrate doping - the funnel distortion increases for decreasing substrate doping. Phase (b) is completed within few tens of picoseconds and is followed by phase (c) where diffusion begins to dominate the collection process. Charge collection through diffusion continues for longer time (from hundreds of picoseconds to nanoseconds) until all excess carriers have been collected, recombined, or diffused away from the junction area.

The current pulse resulting from above phases is shown in Figure 1.6(d). In general, the farther away from the junction that the particle strikes, the smaller the amount of charge that will be collected, and thus the less likely it is that the event will cause a soft error. In integrated circuits, a node is never isolated but is in close proximity with other nodes. Thus, charge sharing among nodes and parasitic bipolar action (resulting from the formation of unintentional bipolar transistors between junctions and wells) can greatly influence the amount of charge collected. In fact, the magnitude of collected charge (Q_{coll}) depends on a complex combination of factors including the size of the device, biasing of the various circuit nodes, substrate structure, substrate doping, the type of the particle, its energy, its trajectory, the initial position of the event within the device, and the state of the device. Q_{coll} does not result in a soft error until it exceeds a critical charge (Q_{crit}), which is defined as the minimum charge required to cause a change in the data state [3]. Thus, in the event of a particle strike, a soft error will result if $Q_{coll} > Q_{crit}$. Otherwise, the circuit will survive the event. Therefore, the critical charge can be used as a figure of merit to assess the soft error susceptibility. However, the critical charge is not constant since the response of the device to the charge injection is dynamic and dependent on the magnitude as well as the temporal characteristics of the pulse [4],[8]. Consequently, the critical charge becomes a function of the node capacitance, operating voltage, and the strength of the restoring mechanisms connected to the node, making it difficult to model.

1.2 Soft Errors in Integrated Circuits

Both the logic circuit and memory of an integrated circuit are susceptible to soft errors. Here, by the term 'memory' we refer to the main memory and the cache memory, which

are commonly realized by dynamic random access memory (DRAM) and static random access memory (SRAM), respectively. Due to the differences in the design and operation of logic circuits and memories, their susceptibility to soft errors is different.

1.2.1 Soft Errors in Logic Circuits

A. Sequential Logic

The effect of particle-induced transients on the typical sequential elements, such as, a latch, a register file cell or a domino cell, is similar to that in an SRAM as the stored bit may change in the case of a soft error event. However, compared to SRAM, the sequential logic is usually less susceptible to soft errors due to the use of larger transistors (hence larger capacitance and driving strength) in latches and associated logic gates.

B. Combinational Logic

A particle-induced transient or SET may propagate through the combinational stages and eventually be latched by a sequential element. However, many transients will not result in a soft error due to three masking effects: logical masking, electrical masking, and latching window masking, which are inherently present in combinational circuits [8].

i). Logical Masking

The logical masking effect can be described with the help of the NAND gate in Figure 1.7(a). If a particle strikes at an input of the NAND gate, but one of the other inputs is in the controlling state (e.g., 0), the strike will be completely masked and the output will not change. Thus, the particle will not be able to cause a soft error. In fact, for an error to propagate in combinational circuits, there must be a sensitized path from the affected node to either the primary output or the input of a flip-flop.

ii). Electrical Masking

Since any CMOS circuit has a limited bandwidth, transients with bandwidths higher than the cutoff frequency will be attenuated. Thus, the amplitude of the particle-induced transient may reduce, the rise and fall times may increase, and, eventually, the pulse

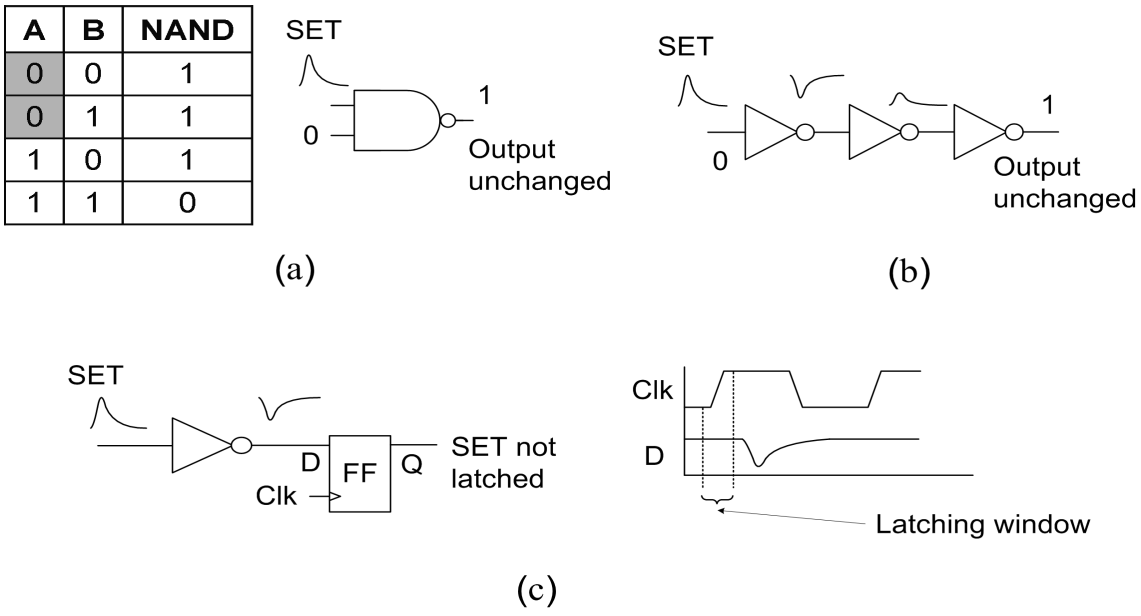


Figure 1.7: Different error masking mechanisms in logic circuits: a) logical masking, b) electrical masking, and c) latching-window masking.

may disappear as it passes through logic gates (see Figure 1.7(b)). This phenomenon is referred to as electrical masking.

iii). Latching Window Masking

As the transient propagates towards a sequential element, such as, at node D in the flip-flop shown in Figure 1.7(c), the transient may occur outside the clock window. Thus, the transient may fail to be latched into the flip-flop, resulting in no soft error. This is called latching window masking or temporal masking. With the increase of operating frequency of logic circuits, the effectiveness of latching window masking decreases, thus increasing the probability of soft error.

1.2.2 Soft Errors in Memories

Compared to logic circuits, memories are more vulnerable to soft error due to their high packing density and the relative lack of transient masking mechanisms. A particle strike directly affects a memory cell and often the neighbouring cells by changing the stored values in these cells. The changed values remain stored until the cells are rewritten.

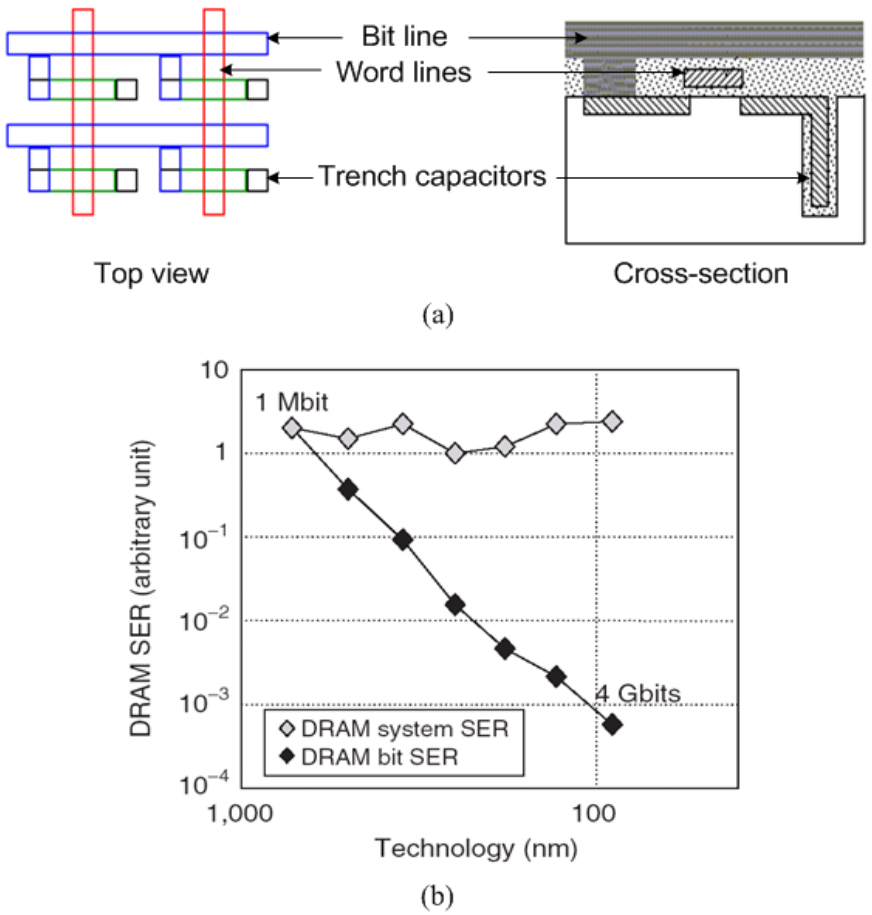


Figure 1.8: a) Typical DRAM layout and cross-section and b) DRAM soft error rate with technology scaling.

A. DRAM

The soft error rates per megabit in DRAM was initially high when signal charge used to be stored on a planar 2D capacitor in each cell. Such cells had large area junctions that were very efficient at collecting particle induced charge. However, with the development of 3D (e.g., trench, stack, etc.) capacitors, not only the packing density increased but also the SER significantly decreased. The latter is due to the reduction in the sensitive junction volume without appreciably decreasing the node capacitance.

Figure 1.8 illustrates the typical DRAM structure and the SER trends as a function of technology scaling. Although voltage reduction with technology scaling reduces the critical charge, the concurrent aggressive junction volume scaling results in more significant reduction in the collected charge. The net result is that the DRAM SER of a single

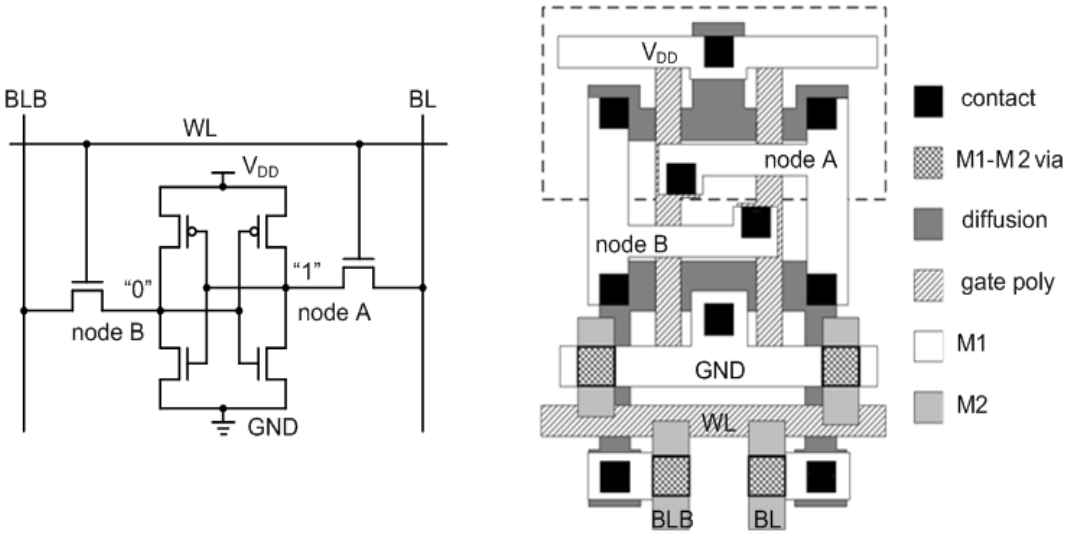


Figure 1.9: A conventional six-transistor SRAM cell schematic and layout. WL: word line, BL: bit line, BLB: complementary bit line.

bit decreases approximately by a factor of 4 to 5 per generation. However, the DRAM system SER remains roughly constant. This is due to the fact that system requirements have increased the memory density (bits per system) almost as fast as the SER reduction that technology scaling provided.

B. SRAM

In contrast to a DRAM cell, an SRAM cell stores one datum and its complement on an active circuit comprising of two cross-coupled inverters (see Figure 1.9). Since the inverters continuously drive each other, the cell can retain the data as long as the power supply is ON - without any need for refresh. This is why the cell is referred to as a *static* RAM cell. However, since the cell stores both '0' and '1', it has two sensitive nodes (nodes A and B in Figure 1.9) that are susceptible to soft errors. In particular, the sensitive regions are the reverse-biased drain junctions of the driver and load transistors, which are OFF.

Early SRAMs were more robust against the soft errors because of their higher operating voltages and larger junction capacitances. With technology scaling, designers have deliberately minimized the SRAM junction area to reduce capacitance, leakage, and cell area while aggressively reducing the operating voltage to minimize power consumption.

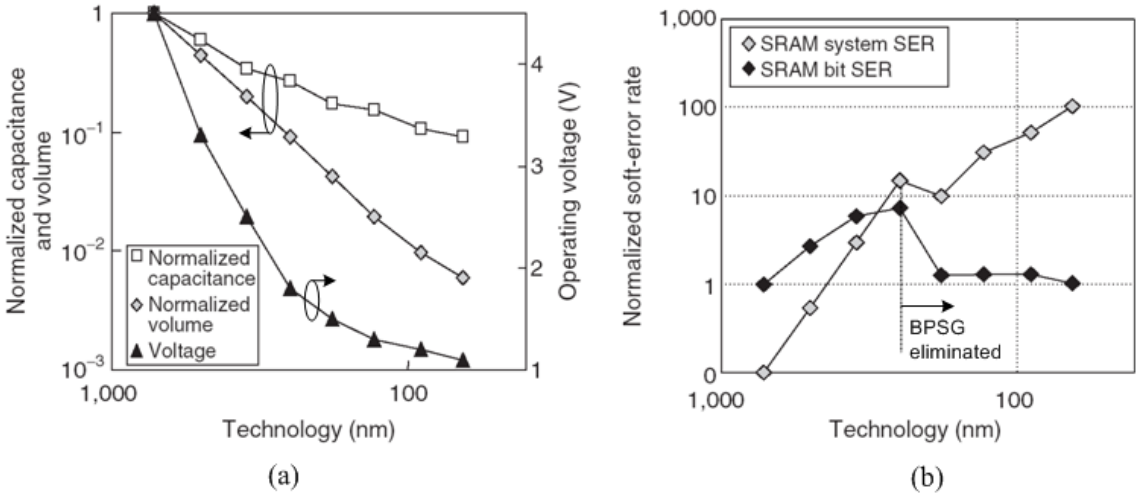


Figure 1.10: a) Capacitance and voltage scaling in SRAM, b) bit-level and system-level soft error rate in SRAM. Adapted from [16].

Figure 1.10(a) shows these scaling trends and Figure 1.10(b) shows the resulting SERs.

The reduction in operating voltage and node capacitance in each successive SRAM generation has cancelled out the reduction in cell collection efficiency caused by shrinking of the cell depletion volume. However, as shown in Figure 1.10(b), the SRAM single bit SER was initially increasing with each successive generation. This happened particularly in products using boro-phosphosilicate glass (BPSG) dielectric. As the BPSG has been eliminated from the process (0.25μm and beyond) and the feature sizes have shrunk into the deep-submicron (DSM) range, the SRAM bit SER has become almost saturated [16]. This saturation can be attributed to the saturation in voltage scaling, reductions in junction collection efficiency, and increased charge sharing with neighboring nodes. However, saturation in the SRAM bit SER does not translate in saturation in the SRAM system SER since scaling also implies an increase in memory density. Accordingly, the SRAM system SER increases in every generation, the increase being exponential as evident in Figure 1.10(b). Thus, among logic circuits and memories, SRAM SER has become the most critical concern, necessitating an in-depth investigation. In fact, due to the ease of integration with logic circuits, the absence of a refresh operation like DRAM, and its high operating speed, SRAM is primarily used to realize embedded memory, which occupies the majority of the die area in today's SoC (see Figure 1.11(a)). Die area dedicated to embedded memory keeps increasing in order to meet consumers' insatiable demand of

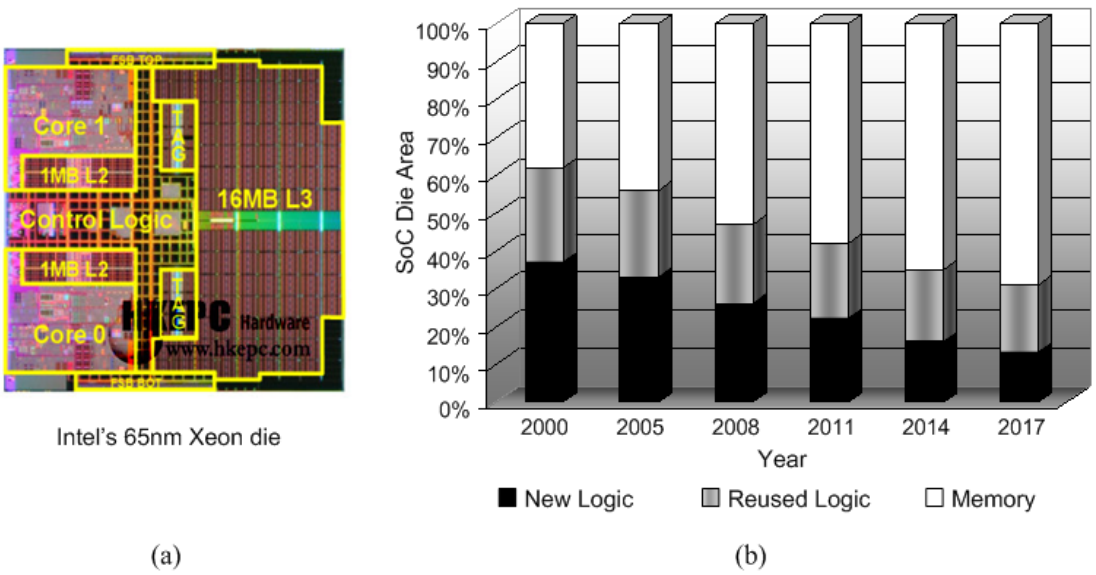


Figure 1.11: a) Intel's Xeon processor with large cache memory and b) typical trend of memory and logic area on an SoC die (Semico Research Corp.).

performance, as shown in Figure 1.11(b). As a result, SRAM SER becomes the limiting factor for the overall soft error performance of the SoC. In addition, being the largest building block, SRAM dominates the yield and leakage power of the SoC, which suffers from increased process variations and higher leakage power consumption in nanoscale technologies.

1.3 SRAM Soft Errors and Process Variations

Process-induced variations in device and interconnect parameters occur when they deviate from their ideal, i.e., as-designed values due to process limitations, such as mask imperfections, lithographic limitations, dopant fluctuations, etc. Process variations have always been an important aspect that influenced manufacturability in IC fabrication processes [17]. However, in nanoscale processes, where feature sizes are extremely small, variations become a larger fraction of designed values, thereby significantly affecting circuit performance and yield [18], [19].

Process variations can exist between runs, wafers, dies on the same wafer or die-to-die (D2D), and even within one die (WID). WID variations are more of a concern as they cause mismatches between two similar devices in a die, resulting in delay and timing

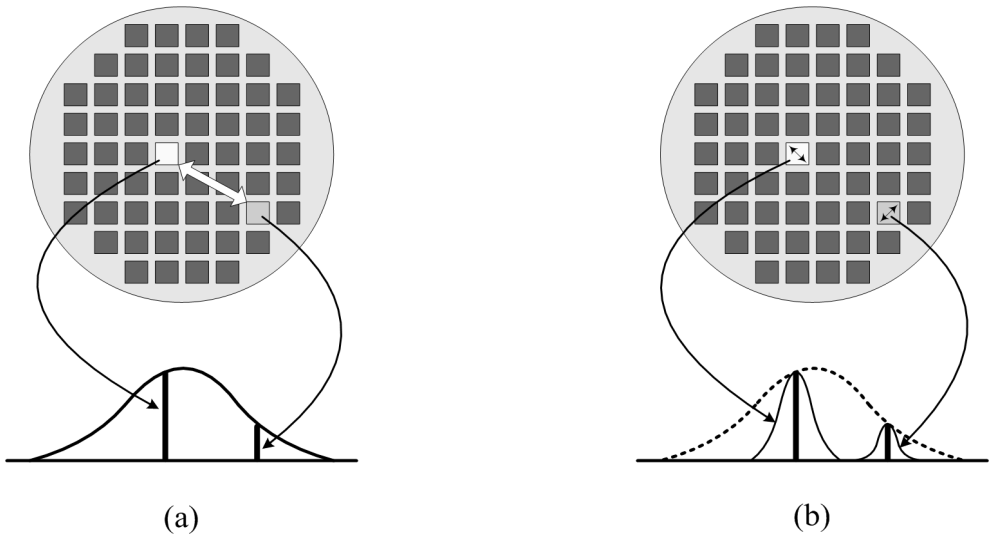


Figure 1.12: a) Die-to-die (D2D) variation across a wafer and b) within die (WID) variations in two dies that are already subject to D2D variations.

variabilities. WID variations are aggravated by D2D variations since the latter skew a given WID variation across the wafer as shown in Figure 1.12. The major WID variations include:

- interconnect sheet resistance variation due to variations in wire width;
- transistor threshold voltage (V_{TH}) variation due to variations in oxide thickness, dopant implant level in the channel region, gate line edge and line width roughness (LER and LWR as shown in Figure 1.13), surface and oxide trapped charge, etc. [20], [21];
- transistor channel width (W) and wire width variations due to variation in field oxide step; and
- transistor channel length (L) variation due to LER or variations in source/drain diffusion and poly silicon width by photolithography proximity effects and plasma etch dependencies.

Some of above variations are systematic while some other are random. Systematic variations, such as interconnect width variation, are predictable. They depend on deterministic factors like layout structure and the surrounding topological environment and

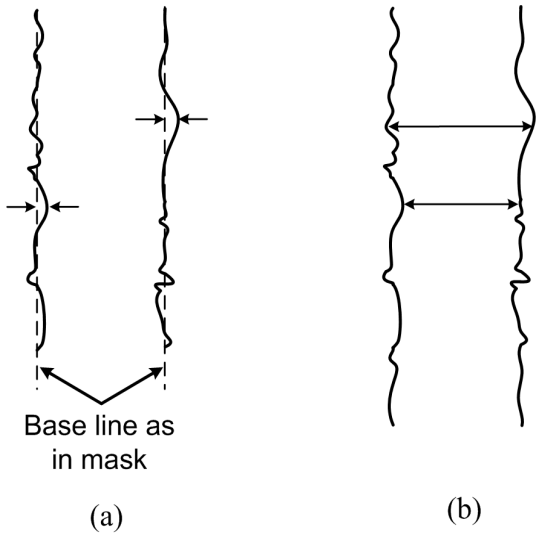


Figure 1.13: Simplistic view of a) line edge roughness (LER) and b) line width roughness (LWR).

show a predictable trend across the chip. On the other hand, random variations, such as channel dopant implant, LER, etc. are unpredictable [22]. They are caused by random uncertainties in the fabrication process, such as microscopic fluctuations in the number and location of dopant atoms. Random variations are the most troublesome as the systematic variations can be minimized by layout techniques. Among the random variations, variations in V_{TH}, L and W are the most critical as they directly affect the current drive capability of transistors. These variations cause more pronounced effects in smaller devices. For example, V_{TH} variation is inversely proportional to the gate area. Since SRAM uses the smallest possible transistors in order to meet tight density requirements, variations in V_{TH}, L and W significantly affect SRAM's stability and performance [20]. In particular, they cause variations in Q_{crit} across an SRAM population, which can potentially lead to poorer soft error performance. Therefore, while characterizing soft error performance of SRAM, process variations needs to be considered.

1.4 SRAM Soft Errors and Leakage

Another key concern from using the smallest geometry transistors in SRAM is increased leakage current. Transistors in sub-100nm technologies exhibit higher sub-threshold and

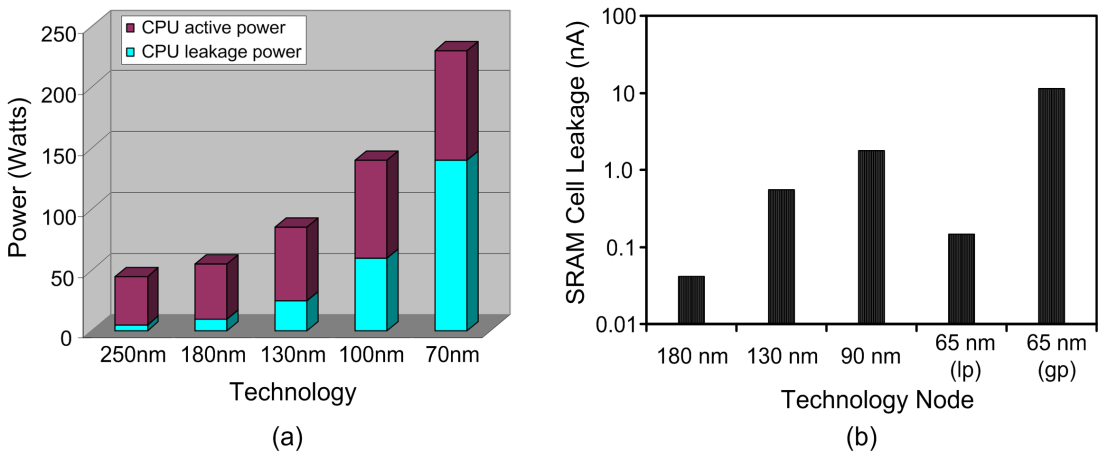


Figure 1.14: a) Increasing leakage power fraction in total power consumption of microprocessors with technology scaling and b) Increasing SRAM cell leakage current with technology scaling (simulated).

gate leakage current due to reduction in channel length and gate dielectric thickness, respectively. As a result, leakage power consumptions in microprocessors, SoCs and SRAM become dominant with technology scaling as shown in Figure 1.14 [23]. In fact, being the largest block and consisting of the maximum number of transistors, SRAM dominates the leakage power consumption of the microprocessors and SoC, playing a key role in sustaining battery life of portable devices.

In a six-transistor (6T) SRAM cell, storage nodes formed by the cross-coupled inverter pair are accessed from the bit lines (BL and BLB) using two NMOS transistors (see Figure 1.15). These access transistors are turned ON by the wordline (WL) whenever the cell is accessed for a read or write operation. Accordingly, when the cell is in standby mode i.e., the cell is not accessed (WL=0V), there are three OFF transistors and two ON transistors, which exhibit subthreshold leakage and gate-to-channel leakage, respectively. Although an OFF transistor (Q2 or Q6 in Figure 1.15) can exhibit gate-to-source and gate-to-drain leakage, the leakage is negligible compared to the gate-to-channel leakage of an ON transistor. Figure 1.15 shows subthreshold and gate leakage current paths, which are the dominant leakage mechanisms in the cell.

To reduce the leakage currents in the SRAM cell, a number of techniques can be employed. For example, the cell supply voltage (V_{DD}) can be lowered so that both the

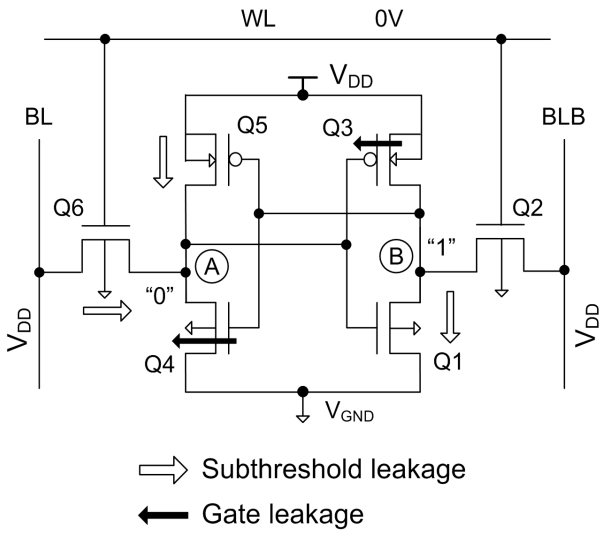


Figure 1.15: Dominant leakage current paths in an un-accessed SRAM cell.

subthreshold and gate leakage are reduced due to reduced a drain-induced barrier lowering (DIBL) effect and a reduced gate-channel electric field, respectively. In another approach, the logic-0 voltage or virtual ground (V_{GND}) voltage can be raised to apply a reverse body bias (RBB) to the leaky driver and access transistors (Q1 and Q6 in Figure 1.15) and thereby reduce DIBL effect (in Q1 in Figure 1.15). In either of these approaches, the rail-to-rail voltage in the cell decreases, which reduces Q_{crit} (since Q=CV). As a result, these low-power SRAMs exhibits higher SER, which is not acceptable in many applications, such as in microprocessors of network servers. Figure 1.16 shows the reduction in Q_{crit} with increasing V_{GND} while Figure 1.17 shows the increase in the SER of commercial SRAMs with decreasing supply voltage [24]. In fact, the increase in the SER is exponential in the sub-1 V supply voltage regime. This underscores the need of soft error mitigation in nanoscale SRAMs, which operate at or below 1V.

1.5 Motivation and Thesis Outline

As mentioned earlier, the reliability and yield of SRAM are crucial for the overall reliability and yield of the SoC. With technology scaling in the sub-100nm regime, SRAM reliability is affected by a number of factors, such as soft errors, leakage power, process variations, etc. In particular, due to smaller critical charge and increasing packing

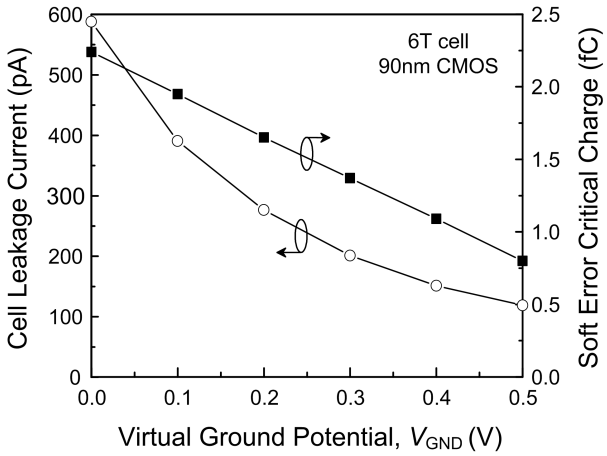


Figure 1.16: Leakage current and critical charge as function of virtual ground potential in a gated-grounded low-power SRAM cell.

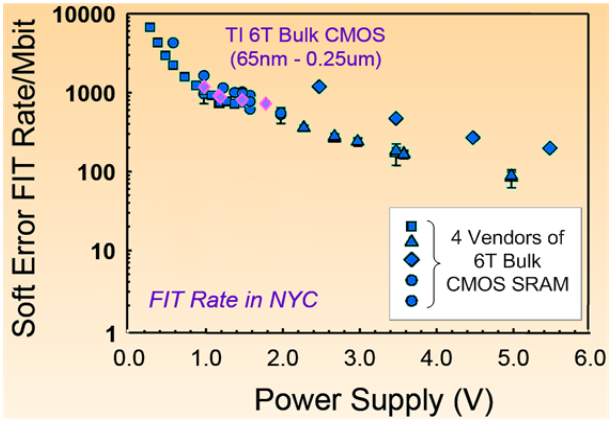


Figure 1.17: Soft error rate of commercial SRAMs as a function of supply voltage.

density, the SRAM soft error rate increases exponentially with technology scaling, thus significantly affecting the data integrity. The soft error rate is accentuated by the increased process variations in nanoscale technologies. In fact, due to the use of minimum geometry transistors, SRAM is more prone to process variations. Furthermore, different low-power techniques that target SRAM leakage reduction significantly increase the SRAM soft error rate. This thesis addresses these issues related to soft errors in SRAMs. In particular, this thesis proposes a comprehensive model of soft error critical charge and devises an area and energy-efficient soft error mitigation technique for low-power SRAMs. The critical charge model will enable designers to estimate and optimize the critical charge and hence the SER at the design stage. Since the process variations can

lead to SER variations across the fabricated chip, the process dependence of the critical charge is also studied. Quantitative information of such dependence can help the process engineer to fine-tune the process in order to reduce the SER in fabricated chips. On the other hand, the proposed soft error mitigation scheme, which is based on a multiword error correction code (ECC), provides a cost-effective solution to limit the SER. In fact, we show how the scheme can be combined with a low-power virtual ground technique to simultaneously reduce the SER and leakage current in SRAMs.

The thesis is organized as follows. Chapter 2 provides an overview of SRAM architecture and operation, and investigates the soft error susceptibility of low-power SRAMs. Chapter 3 reviews the existing soft error modeling and mitigation approaches. Chapter 4 proposes a comprehensive critical charge model for SRAMs. Chapter 5 characterizes the process dependence of the critical charge using the proposed model. Chapter 6 describes the proposed soft error mitigation technique. Chapter 7 summarizes the contributions of this research and draws important conclusions.

Chapter 2

SRAM Architecture and Operation

This chapter discusses the architecture and operation of a typical SRAM and investigates several commonly used low-power SRAMs and their soft error susceptibility.

In order to study the soft error phenomenon in SRAMs, a clear understanding of the architecture and operation of an SRAM is necessary. Accordingly, in this chapter we discuss a typical SRAM architecture, design issues, read/write operations, and soft error susceptibility, particularly of low-power SRAMs.

2.1 SRAM in the Memory Hierarchy

Memory hierarchy refers to the hierarchical arrangement of storage units in a modern computer system. The pyramid-like hierarchy of memory ranges from the faster, smaller capacity but more costly on-chip volatile memories to slower, larger capacity but cheaper non-volatile remote storage [25], [26]. In particular, memory hierarchy consists of following six levels (L0-L5) of memory as shown in Figure 2.1:

- L0: Registers - fastest and smallest memory sitting at the top of the memory hierarchy and closest to the central processing unit (CPU). ; typical size is few

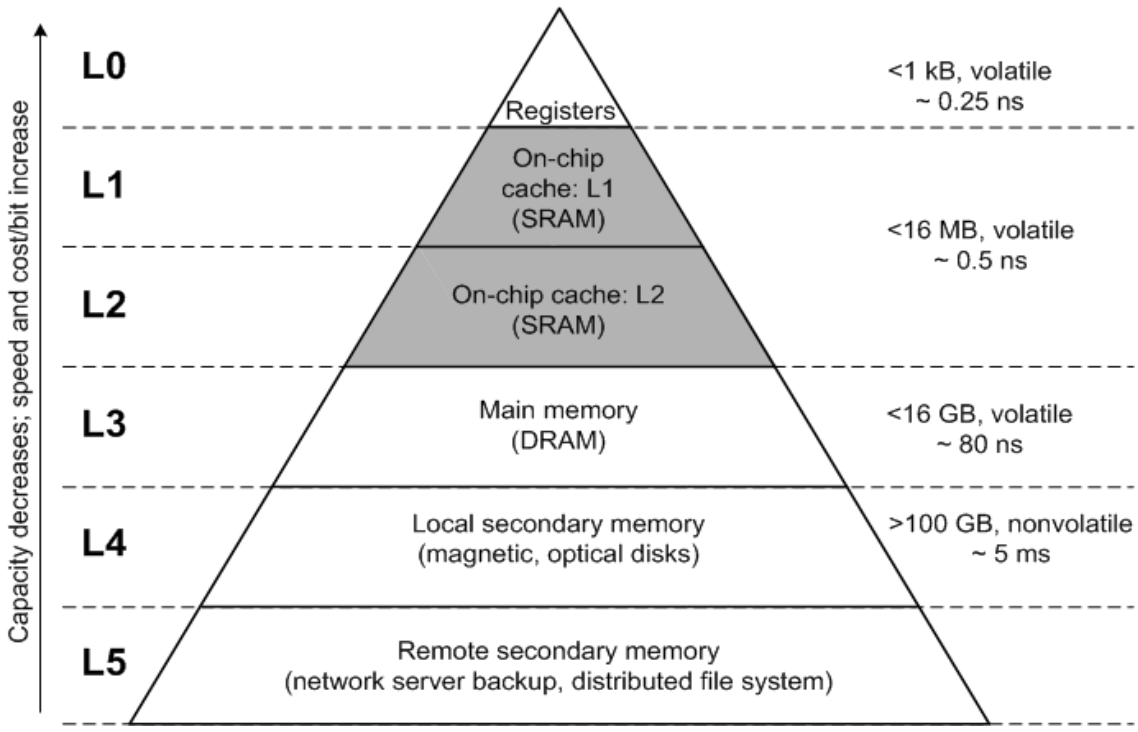


Figure 2.1: Memory hierarchy with typical size and access time in a modern computer system.

hundred bytes; cost per bit is the highest ; typical access time is one CPU cycle ($\sim$ 0.25 ns - 1ns).

- L1: Level 1 Cache - highest performance on-chip memory after the registers; cost per bit is much lower than that of registers since cache has far more storage capacity than is available in all the registers combined; typically consists of SRAMs; size is few tens of kilobytes (kB); access time is one or two CPU cycles.

- L2: Level 2 cache - lower performance on-chip memory compared to L1 cache, however, is still much faster than off-chip memories; typical size is hundreds of kilobytes to several megabytes (MB); cost per bit is lower than L1 cache because of larger capacity; access time is 2 to 10 times that of L1 cache.

- L3: Main Memory - fastest off-chip memory but slower than L2 cache; typically, consists of DRAM or some similar inexpensive memory technology; cost per bit is significantly lower compared to the cache memory systems; size is multiple gigabytes (GB); access time can be hundreds of CPU cycles.

- L4: Local Secondary or Disk Storage - significantly slower than main memory but very large; size can be hundreds of GB; cost per bit is far less ($\sim$ three orders of magnitude) than the main memory; typically realized by magnetic or optical disks, access time is millions of CPU cycles, i. e., few milliseconds.

- L5: Remote Secondary or Network Storage - very slow but huge storage capacity that is distributed in a network; often used as a backup storage; typical access time is few seconds.

The cache memory plays a key role in enhancing the performance of CPUs. The cycle time of the CPU is much less than the access time of the DRAM. This gap is increasing with further improvement in CPUs. Accordingly, multiple levels of on-chip embedded caching in modern processors have been introduced so that the CPU can quickly fetch data from the cache and process them faster. Such levels are represented by L1 and L2 cache memories. Deeper level of cache (L3) has also been introduced in high-end server microprocessors [27]. One of the ways to realize these embedded memories is to use the high-density DRAM. Embedded one transistor (1T) DRAM implemented in the standard logic process can benefit SoCs from its fast low-V_{TH} transistors coupled with high packing density. However, the high subthreshold leakage current restricts employing the 1T embedded DRAM cell. Replacing 1T DRAM cells with alternative DRAM cell designs having more transistors results in an area penalty, which undermines the cell area advantage that embedded DRAMs normally offer over embedded SRAMs. If a typical DRAM process is employed instead of the logic process to fabricate the 1T embedded DRAM, the cell will have a high packing density as well as high-V_{TH} for low leakage. However, such low leakage DRAM is slower, limiting the performance of an SoC [28]. On the other hand, embedded SRAMs is much faster. They use the regular fast (low-V_{TH}) logic process and do not require additional mask steps. In addition, SRAMs do not need periodic refresh and hence can be more power-efficient in read operations. Thus, SRAM has evolved as the dominant embedded memory in present SoCs and microprocessors.

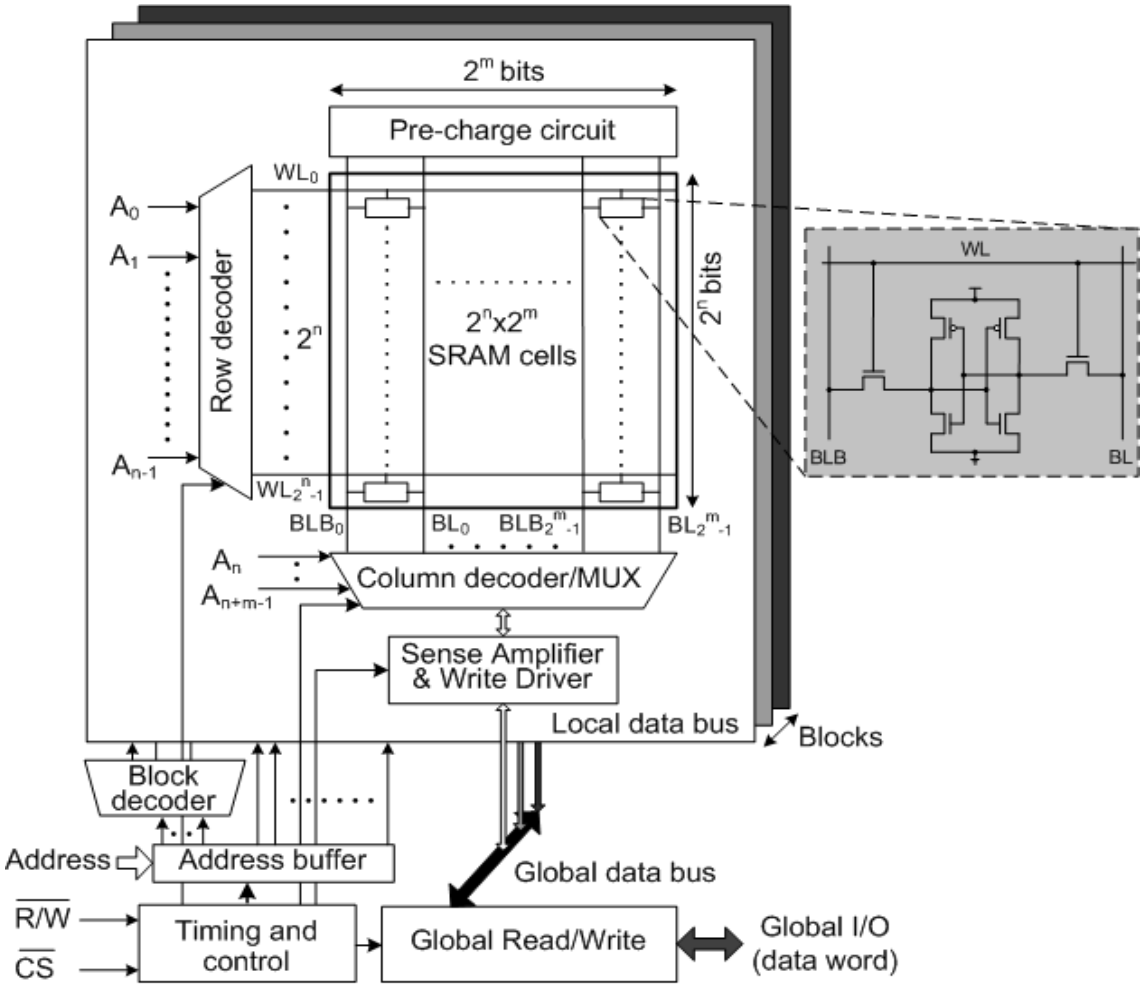


Figure 2.2: A typical SRAM architecture.

2.2 SRAM Architecture

An SRAM consists of an array of memory cells along with peripheral circuits, which enable reading from and writing into the array. Figure 2.2 shows the basic architecture of an SRAM. The memory array consists of 2^n rows and 2^m columns of cells. Since a row is accessed by activating a word line (WL), there are 2^n word lines. Address bits A_0 to A_{n-1} are decoded to select one of these word lines. On the other hand, m address bits, A_n to A_{n+m-1}, are decoded to select the column, i.e., the bit line pair (BL and BLB) to access a particular bit in the memory array for read or write. Typically, a group of bit line pairs is selected where the group corresponds to the data word. A data word can be 16, 32, or 64 bits wide. Thus, for a 32 bit word, each row has $2^m/32$ words, each of

which can be selected by the column decoder. The required number of address bits for the column decoder then becomes $\log_2(2^m/32)$. If the SRAM is large, it can consist of several blocks of arrays, as shown in Figure 2.2. In such a case, few address bits are used by a block decoder to select one of the blocks and multiplex the input/outputs (IOs) of the block. In this case, global sense amplifiers and write drivers can be employed. The timing of the activation of sense amplifier, write driver, decoders, etc. are controlled by a timing and control block. Most modern SRAMs are self-timed, i.e. all the internal timing is generated by the timing block within the SRAM instance. The chip select (CS) signal is often provided in multi-chip architectures while the read/write (R/W) signal determines whether the SRAM is to be read or written.

In a read operation, a sense amplifier is used at every column (often with multiple columns using the column MUX) to read the selected word through the bit lines. On the other hand, in a write operation, a write driver drives BL and BLB of a column to '0' or '1' according to the input data and enables writing of the data into the selected word. Thus, in its simplest form, an SRAM consists of following circuits:

- SRAM cell
- row decoder
- column decoder or multiplexer
- pre-charge and equalizer
- sense amplifier
- write driver
- timing and control

The following sub-sections briefly discuss the above mentioned circuits and review pertinent design considerations.

2.2.1 SRAM Cell

The cell is the key component that stores the binary data bit in an SRAM. A typical cell consists of a latch and access transistors. The latch holds the data bit while the access

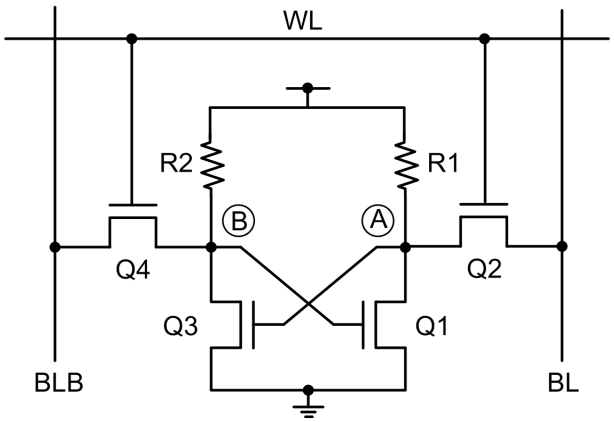


Figure 2.3: 4T SRAM cell with resistor load.

transistors enable read and write access to the cell and provide isolation when the cell is not accessed. In fact, an SRAM cell has to meet the following requirements:

- provide non-destructive read access;
- provide reliable write access;
- infinitely retain the data given the power is supplied to the cell; and
- occupy minimum possible area for high packing density.

In order to meet these requirements, several cell architectures have been proposed. Among these, three cells, namely, the resistive-load four-transistor (4T) cell, the loadless 4T cell, and the six-transistor (6T) CMOS cell, are worth mentioning. Each of these cells has its own design considerations, advantages, and disadvantages. The cell design considerations involve tradeoffs between area, speed, robustness, and power.

A. Resistive Load 4T SRAM Cell

Figure 2.3 shows a 4T SRAM cell with polysilicon resistors (R1 and R2) as loads or pull-up devices and NMOS transistors (Q1 and Q3) as the pull-down devices [1]. The other two NMOS transistors (Q2 and Q4) serve as access transistors to communicate with the storage nodes A and B from the complementary bit lines BL and BLB. The cell is symmetric by design since R1=R2, Q1=Q3, and Q2=Q4. The cell is, in fact, a remnant of the pre-CMOS technologies.

The load resistor compensates for the off-state leakage of pull-down NMOS when a logic '1' is retained at the corresponding node. The load resistor also provides the pull-up current when the cell is written. The value of the load resistor must be as high as possible to reduce the leakage power consumption during retention and maintain a reasonable noise margin (NM) by limiting the logic '0' level degradation during access. However, a high value of the load resistor reduces the pull-up current, increasing the low-to-high transition time. Thus, there exists a trade-off between leakage minimization and speed while choosing the value of the load resistors. In fact, the upper limit of the value of the load resistor is set by the requirement to provide a pull-up current of at least two orders of magnitude larger than the leakage current [1]. On the other hand, the lower limit is set by the noise immunity requirements and power limitations.

The inverters comprising the 4T cell have lower gain in the transition region. As a result, they produce less steep voltage transfer characteristics (VTCs), which imply lower NM and longer recovery time from the metastable state . The stability and soft error performance of the cell are also poor in low-voltage scaled-down technologies. The resistor does not scale very well with technology. Furthermore, the extra processing steps for forming the high-resistivity polysilicon resistor are not a part of the standard logic process. These factors prohibit using the resistive load 4T cells in SoCs, which are traditionally implemented using a standard full CMOS process. Therefore, the resistive load 4T cell will not be considered further in this thesis.

B. Loadless 4T SRAM Cell

The loadless 4T CMOS SRAM cell proposed by Noda *et. al.* is shown in Figure 2.4 [29]. In this cell PMOS transistors Q2 and Q4 serve as access transistors as opposed to NMOS access transistors in a resistive load 4T SRAM cell. If NMOS access transistors are used in the loadless SRAM cell, two major problems arise. First, the logic '1' voltage becomes limited to $V_{DD} - V_{THn}$. Second, the data retention condition requires the V_{TH} of Q2 (Q4) to be smaller than that of Q1 (Q3), which in turn results in larger logic '0' degradation and hence smaller noise margin in a read access.

Data (logic '1') retention in the loadless 4T cell is provided by ensuring that the leakage current of the PMOS transistor (I_{leak-p}) is higher than the leakage current of the

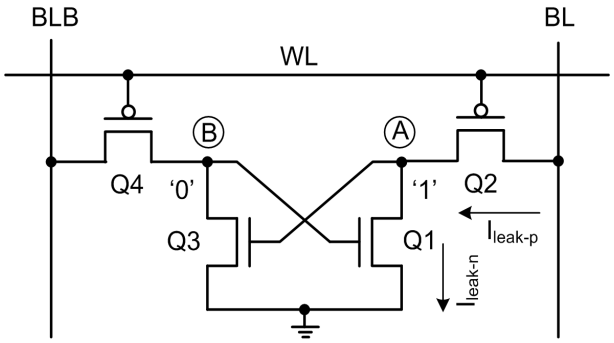


Figure 2.4: 4T loadless SRAM cell.

NMOS transistor (I_{leak-n}). This condition is usually met by using a dual-V_{TH} process with $V_{THp} < V_{THn}$. The non-destructive read operation requires that the NMOS driver transistors (Q1 and Q3) be stronger than the PMOS access transistors (Q2 and Q4). This requirement is easily met even with the minimum size NMOS and PMOS since the NMOS inherently has higher mobility ($\sim 2-3$ times the PMOS mobility). Consequently, the 4T loadless cell becomes highly area-efficient. In fact, for the same design rule, the 4T loadless cell can be 35% smaller than the 6T cell [29].

Since memory blocks typically occupy the majority of SoC die area, the area savings offered by the 4T loadless cell have been one of the main driving forces behind its development. However, the cell is not free from drawbacks. The data retention in the cell can only be guaranteed if I_{leak-p} is significantly larger ($\sim$ 10x - 100x) than I_{leak-n}. In the worst case process voltage temperature (PVT) variations, meeting this condition can be difficult. In addition, the PMOS access transistor cannot pull down the storage node to '0' in a write operation. It makes the write operation slower than that with an NMOS access transistor. Because of these factors, the loadless 4T cell is not used in mainstream high performance SRAMs.

C. 6T CMOS SRAM Cell

The 6T CMOS SRAM cell evolved from the resistive load 4T cell by replacing the resistors with PMOS transistors. Thus, the 6T cell consists of two CMOS cross-coupled inverters that form two complementary storage nodes A and B, as shown in Figure 2.5. Activated by word line (WL), two NMOS access transistors (Q2 and Q6) provide read and write

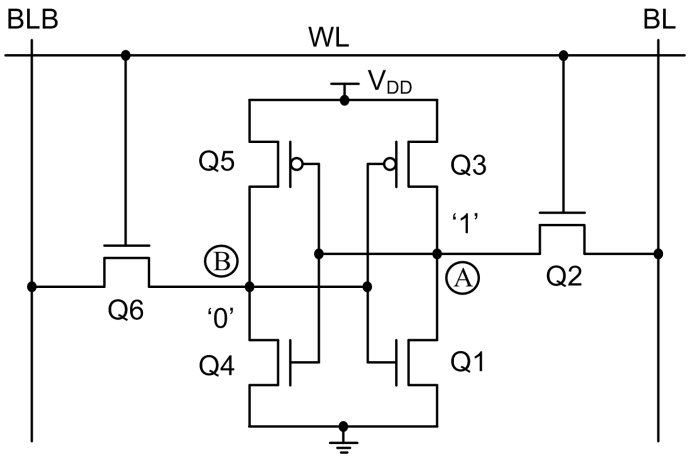


Figure 2.5: 6T CMOS SRAM cell.

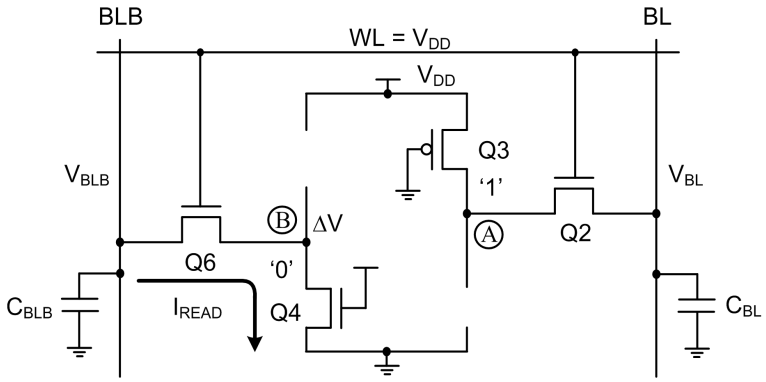


Figure 2.6: Simplified circuit of the 6T CMOS SRAM cell during a read operation.

access to the storage nodes from the bit lines (BL and BLB).

The 6T CMOS cell is the most widely used SRAM cell in today's SoCs and microprocessors. Accordingly, we consider the 6T CMOS cell in this thesis and discuss its operations and design issues in more detail.

Read Operation

The read operation starts by activating WL, which connects the storage nodes to the precharged bit lines. Depending on the value of the storage nodes, one bit line voltage remains at the precharged level while the other bit line voltage starts to drop. In Figure 2.6, the bit line voltage V_{BL} remains at the precharge level equal to V_{DD}. The complementary bit line voltage V_{BLB} is discharged through transistors Q4 and Q6, which are

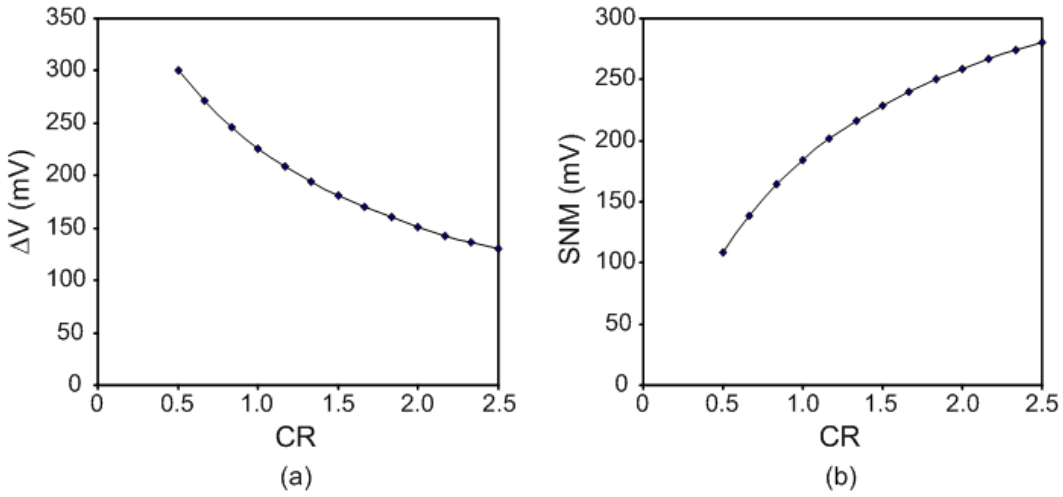


Figure 2.7: a) Logic '0' degradation as a function of cell ratio and b) static noise margin as a function of cell ratio. Simulated in 90nm CMOS technology with V_{DD}=1.0 V.

connected in series. Thus, transistors Q4 and Q6 form a voltage divider between V_{BLB} and ground and, develop a voltage ΔV across Q4. ΔV is often referred to as logic '0' degradation. The value of ΔV should be as low as possible to ensure a nondestructive read operation. In particular, ΔV should be less than the Q1-Q3 inverter's switching threshold plus some safety margin, i.e., the noise margin (NM).

As can be seen in Figure 2.6, ΔV depends on the ON resistance and hence on the relative sizes of Q4 and Q6. If we ignore the short channel effect and the body effect, ΔV can be calculated by equating the DC drain currents of Q4 (operating in the linear region) and Q6 (operating in the saturation region). After some mathematical steps, ΔV can be expressed as [1]:

$$\Delta V = \frac{V_{DSATn} + CR(V_{DD} - V_{THn}) - \sqrt{V_{DSATn}^2(1+CR) + CR^2(V_{DD} - V_{THn})^2}}{CR}, \tag{2.1}$$

where V_{DSATn} is the saturation drain voltage of the NMOS and CR is called the cell ratio, which is defined as $CR = \frac{W_4/L_4}{W_6/L_6}$. CR is the same for the other two transistors Q1 and Q2 since the cell is symmetrical.

Figure 2.7(a) shows the dependence of ΔV on CR. As evident from the figure, CR has to be greater than 1, i.e., the driver transistor has to be larger than the access transistor in order to limit ΔV and ensure a non-destructive read with adequate noise margin.

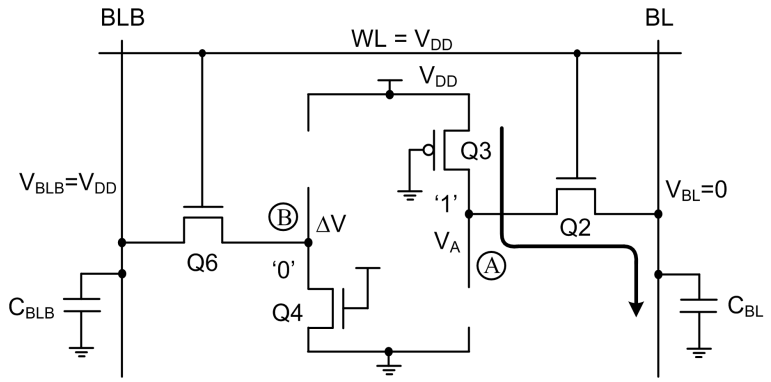


Figure 2.8: Simplified circuit of the 6T CMOS SRAM cell during a write operation.

Typically, CR can vary from 1 to 2.5 depending on the application of the cell. A larger CR provides higher read current (I_{READ}), which translates into higher speed. In addition, a larger CR increases the static noise margin (SNM) (see Figure 2.7(b)), which is defined as the minimum static noise at the storage nodes that can flip the cell. However, a larger CR means larger driver transistors (Q1 and Q4), which increase the cell area. Conversely, a smaller CR reduces the cell area while providing moderate speed and noise margin.

During the read operation, as soon as the complementary bit line voltage (V_{BLB}) discharges to a given voltage level (say, $V_{DD} - \Delta$) sufficient for reliable sensing by the sense amplifier, the sense amplifier is enabled. The sense amplifier then amplifies the small differential voltage Δ between the bit lines into the full-swing rail-to-rail voltage.

Write Operation

The write operation to the SRAM cell is also initiated by activating the WL. However, before the WL is raised to V_{DD}, one of the bit lines is pulled down to 0 V from its precharged state. In Figure 2.8, BL is pulled down to 0 V while BLB is kept at V_{DD}. When WL is raised to V_{DD}, the schematic of the cell can be simplified to the one shown in Figure 2.8.

The logic '0' voltage (V_B) cannot be pulled higher than ΔV, which is set by CR in order to ensure read data stability. Therefore, the new value can only be written into the cell by pulling down the logic '1' voltage (V_A). Thus, in an SRAM cell the writing is always done from the bit line that is at 0 V.

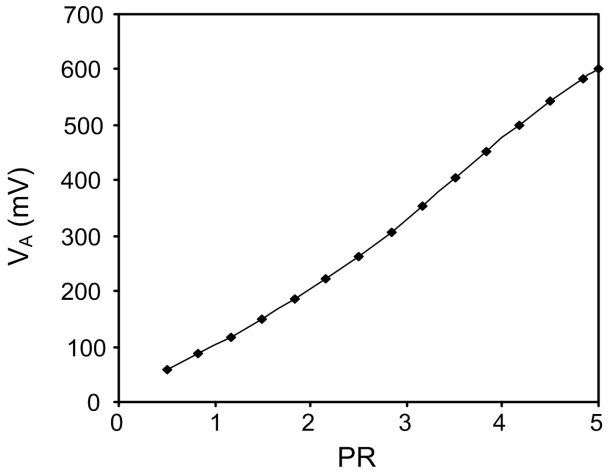


Figure 2.9: Logic '1' voltage as a function of cell pull-up ratio. Simulated in 90nm CMOS technology with V_{DD}=1.0 V.

In order to ensure reliable writing, V_A has to be pulled low enough so that the inverter formed by Q4 and Q5 can switch. Considering some extra margin, it is safer to pull down V_A below V_{TH} of Q4. The condition required for this action can be derived by equating the DC drain currents of Q2 and Q3. If $V_A \leq V_{THn}$, Q2 operates in the linear region while Q3 operates in the saturation region. Equating their drain currents yields, after some mathematical manipulations [1]:

$$V_A = V_{DD} - V_{THn} - \sqrt{(V_{DD} - V_{THn})^2 - 2\frac{\mu_p}{\mu_n} PR \left((V_{DD} - |V_{THp}|) V_{DSATp} - \frac{V_{DSATp}^2}{2} \right)}, \tag{2.2}$$

where μ_p and μ_n are the effective mobilities of the PMOS and NMOS transistors, respectively, and PR is the cell pull-up ratio, which is defined as $PR = \frac{W_3/L_3}{W_2/L_2}$.

The dependence of V_A on PR is shown in Figure 2.9. The lower the PR, the lower the value of V_A. If we wish to pull V_A below V_{THn}, then PR has to be less than 2, which primarily results from the higher mobility of the NMOS transistor (Q2). This constraint is met by using the minimum-sized PMOS pull-up and NMOS access transistors. However, a designer must assure that the writeability constraint is met under all process conditions. The worst case for the write operation occurs with a process with strong PMOS transistors and weak NMOS transistors coupled with a higher operating voltage.

From the discussion of the read and write operations on the 6T CMOS cell, it is evident that the cell access transistors have to be weak enough to ensure a nondestructive read on one hand, and have to be strong enough to ensure writeability on the other hand. This apparent contradictory design requirement can be met by choosing minimum width pull-up transistors (Q3 and Q5), minimum width access transistors (Q2 and Q6), and larger than minimum width (1.5 $\sim$ 1.7 times) driver transistors (Q1 and Q4). The channel lengths can be minimal or non-minimal depending on the subthreshold leakage constraints.

Despite the above design implications coupled with larger number of transistors compared to the other discussed cells, the 6T CMOS SRAM cell offers superior data stability, leakage performance, and speed. In addition, it is fully compatible with the standard logic process. Thus, the 6T CMOS cell has become the most widely used SRAM cell in today's SoCs.

2.2.2 Row Decoder

The row decoder selects one of the rows in the SRAM array by asserting the corresponding word line (WL) signal. Like any binary decoder, the row decoder enables one of 2^n WL signals with only n address bits. Typically, the SRAM address space is defined as the total number of address bits required to access a particular word (or a bit if the SRAM is bit-oriented). For example, a 1 Mb (2^{20}) SRAM having a word size of 32 (2^5) bits, will have 2^{15} words ($2^{20} \div 2^5$). Therefore, 15 address bits are required to access any word in this SRAM. The address bits are assigned to block, row, and column decoders depending on the size and internal organization of the SRAM. For instance, the 1 Mb SRAM can be organized in 32 blocks each having 256 rows and 128 columns.

The SRAM row decoder can have a single or multi-stage architecture. In a single stage decoder all decoding is realized using a single block, such as a wide NOR gate. The fan-in for the NOR gate equals the number of address bits. To simplify the circuit and reduce the layout area, such decoders are often designed using static PMOS transistor loads, as shown in Figure 2.10(a). The PMOS load can be gated by a precharge clock to realize the dynamic version of the decoder (see Figure 2.10(b)). However, implementation of a wide NOR gate single stage decoder poses several pressing challenges [1]. First, the

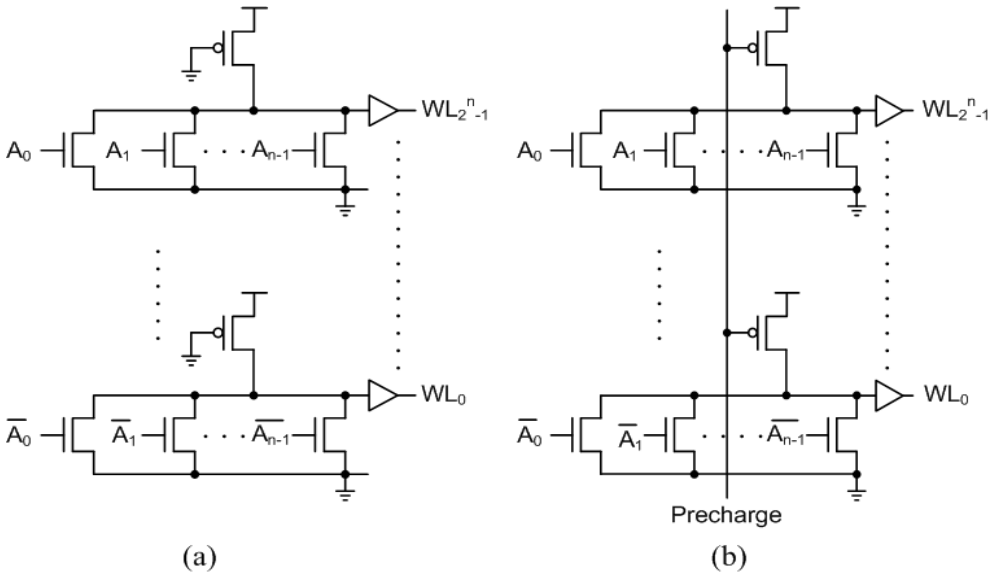


Figure 2.10: Single stage wide NOR row decoder: a) static and b) dynamic.

layout of the wide NOR gate must fit in the word line pitch. Second, the large fan-in of the gate severely affects its switching performance, thereby increasing the read/write access time. Third, the gate has to drive the large load of the WL while not overloading the input addresses. Fourth, the power dissipation has to be limited. Because of these challenges, a multi stage decoder is often a better alternative.

Multiple stage decoders employ several hierarchically-linked stages. Conventionally, the address bits are grouped and decoded at the first logic stage, which is often referred to as *pre-decoder*. Then another logic stage, which is referred to as *post-decoder*, works on the outputs of the pre-decoder to generate the final WL signal. As shown in the following example, such an arrangement offers a number of advantages over a single stage decoder.

The WL_0 in a 4-input active high decoder is given by:

$$WL_0 = \overline{A_0}.\overline{A_1}.\overline{A_2}.\overline{A_3} = \overline{A_0 + A_1 + A_2 + A_3}. \tag{2.3}$$

Equation (2.3) can be implemented by a 4-input AND gate (as in Figure 2.11(a)) or a 4-input single-stage NOR gate (as in Figure 2.10). However, (2.3) can also be implemented by two stages of 2-input AND gates, as shown in Figure 2.11(b). Since the fan-in of the AND gate has been halved, the gate delay reduces by approximately a factor of 4. As a result, the two stages exhibits a propagation delay that is only half that of the delay of the single-stage decoder. In addition, the two-stage decoder requires fewer transistors

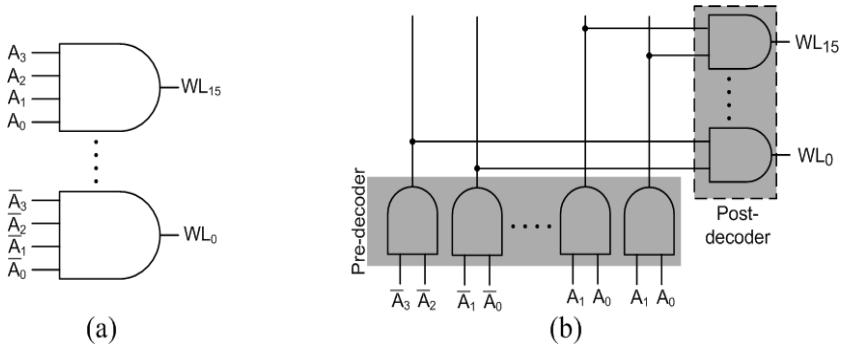


Figure 2.11: a) Single stage 4-to-16 AND decoder and b) two stage 4-to-16 AND decoder.

(144 vs. 160). The number of saved transistors significantly increases for large decoders [1].

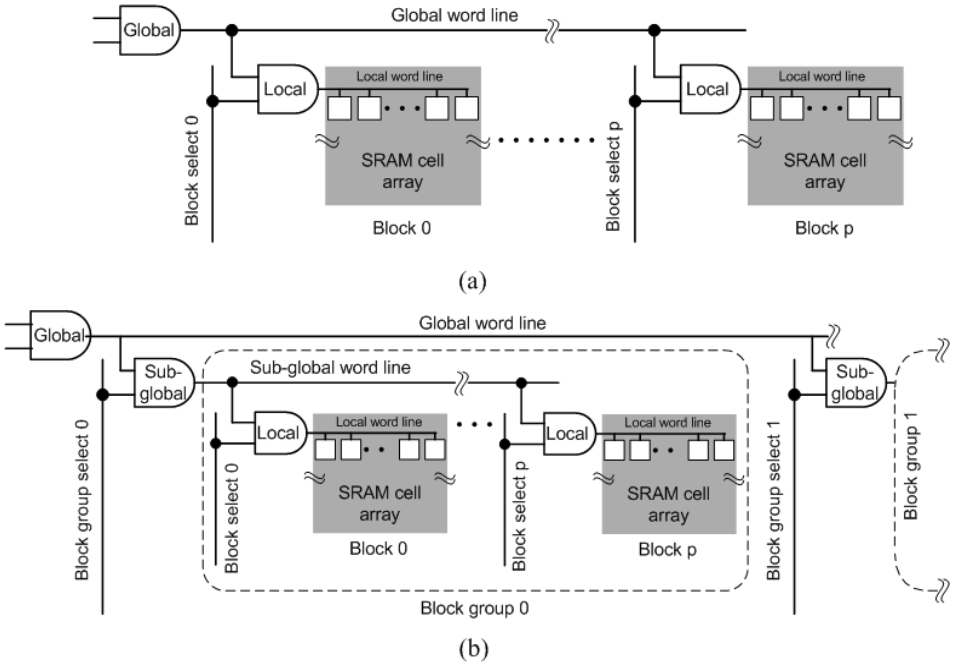


Figure 2.12: a) Divided word line and b) hierarchical word line decoder architectures.

In today's large SRAMs, the row decoders not only consist of pre and post decoders, but also employ several additional stages of decoding. The conventional Divided Word Line (DWL) structure shown in Figure 2.12(a) partitions the SRAM into blocks. A local or block word line is activated when both the global word line and the block select line are asserted. Since only one block is activated at any time, the DWL structure reduces both the word line delay and the power consumption. Incorporating an additional

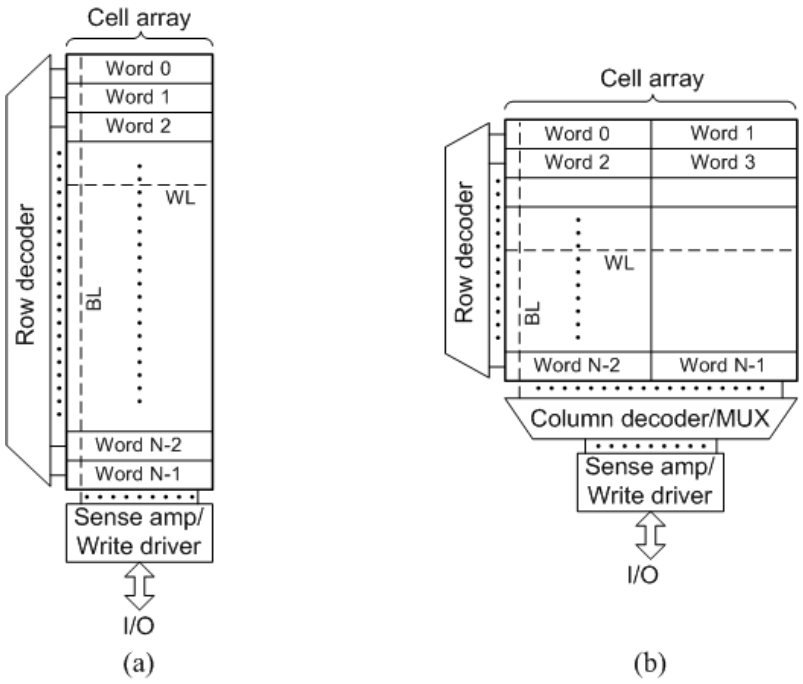


Figure 2.13: Simplified view of an N-word SRAM unit: a) without a column decoder and b) with a column decoder.

decoding level to the DWL, another decoder architecture called Hierarchical Word Decoding (HWD) has been proposed for larger than 4 Mb SRAMs (see Figure 2.12(b)) [30]. The HWD architecture offers ∼20% less delay and ∼30% lower total load capacitance compared to the DWL architecture.

2.2.3 Column Decoder or Multiplexer

A column decoder is a multiplexer (MUX) that facilitates the insertion of multiple words in a row and selecting one word during read/write access. The use of multiple words in a row makes the aspect ratio of the SRAM array closer to unity so that the WL and BL capacitances are in the same order of magnitude. Figure 2.13 illustrates the reduction in BL length due to the use of a column MUX.

Two typical implementations of a column MUX are shown in Figure 2.14. Which one to choose depends upon area, performance, and architectural considerations. Figure 2.14(a) shows a column MUX with PMOS pass-transistors and a 2-to-4 pre-decoder. When enabled by one of the outputs of the pre-decoder, the pass transistors pass the

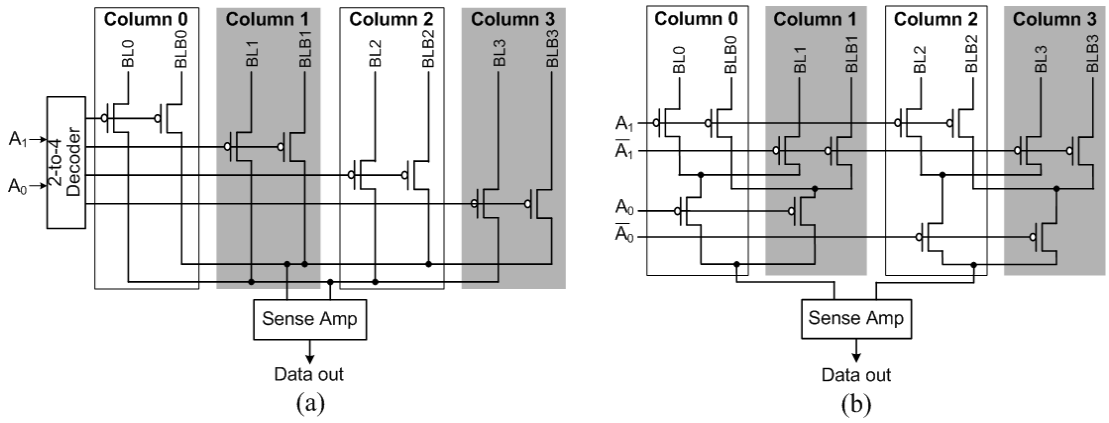


Figure 2.14: 4-to-1 column MUX: a) pre-decoder based and b) tree based.

read differential voltage from the BLs of one out of the four columns to the inputs of a sense amplifier. A simpler version of the column MUX is shown in Figure 2.14(b), which uses a binary tree decoder formed by PMOS pass transistors. This MUX requires no pre-decoding and utilizes fewer transistors. However, since the propagation delay increases quadratically with the number of sections, a large tree-based column MUX introduces extra delay and its usage may be prohibitively slow for large decoders [1]. It should be noted that if the column MUX is shared by both read and write operations, the pass-transistors in both of above implementations must be replaced by complementary transmission gates. This will enable passing full swing (rail-to-rail) voltage in both directions.

2.2.4 Sense Amplifier and Precharge Circuits

The sense amplifier (SA) in an SRAM is employed to perform the non-destructive read operation on a selected cell through the bit lines. Precharge and equalizer circuits, on the other hand, are used to precharge the bit lines to a specific voltage (typically at V_{DD}) before the sense amplifier operates (see Figure 2.15(a)). Thus, designing the sense amplifier and precharge circuits is critical for the functionality, performance, and reliability of the SRAM.

The primary function of a sense amplifier in an SRAM is to amplify a small differential bit line voltage and convert it to full swing digital signal, thus offering a number of advantages. First, the sense amplifier limits the highly capacitive bit line swing to a small voltage, which greatly saves power. Second, the sense amplifier allows the SRAM

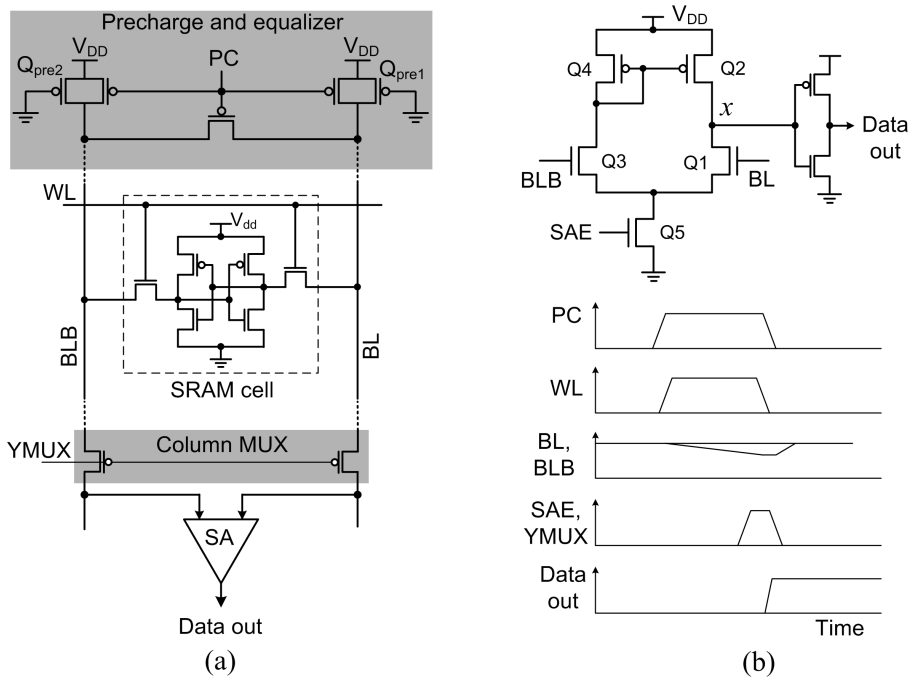


Figure 2.15: a) A typical SRAM column showing the sense amplifier and precharge circuits and b) a simple differential sense amplifier with current mirror load and corresponding timing diagram.

cells to be smaller since each individual cell does not need to fully discharge the bit line. Third, the sense amplifier significantly improves the read speed by avoiding a full swing of the bit line voltages. However, to efficiently serve these purposes, a sense amplifier designer needs to meet following performance objectives:

- high gain
- high sensitivity
- minimum delay
- minimum power
- reliable operation against offset voltage and under various PVT conditions.

At the same time, the designer is subject to following constraints:

- layout area due to tight pitch of SRAM columns

- power budget.

Meeting all of the design objectives while conforming to the constraints is difficult. Therefore, depending on the target application and operating conditions, the designer has to choose the sense amplifier topology and optimize it to serve the specific need. The sense amplifier topologies reported hitherto can be divided into two broad categories: voltage mode sense amplifier and current mode sense amplifier. Each of these topologies requires an in-depth discussion, which is beyond the scope of this thesis. In the following, we briefly discuss a few voltage mode sense amplifiers, which are easy to implement and less power consuming than current mode sense amplifiers.

The most basic single-stage differential sense amplifier with current mirror load is shown in Figure 2.15(b). The bit lines (BL and BLB) are fed to the differential input transistors, Q1 and Q3, while transistors Q2 and Q4 serve as an active current mirror load. The transistor Q5 drives the common source of Q1 and Q3, thus conditioning the amplifier by the enable signal, SAE. At the beginning of the read operation, the precharge and equalization signal PC is asserted. PC makes sure that the bit lines are weakly connected to V_{DD} only through the precharge transistors Q_{pre1} and Q_{pre2}, and the bit line voltages are equal. Then as the word line signal WL is asserted, one of the bit lines starts to drop from V_{DD}. Q_{pre1} and Q_{pre2} are properly sized so that their contention with the driver transistors of the SRAM cell does not flip the cell. This puts a sizing constraint on the precharge transistors. The sizing of these transistors determines the bit line recovery speed, which is especially critical after a write operation when the bit line is completely discharged.

Once sufficient differential voltage is developed between the bit lines, the SAE is enabled and the amplifier evaluates. The gain of the amplifier at node x is given by,

$$A = -g_{m1} \left(r_{01} || r_{02} \right), \tag{2.4}$$

where $-g_{m1}$ is the transconductance of Q1 (=Q2)and r_{01} and r_{02} are the small signal output resistances of Q1 and Q2, respectively. For larger gain, $-g_{m1}$ can be increased by widening the input transistors or by increasing the bias current (i.e., widening Q5). The latter also reduces r_{02}, which undermines the effectiveness of this approach. Typically, the gain is set to around 10. The main goal of the sense amplifier is the rapid production of the output signal. Gain is hence secondary to the response time [1].

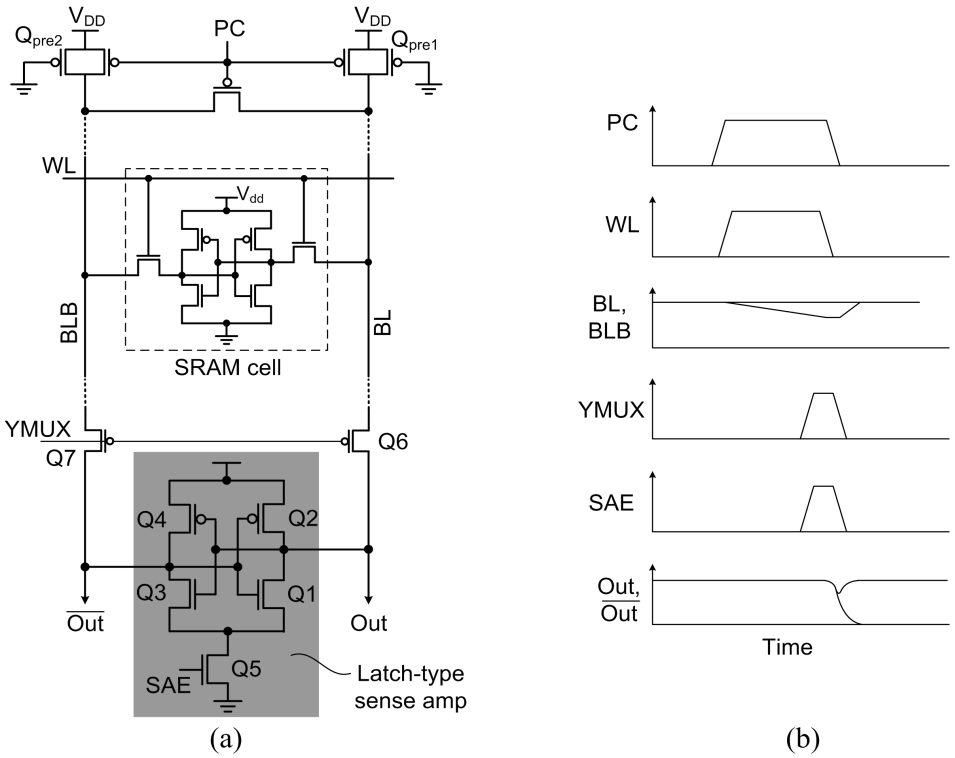


Figure 2.16: a) A latch-type sense amplifier in an SRAM column and b) corresponding timing diagram.

The basic differential sense amplifier has the advantage of high input impedance as the bit lines are connected to the transistor gates. This arrangement also isolates the output of the sense amplifier from the inputs (unlike latch type sense amplifier). The voltage divider action of the serially-connected driver, access and the precharge transistors prevents the complete discharge of the bit lines. Thus, the word line deactivation timing requirements can be relaxed as the bit line discharge will stop at the potential defined by the relative sizing of the precharge, access and driver transistors. However, the basic differential sense amplifier has some drawbacks as well. Its high sensitivity to transistor mismatches causes increased offsets. To compensate for possible offsets, the minimum differential voltage needs to be increased, which in turn slows down the sensing. This issue coupled with the sizable power consumption causes the usage of the basic differential sense amplifier to decline in the scaled-down technologies.

A better alternative to the basic differential sense amplifier is a latch-type sense amplifier, which is shown in Figure 2.16. The amplifier is formed by a pair of cross-

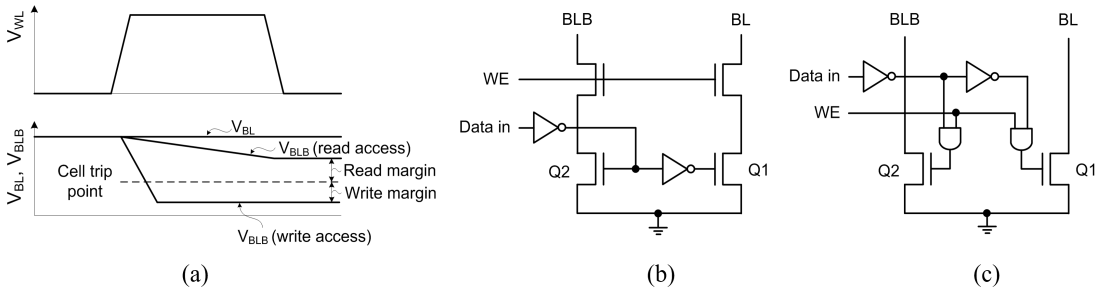


Figure 2.17: a) Illustration of read and write margins, b) write driver using stacked NMOS transistors, and c) write driver using AND gate and NMOS pull-down transistor.

coupled inverters. The sensing starts with biasing the sense amplifier in the high-gain region by precharging and equalizing its inputs to V_{DD}. Unlike the current mirror based basic differential sense amplifier, the inputs are not isolated from the outputs in the latch-type sense amplifier. Therefore, additional transistors, Q6 and Q7 are needed to isolate the latch-type SA from the bit lines and prevent the full discharge of the bit line carrying a logic '0', which costs extra power and delay. When a cell accessed by the word line WL generates sufficient voltage differential between BL and BLB (see Figure 2.16(b)), the SAE signal is enabled. As a result, the column MUX/isolation transistors Q6 and Q7 are turned off, isolating the highly capacitive bit lines from the sense amplifier and preventing the complete discharge of the bit line capacitances. Then, the positive feedback of the cross-coupled inverters Q1-Q2 and Q3 - Q4 quickly drives the low-capacitance outputs Out and $\overline{Out}$ to the full swing complementary voltages.

2.2.5 Write Driver

The write driver enables writing into an SRAM cell by pulling down one of the bit lines of the selected column from the precharge level to below the write margin (see Figure 2.17(a)). Typically, the write driver is enabled by the Write Enable (WE) signal. The order in which the word line and WE are enabled is not crucial for the correct write operation.

Figure 2.17 shows two typical write driver circuits. The write driver in Figure 2.17(b) uses stacked NMOS transistors, the bottom two (Q1 and Q2) of which are driven by the input data ($Data\ in$). Accordingly, either Q1 or Q2 is turned on depending on the value

of *Data in*. When WE enables the upper two NMOS transistors, the corresponding bit line (BL or BLB) is discharged to the ground level. Another implementation of the write driver is shown in Figure 2.17(c). When WE is asserted, depending on the value of *Data in*, one of the AND gates turns on either Q1 or Q2, which discharges the corresponding bit line.

It should be noted that a write operation can be carried out faster than a read operation even though a greater discharge of the highly capacitive bit lines is required for the former. In addition, only one write driver is needed for each column. As a result, the large area required by the pull-down transistors (Q1 and Q2) of a write driver does not pose any challenge in the layout.

2.2.6 Timing and Control Circuits

The timing and control circuits generate the precharge (PC), word line (WL), sense amplifier enable (SAE), and write enable (WE) signals to ensure correct read and write operations. The read cycle involves a tight timing relationship between address latching, PC activation, row and column decoder activation, and SAE activation. If the WL signal precedes PC, then cells on the activated word line will see both the bit lines pulled high and the accessed cells may flip their states. Another timing hazard may arise if the address changes before the read operation is complete. In this case more than one SRAM cell will be discharging the bit lines which may lead to reading erroneous data. Similarly, if the SA is enabled during the write operation, a "write through" can occur and the data being written will appear at the output without an intended read operation. In fact, the control signal path delays must match the delays of target signal, such as, address decoding, bit line discharge, etc., for fast and power-efficient SRAM operation. The variability in delay are dominated by the bit line delay since the minimal-size transistors in SRAM cells are more susceptible to process variations. The timing and control block should provide sufficient timing margins to account for the worst-case process conditions. Thus, designing the timing and control block is a challenging part in any SRAM design. The challenge becomes even harder with technology scaling, which reduces the gate overdrive voltage and increases V_{TH} fluctuations and process variations [31].

Typically three methods are used to implement the timing and control block in

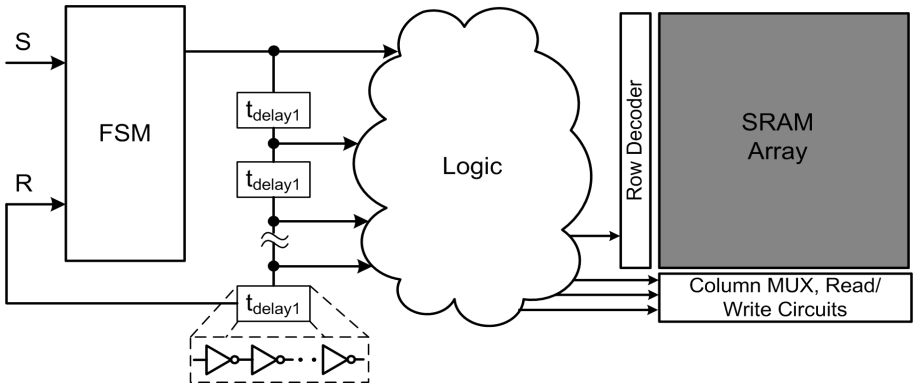


Figure 2.18: Functional diagram of delay-line based timing block.

SRAMs:

- Timed by the clock phase (direct clocking) [32];
- Delay line using a multitude of inverters to define the timing intervals [33]; and
- Self-timed replica (dummy) loop mimicking the signal path delay [34].

The timing method that uses the direct clocking of the WL and SA has limited operation speed due to the larger timing margins necessary for reliable operation. The delay line method allows faster operation than the direct clocking method. However, the delay of the delay loop may not track the delay variability caused by the process variations in modern nanoscale technologies. The self-timed replica (dummy) loop method has proven itself to be the most robust and precise in tracking the process variations and in maintaining tighter timing margins for faster operation. Since the delay-line based timing method has been used in this work to design the SRAM test chip, we will discuss it in more detail.

A functional diagram of a delay-line timing block is shown in Figure 2.18. A control signal S sets the finite state machine (FSM). The timing loop is defined by the total delay through the delay elements $t_{delay1} - t_{delayn}$ in the FSM reset path. Typically, the delay elements are realized using serially connected inverters. The delay time can be extended by using nonminimal length devices in the inverters or by utilizing current-starved inverters. The timing intervals formed by the delay elements $t_{delay1} - t_{delayn}$ as well as some logic stages are used to generate the required control signals for the read/write timing.

2.3 Soft Error Susceptibility of SRAM

Any part of the SRAM, e.g., the decoders, sense amplifier, write driver, and timing control circuit, is susceptible to particle-induced single event transients (SETs). However, due to the short duration of the transient ($\sim$ 200-400ps), the dynamic behaviour of these circuits coupled with their finite timing relationship, the probability of the SET appearing at the IO as an error is small. On the other hand, if a SET occurs in a cell or group of cells in the array, the SET can easily alter the stored bits. Those bits cannot go back to their previous states until they are rewritten, implying the occurrence of soft errors. Thus, the array or the cell is the most vulnerable part of the SRAM to soft errors. The high packing density, the use of smallest geometry transistors, and the large volume accentuate the vulnerability.

2.4 Low Power SRAMs and Soft Errors

Like for other on chip blocks, SRAM power consumption is an important issue for an SoC. Each of the building blocks discussed in the previous section can be optimized or specially designed for leakage and active power savings. However, leakage power is the primary concern for an SRAM since it typically has low data activity. Even when an SRAM is active, only a single row of the entire array is accessed, leaving all other rows in inactive mode. In addition, an SRAM array, which occupies the largest area with highest transistor density, consists of minimum sized transistors. These transistors exhibit higher sub-threshold leakage and process-induced variability, particularly in sub-100nm technologies. Accordingly, different leakage reduction techniques have been proposed to limit the leakage power consumption of the SRAM array. However, these techniques pose a potential threat to the soft error performance of the SRAM. In the following, we discuss different low-leakage SRAM architectures and investigate their soft error performance.

2.4.1 Gated Ground SRAM

The gated-ground SRAM cell reported in [35] uses an extra NMOS transistor on the path to ground (see Figure 2.19). This extra NMOS acts as a switch to shut off the path to

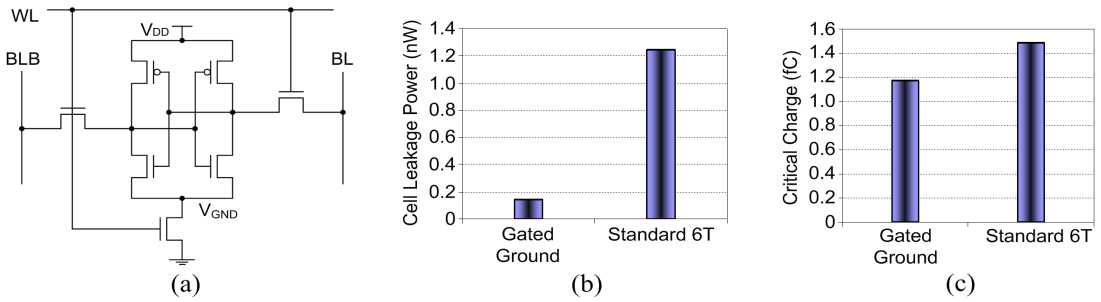


Figure 2.19: A gated-ground SRAM cell: a) schematic, b) leakage power compared to the standard 6T cell, and c) critical charge compared to the standard 6T cell. Simulated in PTM 65nm technology.

ground when the memory cell is not accessed (WL=0). This brings in the stack effect, where the non-zero virtual ground (V_{GND}) potential applies reverse body bias (RBB) on the upper NMOS transistors, i.e., the driver NMOS transistor and the access transistor connected to it. In addition, the non-zero V_{GND} reduces leakage through the load PMOS and access transistors by reducing the drain-induced barrier lowering (DIBL) effect. As a result, the cell leakage drastically decreases. The extra NMOS is turned on only when the bit in the cell is read or written (WL='1').

To evaluate the soft error susceptibility of the gated-ground cell, we extract its Q_{crit} using SPICE simulation (PTM 65nm) when the cell is in low leakage mode, i.e., the extra NMOS is OFF. We see that the cell can reduce leakage power by 88.5% (V_{DD}=1V). However, it also reduces the critical charge by approximately 21% for V_{GND}= 0.3 V. Thus, the gated-ground SRAM cell is expected to exhibit higher SER. The decrease in Q_{crit} can be attributed to the reduced voltage difference between the logic '1' and logic '0' levels. If the V_{GND} potential is higher, then Q_{crit} will be even lower.

2.4.2 SRAM with Sleep Transistor

This technique is similar to the gated-ground technique but is implemented at the block level. In this technique, nonaccessed SRAM blocks are isolated from the ground using an NMOS sleep transistor. The resulting V_{GND} potential can be controlled or programmed by inserting bias transistors as shown in Figure 2.20 [36]. Thus, depending on V_{GND}, the leakage reduction can be 2x∼1000x. However, like the gated-ground SRAM, this

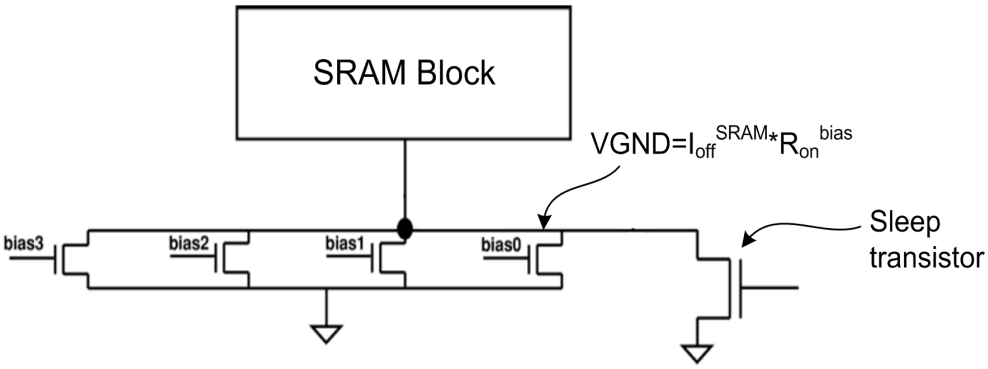


Figure 2.20: SRAM architecture with sleep transistor.

technique lowers the critical charge and increases the soft error susceptibility.

2.4.3 Drowsy Cache

This technique lowers the leakage power by using dynamic voltage scaling (DVS) while using the standard 6T-SRAM cell structure [37]. In this technique, the supply voltage of un-accessed or drowsy cells is set to a lower voltage (V_{DD-LOW}) while the supply of accessed cells are restored to the nominal V_{DD} using controlled switches at the cell's pull-up path (see Figure 2.21(a)). Thus, the DIBL effect in OFF transistors and the gate-to-channel electric field, and hence gate-to-channel leakage, in ON transistors are reduced, resulting in significant leakage power savings.

In order to retain a value in the SRAM cell, V_{DD-LOW} can be set to just about 1.5 times the threshold voltage, V_{TH}. Thus, for a 65nm technology-based SRAM cell that normally operates at 1.0 V, V_{DD-LOW} can be reduced to 0.3 V and substantial leakage energy can be saved. However, a single cycle penalty is incurred when accessing a drowsy cache line, as the supply rails have to be restored to 1.0 V before any read or write operation.

In our simulations, we see that the drowsy cache technique is very effective at reducing the leakage power ($\sim$82% at V_{DD-LOW}=0.3 V); however, it significantly lowers ($\sim$95% at V_{DD-LOW}=0.3 V) Q_{crit} (see Figure 2.21(b) and (c)). The reduction in Q_{crit} can be attributed to decrease in operating voltage as well as to weaker restoring current following the particle-induced transient. Thus, drowsy caches are extremely susceptible

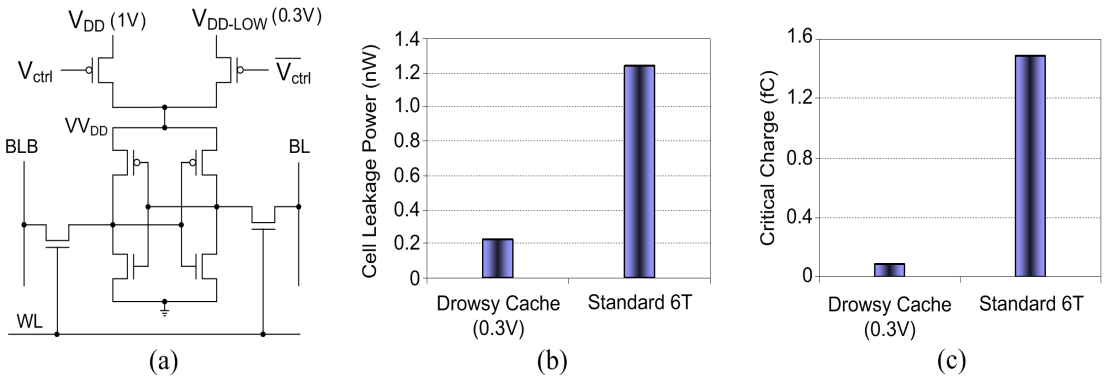


Figure 2.21: A drowsy cache cell: a) schematic, b) leakage power compared to the standard 6T cell, and c) critical charge compared to the standard 6T cell. Simulated in PTM 65nm technology.

to soft errors when they are in the drowsy state.

2.4.4 Leakage-Optimized Dual-V_{TH} SRAM

This technique carefully selects the V_{TH} of the transistors in the SRAM cell to reduce the leakage current [38]. A simple solution to the leakage power problem can be to increase the V_{TH} of all six transistors. However, high V_{TH} transistors are slow and cause a significant performance penalty. Therefore, instead of using all high V_{TH} transistors, high V_{TH} is chosen for transistors that are normally leaking when storing '1' or '0'. Since the leaky transistors are different whenstoring a logic '1' than for storing a logic '0', the optimization results in two arrangements of high V_{TH} transistors (see Figures 2.22 and Figure 2.23). Thus, the cell becomes asymmetric in terms of V_{TH}.

We first consider an asymmetric SRAM cell optimized for storing a '1' as shown in Figure 2.22. The difference between high and low V_{TH} has been assumed 100 mV. The resultant leakage power and Q_{crit} of the cell are shown in Figure 2.22(b) and Figure 2.22(c), respectively. As it can be seen, the leakage power is reduced by 67% while Q_{crit} is increased by almost 8%. The increase in the Q_{crit} can be attributed to the weaker pull-up of the high-V_{TH} PMOS at the non-struck node and hence a slower bit flipping process. If the flipping process is slow, the restoring transistor can supply more charge and thus increase Q_{crit} (this mechanism will be clarified in the next chapter). Therefore, an asymmetric SRAM cell, which is leakage optimized for storing a logic '1', is less

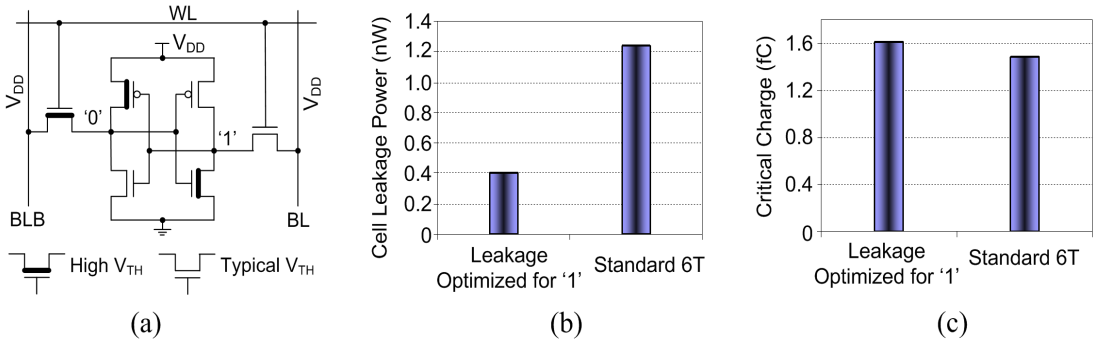


Figure 2.22: A leakage-optimized asymmetric SRAM cell for logic '1': a) schematic, b) leakage power compared to the standard 6T cell, and c) critical charge compared to the standard 6T cell. Simulated in PTM 65nm technology.

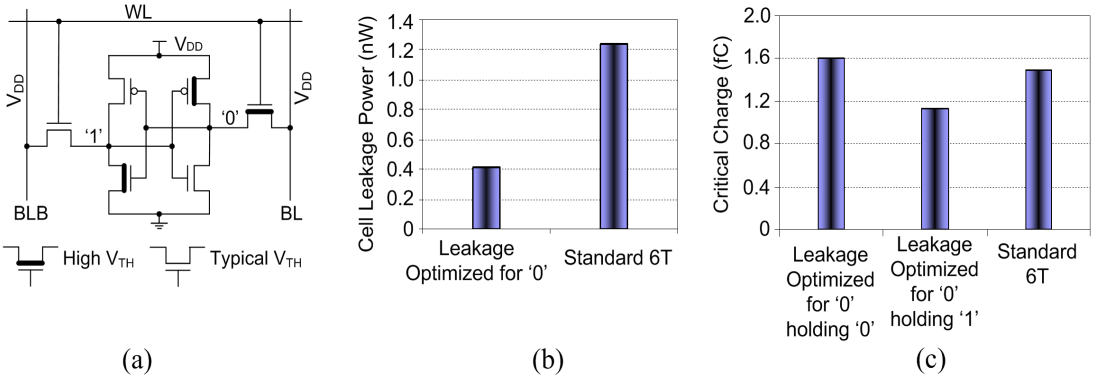


Figure 2.23: A leakage-optimized asymmetric SRAM cell for logic '0': a) schematic, b) leakage power compared to the standard 6T cell, and c) critical charge compared to the standard 6T cell. Simulated in PTM 65nm technology.

susceptible to soft errors compared to the traditional 6T SRAM cell. However, if the cell stores a logic '0', no leakage reduction will be achieved and the soft error susceptibility will increase.

The asymmetric SRAM cell optimized for storing '0' is shown in Figure 2.23. The leakage reduction in this cell is also 67% and no leakage reduction is achieved if the cell stores a logic '1'. Q_{crit} is the same as the cell optimized for logic '1', since Q_{crit} is extracted by injecting the noise current at logic '1' node in both cells. However, Q_{crit} will be smaller ($\sim$ 24% as shown in Figure 2.23(c)) if the cell stores a logic '1' because of smaller restoring current supplied by the high-V_{TH} PMOS load.

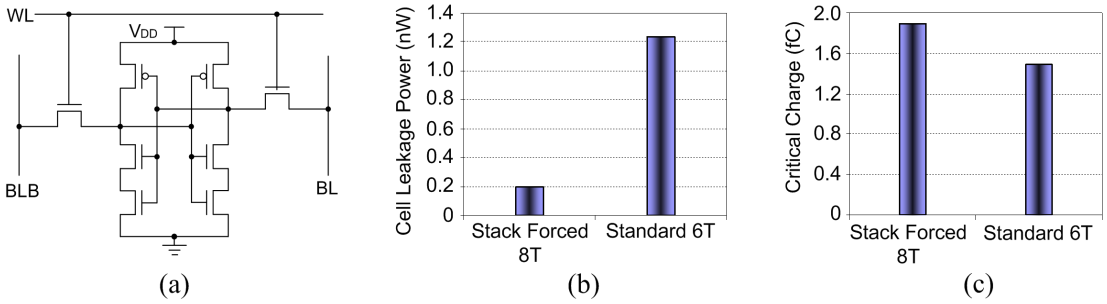


Figure 2.24: A Stack-forced SRAM cell: a) schematic, b) leakage power compared to the standard 6T cell, and c) critical charge compared to the standard 6T cell. Simulated in PTM 65nm technology.

2.4.5 Stack-Forced SRAM

Finally, we consider an SRAM cell where the pull-down NMOS transistors are stack forced by two additional NMOS transistors [39]. The resulting circuit comprises 8 transistors (8T) as shown in Figure 2.24. In this cell, the leakage current is significantly reduced by the stacking effect (RBB on the upper NMOS transistors).

The stack forced cell cell exhibits 27% increase in Q_{crit} while 84.3% reduction in the leakage power. Thus, the stack forced cell gains in both ways. The increase in Q_{crit} can be attributed to an increase in the effective node capacitance due to the addition of gate capacitances of the stacking transistors.

The leakage reduction capability and soft error susceptibility of above mentioned low-power techniques are summarized in Table 2.1. As evident, gated-ground technique provides the minimum leakage and stack forcing provides the maximum soft error robustness while the drowsy cache shows the minimum soft error robustness. In addition, the stack forced cell shows leakage reduction close to the gated-ground cell. Thus, the stack forced cell could be a good choice if we want to achieve low-power operation with increased soft error robustness. However, due the size (8T) of the cell, the area overhead becomes significantly large, undermining the attractiveness of the cell.

Table 2.1: Leakage and soft error performance of different low-leakage SRAMs

Low-power Technique	ΔLeakage (%)	ΔCritical Charge(%)
Gated-ground	-88.5	-21
Drowsy cache (0.3V)	-82	-95
Leakage optimized (1)	-67	+8.3
Leakage optimized (0)	-67	-24
Stack forced	-84.3	+27

2.5 Summary

In this chapter we have presented an overview of the architecture i.e., the basic building blocks of an SRAM and their operation. We have discussed typical SRAM cell design issues, different row and column decoder schemes, sense amplifier and associated precharge circuits, and timing control circuits. In addition, we have discussed different leakage reduction approaches in SRAM and investigated the resulting impact on the soft error performance. We have found that all low-power schemes that reduces the rail-to-rail voltage of the cell have lower critical charge, which can be translated into higher soft error rate. Thus, this chapter has provided the necessary background on SRAM and justifies the need for soft error characterization and mitigation, paving the way for the following chapters.

Chapter 3

Existing Soft Error Characterization and Mitigation Approaches

This chapter reviews the current status of soft error modeling and mitigation techniques in SRAMs and identifies their limitations.

Since the memory array is the most vulnerable part, conventional soft error modeling approaches have primarily modeled the critical charge, which is a key to assessing the soft error vulnerability of the SRAM cell and hence the array. Soft error mitigation approaches, on the other hand, targeted the fabrication process, the cell as well as the architecture of the SRAM.

3.1 Critical Charge Models

As mentioned earlier, the critical charge (Q_{crit}) is the minimum amount of charge that can flip the stored bit in an SRAM cell. Q_{crit} is conventionally calculated by injecting a noise current at the storage node (see Figure 3.1) and then integrating the current that can flip the cell. Several Q_{crit} models have been reported to date. All of these models agree in the qualitative definition of Q_{crit}, however, differ in the quantitative description. For

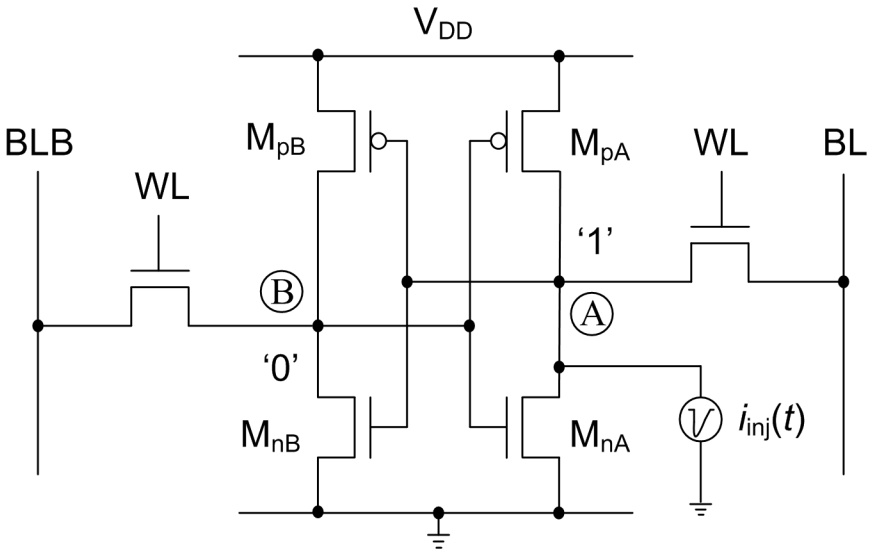


Figure 3.1: 6T SRAM cell with a current source to mimic a particle strike at node A.

example, in [40] and [41], Q_{crit} has been modeled as a sum of capacitance and conduction components:

$$Q_{crit} = C_N V_{DD} + I_{DP} T_F \tag{3.1}$$

where C_N is the equivalent capacitance of the struck node (node A in Figure 3.1), V_{DD} is the supply voltage, I_{DP} is the maximum current of the 'ON' PMOS transistor (M_{pA} in Figure 3.1), and T_F is the cell flipping time. While both the capacitance and conductance components indeed contribute to Q_{crit}, the former in (3.1) has been overestimated. This is due to the fact that the flipping threshold of an inverter is less than V_{DD} (say, $V_{DD}/2$ for perfectly matched NMOS and PMOS). In addition, the conductance term ($I_{DP}T_F$) in (3.1) only considers the peak value of current, which is not realistic for the time-varying restoring current supplied by M_{pA}. These issues have been addressed to some extent by Xu *et. al.* [42] by defining the critical charge in the following way:

$$Q_{crit} = \int_0^{V_{trip}} C_N dV + \eta I_P T_{pulse} = C_N V_{trip} + \eta I_P T_{pulse} \tag{3.2}$$

where V_{trip} is the static tripping point of the SRAM cell, η is a correction factor, I_P is the driving current of M_{pA}, and T_{pulse} is the duration of the particle-induced current pulse. Equation (3.2) provides a better estimate of the capacitance component of Q_{crit}, particularly the effect of junction capacitance and the addition of backend MIM capacitor. However, (3.2) fails to incorporate the dynamics of the voltage transient at the struck

node, the quantitative description of I_P, and the contributions of different transistors that constitute the cell. As a result, the effectiveness of (3.2) in estimating Q_{crit} under process non-idealities becomes limited.

Recently, Zhang et. al. [43] have presented an analytical technique to calculate Q_{crit} in terms of transistor parameters and injected current's magnitude and duration. The most appreciable feature of this technique is that it considers the cell's dynamic response to a particle strike and the non-linear coupling of storage nodes. However, the calculated value of Q_{crit} in this technique exhibits as high as 11% discrepancy with SPICE simulations as reported in [43]. Our simulations have found the discrepancy to be even higher. The underlying reason can be attributed to i) using a rectangular current pulse instead of an exponential pulse to model the noise source and then mapping the former with the latter, and ii) ignoring the current components of the PMOS transistors for logic '0' hit (i.e., NMOS for logic '1' hit). In addition, considering the contributions of only one type of transistor (either NMOS or PMOS) undermines the effectiveness of the technique in determining Q_{crit} under process-induced variations in different transistor parameters.

IBM has developed a reliable simulation tool named Soft Error Monte Carlo Model or SEMM for estimating the soft error rate (SER) in integrated circuits [44]. In contrast to circuit-level models presented in (3.1) and (3.2), SEMM includes device, process and technology parameters, their statistical variations across the chip, and the event-by-event treatment of particle hits. Thus, SEMM can make more realistic and accurate estimates of the chip SER. However, SEMM is a post-design SER simulation tool for the entire chip and cannot be efficiently used while designing a single SRAM cell. Therefore, to provide the designers with a simple and accurate model of the soft error critical charge and address the shortcomings of existing models, an improved model is essential.

3.2 Mitigation of Soft Errors in SRAM

A variety of mitigation techniques have been reported to limit the SER in SRAMs. These techniques can be classified into three major categories:

- process techniques
- circuit techniques

- architecture techniques

In the following, we briefly discuss each of these techniques.

3.2.1 Process Techniques

The primary method for soft error mitigation at the process level is to reduce the charge collection at sensitive nodes. This can be accomplished in SRAMs by introducing extra doping layers to cut off particle induced funneling tails, thereby reducing the collected charge [45]. In advanced SRAMs, triple-well [46] and even quadruple-well [47] structures have been proposed to limit the charge collection. Use of an epitaxial substrate instead of a bulk substrate also decreases the soft error susceptibility by reducing the funneling effect.

Another effective technique for reducing charge collection is to use the SOI substrates. Unlike bulk CMOS, SOI devices collect less charge from an alpha or neutron strike because the silicon layer is much thinner. IBM reports a 5 times reduction in the SER of SRAM devices fabricated in partially-depleted SOI technology [48]. Fully-depleted SOI, in which the silicon layer almost disappears, has the potential to offer further reduction in SER. However, volume manufacturing of fully-depleted SOI chips is still a challenge.

Although process-level techniques significantly improve the soft error performance of SRAMs, the techniques do require modification of standard CMOS process. Thus, companies that do not have control over the process (e. g., fabless companies) cannot use these techniques. In addition, these techniques incur additional processing cost, which undermines their attractiveness.

3.2.2 Circuit Techniques

Circuit and architecture-level techniques provide easier solutions to reduce the SER compared to the process techniques. In circuit techniques, the SRAM cell is made soft error hardened either by slowing down the response of the circuit to fast transients or by increasing Q_{crit}.

Figure 3.2 shows a 6T SRAM cell with extra resistors added in the feedback path to decouple the sensitive nodes [49]. These resistors increase the RC delay around the

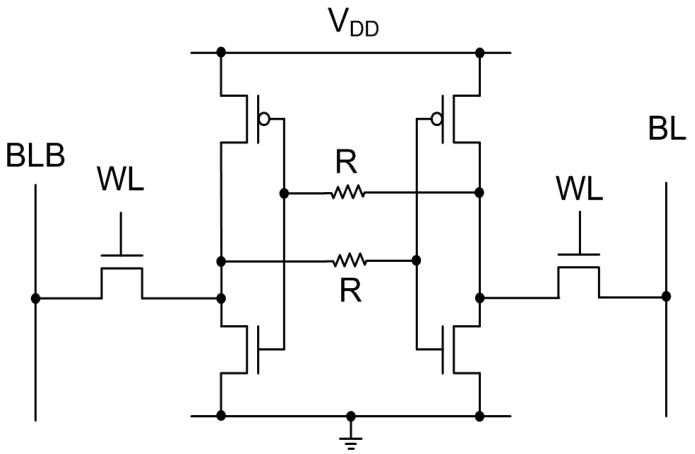


Figure 3.2: A soft error hardened SRAM cell with feedback resistors.

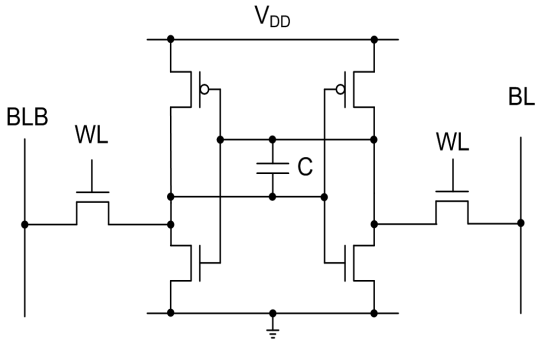


Figure 3.3: A soft error hardened SRAM cell with coupling capacitor.

feedback loop, thus slowing down the propagation of a particle-induced transient from one of the sensitive nodes to the other. Consequently, short-lived transients cannot disturb the other node sufficiently and the cell eventually recovers to its initial state. While this technique is very effective at increasing the soft error immunity of SRAMs, it causes significant speed and area penalty [3]. The RC delay increases the cell write time since the write process in SRAM is similar to the transient event. The feedback resistors, which are typically implemented by lightly-doped polysilicon regions, incur extra silicon area and process complexity. In addition, these resistors are very sensitive to the doping concentration of the polysilicon as well as the operating temperature.

A better alternative to the SRAM cell with feedback resistors is shown in Figure 3.3 [50]. Here, a coupling capacitor is placed between the sensitive nodes. Since the transient voltage changes at the sensitive nodes occur in opposite directions, the effective

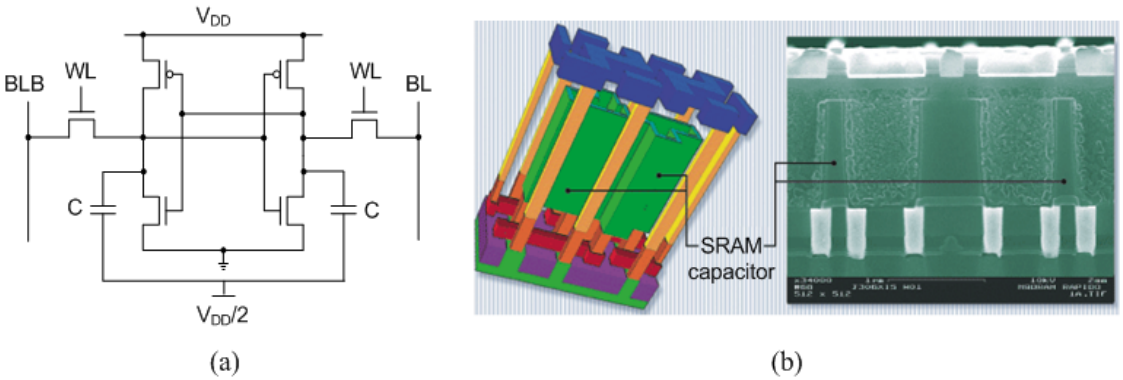


Figure 3.4: A soft error hardened SRAM cell with 3D node capacitors: a) circuit diagram and b) 3D SEM image. Source: ST Microelectronics.

capacitance seen by each node is twice the actual value of the capacitance due to the Miller effect. Thus, the critical charge of the cell is significantly increased, resulting in an appreciable decrease in the SER. The cell is realized by stacking the coupling capacitor on top of the cell, thus avoiding any area penalty. In addition, the cell is less temperature-sensitive compared to the cell with resistive feedback. However, this technique requires extra process steps for realizing the capacitor. In addition, the coupling capacitor increases the write time and writing power ($\propto CV^2$) of the cell.

Figure 3.4 shows another soft error hardening technique, which also increases the cell critical charge by adding extra capacitors [51]. Unlike the coupling capacitor, these capacitors are separately placed at the sensitive nodes of the SRAM cell. These capacitors have 3D structures (see Figure 3.4(b)) and are implemented between the contact layer and metal-1 layer using a standard embedded DRAM process flow. Thus, no extra area is required for the capacitors. However, the cell biases the common node of the capacitors at $V_{DD}/2$, which requires extra bias circuitry. In addition, the cell suffers from the same drawbacks as the cell with coupling capacitor.

Sometimes both the coupling capacitor and the feedback resistors are used in the SRAM cell as shown in Figure 3.5 for maximum soft error immunity in the expense of speed and process simplicity [52]. The increase in the critical charge by this technique as well as the techniques described earlier is shown in Figure 3.6. As expected, the cells with coupling capacitors exhibit significant increase in the critical charge due to the Miller effect.

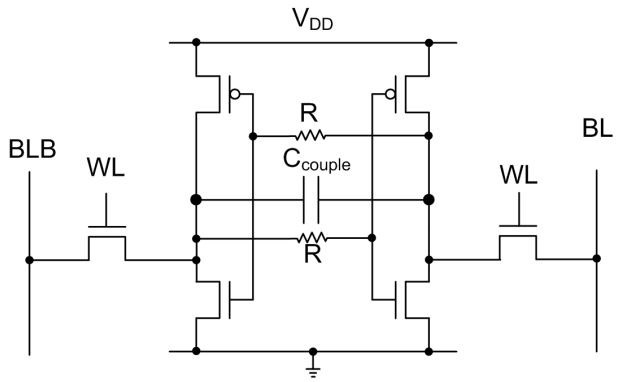


Figure 3.5: A soft error hardened SRAM cell with coupling capacitor and feedback resistors.

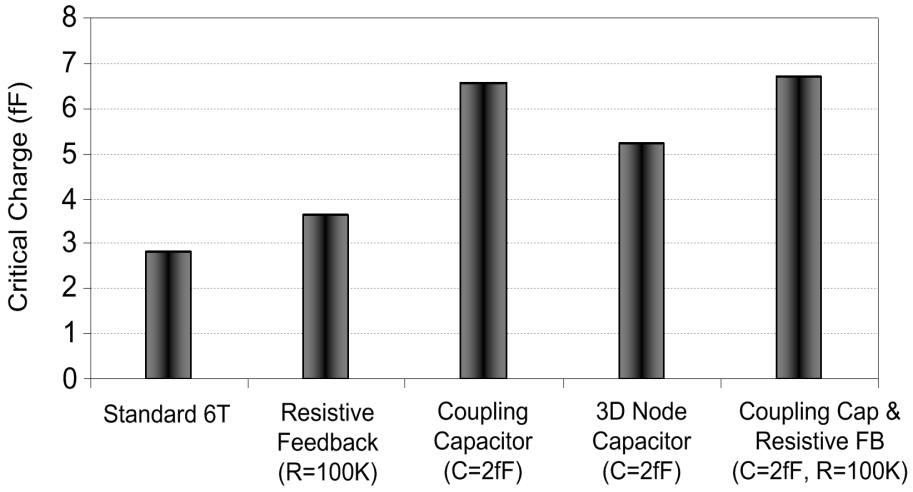


Figure 3.6: Critical charge for different soft error hardened SRAM cells. Simulated in 130nm CMOS technology

An SRAM cell with redundant states has also been proposed [53]. In such a cell each of the logical '0' and '1' states is stored as a combination of four node voltages. In the dual interlocked cell (DICE) shown in Figure 3.7, the logic '0' state corresponds to X1=0, X2=1, X3=0, X4=1 while the logic '1' state corresponds to X1=1, X2=0, X3=1, X4=0. In any of these states, if any of the nodes are struck by a particle, there are always two consecutive nodes (among the remaining three nodes) that have the values '1' and '0'. These two nodes are referred to as the hold nodes and the other two nodes as the affected nodes. When the state of the affected node can be modified by particle strike, the hold nodes preserve their correct values. Since one transistor of each inverter driving one of the

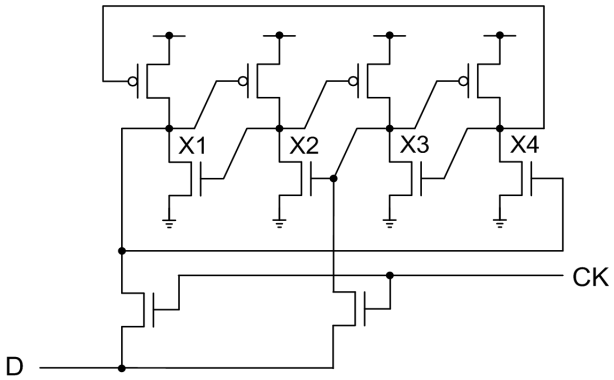


Figure 3.7: Soft error hardened dual interlocked storage cell (DICE).

affected nodes is driven by one hold node, currents through these transistors can quickly restore the correct values at the affected nodes.

The DICE cell provides excellent soft error immunity; however, it incurs about 81% area overhead in addition to increased word line drive complexity.

3.2.3 Architecture Level Techniques

While the circuit techniques are able to improve the SER performance of the SRAM, they incur significant area overhead. Since these techniques add additional components (R, C, etc.) or devices to the cell, the array size become significantly larger than the un-hardened array. Such large area penalty can be avoided by using architecture level techniques. In fact, there are three factors that make architecture-level mitigation techniques more attractive than circuit-level techniques. First, the definition of what an error is, in fact, lies at the architecture level. An error on a cell may not cause a problem if the cell undergoes a write operation before the read operation. Moreover, the error may result from physical weakness of the cell (such as high leakage) in addition to a particle strike. In that case, circuit hardening cannot help. Second, architecture-level solutions can incur less overhead than circuit-level solutions. For example, a single error correcting double error detecting (SECDED) error correction code (ECC) has the overhead of 8 bits per 64 bits of data (i.e., 13%), whereas radiation-hardened cells can have an area overhead of 30-100% depending on the aggressiveness of the technique [54]. Third, ECC can correct hard error or parametric faults, which typically limit the yield of SRAMs.

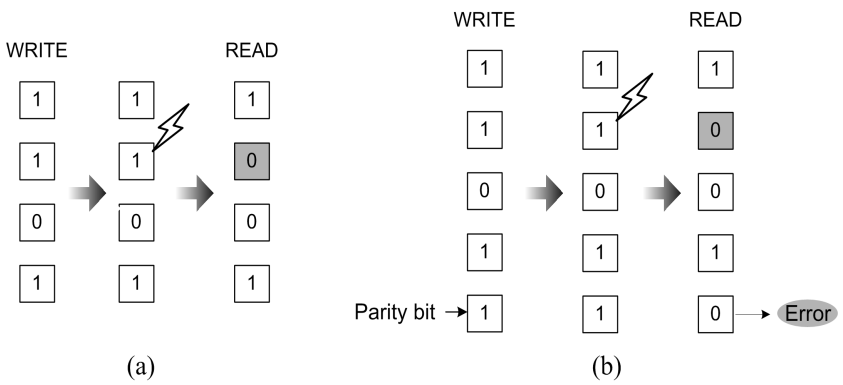


Figure 3.8: Soft error event in a) an unprotected memory word and b) a parity protected memory word.

The cheapest architecture level solution is to add a parity bit to each memory word during a write operation (see Figure 3.8). The parity bit is typically the XOR of all the data bits in the word. If a particle strike alters the state of any bit of the word, the error is discovered by checking the parity bit during each READ operation. Since this scheme detects but does not correct the error, it must be coupled with a technique for error recovery. This limits the usefulness of the scheme as it increases the complexity of the system design. However, in some situations, the error recovery using parity can be very simple. For example, if the memory is an instruction cache, all the data in the cache can also be found in the main memory. Thus, the erroneous data can be recovered from the main memory whenever a parity error is detected. However, in situations where error recovery is more complex, ECC is preferable.

The basic concept of an ECC is to add a number of parity bits or check-bits with the data bits in order to locate and correct a given number of bit errors. The word containing the check-bits and the data bits is referred to as a check-word or codeword. The number of check-bits in the codeword is a function of the number of data bits and the number of correctable errors. If k check-bits are used for n data bits, a single error can occur in any of $n+k$ locations in the codeword. These locations plus a *no error* situation give a total of $n+k+1$ possible ways of having at most one error. To distinctively identify all these possibilities, the number of check-bits must satisfy following relationship [55]:

$$2^k \geq n + k + 1 \tag{3.3}$$

Similarly, if a family of codewords is chosen such that the minimum distance, d, i.e.,

the number of bit locations in which the codewords differ, is given by $d \geq 2y + 1$, then the codewords are said to be valid for correcting y bit errors. Thus, for single bit error correction, $d = 3$. This distance approach is 'geometric' while the above error location argument is 'algebraic'. This type of single error correction method is called the *Hamming Code* and d is referred to as the *Hamming Distance*. For a 4-bit data word, 3 check-bits are needed for Hamming code. The resulting codeword is 7 bit long and is referred to as a (7, 4) Hamming code. In order to explain the conventional ECC operation in the SRAM, we discuss the Hamming code in detail. Principles and operation of other codes, like SECDED or double error correction triple error detection (DECTED), can be understood from the discussion.

The check-bits in the Hamming code are all even parity and are calculated using modulo-2 addition. Each check-bit is computed from a subset of the data bits. Let us assume that the data bits are described by the following 1×4 matrix for a (7, 4) Hamming code:

$$D = [d_1 \ d_2 \ d_3 \ d_4] \tag{3.4}$$

A 4×7 generator matrix can be defined to translate the 4 bit data into the 7 bit codeword of the form $[d_4 \ d_3 \ d_2 \ p_3 \ d_1 \ p_2 \ p_1]$, where p_1, p_2, and p_3 are the check-bits. The generator matrix is given by:

$$G = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}. \tag{3.5}$$

Then the codeword matrix is expressed as:

$$C = D \times G = [d_1 \ d_2 \ d_3 \ d_4] \times \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [d_4 \ d_3 \ d_2 \ p_3 \ d_1 \ p_2 \ p_1], \tag{3.6}$$

where

$$p_1 = d_1 + d_2 + d_4, \tag{3.7}$$

$$p_2 = d_1 + d_3 + d_4, \tag{3.8}$$

$$p_3 = d_2 + d_3 + d_4. \tag{3.9}$$

Here, the '+' sign refers to modulo-2 addition, whose electronic equivalent is the XOR operation. It should be noted that the columns in G can be arranged in any other order. This would just change the positions of data bits and check-bits in the codeword.

In a write operation to the memory, the check-bits given by (3.7) through (3.9) are stored along with the data bits. In a read operation, both the check-bits and data bits are read, i.e., the codeword is read and checked for error. In this case a 3×7 parity check matrix, H, is defined to compute the syndrome S, which is described as

$$S = H \times C^T, \tag{3.10}$$

where

$$H = \begin{array}{c} \begin{matrix} d_4 & d_3 & d_2 & p_3 & d_1 & p_2 & p_1 \end{matrix} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \end{array} \tag{3.11}$$

and C^T is the transpose of C given by (3.6). Substituting C^T in (3.10) yields

$$S = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} d_4 \\ d_3 \\ d_2 \\ p_3 \\ d_1 \\ p_2 \\ p_1 \end{bmatrix} = \begin{bmatrix} S_3 \\ S_2 \\ S_1 \end{bmatrix} = \begin{bmatrix} d_4 + d_3 + d_2 + p_3 \\ d_4 + d_3 + d_1 + p_2 \\ d_4 + d_2 + d_1 + p_1 \end{bmatrix} \tag{3.12}$$

If all the rows or bits in the syndrome are zero, then no error has occurred. Conversely, any non zero value of the syndrome indicates an error and gives the binary bit position of the error. For example, if $[S_3\ S_2\ S_1] = [0\ 1\ 1]$, then an error has occurred at bit position 3 in the checkword, i. e., d_1 has flipped. Subsequently, the error is corrected by flipping d_1 back.

For a SECDED code, the syndrome has an additional bit corresponding to an extra check-bit, p_T. p_T is generated by modulo-2 addition of all data bits and check bits. Thus, for the above example, p_T is given by

$$p_T = d_1 + d_2 + d_3 + d_4 + p_1 + p_2 + p_3 \tag{3.13}$$

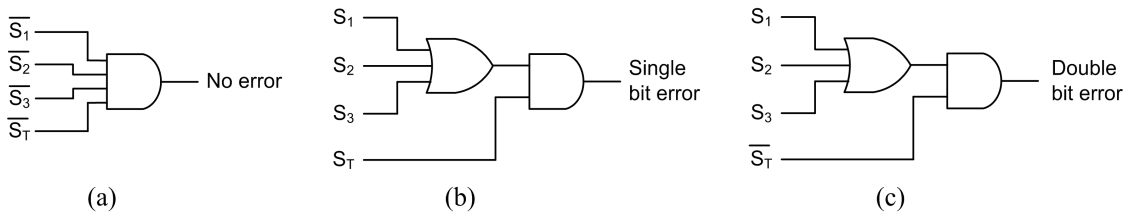


Figure 3.9: Error signal generation from syndrome bits in SECDED code: a) no error, b) single bit error, and c) double bit error.

This extra check-bit generates an additional syndrome bit, S_T, which is only used for error detection. Error bit location is given by the other syndrome bits (same as the Hamming code). Figure 3.9 shows the error detection logic for SECDED code. It should be noted that due to the extra check-bit, the number of check-bits, k', in the SECDED code is given by the following relationship [56]:

$$2^{k'-1} \geq n + k'. \tag{3.14}$$

Thus, depending on the coding algorithm, an ECC-protected SRAM will have a different number of check-bits and different error correction/detection logic. However, in general, an ECC-protected SRAM consists of check-bit memory, check-bit generator (XOR tree), syndrome decoder, and an error corrector as shown in Figure 3.10(a). In a write operation to the SRAM, when data are written into the data memory, check-bits are also generated and written into the check-bit memory. In a read operation, both data bits and check-bits are read out from corresponding memories. Check-bits are regenerated and bitwise XORed with stored check-bits to generate the syndrome bits. Syndrome bits are all zero if there is no error. Otherwise, syndrome bits represent erroneous data bits location, which is decoded using a binary decoder. The number of outputs of the decoder is the same as the number of data bits. The error corrector performs bitwise XOR operation between the decoder outputs and corresponding data bits to correct the erroneous bit.

The generation of check-bits and the possible error correction/detection operation incur extra delay during the write and read operations, respectively. As a result, the latency increases in ECC-protected SRAMs. In addition, storing check-bits causes additional area and power penalty. The penalty is high for shorter words and moderate to low for larger words, as shown in Figure 3.10(b) for the SECDED code. On the other hand, the delay penalty increases for larger words because of the increased logic depth in

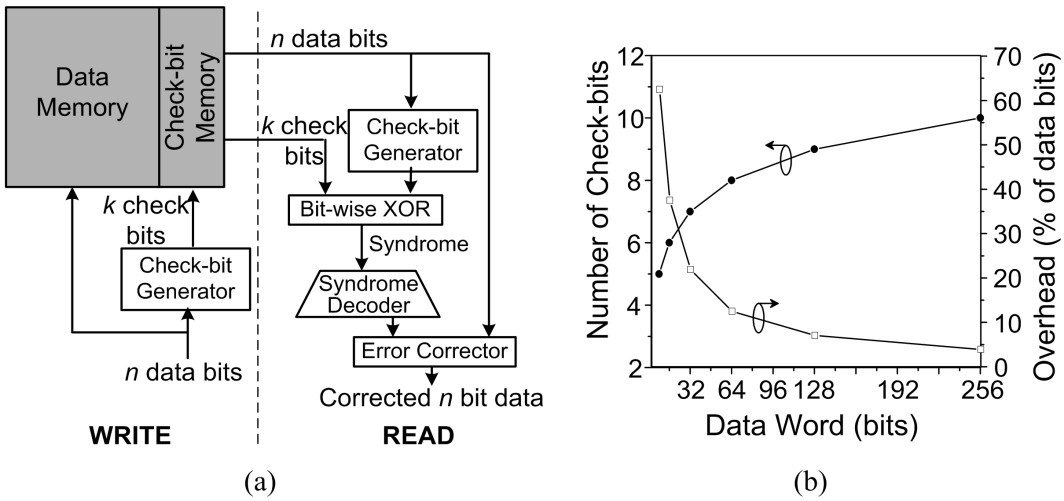


Figure 3.10: a) Block diagram of ECC operation on an SRAM and b) ECC checkbit overhead in SECDED code.

the check-bit generator. A trade-off thus exists between the area and delay penalties in ECC-protected SRAMs. Efficient techniques are, therefore, essential to reduce the data latency while performing the error correction operation to mitigate soft errors.

3.3 Summary

In this chapter we have presented the existing soft error critical charge modeling approaches and identified their limitations. In addition, we have reviewed different process, circuit, and architecture-level soft error mitigation techniques. The circuit and architecture-level techniques are particularly attractive since they do not need process modifications. However, architecture-level techniques, such as ECC, is more attractive for soft error mitigation due to significantly lower area overhead. Therefore, we have discussed the theoretical and implementation aspects of ECC in SRAMs. Since ECC incurs area and delay penalty, efficient techniques are required to minimize the penalty while protecting the SRAM from soft errors.

Chapter 4

Modeling of the Soft Error Critical Charge

This chapter presents the details of a comprehensive model for the soft error critical charge in SRAMs. The model is validated by SPICE simulations and neutron radiation test.

The vulnerability of SRAM to soft errors is typically assessed with the help of its critical charge, Q_{crit} [40]. Q_{crit} is the minimum amount of charge that can flip the data-bit stored in an SRAM cell. It exhibits an exponential relationship with the soft error rate (SER) [57]. For a linear decrease in Q_{crit}, the SER increases exponentially. Accordingly, Q_{crit} should be as high as possible in order to limit the SER. However, different low-power design approaches (e.g., supply voltage reduction, gated grounding, etc.) significantly reduce Q_{crit}, as we have seen in Chapter 2. In addition, the actual value of Q_{crit} in manufactured SRAMs can deviate from the designed value due to process-induced variations in the transistor parameters. Therefore, a tool that can model Q_{crit} and describe its sensitivity to different design approaches and process non-idealities is essential in order to design SRAMs with given SER requirements. In this chapter, we propose a comprehensive Q_{crit} model that can reliably serve this purpose.

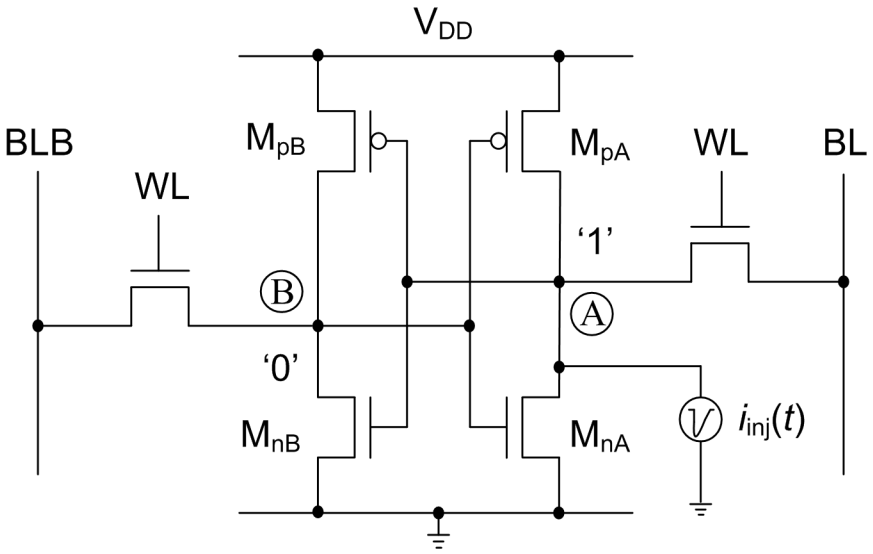


Figure 4.1: 6T SRAM cell with an exponential current source to mimic a particle strike at node A.

4.1 Proposed Critical Charge Model

A typical six-transistor (6T) SRAM cell consists of two cross-coupled inverters that store two complementary logic values '1' and '0' at their outputs (see Figure 4.1). We denote these two nodes by 'A' and 'B'. Nodes A and B are accessed from the bit lines (BL and BLB) through two NMOS transistors. We assume that nodes A and B store logic '1' and logic '0', respectively, so that transistors M_{nA} and M_{pB} are 'OFF' while M_{pA} and M_{nB} are 'ON'. The load PMOS transistors (M_{pA} and M_{pB}) have a smaller aspect ratio (W/L) than the driver NMOS transistors (M_{nA} and M_{nB}) to ensure reliable write and nondestructive read operations. In addition, the mobility of a PMOS transistor is much less, which makes the ON conductance of M_{pA} and M_{pB} lower than M_{nA} and M_{nB}. As a result, node A storing logic '1' becomes weaker than node B in terms of noise tolerance. This implies that node A has a smaller critical charge, which is also evident from the SPICE simulations shown in Figure 4.2. Therefore, for a given cell voltage, we use the critical charge of node A as the Q_{crit} of the SRAM cell. In addition, like previous reports [6],[13], we exclude access transistors from Q_{crit} analysis since the cell is most likely to be in the un-accessed or retention mode when a particle strikes. However, we consider the effect of the width of the access transistor as a component of the capacitance at node

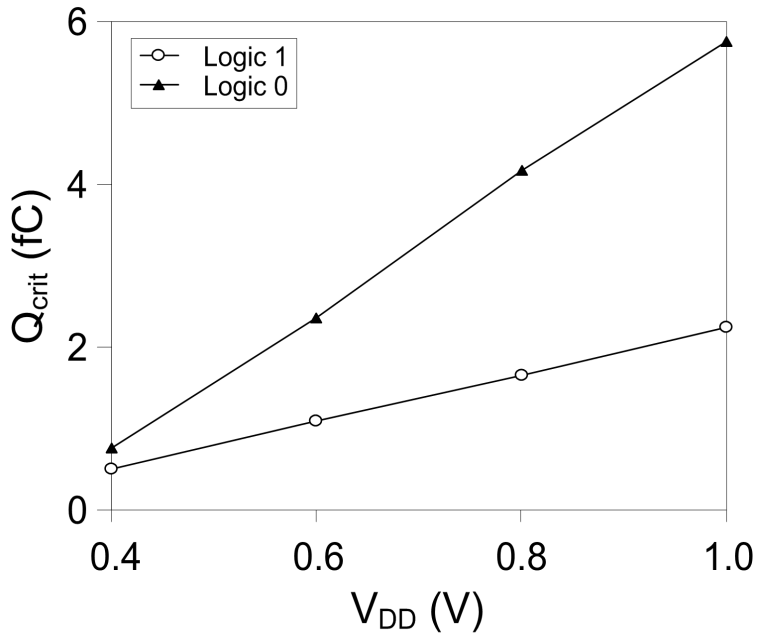


Figure 4.2: Critical charge as a function of cell supply voltage for logic '1' and logic '0' nodes in an SRAM cell.

A. The length of the access transistor does not come into play as the access transistor is 'OFF'. If a particle-induced current has an extremely large amplitude that it pulls down the node A voltage below 0 V, it can turn on the access transistor and cause an additional restoring current from BL. However, the cell flipping time in such a case will be very small, which will undermine the effect of the additional restoring current. Furthermore, as shown later in this section, the cell flips even when the node A voltage is pulled down to a positive value, which does not turn on the access transistor. Thus, ignoring the access transistors does not sacrifice the accuracy of our analysis.

In order to determine Q_{crit}, we now consider the dynamic response of the cell to a transient noise current that mimics a particle strike. Since the cell is a non-linear system, its dynamic behavior can be understood by state-space analysis. Here, two node voltages $\overrightarrow{V_A}$ and $\overrightarrow{V_A}$ constitute the state vectors, $\overrightarrow{V} = (V_A, V_B)$ and current equations at these nodes constitute the state equations [43], [58]. The state equations have three DC solution points - two stable points associated with logic states '1' and '0', and one metastable point. In fact, these solutions are the intersection points of the back-to-back connected inverters' voltage transfer curves, which constitute the commonly known "butterfly curve" of the

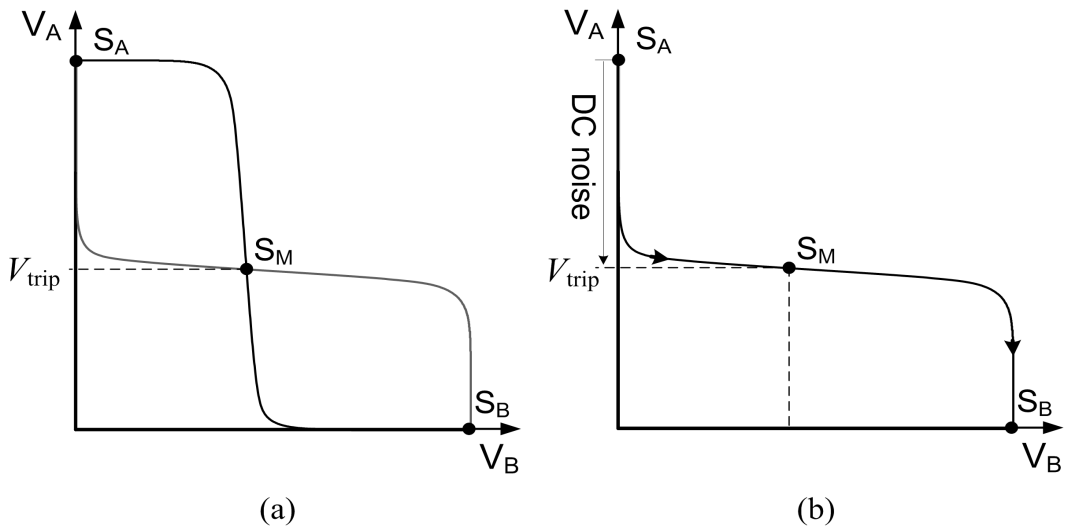
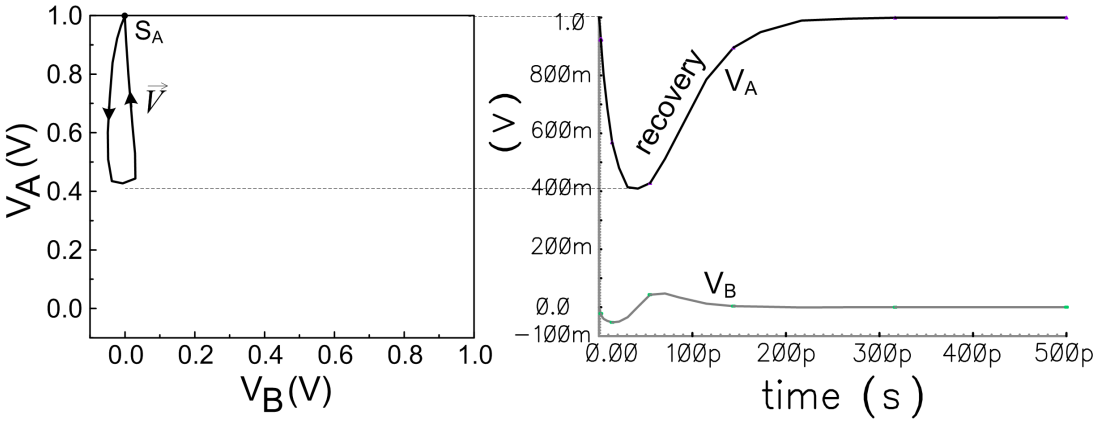
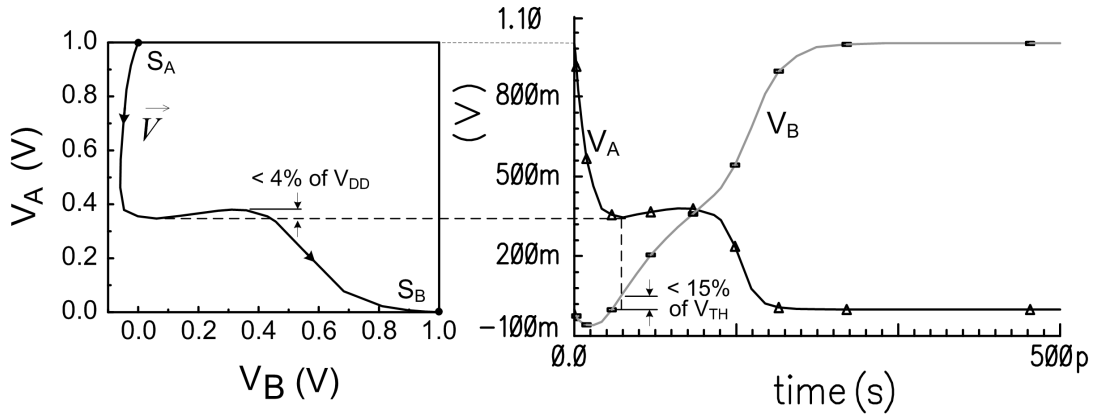


Figure 4.3: a) State-space representation of SRAM cell characteristics and b) trajectory of state vector for a DC noise voltage at node A.

SRAM cell. Figure 4.3(a) shows the butterfly curve in a two-dimensional state space where the DC solution points are labeled as S_A, S_B, and S_M. Here, the metastable point corresponds to $V_A = V_{trip}$, which marks the boundary between logic '1' and logic '0' at node A for nominal V_{DD}. When a DC noise at node A pulls down V_A below V_{trip} and thus drives the state vector beyond S_M, the inherent positive feedback of the cell comes into play. M_{pB} becomes stronger than M_{nB} and raises V_B, which is also the gate voltage of M_{nA}. As a result, M_{nA} starts to conduct and further pulls down V_A. Eventually, V_B rises to V_{DD} and V_A falls to 0, thus flipping the logic states of the cell. The corresponding trajectory of $\overrightarrow{V}$ is shown in Figure 4.3(b). In contrast, when a transient noise, such as a particle-induced current perturbs V_A, the state of the cell may or may not flip depending on the magnitude and duration of the current pulse. Figure 4.4 shows the trajectories of two current pulses having the same magnitude but different durations. The trajectories are obtained by plotting V_A as a function of V_B at various times. The duration of the shorter current pulse is not long enough to initiate the positive feedback of the cell. As a result, $\overrightarrow{V}$ momentarily moves away from S_A but finally returns to S_A - thus recovering from the transient (Figure 4.4(a)). On the other hand, the duration of the longer pulse is large enough to initiate the positive feedback that drives to S_B, thus flipping the cell (Figure 4.4(b)). We need to determine the minimum magnitude and duration of a noise current pulse in order to calculate Q_{crit}.



(a)



(b)

Figure 4.4: a) State-space and time domain plots of cell node voltages for a non state-flipping case and b) state-space and time domain plots of cell node voltages for a state-flipping case.

Conventionally, an exponential current pulse of the form:

$$i_{inj}(t) = \frac{Q}{\tau_f - \tau_r}\left(e^{-t/\tau_f} - e^{-t/\tau_r}\right) \tag{4.1}$$

is used to determine Q_{crit} [9]. Here, Q is the total charge deposited by the current pulse, τ_r is the rise time constant, and τ_f is the fall time constant. Typically, a particle-induced current pulse has a short ($\sim$ 10 ps) rise time and a longer ($\sim$ 200 ps) fall time. Now, the transient voltage at node A can be described as:

$$C_N \frac{dv_A}{dt} = i_{restore}(t) - i_{inj}(t) - i_d(t) \tag{4.2}$$

where C_N is the node capacitance, $i_{restore}(t)$ is the restoring current supplied by M_{pA},

and $i_d(t)$ is the drain current of M_{nA}. C_N can be calculated by adding the parasitic capacitances at node A:

$$C = 2\left(C_{gdp} + C_{gdn}\right) + C_{dbp} + C_{dbn} + C_{dsp} + C_{dsn} + C_{gp} + C_{gn} \tag{4.3}$$

Since the gate-to-source voltage (V_{GS}) of M_{nA} is 0, we can ignore $i_d(t)$. Conversely, the V_{GS} of M_{pA} is -V_{DD}, driving it in the linear region. Therefore, we can replace M_{pA} by a resistor R_p and use (4.1) to express (4.2) as

$$C_N \frac{dv_A}{dt} = \frac{V_{DD} - v_A}{R_p} - \frac{Q}{\tau_f - \tau_r}\left(e^{-t/\tau_f} - e^{-t/\tau_r}\right) \tag{4.4}$$

Equation (4.4) can be solved (see Appendix A)with the initial condition $v_A(0) = V_{DD}$ to yield

$$v_A(t) = V_{DD} - \frac{QR_p}{\tau_f - \tau_r}\left\{ \begin{array}{l} \frac{\tau_f}{\tau_f - R_pC_N}\left(e^{-t/\tau_f} - e^{-t/R_pC_N}\right) \\ -\frac{\tau_r}{\tau_r - R_pC_N}\left(e^{-t/\tau_r} - e^{-t/R_pC_N}\right) \end{array} \right\} \tag{4.5}$$

Equation (4.5) describes $v_A(t)$ for a non state-flipping case when $v_A(t)$ goes through a voltage minimum and finally returns to V_{DD} (see Figure 4.4(a)). However, (4.5) can be used to determine the limiting case when $i_{inj}(t)$ is just strong enough to flip the node voltages. In order to see the characteristics of the limiting case, we iteratively increase Q by a small amount ($\sim$ 0.001 fC) in SPICE until the node voltages, $v_A(t)$ and $v_B(t)$, flip. We find that for such a case, once $v_A(t)$ reaches a voltage minimum, V_{min}, it stays around V_{min} (with deviation $< 4\%$ of V_{DD} as shown in Figure 4.4(b)) until $v_B(t)$ rises to V_{min}. Eventually, $v_A(t)$ drops to 0 and $v_B(t)$ reaches V_{DD} as shown in Figure 4.5.

In addition, we see that $v_B(t)$ stays either below 0 V or a little higher ($< 15\%$ of V_{TH}) than 0V for the time interval over which $v_A(t)$ approaches V_{min}. Such a variation in $v_B(t)$ does not effectively cause any conduction through M_{nA}. These observations have been confirmed through extensive SPICE simulation with a variety of Q, τ_r, and τ_f combinations. Accordingly, we generalize the state-flipping process in the following way into two distinct time intervals (see Figure 4.5):

$$\left.\begin{array}{l} V_{DD} \geq v_A(t) \geq V_{\min} \\ v_B(t) \approx 0 \end{array}\right\} for \;\; 0 \leq t \leq T_1 \tag{4.6}$$

$$\left.\begin{array}{l} v_A(t) \approx V_{\min} \\ 0 \leq v_B(t) \leq V_{\min} \end{array}\right\} for \;\; T_1 \leq t \leq T_{crit} \tag{4.7}$$

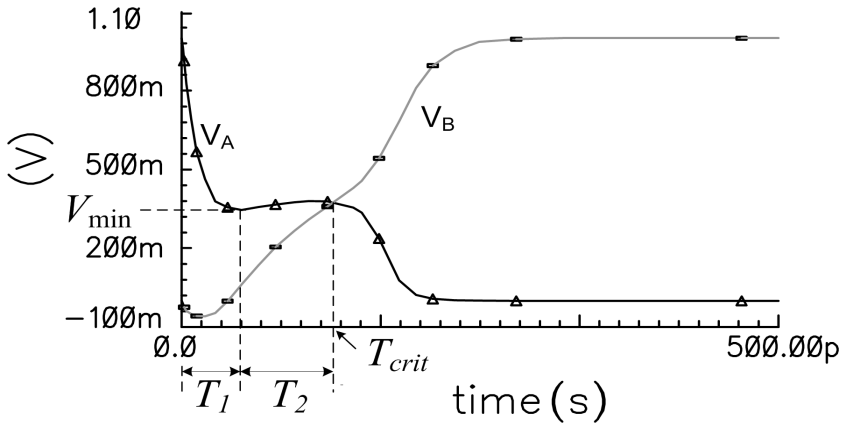


Figure 4.5: Node voltage transients for a state-flipping particle strike at node A.

Equations (4.6) and (4.7) allow us to decouple the cross-coupled inverters of the SRAM cell. Once decoupled, these inverters form a linear system given that the transfer characteristics of the constituent transistors are also linear. This decoupling technique is simpler than the one reported in [43] since the former does not depend on the saturation voltage, V_{dsat}, of the 'ON' transistor connected to the struck node. Reliable extraction of V_{dsat} is difficult when different transistor parameters vary due to process non-idealities.

Now, in order to find T_1 and V_{min} from (4.5), we do an approximation to simplify the mathematical operations. Since $\tau_r \ll \tau_f$, we assume that $\tau_r \approx 0$ and replace τ_f with τ so that (4.1) and (4.5) reduce to

$$i_{inj}(t) = \frac{Q}{\tau} e^{-t/\tau} \tag{4.8}$$

and

$$v_A(t) = V_{DD} - \frac{QR_p}{\tau - R_pC_N}\left(e^{-t/\tau} - e^{-t/R_pC_N}\right), \tag{4.9}$$

respectively. Here, Q is the same total charge as that deposited by the double exponent current pulse given by (4.1). We have confirmed the validity of above assumption and hence (4.9) through SPICE simulations with typical values of τ_r (<10 ps) and τ_f (>50 ps). Differentiating (4.9) and equating the result to 0, we get after mathematical simplification,

$$T_1 = \frac{\tau R_pC_N}{\tau - R_pC_N} \ln \frac{\tau}{R_pC_N}. \tag{4.10}$$

Substituting (4.10) into (4.9) yields

$$V_{\min} = V_{DD} - \frac{QR_p}{\tau}\left(\frac{R_pC_N}{\tau}\right)^{\frac{R_pC_N}{\tau - R_pC_N}}. \tag{4.11}$$

For a state flipping case, $v_A(t)$ stays at V_{min} while $v_B(t)$ rises. $v_B(t)$ can be expressed as

$$C_N \frac{dv_B}{dt} = i_p(t) - i_n(t) \text{ for } \quad T_1 \leq t \leq T_{crit}, \tag{4.12}$$

where $i_p(t)$ and $i_n(t)$ are the currents through M_{pB}, and M_{nB}, respectively. Considering M_{pB} in the saturation and M_{nB} in the linear region, we can write from (4.12)

$$C_N \frac{dv_B}{dt} = g_{mp} \left(V_{DD} - V_{\min} - |V_{THp}| \right) - \frac{v_B}{R_n}, \tag{4.13}$$

where g_{mp} and V_{THp} are the transconductance and threshold voltage of the PMOS M_{pB}, respectively, and R_n is the linear region resistance of the NMOS M_{nB}. Like [43], here we use the "linear gate model" of a transistor in order to linearize its transfer characteristics and describe the saturation current as

$$I_{sat} = g_m \left(V_{GS} - V_{TH} \right) \tag{4.14}$$

This assumption has been validated by observing the DC transfer curve of each individual transistor. In fact, as transistor dimension shrinks, its saturation current becomes a linear function instead of a quadratic function of gate overdrive voltage [1]. Equation (4.13) can now be solved with boundary conditions $v_B(T_1) = 0$ and $v_B(T_{crit}) = V_{\min}$ to yield

$$T_2 = -R_n C_N \ln \left(1 - \frac{V_{\min}}{R_n g_{mp} \left(V_{DD} - V_{\min} - |V_{THp}| \right)} \right), \tag{4.15}$$

where $T_2 = T_{crit} - T_1$. It is evident from (4.15) that in order to have $T_2 > 0$ so that the positive feedback between nodes A and B can occur, the following condition must be met:

$$V_{\min} < \frac{R_n g_{mp} \left(V_{DD} - |V_{THp}| \right)}{1 + R_n g_{mp}} \tag{4.16}$$

or

$$V_{DD} - V_{\min} > \frac{V_{DD} + |V_{THp}| R_n g_{mp}}{1 + R_n g_{mp}} \tag{4.17}$$

Using (4.11), (4.17) can be expressed as:

$$\frac{Q R_p}{\tau} \left(\frac{R_p C_N}{\tau} \right)^{\frac{R_p C_N}{\tau - R_p C_N}} > \frac{V_{DD} + |V_{THp}| R_n g_{mp}}{1 + R_n g_{mp}} \tag{4.18}$$

Equation (4.18) defines the necessary condition for a current pulse to be able to flip the logic states in a given SRAM cell. Thus, we can obtain a $Q - \tau$ diagram, as shown in Figure 4.6, and quickly determine whether a current pulse be detrimental to an SRAM

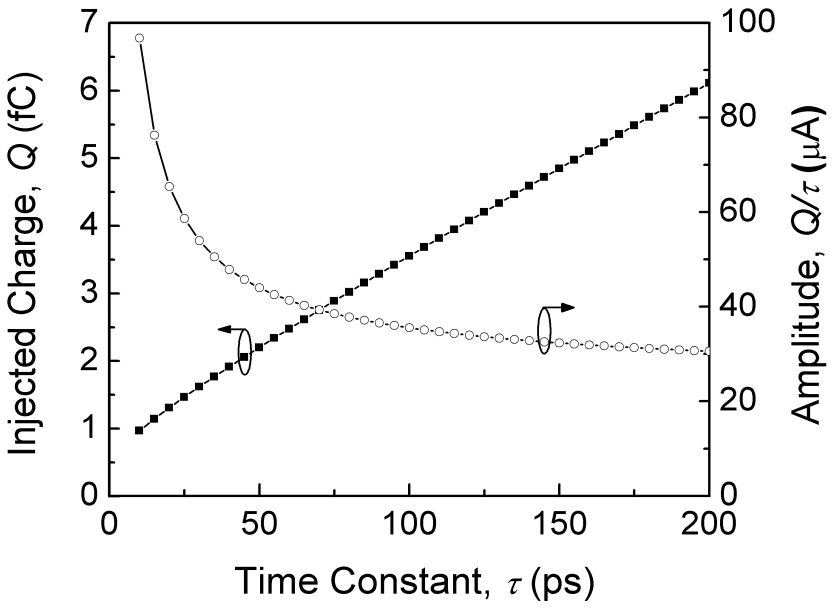


Figure 4.6: Total injected charge necessary to flip the logic states and the amplitude of injected current as a function of the time constant.

cell. Figure 4.6 also shows Q/τ plot, which implies the effective amplitude of the injected current as a function of τ. As evident from the asymptotic behaviour of this plot, the cell can tolerate a larger amplitude current when τ is smaller. This is in line with previous reports such as [43].

Once $v_B(t)$ equals $v_A(t)$ at $t = T_{crit}$, i.e. $v_B(T_{crit}) = v_A(T_{crit}) = V_{min}$ holds, we assume that the positive feedback of the cell becomes strong enough to continue flipping the state of the cell. There is no need for additional charge injection to flip the cell. Accordingly, we define the critical charge as the charge injected by $i_{inj}(t)$ up to $t = T_{crit}$ as shown in Figure 4.7. Thus, the critical charge is given by

$$Q_{crit} = \int_0^{T_{crit}} i_{inj}(t)dt. \tag{4.19}$$

Using (4.8), we can express (4.19) as

$$Q_{crit} = Q\left(1 - e^{-T_{crit}/\tau}\right). \tag{4.20}$$

Thus, by quantifying the transient responses of the storage node voltages to an exponential noise current, we obtain a compact Q_{crit} model for the SRAM cell. With reference to Figure 4.5, the model can be summarized as follows:

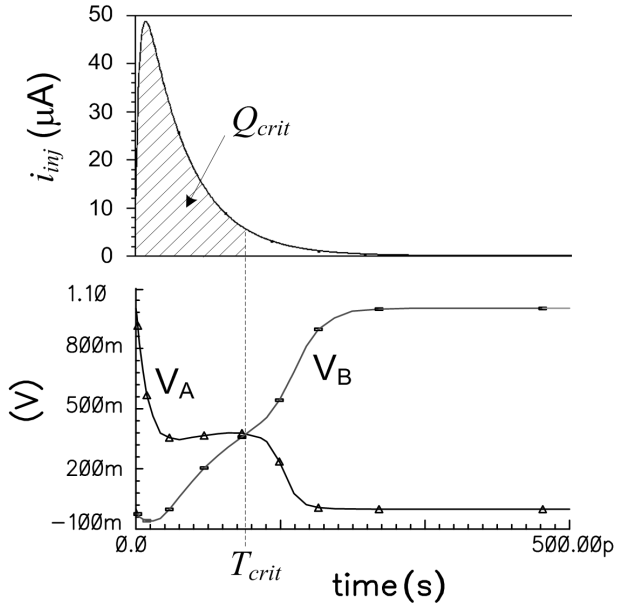


Figure 4.7: Graphical definition of critical charge for the proposed model.

$$\left.\begin{aligned} V_{\min} &= V_{DD} - \tfrac{QR_p}{\tau}\left(\tfrac{R_pC_N}{\tau}\right)^{\frac{R_pC_N}{\tau-R_pC_N}} \\ T_1 &= \tfrac{\tau R_pC_N}{\tau-R_pC_N}\ln\tfrac{\tau}{R_pC_N} \\ T_2 &= -R_nC_N\ln\left(1-\tfrac{V_{\min}}{R_ng_{mp}\left(V_{DD}-V_{\min}-|V_{THp}|\right)}\right) \\ T_{crit} &= T_1+T_2 \\ Q_{crit} &= Q\left(1-e^{-T_{crit}/\tau}\right) \end{aligned}\right\} \tag{4.21}$$

4.2 Model Verification

In order to verify the accuracy of the proposed model, we compare its predictions with SPICE simulations and radiation test results. We use a commercially available 90nm CMOS process available through CMC for the simulation and chip fabrication.

4.2.1 Verification by SPICE

For calculating Q_{crit} using SPICE, we simulate a 6T SRAM cell in the Cadence environment at 27 °C. The cell uses high V_{TH} transistors in order to minimize the leakage current. The transistor sizes are optimized to ensure read stability with minimum area and acceptable speed (∼350 MHz). The nominal supply voltage (V_{DD}) of the cell is 1V

and the node capacitance C_N, which has been extracted from layout, is 0.876 fF. We inject at node A an exponential current of the form (4.1) with $\tau_r = 1$ ps and $\tau_f = 50$ ps. We incrementally increase the current amplitude until the cell flips. Then we compute Q_{crit} by integrating the injected current up to the time when the cell node voltages cross each other.

For calculating Q_{crit} using the proposed model, we first extract transistor parameters, such as, V_{TH}, R_n, R_p, and g_{mp}. We use the DC current-voltage characteristics of individual NMOS and PMOS transistors for this purpose. Then we substitute the extracted values of these parameters in (4.21) to calculate Q_{crit}. While extracting transistor parameters, we notice that the linear region resistance and tranconductance exhibit following characteristics, respectively:

$$R \propto \frac{L}{W} \left(V_{GS} - V_{TH}\right)^{-1} \tag{4.22}$$

and

$$g_m \propto \frac{W}{L} \left(V_{GS} - V_{TH}\right). \tag{4.23}$$

Following the above methodologies, we determine Q_{crit} using SPICE simulation, the model reported in [40], and the proposed model for a particle strike at the logic '1' node. For the model of [40], we use (3.1) and set T_F equal to T_{crit} and I_{DP} to the maximum current through M_{pA} (i.e., for $V_{DS} = V_{DD} - V_{min}$). The results are shown in Figure 4.8 for varying V_{DD}. Clearly, compared to [40], the proposed model is in better agreement with SPICE. The large discrepancy between [40] and SPICE stems from overestimation of both the capacitance ($C_N V_{DD}$) and conductance ($I_{DP} T_F$) components of Q_{crit}. In particular, use of a constant restoring current (I_{DP}) throughout T_F increases the discrepancy with increasing V_{DD}. For the minimum discrepancy (i.e., for V_{DD}=0.6 V), the contribution of $I_{DP} T_F$ term in Q_{crit} is about 44%. This underscores the need for accurate modeling of the restoring current. In fact, the contribution of the restoring current increases as the duration of injected current pulse increases. Accordingly, the proposed model replaces M_{pA} with R_p to more realistically describe the time-varying restoring current. In addition, the model quantifies the dynamic behaviour of storage nodes to precisely estimate Q_{crit}, thus showing a maximum discrepancy of only 2.8% with SPICE in Figure 4.8. This small discrepancy, however, can be attributed to two factors. First, the model ignores the effect

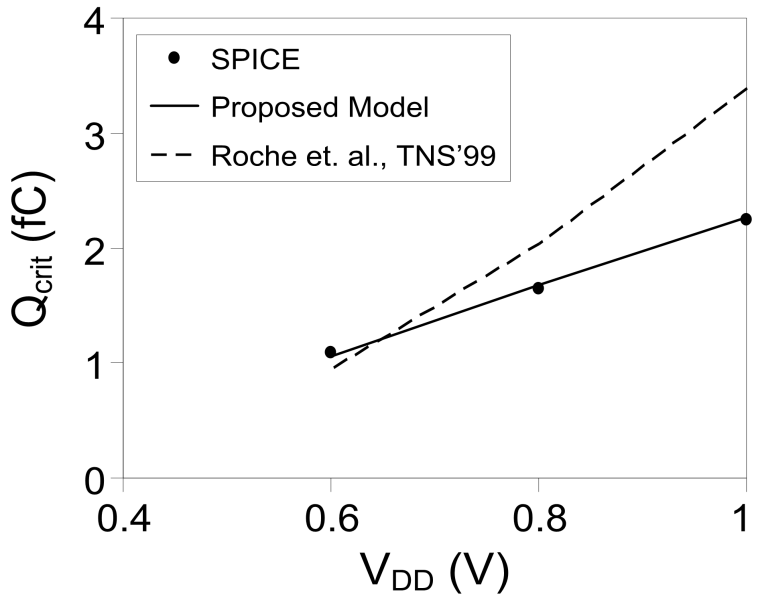


Figure 4.8: Comparison of the proposed model with SPICE when calculating the critical charge at different cell supply voltage.

of the subthreshold current of M_{nA} in the decoupling assumption stated in (4.7), whereas the current can increase as $v_B(t)$ rises over the interval T_2. Second, the model employs the resistor R_p to describe the current through M_{pA}, whereas M_{pA} can deviate from the linear current-voltage characteristics over the interval $0 \rightarrow T_{crit}$. Since the impact of these two factors on the dynamics of the cell (see Figure 4.5) and hence on Q_{crit} is negligible, the assumptions of the proposed model are not violated.

In order to demonstrate the effectiveness of the model further, we compare it with [43] for a particle strike at the logic '0' node, i.e., at node B. We first derive the equations similar to (4.21) for a noise current injected into node B and calculate the corresponding Q_{crit}, as described in Appendix B. Then we determine Q_{crit} according to [43]. Since [43] uses a rectangular current pulse to determine Q_{crit}, we need to map the exponential pulse to an equivalent rectangular pulse. Accordingly, we need to make sure that i) the total charge deposited by the exponential pulse and the rectangular pulse is the same, and ii) both pulses generate similar effects on $v_B(t)$. However, in [43], the rectangular pulse's minimum or critical width for causing a state flip is reported as

$$T_{crit} = -R_n C_N \ln\left[1 - V_{dsat}/\left(I_n R_n\right)\right] - C_N/g_{mn} \ln\left[1 - g_{mn}\left(V_{DD} - V_{THn}\right)/I_n\right] \quad (4.24)$$

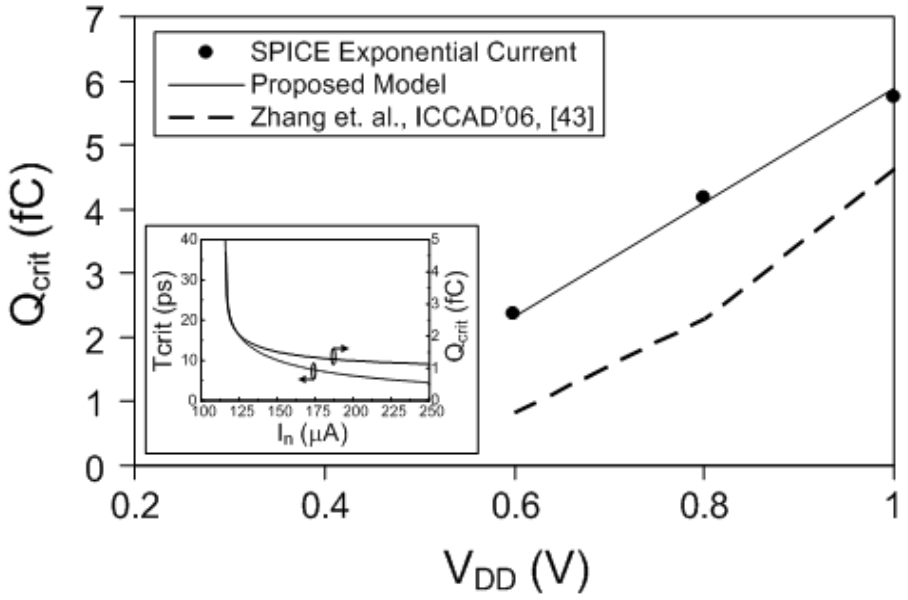


Figure 4.9: Critical charge for node B as a function of cell supply voltage.

Here, R_n , g_{mn}, and V_{THn} are the linear region resistance, transconductance, and threshold voltage of the driver NMOS transistor, respectively, and I_n is the amplitude of the current pulse. The first term on the right hand side of (4.24) is finite only when

$$I_n > V_{dsat}/R_n. \tag{4.25}$$

On the other hand, the rectangular pulse can deposit the same total charge, Q, as the exponential current pulse only when

$$T_{crit} = Q/I_n \tag{4.26}$$

Satisfying (4.25) and (4.26) simultaneously is difficult. Therefore, we follow the following approach to calculate Q_{crit}. We set I_n slightly larger than V_{dsat}/R_n. We then substitute I_n in (4.24) to yield T_{crit}. Q_{crit} is then readily found as $I_n T_{crit}$. Q_{crit} values thus obtained for different cell supply voltage are shown in Figure 4.9. As evident from the figure, [43] significantly underestimates Q_{crit}. Had we used larger I_n, T_{crit} and hence Q_{crit} would be even smaller as shown in the inset of Figure 4.9. Figure 4.9 also shows Q_{crit} values obtained using SPICE simulations and the model proposed in this work. Clearly, the proposed model more closely matches SPICE simulations than [43] does. Thus, the proposed model is in good agreement with SPICE in computing Q_{crit} for both the logic '1' and logic '0' nodes while the proposed model is far less time consuming than using

SPICE. Accordingly, the model manifests itself as an attractive and reliable alternative to time consuming and iterative SPICE simulations. The reliability of the model will further be verified by experimental results in the following section and under process variations in the next chapter.

4.2.2 Verification by Radiation Test

In this section, we verify the efficacy of the proposed Q_{crit} model using results of an accelerated neutron radiation test. The test has been carried out on a 64-kb SRAM at TRIUMF, Vancouver, BC. The details of the test are described in Chapter 6. Here, we only consider the number of soft errors measured in the test.

As mentioned earlier, the SER exhibits an exponential relationship with Q_{crit}. Mathematically the relationship is described by the following empirical model [57]:

$$SER \propto FA \exp\left(-\frac{Q_{crit}}{Q_S}\right) \tag{4.27}$$

Here, F is the neutron flux with energy greater than 1MeV, in particles/cm^2-s; A is the sensitive area of the circuit, in cm^2; and Q_S is the charge collection efficiency of the device, in fC. Typically, Q_S depends on the magnitude of the particle-induced charge, substrate doping, carrier mobility, and the voltage of the collecting node and neighbouring nodes. Equation (4.27) can be written as

$$SER = KFA \exp\left(-\frac{Q_{crit}}{Q_S}\right), \tag{4.28}$$

where K is a proportionality constant. If we know the value of K and Q_S, we can use (4.28) to predict the SER for a given SRAM cell using the proposed Q_{crit} model. In order to extract K and Q_S, we take natural logarithm on both sides of (4.28) and rearrange to yield,

$$\ln\left(\frac{SER}{FA}\right) = \left(-\frac{1}{Q_S}\right) Q_{crit} + \ln K. \tag{4.29}$$

If we calculate the left hand side of (4.29) using measured SER, fluence, and cell area for varying supply voltage and plot those values as a function of Q_{crit} extracted from SPICE simulations, the plot should be a straight line. The slope and vertical axis intercept of the straight line will be $-1/Q_S$ and $\ln K$, respectively. Figure 4.10 shows such a plot for supply voltages ranging from 0.9 V to 1.1 V. The extracted value of Q_S

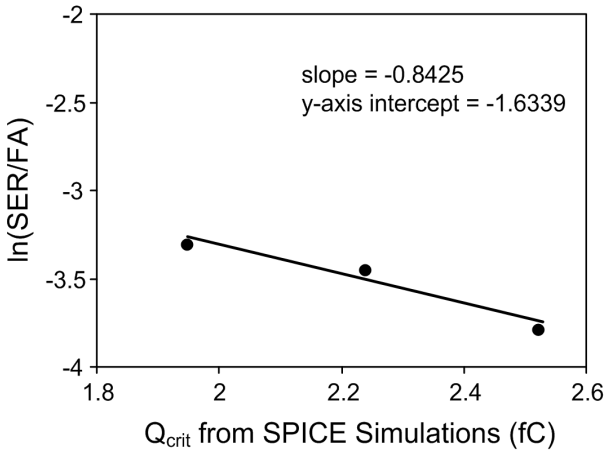


Figure 4.10: Extraction of charge collection efficiency (Q_S).

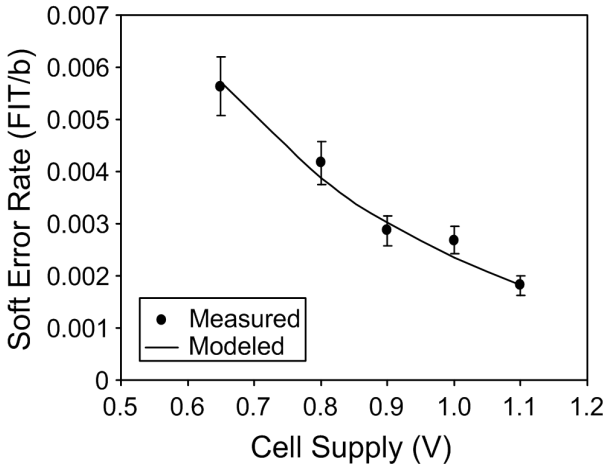


Figure 4.11: Measured and modeled SER as a function of supply voltage. Vertical error bars represent 10% deviation from measured values.

and K from the plot are 1.187 fC and 0.1952 FIT-s/b-n, respectively. Substituting these values in (4.28), we get

$$SER = 0.1952FA \exp\left(-\frac{Q_{crit}}{1.187}\right). \tag{4.30}$$

Equation (4.30) can now be used to verify the proposed Q_{crit} model given by (4.21). We calculate Q_{crit} for different supply voltages using the proposed model and predict the SER using (4.30). Then we compare the predicted SER with the experimentally measured SER. The comparison is shown in Figure 4.11. As evident, the prediction using the proposed Q_{crit} model is in good agreement with the measured SER with a maximum

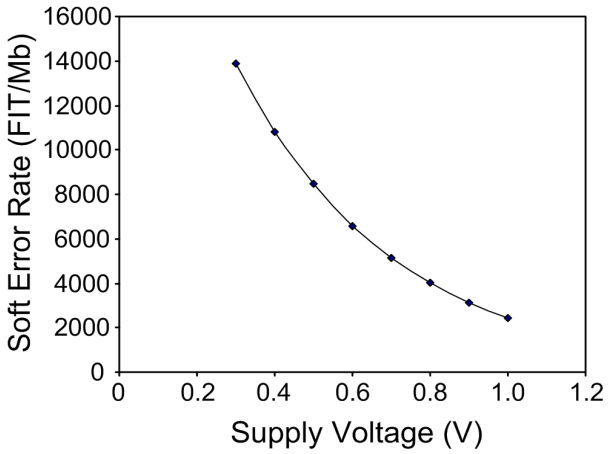


Figure 4.12: Predicted SRAM soft error rate as a function of supply voltage.

discrepancy of less than 10%. In particular, the prediction matches the measured SER in sub-0.9 V supply voltages, which were not used to extract K and Q_S. Thus, the proposed model appears to be an accurate tool for predicting the SER performance of SRAMs.

Figure 4.12 shows the simulated SER per megabits using the proposed Q_{crit} model and (4.30) for a wide range of supply voltage. Clearly, the SER increases significantly at low supply voltages.

4.3 Application of the Model

Since the proposed model accurately describes Q_{crit} as a function of different parameters of transistors constituting the SRAM cell, it can be useful in a variety of ways. For example, the model can be used to readily estimate the impact of process variations on Q_{crit}, the change in Q_{crit} when the supply voltage is reduced or multiple V_{TH} transistors are used to minimize the leakage current, and when a MIM capacitor is added to storage nodes to minimize the SER. In the following, we discuss the efficacy of the model in optimizing the operating voltage of the SRAM and in sizing the MIM capacitor to achieve a given SER. Estimation of the impact of process variations will be discussed in the next chapter.

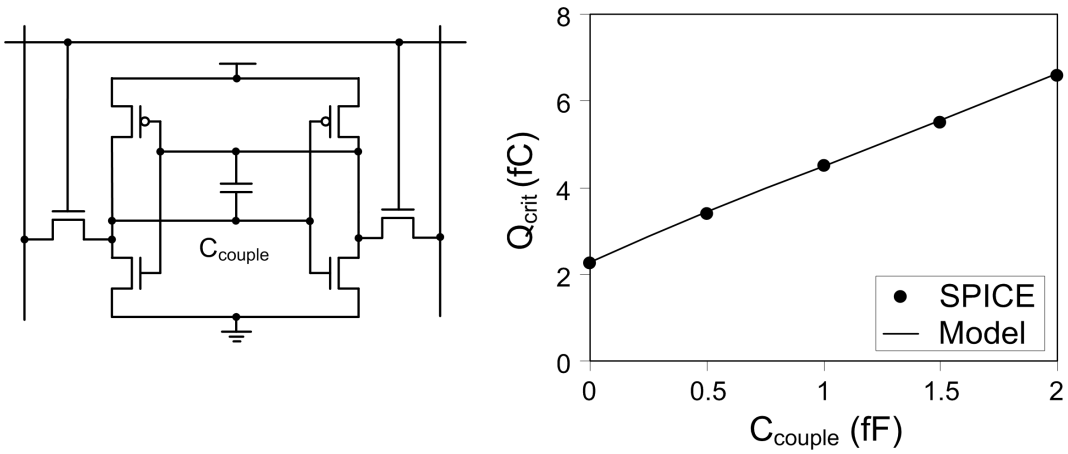


Figure 4.13: a) An SRAM cell with coupling capacitor between storage nodes and b) critical charge as a function of the coupling capacitor.

4.3.1 Optimization of Operating Voltage

In low-power SRAMs, power saving is often achieved by reducing the array supply voltage when the array is not accessed. Since the proposed model can predict how Q_{crit} varies with the cell supply voltage, an optimum supply voltage can be determined so that both low-power operation and soft error resilience can be achieved in a given operating environment. It is well known that a linear decrease in cell operating voltage exponentially reduces the leakage current, which is dominant in nanometric technologies. Designers prefer to lower the cell supply voltage in order to minimize the leakage power consumption. However, as seen in Figure 4.8, decreases in cell operating voltage linearly decrease Q_{crit}, which implies an exponential increase in the SER. Therefore, for a given reduction in cell supply voltage, designers can determine:- i) the increase in soft error vulnerability by calculating the reduction in Q_{crit} using the proposed model, and ii) the saving in power by leakage reduction. Depending on the application, they can find an optimal balance between the two.

4.3.2 Estimation of the MIM Capacitor

One way of boosting the critical charge and reduce the SER is to employ a coupling capacitor, C_{couple}, between the storage nodes (see Figure 4.13(a)). Usually, C_{couple} is stacked on top of the cell as a MIM capacitor in order to avoid any area penalty. The

value of the MIM capacitor is determined by the inter-metal dielectric and the area of the cell, and hence cannot be too large. A typical $1\mu m^2$ C_{couple} can have a value of the order of 1 fF. The proposed model can accurately predict the dependence of Q_{crit} on C_{couple}, which is shown in Figure 4.13(b). The only necessary modification to the model is to change the node capacitance according to the equation:

$$C_{N\_new} = C_N + 2C_{couple}, \tag{4.31}$$

where C_{couple} is doubled to account for the Miller effect.

Since the proposed model can estimate Q_{crit} variation as a function of C_{couple}, it can be reliably used to estimate - i) the amount of improvement in Q_{crit} when a given value of MIM capacitor is added to the cell, or ii) the value of the MIM capacitor needed to achieve a given Q_{crit}. For low-power SRAMs, the latter is more important since it can restore Q_{crit} that may have been reduced due to a reduction in the cell operating voltage.

4.4 Summary

In this chapter, we have presented an analytical model for the soft error critical charge. The model is based on the dynamic response of the SRAM cell to an exponential current pulse, which is the most realistic noise current mimicking a single event transient. The model incorporates both NMOS and PMOS transistor parameters and the temporal profile of the noise current, thus manifesting itself as the most comprehensive and versatile critical charge model reported to date. The critical charge calculated by the model shows less than 5% discrepancy with SPICE simulations while the soft error rate predicted by the model shows less than 10% discrepancy with the soft error rate measured in neutron radiation tests.

Chapter 5

Process Dependence of the SRAM Critical Charge

This chapter investigates the process dependence of the soft error critical charge. In particular, this chapter uses the critical charge model developed in Chapter 4 to quantitatively estimate the impact of process variations on the critical charge.

Like other parameters, such as leakage current, delay, etc., critical charge (Q_{crit}) is also affected by process variations. The impact of process variations on Q_{crit} has traditionally been investigated using Monte Carlo simulations in an SPICE environment and modeled using empirical polynomial equations [59]. However, such simulations are time consuming since every iteration requires calculation of Q_{crit}, which itself requires an iterative injection of current onto the storage node. In addition, while the results of the Monte Carlo simulation or the empirical models show the spread of Q_{crit}, they fail to provide the designer with information concerning the relative impact of different parameters on Q_{crit}.

Since the Q_{crit} model proposed in Chapter 4 describes Q_{crit} as a function of different parameters of both the NMOS and PMOS transistors constituting the SRAM cell, the model can be used to readily estimate the impact of variations of different transistor parameters on Q_{crit}. In this chapter, we investigate the impact of these variations and quantitatively identify their relative contribution to Q_{crit} variations.

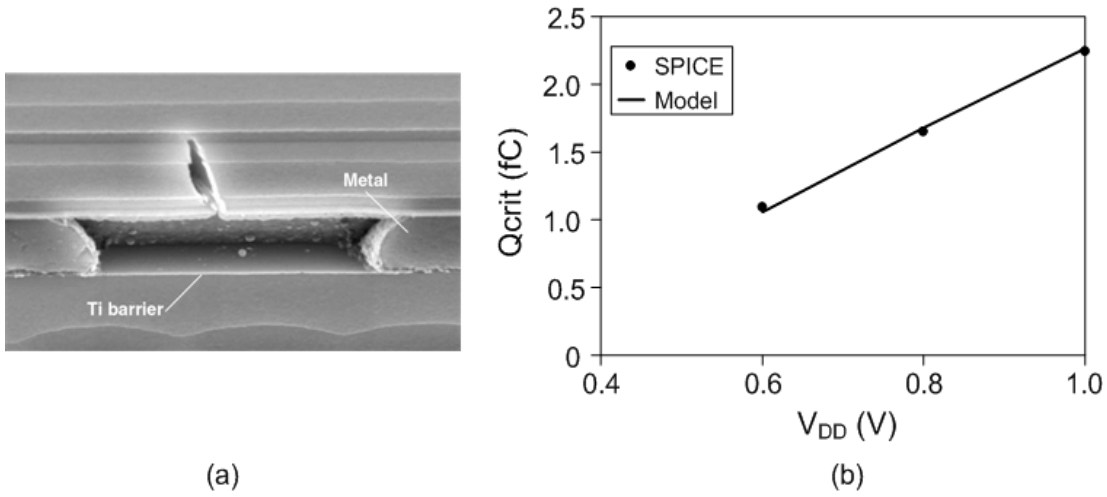


Figure 5.1: a) A void in a metal line and b) critical charge variation as a function of cell supply voltage.

5.1 Impact of Process Variations the on Critical Charge

Nanometric processes cause variations in a number of transistor parameters. These include physical parameters, such as channel length (L) and width (W), oxide thickness, etc., and electrical parameters, such as junction capacitance, threshold voltage (V_{TH}), etc. While there are other parameters (e.g., gate poly dimensions, wire geometry, etc.) that may also vary, variations in L, W, and V_{TH} are the most prominent since they directly affect the transistor's current driving capability. Accordingly, L, W, and V_{TH} variations play a key role in causing Q_{crit} variation in fabricated SRAMs. In addition, due to the high aspect ratio of nanometric technologies, SRAM cells can have defective contacts and vias that fail to properly connect two layers [60]. These defective contacts cause further variation in Q_{crit}, making the Q_{crit} spread wider on a fabricated SRAM population.

5.1.1 V_{DD} Variation

Local V_{DD} variations result from faulty vias or voids in the metal supply voltage line of the SRAM cell. Figure 5.1(a) shows a void in the metal line. If the metal line is the V_{DD} line supplying power to the SRAM cell, such voids cause resistive voltage drops on the line and lower the V_{DD}. In order to determine the impact of such V_{DD} variations, we extract Q_{crit} from SPICE simulations (90nm CMOS) and the proposed model in Chapter 4 for

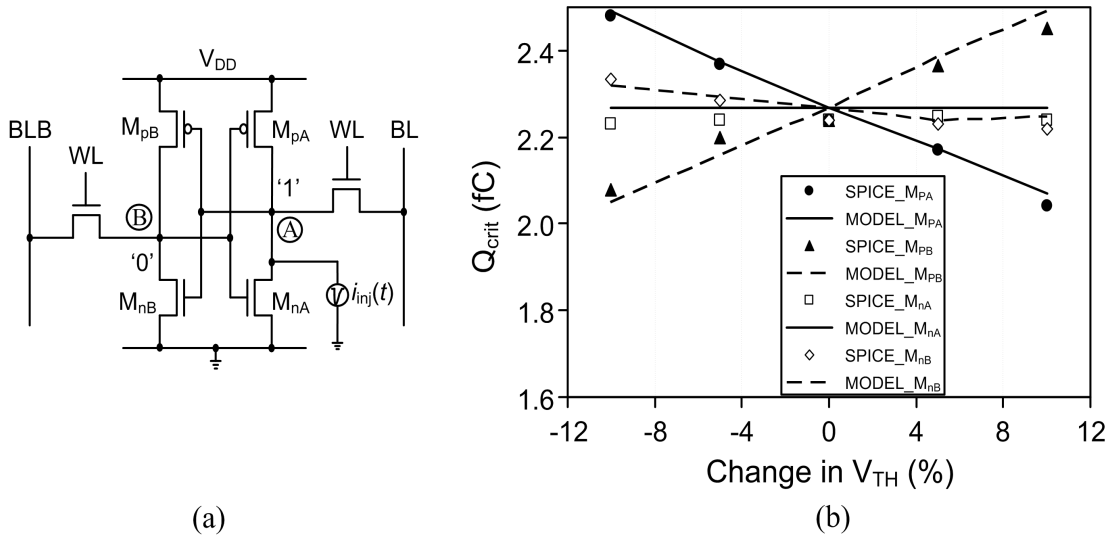


Figure 5.2: a) A 6T SRAM cell considering a particle strike at node A and b) critical charge variations as a function of threshold voltage variation in different transistors.

different V_{DD}. We keep the word line and bit line voltages at their typical values, i.e., at 0 V and 1 V, respectively. Q_{crit} values thus obtained are shown in Figure 5.1(b). Evidently, Q_{crit} varies linearly with local V_{DD} variations. In addition, the calculated values of Q_{crit} using the proposed model are in excellent agreement with SPICE simulation. Therefore, if V_{DD} variations across an SRAM population are known, the proposed model can be used to extract the resulting Q_{crit} variations and hence the SER variations.

5.1.2 V_{TH} Variation

V_{TH} variations in nanoscale processes primarily results from random dopant fluctuations in the channel region, channel length and width variations, and gate line edge roughness (LER). The effects of V_{TH} variation of the driver (M_{nA} and M_{nB}) and load transistors (M_{pA} and M_{pB}) on Q_{crit} of an SRAM cell are shown in Figure 5.2. Here, the '0' point on the x-axis corresponds to the typical values of V_{TH}. As evident from Figure 5.2(b), the proposed model is in good agreement with SPICE simulations with a maximum discrepancy of 2.1%.

The V_{TH} variation of M_{pA} has the largest impact on Q_{crit}. The lower is the V_{TH} of M_{pA}, the higher is Q_{crit} and vice versa. This is due to the fact that a lower V_{TH} of M_{pA} implies a larger restoring current, thus requiring a larger Q_{crit} to upset the cell. Similarly,

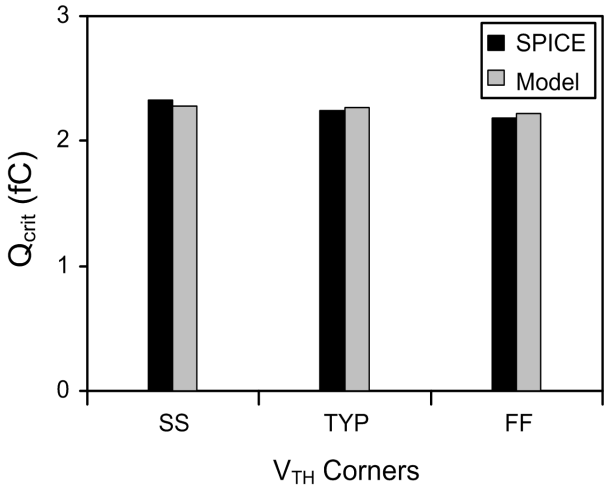


Figure 5.3: Critical charge at different process corners (temperature 27°C).

a lower V_{TH} of M_{nB} means a stronger restoring current for node B, making it harder to raise the voltage at node B. This slows down the flipping process, causing an increase in the Q_{crit}. In contrast, a lower V_{TH} of M_{pB} provides a stronger pull-up current for node B, which facilitates the flipping process. A higher V_{TH} of M_{pB} does the opposite. Thus, the impact of V_{TH} variation of M_{pB} has an opposite effect in comparison to that of M_{pA}. Similarly, the impact of V_{TH} variation of M_{nA} has an opposite effect compared to M_{nB}.

The variations in Q_{crit} with slow and fast corners ($\pm 6\%$ V_{THp}, $\pm 3\%$ V_{THn}) are shown in Figure 5.3 for SPICE simulations and model based calculations. Clearly, the model is in good agreement with SPICE.

5.1.3 *L* and *W* Variation

The variation in L primarily results from gate poly width variation caused by photolithographic limitations. On the other hand, the variation in W primarily stems from variation in field oxide step. Figure 5.4 shows the effect of L variations of different transistors on Q_{crit} of the SRAM cell. As evident from the figure, the proposed model is in good agreement with SPICE simulations with a maximum discrepancy of 3.1%. Similar to V_{TH} variation, L variation of M_{pA} has the maximum impact on Q_{crit}. In contrast, similar variations of M_{nA} have the least effect. In fact, a smaller L of M_{pA} results in a larger restoring current, which acts against the flipping process. Thus a higher Q_{crit} is required

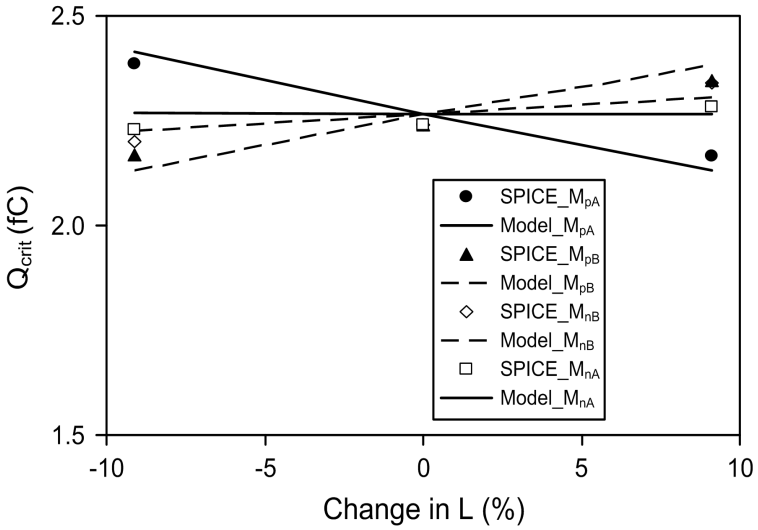


Figure 5.4: a) Critical charge variations as a function of channel length of different transistors in an SRAM cell.

to flip the cell. For M_{pB}, the effect is the opposite because, once turned on, it provides stronger pull-up current for node B and facilitates the flipping process. Similarly, a smaller L of M_{nA} reduces Q_{crit} by supplying stronger pull-down current for node A during the flipping process. However, the effect of L variation of M_{nB} is not straightforward. Increasing L means not only a decrease in the current drive of the transistor, but also an increase in the effective gate capacitance. The increased gate capacitance adds to the total capacitance of node A, whereas the decreased current drive does not come into play until M_{pB} starts to pull up node B. Here, the increase in node capacitance dominates, resulting in an increase in Q_{crit} with increasing L of M_{nB}.

The effect of W variations has been observed to be the opposite of that of L variations for the transistors except for M_{nB} and M_{nA}. Increasing W increases both the gate capacitance and the current drive of M_{nB}, thus increasing Q_{crit}. On the other hand, increasing the W of M_{nA} increases Q_{crit} because of an increase in gate-drain overlap capacitance, which dominates over the increase in its current drive.

5.1.4 Resistive Opens

Providing reliable contacts and vias is a growing challenge in nanometric technologies. Contacts and vias may fail to connect two layers or may only weakly connect the lay-

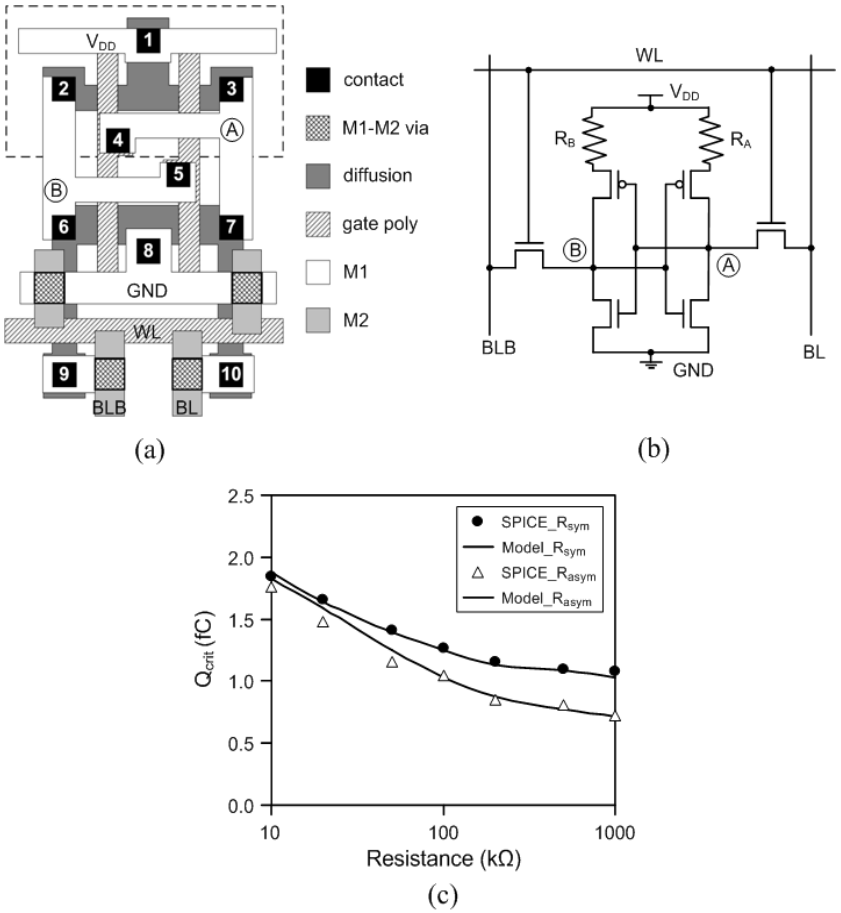


Figure 5.5: a) A 6T SRAM cell layout showing 10 contacts, b) cell schematic with resistive opens on the pull-up paths, and c) critical charge variations as a function of symmetric and asymmetric resistive opens.

ers [60]. The former is referred to as a *strong open* while the latter is referred to as a *weak open*. Strong opens immediately affect an SRAM's yield. On the other hand, weak opens allow the SRAM to function, but degrade its performance by introducing unexpected resistance. Weak opens pose a potential reliability threat as they can escape traditional SRAM functional tests.

Depending on the design, an SRAM cell may have ten to fourteen contacts (see Figure 5.5(a)). These contacts are the potential locations of weak opens. Since in the previous sub-sections we have noticed that the critical charge is highly sensitive to the load transistors, we only consider possible weak opens in the pull-up path. A weak connection between the V_{DD} line and both sources of load transistors is referred to as a symmetric

defect, R_{sym}, which equals R_A or R_B in Figure 5.5(b). A weak connection between only one of the load transistors' sources and the V_{DD} line is referred to as asymmetric defect, R_{asym}. While R_{asym} can be at either side of the cell, we consider it as R_A (i.e., R_B=0) in Figure 5.5(c). As evident from the figure, the model quite reasonably predicts Q_{crit} variations as a function of R_{sym} and R_{asym}. The maximum discrepancy with SPICE is 4.7%. The discrepancy can be attributed to the complex change in V_{TH}, and hence in g_m, due to body effect of load transistors. Figure 5.5(c) also shows that the impact of R_{asym} on Q_{crit} is more severe. This is because, R_{sym} reduces pull-up current without changing the symmetry of the voltage transfer characteristics (VTC) of the cell. Conversely, R_{asym} affects both the VTC and the pull-up current.

5.2 Relative Process Dependence of Critical Charge and SNM

The data stability of an SRAM is conventionally measured by its static noise margin (SNM), which is the minimum DC noise that can flip the cell. Accordingly, the impact of process variations on SRAM data stability has also been investigated using the SNM [61]. Therefore, it would be interesting to compare the relative dependence of SNM and Q_{crit} on process variations.

5.2.1 Definition and Process Dependence of SNM

In order to investigate the process dependence of SNM, we followed the SNM definition that has been used by Seevinck, et. al. [62]. According to the definition, the SNM is given by the side of the largest square that fits into the 'eyes' of the VTC of the SRAM cell (see Figure 5.6). If an SRAM cell is perfectly symmetrical, the two squares embedded into the VTC are of the same size. In quiescent state (WL=0), the size of the squares is larger and the SNM is higher than the SNM in a read-accessed state. During the read-access (i.e., when WL=1), the access transistors form a voltage divider with the driver transistors and degrade the 'zero' level of the cell. As a result, the 'eyes' of the VTC become smaller, reducing the size of the squares and hence the SNM, as shown in Figure 5.6. The degree of the SNM degradation exhibits an inverse relationship with

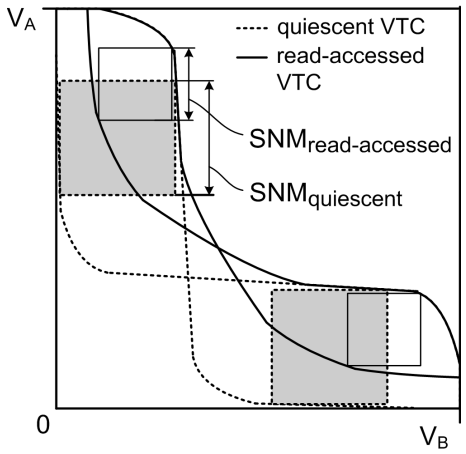


Figure 5.6: SRAM VTCs in quiescent and read-accessed modes with corresponding static noise margin (SNM).

the cell ratio, CR. Since the SNM of a read-accessed cell is the smaller, it represents the worst case scenario. Accordingly, we consider the read-accessed SNM in this analysis. In addition, since process variations and non-catastrophic defects introduce an asymmetry in the VTC and make one of the squares smaller than the other, we define SNM as the side of the smaller of the two squares.

Based on the above definition, we investigate the dependence of SNM on process variations. Similarly to the Q_{crit} extraction process, we vary only one parameter (V_{TH}, L or W) of a transistor at a time. The resulting SNM variations are in line with the SNM variations reported in [61]. The SNM is at its maximum for typical values of V_{TH}, L or W. The SNM decreases if any of the simulated parameters deviates from its typical value. This decrease can be attributed to the reduction of the size of one of the squares inside the 'eyes' of the VTC due to the asymmetry introduced by the variation of a given parameter. The decrease in the SNM is the largest for variations in the parameters of the driver transistor. This is because the 'zero' level degradation during the read-access largely depends on the current driving strength of the driver transistor.

5.2.2 Critical Charge vs. SNM

Now we investigate the relative process dependence of Q_{crit} and SNM. For this purpose, process variations in the load and driver transistors that are connected to the struck node

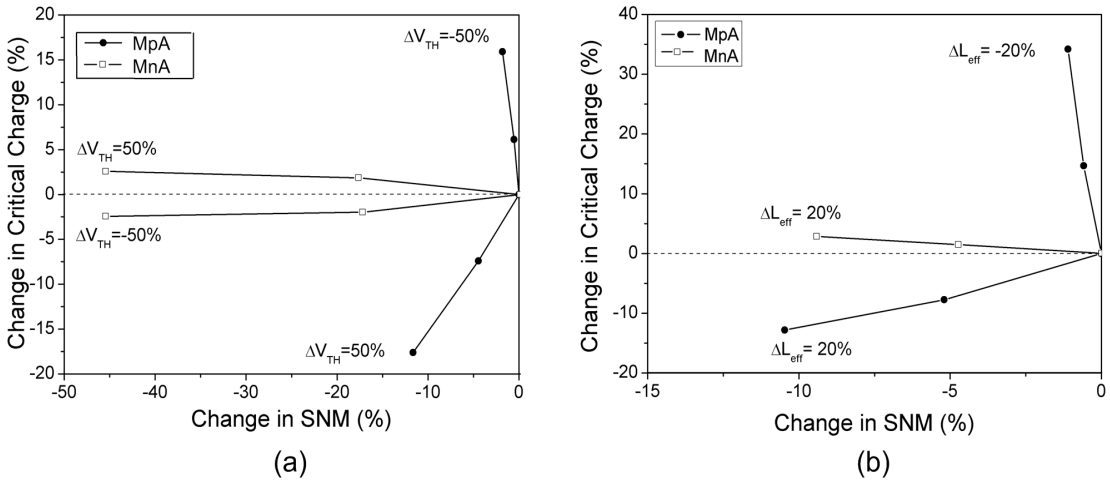


Figure 5.7: SNM vs Q_{crit} for a) varying V_{TH} and b) varying L. Simulated in 130nm CMOS technology

are considered. These transistors have the largest impact on Q_{crit} and SNM, respectively.

Figure 5.7(a) shows the change in Q_{crit} in comparison to the change in SNM when V_{TH} of the load (M_{pA}) and driver (M_{nA}) transistors is varied. For an increase in the V_{TH} of M_{pA}, both Q_{crit} and SNM decrease, the former showing a steeper slope. Therefore, for a given increase in the V_{TH} of the load transistor, some cells that have acceptable SNM may not conform to Q_{crit} specifications, thus reducing the yield. On the other hand, for a decrease in V_{TH} of M_{pA}, the decrease in SNM is minimal while the increase in Q_{crit} is appreciable. Since a higher Q_{crit} implies higher robustness against soft errors, the variation in Q_{crit} for decreasing V_{TH} of the load transistor does not pose any reliability threat in terms of soft error susceptibility. In contrast, the V_{TH} variation of M_{nA} exhibits little change in Q_{crit} but a considerable change ($\pm 45\%$ for $\pm 50\%$ change in V_{TH}) in the SNM. In this case, the reliability of the SRAM array should be assessed based on the SNM distribution across the chip.

Figure 5.7(b) shows the interdependence of Q_{crit} and the SNM when the L of M_{pA} and M_{nA} is varied. When the L of M_{pA} is increasing, deviations in both the SNM and Q_{crit} are similar. However, decreasing L of M_{pA} significantly increases Q_{crit} while merely decreasing the SNM. By contrast, increasing the L of M_{nA} considerably decreases the SNM while causing only small increase in Q_{crit}. Therefore, when L of the load and driver transistors varies, the SNM appears as a key to assessing the SRAM reliability.

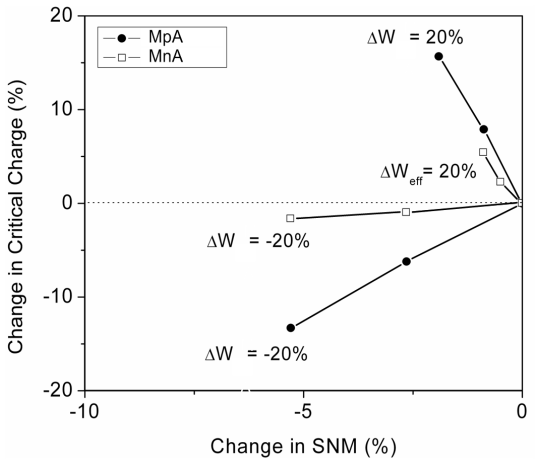


Figure 5.8: SNM vs Q_{crit} for varying W. Simulated in 130nm CMOS technology

Figure 5.8 shows the interdependence of Q_{crit} and the SNM when W of the load and driver transistors is varied. In contrast to L variation, W variation causes a larger variation in Q_{crit} than in the SNM. The increase in W of M_{pA} and M_{nA} causes Q_{crit} to increase due to the increase in the restoring current and node capacitance, respectively. However, a decrease in W of M_{pA} significantly reduces Q_{crit} ($\sim$ -15%) while showing only a small decrease ($\sim$ -5%) in the SNM. Thus, reliability assessment based on SNM may be too optimistic in such a case. On the other hand, a decrease in the W of M_{nA} reduces the SNM at almost a doubled rate as opposed to Q_{crit} reduction.

As evident from above discussions, for a variation in a given process-dependent parameter, the resulting variations in Q_{crit} and SNM are not unidirectional. Depending on the parameter, Q_{crit} may increase or decrease while the SNM always decreases. In particular, an increase in V_{TH} or L, or a decrease in W of the load transistor significantly degrades Q_{crit} while reducing the SNM by a small amount. Thus, some cells having an acceptable SNM may not meet Q_{crit} and hence the SER requirements. Accordingly, the conventional approach of assessing SRAM cell stability using the SNM is not sufficient. Both Q_{crit} and SNM should be used to estimate the possible impact of process variations on the cell stability and hence the data integrity.

5.3 Summary

In this chapter, we have investigated the process dependence of the soft error critical charge using the model developed in Chapter 4 as well as using SPICE simulations. The model is in good agreement with SPICE simulations while predicting the impact of process variations, thus further proving its reliability and versatility. We also have analyzed the relative process-dependence of the critical charge and the SNM. The critical charge is the most sensitive to parameter variations of the load transistor while the SNM is the most sensitive to parameter variations of the driver transistor. Thus, SRAM cells having good SNM can have significantly lower critical charge. If such an SRAM is tested and passed using SNM based tests, the SRAM can have significantly poor soft error tolerance.

Chapter 6

Energy-Efficient Soft Error Mitigation Technique

This chapter presents an energy-efficient high-speed error correction scheme for soft error mitigation in low-power SRAMs. The scheme has been implemented in silicon and tested under neutron radiation to prove its efficacy.

In order to mitigate soft errors in SRAMs, both the circuit level, such as an upset-hardened cell [53], and the architecture level, such as an error correction code (ECC) [56], techniques can be employed. ECC is preferable due to its lower area overhead. However, the cost associated with ECC can be significant. For example, a Single Error Correcting Double Error Detecting (SECDED) code requires storing 7 check-bits for 32 bit data, increasing the memory array size by 22%. This increase manifests itself in higher cost and larger leakage and active power dissipation. Furthermore, conventional ECC operates on the read data path and thus can increase the read delay by up to four clock cycles [63]. Therefore, an area and power efficient ECC with minimal delay penalty is of significant interest. In this chapter, we propose a cost-effective multiword ECC technique that minimizes the check-bit area penalty as well as the read delay penalty.

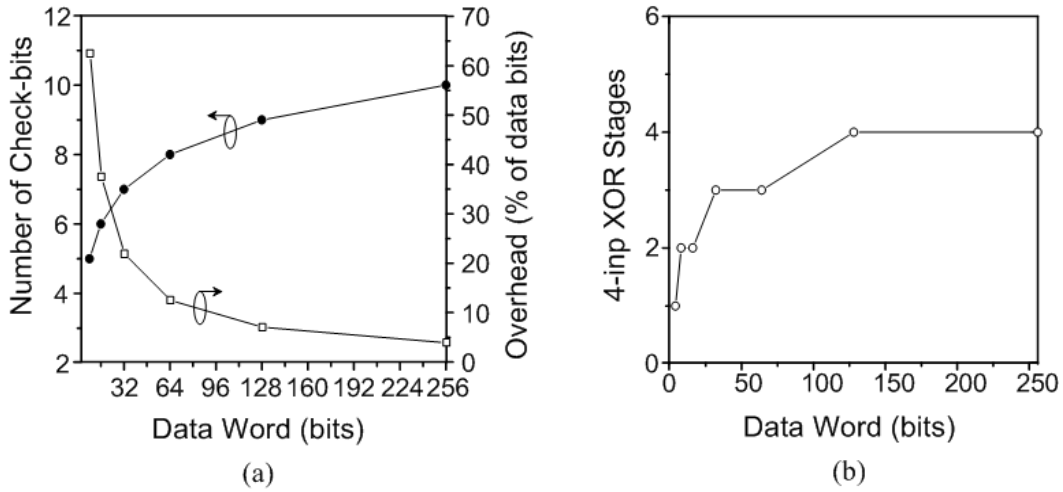


Figure 6.1: a) Number of check-bits and pertinent overhead as a function of the data words protected with ECC and b) number of 4-input XOR stages in the check-bit generator.

6.1 Proposed Multiword ECC

The area overhead for the check-bits decreases with increasing word size, as seen in Figure 6.1. Thus, it is preferable to use larger data words to limit the area overhead. However, a larger data word increases the complexity and delay of the ECC logic. In order to take the advantage of a larger data word and reduce pertinent implementation complexities, we propose the multiword based ECC (MECC) scheme.

6.1.1 ECC Word and Logic Circuits

As seen in Figure 6.1(a), ECC overhead reduction beyond 128 bits of data is not as significant. Accordingly, we choose a word size of 128 bits for our ECC design. Since a typical data size is less than 128 bits, multiple data words can be combined to get a composite 128-bit word. In this work, we consider 32-bit data word so that the composite word consists of four such words. To limit the complexity of the ECC logic, we choose the SECDED code, which is based on a single-error-correctable Hamming code. In addition, we use 4-input XOR gates to reduce the number of stages and hence the delay in the check-bit generator, which is a vital block on the critical path. Figure 6.1(b) shows the number of stages required to implement a 128-bit data-based check-bit generator using 4-input XOR gate. As can be seen, only one additional stage is required if the word size

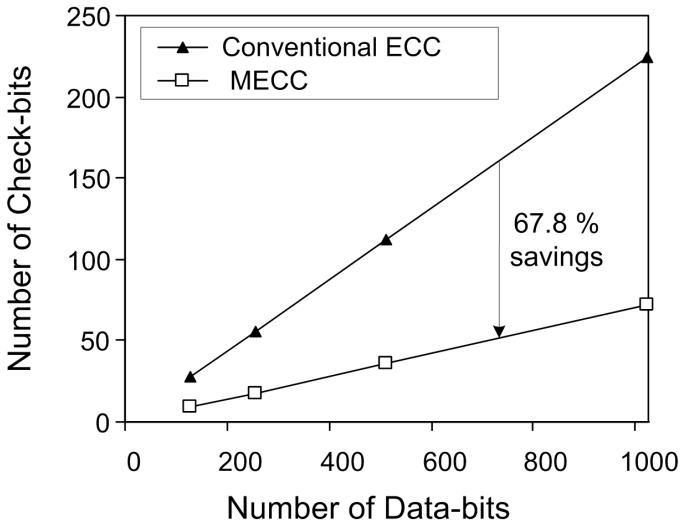


Figure 6.2: Check-bit saving as a function of data bits per row in the SRAM array.

increases from 32 bits to 128 bits. This supports using a 128-bit composite word for ECC, resulting in an incremental delay addition at significant overhead reduction.

Each 128-bit composite word in MECC needs 9 check bits according to the SECDED code. On the other hand, in a conventional scheme, each of 32-bit data words would require 7 check-bits, implying 28 check-bits for four data words. Thus, MECC scheme reduces the number of check-bits by 67.8%. Figure 6.2 illustrates the saving as a function of data bits per row in an SRAM array. It should be noted that the number of check-bits increases with the number of data bits, keeping the percentage saving constant. The saving directly translates into a significant amount of area and power savings. The reduced number of check-bits means an approximately 13% smaller array, i.e., shorter word lines (WLs) and fewer transistors. A shorter WL reduces decoder active power while fewer transistors decrease the leakage power. Thus, compared to conventional ECC, MECC reduces the array leakage by 12.2%.

6.1.2 Array Power Reduction

In order to further reduce the leakage power in the MECC-protected SRAM array, three approaches can be adopted:

- reducing the supply voltage (V_{DD}), like the drowsy cache scheme

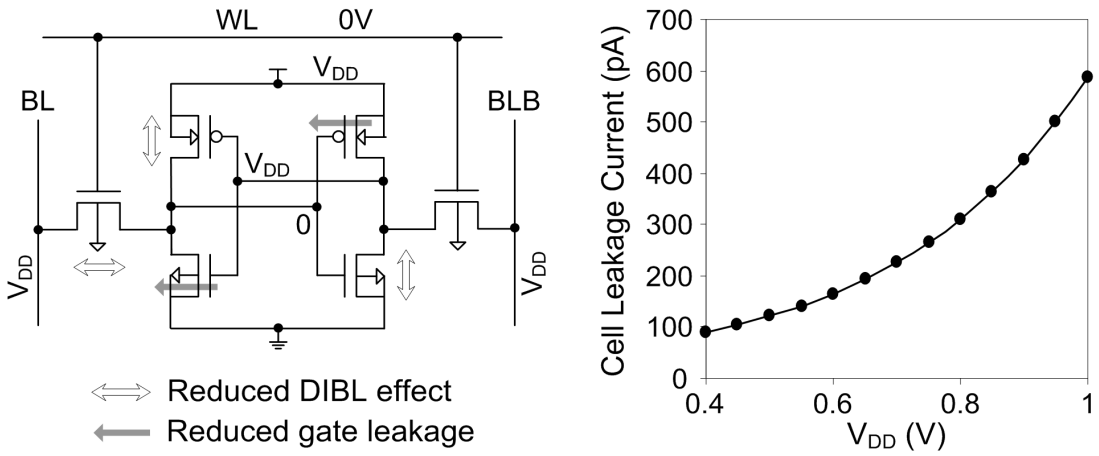


Figure 6.3: DIBL effect minimization in an SRAM cell by V_{DD} reduction and the resulting leakage current reduction.

- controlling the virtual supply (V_H) line, and
- controlling the virtual ground (V_{GND}) line, like the sleep transistor scheme.

In the following we describe the advantages and disadvantages of each of these approaches.

A. Leakage Minimization by V_{DD} Reduction

In this approach, the supply voltage (V_{DD}) of the SRAM array is switched to a lower voltage when the array is not accessed. The leakage power saving results from two mechanisms:

i) reduction of DIBL effect on OFF transistors and

ii) reduction of gate tunneling current in ON transistors.

Figure 6.3 shows these mechanisms in a 6T SRAM cell and the resultant leakage current reduction as a function of V_{DD}.

The V_{DD} reduction technique is primarily applied to SRAM blocks that are inactive. The amount of V_{DD} reduction depends on the aggressiveness of leakage minimization. The reference voltage for reduced V_{DD}, i.e., V_{DD-low} is typically generated on-chip using a voltage converter. However, the value of V_{DD-low} must be greater than or at least equal to the data retention voltage (DRV) so that the data in the SRAM are preserved. With

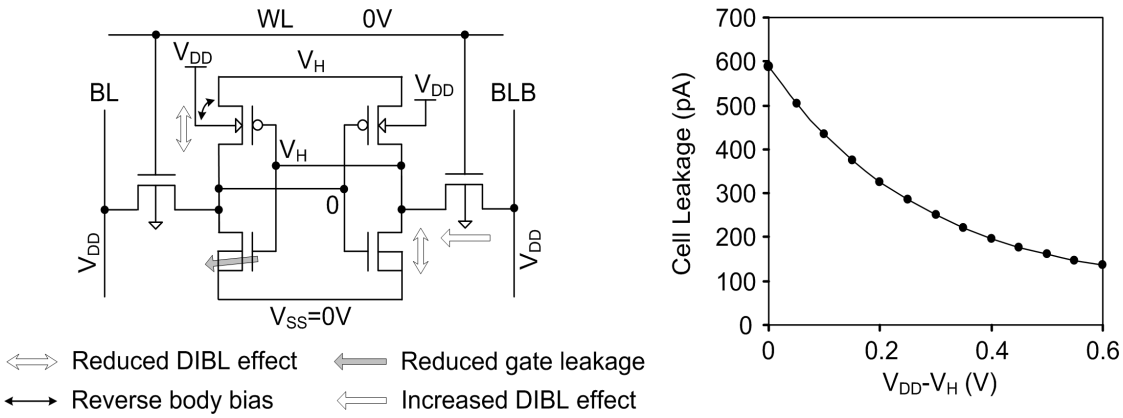


Figure 6.4: Leakage reduction mechanisms in virtual V_H technique and cell leakage current as a function of the voltage difference between V_{DD} and the virtual rail.

the increasing process variations and parametric defects in nanoscale SRAMs, finding a reliable DRV can be difficult. In addition, when the block needs to be accessed, V_{DD-low} is switched back to the nominal voltage (V_{DD}). Thus, V_{DD} reduction technique incurs a power-up time delay.

B. Leakage Minimization by Virtual Supply (V_H) Control

In this technique, the cell supply voltage or logic '1' voltage (V_H) is lowered keeping the n-well at V_{DD}. Thus, the leakage current is reduced by the following three mechanisms:

i) reduction of the DIBL effect on the OFF transistors,

ii) reduction of the gate tunneling current in the ON NMOS transistor, and

iii) application of RBB on the OFF PMOS transistors.

These mechanisms are shown in Figure 6.4. It should be noted that due to a potential difference between the logic '1' voltage (V_H) and the BL voltage, there is a new leakage component through the access transistor. As a result, despite the additional RBB on the leaky PMOS transistor, the total cell leakage current is larger than the cell leakage in the V_{DD} reduction technique. In addition, like the V_{DD} reduction technique, the reference voltage for V_H needs to be generated on chip, which implies a converter power consumption. The constraint of DRV and the power-up delay penalty also apply to the virtual V_H technique.

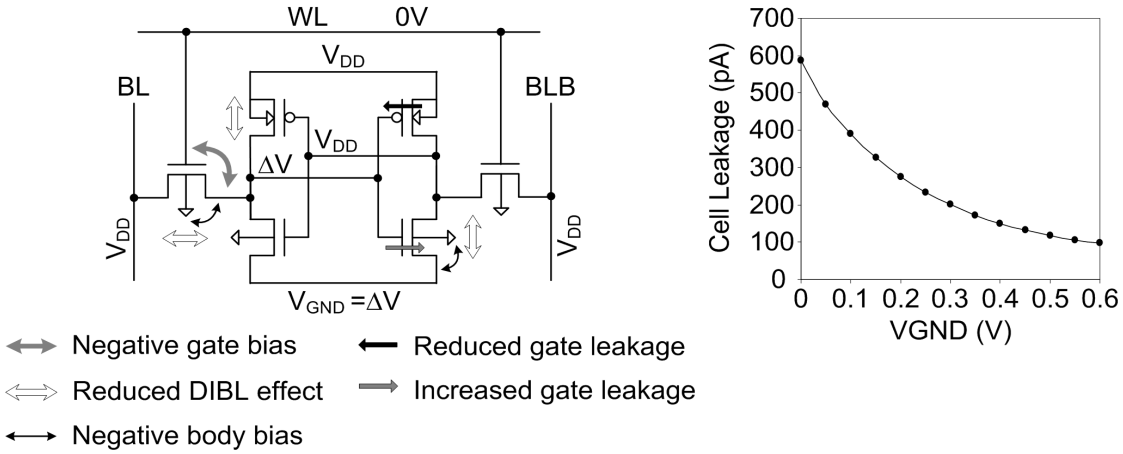


Figure 6.5: Leakage reduction mechanisms in virtual V_{GND} technique and cell leakage current as a function of V_{GND}.

C. Leakage Minimization by Virtual Ground (V_{GND}) Control

In this technique, the source line potential of driver transistors, i.e., the virtual ground potential V_{GND} is raised when the cell is not accessed (see Figure 6.5). Thus, the leakage current is reduced by the following four mechanisms:

i) reduction of the DIBL effect on the OFF transistors,

ii) reduction of the gate tunneling current in the ON PMOS transistor,

iii) application of RBB on the OFF driver and access transistors, and

iv) application of negative gate bias on the OFF access transistor.

While there is a slight increase in gate tunneling current in one of the driver transistors, the reduction in the leakage current by the above mentioned mechanisms dominates. As a result, the total cell leakage is significantly reduced with increasing V_{GND}, as shown in Figure 6.5. However, like the virtual V_H technique, the reference voltage for V_{GND} needs to be generated on chip, which implies a converter power consumption.

The upper limit of V_{GND} is set by the DRV such that the rail-to-rail voltage of the cell does not go below the DRV. In addition, the technique incurs a delay penalty as the V_{GND} needs to be pulled down to 0 V in the read access in order to maximize the read current.

Another leakage reduction technique could be to simultaneously raise V_{GND} and re-

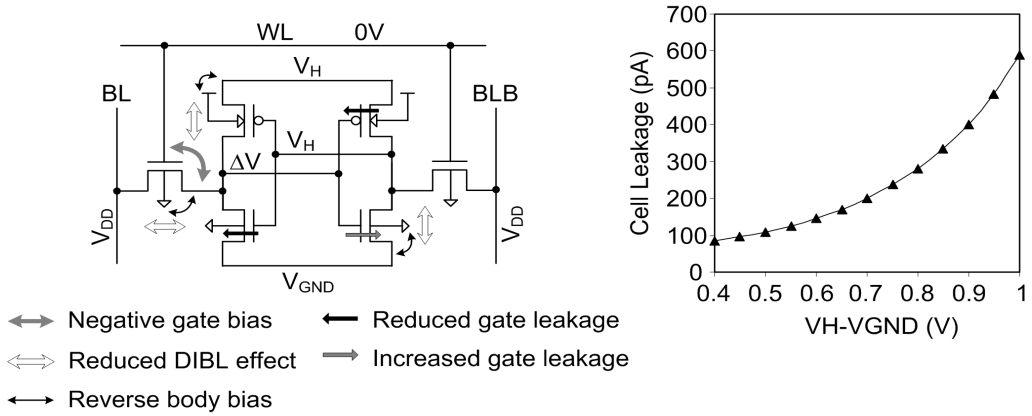


Figure 6.6: Leakage reduction mechanisms in simultaneous control of V_{GND} and V_H and resultant cell leakage current.

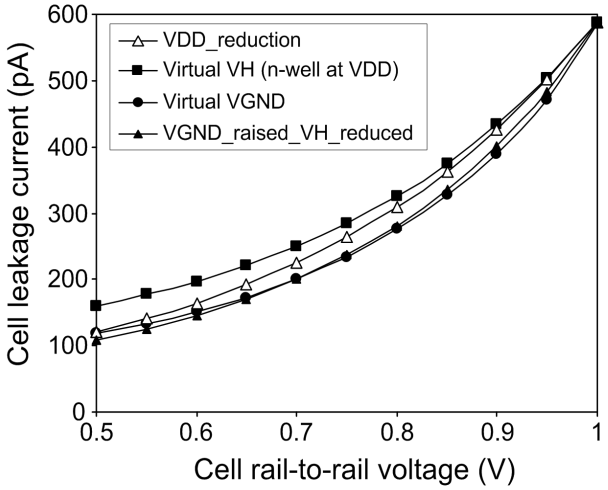


Figure 6.7: 6T SRAM cell leakage current in different leakage reduction techniques.

duce V_H in order to take the advantages of both of these techniques. The resulting cell leakage current in such a case is shown in Figure 6.6. Since this technique requires controlling both V_{GND} and V_H, it involves more complex control circuits, generation of two reference voltages, and the associated power consumption.

Figure 6.7 shows the cell leakage current as a function of the cell rail-to-rail voltage in above mentioned leakage reduction techniques. As is evident from the figure, the simultaneous control of V_{GND} and V_H results in the minimum cell leakage, which is comparable to the cell leakage resulting from the virtual V_{GND} control technique. Since the virtual V_{GND} technique requires only one on chip reference voltage and fewer control

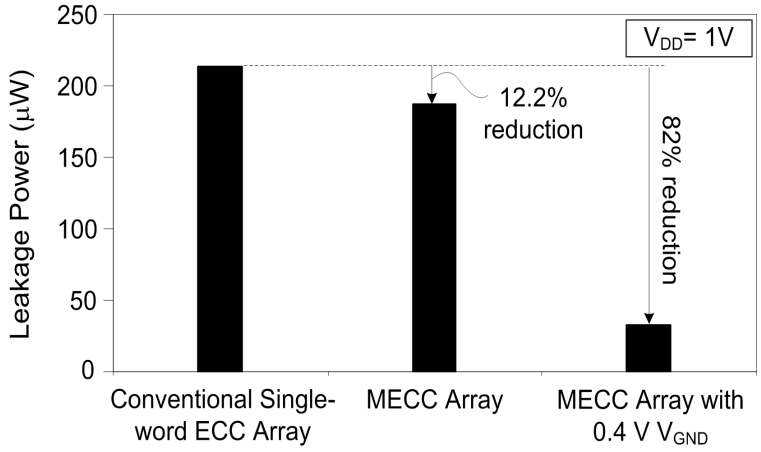


Figure 6.8: Leakage power saving in MECC protected SRAM array.

circuits, it appears as the most attractive leakage minimization technique. Accordingly, we choose this technique in the proposed MECC-protected SRAM.

We apply the V_{GND} control technique on the the composite words. In other words, when a composite word is not accessed, the V_{GND} of all cells in the composite word is raised. In particular, we set the V_{GND} equal to a threshold voltage, V_{TH}, which is easy to generate on chip. Considering V_{TH}= 0.4 V, we achieve 82% reduction in array leakage power compared to the conventional SRAM having nominal supply and ground potentials. This is illustrated in Figure 6.8.

6.1.3 Array Design

While the proposed MECC significantly reduces the check-bit area and leakage power by using 128-bit composite ECC word, it has a limitation in terms of error correction. The MECC scheme can correct only one error and detect two errors in the 128-bit composite word. However, cosmic radiation at ground level can induce multiple bit errors [64]. In order to deal with this issue, we interleave two composite words in a row, as shown in Figure 6.9. Thus, the first bit (b0) of the first composite word (W0) sits right beside the first bit (b0) of the other composite word (W1). Next sits the second bit (b1) of the first composite word (W0) and the second bit (b1) of the other composite word (W1), and so on. Such arrangement of the composite words in the row is illustrated in Figure 6.9 with reference to a row of the conventional single word ECC scheme. The check-bit saving in

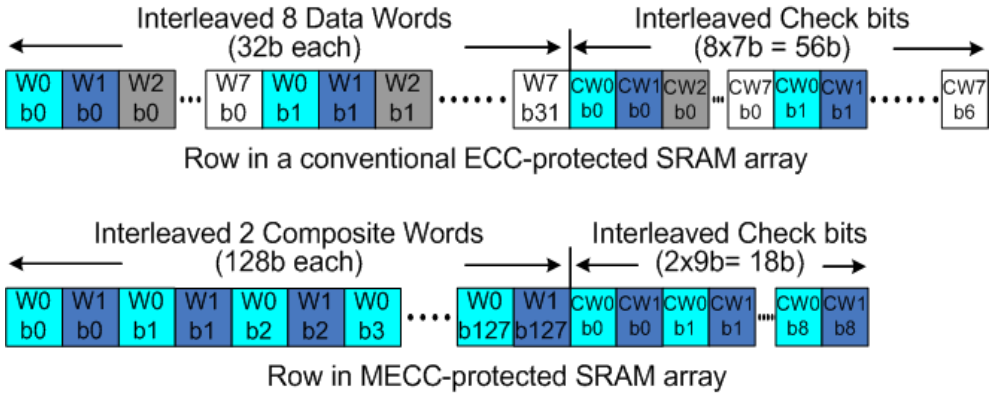


Figure 6.9: A row in conventional ECC- and MECC-protected SRAM.

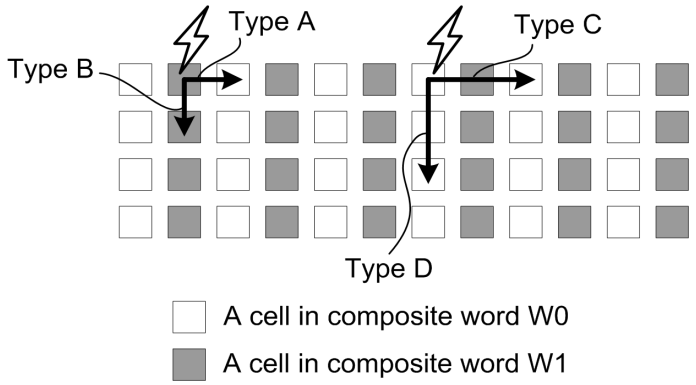


Figure 6.10: Possible error types resulting from a particle strike in the MECC SRAM array.

the MECC scheme is also clearly shown in this figure.

Because of the interleaving of two composite words, two adjacent bit errors in a row (Type A in Figure 6.10) will belong to two different composite words and will easily be corrected. In case of three bit errors in a row (Type C in Figure 6.10), two errors will belong to one composite word and the third error to the other composite word. MECC will then detect (double error) and correct (single error) these errors. In case of two or three bit errors in a column (Type B and D in Figure 6.10), each of the errors will belong to a different composite word and will be corrected. However, the probability of three bit error from a single particle strike is at least two orders of magnitude less than the probability of a single bit error [65].

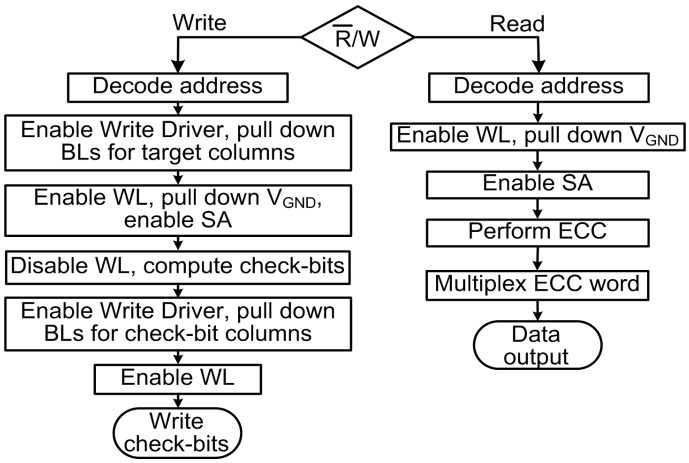


Figure 6.11: Flow chart of read and write operation in RVGND MECC scheme.

6.1.4 Read and Write Operations

The use of composite words in ECC requires special read and write operations. In a read operation, one of the two composite words on the selected row is read, passed through the ECC logic, and multiplexed (4-to-1) to provide the requested 32-bit data as shown in the flow chart in Figure 6.11. In contrast, the write operation is a combination of both read and write. In a write cycle, the selected WL is raised twice: first to write into the target data word and simultaneously read from other three data words in the same composite word, and second to write the new check-bits that are computed based on the new composite word. To reduce power in this case, row decoder decodes the row once; however, WL is activated by a control signal, WLE, that is ANDed with the decoder outputs.

6.2 Chip Integration with MECC

We have implemented a 64-kb SRAM macro with the MECC scheme in a commercial 90nm CMOS process. The SRAM array consists of 256 rows and 274 columns, where 256 columns are dedicated to data bits and 18 columns to check-bits. Each row comprises of eight 32 bit data words, thus totaling 2048 data words in the array. Therefore, 11 address bits ($2^{11} = 2048$) are required to identify any particular data word. The details of the SRAM design is discussed below.

Table 6.1: Cell sizing and performance metrics

Metric	Value
Driver transistors W/L	220 nm/110 nm
Load transistors W/L	120 nm/110 nm
Access transistors W/L	150 nm/130 nm
Read/static SNM	187 mV/370 mV
Read current	47 μA
Leakage current	140 pA
Cell area	3.04 μm^2

6.2.1 The SRAM Cell

We use the conventional 6T SRAM cell in our design. However, in order to minimize the leakage current, we employ high-V_{TH} transistors in the cell. The sizes of the transistors are optimized based on speed-power-area trade-offs. Table 6.1 summarizes the sizing and performance of the cell.

In order to facilitate the V_{GND} control of the composite words, two V_{GND}-lines run horizontally in a row. To accommodate these lines the height of the cell becomes larger than the height of a cell without V_{GND}. In particular, the V_{GND}-lines are implemented using metal-1, which increased the height of the cell by 9%. The resulting height and width of the cell are 2.235μm and 1.36μm, respectively. Figure 6.12 shows the cell layout.

6.2.2 Array and Biasing Circuit

The array is implemented by copying the cell in the horizontal and vertical directions. Thus, the bit lines run vertically while the cell V_{DD} and V_{GND} lines run horizontally. The substrate and n-well contacts were placed every eight cells in a row. The cell V_{DD} line was also connected to a higher metal layer at the same interval. Each V_{GND} line, on the other hand, is connected to two switches at the two ends of the row. These switches pull down the V_{GND}-line to 0 V when the composite word is accessed. To minimize

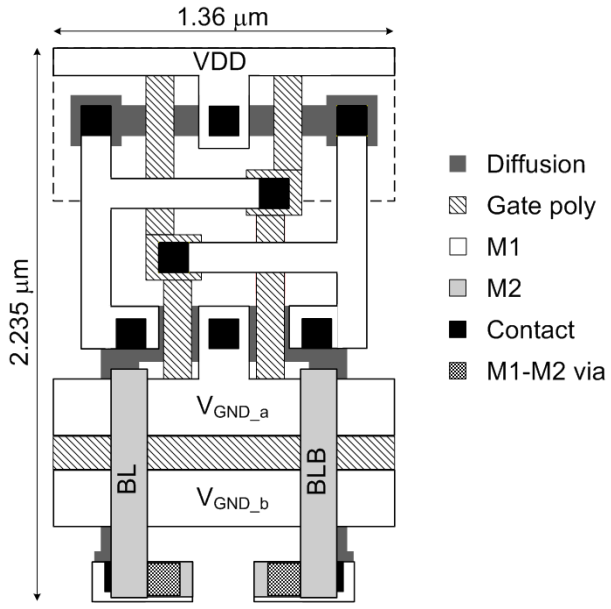


Figure 6.12: Layout of the SRAM cell used in MECC chip.

the associated area overhead, we use low-Vt V_{GND}-switches as shown in Figure 6.13. Thus, one V_{GND}-switch effectively carries the read current of 64 cells of the accessed composite word. Accordingly, we size the V_{GND}-switch and the metal V_{GND}-lines so that read current reduction (by resistive voltage drop) and electromigration are avoided. The resulting array area overhead is 22% with no noticeable degradation in the read current and SNM.

Putting V_{GND}-switches at regular interval for a block of 4 to 8 cells, as suggested in [66], would make the V_{GND}-switch smaller for the same read current and SNM, however, it would incur larger area overhead because of layout design rules. In addition, the area overhead for the V_{GND}-switches is offset by the saving in check-bit area. Considering 13% check-bit area saving by using multiword ECC, the net array area overhead is 9%. This translates into an overhead of 6.3% for the total chip area. The overhead can be reduced if read current and SNM degradations are allowed by downsizing V_{GND}-switches.

The bias voltage (V_{BIAS}) in Figure 6.13 is the reference voltage to which V_{GND} lines are connected when a composite word is not accessed. Since we choose V_{BIAS} to be equal to V_{TH}, it can be generated on chip by the circuit shown in Figure 6.14. The requirement for this circuit is that the ON current of the diode connected Q_{BIAS} be equal to the leakage current on the V_{GND} line. Thus, the width of Q_{BIAS} is typically

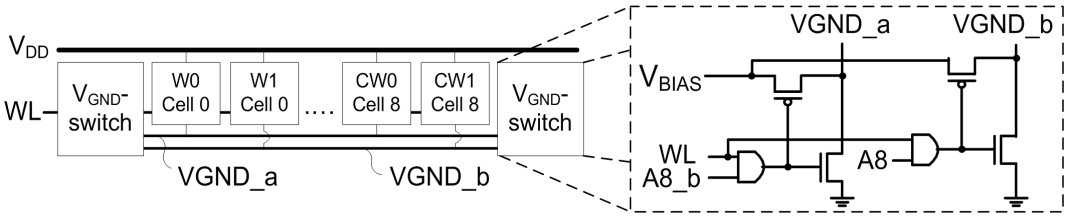


Figure 6.13: VGND-switch in a row and its circuit diagram.

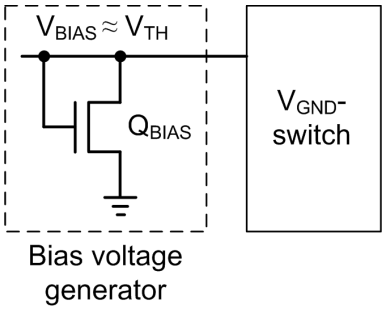


Figure 6.14: Simple on-chip bias voltage generator.

the minimum value while the length of Q_{BIAS} is large to achieve a stable $V_{BIAS} = V_{TH}$. Given the process variations, choosing the right channel length of Q_{BIAS} can be difficult. In addition, if the on-chip bias is implemented, we do not have any external control over the V_{GND} potential. Considering these factors, we supply V_{BIAS} off-chip.

6.2.3 Row Decoder and WL Driver

The row decoder is a 8-to-256 decoder that selects one of the 256 rows in the SRAM array. The decoder is implemented using the pre- and post-decoding architecture. The pre-decoder is a 4-to-16 static decoder. The post decoder uses 2-input static AND gates to AND each output of one of the pre-decoders with the 16 outputs of the other pre-decoder, thus generating the final 256 row select signals.

The word line (WL) drivers buffer the row select signals in order to drive the highly capacitive WLs. Each WL driver is implemented using an inverter chain with an AND gate to facilitate the control on WL by the signal WLE (see Figure 6.15). A WL is only active if the corresponding row select signal is high and WLE is high. This arrangement prevents any unintentional activation of WL as well as ensuring precise timing of the WL activation and deactivation relative to other signals like precharge, sense amplifier enable

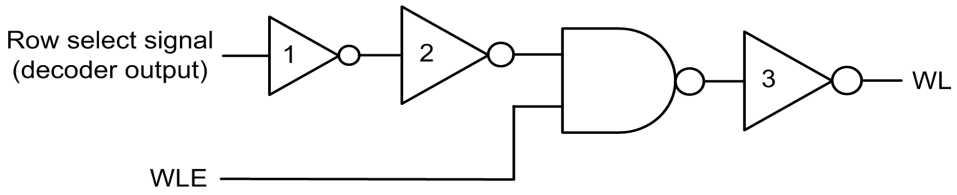


Figure 6.15: Word line driver circuit.

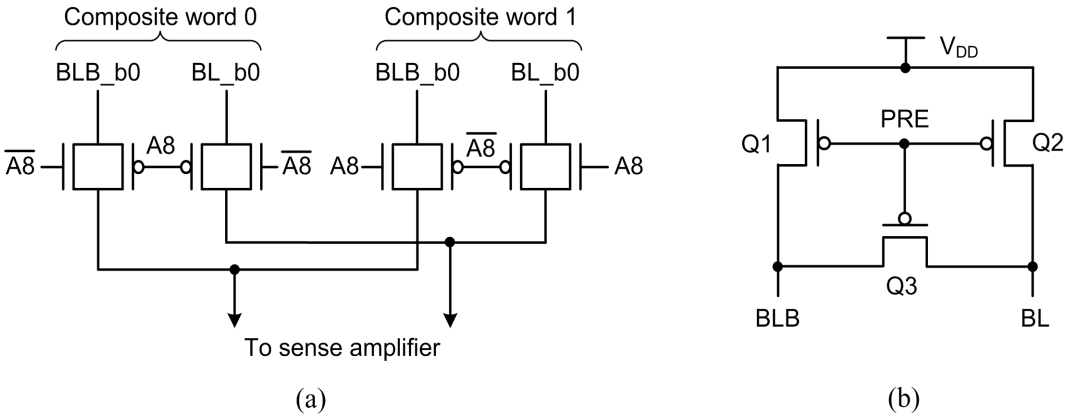


Figure 6.16: a) 2-to-1 column MUX and b) precharge and equalizer circuit.

(SAE), etc.

Since the write operation in the MECC scheme requires the WL activations, we use the WLE signal to serve this purpose. In a given write cycle, the row decoder decodes the row select signal based on the address inputs. Then, as soon as WLE is activated, WL will be activated. Since we need to deactivate the WL and activate it again within the same write cycle, we just deactivate and activate the WLE signal. In order to save the switching power in the inverter chain of the WL driver (see Figure 6.15), we place the AND gate, i.e., the NAND gate and inverter-3, at the end of the chain. As a result, during the activation and deactivation of the WLE signal, inverters 1 and 2 do not switch, thus saving the switching power.

6.2.4 Column MUX and Precharge Circuit

Since we have two 128-bit composite words in a row and one of them is accessed in a read or write cycle, we use a 2-to-1 128-line column MUX. We implement the MUX using transmission gate so that it can pass '1' and '0' in both directions. Figure 6.16(a) shows

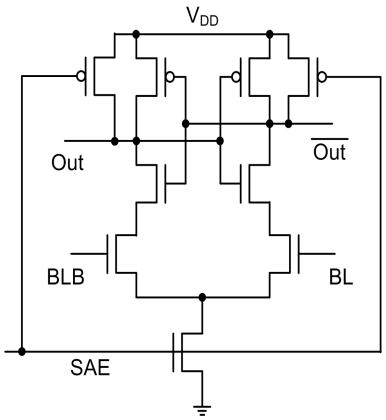


Figure 6.17: Sense amplifier.

the circuit diagram of one line of the column MUX.

The precharge circuit is placed in each column before the column MUX. The precharge circuit is implemented using three PMOS transistors as shown in Figure 6.16(b). As can be seen, transistors Q1 and Q2 precharges the bit lines to V_{DD} when PRE is low. At the same time, Q3 equalizes the potential of the two bit lines. Thus, the circuit in Figure 6.16(b), in fact, serves as both the precharge circuit and the equalizer. In addition, the circuit is free from any contention between the precharge transistor (Q1 or Q2) and the driver transistor of the selected SRAM cell. This is because, before activation of WL in any read access, the PRE signal is set to high, turning of Q1 and Q2.

6.2.5 Sense Amplifier

In order to achieve fast sensing at the power budget, we use the voltage sense amplifier shown in Figure 6.17. The sense amplifier offers a number of advantages over the simple latch-type sense amplifier described in Chapter 2:

- high input impedance due to the presence of a differential pair at the input
- high gain due to the differential pair and the cross-coupled load
- no clock feedthrough between the SAE signal and the output as there is no direct capacitive path between them.

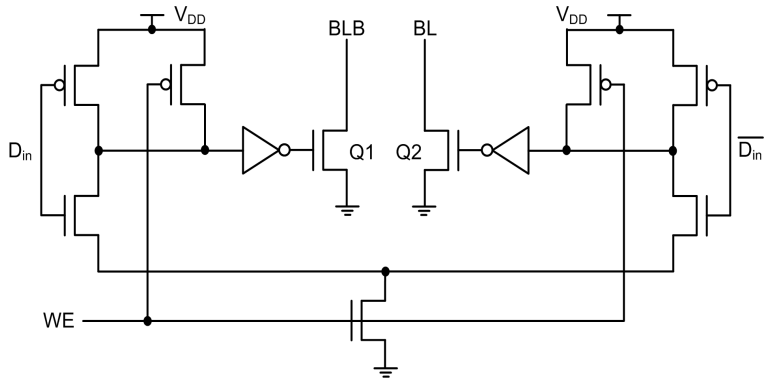


Figure 6.18: Write driver.

The sense amplifier is optimized for speed for a '1' to '0' transition and symmetrically laid out around the vertical axis parallel to the bit line. The layout has a high aspect ratio due to the tight pitch of the bit lines.

The sense amplifier is followed by a simple latch that is enabled as soon as the outputs of the sense amplifier reach full swing differential voltages. The latch holds the data for the rest of the clock cycle.

6.2.6 Write Driver

The write driver is placed on every column in parallel with the sense amplifier. In a write operation, it pulls down the bit lines according to the data input (D_{in}) before the WL is activated. However, it is active only when the write enable (WE) signal is high. Figure 6.18 shows the write driver that we use in the chip. Transistors Q1 and Q2 are made large in order to discharge the large bit line capacitance within the timing constraints.

6.2.7 ECC Circuits

The ECC circuits are designed to perform error correction on the 128-bit composite word. Thus, the check-bit generator generates 9 check-bits from 128 data bits. As shown earlier, using 4-input XOR gates in the check-bit generator, requires only four logic stages to generate the check-bits. In order to minimize the delay in these stages, we use an optimized 4-input transmission-gate (TG) XOR gate, which is shown in Figure 6.19(a). The TG XOR exhibits lower delay and power-delay-product (PDP) compared to other

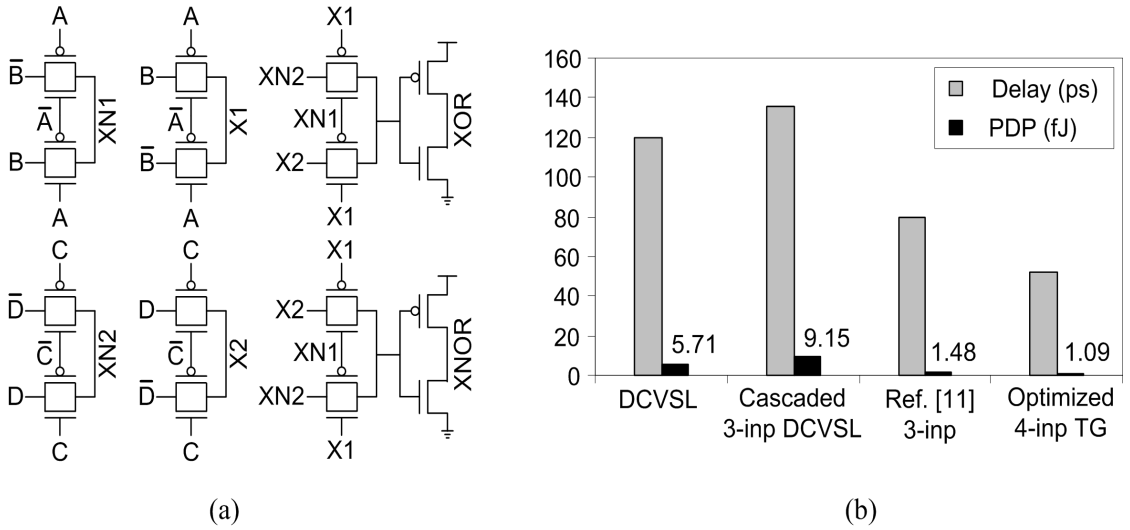


Figure 6.19: Optimized 4-input transmission gate XOR gate a) schematic and b) power delay product compared to other XOR gates.

4-input (e.g., DCVSL) or 3-input [67] XOR gates as shown in Figure 6.19(b). Thus, we minimize the delay in the check-bit generator, which is an important block on the critical path.

The syndrome generator consists of 9 2-input TG XOR gates that perform bitwise XOR operations on the stored check-bits read by the sense amplifier and the new check-bits computed by the check-bit generator. The syndrome decoder is a 8-to-137 decoder, which decodes the erroneous bit location. The decoder is implemented in the same way as the row decoder. The error corrector is implemented with 128 2-input TG XOR gates that perform bitwise XOR operations on read data bits and the syndrome decoder outputs corresponding to the data bits. Like the principle of the XOR operation, if an output of the syndrome decoder is '1', it will flip the corresponding data bit.

Finally, in order to test the functionality of the ECC block, four set/reset switches are placed between the sense amplifier and the the sense amplifier latch at selected bit positions. These switches enable flipping read bit values ('1'↔ '0') for locations 1, 95, and 96 in the composite word and location 1 in the check-bits.

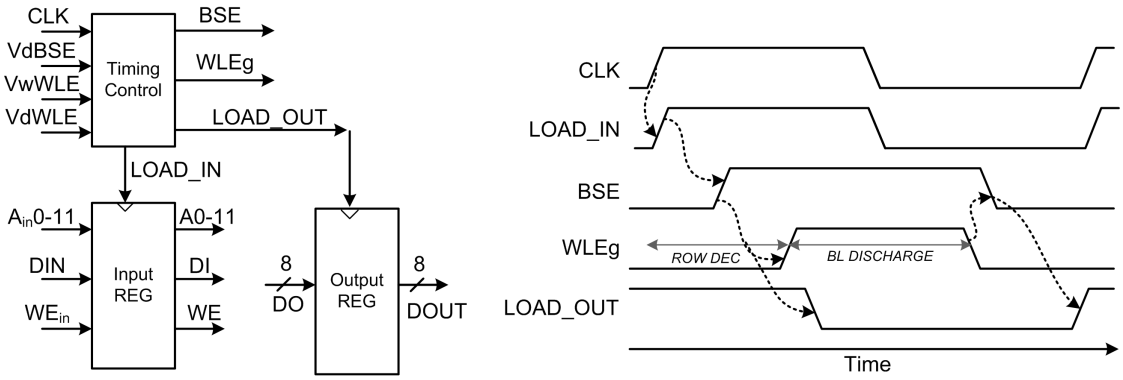


Figure 6.20: Timing diagram of global control signals.

6.2.8 Timing and Control Circuits

The MECC SRAM shares the silicon area and hence the IOs with two other SRAM macros that are not parts of this research. Accordingly, the timing and control circuits are implemented in two stages. The first stage is at the global level where three signals, namely, block select enable (BSE), global word line enable (WLEg), and output latch enable (LOAD-OUT), are generated. These signals are shared by all the three SRAM macros on chip. The rising edge and pulse width of these signals can be controlled using three off-chip analog voltage signals VdBSE, VdWLE, VwWLE, and VdLOADOUT, which are supplied. Figure 6.20 illustrates the global control signals and their timing relationship.

The other stage of the timing and control circuits is implemented at the local level where signals pertinent to proper operation of the MECC SRAM are generated. The local control signals for the MECC SRAM are generated from the two global signals BSE and WLEg. Primarily static delay elements and logic gates like AND, OR, etc. are used to generated the appropriate timing relationship between the signals. In addition, an off-chip analog voltage signal, VdSAE, is provided to move the rising edge of the clock signal for the sense amplifier latch (SA Latch). Figure 6.21 shows the timing diagram of the control signals for the MECC SRAM.

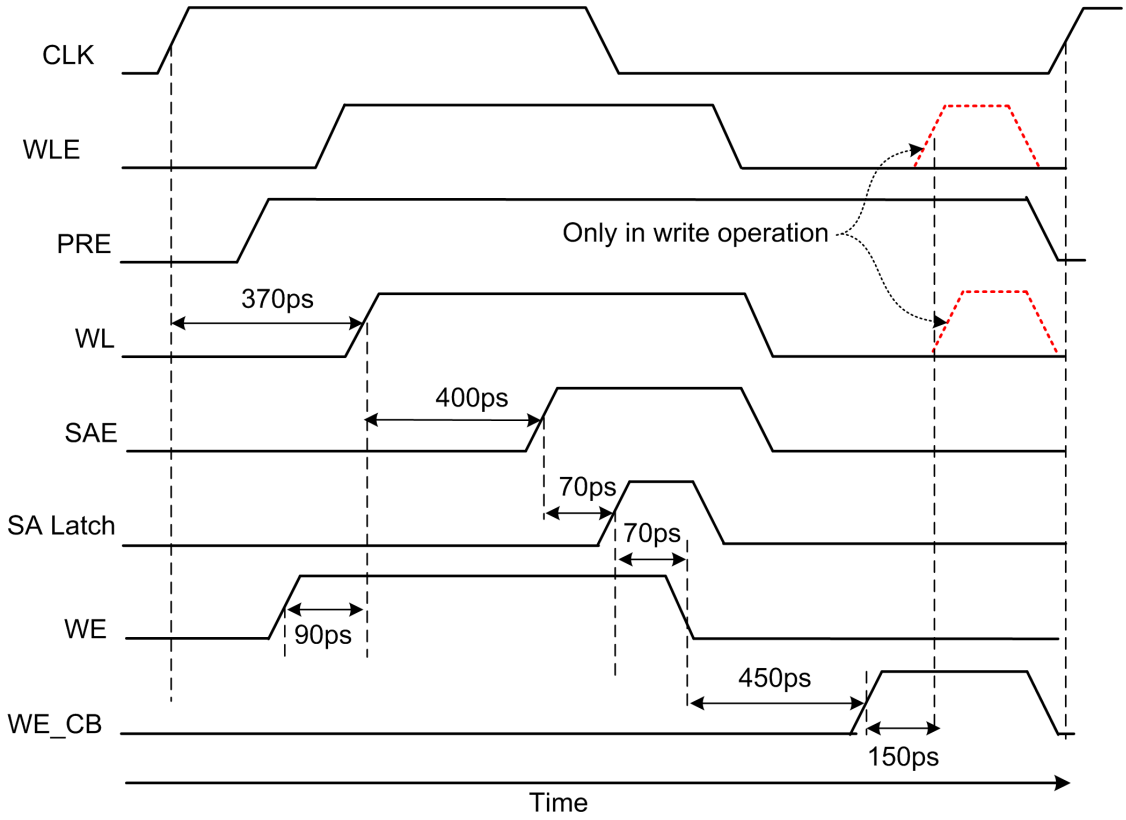


Figure 6.21: Local timing diagram of control signals in the MECC SRAM.

6.2.9 Layout and Simulation

While doing the layout and chip-level simulations, we have used the experience gained from the first chip that we designed and taped out in 180nm technology in November 2006 (CMC Run Code ICFWTSJ1). The micrograph of the chip is shown in Appendix C. The chip implemented a 128-bit ECC logic circuit using the Hamming code. However, due to an unintentional overlap of a top metal line (M6) with the neighbouring block that shared the same silicon area and hence the IOs, the functionality of the ECC logic could not be completely tested. Since this unfortunate incident, we have been extra careful throughout the design and layout of the MECC protected SRAM, which also shares the silicon area on the chip with two other SRAM macros.

The layout of the MECC SRAM is done hierarchically. That is, a block is first designed and laid out and then the next block that uses the first block is designed and laid out. For example, the cell is first designed and laid out to generate the array. The array defines the WL and BL pitches within which the WL driver (and post decoder) and

column MUX, respectively, must fit. Similarly, the pre-decoder based row and syndrome decoders are designed. Every block is optimized through schematic simulations, laid out, and then extracted to perform post layout simulation. Post layout simulation is very critical to ensure the operation of a block within timing constraints. If a block satisfies the timing constraints at minimum power, we design the next block.

It should be mentioned here that the sizing of the SRAM cell is optimized in several iterations. First, the CR and PR are properly chosen with minimum possible transistor dimensions to meet read and write constraints, respectively. Then the cell is laid out and a column is constructed. The column is then extracted to determine the bit line capacitances, C_{BL} and C_{BLB}. The larger of C_{BL} and C_{BLB} is taken to simulate the cell in the schematic again. If the bit line differential voltage, ΔV_{BL}, generated by the cell in a given time, Δt, is not sufficient, we increase the width of the driver transistor in the cell. Typically, ΔV_{BL} can be approximated as

$$\Delta V_{BL} = \frac{I_{READ}}{C_{BL}} \Delta t. \tag{6.1}$$

Upon increasing the width of the driver transistors, we adjust the size of the load and access transistors to meet CR and PR requirements. We continue the iterations until we achieve a ΔV_{BL} of 150 mV.

Once all blocks are designed, we connect them to complete the schematic of the SRAM. The SRAM is then simulated for functionality. However, due to the large number of transistors in the array, such a simulation is extremely time consuming. In order to limit the simulation time, we remove all the rows except one in the array. On every column, we put the lumped capacitances C_{BL} and C_{BLB} obtained from the extraction of a column. These capacitances enable mimicking the bit line load in the absence of all rows. In addition to these capacitances, we use lumped capacitances from 5 fF to 25 fF between blocks to represent the interconnect capacitances. Then the row and peripheral circuits are simulated with 350 MHz clock and 0.4 V V_{GND}.

In a read operation, one of the two composite words in the selected row is read and ECC is performed. If no error is found in ECC operation (first clock cycle in Figure 6.22), the composite word is multiplexed to provide 32 bit data. If an error (two errors) is found, ECC corrects (detects) the error and sets SBE (DBE) signal high, as shown

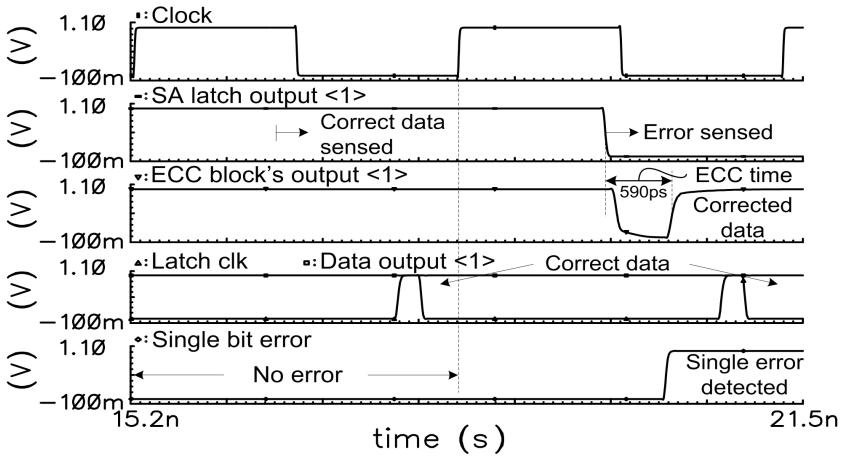


Figure 6.22: Simulated waveforms for two read cycles in the MECC SRAM.

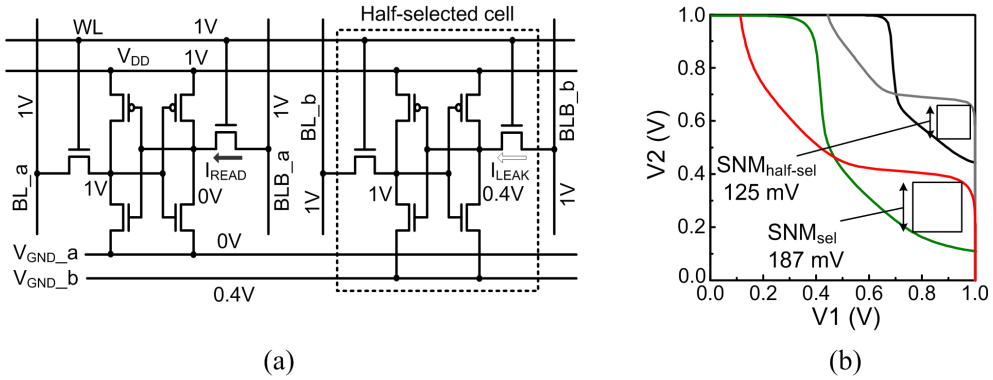


Figure 6.23: a) Adjacent selected and half-selected cells in the accessed row and b) voltage transfer characteristics and SNM of these cells.

in the second clock cycle in Figure 6.22. It should be noted that, when the WL signal is asserted in the read access, the access transistors are turned on in both composite words in the row. However, depending on the address input (A8 to be specific), V_{GND} is lowered for only one composite word, which we refer to as the *selected* composite word. The other one is referred to as the *half-selected* composite word. Figure 6.23(a) shows two adjacent cells that belong to selected and half-selected composite words. In a read access, V_{GND} of the selected composite word is pulled down to 0 V to maximize the read current I_{READ}. For the half-selected composite word, V_{GND} stays at 0.4 V as shown in Figure 6.23(a). Data on this half-selected composite word remain stable as the gate-to-source voltage of access transistors (V_{GS}-access) cannot go below V_{TH}, thus ensuring an SNM of 125 mV (Figure 6.23(b)). In addition, due to high V_{TH} (body effect) and small

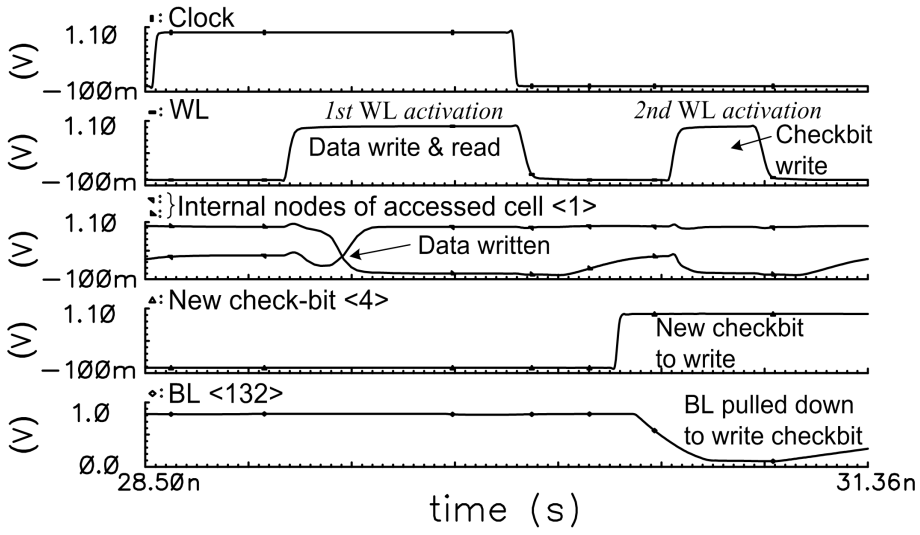


Figure 6.24: Simulated waveforms for a write cycle in the MECC SRAM.

overdrive ($V_{GS-access}$-V_{TH}), the current through access transistor, I_{LEAK}, is small. This saves power by avoiding BL discharge (and subsequent pre-charge) for the half-selected composite word. Thus, MECC successfully uses two composite words per row, eliminating the limitation of the "sense-amplifying-cell" scheme that has only one word per row [66].

In a write operation, WL signal of the decoded row is activated twice as shown in Figure 6.24. In the first WL activation, new data is written into the target word and sense amplifier is enabled to read the new word and the other three words in the composite word. Then, ECC generates new check-bits, which replace stored check-bits in the second WL activation.

Since the read and write functionality with a row is successfully completed, we put the entire array back and complete the schematic of the 64-kb SRAM macro. Then, we proceed to the layout of the SRAM. Upon finishing the layout and performing the design rule check (DRC), we verify the layout versus schematic (LVS). The layout of the SRAM occupies an area of 815 μm $\times$ 500 μm on the chip and shares the IOs with two other SRAMs on the chip. Finally, we send the layout of the chip for fabrication to Canadian Microelectronics Corporation (CMC). Figure 6.25 shows the chip micrograph and block diagram of the MECC-protected SRAM. The run code for the chip is ICLWTSJ2.

It should be noted that despite using 128-bit word for the ECC operation similar to the "alternate ECC" architecture reported in [64], the proposed MECC architecture has

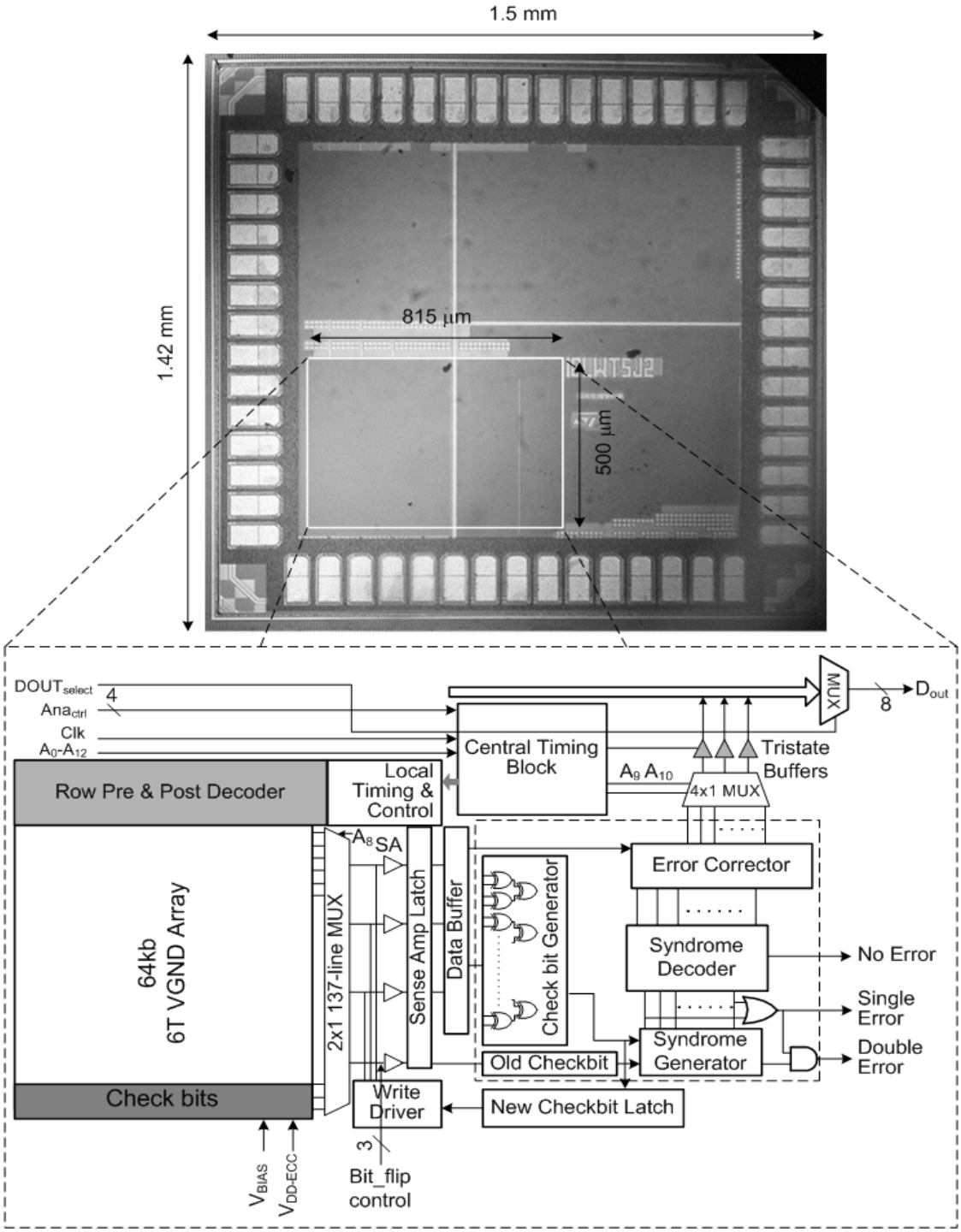


Figure 6.25: Chip micrograph and block diagram of the 64-kb MECC-protected SRAM.

a number of key differences. First, MECC uses 9 check-bits instead of 10 check-bits to provide the same error correction/detection capability. The composite word consists of multiple 32-bit data words instead of 16-bit words. Second, MECC uses an optimized 4-input XOR logic, which significantly reduces read delay and power penalty. Third, MECC controls only the virtual ground potential of composite words instead of controlling WL and BL voltages, thus requiring only one on-chip voltage reference and less control circuitry. In particular, the "alternate ECC" architecture precharges BL and BLB from 1.0 V to 1.5 V during a read access. Since BL and BLB are highly capacitive, this technique significantly increases the read delay penalty and power consumption.

6.2.10 PCB Design

We have designed a four layer PCB to perform various measurements on the test chip. The first or top layer is a signal layer, the second layer is power (V_{DD}), the third layer is ground (V_{SS}) and the fourth or bottom is the other signal layer. Such an arrangement of power and signal layers enables routing of most of the signals on the bottom layer closest to the ground layer and having higher component density on the top layer. The PCB with the test chip and necessary components is shown in Figure 6.26.

The test chip is packaged in a 64-pin ceramic quad flat package (CQFP). The chip layout, bonding diagram, and the pin description are presented in Appendix C. The reference voltages (Vd_BSE, Vw_WLE, VDD_ECC_SRAM, etc.) for the chip are generated using potentiometers on the PCB. Large coupling capacitors are added with these potentiometers to minimize the supply noise. The address inputs (A0-A10), read/write write control (A10), data in (A11), memory select signals (MS_SW1 and MS_SW0), and data output MUX controls (S1_DOUT and S0_DOUT) are provided by slide switches as well as through flat/ribbon cable sockets. The ribbon cable sockets enable remotely controlling these signals using automated test equipment like data generator. Since the designed MECC-protected SRAM shares IO with other SRAMs on the same chip, there is an output MUX with latches. As a result, we cannot measure the operating speed of the MECC-protected SRAM from the data outputs (DOUT$< 0-7 >$). To resolve this limitation, we have dedicated a pad, TP_ECC_SRAM, which is connected directly to the output of the MECC block. This pad is accessed using an SMA connector on the PCB.

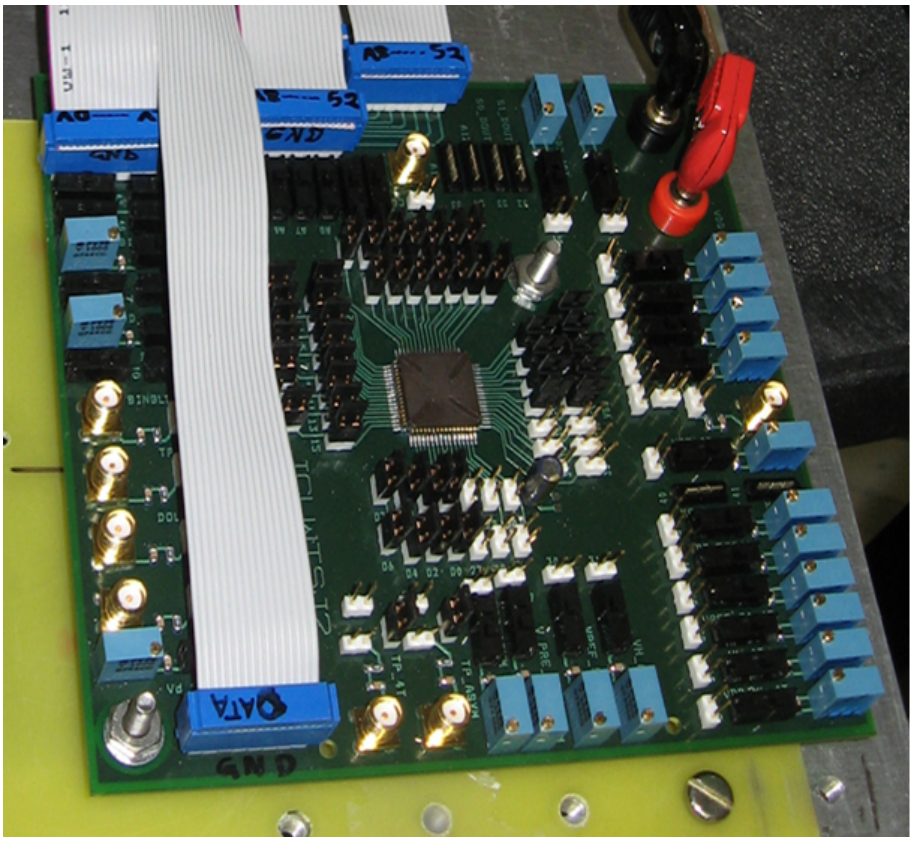


Figure 6.26: PCB for test chip measurements.

Similarly, the clock signal is provided using an SMA connector as well as a pin on the ribbon cable socket.

6.3 Chip Testing

The chip has been tested in two phases. The first phase of testing consists of power and performance measurements at the test lab of CMOS Design and Reliability Group at the University of Waterloo. The second phase of testing comprises of soft error performance measurements under neutron radiation at Canada's National Laboratory for Particle and Nuclear Physics, referred to as TRIUMF, located in Vancouver, BC.

6.3.1 Power and Performance Test

In power-performance tests, we measure the active and leakage power consumption of the chip and measure the operating speed. Accordingly, we use following pieces of test equipment:

- a Precision DC Power Supply (Agilent E3631A) to supply power to the PCB,
- a Data Generator (Tektronix DG2020A) to supply the clock and address signals,
- a Logic Analyzer (Tektronix TLA5201) for reading and evaluating the data outputs,
- a Digital Oscilloscope (LeCroy WaveRunner 6100) for observing clock and output signals, and
- a Precision Multimeter (Fluke 189) for DC voltage and current measurements.

The test chip has a separate pin (VDD_ECC_SRAM) to supply power to the MECC-protected SRAM so that the leakage and active power consumptions can be easily measured. Accordingly, we use the multimeter as an ammeter in series with this pin to monitor the average current drawn by the SRAM unit. In order to measure the active power, we clock the SRAM and observe the ammeter current in read and write mode. However, since the maximum frequency of the data generator is limited to 100 MHz while the SRAM was designed to operate at 350 MHz, we reduce the supply voltage of the SRAM and test its operations at 100 MHz. In particular, we set V_{DD}=0.8 V and V_{GND}=0.4 V, keeping a rail-to-rail voltage of 0.4 V in the SRAM cells. Thus, if the ammeter current is given by $I_{DC-READ}$ and $I_{DC-WRITE}$ for read and write modes, respectively, the corresponding power can be calculated as

$$P_{READ} = I_{DC-READ} \times 0.8 \tag{6.2}$$

and

$$P_{WRITE} = I_{DC-WRITE} \times 0.8. \tag{6.3}$$

The average active power, P_{AVG}, is then calculated from the average of (6.2) and (6.3). The average active energy is simply given by,

$$E_{AVG} = P_{AVG} \times T_{clk}, \tag{6.4}$$

where T_{clk} is the period of the clock and is, therefore, 10 ns for 100 MHz clock.

The leakage power of the SRAM is measured by setting the clock to 0 V. In this condition, if the ammeter reads $I_{DC-LEAK}$, the leakage power is given by

$$P_{LEAK} = I_{DC-LEAK} \times 0.8. \tag{6.5}$$

The operating speed of the SRAM is measured by observing the clock and the output signal at TP_ECC_SRAM on the oscilloscope. This output signal comes directly from the MECC block after error correction. Thus, the dealy between the rising edges of this signal and the clock gives the data latency of the SRAM, and hence defines its speed. Since we have used a large buffer to drive the pad TP_ECC_SRAM, we exclude the delay of the buffer (d_{buffer}) from the delay measured from the oscilloscope (d_{osc}). Thus, the data latency of the SRAM is given by,

$$d_{data} = d_{osc} - d_{buffer}. \tag{6.6}$$

We have estimated d_{buffer} of approximately 300 ps from the post-layout simulation of the buffer. Accordingly, if d_{osc} is measured in ns, (6.6) reduces to

$$d_{data} = d_{osc} - 0.3\ ns. \tag{6.7}$$

6.3.2 Radiation Test

Upon successful completion of power and performance test at the University of Waterloo, the soft error performance of the chip has been tested at TRIUMF. In particular, the chip has been subjected to accelerated neutron radiation at the TRIUMF Neutron Facility (TNF) according to the Joint Electron Device Engineering Council (JEDEC) standards [68]. The neutron beam has the same energy spectrum as the atmospheric neutrons as shown in Figure 6.27. However, the beam's average fluence is $2.639 \times 10^6\ n/cm^2 - s$, which is approximately 4.83×10^8 times higher than the neutron fluence at sea level in New York City (NYC). Thus, the neutron beam has enabled measuring realistic cosmic neutron-induced SER at a shorter irradiation time.

The test facility and equipment setup at TNF is shown in Figure 6.28. The radiation test procedure that we have followed in order to conform to the JEDEC standard is summarized below:

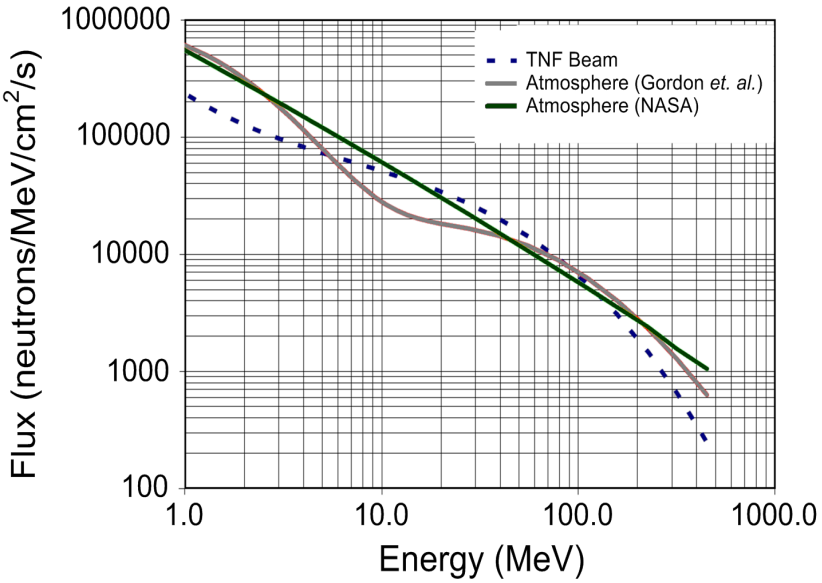


Figure 6.27: Neutron spectrum at TNF compared with the atmospheric spectrum from Gordon et al. (IEEE Trans. Nucl. Sci., vol. 51, page- 3427, 2004) and NASA. Reproduced with permission from TRIUMF.

1. Set up equipment and ensure the connectivity. Since the PCB is about 30 feet away from the test equipment, make sure that the supply voltage and V_{GND} at the PCB are as desired.

2. Perform Read/Write test with neutron beam on, but PCB out of the beam.

3. Place the PCB on the path of the beam by releasing the wire of the pulley on the PCB mounting station. Note the neutron fluence at the Neutron Monitor.

4. Write '1' to entire address space. Read entire address space to make sure that all writes were correct.

5. For two hours, read the entire address space every 30 minutes using the Logic Analyzer and Data Generator. If there are any errors (1→0) over this time, the erroneous data are captured by the Logic Analyzer. We analyze the data and count the errors. We refer these errors as *total errors* (1→0).

6. After two hours of data acquisition, take the PCB out of the neutron beam by pulling the wire of the pulley on the PCB mounting station.

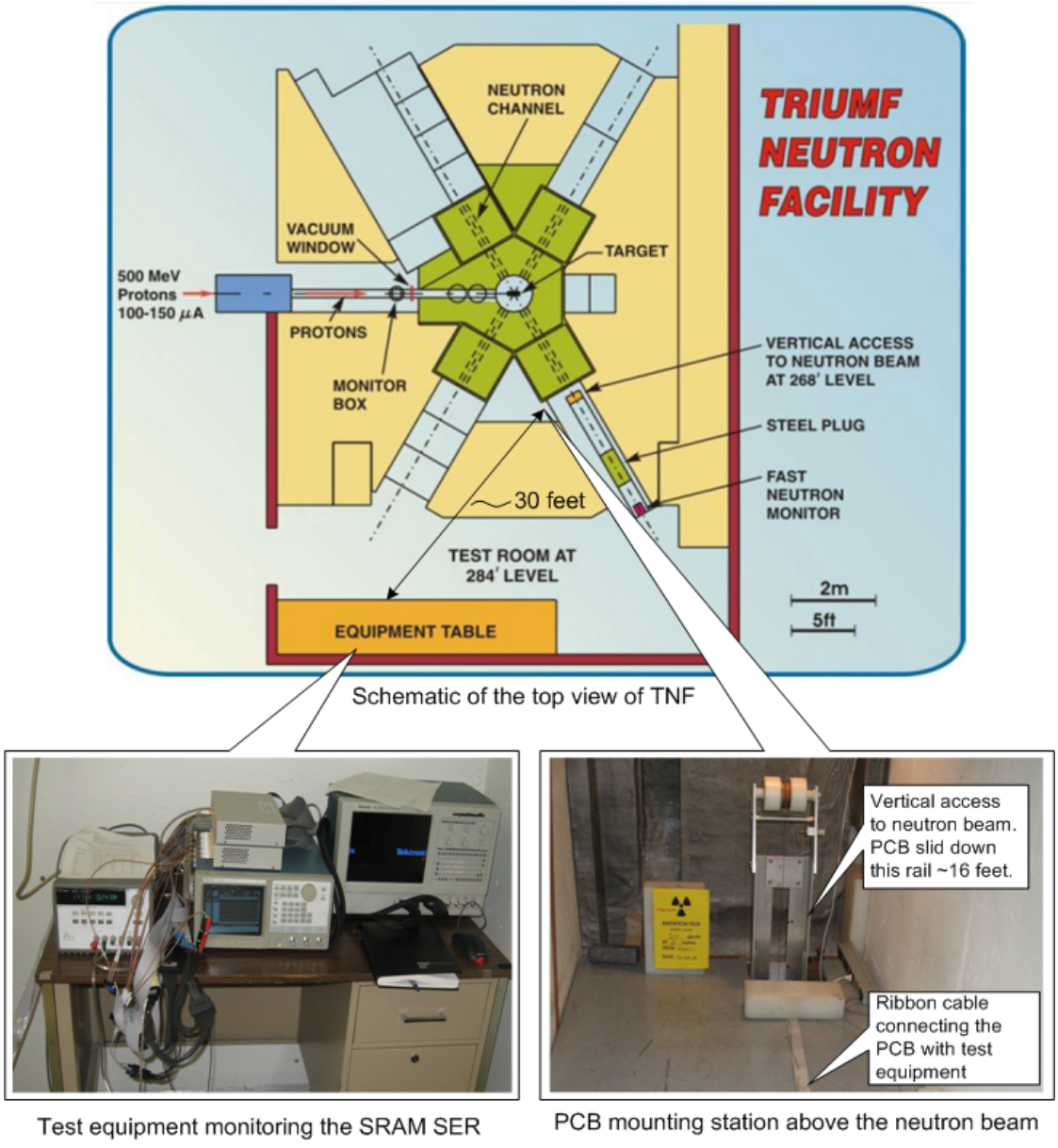


Figure 6.28: Schematic of the TNF and test equipment setup for SER measurements. TNF schematic is reproduced with permission from TRIUMF.

7. Write '0' to the entire address space and read . Then, write '1' to the entire address space again and read. If there are some 0s, some *hard errors* (0) have occurred.

8. Subtract the *hard errors* (0) from the *total errors* to find the soft errors (1→0) in 2 hours.

9. If there is zero *hard errors*, use the same PCB for the next test. Otherwise, use another PCB.

10. Repeat steps 2 through 8 with complementary write operations to find the soft errors (0→1) in 2 hours.

Since we have not found any hard errors, we have used the same PCB throughout the radiation test. In addition, since the numbers of soft errors for 1→0 and 0→1 transitions were similar in the initial tests, we measured only 0→1 soft errors in the later parts of the test where we varied the supply voltage of the SRAM. The reason for having a similar number of soft errors for the 1→0 and 0→1 transitions can be attributed to the fact that an SRAM cell stores both the datum and its complement and that the cell is symmetric.

It should be noted that the counting of errors has not been done on the TNF site. In fact, in every 30 minutes the Logic Analyzer stores the data, which are analyzed at the end of the experiment to count the soft errors. The Logic Analyzer stores data in both graphic and text formats. Figure 6.29 shows a sample of the graphical data acquisition performed by the Logic Analyzer. Upon counting the number of errors, the SER in FIT (1 error in 10^9 hours of device operation at the ground level atmospheric neutron flux) is calculated using the following equation.

$$SER = \frac{1}{a_t a_n} \times 10^9 \times Errors, \tag{6.8}$$

where a_t is the time acceleration factor or the time of neutron irradiation, a_n the neutron fluence acceleration factor, and *Errors* the counted data bit errors. The value of a_n is computed in the following way:

$$Bombarded\, neutrons\, in\, 2\, hours = CF \times \frac{NM\ Reading\ without\ PCB}{NM\ Reading\ with\ PCB} \times Counted\, Neutron, \tag{6.9}$$

where CF is the TNF calibration factor approximated as 2.73×10^3, $NMReadings$ the 10 second average readings of the neutron monitor, and $Counted\ Neutron$ the cumulative

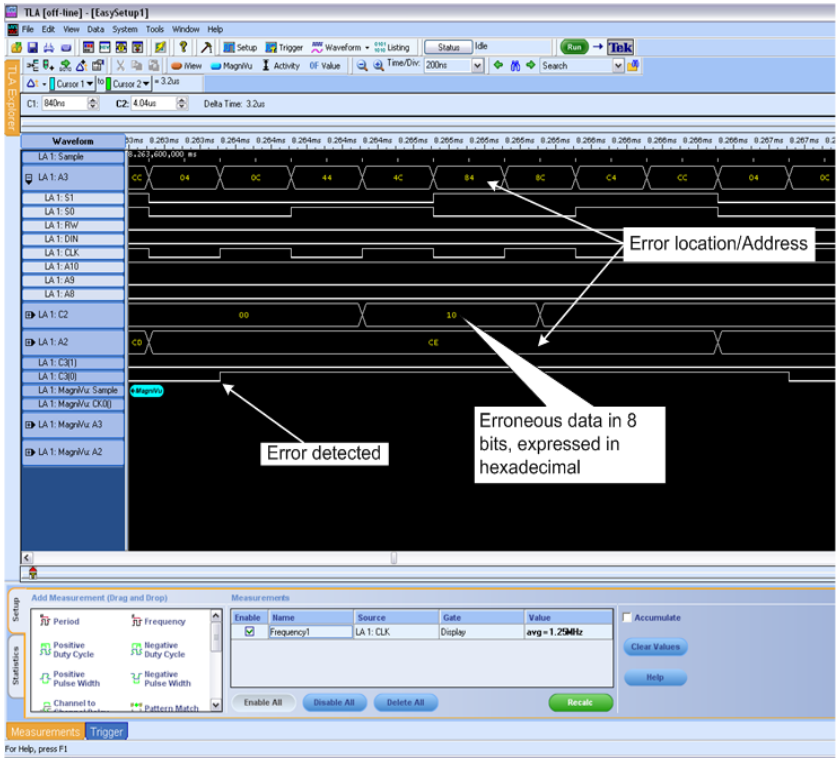


Figure 6.29: View of the Logic Analyzer screen showing the clock, address, data with error, and the error signal.

neutron count per cm^2 counted by the neutron monitor. Substituting the values of these parameters for a given measurement, we get from (6.9)

$$Bombarded\ neutrons\ in\ 2\ hours = 2.73 \times 10^3 \times \frac{10421.33}{8485.33} \times 5765400 = 1.9331 \times 10^{10}\ /cm^2. \tag{6.10}$$

Since fluence is defined as the number of neutrons per unit area (say, cm^2) per unit time (say, 1 h), (6.10) can be used to compute the fluence for the meaurement:

$$Fluence\ at\ TRIUMF = \frac{1.9331 \times 10^{10}}{2} = 9.6655 \times 10^9\ \ n/cm^2 - h. \tag{6.11}$$

The atmospheric neutron fluence at NYC is 20 $n/cm^2 - h$. Therefore, the neutron acceleration factor is given by

$$a_n = \frac{Fluence\ at\ TRIUMF}{Fluence\ at\ NYC} = \frac{9.6655 \times 10^9}{20} = 4.8328 \times 10^8 \tag{6.12}$$

Once a_n is known and the number of bit errors is counted from the Logic Analyzer data, the SER in FIT is computed using (6.8).

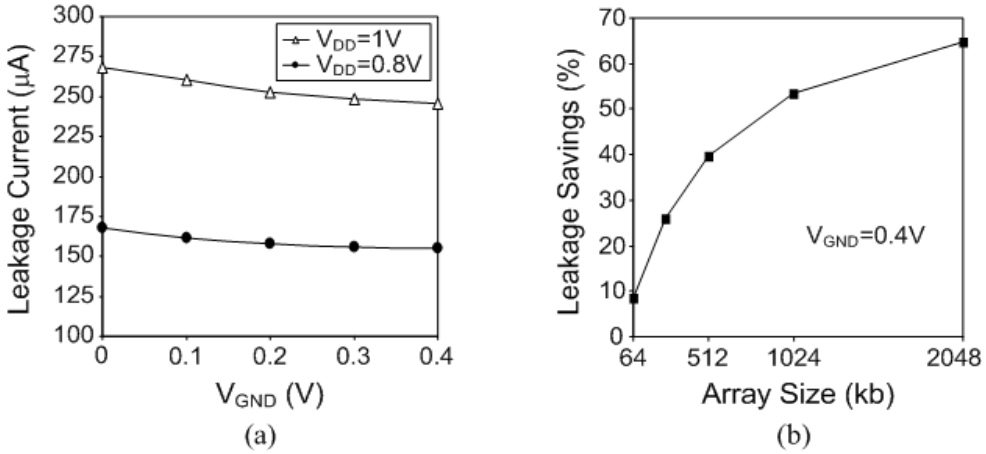


Figure 6.30: a) Measured chip leakage at different V_{GND} and supply voltages, and b) predicted leakage saving for larger arrays.

6.4 Test Results and Discussion

Figure 6.30(a) shows measured chip leakage at different V_{GND}. As is evident from Figure 6.30(a), the leakage saving is only 8% when raising V_{GND} from 0 V to 0.4 V. This may appear to dispute the effectiveness of the proposed V_{GND} scheme with the MECC architecture. In fact, the V_{GND} scheme does indeed reduce the array leakage current by 82%, however, the reduction does not appear at the chip-level leakage measurement. This is because the measured array leakage at 1 V is 16.2 μA, which is only 10.5% of the total chip leakage. The chip leakage is dominated by the peripheral circuits (decoders, ECC logic, buffers, etc.), which use standard-V_{TH} transistors. For a larger array, the array leakage will be a bigger fraction of the chip leakage as the leakage in the periphery circuits does not proportionally increase. Consequently, the leakage current savings with increasing array size will be higher. Figure 6.30(b) shows the leakage current savings as a function of array size with a V_{GND} of 0.4 V.

Figure 6.31 shows measured chip power components. As can be seen, compared to using a conventional 0 V V_{GND}, using a 0.4 V standby V_{GND} reduces both read and write power. The power reduction primarily stems from reduced bit line discharge due to higher V_{GND} of un-accessed composite word in read/write access. The bit line discharge is further limited by quick deactivation of the WL signal after writing into an accessed data word in a write cycle. The measured read and write powers of the chip are 396 μW

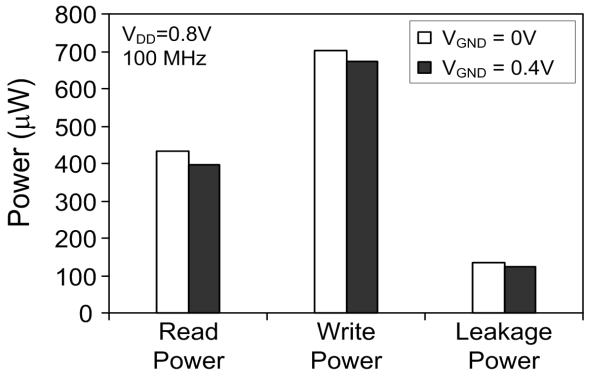


Figure 6.31: Measured chip power components at different V_{GND}.

and 672 μW, respectively. The average active power is, therefore, 534 μW, implying an average active energy consumption of 5.34 pJ. The write power is higher since the writing in the proposed SRAM is, in fact, a combination of read and write.

The read data latency of the MECC SRAM chip is 3.3 ns with an SBE inserted by a set/reset switch. This latency is 8 times smaller than the latency (27 ns) of a similar 128-bit ECC reported in [64] and 15% larger than the latency (2.86 ns) of the 64-bit ECC reported in [63]. However, the proposed SRAM does show a latency of 2.85 ns in simulation with V_{DD} =1 V (see Figure 6.22). Furthermore, [63] uses single 64-bit data word, thus not spending any time in multiplexing the final data output. Table 6.2 shows a comparison of the proposed SRAM with [64] and [63]. Since these schemes have different array sizes, we convert their energy down to the bit level. Accordingly, we find that the proposed SRAM has 82% less per-bit energy consumption and 5.5% less check-bit overhead than [63], and similar energy efficiency and check-bit overhead as [64]. However, [64] uses a variety of voltage references (0.5 V V_{GND}; 1 V bit line voltages, 1.5 V internal, 3.3 V external), which warrant a number of on-chip voltage converters and additional timing and control signals. Furthermore, [64] has a large (16 Mb) array, which makes the per-bit energy consumption so small. Had we implemented the MECC scheme with a similar sized array, the per-bit energy consumption of the MECC scheme would have been much smaller since the power consumption of peripheral circuits does not proportionally increase with increasing array size.

Table 6.3 presents the radiation test data and the SER performance of the chip for varying supply and array ground voltages. Since the chip is irradiated for 2 hours for

Table 6.2: Chip Measurement Results and Performance Comparison - 1

	This work	Ref. [64]	Ref. [63]
Technology	90nm	130nm	130nm
Array size, word size	64kb, 32b	16Mb, 16b	512kB (=4Mb)
ECC word, check-bits	128b, 9b	128b, 10b	64b, 8b
ECC overhead	7%	7.8%	12.5%
Latency with SBE	3.3ns	27ns	2.86ns (4 cycles @ 1.4GHz)
Average power	534μW @ 0.8V, 100MHz	19mW @ 1.5V internal, 3.3V external, 14.3MHz	2.6W @ 1.3V, 1.4GHz
Average energy	5.34pJ	1330pJ	1857pJ
Avg. energy per bit	0.08fJ	0.08fJ	0.44fJ

each supply-ground voltage combination, the time acceleration factor a_t is always 2 hours. However, the neutron acceleration factor a_n varies due to a variation in the neutron fluence, which results from the fluctuations in the current of the proton source at TRIUMF. Therefore, a_n is calculated using (6.12) for each supply-ground voltage combination. Then the substitution of "total errors" in (6.8) yields the "SER without MECC" while the substitution of "multi-bit error" in (6.8) yields the "SER with MECC". The "SER without MECC" thus obtained is 176 FIT for the 64-kb chip or 2816 FIT/Mb at 1 V. The typical commercial SRAM in 90nm technology exhibits an SER of 2000 FIT/Mb [69].

Figure 6.32 summarizes the SER performance of the chip. As evident, the proposed SRAM has an SER of only 57 FIT, which is 68% less than a conventional SRAM (V_{GND}=0 V) and 85% less than a low-power SRAM having V_{GND}=0.4 V and no ECC. More reduction (99.5%) in SER has been reported in [64]. However, the additional SER reduction can be attributed to fewer multi-bit errors. Ref. [64] is implemented in a 130nm process while the proposed SRAM is prototyped in a 90nm process. As a result, [64] has a larger critical charge, which limits the number of multi-bit errors from a single particle strike and helps the ECC to reduce the chip SER. Since the error correction capability of [64] and the proposed SRAM is the same, similar SER reduction can be achieved in the

Table 6.3: Soft Error Rate Calculation from Radiation Test Data

V_{DD} (V)	V_{GND} (V)	Single-bit error	Multi-bit error	Total errors	Fluence (n/cm^2-h)	$\frac{10^9}{a_n a_t}$	SER without MECC (FIT)	SER with MECC (FIT)
1.1	0	103	11	114	9.5591×10^9	1.046	119	12
1.0	0	152	25	177	1.0047×10^{10}	0.9953	176	25
0.9	0	151	24	175	9.2933×10^9	1.076	188	26
0.8	0	202	48	250	9.1554×10^9	1.0923	273	52
1.0	0.4	302	55	357	9.6655×10^9	1.0346	369	57

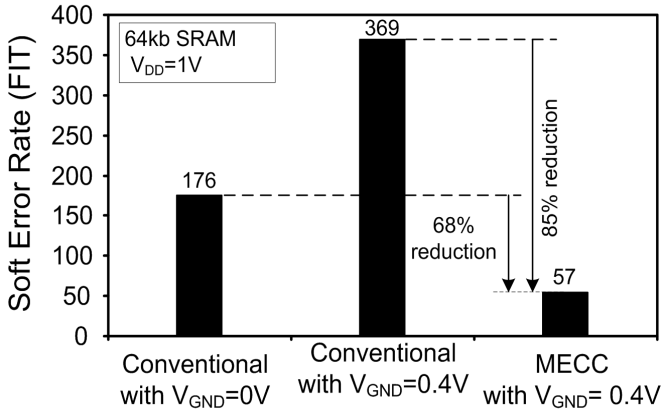


Figure 6.32: Measured chip soft error rate relative to conventional SRAMs.

proposed SRAM if more than two composite words are interleaved in a row.

In order to check the applicability of the MECC scheme to low voltage SRAMs like the drowsy cache, we test the chip by setting V_{GND}=0 V and V_{DD}=0.6 V. Further reduction in V_{DD} is not possible as the global timing block fails below 0.6 V. Thus, at a 0.6 V supply and 100-MHz speed, the SRAM consumes 177 μW of read power and 319 μW of write power. On the other hand, the leakage power consumption is 65 μW and the SER correction capability is 85%. Clearly, the MECC scheme is very energy efficient and hence suitable for soft error mitigation in low-voltage SRAM. Furthermore, the scheme would not require routing any virtual ground line and associated control circuits, thus saving area.

In fact, the MECC scheme with V_{DD}=0.6 V is more energy-efficient than the MECC scheme with V_{GND}=0.4 V and V_{DD}=1 V even though they have the same rail-to-rail

Table 6.4: Chip Measurement Results and Performance Comparison - 2

	MECC at 0.6V and no V_{GND}	Ref. [64]	Ref. [63]
Technology	90nm	130nm	130nm
Array size, word size	64kb, 32b	16Mb, 16b	512kB (=4Mb)
ECC word, check-bits	128b, 9b	128b, 10b	64b, 8b
ECC overhead	7%	7.8%	12.5%
Latency with SBE	10ns	27ns	2.86ns (4 cycles @ 1.4GHz)
Average power	248μW @ 0.6V, 100MHz	19mW @ 1.5V internal, 3.3V external, 14.3MHz	2.6W @ 1.3V, 1.4GHz
Average energy	2.48pJ	1330pJ	1857pJ
Avg. energy per bit	0.04fJ	0.08fJ	0.44fJ

voltage in SRAM cells. This is because the peripheral circuits, including the ECC logic in the former, operates at V_{DD}=0.6 V. As a result, they consume less active and leakage power. However, due to lower operating voltage, they are slower. Accordingly, the data latency with a single-bit error in the MECC scheme with V_{DD}=0.6 V is 10 ns while the same in the MECC scheme with V_{GND}=0.4 V and V_{DD}=1 V is 3.3 ns. Table 6.4 summarizes the performance of the MECC scheme with V_{DD}=0.6 V and compares it with other multiword ECC schemes.

6.5 Summary

This chapter has presented an SRAM architecture that uses row virtual grounding with a multiword based ECC scheme. The virtual ground reduces the array leakage power by 82% without degrading read speed or SNM, and saves active power by reducing bitline discharge for unselected words. The multiword ECC performs error correction on four 32 bit data words using 9 check-bits, which significantly reduce check-bit area. Thus, the net array area overhead for using virtual ground reduces to 9%. The overhead can be further reduced at the expense of read current and SNM degradation. The read delay

and power due to ECC operation is minimized by using an optimized 4-input TG XOR. Bit interleaving is used in a row to correct multi-bit errors. Measurement results and radiation tests on a 64-kb SRAM show that the proposed architecture exhibits improved area, speed, power, and soft error performance than existing SRAMs with multiword ECC.

Chapter 7

Conclusion

This chapter summarizes the contributions and achievements of this research and outlines the future research directions from this work.

With continuously shrinking transistor dimensions but constant cosmic neutron flux at the ground level, the soft error vulnerability of semiconductor devices is increasing. In particular, SRAM, which uses the smallest possible transistors but occupies the majority of the die area, is becoming the circuit subsystem most susceptible to soft errors. In this research, we have performed an in-depth analysis of the soft error mechanism in SRAMs and proposed reliable and efficient soft error modeling and mitigation techniques. The techniques have been experimentally verified using accelerated neutron radiation tests. The key contributions and possible future work from this research are summarized in the following sections.

7.1 Contributions to the Field

7.1.1 A Comprehensive Critical Charge Model

We have proposed a comprehensive model for the critical charge, which is a key to assessing the soft error susceptibility of SRAMs. Unlike existing critical charge models, the proposed model incorporates the dynamic response of the SRAM cell to an exponential noise current and consists of both NMOS and PMOS transistor parameters. The accuracy

and reliability of the model have been verified with SPICE simulations and accelerated neutron radiation tests.

The proposed model is less less time consuming than SPICE simulations and more accurate than existing critical charge models. Accordingly, it will enable fast but accurate estimation of the soft error vulnerability of SRAM cells at the design stage. In particular, the model can be used to estimate the impact of different leakage reduction techniques like supply voltage reduction, virtual grounding, multi-threshold design, etc., on the soft error rate. In addition, the model can be used to estimate the amount of node capacitance (MIM or 3D) needed to achieve a given soft error rate when one of above leakage reduction techniques is employed.

Since the model incorporates both NMOS and PMOS transistor parameters, it can be used to characterize the variability in the soft error rate due to process variations. In particular, if the variations of a transistor parameter (V_{TH}, L, W, etc.) are known, the resulting impact on the soft error susceptibility can be readily estimated by the proposed model. This underscores the importance of the model since process variation is a growing concern in nanoscale technologies.

7.1.2 Process Dependence of Critical Charge

We have investigated the process dependence of the critical charge and showed how accurately the proposed critical charge model can track the dependencies. We have shown that the critical charge is most sensitive to variations in load transistor parameters and resistive defects in the pull-up path. This is in contrast to the SNM, which is most sensitive to driver transistor parameter variations. Accordingly, we have shown that cells having good SNM can have poor critical charge, thus showing high soft error rates at the consumer end.

7.1.3 Multiword-Based ECC with Virtual Ground Array

We have proposed an energy-efficient multiword based error correction code (MECC) scheme that is coupled with a row-based virtual ground SRAM array. The scheme significantly reduces the check-bit area overhead by combining four 32-bit data words in

the ECC operation while minimizing the check-bit generation time and energy by using transmission gate XOR logic. In addition, the scheme interleaves two ECC words, which we refer to as composite words, in a row for tackling cosmic neutron induced multi-bit errors. The virtual grounding technique, on the other hand, raises the ground potential of un-accessed composite words to reduce the leakage power consumption. Approximately 82% array leakage power is thus saved. Therefore, the proposed MECC-protected SRAM architecture limits the soft error rate in a cost-effective way while saving the leakage power. The efficacy of the proposed architecture is verified with accelerated radiation tests that conform to the JEDEC test standards.

The proposed MECC scheme is attractive in three ways. First, the scheme significantly reduces the check-bit area, which in turn reduces the cost of the silicon area. Second, the scheme adds only one logic stage in the check-bit generator and minimizes the total delay in the check-bit generator by using an energy-efficient XOR logic. Third, the scheme marries ECC with a row-based virtual grounding technique to simultaneously achieve a low soft error rate and low leakage power.

While compared to existing multiword ECC techniques, the proposed MECC scheme is simpler but faster and energy efficient. In particular, the MECC scheme requires fewer control signals and voltage references. Furthermore, unlike the existing multiword schemes, the proposed scheme operates on 32 bit data words, thus being more suitable for 32-bit CPU architecture.

7.1.4 Radiation Test of SRAM

We have performed an industry-standard accelerated neutron radiation test on a 64-kb SRAM macro that implemented the proposed MECC scheme. The test has not only verified the real world performance of the MECC chip, but also given us a practical idea of the test procedure. The latter can be very useful in future soft error research as the test implications are already known.

In this thesis, we have described the step-by-step procedure for radiation testing according to the JEDEC standard. In addition, we have demonstrated the FIT calculation technique from the accelerated test data. Thus, anybody interested in soft error rate measurements should benefit from this thesis.

7.2 Future Work

Since the soft error rate in SRAMs continues to increase with technology scaling, the research on soft error modeling and mitigation will be very crucial. In the following, we outline some future research directions along these lines based on the work presented in this thesis.

The proposed critical charge model coupled with the extracted model parameters for soft error rate estimation can be used to develop an automated soft error rate prediction tool. A computer program in C or Verilog A could be used in this purpose. The tool would enable estimating the change in the soft error rate performance when the cell is designed by varying different transistor parameters. The tool would also be able to provide the soft error rate variability due to process variations, and thus can be very useful in design for manufacturability (DFM).

The insight gained while developing the critical charge model can be used to develop the critical charge expressions for other types of SRAM cells (4T, 5T, etc.). This would enable comparisons of the soft error robustness of those cells with that of the 6T cell under process variations, different operating environments (voltage, temperature, etc.), and power budgets. In addition, the model could be used with the analytical model of SNM, thus enabling simultaneous optimization of the critical charge and SNM.

The critical charge model could also be extended to different processes other than the bulk CMOS process. The silicon on insulator (SOI) process can be a good candidate for investigation.

The MECC scheme could be further optimized for better soft error performance and lower area overhead. The radiation test has shown that the MECC scheme can correct approximately 85% of the soft errors. In order to achieve more correction capability, we could further limit the multibit errors by interleaving more than two composite words. In that case, routing virtual ground lines and placing associated switches for those composite words would incur significant area overhead. Efficient ways need to be devised to minimize this overhead. A possible solution could be using common ground and supply lines for all composite words and operating them (i. e., the array) at low voltage to minimize the leakage power. However, this would cost a power up delay when the array is brought

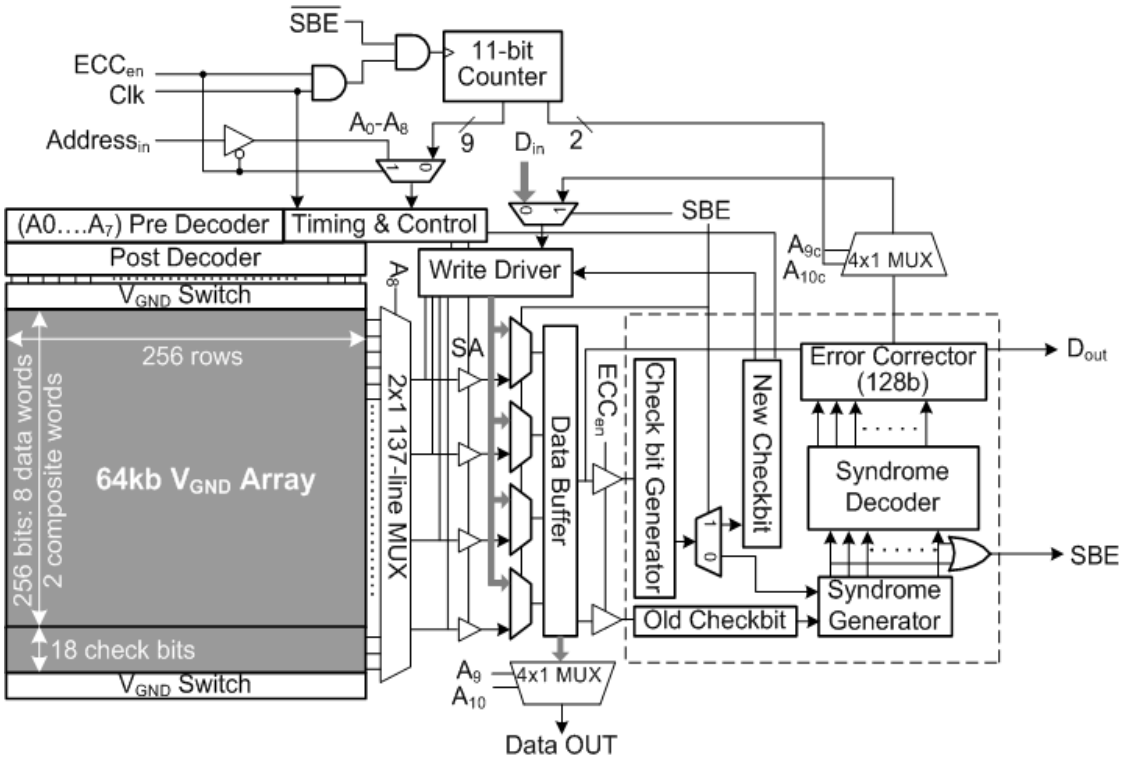


Figure 7.1: An offline MECC scheme for read-delay-free error correction.

back up to the full supply voltage during read access. A trade-off between power saving and delay penalty could be made in this case.

Another way of implementing the MECC scheme could be taking it out of the read data path. The MECC should occasionally operate in the background (like scrubbing) to prevent accumulation of soft errors. In this way, the area saving offered by the MECC can be utilized without any read delay penalty. Figure 7.1 shows such an arrangement, which we refer to as the offline MECC or OMECC scheme. In this scheme, a 11-bit counter generates the addresses to scan the 64 kb array offline. The counter is enabled by the ECC_{en} signal, which disables the original input address and feeds the clock to the counter. If any single bit error (SBE) is detected in the ECC operation, the counter is stopped by disabling the counter's clock signal by the SBE signal. Since the timing and control circuits work on the original clock, they perform error correction in the next cycle taking the address from the stopped counter.

While the scheme offers no read delay penalty and saves significant amount of power by operating only at given intervals, the confidence on the data integrity at the time of

reading can be argued. The OMECC will operate periodically in the background. If an error occurs after the background error correction cycle and before the read operation, the read word will be corrupted. This issue undermines the efficacy of the OMECC scheme and warrants further investigation.

Publications from This Research

Journal Papers

1. S. M. Jahinuzzaman, M. Sharifkhani, and M. Sachdev, "An analytical model for soft error critical charge of nanometric SRAMs," accepted in *IEEE Trans. on Very Large Scale Integration (VLSI) Systems.* July 11, 2008.

2. S. M. Jahinuzzaman, J. S. Shah, D. J. Rennie, and Manoj Sachdev,"Design and analysis of a 5.3-pJ 64-kb gated ground SRAM with multiword ECC," submitted to *IEEE J. Solid-State Circuits*, October 30, 2008.

Refereed Conference Proceedings

1. S. Jahinuzzaman, T. Shakir, S. Lubana, J. S. Shah, and M. Sachdev, "A multiword based high speed ECC scheme for low-voltage embedded SRAMs," in *Proc. IEEE European Solid-State Circuits Conf.*, Edinburgh, UK, September 2008, pp. 226-229.

2. S. M. Jahinuzzaman, M. Sharifkhani, and M. Sachdev, "Investigation of process impact on soft error susceptibility of nanometric SRAMs using a compact critical charge model," in *Proc. Int. Symp. on Quality Electronic Design 2008*, San Jose, CA, pp. 207-212.

3. M. Sharifkhani, S. Jahinuzzaman, and M. Sachdev, "Dynamic data stability in low-power SRAM design," in *Proc. IEEE 2007 Custom Integrated Circuit Conf. (CICC)*, San Jose, CA, pp. 237-240.

4. M. Sharifkhani, S. M. Jahinuzzaman, and M. Sachdev, "Dynamic data stability in SRAM cells and its implications on data stability tests," in *Proc. IEEE Int. Workshop on Memory Technology, Design and Testing 2006 (MTDT'06)*, Taipei, Taiwan, pp. 55-61. (Invited)

Conference Presentations without Proceedings

1. S. M. Jahinuzzaman and M. Sachdev, "Soft errors in nanoscale System-on-Chips," in *Discovery 2008* - annual conference of Ontario Centres of Excellence, Toronto, ON, May 13, 2008.

2. S. M. Jahinuzzaman and M. Sachdev, "Energy and area efficient ECC technique for soft error mitigation and yield enhancement in nanometric SRAMs," in *Discovery 07: To Next* - annual conference of Ontario Centres of Excellence, Toronto, ON, May 1, 2007.

3. S. M. Jahinuzzaman and M. Sachdev, "Estimation of soft error tolerance of nanoscaled static random access memories," in *Graduate Student Research Conference 2006*, University of Waterloo, Waterloo, ON, April 3-6, 2006.

Appendix A

Solving Differential Equation for SRAM Cell Node Voltages

The transient voltage at node A is given by,

$$C_N \frac{dv_A}{dt} = \frac{V_{DD} - v_A}{R_p} - \frac{Q}{\tau_f - \tau_r}\left(e^{-t/\tau_f} - e^{-t/\tau_r}\right), \tag{A.1}$$

Or,

$$\frac{dv_A}{dt} + \frac{1}{R_p C_N} v_A(t) = \frac{V_{DD}}{R_p C_N} - \frac{Q}{(\tau_f - \tau_r) C_N}\left(e^{-t/\tau_f} - e^{-t/\tau_r}\right) \tag{A.2}$$

The solution of (A.2) is of the form:

$$v_A(t) = \frac{\int \mu(t) q(t) dt}{\mu(t)} \tag{A.3}$$

where $\mu(t)$= integration factor $= e^{\int \frac{1}{R_p C_N} dt} = e^{t/R_p C_N}$ and $q(t)$ is the right hand side of (A.2). Therefore, the numerator of (A.3) can be expressed as,

$$\int \mu(t) q(t) dt = \int \left\{ \frac{V_{DD}}{R_p C_N} e^{t/R_p C_N} - \frac{Q}{(\tau_f - \tau_r) C_N}\left(e^{-t/\tau_f} - e^{-t/\tau_r}\right) e^{t/R_p C_N} \right\} dt \tag{A.4}$$

$$= V_{DD} e^{t/R_p C_N} - \frac{Q}{(\tau_f - \tau_r) C_N}\left\{ \frac{R_p C_N \tau_f}{\tau_f - R_p C_N} e^{\left(\frac{t}{R_p C_N} - \frac{t}{\tau_f}\right)} - \frac{R_p C_N \tau_r}{\tau_r - R_p C_N} e^{\left(\frac{t}{R_p C_N} - \frac{t}{\tau_r}\right)} \right\} + B, \tag{A.5}$$

where B is an integration constant. Substituting (A.5) and $\mu(t)$ into (A.3) yields

$$v_A(t) = V_{DD} - \frac{Q R_p}{(\tau_f - \tau_r)}\left\{ \frac{\tau_f}{\tau_f - R_p C_N} e^{-t/\tau_f} - \frac{\tau_r}{\tau_r - R_p C_N} e^{-t/\tau_r} \right\} + B e^{-t/R_p C_N} \tag{A.6}$$

Substituting the initial condition $v_A(0) = V_{DD}$ into (A.6), we get

$$V_{DD} = V_{DD} - \frac{QR_p}{(\tau_f - \tau_r)} \left\{ \frac{\tau_f}{\tau_f - R_p C_N} - \frac{\tau_r}{\tau_r - R_p C_N} \right\} + B. \tag{A.7}$$

Rearranging (A.7) yields,

$$B = \frac{QR_p}{(\tau_f - \tau_r)} \left\{ \frac{\tau_f}{\tau_f - R_p C_N} - \frac{\tau_r}{\tau_r - R_p C_N} \right\}. \tag{A.8}$$

substituting (A.8) into (A.6), we get

$$\begin{aligned} v_A(t) = V_{DD} &- \frac{QR_p}{(\tau_f - \tau_r)} \left\{ \frac{\tau_f}{\tau_f - R_p C_N} e^{-t/\tau_f} - \frac{\tau_r}{\tau_r - R_p C_N} e^{-t/\tau_r} \right\} \\ &+ \frac{QR_p}{(\tau_f - \tau_r)} \left\{ \frac{\tau_f}{\tau_f - R_p C_N} - \frac{\tau_r}{\tau_r - R_p C_N} \right\} e^{-t/R_p C_N} \end{aligned} \tag{A.9}$$

Rearranging (A.9) yields,

$$v_A(t) = V_{DD} - \frac{QR_p}{\tau_f - \tau_r} \left\{ \begin{array}{l} \frac{\tau_f}{\tau_f - R_p C_N} \left(e^{-t/\tau_f} - e^{-t/R_p C_N} \right) \\ -\frac{\tau_r}{\tau_r - R_p C_N} \left(e^{-t/\tau_r} - e^{-t/R_p C_N} \right) \end{array} \right\} \tag{A.10}$$

Appendix B

Q_{crit} Model for Logic '0' Node

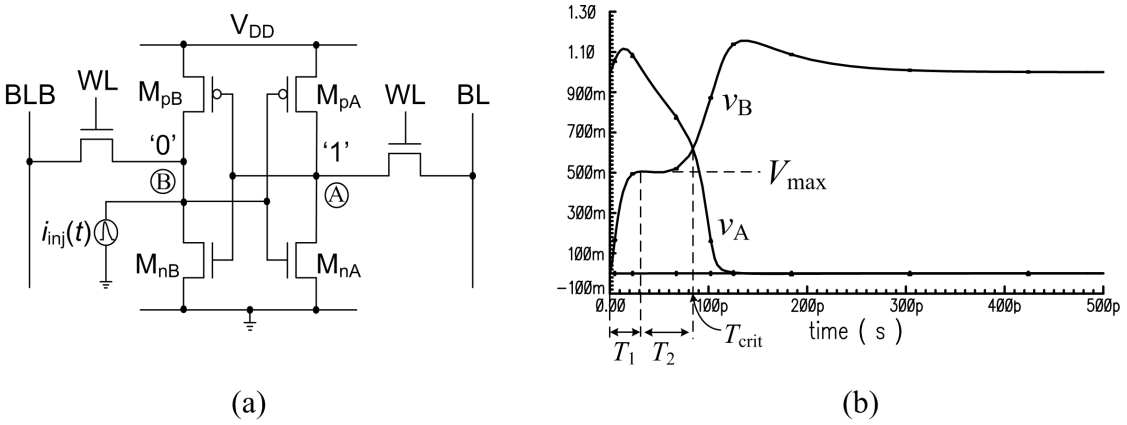


Figure B.1: a) 6T SRAM cell with a noise current injected into node B (logic '0' node) and b) resulting voltage transients at nodes A and B for a state-flipping case.

In order to calculate Q_{crit} for logic '0' node (node B in Figure B.1), we consider the transient resulting from an exponential noise current, $i_{inj}(t)$, injected at the node. The node voltage is given by,

$$C_N \frac{dv_B}{dt} + \frac{v_B}{R_n} = \frac{Q}{\tau} e^{-t/\tau} \tag{B.1}$$

where C_N is the node capacitance, R_n is the ON resistance of transistor, M_{nB}, operating in the linear region, Q the total charge injected by the noise current, and τ the time constant of the noise current. Rearranging (B.1) yields

$$\frac{dv_B}{dt} + \frac{v_B}{R_n C_N} = \frac{Q}{\tau C_N} e^{-t/\tau} \tag{B.2}$$

Solution of (B.2) is of the form:

$$v_B(t) = \frac{\int \mu(t)q(t)}{\mu(t)}. \tag{B.3}$$

Here, $\mu(t) = \text{integration factor} = e^{\int \frac{1}{R_n C_N} dt} = e^{\frac{t}{R_n C_N}}$ and $q(t) = \text{RHS of (B.2)} = \frac{Q}{\tau C_N} e^{-t/\tau}$. Thus, the numerator of (B.3) can be expressed as

$$\int \mu(t)q(t) = \frac{Q}{\tau C_N} \int e^{-\frac{t}{\tau} + \frac{t}{R_n C_N}} dt = \frac{QR_n}{\tau - R_n C_N} e^{-\left(\frac{1}{\tau} - \frac{1}{R_n C_N}\right)t} + B \tag{B.4}$$

where B is an integration constant. Substituting (B.4) and $\mu(t)$ into (B.3 yields

$$v_B(t) = \frac{QR_n}{\tau - R_n C_N} e^{-\frac{t}{\tau}} + Be^{-\frac{t}{R_n C_N}} \tag{B.5}$$

Using the boundary condition $v_B(0) = 0$, we get $B = -\frac{QR_n}{\tau - R_n C_N}$. Substituting B into (B.5) gives the transient voltage at node B:

$$v_B(t) = \frac{QR_n}{\tau - R_n C_N} \left(e^{-\frac{t}{\tau}} - e^{-\frac{t}{R_n C_N}} \right). \tag{B.6}$$

Similar to the noise current injection into node A, the noise injection into node B causes $v_B(t)$ to reach a maximum voltage, V_{max}, at $t = T_1$ and stay at V_{max} for a duration of T_2 when $v_A(t)$ falls from V_{DD}. Thus, the decoupling equations for a state flipping case for a noise current at node B can expressed as

$$\left.\begin{array}{l} 0 \le v_B(t) \le V_{\max} \\ v_A(t) \approx V_{DD} \end{array}\right\} for \;\; 0 \le t \le T_1 \tag{B.7}$$

$$\left.\begin{array}{l} v_B(t) \approx V_{\max} \\ V_{DD} \ge v_A(t) \ge V_{\max} \end{array}\right\} for \;\; T_1 \le t \le T_{crit}, \tag{B.8}$$

where $T_{crit} = T_1 + T_2$.

In order to determine T_1, we differentiate (B.6) and equate to zero to yield

$$-\frac{1}{\tau} e^{-\frac{T_1}{\tau}} + \frac{1}{R_n C_N} e^{-\frac{T_1}{R_n C_N}} = 0$$

Or,

$$e^{\left(-\frac{1}{\tau} + \frac{1}{R_n C_N}\right) T_1} = \frac{\tau}{R_n C_N}$$

Or,

$$T_1 = \frac{\tau R_n C_N}{\tau - R_n C_N} \ln \frac{\tau}{R_n C_N}. \tag{B.9}$$

Substituting (B.9) into (B.6), we get

$$V_{\max} = \frac{QR_n}{\tau - R_nC_N}\left(e^{-\frac{R_nC_N}{\tau - R_nC_N}\ln\frac{\tau}{R_nC_N}} - e^{-\frac{\tau}{\tau - R_nC_N}\ln\frac{\tau}{R_nC_N}}\right)$$
$$= \frac{QR_n}{\tau - R_nC_N}\left(\left(\frac{R_nC_N}{\tau}\right)^{\frac{R_nC_N}{\tau - R_nC_N}} - \left(\frac{R_nC_N}{\tau}\right)^{\frac{\tau}{\tau - R_nC_N}}\right)$$
$$= \frac{QR_n}{\tau - R_nC_N}\cdot\left(\frac{R_nC_N}{\tau}\right)^{\frac{R_nC_N}{\tau - R_nC_N}}\left(1 - \frac{R_nC_N}{\tau}\right)$$

Or,

$$V_{\max} = \frac{QR_n}{\tau}\left(\frac{R_nC_N}{\tau}\right)^{\frac{R_nC_N}{\tau - R_nC_N}} \tag{B.10}$$

Now, in order to find T_2, we consider the transient at node A. The transient voltage at node A can be described as

$$C_N\frac{dv_A}{dt'} + i_n(t') = i_p(t') \tag{B.11}$$

where $t' = t - T_1$ and $i_n(t')$ and $i_p(t')$ are the currents through M_{nA} and M_{pA}, respectively. Since M_{nA} and M_{pA} operate in saturation and linear regions, respectively, we get from (B.11) using the "linear gate model"

$$C_N\frac{dv_A}{dt'} + g_m\left(V_{GS} - V_{THn}\right) = \frac{V_{DD} - v_A}{R_p}$$

Or,

$$\frac{dv_A}{dt'} + \frac{v_A}{R_pC_N} = \frac{V_{DD}}{R_pC_N} - \frac{g_m}{C_N}\left(V_{GS} - V_{THn}\right) \tag{B.12}$$

Similar to (B.2), (B.12) can be solved using the boundary condition $v_A(t') = V_{DD}$ for $t' = 0$ yielding

$$v_A(t') = V_{DD} - g_mR_p\left(V_{\max} - V_{THn}\right)\left(1 - e^{-\frac{t'}{R_pC_N}}\right) \tag{B.13}$$

In order to flip the cell, $v_A(t')$ should equal to V_{max} at $t' = T_2$. Thus, from (B.13) we get

$$V_{\max} = V_{DD} - g_mR_p\left(V_{\max} - V_{THn}\right)\left(1 - e^{-\frac{T_2}{R_pC_N}}\right)$$

Or,

$$e^{-\frac{T_2}{R_pC_N}} = 1 - \frac{V_{\max}}{V_{DD} - g_mR_p\left(V_{\max} - V_{THn}\right)}$$

Or,

$$T_2 = -R_pC_N\ln\left(1 - \frac{V_{\max}}{V_{DD} - g_mR_p\left(V_{\max} - V_{THn}\right)}\right) \tag{B.14}$$

Once T_1 and T_2 are known, T_{crit} and Q_{crit} can be calculated as

$$T_{crit} = T_1 + T_2 \tag{B.15}$$

$$Q_{crit} = \int_0^{T_{crit}} \frac{Q}{\tau} e^{-t/\tau} dt = Q\left(1 - e^{-T_{crit}/\tau}\right) \tag{B.16}$$

Thus, the complete model for Q_{crit} for a particle strike at logic '0' node can be summarized as

$$\left.\begin{aligned} V_{\max} &= \tfrac{QR_n}{\tau}.\left(\tfrac{R_nC_N}{\tau}\right)^{\frac{R_nC_N}{\tau - R_nC_N}} \\ T_1 &= \tfrac{\tau R_nC_N}{\tau - R_nC_N} \ln \tfrac{\tau}{R_nC_N} \\ T_2 &= -R_pC_N \ln\left(1 - \tfrac{V_{\max}}{V_{DD} - g_m R_p (V_{\max} - V_{THn})}\right) \\ T_{crit} &= T_1 + T_2 \\ Q_{crit} &= Q\left(1 - e^{-T_{crit}/\tau}\right) \end{aligned}\right\} \tag{B.17}$$

Appendix C

Details of Test Chips

C.1 Test chip-1: 128 bit ECC Logic

Technology: 180nm CMOS

CMC Run Code: 0604CF

Design Name: ICFWTSJ1

Tape-out Date: November 29, 2006

Test Status: Tested at CDR Group lab

Functionality: Not completely working.

C.2 Test chip-2: MECC Protected 64 kb SRAM

Technology: 90nm CMOS

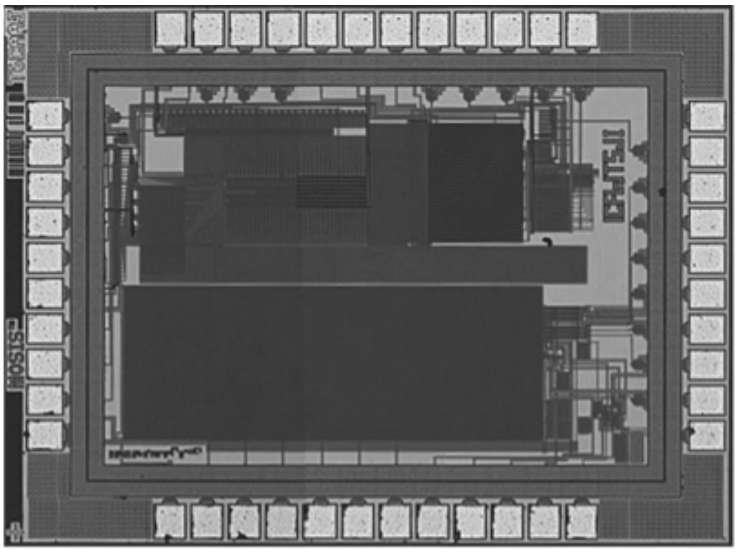
CMC Run Code: 0703CL

Design Name: ICLWTSJ2

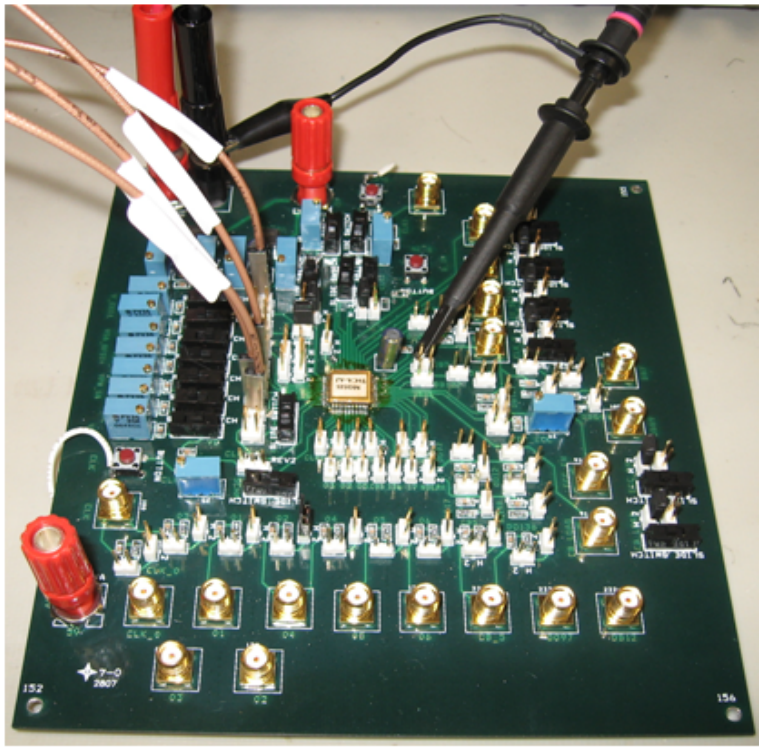
Tape-out Date: July 4, 2007

Test Status: Tested at CDR Group lab and at TRIUMF

Functionality: Completely working. Results presented in this thesis.

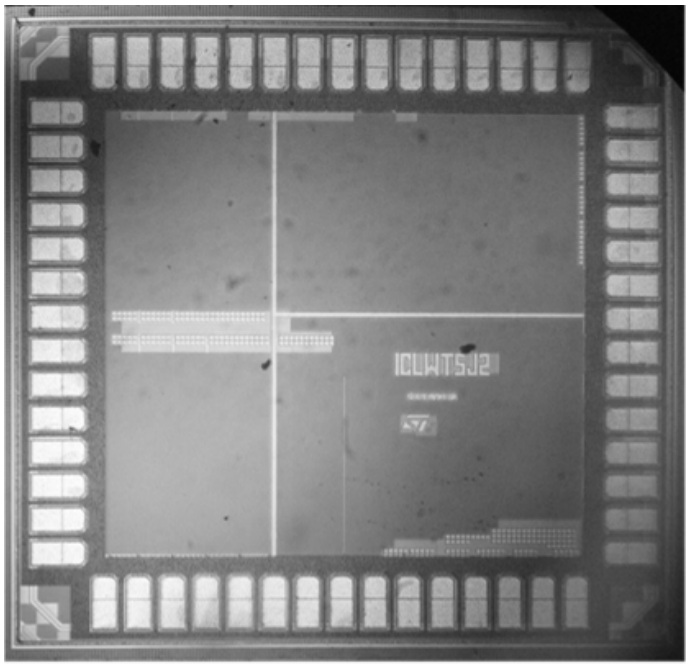


(a)

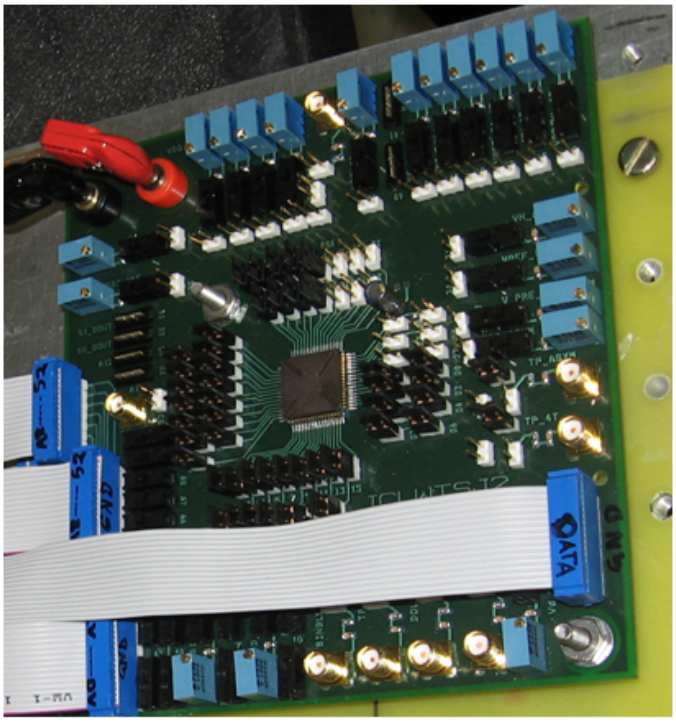


(b)

Figure C.1: a) Micrograph of Test Chip-1 implementing 128 bit data based ECC logic using Hamming Code and b) test board.



(a)



(b)

Figure C.2: a) Micrograph of Test Chip-2 implementing MECC-protected 64 kb SRAM and b) test board.

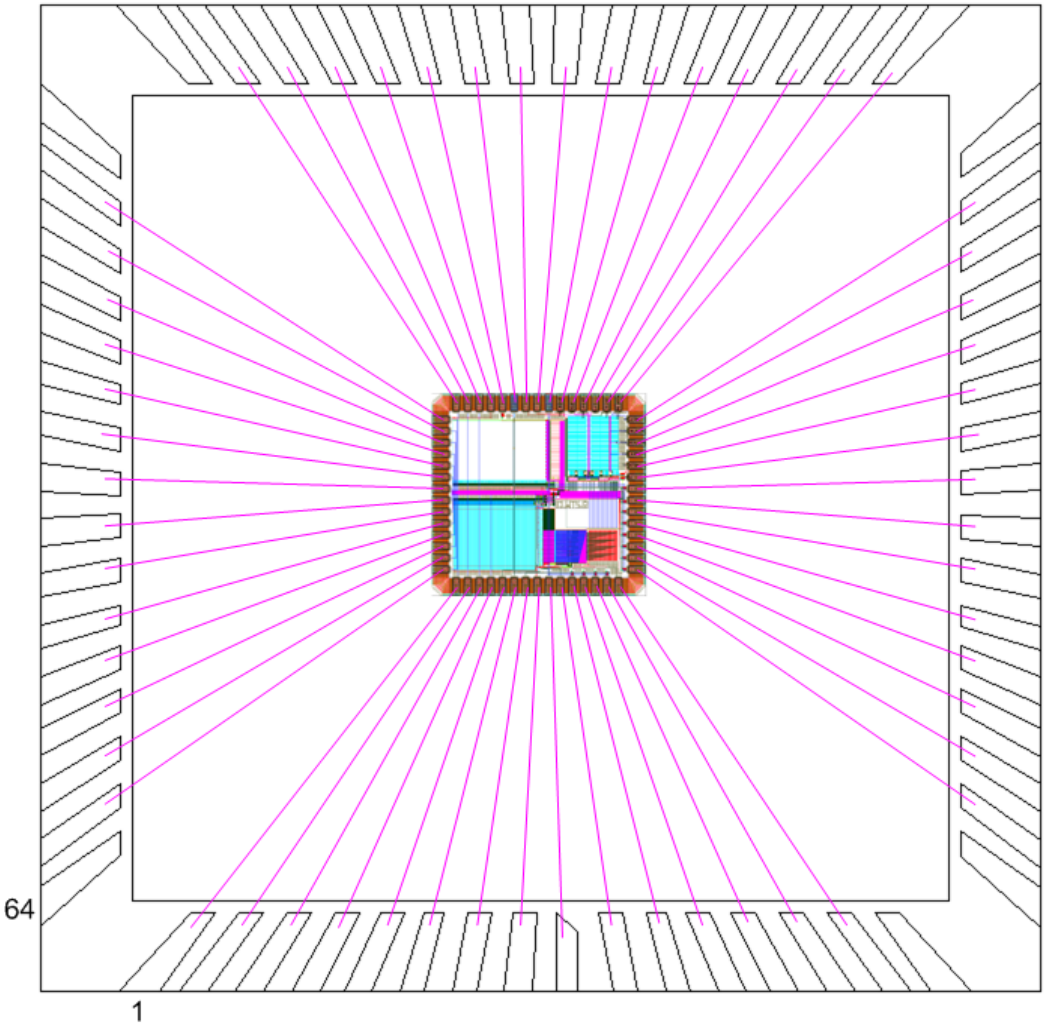


Figure C.3: Bonding diagram of Test Chip-2. Package type: CQFP64.

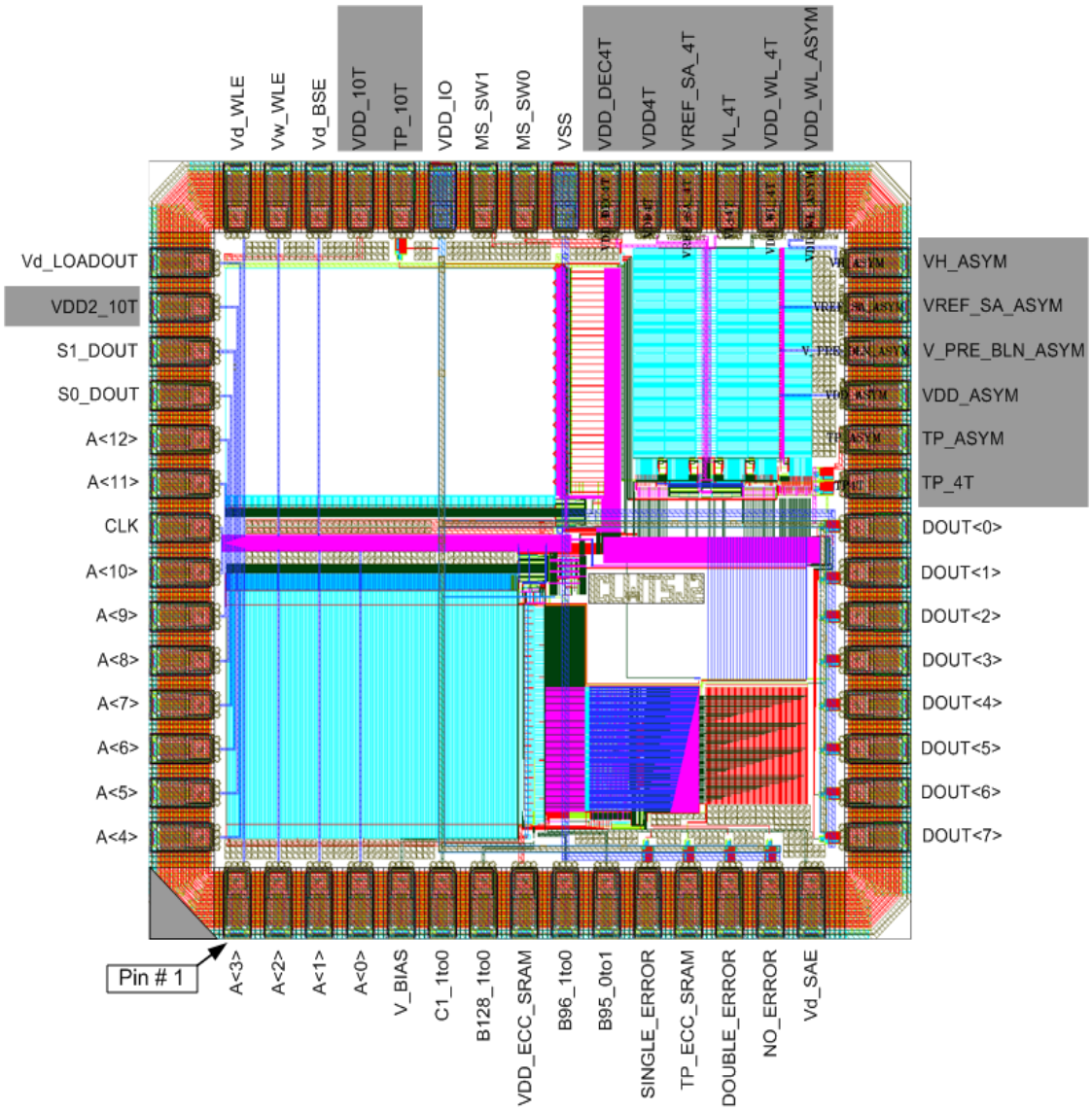


Figure C.4: Pin-out of Test Chip-2. The shaded pins are not related to the testing of the MECC-protected SRAM.

Table C.1: Pin Description of Test Chip-2

Pin No.	Name	Description	Direction
1	A< 3 >	Address	Input
2	A< 2 >	Address	Input
3	A< 1 >	Address	Input
4	A< 0 >	Address	Input
5	V_BIAS	Logic0 voltage for MECC SRAM	Input (DC)
6	C1_1to0	Reset switch for check-bit 1	Input
7	B128_1to0	Reset switch for data bit 128	Input
8	VDD_ECC_SRAM	Power supply to MECC SRAM	Input (DC)
9	B96_1to0	Reset switch for data bit 96	Input
10	B95_0to1	Reset switch for data bit 95	Input
11	SINGLE_ERROR	Single bit error in MECC SRAM	Output
12	TP_ECC_SRAM	Speed test point for MECC SRAM	Output
13	DOUBLE_ERROR	Multi-bit error in MECC SRAM	Output
14	NO_ERROR	No error in MECC SRAM	Output
15	Vd_SAE	Delay control for SA enable	Input (variable DC)
18-25	DOUT< 7 − 0 >	Data output	Output
39	VSS	SRAM and IO ground	Ground
40	MS_SW0	SRAM select switch 0	Input
41	MS_SW1	SRAM select switch 1	Input
42	VDD_IO	Power supply to IO pads	Input (DC)
45	Vd_BSE	Delay control for BSE	Input (variable DC)
46	Vw_WLE	Pulse width control for WLE	Input (variable DC)
47	Vd_WLE	Delay control for WLE	Input (variable DC)
50	Vd_LOADOUT	Delay control for output latch	Input (variable DC)
52	S1_DOUT	Select switch 1 for data-out MUX	Input
53	S0_DOUT	Select switch 0 for data-out MUX	Input
54	A< 12 >	Read(0)/Write(1)	Input
55	A< 11 >	Data in	Input
56	CLK	Chip clock	Input
57-63	A< 10 − 4 >	Address	Input

References

[1] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits - A Design Perspective*. Upper Saddle River, New Jersey: Prentice Hall, 2002. 1, 27, 28, 31, 33, 34, 36, 38, 40, 72

[2] R. C. Baumann, "Soft errors in advanced semiconductor devices - part I: the three radiation sources," *IEEE Trans. Nucl. Sci.*, vol. 1, no. 1, pp. 17–22, Mar. 2001. 2, 4, 5, 7

[3] P. E. Dodd and L. W. Massengill, "Basic mechanisms and modeling of single-event upset in digital microelectronics," *IEEE Trans. Device Mat. Rel.*, vol. 50, no. 3, pp. 583–602, Jun. 2003. 2, 8, 9, 56

[4] R. C. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," *IEEE Trans. Device Mat. Rel.*, vol. 5, no. 3, pp. 305–316, Sep. 2005. 2, 7, 8, 9

[5] T. C. May and M. H. Woods, "Alpha-particle induced soft errors in dynamic memories," *IEEE Trans. on Electronic Devices*, vol. ED-26, no. 1, pp. 2–9, 1979. 3

[6] D. Binder, E. C. Smith, and A. B. Holman, "Satellite anomalies from galactic cosmic rays," *IEEE Trans. Nucl. Sci.*, vol. 22, pp. 2675–2680, Dec. 1975. 3

[7] International Technology Roadmap for Semiconductors. Available: http://public.itrs.net. 3

[8] T. Karnik, P. Hazucha, and J. Patel, "Characterization of soft errors caused by single event upsets in CMOS processes," *IEEE Trans. Dependable and Secure Computing*, vol. 1, no. 2, pp. 128–143, Apr.-Jun. 2004. 3, 9, 10

[9] G. R. Srinivasan, P. C. Murley, and H. K. Tang, "Accurate, predictive modeling of soft error rate due to cosmic rays and chip alpha radiation," in *Proc. Int. Rel. Phys. Symp.*, pp. 12–16, 1994. 3, 69

[10] P. Hazucha, T. Karnik, J. Maiz, S. Walstra, B. Bloechel, J. Tschanz, G. Dermer, S. Hareland, P. Armstrong, and S. Borkar, "Neutron soft error rate measurements in a 90-nm CMOS process and scaling trends in SRAM from 0.25-μm to 90-nm generation," in *Proc. IEDM Tech. Dig.*, pp. 523–526, Dec. 2003. 3

[11] E. S. Fetzer, L. Wang, and J. Jones, "The multi-threaded, parity-protected 128-word register files on a dual-core Itanium®-family processor," in *ISSCC Dig. Tech. Papers*, pp. 382–3833, 2005. 4

[12] B. Stackhouse, B. Cherkauer, M. Gowan, P. Gronowski, and C. Lyles, "A 65nm 2-billion-transistor quad-core itaniumg processor," in *ISSCC Dig. Tech. Papers*, pp. 92–93, 2008. 4

[13] L. Lantz, "Soft errors induced by alpha particles," *IEEE Trans. Reliab.*, vol. 45, no. 2, pp. 174–179, Dec. 1996. 4

[14] M. W. Roberson, "Soft error rates in solder bumped packaging," in *Proc. Int. Symp. on Advanced Packaging Materials*, pp. 111–116, 1998. 4

[15] J. F. Ziegler, "Terrestrial cosmic rays," *IBM J. Res. Develop.*, vol. 40, no. 1, pp. 19–39, Jan. 1996. 5, 6

[16] R. Baumann, "Soft errors in advanced computer systems," *IEEE Design & Test of Computers*, vol. 22,, pp. 258–266, May/Jun. 2005. 14

[17] K. Kuhn *et al.*, "Managing process variation in Intels 45nm CMOS technology," *Intel Technology Journal*, vol. 12, no. 2, pp. 93–109, Jun. 2008. 15

[18] H. Mahmoodi, S. Mukhopadhyay, and K. Roy, "Estimation of delay variations due to random dopant fluctuations in nanoscale CMOS circuits," *IEEE J. Solid-State Circuits*, vol. 40, no. 9, pp. 1787–1795, Sep. 2005. 15

[19] S. R. Nassif, "Modeling and analysis of manufacturing variations," in *Proc. IEEE Custom Integrated Circuit Conf.*, pp. 223–228, 2001. 15

[20] A. J. Bhavnagarwala, X. Tang, and J. D. Meindl, "The impact of intrinsic device fluctuations on CMOS SRAM cell stability," *IEEE J. Solid-State Circuits*, vol. 36, no. 4, pp. 658–665, Apr. 2001. 16, 17

[21] A. Asenov, S. Kaya, and J. Davies, "Intrinsic parameter fluctuations in decananometer MOSFETs introduced by gate line edge roughness," *IEEE Trans. Electron Devices*, vol. 50, no. 5, pp. 1254–1260, May 2003. 16

[22] H. Masuda, S. Okawa, and M. Aoki, "Approach for physical design in sub-100nm era," in *Proc. Int. Symp. on Circuits and Systems*, pp. 5934–5937, 2005. 17

[23] R. K. Krishnarnurthy, A. Alvandpour, V. De, and S. Borkar, "High-performance and low-power challenges for sub-70 nm microprocessor circuits," in *Proc. IEEE Custom Integrated Circuit Conf.*, pp. 125–128, 2002. 18

[24] P. E. Dodd, M. R. Shaneyfelt, J. R. Schwank, and G. L. Hash, "Neutron-induced soft errors, latchup, and comparison of SER test methods for SRAM technologies," in *Proc. IEDM Tech. Dig.*, pp. 333–336, 2002. 19

[25] A. S. Pavlov, *Design and Test of Embedded SRAMs.* PhD Thesis: University of Waterloo, 2005. 22

[26] J. E. Brewer, V. V. Zhirnov, and J. A. Hutchby, "Memory technology for the post silicon era," *IEEE Circuits Devices Mag.*, vol. 21, no. 2, pp. 13–20, Mar./Apr. 2005. 22

[27] J. Wuu, D. Weiss, C. Morganti, and M. Dreesen, "The asynchronous 24MB on-chip level-3 cache for a dual-core Itanium-family processor," in *ISSCC Dig. Tech. Papers*, pp. 488–489, 2005. 24

[28] A. Sharma, *Andvanced Semiconductor Memories: Architectures, Designs and Applications.* Wiley Inter-Science,, 2003. 24

[29] K. Noda and othersr, "A 1.9-μm^2 loadless CMOS four-transistor SRAM cell in a 0.18-pm logic techinology," in *Proc. IEDM Tech. Dig.*, pp. 643–646, Dec. 1998. 28, 29

[30] T. Hirose *et al.*, "A 20-ns 4-Mb CMOS SRAM with hierarchical word decoding architecture," *IEEE J. Solid-State Circuits*, vol. 25, no. 5, pp. 1068–1074, Oct. 1990. 37

[31] M. Eisele *et al.*, "The impact of intra-die device parameter variations on path delays on the design for yield of low voltage digital circuits," in *Proc. IEEE Int. Symp. Low Power Electronic Design*, pp. 237–242, Oct. 1996. 43

[32] S. Tachibana *et al.*, "A 2.6-ns wave-pipelined CMOS SRAM with dual-sensing-latch circuits," *IEEE J. Solid-State Circuits*, vol. 30, pp. 487–490, Apr. 1995. 44

[33] S. Schuster *et al.*, "A 15-ns CMOS 64k RAM," *IEEE J. Solid-State Circuits*, vol. 21, pp. 704–711, Oct. 1986. 44

[34] B. S. Amrutur and M. A. Horowitz, "A replica technique for wordline and sense control in low-power SRAMs," *IEEE J. Solid-State Circuits*, vol. 33, no. 8, pp. 1208–1219, Aug. 1998. 44

[35] A. Agarwal, H. Li, and K. Roy, "DRG-cache: A data retention gated-ground cache for low power," in *Proc. Design Automation Conf.*, pp. 473–478, 2002. 45

[36] K. Zhang *et al.*, "SRAM design on 65-nm CMOS technology with dynamic sleep transistor for leakage reduction," *IEEE J. Solid-State Circuits*, vol. 40, no. 4, pp. 895–901, Apr. 2005. 46

[37] N. S. Kim, K. Flautner, D. Blaauw, and T. Mudge, "Circuit and microarchitectural techniques for reducing cache leakage power," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 2, pp. 167–184, Feb. 2004. 47

[38] N. Azizi, A. Moshovos, and F. N. Najm, "Low-leakage asymmetric-cell sram," in *Proc. IEEE Int. Symp. Low Power Electronic Design*, pp. 48–51, 2002. 48

[39] S. Tang *et al.*, "A leakage-tolerant dynamic register file using leakage bypass with stack forcing (LBSF) and source follower nmos (SFN) techniques," in *Proc. Symp. VLSI Circuits*, pp. 320–321, Jun. 2002. 50

[40] P. Roche, J. M. Palau, C. Tavernier, G. Bruguier, R. Ecoffet, and J. Gasiot, "Determination of key parameters for SEU occurrence using 3-D full cell SRAM simu-

lations," *IEEE Trans. Nucl. Sci.*, vol. 46, no. 6, pp. 1354–1362, Dec. 1999. 53, 65, 75

[41] J. M. Palau, G. Hubert, K. Coulie, B. Sagnes, M. C. Calvet, and S. Fourtine, "Device simulation study of the SEU sensitivity of SRAMs to internal ion tracks generated by nuclear reactions," *IEEE Trans. Nucl. Sci.*, vol. 48, no. 2, pp. 225–231, Apr. 2001. 53

[42] Y. Z. Xu *et al.*, "Process impact on SRAM alpha-particle SEU performance," in *Proc. IEEE Int. Reliability Phys. Symp.*, pp. 294–299, Apr. 2004. 53

[43] B. Zhang, A. Arapostathis, S. Nassif, and M. Orshansky, "Analytical modeling of SRAM dynamic stability," in *Proc. IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 315–322, 2006. 54, 67, 71, 72, 73, 76, 77

[44] G. R. Srinivasan, "Modeling the cosmic-ray-induced soft-error rate in integrated circuits: An overview," *IBM J. Res. Develop.*, vol. 40, no. 1, pp. 77–89, Jan. 1996. 54

[45] S. W. Fu, A. M. Mohsen, and T. C. May, "Alpha-particle-induced charge collection measurements and the effectiveness of a novel p-well protection barrier on VLSI memories," *IEEE Trans. Electron. Devices*, vol. 32, pp. 49–54, Feb. 1985. 55

[46] D. Burnett, C. Lage, and A. Bormann, "Soft-error-rate improvement in advanced BiCMOS SRAMs," in *Proc. IEEE Int. Reliability Phys. Symp.*, pp. 156–160, 1993. 55

[47] J. D. Hayden *et al.*, "A quadruple well, quadruple polysilicon BiCMOS process for fast 16 Mb SRAMs," *IEEE Trans. Electron. Devices*, vol. 41, pp. 2318–2325, Dec. 1994. 55

[48] E. H. Cannon, D. D. Reinhardt, M. S. Gordon, and P. S. Makowenskyj, "SRAM SER in 90, 130 and 180 nm bulk and SOI technologies," in *Proc. IEEE Int. Reliability Phys. Symp.*, pp. 300–304, 2004. 55

[49] T. M. Mnich *et al.*, "Comparison of analytical models and experimental results for single event upset in CMOS SRAM," *IEEE Trans. Nucl. Sci.*, vol. NS-30, p. 4620, 1983. 55

[50] F. Ootsuka *et al.*, "A novel 0.25μm full CMOS SRAM cell using stacked cross couple with enhanced soft error immunity," in *Proc. IEDM Tech. Dig.*, pp. 205–208, 1998. 56

[51] P. Roche, F. Jacquet, C. Callat, and J.-P. Schoellkopf, "An alpha immune and ultra low neutron SER high density SRAM," in *Proc. IEEE Int. Reliability Phys. Symp.*, pp. 671–672, Apr. 2004. 57

[52] J. F. Ziegler and H. Puchner, *SER - History, Trends, and Challenges: A Guide for Designing with Memory ICs*. Cypress Semiconductor Corp., 2004. 57

[53] T. Calin, M. Nicolaidis, and R. Velazco, "Upset hardened memory design for submicron CMOS technology," *IEEE Trans. Nucl. Sci.*, vol. 43, no. 6, pp. 2874–2878, Dec. 1996. 58, 94

[54] S. S. Mukherjee, J. Emer, and S. Reinhardt, "The soft error problem: an architectural perspective," in *Proc. Int. Symp. on High-Performance Computer Architecture (HPCA)*, pp. 243– 247, Feb. 2005. 59

[55] K. Chakraborty and P. Mazumder, *Fault-tolerance and reliability techniques for high-density random-access memories*. Upper Saddle River, New Jersey: Prentice Hall, 2002. 60

[56] C. L. Chen and M. Y. Hsiao, "Error-correcting codes for semiconductor memory applications: a state-of-the-art review," *IBM J. Res. Develop.*, vol. 28, no. 2, pp. 124–134, Mar. 1984. 63, 94

[57] P. Hazucha and C. Svensson, "Impact of CMOS technology scaling on the atmospheric neutron soft error rate," *IEEE Trans. Nucl. Sci.*, vol. 47, no. 6, pp. 2586–2594, Dec. 2000. 65, 78

[58] M. Sharifkhani, S. M. Jahinuzzaman, and M. Sachdev, "Dynamic data stability in low-power SRAM design," in *Proc. IEEE Custom Integrated Circuit Conf.*, pp. 237–240, 2007. 67

[59] Q. Ding, R. Luo, H. Wang, H. Yang, and Y. Xie, "Modeling the impact of process variation on critical charge distribution," in *Proc. IEEE Int. SOC Conf.*, pp. 243–246, 2006. 83

[60] R. R. Montanes, J. P. de Gyvez, and P. Volf, "Resistance characterization for weak open defects," *IEEE Design and Test of Computers*, vol. 19, no. 5, p. 1826, 2002. 84, 88

[61] A. Pavlov, M. Sachdev, and J. P. de Gyvez, "An SRAM weak cell fault model and a DFT technique with a programmable detection threshold," in *Proc. IEEE Int. Test Conf. (ITC)*, pp. 1006 – 1015, 2004. 89, 90

[62] E. Seevinck, F. List, and J. Lohstroh, "Static-noise margin analysis of MOS SRAM cells," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 5, pp. 748–754, 1987. 89

[63] J. L. Shin, B. Petrick, M. Singh, and A. S. Leon, "Design and implementation of an embedded 512-KB Level-2 cache subsystem," *IEEE J. Solid-State Circuits*, vol. 40, no. 9, pp. 1815–1820, Sep. 2005. 94, 126, 127, 129

[64] K. Osada, Y. Saitoh, E. Ibe, and K. Ishibashi, "16.7-fA/cell tunnel-leakage-suppressed 16-Mb SRAM for handling cosmic-ray-induced multierrors," *IEEE J. Solid-State Circuits*, vol. 38, no. 11, pp. 1952–1957, Nov. 2003. 101, 115, 126, 127, 129

[65] C. W. Slayman, "Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations," *IEEE Trans. Device Mater. Rel.*, vol. 5, no. 3, pp. 397–404, Sep. 2005. 102

[66] K. Kanda, H. Sadaaki, and T. Sakurai, "90% write power-saving SRAM using sense-amplifying memory cell," *IEEE J. Solid-State Circuits*, vol. 39, no. 6, pp. 927–933, Jun. 2004. 105, 115

[67] C. Yu, W. Wang, and B. Liu, "A 3-input XOR/XNOR for low-voltage low-power applications," in *Proc. IEEE Asia-Pacific Conf. on Circuits and Systems*, pp. 505–508, Tianjin, China 2000. 110

[68] "Test Method for Beam Accelerated Soft Error Rate," *JEDEC Standard: JESD 89-3A*, Oct. 2007. 120

[69] iRoC Technologies, 2004. 127