

THESIS

AGING-INDUCED DATA IMPRINTING IN SRAM ARRAYS: CHARACTERIZATION, ANALYSIS, AND MITIGATION

Submitted by

Abishek Stephen Vellankanni

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Summer 2026

Master's Committee:

Advisor: Biswajit Ray

Sudeep Pasricha

Indrajit Ray

Copyright by Abishek Stephen Vellankanni 2026

All Rights Reserved

ABSTRACT

AGING-INDUCED DATA IMPRINTING IN SRAM ARRAYS: CHARACTERIZATION, ANALYSIS, AND MITIGATION

Static random-access memory (SRAM) power-up states are widely used as a source of randomness and device uniqueness, but they can also retain traces of previously stored information after prolonged stress, creating an end-of-life security vulnerability. This thesis investigates both the formation of aging-induced data imprinting in commercial off-the-shelf (COTS) SRAMs and the mitigation of such imprints through radiation-based sanitization. Under accelerated thermal aging, multiple SRAM chips from different vendors and technology nodes exhibit reverse imprinting, in which the post-aging power-up state becomes biased toward the complement of the stored data pattern. The imprint strength depends on aging temperature, stress duration, and supply voltage, while relaxation measurements show that the induced patterns can remain visible for months at room temperature. The observed behavior indicates that negative bias temperature instability (NBTI)-induced threshold-voltage shifts in PMOS transistors are the dominant mechanism responsible for the effect. To mitigate this vulnerability, this thesis further demonstrates a total ionizing dose (TID)-based sanitization technique for aged SRAM arrays. Experimental results show near-complete sanitization at approximately 50-100 krad(Si) for most of the aged chips. Overall, this thesis highlights both the security risk posed by aging-induced SRAM data imprinting and the effectiveness of radiation-based sanitization as a practical end-of-life mitigation approach.

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to my advisor, Dr. Biswajit Ray, for his unwavering support, encouragement, and guidance throughout the successful completion of this thesis. His expertise and deep knowledge of memory devices significantly strengthened my understanding of the underlying concepts and helped shape the direction of this research. I am especially grateful for the opportunity to work under his supervision and for the collaborative research environment he fostered. His patience and invaluable support during the experimental phases greatly contributed to the success of this work.

I would also like to extend my sincere gratitude to my committee members, Dr. Sudeep Pasricha and Dr. Indrajit Ray, for their time, thoughtful consideration, and constructive feedback throughout this process.

I am grateful to the Department of Electrical and Computer Engineering at Colorado State University for providing the academic environment, resources, and support necessary for carrying out this research.

Additionally, I would like to acknowledge my colleagues in the Reliable and Assured Microelectronics (RAM) Lab for their support and collaboration throughout this academic journey.

I am deeply grateful to my family for their constant support and encouragement throughout my academic journey. I am especially thankful to my mother, S. Jayalakshmi, whose persistent efforts and dedication have played a vital role in shaping my future. I also thank my father, R. Stephen Vellankanni, my fiancé, Mary Diana, for their prayers, constant encouragement, motivation and heartfelt blessings. Finally, I extend my deepest gratitude to all my close family members and friends who have supported me throughout this endeavor.

I am thankful to God Almighty for blessing me with the strength, opportunities, and perseverance needed to achieve this milestone.

This research was generously supported by the National Science Foundation under Grants 2423249 and 2317563. I gratefully acknowledge this support, which enabled the successful completion of the data collection and analysis presented in this thesis.

TABLE OF CONTENTS

ABSTRACT	ii
ACKNOWLEDGEMENTS	iii
LIST OF TABLES	vii
LIST OF FIGURES	viii
Chapter 1 Introduction	1
1.1 Motivation	1
1.2 Thesis Objective	5
1.3 Thesis Outline	7
Chapter 2 Background	10
2.1 Memory Subsystems in Modern Computers	10
2.2 Fundamentals of SRAM	14
2.3 Aging Mechanisms in CMOS and SRAM	33
2.4 Related Works	36
Chapter 3 Experimental Setup and Flow	39
3.1 Hardware Setup	40
3.2 PCB Design	42
3.3 Firmware Design	46
Chapter 4 Power-Up State Characterization	57
4.1 SRAM Power-Up Behavior	57
4.2 PUS Characterization of SRAM Chips	59
4.3 SRAM Power-Up States as Device Fingerprints	61
4.4 Physical Unclonable Functions (PUFs)	63
4.5 Conclusion	64
Chapter 5 Aging Induced Imprinting Effects on SRAM Array	66
5.1 Introduction	66
5.2 Background	67
5.3 Threat Model	68
5.4 Experimental Flow	69
5.5 Results and Discussion	74
5.6 Conclusion	83
Chapter 6 Radiation-Based Sanitization of Aged SRAM	85
6.1 Introduction	85
6.2 Radiation Effects on SRAM	89
6.3 Experimental Flow	91
6.4 Results and Discussion	95
6.5 Conclusion	100

Chapter 7	Dosimeter Using SRAM Power-Up State	103
7.1	Introduction	103
7.2	Background	106
7.3	Experimental Setup	108
7.4	Results and Discussion	110
7.5	Conclusion	117
Chapter 8	Limitations, Conclusion and Future Work	118
Bibliography		124
Appendix A	Algorithm for SRAM Interfacing	133
A.1	Overall Firmware Flow	133
A.2	Serial Command Processing	134
A.3	Low-Level SRAM Read and Write Operations	135
A.4	Pattern Writing and Full Memory Dump	136
A.5	Repeated Power-Up-State Measurement	137

LIST OF TABLES

3.1	Details of the SRAM devices used in the experimental evaluation	39
3.2	Logic levels required on the SRAM control signals for principal bus operations	47
4.1	Intrinsic power-up-state variability of the SRAM chips	61
5.1	Summary of SRAM devices used in the aging study	69
6.1	Summary of aging and sanitization effects	99
7.1	Summary of SRAM chips used in the dosimeter study	109
7.2	Technology node vs linear equation	115

LIST OF FIGURES

1.1	SRAM applications and end-use sectors.	2
1.2	Representative SRAM manufacturers.	3
2.1	Memory hierarchy in modern computing systems.	11
2.2	Six-transistor SRAM memory cell: (a) 6T SRAM bit cell, where P1–N1 and P2–N2 form two cross-coupled inverters and A1 and A2 are the access transistors; (b) SRAM bit cell showing two cross-coupled inverters that store one bit of data.	15
2.3	Two stable states of a cross-coupled inverter pair.	16
2.4	Architecture of an SRAM array.	18
2.5	Timing diagram during a 6T SRAM read operation for the case where $Q = 1$ and $\overline{Q} = 0$	20
2.6	Timing diagram during a 6T SRAM write operation for the case where a logic ‘0’ is written into node Q	22
2.7	Butterfly curve used to evaluate SRAM static noise margin.	26
2.8	Illustration of process variation and device mismatch in CMOS: (a) lot-to-lot variation, (b) wafer-to-wafer variation, (c) chip-to-chip variation, (d) within-die variation, and (e) mapping of device-parameter variation to performance variation using CMOS process corners.	32
3.1	SRAM pinout and initial Arduino Due interfacing map.	41
3.2	SRAM interfacing hardware setup.	41
3.3	Wired prototype setup for initial SRAM interfacing.	42
3.4	Design and development of the custom PCB used for SRAM interfacing.	43
3.5	PCB schematic for SRAM interfacing.	44
3.6	PCB layout for SRAM interfacing.	45
3.7	Firmware validation menu.	53
3.8	Initial SRAM power-up-state readout.	54
3.9	Full-memory write verification.	55
3.10	Full-memory readback verification.	56
4.1	SRAM power-up behavior: (a) power-up mechanism of a 6T SRAM cell and (b) representative PUS simulation waveform.	58
4.2	Representative fresh PUS patterns from four different COTS SRAM technologies.	60
4.3	SRAM fingerprint map and extracted 64-bit region with power-up probability.	62
5.1	SRAM end-of-life threat model.	68
5.2	(a) Aging setup and (b) experimental flow.	69
5.3	Written image patterns.	71
5.4	Aging-induced imprinting results.	74
5.5	Temperature-Accelerated Aging Effects.	75
5.6	Activation Energy of Aging Effects.	76
5.7	Voltage Acceleration Aging Effects.	77
5.8	Relaxation effects on the imprinted CY 90 nm SRAM chip.	78

5.9	(a) Multi-chip evaluation. (b) End-to-end visualization.	80
5.10	Root-cause analysis of imprinting (a) Before aging (b) After aging.	82
6.1	End-of-life handling paths for electronic devices: (a) non-destructive end-of-life paths and (b) discarding electronic devices.	86
6.2	Bias-dependent radiation response in two NMOS transistors: N_1 is OFF during irradiation and N_2 is ON during irradiation.	90
6.3	(a) Sanitization setup and (b) flow.	92
6.4	Sanitization results overview.	95
6.5	TID-based sanitization results (a) CY 150 nm and (b) ISSI 180 nm.	96
6.6	Sanitization relaxation results (a) CY 150 nm and (b) ISSI 180 nm.	97
6.7	TID-based sanitization mechanism.	99
7.1	Representative low-dose and high-dose radiation environments and applications.	103
7.2	Representative radiation dosimeter technologies and their general characteristics.	104
7.3	Conceptual overview of SRAM-PUS-based radiation dosimetry.	105
7.4	Background of the SRAM-PUS-based dosimeter: (a) schematic of a 6T SRAM cell and (b) illustration of GoldPUF generation from repeated power-up states.	107
7.5	Conceptual illustration of radiation-induced power-up-state change in SRAM: (a) pre-irradiation preferred power-up state, (b) device-level irradiation mechanism, and (c) post-irradiation preferred power-up state.	108
7.6	Experimental setup for SRAM-PUS dosimetry: (a) SRAM interface platform using a TSOP-54 socket and Arduino Due, and (b) gamma irradiation facility used for the experiments.	109
7.7	Pre-irradiation characterization of the SRAM PUS: (a) HD (%) as a function of PUF number for Cypress 90 nm and 150 nm chips, and (b) variance (%) in HD as a function of PUF length.	111
7.8	(a) HD (%) as a function of PUF Numbers for 90 nm node. The same plots for 150 nm and 250nm node are shown in (b), and (c), respectively. (d) PDF as a function of HD (%) for Cypress 90nm, (e) and (f) represents Cypress 150nm and Cypress 250nm respectively.	112
7.9	(a) ΔHD (%) as a function of Total-Ionizing-Dose (TID) in log scale, (b) ΔHD (%) as a function of TID in Linear scale.	113
7.10	HD mean (%) as a function of Total-Ionizing-Dose (TID) for linear model fitting . . .	114
7.11	Error (%) as a function of Total-Ionizing-Dose (TID)	115

Chapter 1

Introduction

1.1 Motivation

Modern digital systems rely heavily on Static Random Access Memory (SRAM) for fast and efficient data storage. SRAM is a high-speed volatile memory widely used in processor caches, embedded controllers, microcontrollers, and system-on-chip (SoC) platforms because of its low access latency and compatibility with standard Complementary Metal-Oxide-Semiconductor (CMOS) design flows. Unlike Dynamic Random Access Memory (DRAM), SRAM does not require refresh operations, which makes it especially attractive for applications requiring deterministic timing, rapid random access, and low-latency operation. As a result, SRAM remains a critical memory technology in artificial intelligence (AI) platforms, edge-computing devices, communication systems, and real-time embedded applications. Extensive circuit- and architecture-level research on low-voltage, high-performance, and low-power SRAM design further underscores its central role in modern embedded and computing systems [1–5].

The continued importance of SRAM is reflected in its wide deployment across modern end-use sectors. As shown in Figure 1.1, SRAM is used extensively in information technology and telecommunication systems, consumer electronics, automotive electronics, industrial platforms, healthcare devices, and aerospace and defense systems. These applications frequently depend on fast and reliable runtime memory for storing temporary program data, control variables, sensor information, and system state. In many such systems, SRAM stores security-sensitive content such as encryption keys, firmware variables, boot data, user credentials, and temporary sensor data that are essential to correct and secure operation.

SRAM is widely adopted not only because of its speed, but also because of its reliability in safety-critical and high-performance systems. In automotive, aerospace, defense, industrial, and medical electronics, SRAM is often preferred where predictable response and robust runtime oper-

ation are required. This broad deployment has led to a large commercial ecosystem of commercial off-the-shelf (COTS) SRAM products supplied by multiple vendors. Representative SRAM manufacturers are shown in Figure 1.2. The wide availability of these devices makes SRAM an important target for security evaluation, especially in scenarios involving hardware reuse, recycling, and end-of-life disposal.

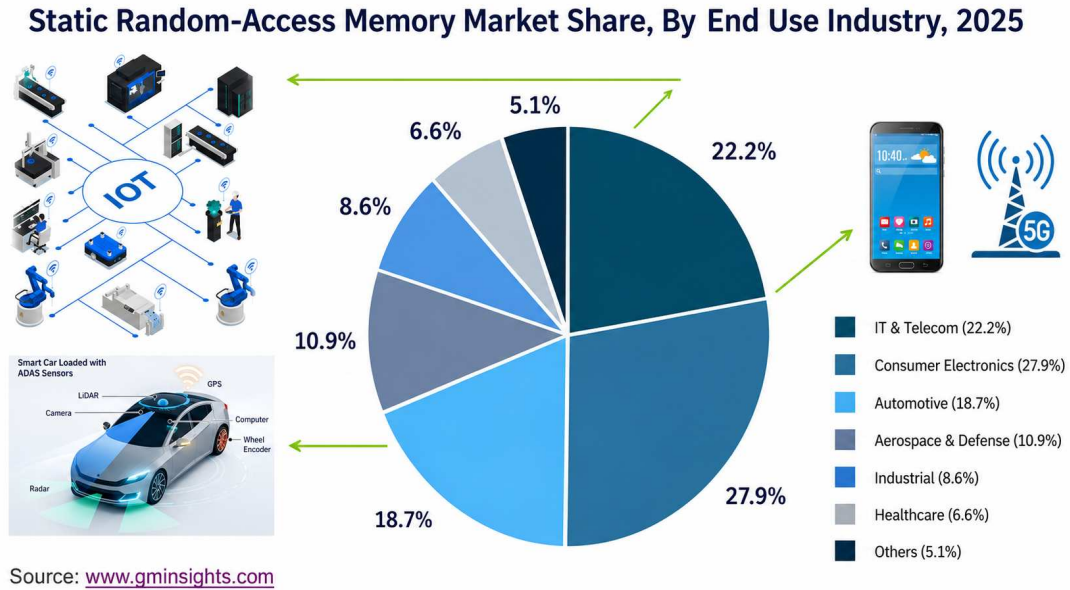


Figure 1.1: SRAM applications and end-use sectors.

In addition to its role as runtime memory, the power-up state (PUS) of SRAM has been widely studied as a source of physical randomness and device uniqueness. Prior work has shown that fabrication-induced transistor mismatch gives rise to characteristic power-up patterns, enabling SRAM to be used for hardware fingerprints, true-random-number generation, and Physical Unclonable Functions (PUFs) [?, 6–9]. These studies have established that SRAM power-up patterns are not merely random artifacts, but physically meaningful signatures that reflect the electrical characteristics of the underlying memory array.

While this power-up behavior has clear value for lightweight security primitives, it also introduces an important security concern. SRAM is typically assumed to lose its stored data after

power-off because it is a volatile memory. However, recent studies have shown that prolonged thermal and electrical stress can alter the future power-up behavior of SRAM cells, causing traces of previously stored data to become imprinted into the subsequent PUS [10, 11]. A memory array that is nominally volatile can therefore exhibit power-up bias related to earlier stored information after sufficient aging. This observation creates a serious end-of-life security risk because data that is expected to disappear after power removal may partially reappear during later power-up measurements.

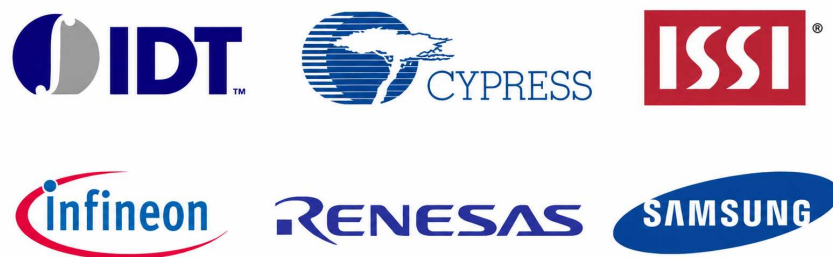


Figure 1.2: Representative SRAM manufacturers.

The practical implications of this phenomenon are significant. If an aged SRAM array retains power-up traces of previously stored information, discarded, reused, or recycled SRAM chips may expose security-sensitive content to an adversary with physical access. In practice, such content may include encryption keys, firmware parameters, boot information, user credentials, and other runtime data needed for system operation. Sensitive information recovered from the PUS can then be analyzed and exploited by attackers. This issue becomes particularly important because SRAM is embedded in a wide range of modern electronic systems and is often handled under the assumption that volatility alone is sufficient for secure disposal.

Recent studies have shown that this concern is not merely hypothetical. Prior work demonstrated that prolonged high-temperature stress can leave visible and recoverable imprints of previously stored information in SRAM power-up states [10]. Additional studies have shown that aging-related bias in memory power-up behavior can also be exploited to expose on-chip secrets or to identify used and recycled systems [12–14]. Together, these results suggest that SRAM aging is not only a device-reliability issue but also a practical information-leakage problem.

From a device-physics perspective, the dominant explanation for this behavior is the gradual development of threshold-voltage mismatch caused by long-term stress. The most relevant mechanism is Negative Bias Temperature Instability (NBTI), which primarily affects PMOS transistors that remain biased in the ON state over extended periods. As the stressed PMOS device degrades relative to its complementary counterpart, the electrical symmetry of the SRAM cell is disturbed, which can alter both its static behavior and its future power-up tendency. Earlier studies on aged memory-cell bias, power-up-state asymmetry, and recycled-chip detection support the view that transistor-level degradation can translate into measurable changes at the array level [13, 14]. This provides the physical basis for understanding how previously stored information may become imprinted into the future power-up behavior of SRAM arrays.

The security consequences of such imprinting become especially serious when considering end-of-life handling of electronic systems. In practice, many systems are not physically destroyed at retirement. Instead, they may be resold, recycled, reused, or disassembled for parts. If aged SRAM retains power-up traces of previously stored data, then second-hand hardware may leak sensitive information long after its intended use. One possible response is destructive disposal through shredding, incineration, or similar techniques. However, such methods are often expensive, environmentally undesirable, and incompatible with hardware reuse or refurbishment. For this reason, there is strong motivation to investigate whether aged and imprinted SRAM arrays can be sanitized in a controlled manner while preserving device functionality.

Ionizing radiation provides a promising direction for such a mitigation strategy. Prior studies have shown that radiation can alter SRAM power-up states, degrade or reshape SRAM-PUF behav-

ior, and induce measurable changes in cell stability through trapped charge and threshold-voltage modification [9, 11, 15]. These findings suggest that radiation can perturb the same transistor-level balance that aging modifies. Radiation-based treatment therefore emerges as a plausible means of deliberately reshaping the power-up preference of aged SRAM cells and suppressing previously imprinted information.

This thesis is motivated by the need to investigate this problem in a systematic way. Existing studies have separately examined SRAM power-up fingerprints, aging-induced imprinting, radiation-induced power-up state changes, and recycled-chip bias detection. However, a unified investigation that connects thermal aging, imprint formation, relaxation behavior, cross-technology variation, and radiation-based sanitization across multiple COTS SRAM technologies remains limited. By addressing this gap, this thesis aims to contribute both to the understanding of SRAM security vulnerabilities and to the development of practical end-of-life mitigation strategies for memory-bearing systems.

1.2 Thesis Objective

The objective of this thesis is to investigate aging-induced data imprinting in commercial SRAM arrays, to evaluate radiation-based sanitization as a possible mitigation technique for removing residual power-up-state bias in aged memories, and to explore the use of SRAM power-up states for radiation dosimetry. This study aims to:

- **Characterize the Power-Up State of SRAM:** Study the natural power-up behavior of commercial SRAM chips and determine how the power-up-state patterns differ across devices, technology nodes, and manufacturers. This objective establishes the intrinsic baseline needed for analyzing later modifications caused by aging and radiation.
- **Characterize Aging-Induced Data Imprinting in SRAM:** Conduct experiments to evaluate how the power-up state of SRAM changes after prolonged thermal stress. This includes studying the imprinting behavior of previously written data patterns and determining how strongly the post-aging power-up state becomes biased by the stored information.

- **Establish a Framework for Imprinting Analysis:** Develop an experimental and analytical framework for quantifying data imprinting in SRAM arrays. This involves writing known image or data patterns into SRAM, applying controlled aging stress, reading the post-aging power-up state, and using suitable metrics to measure the strength and persistence of the imprint.
- **Evaluate the Effects of Stress Conditions:** Provide a detailed understanding of how aging temperature, stress duration, and supply voltage influence the magnitude of data imprinting. This objective is intended to identify the operating conditions under which the imprinting effect becomes strongest and most security-critical.
- **Study Relaxation and Long-Term Persistence:** Examine how the imprinted power-up-state bias evolves after the aging stress is removed. This includes monitoring the relaxation behavior of aged SRAM chips over time in order to determine how long the imprint remains visible and whether the fresh power-up distribution can be restored naturally.
- **Compare Multiple SRAM Technologies:** Investigate whether aging-induced data imprinting is limited to a single SRAM family or is observable across different technology nodes and vendors. This objective helps establish the broader applicability and security significance of the imprinting phenomenon in commercial SRAMs.
- **Evaluate Radiation-Based Sanitization of Aged SRAM:** Investigate whether ionizing radiation can be used to sanitize aged and imprinted SRAM arrays. This includes overwriting aged memory locations with new data patterns, exposing the chips to controlled radiation, and measuring whether the subsequent power-up state no longer reveals the previously imprinted information.
- **Investigate Dosimeter Using SRAM Power-Up State:** Explore whether the power-up state of SRAM can be used as a practical radiation dosimeter. This objective includes evaluating how the Hamming Distance between the pre-irradiation and post-irradiation power-up states

changes with total ionizing dose, determining the useful dose range, and comparing the dosimetry response across different SRAM technology nodes.

- **Explain the Physical Mechanisms of Imprinting, Sanitization, and Dosimetry Response:**

Relate the observed experimental behavior to transistor-level degradation and radiation effects in the SRAM cell. In particular, this objective seeks to explain how threshold-voltage mismatch caused by aging and radiation modifies the preferred power-up direction of the cell and how these changes can be used both for sanitization and for dosimetry.

By achieving these objectives, this thesis aims to contribute to the understanding of SRAM aging as a security vulnerability, to the development of practical end-of-life mitigation techniques for memory-bearing systems, and to the broader study of SRAM power-up behavior under ionizing radiation. The results provide insight into how previously stored information can persist in aged SRAM power-up states, how radiation-based treatment may be used to sanitize such memories before reuse or disposal, and how SRAM power-up states may serve as a compact digital dosimetry signal.

1.3 Thesis Outline

This thesis is organized as follows:

- **Chapter 1: Introduction.** This chapter presents the motivation for the thesis by discussing the growing security importance of SRAM in modern computing systems and the risks associated with aging-induced data imprinting. It introduces the central research problem, explains why secure end-of-life treatment of SRAM is necessary, and states the main objectives of the work. The chapter also provides a roadmap for the remainder of the thesis.
- **Chapter 2: Background.** This chapter provides the technical foundation required for the thesis. It begins with an overview of the memory subsystem in computing systems and the role of SRAM within the memory hierarchy. It then discusses SRAM cell structure,

array architecture, memory operations, power-up behavior, and the use of SRAM power-up states in hardware-security applications such as device fingerprinting and PUFs. The chapter further reviews CMOS aging mechanisms, especially NBTI and PBTI, and explains how these mechanisms affect SRAM cell behavior over time. Related prior works relevant to SRAM power-up behavior, aging, radiation response, and security are also discussed.

- **Chapter 3: Experimental Setup and Flow.** This chapter describes the complete experimental platform developed for this work. It presents the SRAM interfacing hardware, including the PCB, TSOP socket, Arduino Due controller, and firmware used for reading, writing, and validating SRAM contents. The chapter also explains the thermal-aging procedure, the radiation exposure methodology, and the overall experimental flow used for power-up-state characterization, imprinting, relaxation, sanitization, and dosimetry studies. In addition, it defines the data analysis metrics used throughout the thesis, including Hamming distance, Hamming weight, and related measures for quantifying SRAM behavior.
- **Chapter 4: Power-Up State Characterization.** This chapter presents the characterization of the natural power-up behavior of SRAM arrays. It explains how the power-up state originates from transistor-level mismatch, shows representative power-up-state patterns from different SRAM technologies, and discusses how these power-up patterns can be used for device fingerprinting and PUF applications. The chapter concludes by establishing the intrinsic baseline variability of the SRAM chips used in this thesis.
- **Chapter 5: Aging Induced Imprinting Effects on SRAM Array.** This chapter presents the main experimental results on aging-induced data imprinting in commercial SRAM arrays. It first demonstrates the imprinting phenomenon using representative image patterns and then studies how the effect evolves under different stress temperatures, stress durations, and supply voltages. The chapter also investigates room-temperature relaxation to determine how long the induced power-up-state bias persists after aging. A cross-technology comparison is then performed across multiple SRAM devices from different vendors and technology

nodes. Finally, the chapter discusses the physical origin of the observed reverse imprinting behavior and relates it to transistor-level degradation mechanisms.

- **Chapter 6: Radiation-Based Sanitization of Aged SRAM.** This chapter investigates the use of ionizing radiation as a method for sanitizing aged and imprinted SRAM arrays. It introduces the motivation for radiation-based sanitization and explains the experimental procedure used to overwrite aged SRAM locations with new patterns and irradiate them under powered-on conditions. The chapter then presents the measured sanitization results across different technology nodes, evaluates the relaxation behavior after sanitization, and discusses the underlying transistor-level mechanism responsible for the observed power-up-state shift. The effectiveness and limitations of radiation-based sanitization are also discussed in the context of secure end-of-life mitigation.
- **Chapter 7: Dosimeter Using SRAM Power-Up State.** This chapter explores the use of SRAM power-up states as a digital radiation dosimeter. It introduces the motivation for SRAM-based dosimetry, presents the background of GoldPUF-based comparison, describes the irradiation setup and characterization methodology, and analyzes the radiation-induced change in Hamming Distance as a function of total ionizing dose. The chapter also evaluates sensitivity, linearity, and error across different SRAM technology nodes, and discusses the applicability of SRAM power-up states for real-time dosimetry.
- **Chapter 8: Conclusion and Future Work.** This chapter summarizes the major findings of the thesis and highlights the key contributions of the work. It revisits the main observations on power-up-state characterization, aging-induced imprinting, radiation-based sanitization, and SRAM-based dosimetry, and discusses their implications for memory security, hardware reuse, end-of-life device handling, and radiation sensing. The chapter concludes by outlining possible directions for future research.

Chapter 2

Background

In this chapter, the fundamental concepts required for understanding aging-induced data imprinting and radiation-based sanitization in SRAM are presented. First, the memory subsystem in modern computing systems is introduced, with emphasis on the memory hierarchy, the distinction between volatile and non-volatile memories, and the importance of SRAM in high-performance and security-sensitive applications. Next, the discussion narrows to the structure, operation, and key characteristics of SRAM, including cell architecture, stability, leakage, and the influence of process variation and device mismatch. This is followed by an explanation of SRAM power-up behavior and its role in security applications such as device fingerprinting and PUFs. After establishing these fundamentals, the chapter reviews important CMOS aging mechanisms, particularly NBTI and Positive Bias Temperature Instability (PBTI), and discusses how they influence SRAM cell behavior over time. Finally, radiation effects in SRAM are introduced, with focus on how ionizing radiation alters device characteristics, affects power-up-state behavior, and relates to the sanitization techniques investigated in this thesis. Together, these topics provide the background necessary for the experimental studies presented in the subsequent chapters.

2.1 Memory Subsystems in Modern Computers

2.1.1 Memory Hierarchy

Modern computing systems organize memory in a hierarchical manner in order to balance speed, storage capacity, and cost. As shown in Figure 2.1, the memory hierarchy is typically arranged as a pyramid, with the fastest and most expensive storage elements located at the top, and the slowest but highest-capacity storage elements located at the bottom. This organization is essential because no single memory technology can simultaneously provide very high speed, very large capacity, low cost, and low power consumption.

At the top of the hierarchy are processor registers, which provide the fastest data access and are used by the CPU to store operands, intermediate values, and control information required for immediate execution. Just below the registers are cache memories, which are designed to reduce the latency between the processor and main memory by storing frequently accessed data and instructions. Cache memory is usually implemented using SRAM because it offers very fast read and write access compared to other memory technologies. Since cache size is limited by area and cost, cache memory is arranged in multiple levels, such as L1, L2, and L3, to provide a tradeoff between access speed and storage capacity.

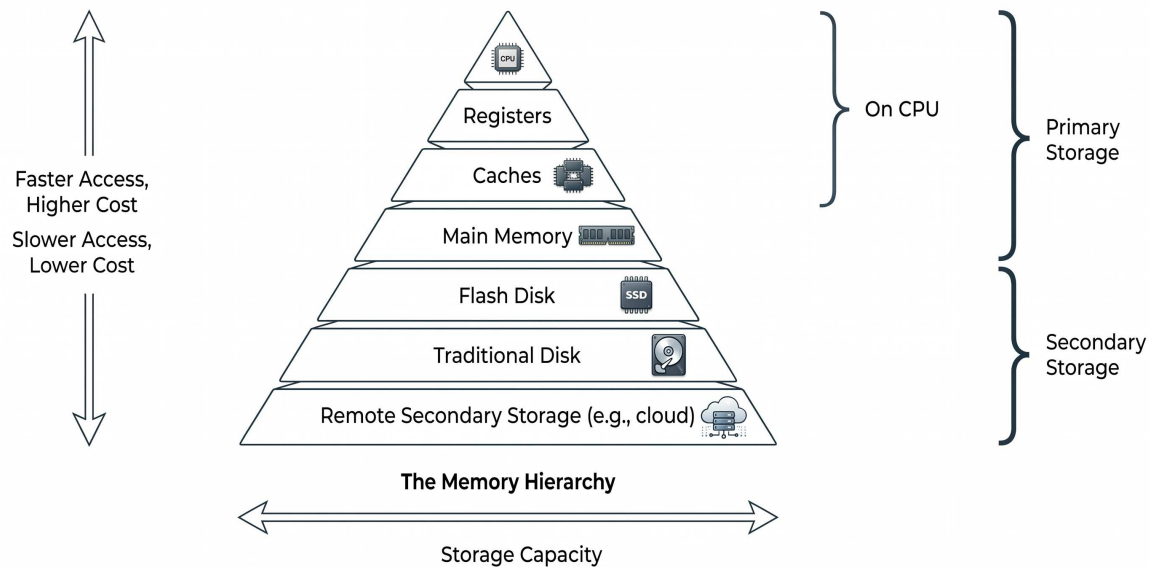


Figure 2.1: Memory hierarchy in modern computing systems.

Below the cache levels lies the main memory, which is commonly implemented using DRAM. Main memory provides a much larger storage space than on-chip caches and serves as the primary workspace for active programs and data during execution. However, DRAM is slower than SRAM and requires periodic refresh operations to retain stored information. For this reason, DRAM is suitable for large-capacity working memory, while SRAM is preferred in locations where speed is more critical.

At lower levels of the hierarchy are secondary storage devices such as solid-state drives (SSDs), flash storage, and hard disk drives (HDDs). These memories provide significantly larger storage capacity and retain data even when power is removed, making them suitable for long-term storage of operating systems, software, and user files. However, their access times are much slower than those of main memory. At the bottom of the hierarchy, remote or cloud-based storage provides even greater capacity and accessibility, but with much higher latency due to network communication overhead.

Thus, the memory hierarchy is designed so that smaller, faster memories close to the processor handle frequently used data, while larger and slower memories store less frequently accessed information. This layered organization allows computing systems to achieve both high performance and large effective storage capacity. Since SRAM occupies the upper part of this hierarchy and is extensively used in performance-critical and security-sensitive applications, understanding its behavior is particularly important in the context of this thesis.

2.1.2 Importance of SRAM in Computer Systems

SRAM plays a critical role in modern computer systems because it provides very fast data access with low latency. It is a volatile memory technology, meaning that stored information is not retained once power is removed. Although SRAM occupies larger silicon area and has a higher cost per bit than DRAM, its speed makes it indispensable in performance-critical parts of the system. For this reason, SRAM is extensively used in processor caches, register files, buffers, and small on-chip memory blocks where rapid access to data is essential for efficient execution.

One of the main reasons SRAM is important is its use in cache memory. Modern processors operate at very high speeds and can execute instructions much faster than data can be fetched from main memory. If the processor had to access DRAM for every instruction or data request, overall system performance would be severely limited by memory latency. SRAM-based caches reduce this bottleneck by storing frequently accessed instructions and data close to the processor. This

allows the processor to access critical information much faster than it could from main memory, thereby improving throughput and reducing execution delays.

SRAM is also widely used in embedded systems and SoC platforms, where fast local storage is required for control logic, temporary computation results, and communication buffering. In many microcontrollers, digital signal processors, and hardware accelerators, SRAM is integrated directly on-chip to support real-time operation. Because of its fast access time and simple interface, SRAM is well suited for storing intermediate data in applications that require deterministic timing, such as automotive electronics, industrial control systems, communication devices, and consumer electronics.

Another important aspect of SRAM is its role as fast working memory during active system operation. While non-volatile memories are used for persistent storage, SRAM is used where data must be accessed quickly and repeatedly during computation. In many systems, SRAM temporarily holds security-sensitive information such as keys, firmware variables, boot data, and runtime control data. As a result, its importance extends beyond performance and into the domains of reliability and security.

In addition to its performance benefits, SRAM is also important in hardware security applications. The power-up behavior of SRAM can be influenced by intrinsic device mismatch, which gives rise to a unique PUS pattern for each chip. This property has enabled SRAM to be used in security primitives such as PUFs, device identification, and lightweight key generation. At the same time, the same physical properties that make SRAM useful for security can also create vulnerabilities when aging, stress, or radiation alter the preferred power-up behavior of the memory cells.

Therefore, SRAM occupies a unique position in modern computer systems. It is not only essential for high-speed computation and efficient processor operation, but it is also closely tied to reliability and security concerns. Since this thesis focuses on aging-induced imprinting and sanitization in SRAM arrays, understanding the importance of SRAM in both performance and

security contexts is necessary for appreciating the broader impact of the phenomena studied in this work.

2.2 Fundamentals of SRAM

SRAM is a widely used volatile memory technology in modern digital systems due to its high speed, low access latency, and compatibility with standard CMOS logic. Unlike DRAM, SRAM does not require periodic refresh operations to retain stored information as long as power is supplied. This feature makes SRAM highly suitable for applications that demand fast and repeated data access, such as processor caches, register files, and on-chip buffers. However, the main trade-off is that SRAM cells occupy larger silicon area than DRAM cells, which increases the cost per bit and limits the overall storage density.

2.2.1 Cell Structure of SRAM

A conventional SRAM bit cell is typically implemented as a six-transistor (6T) circuit. As shown in Figure 2.2, the 6T SRAM cell consists of two cross-coupled CMOS inverters and two access transistors. The two PMOS–NMOS pairs, namely P1–N1 and P2–N2, form the cross-coupled inverters, while the access transistors A1 and A2 connect the internal storage nodes to the bit lines during read and write operations. The word line (WL) controls whether the access transistors are turned on or off [16, 17].

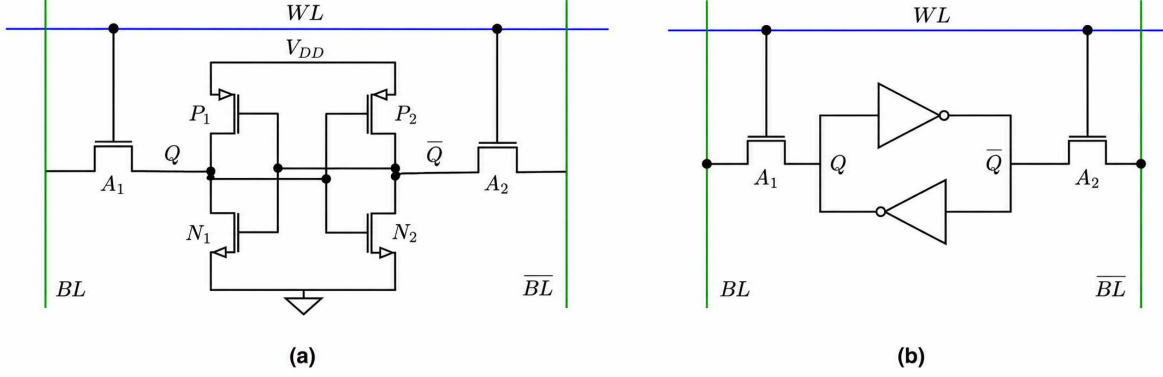


Figure 2.2: Six-transistor SRAM memory cell: (a) 6T SRAM bit cell, where P1–N1 and P2–N2 form two cross-coupled inverters and A1 and A2 are the access transistors; (b) SRAM bit cell showing two cross-coupled inverters that store one bit of data.

The cross-coupled inverter pair creates a regenerative feedback loop that allows the cell to store one bit of information in one of two stable states. In one state, node Q is at logic ‘1’ and node $\bar{Q}$ is at logic ‘0’. In the other state, node Q is at logic ‘0’ and node $\bar{Q}$ is at logic ‘1’. Because of this bistable behavior, the SRAM cell can preserve its stored data indefinitely as long as power is maintained. Figure 2.3 illustrates these two stable operating states of the cross-coupled inverter pair.

Compared with DRAM, the main advantage of SRAM lies in its speed and the absence of refresh overhead. However, because each SRAM bit cell requires six transistors, it consumes significantly more area than a DRAM cell, which reduces memory density for a given silicon area.

In addition to the basic transistor arrangement, proper sizing of the 6T cell is essential for reliable operation. The relative strengths of the pull-down, pull-up, and access transistors determine whether the cell can be read without disturbance and written without failure. A commonly used design parameter is the cell ratio (CR), defined as the strength ratio of the pull-down NMOS transistor to the access transistor:

$$CR = \frac{(W/L)_{\text{pull-down}}}{(W/L)_{\text{access}}}.$$

A larger cell ratio improves read stability because the pull-down transistor can more effectively hold the internal storage node against the voltage division created when the cell is connected to the precharged bit line during a read operation [16, 17].

Another important sizing parameter is the pull-up ratio (PR), which relates the PMOS pull-up transistor to the access transistor:

$$PR = \frac{(W/L)_{\text{pull-up}}}{(W/L)_{\text{access}}}.$$

This ratio influences write ability. To write new data into the cell successfully, the access transistor must be strong enough to overpower the pull-up device on the node being forced low. For this reason, write operation generally becomes easier when the pull-up transistor is weaker relative to the access transistor, that is, when the pull-up ratio is smaller [16].

Thus, SRAM cell design involves an important tradeoff. Increasing the cell ratio improves read stability, while reducing the pull-up ratio improves write ability. These sizing constraints directly affect static noise margin, read disturb behavior, and overall robustness of the SRAM cell. Since this thesis focuses on mechanisms that alter transistor threshold voltages and disturb the balance of the cell, understanding these sizing relationships is important for interpreting how aging and radiation modify SRAM power-up behavior.

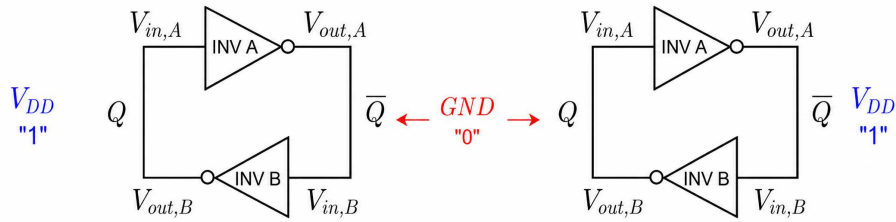


Figure 2.3: Two stable states of a cross-coupled inverter pair.

2.2.2 SRAM Array Architecture

In practical memory implementations, individual SRAM cells are organized into large two-dimensional arrays of rows and columns. Figure 2.4 shows a simplified SRAM array architecture. Each row of cells shares a common word line, while each column is connected through complementary bit lines. A row decoder selects the desired word line based on the input address, and peripheral circuits such as column drivers and sense amplifiers support write and read operations, respectively [15, 16].

During memory access, the address decoder activates one row of the array, thereby connecting all cells in that row to their corresponding bit lines through the access transistors. The column circuitry then determines which bit positions are read or written. This array-based structure allows SRAM to scale from a single bit cell to large memory blocks while still preserving fast random-access behavior [16].

The organization of SRAM into rows and columns is especially important because it determines how peripheral circuits interact with the bit cells. Efficient decoder design, bit-line conditioning, and sensing circuitry all play critical roles in ensuring reliable and high-speed SRAM operation [15, 16].

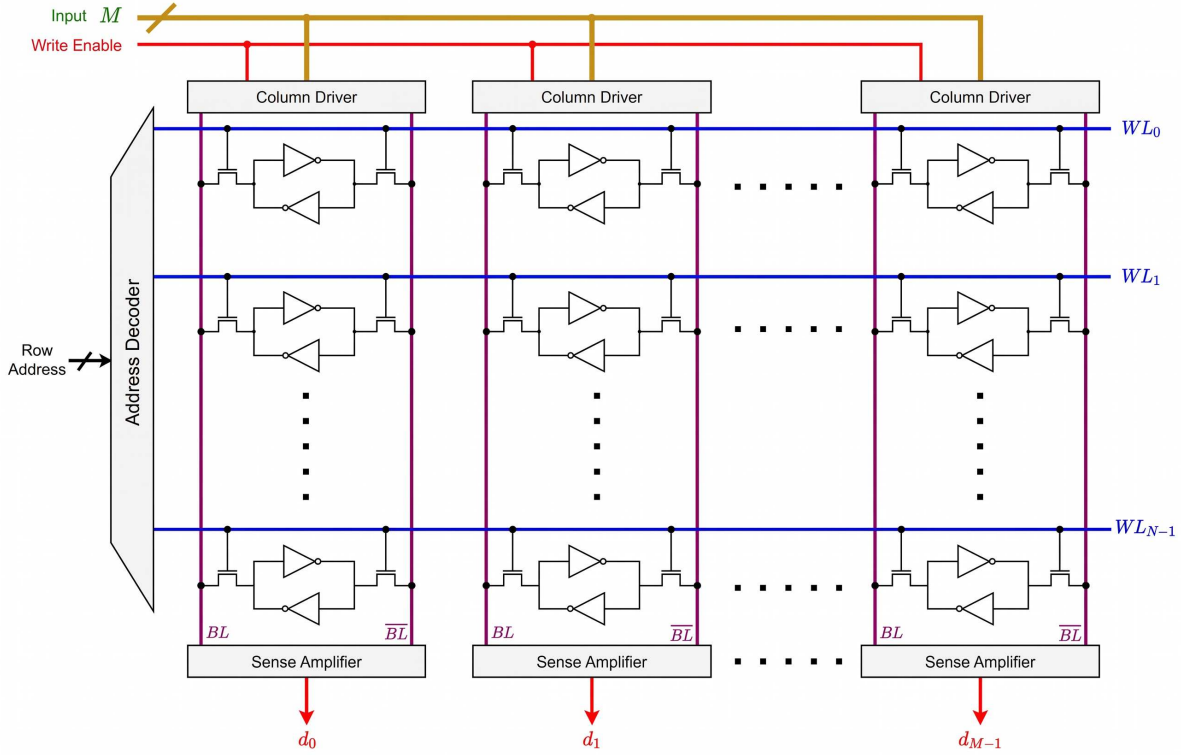


Figure 2.4: Architecture of an SRAM array.

2.2.3 SRAM Operations

The basic operations of an SRAM cell are read, write, and hold. These operations are controlled by the word line and the bit lines connected to the cell. In the hold state, the word line remains disabled, and the access transistors isolate the cell from the bit lines. The cross-coupled inverter pair then maintains the stored value through positive feedback [18, 19].

In a read operation, the bit lines are typically precharged before the word line is asserted. Activating the word line connects the internal storage nodes of the selected cell to the bit lines through the access transistors. Depending on the stored logic value, one of the bit lines experiences a slight voltage change, which is then detected and amplified by the sense amplifier [20, 21].

In a write operation, the desired data and its complement are driven onto the bit lines, and the word line is enabled. The externally driven bit-line values overpower the previous state of the

cross-coupled inverters and force the cell into the new logic state. Once the write is complete and the word line is deasserted, the newly stored value is retained by the bistable inverter pair [20, 21].

These three operations form the basis of SRAM functionality. Since the read, write, and hold conditions depend on transistor strengths, bit-line behavior, and node stability, understanding them is essential for analyzing the reliability, power-up behavior, and aging effects of SRAM cells. For this reason, the following subsections discuss read, write, and hold operations in detail [18, 19].

2.2.4 Read Operation

The read operation in a 6T SRAM cell is performed by sensing the logic value stored at the internal storage nodes Q and $\overline{Q}$ without intentionally changing the stored state. The transistor-level structure of the SRAM cell used for this operation is shown in Figure 2.2(a), where A_1 and A_2 are the access transistors controlled by the word line (WL), and BL and $\overline{BL}$ are the complementary bit lines [20, 21].

To explain the read process clearly, let us first assume that the cell is storing

$$Q = 1 \quad \text{and} \quad \overline{Q} = 0.$$

In this state, the left storage node Q is at a high voltage close to V_{DD} , while the complementary node $\overline{Q}$ is at a low voltage close to ground. Before the word line is enabled, both bit lines BL and $\overline{BL}$ are precharged to V_{DD} . This precharge step is important because it creates a well-defined starting condition for sensing a small voltage difference during the read operation. At this time, the word line is low, and therefore the access transistors A_1 and A_2 remain OFF, isolating the cell from the bit lines [18, 20].

Once the bit lines are precharged, the read operation begins by asserting the word line WL . Turning WL high switches ON the access transistors A_1 and A_2 , thereby connecting the internal storage nodes Q and $\overline{Q}$ to BL and $\overline{BL}$, respectively. Since $Q = 1$, the bit line BL remains close to its precharged value because it is connected to a high storage node. However, since $\overline{Q} = 0$, the complementary bit line $\overline{BL}$ begins to discharge slightly through the path formed by A_2 and the

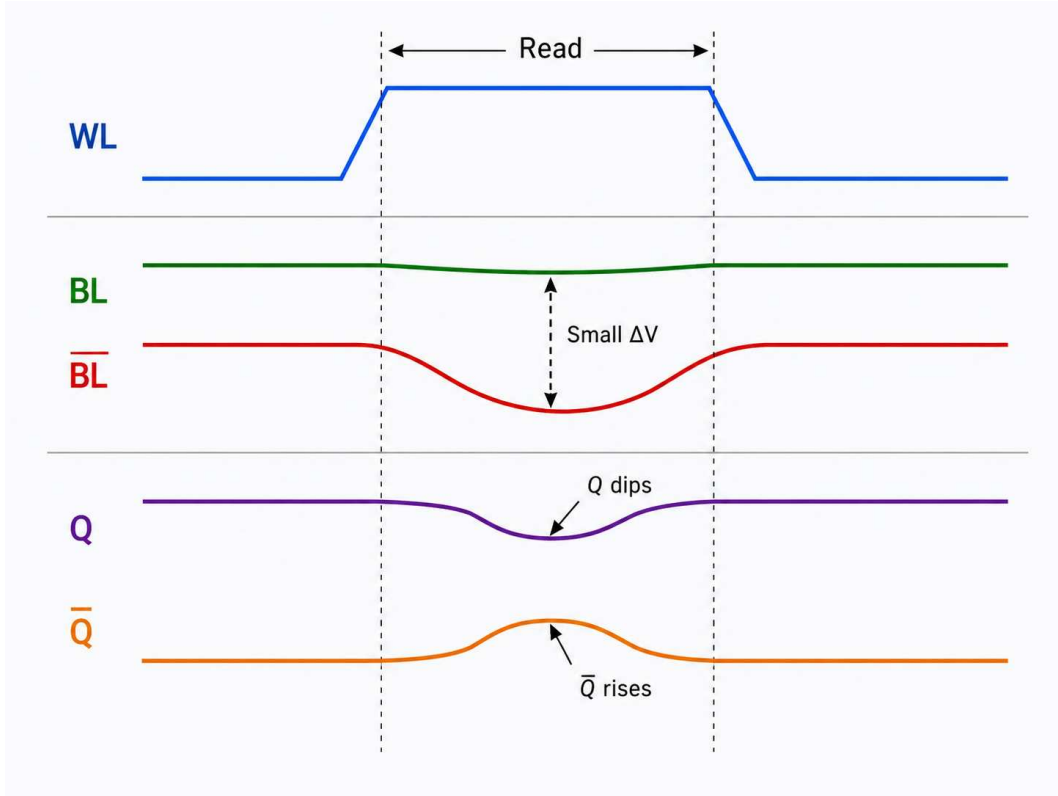


Figure 2.5: Timing diagram during a 6T SRAM read operation for the case where $Q = 1$ and $\bar{Q} = 0$.

pull-down NMOS transistor N_2 . As a result, a small voltage difference develops between BL and $\bar{BL}$, with BL remaining near V_{DD} and $\bar{BL}$ dropping slightly below it [19,21].

Figure 2.5 shows the voltage behavior of the word line, bit lines, and internal storage nodes during the read operation. During the read interval, the low storage node rises slightly and the high storage node dips slightly, but the cross-coupled inverter pair restores the original stored state after the word line is deasserted.

This small differential voltage is then detected by the sense amplifier connected to the bit-line pair. The sense amplifier amplifies the small voltage difference into a full logic level, thereby determining that the cell stores $Q = 1$. After the data is sensed, the word line is deasserted, which turns OFF the access transistors and disconnects the cell from the bit lines. The bit lines are then restored and precharged again in preparation for the next operation [20,21].

If the stored value were reversed, that is,

$$Q = 0 \quad \text{and} \quad \overline{Q} = 1,$$

then the same process would occur in the opposite direction. In that case, BL would discharge slightly while $\overline{BL}$ would remain near V_{DD} , and the sense amplifier would detect that the stored value is logic ‘0’.

It is important to note that the read operation must be carefully designed so that the act of reading does not disturb the stored data. Because the internal node storing logic ‘0’ is directly connected to a precharged bit line during read, there is a possibility that this node may rise slightly in voltage. If this rise becomes too large, the stored state can be disturbed. This effect is known as read disturb, and it is one of the key considerations in SRAM design. Therefore, proper transistor sizing and cell stability are critical for ensuring reliable read operation [18, 19].

2.2.5 Write Operation

The write operation is used to force a new logic value into the SRAM cell by overriding its previously stored state. As shown in Figure 2.2(a), the write operation also uses the access transistors A_1 and A_2 , the word line WL , and the complementary bit lines BL and $\overline{BL}$. Unlike the read operation, where the cell only slightly influences the bit lines, the write operation relies on the external bit-line drivers to overpower the internal feedback of the cross-coupled inverters and impose a new state [20, 21].

To explain the write process clearly, let us assume that the cell is initially storing

$$Q = 1 \quad \text{and} \quad \overline{Q} = 0,$$

and the goal is to write a logic ‘0’ into node Q . In order to do this, the write drivers first force the bit lines to the values

$$BL = 0 \quad \text{and} \quad \overline{BL} = 1.$$

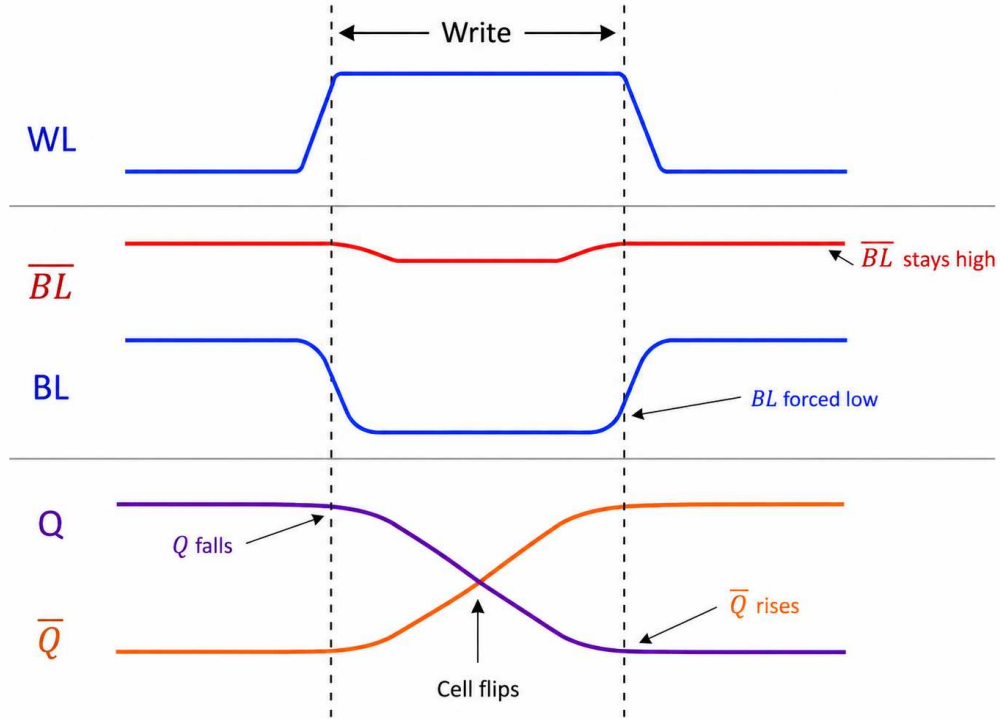


Figure 2.6: Timing diagram during a 6T SRAM write operation for the case where a logic ‘0’ is written into node Q .

This means that the bit line connected to node Q is driven low, while the complementary bit line connected to $\overline{Q}$ is driven high. At this stage, the word line is still low, so the access transistors remain OFF and the internal state of the cell is unchanged [18, 20].

The actual write operation begins when the word line WL is asserted. Once WL goes high, the access transistors A_1 and A_2 turn ON and connect the driven bit lines directly to the internal storage nodes. Since $BL = 0$, node Q is pulled downward through A_1 . At the same time, since $\overline{BL} = 1$, node $\overline{Q}$ is pulled upward through A_2 . As the voltage at Q begins to fall, the cross-coupled feedback structure shown in Figure 2.2(b) reinforces the transition. Once Q drops sufficiently, the inverter connected to that node causes $\overline{Q}$ to rise, and the complementary inverter then further drives Q downward. This regenerative feedback rapidly flips the cell from its original state to the new state

$$Q = 0 \quad \text{and} \quad \overline{Q} = 1.$$

Figure 2.6 illustrates the voltage transitions at the word line, bit lines, and internal storage nodes during a write operation. During the write interval, the externally driven bit lines overpower the previous cell state, forcing Q low and $\overline{Q}$ high through the access transistors.

After the new state has been established, the word line is deasserted, which turns OFF the access transistors and isolates the cell from the bit lines. The newly written value is then preserved by the bistable cross-coupled inverter pair. If the desired write value were the opposite, namely writing a logic ‘1’ into node Q , then the bit-line drivers would set

$$BL = 1 \quad \text{and} \quad \overline{BL} = 0,$$

and the same process would occur in the opposite direction [19, 21].

The success of the write operation depends on the bit-line drivers being strong enough to overcome the previous state stored in the cross-coupled inverters. If the cell is too strong relative to the access devices and write drivers, then changing the stored value becomes difficult. This property is commonly referred to as write ability, and it is a major design consideration in SRAM cell optimization [18, 20].

2.2.6 Hold Operation

The hold operation refers to the condition in which the SRAM cell preserves its stored data while no read or write access is taking place. This is the normal standby operating state of the SRAM cell when power is present but the cell is not actively selected. The hold behavior is fundamentally enabled by the positive feedback of the two cross-coupled inverters shown in Figure 2.2(b) [18, 19].

Let us again assume that the cell is storing

$$Q = 1 \quad \text{and} \quad \overline{Q} = 0.$$

In this case, the inverter connected to node $\overline{Q}$ sees a low input and therefore drives node Q high. At the same time, the other inverter sees a high input from node Q and therefore drives node $\overline{Q}$ low. In transistor terms, as shown in Figure 2.2(a), when $Q = 1$ and $\overline{Q} = 0$, PMOS transistor P_1 and NMOS transistor N_2 are ON, while PMOS transistor P_2 and NMOS transistor N_1 are OFF. This arrangement keeps node Q connected toward V_{DD} and node $\overline{Q}$ connected toward ground, thereby maintaining the stored logic state [20, 21].

During the hold condition, the word line WL remains low. As a result, the access transistors A_1 and A_2 are OFF, which isolates the cell from the external bit lines BL and $\overline{BL}$. Since the storage nodes are disconnected from the bit lines, no external read or write disturbance affects the cell. The data is preserved entirely by the internal regenerative feedback loop formed by the two inverters [18, 19].

If the stored value is instead

$$Q = 0 \quad \text{and} \quad \overline{Q} = 1,$$

then the conducting and non-conducting transistors are reversed, but the same bistable principle applies. In both cases, the cell remains in one of the two stable operating points shown in Figure 2.2(b). As long as the supply voltage is maintained and the cell is not disturbed by access operations or excessive noise, the stored state will persist [20, 21].

The hold operation is therefore the fundamental reason SRAM can store data without refresh. Unlike DRAM, which relies on a charge stored on a capacitor and must be periodically refreshed, SRAM maintains its data using active bistable feedback. However, the ability of the cell to remain in the correct state during hold depends on transistor matching, supply voltage, and noise conditions. This property is commonly referred to as hold stability, and it becomes especially important when considering process variation, voltage scaling, and aging effects in SRAM arrays [18, 19].

2.2.7 Stability and Bistability of SRAM Cells

The ability of an SRAM cell to reliably store data is based on its bistable nature. A conventional 6T SRAM cell consists of two cross-coupled inverters, which create a positive-feedback loop and

allow the cell to remain in one of two stable operating states. These two stable states correspond to the storage of logic '0' and logic '1'. Because of this bistable behavior, once the cell settles to one of these two states, it continues to hold that value as long as power is maintained and no external disturbance is strong enough to change the stored information.

This bistability is fundamental to SRAM operation because it ensures that a single bit can be stored robustly without refresh. In one stable state, node Q is at a high voltage and node $\overline{Q}$ is at a low voltage, while in the other stable state the voltage levels are reversed. The cross-coupled inverter structure continuously reinforces the stored value through regenerative feedback, thereby maintaining the cell state during the hold condition. However, the practical reliability of this stored state depends on how strongly the cell can resist unwanted disturbances caused by noise, supply variation, or device mismatch.

The stability of an SRAM cell is commonly evaluated using the Static Noise Margin (SNM), which represents the maximum dc noise voltage that the cell can tolerate without losing its stored data. A larger SNM indicates that the cell is more robust against noise and unwanted state flipping, whereas a smaller SNM means the cell is more vulnerable to disturbance. Therefore, SNM is one of the most important parameters used to assess SRAM cell reliability.

To quantify SNM, the voltage-transfer characteristics (VTCs) of the two cross-coupled inverters are plotted in mirrored form, producing the well-known butterfly curve shown in Figure 2.7. The two lobes of the butterfly curve correspond to the two stable operating regions of the SRAM cell. The side length of the largest square that can be embedded within either lobe is defined as the static noise margin. Physically, this square represents the maximum tolerable dc disturbance before the cell becomes unstable and transitions away from its stored state.

A high SNM is desirable because it indicates strong immunity to noise and improved stability during SRAM operation. In practice, however, the SNM of an SRAM cell is affected by a number of factors, including transistor sizing, process variation, supply voltage fluctuations, temperature changes, and aging-induced threshold-voltage shifts. Variations in these parameters can distort the butterfly curve and reduce the available SNM, thereby increasing the likelihood of read failure,

write failure, or unintended power-up bias. For this reason, SRAM stability is closely tied to process, voltage, and temperature (PVT) variation, and it also plays an important role in understanding aging-induced imprinting effects in SRAM arrays.

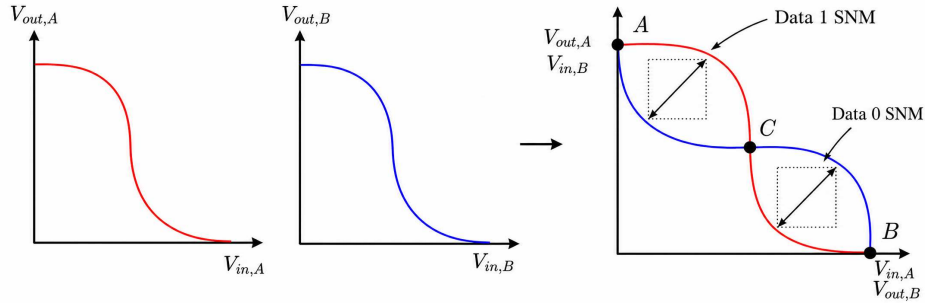


Figure 2.7: Butterfly curve used to evaluate SRAM static noise margin.

2.2.8 SRAM Leakage

Leakage current is an important non-ideal effect in SRAM cells because it influences both data retention and cell stability. Referring again to the conventional 6T SRAM cell shown in Figure 2.2(a), the cell consists of two pull-up PMOS transistors (P_1 and P_2), two pull-down NMOS transistors (N_1 and N_2), and two access transistors (A_1 and A_2). In an ideal SRAM cell, the stored state would remain unchanged indefinitely as long as power is supplied. In practice, however, several leakage mechanisms exist in the transistors, causing unwanted current flow even when the devices are intended to be OFF. These leakage components become increasingly important in deeply scaled CMOS technologies and are closely related to SRAM stability, retention, and remanence behavior.

Among the different leakage components in CMOS devices, subthreshold leakage is one of the most important in SRAM analysis. Subthreshold leakage arises when a MOS transistor operates below its threshold voltage but still allows a small diffusion current to flow between source and drain. In the hold state, where the word line is disabled and the cell is isolated from the bit lines, one or more transistors in the 6T cell remain nominally OFF. Even in this OFF condition, a weak

current can flow through the pull-down NMOS transistors and the access transistors, leading to gradual charge loss or redistribution at the internal storage nodes.

The subthreshold leakage current can be approximately expressed as

$$I_{\text{sub}} = I_0 \exp \left(\frac{V_{GS} - V_{th}}{nV_T} \right), \quad (2.1)$$

where I_0 is a device-dependent pre-exponential factor, V_{GS} is the gate-to-source voltage, V_{th} is the threshold voltage of the transistor, n is the subthreshold swing factor, and V_T is the thermal voltage given by $V_T = kT/q$. This equation shows that the leakage current depends exponentially on the difference between V_{GS} and V_{th} , making it highly sensitive to threshold-voltage variation and temperature.

Another important quantity related to leakage is the subthreshold slope, which describes how effectively a transistor can be turned off. It is given by

$$SS = \frac{dV_G}{d(\log_{10} I_D)} \approx nV_T \ln(10), \quad (2.2)$$

where I_D is the drain current. A smaller value of SS indicates sharper switching behavior and better suppression of leakage current in the OFF region. The factor n in (2.2) depends on the capacitive coupling within the MOS device and can be expressed as

$$n = 1 + \frac{C_{\text{dep}}}{C_{\text{ox}}}, \quad (2.3)$$

where C_{dep} is the depletion capacitance and C_{ox} is the gate oxide capacitance. This relation shows that the subthreshold behavior is influenced by device structure and process parameters, and therefore varies across technology nodes.

In a typical SRAM hold state, the cross-coupled inverter pair continuously reinforces the stored value as long as power is applied. For example, if the cell stores $Q = 1$ and $\overline{Q} = 0$, the pull-up PMOS connected to the high node supplies current to maintain that node near V_{DD} , while the pull-down NMOS connected to the low node helps maintain the complementary node near ground. In

this condition, leakage through the OFF transistors is still present, but the active feedback of the cross-coupled inverters compensates for the lost charge and preserves the stored data. Thus, during normal powered hold operation, the effect of leakage is usually limited and does not immediately destroy the stored state.

The situation changes significantly when power is removed. Once V_{DD} is turned off, the pull-up PMOS transistors can no longer replenish the charge lost from the internal nodes. As a result, the voltages at the storage nodes begin to decay due to the combined effect of subthreshold leakage, junction leakage, and parasitic discharge paths. The rate of this decay depends on the leakage magnitude, the parasitic node capacitances, and the initial stored state. This behavior is particularly relevant when discussing data remanence in SRAM, because the residual voltage distribution during power-down and the subsequent power-up tendency can be influenced by these leakage-driven decay mechanisms.

Temperature plays an especially important role in SRAM leakage. From (2.1) and (2.2), it is evident that leakage current is strongly dependent on thermal voltage and therefore increases with temperature. At higher temperatures, the subthreshold current becomes larger, which accelerates charge loss and weakens data retention. Conversely, lower temperature reduces the leakage current and slows the decay of the internal node voltages. As a result, the probability of residual data persistence or remanence can become higher at lower temperatures, while elevated temperature generally accelerates data loss and can also interact with aging mechanisms such as NBTI.

Therefore, SRAM leakage is not only a circuit-level power concern, but also a phenomenon closely tied to stability, retention, and security. In the context of this thesis, leakage is important because it influences the persistence of stored states, the decay of internal nodes after power removal, and the conditions under which aging-induced imprinting and remanence effects may be observed.

2.2.9 Effects of Process Variation and Device Mismatch

Process variation is an inherent feature of semiconductor manufacturing and plays a major role in determining the electrical behavior of SRAM cells. During fabrication, parameters such as channel length, channel width, oxide thickness, threshold voltage, line-edge roughness, and doping concentration cannot be controlled perfectly. As a result, the transistors fabricated on a chip do not all have exactly identical characteristics. In SRAM, where each bit cell relies on a delicate balance between two cross-coupled inverters, even small transistor-level variations can significantly affect stability, read and write margins, leakage, and power-up behavior.

In a conventional 6T SRAM cell, the pull-up PMOS transistors, pull-down NMOS transistors, and access transistors are intended to be closely matched so that the two storage nodes behave symmetrically. In practice, however, mismatch between paired transistors introduces asymmetry into the cell. For example, if one pull-down NMOS transistor is slightly stronger than the other, or if one pull-up PMOS transistor has a slightly different threshold voltage, the regenerative feedback loop in the cross-coupled inverter pair becomes unbalanced. This imbalance can shift the preferred operating condition of the cell and influence how easily the cell settles to one stored state versus the other.

Process variation is commonly described at multiple spatial levels. As illustrated in Figure 2.8, variation may occur from lot to lot, wafer to wafer, chip to chip, and even within a single die. Lot-to-lot and wafer-to-wafer variations generally change the average characteristics of devices across larger manufacturing populations. Chip-to-chip variation causes one die to behave differently from another die on the same wafer. Within-die variation is especially important for SRAM because it directly affects the matching of neighboring transistors inside the same bit cell. These variations may be systematic, producing spatially correlated patterns across the die, or random, producing local mismatch that is difficult to predict from one cell to another.

One of the most important electrical effects of process variation is the change in transistor threshold voltage (V_{th}). Variations in V_{th} alter the voltage at which a transistor turns on and therefore change the relative current-driving strength of the PMOS and NMOS devices inside the SRAM

cell. Similarly, variation in geometry and doping can affect carrier mobility and effective channel resistance, which in turn changes the drain current of the transistor. Because SRAM operation depends on the competition between matched transistors, these parameter shifts can reduce symmetry in the cell and modify its noise margins and timing behavior.

This effect is particularly important from the perspective of SRAM stability. A mismatched cell typically exhibits a distorted butterfly curve and reduced static noise margin (SNM), meaning that it can tolerate less noise before its stored state is disturbed. Process variation can therefore make some SRAM cells more vulnerable to read disturb, write failure, or unintended state flipping. In addition, the same mismatch that affects SNM can also influence the power-up state of the SRAM cell. Since the power-up behavior of a cross-coupled inverter pair depends on tiny differences in transistor strengths, even a small mismatch can bias the cell toward one preferred power-up state. This is one of the main physical reasons why SRAM power-up states can be used as hardware fingerprints and why different cells exhibit non-uniform power-up behavior.

Figure 2.8(e) also illustrates the concept of process corners, which are commonly used to represent combinations of NMOS and PMOS parameter variations. The typical corner is denoted as TT, where both NMOS and PMOS devices exhibit typical behavior. The FF and SS corners correspond to both transistor types being fast or slow, respectively, while SF and FS represent mixed conditions in which one device type is slow and the other is fast. These corners are useful for analyzing worst-case behavior in digital and memory circuits because they capture how device-parameter variation maps into performance variation. In SRAM design, process corners are especially important for evaluating read stability, write ability, leakage, and power-up behavior under different manufacturing extremes.

For the purposes of this thesis, process variation and device mismatch are fundamentally important because they establish the initial imbalance of the SRAM cell even before aging or radiation effects are introduced. In other words, each SRAM cell begins with a small built-in preference determined by fabrication-induced mismatch. Aging mechanisms such as NBTI and radiation-induced threshold-voltage shifts then modify this original mismatch over time. Therefore, under-

standing process variation is essential for explaining both the natural power-up bias of SRAM cells and the way in which aging or sanitization mechanisms alter that bias in later chapters.

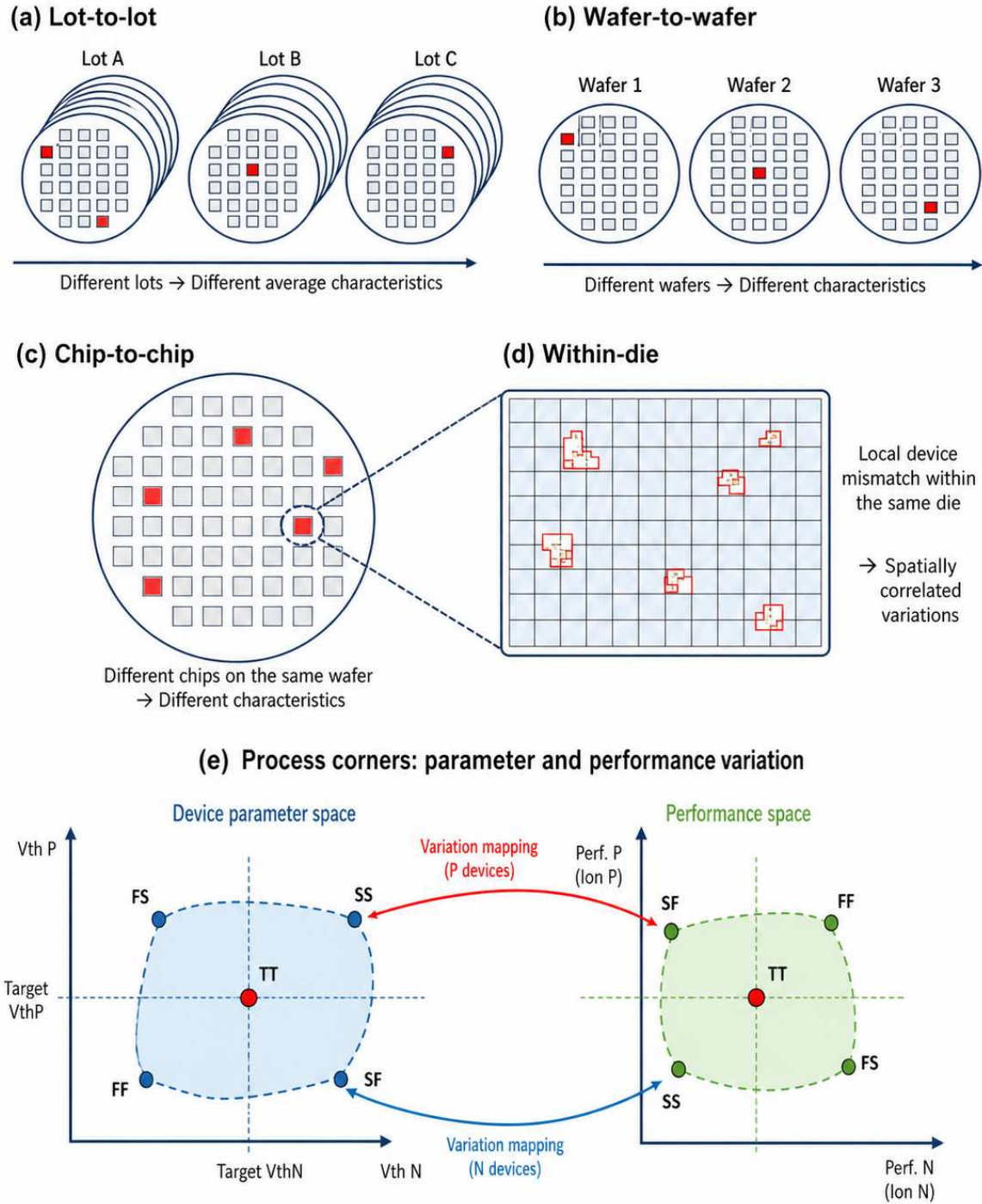


Figure 2.8: Illustration of process variation and device mismatch in CMOS: (a) lot-to-lot variation, (b) wafer-to-wafer variation, (c) chip-to-chip variation, (d) within-die variation, and (e) mapping of device-parameter variation to performance variation using CMOS process corners.

2.3 Aging Mechanisms in CMOS and SRAM

2.3.1 Overview of Semiconductor Aging Effects

Semiconductor devices do not remain electrically constant throughout their operating lifetime. Under prolonged electrical stress, elevated temperature, and repeated switching activity, MOS transistors gradually experience parameter drift that alters threshold voltage, carrier mobility, transconductance, leakage behavior, and timing response. In scaled CMOS technologies, the most widely discussed wear-out mechanisms include bias temperature instability (BTI), hot-carrier injection (HCI), time-dependent dielectric breakdown (TDDB), and electromigration (EM) [22]. These mechanisms do not affect all circuits equally; their severity depends on operating bias, workload, duty cycle, temperature, and technology node.

From the perspective of SRAM, aging is especially important because even a small change in the characteristics of one transistor can disturb the balance of the cross-coupled inverter pair. An SRAM cell is fundamentally a bistable circuit whose correct operation depends on close matching between the pull-up PMOS devices, pull-down NMOS devices, and access transistors. When aging modifies one device more strongly than its complementary counterpart, the symmetry of the cell is disturbed. This altered symmetry can degrade static noise margin, reduce read stability, affect write ability, and modify the natural power-up tendency of the cell [23, 24]. Therefore, while several aging mechanisms exist in CMOS, the mechanisms that directly change threshold voltage and current drive are the most relevant to SRAM reliability and security.

Among the major wear-out mechanisms, bias temperature instability is the most important for the present thesis. BTI directly shifts transistor threshold voltage under long-term gate bias and elevated temperature, and it may also exhibit partial recovery after the stress is removed [25, 26]. In digital logic, BTI is often discussed in terms of timing degradation. In SRAM, however, the same underlying physics has a more direct structural consequence: it changes the relative strength of the two sides of the storage cell. Since this thesis focuses on aging-induced data imprinting and power-up-state modification in SRAM, BTI provides the most relevant physical framework for analyzing how long-term stress changes cell behavior.

2.3.2 Negative Bias Temperature Instability (NBTI)

Negative Bias Temperature Instability (NBTI) primarily affects PMOS transistors under negative gate bias and elevated temperature. When a PMOS device remains in the ON state for extended periods under such stress, its threshold-voltage magnitude tends to increase and its effective drive capability tends to weaken. This degradation is widely associated with a combination of interface-state generation and charge trapping in the gate dielectric stack [25, 26]. At the circuit level, the result is that a stressed PMOS transistor becomes less effective at pulling its output node toward V_{DD} .

NBTI is strongly dependent on bias, temperature, and time. Higher temperature and stronger stress bias accelerate the degradation, while the accumulated threshold-voltage shift is often modeled using a power-law dependence on stress time [26]. Another important characteristic of NBTI is partial recovery. When the stress is relaxed or removed, part of the threshold-voltage shift may recover, although not always completely [25, 26]. This recoverable component is particularly relevant in circuits that alternate between active and idle operating modes, because the effective degradation depends not only on elapsed time but also on signal probability and duty cycle.

In SRAM cells, NBTI is highly important because the two pull-up PMOS transistors do not necessarily experience the same long-term stress history. If one side of the cell repeatedly stores a logic state that keeps one PMOS transistor under stronger stress than its complementary device, the resulting threshold-voltage mismatch grows over time. That mismatch weakens one side of the cell and strengthens the relative dominance of the other side. As a consequence, the cell becomes increasingly asymmetric. This asymmetry affects read and write behavior, reduces stability margins, and can bias the preferred power-up state of the cell [23, 24]. For this reason, NBTI is one of the most natural physical explanations for aging-induced data imprinting in SRAM.

2.3.3 Positive Bias Temperature Instability (PBTI)

Positive Bias Temperature Instability (PBTI) is the corresponding BTI mechanism associated with NMOS transistors under positive gate bias, especially in technologies employing high- k gate

dielectrics and metal-gate stacks. Under prolonged positive gate stress and elevated temperature, trapped charge and oxide-related defects can shift the NMOS threshold voltage and degrade device performance [27, 28]. Compared with NBTI, PBTI became much more prominent after the introduction of high- k /metal-gate CMOS, where electron trapping in the dielectric stack emerged as a significant reliability concern for nMOS devices.

At the circuit level, PBTI reduces the effective strength of stressed NMOS devices by altering their threshold voltage and drive current. In SRAM, this degradation can appear in the pull-down network and, depending on the design, also in the access transistors. Since SRAM read stability depends strongly on the ability of the pull-down NMOS devices to hold a stored '0' against a precharged bit line, and write ability depends on the competition between internal devices and external bit-line drivers, PBTI can influence both read and write margins [23, 29]. Although NBTI is often the dominant concern in conventional bulk-CMOS SRAM because it strongly impacts the PMOS pull-up pair, PBTI becomes increasingly relevant in advanced gate stacks and in operating scenarios where the NMOS devices spend long intervals under positive bias.

For the purposes of this thesis, the significance of PBTI is not that it replaces NBTI as the dominant aging mechanism, but that it complements the reliability picture. NBTI changes the PMOS balance of the storage latch, while PBTI can alter the NMOS balance and the access-path behavior. Together, these effects determine how the electrical symmetry of the 6T cell evolves with time.

2.3.4 Impact of Aging on SRAM Cell Behavior

The most direct impact of aging on an SRAM cell is the gradual loss of symmetry between the two cross-coupled inverter branches. In a fresh cell, the pull-up PMOS pair, pull-down NMOS pair, and access devices are designed to be closely matched, with only small fabrication-induced mismatch. Aging changes this balance by shifting the threshold voltage of stressed devices and altering their current-drive strength. Once the two sides of the cell are no longer equally strong, the static transfer characteristics of the cell become skewed, the butterfly curve becomes distorted,

and the available static noise margin can be reduced [23]. As a result, the cell becomes more vulnerable to read disturb, write failure, or state instability under noise and supply fluctuation.

Several studies specifically report that BTI aging degrades SRAM read and write robustness. NBTI has been shown to affect SRAM read stability and reduce the static noise margin of the cell [24]. Combined NBTI/PBTI stress further degrades static and dynamic noise margins and increases cell failure probability over time [23]. In addition, studies on power-gated SRAM report that PBTI can significantly worsen read and write behavior and further amplify the overall margin loss when combined with NBTI [29]. These results show that aging is not merely a device-level reliability concern; it produces measurable circuit-level consequences in SRAM operation.

For this thesis, one additional consequence is especially important: aging does not only change access margins, it can also alter the power-up preference of the cell. A cross-coupled SRAM cell resolves to one of two stable states during power-up based on small differences in transistor strength. If aging systematically weakens one PMOS or NMOS device more than its counterpart, the preferred power-up direction of the cell can shift over time. This provides the physical bridge between classical reliability literature and the data-imprinting phenomenon studied in this work. In other words, the same BTI-induced mismatch that reliability engineers observe as margin degradation can also be interpreted as a change in the natural power-up-state bias of the memory cell. This interpretation is central to understanding how previously stored data can become imprinted into the future power-up state of an SRAM array.

2.4 Related Works

Prior research relevant to this thesis spans three closely connected areas: SRAM power-up behavior as a security primitive, aging-induced modification of SRAM power-up states, and radiation-induced changes in SRAM cell behavior. Early work on SRAM power-up behavior established that fabrication-induced device mismatch causes each memory array to exhibit a characteristic and partially repeatable power-up pattern, enabling SRAM to be used for hardware identification, true-random-number generation, and Physical Unclonable Functions (PUFs) [?, ?, 6–8]. These studies

showed that the power-up response of SRAM is not merely random noise, but a physical signature that depends on local transistor mismatch, environmental conditions, and sensing methodology.

A separate line of work later showed that the SRAM power-up state can also become a security liability after prolonged bias stress. Hovanes *et al.* showed that used SRAMs can retain traces of previously stored information in their subsequent power-up states after high-temperature stress, and demonstrated partial recovery of imprinted data using a fresh reference PUS [10]. Garg *et al.* further reported that device aging can be deliberately used to bias SRAM power-up behavior, demonstrating that supply-voltage-accelerated degradation can make the power-up response more predictable [30]. More recently, Mahmood and Hicks examined accelerated aging as an attack surface for exposing secrets from on-chip SRAM and showed that power-up behavior can be manipulated and exploited in practical attack scenarios [12,31,32]. These works collectively establish that SRAM aging is not only a reliability issue, but also a serious end-of-life and run-time security concern.

Recent work has also advanced data-recovery methodology beyond the original reference-based approach. Tisha *et al.* developed recovery strategies that infer information from aged SRAM arrays without relying on a pristine reference PUS, thereby making post-aging data extraction more practical in realistic settings [33–35]. In parallel, prior studies on memory-aging signatures have shown that power-up-state bias can also be used in the opposite direction, namely, as a means of identifying reused or recycled chips. In particular, aging-induced bias in memory cells has been exploited for recycled-system detection, and self-referencing approaches using memory power-up states have been proposed for COTS SRAM screening and classification [13, 14]. These studies reinforce the central role of power-up-state bias as both a vulnerability and a diagnostic signature.

Another important body of related work concerns radiation effects in SRAM. Prior research has shown that ionizing radiation can alter SRAM behavior through charge trapping and interface-trap generation, thereby modifying threshold voltage, leakage, stability, and power-up behavior. Surendranathan *et al.* demonstrated that total ionizing dose can shift the SRAM power-up state and affect SRAM-based PUF integrity, with the response depending on stored data and technology

node [9, 15]. In later work, the same group showed that ionizing radiation can also produce data-imprinting effects in SRAM arrays, indicating that power-up bias is sensitive not only to thermal aging but also to radiation-induced device degradation [11]. Additional studies have examined radiation-induced stability degradation in scaled SRAM technologies and have reported measurable changes in retention behavior under high total ionizing dose [36, 37]. Together, these results suggest that radiation can significantly reshape the electrical balance of SRAM cells and therefore has implications for both security risk and mitigation.

Although these prior works have established that SRAM power-up behavior can be influenced by manufacturing variation, thermal aging, and ionizing radiation, important gaps remain. Existing studies typically focus on only one aspect at a time, such as fingerprint generation, thermal data recovery, recycled-chip detection, or radiation response. A unified experimental treatment that connects aging-induced imprint formation, technology-node dependence, long-term relaxation, and post-aging sanitization remains limited. In particular, the interaction between thermal aging and radiation as a possible end-of-life sanitization mechanism for previously imprinted SRAM has not been comprehensively explored across multiple commercial technologies. This thesis addresses that gap by experimentally studying aging-induced imprinting across multiple COTS SRAM devices and then investigating radiation-based sanitization as a practical mitigation strategy.

Chapter 3

Experimental Setup and Flow

This chapter describes the experimental framework developed to interface the standalone asynchronous SRAM devices used in this thesis and to evaluate their behavior under aging and sanitization conditions. The SRAM chips listed in Table 3.1 were interfaced with an Arduino Due to perform memory operations such as write, read, repeated power-up state acquisition, and post-stress characterization. The selected devices are Cypress 4-Mbit SRAMs organized as $256\text{K} \times 16$ with 18 address lines and standard asynchronous control signals, including Chip Enable (CE), Output Enable (OE), Write Enable (WE), and byte enables for the lower and upper data buses. To enable a comparative study of aging-induced effects and sanitization behavior across different generations of commercial off-the-shelf memory technologies, devices from 90 nm, 150 nm, and 250 nm process nodes were evaluated. All devices use the TSOP II package, which provides a consistent platform for controlled laboratory characterization and hardware-level analysis.

Table 3.1: Details of the SRAM devices used in the experimental evaluation

Part Number	Manufacturer	Memory Type	Organization	Tech. Node
CY7C1041CV33	Cypress	Asynchronous SRAM	$256\text{K} \times 16$	150 nm
CY7C1041DV33	Cypress	Asynchronous SRAM	$256\text{K} \times 16$	90 nm
CY7C1041G30-10ZSXE	Cypress	Asynchronous SRAM	$256\text{K} \times 16$	65 nm
IS61LV25616AL	ISSI	Asynchronous SRAM	$256\text{K} \times 16$	180 nm

Among these devices, the CY7C1041DV33 was used as the primary SRAM platform for most of the experiments in this thesis. The remaining devices were selected to maintain a common memory organization and interface while allowing the impact of process technology on aging and sanitization behavior to be examined. The next section describes the hardware platform developed

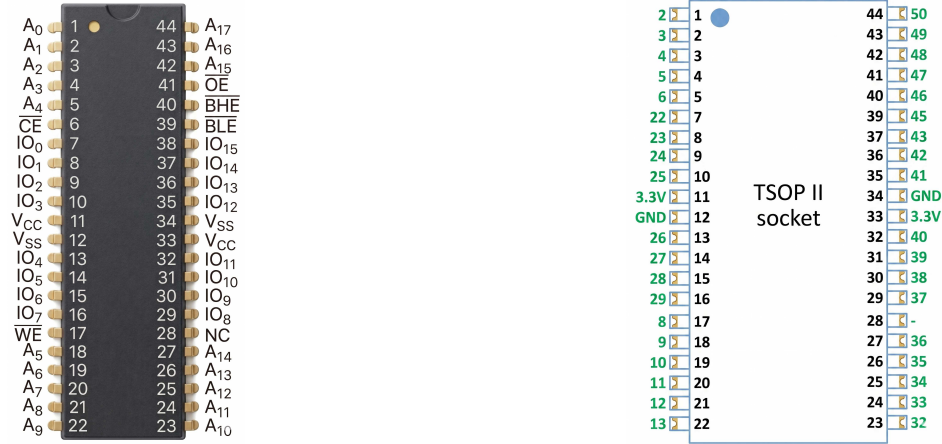
for SRAM interfacing, followed by the firmware structure used for memory access, power-up state collection, data logging, and debugging. The chapter then presents the validation of SRAM operation and interfacing reliability, which forms the experimental basis for the analyses reported in the subsequent chapters.

3.1 Hardware Setup

All SRAM devices used in this work are packaged in a 44-pin TSOP II form factor and share a common pin configuration, which allowed the same socket board and wiring scheme to be used across all experiments. Figure 3.1a shows the top-view pin configuration of the SRAM package used in this thesis. The device provides 18 address pins, A_0 through A_{17} , which are used to access the 262,144 word locations of the $256K \times 16$ memory array. Data transfer is performed through 16 bidirectional data pins, IO_0 through IO_{15} , where IO_0 – IO_7 form the lower byte and IO_8 – IO_{15} form the upper byte.

Memory operation is controlled using five active-low control signals: Chip Enable (CE), Output Enable (OE), Write Enable (WE), Byte Low Enable (BLE), and Byte High Enable (BHE). The CE pin is used to select the SRAM device, OE enables the output drivers during read operations, and WE controls write operations. The BLE and BHE pins independently enable the lower and upper byte lanes, respectively, allowing either byte-wise or full 16-bit word access. In addition to the address, data, and control pins, the package includes two power pins (VCC), two ground pins (VSS), and one no-connect (NC) pin. This standardized TSOP II pinout simplified the hardware interfacing process and provided a consistent physical interface for all SRAM chips evaluated in this work.

Figure 3.1b illustrates the initial hardware-level interfacing concept used to connect the Arduino Due to the TSOP II socket that holds the SRAM device. In this figure, the green labels represent the corresponding Arduino Due pins, while the inner socket outline represents the TSOP II socket pins used to access the SRAM chip. This pin-level mapping forms the primary step in the interfacing process, as it establishes the physical connection between the microcontroller and



(a) 44-pin TSOP II SRAM pin configuration.

(b) Arduino Due to TSOP II socket mapping.

Figure 3.1: SRAM pinout and initial Arduino Due interfacing map.

the memory device before implementing read, write, and power-up-state acquisition operations. The mapped connections include the supply and ground lines, along with the pins required for subsequent routing of address, data, and control signals between the Arduino Due and the SRAM. Establishing this one-to-one hardware connection was essential for ensuring reliable device access and for developing the firmware used in the later stages of the experimental setup.

Figure 3.1 shows the overall hardware setup developed for interfacing the SRAM devices used in this thesis with a host computer through an Arduino Due. Each SRAM chip is inserted into a TSOP II socket adapter, which routes the package pins to the corresponding Arduino Due pins for address, data, power, and control interfacing. The Arduino Due communicates with the host computer through USB and executes custom firmware for SRAM read, write, and repeated power-up state acquisition.

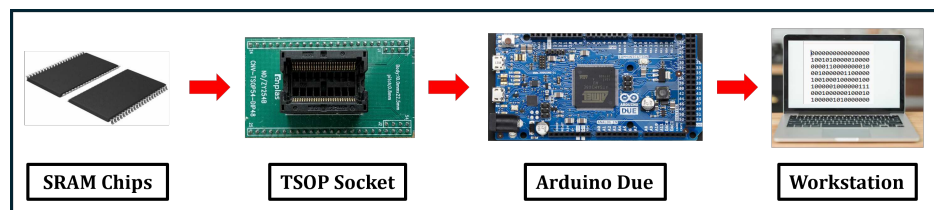


Figure 3.2: SRAM interfacing hardware setup.

During the initial phase of development, the hardware connections were implemented using a wired prototype to validate the pin mapping and verify correct SRAM operation. Although this approach was effective for early testing, the large number of parallel connections required for SRAM interfacing made the setup cluttered and difficult to maintain, as illustrated in Figure 3.2. The extensive wiring also increased the likelihood of noise, loose contacts, and signal integrity problems. In addition, the setup was not convenient to transport or repeatedly reassemble without introducing connection errors. For these reasons, a custom Printed Circuit Board (PCB) was designed to provide a more compact, stable, and reliable interfacing platform for the experiments in this work.

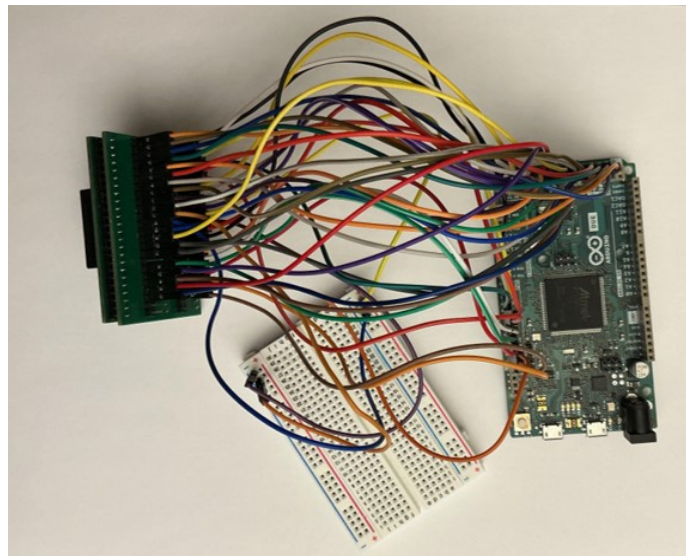


Figure 3.3: Wired prototype setup for initial SRAM interfacing.

3.2 PCB Design

A Printed Circuit Board (PCB) is a laminated structure composed of conductive copper layers and insulating substrate materials. The copper layers are patterned to form traces and planes that provide electrical interconnections between different components on the board. Compared to a manually wired prototype, a PCB offers a more compact, reliable, and mechanically stable solution for hardware interfacing. Electronic components are mounted and soldered onto designated pads, thereby establishing both electrical connectivity and physical support.

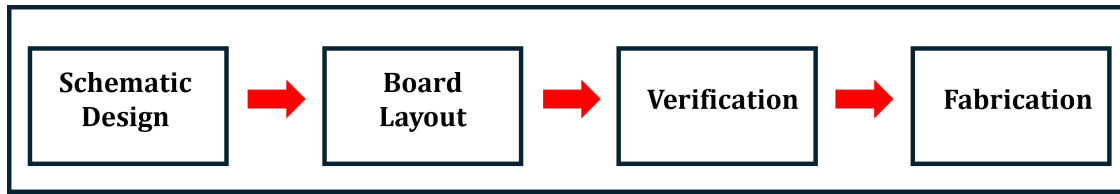


Figure 3.4: Design and development of the custom PCB used for SRAM interfacing.

The custom PCB used in this work was designed using Autodesk Eagle PCB design software. Figure 3.3 illustrates the main stages involved in the PCB design process, which are summarized as follows:

- **Schematic Design:** Selection of the required electronic components, creation of the circuit diagram, and assignment of signal names and component values.
- **Board Layout:** Physical placement of components on the board, routing of signal traces, and addition of power and ground connections.
- **Verification:** Electrical Rule Check (ERC) and Design Rule Check (DRC) to verify correctness and manufacturability.
- **Fabrication:** Generation of Gerber files for manufacturing the PCB.

The first step in the PCB design flow is the creation of a schematic representing the required hardware connections. A schematic is a circuit-level representation of the electronic components and their interconnections. The schematic of the custom board developed for SRAM interfacing is shown in Figure 3.4. It includes the Arduino Due, the TSOP II socket used to hold the SRAM chip, and dedicated header pins for monitoring selected control signals. The socket pins were mapped to the Arduino Due GPIO pins according to the required SRAM functionality, including address lines, bidirectional data lines, control signals, and the 3.3 V power and ground connections. Additional header pins were included to probe the key SRAM control signals, such as Chip Enable (CE), Write Enable (WE), Output Enable (OE), Byte Low Enable (BLE), and Byte High Enable (BHE).

A board layout is the physical realization of the schematic and shows the placement of components and routing of traces on the PCB. Figure 3.5 presents the layout of the custom SRAM interface board. The components were arranged to provide convenient connection between the Arduino Due and the SRAM socket while reducing routing complexity. Signal traces were routed on both the top and bottom copper layers, shown as red and blue traces, respectively. Care was taken during routing to reduce interference, maintain signal integrity, and support stable SRAM operation during repeated read, write, and power-up-state measurements.

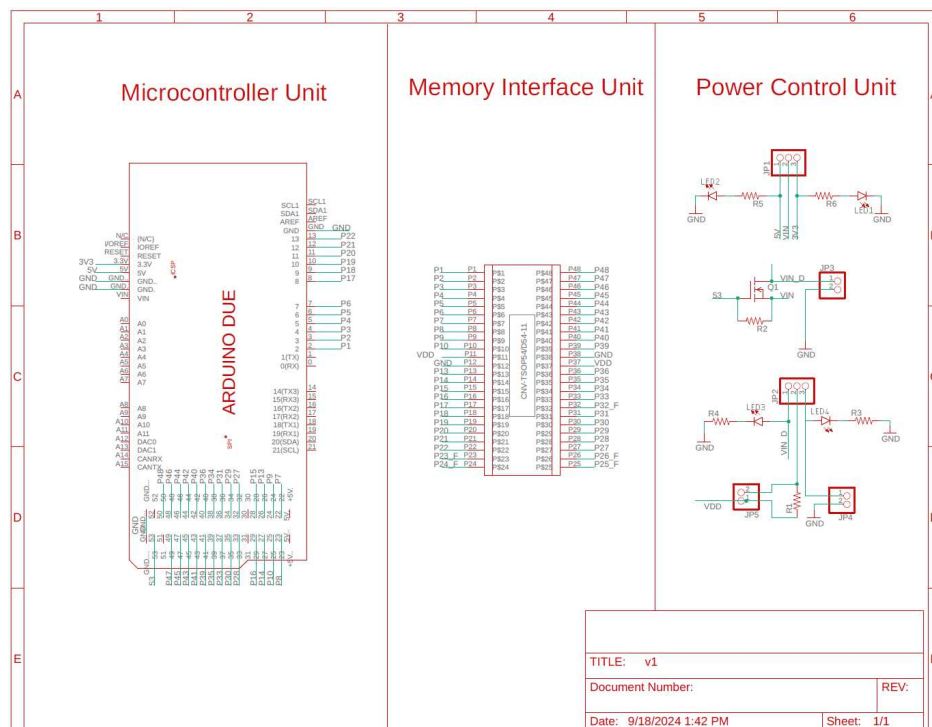


Figure 3.5: PCB schematic for SRAM interfacing.

When designing the PCB, it is necessary to follow layout rules and fabrication constraints to ensure that the board can be manufactured correctly and will operate reliably. Typical design constraints include minimum trace width and spacing, minimum drill size, clearance requirements, and the number of board layers. To verify compliance with these constraints, Autodesk Eagle provides Electrical Rule Check (ERC) and Design Rule Check (DRC) tools. Since the PCB developed in this work is a two-layer board, the corresponding manufacturer-specified DRC rules were

imported into the software before final verification. After the design passed both ERC and DRC without errors, Gerber files were generated and submitted for fabrication. These files contain the complete layer-by-layer information required for PCB manufacturing.

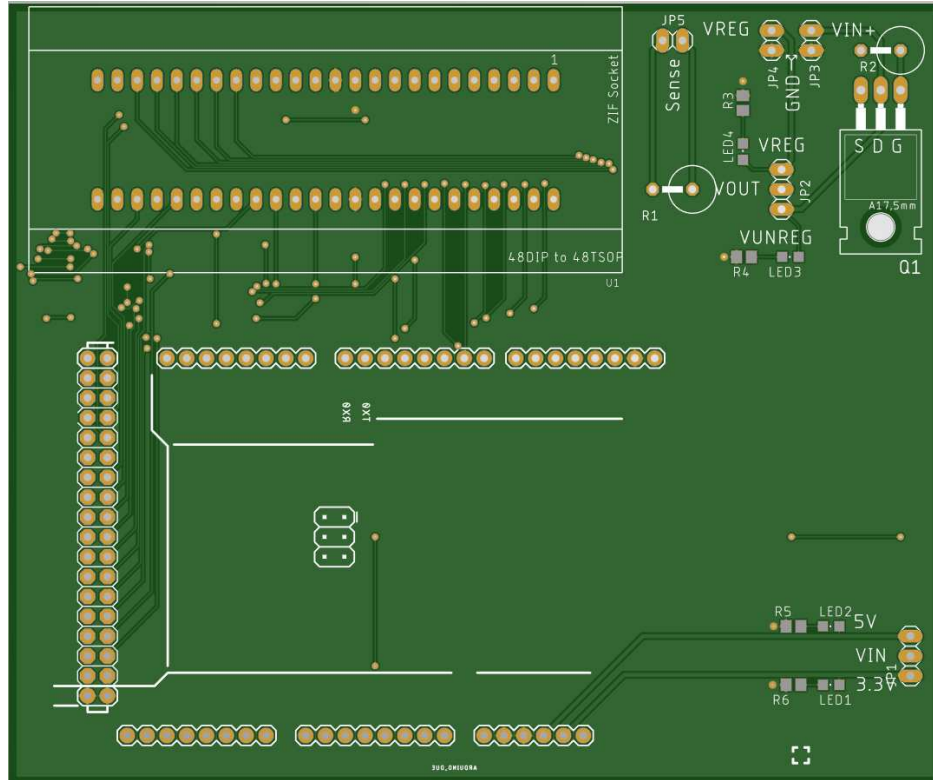


Figure 3.6: PCB layout for SRAM interfacing.

After fabrication, the Arduino Due headers, the TSOP II socket, and the required monitoring headers were assembled and soldered onto the custom PCB. This completed the hardware integration stage and provided a stable interface between the microcontroller and the SRAM device. Compared to the earlier wired prototype, the PCB-based setup improved connection reliability, reduced clutter, and provided better mechanical robustness for repeated experimental use. Figure 3.6 shows the final assembled PCB used in the experiments presented in this thesis. The PCB-based setup enabled reliable SRAM interfacing and supported repeatable measurements during aging and sanitization experiments.

3.3 Firmware Design

Firmware provides the low-level control required to operate the asynchronous SRAM devices through the Arduino Due platform. It acts as the interface between the microcontroller hardware and the memory device by managing signal initialization, address placement, data transfer, and execution of memory access cycles. In this work, the firmware was developed using the Arduino programming environment and used to perform the core SRAM interfacing operations required for the experimental setup, namely word read, word write, and sequential full-memory readout.

The SRAM devices used in this thesis have a common organization of $256\text{K} \times 16$, which requires 18 address inputs (A_0 – A_{17}), 16 bidirectional data pins (IO_0 – IO_{15}), and a set of active-low control signals to manage memory access. Accordingly, the firmware was designed to control the address bus, configure the direction of the bidirectional data bus, and apply the proper timing and logic levels to the control pins during each operation. A modular firmware structure was adopted so that the basic memory functions could be executed in a repeatable and organized manner.

The control pins used for SRAM interfacing are summarized below:

- **Chip Enable (CE):** This signal selects the SRAM device. A logic low on CE activates the chip, while a logic high deselects the device and places the outputs in a high-impedance state.
- **Write Enable (WE):** This signal controls write operations. When WE is driven low while the device is selected, the data present on the data bus is written into the memory location specified by the address lines.
- **Output Enable (OE):** This signal controls read access to the data bus. During read operation, OE is driven low so that valid data appears on the output lines.
- **Byte Low Enable (BLE):** This active-low signal enables the lower byte of the data bus, corresponding to IO_0 – IO_7 .

- **Byte High Enable (BHE):** This active-low signal enables the upper byte of the data bus, corresponding to IO₈–IO₁₅.

The combination of these control signals determines the type of memory operation performed by the SRAM. For a full-word read operation, the device is selected by driving CE low, the output drivers are enabled by driving OE low, the write path remains disabled by keeping WE high, and both byte-enable signals are asserted low so that the full 16-bit word appears on the data bus. For a full-word write operation, CE, WE, BLE, and BHE are driven low while the input data is applied on the bidirectional data bus and written into the address specified on A₀–A₁₇. When CE is high, the device is deselected and the outputs remain in a high-impedance state. The principal control-signal combinations used in the firmware are summarized in Table 3.2.

Table 3.2: Logic levels required on the SRAM control signals for principal bus operations

Operation	CE	WE	OE	BLE	BHE
Read (16-bit word)	LOW	HIGH	LOW	LOW	LOW
Write (16-bit word)	LOW	LOW	HIGH	LOW	LOW
Standby / Deselected	HIGH	X	X	X	X
Selected, outputs disabled	LOW	HIGH	HIGH	X	X

All five SRAM control signals are active low. In the firmware, each memory access begins by placing a valid address on the address lines. Depending on the desired operation, the data bus is then configured either as an input bus for read operations or as an output bus for write operations. The control signals are asserted according to the required read or write cycle, and after the access is completed, they are returned to their inactive state. Since the SRAM devices used in this work are asynchronous memories, correct operation depends on satisfying the required timing constraints such as address setup time, data setup time, access delay, and write pulse width. These timing requirements were incorporated into the firmware implementation to ensure stable operation during repeated measurements.

Pin Initialization and Default Bus Configuration

The firmware begins by initializing all Arduino Due pins used for SRAM interfacing. The SRAM requires 18 address lines, 16 bidirectional data lines, and five control signals, so the first task of the initialization routine is to assign the direction of each interface pin. The address lines A_0 through A_{17} are configured as output pins because the microcontroller must actively drive a valid memory address during both read and write operations. Similarly, the control pins CE, OE, WE, BLE, and BHE are configured as outputs so that the firmware can directly control the state of the SRAM bus.

After configuring the pin directions, the firmware establishes a default bus state that keeps the interface stable before any read or write cycle is initiated. The CE signal is asserted low so that the SRAM remains selected and available for access. At the same time, OE and WE are held high so that neither a read nor a write operation is triggered unintentionally during initialization. The BLE and BHE signals are driven low so that both bytes of the 16-bit word remain enabled for subsequent full-word access. This default condition prevents undefined behavior at power-up and provides a known and repeatable starting state for all later memory operations.

An additional aspect of initialization is the management of the bidirectional data bus. Because the SRAM data lines are used for both reading and writing, the firmware must later switch their direction dynamically depending on the type of access being performed. Although the actual direction change is handled inside the read and write routines, the initialization stage defines the overall interface framework and ensures that the control and address signals are placed in their correct default operating state before memory transactions begin.

SRAM Word Read Operation

The word read operation is used to retrieve a 16-bit data value from a selected SRAM location. This routine forms the basis for basic memory verification as well as complete array extraction. At the beginning of the read sequence, the firmware first ensures that the write path is disabled by driving WE high. This guarantees that the memory cycle will not be interpreted as a write

Algorithm 1 Pin Initialization and Default Bus Configuration

```
1: for each address pin  $A_0$  to  $A_{17}$  do
2:   Configure pin as output
3: end for
4: for each control pin CE, OE, WE, BLE, BHE do
5:   Configure pin as output
6: end for
7: Drive CE LOW
8: Drive OE HIGH
9: Drive WE HIGH
10: Drive BLE LOW
11: Drive BHE LOW
```

operation. The 16 data pins connected to the SRAM data bus are then configured as input pins so that the Arduino Due can receive the data driven by the memory device.

Once the data bus is configured for input, the target memory address is placed on the address lines A_0 through A_{17} . The firmware decomposes the address value into its individual bits and drives each bit onto the corresponding address pin. After the address has been made valid on the bus, OE is asserted low to enable the SRAM output drivers. Since the SRAM is an asynchronous device, the output data does not appear immediately; therefore, a short delay is inserted to allow the address and control signals to settle and for the data on the bus to become stable.

After the settling interval, the firmware samples all 16 data pins individually and combines their bit values into a single 16-bit word. Once the data has been reconstructed, OE is driven high again to disable the SRAM outputs and return the data bus to its inactive state. This prevents the memory from continuing to drive the bus when no read cycle is active. The final output of the routine is the 16-bit word stored at the selected address.

This read mechanism is central to the overall firmware architecture because it is reused in higher-level routines such as sequential full-memory readout. By following the same sequence each time, namely bus-direction configuration, address placement, output enable assertion, settling delay, and data capture, the firmware ensures that read access is performed in a consistent and repeatable manner across the full memory array.

Algorithm 2 SRAM Word Read Procedure

Require: Target address $addr$

Ensure: 16-bit data word $data$

- 1: Drive WE HIGH
 - 2: Configure data pins D_0 – D_{15} as input
 - 3: **for** $k = 0$ to 17 **do**
 - 4: Drive address line A_k with bit k of $addr$
 - 5: **end for**
 - 6: Drive OE LOW
 - 7: Wait for a short settling delay
 - 8: Read D_0 – D_{15} and combine them into a 16-bit word $data$
 - 9: Drive OE HIGH
 - 10: **return** $data$
-

SRAM Word Write Operation

The word write operation is used to store a 16-bit value into a specified SRAM address. This routine is required whenever the memory array must be initialized with a known data pattern before subsequent measurement or evaluation. At the start of the write cycle, the firmware first disables the SRAM output drivers by driving OE high. This is necessary to avoid any possibility of bus contention while the Arduino Due is preparing to place data onto the bidirectional data bus. The data pins are then configured as outputs so that the microcontroller can actively drive the desired write value onto the SRAM input lines.

The 16-bit input word is next decomposed into its individual bits, and each bit is assigned to the corresponding data pin connected to the SRAM bus. After the input data has been placed on the bus, the target address is driven onto the address lines A_0 through A_{17} in the same manner used for the read operation. A short delay is introduced at this stage to allow both address and data signals to stabilize before the write cycle is triggered.

The actual write operation is initiated by driving WE low while the chip remains selected and both byte-enable signals remain active. This causes the SRAM to capture the data present on the bus and store it at the address specified by the address lines. After holding the write condition for the required interval, WE is driven high again to terminate the write cycle. The data bus remains

driven by the firmware during this period so that the SRAM sees valid input data throughout the entire write interval.

This routine provides a deterministic method for loading the SRAM with controlled data patterns. It is therefore a fundamental part of the firmware framework, as many experimental procedures depend on first writing known values to the memory array before later reading them back. By enforcing a fixed write sequence consisting of output disable, bus-direction setup, data placement, address placement, and write-enable pulsing, the firmware achieves consistent write behavior over the full address range of the device.

Algorithm 3 SRAM Word Write Procedure

Require: Target address $addr$, input data word $data$

- 1: Drive OE HIGH
 - 2: Configure data pins D_0 – D_{15} as output
 - 3: Drive each bit of $data$ onto D_0 – D_{15}
 - 4: **for** $k = 0$ to 17 **do**
 - 5: Drive address line A_k with bit k of $addr$
 - 6: **end for**
 - 7: Wait for a short settling delay
 - 8: Drive WE LOW
 - 9: Wait for write completion delay
 - 10: Drive WE HIGH
-

Sequential Full-Memory Readout

In addition to single-word access, the firmware includes a routine for sequentially reading the entire SRAM array. This routine is particularly important for collecting complete memory images for later analysis. Rather than manually accessing individual word locations, the sequential readout routine automates the process by scanning the full address space from the first word address to the final valid location.

During execution, the firmware iterates through the SRAM word addresses one by one and invokes the single-word read routine at each address. The value returned by the read routine is then formatted and transmitted to the host computer through the serial interface. In the implementation

used in this work, the memory contents are sent in hexadecimal format, which provides a compact and convenient representation for storage, logging, and post-processing. Because the same underlying read routine is used for each address, the sequential readout process preserves consistency across the entire memory array.

This routine provides an efficient and repeatable method for full-array data acquisition. It ensures that the memory contents are captured in an orderly sequence and transferred directly to the host system without requiring repeated user interaction. The routine also benefits from the modular firmware structure, since any improvement made to the single-word read operation is automatically reflected in the full-memory readout mechanism. As a result, the sequential readout routine serves as a reliable high-level function for extracting the complete SRAM state during experimental evaluation.

Algorithm 4 Sequential SRAM Full-Memory Readout

Require: SRAM size N

- 1: **for** $i = 0$ to $N - 1$ **do**
 - 2: $data \leftarrow \text{ReadWord}(i)$
 - 3: Transmit $data$ over serial in hexadecimal format
 - 4: **end for**
-

SRAM Interface Validation

After completing the hardware development and firmware implementation, the SRAM interface was validated to ensure that the Arduino Due could correctly access the memory array and perform the required operations reliably. This validation step was necessary before proceeding to the aging and sanitization experiments, since any error in the interfacing framework could affect the correctness of the collected data. The validation focused on confirming correct firmware execution, successful acquisition of the initial SRAM power-up state, and reliable full-memory write and readback operation across the complete SRAM array.

The validation operations were performed through the serial-monitor-based firmware interface developed for the Arduino Due. This interface provided a set of selectable commands for SRAM

access and verification, including memory dump and full-memory pattern writing. Figure 3.7 shows the serial monitor output after firmware initialization, where the available SRAM access options are displayed. This menu-driven interface provided a simple and repeatable mechanism for invoking the validation procedures used in this work.

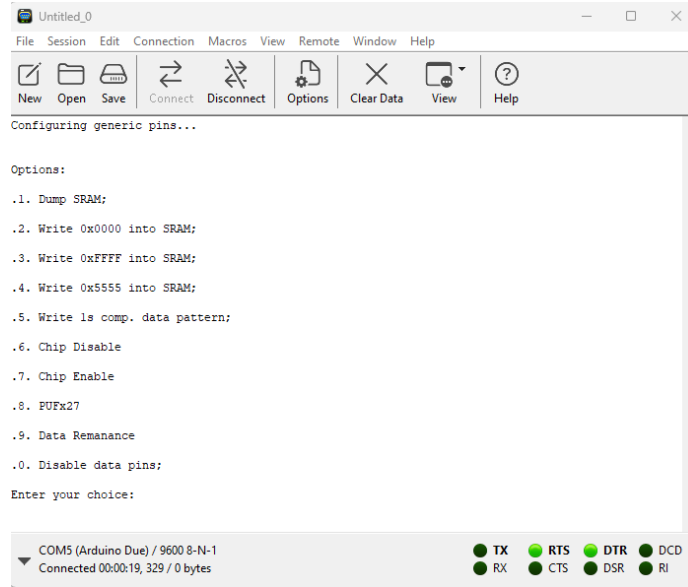


Figure 3.7: Firmware validation menu.

Power-Up State Acquisition

After confirming that the firmware interface was operating correctly, the next step was to validate power-up-state acquisition of the SRAM. In this procedure, the memory contents were read immediately after power-up and sequentially dumped through the serial monitor without first writing a new data pattern to the array. This verification was important because the later experimental analysis in this thesis relies on accurate acquisition of the natural initial state of the SRAM.

During this procedure, the firmware iterated through the SRAM address space and read the contents of each word location using the sequential readout routine. The resulting 16-bit words were transmitted to the host computer in hexadecimal format through the serial interface. Successful completion of this operation confirmed that the firmware could correctly perform full-array

readout of the SRAM in its initial state and stream the collected data to the host system without interruption.

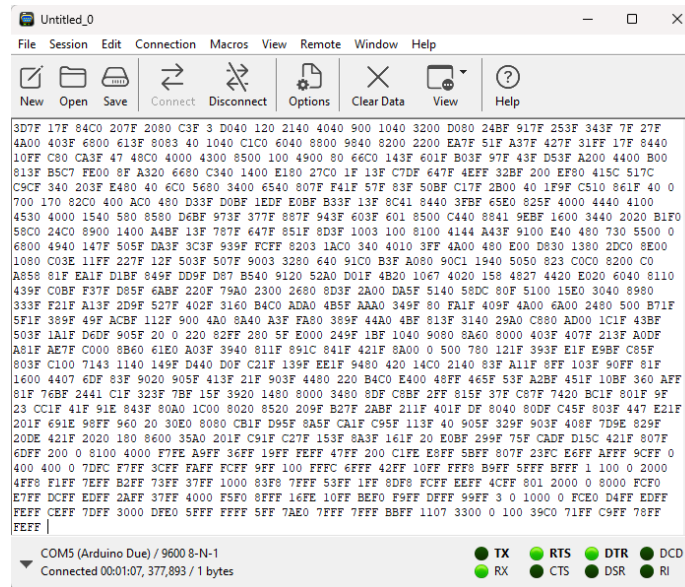


Figure 3.8: Initial SRAM power-up-state readout.

This validation step verified several important functions of the interfacing framework. It confirmed that the address iteration logic operated correctly across the complete memory depth, that the SRAM read routine consistently retrieved data from each address location, and that the serial transmission format was suitable for later storage and post-processing. Figure 3.8 shows a representative portion of the serial output obtained during acquisition of the initial SRAM power-up state.

Full-Memory Write and Read Verification

After validating correct acquisition of the initial SRAM state, the interface was further verified using a full-memory write and readback procedure. In this validation, a known 16-bit data pattern was written across the complete SRAM array and then read back using the sequential readout routine. This step was performed to confirm that the firmware could reliably initialize the entire

memory space and subsequently recover the written contents correctly. In the implementation used in this work, the pattern 0x5555 was written to all word locations in the SRAM.

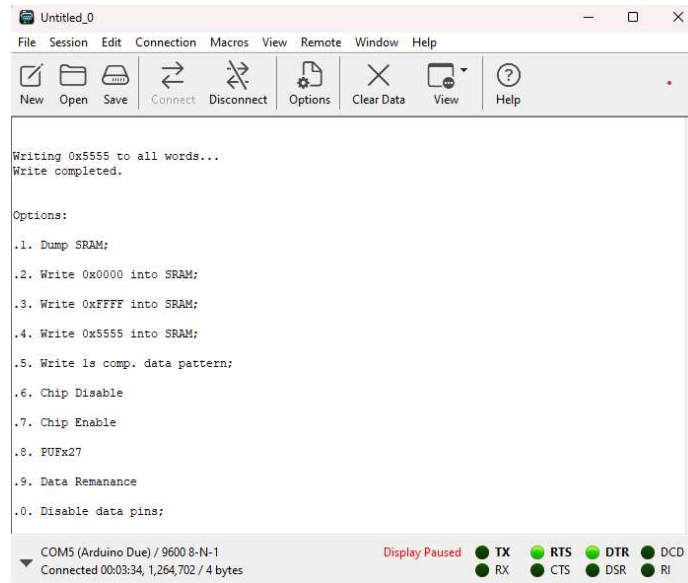


Figure 3.9: Full-memory write verification.

The serial output corresponding to this step is shown in Figure 3.9, where the firmware reports completion of the full-memory write operation. Successful execution of this command confirms that the address bus, data bus, and write control sequence were functioning correctly over the full valid address range of the SRAM. Following the write operation, the SRAM contents were read back sequentially and dumped through the serial monitor. Figure 3.10 shows the corresponding readback output, where the SRAM array returns the expected 0x5555 pattern repeatedly across the dumped memory locations. This result verifies that the interface could correctly write to and read from the full SRAM array without address corruption, bus-direction errors, or incomplete memory access. Together, the write and readback results establish that the developed hardware and firmware provide a reliable SRAM access platform for the experimental procedures presented in this thesis. The successful reproduction of the expected full-memory pattern demonstrates correct operation of the address placement logic, bidirectional data bus control, and the timing sequence required for read and write access.

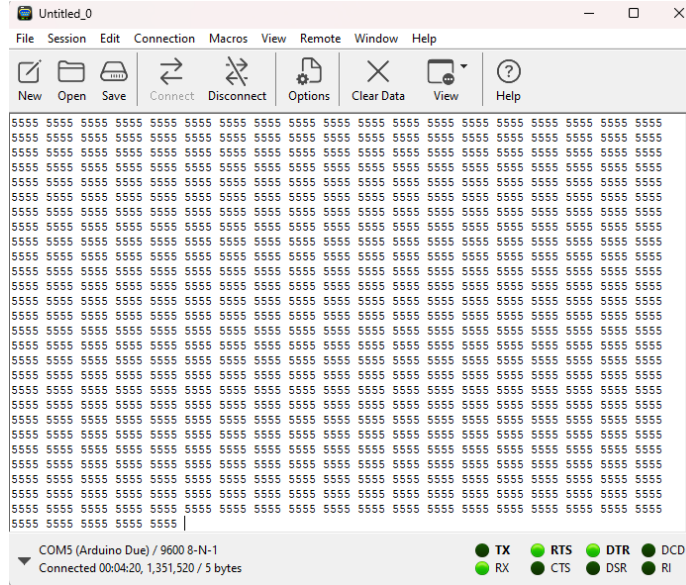


Figure 3.10: Full-memory readback verification.

The validation procedures described above confirmed that the developed interfacing framework could correctly acquire the initial power-up state of the SRAM and perform reliable full-memory write and readback operations. These checks established the correctness of the hardware and firmware platform and provided confidence that the data collected in the later experimental studies accurately reflected SRAM behavior rather than artifacts of the interface.

Chapter 4

Power-Up State Characterization

4.1 SRAM Power-Up Behavior

The PUS of an SRAM array is the binary pattern observed immediately after the supply voltage is applied, before any intentional read or write operation is performed. At power-up, each SRAM bit cell must settle into one of its two stable states. Since an SRAM array contains a large number of cross-coupled bistable cells, the collection of power-up values across all cells forms the PUS. In an ideal and perfectly symmetric cell, there would be no preferred power-up direction. In practice, however, transistor-level mismatch causes one side of the cell to become slightly stronger than the other, and this small imbalance determines whether the cell powers up to logic ‘0’ or logic ‘1’ [?, 9, 15, 38, 39].

A conventional 6T SRAM cell consists of two cross-coupled CMOS inverters and two access transistors, as shown in Figure 4.1. The access transistors connect the cell to the bit lines during read and write operations, while the cross-coupled inverter pair provides the regenerative feedback required for storing one bit of information. When the supply voltage is initially turned on, both internal storage nodes begin from an undefined condition and rise from zero as the power rail ramps upward. During this transient, even a very small difference in threshold voltage, current drive, parasitic capacitance, or mobility can cause one node to charge slightly faster than the other [7, 15, 40].

To illustrate this mechanism, consider the case in which the threshold-voltage magnitude of PMOS transistor P_1 is slightly lower than that of P_2 , that is,

$$|V_{th}^{(P_1)}| < |V_{th}^{(P_2)}|.$$

Because P_1 turns on more strongly during the supply ramp, node Q rises faster than node $\overline{Q}$. Although this initial voltage difference is very small, the positive feedback of the cross-coupled

inverter pair amplifies it rapidly. As Q rises, the complementary side is driven downward more strongly, which further reinforces the increase in Q . This regenerative action continues until one node approaches V_{DD} and the other approaches ground. Thus, the final power-up value is determined by the inherent electrical mismatch between the two sides of the cell [7, 15].

The power-up transient can be viewed in two conceptual phases. In the early phase, both internal nodes begin charging as the supply is applied, and the PMOS mismatch largely determines which node gains voltage first. In the later phase, the corresponding NMOS pull-down network and the inverter feedback amplify this initial imbalance, forcing the cell decisively into one of its two stable states. Figure 4.1 illustrates both the transistor-level power-up mechanism and a representative simulation waveform showing how one node is driven toward V_{DD} while the other collapses toward 0 V [7, 15, 40].

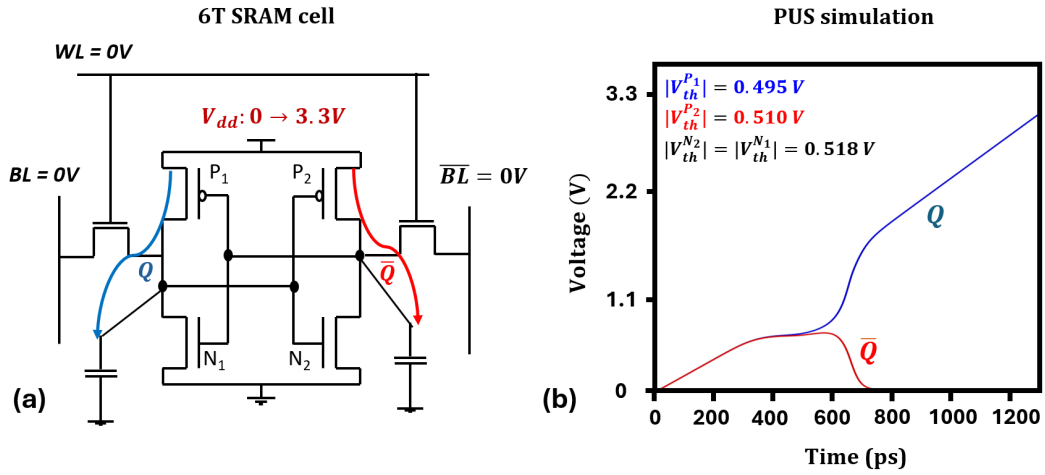


Figure 4.1: SRAM power-up behavior: (a) power-up mechanism of a 6T SRAM cell and (b) representative PUS simulation waveform.

At the array level, this mechanism produces a power-up bit pattern in which some cells preferentially power up to ‘0’ while others preferentially power up to ‘1’. Since the mismatch pattern is determined by fabrication-induced process variation, the resulting PUS is characteristic of a given chip and tends to be repeatable under similar operating conditions. This property is the basis of both SRAM fingerprinting and SRAM-based PUFs. At the same time, because the power-up be-

havior depends on transistor mismatch, it is also sensitive to effects that modify that mismatch over time, such as aging and ionizing radiation. This dependence is central to the imprinting and sanitization phenomena investigated later in this thesis [9, 15, 39].

4.2 PUS Characterization of SRAM Chips

To characterize the intrinsic PUS of the SRAM chips, the power-up response from the same SRAM region is captured repeatedly over multiple power cycles. First, 101 consecutive power-up states are collected from the same memory location, and a stable reference state is then generated by applying majority voting at each bit position. This majority-voted reference is referred to as the GoldPUF. Because a small number of cells may change their state across repeated power cycles due to electrical noise and marginal stability, the GoldPUF provides a robust reference for quantifying the intrinsic variation of the power-up behavior [41].

The deviation of each measured PUS from the corresponding GoldPUF is quantified using the Hamming Distance (HD), defined as

$$HD = \frac{\# \text{ of set bits (GoldPUF} \oplus \text{CurrentPUF)}}{\text{Total number of PUF bits}} \times 100.$$

In this chapter, the resulting value is denoted as $HD^{\text{intrinsic}}$ because it represents the natural power-up-state variation of the SRAM before any aging or irradiation effects are introduced. A smaller value of $HD^{\text{intrinsic}}$ indicates better repeatability of the PUS and therefore stronger intrinsic power-up stability.

After establishing the intrinsic characterization methodology, the PUS can also be observed by reading the binary images of different SRAM chips. Figure 4.2 shows representative fresh PUS patterns from four different COTS SRAM technologies used in this work. These images clearly show that not all SRAM chips exhibit the same PUS. Instead, the power-up pattern is unique to each chip and is determined by the underlying device mismatch, process variation, and technology characteristics [?, 9, 38, 39].

A clear difference can be observed across the four devices shown in Figure 4.2. The CY 90 nm chip exhibits a random-looking power-up pattern, but it is visibly biased toward black pixels, which correspond to logic ‘0’. In contrast, the other three chips show random patterns with a much more balanced distribution of logic ‘0’ and logic ‘1’, indicating an approximately equal proportion of zeros and ones. In addition, the locations of the dark and light pixels are not the same across the four SRAM chips. This shows that even when different devices serve the same memory function, their power-up states are spatially different and do not repeat in the same positions from chip to chip.

These observations confirm that the PUS is not only random-looking, but also chip-specific. The uniqueness of the power-up pattern is an important reason why SRAM can be used for fingerprinting and PUF applications. At the same time, the variation observed across the four chips also provides an important baseline for this thesis, because any change in the PUS due to aging or radiation must be interpreted relative to the natural power-up characteristics of the individual device.

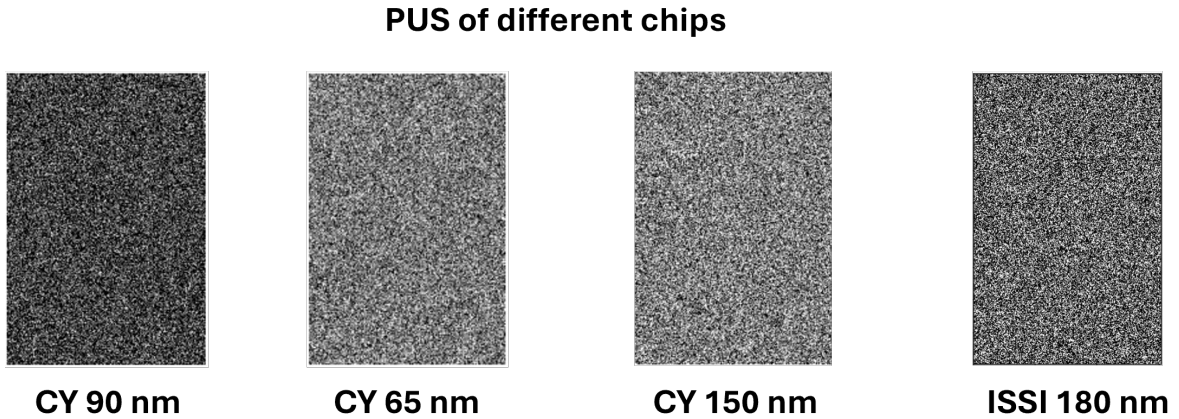


Figure 4.2: Representative fresh PUS patterns from four different COTS SRAM technologies.

Table 5.1 summarizes the $HD^{\text{intrinsic}}$ values for the four SRAM technologies used in this work. All four devices exhibit relatively small intrinsic HD values, indicating that their power-up responses are reasonably stable across repeated power cycles. These results establish the natural

baseline variability of each SRAM chip and provide the reference point against which later aging-induced imprinting and radiation-induced changes are evaluated.

Table 4.1: Intrinsic power-up-state variability of the SRAM chips

Device	Manufacturer	Capacity	Supply Voltage	Technology Node	$HD^{\text{intrinsic}}$
CY 90 nm	Cypress	4 Mbits	3.3 V	90 nm	3.37%
CY 65 nm	Cypress	4 Mbits	3.3 V	65 nm	2.63%
CY 150 nm	Cypress	4 Mbits	3.3 V	150 nm	2.26%
ISSI 180 nm	ISSI	4 Mbits	3.3 V	180 nm	3.38%

4.3 SRAM Power-Up States as Device Fingerprints

The power-up state of each SRAM cell is influenced by fabrication-induced mismatch, the aggregate power-up pattern of an SRAM array can serve as a hardware-specific fingerprint. Even though each individual cell stores only one bit after power-up, the full array contains a large number of such cells, and their combined responses produce a chip-specific binary pattern. This pattern is not intentionally programmed into the memory; instead, it arises naturally from the microscopic differences introduced during semiconductor processing. As a result, two nominally identical SRAM chips fabricated using the same design and process technology will still exhibit different power-up patterns as shown in the previous section. [?, 9, 38, 39].

The usefulness of the SRAM power-up state as a fingerprint comes from a balance of randomness and repeatability. Across different chips, the mismatch distribution is different, which provides uniqueness. Across repeated power cycles of the same chip under similar conditions, many cells tend to power up to the same value, which provides reproducibility. Some cells are strongly biased and almost always initialize to the same logic value, while others are weakly biased and may switch more often. The overall PUS therefore contains both highly stable cells and less stable cells. In practical fingerprint extraction, the most stable cells are especially useful because they provide reliable identification over repeated measurements [?, 42, 43].

In addition to direct bit-pattern comparison, the fingerprint property of SRAM can also be described statistically. If the same SRAM array is powered up repeatedly and each cell is observed over many trials, then the probability that a particular cell powers up to logic ‘1’ can be estimated. Cells that strongly prefer one state will have probabilities close to 0 or 1, whereas weaker cells will have intermediate values. Figure 4.3 illustrates this idea using a full-chip fingerprint map and an extracted 64-bit region. In the enlarged region, the grayscale level of each cell represents its probability of powering up to logic ‘1’. This type of representation is useful because it shows that the SRAM fingerprint is not merely a single binary image, but can also be interpreted as a spatial map of power-up preference across the array [?, 39].

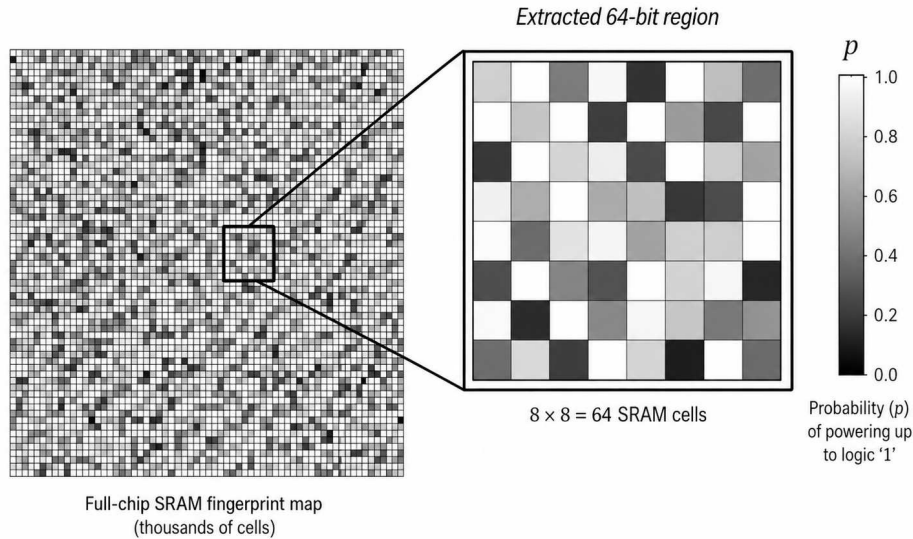


Figure 4.3: SRAM fingerprint map and extracted 64-bit region with power-up probability.

The use of SRAM power-up states as fingerprints is attractive because the identifier is derived directly from the physical properties of the chip and does not need to be permanently stored in non-volatile memory. The fingerprint can be captured immediately after power is applied, before the memory array is overwritten with program data. This property makes SRAM-based fingerprints useful for secure boot, device binding, authentication, and anti-counterfeiting. However, their reliability depends on environmental conditions and long-term device behavior. Since temperature,

voltage, aging, and radiation can modify transistor mismatch and cell stability, the same factors can also influence fingerprint reproducibility. This is especially relevant to this thesis because aging-induced imprinting alters the natural PUS and therefore changes the fingerprint characteristics of an SRAM array over time [9, 42, 43].

4.4 Physical Unclonable Functions (PUFs)

PUFs are hardware security primitives that exploit uncontrollable manufacturing variation to generate device-specific responses. In SRAM-based PUFs, the response is derived from the PUS of the SRAM array. When the device is powered on, the power-up values of selected SRAM cells are read before any data is written. The resulting binary sequence serves as the PUF response. Since this sequence is rooted in transistor-level mismatch created during fabrication, it is extremely difficult to reproduce exactly in another chip, even if the two chips were manufactured using the same process [6–9, 39].

The main advantage of an SRAM-based PUF is that the response is generated only when needed and does not have to be permanently stored as a secret key in non-volatile memory. This reduces the exposure of stored secret material to direct memory extraction attacks. In a typical usage flow, the device is first enrolled by recording a reference power-up response. Later, when authentication is required, the device is powered up again and a new response is measured. If the newly observed response matches the enrolled reference within an acceptable tolerance, the device is considered authentic [6, 7, 38].

SRAM-based PUFs are widely used in hardware security because they provide a lightweight and low-cost method for deriving chip-specific information from memory that already exists on the device. One major application is secure boot, where the power-up response can be used to derive or reconstruct device-specific cryptographic material needed during the boot process. Another important application is device authentication, where a measured SRAM PUF response is compared against a stored reference to verify the identity of the hardware. SRAM-based PUFs are also

useful in anti-counterfeiting systems because they allow genuine components to be distinguished from cloned or substituted parts [6, 8, 42, 44].

For a PUF to be useful, it must satisfy several important properties. First, it must be unique, meaning that different chips should produce sufficiently different responses. Second, it must be reliable, meaning that the same chip should produce the same or nearly the same response across repeated measurements under nominal operating conditions. Third, the response should be difficult to predict or clone without access to the actual physical device. SRAM naturally provides these properties to a useful degree because the power-up response originates from local device mismatch distributed across a large number of cells [6–8, 39, 43].

Despite these advantages, SRAM-based PUFs are not completely immune to variation over time. Environmental changes such as temperature, supply voltage, and ramp conditions can influence power-up repeatability. More importantly for this thesis, long-term aging and radiation exposure can modify the threshold voltages and relative strengths of the transistors in the SRAM cell, thereby changing the preferred power-up direction of some cells. As a result, the original PUF response may drift with time, and in severe cases the altered power-up pattern may even reveal traces of previously stored information rather than only the intrinsic fresh-state fingerprint. This observation directly connects the PUF literature to the security problems addressed in this thesis, because the same physical mechanism that makes SRAM useful for identification can also create a vulnerability when aging or radiation changes the natural power-up characteristics of the memory array [7–9, 40, 43].

4.5 Conclusion

This chapter established the fundamental behavior of SRAM power-up states and explained how transistor-level mismatch determines the initial state of a 6T SRAM cell during power-up. The major conclusions of this chapter are summarized as follows:

- **Physical Origin of SRAM PUS:** This chapter explained that the SRAM power-up state originates from small transistor-level mismatches between the two cross-coupled inverter

branches of the SRAM cell. During power-up, even a slight electrical imbalance between the two sides of the cell is sufficient to force the cell into one of the two stable logic states. Therefore, the final power-up value of each bit is determined by the local device mismatch present in that SRAM cell.

- **Chip-Specific Power-Up Behavior:** The representative PUS images of four different SRAM chips showed that the power-up pattern is not identical across devices. Each chip exhibits its own spatial distribution of logic ‘0’ and logic ‘1’, and the bit locations are different from one chip to another. The CY 90 nm device also shows a visibly stronger bias toward logic ‘0’, whereas the other devices exhibit a more balanced random distribution. These observations confirm that the natural PUS is chip-specific and technology-dependent.
- **Fingerprinting and PUF Applications:** This chapter also established the relationship between SRAM PUS behavior, device fingerprinting, and PUF applications. Since the power-up state is caused by local process-induced mismatch, it can be used as a hardware-specific identifier and as a source of device uniqueness. This makes SRAM PUS useful for fingerprinting and PUF-based security applications.
- **Aging and Radiation Effects on PUS:** The same physical mechanism that makes SRAM PUS unique also makes it sensitive to environmental conditions, aging, and radiation. Changes in transistor parameters caused by long-term stress or irradiation can alter the mismatch balance inside the SRAM cell and modify the future power-up state. This sensitivity provides the foundation for the security concerns and experimental analysis presented in the later chapters of this thesis.
- **Baseline characterization on PUS:** Overall, this chapter provides the baseline understanding of natural SRAM power-up behavior before aging or irradiation effects are introduced. The qualitative PUS images presented in this chapter show the device-dependent nature of SRAM startup behavior, while the exact percentages of logic ‘0’ and logic ‘1’ for each chip are discussed in the next chapter as part of the experimental characterization.

Chapter 5

Aging Induced Imprinting Effects on SRAM Array

5.1 Introduction

Static Random Access Memory (SRAM) is a fundamental building block in modern computing systems, widely used in CPU caches and on embedded platforms that hold sensitive data such as cryptographic keys and passwords. Although SRAM is considered volatile and is expected to lose its contents upon power removal, recent studies reveal data imprinting effects in commercial SRAM after prolonged data retention [10–12]. These studies demonstrate that the SRAM power-up state (PUS) is influenced by aging, potentially enabling adversaries to recover sensitive information stored in the SRAM through careful analysis of its power-up state [10, 12].

The data imprinting effects in SRAM raise substantial concerns regarding the lingering of sensitive information in memory beyond its intended use. For instance, when electronic systems are discarded, there is a noteworthy chance that the SRAM memory remains operational, creating a risk of leaking previously stored sensitive information. This persistent data remanence poses a serious threat, as it can lead to unauthorized access and compromise. These risks highlight the urgent need to fundamentally understand aging-induced imprinting effects in commercial SRAM across different technology nodes and manufacturers. They also emphasize the importance of developing robust sanitization techniques to ensure secure disposal of electronic systems and to mitigate data leakage from residual SRAM contents [10–12, 45].

In this chapter, we conduct a comprehensive experimental evaluation of aging-induced data imprinting effects in COTS SRAM arrays. We evaluate multiple SRAM chips across four different technology nodes from two major vendors under accelerated thermal stress to characterize the imprinting effects. We systematically assess the impact of stress from temperature, duration, and supply voltage to quantify the data-imprinting vulnerability in aged SRAM chips. Finally, we

evaluate room-temperature relaxation to determine how long the induced bias persists in the PUS after aging.

5.2 Background

Recent studies show that prolonged data retention and aging cause data imprinting effects on the SRAM power-up state [10, 11, 32, 46, 47]. For example, J. Hovanes et al. demonstrated reverse-imprinted images on SRAM-PUS using elevated-temperature stress for approximately 12 hours and successfully recovered partial data [10]. Their data recovery approach assumes access to an initial PUS as a reference for comparison. Later, Tisha et al. improved the data recovery method from aged SRAM arrays without relying on the initial PUS measurement [33–35]. Similarly, Mahmood et al. explored how temperature variation influences the stability and repeatability of power-up behavior [32], and an active attack model to expose on-chip data stored in SRAM was also proposed [31]. Likewise, A. Garg et al. reported accelerated NBTI by increasing the supply voltage to bias the PUS [30]. Additionally, Surendranathan et al. demonstrated that data imprinting on SRAM PUS can be caused by ionizing radiation through threshold-voltage shifts and charge-trapping effects [11, 36]. From a device-physics perspective, NBTI-driven threshold-voltage shifts in PMOS transistors are widely reported [25, 48, 49] and can manifest as a measurable imbalance in the power-up distribution [?, 7]. Such aging signatures have additionally been leveraged to identify recycled or aged chips [13, 14].

While prior work has established that SRAM imprinting can arise from aging and that partial data recovery is feasible, a systematic characterization of imprint magnitude as a function of aging temperature, duration, supply voltage, relaxation behavior, and CMOS technology node remains lacking. Consequently, it is still unclear under which operating conditions or technology generations the data-leakage risk becomes severe. Our study addresses this gap by quantitatively analyzing imprint formation and recovery under controlled and repeatable conditions across multiple COTS SRAM devices spanning different technology nodes.

5.3 Threat Model

Figure 5.1 illustrates the threat model conceptually. We consider an adversary with physical access to a discarded SRAM device who aims to recover previously stored sensitive data. The adversary may obtain devices containing on-chip or standalone COTS SRAM through resale, recycling, disposal channels, or temporary access prior to end-of-life decommissioning. With direct chip access, the adversary can perform low-level operations, including reading the SRAM power-up state (PUS). By repeatedly measuring the PUS across multiple power cycles and conducting offline statistical analysis, the adversary may infer remnants of previously stored confidential information.

Security-critical content such as firmware, program data, and cryptographic keys often resides in fixed SRAM addresses during normal operation, causing the same cells to store the same values for long periods and accumulate value-dependent aging. In practice, SRAM is frequently treated as safe at retirement because it is volatile, and sanitization is therefore often limited to simply powering the system off. If aging-induced imprinting biases the PUS toward previously stored values, repeated PUS measurements can enable post-retirement inference of that content.

To evaluate the risk in this threat model, we summarize our results using three simple indicators. First, we measure the extent to which the SRAM PUS biases toward the stored data during aging. Second, we measure how quickly this change appears as stress increases. Third, we measure how long this change persists after power-off during room-temperature relaxation.

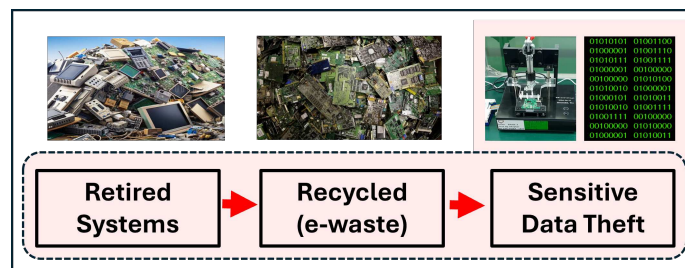


Figure 5.1: SRAM end-of-life threat model.

5.4 Experimental Flow

The aging study was carried out using multiple commercial off-the-shelf (COTS) SRAM chips from Cypress and ISSI spanning four technology nodes: CY 90 nm, CY 65 nm, CY 150 nm, and ISSI 180 nm. All devices used in this study have a capacity of 4 Mbits and operate at a nominal supply voltage of 3.3 V. The overall measurement setup and the corresponding experimental flow used for the aging study are shown in Figure 5.2. The setup consists of a workstation for data storage and processing, the custom-designed SRAM interface platform described in Chapter 3, and a controlled-temperature chamber used to apply thermal stress.

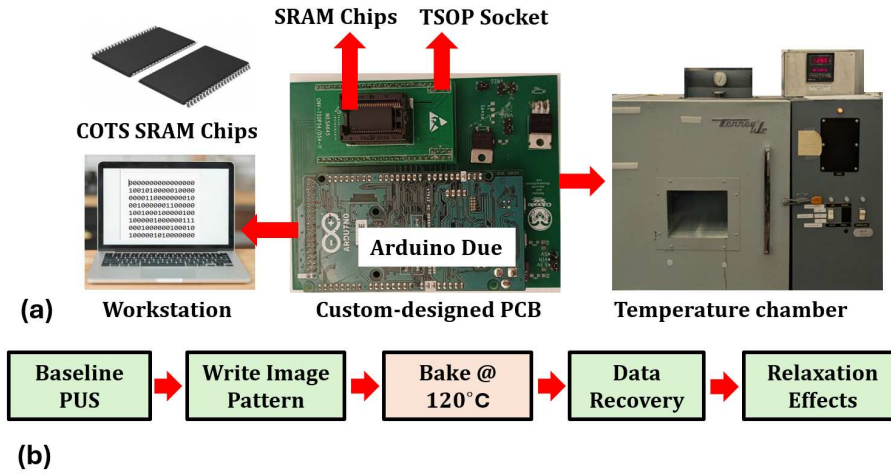


Figure 5.2: (a) Aging setup and (b) experimental flow.

Table 5.1: Summary of SRAM devices used in the aging study

Device	Manufacturer	Capacity	Supply Voltage	Technology Node	HD^{fresh}
CY 90 nm	Cypress	4 Mbits	3.3 V	90 nm	33.00%
CY 65 nm	Cypress	4 Mbits	3.3 V	65 nm	51.64%
CY 150 nm	Cypress	4 Mbits	3.3 V	150 nm	50.06%
ISSI 180 nm	ISSI	4 Mbits	3.3 V	180 nm	49.83%

The experimental flow consists of four main stages: baseline characterization, aging characterization, imprint quantification, and relaxation characterization. These stages are described below.

1. **Baseline Characterization:** Before applying any data pattern or stress, each SRAM device was first characterized in its fresh condition to determine its baseline PUS. This baseline measurement is important because the PUS of a commercial SRAM is not perfectly random and may exhibit an intrinsic bias toward logic ‘0’ or logic ‘1’ depending on the underlying device mismatch and technology characteristics. As a result, different SRAM chips can show different initial similarity levels when compared with a chosen reference pattern.

For each device, the complete SRAM array was read out immediately after power-up and transferred to the workstation for storage and later analysis. This fresh-state acquisition provided the pre-aged reference state used throughout the study. The similarity between the fresh PUS and the written image is denoted as HD^{fresh} . Ideally, if both the PUS and the image contain balanced proportions of zeros and ones, then HD^{fresh} is expected to be close to 50%. However, as shown in Table 5.1, some devices exhibit noticeable deviation from this ideal case, indicating an inherent bias in the fresh PUS.

2. **Aging Characterization:** A known image pattern was written into the SRAM array prior to thermal aging. In the main set of experiments, the Einstein image was used as the programmed data pattern. The image was represented as a binary image of size 450×305 pixels, corresponding to a linear array of 137,250 bits. Each binary bit was stored in one SRAM cell.

To improve visibility and provide repeated observation regions within the same chip, multiple copies of the image were written to different address locations of the SRAM array. In addition to Einstein, other reference image families such as Mona Lisa, FullBlack, and Full-White were also prepared and programmed into different memory regions for later analysis and comparison. The image patterns used for SRAM programming prior to thermal aging are shown in Figure 5.3.

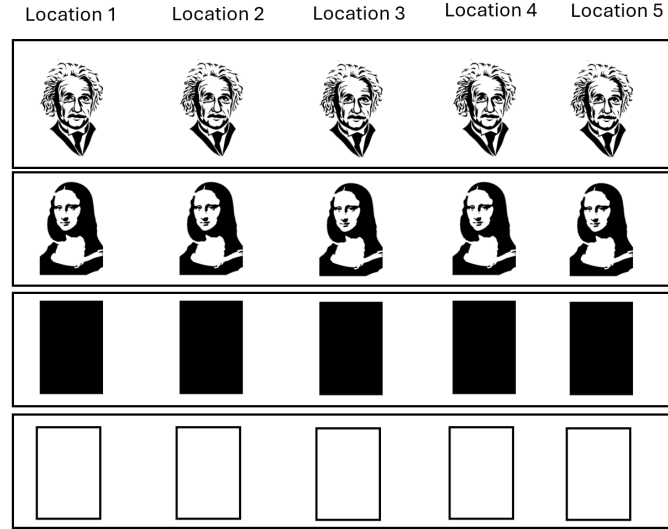


Figure 5.3: Written image patterns.

The purpose of writing a known image prior to aging is to establish a controlled stored state that can later influence the post-aged PUS of the SRAM cells. Since the objective of the study is to determine whether prolonged thermal stress can imprint previously stored information into the future power-up state of the device, it is necessary to begin with a clearly defined and visually recognizable data pattern.

Thermal stress was then applied by placing the SRAM device inside a controlled-temperature chamber. The aging experiments were performed at multiple temperatures ranging from 50°C to 120°C in order to evaluate the temperature dependence of the imprinting phenomenon. During the aging phase, the SRAM chip was maintained in the powered-on condition so that the stored image pattern remained electrically present in the memory cells while the device was exposed to elevated temperature.

The use of powered-on thermal stress is important because it preserves the stored logic states during the aging interval and allows long-term bias-dependent degradation mechanisms to act on the SRAM cell transistors. The bake duration was divided into predefined intervals, and after each interval the device was evaluated again. This interval-based approach made it

possible to track the development of imprinting as a function of both stress time and temperature, rather than considering only the final aged condition.

Data retention was checked immediately after each bake interval by reading back the contents stored in the SRAM before performing a power cycle. This step was used to confirm that the originally written image data was still retained during the aging phase. The retained data was dumped and transferred to the workstation for logging and comparison.

This intermediate verification is important because it separates retention of the programmed data from the later power-up-state imprinting effect. If the written image were not retained during the bake interval, then any subsequent power-up-state behavior could not be confidently attributed to the intended stored pattern. By verifying that the programmed image remained present after thermal stress, this step confirms that the memory cells continued to hold the target data during the aging interval.

Post-aging characterization was then performed by power cycling the SRAM device and capturing the post-aged PUS. Following the power cycle, the complete SRAM array was read out again using the same acquisition framework used for the fresh baseline characterization. This post-aged PUS represents the power-up state of the SRAM after the device has undergone both image programming and thermal stress.

If the aging process has introduced a significant imprinting effect, the measured PUS no longer resembles the original fresh PUS but instead shows increased similarity or dissimilarity with respect to the written image. Since the image pattern is spatially structured, the imprinting effect can often be observed visually after reconstruction of the corresponding binary data. This post-aging acquisition step was repeated after each bake interval so that the evolution of the PUS modification could be tracked progressively.

3. **Imprint Metric:** To quantify the degree of imprinting introduced by aging, the Hamming distance between the post-aged PUS and the written image was computed. This metric is denoted as HD^{age} and is defined as

$$HD^{age}(\%) = \frac{\# \text{ of set bits (PUS} \oplus \text{Image)}}{\text{Total number of bits}} \times 100. \quad (5.1)$$

In this expression, the exclusive-OR operation compares the post-aged PUS with the written image bit by bit. A value of $HD^{age} = 0\%$ indicates perfect forward imprinting, meaning that the post-aged PUS exactly matches the written image. A value of $HD^{age} = 100\%$ corresponds to perfect reverse imprinting, meaning that the post-aged PUS matches the complement of the written image. Intermediate values indicate partial imprinting behavior.

The fresh reference case is denoted by HD^{fresh} , which represents the Hamming distance between the pre-aged baseline PUS and the written image. Ideally, $HD^{fresh} \approx 50\%$, assuming that both the PUS and the written image contain equal proportions of zeros and ones. However, some commercial SRAM chips exhibit intrinsic bias in their PUS toward ‘0’ or ‘1’, and in such cases HD^{fresh} can deviate noticeably from 50%, as shown in Table 5.1. When $HD^{age} \approx HD^{fresh}$, the imprinting effect is considered negligible.

4. **Relaxation Characterization:** After the aging and post-aging PUS acquisition steps were completed, the relaxation behavior of the imprinted SRAM devices was also monitored. In this phase, the aged devices were allowed to remain at room temperature, and their PUS were periodically acquired again over extended time intervals. The purpose of this step was to determine whether the aging-induced imprinting effect remained stable or gradually weakened after removal of the thermal stress.

The relaxation study is important because it distinguishes between short-lived stress effects and long-term imprinting behavior. If the imprinted PUS remains visible even after extended room-temperature storage, then the aging effect can be considered persistent and relevant to long-term data remanence concerns. On the other hand, if the imprint fades rapidly during relaxation, the effect is more transient in nature. By including this final stage in the experimental flow, the study captures not only the formation of imprinting under stress but also its persistence after stress removal.

5.5 Results and Discussion

5.5.1 Demonstration of Imprinting Effects

We demonstrate the imprinting effects using a CY 90 nm COTS SRAM chip. Figure 5.7 visualizes how imprint strength increases with stress duration under a fixed bake temperature of 120°C. From left to right, the figure shows the fresh PUS (random pepper-salt pattern), the programmed Einstein pattern (image stored during stress), and the post-baked PUS images after progressive bake intervals. We find that the post-bake PUS becomes increasingly biased toward the complement of the written data pattern as aging duration increases. [50]

It is interesting to note that the Einstein features become partially visible even after 1 minute of baking, indicating that significant imprinting can take place even with minimal aging duration in this chip. The imprinting effect is quantified using HD^{age} , which increases from 33% in the fresh state to 55% after 1 minute of baking, and further to 80% after 12 hours, demonstrating a monotonic increase in reverse imprinting.

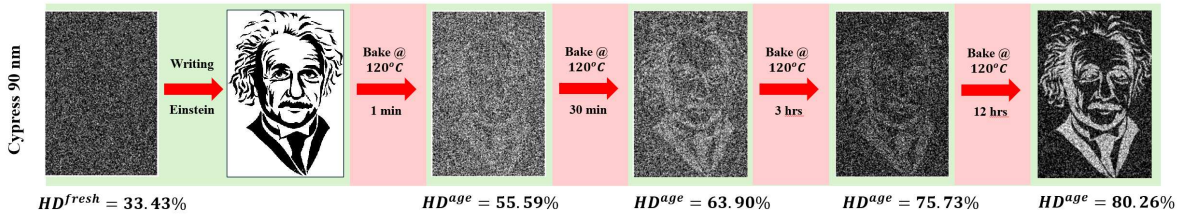


Figure 5.4: Aging-induced imprinting results.

Note that 12 hours at 120°C corresponds to approximately one week at room temperature using the Arrhenius acceleration model with activation energy $E_A = 0.2$ eV, as discussed in the next section. The experiments were terminated at 12 hours, as the imprinting rate exhibits a logarithmic slowdown with extended aging duration.

5.5.2 Temperature-Accelerated Aging Effects

We evaluate temperature-accelerated aging effects using CY 90 nm SRAM chips. Figure 5.7(b) plots the HD percentage of the post-bake PUS as a function of baking duration for four different bake temperatures (50°C, 70°C, 90°C, and 120°C). For each temperature, we use a different CY 90 nm chip of the same part number. We observe a near-logarithmic dependence of the increase in the HD percentage with aging duration at each temperature [10, 12, 50].

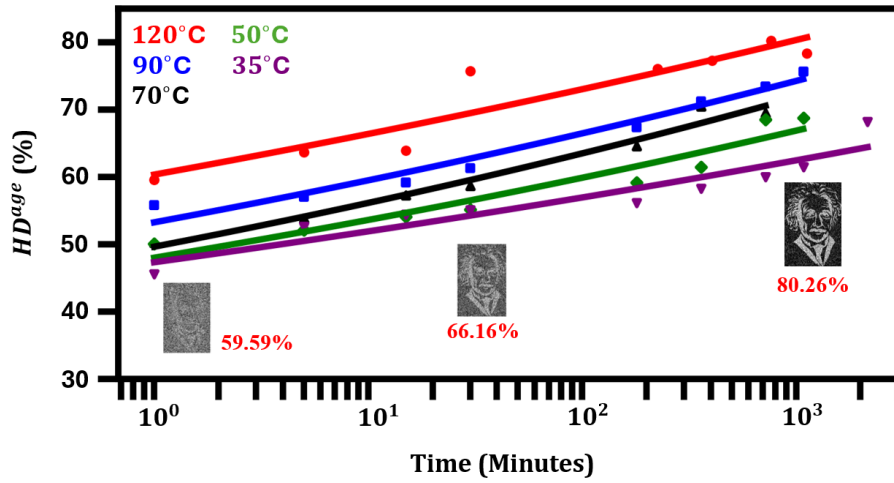


Figure 5.5: Temperature-Accelerated Aging Effects.

At 120°C, the Einstein image becomes discernible within the first minute of stress, and nearly 80% of the image is reverse imprinted after 12 hours, resulting in an almost complete reconstruction of the original content. In contrast, at 50°C, the aging-induced imprint effect is significantly weaker over the same stress duration, indicating substantially slower degradation kinetics. The use of image data in this experiment is intended solely for visual demonstration. Comparable reverse-imprint percentages are observed for other data types, such as text. However, full reconstruction of the stored information from aging-induced imprints may require advanced data recovery techniques [33, 50].

Overall, the HD^{age} increases monotonically with aging time at all temperatures. Higher aging temperatures accelerate the imprinting process and produce stronger imprints, highlighting the

pronounced temperature dependence of the underlying aging mechanism. Figure 5.7(c) shows the temperature dependence of the imprinting effect for a fixed aging duration. An Arrhenius representation is employed to extract the activation energy (E_a) associated with thermally accelerated aging. The solid line corresponds to the linear fit of the Arrhenius model, from which an E_a of approximately 0.2 eV is extracted. Such a relatively low activation energy is characteristic of NBTI-dominated degradation mechanisms, which predominantly affect PMOS transistors [51–53].

Specifically, the threshold voltage magnitude of the PMOS transistor that remains biased in the ON state during aging (e.g., P1 in Fig. 1(a) when the cell stores logic ‘1’) increases due to NBTI stress, whereas the complementary PMOS (P2) experiences comparatively less degradation. As a result, during subsequent power-up, the mismatch $|V_{th}^{(P1)}| > |V_{th}^{(P2)}|$ biases the SRAM cell toward the opposite logic state (logic ‘0’), leading to a reverse imprint of the originally stored data [51–53].

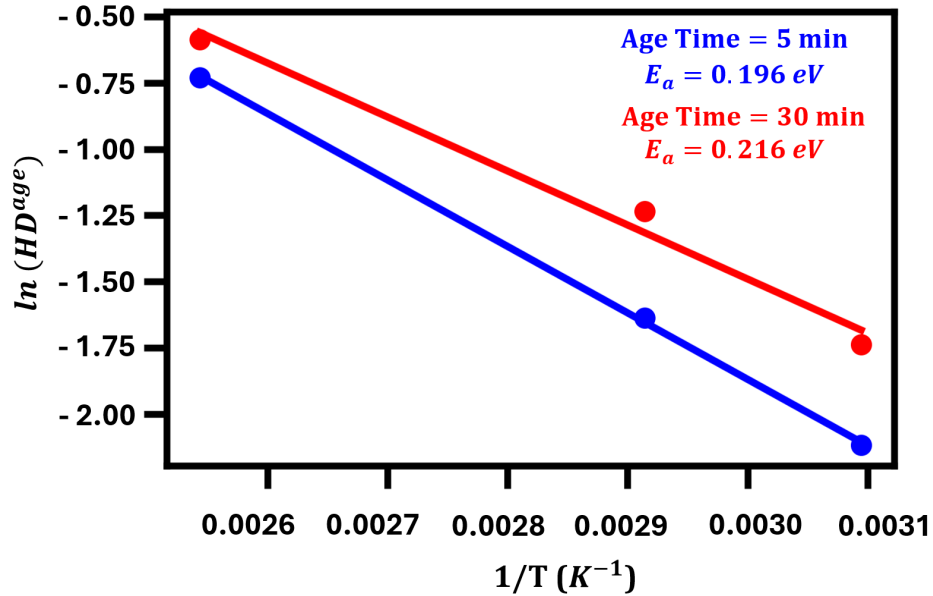


Figure 5.6: Activation Energy of Aging Effects.

5.5.3 Voltage Acceleration Effects

To investigate voltage-acceleration effects on SRAM aging, two identical chips were programmed with the same Einstein image and subjected to bias stress at different supply voltages,

one at 3.3 V and the other at 1.8 V, while maintaining identical thermal stress conditions at 120°C. Figure 5.7(d) illustrates the impact of supply voltage on data imprinting. Under higher-voltage stress (3.3 V), the reverse imprint develops rapidly and reaches a significantly higher magnitude. In contrast, under reduced-voltage stress (1.8 V), the imprint evolves more slowly and remains substantially weaker over the same aging duration.

These results clearly demonstrate that both the rate and the strength of the imprinting effect are strongly dependent on the applied supply voltage during aging, consistent with voltage-accelerated NBTI degradation mechanisms.

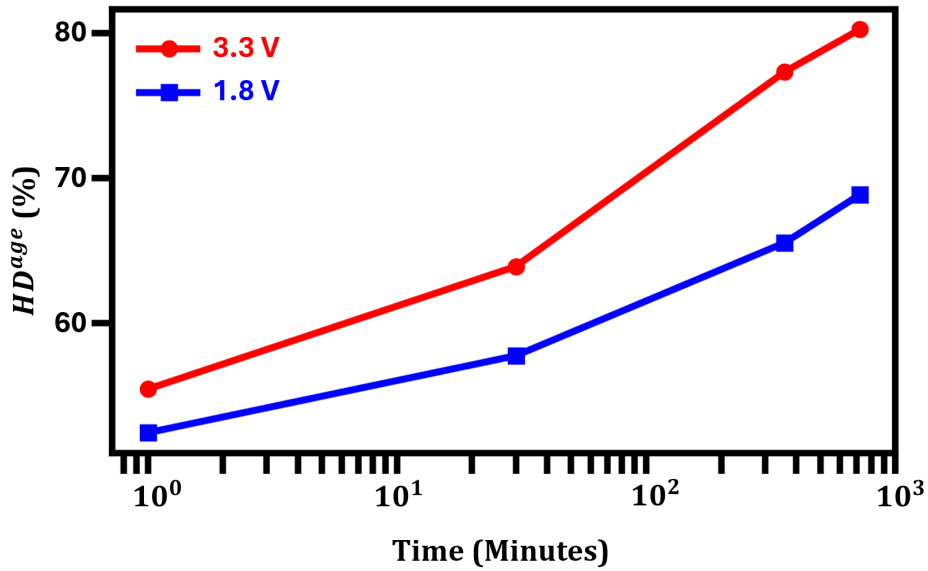


Figure 5.7: Voltage Acceleration Aging Effects.

5.5.4 Relaxation Effects on Imprinted Chips

To evaluate the robustness of the aging-induced data imprinting effect, all aged chips were allowed to relax at room temperature with all pins grounded. The evolution of the post-imprint PUS was monitored over several months by periodically reading the SRAM contents at multiple time intervals. Figure 5.8 presents the relaxation behavior of the Cypress 90 nm SRAM chip.

Within the first hour of relaxation, a noticeable reduction in imprint percentage is observed, indicating rapid partial recovery of the aging-induced degradation. This fast recovery is a well-known

characteristic of NBTI mechanisms [51, 52]. However, despite the initial recovery, a measurable imprint signature persists even after several months. Notably, the pristine (fresh) PUS distribution is not fully restored even after six months of relaxation.

These results suggest that aging-induced imprinting is a long-lasting effect, raising significant concerns regarding data remanence. Previously stored information may persist in SRAM devices long after they are powered down and discarded, posing potential security risks.

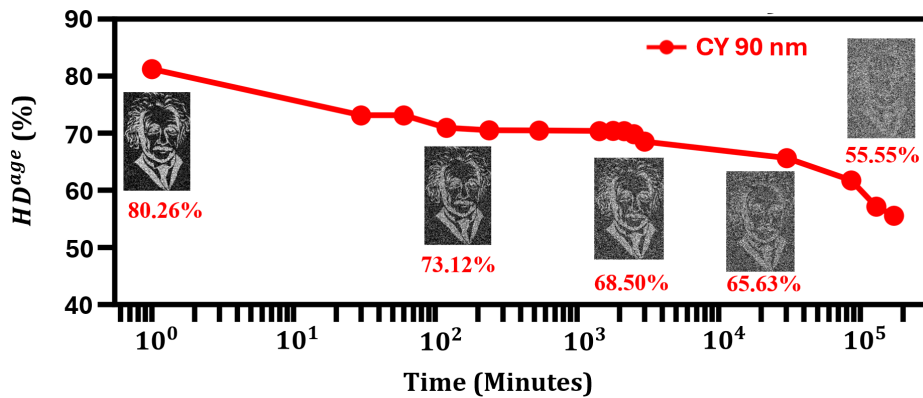


Figure 5.8: Relaxation effects on the imprinted CY 90 nm SRAM chip.

5.5.5 Multi-Chip Evaluation

Although most of the results presented thus far are obtained from the CY 90 nm SRAM chip, we have conducted similar evaluations across multiple chips spanning different technology nodes and vendors. Figure 5.9(a) summarizes the measured aging-induced data imprint for four SRAM types corresponding to four distinct technology nodes. Among the evaluated devices, the CY 90 nm chip exhibits the fastest imprint development under identical stress conditions. In contrast, chips from other technology nodes show measurable imprinting only after significantly longer aging durations at the same temperature. For example, the ISSI 180 nm chip shows lower susceptibility to aging effects, as its PUS remains close to the fresh condition for much of the stress interval and increases only moderately after extended stress of 28 hours, indicating higher resistance under identical conditions. Most notably, the CY 150 nm chip remains essentially flat and does not show

a meaningful imprint even after more than 100 hours of stress, making it the least susceptible and the most resistant to aging-induced imprinting in this study. This observation is generally consistent with the expectation that NBTI effects are less pronounced in older technology nodes, leading to weaker imprint formation [25, 48, 49].

However, the trend is not strictly monotonic with scaling. For example, the CY 65 nm chip does not demonstrate a higher imprinting rate than the 90 nm device, despite being a more scaled technology. One plausible explanation is that the CY 65 nm device may operate with a lower effective SRAM cell supply voltage than the 90 nm part, as many 65 nm SRAM designs use dual-rail supplies with on-chip regulation where the array voltage is typically lower than the externally applied supply to enable low-power operation and voltage scaling, resulting in relatively lower NBTI-induced degradation and a lower imprinting rate [54]. Since all evaluated parts are COTS products, detailed process information, such as gate oxide thickness, oxide materials, channel engineering, and reliability optimizations, remains proprietary. These undisclosed design and process variations likely influence the observed imprint behavior and prevent a definitive explanation of node-to-node dependence. Nevertheless, the key takeaway from this cross-technology study is that aging-induced data imprinting is not confined to a single node or vendor. Instead, it is a broadly observable phenomenon across multiple SRAM technologies, underscoring its generality and potential security implications.

Figure 5.9(b) presents representative imprinted images from multiple technology nodes. Specifically, we show the fresh PUS, the written image, the thermally aged condition, the post-aging PUS, and the room-temperature relaxation results for three SRAM technologies: CY 65 nm, CY 150 nm, and ISSI 180 nm.

The fresh PUS for all these chips exhibits a characteristic salt-and-pepper pattern, reflecting the intrinsic randomness of an unaged SRAM array. During the experiments, different images (e.g., Einstein and Mona Lisa) were programmed into the arrays, and all chips were subjected to thermal stress at 120°C. The post-aging PUS images in Figure 5.9(b) clearly demonstrate reverse imprinting on CY 65 nm and ISSI 180 nm devices. However, the CY 150 nm chip remains similar

to the fresh PUS and does not show a visual imprint even after more than 100 hours of stress. During subsequent room-temperature relaxation, the PUS patterns gradually degrade due to partial recovery; however, visible traces of the aged image pattern persist even after four months across all the imprinted technologies. These observations further confirm the long-lasting and cross-technology nature of aging-induced data imprinting.

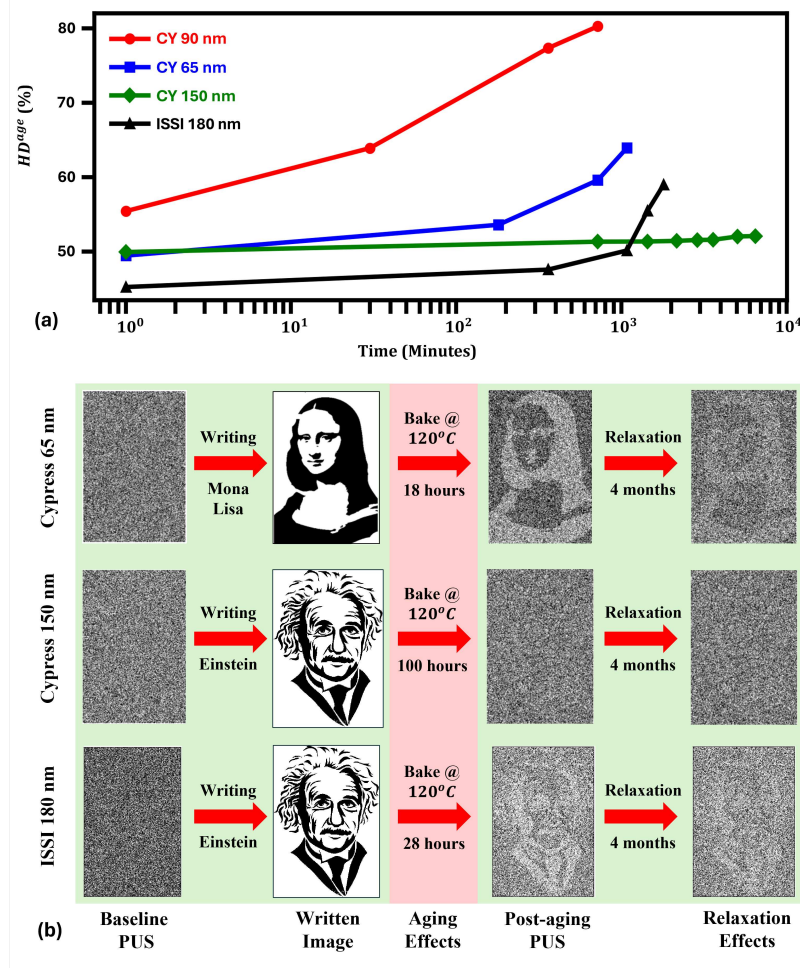


Figure 5.9: (a) Multi-chip evaluation. (b) End-to-end visualization.

5.5.6 Root Cause Analysis for Imprinting Effects

Multiple aging mechanisms, including NBTI, HCI, PBTI, and oxide charge trapping, can induce threshold-voltage shifts in the NMOS and PMOS transistors of an SRAM cell, thereby alter-

ing its PUS. However, the observed experimental trends consistently identify NBTI as the dominant contributor to the reverse imprinting effect [25, 48, 49, 51–53].

First, Arrhenius analysis yields an activation energy of approximately 0.2 eV. Such a low activation energy is characteristic of NBTI degradation, whereas HCI and oxide charge-trapping mechanisms typically exhibit higher activation energies of approximately 1 eV. Second, relaxation measurements show a rapid reduction in imprint magnitude within the first hour at room temperature, reflecting the well-known fast-recovery behavior associated with NBTI. Third, increased supply-voltage stress leads to faster and stronger imprint formation, demonstrating a pronounced bias-acceleration dependence in the imprint kinetics, again consistent with NBTI-driven degradation. Collectively, these observations strongly support NBTI-induced PMOS threshold-voltage shifts as the primary physical mechanism underlying the aging-induced reverse imprinting phenomenon [25, 48, 49, 51–53].

Figure 5.10(a) explains the aging effects on the SRAM cell, where mainly the PMOS transistors' threshold voltage is modified due to NBTI effects. Let us consider a 6T SRAM cell storing $Q/\bar{Q} = 1/0$, as shown in Figure 5.10(a). In this state, the pull-up transistor P1 is ON, while P2 is OFF. Since the ON PMOS (P1) is more strongly affected by NBTI, P1 experiences a significant threshold-voltage increase, causing a threshold-voltage mismatch as follows:

$$|V_{th}^{(P1)}| \gg |V_{th}^{(P2)}|,$$

irrespective of their pre-aged values. This significant mismatch forces the cell to power up in the opposite state ($Q = 0$), as shown in Figure 5.10(b), thereby producing a strong reverse imprint of the written image [51–53].

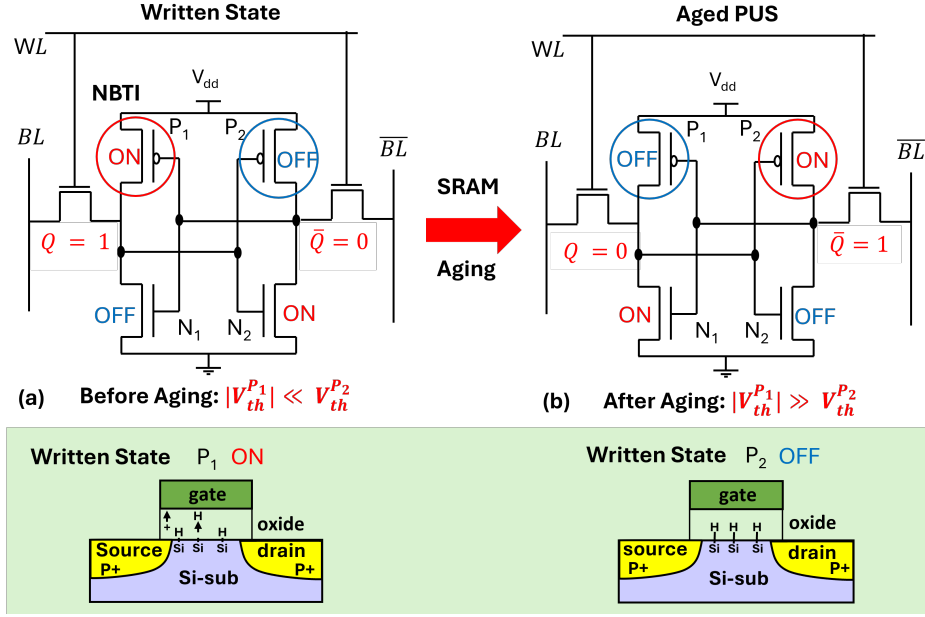


Figure 5.10: Root-cause analysis of imprinting (a) Before aging (b) After aging.

5.5.7 Potential Sanitization Strategies

In this section, we discuss potential sanitization strategies for mitigating the effects of data imprinting in COTS SRAM memories. Unlike destructive disposal methods such as shredding, incineration or pulverization, these strategies aim to suppress residual data recovery while preserving device functionality, offering a cost-effective and reusable alternative. One practical method is to reduce the effect of data imprinting during normal operation. Since aging-induced imprinting becomes more pronounced when a cell remains in the same logic state under power for an extended duration, periodic data remapping or reshuffling during normal operation can reduce prolonged bias on individual cells and thereby limit its impact on the subsequent PUS. Additionally, end-of-life sanitization may be achieved by overwriting the array with different data patterns, such as random data, followed by thermal annealing to reduce the aging-induced bias on the PUS. Another possible strategy is the use of X-ray irradiation on aged chips to mitigate residual imprint effects. Prior studies have shown that irradiation induces charge trapping in oxide layers [11], which may help counteract aging-induced imprinting. Although irradiation and thermal annealing may sanitize the chips from aging biases, these techniques can significantly degrade device characteristics,

such as standby power and static noise margin. Therefore, further investigation is required before such approaches can be considered viable for chip reuse.

5.6 Conclusion

- **Aging-Induced Data Imprinting in COTS SRAM:** This chapter presented a comprehensive experimental evaluation of aging-induced data imprinting effects in COTS SRAM arrays across multiple vendors and technology nodes. The results showed that the SRAM PUS can be measurably altered by prolonged thermal and electrical stress, causing previously stored information to influence the future power-up behavior of the memory. This behavior establishes aging-induced imprinting as a realistic end-of-life security concern for SRAM-based systems, particularly when sensitive data remains mapped to fixed memory locations for extended periods.
- **Aging Dependence on Stress Conditions and Technology Node:** The experiments further demonstrated that the imprinting effect is strongly dependent on stress conditions. Higher temperature, longer stress duration, and increased supply voltage all accelerate imprint formation and increase its magnitude. In the most vulnerable device studied, the CY 90 nm SRAM exhibited pronounced reverse imprinting, with the post-aged PUS becoming increasingly biased toward the complement of the written image. In contrast, other technologies such as ISSI 180 nm and CY 150 nm showed much weaker susceptibility under the same general stress conditions, indicating that aging-induced imprinting is technology-dependent and influenced by both scaling and proprietary process details.
- **Aging Persistence After Relaxation:** The relaxation measurements showed that the imprinting effect is not purely transient. Although partial recovery occurs after stress removal, visible and measurable traces of the imprinted pattern remain for extended durations at room temperature. This persistence confirms that aging-induced imprinting can create long-lasting

data remanence in SRAM arrays, thereby increasing the risk of information leakage from discarded, reused, or recycled hardware.

- **Physical Mechanism of Reverse Imprinting:** The device-level analysis presented in this chapter identified NBTI-induced PMOS threshold-voltage mismatch as the dominant physical mechanism underlying the observed reverse imprinting phenomenon. The Arrhenius activation behavior, voltage-acceleration effects, and partial room-temperature recovery collectively support this interpretation. Specifically, the PMOS transistor that remains biased in the ON state during data retention undergoes stronger degradation than its complementary device, producing a threshold-voltage imbalance that shifts the future power-up preference of the SRAM cell toward the opposite logic state.
- **Security Implications and Mitigation:** Overall, the results of this chapter establish that aging-induced imprinting in SRAM is measurable, persistent, and broadly observable across multiple COTS technologies, although with strongly different susceptibility levels. These findings emphasize the need for robust end-of-life protection strategies for SRAM-based systems. Possible mitigation approaches include reducing prolonged static data retention during normal operation through periodic data reshuffling, as well as applying dedicated post-use sanitization techniques such as controlled thermal annealing or radiation-based treatment before device disposal or controlled reuse.

Chapter 6

Radiation-Based Sanitization of Aged SRAM

6.1 Introduction

SRAM is widely used in modern computing systems because of its speed, low latency, and compatibility with digital logic. It is a core component of processor caches, microcontrollers, embedded systems, and SoC platforms, where it frequently stores security-critical information such as firmware states, temporary program data, authentication values, passwords, and cryptographic keys. SRAM is traditionally treated as a volatile memory, and for this reason it is often assumed that simply removing power is sufficient to eliminate the stored information. However, the results presented in the previous chapter show that this assumption does not always hold in aged devices. Under prolonged bias and thermal stress, SRAM cells can develop aging-induced imprinting effects that alter the future PUS of the array and create a bias toward previously stored data. This behavior introduces a serious end-of-life security concern because information that was expected to disappear after power removal may continue to influence the power-up behavior of the memory long after normal operation has ended.

This issue becomes especially important when semiconductor memories reach the end of their intended lifetime. In practice, integrated circuits and electronic systems are rarely handled through a single uniform disposal path. Some devices are recycled and re-enter material recovery or refurbishment chains. Others are logically deleted and reused in development boards, prototype systems, secondary markets, or repair workflows. Some systems are physically shredded, while others are dumped into e-waste streams or subjected to burning and incineration. Although these approaches differ in cost, scale, and environmental impact, they do not all provide the same level of data security. In particular, recycling and reuse represent a major risk because the memory device may remain functional and accessible even after the original system has been retired. In such cases, an adversary with direct physical access to the chip may still be able to observe the SRAM

power-up behavior and infer traces of previously stored information. Figure 6.1(a) highlights non-destructive end-of-life paths such as recycling and dumping e-waste, while Figure 6.1(b) illustrates common discarding approaches such as delete, burning, and shredding.

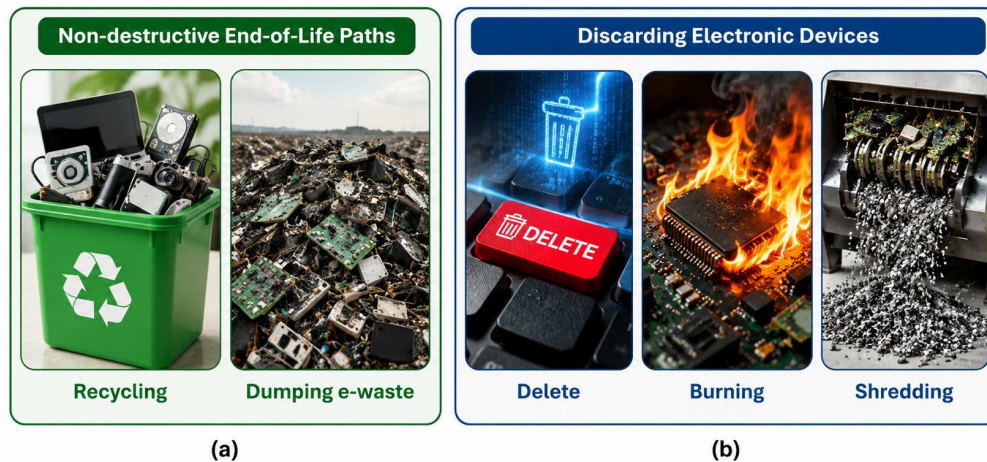


Figure 6.1: End-of-life handling paths for electronic devices: (a) non-destructive end-of-life paths and (b) discarding electronic devices.

Conventional assumptions about secure disposal are therefore insufficient for aged SRAM-based systems. Logical deletion, power cycling, or simply shutting the system down does not address the physical bias that develops inside the cell as a result of aging. Even when the memory contents are overwritten during normal operation, the cell may still retain an altered preference at the transistor level, causing the future power-up state to reveal information about earlier data patterns. Physical destruction methods such as shredding or burning can reduce this risk, but they are often undesirable in practice because they may be costly, environmentally harmful, difficult to verify at the component level, or incompatible with reuse and recovery workflows. In many real-world scenarios, components are not immediately destroyed after decommissioning; instead, they pass through intermediate handling, resale, storage, repair, or reclamation paths where the memory may remain intact and readable. This creates a strong need for a sanitization method that

directly targets the physical origin of residual SRAM bias rather than relying only on conventional assumptions about volatility.

A proper sanitization mechanism for aged SRAM must therefore satisfy a stricter requirement than ordinary data overwrite. It must not only write a new pattern into the array, but also suppress, neutralize, or replace the residual device-level imbalance responsible for aging-induced imprinting. In other words, sanitization must alter the future power-up preference of the SRAM cell itself. This requirement is particularly important for secure end-of-life management of embedded systems, where the memory is often deeply integrated into larger platforms and cannot be individually removed or destroyed without affecting the rest of the hardware. A memory-aware sanitization technique is also important for recycling and controlled reuse, where the goal is not to destroy the device, but to ensure that any previously stored confidential data can no longer be inferred from its power-up behavior.

The sanitization problem studied in this thesis is motivated by that exact challenge. The previous chapter established that aging-induced reverse imprinting can be strong, persistent, and observable across multiple commercial SRAM technologies. The next question is whether such imprints can be effectively removed. This chapter addresses that question by examining a sanitization flow in which aged SRAM arrays are first overwritten with a new target pattern and then exposed to ionizing radiation while maintained in the powered-on condition. The rationale for this approach is that the overwritten state defines the desired target bias, while radiation exposure modifies the device characteristics of the SRAM cell and can potentially force the future power-up state away from the previously aged imprint and toward the newly written sanitization pattern.

Among the possible mechanisms available for physically altering device behavior, ionizing radiation provides a particularly interesting path because it directly affects transistor threshold voltages through charge trapping and related effects in the oxide and interface regions. In the context of SRAM cells, such shifts can modify the mismatch between the pull-down NMOS devices and thereby alter the preferred post-irradiation power-up state. When combined with a controlled overwrite step, this creates the possibility of intentionally steering the future power-up state toward

a chosen all-zero or all-one condition. If successful, such an approach would provide a practical way to suppress previously imprinted information without requiring destructive treatment of the entire device.

This chapter focuses on total ionizing dose (TID)-based sanitization of aged SRAM arrays using a Cs-137 gamma source. The study begins with SRAM chips that have already developed measurable aging-induced imprints under thermal stress. These imprinted locations are then overwritten with controlled sanitization patterns, specifically all-zero or all-one states, and irradiated in the powered-on condition. After each irradiation step, the post-irradiation power-up state is measured to evaluate how effectively the previous imprint has been removed and how strongly the new sanitization pattern has influenced the power-up behavior of the SRAM. In this way, the study examines not only whether sanitization is possible, but also how rapidly it develops as a function of accumulated dose and how the required dose depends on technology node.

In addition to the immediate post-irradiation response, this chapter also examines the long-term stability of the sanitized state through room-temperature relaxation measurements. This step is important because a useful sanitization method must do more than temporarily mask the earlier imprint. It must provide a durable change in the SRAM power-up behavior so that traces of the aged pattern do not reappear after the radiation stress is removed. By monitoring the evolution of the sanitized power-up state over time, the study evaluates whether the radiation-induced transformation is persistent and whether the original imprinted information has been effectively and permanently suppressed.

The results presented in this chapter establish the effectiveness of TID-induced sanitization as an end-of-life mitigation approach for aged SRAM-based systems. The study evaluates sanitization across multiple aged SRAM devices, compares the required dose across technology nodes, examines the behavior of all-zero and all-one overwrite conditions, and analyzes the transistor-level mechanisms responsible for the observed transformation. Together, these results provide a foundation for understanding radiation-based sanitization as a memory-aware strategy for aged SRAM devices intended for disposal, recycling, or controlled reuse.

6.2 Radiation Effects on SRAM

Recent studies have shown that prolonged retention and aging, especially under elevated temperature, can induce data imprinting in SRAM power-up states by increasing the threshold-voltage mismatch between the pull-up PMOS transistors, with NBTI identified as the dominant mechanism. In this process, the PMOS device that remains biased in the ON state during data retention experiences a larger threshold-voltage shift than its complementary device, thereby altering the subsequent power-up preference of the SRAM cell. This aging-induced mismatch can bias the power-up state toward previously stored information and creates a pathway for unintended data remanence [10, 30].

A number of prior works have explored the security implications of this phenomenon. Hovanes et al. demonstrated that elevated-temperature stress over approximately 12 hours can imprint recognizable images into SRAM power-up states and enable partial data recovery from aged arrays. Garg et al. further showed that increasing the supply voltage accelerates NBTI, making the power-on state more predictable and thereby strengthening the imprinting effect. Together, these results establish that both temperature and supply-voltage stress can accelerate aging-induced imprinting and motivate the need for techniques that can remove or suppress such residual bias in aged SRAM devices [10, 30].

Ionizing radiation introduces a related but distinct class of effects in SRAM. Prior research has examined radiation-induced changes in subthreshold leakage, noise margin, data-retention voltage, and the reliability of SRAM-based PUFs. In general, ionizing radiation such as X-rays and gamma rays produces parametric threshold-voltage shifts rather than conventional soft errors. These shifts arise primarily from charge trapping in the gate oxide and from trap generation at the oxide–silicon interface, which in turn modify carrier mobility and transistor drive strength. Because SRAM power-up behavior depends on the relative strength of paired PMOS and NMOS transistors, such radiation-induced parametric changes can directly influence the subsequent power-up state of the cell [15].

Figure 6.2 illustrates a device-level view of this bias-dependent radiation response for two NMOS transistors in different operating conditions during irradiation. In the example shown, transistor N_1 is OFF during irradiation, whereas transistor N_2 is ON. Exposure to ionizing radiation generates trapped charge in the oxide and creates interface traps near the Si–SiO₂ boundary. Since the bias conditions are different for the two devices, the amount and distribution of trapped charge and interface traps are also different. As illustrated in Figure 6.2, the ON transistor N_2 experiences a stronger radiation-induced modification than the OFF transistor N_1 . This bias-dependent response produces an imbalance in the post-irradiation threshold voltages, which can be represented as $V_{th}^{N_1} \gg V_{th}^{N_2}$. Such a threshold-voltage mismatch directly affects the relative pull-down strengths of the paired NMOS transistors in the SRAM cell and can therefore alter the preferred power-up state after irradiation.

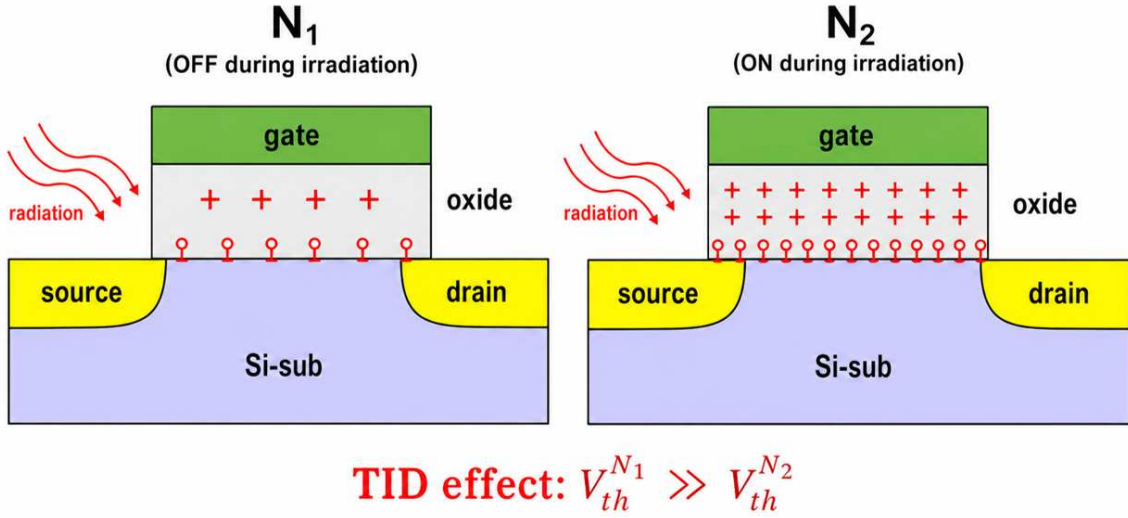


Figure 6.2: Bias-dependent radiation response in two NMOS transistors: N_1 is OFF during irradiation and N_2 is ON during irradiation.

Previous work has also shown that radiation can be used to intentionally alter SRAM power-up characteristics. It has been reported that suitable data patterns written during irradiation can steer the post-irradiation power-up state in a controlled manner, and earlier studies observed both direct and reverse imprinting under very high radiation doses. However, many of those earlier

investigations focused on older technology nodes or on extremely high total ionizing dose levels, which limits their relevance to contemporary SRAM technologies and realistic threat conditions. More recent work therefore emphasizes the need to study how modern COTS SRAMs respond to low- and moderate-dose ionizing radiation and how these responses interact with pre-existing aging-induced bias [15].

Taken together, the current literature suggests that SRAM power-up behavior is sensitive to both long-term aging and ionizing-radiation exposure, and that both mechanisms act through transistor-level threshold-voltage modification. Aging-induced imprinting is typically governed by NBTI-driven PMOS mismatch, whereas radiation-induced effects arise from oxide charge trapping, interface-trap creation, and associated changes in device strength. These findings provide the physical basis for investigating radiation not only as a source of imprinting risk, but also as a possible means of sanitizing aged SRAM arrays by deliberately reshaping their future power-up-state bias [10, 15, 30].

6.3 Experimental Flow

The sanitization study was carried out on COTS SRAM chips from two technology nodes, namely CY 150 nm and ISSI 180 nm. The experimental setup and overall flow used for this study are shown in Figure 6.3. The irradiation experiments were performed at Colorado State University’s Gamma Irradiation Facility using a Cs-137 source with a dose rate of 8.2 rad(Si)/s. The goal of this study was to determine whether aging-induced imprints in SRAM arrays could be effectively removed by overwriting the imprinted memory locations with a new pattern and subsequently exposing the chips to ionizing radiation in the powered-on condition.

The sanitization flow builds directly on the aging study presented in the previous chapter. Each device was first characterized in its fresh state, then thermally aged to induce measurable imprinting, and finally subjected to radiation-based sanitization. This sequence was chosen because the objective of this chapter is not merely to write a new data pattern into a fresh SRAM array, but to

examine whether an already aged and imprinted SRAM can be driven toward a new and controlled post-irradiation state.

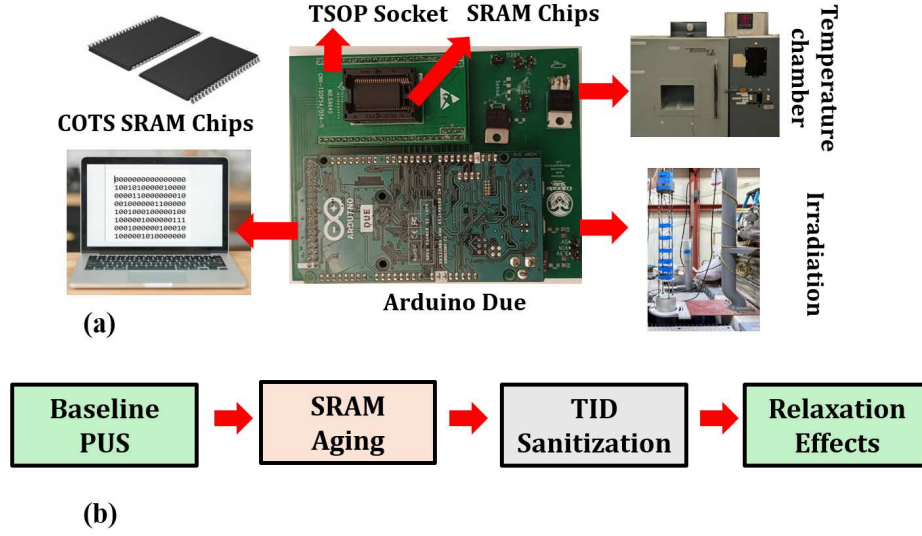


Figure 6.3: (a) Sanitization setup and (b) flow.

The sanitization experimental flow consists of five main stages, as described below.

1. **Baseline and Aging Characterization:** Prior to irradiation, each SRAM device was first characterized to determine its baseline PUS. After baseline acquisition, multiple copies of the Einstein image were written into different address locations of the SRAM array. The chip was then thermally aged in the powered-on condition for several hours to induce data imprinting. After baking, the PUS of the aged chip was read out multiple times to confirm that the intended aging-induced imprint had formed.

To quantify the imprinting effect before sanitization, the Hamming distance between the post-aging PUS and the written image was computed. In this chapter, this metric is denoted as HD^{IM} and is defined as

$$HD^{IM}(\%) = \frac{\# \text{ of set bits } (PUS \oplus \text{Image})}{\text{Total number of bits}} \times 100. \quad (6.1)$$

A value of $HD^{IM} = 0\%$ indicates perfect forward imprinting, whereas a value close to 50% indicates negligible imprinting when both the image and the PUS contain approximately equal numbers of zeros and ones.

2. **Sanitization Characterization:** Once the aged state was confirmed, the imprinted memory locations were overwritten with a controlled sanitization data pattern. Two overwrite conditions were used in this study: an all-zero pattern corresponding to a solid black image, and an all-one pattern corresponding to a solid white image. This overwrite step established the target state that the subsequent irradiation process was intended to reinforce.

By writing a known all-zero or all-one pattern before irradiation, the experiment ensures that the post-irradiation PUS can be interpreted relative to a well-defined sanitization target. This also enables direct comparison between the two sanitization modes, namely black-pattern sanitization and white-pattern sanitization.

3. **Irradiation Characterization:** After the overwrite step, the SRAM chips were irradiated using the Cs-137 gamma source in the powered-on condition. The radiation exposure was applied in multiple dose steps from 0 to 100 krad(Si), rather than in a single continuous exposure. This stepwise approach made it possible to track the evolution of sanitization as a function of accumulated TID.

Maintaining the chip in the powered-on condition during irradiation is an important part of the flow because the stored overwrite pattern remains electrically present in the SRAM cells while the device accumulates radiation dose. This allows the stored state and radiation-induced threshold-voltage modifications to interact during exposure. At each dose step, the chip was irradiated for the duration required to reach the target cumulative dose, based on the known dose rate of the source.

After each irradiation step, post-irradiation PUS collection was performed. The SRAM content was first read back to verify that there was no data loss during irradiation. A power cycle was then performed by briefly turning off the power supply to the SRAM chip and

subsequently restoring it. The PUS was then measured again at the same memory locations that had previously held the overwrite pattern.

This post-irradiation PUS collection is the key observation point in the sanitization experiment. It reveals whether the future power-up state of the SRAM has shifted toward the new overwrite pattern after radiation exposure. By repeating the same procedure after each dose step, the progression of sanitization can be monitored continuously rather than only at the final dose.

4. **Sanitization Metric:** To quantify sanitization efficiency, the post-irradiation PUS was analyzed using Hamming weight (HW), defined as the percentage of logic ‘1’ bits in the measured PUS:

$$HW(\%) = \frac{\# \text{ of 1 bits in PUS}}{\text{Total number of bits}} \times 100. \quad (6.2)$$

For the all-one overwrite condition, $HW = 100\%$ indicates complete sanitization, since every bit in the post-irradiation PUS matches the desired all-one target. For the all-zero overwrite condition, $HW = 0\%$ indicates complete sanitization, since no logic ‘1’ bits remain in the measured PUS. Intermediate values indicate partial sanitization.

The HW metric is especially suitable for this study because the sanitization targets are spatially uniform. Unlike the aging-imprint analysis, where image reconstruction is useful for visual interpretation, the sanitization process can be directly quantified by the fraction of ones in the measured post-irradiation PUS.

5. **Relaxation Characterization:** After completion of the irradiation sequence, the sanitized chips were stored at room temperature with all pins grounded in order to study the long-term stability of the sanitized state. The PUS was periodically measured over several months to observe the temporal evolution of the sanitized pattern after removal of the radiation source. This final step was included to determine whether the radiation-induced sanitization effect is stable or whether the SRAM gradually relaxes away from the target state over time. The re-

laxation study is important because a useful sanitization technique must not only suppress the previously imprinted pattern immediately after treatment, but also prevent its reappearance during long-term storage. Together, the overwrite step, powered-on irradiation, repeated post-irradiation readout, HW-based analysis, and long-term relaxation monitoring form the complete sanitization experimental flow used in this thesis.

6.4 Results and Discussion

6.4.1 Overview of Sanitization Behavior

Figure 6.4 shows the measured results obtained during each experimental stage, namely, baseline PUS, SRAM aging, sanitization, and relaxation effects. The reference power-up state represents a pepper-salt image highlighting the natural randomness of the SRAM. We then write an Einstein image and thermally age the chip at 120°C for 18 hours. The subsequent post-aging power-up state is shown in Figure 6.4(c), which exhibits reverse imprinting.

Next, we overwrite the same location with an all-zero or all-one pattern and irradiate the chip up to 100 krad(Si). After irradiation, we observe a sanitized black or white imprint, as shown in Figure 6.4(e). Finally, we monitor relaxation effects at room temperature and find that the PUS evolves to a new random state with no visible traces of the aged Einstein pattern.

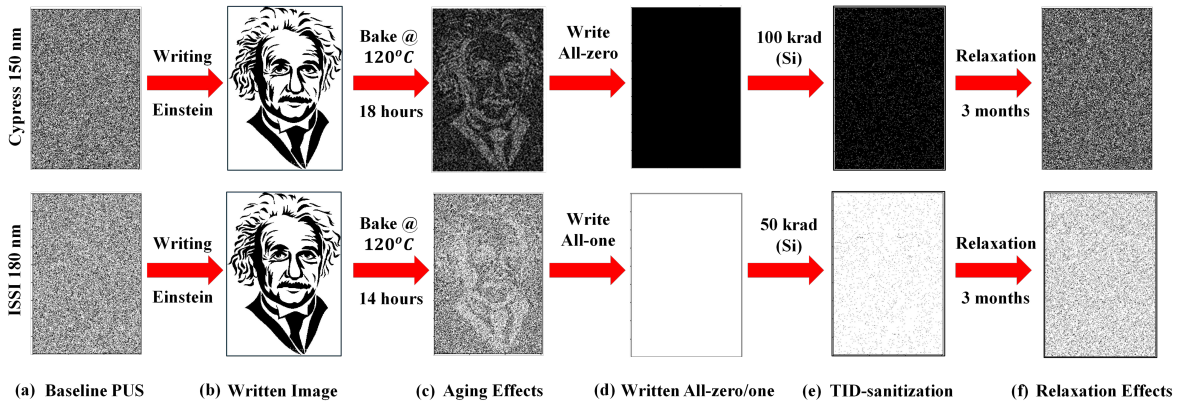


Figure 6.4: Sanitization results overview.

6.4.2 Sanitization of Aged SRAMs Using Gamma Rays

To perform sanitization in the aged SRAMs, we first overwrite the aged chips with solid black or white patterns in the same aged memory locations of the SRAM array and subsequently irradiate them in the powered-on condition. After irradiation, we read the SRAM content to verify that there was no data loss during irradiation. Next, we initiate a power-up cycle by deactivating the power supply to the SRAM chip, allowing for a brief interval of a few seconds before reactivating it. We then read the SRAM locations that were holding the newly written image pattern (solid black or white in this case) to quantify the sanitization efficiency using HW. We repeat this entire process for different amounts of TID ranging from 0 to 100 krad(Si). The aged SRAM PUS is shown in Figure 6.4(c). However, when subjected to irradiation, the PUS undergoes significant changes due to the increase in threshold-voltage mismatch between the NMOS transistors [9, 11].

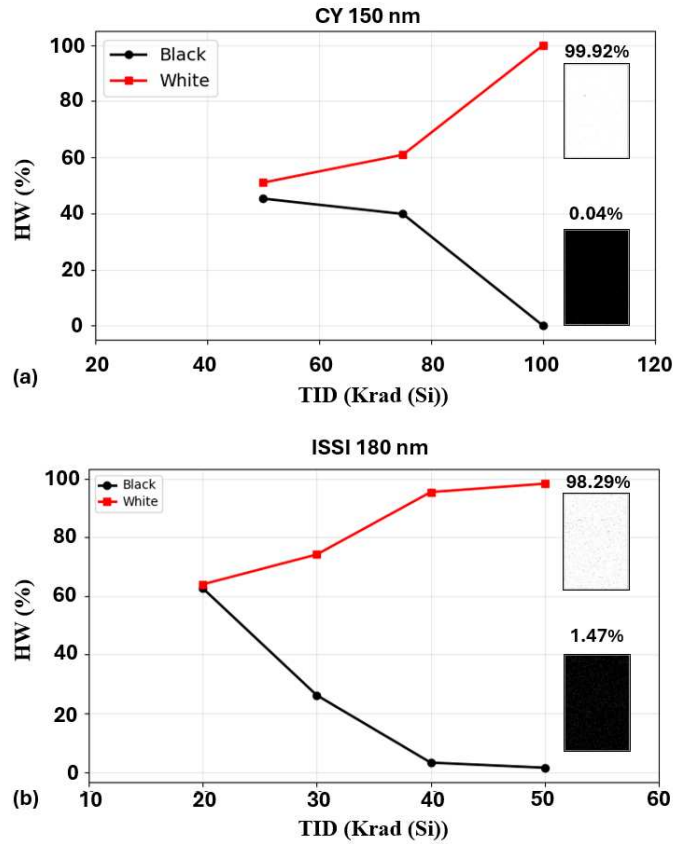


Figure 6.5: TID-based sanitization results (a) CY 150 nm and (b) ISSI 180 nm.

Figure 6.5(a) and Figure 6.5(b) show the HW percentage as a function of TID for two thermally aged SRAMs from two different technology nodes, namely, (a) CY 150 nm and (b) ISSI 180 nm. From Figure 6.5(b), the older technology node shows higher sensitivity to TID-induced sanitization. Both the CY 150 nm and ISSI 180 nm chips reach near-complete sanitization of approximately 99%. However, the ISSI 180 nm chip sanitizes much faster, reaching this condition by 50 krad(Si), while the CY 150 nm device requires 100 krad(Si). Since smaller or newer technology nodes employ thinner gate dielectrics, sanitization requires higher TID for more advanced technology nodes [9, 37]. Overall, Figure 6.5(a) and Figure 6.5(b) demonstrate that TID-induced sanitization is possible in SRAM memory arrays, but the minimum TID required for achieving complete sanitization may vary across different technology nodes [9, 11, 37].

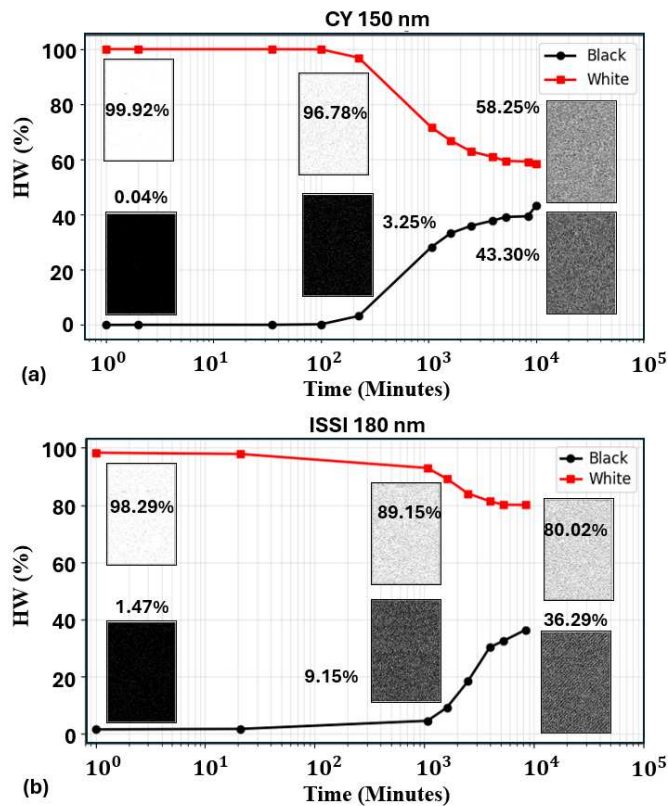


Figure 6.6: Sanitization relaxation results (a) CY 150 nm and (b) ISSI 180 nm.

6.4.3 Relaxation Effects on Sanitization

To examine the durability of the sanitization effects, we allow all irradiated chips to relax at room temperature with all pins grounded. We analyze the progression of HW% over time by reading the power-up state at multiple intervals for several months. Figure 6.6(a) and Figure 6.6(b) show the relaxation behavior of the two sanitized SRAMs.

In both chips, HW% gradually decreases with time. During the first hour, the sanitized images remain nearly unchanged with minimal visual degradation. After the first hour, the decrease in the sanitized PUS accelerates for a few hours, and then evolves into a new random pattern after a day. Notably, even after three months, the aged Einstein pattern does not reappear, indicating that it was permanently erased due to TID effects.

6.4.4 Explanation for TID-Induced Sanitization

Figure 6.7(a) explains the aging effects on the SRAM cell, where mainly the PMOS transistors' threshold voltage is modified due to NBTI effects. Let us consider a 6T SRAM cell storing $Q/\bar{Q} = 1/0$, as shown in Figure 6.7(a). In this state, transistor P1 is ON and P2 is OFF. Since the ON PMOS (P1) is mostly affected by NBTI effects, P1 experiences a significant threshold-voltage magnitude increase, causing threshold-voltage mismatch as follows:

$$|V_{th}^{(P1)}| \gg |V_{th}^{(P2)}|,$$

irrespective of their pre-aged values. This significant change in the threshold-voltage mismatch forces the cell to power up in the opposite state ($Q = 0$), producing a reverse imprint of the written image [10].

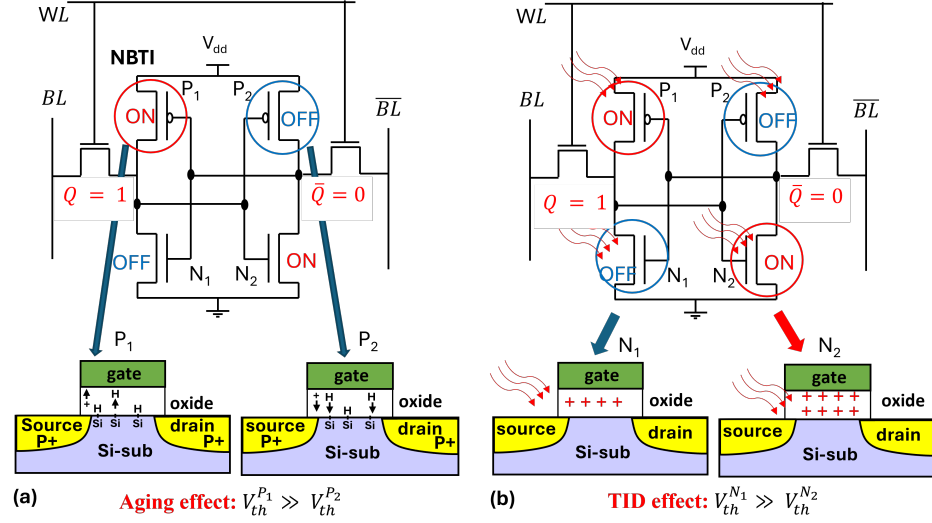


Figure 6.7: TID-based sanitization mechanism.

Table 6.1: Summary of aging and sanitization effects

Case	Aging Effects	Sanitization Effects
Case 1 Aging data: 1 Sanitization: 1	P1 ON, P2 OFF $ V_{th}^{(P1)} > V_{th}^{(P2)} $, $Q = 0$	N1 OFF, N2 ON $V_{th}^{(N1)} > V_{th}^{(N2)}$, $Q = 1$
Case 2 Aging data: 1 Sanitization: 0	P1 ON, P2 OFF $ V_{th}^{(P1)} > V_{th}^{(P2)} $, $Q = 0$	N1 ON, N2 OFF $V_{th}^{(N1)} < V_{th}^{(N2)}$, $Q = 0$
Case 3 Aging data: 0 Sanitization: 1	P1 OFF, P2 ON $ V_{th}^{(P1)} < V_{th}^{(P2)} $, $Q = 1$	N1 OFF, N2 ON $V_{th}^{(N1)} > V_{th}^{(N2)}$, $Q = 1$
Case 4 Aging data: 0 Sanitization: 0	P1 OFF, P2 ON $ V_{th}^{(P1)} < V_{th}^{(P2)} $, $Q = 1$	N1 ON, N2 OFF $V_{th}^{(N1)} < V_{th}^{(N2)}$, $Q = 0$

Let us assume that after aging effects the preferred PUS is $Q = 0$, in which case the mismatch between the NMOS transistors is not significant enough to influence the power-up state. Now, all-one sanitization requires irradiation of the SRAM cell with $Q = 1$ written on it, as illustrated in Figure 6.7(b). Since N1 remains OFF and N2 remains ON during irradiation, N2 is more strongly affected by irradiation and experiences hole trapping in its oxide, resulting in a significant reduction

in its threshold voltage. Post-irradiation, the mismatch between the NMOS transistors (N1 and N2) leads to

$$V_{th}^{(N1)} \gg V_{th}^{(N2)},$$

due to differential charge trapping influenced by their ON/OFF states. Consequently, the SRAM cell's preferred PUS shifts to $Q = 1$ after irradiation, effectively sanitizing the data held by the cell during irradiation.

Table 6.1 summarizes how thermal aging and TID sanitization change the NMOS/PMOS threshold voltages under four different scenarios. Consider Case 3, where the aging data is $Q = 0$. In this state, P1 is OFF and P2 is ON, so after aging, $|V_{th}^{(P1)}| \ll |V_{th}^{(P2)}|$. As a result, during the next PUS, the cell flips to the opposite state ($Q = 1$), which produces reverse imprinting. For sanitization in Case 3, $Q = 1$, so N1 is OFF and N2 is ON during irradiation. Irradiation can create an NMOS mismatch such that $V_{th}^{(N1)} > V_{th}^{(N2)}$, which biases the next PUS toward $Q = 1$. The radiation-induced NMOS skew reinforces $Q = 1$, so sanitization is more effective and stronger. Cases 1, 2, and 4 from Table 6.1 follow similar mechanisms, but with different aging and sanitization data.

6.5 Conclusion

- **Radiation-Based Sanitization of Aged SRAM:** This chapter demonstrated that aging-induced imprints in SRAM power-up states can be effectively suppressed using radiation-based sanitization. The proposed sanitization flow overwrites the previously aged memory locations with a controlled all-zero or all-one pattern and then exposes the powered-on SRAM chip to ionizing radiation using a Cs-137 gamma source. The results show that the post-irradiation PUS shifts away from the previously aged Einstein imprint and toward the newly written sanitization pattern, confirming that TID exposure can modify the future power-up preference of aged SRAM cells.

- **Sanitization Effectiveness Across Technology Nodes:** The sanitization experiments performed on CY 150 nm and ISSI 180 nm SRAM chips showed that both devices can reach near-complete sanitization of approximately 99%. However, the required total ionizing dose depends strongly on the technology node. The ISSI 180 nm chip reached near-complete sanitization at 50 krad(Si), whereas the CY 150 nm chip required 100 krad(Si). This difference indicates that the radiation dose required for sanitization is technology-dependent, with older technology nodes showing stronger sensitivity to TID-induced sanitization.
- **Sanitization Pattern:** This chapter also showed that the overwrite pattern applied before irradiation plays a central role in determining the final sanitized PUS. When the aged SRAM locations were overwritten with an all-zero or all-one pattern and irradiated in the powered-on condition, the stored overwrite state influenced the radiation-induced threshold-voltage mismatch inside the SRAM cell. As a result, the future PUS could be intentionally steered toward the desired black or white sanitization state. This confirms that radiation alone is not the only factor; the combination of controlled overwrite and powered-on irradiation is critical for effective sanitization.
- **Durability of the Sanitized State:** The room-temperature relaxation measurements showed that the sanitized PUS evolves with time after irradiation. In both CY 150 nm and ISSI 180 nm devices, the HW% gradually decreased after the radiation treatment, and the sanitized images eventually evolved into a new random pattern. However, even after three months of relaxation, the original aged Einstein pattern did not reappear. This result confirms that the radiation-based sanitization process does not merely mask the aged imprint temporarily, but effectively removes the previously imprinted data from the future power-up behavior of the SRAM.
- **Device-Level Mechanism of TID-Induced Sanitization:** The physical explanation presented in this chapter shows that aging-induced reverse imprinting is mainly caused by NBTI-driven PMOS threshold-voltage mismatch, whereas radiation-based sanitization is

driven by bias-dependent NMOS threshold-voltage changes during irradiation. When the SRAM cell is irradiated in the powered-on condition, the ON and OFF NMOS transistors experience different radiation-induced charge trapping effects. This creates a new NMOS mismatch that can dominate the earlier aging-induced PMOS mismatch and shift the preferred PUS toward the sanitization pattern written during irradiation.

- **Security Implication for End-of-Life SRAM Devices:** Overall, the results of this chapter establish TID-induced sanitization as a practical end-of-life mitigation technique for aged SRAM devices. Since conventional approaches such as power cycling, logical deletion, or ordinary overwrite do not remove transistor-level aging bias, radiation-based sanitization provides a memory-aware method for suppressing residual data imprints. These findings are especially important for SRAM-based systems that may be recycled, reused, repaired, or handled through non-destructive disposal paths where the memory remains physically intact and accessible.

Chapter 7

Dosimeter Using SRAM Power-Up State

7.1 Introduction

Dosimetry plays a critical role in protecting both personnel and electronic systems from the harmful effects of ionizing radiation. A dosimeter enables measurement of accumulated radiation dose over time and is therefore an essential component of radiation monitoring, safety assessment, and exposure control. In practice, radiation environments span a very wide range, from low-dose occupational exposure in medical, industrial, and nuclear settings to extremely high-dose conditions encountered in severe accident scenarios and space systems. Figure 7.1 illustrates representative low-dose and high-dose radiation environments and applications relevant to dosimetry [55–57].

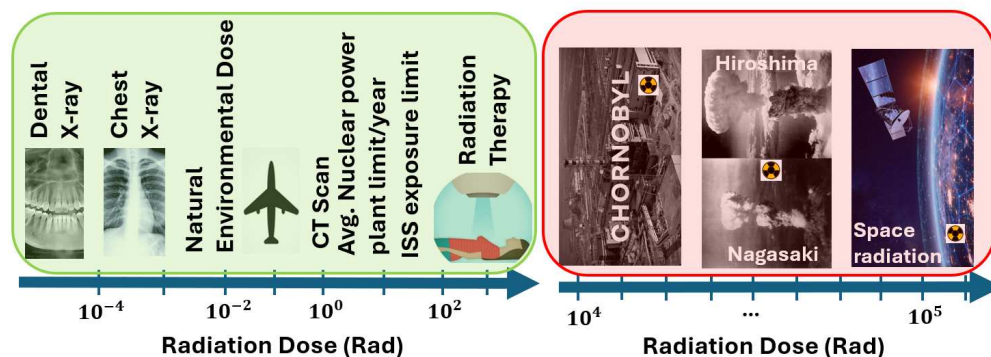


Figure 7.1: Representative low-dose and high-dose radiation environments and applications.

In high-radiation scenarios such as nuclear accidents, rapid and reliable dose estimation is essential for immediate situational awareness and safety decision-making. This requirement is different from routine occupational monitoring, where repeated low-dose exposure may accumulate gradually over time and still exceed acceptable safety limits. These different use cases place different demands on dosimeter design. A dosimeter intended for routine low-dose monitoring may not

remain suitable in very high-dose environments, which creates a need for compact and practical total-dose monitoring technologies that can operate across a broader radiation range [55, 56].

A variety of dosimeter technologies have been developed over the years, each with specific advantages and limitations. Traditional radiographic and radiochromic films can indicate radiation exposure, but they are limited in direct quantitative dose measurement. Thermoluminescent dosimeters (TLDs) and optically stimulated luminescence dosimeters (OSLDs) provide improved dose quantification for many practical applications, but they generally require specialized analysis equipment and off-line processing. Ion-chamber-based sensors offer high accuracy and resolution, although they tend to be bulky. Radiation-sensitive field-effect transistor (RADFET)-based dosimeters provide an attractive solid-state alternative, but they require dedicated readout circuitry and are better suited to higher dose ranges. Floating-gate dosimeters offer high sensitivity and resolution, but they also introduce readout complexity and temperature dependence. Figure 7.2 summarizes the major classes of radiation sensors and their general characteristics [57–62].

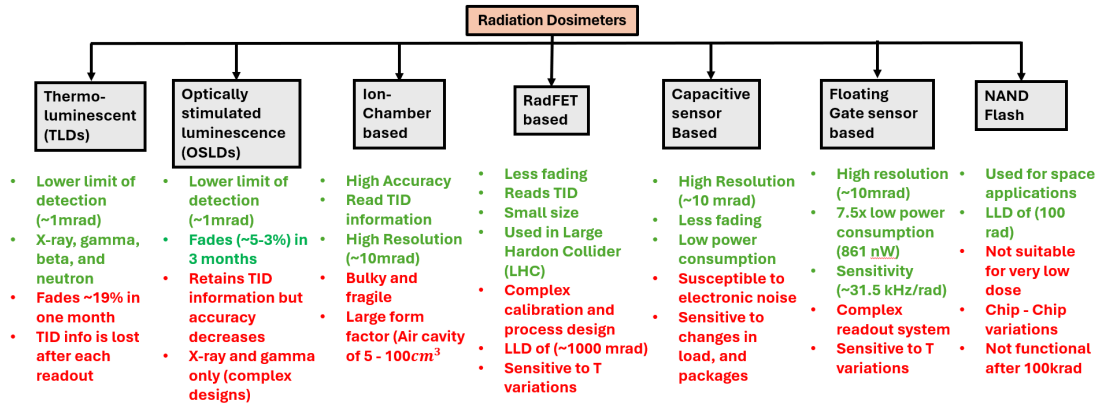


Figure 7.2: Representative radiation dosimeter technologies and their general characteristics.

More recent silicon-based dosimetry approaches have explored commercial memory and CMOS-compatible structures. Floating-gate MOSFET dosimeters and related integrated floating-gate sensing circuits have shown promising behavior for dose measurement. NAND flash memories have also been studied as dosimeters because radiation-induced threshold-voltage shifts can be observed through memory-state degradation. However, flash-based dosimeters typically lose use-

fulness at higher dose levels because peripheral circuitry such as charge pumps can fail. SRAM-based active radiation detectors using single-event upset (SEU) behavior have also been reported, but those approaches are intended for particle-induced upset detection and are not appropriate for measuring TID caused by X-rays or gamma rays [62–68].

These limitations motivate the need for a real-time dosimetry approach that is compact, low-cost, digitally readable, and compatible with existing electronic platforms, while also remaining functional in high-dose environments. In this chapter, the PUS of SRAM is investigated as a radiation dosimeter. Although the SRAM PUS has traditionally been studied as a chip fingerprint or PUF for authentication, the underlying power-up behavior is highly sensitive to radiation-induced device-parameter shifts. This sensitivity can be exploited to convert radiation-induced changes in the PUS into a measurable dose. [64, 65, 68].

The dosimetry concept investigated in this work is illustrated in Figure 7.3. A pre-irradiation SRAM PUS is first captured as a reference state. After radiation exposure, the post-irradiation PUS is measured again and compared with the pre-irradiation state. The radiation-induced change in the power-up pattern serves as the basis for digital dose readout. In the underlying experimental study, three COTS SRAM chips from Cypress, fabricated in 90 nm, 150 nm, and 250 nm technologies, were evaluated. All three chips showed measurable sensitivity at relatively low radiation dose, while remaining operational up to much higher dose levels. The older technology nodes also exhibited stronger sensitivity and better linearity over a useful TID range, indicating that SRAM PUS can serve as a practical basis for real-time dosimetry in high-radiation environments [62–64].

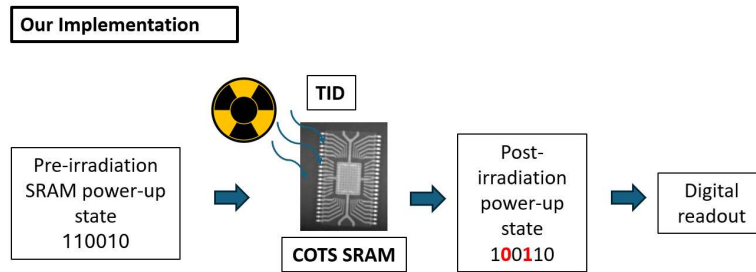


Figure 7.3: Conceptual overview of SRAM-PUS-based radiation dosimetry.

This chapter is organized as follows. It first introduces the dosimetry motivation and the limitations of conventional and existing silicon-based radiation sensors. It then presents the use of SRAM PUS as a digital dosimetry signal, followed by the experimental setup, calibration approach, and radiation-response characterization discussed in the later sections of the chapter.

7.2 Background

A conventional six-transistor (6T) SRAM cell consists of two cross-coupled CMOS inverters and two access transistors, as shown in Figure 7.4. The cross-coupled inverter pair provides two stable states corresponding to logic ‘0’ and logic ‘1’. After power-up, each SRAM cell settles into one of these two states depending on the small process-variation-induced mismatch between the transistors on the two sides of the cell. As a result, the power-up pattern of an SRAM array forms a random but device-specific signature that can be used as a digital fingerprint or as a Physical Unclonable Function (PUF) [?, 6, 7, 15].

Although repeated power-up measurements from the same SRAM chip are generally similar, they are not perfectly identical. A small number of cells may switch states across repeated captures due to electrical noise and marginal cell stability. This behavior is illustrated in Figure 7.4(b), where several representative PUF states are shown together with a few unstable bit positions. These unstable bits reduce the repeatability of the PUF response and therefore reduce its accuracy. To obtain a stable reference state, a GoldPUF can be generated by repeatedly measuring the same SRAM location and applying majority voting at each bit position. In the experimental study associated with this chapter, the GoldPUF was generated using 101 consecutive power-up measurements from the same SRAM region.

The difference between a measured PUF and the corresponding GoldPUF is quantified using the Hamming Distance (HD), defined as

$$HD(\%) = \frac{\# \text{ of set bits (GoldPUF} \oplus \text{CurrentPUF)}}{\text{Total number of PUF bits}} \times 100. \quad (7.1)$$

This metric provides a direct measure of how much the current power-up state deviates from the reference power-up pattern. In the context of dosimetry, radiation-induced changes in the SRAM power-up behavior can therefore be captured as an increase in HD with respect to the pre-irradiation GoldPUF.

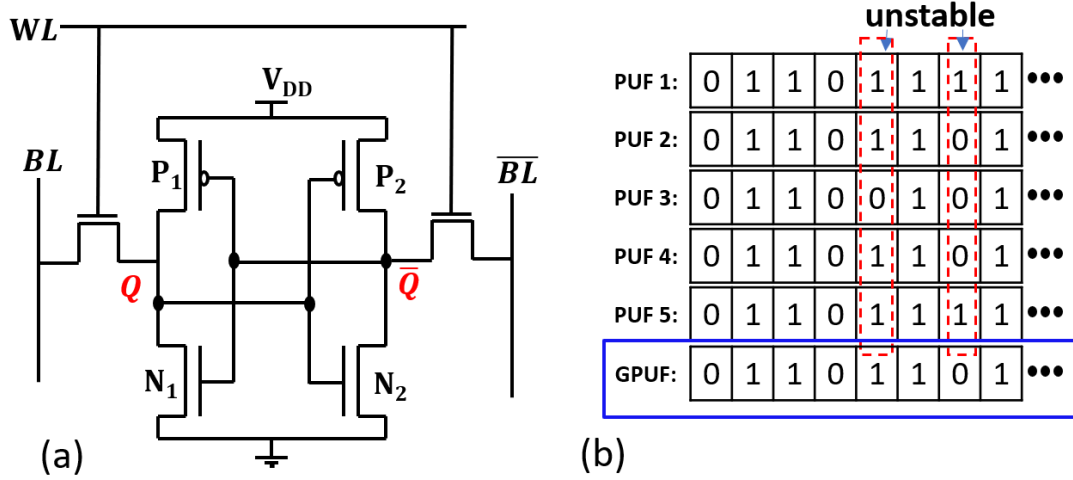


Figure 7.4: Background of the SRAM-PUS-based dosimeter: (a) schematic of a 6T SRAM cell and (b) illustration of GoldPUF generation from repeated power-up states.

Gamma rays interact with an SRAM cell primarily by ionizing the gate dielectric materials and generating electron-hole pairs. Since electrons are more mobile, they may escape through the gate or the substrate, while holes are more likely to become trapped in the oxide. This charge trapping modifies the electrical characteristics of the transistors. In particular, positive trapped charge in the oxide can reduce the threshold voltage of NMOS devices and increase the threshold voltage of PMOS devices. In addition, total ionizing dose (TID) can create oxide-silicon interface traps and degrade carrier mobility in the channel. These effects together alter the threshold voltages and relative drive strengths of the transistors in the SRAM cell, which can in turn modify the power-up state of the memory after irradiation [9, 69, 70].

Figure 7.5 illustrates this mechanism conceptually. Let the default power-up state of the SRAM cell before irradiation be $Q = 1$ and $\bar{Q} = 0$, as shown in Figure 7.5(a). This preferred power-up state can be understood in terms of the threshold-voltage mismatch between the paired NMOS

devices. Before irradiation, the mismatch may satisfy $V_{th}^{N_2} > V_{th}^{N_1}$, which biases the cell to power up with $Q = 1$. During irradiation, trapped charge and interface-trap generation modify the device characteristics, as illustrated in Figure 7.5(b). After sufficient dose accumulation, the mismatch may reverse to $V_{th}^{N_1} > V_{th}^{N_2}$. This reversal changes the preferred power-up state of the cell from $Q = 1$ to $Q = 0$, as illustrated in Figure 7.5(c). Such a radiation-induced power-up-state change produces an increase in HD relative to the pre-irradiation GoldPUF and provides the basis for using SRAM PUS as a dosimeter [9, 15].

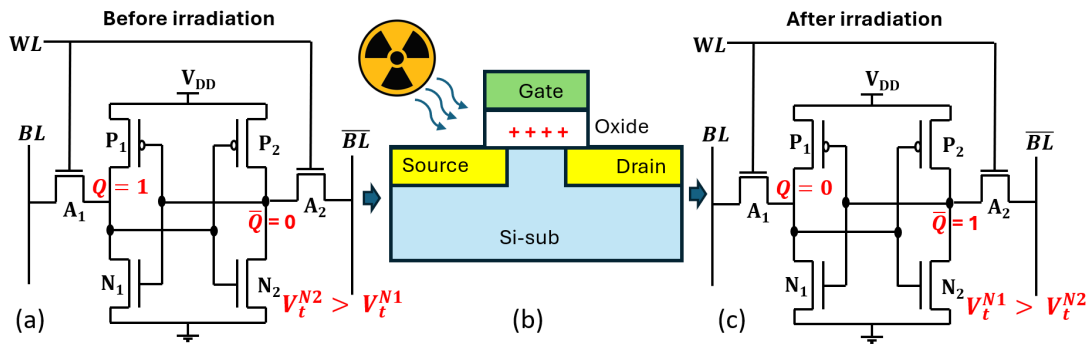


Figure 7.5: Conceptual illustration of radiation-induced power-up-state change in SRAM: (a) pre-irradiation preferred power-up state, (b) device-level irradiation mechanism, and (c) post-irradiation preferred power-up state.

7.3 Experimental Setup

The experimental study in this chapter was carried out using COTS SRAM chips from Cypress fabricated in three different technology nodes: 90 nm, 150 nm, and 250 nm. All three devices have a memory capacity of 4 Mbits and operate at a nominal supply voltage of 3.3 V. The purpose of the setup was to repeatedly capture the SRAM PUS before and after gamma-ray irradiation and to quantify the radiation-induced deviation of the post-irradiation PUS from the pre-irradiation reference state [15].

To interface the memory chips, a custom-designed TSOP-54 socket board was used, as shown in Figure 7.6. The SRAM chip was mounted on the TSOP-54 socket and connected to an Arduino Due microcontroller for automated acquisition of the PUS. The Arduino Due was further connected

to the workstation through its native USB port, which enabled transfer of the measured power-up-state data for storage and offline processing [15].

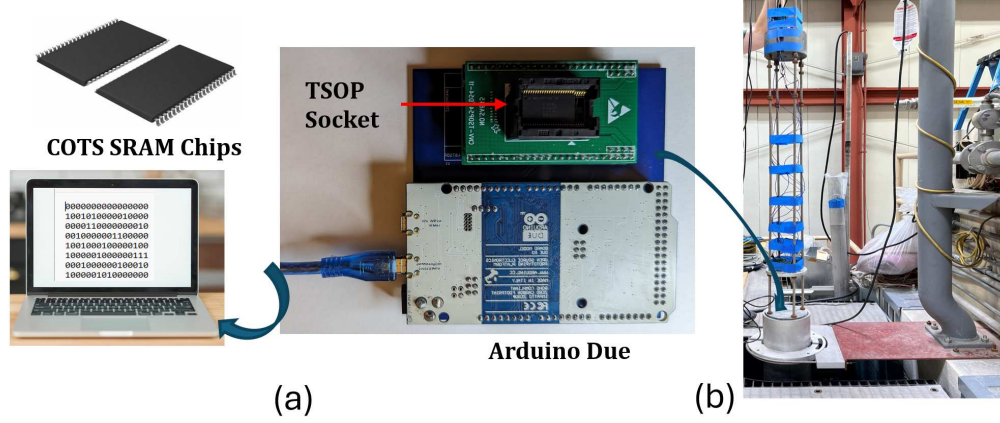


Figure 7.6: Experimental setup for SRAM-PUS dosimetry: (a) SRAM interface platform using a TSOP-54 socket and Arduino Due, and (b) gamma irradiation facility used for the experiments.

The radiation experiments for the 90 nm and 150 nm chips were performed at the Ohio State University Gamma Irradiation Facility using a Co-60 source with a dose rate of 3.25 rad(Si)/s. The radiation experiment for the 250 nm chip was performed at the Colorado State University Irradiation Facility using a Cs-137 source with a dose rate of 8.2 rad(Si)/s. In all cases, gamma irradiation was applied to the packaged TSOP device with all pins grounded, and the direction of the incident radiation was perpendicular to the top surface of the chip [15].

Table 7.1: Summary of SRAM chips used in the dosimeter study

Manufacturer	Capacity	Supply Voltage	Technology Node
Cypress	4 Mbits	3.3 V	90 nm
Cypress	4 Mbits	3.3 V	150 nm
Cypress	4 Mbits	3.3 V	250 nm

The experimental flow followed a repeated reference-and-compare methodology. First, a Gold-PUF was generated for each SRAM region by collecting 101 consecutive power-up states from the same location and then applying majority voting bit by bit. This GoldPUF served as the pre-

irradiation reference state. The memory chips were then exposed to gamma radiation up to the desired dose level. After irradiation, the PUS of the irradiated chip was measured again and compared with the corresponding GoldPUF in order to compute the HD. This HD-based comparison provides the fundamental dosimetry signal used in this work, because radiation-induced shifts in transistor characteristics manifest as measurable changes in the post-irradiation power-up pattern [15].

Thus, the overall setup combines a simple digital readout platform, a controlled gamma-ray exposure environment, and a reference-based PUS comparison framework. This combination enables experimental evaluation of SRAM power-up-state sensitivity to ionizing radiation across multiple technology nodes while keeping the readout and analysis flow entirely digital.

7.4 Results and Discussion

7.4.1 Pre-Irradiation Characterization of SRAM Power-Up State

We begin by performing a baseline characterization of the SRAM PUS by computing its HD with respect to the reference power-up state, namely the GoldPUF. For this analysis, 256 different PUF instances were generated from different memory address locations on the same SRAM chip. Each PUF had a length of 16 kb, and its HD was computed with respect to the corresponding GoldPUF.

Figure 7.7(a) shows the pre-irradiation characterization results for two Cypress SRAM chips fabricated in 90 nm and 150 nm CMOS technologies. The measured HD values show that the 150 nm node exhibits a lower pre-irradiation HD than the 90 nm node. This difference can be attributed to the larger gate area of the individual transistors in the 150 nm device, which makes the cells less susceptible to thermal noise. As a result, the baseline power-up-state deviation from the GoldPUF is smaller in the 150 nm chip than in the 90 nm chip.

A noticeable variability in HD is also observed across the different PUF instances for both chips. To quantify this spread, the variance is defined as

$$\text{Variance} = HD_{\max} - HD_{\min}. \quad (7.2)$$

Figure 7.7(b) shows the dependence of this variance on PUF length. The results indicate that the variance decreases as the PUF length increases. However, the improvement becomes very small beyond a PUF length of 16 kb. Therefore, 16 kb was selected as the PUF length for the subsequent dosimeter analysis, since it provides a practical tradeoff between reduced variability and measurement efficiency.

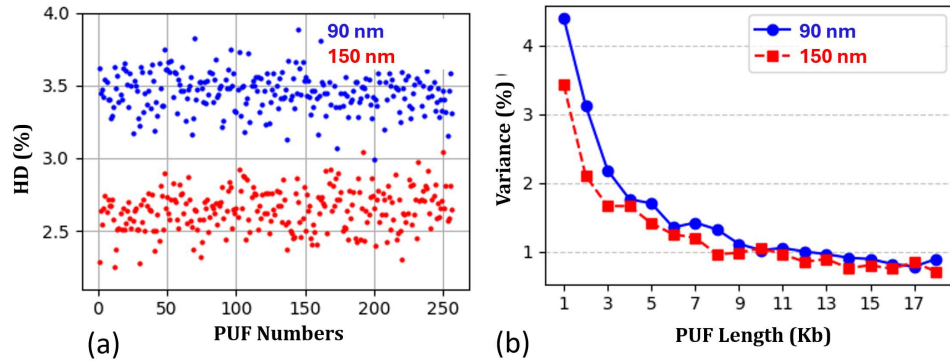


Figure 7.7: Pre-irradiation characterization of the SRAM PUS: (a) HD (%) as a function of PUF number for Cypress 90 nm and 150 nm chips, and (b) variance (%) in HD as a function of PUF length.

7.4.2 Characterization of the SRAM Dosimeter with TID

We evaluate the response of the SRAM dosimeter to TID by monitoring the increase in HD of the corresponding SRAM PUF after irradiation. Figure 7.8(a) shows a scatter plot of HD as a function of TID for the 90 nm chip. Similar results for the 150 nm and 250 nm chips are shown in Figure 7.8(b) and Figure 7.8(c), respectively. The corresponding change in HD, denoted as ΔHD , is calculated with respect to the pre-irradiation condition for each radiation dose. For all three chips, a monotonic increase in HD is observed with increasing TID. In addition, all three chips exhibit a discernible response even at a total dose of 0.5 krad(Si), indicating the lower limit of TID detectability. A significant variability in HD is also observed among different PUFs, particularly for the 90 nm node. This variability may reduce the dosimeter accuracy in the lower TID range. However, at higher TID levels, the variability in HD becomes smaller relative to the absolute change in HD, indicating the suitability of the SRAM dosimeter for higher-dose applications.

To further quantify the HD variability, the results are also presented as distribution plots in Figure 7.8. For example, Figure 7.8(d) shows the Probability Density Function (PDF) as a function of HD for the 90 nm chip, while similar plots for the 150 nm and 250 nm chips are shown in Figure 7.8(e) and Figure 7.8(f), respectively. The solid lines represent fitted Gaussian distributions, while the symbols correspond to the measured HD data. For all three COTS SRAM chips, the measured HD values at different radiation doses closely follow a Gaussian distribution. Notably, the 90 nm node exhibits overlapping distributions for different radiation doses, whereas the 150 nm and 250 nm chips show much less overlap in HD for the same TID range. These results further support the suitability of SRAM PUS for radiation dosimetry.

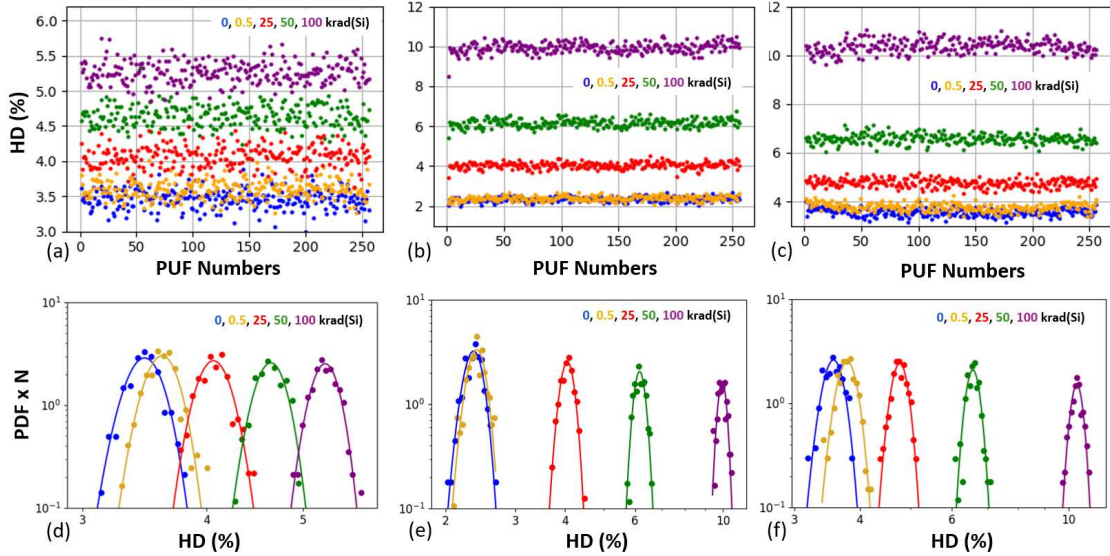


Figure 7.8: (a) HD (%) as a function of PUF Numbers for 90 nm node. The same plots for 150 nm and 250nm node are shown in (b), and (c), respectively. (d) PDF as a function of HD (%) for Cypress 90nm, (e) and (f) represents Cypress 150nm and Cypress 250nm respectively.

Figure 7.9(a) summarizes the scatter plot data from Figure 7.8 using a line plot of the average HD value at each radiation dose. A logarithmic scale is used for TID in order to clearly visualize the full set of data points. The figure also compares the TID responses of the three chips from the same manufacturer but fabricated in different technology nodes. As shown in Figure 7.9(a), all three SRAM chips exhibit a monotonic and nonlinear increase in HD as the radiation dose

increases from 0 krad(Si) to 350 krad(Si). Both SRAM chips remain fully functional even after exposure to 350 krad(Si), demonstrating the applicability of the SRAM dosimeter for higher TID ranges.

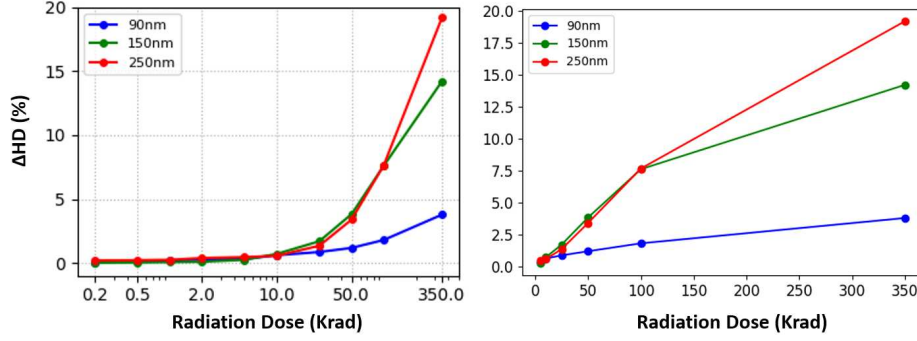


Figure 7.9: (a) ΔHD (%) as a function of Total-Ionizing-Dose (TID) in log scale, (b) ΔHD (%) as a function of TID in Linear scale.

7.4.3 Sensitivity Analysis of the SRAM Dosimeter

We define the sensitivity (S) of the SRAM dosimeter as

$$S = \frac{\Delta HD}{TID}. \quad (7.3)$$

To quantify the sensitivity of the SRAM dosimeter, the average ΔHD was plotted as a function of TID on a linear scale, as shown in Figure 7.9(b). The results show that the 250 nm node exhibits slightly greater sensitivity to TID than both the 90 nm and 150 nm nodes. This trend continues into the higher dose range, where the 250 nm chip shows a noticeably stronger overall response. The increased sensitivity of the 250 nm node can be attributed to the thicker gate dielectric of the individual CMOS transistors, which makes them more susceptible to radiation-induced charge trapping.

In addition, the SRAM dosimeter exhibits an approximately linear response in the dose range from 5 krad(Si) to 100 krad(Si), with this behavior being particularly noticeable for the 150 nm and 250 nm chips. However, for doses greater than 100 krad(Si), all three chips begin to show a

saturating response in terms of ΔHD . Since linearity is generally a desirable characteristic for practical dosimetry, the observed linear region from 5 krad(Si) to 100 krad(Si) is used for further analysis in the next section.

7.4.4 Linear Model Fit of the SRAM Dosimeter

Figure 7.10 presents the HD as a function of radiation dose in the range from 5 krad(Si) to 100 krad(Si). In this dose range, the fitted linear model is represented as

$$y = ax + b, \quad (7.4)$$

where y represents the HD, x denotes the radiation dose, a is the slope, and b is the y-intercept. This equation captures the overall trend observed in the measured data. The slope a represents the rate of change in HD per unit increase in radiation dose, while the intercept b corresponds to the baseline HD when the radiation dose is zero.

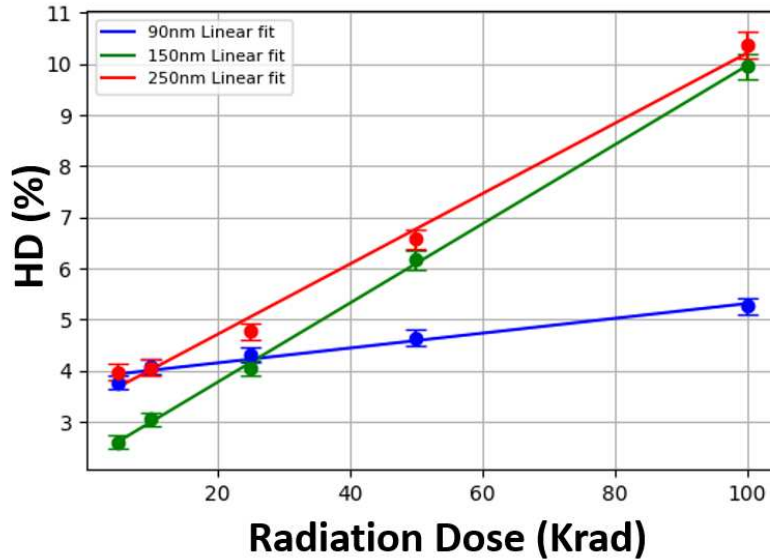


Figure 7.10: HD mean (%) as a function of Total-Ionizing-Dose (TID) for linear model fitting

Table 7.2 summarizes the fitted linear equations for the different technology nodes. The results show that the fitted model parameters vary across technology nodes, indicating that the SRAM

Table 7.2: Technology node vs linear equation

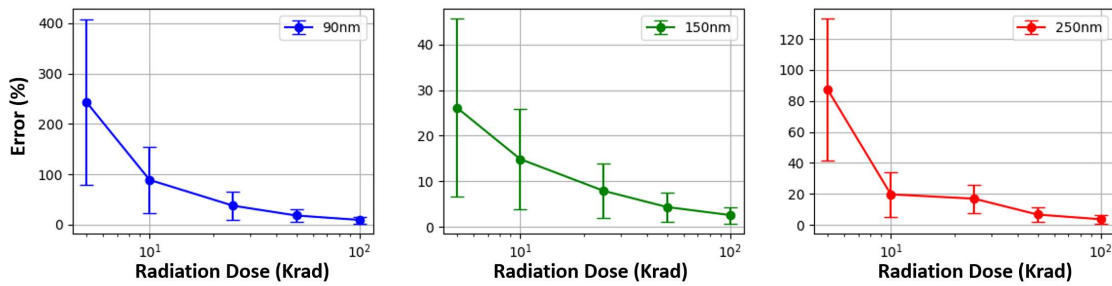
Technology node	Linear equation
90 nm	$y = 0.015x + 3.864$
150 nm	$y = 0.078x + 2.237$
250 nm	$y = 0.069x + 3.336$

radiation response is technology-dependent. Therefore, for accurate prediction of radiation dose using SRAM power-up states, the linear model must be calibrated separately for each specific technology.

7.4.5 Error Percentage Analysis for the SRAM Dosimeter

Figure 7.11 presents the error (%) as a function of radiation dose, ranging from 5 krad(Si) to 100 krad(Si), for all three technology nodes. The error percentage was determined by first estimating the radiation dose using the fitted linear equation applied to the measured HD. For each HD measurement, the estimated dose was then compared with the actual radiation dose to compute the error percentage. The error is calculated as

$$\text{error}(\%) = \frac{(\text{Estimated dose} - \text{Actual dose})}{\text{Actual dose}} \times 100. \quad (7.5)$$

**Figure 7.11:** Error (%) as a function of Total-Ionizing-Dose (TID)

The discrepancy between the estimated and actual doses varies with radiation exposure for all three chips. As the radiation dose increases, the error percentage gradually decreases. Notably, the 150 nm and 250 nm nodes exhibit significantly lower error percentages than the 90 nm node at

higher TID levels. At 100 krad(Si), the error for the 150 nm and 250 nm chips is approximately in the range of 2% to 3%. This indicates that the 150 nm and 250 nm SRAM chips provide higher dose-estimation accuracy, particularly at elevated radiation levels. These results further support the applicability of the SRAM dosimeter for higher TID ranges.

7.4.6 Accuracy Analysis of the SRAM Dosimeter

The results of this study indicate that the SRAM PUS is a promising candidate for dosimetry applications. However, several factors must be investigated further in order to improve the overall accuracy and applicability of the SRAM dosimeter. In the present work, a linear relationship between HD and radiation dose was observed over the medium-dose range. This relationship may not necessarily remain valid at very high dose levels, and additional experiments are required to study the possibility of nonlinear behavior beyond the range examined here. Another factor that can influence dosimeter accuracy is relaxation in the irradiated chips. After irradiation, slight changes in HD may occur over time due to relaxation effects within the SRAM device. Such temporal variation can affect the stability of the measured response and should therefore be characterized in more detail for long-term dosimetry applications.

In addition, this study focused only on Cypress SRAM chips fabricated in 90 nm, 150 nm, and 250 nm technologies. A broader investigation using SRAM chips from other vendors, such as IDT and ISSI, would help determine how general the observed dosimetry behavior is across different commercial memory families. Temperature variation is another important factor that may influence HD measurements. Since environmental temperature can affect the power-up stability of SRAM cells, additional studies are needed to quantify the influence of temperature on the radiation response of SRAM PUFs. Addressing these factors will help further refine the use of SRAM power-up states for dosimetry applications.

7.5 Conclusion

- **Radiation-Induced Changes in SRAM PUS:** In summary, the results of this chapter show that the PUS of SRAM changes significantly under ionizing radiation exposure. The measured HD increases with TID, confirming that the power-up behavior of SRAM is sensitive to radiation-induced device-level changes. All three COTS SRAM chips evaluated in this study, namely Cypress 90 nm, 150 nm, and 250 nm, showed measurable sensitivity even at low radiation dose levels of 0.2 krad(Si). At the same time, all three chips remained operational up to 350 krad(Si), demonstrating that SRAM can continue to function while still exhibiting a clear radiation-dependent response.
- **Feasibility of SRAM PUS as a Dosimeter:** The observed monotonic increase in HD with TID establishes the feasibility of using SRAM PUS as a dosimetry signal. Although there is noticeable variability among individual PUF measurements, the radiation-induced change in HD shows much lower variability, which improves the practicality of this approach for radiation sensing. Among the evaluated devices, the older 250 nm Cypress SRAM chip exhibited greater sensitivity to irradiation at higher dose levels than the 90 nm and 150 nm chips.
- **Linear Dose Response and Dose Estimation Accuracy:** This chapter also showed that the SRAM dosimeter follows a useful linear response over a specific intermediate radiation range from 5 krad(Si) to 100 krad(Si). Within this range, all three chips could be modeled using linear equations, although the fitted parameters differed across technology nodes. The 150 nm and 250 nm chips showed better linearity and lower error percentages than the 90 nm chip, indicating improved dose-estimation accuracy in those technologies. Overall, these results demonstrate that the SRAM PUS can serve as a practical basis for dosimetry applications. The combination of low-dose sensitivity, monotonic response with TID, continued functionality at high total dose, and a measurable linear region supports the use of SRAM as a compact and digitally readable radiation dosimeter.

Chapter 8

Limitations, Conclusion and Future Work

This chapter first discusses the main limitations of this thesis in order to clearly define the scope and boundaries of the presented results. It then summarizes the major contributions of this work and outlines several directions for future research. While the results clearly demonstrate aging-induced data imprinting, radiation-based sanitization, and dosimetry behavior in the evaluated COTS SRAM devices, the scope of the study was intentionally limited to a specific set of memory types, operating conditions, and overwrite patterns. These limitations do not diminish the main findings of this work, but they do define the boundaries within which the conclusions should be applied.

- **Synchronous and Asynchronous SRAM Devices:** The experimental results in this thesis were obtained using asynchronous COTS SRAM chips. Although the observed imprinting behavior is fundamentally tied to device-level transistor aging and SRAM cell mismatch, the present work does not directly establish whether the same imprinting trends will appear with the same magnitude in synchronous SRAM devices. In synchronous SRAMs, the core storage element is still typically based on bistable SRAM cells, but the memory interface, timing control, clocked operation, and peripheral circuitry differ from those of asynchronous devices. These differences may affect practical access patterns, duty cycles, stress conditions, and the way data is repeatedly written and retained over long periods. Therefore, the conclusions of this thesis should not be directly generalized to synchronous SRAM memories without dedicated experimental validation.
- **Limited Validation on FPGA Configuration Memories:** One important security implication of this work is that similar aging-induced effects may be relevant to FPGA platforms, especially because configuration bits are often stored in the same physical memory locations for long periods of time. However, this thesis did not experimentally evaluate on-chip FPGA

configuration SRAM directly. The results were instead obtained from standalone COTS SRAM devices under controlled laboratory conditions. FPGA configuration memories may differ in cell design, process technology, peripheral circuitry, redundancy, error mitigation techniques, refresh or scrubbing strategies, and actual field usage conditions. As a result, while the present findings strongly suggest a possible risk for SRAM-based FPGAs, the extent of that risk remains a limitation of this work and must be confirmed through direct FPGA-focused studies.

- **Thermal Sanitization Constraints:** This work primarily focused on radiation-based sanitization and did not comprehensively evaluate thermal sanitization as a standalone mitigation technique. Although elevated temperature may potentially reduce or modify previously imprinted aging effects, practical exploration of thermal sanitization is limited by the maximum operating and storage temperatures specified in the device data sheets. For many COTS SRAMs, the allowable temperature range is significantly below the levels that would be desirable for aggressive recovery or rebalancing of aging-induced threshold-voltage shifts. Exceeding these limits risks permanent damage, accelerated degradation, packaging stress, or complete device failure. Consequently, the ability of thermal treatment to serve as a safe and reusable sanitization method remains only partially explored within the bounds of this thesis.
- **Sanitization Pattern:** The sanitization experiments in this thesis used a limited set of overwrite patterns, primarily structured and uniform data states, to evaluate the shift in future SRAM PUS after aging and radiation exposure. While these patterns were sufficient to demonstrate the feasibility of imprint manipulation and sanitization, they do not represent the full space of possible data backgrounds. Different sanitization patterns, including random data, checkerboard-like structures, inverted baseline states, or pattern sequences applied over multiple cycles, may interact differently with pre-existing aging asymmetry and radiation-induced threshold-voltage shifts. Because the number of possible spatial and temporal pattern combinations is extremely large, this thesis could not exhaustively determine

which overwrite pattern is optimal for all devices and technologies. Therefore, the reported sanitization effectiveness should be interpreted as pattern-dependent rather than universally pattern-independent.

““

The conclusions are organized according to the four major themes of this work, namely PUS characterization, aging-induced imprinting effects, sanitization of aged SRAM arrays, and dosimetry using SRAM PUS. Together, these results establish both the security implications of long-term SRAM aging and the broader applicability of SRAM power-up behavior in radiation sensing.

- **PUS Characterization:** This thesis established that the natural PUS of SRAM is chip-specific and technology-dependent. The power-up-state images showed that different SRAM chips do not exhibit the same spatial distribution of logic ‘0’ and logic ‘1’, and even devices with similar overall randomness do not produce the same bit locations across the array. These observations confirmed that the natural PUS provides the correct baseline for understanding the effects caused by aging and radiation.
- **Aging-Induced Imprinting Effects:** This thesis demonstrated that long-term thermal and electrical stress can significantly modify the future PUS of SRAM arrays and create measurable data imprinting effects. The results showed pronounced reverse imprinting across multiple devices, with imprint strength depending strongly on stress temperature, aging duration, supply voltage, and post-stress relaxation behavior. The combined evidence from Arrhenius analysis, voltage acceleration, and recovery kinetics identified NBTI-induced PMOS threshold-voltage mismatch as the dominant physical mechanism responsible for the observed aging-induced PUS alteration.
- **Sanitization of Aged SRAM:** This thesis further showed that aged SRAM arrays can be effectively sanitized using radiation-based treatment. By overwriting the imprinted memory locations with a new target pattern and then exposing the chip to ionizing radiation in the powered-on condition, the future PUS was shifted away from the previous aged imprint and

toward the newly written sanitization state. Near-complete sanitization was achieved in both ISSI 180 nm and CY 150 nm devices, although the required TID differed across technology nodes, demonstrating a clear technology dependence in sanitization response.

- **Dosimeter Using SRAM PUS:** This thesis also showed that the PUS of SRAM can be used as a practical dosimetry signal under ionizing radiation exposure. The HD between the post-irradiation PUS and the pre-irradiation reference state increased monotonically with TID, all evaluated COTS SRAM chips remained operational up to 350 krad(Si), and measurable sensitivity was observed even at low dose. In addition, the 150 nm and 250 nm chips showed better linearity and lower error than the 90 nm chip over the useful intermediate dose range, supporting the use of SRAM PUS as a compact and digitally readable radiation dosimeter.

The results of this thesis also open several important directions for future work. These directions can extend the present study toward newer memory technologies, broader sanitization strategies, and deeper device-level understanding of SRAM security behavior.

- **Explore Newer Technology Nodes:** This work mainly investigated older planar CMOS SRAM technologies down to 65 nm. A natural next step is to study more advanced SRAM technologies fabricated in newer technology nodes, including FinFET-based processes and other highly scaled memory architectures. Since modern devices may use different transistor structures, supply schemes, and reliability optimizations, the imprinting rate, persistence, and sanitization response may differ significantly from the trends observed in older nodes. Extending this study to newer technologies would clarify whether aging-induced imprinting remains a practical security concern in advanced commercial memories. It would also help determine whether radiation-based sanitization continues to be effective in more modern SRAM implementations.
- **Study More COTS SRAM Devices:** This thesis focused on a limited set of COTS SRAM chips from Cypress and ISSI. Future work should expand the study to a broader range of vendors and product families, including manufacturers such as Renesas, IDT, and other

commercially available SRAM suppliers. A larger cross-vendor comparison would improve understanding of how proprietary process choices, circuit design techniques, and reliability optimizations influence imprint formation and sanitization behavior. Such a study would also make it possible to identify which classes of COTS SRAM are more vulnerable and which are more naturally resistant to end-of-life data leakage.

- **Explore Other Sanitization Techniques:** Although this thesis demonstrated the effectiveness of gamma-ray-based sanitization, other non-destructive sanitization methods should also be investigated. Examples include X-ray exposure, controlled thermal annealing, and other treatments that can modify device-level threshold-voltage mismatch without immediately destroying functionality. Comparing these methods would help determine tradeoffs among sanitization effectiveness, treatment time, device degradation, and practical deployment cost. Such work would be valuable for identifying the most feasible sanitization strategy for secure disposal or controlled reuse of aged SRAM devices.
- **Study Data Dependencies in Sanitization Patterns:** Another important direction is to evaluate how the choice of sanitization pattern influences the final post-stress PUS. In this thesis, uniform overwrite patterns such as all-zero and all-one states were studied, but other possibilities are also important, including the initial baseline PUS, the inverted baseline PUS, random data, and the inverse of the aged imprint. Different overwrite patterns may interact differently with the existing device-level aging bias and with the radiation-induced threshold-voltage shifts. A systematic study of such data dependencies could lead to more optimized sanitization flows and help identify which target pattern is most effective for suppressing previously imprinted information.
- **Investigate DRV Changes Under Radiation:** Future work should also examine how the data retention voltage (DRV) of SRAM is affected by ionizing radiation and how this interacts with imprinting and sanitization behavior. Since radiation can alter transistor threshold voltages, leakage, and stability margins, it may also shift the minimum voltage at which the

cell can safely retain data. Studying radiation-induced DRV variation across different technologies could provide deeper insight into the relationship between cell stability, power-up bias, and long-term sanitization effectiveness. Such an analysis would also strengthen the connection between reliability metrics and security-oriented SRAM behavior in aged and irradiated devices.

Bibliography

- [1] K. Zhang, U. Bhattacharya, Z. Chen, F. Hamzaoglu, D. Murray, N. Vallepalli, Y. Wang, B. Zheng, and M. Bohr. SRAM design on 65-nm CMOS technology with dynamic sleep transistor for leakage reduction. *IEEE Journal of Solid-State Circuits*, 40(4):895–900, 2005.
- [2] Benton H. Calhoun and Anantha P. Chandrakasan. A 256-kb 65-nm sub-threshold SRAM design for ultra-low-voltage operation. *IEEE Journal of Solid-State Circuits*, 42(3):680–688, 2007.
- [3] Koichi Takeda, Yasuhiko Hagihara, Yoshiharu Aimoto, Masahiro Nomura, Yoetsu Nakazawa, Toshio Ishii, and Hiroyuki Kobatake. A read-static-noise-margin-free SRAM cell for low-VDD and high-speed applications. *IEEE Journal of Solid-State Circuits*, 41(1):113–121, 2006.
- [4] Jaydeep P. Kulkarni, Keejong Kim, and Kaushik Roy. A 160 mv robust schmitt trigger based subthreshold SRAM. *IEEE Journal of Solid-State Circuits*, 42(10):2303–2313, 2007.
- [5] Ramy E. Aly and Magdy Bayoumi. Low-power cache design using 7T SRAM cell. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 54(4):318–322, 2007.
- [6] Christoph Böhm, Maximilian Hofer, and Wolfgang Pribyl. A microcontroller SRAM-PUF. In *2011 5th International Conference on Network and System Security*, pages 269–273, 2011.
- [7] Mafalda Cortez, Apurva Dargar, Said Hamdioui, and Geert-Jan Schrijen. Modeling SRAM start-up behavior for physical unclonable functions. In *2012 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, pages 1–6, October 2012.
- [8] Sujay Pandey, Sabyasachi Deyati, Adit Singh, and Abhijit Chatterjee. Noise-resilient SRAM physically unclonable function design for security. In *2016 IEEE 25th Asian Test Symposium (ATS)*, pages 55–60, 2016.

- [9] Umeshwarnath Surendranathan, Horace Wilson, Lei R. Cao, Aleksandar Milenkovic, and Biswajit Ray. Analysis of SRAM PUF integrity under ionizing radiation: Effects of stored data and technology node. *IEEE Transactions on Nuclear Science*, 71(4):485–491, 2024.
- [10] J. Hovanes, Y. Zhong, and U. Guin. Beware of discarding used SRAMs: Information is stored permanently. In *2022 IEEE Physical Assurance and Inspection of Electronics (PAINE)*, pages 1–7, October 2022.
- [11] U. Surendranathan, A. S. Vellankanni, A. Milenkovic, U. Guin, and B. Ray. Ionizing radiation-induced data imprinting effects in SRAM arrays. *IEEE Transactions on Nuclear Science*, 72(4):1252–1258, April 2025.
- [12] J. Mahmood and M. Hicks. UnTrustZone: Systematic accelerated aging to expose on-chip secrets. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 4107–4124, May 2024.
- [13] U. Guin, W. Wang, C. Harper, and A. D. Singh. Detecting recycled SoCs by exploiting aging induced biases in memory cells. In *2019 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 72–80, May 2019.
- [14] G. Odom, Z. T. Tisha, and U. Guin. A novel self-referencing approach using memory power-up states for detecting COTS SRAMs. In *2024 IEEE 42nd VLSI Test Symposium (VTS)*, pages 1–7, April 2024.
- [15] Umeshwarnath Surendranathan, Horace Wilson, Maryla Wasiolek, Khalid Hattar, Aleksandar Milenkovic, and Biswajit Ray. Total ionizing dose effects on the power-up state of static random-access memory. *IEEE Transactions on Nuclear Science*, 70(4):641–647, 2023.
- [16] Neil H. E. Weste and David Money Harris. *CMOS VLSI Design: A Circuits and Systems Perspective*. Addison-Wesley, 4 edition, 2011.
- [17] E. Seevinck, F. J. List, and J. Lohstroh. Static-noise margin analysis of MOS SRAM cells. *IEEE Journal of Solid-State Circuits*, 22(5):748–754, 1987.

- [18] Jan M. Rabaey, Anantha Chandrakasan, and Borivoje Nikolić. *Digital Integrated Circuits: A Design Perspective*. Prentice Hall, 2 edition, 2003.
- [19] Sung-Mo Kang and Yusuf Leblebici. *CMOS Digital Integrated Circuits: Analysis and Design*. McGraw-Hill, 3 edition, 2003.
- [20] R. Jacob Baker. *CMOS: Circuit Design, Layout, and Simulation*. Wiley-IEEE Press, 3 edition, 2010.
- [21] Betty Prince. *Semiconductor Memories: A Handbook of Design, Manufacture and Application*. John Wiley & Sons, 2 edition, 1996.
- [22] Xiaolu Wang, John Keane, Tae-Hyoung Kim, Pranita Jain, Qiang Tang, and Chris H. Kim. Silicon odometers: Compact in situ aging sensors for robust system design. *IEEE Micro*, 34(6):74–85, 2014.
- [23] Aditya Bansal, Rahul Rao, Jae-Joon Kim, Zafar Sufi, James H. Stathis, and Ching-Te Chuang. Impacts of NBTI and PBTI on SRAM static/dynamic noise margins and cell failure probability. *Microelectronics Reliability*, 49(6):642–649, 2009.
- [24] Sanjay V. Kumar, Chris H. Kim, and Sachin S. Sapatnekar. Impact of NBTI on SRAM read stability and design for reliability. In *Proceedings of the 7th International Symposium on Quality Electronic Design (ISQED)*, pages 210–218, 2006.
- [25] D. K. Schroder. Negative bias temperature instability: What do we understand? *Microelectronics Reliability*, 47(6):841–852, June 2007.
- [26] Santanu Mahapatra, N. Goel, Shaishav Desai, Souvik Gupta, B. Jose, G. Gupta, K. Joshi, A. Alam, K. Roy, and S. Mukhopadhyay. A comparative study of different physics-based NBTI models. *IEEE Transactions on Electron Devices*, 60(3):901–916, 2013.
- [27] J. H. Stathis, M. Wang, and K. Zhao. Reliability of advanced high-*k*/metal-gate n-FET devices. *Microelectronics Reliability*, 50(9–11):1199–1202, 2010.

- [28] Subhadeep Mukhopadhyay, N. Goel, and Souvik Mahapatra. A comparative study of NBTI and PBTI using different experimental techniques. *IEEE Transactions on Electron Devices*, 63(8):3163–3169, 2016.
- [29] Ping Huang and Zuo cheng Xing. Impacts of NBTI/PBTI on power gated SRAM. *Journal of Central South University*, 20:1298–1306, 2013.
- [30] A. Garg, Z. C. Lee, L. Lu, and T. T.-H. Kim. Improving uniformity and reliability of SRAM PUFs utilizing device aging phenomenon for unique identifier generation. *Microelectronics Journal*, 90:29–38, August 2019.
- [31] J. Mahmod and M. Hicks. SRAM has no chill: Exploiting power domain separation to steal on-chip secrets. In *Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS '22*, pages 1043–1055, New York, NY, USA, February 2022. Association for Computing Machinery.
- [32] J. Mahmod and M. Hicks. SRAM imprinting for system protection and differentiation. In *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security, ASIA CCS '24*, pages 453–466, New York, NY, USA, July 2024. Association for Computing Machinery.
- [33] Z. T. Tisha, G. Odom, B. Ray, and U. Guin. Decoding the past: Recovering sensitive data from SRAM aging imprints. Presented at the Design Automation Conference, 2026.
- [34] Z. T. Tisha, G. Odom, B. Ray, and U. Guin. Unlocking hidden secrets: Leveraging SRAM aging imprints for sensitive data recovery. Presented at the Design, Automation and Test in Europe Conference, pp. 1–3, 2026.
- [35] Z. T. Tisha, G. Odom, B. Ray, and U. Guin. Recovering the unstored: Exploiting SRAM aging for data retrieval in recycled devices. Presented at the Government Microcircuit Applications & Critical Technology Conference, pp. 1–6, 2026.

- [36] J. Cui, Q. Zheng, Y. Li, and Q. Guo. Impact of high TID irradiation on stability of 65 nm SRAM cells. *IEEE Transactions on Nuclear Science*, 69(5):1044–1050, May 2022.
- [37] N. J. Pieper, Y. Xiong, J. Pasternak, D. R. Ball, and B. L. Bhuva. Effects of TID on SRAM data retention stability at the 5-nm node. *IEEE Transactions on Nuclear Science*, 71(8):1864–1871, August 2024.
- [38] J. Kim, J. Lee, and J. A. Abraham. Toward reliable sram-based device identification. In *2010 IEEE International Conference on Computer Design*, pages 313–320, 2010.
- [39] R. Maes, V. van der Leest, G.-J. Schrijen, P. Tuyls, and H. Handschuh. Experimental evaluation of physically unclonable functions in 65 nm cmos. In *2012 Proceedings of the ESSCIRC (ESSCIRC)*, pages 486–489, 2012.
- [40] A. T. Elshafey, P. Zarkesh-Ha, and J. Trujillo. The effect of power supply ramp time on sram pufs. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 946–949, 2017.
- [41] Sayan Samanta, Biswajit Ray, and Aleksandar Milenkovic. Sram dna: Spatial signatures in power-up states for memory family identification. In *Proceedings of the Great Lakes Symposium on VLSI 2026, GLSVLSI '26*, pages 701–708, New York, NY, USA, 2026. Association for Computing Machinery.
- [42] K. Xiao, M. T. Rahman, D. Forte, Y. Huang, M. Su, and M. Tehranipoor. Bit selection algorithm suitable for high-volume production of sram-puf. In *2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pages 101–106, 2014.
- [43] Y. Shifman, A. Miller, O. Keren, Y. Weizmann, and J. Shor. A method to improve reliability in a 65-nm sram puf array. *IEEE Solid-State Circuits Letters*, 1(6):138–141, 2018.
- [44] Sadman Sakib, Aleksandar Milenković, and Biswajit Ray. Flash watermark: An anti-counterfeiting technique for NAND flash memories. *IEEE Transactions on Electron Devices*, 67(10):4172–4177, 2020.

- [45] W.-G. Ho, Z. Zheng, K.-S. Chong, and B.-H. Gwee. A comparative analysis of 65nm CMOS SRAM and commercial SRAMs in security vulnerability evaluation. In *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, pages 1–5, November 2018.
- [46] F. Hoque, I. Halseide, A. Milenkovic, and B. Ray. Data remanence vulnerabilities in commercial sram at low temperature. In *2024 IEEE Physical Assurance and Inspection of Electronics (PAINE)*, pages 1–7, Huntsville, AL, USA, 2024.
- [47] U. Surendranathan, F. Hoque, A. Milenkovic, and B. Ray. Technology scaling effects on SRAM data remanence. In *2024 Device Research Conference (DRC)*, pages 1–2, College Park, MD, USA, 2024.
- [48] V. Reddy et al. Impact of negative bias temperature instability on digital circuit reliability. *Microelectronics Reliability*, 45(1):31–38, January 2005.
- [49] D. K. Schroder and J. A. Babcock. Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing. *Journal of Applied Physics*, 94(1):1–18, July 2003.
- [50] Abishek Stephen Vellankanni, Zakia Tamanna Tisha, Nayan Patel, Ujjwal Guin, and Biswajit Ray. Aging induced data imprinting effects on sram array. In *2026 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2026.
- [51] M. A. Alam and S. Mahapatra. A comprehensive model of PMOS NBTI degradation. *Microelectronics Reliability*, 45(1):71–81, January 2005.
- [52] S. Mahapatra and M. A. Alam. A predictive reliability model for PMOS bias temperature degradation. In *Digest. International Electron Devices Meeting*, pages 505–508, December 2002.
- [53] M. Denais, V. Huard, C. Parthasarathy, A. Bravaix, E. Vincent, G. Ribes, N. Revil, J.-M. Roule, S. Bruyere, and G. Ghibaudo. Interface trap generation and hole trapping under NBTI

- and PBTI in advanced CMOS technology with a 2-nm gate oxide. *IEEE Transactions on Device and Materials Reliability*, 4(4):715–722, December 2004.
- [54] G. Fukano et al. A 65nm 1mb SRAM macro with dynamic voltage scaling in dual power supply scheme for low power SoCs. In *2008 Joint Non-Volatile Semiconductor Memory Workshop and International Conference on Memory Technology and Design*, pages 97–98, May 2008.
- [55] U.S. Nuclear Regulatory Commission. PART 20—Standards for Protection Against Radiation. NRC. Referenced in the dosimeter paper introduction.
- [56] F. A. Mettler, W. Huda, T. T. Yoshizumi, and M. Mahesh. Effective doses in radiology and diagnostic nuclear medicine: A catalog. *Radiology*, 248(1):254–263, July 2008.
- [57] E. Wagner, R. Sorom, and L. Wiles. Radiation monitoring for the masses. *Health Physics*, 110(1):37–44, January 2016.
- [58] E. B. Podgorsak. Radiation oncology physics, July 2005. Referenced in the dosimeter paper introduction.
- [59] B. Chatterjee, C. Mousoulis, D. Seo, A. Kumar, S. Maity, S. M. Scott, D. J. Valentino, D. T. Morissette, D. Peroulis, and S. Sen. A wearable real-time CMOS dosimeter with integrated zero-bias floating-gate sensor and an 861nm 18-bit energy-resolution scalable time-based radiation to digital converter. In *2019 IEEE Custom Integrated Circuits Conference (CICC)*, pages 1–4, April 2019.
- [60] E. Tochilin, N. Goldstein, and W. G. Miller. Beryllium oxide as a thermoluminescent dosimeter. *Health Physics*, 16(1):1, January 1969.
- [61] A. Jaksic, G. Ristic, M. Pejovic, A. Mohammadzadeh, C. Sudre, and W. Lane. Gamma-ray irradiation and post-irradiation responses of high dose range RADFETs. In *RADECS 2001. 2001 6th European Conference on Radiation and Its Effects on Components and Systems*, Grenoble, France, 2001.

- [62] M. Brucoli, S. Danzeca, M. Brugger, A. Masi, A. Pineda, J. Cesari, L. Dusseau, and F. Wrobel. Floating gate dosimeter suitability for accelerator-like environments. *IEEE Transactions on Nuclear Science*, 64(8):2054–2060, August 2017.
- [63] N. G. Tarr, G. F. Mackay, K. Shortt, and I. Thomson. A floating gate MOSFET dosimeter requiring no external bias supply. In *RADECS 97. Fourth European Conference on Radiation and Its Effects on Components and Systems*, pages 277–281, September 1997.
- [64] P. Kumari, L. Davies, N. P. Bhat, E. Zhang, M. W. McCurdy, D. M. Fleetwood, and B. Ray. State-of-the-art flash chips for dosimetry applications. In *2018 IEEE Radiation Effects Data Workshop (REDW)*, pages 1–4, July 2018.
- [65] P. Kumari, U. Surendranathan, M. Wasiolek, K. Hattar, N. Bhat, and B. Ray. Analytical bit-error model of NAND flash memories for dosimetry application. *IEEE Transactions on Nuclear Science*, 69(3):478–484, March 2022.
- [66] D. Makowski, M. Grecki, B. Swiercz, A. Napieralski, B. Mukherjee, and S. Simrock. SRAM-based passive dosimeter for high-energy accelerator environments. In *Proceedings of DIPAC*, 2005. Referenced in the dosimeter paper introduction.
- [67] G. Korkian, Daniel León, Francisco J. Franco, et al. Single event upsets under proton, thermal, and fast neutron irradiation in emerging nonvolatile memories. *IEEE Access*, 10:114566–114585, 2022.
- [68] S. Woo, R. Yoon, D. Bae, K. Kim, H. Lee, S. Chung, C. Cho, J. Kim, S. A. Wender, and Y. Kim. Analysis of efficiency and utilization with SRAM dosimetry for single-event effect evaluation under irradiation. In *2022 22nd European Conference on Radiation and Its Effects on Components and Systems (RADECS)*, pages 1–6, October 2022.
- [69] D. M. Fleetwood. Total-ionizing-dose effects, border traps, and 1/f noise in emerging MOS technologies. *IEEE Transactions on Nuclear Science*, 67(7):1216–1240, July 2020.

[70] V. Danchenko. Radiation damage in MOS integrated circuits, part i, September 1971.

Appendix A

Algorithm for SRAM Interfacing

This appendix summarizes the overall firmware workflow used to interface the external SRAM chips with the Arduino Due and to perform the experiments presented in this thesis. Instead of including the complete source code, the main control logic is presented in algorithmic form for clarity and reproducibility. The firmware consists of three main components: initialization and menu control, serial command processing, and low-level SRAM read/write operations.

A.1 Overall Firmware Flow

Algorithm 5 Overall Arduino Due Firmware Flow

```
1: Start program
2: Initialize SerialUSB at 115200 baud
3: Configure SRAM address, data, and control pins
4: Set SRAM control signals to default inactive state
5: Print command menu to serial terminal
6: while true do
7:   if serial input is available then
8:     Read input command
9:     Dispatch command to the corresponding SRAM routine
10:  else
11:    Continue waiting for user input
12:  end if
13: end while
```

A.2 Serial Command Processing

Algorithm 6 Serial Command Dispatcher

```
1: procedure PROCESSCOMMAND(cmd)
2:   if cmd = dump memory then
3:     Call DUMPSRAM()
4:   else if cmd = write all zeros then
5:     Call WRITEPATTERN(0x0000)
6:   else if cmd = write all ones then
7:     Call WRITEPATTERN(0xFFFF)
8:   else if cmd = write alternating pattern then
9:     Call WRITEPATTERN(0x5555)
10:  else if cmd = write image pattern then
11:    Call WRITEIMAGEDATA()
12:  else if cmd = repeated PUF measurement then
13:    Call MEASUREPUFREPEATEDLY()
14:  else if cmd = enable chip then
15:    Assert chip-enable signal
16:  else if cmd = disable chip then
17:    Deassert chip-enable signal
18:  else
19:    Print invalid-command message
20:  end if
21:  Print command menu again
22: end procedure
```

A.3 Low-Level SRAM Read and Write Operations

Algorithm 7 SRAM Read Operation

```
1: function READSRAM(address)
2:   Configure data bus pins as input
3:   Place address on address bus
4:   Assert output-enable and chip-enable
5:   Wait for data bus to settle
6:   Read all SRAM data pins
7:   Combine bits into one 16-bit word
8:   Deassert output-enable
9:   return data word
10: end function
```

Algorithm 8 SRAM Write Operation

```
1: function WRITESRAM(address, data)
2:   Configure data bus pins as output
3:   Place address on address bus
4:   Place data on data bus
5:   Deassert output-enable
6:   Assert chip-enable
7:   Pulse write-enable low
8:   Wait for write cycle to complete
9:   Deassert write-enable
10: end function
```

A.4 Pattern Writing and Full Memory Dump

Algorithm 9 Write Fixed Pattern to Entire SRAM

```
1: procedure WRITEPATTERN(pattern)
2:   for address = 0 to last SRAM address do
3:     WRITESRAM(address, pattern)
4:   end for
5:   Print completion message
6: end procedure
```

Algorithm 10 Write Image Data to SRAM

```
1: procedure WRITEIMAGEDATA
2:   Load predefined image-derived data array
3:   for address = 0 to image data length - 1 do
4:     Fetch next 16-bit image word
5:     WRITESRAM(address, image word)
6:   end for
7:   Print completion message
8: end procedure
```

Algorithm 11 Dump Full SRAM Contents

```
1: procedure DUMPSRAM
2:   Print start-of-dump marker
3:   for address = 0 to last SRAM address do
4:     data  $\leftarrow$  READSRAM(address)
5:     Send data through serial interface
6:   end for
7:   Print end-of-dump marker
8: end procedure
```

A.5 Repeated Power-Up-State Measurement

Algorithm 12 Repeated SRAM Power-Up-State Measurement

```
1: procedure MEASUREPUFREPEATEDLY
2:   for trial = 1 to number of required trials do
3:     Turn SRAM power OFF
4:     Wait for predefined off-time
5:     Turn SRAM power ON
6:     Restore default SRAM control signals
7:     for address = 0 to last SRAM address do
8:       data  $\leftarrow$  READSRAM(address)
9:       Send data through serial interface
10:    end for
11:    Print end-of-trial marker
12:  end for
13: end procedure
```
