Exploiting Power-Up State of Latches as Hardware Security Primitives: PUF, TRNG and Recycled IC Detection

by

Wendong Wang

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
May 7, 2022

Keywords: Hardware Security, PUF, TRNG, Power-up state, SRAM, Recycling ICs.

Approved by

Adit D. Singh, Godbold Chair Professor, Electrical and Computer Engineering
Ujjwal Guin, Chair, Assistant Professor, Electrical and Computer Engineering
Vishwani D. Agrawal, Professor Emeritus, Electrical and Computer Engineering
Mehdi Sadi, Assistant Professor, Electrical and Computer Engineering
Sanjeev Baskiyar, Reader, Professor, Computer Science and Software Eng.

Abstract

Over the last two decades, hardware security has become more and more important and became a hot topic in both academia and industry. Threats seen on cyber-infrastructure and electronic devices are becoming more and more advanced each year, resulting in huge corresponding economy and resource loss. Furthermore, in a world that relies heavily on technology to function on a daily basis, individuals should be able to trust that it is safe to use and that our information will be protected [1]. Whether it be your car, laptop, phone, or smart thermostat, we often utilize these smart devices to help us in our daily life. However, we barely give much thought as to how the information collected could be used against us if it falls into the wrong hands. The reality is that the things such as identity theft based on a computer virus, attacks via weaknesses in device security, are becoming more and more prevalent today. This puts a big responsibility on the designers of such devices to ensure that information stays secure, and this is where the constantly evolving field of hardware security comes into play.

Memory such as SRAM has become the hot spotlight due to its possible hardware security application. Due to the unavoidable process variation during the manufacturing process, each SRAM cell will obtain a unique electrical character and therefore, have been proposed as a source for hardware security primitives such as physical unclonable function (PUF), true random number generator(TRNG) and recycling integrated circuit detection.

The SRAM-based PUFs is one of the appealing PUFs candidates because of the easy implementation and nearly zero hardware penalty [2]. Holcombet al. proposed the very first idea of using the power-up value of SRAM array as the response of PUFs [3]. However, part of SRAM cells will not have consistent value under multiple powering up. This instability in the SRAM PUF response over the expected range of operating voltages and temperature, as well as environmental noise and aging degradation over time, is a challenge. Recent proposals aim at identifying a subset of all the cells in an SRAM, the most robust or “strong” cells, and using only these to construct a PUF [4, 5]. However, it requires modification of the original SRAM design, which will increase the overall design cost and limit the range of application. In this dissertation, we will propose a systematic bit selection method to identify those most ”strong” cells for PUF application. Furthermore, the proposed method will not require any hardware-level modification of the original design of SRAM. The silicon data indicates that the selected cell can reach 100% reliability under multiple powering up and temperature/voltage variation.

Besides the SRAM-based PUF, the power-up value of an SRAM array has also been widely accepted as an entropy source for generating random numbers. However, only a few cells of the SRAM are truly random upon repeated power-ups; the vast majority of cells display a distinct bias due to manufacturing process variations. Consequently, a relatively large SRAM array is required to obtain sufficient entropy for generating random numbers. To improve the overall entropy of an SRAM array and avoid large SRAM array being the entropy source, researchers have proposed the use of controlled device aging at the pre-deployment stage to enhance the initial entropy of an SRAM array. However, aging in the field can adversely affect entropy and degrade randomness. In this dissertation, we exhibit how the initially controlled aging to increase SRAM entropy will have a counter effect when the SRAM array has been used in the field. To cope with this, we proposed an SRAM-based TRNG, that relies on periodically powering up and down the device during deployment to maintain/maximize the entropy of the entire SRAM array. The key idea is to continuously stress the SRAM cells in their power-up states at regular intervals. This helps to counter the aging caused by the random memory states that occur during operation. Silicon results are presented to validate our proposed approach.

The power-up value of the SRAM array also can be used for recycled ICs detection. The recycled ICs sold as new parts pose a serious security problem due to the shorter lifetime, potentially poorer performance compared to fresh and authentic parts. In this dissertation, we propose a novel method to identify the recycled ICs based on the power-up state of the SRAM array. The proposed method will not require any pre-historic information from those obsolete chips and only requires the ability to read out the power-up value of the SRAM array. Our methodology is based on a basic observation - an unused SRAM will always obtain a virtually equal number of logical 1 and logical 0 during power-up. Afterward, the ratio of logical 1 and logical 0 will be skewed due to biased aging during normal in-field operation. In this dissertation, we will discuss two different

application scenarios. First, where SRAM arrays have been manufactured in advanced technology. Second, where the SRAM chip is fabricated by very mature/old technology (in um level). For both scenarios, we will propose our method to detect the recycled SRAM based on the power-up state. Our silicon data validate the efficiency of our proposed method in the real world.

Although the method to detect recycled ICs based on the power-up state of the SRAM array is very effective, its applicability is somewhat limited as a large number of older designs do not have large on-chip memories. To be effective in such scenarios, we also propose an alternate detection approach based on the initial power-up state of scan flip-flops, which are present in virtually every digital circuit. Since the flip-flops, unlike SRAM cells, are generally not perfectly symmetrical in layout, an equal number of scan cells will not power up to 0 or 1 logic states in most designs. Consequently, a stable time zero reference of 50% logic 0s and 1s cannot be used for determining the subsequent usage of a chip. To overcome this key limitation, we first identify a significant number of asymmetrically stressed flip-flops in the design, divided into two groups. One group of flip-flops is selected such that it mostly experiences the 1 logic state during functional operation, while the other group mostly experiences the 0 states. The resulting differential stress during operation causes growing disparity over time in the number of 0s (and 1s) observed in these two groups on power-up. When the chip is new and has not experienced aging, these two groups behave similarly, with a similar percentage of 1s (or 0s). However, over time the differential stress makes these counts diverge. We show that this changing count can be a measure of operational aging. Our simulation results show that it is possible to reliably detect used ICs after as little as three months of operation.

SRAM/DFF has the potential to be one of the hardware security primitive. However, many challenges exist before the SRAM-based hardware security solution. In this dissertation, we proposed a new systematic bit selection method to reach 100% reliability for SRAM-based PUF applications. For the SRAM-based TRNG, we can maintain/maximize the entropy of the SRAM array through our proposed aging strategy. For the recycled ICs detection, to the best of our knowledge, it is the first time that a piratical method has been proposed to identify the recycled ICs based on the power-up state of SRAM without requiring any pre-historic information from obsolete ICs. We also proposed a novel method to detect recycled ICs by exploiting the power-up state of DFF.

Acknowledgments

I would like to express my heartfelt gratitude to Dr. Adit Singh, my graduate advisor, for his encouragement and guidance during my time at Auburn University. His encouragement and guidance paved the way for my successful research projects and thesis completion. I would like to express my gratitude to Dr. Ujjwal Guin for his experience and insight into research. This dissertation would not have been possible without his constant support and guidance. Also, special thanks to Dr. Vishwani Agrawal and Dr. Mehdi Sadi, for their great help and efforts. I would also like to thank Dr. Baskiyar for being my university reader providing me with valuable comments and kind support.

I want to extend my gratitude to the committee members again for their time, support, and advice toward my research and thesis preparation. Thanks to all of my labmates and colleagues for the valuable information I acquired during my course and study work. The lab exercises and brainstorming sessions taught me a lot about my field of study, and I owe much gratitude to them. Finally, I would like to thank my parents and friends for their unwavering support during my academic career.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CPS	Cyber-Physical Systems
DfAC	design for anti-counterfeit
DFF	D flip-flop
DoD	U.S. Department of Defense
ECC	Error correction code
ICs	Integrated Circuits
OCMs	original component manufactures
PUF	Physical Unclonable Function
RoT	Root of Trust
SRAM	Static random-access memory
TRNG	True Random Number Generator

Chapter 1

Introduction

With the advancement of ubiquitous distributed computing under the broad umbrella of the Internet of Things (IoT), the number of connected devices (edge devices) is expected to grow exponentially in the coming decades. Wide use of these edge devices, which can be expected to include a large share of Cyber-Physical Systems (CPS), in critical applications (smart home, smart city, automated transportation) will present unique security challenges. Due to the resource constraints of these low cost and low power hardware, the majority of edge devices today are unable to use standard cryptographic protocols to communicate securely. Trappe et al. have shown how the power constraint in IoT edge devices limits the encryption functionality of the sensor nodes, which leads to poorly encrypted communication, or often no encryption at all [9]. In particular, the inability to incorporate cryptographic primitives in low cost IoT edge devices results in significant vulnerability in their communications [10, 11]. Consequently, reliable authentication of the devices and securely controlling access to them, maintaining confidentiality for all communications, and also ensuring data integrity, all become very challenging.

Clearly, in order to facilitate large scale adoption of the diverse IoT/CPS applications, the IoT devices must be of low-cost. This requires that any added security features should reuse available hardware resources to the greatest extent possible. It has been widely recognized that the power-up state of SRAM can be exploited to hardware based security primitives, such as PUF , TRNG and recycled ICs detection. Intrinsic implementations (SRAM), which do not require extra hardware components, have been proposed as a lightweight and cost-efficient basis for security solutions.

1.1 Motivation

Physically unclonable functions (PUFs) is die-specific random function, which can generate unique response under given inputs (challenges). It takes advantage of the unavoidable small random manufacturing variations in highly scaled integrated circuits (ICs) for different cryptographic applications [12]. SRAM is a widespread building block in FPGA and many System-on-chip (SoC), which make the implementation of SRAM PUF simple and require no further design process. These exploit the power-up value of cells to provide the PUF response, with the corresponding address serving as the challenge. Although the SRAM PUF provides many attractive properties, one of the challenges is the reliability of SRAM cells. Firstly, the response of part of SRAM cells will be very sensitive to the noise and temperature/voltage variation. Some researchers have reported that about 5-10 percent of cells will not obtain the same value during multiple power-up(s). Secondly, the aging degradation will affect the threshold voltage of MOSFETs and twist the strength of two back-to-back inverters. Lastly, the response of SRAM cells will possibly be flipped during operation in the field. In order to exploit the start-up value of SRAM cells to perform some cryptography operations (i.e. key generation), the response/start-up value for each SRAM cell ideally should be reliable under different operating conditions (i.e. different temperature or voltage). Several mechanisms have been proposed to reduce a flipping bit and increase the overall cell reliability [2, 4, 5, 13–15, 15–18]. However, these proposed methods will introduce significant hardware overhead and require the inner modification of traditional SRAM array structure. In this dissertation, we will propose a testing method that facilitates us to select the most reliable SRAM cells for PUF application. The proposed method will not require any modification from the traditional SRAM structure and does not need ECC as well, which can incur significant hardware overhead. Our silicon result indicates our proposed method can achieve 100% reliability for PUF applications.

Researchers have developed TRNG structures utilizing the power-up state of SRAM cells as an entropy source [19–21]. However, limited entropy from an SRAM array poses a potential risk for any SRAM-based TRNG as a majority of cells are stable at either 0 or 1. Herrewege et al. presented a PRNG structure based on the power-up state of an SRAM array by implementing it through the commercial off-the-self microcontrollers [22]. The author showed that the SRAM present in these microchips (e.g. PIC16F1825) cannot securely seed the PRNG. Researchers have also shown that the entropy of an SRAM array decreases with the decrease in temperature [22, 23]. This can lead to freezing attacks, where an attacker can expose an SRAM chip to lower temperatures. These prior works indicate that entropy of the power-up state of SRAM is a very crucial parameter for creating a secure PRNG. Consequently, researchers have introduced new methods to improve and preserve the entropy of the SRAM array. The authors in [24, 25] proposed a minor modification to the conventional SRAM array to improve the instability of SRAM cells. Kiamehr et al. exploited device aging to improve the initial entropy of the SRAM array at deployment [26]. In this dissertation, we will prove that initial enhancements with respect to SRAM entropy through controlled aging will pose a potential security risk after SRAM has been used in the field. Instead, we propose an advanced strategy, where we perform periodic controlled aging during operation. This strategy will maintain the entire entropy of the SRAM array during its operation life.

Besides the PUF and TRNG, recycled ICs detection is another main topic in this dissertation. The rise of recycled ICs being sold as new through the global semiconductor supply chain is a serious threat due to their inferior quality, shorter remaining life, and potentially poorer performance, compared to their authentic counterparts [27–29]. Solutions such as on-chip age monitors, have been proposed for new designs. However, there is no reliable solution that helps in detecting the recycling of older legacy ICs already in use. Some researchers have proposed test methods based on statistical data analysis to identify recycled parts [30–35]. However, to establish the statistical data analysis model, a significant number of fresh chips for testing are required, which is very difficult to obtain for those obsolete chips. In this dissertation, we propose two novel testing methods for recycled IC detection: One exploits the power-up state of SRAM, and the other exploits the power-up state of DFF . Our proposed methods do not require any hardware modification to the existing design and can be applied to a wide variety of SoCs that have SRAM-based memory, including FPGAs. This solution can be applied to both, ICs already circulating in the market, as well as those that will be manufactured in the future. The proposed approach is simple, effective, and low-cost, and requires minimal test support: a capability to read out the initial power-up state of the on-chip SRAM and DFF.

1.2 Contributions

- A novel systematic testing method has been proposed for 100% reliable SRAM PUF. The testing method will allow us to identify the most ’strong’ cells for PUF application and the selected cells will be reliable under different operating conditions, such as temperature and voltage. Moreover, this method will not require any modification of SRAM array structure and any heavy overhead component, such as ECC.
- For SRAM-based TRNG, firstly we demonstrate an initial enhancement of the entropy of an SRAM array through controlled and selective aging of devices prior to deployment (as proposed, for example, in [26]) is not retained for very long during actual use. This initial enhancement of the entropy may even have an adverse effect on security if relied upon at design time. Secondly, a developed strategy based on controlled periodic aging during operation has been proposed. This prevents any significant degradation of entropy in the SRAM throughout its operational life and thereby ensures that the RNG always satisfies its randomness specifications.
- For recycled IC detection, we proposed the first reliable electrical test method for aging and recycled IC detection in SRAMs by exploiting the power-up state of SRAM cells. The proposed method will not require any prehistory information from those recycled SRAMs and only need the ability to read out the power-up state of SRAMs.
- Besides the recycled IC detection using power-up states of SRAMs, we have also proposed a novel method for recycled IC detection by exploiting the aging effect in DFF. Again, the proposed method will not require any pre-historic information for the recycled ICs, which allows us can detect the recycled IC that are already circulating in the market.

1.3 Organization of Dissertation

The rest of this dissertation is organized as follows:

- Chapter 2 will introduce some background knowledge regarding the PUF and the reason that SRAM provides an attractive design option for PUF. Additionally, the TRNG and SRAM-based TRNG will also be included. Finally, we will discuss the threat of recycled ICs and the corresponding challenge.

- Chapter 3 will introduce all the necessary and crucial details about the power-up state of a simple latch. Note the latch is the core structure either in SRAM cells or DFF. In this chapter, the power-up state of a latch will be modeled firstly. Then, the ramp rate and aging effect have been included in the circuit model and the corresponding impact has been discussed in detail. This chapter is the theory base of the entire dissertation and the concepts will be repeatedly referred to in the following chapters.
- Chapter 4 and chapter 5 will discuss latch-based hardware security primitives. Chapter 4 will be the SRAM-based PUF. In this part, a novel systemic cell selection method will be introduced. The corresponding simulation and silicon results supporting the claim that the selected SRAM cells will be 100% reliable for PUF application will also be provided. Chapter 5 will be the SRAM-based TRNG. In this part, a novel periodic controlled aging strategy has been proposed. The proposed strategy will maintain the entropy of the SRAM array during the entire normal operation.
- Chapter 6 will be SRAM based recycled ICs detection. In this part, we propose a novel reliable electrical test for recycled ICs detection by using power-up states of SRAM. The proposed method will require no pre-historic information from obsolete chips and can be applied to any chip that has been already circulated in the market. Chapter 7 will be DFF-based recycled ICs detection. In this part, a novel recycled ICs detection based on the power-up state of DFF has been proposed. Further, we will provide corresponding simulation results to prove that the proposed method can be used to detect recycled ICs efficiently.
- Chapter 8 is the conclusion of the entire dissertation. In this chapter, we will conclude all the important concepts and contributions of this dissertation and list the possible directions for future research.

Chapter 2

Background and Prior Work

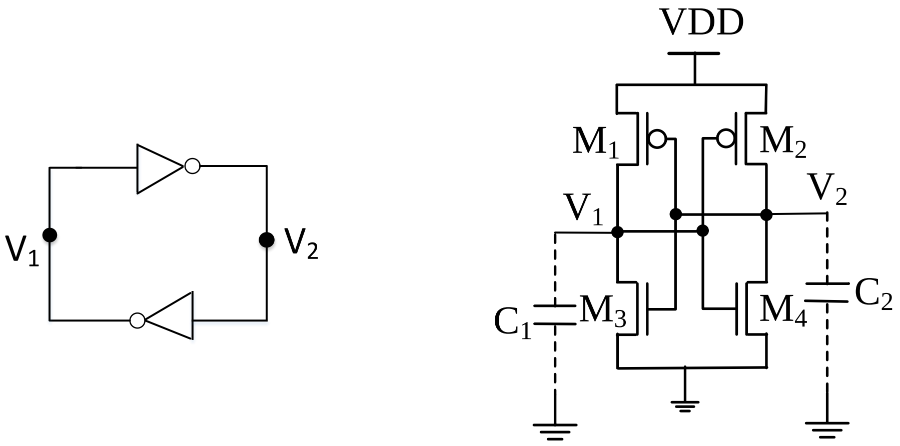
In this chapter, we discuss the fundamentals that are essential for understanding the core concepts in this dissertation. We provide the background of a simple electronic latch structure and discuss the Static random-access memory and DFF. Then, we will provide a comprehensive literature review about PUF, TRNG, and recycled ICs detection.

2.1 Electronic Latch

A typical electronic latch is a combination of two inverters connected to each other. Figure 2.1 shows the gate level and transistor-level diagram of an electronic latch. The figure 2.2 shows a simplified architecture of an SRAM array and cell. Based on the diagram, it is not very hard to observe that the core element of a simple SRAM cell is an electronic latch. Moreover, the electronic latch is also the core element of DFF. Since the entire dissertation will cover the application of the power-up state of SRAM cells and DFF, a deep understanding of the power-up states of an electronic latch will be necessary as it forms the base for the entire dissertation. A more detailed analysis of the power-up state of the electronic latch will be illustrated in the next chapter.

2.2 Physical Unclonable Function

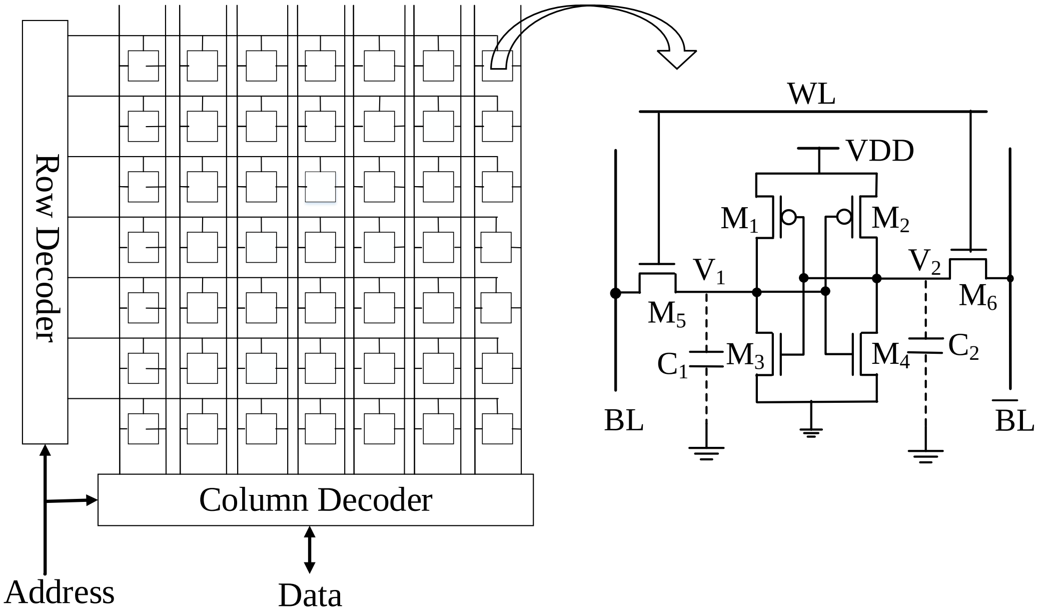
The physical unclonable function (PUF) is a die-specific random function or a silicon biometric, which can generate a unique response by applying a specific challenge. The uniqueness of PUF is derived from the variation during the fabrication process in an unintentional and uncontrollable way. These unique responses can be used for key generation and authentication in hardware security



(a) Gate level schematic for electronic Latch

(b) Transistor level schematic for CMOS Latch

Figure 2.1: Electronic latch gate level and transistor level schematic.



(a) A typical SRAM array.

(b) A six-transistor SRAM cell.

Figure 2.2: Simplified architecture of an SRAM array and a six-transistor SRAM cell

application [36]. Compared to storing the random information in non-volatile memory, PUF provides more resilient resistance to physical attack since information will disappear during power off. Moreover, each response of PUF will be unique ideally even for the same design in different dies.

Recently, a lot of PUFs structures have been proposed in past decades, such as arbiter PUFs, Ring Oscillator PUFs, Latch PUFs and many more [12, 37]. Bhargava et al firstly proposed a bit-stable PUF design (SRAM and sense amplifier) based on latch style structure [38]. The SRAM-based PUFs have been proposed in [3, 39], which exploit the power-up value of cells as response and corresponding address is the challenge. More specifically, one SRAM cell can store one specific desired information: either '0' or '1', which depends on the information that has been written in the SRAM writing process. However, the information stored in SRAM cells will be unpredictable once the array has been powered on without any write operation. The unpredictable state in SRAM cells will be decided by the strength of two back-to-back inverters in SRAM cells. Ideally, two back-to-back inverters will be slightly different during the fabrication process because of random-dopant fluctuation and line-edge roughness [40]. Ultimately, half of SRAM cells will be biased towards logical '0' and others will be towards logical '1' and this property has been used for implementing a PUF based on the power-up state of SRAM cells.

SRAM is a widespread building block in FPGA and many System-on chips, which make the implementation of SRAM PUF simple and require no further design additions. Although the SRAM PUF provides many attractive properties, one of the challenges is the reliability of SRAM cells. Firstly, the response of part of SRAM cells will be very sensitive to the noise and temperature/voltage variation and some researchers have reported that about 5-10 percentage cells will not obtain the same value every time it is powered up. Secondly, the aging degradation will affect the threshold voltage of MOSFETs and twist the strength of two back-to-back inverters. Finally, the response of SRAM cells will possibly be flipped during operation in the field. In order to exploit the start-up value of SRAM cells to perform some cryptography function (i.e. key generation), the response/startup value for each SRAM cell ideally should be reliable under different operating conditions (i.e. different temperature or voltage). Furthermore, in order to be applied to an end-user device, the requirement of reliability of SRAM cells is extremely high and typically the bit-error-rate (BER) should be nearly close to zero [4]. To cope with this, several mechanisms have been proposed to reduce bit-flipping and increase the overall cell reliability:

- Error Correction Codes (ECC): Usually, an ECC such as BCH code [13] has been exploited to reduce the bit-error rate to a specific level so that the SRAM array can be directly used as PUF.

- Preselection: the reliable SRAM cells have been selected during sampling at a earlier stage, before using it as a PUF [2,4,5,14].
- Hardening : a reverse burn-in aging have been applied to the SRAM cells that strengthens the response of cells [15,15–18]. Ultimately, the BER has to reach a reasonable level before SRAM has been deployed as a PUF.

However, ECC will introduce significant hardware overhead and hardening cannot ensure 100% reliability for PUF application. On the other hand, the preselection method requires inner design modification of traditional SRAM array structure, which will increase the design cost. In this dissertation, we will propose a new systematic preselection method toward 100% reliability for SRAM PUF. Moreover, the proposed method will not require any modification of the traditional SRAM array structure.

2.3 True Random Number Generators

Random number generators (RNG) have extensive practical applications. Besides cryptographic algorithms and secure protocols, random number also play a crucial role in IP piracy and IC overproduction [41,42], countermeasures against side-channel attacks [43,44], generating nonces or random seeds [45–47], and anti-counterfeit measures [48]. Consequently, much research is focused on developing “true” RNGs (TRNGs) with provable randomness properties as specified in certification tests, such as those recommended by the National Institute of Standards and Technology (NIST). Many authors have proposed ring oscillator-based TRNG by taking advantage of process variations and dynamic temperature variations [49–52]. Several researchers have also implemented TRNGs utilizing memories. In [53], the author proposed a flash memory-based TRNG by taking advantage of random telegraph noise in the intrinsic circuit. The DRAM-based TRNG is another direction. In [54], the author used the startup value of DRAM to implement a random number generator. In [55], the authors used the data remanence effect of DRAM to implement a TRNG. As static random-access memory (SRAM) is one of the basic memory components in computing systems, many researchers have tried to use it to develop a random number generator.

Researchers have developed TRNG structures utilizing SRAM as an entropy source [19–21]. However, the actual entropy from the power-up state of SRAM cells is very limited since the majority of SRAM cells will pose a stable response during power-up. Herrewege et al. presented a PRNG structure based on the power-up state of an SRAM array by implementing it in commercial off-the-self microcontrollers [22]. The author showed that the SRAM present in these microchips (e.g. PIC16F1825) cannot securely seed the PRNG. Some researchers also proved that the entropy of the SRAM array can decrease as the temperature decreases [22, 23]. These prior works indicate that entropy of the power-up state of SRAM is a very crucial parameter for creating a secure PRNG. However, the entropy of SRAM is very sensitive to operational conditions such as temperature. Consequently, researchers have introduced new methods to improve and preserve the entropy of the SRAM array. In [24] and [25], the authors proposed a minor modification to the conventional SRAM array to improve the instability of SRAM cells. In [26], the authors exploit device aging to improve the initial entropy of the SRAM array at deployment.

In this dissertation, we will demonstrate that the initial enhancement of entropy of an SRAM array by applying periodic controlled aging will have a counter effect. The initial enhancement of entropy will pose a risk of losing the entropy of an SRAM array during the normal operation in the field. Secondly, we propose a strategy, which applies periodic controlled aging during the deployment of SRAM. This prevents any significant degradation of entropy in the SRAM throughout its operational life and thereby ensures that the RNG always satisfies its randomness specifications.

2.4 Recycled ICs Detection

The problem of old recycled integrated circuits (ICs) being supplied and sold as new continues to grow due to the lack of efficient detection and avoidance techniques. The entry of these ICs into the critical global infrastructure (defense, aerospace, transportation, medical, etc.) can result in system and security failures with potentially serious consequences for social well-being. Electronic parts from old, discontinued production runs are often required to maintain outdated infrastructure and defense systems as the operational life of such systems is frequently extended far beyond initial plans because of budget limitations. (B-52 bomber aircraft that first flew in the 1950s are today being flown by the grandchildren of some of the original pilots.) The original component manufacturers (OCMs) have, meanwhile, long moved on to newer designs and technologies, and discontinued production of the obsolete ICs. To meet this critical need, maintenance and repair facilities have no option but to reach out to all possible suppliers of the legacy ICs, including untrusted third-party suppliers overseas. Information Handling Services Inc. has reported that counterfeit ICs represent a potential annual risk of $169 billion in the global supply chain [56]. Recycled ICs contribute around 80% of all the reported counterfeiting incidents [27]. As these recycled ICs often exhibit lower performance and reduced remaining useful lifetime [28], the reliability and safety of any system is significantly compromised if recycled chips are used in it. Additionally, the dis-assembly, cleaning, and restoration processes often employed to make a recycled part look new can also create other defects and anomalies [27–29] that can cause system malfunction.

Detection methods for recycled ICs can be broadly classified into two categories: test methods for detecting recycled ICs that are already in the market, and design for anti-counterfeit (DfAC) measures that can be implemented in new designs being readied for manufacturing. For the older designs, there are different standards (AS6171, AS5553, CCAP-101, and IDEA-STD-1010) currently in practice, which recommend conventional tests for detecting recycled ICs [57–60]. Among these standards, AS6171 has been adopted by the U.S. Department of Defense (DoD) . The primary challenges in implementing the test methods recommended in these standards are excessive test time and cost, lack of automation, and low detection confidence. While DNA markings are now commercially available for providing traceability of electronic parts [61] in the supply chain, the complexity of the authentication process, and excessive test costs, limit their wide adoption by the semiconductor industry [62]. Over the years, a number of researchers have also proposed test methods based on statistical data analysis to identify recycled parts [30–35]. However, a large number of chips are often required for creating the statistical models, which may be difficult to acquire for obsolete ICs. In recent research, several design-for-anti-counterfeit (DfAC) measures have been proposed as an alternative to the conventional recycling detection methods [63–69]. Unfortunately, DfAC measures cannot be applied for detecting recycled ICs already manufactured and circulating in the market.

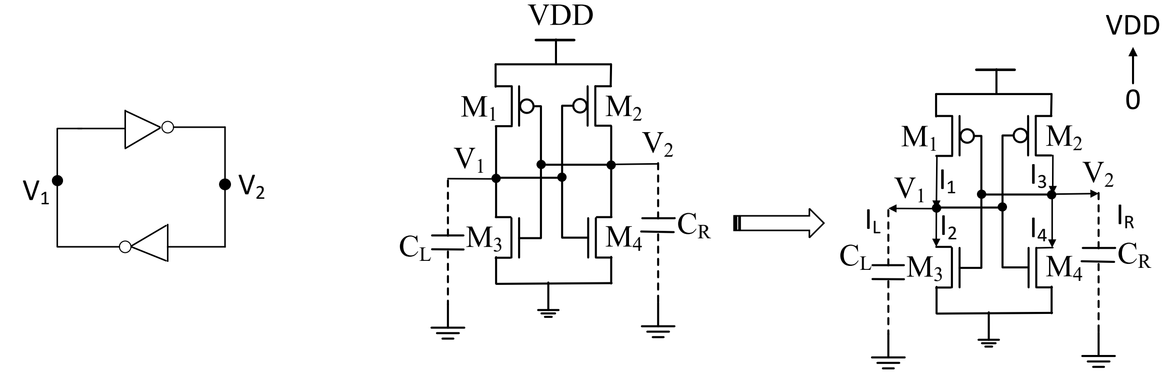
In this dissertation, we will exploit the power-up states of SRAM and DFF to identify the recycled ICs. Our proposed solution does not require any hardware modification to the existing design and can be applied to a wide variety of SoCs. The solution can be applied to both, ICs already circulating in the market, as well as those, to be manufactured in the future. The proposed approach is simple, effective, and low-cost, and requires minimal test support: a capability to read out the initial power-up state of the on-chip SRAM and DFF.

Chapter 3

Power-up States of Electronic Latch

In the last chapter, we mention that the electronic latch is the core element for either SRAM cells or DFF. Since the entire dissertation will discuss the power-up states of SRAM cells and DFF and its application, the power-up states of the electronic latch will be investigated in a detailed way in this chapter. This chapter will have three sections: the first section, we will build the general circuit model for power-up states of the electronic latch; the second section will be the ramp rate effect on power-up states of the electronic latch; the last section will be the aging effect on power-up states of an electronic latch.

3.1 The Modelling of Power-up States of an Electronic Latch



(a) Gate level schematic for electronic Latch

(b) Transistor level schematic for CMOS Latch

(c) Power up state of a latch

Figure 3.1: General circuit modelling for powering-up states of electronic latch

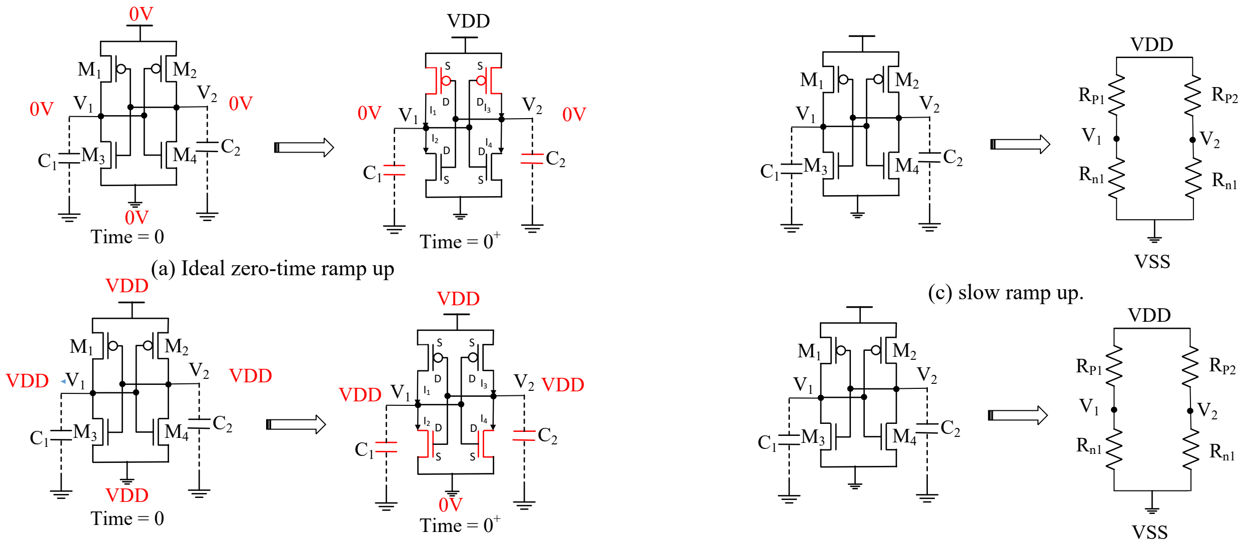
In this section, we will discuss the general modeling of power-up states of an electronic latch and the parameters that decide the final power-up states of the electronic latch.

In the figure 3.1, the figure (a) indicate a simplified gate-level schematic for a electronic latch. Two inverters are connected to each other. Note this connection poses positive feedback in this circuit structure. Figure (b) is the transistor level schematic for an electronic latch. As powering up this electronic latch, the Potential of the power pin will be increased from 0 to a high level (see figure (c)). This process can be simply modeled that two currents ($I_L and I_R$) charge up to two capacitors ($C_L and C_R$). Depending on the value of two capacitors and the currents, either V_1 or V_2 will be charged up quicker than the other. Since two inverters form positive feedback, either V_1 or V_2 will finally become one and the other will become zero. More specifically, three types of electronic latches will be categorized during the power-up process. The first type will be biased zero. When the electronic latch has been powered up, V_2 increases much quicker than V_1 and finally become one. The second type will be a biased one. When the electronic latch has been powered up, V_1 increases much quicker than V_2 and finally become one. The third type will be the balance cell. When the electronic latch have been powered up, the increasing rate of V_1 and V_2 are closed to each other. In this case, the circuit noise will determine the finial state of V_1 and V_2. These three types of latches will be applied to physical unclonable function and true random number generators and will have a more detailed discussion in later chapters.

3.2 Ramp Rate Effect on Power-up state of Electronic Latch

In this section, the ramp rate effect on the start-up value of the electronic latch will be discussed systematically and a corresponding analysis model will be also established.

In theory, the power supply to an electronic latch can be turned on very quickly, for example raising the high voltage power rail from VSS (ground) to VDD in nanoseconds or less. It can also be raised very slowly, over several seconds. Observe that here fast and slow time ramp rates must be defined relative to the charging/discharging time constants of the internal capacitances at the circuit nodes of the SRAM. These can range from hundreds of picoseconds (ps) to hundreds of milliseconds (ms) depending on whether the nodes are being charged/discharged by actively conducting transistors or by extremely small leakage currents. In practice, while there are generally no lower limits on allowable slow power supply ramp rates, a very fast power supply ramp can be limited by the drive strength needed to charge the large power rail capacitance. This is typically



(a) Ideal zero-time ramp up

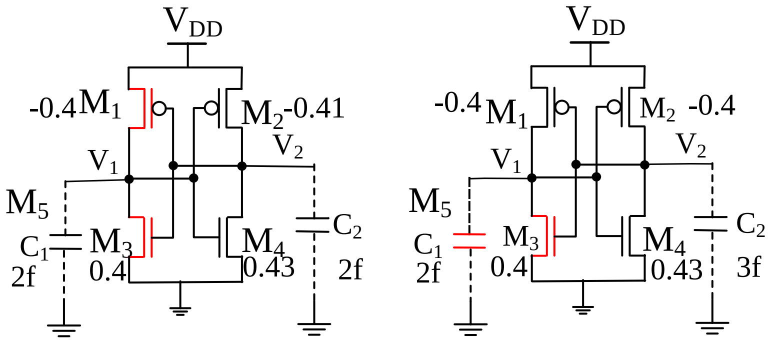
(b) Ideal zero-time ramp down

(c) slow ramp up.

(d) slow ramp down.

Figure 3.2: SRAM power up analysis model under different ramp

design limited, particularly if the SRAM power supply has to be switched on-chip, as would be the case if the SRAM PUF is to be read while the system-on-chip (SoC) containing the PUF is operating.

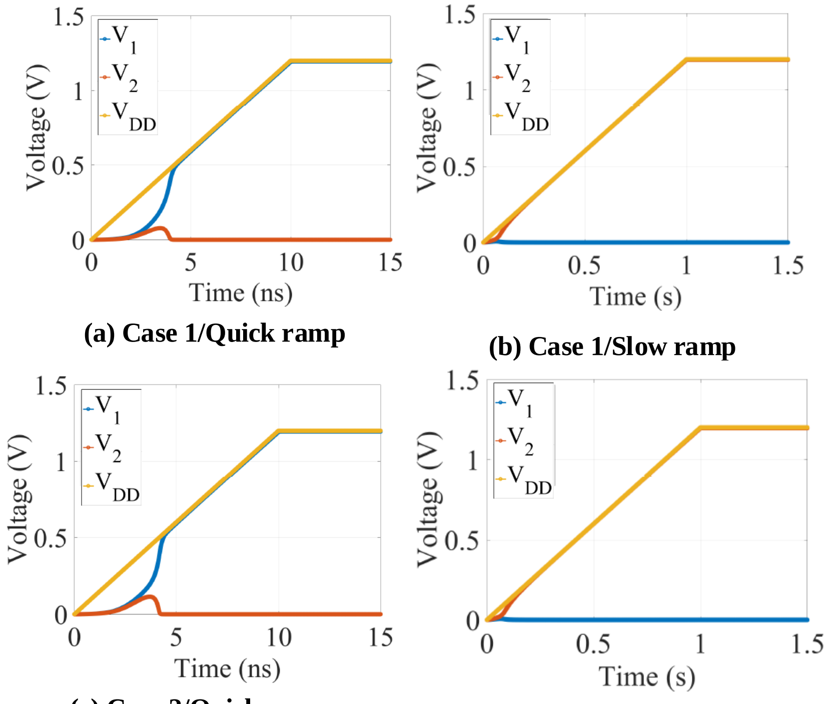


Case 1: $C_1 = C_2$

Case 2: $C_1 < C_2$

Figure 3.3: SRAM power up analysis model under different ramp

In the figure 3.2 (a), it indicate a typical electronic latch. When the electronic latch have been powered up under extreme quick ramp, M_1 and M_2 are complete on since the $V_{sg} > |v_{th}|$ if we assume V_1 and V_2 is starting with 0. This is because the potential of V_1 and V_2 cannot change suddenly during power up because of the capacitor of C_1 and C_2. Furthermore, the M_3 and M_4 are off during the early phase of powering up due to the same reason. Consequently, the currents which flow into the $C1$ and C_2 to built the potential of V_1 and V_2 are mainly depend on the strength of M_1 and M_2. The effect of two NMOS M_3 and M_4 can be ignored. If the VSS has been ramped down to



(a) Case 1/Quick ramp

(b) Case 1/Slow ramp

(c) Case 2/Quick ramp

(d) Case 2/Slow ramp

Figure 3.4: SRAM power up analysis simulation result under different ramp

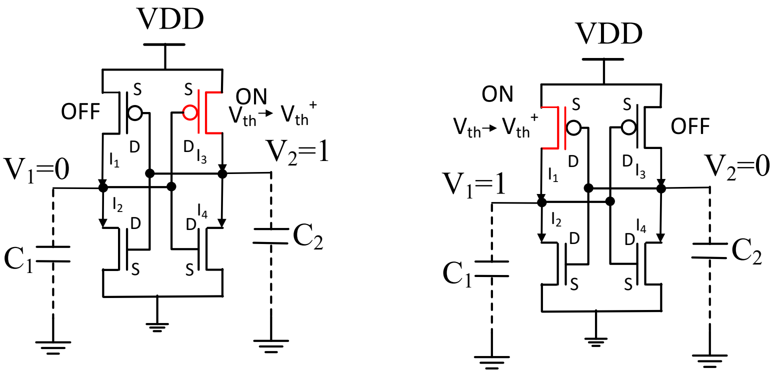
0 very quickly (ideal zero-time ramp down), the effect of PMOS will be ignored since the electronic latch implies a perfect up to down symmetric. When the electronic latch has been powered up under an extreme slow ramp, at each time of point, the power can be nearly regarded as a DC power since the changing of power is very small. If we assume the VDD becomes the DC power, the capacitors C_1 and C_2 will be almost quasi-equilibrium and can be treated as disconnection. under this extremely slow ramp, the two MOSFETs pairs can be equivalent to channel resistors for simplifying the analysis. The figure 3.2 (c) indicate corresponding analysis model for extremely slow ramp. The $R_{P1}R_{P2}R_{n1}R_{n2}$ represent the channel resistor for the four MOSFETS. Under this analysis model, the V_1 and V_2 will be decided by the value of four resistors because of the voltage divider structure. In other words, either V_1 will be larger than V_2 or V_2 will be larger than V_1 will be decided by the strength of four MOSFETs. As the positive feedback in the SRAM cells, either V_1 or V_2 will be logical 1 in a short time finally. Note in this model, the effect of the capacitor has been removed. To simply sum up, if powering the electronic latch under an extremely quick ramp, the effect of two NMOS (M_3 and M_4) can be ignored. If powering the electronic latch under an extremely slow ramp, the effect of the capacitor can be ignored. If powering the electronic latch under the

middle range ramp, the strength of both four MOSFETs and the value of two capacitors will all be included to decide the final power-up state of the SRAM cell (either V_1 or V_2 will be logical 1).

To further prove the ramp rate effect on the power-up state of the electronic latch, two specific simulation cases have been investigated through the HSPICE. The figure 3.3 indicate the parameter of corresponding simulation. In case 1, we only introduce the difference between four MOSFETs (M_1 M_2 M_3 M_4) and the value in figure 3.3 represent the threshold voltage of MOSFETs. Note that the M_1 is stronger than M_2 and M_3 is stronger than M_4 in this case. if these cells have been powered up by a quick ramp, the effect of M_3 and M_4 will be removed. The V_1 will be logical 1 after powering up since the M_1 is stronger than M_2 and more current will flow into C_1 to built the potential of V_1. The figure 3.4 (a) shows the corresponding HPSICE simulation result and blue curve (V_1) become logical 1. This validates our previous analysis. If the latch in case 1 has been powered up under a slow ramp, all four MOSFETs will be included to decide the final value. Since the V_{th} difference between M_3 and M_4 is larger than the difference between M_1 and M_2, the V_1 will finally be logical 0. HSPICE simulation also validate this result (see figure 3.4 (b)). Similarly, for the SRAM cell of Case 2, we only introduce the difference between two NMOS and capacitors. Under the quick ramp, since the NMOS effect will be removed and C_1 is less than C_2, the V_1 will finally become logical 1. Under the slow ramp, since the capacitor effect on power-up stated has been removed and M_3 is stronger than the M_4, the V_1 will be logical 0. The HSPICE simulation result also validate the analysis (see figure 3.4 (c) and (d)).

3.3 Impact of NBTI Aging Effect on Power-up State of Latch

The threshold voltage of a transistor increases under operational stress when the chip is used in the field. This is also true for an electronic latch when it is used for storing data. One of the main aging phenomena in ICs is negative bias temperature instability (NBTI), which occurs in PMOS transistors when they are negatively stressed [70, 71]. Interface traps are created at the Si-SiO_2 interface of PMOS transistor when its gate is pulled down to logic 0. Releasing the stress can achieve some but not complete recovery. As a result, the threshold voltage (v_{th}) of PMOS transistors increases over time [72]. This increase tends to saturate over a period of months and becomes minimal after 5-10 years in use. In summary, a PMOS transistor ages when it is turned on (the input is at logic 0) and relaxes when it is turned off (the input is logic 1). NMOS transistors experience much smaller threshold shifts from PBTI aging, although that may change at advanced technology nodes. A different aging phenomenon in CMOS circuits is hot carrier injection (HCI). [73,74]. Some high-energy electrons can attain sufficient energy when the transistor is conducting (on) to get trapped in the $Si\text{-}SiO_2$ interface near the drain terminal due to the lateral gate electric field. NMOS transistors are primarily affected by HCI because of higher carrier mobility, whereas it has very little effect in PMOS transistors [75]. Observe, however, that HCI occurs when there is current flow in the transistor channel. In practice, the impact of HCI in SRAM cells is minimal and can be ignored, because the transistors in memory cells are mostly non-conducting and experience much less switching activity than logic.

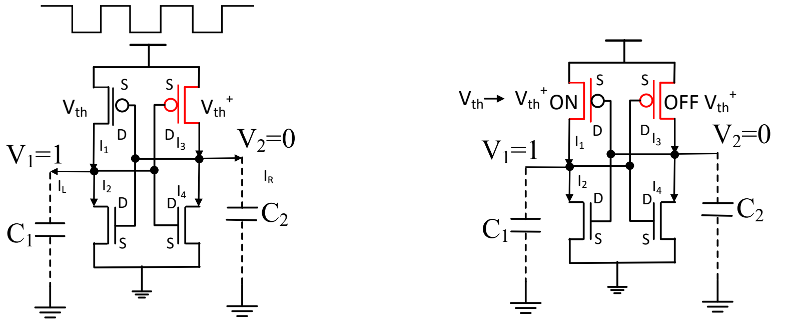


(a) Balanced latch stored 0

(b) Balanced latch stored 1

Figure 3.5: Constant Aging Effect on Power-up State of Latch

The effect of aging on the power-up behavior of an electronic latch can be explained using Figure 3.6. To begin with, we ignore process variation and assume all the transistor pairs possess the same device parameters. As a result, the threshold voltages of (M_1 and M_2) have same value ($v_{t1} = v_{t2}$). Similarly (M_3 and M_4) are identical. (We ignore any PBTI aging in the NMOS transistors.) Assume that for some initial period, the latch contains 1 ($V_1 = 1$), which sets the internal nodes $V_1 = 1.2V$, and $V_2 = 0V$. Consequently, the transistor M_1 will experience aging due to NBTI (as its V_{gs} is negatively stressed) and its threshold voltage will increase in magnitude over this time to $v_{t1}^*(> v_{th})$. Based on discussion in the previous subsection, this electronic latch is now biased and will power-up with logical 0 ($V_1 = 0$ and $V_2 = 1$) as threshold voltage of M_1 becomes larger (in magnitude) than M_2 after aging. *If we age the latch with 0, it will power up with 1 (and vice versa), for a perfectly unbiased latch.*



(a) Periodic powering up of a unbalance latch

(b) Aging effect with periodic powering up

Figure 3.6: Aging Effect on Power-up State of Latch with periodic powering up and down

We next analyze how the aging with the periodic powering up and down affects the bias of an SRAM cell. Assume that initially the threshold voltage of M_2 (v_{t2}) is higher than the threshold voltage of M_2 (v_{t1}) (see Figure 5.5). For simplicity, the effect of noise is omitted in the discussion. The cell will power up with 0 as M_1 will quickly reach saturation, and node 1 (V_1) will pull up to VDD. If we keep this state, transistor M_1 will be NBTI stressed as its gate is negative and v_{t1} will be increased in magnitude to v_{t1}^*. On the other hand, transistor M_2 will remain fresh. If we repeatedly power up the SRAM array and hold the state each PMOS transistor will age alternatively and remain unbiased if the v_{th}s of both the PMOS transistors are the same. If they are not, v_{t1} will be increased incrementally to reach $v_{t2}(> v_{th1})$ (or vice versa), and the cell will then stay unbiased.*If we age the latch with periodic powering up and down, the biased latch will always approach to unbiased state*

Chapter 4

Proposed Bit Selection Method for Robust SRAM based PUF

A physical unclonable function (PUF) is a digital circuit that can generate a die specific unique and stable response, which can be used for authentication and key generation. Since no major design or manufacturing modifications are required, exploitation of SRAMs to implement PUFs is a promising option. When initially powered up, individual SRAM cells acquire unique logic states based on the inherent bias of the cell. At advanced technology nodes, this bias is primarily due to unavoidable random manufacturing process variations, which are unpredictable and vary randomly from cell to cell, as well as chip to chip. When an SRAM is read out, these power-up states provide a unique output that is largely consistent during repeated power-up cycles for a given SRAM, but varies for different copies of the same part, as required of a PUF. However, this power-up state of SRAMs cannot be directly used (e.g. in cartographic key generation), due to unpredictability in some of the SRAM cells caused by electrical and electromagnetic noise and temperature fluctuations. We show in this paper that power-up states are also influenced by the power supply ramp rate at power-up, which can be yet another source of cell instability. To address the general problem of instability in SRAM power-up states that can result in inconsistent responses from SRAM PUFs, we present an effective stable cell selection method to identify the cells in the SRAM that are strongly biased, thereby resistant to circuit noise, voltage and temperature changes, and also aging. The data from the Silicon experiments presented here shows that the selected stable SRAM cells are highly reliable over temperature and voltage variations, with a bit error rate (BER) close to zero.

The rest of the chapter has been organized as follows. In section 4.1, the necessary background of the SRAM PUF has been discussed: SRAM cell characteristic along with the related works to improve the SRAM PUF reliability and uniqueness. In section 4.2, the data retention test have been introduced and it shows how it can facilitate us to select the strong cells for PUF application. Also, we proposed a comprehensive methodology to identify those strong SRAM cells for PUF application. In section 4.3, the silicon data have been provided and the reliability of the selected SRAM cells have been discussed. The conclusion and future works will be included in section 4.4. The majority of work in this chapter have been published initially in [76].

4.1 Motivation and Background

In this section, we will discuss the detail of SRAM PUF, its circuit theory behind it and main challenge. Then we will discuss the related works to address this challenge.

4.1.1 SRAM PUF

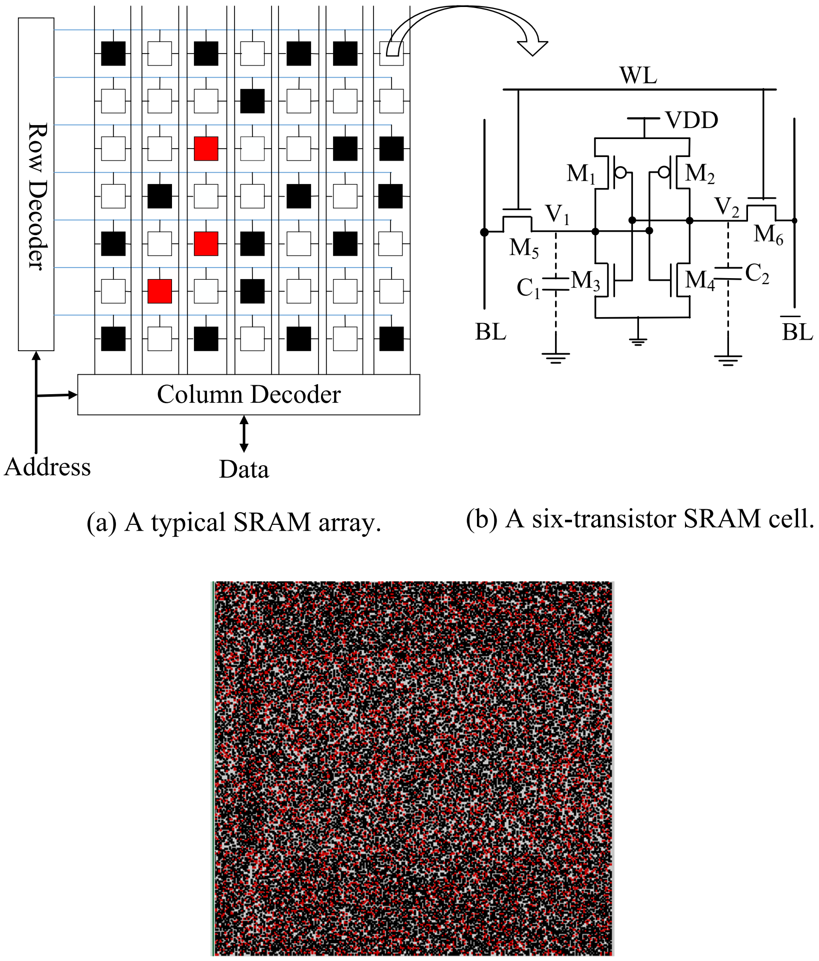
The SRAM PUF is one of the appealing PUF candidate due to it's easy implementation and nearly zero hardware penalty [2]. The idea of using the power-up value of the SRAM array as a response of PUFs was firstly proposed in [3]. The response uniqueness of SRAM PUFs is also competitive among the candidate of the existing PUFs [3]. A particularly attractive design for PUFs is based on the static random access memory (SRAM) array, as depicted in figure 4.1 (a). When an SRAM is initially powered up, each cell acquires a '0' or a '1' logic value. Figure 4.1 (b) shows the circuit schematic for a 6-transistor SRAM cell. Each cell has a pair of NMOS pull-down transistors, PMOS pull-up transistors, and NMOS pass transistors connecting each of the two (complimentary) cell output to the bit lines. In an ideal SRAM cell, if each transistor pair as described above is identical in every respect, including layout associated parasitic components, then the cell is perfectly balanced. In the absence of an asymmetric electrical noise, such a cell has a random 50% chance of acquiring either a '0' or a '1' state at power-up. However, even a small imbalance within a pair of transistors can result in a cell being biased towards either a '0' or a '1' power-up state. In nanometer-scale technologies, because of uncontrollable small random manufacturing variations, no two transistors in an SRAM cell are truly identical in practice. Consequently, when the SRAM array has been powered up, the process variation along with the noise and environment variation will classify the cells into two main parts:

- Neutral cell: The cell has no strong mismatch among pull-up PMOS pairs and pull-down NMOS pairs. It does not mean no process variation happened in M_1 M_2 M_3 M_4 as depicted in figure 4.1 (b). It only indicates the mismatch among these MOSFETs cancel each other and overall cells have no preference to the states 0 or 1. The final state of these cells will be determined by the noise present in the circuit.
- Skewed cell: The cell has relatively high mismatch among pull-up PMOS pairs (M_1 and M_2) and pull-down NMOS pairs (M_3 and M_4). These cells will have their preferred/consistent state, either a '0' or a '1'.

The figure 4.1 (c) is a bitmap/mask for a 64k bit SRAM array. In this bitmap, the red dots represent neutral cells, which show inconsistent power-up state during 100 power-up scenarios. The white dots and black dots represent the skewed cells, which indicate a consistent power-up value during 100 power-up scenarios. Based on the bitmap, about 90% cells hold consistent value. However, among these 90% cells, some cells may only obtain limited mismatch among pull-up MOSFETs and pull-down MOSFETs. In other words, these cells may change their response over time due to device degradation or under different environments such as temperature, supply voltage, or electromagnetic noise. These potential 'weak'/'neutral' cells will raise a challenge for SRAM-based PUF since it requires 100% reliability under PUF application.

4.1.2 Related Works

In order to address the reliability problem for SRAM based PUFs, some complex statistical solutions have been proposed to extract a stable signature [39, 77–79]. Figure 4.2 shows a basic step for PUF key generation based on soft error correction. In Figure 4.2, the overall procedure is divided into two phases: enrollment and key generation. For the enrollment phase, the raw PUF response will be fed into the ECC encoding part and the corresponding help data will be the output. These help data will be stored in the non-volatile memory (NVM) and are public in nature. When the users target to generate the key in the field, the help data will be loaded from NVM and fed into the ECC decoding part. Along with the new raw PUF response, the key will be generated as the output. This key generation scheme has two main drawbacks. Firstly, the ECC implementation usually requires significant hardware overhead. For example, if the target is to generate 128 bits key



(a) A typical SRAM array.

(b) A six-transistor SRAM cell.

(c) Bitmap of 64kb SRAM array

Figure 4.1: Systematic and random process variations.

with a bit-error-rate less than $1e-6$, the ECC typically needs 3k-10k PUF raw bits and the nature bit-error-rate for this raw PUF bits have to be less than 15%. Furthermore, for this case, the ECC will generate 3k-15k bits of help data and need to be store in the NVM [80]. The other drawback will be the weakness of the help data. It has been shown that the help data/syndrome bits are a source of leaking information [77, 78, 81]. This requires further careful design of the help data generation.

It has also been shown that the SRAM PUF cells will suffer from the reliability issue due to the aging effect in [82]. This is the first time that the authors in [15] exploit the potential benefit of the aging effect to reinforce the reliability of the SRAM PUF cells. The idea behind this is that the aging effect will increase the threshold voltage of the MOSFETs. Since the power-up state of the SRAM cells is highly dependent on the mismatch of two pairs of MOSFETs, the nature mismatch can be exaggerated by precisely choosing to apply the aging effect to specific MOSFETs.

In the [83], the authors show that the cells will have a bias of 1 when the cells have been aged with 0 (storing 0) and the cells will lean towards 0 bias when the cell have been aged with 1. By writing a proper value to the SRAM array and applying burn-in aging, the SRAM PUF cells can become more reliable. However, the bit-error rate cannot achieve a safe level by purely applying the burn-in aging and the ECC is still required afterwards. [15,15–17,84]. In the [18], instead of Negative-Bias Temperature Instability (NBTI) burn-in, the author exploit the Hot Carrier Injection (HCI) burn-In to reach nearly 100% reliable SRAM PUF cells. However, to efficiently introduce controlled HCI degradation into SRAM cells, the design of the SRAM cells needs to be modified and this increases the design cost and time, and also limits occasional reuse of functional SRAMs as PUFs.

In this chapter, to address the reliability problem of SRAM PUFs caused by unstable cells, the cell pre-selection method is utilised as the primary tool. The overall strategy is to run tests that select only highly stable cells as the PUF cells, ensuring high reliability. As a result, the bit-error rate (BER) of the selected cells will be extremely low along with reliability close to 100% under varied operating environments (i.e., different supply voltage and temperature). Other similar approaches include [2], where the authors exploit the spatial relation between the strong cells and selects the potentially qualified cells. However, this approach is more effective against systematic variations than the random variations observed in modern processes. Moreover, the BER cannot reach a safe level, and it still requires the application of ECC following the cell selection methodology. In [4,5], the authors introduce a tilt or bias to the SRAM cells to facilitate the selection of strong 1 and strong 0. However, to introduce the tilt, the SRAM cell needs to be modified. In the [14], the authors exploit a phenomenon that the cell will flip to its preferred value after a short power-off time when an opposite value has been written previously. However, the design that is used in [14] is based on ultra-low leakage technology. In more typical advanced technology with significant leakage, this method will be less practical. Moreover, the ramp rate at power-on and power-off impacts bit flipping, and this has not been considered.

4.2 Proposed Bit Section Method

In this section, firstly data retention test have been introduced and discuss the possibility of facilitating us select the strong SRAM for PUF application. By combining the data retention test

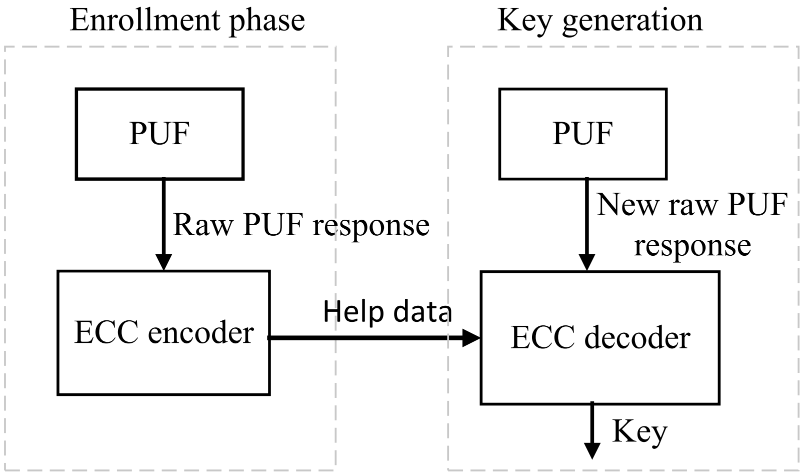


Figure 4.2: operation of SRAM PUF based key generation system

and the ramp rate effect on power-up state of latch (recalling the chapter 3), we proposed our systematic bit section method toward robust SRAM PUF.

4.2.1 Data Retention Voltage for Strong Cells Selection

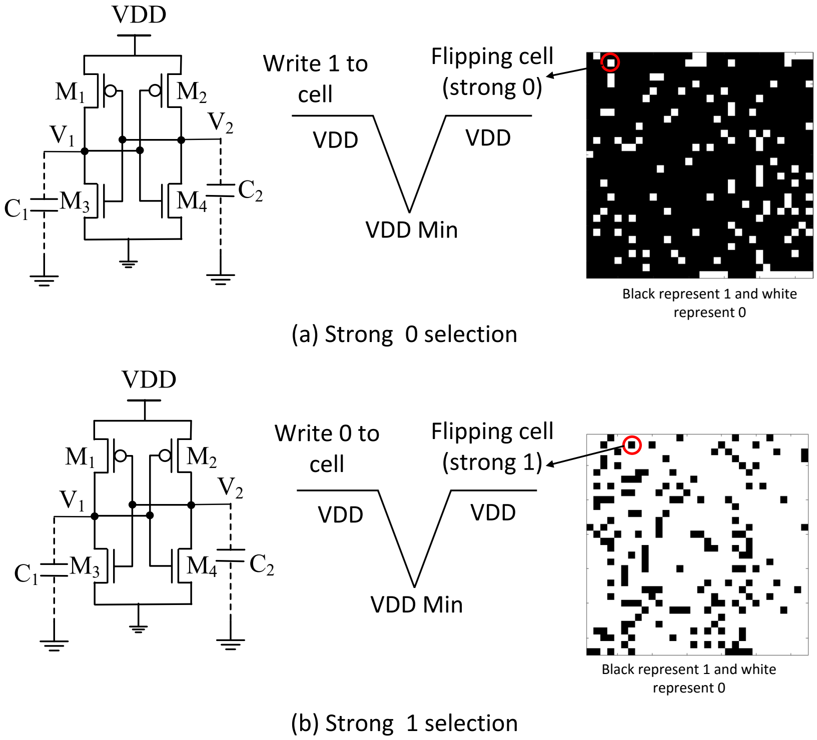


Figure 4.3: Selection of strong cells based on data retention voltage

The traditional data retention test for memories is designed to identify the most unstable SRAM cell that will cause faulty behavior during reading and writing operation [85]. The test is performed by writing all '1' to the cell and then lowering the SRAM supply voltage to a critical level ($V_{DD,Min}$). The entire SRAM array is held at this lower power level for a while, following which, V_{DD} is raised back up to its nominal value. A read operation is then performed for the entire array to identify the potential unstable or faulty SRAM cells that flip their values to '0' during this lower voltage power supply excursion. If not containing a stuck-at fault, these cells are inherently strongly biased towards the logic 0 state The same procedure can be repeated by writing '0' to SRAM array initially to identify cells with a strong 1 bias.

Table 4.1: Simulation Results for Strong Cell Calibration and Selection.

VDD Minimum	Ave $\Delta NMOS$	Ave $\Delta PMOS$	Δ NMOS+ $\Delta PMOS$	No of selected cells
0.14	0.044	0.042	0.086	356
0.15	0.048	0.046	0.094	253
0.16	0.051	0.049	0.101	198
0.17	0.054	0.055	0.109	139
0.18	0.059	0.060	0.120	93
0.19	0.066	0.068	0.135	51
0.2	0.066	0.075	0.141	34
0.21	0.074	0.083	0.157	17
0.22	0.079	0.083	0.163	7

This same data retention test can be exploited for identifying the most strongly biased stable SRAM cells that power up to either a 0 or 1 state. Recalling the definition of skewed cells in section II, these asymmetric cell will also be the most reliable candidates for PUF application. Figure 4.4 (a) show the selection of strong '0' process. Initially, the entire SRAM array has been written with 1 and then V_{DD} has been reduced to a critical level. After that, the V_{DD} is powered back to the nominal value again and the entire SRAM read out. The bitmap in figure 4.4 (a) shows the bit values after the final read operation. Here the white dots represent the flipped bits and are the potential strong 0 cells for PUF candidates. Similarly Figure 4.4 (b) shows the process of selecting strong '1'. The black dots represent the candidates cell for strong '1'.

Table 4.1 shows the simulation results and indicate the efficiency of selecting strong cell for PUF application based on the data retention test. In the simulation, 500 SRAM cells have been created and have a consistent bias from both the NMOS and PMOS transistors. The V_{th} for each transistor is randomly drawn from a normal distribution with standard deviation sigma of 20mv. From the 32nm technology files, the NMOS nominal V_{th} is 0.42252V and the PMOS nominal V_{th} is -0.41174V. All cells are firts written with all 0 (1), powered down to V_{DD} minimum for several seconds and then powered back up to nominal value again. The cells that observed to have flipped their initial value are selected as the strong cells. In Table 4.1, the first column shows the V_{DD} minimum value and the last column represents the number of cell have flipped their initial value. The fourth column presents the sum of biases V_{th} from the NMOS pair and PMOS pair. Observe from Table 4.1, that as the V_{DD} minimum value increases, the selected cell will be more biased because of the larger value of $\Delta NMOS + \Delta PMOS$. This simulation result shows that the data retention $V_{DD,Min}$ voltage can allow us to calibrate the strength of the selected cells for PUF applications.

4.2.2 Systematic Selection Method for Reliable SRAM PUFs

Our challenge is how to reliably select those reliable SRAM cell for PUF application. In this section, a systematic selection method for reliable SRAM PUF cells is proposed.

Recalling the discussion in chapter 3, the power-up state of some SRAM cells may have different power-up values under quick and slow V_{DD} ramp rates because of opposing inherent bias in the PMOS and NMOS transistors. Thus cells display conflicting power-up bias from the individual contributions of capacitive and MOSFETs imbalances. Recall the case 1 and 2 examples in section III. These cells are potentially unreliable cells for PUF application. In strong cells both the PMOS and NMOS transistors should display the same directional bias. Furthermore, in Section IV, we proposed exploiting data retention voltage to calibrate the strength of this bias in the SRAM cells. By combining these two properties, a systematic selection strategy can be developed for targeting virtually 100% reliable cells for PUF application.

Figure 4.4 shows our procedure of selecting reliable cells. A two-level selection process has been created. In the first level, individually, quick and slow V_{DD} power on ramp rates will be applied to the SRAM array. Then the bitmaps of the SRAM array under these two power up ramp rates are collected. Next, a combined bitmap is created indicating where the two bitmaps agree, i.e. by the XNOR the bitmaps of the quick ramp and slow ramp. All the black bits in this bitmap represent potential reliable bits for our application. Data retention tests, which comprise the second-level in this test procedure, are next performed to calibrate the strength of the potentially reliable bits identified by the first level testing. In Figure 4.4, the bitmaps 3 and 4 represent the corresponding bitmap after performing the data retention test by writing all 1 and all 0. The white dots represent potential strong 0 cells in bitmap 3 and black dots represent potential strong 1 in bitmap 4. In the bitmap 6 is the final bitmap pool for all potential reliable cells for PUF application (black dots in bitmap6). The bitmap 6 is generated by checking whether black dots in bitmap 3 is still black in bitmap 5 and whether white dots in bitmap 4 is still black in bitmap 5. If the dots in bitmap 3 and 4 fulfill the condition, black dots will be placed in the same location in bitmap 6 and this represents this cell is a reliable cell for PUF application and the rest of the location will be put white dots.

Following manufacturing, we propose the application of the two level test to select candidate stable cells and develop the dark-bit mask. The specific PUF register size, for example 128 bits or 256 bits, can be chosen from the dark-bit mask based on the application requirement. To maximize PUF response stability, the dark bits can be further rank ordered based on the $V_{DD,Min}$ voltage of the second level test to allow selection of the most stable bits for use in the PUF. The address information of selected cells can be stored in non-volatile memory, for use in efficiently generating the PUF response.

4.3 Silicon Result

In this section, we present and discuss silicon results for the power-up stability and reliability of SRAM cells that have been selected based our proposed method. We have conducted experiments using commercial off-the-shelf (COTS) SRAM memories to demonstrate the effectiveness of our proposed approach. The chips are Microchip and 23K640-I/SN SPI Bus Low-Power Serial SRAM memories. The total memory capacities of both SRAM chips are 64K bits.

In first test, the SRAMs are powered up under extreme quick and slow ramps and corresponding bit-maps generated (see figure 4.4). The second level test is the data retention test. Here, different $V_{DD,Min}$ are employed for data retention testing to select and rank order the strongest cells in

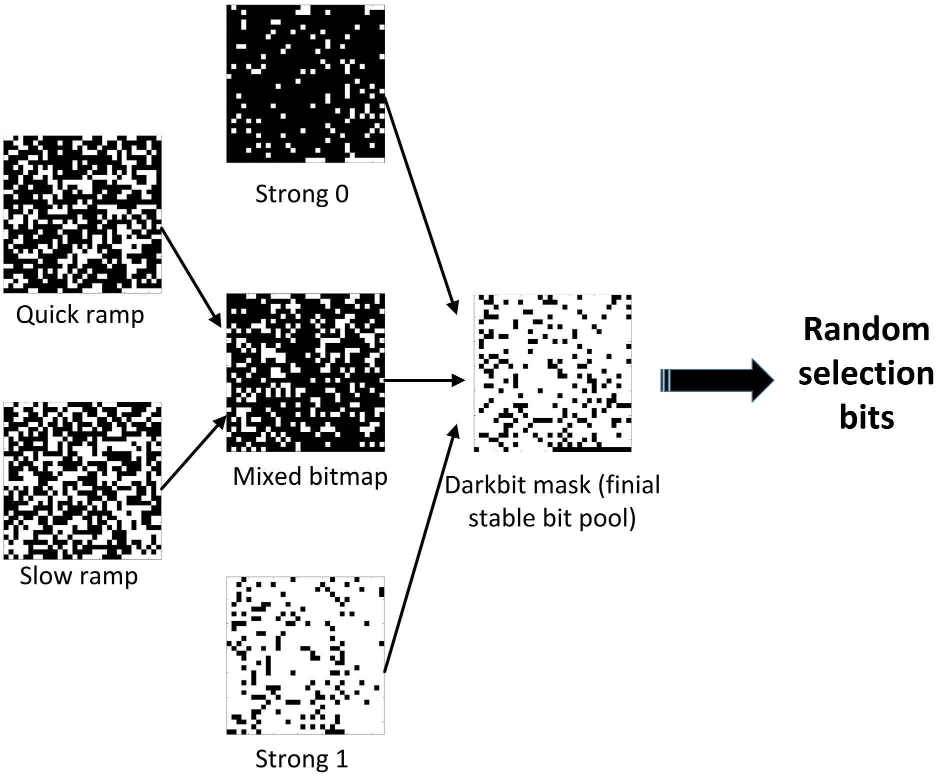


Figure 4.4: A systematic selection method toward reliable SRAM PUF

the array. The results from the two test levels are combined as describe above. In this way, the selected cells can achieve very high reliability. Figure 4.5 presents the different data retention voltage $V_{DD,Min}$ versus % of 1s observed after powering up back to nominal V_{DD} voltage. The x axis represents the $V_{DD,Min}$ which is the minimal voltage in the data retention test (see section IV). From the plot, it can be observed that the number of cells that flip from the initial written value become fewer as the $V_{DD,Min}$ is raised. Thus the silicon experiments validate our results from simulation in Table 1.

In the next experiment, these cells will be powered up under different temperature and voltage to evaluate their reliability in different environmental conditions.

Groups of SRAM cells selected using the stability testing approach described earlier (for different V_{DD} minimum) were tested for consistency of their power-up states under different temperature (25°C 50°C 85°C). Three chips were heated using a ThermoSpot direct contact probe system (see experimental setup for accelerated aging in Figure 4.6). This system is an industry standard benchtop temperature cycling system, used for testing circuits over a wide range of

Table 4.2: Evaluation of reliability of selected cells under temperature variation

Minimum VDD	Chip1			Chip2			Chip3		
	80°C	50°C	25°C	80°C	50°C	25°C	80°C	50°C	25°C
	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells
0.53	5.01	4.40	4.30	5.71	5.45	5.43	5.13	4.36	4.31
0.54	2.91	2.71	2.71	4.81	4.33	4.33	3.21	3.17	3.12
0.55	1.29	1.28	1.28	4.32	4.08	4.01	1.60	1.58	1.55
0.56	0.46	0.46	0.46	0.95	0.88	0.88	0.61	0.60	0.58
0.57	0.12	0.12	0.12	0.13	0.13	0.13	0.15	0.15	0.15
0.58	0	0	0	0	0	0	0	0	0

Table 4.3: Evaluation of reliability of selected cells under voltage variation

Minimum VDD	Chip1			Chip2			Chip3		
	3.6V	3.3V	2.7V	3.6V	3.3V	2.7V	3.6V	3.3V	2.7V
	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells	% of unstable cells
0.53	4.18	4.30	4.07	5.47	5.43	5.41	4.50	4.31	4.37
0.54	2.65	2.71	2.60	4.33	4.33	4.33	3.01	3.12	3.15
0.55	1.20	1.28	1.16	4.01	4.01	4.01	1.68	1.55	1.57
0.56	0.466	0.46	0.46	0.88	0.88	0.883	0.56	0.58	0.59
0.57	0.12	0.12	0.12	0.13	0.13	0.135	0.17	0.15	0.18
0.58	0	0	0	0	0	0	0	0	0

temperatures. Table 4.2 presents the results of the reliability test for three different temperature for the three chips. The chips were powered up 1000 times at each temperature and the power-up value read out and recorded. To evaluate the reliability of the selected cells, the percentage (%) of unstable cells is calculated. This is defined as follows: number of unstable cells divided by number of selected cells. The number of unstable cells are those that show an inconsistent power-up value anytime during the 1000 read-outs. Tables 2 verifies in silicon that, as expected, cells that have been selected using larger $V_{DD,Min}$ show better reliability because the percentage (%) of unstable cells decrease as $V_{DD,Min}$ increases. Note that if the $V_{DD,Min}$ of 0.58V (or higher) is employed during the data retention test, all the selected cells are stable for 1000 power up cycles, and their use in a PUF will ensure very high reliability. Observe also, from comparing data for 85°C and the data for 50°C, the percentage (%) of unstable cells at 85°C is larger than that at 50°C. This is because it is well known that higher temperature causes higher cell instability [86].

Groups of strong cells were similarly also selected to evaluate their stability for different supply voltage level (2.7V 3.3V 3.6V). Table 4.3 shows the corresponding test results. All chips were again powered up 1000 times to test stability at the different operating voltages. The percentage (%) of unstable cells is again calculated for different groups of selected cells. In Table 4.3, also observe that the percentage (%) of unstable cells reduces as V_{DD} minimum keep increase. If the V_{DD} minimum is 0.58V, the corresponding group of selected cells is 100% reliable.

Finally, the experiment was also repeated to study the impact of aging on the stability of the selected strong cells. Here we applied controlled random aging to the SRAM chips to mimic normal operation in the field. Random patterns were written into the SRAM, with the ratio of 40% 1s and 60% 0s. These random patterns are modified and updated periodically. After 2 weeks of this controlled random aging, the start-up value of each SRAM chip was read out 1000 times to check the reliability of the selected cells. Table 4.4 shows the percentage (%) of unstable cells for selected cells after controlled random aging. The result again indicates the of unstable cells have been decreases as the $V_{DD,Min}$ increases. If the V_{DD} minimum is 0.58V, the selected groups of cells is 100% stable and reliable.

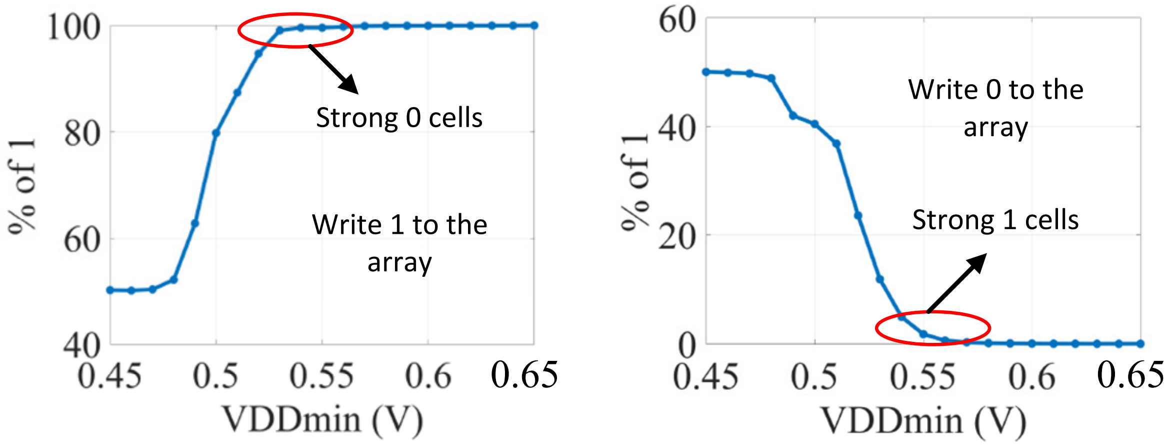


Figure 4.5: % of 1 on the 64k bits SRAM array during data retention test

Based on the reliability expriments presented for varying voltages, temperatures and aging, it has been shown that by selecting an appropriate $V_{DD,Min}$ voltage for stable cell strength selection, it is always possible to ensure desired reliability of an SRAM PUF. The trade-off is that while

Table 4.4: Evaluate the reliability of selected cells under aging

VDD Minimum	Chip1	Chip2	Chip3
	% of unstable cells	% of unstable cells	% of unstable cells
0.53	4.45	5.75	4.58
0.54	2.73	4.43	3.15
0.55	1.31	4.16	1.58
0.56	0.46	0.88	0.61
0.57	0.12	0.13	0.15
0.58	0	0	0

using a higher $V_{DD,Min}$ during cell selection yields more stable and reliable cells, there are fewer such cells in any given size of an SRAM array. This limits the reliability of the PUF (with a given number of bits) that can be realized from any size of SRAM array.

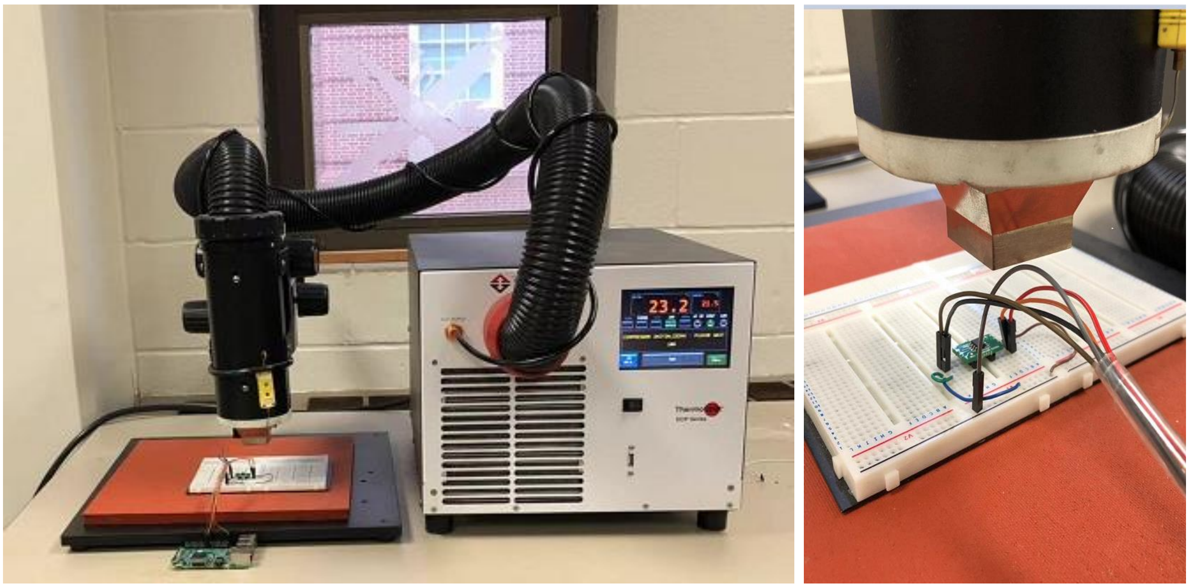


Figure 4.6: Testing the chip under different temperature by using ThermoSpot direct contact probe system.

4.4 Conclusion

While SRAM arrays are particularly attractive for use as PUFs, errors in the PUF response due to instability caused by voltage, temperature, environmental noise, and degradation due to aging is a challenge. In this chapter we show for the first time that power-up states are also influenced by the power supply ramp rate at power-up, which can be yet another source of cell instability. To address the general problem of instability in SRAM power-up states that can result in inconsistent responses from SRAM PUFs, we present an effective stable cell selection method to identify the cells in the SRAM that are strongly biased, and thereby resistant to circuit noise, voltage and temperature changes, and also aging. The data from the Silicon experiments presented here shows that the selected subsets of SRAM cells are highly reliable over temperature and voltage variations, with a bit error rate that can be brought down (BER) close to zero.

Chapter 5

Aging-Resilient SRAM-based True Random Number Generator

A random number generator (RNG) is an important building block for cryptographic operations primarily to generate random nonces and secret keys. The power-up value of an SRAM array has been widely accepted as an entropy source for generating random numbers. However, only a few cells of the SRAM are truly random upon repeated power-ups; the vast majority of cells display a distinct bias from manufacturing process variations. Consequently, a relatively large SRAM array is required to obtain sufficient entropy for generating random numbers. In this chapter, we propose an SRAM-based random number generation approach, which continually manipulates device aging during operation to constantly maximize entropy for the entire deployment period. Silicon results are presented to validate our proposed approach. The contribution of this chapter are two-fold, and described as follows:

- The first contribution of the paper is that we demonstrate any initial enhancement of the entropy of an SRAM array through controlled and selective aging of devices prior to deployment (as proposed, for example, in [26]) is not retained for very long during actual use, and can have an adverse effect on security if relied upon at design time. This is because, with many different random data patterns stored in a functional SRAM during normal operation, all devices get maximally aged from stress over time as the increase in threshold voltages of transistors due to NBTI saturates. Note that there is a bias towards 0 in the random data stored in SRAMs [87]. All transistors now experience nearly equal elevation in threshold voltage. Consequently the net imbalance in threshold voltages in each SRAM cell, which decides the power-up state of the cell, returns to what it was before any compensating aging stress was applied, i.e., as at the time of manufacture. Thus aging cannot be reliably used to increase long term entropy in the

SRAM states at power-up if the memory is simultaneously also used for functional purposes, as is common in low cost applications. The resulting loss of randomness in the RNG due to those uneven aging pattern in SRAM cells can present a significant security risk.

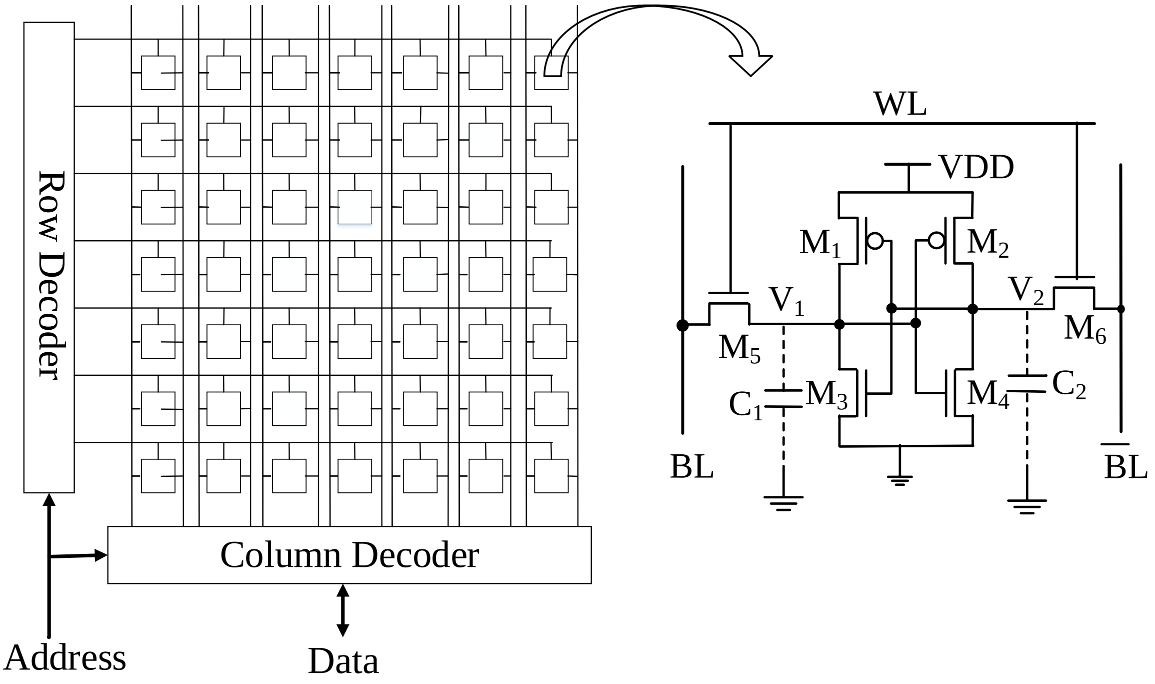
- The first contribution motivates the second contribution of this paper, which is the development of a controlled periodic aging strategy during operation. This prevents any significant degradation of entropy in the SRAM thr-oughout its operational life, and thereby ensures that the RNG always satisfies its randomness specifications. We have illustrated this using Figure 5.3. The concepts and methodologies presented in this paper have been validated through extensive silicon experiments as discussed later Sections.

The previously published proposals to boost the entropy in SRAMs by deliberately aging them after manufacturing may have adverse effects on the quality of the random numbers generated over time. Instead, we show how one can preserve the initial entropy for an SRAM array and maintain the quality of the random numbers throughout its lifetime. We propose an approach to restore the aging degradation, which reduces the entropy of an SRAM array. Now, the specific details of the optimum restoration approach will depend on the application, SRAM types, manufacturing technology, operating conditions, and many other variables. These question cannot be meaningfully answered in general terms and are best left to be addressed by the designers of specific (IoT) applications. It is up to them to decide how best to implement the restoration process for their design. Our contribution here is to show (for the first time) that it is indeed possible to compensate for the loss of entropy from aging through a simple and general approach, which may admittedly not be optimal for all designs.

The rest of the paper is organized as follows. Section 5.1 introduces the corresponding background and motivation. Section 5.2 discusses our proposed approach for generating true random numbers. Experimental results are given in Section 5.3. Finally, we conclude our paper in Section 5.4. The majority of work in this chapter have been published initially in [86].

5.1 Background: Randomness and the Power Up-state of SRAMs

The most significant device parameters influencing the power-up state of an SRAM cell are the threshold voltages (v_{th}) of the MOS transistors. Note that a conventional SRAM cell consists of six transistors as shown in Figure 5.1. Here four of the transistors (M_1, M_2, M_3 and M_4) form a bistable latch that stores the 1-bit of data in the cell. BL, and $\overline{BL}$ are the complementary bit lines that provide access to the stored bit through the access transistors M_5 and M_6.



(a) A typical SRAM array. (b) A six-transistor SRAM cell.

Figure 5.1: Simplified architecture of an SRAM array and a six-transistor SRAM cell.

5.1.1 Effect of Process Variations on the Power-up State

When an SRAM array is powered-up, initially each individual memory cell randomly acquires either a logic 0 or logic 1 value. During design, the MOS transistors in each matched pair (M_1 and M_2, M_3 and M_4, and M_5 and M_6) in Figure 5.1 are carefully layed out and fabricated to be completely identical, including all their layout related parasitic components. The perfect symmetry of the memory latch in a SRAM cell maximizes noise margins during operation. Consequently, each SRAM cell should ideally have a 50% chance of acquiring either logic 0 or logic 1 when powered up, the actual value decided by random noise. However, in practice most MOS transistor pairs will not be perfectly matched due to random manufacturing process variations, the most significant of which are the small variations in the v_{th} in each MOSFET. v_{th} differences in the PMOS and NMOS transistor pairs can either both cause a cell bias in the same direction or in opposite directions; the net imbalance decides the overall bias towards either 1 or 0 at power-up. A larger net imbalance in the transistor pairs result in a more skewed SRAM cell. If the net v_{th} difference is small, the power-up values may be still be somewhat random, with a bias towards either 0 or 1. On the other hand, for relatively large net v_{th} imbalances, the power-up state may be stable and always the same over multiple power-up cycles. Based on this behavior at power-up, the SRAM cells have been divided into three categories in the literature – no-skewed cells, partially-skewed cells, and fully-skewed cells [88]. Experience shows that 80-90% of the cells in an SRAM are typically fully-skewed or stable cells at power-up, 5-15% are partially skewed unstable cells that sometimes display random behavior.

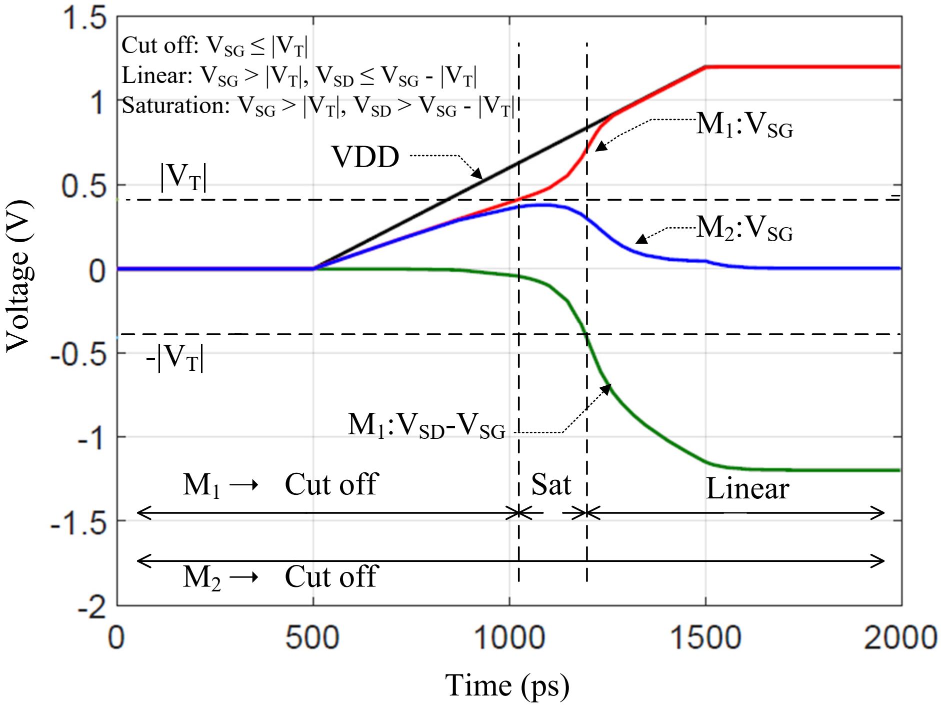


Figure 5.2: Timing diagram of internal nodes of an SRAM Cell during the power-up.

In order to investigate how the difference in v_{th} influence the power-up state of SRAM in more detailed way, the HSPICE simulation have been performed by using 32 *nm* bulk Predictive Technology Model (PTM). Figure 5.2 shows the voltage potential of different nodes for a single SRAM cell corresponding to Figure 5.1.(b). In the design stage, the parameter of all the MOSFETs are intended to be identical (M_1,M_2, M_3,M_4). However, each MOSFET will obtain variability resulted from manufacturing process variation. To explain the power-up behavior, we ignore the process variation of NMOS (M_3 and M_4) transistors (for simplicity and the power-up of an SRAM cell depends of PMOS transistors for fast power supply ramp [76]. The v_{th} of M_2 is assumed to be higher than M_1 (absolute value) for 20%. As depicted from Figure 5.2, the potential of node 1 (V_1) and node 2 (V_2) increase nearly at the same rate initially. However, at some point of time, M_1 reach the saturation region due to lower V_{th}. In contrast, M_2 remain in cut-off region. Consequently, potential across node 1 (V_1) reaches to VDD as higher current will flow through M_1 compared with M_2, while potential across node 2 (V_2) decreases to zero. In summary, the cell will power up with 1 if the absolute value of v_{th} of M_2 is higher than M_1, and vice versa.

5.1.2 SRAM-based Random Number Generation

While the vast majority of bits in any SRAM are fully-skewed and therefore, stable at power-up, a large SRAM still has enough randomness (entropy) to support generation of a random number. This is readily seen by considering the result of an XOR operation over all the bits of the SRAM. The resulting single binary bit will become random over multiple power-up instances if even a relatively few bits in the SRAM array are unstable. A multi-bit random number, for example 32-bits, can be easily obtained by partitioning a large SRAM into 32 arbitrary blocks, and then performing XOR operations on each of the individual blocks to generate the 32 random bits, i.e., one bit from each block. In practice, as discussed in the next section, RNGs used in security applications employ more sophisticated methods to extract the entropy (e.g. using a hash function, see Figure 5.4) instead of the simple XOR operation.

5.1.3 SRAM Entropy and Aging

Observe that RNG randomness measures can be further improved if there are more unstable bits in the SRAM. Some research, e.g., [24], has proposed modifications to the conventional SRAM array to increase the instability of SRAM cells. However, such a custom approach may not be appropriate for low cost IoT nodes. More recent work has suggested the use of controlled device aging to maximize the number of unstable cells. Recall that PMOS transistors experience NBTI aging under negative bias stress, resulting in an increase in the magnitude of their threshold voltage over time from a negative bias stress at the gate. NMOS devices experience a similar PBTI effect, but this has traditionally been observed to be much smaller. The NBTI stress windows are the periods when the PMOS transistors are ON; similarly NMOS transistors are stressed when they are ON. There is some partial recovery from BTI degradation when the stress is removed, i.e., when the transistors are OFF, but, for random inputs, the average threshold voltage shift due to aging tends to increase over time and finally saturate in the long run as shown in Figure 5.6.

In newly manufactured SRAMs, where the threshold shifts from aging have not yet occurred and are far from saturation, controlled aging offers a way to counteract the inherent bias in cells from process variations and increase the number of unstable cells. Consider an SRAM cell that is powered-up and acquires a low (state 0) at the left output of the latch (transistors M_1 and M_3 in Figure 1). This cell obviously has an inherent 0 bias which can be caused by the left NMOS transistor in the NMOS transistor pair having a smaller threshold voltage, and/or the right PMOS in the PMOS pair having a smaller (in magnitude) threshold voltage . Lower threshold transistors are the first to turn ON at power up, which in this case would force the left output to 0 and the right output to 1. Notice that after power up these very same transistors, the left NMOS and the right PMOS are ON and therefore under BTI stress. This tends to increase the magnitude of their threshold voltages, thereby reducing the bias in the cell over time because the complementary transistors in each pair are OFF and therefore do not similarly degrade with time. Thus, BTI aging can be used to reduce, and even overcome, the inherent bias in SRAM cells through deferentially aging transistor pairs to create an opposite bias by holding the initial power-up state for a controlled period. This approach has been proposed [26] as a method to increase cell instability and thus to increase the SRAM entropy before it is deployed in the field.

However, the problem with such an methodology is that this increase in entropy, which depends on differential aging of the transistor pairs in the SRAM cells, can only work when the transistors are new and relatively unaged. Over time, once all transistors in the SRAM get significantly aged from the stress of a very large number of random functional data cycles, the threshold voltage degradation in all PMOS transistors saturates at a nearly the same levels, eliminating any differential impact, and hence returning the SRAM cells to their biases at the time of manufacture. Thus, if a RNG is designed with an SRAM assuming the higher level of entropy achieved through selective aging pre-deployment, it may fail randomness specifications over time as the number of unstable bits in the SRAM decrease. This is clearly observed in the silicon results presented in later sections.

From the above discussion, it is clear that RNGs must be designed assuming that SRAM entropy cannot be reliably increased beyond the level inherent at manufacture. On the other hand, our silicon results show that on the SRAM entropy can actually drop below its initial level during the early period of deployment, before device aging saturates. This occurs because not all stress in functional memory during operation is entirely random. For example, memory data statistics show that in some applications, SRAM cells have a distinctly higher likelihood of storing a 0 value [87]. Such a storage pattern will tend to increase cell bias dis-proportionally in one direction, reducing the number of unstable cells. We therefore develop a controlled aging strategy to ensure that the SRAM entropy never drops (within bounds) below its initial entropy at manufacture anytime during deployment. This can be achieved by periodically powering up the SRAM and then holding the power-up state for a calculated window of time to neutralize any built up bias from systematic stress. This period should be small enough to allow only minimum shifts in the threshold voltage, well within the threshold voltage mismatch range acceptable for RNG operation, to ensure that the controlled BTI stress cycle does not itself leave an unacceptable bias in any cell. The power up and hold operation can then be repeated as frequently as needed to ensure proper RNG operation. Notice that any SRAM cell, utilized in the RNG, that for any reason develops an unwanted bias will be stressed back towards neutrality during each controlled BTI stress cycle, with small threshold voltage shifts. In case the threshold shift caused by the BTI stress overshoots neutrality and creates an opposite bias in the cell, then during the next controlled stress cycle, the cell will power up in the complementary state creating BTI stress in the complementary set of transistors, thereby again driving the cell bias towards neutrality. The proposed methodology is thus fully self-correcting.

The plot in Figure 5.3.a illustrates the change in entropy from uncontrolled aging during the normal operation of an SRAM in the field. Following deployment, the entropy initially decreases from the biased stress that exists even in the random data, because v_{th} shifts from aging are the greatest in new devices. However, it soon starts recovering as the v_{th} shifts start saturating and therefore equalizing in the SRAM transistor pairs (Figure 5.3.a). The entropy ultimately recovers to its initial level (E_0) at time T. At this stage, aging degradation has saturated and virtually equalized in all the transistors, bring entropy back to the unaged level. The approach proposed in [26] increases the initial entropy ($E_0 + \Delta E$) with careful pre-deployment aging (Figure 5.3.b).

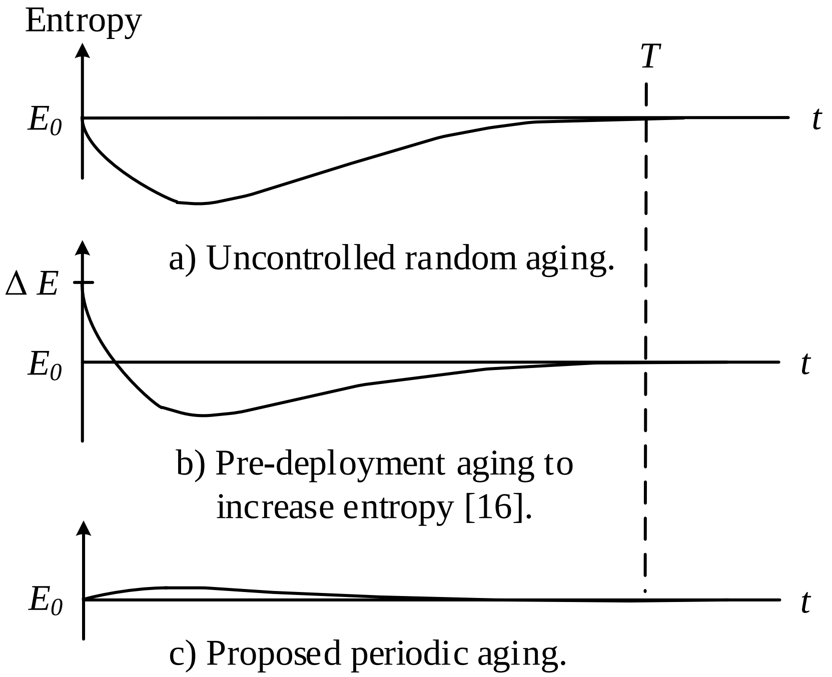


Figure 5.3: SRAM entropy versus deployment time for different aging strategies.

However, the entropy quickly drops below E_0 after just a short period of deployment in the field, and again slowly recovers back to E_0. Finally, Figure 5.3.c shows how we can always maintain (or exceed) the initial entropy using our proposed periodic aging strategy. This guarantees the randomness specifications for the TRNG throughout the deployment window.

5.2 Proposed Approach for Creating an Aging-Resilient SRAM-based RNG

As discussed earlier, the power-up bits generated from an SRAM suffer from low entropy as the majority of the SRAM cells are biased towards 0 or 1 due to process variation. In a standard SRAM array, around 80-90 percent of cells are stable (stable at either 0 or 1) [82]. As a result, we need a large SRAM array to create a true random number. It is often tempting to create an SRAM-based RNG with increased entropy, such that it can be well-suited to resource constrained devices in an IoT or CPS application. In this section, we describe why the initial increase of the entropy of an SRAM array using accelerated aging can create an adverse effect on the bits produced from a RNG while they are operating in the field. We also propose an approach to compensate for any degradation from aging in SRAM-based RNGs.

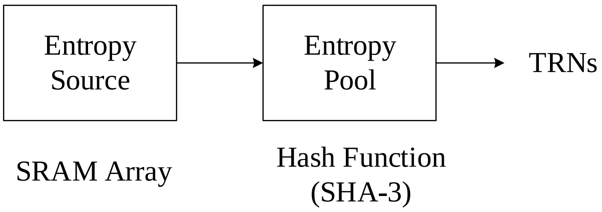


Figure 5.4: A typical RNG implementation [6, 7], which uses an SRAM memory.

5.2.1 Implementation of an SRAM-based RNG

There are two ways for implementing a random number generator (RNG), namely, non-deterministic and deterministic [7]. In a non-deterministic RNG, every bit comes from the physical random source and such a bit is completely random and unbiased. Deterministic RNGs use an algorithm to generate the sequence of true random bits starting with a seed (entropy source) which is random, but whose individual bits may display some bias. In this paper, we treat the SRAM as this entropy source, and then use a hash function to generate the true random number [89]. Figure 5.4 shows an implementation of the SRAM-based RNG, where the SRAM array provides the necessary entropy to create the random bits. The entropy pool behaves like an entropy extractor. A secure hash function (SHA-2/SHA-3) is used for extracting the entropy [20]. One can also use AES-CBC-MAC for such purpose [6]. Note that the size of the SRAM array should be large enough to provide the necessary entropy (e.g., entropy should be approximately 256 bits (or more) for a 256-bit random number).

The entropy, which refers to the disorder or uncertainty, of a source can be calculated using Shannon's formula [90]:

$$E = -\sum_i p(x_i) \log_2 p(x_i) \tag{5.1}$$

where $p(x_i)$ denotes the probability of observing pattern x_i. For a single bit binary source, the entropy can be calculated as:

$$E_b = -\{p(x_0) \log_2 p(x_0) + p(x_1) \log_2 p(x_1)\} \tag{5.2}$$

where $p(x_0)$, and $p(x_1)$ denotes the probability of occurring $x = 0$, and $x = 1$, respectively. As an SRAM array consists of multiple SRAM cells, where the power-up state of one SRAM cell is independent of others, we can compute the average entropy per bit using the following equation:

$$E_{b_{avg}} = \frac{1}{S}\sum_{i=1}^{S} E_{b_i} \tag{5.3}$$

where S represents the SRAM array size, i.e., the number of SRAM cells. To design an SRAM-based RNG, the entropy should be at least equal to or larger than the random bits generated from it.

5.2.2 Aging-Resilient SRAM-based RNG

An unbiased SRAM cell, which primarily contributes to the entropy of the SRAM array, may become biased due to aging from usage in the field. As a result the overall entropy from an SRAM array can reduce over time. In this section, we explain why pre-deployment initial aging (simple or accelerated) to increase the entropy of an SRAM array can have a negative impact on the randomness of the bits generated from a RNG over time. Instead, we propose an alternate strategy of controlled aging of the SRAM array at regular intervals to compensate for the loss of bias in normal usage and maintain SRAM entropy.

The entropy of an SRAM array results from the balanced cells, where the threshold voltages of transistor pairs is virtually equal at manufacturing and the power-up state is equally likely to be 0 or 1. However, more than 80% of the SRAM cells are usually biased towards either 0 or 1 due to manufacturing process variations. Kiamehr et al. [26] proposed a solution by aging an SRAM array just after manufacturing to increase the number of unbiased cells in the array. The scheme exploits NBTI aging to selectively increase v_{Thu} of the PMOS in one inverter by stressing the cell in the power-up state. It can thereby obtain, on average, more balanced cells and higher entropy. The approach is particularly effective because v_{Thu} shifts from aging are faster and larger in a newly manufactured part. However, this scheme creates a vulnerability when an SRAM is deployed in the field. As the other PMOS transistor still remains fresh after this initial aging, it can age at a much faster rate when random data is stored in the SRAM, quickly neutralizing any initial increase in entropy.

Figure 5.5 shows a simplified schematic of the SRAM cell (depicted in Figure 5.1.b) to analyze it's behavior under aging. We can model the output node capacitance by adding two large lumped capacitors, C_1 and C_2 at the node 1 (output of inverter 1) and node 2 (output of inverter 1),

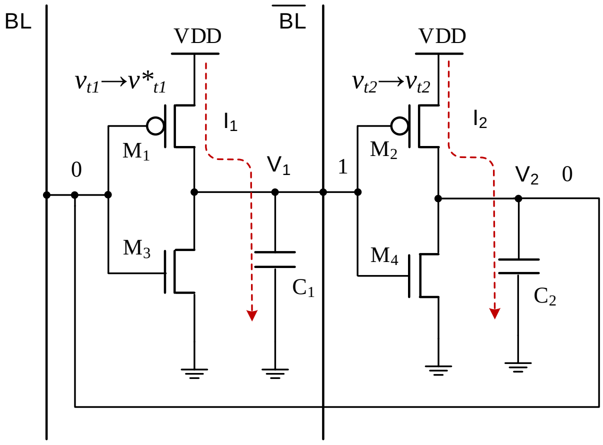


Figure 5.5: Simplified schematic of an SRAM cell to explain the power-up behavior.

respectively. For the simplicity, we have removed the access transistors (M_5, and M_6). At the design stage, both transistor pairs are balanced such that $M_1 - M_2$, and $M_3 - M_4$ have the same parameters, and all parasitics are the same for both inverters.

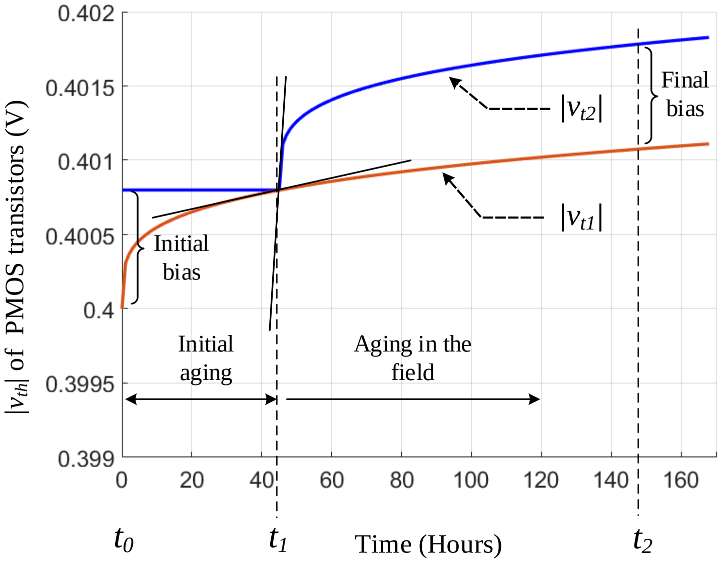


Figure 5.6: Controlled pre-deployment aging effect on SRAM cell.

Figure 5.6 shows the change in (the absolute value of) v_{th} due to post-deployment aging on a cell where controlled pre-deployment aging is used to balance the cell and increase cell entropy. We use HSPICE MOSRA [91] to simulate the aging. An SRAM cell is designed using $32nm$ PTM technology [92], where the threshold voltages (in absolute value) are $v_{t1} = 0.4$ and $v_{t2} = 0.408$. If we initially age the cell while storing 0, the transistor M_1, with the lower v_{th} (in magnitude) will experience aging due to NBTI and its threshold voltage increases to $v^*_{t1} (> v_{th})$ (see Figure

5.5). After the initial aging (t_1), the v_{th} for both the transistors will become equal and the SRAM cell becomes unbiased. However, once the chip is used in the field, the SRAM sees random data. To simulate this part, we age the SRAM cell with random data. At time t_1, the rate of v_{th} change for both the transistors is not equal. The transistor M_2 ages much faster, and the initial bias quickly returns. As a result, any initial compensation becomes ineffective.

We next analyze how the aging with the power-up state affects the bias of an SRAM cell. Assume that initially the threshold voltage of M_2 (v_{t2}) is higher than the threshold voltage of M_2 (v_{t1}) (see Figure 5.5). For simplicity, the effect of noise is omitted in the discussion. The cell will power-up with 0 as M_1 will quickly reach to saturation, and the node 1 (V_1) will pull up to VDD. If we keep this state, transistor M_1 will be NBTI stressed as its gate is negative and v_{t1} will be increased in magnitude to v_{t1}^*. On the other hand, transistor M_2 will remain fresh. If we repeatedly power up the SRAM array and hold the state each PMOS transistor will age alternatively and remain unbiased if the v_{th}s of both the PMOS transistor's are the same. If they are not, v_{t1} will be increased incrementally to reach $v_{t2} (> v_{th1})$ (or vice versa), and the cell will then stay unbiased.

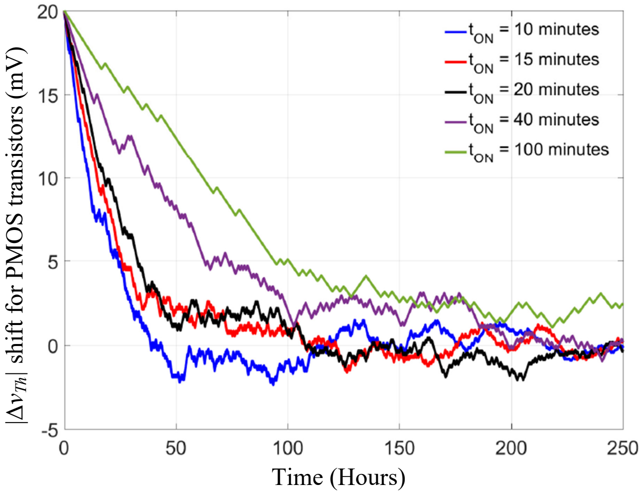


Figure 5.7: Simulation for Δv_{th} shift for the PMOS pairs in periodic aging.

Noise can play an important role to create bias in SRAM cells if we perform aging with the power-up state. Stressing a PMOS transistor for long time can create a significant shift in its v_{th}, which can create an increased bias for an SRAM cell. We need to carefully choose aging duration (t_{ON}) such that no significant bias is created. We next report on experiments that analyze how quickly the v_{th} of both PMOS transistor converge. HSPICE simulation is performed, where We choose $v_{t1} = 0.042V$ and $v_{t2} = 0.04V$, and keep the threshold voltages for the NMOS transistor the same for this experiment. Gaussian noise with $\mu = 0$ and $\sigma = 15mV$ (at node 1 and 2 in Figure 5.5) to emulate the power-up behavior. Figure 5.7 show the simulation results for Δv_{th} of PMOS transistors versus aging time with different t_{ON} values. A smaller t_{ON} value results in the convergence of both the threshold voltages. We can select this value for compensating the degradation caused during the normal operations of an SRAM chip.

5.3 Experimental Results from Silicon

This section analyzes the entropy of an SRAM array and discuss the resiliency of the random bits generated for a SRAM-based TRNG under aging compensation. We have selected a commercial off-the-shelf SRAM memory (Microchip 23A640-I/SN - SPI Bus Low-Power Serial SRAM) to perform the experiment. The size of the SRAM chip is 64K bits. Figure 5.8 shows the experimental setup for reading the power-up state of the SRAM chip. A voltage shifter (Texas Instruments PCA9306 Dual Bidirectional I^{2}C Bus and SMBus Voltage-Level Translator) is used to interface the Raspberry Pi with the SRAM.

5.3.1 Entropy Analysis

Initial aging after manufacturing can increase the overall entropy of an SRAM array, however, over time it can degrade in use and adversely affect the random numbers generated from an SRAM array. This section demonstrates the reduction of entropy from an SRAM array when it gets aged in the field using silicon data. The experiment is conducted at the room temperature.

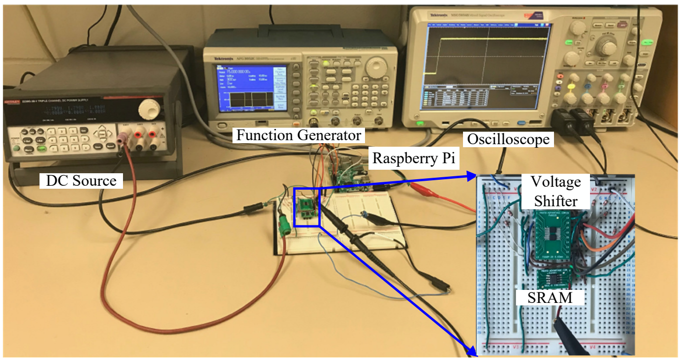


Figure 5.8: Experimental setup to measure SRAM power-up states.

Figure 5.9 shows the change in average entropy per bit over time. We measure the number of 0s and 1s, and thereby estimate the probability of observing a 0 and 1, at each address location by performing repeated (100) power-up cycles. This allows us to calculate the average entropy per bit (using Equation 5.3) for the SRAM. Initially, three new SRAM chips were aged for six days at room temperature using periodic aging. We select a controlled aging interval of 15 minutes. We observe that, as expected, the entropy increases quickly initially and then starts saturating. The SRAM chips are then kept on the shelf for few days. Due to some partial recovery of v_{th}, the average entropy per bit decreases during this period. We now emulate the behavior of usage in the field by aging the chip with random patterns, and measure entropy once in a day. We observe a rapid decrease of the entropy which was initially gained through compensation. Over time this starts saturating, but at levels even below the initial entropy. Note that all the three SRAM chips follow a similar trend.

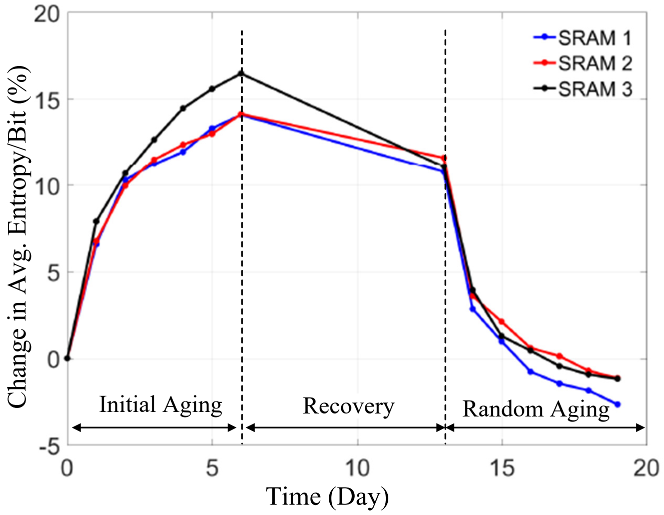


Figure 5.9: SRAM entropy versus time.

Figure 5.10 shows the change in average entropy per bit by using our proposed method. In this experiment, we use two new SRAM chips and the average entropy per bit is calculated by measuring 100 power-up states. First, we measure the entropy of the new SRAM chip. To emulate the usage behavior in the field, these SRAM chips have been aged using random patterns for one day and entropy have been measured. We then perform the periodic aging to compensate the degradation. From the Figure 5.10, we can observe the entropy start to drop after aging with random patterns. The periodic aging for 3 hours boosts the entropy to its initial value. The dotted

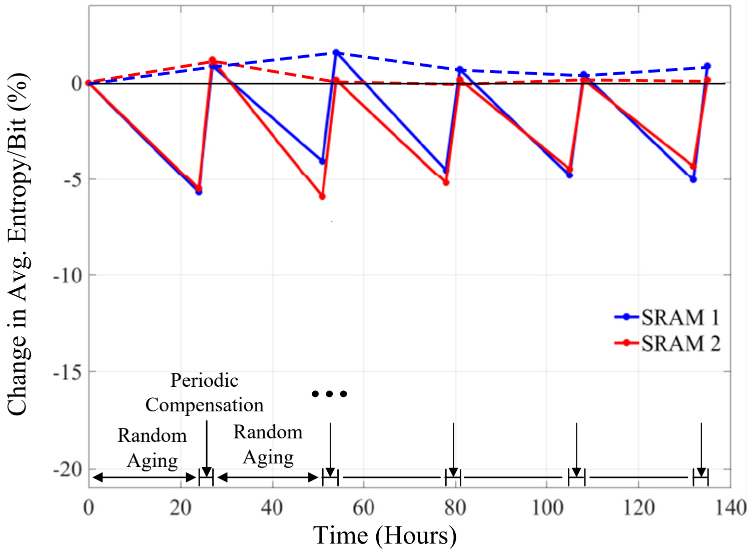


Figure 5.10: Periodic compensation by using our proposed method.

line shows the overall trend of entropy change. The silicon data from these two SRAMs clearly shows that our proposed method preserves the initial entropy of the SRAM chips.

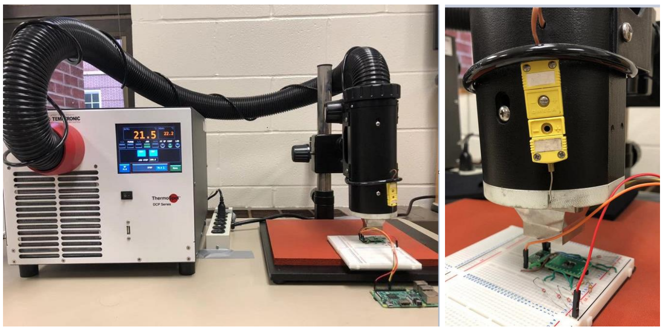


Figure 5.11: The experimental setup for temperature variation using ThermoSpot direct contact probe system [8].

5.3.2 Attack Analysis

An attacker can bias an SRAM-based TRNG with extreme temperature and voltage variations. These attacks can be applied to any SRAM TRNGs to break a cryptographic protocol, and explored extensively by the research community [20,22,93,94]. We, however, explore the same to completeness of this paper. We have used four SRAM chips to validate the average entropy variations. These four SRAMs are of two groups – one group is new and the other group have been aged with random patterns. This ensures that both scenarios that an adversary can perform attacks new TRNGs, which are just deployed in the filed and old one, which are in use for some time. First, we analyze the

Table 5.1: $E_{b_{avg}}$ of SRAM chips under different temperatures.

Temperature (^0C)	SRAM 1 (New)	SRAM 2 (New)	SRAM 3 (Old)	SRAM 4 (Old)
0	0.0853	0.087	0.0862	0.0866
25	0.0899	0.0905	0.0886	0.089
50	0.0993	0.107	0.0978	0.1003
85	0.1121	0.119	0.1023	0.1123

entropy variations on temperature, which is achieved by using a ThermoSpot direct contact probe system (see experimental setup in Figure 5.11). This system is an industry standard benchtop temperature cycling system, used for accelerated aging [8]. The device can operate to control temperature ranging from -65°C to 175°C, with a transition rate of less than 35 seconds over 25°C to -40°C.

Average entropy per bit ($E_{b_{avg}}$) of an SRAM TRNG is calculated using Equation 5.3. Table 5.1 shows the result of average entropy per bit of SRAM under different temperatures. The first column represents the experimental temperature and the following columns represent the corresponding value of $E_{b_{avg}}$. The average entropy reduces with the temperature as expected, and is also reported in prior work [22].

The second experiment is conducted by varying the supply voltages, and the result is presented in Table 5.2. The first column represents value of supply voltage, whereas the following columns represent corresponding value of $E_{b_{avg}}$. Based on the data of these two table, We observe a very little change in average entropy under different supply voltages. We observe similar trend for all the SRAM chips.

Table 5.2: $E_{b_{avg}}$ of SRAM chips under different supply voltages.

Voltage (V)	SRAM 1 (New)	SRAM 2 (New)	SRAM 3 (Old)	SRAM 4 (Old)
1	0.0876	0.0874	0.0862	0.0878
1.5	0.0888	0.0885	0.0873	0.0881
1.7	0.0899	0.0898	0.088	0.0889
1.8	0.0901	0.906	0.0885	0.0891
1.9	0.0902	0.907	0.0892	0.0894

Note that an attacker can also use aging maliciously to influence the random numbers generated from traditional SRAM-based TRNGs. However, our proposed aging compensation on a TRNG makes it more resilient compared to traditional SRAM-based one as we age the cells in both directions. As a result, the aging degradation becomes smaller in successive times as the aging degradation follows an exponential curve [95–97]. An attacker can influence our proposed TRNG much less as compared to the traditional design.

5.4 Conclusion

In this paper, we have presented an approach to preserve the entropy of the power-up states of an SRAM array during deployment. We have shown that SRAM entropy can decrease in deployment from device aging while in use. Consequently, SRAM based RNGs need to manage entropy to ensure high quality random numbers. Unfortunately, any pre-deployment increase in entropy through controlled aging as suggested in prior research cannot be sustained in deployment. Our experimental result shows that we can preserve the initial entropy of COTS SRAM chips using periodic compensation by repeatedly powering up the SRAM chip and then holding its power-up state.

Chapter 6

Recycled ICs Detection based on Power-Up State of SRAM

In this chapter, we propose a new and highly effective approach for detecting recycled ICs by exploiting the power-up state of on-chip SRAMs to evaluate the age of the chip. Our methodology does not require the introduction of any special aging detection circuitry, nor the recording and saving of historical circuit performance data as a reference to detect degradation from use. Instead, we exploit the novel observation that in a new unused SRAM, an equal number of cells power up to the 0 and 1 logic states, and also that this distribution becomes skewed in time due to aging in operation. It is also low cost since does not require any special test equipment. The overall chapter can be divided by two main parts. The first part we proposed a novel method that detect the recycling SRAM chips by exploiting the overall ratio of 1s of power-up state of SRAM. We try to validate our proposed method based on the SRAM that have been manufactured by new technology. We present experimental results using commercial off-the-shelf SRAM chips to validate the effectiveness of the proposed approach. The second part, we mainly focus on the SRAM which have been fabricated by very mature technology node and proposed a significantly improved and innovative method to detect recycled SRAMs which additionally modulates the power-up ramp rate to minimize the impact of systematic biases in the cells, thereby amplifying the impact of NBTI aging. The silicon experiments presented here indicate that our proposed new method can accurately detect most recycled SRAM chips across a range of old and new technologies.

6.1 Detecting Recycled SoCs by Exploiting Aging Induced Biases in Memory Cells

The problem of old recycled integrated circuits (ICs) being supplied and sold as new continues to grow due to the lack of efficient detection and avoidance techniques. The entry of these ICs into the critical global infrastructure (defense, aerospace, transportation, medical, etc.) can result in system and security failures with potentially serious consequences for societal well being. In this subsection, we propose a novel approach for detecting recycled ICs with the help of the power-up state of one or more SRAMs available in the chip. Our proposed solution does not require any hardware modification to the existing design, and can be applied to a wide variety of SoCs that have SRAM based memory, including FPGAs. This solution can be applied to both, ICs already circulating in the market, as well as those to be manufactured in the future. The proposed approach is simple, effective and low-cost, and requires minimal test support: a capability to read out the initial power-up state of the on-chip SRAM. Our experimental results show that we can accurately detect if the IC has been used in operation for a period as little as few days.

Our new approach exploits the degradation in device threshold voltages caused by stress due to aging in operation. Unfortunately, identification of a chip as recycled based on parameter shifts over time critically requires the initial parameter values for a new and unused part against which any degradation in use can be evaluated. Prior approaches suffered from the lack of an accurate reference parameter due to the significant process variations that are experienced in IC manufacturing. Such starting differences in circuit parameters among new parts can often exceed any changes from aging in operation. This makes recycling detection virtually impossible, except for the highly unlikely case where the target parameters for individual ICs were measured at manufacture and are still available when the part is to be evaluated many years, even decades, later.

The critical innovation in our proposed approach is that it does not need such a saved reference. Instead, it exploits two key properties of SRAMs: (i) that individual SRAM cells are designed to be completely symmetric in layout (so as to maximize noise margins), and therefore completely unbiased with respect to the logic state they acquire at initial power-up, and (ii) any bias that is introduced by the random manufacturing variations can be in either direction with equal probability, i.e., any imbalances in the memory cells caused by process variations result in an equal likelihood of the cell being biased to power up in either the 0 or 1 logic state. Consequently, in a newly manufactured SRAM, at initial power-up, the percentage of 1s in the memory cells should be the same as the 0s, both very close to 50% (typically well within one percent) because of the statistically large number of cells. This initial 50% statistic, which holds for all new SRAMs, forms a reliable base reference for a new memory.

The 50% number degrades over time due to asymmetric shifts in the SRAM cell transistor threshold voltages from Bias Temperature Instability (BTI), which is mostly observed to impact PMOS transistors as Negative BTI (NBTI) in traditional bulk technologies [70, 71]. (The discussion here equally applies to PBTI, which is also experienced by the NMOS transistors in some technologies.) Observe that it is virtually impossible for every individual cell in the memory to store a 1 and a 0 logic value for exactly the same total time during an operating life of arbitrary duration, and thereby always retain its initial bias in use. Any imbalance in this storage time results in asymmetric shifts in transistor threshold voltages in the cell from NBTI aging which causes changes in the cell power-up bias. Skewed data patterns in functional memory usage further ensure that these changes from operational stress with the cells result in a move away from the initial balanced 50% 1 and 0 cell bias. For example, Wei et al. [87] have reported that the ratio of 1s to 0s in most files is less than 50%. This number is even lower, at only 20 to 35%, for system files. In addition, many SRAMs, such as the block RAMs (BRAMs) in Xilinx FPGAs, are initialized to 0 [98], which again increases the fraction of time the memory cells are stressed in the 0 state. The detection of recycled ICs in the proposed approach is based on this inevitable shift in the percentages of 1s and 0s in the power-up state as an SRAM is used. We validate our methodology with results from ongoing silicon experiments in our effort to collect long term data.

The rest of the section is organized as follows. Section 6.1.1 introduces the modeling of power-up state for an SRAM, and how it is impacted by the aging. Section 6.1.2 discusses the our proposed for detecting recycled SoCs. Experimental results are results are given in Section 6.1.3. Finally, we conclude entire section in Section 6.1.4. The majority of works in this chapter was published initially in [83].

6.1.1 Effect of Threshold Voltage Variation on the Power-Up State

The power-up state of an SRAM cell depends on the threshold voltages (v_{th}) of the MOS transistors. This section presents the effect of threshold voltages on the power-up state of an SRAM cell. Note that an SRAM array consists of multiple SRAM cells, and each cell consists of six transistors shown in Figure 5.1. The four transistors (M_1, M_2, M_3 and M_4) form a bistable latch to store 1-bit of data. BL, and $\overline{BL}$ provides the access to the latch through M_5 and M_6 transistors.

When an SRAM array is powered-up, initially each individual memory cell randomly acquires either a logic 0 or logic 1 value. During design, the MOS transistors in each matched pair ($M_1 - M_2$, $M_3 - M_4$ and $M_5 - M_6$ in Figure 5.1(b) are carefully made completely identical, including all their layout related parasitic components. The perfect symmetry of the memory latch in the SRAM cell maximizes noise margins during operation. Consequently, each SRAM cell should ideally have a 50$\%$ chance of acquiring either logic 0 or logic 1 when powered up, the actual value decided by random unbiased noise. However, in practice most MOS transistor pairs will not be perfectly matched due to random manufacturing process variations, the most significant of which, in the context of this discussion, are the small variations in the threshold voltages in each MOSFET. The v_{th} differences in the PMOS and NMOS transistor pairs can either cause a cell bias in the same direction or in opposite directions. The net imbalance decides the overall bias towards either 1 or 0 at power-up. A larger net imbalance in the transistor pairs result in a more skewed SRAM cell. If the net v_{th} difference is small, the power-up values may be still be somewhat random, with a bias towards either 0 or 1. On the other hand, for relatively large net v_{th} imbalances, the power-up state will be stable and always the same over multiple power-up cycles.

The effect of the transistor threshold voltages on the power-up behavior of an SRAM cell can be seen in more detail with the help of Figure 5.1(b). The SRAM cell is basically two inverters connected in a ring. The output of one inverter is connected to the BL. Similarly, the output of the second inverter is connected to $\overline{BL}$. We model the output node capacitances by adding two lumped capacitors, C_1 and C_2, at node 1 (output of inverter 1) and node 2 (output of inverter 2), respectively. The effect of the transistors (M_5, and M_6) on the power-up state can be mostly ignored as they remain off during the power-up time. At the design stage, all transistors are balanced so that $M_1 - M_2$, and $M_3 - M_4$ have the same parameters, and all parasitics are same for both inverters. Assume (for simplicity) that after manufacture, the threshold voltage changes only for M_1 due to process variation, and it's threshold voltage increases (in magnitude) to v_{t1}^*. Initially, the threshold voltages for both M_1 and M_2 were identical, i.e. $v_{t1} = v_{t2}$. Clearly after manufacturing, $v_{t1}^* > v_{t2}$. If we consider a relatively fast ramp rate at the power supply during the power-up time, this PMOS transistor mismatch will decide the state of the SRAM cell [76]. As the v_{th} of M_1 is larger (in magnitude) than M_2, M_2 with the smaller threshold magnitude will turn on first, forcing it's output high and the complimentary M_1 output low. The cell will power-up to 0.

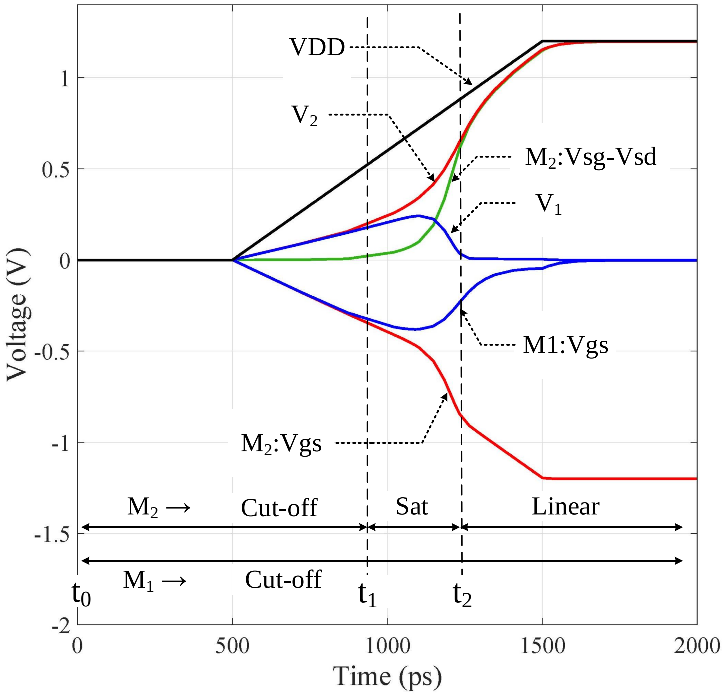


Figure 6.1: Timing diagram of internal nodes of an SRAM Cell during the power-up.

Figure 6.1 shows the timing diagram of the potentials at different nodes of a single SRAM cell using the Synopsys HSPICE simulation tool. $32nm$ bulk Predictive Technology Model (PTM) is selected for the simulation. The nominal threshold voltages (v_{th}) of NMOS, and PMOS transistors are 0.42252V and -0.41174V, respectively. To simplify the situation, the process variation in NMOS transistor parameters have been ignored. The new threshold voltage of M_1 (v^*_{th1}) has been increased 20% (in magnitude) from its nominal value. Initially, the potentials at node 1 (V_1) and node 2 (V_2) rise at the same rate. The currents, I_1 and I_2, result from the subthreshold leakages of M_1, and M_2. The potentials V_1 and V_2 are of the same order. At time t_1, transistor M_2 goes to saturation as $V_{sg} - V_{sd} < |v_{th}|$ and $V_{sg} > |v_{th}|$. On the other hand, transistor M_1 still remains in the cutoff region as $V_{sg} < |v_{th}|$. This happens due to $v^*_{t1} > v_{t2}$. We observe a sharp rise in V_2, as M_2 is in saturation and can provide much larger current ($I_2 \gg I_1$). Finally at time t_2, transistor M_2 goes to the linear region as $V_{sg} - V_{sd} > |v_{th}|$ and $V_{sg} > |v_{th}|$, and we observe a different slope in V_2. Note that transistor M_1 never gets out of the cut off region.

6.1.2 Proposed Approach for Detecting Recycled System on Chips

Detection of used and recycled SoCs can be performed effectively by observing either the percentage of 1s ($\%1s$) or percentage of 0s ($\%0s$) in the power-up state of an on-chip SRAM. As discussed earlier, the requirement for a reference parameter from the chip in the unused state, which is generally needed to make a decision whether the chip recycled or not, is not necessary. This is because the $\%1s$ and $\%0s$ are known to virtually identical in a new chip, typically to well within a percent. Detection can easily be carried out by observing even a small change in $\%1s$ from this reference value of 50$\%$. In this section, we will provide a more in-depth analysis of our proposed counterfeit detection approach, particularly with regard to how the SRAM start-up state is impacted by process variations and device aging in operation.

Effect of Process Variation on Transistor Threshold Voltages

Process variations (PV) cause the threshold voltage of a transistor to vary from its nominal value [99, 100]. This variation has two components – (i) systematic variation and (ii) random variation [101]. Systematic variation is the variation among different dies (chips or regions in chips), and may resulted from the imperfections in the lithographic process (mask alignment errors, lens aberrations, etc.), and small changes in the environmental conditions during the fabrication. It moves the threshold voltage of all transistors of chip in one direction. On the other hand, random process variation is the variation among the MOS transistors within a die. In advanced technology nodes, this arises from factors such as the random fluctuations in the numbers of dopant atoms in the channel, gate line edge roughness and surface orientation [102–104]. Random variations are commonly modelled using the zero mean Gaussian process [101].

Figure 6.2 shows plots of the resulting variations, where the mean of the random variation within each chip is determined by the systematic variation. The threshold voltage of a transistor can be represented as: $v_{th} = v_{th0} \pm \Delta v_{thS} \pm \Delta v_{thR}$, where Δv_{thS} and Δv_{thR} represent the change in threshold voltage due to systematic and random variations, respectively. The threshold voltage difference between the two PMOS and NMOS transistors in a SRAM cell (see Figure 5.1(b) will result from the random process variation, as the systematic variation moves the v_{th} for all the

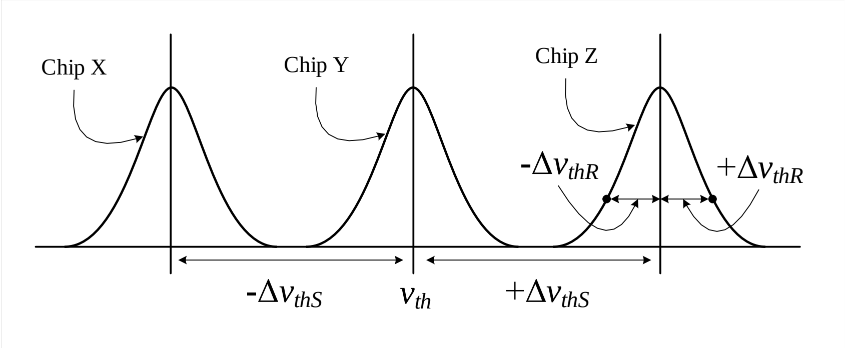


Figure 6.2: Systematic and random process variations.

transistors in a chip in the same direction. This can be described as:

$$\begin{aligned}\Delta v_{th} &= v_{th1} - v_{th2} = (v_{th0} + \Delta v_{thS} + \Delta v_{thR1}) \\ &\quad -(v_{th0} + \Delta v_{thS} + \Delta v_{thR2}) \\ &= \Delta v_{thR1} - \Delta v_{thR2} \qquad (6.1)\end{aligned}$$

where v_{th0}, Δv_{thS}, and Δv_{thR} represent nominal threshold voltage, systematic and random v_{th} variations, respectively. From Equation 6.1, we can conclude that the distribution for Δv_{th} will be zero mean Gaussian, since the distribution for random process variation is zero mean Gaussian. This reveals the interesting fact that there is a 50$\%$ probability of v_{th1} is greater than v_{th2}, and vice versa.

Effect of Aging on the Power-up State

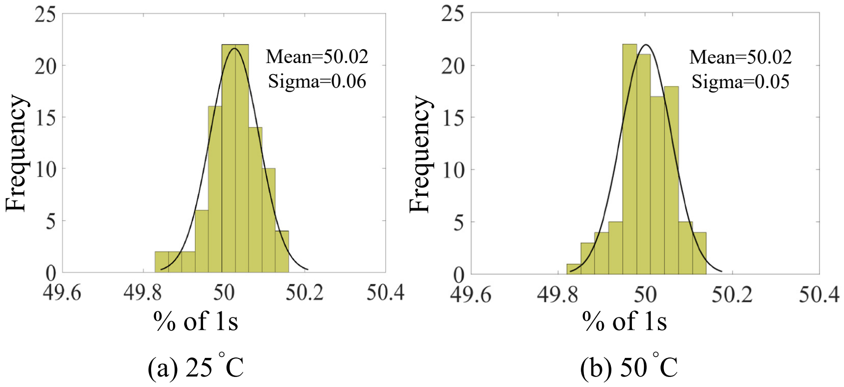
The threshold voltage of a transistor increases under operational stress when the chip is used in the field. This is also true for an SRAM circuit, when it is used for storing data. One of the main aging phenomena in ICs is negative bias temperature instability (NBTI), which occurs in PMOS transistors when they are negatively stressed [70, 71]. Interface traps are created at the Si-SiO_2 interface of PMOS transistor when its gate is pulled down to logic 0. Releasing the stress can achieve some but not complete recovery. As a result, the threshold voltage (v_{th}) of PMOS transistors increases over time [72]. This increase tends to saturate over a period of months, and becomes minimal after 5-10 years in use. In summary, a PMOS transistor ages when it is turned on (the input is at logic 0) and relaxes when it is turned off (the input is logic 1). NMOS transistors experience much smaller threshold shifts from PBTI aging, although that may change at advanced technology nodes. A different aging phenomenon in CMOS circuits is hot carrier injection (HCI). [73,74]. Some high energy electrons can attain sufficient energy when the transistor is conducting (on) to get trapped in the Si-SiO_2 interface near the drain terminal due to the lateral gate electric field. NMOS transistors are primarily affected by HCI because of higher carrier mobility, whereas it has very little effect in PMOS transistors [75]. Observe, however, that HCI occurs when there is current flow in the transistor channel. In practice, the impact of HCI in SRAM cells is minimal, and can be ignored, because the transistors in memory cells are mostly non-conducting and experience much less switching activity than logic.

The effect of aging on the power-up behavior of an SRAM cell can be explained using Figure 5.1(b). To begin with, we ignore process variation and assume all the transistor pairs possess the same device parameters. As a result, the threshold voltages of (M_1 and M_2) have same value ($v_{t1} = v_{t2}$). Similarly (M_3 and M_4) are identical. (We ignore any PBTI aging in the NMOS transistors.) Assume that for some initial period, the cell contains 1 ($BL = 1$, and $\overline{BL} = 0$), which sets the internal nodes $V_1 = 1.2V$, and $V_2 = 0V$. Consequently, the transistor M_1 will experience aging due to NBTI (as its V_{gs} is negatively stressed) and its threshold voltage will increase in magnitude over this time to $v^*_{t1}(> v_{th})$. Based on discussion in the previous subsection, this SRAM cell is now biased and will power-up with logical 0 ($V_1 = 0$ and $V_2 = 1$) as threshold voltage of M_1 becomes larger (in magnitude) than M_2 after aging. *If we age the cell with 0, it will power up with 1 (and vice versa), for a perfectly balanced SRAM cell.*

Effect of Noise on the Power-up State

The power-up state of an SRAM can be affected by the noise, and percent of 1s in the power-up state of an SRAM array can vary from the mean (i.e., 50%). We perform an experiment to analyze the effect of noise on the power-up state. A commercial off-the-shelf (COTS) SRAM (Microchip 23A640-I/SN: SPI Bus Low-Power Serial SRAM) chip is powered up 100 times and its power-up states are measured. Figure 6.3 shows the histogram plot of the percentage of 1s in the power-up states. We also perform the same experiment for different environmental conditions to determine the effect of noise in the power-up states.

We observe a Gaussian distribution for percentage of 1s with mean (μ) of approximately 50% for two different (25°C and 50°C) environmental conditions. The standard deviation (σ) also varies



(a) 25 °C

(b) 50 °C

Figure 6.3: Effect of noise on the power-up state of an SRAM array.

slightly at different environment corners (see Figures 6.3.a and 6.3.b). The value of σ is 0.06, and 0.05 when we perform the experiment at 25°C and 50°C, respectively. We can conclude from this experiment, that the noise has little effect on the power-up behavior of an SRAM array.

Proposed Approach based on Memory Power-up State

As the percentages of 1s and 0s in the power-up state of an SRAM array are virtually identical in a new chip, we can detect recycled ICs using this information. When a chip ages, the mean value of percentage of 1s (or percentage of 0s) shifts over time, and a decision can be made based on this shift. Note that this proposed solution does not require any hardware modification in any way to an existing design, and thus can be applied to a wide variety of SoCs, which contain SRAM memory. This approach is designed for detecting old chips, those are already circulating in the market. It is not necessary to have any knowledge of the inner details of the circuit to determine whether a chip is recycled or not.

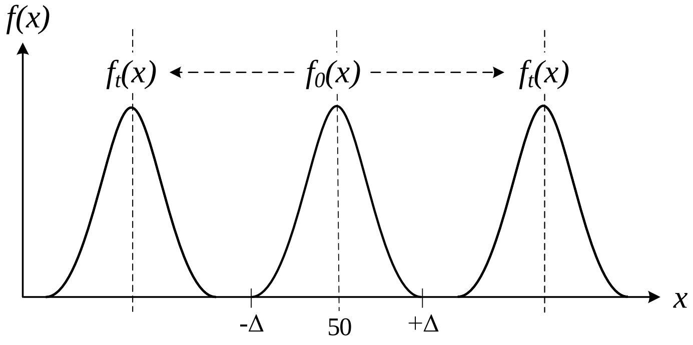


Figure 6.4: The probability density functions of %1s over time.

Figure 6.4 shows the distribution of $\%1$s (represented as $f()$ with variable x) for a new chip and old chip, respectively. The distribution for a new SRAM chip ($f_0(x)$) is centered near 50$\%$ as the random process variation is a zero mean Gaussian process. There is an equal probability that an SRAM cell will power up either 1 or 0. Due to the noise, the $\%1$s can vary slightly and we observe a Gaussian distribution (see Figure 6.3). During normal operation, the data stored in an SRAM chip causes biasness and can age an SRAM cell opposite towards its stability. Moreover, majority of the on-chip SRAMs are initialized to 0 (or 1) until the memory is overwritten with a random data. As a result, the distribution of $\%1$s for an old SRAM chip ($f_t(x)$) can either shift to right or left. We can clearly identify a recycled chip if the $\%1$s distributions, $f_0(x)$ and $f_t(x)$, do not overlap each other. Misprediction (i.e., recycled ICs identified as new and vice versa) may arise if these two distributions overlap. Note that the impact of noise can be minimized while considering a large SRAM array. The spread of $\%1$s distribution from a COTS SRAM with 64K bits is very small (i.e., $\sigma = 0.06$).

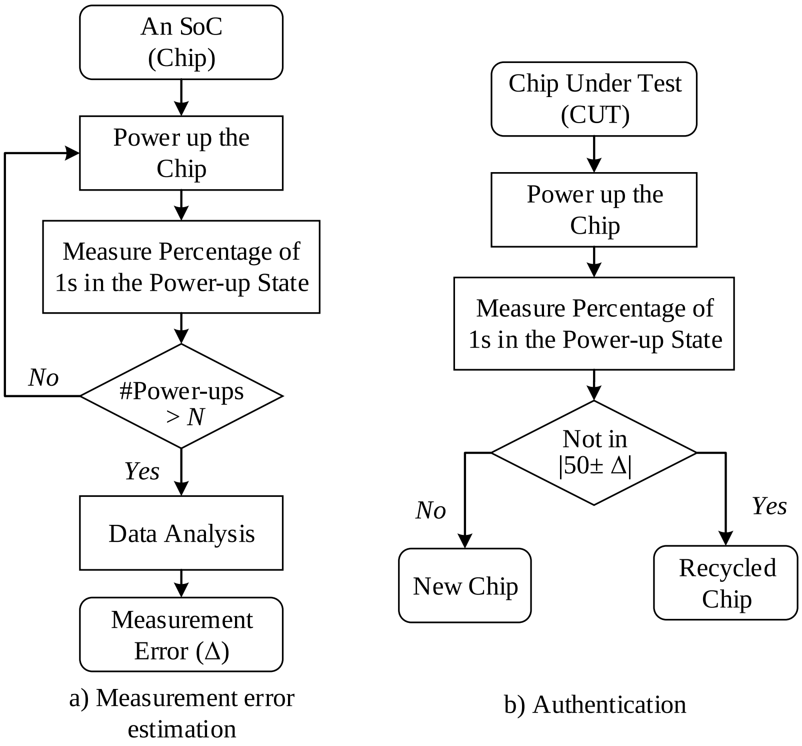


Figure 6.5: Proposed approach for detecting recycled ICs using memory power-up state.

The proposed approach for detecting recycled ICs using the memory power-up state is illustrated in Figure 6.5. Note that it is not necessary to know the value of measurement error (Δ), when a chip is used more than a week (see silicon results in Section 6.1.3). However, it is necessary to determine

Δ due to the noise, which impacts the power-up state of an SRAM, and this process is depicted in Figure 6.5.a, when the chips are manufactured. The process of measuring Δ is described as follows:

- *Step-1*: The power-up state of the SRAM array is recorded after powering up an SoC.
- *Step-2*: The percentage of 1s ($\%1s$) is measured from the recorded power-up state.
- *Step-3*: *Step-1* and *Step-2* are repeated N (large enough for statistical inference) times. We perform 100 power-ups to plot the distribution, which is shown in Figure 6.3.
- *Step-4*: Data analysis is performed to measure Δ from the distribution. We can choose 3σ as the measurement error Δ, and record this value for future.

It is recommended that this process is repeated with more than one SoC to accurately measure the effect of noise. In addition, measurement at different environmental conditions (e.g., $50°$C) helps up to measure Δ, such that accuracy of identifying a chip is recycled is increased. The effect on the noise can also be minimized using considering larger size memory.

The authentication process of determining a chip being recycled is a straight forward process, and shown in Figure 6.5.b. The Chip Under Test (CUT) is powered up and its power-up state is recorded. The percentage of 1s in the power-up state is calculated. If the $\%1s$ does not fall within $50 \pm \Delta$, the chip can be identified as recycled chip; otherwise, it is new. Note that it is not necessary to have the information of Δ during authentication. As the shift of the distribution (e.g., shift in mean μ) is much larger than Δ (see the silicon data in Section 6.1.3), a decision can be made just observing this shift.

6.1.3 Silicon Experimental Results

This section presents a detailed analysis of the effect of aging in the power-up state of SRAM arrays. We have conducted experiments using two different types of commercial off-the-shelf (COTS) SRAM memories to demonstrate the effectiveness of our proposed approach for detecting recycled ICs. These are Microchip 23A640-I/SN [105], and 23K640-I/SN [106] SPI Bus Low-Power Serial SRAM memories. The total memory capacities of both SRAM chips are 64K bits .

Figure 6.6 shows the experimental set-up for measuring the power-up state of these Microchip SRAMs. The supply voltage for chip 23A640 is 1.8V. It is thus necessary to use a voltage shifter to interface with the Raspberry Pi, which is programmed to collect the power-up states

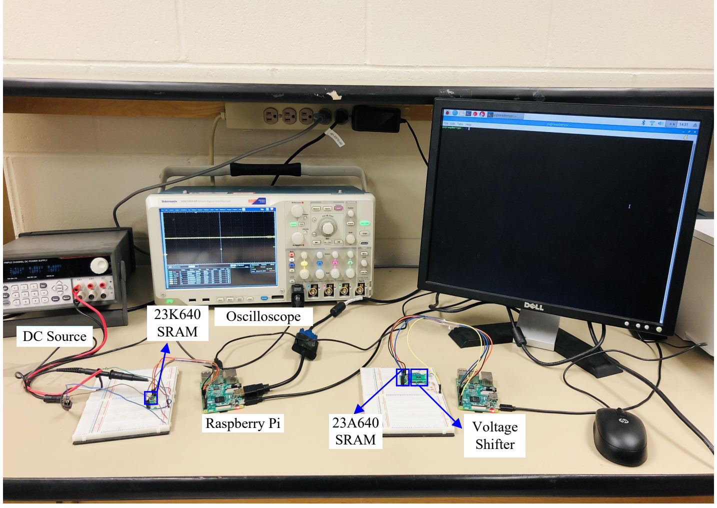


Figure 6.6: Experimental set-up to measure SRAM power-up states.

of these SRAMs. We use Texas Instruments PCA9306 Dual Bidirectional I^{2}C Bus and SMBus Voltage-Level Translator [107] for this purpose. On the other hand, we can directly interface the Raspberry Pi with Microchip 23A640 due to its supply voltage requirement is 3.3V. The first set of experiments are conducted at the room temperature. *Note that new SRAM chips must be used at the start of each new experiment to ensure a starting 50% distribution of 1s and 0s.*

Figure 6.7 shows the distribution of 1s ($\%1s$) at the power-up state for a Microchip 23A640 memory chip (denoted as Chip 1). The power-up states of this SRAM chip are measured 100 times, and the $\%1s$ distributions are plotted. The chip is aged after loading and then holding all 0s in all the memory locations. After 3 days of aging interval, the power-up states of the SRAM chip are collected 100 times, and the $\%1s$ distributions are plotted. From the figure, we observe that the initial distribution for the new chip is quite tight and centered almost exactly at 50% (e.g., $\mu = 50\%$). The distribution significantly shifts rapidly towards the right once the chip is aged. The non-overlapping property of the new and aged SRAM distributions indicates that we can reliably detect recycled ICs by observing $\%1s$ in the SRAM power-up states.

The initial reference value (approximately 50% of 1s in the power-up state) shifts over time due to asymmetric shifts in the V_{th} of transistors in SRAM cells from Bias Temperature Instability (BTI). It is thus necessary to study how the mean of $\%1s$ distributions shift over time to ensure the accurate detection of recycled ICs. We have analyzed four Microchip SRAMs (two 23A640 and two 23K640) to reliably evaluate this shift. Figure 6.8 shows the change in mean of the of $\%1s$ distribution.

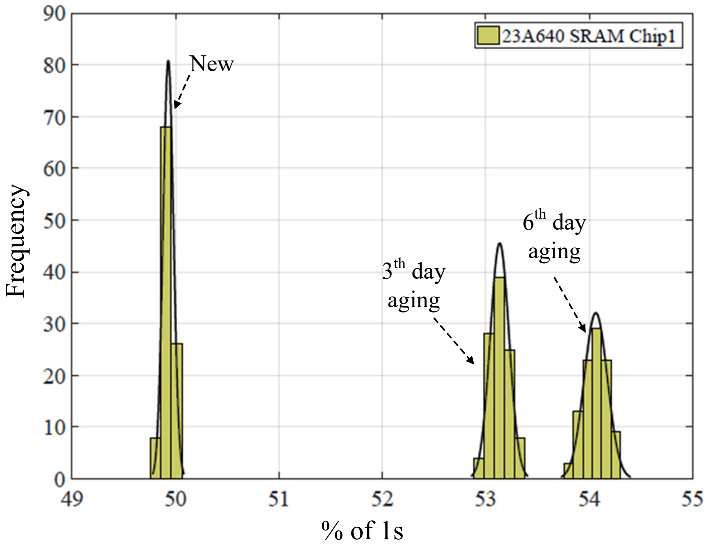


Figure 6.7: The distribution of $\%1s$ on the power-up states for Microchip SRAM chips.

Each point on this plot is bounded by $\pm 3\sigma$ over the mean (μ) of the $\%1s$ distribution. The μ and σ values are computed over 100 measurements of the power-up states after every one day of aging. The mean of $\%1s$ distribution changes around 2% after one day of aging. The mean changes at an accelerated rate over the early period of aging. After a week of aging, we have observed a shift of around 4% for all the SRAM chips. We also observe a minor increase in the standard deviation once the chip is getting aged. However, these 3σ values are much smaller than the change in μ values.

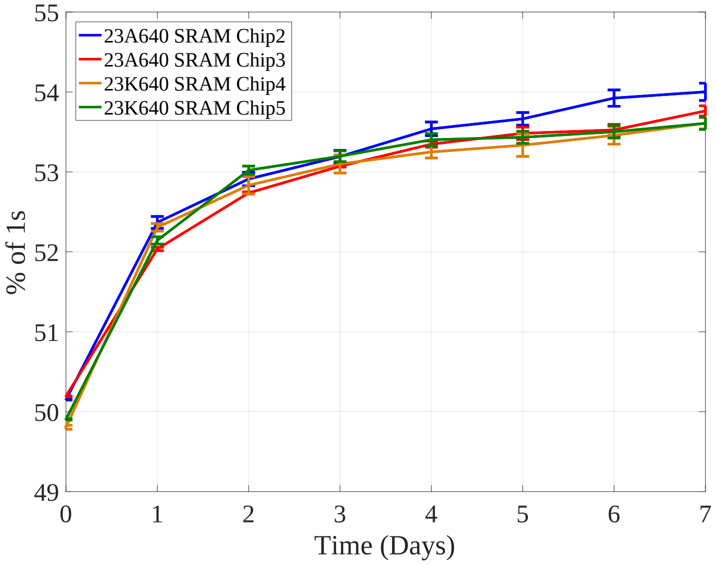


Figure 6.8: The shift of mean of $\%1s$ distribution over time.

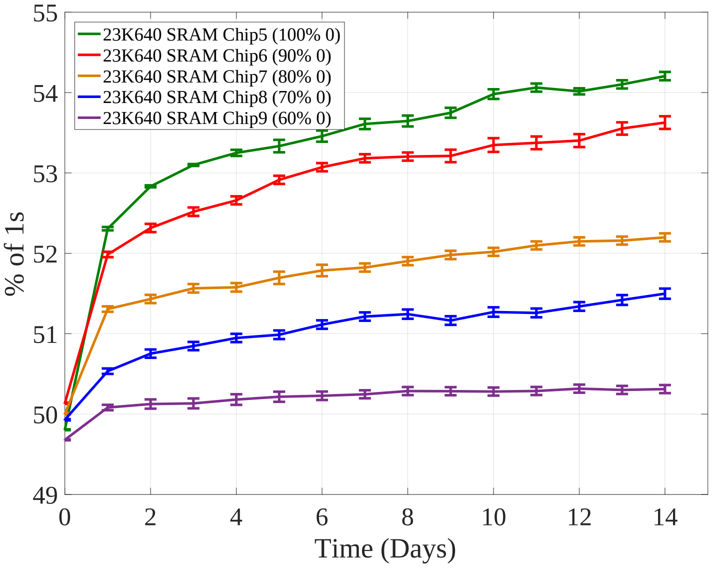


Figure 6.9: The shift of mean of $\%1s$ distribution over time with different work load.

While the first set of experiments stress the SRAM chips with 0s in all locations, it is also important to study the shift of the μ of $\%1s$ distribution from stress caused by normal operation, as this would occur in typical use in the field. Even in this scenario, the data bits in an SRAM memory are not random over time. It has been reported that there are usually more 0s (65-80%) [87]. Therefore, to mimic the normal operation, we perform aging with data that with different percentage of 0s. We choose five new SRAM chips (Microchip 23K640) and perform aging with 100%, 90%, 80%, 70% and 60% of 0s stored in them. We update the contents of the SRAM chips in every 5 minutes during the aging to mimic the realistic operation in the field. The experiment for aging is conducted at room temperature. Figure 6.9 shows the shift of μ of $\%1s$ distribution over time. The x-axis represents the normal aging time and y-axis represents the μ of $\%1s$ distribution. This figure provides an insight that rate of aging degradation depends on the percentage of 0s. If we perform aging with more zeros, the percentage shift becomes larger. For example, the μ of $\%1s$ becomes 52.30%, 51.98%, 51.30%, 50.5%, 50.08% after one day of aging, while the aging pattern contains 100%, 90%, 80%, 70% and 60% of 0s, respectively. After 14 days of aging, we observe a shift of 4.20%, 3.62%, 2.20%, 1.49%, 0.63% from their initial approximate 50% value for SMAM Chip 5, 6, 7, 8 and 9, respectively.

As the rate of shift for $\%1s$ distribution when aged with 60% of 0s is comparatively low, accelerated aging is performed using a ThermoSpot direct contact probe system (see experimental setup for accelerated aging in Figure 6.10). This system is an industry standard benchtop temperature cycling system, used for accelerated aging [8]. The device supports temperatures ranging from -65°C to 175°C, with a transition rate of less than 35 seconds over 25°C to -40°C. Accelerated aging has been performed at $85°C$ with "functional" random patterns, which contain 60% 0s and 40% 1s.

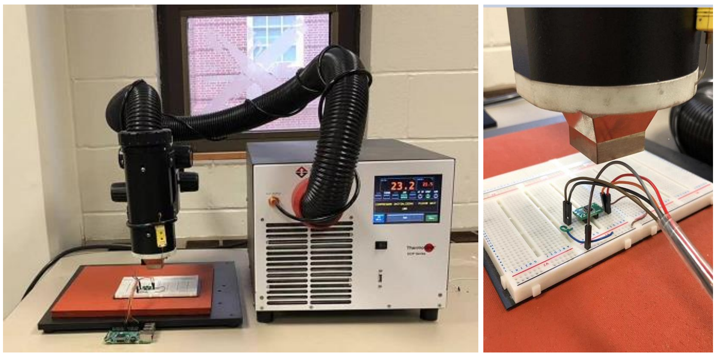


Figure 6.10: Accelerated aging set-up using ThermoSpot direct contact probe system [8].

Figure 6.11 shows the distribution of $\%1s$ with 60% 0s during the aging. The update of SRAM contents are performed in every 5 minutes while aging like before. The power-up states are also measured 100 times when the chip is cooled down to the room temperature. We have noticed the change of the mean at an accelerated rate compared to Figure 6.9. Approximately, 1% change in mean is observed after 2 Hrs of accelerated stress. Similar trend for the rate of change of the mean is also observed. Finally, we see a change of 2.23% after 70 hours of accelerated stress.

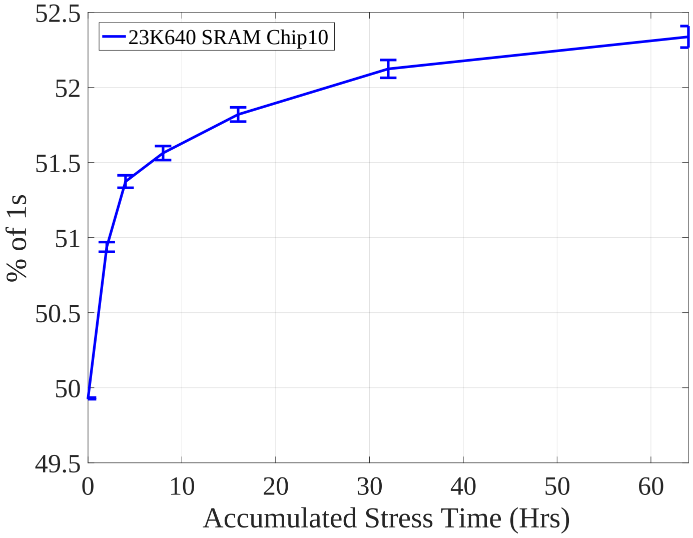


Figure 6.11: The distribution of $\%1s$ on the power-up states for Microchip 23K640 by aging with random patterns that contains 60% 0s and 40% 1s at $85°C$.

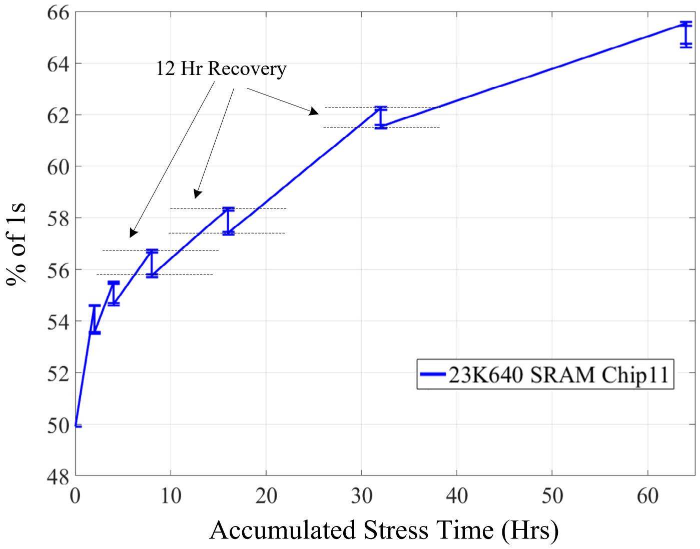


Figure 6.12: Accelerated aging and normal recovery.

The final set of experiments are conducted to analyze the recovery from the aging degradation. The recovery of transistor threshold voltages is common when a device experiences no stress. We have carefully looked and analyze the shift of $\%1s$ distribution when a chip sits on the shelf. First, we performed an accelerated stress to age the chips at a much faster rate and then relax the chip for 12 hours. Figure 6.12 shows the aging and recovery behavior for an SRAM memory (Chip 11). The chip has been aged with all 0s to accelerate the aging degradation. The power-up states are measured after the chip is cooled down to the room temperature as before. After 2 hours of accelerated stress, 4.68% change in the mean of $\%1s$ distribution is observed. We find a 1.04% of recovery occurred after 12 hours of relaxation. The amount of recovery gets reduced when the chip is relaxed multiple times. For example, we observed 0.7% of recovery after 32 hours of accumulated stress.

We also analyzed the recovery of these chips when they are sitting on the shelf. As aging was performed at different times, all our aged SRAM chips get time to recovered as if they are in the shelf. Table 6.1 summarizes the result. The first column of this table represent the recovery time period for each selected chips. Next two columns indicate the specific chip number and previous aging condition (whether aging is performed at an elevated temperature or not). The fourth and fifth columns represent the initial % of 1s (before the start of aging) and final % of 1s (after completion of aging). Finally, the last column represents the % of Δ recovery, which can be defined as the following equation:

$$\% \, of \, \Delta \text{ Recovery} = \frac{\mu_F - \mu}{\mu_F - \mu_I} \times 100\%$$

where,

μ_F : Mean of $\%1s$ distribution when aging is complete.

μ_I : Mean of $\%1s$ distribution before aging is started.

μ : Mean of $\%1s$ distribution when measurement for recovery is performed.

From Table 6.1, we can observe that chip can experience about 15% recovery for first day. The recovery slows down significantly afterwards. For example, the chip only gets a cumulative recovery of 20% in 8 days and then 22% in 10 days. However, we observe a different behavior for Chip 11. It only recovers 5% in 4 days. This anomaly can be explained as this chip have already experienced multiple recovery cycles during the accelerated aging experiment (see Figure 6.12).

Table 6.1: Recovery of aged chips sitting on the shelf.

Recovery time	SRAMs	Aging Condition	Initial % of 1	Finial % of 1	% of Δ Recovery
1 day	chip 8	Normal	49.921	51.232	15.7
4 days	chip11	Accelerate	49.919	64.086	5.09
8 days	chip7	Normal	50.002	52.027	19.68
10 days	chip5	Normal	49.805	54.626	22.10

It is practically infeasible to recover all degradation, and we have shown that some amount of aging can be recovered if the chips remain idle. In conclusion, the recovery is never complete, and majority of the aging degradation typically remains. From this analysis, we can safely conclude that recycled chips can be detected even though they are on the shelf for a long time.

6.1.4 Conclusions

The excessive growth of recycled ICs in the DoD and other critical infrastructures poses a serious threat because of their inferior quality, shorter remaining life and lower performance. Lack of efficient detection and avoidance technologies make our critical infrastructure vulnerable to these counterfeit chips. In this paper, we have presented a low-cost approach to detect the recycled SoCs using the power-up state of on-chip memories. This method does not require any prior information regarding a chip, which makes this solution well suited for the chips already circulating in the market. Our solution can be attractive to different test laboratories as it requires a simple test setup which consists of a extremely low-cost Raspberry Pi to read out the SRAM state. We have validated our proposed method using two different types of commercial off-the-shelf SRAM chips and have shown the efficiency of detecting recycled chips.

6.2 Exploiting Power-Up States of SRAMs under Varying Ramp Rates to Detect Used Recycled Parts

In previous subsection, it has suggested a novel and innovative approach for detecting recycled ICs by analyzing the power-up state of the on-chip SRAMs to evaluate the age of the chip. This new methodology does not require the introduction of any special aging detection circuitry, nor the recording and saving of historical circuit performance data as a reference to detect degradation from use. Instead, it exploits the randomness in the 1s and 0s observed in the SRAM array during initial power-up. In a new SRAM, this number of 1s and 0s should be nearly equal. Over time in use, the share of each state will deviate from 50% due to NBTI aging in operation. The earlier work has proposed exploiting this observation to detect recycled chips. However,this methodology only works reliably if the impact of all random and systematic process variations on a new SRAM state are unbiased. The silicon experiments reported in this subsection indicate that this is not always true, particularly for older technologies. In this subsection, we mainly focus on the SRAM which have been fabricated by very mature technology node and proposed our improved method to detect the recycled SRAM. Our silicon data indicate that our proposed improved method can accurately detect most recycled SRAM chips.

6.2.1 Motivation and Contribution

Recycled integrated circuits (ICs) pose a major risk in the semiconductor supply chain due to compromised reliability. These recycled ICs often exhibit degraded performance and reduced life expectancy [28], because of degradation from aging. Additionally, the steps employed in IC recycling such as dis-assembly, cleaning, and restoration, can also result in other defects and anomalies that can cause system malfunction. Consequently, the reliability and safety of any system can be significantly compromised if recycled chips are used in it.

The proposed solutions to this challenge can be broadly classified into two categories: recycled ICs detection and recycled ICs avoidance. The traditional and straightforward approach for counterfeit IC detection includes physical and electrical inspection which is time-consuming, lacks automation, and usually incurs significant cost. Recently, many researchers have proposed utilizing machine learning and imagine processing to automate physical inspection [74]. However, these techniques have several drawbacks, such as computation complexity, and requirement of expensive equipment. Furthermore, to train the neural network models, large number of data-sets are needed from known new and unused chips, which is challenging to be acquire for obsolete parts. The statistical test method, which is another type of counterfeit detection method, exploits electrical performance, or circuit parameters to distinguish between newly produced ICs and recycled counterfeit ICs [108]. Again, statistic model model building here is also hampered by the need for large data-sets [109]. Moreover, the electrical parameters can vary significantly from die-to-die due to manufacturing process variation and are also impacted by operating conditions (temperature, voltage). This affects detection accuracy of recycled counterfeit ICs because the impact of wear-out from aging on circuit parameters is extremely small. Counterfeit avoidance, which refers to Design-for-Anti-counterfeit (DfAC), comprises of methods to prevent recycling by modifying the original design [63]. The most prominent challenge faced by the counterfeit avoidance methods is its limited application. Since it requires design modification, it does not serve as a viable solution for the chips that have already entered the supply chain and are being circulated in the market.

Recently, Guin et al. proposed an innovative testing method to differentiate between the recycled SRAM and fresh SRAM by exploiting the power-up state of SRAM [83]. This method is based on the assumption that the number of 1s and 0s acquired by a new SRAM array when first powered up (without any data written into it) should be nearly equal since any imbalance in the individual SRAM cells due to the random manufacturing variations would be statistically unbiased. Half of the SRAM cells will be biased towards 0, and the other half SRAM cells will be biased towards 1, thereby resulting in nearly 50% power-up states of 1s and 0s in a newly manufactured SRAM array. This inherent initial power-up property of all new SRAMs gets skewed over time due to NBTI aging from the asymmetric stress from normal usage in the field. This is because the memory content of each cell, averaged over all the time in operation, is rarely unbiased. The cells that experience mostly 0s stored in them shift more towards the 1 bias at power-up, and vice versa. Consequently, the overall ratio of 1s deviates from 50% over time as the SRAM array is been used in the field. This can be used to identify not only used SRAMs ICs, but also the many other SOCs that commonly contain SRAMs.

Note that the above approach critically assumes that random process variations from the manufacturing dominate the biases in the SRAM cells. However, in practice systematic parameter variations are also observed in IC manufacturing, and these are particularly dominant in older technology nodes produced a decade or more ago when the impact of random process variations was also more limited. This suggests that the expected 50% ratio of 1s and 0s at power-up in a new SRAM as assumed above, may not always be a stable reference to reliably detect recycled SRAMs. Furthermore, even a modest systematic bias is the SRAM cells caused by layout or processing asymmetries may be too strong to be overcome by the small threshold changes typically experienced from NBTI aging. This will limit any change in the statistics of the SRAM power-up states with age. Thus, the method proposed in [83] may not be able to reliably detect recycled SRAMs, unless the inherent cell biases are dominated by truly random manufacturing variations.

To solve this practical challenge, in this section, we propose a significantly improved method to address the same problem and reliably detect recycled SRAM chips. As shown by the silicon experiments presented in this paper, our approach works for most SRAMs, even in decades old technology with significant systematic power-up bias in the SRAM cells. The systematic bias is overcome in two ways: (1) we depend on the architectural symmetry inherent in SRAMs to cancel out any systematic bias and yield an equal number of 1s and 0s at power-up in the blocks analyzed for aging, and (2) we modulate the power supply ramp rate at power-on to reduce the "strength" of the systematic cell bias such that the impact of NBTI aging is amplified sufficiently to influence the power up state. We present silicon data for 29 SRAMs, both new and old, representing a range of sizes and technologies, and show that our new method can successfully classify whether the part is new or used in the large majority of the cases. Note that our proposed method does not require any historic parameteric data about the part and can be directly applied to all SRAMs, old and new, as long as the the initial power-up states of SRAMs can be readout.

The rest of the paper is organized as follows: Section 6.2.2 introduces the necessary background/theory that will facilitate us to understand the power-up value of old SRAM chips. Section 6.2.3 discusses the power-up value of old SRAM chips and our proposed approach for detecting recycled SRAM chips in detail. Section 6.2.4 consists of the silicon results and demonstrates the efficiency of our proposed method to detect the recycled SRAM chips in real-world application. Finally, we conclude the paper in Section V.

6.2.2 Related Theory

In this subsection, we will mainly focus on the basic theory that will facilitate us to understand the power-up state of old technology SRAM chips.

Twist Bitline in SRAM Architecture

In this subsection, we will introduce the twist bitline in SRAM architecture. This mechanism will help us establish the concept for power-up state in old technology SRAM chips in the later sections.

In digital CMOS circuits, capacitive and inductive coupling among interconnects, power network IR voltage drop, simultaneous switching noise, etc. contribute to various forms of digital noise [110]. Inter-bitline coupling noise is due to the capacitice coupling between bitlines of adjacent columns of memory cells. This can lead to varying memory read time depending on the data stored in the adjacent memory columns [111]. To cope up with this challenge, the twisted bitline technique is commonly implemented to cancel the affect of the inter-bitline coupling noise.

Figure 6.13 shows a typical twisted bitline structure for a SRAM array. Each colored square (red or blue) represents single SRAM cell and $b0$ ($\overline{b0}$) represents a pair of bitlines. It is easy to see that for this bitline structure, the data stored in the adjacent columns will equally impact the $b0$ and $\overline{b0}$ lines of any column of cells. As a result, the overall noise will be equalized between the two, maintaining the desired balance between the lines. In general, this twisted bitline structure is just one form of the many symmetries in a typical SRAM array layout. However, it has a major impact on the power up statistics. Note that even if all the memory cells in the array have a systematic bias based on layout and processing, such that every cell always powers-up with the left output of the cell at a 1 and the right output at a 0, the twisted bitline architecture ensures that exactly half the cells will read out a 1 and the other half a 0 when initially powered up. This is because, as shown in Figure 1, half the left outputs of the cells in every column are connected to the bit output line and the other half to $\overline{bit}$ line. Thus the flipped bitlines in SRAM arrays nearly always ensure (we have found only one instance of an unusual alternative architecture) that a new SRAM always powers up to with an equal number of 1s and 0s, even if the cells have a strong systematic bias in a fixed physical direction in the layout . This allows the 50% initial statistic to be used as a baseline reference in detecting used SRAMs even in the presence of systematic bias in the cells.

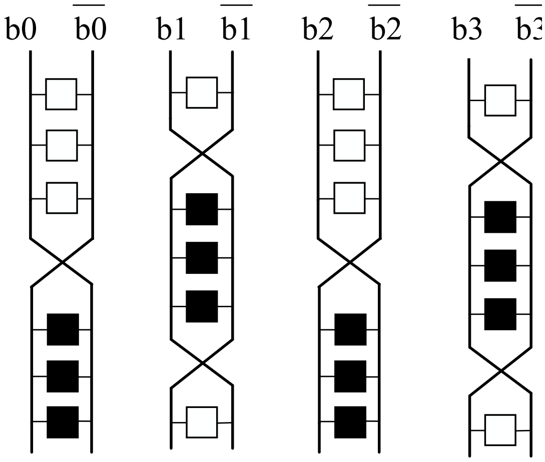


Figure 6.13: The architecture of bitline flipping in SRAM array.

The Ramp Rate Effect on Power-up State of SRAM

This section presents the ramp rate effect on the power-up state of SRAM. Further detailed analysis can be found in chapter 5.

In most discussions of the power up state of SRAMs, including the earlier work on recycling detection [83] in SRAMs, the ramp rate of the supply voltage is not considered. It is assumed that the SRAM is powered-up relatively quickly when switched on. This ramp rate is, in practice, limited by the intrinsic time constant of the power rail internal to the IC, which can be of the order of a nanosecond or so. On the other hand, the supply voltage can also be raised very slowly, over several seconds. In this Section we show that this ramp rate can have a very major influence on the power up state of the SRAM. Importantly, it is possible to find a ramp rate at which the power-up state is most influenced by NBTI aging to help detect recycled parts.

In figure 6.14, a typical SRAM cell schematic has been presented with two pairs of MOSFET and two capacitors C_1 and C_2. If VDD is raised instantaneously, M_1 and M_2 will be fully switched on and the M_3 and M_4 will be off since V_1 and V_2 cannot change instantaneously and will be zero at the time zero. Consequently, the current that charge up V_1 and V_2 will be dominated by M_1 and M_2. Depending on the parameter of M_1 and M_2 and C_1 and C_2, one side will be charged up quicker than the other and finally obtain a logic value of 1. The final state of V_1 and V_2 will be decided by the parameters of two PMOS and two parasite capacitors. On the other hand, if VDD is raised very slowly, at each point of time, the power can be nearly regarded as a DC power since the change or delta in power is very small. If we assume VDD as DC power, the capacitors C_1 and C_2 will be in quasi-equilibrium and can be treated as disconnected. The final state of V_1 and V_2 will be decided by the parameters of four MOSFETS.

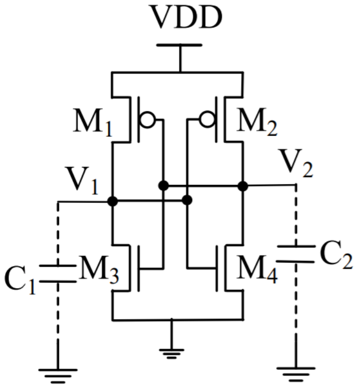


Figure 6.14: The model of a typical SRAM cell.

In general, under quick ramp, $M_1(M_2)$ and $C_1(C_2)$ will determinate the final state of V_1 and V_2. Under slow ramp, M_1, M_2, M_3 and M_4 will determinate the final state of V_1 and V_2. These concepts will be referred throughout the remaining paper to understand the proposed method.

Note that the above discussion suggests that the power-up states can be vastly different because different devices and parasitic capacitances determine the outcome depending on the ramp rate. To illustrate this, Figure 6.16 shows bitmaps of an SRAM in relatively mature technology under different power ramp-up rates ranging from 7ns to 10s. In the bitmaps, the black dots represent cells that have a 1 during power-up and the white dots represent the cells that have a 0. The bitmap for the fast 7ns ramp rate clearly show a strong systematic bias is the cells, with half the SRAM power-up to a 1 because of the architectural symmetry discussed above. Overall, the SRAM still displays 50% 1s and 0s. It is interesting to observe how the power-up states change as the ramp rate is slowed down,completely fliping the power-up states for a 1s ramp. In the rest of the paper, we will interpret these changes to identify used SRAMs.

Local Bias of SRAM Cells

The power-up state of the SRAM array has also been investigated by a number of researchers in attempts to attack SRAM based PUFs. Some of the researchers have proposed that the power-up value shows systematic/local bias in the SRAM array. In [2], the authors observed the neighborhood effect in the power-up values of SRAM cells. In other words, the strong bias of '1' bit will be always surrounded by the bias '1' bits and the strong bias '0' bit will be always surrounded by the bias '0' bits. In [112], it experimentally demonstrated that signatures generated from two memory chips may have highly correlated properties if they possess the same set of specifications and came out of a similar manufacturing facility, which was used to mount a non-invasive attack against memory-based PUFs [112]. This also implies that the power-up value of SRAM displays systematic/local bias.

In the next section, we present an innovative method for detecting used SRAMs that leverages architectural symmetry to balance out the systematic cell bias and also uses different power-up ramp rates to highlight the shifts in power-up states from NBTI stress. Our approach is validated in the silicon experiments in later sections.

6.2.3 Proposed Method to Detect Recycling SRAM

In this section, we began by discussing regular pattern in power-up state of SRAM array. Then we shift our focus to the old technology SRAM chips, where we observe that SRAM cells experience power-up state bit flipping under different ramp rates. To validate this, corresponding simulations have been performed. Lastly, an improved testing method has been proposed for detecting the recycled SRAM chips.

Systematic pattern of power-up state of SRAM array

As seen earlier in Figure 6.16, the bit maps of the power-up states of an SRAM array display systematic patterns, with some randomness in each area. These patterns change with ramp rate. The selected ramp rates here are $7ns$, $1us$, $1s$ and $10s$ and the corresponding bitmaps are shown in the figure 6.16. The experiment was conducted on MS62256 which contains 256K bits and the figure shows 256K corresponding dots representing the power-up bit values. The black dots represent the logical '1' while the white ones show logical '0' during the power-up process. We can clearly observe a regular pattern or systematic bias that can be seen across different ramp rates. For example, the bitmap for $7ns$ ramp rate indicates that the top half cells in the array are nearly all '1' (black) and the bottom half nearly all '0'(white).

These regular clusters of 1s and 0s can be explained by a systematic cell bias that could have resulted from several possible sources in layout or the manufacturing process. For example, layout asymmetries in the cell template can give rise to small (but identical) imbalances in each pair of transistors or interconnect capacitances in all the SRAM cells. Processing irregularities can also result in regular asymmetries in all the cells. For example, in the figure 6.14, the size of M_1 could be marginally wider (stronger) than M_2, for all the cells in an array. Assuming that the rest of the parameters of this cell are fully symmetrical, then the power up behavior would be a reflection of this size difference. If this cell powers-up at a slow ramp rate, then V_1 will charge quicker than V_2 since M_1 have a higher current drive. This will result in V_1 power-up to 1, and V_2 0. In other words, such a cell is biased to 1 under a slow ramp rate. If this is a systematic imbalance in all the cells, then the entire array of physical cells, as laid out will be biased to 1.

However, to minimize the coupling noise, the bitlines carrying the output signals from the SRAM array are twisted regularly and periodically as discussed in Section II. Because of this bitline twisting, an equal number of say the left cell outputs will be connected to the bit and ($\overline{bit}$) lines in any column. Thus half the cells will appear to be biased to 0 instead of 1 because of this interleaving of the bit lines. Consequently, we can still expect 50% 1s and 50% 0s for an unused SRAM chip, even in the presence of systemmatic cell biases.Clearly these statistic will also hold in the presence of any additional (neutral) random manufacturing variations.

Power-up value flipping of SRAM cells under different ramp rate

In this subsection, power-up value flipping under different ramp rates has been discussed and corresponding simulation has been performed to further validate and support our theory.

In figure 6.16, the changing power-up states as the ramp rate is slowed down is another, very significant observation. As we move towards right, i.e. from a very quick ramp rate to an increasingly slower one, we see that the bits which were 1 (black dots) under the 7ns ramp have nearly flipped to

0 (white dots) at $1s$ (referring to upper half of bitmaps). Careful observation indicates that some of these cells changed back to 1 between $1s$ and $10s$. A similar, although complimentary trend can be seen for the lower half of each bitmap corresponding to each ramp rate. In other words, most of the cells change their power-up value as the ramp rate slows down. Some of the cells even flip multiple times as the ramp rate slows. (Powering up with 1 under $7ns$, 0 under $1s$ and 1 under $10s$.)

Recall from the background theory discussion in the section II, that under different ramp rates, the MOSFETs and parasitic capacitors will have different weights in determining the final state of SRAM cells. Additionally, the MOSFETs in each pair may be unbalanced because of some systematic asymmetry. To study these effects using a cell circuit model, we performed HSPICE simulations using the predictive technology (PTM) $180nm$ model. In figure 6.15, we have detailed the parameters of each MOSFET that was used during the simulation. Figure 6.15.(a) shows cell1, where only the two PMOS transistors have slightly different sizes and rest of the parameters are totally symmetric and equal. Whereas, for cell2 in Figure 6.15.(b), the sizes of the two PMOS and two NMOS and access transistors are all different. Note that the size difference between two MOSFETs is not very large with all sizes shown are in nm. These two SRAM cells have been simulated under different power-up ramp rates and the power-up state have been recorded as shown in table 6.2. Each column in the table specifies different ramp rates and the corresponding power-up value for cell1 or cell2 is provided. Selected ramp rates for the simulation are $7n$, 1μs, 100μs, $1ms$, $10ms$ and $1s$. The cell1 power-up with 0 under quick ramp (1μs) becomes 1 under slow ramp ($1s$). For cell 1, the size of $X1\_M1$ is larger than $X2\_M2$. Under a quick ramp rate, $V1$ will charge up more slowly than $V2$ because larger size of $X1\_M1$ will introduce larger parasitic gate capacitors. Under a slow ramp rate, $V1$ will be charge quicker than $V2$ since $X1\_M1$ have the larger current drive. Note under quick ramp rate, only PMOS transistors and the parasitic capacitors decide the power-up state. Under slow ramp rate, the four MOSFETs decide the power-up state. A similar analysis can be done for cell2 which powers-up to 1 under quick ramp ($7ns$) and changes to 0 as the ramp rate decreases to 100μs and finally becomes 1 again under slow ramp ($1s$). This simulation study confirms that systematic biases in transistor sizes can cause the power-up logic value to flip state once or even multiple times under different ramp rates.

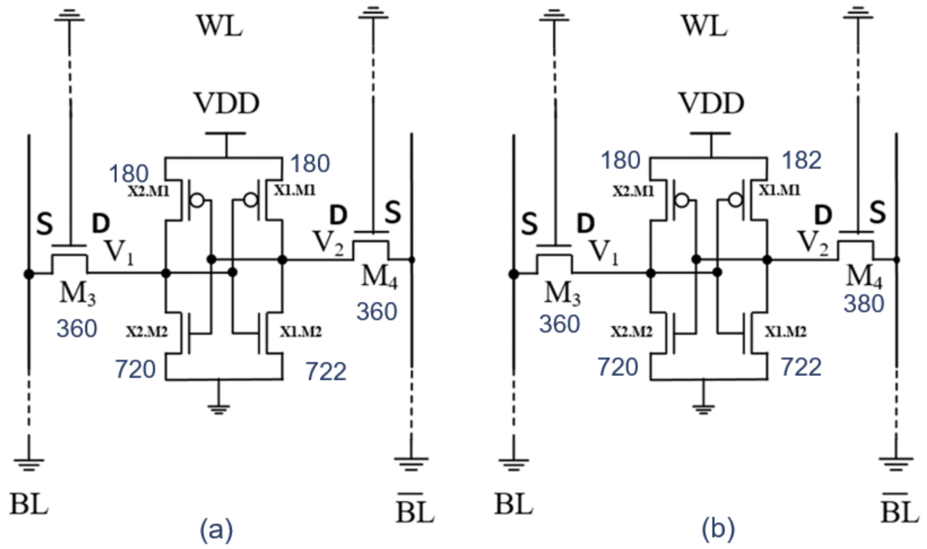


Figure 6.15: SRAM model for powering up analysis under different ramp rate by introducing systematic sizing bias. (a) cell 1 (b) cell 2

Table 6.2: Simulation result for sample SRAM cells under different ramp rate

Ramp rate	7ns	1us	100us	1ms	10ms	1s
Powering up value of cell 1	0	0	1	1	1	1
Powering up value of cell 2	1	1	0	1	1	1

Recycled SRAM Detection

Due to the symmetry of the SRAM array, along with twisted bitline in the layout, a 50% ratio of 1's during power-up can still be expected for any new unused SRAM implemented in a traditional architecture symmetric . However, simply analyzing the ratio of 1s and 0s at power-up without concern for the power supply ramp rate may not result in the reliable detection of used SRAMs. This is primarily because of the strong systematic cell biases seen in some SRAMs, as observed in the well defined half black and half white bitmap for the 7ns ramp rate in Figure 4. Such cells can be difficult to flip even by extreme long duration NBTI stress; in practice SRAM cells experience both 1 and 0 states in operation, with the accumulated duration in the 0 state statistically only somewhat greater than in the 1 state. Thus not enough cells may flip from operational stress to provide a statistically reliable indication of prior use.

To tackle this challenge, we propose to vary the power-up ramp rate and find one where the systematically biased cells in the array are in the middle of flipping state. Observe that as the ramp rate is slowed down from 7ns, the strength of the initial systematic bias in both the black and white cell decreases as the cells move towards flipping state. However, this decrease is symmetric and retains the balanced count of 1s and 0s in a new SRAM even as the cells flip state. However, in used SRAMs, the threshold voltage shifts from NBTI aging can have the most influence on the power up state when the systematic bias is minimal. This appears to be the case at the 1microsec ramp rate in Figure 4, as the array is nearly uniformly grey suggesting an even mix of both cell polarities. Here the disparity in the 1 and 0 power-up counts is maximized in used SRAMs and can allow used parts to be identified.

The silicon experiments reported in the next Section confirm that tracking the ratio of 1s and 0s while sweeping the ramp rate can reliably identify recycled IC. A 10% maximum difference between the number of 1s and 0s (at the ramp rate where systematic manufacturing cell bias is minimized and aging is most evident) appears to be a robust threshold between new and used SRAMs. However the larger this difference, the more confidence one can have that this is a used part.

It should be noted that some unusual SRAMs may not be detected by our method. The architecture of these outlier SRAMs is not fully symmetrical, implying that an equal number of 1 and 0 bits cannot be expected at power-up even in a new chip. However, based on our experiments with the random collection of SRAM chips reported in the next section, only one in 29 appeared to have such an architecture. Another class for which it can be difficult to make a recycling decision are SRAMs that have an extremely strong systematic bias that does not shift significantly with ramp rate. The power up states of these chips are not significantly impacted by aging. Fortunately, such SRAMs are encountered in very limited numbers in our experience.

Table 6.3: Ratio of 1 during Powering up under different ramp rate for chip MS62256

Ramp rate	7ns	1us	100us	1ms	10ms	1s
% of 1 during powering up	50.91	50.8	51.19	59.14	49.92	48.72

6.2.4 Silicon Result

To validate the efficiency of our proposed method to detect recycled SRAM chips, we conducted experiments on different SRAM chips procured from several vendors (eBay, Mouser, and Digikey). Some of these SRAM chips are used and some of them are new. The manufacturing date of these chips dates back from 1980s to the 2000s. The supply voltage of these chips is from 1.8v to 5v. We tested nearly 30 chips using our proposed method to classify.

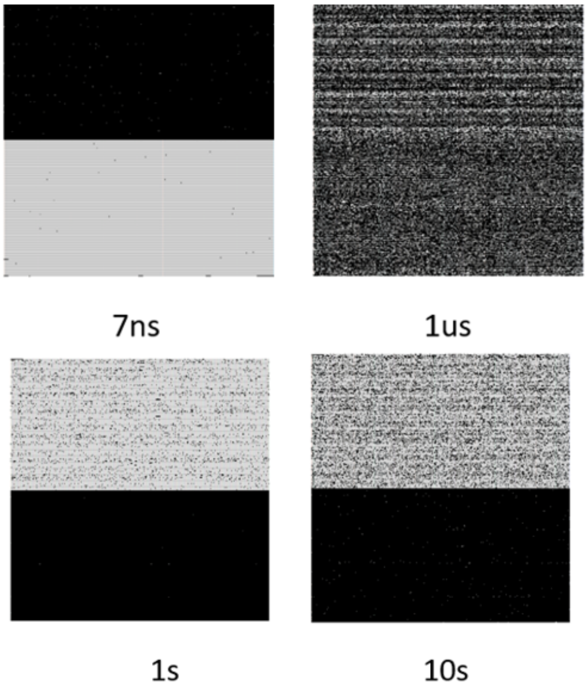


Figure 6.16: Bitmaps of power-up states under different ramp rate for MS62256.

In Figure 6.17, bitmaps of three SRAM chip samples under different ramp rates is presented. They belong to different manufacturing dates. In other words, likely, different technology nodes have been applied to these three chips during the manufacturing process. The first row shows the bitmaps for the chip MS62256 and the possible manufacturing date is in 1990s and the second row shows the results for chip cy7c186 and the manufacturing date is 1995. The last row is for the the chip 23A230 which is manufacturing date in 2005. From the top row to the last row, we can clearly observe regular pattern in the bitmaps under the quick ramp ($7ns$). Secondly, the regular pattern becomes less systematical as the manufacturing technology becomes more advanced. Mainly, because regular patterns are caused by the systematical bias which is more dominant in older technology. More specifically, it is caused by the systematical sizing change of MOSFETs during the manufacturing process which gets applied to all the cells. The random process variation becomes more and more significant as we move towards more advanced technology nodes. Finally, the random process variation will fully overcome the systematic bias. Due to this, the power-up value of SRAM chips that have been manufactured by advanced technology will be random, and a very less regular pattern can be observed. Lastly, if we observe the bitmaps under different ramp rates for one specific chip, the power-up value flipping can be easily observed as we move from quick ($7ns$) to slow ($10s$) ramp rate. For example, the first row represents the bitmaps for chips MS62256. Under the quick ramp rate, the first half cell has the powering up value of 1. Under a slow ramp rate, the first half cell becomes 0 during the powering up. As explained earlier, this power-up value flipping is caused by the systematical sizing bias during the manufacturing process. However, since the last chip (i.e. 23A230, last row) has been manufactured by advanced technology, the power-up value flipping effect also cannot easily be observed. This also indicates that the random process variation becomes the dominant effect as the manufacturing technology becomes advanced which is well understood so far.

Table 6.4: Experimental silicon results for recycled SRAM detection

Chip #	Part #	% of 1s quick ramp	% of 1s slow ramp	% of 1s mid ramp	VDD	Memory Size	Condition	Source	Prediction
1	23A230	49.53	49.11	49.83	1.8	64K	New	Digkey	Pass
2	23K230	50.02	49.97	50.22	3.3	64K	New	Digkey	Pass
3	23K230	49.96	49.90	50.11	3.3	64K	New	Digkey	Pass
4	23K230	50.05	49.93	50.10	3.3	64K	New	Digkey	Pass
5	CY7C198	55.85	79.06	83.38	5	256K	Used	Ebay	Pass
6	CY7C198	53.68	62.06	62.77	5	256K	Used	Ebay	Pass
7	CY7C199	50.14	50.95	50.98	5	256K	New	Ebay	Pass
8	CY7C199	49.89	50.67	50.67	5	256K	New	Ebay	Pass
9	CY7C199	50.04	51.91	51.93	5	256K	New	Ebay	Pass
10	UM61256	47.08	45.51	47.3	5	256K	Used	Ebay	Fail
11	UM61256	49.68	48.32	49.77	5	256K	Used	Ebay	Fail
12	W24256	50.01	50.02	50.02	5	256K	Used	PCB	*
13	W24256	49.99	50.01	50.02	5	256K	Used	PCB	*
14	W24256	50	50.02	50.02	5	256K	Used	PCB	*
15	W24256	49.99	50.02	50.02	5	256K	Used	PCB	*
16	AS7C256	25.37	40.14	40.14	5	256K	New	Digkey	*
17	CY62256	51.12	50.72	51.12	5	256K	New	Ebay	Pass
18	TC5565	49.74	64.83	99.44	5	64K	Used	Ebay	Pass
19	TC5565	52.76	74.23	98.54	5	64K	Used	Ebay	Pass
20	6264BL	50.25	50.01	50.33	5	64K	Used	Ebay	Fail
21	TC5565	58.59	77.47	99	5	64K	Used	Ebay	Pass
22	TC5565	48.45	77.82	97.12	5	64K	Used	Ebay	Pass
23	MS62256	50.91	46.31	59.4	5	256K	Used	Ebay	Pass
24	MS62256	52.32	55	63.67	5	256K	Used	Ebay	Pass
25	MS62256	52.13	42.94	61.58	5	256K	Used	Ebay	Pass
26	MS62256	50.64	46.12	57.48	5	256K	Used	Ebay	Pass
27	AS6C6264	50.23	51.35	51.8	5	64K	New	Mouser	Pass
28	AS6C6264	55	49.88	55.86	5	64K	New	Mouser	Pass
29	AS6C6264	54.72	50.42	54.81	5	64K	New	Mouser	Pass

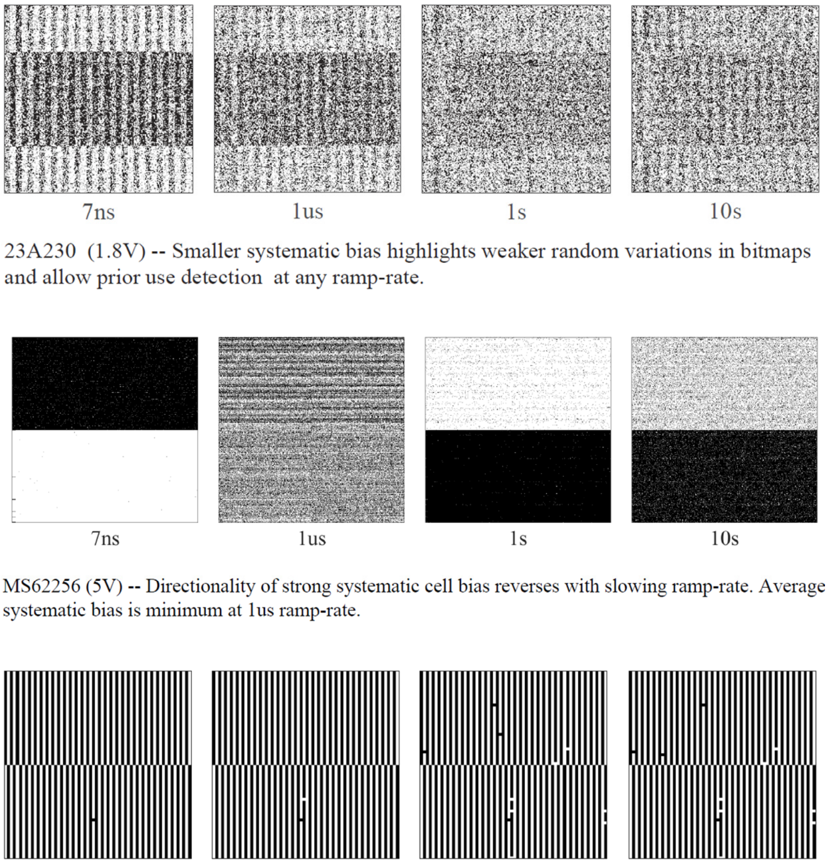
The main aim of this paper is to classify and detect whether a chip is recycled or not for old/mature technology nodes with the help of our proposed method. In order to validate the efficiency of our proposed method, we conducted experiments on nearly 30 chips, procured from different vendors and the results are presented in the table 6.4. These chips differ in, supply voltage, manufacturing dates, and use conditions. The first column represents the chip number and the second column represents the part number of chips. The third column represents the ratio of 1 if the chip has been powered up with a quick ramp rate (7ns). The fourth column indicates the ratio of 1 if the chip has been powered up with a slow ramp rate (10s). Since the chip has also been tested in the middle range (1us,10us,100us et al.) of ramp rate for powering up, the fifth column represents the maximum ratio of 1 of power-up value for the chip amongst all the ramp rates under test. The next two columns represent the supply voltage and memory size. The eighth column indicates the use condition of the chip. We have also listed the purchase source of the chip in ninth column. The last two columns being the important ones, represent the % of cells that experiences the power-up value flipping under different ramp rate of powering. Our threshold to determinate the unused SRAM is from 45% to 55%. In other words, after we check the ratio of 1 of powering up values under different ramp rates, if all the values are within the range (45% to 55%), the chip will be considered as an unused part. Otherwise, the chip will be regarded as recycled SRAM chips. The final result indicates that most chips have been correctly identified as unused or recycled. The accuracy can be classified using two different parameters, namely yield loss and defect level. Yield loss is the percentage of good/unused chips that were rejected due to identifying them as used chips:

$$Yield\ loss = \frac{AU - DU}{AU}\% = \frac{12 - 12}{12} = 0\% \tag{6.2}$$

Where AU and DU represents actual unused chips and detected unused chips respectively. Misprediction rate represents the percentage of chips that we make the wrong decision on the condition.:

$$Mis - prediction\ rate = \frac{TN - CN}{TN}\% = \frac{24 - 21}{24} = 12.5\% \tag{6.3}$$

Where, TN and DR represents the total number of chips that we can make a decision and the number of chip that we make the correct decision on the usage condition using our method.



23A230 (1.8V) -- Smaller systematic bias highlights weaker random variations in bitmaps and allow prior use detection at any ramp-rate.

MS62256 (5V) -- Directionality of strong systematic cell bias reverses with slowing ramp-rate. Average systematic bias is minimum at 1us ramp-rate.

W24256 (5V) -- Very strong systematic bias is uninfluenced by random variation or ramp rate. Aging cannot be detected.

Figure 6.17: Bitmaps of SRAM chips sample.

The overall yield loss was 0% which signifies that our method is accurate in classifying unused chips. The mis-predication rate is 12.5% where 3 of such chips couldn't converge to correct classification based on the proposed method and pass the test. In the table, two type of chips are indeterminate. One is as7c256 and the other is w24256. The chip as7c256 do not have the fully symmetrical architecture after checking the bitmap of power-up value. The chip w24256 have the extremely strong systematic bias and no power-up flipping was observed under multiple ramp rate of powering up.

6.2.5 Conclusion

Counterfeit ICs pose an annual risk of $169 billion in the global supply chain and Recycled ICs contribute around 80% of all reported ”counterfeiting” incidents. However, the Lack of efficient detection and avoidance technologies makes our critical infrastructure vulnerable to these counterfeit chips. In previous paper, a low-cost method has been proposed to detect the recycled SRAM chips by exploiting the power-up state of SRAM. However, the method only works efficiently if the random and systematic variation are unbiased for entire SRAM array. In this paper, we improved the detection method by exploiting the power-up state of SRAM under different ramp rates of powering. This improved method can be applied to those SRAM chips manufactured by using very mature technology. Our silicon result indicates that the proposed method can accurately detect most recycled SRAM chips in the real world.

Chapter 7

A Zero-Cost Detection Approach for Recycled ICs using Scan Architecture

The recycling of used integrated circuits (ICs) has raised serious problems in ensuring the integrity of today's globalized semiconductor supply chain. This poses a serious threat to critical infrastructure due to potentially shorter lifetime, lower reliability, and poorer performance from these "counterfeit" new chips. In last chapter, we have proposed a highly effective approach for detecting such chips by exploiting the power-up state of on-chip SRAMs. Due to the symmetry of the memory array layout, an equal number of cells power-up to the 0 and 1 logic states in a new unused SRAM; this ratio gets skewed in time due to uneven NBTI aging from normal usage in the field. Although this solution is very effective in detecting recycled ICs, its applicability is somewhat limited as a large number older designs do not have large on-chip memories. In this paper, we propose an alternate approach based on the initial power-up state of scan flip-flops, which are present in virtually every digital circuit. Since the flip-flops, unlike SRAM cells, are generally not perfectly symmetrical in layout, an equal number of scan cells will not power-up to 0 or 1 logic states in most designs. Consequently, a stable time zero reference of 50% logic 0s and 1s cannot be used for determining the subsequent usage of a chip. To overcome this key limitation, we propose a novel solution in this paper that reliably identifies used ICs from testing the part alone, without the need for any additional reference data or even the netlist of the circuit. Through scan testing of the IC, we first identify a significant number of asymmetrically stressed flip-flops in the design, divided into two groups. One group of flip-flops is selected such that it mostly experiences the 1 logic state during functional operation, while the other group mostly experiences the 0 state. The resulting differential stress during operation causes growing disparity over time in the number of 0s (and 1s) observed in these two groups at power-up. When new and unaged, these two groups behave similarly, with similar percentage of 1s (or 0s). However, over time the differential stress makes these counts diverge. We show that this changing count can be a measure of operational aging. Our simulation results show that it is possible to reliably detect used ICs after as little as three months of operation.

The main contributions of this chapter are summarized below:

- To the best of our knowledge, this is the first approach that can detect any used and recycled digital IC, by testing the part alone, without the need for any additional information or even the netlist of the circuit. There is no need for any stored time-zero reference data for each new part, or even statistical aggregate data that characterizes new unused parts. It is primarily the unavailability of such reference data for discontinued parts that limits the detection of recycled ICs through tests for signal degradation in use.
- We show that the diverging percentage of 0s (or 1s) at power-up in the two groups of selected FFs can provide an indication of the operational age of the digital circuit. In general, age detection accuracy depends on the size of the chip, and the resulting number of flip-flops that can be identified as strongly 0 biased and strongly 1 biased during operation. For large circuits with hundreds of thousands of FFs, our simulation results show that it is possible to reliably detect recycled chips in use for as little as three months in the field.

The rest of the chapter is organized as follows: Section 7.2 introduces modeling of the power-up state for a flip-flop, and how it is impacted by the aging. Section 7.3 discusses our proposed approach for detecting recycled ICs in detail. Simulation results are presented in Section 7.4. Finally, we conclude the paper in Section 7.5. The majority of works in this chapter was published initially in [113].

7.1 Background and Motivation

The illicit recycling of used ICs as new poses a severe threat to the reliability and security of electronic systems due to the lack of effective detection methods to identify such parts. In last chapter, we have proposed a novel and very effective approach for detecting recycled ICs and systems-on-chip (SoCs) using the power-up state of an on-chip SRAMs [114]. Importantly, in our new approach, the process of determining whether a chip is recycled or not uses a reference that is an invariant property of all new SRAMs; it does not need any reference data to be measured and obtained from new parts. Note that since each SRAM cell in a large memory array is designed with perfect symmetry, its power up logic state will depend on the random manufacturing process variations within each cell, as well as the electrical and thermal noise experienced during power-up. As these processes are Gaussian in nature, observing the power-up state in a large number of cells should yield 50% 1s and 50% 0s in a new unused SRAM. This inherent initial property of all new SRAMs gets skewed over time due to aging stress from normal usage in the field because the memory content of each cell, averaged over all the time in operation, is rarely unbiased. Cells those spend the majority of the time in the 1 state increase their bias towards powering up in the 0 state, while those that mostly experience the 0 state increase their bias towards powering up in the 1 state. As was verified by the silicon experiments reported in last chapter, a shift in either direction from a balanced number of 1s and 0s can be used to identify a used part. Since SRAMs cores are commonly found today in processors and SoCs, this method for detecting recycling has wide applicability. However, a variety of older digital ICs circulating in today's complex supply chain, and also some modern ICs, may not have on-board SRAMs. Recycling of these parts cannot be detected using the solution as proposed in last chapter.

In this chapter, we propose a new recycled part detection approach that can be applied to virtually every large digital circuit, even older parts already in the supply chain. Our new approach is based on the power-up state of the flip-flops (FFs) in the circuit, accessed directly through the scan chains [115]. Importantly, it only requires the scan testing the part alone, and no additional information or even the netlist of the circuit.

The new approach uses the power-up state of the scan FFs to determine whether a chip has been used and recycled. However, unlike our previous approach for SRAMs [114], a stable time zero reference of 50% logic 0s and 1s at power-up cannot be relied upon because the FFs memory cells may not be perfectly symmetrical in layout. As a result, they may display some systematic bias at power-up. (Note however, that any bias is typically small, since it must be minimized by the cell designers so as to maximize cell stability and noise margins.) Nevertheless, while the FFs in a new IC cannot be expected to always display an equal number cells powering-up to the 0 and 1 logic states, in a new unstressed part, any statistically large subset of identical FFs should power-up to the same percentage of 0s and 1s. We propose to use this alternate time zero reference property for the FFs in this paper. Our new approach exploits the differential NBTI aging stress in two groups of FFs that mostly experience the logic 0 and logic 1 states, respectively, during functional operation. These two groups, one strongly biased towards 0s and the other towards 1s, can be easily identified through scan testing. While their numbers may be only 1-3% of the total, they would still be at least a thousand or more in each group for a large design. The power up states of these two groups of FFs in a new part (at time zero) can be expected to be statistically very similar as discussed above (although not 50% 1s and 0s as is the case for SRAMs), before operational stress is applied. However, over time the differential stress in operation causes the statistics for 0 and 1 states in the two flip-flop groups to move in opposite directions from their time zero percentages. We show in later Sections that this growing difference in the percentage of 0s (or 1s) observed at power-up in the two groups of flip flops is quite stable (for large groups of at least a few hundred FFs) and can be used to reliably detect used parts after only a few months of aging. Observe that this new approach does not need any statistical data from other parts, nor any recorded reference data from the part when new, to identify used parts.

7.2 Modelling of the Flip-flop Power-up State

Unlike SRAM cells, the initial power-up value of a D flip-flop (DFF) is not equally likely to be logic 0 or 1 due to possible asymmetry in the cell layout. In this section, we present the power-up behavior of a DFF, and the impact of manufacturing process variations and aging on it during operational deployment.

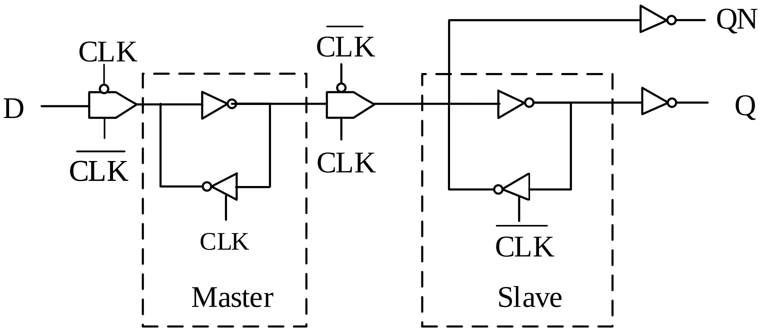


Figure 7.1: The schematic of a DFF.

7.2.1 Power-up State of a Flip-Flop

Figure 7.1 shows the typical structure of a D-type Flip-flop (DFF). Observe that the output of the clocked inverters shown in each latch are enabled by the appropriate clock signal; they assume a high impedance state when not enabled. We focus on the logic state acquired at power up by the latches in the flip-flop. Note from Figure 7.1, that depending on whether the clock signal CLK is held high or low during power-up, only one of the two latches (either master or slave) will have its feedback path enabled to act as a storage element. For example, if the value of CLK is held at logic 0, the slave latch will be active and decide the power-up state of the FF. Also, the input lines to the latch will not influence this power up state since the slave latch is completely isolated by the middle transmission gate during power up time. While a power-up state can also be captured in the master latch by holding the clock high during power-up, in this paper we work with the slave latch. We assume throughout (including in our simulation experiments) that CLK is at logic 0 during the power-up time. In addition, we assume that the state of the slave latch is the same as that of the DFF for purposes of the following discussion. A similar analysis can be performed for the master latch with CLK held at logic 1.

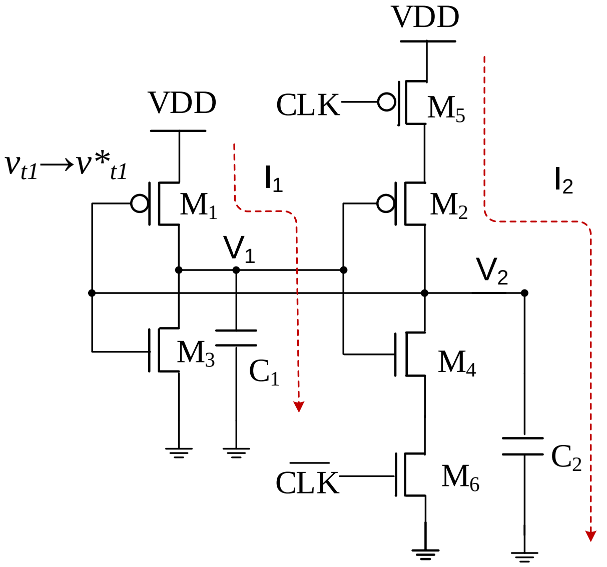


Figure 7.2: Simplified schematic of slave D latch.

Figure 7.2 shows a simplified schematic of the slave latch of a DFF. The latch usually contains two back to back inverters, where one of them, as discussed earlier, is a tri-state inverter controlled by the CLK signal. The latter is implemented using two additional transistors as shown in the figure. In the design, M_1, M_3 and M_2, M_4 consist of two back to back inverters, while M_5,M_6 are controlled by CLK. Note that CLK must be set to logical 0, in order to isolate the slave latch from the master latch. Consequently M_5 and M_6 are always on as long as CLK is 0, and can be approximately modelled by their channel resistances, R_{on}. Note that R_{on} for the NMOS and

PMOS transistors may be different due to transistor sizing and device parameters. Moreover, the capacitances C_1 and C_2 may also end up with different values due to asymmetry in the layout. Thus, when compared with the perfectly regular layout of 6 transistor SRAM cells [114], the latches in a FF have several potential sources of asymmetry. In practice, the cell designers do attempt to minimize such asymmetry and resulting biases in the latches because such biases reduce noise immunity and makes the FFs more vulnerable to bit flip errors. However, in general, the latches will not attain a perfect 50% chance to power-up to 0 or 1, even if the $M_1 - M_2$ and $M_3 - M_4$ transistor pairs are designed to be identical in the layout.

7.2.2 Impact of process variation and aging on the power-up states

In this subsection, the impact of aging and process variation will be investigated in detail using Figure 7.2. Recall that the slave latch in a DFF can have inherent biases due to differences in node capacitances, C_1, C_2, and the presence resistance R_{on} in one path. Furthermore, any bias in the layout is further impacted by process variation in the transistors and passive components. For simplicity, we ignore differences in the NMOS transistors as the power-up state primarily depends on the PMOS transistors when a fast ramp is applied to the power supply [76]. As depicted in Figure 7.2, assume V_{t1} increases to v_{t1}^* due to process variation. Consequently, I_1 will decrease, which leads to a larger time needed to change the voltage across C_1. If the voltage V_1 increases less than the voltage V_2, the latch will power-up to logical 1 (output node $V_2 = 1$ and $V_1 = 0$). In our simulations, the threshold voltage of each MOS transistor is taken from a Gaussian distribution with a standard deviation of 5% of the mean value. The possibility that threshold voltage of M_1 exceed M_2 is exactly 50%. However, based on previous analysis, in order to power-up to logical 1, this threshold voltage difference between M_1 and M_2 should be large enough to overcome any other biases in the latch. .

The next step is to investigate the impact of aging. For simplicity, we assume that all the transistor pairs, M_1, M_2 and M_3, M_4, have the same threshold voltages. We now age this latch with logical 0 ($V_2 = 0$ and $V_1 = 1$). The (magnitude of the) threshold voltage of the stressed (ON) transistor M_1 starts to increase due to NBTI and reaches to $v_{t1}^*(> v_{t1})$ while the threshold voltage of the unstressed (OFF) transistor M_2 will remain the same. Consequently, the latch will begin to power-up to the logic 1 state when the impact of aging exceeds that of the other imbalances in the cell in the opposite direction. We can therefore conclude that more DFFs will power up with logical 1 values over time when aged in the logic 0 state, and vice versa.

Table 7.1: Start-up value of 1000 DFFs with 3 months of usage.

%0s in aging patterns	100	80	60	40	20	0
%1s in power-up state	21.5	19.5	12.5	9.6	7.2	4.3

Table 7.1 shows HSPICE simulation results that include the impact of aging for 1000 latches (in DFFs). The 3 month aging experiment was repeated for the latches randomly preset to different percentages of 0s ranging from 100% to 0%. The simulations were run using 32*nm* bulk Predictive Technology Model (PTM) [92], with 20 mV (standard deviation) random process variation introduced into the threshold voltages of each MOS transistor (M_1, M_2, M_3, and M_4). This aging simulation was performed using Synopsys HSPICE MOS Reliability Analysis (MOSRA) [91]. Table 7.1 shows the percentage of 1s observed in the power-up states of the latches (DFFs) in each experiment. First observe that latch has an inherent bias towards the 0 state because the percentage of 1s is always less than 50%. Nevertheless, the percentage of 1s observed at power-up increases when the latches are mostly aged in the 0 state, as compared to being aged in the 1 state. This observation is the basis of the proposed recycling detection approach.

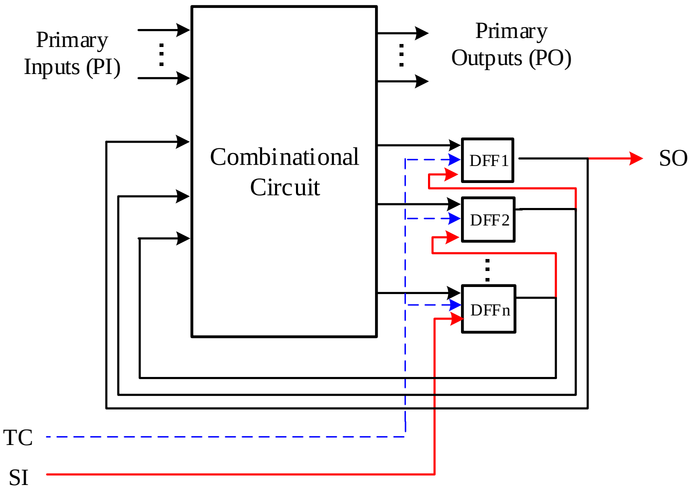


Figure 7.3: Schematic of a sequential circuit.

7.2.3 Non-uniformed Aging

In a typical sequential circuit, not all flip-flops will experience same rate and bias from aging after deployment. The aging of each individual flip-flop will be determined by its aggregate logic state over time (described in Section 7.2.2), which is governed by the relative frequencies of 0s and 1s at its input. It is therefore possible to identify flip-flops which age mostly in the 0 or 1 state from their input controllability measures. The controllability of a node is defined by the probability that it is observed in the 0 or 1 state. For example, the SCOAP testability measure is widely used by test tools [115]. However, testability measures such as SCOAP are only approximate, and when possible, circuit simulations or actual test measurements through the use of scan chains for a large set of test patterns, can more accurately estimate circuit signal probabilities. Importantly, the scan testing proposed here avoids the need for a netlist. Observing a high percentage of logic 0s (1s) in a flip-flop during test indicates that the flip-flop will mostly be found in the 0 (1) state in operation.

Figure 7.3 shows the classical schematic of a sequential circuit where the flip-flop states can be directly accessed through the scan chains. The state probability of each flip-flop can be estimated through scan testing as follows:

- *Step 1*: A random pattern is shifted into the scan chains ($TC = 1$).
- *Step 2*: The response of the combinational part is captured in the FFs ($TC = 0$).
- *Step 3*: The states of the individual FFs are shifted out and recorded ($TC = 1$).

A different random pattern is selected and Steps 1-3 are performed. We apply a large number (e.g., 10,000) of these random patterns to determine the each FFs probability of being in the 1 (0) state using the following equation:

$$Pr(1) = \frac{\#1s \text{ observed in the target FF}}{\#\text{Total patterns}} \tag{7.1}$$

Similarly, the $Pr(0)$ can be calculated by replacing 1s with 0s in Equation 7.1. Note that single cycle operation from random initial states may not precisely mimic continuous functional operation, which can introduce some error in the estimated input controllability values. However, such inaccuracies are allowed due to the inherent statistical nature of the proposed approach.

Figure 7.4 shows the distribution for the input probability of 1 ($Pr(1)$) for all the FFs in b19 benchmark circuit (ITC'99) to demonstrate asymmetric aging. We observe a near normal distribution, where the two tails are of our interest. The first group, denoted as *Group-1*, contains FFs that age with mostly 1, and second group, denoted as *Group-2*, which experiences mostly 0

in operation. As the FFs in *Group-1* are mostly stressed in the 1 state during operation, this group will exhibit an increase in the number of 0s at power-up over time. As a result, the percentage of 1s in this group will reduce. On the other hand, the FFs in *Group-2* will age with 0 and the percentage of 1s in this group at power-up will increase over time. The difference in the percentage of 1s (or 0s) among these two groups will reflect the length of the usage of a chip in the field. Ideally, the percentage of 1 in power-up state for two groups should be exactly the same (given a relatively large number of FFs in each group) for a new part that has not experienced any aging. However, based on the previous discussion, random process variation will have some impact on power-up state of FFs. Consequently, the initial percentage of 1 in power-up state for two groups can have some statistical difference, which scales down as the number of FFs in group increases. We define this difference as Δ. Generally, the value of Δ should be very small (less 2%) if we select a large number of FFs (see details in Section 7.4). In practice, this can be ignored when evaluating recycled chips those have been used for many months of years.

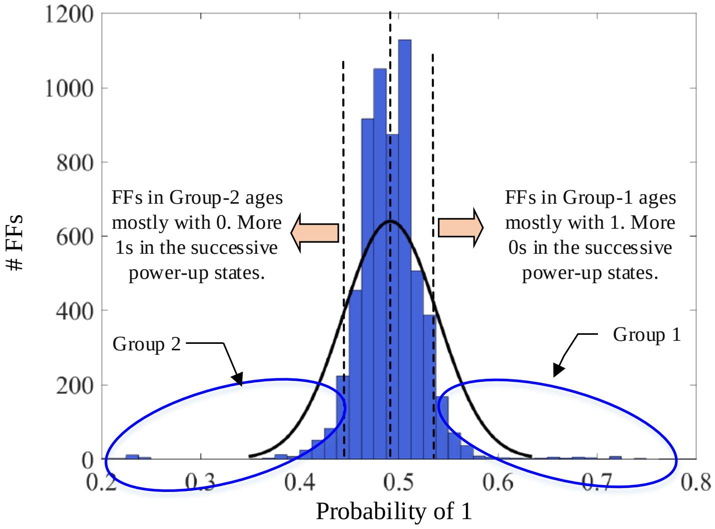


Figure 7.4: Impact of aging on different groups of FFs.

7.3 Proposed Approach for recycled IC Detection based on power-up state of flip-flops

The proposed approach utilizes the power-up state of the scan FFs to determine the prior usage of a chip, and can be effectively used for detecting recycled ICs. However, the power-up state of a FF will be skewed due to its inherent asymmetry. It is necessary to construct two groups of FFs, which mostly ages with 0 and 1, respectively. In this section, we develop a detailed step-by-step process for the detection.

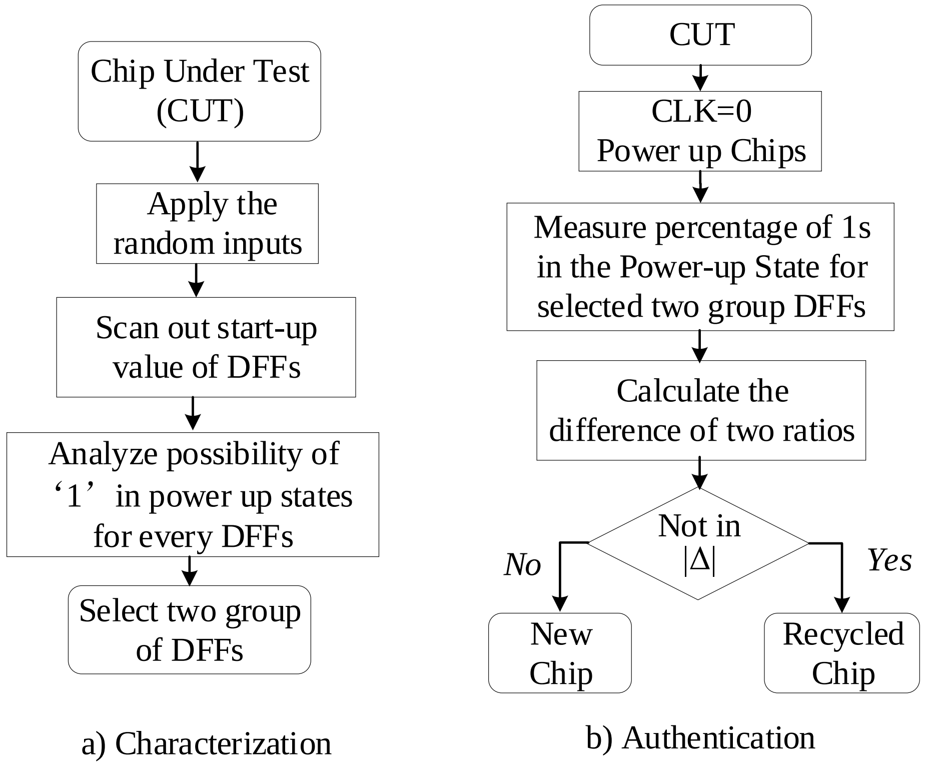

Chip Under Test (CUT)
Apply the random inputs
Scan out start-up value of DFFs
Analyze possibility of ‘1’ in power up states for every DFFs
Select two group of DFFs
a) Characterization
CUT
CLK=0 Power up Chips
Measure percentage of 1s in the Power-up State for selected two group DFFs
Calculate the difference of two ratios
Not in |Δ|
No
Yes
New Chip
Recycled Chip
b) Authentication

Figure 7.5: Proposed method to detect recycled ICs by power-up state of FFs.

Figure 7.5 shows our proposed approach for detecting recycled digital chips, which consists of two phases. In Phase I, the characterization is performed and is shown in Figure 7.5.a. Two groups of FFs are selected for future authentication. The authentication for a chip whether it is new or recycled is performed in Phase II. The details for these two phases are summarised as follows:

- *Phase I – Characterization*: To identify two groups of FFs – one mostly aged with 0 and the other aged with 1 – is primary objective of this phase. We extract the FF input controllability information from the chip under test (CUT). We apply 1000 random input patters, capture the response in FFs and shifted out the state using the scan chains (see the details in Section 7.2.3). Based on the input probabilities, two different groups of FFs are constructed for authentication. *Group-1* is the group of FFs which will most likely age with logical 1 , and *Group-2* is the group of FFs which will experience logical 0 with higher possibilities. Note that both these groups will have similar statistics (e.g., percentage of 1s or 0s) at time zero with a small error, which can be less than 1% when a large number of FFs are selected (see Table 7.2).

- *Phase II – Authentication*: The process of determining a chip being recycled is relatively straight-forward. For any CUT, it is necessary to measure the start-up value of all the FFs. Note that we need to keep the clock at logic 0 ($CLK = 0$) during the power-up so that the slave latch of the DFFs are selected. One can also make $CLK = 1$ to select the master latch. Two groups of FFs are now constructed based on the data from the characterization phase described above. For each group, percentage of 1 are calculated. The difference of percentage of 1s for two group are calculated. If this difference lies within Δ (which is negligible for a large group of FFs and approximately 1%), the chip will be identified as new, otherwise, it is an used/recycled one.

7.4 Simulation Results and Analysis

In order to verify our proposed method of detecting recycled ICs, we perform the HSPICE aging simulation on different benchmark circuits [116]. We use the MOS Reliability Analysis (MOSRA) tool by Synopsys [91] to perform aging analysis. Synopsys 32nm technology library is used for implementing DFF for simulation [117]. MOSFET models are based on 32nm low power metal gate Predictive Technology Model [92]. Aging simulation is performed at 25°C room temperature and nominal supply voltage of 1V. The benchmark circuits are synthesised in Synopsys Design compiler (DC) and IC Validator is used to covert synthesized netlist to SPICE netlist. We use Synopsis VCS to perform logic simulation to compute the probabilities at the input of each FF.

The first experiment is performed to show that the different groups of FFs have similar percentage of 1 before a chip has been deployed in the field. To implement process variation, we add random variation with standard deviation σ of 20 mV to the threshold voltage of each MOS transistors. We measure percentage of 1s for 100 groups of unaged FFs and plot the corresponding distribution. In addition, different FF group sizes are also explored to find out how the percentage of 1s varies among these groups. Figure 7.6 shows the distribution of percentage of 1 for different group sizes. X-axis represents the percentage of 1 in power-up state for every groups, whereas, the Y-axis represents the number of such groups. From the figure, it is obvious that the standard deviation (σ) decreases with the increase of the group size. Note that σ drops to 0.751 from 2.663, when the group sizes are increases to 1000 DFFs from 100. Note that we expect an error of 68%, 95%, and 99.7% when we consider σ, 2σ, and 3σ values [118]. In other words, if we take any two groups

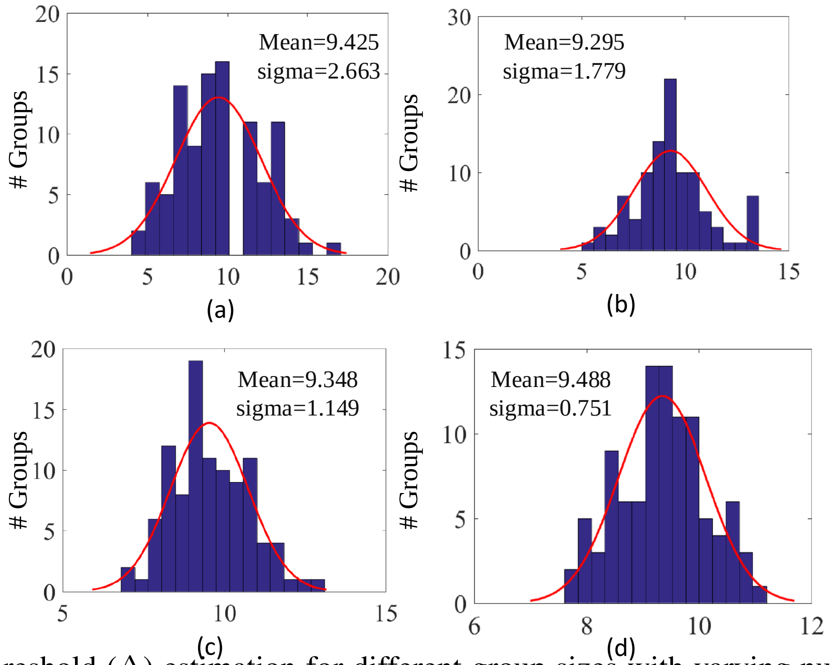


Figure 7.6: Threshold (Δ) estimation for different group sizes with varying number of DFFs. (a) Group size of 100, (b) Group size of 200, (c) Group size of 500, and (d) Group size is 1000.

of DFFs, the similarity of the percentage of 1s among different groups will 68%, 95%, and 99.7% if we consider the threshold (Δ, see Figure 7.5 for details) of σ, 2σ, and 3σ. We choose Δ of 2σ value in detecting recycled ICs. However, one can choose 3σ to increase the confidence.

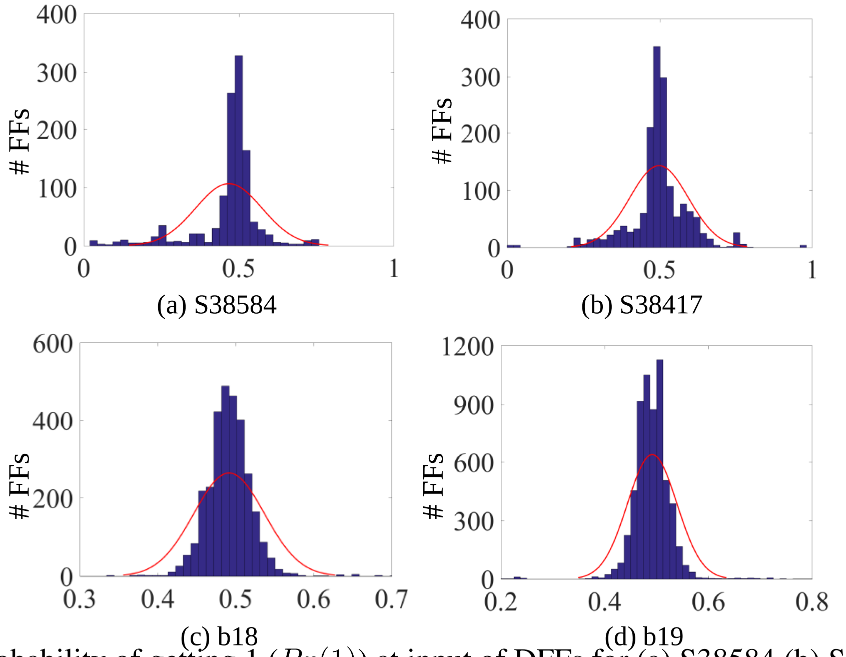


Figure 7.7: Probability of getting 1 ($Pr(1)$) at input of DFFs for (a) S38584 (b) S38417 (c) b18 (d) b19 benchmark circuits.

The second experiment is performed to verify the effectiveness of our proposed method using ITC'99and ISCAS'85 [116] benchmark circuits. We perform the necessary steps to select two groups of FFs, where one group is aged mostly with 0, the other is aged with 1 (see Figure 7.5.a).

Figure 7.7 shows the distribution of the inputs of each FF for different benchmarks circuits. The X-axis represents the probability of getting 1 at the input of a DFF, whereas, the Y-axis represents the number of FFs. We observe Gaussian distributions for all four benchmark circuits. Generally, 2/3 of the FFs in these circuits experience uniform aging with 1 or 0s. On the other hand, 1/3 of the total FFs experience non-uniform aging and those are of our interest for selecting two groups. Figure 7.7.a, shows the distribution for a medium size S38548 benchmark circuit. We have selected 500 FFs from both the tails of the distribution to form the two groups. Figure 7.7.s, shows the distribution for a large b19 benchmark circuit, where we can easily find 1000 FFs for each group. Similar analysis can be performed for Figure 7.7.b and 7.7.c.

Table 7.2 represents the simulation result with aging intervals of 3 months, 6 months and 12 months. As 1/3 of total FFs experience non-uniform aging (see Figure 7.7 for details), we will distribute these FFs in two different groups. As b18 and b19 contains thousands of FFs, it is easy to form these two groups with 1000 FFs. On the other hand, we do not have enough FFs to form groups with 1000 FFs for b22, s38417 and s38548 benchmark circuits. As a result, 500 FFs are assigned to each group. The first column of Table 7.2 denotes the aging duration. The second, third, and forth columns represent the benchmark circuits, the group size, and the 2 threshold value, respectively. The fifth and sixth columns represent the percentage of 1s for group 1 (G_1) and group 2 (G_2), respectively. The last column represents the difference of percentage of 1 for two group after aging.

We can detect the recycled ICs, if the value of last column exceeds 2σ threshold value (shown in the forth column of Table 7.2). For example, the difference of percentages of 1s for G1 and G2 are 7.16%, 9.25%, 9.58% after 3, 6, 12 months of aging, respectively for b19 benchmark circuit. Note that the difference of percentages of 1s for the two groups increases as the chips are used for a longer duration. As the 2σ value (threshold, Δ) is only 1.5% and difference of percentages of 1s for two groups are much greater than Δ, it is safe to conclude that we can detect recycled b19 designs when they have been used at least for three months. The same analysis can be performed for other benchmark circuits. Note that the accuracy of our proposed solution increases with the size of the circuit.

Table 7.2: Power-up value of FFs for used benchmark circuit

Usage (months)	**Bench -marks**	**Group Size**	2σ **Value**	$\%1s$ **for G1**	$\%1s$ **for G2**	**Differ -ence**
3	b22	500	2.3	9.95	13.12	3.17
	b17	500	2.3	6	12.49	6.49
	b18	1000	1.5	9,14	16.3	7.16
	b19	1000	1.5	8.06	15.7	7.64
	s38417	500	2.3	8.14	14.65	6.51
	s38584	500	2.3	10.44	17.89	7.45
6	b22	500	2.3	9.41	15.61	6.2
	b17	500	2.3	5.62	15.01	9.39
	b18	1000	1.5	8.14	17.98	9.84
	b19	1000	1.5	7.86	17.11	9.25
	S38417	500	2.3	8.05	15.08	7.03
	S38584	500	2.3	10.48	20.62	10.14
12	b22	500	2.3	9.27	16.47	7.2
	b17	500	2.3	5.57	15.88	10.31
	b18	1000	1.5	8.12	18.14	10.02
	b19	1000	1.5	7.62	17.2	9.58
	S38417	500	2.3	7.88	16.3	8.42
	S38584	500	2.3	10.38	20.87	10.49

7.5 Conclusion

In this paper, we have proposed a zero-cost approach to detect recycled ICs by observing the power-up states of the scan flip-flops in the chip. The proposed solution performs scan tests on only the target IC, and does not require any other information such as reference data for new parts, or even the netlist. Instead, it uses a differential self-referencing methodology based on two selected groups of FFs in the part, one group that mostly ages in the 0 state, and the other which ages in the 1 state. The percentage of 1s observed at power-up in these two groups is virtually the same when a chip is new, because of the commonality in design and process. However, after aging in use, due to the differential NBTI stress, the percentage of 1s increases for one group and decreases for the other. This creates a differential signal indicating functional use which increases in magnitude over time. Our current future work is focused on implementing this and related solutions in silicon.

Chapter 8

Conclusion and Future Work

This chapter provides the summary of this dissertation and directions for future work.

8.1 Conclusion of Dissertation

With the advancement of ubiquitous distributed computing under the broad umbrella of the Internet of Things (IoT), the number of connected devices (edge devices) is expected to grow exponentially in the coming decades. The physical attack on cyberinfrastructure and electronic edge devices become a critical threat for the system designer and gain more and more attention in recent years. This has to lead an increasing interest in hardware-based security primitives, such as PUF, TRNG, which can provide the Root of Trust (RoT) for the electronic system. The instinct implementation, for example, the SRAM, requires no further hardware resources and can be used to establish a PUF, TRNG, and recycled IC detection. However, the SRAM-based hardware primitives have multiple challenges, which prohibit its practical application in the real world. In this dissertation, we provide the corresponding solutions to those challenges and make SRAM-based hardware primitives more practical.

SRAM arrays are particularly attractive for use as PUFs. However, errors in the PUF response due to instability caused by voltage, temperature, environmental noise, and degradation due to aging is a challenge. In chapter 4, a novel systematic bit selection method has been proposed to address the reliability issue of SRAM-based PUF. We showed that the selected SRAM cells using our proposed method indicate 100% reliability under multiple power-ups, supply voltage level, and temperature.

The vast majority of bits in any SRAM are fully-skewed and therefore, stable at power-up, a large SRAM still has enough randomness (entropy) to support the generation of a random number.

To maintain the randomness (entropy) of the entire SRAM array during deployment is a critical requirement for the SRAM-based TRNG. In chapter 5, we have presented an approach to preserve the entropy of the power-up states of an SRAM array during deployment. We have shown that SRAM entropy can decrease due to device aging while in use. Consequently, SRAM-based RNGs need to manage entropy to ensure high-quality random numbers. However, the strategy that increases the entropy by controlled aging before deployment may have a counterproductive effect. In this chapter, we proposed the strategy that maintains the entropy by exploiting controlled aging during deployment. Our experimental result shows that we can preserve the initial entropy of COTS SRAM chips using periodic compensation by repeatedly powering up the SRAM chip and then holding its power-up state.

The recycled ICs sold as new parts pose a serious security problem due to the shorter lifetime, potentially poorer performance compared to fresh and authentic parts. In chapter 6, we have proposed a novel testing method to differentiate the recycling ICs by exploiting the power-up state of the embedded SRAM array. Our methodology is based on a basic observation which is an unused SRAM will always obtain a virtually equal number of logical 1 and logical 0 on power-up [83]. Afterward, the ratio of logical 1 and logical 0 will be skewed due to biased aging during normal operation. Furthermore, we also improve this testing method in the second part of this chapter. We measure the power-up state of SRAM chips under different ramp rates and a nearly 50% ratio of 1 is still expected/seen for unused chips. This improved testing method allows us to differentiate the recycled SRAM chips across a range of old and new technologies. Our silicons result indicates that the proposed method can detect the most recycled chips.

In chapter 7, we have proposed a highly effective approach for detecting such chips by exploiting the power-up state of on-chip SRAMs. Although this solution is very effective in detecting recycled ICs, its applicability is somewhat limited as a large number of older designs do not have large on-chip memories. In chapter 7, we propose an alternative approach based on the initial power-up state of scan flip-flops, which are present in virtually every digital circuit. The proposed method uses a differential self-referencing methodology based on two selected groups of FFs in the part, one group that most ages in the 0 state, and the other which ages in the 1 state. The percentage of 1s observed at power-up in these two groups is virtually the same when a chip is new, because of the commonality in design and process. However, after aging, due to the differential NBTI stress, the percentage of 1s increases for one group and decreases for the other. The difference in percentage can help us to detect the recycled ICs.

8.2 Future Works

The counterfeit electronics, which contain recycled, remarked, overproduced, cloned, out-of-spec, and forged documentation parts, harshly influence the security of the electronics supply chain. In this dissertation, we proposed an efficient method, which utilizes the ratio of 1 of the startup values of SRAM as a reference to detect recycled counterfeit ICs. The proposed method does not require any data/reference extracted from known fresh parts and is resistant to process variation and different operating conditions. However, the proposed method only can differentiate the recycled ICs. For future works, firstly, it must consider all the types of counterfeit ICs. Techniques that detect remarked ICs are limited. Secondly, the proposed solution should be practical and easily implemented (low cost, less complexity, no required knowledge/data from a known fresh IC sample, and no require sophisticated equipment). The other possible future research direction should be the uniqueness of SRAM-based PUF. In this dissertation, the reliability of SRAM-based PUF has been investigated in a very systematic way and a bit selection method has been proposed to address the reliability challenges of SRAM-based PUF. However, uniqueness is another extremely important parameter to evaluate the overall quality of PUF. Based on the silicon data in this dissertation, a relatively non-negligible systematic bias have been observed for the power-up value of SRAM. This may cause a decrease in the uniqueness of SRAM-based PUF. In future work, a strategy need to be proposed to improve or maximize the uniqueness of SRAM-based PUF.

References

[1] H. Gordon, J. Edmonds, S. Ghandali, W. Yan, N. Karimian, and F. Tehranipoor, "Flash-based security primitives: Evolution, challenges and future directions," *Cryptography*, vol. 5, no. 1, p. 7, 2021.

[2] 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 *Hardware-Oriented Security and Trust (HOST), 2014 IEEE International Symposium on.* IEEE, 2014, pp. 101–106.

[3] D. E. Holcomb, W. P. Burleson, and K. Fu, "Power-up sram state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198–1210, 2008.

[4] A. Miller, Y. Shifman, Y. Weizman, O. Keren, and J. Shor, "A highly reliable sram puf with a capacitive preselection mechanism and pre-ecc ber of 7.4 e-10," in *2019 IEEE Custom Integrated Circuits Conference (CICC).* IEEE, 2019, pp. 1–4.

[5] 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*, vol. 1, no. 6, pp. 138–141, 2018.

[6] M. John, "Intel digital random number generator (drng) software implementation guide," 2018, https://software.intel.com/en-us/articles/intel-digital-random-number-generator-drng-software-implementation-guide.

[7] E. B. Barker and M. Kelsey, John (2015), *Recommendation for random number generation using deterministic random bit generators (revised).* National Institute of Standards and Technology, 2007.

[8] ThermoSpot DCP-201-1010-2 ThermoStream Thermal Inducing System.

[9] W. Trappe, R. Howard, and R. S. Moore (2015), "Low-energy security: Limits and opportunities in the internet of things," *IEEE Security & Privacy*, vol. 13, no. 1, pp. 14–21, 2015.

[10] U. Guin, A. Singh, M. Alam, J. Canedo, and S. A (2018), "A Secure Low-Cost Edge Device Authentication Scheme for the Internet of Things," in *Proc. IEEE International Conference on VLSI Design*, 2018.

[11] U. Guin, C. P, and S. A (2018), "Ensuring Proof-of-Authenticity of IoT Edge Devices using Blockchain Technology," in *Proc. IEEE International Conference on Blockchain*, 2018, pp. 1042–1049.

[12] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1126–1141, 2014.

[13] C. Böhm and M. Hofer, *Physical unclonable functions in theory and practice.* Springer Science & Business Media, 2012.

[14] M. Liu, C. Zhou, Q. Tang, K. K. Parhi, and C. H. Kim, "A data remanence based approach to generate 100% stable keys from an sram physical unclonable function," in *Low Power Electronics and Design (ISLPED, 2017 IEEE/ACM International Symposium on.* IEEE, 2017, pp. 1–6.

[15] M. Bhargava, C. Cakir, and K. Mai, "Reliability enhancement of bi-stable pufs in 65nm bulk cmos," in *2012 IEEE International Symposium on Hardware-Oriented Security and Trust.* IEEE, 2012, pp. 25–30.

[16] S. K. Mathew, S. K. Satpathy, M. A. Anders, H. Kaul, S. K. Hsu, A. Agarwal, G. K. Chen, R. J. Parker, R. K. Krishnamurthy, and V. De, "16.2 a 0.19 pj/b pvt-variation-tolerant hybrid

physically unclonable function circuit for 100% stable secure key generation in 22nm cmos," in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*. IEEE, 2014, pp. 278–279.

[17] S. Satpathy, S. K. Mathew, V. Suresh, M. A. Anders, H. Kaul, A. Agarwal, S. K. Hsu, G. Chen, R. K. Krishnamurthy, and V. K. De, "A 4-fj/b delay-hardened physically unclonable function circuit with selective bit destabilization in 14-nm trigate cmos," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 4, pp. 940–949, 2017.

[18] K. Liu, X. Chen, H. Pu, and H. Shinohara, "A 0.5-v hybrid sram physically unclonable function using hot carrier injection burn-in for stability reinforcement," *IEEE Journal of Solid-State Circuits*, 2020.

[19] D. E. Holcomb, W. P. Burleson, and K. Fu (2009), "Power-up sram state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198–1210, 2009.

[20] V. van der Leest, E. van der Sluis, G.-J. Schrijen, P. Tuyls, and H. Handschuh (2012), "Efficient implementation of true random number generator based on sram pufs," in *Cryptography and Security: From Theory to Applications, springer*, 2012, pp. 300–318.

[21] S. Chen and B. Li (2016), "A dynamic reseeding drbg based on sram pufs," in *Proc. IEEE Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, 2016, pp. 50–53.

[22] A. Van Herrewege, V. van der Leest, A. Schaller, S. Katzenbeisser, and I. Verbauwhede (2013), "Secure prng seeding on commercial off-the-shelf microcontrollers," in *ACM Proc. of the 3rd international workshop on Trustworthy embedded devices*, 2013, pp. 55–64.

[23] K.-F. Krentz, C. Meinel, and H. Graupner (2017), "Secure self-seeding with power-up sram states," in *Proc. IEEE Symposium on Computers and Communications (ISCC)*, 2017, pp. 1251–1256.

[24] L. T. Clark, S. B. Medapuram, and D. K. Kadiyala (2018), “Sram circuits for true random number generation using intrinsic bit instability,” *IEEE Transactions on Very Large Scale Integration Systems (TVLSI)*, no. 99, pp. 1–11, 2018.

[25] M. T. Rahman, D. Forte, X. Wang, and M. Tehranipoor (2016), “Enhancing noise sensitivity of embedded srams for robust true random number generation in socs,” in *Proc. IEEE Asian Hardware-Oriented Security and Trust (AsianHOST)*, 2016, pp. 1–6.

[26] S. Kiamehr, M. S. Golanbari, and M. B. Tahoori (2017), “Leveraging aging effect to improve sram-based true random number generators,” in *IEEE Proc. of the Conference on Design, Automation & Test in Europe*. European Design and Automation Association, 2017, pp. 882–885.

[27] M. M. Tehranipoor, U. Guin, and D. Forte, *Counterfeit Integrated Circuits: Detection and Avoidance*. Springer, 2015.

[28] U. Guin, K. Huang, D. DiMase, J. Carulli, M. Tehranipoor, and Y. Makris, “Counterfeit integrated circuits: A rising threat in the global semiconductor supply chain,” *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1207–1228, Aug 2014.

[29] U. Guin, D. DiMase, and M. Tehranipoor, “Counterfeit integrated circuits: Detection, avoidance, and the challenges ahead,” *Journal of Electronic Testing*, vol. 30, no. 1, pp. 9–23, 2014.

[30] X. Zhang, K. Xiao, and M. Tehranipoor, “Path-delay fingerprinting for identification of recovered ICs,” in *Proc. International Symposium on Fault and Defect Tolerance in VLSI Systems*, October 2012.

[31] K. Huang, J. Carulli, and Y. Makris, “Parametric counterfeit IC detection via Support Vector Machines,” in *Proc. International Symposium on Fault and Defect Tolerance in VLSI Systems*, 2012, pp. 7–12.

[32] Y. Zheng, A. Basak, and S. Bhunia, “CACI: Dynamic current analysis towards robust recycled chip identification,” in *Design Automation Conference (DAC)*, June 2014, pp. 1–6.

[33] H. Dogan, D. Forte, and M. Tehranipoor, “Aging analysis for recycled FPGA detection,” in *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, Oct 2014.

[34] Y. Zheng, X. Wang, and S. Bhunia, “SACCI: Scan-based characterization through clock phase sweep for counterfeit chip detection,” *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, 2014.

[35] Z. Guo, M. T. Rahman, M. M. Tehranipoor, and D. Forte, “A zero-cost approach to detect recycled soc chips using embedded sram,” in *IEEE Int. Symp. on Hardware Oriented Security and Trust*, 2016.

[36] J.-L. Zhang, G. Qu, Y.-Q. Lv, and Q. Zhou, “A survey on silicon pufs and recent advances in ring oscillator pufs,” *Journal of computer science and technology*, vol. 29, no. 4, pp. 664–678, 2014.

[37] M. T. Rahman, D. Forte, F. Rahman, and M. Tehranipoor, “A pair selection algorithm for robust ro-puf against environmental variations and aging,” in *2015 33rd IEEE International Conference on Computer Design (ICCD)*. IEEE, 2015, pp. 415–418.

[38] M. Bhargava, C. Cakir, and K. Mai, “Attack resistant sense amplifier based pufs (sa-puf) with deterministic and controllable reliability of puf responses,” in *2010 IEEE international symposium on hardware-oriented security and trust (HOST)*. IEEE, 2010, pp. 106–111.

[39] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, “Fpga intrinsic pufs and their use for ip protection,” in *International workshop on cryptographic hardware and embedded systems*. Springer, 2007, pp. 63–80.

[40] Y. Li, C.-H. Hwang, T.-Y. Li, and M.-H. Han, “Process-variation effect, metal-gate work-function fluctuation, and random-dopant fluctuation in emerging cmos technologies,” *IEEE Transactions on Electron Devices*, vol. 57, no. 2, pp. 437–447, 2009.

[41] M. T. Rahman, D. Forte, Q. Shi, G. K. Contreras, and M. Tehranipoor (2014), “Csst: Preventing distribution of unlicensed and rejected ics by untrusted foundry and assembly,”

in *Proc. IEEE International symposium on defect and fault tolerance in VLSI and nanotechnology systems (DFT)*, 2014, pp. 46–51.

[42] U. Guin, Q. Shi, D. Forte, and M. M. Tehranipoor (2016), "FORTIS: a comprehensive solution for establishing forward trust for protecting IPs and ICs," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 21, no. 4, p. 63, 2016.

[43] V. Fischer (2012), "A closer look at security in random number generators design," in *Proc. International Workshop on Constructive Side-Channel Analysis and Secure Design*, 2012, pp. 167–182.

[44] S. Srinivasan, S. Mathew, R. Ramanarayanan, F. Sheikh, M. Anders, H. Kaul, V. Erraguntla, R. Krishnamurthy, and G. Taylor (2010), "2.4 ghz 7mw all-digital pvt-variation tolerant true random number generator in 45nm cmos," in *Proc. IEEE Symposium on VLSI Circuits*, 2010, pp. 203–204.

[45] D. Li, Z. Lu, X. Zou, and Z. Liu (2015), "PUFKEY: A high-security and high-throughput hardware true random number generator for sensor networks," *Sensors*, vol. 15, no. 10, pp. 26 251–26 266, 2015.

[46] M. Majzoobi, F. Koushanfar, and S. Devadas (2011), "Fpga-based true random number generation using circuit metastability with adaptive feedback control," in *Proc. International Workshop on Cryptographic Hardware and Embedded Systems, springer*, 2011, pp. 17–32.

[47] V. B. Suresh and W. P. Burleson (2010), "Entropy extraction in metastability-based trng," in *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2010, pp. 135–140.

[48] U. Guin, S. Bhunia, D. Forte, and M. Tehranipoor (2016), "Sma: A system-level mutual authentication for protecting electronic hardware and firmware," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 3, pp. 265–278, 2016.

[49] T. Amaki, M. Hashimoto, and T. Onoye (2015), “An oscillator-based true random number generator with process and temperature tolerance,” in *Proc. IEEE Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2015, pp. 4–5.

[50] M. A. Şarkışla and S. Ergün (2018), “An area efficient true random number generator based on modified ring oscillators,” in *Proc. IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, 2018, pp. 274–278.

[51] ——, “Ring oscillator based random number generator using wake-up and shut-down uncertainties,” in *Proc. IEEE Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, 2018, pp. 104–108.

[52] K. Wang, Y. Cao, C.-H. Chang, and X. Ji (2019), “High-speed true random number generator based on differential current starved ring oscillators with improved thermal stability,” in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 2019, pp. 1–5.

[53] Y. Wang, W.-k. Yu, S. Wu, G. Malysa, G. E. Suh, and E. C. Kan (2012), “Flash memory for ubiquitous hardware security functions: True random number generation and device fingerprints,” in *Proc. IEEE Symposium on Security and Privacy (SP)*, 2012, pp. 33–47.

[54] C. Eckert, F. Tehranipoor, and J. A. Chandy (2017), “Drng: Dram-based random number generation using its startup value behavior,” in *Proc. IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2017, pp. 1260–1263.

[55] F. Tehranipoor, W. Yan, and J. A. Chandy (2016), “Robust hardware true random number generators using dram remanence effects,” in *Proc. IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, 2016, pp. 79–84.

[56] IHS iSuppli, “Top 5 Most Counterfeited Parts Represent a $169 Billion Potential Challenge for Global Semiconductor Market,” 2011.

[57] G-19A Test Laboratory Standards Development Committee, “Test Methods Standard; General Requirements, Suspect/Counterfeit, Electrical, Electronic, and Electromechanical Parts,” 2016, https://saemobilus.sae.org/content/as6171.

[58] G-19CI Continuous Improvement, "Counterfeit Electronic Parts; Avoidance, Detection, Mitigation, and Disposition," 2009, https://saemobilus.sae.org/content/as5553.

[59] CTI, "Certification for Counterfeit Components Avoidance Program," 2011, http://www.cti-us.com/pdf/CCAP101Certification.pdf.

[60] IDEA, "Acceptability of Electronic Components Distributed in the Open Market," 2017, http://www.idofea.org/products/118-idea-std-1010b.

[61] M. Miller, J. Meraglia, and J. Hayward, "Traceability in the age of globalization: A proposal for a marking protocol to assure authenticity of electronic parts," in *SAE Aerospace Electronics and Avionics Systems Conference*, October 2012.

[62] Semiconductor Industry Association (SIA), "Public Comments - DNA Authentication Marking on Items in FSC5962," November 2012.

[63] T.-H. Kim, R. Persaud, and C. Kim, "Silicon odometer: An on-chip reliability monitor for measuring frequency degradation of digital circuits," *Solid-State Circuits, IEEE Journal of*, vol. 43, no. 4, pp. 874–880, April 2008.

[64] X. Zhang, N. Tuzzio, and M. Tehranipoor, "Identification of recovered ICs using fingerprints from a light-weight on-chip sensor," in *Proc. IEEE-ACM Design Automation Conference*, June 2012.

[65] X. Zhang and M. Tehranipoor, "Design of on-chip lightweight sensors for effective detection of recycled ICs," *IEEE Transactions on Very Large Scale Integration Systems*, pp. 1016–1029, 2014.

[66] U. Guin, X. Zhang, D. Forte, and M. Tehranipoor, "Low-cost On-Chip Structures for Combating Die and IC Recycling," in *Proc. of ACM/IEEE on Design Automation Conference*, 2014.

[67] U. Guin, D. Forte, and M. Tehranipoor, "Design of accurate low-cost on-chip structures for protecting integrated circuits against recycling," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 4, pp. 1233–1246, 2016.

[68] K. He, X. Huang, and S. X. D. Tan, “EM-based on-chip aging sensor for detection and prevention of counterfeit and recycled ICs,” in *IEEE/ACM International Conference on Computer-Aided Design*, Nov. 2015, pp. 146–151.

[69] M. Alam, S. Chowdhury, M. Tehranipoor, and U. Guin, “Robust, low-cost, and accurate detection of recycled ICs using digital signatures,” in *IEEE Int. Symposium on Hardware Oriented Security and Trust (HOST)*, 2018.

[70] D. K. Schroder and J. A. Babcock (2003), “Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing,” *Journal of applied Physics, AIP*, vol. 94, no. 1, pp. 1–18, 2003.

[71] V. Reddy, A. T. Krishnan, A. Marshall, J. Rodriguez, S. Natarajan, T. Rost, and S. Krishnan (2005), “Impact of negative bias temperature instability on digital circuit reliability,” *Microelectronics Reliability, Elsevier*, vol. 45, no. 1, pp. 31–38, 2005.

[72] D. K. Schroder, “Negative bias temperature instability: What do we understand?” *Microelectronics Reliability*, 2007.

[73] K.-L. Chen, S. Saller, I. Groves, and D. Scott, “Reliability effects on mos transistors due to hot-carrier injection,” *Electron Devices, IEEE Transactions on*, 1985.

[74] S. Mahapatra, D. Saha, D. Varghese, and P. Kumar, “On the generation and recovery of interface traps in mosfets subjected to nbti, fn, and hci stress,” *Electron Devices, IEEE Transactions on*, 2006.

[75] E. Takeda, Y. Nakagome, H. Kume, N. Suzuki, and S. Asai, “Comparison of characteristics of n-channel and p-channel MOSFET’s for VLSI’s,” *IEEE Transactions on Electron Devices*, 1983.

[76] W. Wang, A. Singh, U. Guin, and A. Chatterjee, “Exploiting power supply ramp rate for calibrating cell strength in sram pufs,” in *2018 IEEE 19th Latin-American Test Symposium (LATS)*. IEEE, 2018, pp. 1–6.

[77] M.-D. Yu and S. Devadas, "Secure and robust error correction for physical unclonable functions," *IEEE Design & Test of Computers*, vol. 27, no. 1, pp. 48–65, 2010.

[78] R. Maes, A. Van Herrewege, and I. Verbauwhede, "Pufky: A fully functional puf-based cryptographic key generator," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2012, pp. 302–319.

[79] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *International conference on the theory and applications of cryptographic techniques*. Springer, 2004, pp. 523–540.

[80] M. Bhargava and K. Mai, "An efficient reliable puf-based cryptographic key generator in 65nm cmos," in *2014 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2014, pp. 1–6.

[81] L. Kusters, T. Ignatenko, F. M. Willems, R. Maes, E. van der Sluis, and G. Selimis, "Security of helper data schemes for sram-puf in multiple enrollment scenarios," in *2017 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2017, pp. 1803–1807.

[82] R. Maes, V. Rozic, I. Verbauwhede, P. Koeberl, E. Van der Sluis, and V. van der Leest, "Experimental evaluation of physically unclonable functions in 65 nm cmos," in *2012 Proceedings of the ESSCIRC (ESSCIRC)*. IEEE, 2012, pp. 486–489.

[83] 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)*. IEEE, 2019, pp. 72–80.

[84] R. Maes and V. Van Der Leest, "Countering the effects of silicon aging on sram pufs," in *2014 IEEE International symposium on hardware-oriented security and trust (HOST)*. IEEE, 2014, pp. 148–153.

[85] E. I. Vatajelu, G. Di Natale, and P. Prinetto, "Towards a highly reliable sram-based pufs," in *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2016*. IEEE, 2016, pp. 273–276.

[86] W. Wang, U. Guin, and A. Singh, “Aging-resilient sram-based true random number generator for lightweight devices,” *Journal of Electronic Testing*, vol. 36, pp. 301–311, 2020.

[87] D. Wei, L. Deng, P. Zhang, L. Qiao, and X. Peng (2016), “Nrc: A nibble remapping coding strategy for nand flash reliability extension,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 11, pp. 1942–1946, 2016.

[88] M. Cortez, A. Dargar, S. Hamdioui, and G.-J. Schrijen, “Modeling sram start-up behavior for physical unclonable functions,” in *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 1–6.

[89] E. Barker and J. Kelsey (2007), “Recommendation for random number generation using deterministic random bit generators,” *NIST Special Publication*, vol. 800, p. 90A, 2007.

[90] C. E. Shannon (1951), “Prediction and entropy of printed english,” *Bell system technical journal*, vol. 30, no. 1, pp. 50–64, 1951.

[91] B. Tudor, J. Wang, W. Liu, and H. Elhak (2011), “Mos device aging analysis with hspice and customsim,” *Synopsys, White Paper*, 2011.

[92] Predictive Technology Model (PTM), http://ptm.asu.edu/.

[93] I. Baturone, M. A. Prada-Delgado, and S. Eiroa (2015), “Improved generation of identifiers, secret keys, and random numbers from srams,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2653–2668, 2015.

[94] J. Kim, J. Lee, and J. A. Abraham (2010), “Toward reliable sram-based device identification,” in *2010 IEEE International Conference on Computer Design*, 2010, pp. 313–320.

[95] M. A. Alam, H. Kufluoglu, D. Varghese, and S. Mahapatra (2007), “A comprehensive model for pmos nbti degradation: Recent progress,” *Microelectronics Reliability*, vol. 47, no. 6, pp. 853–862, 2007.

[96] M. A. Alam and S. Mahapatra (2005), “A comprehensive model of pmos nbti degradation,” *Microelectronics Reliability*, vol. 45, no. 1, pp. 71–81, 2005.

[97] R. Vattikonda, W. Wang, and Y. Cao (2006), "Modeling and minimization of pmos nbti effect for robust nanometer design," in *2006 43rd ACM/IEEE Design Automation Conference*, 2006, pp. 1047–1052.

[98] 7 Series FPGAs Memory Resources User Guide, https://www.xilinx.com/support/documentation/user_guides/ug473_7Series_Memory_Resources.pdf.

[99] A. Asenov, "Simulation of statistical variability in nano MOSFETs," in *Proc. IEEE Symposium on VLSI Technology*, 2007, pp. 86–87.

[100] R. Rao, A. Srivastava, D. Blaauw, and D. Sylvester, "Statistical estimation of leakage current considering inter-and intra-die process variation," in *Proc. International Symposium on Low Power Electronics and Design*, 2003, pp. 84–89.

[101] K. J. Kuhn, M. D. Giles, D. Becher, P. Kolar, A. Kornfeld, R. Kotlyar, S. T. Ma, A. Maheshwari, and S. Mudanai, "Process technology variation," *IEEE Transactions on Electron Devices*, vol. 58, no. 8, pp. 2197–2208, 2011.

[102] C. Shin, X. Sun, and T.-J. K. Liu, "Study of random-dopant-fluctuation (RDF) effects for the trigate bulk MOSFET," *IEEE Transactions on Electron Devices*, vol. 56, no. 7, pp. 1538–1542, 2009.

[103] A. Asenov, S. Kaya, and A. R. Brown, "Intrinsic parameter fluctuations in decananometer MOSFETs introduced by gate line edge roughness," *IEEE Transactions on Electron Devices*, vol. 50, no. 5, pp. 1254–1260, 2003.

[104] K. Kuhn, C. Kenyon, A. Kornfeld, M. Liu, A. Maheshwari, W.-k. Shih, S. Sivakumar, G. Taylor, P. VanDerVoorn, and K. Zawadzki, "Managing process variation in Intel's 45nm CMOS technology." *Intel Technology Journal*, vol. 12, no. 2, 2008.

[105] Microchip 23A640/23K640: 64K SPI Bus Low-Power Serial SRAM, https://www.mouser.com/datasheet/2/268/22126B-54007.pdf.

[106] Microchip 23A640/23K640: 64K SPI Bus Low-Power Serial SRAM, http://ww1.microchip.com/downloads/en/DeviceDoc/22126E.pdf.

[107] PCA9306 Dual Bidirectional I2C Bus and SMBus Voltage-Level Translator,http://www.ti.com/lit/ds/symlink/pca9306.pdf.

[108] C. I. Circuits, "A rising threat in the global semiconductor supply chain, ujjwal guin," in *Proceedings of the*, 2014.

[109] X. Zhang, K. Xiao, and M. Tehranipoor, "Path-delay fingerprinting for identification of recovered ics," in *2012 IEEE International symposium on defect and fault tolerance in VLSI and nanotechnology systems (DFT)*. IEEE, 2012, pp. 13–18.

[110] L. Ding and P. Mazumder, "The impact of bit-line coupling and ground bounce on cmos sram performance," in *16th International Conference on VLSI Design, 2003. Proceedings.* IEEE, 2003, pp. 234–239.

[111] K. L. Shepard and V. Narayanan, "Noise in deep submicron digital design," in *Proceedings of International Conference on Computer Aided Design*. IEEE, 1996, pp. 524–531.

[112] B. B. Talukder, F. Ferdaus, and M. T. Rahman, "Memory-based pufs are vulnerable as well: A non-invasive attack against sram pufs," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 4035–4049, 2021.

[113] W. Wang, U. Guin, and A. Singh, "A zero-cost detection approach for recycled ics using scan architecture," in *2020 IEEE 38th VLSI Test Symposium (VTS)*. IEEE, 2020, pp. 1–6.

[114] U. Guin, W. Wang, C. Harper, and A. D. Singh, "Detecting recycled socs by exploiting aging induced biases in memory cells," in *IEEE Int. Symposium on Hardware Oriented Security and Trust (HOST)*, 2019, pp. 72–80.

[115] M. Bushnell and V. Agrawal, *Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits*. Springer Science & Business Media, 2004, vol. 17.

[116] ISCAS-85 Benchmark Circuits, http://www.pld.ttu.ee/ maksim/benchmarks/iscas85/.

[117] Synopsys 32/28nm Generic Library for Teaching IC Design, https://www.synopsys.com/COMMUNITY/UNIVERSITYPROGRAM/ Pages/32-28nm-generic-library.aspx.

[118] D. J. Wheeler, D. S. Chambers *et al.*, *Understanding statistical process control.* SPC press, 1992.