Spring March 2015

MANAGING AND LEVERAGING VARIATIONS AND NOISE IN NANOMETER CMOS

Vikram B. Suresh
University of Massachusetts - Amherst

MANAGING AND LEVERAGING VARIATIONS AND NOISE IN NANOMETER CMOS

A Dissertation Presented

by

VIKRAM B. SURESH

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

February 2015

Electrical and Computer Engineering

MANAGING AND LEVERAGING VARIATIONS AND NOISE IN NANOMETER CMOS

A Dissertation Presented

by

VIKRAM B. SURESH

Approved as to style and content by:

Wayne P. Burleson, Chair

Sandip Kundu, Member

Christof Paar, Member

Anna Liu, Member

Christopher V. Hallot, Head
Electrical and Computer Engineering

ACKNOWLEDGMENTS

As I sit down to write the last few lines of my thesis document, I realize it is the journey that has been more satisfying than the destination. The next couple of hundred pages of this document contain the work I consider to be my contribution to the field of engineering that I love. But, what I take with me is much more than the technical expertise. The last five years have imbibed in me the qualities to aid my growth not just as an engineer, but as a person. The virtues of patience, perseverance and dedication put into this work will remain with me for the rest of my life. I would like to thank all the people involved in this journey.

Firstly I would like to thank my parents Mr. Suresh. B. S and Late. Sudha Suresh for their constant support and encouragement throughout my life. They have always respected my career decisions and I am happy to have proved them right. I thank my younger sister Manasa for always keeping me motivated about pursuing higher education. I hope to reciprocate the same influence as she embarks on a similar journey in her graduate life.

I am immensely grateful to my advisor Prof. Wayne Burleson for having the trust and confidence in accepting me into his research group. I appreciate the technical advice and the freedom he offered me to explore a number of interesting research problems throughout my stint at UMass. I will always remember his advice to appreciate and give due credit to others work. I would like to thank Prof. Sandip Kundu for being a wonderful mentor and a source of inspiration at UMass and in future as well. His depth of knowledge and hard work have amazed and motivated me during every interaction. I thank Prof. Christof Paar and Prof. Anna Liu for agreeing to

serve on my dissertation committee and for you valuable guidance and input to my thesis.

The VLSI Circuits and Systems Group's lab is the single place where I have spend most of my life as a graduate student. I thank my fellow lab mates for creating such a wonderful atmosphere for everyone in the group. Special thanks to Basab Datta, Jinwook Jang and Ibis Benito for guiding me during the initial phase of my research life. To Georg Becker and Gesine Hinterwaelder for all the wonderful times in the lab, daily lunch and the numerous technical and non-technical discussions. I would like to thank all my friends in Amherst; the ones I knew for ages and the many new ones I met here. Special thanks to Pavan with whom I started my application process, shared a house for two years and who has always been someone I could turn to when in need. I will always cherish the social interactions, birthday parties and cricket matches in Amherst.

Last, but most importantly I thank my wife Priyamvada for being the backbone of my PhD life. All this would not have been possible without her constant encouragement and support. Her immense confidence in my abilities and motivation during my slumps gave me the confidence to go through the ups and downs of graduate life. I was also fortunate to have her technical feedback and reviews of my papers. I am very lucky to have her in my life.

I finally thank the numerous other people who have contributed to my professional, educational and personal life from my childhood to this day.

ABSTRACT

MANAGING AND LEVERAGING VARIATIONS AND NOISE IN NANOMETER CMOS

FEBRUARY 2015

VIKRAM B. SURESH

B.E., VISHVESHWARIAH TECHNOLOGICAL UNIVERSITY, INDIA

M.S., UNIVERSITY OF MASSACHUSETTS AMHERST

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Wayne P. Burleson

Advanced CMOS technologies have enabled high density designs at the cost of complex fabrication process. Variation in oxide thickness and Random Dopant Fluctuation (RDF) lead to variation in transistor threshold voltage (V_{th}). Current photo-lithography process used for printing decreasing critical dimensions result in variation in transistor channel length and width. A related challenge in nanometer CMOS is that of on-chip random noise. With decreasing threshold voltage and operating voltage; and increasing operating temperature, CMOS devices are more sensitive to random on-chip noise in advanced technologies.

In this thesis, we explore novel circuit techniques to manage the impact of process variation in nanometer CMOS technologies. We also analyze the impact of on-chip noise on CMOS circuits and propose techniques to leverage or manage impact of noise based on the application. True Random Number Generator (TRNG) is an interesting cryptographic primitive that leverages on-chip noise to generate random bits;

however, it is highly sensitive to process variation. We explore novel metastability circuits to alleviate the impact of variations and at the same time leverage on-chip noise sources like Random Thermal Noise and Random Telegraph Noise (RTN) to generate high quality random bits. We develop stochastic models for metastability based TRNG circuits to analyze the impact of variation and noise. The stochastic models are used to analyze and compare low power, energy efficient and lightweight post-processing techniques targeted to low power applications like System on Chip (SoC) and RFID. We also propose variation aware circuit calibration techniques to increase reliability. We extended this technique to a more generic application of designing Post-Si Tunable (PST) clock buffers to increase parametric yield in the presence of process variation. Apart from one time variation due to fabrication process, transistors undergo constant change in threshold voltage due to aging/wear-out effects and RTN. Process variation affects conventional sensors and introduces inaccuracies during measurement. We present a lightweight wear-out sensor that is tolerant to process variation and provides a fine grained wear-out sensing. A similar circuit is designed to sense fluctuation in transistor threshold voltage due to RTN. Although thermal noise and RTN are leveraged in applications like TRNG, they affect the stability of sensitive circuits like Static Random Access Memory (SRAM). We analyze the impact of on-chip noise on Bit Error Rate (BER) and post-Si test coverage of SRAM cells.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Motivation

In today's world of increasing connectivity and the *Internet of Things*, computing devices are omni-present ranging from enterprise servers and data centers to small household electronic appliances. People interact with and use a variety of electronic devices for computation, storage, communication and entertainment. Smart phones and hand held mobile devices are becoming more of a necessity than luxury. A major factor enabling these technological changes is the development of semiconductor technology. In particular, Complimentary Metal Oxide Semiconductor (CMOS) devices have enabled the design of high speed and energy efficient computing devices at affordable cost. CMOS devices provide great flexibility is designing a variety of circuits and systems. They are used to design high performance multi-core processors and graphic cores operating in frequency ranges of 3-4GHz. They are also used to design low power and energy efficient System-on-Chip (SoC) for battery powered mobile applications. On the other end of the spectrum, CMOS technology is also used to design extremely lightweight and ultra-low power devices like Radio Frequency Identification (RFID) tags and implantable medical devices.

A major factor affecting the viability and cost of CMOS devices in the development in CMOS fabrication technology. The CMOS technology follows *Moore's Law*, a prediction by Gordon Moore that the number of transistors on a chip double every 18 months. Device manufacturers have been trying to keep pace with Moore's law to advance technology every 18 months and shrink the critical dimension of transistor,

also known as the technology node, by 2X with each technology. The current commercially available Intel microprocessors are fabricated in 22nm technology node. The shrinking geometry has enabled the design of complex circuits and systems consisting of billions of transistors. However, as the CMOS fabrication technology approaches the physical limits, the complexity of fabrication process is increasing many folds. This complexity is manifesting in the form of yield loss during fabrication process, decrease in reliability of the fabricated devices and large variations in the performance of circuits, Figure 1.1. Devices fabricated in advanced technology nodes are also highly sensitive to noise and temperature. Researchers are exploring a number of silicon and non-silicon technologies for the post-silicon era of semiconductor devices. These may use different device structures like silicon nanowires, carbon nanotubes or graphene. However, none of these technologies have advanced enough to provide high yield and affordable large volume manufacturing. As a result, CMOS circuit designers face the challenge of managing variations and noise in the existing fabrication technology.

Managing variation and noise in nanometer CMOS technologies require various circuit and architectural techniques. The circuit techniques will further depend on nature of the circuit and its application. These techniques come at the cost of increased silicon area and power consumption. In the last decade, electronic applications have seen a paradigm shift from high performance application to mobile/battery powered application. Personal computing is no longer limited to desktop computers. Laptops, tablets and smart phones are expected to provide a similar user experience, quality of communication and large data storage. The increasing popularity of these applications has necessitated low power and energy efficient circuit design. This adds to constraints of circuit and architectural solution to compensate for process variation and noise. Another class of low power devices include RFID tags and smart cards. RFID tags are passively power devices used for identification in industrial supply

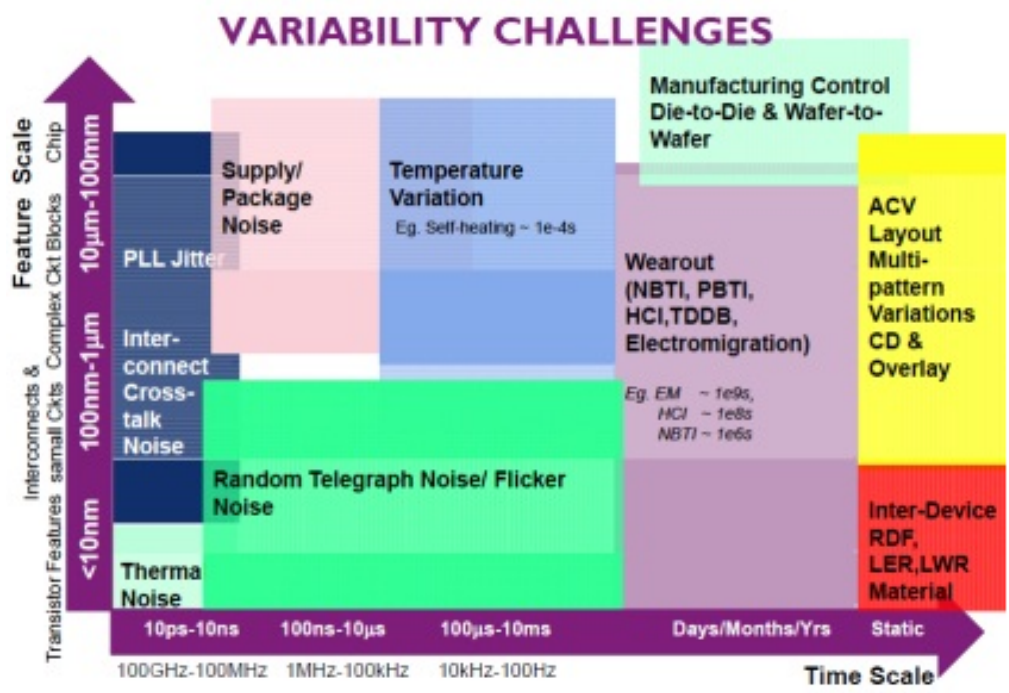


Figure 1.1. Variability and Noise in Advanced CMOS Technologies (Source Imec Technology Forum

chain management. Smart cards are similar to conventional banking and access cards with magnetic strips, but provide a higher level of security. These devices do not only need to be ultra-low powered, but also be extremely low cost for mass deployment. They can not afford extensive post-fabrication testing to identify and debug issues due to variation. Although these devices are currently fabricated in older and stable technology processes, RFID applications with on-chip computation (Computation-RFID) and high speed wireless communication will soon need to be fabricated in sub-65nm process technologies.

Process variation has been an important topic of research for sub-65nm CMOS technologies. Even with great advancement in CMOS fabrication process, shrinking geometries of transistors and interconnects; decreasing dopant concentrations; and reduced oxide thickness to decrease device threshold voltages have all resulted in

increasing process variation. The impact of process variation on circuit behavior depends on the nature of the circuit. Conventional digital logic circuits experience large variation in path delays compared to estimated values during design; Mixed signal circuits like sense amplifiers experience large off-set voltage; Memory circuits like Static Random Access Memory (SRAM) have stability issues due to process variation. Metastability based True Random Number Generator (TRNG) circuits are mixed-signal circuits whose statistics are affected by process variation. TRNG circuits and variation tolerance mechanisms for TRNG circuits will be explored in detail in this work. There are also applications which leverage process variation. Physically Unclonable Functions (PUF) are circuits that harness random process variation to generate unique keys and ids for chips. These circuits require large process variation for stable operation. Apart from one-time variation due to fabrication process, CMOS devices also undergo constant variation due to operating conditions - temperature and supply voltage. They also undergo variation due to aging effects like Negative Bias Temperature Instability (NBTI), Hot Carrier Injection (HCI) and Time Dependent Dielectric Breakdown (TDDB).

Another emerging challenge for digital and mixed signal circuit design in nanometer CMOS is that of noise. The most commonly observed noise in CMOS devices are thermal noise and shot noise. Thermal noise has been studied and understood for a long time since the time it was introduced by Johnson and Nyquist in 1928. Traditionally, the impact of thermal noise was seen only in analog and RF circuits. However, with decreasing threshold voltages and increased device sensitivity, mixed signal circuits in advanced CMOS technologies are also affected by device noise. Random thermal noise can be leveraged in circuits like TRNG to generate random bits. However, the circuits used to sample and digitize thermal noise need to highly sensitive to noise and tolerate impact of process variation. While thermal noise is advantageous for certain applications, it causes reliability concerns for memory circuits like SRAM.

The randomness of thermal noise introduces uncertainties during silicon test and random bit errors during normal SRAM operation. Random telegraph Noise (RTN) is a new source of random noise observed in nano-CMOS devices. RTN is a good source of randomness for low power TRNG circuits. However, it can affect the test coverage and bit stability in highly biased SRAM bit cells.

Variations and noise are inherent part of current CMOS technologies. Variations occur during the fabrication process as well as during normal operation of circuits. As a result, it is important to understand, model and mitigate impact of process variation on reliability and performance of CMOS circuits. The mitigation techniques can range from circuit and architectural modifications to logic or algorithmic error correction. Apart from designing circuits to tolerate variations, it is also critical to detect static and dynamic variations due to process and aging effects respectively. Similarly, it is important to understand the impact of noise on CMOS devices. Circuits that leverage noise need be designed sensitive enough to sample noise even in the presence of process variation. The impact of noise on stability of circuits like SRAM need to analyzed to design appropriate counter measures.

1.2 Contributions and Organization

The main contribution of this dissertation are:

1. Variation tolerant and noise sensitive circuit design of True Random Number Generators (TRNG).

2. Logic techniques to correct bias in TRNG circuits due to process variation.

3. Circuit techniques for variation tolerance in conventional digital data path.

4. Sensing variations in threshold voltage due to aging and noise.

5. Impact of on-chip random noise on SRAM testing and operation.

In chapter 2, we discuss in detail the main sources of variation in nanometer CMOS and current state-of-art techniques for variation aware/tolerant circuit design. We also discuss random on-chip noise sources and their impact on nanometer SRAM circuits. In chapter 3, we discuss a novel variation tolerant metastability circuits for random number generation. The proposed circuits samples thermal noise and RTN in the form of resolution time to generate high quality random bits. We also present a variation aware technique for design of Post-Si Tunable clock buffers (PST buffers) to redistribute data path slack and reduce parametric yield loss due to process variation. Chapter 4 discusses low power logic techniques for bias correction in TRNG due to process variation. We present an entropy and energy bound for low power post-processing and propose REFLEX- a novel reconfigurable logic for entropy extraction in TRNG. In chapter 5, we present on-chip statistical tests to sense the bias in TRNGs and hence the degree of process variation. We also present a lightweight, fine grained sensor to detect threshold voltage variations due to aging effects and RTN. In chapter 6, we discuss the impact of random on-chip noise on SRAM testing. Random thermal noise and telegraph noise increase uncertainty in SRAM test results. We propose a new metric to estimate the probabilistic test coverage and propose two techniques to improve confidence of post-Si SRAM test. We also analyze the impact of thermal noise on SRAM Bit Error Rate (BER) and propose countermeasures.

CHAPTER 2

BACKGROUND

2.1 Variation

Process variation is one of the biggest challenges in nanometer CMOS design. Transistors in advanced technology nodes are sensitive to variations in Process, Voltage and Temperature; more commonly known as PVT variation [10, 11]. The impact of process variation was observed mostly in the sub-90nm era. Subsequently a number of circuit and system techniques were introduced to mitigate impact of process variation. Deterministic timing/power/yield analysis techniques were replaced by statistical methods to incorporate the effect of process variation.

2.1.1 Sources of Variation

The two main sources of process variation are Random Dopant Fluctuation (RDF) and Optical Lithography. While RDF causes variation in transistor threshold voltage, the optical lithography process causes variation in transistor length and width. Other sources of transistor parameter variations include variation in oxide thickness and contact resistance.

Random Dopant Fluctuation (RDF) is the variation in number and position of dopant atoms in the MOSFET channel [71]. These varaitions affect the gate to source voltage of the MOSFET at which channel is formed. In other words, RDF results in variation of transistor threshold voltage. The physical phenomenon of RDF has been prevalent even in older technology nodes. However, the σ/μ of threshold voltage due to RDF was negligibly small to have any impact on the performance or reliability

of circuits. With advanced technology nodes, smaller transistor channel length and width have led to fewer dopant atoms in the channel, as shown in Figure 2.1. As a result, even a small deviation in the number of dopant results in significant change in threshold voltage [50, 107].

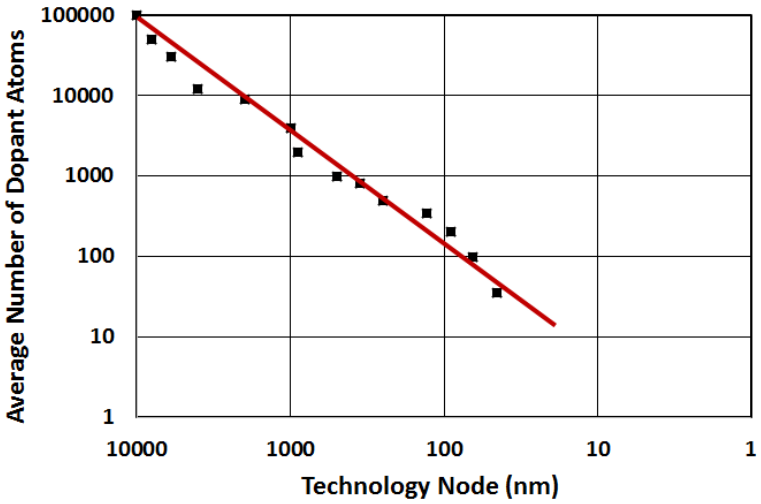


Figure 2.1. Decreasing dopant atoms at advanced technology nodes [50]

RDF and its impact on threshold voltage depends on channel length and designed threshold voltage of the device. A single process technology may offer transistors of different lengths and threshold voltages specifically to design high performance and low power circuits. Longer channel devices have lesser impact of RDF. Higher V_t devices have fewer dopant atoms in the channel and hence more sensitive to RDF. The threshold voltage variation due to RDF follows a Gaussian distribution [107]. The drain current for an NMOS device in saturation mode for super-threshold and sub-threshold operation are given by,

$$Id_{sat} = \begin{cases} \mu_n Cox \frac{W}{2L} (V_{gs} - V_t)^2 & \text{super-threshold} \\ \mu_n C_{ox} \frac{W}{L} (n-1) \frac{k^2 T^2}{q^2} \exp\left(\frac{(V_{gs}-V_t)q}{n(kT)}\right) \left[1 - \exp(\frac{-V_{ds}q}{kT})\right] & \text{sub-threshold} \end{cases}$$

Threshold voltage has a quadratic impact on transistor current in super-threshold operation. In conventional digital logic, the delay of standard cells vary by up to 5% for a σ/μ of 30% in V_t in 25nm technology [60]. However, in sub-threshold operation, the transistor current is exponentially related to V_t. As a result, even a minor variation in threshold voltage can have a significant impact on performance and reliability in sub-threshold operation. The σ/μ of I_{on} can exceed 100% due to RDF in sub-threshold operation [128]. RDF also has a significant impact on matched device circuits like SRAM bit cells and sense amplifiers. Threshold voltage variation is SRAM cells result in read and write stability issues [13, 17, 27, 32]. Similarly, RDF increase offset voltage of sense amplifier; there by increasing the read time of SRAM arrays. A similar matched device circuit is True Random Number Generator (TRNG) using metastable circuits. RDF introduced mismatch in devices of the metastable cell and biases the circuit to generate deterministic outputs [98]. Apart from performance and reliability, decrease in threshold voltage due to RDF also increases sub-threshold leakage. This is a major concern for low power/energy applications.

Variation due to optical lithography is emerging as another major challenge in advanced CMOS technologies [36]. The wavelength of light source used in lithography process has not scaled at the same rate as transistor feature size, leading to sub-wavelength lithography [59]. Currently, all known high volume CMOS manufacturing processes use light source of wavelength 193nm to print feature sizes up to 22nm. As a result, printed patterns on wafer suffer from edge placement error (EPE) relative to patterns in mask layout. Although Extreme Ultra-Violet (EUV) based lithography is a potential solution for technology nodes 14nm and beyond; its adoption does not seem viable in the foreseeable future. Hence, a number of Resolution Enhancement Techniques (RET) like Optical Proximity Correction (OPC), Off-Axis Illumination (OAI) and Phase Shift Masking (PSM) are are used in the current lithography process. These techniques can only help in reducing thevariations introduced by lithography,

but cannot completely eliminate them. Variations mainly occur due to variation in exposure dose and focus during optical lithography. These variations impact the critical dimensions of transistors and interconnect as shown in Figure 2.2. This affects both performance and yield [91].

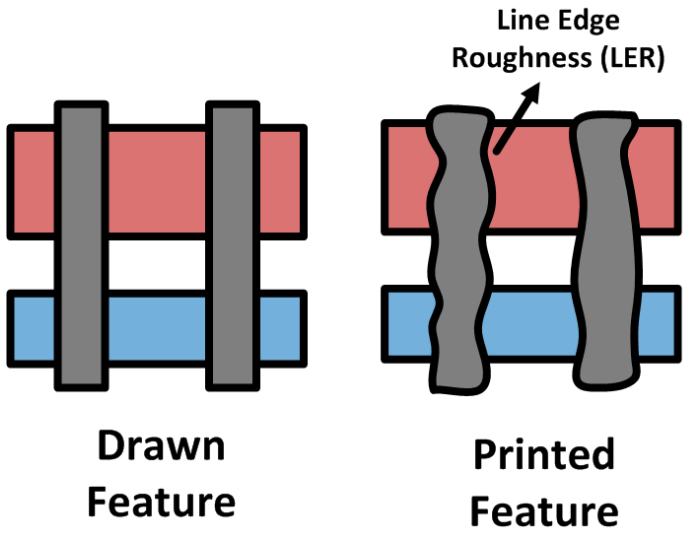


Figure 2.2. Polysilicon Line Edge Roughness (LER) due to lithography variation

Variation in process have different levels of granularity. These can be lot-to-lot variation, wafer-to-wafer variation, inter-die and intra-die variations [116]. The granularity depends on the source of variation. Even if the same recipe is used for fabricating each lot and wafer, variations may occur during lithography process due to wafer thickness and tilt. These variations affect all dies fabricated on the same wafer. Other variations in exposure or focus of mask can lead to inter-die variation. Same gates or transistors on different dies may have different amounts of variation even if they are fabricated on the same wafer. Intra-die variations occur due to lithography variations and RDF. Intra-die variations largely affect matched device circuits causing performance and stability issues. Apart from variation due to fabrication process, CMOS device parameters undergo constant wearout effects

which cause threshold voltage shift. Negative Bias Temperature Instability (NBTI) in PMOS devices and Positive Bias Temperature Instability (PBTI) in NMOS devices increase device threshold voltage during chip lifetime [39, 83, 89]. A related aging effect is Hot Carrier Injection (HCI) [58]. Current high density CMOS circuits are impacted by significant local heating due to large activity factors. This increase in operating temperature further accelerates aging effects causing serious concerns for chip performance and reliability.

2.1.2 Variation Tolerant Design and Methodology

Process variation is inherent to current CMOS fabrication technologies. As a result, architects and circuit designers have been exploring various techniques to design variation tolerant circuits and systems. Techniques to compensate for process variation range from architectural changes to circuit implementation. Architectural techniques for variation tolerance require the chip to be able to operate in different voltage and frequency values. Post fabrication, depending on the process corner, V_{min} and F_{max} for each chip may vary. The chip is tuned accordingly to increase overall yield. Since different functional units on a single chip may have different degrees of process variation, variation tolerant architectures use scheduling of functional units with variable performance numbers [76]. Another technique to manage variation is to provide separate clock domains for different functional blocks or cores and managing data flow asynchronously at the top level. Such architectures are said to be Globally Asynchronous Locally Synchronous (GALS) [61]. A critical part of processors and SoCs that are affected by intra-die variations are cache. Special architectural techniques are used to detect faulty SRAM bit cells and replace them dynamically [3, 2].

Adaptive Body Biasing (ABB) was one of the initial circuit techniques used to counter threshold voltage variation [114]. Decreasing body bias helps in reducing

threshold voltage for devices affected by RDF. This technique compensates for performance degradation due to process variation. Similarly increasing body bias increases threshold voltage for devices whose V_t is decreased due to RDF. This helps in reducing leakage power. As the transistor threshold voltage decreased and the difference between threshold voltage and supply voltage shrunk, the impact of body biasing also diminished. Adaptive circuit tuning mechanisms emerged as the new effective solution to counter process variation [44]. SRAM circuits use adaptive self-repair [72] and selective word line boosting to improve stability in the presence of process variation [81]. Similar word line boost techniques are also used to counter variations introduced by device aging [85]. Physical layout implementations and regularity of features also help in reducing variations due to optical lithography [57].

Apart from circuit and architectural techniques, design methodologies have also evolved to account for variability. The prominent change in design methodology and analysis has been with respect to statistical techniques to model impact of process variation. Performance and power evaluations are performed by incorporating impact of PVT variations [40]. A paradigm shift in static timing analysis of conventional digital logic included parameter variations to perform statistical static timing analysis (SSTA) [7, 37]. The statistical timing techniques estimate variations depending on logic depth, physical proximity of gates [126] and impact of lithography variations [92]. Statistical SRAM design techniques include probabilistic stability analysis and design optimization [77].

2.2 On-chip Random Noise

2.2.1 Thermal Noise

Thermal noise is statistical fluctuation of electric charge caused by thermal agitation of atoms. It has a Gaussian distribution and a constant power spectral density across wide range of frequencies. The first observation and study of this random po-

tential in conductors was reported by Johnson [38] and Nyquist [79] in 1928. With the advancement in semiconductor devices and integrated circuit design, thermal noise is predominant in most circuits. It is seen as a major hindrance to extending Moores law into very deep sub-micron technologies [47]. Traditionally, impact of thermal noise has been seen only on analog and RF circuits. Signal to Noise Margin (SNM) is one of the most important parameters of analog amplifiers. Statistical techniques [109] and circuit models [88] have been studied to better incorporate the effect of thermal noise in transistors. A distribution of thermal noise for CMOS transistors simulated in 32nm Predictive Technology Model is shown in Figure 2.3. At 100C, thermal noise potential has a Gaussian distribution with mean 0V and a standard deviation of 3.1mV.

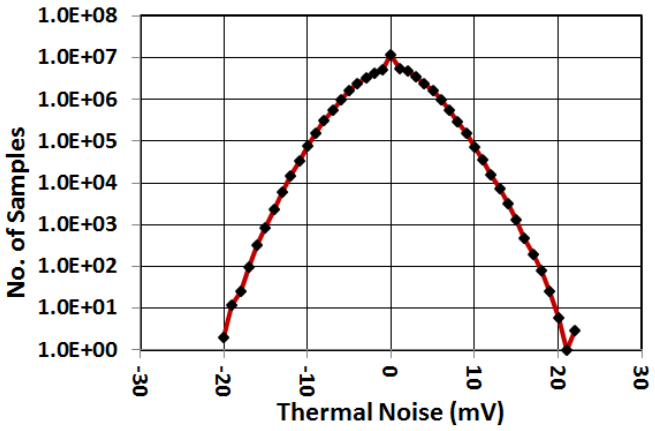


Figure 2.3. Distribution of Thermal Noise (Simulation in 32nm CMOS @ 30C)

Shrinking device sizes in nanometer CMOS has resulted in decrease of diffusion capacitance. Interconnect dimensions have also scaled resulting in reduced load capacitance. SRAM bit cells are shrinking to provide better memory density and reduce leakage during stand-by. High performance array architectures thrive to reduce word line and bit line capacitances to improve access time. Figure 2.4 shows the variation of standard deviation in thermal noise with device sizes. Smaller devices and shorter

interconnects increase the impact of thermal noise. Modern multi-core microprocessor architectures contain billions of transistors. Activity based local heating limits the operating frequency of designs. Thermal related reliability issues are of major concern in 3D chip integration. Since thermal noise is an artifact of thermal agitation of atoms in conductors, the standard deviation in thermal noise increases with increasing operating temperature, Figure 2.4.

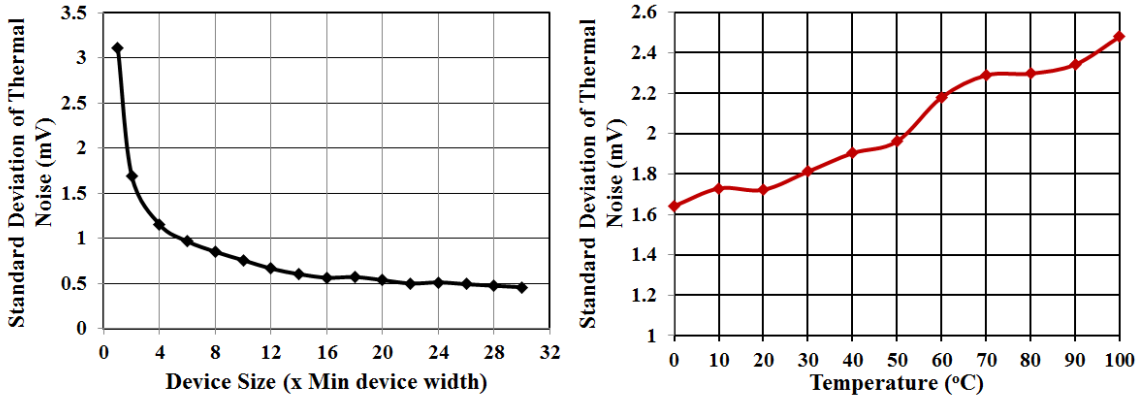


Figure 2.4. Variation of thermal noise with device size and temperature (Simulation in 32nm CMOS @ 30C)

2.2.2 Random Telegraph Noise

Random Telegraph Noise (RTN) causes threshold voltage fluctuation due to frequent trapping and detrapping of charge carries at the traps inside oxide layer [110, 15], Figure 2.5. RTN is characterized by two important factors; the capture (trapped) and emit (detrapped) time constants and the magnitude of threshold voltage fluctuation. The threshold voltage fluctuation cam also be modeled as variation in the transistor drain current. The capture and emit times have an exponential distribution with time constants of the order of few milliseconds.

The magnitude of RTN when modeled in the form of drain current variation has a log-normal distribution with a mean of $\Delta Id/Id = 5\%$. Figure 2.6 shows the charac-

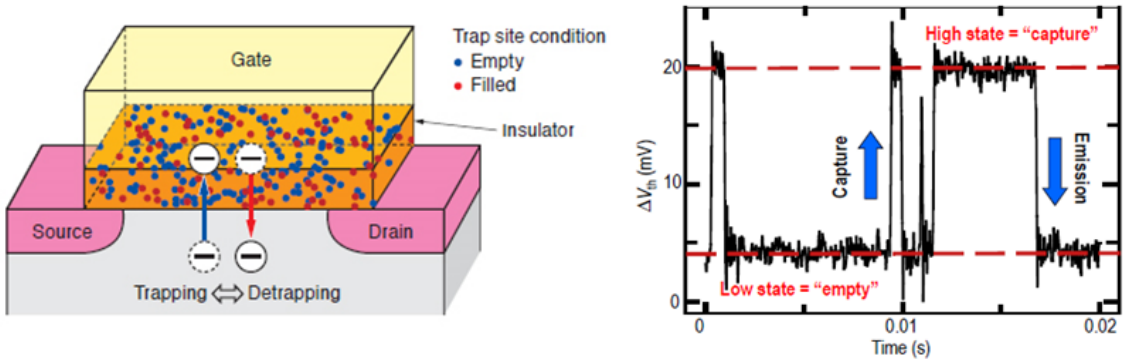


Figure 2.5. Random Telegraph Noise

teristics of RTN in 32nm bulk CMOS technology. The PSD of RTN has a Lorentzian spectrum with a critical $fc \sim 1kHz$ and a $1/f^2$ slope for all $f > fc$, Figure 2.7. The impact of RTN on memory elements like flash [73] and SRAM cells [127] are predominant in sub-45nm technologies. In [26], Fan, *et.al* present an analysis of the impact of V_{th} variation in FinFET device based SRAM and logic cells. Statistical and circuit techniques have been proposed to characterize RTN at the device level [53] and predict its impact on stability of SRAM cells [108]. At 3σ variations, the magnitude of RTN can be comparable or even greater to V_{th} variations due to Random Dopant Fluctuation (RDF) in planar devices and Work Function Variation in FinFETs. Therefore, a combination of process induced threshold voltage variation along with RTN can be a significant source of reliability concerns in nanometer CMOS circuits. A single MOSFET may have multiple trap locations and each associated with a different magnitude of $\Delta Id/Id$. However, at technologies 32nm and below, the probability of a single device having multiple traps is negligibly small due to the small channel lengths and decreasing device widths.Therefore, all analyses and experiments in this work are limited to single trap RTN noise conditions.

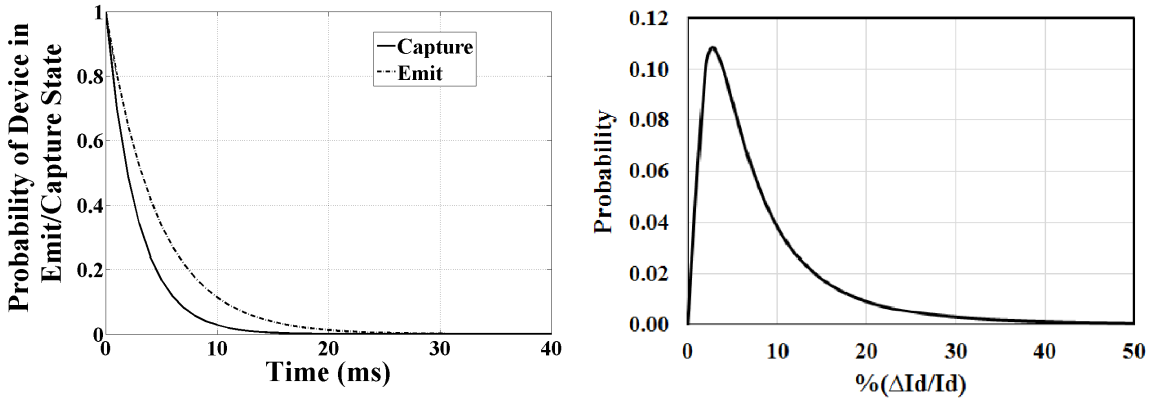


Figure 2.6. Characteristics of RTN

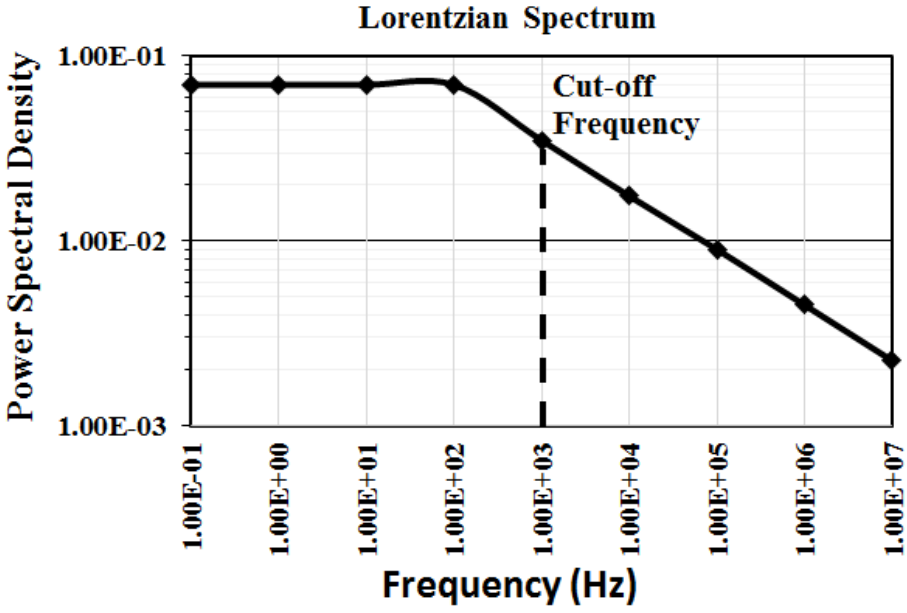


Figure 2.7. Lorentzian PSD of RTN

CHAPTER 3

CIRCUIT TECHNIQUES TO MANAGE VARIATIONS

Circuit techniques to manage process variation should either provide complete tolerance to variations or employ post-silicon techniques to compensate for impact of variations. In this chapter we present circuit techniques for both scenarios. In the first two sections we present a novel variation tolerant True Random Number Generator (TRNG) circuit using metastability resolution time[1]. The circuit is tolerant to variations in fabrication process as well as operating conditions and samples thermal noise or Random Telegraph Noise (RTN) to generate random bit stream. In the third section of this chapter, we present variation aware design of Post-Si Tunable (PST) clock buffer. The variation aware technique used statistical timing analysis to optimize transistor sizing in PST buffer for each clock group depending on the distribution of data path delay. Post fabrication, the PST buffer can be tuned to re-distribute slack among timing paths to achieve optimum chip performance [99].

3.1 Variation Tolerant True Random Number Generator using Thermal Noise

True Random Number Generators (TRNG) are important cryptographic primitives that are used for security protocols, data encryption and software simulations. TRNGs are used to generate nonces, one-time keys and session ids; additional bit padding for block ciphers [1, 64]. High performance TRNGs are also used in se-

[1]This work is covered under US patent application no. US 14/139,020

cure Ultra Wide Band (UWB) communication [48] and random instruction shuffling [6]. Non-cryptographic applications of TRNG include random software simulations, gaming and compressed sensing [23]. Fundamentally, a TRNG circuit samples and digitizes on-chip random physical phenomena. This source of randomness can be thermal noise, telegraph noise, shot noise and power supply noise. Analog TRNG circuits amplify thermal noise or flicker noise and digitize the signal using Analog-to-Digital Converters (ADC) [14, 25]. One of the most popular TRNG circuit technique is to sample thermal noise and power supply noise using Ring Oscillators (RO). This is done by combining the outputs of multiple ROs with relatively prime number of stages [95, 8, 56] or using a single self-time RO [18]. Chaos based TRNG circuits are deterministic, but non-linear systems with large periodicity [19, 24]. Metastable circuits use power up state of SRAM [33] or resolution state of cross-coupled inverters to sample thermal noise and generate random bit stream [29, 66, 98, 111].

Analog TRNG circuits are not energy efficient and do not scale with voltage and technology. RO based TRNG circuits require multiple ring oscillators leading to increased area and power consumption. Further, ring oscillators sample jitter which is significantly affected by global power supply noise. This renders the TRNG circuit vulnerable to invasive attacks. Frequency injection [62] and electromagnetic emanation [5] are shown to be effective in locking frequencies of oscillators, thereby compromising randomness of TRNG output. Metastable circuits are lightweight and energy efficient. However, they are extremely sensitive to device mismatch due to with-in variations. Adaptive compensation techniques use charge dump [29, 111] or digital circuit calibration [66]. These adaptive techniques require additional control logic to monitor the output of TRNG and perform circuit compensation. This adds to area and energy overhead. Adaptive calibration techniques also increase the risk of introducing correlation in the TRNG bit stream. Algorithmic post-processing

techniques may also be used instead of circuit tuning, but at the cost of significantly larger overhead.

3.1.1 Impact of Thermal Noise on Metastability Resolution Time

A conventional metastable cell is shown Figure 3.1. During the negative level of clock, nodes A and B are pre-charged to VDD. At the positive edge of clock, pre-charge is released and the cross-coupled inverters are allowed to resolve to a stable state. In the absence of any device mismatch, the resolution state depends on the thermal noise in the circuit. However, mismatch in parameters of the matched transistors biases the TRNG to generate more zeros or ones.

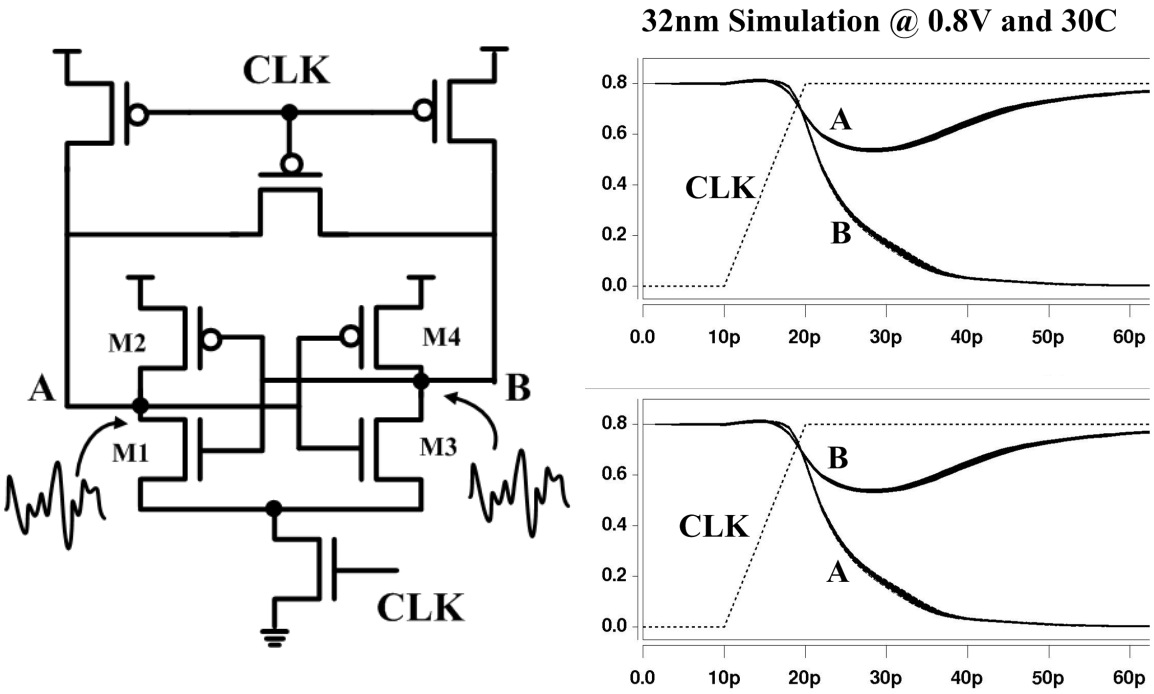


Figure 3.1. Metastable cell

Apart from resolution state, the time which cross-coupled inverters take to resolve to a stable state is also affected by thermal noise. In Figure 5.8, even though the output state of node A is always one, the time it takes to resolve to that stable state varies each cycle depending on thermal noise. Hence, resolution time is an alternate

metric to sample thermal noise. Figure 3.3 shows the distribution of resolution time. Assuming $\sigma_{Vt} = 10\%V_t$, a metastable cell with perfectly matched devices has large variance in resolution time. The resolution time of a biased metastable cell with 3σ mismatch in V_t and whose output is always a one also varies with thermal noise. However, the variance in resolution time decreases to as low as 0.8ps with increase in bias. Hence a variation tolerant TRNG can be designed by amplifying and digitizing the resolution time of a metastable cell.

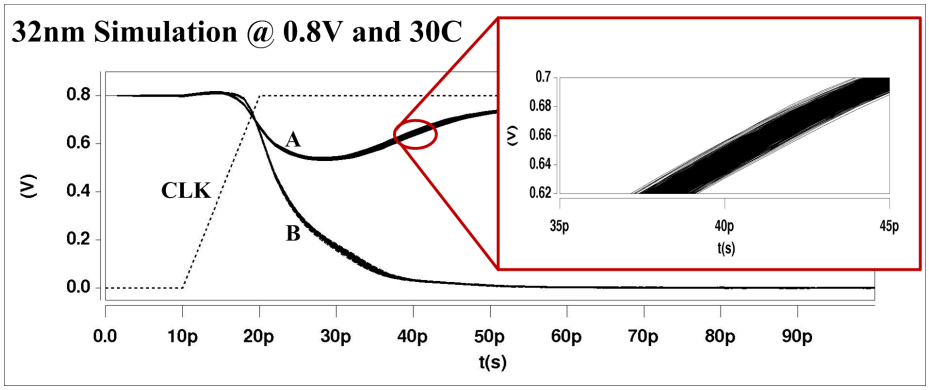


Figure 3.2. Variation of resolution time due to thermal noise

3.1.2 Metastability Resolution Time based TRNG

The proposed Metastability Resolution Time based TRNG consists of three major design blocks as shown in Figure 3.4. The MetaCell consists of pre-charge based cross-coupled inverter to sample thermal noise. Since the standard deviation of thermal noise can be as low as 0.8ps due to process variation, a Time Difference Amplifier (TDA) is used to amplify the resolution time variation. A large enough amplification is achieved to practically capture the resolution time variation. Digitization circuit using Time to Digital Converter (TDC) and Parity circuit measures the resolution time and digitizes to generate a single bit output.

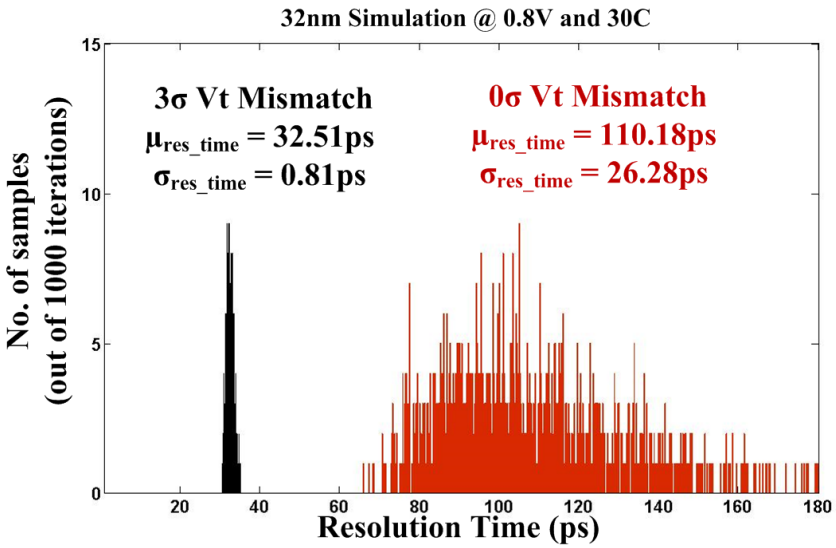


Figure 3.3. Distribution of Resolution Time

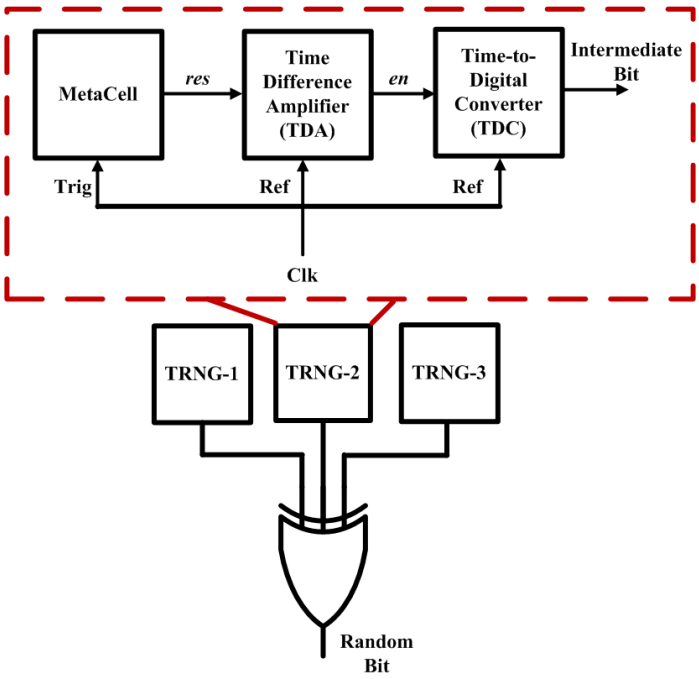


Figure 3.4. Metastability Resolution Time based TRNG

Metacell: Conventional MetaCells used in TRNG circuits have matched devices with equal channel widths and symmetric layout implementation. As seen in Figure 3.3, the mean resolution time of a MetaCell can vary significantly depending on the bias. A TDA which provides constant amplification gain for all values of resolution time will require on-chip capacitive tuning. Furthermore, such a circuit is not reliable across all operating temperatures and voltage. We present an asymmetric MetaCell where one of the inverters has larger transistor widths than the other. Figure 3.5 shows the standard deviation in resolution time due to V_t variation in the inverters. If both the inverters are designed with matched widths, the variation in resolution time can have a standard deviation of 20ps. For an asymmetric MetaCell with 1X-8X inverter sizing, variation in V_t has negligible impact and the standard deviation in resolution time is 2.4ps. This facilitates a simpler TDA design for a single Δt. The 1X-8X asymmetric MetaCell is shown in Figure 3.6. The asymmetry in MetaCell also eliminates the need for symmetric layout design. Any mismatch in interconnect loads and contact resistance due to lithography variations have negligible impact on the resolution time.

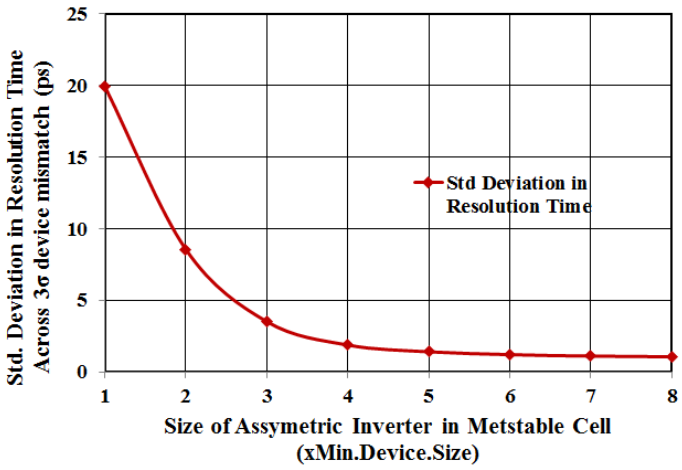


Figure 3.5. Variation of standard deviation in mean resolution time with asymmetric inverter size

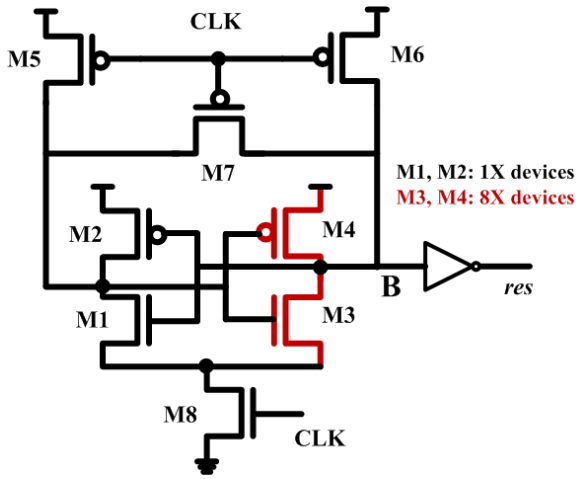


Figure 3.6. Asymmetric MetaCell with 1X-8X cross-coupled inverter

Time Difference Amplifier (TDA): The Time Difference Amplifier is as shown in Figure 3.7. It consists of a pair of cross-coupled NAND gates. One of the inputs to the TDA is a delayed version of the clock signal. This is the reference signal to the TDA. The other input is the signal from MetaCell indicating the resolution of the MetaCell. The TDA amplifies the time difference between *clk* and *res*. The delay chain on the clock signal is optimized for expected resolution time. Since the asymmetric structure of MetaCell ensures negligible deviation in mean resolution time, a delay chain designed for a single Δt is sufficient even in the presence of process variation. The amplification gain depends on the capacitive load at the output of NAND gates. We design the capacitors by shorting the drain and source of large NMOS and PMOS devices. Figure 3.8 shows a sample variation of TDA gain with Δt. If the resolution time of TRNG has a mean value of 30ps and standard deviation of 0.8ps, The output of TDA will amplify the variation to have a standard deviation of 24ps; a gain of 30X. However, if mean Δt between clock and resolution time varies due to process variation, the gain of TDA decreases. The asymmetric design ensures a standard deviation in mean resolution time of 2.4ps. As a result, a TDA design

that is optimized for a resolution time of 30ps will still provide greater than 20X amplification gain.

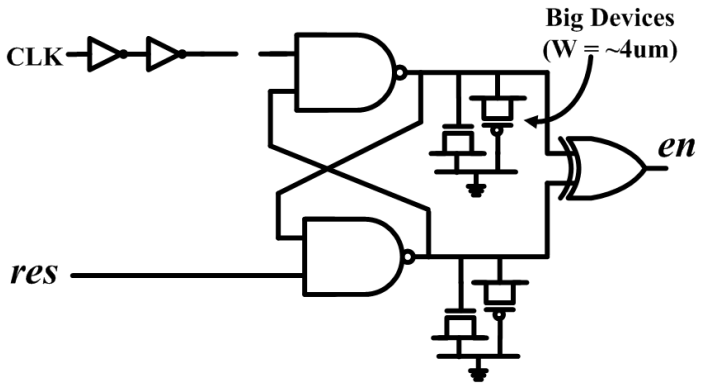


Figure 3.7. Time Difference Amplifier (TDA)

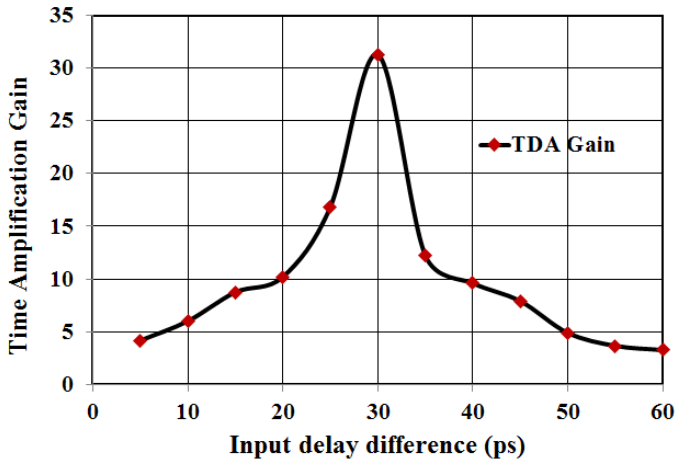


Figure 3.8. Variation of TDA gain with Δt

Time-to-Digital Converter (TDC): The Time-to-Digital Converter is as shown in Figure 5.11. It consists of 7-stage Ring Oscillator (RO) that is enabled by the rising edge of *clk* signal. The *en* signal from the TDA is delayed by a 6-stage delay chain to generate *den[6:1]*. These delayed enable signal sample the output of rind oscillator to generate bits *b1* to *b6*. All functional blocks of the TRNG work in reference to the

clock. As a result, once the clock enables the RO, the *en* and hence the delayed enable signals depend on the resolution time of the MetaCell.This affects the sampled bits *b1* to *b6* in each iteration. All the six sampled bits may not change every cycle. Hence, they are combined using XOR function to calculate the odd parity and generate one single bit out of the TRNG. The overall behavior of the TRNG is shown in Figure 3.10.

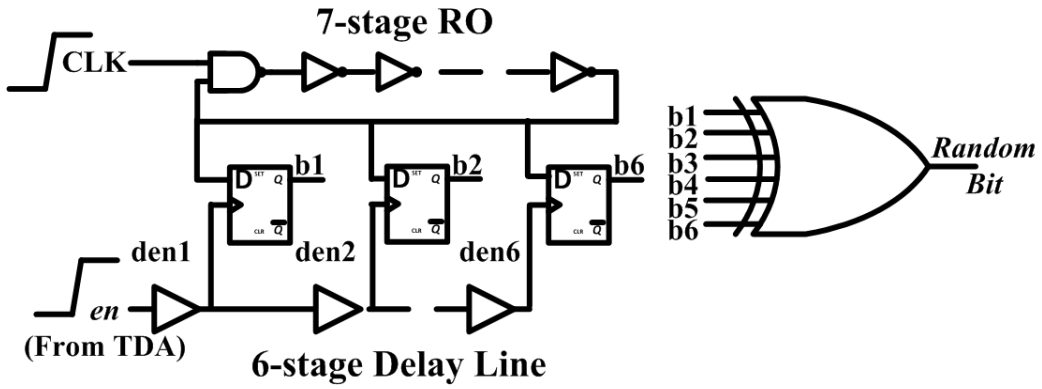


Figure 3.9. Time-to-Digital Converter (TDC)

3.1.3 Implementation and Results

The proposed TRNG circuit was implemented in 32nm IBM SOI process, Figure 3.11. The prototype chip, of size 2mmx2mm, consisted of 512 instances of metastability resolution time based TRNG, Figure 3.12. Internal ring oscillators were used as clock source and a 8k bit shift register was used to store the output of the TRNG. A scan chain based configuration setup was used to choose the TRNG instances and configure the clock frequencies.

The distribution of amplified resolution time across 3σ variation in V_t is shown in Figure 3.13. Process variation has negligible impact on amplified resolution time and hence the TRNG is variation tolerant. The output of TRNG was validated for quality using a subset of statistical tests provided by American National Institute

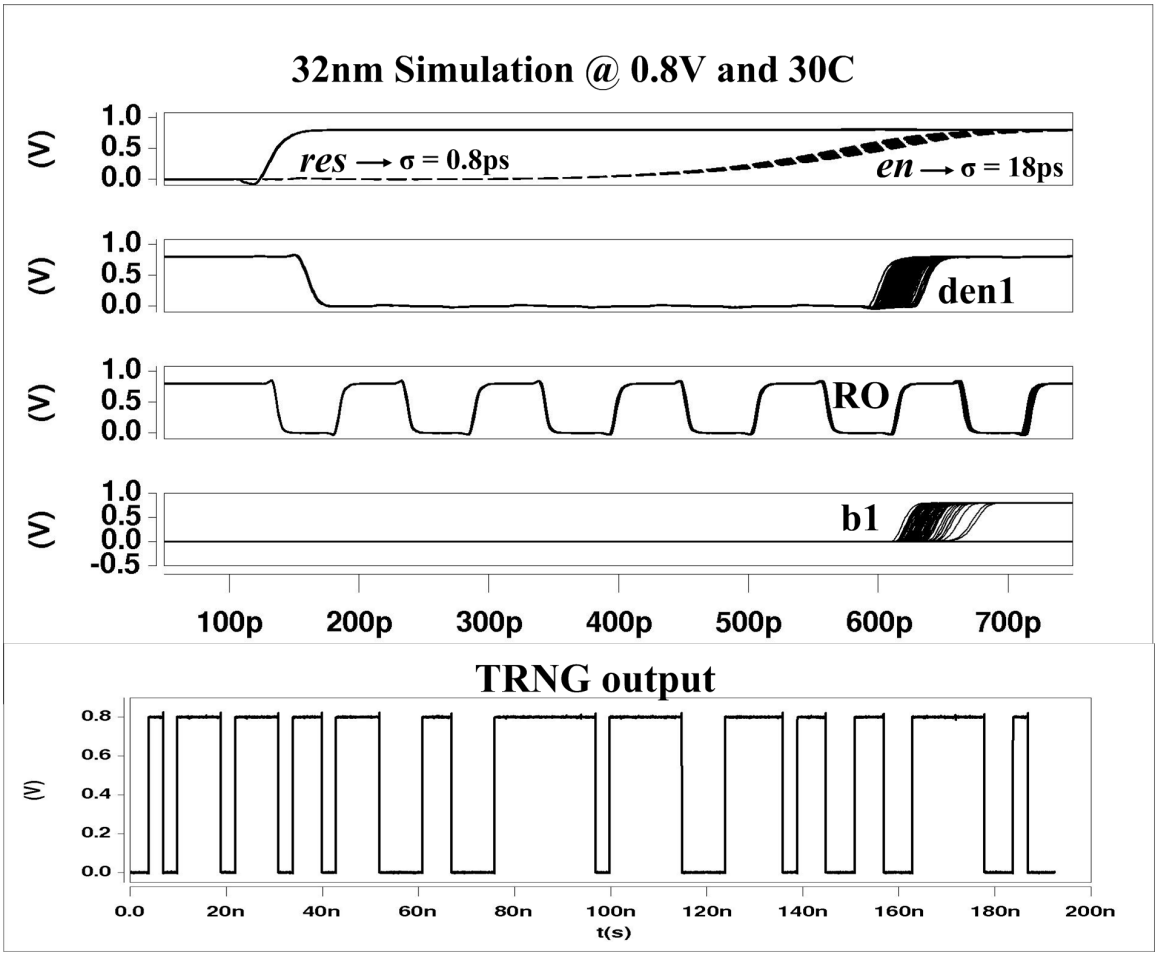


Figure 3.10. Operation of Metastability Resolution Time based TRNG

of Standards and Test (NIST) [80]. The results of NIST tests are shown in Table 4.6. Figure 3.14 shows the autocorrelation of TRNG output. Since the TRNG does not involve adaptive circuit tuning, the autocorrelation is negligible. The NIST test results and autocorrelation plot shows that the proposed TRNG circuit generates high quality random bits even in the presence of process variation with out requiring additional post-processing.

The TRNG circuit was analyzed under varying voltage and temperature conditions. The bit entropy is maintained around unity for supply voltage of 500mV to 1V, Figure 3.15. At nominal operating voltage of 0.9V, the total circuit power is

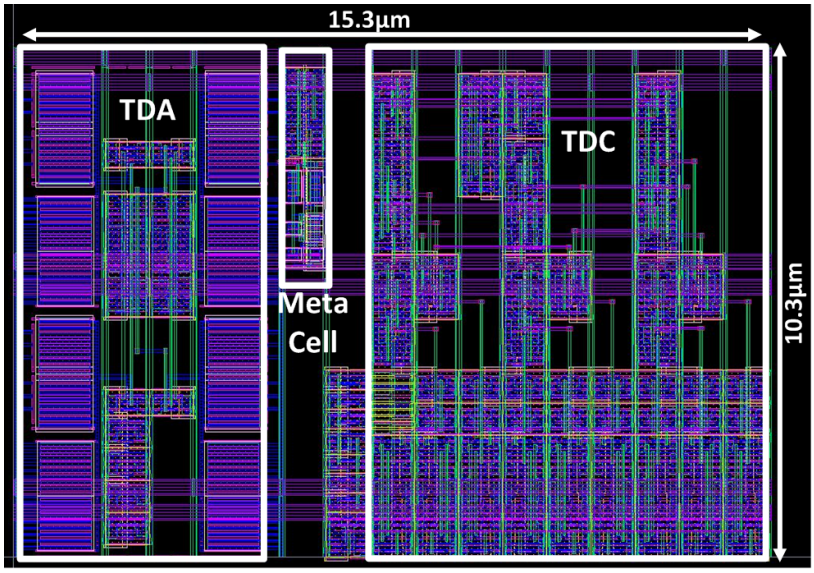


Figure 3.11. Layout of Metastability Resolution Time based TRNG

Table 3.1. Result of NIST Statistical Tests

Test	**p-value**	**Result**
Frequency	0.8153	PASS
Block Frequency	0.2395	PASS
Cumulative Sum	0.1845	PASS
Runs	0.3221	PASS
Longest Ones	0.1976	PASS
Rank	0.4695	PASS
FFT	0.7679	PASS
Non Overlapping Template	0.4995	PASS
Overlapping Template	0.2641	PASS
Approximate Entropy	0.7984	PASS
Universal	0.6729	PASS
Serial	0.3126	PASS
Matrix Rank	0.0825	PASS
Linear Complexity	0.4185	PASS

0.66mW, Figure 3.16. The leakage power constitutes 1.8% of the total power at 12W. The nominal bit rate achieved is 1Gbps at a high energy efficiency of 0.66pJ/bit. At Near Threshold Voltage (NTV) operation of 500mV, the circuit operates with a bit rate of 12Mbps and 0.48pJ/bit.

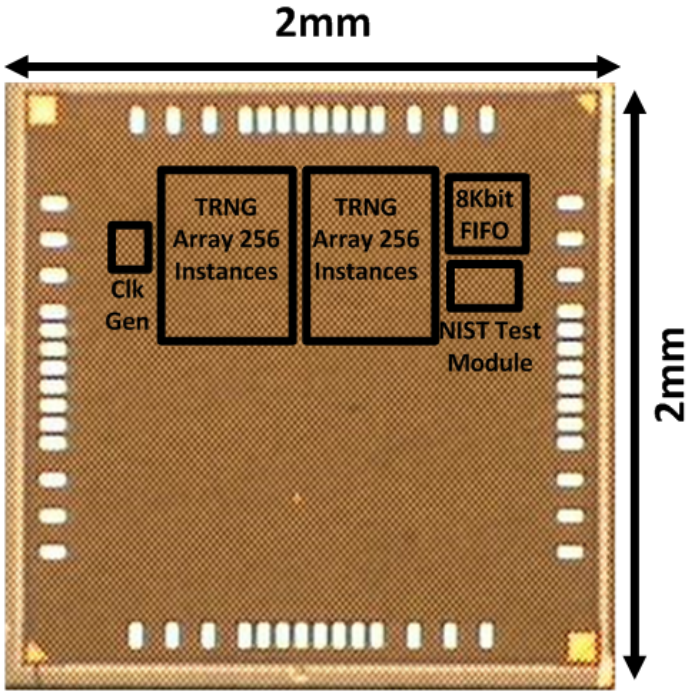


Figure 3.12. Chip Micrograph - 32nm IBM SOI

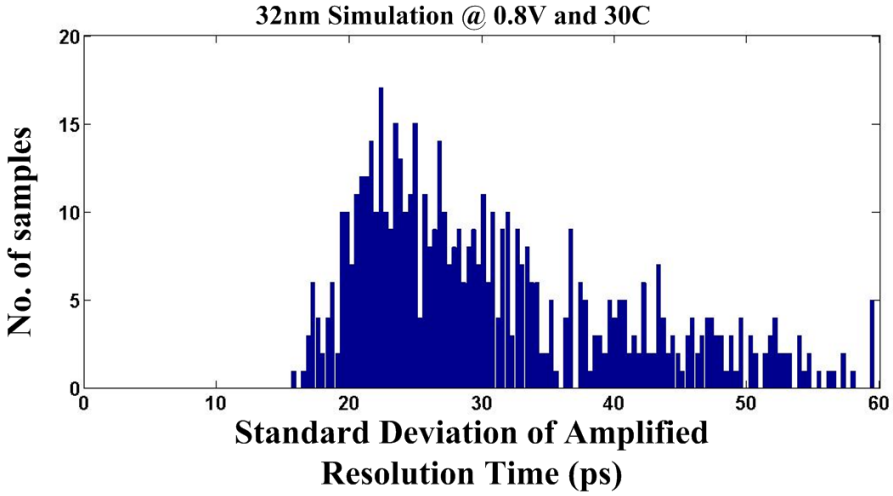


Figure 3.13. Distribution of Amplified Resolution Time

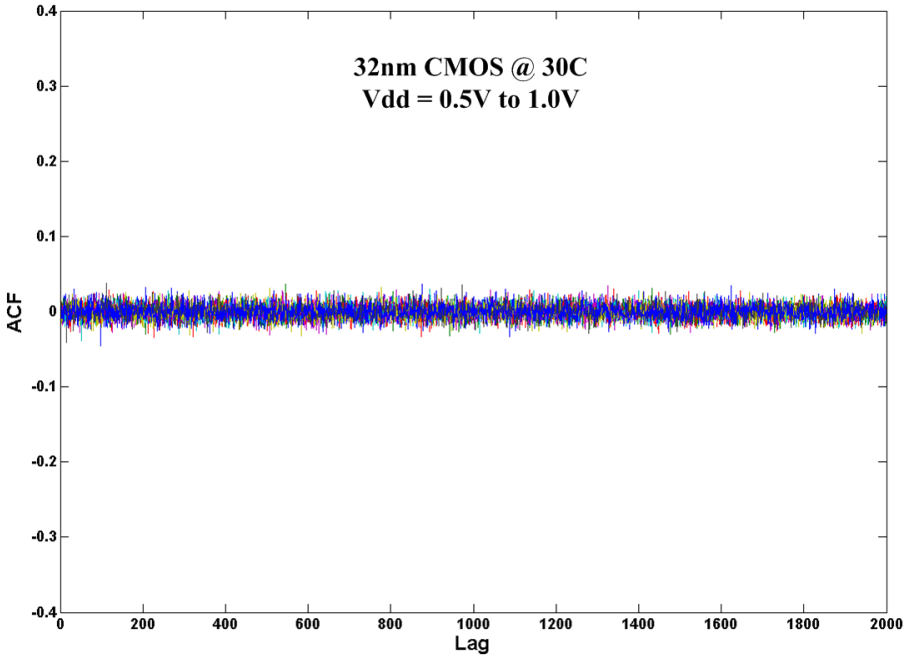


Figure 3.14. Autocorrelation plot of TRNG output

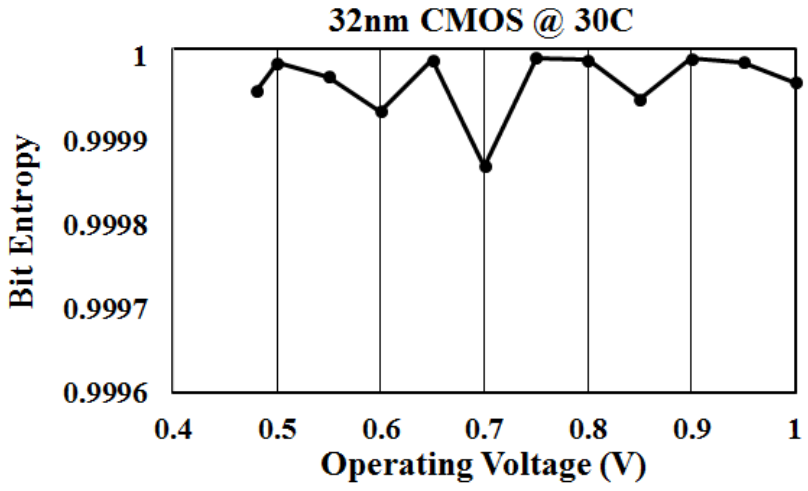


Figure 3.15. Bit Entropy across different supply voltages

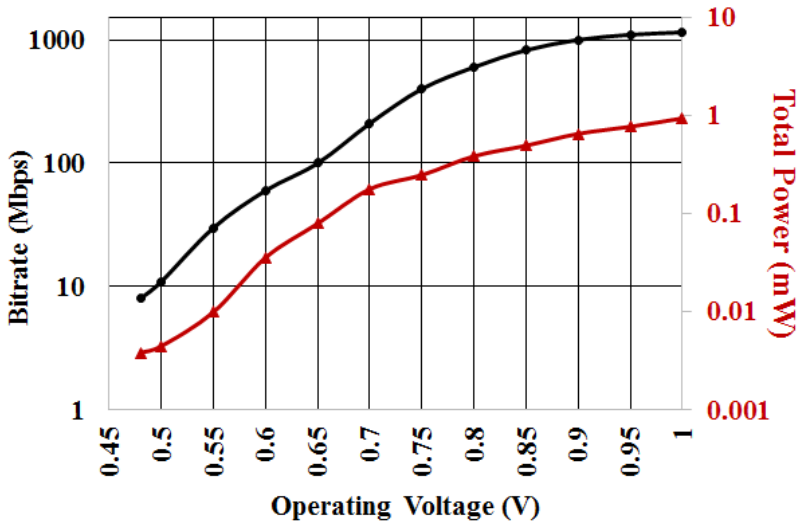


Figure 3.16. Power numbers at varying supply voltage

3.2 Variation Tolerant True Random Number Generator using Random Telegraph Noise

Random Telegraph Noise (RTN) is a prominent source of noise in nanometer CMOS. RTN causes serious reliability threat in analog circuits, flash memory and Static Random Access Memory (SRAM) in sub-32nm technologies. While RTN has a negative impact on sensitive circuits, it can also be used as a source of randomness to design variation tolerant circuit for random number generation. Existing TRNG circuits using RTN are either based on analog circuits [12] or memory cells [35]. These circuits have very low bit rate of the order of few kbps and are not energy efficiency.

3.2.1 Impact of RTN on Metastability Resolution Time

A metastable circuit consists of cross-coupled structure that can be pre-charged to Vdd and allowed to resolve to a stable state. In this work, we use a *MetaCell* consisting of a pair of cross coupled inverters with pre-charge PMOS devices, Figure 3.17. Once the nodes A and B are pre-charged and the pre-charge voltage released, the state to which the nodes resolve depend on the noise in the circuit. However, with-in die variations bias the circuit resulting in highter probability of bit 0 or bit 1. Apart from the resolution state, the time taken by the circuit to resolve to a stable state also depends on the magnitude of noise. In this case, RTN in transistors M1 and M3 define the resolution time of the MetaCell. Since the technique for sampling noise is the resolution time and not the resolved states, the circuit is tolerant to with-in die variations.

Figure 3.18 shows the actual MetaCell used in the TRNG. The nodes A and B are pre-charged and allowed to resolve. Since we do not depend on the resolved state, an XOR gate with A and B as inputs detects when the circuit has stabilized to a stable state. The actual resolution states is immaterial. If an RTN captured state (high V_{th})

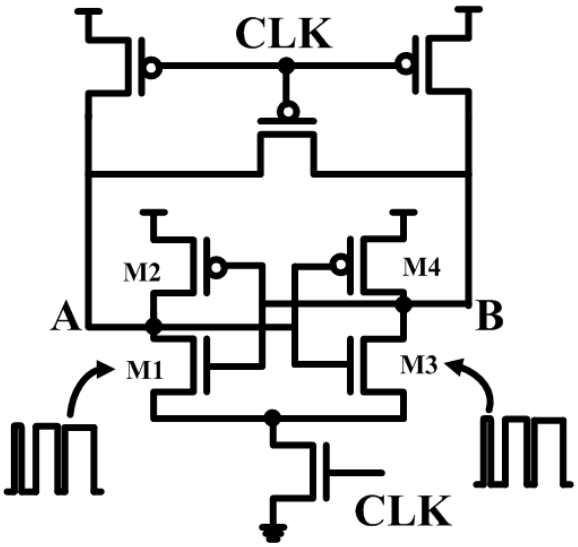


Figure 3.17. Impact of RTN on Metastable Circuit

is defined by 1 and empty/emit state (low V_{th}) is defined by 0, the circuit can be one of the four noise conditions during resolution:

1. M1=0 and M3=0
2. M1=0 and M3=1
3. M1=1 and M3=0
4. M1=1 and M3=1

Since variation in the resolution time is the source of randomness, it is desirable to have detectable difference between the resolution times when the circuit is in different noise conditions. This variation will depend on the magnitude of RTN noise , which is the fluctuation in $\Delta Id/Id$. Therefor, the XOR output drives a large capacitive load to provide necessary amplification for the resolution time and hence the variation in resolution time. A Time-to-Digital Converter (TDC) is used to digitize the amplified resolution time, Figure 3.19.

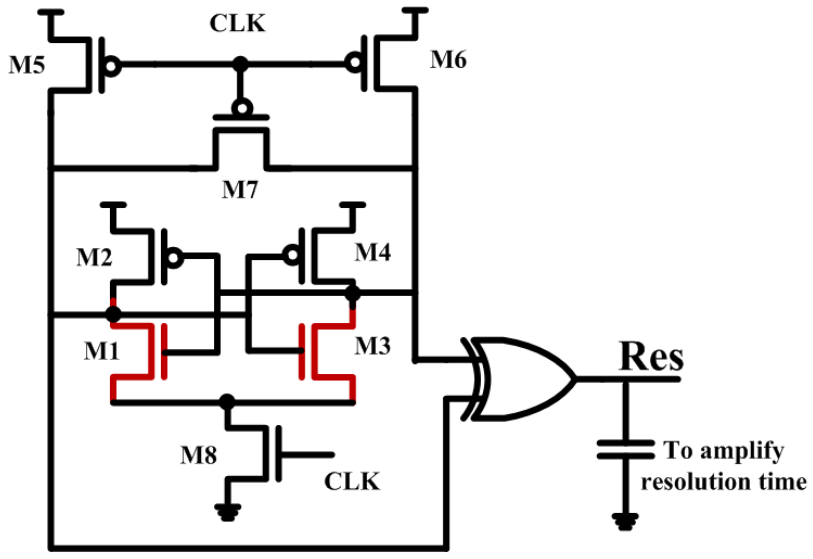


Figure 3.18. MetaCell with Resolution Time Amplification

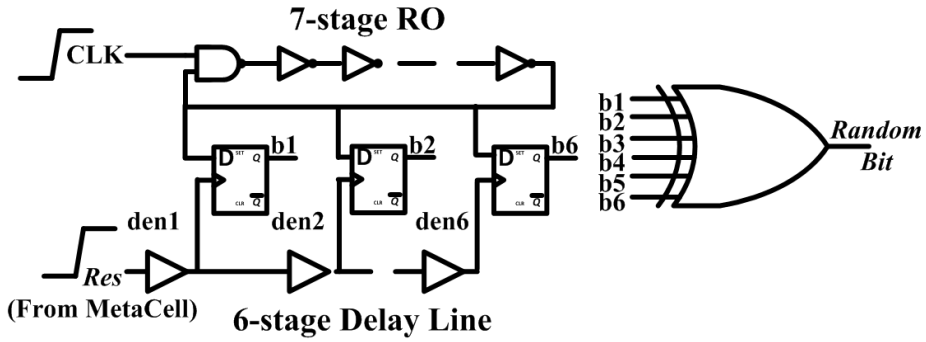


Figure 3.19. Time-to-Digital Converter (TDC)

3.2.2 Multi-Stage TRNG Design

One of the main issues of using RTN as a source of randomness is the low frequency spectrum of the noise. Currently reported TRNG circuits that use RTN achieve a maximum bit rate of up to 50kbps [12, 35]. This may not be suitable for all applications and scenarios where additional post-processing has to be performed on the generated bits. Figure 3.20 shows the bit stream of 10000 bits generated using a single MetaCell sampled at 100MHz. Since a single MetaCell has only four possible

noise states each with an average time constant of the order of milliseconds, the TRNG circuit can only be sampled at very low frequencies. This is further illustrated by the Power Spectral Density (PSD) of resolution time variation in a single MetaCell TRNG, shown in Figure 3.21.

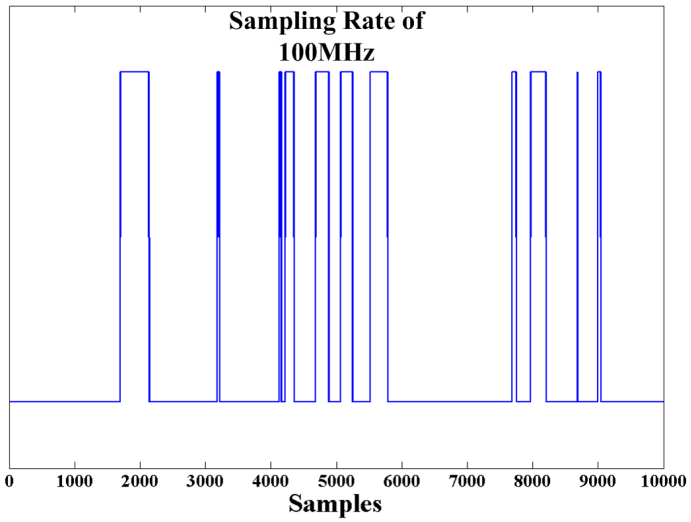


Figure 3.20. Correlated bit stream from single MetaCell

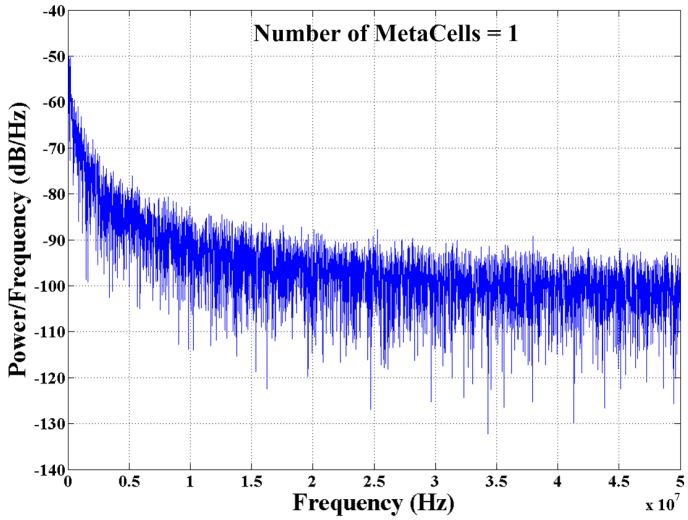


Figure 3.21. Power Spectral Density of Resolution Time with Single MetaCell

Therefore, in the proposed TRNG circuit, we use a chain of MetaCells with internal self-triggering and measure the sum of resolution times to generate random bits, Figure 3.22. Each MetaCell uses the output of XOR gate to trigger the next one once it has resolved to a stable state. The clock signal is used to trigger the first MetaCell and as reference to the TDC. For a n-stage TRNG (consisting of a chain of n MetaCells), there can be 2^{2n} number of possible noise states and corresponding net resolution times. Figure 3.23 shows the comparison of PSD of net resolution time for different number of MetaCell stages for sampling rate up to 50MHz. As the number of MetaCell stages increase, the PSD approaches a more uniform distribution across all sampling frequencies.

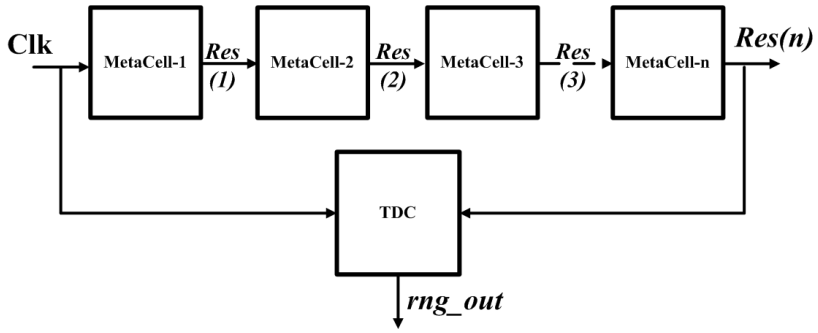


Figure 3.22. Design of multi-stage TRNG with MetaCell chain

3.2.3 Implementation and Results

A 256-stage TRNG was implemented in 32nm Predictive Technology Model and simulated using NGSPICE. All MetaCells were modeled with threshold voltage variation with $\sigma V_{th} = 0.1 * V_{th}$. Transient noise simulations were used to simulate the effect of RTN. Figure 3.24 shows the PSD of resolution time for a 256-stage TRNG. It can be seen that increasing the number of stages increases the uniformity of the PSD across all sampling frequencies. Figure 3.25 shows the distribution of the sum of resolution times across the 256 stages. The 256 stage design provides potentially 2^{512}

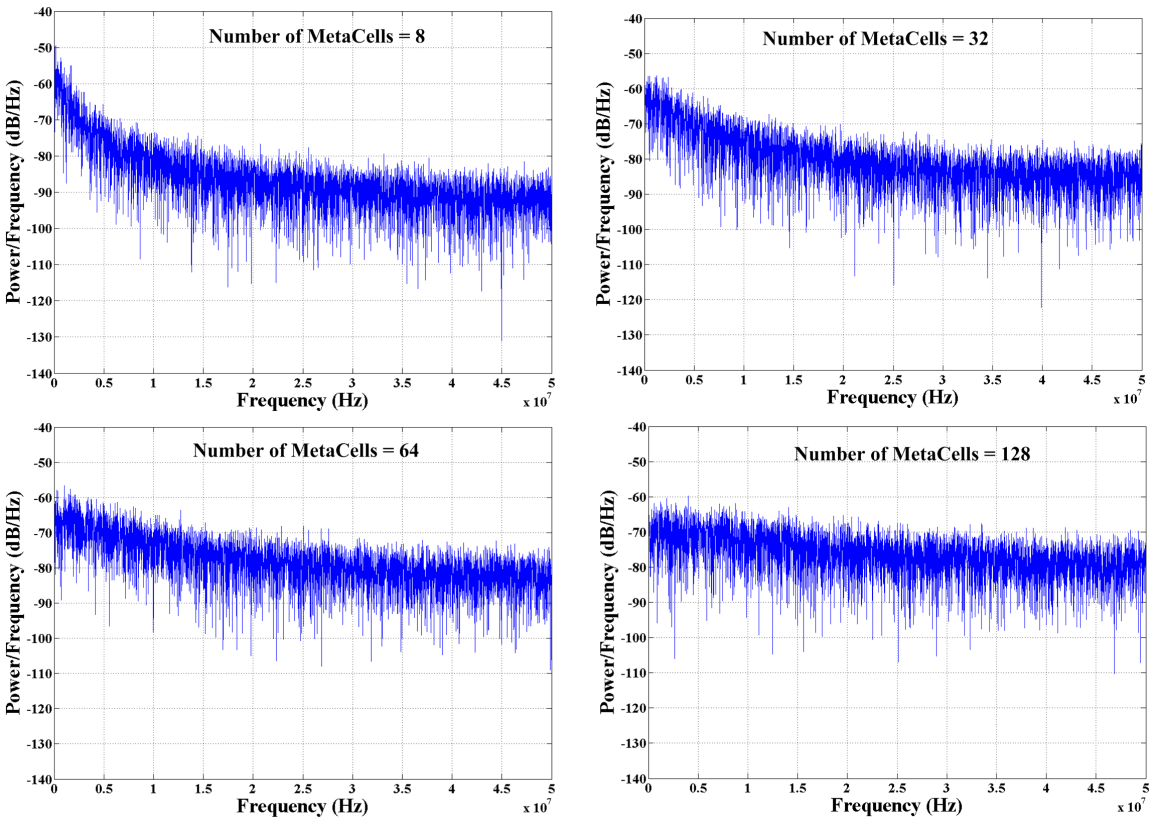


Figure 3.23. Comparison of PSD of Resolution Time for Different Number of Meta-Cell Stagesl

unique noise conditions, leading to an overall standard deviation in resolution time of 85ps. This is a significantly large variation compared to using a single or a few 10s of MetaCells.

The output of the TRNG was validated using a subset NIST randomness test suite. Figure 3.26 and Figure 3.27 show the auto-correlation plot and run lengths of 1 for a 256 stage TRNG and 100k bits. Since the TRNG does not employ any adaptive feedback, correlation is negligible up to a sampling rate of 60MHz. The total power of the 256-stage TRNG is $\sim 16\mu W$ with a leakage component of $\sim 3\mu W$, Figure 3.28. Figure 3.29 shows a comparison of energy/bit for different TRNG configurations. Fewer TRNG stages constitute smaller dynamic and leakage power. However, due to

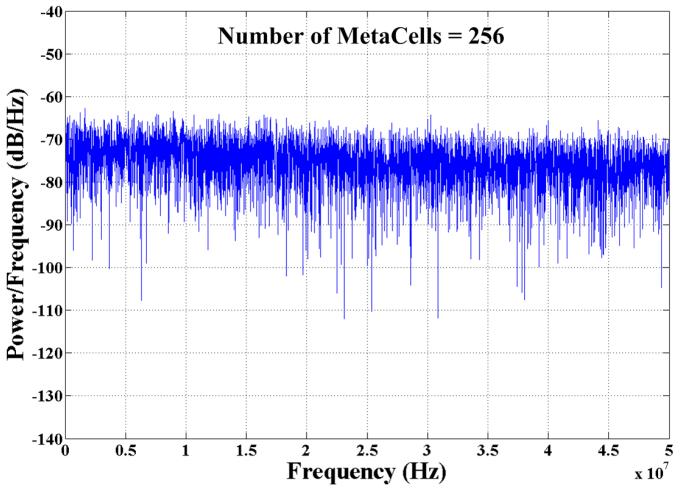


Figure 3.24. Power Spectral Density of Resolution Time with 256 Stage TRNG

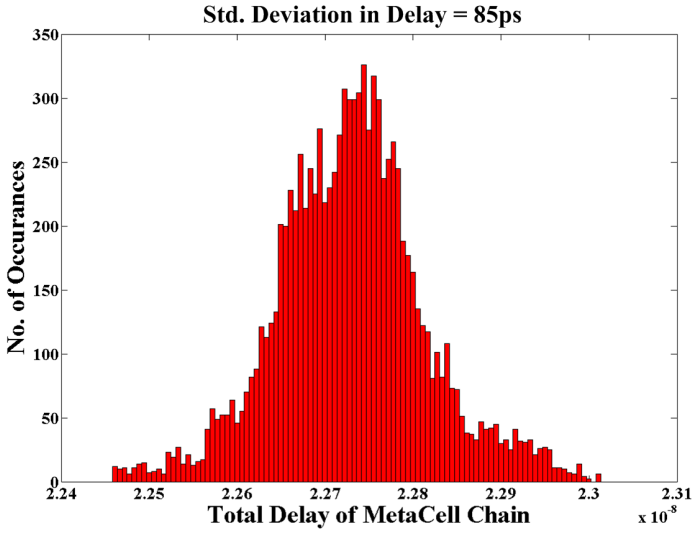


Figure 3.25. Distribution of Resolution Time with 256 Stage TRNG

a higher bit rate, 256-stage TRNG provides the most energy efficient random number generation.

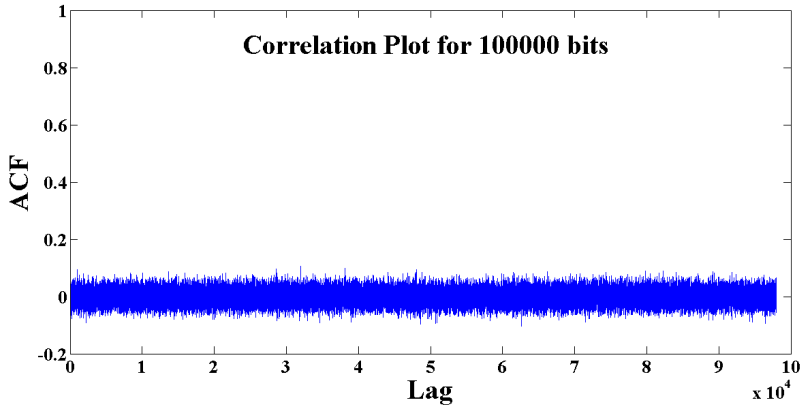


Figure 3.26. Auto-correlation of Output Bit Stream from 256-Stage TRNG

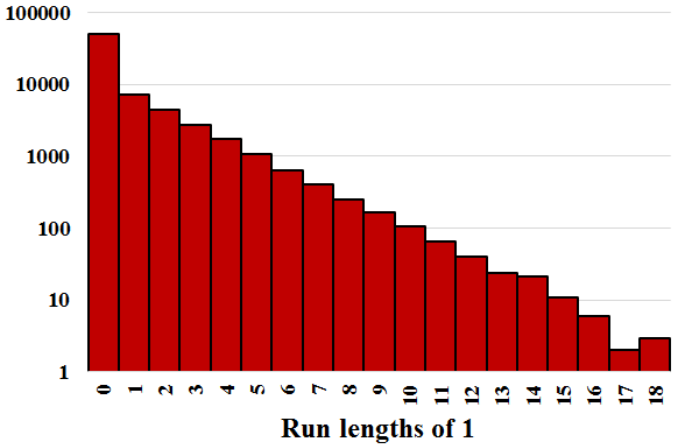


Figure 3.27. Run length of 1s for Output Bit Stream from 256-Stage TRNG

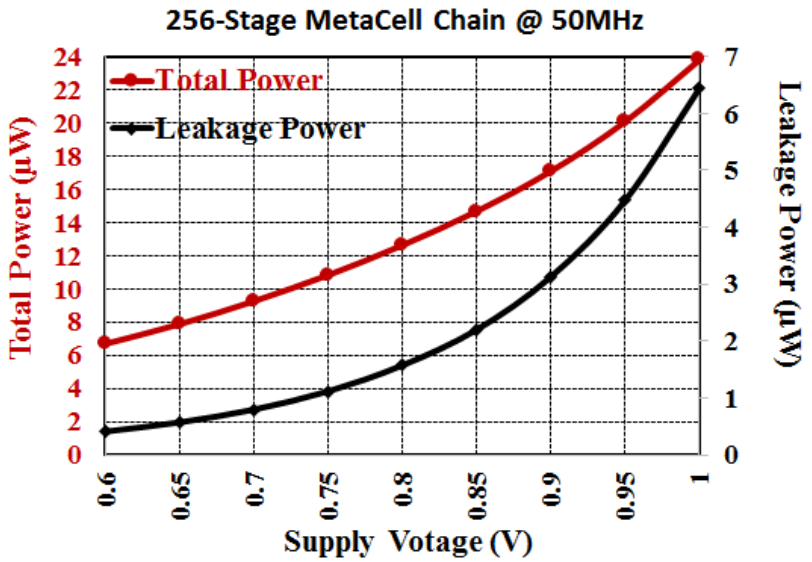


Figure 3.28. Total and Leakage Power of 256-Stage TRNG

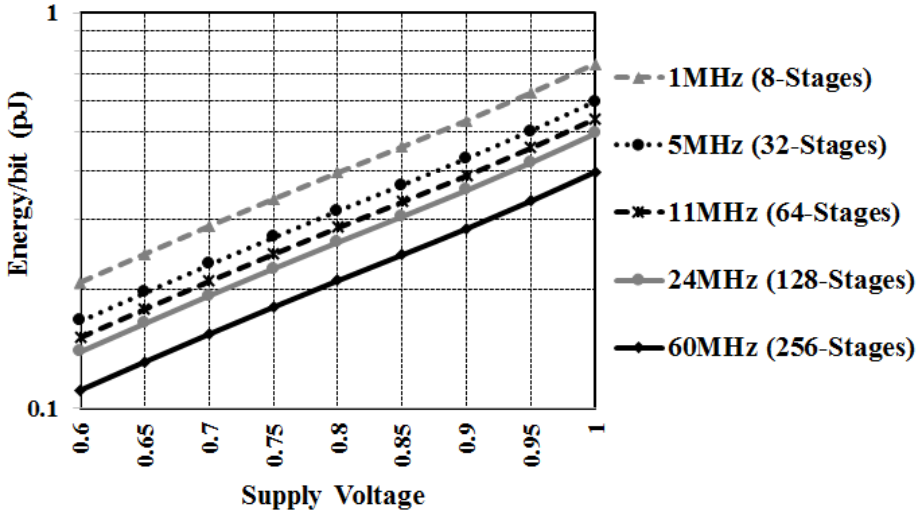
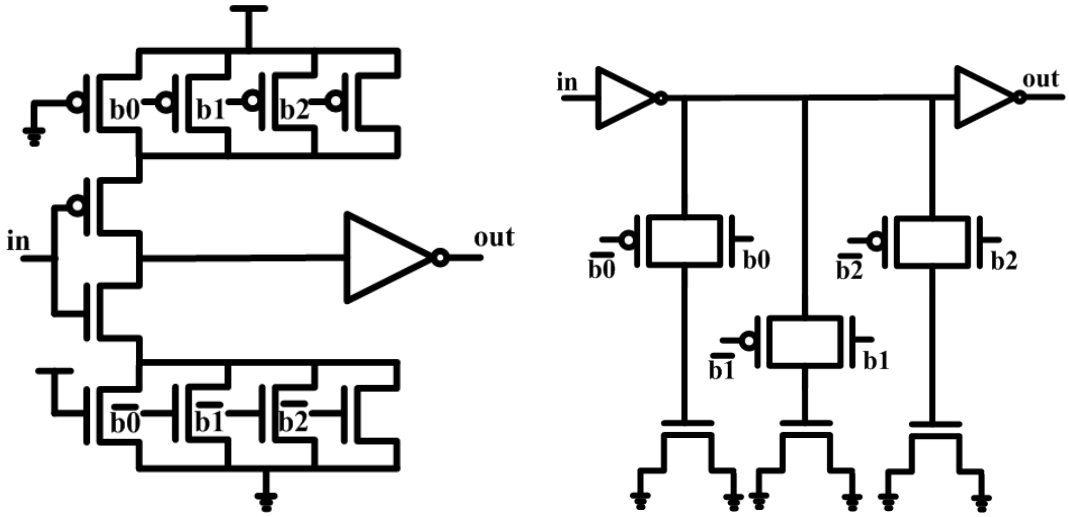


Figure 3.29. Comparison of Energy/bit for different TRNG Configurations

3.3 Variation Aware Design of Post-Si Tunable Clock Buffer

In the previous section, we presented a novel TRNG circuit that is tolerant to with-in die variations. Conventional digital data path logic are affected by die-to-die and wafer-to-wafer variations. The delay of standard cells vary by up to 5% for a σ/μ of 30% in V_t in 25nm technology [60]. Circuit designers already provide enough guard band during the design phase to compensate for operation in multiple voltage and frequency modes; clock jitter and device aging. Adding additional guard band to account for process variation will lead to significant increase in area and power overhead. Furthermore, chips which are fabricated under fast process corner tend to be over designed; increasing leakage power.

A commonly used technique to reduce guard band against process variation is using Post-Silicon Tunable clock buffers (PST) [106]. A PST buffer is a variable delay element used in clock paths to vary clock latency latency [49, 67, 76]. Conventional PST buffers can be categorized as current starved inverter based PSt and shunt capacitor based PST, shown in 3.30. After fabrication, PST buffers are configured to vary clock path latencies and redistribute clock skew. This helps in sharing slacks between different timing paths to achieve the best possible performance for a chip at a given process corner. It is a common practice to bin fabricated chips based on maximum operating frequency. Therefore, optimal configuration of PST buffers increases performance binning yield. A number of algorithms have been proposed in literature to optimize insertion of PST buffers [28, 43, 104, 113]. The number and levels of PST buffer insertion is based on statistical timing analysis to identify process critical timing paths. Optimum insertion of PST buffers ensures maximize binning yield with minimize area/power overhead. In [43] and [129], a gate sizing technique is proposed for data path along with clock buffer insertion to reduce over compensation in shorter timing paths. Appropriate test methodology is also critical to trace and sensitize critical paths to determine the optimal configuration for maximum operating



(a) Current Starved Inverter PST Buffer (b) Shunt Capacitor PST Buffer

Figure 3.30. PST Buffers

frequency [78]. PST buffers are also used to counter variations beyond fabrication process. They can be used to counter aging related performance degradation [55] and for on-chip thermal management [16].

Apart from identifying clusters of critical timing paths and optimum insertion of PST buffers, the design of PST buffer circuits and the tunable delay values it provides also affects parametric yield. Figure 3.31 shows the distribution of delays of a 16-inverter delay chain with $\Delta V_t \sim N(0, 10\%V_t)$ and 100,000 Monte-Carlo simulations in 32nm predictive technology. The delay variation has a Gaussian distribution with mean 73ps and standard deviation of 4ps. Assuming the delay chain is a part of data path, delay values less than the mean are synonymous to chips with timing paths that could fail hold time requirements due to process variation. Similarly, the delays larger than mean are synonymous to chips with degraded data path delay and hence reduces performance. The Gaussian distribution signifies that a large number of fabricated chips have performance degradation close to the expected mean. Having linear tunable delays over compensates in a large set of chips. It can lead to limited

chip performance (when fixing hold timing violations) or decreased yield (when fixing setup timing violations). The mean and standard deviation of delays also depend on the number of logic stages in the data path. In this section, we propose a variation aware design of PST buffer using non-linear tunable delay values to close match the distribution of expected delay/slack variation. We estimating potential variation in the delay of a timing path or group of timing paths driven by a PST buffer and correspondingly choose the delay values.

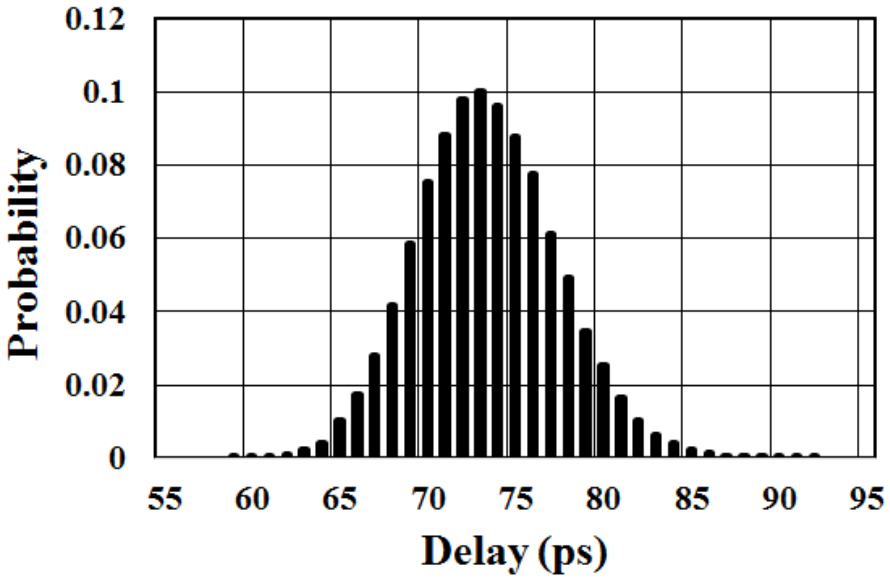


Figure 3.31. Delay distribution of 16-inverter delay chain

3.3.1 Variation Aware Non-linear Tunable Delay Estimation

Statistical Variation in Timing Slack : The two most critical timing requirements for a digital data path are *setup time (max path)* and *hold time (min path)*. Setup slack for a timing path is given by,

$$Slack_{setup} = (T_{period} - T_{setup} + T_{skew}) - T_{data} \tag{3.1}$$

where, T_{data} is the sum of clock to Q delay of the launch flop and delay of data path, T_{period} is the clock period, T_{setup} is the setup time requirement of the capture flop and T_{skew} is the difference in clock latency between the capture flop and launch flop. Assuming the clock period and setup time requirement to be constant, the two delay parameters varying due to process variation are T_{data} and T_{skew}. They both have a Gaussian distribution and mutually independent in a worst case scenario. If,

$$T_{data} = N\left(\mu_{data}, \sigma^2_{data}\right)\ T_{skew} = N\left(\mu_{skew}, \sigma^2_{skew}\right)$$

$$Slack_{setup} = N\left(\mu_{slack_{setup}}, \sigma^2_{slack_{setup}}\right) \tag{3.2}$$

$$where\ \mu_{slack_{setup}} = \mu_{data} + \mu_{skew}\ and\ \sigma^2_{slack_{setup}} = \sigma^2_{data} + \sigma^2_{skew}$$

Similarly slack for hold time check is given by,

$$Slack_{hold} = T_{data} - (T_{hold} + T_{skew}) \tag{3.3}$$

where, T_{hold} is the hold time requirement of the capturing flip flop. Variation in hold slack due to process variation is given by,

$$Slack_{hold} = N\left(\mu_{slack_{hold}}, \sigma^2_{slack_{hold}}\right) \tag{3.4}$$

$$where\ \mu_{slack_{hold}} = \mu_{data} - \mu_{skew}\ and\ \sigma^2_{slack_{hold}} = \sigma^2_{data} + \sigma^2_{skew}$$

Therefore, setup and hold time slacks have a Gaussian distribution that has to considered when designing PST buffers. The distribution of slack differs for each timing path and accordingly, the tunable delay values required to compensate for the variation.

Estimation of Non-linear Tunable Delay Values :

Let us consider the design snippet shown in Figure 3.32 and assume a maximum of 3σ variation in delays of all paths across all fabricated chips. The slack on timing paths P1 and P3 determine the performance of each chip.

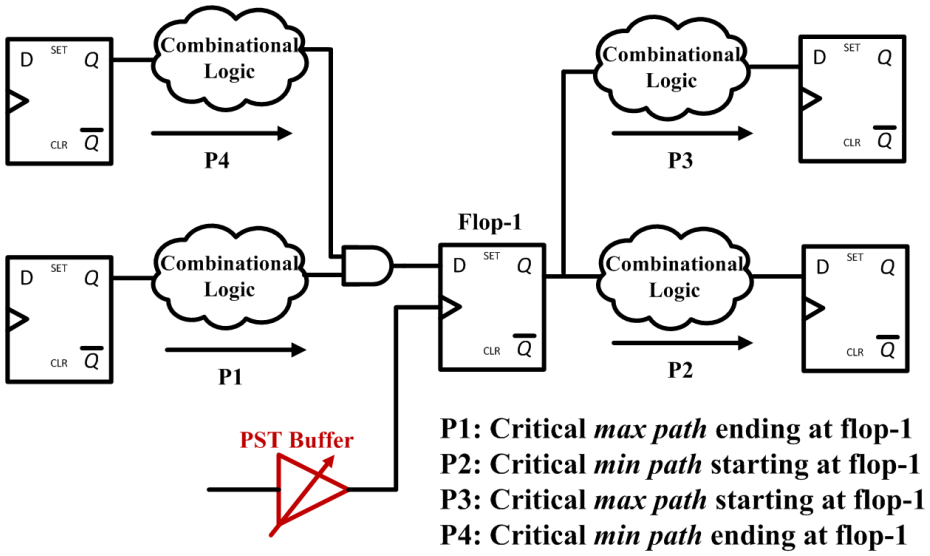


Figure 3.32. Sample design snippet with critical timing paths

Assuming a current starved inverter based PST buffer, Figure 3.30 with 3-bit configuration and monotonically increasing delays from 000 to 111, the buffer can be used to fix setup violations on path P1 and hold violations on path P2. However, increasing delay on the PST buffer also reduces slack of the most critical max path starting at flop-1 (P3) and most critical min path ending at flop-1 (P4). To fix 3 hold time violation on path P2 using a PST buffer at flop-1,

$$Slack_{hold}\left(P4\right) \geq abs\left(min\left(Slack_{hold}\left(P2\right)\right)\right) \tag{3.5}$$

Clock tree synthesis/optimization step has to satisfy equation 3.5 by inserting a PST buffer at the start point of P4 or having adequate hold slack to account for process variation. Any additional positive slack beyond this provides additional margin for tuning the max path P1 and has negligible impact on choosing the granularity of tunable delay values. Hence, we do not consider timing path P4 in the rest of this analysis.

The PST buffer tuning steps are as shown in Figure 3.33. Since fixing hold time violations are more critical for the functionality of a chip, the PST buffer is first configured to meet timing on min path P2. At the end of hold fixing, the PST buffer is further tuned if path P1 does not meet the setup requirement for maximum design frequency. The tuning process ends if the configuration bits b[2:0] = 111 or the best trade-off between slacks of paths P1 and P3 is achieved. In practice, this algorithm/flow may be implemented during test by configuring the PST buffer, sensitizing critical min (P2) and max timing paths (P1, P3) and checking for hold time violations and maximum operating frequency.

Tuning the PST buffer to fix path P2 results in a corresponding decrease in slack of path P3, thereby reducing binning yield. In a conventional n-bit configuration linear delay n-bit configuration PST buffer, the 3 setup slack variation is divided into 2n-1 equal intervals for. Since 67% of chips violating min time have a negative slack in the range of 0 to $\sigma_{slack_{hold}}$, linear delay values force large shift in hold slack in these chips. Similarly, 67% of chips with positive slack on path P3 have slack in the range of 0 to $\sigma_{slack_{setup}}$. With increase in $Slack_{hold}(P2)$, path P3 in these chips observe a corresponding reduction in setup slack and can potentially be decreased to less than zero. This results in parametric yield loss. Non-linear tunable delay values with smaller delay shifts with in one standard deviation of hold slack lead to smaller degradation of $Slack_{setup}(P3)$ and the performance of the chip. Similarly, when tuning the PST buffer to increase $Slack_{setup}(P1)$, there is a corresponding decrease in $Slack_{setup}(P3)$. Smaller tunable delay values with in one standard deviation of $Slack_{setup}(P1)$ provides a higher probability of balancing slack between paths P1 and P3 to achieve the smallest possible Worst Negative Slack (WNS) in the chip. Larger tunable delay values can be used to tune slack values smaller than -2σ to cover the entire range.

Since critical max paths and critical min paths in a clock group have different mean and variance of slack distribution, we estimate tunable delay values based on

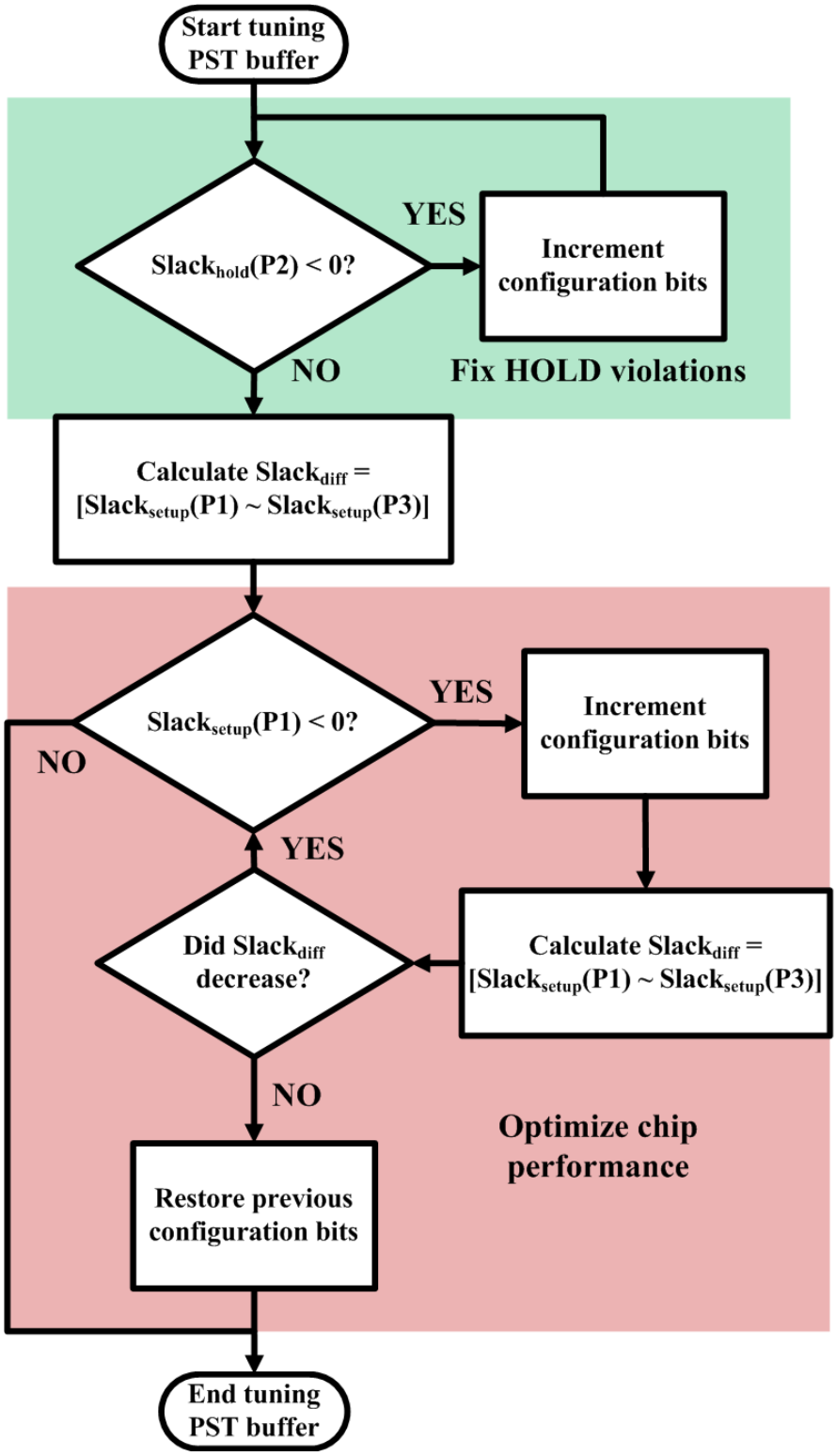


Figure 3.33. Steps for tuning PST buffers

the distribution of max paths. Timing paths that have setup time violations have larger delays and hence larger standard deviation in slack distribution. PST buffer designed to target 3σ variation in setup slack is guaranteed to cover 3σ variation in hold slack. Tunable delay values may be estimated using only setup slack distribution or hold slack distribution based on the criticality of timing paths. Since the setup slack has a Gaussian distribution, the percentage of failing chips with slack range $[0to{-}1\sigma] \sim 68.2\%$; with slack range $[-1\sigma to{-}2\sigma] \sim 27.2\%$; range $[-2\sigma to{-}3\sigma] \sim 4.2\%$. Accordingly, we also divide the available 7 (001-111) configurations into groups of 4, 2 and 1. The delay values of configurations in a group are equally spaced to cover one standard deviation of slack variation. Figure 3.34 shows a sample distribution of setup slack and tunable PST buffer delay values with linear and non-linear intervals. Slack greater than zero has the default PST buffer configuration of 000 (0).

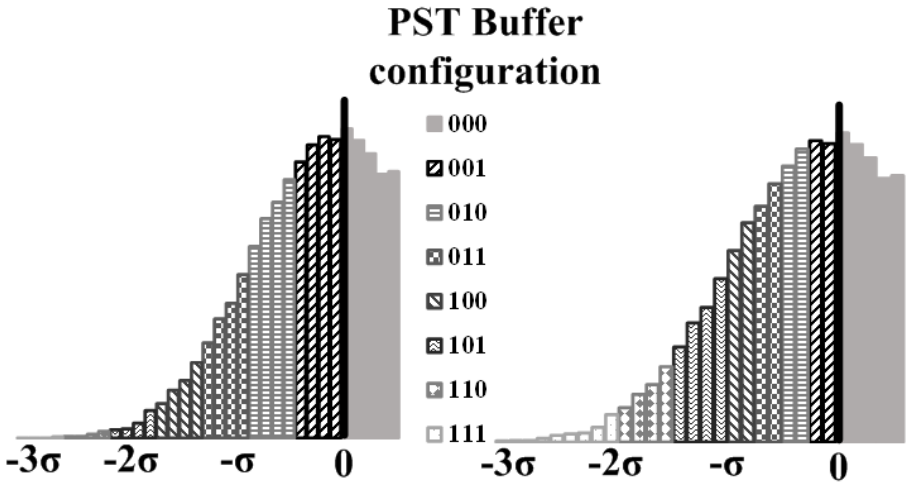


Figure 3.34. Linear and Non-linear PST delay distribution

To further quantify the benefit of using non-linear delay tuning, let us assume the design in Figure 3.32 operating at 3GHz with Slackhold(P2)=N(0,5ps) and Slackhold(P1,P3)=N(0,15ps). The tunable delay values and frequency binning using conventional linear delay and the proposed non-linear delay techniques are shown in

Table 3.2 and Figure 3.35 respectively. Using non-linear delays increases the number of chips in higher frequency bin from 89.68% to 93.37%; thereby improving performance binning yield by $\sim 4.11\%$. Apart from increased binning yield, designing PST buffer for each clock group depending on the expected variance of timing slack provides optimized configurable device sizes. This reduces silicon area and decreases leakage power overhead due to PST buffers.

Table 3.2. Delay values for Linear and Non-linear PST buffers

Configuration	Linear Δdelay (ps)	Non-linear Δdelay (ps)
001	6.42	3.75
010	12.85	7.50
011	19.28	11.25
100	25.71	15.00
101	32.14	22.50
110	38.57	30.00
111	45.00	45.00

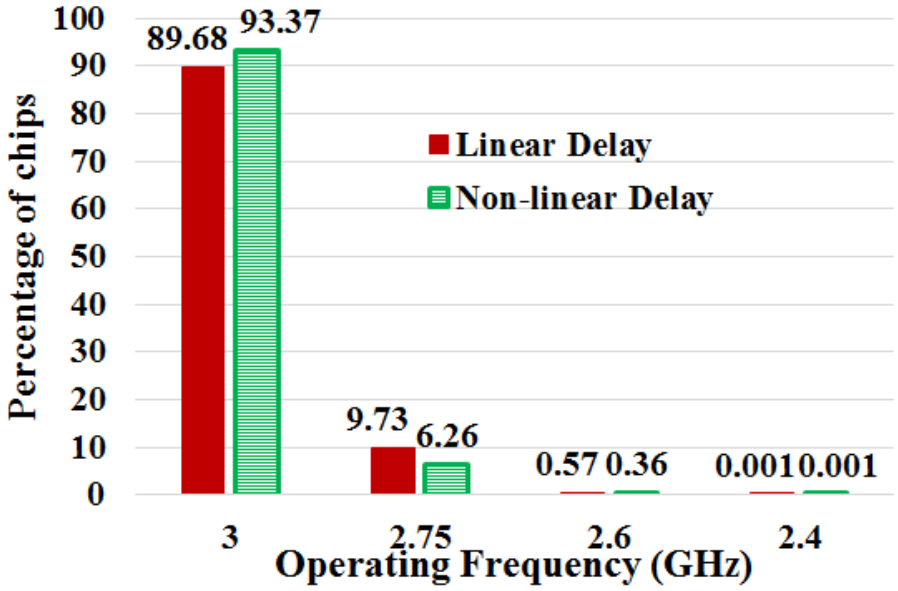


Figure 3.35. Performance binning yield

3.3.2 PST Buffer Design for Non-linear Tunable Delay

Once the range and delay values of each PST buffer is estimated, device sizes for the configurable PMOS/NMOS in the current starved inverter PST have to be calculated. We propose Linear Programming (LP) based technique to estimate the width of devices. The proposed PST buffer design is as shown in Figure 3.36. The buffer has three variable PMOS header devices and three variable NMOS footer devices. These are configured using the configuration bits b[2:0]. The number of configurable PMOS/NMOS devices can be varied for different tuning range and delay precisions. The output inverter boosts the transition time of the clock edges. The non-linear tunable delay technique and LP based device sizing can be extended to other PST buffer designs as well.

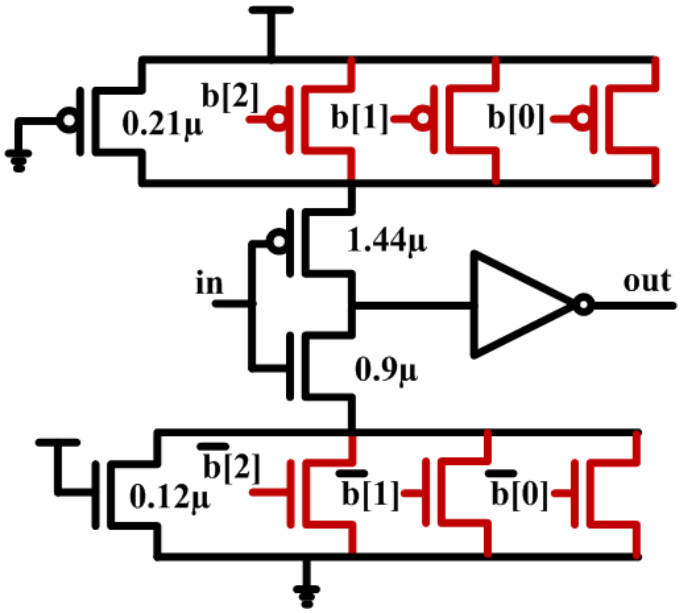


Figure 3.36. Non-linear Delay PST Buffer

The first step of device sizing is generating a curve or equation for PST buffer delay vs configurable device sizes. Figure 3.37 shows the delay of PST buffer for various effective sizes (W_{var}) of configurable transistors. The effective size W_{var} is the sum of widths of all configurable PMOS/NMOS devices that are ON for a given

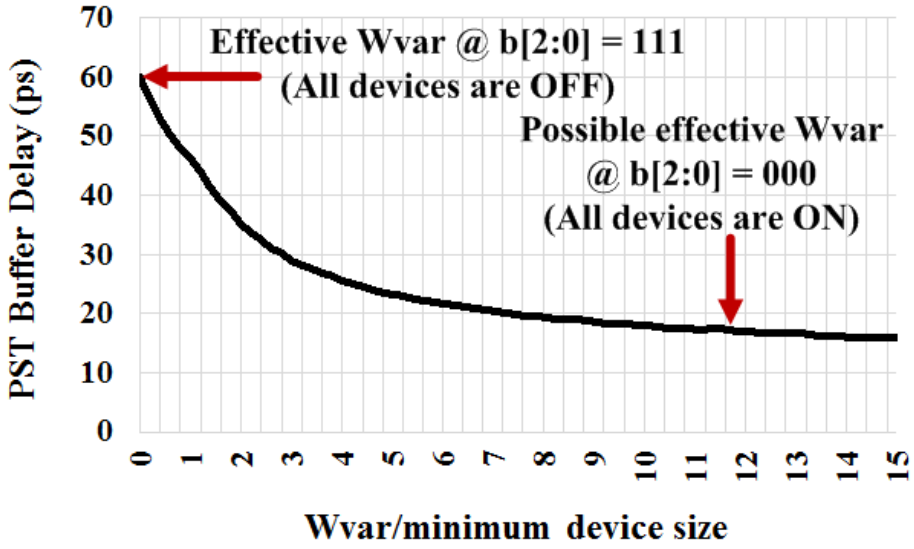


Figure 3.37. Variation of buffer delay with W_{var}

configuration of b[2:0]. The effective width is largest at b[2:0]=000 when all devices are ON. The buffer delay at $W_{var} = 0$ is synonymous to the configuration b[2:0]=111 when all configurable devices are OFF. This data is used to derive an equation for effective width as function of PST buffer delay,

$$W_{var}(delay) = fn\,(PST\ buffer\ delay) \tag{3.6}$$

Algorithm: Generate Equations for Configurable Device Width

Require: DefaultBufferDelay, DeltaDelay[6:0]
Function: Wvar(delay) = fn(PST buffer delay)

1. **for i = 1 to 7 do**
2. b[2:0] = binary(i)
3. $(\sim b[2]Wp_{var_2}) + (\sim b[2]Wp_{var_1}) + (\sim b[2]Wp_{var_0})$ = Wvar(DefaultBufferDelay + DeltaDelay[i-1])
4. **end for**

Figure 3.38. Algorithm for generating equality conditions

To calculate the required configuration devices sizes, the estimated tunable delay values are mapped to effective widths Wp_{var} and Wn_{var} using Equation 3.6. A set of seven equation for each ΔPST buffer delay values are obtained starting with the target default buffer delay, as shown in Figure 3.38. Linear Programming is used to estimate the minimum sizes of $Wp_{var}[2:0]$ which satisfy the seven equality conditions. In most cases, LP does not converge into a feasible solution to satisfy all equalities. The solution with least deviation from the equality conditions is chosen as the transistor sizes. This solution also results in the least deviation from estimated PST buffer delay values. To further validate the results, delay distributions of timing paths are also modeled along with LP solutions to estimate the binning yield and choose the most appropriate device sizes.

3.3.3 Implementation and Results

The proposed PST buffer design was implemented for a set of ISCAS89 benchmark circuits. The benchmark circuits were implemented in IBM 32nm SOI process using Synopsys tool suite. Statistical timing analysis was performed using Synopsys PrimeTime and linear programming in MatLab. HSPICE was used for circuit simulation and power estimation of PST buffers.

Figure 3.39 shows the μ/σ of critical paths in ISCAS89 circuit s1423. The end path slack refers to the critical max path that needs to be fixed using the PST buffer (Path P1 in Figure 3.32). The start path slack refers to the most critical path starting from the flip-flop clocked by the PST buffer (Path P3 in Figure 3.32). The critical case for device sizing of PST buffer are critical end paths whose corresponding start paths have a mean slack close to zero and large standard deviation.

Fig. 12 shows the percentage of chips that can be tuned for eight critical paths to meet the target frequency of 3GHz using linear and non-linear delays. We also demonstrate a third scenario where a single buffer designed for the most critical path (path with

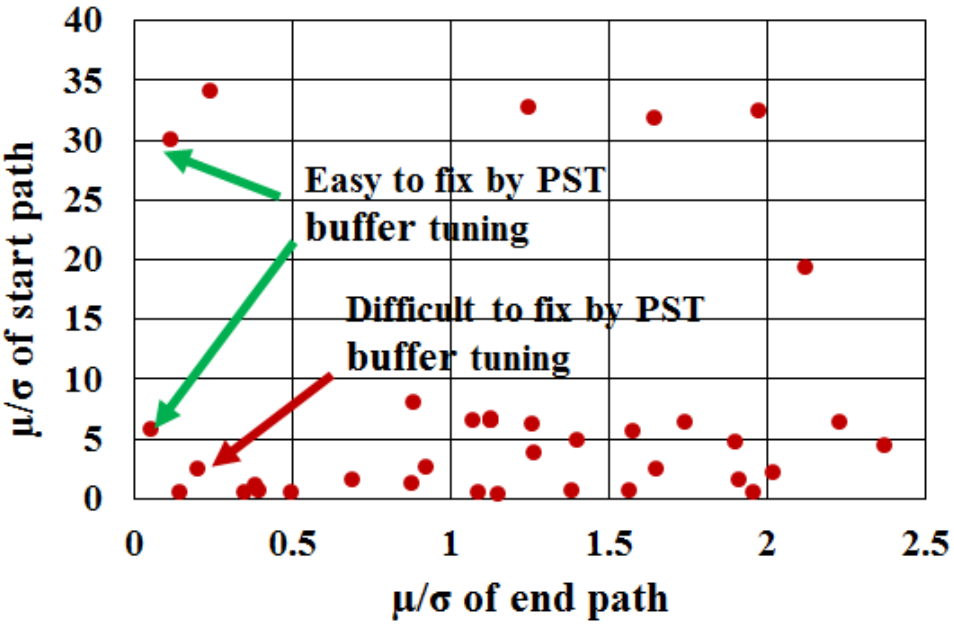


Figure 3.39. Criticality of timing paths

least difference between mean end path slack and mean start path slack). This single non-linear delay PST buffer is used to tune all the critical paths. Designing dedicated buffers for each path provides the best performance improvement. Using a single non-linear delay PST buffer does not provide the same level of performance, but involves significantly less design effort. This technique can be extended by pre-designing more than one standard non-linear delay PST buffer and choosing the appropriate one for each critical path depending on the delay statistics.

Table 3.3. Comparison of binning yield, area and leakage power for ISCAS'89 benchmark circuits

ISCAS'89 Circuits	Performance Binning Yield			Effective Area (μm)			Leakage Power (μW)		
	PST-1	PST-2	PST-3	PST-1	PST-2	PST-3	PST-1	PST-2	PST-3
s1423	92.12	89.96	87.50	50.2	56.89	79.1	1.38	1.4	1.55
s9234	91.46	89.32	88.18	61.7	64.98	94.13	1.71	1.72	1.84
s15850	91.13	88.94	87.48	274.03	310.57	397.48	6.83	7.53	8.04
s35932	89.28	88.38	86.53	256.86	275.12	320.39	5.96	6.91	7.76
s38584	90.12	89.76	87.19%	181.81	218.58	271.7	5.03	5.12	6.74

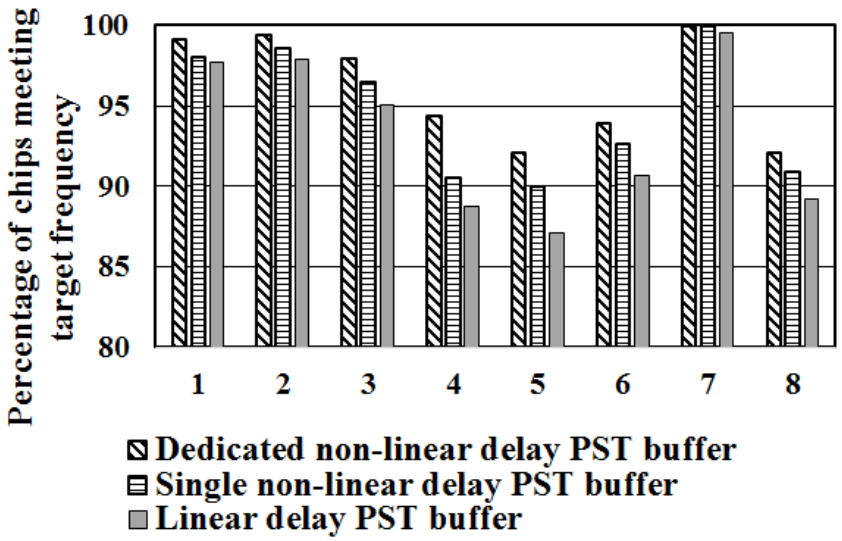


Figure 3.40. Performance enhancement of critical paths in s1423 circuit

Similar experiments were performed on other ISCAS89 benchmark circuits. The results are as shown in Table 3.3, where PST-1: Dedicated non-linear delay PST buffers; PST-2: Single non-linear delay PST buffer; PST-3: Linear delay PST buffer. The relative area of PST buffers are compared using the sum of widths of configurable transistors. Using dedicated PST buffers for each critical path with non-linear delay values improves performance binning yield by more than 4% compared to linear delay PST buffers. The parametric yield can be trade-off for design time by using a single non-linear delay PST buffer for all critical paths. This technique increases binning yield by ∼2.2% over traditional PST buffers. This can be further improved by designing a set of non-linear delay PST buffers and choosing the appropriate delay distribution for each critical path. The improvement in parametric yield due to non-linear buffer delay also depends on the number of critical end-path/start-path pairs in the design. Non-linear sizing also optimizes buffer size and leakage power. Compared to conventional sizing, non-linear delay buffers reduce configurable transistor area by more than 30% and leakage power by up to 20%.

CHAPTER 4

LOGIC TECHNIQUES TO MANAGE VARIATIONS

Logic techniques use algorithmic post-processing to detect and correct errors due to process variation. Error Correction Codes (ECC) are widely used for memory systems to detect and correct bit errors . These errors could occur due to process variation, device aging or single event disturbances (soft errors). In this chapter we explore logic correction techniques for a biased True Random Number Generator (TRNG). In the first section, we estimate entropy and energy bounds for using lightweight post-processing techniques for TRNG. We develop a stochastic model for metastability based TRNG to incorporate the impact of variations and noise to estimate expected entropy. The stochastic model is then used to determine the lower bound on energy overhead to use lightweight post-processing to generate high quality random bits. In the second section, we present REFLEX: A novel Re-configurable Logic for Entropy Extraction, that uses re-configurable logic to choose xor and non-xor functions for entropy extraction, based the raw bit entropy of TRNGs.

4.1 Energy and Entropy bounds for Lightweight Post-processing

Post-processing techniques for TRNGs lead to additional area and power overhead. The power overhead translates into overhead in energy/bit generated from the TRNG. This is a major limiting factor in low-power application. Categorizing post-processing techniques as lightweight or not is subjective to the application. An AES post-processing may not constitute a large overhead in a multi-core processor. However, it is a significant overhead for TRNGs implameneted in an SoC. In this work,

we consider a post- processing technique to be lightweight if it can be implemented using less than a thousand logic gates. The implementation area of post-processing techniques considered in this work range from a few tens of gates in von Newmann corrector and XOR tree to few thousand gates in PRESENT cipher. Previous literature on lightweight techniques analyze the security impact on random bits generated [52, 22]. In [52], Kwon *et. al* compare the security, throughput and area impacts of linear compression technique (a varient of LFSR) and von Neumann correction. They propose a metric called advesary bias to quantify how accurately an adversary can predict the post-processed TRNG bits. In [22], a post-processing compression function is proposed that combines multiple bits from a biased TRNG to output high quality rnadom bits. Apart from conventional lightweight post-processing techniques, resilient functions and cyclic codes may also be used to improve randomness of a biased TRNG [54, 95].

Previous works on lightweight post-processing do not consider the impact of physical implementation of TRNG circuit and the related variations. The bias introduced due to process variation has to be considered to choose the appropriate post-processing technique and minimize area and energy overhead. In this work, we estimate the lower entropy bounds at the output of a TRNG required for each post-processing technique. We present a formal stochastic model of metastability based TRNG considering the impact of variation and noise. The stochastic model is not limited to a formal analysis of metastability based TRNG circuits. We use the stochastic model to predict the expected bias in the circuit. Further, we collate the estimated entropy bounds and derived estimtaed entropy to choose the best lightweight post-processing technique depending on the bias of the TRNG. We also estimate the lower energy bound; the minimum energy overhead depending on the amount of variation, noise and the post-processing technique used.

4.1.1 Stochastic Model for Metastability based TRNG

Analog TRNG circuits use amplifier and ADC which are not energy efficient and do not scale well with technology. Ring Oscillator based TRNG circuits require constant oscillation of multiple RO resulting in large dynamic power. They are also sensitive to noise on the power supply. This makes them vulnerable to attacks on the global power grid [62]. Metastability based TRNG circuits have a very lightweight implementation. They are active only during random bit generation and are highly energy efficient. A metastability based TRNG consists of a cross-coupled inverter pair, shown in Figure 4.1. The nodes A and B are pre-charged to supply voltage during the negative level of clk and allowed to resolve during the positive cycle. Ideally, the resolution states of A and B depend solely on the thermal noise present at the nodes. However, process variation leads to device mismatch between transistors M1 and M2, making one stronger than the other. This introduced bias in the TRNG requiring additional post- processing.

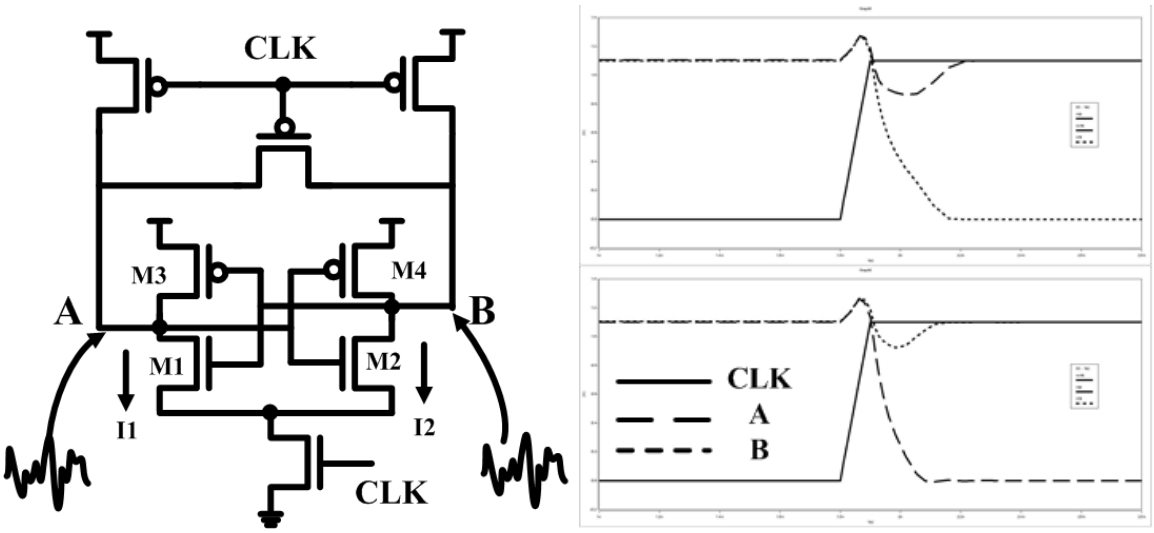


Figure 4.1. Metastability based TRNG

When the metastability based TRNG circuit is precharged and then allowed to resolve to a stable state, the pull down NMOS devices M1 and M2 dominate the

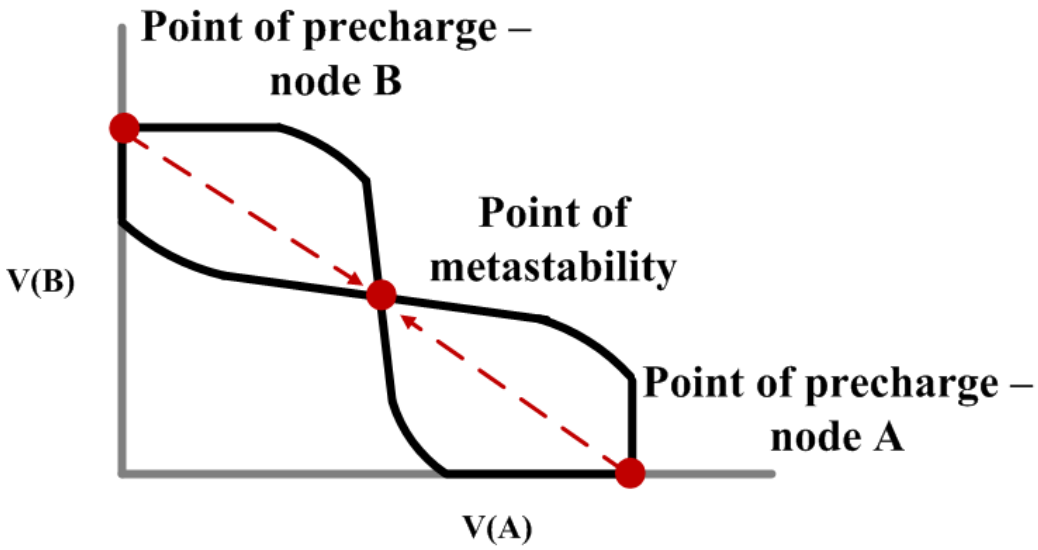


Figure 4.2. Voltage Transfer Characteristic of cross coupled inverters

resolution state. As a result, in the following stochastic model we consider the currents through M1/M2 and ignore the effect of transistors M3/M4. We also do not consider any short channel effects on transistor behavior to decrease the complexity of this model. Figure 4.2 shows the Voltage Transfer Characteristic (VTC) of the cross coupled inverter. During precharge, both nodes A and B are VDD. Once the precharge is released, both the nodes move towards the point of metastability. During this phase, the NMOS devices M1 and M2 are operating in the saturaion region. The drain currents I1 and I2 in saturation mode are given by,

$$I1 = \frac{\mu_n C_{ox} W}{2L_1}(V_{gs} + V_{noise1} - V_t)^2 \tag{4.1}$$

$$I2 = \frac{\mu_n C_{ox} W}{2L_2}(V_{gs} + V_{noise2} - V_t)^2 \tag{4.2}$$

where, $\mu_n$is the mobility of charges in NMOS, $C_{ox}$is the oxide thickness, W is the transistor length, L is the transistor channel length, V_t is the threshold voltage, V_{gs} is the gate to source voltage and V_{noise} is the thermal noise voltage. For a given length

L_1 and L_2all parameters of the current equation are constant except thermal noise. Thermal noise has a gaussian distribution and hence,

$$V_{noise} \sim N(\mu_{noise}, \sigma^2_{noise}) \tag{4.3}$$

Let,

$$\beta_1 = \frac{\mu_n C_{ox} W}{2L_1} \, and \, \beta_2 = \frac{\mu_n C_{ox} W}{2L_2}$$

and

$$I1 = X1^2 \, and \, I2 = X2^2$$

Then,

$$X1 = \sqrt{\beta_1}(V_{gs} + V_{noise1} - V_t) \tag{4.4}$$

$$X2 = \sqrt{\beta_2}(V_{gs} + V_{noise2} - V_t) \tag{4.5}$$

Since both $V_{noise1}$and V_{noise2} have a gaussian distribution, $X1$and $X2$also have a gaussian distribution with a mean and varience given by,

$$X1 \sim N \left[\sqrt{\beta_1}(V_{gs} + \mu_{noise} - V_t), \, \beta_1 \sigma^2_{noise}\right] \tag{4.6}$$

$$X2 \sim N \left[\sqrt{\beta_2}(V_{gs} + \mu_{noise} - V_t), \, \beta_2 \sigma^2_{noise}\right] \tag{4.7}$$

If the output of the TRNG is node A, the probability of output being a zero is,

$$P(0) = P(I1 > I2)$$

$$= P(X1^2 > X2^2)$$

$$= P(X1 > X2) + P(X1 < -X2)$$

The currents through M1 and M2 are uni-directional. Therefore,

$$P(0) = P(X1 - X2) \tag{4.8}$$

Let

$$Y = X1 - X2$$

Since Y is a linear combination of two gaussian variables, Y also has a gaussian distribution with $\mu_Y = \mu_{X1} - \mu_{X2}$ and $\sigma_Y^2 = \sigma_{X1}^2 + \sigma_{X2}^2$. As a result,

$$\mu_Y = \left(\sqrt{\beta_1} - \sqrt{\beta_2}\right)(V_{gs} + \mu_{noise} - V_t) \tag{4.9}$$

$$\sigma_Y = \sqrt{\beta_1 + \beta_2} * \sigma_{noise} \tag{4.10}$$

Since,

$$P(0) = P(Y > 0) = 1 - P(Y \leq 0)$$

$$P(0) = 1 - \left[\frac{1}{2} + erf\left(\frac{0 - \mu_Y}{\sigma_Y}\right)\right] \tag{4.11}$$

Therefore, the probability of zero is,

$$P(0) = \frac{1}{2} + erf\left(\frac{\mu_Y}{\sigma_Y}\right) \tag{4.12}$$

and the probability of one is,

$$P(1) = 1 - P(0) = P(0) = \frac{1}{2} - erf\left(\frac{\mu_Y}{\sigma_Y}\right) \tag{4.13}$$

For an ideal TRNG, both the NMOS devices M1 and M2 are equally matched in terms of transistor width, length and thresold voltage. As a result, $\beta_1 = \beta_2 = \beta$, $\mu_Y = 0$ and $\sigma_Y = \sqrt{2\beta} * \sigma_{noise}$. This results in $P(0) = P(1) = \frac{1}{2}$, the ideal condition for TRNG and an entropy of 1. However, if the transistior length of M1 is smaller than that of M2, M1 drives a larger current compared to M2. This should bias the TRNG to generate more zeros or in other words, $P(0) > P(1)$. Assuming the condition $L_1 < L_2$,

$$\beta_1 > \beta_2 \, and \, \mu_Y > 0$$

$$P(0): \quad \frac{1}{2} + erf\left(\frac{\mu_Y}{\sigma_Y}\right) > 0$$

$$P(1): \quad \frac{1}{2} - erf\left(\frac{\mu_Y}{\sigma_Y}\right) < 0$$

Similarly, when $L_1 > L_2$, $\mu_Y < 0$ resulting in $P(0) < P(1)$. The values of probabilities of ones and zeroes obtained from the stochastic model can be used to estimate the entropy for a given variation in transistor length. The stochastic model can also be extended to study variations in transistor width and threshold voltage.

Application of Stochastic Model: The stochastic model is not just a formal description of metastability based TRNG circuit, but can be used to further study the impact of noise and process variation on the expected entropy. This helps in choosing the appropriate post-processing technique and estimating a lower bound on energy overhead. The stochastic model was implemented in MatLab using transistor parameters from 32nm Predictive Technology Models. The mean and varience of thermal noise

voltage were obtained from circuit simulation in HSPICE. Figure 4.3 shows the variation of entropy for different effective channel lengths of NMOS in the two inverters. It should be noted that transistor current depends on the effective physical channel length, which is smaller than the designed/drawn channel length. The TRNG is affected by the relative variation in the pull down devices of the two inverters and not on the absolute variation. Hence, maximum entropy is observed when the devices are exactly matched and the entropy decreases with increasing mismatch. The variation in L_{eff} has a gaussian distribution $N(\mu_L, \sigma_L^2)$. As a result, the probability of a particular entropy value depends on the probabilities of L_1 and L_2 for a given $\mu_L$and σ_L.

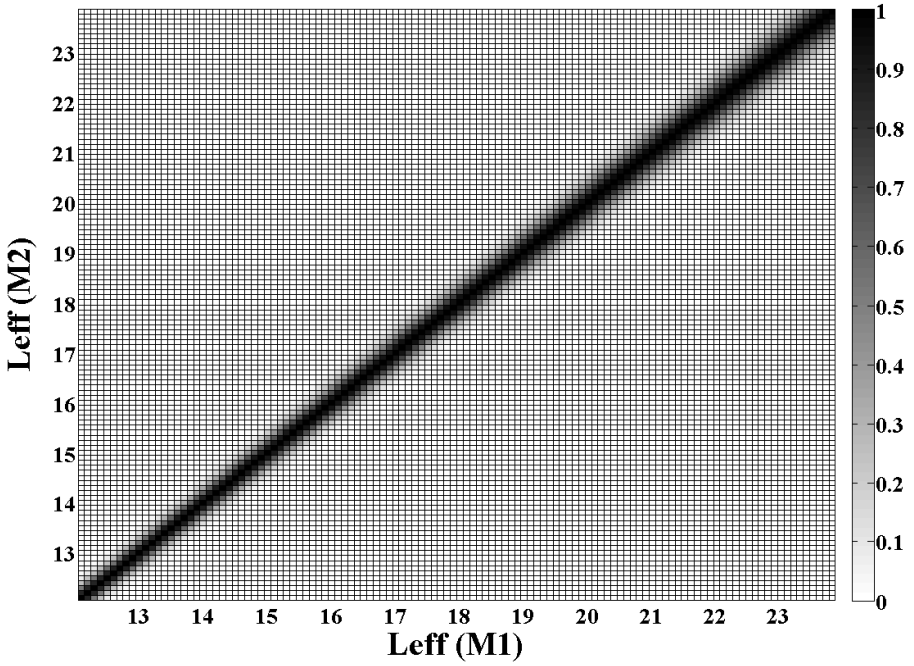


Figure 4.3. Distribution of entropy for variation in L_{eff}

Figure 4.4. shows the weighted distribution for two different σ_L. Lower the standard deviation in length, smaller will be the device mismatch. As a result, the probability of achieving higher entropy increases. Similarly, with increase in varaition of transistor length, the probability of mismatch increases; thereby increasing the probability of

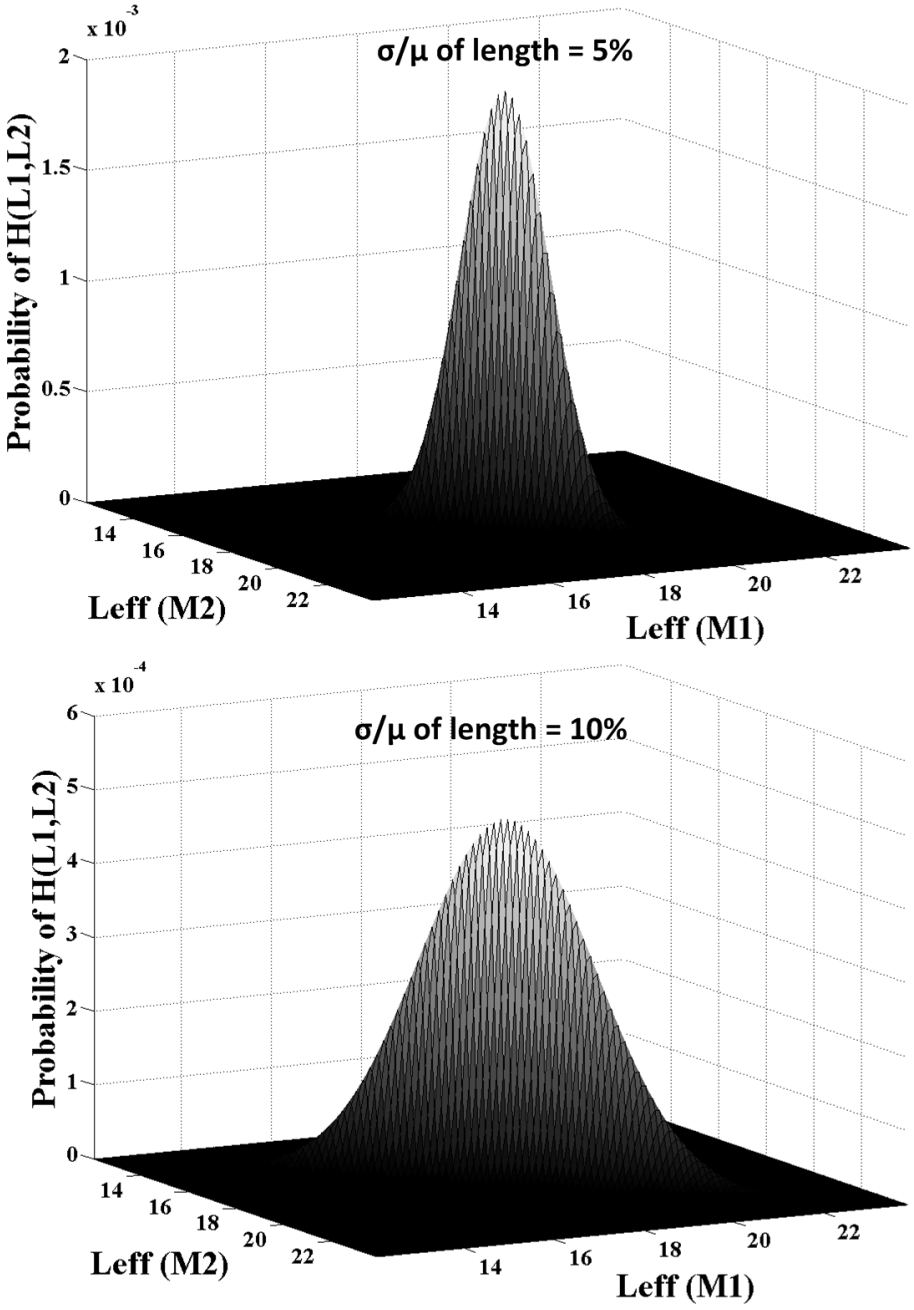


Figure 4.4. Weighted Entropy with variation in σ_L

bias. This factor is synonimous to the variation observed in a particular process or comparing the bias in TRNG across different CMOS process technologies. To further quantify the impact of process variation on TRNG bias, we use the metric of *Expected Entropy*, given by,

$$E[H] = \sum_{i=0.7}^{1.3} \sum_{j=0.7}^{1.3} H(L_1 = i, L_2 = j) * P(L_1 = i, L_2 = j) \tag{4.14}$$

The standard deviation of variations in transistor parameters also depend on the device widths. The variations in L_{eff} and V_{th} decrease for large device widths due to avergaing effect [116, 103]. The stochastic model can be used to explore the impact of variation in L_{eff} for different device widths. Increasing device width decreases the variance in effective channel length; there by resulting in a smaller variance in expected entropy. Similarly, from equation (7), the offset in expected entropy due to intra-die variations is proportional to the difference of $\beta 1$ and $\beta 2$. This difference is amplified by the factor $(V_{gs} + \mu_{noise} - V_{th})$. Lowering the supply voltage (Vdd) and hence the value of V_{gs} decreases the offset in the expected entropy. The stochastic model can be used to explore the variation in TRNG bias for a given process corner and device widths across different values of supply voltages.

The expected entropy provides information about the bias that can be expected for a particular process corner. Accordingly, the appropriate post-processing technique can be chosen. The stochastic model and expected entropy provide the following information:

1. Expected bit rate if von Neumann correction is used.
2. Number of TRNGs to be used or the number of XOR stages.
3. Number of PRESENT cipher rounds to compensate for the bias.

Further, the stochastic model can also be used to estimate the energy overhead for each of these lightweight post-processing techniques.

4.1.2 Lightweight Post-processing Techniques

von Neumann Corrector: The von Neumann corrector was proposed by John von Neumann in 1951 [119]. It takes two consecutive bits from a TRNG and generates one

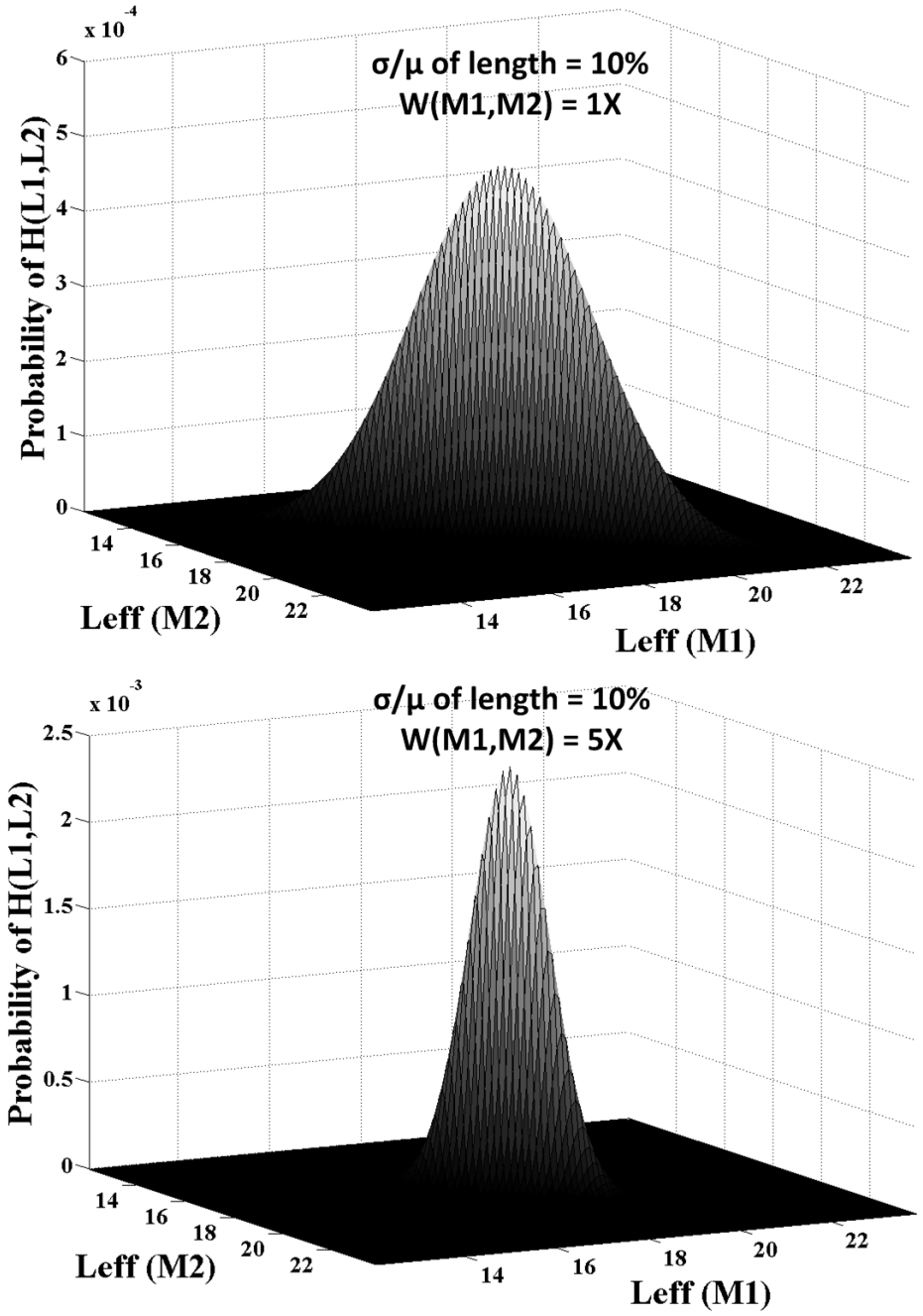


Figure 4.5. Weighted Entropy with variation in transistor width

based on the logic shown in Table 4.1. An illustration of TRNG with von Neumann corrector is shown in Figure 4.6. Assuming uncorrelated bits from the TRNG, von Newmann corrector always generates an output with entropy equal to 1. However, von Newumann corrector significantly reduces the output bit rate. Even with a

completely unbiased TRNG, the probability of von Neumann corrector generating an output bit is equal to 0.25. Hence, this technique invariably reduces the throughput of TRNG by atleast 75%. Since only bit pairs 01 and 10 generate a valid output, the output of von Neumann corrector has a variable bit rate. Any application or protocol using von Newmann corrector should be able to accommodate the variable bit rate.

Table 4.1. von Neumann Corrector

Input bit pair (from TRNG)	Output from von Neumann Corrector
00	No output
01	1
10	0
11	No output

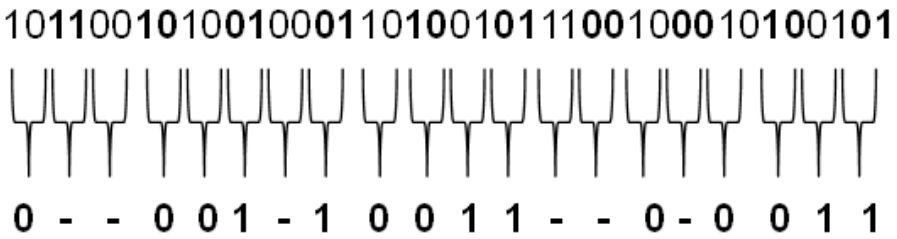


Figure 4.6. TRNG with von Neumann corrector

XOR Function: XOR function is a commonly used post-processing technique [95, 56]. The XOR function is used in the form of an XOR tree which calculates the odd parity of output of two or more TRNG circuits. An illustraion of XOR tree with four TRNGs is shown in Figure 4.7. Although each of the four TRNG is biased, the XOR function accumulates entropy from each TRNG. The net entropy is always greater than or eaqual to the highest input entropy from the TRNGs. Larger bias in TRNGs will require more TRNGs and XOR stages to accumulate enough bias to generate high quality random bits.

PRESENT- Block Cipher: PRESENT is a Substitution and Permutation (SP) network based cipher that was proposed as an alternative to AES for passively power devices like RFID and smart cards [9]. PRESENT takes a 64-bit cipher text and an

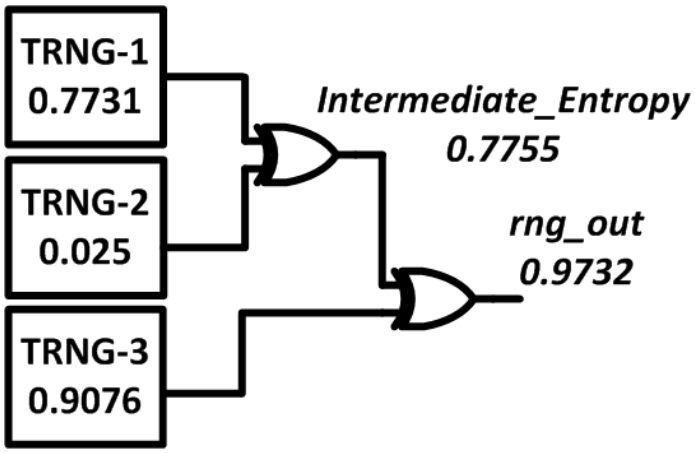


Figure 4.7. XOR tree for entropy extraction

80-bit or 128-bit key to generate a 64-bit ciphertext, shown in Figure 4.8. The default number of rounds in PRESENT is 32. Since the fundamental concept of ciphers is to scramble the input text to appear random, ciphers form an efficient post-processing technique for biased TRNGs. We study post-processing a biased TRNG using 80-bit and 128-bit versions of PRESENT when both the plaintext and key are generated by the TRNG, as shown in Figure 4.9.

Algorithm: PRESENT Block CIpher

generateRoundKeys( ) **for** i =1 to 31 **do**
  addRoundKey(STATE,K_i)
  sBoxLayer(STATE)
  pLayer(STATE)
end for
addRoundKey(STATE,K_{32})

Figure 4.8. PRESENT

4.1.3 Impact of Bias and Entropy Bounds

In this section, we study the impact of TRNG bias on post-processed output of each of the four lightweight techniques. The post-processing techniques are modeled using a combination of C and Perl scripts. The entropy bounds for each post-

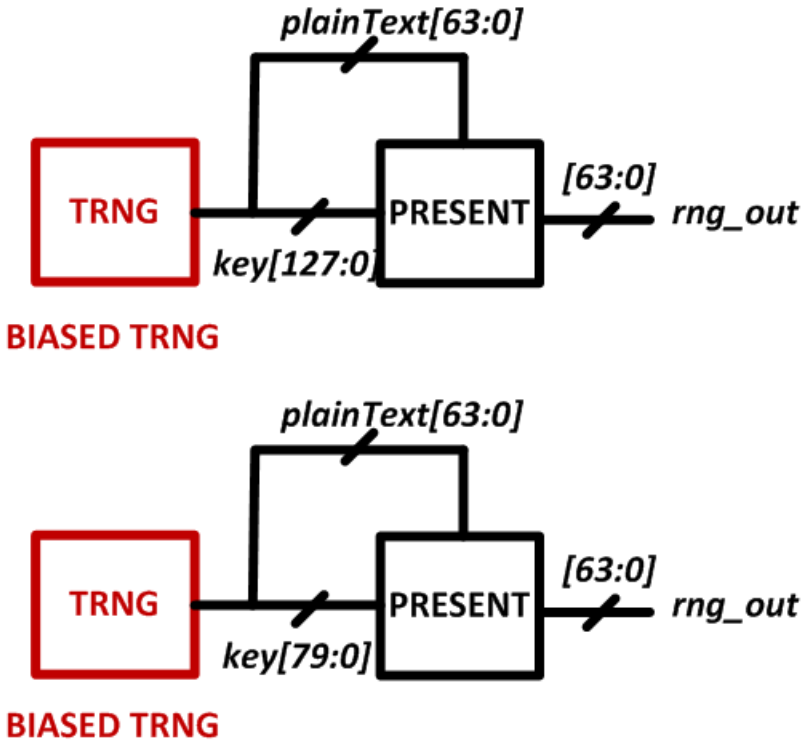


Figure 4.9. Biased TRNG with PRESENT post-processing

processing technique is estimated in the form of minimum Shanon's bit entropy given by,

$$H = -[p(1) * log_2 p(1) + p(0) * log_2 p(0)] \tag{4.15}$$

The entropy bound is defined as the minimum bit entropy required at the output of TRNGV (in other words input of post-processing unit) to obtain high quality random bits. The benchmark for quality of random bits is the NIST statistical test suite with default *p-value* targets [80] and 1 million bits. The input to the post-processing techniques is mimiced by generating random independent bit streams with a sweep of maximum entropy. For scenarios with multiple TRNGs, multiple independent random bit streams are generated with a distribution *uniform[0, max_entropy]*. We also assume the multiple TRNG sources to be un-correlated.

von Newmann Correction: The von Newmann corrector filters simultaneous zeroes/ones and provides an entropy of 1. However, with increase in bias, the number of consecutive zeros or ones generated by a TRNG increase. As a result, a long run lengths of zeroes and ones are discarded leading to decrease in bit rate. The varaition of bit rate and energy ovehead normalized to ideal TRNG is shown in Figure 4.10.

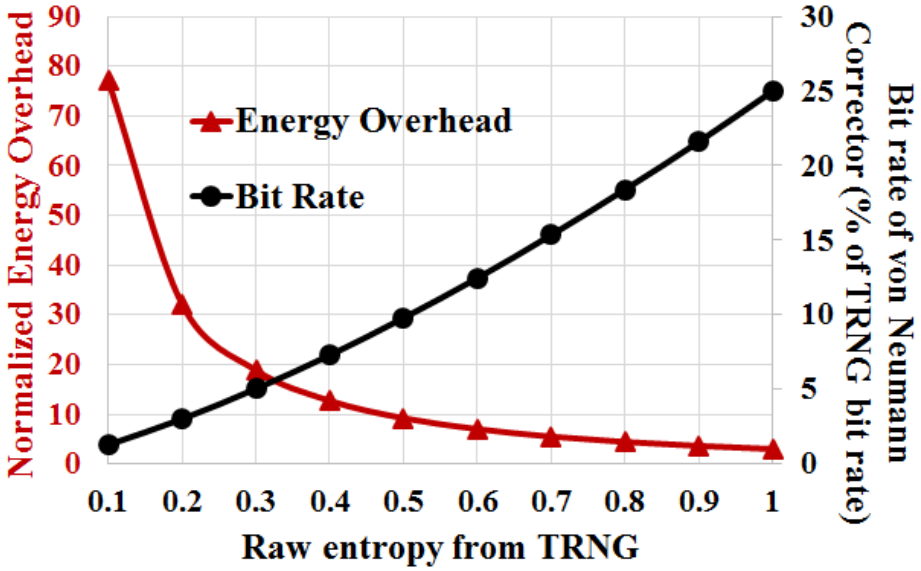


Figure 4.10. Variation of bit rate with input entropy (von Newmann Corrector)

Since the von Newmann corrector always provides near ideal entropy for uncorrelated TRNG bits, all NIST tests pass irrespective of the input bias. However, the bit rate reduces significantly with increase in bias of raw TRNG. Even with an ideal TRNG and input entropy equal to one, the maximum output bit rate of von Neumann corrector is 25% the bit rate of TRNG. Decrease in bit rate will require more TRNG bits to be generated per valid bit generated out of the von Neumann corrector. Therefore, the energy overhead increases with increase in bias, given by,

$$Energy\,Overhead\ = \frac{1 - bitrate}{bitrate} * energy/bit_{trng} \tag{4.16}$$

As a result bias leads to a non-linear increase in energy overhead. To achieve atleast 10% of TRNG bit rate, the minimum entropy at the output of TRNG has to be atleast 0.5 and a 10X energy overhead.

XOR Function: The XOR function calculates the parity of two or more TRNGs. If very few TRNGs are used, although the net entropy of XOR function is always greater than the maximujm input entropy, it will not be of cryptographic quality. Table 4.2 shows the minimum bit entropy required from the second TRNG to pass atleast 12 out of 15 NIST tests when the first TRNG is biased.

Table 4.2. Minimum bit entropy requirement in a two TRNG system with XOR post-processing

Entropy of Biased TRNG	Minimum Entropy of other TRNG
0.91	1
0.92	1
0.93	0.99
0.94	0.99
0.95	0.99
0.96	0.99
0.97	0.98
0.98	0.98
0.99	0.98

XOR function provides entropy extraction for very small bias. In most scenarios, the output does not pass all the NIST tests indicating deficiency in the quality of random bits generated. However, this can be corrected by using more TRNGs and XOR stages to accumulate entropy. To model this, we generated 10,000 samples of chips with number of TRNGs ranging from 2 to 20. The entropies of the TRNGs were modeled to have a distribution of *uniform[0,1]*. Figure 4.11 shows the average bit entropy required to pass all NIST tests and the percentage of chips which pass all NIST tests. As more number of TRNGs are used, the average entropy required to pass all NIST tests decreases. In other words, XOR tree with larger depth can tolerate larger bias in TRNGs. Since the entropies of TRNGs were generated using a

uniform distribution, not all chips pass NIST tests. However, with increase in number of TRNGs the probability of acieveing the minimum average entropy increases and hence a larger percentage of chips pass NIST tests. The energy overhead using XOR function is given by,

$$Energy\,Overhead\ = n * [energy_{XOR} + energy/bit_{trng}] \tag{4.17}$$

$$where\, n = number\, of\, XOR\, stages$$

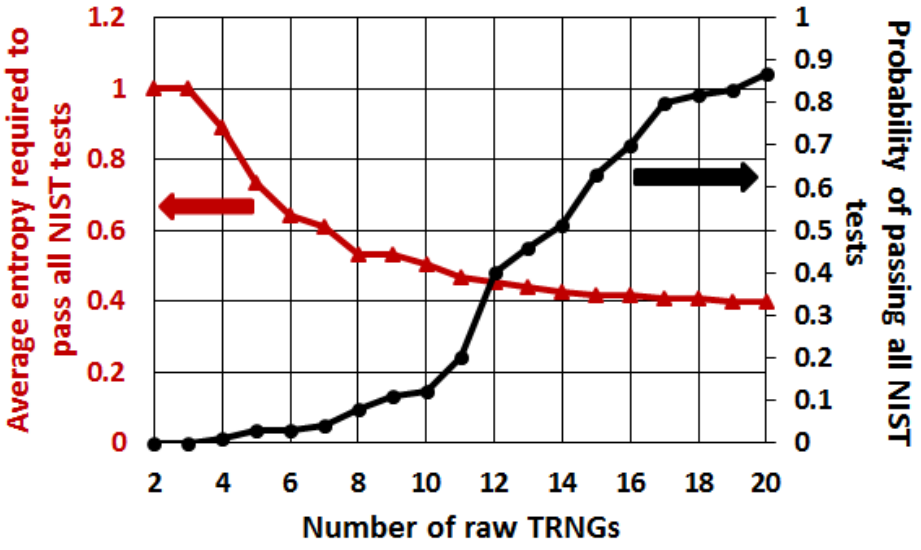


Figure 4.11. XOR post-processing using multiple TRNGs

PRESENT cipher: The PRESENT block cipher was implemented in both 80-bit key and 128-bit key modes. Figure 4.12 shows the number of encryption iterations (each of 32 rounds) for varying raw entropy from TRNG that pass NIST tests. Depending on the input bias, number of iterations of encryption can be varied, Figure 4.13. Accordingly, the energy overhead also increases. Similar analysis for 128-bit key PRESENT is shown in Figure. The energy overhead for 80-bit and 128-bit PRESENT post-processing with variable iterations is,

$$EnergyOverhead = (No_{iter} * energy/bit_{PRESENT-80}) + \{[1 + (1.25 * No_{iter})]\, energy/bit_{trng}\} \tag{4.18}$$

$$EnergyOverhead = (No_{iter} * energy/bit_{PRESENT-128}) + \{[1 + (2 * No_{iter})]\, energy/bit_{trng}\} \tag{4.19}$$

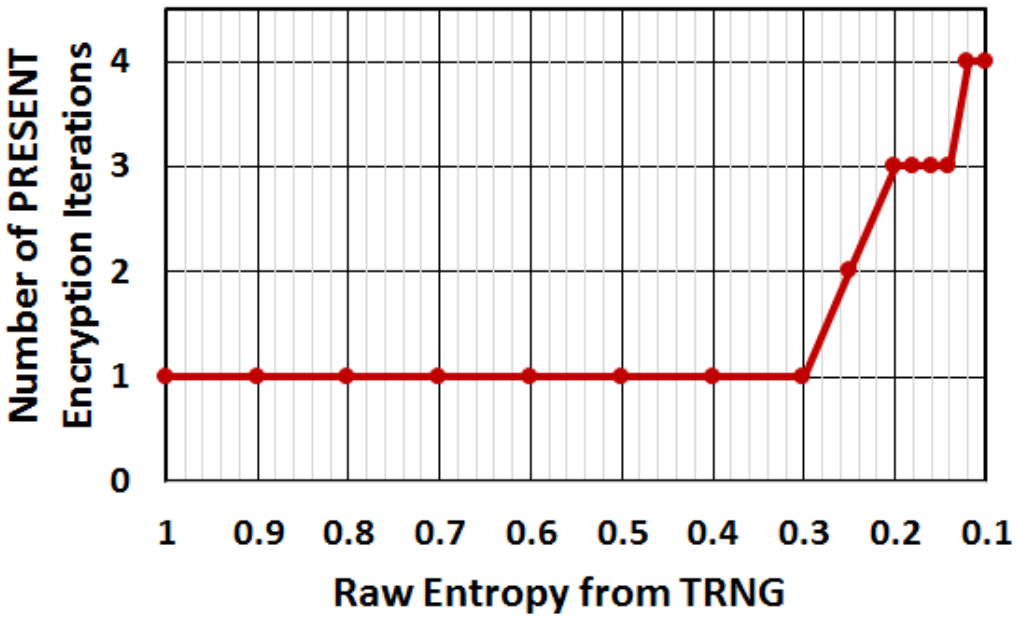


Figure 4.12. Number of PRESENT (80-bit key) encryption cycles required to pass NIST tests

4.1.4 Energy Bounds for Lightweight Post-processing

The stochastic model is implemented in MatLab with transistor parameters derived from 32nm CMOS Predictive Technology Models (PTM). Circuit simulations in HSPICE indicated a thermal noise distribution of $\mu_{noise} = 0\, and\, \sigma_{noise} = 2mV$. The nominal operating condition is 0.9V and 30C. The stochastic model was used to

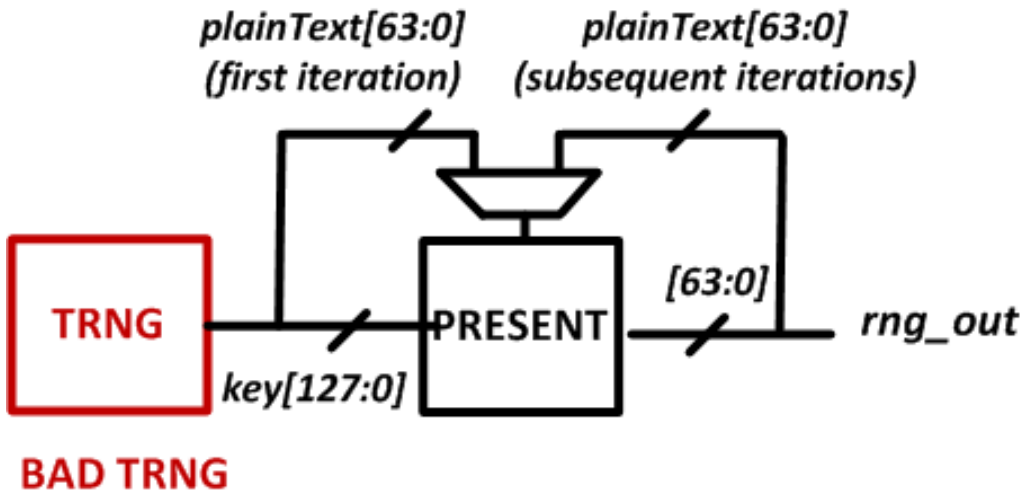


Figure 4.13. PRESENT Post-processing with Variable Iterations

sweep a range of standard deviation of channel length (process corner) and estimate the expected entropy at each process corner. A similar analysis is performed using HSPICE Monte Carlo simulations and transient noise analysis. The data set consists of 100,000 chips for each standard deviation of channel length. Although the results presented in this section pertain to an advanced CMOS technology node, the impact of entropy on post-processing, validation of stochastic model and comparison of energy overhead can be extended to older technology nodes as well. Fig. 4.14 shows the comparison of expected entropy obtained from stochastic model and the mean entropy obtained from HSPICE simulations. The values of *Expected Entropy* obtained from the stochastic model are more pessimistic compared to circuit simulations. This is due to the fact that stochastic model does not incorporate secondary effects of PMOS devices M3/M4 and other CMOS short channel effects. A pessimistic result will only lead to an over designed post-processing stage and provides additional fault tolerance. Fig. 4.15 shows the expected entropy for different widths of transistors M1 and M2 and for different supply voltages (for W(M1,M3) = 1X), as estimated from the stochastic model.

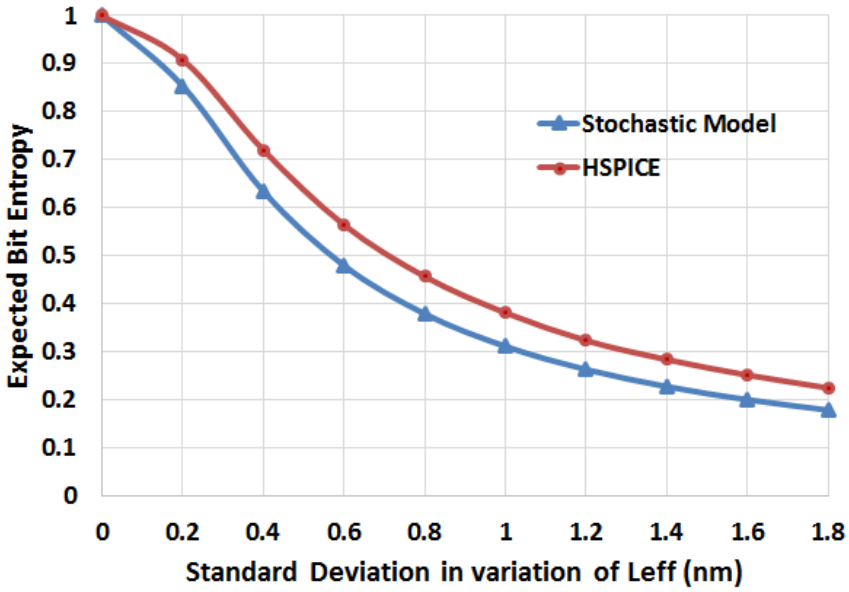


Figure 4.14. Comparison of Expected Entropy

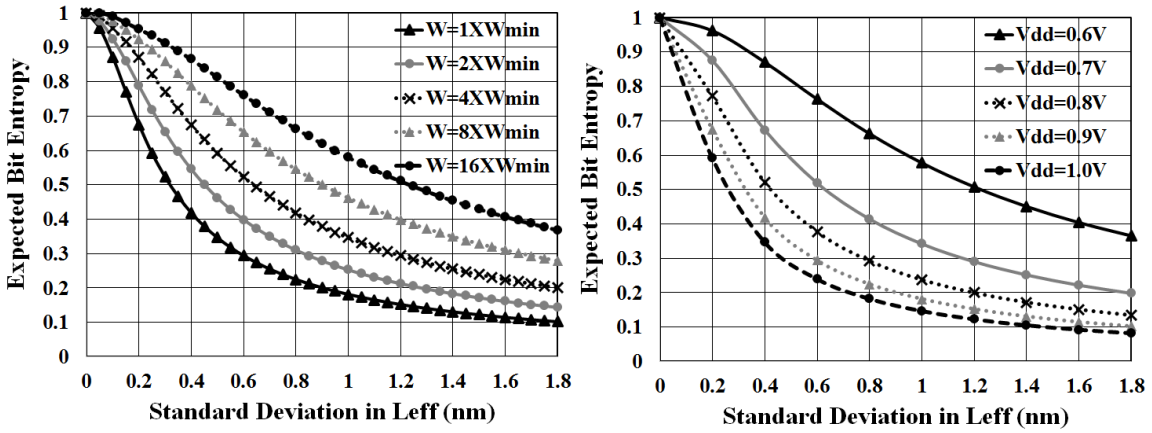


Figure 4.15. Variation of expected entropy with device width and supply voltage

Since the bit rate of von Neumann Corrector can be increased by increasing the entropy of the TRNG sourcing it, the stochastic model can be used to up size the device widths or operate the TRNG at a lower voltage to achieve the optimum bit rate. Fig. 4.16 shows the expected bit rate at the output of von Neumann corrector for different widths of cross coupled inverters and operating voltages. Increasing the device width or operating at a lower voltage increases the performance of random

number generation. However, an important factor for low power applications and SoCs is the energy consumption per bit. Fig. 4.17 shows the variation in expected energy overhead per bit for different device widths and supply voltages. Smaller device widths provide better energy efficiency for process corners with σ_{Leff} less than 0.6nm. However, for large σ_{Leff}, increasing the device widths result in better raw entropy coming from the TRNG. Although this increases the total power per clock cycle, the increase in bit rate at the output of von Neumann corrector increases the overall energy efficiency. Assuming a constant clock period across all voltages, lower Vdd results both in a higher bit rate as well as lower power; thereby resulting in higher energy efficiency.

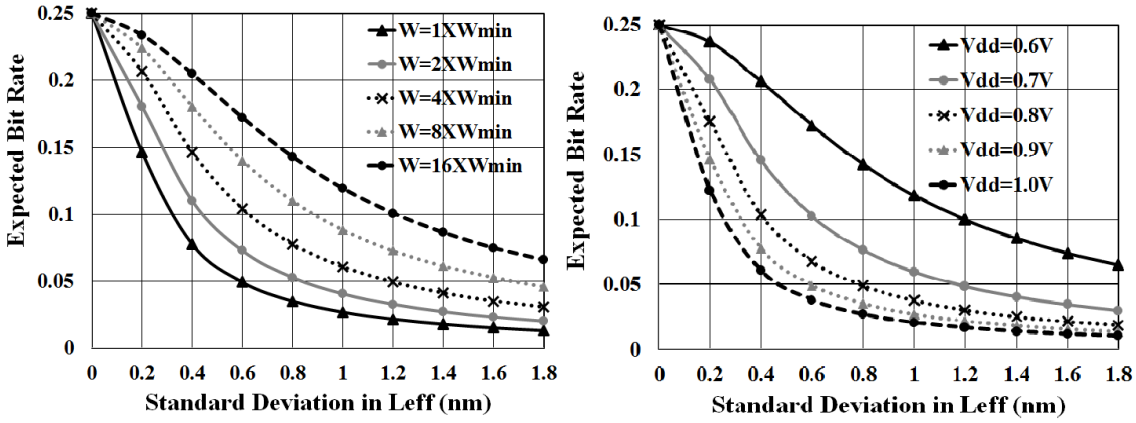


Figure 4.16. Variation of expected bit rate with device width and supply voltage

A similar analysis was performed using the stochastic model for XOR function. The number of TRNGs shown in this result provide a statistical yield of at least 99%; meaning at least 99% of chips fabricated in each of the process corners give high quality random bits by using the estimated number of TRNGs. With increase in device width, the intra-die variation decreases resulting in higher entropy at the output of TRNGs. As a result, fewer TRNGs are required to enhance the over all entropy to pass the statistical tests, 4.18. Further, fewer TRNG circuits and smaller XOR tree lead smaller silicon area for random number generation. Similarly, since

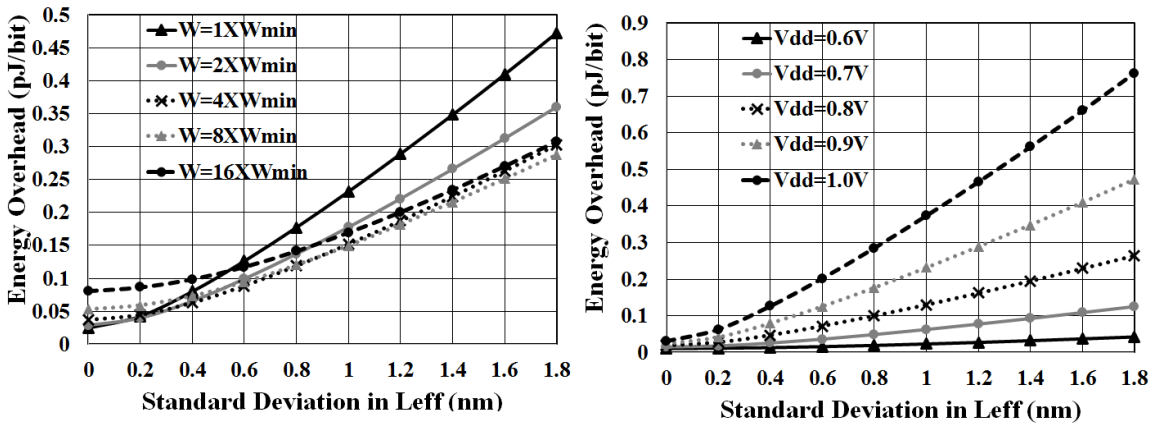


Figure 4.17. Variation of expected energy overhead for different device widths and supply voltages (von Neumann)

decreasing the supply voltage increases the expected entropy of a TRNG, using the XOR function at lower supply voltages requires fewer TRNGs and XOR stages, fig. 4.18. Fig. 4.19 shows the expected energy for different device widths and supply voltages. Although larger device widths require fewer TRNG circuits and XOR gates, the over all energy overhead increases with increase in device width. This is due to the reason that increasing device width does not enhance the raw entropy of the TRNGs enough to cover for the additional power (both dynamic and leakage) due to larger device sizes. Similar to the von Neumann Corrector, lower operating voltages result in the most energy efficient random bit generation.

Post-processing using PRESENT cipher yields similar results with both 80-bit key and 128-bit key modes. Increasing bias in TRNG will require more iterations of encryption, thereby increasing the energy overhead. Since, larger device widths and lower operating voltages will require fewer encryption iteration due to better statistical quality of the bits generated by the TRNG., fig. 4.20. Unlike the von Neumann Corrector and XOR function, the energy per bit for post-processing using PRESENT is dominated by the encryption circuit. As a result, increasing power/energy of the TRNG with increase in device width has negligible impact on the overall energy

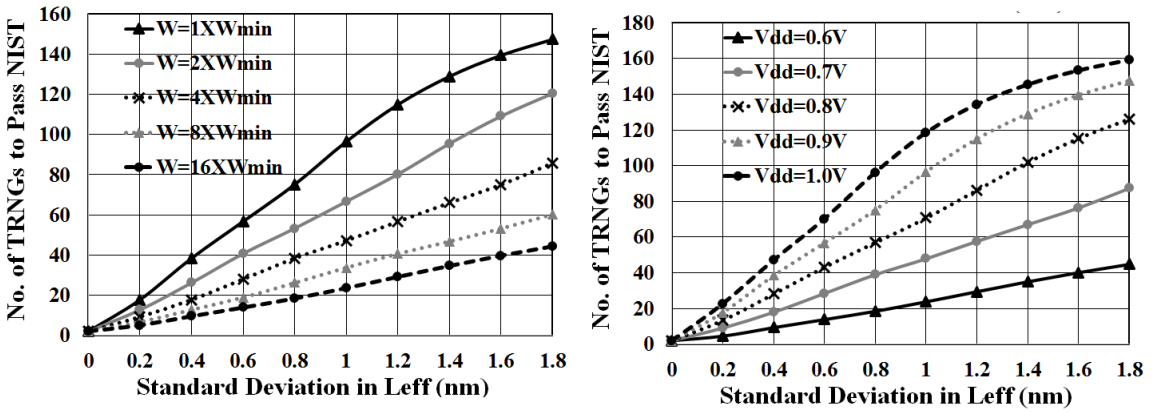


Figure 4.18. Minimum Number of TRNGs required for varying device widths and supply voltages

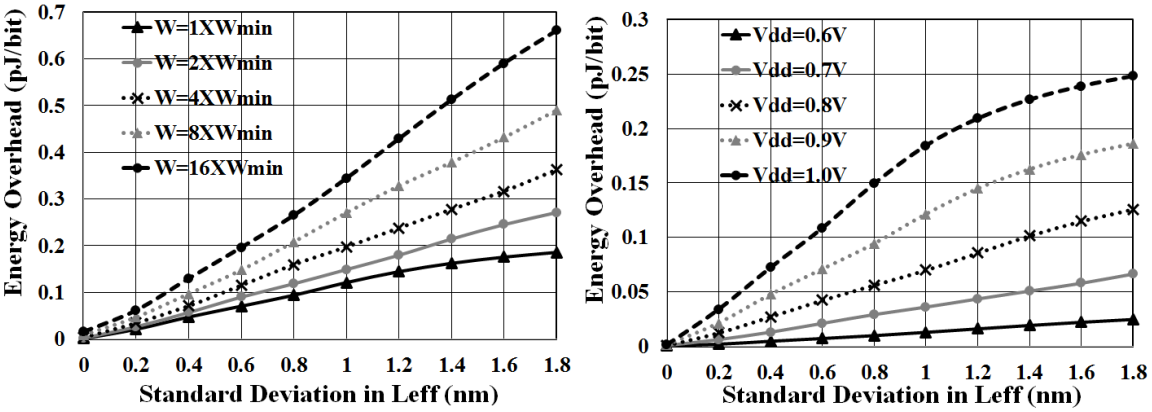


Figure 4.19. Expected energy overhead per bit for varying device widths and supply voltages (XOR Function)

consumption, if the number of PRESENT iterations remain constant. The over all performance of the PRESENT circuit will depend on the circuit implementation and hence the energy/bit values may change depending on the maximum operating frequencies for each supply voltage.

Fig. 4.21 shows the energy overhead of each of the lightweight post-processing techniques for varying σ_{Leff} for minimum device width and Vdd=0.9V . The von Neumann Corrector provides the least energy overhead followed by XOR function. However, von Neumann technique can only be used in applications that can accom-

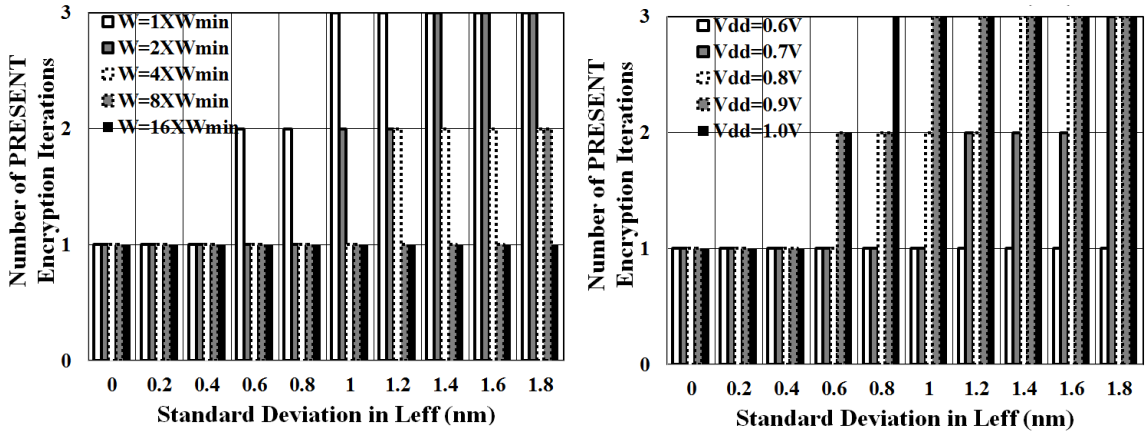


Figure 4.20. Minimum Number of PRESENT iterations required for different device widths and supply voltages

modate variable bit rate RNG. PRESENT consumes the highest energy overhead. However, it is also the most reliable post-processing technique. The energy overhead of PRESENT ranging from ∼1pJ/bit to 2.5pJ/bit is significantly less compared to techniques like AES.

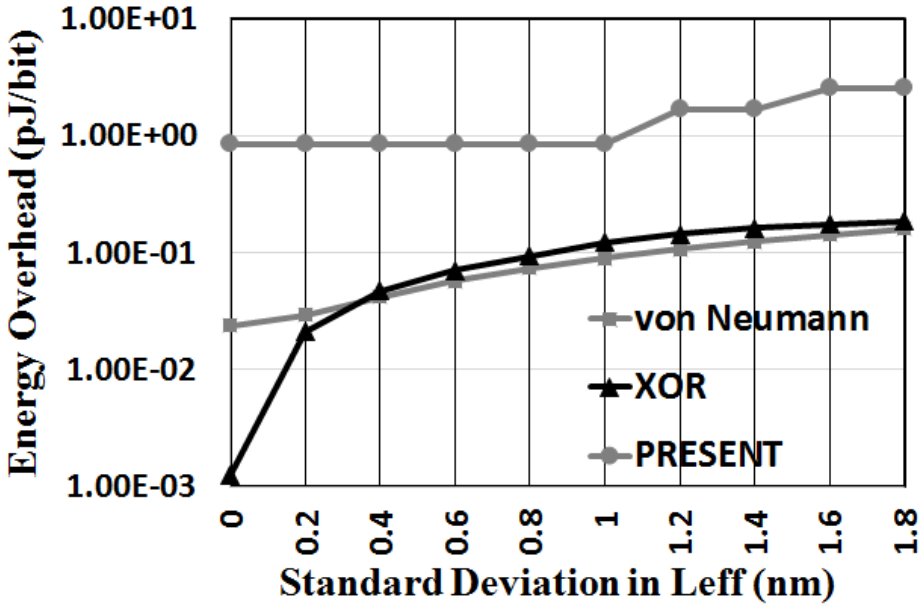


Figure 4.21. Energy overhead for different post-processing techniques for varying σ_{Leff}

4.2 REFLEX: Reconfigurable Logic for Entropy Extraction

Circuit calibration techniques that require calibration during post-Si testing increase test time and cost. Implementing on-chip control for self-calibration require additional control logic depending on the kind of TRNG and increase power/energy overhead. They also introduce the risk of correlation by performing cycle-to-cycle calibration. Complex block ciphers like AES lead to significant area and power overhead. XOR function and von Neumann corrector are two of the most lightweight post-processing techniques. The von Neumann corrector provides near ideal entropy [97]. However, the bit rate reduces drastically with increase in bias.

4.2.1 Xor and Non-Xor functions for Entropy Extraction

XOR function is a commonly used lightweight post-processing technique. It requires multiple TRNGs feeding into an XOR tree. Although statistical tests like the NIST randomness test suite [80] have to be used to validate the cryptographic quality of random numbers, entropy provides a convenient first level estimate of randomness. The XOR tree accumulates entropy from each TRNG to improve the net entropy of the output. Highly biased TRNGs will require more number of XOR stages and hence more number of TRNGs to achieve near ideal entropy. A plot of expected entropy for varying P(1) in a two TRNG system is shown in Figure 4.22. XOR provides near ideal entropy if one of the TRNG has good randomness, that is a P(1)$\sim$0.5. If both TRNGs have large bias, the entropy at output of XOR function is limited to values lesser than 0.9.

Given the output bits A and B of two TRNGs, five unique logic functions can be used to combine the two bits. These are,

1. $Z = \sim A \& \sim B$
2. $Z = \sim A \& B$
3. $Z = A \& \sim B$

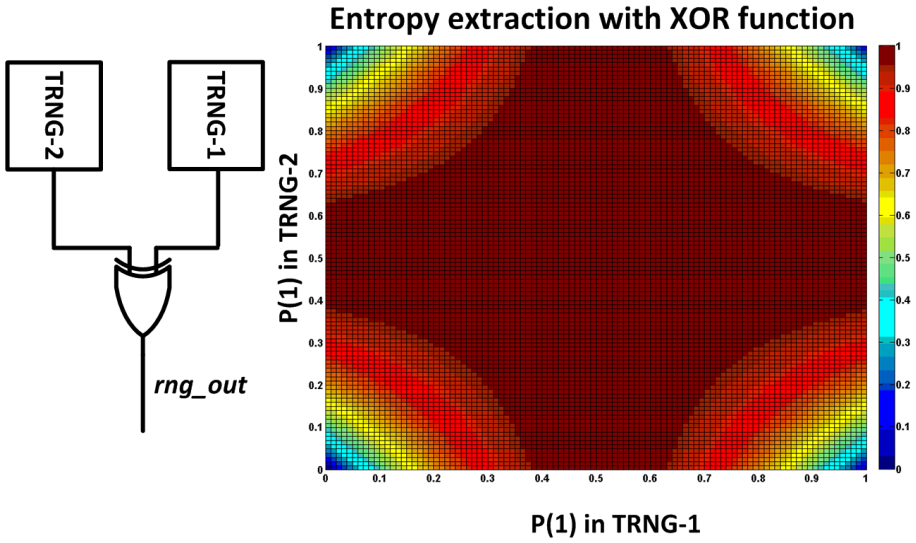


Figure 4.22. Expected Entropy using XOR Function

4. $Z = A\&B$

5. $Z = AxorB$

Other logical functions can be reduced to one of the above unique functions. For instance Z = A or B is equivalent to ∼Z = ∼A & ∼B. Table 4.3 shows the probability of ones, P(1) of the five unique functions given $P(1)_{trng1} = x$ and $P(1)_{trng2} = y$.

Table 4.3. Unique Functions for Combining Output of TRNGs

Logic Function	**P(1) of Z**	**|P(1)-P(0)| of Z**
Z= A& B	(1-x)(1-y)	|1-2x-2y+2xy|
Z= A&B	(1-x)y	|2y-2xy-1|
Z=A& B	x(1-y)	|2x-2xy-1|
Z=A&B	xy	|2xy-1|
Z=A xor B	[(1-x)y]+[x(1-y)]	|2x+2y-4xy-1|

Maximum entropy is achieved when $P(1)_Z = P(0)_Z$ or $\mid P(1)_Z - P(0)_Z \mid = 0$. For a given pair of probabilities of ones, x and y, the logic function that provides the best entropy is the one with least value of $\mid P(1)_Z - P(0)_Z \mid$. For instance, if x=0.2 and y=0.2, $\mid P(1)_Z - P(0)_Z \mid_{z=AxorB} = 0.36$ and $\mid P(1)_Z - P(0)_Z \mid_{z=\sim A\&\sim B} = 0.28$.

Therefore, the function ∼A&∼B provides better entropy extraction compared to XOR function. A plot of entropy for non-XOR functions with varying values of x and y is shown in Figure 4.23. For large biases, the non-XOR functions provide better entropy extraction compared to XOR. The highest entropy obtained in each of these functions is complementary to XOR function. As a result, the best entropy for a given set of TRNGs is obtained using a logic tree based on entropies of the TRNG circuits feeding it. Figure 4.24 shows the output entropy with configurable logic for varying input probabilities of ones in a two TRNG system. Figure **??** shows the comparison of output entropy of XOR function and configurable logic. The highlighted regions show combinations of input P(1) for which non-XOR functions provide better entropy than XOR function.

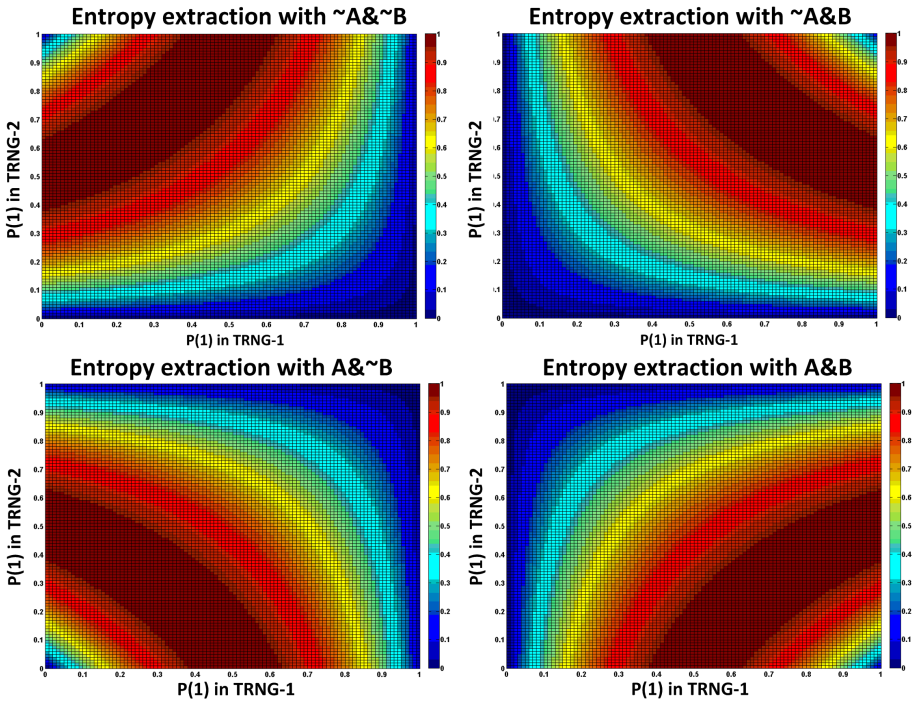


Figure 4.23. Expected Entropy using Non-XOR Function

In this work, we propose a reconfigurable logic block, REFLEX which monitors the entropy of TRNGs and accordingly configures a logic tree to achieve the best possible

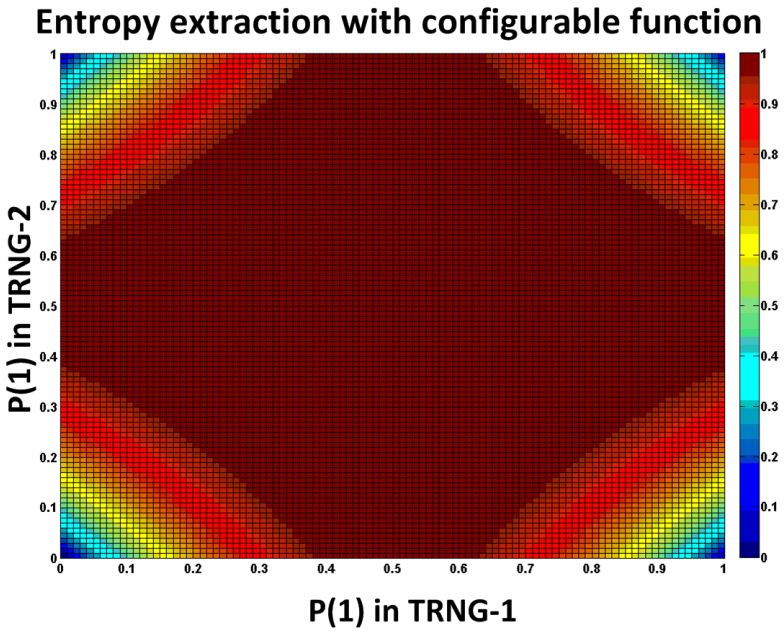


Figure 4.24. Expected Entropy using Configurable Function

net entropy. The reconfiguration is performed once upon power up or periodically depending on the expected variation in the entropy of TRNGs. Since the configuration process does not happen every cycle, the power and energy overhead due to REFLEX is significantly small compared to complex cipher based post-processing. This is critical for low power applications like SoC and passively powered devices like RFID and smart card. Although the proposed technique incurs an area overhead compared to conventional XOR tree, the overhead in energy/bit is comparable to XOR; but with better entropy extraction. Figure 4.26 further illustrates the advantage of using REFLEX over conventional XOR tree.

4.2.2 REFLEX: Reconfigurable Logic for Entropy Extraction

Logic Selection: The design of logic selection module for 4-TRNG system is shown in Figure 4.27. The logic selection module estimates the probability of ones of each TRNG and selects the logic function which provides the best entropy extraction. In

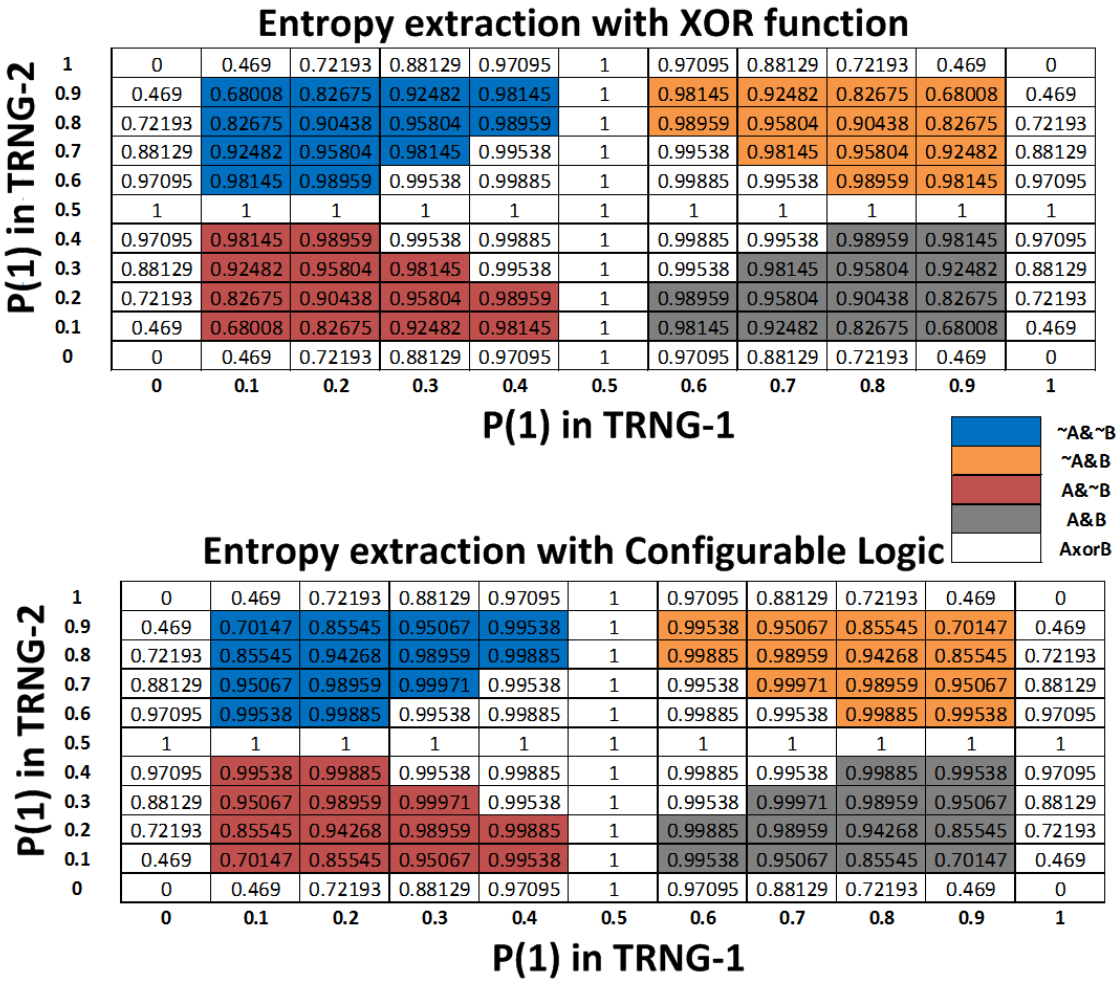


Figure 4.25. Improvement in entropy by using non-XOR functions

the first step, P(1) of TRNG-0 and TRNG-1 are estimated. These values are used in a logic decoder that generates a 4-bit configuration value for the reconfigurable logic block. The logic decoder is designed to generate the configuration value based on the optimum function shown in Figure 4.25 for a given pair of P(1). The reconfigurable logic block, is designed as a 4-input MUX whose select lines are driven by TRNG outputs. In the second round of configuration, output of logic function Fn0 and TRNG-2 are multiplexed into the 6-bit counters. Depending on the probability of ones, the configuration bits for logic function Fn1 is decoded in the logic decoder.

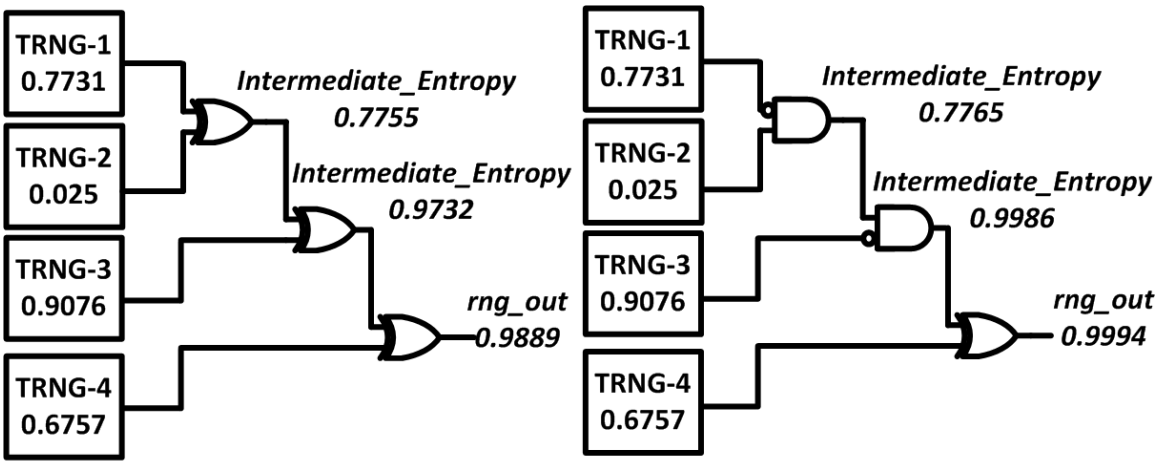


Figure 4.26. Illustration of biased TRNGs with XOR tree and REFLEX

This process is continued till all logic blocks are configured. The configuration time for an n-TRNG system is given by,

$$Configuration\,time = \left(2^{counter\_depth} + 1\right) * (n-1) \tag{4.20}$$

To compare logic selection technique to conventional XOR tree, 10,000 sample chips with four TRNGs were generated. The probabilities of ones at the output of TRNGs were modelled to have uniform distribution of the form uniform[0, max_prob(1)]. The value of max_prob(1) was varied from 0.5 (small bias) to 0.1 (large bias). The comparison of net entropy using XOR function only and reconfigurable logic using REFLEX for max_prob(1)=0.5 is shown in Figure 4.28. REFLEX always provides the same or better entropy extraction compared to XOR tree. Another parameter to compare the two techniques is the difference in probabilities of ones and zeroes. Figure 4.29 shows the entropy and |P(1)-P(0)| comparison for max_prob(1)=0.3. It can be seen that REFLEX provides significant improvement in entropy for large TRNG biases.

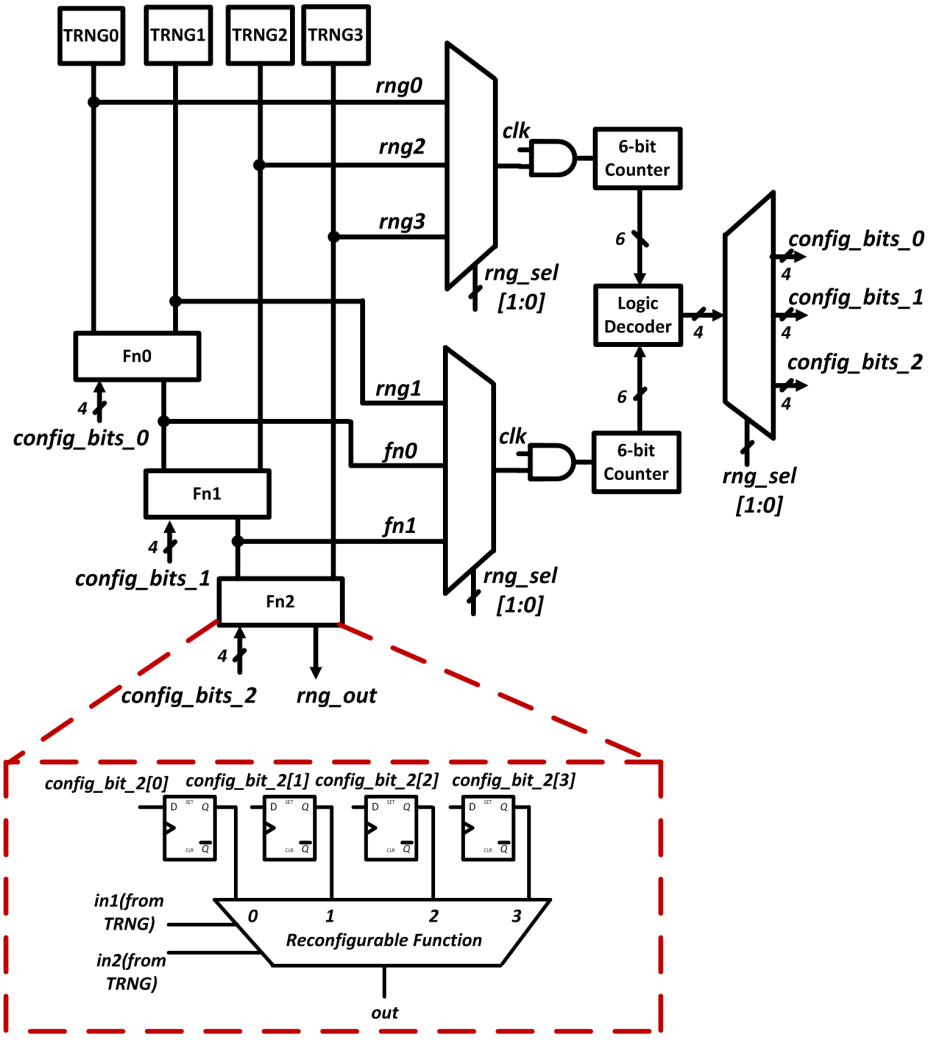


Figure 4.27. Logic Selection Module

Re-order: Configuring logic blocks depending on entropy of TRNGs provides best entropy extraction. However, the order in which TRNGs are paired and their outputs fed into the logic tree also affect the final net entropy, as shown in Figure 4.30. Changing the order of TRNG outputs does not only change the decoded logic in REFLEX, but can also further improve entropy extraction. Although a single unique order is not seen to provide the best entropy, re-ordering the TRNGs in descending order of their entropies provides maximum entropy extraction in most cases. To verify this, 10,000 sample chips with 4-TRNGs were generated with P(1) uniform[0,

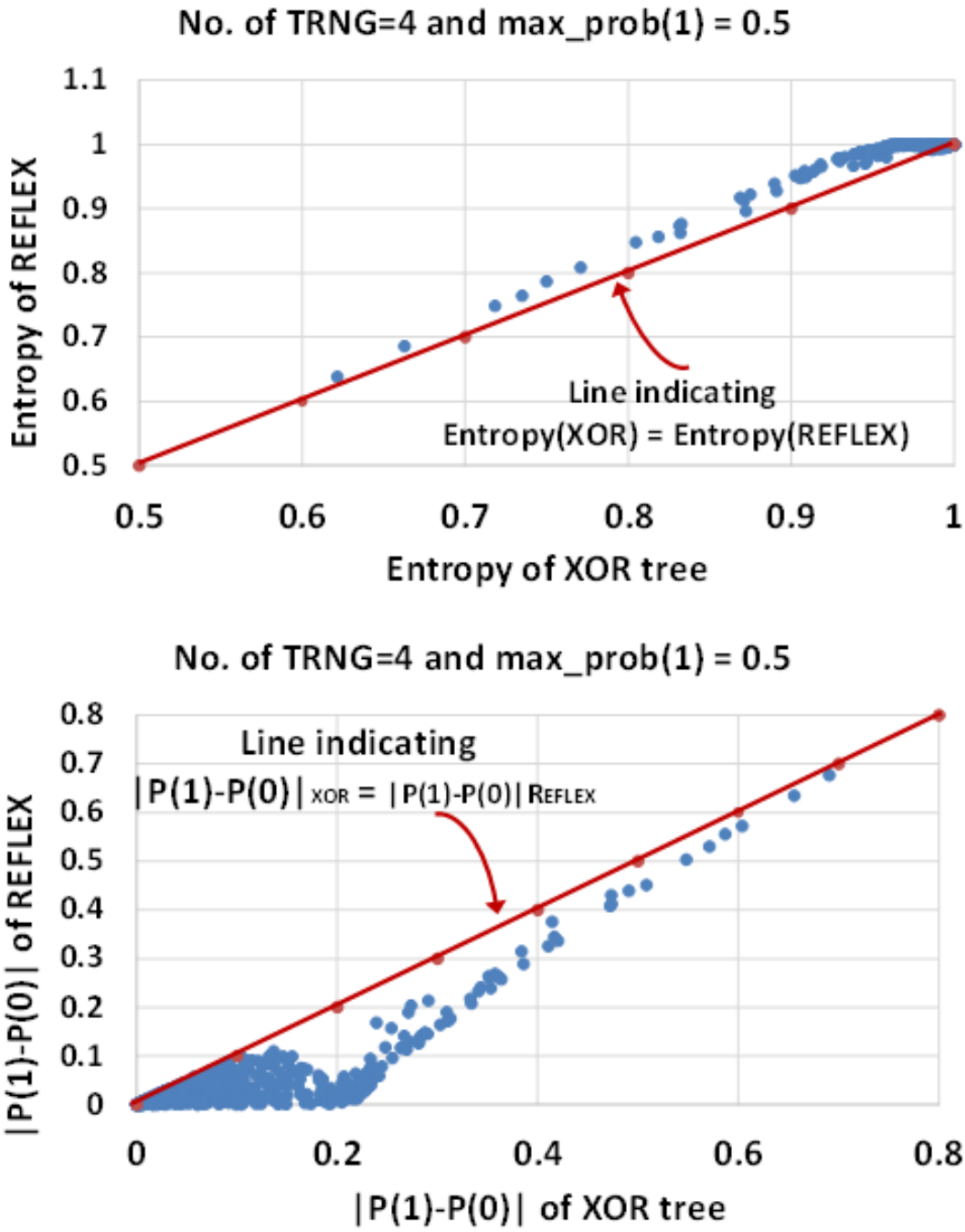


Figure 4.28. Comparison of entropy and |P(1)-P(0)| for max_prob=0.5

max_prob(1)]. The percentage of chips for which re-ordering resulted in better entropy is shown in Table 4.4.

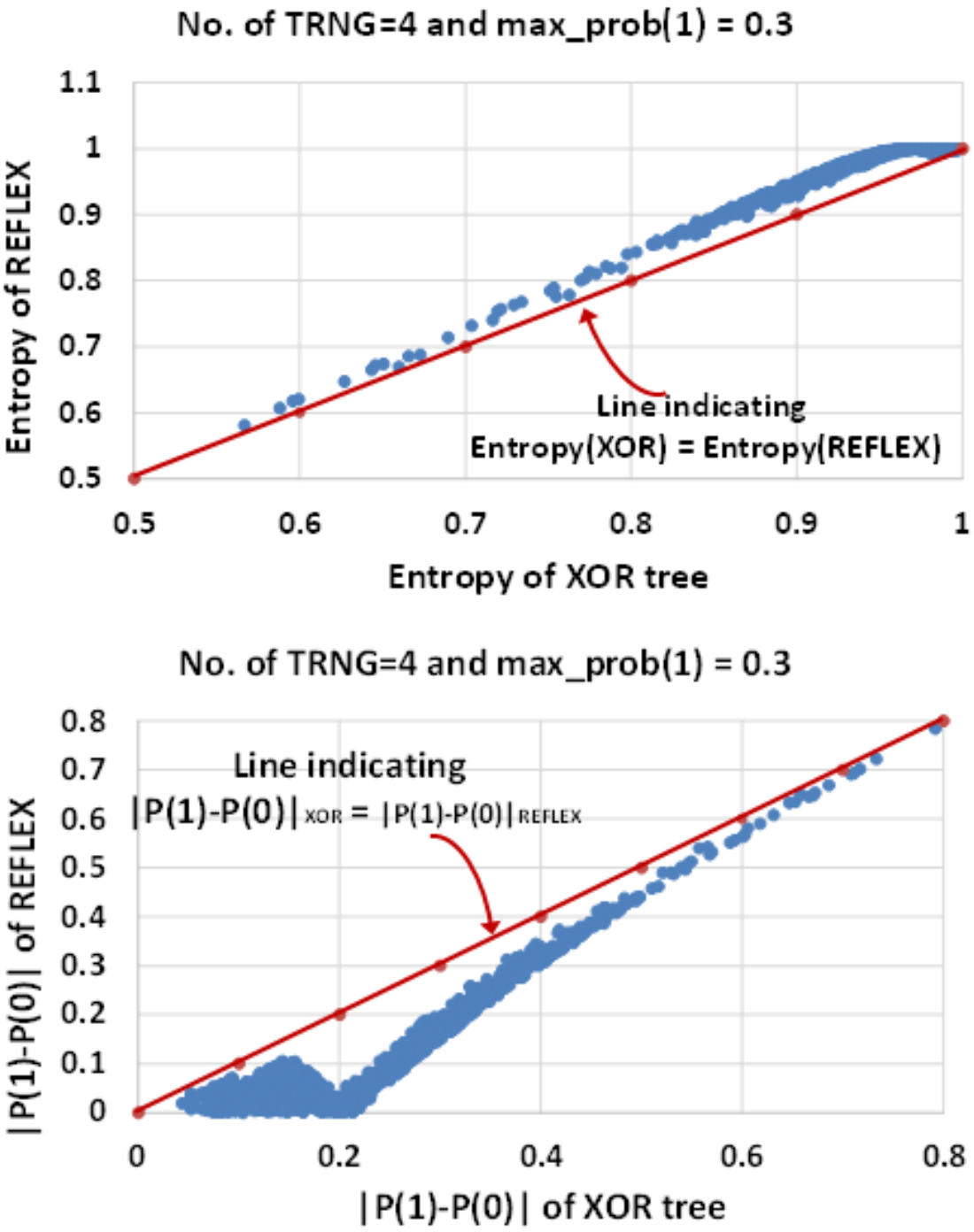


Figure 4.29. Comparison of entropy and |P(1)-P(0)| for max_prob=0.3

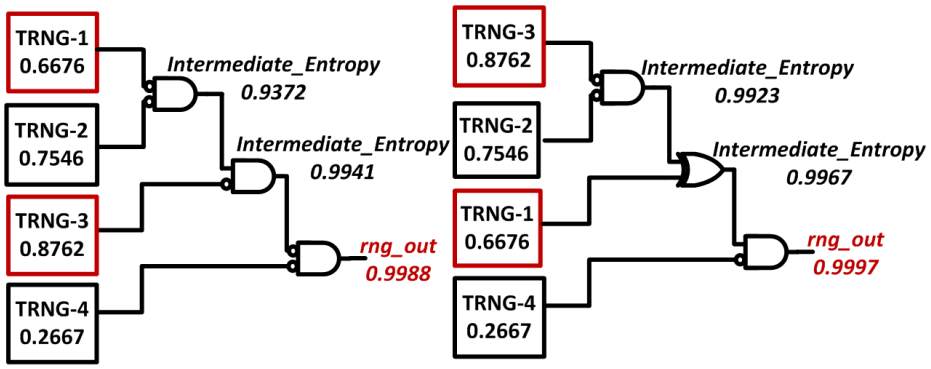


Figure 4.30. Re-ordering TRNGs and impact on net entropy

Table 4.4. Improvement in entropy extraction by re-ordering

No. of TRNG=4

max_prob(1)	Percentage of chips with better entropy when re-ordered
0.5	59.7
0.4	71.4
0.3	82.5
0.2	86.6
0.1	89.2

For smaller bias, only 59% of chips are impacted by TRNG re-ordering. However, as bias increases, re-ordering provides better entropy extraction in 90% of chips. Re-ordering the TRNG inputs into the logic function tree requires additional re-order logic. This increases the overall configuration time and incurs additional area and power overhead. Hence, the re-order logic may be gated for small biases in TRNGs and used only for large bias. The re-order logic, shown in Figure 4.31, multiplexes the output of each TRNG. A 6-bit counter counts the number of ones. At the end of 64 cycles, an offset calculator is used to calculate the offset of probability of ones from the ideal value of 0.5. The TRNG with smallest offset has the highest entropy. The comparator logic, shown in Figure 4.32, re-orders the contents of 2-bit decode registers (in case of 4-TRNG system) in descending order of entropy. If the input offset value is less than the value stored in offset_reg_3, the contents of all offset

registers and decode registers are shifted right. The offset_reg_3 is updated with the new offset value and dec_reg_3 is updated with rng_sel. If offset_val ¿ offset_reg_3, the input offset value is compared with offset_reg_2. At the end of re-order process, the decode registers contain the TRNG numbers in the descending order of entropy. A one-hot demux generates 4-bit configuration for crossbar switch network to re-order the output of TRNGs. The re-order time for an n-TRNG system is given by,

$$Configuration\,time = \left(2^{counter\_depth} * n\right) + n \tag{4.21}$$

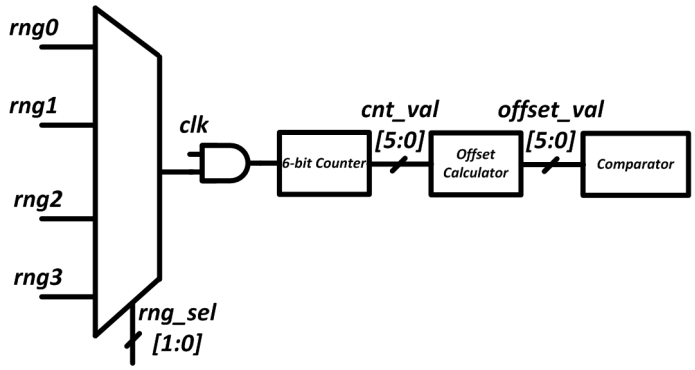


Figure 4.31. Re-order Module

REFLEX: The overall architecture of REFLEX is shown in Figure 4.33.
The re-order block reads the input from each TRNG and generates one-hot switch configuration output. This is used to configure an nxn crossbar switch circuit to re-order the output of TRNGs. The logic select module receives TRNG inputs from the re-order switch circuit and accordingly configures each of the logic functions. The re-order and logic select modules may be run once during power up to configure REFLEX and then power gated to reduce leakage power. REFLEX may also be used to periodically monitor changes in TRNG entropies and modify the configuration logic. Periodic configuration is critical when TRNGs are sensitive to variation in

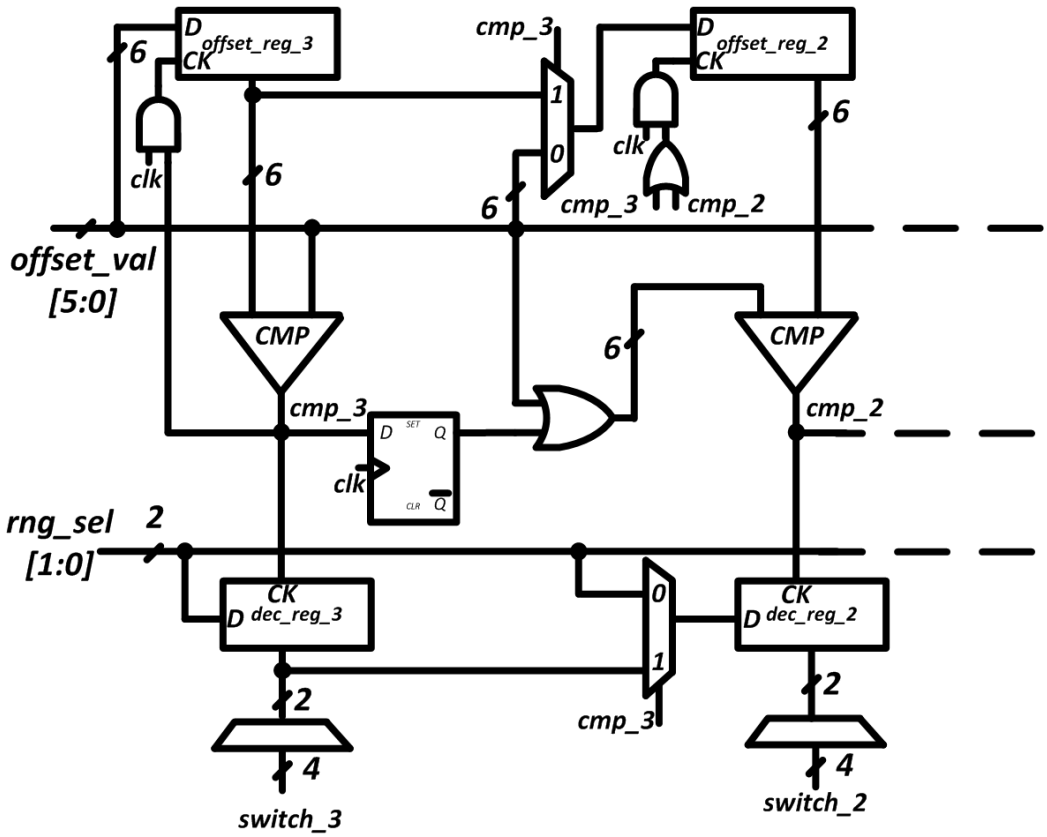


Figure 4.32. Entropy comparator and re-order decode registers

temperature or operating voltage. REFLEX also provides countermeasure against both invasive attack on TRNGs and increase in TRNG bias due to device aging.

Table 4.5. REFLEX configuration time

No. of TRNG	Configuration Time (No. of Cycles)
3	328
4	459
5	590
6	721
7	852
8	983

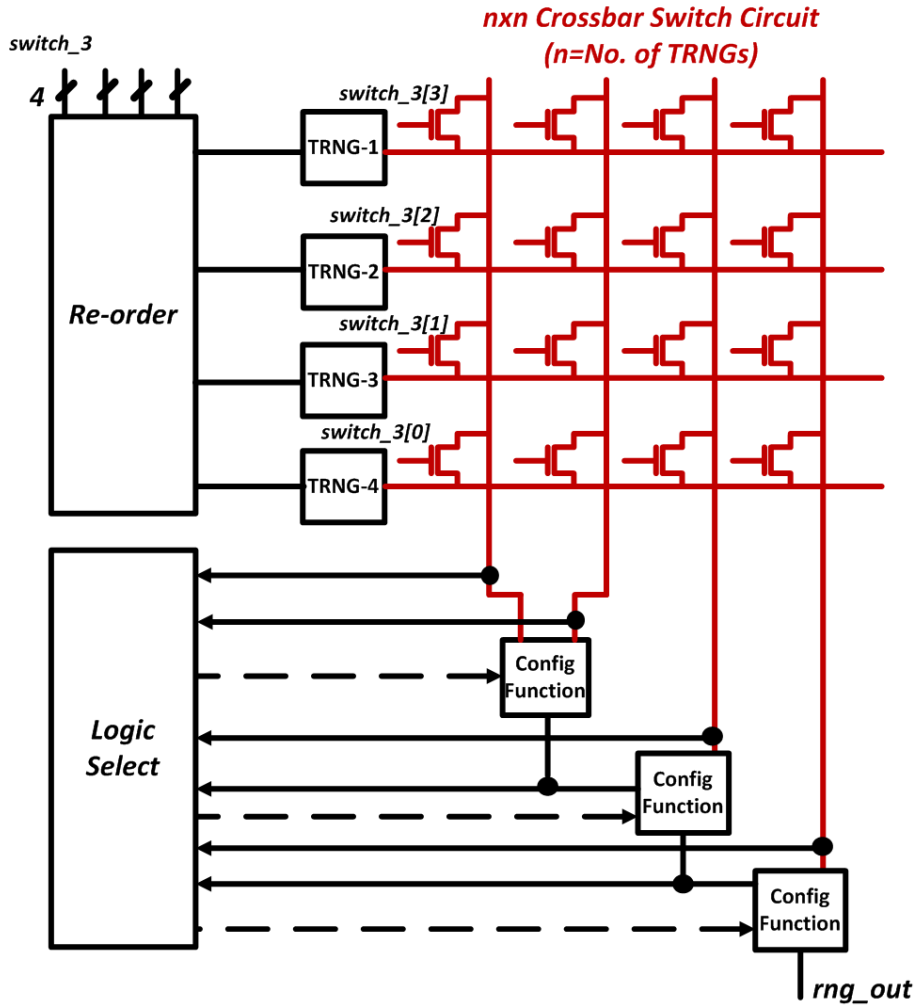


Figure 4.33. REFLEX Architecture

4.2.3 Implementation and Results

The proposed reconfigurable logic was designed and validated in Verilog and synthesized using IBM 32nm SOI standard cell libraries. The synthesized area for re-order, logic select and overall REFLEX circuit is shown in Figure 4.34.

These area numbers do not consider the area of TRNG circuits and assume them to be black box. The areas of comparator module in re-order block and mux/demux logic in logic select block increase with the number of TRNGs used. Although the area overhead of REFLEX is large compared to a handful of XOR gates, it is significantly

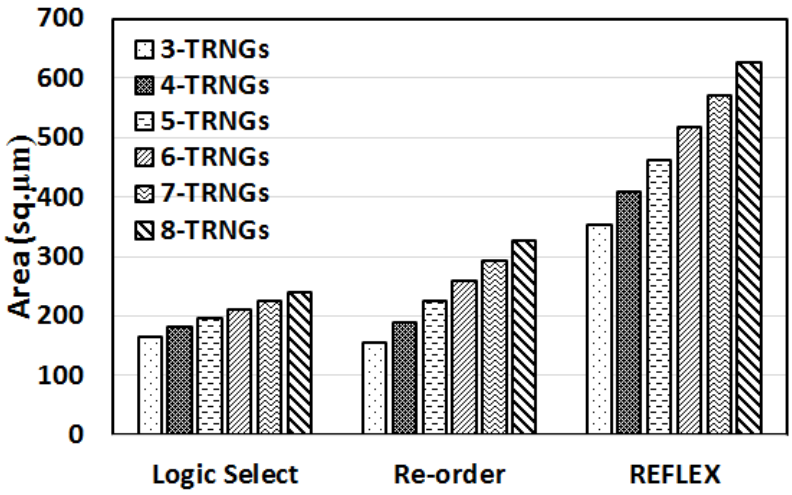


Figure 4.34. REFLEX Synthesized Area

less compared to complex ciphers like AES. The total power (during configuration) and leakage power (during TRNG operation) are shown in Figure 4.35. For an 8-TRNG system with REFLEX configuring at 2GHz, the total power is ∼1.3mW and the leakage power is 70W. Since REFLEX is used periodically, dynamic power is relevant only during configuration. The leakage power during stand-by can be reduced by using higher Vt devices or power gating the REFLEX logic. A more important factor for low power and energy efficient systems is the energy overhead per random bit generated. Figure 4.36 shows the energy overhead for different configuration frequencies.

The energy overhead is amortized as more random bits are generated per REFLEX configuration. For an 8-TRNG system, if REFLEX is configured once every 1000 bits, the energy overhead is 0.9pJ. However, if REFLEX is configured every 1e6 bits, the energy overhead is 0.03pJ; a 30X decrease in energy overhead.

REFLEX was validated by simulating 10,000 chips with number of TRNGs varying from three to eight and P(1)∼uniform[0, max_prob(1)] with max_prob(1) varying from 0.1 to 0.5. Figure 4.37 shows the percentage of chips where REFLEX provides better

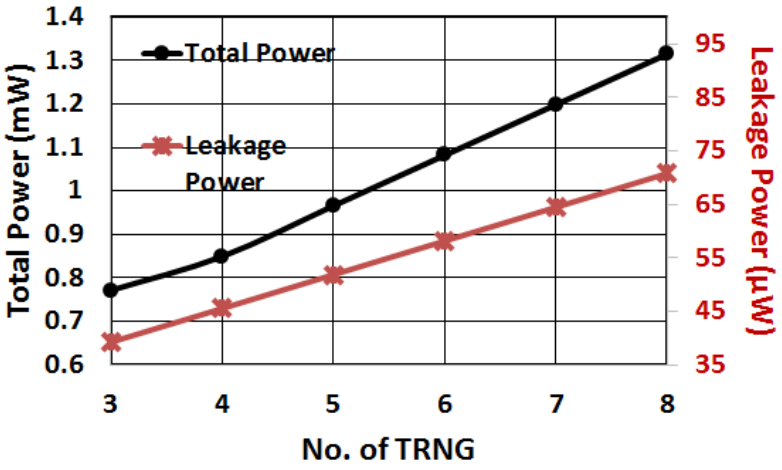


Figure 4.35. REFLEX Total and Leakage Power

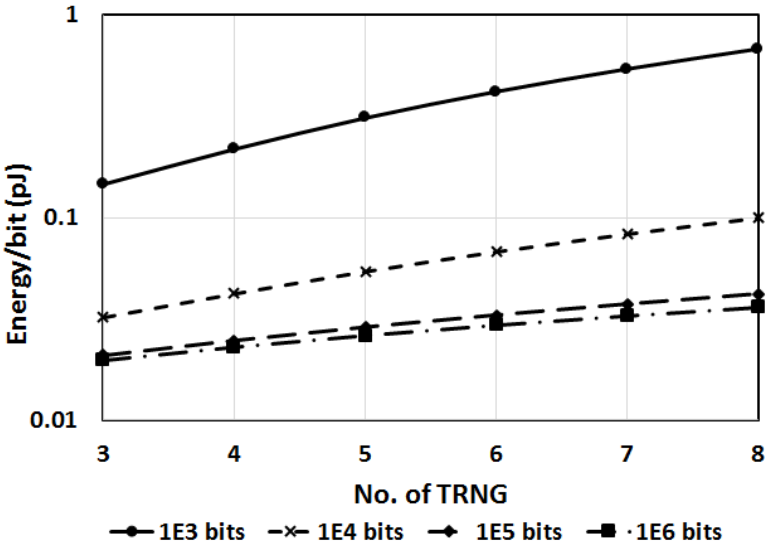


Figure 4.36. REFLEX Energy Overhead

entropy extraction than XOR tree. At lower bias points, ∼75% of chips provide better entropy with REFLEX. However, for large biases, REFLEX provides better net entropy compared to XOR tree. Figure 4.38 shows the average bit entropy for max_prob(1)=0.3.

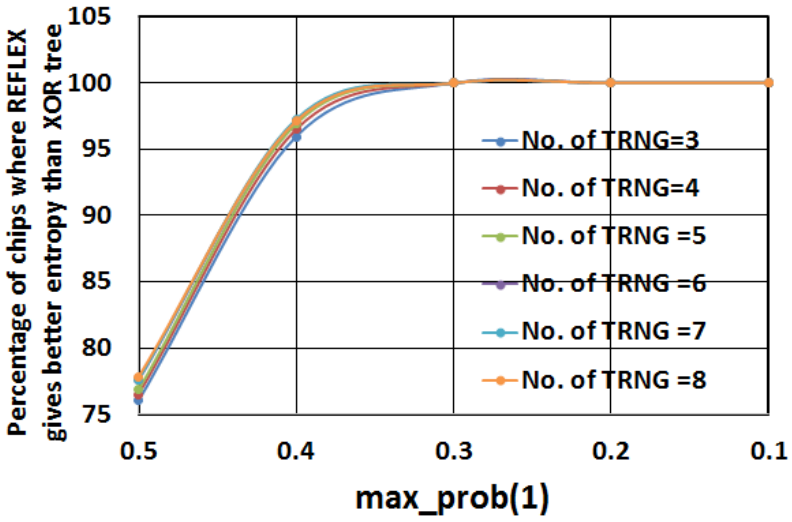


Figure 4.37. Impact of REFLEX on chips with biased TRNGs

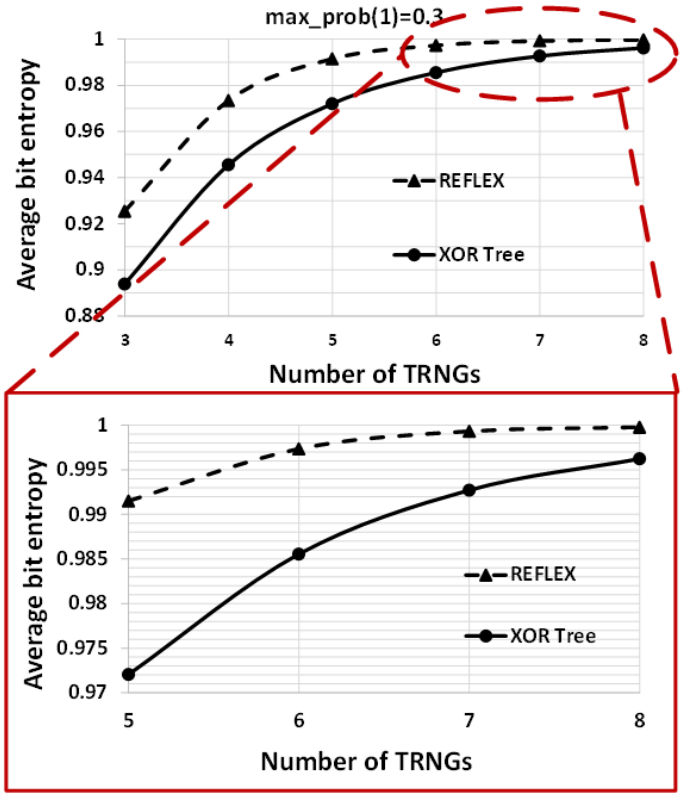


Figure 4.38. Comparison of Average Entropy

Table 4.6. Result of NIST statistical tests

	XOR				**REFLEX**			
	max_prob(1)				**max_prob(1)**			
No. of TRNG	**0.2**	**0.3**	**0.4**	**0.5**	**0.2**	**0.3**	**0.4**	**0.5**
5	0	0	7.9	54.6	7.7	43.8	65.7	84.7
6	0	0	21.5	71.2	26.2	65.9	85.4	94.9
7	0	0.81	39.1	83.5	50.3	82.8	94.7	98.6
8	0	3.7	59.6	91.6	71.8	92.9	98.6	

An improvement in entropy from 0.99 using XOR function to 0.999 using REFLEX may not appear to be significant, however, it has a major impact on statistical tests for randomness. Table 4.6 shows the percentage of chips that pass NIST statistical tests for different number of TRNGs and max_prob(1). REFLEX provides 9% to 55% improvement in randomness even in the least bias corner.

CHAPTER 5

SENSING VARIATION

Variation tolerant circuits can be best designed by understanding and sensing the nature of variations. This can be achieved by designing test circuits on the chip to detect die-to-die variations and with-in die variations, which can then be incorporated in device models used for simulations and analysis. However, circuits which use adaptive techniques to correct the impact of variation require on-chip sensing circuits to estimate the impact of variations and take appropriate corrective action. In the first section of this chapter, we present an on-chip NIST statistical test suite implementation [80] to detect bias in TRNGs due to variations [96]. The bias may be due to process variation, variation in operating conditions or aging effects during the chip lifetime. The on-chip test design can be incorporated with any kind of TRNG circuit. In the second section, we present an area and power efficient wear-out sensing circuit to detect V_t shift in devices due to aging effects [102]. Apart from process variation, constant aging effects also affect digital circuits. Since these effects depend on the chip workload, it is critical to sense the variation and perform dynamic scaling in operating frequency and voltage. Sensing variations facilitates on-the-fly appropriate corrective action. This provides necessary compensation for each chip eliminating the need for large guard-bands during design. Apart from variations, on-chip noise also affects circuit performance and reliability. One of the most important requirements to account for noise during circuit design is accurate characterization of the noise source. Random Telegraph Noise (RTN) is emerging to be a prominent source of noise in nano-CMOS circuits, specifically memory elements. It is imperative to

design on-chip sensors to sense and characterize RTN [15]. In final section of this chapter, we present a novel sensor that can be used to characterize the magnitude and time constants of RTN.

5.1 Sensing Variations using On-chip Statistical Tests

The conventional metrics for evaluating an analog/digital circuit are area, power and performance. However, RNG circuits need a fourth metric which measures the degree of randomness. Process variation is one of the main sources of bias in TRNG circuits. The most basic metric of randomness is the Shannon bit entropy which measures the proportions of bit 0's and 1's in a sequence to give an entropy value in the range of 0 to 1. However, equal proportions of 0s and 1s do not validate all possible weaknesses of an RNG. Sophisticated statistical tests are required to quantify the various aspects of randomness like correlation, run length and random distribution of bits. The most popularly used test suites are the RNG test suite by American National Institute of Standards and Technology (NIST) [80] and the DIEHARD Tests [63]. These test suits evaluate large sequences of bit streams for a pre-defined null hypothesis to predict the randomness of the source. With advancing CMOS technologies, transistors are increasingly sensitive to dynamic variation in operating temperature and power supply noise. Further, time dependent wear-out like Negative Bias Temperature Instability (NBTI) and Hot Carrier Injection (HCI) also affect circuit performance. A one-time statistical test performed during the chip testing phase does not provide constant monitoring of circuit behavior. A good TRNG could degrade intermittently due to temperature or voltage variation and permanently degrade over time due to wear out. An on-chip test module can be used to continuously monitor the output of RNG. The cryptographic system using the RNG can make an informed decision about the randomness of inputs in run-time. TRNG in Intels Ivy Bridge Microprocessor incorporates a health check by counting

empirically arrived bit patterns [29]. The health check only detects extreme cases of bias with Shannon entropy less than 0.5. The TRNG module depends on AES based conditioning for entropy improvement. In [34], a complete NIST test suite implementation has been proposed for real-time statistical test. However, such an implementation adds tremendous area overhead to the system. The authors indicate that only 2 NIST tests could be ported completely on a Xilinx Virtex II Pro FPGA V2P30. The test suite implementation itself consumes a large area of the order of 1000s of flip flops.

Since RNG could serve as a single point of failure, they provide an avenue for malicious attack on cryptographic systems. In [42], J. Kelsy, et. al discuss in detail the various cryptanalytic attacks on PRNG. A practical attack on ring oscillator based TRNG is demonstrated in [62]. The authors present a real attack on EMV card using frequency injection. The randomness of the TRNG is compromised and key space of the secure micro controller is reduced from 264 to 3300. Other practical attacks have also been reported in literature, like breaking weak RNG in MIFARE card [41], eavesdropping on a weak PRNG on EPC Gen2 compliant RFID tag [68] and contactless electromagnetic attack on ring oscillator based TRNG [5]. Commercial microprocessor manufacturers use on-chip sensors (temperature, voltage) to detect invasive attacks. An on-chip NIST module can detect variation in randomness, which is the symptom of an attack, thereby protecting the device against all sources of attack. The randomness information can be used by the secure module (Crypto module/ Micro controller) to take necessary evasive action, like a Denial of Service to the attacker and thereby prevent economic and personal identity loss. In a shared key protocol for secure communication, the system receiving a key can use on-chip NIST module to get a measure of the randomness. Based on the NIST test results, the received key can be used or a request can be made for a new key. All systems using a TRNG employ some form of calibration or post-processing to mitigate the

effect of physical variation on the statistics of RNG. The post-processing techniques could vary from very lightweight XOR function or von Neumann corrector [97], to robust entropy extraction using HASH functions or AES [29]. While this is a fault tolerant approach to entropy extraction, an on-chip NIST module allows multiple post-processing units to be implemented and then select the appropriate one based on the quality of the TRNG. The other blocks can be powered OFF to reduce leakage power.

5.1.1 Reduced NIST Test Suite

The NIST test suits perform an exhaustive statistical analysis and hence are computation intensive. In [82], approximations in the test suite are proposed for a computation friendly implementation. Byte-wise implementation [93] and parallelizing the tests [94] further improve run time of NIST test suite. A test methodology for TRNG circuits in compliance with NIST standards is proposed in [115], while a run-time hardware implementation for NIST tests on FPGA is proposed in [34]. We propose a lightweight implementation of 6 statistical tests from the NIST test suite. The tests are chosen based on the minimum sample set recommended by NIST and the complexity of storage and computation in terms of ultra-low power implementation. Each of the 6 tests is reduced to obtain a hardware design friendly structure. However, the tests comply with the NIST standard and do not result in a deviation in accuracy of the results. The reduced implementations do not require any complex arithmetic or statistical computation and result in a very low area and power overhead. Although the reduced test set is not as exhaustive as using the entire NIST test suite, it provides an effective layer of security and monitoring for lightweight RNG implementations.

The NIST test suite consists of a collection of fifteen statistical tests. Each of these tests hypothetically quantifies a certain aspect of randomness. Given a bit sequence a of length n, an assumption of randomness, defined as null hypothesis H(0), is fixed. A

counter assumption is defined as alternate hypothesis H(a) along with a significance level alpha (). The value of , also called critical value, is fixed apriori in the range of [0.001, 0.01]. The bit sequence from an RNG is analyzed using a specific data processing technique for each test. The output of the data processing stage, variable X, is used as one of the inputs to either Complimentary Error Function (erfc) or the Upper Incomplete Gamma Function (igamc) depending on the test. The output of this stage is called the P-value. This value is compared with the constant significance level α. If the P-value is greater or equal to α, the null hypothesis is accepted with a confidence of $1 - \alpha$, else is rejected with a confidence of $1 - \alpha$, Figure 5.1.

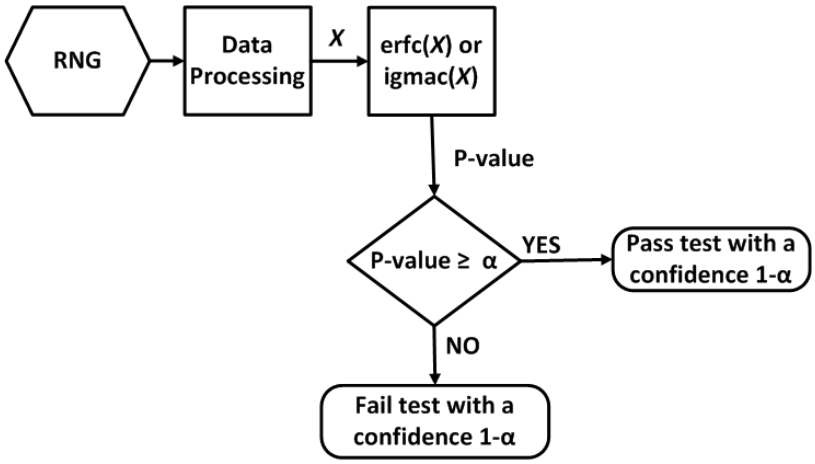


Figure 5.1. Generic Flow of NIST Tests

A summary of the tests in the NIST suite with the null hypothesis for each test is shown in Table 5.1. Each test in the NIST suite requires a pre-defined minimum sample set of size ranging from 100 bits to 100,000 bits. The size of hardware for computation and data storage required for a test is proportional to the size of the data sample. Since the focus of this work is to realize a test module for ultra-lightweight applications, we choose six of the fifteen NIST tests which are viable for lightweight implementation. The reduced set of NIST tests consist of Frequency Monobit Test, Frequency Block Test, Runs Test, Test for Longest Runs of Ones, Binary Matrix

Rank Test and Non Overlapping Template Matching. The reduced set includes the four tests mandated by FIPS 140-1 standard.

Table 5.1. Summary of NIST SP 800-22 test Suite

Test	**Null Hypothesis**
Frequency Monobit Test	Good proportion of 1s and 0s
Frequency Block Test	Good proportion of 1s and 0s within N blocks of M-bit
Runs test	No clusters of 1s and/or 0s in a sequence.
Test for the Longest Run of 1s in a Block	No clusters of 1s and/or 0s in N blocks of M-bit
Binary Matrix Rank Test	Low linear dependence between sub-strings, sub-matrices with high rank
Discrete Fourier Transform (Spectral) Test	Balanced amount of peaks in the frequency domain
Non Overlapping Template Matching Test	Not too many non-overlapping equal sequences
Overlapping Template Matching Test	Not too many overlapping equal sequences
Maurer's Universal Statistics Test	Sequence cannot be significantly compressed
Linear Complexity Test	Sequence with long LFSRs
Serial Test	Every m-bit template has equal probability to arise
Approximate Entropy Test	Non-regular occurrence of the same overlapping template
Cumulative Sums Test	Cumulative sum excursion near zero
Random Excursion Test	Random distribution of visits among cycles to eight states
Random Excursion Variant Test	Random distribution of visits among cycles to eighteen states

As described, the data processing stage of NIST tests consists of either the Complementary Error Function (erfc) or the Upper Incomplete Gamma Function (igmac). These two functions are the bottleneck in computing the P-value for each test. For a given sample size n, the test is considered to be passed if the P-value is greater or equal to target critical value . To reduce the computation required, we fix the value of n and compute the range of input X to the erfc or igmac which lead to a P-value greater than the critical value of . This completely eliminates the need for complex

computation of error function or gamma function, thereby significantly reducing the complexity of hardware required to implement the tests. In the proposed lightweight implementation, all computations and test results are quantified in terms of the value of X instead of the P-value. Since the value of X and the corresponding P-value have a one-to-one mapping, the accuracy of the results is not lost. The partial reconfigurable feature allows the user to set the critical value depending on the need of the application to make the tests more stringent. Modifying the critical value only varies the bounds for the input to the complex function. The bounds can be set one-time by blowing fuses or can be configured using registers to store the values. Further, all computations are performed serially on the incoming bits from the RNG. This eliminates the need for additional storage devices except for byte-wide shift registers.

5.1.2 Lightweight Implementation of NIST Randomness Test Suite

In this section a detailed description of the lightweight implementation for Frequency Block Test (FBT) is presented. The implementation of the other five tests is also based on similar techniques. For a detailed statistical analysis of each test, interested readers may refer to [80].

The FBT checks for a good proportion of 1s and 0s in blocks of random bits. Given a sample bit stream of length n, the test breaks the bit stream into N blocks of size M bits. For each block the distance of 1s proportion (or 0s) over M-bit from 0.5, the expected mean value is computed. The partial results are squared and accumulated; the resulting number is multiplied by a constant and used as one input to igmac. The gamma function returns the P-value that is compared with the predefined significance level, α. If the P-value is greater or equal to , the bit stream is predicted to be random with a confidence of $1 - \alpha$. The above steps can be reduced to accommodate a lightweight hardware implementation as follows:

$$Let\ n = 128\ (Total\ number\ of\ bits)\ and\ M = 8\ (Number\ of\ bits\ per\ block)$$

$$N = \frac{n}{M} = 16,\ the\ number\ of\ non-overlapping\ blocks$$

$$Assuming\ \alpha = 0.01,\ the\ test\ passes\ for\ P-value \geq 0.01,$$

$$In\ the\ Frequency\ Block\ Test,\ P-value = igmac\left(\frac{N}{2}, \frac{\chi^2}{2}\right)$$

$$Hence,\ igmac\left(\frac{N}{2}, \frac{\chi^2}{2}\right) \geq 0.01\ where\ \chi^2\ is\ the\ output\ of\ data\ processing\ stage$$

$$Computing\ the\ inverse\ of\ igmac(),\ \chi^2 \leq 16$$

$$In\ the\ Frequency\ Block\ Test,\ X = \chi^2 = 4M\ \sum_{i=1}^{N}\left(\pi_i - \frac{1}{2}\right)^2$$

$$Hence,\ 4M\ \sum_{i=1}^{N}\left(\pi_i - \frac{1}{2}\right)^2 \leq 16,\ where\ \pi_i\ is\ the\ ratio\ of\ 1's\ in\ each\ of\ the\ N\ blocks$$

If a counter c is used to count the number of 1s in each block,

$$4M\ \sum_{i=1}^{N}\left(\frac{c}{8} - \frac{1}{2}\right)^2 \leq 16$$

$$\sum_{i=1}^{N}(c-4)^2 \leq 32 \tag{5.1}$$

The complex computation to estimate the P-value and hence deem a bit stream to pass/fail the Frequency Block Test can be reduced to a series of counter, offset calculation and squaring operations. Since the number of bits in the sample, n and the block size M are positive whole numbers; all computations involve whole numbers allowing a simpler combinatorial logic based implementation. A similar calculation is performed for the other five NIST tests to reduce the computation and design a lightweight implementation. Further, the NIST module can be turned ON intermittently to reduce power overhead. By calculating the bounds of input to igmac function, the FBT is optimized to a series of count, accumulate and compare operations as shown in Figure 5.2. The complex igmac function is avoided enabling a

lightweight implementation. The hardware implementation for each stage of FBT is as show in Figure 5.3.

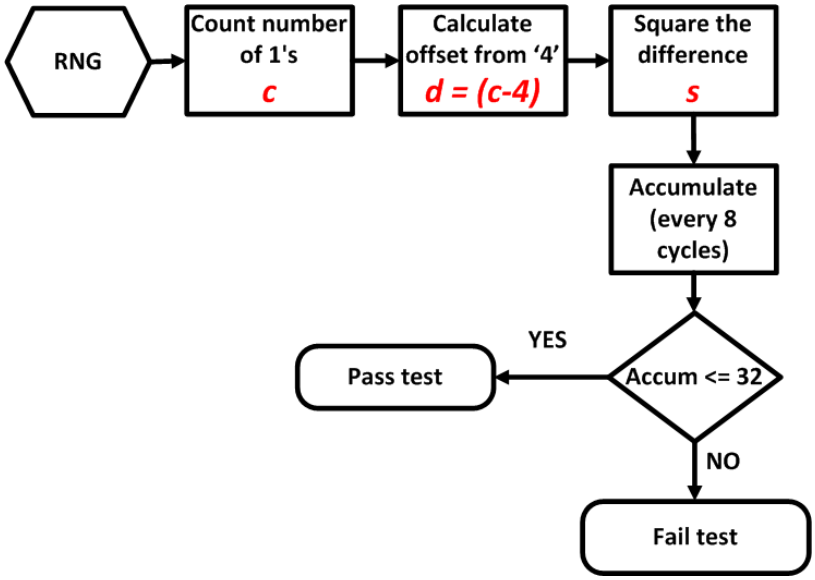


Figure 5.2. Flow of optimized Frequency Block Test

Similar to Frequency Block Test, other NIST tests are also reduced to facilitate optimum hardware implementation. All the tests operate on each incoming bit from the RNG serially, thereby minimizing additional hardware to store the bits. Only for the Non Overlapping Template Matching test and the Binary Matrix Rank test, a 10 bit and 8 bit shift registers are used respectively to store the previous bits. The counters and control logic is shared across different tests to further optimize area and power. The partial reconfigurable feature of the reduced NIST test module provides the flexibility of choosing a different critical value for each test based on the requirement of the platform/application. The reconfigurable bound registers in each test module allows appropriate value of to be set as the threshold critical value for test pass/fail.

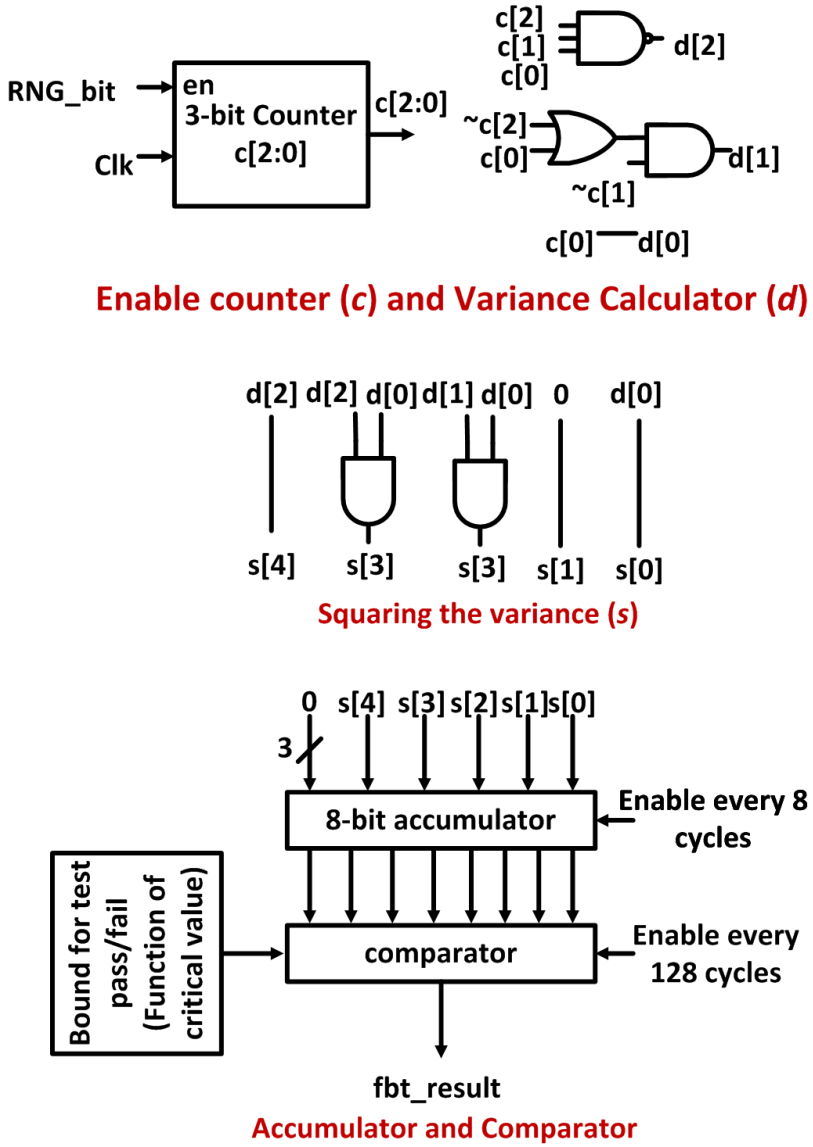


Figure 5.3. Digital logic for lightweight FBT implementation

5.1.3 Implementation and Results

The proposed lightweight implementation for the reduced NIST test suite consisting of six tests was designed in Verilog and verified for functionality using ModelSim. The designs were synthesized in Synopsys Design Compiler using the 45nm SOI NCSU/OSU Open Source Standard Cell Library. The synthesized designs were

optimized for a cycle time of 0.5ns (2GHz). The area and power numbers for each 128 bit test are as shown in Figure 5.4 and Figure 5.5 respectively.

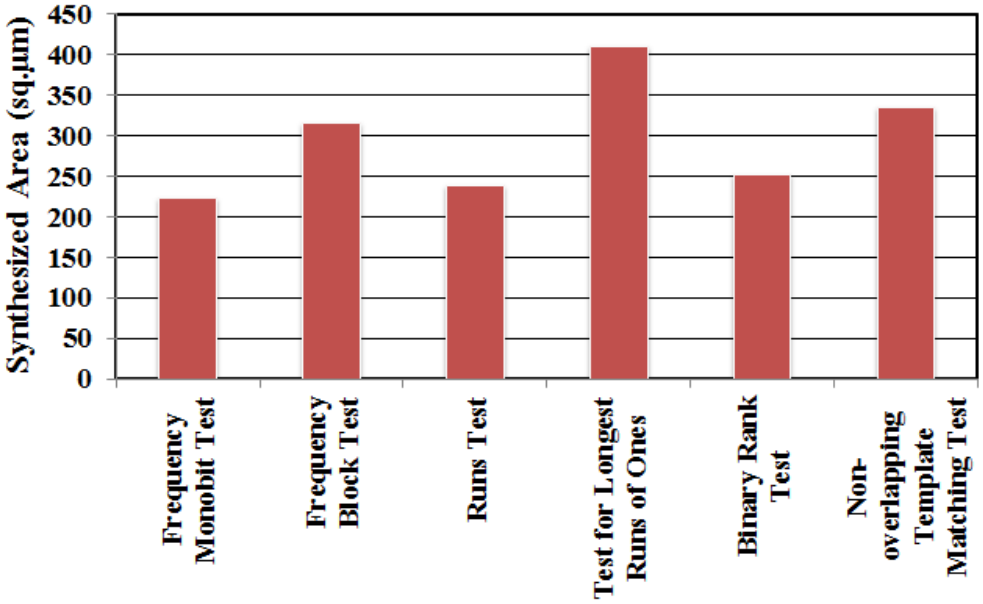


Figure 5.4. Synthesized area of lightweight NIST test implementation

The lightweight implementation results in a synthesized area ranging from $240\mu m^2$ to $460\mu m^2$ for each test. The shared control logic and counters reduce the overall implementation area. The common counter and control logic consume an area of around $200\mu m^2$ resulting in the overall NIST module area of $1926\mu m^2$. This translates to 1026 NAND gates equivalent in 45nm technology. The active power for each test is of the order of 0.4mW to 0.8mW. All the tests are designed to operate in parallel, resulting in an overall active power of 3.75mW for the NIST module operating at 2GHz. The overall cell leakage is ∼10.5W which is 0.28% of the total power. Since the target applications include passively powered and battery operated devices like RFIDs and smart cards, energy/bit is an important metric. The 128-bit reduced NIST module operating on 2Gbps consumes 1.87 pJ/bit.

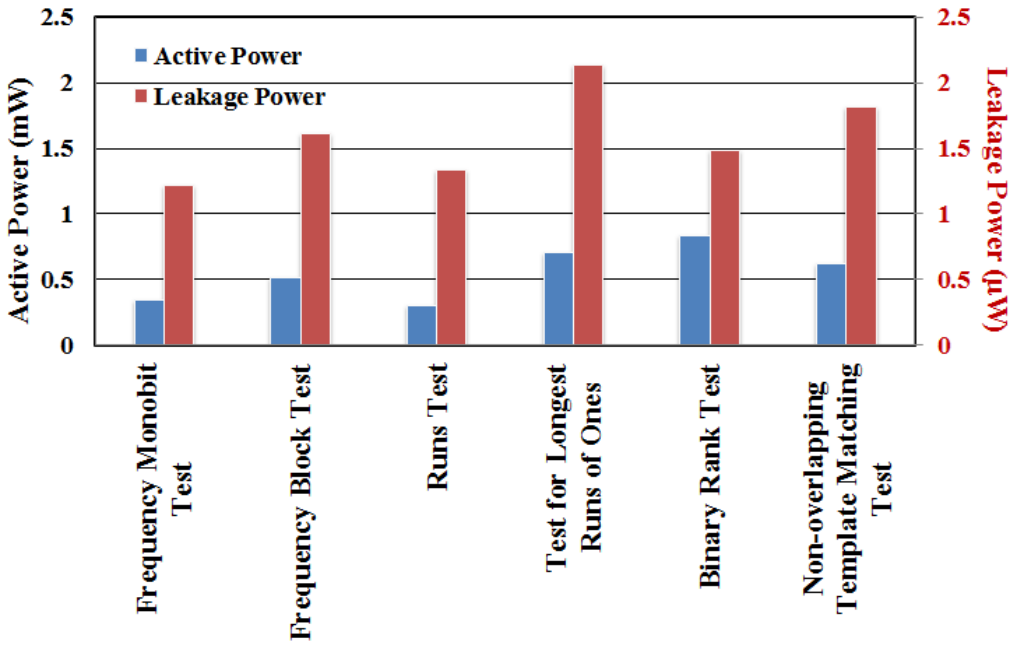


Figure 5.5. Active and leakage power of lightweight NIST test implementation

Table 5.2. Area, Power and Energy/bit of 256bit and 512bit NIST implementations

Bit length	Area (m2)	Power (mW)	Energy/bit @ 2Gbps (pJ/bit)
256 bits	2394	4.03	2.01
512 bits	2787	4.37	2.18

The proposed implementation is scalable to larger number of bit samples as well. Depending on the number of bits n and block sizes, the bounds for each test varies. The Binary Matrix Rank Test and Non Overlapping Template Test are implemented with a minimum bit sample size greater than 512 bits. The optimized tests can be scaled for larger bit sequence range at the cost of increased area and power. The area, power and energy/bit for 256-bit and 512-bit implementations are as shown in Table 5.2.

5.2 Fine Grained wear-out Sensing Using Metastability Resolution Time

As device sizes in nanometer CMOS are getting smaller, various short channel effects like Negative Bias Temperature Instability (NBTI) [83, 89], Hot Carrier Injection (HCI) and Time Dependent Dielectric Breakdown (TDDB) have posed serious reliability concerns. NBTI in particular manifests in the form of threshold voltage (Vth) shift and hence degradation in performance of PMOS devices over the lifetime of a chip [121]. A similar phenomenon, Positive Bias Temperature Instability (PBTI) degrades performance of NMOS devices. In memory devices like SRAM, NBTI/PBTI affect stability leading to increased bit error rate (BER)over time [124]. The impact of performance degradation can be alleviated by providing enough design margin for critical paths in the design. However, this would incur significant performance penalty in the early lifetime of the system. In [122], Wang, et. al propose statistical metrics to identify aging critical devices or gates in the design to develop realistic guard bands. A more efficient technique to counter device aging is run time monitoring of device performance. The impact of NBTI/PBTI depends on the stress and recovery time experienced by devices. This is a direct consequence of workload or signal activity in the design. BTI effects are further accelerated by high operating temperatures. In [70], Mintarno, et.al propose a runtime self-tuning technique to increase chip lifetime in the presence of device aging. The proposed technique uses runtime device aging information and correspondingly moderates system performance by Dynamic Voltage and Frequency Scaling (DVFS). Adaptive circuit tuning provides a more optimal performance-lifetime trade-off compared to one time guard band during the design phase. Adaptive tuning techniques require constant monitoring of device wear-out.

5.2.1 NBTI/PBTI and wear-out Sensing

NBTI/PBTI: NBTI and PBTI phenomenon have been widely studied in the last decade. NBTI is described using the Reaction-Diffusion model (R-D). In the reaction phase, interface charges are induced in the Si-SiO2 boundary increasing the V_t of the device. In general this is termed as the stress mode. During diffusion, some of these reaction generated bonds diffuse away, leading to reduction in V_t. This is known as the recovery mode. The variation in Vth during the stress and recovery periods are modeled as,

$$\Delta V_t = \begin{cases} \left(K_v \left(t - t_0\right)^{0.5} + \sqrt[2n]{\triangle Vth_0}\right)^{2n} & \text{stress} \\ Vth_0 \left(1 - \dfrac{2\xi_1 t_e + \sqrt{\xi_2 C \left(t - t_1\right)}}{2t_{ox} + \sqrt{Ct}}\right) & \text{recovery} \end{cases}$$

More details of NBTI modeling can be found in [121]. From a design perspective, Vth variation with aging is mainly governed by stress and recovery periods of devices. This depends on signal activity or workload. Higher operating temperature accelerates BTI process, resulting in larger variation in Vth. An ideal wear-out sensor should not have any initial Vth offset due to process variation. A sensor with initial V_t offset either overestimates V_t shift due to NBTI/PBTI resulting in performance degradation, or underestimates Vth shift resulting in reliability issues. Further, the sensor should be tolerant to variation in operating temperature and supply voltage during the measurement phase.

Existing wear-out Sensors: One of the most popular techniques for wear-out sensing is using on-chip ring-oscillators and monitoring its frequency through the life-time of the chip. In [46], a variation of RO based NBTI sensor is proposed using two 105-stage ring oscillators. Although this technique provides extremely fine grained wear-out sensing, the number of stages in RO could be a limiting factor in embedding multiple sensors in a design. Large spatial variation in thermal profiles and advancement in

3D IC technology will require more sensors per silicon area. Another major issue with scaling RO based circuits in 45nm and below technologies is intra-die variation among RO devices. At t=0, the reference RO and tracking RO should ideally have a precise frequency difference to give maximum resolution. In [46], configurable capacitors are used in ROs to tune the frequencies, called frequency trimming. In nanometer CMOS, the intra-die variation increases requiring large range and higher precision of frequency trimming along with additional control logic to configure capacitances. Variation in temperature and power supply noise can significantly impact the counter values before and after stress, thereby introducing error in measured Vth values. The long measurement times of RO based sensors also increases the recovery time for stressed devices resulting in loss of tracked data. In [86], a precise NBTI tracking circuit is proposed using a hybrid combination of RO based sensor and delay chain based sensor. The above sensor incorporates the advantage of measurement speed in a delay line based sensor with smaller area of RO based sensor. The precision of measurement is compromised for faster measurement and smaller area. A measurement time of 70ns and precision of 0.2% change in RO frequency provides an attractive wear-out sensing technique. However, the design area of 7242m2 in 90nm technology is still a significant overhead to incorporate a large number of sensors in the design. Further, impact of Vth offset in ROs due to fabrication, impact of temperature and supply noise during measurement are major limiting factors.

While techniques such as the ones proposed in [46] and [86] target high precision wear-out sensing, [125] implements small area sensors using metastable circuit. The fundamental sensor circuit consists of a pair of cross coupled inverters. The PMOS device in degradation inverter is sized X% larger than the reference inverter. During tracking mode, a stress signal is applied to the degradation PMOS. This increases its threshold voltage, reducing the mismatch between degradation and reference inverters. An ideal metastable point is reached when the threshold voltage of degradation

inverter has increased large enough to have Id(deg) = Id(ref), indicating X% degradation due to NBTI. The circuit only provides a binary information of whether device wear-out is ¡= X%. Multiple sensors need to be designed for fine grained measurement. A major concern for using resolution state is initial process variation induced Vth offset between reference and degradation PMOS devices. The authors address this issue by using a large number of sensors and performing a majority voting. For a measurement accuracy of >90%, a sensing area (No. of sensors * Area of each sensor) is shown to be $7600\mu m^2$ in 150nm technology. This is still a large area considering that this sensing area provides detection of one single degradation value. A fine grained degradation sensing of N steps will require $N(sensing\_area/step)$. This increases the complexity of placing sensors throughout the die and creating a sensor NoC for polling.

5.2.2 Metastability Resolution Time based wear-out Sensor

The proposed metastability resolution time based wear-out sensor is as shown in Figure 5.6. It consists of a metastable cell with a pair of cross coupled inverters and a Time to Digital Converter to measure resolution time.

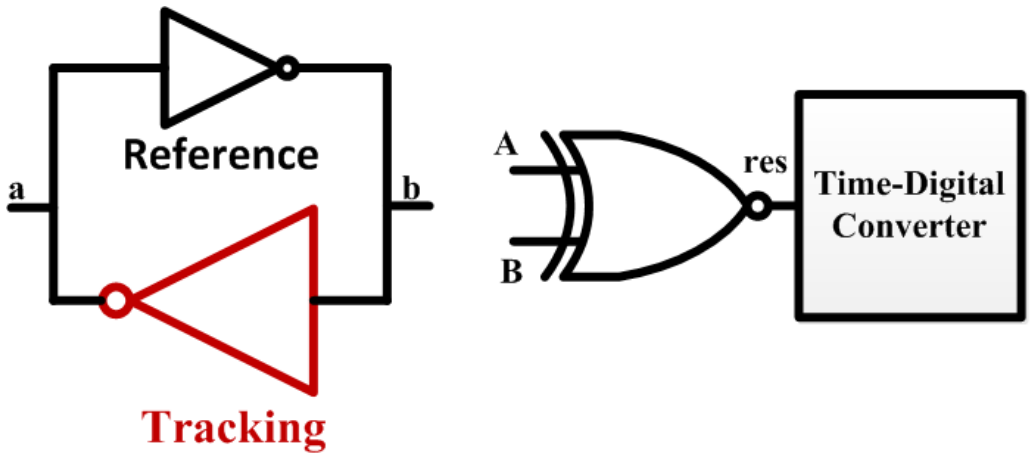


Figure 5.6. Metastability Resolution Time based wear-out Sensing

MetaCell: The metastable cell, as shown in Figure 5.7, consists of a pair of cross coupled inverters formed using transistors m1-m4. Transistors m1 and m2 are the tracking transistors and are sized larger than m3, m4. During tracking mode, the inverters are disconnected and a stress signal is applied to m1-m2. The stress signal can be a pre-designed signal or a sampled signal activity from a critical path. During periodic measurement, the cross coupled inverters are re-connected.

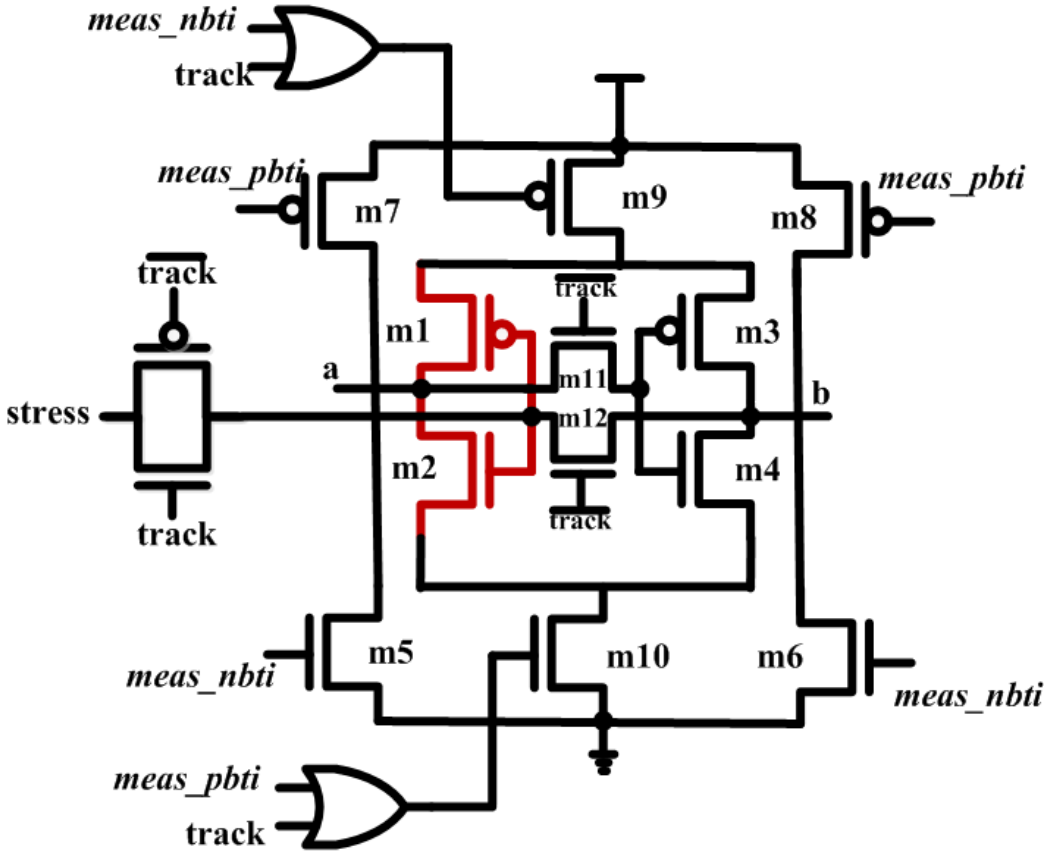


Figure 5.7. Metastable cells with tracking devices for NBTI/PBTI

Depending on the kind of test, NBTI or PBTI, the meas_nbti or meas_pbti signal is asserted. For NBTI measurement, the nodes a and b are discharged to ground and then allowed to resolve to a stable state. Since W(m1) $\gg$ W(m3), the resolution state is always constant (a=1 and b=0). However, the resolution time tracks the

threshold voltage of m1. As the stress applied on m1 increases, the V_t of m1 increases due to NBTI. Accordingly, the resolution time also increases during measurement. To measure PBTI, a pulse is asserted on the meas_pbti signal. Nodes a and b are pulled up to VDD and then allowed to discharge. Similar to NBTI measurement, the resolution time depends on the threshold voltage shift in m2. Resolution time gives an estimate of device degradation due to NBTI, as shown in Figure 5.8.

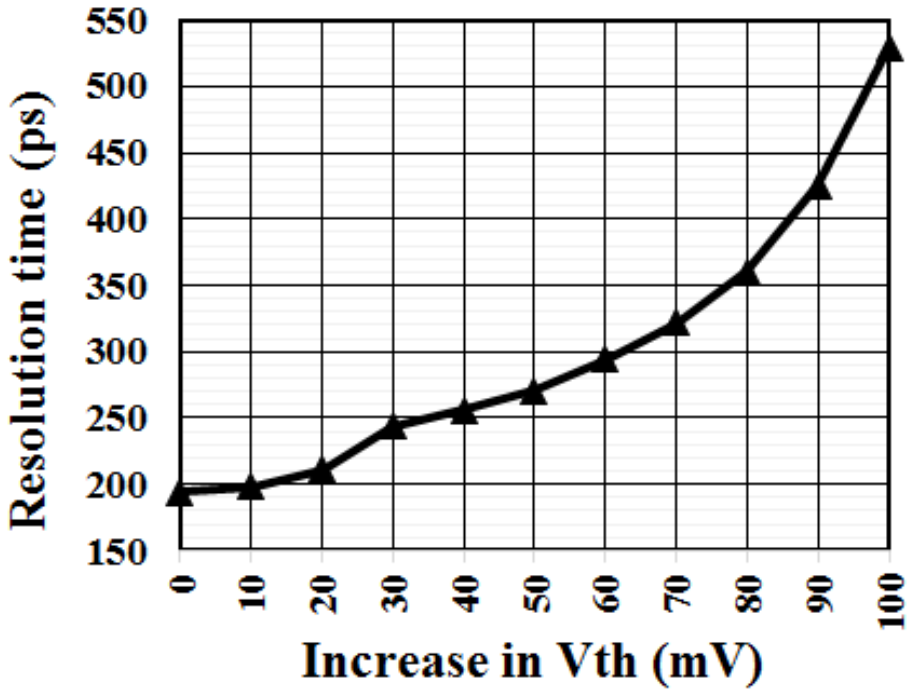


Figure 5.8. Variation of Resolution Time with V_t shift

The sensitivity of resolution time is ∼1ps/mV at the beginning of stress period and increases to >8ps/mV for large Vth degradations. One of the major issues with existing wear-out sensors is the impact of process variation on sensor behavior. In the proposed circuit, since resolution state of the metastable cell is irrelevant, the tracking transistors can be sized large enough to negate impact of process variation. Since V_t offset depends on device size, $\sigma(V_t)\alpha 1/\sqrt{(W*L)}$, larger tracking devices have inherently very small V_t offset. Further, larger tracking devices also dominate

the resolution time. As a result, any variation in reference inverter has negligible impact on resolution time. Figure 5.9 shows the standard deviation in resolution time before stress for W(m3,m4) = 0.36m and assuming V_t variation with σ=10%Vth.

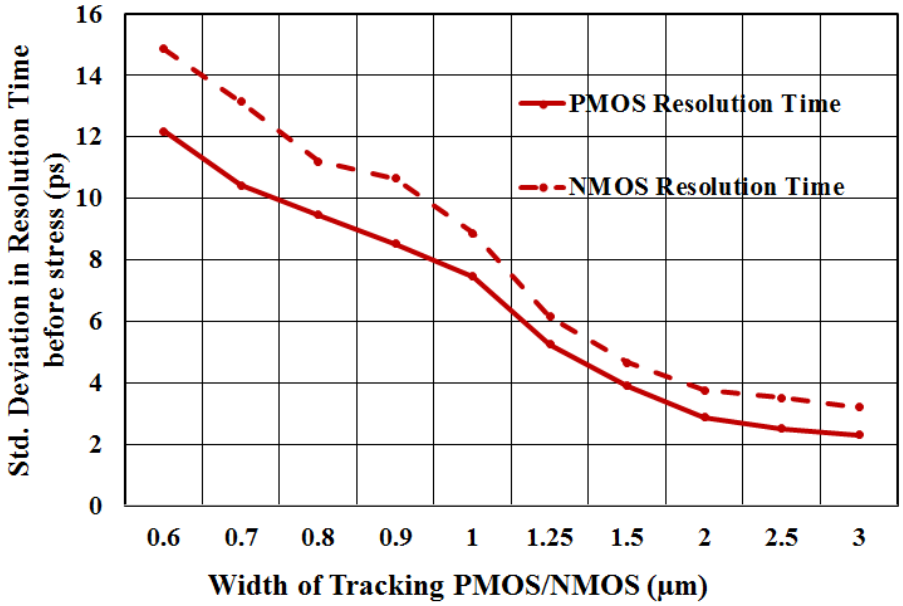


Figure 5.9. Minimizing resolution time offset due to process variation

If the tracking devices are sized at 3μm, the standard deviation in resolution time is ~2ps. This translates into a worst case 2mV standard deviation in V_t offset. Hence, the metastable tracking circuit is highly tolerant to intra-die process variation. Any global V_t offset does not impact the circuit performance. Further, using large devices decreases the impact of thermal noise on the resolution time of metastable circuit. Transient noise simulations in 32nm PTM models with tracking device width of 3μm show a standard deviation in resolution time of 0.6ps. Figure 5.10 shows the signals during tracking and measurement.

Time to Digital Converter: The Time to Digital Converter (TDC) measures the resolution time to estimate Vth degradation due to NBTI/PBTI. The TDC design determines the range and precision of wear-out sensing. A simple counter triggered by

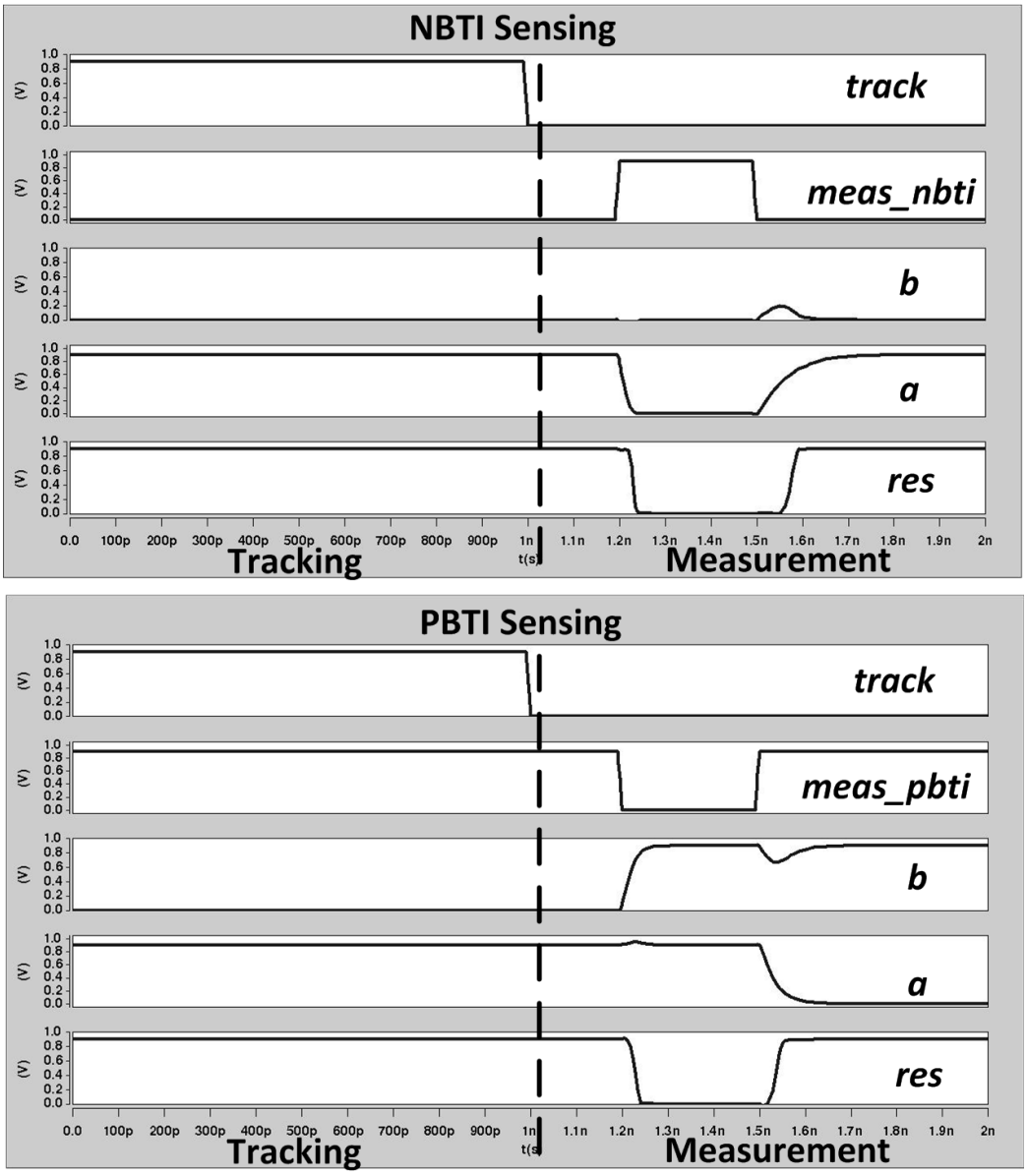


Figure 5.10. Time to Digital Converter (TDC)

a ring oscillator can be used as a TDC for large range but poor precision measurement. A vernier delay line based TDC can be used for a high precision resolution time detection. In this work, we propose a TDC with range of 880ps and precision of 10ps. A practical runtime application of wear-out sensor will have adaptive design control to vary voltage or frequency. This helps in graceful degradation of circuit

with device aging, but guaranteeing reliable operation. Scaling frequency and voltage pose additional challenges in designing PLL, clock distribution and creating robust power grid for different voltage ranges. As a result, it is fair to assume that a given design can be scaled only to a handful of frequency and voltage values. As seen in Figure 5.8, a 10ps precision in resolution time detection gives a worst case 10mV precision in ΔV_t detection. The precision improves with device aging since resolution time per ΔV_t increases. The proposed TDC consists of 11-stage ring oscillator with each inverter delay of 5ps (Period = 110ps), shown in Figure 5.11.

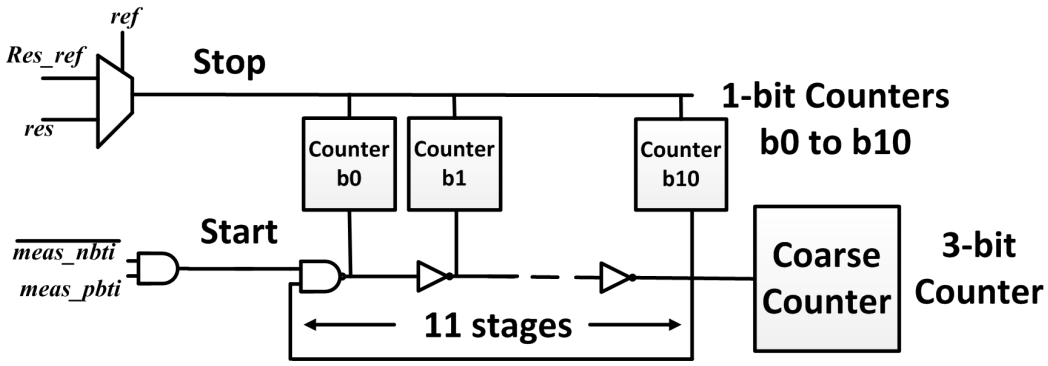


Figure 5.11. Tracking and Measurement for NBTI and PBTI

The ring oscillator triggers a counter for coarse time measurement. Further, the output of each inverter triggers a 1-bit counter for a more fine grained time measurement. The inverter delay determines the precision of resolution time measurement and the size of coarse counter determines the range of measurement. A 3-bit coarse counter gives a measurement range of 880ps. The resolution time measurement consists of two stages. In the first stage, either the meas_pbti or meas_nbti signal triggers the ring oscillator. A pre-designed reference resolution signal ref_res is then used to stop the counter. Since the reference signal is generated at a predefined time, the counter output after the first stage is used to estimate the precision of resolution time measurement. The coarse counter measures time as a multiple of ring oscillator period.

The 1-bit counters are triggered by individual inverters of the ring oscillator. As a result, they encode a fine grained time measurement smaller than the ring oscillator period. The 1-bit counters are alternately triggered in the order of b0, b2, b4 b10, b1, b3 and so on. Hence, the precision of fine grained time measurement is $2 * inverter_{delay}$. For instance, let a ref_res signal be generated 100ps after meas_$*$ signal. If the 1-bit counter output (b0-b10) is 11111010101, this indicates that 8 counters received a trigger. Hence, the reference measurement precision is 12.5ps. In the second stage, the ring oscillator is restarted again by one of the meas_$*$ signals. The counting is now stopped by the real time res signal coming from the metastable cell. The measured resolution time is equal to decoded counter value times the reference measurement precision. The two stage measurement technique is used to generate a reference precision before each measurement.

Tolerance to Process and Temperature Variation: The two stage measurement helps in minimizing measurement errors due to process induced variation in the ring oscillator or counters. The run time precision detection of ring oscillator also eliminates any error due to aging of devices in ring oscillator. Also, resolution time of metastable cell is affected by temperature. The two stage measurement technique provides temperature reference for measurement in the event of temperature variation. Assuming the ref_res signal is generated after 110ps (the nominal period of ring oscillator) after the meas_$*$ signal, it is expected to see all the 1-bit counters triggered once. This would set all bits b0-b10 to 1. If however, b0-b10 is observed to be 11111010101, it would indicate that the precision at this measurement temperature is $\sim$13.75ps. Since precision is a direct indication of inverter delays, the expected period is 137.5ps instead of the nominal 110ps. This indicates the temperature during measurement (T_m). To estimate ΔV_t, the resolution time curve similar to the one shown in 5.8, but for T_m is chosen. As a result, a more accurate ΔV_t estimation is performed even if temperature varies between different measurements.

5.2.3 Implementation and Results

The proposed circuit was implemented in 32nm Predictive Technology Models. Circuit simulations with Monte Carlo analysis was performed in HSPICE. Cadence UltraSim was used for device aging simulations. The nominal operating conditions were at 0.9V and 300K. A common sample stress signal was used for generating results and analysis. An approximate area estimate in terms of NAND2 gate equivalent is ∼150 gates. In 32nm technology, this translates into $105\mu m^2$ excluding routing overhead. Although this is significantly large compared to a single sensor area in [125], our proposed wear-out sensor is tolerant to process variation and does not require multiple instantiations. Further, a single sensor can provide finer resolution measurement. Hence, in terms of sensing area, an improvement of ∼8X can be expected for 5 measurement intervals.

The resolution time of metastable cell for NBTI sensing with device aging is as shown in Figure 5.12. The resolution time tracks degradation in Vth of stressed PMOS. Further, the resolution time also increases with operating temperature under stress. This signifies impact of temperature in accelerating NBTI effect. A similar analysis for PBTI sensing using stressed NMOS device is shown in Figure 5.13.

One of the major advantages of the proposed wear-out sensor is tolerance to process variation. The large tracking devices inherently have minimal variation. They also dominate the resolution time negating any impact of variation in reference transistors. The worst case 3σ deviation in resolution time for NBTI sensing at 300K is as shown in Figure 5.14. A worst case deviation in resolution time is ∼14ps. It should be noted from Figure 5.8 that $resolutiontime/\Delta V_t$ increases with increase in V_t. This means 14ps deviation in resolution time can result in a worst case error of 9.3% in ΔV_t. To provide fault tolerance, a sensor can be implemented with multiple metastable cells and a shared TDC. The ΔV_t estimation error can be significantly reduced by polling 3-5 metastable cells.

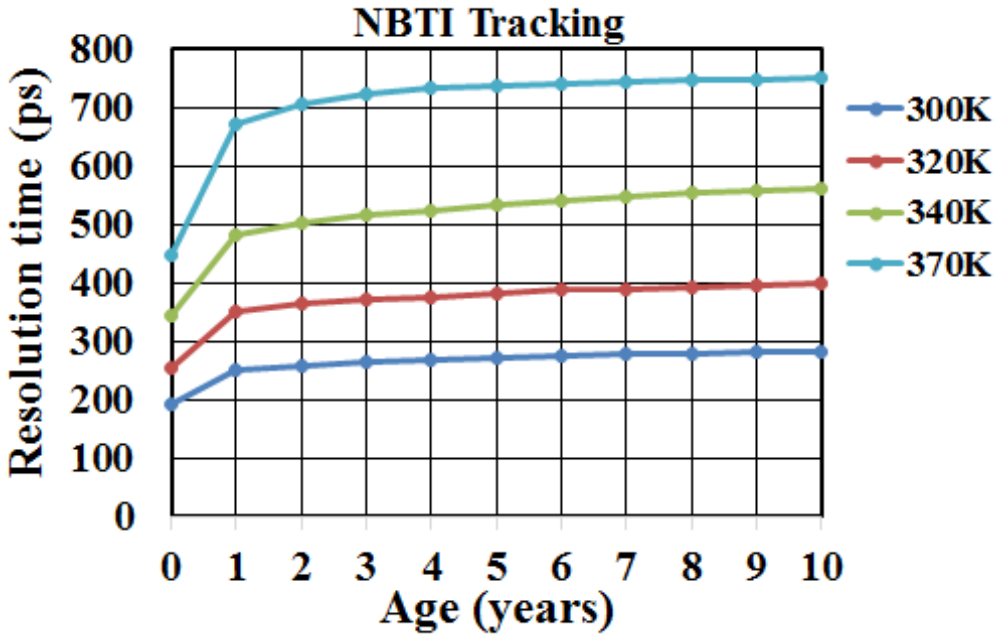


Figure 5.12. Tracking NBTI effect using resolution time

As previously described, the two stage TDC measurement technique provides accurate ΔV_t estimation across all measurement temperatures. Figure 5.15 shows the estimated period of ring oscillator for a constant START-STOP period of 110ps. The estimated ring oscillator period indicates the approximate measurement temperature. Since the focus of this work is not to design high precision wear-out sensor, minor error in temperature estimation can be tolerated. Any offset in RO period due to process variation can be captured by measuring the reference RO period before stress and at nominal temperature. Figure 5.16 shows the variation in resolution time with temperature. The reference TDC measurement provides an approximate measurement temperature. Accordingly, the V_t estimation is based on the appropriate temperature curve in Figure 5.16. For instance, if the estimated RO period during measurement is 122ps, this indicates the measurement temperature is 320K. Now, a resolution time measurement of 600ps will translate into a ΔV_t of 40mV.

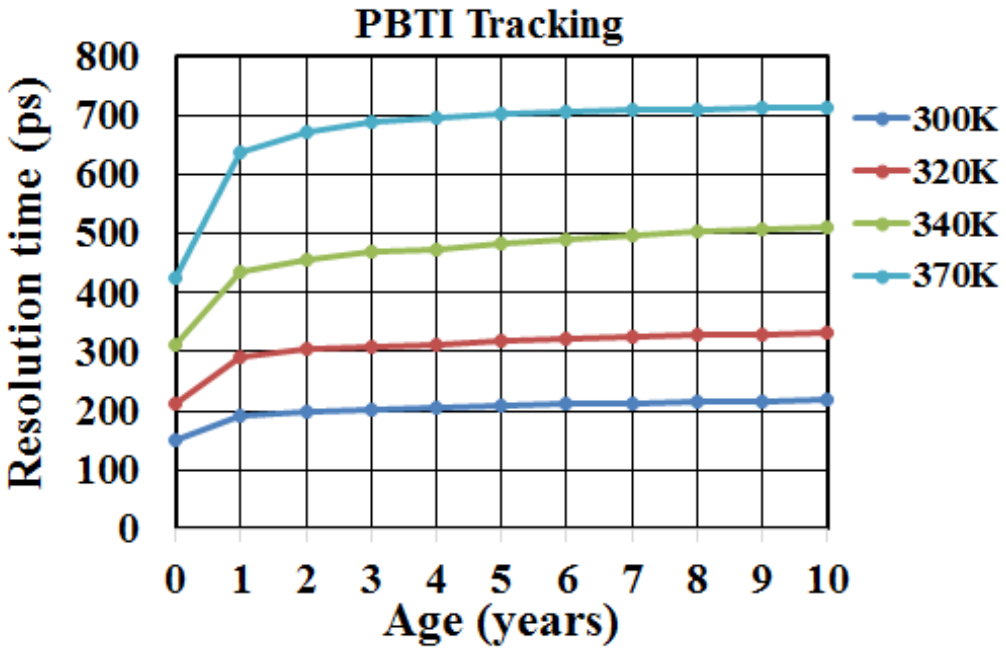


Figure 5.13. Tracking PBTI effect using resolution time

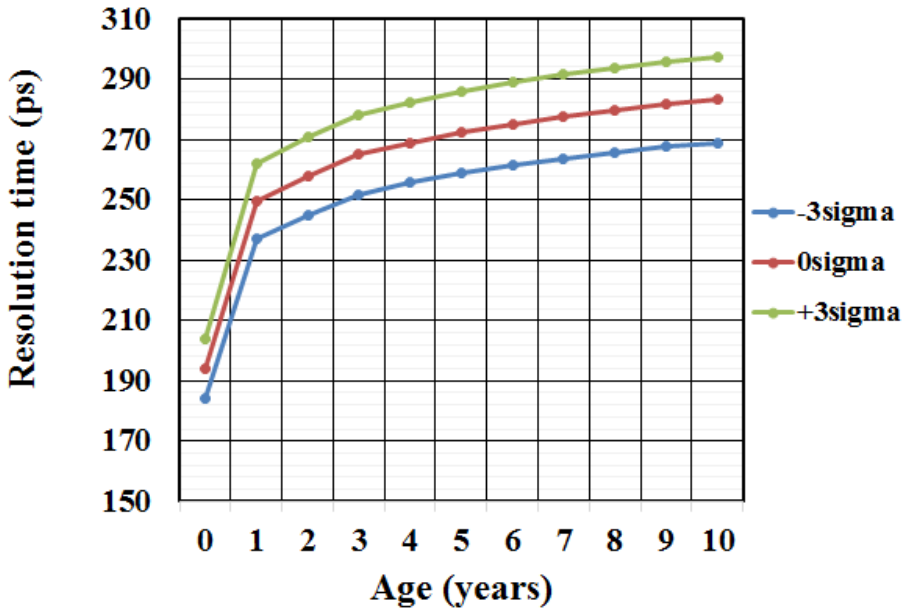


Figure 5.14. Worst case error in estimation of V_t degradation

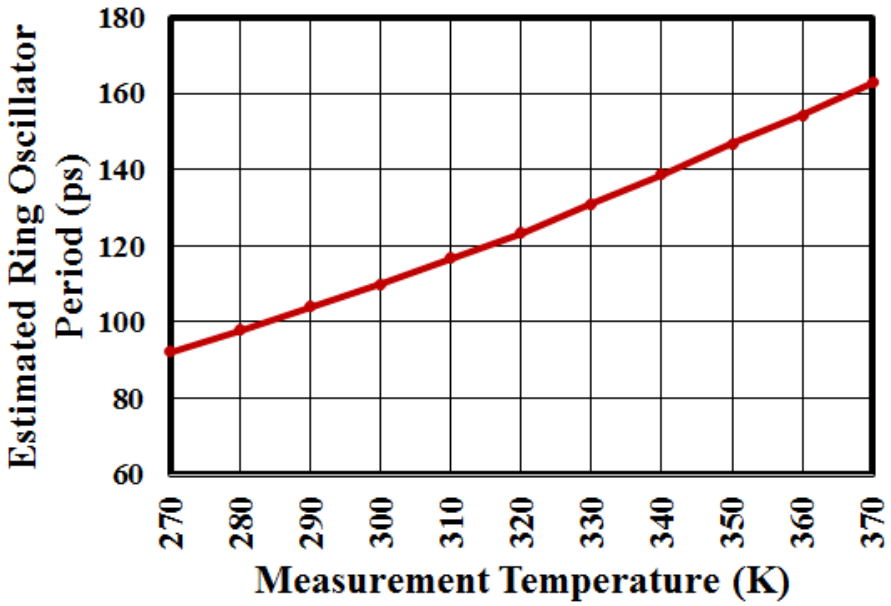


Figure 5.15. Tracking measurement temperature

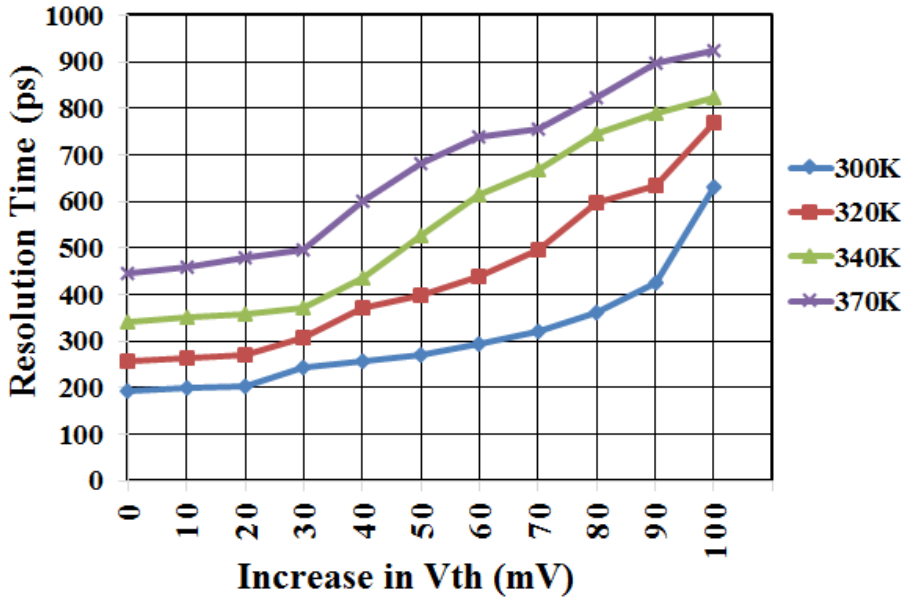


Figure 5.16. Variation in expected resolution time with measurement temperature

Without a two-step measurement, the 600ps resolution time will be erroneously detected as 100mV Vth degradation on nominal measurement temperature curve. The

measurement time including reference measurement is ∼1ns for a 100mV ΔV_t. Fast measurement prevents tracking devices from recovering and reducing the detected ΔV_t. The tracking power, which is leakage of metastable cell and TDC is 239nW at 300K.

5.3 On-chip Sensor for Characterization of Random Telegraph Noise

The wear-out sensor presented in the previous section can also be used to characterize Random Telegraph Noise (RTN). RTN is caused due to mechanisms similar to BTI and results in threshold voltage fluctuation. However, unlike BTI, RTN is a dynamic phenomena with the threshold voltages fluctuating between two values depending on the trap location in the oxide. Since variation in threshold voltage results in variation of transistor drain current, conventional techniques to characterize RTN use direct measurement of transistor current fluctuation [15]. However, this technique requires efficient Current-Voltage (I-V) amplification and digitization of the voltage values. Measurement noise during die probing and I-V conversion/amplification can cause errors in characterization. In this section, we present a fully digital on-chip characterization circuit for RTN using metastability resolution time.

5.3.1 Circuit for RTN characterization

Random Telegraph Noise is has two important characteristics:

1. Magnitude of Threshold voltage Variation / Magnitude of Current Variation ($\Delta Id/Id$)

2. Time constants for capture state and empty/emit state

The magnitude of RTN, which manifests in the form of $\Delta Id/Id$, depends on the location of the trap with respect to drain and source. The time constants for how long the device is in capture mode or empty mode depends on the material ad thickness of the oxide layer. The proposed sensor for RTN characterization is shown in Figure 5.17.

The sensor, called the MetaCell, consists of a cross coupled inverter which can be pre-charged or discharged and allowed to resolve to a stable state. The pre-charge

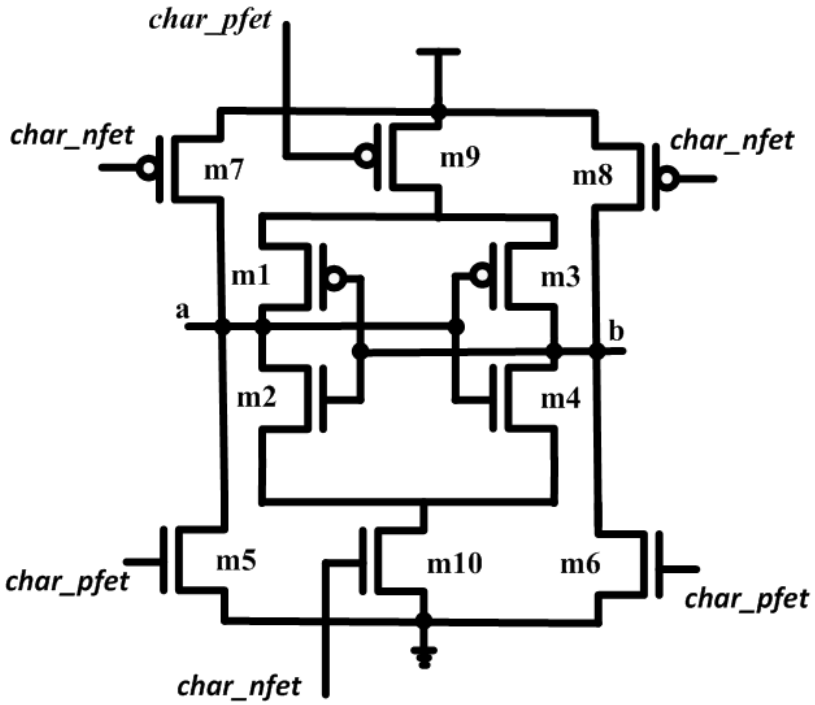


Figure 5.17. MetaStable Cell for sensing Random Telegraph Noise

and resolve signal $char\_nfet$ is used to characterize NFETs and the discharge and resolve cycle $char\_pfet$ is used to characterize PFETs. The signals $char\_pfet$ and $char\_nfet$ are one-hot, meaning, only one of the two signals can be active at a time. In the MetaCell, transistors m1 and m2 are sized 4X times the sizes of transistors m3 and m4 respectively, which are the devices under test. The resolution time of the MetaCell depends on the drain current in m1, m3 during PFET characterization and m2, m4 during NFET characterization. The resolution time can be amplified using a large capacitive load the the output of MetaCell and is measured using an on-chip Time-to-Digital Converter (TDC), Figure 5.18.

During characterization of NFETs, the $char\_pfet$ is maintained at 0. The $char\_nfet$ is initially pulled down to 0 to pre-charge the nodes a and b. Once the signal $char\_nfet$ is pulled hight, the MetaCell begins to move towards a metastable state before the transistor m1 pulls node a to 0 resulting in 1 at node b. The time taken by the

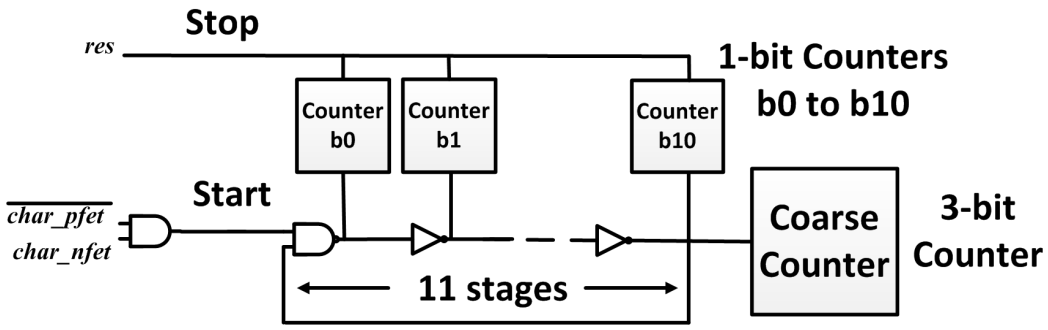


Figure 5.18. Time-to-Digital Converter to measure Resolution Time

MetaCell to resolve to stable states depends on the currents in transistor m1 and m2. If the capture state of RTN is denoted by 1 and the empty/emit state denoted by 0, the MetaCell may be in one of the following four states during resolution:

1. State 0: m2=0 and m4=0 - Resolution time t_0; the no-noise state
2. State 1: m2=0 and m4=1 - Resolution time t_1; $t_1 < t_0$
3. State 2: m2=1 and m4=0 - Resolution time t_0; $t_2 > t_0$
4. State 3: m2=1 and m4=1 - Resolution time t_0; max(t_0,t_3) depends on the relative magnitude of RTN in m2 and m4

The top level measurement setup is shown in Figure 5.19. The $char\_nfet$ signal is switched with a cycle time of $1\mu s$. Since RTN is a low frequency noise with capture and emit time constants of the order of milliseconds, 1000s of resolution time measurements are obtained for each of the four noise states. The measurements provide the following information:

1. t_0, t_1, t_2, t_3 - The resolution times of four noise states
2. T_0, T_1, T_2, T_3 - The average duration of each noise state

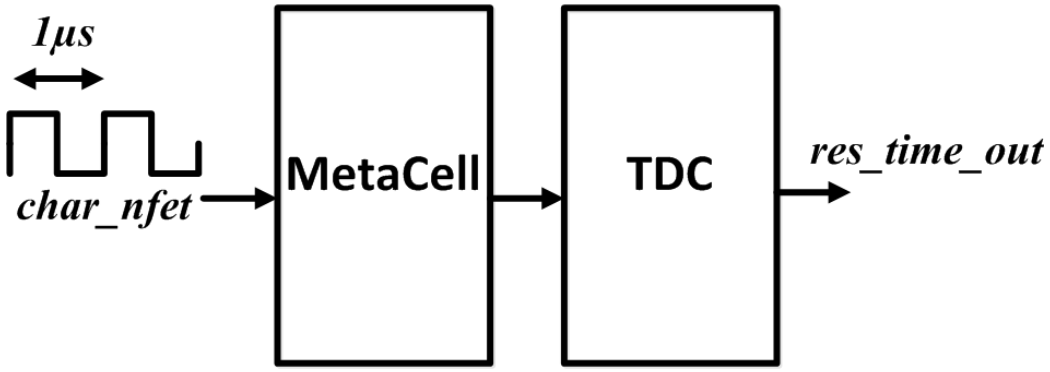


Figure 5.19. RTN Measurement with Single MetaCell

Since τ_{empty} of RTN is usually 2 to 3 times the $\tau_{capture}$, state 0 with m2=0 and m4=0 will have the largest time constant. Since the MetaCell is asymmetric with $W(m1, m2) = 4 * W(m3, m4)$, the offset in resolution time due to process variation is negligibly small. The noise condition of interest to characterize RTN in the device under test, m3, is state 1. Since both state 1 and state 3 have m3=1, both states may have $(t_1, t_3) > t_0$. However, state 3 with m2=1 and m4=1 has the least time constant and hence can be used to differentiate state 1 from state 3. The average resolution time in state 1 provides information about the magnitude of RTN (ΔV_t) and the average time durations of each state provide information about the capture and emit time constants, τ_c and τ_e respectively.

5.3.2 Implementation and Results

The test circuit for RTN characterization was implemented in 32nm Predictive Technology Models and simulated using NGSPICE. Transient noise simulation was used to model RTN. Since the characterization of RTN requires more than one device under test, an array of 1000 MetaCells are used in the measurement circuit, Figure 5.20. The $char\_nfet$ is switched at 1GHz and the signal is multiplexed to the MetaCell; there by measuring the resolution time of each MetaCell once every $1\mu s$. The actual number of MetaCells required for characterization are much higher than

1000 and the switching rate of $char\_nfet$ depends on the post-Si measurement setup. However, the array size and frequency number of $char\_nfet$ used in these experiments are for the purpose of reducing extensive simulation.

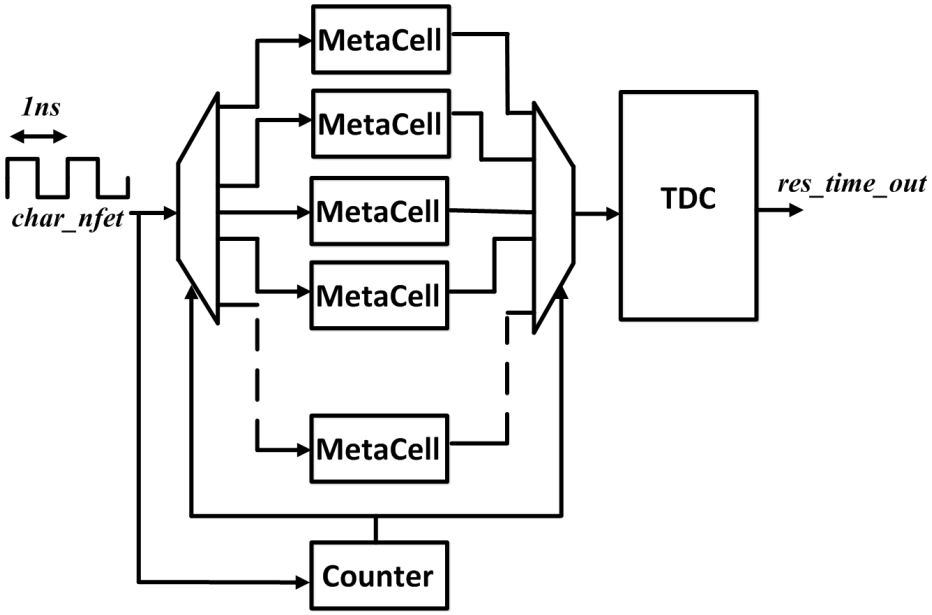


Figure 5.20. RTN Characterization using 1000 MetaCell Array

The variation in resolution time due to RTN is shown in Figure 5.21. Once the measurements are completed, the average resolution time in state 1 can be compared to the above data to estimate the percentage fluctuation in $\Delta Id/Id$ for each MetaCell. This can be translated into a the corresponding threshold voltage variation due to charge trapping in RTN. The large resolution time difference is achieved by adding capacitive load to the output of the MetaCell to increase fine-grained characterization. Figure 5.22 shows the distribution of average resolution time measured in state 1 across 1000 instances of MetaCell. For each resolution time, the corresponding expected RTN noise magnitude is calculated using the data in Figure 5.21. The magnitude and distribution of $\Delta Id/Id$ corresponds to the expected values and trend of RTN.

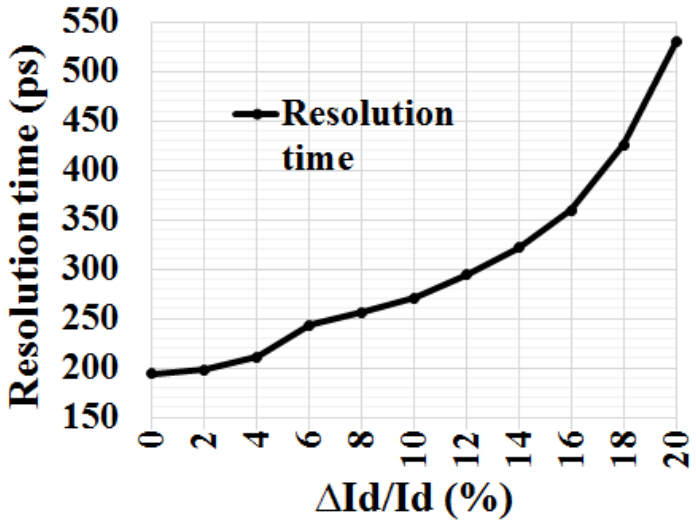


Figure 5.21. Variation in Resolution Time due to RTN

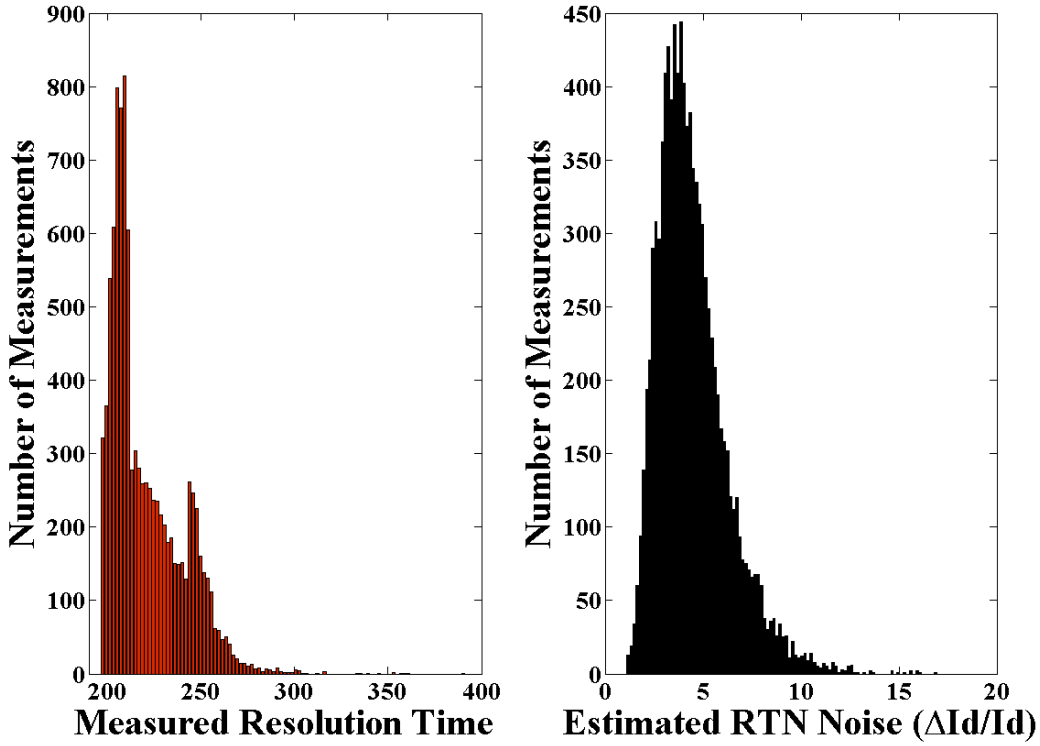


Figure 5.22. Measured Resolution Time for1000 MetaCell Array and Corresponding Estimation of $\Delta Id/Id$

CHAPTER 6

IMPACT OF NOISE IN NANOMETER SRAM

Static Random Access Memory (SRAM) is one of the most critical components of modern day microprocessors and embedded systems. SRAM access time and reliability define the performance of microprocessors. Process variability in advanced CMOS technologies have not only impacted the performance of SRAM cells, but also decreased yield and reliability of SRAM arrays [13, 32]. SRAM access time defines the performance threshold in microprocessors. A more critical issue in nanometer SRAM circuits is that of stability of storage cells. Increasing local variation between transistors affects the stability of SRAM cells. Read and write instability leads to performance degradation due to increased cache miss and necessitates expensive Error Correction Codes (ECC) [27]. A number of architectural techniques have been proposed to improve stability of SRAM cells. In [2], a variation aware cache architecture is proposed based on dynamic cache reconfiguration. Bit cells affected by process variation are detected and replaced to improve the yield of SRAM array. S. Mukhopadhyay, et.al propose a statistical approach to SRAM circuit design to identify variability issue early in process cycle [77]. A number of circuit techniques have also been proposed to improve stability of SRAM cells. In [81], a selective word line boosting technique is used to improve the performance and reliability of SRAM array. Apart from circuit and architectural techniques, statistical modeling of impact of process variability is used to provide adequate design margin and fault tolerance in large SRAM arrays [123]. Other reliability issues in SRAM cells arise from increasing sensitivity of transistors to operating conditions [75]. Power supply noise and

increasing on-chip thermal profile affect the performance and stability of SRAM cells as well [69]. Apart from process induced defects in SRAM cells, run time bit errors are observed due to random alpha particle and cosmic neutron strikes [65]. These events termed as Soft-Errors have been widely studied for their impact on all forms of storage devices. Soft Error Rate (SER) is characterized using empirical methods [90], circuit sensors [31] and accelerated test methodologies [87].

In this chapter, we study the impact of random on-chip thermal noise and Random Telegraph Noise (RTN) on SRAM circuits. In the first section, we present the impact of thermal noise on SRAM stability Bit Error Rate (BER) [101] during operation. In the second and third sections, we present impact of thermal noise [100] and RTN during SRAM testing.

6.1 Impact of Thermal Noise on SRAM Stability

Conventional 6T SRAM circuits rely on matched pair of inverters to store a single bit. Access transistors are used to read from and write into SRAM cells. The pull-up, access transistors and pull-down devices are appropriately sized to provide adequate stability during read and write operations. The SRAM cell designed and analyzed in this work is not optimized for performance. The objective of this work is to study the impact of thermal noise on stability of degraded SRAM cells. Figure 6.1 depicts the thermal noise at different nodes of an SRAM cell.

When a bit zero stored in the cell is read, the BL and $\overline{BL}$ signals are pre-charged to Vdd. The WL is turned ON to discharge the BL and hence read the bit 0. Figure 6.2 shows the transient noise simulation results for three different SRAM cells, each storing a bit 0 and with WL turned HIGH.

The results include simulation with 1000 different noise samples generated by HSPICE using thermal noise parameters in transistor models at 100C and nominal supply voltage of 0.8V. Since thermal noise is a random independent phenomenon, the 1000

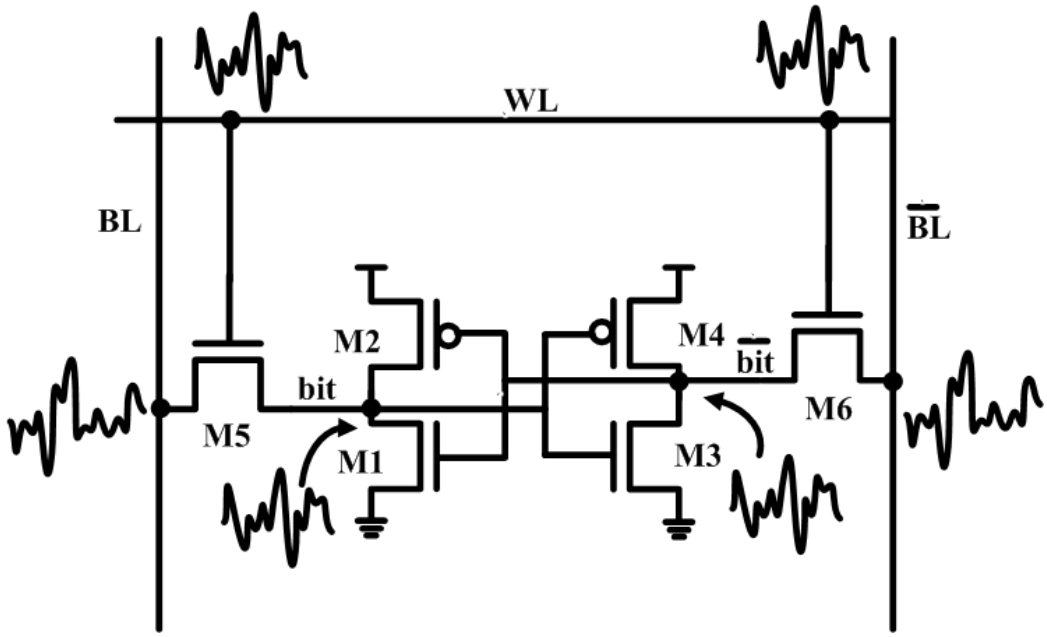


Figure 6.1. Thermal noise in 6T SRAM bit cell

samples are generated using 1000 unique seed values in HSPICE for random noise simulation. The waveforms show the stability of the three bit cells. In Cell-1, the Vt of pull down transistor is degraded by 6σ. Throughout the 10ns simulation time, the internal node of this cell does not switch. This shows that the cell is read stable. It would be fair to consider all bit cells with degradation smaller than 6σ to also be stable at extreme operating condition. In case of Cell-2 with 6.5σ variation, the internal bit always flips for all 1000 noise samples. This indicates that cell above 6.5σ variation are read-unstable. They will be flipped during read operation or during dummy read, when the WL is turned ON to access a different bit cell on the same SRAM row. However, it is very interesting to note that each bit flip is observed at a difference instant of time. This is due to the impact of thermal noise on WL, BL and internal nodes of the bit cell. The most interesting scenario for this work is Cell-3 with a degradation of 6.25σ. The results show that the internal bit may or may not flip within 10ns depending on the thermal noise sequence. Further, the time taken

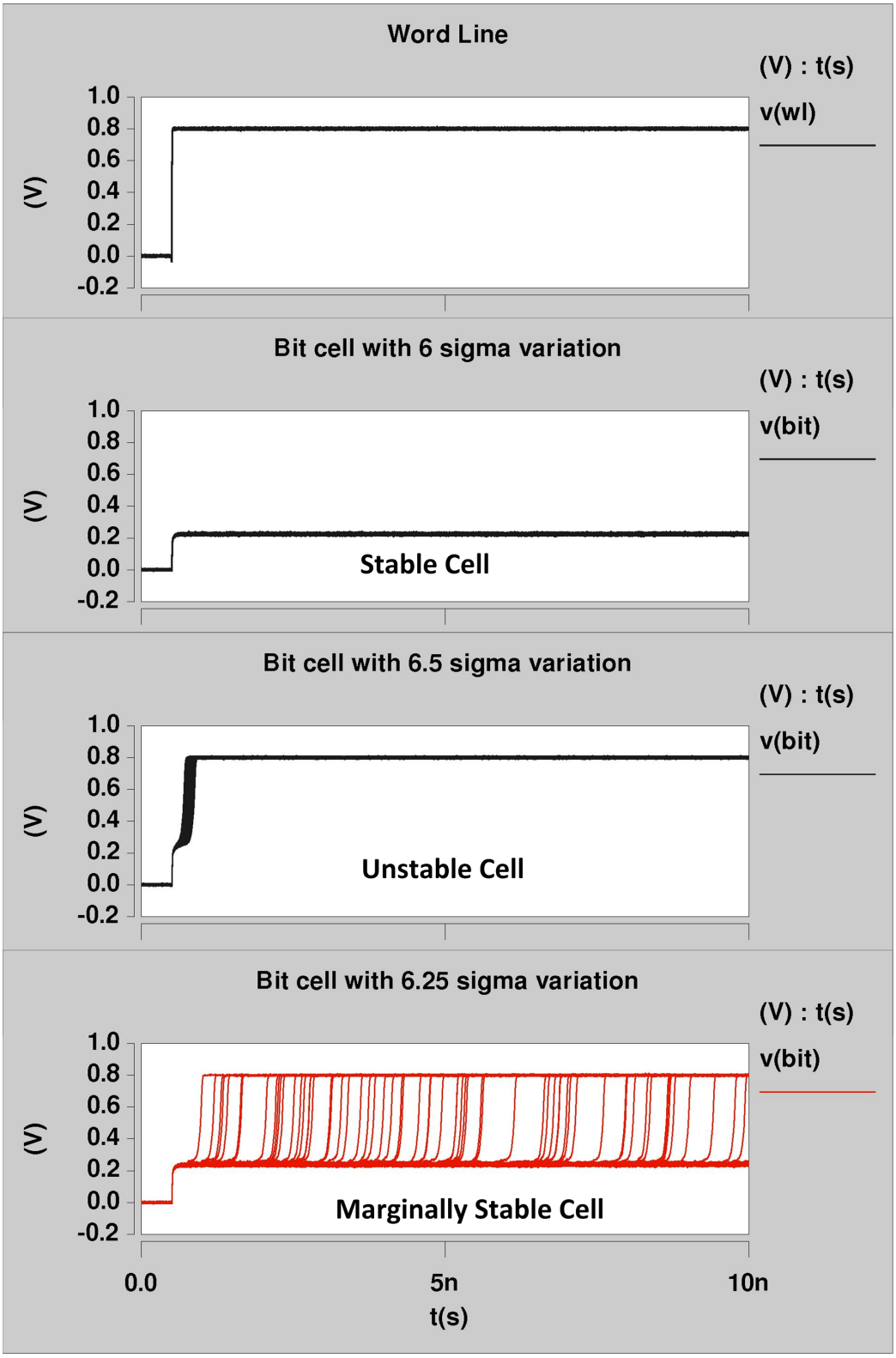


Figure 6.2. Impact of thermal noise on SRAM stability

for the internal bit to flip has significant temporal variation. We refer to these bit cells as Marginally Stable cells, Figure 6.3.

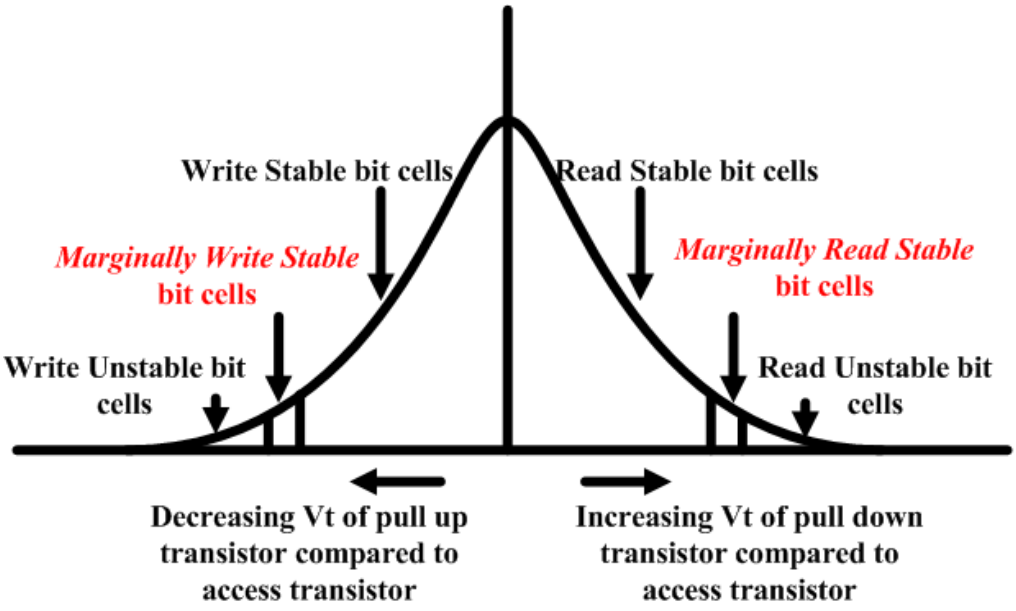


Figure 6.3. SRAM Stability with V_{th} Variation

Marginally Stable cells have a degradation small enough to appear stable under most operating conditions. However, they may flip the internal state under specific noise conditions. This leads to random bit errors during normal SRAM operations. Each bit cell in an array is expected to be stable at all specified operating conditions. Memory testing is done at these corner cases to detect unstable bits. A similar analysis can be performed to study impact of thermal noise on marginally write-stable bits. Marginally write stable bits will cause random write errors based on thermal noise in the pull-up and access transistors. Although ECC is used for error correction, the error detection and correction based on ECC are expensive in terms of area and power. Typical, ECC provides two bit error detection and one bit error correction. Random thermal noise coupled with other errors like soft error could increase the rate of double bit errors. This affects correctness of memory accesses. In

microprocessors, invalidating a cache page has been used to reduce bit errors due to unstable bit cells. However, such techniques are only effective for clustered unstable bits. Process variation and effect of thermal noise on bit cells is random and can be spread spatially across a swathe of memory array. Invalidating aging memory blocks based on few random bit errors reduces the effective cache size and hence impacts the overall memory access time.

6.1.1 SRAM Bit Error Rate (BER) due to Random Thermal Noise

Highly unstable SRAM cells are detected and replaced during manufacturing test and repair. However, bit cells that are marginally stable due to process variation or become marginally stable due to long term aging may pass the manufacturing test due to favorable thermal noise events during test, but fail during functional operation. SRAM cell stability is a major determinant of V_{min} for ultra-low power mobile computing. At low voltages, thermal noise plays a role in SRAM cell stability.

6.1.2 Characterizing Bit Error Rate

Random physical phenomena causing computation or storage errors cannot be avoided. However, they need to be characterized to better margin the design and have fault tolerance schemes in the event of a bit error. Soft Error Rate (SER) is a prominent example of bit error characterization [4]. In recent times, researchers have tried to characterize errors due to RTN through accelerated testing [105]. Random bit errors due to thermal noise are caused by a combination of noise potentials at the different nodes of a SRAM cell. As a result, we define a thermal event, as a combination of thermal noise potentials that causes read or write instability in a SRAM bit cell. The probability of thermal event is dependent on the degradation of each cell. A bit cell with less than 4σ variation will require thermal noise potentials in excess of 125mV to induce any error. Such large noise potentials are highly improbable

even at corner voltage and temperature operations. Similarly, a bit cell with 8σ degradation cannot avoid read and write error for most thermal noise conditions.

We define *Random Bit Error Rate (BER)* as the *probability of a random event causing read disturb or write error in the marginally stable cells of an SRAM array, assuming uniform access to all the physical cells.* Since thermal events are random and independent across time, we estimate the occurrence of a thermal event $p(sw_i)$, the probability of switching in case of read and $p(we_i)$, the probability of successful write, within a small window of t sec. The values of $p(sw_i)$ and $p(we_i)$ are estimated using circuit simulation and are a function of the degradation of bit cell i. The probability of read disturb in a marginal bit cell during a single access of duration T is given by $p(swr_i)$,

$$p\left(swr_i\right) = 1 - \left(1 - p\left(sw_i\right)\right)^{\frac{T}{t}} \tag{6.1}$$

Similarly, the probability of write error during a single write operation with WL turned ON for T sec is given by,

$$p\left(wer_i\right) = \left(1 - p\left(we_i\right)\right)^{\frac{T}{t}} \tag{6.2}$$

The overall estimated BER of an SRAM array is a stochastic measure of the number of times these marginal cells are accessed and the probability of an error occurring during that access. For an array size N, the expected BER per access is,

$$E(BER) = \frac{\sum_{i=0}^{m} p\left(swr_i\right) + \sum_{j=0}^{m} p\left(wer_j\right)}{N} \tag{6.3}$$

A plot of expected BER in 32nm 2MB SRAM array shows a higher random error rate at lower operating voltages, shown in Figure 6.4.

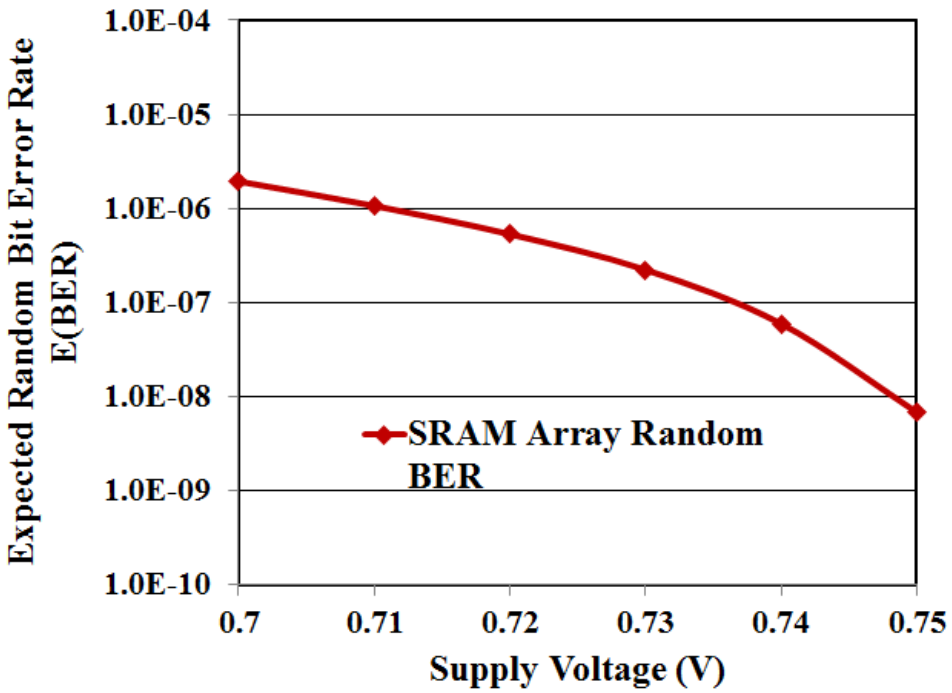


Figure 6.4. Expected BER in a sample 2MB SRAM array

6.1.3 Techniques to Minimize Random Bit Error Rate

Conventional SRAM self-repair techniques detect unstable bit cells and replace them with spare rows. However, using a similar technique for marginally stable cells will decrease SRAM yield and cache size over the lifetime of a microprocessor, affecting processor performance. As a result, it is critical to track and reduce random BER due to process variation and long term aging.

Increased Vmin in Low Power Mode: Current day microprocessors operate in multiple voltage and frequency modes based on work load for energy efficient computing. The Vmin for SRAM arrays may vary from 700mV to 800mV. The standard deviation of thermal noise increases at lower supply voltages. In this architectural technique we propose implementation of a bit stability counter to track the number of random bit errors. Conventional Error Correction Codes use additional parity bits for error detection and correction. Based on this detection, the counter is incremented. Once the

counter value reaches a predefined threshold, it indicates significant stability issue. The supply voltage in low power mode can then be increased to reduce the impact of thermal noise. However, a global increase in SRAM supply voltage increases both dynamic and leakage power. Cache architectures employing a fine grained voltage scaling or power gating can keep track of marginally stable bits and boost supply voltage only in appropriate memory banks.

Selective Multi-level Word Line Voltage: Word Line (WL) voltage boosting and scaling are used to improve SRAM stability. Reducing the WL voltage reduces read disturbs [74, 81]. However, they potentially increase write errors due to reduced overdrive on access transistors. Boosting WL voltage improves write stability. The boost is limited by the stability of SRAM cells experiencing dummy read during a write operation. Similar multi-level WL technique can be used to reduce the impact of thermal noise. Scaling WL during read operation will require larger thermal noise at the RAM cell nodes to disturb the internal state. However, this also increases the read access time and hence is a trade-off with SRAM performance. Similarly, boosting WL during write operation reduces write errors in marginally stable cells. Again, depending on error detection by conventional ECC coding, cache rows can be selectively scaled or boosted. If a marginally stable cell is detected in a particular array row, selective multi-level WL can be enabled only in those rows based on read or write instability.

The reliability improvement techniques proposed above make use of existing SRAM stability improvement techniques. The techniques reduce random bit error rate with minimal loss in SRAM yield. The cache size doe not decrease drastically with aging and the performance of the SRAM array is degraded gracefully over time.

6.1.4 Implementation and Results

The SRAM implementation and analysis were performed in 32nm Predictive Technology Models. HSPICE was used for transient noise analysis. Unique noise sequences were used to analyze the impact of random thermal noise. The random noise was generated by HSPICE using transistor model and based on device sizes and operating conditions.

Figures 6.5 and 6.6 show the probabilities of a thermal event occurring during read and write operations in a sampling window of 10ps at 30C and supply voltage of 700mV.

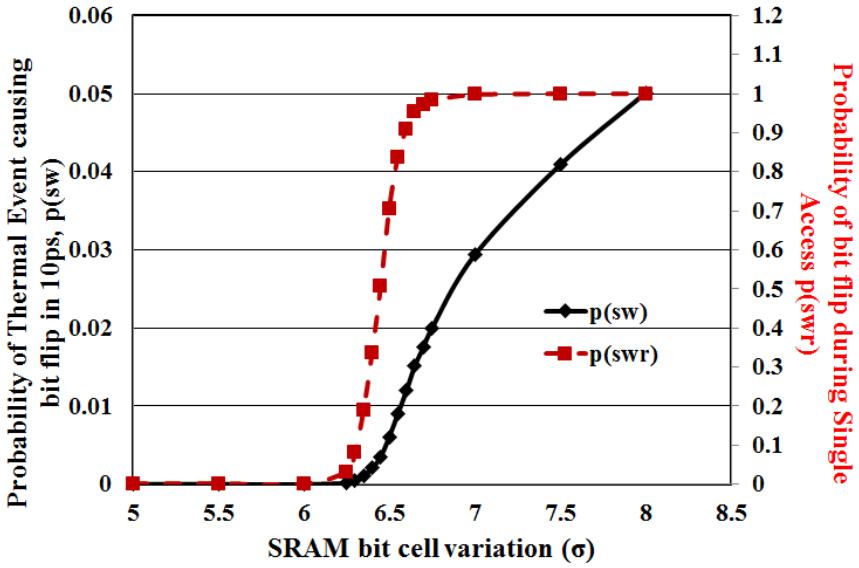


Figure 6.5. Probability of Bit Flip during Read

This translates into probability of read disturb and write error across different cell degradation. It can be seen that bit cells with degradation 6σ to 6.5σ have variable probabilities of read disturb. These cells are marginally read stable. The stability issues are further exacerbated by increased on-chip temperature. Similarly, bit cells in the region of 6.25σ to 6.75σ are marginally write stable. Figure 6.7 shows the probabilities of read disturb and write error in a SRAM cell with 6.2σ and 6.5σ

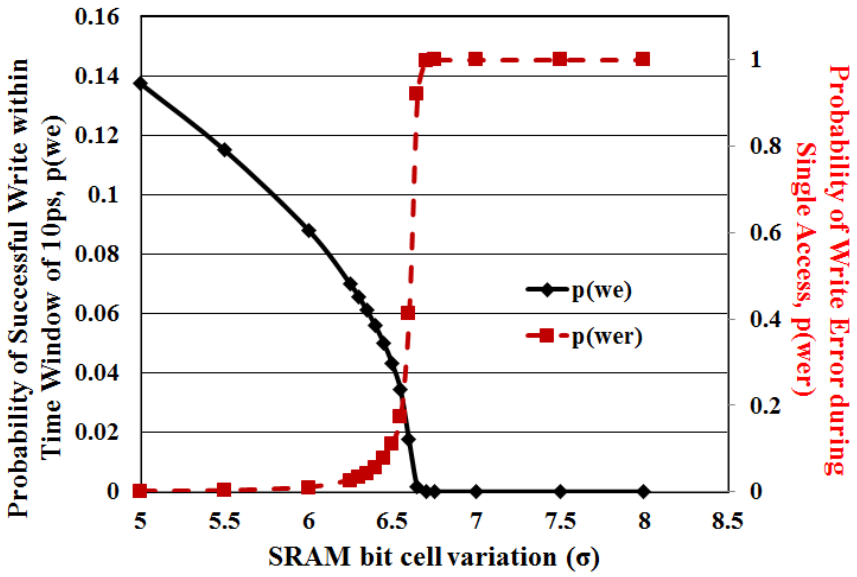


Figure 6.6. Probability of Write Error

degradation respectively, with Vmin Boost technique. It is evident that boosting operating voltage significantly reduces probability of random bit flip or write error. Stability also improves for the same cells with Multi-level WL technique, Figure 6.8. Boosting/scaling WL voltage reduces probability of bit errors. However, WL scaling technique incurs a performance overhead. Lower WL voltage reduces the overdrive on access transistor. While this is advantageous for read stability, it increases the access time of the SRAM cell. Simulations indicate that for a 100mV scaling in WL voltage, the time required to build a reference 100mV differential across bit_line and ∼bit_line increases by 10%.

A statistical analysis based on the proposed BER estimation technique was performed using MatLab. Assuming NBTI induced SNM shift of ∼10% every 10^8 sec (∼3Years) from [51], the expected BER is calculated for a 2MB SRAM array, Figure 6.9. BER of baseline SRAM array is compared with the two BER reduction techniques. Both techniques provide >10X improvement in BER. This prevents marginal

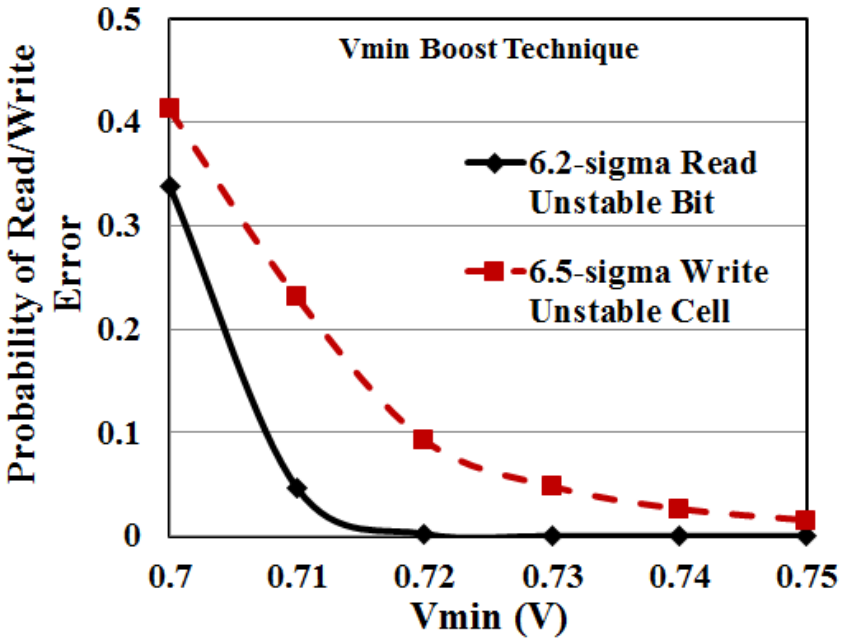


Figure 6.7. Variation with BER with Vmin

cells from being replaced by conventional self-test. The yield of the SRAM array and the longevity of physical storage are improved.

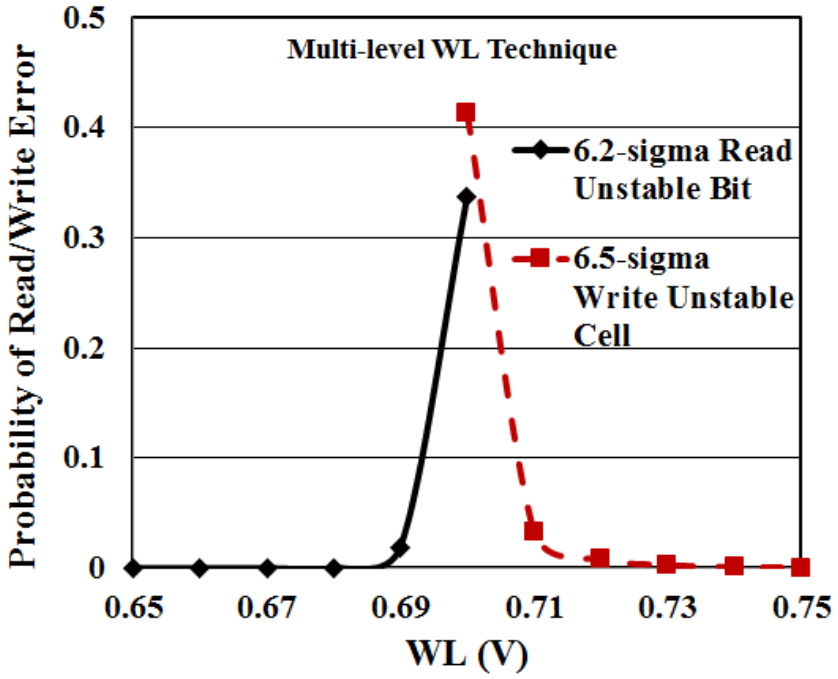


Figure 6.8. Variation of BER with WL voltage

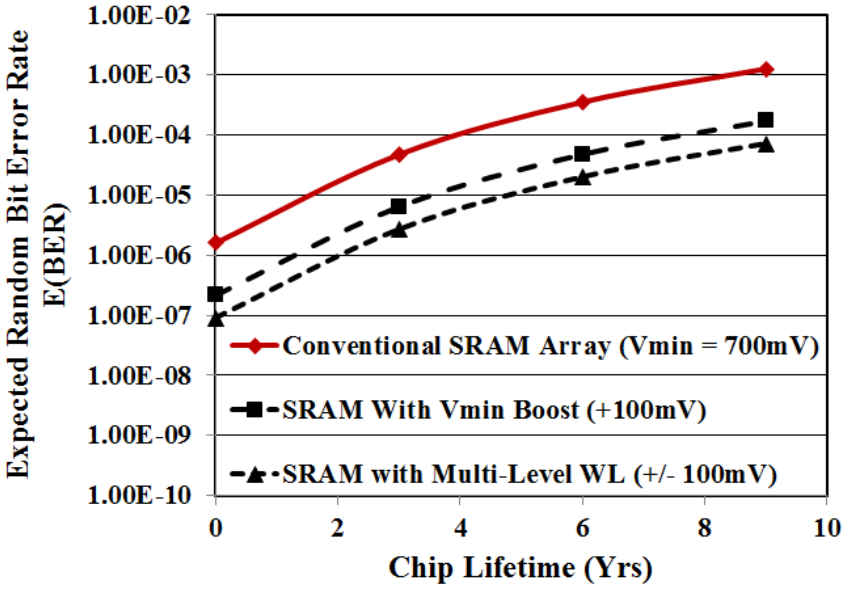


Figure 6.9. E(BER) for sample 2MB SRAM array

6.2 SRAM Test Coverage Uncertainty due to Random Thermal Noise

Fault models and test methodology for SRAM cell arrays differ from random logic [20, 21]. Defects like shorts, resistive opens and missing contacts causing a bit cells to be stuck at a constant value are modeled using conventional stuck-at faults. The high density regular arrays of SRAM cells result in coupling effects among neighboring cells necessitating additional fault models. *State coupling fault-* models coupling between two bit cells, *multiple access fault-* models impact of write operation in a bit cell on its neighboring cells and *data retention fault-* models loss of bit cell state after finite amount of time. Coupling faults between cells are also modeled as Coupling, Inverse Coupling and Idempotent faults. Traditionally, memory testing is performed using Built In Self-Test (BIST) methodology [112] or Direct Access Test (DAT) . Since SRAM bit cells are structured storage elements, defective and unstable bit cells can be replaced with spare bit cell rows. A number of self repair algorithms [45] and implementations have been proposed and used to compensate for SRAM yield loss [120].

6.2.1 Impact of Thermal Noise on SRAM Testing

Impact of Thermal Noise on Fault Coverage: The effect of thermal noise on marginally stable bits impacts memory test coverage in sub 45nm SRAM circuits. Test algorithms are run in corner conditions of operation only during the test stage. It is impossible to mimic the exact thermal noise conditions on all nodes of marginally stable cells to induce read/write errors. As a result, such cells may remain undetected during test. If not detected during chip test, normal power-up BIST procedure cannot detect stability issues in corner cases.Conventional SRAM testing is based on Built-in Self Test or Direct Access Test. The March algorithm is the fundamental basis for memory self-test [117]. A number of variations and improvements of March algorithm

like March-X, March-Y and March-B tests are used in the industry [118, 30]. The tests perform a sequence of read and write operations optimized to induce maximum number of faults; both stuck-at/transition faults as well as neighborhood coupling faults. The length of the test sequence governs the fault coverage, at the same time increasing test time. However, any test algorithm performs a finite number of accesses (read/write) per bit cell. Figure 6.10 shows the simulation result to mimic a 10 cycle read test on a 32nm technology 6.25σ degraded bit cell at 100C and 0.8V.

The simulations were performed for 1000 unique samples of thermal noise. For the sample set observed, the bit cell stability issue maybe undetected if accessed only once during the test cycle. However, with adequate thermal noise, the cell may flip and result in bit error during normal operation. So, test methodology should induce fault or characterize such marginally stable cells. The motive in write test is to induce a write failure in a marginally write stable cell. Figure 6.10 also shows results from a simulation to mimic 20 write cycles into a 7σ cell at 100C and nominal power supply of 0.8V. Across 1000 samples of thermal noise sequences it can be seen that a number of write operations fail during the first write cycle. During chip testing, if a thermal noise condition results in a correct write, the cell will be deemed stable. However, during normal operation there could be multiple write failures leading to bit errors. Therefore, fault coverage metric of conventional test methodology needs to be redefined in terms of statistical metric. For a given process and SRAM cell design, a stochastic value for fault coverage needs to be calculated. This number provides a confidence level for memory self-test and repair. It can also be used to characterize random bit error rate or erratic bits during normal operation of the SRAM array.

Stochastic Fault Coverage: We define *stochastic fault coverage* as the *combined probability of thermal noise events occurring during test to induce a bit flip during read and error during write for all unstable cells.* Consider an SRAM array of n-bits in a process where bit-cell with greater than X-σ degradation are marginally stable

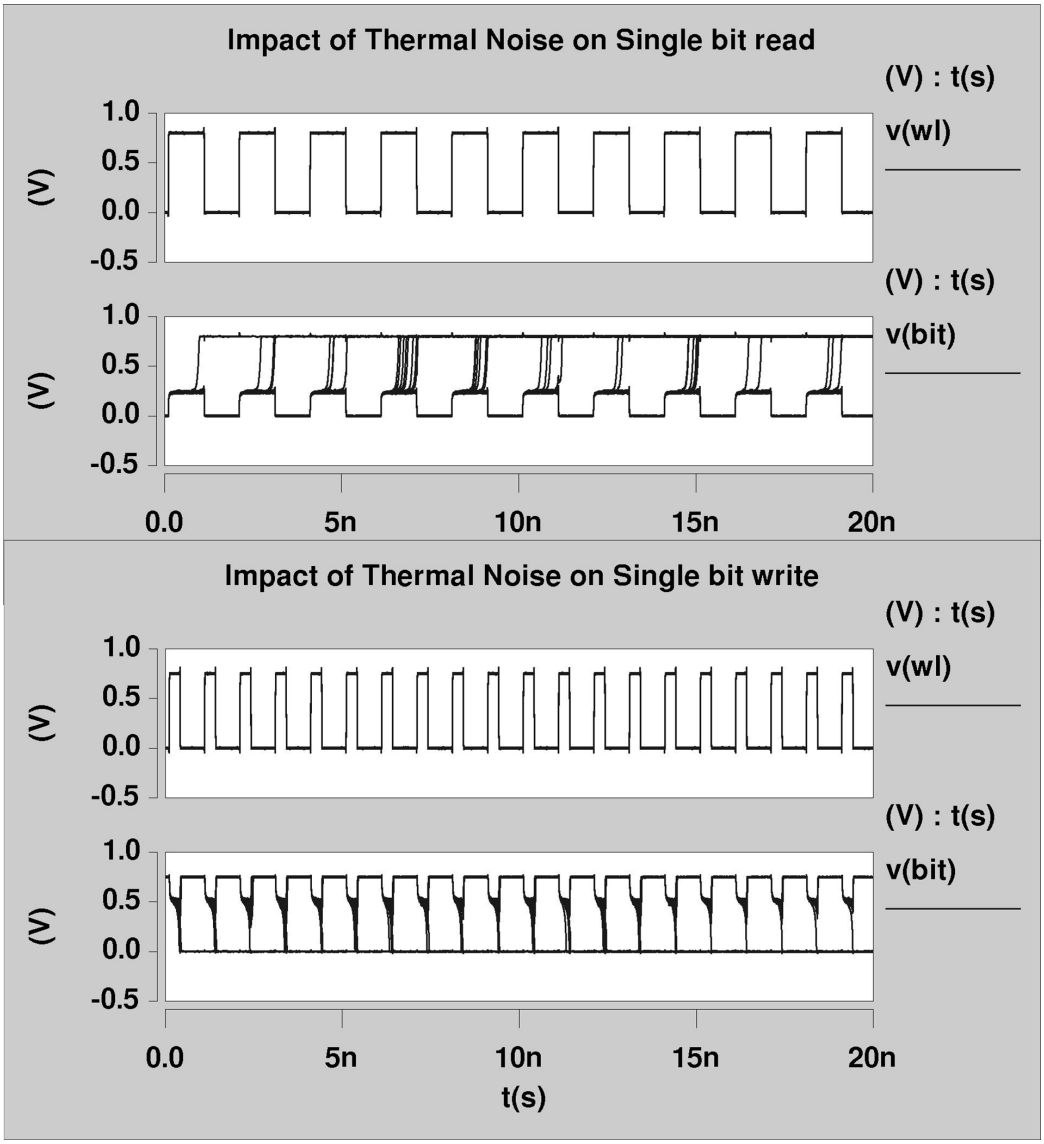


Figure 6.10. Impact on thermal noise on read/write failure during test

or completely unstable. In an ideal scenario, all the cells beyond X- degradation flip during testing. This would result in 100% fault coverage. Since thermal noise at different nodes of the 6T cell could impact cell stability, we define *thermal noise event* as the combined effect of thermal noise at each node of the bit cell which causes the internal bits to flip during read or write operation. Each thermal noise event is random with no temporal correlation. We estimate probability of a thermal event

large enough to cause a bit flip in a small time window, as shown in Figure 6.11. During the read cycle, a thermal event large enough to flip the bit may occur. The flipping of bit during test is a one-way process. Once the bit has flipped, it cannot be restored till the end of the read cycle. The probability of a given degraded cell switching during read test is *One Minus* the probability of a thermal event large enough to cause a flip not occurring in any timing window during the read cycle,

$$p\left(swt_i\right) = 1 - \left(1 - p\left(sw_i\right)\right)^{\frac{T}{N}} \tag{6.4}$$

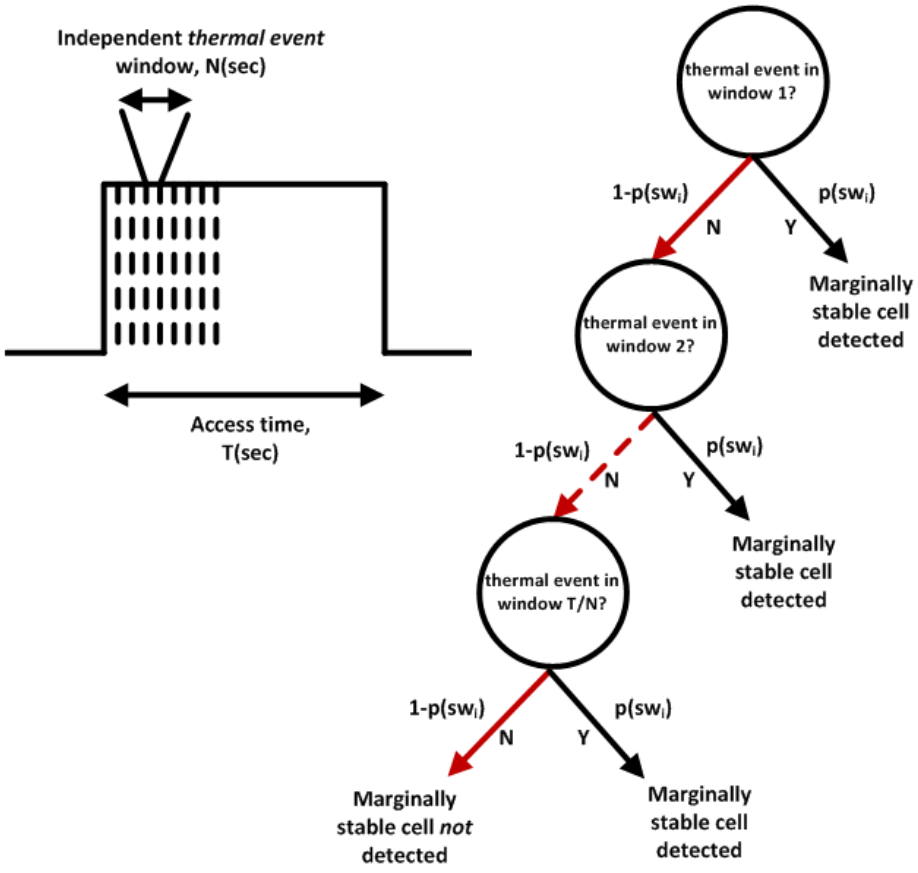


Figure 6.11. Stochastic read Stability Coverage

where, $p\left(swt_i\right)$ is the switching probability of bit cell with degradation i, T is the time for which word line is ON, $p\left(sw_i\right)$ is the probability of a thermal noise event

occurring in a window of N sec for degradation i. During write test, it is desirable to observe a write error in order to detect an unstable cell, Figure 6.12.

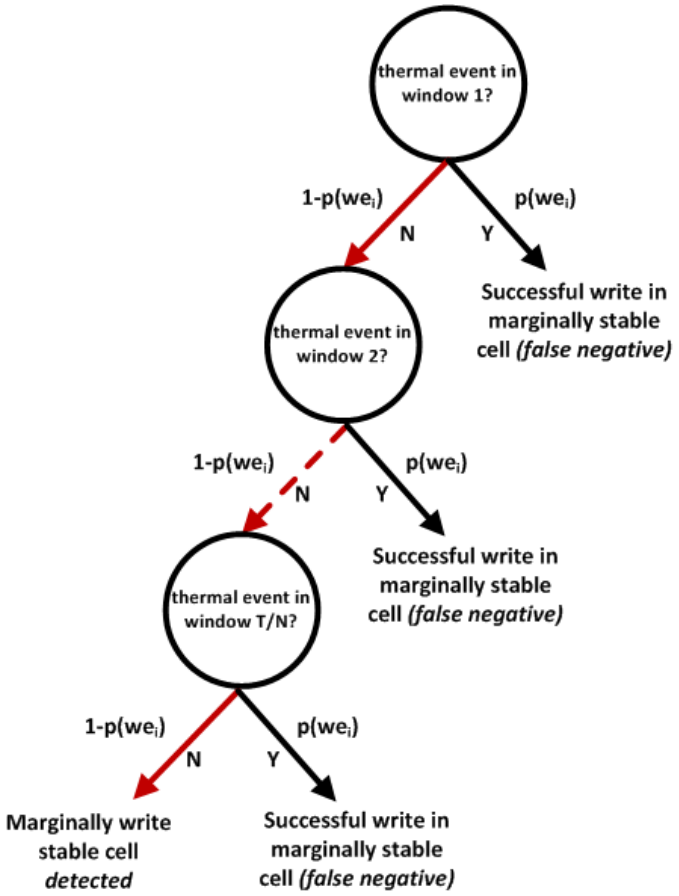


Figure 6.12. Stochastic write Stability Coverage

The probability of write error during test, $p\left(wet_j\right)$ is,

$$p\left(wet_i\right) = \left(1 - p\left(we_i\right)\right)^{\frac{T}{N}} \tag{6.5}$$

where, T is the time for which word line is ON, $p\left(we_i\right)$ is the probability of a thermal noise event occurring in a window of N sec for degradation i. For an SRAM array has m marginal bits, the probability of all read and write unstable bits being detected is given by, probability of fault coverage $p(fc)$

$$p(fc) = \frac{\sum_{i=0}^{m} p\left(swt_i\right) + \sum_{j=0}^{m} p\left(wet_j\right)}{2m} \tag{6.6}$$

Theoretically, $p(fc)$ is always less than 1 (or 100%). A test methodology with $p(fc)$ close to the ideal value of 1 has a higher confidence of covering all unstable bits. Figure 6.13 shows the algorithm to estimate stochastic fault coverage for a given process.

Algorithm: Estimation of Stochastic Fault Coverage

Require: SigmaVtVariation, MemoryArraySize
TestCyclePeriod, ThermalEventWindow
1. size ← MemoryArraySize
2. sig ← SigmaVtVariation
3. T ← TestCyclePeriod
4. N ← ThermalEventWindow
5. Xr , Xw ← Set threshold for read and write unstable bit
6. **for** i = 0 to size-1 **do**
7. Generate bit cell with random variation r ϵ N(0,sig)
8. **if** r >Xr **then** add r to ListOfReadUnstableBits
9. **if** r >Xw **then** add r to ListOfWriteUnstableBits
10. **end for**
11. ndr ← size(ListOfReadUnstableBits)
12. ndw ← size(ListOfWriteUnstableBits)
13. **for** i = 0 to ndr-1 **do**
14. $p\left(swt_i\right) = 1 - \left(1 - p\left(sw_i\right)\right)^{\frac{T}{N}}$
15. ProbOfBitFlip = ProbOfBitFlip + $p\left(swt_i\right)$
16. **end for**
17. **for** i = 0 to ndw-1 **do**
18. $p\left(wet_i\right) = \left(1 - p\left(we_i\right)\right)^{\frac{T}{N}}$
19. ProbOfWrErr = ProbOfWrErr + $p\left(wet_i\right)$
20. **end for**
21. p(fc) = (ProbOfBitFlip + ProbOfWrErr) / (ndr + ndw)

Figure 6.13. Estimation of Stochastic Fault Coverage

6.2.2 Techniques to Improve Fault Coverage

N-Detect Memory Test methodology: N-detect test methodologies detect a single target fault multiple times [84]. Multiple detections are done using a single pattern or a

set of patterns. This increases fault coverage in combinational as well as sequential circuits. N-detect test techniques are also used to detect delay defects. Based on Equation 6.4, the probability of a marginally stable cell switching during the test sequence is a function of the time period T for which word line is turned ON. By accessing a single bit cell multiple times increases the ON period of word line and hence the probability of the bit cell switching. In theory even an infinite number of accesses cannot guarantee that the bit cell will switch. However, a bit cell which appears to be stable across a large number of accesses (N-detects) can be considered to be stable with a high degree of confidence. Similarly, in case of write, a successful write in the first attempt does not necessarily mean that a cell is write stable. A large number of writes increases the probability of observing a write error during test. As a trade-off to better fault coverage, N-detect tests increase test time as a function of N.

Multi-level Word Line Voltage during Test: Word Line (WL) boost techniques are used during SRAM write operation to improve write stability in degraded bit cells. Adaptive WL boost circuit to overcome write stability issue due to PVT variation and aging are widely used. Similar circuits can be used during test to boost the WL voltage. This accelerates instability in marginally stable cells and increases the probability of bit flip during read test. Similar to WL boost, multi-level WL drivers are proposed to improve read stability in degraded bit cells [74]. During SRAM read operation, the WL voltage is lowered to reduce the overdrive on access transistor and hence prevent the stored bit from flipping. However, reducing the WL voltage increases write errors. As a result, a reduced WL voltage during write tests increases the probability of inducing a write error in marginally stable bit cell and hence improves fault coverage. A common multi level WL voltage driver can be used to selectively boost or scale the WL voltage depending on read or write test. This requires additional test circuitry, but does not increase test time. Designing the WL

driver itself is not in the scope of this work. Implementations of multi-level WL boost circuits can be found in reference [74]. WL boost/scale may also lead to few stable cells being detected as unstable. These are known as false positive test. From a test perspective, it is safer to have false positive tests than false negatives, where an unstable cell is falsely regarded as being stable.

6.2.3 Implementation and Results

SRAM circuit design and stability analysis were performed using 32nm Predictive Technology Models (PTM). HSPICE was used to simulate the design with Transient Noise Analysis. Monte Carlo Simulations were performed for read and write stability analysis. The distribution of magnitude of thermal noise was completely dependent on the transistor models and the individual device sizes of the SRAM bit cell. MATLAB was used for all statistical analysis. All simulations were performed at 0.8V and corner temperature of 100C.

The probability of a random event, *p(sw)* causing bit flip during read is calculated in simulation for a time window of 10ps, Figure 6.14.

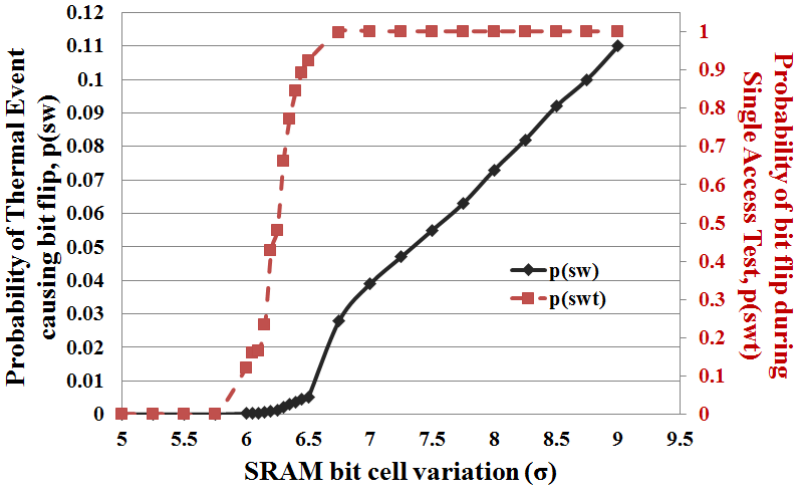


Figure 6.14. Probability of bit flip during single access test

The probabilities of a noise event large enough to cause stability issues in cells below 5σ are ignored. Using the calculated value, the probability *p(swt)* of a bitcell flipping in single test read cycle of 2ns, is calculated. It can be seen that marginally stable bits in the range of 6 to 6.5σ have a variable chance of being detected during a single read. Bit cells below 6σ variation are read stable and those above 6.5σ have a near 100% probability of being detected during test. A similar analysis is also performed to calculate the probability of a thermal event, *p(we)* that assists a successful write operation in a time window of 10ps, Figure 6.15.

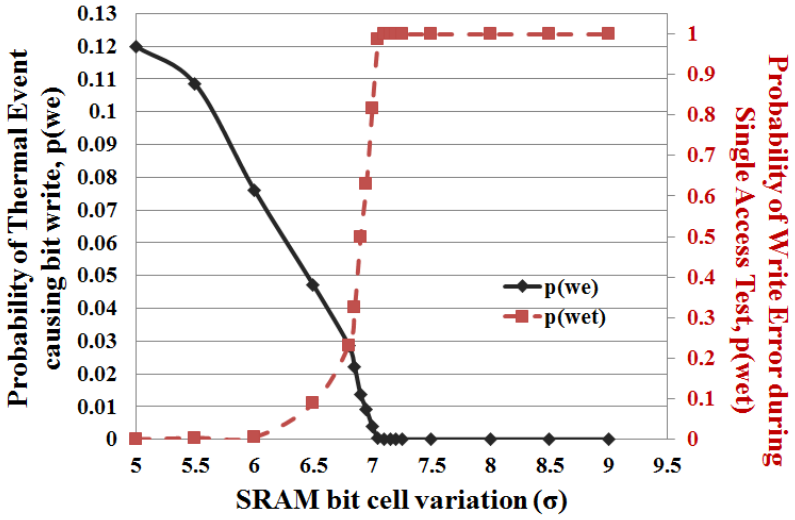
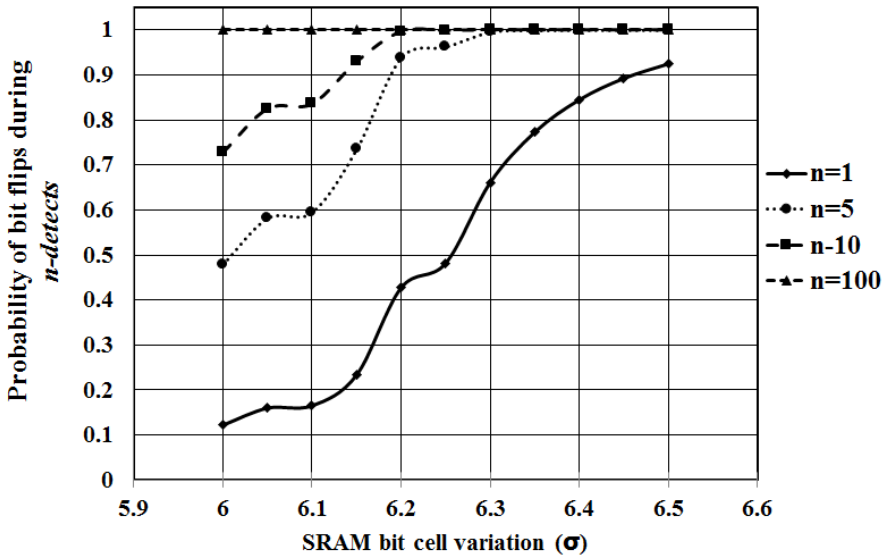


Figure 6.15. Probability of write error during single access test

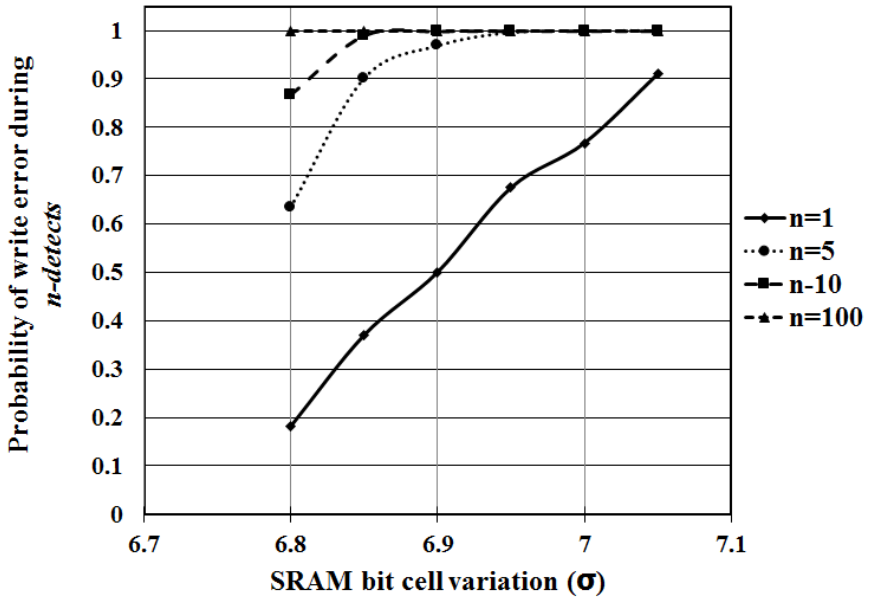
This value is used to estimate the probability of observing a write failure during a single write test procedure, *p(wet)*. It is important to note here that for write test, we estimate the probability of a thermal noise event that assists a successful write into a marginally stable cell not occurring during test. SRAM cell instances with pull-up to access transistor variation of 6.5σ to 7σ are marginally write stable. It is evident from the results that marginally stable bit cells could potentially be undetected during test and result in random bit errors during operation. As proposed in the previous section, accessing bit cells multiple times during test increases the probability of detecting bit

flip during read test or failure during write test. Figure 6.16 shows the results of N-detect test technique for read and write stability tests. It is seen that the probability of detecting marginally stable bit cells increases with the number of detects. For N=100, the probability of all unstable cells being detected approaches the ideal value of 1. However, the test time also increases proportionally with the number of detect. Selective Word Line voltage boosting or scaling improves stability fault coverage. Figure 6.17 shows the probability of bit flip or write error during a single read or write test cycle respectively. WL boost and scale help in increasing bit errors during test and hence detect majority of unstable bits. With additional circuitry, the same level of coverage is achieved as a large N-detect test.

To further validate the effect of thermal noise on SRAM stability, a 1KB SRAM array was simulated performing a single read and write access. A smaller array was chosen for simulation due to significant run-time of transient noise analysis. However, similar results and trends can be expected in case larger array sizes as well. The read and write operations were repeated across 100 unique noise samples. Figure 6.18 shows the distribution of bit errors detected across 100 samples. Depending on the noise sequence used, different numbers of bit cells were detected to be unstable, indicating the impact of thermal noise on SRAM arrays and uncertainty in conventional memory testing. Based on the statistical equations proposed and the number of noise sequences for which each bit cell was unstable, a probability of fault coverage is estimated. For a single access test and 100 noise sequences, the expected coverage is 0.88. However, with increasing number of test access, the probability of coverage 1, Figure 6.19. A similar analysis for WL voltage boosting during read test, Figure 6.20, shows that 100% coverage can be attained with a WL boost of 20mV. Further WL boost increases false positives leading to increased yield loss. The accuracy of these estimations increase with number of noise sequences used.

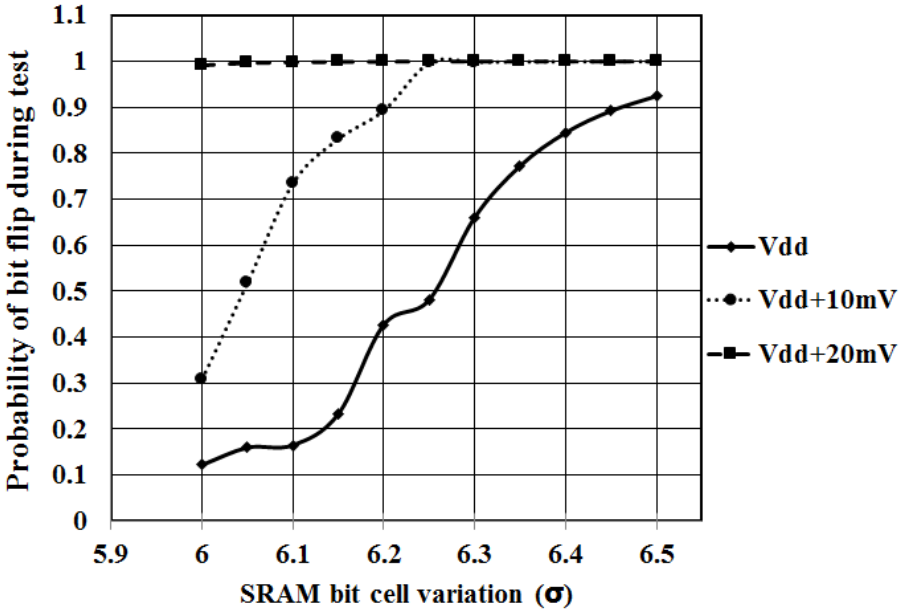


(a) Probability of bit flip during *n-detect*

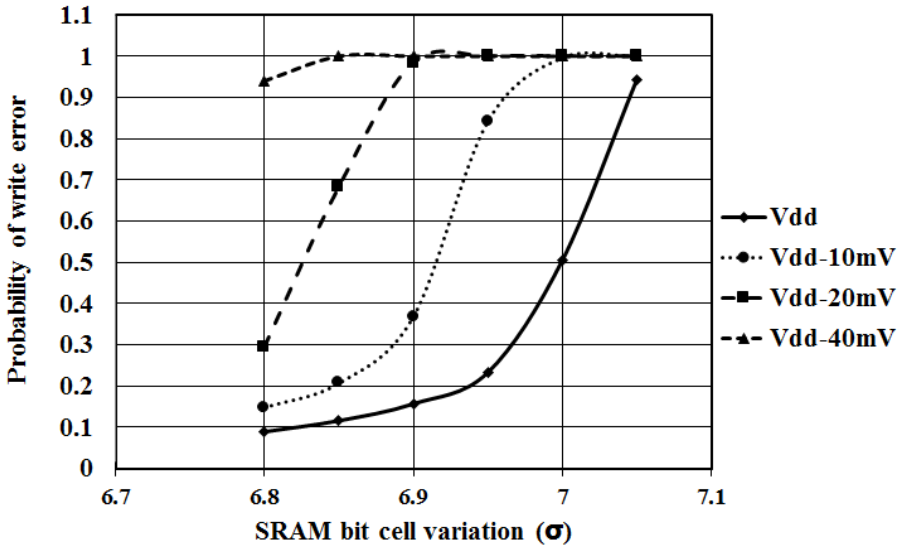


(b) Probability of write error during *n-detect*

Figure 6.16. Probability of Fault Detection using *n-detect*



(a) Probability of bit flip during using WL boosting



(b) Probability of write error using WL scaling

Figure 6.17. Probability of Fault Detection using Multi-level WL

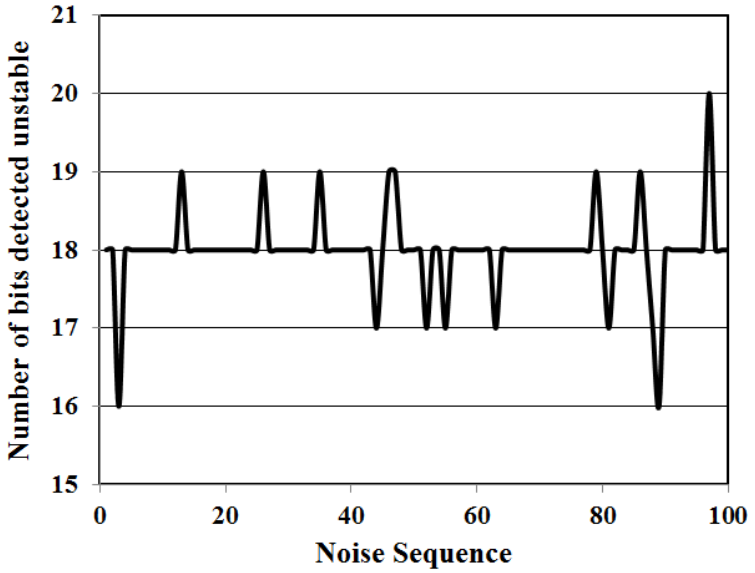


Figure 6.18. Varying fault coverage in 1kB SRAM array

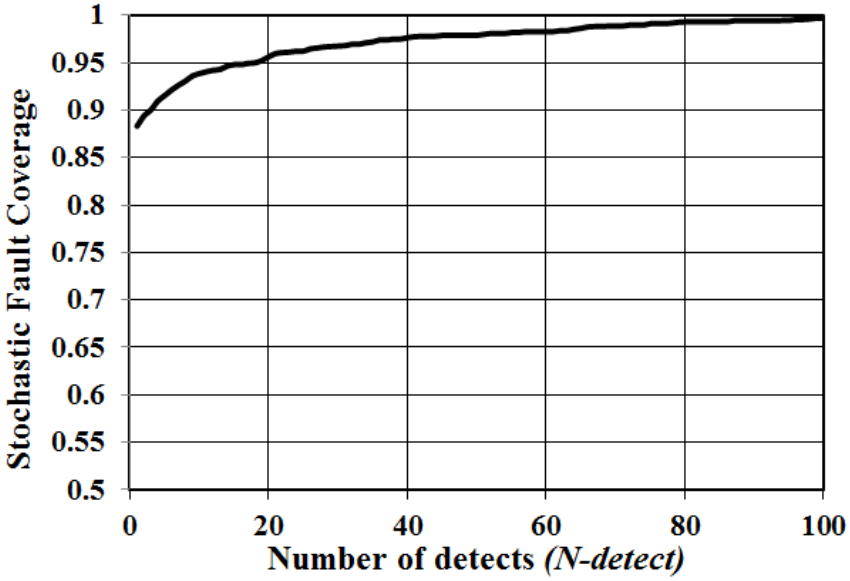


Figure 6.19. Probabilistic Fault Coverage with N-detect

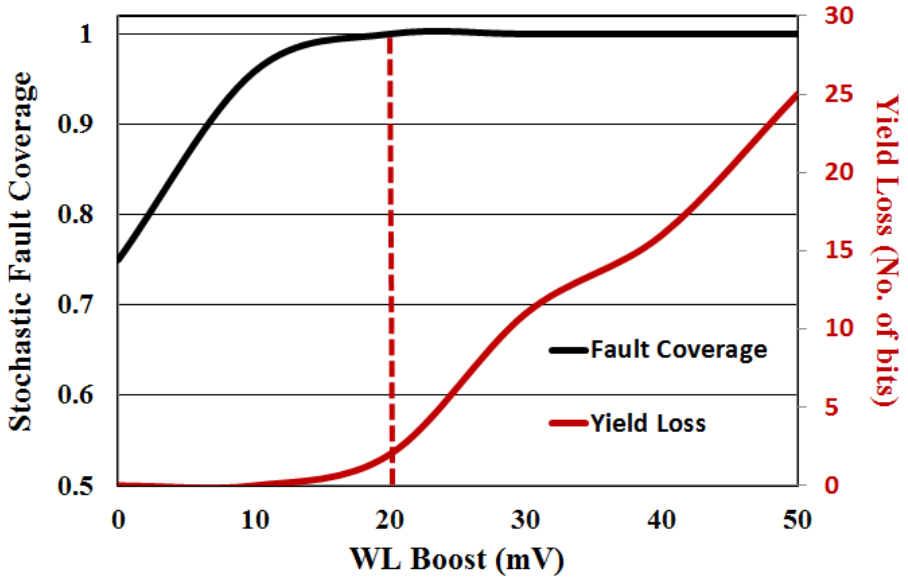


Figure 6.20. Probabilistic Read Fault Coverage with WL Boost

6.3 SRAM Test Coverage Uncertainty due to Random Telegraph Noise

A number of statistical and circuit techniques have been proposed to characterize RTN at the device level [53] and predict its impact on stability of SRAM cells [108]. However, it is important to study the impact of RTN on SRAM cells during test and quantize the uncertainty introduced into the fault coverage. In this section, we present an analysis of the impact of RTN during SRAM testing. The fluctuations in current in the SRAM devices during test can potentially mask unstable bitcells and result in false negatives. We also propose a stochastic fault coverage metric to account for the impact of RTN. All the analyses are based on single trap RTN events since the probability of observing multiple trap RTN events are negligible in advanced technology nodes. However, the proposed stochastic metrics can be extended for multiple RTN scenarios as well.

6.3.1 Impact of RTN on SRAM Testing

Figure 6.21 shows the impact of RTN on a 6T-SRAM bitcell. We consider the impact of Id variation in transistors M1 and M5 for read test and transistors M4 and M6 for write test. The circuit was simulated using 32nm Predictive Technology Models at 30C and 0.8V for an read cycle of 200us, Figure 6.22. The V_{th} variation on transistor M1 is modeled to be 6.4σ. It can be seen from the result that the bitcell appears to be stable during the initial part of the simulation.This is when the trap in transistor M1 is empty or M1 is in the low Vth state. However, once the transistor M1 goes into capture state, the drain current decreases (or threshold voltage increases) leading to a bit flip. During testing, there is no guarantee that the worst case RTN scenario will be observed in the finite number of read operations performed. This will result in false negatives and hence a loss in fault coverage. A similar phenomenon

can be observed during write test when transistor M6 changes from empty state to captured state, resulting in write error.

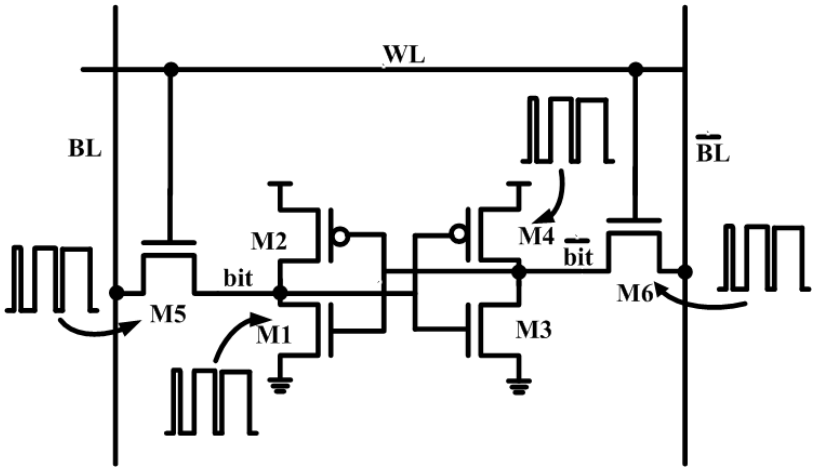


Figure 6.21. Impact of RTN on SRAM bitcell

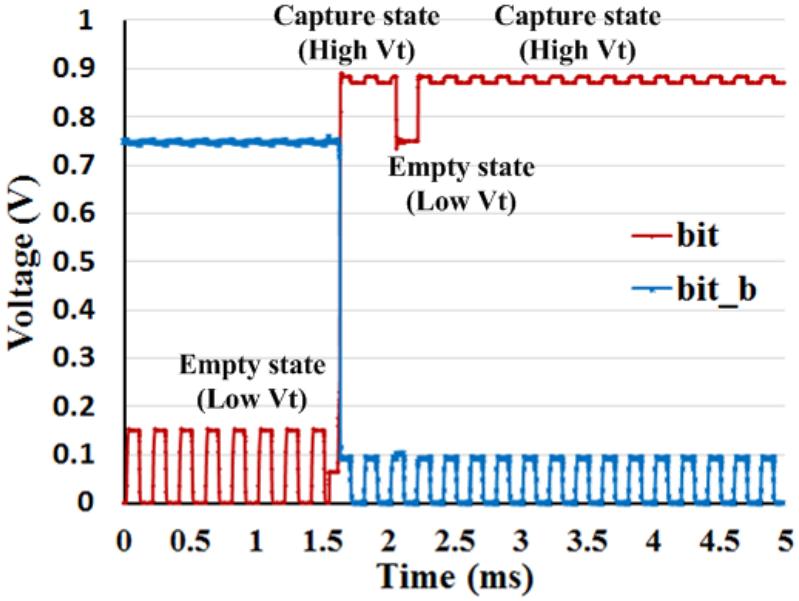


Figure 6.22. Impact of RTN during read test

Figure 6.23 shows the magnitude of RTN noise required in transistor M1 (assuming RTN noise = 0 in M5; worst case noise condition) to cause a bit flip for different variations in M1. Variations less than 6σ require large fluctuations in current due

to RTN to cause bit flips. The probability of these fluctuations occurring can be considered to be zero for all practical purposes. Variations larger than 7.2σ are highly unstable bits. These bitcells are not affected by RTN and will always result in a bit flip. The bitcells with variations in V_{th} ranging from 6σ to 7.2σ have a probability of being unstable if the magnitude of RTN noise is greater than the required threshold value. We define these bitcells as *Marginally Stable* if the bit flips or write errors are affected by RTN. During read test, the transistor pair of M1 and M5 may be in one of the following states, assuming 1 indicates the trap is filled (captured mode/high Vth mode) and 0 indicates the trap is empty (emit mode/low Vth mode):

1. State 0: M1=0 and M5=0 - Bit-flips will be observed only in unstable bits.
2. State 1: M1=0 and M5=1 - Bit flips may not be observed in marginally stable bits; false negatives.
3. State 2: M1=1 and M5=0 - Bit flips will be observed in both unstable and marginally stable bits.
4. State 3: M1=1 and M5=1 - Bit flips may not be observed in marginally stable bits; false negatives.

State 0 is equivalent to zero RTN noise in the circuit. State 2 with M1=1 and M5=0 is the ideal state to achieve maximum fault coverage. Fault coverage is the least in state 1 where transistor M5 is in the high Vth state. In state 3, the probability of observing a fault during test will depend on the magnitude of $\Delta Id/Id$ of transistor M5 for a given $\Delta Id/Id$ in M1. Similarly, transistors M4 and M6 will be in one of the four possible noise states during write test, resulting in a corresponding probability of observing write errors.

Stochastic Fault Coverage in the Presence of Random Telegraph Noise: In this section we present a stochastic metric for fault coverage that incorporates the impact of RTN

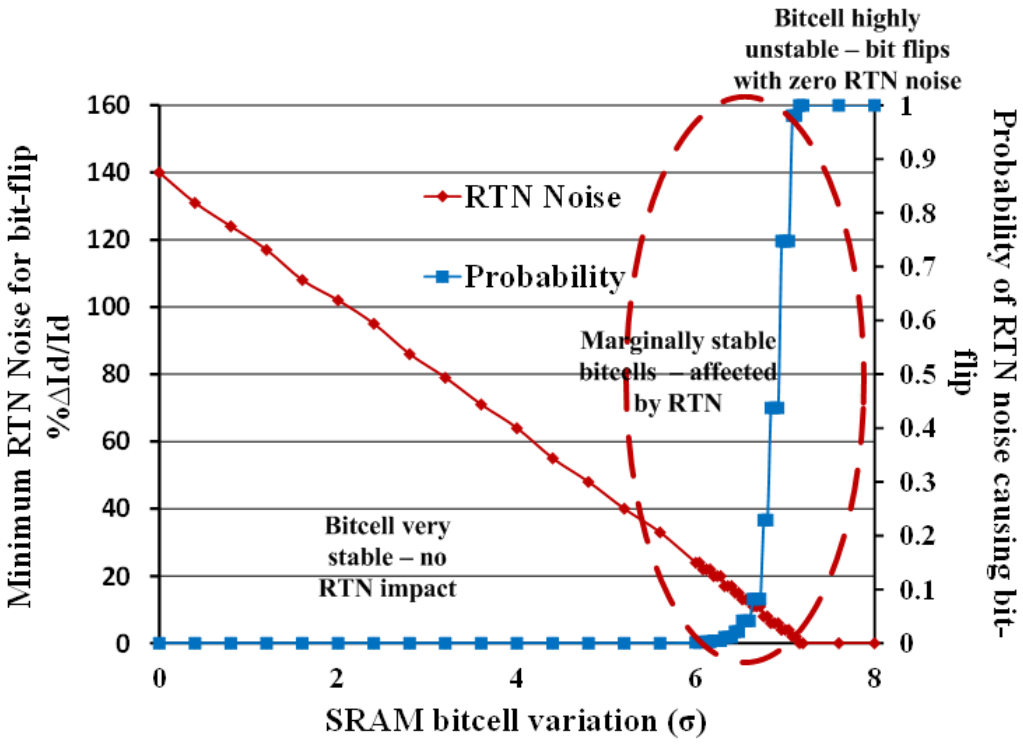


Figure 6.23. Probability of bit flip due to RTN

on SRAM stability along with the probability of observing a fault during test. Figure 6.23 indicates that bitcells with variation ranging from 6σ to 7.2σ may be marginally stable . The worst case noise condition is when the bitcell is in state 2. Therefore, a bitcell with a threshold voltage variation in this range will fail during normal operation if the magnitudes of $\Delta Id/Id$ in transistors M1 is greater than the minimum noise value shown in Figure 6.23. For instance, a bitcell with 6.5σ variation is unstable only if the RTN trap location in transistor M1 produces a $\Delta Id/Id > 14\%$ (assuming $\Delta Id/Id$ of M5 $= 0$). In other words, a bitcell with 6.5σ variation has a probability of 4% of being unstable. Unlike thermal noise, for a given device, the magnitude of RTN can fluctuate only between zero and a fixed $\Delta Id/Id$. Therefore, the expected number of faulty bits $E(FB)$ in a N-bit SRAM array, for a given process corner is given by,

$$E\left(FB\right) = \sum_{i=1}^{N} Prob\left(\Delta Id/Id\left(M1\right) \geq RTN_{th}\left(\sigma_{Vth}\left(i\right)\right)\right) \tag{6.7}$$

During test, the bitcell may be in one of the four noise states. Accordingly, the probability of observing a bit flip during read test or error during write test will depend on the probability of the corresponding minimum values of RTN noise in transistors M1 and M5. For a bitcell with variation i and states with M5=1, the probability of observing a fault during test depends on the magnitude of $\Delta Id/Id$ in transistor M1 for a given value of $\Delta Id/Id$ in M5. Higher the noise in M5, larger will be the RTN noise required in M1 to observe a fault. The probability of observing a fault for states M5=1, $P(fault)$, is given by,

$$\sum_{n=0}^{0.5} P\left(\frac{\Delta Id}{Id}\left(M5\right) = n\right) * P\left(\frac{\Delta Id}{Id}\left(M1\right) \geq RTN_{th}\left(\sigma_{Vth}\left(i\right)\right) | \frac{\Delta Id}{Id}\left(M5\right) = n\right) \tag{6.8}$$

The above equation is valid for any noise state since $Prob\left(\Delta Id/Id\left(M1\right)\right) > 0$ is equal to zero for states 0 and 1 and $Prob\left(\Delta Id/Id\left(M5\right)\right) > 0$ is equal to zero for states 0 and 2. The expected expected number of faulty bits during test $E(FBT)$ in a N-bit SRAM array is given by,

$$E\left(FBT\right) = \sum_{i=1}^{N} \sum_{j=0}^{3} P(State = j) * P(fault|i,j) \tag{6.9}$$

The expected fault coverage for read and write unstable bits are each given by,

$$E\left(fc\right) = \frac{E\left(FBT\right)}{E\left(FB\right)} \tag{6.10}$$

Figure 6.24 and Figure 6.25 show the expected probability of faulty bits across different threshold voltage variations and the corresponding expected probability of observing a faulty bit during test. The results are based on SRAM bitcells designed in using 32nm PTM models and operating at 0.8V and 30C. As expected, highly

stable and unstable bits are always detected. However, marginally stable bits that depend on the state and magnitude of RTN noise result in the loss of fault coverage.

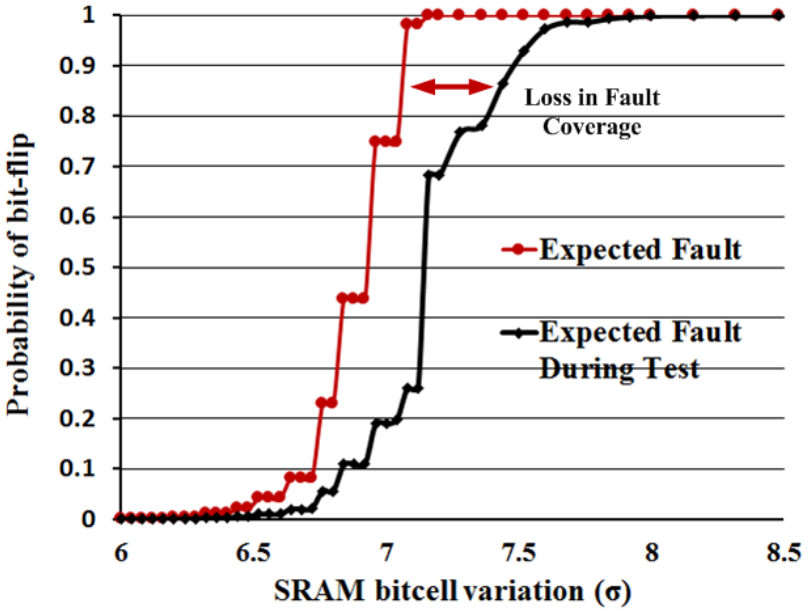


Figure 6.24. Expected Fault Coverage during read Test

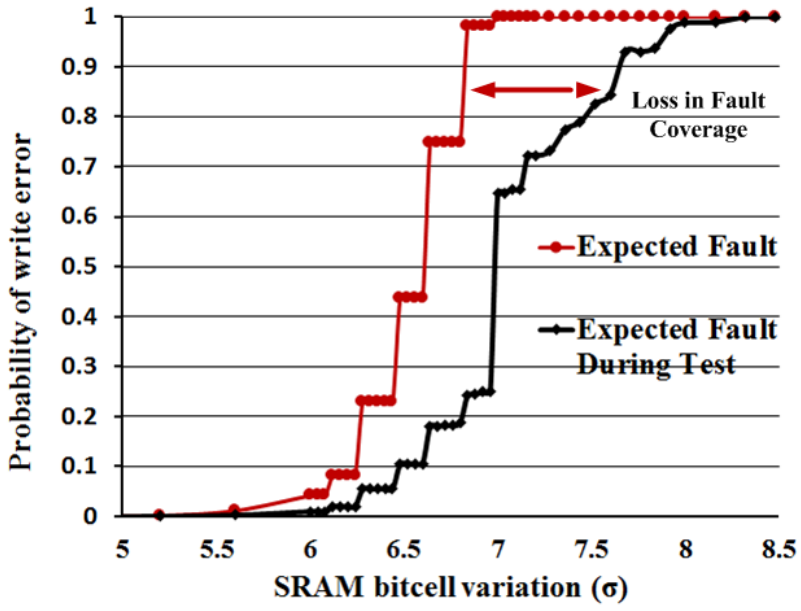


Figure 6.25. Expected Fault Coverage during Write Test

6.3.2 Techniques to Improve Fault Coverage:

The fault coverage of SRAM arrays depend on the detection of marginally stable bits. Faults in marginally stable bits are observed only under specific thermal noise events or RTN noise states. In this section, we present two techniques to increase fault coverage in the presence of on-chip random noise during test.

N-Detect Memory Test methodology: In case of RTN, the best noise condition for test is state 2, where transistor M1 is in capture state and M5 is in emit/empty state. Similarly, the best noise state for write test is when M6 is in capture state and M4 is in emit/empty state. If τ_c and τ_e are the time constants for capture and emit states, the probability of the bitcell being in state 2 during test is,

$$P\left(state = 2\right) = \frac{\tau_c}{\tau_c + \tau_e} * \frac{\tau_e}{\tau_c + \tau_e} \tag{6.11}$$

The probability of the bitcell being in state 2 after n-detects is,

$$P\left(state = 2\right) + \sum_{i=2}^{n} \left(1 - P\left(state_{i-1} = 2\right)\right) * P\left(state_i = 2\right) \tag{6.12}$$

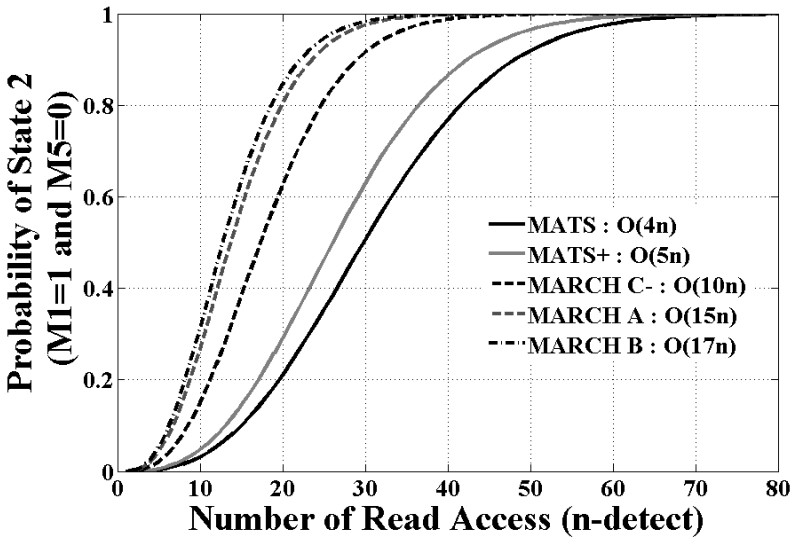


Figure 6.26. Probability of bit flip with N-detect

With increase in the value of n, the probability of the bitcell being in the state 2 during test approaches unity. The number of accesses is a trade-off between the expected fault coverage and the increase in test time. Since RTN is a low frequency noise, reading or writing into the same bitcell in two consecutive cycles does not provide significant improvement in fault coverage. Further, as discussed in section II(B), the most commonly used test methodology for SRAM arrays are variants of the March test. In a March test, the addresses of an SRAM array are incremented (or decremented) before each read/write access. As a result, the period between two read operations for a single bitcell using N-detect depends on the kind of March test used and the size of the SRAM array. For instance, the basic March test has a test time of O(4n), where n is the number of bytes assuming byte access during read. This implies that for a 1kB SRAM array with an access cycle of 10ns, the time between two consecutive reads from a given bitcell are 40.96μs. Assuming the bitcell was initially in the noise state 0, the probability of detecting a fault during the second read is equal to the probability of the bitcell changing to the noise state 2 within 8.19μs. Since the capture and emit time constants of RTN are of the order of few milliseconds, the probability of seeing any change after the second read iteration is not significantly large. Therefore, the number or read/write accesses required for N-detect also depends on the time interval between two consecutive accesses. Figure 6.26 show the number of detects required for a 1kB SRAM array with different versions of March algorithm, each having a different O(n) runtime. These trends will change as the array size increases.

Multi-level Word Line Voltage during Test: . In the case of impact of RTN, scaling the WL voltage acts like an increased V_{th} state of the access transistor (M6); there by increasing the probability of detecting a marginally stable cell. Scaling or boosting of WL voltage helps in detecting unstable bits in a single cycle without increasing the test time. However, multi-level WL voltage during test also increases the probability

of $FalsePositives$, where perfectly stable bitcells may result in bit flip or write errors due to change in WL voltage. The ΔV of boost or scale in WL voltage is a trade-off between the expected fault coverage and the yield loss due to multi-level WL. Figure 6.17a shows the probability of bit flips using WL scaling during read test.

6.3.3 Implementation and Results

The proposed stochastic metric and techniques to improve fault coverage were analyzed using a sample SRAM array with $\sigma_{Vth} = 0.1V_{th}$. SRAM circuit design and stability analysis were performed using 32nm Predictive Technology Models (PTM). The simulations for Random Telegraph Noise were performed using NGSPICE with the in-built transient noise simulator. Capture and emit times were generated using an exponential distribution. The magnitude of RTN, modeled as a fluctuation in transistor current ($\%\Delta Id/Id$), followed a log normal distribution with $\mu_{log} = 1.71$ and $\sigma_{log} = 0.89$ [26]. MATLAB was used for all statistical analysis. All simulations were performed at 0.8V. Analysis of thermal noise was performed at 100C to mimic worst case thermal noise condition and the analysis for RTN was performed at nominal temperature of 30C since RTN is a temperature independent phenomenon [110]. The size of the SRAM array and the number of unique noise sequences considered for analysis were limited by the extensive runtime of transient noise simulations.

To study the impact of RTN during SRAM testing, 10000 instances of the 1kB SRAM array were simulated, each with unique values of $\%\Delta Id/Id$. Figure 6.27 shows the percentage fault coverage during read test for different noise sequences. Since RTN noise values are constant for a given bitcell, there may be scenarios with zero marginally stable bits in an SRAM array. In such cases, 100% fault coverage is achieved irrespective of the RTN noise state during test. Figure 6.28 shows the stochastic fault coverage for a 1kB SRAM with a read access time of 10ns using

different March algorithms. As expected, larger time interval between two consecutive accesses will require fewer number of detects to achieve 100% fault coverage.

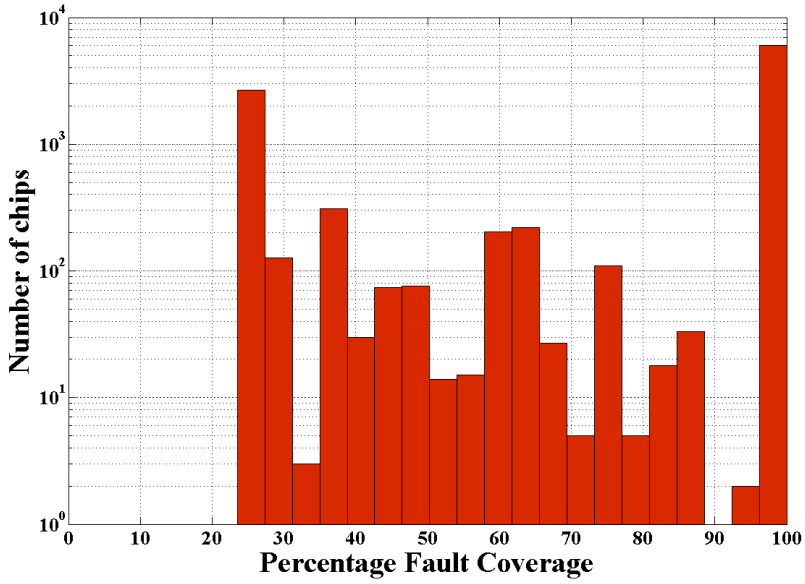


Figure 6.27. Varying fault coverage during read test in 1kB SRAM array

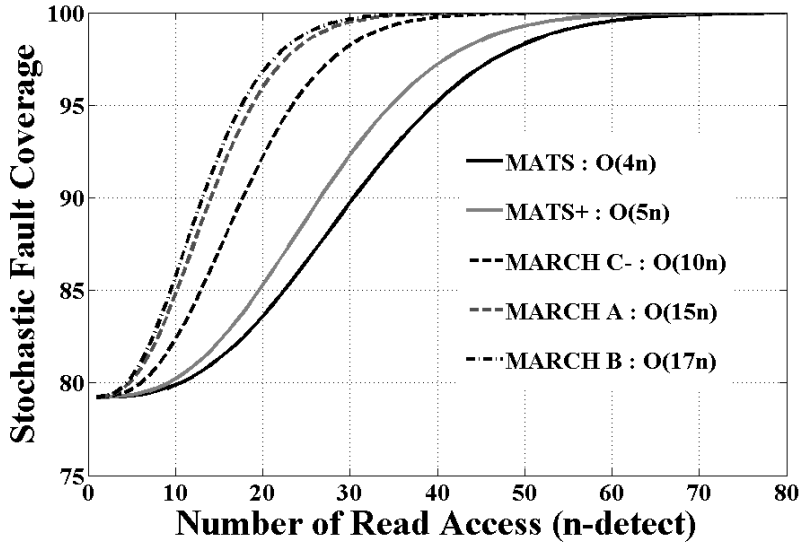


Figure 6.28. Stochastic read fault coverage with N-detect

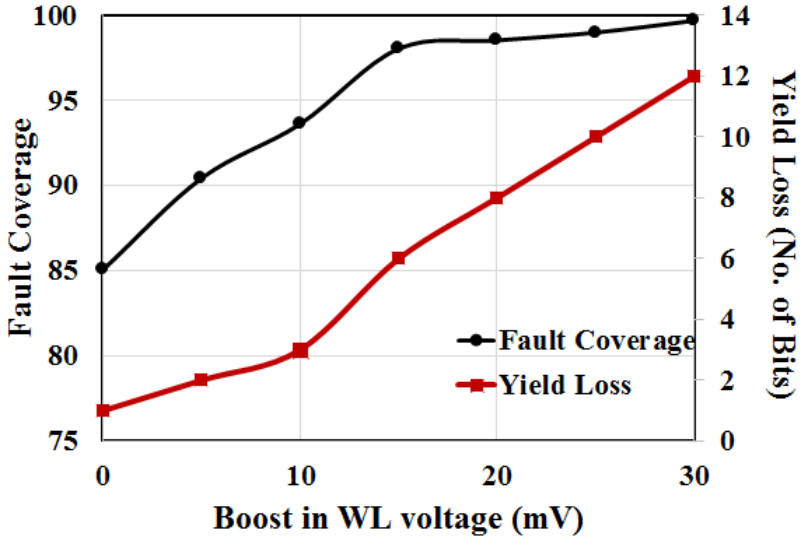


Figure 6.29. Probabilistic Read Fault Coverage with WL Boost

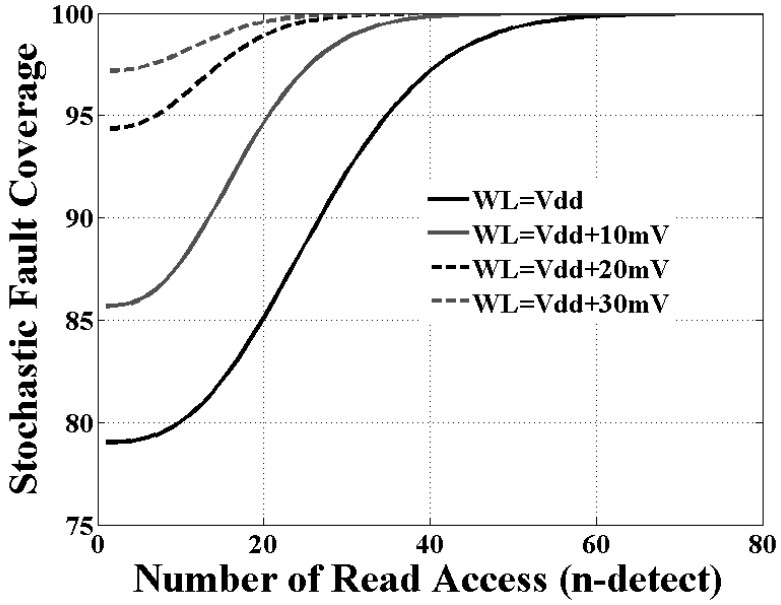


Figure 6.30. Probabilistic Read Fault Coverage with combination of N-detect and WL Boost

Figure 6.29 shows the fault coverage for different values of WL voltage boost during read test and the corresponding yield loss. The fault coverage approaches 100% for a WL voltage boost of 30mV with a penalty of 12-bit yield loss. Combining

both the N-detect and WL boost technique provides better fault coverage with fewer detects, Figure 6.30. Similar to thermal noise, this is a trade-off between the test time and yield loss. The above experiments were also performed for write test on a 1kB SRAM array to analyze the impact of RTN.

CHAPTER 7

CONCLUSION

Semiconductor industry and CMOS technology have made significant contributions in the fields of communication, security, transportation and medical devices. The ways of human lives have changed with the advent of mobile computing. As CMOS technology heads towards its physical limits, it provides tremendous opportunities in circuit design to develop novel techniques for improving power and performance of Integrated Circuits. The major challenge faced by designers today are process variation and on-chip noise. With no viable post-CMOS device/fabric available for large volume manufacturing, circuit designers have to develop new techniques and methodologies to counter the effect of variations and noise. Further, the paradigm shift from high performance computing to energy efficient computing requires the variation and noise compensation techniques to be ultra-low power and energy efficient.

It is important to understand the impact of variations depending on the circuit and application. Circuit designs have to be variation aware and perform post-Si tuning to improve performance and power yield. With increasing standard deviation in transistor parameters, over designing the circuits for worst-case variations is no longer an option. Adaptive body bias, digital circuit tuning and clock skew re-distribution techniques have to be used to achieve optimum performance with minimal power overhead. Statistical analyses have to be used to accurately size and design circuits for post-Si tuning. Sensitive circuits like metastability based TRNG, which sample minute voltage/current fluctuations due to noise, are adversely affected by intra-die variations. Unconventional circuit techniques like using metastability resolution

time instead of the resolution state can provide tolerance to process variation, thereby eliminating the need for post-Si tuning or additional post-processing. Error correction techniques can no longer be uniform across all fabricated chip. Post-processing and error correction techniques have to be adaptable on a die-to-die basis depending on the variations. The stochastic models developed for circuits like TRNG provide an efficient platform to determine the expected impact of variation and accordingly choose the necessary error correction. These models can be extended to conventional circuits like SRAM and sense amplifiers. The logic used for post-processing also need to be re-configurable to achieve high reliability in the presence of process variation, device aging and potential fault attacks. Designing variation tolerant circuits and post-Si adaptive tuning require accurate sensing and characterization of the sources of these variations. Since CMOS devices experience variations in transistor parameters throughout their lifetime, lightweight sensors are essential in monitoring the wear-out effects on circuit performance and reliability. These sensor circuits are affected by process variation. Therefore, designing variation tolerant sensor circuits provide accurate sensing and help in increasing the longevity of CMOS circuits.

While on-chip random noise sources can be leveraged in applications like TRNG, they are deterrent to reliability and performance of critical circuits like SRAM. The random nature of these noise sources require statistical metrics to estimate the error rates and develop appropriate counter measures. Faulty circuits that are prone to the impact of thermal noise and Random Telegraph Noise (RTN) cannot be guaranteed to be detected during post-Si test. Conventional fault coverage metrics for SRAM cells do not account for the loss of coverage in these marginally stable cells. Stochastic fault coverage metrics have to be developed to accurately estimate the expected fault coverage. This presents a better picture of the impact of noise during silicon testing. Existing test methodologies like N-detect and stability enhancement techniques like

multi-level WordLine voltage can be modified to increase the expected fault coverage and hence the reliability of manufactured products.

Current CMOS fabrication technology will be the foundation of the semiconductor industry for the foreseeable future. Novel circuit and architectural solutions have to be developed to continue on the path of designing high density designs, operating at GHz frequencies and with increasing energy efficiency. The numerous challenges in managing and leveraging variations and noise promise an exciting phase in the field of nano-CMOS circuit design.

BIBLIOGRAPHY

[1] PKCS #1 v2.1: RSA encryption standard, 2002.

[2] Agarwal, A., Paul, B.C., Mahmoodi, H., Datta, A., and Roy, K. A process-tolerant cache architecture for improved yield in nanoscale technologies. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems 13*, 1 (Jan. 2005), 27 –38.

[3] Agarwal, A., Paul, B.C., Mukhopadhyay, S., and Roy, K. Process variation in embedded memories: failure analysis and variation aware architecture. *IEEE Journal of Solid-State Circuits 40*, 9 (Sept. 2005), 1804 – 1814.

[4] Baumann, R. The impact of technology scaling on soft error rate performance and limits to the efficacy of error correction. In *Electron Devices Meeting, 2002. IEDM '02. International* (2002), pp. 329 – 332.

[5] Bayon, Pierre, Bossuet, Lilian, Aubert, Alain, Fischer, Viktor, Poucheret, Franois, Robisson, Bruno, and Maurine, Philippe. Contactless electromagnetic active attack on ring oscillator based true random number generator. In *Constructive Side-Channel Analysis and Secure Design*, Werner Schindler and Sorin A. Huss, Eds., no. 7275 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, Jan. 2012, pp. 151–166.

[6] Bayrak, Ali Galip, Velickovic, Nikola, Ienne, Paolo, and Burleson, Wayne. An architecture-independent instruction shuffler to protect against side-channel attacks. *ACM Trans. Archit. Code Optim. 8*, 4 (Jan. 2012), 20:1–20:19.

[7] Blaauw, D., Chopra, K., Srivastava, A., and Scheffer, L. Statistical timing analysis: From basic principles to state of the art. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 27*, 4 (Apr. 2008), 589–607.

[8] Bochard, N., Bernard, F., and Fischer, V. Observing the randomness in RO-based TRNG. In *International Conference on Reconfigurable Computing and FPGAs, 2009. ReConFig '09* (Dec. 2009), pp. 237 –242.

[9] Bogdanov, A., Knudsen, L. R., Leander, G., Paar, C., Poschmann, A., Robshaw, M. J. B., Seurin, Y., and Vikkelsoe, C. PRESENT: An ultra-lightweight block cipher. In *Cryptographic Hardware and Embedded Systems - CHES 2007*, Pascal Paillier and Ingrid Verbauwhede, Eds., no. 4727 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, Jan. 2007, pp. 450–466.

[10] Borkar, Shekhar, Karnik, Tanay, Narendra, Siva, Tschanz, Jim, Keshavarzi, Ali, and De, Vivek. Parameter variations and impact on circuits and microarchitecture. In *Proceedings of the 40th Annual Design Automation Conference* (New York, NY, USA, 2003), DAC '03, ACM, pp. 338–342.

[11] Bowman, K.A., Duvall, S.G., and Meindl, J.D. Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration. *IEEE Journal of Solid-State Circuits 37*, 2 (Feb. 2002), 183–190.

[12] Brederlow, R., Prakash, R., Paulus, C., and Thewes, R. A low-power true random number generator using random telegraph noise of single oxide-traps. In *Solid-State Circuits Conference, 2006. ISSCC 2006. Digest of Technical Papers. IEEE International* (Feb. 2006), pp. 1666 –1675.

[13] Burnett, D., Erington, K., Subramanian, C., and Baker, K. Implications of fundamental threshold voltage variations for high-density SRAM and logic circuits. In *1994 Symposium on VLSI Technology, 1994. Digest of Technical Papers* (June 1994), pp. 15 –16.

[14] Callegari, S., Rovatti, R., and Setti, G. Embeddable ADC-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos. *IEEE Transactions on Signal Processing 53*, 2 (Feb. 2005), 793–805.

[15] Campbell, J.P., Qin, J., Cheung, K.P., Yu, L.C., Suehle, J.S., Oates, A., and Sheng, K. Random telegraph noise in highly scaled nMOSFETs. In *Reliability Physics Symposium, 2009 IEEE International* (Apr. 2009), pp. 382–388.

[16] Chakraborty, A., Duraisami, K., Sathanur, A., Sithambaram, P., Benini, L., Macii, A., Macii, E., and Poncino, M. Dynamic thermal clock skew compensation using tunable delay buffers. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems 16*, 6 (2008), 639–649.

[17] Cheng, B., Roy, S., and Asenov, A. The impact of random doping effects on CMOS SRAM cell. In *Solid-State Circuits Conference, 2004. ESSCIRC 2004. Proceeding of the 30th European* (Sept. 2004), pp. 219 – 222.

[18] Cherkaoui, A., Fischer, V., Aubert, A., and Fesquet, L. A self-timed ring based true random number generator. In *2013 IEEE 19th International Symposium on Asynchronous Circuits and Systems (ASYNC)* (2013), pp. 99–106.

[19] Cicek, I., and Dundar, G. A hardware efficient chaotic ring oscillator based true random number generator. In *2011 18th IEEE International Conference on Electronics, Circuits and Systems (ICECS)* (Dec. 2011), pp. 430–433.

[20] Dekker, R., Beenker, F., and Thijssen, L. Fault modeling and test algorithm development for static random access memories. In *Test Conference, 1988. Proceedings. New Frontiers in Testing, International* (Sept. 1988), pp. 343 – 352.

[21] Dekker, R., Beenker, F., and Thijssen, L. A realistic fault model and test algorithms for static random access memories. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 9*, 6 (June 1990), 567 –572.

[22] Dichtl, Markus. Bad and good ways of post-processing biased physical random numbers. In *Fast Software Encryption*, Alex Biryukov, Ed., no. 4593 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, Jan. 2007, pp. 137–152.

[23] Donoho, D.L. Compressed sensing. *IEEE Transactions on Information Theory 52*, 4 (Apr. 2006), 1289–1306.

[24] Drutarovsky, M., and Galajda, P. A robust chaos-based true random number generator embedded in reconfigurable switched-capacitor hardware. In *Radioelektronika, 2007. 17th International Conference* (Apr. 2007), pp. 1 –6.

[25] Eberlein, M., and Abu Bakar, R. An integrated channel noise-based true random number generator. In *7th International Conference on ASIC, 2007. ASICON '07* (Oct. 2007), pp. 391 –394.

[26] Fan, M.-L., Hu, V. P.-H., Chen, Y.-N., Su, P., and Chuang, C.-T. Analysis of single-trap-induced random telegraph noise on FinFET devices, 6t SRAM cell, and logic circuits. *IEEE Transactions on Electron Devices 59*, 8 (Aug. 2012), 2227 –2234.

[27] Grossar, E., Stucchi, M., Maex, K., and Dehaene, W. Read stability and writeability analysis of SRAM cells for nanometer technologies. *IEEE Journal of Solid-State Circuits 41*, 11 (Nov. 2006), 2577 –2588.

[28] Guthaus, M.R., Wilke, G., and Reis, R. Non-uniform clock mesh optimization with linear programming buffer insertion. In *2010 47th ACM/IEEE Design Automation Conference (DAC)* (2010), pp. 74–79.

[29] Hamburg, Mike, Kocher, Paul, and Marson, Mark. Analysis of intels ivy bridge digital random number generator, Mar. 2012.

[30] Hamdioui, S., van de Goor, A.J., and Rodgers, M. March SS: a test for all static simple RAM faults. In *Proceedings of the 2002 IEEE International Workshop on Memory Technology, Design and Testing, 2002. (MTDT 2002)* (2002), pp. 95 – 100.

[31] Hazucha, P., and Svensson, C. Circuit technique for accurate soft error rate measurements. In *Solid-State Circuits Conference, 1999. ESSCIRC '99. Proceedings of the 25th European* (Sept. 1999), pp. 190 –193.

[32] Heald, R., and Wang, P. Variability in sub-100nm SRAM designs. In *IEEE/ACM International Conference on Computer Aided Design, 2004. ICCAD-2004* (Nov. 2004), pp. 347 – 352.

[33] Holcomb, D.E., Burleson, W.P., and Fu, K. Power-up SRAM state as an identifying fingerprint and source of true random numbers. *IEEE Transactions on Computers 58*, 9 (Sept. 2009), 1198 –1210.

[34] Hotoleanu, D., Cret, O., Suciu, A., Gyorfi, T., and Vacariu, L. Real-time testing of true random number generators through dynamic reconfiguration. In *2010 13th Euromicro Conference on Digital System Design: Architectures, Methods and Tools (DSD)* (Sept. 2010), pp. 247 –250.

[35] Huang, Chien-Yuan, Shen, Wen Chao, Tseng, Yuan-Heng, King, Ya-Chin, and Lin, Chrong-Jung. A contact-resistive random-access-memory-based true random number generator. *IEEE Electron Device Letters 33*, 8 (Aug. 2012), 1108 –1110.

[36] Ito, Takashi, and Okazaki, Shinji. Pushing the limits of lithography. *Nature 406*, 6799 (Aug. 2000), 1027–1031.

[37] Jess, J. A G, Kalafala, K., Naidu, S.R., Otten, R.H.J., and Visweswariah, C. Statistical timing for parametric yield prediction of digital integrated circuits. In *Design Automation Conference, 2003. Proceedings* (June 2003), pp. 932–937.

[38] Johnson, J. B. Thermal agitation of electricity in conductors. *Physical Review 32*, 1 (July 1928), 97–109.

[39] Kaczer, B., Grasser, T., Roussel, P.J., Franco, J., Degraeve, R., Ragnarsson, L. A, Simoen, E., Groeseneken, G., and Reisinger, H. Origin of NBTI variability in deeply scaled pFETs. In *Reliability Physics Symposium (IRPS), 2010 IEEE International* (May 2010), pp. 26–32.

[40] Karnik, T., De, V., and Borkar, S. Statistical design for variation tolerance: key to continued moore's law. In *International Conference on Integrated Circuit Design and Technology, 2004. ICICDT '04* (2004), pp. 175–176.

[41] Kasper, Timo, Silbermann, Michael, and Paar, Christof. All you can eat or breaking a real-world contactless payment system. In *Financial Cryptography and Data Security*, Radu Sion, Ed., no. 6052 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, Jan. 2010, pp. 343–350.

[42] Kelsey, John, Schneier, Bruce, Wagner, David, and Hall, Chris. Cryptanalytic attacks on pseudorandom number generators. In *FAST SOFTWARE ENCRYPTION, FIFTH INTERNATIONAL PROCEEDINGS* (1998), Springer-Verlag, pp. 168–188.

[43] Khandelwal, V., and Srivastava, A. Variability-driven formulation for simultaneous gate sizing and postsilicon tunability allocation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 27*, 4 (2008), 610–620.

[44] Kim, C.H., Hsu, S., Krishnamurthy, R., Borkar, S., and Roy, K. Self calibrating circuit design for variation tolerant VLSI systems. In *On-Line Testing Symposium, 2005. IOLTS 2005. 11th IEEE International* (July 2005), pp. 100–105.

[45] Kim, Ilyoung, Zorian, Y., Komoriya, G., Pham, H., Higgins, F.P., and Lewandowski, J.L. Built in self repair for embedded high density SRAM. In *Test Conference, 1998. Proceedings., International* (Oct. 1998), pp. 1112 –1119.

[46] Kim, Tae-Hyoung, Persaud, R., and Kim, C.H. Silicon odometer: An on-chip reliability monitor for measuring frequency degradation of digital circuits. *IEEE Journal of Solid-State Circuits 43*, 4 (2008), 874–880.

[47] Kish, Laszlo B. End of moore's law: thermal (noise) death of integration in micro and nano electronics. *Physics Letters A 305*, 34 (Dec. 2002), 144–149.

[48] Ko, Miyong, and Goeckel, D.L. Wireless physical-layer security performance of UWB systems. In *Military Communications Conference, 2010 - MILCOM 2010* (Nov. 2010), pp. 2143 –2148.

[49] Kobenge, S.B., and Yang, Huazhong. A power efficient digitally programmable delay element for low power VLSI applications. In *1st Asia Symposium on Quality Electronic Design, 2009. ASQED 2009* (2009), pp. 83–87.

[50] Kuhn, K.J., Giles, M.D., Becher, D., Kolar, P., Kornfeld, A., Kotlyar, R., Ma, S.T., Maheshwari, A., and Mudanai, S. Process technology variation. *IEEE Transactions on Electron Devices 58*, 8 (Aug. 2011), 2197–2208.

[51] Kumar, S.V., Kim, K.H., and Sapatnekar, S.S. Impact of NBTI on SRAM read stability and design for reliability. In *7th International Symposium on Quality Electronic Design, 2006. ISQED '06* (Mar. 2006), pp. 6 pp. –218.

[52] Kwok, Siew-Hwee, Ee, Yen-Ling, Chew, Guanhan, Zheng, Kanghong, Khoo, Khoongming, and Tan, Chik-How. A comparison of post-processing techniques for biased random number generators. In *Proceedings of the 5th IFIP WG 11.2 International Conference on Information Security Theory and Practice: Security and Privacy of Mobile Devices in Wireless Communication* (Berlin, Heidelberg, 2011), WISTP'11, Springer-Verlag, pp. 175–190.

[53] Kwon, Hyuk-Min, Han, In-Shik, Bok, Jung-Deuk, Park, Sang-Uk, Jung, Yi-Jung, Lee, Ga-Won, Chung, Yi-Sun, Lee, Jung-Hwan, Kang, Chang Yong, Kirsch, P., Jammy, R., and Lee, Hi-Deok. Characterization of random telegraph signal noise of high-performance p-MOSFETs with a high- dielectric/metal gate. *IEEE Electron Device Letters 32*, 5 (May 2011), 686 –688.

[54] Lacharme, Patrick. Post-processing functions for a biased physical random number generator. In *Fast Software Encryption*, Kaisa Nyberg, Ed., no. 5086 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, Jan. 2008, pp. 334–342.

[55] Lak, Z., and Nicolici, N. On using on-chip clock tuning elements to address delay degradation due to circuit aging. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 31*, 12 (2012), 1845–1856.

[56] Liberty, J.S., Barrera, A., Boerstler, D.W., Chadwick, T.B., Cottier, S.R., Hofstee, H.P., Rosser, J.A., and Tsai, M.L. True hardware random number generation implemented in the 32-nm SOI POWER7+ processor. *IBM Journal of Research and Development 57*, 6 (Nov. 2013), 4:1–4:7.

[57] Liebmann, Lars, Pileggi, Larry, Hibbeler, Jason, Rovner, Vyacheslav, Jhaveri, Tejas, and Northrop, Greg. Simplify to survive: prescriptive layouts ensure profitable scaling to 32nm and beyond. vol. 7275, pp. 72750A–72750A–9.

[58] Lorenz, D., Georgakos, G., and Schlichtmann, U. Aging analysis of circuit timing considering NBTI and HCI. In *On-Line Testing Symposium, 2009. IOLTS 2009. 15th IEEE International* (June 2009), pp. 3 –8.

[59] Mack, Chris. *Fundamental Principles of Optical Lithography: The Science of Microfabrication*. John Wiley & Sons, Mar. 2008.

[60] Mahmoodi, H., Mukhopadhyay, S., and Roy, K. Estimation of delay variations due to random-dopant fluctuations in nanoscale CMOS circuits. *IEEE Journal of Solid-State Circuits 40*, 9 (Sept. 2005), 1787–1796.

[61] Marculescu, D., and Talpes, E. Variability and energy awareness: a microarchitecture-level perspective. In *Design Automation Conference, 2005. Proceedings. 42nd* (June 2005), pp. 11–16.

[62] Markettos, A. Theodore, and Moore, Simon W. The frequency injection attack on ring-oscillator-based true random number generators. In *Proceedings of the 11th International Workshop on Cryptographic Hardware and Embedded Systems* (Berlin, Heidelberg, 2009), CHES '09, Springer-Verlag, pp. 317–331.

[63] Marsaglia, George. DIEHARD statistical tests.

[64] Marton, K., Suciu, A., and Ignat, I. Randomness in digital cryptography: A survey. *Rmanian Journal of Iformation Science and Technology 13*, 3 (2010), 291–240.

[65] Massengill, L.W., Alles, M.L., Kerns, S.E., and Jones, K.L. Effects of process parameter distributions and ion strike locations on SEU cross-section data [CMOS SRAMs]. *IEEE Transactions on Nuclear Science 40*, 6 (Dec. 1993), 1804 –1811.

[66] Mathew, S.K., Srinivasan, S., Anders, M.A., Kaul, H., Hsu, S.K., Sheikh, F., Agarwal, A., Satpathy, S., and Krishnamurthy, R.K. 2.4 gbps, 7 mW all-digital PVT-variation tolerant true random number generator for 45 nm CMOS high-performance microprocessors. *IEEE Journal of Solid-State Circuits 47*, 11, 2807–2821.

[67] Maymandi-Nejad, M., and Sachdev, M. A monotonic digitally controlled delay element. *IEEE Journal of Solid-State Circuits 40*, 11 (2005), 2212–2219.

[68] Meli-Segu, Joan, Garcia-Alfaro, Joaquin, and Herrera-Joancomart, Jordi. A practical implementation attack on weak pseudorandom number generator designs for EPC gen2 tags. *Wireless Personal Communications 59*, 1 (July 2011), 27–42.

[69] Meterelliyoz, M., Kulkarni, J. P., and Roy, K. Analysis of SRAM and eDRAM cache memories under spatial temperature variations. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 29*, 1 (Jan. 2010), 2 –13.

[70] Mintarno, E., Skaf, J., Zheng, Rui, Velamala, J.B., Cao, Yu, Boyd, S., Dutton, R.W., and Mitra, S. Self-tuning for maximized lifetime energy-efficiency in the presence of circuit aging. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 30*, 5 (May 2011), 760 –773.

[71] Mizuno, Tomohisa, Okumtura, J., and Toriumi, Akira. Experimental study of threshold voltage fluctuation due to statistical variation of channel dopant number in MOSFET's. *IEEE Transactions on Electron Devices 41*, 11 (Nov. 1994), 2216–2221.

[72] Mojumder, N.N., Mukhopadhyay, S., Kim, Jae-Joon, Chuang, Ching-Te, and Roy, K. Self-repairing SRAM using on-chip detection and compensation. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems 18*, 1 (Jan. 2010), 75 –84.

[73] Monzio Compagnoni, C., Gusmeroli, R., Spinelli, A.S., Lacaita, A.L., Bonanomi, M., and Visconti, A. Statistical model for random telegraph noise in flash memories. *IEEE Transactions on Electron Devices 55*, 1 (Jan. 2008), 388 –395.

[74] Moradi, F., Panagopoulos, G., Karakonstantis, G., Wisland, D., Mahmoodi, H., Madsen, J.K., and Roy, K. Multi-level wordline driver for low power SRAMs in nano-scale CMOS technology. In *2011 IEEE 29th International Conference on Computer Design (ICCD)* (Oct. 2011), pp. 326 –331.

[75] Moradinasab, M., Karbassian, F., and Fathipour, M. A comparison study of the effects of supply voltage and temperature on the stability and performance of CNFET and nanoscale si-MOSFET SRAMs. In *1st Asia Symposium on Quality Electronic Design, 2009. ASQED 2009* (July 2009), pp. 19 –23.

[76] Mueller, J., and Saleh, R. A tunable clock buffer for intra-die PVT compensation in single-edge clock (SEC) distribution networks. In *9th International Symposium on Quality Electronic Design, 2008. ISQED 2008* (2008), pp. 572–577.

[77] Mukhopadhyay, S., Mahmoodi, H., and Roy, K. Statistical design and optimization of SRAM cell for yield enhancement. In *IEEE/ACM International Conference on Computer Aided Design, 2004. ICCAD-2004* (Nov. 2004), pp. 10 – 13.

[78] Nagaraj, K., and Kundu, S. An automatic post silicon clock tuning system for improving system performance based on tester measurements. In *Test Conference, 2008. ITC 2008. IEEE International* (2008), pp. 1–8.

[79] Nyquist, H. Thermal agitation of electric charge in conductors. *Physical Review 32*, 1 (July 1928), 110–113.

[80] of Standards and Technology, National Institute. A statistical test suite for random and pseudorandom number generators for cryptographic applications, Apr. 2010.

[81] Pan, Yan, Kong, Joonho, Ozdemir, S., Memik, G., and Chung, Sung Woo. Selective wordline voltage boosting for caches to manage yield under process variations. In *46th ACM/IEEE Design Automation Conference, 2009. DAC '09* (July 2009), pp. 57 –62.

[82] Pareschi, F., Rovatti, R., and Setti, G. On statistical tests for randomness included in the NIST SP800-22 test suite and based on the binomial distribution. *IEEE Transactions on Information Forensics and Security 7*, 2 (Apr. 2012), 491 –505.

[83] Paul, B.C., Kang, Kunhyuk, Kufluoglu, H., Alam, M.A., and Roy, K. Impact of NBTI on the temporal performance degradation of digital circuits. *IEEE Electron Device Letters 26*, 8 (2005), 560–562.

[84] Pomeranz, I., and Reddy, S.M. On n-detection test sequences for synchronous sequential circuits. In *, 15th IEEE VLSI Test Symposium, 1997* (May 1997), pp. 336 –342.

[85] Raychowdhury, A., Geuskens, B., Kulkarni, J., Tschanz, J., Bowman, K., Karnik, T., Lu, Shih-Lien, De, V., and Khellah, M.M. PVT-and-aging adaptive wordline boosting for 8t SRAM power reduction. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International* (Feb. 2010), pp. 352 –353.

[86] Saneyoshi, E., Nose, K., and Mizuno, M. A precise-tracking NBTI-degradation monitor independent of NBTI recovery effect. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International* (Feb. 2010), pp. 192 –193.

[87] Sanyal, A., Ganeshpure, K., and Kundu, S. Accelerating soft error rate testing through pattern selection. In *On-Line Testing Symposium, 2007. IOLTS 07. 13th IEEE International* (July 2007), pp. 191 –193.

[88] Scholten, A.J., Tiemeijer, L.F., van Langevelde, R., Havens, R.J., Zegers-van Duijnhoven, A.T.A., and Venezia, V.C. Noise modeling for RF CMOS circuit simulation. *IEEE Transactions on Electron Devices 50*, 3 (Mar. 2003), 618 – 632.

[89] Schroder, D.K., and Babcock, J.A. Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing. *Journal of Applied Physics 94*, 1 (2003), 1–18.

[90] Seifert, N., Slankard, P., Kirsch, M., Narasimham, B., Zia, V., Brookreson, C., Vo, A., Mitra, S., Gill, B., and Maiz, J. Radiation-induced soft error rates of advanced CMOS bulk devices. In *Reliability Physics Symposium Proceedings, 2006. 44th Annual., IEEE International* (Mar. 2006), pp. 217 –225.

[91] Sreedhar, A., and Kundu, S. On linewidth-based yield analysis for nanometer lithography. In *Design, Automation Test in Europe Conference Exhibition, 2009. DATE '09.* (Apr. 2009), pp. 381–386.

[92] Sreedhar, A., and Kundu, S. Statistical timing analysis based on simulation of lithographic process. In *IEEE International Conference on Computer Design, 2009. ICCD 2009* (Oct. 2009), pp. 29–34.

[93] Suciu, A., Marton, K., Nagy, I., and Pinca, I. Byte-oriented efficient implementation of the NIST statistical test suite. In *2010 IEEE International Conference on Automation Quality and Testing Robotics (AQTR)* (May 2010), vol. 2, pp. 1 –6.

[94] Suciu, A., Nagy, I., Marton, K., and Pinca, I. Parallel implementation of the NIST statistical test suite. In *2010 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)* (Aug. 2010), pp. 363 –368.

[95] Sunar, B., Martin, W.J., and Stinson, D.R. A provably secure true random number generator with built-in tolerance to active attacks. *IEEE Transactions on Computers 56*, 1 (Jan. 2007), 109 –119.

[96] Suresh, V.B., Antonioli, D., and Burleson, W.P. On-chip lightweight implementation of reduced NIST randomness test suite. In *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)* (June 2013), pp. 93–98.

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

[98] Suresh, V.B., and Burleson, W.P. Robust metastability-based TRNG design in nanometer CMOS with sub-vdd pre-charge and hybrid self-calibration. In *2012 13th International Symposium on Quality Electronic Design (ISQED)* (Mar. 2012), pp. 298 –305.

[99] Suresh, V.B., and Burleson, W.P. Variation aware design of post-silicon tunable clock buffer. In *2014 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)* (July 2014), pp. 1–6.

[100] Suresh, V.B., and Kundu, S. Managing test coverage uncertainty due to thermal noise in nano-CMOS: A case-study on an SRAM array. In *2013 IEEE 31st International Conference on Computer Design (ICCD)* (Oct. 2013), pp. 201–206.

[101] Suresh, V.B., and Kundu, S. On analyzing and mitigating SRAM BER due to random thermal noise. In *2013 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)* (Aug. 2013), pp. 159–164.

[102] Suresh, Vikram B., and Burleson, Wayne P. Fine grained wearout sensing using metastability resolution time. In *Quality Electronic Design (ISQED), 2014 15th International Symposium on* (Mar. 2014), pp. 480–485.

[103] Sylvester, Dennis, Agarwal, Kanak, and Shah, Saumil. Variability in nanometer CMOS: Impact, analysis, and minimization. *Integration, the VLSI Journal 41*, 3 (May 2008), 319–339.

[104] Tadesse, D., Grodstein, J., and Bahar, R. I. AutoRex: An automated post-silicon clock tuning tool. In *Test Conference, 2009. ITC 2009. International* (2009), pp. 1–10.

[105] Takeuchi, K., Nagumo, T., Takeda, K., Asayama, S., Yokogawa, S., Imai, K., and Hayashi, Y. Direct observation of RTN-induced SRAM failure by accelerated testing and its application to product reliability assessment. In *2010 Symposium on VLSI Technology (VLSIT)* (June 2010), pp. 189 –190.

[106] Tam, Simon, Rusu, S., Nagarji Desai, U., Kim, R., Zhang, Ji, and Young, Ian. Clock generation and distribution for the first IA-64 microprocessor. *IEEE Journal of Solid-State Circuits 35*, 11 (2000), 1545–1552.

[107] Tang, X., De, V.K., and Meindl, J.D. Intrinsic MOSFET parameter fluctuations due to random dopant placement. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems 5*, 4 (Dec. 1997), 369–376.

[108] Tanizawa, M., Ohbayashi, S., Okagaki, T., Sonoda, K., Eikyu, K., Hirano, Y., Ishikawa, K., Tsuchiya, O., and Inoue, Y. Application of a statistical compact model for random telegraph noise to scaled-SRAM vmin analysis. In *2010 Symposium on VLSI Technology (VLSIT)* (June 2010), pp. 95 –96.

[109] Tedja, S., Van der Spiegel, J., and Williams, H.H. Analytical and experimental studies of thermal noise in MOSFET's. *IEEE Transactions on Electron Devices 41*, 11 (Nov. 1994), 2069 –2075.

[110] Tega, N., Miki, H., Pagette, F., Frank, D.J., Ray, A., Rooks, M.J., Haensch, W., and Torii, K. Increasing threshold voltage variation due to random telegraph noise in FETs as gate lengths scale to 20 nm. In *2009 Symposium on VLSI Technology* (June 2009), pp. 50 –51.

[111] Tokunaga, C., Blaauw, D., and Mudge, T. True random number generator with a metastability-based quality control. *IEEE Journal of Solid-State Circuits 43*, 1 (Jan. 2008), 78 –85.

[112] Treuer, R., and Agarwal, V.K. Built-in self-diagnosis for repairable embedded RAMs. *IEEE Design Test of Computers 10*, 2 (June 1993), 24 –33.

[113] Tsai, Jeng-Liang, Zhang, Lizheng, and Chen, C.C. Statistical timing analysis driven post-silicon-tunable clock-tree synthesis. In *IEEE/ACM International Conference on Computer-Aided Design, 2005. ICCAD-2005* (2005), pp. 575–581.

[114] Tschanz, J.W., Kao, J.T., Narendra, S.G., Nair, R., Antoniadis, D.A., Chandrakasan, A.P., and De, V. Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage. *IEEE Journal of Solid-State Circuits 37*, 11 (Nov. 2002), 1396–1402.

[115] Udawatta, K., Ehsanian, M., Maidanov, S., and Musunuri, S. Test and validation of a non-deterministic system #x2014; true random number generator. In *High Level Design Validation and Test Workshop, 2008. HLDVT '08. IEEE International* (Nov. 2008), pp. 77 –84.

[116] Unsal, O.S., Tschanz, J.W., Bowman, K., De, V., Vera, X., Gonzalez, A., and Ergin, O. Impact of parameter variations on circuits and microarchitecture. *IEEE Micro 26*, 6 (Nov. 2006), 30–39.

[117] van de Goor, A.J. Using march tests to test SRAMs. *IEEE Design Test of Computers 10*, 1 (Mar. 1993), 8 –14.

[118] van de Goor, A.J., Gaydadjiev, G.N., Mikitjuk, V.G., and Yarmolik, V.N. March LR: a test for realistic linked faults. In *VLSI Test Symposium, 1996., Proceedings of 14th* (May 1996), pp. 272 –280.

[119] von Neumann, John. Various techniques used in connection with random digits. *National Bureau of Standards, Applied Math Series 12*, 1951, 36–38.

[120] Wang, Chih-Wea, Wu, Chi-Feng, Li, Jin-Fu, Wu, Cheng-Wen, Teng, T., Chiu, K., and Lin, Hsiao-Ping. A built-in self-test and self-diagnosis scheme for embedded SRAM. In *Test Symposium, 2000. (ATS 2000). Proceedings of the Ninth Asian* (2000), pp. 45 –50.

[121] Wang, Wenping, Reddy, V., Krishnan, A.T., Vattikonda, R., Krishnan, Srikanth, and Cao, Yu. Compact modeling and simulation of circuit reliability for 65-nm CMOS technology. *IEEE Transactions on Device and Materials Reliability 7*, 4 (2007), 509–517.

[122] Wang, Wenping, Wei, Zile, Yang, Shengqi, and Cao, Yu. An efficient method to identify critical gates under circuit aging. In *IEEE/ACM International Conference on Computer-Aided Design, 2007. ICCAD 2007* (Nov. 2007), pp. 735 –740.

[123] Wann, C., Wong, R., Frank, D.J., Mann, R., Ko, Shang-Bin, Croce, P., Lea, D., Hoyniak, D., Lee, Yoo-Mi, Toomey, J., Weybright, M., and Sudijono, J. SRAM cell design for stability methodology. In *2005 IEEE VLSI-TSA International Symposium on VLSI Technology, 2005. (VLSI-TSA-Tech)* (Apr. 2005), pp. 21 – 22.

[124] Wittmann, R., Puchner, H., Ceric, H., and Selberherr, S. Impact of random bit values on NBTI lifetime of an SRAM cell. In *Physical and Failure Analysis of Integrated Circuits, 2006. 13th International Symposium on the* (July 2006), pp. 41 –44.

[125] Wooters, S.N., Cabe, A.C., Qi, Zhenyu, Wang, Jiajing, Mann, R.W., Calhoun, B.H., Stan, M.R., and Blalock, T.N. Tracking on-chip age using distributed, embedded sensors. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems 20*, 11 (2012), 1974–1985.

[126] Xiong, Jinjun, Zolotov, V., and Visweswariah, C. Efficient modeling of spatial correlations for parameterized statistical static timing analysis. In *IEEE 8th International Conference on ASIC, 2009. ASICON '09* (Oct. 2009), pp. 722–725.

[127] Yamaoka, M., Miki, H., Bansal, A., Wu, S., Frank, D.J., Leobandung, E., and Torii, K. Evaluation methodology for random telegraph noise effects in SRAM arrays. In *Electron Devices Meeting (IEDM), 2011 IEEE International* (Dec. 2011), pp. 32.2.1 –32.2.4.

[128] Zhai, Bo, Hanson, Scott, Blaauw, David, and Sylvester, Dennis. Analysis and mitigation of variability in subthreshold design. In *Proceedings of the 2005 International Symposium on Low Power Electronics and Design* (New York, NY, USA, 2005), ISLPED '05, ACM, pp. 20–25.

[129] Zhuo, Cheng, Blaauw, D., and Sylvester, D. Variation-aware gate sizing and clustering for post-silicon optimized circuits. In *2008 ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED)* (2008), pp. 105–110.