The Pennsylvania State University

The Graduate School

ANALYZING FAILURES IN NANOSCALE DEVICES

A Dissertation in

Electrical Engineering

by

Ramakrishnan Krishnan

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

December 2009

The dissertation of Ramakrishnan Krishnan was reviewed and approved* by the following:

Vijaykrishnan Narayanan
Professor of Computer Science & Engineering
Dissertation Co-Advisor, Co-Chair of Committee

Suman Datta
Associate Professor of Electrical Engineering
Dissertation Co-Advisor, Co-Chair of the Committee

Mary Jane Irwin
Professor of Computer Science & Engineering
Evan Pugh Professor

Kenneth W. Jenkins
Professor of Electrical Engineering
Department Head, Electrical Engineering

Yuan Xie
Associate Professor of Computer Science & Engineering

Ram M. Narayanan
Professor of Electrical Engineering

Kenan Unlu
Professor of Mechanical & Nuclear Engineering

*Signatures are on file in the Graduate School.

Abstract

Aggressive downscaling of transistor sizes for increased performance and lower costs have pushed the devices to their physical limits. Design goals are governed by several factors other than power, performance and area such as variations and reliability margins. The reliability of a system is affected from it's birth to death by various phenomena such as process variations, soft and hard faults and aging mechanisms. Reliability effects have also become a major bottleneck due to different physical phenomena and introduction of newer materials. Process variation poses a huge threat to the reliability of the system during the initial days of chip operation. Various transient mechanisms such as radiation induced soft errors and cross talk alter system operation during the useful lifetime of a chip. In the near end of it's lifetime, several aging phenomena such as Negative Bias Temperature Instability (NBTI) and Hot Carrier Injection (HCI) limit the frequency of operation. Soft errors and NBTI are two major reliability threats that require detailed analysis and methods to ensure high reliability limits. Accelerated testing mechanisms, fast and accurate CAD tools to estimate the impact of failures and good design techniques are required for designers to promise high margins of reliability. These necessities along with the interaction of the reliability phenomena with concerns such as variations have been addressed comprehensively. Tool for faster and accurate calculation of Soft Error Rate (SER) of hierarchical structures called Hierarchical Soft Error Estimation Tool (HSEET) have been developed. A framework called New-Age for estimation of degradation caused due to NBTI has also been developed for complete system analysis. Analysis of the interplay of variations with the SERs have been performed. FPGAs which have become a highly popular architectures due to its reconfigurability are also susceptible to these reliability dangers. We also show the impact of NBTI on FPGAs and proposed solutions to increase its lifetime. The impact of NBTI on sequential circuits have

been performed.

Table of Contents

List of Figures

List of Tables

Abbreviations

CMOS Complementary Metal-Oxide-Semiconductor

FPGA Field Programmable Gate Arrays

HCI Hot Carrier Injection

HSEET Hierarchical Soft Error Estimation Tool

HSPICE H-Simulation Program With Integrated Circuit Emphasis

MOSFET Metal-Oxide-Semiconductor Field Effect Transistor

NBTI Negative Bias Temperature Instability

SEAT-LA Soft Error Estimation Tool - Logic Analyzer

SER Soft Error Rate

TDDB Time Dependent Di-electric Breakdown

Acknowledgments

Dedication

Chapter 1

Introduction

For the past 50 years, the semiconductor industry has continuously witnessed a relentless progress in reduction of transistor feature sizes and increase in number of transistors in a chip. Chip designers concentrated solely for optimization on the golden triptych of ***Power,Performance and Area***. However due to aggressive scaling of technology, reliability issues which took the back seat for the past few decades is prominently looming as a threat for present and future technology nodes [1].

1.1 Overview of Circuit Reliability

In the relentless pursuit of satisfying Moore's law and reducing costs, the world of digital design has shrunk the sizes of devices to nanometer scale. This reduction in transistor sizes along with V_{DD} scaling and higher clock frequencies have started having a negative impact on the reliability of circuits. Existing CMOS materials have been pushed to their physical limits with gate oxides lingering at few atomic layers. The failure rate of a product or a circuit can be described using the bathtub curve as shown in figure 1.1. The bathtub curve shows the failure rate of a product over it's entire lifetime. The initial period called the '*infant mortality period*' exhibits a high and decreasing failure rate. This period is due to material defects, design blunders, errors in process-assembly and variability and are enhanced by processes such as burn-in during testing. The second period which is called 'constant failure' region is the period during which the failure rate remains almost constant and is due to various mechanisms such as radiation induced transient errors and cross-talk. The failure rates start to go up after the constant failure period due to the wear out of the materials and the degradation failures which is called the '*wearout failure region*'.

1.2 Process Variations

Process variations are a major factor in the cause of early failures in the bathtub curve. Semiconductor manufacturing variations occur when process parameters deviate from their ideal values used during design time. Process variations have always been a key concern for manufacturability and circuit design. Technology scaling has exacerbated the importance of understanding and controlling variations. System failure rates have also increased tremendously due to methods such

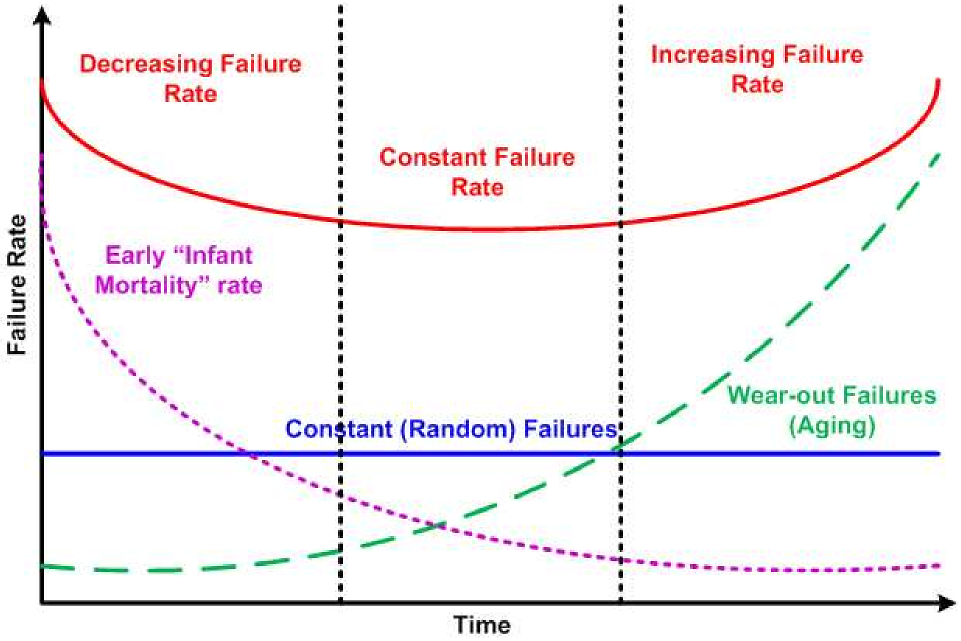


Figure 1.1. The Bathtub Curve

as burn-in tests in the presence of process variations.

Process variations have become unavoidable in the nanoscale regime and affect the physical parameters such as gate length, inter layer dielectric thickness, gate oxide thickness etc. The changes in these physical parameters affect the electrical characteristics of the Metal-Oxide-Semiconductor Field Effect Transistor (MOSFET) transistor such as on-current (I_{ON}), threshold voltage (V_{TH}). These electrical parameters in turn affect the delay and power consumption of the circuits. These variations also have a huge impact on the reliability of the system and are one of the main causes for the failure of digital circuits.

1.3 Transient Failures

Transient failures are soft failures that do not cause a hard fault in the system and are transient in nature. The two most important causes of transient failures in digital systems are radiation induced soft errors and cross talk between wires.

Radiation induced soft errors if left untreated are prone to be the single most dominant cause for transient errors than all the other error mechanisms combined. So, we have mainly concentrated on radiation induced soft errors as the source of transient errors for our study. Radiation induced soft errors arise due to the interaction of radio-active particles with silicon substrate producing a cloud of electron-hole pairs on the substrate. The electron-hole pairs produce a spike of current in the circuit causing a transient fault in the system. These transient faults are non-permanent in nature and do not cause and hence the name 'soft errors'. The faults are intermittent in nature and are main causes of failure in semiconductor memories and chips.

1.4 Aging Mechanisms

As the chip wears out due to continuous operation, the chips become faulty over time due to unacceptable timing or functionality. As technology scales, physical limits of devices have been reached due to continuous shrinking of physical limits of materials such as gate oxides. Aggressive reduction in gate oxide thickness upto 1nm has drastically increased the vertical electric field and are main cause of failures due to aging. Time Dependent Di-electric Breakdown (TDDB), NBTI and HCI are the three most dominant failure mechanisms due to aging. Figure 1.2 [2] shows the transition of aging mechanisms as technology scales as a function of thickness of gate oxide and supply voltage [3]. The arrival of the high-k dielectrics from 45nm technology also exacerbates the problem of NBTI degradation [4]. The high-k gate stack consists of two layers namely a high-k gate stack layer and an interfacial SiO_2 layer [5], the overlaying high-k film induces more defects into the SiO_2 interfacial layer which also increases NBTI degradation.

NBTI is predominant in PMOS transistors while HCI is predominant in NMOS transistors. NBTI occurs due to generation of interface traps in $Si-SiO_2$ interface when negative bias is applied to the gate. HCI happens when there is a transition in the gate voltage and generation of interface traps at $Si - SiO_2$ in the drain end of the device. These two mechanisms cause a change in the threshold voltage V_{TH} of the device thereby altering the performance and power consumption of the device. Huge changes in the V_{TH} of the device increases the delay of the circuit thereby causing a timing failure. NBTI and HCI are reported as the two most dominant failure mechanisms by ITRS [45].

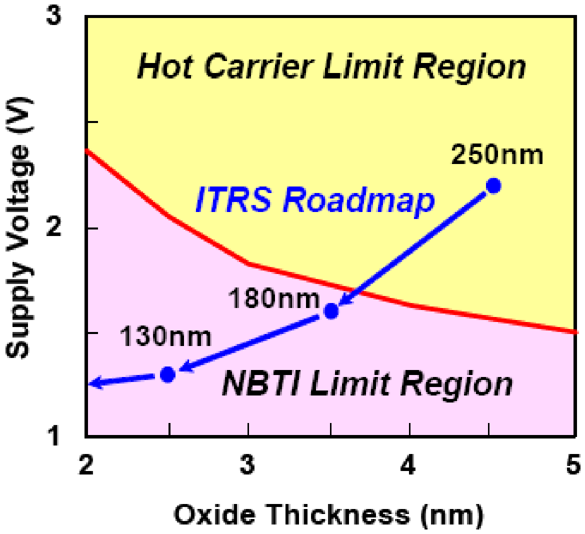


Figure 1.2. Transition of aging mechanisms as function of gate oxide thickness

1.5 Organization of the dissertation

In this dissertation, all the above mechanisms that cause failures in semiconductor memories and circuits have been addressed. The remainder of this dissertation is organized as follows. In Chapter 2, CAD tools were developed for estimation of error rates due to radiation induced soft errors and NBTI namely HSEET and New-Age respectively. Chapter 3 explains the impact of various aging mechanisms on sequential circuits and Field Programmable Gate Arrays (FPGA)s. Chapter 4 explains the interplay between various failure mechanisms. Finally, chapter 5 concludes the dissertation summarizing the results and contributions.

Chapter 2

CAD Tools for Reliability

2.1 Radiation Induced Soft Errors

Soft errors, also called transient faults or single-event upsets (SEUs) are non-permanent in nature and are caused due to electrical noise or external radiation. Specifically, a radiation induced soft error is caused due to the transient current pulse that is generated at the p-n junction. The radiation particle which passes through a strong electric field generates a large number of electron-hole pairs. The electron-hole pairs are collected efficiently in the reverse biased p-n junction region creating a short current pulse flowing through the device. Specifically, sub-90 nm circuits will face a huge problem of radiation induced soft errors[6, 7, 8].

Reduced feature sizes, high logic densities, lower operating voltage and shrinking nodal capacitances have added to the increase in the sensitivity of circuits to transient errors and have been a concern for the past three decades [9]. These errors cause bit flips in memories and latch incorrect data onto the registers in combinational circuits. However, the SER of SRAMs have been constant over technology generations and the usage of error correcting codes (ECC) has greatly reduced their SER. The impact of soft errors on combinational circuits have gained huge attention in recent times. It has been predicted that at 45 nm technology node, the majority of the transient failures will be due to soft errors that occur in combinational blocks [10]. Present techniques calculate the SER of combinational circuits quickly for smaller circuits, but take longer times for large circuits such as adders and multipliers. Therefore, it is necessary to devise methods to quantify the impact of SERs of combinational circuits accurately and quickly.

In the recent past, there have been various approaches proposed to calculate the SER in combinational circuits accurately and efficiently [11]-[12]. All these mechanisms combine different kinds of circuit analysis, symbolic and statistical methods to compute the error rates. Path based circuit analysis methods on large flattened circuits are more time consuming, but have lesser error percentage than symbolic methods which are much faster. Hence, it is imperative to have fast circuit analysis methods without sacrificing accuracy.

In this paper, we have presented such a novel analysis methodology and an accurate tool known as **Hierarchical Soft Error Estimation Tool (HSEET))** for the calculation of SER making use of hierarchical circuits using block level characterizations as opposed to gate level characterizations done in [13]-[14]. We define **hierarchical circuits** as circuits which employ standard blocks interconnected to form the required logic. Block level characterization aids in improving the run-time of the designs since the gate level designs consume huge run-time due to the long paths through the gates and calculation along the path. This method is extremely effective when applied to designs such as adders and multipliers since these designs employ hierarchical blocks in them. This approach helps us in faster computation than path based analysis due to larger coverage of cells while maintaining similar accuracy. To characterize the blocks, we use highly accurate Soft Error Estimation Tool - Logic Analyzer (SEAT-LA) [14]. SEAT-LA is an accurate tool to estimate the SER of combinational circuits. It calculates the SER of circuits using cell libraries characterized for soft error analysis and analytical equations to model the propagation of the voltage or current pulse to the input of a state element. However, SEAT-LA and other methods are slow for larger circuits such as adders and multipliers since it does not take advantage of its hierarchical blocks. Using SEAT-LA for our block characterizations, we propose a fast novel methodology using circuit methods and probability theory for computing the SER of logic circuits employing hierarchical architectures in this paper. This is the first work that exploits the hierarchical nature in circuits and block based approach to compute the SERs. We have demonstrated our technique on various hierarchical structures. The main contributions of this tool are (i) a novel technique to estimate the SER of hierarchical circuits and (ii) a tool to calculate the SER of architectures using hierarchical structures. The remainder of the chapter is organized as follows. Section 2.2 explains the background and related work in this field. Section 2.3 talks about the characterization of basic blocks and the proposed methodology for the tool. Section 2.4 explains the design and tool-flow. The experimental SER results of various adders and multipliers have been described in section 2.5.

2.2 Background and Related Work

Radiation induced soft errors are transient errors generated due to the bombardment of high energy cosmic particles on the silicon substrate. These result in generation of electron-hole pairs as they pass through the p-n junction. These pairs generate a short transient current pulse which cause soft errors in the form of bit-flips in memories and state elements.They also cause transient glitches in combinational logic circuits which get latched on to the registers and latches causing errors. The three inherent masking mechanisms in logic circuits that mitigate the propagation of the glitches are **logical masking, electrical masking** and **latch window or timing window masking** mechanisms. Logical masking occurs when the error is suppressed from appearing at the output of a gate whose result is entirely decided by the other inputs. Electrical masking occurs when the output is not affected by the pulse due to its attenuation during the propagation through the gates which is dependent on the electrical properties of the gates. Latch window masking or Timing window masking eliminates the error when the propagated pulse cannot get latched onto the register due to its arrival at the wrong time of the clock transition. Thus, the voltage glitch caused due to the current pulse has to satisfy the setup and the hold time of the flip flop to get latched on to the output. These three masking mechanisms suppress the overall likelihood of a wrong pulse appearing at the output and getting latched on by a register.

Many of the previous works have modeled the masking effects in a detailed manner. These works have concentrated on calculating the error rates using different techniques such as circuit analysis methods and symbolic methods. Massengill in [11] computed the SER by path analysis, propagation probabilities in terms of probability matrices. ASSA by Zhao [13] obtains the softness distribution which represents the noise tolerance capability of a given circuit. SERA [15] uses probability theory, circuit simulation methods, graph theory and fault simulation methods for analysis. ASERTA by Dhillon in [16] use mathematical equations to model electrical masking without considering voltage pulse amplitudes. FASER [17] makes use of binary decision diagrams and pre-characterization methods to compute the error rates.This method is fast while the error rate is high due to single ramp approximation electrical masking models. Rao in [18] uses waveform models and predictors to compute the SER. It is extremely fast while the error percentage is high compared to [14]. SEAT-LA uses pre-characterized cell libraries and analytical equations for SER estimation. SEAT-LA [14] describes how it takes the all the masking effects into consideration while estimating the error rates. Krishnaswamy in [19] uses computational framework based on probabilistic transfer matrices (PTM) and algebraic decision diagrams (ADD) to study the effects of soft errors on circuits. However, the PTMs represent only the functionality of the gates representing logical masking while the electrical masking and timing window masking are not taken into account by them. MARS-C by Zivanov in [12] also uses BDDs and ADDs to study the circuit reliability factors.

In this work, we use pre-characterized blocks for pulse generation and propagation and logic level simulation methods in HSEET as explained in section **??** to compute the SER efficiently. We account for all masking effects and present an estimation method.

2.3 Characterization & Methodology

2.3.1 Characterization of Basic Blocks

According to the proposed method, the gate level definition of the circuit architecture is first defined in terms of basic building blocks. For example, in the case of a multiplier, a 3:2 counter or 4:2 counter is considered as the basic building block and for prefix adders: the propagate block, generate block and the carry cells form the basic building blocks. These building blocks are first characterized using a logic level tool, SEAT-LA [14]. SEAT-LA is shown to be accurate when compared with H-Simulation Program With Integrated Circuit Emphasis (HSPICE) and is publicly available. It provides better speed and can be used for faster characterization. The gate level net-list for each of the building blocks are synthesized using Synopsys Design Compiler. The corresponding nodal capacitances are determined for 65 nm PTM technology [20] using Micromagic, a layout tool. The states of the internal nodes are obtained using modelsim and fed along with the paths and capacitances to SEAT-LA to perform the characterization. We assume a double exponential current pulse injection model as described in [21] and given by equation 2.1 to model a soft error in circuits.

$$I_{current\_pulse}(t) = I_{peak}.(e^{-t/\tau_a} - e^{-t/\tau_b}) \tag{2.1}$$

where $I_{peak} = Q/(\tau_a - \tau_b)$ in which Q is the collected charge as the result of the particle strike. τ_a and τ_b are the collection time constant and ion-track establishment time constants respectively. These are completely technology dependent process parameters. We perform two kinds of characterizations for each building block.

2.3.1.1 Generation

The first type of characterization done on each of the block is to determine the range of voltage pulses that appear at the output for a particle strike occurring at that block. Thus, this table contains a set of current spikes that are created due to a particle strike and the corresponding voltage pulses at the outputs. This characterization is done for a particle strike at each susceptible node in the basic block and for all possible input states. This table also contains the probability of the particle strike $P(Q_c)$ striking a node and hence the particular current pulse generated due to the charge deposited. This is computed from the equation presented in equation 2.2.

To calculate $P(Q_c)$, we use existing data and methods. We obtain the data points for neutron energy and the corresponding differential flux at sea level from the JEDEC Solid State Technology Association Standard JESD89 [?]. From the obtained points, we construct an exponential curve for the charge probability given by equation 2.2.

$$f(c) = A.exp(-\lambda * c) + f \tag{2.2}$$

where A, λ and f are constants. We consider neutron energies only between 0.5 MeV to 4 MeV and approximately,1 MeV deposits 20 fC of charge on silicon (so, 10-80 fC). The upper-bound is chosen to be 4 MeV since the charge deposition in 65 nm technology is limited by the charge collection depth which in turn is dependent on the doping which is almost doubled from 130 nm [22]. In other words, we assume that a strike of <10 fC does not produce any transient current pulse while the probability of a strike producing >80 fC is very minimal. The curve is then integrated to obtain the total area or the probability $P(Q_c)$ which is then normalized and is shown in the figure 2.1. Similar methods have been widely used for accounting the rate of strikes in [18, 23]. Thus, we have a set of known voltage pulses at each output for all possible strikes at each susceptible node.

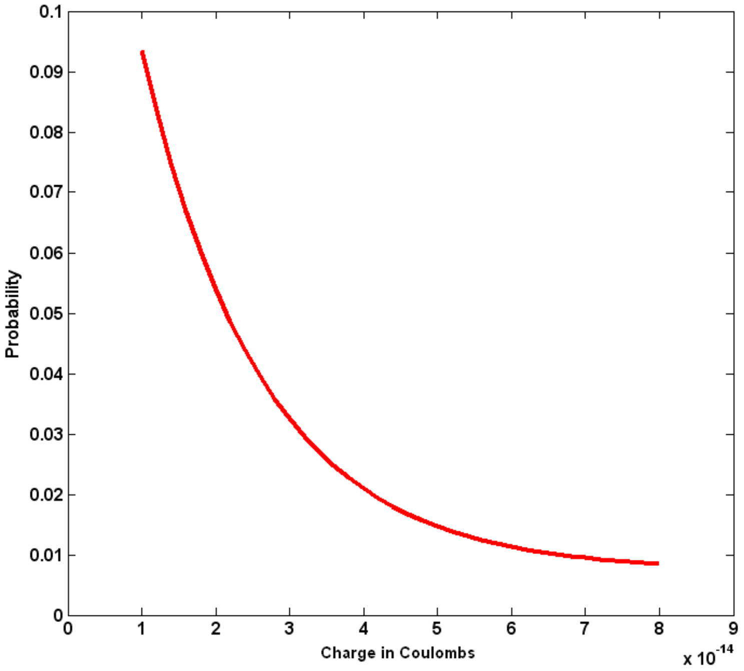


Figure 2.1. Probability of Charge Deposition (Current Pulse Generation)

2.3.1.2 Propagation

Secondly, we characterize each of the blocks for a range of voltage pulses that appear at the outputs of the block for different voltage pulse appearing at the input to the block. This table contains the voltage pulse at each output corresponding to an input voltage pulse of the block. Input pulses of varying amplitudes and pulse-widths are injected on to the inputs and their results at the output blocks are noted. This table provides us the pulse propagation characteristics of the block from input to the output. This effectively captures the electrical masking properties, which are well explained in [14]. Such propagation tables are obtained for all possible input vector states. These propagation tables also inherently contain the logical masking phenomenon due to the states of the internal nodes and do not produce an error at the output.

2.3.2 Flip-Flop Characterization

Apart from the pulse generation and propagation properties of the blocks, the flip-flop characteristics also play an important role in determining the soft error rate of logic circuits. Flip-flops provides the natural boundary for the termination of pulse propagation and the soft error rates differ across flip-flop boundaries. So, it is imperative to study the characteristics of the flip-flops based on the voltage pulse at it's input. To take this into account, the flip-flops (FF) connected to the output of the logic circuits are characterized for the probability of a voltage pulse to cause an error at the output. This is determined in terms of timing window (tw). This timing window is calculated by dividing the time for which a given voltage pulse can cause an error at the output by the clock period in which the flip-flop (and hence the logic circuit) is operating. Once the tw is determined for a range of pulse sizes, the soft error rate (SER) at a given output O with a circuit of N nodes can be calculated by the following equation 2.3 explained later.

We perform timing window characterizations for two different flip-flops namely Transmission Gate Flip-Flop (TGFF) and Hybrid Latch Flip-Flop (HLFF) using HSPICE. The timing window characteristics of the flip-flop greatly determine the error rates since large timing windows increase the SER of logic circuits tremendously. The timing windows for varying pulse-widths and a fixed voltage amplitude are shown in the figure 2.2 for both TG and HL based flip-flops. We observe that the timing window of HLFF is more than that of TGFF for the same pulse width due to HLFF being pulse triggered in nature. Particularly HLFF also has timing windows for lower pulse-widths for which TGFF does not latch on the error for. The effect of these FF timing windows on the error rates are explained in section 2.5.

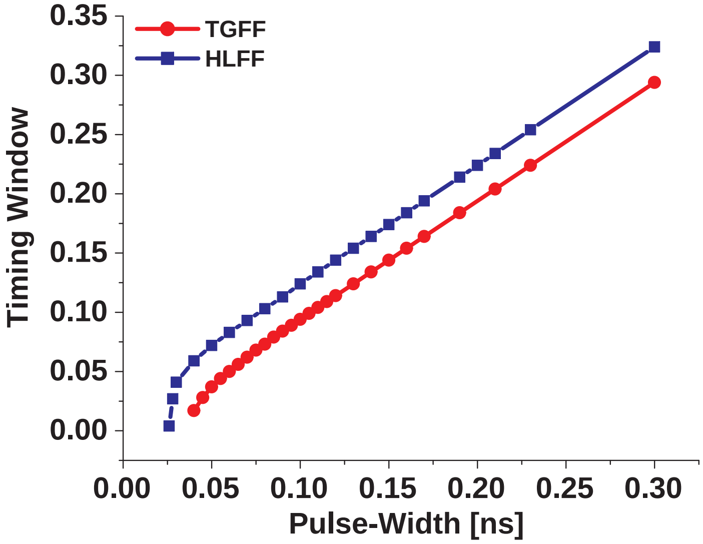


Figure 2.2. Timing Window of Flip-Flops

Note that all these characterizations need to be performed only once for each technology node. Methods that involve gate resizing can be easily included by characterizing them as new blocks.

2.3.3 Methodology

The new methodology proposed uses equations 2.3-2.7 for computing the SER of a logic circuit. The Soft Error Rates (SER) here is a statistical quantity computed as summation of probabilities of a voltage pulse appearing at the output multiplied by the corresponding timing windows of the register at the output. Equations 2.3-2.7 summarize the methodology in a mathematical form.

$$SER_{circuit} = \sum_{i=1}^{i=O} P(V_{out})_i * tw \tag{2.3}$$

$$P(V_{out})_i = P(V_{out})^{gen} + P(V_{out})^{prop} \tag{2.4}$$

$$P(V_{out})^{gen} = \sum_{j=1}^{j=N} P(V_{out})^{gen}_{o=0/1} \tag{2.5}$$

$$P(V_{out})^{gen}_{o=0/1} = P(V_{out})^{gen}_{n=0} + P(V_{out})^{gen}_{n=1} \tag{2.6}$$

$$P(V_{out})^{gen}_{n=0/1} = P(Q_c) * \frac{A_{node}}{A_{total}} * P(Node_n) \tag{2.7}$$

where,

O = Total number of Outputs

tw = Timing window of the flip-flop for a particular pulse-width

N = Total number of internal susceptible nodes in the block

A_{node} = Active area of the suceptible node

A_{total} = Total Area of the circuit

$P(Node_n)$ = Static state probability of the node n

The overall SER of a circuit is calculated as product of sum of probabilities of voltage pulses appearing at an output and their corresponding timing windows of the flip-flop. For each output in the circuit, we obtain the probability of a particular voltage pulse in the timing window table that could reach the output of the block as the input to a state element. The corresponding current pulses that generate the error at the output is computed and the probability of charge required for it calculated from equation 2.2.

$P(V_{out})^{prop}$ in equation 2.4 does not imply a probability of propagation. It indicates that the error could be due to a pulse propagated from the previous block if it is not generated in that block denoted by $P(V_{out})^{gen}$. $P(V_{out})^{gen}$ is computed as a summation as shown in equation 2.6 where a strike at any internal node (out of N) could result in a glitch at the output. It is calculated such that the generation of the current pulse results in an error at the output. Equation 2.6 denotes that the analysis is done for both possible states of the internal node. Finally, the probability of generation at a particular node is calculated as the product of the probability of that particular charge generation $P(Q_c)$, the area factor and static state probability of the node $P(Node_n)$ using equation 2.7. A_{node} is the total drain area susceptible in a node for a single event upset to occur is calculated from the size of the transistors and A_{total} is the total area of the circuit. These values are obtained from the synthesis tool Synopsys Design Compiler or a layout editor. The internal static state probabilities of the all the nodes can be calculated using the simple equations and methods (omitted here due to space constraints) described in [**?**] and the SER is calculated using the equations 2.3-2.7.

In this methodology, we do not consider the effect of re-convergent paths since the effect of re-convergence do not change the nature of the waveforms greatly [15] and moreover, these arithmetic circuits considered do not exhibit re-convergence (neglected in muxes and decoders). In HSEET, we proceed backwards by using the timing window unlike SEAT-LA to calculate the current pulses that produce them. By this, we eliminate the timing windows that do not produce an error at the output. Current pulses that are too large ($>$ 80fC) are also eliminated due to rare occurrence by this method. The process is carried on backwards from each output to all the inputs for each possible timing window until the pulse is completely electrically masked or logically masked. We performed the analysis for 20 timing windows ranging from the minimum pulse-width and constant amplitude(for sake of simplicity) to get latched to 0.2ns. The obtained current pulses are multiplied by the static probabilities which are obtained using a modified version of ACE [24].

2.3.4 Masking Mechanisms

By characterizing the basic blocks, the masking effects are taken into consideration in a detailed way. The generation and propagation tables do not contain voltage pulse values if the other inputs to the gate or the block determine the output of the gate. This captures the logical masking effect and is inherently embedded in SEAT-LA. Electrical masking effects of the internal nodes are captured in the characterization process. The effect of electrical masking on the output nodes are captured as voltage pulse amplitudes and pulse-widths. This is significant since pulses which are not greater than $0.5V_{dd}$ for a positive pulse and has a pulsewidth lesser than the setup and hold-time do not get latched by the flip-flop. Thus, the set of entire voltage amplitudes and pulse-widths obtainable are present in the table. Timing window masking is accounted for by calculating the timing windows for each pulse-width and amplitude as explained in section 2.3.2.

2.3.5 Illustration & Verification

In this section, we provide examples for the methodology proposed above. The first verified circuit here is a ISCAS benchmark C17 circuit. The schematic of C17 is shown in figure 2.3. The extracted gate level net-list along with the capacitances was given as an input to both SEAT-LA and HSEET and the comparable outputs of both the methods for verification are presented here. In this case, we characterize a 2-input NAND gate as a block (we use a simple block for illustration purposes). The NAND block is characterized for both generation and propagation tables along with the characterization of the flip-flops. Here, we use NAND gates as blocks for characterization just to show that blocks can be of variable size and the characterization tables are a superset of tables in SEAT-LA. It should be noted that if there are other gates present in the circuit, this methodology becomes similar to SEAT-LA with no room for improvement.

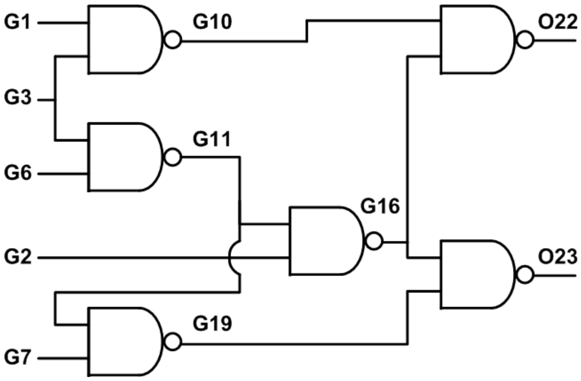


Figure 2.3. Schematic of C17

The proposed methodology is applied to the circuit and the results are checked with pre-verified SEAT-LA. Since SEAT-LA computes SER for fixed input vector states, we compare the observed timing windows for a given current pulse injected at all the internal nodes in SEAT-LA and HSEET. Comparing the timing windows verify the exactness of the pulse generation and propagation properties of the circuits. The timing windows (tw) are also verified using HSPICE with the design laid out using Micromagic Max. It is performed by moving a voltage pulse of fixed pulsewidth over an entire clock cycle and observing the window of latching. The results are shown in the figure 2.4. Since our tool uses SEAT-LA for characterization, the difference in the timing windows between the two is almost zero except incase of G11 to O22 and O23 where it is due to interpolation of timing windows from the tw tables. The observed average error rate in timing windows between HSPICE and HSEET is 7.65%. The difference in the timing windows between HSPICE and our tool is due to the approximation of the capacitances during the characterization and interpolation of timing window values.

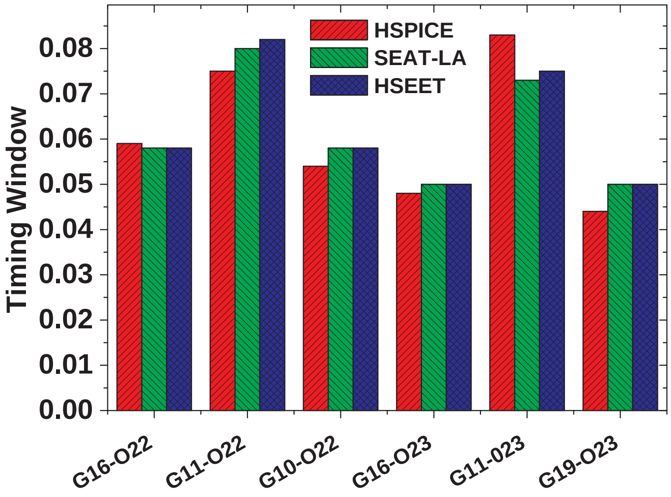


Figure 2.4. Timing Windows of C17

Secondly, we present the results of the few lower order bits of the output of a 4-bit array multiplier. The typical structure of an array multiplier is as shown in the figure 2.5. Unlike the first example, we employ bigger structures for characterization to speed up the process of calculation. As the number of inputs N

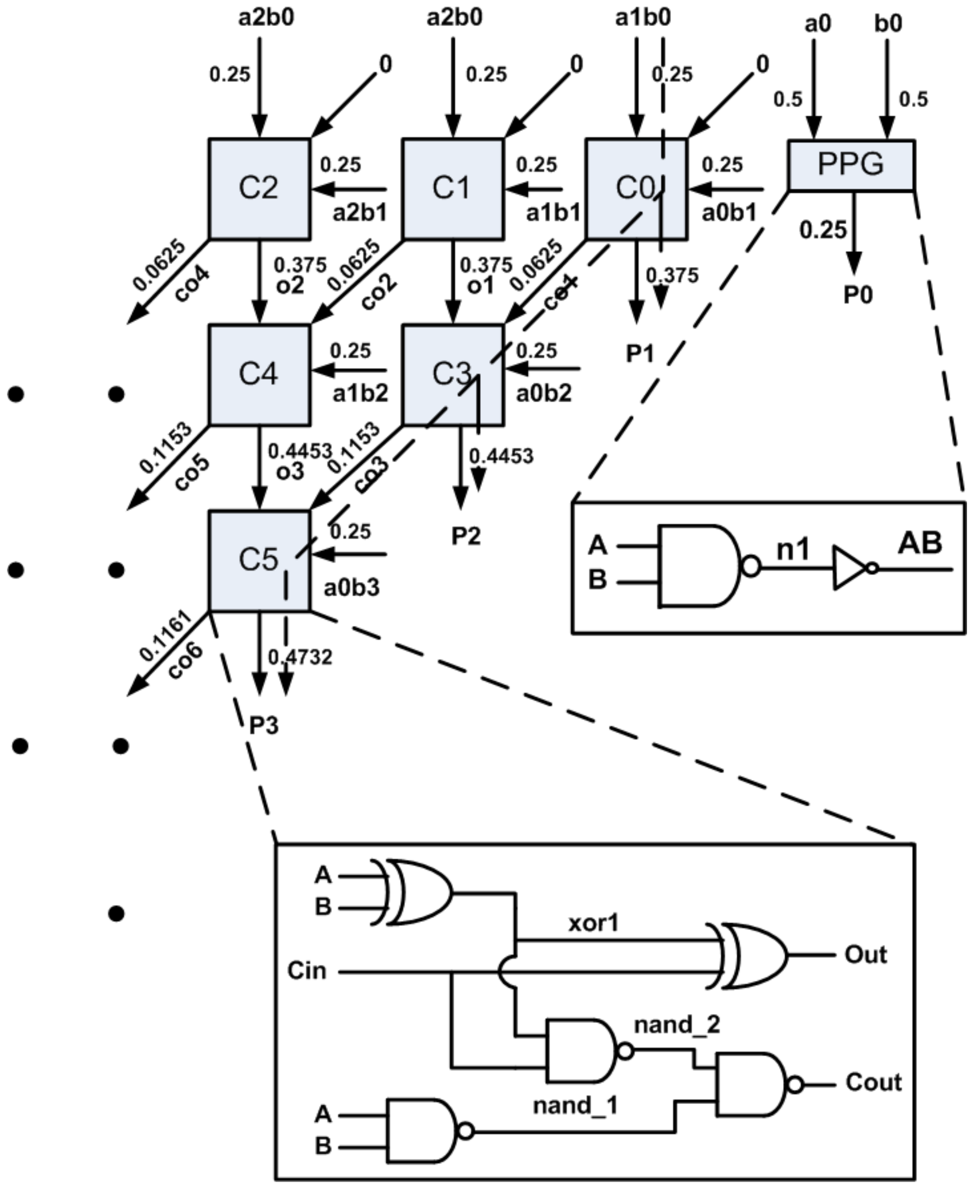


Figure 2.5. 4-Bit Array Multiplier

to the block increase ,the characterization tables become large in number ((2^N) tables of generation and propagation each for N inputs). So, it is necessary to break up the components into smaller components for easier characterization and faster computations. The components characterized for this design are 3:2 coun-

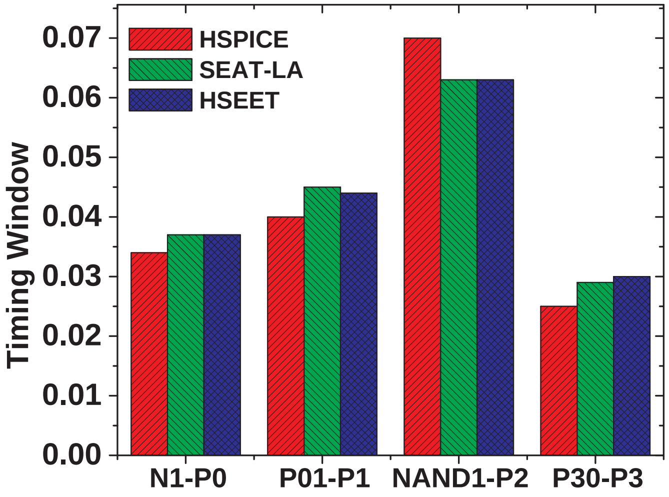


Figure 2.6. Timing windows for lower order bits of 4-Bit Array Multiplier

ters and partial product generators. Figure 2.5 shows the construction of a 4-bit array multiplier using 3:2 counter and partial product generators. For the purpose of illustration, we have shown the lower order bits $(P_0 - P_3)$ of an array multiplier in figure 2.5. The array multiplier is constructed using 3:2 counters and partial product generators. Array multiplier consists of arrays of identical cells that generate the partial products and accumulate them at the same time. It has a reduced execution time but increased hardware complexity. The blocks C in the figure 2.5 represent a 3:2 counter and PPG represents a partial product generator. Partial product generators are simple AND gates which are synthesized using a NAND gate and an inverter since the mapping in SEAT-LA is always performed using 4 cells which are an inverter,NAND, NOR and EXOR. For verification purposes, the timing windows computed by HSPICE, SEAT-LA and HSEET are shown in the figure 2.6. In figure 2.5 , the dotted line shows the path of error propagation in the input block C0 to output bits P1, P2 and P3. Nodes N1, P01, NAND1 and P30 are few of the internal nodes in the array multiplier that when struck by a particle produces an error at the output. The average difference in the timing window between HSPICE and HSEET is 12.21%.

2.3.6 Speedup

In this section, we describe the advantage of using block based approach to speed up computation. Consider the circuit in the figure 2.7 which consists of two 3:2 counters interconnected with 10 gates in them. Partitioning them into 2 blocks gives us a much faster computational capabilities. For example, figure 2.7 shows us the error strike and propagation path through 4 transistors. On a flattened circuit layout, SEAT-LA operates on calculating the current to voltage transfer characteristics in the first gate followed by propagation through the next 3 gates while HSEET computes the exact output characteristics with one single generation and propagation look-up table. Let D be the time taken by SEAT-LA and HSEET to perform a single table look-up operation. Then the computational time required for SEAT-LA is 4D and 5D for Out and Cout respectively (number of gates on the path from the point of strike to the output observed) while HSEET only takes 2D for both the cases. This is based on the number of gates for which the transfer functions need to be indexed incase of SEAT-LA. In contrast, HSEET only requires look-up at block level reducing the run-time hugely. As the number of blocks increase (the length of the path), the time taken for SEAT-LA to compute the vulnerability on a particular output due to an error at an internal node can be given by CN-1 where N is the number of blocks and C is the number of gates in the longest path inside the block (if partitioning into blocks) whereas HSEET only takes N units. Thus, a much faster computational tool for calculating SERs has been developed.

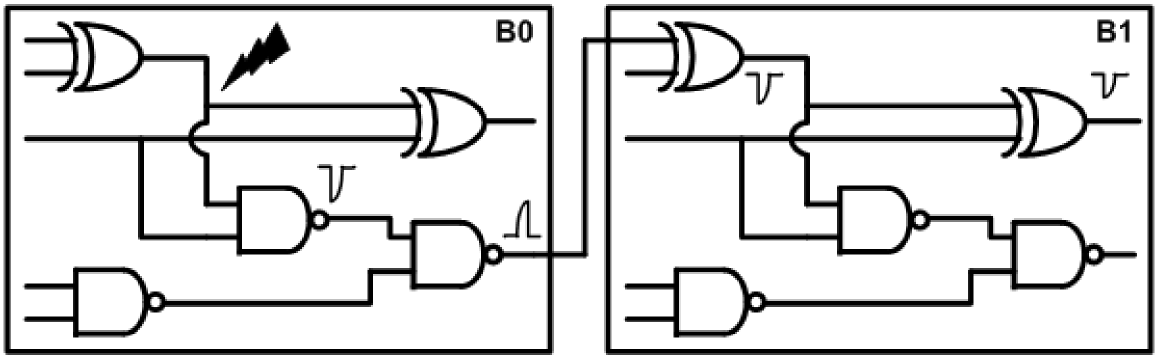


Figure 2.7. Example for Speedup

2.4 Design & Tool-flow

The tool flow of HSEET is shown in the figure 4.3 and 2.9. The back annotated gate level net-list only using pre-characterized blocks is obtained initially and the paths from all the inputs to the outputs are extracted using Synopsys Design Compiler. The static state probability of all the gate inputs are obtained using the EDIF net-list generated using Design Compiler which is fed as an input to the modified probabilistic ACtivity Estimation tool (ACE) [24] to compute them. The static state probabilities of the internal nodes in the blocks are computed dynamically using an in-house tool written in Perl. The pulses (both negative and positive) for (pre-computed) timing windows table are taken and the corresponding required voltage pulses are computed using the generation and the propagation tables as indicated in the figure 2.9. The probability of a particular current pulse occurring at the node is calculated as shown in the figure 2.1. The area parameters are obtained from the layout editor and Design Compiler. The total SER of the design is calculated as explained in section 2.3.3. The complete tool has been written using Perl and Tcl and there is still a huge room for improvement from its current status to increase the speedup of our tool.

2.5 Experimental Results

In this section, we present the SER results obtained for different hierarchical circuits such as multiplexers, decoders, adders and multipliers. All the simulations were performed on a Sun Blade 1000s running Solaris 9 operating system with 1GB RAM. The circuits were designed using Verilog and synthesized using Design Compiler. The blocks characterized for analysis are 2:1 muxes, 1:2 decoders, 3:2 counters and partial product generators. The SER of various designs in table 2.2 and 3.6 were obtained with a TGFF as the register at the output. Run-times for

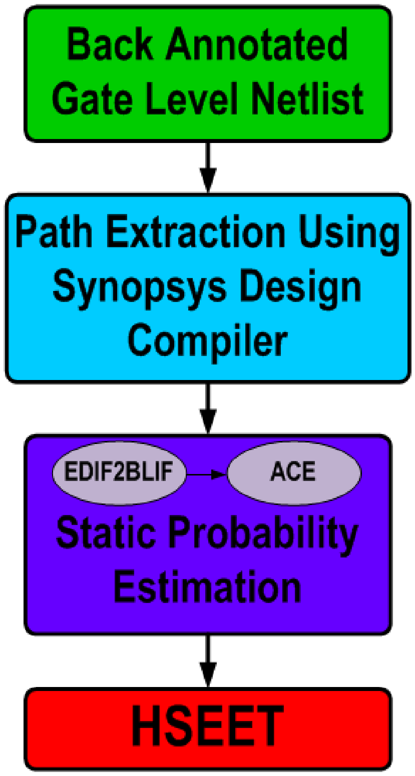

Back Annotated Gate Level Netlist
Path Extraction Using Synopsys Design Compiler
EDIF2BLIF
ACE
Static Probability Estimation
HSEET

Figure 2.8. SER Estimation Tool-flow

HSPICE were adapted from [14] for static input vectors. Simulations were also performed in SEAT-LA for fixed input vectors and their run-times were noted. As HSEET does not operate on fixed input states, the static probabilties were modified to mimic SEAT-LA and the corresponding run-times are shown in the table 2.2.

Table 2.2 shows the comparison of speedup between HSPICE, SEAT-LA and HSEET. HSEET has an average speedup of 14084X times over HSPICE and 12.25X times over its contemporary tool SEAT-LA. HSEET also has minimum speedup of 10.1X over SEAT-LA for the designs considered above and hence shows the impact of block

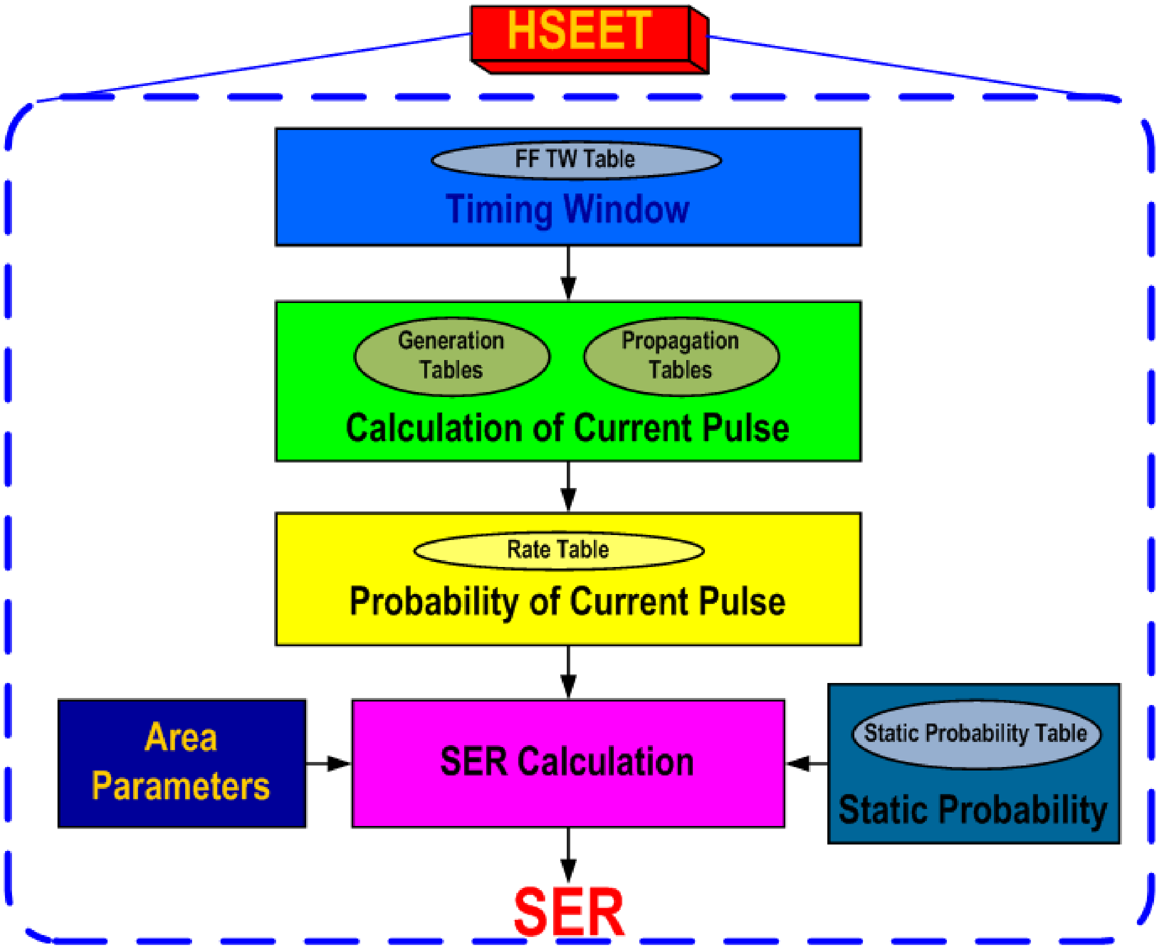


Figure 2.9. HSEET

based approach over gate level methods. Comparison across tools were not possible due to unavailability of data for all the designs. Exact comparisons across tools are not possible due to difference in methodologies and dissimilar platform setups. However, from [13], [15] and [18], we observe that our runtimes are faster for an adder in [13] and multipliers in [15]. Our tool is also very closely comparable or faster than [18] which lists the runtime for one input vector. Table 4.1 shows the SER results of various components such as muxes, decoders, adders and multipliers computed using the proposed methodology along with the run time in minutes for each design. For most designs, we observe that the run time is of the order of only few minutes for most of the designs. The methodology is extremely efficient with very less run time for wide number of current pulses as shown in table 2.2 and 4.1.

Table 2.1. Soft Error Rates of Hierarchical Circuits

Circuit	# of gates	# of Inputs	# of Outputs	SER	Run time(min)
Multiplexers					
mux4	12	6*	1	8.249X10^{-4}	<0.1
mux8	28	11*	1	3.9169X10^{-3}	0.15
mux16	60	20*	1	8.507X10^{-3}	0.4
Decoders					
dec2:4	15	3*	4	1.2985X10^{-3}	<0.1
dec3:8	35	4*	8	1.580X10^{-3}	0.1
dec4:16	75	5*	16	2.7903X10^{-3}	0.9
Adders					
rca4	20	8	5	7.439X10^{-4}	0.8
rca8	40	16	9	3.026X10^{-3}	5.5
rca16	80	32	17	7.565X10^{-3}	46.5
Parallel Multipliers					
p4x4	122	8	8	8.742X10^{-4}	1.8
p8x8	478	16	16	1.229X10^{-3}	17.9
p16x16	1892	32	32	1.97X10^{-3}	117.5
Array Multipliers					
a4x4	102	8	8	3.119X10^{-4}	3.5
a8x8	792	16	16	6.066X10^{-4}	24.5
* - includes control or enable signals also					

Table 2.2. Comparison of Speedup

Circuit	HSPICE(min) [14]	SEAT-LA(sec) [14]	HSEET(sec)
c17	426	10.43(2450X)	1.03(24815X)
rca4	872	252.3(208X)	15.6(3354X)
a4x4	–	273.6	26.2

Figure 2.10 shows us the comparison of SER of designs when TGFF and HLFF

is used as the register element. We observe that the SER of designs with HLFF are much higher than the ones with TGFF due to higher timing windows for same pulse-widths and HLFF's capability to latch on smaller pulse-widths than TGFF. Figure 2.11 shows the SER/BIT analysis for the designs. It is observed that the SER/Bit decreases for decoders, parallel and array multipliers while it increases for RCAs. This is due to the decrease in the number of susceptible nodes (and therefore the area) closer to the flip-flop to the total area of the circuit which governs the circuit SER in equation 2.7. However, the total number of susceptible nodes increase (and their area) in RCAs and due to longer paths available for an error to propagate, the SER/Bit for RCA increases. SER of muxes increase as the number of inputs increase due to the increase in the number of paths connected to the single output.

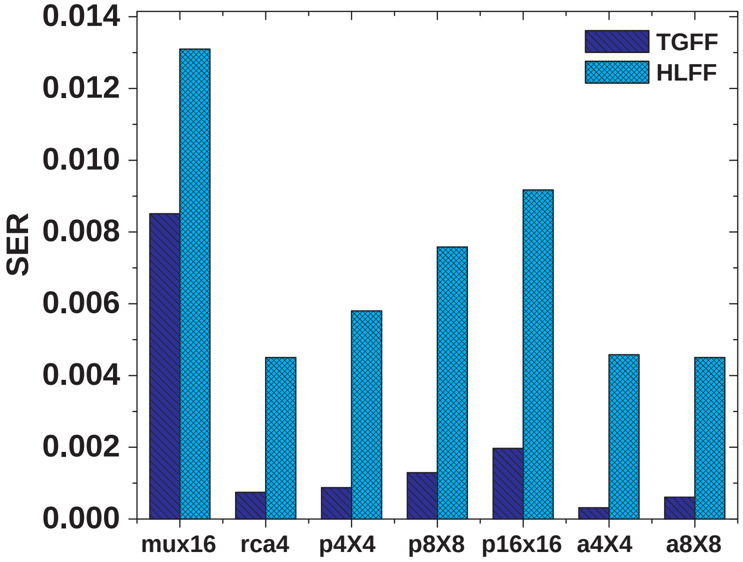


Figure 2.10. Comparison of SERs for TGFF and HLFF

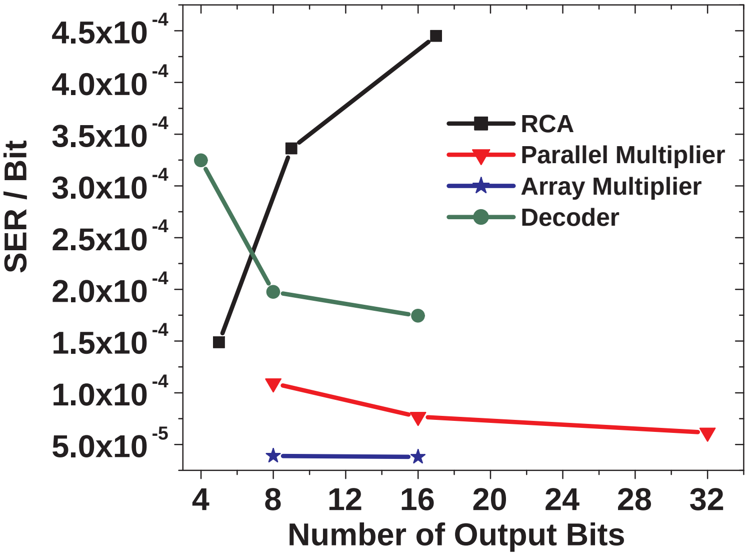


Figure 2.11. Analysis of SER / BIT

2.6 Negative Bias Temperature Instability (NBTI)

The physics behind NBTI and its underlying phenomena has been widely investigated by researchers for many years, with more recent work appearing in [25, 26, 27, 28, 29, 30, 31]. This includes the emergence of NBTI studies and modeling for combinational and sequential circuit elements, along with memory elements. In [32, 33] Wang, et. al performed a study of NBTI on both sequential and combinational circuits, developing gate-level models based on the Berkeley Predictive Technology Model (PTM) [20]. The impact of NBTI on the performance degradation and SNM loss of SRAM cells was studied in [34]. In addition to understanding the underlying causes of NBTI, and its effects at both the physical and circuit level, work has also been done to improve a circuits lifetime when faced with NBTI degradation. Authors in [35] studied the impact of NBTI on logic circuits and proposed mitigation techniques. NBTI aware synthesis was proposed in [36] to reduce the effect due to NBTI. Finally, the authors in [37] proposed a method to identify the critical gates and strengthen them. In addition, at the micro-architectural level, authors in [38] introduce techniques to reduce NBTI effects of structures found within microprocessors.

2.7 NBTI: Physics & Modeling

2.7.1 Overview

Negative-bias-temperature instability (NBTI) is a result of the continuous generation of traps at the Si/SiO_2 interface of the PMOS transistor. The interaction between inversion layer holes and hydrogen passivated Si atoms breaks Si H bonds created during the oxidation process, creating interface traps and neutral H atoms. These new H atoms can then form H_2 molecules, which either diffuse away from the interface through the oxide or can anneal an existing trap. The general physical mechanism of NBTI is explained quantitatively through the reaction-diffusion model.

From a circuit standpoint this translates to the degradation of the PMOS transistor when its gate voltage is negative, or its input is 0. This effect is exacerbated by the fact that the SiH bonds begin to break more easily over time leading to the PMOS device becoming more vulnerable to these physical effects. Broken bonds that do not anneal act as interface traps and result in an increase in the threshold voltage (V_{TH}) of the transistor. This effectively causes a slowdown in the speed of the PMOS transistor, leading to reduced performance over time within digital circuits. In addition, NBTI is of greater concern as technology scales due to higher operating temperatures and the use of thin-oxides.

2.7.2 Reaction-Diffusion Model

NBTI can be described as the generation of interface charges (N_{it}) at the Si/SiO2 interface. This process is described through the Reaction-Diffusion (R-D) model which consists of two critical steps:

2.7.2.1 Reaction

The Si-H bonds at the Si/SiO2 interface are broken under vertical electrical stress. Consequently, interface charges are induced, which cause an increase of Vth. Given the initial concentration of the Si-H bonds, i.e., N0, and the concentration of the inversion carriers, i.e., P, the generation rate of Nit is given by [26].

$$\frac{dN_{it}}{dt} = K_f(N_0 - N_{it})P - K_r N_H N_{it} \tag{2.8}$$

With continued reaction, two H atoms combine to generate a H2 molecule. The concentration of H2, i.e., H2 N is:

$$N_{H2} = k_H N_H^2 \tag{2.9}$$

2.7.2.2 Diffusion

The reaction generated species diffuse away from the interface toward the gate, driven by the gradient of the density. This process influences the balance of the reaction and is governed by:

$$\frac{dN_H}{dt} = D_H \frac{d^2 N_H}{dx^2} \tag{2.10}$$

By integrating these two steps together, we can solve the models for Vth increases (Vth), for both static and dynamic NBTI, as shown in Table 1, where A and Kv are functions of the vertical electrical field (T) and the carrier concentration (C = exp (-Ea/kT)/T0), respectively; .., ..1 and ..2 are constant parameters. There are also several fitting parameters in the model which are responsible for the degradation under various temperatures. These fitting parameters in the model may change from one technology to another, but they are relatively insensitive to local process variations at the transistor level. More details on the R-D model along with experimentally validated results can be found in [33].

Using V_{TH}, as computed by the R-D model, a first-order approximation for the propagation delay of the gate (T_d) can be given by [**?**].

$$Td = a_0 + a_1.\Delta V_{TH} + a_2.C_l \tag{2.11}$$

Where a_o is the intrinsic delay of the gate without NBTI degradation, a_1 and a_2 are constants, and ΔV_{th} is the degradation due to NBTI in the PMOS transistor. Given that a gate can have multiple PMOS transistors, the NBTI degradation of each of the transistors can be different based on their inputs. Because characterizing the gate delay capturing these differences can become exceedingly time consuming two corner cases are considered. The first calculates the circuits timing degradation where the ΔV_{th} is based on the least-most degraded PMOS transistor. The second considers the most degraded PMOS transistor in the gate to determine the circuits timing. The constants in the above equation are obtained through HSPICE circuit simulations which are used to create a library of primitive cells. Each of the gates within the cell libraries have their nominal and degraded timing characterized for various temperatures, supply voltages, and across several technology nodes (65, 45, and 32 nm).

2.8 Aging Assessment

The evaluation of NBTI stress on modern microprocessor architectures requires an integrated framework that consists of architectural simulation, gate level timing analysis, and pre-characterized gate libraries. In this section, an approach to synthesis-level evaluation is presented, Figure 2.12 depicts the complete framework, including New-Age, when configured for micro-architectural studies, tying together the basic New-Age analysis tool and the micro-architectural simulation framework. The separation of the analysis tool-flow from the micro-architectural design and simulator allows for other micro-architectural simulators to be integrated without requiring any major changes to the overall framework.

In order to simulate the aging due to NBTI, the New-Age analysis tool requires several parameters, such as operating conditions (temperature, voltage, and time ranges), the technology node, and the design files (HDL or a specified net-list of the design). In addition, input value probabilities are needed for determining a gates input switching used to later determine degradation of a particular gate in the net-list.

Using the operating conditions and specified technology the tool begins by performing library characterization if needed. These custom cell libraries are created automatically for the various operating conditions using the first-order approximation to gate delay and NBTI models as described in Section *****. The cell libraries contain nominal cells with gate delays obtained by setting $\delta V_{th} = 0$ in the first order gate delay model to serve as a baseline and degraded cells with gate delays based the appropriate δVth resulting from the gates input probabilities. It is important to note that since library characterization can be time consuming it is only carried out with newly specified operating conditions, technologies, or changes to the degradation models. Therefore, for subsequent runs of a design, the libraries do not need to be re-characterized each time.

Following cell library creation, the tool then performs synthesis if HDL of the component is received. Synthesis is performed using the cells with nominal delay to produce a net-list of the desired component. If a net-list is specified, synthesis is skipped and the tool proceeds immediately to propagating the top level switching probabilities to the internal nodes of the net-list. As a result of signal probability propagation, two net-lists for aging analysis are created corresponding to the lower and upper bounds for NBTI vulnerability. For the lower bound analysis, the static probability for each of the inputs of a gate is set to correspond to that of its least stressful input, the input with the smallest static probability of 0. Similarly, for the upper bound analysis, each of the inputs of a gate corresponds to that of its most stressful input, the input with the highest static probability of 0.

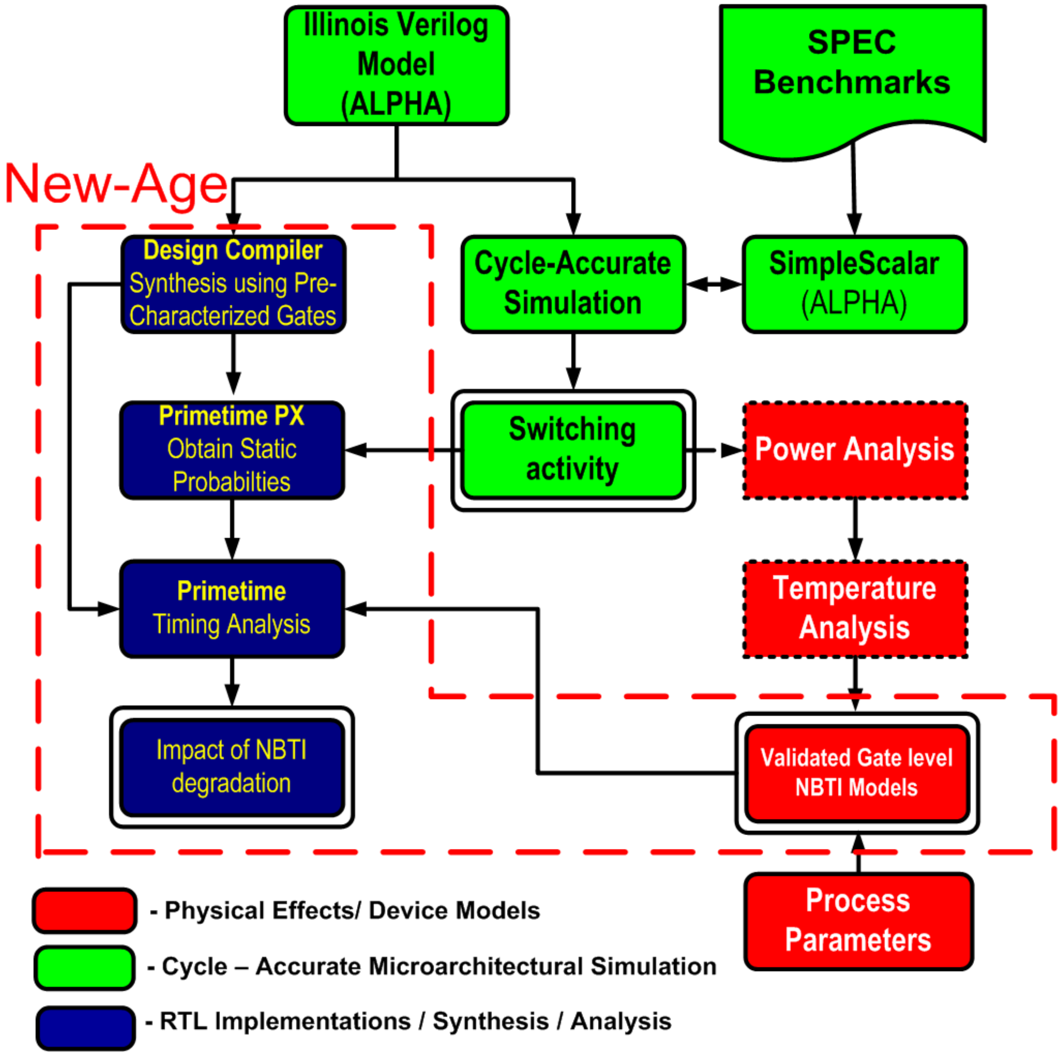

Illinois Verilog Model (ALPHA)
SPEC Benchmarks
New-Age
Design Compiler Synthesis using Pre-Characterized Gates
Cycle-Accurate Simulation
SimpleScalar (ALPHA)
Primetime PX Obtain Static Probabilties
Switching activity
Power Analysis
Primetime Timing Analysis
Temperature Analysis
Impact of NBTI degradation
Validated Gate level NBTI Models
Process Parameters
- Physical Effects/ Device Models
- Cycle – Accurate Microarchitectural Simulation
- RTL Implementations / Synthesis / Analysis

Figure 2.12. NBTI Micro-architectural Assessment Framework

This results in three final netlists corresponding to the baseline non-degraded design, the worst case design, and the best case design. The best-case net-list assigns probabilities to the gates input which minimizes the delay through a particular gate. Conversely, the worst-case net-list assigns probabilities to the gates input which maximizes the delay through a particular gate. The net-lists corresponding to the baseline non-degraded design and the worst case and best case aged designs are fed to Synopsys Primetime to perform static timing analysis using the custom cell libraries. The timing analysis reveals the increase in critical path of the circuit as well as the reduction of slacks in the non-critical paths. In some cases our analysis revealed the emergence of a new critical path (a non-timing critical path changing to a critical path) due to aging.

As shown in Fig 2.12 the static signal probabilities of the internal nodes and the gate temperature are obtained using a hybrid simulator consisting of the Illinois Verilog Model (IVM), a cycle-accurate register transfer level (RTL) model of the processor and Simplescalar [39], a functional simulator. The RTL model is used for monitoring the static probabilities of the signals at each of the internal nodes of a circuit while the functional simulator is used for accelerating the architectural states to desired sample points for fast simulation speeds. The RTL model augments the IVM to capture the switching probabilities of all the nodes in the design. The switching activity is translated to input state probability as well as used for power estimation. The power estimates along with the architectural layout are fed to a thermal estimation tool, Hotspot 4.0 [40], to estimate the temperature. Once all the operating conditions and input state probabilities have been obtained they are then provided to New-Age to perform the final aging analysis of the design.

Separating the framework in this way enables the simulation of different architectural configurations and application suites while easily allowing for alternative simulators and architectures to be used. Further, it permits the exploration of changes in temperature and voltages. These features can be useful for evaluating the NBTI degradation in conjunction with circuit and architectural support designed for other constraints such as power consumption.

2.9 Experimental Results

In this section, New-Age is used to carry out NBTI aging analysis for two designs. First, the New-Age simulation framework is targeted at a processor architecture similar to that of the Alpha 21264[41] and the AMD Athlon [42]. The processor architecture is based on the IVM, a synthesizable cycle-accurate register transfer level (RTL) model [43]. Within the IVM several pipeline stages are examined for NBTI vulnerabilities. In the study, workload driven analysis, including impacts of temperature, voltage, and technology scaling on NBTI-induced degradation are performed. Second, several sub-components found within a typical ALU have their lifetimes evaluated.

2.9.1 Evaluation of Processor Pipeline

This section provides the evaluation of the Illinois Verilog Model (IVM), a superscalar, dynamically-scheduled pipeline which executes a subset of the Alpha instruction set. The IVM consists of several pipeline stages which are detailed in [43].Within the IVM, the NextPC, Fetch, Decode, and Rename stages are considered for vulnerability analysis. Within those stages memory structures were removed and replaced with the corresponding activity traces to overcome limitations due to the synthesis tools. Each of the stages was evaluated for NBTI under several conditions including temperature, supply voltage, and over technology generations. The SPEC2000 benchmark suite [44] was used for providing workloads for processor evaluation[1].

Figure 2.13 depicts the average NBTI-induced delay degradation found through New-Age for the IVM pipeline stages at the 65 nm technology node after 10 years of operation. Also reported from the analysis are the best- and worst-case delay degradation ranges as described in Sect. ******. The results show that at 65nm Fetch1 undergoes 3.5% degradation on average while Fetch2 (the highest temperature unit) sees only 3%. Average workload-dependent temperature results for these stages have also been reported and are shown in figure 2.13. This graph highlights that a ±2.5% variation in temperature leads to no visible correlation between a structures temperature and its corresponding degradation. For example,

[1] Traces from four representative benchmarks are presented for clarity: Art, Bzip2, Crafty, and Mesa.

Fetch1 which reports the lowest temperature surprisingly has the highest amount of degradation. Alternatively, NextPC, which has an elevated temperature due to its location in the floor-planned design, experiences the least amount of degradation of all the structures. This implies that while temperature is an important parameter for contributing to the transistors V_{th} shifts, an accurate NBTI characterization tool must take into account factors, such as input switching, that also contribute to the degradation of the circuit.

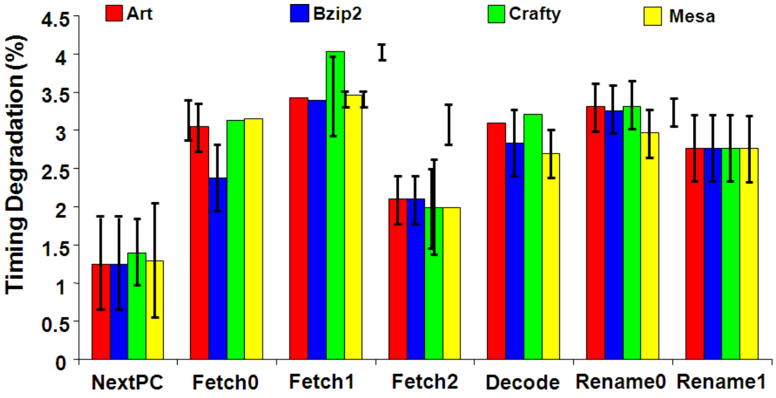


Figure 2.13. NBTI-induced delays on pipeline stages within IVC for 65nm technology

In addition to performance degradation and temperature analysis the New-Age framework further allows for more in-depth analysis of a given circuit. This includes critical path analysis before and after degradation effects are considered. This is a fundamentally important aspect of the tool because it was recently found in [37] that critical and sub-critical paths are those which typically define the critical path after degradation effects are included in the delay measurements. Using New-Age it is possible to analyze these paths for the pipeline stages in a

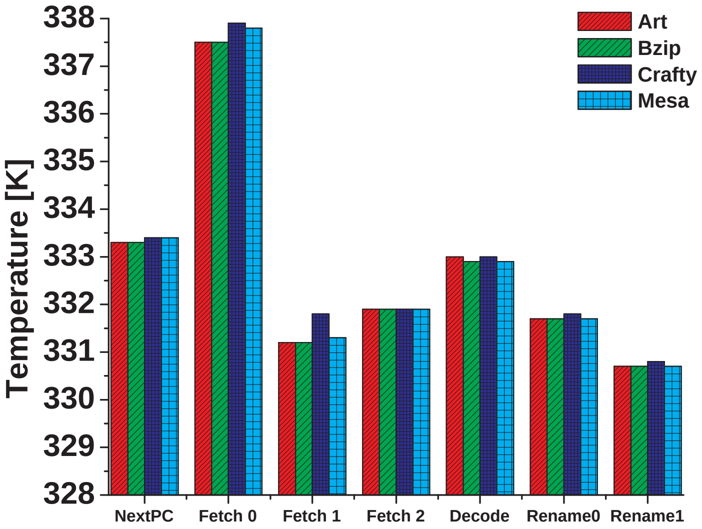


Figure 2.14. Temperature ranges with Derived workload-dependent steady state temperature and NBTI-induced 10 year timing degradation. An ambient temperature of 300K was used for Hotspot

given micro-architecture and determine how structures are affected across multiple applications.

2.9.1.1 New-Age Predictive Analysis

New-Age also includes several features which can be used to assist circuit designers determine the vulnerability of their designs to various conditions. These features allow for degradation analysis of circuits under various supply voltages, technology nodes, and temperatures.

1. *Effects Due to Supply Voltage:*

   A common design technique to reduce power consumption is using multi-V_{dd}, a technique where circuits operate at several lower supply voltages within the same chip. In addition to helping reduce temperatures, reducing the supply voltages also affects the amount of NBTI degradation a circuit undergoes. Figure 2.15 shows the maximal performance degradation due to NBTI for the Fetch1 stage as the supply voltage is reduced from 1.1 to 0.8V operating at several temperatures, for the 65 nm node. It can be observed that a decrease in supply voltage can help mitigate the performance degradation for a range of temperatures. In all pipeline stages there was approximately 30, 50, and 70% improvements in the amount of performance degradation as voltage scaled from 1.1 to 0.8V.

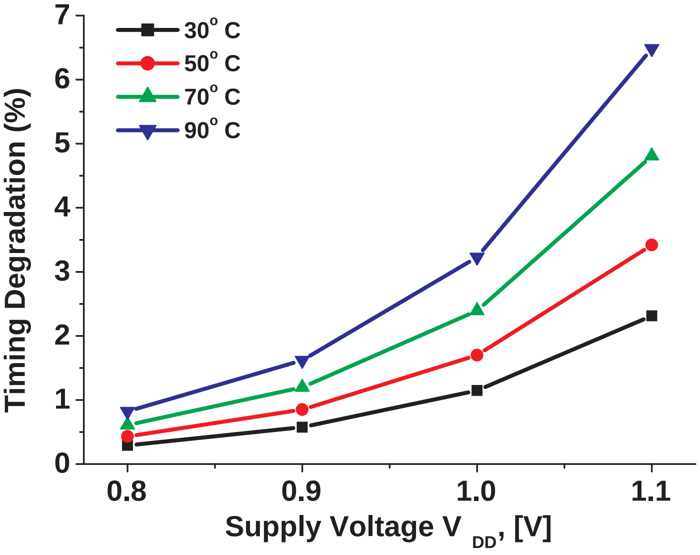


Figure 2.15. Voltage scaling effects on NBTI degradation for Fetch1 stage in 65nm technology

2. *Technology Scaling:*

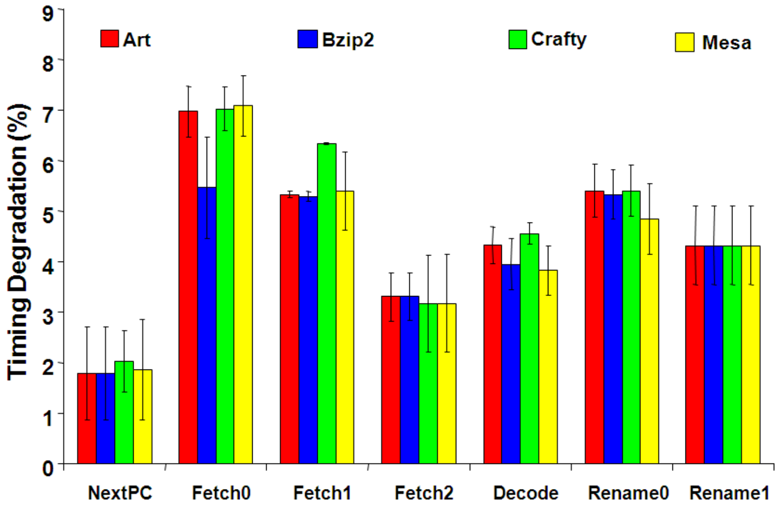


Figure 2.16. NBTI-induced delays on pipeline stages within IVC for 45nm technology

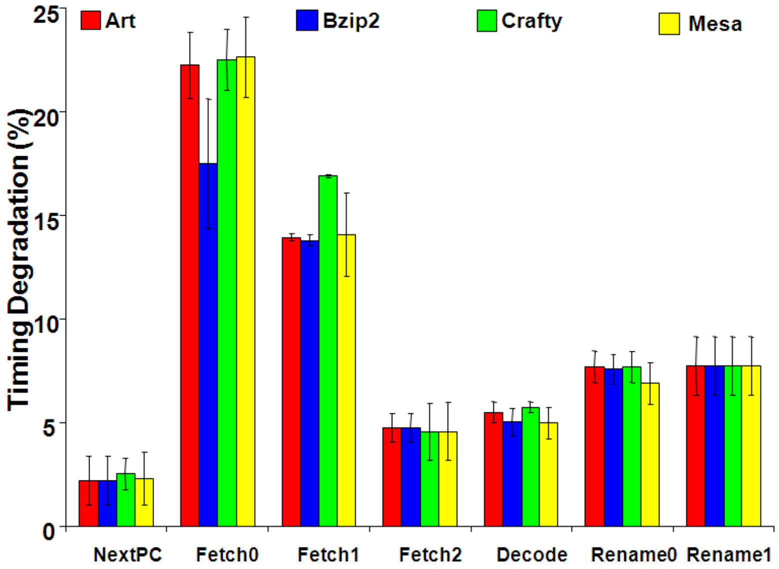


Figure 2.17. NBTI-induced delays on pipeline stages within IVC for 32nm technology

Figures 2.16 and 2.17 shows the NBTI degradation results for the seven pipeline stages for 45 and 32nm technologies respectively with technology dependent parameters scaled according to recent projections by ITRS [45]. The results from New-Age show that Fetch0 now exceeds the degradation

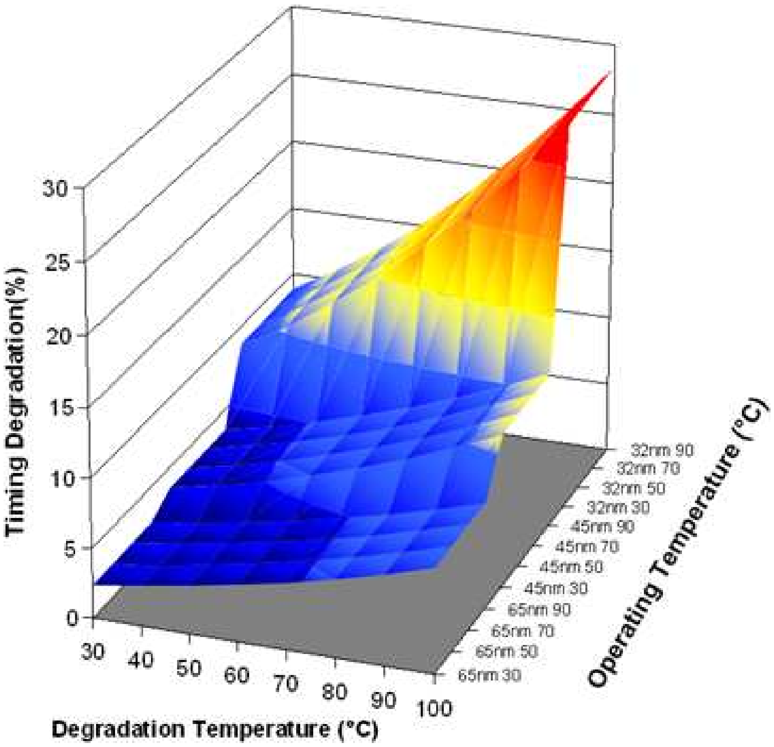


Figure 2.18. Temperature effects on NBTI-induced degradation for Fetch1 stage across technologies.

of Fetch1, with performance degradation reaching 7.5 and 23.5% for 45 and 32nm technologies respectively. In addition, the results also show a disproportionate increase in the amount of performance degradation for the Fetch0 and Fetch1 stages over the other stages as technology scales down from 45 to 32 nm. Figure 4e explains these results. Figure 4e depicts the change in critical path delay for each of the pipeline stages delay as technology scales across several of the benchmarks, with the delay being normalized to 65 nm. Included in the measurements are the effects of scaling on a stages power density and temperature, used in conjunction with updated cell libraries for each technology, to account for changes in the corresponding gate delays. The graph also shows the proportion of NBTI degradation superimposed onto the delay of each structure. This shows that as technology scales, the baseline critical path delays change non-uniformly from one stage to the next and, in most cases, decrease when compared to the previous technology. At the same time, the amount of delay due to NBTI increases as technology scales, resulting in certain structures becoming more sensitive to NBTI degradation. This is what is observed for the Fetch0 and Fetch1 stages. As technology scaled these two structures had critical path delays which significantly reduced, while simultaneously they had NBTI delays which increased the most. This resulted in those two structures performance degradations being increased up 23.5 and 18% respectively overall.

3. *Effects Due to Temperature:*

   Figure 2.18 shows timing degradation of the Fetch1 stage across various degradation and operating temperatures, and with different technologies. This graph assumes that for the first 50% of the circuits lifetime (5 out of 10 years) the circuit is operating at one temperature undergoing degradation (along the x-axis, denoted degradation temperature) while for the remaining period of its lifetime it is operating at another temperature (along the y-axis, denoted operating temperature). This demonstrates how temperature can exacerbate NBTI-induced timing degradation when extremes in temperature are considered. This differs from the temperature results seen so far, where small variations in temperatures are shown across structures, but peek temperature and hotspot phenomena are absent. These results indicate that purely reducing peak operating temperatures may not be adequate for achieving longer lifetimes. For example, at 32 nm, even when the Fetch1 stage only reaches temperatures between 30 and 50°C during its lifetime, the circuit still undergoes significant timing degradation in the range of 10.15%. This implies that other techniques need to be explored which can further help reduce the effects of NBTI, such as supply voltage scaling and supply gating.

2.9.2 Evaluation of ALU Sub-Components

In this section, several of the common components which make up an ALU are investigated for NBTI vulnerabilities. Here we demonstrate the frameworks ability to quickly analyze different designs under a variety of operating conditions. For this experiment the main micro-architectural simulator is removed and the New-Age tool is supplied with HDL of custom implementations of several adders, multipliers, and shifters[2]. For demonstration purposes the operating temperature was set at 100°C, and results were gathered for 65, 45, and 32nm technologies.

1. *Performance Degradation:*

[2] 16/32-bit Brent Kung and Kogge Stone adders; 4/ 8,/16 array, and 9x9 Booth multipliers; 4/ 8/16 parallel multipliers; 16/ 32/64 bit log-shifters.

The performance degradation of ALU components is depicted in figure 2.19. with results shown for a constant circuit operation of 10 years. Adders and multipliers have a similar amount of performance degradation, with less than a ±3% variation amongst them, while the log shifters did not follow this trend and instead had different amounts of degradation depending on the operand size (i.e. 16, 32, or 64 bits). The critical paths of the log shifter were further examined and it was found that this resulted from changes in the critical path and how it affected the input value probabilities at the internal nodes. In particular, the 16- and 32-bit shifter had the same gate sequence on the critical path. However, between these two circuits the switching at the internal nodes was not the same, ultimately leading to differences in performance degradation. After performing a more in-depth analysis of the paths within each log shifter it is possible to conclude that the 32-bit shifter is more vulnerable to NBTI-induced failure then the 16-bit shifter. The 64-bit log shifter proved to be even more vulnerable where the gate sequence, critical path length, and internal node switching changed.

2. *Input-Vector Control:*

One technique that has been frequently mentioned as a way to mitigate the effects from NBTI is input vector control (IVC). IVC is a technique where specific inputs are purposely applied to a circuit in order to minimize the number of internal nodes which will degrade. To this end three criteria

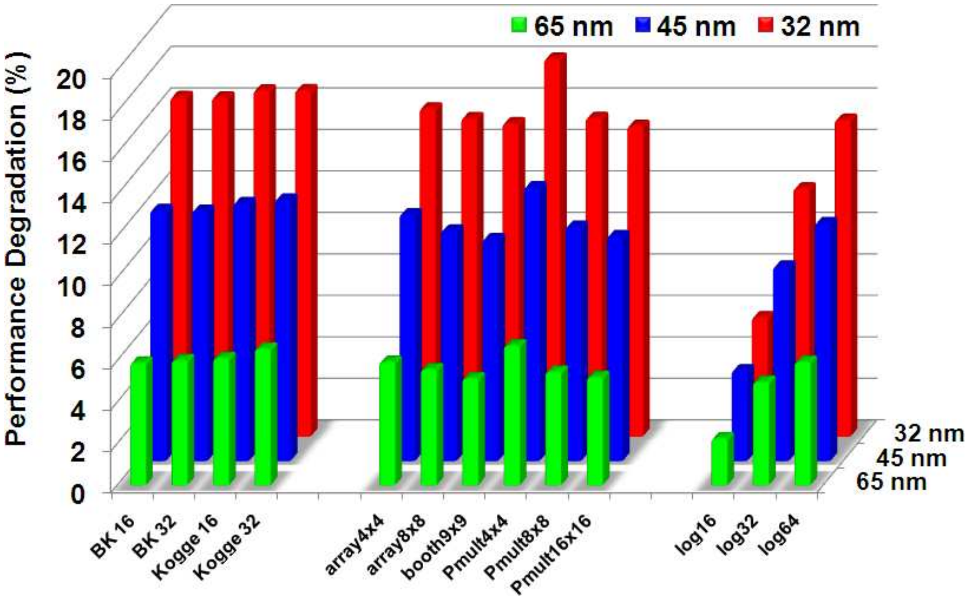


Figure 2.19. NBTI induced path delay across ALU components

are used to measure IVCs performance: CapacityThe potential for reducing degradation; EffectivenessThe amount of degradation improvement; Component IdlenessHow often IVC can be applied.

Capacity:

Figure 2.20 shows an alternate view for the performance degradation of the ALU components considering that they are idle for some portion of their lifetime. This idle period is identified as the active-to-standby ratio (RAS), which is defined as the fraction of the time the circuit spends in active mode to the time the circuit is idle. In this analysis a large idle period for RAS (1/9) was chosen, which is effectively 1 active year per 10 years of operation. During active-mode all inputs were assumed to have a 0.5 input probability of one, while in standby-mode two alternatives we considered. The first option is idle degradation analysis, defined to be when all internal nodes are set to 1. The second possibility is always degrading degradation analysis, corresponding to all internal nodes being a 0. Note that the timing degradation is not linear with stressing. It is most dominant in the initial stage [32].

To evaluate IVCs capacity, the difference between the amount of degradation for the best and worst cases is calculated. These results show that IVC does have the capability to cause improvement in the amount of performance degradation which stays relatively constant in most cases. More specifically, "perfect" IVC can result in up to 40% improvement for adders and multipliers and up to 70% improvement for the 16-bit log shifter. This consistency exists across modules and technologies because, under constant operating conditions, the active-to-standby ratio is effectively amplifying or reducing the effects of NBTI proportionally to the RAS value. This causes the improvement due to IVC to be very optimistic since the idle periods impact on a particular components power and temperature is not taken into account.

Effectiveness:

The performance of IVC is also impacted by the amount of degradation that still exists after IVC is applied. Referring back to figure 2.19, the ideal effectiveness of IVC can be associated to the best-case degradation. These results show that the amount of degradation in most circuits can be reduced to no less then 7% for 45nm and 10% for 32 nm. In this case IVC proves to be effective, depending on the size of the initial guard-band. For example, if large guardbands were used (tolerating 10% degradation) then the component would be able to function longer. However, if IVC was intended to be used as a way to reduce the guardband, the lifetime of the device will still suffer. More significantly, guardbands would still have to increase across technology generations since performance degrades more overall.

Component Idleness:

Figures 2.21, 2.22, 2.23 shows results for the Kogge Stone 32-bit adder, 16X16 Parallel Multiplier, and 16-bit log shifter at 45 nm, as the active-to-standby ratio is varied from 1/9 to 9/1. The most immediate observation is that decreasing the amount of idle time decreases the amount of improvement. This is an expected result because if no time exists in which the input vector can be applied then there is no opportunity for improvement. More interesting is the point at which IVC will no longer yield any effective improvement. For the Kogge Stone adder and 1616 parallel multiplier, this point appears to be reached at an active-to-standby ratio of 1/1. This corresponds to around 20% performance improvement, where best-case degradation decays to 10% degradation. In the case of the 16-bit log shifter, the break-even point is different. First, observe that in general the best-case and worst-case degradation due to IVC grow towards the original degradation reported as the active-to-standby ratio decreases. In the case of the log shifter, the original degradation was more indicative of the best-case degradation, so the ratio between best and worst case remained large. Following trends on how the other components degrade it would be expected that most modules will see similar IVC improvement drop off as with the Kogge Stone adder and parallel multiplier. Taking these three factors into consideration a conclusion on the performance of IVC can be discussed. On the one hand the capacity for improvement clearly exists. It was shown that at 45 and 32nm that there can be a 4070% improvement in a circuits degradation if an ideal input vector had been applied. However, the resulting performance degradation was not sufficient to reduce guardbands, only sustain them for a particular technology

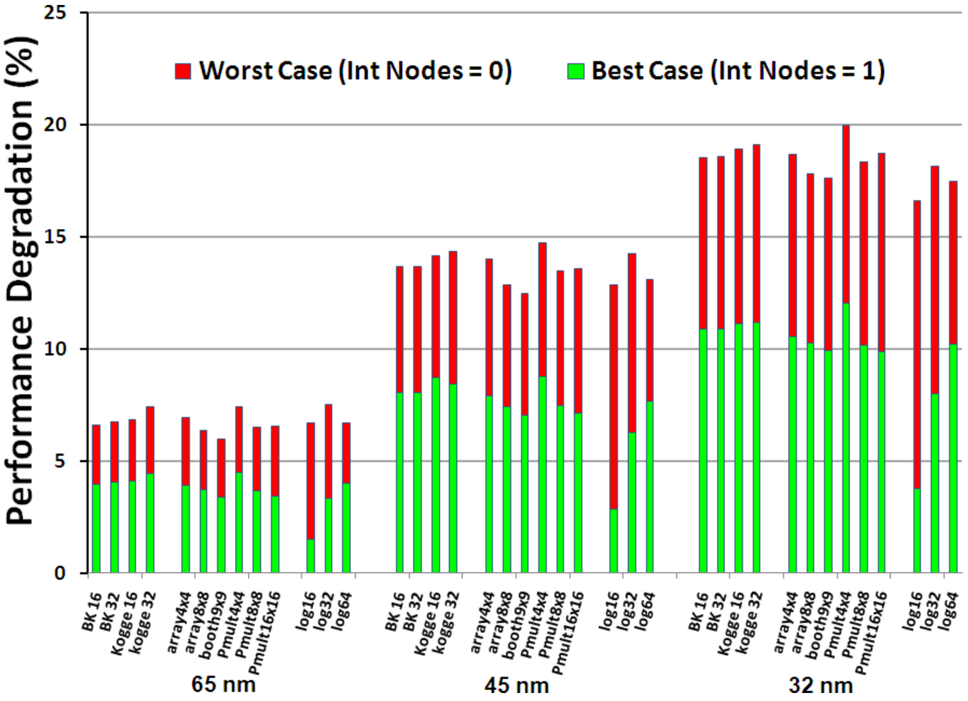


Figure 2.20. NBTI induced path delay across ALU components and technology with RAS

node, given that they had a significant amount of idle time. In addition, the minimal guardbands from 65nm technology could not be sustained across 45 and 32nm technology generations. This makes the use of larger and larger guardbands necessary for smaller technology generations, defeating the goal of IVC, to reduce degradation and effectively, the required guardband.

Overall, IVC appears to be a short-term solution. In the cases where it could be applied a thorough knowledge of the workload being run along with a prior knowledge of path degradation must be known. Further, in cases where a critical component is not significantly idle, effort would need to be placed around increasing its idle periods using various hardware and software techniques. Even under these circumstances there is no guarantee that IVC will provide the savings necessary to meet lifetime requirements once the assumption that there exists a perfect input vector is removed. Finally, a potentially defeating factor of IVC as a long-term solution is that it is unable to reduce, or sustain, the amount of degradation from one technology generation to the next.

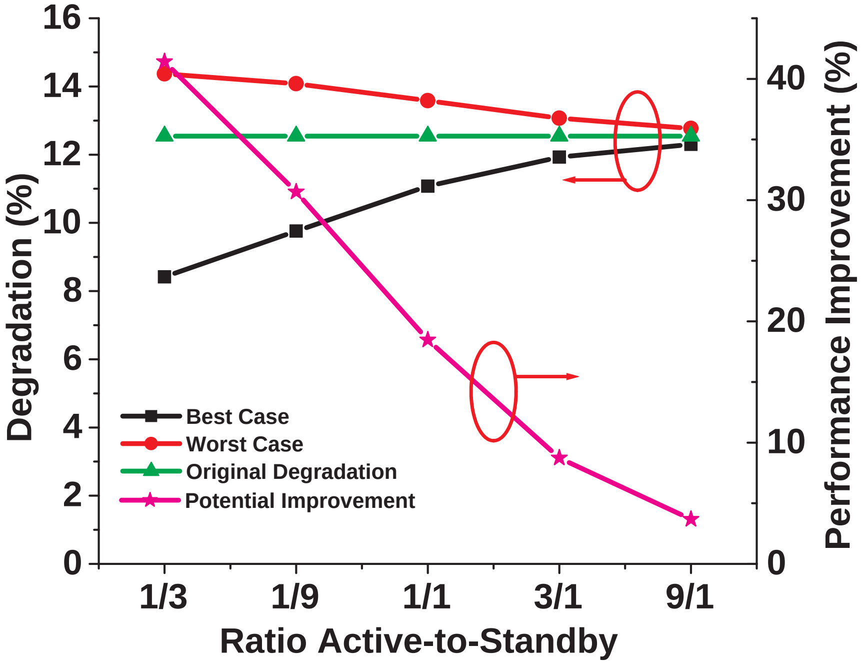


Figure 2.21. Performance degradation with variation in RAS for a 32 bit Kogge-Stone Adder

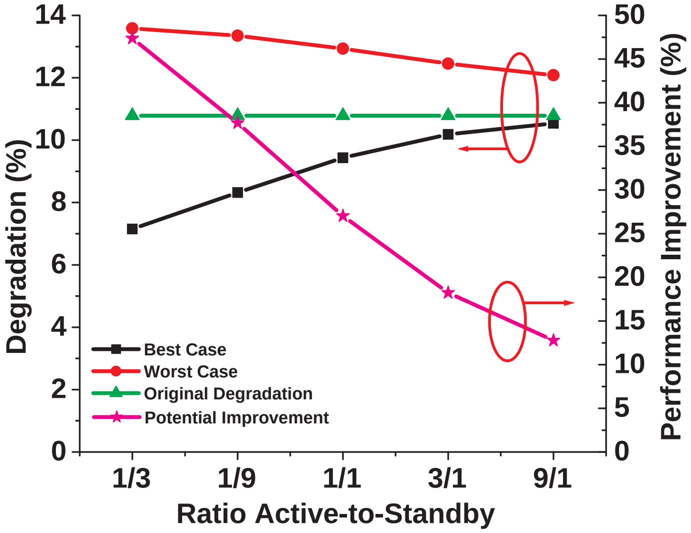


Figure 2.22. Performance degradation with variation in RAS for a 16 bit Parallel Multiplier

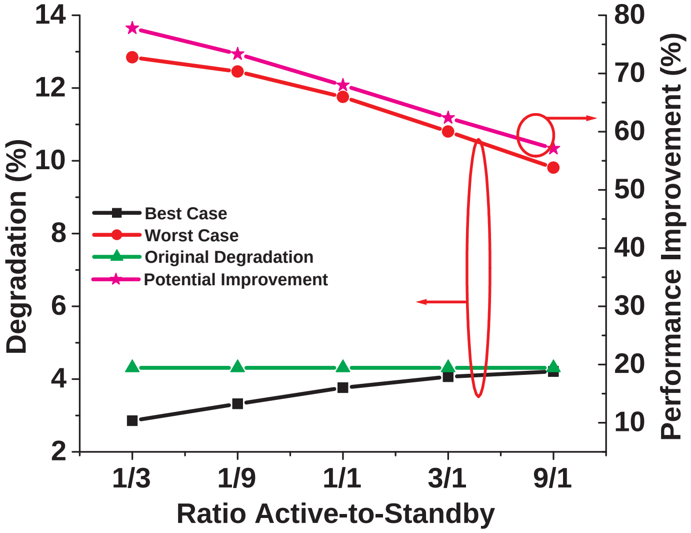


Figure 2.23. Performance degradation with variation in RAS for a 16 bit Log Shifter

Chapter 3

Impact of Aging

3.1 Introduction

Flip-flops are one of the most important structures in a micro-processor. Flip-flops are clocked storage elements which hold the states in sequential circuits sampled at a preferred clock edge [46]. They play a vital role in the design of synchronous circuits and clocking of the system. The timing characteristics of the flip-flops decide the frequency of operation of the circuit. Thus, it is imperative for the flip-flop timing characteristics to be unperturbed by external factors such as variations and aging. In this paper, we study the effect of NBTI on the timing characteristics of various low power and high performance flip-flops.

NBTI has different effects on combinational and memory circuits and they have been widely studied by various researchers in [35][34]. Various methods such as transistor up-sizing and gate replacement techniques for combinational circuits have been proposed in [35]. Performance evaluation and reduction in static noise margin (SNM) of memory circuits due to NBTI have been done in [34]. Adaptive body biasing has been shown as a promising compensation technique to combat NBTI in [47]. Onchip monitors to compute the degradation due to NBTI have been proposed in [47] and [48]. [32] claim that sequential circuits are not affected by NBTI. However, authors in [49] have shown it to be a problem and that upsizing the NBTI prone transistors increase the lifetime of the flip-flops. However, they consider the input data probability as 0.5 which does not exhibit the worst case condition as shown in the later sections. To the best of authors' knowledge, this is the first work that studies the effect of NBTI on different types of flip-flops. The major contribution of this work is a comprehensive analysis of the timing characteristics of the various low power and high performance flips-flops affected by NBTI. This analysis will help the designers in choosing the right flip-flop judiciously during the design time ensuring long term reliable circuits.

3.2 Motivation

3.2.1 Long Term Prediction Model

We use the long term threshold degradation model explained in [33][37] to calculate the change in threshold voltage of a PMOS transistor.

$$\Delta V_{th} = (\sqrt{K_v^2.T_{clk}.\alpha/(1-\beta_t^{1/2n})})^{2n} \tag{3.1}$$

where

$$\beta_t = 1 - \frac{2.\xi_1.t_e + \sqrt{\xi_2.C.(1-\alpha).T_{clk}}}{2.t_{ox} + \sqrt{C.t}} \tag{3.2}$$

Equation 3.1 calculates the ΔV_{th} of a degraded PMOS device where K_v is a function of electrical field, temperature and carrier concentration, n is the time exponential constant equal to 0.16, α (SP) is the signal probability of the input. α is the total amount of time the input to the PMOS transistor is LOW (Logic '0'). Note that the input data probability (DP) is the probability of the input D being HIGH (Logic '1') is equal to 1-α and henceforth, we will use DP for our illustration purposes. Figure 3.1 shows the change in threshold of a 65nm PMOS device over 5 years at T=27° C for various values of DP.

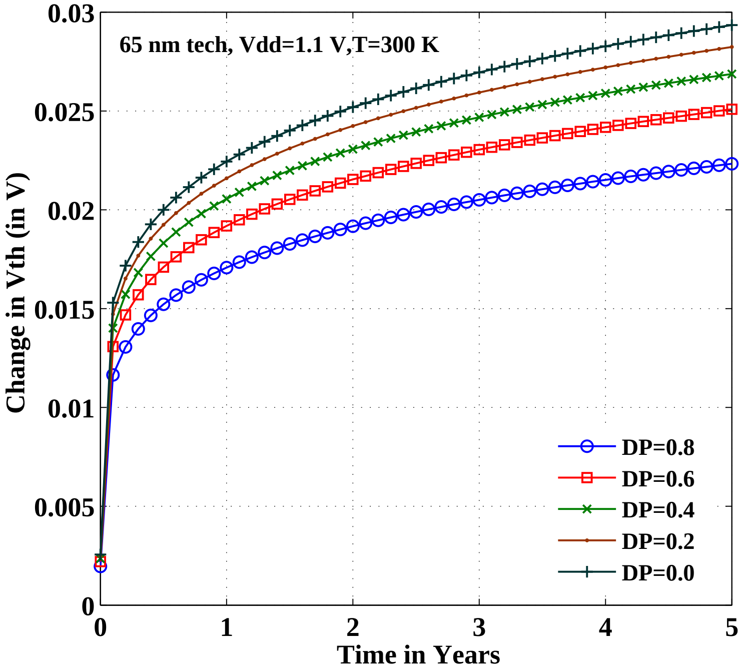


Figure 3.1. V_{th} degradation for 65nm technology for various input data probabilities (DP)

3.2.2 Flip-Flop Timing Metrics

The important timing metrics of flip-flops for consistent estimation of various parameters are described here. These parameters are inter-dependent and well-established metrics also explained in [50] [51].

- T_{su} is the setup time and is defined as the minimum time required for the input D to remain stable before the edge of clock CLK for the output to latch on to the correct value even under the worst case conditions.

- T_h is the hold time and is the minimum time for the input D to remain stable after the edge of clock CLK for the output to latch on the correct value.

- T_{cq} is the propagation delay from CLK to the output Q of the sequential element assuming that the input D arrived well early before the preferred egde of the clock CLK.

- T_{dq} is the propagation delay from the input D to the output Q assuming that the clock CLK arrived well early than the input D.

Figure 3.2 shows the important timing parameters of sequential elements. T_{cq} of the design increases monotonously as the data arrives close to the clock edge before the absolute setup time after which the operation fails. The region of failure as shown in figure 3.2 is the region when the input D changes from one state to another which results in latching of wrong output. T_{sua} and T_{ha} are the absolute values of setup and hold time beyond which faulty operation occurs. It is not unusual for a flip-flop to have negative setup time and zero hold time due to it's topology and design. Violation of a setup time constraint results in latching the wrong value on to the sequential element and can be rectified by decreasing the clock frequency. Hold time violations cannot be rectified and renders the chip useless. Buffers are usually added to increase the delay to eliminate hold time violations.

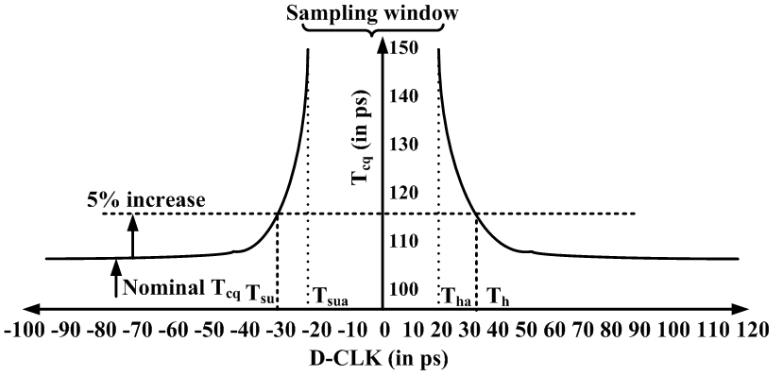


Figure 3.2. Typical timing characteristics of a flip-flop

3.2.3 Motivation

The minimum time period for the clock governing a pipeline stage with flip-flops has to satisfy equation 3.3 for correct operation.

$$T \geq T_{ff} + T_{logic} + T_{skew} \tag{3.3}$$

where T is the clock period, T_{logic} is the combinational logic delay between two pipeline stages and T_{skew} is the clock skew and $T_{ff} = T_{su} + T_{cq}$, where T_{su} is the setup time and T_{cq} is the CLK-Q delay. Various approaches are followed to set the value of T_{su} which is used to compute the value of T_{ff}. One approach is setting the setup time as the point at which T_{cq} is 5 or 10% more than the nominal T_{cq} as shown in figure 3.2. Another approach maintains the optimal setup time as the point at which T_{dq} is minimal as shown in figure 3.3 [51][52]. In this work, we analyze both the cases. We define case 1 as the approach where the setup and hold times of the design are set at the point of 5% increase in T_{cq} of the nominal value.

The point of minimal T_{dq} is represented as case 2. The value of T_{ff} is dependent on the type of transition of the output (0→1 or 1→0) and is taken as the maximum of both the transitions. In this paper, we have shown both the cases for explanatory purposes. The worst case conditions occur when there is no combinational logic between the two elements and the internal race immunity is shown in equation 3.4.

$$T_{cq} \geq T_h + T_{skew} \tag{3.4}$$

From equations 3.3 & 3.4, it is clear that the clock period is determined by T_{ff}. Equation 3.3 includes the worst case timing of T_{ff} for combinational logic delay for all operating conditions. Therefore, it is important to reduce (T_{ff}) which the flip-flop uses from the clock cycle to achieve higher performance. It should also be noted that as the data arrives closer to the clock edge, T_{cq} increases. Thus, the data cannot be allowed to arrive closer to the clock edge than the pre-determined time as this may result in increased T_{su} & T_{cq} which will result a wrong operation due to violation of equation 3.3. Due to NBTI, the data arrives later than expected due to the increase in the combinational logic delay which is also severely affected due to NBTI. So, it is imperative to understand the response of flip-flops to both late arriving signals and self degradation.

In figure 3.3, we observe that the T_{cq} of an aged Transmission Gate Master-Slave Flip-Flop (TG-MSFF) increases as time progresses. The absolute increase in the T_{cq} causes a need for increasing the clock period T. This in turn also increases T_{ff}

calculated for equation 3.3. Thus, we observe that under the worst case conditions, the minimum time period T in equation 3.3 could be violated making the pipeline stage faulty. The degradation due to NBTI is dependent on the input signal probability as seen from figure 3.1 and is explained in section 2.5. Thus, it is imperative to analyze the impact of NBTI on various high performance and low power flip-flops under various conditions.

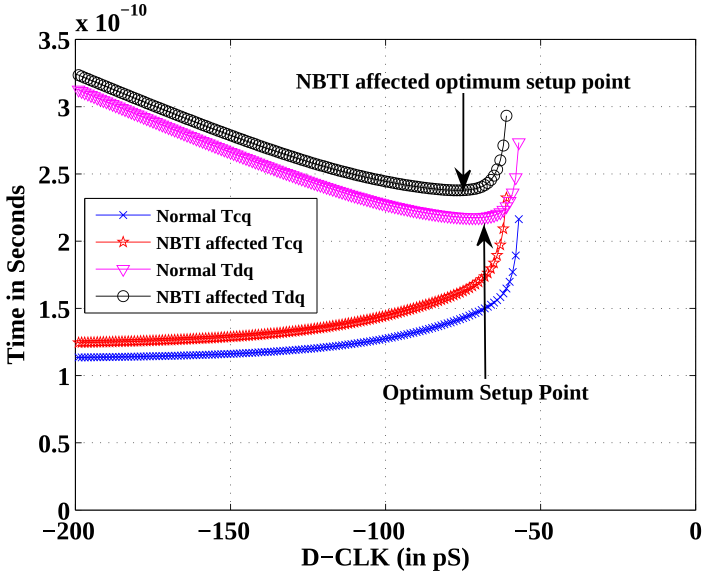


Figure 3.3. Timing characteristics of a TG-MSFF

3.3 Topologies

3.3.1 Topologies

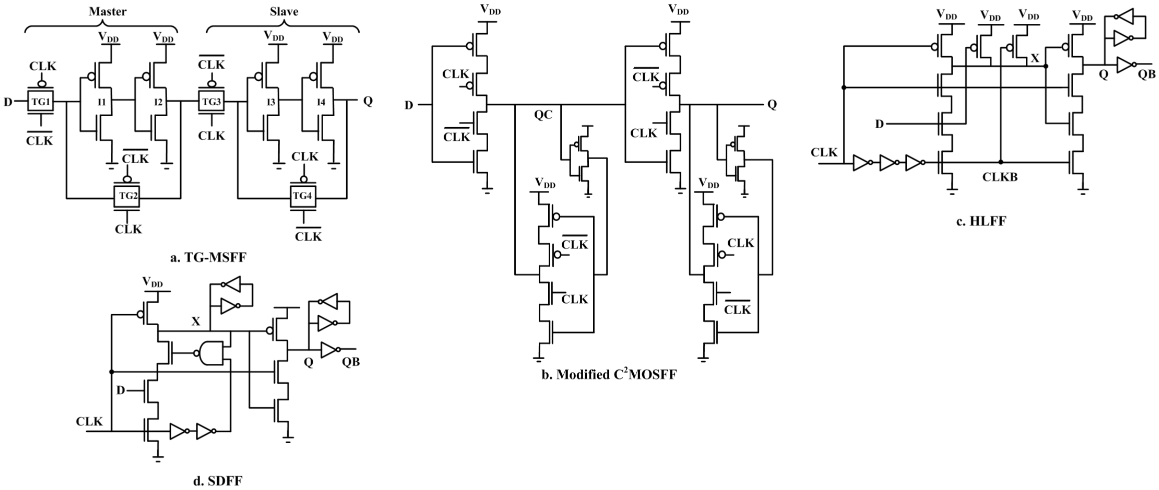
Flip-flops come in various forms depending on whether they are used in high performance or low power designs. In this section, we describe the designs of four such flip-flops used in commercial processors.

3.3.1.1 Master-Slave Latch Pairs

A flip-flop can be designed using a pair of latches when one is Transparent High (TH) and the another one is Transparent low (TL). Figure 3.4 (a) shows the setup of a Transmission Gate Master-Slave Flip-Flop (TG-MSFF). TG-MSFF exhibits a positive setup time and negative hold time. Figure 3.4 (b) shows the design of a modified clocked CMOS master-slave flip-flop (C^2MOSFF). This is a modified version of a standard dynamic C^2MOSFF for lower power consumption. Both of these are low power flip-flops. They have high T_{cq} and T_{dq} compared to pulse triggered latches explained later. Both of these latches are used in low power designs where speed penalty can be incurred.

3.3.1.2 Pulse Triggered Latches

Pulse triggered latches can be considered as master-slave latches with very small transparent window. The master latch serves as the pulse generator while the slave latch captures the input. Here, we have analyzed two such pulse triggered latches namely Hybrid Latch Flip-Flop (HLFF) and Semi-Dynamic Flip-Flop (SDFF). HLFF proposed by Partovi in [53] was used in AMD K6 and has extremely small delay with static pulse generation mechanism. Meanwhile, SDFF has a static and dynamic pulse method with a dynamic pulse generator feeding the static latch [54]. Pulse triggered latches are extremely fast flip-flops with very small delay. However, they consume high power due to internal switching power due to it's precharge and evaluate operation. HLFF and SDFF are used in high performance designs where delay is the first concern.



a. TG-MSFF

b. Modified C^2MOSFF

c. HLFF

d. SDFF

Figure 3.4. Flip-flop Designs

3.4 Experimental Results & Discussions

All the designs shown in figure 3.4 were simulated using Predictive Technology Models (PTM) [55] using HSPICE. We use the in-built Levenberg-Marquardt algorithm in HSPICE for optimal gate sizing for the minimum power-delay product (PDP). The algorithm uses steepest descent and Gauss-Newton method for optimization. Nominal rise and fall slopes for the clock was provided along with a standard load capacitance of FO4 inverters. All the transistors are sized at minimum length (L=1) while the widths are optimized. The duty cycle of the clock was 50% with nominal temperature at 80° Celsius. We assume that the clock signals

CLK and $\overline{CLK}$ do not suffer from clock skew. The base designs were simulated in 65nm technology and we use a supply voltage of V_{dd}=1.1V.

In this section, we present the set of results for each flip-flop. The threshold degradation of each PMOS transistor in the design is obtained using equations 3.1 and 3.2 depending on the amount of time they were stressed. The metrics are obtained with the setup and hold skews to be optimistic (long enough) and only one transition is allowed in a clock cycle. The results are presented for a dynamic circuit operation of 5 years.

The effect of NBTI is dependent on the amount of time the device is stressed. This stress time is dependent on the data input probability DP (DP=$1 - \alpha$) and is shown in the figure 3.1. In the following section, we also illustrate how the flip-flop timing characteristics vary with this parameter. The effect of transistor stacking on NBTI as explained in [56] should also be taken into account for proper characterization of the effects.

3.4.1 Master-Slave Latch Pairs

NBTI in a PMOS transistor increases the rise time of a $0 \rightarrow 1$ transition and lowers (negligible) the $1 \rightarrow 0$ transition when used in a pull-up network (PUN). TG-MSFF exhibit positive setup and negative hold times. Since, the hold time is negative for both the nominal and NBTI affected designs, we present only the other timing metrics here for the design. Figure 3.5 shows the input data probability has a considerable impact on CLK-Q T_{cq} and setup time T_{su} (Note that α is (1 - DP) where DP is the data input probability). The setup time of the TG-MSFF in figure 3.4 (a) is dependent on TG1 and the set of inverters I1, I2 of the master latch. The CLK-Q delay T_{cq} is dependent on TG3 and the set of inverters I3 and I4 of the slave latch. The absolute setup time T_{sua} in figure 3.5 also increases as DP increases. The increase in T_{cq} for $1 \rightarrow 0$ transitions (shown in the inner figure) shows marginal change and is insignificant compared to $0 \rightarrow 1$ transition and does not contribute to maximum T_{ff}.

Due to the dynamic nature of the transmission gates along with the clock input of 50%, they undergo less stress than a constantly stressed static gate. Accounting for the correct stress in transmission gates, the degradation of T_{ff} due to NBTI for various input data probabilities is calculated. Figure 3.6 shows the increase in T_{ff} of a NBTI affected flip-flop if the signal arrives at the setup time (case 1) decided for the unaged flip-flop for both the transitions. Such an increase in T_{ff} can result in violation of the condition for T in equation 3.3. The design is affected the most for the input data probability of DP=1 (input remains '1' all the time). We observe a decrease in T_{cq} for DP=0 (which means input is 0 all the time) since this does not affect the PMOS in I4 and the completely stressed PMOS in I3 helps in the transition of $1 \rightarrow 0$.

Figure 3.7 shows the analysis for deciding the value of T_{ff} in equation 3.3 after degradation due to NBTI for various input data probabilities in a TG-MSFF. The value of T_{ff} is calculated using both the methods (5% increase in nominal T_{cq} and minimal D-Q point hereafter denoted as (1) and (2) respectively in all the figures marked) for the degraded design and the worst case T_{ff} can be computed. After 5 years, We clearly see that Low to High (LH) T_{ff} dominates the other as observed similarly in figure 3.5. Thus, deciding the values of T_{ff} from this analysis will yield a fail proof design even after NBTI degradation. The points in Y axis on figure 3.7 indicate the values of T_{ff} of the unaged design.

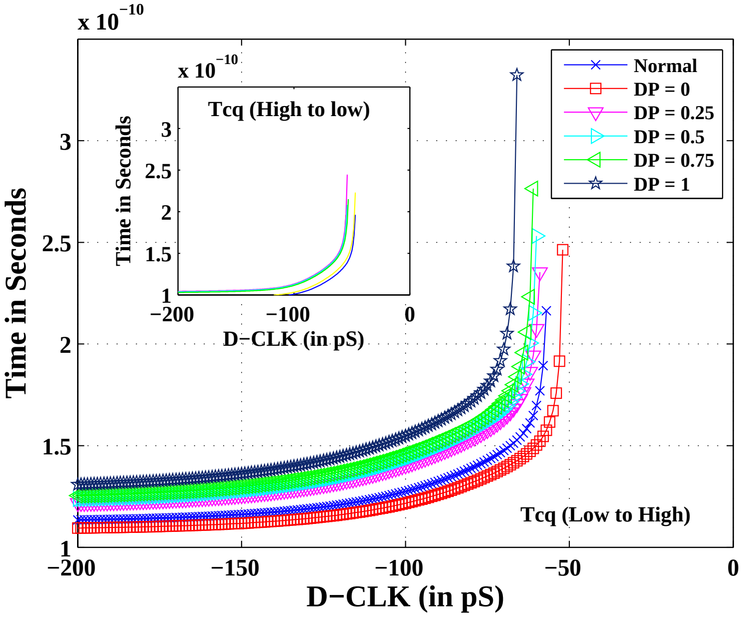


Figure 3.5. Impact of input data probability on the timing characteristics of TG-MSFF

Modified C^{2}MOSFF shown in figure 3.4 (b) is more complex to analyze for the effect of NBTI. The setup time T_{su} and CLK-Q T_{cq} of modified C^{2}MOSFF is governed by the master and slave latches respectively similar to the TG-MSFF. From figure 3.8, we observe that T_{cq} of $0 \rightarrow 1$ transitions increase as the circuit

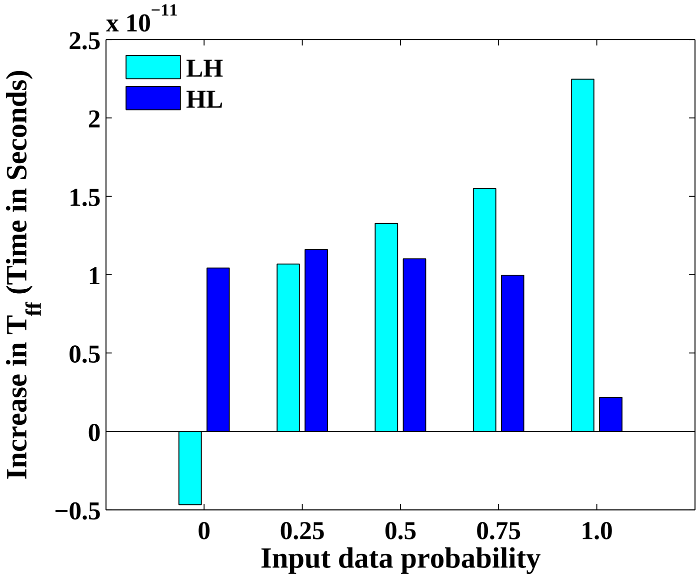


Figure 3.6. Increase in T_{ff} with input data probability for a NBTI affected TG-MSFF

undergoes aging. Figure 3.9 shows the increase in T_{ff} of a NBTI affected flip-flop if the signal arrives at the setup time (case 1) decided for the unaged flip-flop for both the transitions. This proves that the flip-flop is highly stressed if the input data remains at '1' most of the time. We observe that the T_{ff} which was dominant for 1 → 0 transitions (denoted as HL) change to 0 → 1 (denoted as LH) at the end of stress period. Thus, it is imperative to study the impact of input data probability to calculate the worst case degradation when the effect of NBTI is analyzed. Figure 3.10 shows the analysis for deciding the value of T_{ff} in equation 3.3 after degradation due to NBTI for various input data probabilities in a C^{2}MOSFF. The nominal T_{ff} of an unaged design is shown on the Y axis. The points in Y axis on figure 3.7 indicate the values of T_{ff} of the unaged design. The hold time is still negative for C^{2}MOSFF and is not presented here.

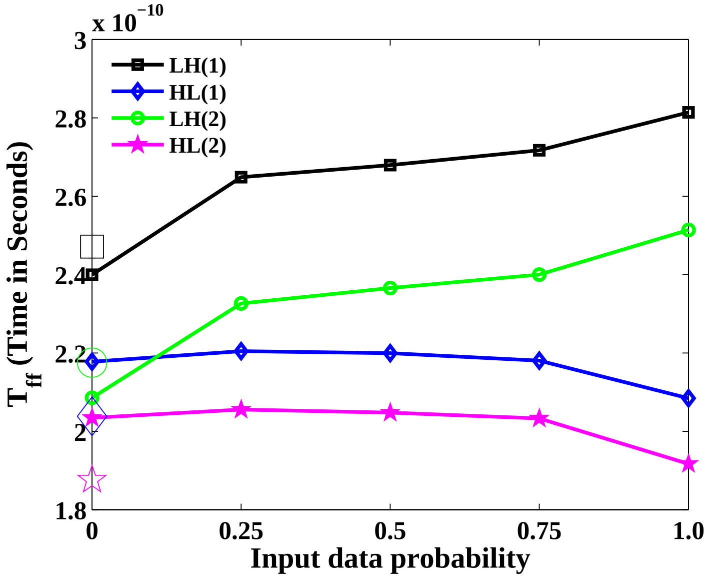


Figure 3.7. Variation of optimal T_{ff} with input data probability for NBTI affected TG-MSFF

3.4.2 Pulse Triggered Latches

Hybrid latch flip-flop (HLFF) which is a high performance storage element with very small delay. The data is latched during the transparency period created by the 1-1 overlap of CLK and CLKB shown in figure 3.4 (c). Odd number of inverters are inserted after the CLK signal to achieve CLKB. HLFF exhibits negative setup time and positive hold time. So, the data is allowed to arrive even after the clock edge and still be latched correctly. The hold time constraint is created by the falling edge of CLKB. Negative setup time helps in high speed circuit designs for slack borrowing, slack passing and absorption of clock skew. Positive hold time posts a negative impact on the circuit. Narrow transparency periods are advantageous because they reduce potential race through problems and increase immunity to

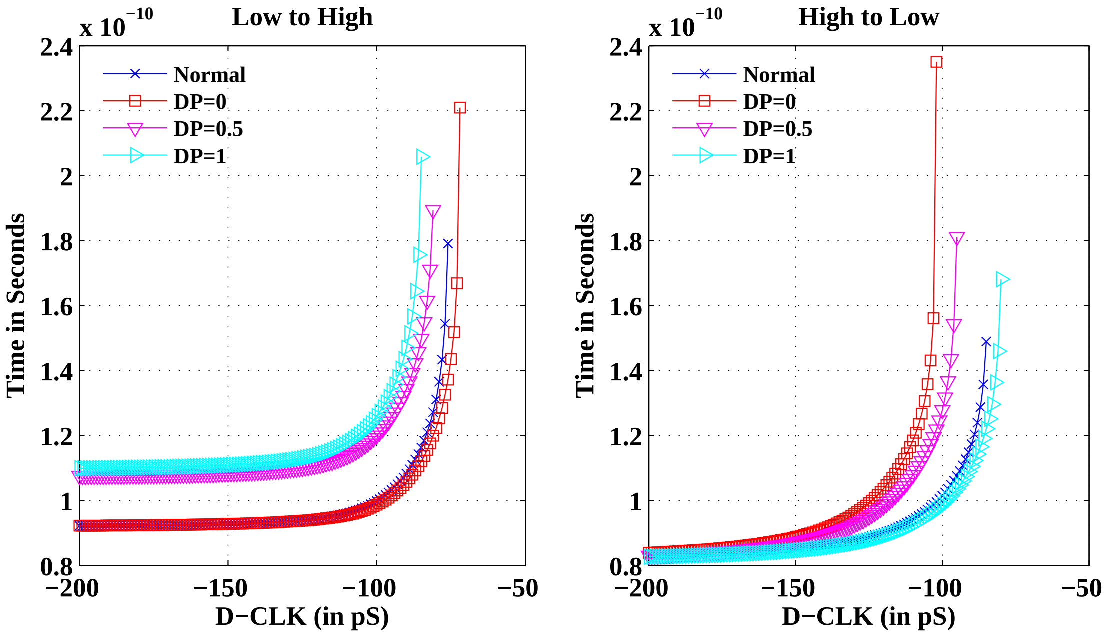


Figure 3.8. Impact of input data probability on the timing characteristics of C^{2}MOSFF

noise. However, they should be long enough for the flip-flop to latch on correctly and to utilize the slack allowed for the data.

Figure 3.11 shows the increase in T_{ff} of a NBTI affected circuit for the zero setup time. It is obvious that the change in HLFF is negligible while SDFF shows a minimal increase than HLFF. As shown in figure 3.12 and 3.13, the variation of T_{ff} with the input data even after 5 years is minimal compared to master-slave latches. This is due to the basic topology of the circuits. The $0 \rightarrow 1$ transitions in pulsed latches depend on the strength of the NMOS stack of the pulse trigger circuit and the final PMOS transistor. In the case of HLFF, the intermediate node X is precharged to '1' (except the negligible time of discharge) and put the PMOS transistor in recovery mode. During the evaluate phase, it discharges to '0' if the

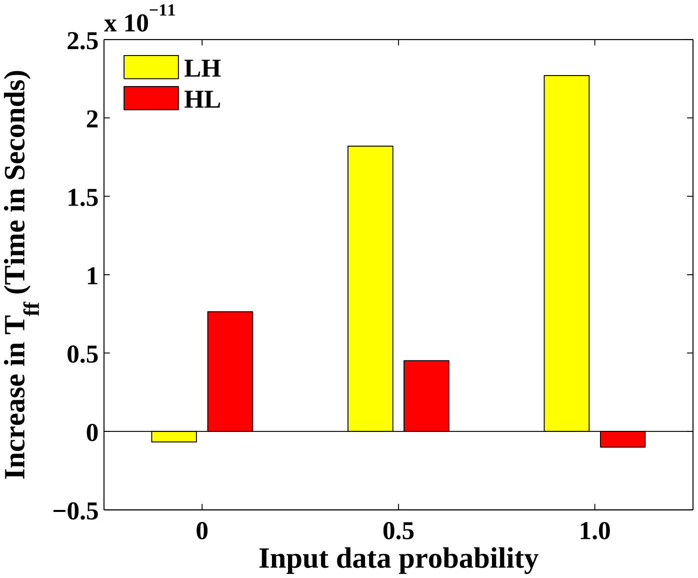


Figure 3.9. Increase in T_{ff} with input data probability for a NBTI affected C^{2}MOSFF

input data is '1'. This causes the T_{cq} of HLFF to be almost constant thereby not degrading the performance (infact, slightly faster as the pull up PMOS in the pulse generator stage gets weakened). In the case of SDFF, the node X discharges only if the input D='0' and remains at the stress mode only for half of the clock cycle (it becomes a '1' at the precharge portion of the clock cycle). Therefore, the T_{cq} variation is SDFF is also minimal. The variation in figure 3.13 reflects this change in the stress of the PMOS according to the data rate. However, this is also minimal compared to master-slave latches. The points in Y axis on figures 3.12 and 3.13 indicate the values of T_{ff} of the unaged design.

The transparency period play a critical role and should be long enough for the correct pulse to get latched while it imposes a longer hold time if it is large. NBTI causes an increase in the delay through the buffers and therefore increases the width

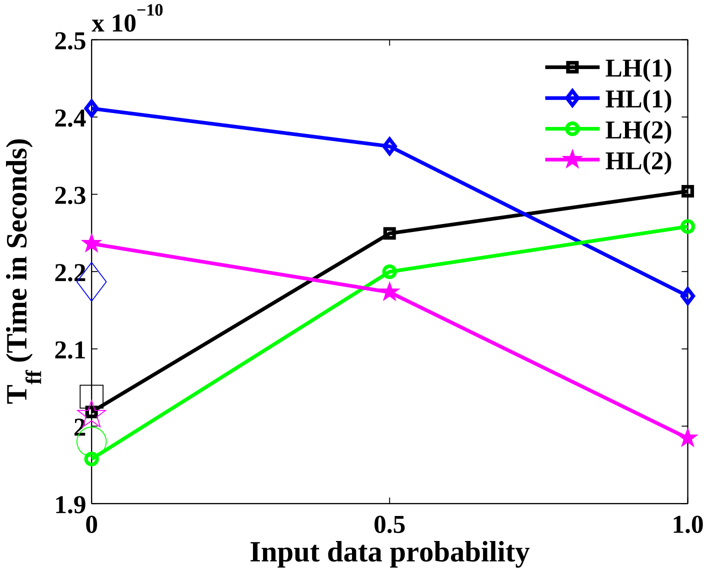


Figure 3.10. Variation of optimal T_{ff} with input data probability for a NBTI affected C^{2}MOSFF

of the transparency region. Table 3.1 shows the increase in the transparency widths of HLFF and SDFF over a period of 5 years with the duty cycle of clock being 50%. The increase in delay is due to the increase in the threshold of the second inverter in the buffer section of the pulse generator which results in a longer $0 \rightarrow 1$ transition while the first and third in HLFF undergo a $1 \rightarrow 0$ transition(the third inverter in SDFF is replaced by a NAND gate as shown in the figure 3.4(d)). The NAND gate also undergoes aging depending on the input data and plays a role in the conditional shut off time. The hold time for these pulsed latches are generally taken as the width of the transparency regions and as shown in table 3.1, it shows a minimal increase. This can be easily compensated by using buffer pads used for minimal combinational logic delay between flip-flop stages when there is no logic between them.

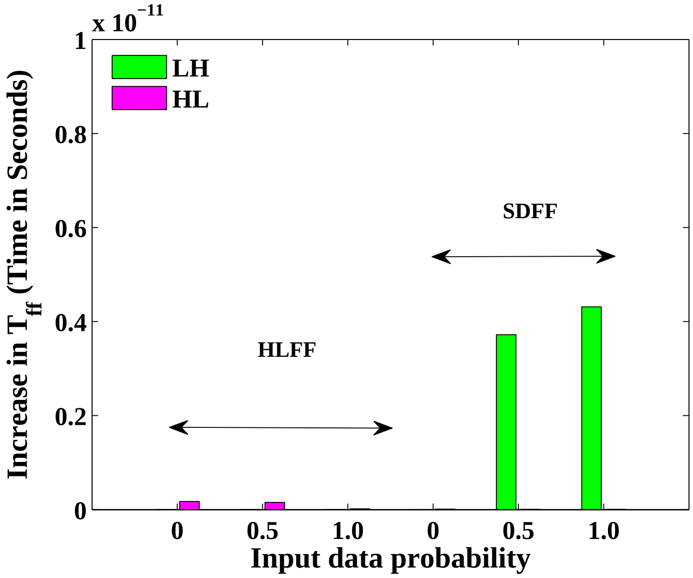


Figure 3.11. Increase in T_{ff} with input data probability for a NBTI affected HLFF and SDFF

Table 3.1. Transparency pulse-widths of pulse triggered latches after 5 years

Flip-Flop	Normal (in pS)	NBTI affected (in pS)
HLFF	68.1	72.7
SDFF	35.0	38.2

Figure 3.14 shows the nominal T_{cq} (average of both the transitions) of all the flip-flops. It is clear that the T_{cq} of TG-MSFF and C^{2}MOSFF are affected more while the HLFF undergoes the least degradation. SDFF on the other hand undergoes a much lesser degradation than the master-slave pairs, but more than HLFF due to the reasons explained in the earlier sections.

Figure 3.15 shows the impact of temperature on NBTI on flip-flops. The impact of NBTI is dependent on the operation temperature as the change in threshold

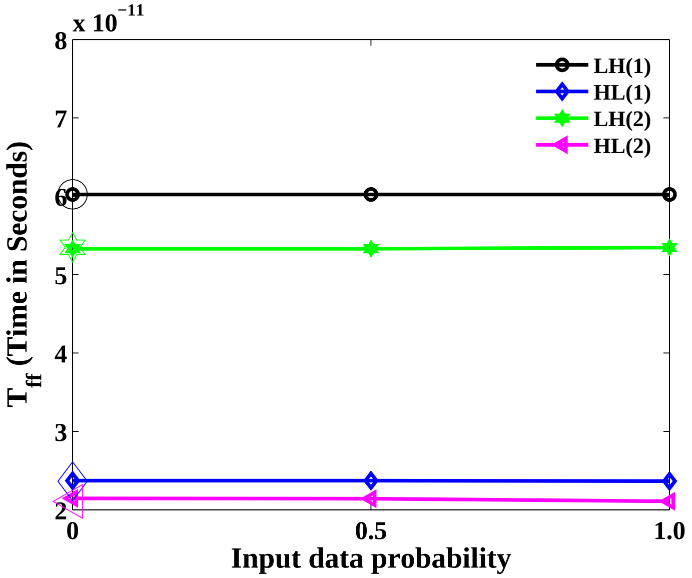


Figure 3.12. Variation of optimal T_{ff} with input data probability for a NBTI affected HLFF

(ΔV_{th}) is exponentially dependent on temperature in the constant K_v in equation 3.1 and increases as temperature increases. The percentage degradation values of T_{ff} shown in figure 3.15 for three different temperatures were normalized to their base values at the same temperature. Higher temperatures cause higher percentage increase in the delays with master slave latches showing more degradation than pulse triggered latches. The absolute increase in the $0 \rightarrow 1$ transition in SDFF is still extremely small though the percentage degradation is higher as they are normalized to the base values.

Figure 3.16 shows the percentage variation of T_{ff} for various technologies at 80° for the input data probability of 0.5(for realistic purposes) and at the setup time decided by case 1 for a unaged flip-flop. They values are normalized for

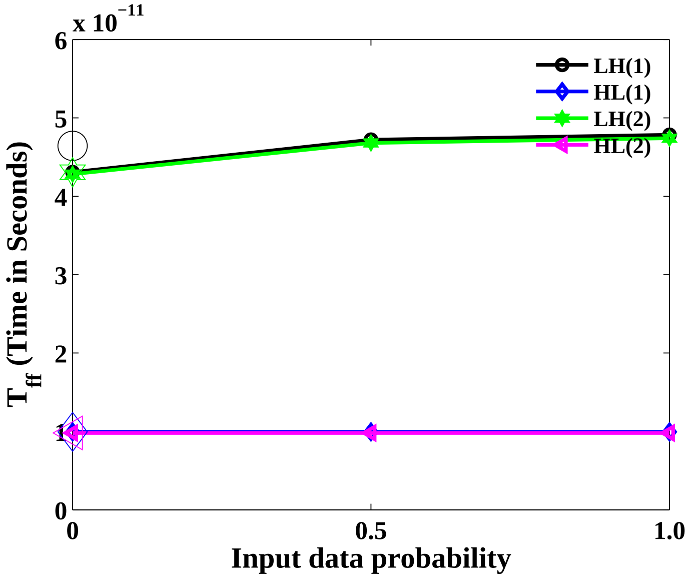


Figure 3.13. Variation of optimal T_{ff} with input data probability for a NBTI affected SDFF

the nominal unaged design of the technology. The change in threshold (ΔV_{th}) for lower technologies are much higher than 65 nm and the reduction in supply voltage augments the increase in the delays as shown in the figure. Even though, the percentage increase in delay for pulse triggered latches are high (SDFF 0-1), the absolute increase in delay from the nominal value is much smaller than the master slave latches.

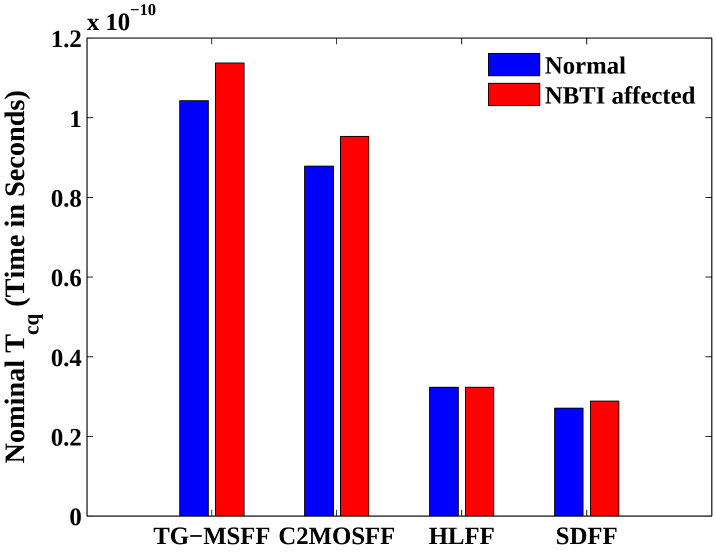


Figure 3.14. Comparison of Normal and NBTI affected nominal T_{cq} for various flip-flops

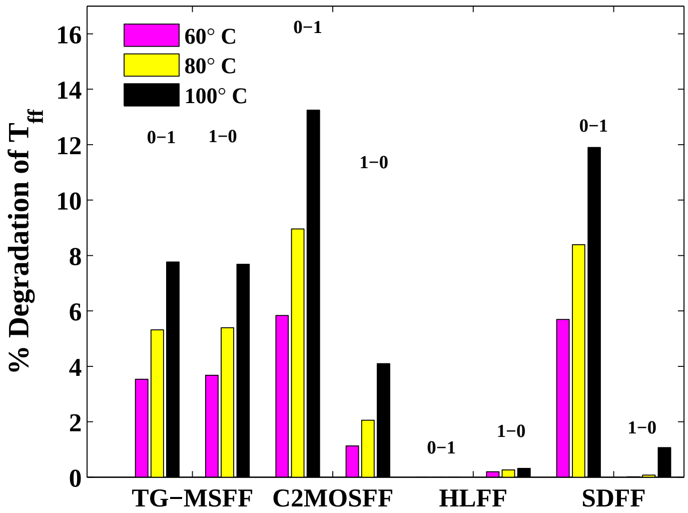


Figure 3.15. Effect of temperature on NBTI degraded flip-flops

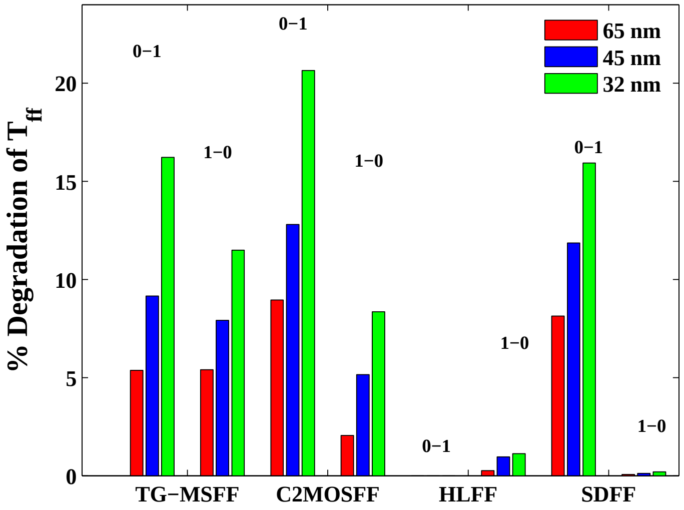


Figure 3.16. Effect of NBTI on different technology nodes

3.5 Introduction

FPGA (FPGAs) have been aggressive with their scaling trends primarily due to tremendous advantages provided by them in the form of low NRI costs and symmetric designs. However, the limitations due to the basic physical nature of devices have become quite evident with aggressive scaling of technology [?]. Consequently, apart from the well researched issues of power,performance and process variations,the biggest threat with minuscule feature sizes is their reliability concerns. Reliability issues such as Electro-migration(EM), Hot Carrier Effects(HCE), Time Dependent Dielectric Breakdown (TDDB) and Negative Bias Temperature Instability(NBTI) tend to pose serious problems as technology scales. In particular, there has been recent increasing interest on the impact of NBTI on PMOS transistors [2] [57]. We analyze the impact of NBTI on various components of a FPGA and provide solutions to subsidize the effect.

FPGAs typically use memory elements in the form of 6T SRAM cells to store the configuration bits encoding the hardware. Such memory elements storing the configuration bits, once configured for some designs usually reside in the same state for long periods of time. Although the read/write delays of such cells are of no consequence in FPGAs, their stability is of prime concern due to the criticality of such configuration cells. The continuous stressing of the PMOS device in a memory cell decreases the stability of the cells gradually. Such instability is quantified in terms of reduced Static Noise Margins (SNM) of the cell. Reduced stability increases the vulnerability of the SRAM to noise and transient errors like soft errors, which is a prime concern in such devices. Apart from the stability of the device, the increased threshold voltage can lead to increased delays of the individual circuits, thereby affecting the timing constraints of the design implemented in them. These critical problems have been analyzed comprehensively and addressed with some novel solutions in this work. Our contributions include the following (a) Analyzing the impact of NBTI on different components of FPGAs over time (b) Observing the performance and stability of the device and applications mapped on the devices (c) Solutions to counter such problems by effective interleaving of stress and recovery cycles.

3.6 NBTI Modeling

Till date, research on NBTI has been active in the fields of device and reliability physics. There have been different models worked upon in [62] [63]. The most commonly employed stress model is the Reaction Diffusion (R-D) model [26] with fine modifications for technologies. For this work, we use the stress and the recovery equation as shown in equations from [56] for estimating the change in threshold voltage ΔV_{th} after a period of time.

At Stress,

$$\Delta V_{th} = \sqrt{K_v^2.(t - t_0)^{0.5} + \Delta V_{th1}^2} + \delta_v \tag{3.5}$$

At Recovery,

$$\Delta V_{th} = (\Delta V_{th2} - \delta_v).\Big(1 - \sqrt{\eta(t - t_0)/t}\Big) \tag{3.6}$$

where

$$\begin{aligned} K_v = A.T_{ox}.\sqrt{C_{ox}(V_{gs} - V_{th})}.exp\Big(\frac{E_{ox}}{E_o}\Big). \\ \left(1 - \frac{V_{ds}}{\alpha(V_{gs} - V_{th})}\right).exp\Big(-\frac{E_a}{kT}\Big) \end{aligned} \tag{3.7}$$

where E_{ox}=$(V_{gs} - V_{th})/T_{ox}$ and k the Boltzmann constant. The value of the coefficients are E_o=2.0 MV/cm, Ea=0.12 eV, A=1.8 mV/nm/$C^{0.5}$, $\eta = 0.35$ and δ_v=5mV. δ_v is a constant added to include the impact of oxide traps and other charge residues. Note that during the continuous operation of the circuit however, the device is under both stress and recovery based on the gate inputs' static probability or the duty cycle. The change in threshold voltage for long term degradation after 'n' cycles of stress and recovery is obtained from equation 3.8. These equations are used in conjunction with the PTM 90, 65 and 45 nm technologies [20] and all the simulations were performed at 100 °C.

$$\begin{aligned} \Delta V_{th} = K_v.\beta^{0.25}.T^{0.25}. \\ \left(\frac{1 - (1 - \sqrt{\eta(1 - \beta)/n})^{2n}}{1 - (1 - \sqrt{\eta(1 - \beta)/n})^{2}}\right) + \delta_v \end{aligned} \tag{3.8}$$

where β is the duty cycle(ratio of time of stress to stress+recovery) and T is the clock period. Figure 3.17 shows the results obtained using the equation 3.5 for different technology nodes at 100 °C. It could be observed from the figure that the threshold voltage rise of the PMOS transistor is close to 10% for the period of 10^8 seconds ($\sim$3 yrs), which clearly indicates the severity of the problem.

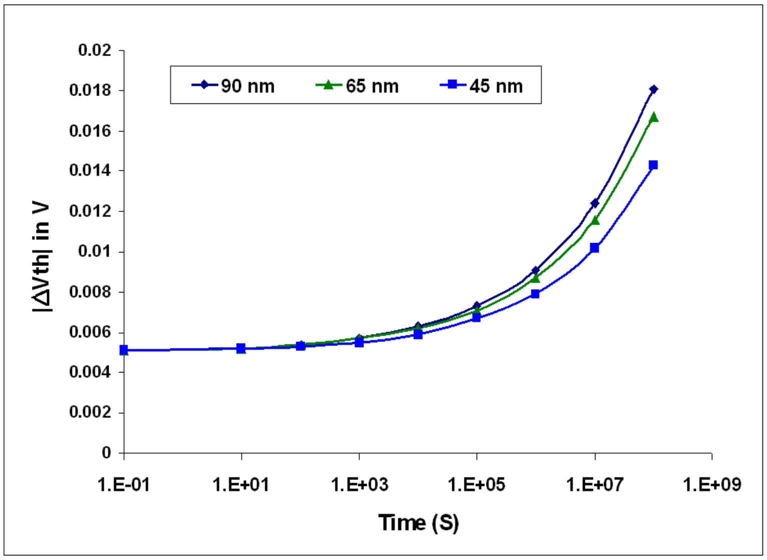


Figure 3.17. ΔV_{th} Variation of different technology nodes with age

Another important observation is the strong dependence of ΔV_{th} on the electric field, T_{ox} and temperature, which however are absent in the recovery equation. Such a bias significantly impacts the aging due to NBTI with technological fluctuations or process variations. Figure 3.17 shows the degradation in threshold of the PMOS (for V_g=V_{dd} as in table 3.2)after a period of $\sim$3yrs for three different technology nodes (Note that ΔV_{th} starts with 5mV due to δ_v). It is important to note that the change in threshold ΔV_{th} decreases as technology scales. This is due to the fact that the electric field across the gate oxide decreases for future technologies (since $(V_{gs} - V_{th})$ scales down faster compared to T_{ox} as shown in table 3.2) [20].

3.7 NBTI Analysis in FPGAs

FPGAs have many distinct features which require the analysis of each of its components separately with respect to the NBTI problem. The prime components under analysis in this paper are the configuration SRAMs, level restorers, buffers,flip flops and latches. The analysis is specifically targeted at current and future FPGA technologies, i.e, 65 nm and 45nm gate length devices. Each of the components affect the FPGAs in a different manner and is studied in detail in the following sections. The components used in our studies typically resemble the design used commonly by most FPGA vendors.

3.7.1 Configuration Bits

Most FPGAs store their configuration bits in 6T SRAM cells as shown in the figure 3.18.Such configuration SRAM cells are used to store both the logic in the form

Table 3.2. Technology Parameters

Technology Node(nm)	180	130	90	65	45	32
V_{dd} (V)	1.5	1.3	1.2	1.1	1.0	0.9
V_{th} (V)	0.22	0.2	0.2	0.2	0.2	0.2
T_{ox} (nm)	3.0	2.25	2.05	1.85	1.75	1.65

of Look Up Table(LUTs) and the routing information for controlling the routing switches. Since the configuration of the device stays the same once programmed, the SRAM cells hold a value for long periods of time up until the FPGA is re-configured. Such a scenario leads to stressing of one of the PMOS transistors in the SRAM cell without recovery. Since the configuration SRAM cells are not in the critical path their timing degradation does not impact the performance of the application but it does affect the overall stability of the SRAM cell. Using the equation 3.5, we obtain the V_{th} degradation of a PMOS device while the value stored by the SRAM cell does not change. The severity of the problem in configuration bits is however subsidized a little due to the usage of medium-oxide (one of the triple-oxide thickness) [64] gates to reduce leakage ower in FPGAs. We performed our experiments on such SRAM cells to obtain the degradation in the device's SNM, read and write delays.

The performance degradation of a 45 nm degraded SRAM was studied. It was observed that the read delay is almost unaffected while the write delay improves slightly and is in consonance with [34]. Figure 3.19 demonstrates a graph of SNM degradation of SRAM cells for three technology nodes. It also shows the degradation for three different oxide thicknesses (for triple-oxide thicknesses of Xilinx) for a 45nm technology node over a period of 10^8 seconds (∼3 yrs). Note that even in the thickest of gate oxides, the SNM degrades by 2% which is quite significant with respect to the read stability of the SRAM cell [34]. Increasing manufacturing

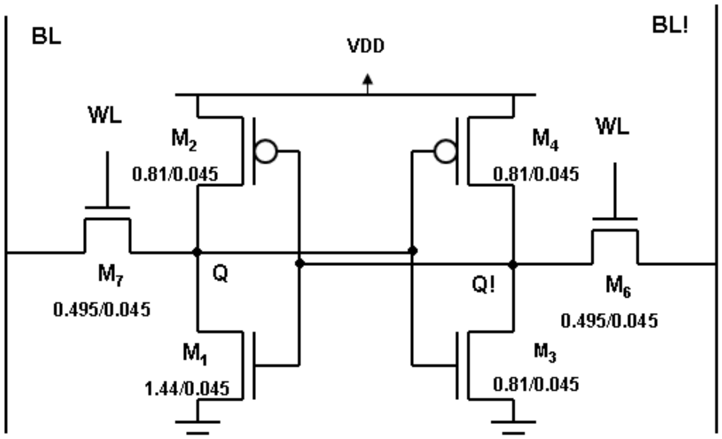


Figure 3.18. Nominal 6T SRAM Cell

uncertainty leading to variable oxide thicknesses may have a significant impact on the time to failure of different devices and impact the reliability yield of the chips.

Table 3.3. Read and Write Delays of a SRAM

Time in Years	% Increase in Read Delay	% Decrease in Write Delay
0	0	0
0.01	0.0014	0.1482
0.1	0.0299	1.5719
1	0.0477	2.264
5	0.0646	2.8343
10	0.0803	3.1366

Moreover, the degradation in the SNM values not only decreases the stability of the SRAM cell, but also increases the vulnerability of transient errors like soft errors, crosstalk etc. Therefore, it is imperative to analyze the soft error susceptibility of the affected SRAM cells. Critical charge Q_c is defined as the minimum charge that should be generated by the strike to cause an upset. Table 3.4 analyzes the impact of NBTI on the susceptibility of the SRAM cell to soft errors. It is observed that the critical charge (Q_c) decreases as the PMOS transistor gets degraded over time. Critical Charge (Q_c) for flipping a bit Q from 1→0 is lesser than 0→1 due to wider N^+ diffusion of the NMOS as shown in the figure 3.18.The Q_c for 0→1 is nearly unaffected since the strike occurs at Q! and the affected PMOS lies in the regenerative side in this case. We observe that Q_c for the bit flip from 1→0 decreases as the PMOS device gets affected due to NBTI. This arises due to the asymmetry in the affected cell. The affected PMOS transistor with its degraded current drive fails to bring back the node to 1 easily than the unaffected device. We also calculate the FIT rate (FIT is 1 failure in 10^9 hours of operation) of the degraded device using the model presented in [9]. A conventional SRAM's

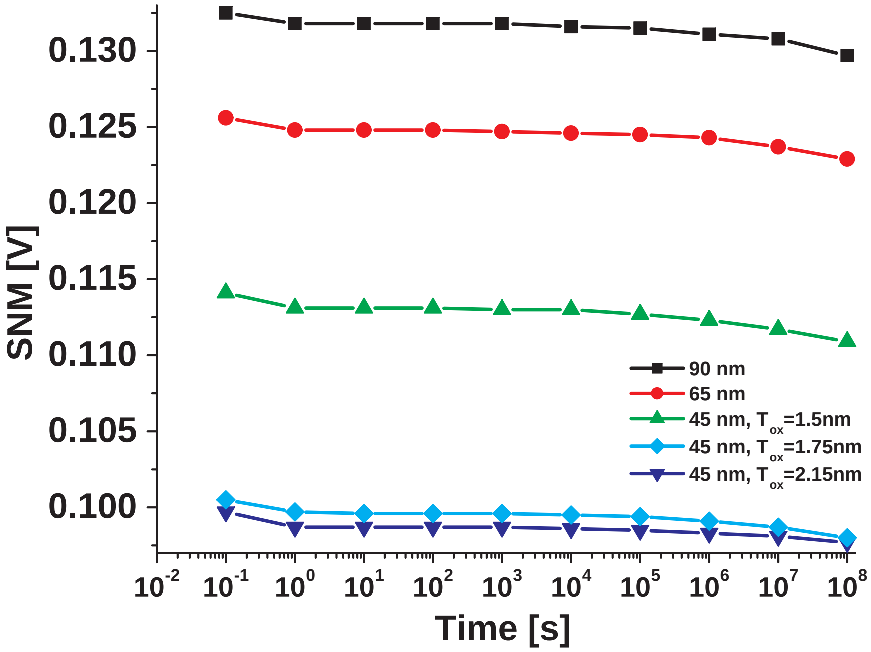


Figure 3.19. Variation of SNM with Age

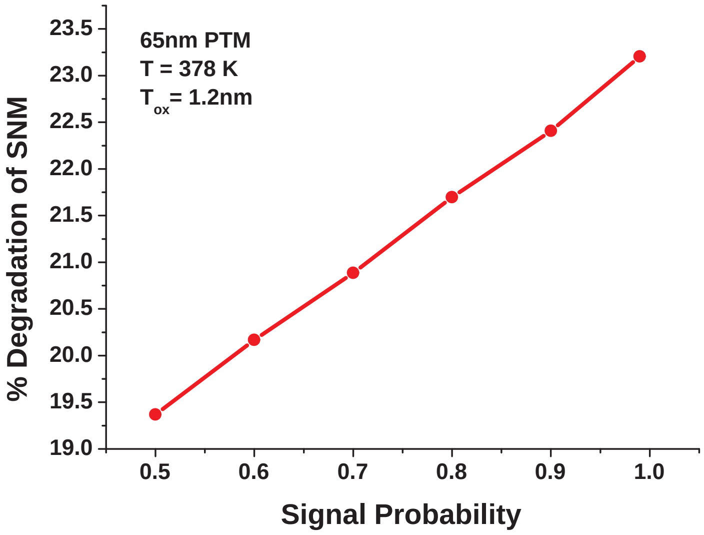


Figure 3.20. Impact of signal probability on SNM degradation

FIT rate is assumed to be 1000 FIT/MBit. Q_s which is the slope transformation factor is derived from [9] to be 4fC. The new FIT/MBit calculated of degraded SRAM is higher showing lesser resilience to errors.

Table 3.4. Critical Charge (Q_c) and FIT/MBit of Nominal and NBTI affected 45 nm SRAM Cell after 1 Year

	Bit	Flip	Nominal SRAM	Degraded SRAM
Q_{crit}	Q=1	0→1	11.244 fC	11.245 fC
	!Q=0	1→0	9.5366 fC	9.3896 fC
FIT/Mbit			1000	1037.5

3.7.2 Level Restorers and Buffers

Buffers and level restorers are commonly used in the FPGA interconnect circuitry. The interconnect circuitry comprise of multiplexers whose switches are stored in

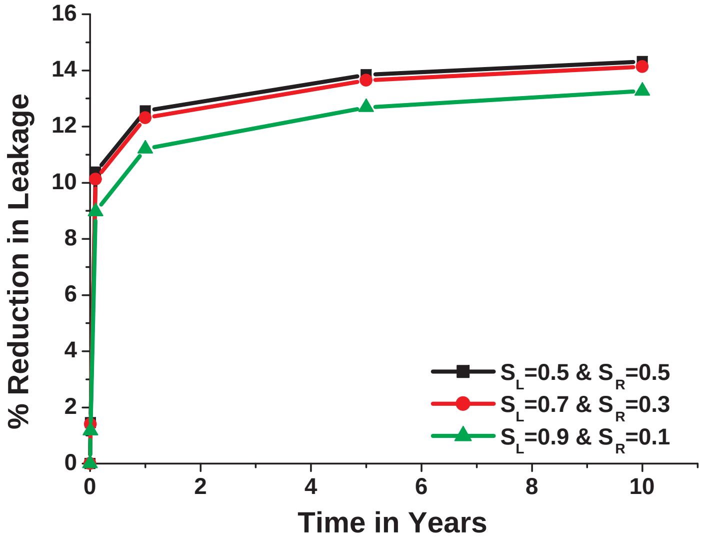


Figure 3.21. Leakage reduction percentage for various signal probabilities

the configuration SRAM cells. FPGA interconnect multiplexers typically comprise of pass transistors, which need a level restorer and buffers to retain the signal high as shown in figure 3.22. Such components are in the critical path of the applications and hence may impact the timing of the applications implemented on the FPGA. An analysis of the delay increase in a level restorer with the increase in the V_{th} of the PMOS gates, is presented in figure 3.23. However, estimating the delay increase in an application requires us to know the static probabilities at the different gates of level restorers and buffers in the device for a given application followed by obtaining the degradation of PMOS devices. Then the delay impact is calculated using the obtained ΔV_{th}s. The different ΔV_{th} for all the transistors are calculated for different duty cycles varying from 0.1 to 0.9 (Note that the duty cycle of of M_2 is β and of M_r and M_4 is (1-β) in figure 3.22) and the corresponding delays after one year of operation at the frequency of 100 MHz. The different duty cycles capture the effect of different static probabilities at the input of the level restorer and buffer.

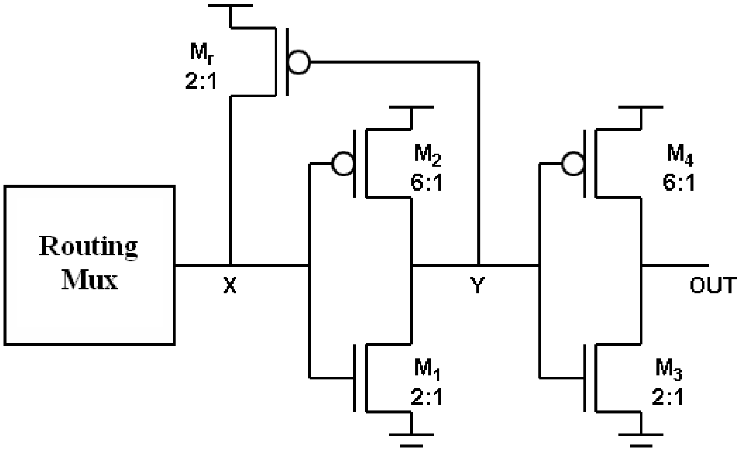


Figure 3.22. Level restorer and Buffer

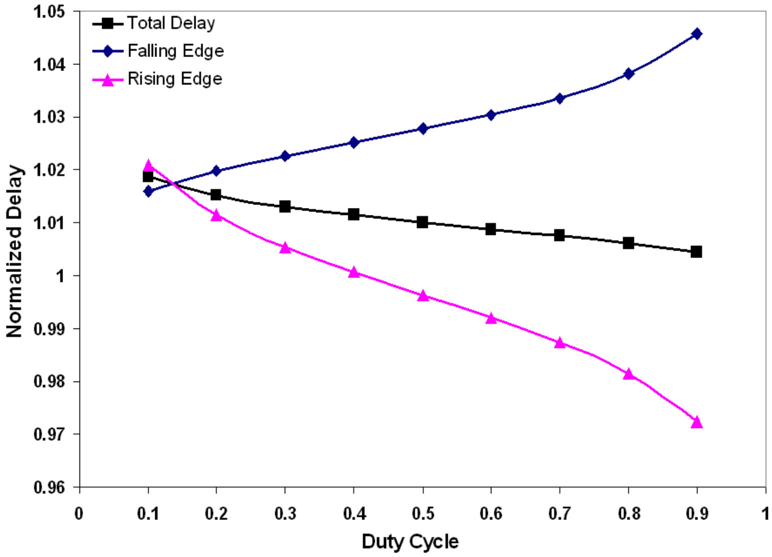


Figure 3.23. Impact of duty cycle on delay of level restorer and buffer

Figure 3.23 shows the delays for rising edge, falling edge and the total delay. We observe that the level restorer plays a critical role in determining the speed of the circuit. The rising edge and falling edge shows different trends due to different ΔV_{th}s for varying values of β and also the effect of level restorer in pulling up node X to '1' with increased thresholds. The total delay decreases as β increases from 0.1 to 0.9 (Note that this delay is still higher than the normal delay).

To analyze such an impact of NBTI on the performance of the circuits we use a set of 9 MCNC benchmarks. Based on the static probabilities of different routing elements and the level restorer degradation results presented in figure 3.23, we compute the degradation in the speed of the routers. The new delays are employed to obtain the performance degradation of the applications mapped on to the device over a period of time. We use the open source Versatile Place and Route (VPR) tool [65] to perform our experimentation. The critical path delay is computed after a period of 1 year for different benchmarks and plotted in figure 3.24 with the original delays annotated at the top. It is evident that all the benchmarks have an average increase in the critical path delay by 6.63%.

3.7.3 Flip Flops and Latches

Flip Flops and latches are predominantly used in I/O Blocks and Combinational Logic Blocks (CLBs). The register elements used in these blocks are edge-triggered D flip flops or level sensitive D latches. We implemented Transmission gate based D

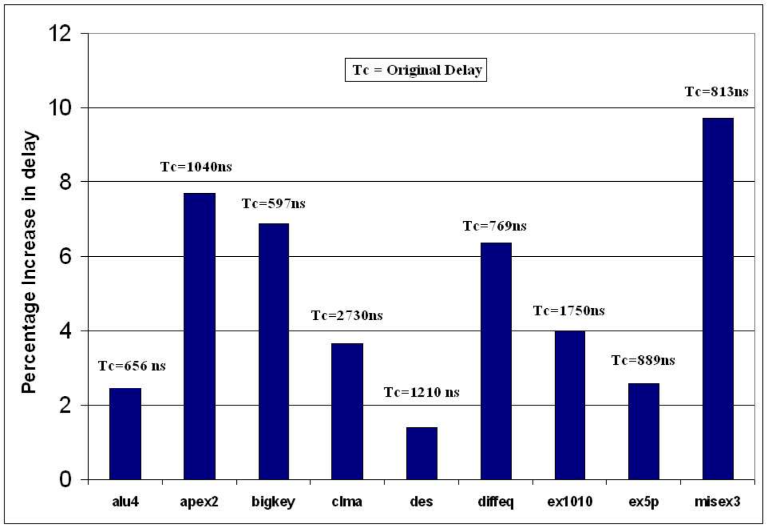


Figure 3.24. Performance degradation due to level restorers and buffers

flip-flop and D latches. The static probabilities of each internal node was calculated to be 0.5 and the corresponding ΔV_{th} for each of the PMOS at the end of one year is estimated using equation 3.8 at f=100MHz. The total delays after one, two and three years of operation for the flip-flops and latches are estimated using HSPICE and are shown in figure 3.25.

3.8 Mitigating NBTI effects

The impact of NBTI in a device can easily be inverted by flipping the gate inputs. In case of the configuration bits it requires us to flip the value stored in the configuration SRAM cells to restore the SNM. To recover from the increased delays of the level restorers and the buffers we need to invert the inputs driving the buffers. We present a methodology to achieve both, based on some existing as well as novel

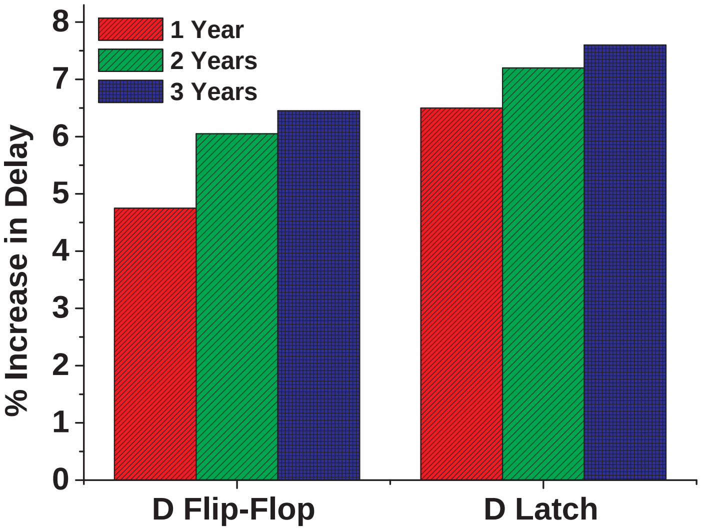


Figure 3.25. Change in delays of D flip-flops and D latches

schemes. The existing schemes are derived from the power aware schemes which work on Input Vector Control (IVC) based strategies. Note that not all the configuration bits can be inverted all together to relax the FPGAs, since such a reversal may lead to damaging of devices. One such example of shorting the circuit by flipping the routing multiplexer bits is shown in figure 3.26. It is therefore evident that the configuration bits governing different circuitries of the FPGAs require different schemes. In this work we demonstrate how an existing flipping policy of LUTs may increase the age of the device and also present a methodology to flip the configuration of the routers.

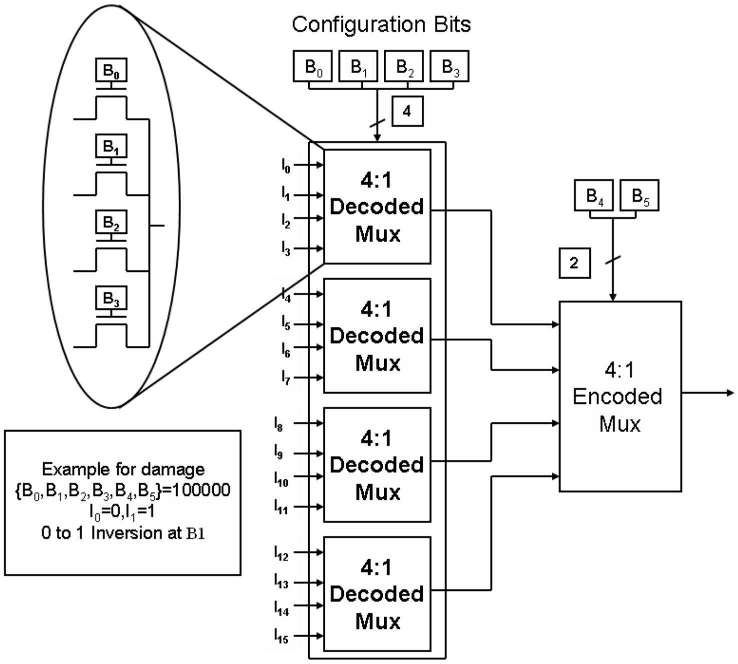


Figure 3.26. Routing mux architecture

3.8.1 Flipping configuration bits

The flipping of configuration bits may be performed by loading a new bitstream which has to be provided and stored in any external memory in the FPGAs. This bitstream which may also be termed as a *Relaxing Bitstream* (RBIT), may be loaded over a period of time onto the device to relax the various devices while continuing to perform the required application implementation. Such RBIT(s) may be generated in a orderly manner. The configuration bits mainly store the LUT logic and the routing information. A technique similar to the SER aware technique presented in [66] may be employed to flip the bits used for configuring the LUTs. Figure 3.28 demonstrates the flipping operation which does not impact the LUT logic at all. All the LUT bits are flipped and shuffled appropriately to maintain the functionality of the application. However, performing both the shuffling and flipping together prevents inversion of all the configuration bits, since some of the bits retain their original values.The flipping algorithm is implemented by directly operating on the configuration bits of the FPGA using the opensource Java APIs provided by JBits(ver 3.0) [67] FPGAs.

The flipping of the configuration bits storing the routing information requires us to delve into the different types of routers in FPGAs. The routing multiplexers in FPGAs may be classified into four prime types as shown in table 3.5. Such a classification is important with respect to the strategies used for flipping the configuration SRAM cells of the multiplexers. Assuming a multi level multiplexer design as shown earlier in figure 3.26 we present different strategies to solve the bit inversion problem. Note that if any two inputs of opposite polarities are driving the multiplexer inputs, inverting the bits may turn on a 1-0 path and lead to shorting of the device as shown in figure 3.26. We therefore tackle such a shorting problem carefully for different multiplexers.

Table 3.5. Characterizing FPGA routing multiplexers

Router Type	Output State	Inputs State
Dead	Unused	All Undriven
Inactive	Unused	Some Driven
Active	Used	Some Driven
Fully-Used	Used	All Driven

(a) Dead Muxes: Such multiplexers comprise of nearly 60% of the total FPGA multiplexers even in the most used case. Since,all the inputs are undriven and the output is also unused, all the configuration bits may be conveniently inverted. The inputs and the outputs of the multiplexer may also be controlled appropriately to prevent the shorting, using the input vector control based strategies presented in [68].

(b) Inactive Muxes: These multiplexers have some of the inputs driven, but the output is unused. Consequently, they face the problem of shorting when the configuration bits are inverted. As demonstrated in the algorithm shown in figure 3.27, the bits of such routers may be flipped one at a time in a round robin manner.It is done for a time frame controlled using the variable i in the algorithm determined by the user. Note that this strategy requires either a processor support which is common in modern FPGAs or may be achieved by storing multiple frames of such configurations that may be loaded dynamically onto the FPGA.

(c) Active and Used multiplexers: Such multiplexers may not be inverted at all since their outputs impact the functionality. These routers however as demonstrated in the algorithm may be selectively rested by selecting alternate paths. Such a alternate path may be obtained at a bitstream level by using bitstream modulating tool JRoute [69]. Similar strategies for aging have been demonstrated for Electromigration aware design in [60].

```
i=0
x=0
Max_X_Coord = Maximum X coordinate of the device
MAX_ROUTE_INPUTS = Maximum inputs to any router
Invert_Route_Conf(Bitstream)
{
 Obtain the input and output information of the routers.
 for each( R in Routers )
   INPUTS = total inputs to the router R
   if(R's output is undriven)
    if(All inputs are undriven))
     Flip all the inputs/outputs/conf-SRAM-switches
      to same value
    if(Some or all inputs are driven))
      Drive $(i\ mod\ INPUTS)^{th}$ config bit to 1
   else if(R's output is driven)
    if(R's location's X == x)
      Reroute the connection using JRoute
     if(i < MAX_ROUTE_INPUTS) i=i+1 else i=0
      if(x < Max_X_Coord_FPGA) x=x+1 else x=0
}
```

Figure 3.27. Algorithm for flipping the configuration bits of the routers in an orderly manner

3.8.2 Relaxing the Level Restorers and Buffers

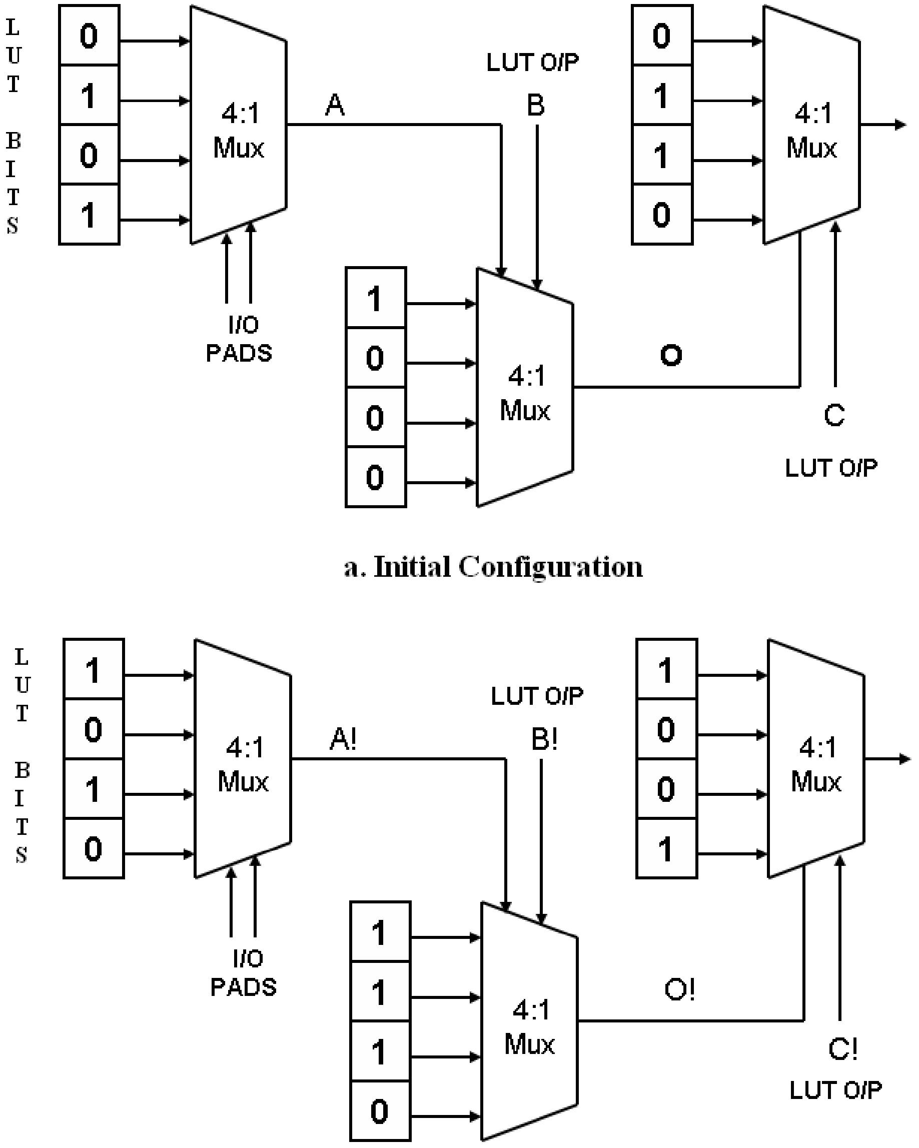
The inputs to level restorers and buffers come from the routing muxes as shown in the figure 3.22. The migration of the routing to alternate routers has a direct impact on the resting of the level restorers. Such re-routing helps the overall reliability of the used routers significantly, not only due to NBTI based degradation but also other aging phenomena. The resting of LRs and buffers also helps gaining back the performance significantly.

3.8.3 Experimental Results

The experimental results obtained from the proposed algorithms are explained in this section. From figure 3.19, we observed that the SNM degraded by 2% at the end of 2 years for a 45nm medium oxide SRAM cell.So, we chose 2 years as the total time frame for observation. Bit flipping was performed after 1 year on 10 Xilinx reference designs implemented on a Virtex-II device and we present the average SNM regained and FIT improvement of the devices. At an average, it was observed that 75.3% of the LUT Bits were flipped from its original configuration at the end of 1 year. The SNM at the start time of operation was 0.1005 V and the SNM of all benchmarks without cell flipping at the end of 2 years was 0.09828 V. The regained SNM results for the benchmarks (with bits flipped at the end of first year) at the end of second year are shown in table 3.6. Also, the FIT rate of the inverted cell also improved from 1038.5 FIT/MBit (for 2 years)to 1012.3 FIT/MBit (Bit flipped cell with maximum stress period of 1 year due to flipping) with higher critical charge. An estimated 2.5% decrease in FIT for cells inverted in X4VFX40 device was obtained for cell flipped designs with higher critical charge.

Table 3.6. SNM Improvement for Benchmark Designs at the end of 2 years

Benchmark	Average SNM after Bit Flip(V)	% of SNM regained
xapp248	0.09913	38.71
xapp288	0.09975	66.46
xapp289	0.09957	58.34
xapp298	0.09936	48.88
xapp299	0.09953	56.39
xapp610	0.09917	40.47
xapp615	0.09958	58.58
xapp621	0.09959	59.34
xapp625	0.09927	44.94
xapp645	0.09962	60.43
Average		53.26



a. Initial Configuration

b. After shuffling and bit flipping

Figure 3.28. Bit inversion to mitigate NBTI

Chapter 4

Interplay

4.1 Introduction

Sub-90nm circuits currently face an unmanageable problem of unpredictability in process parameters of their individual devices [45]. Such uncertainties not only affect the circuit power and performance [70], but also their resilience to transient faults like soft errors. Since the soft error resilience of any device has a strong dependence on the process parameters like device length, threshold voltage etc,the effect of process variability on SER of the circuits cannot be neglected. One of the main causes of such variations in process parameters is manufacturing parameter fluctuations due to increasingly challenging fabrication requirements [71]. Such effects are of static nature which may be characterized and detected immediately after manufacturing to some extent. Yet another reason for changes in device behavior is the dynamic and aging related variations due to runtime phenomenon like the power supply noise, temperature imbalance and device degradation. Each of these phenomenon impact the device characteristics and thereby the resilience of the devices to soft errors.

Manufacturing parameter fluctuations are one of the biggest problems faced by current circuit designers. The aggravation of such defects is attributed to technology scaling which poses very difficult fabrication challenges. The expected device parameters vary across different dies and within a die itself, namely inter-die and intra-die variations respectively[70]. Such variations may significantly affect power and delay characteristics of the circuit and thereby impose challenges in meeting the necessary budgets. Apart from being a reliability concern by themselves, such variations introduce newer reliability concerns in the form of modulating the susceptibility and resilience of the circuits to transient errors like radiation induced soft errors. Due to reduced feature sizes, any slight change in the process parameters contributes to a significant percentage of variations and thereby their impact is quite significant. Consequently, dynamically natured variations due to power supply variations and increased temperature across chips have also been one of the important causes for variations [72][73]. Also, variations due to device degradation because of Hot Carrier Effects (HCE) and Negative Bias Temperature Instability (NBTI), which are attributed to the circuit usage over a period of time, may lead to run time degradation and uncertainties in the circuits [74][56]. Such changes although may contribute to more of permanent failures, the degradation may once again affect the transient error vulnerability of the circuit since they change the operating conditions and the device parameters of the circuits, both of which affect the SER of a circuit significantly.

Soft errors in combinational circuits are becoming as important as those in unprotected memory circuits as technology scales due to reduced voltage and nodal capacitance, increased speed and decreased pipeline depths [10]. Modeling SER in combinational circuits has been a challenge due to the presence of the inherent masking mechanisms. Various methodologies have been proposed to model these logical, electrical and latch window masking effects in combinational circuits [10][14]. In this work, we used HSPICE to obtain accurate SER estimates due to variations in our custom benchmark circuits for accuracy and flexibility that HSPICE provides. We also performed SER analyses on larger ISCAS benchmarks using the Soft Error Analysis Toolset - Logic Analyzer (SEAT-LA) [14]. SEAT-LA uses an approach that is applied to designs that use cell libraries characterized for soft error analysis and utilizes analytical equations to model the propagation of a voltage pulse to the input of a flip-flop.

Although there have been many methodologies that propose to modeling and optimizations for soft errors [10][14], it is important to address soft errors in the presence of other reliability issues as well. This is first work that has looked at effect of other reliability issues on SER.

4.2 Modeling Variations

We have considered three different categories of variations, namely, static, dynamic and aging related variations. The methodologies used to model these types of variations are discussed in detail in this section.

4.2.1 Static Variations

Static variations are primarily due to manufacturing uncertainties, like variations in channel length, channel width, thickness of gate oxide and threshold voltage. Inter-die variations change the value of the parameters in all the transistors in a die in the same direction. These variations are caused by processing temperatures, equipment quality, wafer polishing and placement. Examples are channel length, channel width and variations between individual metal layers used for routing [71]. These variations mainly result in differences in power and delay. On the contrary, intra-die variations can be either systematic or random. These variations arise due to misalignment of wafer, Random Dopant Fluctuations (RDF) and uneven planarization steps. Transistor parameter shifts resulting from systematic variations are correlated and are dependent on the neighboring transistor parameters. Random variations shift the transistor parameters independent of the locality. Particularly, RDF leads to non-uniform distribution in transistor threshold voltages (V_{th}) in the circuit and is a key example to random variations. These static variations can be translated into change in the effective threshold voltage (V_{th}) [75].

Hence the effect of change of V_{th} due to process variations on the SER is studied in this work.

4.2.2 Dynamic Variations in Power Supply and Temperature

The dynamic variations considered by us in this work are the variations due to temperature changes in circuits and power supply variations. In this section we briefly discuss the impact of such variations on circuits.

4.2.2.1 Power supply variations

The variations in power supply have been one of the most important challenges as technology scales. This is because of the decreased supply voltage which results in a much larger ratio of the peak noise voltage to the ideal supply voltage [72]. This power supply noise is primarily a voltage drop in power distribution networks resulting in different voltages at different parts of the same chip. These variations are mainly due to resistive and inductive voltage drops across power supply networks. A power supply noise analysis methodology for circuits and microprocessors has been discussed in [76] . A minimum power supply fluctuation of 10% is acceptable [72]. Thus, in our work we have varied the power supply for the benchmarks by 10% and presented the corresponding variation of SER in section 4.4.

4.2.2.2 Variations in temperature

Heat generation in chips has increased rapidly with recent scaling trends and increased transistor density. This has led to non-uniform substrate temperature profile, affecting both interconnect and transistor delays. In the case of interconnects, the rise in temperature increases the resistivity of metals thus resulting in increased delays. In devices, temperature affects both the mobility and the threshold voltage. Rise in substrate temperature reduces the mobility of electrons/holes in MOSFETs because of increased scattering at higher temperature [73]. The threshold voltage also decreases with increase in temperature because of the change in fermi-potential ($\emptyset_f$) [73]. These two effects determine the trends in delay in logic circuits and thus, it affects the electrical masking capability of the logic circuits. The change in transistor delays also affects flip-flop characteristics like the set-up and hold times. This in turn results in a change in latch window masking capability of logic circuits. Thus, it is important to analyze these effects in detail and so far no work has contributed to such an analysis.

4.2.3 Variations Due to Aging

The aging related variations considered by us in this work are the variations due to device degradation over a period of time because of HCE, NBTI and power supply variations. To analyze the SER due to such variations, we first built tools using existing models, which are described in this section.

4.2.3.1 NMOS degradation due to HCE

The variations in threshold voltages of different devices over a period of time are considered for dynamic variations in the process parameters. Such change in process parameters are primarily due to factors like temperature and activity of the device, which impact the basic I-V characteristics of the devices.

One such phenomenon which leads to the degradation of the device due to the aforementioned factors is Hot Carrier Effect (HCE). Hot carrier effect is the phenomenon of trapping of high energy charge carriers at the gate oxide or creating new traps due to impact ionization effect. This trapping of charges increases the transistor threshold and thereby affecting the power and performance of the device. Such variations are more predominant in the NMOS transistors compared to PMOS transistors primarily due to negligible degradation rate of the saturation current of PMOS when compared to NMOS [77]. In this work we developed a tool using analytical models presented in [78] for individual devices to observe the aging impact on devices. Equations 4.1, 4.2 and 4.3 are the prime equations governing the degree of degradation of a device.

$$\frac{dN_{it}}{dt}[1 + AN_{it}] = KI_{bb} \tag{4.1}$$

$$I_{bb} = \frac{C_1}{W}.I_{DS}.exp(-\frac{B_i}{E_m}) \tag{4.2}$$

$$E_m = \frac{V_{ds} - V_{dsat}}{l_d} \tag{4.3}$$

where, N_{it} is the number of trapped charges per cm^2, constants, A= 5 x $10^{-9}cm^2$, K=5 x 10^{15}, C=2, $B_i = 4.41$ X 10^6 V/cm is the ionization coefficient, E_m is related to peak electric field along the channel and is given by equation 4.3. I_{DS} is the drain-source which flows through the device during a transition as demonstrated in figure 4.2. We use these equations along the analytical model presented in [78] to estimate the threshold variation of a single NMOS device under constant current conditions. The degradation of V_{th} under such constant stressing conditions is demonstrated in figure 4.1 for a 70nm NMOS transistor.

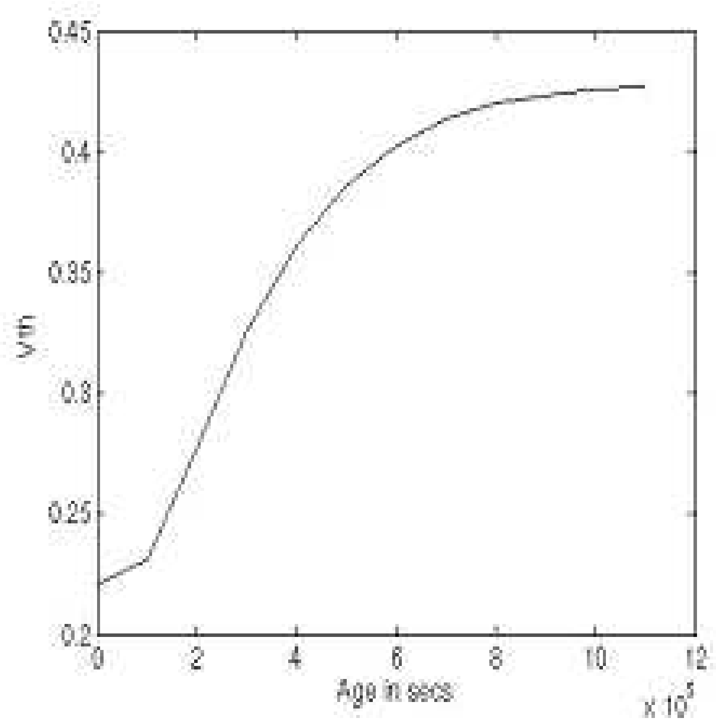


Figure 4.1. Variation of V_{th} of NMOS with aging

Note that this degradation is under the assumption of constant stressing of the NMOS device, which is essentially continuous current flowing through it for the given period of time. In circuits, however, current flow through the device is only during switching and over a short period of time. Hence the actual age of the circuits can be related to the stressed age using the equation 4.4.

$$\Delta S = \frac{T * P}{CLK} \left(\frac{\int_{t1}^{t2} I_{sub}\, dt}{I_{sub\_dc}} \right) \tag{4.4}$$

where, ΔS is the stressed age, T is the actual age, P is the switching probability of the gate of the NMOS transistor under consideration, I_{sub} is the substrate current, CLK is the clock frequency and I_{sub_dc} is the total constant current under stressed conditions that may flow during time T. The equation primarily exploits the number of transitions of the gate over a period of time and the current flowing through the device based upon the exact current estimates obtained from HSPICE. MATLAB model is integrated with the circuit simulation tool HSPICE for precise estimation of the individual device threshold changes over a period of time. We obtain the actual ages of the each of the NMOS transistors in a circuit during its operation based on the actual current flowing through the devices during any transition. These numbers are used to obtain the actual ages of the devices and thereby the new threshold of the devices at the end of a time window of observation, which is one day in our case. The thresholds are then used to obtain the new current estimates and the ages at the end of next day. Figure 4.3 demonstrates the flow of the tool developed by us which iteratively estimates the degradation of the devices given a SPICE model of any circuit. Note that unlike the random behavior of static process variations, these dynamic variations considered by us are more deterministic due to their strong dependence on the activity of the transistors.

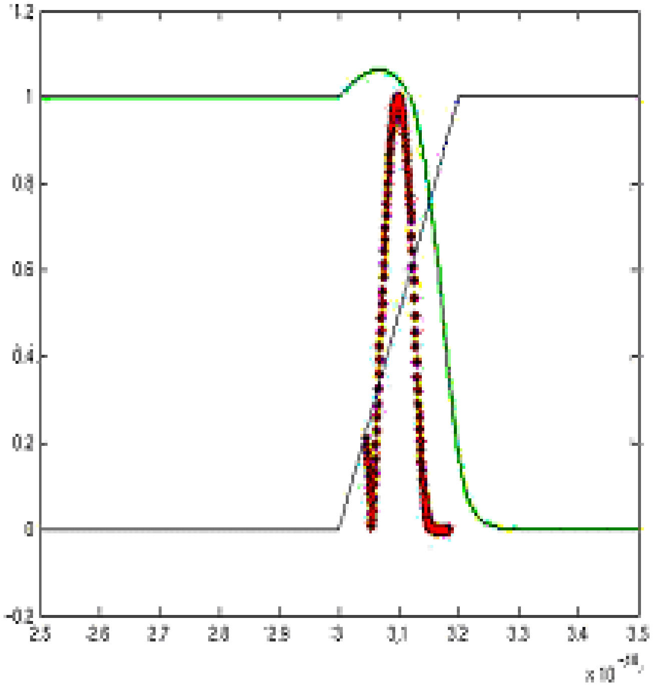


Figure 4.2. Device current during a transition

4.2.3.2 PMOS degradation due to NBTI

Another phenomenon that leads to slow degradation of especially PMOS devices is NBTI. NBTI in PMOS occurs under negative gate voltage ($V_{gs} = -V_{dd}$) and results in increased threshold voltage with time. The main reason for the NBTI effect is found to be the presence of increased number of positive interface traps

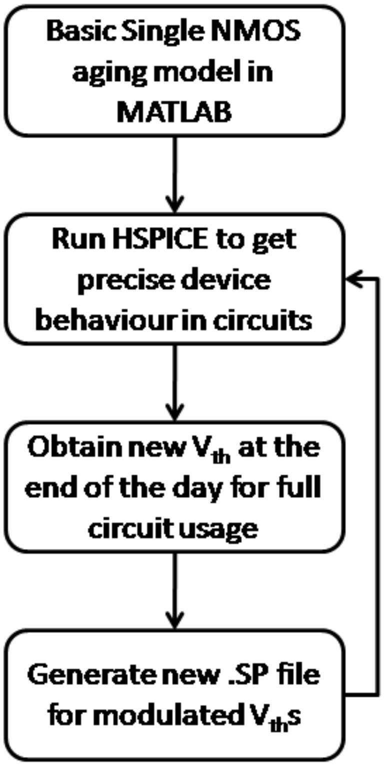

Basic Single NMOS aging model in MATLAB
Run HSPICE to get precise device behaviour in circuits
Obtain new V_{th} at the end of the day for full circuit usage
Generate new .SP file for modulated V_{th}s

Figure 4.3. Tool flow for estimating degradation

caused by the displacement of Si-H bonds, which are induced by positive holes from the channel. NMOS transistor has a negligible level of holes in their channel and hence does not suffer from NBTI degradation [56].

There are number of works recently which have attempted to model NBTI [26]. In this work, we follow the approach used in [56] as described in the earlier sections.

Using those equations, the change in threshold voltages are estimated taking into consideration the stress and recovery time based on the static probability of the gate voltage for each node in a circuit. With this information, the SER for any given circuit can be estimated with degraded PMOS in these circuits.

4.3 SER Estimation Tools

In this section a brief background on the experimental set-up and methodology used for our SER analysis is presented. In our experiments we analyze soft error rates in small custom benchmark circuits that represent combinational logic in general. We use a HSPICE circuit level tool to estimate SER accurately in these custom benchmark circuits. We also estimate the SER variations for bigger ISCAS-85 benchmarks with respect to static threshold changes in circuits. This section describes the two experimental setups that are used for SER estimation in the benchmark circuits.

The HSPICE estimation tool uses an accurate method for SER estimation. This tool requires a SPICE netlist of the circuit for which the SER needs to be estimated. This tool provides an SER for a given current pulse generated by a pulse strike. A double exponential current pulse similar to the ones used in [14],[79]. The HSPICE tool then calculates the Timing Window (tw) for which this current pulse at each node in the logic circuit causes an error at the output of the flip-flop. As defined in [14], timing window is the amount of time for which the current pulse at a node cause an error at the flip-flop output divided by the clock period. Thus, tw is the probability that an error occurs given a current pulse at a node N. If the probability of the current pulse occurring at this node N is known, then as in [14], the SER for that circuit (with N nodes and a single output O)can be calculated using the following equation.

$$SER_O = \Sigma_N P_N * tw \tag{4.5}$$

Apart from being accurate, the HSPICE also allows the user to change parameters of individual devices that occur due to variations. But the HSPICE tool works effectively for only very small circuits with 5-10 gates. When circuits get bigger, analysis using HSPICE becomes very tedious and time consuming. Hence to analyze SER variations in bigger ISCAS benchmarks, we use SEAT-LA. SEAT-LA requires a gate level design of the circuit to estimate the SER. In [14], a detailed explanation on how SEAT-LA can be used to predict SER in logic circuits is presented.

4.4 Experimental results

We used a set of custom designed circuit layouts and gate level designs of ISCAS-85 benchmarks for our simulations. The custom designed circuits were laid out in 70nm PTM technology [?] and simulated using HSPICE. These circuits include a ten stage inverter chain, logic chain similar to the one used in [14], C17 ISCAS benchmark, a 2X4 decoder and a four bit Ripple Carry Adder (RCA). The gate level designs of the ISCAS-85 benchmarks were tested using SEAT-LA.

4.4.1 Static Variations

In our first set of experiments on static variations, we tested out the impact of inter-die variations on both the custom designed benchmarks and ISCAS-85 benchmarks. Results of these tests are shown in Figure 4.4 and show a maximum variation of 15.93% for a 3σ (maximum) variation of 10% in threshold voltage (ΔV_{th}). Importantly, we observe an increase in the SER as threshold increases. This trend is the opposite of what is expected and explained in [79].

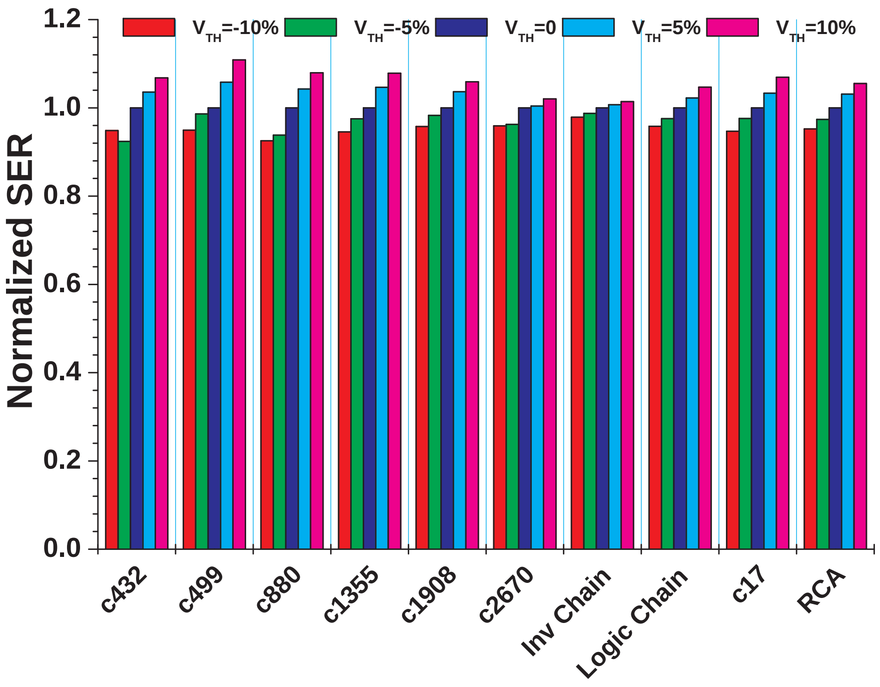


Figure 4.4. Normalized SER due to static variation for ISCAS and custom benchmarks

As mentioned in [79], there are two different phenomena that determine the trend of SER with change in threshold voltage. The first being the increase in gain of static logic circuits with increase in threshold voltage. This in turn reduces the electrical masking capability of static logic circuits resulting in increase in SER. Also, the flip-flop set-up and hold time increases with increased threshold voltage. This increases the (latch-window) masking capability of the flip-flops as larger pulses are now needed to get latched on by the flip-flop. Thus, the SER trend depends on which of the above factor dominates more. In [79], it was found that the SER decreased with large increase in V_{th}. This was due to the large increase in the flip-flop set-up and hold times while the increase in gain of the logic circuits had a lot lesser effect on the overall SER trend.

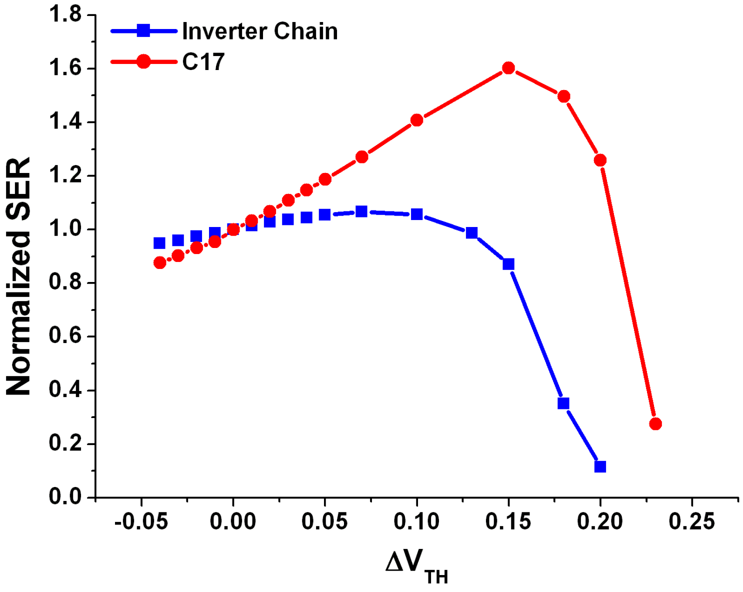


Figure 4.5. SER variation with increase in V_{th}

Since our initial analysis on SER variation due to variations in V_{th} had an opposite trend to what is expected as in [79], we extended our experiments to estimate the change in SER by increasing V_{th} beyond 10% for two of the smaller benchmarks. These results are presented in figure 4.11. From the figure it is noted that there is an initial linear increase in SER with small increase in V_{th}. This is clearly due to the dominant influence of CMOS gate gains in the logic circuits. But as V_{th} increases further, the reduction in setup and hold times of the flip-flops at the end of the data-paths starts playing a more important role, thus reducing SER drastically after a point as seen in figure 4.11.

These curves indicate interesting trade-offs that can be used for design optimizations in circuits for both power and SER mitigation. Hence, this can be used to estimate the V_{th} for which the circuits can be used to save leakage power and also decrease SER. Also, this technique can be applied for noise mitigation in general rather than just mitigating radiation induced soft errors.

To model the effect of intra-die random variations, we performed SER analysis on two of the custom benchmarks with random V_{th} assignments for the different devices. A Gaussian distribution in threshold voltage with $\mu = 0.2$ and $\sigma = 0.02$ for NMOS and $\mu = 0.22$ and $\sigma = 0.02$ for PMOS was used to assign V_{th} to each of the devices. Once the V_{th} was assigned for each device in the circuit, the circuit was simulated for SER analysis. Due to long simulation times, SER analyses for 10 different assignments were performed. Figure 4.6 shows the variation of SER normalized to SER for nominal threshold values. The peak-peak variation of 41% was found for these simulations. As seen in the Figure, large variation in SER was seen even for a small sample of V_{th}. Thus, a faster way to model SER variations with random variations is required to do a deeper analysis.

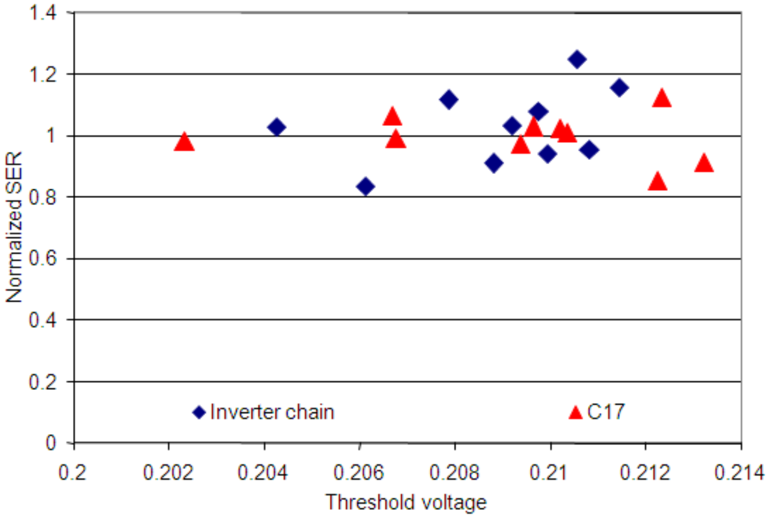


Figure 4.6. Impact of random V_{th} variations

4.4.2 Dynamic Variations in Power Supply and Temperature

Our next set of simulations studied the effects of power supply variations on the SER on different benchmark circuits. Figure 4.7 presents these and shows that for a 10% fluctuation of variation in power supply the SER varied by a maximum of 24.85% among all the circuits considered. Here as expected, the SER increased with decrease in voltage. Also, it is interesting to note that the variation in SER is almost linear with respect to change in voltage.

Next we also studied the effect of increased temperature on SER on the custom benchmark circuits. Figure 4.8 shows the variations in SER with increase in temperature. As seen here, there are vast differences in trends in different design. To

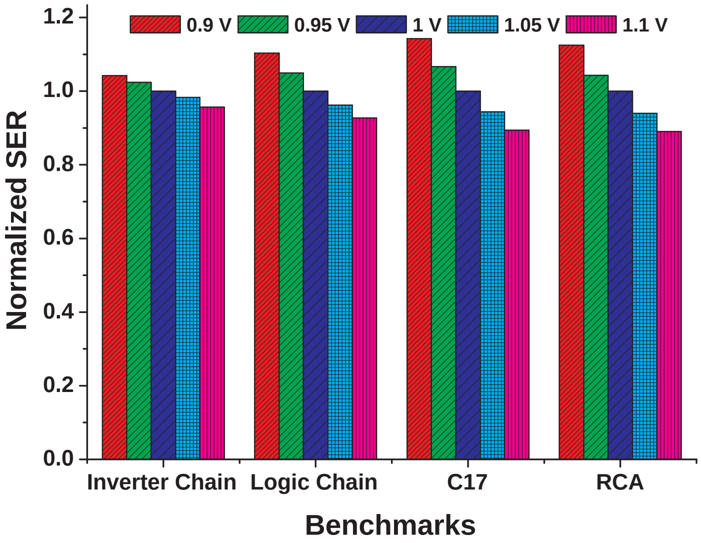


Figure 4.7. Variation of SER with power supply variations

determine the reasons for the different trends, it is important to study the effect of temperature on flip-flop characteristics. For this, the tw for trapezoidal pulses with increasing widths were studied. Here it was noted that the timing window was much lower at higher temperatures for small pulse widths while it was similar or in fact slightly greater (than that for lower temperatures) for large pulse widths. Thus, for designs with longer data-paths, the pulse widths that reach the flip-flop are very small resulting in very small SER at higher temperatures as seen for inverter chain and decoder in figure 4.8. On the other hand, for designs with very short data paths like C17, the pulse width (due to the same strike) reaching the flip-flop is larger thus having an opposite trend in SER as seen in figure 4.8.

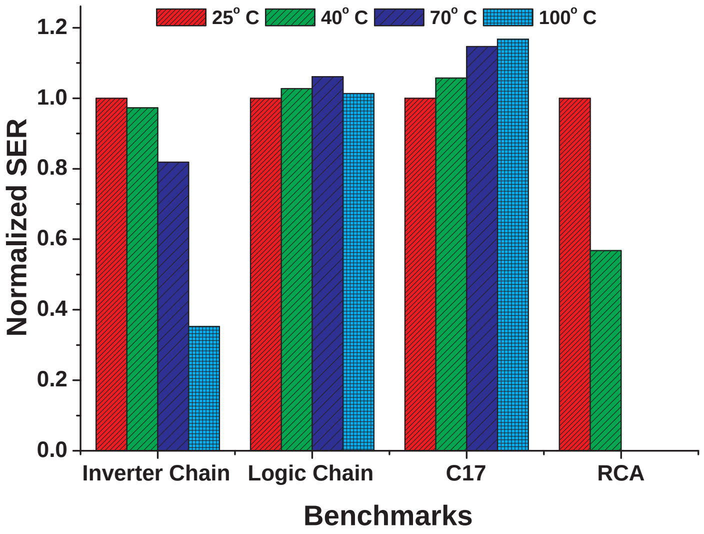


Figure 4.8. Effect of temperature on SER

4.4.3 Variations Due to Aging

Next, we performed experiments on the custom benchmark circuits to find the effect of device degradation due to HCE by using the tool discussed in section 4.3. Thus we studied the effect of HCE on the threshold voltages of different NMOS transistors in each of the circuits. These threshold variations were obtained by assuming a 50% switching activity at all the inputs of each of the circuit considered. The variation of average V_{th} after every 200 days in these circuits is presented in figure 4.12. From figure 4.12, it is clear that the V_{th} variation depends on the type of circuit. For example in circuits like inverter chain and C17, where the input transitions results in large number of transitions of devices in the circuit, there is a large increase in V_{th} with age. For circuits like logic chain and RCA, there is not a large increase in the V_{th}. We also studied the degradation of PMOS devices due

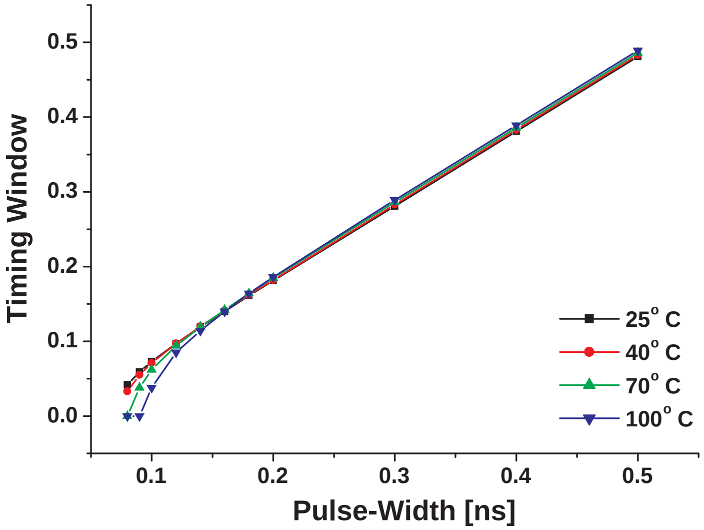


Figure 4.9. Reason for different trends in SER at different temperatures

to NBTI modeled as presented in section 4.2.3.2. For this, the static probability of each of the nodes was obtained based on a 0.5 input transition probability for all the circuits using which the change in V_{th} in each PMOS device in the circuits was obtained. Figure 4.13 presents the average variation of V_{th} of the PMOS devices in different circuits. As seen from the figure 4.13, there is only a small increase in threshold, which is in fact smaller than those observed for HCE. Next, SER for different bench mark circuits was obtained by incorporating the changes in V_{th} due to the two aging effects. Figure 4.14 presents the results obtained for aging effects on SER. As expected, the SER variations are not as much because of very small variations in the threshold voltages due to aging. Also, there is not a very clear trend in these variations mainly because the change in V_{th} is different for an NMOS and PMOS due to the two effects which make it difficult to predict the

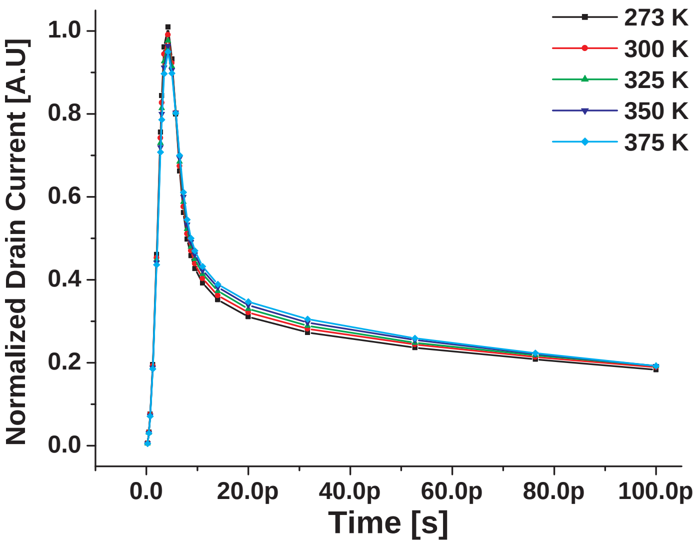


Figure 4.10. SER variation with increase in V_{th}

masking effects of different designs.

Finally, simulations were performed on one of the benchmark circuits (inverter chain) after incorporating different variations simultaneously to find the overall impact on SER. Table 4.1 shows the overall impact of these variations in SER. Here, the first column (%dvt) is the percentage change in V_{th} due to inter-die variations, the second column (days) represents the age of the circuit in number of days, the third column (Temp) represents the temperature at which the circuit works in Celsius and the fourth column presents the normalized SER values for each case. From these results, it can be noted that temperature has the maximum impact on SER as the results are comparable to its effects on inverter chain as shown in figure 4.14. As expected, aging has very little impact when compared to the other variations on SER.

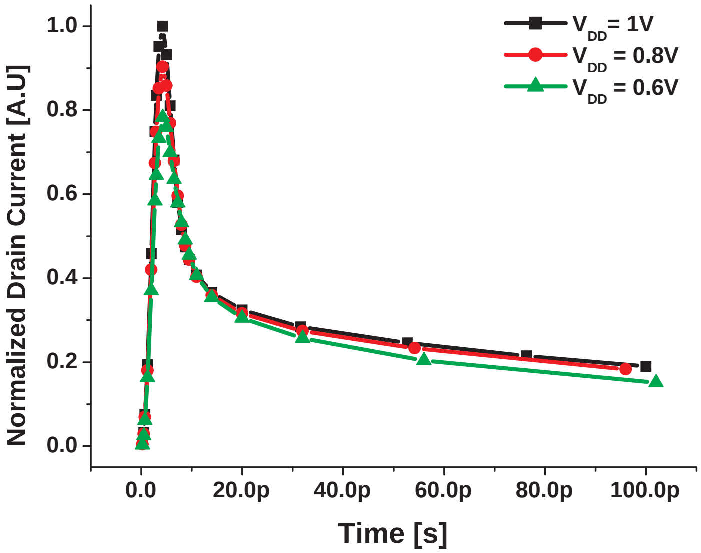


Figure 4.11. SER variation with increase in V_{th}

Table 4.1. Overall variation impact on SER of inverter chain

%dvt	Days	Temp(°C)	SER
0	0	25	1
5	999	100	0.422096
10	600	100	0.40085
10	999	70	0.842776
10	999	100	0.403683
-5	999	100	0.450425
-10	600	100	0.494334
-10	999	70	0.878187
-10	999	100	0.471671

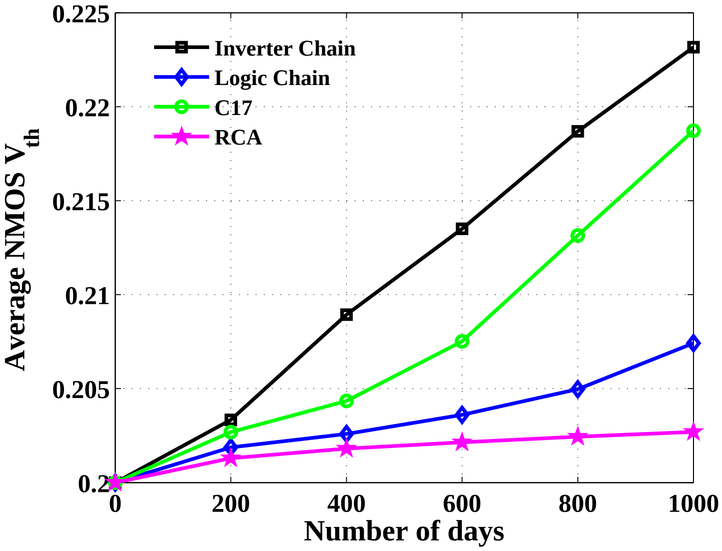


Figure 4.12. V_{th} variations due to HCE

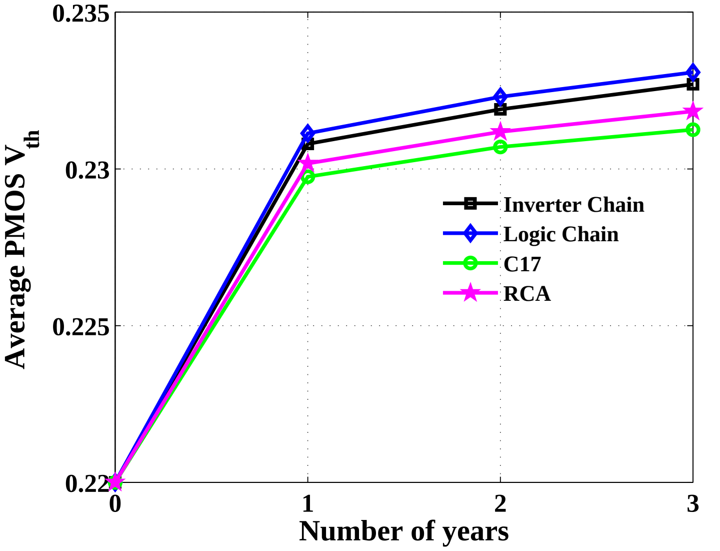


Figure 4.13. V_{th} variations due to NBTI

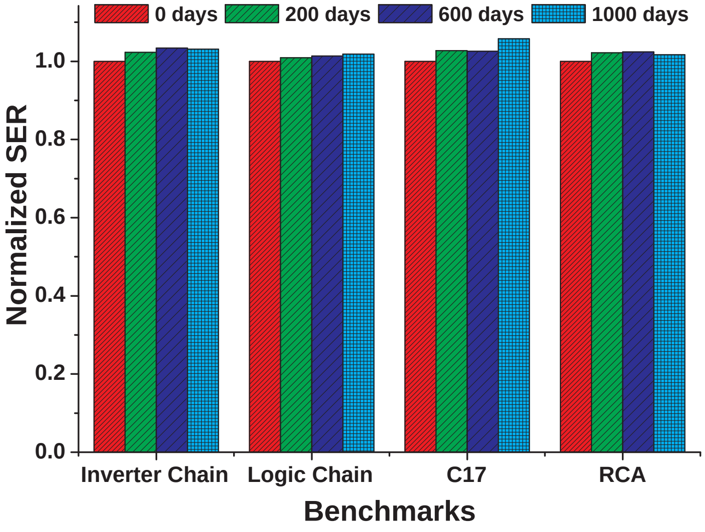


Figure 4.14. SER variations due to NBTI and HCE

Chapter 5

Conclusion & Future Work

Failure mechanisms that have contributed to the destruction of semiconductor memories and combinational circuits have been described. A new tool for faster estimation of SER known as HSEET for hierarchical structures have been proposed and is shown to have very high speedup over existing tools. Also, a complete framework for estimation of degradation due to NBTI in digital circuits and memories have been designed. The impact of variations on the SER of combinational circuits have been analyzed in detail. The impact of NBTI on FPGAs have been analyzed and solutions have been proposed to enhance the lifetime of these FPGAs. Sequential circuits play an important role in microprocessor design and hence the impact of NBTI on sequential circuits have also been analyzed. Tools or methods for faster estimation of SER due to intra die process variations are to be built for precise estimation. The impact of failure mechanisms in emerging devices such as tri-gate architectures and inter band tunnel transistors (TFETs) are to be analyzed to ensure high reliability margins.

Bibliography

[1] McPherson, J. W. (2006) "Reliability challenges for 45nm and beyond," in *DAC '06: Proceedings of the 43rd annual Design Automation Conference*, ACM, New York, NY, USA, pp. 176–181. 1

[2] Kimizuka, N., T. Yamamoto, T. Mogami, K. Yamaguchi, K. Imai, and T. Horiuchi (1999) "The impact of bias temperature instability for direct-tunneling ultra-thin gate oxide on MOSFET scaling," in *VLSI Technology, 1999. Digest of Technical Papers. 1999 Symposium on*, pp. 73–74. 5, 82

[3] Wang, W. (2008) *Circuit aging in scaled cmos design: modeling, simulation and prediction*, Ph.D. thesis, Arizona State University, Tempe, Arizona. 5

[4] Bersuker, G., J. Sim, C. Park, C. Young, S. Nadkarni, R. Choi, and B. Lee (2006) "Intrinsic Threshold Voltage Instability of the HFO2 NMOS Transistors," in *Reliability Physics Symposium Proceedings, 2006. 44th Annual., IEEE International*, pp. 179–183. 5

[5] Kwong, D.-L. (2004) "CMOS integration issues with high-k gate stack," in *Physical and Failure Analysis of Integrated Circuits, 2004. IPFA 2004. Proceedings of the 11th International Symposium on the*, pp. 17–20. 5

[6] Baumann, R. (2001) "Soft errors in advanced semiconductor devices-part I: the three radiation sources," *Device and Materials Reliability, IEEE Transactions on*, **1**(1), pp. 17–22. 7

[7] ——— (2005) "Soft Errors in Advanced Computer Systems," *IEEE Des. Test*, **22**(3), pp. 258–266. 7

[8] ——— (2005) "Radiation-induced soft errors in advanced semiconductor technologies," *Device and Materials Reliability, IEEE Transactions on*, **5**(3), pp. 305–316. 7

[9] Hazucha, P. and C. Svensson (2000) "Impact of CMOS technology scaling on the atmospheric neutron soft error rate," *Nuclear Science, IEEE Transactions on*, **47**(6), pp. 2586–2594. 7, 89, 90

[10] Shivakumar, P., M. Kistler, S. Keckler, D. Burger, and L. Alvisi (2002) "Modeling the effect of technology trends on the soft error rate of combinational logic," in *Dependable Systems and Networks, 2002. Proceedings. International Conference on*, pp. 389–398. 8, 105

[11] Massengill, L. W., M. S. Reza, B. Bhuva, and T. Turflinger (1998) "Single event upset cross-section modeling in combinational CMOS logic circuits," *Journal of Radiation Effects, Research and Engineering*. 8, 11

[12] Miskov-Zivanov, N. and D. Marculescu (2006) "MARS-C: modeling and reduction of soft errors in combinational circuits," in *DAC '06: Proceedings of the 43rd annual Design Automation Conference*, ACM, New York, NY, USA, pp. 767–772. 8, 12

[13] Zhao, C., X. Bai, and S. Dey (2004) "A scalable soft spot analysis methodology for compound noise effects in nano-meter circuits," in *DAC '04: Proceedings of the 41st annual Design Automation Conference*, ACM, New York, NY, USA, pp. 894–899. 8, 11, 30

[14] Rajaraman, R., J. Kim, N. Vijaykrishnan, Y. Xie, and M. Irwin (2006) "SEAT-LA: a soft error analysis tool for combinational logic," in *VLSI Design, 2006. Held jointly with 5th International Conference on Embedded Systems and Design., 19th International Conference on*, p. 4pp. 8, 9, 11, 12, 15, 29, 31, 105, 114, 115

[15] Zhang, M. and N. R. Shanbhag (2004) "A soft error rate analysis (SERA) methodology," in *ICCAD '04: Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design*, IEEE Computer Society, Washington, DC, USA, pp. 111–118. 11, 20, 30

[16] Dhillon, Y. S., A. U. Diril, and A. Chatterjee (2005) "Soft-Error Tolerance Analysis and Optimization of Nanometer Circuits," in *DATE '05: Proceedings of the conference on Design, Automation and Test in Europe*, IEEE Computer Society, Washington, DC, USA, pp. 288–293. 11

[17] Zhang, B., W.-S. Wang, and M. Orshansky (2006) "FASER: Fast Analysis of Soft Error Susceptibility for Cell-Based Designs," in *ISQED '06: Proceedings of the 7th International Symposium on Quality Electronic Design*, IEEE Computer Society, Washington, DC, USA, pp. 755–760. 11

[18] Rao, R., K. Chopra, D. Blaauw, and D. Sylvester (2006) “An Efficient Static Algorithm for Computing the Soft Error Rates of Combinational Circuits,” in *Design, Automation and Test in Europe, 2006. DATE ’06. Proceedings*, vol. 1, pp. 1–6. 11, 14, 30

[19] Krishnaswamy, S., G. Viamontes, I. Markov, and J. Hayes (2005) “Accurate reliability evaluation and enhancement via probabilistic transfer matrices,” in *Design, Automation and Test in Europe, 2005. Proceedings*, pp. 282–287 Vol. 1. 12

[20] Zhao, W. and Y. Cao (2006) “New Generation of Predictive Technology Model for Sub-45nm Design Exploration,” in *ISQED ’06: Proceedings of the 7th International Symposium on Quality Electronic Design*, IEEE Computer Society, Washington, DC, USA, pp. 585–590. 13, 34, 84, 86

[21] Dharchoudhury, A., S. M. Kang, H. Cha, and J. H. Patel (1994) “Fast timing simulation of transient faults in digital circuits,” in *ICCAD ’94: Proceedings of the 1994 IEEE/ACM international conference on Computer-aided design*, IEEE Computer Society Press, Los Alamitos, CA, USA, pp. 719–722. 13

[22] Zhou, Q. and K. Mohanram (2004) “Cost-effective radiation hardening technique for combinational logic,” in *ICCAD ’04: Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design*, IEEE Computer Society, Washington, DC, USA, pp. 100–106. 14

[23] Deogun, H. S., D. Sylvester, and D. Blaauw (2005) “Gate-Level Mitigation Techniques for Neutron-Induced Soft Error Rate,” in *ISQED ’05: Proceedings of the 6th International Symposium on Quality of Electronic Design*, IEEE Computer Society, Washington, DC, USA, pp. 175–180. 14

[24] Lamoureux, J. and S. Wilton (2006) “Activity Estimation for Field-Programmable Gate Arrays,” in *Field Programmable Logic and Applications, 2006. FPL ’06. International Conference on*, pp. 1–8. 20, 28

[25] Kaczer, B., V. Arkhipov, M. Jurczak, and G. Groeseneken (2005) “Negative bias temperature instability (NBTI) in SiO2 and SiON gate dielectrics understood through disorder-controlled kinetics,” *Microelectron. Eng.*, **80**(1), pp. 122–125. 34

[26] Alam, M. and S. Mahapatra (Jan 2005) “A comprehensive model of PMOS NBTI degradation,” *Microelectronics Reliability*, **45**(1), pp. 75–81. 34, 36, 83, 113

[27] Mahapatra, S., P. Kumar, and M. Alam (2004) "Investigation and modeling of interface and bulk trap generation during negative bias temperature instability of p-MOSFETs," *Electron Devices, IEEE Transactions on*, **51**(9), pp. 1371–1379. 34

[28] Zafar, S., B. Lee, and J. Stathis (2004) "Evaluation of NBTI in HfO2 gate-dielectric stacks with tungsten gates," *Electron Device Letters, IEEE*, **25**(3), pp. 153–155. 34

[29] Alam, M. (2003) "A critical examination of the mechanics of dynamic NBTI for PMOSFETs," in *Electron Devices Meeting, 2003. IEDM '03 Technical Digest. IEEE International*, pp. 14.4.1–14.4.4. 34

[30] Varghese, D., S. Mahapatra, and M. Alam (2005) "Hole energy dependent interface trap generation in MOSFET Si/SiO2 interface," *Electron Device Letters, IEEE*, **26**(8), pp. 572–574. 34

[31] Chakravarthi, S., A. Krishnan, V. Reddy, C. Machala, and S. Krishnan (2004) "A comprehensive framework for predictive modeling of negative bias temperature instability," in *Reliability Physics Symposium Proceedings, 2004. 42nd Annual. 2004 IEEE International*, pp. 273–282. 34

[32] Wang, W., S. Yang, S. Bhardwaj, R. Vattikonda, S. Vrudhula, F. Liu, and Y. Cao (2007) "The impact of NBTI on the performance of combinational and sequential circuits," in *Proceedings of the IEEE Design Automation Conference (DAC)*. 34, 53, 60

[33] Wang, W., V. Reddy, A. Krishnan, R. Vattikonda, S. Krishnan, and Y. Cao (16-19 Sept. 2007) "An Integrated Modeling Paradigm of Circuit Reliability for 65nm CMOS Technology," in *CICC '07: Proceedings of IEEE Custom Integrated Circuits Conference, 2007*, pp. 511–514. 34, 37, 61

[34] Kumar, S. V., C. H. Kim, and S. S. Sapatnekar (2006) "Impact of NBTI on SRAM Read Stability and Design for Reliability," in *ISQED '06: Proceedings of the 7th International Symposium on Quality Electronic Design*, IEEE Computer Society, Washington, DC, USA, pp. 210–218. 34, 59, 60, 87

[35] Paul, B. C., K. Kang, H. Kufluoglu, M. A. Alam, and K. Roy (2006) "Temporal performance degradation under NBTI: estimation and design for improved reliability of nanoscale circuits," in *DATE '06: Proceedings of the conference on Design, automation and test in Europe*, pp. 780–785. 34, 59, 60

[36] Kumar, S. V., C. H. Kim, and S. S. Sapatnekar (2007) "NBTI-aware synthesis of digital circuits," in *DAC '07: Proceedings of the 44th annual Design Automation Conference*, ACM, New York, NY, USA, pp. 370–375. 34

[37] Wang, W., Z. Wei, S. Yang, and Y. Cao (2007) "An efficient method to identify critical gates under circuit aging," in *ICCAD '07: Proceedings of the 2007 IEEE/ACM international conference on Computer-aided design*, IEEE Press, Piscataway, NJ, USA, pp. 735–740. 34, 44, 61

[38] Abella, J., X. Vera, and A. Gonzalez (2007) "Penelope: The NBTI-Aware Processor," in *Microarchitecture, 2007. MICRO 2007. 40th Annual IEEE/ACM International Symposium on*, pp. 85–96. 34

[39] Burger, D. and T. Austin, "The SimpleScalar Toolset, Version 3.0," Http://www.simplescalar.com. 41

[40] Huang, W., K. Sankaranarayanan, R. J. Rib, M. R. Stan, and K. Skadron (2007) "An Improved Block-Based Thermal Model in HotSpot 4.0 with Granularity Considerations," in *WDDD: Workshop on Duplicating, Deconstructing, and Debunking, in conjunction with the 34th International Symposium on Computer Architecture (ISCA)*, pp. 370–375. 42

[41] Gieseke, B., R. Allmon, D. Bailey, B. Benschneider, S. Britton, J. Clouser, I. Fair, H.R., J. Farrell, M. Gowan, C. Houghton, J. Keller, T. Lee, D. Leibholz, S. Lowell, M. Matson, R. Matthew, V. Peng, M. Quinn, D. Priore, M. Smith, and K. Wilcox (1997) "A 600 MHz superscalar RISC microprocessor with out-of-order execution," in *Solid-State Circuits Conference, 1997. Digest of Technical Papers. 43rd ISSCC., 1997 IEEE International*, pp. 176–177, 451. 42

[42] Meyer, D. (1998) "AMD-K7(TM)$^{\mathrm{TM}}$," in *Technology Presentation*. 42

[43] Wang, N., J. Quek, T. Rafacz, and S. Patel (2004) "Characterizing the effects of transient faults on a high-performance processor pipeline," in *Dependable Systems and Networks, 2004 International Conference on*, pp. 61–70. 42, 43

[44] Henning, J. L. (2000) "SPEC CPU2000: Measuring CPU Performance in the New Millennium," *Computer*, **33**(7), pp. 28–35. 43

[45] (2007), "International Technology Roadmap for Semiconductors," . 5, 47, 103

[46] Shoji, M. (1992) *Theory of CMOS Digital Circuits and Circuit Failures*, Princeton University Press, Princeton, NJ. 59

[47] Qi, Z. and M. R. Stan (2008) "NBTI Resilient Circuits Using Adaptive Body Biasing," in *GLSVLSI '08: Proceedings of the IEEE/ACM Great Lakes Symposium on VLSI*. 60

[48] Keane, J., T.-H. Kim, and C. H. Kim (2007) "An on-chip NBTI sensor for measuring PMOS threshold voltage degradation," in *ISLPED '07: Proceedings of the 2007 international symposium on Low power electronics and design*, ACM, New York, NY, USA, pp. 189–194. 60

[49] Abrishami, H., S. Hatami, B. Amelifard, and M. Pedram (2008) "NBTI-Aware flip-flop characterization and design," in *GLSVLSI '08: Proceedings of the IEEE/ACM Great Lakes Symposium on VLSI*. 60

[50] Stojanovic, V. and V. Oklobdzija (Apr 1999) "Comparative analysis of master-slave latches and flip-flops for high-performance and low-power systems," *Solid-State Circuits, IEEE Journal of*, **34**(4), pp. 536–548. 62

[51] Unger, S. H. and C.-J. Tan (1986) "Clocking Schemes for High-Speed Digital Systems," *IEEE Trans. Comput.*, **35**(10), pp. 880–895. 62, 64

[52] Oklobdzija, V. G. (2003) "Clocking and clocked storage elements in a multi-gigahertz environment," *IBM J. Res. Dev.*, **47**(5-6), pp. 567–583. 64

[53] Partovi, H., R. Burd, U. Salim, F. Weber, L. DiGregorio, and D. Draper (Feb 1996) "Flow-through latch and edge-triggered flip-flop hybrid elements," *Solid-State Circuits Conference, 1996. Digest of Technical Papers. 42nd ISSCC., 1996 IEEE International*, pp. 138–139. 67

[54] Klass, F., C. Amir, A. Das, K. Aingaran, C. Truong, R. Wang, A. Mehta, R. Heald, and G. Yee (May 1999) "A new family of semidynamic and dynamic flip-flops with embedded logic for high-performance processors," *Solid-State Circuits, IEEE Journal of*, **34**(5), pp. 712–716. 67

[55] Zhao, W. and Y. Cao (2006) "New Generation of Predictive Technology Model for Sub-45nm Design Exploration," in *ISQED '06: Proceedings of the 7th International Symposium on Quality Electronic Design*, IEEE Computer Society, Washington, DC, USA, pp. 585–590. 68

[56] Vattikonda, R., W. Wang, and Y. Cao (2006) "Modeling and minimization of PMOS NBTI effect for robust nanometer design," in *DAC '06: Proceedings of the 43rd annual conference on Design automation*, ACM, New York, NY, USA, pp. 1047–1052. 69, 83, 104, 113

[57] Reddy, V., A. Krishnan, A. Marshall, J. Rodriguez, S. Natarajan, T. Rost, and S. Krishnan (2002) "Impact of negative bias temperature instability on digital circuit reliability," in *Reliability Physics Symposium Proceedings, 2002. 40th Annual*, pp. 248–254. 82

[58] Srinivasan, J., S. Adve, P. Bose, and J. Rivers (2004) "The impact of technology scaling on lifetime reliability," in *Dependable Systems and Networks, 2004 International Conference on*, pp. 177–186.

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

[60] Srinivasan, S., P. Mangalagiri, Y. Xie, N. Viiaykrishnan, and K. Sarpatwari (2006) "FLAW: FPGA lifetime awareness," in *Design Automation Conference, 2006 43rd ACM/IEEE*, pp. 630–635. 98

[61] Samudrala, P., J. Ramos, and S. Katkoori (2004) "Selective triple Modular redundancy (STMR) based single-event upset (SEU) tolerant synthesis for FPGAs," *Nuclear Science, IEEE Transactions on*, **51**(5), pp. 2957–2969.

[62] Schroder, D. K. and J. A. Babcock (2003) "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," *Journal of Applied Physics*, **94**(1), pp. 1–18.
URL http://link.aip.org/link/?JAP/94/1/1 83

[63] Massey, J. (2004) "NBTI: what we know and what we need to know - a tutorial addressing the current understanding and challenges for the future," in *Integrated Reliability Workshop Final Report, 2004 IEEE International*, pp. 199–211. 83

[64] (2006), "Power Consumption in 65nm FPGAs," White Paper. 87

[65] Betz, V. and J. Rose (1997) "VPR: A new packing, placement and routing tool for FPGA research," in *FPL*, pp. 213–222. 93

[66] Srinivasan, S., A. Gayasen, N. Vijaykrishnan, M. Kandemir, Y. Xie, and M. Irwin (2004) "Improving soft-error tolerance of FPGA configuration bits," in *Computer Aided Design, 2004. ICCAD-2004. IEEE/ACM International Conference on*, pp. 107–110. 96

[67] "Xilinx product datasheet and application tools for Virtex-II," . 97

[68] Srinivasan, S., A. Gayasen, and N. Vijaykrishnan (2005) "Leakage control in FPGA routing fabric," in *Design Automation Conference, 2005. Proceedings of the ASP-DAC 2005. Asia and South Pacific*, vol. 1, pp. 661–664 Vol. 1. 98

[69] Keller, E. (2000) "JRoute: A Run-Time Routing API for FPGA Hardware," in *IPDPS '00: Proceedings of the 15 IPDPS 2000 Workshops on Parallel and Distributed Processing*, Springer-Verlag, London, UK, pp. 874–881. 98

[70] Borkar, S., T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De (2003) "Parameter variations and impact on circuits and microarchitecture," in *DAC '03: Proceedings of the 40th annual Design Automation Conference*, ACM, New York, NY, USA, pp. 338–342. 103, 104

[71] Bowman, K., S. Duvall, and J. Meindl (2002) "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *Solid-State Circuits, IEEE Journal of*, **37**(2), pp. 183–190. 103, 106

[72] Ajami, A. H., K. Banerjee, and M. Pedram (2005) "Scaling Analysis of On-Chip Power Grid Voltage Variations in Nanometer Scale ULSI," in *in Namometer Scale ULSI, Journal of Analog Integrated Circuits and Signal Processing*, Springer, pp. 277–290. 104, 107

[73] Power, J., R. Clancy, W. Wall, A. Mathewson, and W. Lane (1992) "An investigation of MOSFET statistical and temperature effects," *Microelectronic Test Structures, 1992. ICMTS 1992. Proceedings of the 1992 International Conference on*, pp. 202–207. 104, 108

[74] Xuan, X., A. Chatterjee, and A. Singh (2004) "Local redesign for reliability of CMOS digital circuits under device degradation," in *Reliability Physics Symposium Proceedings, 2004. 42nd Annual. 2004 IEEE International*, pp. 651–652. 104

[75] Agarwal, A., B. Paul, S. Mukhopadhyay, and K. Roy (2005) "Process variation in embedded memories: failure analysis and variation aware architecture," *Solid-State Circuits, IEEE Journal of*, **40**(9), pp. 1804–1814. 106

[76] Chen, H. H. and D. D. Ling (1997) "Power supply noise analysis methodology for deep-submicron VLSI chip design," in *DAC '97: Proceedings of the 34th annual conference on Design automation*, ACM, New York, NY, USA, pp. 638–643. 107

[77] Fang, P., J. Tao, J. Chen, and C. Hu (1998) "Design in hot-carrier reliability for high performance logic applications," in *Custom Integrated Circuits Conference, 1998. Proceedings of the IEEE 1998*, pp. 525–531. 109

[78] Jean, Y.-S. and C.-Y. Wu (1997) "The threshold-voltage model of MOSFET devices with localized interface charge," *Electron Devices, IEEE Transactions on*, **44**(3), pp. 441–447. 109, 110

[79] Degalahal, V., R. Ramanarayanan, N. Vijaykrishnan, Y. Xie, and M. Irwin (2004) "The effect of threshold voltages on the soft error rate

[memory and logic circuits],” in *Quality Electronic Design, 2004. Proceedings. 5th International Symposium on.* 114, 116, 117

Vita

Ramakrishnan Krishnan

Ramakrishnan Krishnan obtained his Bachelors in Electronics and Communications Engineering from the National Institute of Technology Karnataka, Surathkal in the year 2004. He joined the Ph.D program in the Department of Electrical Engineering at the Pennsylvania State University the same year. He has published in various IEEE and ACM conferences and journals. He has served as a technical reviewer for several IEEE and ACM conferences and journals such as ICCAD, ISVLSI, DAC, VLSI Design, ISPASS, ICCD and TVLSI. He is also a graduate student member of IEEE.