Design of Variation-Tolerant Circuits for Nanometer CMOS Technology: Circuits and Architecture Co-Design

by

Mohamed Hassan Abu-Rahma

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

Waterloo, Ontario, Canada, 2008

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

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

Abstract

Aggressive scaling of CMOS technology in sub-90nm nodes has created huge challenges. Variations due to fundamental physical limits, such as random dopants fluctuation (RDF) and line edge roughness (LER) are increasing significantly with technology scaling. In addition, manufacturing tolerances in process technology are not scaling at the same pace as transistor's channel length due to process control limitations (*e.g.*, sub-wavelength lithography). Therefore, within-die process variations worsen with successive technology generations. These variations have a strong impact on the maximum clock frequency and leakage power for any digital circuit, and can also result in functional yield losses in variation-sensitive digital circuits (such as SRAM). Moreover, in nanometer technologies, digital circuits show an increased sensitivity to process variations due to low-voltage operation requirements, which are aggravated by the strong demand for lower power consumption and cost while achieving higher performance and density. It is therefore not surprising that the International Technology Roadmap for Semiconductors (ITRS) lists variability as one of the most challenging obstacles for IC design in nanometer regime.

To facilitate variation-tolerant design, we study the impact of random variations on the delay variability of a logic gate and derive simple and scalable statistical models to evaluate delay variations in the presence of within-die variations. This work provides new design insight and highlights the importance of accounting for the effect of input slew on delay variations, especially at lower supply voltages. The derived models are simple, scalable, bias dependent and only require the knowledge of easily measurable parameters. This makes them useful in early design exploration, circuit/architecture optimization as well as technology prediction (especially in low-power and low-voltage operation). The derived models are verified using Monte Carlo SPICE simulations using industrial 90nm technology.

Random variations in nanometer technologies are considered one of the largest design considerations. This is especially true for SRAM, due to the large variations in bitcell characteristics. Typically, SRAM bitcells have the smallest device sizes on a chip. Therefore, they show the largest sensitivity to different sources of variations. With the drastic increase in memory densities, lower supply voltages and higher variations, statistical simulation methodologies become imperative to estimate memory yield and optimize performance and power. In this research, we present a methodology for statistical simulation of SRAM read access yield, which is tightly related to SRAM performance and power consumption. The proposed flow accounts for the impact of bitcell read current variation, sense amplifier offset dis-

tribution, timing window variation and leakage variation on functional yield. The methodology overcomes the pessimism existing in conventional worst-case design techniques that are used in SRAM design. The proposed statistical yield estimation methodology allows early yield prediction in the design cycle, which can be used to trade off performance and power requirements for SRAM. The methodology is verified using measured silicon yield data from a 1Mb memory fabricated in an industrial 45nm technology.

Embedded SRAM dominates modern SoCs and there is a strong demand for SRAM with lower power consumption while achieving high performance and high density. However, in the presence of large process variations, SRAMs are expected to consume larger power to ensure correct read operation and meet yield targets. We propose a new architecture that significantly reduces array switching power for SRAM. The proposed architecture combines built-in self-test (BIST) and digitally controlled delay elements to reduce the wordline pulse width for memories while ensuring correct read operation; hence, reducing switching power. A new statistical simulation flow was developed to evaluate the power savings for the proposed architecture. Monte Carlo simulations using a 1Mb SRAM macro from an industrial 45nm technology was used to examine the power reduction achieved by the system. The proposed architecture can reduce the array switching power significantly and shows large power saving - especially as the chip level memory density increases. For a 48Mb memory density, a 27% reduction in array switching power can be achieved for a read access yield target of 95%. In addition, the proposed system can provide larger power saving as process variations increase, which makes it a very attractive solution for 45nm and below technologies.

In addition to its impact on bitcell read current, the increase of local variations in nanometer technologies strongly affect SRAM cell stability. In this research, we propose a novel single supply voltage read assist technique to improve SRAM static noise margin (SNM). The proposed technique allows precharging different parts of the bitlines to V_{DD} and GND and uses charge sharing to precisely control the bitline voltage, which improves the bitcell stability. In addition to improving SNM, the proposed technique also reduces memory access time. Moreover, it only requires one supply voltage, hence, eliminates the need of large area voltage shifters. The proposed technique has been implemented in the design of a 512kb memory fabricated in 45nm technology. Results show improvements in SNM and read operation window which confirms the effectiveness and robustness of this technique.

Acknowledgements

I would like to give my sincere appreciation to my academic advisor Prof. Mohab Anis for his assistance, support, and strong encouragement throughout my research. I would also like to thank my committee members, Prof. Manoj Sachdev, Prof. Mark Aagaard, Prof. Eihab Abdel-Rahman, and Prof. Massoud Pedram. They have provided me with valuable comments and suggestions on my research work.

My stay at University of Waterloo was stimulating and entertaining, thanks to the friendship of Hassan Hassan, Hazem Shehata, Ahmed Youssef, Muhammad Nummer, Aymen Ismail, Mohamed Elsaid, Mohammed El-Abd, and many other friends. I also thank all my lab mates in the VLSI lab for many invaluable discussions on different research topics. I would also like to thank Lisa ter Woort from CECS for her support in my first internship with Qualcomm, which reshaped my research with industrial experience. I would like to acknowledge Wendy Boles, Annette Dietrich and Lisa Hendel from ECE graduate office for their help on administrative issues. I also thank Phil Regier for his great support for all IT related issues.

I am very appreciative of the support from all of my colleagues at Qualcomm Incorporated. In particular, I would like to express my gratitude to Sei Seung Yoon, Mehdi Sani, and Nick Yu for their technical leadership and support. I also thank my teammates Kinshuk Chowdhury, Muhammad Nasir, and Dongkyu Park for support and encouragement and many valuable discussions. I would like to thank Dr. Muhammed Khellah from Intel and Dr. Mohamed Elgebaly from Sun Microsystems for their help and support.

All of this was only made possible by the encouragement and love of my family. My deepest gratitude goes to my mother and father for their never ending support, and for remembering me in their prayers. No words of appreciation could ever reward them for all they have done for me. I am, and will ever be, indebted to them for all the achievements in my life. I am also thankful to my sisters Marwa, Mona, and Maha, for their endless support. My loving wife, Enas, shared with me every moment of my Ph.D. (even when she was away). She supported me with her unconditional love and care during all the ups and downs of my research, and she was always there to motivate me. I thank her for everything she have done.

Finally, I express my gratitude to Allah (God) for providing me the blessings and strength to complete this work.

Mohamed H. Abu-Rahma
November, 2008

Dedication

To my mother and father

Contents

List of Tables **x**

List of Figures **xi**

1 Introduction **1**

1.1 Motivation: Variation-Tolerant Design 1

1.2 Thesis Outline 2

2 Background **4**

2.1 Introduction 4

2.2 Sources of Variability: Device 5

2.2.1 Random Dopant Fluctuations (RDF) 5

2.2.2 Channel Length Variation 8

2.2.3 Other Sources 10

2.3 Sources of Variability: Interconnect 12

2.4 Sources of Variability: Environmental 13

2.5 Impact of Process Variations on Performance and Power 15

2.6 Techniques to Deal with Variability 17

2.6.1 Analysis and CAD 17

2.6.2 Circuits 18

2.6.3 Architecture 24

2.7 SRAM Scaling Trends 27

2.8 Variability and SRAM Failure Mechanisms 28

2.8.1 Hard and Catastrophic fails 28
2.8.2 Bitcell Stability Failures 31
2.8.3 Radiation Induced Soft Errors 37
2.9 Summary 39

3 A Statistical Design-Oriented Delay Variation Model Accounting for Within-Die Variations 41

3.1 Introduction 42
3.2 Model Assumptions and Derivation 43
3.2.1 Variation in Charging/Disharging Current 44
3.2.2 Impact of Finite Input Slew on Delay Variation 47
3.2.3 Minimum Relative Delay Variation σ_{Tp}/Tp 51
3.2.4 Input Slew Variation 54
3.3 Results and Discussion 55
3.4 Design Insights 64
3.5 Summary 66

4 A Methodology for Statistical Estimation of Read Access Yield in SRAMs 68

4.1 Introduction 68
4.2 Challenges of SRAM Statistical Design 70
4.3 Modeling of Read Access Failures 73
4.3.1 Read Current and Sensing Slope Variations 74
4.3.2 Sense Amplifier Variations 74
4.3.3 Sensing Window Variations 77
4.3.4 Pass-Gate Leakage 78
4.4 Proposed Yield Estimation Flow 80
4.5 Experimental Results 86
4.6 Summary 93

5 Reducing SRAM Power using Fine-Grained Wordline Pulse Width Control **94**

5.1 Introduction 94

5.2 SRAM Yield and Power Tradeoff 97

5.3 Fine-Grained Wordline Pulse Width Control 102

5.3.1 SRAM Built-in Self-Test 102

5.3.2 WL Programmable Delay Elements 103

5.3.3 Pulse Width Control Logic 104

5.3.4 System Operation 104

5.4 Results and Discussion 105

5.5 Summary 112

6 A Robust Single Supply Voltage SRAM Read Assist Technique Using Selective Precharge **114**

6.1 State of the Art Read Assist Techniques 114

6.2 Background 118

6.3 Selective Precharge Technique 119

6.4 Access Time Improvement 124

6.5 Results and Discussion 125

6.6 Summary 134

7 Conclusions **136**

7.1 Summary of Contributions 136

7.2 Future Research Directions 137

Appendices **139**

A Publications from this Work **139**

B Glossary **140**

References **142**

List of Tables

3.1 90nm Technology Information 55

4.1 Read failure model inputs for the proposed statistical yield estimation methodology. 83

6.1 State of the art read assist techniques. 117

6.2 512kb memory design information. 125

6.3 ΔV_{BL} for different conditions. 129

List of Figures

1.1 Different levels of abstraction studied in this research. 3

2.1 Atomistic process simulation incorporating random dopant fluctuation (RDF) and line edge roughness (LER) as the sources of intrinsic fluctuation. 6

2.2 Number of dopant atoms in the depletion layer of a MOSFET versus channel length L_{eff}. 7

2.3 Lithography wavelength scaling for different technology nodes. . . . 9

2.4 Measured V_{th} versus channel length L in a 90nm technology which shows strong short channel effects causing a sharp roll-off for V_{th} for shorter L. 9

2.5 Simulated V_{th} versus channel length L for IBM 130nm technology showing V_{th} versus L roll-off under different conditions, low and high V_{DS}, and forward (FBB) and reverse body bias (RBB). 10

2.6 Predicted $\sigma_{V_{th}}$ including RDF and LER versus technology nodes . . 11

2.7 A schematic cross-section of interconnect showing dishing and erosion impact on metal height. 13

2.8 Thermal image showing within die temperature variation for a microprocessor. Hot spots with temperatures as high as 120 °C are shown. 14

2.9 Dynamic and static power versus technology scaling, showing the exponential increase in leakage power. 15

2.10 Measured I_{on} versus I_{off} scatter plot showing large spread in I_{off} for a 0.15μm technology. 16

2.11 Leakage and frequency variations for an Intel processor in 0.13μm technology. 17

2.12 Block diagram of speed adaptive-V_{th} technique. 19

2.13 Measured leakage versus frequency distribution for 62 dies in a 150nm technology (V_{DD} = 1.1V, T=110C), showing the distributions after utilizing ABB and WID-ABB. In the lower figure, the percentage of accepted dies at a certain frequency bin are shown 21

2.14 Schematic showing the implementation of self-adjusted forward body bias (SA-FBB) circuit technique 22

2.15 Comparison between ZBB, FBB and SA-FBB showing measured V_{th} distribution in a 90nm technology. 23

2.16 Register file with programmable keeper to compensate for process variations impact on leakage. 24

2.17 The WID maximum critical path delay distribution for different values of independent critical paths N_{cp}. As N_{cp} increases, the mean of maximum critical path delay increases. 25

2.18 SRAM and logic area versus technology scaling. SRAM dominates chip area in modern SoCs and microprocessors. 27

2.19 SRAM bitcell area scaling from 350nm down to 45nm technology nodes. 28

2.20 Layout and planar SEM image of a 0.246um^2 ultra high density SRAM bitcell in an industrial 45nm technology node. 29

2.21 SRAM devices V_{th} variation scaling trend. 29

2.22 Different types of SRAM failures. 30

2.23 SRAM hard and soft fails scaling trend. 31

2.24 Bitcell in read operation. 33

2.25 SNM butterfly curves for a 45nm SRAM bitcell for different process corners. 34

2.26 Measured SNM butterfly curves for 512 bitcells in 65nm technology node. 34

2.27 Minimum SRAM supply voltage (V_{ccmin}) distribution before and after NBTI stress. 35

2.28 Bitcell in write operation. 36

2.29 SRAM Write trip simulation. 36

2.30 Data retention failure mechanism. 38

2.31 SER fail rate for different technology nodes. 39

3.1 Typical bias dependence of g_m/I_D versus overdrive voltage $V_{GS} - V_{th}$ for a device in saturation region. In strong inversion, g_m/I_D initially increases proportional to $(V_{GS} - V_{th})^{-1}$ as V_{GS} is reduced and saturates toward a maximum value of $2.3/S$, where S is the subthreshold slope. 47

3.2 Delay variation for an inverter driven by a slow input rise time. Variation in V_{thn} affects the starting point of the switching, hence, introduces delay variation. Inverter delay is shown for three cases: 1) Nominal case with no V_{thn} variation ($\Delta V_{thn} = 0$) and nominal propagation delay T_p, 2) case with positive V_{thn} shift ($\Delta V_{thn} > 0$) and increased propagation delay $T_p + \Delta T_p$ and 3) case with negative V_{thn} shift ($\Delta V_{thn} < 0$) and decreased propagation delay $T_p - \Delta T_p$. . 52

3.3 T_p Histogram for a single stage using Monte Carlo SPICE simulation (4000 runs): a) T_{pHL} at $V_{DD} = 0.6$ V ($T_r = 42.3$ ps), b) T_{pHL} at $V_{DD} = 1.2$ V ($T_r = 25.8$ ps), c) T_{pLH} at $V_{DD} = 0.6$ V ($T_f = 34.5$ ps) and d) T_{pLH} at $V_{DD} = 1.2$ V ($T_f = 21.7$ ps). Also shown: a Gaussian distribution having the same mean and standard deviation. T_r and T_f values correspond to the rise and fall times assuming the inverter is driven by another inverter of the same size. 56

3.4 T_p versus V_{DD} for a single stage showing the nominal High-to-Low (T_{pHL}), Low-to-High (T_{pLH}), and the average T_p. Also shown are $\mu_{T_{pHL}}$ and $\mu_{T_{pLH}}$ for both nominal rise/fall time and step input using Monte Carlo simulation. 57

3.5 Delay variation $\sigma_{T_{pHL}}$ versus T_r at $V_{DD} = 0.7$ V for FO4 inverter from Monte Carlo simulation. Also shown are the results from the proposed model Eq. (3.24) 58

3.6 Relative delay variation $\frac{\sigma_{T_{pHL}}}{T_{pHL}}$ versus T_r at $V_{DD} = 0.7$ V for FO4 inverter from Monte Carlo simulation. Also shown are the results from the proposed model. A minimum point for $\frac{\sigma_{T_{pHL}}}{T_{pHL}}$ is shown at T_r =90 ps. 59

3.7 Relative delay variation $\frac{\sigma_{T_{pHL}}}{T_{pHL}}$ versus T_r at $V_{DD} = 1.0$ V for FO4 inverter from Monte Carlo simulation. Also shown are the results from the proposed model. 60

3.8 $\sigma_{Tp_{HL}}$ versus T_r at $V_{DD} = 0.7$ V for different loading conditions (FO1, FO2 and FO4). 61

3.9 $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ versus T_r at $V_{DD} = 0.7$ V for different loading conditions (FO1, FO2 and FO4). 61

3.10 $\sigma_{Tp_{HL}}$ from our proposed models Eq. (3.24) versus Monte Carlo simulation results for different V_{DD} and loading (FO1, FO2 and FO4) conditions. 62

3.11 Impact of T_r variation on delay variation at $V_{DD} = 0.7$ V for FO4 inverter. Different values of $\frac{\sigma}{\mu}|_{T_r}$ are shown. 63

4.1 Yield versus performance tradeoff for SRAM design. 71

4.2 Simplified SRAM read path. 72

4.3 Timing diagram for SRAM read operation. 73

4.4 6T bitcell showing the different read current paths in the case of read 0 and read 1. 75

4.5 Current latch sense amplifier (CLSA). 76

4.6 SA input offset and read 0/1 distributions. 77

4.7 Modeling of SA input offset as a noise source connected at the input terminal of SA. 77

4.8 Timing delay variation between WL and SAEN paths. 79

4.9 Main sources of variation affecting access failures. 79

4.10 Impact of pass-gate leakage on the SA's input differential voltage (V_{SAin}). Pass-gate leakage from adjacent bitcells on the same column reduces the effective read current for the selected bitcell. 81

4.11 Typical SRAM architecture used in the proposed statistical yield estimation flow. 84

4.12 Flowchart of the proposed statistical yield estimation flow. 85

4.13 Yield estimation flow. 86

4.14 I_{read} histogram from Monte Carlo simulation (100k runs). 87

4.15 Characterization results for bitcell I_{read} variation using DC Monte Carlo simulation. 88

4.16 Characterization results for SA offset using transient Monte Carlo Analysis for different conditions. SA input offset follows a normal distribution as shown by the cdf (cumulative distribution function) of simulation and model. 89

4.17 Characterization results for $\sigma_{V_{SAoffset}}$ versus V_{DD} at different temperatures using Monte Carlo simulation. 90

4.18 Characterization results sensing window variation versus V_{DD} at different temperatures using Monte Carlo simulation. 90

4.19 Pass-gate leakage distribution. 91

4.20 Characterization results for $\mu_{I_{off,PG}}$ versus V_{DD} at different temperatures using Monte Carlo simulation. 91

4.21 Comparison between simulation results using the proposed yield estimation methodology and the measured access yield for a 1Mb memory in 45nm technology. 92

4.22 Comparison between the proposed statistical yield estimation methodology and worst-case analysis. 93

5.1 Memory read power and bitline differential versus T_{wl} for a 512kb memory in 65nm technology. 96

5.2 Typical SRAM architecture. 98

5.3 I_c probability density function (PDF) showing the points corresponding to 3,4 and $5\sigma_{I_c}$ (corresponding to different memory yield targets). $\frac{\sigma}{\mu}|_{I_c} = 15\%$ is assumed. 100

5.4 T_{wl} PDF. Also shown are the points corresponding to 3, 4 and $5\sigma_{I_c}$. Notice how the PDF is skewed towards higher T_{wl}. $\frac{\sigma}{\mu}|_{I_c} = 15\%$ is assumed. 101

5.5 Proposed architecture: Fine-grained wordline pulse width control. . 103

5.6 Programmable delay element. 104

5.7 Flowchart for the operation of the fine-grained wordline pulse width control system. 106

5.8 Power reduction using the proposed architecture versus chip level memory density. Different values of yield targets are shown. 108

5.9 Power reduction using the proposed architecture versus yield target for two cases of chip level density a)1Mb and b)48Mb. 109

5.10 Power reduction using the proposed architecture versus chip level memory density for different values of I_c variation for a yield target of 90%. 110

5.11 Power reduction using the proposed architecture versus chip level memory density for different values of the minimum controlled memory instance (or subbanks) for a yield target of 95%. 111

6.1 $\frac{\mu}{\sigma}$ SNM versus bitline precharge voltage. Lower bitline voltage during a read access improves bitcell read stability (SNM). 119

6.2 Selective Precharge operation. Step 1: Precharge to V_{DD} and GND. 120

6.3 Selective Precharge operation. Step 2: Charge sharing. 121

6.4 Selective Precharge operation. Step 3: Unselected columns disabled. 121

6.5 Selective precharge schematic. 122

6.6 Precharge to V_{DD} and GND circuits, including equalize transistors. 123

6.7 Selective precharge timing diagram. 123

6.8 Achieving larger range of ΔV_{BL} using by precharging on of the bitlines to GND. 124

6.9 Layout of the designed 512kb memory in 45nm technology. 126

6.10 Results for selective precharge read operation. MUX0/1 are the gates voltages for the PMOS devices in the column select for column 0 and 1, respectively. 127

6.11 Sense amplifier delay and input offset versus ΔV_{BL} after charge sharing. 129

6.12 Results for selective precharge write operation. $BL0/BLB0$ are accessed for write operation while $BL1/BLB1$ are half selected bitlines. 130

6.13 Full-swing CMOS write driver used to improve write margin. . . . 130

6.14 SNM simulation results for nominal devices without WID variations. 131

6.15 SNM simulation results using Monte Carlo simulation for 1000 MC runs. 131

6.16 SNM improvement using the proposed technique versus ΔV_{BL} after charge sharing. 133

6.17 V_{th} windows showing the improvement in read stability operating window for selective precharge (solid line) compared to the conventional approach (dotted line). Simulation accounts for 6σ of local variations. 133

6.18 Measured failure probability for the fabricated 512kb memory using the proposed technique (selective precharge) and compared to the conventional approach. 134

6.19 Chip micrograph for the fabricated 512kb memory in 45nm technology. Upper figure shows the location of the memory and the lower figure overlays the memory layout. 135

Chapter 1

Introduction

This chapter gives a short introduction on the importance of variation-tolerant design for nanometer regime. Section 1.1 presents the motivation for this research. Section 1.2 provides the outline of the thesis.

1.1 Motivation: Variation-Tolerant Design

Four decades of technology scaling in CMOS has been the largest driver for the electronics industry. Scaling of CMOS transistor has allowed having chips with more than one billion transistors in modern ICs and a wide range of products with very high levels of integration [1]. However, the aggressive scaling of CMOS technology in sub-90nm nodes has created huge design challenges. Due to process control limitations, manufacturing tolerances in process technology are not scaling at the same pace as transistor's channel length [2–6]. Moreover, variations due to fundamental physical limits are increasing significantly with technology scaling [2, 7, 8]. Due to all these sources, statistical parameter variations worsen with successive technology generations, and variability is currently one of the biggest challenges facing the semiconductor industry [5]. This variability has been affecting analog design for some time, and now it is dramatically impacting digital design at nanometer technology nodes.

Process variations strongly impact different aspects of digital circuit operation. For example, in random logic, the overdrive voltage ($V_{DD} - V_{th}$) becomes unpredictable even for neighboring identically-sized transistors. As a result, the gate delay becomes a stochastic random variable, which complicates timing closure techniques [2,9–11]. Moreover, traditional techniques that deal with inter-die variability

(such as slow/fast corner models and worst-case analysis) cannot be used in dealing with the large increase in intra-die variability. This is because these techniques tend to be inefficient and overly pessimistic in the presence of large variations. Therefore, statistical design methodologies instead of worst-case algorithms are required to deal with variations in nanoscale technologies [2, 9, 11].

Not only does variability affect random digital circuits, but it even has a much stronger impact on static random access memory (SRAM) [12–14]. With the exponential increase in embedded SRAM content in microprocessors and system on a chip (SoCs), SRAM yield has strong impact on the overall product yield (and of course cost) [5, 15]. In addition, SRAM uses the most aggressive design rules to achieve the highest possible integration density, which makes SRAM the most sensitive circuit for process variations. Due to the ubiquitous nature of embedded memories, SRAM yield loss due to variability can be the dominant cause of yield loss in modern ICs. Therefore, it is not surprising that the main focus of SRAM design in sub-90nm technologies is towards variation-tolerant techniques to reduce SRAM's sensitivity to variations and increase memory yield [12–14].

1.2 Thesis Outline

In order to continue digital design success in the nanometer regime, it is critical to explore variation-tolerant design solutions to mitigate the impact of process variability. This thesis focuses on dealing with the increase of variability in nanometer technologies and how it impacts digital CMOS circuits used in microprocessors and SoCs. This research intends to fill the gap between different levels of abstraction by introducing models and methodologies that can predict the impact of variations on digital circuits. In addition, circuit as well as architecture techniques that mitigate variability are also investigated, as shown in Fig. 1.1.

To set the stage for our discussion on variation-tolerant design, we begin in Chapter 2 by reviewing sources of variability and their impact on digital circuits. We also examine how SRAM operation is affected by variations since it is by far one of the most sensitive circuits for process variations.

To bridge the gap between technology and circuit design, in Chapter 3 we present a new design-oriented delay variation model that accounts for process variations. The model is based on analytical modeling of delay variability, and provides information on how process variations and circuit level design decisions interact to affect gate delay variation.

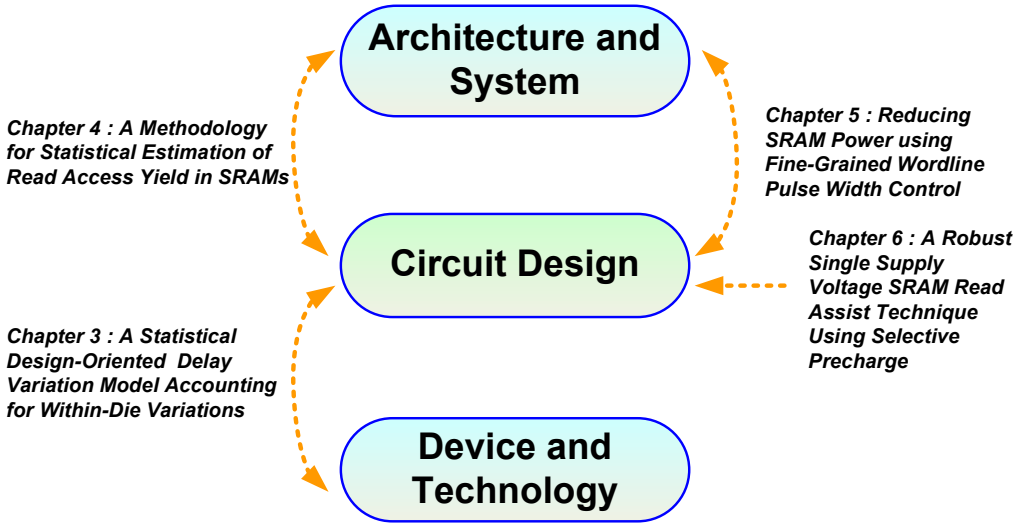


Figure 1.1: Different levels of abstraction studied in this research.

In Chapter 4, we examine how process variations can cause failures in SRAM read operation and present a new methodology for statistical estimation of read access yield. The proposed yield estimation flow provides yield and performance tradeoffs in the design time, which can be used to optimize the memory performance and architecture. Results from this methodology are verified with measured SRAM yield from 45nm technology.

In Chapter 5, we show how SRAM power consumption increases to ensure a correct read operation in the presence of large variations. We propose a new variation-tolerant architecture which reduces SRAM power consumption. The new architecture reduces SRAM switching power consumption by using fine-grained wordline pulse width control. The proposed solution combines memory built-in self test (BIST) with programmable delay elements in a closed-loop to reduce memory power consumption.

In Chapter 6, we focus on circuit techniques to mitigate SRAM failures which are deteriorated by process variations. We implement a new read assist technique, and show how this technique can improve bitcell stability without increasing the memory access time, and using only one supply voltage. A 512kb memory is designed in 45nm technology as a test vehicle to show the effectiveness of the proposed technique.

Summary of this work and suggestions for future work are discussed in Chapter 7.

Chapter 2

Background

This chapter presents a summary of relevant areas to the topic of variation-tolerant design. We start by discussing the different sources of variability in nanometer CMOS technology which affect the devices (Section 2.2), interconnect (Section 2.3), and environmental variations (Section 2.4). In Section 2.5, we discuss the impact of process variations on performance and power. In Section 2.6, we look at the different techniques that are used to deal with variability, beginning from analysis and CAD tools, moving to circuit and architecture techniques. Next we focus on SRAM scaling trends (Section 2.7) and how different types of SRAM failures are strongly affected by process variations (Section 2.8). In Section 2.9, we summarize this chapter.

2.1 Introduction

Variation is the deviation from intended values for structure or a parameter of concern. The electrical performance of modern IC are subject to different sources of variations that affect both the device (transistor) and the interconnects. For the purposes of circuit design, the sources of variation can broadly be categorized into two classes [4, 9, 16, 17]:

- **Die-to-Die (D2D)**: also called global or inter-die variations, are variations from die to die, and affect all devices on the same chip in the same way (*e.g.*, they may cause all the transistors' gate lengths' to be larger than a nominal value).

- **Within-Die (WID)**: also called local or intra-die variations, correspond to variability within a single chip, and may affect different devices differently on the same chip (*e.g.*, some devices on the same die may have larger channel length L than the rest of the devices).

D2D variations have been a longstanding design issue, and are typically dealt with using corner models [4, 9, 18]. These corners are chosen to account for the circuit behavior under worst-case variation and were considered efficient in older technologies where the major sources of variation were D2D variations.

However, in nanometer technologies, WID variations have become significant and can no longer be ignored [3,6,19–23]. As a result, process corners based design methodologies, where verification is performed at a small number of design corners, are currently insufficient.

WID variations can be subdivided into two classes [4, 9, 16, 17]:

- **Random variations**: as the name implies, are sources that show random behavior, and can be characterized using their statistical distribution.
- **Systematic variations**: show certain variational trends across a chip and are caused by physical phenomena during manufacturing such as distortions in lens and other elements of lithographic systems. Due to difficulties in modeling this type of variation, they are usually modeled as random variations with certain value of spatial correlation.

2.2 Sources of Variability: Device

Process variations impact device structure and therefore change the electrical properties of the circuit. In the following subsections, we review the main sources of variations that affect device performance.

2.2.1 Random Dopant Fluctuations (RDF)

As CMOS devices are scaled down, the number of dopant atoms in the depletion region decreases, especially for a minimum geometry device. Due to the discreteness of atoms, there is statistical random fluctuation of the number of dopants within a given volume around its average value [8, 24–26]. This fluctuation in the number of

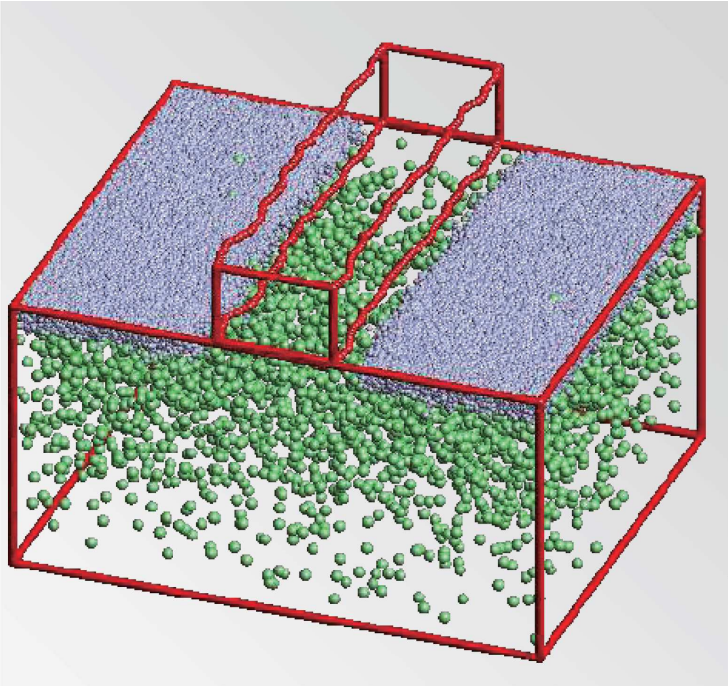


Figure 2.1: Atomistic process simulation incorporating random dopant fluctuation (RDF) and line edge roughness (LER) as the sources of intrinsic fluctuation [27]. The green dots show the dopant atoms that determine device's threshold voltage. Blue dots show the source/drain doping.

dopants in the transistor's channel results in variations in the observed threshold voltage V_{th} for the device. Fig. 2.1 shows how dopants are placed in the transistor's channel.

For example, in a uniformly doped $W = L = 0.1$ μm NMOS, if the doping concentration $N_a = 10^{18}$ cm^{-3} and depletion width at zero body bias $W_{dmo} = 350$ $\mathring{A}$, the average number of acceptor atoms in the depletion region can be calculated as $N = N_a.L.W_{dmo} = 350$ atoms. Due to the statistical nature of dopants, the actual number fluctuates from device to device with a standard deviation following a Poisson's distribution, and therefore $\sigma_N =< (\triangle N)^2 >^{1/2} = \sqrt{N}$, which for our example yields $\sigma_N = 18.7$, which is a significant fraction of the average number N (σ_N/N is 5% in this example). This has a direct impact on the threshold voltage of a MOSFET, since V_{th} depends on the charge of the ionized dopants in the depletion region [24].

These fluctuations were anticipated long ago [25, 28], but at that time, most FETs had sufficiently large number of dopants. Hence, these fluctuations were not causing problems for digital designers. However, they have always been important

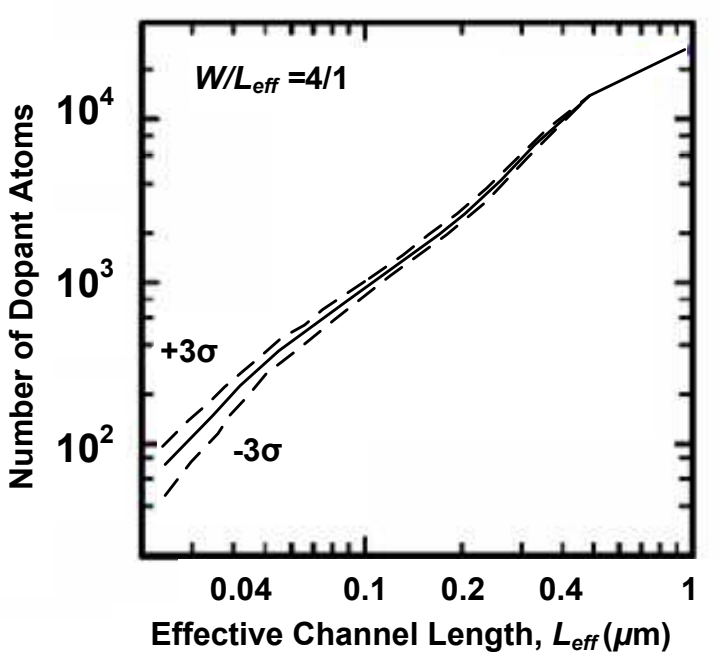


Figure 2.2: Number of dopant atoms in the depletion layer of a MOSFET versus channel length L_{eff} [7].

for analog circuits [28] (due to their sensitivity to mismatch) and SRAM bitcells [25]. Since then, however, the number of dopants in the depletion region of an FET has been decreasing steadily with scaling, as shown in Fig. 2.2. The decrease has been roughly proportional to $L^{1.5}$, so that we are now into the regime in which the smallest FETs have fewer than 1000 dopants determining the threshold voltage. Following Poisson statistics, fluctuations in the dopant number have a standard deviation equal to the square root of the number of dopants, hence the $\pm 3\sigma_N$ bounds shown in Fig. 2.2 become extremely large as channel length is scaled for sub-90nm technologies (with effective channel lengths approaching 40-60nm) [7].

The pioneering work of [25,26,28] showed that the variation in V_{th} due to random dopant follows a Gaussian distribution, and its standard deviation can be modeled as:

$$\sigma_{V_{th}} = (\sqrt[4]{2q^3 \varepsilon_{Si} N_a \phi_B}) \cdot \frac{T_{ox}}{\varepsilon_{ox}} \cdot \frac{1}{\sqrt{3\,W\,L}} \tag{2.1}$$

where q is the electron charge, ε_{Si} and ε_{ox} are the permittivity of the silicon and gate oxide, respectively, N_a is the channel dopant concentration, ϕ_B is the difference between Fermi level and intrinsic level, T_{ox} is the gate oxide thickness, and W and L are the channel width and channel length for the transistor, respectively.

Eq. (2.1) shows that $\sigma_{V_{th}}$ is inversely proportional to the square root of the active

device area. Hence, sizing up the transistors can be used to mitigate variations, which is one of the main techniques used in analog design to reduce mismatch between transistors [29]. Moreover, for SRAM devices, which typically have the minimum sizing, V_{th} variation will be the largest.

In addition, Eq. (2.1) shows that variation increases with technology scaling. Fig. 2.6 shows the large increase in $\sigma_{V_{th}}$ with technology scaling [1], and can reach about 50% of V_{th} in advanced technologies, which causes a large spread in performance and power. This is why random dopant fluctuation have been gaining large attention recently.

2.2.2 Channel Length Variation

The patterning of features smaller than the wavelength of light used in optical lithography results in distortions due to the diffraction of light, which is usually referred as optical proximity effects (OPE) [4,30]. These OPEs cause large variations in defining the minimum feature sizes. Fig. 2.3 shows that nanometer technologies are using light sources with wavelengths which are much larger than the minimum feature size [6], especially for 90nm to 32nm technologies. This makes lithography at these ranges extremely challenging. OPEs are layout dependent, therefore result in different critical dimension (CD) variations depending on neighboring lines as well as orientation [17].

Controlling these variations has become extremely difficult in current technologies, and are expected to increase in future technology nodes, until there is radical change in lithography technology (*e.g.*, EUV lithography). This is the main reason why there is large variation in the minimum feature sizes in current technologies. In addition, since these variations are layout dependent, they cause systematic type of variation and are usually treated as spatially correlated intra-die variations [9, 31].

The variation in transistor's channel length has direct a impact on several electrical properties of a transistor, however, the most affected parameters are the transistor's drive current ($I_D \, \alpha \, 1/L$) and V_{th} [18, 24]. The variation in V_{th} arises due to the exponential dependence of V_{th} on channel length L for short channel devices, mainly due to drain induced barrier lowering (DIBL) effect [18, 24]. DIBL causes V_{th} to be strongly dependent on the channel length L as shown in Fig. 2.4. V_{th} reduction due to DIBL can be modeled as [18, 24]:

$$V_{th} \approx V_{th0} - (\zeta + \eta \, V_{DS}) e^{-L/\lambda} \tag{2.2}$$

[1] LER effect shown on the figure will be explained later.

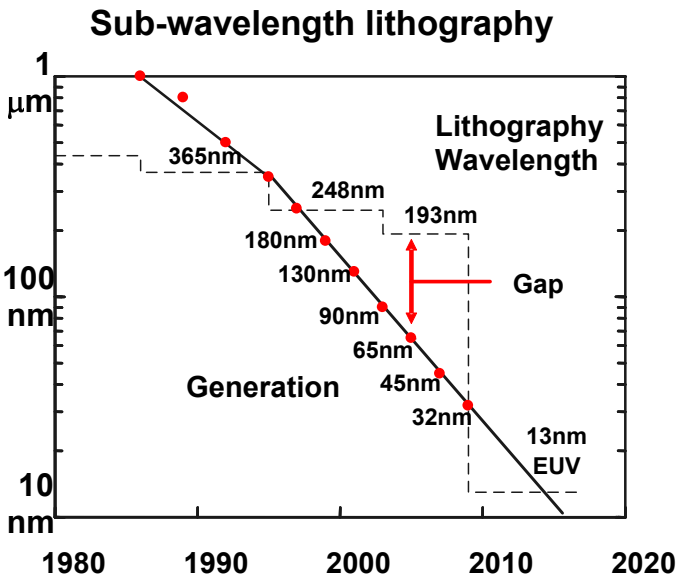


Figure 2.3: Lithography wavelength scaling for different technology nodes [6].

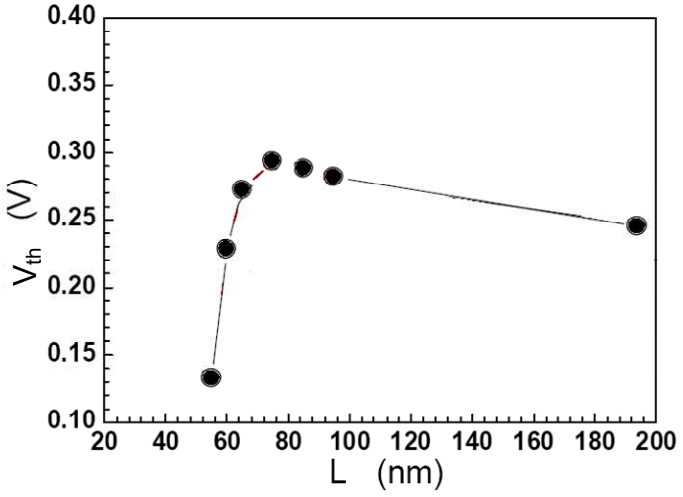


Figure 2.4: Measured V_{th} versus channel length L for a 90nm [32] which shows strong short channel effects causing sharp roll-off for V_{th} for shorter L.

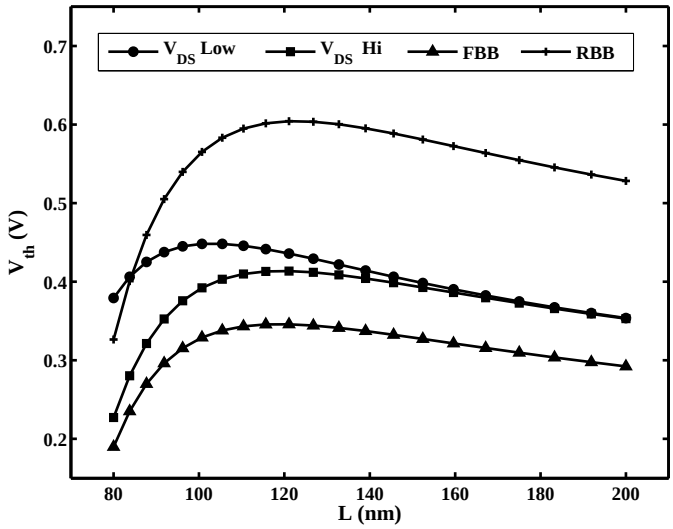


Figure 2.5: Simulated V_{th} versus channel length L for IBM 130nm technology showing V_{th} versus L roll-off under different conditions, low and high V_{DS}, and forward (FBB) and reverse body bias (RBB).

where η is the DIBL effect coefficient, and V_{th0} is the long channel threshold voltage. Therefore, a slight variation in channel length will introduce large variation in V_{th}, as shown in Fig. 2.4.

This type of variation strongly depends on the applied drain to source voltage V_{DS}, and also body bias V_{BS} as shown in Fig 2.5. This is because DIBL has strong dependence on both V_{DS} and V_{BS} voltages [18, 24]. The roll-off increases as V_{DS} increases. Moreover, as shown in the figure, V_{th} roll-off reduces when forward biasing the body (*i.e.*, V_{BS} positive for NMOS), and vice versa for reverse body biasing. Therefore, the impact of L variation on V_{th} reduces when applying forward body bias (FBB) [18, 24].

2.2.3 Other Sources

While random dopant fluctuation and channel length variations are the dominant sources of device variations nowadays, there are many other sources which may become significant in the future technologies. Below we list other sources of device variations:

- **Line Edge Roughness (LER)**: Gate patterning introduces a nonideal gate edge which exhibits a certain level of roughness referred to as line edge rough-

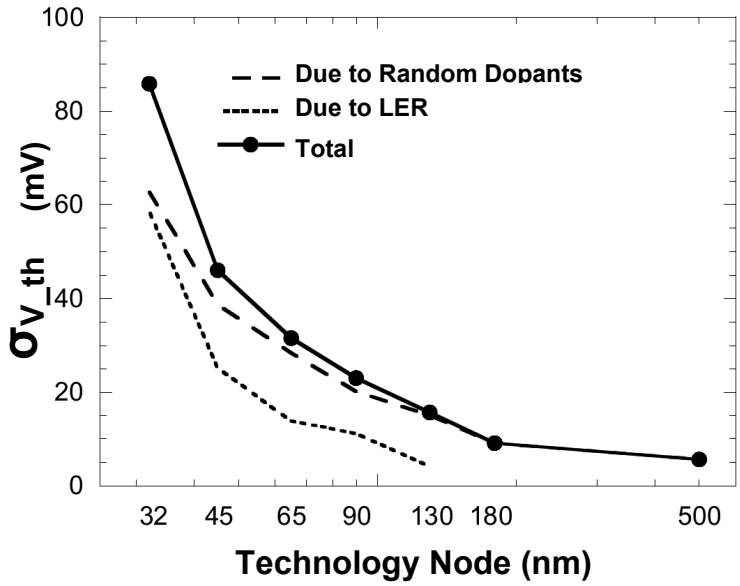


Figure 2.6: Predicted $\sigma_{V_{th}}$ including RDF and LER versus technology nodes for the smallest transistor with an area $3\,L^2$. The inset shows the technological parameters used [33].

ness, as shown in Fig. 2.1. This effect was neglected previously since the LER effect was much smaller than the CD variation. As device scaling continues into sub-50nm regime, LER is expected to become a significant source of variation due to its direct impact on $\sigma_{V_{th}}$, as shown in Fig. 2.6 [1, 8].

- **Oxide Charges Variation**: Interface charges can also cause V_{th} variation. However, its effect is not dominant in modern-day nitrided gate oxides [8]. Nevertheless, future adoption of high-k gates to reduce gate-tunneling leakage current will probably worsen oxide charge variations [8]. In addition, oxide charge variations can introduce mobility fluctuations, as it affects scattering mechanisms in a transistor's channel.

- **Mobility Fluctuation**: Variations in a transistor's drive current can also be caused by mobility fluctuation. Mobility fluctuation can arise from several complex physical mechanisms such as fluctuations in effective fields, fixed oxide charges, doping, inversion layer, and surface roughness [8]. Moreover, due to its dependence on many physical variation mechanisms, mobility variation can also be correlated with V_{th} variations. However, device measurements show this correlation is small [34]. Therefore, mobility variations and V_{th} variations are typically assumed to be independent in circuit modeling [34].

- **Gate Oxide Thickness Variation**: Any variation in oxide thickness affects many electrical parameters of the device, especially V_{th}. However, oxide thickness is one of the most well-controlled parameters in MOSFET processing. Therefore, it may not affect V_{th} variation significantly.

- **Channel Width Variation**: Due to lithography limitations, transistor channel width also varies. This variation in width causes V_{th} to change, due to the narrow-width effects, which causes V_{th} to be a function of channel width W [18]. However, since W is typically 3-4 times larger than L, the impact of W variation on V_{th} is considered to be much smaller than the impact due to L variation [18].

It is important to note that there are other sources of time dependent device variation which include device degradation due to aging effects such as hot carrier effect [18, 24] and negative bias temperature instability (NBTI) [35].

2.3 Sources of Variability: Interconnect

In addition to sources of variations that alter device characteristics, there are several sources that affect interconnects. The mains sources of variations in interconnects include [16]:

1. **Line Width and Line Space**: Deviations in the width of patterned lines arise primarily due to photolithography and etch dependencies. At the smallest dimensions, which typically occur at lower metal levels, proximity and lithographic effects may be important. However, at higher metal levels, aspect ratio dependent etching, which depend on line width and local layout, can be significant. Variations in line width directly impact line resistance as well as line capacitance [16, 17].

2. **Metal and Dielectric Thicknesses**: In a conventional metal interconnect, the thickness of metal films is usually well controlled, but can vary from wafer-to-wafer and across the wafer. However, in advanced damascene copper interconnect processes, this is not the case. Unlike older aluminum interconnect processes where the metal is patterned and the oxide is polished, the oxide is patterned and the metal is polished in a damascene process for copper interconnects. Following that, chemical mechanical polishing (CMP) is used to achieve flat topography on the wafer. However, copper (as well

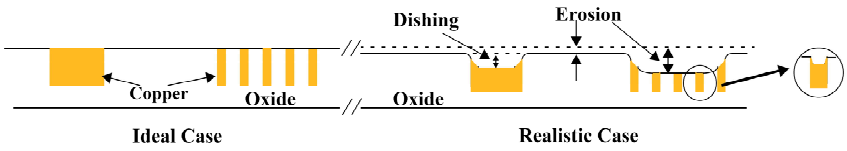


Figure 2.7: A schematic cross-section of interconnect showing dishing and erosion impact on metal height.

as adjacent dielectric) are removed from the wafer at different rates during CMP, depending on the surrounding layout such as pattern geometry (dense versus isolated). This creates surface anomalies and varying metal thickness. Dishing and erosion are the two most common surface anomalies that come with copper CMP. Dishing occurs when the copper recedes below the level of adjacent dielectric. Erosion is a localized thinning of the dielectric, which normally happens when CMP is applied to an array of dense lines as shown in Fig. 2.7. The oxide between wires in a dense array tend to be over-polished compared to the nearby areas of wider insulators. Both dishing and oxide erosion are problematic in wide lines and dense arrays, respectively, and are therefore layout dependent. They lead to higher resistances and more surface non-uniformity. In damascene processes with copper interconnects, dishing and erosion can significantly impact the final thickness of patterned lines, with line thickness losses of $10-20\%$ [16, 17].

3. **Contact and Via Size**: Contact and via sizes can be affected by etch process variations, as well as systematic layer thickness dependencies. Depending on the via or contact location, the etch depth may be substantially different, resulting in different degrees of lateral opening. Such size differences can directly change the resistance of the via or contact [16].

2.4 Sources of Variability: Environmental

In addition to process variations which are static in nature, there are also environmental factors that arise during the operation of a circuit and are typically dynamic. These include variations in power supply and temperature of the chip or across the chip [6, 9, 20].

Variation in switching activity across the die result in uneven power dissipation across the die. This variation results in uneven supply voltage distribution which

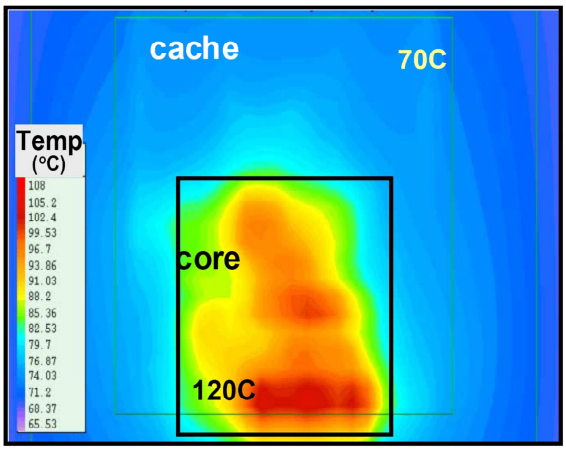


Figure 2.8: Thermal image showing within die temperature variation for a microprocessor [19]. Hot spots with temperatures as high as 120 °C are shown.

affects device performance as well as power dissipation [19]. A reduced power supply lowers drive strengths and degrades speed.

Within die temperature fluctuations have always been a major performance and packagings challenge, especially for high-performance processors. This is because both device and interconnect have temperature dependence, causing performance to degrade at higher temperatures. Moreover, temperature variation across communicating blocks on the same die can cause performance mismatches, which may lead to functional failures [19]. Fig. 2.8 shows WID temperature variation for a microprocessor, with hot spots in the core reaching 120 °C.

Leakage currents have a strong dependence on temperature (especially subthreshold leakage), therefore leakage power increases at higher temperatures [17]. In the meantime, higher leakage power will cause die temperature to increase. This type of positive feedback may cause thermal runaway where leakage currents, and temperature continue to increase until failure [17].

Both supply and temperature variations depend on the work-load of the processor and are, hence, time-dependent. However, identifying worst-case conditions for temperature and supply is very difficult [17]. Therefore, designers often focus on minimizing temperature and supply variations as much as possible; for example, ensuring that the voltage drop on the power grid is always less than 10% of the nominal supply voltage, and adding large decoupling capacitors [6, 17].

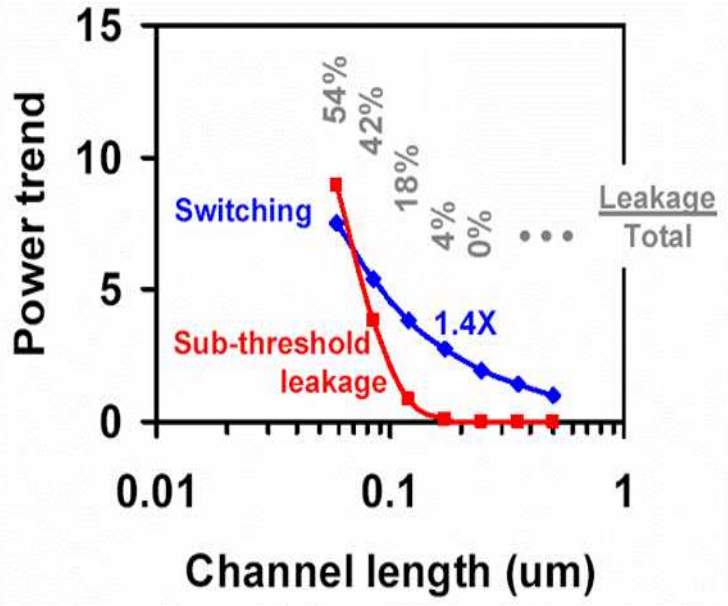


Figure 2.9: Dynamic and static power versus technology scaling, showing the exponential increase in leakage power [38].

2.5 Impact of Process Variations on Performance and Power

In nanometer devices there are several sources of leakage current, such as subthreshold, gate oxide tunneling, junction band-to-band tunneling (BTBT) and gate-induced drain leakage (GIDL), all of which increase with technology scaling [7, 36, 37]. That is why for designs in sub-90nm, leakage power consumption is considered a significant part of the total power. It is expected that leakage power can reach more than 50% in 65nm technology as shown in Fig. 2.9.

The large variability in advanced CMOS technologies is playing an increasing role in determining the total leakage of a chip [39, 40]. This is because leakage currents have strong dependence on process variations. For example, variation in V_{th} introduces large spread in subthreshold leakage due to the exponential dependence on V_{th}. Similarly, gate-tunneling leakage current is very sensitive to oxide variation. This has accentuated the need to account for statistical leakage variations during the design cycle [39–41].

Fig. 2.10 shows measured I_{on} and I_{off} scatter plot for a 150nm technology [42]. Even in that mature technology, there is an excessively large spread in I_{off} (100X) as compared to the 2X spread in I_{on}.

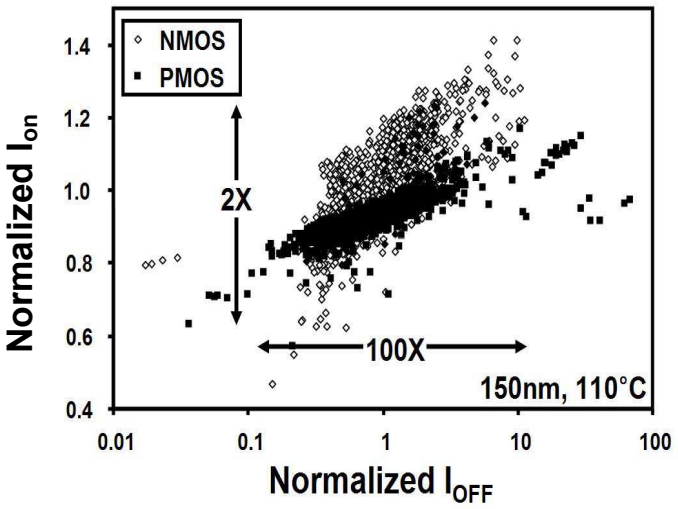


Figure 2.10: Measured I_{on} versus I_{off} scatter plot showing large spread in I_{off} for a 0.15μm technology [42].

For a whole chip, this can cause large variations in leakage power. Fig. 2.11 shows measured variations for frequency and leakage power for a 130nm technology [21]. The figure shows that there is about 5X variation in leakage for a 30% variation in chip frequency. The highest frequency chips have a wide distribution of leakage, and for a given leakage there is wide distribution in the frequency of the chip. These are considered very large variations in leakage power, especially in the time where leakage power is increasing exponentially with each technology node. This excessively large spread in leakage current makes it very difficult to achieve the required speed while meeting the power constraints.

It has been shown in [19] that among the chips that meet the required operating frequency, a large fraction dissipate a large amount of leakage power, which makes them unsuitable for usage, and thus degrade yield. This is due to the inverse correlation between leakage current and circuit delay. In that mature technology, it can be assumed that the channel length variation is the dominant source of process variations. For devices with smaller channel length than nominal, V_{th} decreases due to DIBL, and therefore, the subthreshold leakage current increases exponentially. In the meantime the circuit delay decreases due to the increase in I_{on}, since the overdrive voltage $V_{DD} - V_{th}$ increased. Hence, these chips have higher operating frequency, but suffer from large leakage power which makes them unacceptable. It is important to note that for the high frequency chips shown in Fig. 2.11, both the mean and standard deviation of leakage current increases considerably.

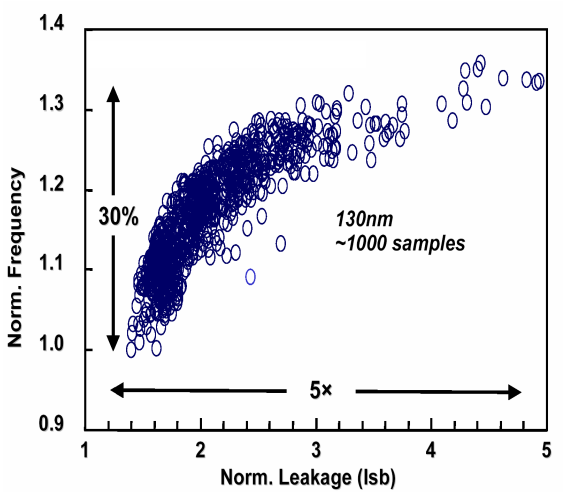


Figure 2.11: Leakage and frequency variations for an Intel processor in 0.13μm technology [21].

This trend causes yield to decrease substantially [19]. Therefore, there is a crucial need to account for leakage power and its dependence on process variations when analyzing the impact of variability on design techniques [43]. Moreover, variation-tolerant circuit techniques that can reduce variability, and hence reduce leakage power variation, are of great importance to improve the yield in advanced CMOS technologies [6, 19, 21].

2.6 Techniques to Deal with Variability

In this section, we review state of the art research work dealing with the increase in variability in nanometer technologies. We begin by presenting work in analysis and CAD followed by circuits and finally architecture.

2.6.1 Analysis and CAD

Recently, a large number of research work has been done in the area of CAD tools that are "Variation-Aware." One of the most researched topics in this area is statistical static timing analysis (SSTA) as compared to the well known static timing analysis (STA) tools [9, 17, 44, 45].

The goal of timing verification is to ensure a chip will operate at a frequency or a range of frequencies with a desired yield under the specified range of operation

conditions. In timing verification, speed (set-up time) and functional (hold-time) are usually checked to verify that the design will meet the maximum frequency target, as well as correct functionality, respectively [16]. STA propagates arrival times through the circuit, and as the arrival times traverse gates, the delay of the gate is added to the arrival time and a maximum arrival time is selected when multiple arrival times converge at a gate [16].

STA has been used in performance verification for the past two decades. Traditionally, process variations have been addressed in STA using corner-based analysis, where all the gates are assumed to operate at worst, typical, or best-case conditions [9]. This technique is very efficient when dealing with D2D (inter-die) variation. However, since WID (intra-die) variation has become a substantial portion of the overall variability, the corner-based STA can suffer from significant pessimism and inaccuracy, which have given rise to SSTA [9].

In SSTA, the circuit delay is considered a random variable and SSTA computes the probability density function (pdf) of the delay at a certain path [9]. The arrival times also become random variables, and therefore, the addition and maximum operations of STA are replaced by convolution and statistical maximum, respectively [9]. Much of the work on SSTA, however, has been in the area of finding efficient algorithms to perform addition and maximum operations [9].

While SSTA is more appropriate in dealing with WID variations, and can give accurate results without going through the lengthy Monte Carlo simulations, it is still considered a newly emerging tool. Even after computing the delays statistically, there are no clear methods to how to optimize these delays distributions from a design perspective. Unfortunately, there is a disconnect between CAD developers and designers in this area, where CAD developers seem to be working on tools that simply propagate delay pdf's, and designers do not know what to do with these pdf's [10].

2.6.2 Circuits

In the area of circuit techniques to mitigate variability, some work has been done in the last few years. A common idea in that work was to measure variability and use a certain type of feedback to mitigate it. These techniques use control knobs such as supply voltage, body bias, or programmable sizing to achieve this control.

A speed adaptive body bias technique was utilized in [46, 47] to compensate for variability. The scheme was implemented in a 1.2 GIPS/W microprocessor

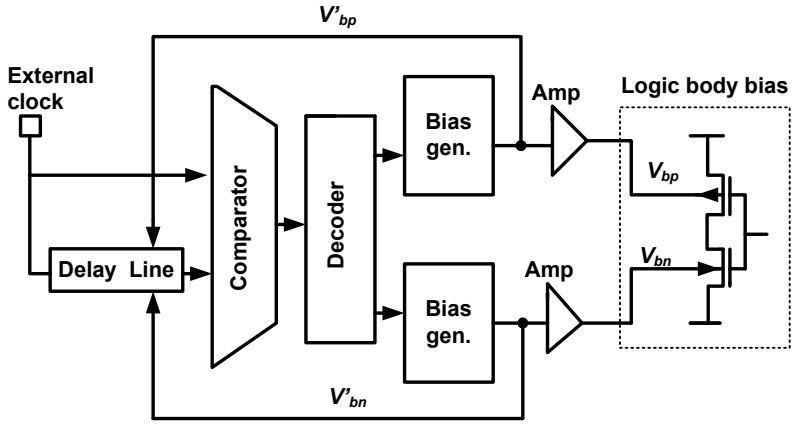


Figure 2.12: Block diagram of speed adaptive-V_{th} technique [47].

running at 220 MHz in 0.2 μm technology. The speed adaptive-V_{th} is composed of a delay line, a delay comparator, a decoder and body bias generators, as shown in Fig. 2.12. The comparator measures the delay between an external clock signal and an output signal from the delay line and then converts the amount of delay into a register address in the decoder. The generators supply V'_{bp} and V'_{bn} for PMOS and NMOS bodies, respectively, to keep the delay line's delay constant by changing the V_{thp} and V_{thn}, respectively. If the speed of the delay line changes due to variations, the comparator output changes, and hence the generated body bias is modified. The junction leakage and gate-induced drain leakage (GIDL) current determine the reverse-bias bound, which was set to 1.5V, while the 0.5V forward biased was defined by the subthreshold leakage. In addition, forward body bias has the desirable result of improving short-channel effects of a transistor, and thus reduces the sensitivity to channel length variation.

This technique is very efficient in dealing with D2D variations, however, it cannot be used to mitigate WID variations effectively. The reason is because this technique supplies the same body bias to the entire chip, while WID variations will affect different parts of the chip in different ways.

A similar technique was presented in [48], where again forward body bias and reverse body bias were used to improve performance and decrease leakage, respectively. This adaptive body bias (ABB) allows each die on a wafer to have the optimum threshold voltage which maximizes the die frequency subject to power constraint. A critical path emulator containing key circuit elements of a process critical path are used to model the effect of body bias on the frequency and leakage of a real processor. The technique was tested in a 0.15μm technology.

The authors in [48] used multiple delay sensors which are distributed on the die to be able to get an average body bias that accounts for WID variations. With no body bias used, only 50% of the dies are acceptable, mainly in the lowest frequency bin. When ABB was utilized using only one delay sensor, ABB reduced the frequency variation σ/μ from 4% to 1%, however a significant number of dies failed to meet leakage constraint. Using multiple sensors on die for ABB, the frequency variation reduced to 0.69% and all dies met the leakage constraint with 32% in the highest frequency bin.

While the ABB scheme with several sensors considers WID variations in determining the optimum bias combination per die, it is still not possible to completely compensate for these variations using only a single bias combination per die. Therefore, in [48], the authors proposed the WID-ABB technique, which allows each large circuit block in the design to have its own unique body bias combination which controls the frequency and leakage of that circuit block. In that case, the NMOS implementation requires a triple-well process, which adds additional cost to the process. WID-ABB enabled 99% of the dies to be accepted in the highest frequency bin. Fig. 2.13 shows the results after using WID-ABB technique.

In [49], the authors extended the ABB technique and combined it with adaptive supply voltage V_{DD} to control the frequency and leakage distribution of processors. It was shown that using adaptive V_{DD} in conjunction with ABB is more effective than using either of them. Once again, ABB uses forward body bias (FBB) to speed up dies that are too slow and reverse body bias (RBB) to reduce frequency and leakage power of dies that are too fast and leaky. Adaptive V_{DD} + ABB, on the other hand, recovers the dies that exceed the power limit by first lowering V_{DD}, and hence the operating frequency, thus bringing the total switching and leakage power below the power limit and then applying FBB to speed up and move them to the highest frequency bin allowed by the power limit. Using adaptive V_{DD} combined with WID-ABB, the number of dies accepted in the highest two frequency bins increased from 26% to 80%. However, this improvement comes at the cost of additional area, design complexity and cost.

Recently, a new technique to reduce random fluctuations was presented in [50]. In this technique, a forward body bias is applied to the logic circuit blocks, using a body bias generation circuit shown in Fig. 2.14. A current source is used to determine the substrate potential by forward biasing the junction diode. The current source limits the maximum currents that the forward diodes can conduct, and the body potential is self adjusted by the diode current.

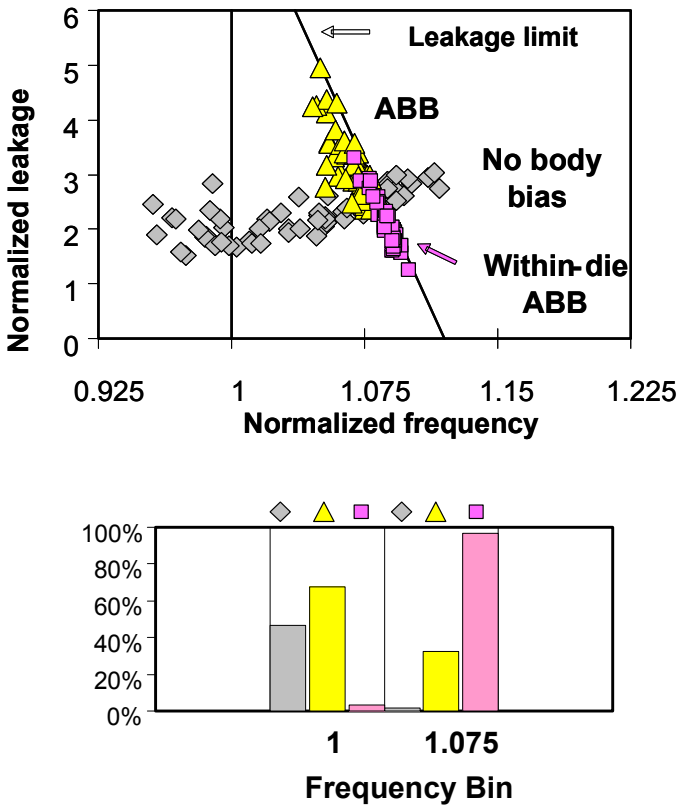


Figure 2.13: Measured leakage versus frequency distribution for 62 dies in a 150nm technology (V_{DD} = 1.1V, T=110C), showing the distributions after utilizing ABB and WID-ABB. In the lower figure, the percentage of accepted dies at a certain frequency bin are shown [48].

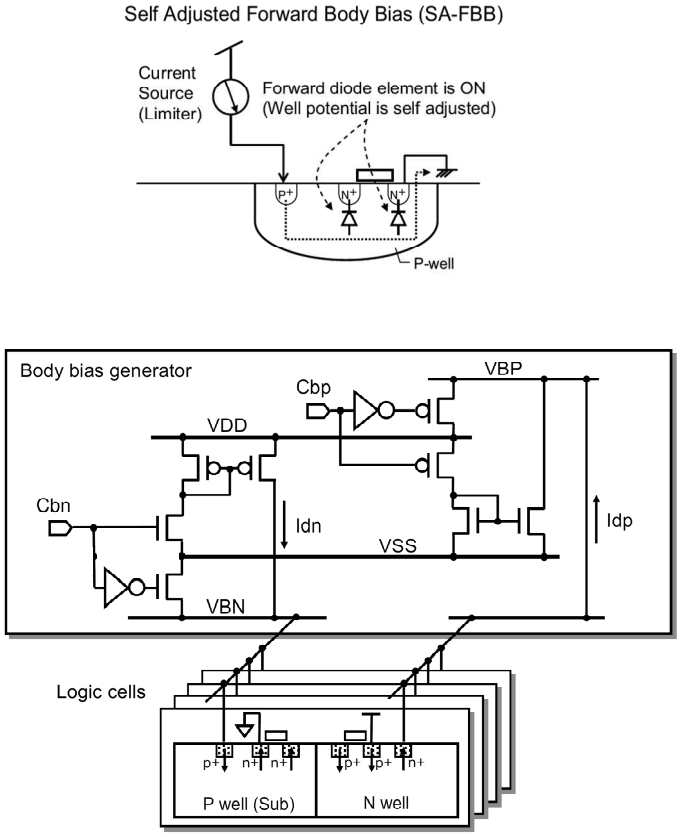
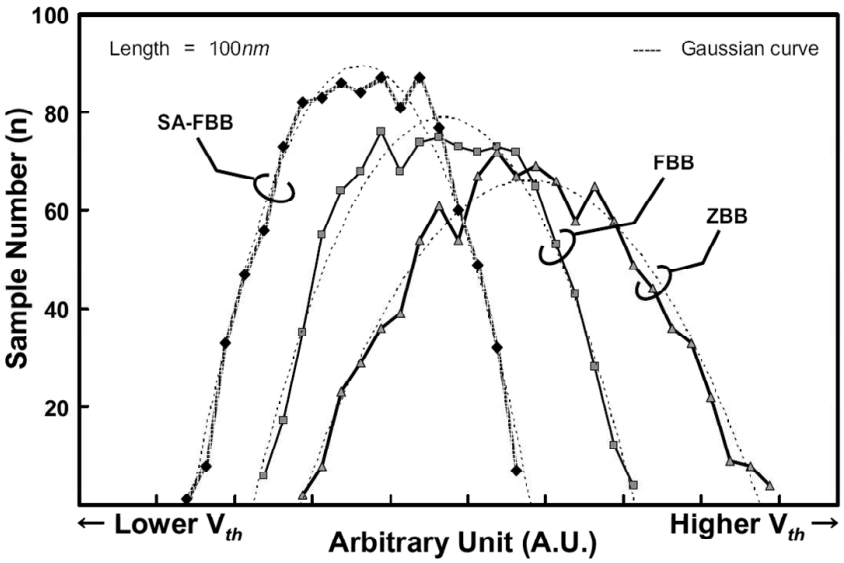


Figure 2.14: Schematic showing the implementation of self-adjusted forward body bias (SA-FBB) circuit technique [50].



	$\sigma(I_{ds})$ PMOS (%)	$\sigma(I_{ds})$ NMOS (%)	$\sigma(V_{th})$ PMOS (%)	$\sigma(V_{th})$ NMOS (%)
ZBB	1	1	1	1
FBB	0.929 (7%)	0.945 (5.4%)	0.950 (5%)	0.957 (4.3%)
SA-FBB	0.421 (57.9%)	0.767 (23.2 %)	0.650 (35%)	0.783 (21.7%)

Figure 2.15: Comparison between ZBB, FBB and SA-FBB showing measured V_{th} distribution in a 90nm technology [50].

Under this self-adjusted forward body bias (SA-FBB) condition, $\sigma_{V_{th}}$ decreases as shown in Fig. 2.15, and 35% reduction in $\sigma_{V_{th}}$ was achieved compared with the zero body bias (ZBB) case. Another interesting result in this technique, is that the improvement achieved in using SA-FBB was larger than the improvement using conventional FBB technique. This may be due to the fact that SA-FBB enables the body bias to reach a higher value (become more forward biased) compared to conventional FBB. This is because the body voltage is set by the diode current current used. However, in conventional FBB, the maximum forward body bias is defined by stability requirements for the substrate and preventing latch-up. The distributions shown in Fig. 2.15 also justifys this reasoning, since the mean of V_{th} is not equal for FBB and SA-FBB which means the body bias voltage for both cases is not the same.

Dynamic circuits are usually used for high-performance gates such as high-speed register files in multi-GHz operation [51]. Keepers are used to prevent the

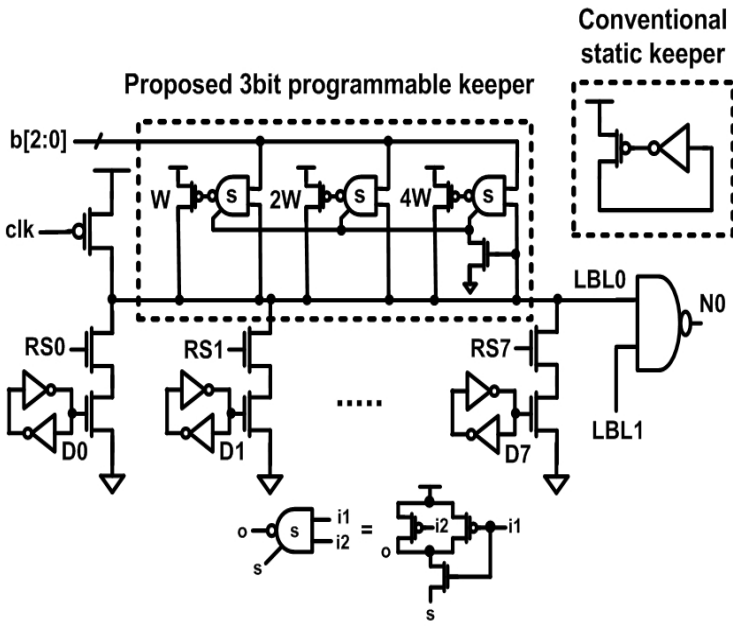


Figure 2.16: Register file with programmable keeper to compensate for process variations impact on leakage [51].

dynamic node from floating and hold it to V_{DD} when none of the pull-downs are evaluated [52]. In previous technologies, a small keeper was sufficient to hold the dynamic node to high. As technology scales, a stronger keeper is required to prevent the domino node from collapsing under increasing pull-down leakage levels. In addition, due to the increase in leakage variations, the keeper should be sized up further, to account for worst-case leakage. However, increasing the keeper size slows down the speed of dynamic circuits and limits its advantage over static CMOS [42].

In [42, 51], a process variation compensation technique for dynamic circuits is proposed, using programmable sizing, as shown in Fig 2.16. The circuit shows a 5X reduction in number of failing dies compared to conventional designs. An on-chip leakage sensing circuit is used to measure leakage current, and is used to select the optimal keeper width, which is programmed via fuses.

2.6.3 Architecture

One of the first pieces of work that related variability to architecture was the work by Bowman *et al* [23, 53, 54], and presented a statistical predictive model for the distribution of the maximum operating frequency ($FMAX$) for a chip in the presence of process variations. This technique provides insight on the impact of

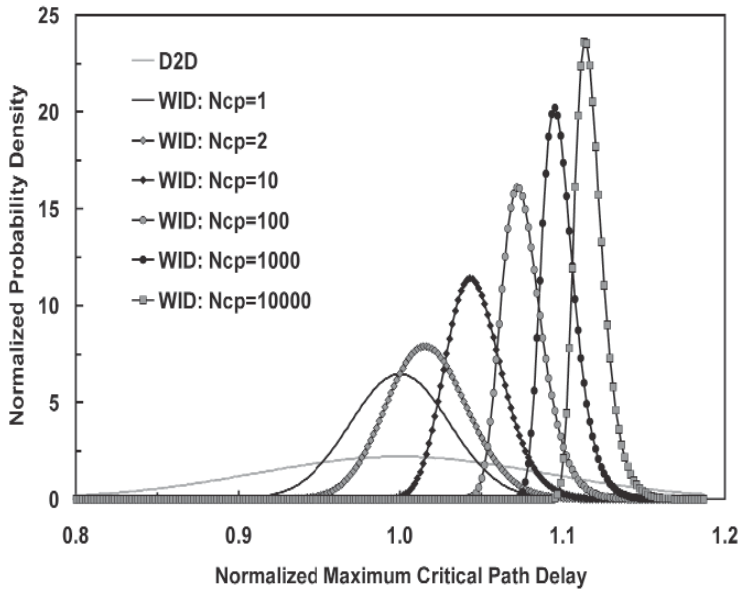


Figure 2.17: The WID maximum critical path delay distribution for different values of independent critical paths N_{cp}. As N_{cp} increases, the mean of maximum critical path delay increases [53].

different components of variations on the distribution of $FMAX$. The WID delay distribution heavily depends on the total number of independent critical paths for the entire chip N_{cp}. For a larger number of critical paths, the mean value of the maximum critical path delay increases as shown in Fig. 2.17. As the number of critical paths increases, the probability that one of them will be strongly affected by process variations is higher, and hence, increases the mean of critical path delay. On the other hand, the standard deviation (or delay spread) decreases with larger N_{cp}, thus making the spread of the overall critical path determined mainly by D2D variations. The results showed that WID variations directly impact the mean of the maximum frequency, while D2D fluctuations impact the variance.

Another factor that affects the delay distribution is the logic depth per critical path. The impact of logic depth on delay distribution is different when dealing with random or systematic WID variations. Random WID variations have an averaging effect on the overall critical path distribution, while systematic WID variations affect all the gates on the path, hence, increase delay spread.

Towards analyzing the impact of variability on architecture decisions, the authors in [55,56] extend the Bowman's model by assuming that the number of critical paths per stage is proportional to stage's device count. In addition, they introduce

metrics and models to evaluate variability in the architectural domain. This is considered one of the first works that accounts for variability at that high level of abstraction.

Other variation-tolerant research at the architectural level was presented in [57], where a statistical methodology for pipeline delay analysis was presented. The importance of logic depth in variability studies was emphasized, and it was shown that the change in logic depth and imbalance between stage delays can improve the yield of a pipeline. Techniques such as deep pipelining and the push for high clock speeds decreases logic depth and have an undesirable impact on design variability [57].

In [58], variations impact on low-power parallel system was investigated. A generic parallel system consisting of inverter chains was used to model critical paths in a microprocessor. It was shown that neglecting WID variation would underestimate the optimum supply voltage that minimizes power consumption for the parallel system. Moreover, a parallel system which was optimized neglecting WID variations, will not provide the anticipated power savings since it would consume almost similar power to the original system (without parallelism). It was shown that the number of blocks needed in parallel to achieve certain throughput increases significantly when WID process variations are considered. As a consequence, the optimum supply voltage that provides the lowest power becomes higher, and therefore, the targeted power reduction using parallelism decreases.

Recently, a study on the impact of parameter variations on multi-core chips was presented [59]. In that study, the authors argue that WID variation will be more important for core-to-core granularity, rather than at unit-to-unit granularity. In addition, they show that WID systematic process variations will result in a major leakage variation across multiple cores on a single chip [59, 60]. Therefore, core-to-core leakage can differ by as much as 45% in a 45nm technology.

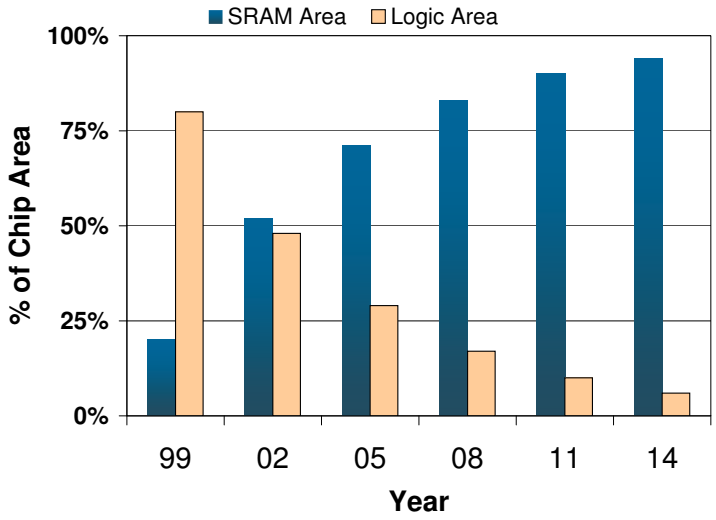


Figure 2.18: SRAM and logic area versus technology scaling. SRAM dominates chip area in modern SoCs and microprocessors [15].

2.7 SRAM Scaling Trends

In today's SoC, embedded SRAM dominates the chip area as shown in Fig. 2.18. It is expected that SRAM area will exceed 90% of overall chip area by 2014 [15]. This is driven by the demand of higher performance (multiprocessing and multicores), lower power and higher integration.

To increase memory density, memory bitcells are pushed to the achieve 50% of scaling each technology node as shown in Fig. 2.19. This requires very aggressive design rules which makes SRAM more vulnerable for variations [61, 62]. For example, in state of the art 45nm technology, an ultra high density bitcell area is approximately $0.25\mu m^2$ as shown in Fig. 2.20. This extremely compact bitcell enables an integration of 152Mbit/cm^2 [62].

While process variation affects performance and leakage of random logic, its impact on SRAM is much stronger. In advanced CMOS technology nodes, the predominant yield loss comes from the increase of process variations which strongly impacts SRAM functionality as the supply voltage is reduced [12, 13, 63–65]. In particular, WID variations due to RDF and LER strongly impact SRAM operation. Fig. 2.21 shows that V_{th} variation for SRAM devices increases significantly with scaling, which pose a major challenge for SRAM design [5].

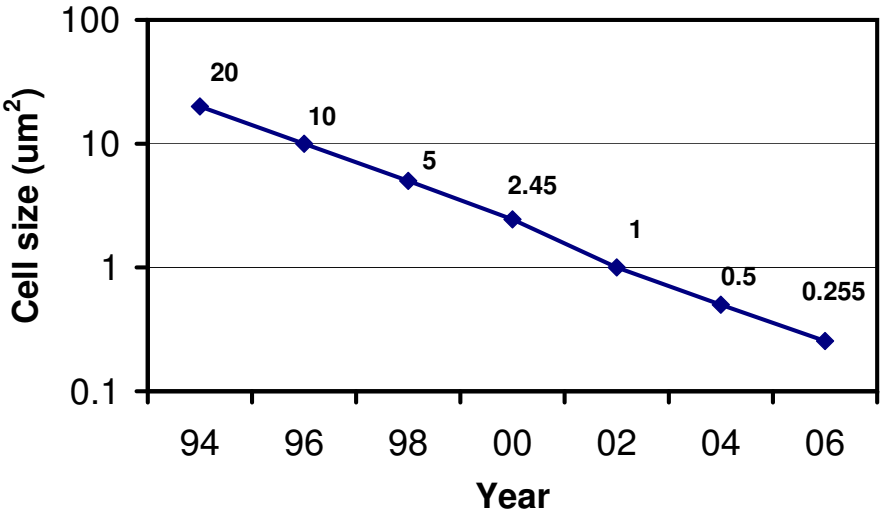


Figure 2.19: SRAM bitcell area scaling from 350nm down to 45nm technology nodes [62].

2.8 Variability and SRAM Failure Mechanisms

Due to its small size and high density, there are many sources that cause SRAM to fail. Fig. 2.22 shows different SRAM failures such as hard fails, stability fails and radiation induced soft errors.

2.8.1 Hard and Catastrophic fails

Catastrophic fails due to physical defects can cause permanent damage for memory and digital circuits. Physical defects cover a wide range of possible defects such as voids, shorts, metal bridges, missing contacts or vias, oxide pin holes and many others [16]. Collectively, these fails are called hard fails. Because memories are designed with aggressive design rules, they tend to be more sensitive to manufacturing defects and reliability problems compared to any other cores on the chip [15]. Fig. 2.23 shows that hard fails reduce with process technology due to lower defect density, while soft fails due to intrinsic variation show an opposite trend [61].

Since a large area of modern chips is consumed by memories, they can adversely affect chip yield. To improve yield, SRAM uses redundant elements such as redundant rows, columns or banks which can be used to replace defective elements, hence, significantly improve yield [15,16,67]. Historically, this type of repair capability was implemented to address hard fails. However, nowadays, memory redundancy is also used to recover from yield loss due to bitcell stability failures as well [14].

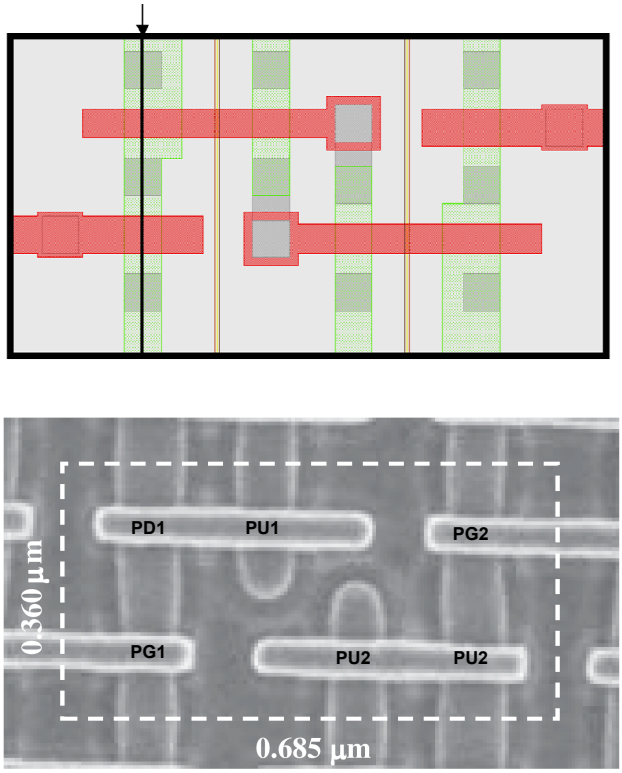


Figure 2.20: Layout and planar SEM image of a 0.246um^2 ultra high density SRAM bitcell in an industrial 45nm technology node [66].

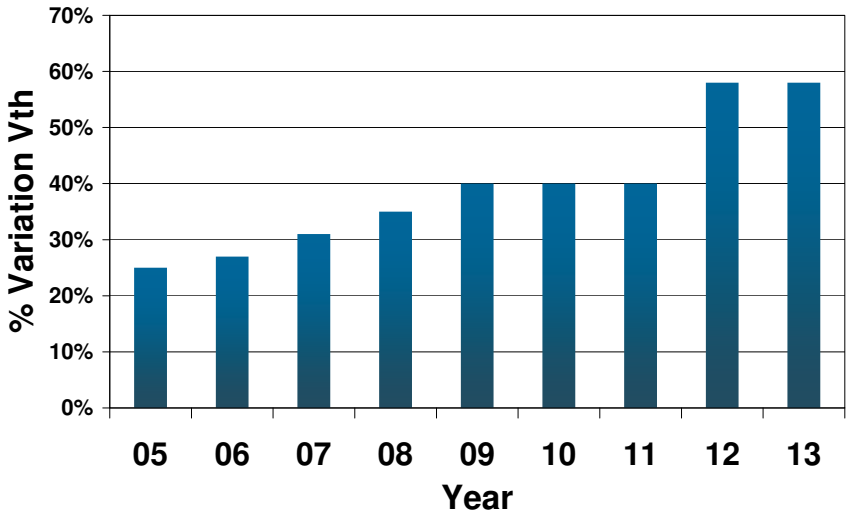
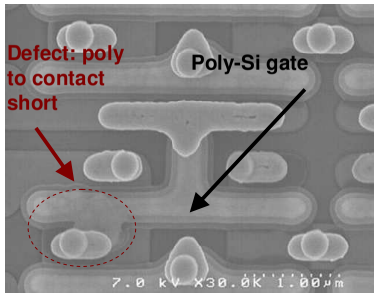


Figure 2.21: SRAM devices V_{th} variation scaling trend [5].

SRAM Failures

Hard Fails (Catastrophic Faults)

Defect density, opens, shorts, device breakdown.

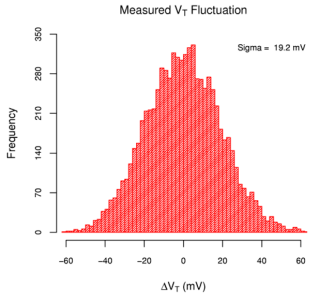


No dependence on supply voltage

Stability (soft) Fails

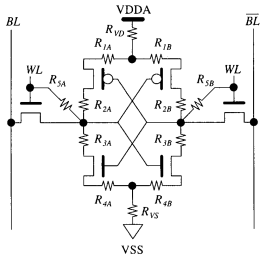
Intrinsic Device Variations

Vth variation due to RDF, LER...etc.



Baseline Process (extrinsic)

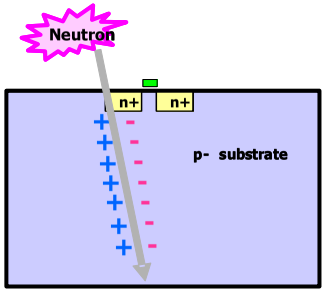
Contact resistance, silicide issues, oxide shorts, device breakdown...etc.



Bias dependent failures
Read Access
Write Stabiliy (WM)
Read Stability (SNM)
Hold or Retention

Radiation Induced Soft Error Rate (SER)

Caused by alpha particles (packaging) and neutrons (cosmic rays)



Loss of stored data (switching state)

Figure 2.22: Different types of SRAM failures.

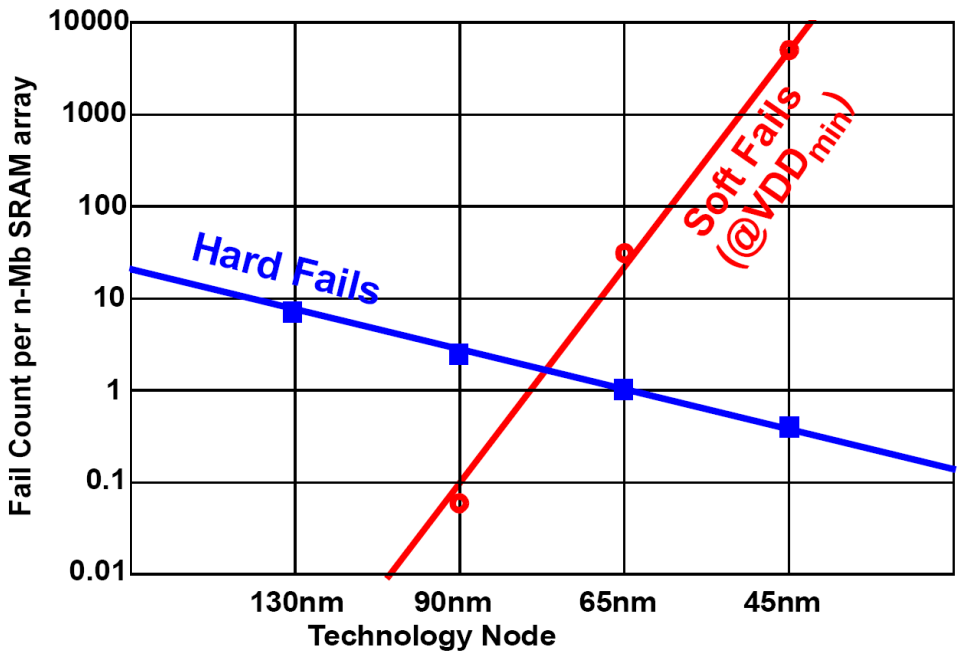


Figure 2.23: SRAM hard and soft fails scaling trend [64].

2.8.2 Bitcell Stability Failures

The predictions for failure probability is shown in Fig 2.23, where the fail count per memory density is shown for advanced technology nodes spanning 130nm down to 45nm. Traditional hard fails due to defect density decrease due to the reduction of bitcell size and improvement in defect density. However, as the bitcell size is reduced by about 50% every technology node, process variations increase significantly and become the dominant cause of bitcell failure [61, 63, 64]. This increase in SRAM failures has a strong impact on the overall product yield due to the high memory densities on chip. Moreover, lower V_{DD} operation becomes limited by the SRAM minimum supply voltage V_{DDmin}.

There are four main parametric failure mechanisms (also known as SRAM stability failures) [12–14, 68, 69]:

1. read access failure;
2. read stability;
3. write failure;
4. hold or retention fail.

These failures are parametric in nature since they affect the memory operation under specific conditions, but not under all environmental possibilities. For example, these failures mostly appear as V_{DD} is reduced, while they can be recovered at higher supply voltages. Therefore, these failure mechanisms become the limiter for SRAM supply voltage scaling [70–72].

2.8.2.1 Read Access Failure

During the read operation, the wordline (WL) is activated for a small period of time determined by the cell read current, bitline loading (capacitance) as shown in Fig. 2.24. In sense amplifier based memory architectures, the content of a cell is read by sensing the voltage differential between the bitlines. For successful read operation, the precharged to V_{DD} bitlines should discharge to a sufficient value which can trigger the sense amplifier correctly. A failure happens if bitcell read current (I_{read}) decreases below a certain limit. This may occur due to the increase in V_{th} for the pass-gate (PG) or pull-down (PD) transistors, or both. This decrease in I_{read} reduces the bitline differential sensed using sense amplifier. This may result in wrong evaluation using the sense amplifier. This type of failure shows a strong impact on memory speed [12–14]. This is because the WL activation period is about 30% of memory access time, and it is always desirable to reduce it to achieve higher speed operation [73].

2.8.2.2 Read Stability Failure

SRAM cells are designed to ensure that the contents of the cell do not get altered during read access while the cell should be able to quickly change its state during write operation. These conflicting requirements for read and write operations are satisfied by sizing the bitcell transistors to provide stable read and write operations [12–14].

In read operation, an SRAM bitcell is most prone to failure. After the wordline is enabled, the internal storage node storing a zero (Q) slightly rises due to the voltage divider between the pass-gate transistor (PG1) and the pull-down (PD1), as shown in Fig. 2.24. If the voltage at Q rises close to the threshold voltage of PD2, the cell may flip its state. In this case, stable read operation requires that PD1 should be stronger than PG1. Read stability is exacerbated by process variations which affect all the transistors in the bitcell [12–14]. To quantify the

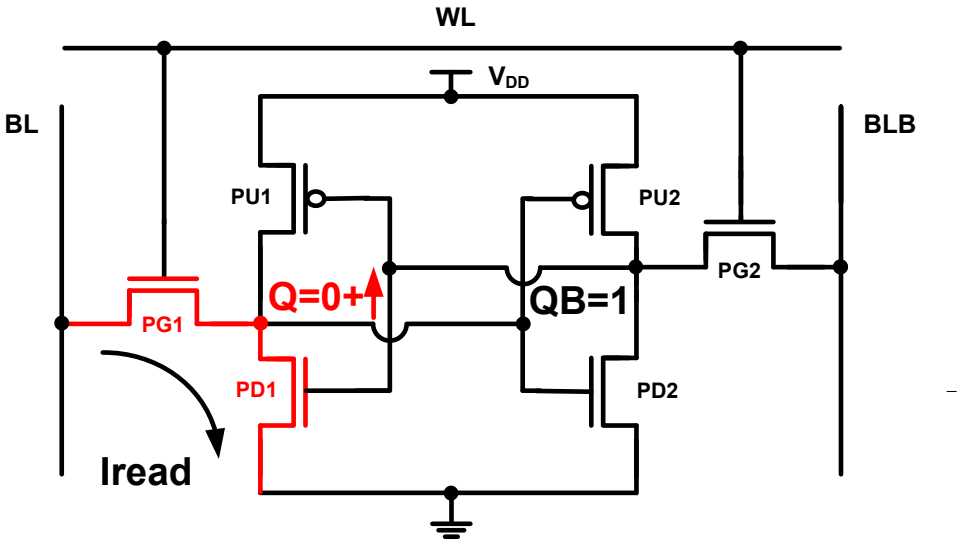


Figure 2.24: Bitcell in read operation.

bitcell's robustness against this type of failure, static noise margin (SNM) is the most commonly used metric [74].

SNM is defined as the maximum amount of voltage noise that a cell can tolerate [74]. SNM is calculated by finding the largest square which fits inside the butterfly curves, as shown in Fig. 2.25. A larger SNM implies higher robustness for the bitcell. However, due to WID variations, each transistor in the bitcell experiences different type of variation, hence, the symmetry of the bitcell is lost. This causes large spread in SNM as shown in measured SNM butterfly curves in Fig. 2.26. A read stability failure can occur if SNM reaches zero for any bitcell [12, 14, 74].

It is important to note that read stability failure can occur anytime the WL is enabled even if the bitcell is not accessed for read nor write operations. For example, in the case of half-selected bitcells, the wordline is enabled while the bitlines column is not selected (hence the bitcells are not actively accessed for read or write). These bitcells experience a dummy read operation because the bitlines begin to discharge and hence they become prone for SNM stability failure. Dealing with read stability failures from circuit design is one of the biggest challenges for SRAM design and has recently seen an extensive amount of research, especially for low voltage operation [12–14]. Moreover, it has been shown that SNM related failures are the limiter for V_{DD} scaling especially after accounting for device degradation due to negative bias temperature instability (NBTI). This trend is shown in Fig 2.27 where SNM failures for stressed bitcells cause the minimum supply voltage to increase (worsen) [71, 76, 77].

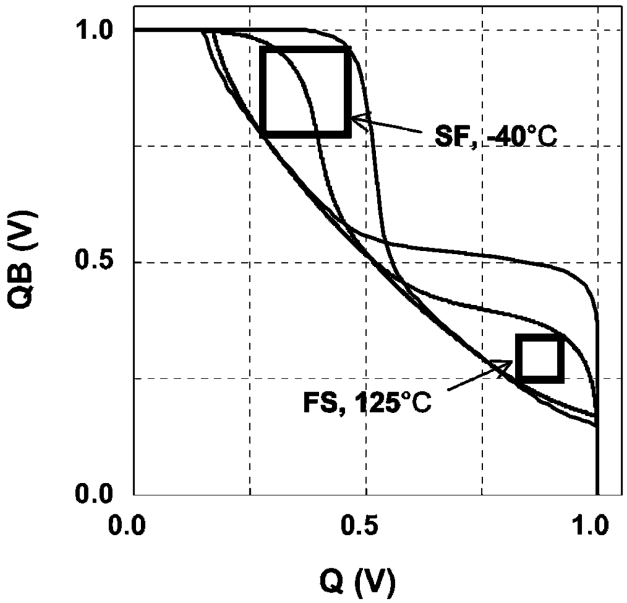


Figure 2.25: SNM butterfly curves for a 45nm SRAM bitcell for different process corners, FS: fast NMOS, slow PMOS and SF: slow NMOS, fast PMOS. These curves correspond to the case when WID variations are not included where good symmetry is shown for the butterfly curves [75].

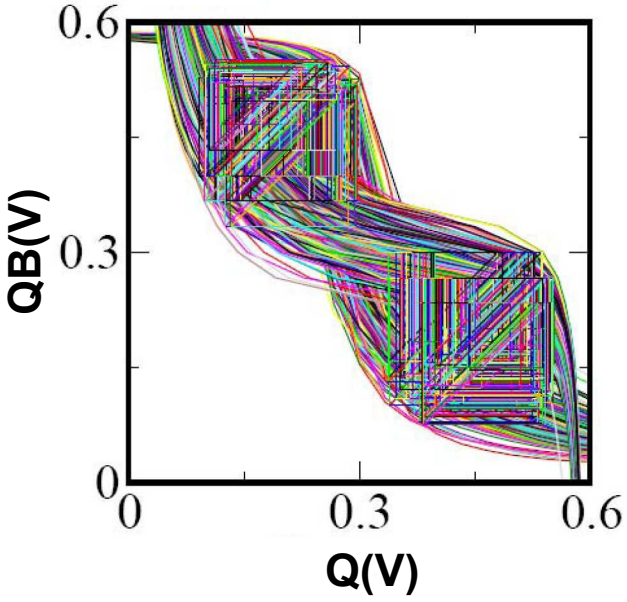


Figure 2.26: Measured SNM butterfly curves for 512 bitcells in 65nm technology node showing the strong impact of WID variations on SNM [63]. Large spread in butterfly curves causes SNM to be unsymmetrical and increases the probability of bitcell failure.

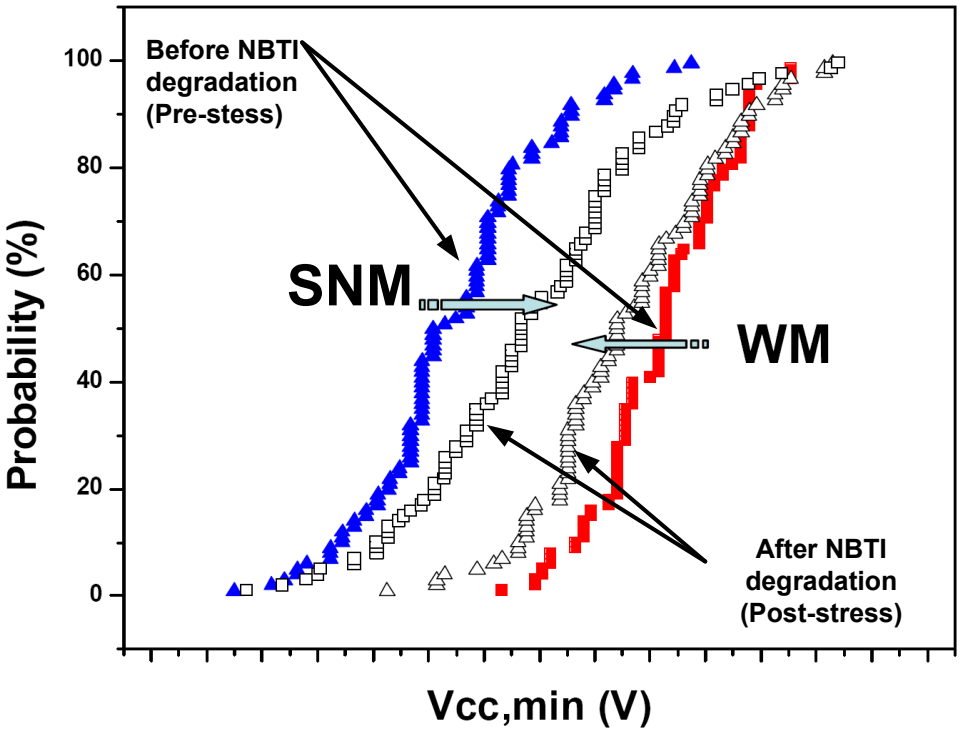


Figure 2.27: Minimum SRAM supply voltage (V_{ccmin}) distribution for SNM and WM limited failures before and after NBTI stress [71].

2.8.2.3 Write Stability Failure

Write stability (or write-ability) is also as important as having a good read stability. In write operation, BLB is pulled to zero using write driver as shown in Fig. 2.28. Therefore, the NMOS PG2 is turned ON, which results in a voltage drop in the storage node QB holding data 1 until it reaches below $V_{DD} - V_{th}$ for the PU1, where the feedback action begins. For stable write operation, PG2 should be stronger than PU2. One way to quantify a cell's write stability is using write trip voltage or write margin (WM), which is the maximum BLB voltage that can cause the bitcell to write, as shown in Fig. 2.29. This voltage should be far enough from either supply that no combination of offsets can cause a write failure or a write when a read is intended [14, 68, 78].

Due to WID variations, WM varies, and a write failure happens when an SRAM cell fails to write a desired state during the write operation. Mathematically, this occurs if WM is less than zero, which means that bitcell cannot be written [14, 78].

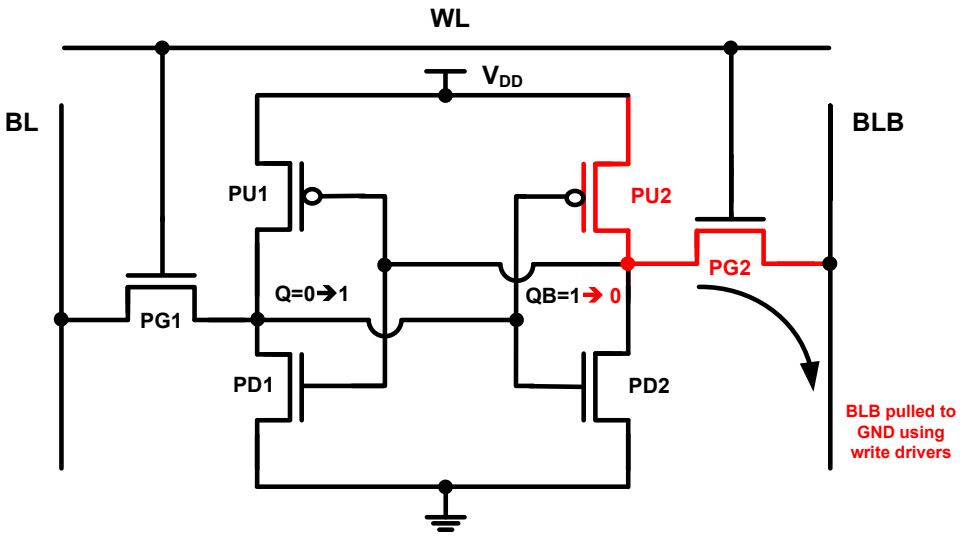


Figure 2.28: Bitcell in write operation.

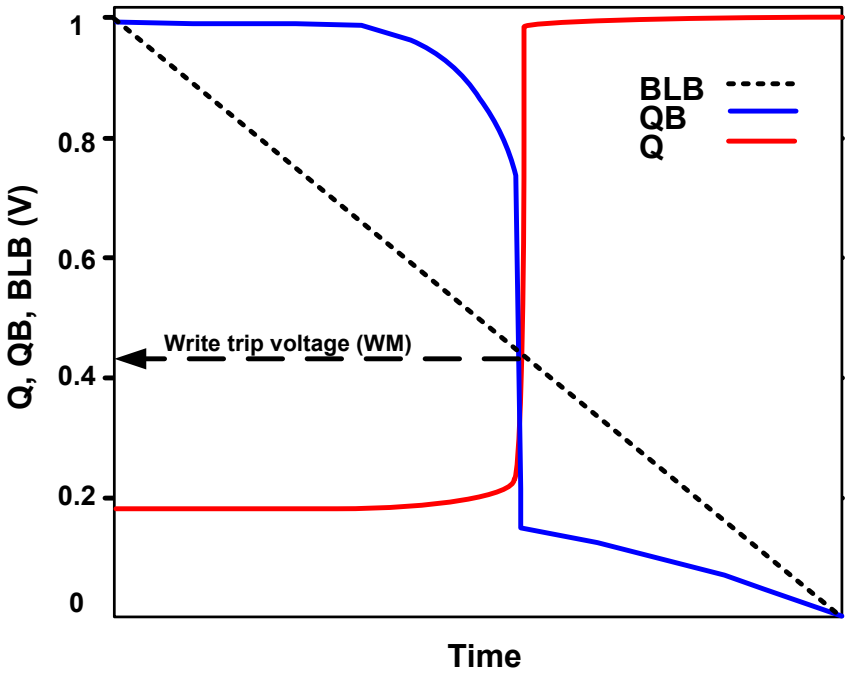


Figure 2.29: Write trip simulation where BLB voltage is swept from V_{DD} to zero, and the internal storage nodes Q and QB are monitored. WM is defined as BLB voltage at which Q reaches V_{DD} (the bitcell is written successfully).

2.8.2.4 Data Retention Failure

Reducing supply voltage (V_{DD}) is one of the most effective techniques to reduce both static and dynamic power consumption for digital circuits [52]. In SRAM, the data retention voltage (DRV) defines the minimum V_{DD} under which the data in a memory is still preserved. It is important to note that when V_{DD} is reduced to the DRV, all six transistors in the SRAM cell operate in subthreshold region, hence, show strong sensitivity to variations [70, 72].

DRV depends strongly on WID variations in the back to back inverters, which causes the bitcell to be imbalanced. This imbalance can be examined using SNM in standby (wordline is disabled) as shown in Fig. 2.30. If the bitcell is symmetric, its internal storage nodes Q and QB converge to the metastable point (V_M) as V_{DD} is reduced. Hence, the bitcell does not have stable point, and the stored data is lost. In contrast, if the bitcell is asymmetric due to WID variations, the bitcell tend to have a higher DRV than the symmetric case. This can be explained using SNM, where DRV voltage can be defined as the voltage when SNM is equal to zero. Note that in the symmetric case, both SNM High (upper-left square) and SNM Low (lower-right square) decrease symmetrically to zero. However, in the case of asymmetric bitcell shown in Fig. 2.30, SNM Low is always larger than SNM High, and the bitcell DRV is limited by the SNM High case. Therefore, the unbalanced bitcell is more sensitive to V_{DD} when Q node stores a zero [70, 72].

2.8.3 Radiation Induced Soft Errors

SRAM are susceptible to dynamic disruptions known as soft event upsets (SEU) [16]. SEU arise due to energetic radiation (Alpha particles or cosmic rays) that hits the silicon substrate and generates free electron-hole pairs. These electron-hole pairs can affect the potential of bitcell storage nodes and flip the stored data. To determine the susceptibility of SRAM to SEUs, the critical charge that can cause a storage node to be disrupted (Q_{crit}) is calculated. However, with technology scaling, SRAM junction capacitance, cell area and supply voltage are all scaled down. These reductions have opposing effect on Q_{crit} and the collected charges. However, it has been shown that the combined effect causes SRAM single bit SER to saturate or slightly decrease with technology scaling [79–81], as shown in Fig. 2.31. It is important to note that this trend in single-bit SER does not translate to reduction in the overall system failure rate due to the rapid growth in embedded SRAM density. In fact, SRAM systems failure rates are increasing significantly with scaling

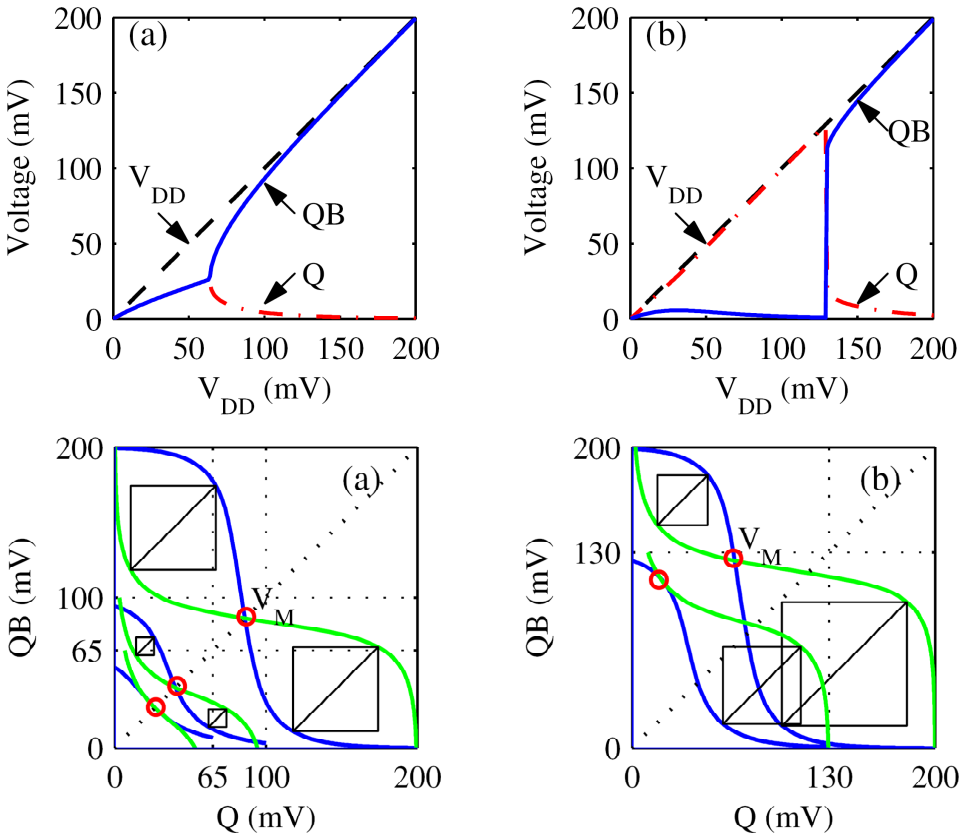


Figure 2.30: Data retention failure mechanism. Upper figures show the bitcell internal node voltages Q and QB for (a) balanced and (b) imbalanced cell as V_{DD} is reduced. Lower figures show the voltage transfer characteristics (VTC) of (a) balanced and (b) imbalanced cell with varying V_{DD}. V_M is the trip point of the VTCs [72].

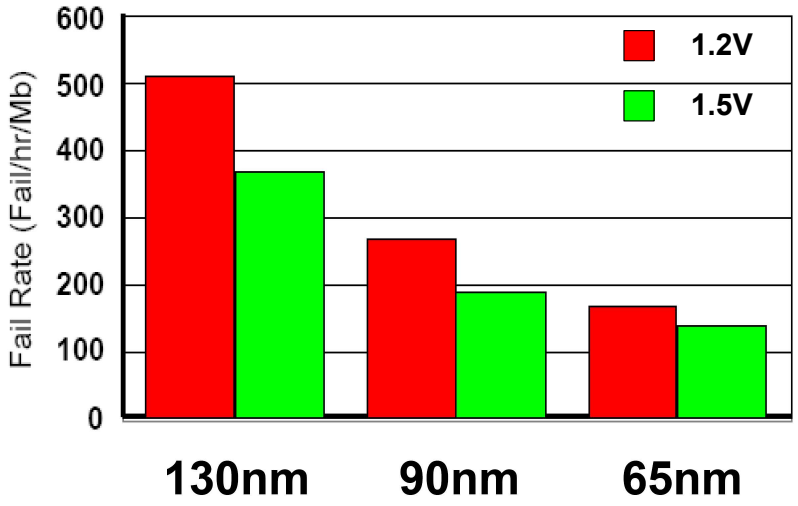


Figure 2.31: SER fail rate for different technology nodes. SER per bit value tend to decrease with scaling [61].

and have now become a major reliability concern for many applications [61, 79]. Moreover, it has been shown that process variations lead to large variation in Q_{crit} which also affects SER [82].

To mitigate soft errors, several radiation-hardening techniques can be implemented through process technology (*e.g.*, SOI technology), circuit design (*e.g.*, adding feedback capacitor, larger transistors, memory words interleaving) and architecture (*e.g.*, parity, error correction codes) or a combination of all these techniques [79]. It is important to note that SER budget for chips or systems is typically set based on target market requirements (which determines the level of required mitigation). For example, for single users, single chip applications as in mobile phones, it is acceptable to have an average failure rate of about one error every two years due to SER. On the other extreme, the same failure rate is not acceptable for high reliability systems utilizing hundreds of chips as in telecom base stations or servers applications [79, 80, 83].

2.9 Summary

In this chapter, we have presented a survey on the sources of variations that affect nanometer CMOS technology. It was shown that variability is worsening with technology scaling due to the increase in device variations such as RDF, CD variation and LER as well as interconnect variations. We also presented an overview on different research works in the area of analysis and mitigation of variability, where

there has been a clear trend in utilizing both circuits and architecture approaches to mitigate variability. In this chapter, we have also shown how process variations present a huge challenge for SRAM design. This is caused by the increase in SRAM failure mechanisms as supply voltage is reduced. The following chapters will present techniques to understand the impact of variation on different circuits, and to cope with these variations in the design phase (pre-fabrication) and in post-fabrication phase.

Chapter 3

A Statistical Design-Oriented Delay Variation Model Accounting for Within-Die Variations

The increase of statistical variations in advanced nanometer CMOS technologies poses a major challenge for digital circuit design. In this chapter, we study the impact of random variations on the delay variability of a gate, and derive simple and scalable statistical models to evaluate delay variations in the presence of within-die (WID) variations. The derived models are verified and compared to Monte Carlo SPICE simulations using industrial 90nm technology. This work provides new design insight and highlights the importance of accounting for the effect of input slew on delay variations, especially at lower supply voltages. This chapter is organized as follows: in Section 3.1, we introduce the problem of delay variation modeling, and the objectives of this work. In Section 3.2, we explain the modeling methodology and present the steps to derive our models. In Section 3.3, our models are compared with Monte Carlo SPICE simulations using an industrial 90nm technology. In Section 3.4, we discuss insights from our models. In Section 3.5, we present the conclusions.

3.1 Introduction

Variability is a major challenge facing the semiconductor industry [3, 5, 6] due to aggressive scaling of CMOS technology. As discussed in Chapter 2, variations due to fundamental physical limits, such as random dopants fluctuation and line edge roughness (LER) are increasing significantly with technology scaling [3, 7, 25, 33]. Moreover, manufacturing tolerances in process technology are not scaling at the same pace as transistor's channel length, due to process control limitations (*e.g.*, sub-wavelength lithography) [5, 6]. Therefore, within-die (WID) statistical process variations worsen with successive technology generations. Additionally, digital circuits show an increased sensitivity to process variations due to low-power and low-voltage operation requirements, which can result in failing to meet timing constraints.

To overcome variability challenges, a lot of recent research has attempted to address the impact of variations on timing. However, most of this research is primarily intended for statistical static timing analysis (SSTA) tools [9,11,17]. From a design perspective, a few works have been published in statistical delay modeling to derive analytical models that provide insight on how variations impact delay. In [84], a model for gate delay variation was proposed and the delay variation's dependence on supply voltage was derived based on Alpha-power model [85]. Here, a step input was assumed which does not account for the input rise time effect on delay variation. In [86], the authors presented a semi-analytical model to estimate the impact of random V_{th} variation on delay. Despite its accuracy in modeling gate delay variation, the model is complex and provides little insight to circuit designers. Therefore, it is more appropriate for a CAD implementation. Recently, an analytical approach was used in [87] to develop a delay model to study the impact of process variations on gate delay in both subthreshold and superthreshold regions. However like [84], the resulting model failed to account for the impact of input rise time on delay variation.

To facilitate variation-aware design, it is important to derive analytical delay models that can be used in performance estimation. These models should be simple, and provide insight on the impact of process variation on delay. In addition, having scalable models (in terms of bias dependence and technology scaling) is a fundamental requirement in order to use them for circuit optimization and technology exploration. Bearing this in mind, in this research we study the impact of WID variation on gate delay variability. Our goal is to clearly identify how process variations interact with circuit and design conditions to affect delay variability. In

particular:

1. we derive simple and scalable statistical models to estimate the impact of random variation on the delay variation of a gate;

2. the derived models show explicit dependence on design parameters such as supply voltage, input slope, and output load;

3. for the first time, we model and show the strong impact of input slew on delay variability and show that delay variation is affected by whether the input slew is slow or fast[1];

4. we analytically derive the conditions to achieve the minimum relative delay variation and verify the results with industrial 90nm technology using thorough Monte Carlo simulations.

These results are particularly important for the design in nanometer technologies showing large WID variations [3, 21, 88], as well as for low-power circuits with reduced supply voltages V_{DD}.

3.2 Model Assumptions and Derivation

One of the main objectives of this work is to derive simple and scalable models that can be used in design optimization, give insights on how random WID variations affect delay, and how different design decisions can be used to reduce delay variation. We ensure that the model is simple and accurate enough to provide clear design insights into the impact of random variation on delay variability. Having a simple model is a key requirement to be able to use the model in the optimization at the circuit and architecture levels. In addition, the model should also account for the dependence of delay variation on important circuit design decisions such as V_{DD}, sizing and gate loading. Toward that end, we make the following assumptions:

1. The dominant source of a gate's delay variation is the transistor's driving current variation. While variations in channel length will also introduce fluctuation in the input gate capacitance, nevertheless, this contribution on delay

[1]The definition of fast or slow input slew will be presented in Section 3.2.2.

is much smaller than the variation in drive current variation [88]. Moreover, variations in the interconnect are also much smaller than current variations [3, 88]. Nevertheless, the model can be easily extended to account for variations in interconnect capacitance.

2. The impact of process variations on delay can be computed using linear approximation. This assumption is accurate since variations are (by definition) small compared to the mean value, and device characteristics can be linearized around their nominal values [34, 84, 87, 89]. Hence, under linear approximation, the mean propagation delay of the gates can be approximated by the deterministic gate delay when variations are neglected. Therefore, process variations will mainly affect the spread (or variance).

3. V_{th} variations are assumed to be the dominant source of delay variability [3, 84,88]. To simplify the analysis, channel length variations are assumed to only affect V_{th} via short channel effects as shown in Eq. (2.2), therefore, can be included in $\sigma^2_{V_{th}}$ as shown in Eq. (3.1). It is important to note that the effect of other process variations on transistor's drive current can be translated into an effective variation in threshold voltage.

From a circuit modeling approach, the total variation in V_{th} due to RDF and channel length variations as well as other sources of variation, can be formulated as:

$$\sigma^2_{V_{th}} \approx \sigma^2_{V_{th,RDF}} + \sigma^2_{V_{th,L}} + \sigma^2_{V_{th,other}} \tag{3.1}$$

Throughout this work, we will be dealing with the total variation in threshold voltage ($\sigma_{V_{th}}$) as expressed by Eq. (3.1).

We first look into how process variations affect charging/discharing currents, that affect the gate delay under a step input $T_{pHL,step}$. Then, we derive models to account for the impact of input rise time on delay variation.

3.2.1 Variation in Charging/Disharging Current

In the following sections we look into a High-to-Low transition for an inverter, where the pull-down NMOS transistor discharges the output capacitance. However, the results are also applicable for Low-to-High transitions. To a first order, the High-to-Low propagation delay for a step input $T_{pHL,step}$ of an inverter can be estimated as [67]:

$$T_{pHL,step} = \frac{C\Delta V}{I_{av}} \tag{3.2}$$

where C is the output capacitance, I_{av} is the average discharging current for the output capacitance, and ΔV is the output voltage swing, where usually $0.5V_{DD}$ is used.

In the presence of process variations, both I_{av} and C will vary due to several statistical variations mechanisms (*e.g.*, random dopant fluctuation, channel length variation, etc., as explained in Section 2.2). However, C variation due to interconnect is much smaller than driving current variation [3], and therefore, it will be neglected as explained earlier. Hence, it is reasonable to say that the main contributor for delay variation is due to variation in I_{av} [3]. Nevertheless, in case of large C variations, the model can be easily extended to account for this effect.

A small change in I_{av} (ΔI_{av}) will cause incremental change in propagation delay T_p (ΔT_p) which can be calculated using Taylor expansion for Eq. (3.2) around the nominal value as follows:

$$\begin{aligned} \Delta T_{pHL,step} &\approx \frac{\partial T_{pHL,step}}{\partial I_{av}} \Delta I_{av} \\ &= -T_{pHL,step} \frac{\Delta I_{av}}{I_{av}} \end{aligned} \tag{3.3}$$

where ΔI_{av} is the variation in the average discharging current.

For a step input with zero rise/fall time, I_{av} is comprised of the driving current of only one transistor in the inverter since the other transistor will be immediately OFF after the input changes state. However, if rise/fall times are finite, I_{av} will be a function of NMOS and PMOS currents since both transistors will be ON simultaneously for a certain duration in the switching. In our derivation, we will neglect the contribution of the partially OFF transistor in I_{av} variations to simplify the analysis. Therefore, ΔI_{av} variation will be composed of only NMOS current variation when discharging (*e.g.*, $\Delta I_{av}/I_{av} = \Delta I_n/I_n$) and PMOS current variation when charging (*e.g.*, $\Delta I_{av}/I_{av} = \Delta I_p/I_p$), where I_n and I_p are the drain saturation currents for NMOS and PMOS devices, respectively. This assumption will be justified by the good accuracy of the model compared to Monte Carlo simulations as will be shown in Section 3.3.

Due to device variations, ΔI_{av} will be a function of different types of process variations as shown in Section 2.2. However, since V_{th} fluctuations due to RDF and LER increase significantly with technology scaling at a rate much higher than the other types of variations [3, 33], it is useful to concentrate on the impact of V_{th} variations on I_{av}. In addition, the variation in other sources can be lumped into V_{th} variation as shown in Eq. (3.1). Using Taylor series, we can approximate the

variations in I_{av} as follows:

$$\Delta I_{av} \approx \frac{\partial I_{av}}{\partial V_{th}} \Delta V_{th}. \tag{3.4}$$

The effect of V_{th} fluctuation on transistor's current can be calculated by assuming that V_{GS} fluctuates with a value of ΔV_{th}, while V_{th} itself is constant. This idea is not new as it has been widely used in analog design in combination with small signal analysis to find the impact of statistical variations on sensitive analog circuits (*e.g.*, how V_{th} mismatch affects differential amplifier offset and current mirror accuracy [29], [34]). Therefore, Eq. (3.4) can be written as:

$$\Delta I_{av} = -\frac{\partial I_{av}}{\partial V_{GS}} \Delta V_{th} = -g_m \Delta V_{th} \tag{3.5}$$

where $g_m = \partial I_{av}/\partial V_{GS}$ is the transconductance of the device. Mathematically, this can also be justified by noticing that in all regions of MOSFET operation, the drain current shows a dependence on $V_{GS} - V_{th}$, hence, by using differentiation by substitution, we can reach the same conclusion [18, 24, 34, 90].

By substitution from Eq. (3.5) in (3.3), we get:

$$\frac{\Delta T_{pHL,step}}{T_{pHL,step}} = \frac{g_m}{I_{av}} \Delta V_{th} = \frac{g_m}{I_D} \Delta V_{th}. \tag{3.6}$$

From this equation, it is clear that V_{th} fluctuation directly impacts delay variation after being multiplied by g_m/I_D. This shows that g_m/I_D has a strong impact on delay variations. Therefore, it is important to investigate the bias dependence of g_m/I_D.

Fig. 3.1 shows typical dependence of g_m/I_D [2] versus the overdrive voltage $V_{GS} - V_{th}$. At a high overdrive voltage, the transistor is in strong inversion and the value of g_m/I_D is small and it increases as the overdrive voltage is reduced (approximately following a $1/(V_{GS} - V_{th})$ dependence). However, as the device enters weak inversion or subthreshold operation, g_m/I_D saturates and reaches a maximum value of $g_m/I_D = 2.3/S$, where S is the subthreshold slope [24]. Typically, S ranges from 80-100 mV/decade for advanced CMOS technologies, hence, the maximum value for g_m/I_D is typically around 23 to 29 V^{-1}. Since delay variations are directly proportional to g_m/I_D, $\Delta T_{p,step}/T_{p,step}$ will follow the same bias dependence for g_m/I_D as shown in Eq. (3.6). Therefore, as V_{DD} is reduced, the impact of variations increases significantly.

[2] In analog design, this ratio of transconductance to current is called the transconductance efficiency. This ratio is fundamental to MOSFET and provides guidance to designer, on the region with highest gain at small current dissipation [90].

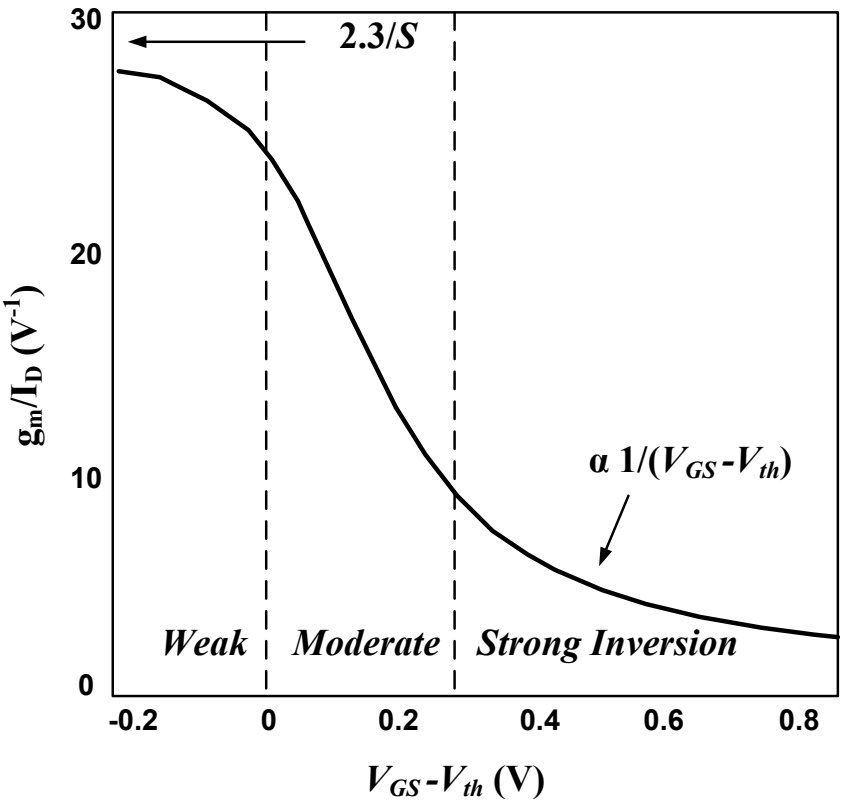


Figure 3.1: Typical bias dependence of g_m/I_D versus overdrive voltage $V_{GS} - V_{th}$ for a device in saturation region [34]. In strong inversion, g_m/I_D initially increases proportional to $(V_{GS} - V_{th})^{-1}$ as V_{GS} is reduced and saturates toward a maximum value of $2.3/S$, where S is the subthreshold slope.

3.2.2 Impact of Finite Input Slew on Delay Variation

The delay variation of a gate in a real circuit cannot be predicted by assuming a step input as in Eq. (3.6). This is because the dynamics of switching for a gate are much more complicated when the input has a finite rise time. It is well known that the input signal rise time (*i.e.*, input slope or slew) has a strong impact on delay [85, 91, 92]. Therefore, there has been much work to model the impact of finite input slope on delay. Since these models were mainly derived to give accurate predication of delay, they tend to be very complicated. This limits their capability of providing design insights or guidelines when accounting for statistical variations [91, 92].

While there has been much work on how delay is affected with input slope, interestingly, there has been very limited work that accounts for input slopes effect on delay variation. In [86], the authors account for impact of input slew on delay

variation using a semi-empirical model, which, although accurate, does not present clear design insights due to it complexity. Recently in [89], a numerical model was developed to account for input slew effect on delay variation. However, the design parameters are not explicit and the model does not provide an intuitive understanding of how different design decisions affect delay variation.

In this section, we model the impact of input rise time (T_r) on delay variation σ_{TpHL} of an inverter. As mentioned earlier, our goal is to derive simple delay variation models that provide insight for designers. For example, the model should enable us to estimate delay variation from the knowledge of basic technological and design parameters. It is important to note that the accuracy of modeling gate delay itself is not required in this case. Therefore, in our derivations, instead of focusing on accurately modeling the propagation delay as in [91, 92], we explore several simplifications that would allow us to reach a design-oriented delay variation model that is simple, accurate, and enables us to explore different design tradeoffs.

The input/ouput characteristics for an inverter is governed by the following differential equation:

$$C_L \frac{dV_{out}}{dt} \approx I_p - I_n \tag{3.7}$$

where C_L is the load capacitance (including diffusion, wire loading, the following gate's input capacitance and the impact of Miller capacitance). I_p and I_n are the PMOS charging and NMOS discharging currents, respectively. For a High-to-Low transition, and neglecting the PMOS current to simplify the analysis, we get:

$$C_L \frac{dV_{out}}{dt} \approx -I_n. \tag{3.8}$$

In our derivation, we focus on the supply voltage range covering strong inversion region and we do not account for subthreshold operation. To simplify the analysis, we use the well-known Alpha-power model for the NMOS discharging current [85]:

$$I_n = \begin{cases} 0 & V_{GS} \leq V_{th} \\ k_n (V_{GS} - V_{th})^{\alpha} & V_{GS} \geq V_{th} \end{cases} \tag{3.9}$$

$$k_n = k_n^{'} \frac{W}{L} \tag{3.10}$$

where V_{th} is the threshold voltage of the NMOS pull-down transistor, $k_n^{'}$ is a technological parameter, α is an exponent ranging from 1.4 to 2 depending on whether the transistor is in velocity saturation or pinch-off saturation, W and L are the width and length of the transistor, respectively.

Assuming a linear input ramp:

$$V_{GS} = \frac{t}{T_r} V_{DD} \tag{3.11}$$

where T_r is the input rise time defined from 0 to 100%. In a real circuit, the input is not exactly linear, since it is essentially the output of the preceding gate. Nevertheless, the assumption of a linear input ramp will not affect the validity of the final results [91, 92].

Solving the differential equation Eq. (3.8) for V_{out}, and noticing that there are different regions for V_{out} as determined by input rise time T_r, we get:

$$V_{out}(t) = \begin{cases} V_{DD} \quad \text{for} \;\; t \leq \frac{V_{th}}{V_{DD}} T_r \\ \\ V_{DD} - \frac{k_n T_r}{C_L V_{DD}(\alpha+1)} \left[\frac{t}{T_r} V_{DD} - V_{th} \right]^{\alpha+1} \quad \text{for} \;\; \frac{V_{th}}{V_{DD}} T_r \leq t \leq T_r \\ \\ V_{DD} - \frac{k_n T_r}{C_L V_{DD}(\alpha+1)} (V_{DD} - V_{th})^{\alpha} \times \left[\frac{t}{T_r} V_{DD}(\alpha+1) - (\alpha V_{DD} + V_{th}) \right] \\ \quad \text{for} \;\; t \geq T_r \end{cases} \tag{3.12}$$

For a given input rise time, V_{out} discharges toward $V_{DD}/2$ following one of the above two equations, depending on the gate output loading C_L and driving capability through k_n and V_{DD}. For a fast input transition, the output load capacitor discharges and reaches $V_{out} = V_{DD}/2$ after V_{in} reaches its maximum value (*i.e.*, $t_{V_{out}=\frac{V_{DD}}{2}} > T_r$). In this case, the High-to-Low propagation delay Tp_{HL} can be expressed as [85]:

$$Tp_{HL,fastT_r} = T_r \left[\frac{1}{2} - \frac{1 - \frac{V_{th}}{V_{DD}}}{\alpha + 1} \right] + \frac{C_L \frac{V_{DD}}{2}}{k_n (V_{DD} - V_{th})^{\alpha}} \tag{3.13}$$

$$= T_r \left[\frac{1}{2} - \frac{1 - \frac{V_{th}}{V_{DD}}}{\alpha + 1} \right] + T_{pHL,step} \tag{3.14}$$

On the contrary, for a slow input transition [3], the output load capacitor discharges to $V_{DD}/2$ before V_{in} reaches V_{DD}. Therefore, for a slow input transition case, Tp_{HL} is calculated as:

$$Tp_{HL,slowT_r} = T_r \left[\frac{V_{th}}{V_{DD}} + \left(\frac{C_L V_{DD}(\alpha+1)}{2 k_n T_r} \right)^{\left(\frac{1}{\alpha+1}\right)} - \frac{1}{2} \right] \tag{3.15}$$

[3] In this case, we use $V_{out}(t)$ which is valid in the region $\frac{V_{th}}{V_{DD}} T_r \leq t \leq T_r$ and find the time t where $V_{out} = V_{DD}/2$.

The question now is what is the value of T_r which defines the boundary between fast and slow input rise times. By equating Eq. (3.13) and Eq. (3.15) we can find the value of the boundary rise time T_{rb} as follows:

$$T_{rb} = T_r|_{Tp_{HL,fastT_r}=Tp_{HL,slowT_r}} \tag{3.16}$$

$$= \left(\frac{\alpha+1}{1-\frac{V_{th}}{V_{DD}}}\right) Tp_{HL,step} \tag{3.17}$$

It is important to note that T_{rb} defines the boundary between a fast or slow input rise time T_r. These different regions of input rise time will have strong impact on delay variation as will be shown later in the results section.

Therefore, Tp_{HL} for any give value of rise time can calculated as:

$$Tp_{HL} = \begin{cases} T_r\left[\frac{1}{2} - \frac{1-\frac{V_{th}}{V_{DD}}}{\alpha+1}\right] + T_{pHL,step} & T_r < T_{rb} \\ T_r\left[\frac{V_{th}}{V_{DD}} + \left(\frac{C_L(\alpha+1)}{2k_n T_r {V_{DD}}^{\alpha-1}}\right)^{\frac{1}{\alpha+1}} - \frac{1}{2}\right] & T_r > T_{rb} \end{cases} \tag{3.18}$$

where T_{rb} is defined in Eq. (3.17).

We can now use Eq. (3.18) to calculate $\sigma_{Tp_{HL}}$ assuming that the random variable $\mathbf{V_{th}}$ varies around its nominal value V_{th} with a $\mathbf{\Delta V_{th}}$ having a zero mean and a standard deviation of $\sigma_{V_{th}}$. For the case where $T_r < T_{rb}$, from Eq. (3.18) we have:

$${\sigma^2}_{Tp_{HL}} = var\left(T_r\left[\frac{1}{2} - \frac{1-\frac{\mathbf{V_{th}}}{V_{DD}}}{\alpha+1}\right] + T_{pHL,step}\right) \tag{3.19}$$

$$= var\left(T_r\frac{\mathbf{V_{th}}}{V_{DD}(\alpha+1)} + \frac{C_L\frac{V_{DD}}{2}}{k_n(V_{DD}-\mathbf{V_{th}})^\alpha}\right) \tag{3.20}$$

$$\approx var\left(T_r\frac{V_{th}}{V_{DD}(\alpha+1)} + T_r\frac{\mathbf{\Delta V_{th}}}{V_{DD}(\alpha+1)} + \frac{C_L\frac{V_{DD}}{2}}{k_n(V_{DD}-V_{th})^\alpha} + T_{pHL,step}\frac{g_m}{I_D}\mathbf{\Delta V_{th}}\right)$$

$$\approx var\left(\left(\frac{T_r}{V_{DD}(\alpha+1)} + T_{pHL,step}\frac{g_m}{I_D}\right)\mathbf{\Delta V_{th}}\right)$$

$$\approx \left(\frac{T_r}{V_{DD}(\alpha+1)} + T_{pHL,step}\frac{g_m}{I_D}\right)^2 {\sigma^2}_{V_{th}} \tag{3.21}$$

$$\approx \left(\frac{T_r}{V_{DD}(\alpha+1)} + T_{pHL,step}\frac{\alpha}{(V_{DD}-V_{th})}\right)^2 {\sigma^2}_{V_{th}} \tag{3.22}$$

where Eq. (3.6) was used to model the variation in the second term in Eq. (3.20), and Alpha-power model was used to approximate the term g_m/I_D in Eq. (3.21) to get Eq. (3.22).

Similarly, for the case when $T_r > T_{rb}$, using Eq. (3.18) we get:

$$\sigma^2{}_{Tp_{HL}} = (\frac{T_r}{V_{DD}})^2 \sigma^2{}_{V_{th}} \tag{3.23}$$

Therefore, the delay variation $\sigma_{Tp_{HL}}$ due to V_{th} variation can be computed for any input slew T_r as follows:

$$\sigma_{Tp_{HL}} = \begin{cases} \left(\frac{T_r}{V_{DD}(\alpha+1)} + T_{pHL,step}\frac{g_m}{I_D}\right)\sigma_{V_{th}} \\ \quad \text{for} \quad T_r < T_{rb} \\ \frac{T_r}{V_{DD}}\,\sigma_{V_{th}} \\ \quad \text{for} \quad T_r > T_{rb} \end{cases} \tag{3.24}$$

where T_{rb} is defined in Eq. (3.17).

As shown in Eq. (3.24), when T_r is slow, delay variation is simply $T_r\,\sigma_{V_{th}}/V_{DD}$. To understand how at slow input slew delay variation takes this form, let's look at the dynamics of switching for an inverter driven by a slow input signal V_{in} as shown in Fig. 3.2. Let's further assume that the output voltage will not begin discharging except when V_{in} is greater than V_{thn} for the NMOS device. In the case of no variations ($\Delta V_{thn} = 0$), the propagation delay for the inverter is equal to the nominal propagation delay T_p, measured from the time V_{in} crosses $V_{DD}/2$ to V_{out} crossing the same value, as shown in Fig. 3.2. However, if due to V_{thn} variation we have $\Delta V_{thn} > 0$, as shown in Case 2, the starting point for discharging the output shifts to the right. Hence, the propagation delay increases by ΔT_p. Similarly, if $\Delta V_{thn} < 0$, the starting point for V_{out} discharge shifts to the left and T_p reduces by ΔT_p. It is clear that the variation in V_{th} causes delay variation due to the finite input rise time as shown in Fig. 3.2 and explained above. As V_{th} fluctuates, the starting point of discharging changes, which consequently adds up to delay variation. It can be shown that this effect will give delay variation of $\frac{T_r}{V_{DD}}\,\sigma_{V_{th}}$, as captured in Eq. (3.24) for the case of $T_r > T_{rb}$.

3.2.3 Minimum Relative Delay Variation σ_{Tp}/Tp

Based on the derived delay variation model in Eq. (3.24), it is useful to investigate whether there are certain conditions that can be used to minimize the relative delay variation, defined as the ratio of delay variation to nominal delay σ_{Tp}/Tp. This quantity is an important metric for delay variability, and is especially useful in

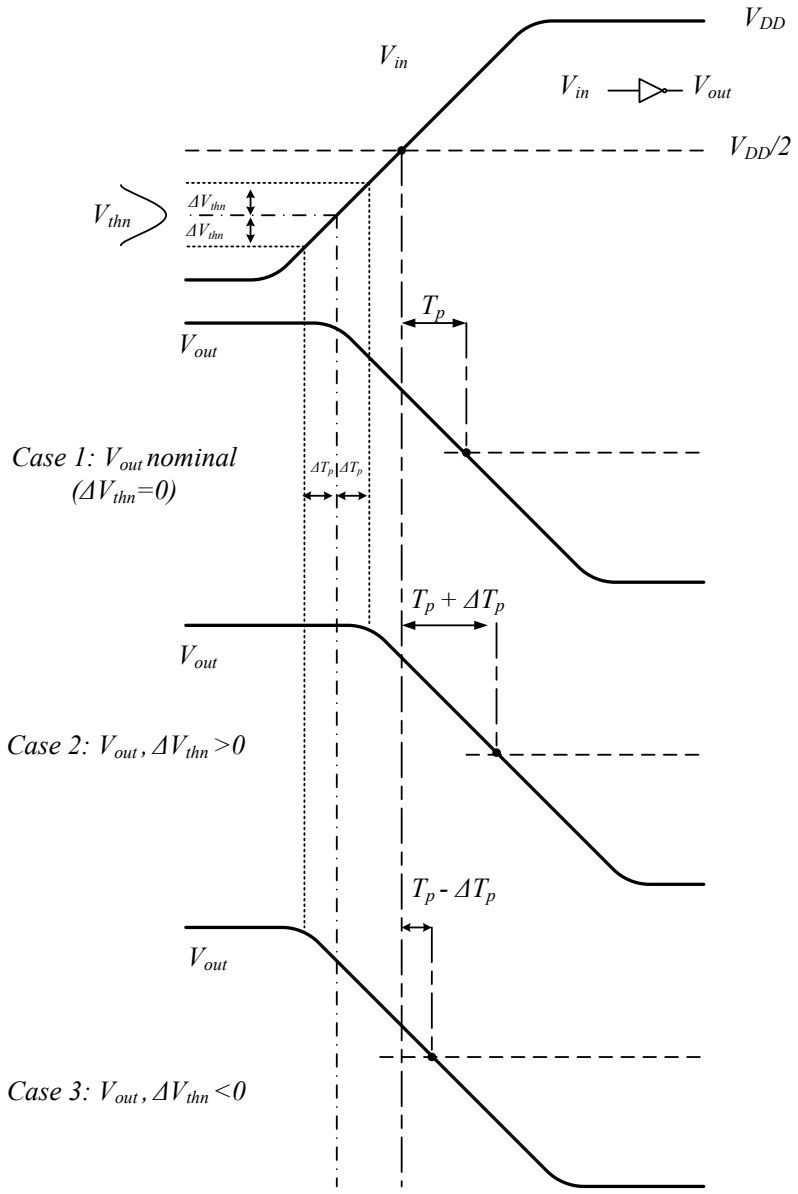


Figure 3.2: Delay variation for an inverter driven by a slow input rise time. Variation in V_{thn} affects the starting point of the switching, hence, introduces delay variation. Inverter delay is shown for three cases: 1) Nominal case with no V_{thn} variation ($\Delta V_{thn} = 0$) and nominal propagation delay T_p, 2) case with positive V_{thn} shift ($\Delta V_{thn} > 0$) and increased propagation delay $T_p + \Delta T_p$ and 3) case with negative V_{thn} shift ($\Delta V_{thn} < 0$) and decreased propagation delay $T_p - \Delta T_p$.

the design of clock distribution networks to minimize skew as well as in self-timed paths used in memory timing.

From Eq. (3.24) and Eq. (3.18), we can calculate the relative delay variation

$\sigma_{Tp_{HL}}/Tp_{HL}$. For a fast T_r, $\sigma_{Tp_{HL}}/Tp_{HL}$ can be expressed as:

$$\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r<T_{rb}} = \frac{\left(\frac{T_r}{V_{DD}(\alpha+1)} + T_{pHL,step}\frac{g_m}{I_D}\right)\sigma_{V_{th}}}{T_r\left[\frac{1}{2} - \frac{1-\frac{V_{th}}{V_{DD}}}{\alpha+1}\right] + T_{pHL,step}} \tag{3.25}$$

and for a slow T_r:

$$\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r>T_{rb}} = \frac{\frac{\sigma_{V_{th}}}{V_{DD}}}{\left[\frac{V_{th}}{V_{DD}} + \left(\frac{C_L(\alpha+1)}{2k_nT_rV_{DD}{}^{\alpha-1}}\right)^{\frac{1}{\alpha+1}} - \frac{1}{2}\right]}. \tag{3.26}$$

In the case of fast T_r, from Eq. (3.25), T_r is in both the numerator and denominator. Therefore, $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r<T_{rb}}$ versus T_r may show an increasing or decreasing trend depending of the value of V_{DD}, as will be shown in Section 3.3. However, in the case of slow T_r, from Eq. (3.26), $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r>T_{rb}}$ will always increase as T_r increases, hence, its minimum occurs at $T_r = 0$. Therefore, for a given V_{DD}, if both Eq. (3.26) and Eq. (3.25) increase as T_r increases, then the minimum relative delay variation will occur at $T_r = 0$. However, if $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r<T_{rb}}$ and $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r>T_{rb}}$ show opposite trends versus T_r, then the minimum point will occur in the boundary point between these two curves (*i.e.*, when $T_r = T_{rb}$).

By calculating $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ at $T_r = 0$ and $T_r = T_{rb}$, we can derive the value of V_{DD} at which $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r<T_{rb}}$ changes its slope versus T_r. Therefore, by utilizing the above mentioned trends, we can show that for a given supply V_{DD}, there exists a certain value of T_r which minimizes $\sigma_{Tp_{HL}}/Tp_{HL}$ as follows:

$$\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{min} = \begin{cases} \frac{2\sigma_{V_{th}}}{V_{DD}} \\ \text{at} \quad T_r = T_{rb}\,, \text{ for} \quad V_{th} < V_{DD} < \frac{2V_{th}}{2-\alpha} \\ \frac{\alpha\,\sigma_{V_{th}}}{V_{DD}-V_{th}} \\ \text{at} \quad T_r = 0\,, \text{ for} \quad V_{DD} > \frac{2V_{th}}{2-\alpha} \end{cases} \tag{3.27}$$

where $V_{DD} = \frac{2V_{th}}{2-\alpha}$ is the supply voltage at which $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{T_r<T_{rb}}$ changes its trend versus T_r.

From Eq. (3.27), we can see that V_{DD} and T_r have strong impact on $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$. At high supply voltages, $V_{DD} > \frac{2V_{th}}{2-\alpha}$, the minimum $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ occurs when $T_r = 0$, therefore, $\frac{\alpha\,\sigma_{V_{th}}}{V_{DD}-V_{th}}$ defines the lower bound on relative delay variation. Hence, any further increase in T_r will not only increase the absolute delay variation $\sigma_{Tp_{HL}}$ as shown in Eq. (3.24), but will also increase the relative delay variation $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$.

However, an interesting behavior occurs as the supply voltage is reduced below $\frac{2V_{th}}{2-\alpha}$. The minimum $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ occurs at $T_r = T_{rb}$ defined in Eq. (3.17), and its value is determined by the ratio of $2\sigma_{V_{th}}$ to the supply voltage V_{DD}. Identifying such trends is important, as they can be utilized in optimizing delay variation for circuits working at lower supply voltages to reduce power consumption. In addition, circuits which are sensitive to skew (*e.g.*, clock networks, memory self-timed paths), rather than the delay itself, can benefit from this finding as well.

3.2.4 Input Slew Variation

In our derivation, we assumed that T_r is constant. In reality, statistical process variations in a stage will also add to delay variation in the following stage. This is due the slope of the transition at the intermediate node between the two stage [86]. T_r will show statistical variation due to random variations in the previous gate's driving current. To calculate the impact of T_r variation on delay, we assume that T_r follows a normal distribution (similar to delay distribution). Hence, the random variable $\mathbf{T_r}$ varies around its nominal value T_r with a standard deviation of σ_{T_r}. The variation in Tp_{HL} due to T_r variation, σ_{Tp_{HL},T_r}, can be calculated as follows:

$$\sigma_{Tp_{HL},T_r} = \frac{\partial Tp_{HL}}{\partial T_r}\sigma_{T_r} \tag{3.28}$$

where $\frac{\partial Tp_{HL}}{\partial T_r}$ can be calculated from Eq. (3.18).

Now, the total delay variation due to V_{th} and T_r variations can be calculated from Eq. (3.24) and Eq. (3.28):

$$\sigma_{Tp_{HL},tot} = \sqrt{{\sigma_{Tp_{HL}}}^2 + {\sigma_{Tp_{HL},T_r}}^2} \tag{3.29}$$

where $\sigma_{Tp_{HL},tot}$ is the total delay variation due to both V_{th} and T_r variations. Eq. (3.29) shows that the total variation is the root mean square of delay variation from each component independently. This is valid since we are dealing here with random variations, hence, the correlation between the two delay variation components is zero.

As shown in Eq. (3.29), T_r variation increases the total delay variation. In addition, the total delay variation trend will depend on the relative magnitude of both $\sigma_{Tp_{HL}}$ and σ_{Tp_{HL},T_r} (if one component is much larger than the other, it will dominate the total delay variation trend). Therefore, it is important to analyze how T_r variation affects the minimum relative delay variation discussed in the Section 3.2.3. Unfortunately, detailed derivation for the minimum relative delay

variation including T_r variation is not easy, and does not provide clear design insight. Therefore, we will rely on simulation results to analyze this issue, as will be shown in Section 3.3

3.3 Results and Discussion

To verify the stage delay variation models, we compare the analytical models to simulation results using an industrial 90nm technology with technological parameters shown in Table 3.1. A thorough analysis using SPICE simulation is performed to validate the model. Monte Carlo SPICE simulation is used to estimate the impact of statistical variations on delay variability.

Table 3.1: 90nm Technology Information and inverter sizing for LX drive strength.

	NMOS	PMOS
Nominal V_{DD}	1.0—1.2 V	
W/L (μm/μm)	0.22/0.1	0.39/0.1
${V_{th}}^{\mathrm{A}}(m\mathrm{V})$	260	290
$\sigma_{V_{th}}(m\mathrm{V})$	18	13.5
${I_{D,sat}}^{\mathrm{B}}(\mu\mathrm{A})$	183	131

[A] $|V_{GS}| = |V_{DS}| = 1.2$ V.

In the following, we present the validation results for the proposed delay variation model. Inverters of different fan-out (FO) were used to examine the model's accuracy. Input slew was varied to find the impact of input rise time T_r on delay variation. For each Monte Carlo run, the delay of the inverter is measured using transient simulation. Large number of Monte Carlo runs (1000 to 4000 runs) were used to reduce the error associated with the statistical determination of the delay's mean and standard deviation. The simulations are repeated for different supply voltages from 0.6 V to 1.2 V to find the impact of reducing supply voltage on delay variability.

Inverters with LX drive strength from the standard cell library in this 90nm technology were used in simulation setups, as shown in Table 3.1. Hardware calibrated statistical models were used to account for V_{th} variations. Random variations are typically inversely proportional to the square root of the active device's area, as shown in Section 2.2 [93]. Therefore, the gates with lowest driving strength (LX drive) from the standard cell library typically show the largest delay variation.

Hence, they are appropriate for use to verify the proposed models. Nevertheless, the results are also valid for inverters of larger sizes which are used in critical paths.

Fig. 3.3 shows typical histograms for T_p (T_{pHL} and T_{pLH} for different supply voltages) from Monte Carlo simulation with a superimposed Gaussian distribution, which shows that delay distribution can be approximated by a Gaussian distribution. Fig. 3.3 also shows that reducing supply voltage significantly increases the relative delay variation.

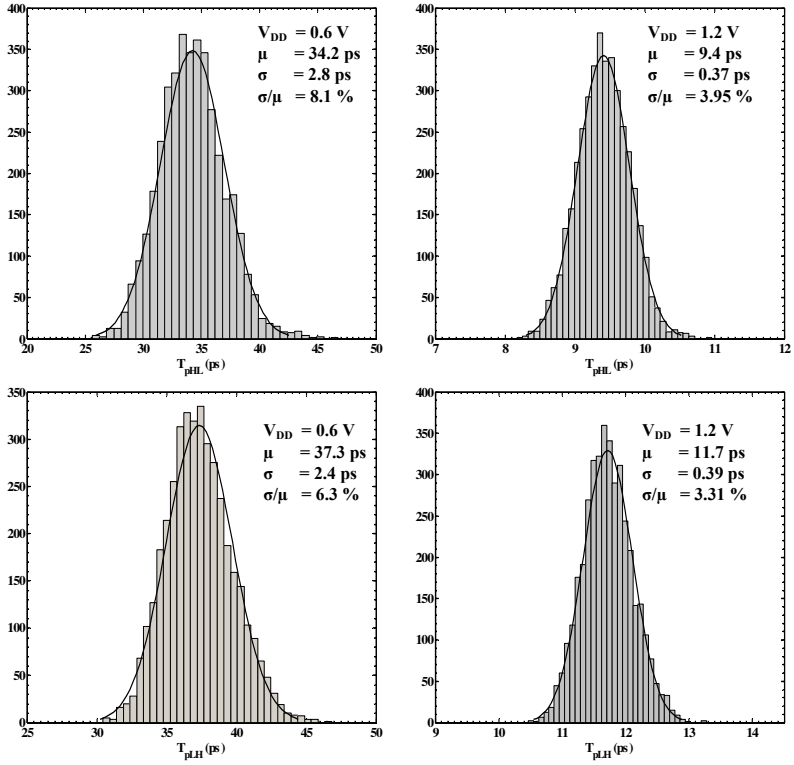


Figure 3.3: T_p Histogram for a single stage using Monte Carlo SPICE simulation (4000 runs): a) T_{pHL} at $V_{DD} = 0.6$ V ($T_r = 42.3$ ps), b) T_{pHL} at $V_{DD} = 1.2$ V ($T_r = 25.8$ ps), c) T_{pLH} at $V_{DD} = 0.6$ V ($T_f = 34.5$ ps) and d) T_{pLH} at $V_{DD} = 1.2$ V ($T_f = 21.7$ ps). Also shown: a Gaussian distribution having the same mean and standard deviation. T_r and T_f values correspond to the rise and fall times assuming the inverter is driven by another inverter of the same size.

Fig. 3.4 shows the deterministic T_{pHL}, T_{pLH} and T_p using transient simulation for FO1 inverter. Also on the same plot, $\mu_{T_{pHL}}$ and $\mu_{T_{pLH}}$ from Monte Carlo simulations are shown. Clear agreement between T_{pHL} and $\mu_{T_{pHL}}$, as well as between T_{pLH} and $\mu_{T_{pLH}}$, justifies the linearity assumption used in Section 3.2 down to V_{DD}=0.6 V (*i.e.*, process variations do not affect the delay's mean, but affect the delay spread).

In Section 3.2.2, we have shown that input slew has a strong impact on delay variations. Fig. 3.5 shows the simulation result for $\sigma_{Tp_{HL}}$ at $V_{DD} = 0.7$ V. Note that each data point represents the delay variation $\sigma_{Tp_{HL}}$ calculated from 1000 Monte Carlo runs at that specific value of input slew T_r. Fig. 3.5 also shows the results from our proposed models for $\sigma_{Tp_{HL}}$ for fast and slow T_r (Eq. (3.24)). The proposed models matches the simulation results.

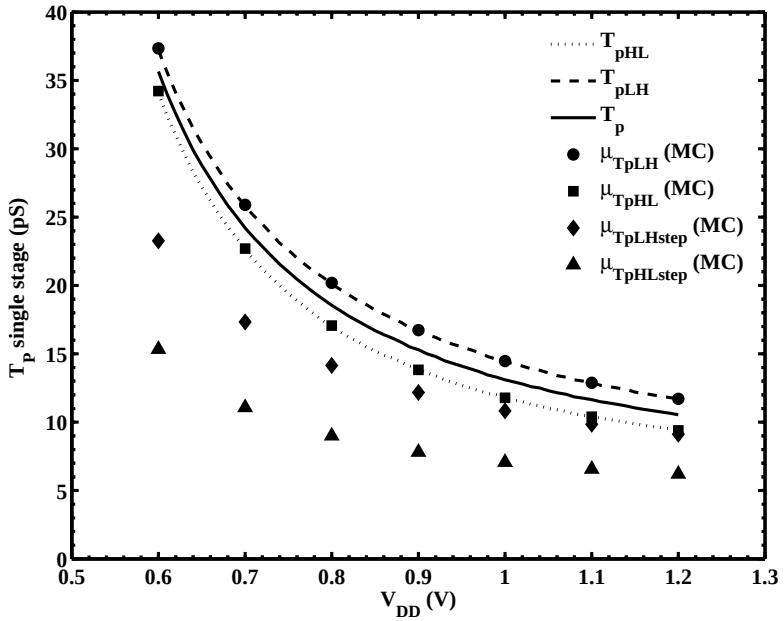


Figure 3.4: T_p versus V_{DD} for a single stage showing the nominal High-to-Low (T_{pHL}), Low-to-High (T_{pLH}), and the average T_p. Also shown are $\mu_{T_{pHL}}$ and $\mu_{T_{pLH}}$ for both nominal rise/fall time and step input using Monte Carlo simulation.

Clearly, $\sigma_{Tp_{HL}}$ increases linearly with T_r as was shown in the proposed model Eq. (3.24). In addition, as expected from the proposed model derivation in Section 3.2.2, there is certain value of T_r which defines a boundary point between fast and slow input slew. This point is T_{rb} as shown in Eq. (3.17). For $Tp_{HL,step} = 22.8$ ps at $V_{DD} = 0.7$ V, $V_{th} = 0.26$ V, $\alpha = 1.5$ (extracted from fitting $Id - V_{GS}$ characteristics to the Alpha-power model), and using Eq. (3.17) we find that $T_{rb} = 90.75$ ps which agrees well with the point where the slope of $\sigma_{Tp_{HL}}$ changes abruptly.

Eq. (3.24) shows that $\sigma_{Tp_{HL}}$ is an increasing function of T_r, and $\sigma_{Tp_{HL}}$ is the maximum of two lines intersecting at T_{rb} and therefore, Eq. (3.24) can also be

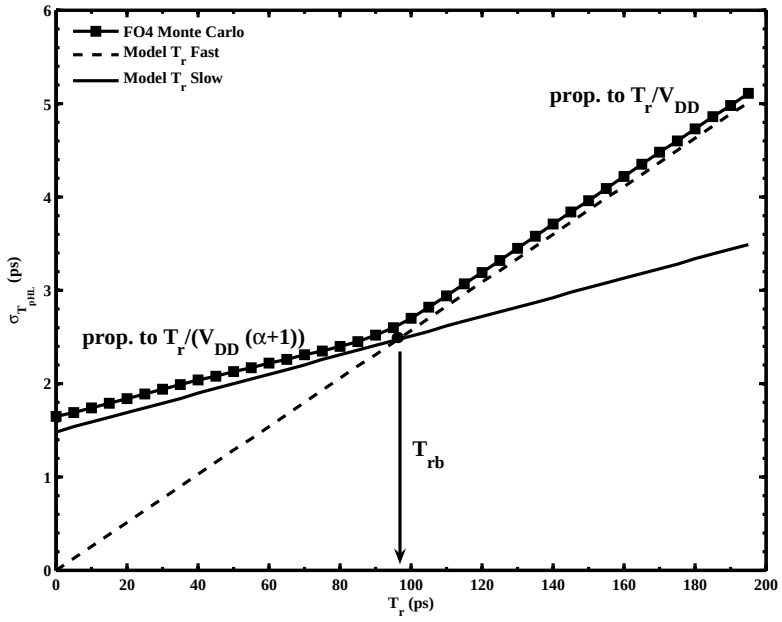


Figure 3.5: Delay variation $\sigma_{Tp_{HL}}$ versus T_r at $V_{DD} = 0.7$ V for FO4 inverter from Monte Carlo simulation. Also shown are the results from the proposed model Eq. (3.24)

.

written as:

$$\sigma_{Tp_{HL}} = max\left\{\left(\frac{T_r}{V_{DD}(\alpha+1)} + T_{pHL,step}\frac{\alpha}{(V_{DD}-V_{th})}\right), \frac{T_r}{V_{DD}}\right\} \times \sigma_{V_{th}} \quad (3.30)$$

where $max\{\}$ is the maximum of the two terms inside the brackets.

It is important to note how the slope of $\sigma_{Tp_{HL}}$ increases significantly when T_r is larger than T_{rb}, as shown in Fig. 3.5. Therefore, as a design guideline, it is useful to always try to reduce T_r for circuits and paths which are sensitive to variability.

In addition to the absolute delay variation $\sigma_{Tp_{HL}}$, it is useful to look at the relative delay variation $\sigma_{Tp_{HL}}/Tp_{HL}$. Fig. 3.6 shows the measured and modeled $\sigma_{Tp_{HL}}/Tp_{HL}$ for FO4 inverter versus T_r at $V_{DD} = 0.7$ V. Good agreement is shown between the measurements and the proposed model. Fig. 3.6 also shows that initially, as T_r increases, $\sigma_{Tp_{HL}}/Tp_{HL}$ reduces and reaches a minimum point and any future increase in T_r increases the relative delay variation. This was expected from our analysis, as derived in Section 3.2.3. Using Eq. (3.27) and substituting for

$V_{th} = 0.26$ V and $\alpha = 1.5$, we find that $V_{DD} = 0.7$ V is smaller than $\frac{2V_{th}}{2-\alpha} \approx 1$ V. Therefore, $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{min}$ is computed as $\frac{2\sigma_{V_{th}}}{V_{DD}} = 2 \times 18\text{mV}/0.7\text{V} = 5.14\%$. This agrees very well with the minimum $\sigma_{Tp_{HL}}/Tp_{HL}$ shown in figure Fig. 3.6. This shows the accuracy of the proposed model.

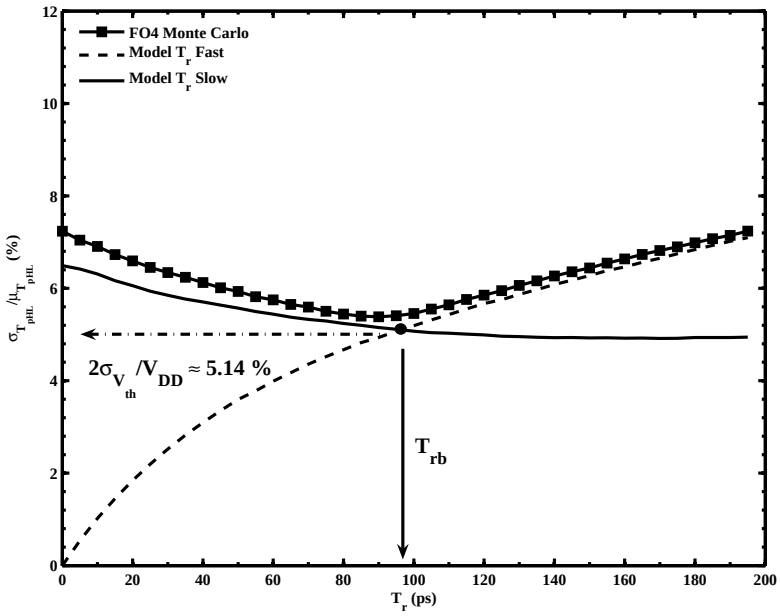


Figure 3.6: Relative delay variation $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ versus T_r at $V_{DD} = 0.7$ V for FO4 inverter from Monte Carlo simulation. Also shown are the results from the proposed model. A minimum point for $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ is shown at T_r =90 ps.

From Fig. 3.6, we see that $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{min}$ occurring at T_{rb} is 25% lower than $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{step}$. Therefore, this minimum point can be used to minimize the relative delay variability by noticing that the optimum T_r value can be converted to a constraint on gate sizing optimization for cascaded stages (*i.e.*, T_r is determined by the driving capability of the driving gate and the capacitive loading, which is the sum of output capacitance of the driving gate and input capacitance of the following gate).

As V_{DD} value is increased above $\frac{2V_{th}}{2-\alpha} \approx 1$ V, Eq. (3.27) shows that the minimum $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ occurs at $T_r = 0$. This is shown in Fig. 3.7, where $\sigma_{Tp_{HL}}/Tp_{HL}$ versus T_r is shown for $V_{DD} = 1.0$ V. Good agreement is shown between the proposed model and the measurement results. While at this supply voltage, the increase in $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ is very slow when $T_r < T_{rb}$. However, as T_r exceeds T_{rb}, the increase in $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ is significant. Relative delay variation increases ∼2X when T_r increase from T_{rb} to $3T_{rb}$. From the above discussion, we can say that by trying to constrain T_r values to be approximately equal to T_{rb}, we can either achieve the minimum relative delay

variation (for $V_{th} < V_{DD} < \frac{2V_{th}}{2-\alpha}$) or we will ensure that $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ is not in the range which is strongly dependent on T_r (for $V_{DD} > \frac{2V_{th}}{2-\alpha}$ as shown in Fig. 3.7).

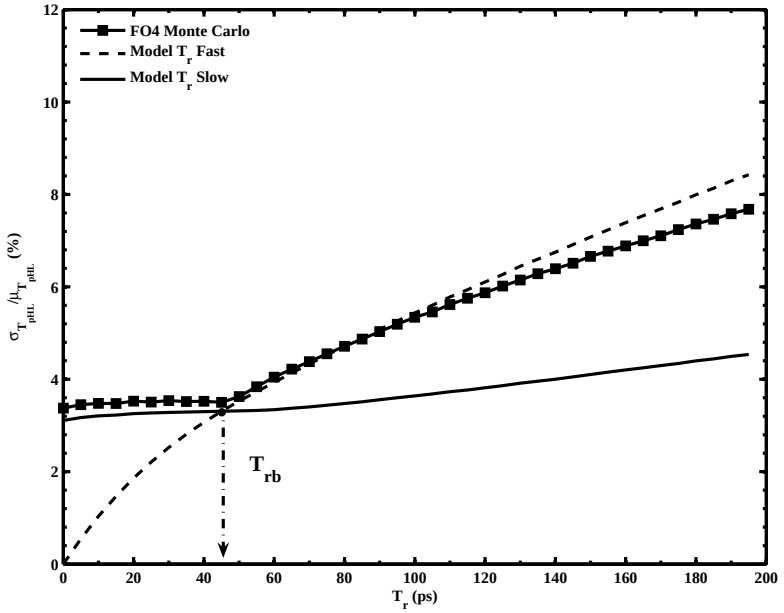


Figure 3.7: Relative delay variation $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ versus T_r at $V_{DD} = 1.0$ V for FO4 inverter from Monte Carlo simulation. Also shown are the results from the proposed model.

Fig. 3.8 shows how changing FO affects $\sigma_{Tp_{HL}}$. For each FO, there are different values of T_{rb}, since T_{rb} is proportional to $Tp_{HL,step}$ as shown in Eq. (3.17). It is clear that when T_r is slow ($T_r > T_{rb}|_{FO}$), the value of $\sigma_{Tp_{HL}}$ becomes independent of FO. This is because for slow T_r values, $\sigma_{Tp_{HL}}$ is expressed as $\frac{T_r}{V_{DD}}\sigma_{V_{th}}$ as shown in Eq. (3.24), which is independent of the gate's FO. However, for fast input slew ($T_r < T_{rb}$), increasing the FO directly translates to higher delay variation, which is also predicted by the proposed model. The relative delay variation trends for different FOs are shown in Fig. 3.9.

In Fig. 3.10, $\sigma_{Tp_{HL}}$ from our model is plotted versus Monte Carlo simulation results for different loading (FO1, FO2, FO4) and V_{DD} (0.7 V, 1 V) conditions. The maximum error is 10.3% and the average error is 3.5%. Good agreement between our model and Monte Carlo simulations results justifies the assumptions used to derive the model as explained in Section 3.2.

In Section 3.2.4, we showed that the T_r variation affects delay variation of the gate, as shown by Eq. (3.28) and Eq. (3.29). Fig. 3.11 shows how T_r variation affects $\sigma_{Tp_{HL}}$. T_r variation increases delay variation as shown in the figure. For small values of T_r variations, we see that there exists a minimum relative delay

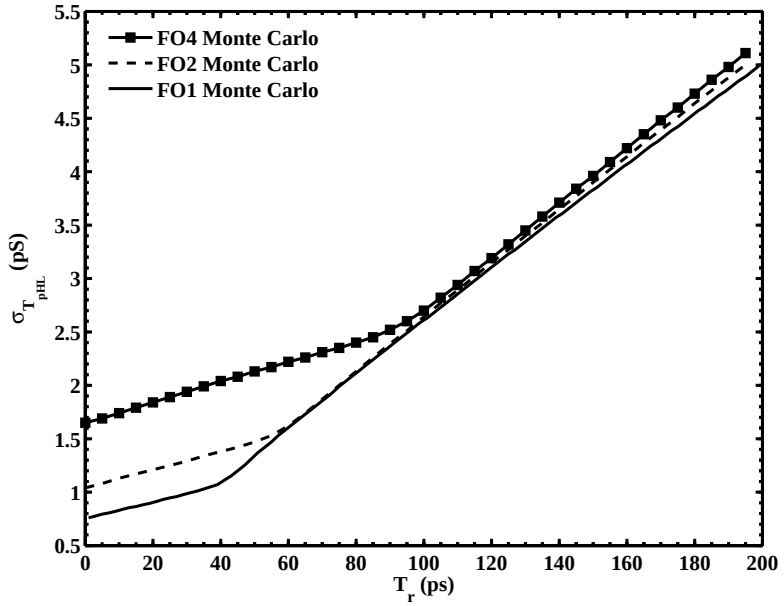


Figure 3.8: $\sigma_{Tp_{HL}}$ versus T_r at $V_{DD} = 0.7$ V for different loading conditions (FO1, FO2 and FO4).

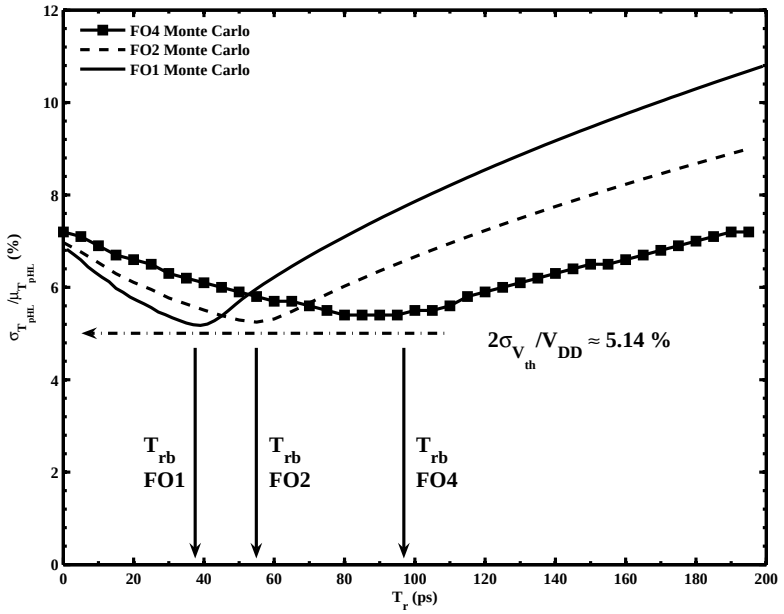


Figure 3.9: $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$ versus T_r at $V_{DD} = 0.7$ V for different loading conditions (FO1, FO2 and FO4).

variation point, as shown for the cases of $\frac{\sigma}{\mu}|_{T_r}$ =3% and $\frac{\sigma}{\mu}|_{T_r}$ =6%. However, as $\frac{\sigma}{\mu}|_{T_r}$ increases, its impact on delay variation increases, hence it can dominate the total delay variations trend. This can be seen in the case of $\frac{\sigma}{\mu}|_{T_r}$ =9%, where the

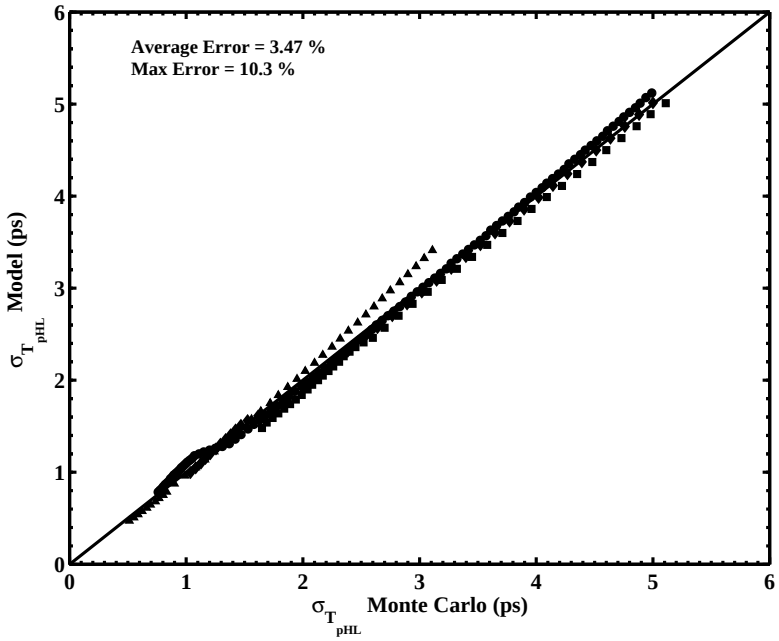


Figure 3.10: $\sigma_{Tp_{HL}}$ from our proposed models Eq. (3.24) versus Monte Carlo simulation results for different V_{DD} and loading (FO1, FO2 and FO4) conditions.

minimum relative delay variation point moves towards lower T_r. Further increase in $\frac{\sigma}{\mu}|_{T_r}$ causes T_r variation to dominate the delay variation, hence, the minimum point disappears and $\frac{\sigma_{Tp_{HL,tot}}}{Tp_{HL}}$ shows monotonic increase versus T_r. However, it is important to note that T_r variation depends on the previous gate driving capability ($\frac{\sigma}{\mu}|_{T_r}$ will be proportional to $\frac{\sigma}{\mu}|I_p$ of the previous gate). Therefore, $\frac{\sigma}{\mu}|_{T_r}$ should be in the range of 6-7% at $V_{DD} = 0.7$ V. Hence, our models are valid within a practical range of $\frac{\sigma}{\mu}|_{T_r}$.

As was shown in the previous discussions, the proposed delay variation model is based on easily measurable parameters, which can be directly extracted from measurement or from simulation (*i.e.*, DC simulation for g_m/I_D and transient simulation for T_p) as well as from technology information (*i.e.*, $\sigma_{V_{th}}$, V_{th}, α). In addition, our delay variation model is also very simple and efficient (compared to Monte Carlo simulation, which is computationally intensive).

The model can be used to explore different design tradeoffs to reduce delay variability. Moreover, the model can be used to apply constraints on T_r when optimizing gates size in order to minimize delay variation.

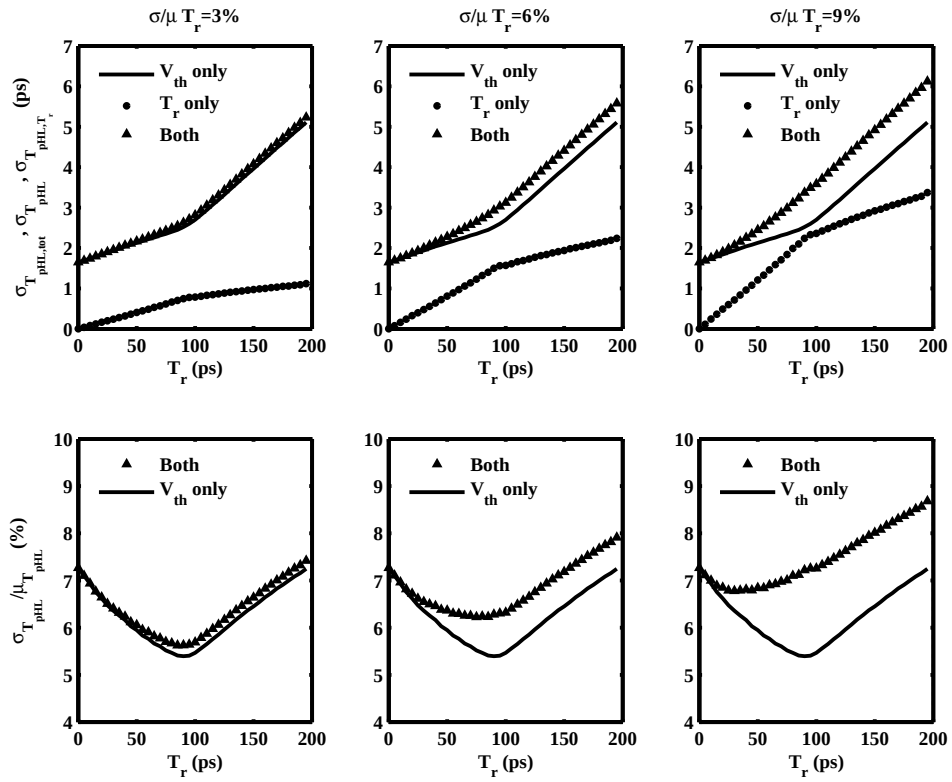


Figure 3.11: Impact of T_r variation on delay variation at $V_{DD} = 0.7$ V for FO4 inverter. Different values of $\frac{\sigma}{\mu}|_{T_r}$ are shown.

3.4 Design Insights

In this section, we look at some of the design insights and implications that can be identified from the proposed models in this work. The simple forms of Eq. (3.24) and Eq. (3.27) allow us to have clear insights for several design issues as discussed below:

1. Input rise time has a strong impact on delay variation $\sigma_{Tp_{HL}}$, and the sensitivity of delay variation to T_r has two different sensitivities depending on whether T_r is fast or slow (*i.e.*, $T_r < T_{rb}$ or $T_r > T_{rb}$, respectively). This is shown in the model as well as verified using Monte Carlo simulation (as was shown in Fig. 3.5). Therefore, to minimize delay variation, it is important to reduce T_r in the design, which can be achieved using adequate sizing. This condition is similar to the conditions required to reduce short-circuit power consumption by optimizing the input rise time of the gates [94].

2. For a fast input rise time ($T_r < T_{rb}$), $\sigma_{Tp_{HL}}$ is proportional to $T_r/(1+\alpha)$. However, as input rise time is increased exceeding T_{rb}, delay variation increases proportional to T_r. Therefore, for large T_r, delay variation sensitivity to input rise time is 2.5X larger than the fast input rise time case (assuming a typical value of $\alpha \approx 1.5$ in deep submicron technologies). This shows the importance of accounting for delay variation especially for slow input rise times.

3. For slow input rise time, delay variation is independent of the step input delay variation or the output loading of the gate. In fact, $\sigma_{Tp_{HL}}$ is only function of V_{th} variation through $\sigma_{V_{th}}/V_{DD}$ term in Eq. (3.24), as well as on the input rise time T_r which is determined by the ratio of preceding gate driving capability to input capacitance of this gate. This is also confirmed using the simulation results in Fig. 3.8, where $\sigma_{Tp_{HL}}$ is identical for inverters having different FOs when T_r is large.

4. $\sigma_{Tp_{HL,step}}$ is the minimum delay variation for a gate and can only be achieved for very fast input slope (*i.e.*, step input with $T_r \rightarrow 0$). However, in reality, delay variation will always be larger than the value predicted by $\sigma_{Tp_{HL,step}}$, especially at large values of T_r, and can be an order of magnitude larger than $\sigma_{Tp_{HL,step}}$.

5. Variation in V_{th} increases delay variation linearly. With the significant increase in V_{th} variability with scaling, similar increase in a gate's delay variation is expected, and is exacerbated with the reduction of V_{DD}.

6. Depending on the value of the supply voltage, there exists a minimum relative delay variation $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{min}$, which can be calculated using Eq. (3.27). As V_{DD} is reduced below $2V_{th}/(2-\alpha)$, $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{min} = 2\sigma_{V_{th}}/V_{DD}$ and occurs when input rise time equals T_{rb}. This finding can be used to reduce the relative delay variation of a path by adding T_r as one of the parameters that can be used to reduce $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}$. It is important to note that $T_{rb}/Tp_{HL,step}$ increases as V_{DD} is reduced as shown in Eq. (3.17). This implies that the optimum FO to achieve $\frac{\sigma_{Tp_{HL}}}{Tp_{HL}}|_{min}$ increases as V_{DD} is reduced.

7. T_r variation from previous stage increases the delay variation for the current stage. In addition, T_r variation affects the minimum relative delay variation point and moves it towards lower T_r values. For excessively large T_r variation, the minimum relative delay variation point disappears since the gate delay variation becomes dominated by the input slew variation effect.

While this work has focused on delay variation of a single stage (inverter), it can be extended to model path delay variation. In that case, the input T_r used in the proposed model can be computed from the output slew of the preceding gate. Therefore, the delay variation of each gate in a path can be computed using Eq. (3.24) and the total delay variation can be computed using Eq. (3.29) [23].

Process variations in the interconnect lead to gate delay variations. However, interconnect capacitance variation impact on delay variation is much smaller than drive current variation [3, 88]. In addition, the negative correlation between interconnect resistance and capacitance further reduce the impact of RC delay variation [95]. Therefore, it is fair to neglect interconnect variation in this work, since the objective focus is to derive simple models that provide design insights. Nevertheless, the proposed models can be easily extended to include capacitance variation, since interconnect capacitance shows no bias dependence and appears as a multiplicative term in delay calculation.

In this research, we focus on WID variations because, from a circuit point of view, WID variations are much more difficult and complex to model compared to die-to-die (D2D) or global variations. Accounting for D2D variations can be easily performed using corner models, however, WID variations require accounting for delay variation in each gate differently. Our models facilitate this type of WID delay variation analysis. Corner based D2D simulation can be combined with our WID models to predict the delay variation of the path without going through Monte Carlo simulation. Our models for delay variation are not limited to WID variations

in the sense that they can handle process variations in general. However, since WID variations are typically smaller in magnitude compared to D2D variations, therefore, the linear assumption described in Section 3.2 is valid for the case of WID. But, its validity may be questioned in the case of D2D variations. Therefore, it is more accurate to use D2D based corner model simulation and combine it with our WID models to find the impact of total variations (WID and D2D) on delay.

It is important to note that the key for reaching our simple models is that we focus on the accuracy of *delay variation*, while in previous works [91, 92] the focus was on the accuracy of *delay* itself. In our models, as show in Section 3.2.2, several simplifications were made in order to reach the simple closed form shown in Eq. (3.24) and Eq. (3.27). To simplify the analysis, we neglect effects that are not function of V_{th} of the active device (NMOS in the case of High-to-Low transition), while we completely understand their strong impact on delay. Miller effect, PMOS current and capacitance bias dependence strongly affect Tp_{HL} of the gate. Nevertheless, to a first order, these effects do not change $\sigma_{Tp_{HL}}$. This was also confirmed by the good accuracy of our models as shown in Section 3.3. Therefore, neglecting these effects does not limit the accuracy of our models. Instead, these assumptions allow us to reach the simple form of delay variation which is attractive for design.

3.5 Summary

The increase in WID process variations in nanometer technologies has a strong impact on delay variability. In this chapter, we presented analytical derivation for statistical delay variation model in the presence of WID statistical variations. The accuracy of the proposed models has been validated with Monte Carlo SPICE simulation results for an industrial 90nm technology over a wide range of supply voltages, input slew and output loading. Using the derived model, we showed that input slew has a strong effect on delay variation. In addition, slow input slews increase delay variation significantly compared to fast input slews where the boundary between fast and slow input slew can be estimated as shown in Eq. (3.17). In addition, for slow input slew, delay variation is simply $\frac{T_r}{V_{DD}}\,\sigma_{V_{th}}$, which is independent of the gate's electrical properties (*i.e.*, driving capability) or loading. It was also shown that as supply voltage is reduced below $\frac{2V_{th}}{2-\alpha}$, there is an optimum value of input slew that achieves the minimum relative delay variation expressed as $\frac{\sigma_{Tp}}{Tp} = \frac{2\sigma_{V_{th}}}{V_{DD}}$. This finding is important in designing clock distribution network where the objec-

tive is to minimize skew in different branches of the distribution networks and for self-timed paths used in memory timing.

The derived statistical models are simple, scalable, bias dependent and only require the knowledge of easily measurable parameters. In addition, the models are also very efficient compared to Monte Carlo simulations. This makes them useful in early design exploration, circuit/architecture optimization as well as technology prediction.

Chapter 4

A Methodology for Statistical Estimation of Read Access Yield in SRAMs

The increase of process variations in advanced CMOS technologies is considered one of the biggest challenges for SRAM designers. This is aggravated by the strong demand for lower cost and power consumption, higher performance and density which complicates SRAM design process. In this chapter, we present a methodology for statistical simulation of SRAM read access yield, which is tightly related to SRAM performance and power consumption. The proposed flow enables early SRAM yield predication and performance optimization in the design time, which is important for SRAM in nanometer technologies. Section 4.1 introduces the problem of SRAM yield estimation, and in section Section 4.2, some challenges for statistical SRAM design are described. In Section 4.3, we describe and model the dominant sources that affect SRAM read operation and increases read failures. In Section 4.4, the proposed read failure definition is presented, and the yield estimation flow is explained. In Section 4.5, the proposed methodology is verified using measured silicon yield data from a 1Mb memory fabricated in an industrial 45nm technology. Finally, in Section 4.6, we summarize our findings.

4.1 Introduction

Random variations in nanometer ranges technologies are considered one of the largest design considerations [3,5]. This is especially true for SRAM memories, due

to the large variations in bitcell characteristics. Typically, SRAM bitcells have the smallest device sizes on a chip. Therefore, they show the largest sensitivity to different sources of variations - such as random dopant fluctuations (RDF), line-edge roughness (LER) and others [14, 96]. While variations in logic circuits have been shown to cause delay spread [17, 84] which reduces parametric yield, for SRAMs, process variations also cause the memory to functionally fail, which reduces the chip's functional yield. With lower supply voltages and higher variations, statistical simulation methodologies become imperative to estimate memory yield and optimize performance and power.

As explained in Section 2.8, there are different types of SRAM bitcell failure mechanisms, such as static noise margin stability fails (cell may flip when accessed), write fails (bitcell cannot be written within the write window), read access fails (incorrect read operation), and retention fails [12–14]. In this work, we concentrate on estimating yield loss due to read access failures, as this type of failure has a strong impact on determining performance and power consumption of the memory. Moreover, it has been shown that read access failure is the dominant failure mechanism at normal operating conditions [13].

Recently, there have been few works in the area of SRAM design methodologies. In [14], the authors present a worst-case analysis to account for weak cells and presented guidelines for SRAM timing to achieve high yield. In [12, 13] the authors model access failures by statistically accounting for bitcell read current variation as well as for the impact of access transistor leakage [13]. These previous works have focused on determining memory yield for a given sense amplifier (SA) offset (*i.e.*, estimating access yield for a fixed value of bitlines differential voltage), which implies that worst-case analysis is assumed for the SA offset, although statistical analysis is used for bitcell read current variations.

In this chapter, we generalize the access failures to "statistically" include the SA offset distribution. This is important for SRAM circuit designers as it reduces the pessimism of assuming worst-case SA offset and worst-case bitcell. In addition, for the first time, we include the impact of sensing window variation on yield, which can have a strong impact on memory performance. The proposed statistical yield estimation methodology for access failures accounts for bitcell read current variations, sense amplifier offset, and sensing window variations, as well as leakage from other bitcells on the same column. In particular, the proposed methodology helps answer the following questions for SRAM designers:

1. What is the maximum achieved performance (minimum sensing window) for

a given yield requirement;

2. How much is the achievable improvement in yield if SA offset is improved by a certain amount (*i.e.*, increasing SA area or changing SA topology);

3. How to compare the expected yield for memories having similar densities but different architectures (*i.e.*, yield for different memory options).

4.2 Challenges of SRAM Statistical Design

The read path in SRAM memories is typically a part of the critical path, which determines the memory access time (performance) [97]. Fig. 4.2 shows read path in an SRAM memory, which consists of array of bitcells accessed using a shared sense amplifier (SA). Each column of bitlines is selected using a column select multiplexer depending on the input addresses. Prior to selecting row and columns, the bitlines (BL and BLB) and sense lines (SL and SLB) are precharged to V_{DD}. Read operation begins by selecting the column using the PMOS pass-gate and activating the wordline (WL) of the selected row, as shown in Fig. 4.3. Depending on the stored data in the bitcell, one side of the bitlines begins to discharge the bitline capacitance using the bitcell read current (I_{read}). Therefore, a small differential voltage is generated at the inputs of the voltage sense amplifier (V_{SAin}). To ensure correct read operation, the SA is enabled using a control signal (SAEN) after a sufficient differential signal V_{SAin} is developed, which is amplified by the SA to a digital output level.

The delay difference between the WL activated and the SA enabled is called "read sensing window" ($t_{wl2saen}$), as shown in Fig. 4.3. $t_{wl2saen}$ has direct impact on the memory performance as it contributes a large percentage of the memory access time ($\sim$ 30%) [73]. In addition, $t_{wl2saen}$ has direct impact on the dynamic power consumption. As $t_{wl2saen}$ increases, the bitlines differential increases, which should be recovered by the precharge circuitry after each memory access cycle [1]. In the meantime, increasing $t_{wl2saen}$ increases V_{SAin}, which reduces the probability of read failure due to SA input offset. Hence, it is always desirable to reduce $t_{wl2saen}$ as long as correct read operation is ensured. Therefore, there is a strong tradeoff between yield and performance/power for SRAM, which is one of the most important design decisions for memory designers.

[1] More discussions on this type of power consumption will be presented in Chapter 5.

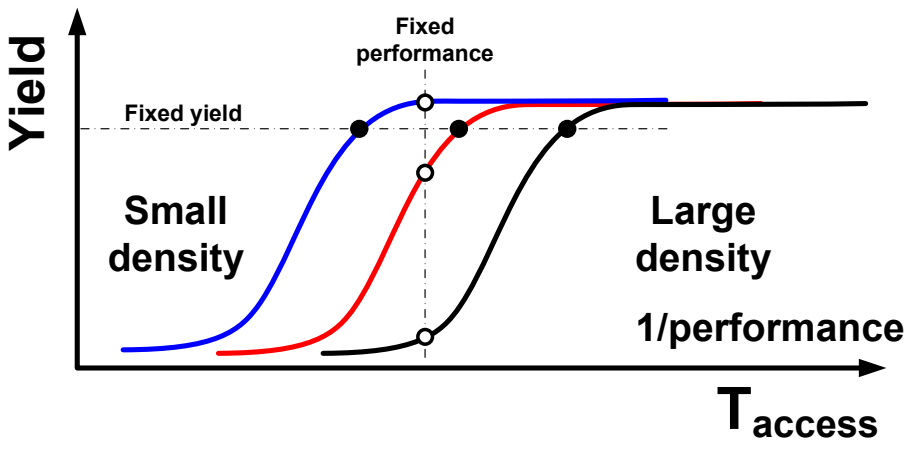


Figure 4.1: Yield versus performance tradeoff for SRAM design.

Fig. 4.1 shows typical yield[2] versus memory access time which is inversely proportional to the memory's maximum frequency (performance) for a given memory design and architecture. As the access time is shortened, by reducing $t_{wl2saen}$, memory yield drops. This is because as $t_{wl2saen}$ reduces, the SA input differential V_{SAin} reduces causing the sense amplifier to sense the stored data incorrectly. Moreover, memory density has strong impact on the yield v. performance tradeoff. As the memory density increases (*e.g.*, by using multiple instances of the same small memory macro), the whole curve shifts towards higher access time. Therefore, to achieve the same yield target similar to the smaller density, a larger memory will require larger access time. If the performance is fixed, memory yield reduces for higher densities. This type of yield v. performance tradeoff is very critical for memory design, and requires statistical analysis and approaches.

The statistical nature of SRAM failures requires statistical simulation techniques in order to account for these failure mechanisms early in the design cycle. Unfortunately, the problem of statistical design from memories is aggravated by circuit simulation speed and capacity limitations. Due to the large size of SRAM memories, it is very difficult to run Monte Carlo simulations for the whole memory. Even if the computational resources allow Monte Carlo simulation for the whole memory, a large number of Monte Carlo runs is required. For example, more than 2×10^6 Monte Carlo runs are required to examine for one failure in a 2Mb memory, due to the rare event of having read current a weak bitcell exceeding 5σ of bitcell variations. Therefore, SRAM designers typically use worst-case approaches to ensure high yield by designing for the worst-case bitcell for a given memory density [14].

[2]This chapter focuses on read access yield. Therefore, we use the word yield to refer to read access yield.

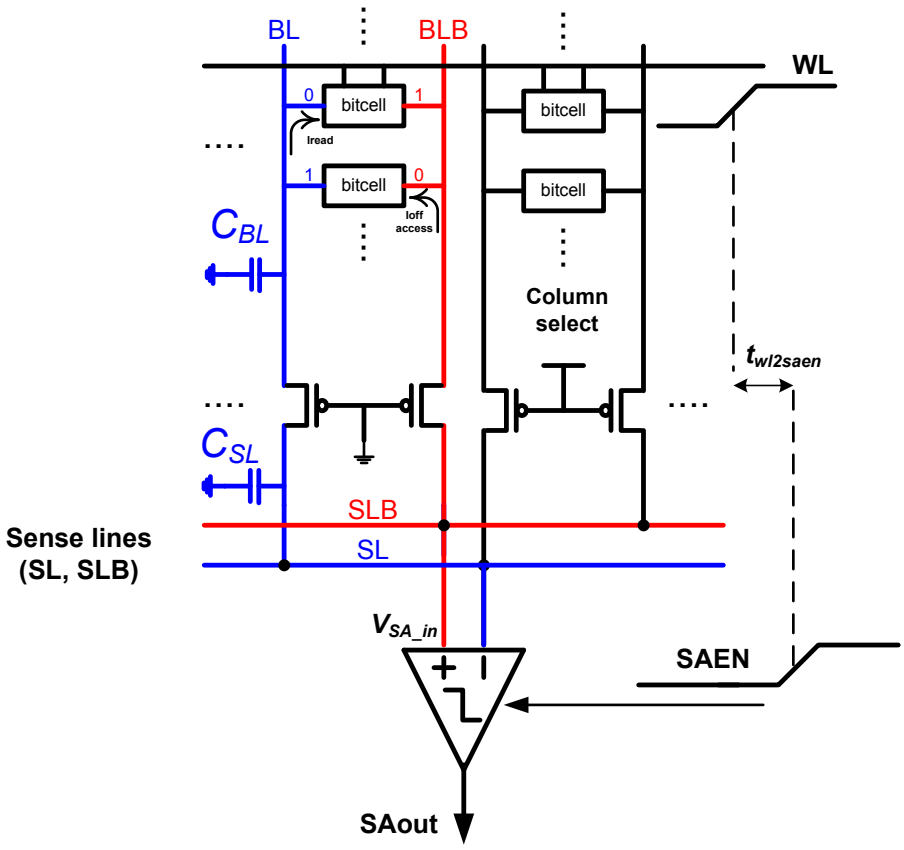


Figure 4.2: Simplified SRAM read path.

However, this worst-case design technique negatively impacts the performance as well as increases power consumption. In addition, worst-case approaches show no clear relation between yield and performance.

Previous works in the area of statistical design for memories define a successful read access as the probability of having the bitlines reach a **fixed** voltage Δ_{min} for a **fixed** access (sensing) window $t_{wl2saen}$ [12, 13]. In [12, 13], although statistical analysis is performed on I_{read}, however, by assuming fixed Δ_{min} and $t_{wl2saen}$, this means that worst-case is assumed for the sense amplifiers as well as for the sensing window. In addition, previous models assumed that BL discharge could be coupled directly to the SA inputs. However, in reality, due to the on resistance of PMOS column select device, the sense line is usually slower than the bitline discharge, and longer time is required to achieve certain differential voltage [97]. Hence, the above mentioned techniques are more appropriate for bitcell technology optimization, while a new access failure estimation methodology is required for memory circuit design that can account for different sources of access failures in a single statistical yield estimation flow.

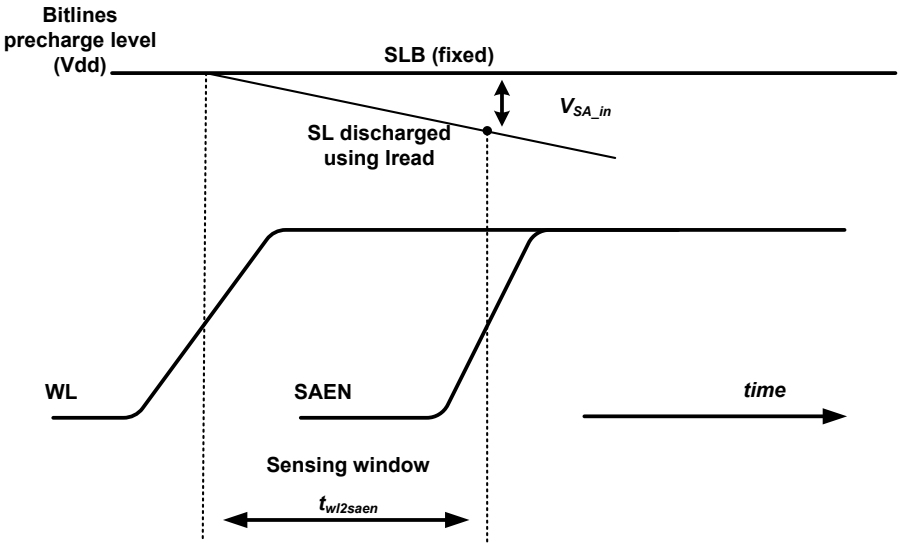


Figure 4.3: Timing diagram for SRAM read operation.

In the following sections, we go through the different sources that affect access failures. Following that, we present a new read failure definition that is used in the proposed flow.

4.3 Modeling of Read Access Failures

As discussed in the previous section, the industry standard worst-case analysis for SRAM design limits the archived performance and increases pessimism. In the meantime, full statistical approach using circuit simulation is not an option due to practical limitations (simulation speed and capacity). In this work, we propose using full statistical for the whole read path instead of worst-case analysis. To overcome the practical limitations of using circuit simulation, instead, we use simple (but accurate enough) behavioral modeling for read access failure, as will be described later. The emphasis here is on the behavioral model's simplicity. A simple model can significantly improve simulation efficiency, since it can be used to run extensive Monte Carlo simulation. However, to derive a simple model for SRAM read failure, it is important to capture the dominant sources that affect read operation.

There are four major contributors for read access failures in SRAM and they are all strongly affected process variations, as shown in Fig 4.2:

1. Bitcell read current variation;

2. sense amplifier input offset;

3. sensing window delay variation;

4. pass-gate (access transistor) leakage.

4.3.1 Read Current and Sensing Slope Variations

Due to the small size of SRAM bitcell and the inverse relation between transistor variation and device area [24, 93], bitcell read current I_{read} shows large within die (WID) variations [13, 96], and typically follows a normal distribution. From a memory design point of view, I_{read} determines the time required to develop enough differential signal before enabling the SA. I_{read} variation is considered one of the largest sources of parametric yield loss in memories [13].

As mentioned earlier, sense lines are discharged using the bitcell I_{read}. However, sense lines discharge rate is slower than bitlines due to the ON resistance of the column select device (PMOS) shown in Fig. 4.2, which adds RC delay at the SA input [97]. Let's define the sense lines discharge slope as $K_{eff} = |\Delta V_{SL}/\Delta t|$. It can be shown that K_{eff} is proportional to I_{read} [97]. The statistical variation in I_{read} will also cause similar variation in K_{eff}, therefore, $\frac{\sigma}{\mu}|_{K_{eff}} = \frac{\sigma}{\mu}|_{I_{read}}$.

Due to random variations in bitcells transistors, it is important to note that each bitcell has two values of I_{read} currents depending on the stored data (whether it is 0 or 1). This is shown in Fig. 4.4 where I_{read} path is different for the case of read 0 or read 1. This difference in I_{read} values occurs since each transistor in the bitcell experiences different value of random WID variation. Therefore, I_{read} for read 0 and read 1 cases are statistically independent.

4.3.2 Sense Amplifier Variations

Sense Amplifier (SA) is typically used to amplify the small differential voltage on the bitlines ($\sim$100mv) to a digital output level [14]. Fig. 4.5 shows one of the most widely used SA in memory design due to its fast decision time. This SA is also called decoupled SA (DSA) since the inputs and outputs of the SA are separated [98, 99]. The decision threshold of the SA is ideally zero. That is, if $V_{SAin} > 0$, the output is high, and vice versa, if $V_{SAin} < 0$. The amount by which the threshold point shifts is called the input offset [100].

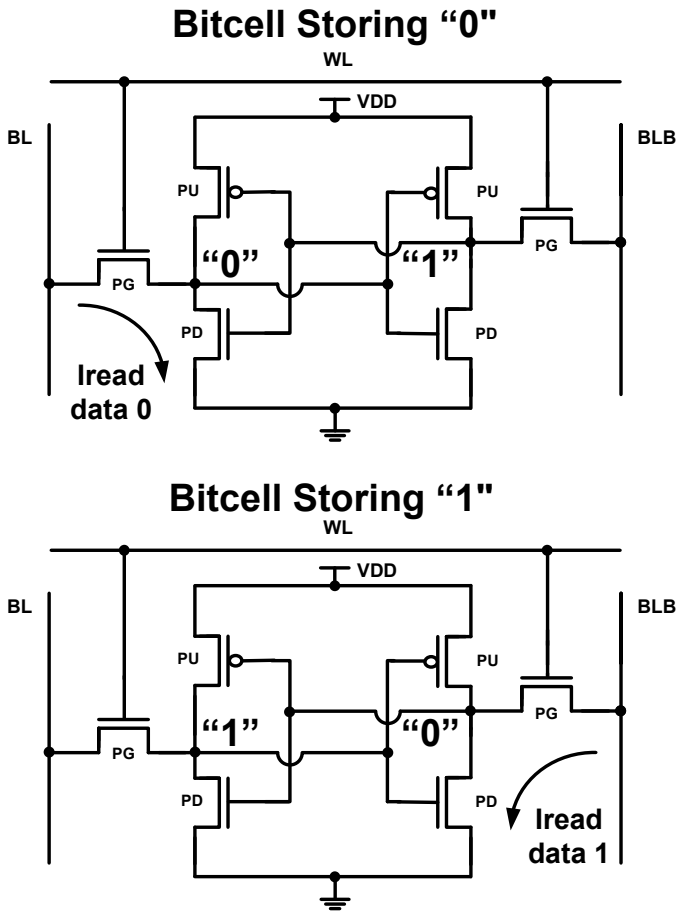


Figure 4.4: 6T bitcell showing the different read current paths in the case of read 0 and read 1.

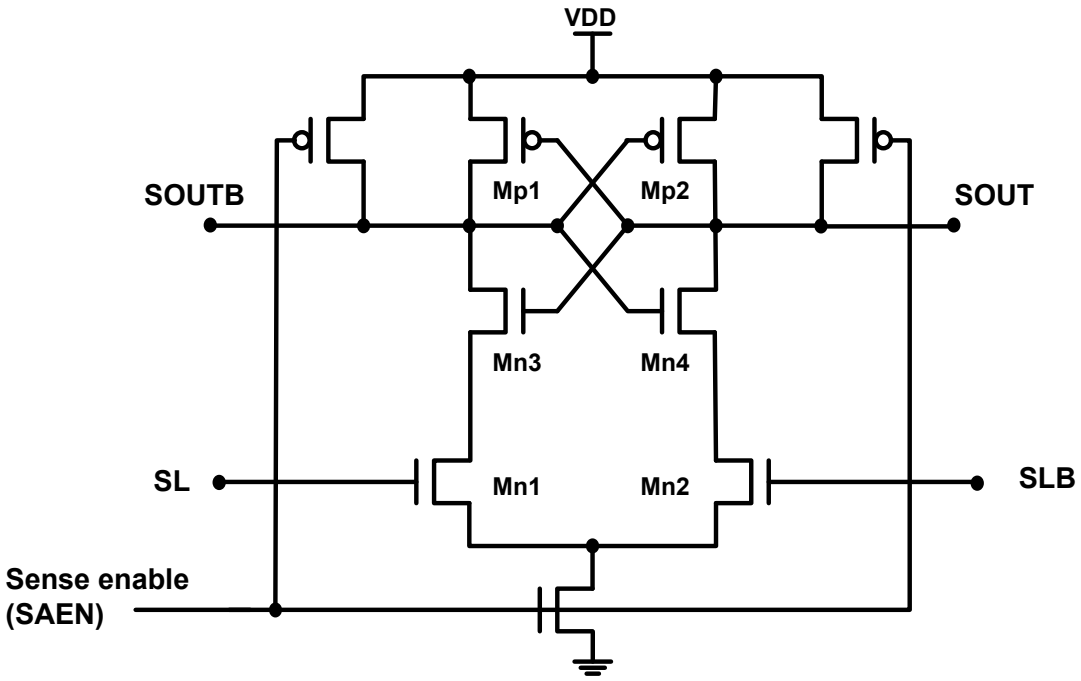


Figure 4.5: Current latch sense amplifier (CLSA) [98, 99].

SAs are extremely sensitive to WID variations (mismatch) [34, 99, 101], which cause SAs to show offset voltages that affect the accuracy of read operation. In addition, systematic variations due to asymmetric layout can increase the SA input offset. One way to reduce SA's input offset is to increase the size of input devices [93, 99]. Due to the strict limitations on area in memory design, the SA area-mismatch tradeoff is difficult because the SA should pitch-match the accessed bitcells. Therefore, the specification on SA offset is an important metric for memory designers.

Monte Carlo transient simulation is usually used to estimate the input offset distribution of SA [14]. Typically, the SA input offset can be modeled using a Gaussian distribution with a mean of ∼zero and standard deviation of $\sigma_{V_{SAoffset}}$ as shown in Fig. 4.6. The SA input offset can be modeled as a noise voltage source connected at the input of and ideal SA, as shown in Fig. 4.7, where the voltage source follows the normal distribution of the $V_{SAoffset}$.

As mentioned in Section 4.3.1, I_{read} for read 0 and read 1 cases are statistically independent. However, a shared SA is used to sense both states as shown in Fig. 4.2. Therefore, two distributions are used to model the SA input voltage (proportional to I_{read}), while one distribution is used to model the SA input offset ($V_{SAoffset}$) as shown in Fig. 4.6. In a worst-case design scenario, the minimum sensing voltage V_{SAin} is required to be larger than the worst-case SA offset (as shown in Fig. 4.6).

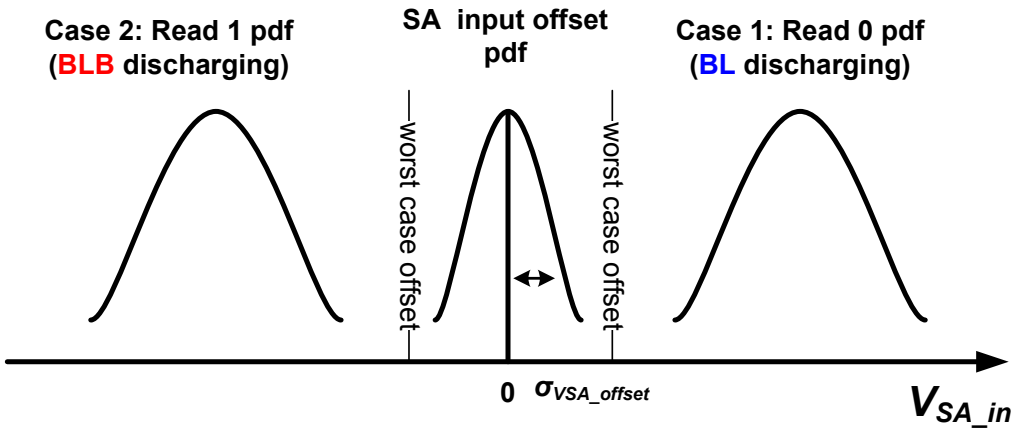


Figure 4.6: SA input offset and read 0/1 distributions.

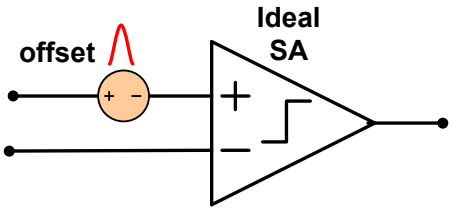


Figure 4.7: Modeling of SA input offset as a noise source connected at the input terminal of SA.

This is a pessimistic approach because the probability of having the slowest bitcell accessed using the SA suffering the largest offset is very small.

Another source that can increase read access failures is dynamic noise coupling at the SA inputs. Due to the small differential signal developed on the sense lines, an aggressor located near the SA may couple large noise at the SA input which can affect the accuracy of read operation. This situation is exacerbated if a weak bitcell is selected, the read sensing window is reduced, or if the noise occurs just before the SA is enabled. However, modeling of this dynamic noise component is very complex, as it strongly depends on the layout of the SA and sense lines, as well as the timing of the aggressors relative to the SA enable signal (SAEN). Nevertheless, by using layout noise shielding techniques and highly symmetric SA layout styles, the impact of this component can be minimized.

4.3.3 Sensing Window Variations

As mentioned earlier, the read sensing window $t_{wl2saen}$ is an important parameter for correct read operation. In memory design, a centralized control block (timer

circuit) is used to generate the timing for all the critical signals for memory operation - which include WL and SAEN signals [14]. To ensure good tracking with PVT variations, usually similar transistor sizes are used in the two logic paths [14, 102]. However, due to random WID variations, the delay in these paths will show statistical variation [23, 84, 86]. Therefore, the sensing window will have spread around its mean value (as shown in Fig. 4.9).

We have seen in Chapter 3 how WID variations affect gate delay variations. Here we discuss the impact of delay variation in the WL and SAEN paths. Let's assume that the number of logic stages between internal CLK to WL and SAEN is m and n stages, respectively, as shown in Fig. 4.8. For sake of simplicity, let's further assume that delay of each path can be modeled as a chain of inverters, with t_d being the delay of one stage. In an ideal scenario with no random WID variations, $t_{wl2saen}$ can be computed as $(n-m)t_d$. However, due to uncorrelated random variations, $t_{wl2saen}$ will have a statistical distribution, which is typically assumed Gaussian [17, 84, 86]. Therefore, the mean and variance of $t_{wl2saen}$ can be computed as $\mu_{t_{wl2saen}} = (n-m)\mu_{t_d}$ and ${\sigma^2}_{t_{wl2saen}} = (n^2+m^2){\sigma^2}_{t_d}$, respectively, where ${\sigma^2}_{t_d}$ is the variance of one delay stage. In the case of memories, n and m are comparable, where $n-m$ determines the nominal $t_{wl2saen}$. However, there is a large spread in the sensing window since the spread in each logic path adds up to the $t_{wl2saen}$ variation (n^2+m^2 term). Note also that the spread $\sigma_{t_{wl2saen}}$ increases as n and m increase even if $n-m$ is constant (*i.e.*, for a fixed $t_{wl2saen}$ delay). This implies that as the memory size increases and more logic stages are required in the CLK to WL and SAEN paths, this effect becomes more severe. This variation in sensing window can reduce the SA input voltage, which increases access failure probability - especially at low supply voltages (since $\frac{\sigma}{\mu}|_{t_{wl2saen}}$ increases due to reduced headroom as described in Chapter 3 [84]).

While a chain of inverters can be used to qualitatively explain the importance of accounting for read window variations, a more comprehensive delay variation analysis is required to account for different logic gates, input slews and fanouts in the CLK to WL and CLK to SAEN paths, as was described in Chapter 3. In this work, we use Monte Carlo simulation to determine $\mu_{t_{wl2saen}}$ and $\sigma_{t_{wl2saen}}$. Nevertheless, statistical timing analysis [17] can also be used for the same purpose.

4.3.4 Pass-Gate Leakage

It is well known that bitcell pass-gate (access) device leakage also reduces the SA input differential due to subthreshold leakage from the other side of the bitlines [13].

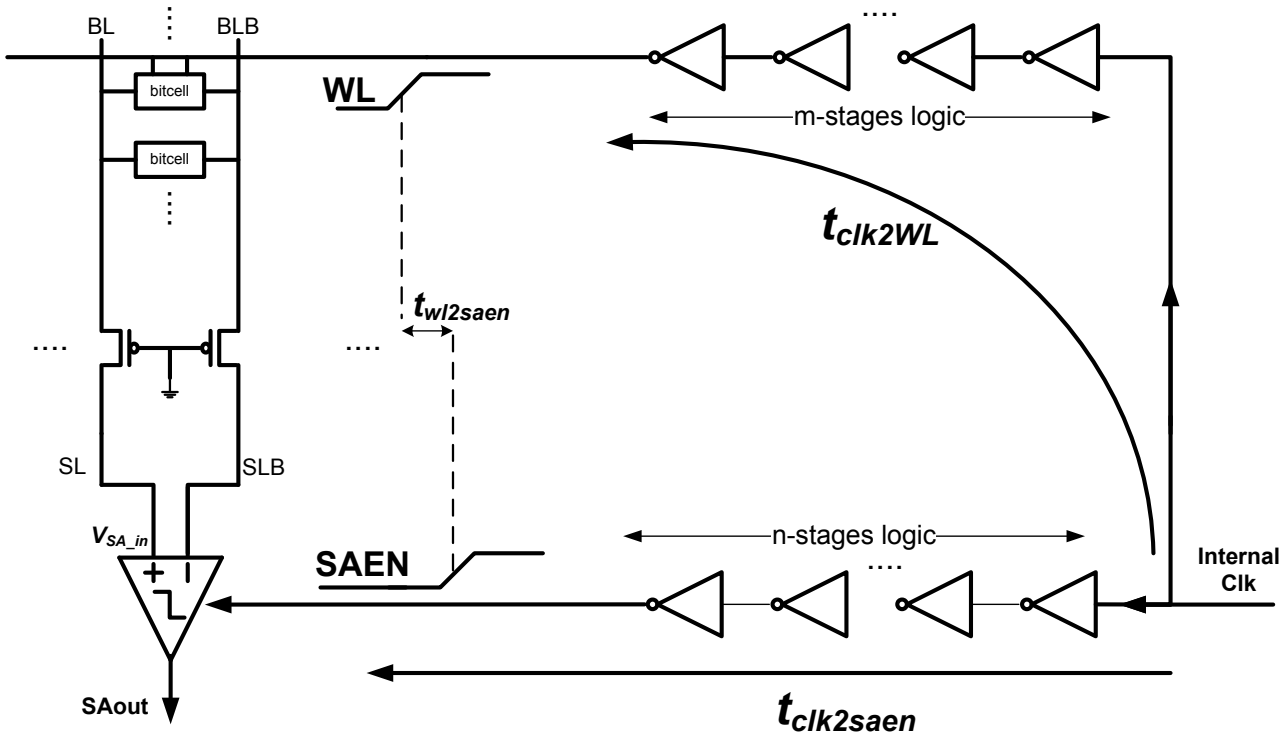


Figure 4.8: Timing delay variation between WL and SAEN paths.

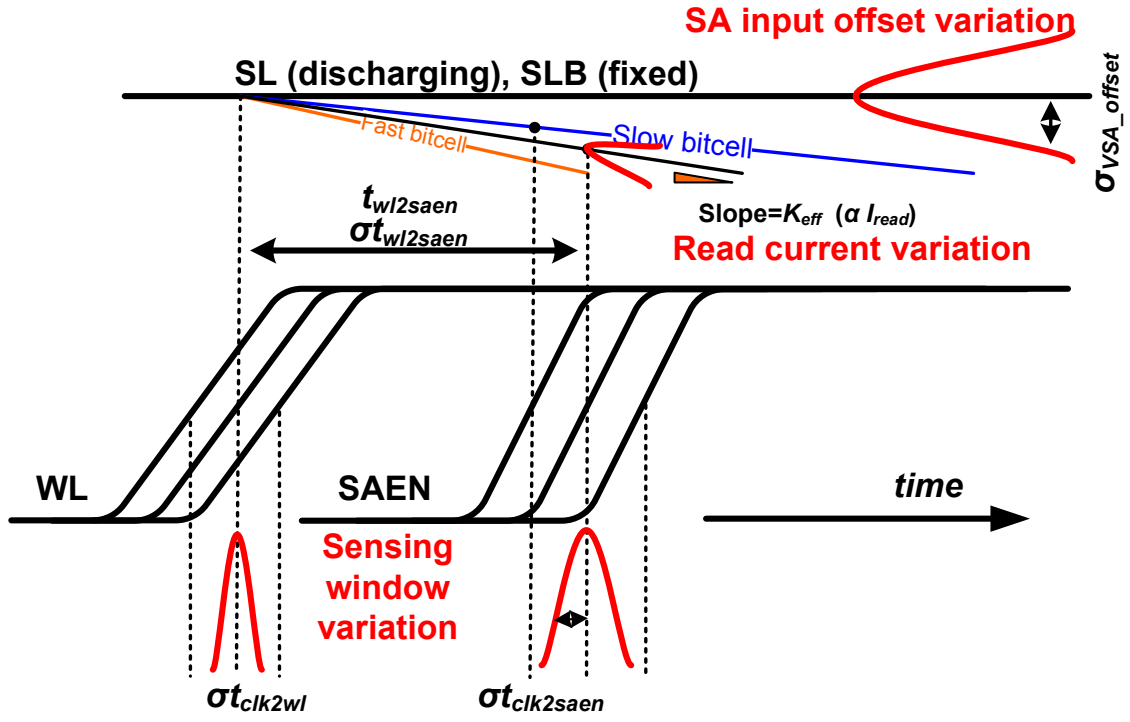


Figure 4.9: Main sources of variation affecting access failures.

The worst-case sensing occurs when all the unselected bitcells on the column store the opposite data polarity of the selected bitcell, as shown in Fig. 4.10. Pass-gate transistor leakage determines the upper limit of the number of bitcells per column. This effect is usually important in high performance memories due to the high leakage (low V_{th}) of the pass-gate device. The effective read current can be calculated as [13]:

$$\mathbf{I_{read,eff}} = \mathbf{I_{read}} - \sum_{i=1}^{N_c-1} \mathbf{I_{off,PG,i}} \tag{4.1}$$

where $\mathbf{I_{read,eff}}$ is the effective read current for a bitcell after accounting for the pass-gate leakage for adjacent bitcells on the same column, N_c is the number of bitcells per column, and $\mathbf{I_{off,PG,i}}$ is the pass-gate leakage for one bitcell[3].

Due to the exponential dependence of subthreshold leakage on V_{th} variations, it is important to statistically calculate the total leakage of all pass-gate devices. Assuming subthreshold leakage is the dominant and that there is large number of bitcells on the same column, it can be shown that [103, 104]:

$$\begin{aligned} \sum_{i=1}^{N_c-1} \mathbf{I_{off,PG,i}} &= (N_c - 1)\, \mu_{I_{off,PG}} \\ &\approx (N_c - 1)\, I_{off,PG} \left(1 + \frac{\ln^2(10)}{2} (\frac{\sigma_{V_{th}}}{S})^2\right) \end{aligned} \tag{4.2}$$

where $I_{off,PG}$ is the nominal PG leakage (assuming there is no V_{th} variation), $\sigma_{V_{th}}$ is the variation in V_{th} for the pass-gate device due to random WID variations, S is the subthreshold slope. From Eq. (4.2), it is clear that larger $\sigma_{V_{th}}$ increases the total PG leakage, which reduces the effective read current as shown in Eq. (4.1).

4.4 Proposed Yield Estimation Flow

Fig. 4.9 shows a simplified timing diagram for an accessed bitcell including critical signals such as WL, SAEN and sense lines (SL). Also shown in the figure are the statistical variations on different components that affect the probability of access failure, which were described in the previous section (Section 4.3). When the WL is enabled, SL begins discharging, and the slope of SL discharge varies statistically depending on I_{read} variations as well as leakage from other bitcells (Section 4.3.4).

[3] A **bold** symbol is used to indicate a random variable.

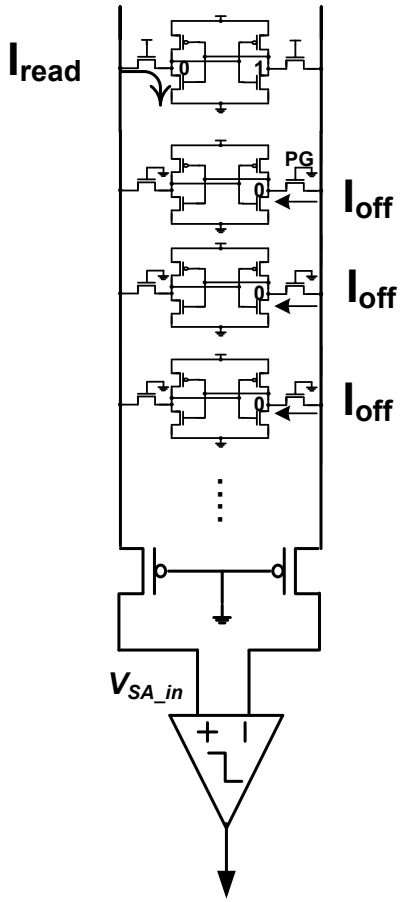


Figure 4.10: Impact of pass-gate leakage on the SA's input differential voltage (V_{SAin}). Pass-gate leakage from adjacent bitcells on the same column reduces the effective read current for the selected bitcell.

For the SA, the offset voltage distribution is superimposed on the V_{DD} (precharge level). As explained in Section 4.3.2, SA offset distribution is centered around zero (typically small asymmetry), as shown in Fig. 4.6, which means that some SAs will show positive or negative offsets. Note that positive offset will increase the failure probability of reading a 0 and reduce the failure probability of reading a 1, and vice versa. Therefore, in order to account for SA offset statistically in yield estimation, both read 0 and read 1 cases need to be addressed.

In addition to I_{read} and SA offset variations, $t_{wl2saen}$ variation can affect access failure probability, as described in Section 4.3.3. As shown in Fig. 4.9, if $t_{wl2saen}$ decreases due to statistical variations, V_{SAin} decreases, and hence the probability of access failure increases.

In order to estimate SRAM yield, it is important to statistically account for all the above mentioned sources in the same flow. Therefore, we define access failure for a certain bitcell as follows: For read 0 case, the probability of access failure is the probability of having SA input voltage V_{SAin} less than SA input offset $V_{SAoffset}$ **of that particular SA**. Note that in this case we are not assuming a fixed value of SA offset as in [12, 13]. Instead, the SA offset follows the normal distribution that can be determined from Monte Carlo simulations. Moreover, V_{SAin} needs to be computed statistically since it is a function of the statistical distribution of bitcell I_{read} and $t_{wl2saen}$ distributions.

Therefore, the probability of access failure for bitcell $P_{AF,cell}$ in case of reading a 0 can be expressed mathematically as follows:

$$\begin{aligned} P_{AF,cell,read0} &= P(\mathbf{V_{SAin}} - \mathbf{V_{SAoffset}} < 0) \\ &= P(\mathbf{K_{eff0}}\mathbf{t_{wl2saen}} - \mathbf{V_{SAoffset}} < 0) \end{aligned} \tag{4.3}$$

where $\mathbf{K_{eff0}}$, $\mathbf{t_{wl2saen}}$ and $\mathbf{V_{SAoffset}}$ are all random variables following a normal distribution, as explained in Section 4.3. Similar expression can be derived for the read 1 access failure. To account for pass-gate leakage, K_{eff} distribution can be calculated as:

$$\begin{aligned} \mathbf{K_{eff0}} \text{ follows } \mathcal{N} &\sim (\mu_{K_{eff}}, {\sigma^2}_{K_{eff}}) \\ \mu_{K_{eff}} &= \left|\frac{\Delta V_{SL}}{\Delta t}\right| (1 - \frac{(N_c - 1)\mu_{I_{off,PG}}}{\mu_{I_{read}}}) \\ \sigma_{K_{eff}} &= \left|\frac{\Delta V_{SL}}{\Delta t}\right| \frac{\sigma_{I_{read}}}{\mu_{I_{read}}} \end{aligned} \tag{4.4}$$

Table 4.1: Read failure model inputs for the proposed statistical yield estimation methodology.

	Parameter	Estimation Technique
I_{read} variation	$\mu_{I_{read}}$, $\sigma_{I_{read}}$	DC MC* simulation for the bitcell.
Sensing slope	$\lvert\frac{\Delta V_{SL}}{\Delta t}\rvert$	Transient simulation at nominal conditions (no variations) for bitlines discharge.
SA input offset	$\mu_{V_{SAoffset}}$, $\sigma_{V_{SAoffset}}$	Transient MC simulation for SA.
Sensing window variation	$\mu_{t_{wl2saen}}$, $\sigma_{t_{wl2saen}}$	Transient MC simulation for WL and SAEN paths.
PG leakage	$\mu_{I_{off,PG}}$	DC MC simulation for PG transistors.

* MC: Monte Carlo.

Table 4.1 summarizes the inputs for the read failure model. For typical memory architecture shown in Fig. 4.11, and using the proposed access failure definition in Eq. (4.4), the flow for read access yield computation is implemented as follows (as shown in the flowchart in Fig. 4.12):

1. From the memory architecture (density, word length, number of columns, muxing), find the number of banks (N_{banks}), SAs per bank ($N_{SA-bank}$) and number of bitcells accessed by one SA ($N_{bits-SA}$);
2. Initialize the chip counter (number of Monte Carlo runs);
3. Generate one sample of $t_{wl2saen}$: $\mathcal{N} \backsim (\mu_{t_{wl2saen}}, {\sigma^2}_{t_{wl2saen}})$;[4]
4. Generate one sample SA input offset distribution : $\mathcal{N} \backsim (\mu_{V_{SAoffset}}, {\sigma^2}_{V_{SAoffset}})$;
5. Generate $2N_{bit-bank}$ samples of K_{eff} normal distribution using Eq. (4.4) to represent the read 0 and read 1 sensing slope distributions ($\mathbf{K_{eff0}}$, $\mathbf{K_{eff1}}$);
6. Failure calculation step: loop on all the bitcells accessed using this particular SA. Check the following fail conditions for each bitcell;

[4]Here we assume that a bank contains one control block which generates WL and SAEN signals as shown in Fig 4.11. Nevertheless, different types of banking styles can be easily included in the flow.

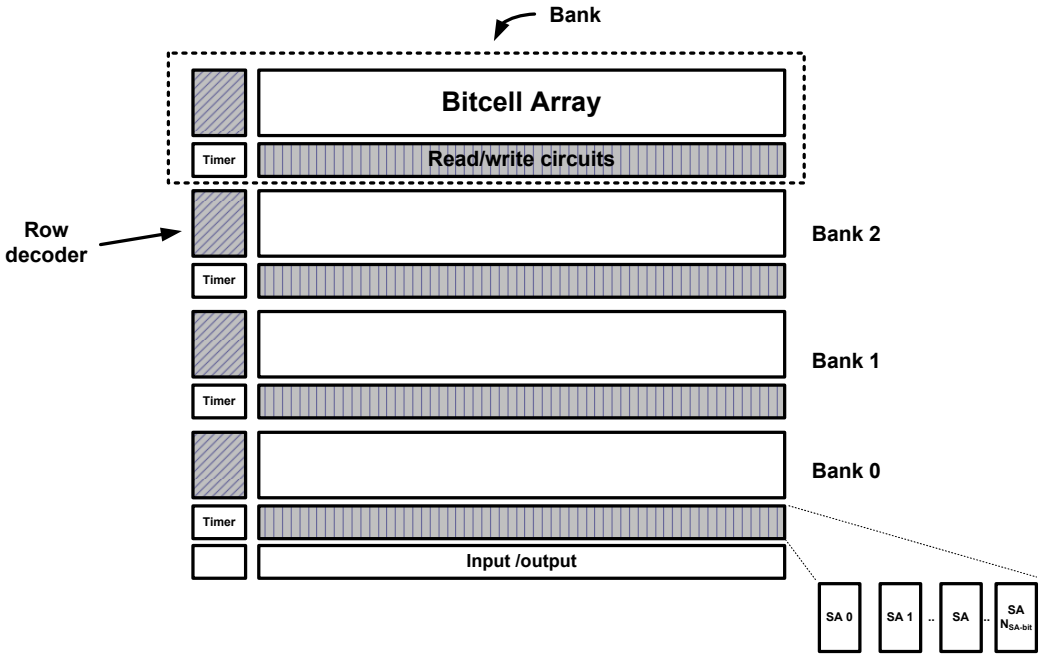


Figure 4.11: Typical SRAM architecture used in the proposed statistical yield estimation flow.

- Read 0 fail: $\mathbf{K_{eff0}\, t_{wl2saen} - V_{SA\,offset}} < 0$
- Read 1 fail: $\mathbf{K_{eff1}\, t_{wl2saen} - V_{SA\,offset}} > 0$
- Count the number of read failures.

7. Repeat all the above steps for all SA per bank ($N_{SA-bank}$);
8. Repeat all the above steps for different banks (N_{banks});
9. Repeat all the above steps for large number of chips (Monte Carlo runs), count the number of failing chips;
10. Calculate the yield based on the number of chips that can correctly be accessed for read 0 and 1 cases, Yield=(number of passing chips)/(total number of chips).

While the above steps focused on WID variation, the proposed methodology can be easily extended to account for die-to-die (D2D) variations. This can be done by including the statistical distributions of D2D variations and including different D2D samples for each run at the chip level (*i.e.*, in step 9 shown above). However, this will also require pre-characterization at all the D2D sample points.

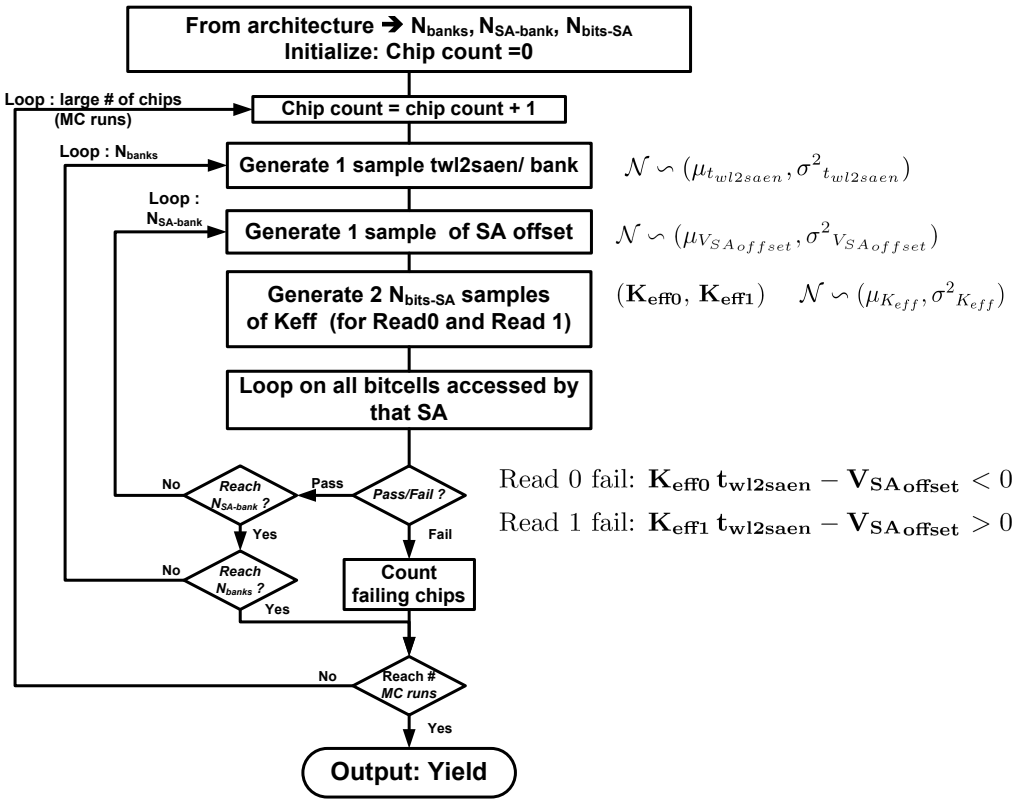


Figure 4.12: Flowchart of the proposed statistical yield estimation flow.

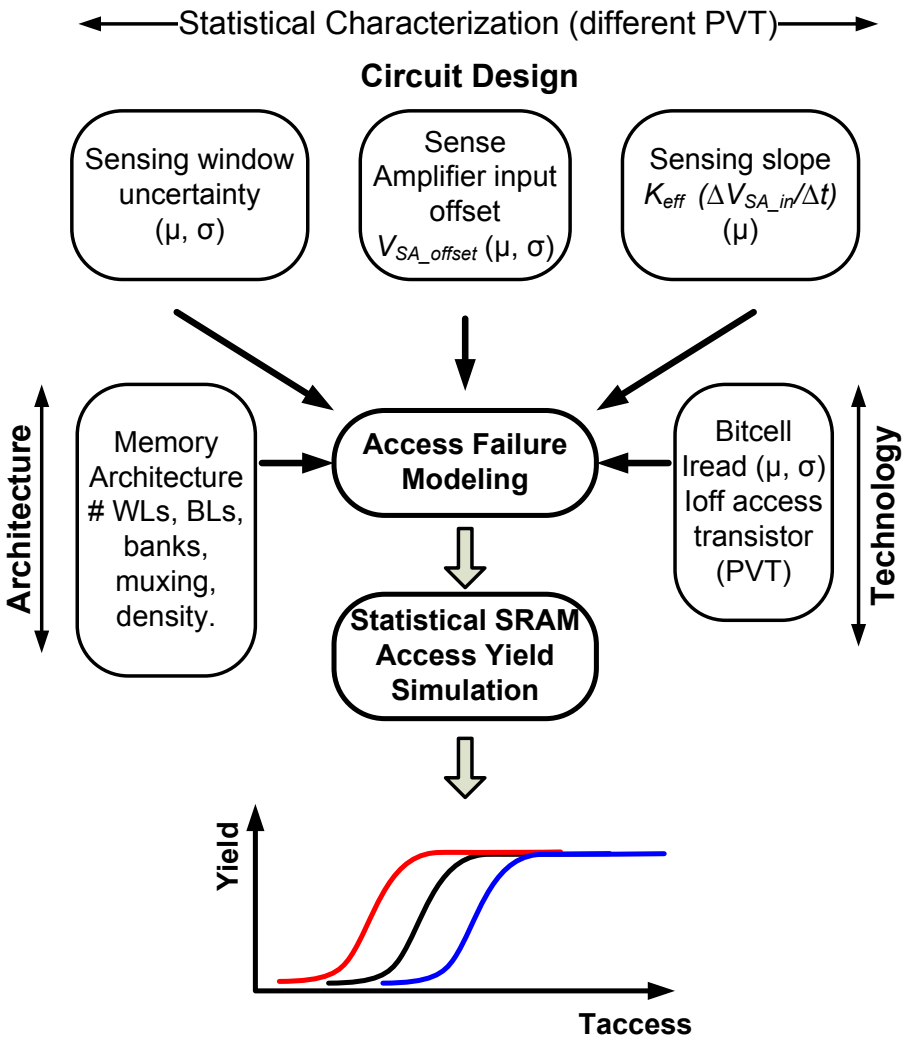


Figure 4.13: Yield estimation flow.

4.5 Experimental Results

The proposed yield estimation methodology was verified using a 1Mb SRAM design fabricated in an industrial 45nm CMOS technology. Prior to running the proposed yield estimation flow, a characterization step is required to compute the inputs for the proposed flow (Table 4.1). However, this characterization step is not computationally expensive due to the reduced number of circuit elements for these simulation setups. In addition, these inputs simulations are an integral part of any SRAM design and should be readily available even when using worst-case analysis.

Characterization for the different components of yield failures was performed as shown in Fig. 4.13 for different conditions. I_{read} was characterized using DC Monte Carlo SPICE simulations to estimate the mean and standard deviation. Fig. 4.14

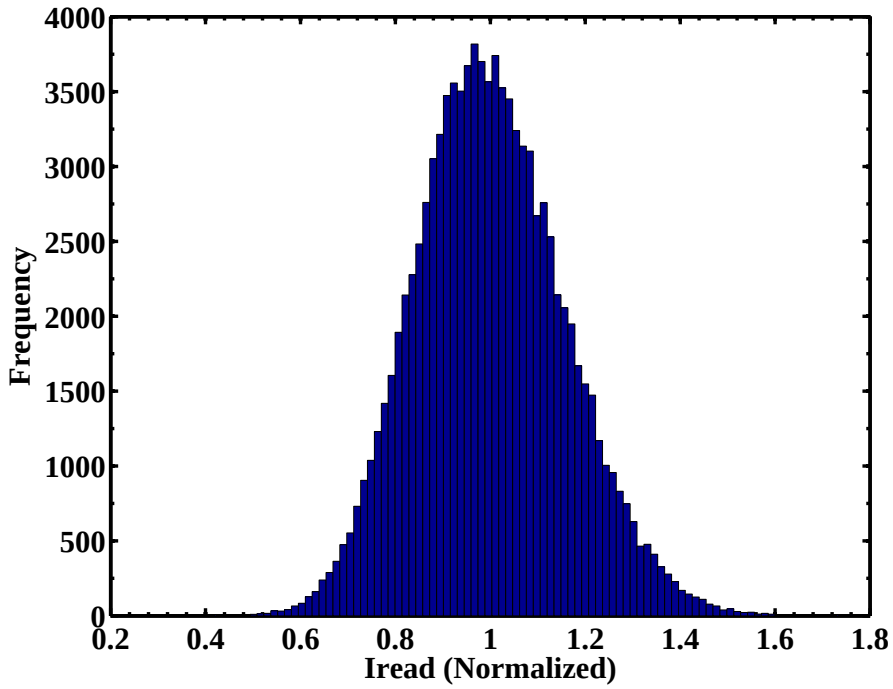


Figure 4.14: I_{read} histogram from Monte Carlo simulation (100k runs).

shows the I_{read} histogram for 100k MC runs. As expected, I_{read} follows a normal distribution. From I_{read} distribution, $\mu_{I_{read}}$ and $\sigma_{I_{read}}$ are computed. Fig. 4.15 shows how $\sigma/\mu|_{I_{read}}$ changes for different voltage and temperature conditions. This is an important part of the characterization as I_{read} mean and variances changes significantly with temperature and voltage. Note that $\sigma/\mu|_{I_{read}}$ reaches 20% at low V_{DD} and low temperature. This shows the strong impact of process variations on bitcell I_{read}.

The sensing slope ($|\frac{\Delta V_{SL}}{\Delta t}|$) was extracted by running transient SPICE simulation for bitline discharge rate. Note that Monte Carlo simulation is not required in this case since we use I_{read} variations calculated from the first step to estimate K_{eff} variations as shown in Eq. (4.4). SA offset distribution was simulated using Monte Carlo transient simulation as shown in Fig. 4.16, which shows the simulated/modeled cumulative distribution functions (CDF) for the SA input offset (normal distribution). Also shown in Fig. 4.17 is the impact of V_{DD} and temperature of SA input offset where $\sigma_{V_{SAoffset}}$ shows strong sensitivity to both. However, interestingly, $\sigma_{V_{SAoffset}}$ shows opposite trend to I_{read} variation, which means that SA characteristics can slightly compensate the significant increase in I_{read} variation as V_{DD} is reduced.

Sensing window variation $t_{wl2saen}$ distribution was estimated using Monte Carlo

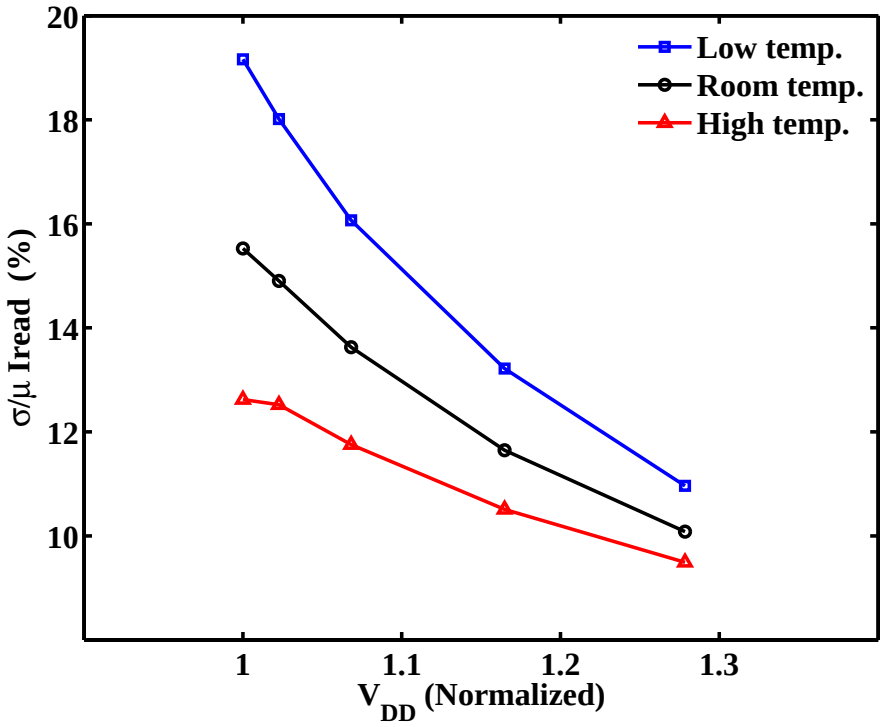


Figure 4.15: Characterization results for bitcell I_{read} variation using DC Monte Carlo simulation.

transient simulation on the WL and SAEN paths of the memory. Fig. 4.18 shows the impact of V_{DD} and temperature on $t_{wl2saen}$ variation. Similar to I_{read} variation, $t_{wl2saen}$ variation increases as V_{DD} or temperature is reduced.

For pass-gate leakage, DC Monte Carlo simulation was used to estimate $\mu_{I_{off,PG}}$. Fig. 4.19 shows the pass-gate leakage histogram. Notice how the distribution follows a lognormal shape. From simulation results, it was found that $\mu_{I_{off,PG}}$ is 1.3X larger than nominal pass-gate leakage. Fig. 4.20 shows how $\mu_{I_{off,PG}}$ varies at different temperatures and V_{DD} conditions. Since pass-gate leakage is dominated by subthreshold leakage, it is clear that the impact of temperature is the dominant.

After generating the characterization data, and inputting the memory architecture information, the statistical yield simulation described in Section 4.4 was executed using Matlab. Fig. 4.21 shows the measured yield from the 1Mb memory compared to the simulation for different supply voltage conditions. Good agreement between silicon and simulation results validate the accuracy of the proposed methodology. For these simulation results, 1000 chips of the 1Mb memory were simulated using the proposed flow. All bitcells were tested for read 0 and read 1 fails. Yield estimation for the proposed methodology takes less than 30 min-

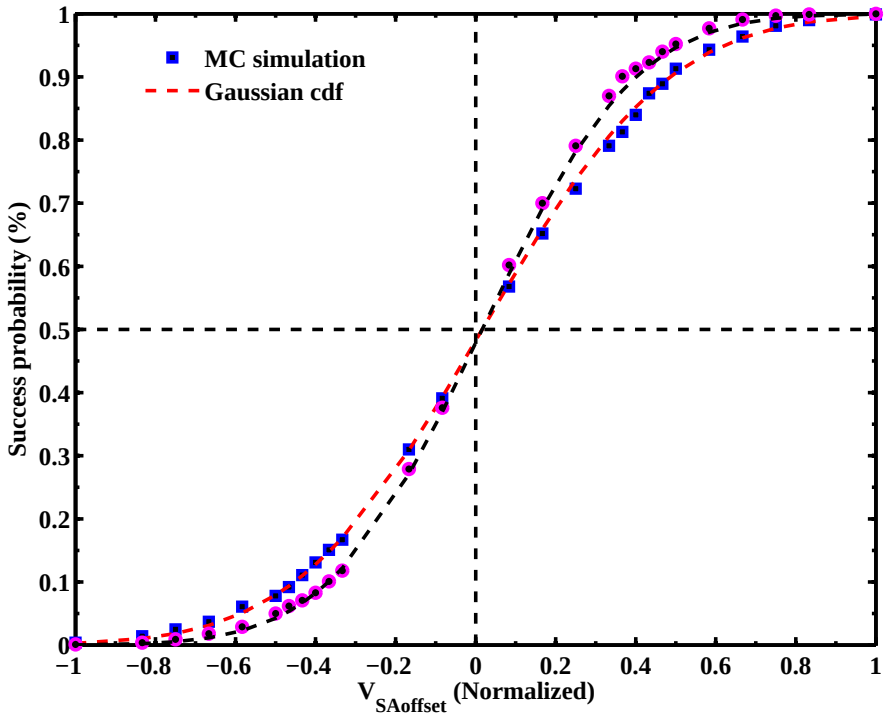


Figure 4.16: Characterization results for SA offset using transient Monte Carlo Analysis for different conditions. SA input offset follows a normal distribution as shown by the cdf (cumulative distribution function) of simulation and model.

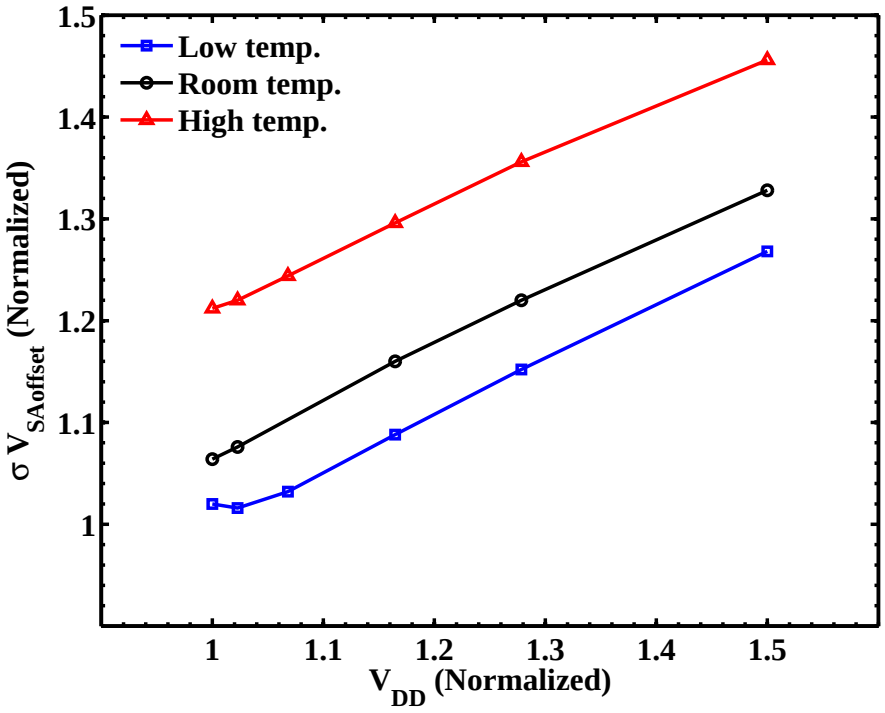


Figure 4.17: Characterization results for $\sigma_{V_{SA_{offset}}}$ versus V_{DD} at different temperatures using Monte Carlo simulation.

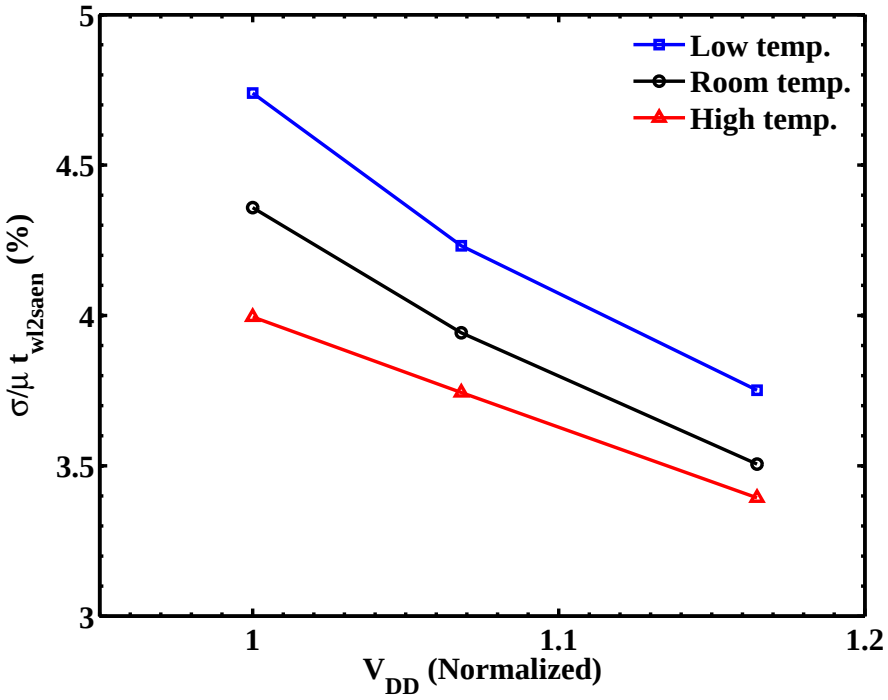


Figure 4.18: Characterization results sensing window variation versus V_{DD} at different temperatures using Monte Carlo simulation.

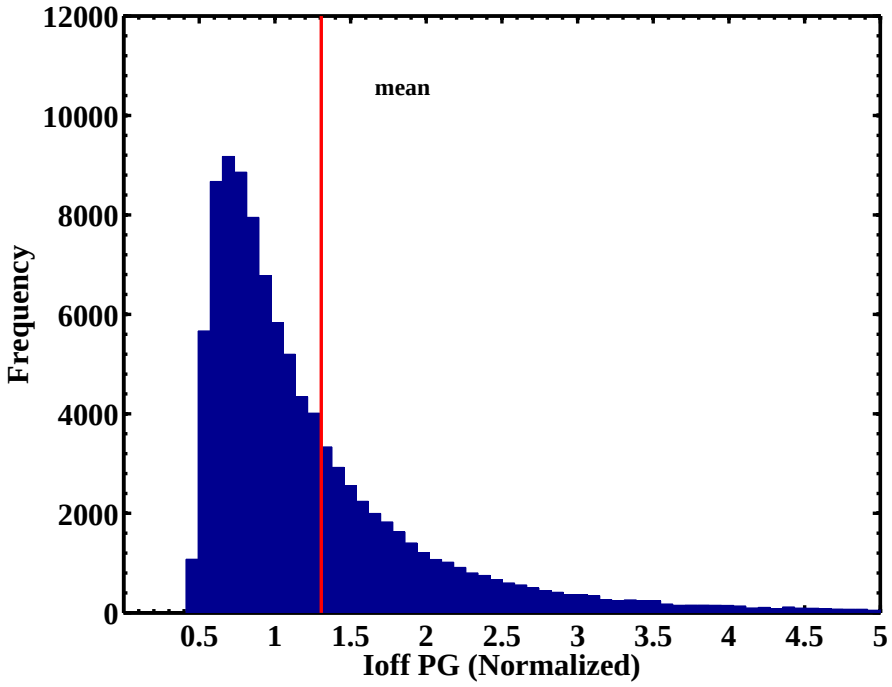


Figure 4.19: Pass-gate leakage distribution.

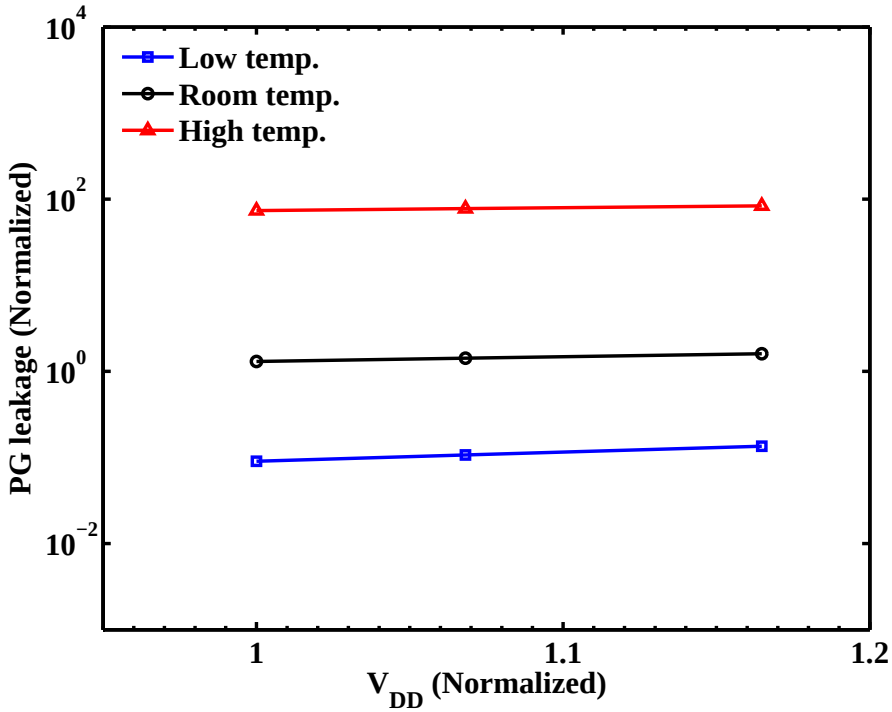


Figure 4.20: Characterization results for $\mu_{I_{off,PG}}$ versus V_{DD} at different temperatures using Monte Carlo simulation.

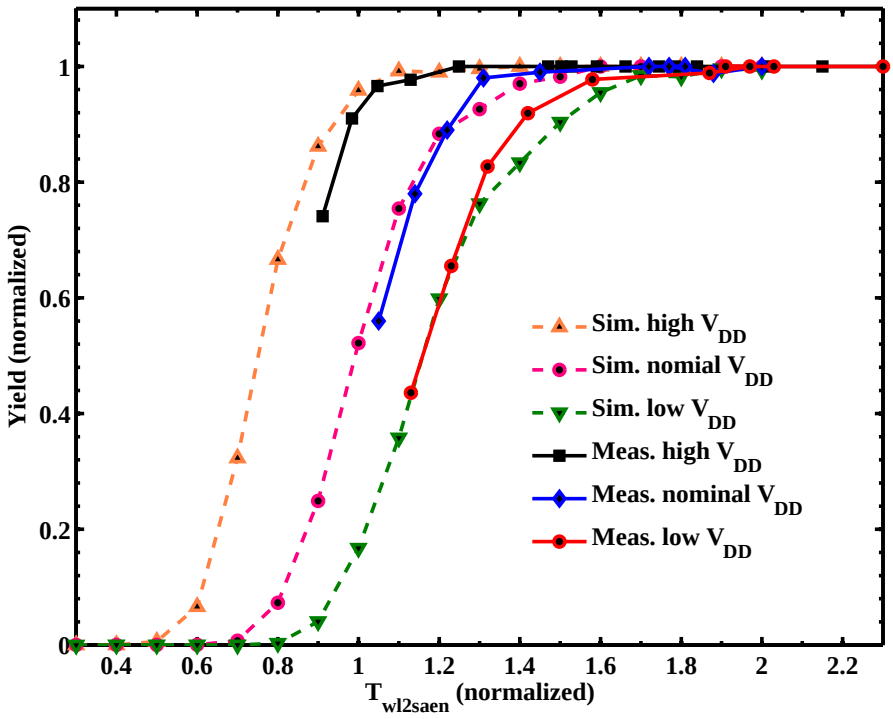


Figure 4.21: Comparison between simulation results using the proposed yield estimation methodology and the measured access yield for a 1Mb memory in 45nm technology.

utes to generate the results shown in Fig. 4.21 using a 3GHz PC with 1.5GB of memory which shows the efficiency of the proposed flow. The simulation results in Fig. 4.21 can be used explore the critical tradeoff between performance and yield requirement.

It is useful to compare the difference between using the proposed statistical yield estimation flow versus the worst-case analysis. This is shown in Fig. 4.22, where in the worst-case approach, the worst bitcell is assumed to occur with the SA having the largest offset and the smallest sensing window. Also shown is the statistical design approach to meet a yield of 99.7%. The statistical design enables reducing $t_{wls2saen}$ by 26%, which translates to higher memory performance. This translates to 18% faster access time assuming $t_{wls2saen}$ is 30% of access time [73]. In the meantime, the memory read power consumption also reduces because of reduced differential voltage on the bitlines.

It is important to note that the performance benefit of using statistical methodology versus worst-case approaches is a function of the memory density. It is expected that the difference between the two approaches increases with scaling due

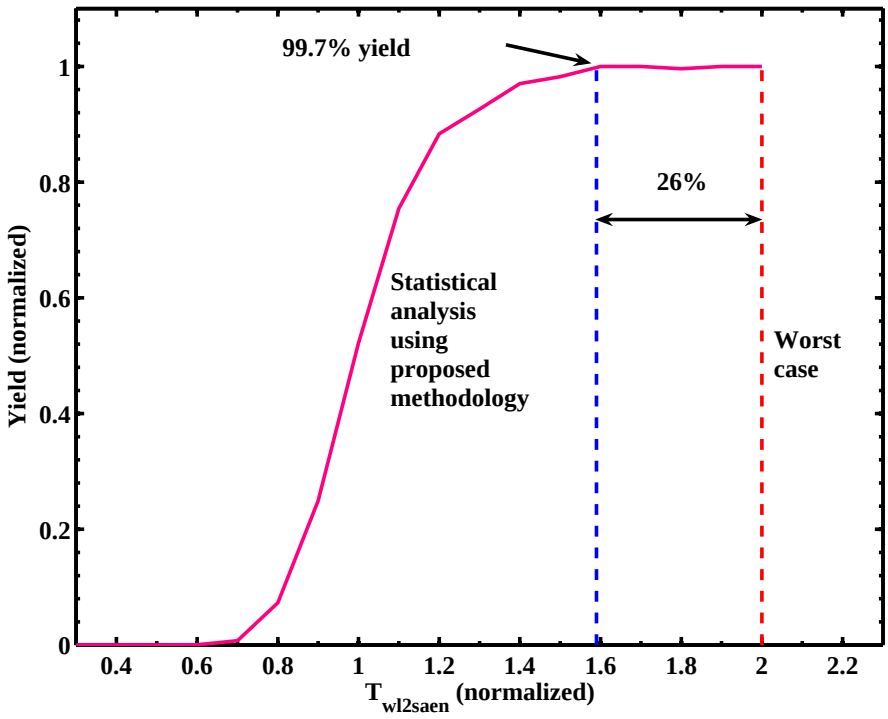


Figure 4.22: Comparison between the proposed statistical yield estimation methodology and worst-case analysis.

to the continuous increase in process variations as well as the higher SRAM density requirements. This shows the importance of statistically accounting for different components of read failures as proposed in this work, so that pessimism in worst-case approaches can be recovered in the design time.

4.6 Summary

The large increase in statistical variations in nanometer technologies is presenting huge challenges for SRAM designers. In this chapter, a methodology for statistical estimation of access yield is proposed. The proposed flow accounts for the impact of bitcell read current variation, sense amplifier offset distribution, timing window variation and leakage variation on read failure. The methodology overcomes the pessimism in worst-case design techniques that are usually used in SRAM design. The methodology is verified using measured yield data from a 1Mb memory in an industrial 45nm technology. The proposed statistical SRAM yield estimation methodology allows early yield prediction in the design cycle, which can be used to trade off yield, performance and power requirements for SRAM.

Chapter 5

Reducing SRAM Power using Fine-Grained Wordline Pulse Width Control

In the previous chapters, we showed how process variations affect SRAM operation. A bi-product of variation-tolerant techniques is the increase of power consumption. This chapter introduces Fine-Grained Wordline Pulse Width Control which is a new variation-tolerant architecture to reduce SRAM switching power consumption. The proposed solution combines memory built-in self test (BIST) with programmable delay elements in a closed-loop to reduce the switching power consumption for the memory. Section 5.1 introduces the problem of SRAM power consumption. In Section 5.2, we derive statistical models for memory read access yield and array read power consumption, which show the tradeoff between yield and power metrics. In Section 5.3, we describe the proposed system and its operation. In Section 5.4, we present the statistical simulation flow used to estimate power savings using the proposed system (applied for memories in an industrial 45nm technology). In addition, we discuss some design considerations related to the proposed system. In Section 5.5, we summarize our findings.

5.1 Introduction

With technology scaling, the requirements of higher density and lower power embedded SRAM are increasing exponentially. It is expected that more than 90% of die area in future System-on-Chip (SoC) will be occupied by SRAM [105]. This is

driven by the high demand for low-power mobile systems, which integrate a wide range of functionality such as digital cameras, 3D graphics, MP3 players and other applications. In the meantime, random variations are increasing significantly with technology scaling. Random dopant fluctuation (RDF) is the dominant source of random variation in the bitcell's transistors. The variations in V_{th} due to RDF are inversely proportional to the square root of device area [93]. Therefore, SRAM bitcells experience the largest random variations on a chip, as bitcell transistors are typically the smallest devices for given design rules [1, 12, 14, 106].

Embedded SRAMs usually dominate the SoC silicon area and their power consumption (both dynamic and static) is a considerable portion of the total power consumption of an SoC. It has been shown that SRAM caches alone can consume up to 40% of total chip power [107]. SRAM array switching power consumption is considered one of the largest components of power in high density memories [107–109]. In [108], it was shown that array switching power can reach $\sim$ 90% of memory switching power (depending on memory architecture). This is mainly because of the large memory arrays and the requirements for high area efficiency which forces SRAM designers to use the maximum numbers of rows and columns enabled by the technology [1]. Fig. 5.1 shows dynamic power consumption for read operation versus wordline pulse width for a 512kb memory macro designed in an industrial 65nm technology. Power consumption results are extrapolated to $T_{wl} = 0$ to estimate the component of switching power due to the peripheral circuits. For normal operating conditions, array power consumption is more than 60% of read power. Therefore, it is important to reduce the array switching due to its strong impact on the memory's total power as well as the SoC's power.

Several circuit techniques have been proposed to reduce SRAM array switching power consumption by reducing wordline pulse width (T_{wl}). One of the most common techniques to control T_{wl} is using a bitcell replica path, which reduces the bitline differential, hence, lowering power consumption [14,68,102,106,110]. Replica path (*e.g.*, self-timed) techniques provide a simple approach of process tracking for *global variations* (interdie or systematic within-die) as well as environmental variations (voltage and temperature). However, these circuit techniques are not efficient when memory bitcells experience large random variation, since circuit techniques cannot adapt to *random variations*. Therefore, their effectiveness decreases with process scaling, and larger design margins are used which increases power consumption due to larger T_{wl}. To reduce the loss due to excessive margining, circuits and

[1] Assuming speed constraints are met.

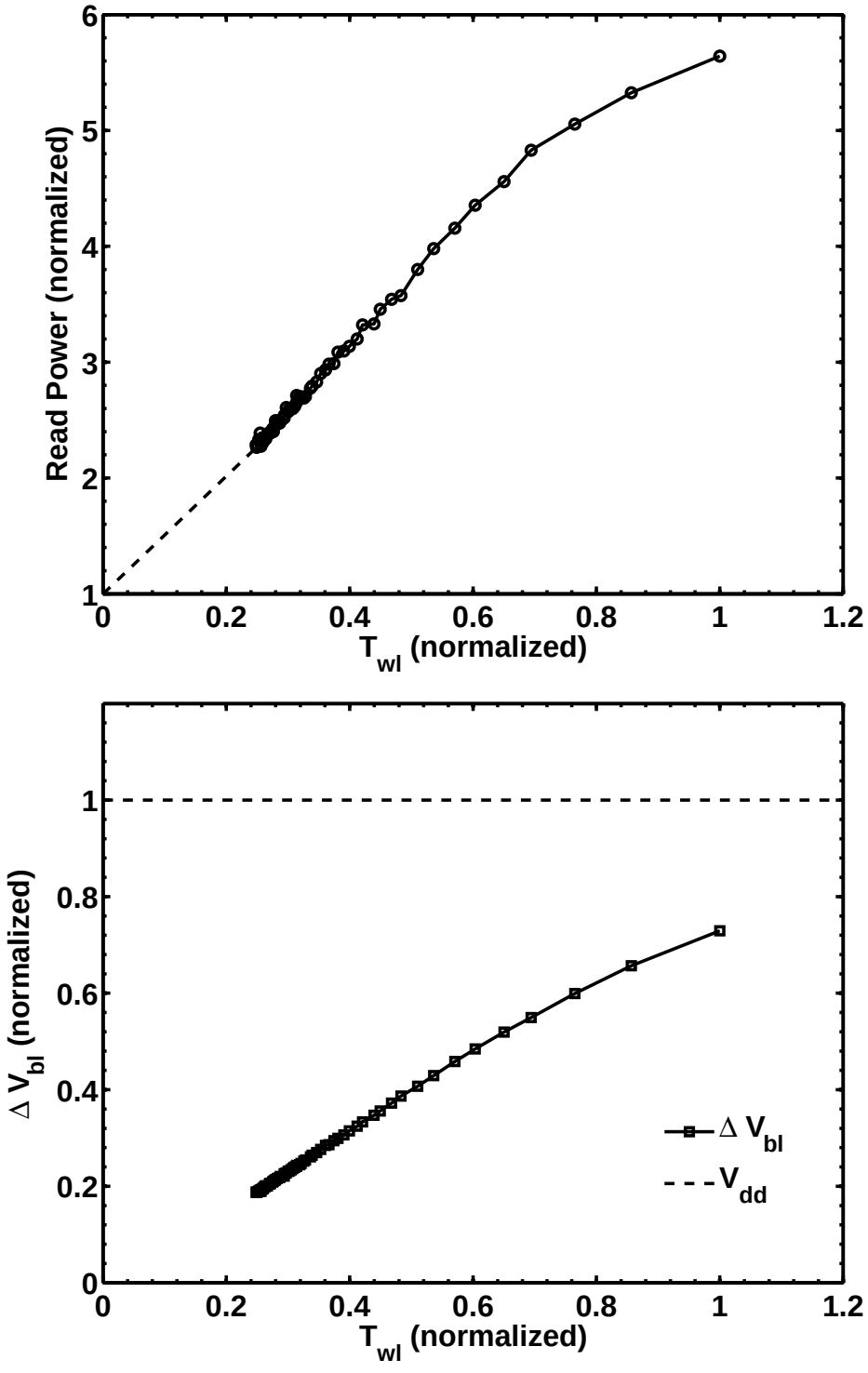


Figure 5.1: Memory read power and bitline differential versus T_{wl} for a 512kb memory in 65nm technology.

architectures must be designed together to reduce power and manage variability. Higher levels of design abstraction can have better variation-tolerance capabilities because the impact of random variation can be measured at these levels [1, 19, 55]. Therefore, combining architecture techniques with circuit level designs can reduce the pessimism in using worst-case approaches, and can help adapt the circuit to random variation, which can reduce power consumption [1, 19, 55].

5.2 SRAM Yield and Power Tradeoff

Due to large area of SRAM in modern SoC, SRAM yield can dominate the overall chip yield. Hence, statistical design margining techniques are used to guarantee high memory yield. However, to achieve high yield, memory power consumption (and speed) are negatively impacted. The stringent requirements of high yield and low power consumption requires combining circuit and architectural techniques to reduce SRAMs power consumption. In this section, we derive simple models for SRAM yield and array power consumption.

Due to random variation in SRAM bitcell, there is a tight coupling between memory yield and power consumption. To achieve high yield, read access failures should be minimized. Read access yield is defined as the probability of correct read operation. In read operation, the selected wordline is activated for a period of time to allow the bitlines to discharge. The wordline activation time, T_{wl}, is a critical parameter for memory design since it affects the memory speed (access time) as well as memory power. To reduce read access failures, the wordline pulse T_{wl} should be large enough to guarantee adequate bitline differential, which can be sensed correctly using the sense amplifier.

The total power consumption for a memory in a read or write cycle can be expressed as:

$$P_{mem} = P_{leak} + P_{s_{array}} + P_{s_{peri}} \tag{5.1}$$

where P_{leak} is the total leakage power from the array and the peripheral circuitry, $P_{s_{array}}$ and $P_{s_{peri}}$ are the switching power from the array and the peripheral circuitry, respectively (as shown in Fig. 5.2).

In a read access, the array switching power can be calculated as:

$$P_{s_{array}} = N_{bl}\, N_{wl}\, C_{bit}\, \Delta V_{bl}\, V_{dd}\, f \tag{5.2}$$

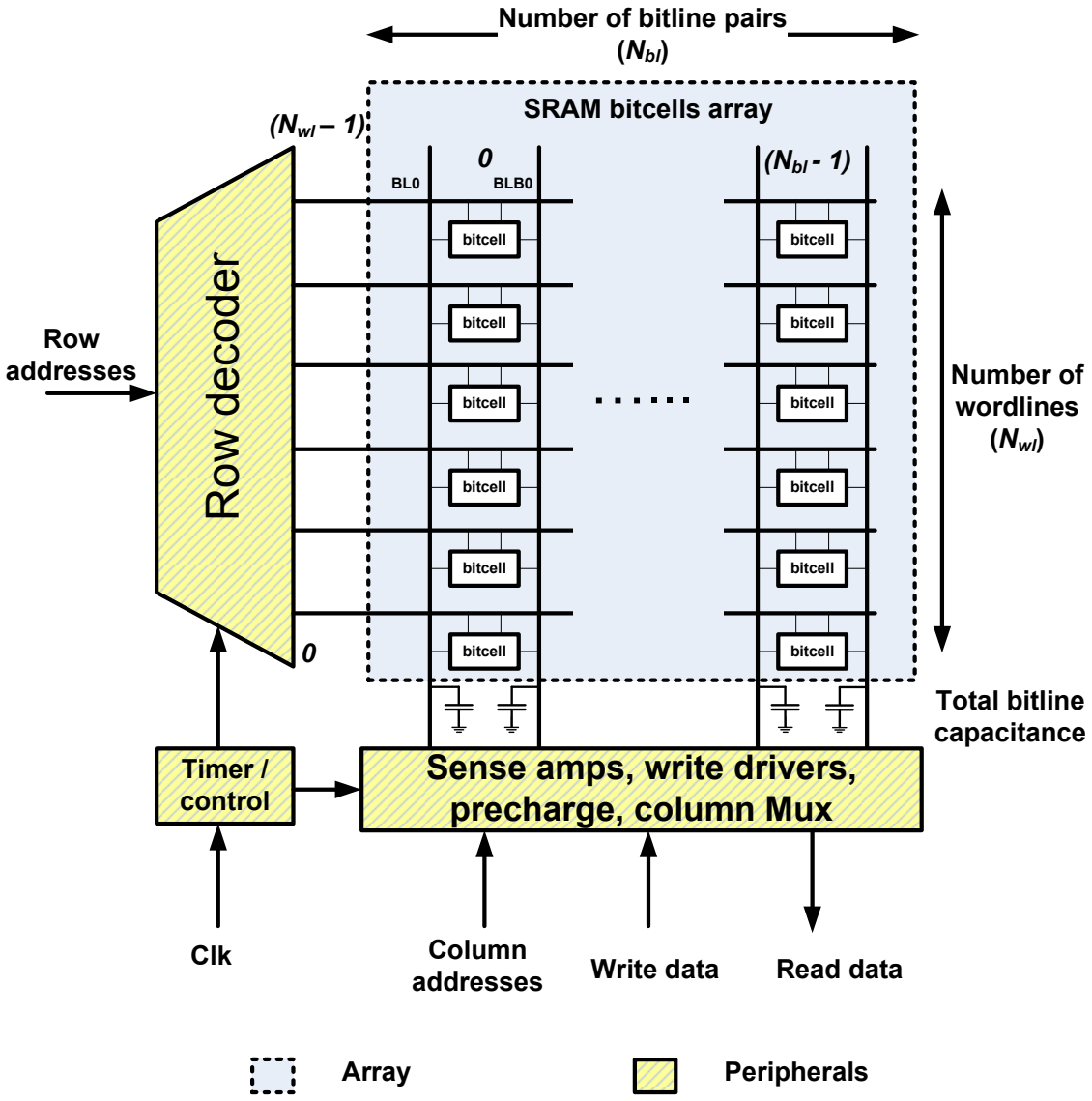


Figure 5.2: Typical SRAM architecture.

where N_{bl} are N_{wl} are the number bitline pairs and wordlines in a memory bank, respectively. C_{bit} is the bitline capacitance per bitcell, ΔV_{bl} is the bitline differential in read access (used to sense the bitcell's stored value), V_{dd} is the supply voltage, and f is the operating frequency.

ΔV_{bl} can be calculated as:

$$\Delta V_{bl} \simeq \begin{cases} \frac{I_c\, T_{wl}}{N_{wl}\, C_{bit}} & \text{for} \quad T_{wl} \leq \frac{V_{dd}\, N_{wl}\, C_{bit}}{I_c} \\ V_{dd} & \text{for} \quad T_{wl} > \frac{V_{dd}\, N_{wl}\, C_{bit}}{I_c} \end{cases} \tag{5.3}$$

where I_c is the bitcell read current. To a first order, ΔV_{bl} can be approximated by assuming linear dependence on T_{wl}, for the range of T_{wl} where $\Delta V_{bl} < V_{dd}$, as

shown in Fig. 5.1.

Therefore, from Eq. (5.2) and Eq. (5.3), the array switching power can be computed as:

$$P_{s_{array}} \simeq \begin{cases} N_{bl}\, I_c\, T_{wl} V_{dd}\, f & \text{for} \quad T_{wl} \leq \frac{V_{dd}\, N_{wl}\, C_{bit}}{I_c} \\ N_{bl}\, N_{wl}\, C_{bit}\, {V_{dd}}^2\, f & \text{for} \quad T_{wl} > \frac{V_{dd}\, N_{wl}\, C_{bit}}{I_c} \end{cases} \tag{5.4}$$

From Eq. (5.4), it is clear that $P_{s_{array}}$ is directly proportional to T_{wl} (when $T_{wl} \leq \frac{V_{dd}\, N_{wl}\, C_{bit}}{I_c}$), which is confirmed by the read power results shown in Fig. 5.1.

A correct read operation requires ΔV_{bl} to be large enough to guarantee correct sensing using the sense amplifier. Hence, large ΔV_{bl} implies having sufficiently large T_{wl} that enable weak bitcells (with low I_c) to be correctly sensed, for a given yield requirement. Increasing T_{wl} increases read access yield, however, in the same time increasing T_{wl} increases power consumption, as shown in Eq. (5.4). Moreover, T_{wl} has a direct impact on a memory's access time [14]. Therefore, T_{wl} is usually set to ensure correct read operation for a given read access yield requirement and memory density (as discussed in Chapter 4).

Bitcell read current is strongly affected by the random V_{th} variations in the bitcell access device (pass-gate) as well as the pull-down device. Due to these variations, I_c has been shown to follow a normal distribution $\mathcal{N} \sim (\mu_{I_c}, {\sigma^2}_{I_c})$ with a mean of μ_{I_c} and standard deviation of σ_{I_c} [12–14].

Therefore, to guarantee correct read operation, the following condition should be satisfied [12–14] [2]:

$$T_{wl,wc} = \frac{\Delta V_{min} N_{wl}\, C_{bit}}{\mu_{I_c}(1 - \frac{\sigma}{\mu}|_{I_c} N_\sigma)} \tag{5.5}$$

where ΔV_{min} is the minimum required bitlines differential voltage, which is a function of the sense amplifier input offset (typically $\Delta V_{min} = 0.1 V_{dd}$). μ_{I_c} is the mean bitcell read current, and $\frac{\sigma}{\mu}|_{I_c}$ is the relative variation in I_c. N_σ is the required design coverage, which is related to the target yield and the memory density [13, 14], and can be computed as:

$$N_\sigma = \Phi^{-1}({Y_{mem}}^{\frac{1}{N_{bits}}}) \tag{5.6}$$

[2] Here, we use the conventional worst-case read access yield condition to simplify the mathematical analysis. This model assumes that the failure is dominated by bitcell read current variation. For accurate yield calculations, the proposed statistical simulation flow described in Chapter 4 can be used.

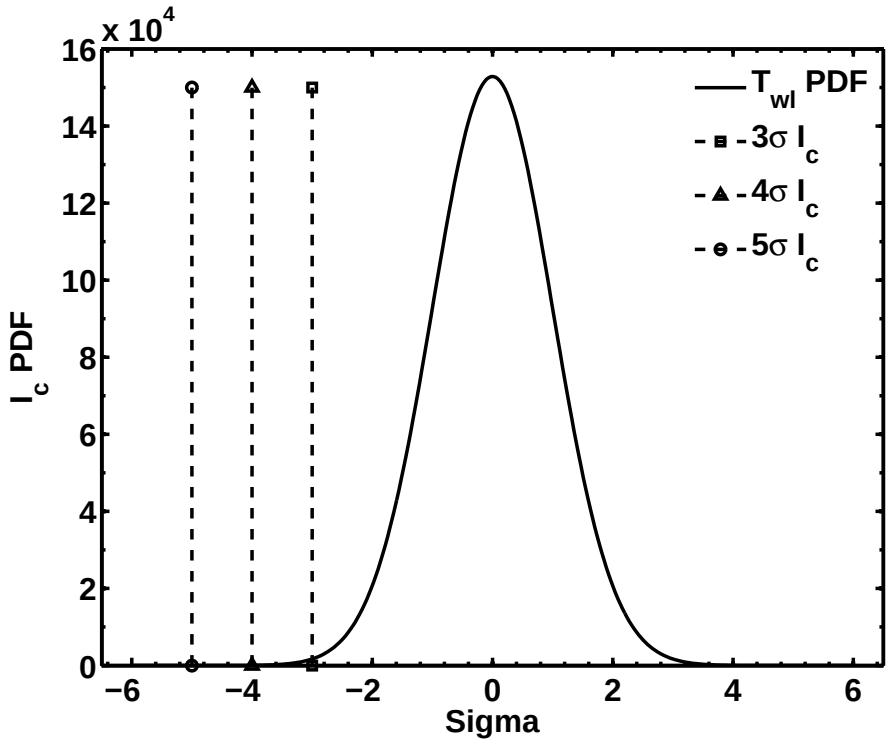


Figure 5.3: I_c probability density function (PDF) showing the points corresponding to 3,4 and $5\sigma_{I_c}$ (corresponding to different memory yield targets). $\frac{\sigma}{\mu}|_{I_c} = 15\%$ is assumed.

where Φ^{-1} is the inverse standard normal cumulative distribution function, Y_{mem} is the memory read access yield target, and N_{bits} are the total number of bitcells in the memory. For example, for a 1Mb memory, if the target read access yield is 95%, then the required design coverage is $N_\sigma = 5.33$. Therefore, to achieve the same yield for large memory density, N_σ should be increased. From Eq. 5.5, this means that larger $T_{wl,wc}$ is required.

It is important to note that the relation between $T_{wl,wc}$ and I_c is nonlinear as shown in Eq. 5.3 and Eq. 5.5. In fact, assuming I_c is a normal distribution, the probability density function (PDF) of $T_{wl,wc}$ can be calculated using one-to-one mapping from Eq. 5.5 as follows [111]:

$$P_{T_{wl}} = \frac{\Delta V_{min} N_{wl} C_{bit}}{{T_{wl}}^2} \varphi_{I_c}\left(\frac{\Delta V_{min} N_{wl} C_{bit}}{T_{wl}}\right) \tag{5.7}$$

where $P_{T_{wl}}$ is the PDF for T_{wl} and $\varphi_{I_c}()$ is the PDF for I_c, which is a normal distribution.

Fig. 5.3 and Fig. 5.4 show the distributions of bitcell I_c and T_{wl}, respectively. Note that T_{wl} PDF is not symmetric, but instead, it is skewed towards larger T_{wl}

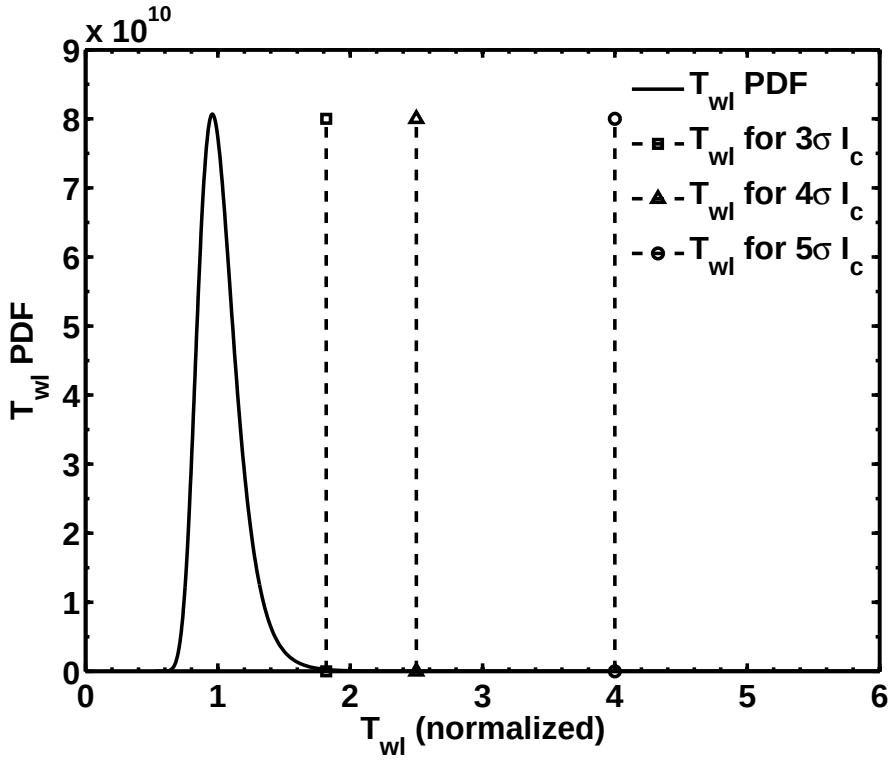


Figure 5.4: T_{wl} PDF. Also shown are the points corresponding to 3, 4 and $5\sigma_{I_c}$. Notice how the PDF is skewed towards higher T_{wl}. $\frac{\sigma}{\mu}|_{I_c} = 15\%$ is assumed.

values. Also, the 3,4 and $5\sigma_{I_c}$ values are shown for I_c, and the corresponding T_{wl} for these I_c values. It is clear that T_{wl} is very sensitive to I_c variations. For $5\sigma_{I_c}$, T_{wl} increases by 4X compared to its nominal value (calculated using μ_{I_c}).

Because of $P_{T_{wl}}$ skewed distribution, large values of T_{wl} are required to ensure an acceptable read access yield. Moreover, to achieve the same yield as the memory size increases, higher N_σ coverage is required as shown in Eq. 5.6, which significantly increases T_{wl} (due to the nonlinear relation between I_c and T_{wl}). Therefore, due to statistical variations in the bitcell, T_{wl} should be pessimistically large to achieve the required yield target. This will negatively impact the dynamic power for memories, which do not have weak bitcells (or have small spread of I_c around its mean value). Therefore, it is desirable to have a post-silicon approach that can recover the pessimism in determining T_{wl}.

5.3 Fine-Grained Wordline Pulse Width Control

As discussed in the previous section, read switching power increases significantly due to process variations, because at the design time we do not have information about which memories will have weak bitcells with low I_c. Therefore, a worst-case approach is applied to determine $T_{wl,wc}$ for a given memory density and yield requirement, as shown in Eq. 5.5.

A post-silicon approach is required to recover the increase in switching power due to the worst-case design practices used in SRAM design. Fig. 5.5 shows a conceptual view of the proposed architecture. The memory BIST is used to test the memory functionality using different testing patterns. Each memory instance contains a "WL delay" block, which is a digitally programmable delay element that controls the wordline pulse width T_{wl}. This adjustment for T_{wl} is achieved by adding the programmable delay element in the disable path of the wordline. The delay element is controlled using digital code provided by the "pulse width control" logic [112–114]. The pulse width control logic increases or decreases the digital code based on the BIST result. Therefore, the proposed system creates a closed feedback loop between the BIST, WL pulse width control and the memory internal timing, which can be used to reduce the power consumption as explained below.

Next, we present more details on the three main components of the proposed system: BIST, "WL delay" and "pulse width control" blocks.

5.3.1 SRAM Built-in Self-Test

Modern SoCs employ large embedded SRAMs which typically dominate the chip area. Due to their large size, memory arrays also become the dominant yield limiter. SRAM are prone to different types of failure, as described in Chapter 2. To address yield problems in SRAMs, large arrays are provided with redundant rows and/or columns, which can be used replace defected bitcells, hence, repairing the memory. To enable the repair, the memory should be tested first to locate the defected bitcells. However, since embedded memories lack direct access to chip input/output signal, embedded memory testing becomes complicated and time consuming.

Memory built-in self-test (BIST) is nowadays an integral part of SoC [105, 115]. Typically, a memory BIST engine is used to generate patterns to test a memory and detect memory failures. Based on the memory failure information, the unit under test is either discarded or repaired using memory redundancy. BIST not only

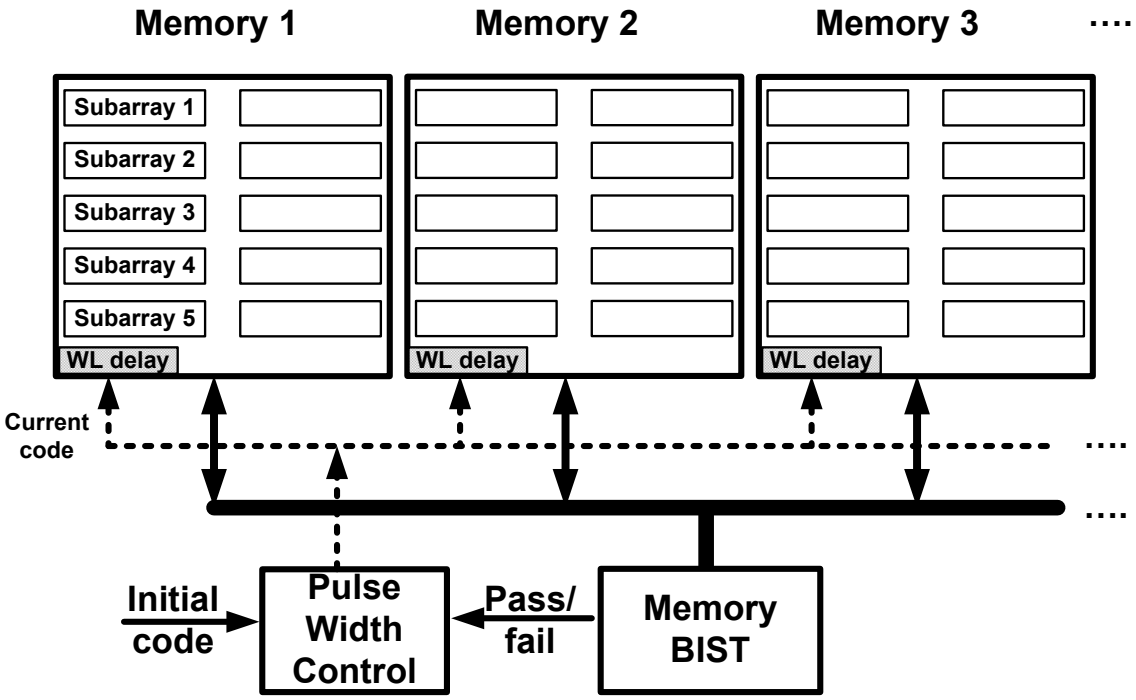


Figure 5.5: Proposed architecture: Fine-grained wordline pulse width control.

reduces testing time (and cost), but also allows testing the memory under actual clock speed (at-speed testing). Moreover, BIST can be used to enable built-in self-repair (BISR), where the BISR logic analyzes the failing addresses from BIST and generates a failure bitmap for memory. The BIST then uses the failure bitmap to replace failing bitcells using the redundant rows and columns. BISR is also used to enable soft repair instead of laser repair (hard repair) and without using automatic test equipment (ATE) which further reduces testing cost [105, 115]. Memory self-test can be performed every time the chip is reset (in start-up or power-up test mode).

5.3.2 WL Programmable Delay Elements

Wordline and sense amplifier timing are of utmost importance for correct read operation. The timer block shown in Fig. 5.2 is responsible of generating these critical internal timing signals. For SRAM post-silicon debugging purposes, and yield learning, programmable delay elements are used to control internal timing for wordline and sense amplifier [112–114]. For example, these delay elements are used to characterize the margin in bitline differential voltage. Moreover, they are used in relaxing address setup time requirements by delaying the clock edge which starts

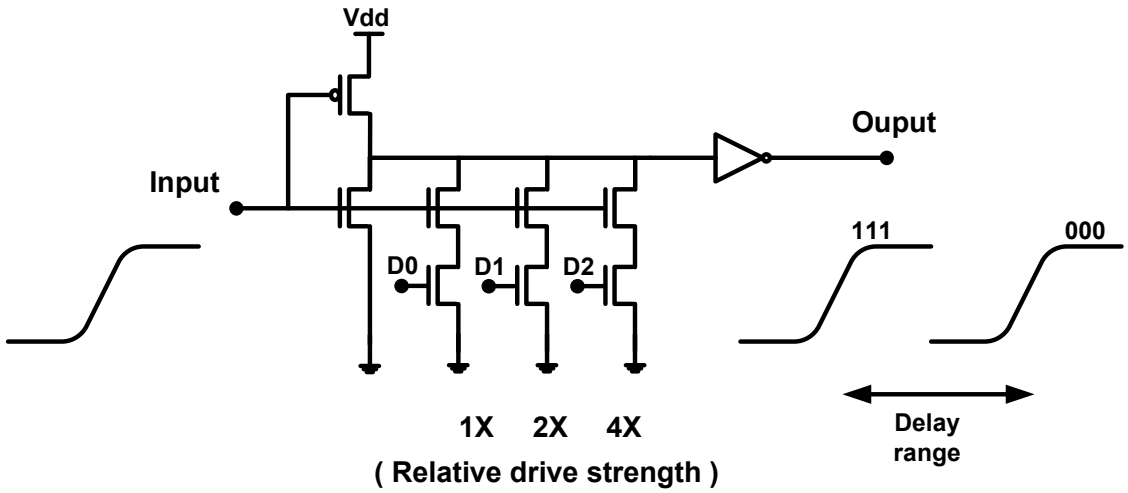


Figure 5.6: Programmable delay element [112].

an access [112]. Fig. 5.6 shows one type of programmable delay elements [112, 114].

5.3.3 Pulse Width Control Logic

Pulse width logic is the control logic for the proposed architecture. It can be as simple a digital counter, which varies the digital code for the delay element depending on the output of the BIST. It can also be implemented in software. For example, a programmable processor can be used to control both the BIST and the programmable delay elements. Fortunately, modern SoCs include programmable processors that can be used at start-up time to test and verify the operation of other modules [16, 52].

5.3.4 System Operation

Fig. 5.7 shows the operation of the proposed system: Initially, the pulse width control logic provides the initial code for the memory. This initial digital code will correspond to the required worst-case $T_{wl,wc}$ for a give yield requirement, which is determined in the design-time (using Eq. 5.5 or using the statistical flow described in Chapter 4). The BIST tests the memory instance using the initial code. If the memory fails, it may require repairing or it may be discarded, which is a typical BIST testing sequence. However, if the memory passes the BIST testing using the initial code, the BIST signals the pulse width control logic to reduce the digital code, hence reducing T_{wl} using the programmable delay element. Using the new digital code for T_{wl}, the BIST tests the memory once again, and this process of memory

testing and T_{wl} reduction is repeated until the memory fails in read operation [3]. The last passing code is then stored on the built-in registers inside the memory instances. If the lowest code is reached without the memory failing, the code is also stored and the operation is terminated. The above mentioned steps are repeated for all the memories in the chip. Hence, the proposed architecture reduces T_{wl} for all memories in which the bitcells have sufficient ΔV_{bl} that ensures correct read operation. Therefore, by reducing T_{wl}, read power of the memories can be reduced. This operation can be part of the system testing or power-up, where the final codes for each memory can be stored on built-in registers or burned on efuse (or any programmable ROM).

5.4 Results and Discussion

To test the power savings using the proposed architecture, Monte Carlo simulations are used to capture the impact of device variation on bitcell I_c, and the corresponding T_{wl}. Simulations are performed using 1Mb macro from an industrial 45nm technology. The 1Mb macro uses replica path to reduce power consumption and improve process tracking [14, 68, 102, 110]. Hardware correlated bitcells statistical models are used to compute μ_{I_c} and σ_{I_c} which are used in the simulation flow shown below. Post-layout switching power simulations were used to measure the power versus T_{wl} dependence, as shown in Fig. 5.1.

In SoC design, typically a high density macro (in the order of 512k or 1Mb) is used as a building block for larger size memories. Therefore, in our simulations, we assume that the minimum memory macro size is 1Mb, and multiple instances of that macro are used to realize larger memories in an SoC. We further assume that each 1Mb instance can have a specific T_{wl}.

To estimate power saving using the proposed system, a Monte Carlo simulation flow has been developed as follows:

1. For every memory instance in the chip, generate $Nbit$ samples of I_c normal distribution with μ_{I_c} mean and standard deviation of σ_{I_c} to represent the read current variation in the macro;

2. Find the minimum current in each memory instance and compute the $T_{wl,inst}$ from the information of the weakest bitcell which has the lowest I_c (Eq. 5.5).

[3] More efficient search algorithms such as successive approximation can be used to reduce the test time.

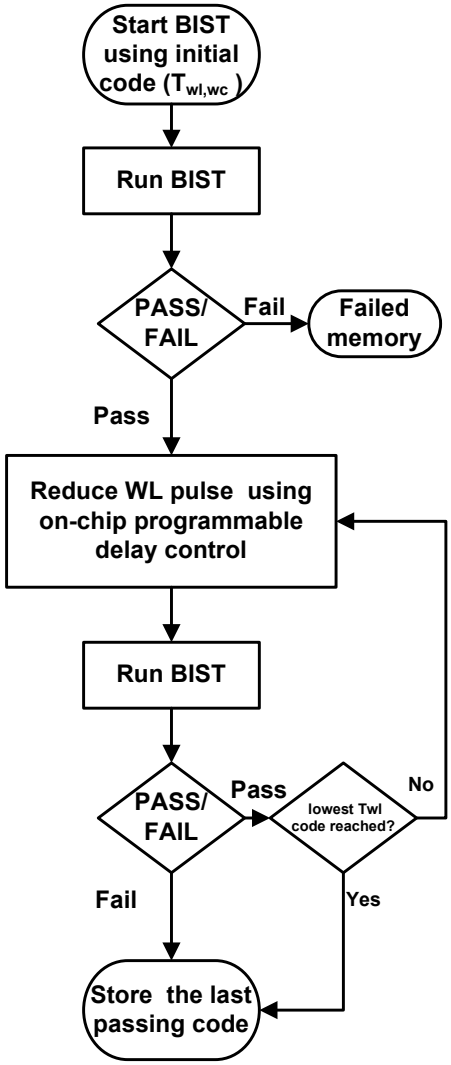


Figure 5.7: Flowchart for the operation of the fine-grained wordline pulse width control system.

Therefore, $T_{wl,inst}$ represents the minimum wordline pulse width that guarantees correct read operation for *that instance*. This value of $T_{wl,inst}$ should be automatically determined by the proposed system, since the wordline control block shown in Fig. 5.5 will reduce $T_{wl,inst}$ until that memory instance fails a read operation;

3. Using Eq. (5.4) or power versus T_{wl} curves as in Fig. 5.1, calculate the power consumption of that memory instance;

4. Repeat all the above steps for all memory instances on that chip and compute the total read power for memories on that chip[4];

5. Repeat all the above steps for a large number of chips and get the average power.

6. From the chip level yield target and the total memory density, calculate the design coverage using Eq. (5.6) and find the worst-case $T_{wl,wc}$. This will be the value of T_{wl}, which would have been used for all memory instances if the proposed architecture was not enabled. For $T_{wl,wc}$, the corresponding power consumption P_{wc} can be calculated using Eq. (5.4);

7. From the last two steps, calculate the power reduction using the proposed architecture.

Fig. 5.8 shows the power reduction achieved using the proposed architecture for different memory densities and different yield targets. Note that these yield values represent the intrinsic read acces yield before applying any repair or correction (*i.e.*, redundancy or ECC). As the memory density increases, the power saving increases, which shows the effectiveness of the proposed system especially if high yield target is required (as in high volume, low cost products). Array switching power consumption can be reduced between 15% and 35% for a 48Mb memory density depending on the yield target. Fig. 5.9 shows the achievable power saving versus memory yield target. For a 1Mb memory density, the array switching power savings can be as high as 15% for a yield of 99%. From Fig. 5.8 and Fig. 5.9 it is clear that the proposed architecture reduces array switching power significantly. As the SRAM content increases in an SoC, it is expected that higher power saving can be achieved using the proposed architecture.

[4]Here we assume that all memories are accessed simultaneously. Hence, we add the individual read power of each memory. For our analysis, this is a fair assumption since we do not make any assumptions related to how the system accesses these memories. Nevertheless, the same simulation flow can be used if the switching activity for each memory is known.

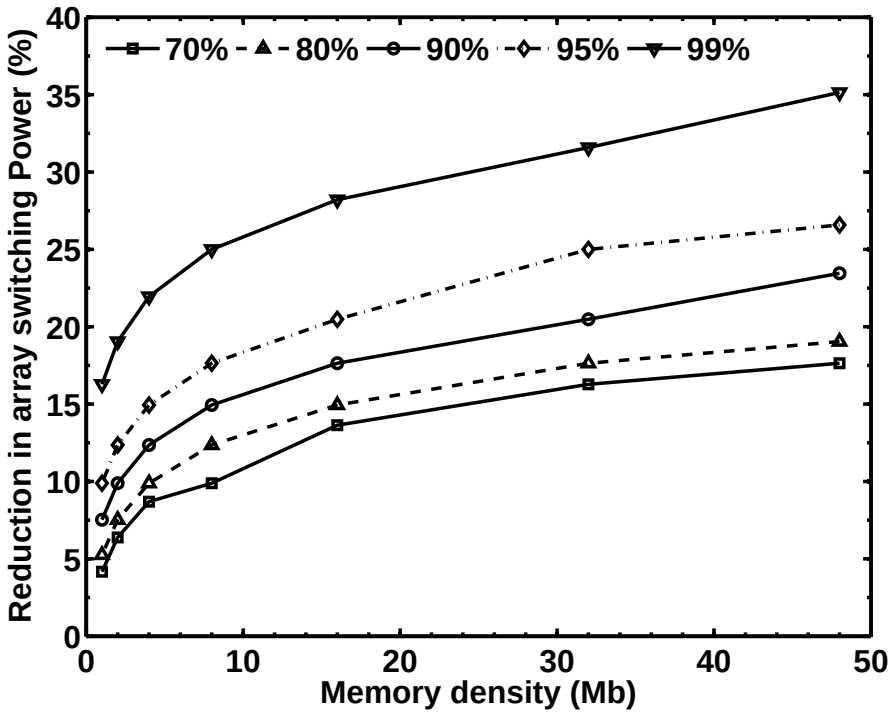


Figure 5.8: Power reduction using the proposed architecture versus chip level memory density. Different values of yield targets are shown.

SRAM bitcells show strong sensitivity to device variations, which increase significantly with scaling. To investigate the impact of variations on the power, we increase I_c variations from 10% to 12.5% [62]. Fig. 5.10 shows power reduction for these two cases of I_c variation. It is obvious that significant power savings can be achieved. For the 48Mb case, power saving increases by more than 2X, from 25% to 55% (even though I_c variations only increased from 10% to 12.5%). This shows the effectiveness of the proposed architecture, which makes it attractive for power reduction in the presence of large process variations which are expected to worsen with technology scaling.

For the previous simulation results, we assumed that a 1Mb memory is the minimum memory instance size that can have a specific T_{wl}. However, in memory design, multiple subarrays (*i.e.*, banks) are used to implement a single memory instance, as shown in Fig. 5.5. Typically, a timer circuit is used per subarray, hence, fine-grained concept can be further applied to the sub-array level. This requires adding the digitally controlled delay element per subarray, and have the capability of storing the digital code per subarray. Nevertheless, the area overhead can still be very small since the additional area is amortized over the large size of

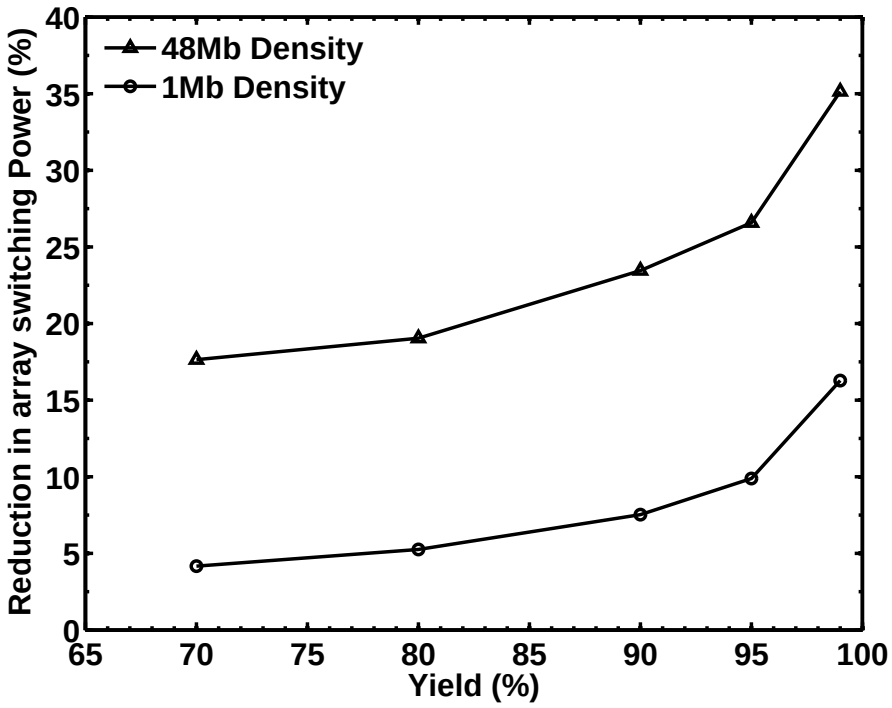


Figure 5.9: Power reduction using the proposed architecture versus yield target for two cases of chip level density a)1Mb and b)48Mb.

the memory macro size. To evaluate the benefits of T_{wl} control at the subarray level, we assume that we have 16 subarrays in the 1Mb macro. Each subarray is composed of 256 bitlines by 256 rows. Therefore, using the proposed architecture, T_{wl} can be adjusted for a 64kb block. Fig. 5.11 shows the achieved power saving for the 64kb block size as well as the 1Mb full macro. By adjusting T_{wl} at the subarray level, power saving increases from 24% to 42% for the same 48Mb chip level memory density (1.75X improvement). Also, for the 1Mb chip level memory density, power saving increases from 7.5% to 20% (2.67X improvement). This shows the importance of reducing the size of the memory block which can be individually controlled using T_{wl}, as this will significantly reduce the array switching power consumption.

The proposed architecture reduces power consumption by reducing the wordline pulse width for memories which have enough margin for read operation. It is important to investigate how reducing T_{wl} will affect the system timing for both setup time and hold time requirements. Reducing T_{wl} also reduces the access time of the memory T_{cq}, which implies that the logic delay through the memory is now faster. Therefore, the setup time requirement will be automatically met. Reducing T_{cq}, however, may cause hold violations. This situation can be easily resolved in

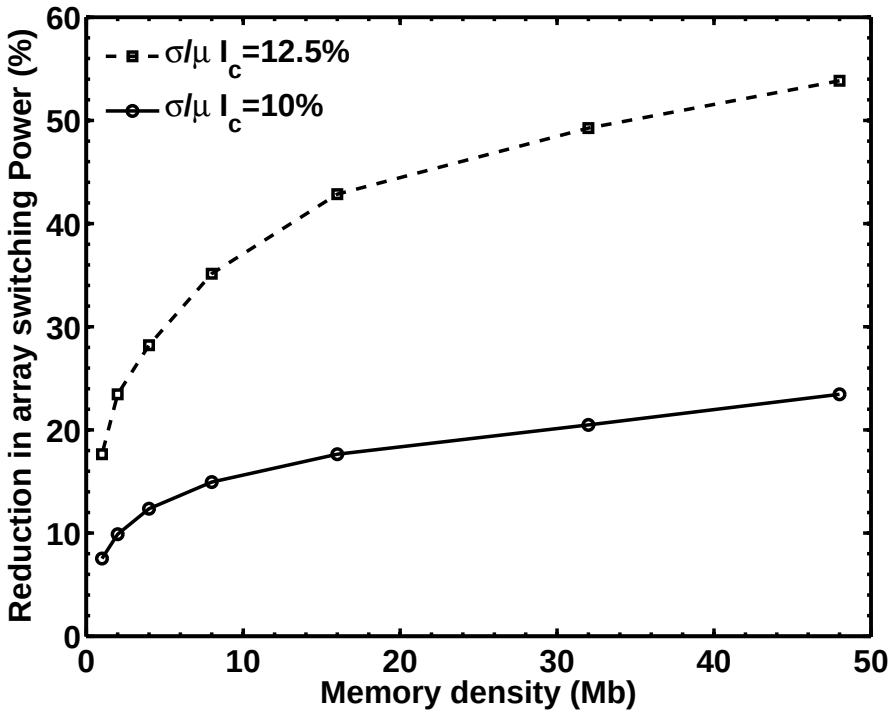


Figure 5.10: Power reduction using the proposed architecture versus chip level memory density for different values of I_c variation for a yield target of 90%.

design-time by using the lowest expected T_{cq} for hold time verification (*i.e.*, using the minimum T_{wl} that the programmable delay element provides).

In the above analysis, the discrete quantization effect on power savings is not considered. In reality, since we are using a digitally controlled delay elements, T_{wl} can only take discrete values. However, this may not have a significant impact on the shown results, since small area and low power delay elements can cover large range of delays with fine control [112,113]. In addition, by using the proposed Monte Carlo simulation flow, we can determine the range of T_{wl} with highest probability of occurring, and modify the delay elements in design-time to have enough steps in that region. It is important to note that delay elements do not add extra area as they are typically used in memories for debugging purposes [112, 113].

While the proposed system accounts for process variations which are static in nature, it cannot adapt to environmental variations such as voltage or temperature variation [19] due to their dynamic nature. This is because T_{wl} will be fixed after running the pulse width control system in the start-up. Hence, environmental variations may cause the memory to fail if they reduce the sensing margin. This problem can be addressed in design time, by ensuring that the minimum step size

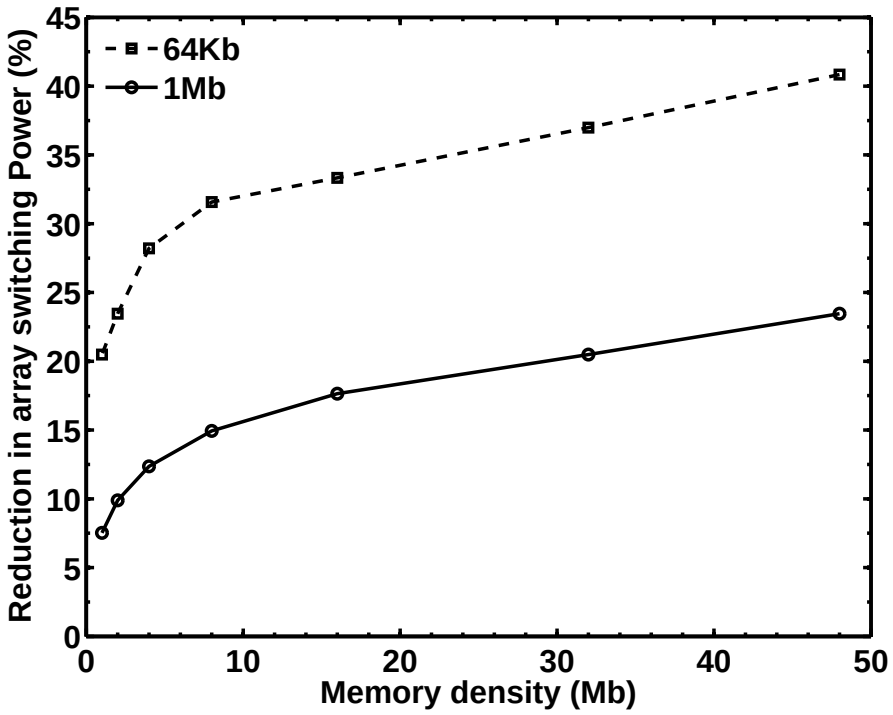


Figure 5.11: Power reduction using the proposed architecture versus chip level memory density for different values of the minimum controlled memory instance (or subbanks) for a yield target of 95%.

for delay control provides sufficient margin for voltage and temperature variations. Moreover, if self-timed memories are used, the replica path will provide efficient tracking for environmental variations [14, 68, 102, 110]. In addition, in product testing, low voltage screening can be used in start-up to set T_{wl}, which guarantees a sufficient margin for environmental variations, since the product will operate at a supply voltage typically larger than the low voltage test condition.

In this work, we presented the analysis and results only for read power reduction. Nevertheless, the proposed system can also be used to reduce switching power in write operation as well. In write operation, although selected bitlines are fully pulled down to ground, the half-selected bitcells (bitcells on the same selected wordline) still experience bitline discharge similar to read operation. Hence, array switching power due to half-selected bitcells can contribute significantly to the total write power especially in high density memories with small IO width and large muxing option. Therefore, the proposed architecture can be applied for write power as well. In that case, wordline pulse can be reduced until a write failure occurs [12, 13].

5.5 Summary

Array switching power is one of the largest components of power consumption in high density SRAM. Moreover, with the large increase in process variation, SRAMs are expected to consume even larger array switching power to ensure correct read operation and meet yield targets. In this chapter, we proposed a new architecture that significantly reduces array switching power. The proposed architecture combines BIST and digitally controlled delay elements to reduce the wordline pulse width for memories while ensuring correct read operation, hence, reducing switching power.

Combining both architecture and circuit techniques enables the proposed system to detect failing bitcells using the BIST and adjust T_{wl} accordingly. Therefore, the proposed architecture recovers the power consumption since it tests each memory individually and adjusts T_{wl} to ensure correct read operation. A new statistical simulation flow was developed to evaluate the power savings for the proposed architecture. Monte Carlo simulations using a 1Mb SRAM macro from an industrial 45nm technology was used to examine the power reduction achieved by the system. The proposed architecture can reduce the array switching power significantly, and show large power saving especially as the chip level memory density increases. For

a 48Mb memory density, a 27% reduction in array switching power can be achieved for a read access yield target of 95%. In addition, the proposed system can provide larger power saving as process variations increase, which makes it a very attractive solution for 45nm and below technologies.

Chapter 6

A Robust Single Supply Voltage SRAM Read Assist Technique Using Selective Precharge

In this chapter, we present a new read assist technique for SRAM to improve bitcell read stability. The new technique utilizes selective precharge where different parts of the bitlines are precharged to V_{DD} or GND. Using charge sharing, the required value of bitline voltage can be precisely set to increase bitcells' SNM, while using only one supply voltage. A 512kb memory was designed to demonstrate this technique in an industrial 45nm technology. Results show large improvement in SNM and high robustness against process variations. In addition, the proposed technique reduces the memory access time compared to the conventional approaches. Section 6.1 discusses state of the art read assist techniques. In Section 6.2, we discuss the relationship between bitline precharge voltage and bitcell SNM. In Section 6.3, we describe the proposed read assist technique and its operation, and in Section 6.4 we show how the proposed circuit can improve the memory's access time. In Section 6.5, we provide the design details and results for the proposed read assist technique applied on the design of a 512kb memory in an industrial 45nm technology. Finally, in Section 6.6, we summarize our findings.

6.1 State of the Art Read Assist Techniques

In today's SoCs, there are stringent requirements to achieve low power consumption, and in the meantime achieve higher speeds. Voltage scaling combined with

technology scaling has been effective in targeting both requirements. However, the large increase in random variations in advanced CMOS technology nodes is creating huge challenges for SRAM design. This is exacerbated by the high demand for low voltage and high density memories for SoC [116]. Dealing with SRAM cell stability at lower supply voltages is currently one of the biggest challenges in SRAM design [13, 117, 118].

As discussed in Chapter 2, bitcell stability is defined depending on the SRAM operation mode (whether read or write). In read operation, the static noise margin (SNM) is used as the measure of SRAM robustness and is defined as the maximum internal noise voltage that the bitcell can tolerate [74]. It has been shown that SNM limits the bitcell stability and dictates the minimum supply voltage at which the memory can operate [61, 68, 71].

Recently, there has been extensive research to improve SRAM bitcell stability. Table 6.1 lists state of the art read assist techniques implemented in 90nm down to 45nm technology. These works improve SRAM stability through bitcell and/or periphery circuit design. For example, 8T bitcell was proposed as an alternative for 6T bitcell as a way to improve bitcell robustness by decoupling read and write paths, which is more like a register file (RF) or multi-port memory [119–121]. However, 8T bitcell show significant increase in bitcell area ($\sim$ 20%) which may limit its adoption as a 6T bitcell replacement.

Other read assist techniques can be broadly classified into two categories, single supply and dual supply technique. In dual supply approach, one power supply is used for the bitcell array, and a different supply voltage is used for the periphery circuit [73, 122–124]. In this way, the SRAM's supply voltage can be kept at a relatively higher voltage compared to the logic. The logic's supply voltage can be scaled to reduce power consumption, while the SRAM's supply is kept constant. This ensures that the SRAM read stability is sufficient, since SRAM SNM failures can be eliminated at higher supply voltage (at higher V_{DD} the impact of WID variations reduces). Other approaches of dual supply uses two voltage levels to control body bias.

Although dual V_{DD} concept seems relatively simple, however, it introduces significant challenges. First, voltage level shifters are required at the interface between the bitcell array voltage and logic voltage. These level shifters tend to consume large area which lowers the memory's area efficiency. In addition, level shifters introduce additional delay in the memory critical path, hence, can cause speed penalty. Moreover, the power grid design at the chip level becomes challenging since a dedicated

power grid is required for memories. It is important to note that difficulty in implementing a dual supply power grid depends on the chip architecture. For example, microprocessor designs uses relatively few kinds of SRAM architectures having large capacity such as caches, which can be physically placed in a close proximity on a chip [75]. This simplifies the design of a dual power grid since all the memories are physically located near each other. However, in SoC design, there are typically hundreds of SRAM architectures and they are not necessarily placed in close proximity on a chip. This makes it difficult to have a dedicated power grid for all the memories. Therefore, for an SoC, it is always desirable to use a single power supply for the SRAM [68, 75, 125].

In dual V_{DD} read assist techniques, a column-based dynamic power supply was proposed in [126]. The bitcell's dynamic dual supply is switched during read and write operations at a column level. Therefore, it decouples the read and write operations and improves cell stability. In [122], embedded level shifters instead of conventional level shifters are used to reduce level shifters area. A similar implementation for an L2 cache using dual V_{DD} was shown in [127]. In [123], a high performance domino type read and write circuitry are used in dual V_{DD} approach, which improves cell stability and supports very high speed operation (6 GHz). In [73], dual V_{DD} is used to control body bias of the PMOS pull-up (PU) devices in a column, and in read operation the PMOS body is forward biased to reduce its V_{th} which increases the bitcell stability. In write, the PMOS body bias is reversed, hence, V_{th} increases for the PU device, and write margin improves. In [128], a dual V_{DD} approach is used to control the WL voltage and the bitcell array voltage. In read operation, WL voltage is lower than the array voltage which increases SNM (pass-gate drive capability reduces). In write operation, the WL voltage is higher than the array voltage which improves the bitcell write margin. Similar approach was propped in [129], but implemented on body tied SOI technology. However, local WL drivers are required which increases the memory area substantially.

Table 6.1: State of the art read assist techniques.

Read Assist Technique	**Technology Node**
8T bitcell [119]	90nm
8T bitcell [120]	65nm
8T bitcell [121]	65nm SOI
Dual V_{DD} PMOS body bias control [73]	90nm
Dual V_{DD} WL voltage and PMOS body bias [128]	90nm
Dual V_{DD} with integrated column based dynamic supply [126]	65nm
Dual V_{DD} with embedded level shifters [122]	65nm
Dual V_{DD} with domino read/write circuitry [123]	65nm SOI
Dual V_{DD} for read and write [127]	65nm
Dual V_{DD} using PMOS body bias [129]	65nm SOI
Dual V_{DD} for read and write [130]	45nm
Single V_{DD} read and write back [64]	65nm
Single V_{DD} pulsed wordline and bitline [131]	65nm
Single V_{DD}/ dual V_{DD} [63, 117]	65nm SOI
Single V_{DD} adaptive WL voltage for read [75]	45nm
This work : Single V_{DD} selective precharge [132]	45nm

In single supply read assist techniques, additional circuitry is added to assist read operation and provide adequate read stability (SNM). To mitigate the impact of variations on bitcell SNM, several single supply read assist techniques have been proposed [64, 75, 117, 131].

A read and write back technique was proposed in [64], where a voltage latch sense amplifier is integrated per column. This SA is used to read the data from the bitcell and write it back at the end of read operation. Hence, it provides data recovery by writing back the original data. This technique increases power consumption since every column undergoes full signal amplification. Moreover, it has large area overhead since a SA is integrated per column and cannot be shared between several columns as in modern SRAM design. In addition, mismatch in the SA may cause the bitcell to write incorrect data, hence, corrupting the stored one. In [75], an adaptive WL approach is used where WL voltage varies depending on D2D and temperature variations. Replica transistors are used to control the WL voltage level, hence, improving process tracking and increasing SNM. However, the design increases the memory access time, since the reduction of WL voltage reduces the bitcell read current.

In [63, 117, 131], the bitline precharge voltage is reduced before the bitcell is accessed, hence, increasing SNM. More details about this approach will be discussed in the next section.

6.2 Background

In SRAM design, the bitlines are typically precharged to V_{DD} before accessing the bitcell. However, in [63], it was shown that reducing the bitline voltage V_{BL} before accessing a bitcell improves the bitcell's read stability, as shown in Fig. 6.1. This is because the pass-gate (access transistor) strength reduces as the bitline voltage decreases. This effectively increases the bitcell α ratio (defined as the ratio of pull-down to pass-gate strength). Hence, SNM improves as ΔV_{BL} increases where ΔV_{BL} is defined here as $V_{DD} - V_{BL}$ just before the wordline (WL) is asserted. Note that as ΔV_{BL} increases[1], SNM reaches a maximum point, and further increase in ΔV_{BL} causes significant SNM reduction. This is mainly due to read disturbs. Therefore, accurate control of ΔV_{BL} level is important to prevent ΔV_{BL} from exceeding the maximum SNM point.

From a circuit point of view, the relation between ΔV_{BL} and SNM has been exploited in [117, 131] to increase bitcell stability. In [131], a pulsed bitline approach is used where a pulse is presented to control an NMOS pull-down. This pull-down device discharges the bitline, which increases ΔV_{BL} just before the WL is enabled. However, this technique is sensitive to PVT variations since ΔV_{BL} is a strong function of the pulse duration which will vary with PVT variations. Therefore, a complex timing scheme may be required to control ΔV_{BL} at different PVT conditions. In [117], an NMOS device is used to precharge the bitlines, hence, having one V_{th} drop on the bitlines. Due to the strong sensitivity of V_{th} to PVT variations, the effectiveness of this technique reduces in different PVT corners. Moreover, low V_{th} devices are required to ensure that ΔV_{BL} does not cause read disturbs in the worst-case conditions, which adds additional processing cost (especially for low cost SoCs).

[1] ΔV_{BL} in this chapter should not be confused with the bitline differential voltage ΔV_{bl} used in Chapter 4 and Chapter 5. Here, ΔV_{BL} is the reduction in bitline precharge level before accessing the bitcell.

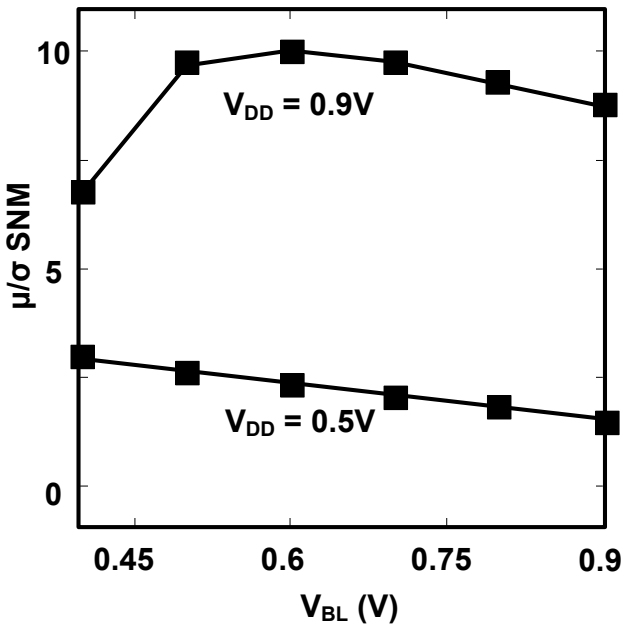


Figure 6.1: $\frac{\mu}{\sigma}$ SNM versus bitline precharge voltage. Lower bitline voltage during a read access improves bitcell read stability (SNM) [63].

6.3 Selective Precharge Technique

To overcome the obstacles in implementing dual supply read assist techniques, we focus on using single supply voltage. Moreover, we try to improve the robustness of previous single supply read assist techniques.

Instead of precharging the bitlines to V_{DD}, in our proposed technique, different parts of the bitlines are precharged to V_{DD} or (predischarged to) GND. Using charge sharing, the final required value of bitline voltage can be precisely controlled. This technique relies on the capacitance ratio to control the bitline voltage. Therefore, this technique shows high immunity against process variations (both front-end and back-end) since the capacitance ratio shows weak dependence on PVT corners.

Fig. 6.2 shows a simple schematic for the selective precharge technique with four bitline columns connected to the read and write circuitry (sense amplifier and write drivers). Bitlines, BL/BLB, refer to the upper part of the bitlines connected directly to the bitcells (before the column select). Sense/Write lines, SL/SLB, refer to the lower part of the bitline connected to the sense amplifier and write drivers (after the column select).

Selective precharge operation can be divided into three main steps. First, BL/BLB are precharged to V_{DD} as in conventional approaches, while SL/SLB is

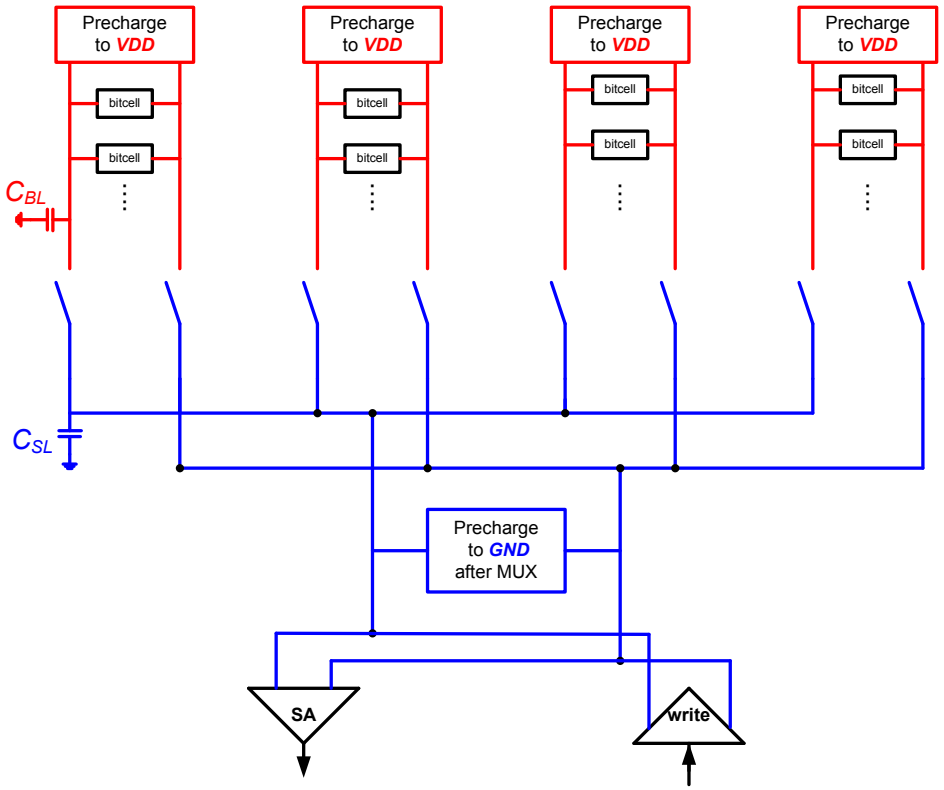


Figure 6.2: Selective Precharge operation. Step 1: Precharge to V_{DD} and GND.

precharged (*pre-discharged*) to GND, as shown in Fig. 6.2. In the second step, the column select devices (MUX) on each bitline column are enabled. Hence, charge sharing occurs between the upper and lower bitlines. Note that $BL0 - BL3$ all experience charge sharing. The final bitlines voltage after charge sharing is determined by the capacitance ratio of upper and lower bitlines (BL and SL). Using charge sharing, the bitline voltage can be reduced. Therefore, SNM improves as discussed in Section 6.2. In the third step, the MUX devices for all unselected columns are disabled, while the selected column MUX stays on. In this case, the selected column allows access to the required bitcell, while half-selected[2] bitcells also see improvement in SNM since their bitline voltages have also been reduced.

Fig. 6.5 shows the implementation of selective precharge technique. A NOR gate is added for each bitline column to control the column select. Fig. 6.6 shows the precharge circuits for both V_{DD} and GND. Fig. 6.7 shows the timing diagram for selective precharge operation. *ch_sh* is activated using the rising edge of precharge disable (for PMOS pull-up). When *ch_sh* is high, the PMOS devices in the column

[2]Half-selected bitcells: bitcells on the same row where WL is asserted, although they are not accessed for read or write operations.

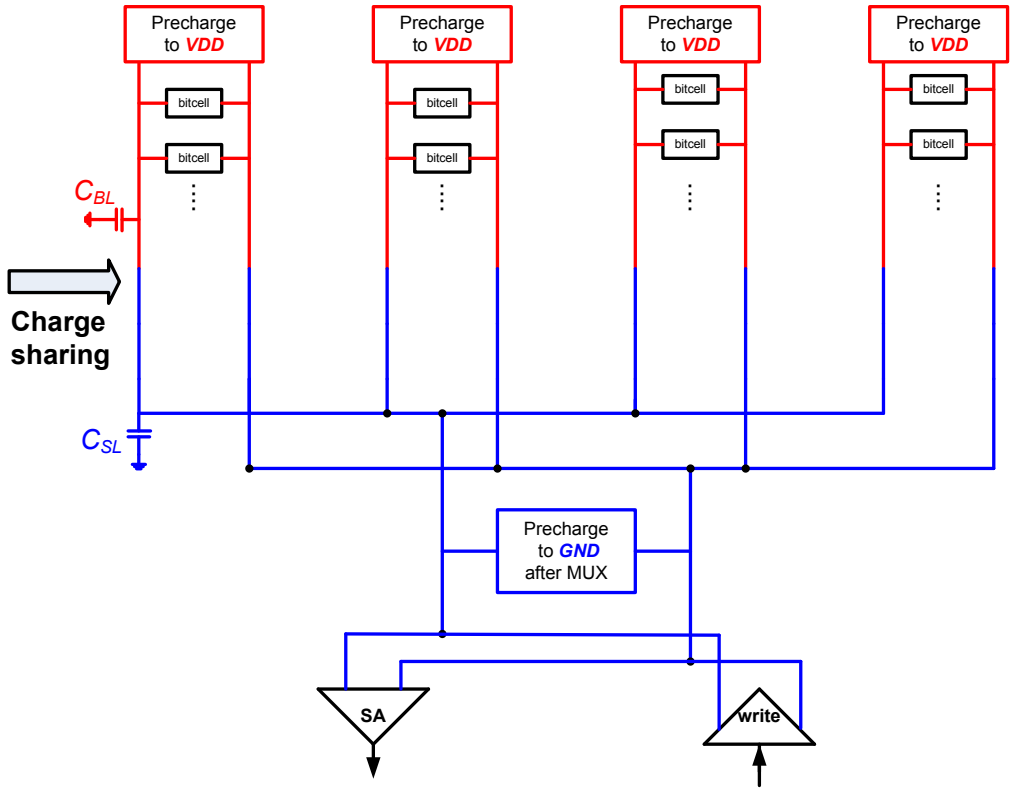


Figure 6.3: Selective Precharge operation. Step 2: Charge sharing.

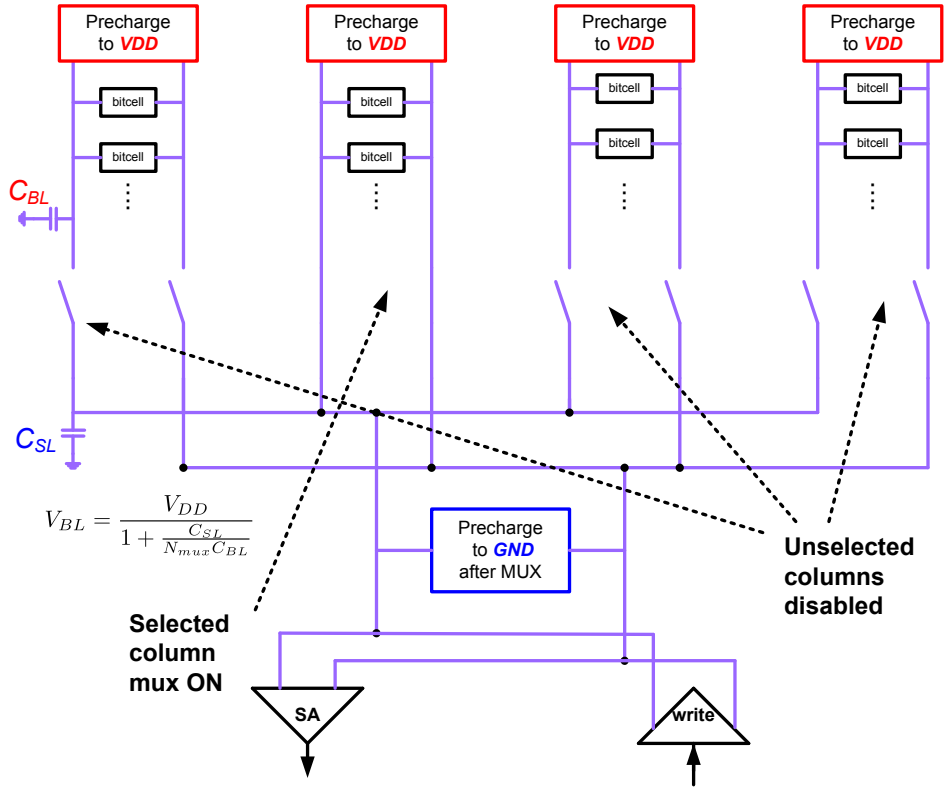


Figure 6.4: Selective Precharge operation. Step 3: Unselected columns disabled.

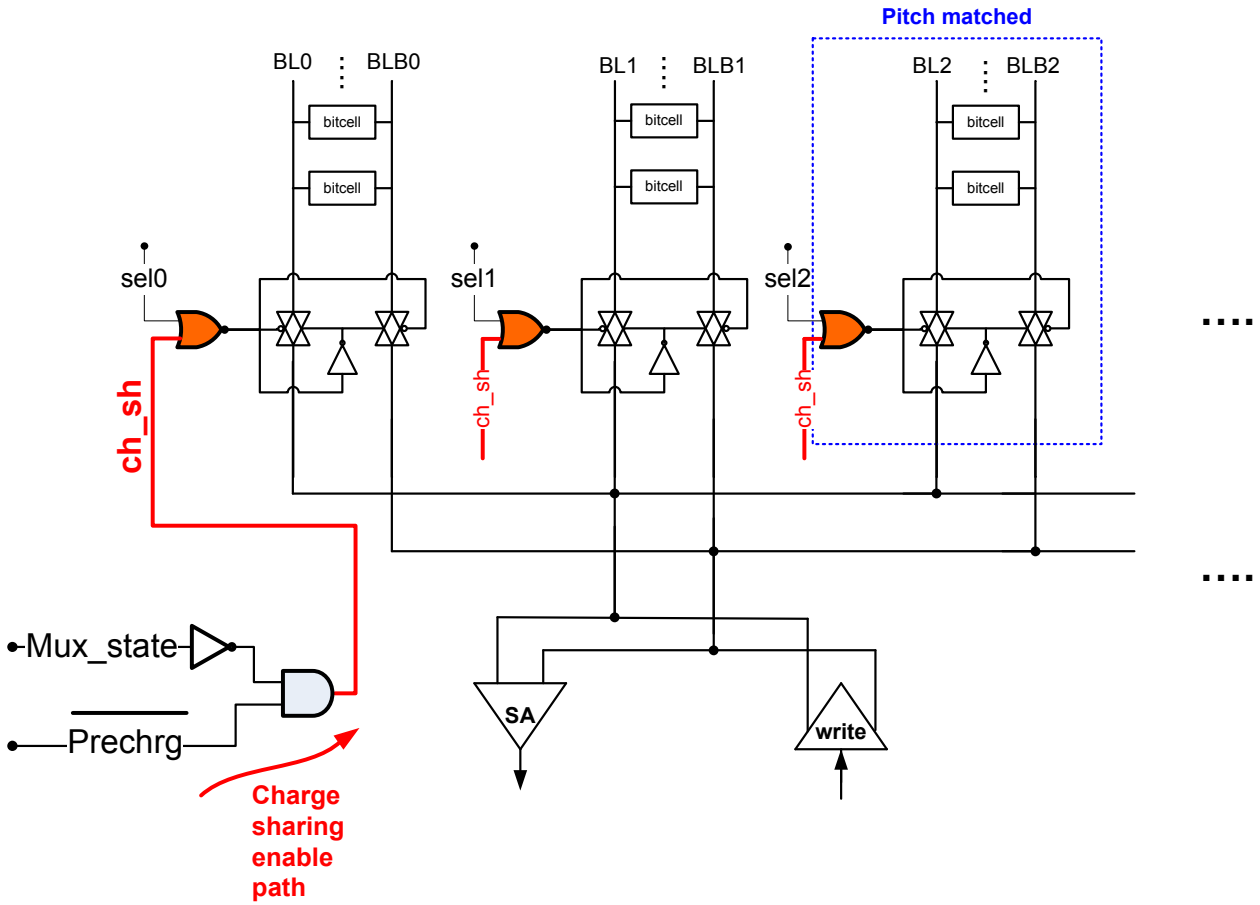


Figure 6.5: Selective precharge schematic.

select MUX are on. Hence, charge sharing between all bitlines sharing the same read/write circuitry and SL/SLB line is enabled. Therefore, BL/BLB voltage decreases while SL/SLB voltage increases. ch_sh is disabled using mux_state which is a dummy column select signal. Therefore, bitcells see reduced bitline voltage when the bitcell is accessed. At the end of operation, BL/BLB are precharged back to V_{DD} while SL/SLB are precharged to GND, as shown Fig. 6.7.

By selecting the location of precharge to V_{DD} or GND, the required value of ΔV_{BL} is set. For example, if a larger ΔV_{BL} is required, one or more of the bitlines can be precharged to GND instead of V_{DD}, as shown in Fig. 6.8. Therefore, the proposed technique allows changing ΔV_{BL} by selecting which points to be precharged to V_{DD} or GND. Note that in this technique, no additional supply voltages are required to generate the desired bitline voltage, which reduces the design complexity. In addition, since the final ΔV_{BL} voltage depends solely on capacitance ratio, its value is very robust against process variations.

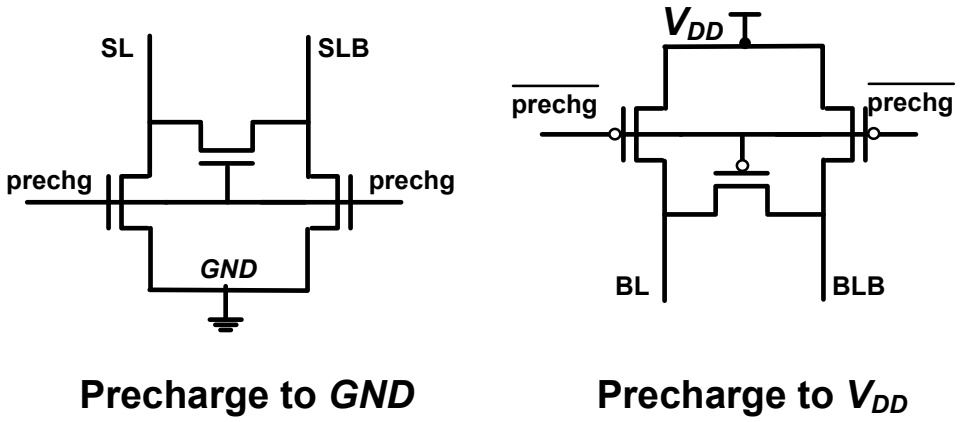


Figure 6.6: Precharge to V_{DD} and GND circuits, including equalize transistors.

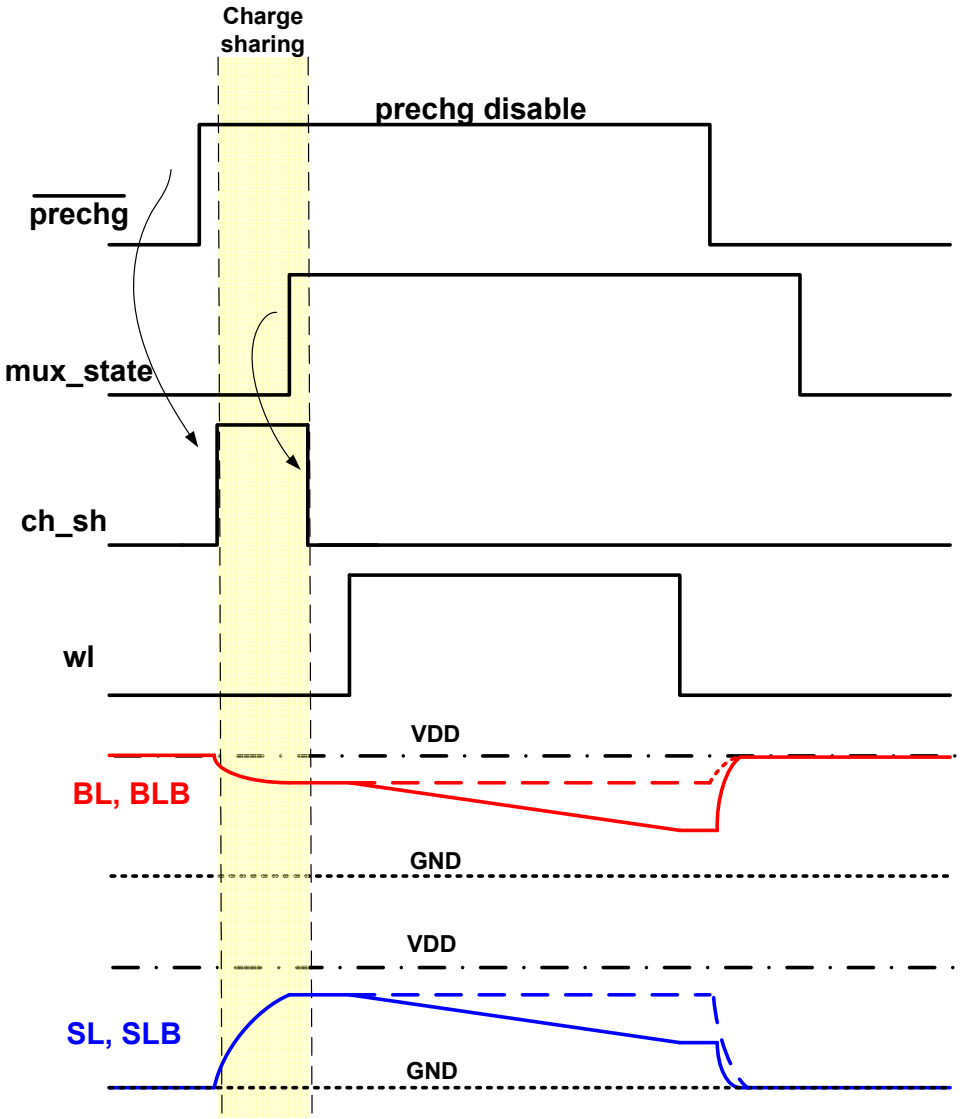


Figure 6.7: Selective precharge timing diagram.

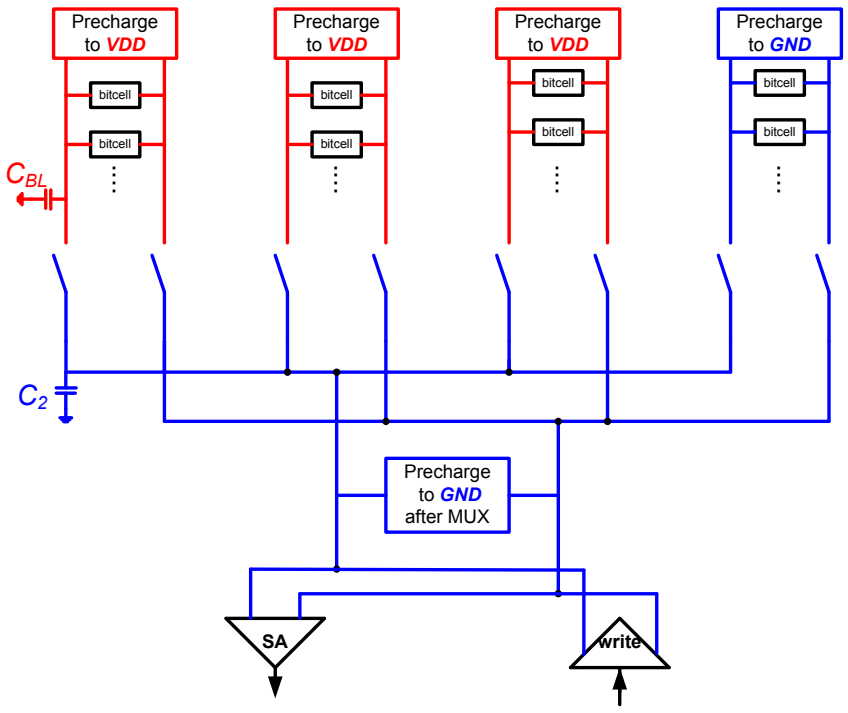


Figure 6.8: Achieving larger range of ΔV_{BL} using by precharging on of the bitlines to GND.

6.4 Access Time Improvement

In SRAM, the read operation determines the access time of the memory. The clock to WL enable contributes to memory's access time. Hence, it is usually optimized for speed. However, the proposed technique introduces another signal, *ch_sh*, which should be enabled before the WL is asserted. To accommodate the *ch_sh* signal shown in Fig. 6.7, the WL enable path may be delayed. This delay will therefore increase the memory access time. Hence, a technique to reduce (or recover) access time is required.

In addition to the clock to WL delay, another contributor to the memory's critical path is the WL pulse width T_{WL}. T_{WL} is the time required for the bitcell to discharge the bitlines and generate sufficient input differential for the sense amplifier to allow correct read operation. T_{WL} typically contributes to approximately 30% of the memory access time [73]. To reduce this delay component, we exploit the relation between ΔV_{BL} and the SA input offset.

There are many types of sense amplifiers used in SRAM design. However, current latch sense amplifier (CLSA) is one of the most widely used due to its high speed and isolation as discussed in Section 4.3.2 (CLSA shown in Fig. 4.5).

Moreover, it has been shown in [99] that reducing bitline voltage (common mode) improves the SA robustness. This characteristic of CLSA makes it very attractive in the proposed selective precharge technique. By reducing the bitline voltage (increasing ΔV_{BL}) the SA offset ($\sigma_{SA,offset}$) reduces, hence, allowing T_{WL} to be reduced for a give failure probability. The reduction in T_{WL} can therefore compensate for the increase in clock to WL delay, as will be shown in Section 6.5.

6.5 Results and Discussion

To test the proposed read assist technique, a full-custom 512kb SRAM was designed and implemented in an industrial 45nm technology as shown in Fig 6.9. Table 6.2 shows details on the memory architecture. In this section, we show post-layout simulation results for the proposed read assist technique.

Table 6.2: 512kb memory design information.

Technology	45nm low power (LP) CMOS
Density	512kb
Memory width (word size)	64 bits
Memory depth (number of words)	8192 words
Banks	16 (32kb each)
Rows/bank	128
Columns/bank	256

In Fig. 6.10, the read operation is shown for a bitcell on the first column (enabled using MUX0). In the beginning of the operation, $BL0$ and $BLB0$ are set to V_{DD} while SL and SLB are set to zero. Charge charing operation is activated using ch_sh, which activates all the MUX transistors. Therefore, $BL0/BLB0$ voltage decrease while SL/SLB increases as shown in Fig. 6.10, and they settle to a value determined by the capacitance ratio. Note that charge sharing happens quickly and that it is not sensitive to the ch_sh pulse width (wider pulse does not affect the settling voltage after charge sharing). After charge sharing is completed, the MUX devices (PMOS) for all unselected columns are disabled (MUX1), while the selected column stays selected (MUX0). Therefore, when WL is asserted, the accessed and half-selected bitcell see a reduced bitline voltage, which increases the bitcell's SNM. At the end of read operation, the bitlines are precharged to V_{DD} while the sense lines are precharged to GND.

The impact of bitline voltage on the CLSA speed and input offset is shown in

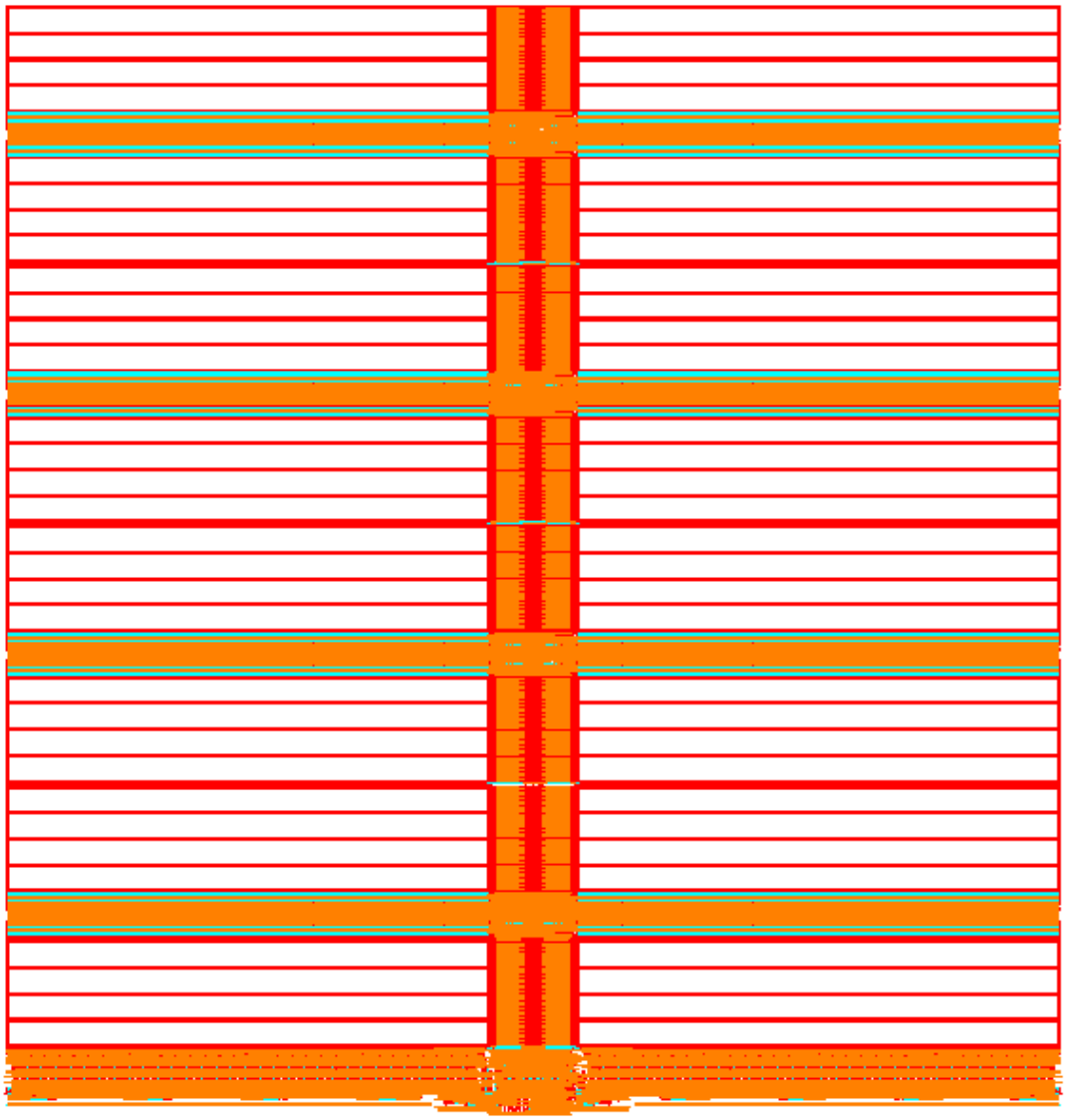


Figure 6.9: Layout of the designed 512kb memory in 45nm technology.

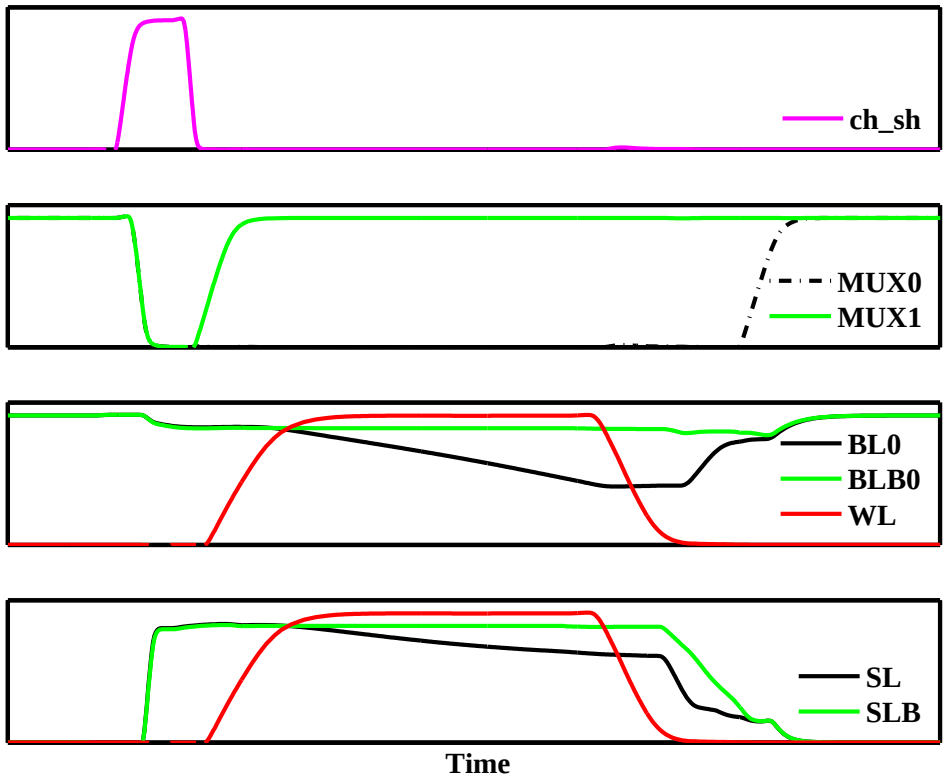


Figure 6.10: Results for selective precharge read operation. MUX0/1 are the gates voltages for the PMOS devices in the column select for column 0 and 1, respectively.

Fig. 6.11. Monte Carlo transient simulations were used to measure the SA's offset. As ΔV_{BL} increases, the SA delay slightly decreases until it reaches a minimum point. Beyond that point, the SA delay increases. In the meantime, the SA input offset (σ_{offset}) decreases monotonically with the increase in ΔV_{BL}. The reduction in σ_{offset} improves the robustness of the SA and decreases the probability of read access failures. Therefore, the WL pulse width can be reduced accordingly based on the following:

$$\frac{T_{WL2}}{T_{WL1}} = \frac{\sigma_{SA,offset_2}}{\sigma_{SA,offset_1}} \tag{6.1}$$

where T_{WL} is the time allowed for bitcell to generate bitline differential before enabling the SA. This large reduction in σ_{offset} reduces access time of the memory. Typically, T_{WL} is about 30% of memory access time. As shown in Fig. 6.11, SA offset can be reduced by up to 25%. Therefore, access time improves by 7%. In reality, to accommodate the *ch_sh* pulse, the WL enable path may be slightly delayed. Hence, this improvement in speed is reduced. Nevertheless, since charge sharing requires very short time, the impact on access time improvement is negligible. Therefore, in the worst-case scenario, the proposed technique will allow having the same access time as in conventional approach or better depending on the implementation.

Charge sharing operation is also enabled when a bitcell is accessed for write operation, as shown in Fig. 6.12. In that case, half-selected bitcells experience reduced BL voltage to improve read stability (bitlines $BL1$ and $BLB1$). However, write trip point degrades due to lower drive capability of the pass-gate [127]. This may reduce bitcell WM which may cause a write failure. To improve the writeability of the selected bitcell, we use a CMOS write driver for write operation, as shown in Fig. 6.13. Therefore, BL voltage lost in charge sharing is recovered using the pull-up device in write driver. Hence, the write margin for the selected bitcell in write operation is not deteriorated.

To estimate the bitcell SNM, we simulate the butterfly curve as shown in Fig. 6.14, which shows the bitcell SNM for a nominal bitcell which does not include WID variations. However, as discussed in Chapter 2, WID variations will cause each transistor in the bitcell to have different V_{th}, which will cause the bitcell be be asymmetric. This is shown by the Monte Carlo simulation results in Fig. 6.15, which shows large spread in the VTC characteristics of the bitcell. This spread translates to large variation in SNM.

The improvement in SNM using the proposed technique is shown in Fig. 6.16. Monte Carlo simulations are used to measure the impact of local variation on the

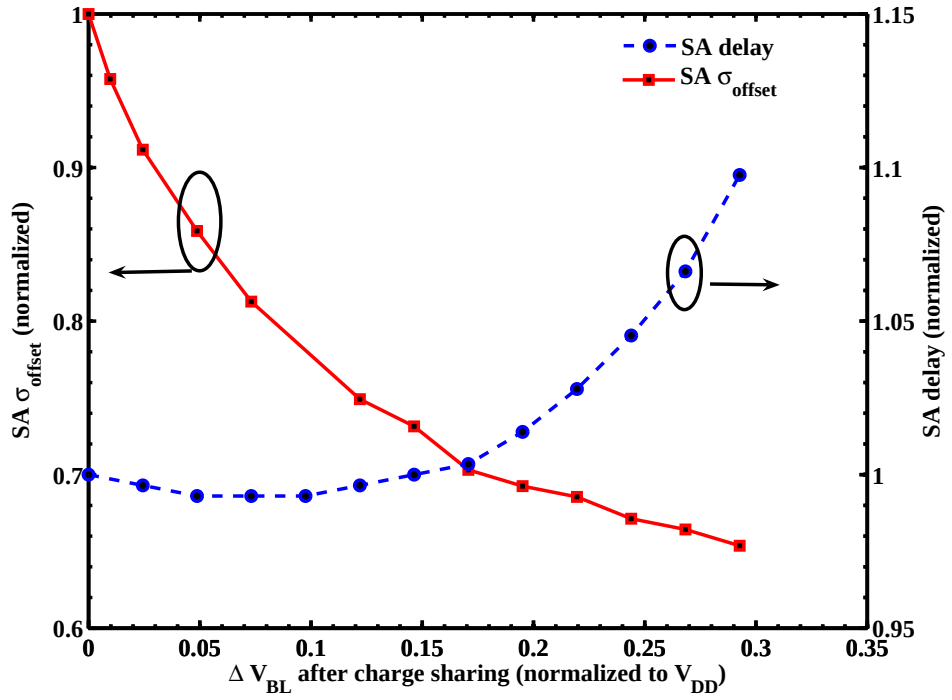


Figure 6.11: Sense amplifier delay and input offset versus ΔV_{BL} after charge sharing.

bitcell's SNM. To ensure high yield target for the embedded memories, 6σ of SNM local variation is included. As ΔV_{BL} increases, SNM increases linearly until it reaches a maximum point. Any further increase in ΔV_{BL} causes SNM to decrease significantly, which deteriorates cell read stability, which agrees with the results in [127, 131].

To evaluate the robustness of the proposed scheme in precisely controlling ΔV_{BL}, different process corners and post-layout RC extraction options were simulated to measure how ΔV_{BL} is affected. Table 6.3 shows ΔV_{BL} for different process corners, interconnect capacitances and temperature. It is clear that ΔV_{BL} shows negligible change across different conditions (9-12%), which shows the robustness of the proposed technique against PVT variations.

Table 6.3: ΔV_{BL} for different conditions.

Process	Slow	Slow	Nominal	Fast
Temp.	-40	125	25	-40
Parasitic C	max	max	nominal	min
ΔV_{BL}[A]	9.8%	9.4%	11.1%	12%

[A] Normalized to V_{DD}.

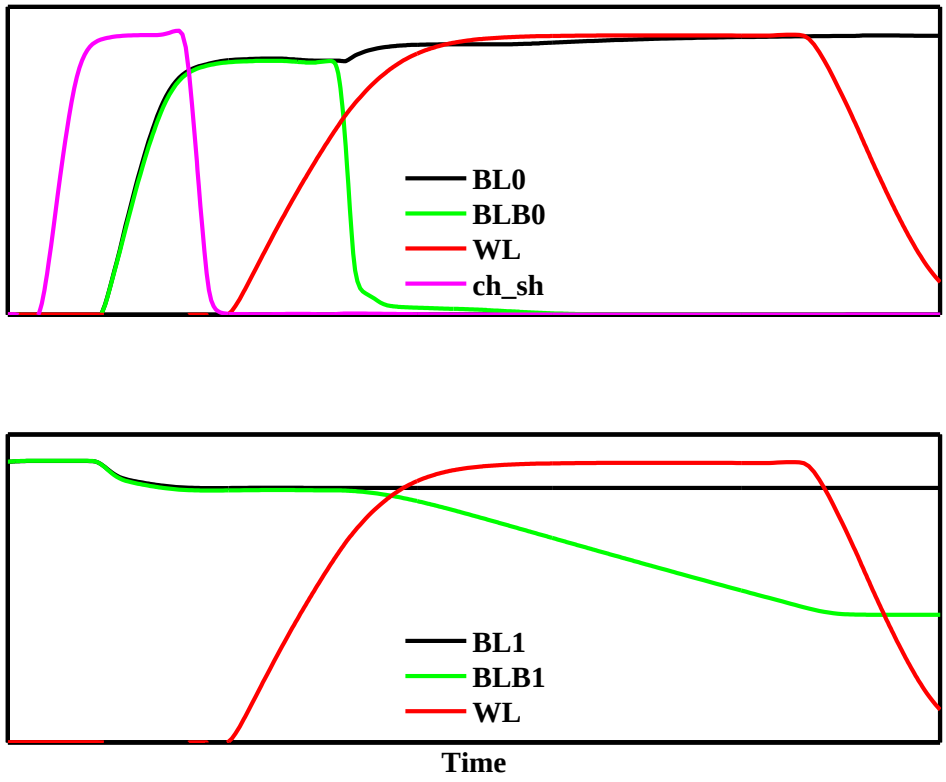


Figure 6.12: Results for selective precharge write operation. $BL0/BLB0$ are accessed for write operation while $BL1/BLB1$ are half selected bitlines.

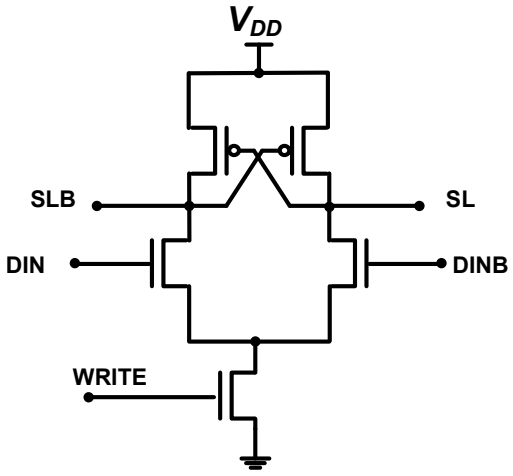


Figure 6.13: Full-swing CMOS write driver used to improve write margin.

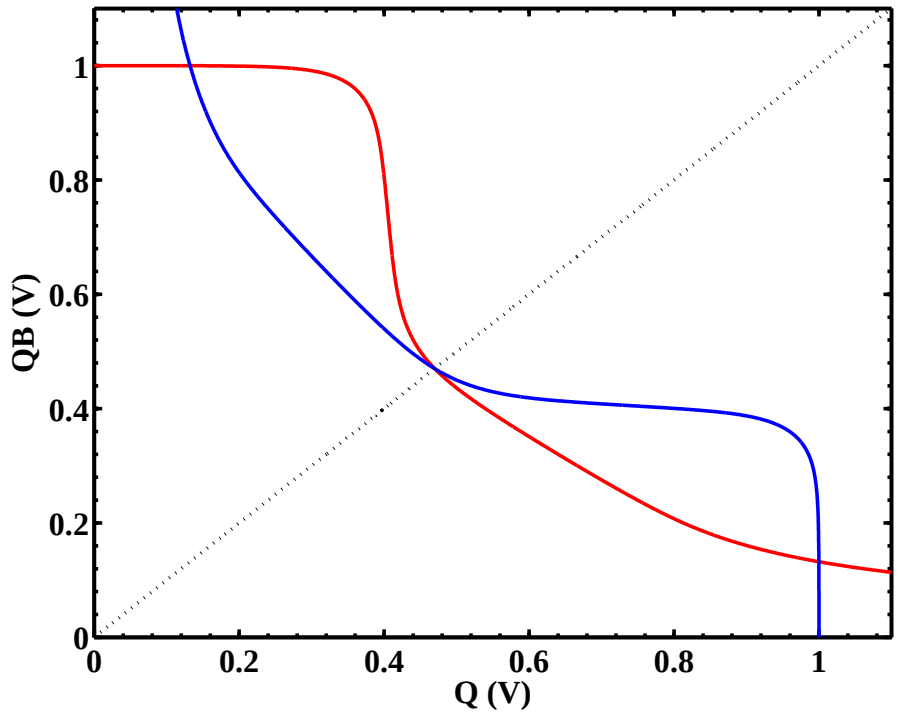


Figure 6.14: SNM simulation results for nominal devices without WID variations (voltage is normalized).

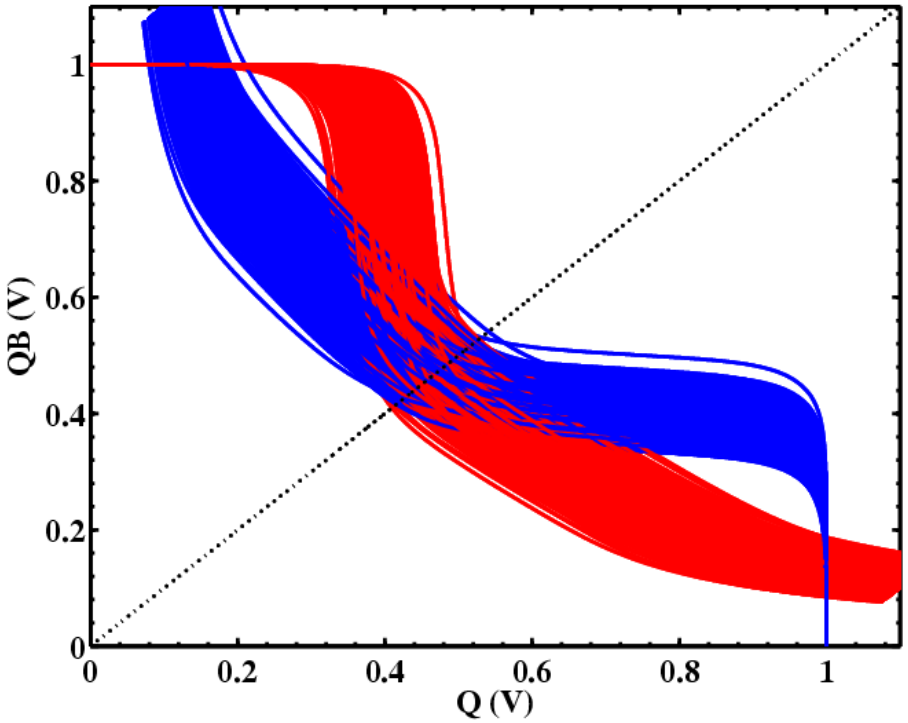


Figure 6.15: SNM simulation results using Monte Carlo simulation for 1000 MC runs (voltage is normalized).

Fig. 6.17 shows the process window curves (V_{th}), which are used to determine the operating limit of the memory with 6σ of local variation coverage. In this simulation, the D2D variations are swept for NMOS and PMOS V_{th}. For each point of D2D variations, Monte Carlo simulation using WID variations is used to find the mean and sigma for SNM. We define the failure region as the region where SNM reaches zero. Using the selective precharge technique (solid-line) the operating window is expanded as compared to the conventional approach. This increase in operation window reduces the failure probability by more than 100X.

To validate the improvements in cell stability using the proposed selective precharge technique, the designed 512kb memory was fabricated in 45nm technology. Fig. 6.19 shows the fabricated test chip micrograph. Measured results for cell failure probability are shown in Fig. 6.18 for the conventional and the proposed technique. In this test, a known data pattern is written on the memory at a high supply voltage. Following that, the supply voltage is reduced and the memory is accessed and the contents of all the bitcells are read. The number of bitcells that flipped their original state (written at high supply voltage) are recorded. These failing bitcells indicate that a read stability failure had occurred. To eliminate the impact of read access failures from distorting measurement results, a very long WL pulse is used to ensure that no failures occur in the sense amplifier as V_{DD} is reduced. These steps are repeated for different values of lower supply voltages (write at high V_{DD}, read at low V_{DD}). The same test is also executed on another memory of the same density which uses the conventional approach[3] which is considered our reference for comparison purposes. Fig. 6.18 shows that the proposed technique reduces the failure probability by more than 120X which validates the large improvement in cell read stability using our read assist technique[4].

The proposed technique has small area overhead ($< 2\%$) and shows strong robustness against process variations. In addition, it requires only one supply voltage. Hence, it does not require any additional level-shifters that cause significant area and speed penalty. Moreover, the timing generation is simple since it re-uses timing signals available in SRAM design. The memory speed also improves using the proposed technique. The large improvement in bitcell SNM and operation window show the effectiveness of this technique.

[3]Bitlines precharged to V_{DD}.

[4]The silicon data is for a limited number of parts. To generalize the conclusions and have high confidence in the measured results, it is recommended to measure large number of parts. However, this was not available when we tested this macro.

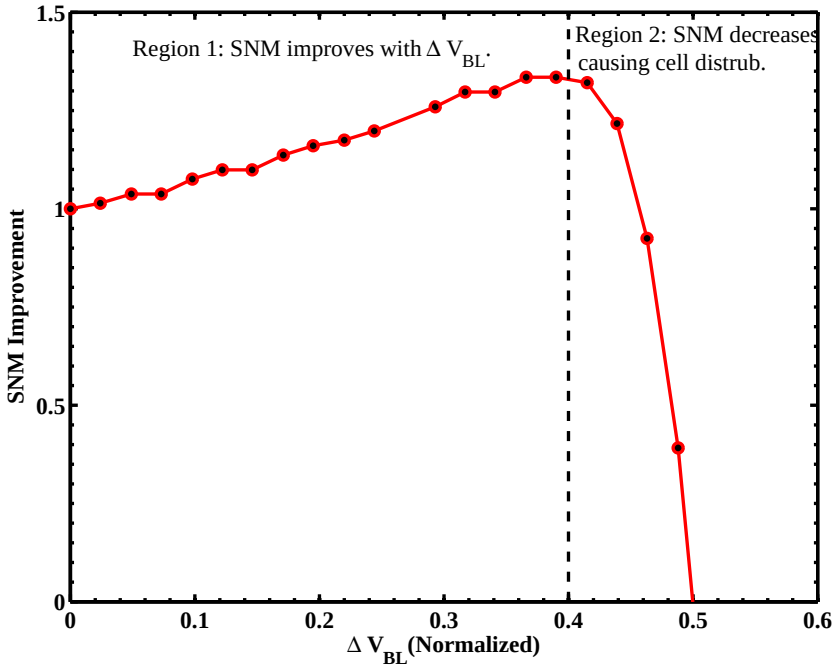


Figure 6.16: SNM improvement using the proposed technique versus ΔV_{BL} after charge sharing.

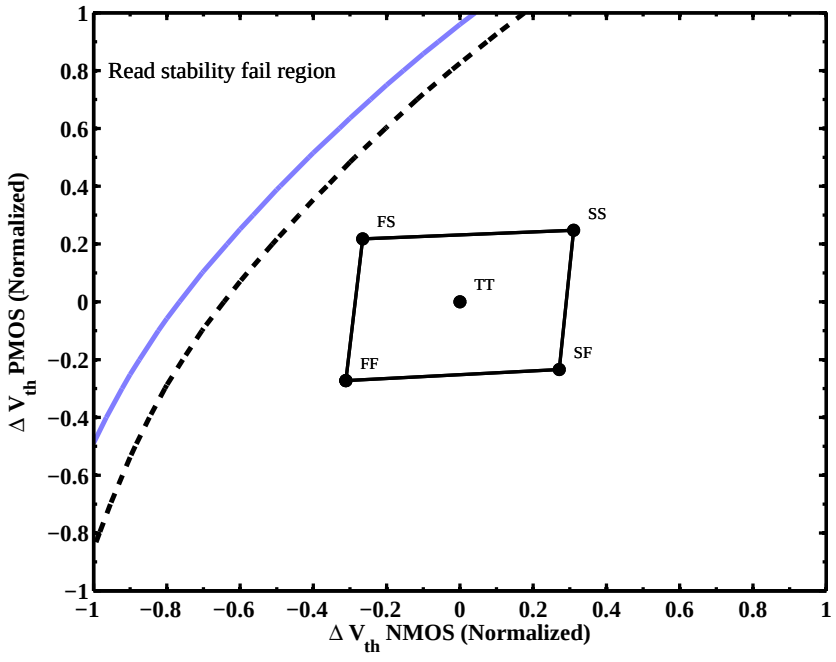


Figure 6.17: V_{th} windows showing the improvement in read stability operating window for selective precharge (solid line) compared to the conventional approach (dotted line). Simulation accounts for 6σ of local variations.

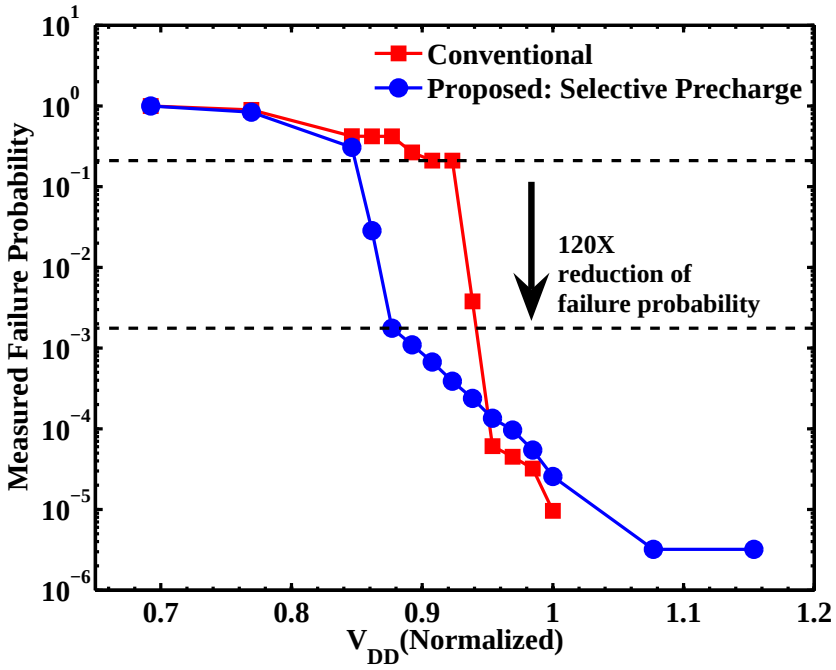


Figure 6.18: Measured failure probability for the fabricated 512kb memory using the proposed technique (selective precharge) and compared to the conventional approach.

6.6 Summary

The increase of local variations in nanometer technologies strongly affects SRAM cell stability. In this chapter, we proposed a novel read assist technique to improve SRAM static noise margin. The proposed technique, selective precharge, allows precharging different parts of the bitlines to V_{DD} and GND and uses charge sharing to precisely control the bitline voltage which increases the bitcell stability. In addition to improving SNM, the proposed technique also improves memory access time. Moreover, it only requires one supply voltage. The proposed technique has been implemented in the design of 512kb memory in 45nm technology. Results show improvements in SNM and read operation window which confirms the effectiveness of this technique. Measurements show a 120X reduction in failure probability which validates the large improvement in read stability using this technique.

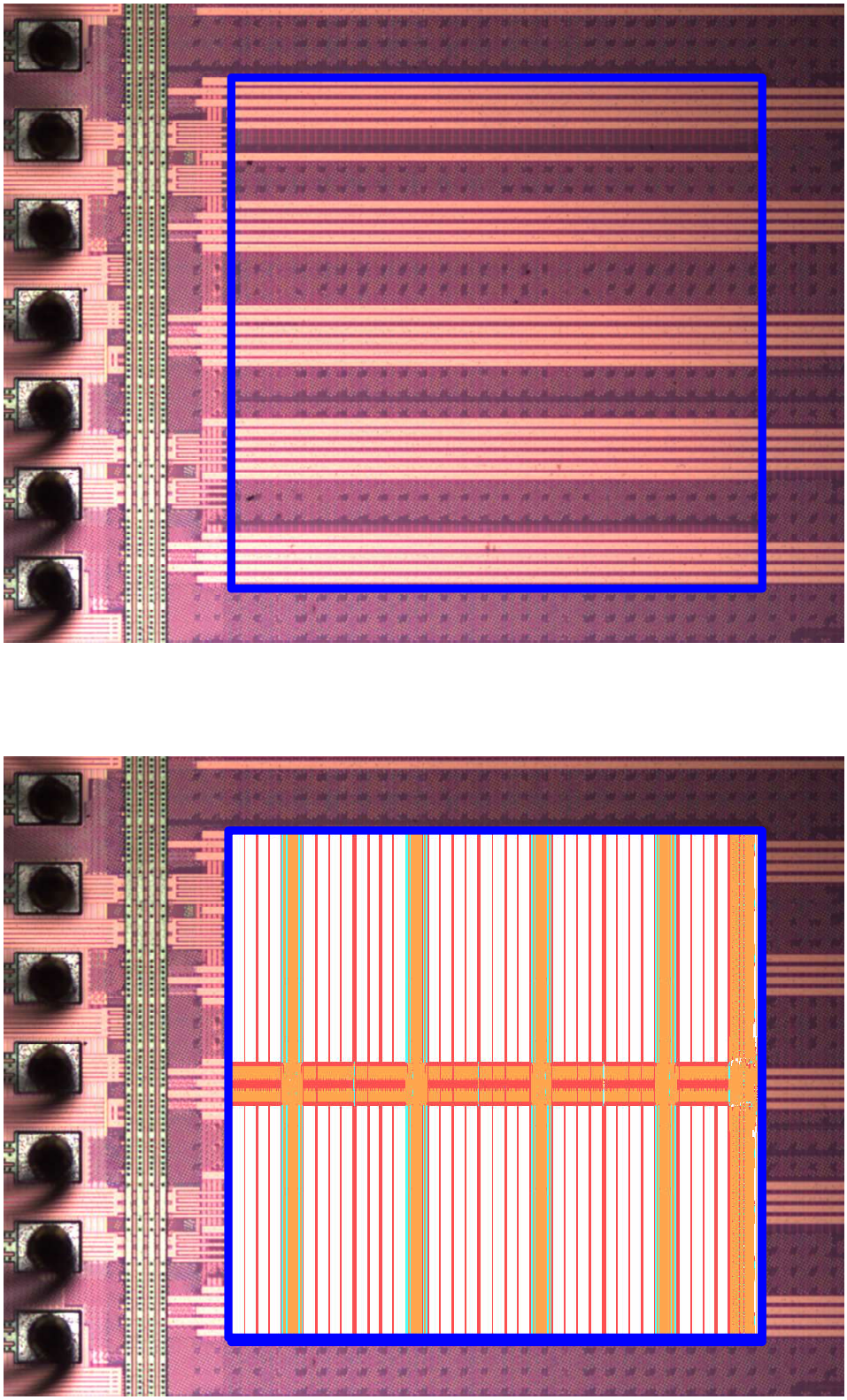


Figure 6.19: Chip micrograph for the fabricated 512kb memory in 45nm technology. Upper figure shows the location of the memory and the lower figure overlays the memory layout.

Chapter 7

Conclusions

In this chapter, we summarize our research contributions in Section 7.1 and discuss future research directions in Section 7.2.

7.1 Summary of Contributions

Aggressive technology scaling has led to numerous challenges for circuit designers. In this thesis, we studied the challenges of variation-tolerant design in digital circuits including SRAM. This research work has contributed to new techniques to address process variability in the design of nanometer circuits.

At the circuit/device levels, in Chapter 3, we presented analytical derivation for statistical delay variation model in the presence of WID statistical variations. Using the derived model, we showed that input slew has a strong effect on delay variation. It was also shown that as supply voltage is reduced there is an optimum value of input slew that achieves the minimum relative delay variation. The derived statistical models are simple which helps providing design insights on the relationship between process variations and delay variations. In addition, having simple delay models is important for early design exploration and circuit optimization. The proposed models have been validated with Monte Carlo simulation in an industrial 90nm technology

For SRAM memories, in Chapter 4, we presented a methodology for statistical estimation of read access yield. Our approach accounts for all the sources that affect SRAM read operation in one statistical flow including bitcell read current, sense amplifier offset, sensing window variations, and pass-gate leakage. We showed how the proposed methodology can be used to evaluate yield versus performance tradeoff

in the design time. In addition, this approach reduces the pessimism in using conventional worst-case analysis. Our methodology was verified using measured silicon yield results for a 1Mb memory designed in 45nm technology.

At the circuit/architecture levels, in Chapter 5, we proposed fine-grained wordline pulse width control to reduce memory power consumption in the presence of WID variations. The proposed architecture combines both architecture and circuit techniques to enable the system to detect weak bitcell, and adjust wordline pulse width accordingly. We have shown that the proposed architecture can reduce the array switching power significantly, and show large power saving especially as the chip level memory density increases. For a 48Mb memory in 45nm technology, a 27% reduction in array switching power can be achieved for a read access yield target of 95%.

At the SRAM circuit level, in Chapter 6 we proposed a new read assist technique to improve SRAM static noise margin. Our approach which is based on selective precharge, uses single supply voltage which makes it attractive for SoC implementation. We have also shown that the use of CLSA sense amplifies combined with selective precharge can reduce memory access time. The proposed technique has been implemented in the design of 512kb memory in 45nm technology. Results show improvements in SNM and read operation window which confirms the effectiveness of this technique. Test chip measurements show a 120X reduction in failure probability which validates the operation of the proposed technique.

7.2 Future Research Directions

The current technology trends show that process variations will increase further with technology scaling and more research is required in the area of variation-tolerant design. More emphasis on statistical design methods is required to enable the design of robust circuits. In addition, new variation-tolerant circuits and architectures should be investigated.

In the area of statistical delay modeling, results from our models suggest future work in utilizing the models to size and optimize logic paths to reduce delay variation. In particular, using the minimum relative delay variation point can help reduce skew in clock distribution network and reduce delay variation in the self-timed paths in memory design. In addition, the extension of the proposed models to account for different logic gates is important to estimate delay variation for any logic path.

For SRAM yield optimization, our proposed yield estimation methodology can be used to study different memory architectures, and evaluate the robustness of each architecture against process variations. This can be the basis for yield-aware memory architectures. It will also be useful to develop simple statistical models for memory yield that can be used at the architecture level. These models can be calibrated using results from our yield estimation flow.

Our proposed fine-grained wordline pulse width control can be extended to include write power consumption. This can be used to evaluate the power savings at the system level by using read and write memory activity profiles from the system information. It will also be interesting to extend this technique to account for dynamic variations such as temperature and supply voltage noise. This will require studying new digital sensors for these types of dynamic variations. This information from sensors can be directly fed into the programmable delay elements to control the wordline pulse width.

In the area of circuit techniques for SRAM stability, more work is needed to address the new dynamic concerns such as supply noise impact on SNM [133]. In addition, with the increased impact of device degradation mechanism, such as NBTI, it is important to explore design techniques that mitigate this type of variability.

Appendix A

Publications from this Work

1. **Mohamed H. Abu-Rahma** and Mohab Anis, "Variability in VLSI Circuits: Sources and Design Considerations," *ISCAS 2007: Proceedings of the IEEE International Symposium on Circuits and Systems*, 2007, **invited paper**.

2. **Mohamed H. Abu-Rahma**, Kinshuk Chowdhury, Joseph Wang, Zhiqin Chen, Sei Seung Yoon and Mohab Anis, "A Methodology for Statistical Estimation of Read Access Yield in SRAMs," *DAC '08: Proceedings of the 45th Conference on Design Automation*, 2008.

3. **Mohamed H. Abu-Rahma**, Mohab Anis and Sei Seung Yoon, "A Robust Single Supply Voltage SRAM Read Assist Technique Using Selective Precharge," *Proceedings of the 34th European Solid State Circuits Conference, ESSCIRC*, 2008.

4. **Mohamed H. Abu-Rahma** and Mohab Anis, "A Statistical Design-Oriented Delay Variation Model Accounting for Within-Die Variations," *IEEE Transactions on Computer-Aided Design of Integrated Circuits (IEEE TCAD)*, pp.1983-1995, vol. 27, November, 2008.

5. **Mohamed H. Abu-Rahma**, Mohab Anis and Sei Seung Yoon, "Reducing SRAM Power using Fine-Grained Wordline Pulse Width Control," submitted to *IEEE Transactions on VLSI Systems (IEEE TVLSI)*.

6. Mohab Anis and **Mohamed H. Abu-Rahma**, "Leakage Current Variability in Nanometer Technologies," in *IWSOC '05: Proceedings of the Fifth International Workshop on System-on-Chip for Real-Time Applications*, 2005.

Appendix B

Glossary

ATE	Automatic/automated test equipment
BL, BLB	Bitline / bitline bar (complimentary)
BISR	Built-in self-repair
BIST	Built-in self-test
BTBT	Band to band tunneling
CD	Critical dimension
CDF	Cumulative distribution function
CLSA	Current latch sense amplifer
CMOS	Complementary metal oxide semiconductor
D2D	Die-to-die variations
DIBL	Drain induced barrier lowering
ECC	Error correction codes (or circuits)
EUV	Extreme ultraviolet lithography
FBB	Forward body bias
FET	Field effect transistor
GIDL	Gate induced drain leakage
LER	Line edge roughness
MC	Monte Carlo
NBTI	Negative bias temperature instability
OPE	Optical proximity effects
PDF	Probability density function
PVT	Process-voltage-temperature
RDF	Random dopant fluctuation

RBB	Reverse body bias
RF	Register file
SA	Sense amplifier
SL, SLB	Sense line / sense line bar (complimentary)
SER	Soft error rate
SEU	Single event upset
SNM	Static-noise Margin
SoC	System-on-a-chip
SOI	Silicon-on-insulator technology
SRAM	Static random access memory
SSTA	Statistical static timing analysis
STA	Static timing analysis
VTC	Voltage transfer characteristics
VLSI	Very large scale integration
WID	Within-die variaitons
WM	Write margin or write trip point

References

[1] T.-C. Chen, "Where is CMOS going: trendy hype versus real technology," in *Proceedings of the International Solid-State Circuits Conference ISSCC*, 2006, pp. 22–28.

[2] S. R. Nassif, "Modeling and analysis of manufacturing variations," in *Proceedings of IEEE Custom Integrated Circuits conference*, 2001, pp. 223–228.

[3] H. Masuda, S. Ohkawa, A. Kurokawa, and M. Aoki, "Challenge: variability characterization and modeling for 65- to 90-nm processes," in *Proceedings of IEEE Custom Integrated Circuits conference*, 2005, pp. 593– 599.

[4] B. Wong, A. Mittal, Y. Cao, and G. W. Starr, *Nano-CMOS Circuit and Physical Design*. Wiley-Interscience, 2004.

[5] The International Technology Roadmap for Semiconductors (ITRS). [Online]. Available: http://public.itrs.net

[6] J. Tschanz, K. Bowman, and V. De, "Variation-tolerant circuits: circuit solutions and techniques," in *DAC '05: Proceedings of the 42nd annual conference on Design automation*, 2005, pp. 762–763.

[7] D. Frank, R. Dennard, E. Nowak, P. Solomon, Y. Taur, and H. S. Wong, "Device scaling limits of Si MOSFETs and their application dependencies," *Proc. IEEE*, vol. 89, no. 3, pp. 259–288, Mar 2001.

[8] J. A. Croon, W. Sansen, and H. E. Maes, *Matching Properties of Deep Sub-Micron MOS Transistors*. Springer, 2005.

[9] S. Sapatnekar, *Timing*. Springer, 2004.

[10] F. N. Najm, "On the need for statistical timing analysis," in *DAC '05: Proceedings of the 42nd annual conference on Design automation*, 2005, pp. 764–765.

[11] A. Agarwal, D. Blaauw, and V. Zolotov, "Statistical timing analysis for intra-die process variations with spatial correlations," in *ICCAD '03: Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design*, 2003, pp. 900–907.

[12] K. Agarwal and S. Nassif, "Statistical analysis of SRAM cell stability," in *DAC '06: Proceedings of the 43rd annual conference on Design automation*, 2006, pp. 57–62.

[13] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Statistical design and optimization of SRAM cell for yield enhancement," in *Proceedings of International conference on Computer Aided Design*, 2004, pp. 10–13.

[14] R. Heald and P. Wang, "Variability in sub-100nm SRAM designs," in *Proceedings of International conference on Computer Aided Design*, 2004, pp. 347–352.

[15] Y. Zorian, "Embedded memory test and repair: Infrastructure IP for SOC yield," in *Proceedings the International Test Conference (ITC)*, 2002, pp. 340–349.

[16] A. Chandrakasan, W. J. Bowhill, and F. Fox, *Design of High-Performance Microprocessor Circuits*. Wiley-IEEE Press, 2000.

[17] A. Srivastava, D. Sylvester, and D. Blaauw, *Statistical Analysis and Optimization for VLSI: Timing and Power (Series on Integrated Circuits and Systems)*. Springer, 2005.

[18] Y. Cheng and C. Hu, *MOSFET Modeling and BSIM User Guide*. Kluwer Academic Publishers, 1999.

[19] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *DAC '03: Proceedings of the 40th conference on Design automation*, 2003, pp. 338–342.

[20] T. Karnik, S. Borkar, and V. De, "Sub-90nm technologies: challenges and opportunities for CAD," in *ICCAD '02: Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design*, 2002, pp. 203–206.

[21] S. Borkar, T. Karnik, and V. De, "Design and reliability challenges in nanometer technologies," in *DAC '04: Proceedings of the 41st annual conference on Design automation*, 2004, pp. 75–75.

[22] A. Keshavarzi, G. Schrom, S. Tang, S. Ma, K. Bowman, S. Tyagi, K. Zhang, T. Linton, N. Hakim, S. Duvall, J. Brews, and V. De, "Measurements and modeling of intrinsic fluctuations in MOSFET threshold voltage," in *ISLPED '05: Proceedings of the 2005 international symposium on Low power electronics and design*, 2005, pp. 26–29.

[23] K. Bowman, S. Duvall, and J. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 2, pp. 183–190, 2002.

[24] Y. Taur and T. H. Ning, *Fundamentals of Modern VLSI Devices*. Cambridge University Press, 1998.

[25] T. Mizuno, J. Okumtura, and A. Toriumi, "Experimental study of threshold voltage fluctuation due to statistical variation of channel dopant number in MOSFET's," *IEEE Trans. Electron Devices*, vol. 41, no. 11, pp. 2216–2221, Nov 1994.

[26] K. Takeuchi, T. Tatsumi, and A. Furukawa, "Channel engineering for the reduction of random-dopant-placement-induced threshold voltage fluctuations," in *Proceedings of the International Electron Devices Meeting (IEDM)*, 1996, pp. 841–844.

[27] M. Miyamura, T. Fukai, T. Ikezawa, R. Ueno, K. Takeuchi, and M. Hane, "SRAM critical yield evaluation based on comprehensive physical / statistical modeling, considering anomalous non-gaussian intrinsic transistor fluctuations," *Proceedings of IEEE Symposium on VLSI technology*, pp. 22–23, June 2007.

[28] M. Pelgrom, A. Duinmaijer, and A. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1439, Oct 1989.

[29] B. Razavi, *Design of Analog CMOS Integrated Circuits*. McGraw-Hill, 2000.

[30] J.-T. Kong, "CAD for nanometer silicon design challenges and success," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 12, no. 11, pp. 1132–1147, 2004.

[31] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos, "Modeling within-die spatial correlation effects for process-design co-optimization," in

ISQED '05: Proceedings of the Sixth International Symposium on Quality of Electronic Design, 2005, pp. 516–521.

[32] C. Wu, Y. Leung, C. Chang, M. Tsai, H. Huang, D. Lin, Y. Sheu, C. Hsieh, W. Liang, L. Han, W. Chen, S. Chang, S. Wu, S. Lin, H. Lin, C. Wang, P. Wang, T. Lee, C. Fu, C. Chang, S. Chen, S. Jang, S. Shue, H. Lin, Y. See, Y. Mii, C. Diaz, B. Lin, M. Liang, and Y. Sun, "A 90-nm CMOS device technology with high-speed, general-purpose, and low-leakage transistors for system on chip applications," in *Proceedings of the International Electron Devices Meeting (IEDM)*, 2002, pp. 65–68.

[33] J. Croon, S. Decoutere, W. Sansen, and H. Maes, "Physical modeling and prediction of the matching properties of MOSFETs," in *Proceeding of the 34th European Solid-State Device Research conference ESSDERC*, 2004, pp. 193–196.

[34] P. Kinget, "Device mismatch and tradeoffs in the design of analog circuits," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 6, pp. 1212–1224, 2005.

[35] K. Kang, H. Kufluoglu, K. Roy, and M. Ashraful Alam, "Impact of negative-bias temperature instability in nanoscale SRAM array: Modeling and analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 10, pp. 1770–1781, Oct. 2007.

[36] K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand, "Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits," *Proceedings of the IEEE*, vol. 91, no. 2, pp. 305–327, 2003.

[37] Y.-S. Lin, C.-C. Wu, C.-S. Chang, R.-P. Yang, W.-M. Chen, J.-J. Liaw, and C. Diaz, "Leakage scaling in deep submicron CMOS for SoC," *IEEE Transactions on Electron Devices*, vol. 49, no. 6, pp. 1034–1041, 2002.

[38] J. Kao, S. Narendra, and A. Chandrakasan, "Subthreshold leakage modeling and reduction techniques," in *ICCAD '02: Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design*, 2002, pp. 141–148.

[39] S. Narendra, V. De, S. Borkar, D. Antoniadis, and A.Chandrakasan, "Full-chip subthreshold leakage power prediction and reduction techniques for sub-0.18μm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 2, pp. 501–510, 2004.

[40] J. P. de Gyvez and H. Tuinhout, "Threshold voltage mismatch and intra-die leakage current in digital CMOS circuits," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 1, pp. 157–168, 2004.

[41] S. Narendra, V. De, S. Borkar, D. Antoniadis, and A. Chandrakasan, "Full-chip sub-threshold leakage power prediction model for sub-0.18μm CMOS," in *ISLPED '02: Proceedings of the 2002 international symposium on Low power electronics and design*, 2002, pp. 19–23.

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

[43] R. Rao, A. Srivastava, D. Blaauw, and D. Sylvester, "Statistical analysis of subthreshold leakage current for VLSI circuits," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 12, no. 2, pp. 131–139, 2004.

[44] S. H. Choi, B. C. Paul, and K. Roy, "Novel sizing algorithm for yield improvement under process variation in nanometer technology," in *DAC '04: Proceedings of the 41st annual conference on Design automation*, 2004, pp. 454–459.

[45] A. Agarwal, K. Chopra, and D. Blaauw, "Statistical timing based optimization using gate sizing," in *DATE '05: Proceedings of the conference on Design, Automation and Test in Europe*, 2005, pp. 400–405.

[46] M. Miyazaki, G. Ono, T. Hattori, K. Shiozawa, K. Uchiyama, and K. Ishibashi, "A 1000-MIPS/w microprocessor using speed adaptive threshold-voltage CMOS with forward bias," in *Proceedings of the International Solid-State Circuits Conference ISSCC*, 2000, pp. 420–421.

[47] M. Miyazaki, G. Ono, and K. Ishibashi, "A 1.2-GIPS/w microprocessor using speed-adaptive threshold-voltage CMOS with forward bias," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 2, pp. 210–217, 2002.

[48] J. Tschanz, J. Kao, S. Narendra, R. Nair, D. Antoniadis, A. Chandrakasan, and V. De, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp. 1396–1402, 2002.

[49] J. Tschanz, S. Narendra, R. Nair, and V. De, "Effectiveness of adaptive supply voltage and body bias for reducing impact of parameter variations in

low power and high performance microprocessors," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 5, pp. 826–829, 2003.

[50] Y. Komatsu, K. Ishibashi, M. Yamamoto, T. Tsukada, K. Shimazaki, M. Fukazawa, and M. Nagata, "Substrate-noise and random-fluctuations reduction with self-adjusted forward body bias," in *Proceedings of IEEE Custom Integrated Circuits conference*, 2005, pp. 35–38.

[51] C. Kim, K. Roy, S. Hsu, A. Alvandpour, R. Krishnamurthy, and S. Borkar, "A process variation compensating technique for sub-90 nm dynamic circuits," in *Proceedings of IEEE Symposium on VLSI Circuits*, 2003, pp. 205–206.

[52] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits (2nd Edition)*. Prentice Hall, 2002.

[53] K. Bowman, S. Duvall, and J. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution," in *Proceedings of the International Solid-State Circuits Conference ISSCC*, 2001, pp. 278–279.

[54] K. Bowman and J. Meindl, "Impact of within-die parameter fluctuations on future maximum clock frequency distributions," in *Proceedings of IEEE Custom Integrated Circuits conference*, 2001, pp. 229–232.

[55] D. Marculescu and E. Talpes, "Variability and energy awareness: a microarchitecture-level perspective," in *DAC '05: Proceedings of the 42nd annual conference on Design automation*, 2005, pp. 11–16.

[56] ——, "Energy awareness and uncertainty in microarchitecture-level design," *IEEE Micro*, vol. 25, no. 5, pp. 64–76, 2005.

[57] A. Datta, S. Bhunia, S. Mukhopadhyay, N. Banerjee, and K. Roy, "Statistical modeling of pipeline delay and design of pipeline under process variation to enhance yield in sub-100nm technologies," in *DATE '05: Proceedings of the conference on Design, Automation and Test in Europe*, 2005, pp. 926–931.

[58] N. Azizi, M. M. Khellah, V. De, and F. N. Najm, "Variations-aware low-power design with voltage scaling," in *DAC '05: Proceedings of the 42nd annual conference on Design automation*, 2005, pp. 529–534.

[59] E. Humenay, D. Tarjan, and K. Skadron., "Impact of parameter variations on multi-core chips," in *In Proceedings of the 2006 Workshop on Architectural*

Support for Gigascale Integration, in conjunction with the 33rd International Symposium on Computer Architecture (ISCA), June 2006.

[60] E. Humenay, W. Huang, M. R. Stan, and K. Skadron, "Toward an architectural treatment of parameter variations," Univ. of Virginia Dept. of Computer Science, Tech. Rep. CS-2005-16, Sept. 2005.

[61] H. Pilo, "IEDM SRAM short course," 2006.

[62] H. Yamauchi, "Embedded SRAM circuit design technologies for a 45nm and beyond," *ASICON '07. 7th International Conference on ASIC*, pp. 1028–1033, 22-25 Oct. 2007.

[63] A. Bhavnagarwala, S. Kosonocky, C. Radens, K. Stawiasz, R. Mann, Q. Ye, and K. Chin, "Fluctuation limits and scaling opportunities for CMOS SRAM cells," *Proceedings of the International Electron Devices Meeting (IEDM)*, pp. 659–662, 2005.

[64] H. Pilo, C. Barwin, G. Braceras, C. Browning, S. Lamphier, and F. Towler, "An SRAM design in 65-nm technology node featuring read and write-assist circuits to expand operating voltage," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 4, pp. 813–819, April 2007.

[65] M. Yamaoka, N. Maeda, Y. Shinozaki, Y. Shimazaki, K. Nii, S. Shimada, K. Yanagisawa, and T. Kawahara, "90-nm process-variation adaptive embedded SRAM modules with power-line-floating write technique," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 3, pp. 705–711, March 2006.

[66] R. Morimoto, T. Kimura, T. Hirai, and T. Hoshino, "Layout-design methodology of 0.246-um2-embedded 6T-SRAM for 45-nm high-performance system LSIs," *Proceedings of IEEE Symposium on VLSI technology*, pp. 28–29, June 2007.

[67] N. H. Weste and D. Harris, *CMOS VLSI Design : A Circuits and Systems Perspective (3rd Edition)*. Addison Wesley, 2004.

[68] M. Yamaoka, N. Maeda, Y. Shinozaki, Y. Shimazaki, K. Nii, S. Shimada, K. Yanagisawa, and T. Kawahara, "Low-power embedded SRAM modules with expanded margins for writing," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 480–611 Vol. 1, 2005.

[69] M. Yabuuchi, K. Nii, Y. Tsukamoto, S. Ohbayashi, S. Imaoka, H. Makino, Y. Yamagami, S. lshikura, T. Terano, T. Oashi, K. Hashimoto, A. Sebe, G. Okazaki, K. Satomi, H. Akamatsu, and H. Shinohara, "A 45nm low-standby-power embedded SRAM with improved immunity against process and temperature variations," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 326–606, 11-15 Feb. 2007.

[70] H. Qin, Y. Cao, D. Markovic, A. Vladimirescu, and J. Rabaey, "SRAM leakage suppression by minimizing standby supply voltage," *Proceedings of the International Symposium on Quality of Electronic Design ISQED*, pp. 55–60, 2004.

[71] J. Lin, A. Oates, H. Tseng, Y. Liao, T. Chung, K. Huang, P. Tong, S. Yau, and Y. Wang, "Prediction and control of NBTI induced SRAM Vccmin drift," *Proceedings of the International Electron Devices Meeting (IEDM)*, 2006.

[72] J. Wang, A. Singhee, R. Rutenbar, and B. Calhoun, "Statistical modeling for the minimum standby supply voltage of a full SRAM array," *33rd European Solid State Circuits Conference, 2007. ESSCIRC*, pp. 400–403, Sept. 2007.

[73] M. Yamaoka and T. Kawahara, "Operating-margin-improved SRAM with column-at-a-time body-bias control technique," *33rd European Solid State Circuits Conference, 2007. ESSCIRC*, pp. 396–399, 11-13 Sept. 2007.

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

[75] K. Nii, M. Yabuuchi, Y. Tsukamoto, S. Ohbayashi, S. Imaoka, H. Makino, Y. Yamagami, S. Ishikura, T. Terano, T. Oashi, K. Hashimoto, A. Sebe, S. Okazaki, K. Satomi, H. Akamatsu, and H. Shinohara, "A 45-nm bulk CMOS embedded SRAM with improved immunity against process and temperature variations," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 180–191, Jan. 2008.

[76] A. Krishnan, V. Reddy, D. Aldrich, J. Raval, K. Christensen, J. Rosal, C. O'Brien, R. Khamankar, A. Marshall, W.-K. Loh, R. McKee, and S. Krishnan, "SRAM cell static noise margin and VMIN sensitivity to transistor degradation," *Proceedings of the International Electron Devices Meeting (IEDM)*, 2006.

[77] K. Kang, H. Kufluoglu, K. Roy, and M. Ashraful Alam, "Impact of negative-bias temperature instability in nanoscale SRAM array: Modeling and analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 10, pp. 1770–1781, Oct. 2007.

[78] K. Takeda, H. Ikeda, Y. Hagihara, M. Nomura, and H. Kobatake, "Redefinition of write margin for next-generation SRAM and write-margin monitoring circuit," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 2602–2611, 2006.

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

[80] ——, "Soft errors in advanced computer systems," *Design and Test of Computers, IEEE*, vol. 22, no. 3, pp. 258–266, May-June 2005.

[81] E. Cannon, D. Reinhardt, M. Gordon, and P. Makowenskyj, "SRAM SER in 90, 130 and 180 nm bulk and SOI technologies," *Proceedings of the 42nd IEEE International Reliability Physics Symposium*, pp. 300–304, 2004.

[82] A. Balasubramanian, P. Fleming, B. Bhuva, A. Sternberg, and L. Massengill, "Implications of dopant-fluctuation-induced v_t variations on the radiation hardness of deep submicrometer CMOS SRAMs," *IEEE Transactions on Device and Materials Reliability*, vol. 8, no. 1, pp. 135–144, March 2008.

[83] S. S. Mukherjee, J. Emer, and S. K. Reinhardt, "The soft error problem: An architectural perspective," in *HPCA '05: Proceedings of the 11th International Symposium on High-Performance Computer Architecture*, 2005, pp. 243–247.

[84] M. Eisele, J. Berthold, D. Schmitt-Landsiedel, and R. Mahnkopf, "The impact of intra-die device parameter variations on path delays and on the design for yield of low voltage digital circuits," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 5, no. 4, pp. 360–368, 1997.

[85] T. Sakurai and A. Newton, "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 2, pp. 584–594, 1990.

[86] H. Mahmoodi, S. Mukhopadhyay, and K. Roy, "Estimation of delay variations due to random-dopant fluctuations in nanoscale CMOS circuits," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 9, pp. 1787–1796, 2005.

[87] Y. Cao and L. T. Clark, "Mapping statistical process variations toward circuit performance variability: an analytical modeling approach," in *DAC '05: Proceedings of the 42nd annual conference on Design automation*, 2005, pp. 658–663.

[88] H. Masuda, S. Okawa, and M. Aoki, "Approach for physical design in sub-100 nm era," in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, 2005, pp. 5934–5937.

[89] K. Shinkai, M. Hashimoto, A. Kurokawa, and T. Onoye, "A gate delay model focusing on current fluctuation over wide-range of process and environmental variability," in *ICCAD '06: Proceedings of the 2006 IEEE/ACM international conference on Computer-aided design*, 2006, pp. 47–53.

[90] W. Liu, *MOSFET Modeling for SPICE Simulation, Including BSIM3v3 and BSIM4*. John Wiley and Sons, 2001.

[91] N. Hedenstierna and K. Jeppson, "CMOS circuit speed and buffer optimization," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 6, no. 3, pp. 270–281, March 1987.

[92] J. M. Daga and D. Auvergne, "A comprehensive delay macro modeling for submicrometer CMOS logics," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 1, pp. 42–55, Jan. 1999.

[93] M. Pelgrom, H. Tuinhout, and M. Vertregt, "Transistor matching in analog CMOS applications," in *Proceedings of the International Electron Devices Meeting (IEDM)*, 1998, pp. 915–918.

[94] C. Piguet, *Low-Power CMOS Circuits: Technology, Logic Design and CAD Tools*. CRC, 2005.

[95] V. Nguyen Hoang, A. Kumar, and P. Christie, "The impact of back-end-of-line process variations on critical path timing," *International Interconnect Technology Conference (IITC)*, pp. 193–195, 2006.

[96] A. Asenov, A. Brown, J. Davies, S. Kaya, and G. Slavcheva, "Simulation of intrinsic parameter fluctuations in decananometer and nanometer-scale MOSFETs," *IEEE Trans. Electron Devices*, vol. 50, no. 9, pp. 1837– 1852, Sept 2003.

[97] B. Amrutur and M. Horowitz, "Speed and power scaling of SRAM's," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 2, pp. 175–185, Feb 2000.

[98] Y. Wang, H. J. Ahn, U. Bhattacharya, Z. Chen, T. Coan, F. Hamzaoglu, W. Hafez, C.-H. Jan, P. Kolar, S. Kulkarni, J.-F. Lin, Y.-G. Ng, I. Post, L. Wei, Y. Zhang, K. Zhang, and M. Bohr, "A 1.1 GHz 12 A/Mb-Leakage SRAM design in 65 nm ultra-low-power CMOS technology with integrated leakage reduction for mobile applications," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 172–179, Jan. 2008.

[99] B. Wicht, T. Nirschl, and D. Schmitt-Landsiedel, "Yield and speed optimization of a latch-type voltage sense amplifier," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 7, pp. 1148–1158, July 2004.

[100] T. Matthews and P. Heedley, "A simulation method for accurately determining dc and dynamic offsets in comparators," *Circuits and Systems, 2005. 48th Midwest Symposium on*, pp. 1815–1818 Vol. 2, Aug. 2005.

[101] S. Mukhopadhyay, K. Kim, K. Jenkins, C.-T. Chuang, and K. Roy, "Statistical characterization and on-chip measurement methods for local random variability of a process using sense-amplifier-based test structure," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 400–611, Feb. 2007.

[102] B. Amrutur and M. Horowitz, "A replica technique for wordline and sense control in low-power SRAM's," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 8, pp. 1208–1219, Aug 1998.

[103] E. Morifuji, D. Patil, M. Horowitz, and Y. Nishi, "Power optimization for SRAM and its scaling," *IEEE Transactions on Electron Devices*, vol. 54, no. 4, pp. 715–722, April 2007.

[104] C. Pacha, B. Martin, K. von Arnim, R. Brederlow, D. Schmitt-Landsiedel, P. Seegebrecht, J. Berthold, and R. Thewes, "Impact of STI-induced stress, inverse narrow width effect, and statistical v_{TH} variations on leakage currents in 120 nm CMOS," *Proceeding of the 34th European Solid-State Device Research conference ESSDERC*, pp. 397–400, 2004.

[105] Y. Zorian and S. Shoukourian, "Embedded-memory test and repair: Infrastructure IP for SoC yield," *IEEE Design and Test of Computers*, vol. 20, no. 3, pp. 58–66, 2003.

[106] R. Venkatraman, R. Castagnetti, and S. Ramesh, "The statistics of device variations and its impact on SRAM bitcell performance, leakage and stability," in *Proceedings of the International Symposium on Quality of Electronic Design ISQED*, 2006, pp. 190–195.

[107] M. Mamidipaka, K. Khouri, N. Dutt, and M. Abadir, "Idap: a tool for high-level power estimation of custom array structures," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 23, no. 9, pp. 1361–1369, Sept. 2004.

[108] M. Q. Do, M. Drazdziulis, P. Larsson-Edefors, and L. Bengtsson, "Parameterizable architecture-level SRAM power model using circuit-simulation backend for leakage calibration," in *Proceedings of the International Symposium on Quality of Electronic Design ISQED*, 2006, pp. 557–563.

[109] A. Macii, L. Benini, and M. Poncino, *Memory Design Techniques for Low Energy Embedded Systems*. Kluwer Academic Pub, 2002.

[110] K. Osada, J.-U. Shin, M. Khan, Y.-D. Liou, K. Wang, K. Shoji, K. Kuroda, S. Ikeda, and K. Ishibashi, "Universal-Vdd 0.65-2.0V 32 kB cache using voltage-adapted timing-generation scheme and a lithographical-symmetric cell," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 168–169, 443, 2001.

[111] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. McGraw-Hill, 1991.

[112] W. Kever, S. Ziai, M. Hill, D. Weiss, and B. Stackhouse, "A 200 MHz RISC microprocessor with 128 kB on-chip caches," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 410–411, 495, 6-8 Feb 1997.

[113] Y. H. Chan, T. J. Charest, J. R. Rawlins, A. D. Tuminaro, J. K. Wadhwa, and O. M. Wagner, "Programmable sense amplifier timing generator," United States Patent 6,958,943, October 2005.

[114] M. Min, P. Maurine, M. Bastian, and M. Robert, "A novel dummy bitline driver for read margin improvement in an eSRAM," *Proceedings of the*

IEEE International Symposium on Electronic Design, Test and Applications DELTA, pp. 107–110, Jan. 2008.

[115] R. Rajsuman, "Design and test of large embedded memories: An overview," *IEEE Design and Test of Computers*, vol. 18, no. 3, pp. 16–27, 2001.

[116] K. Itoh, M. Horiguchi, and M. Yamaoka, "Low-voltage limitations of memory-rich nano-scale CMOS LSIs," *33rd European Solid State Circuits Conference, 2007. ESSCIRC*, pp. 68–75, 11-13 Sept. 2007.

[117] A. Bhavnagarwala, S. Kosonocky, Y. Chan, K. Stawiasz, U. Srinivasan, S. Kowalczyk, and M. Ziegler, "A sub-600mv, fluctuation tolerant 65nm CMOS SRAM array with dynamic cell biasing," *Proceedings of IEEE Symposium on VLSI Circuits*, pp. 78–79, 2007.

[118] E. Grossar, M. Stucchi, K. Maex, and W. Dehaene, "Read stability and write-ability analysis of SRAM cells for nanometer technologies," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 11, pp. 2577–2588, Nov. 2006.

[119] Y. Morita, H. Fujiwara, H. Noguchi, Y. Iguchi, K. Nii, H. Kawaguchi, and M. Yoshimoto, "An area-conscious low-voltage-oriented 8T-SRAM design under DVS environment," *Proceedings of IEEE Symposium on VLSI Circuits*, pp. 256–257, June 2007.

[120] L. Chang, Y. Nakamura, R. Montoye, J. Sawada, A. Martin, K. Kinoshita, F. Gebara, K. Agarwal, D. Acharyya, W. Haensch, K. Hosokawa, and D. Jamsek, "A 5.3GHz 8T-SRAM with operation down to 0.41v in 65nm CMOS," *Proceedings of IEEE Symposium on VLSI Circuits*, pp. 252–253, June 2007.

[121] R. Joshi, R. Houle, K. Batson, D. Rodko, P. Patel, W. Huott, R. Franch, Y. Chan, D. Plass, S. Wilson, and P. Wang, "6.6+ GHz low Vmin, read and half select disturb-free 1.2 Mb SRAM," *Proceedings of IEEE Symposium on VLSI Circuits*, pp. 250–251, June 2007.

[122] M. Khellah, D. Somasekhar, Y. Ye, N. S. Kim, J. Howard, G. Ruhl, M. Sunna, J. Tschanz, N. Borkar, F. Hamzaoglu, G. Pandya, A. Farhang, K. Zhang, and V. De, "A 256-kb dual-V_{CC} SRAM building block in 65-nm CMOS process with actively clamped sleep transistor," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 1, pp. 233–242, Jan. 2007.

[123] J. Pille, C. Adams, T. Christensen, S. Cottier, S. Ehrenreich, T. Kono, D. Nelson, O. Takahashi, S. Tokito, O. Torreiter, O. Wagner, and D. Wendel, "Implementation of the cell broadband engine in a 65nm SOI technology featuring dual-supply SRAM arrays supporting 6GHz at 1.3V," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 322–606, 11-15 Feb. 2007.

[124] K. Zhang, U. Bhattacharya, Z. Chen, F. Hamzaoglu, D. Murray, N. Vallepalli, Y. Wang, B. Zheng, and M. Bohr, "A 3-GHz 70MB SRAM in 65nm CMOS technology with integrated column-based dynamic power supply," *Proceedings of the International Solid-State Circuits Conference ISSCC*, pp. 474–611 Vol. 1, 10-10 Feb. 2005.

[125] S. Ohbayashi, M. Yabuuchi, K. Nii, Y. Tsukamoto, S. Imaoka, Y. Oda, T. Yoshihara, M. Igarashi, M. Takeuchi, H. Kawashima, Y. Yamaguchi, K. Tsukamoto, M. Inuishi, H. Makino, K. Ishibashi, and H. Shinohara, "A 65-nm SoC embedded 6T-SRAM designed for manufacturability with read and write operation stabilizing circuits," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 4, pp. 820–829, April 2007.

[126] K. Zhang, U. Bhattacharya, Z. Chen, F. Hamzaoglu, D. Murray, N. Vallepalli, Y. Wang, B. Zheng, and M. Bohr, "A 3-GHz 70-Mb SRAM in 65-nm cmos technology with integrated column-based dynamic power supply," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 1, pp. 146–151, Jan. 2006.

[127] B. Campbell, J. Burnette, N. Javarappa, and V. von Kaenel, "Power-efficient dual-supply 64kB L1 caches in a 65nm CMOS technology," *Proceedings of IEEE Custom Integrated Circuits conference*, pp. 729–732, 2007.

[128] Y. Morita, H. Fujiwara, H. Noguchi, K. Kawakami, J. Miyakoshi, S. Mikami, K. Nii, H. Kawaguchi, and M. Yoshimoto, "A Vth-Variation-Tolerant SRAM with 0.3-V minimum operation voltage for memory-rich SoC under DVS environment," *Proceedings of IEEE Symposium on VLSI Circuits*, pp. 13–14, 2006.

[129] Y. Hirano, M. Tsujiuchi, K. Ishikawa, H. Shinohara, T. Terada, Y. Maki, T. Iwamatsu, K. Eikyu, T. Uchida, S. Obayashi, K. Nii, Y. Tsukamoto, M. Yabuuchi, T. Ipposhi, H. Oda, and Y. Inoue, "A robust SOI SRAM architecture by using advanced ABC technology for 32nm node and beyond LSTP

devices," *Proceedings of IEEE Symposium on VLSI technology*, pp. 78–79, June 2007.

[130] T. Suzuki, H. Yamauchi, K. Satomi, and H. Akamatsu, "A stable SRAM mitigating cell-margin asymmetricity with a disturb-free biasing scheme," *Proceedings of IEEE Custom Integrated Circuits conference*, pp. 233–236, 2007.

[131] M. Khellah, Y. Ye, N. Kim, D. Somasekhar, G. Pandya, A. Farhang, K. Zhang, C. Webb, and V. De, "Wordline and bitline pulsing schemes for improving SRAM cell stability in low Vcc 65nm CMOS designs," *Proceedings of IEEE Symposium on VLSI Circuits*, pp. 9–10, 2006.

[132] M. H. Abu-Rahma, M. Anis, and S. S. Yoon, "A robust single supply voltage SRAM read assist technique using selective precharge," in *Proceedings of the 34th European Solid State Circuits Conference ESSCIRC*, 2008, pp. 234–237.

[133] M. Khellah, D. Khalil, D. Somasekhar, Y. Ismail, T. Karnik, and V. De, "Effect of power supply noise on SRAM dynamic stability," *Proceedings of IEEE Symposium on VLSI Circuits*, pp. 76–77, June 2007.