VARIABILITY AWARE ANALYSIS AND OPTIMIZATION OF VLSI CIRCUITS

by

Vivek Joshi

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering)
in the University of Michigan
2011

Doctoral Committee:

Professor Dennis M. Sylvester, Chair
Professor David Blaauw
Professor Marios C. Papaefthymiou
Assistant Professor Akram Boukai

DEDICATION

To my family & friends…

This dissertation is dedicated to my family and friends that supported, encouraged, and inspired me throughout my education. I'd especially like to thank my parents, Dinesh Chandra Joshi and Asha Joshi, and my sister Deepti. I love you all.

ACKNOWLEDGEMENT

I would like to offer my most sincere gratitude to a number of people. I am deeply grateful to my advisor, Prof. Dennis Sylvester, for his guidance and support throughout these years. I would like to thank Prof. David Blaauw, and Dr. Kanak Agarwal for collaborating on most of my projects, and providing valuable guidance and suggestions. I would also like to acknowledge Dr. Andres Torres, and Dr. Valeriy Sukharev for their collaboration on Chapter 5. I would like to thank my colleague Brian Cline for his collaboration on Chapter 4. Finally, I would also like to thank all of my colleagues that worked under Prof. Dennis Sylvester and Prof. David Blaauw during my years at the University of Michigan.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Continued scaling of semiconductor technology has greatly increased the complexity of the manufacturing process, and Design for Manufacturing (DFM) has emerged as an important topic of research over the last decade. DFM strives to reduce variability in Integrated Circuits (ICs) through extensive modeling and analysis of process induced variability, to enable yield-optimal design of semiconductor devices, libraries, and circuits. This dissertation focuses on modeling, analysis and optimization techniques to manage variability within IC design. Parameter variations cause high yield loss due their strong impact on circuit delay. This dissertation begins with proposing the use of so called soft-edge flip-flops with a small window of transparency, instead of a hard edge for capturing data. Soft edge flip-flops allow time borrowing and averaging across stages, making the design less sensitive to process variations, and experimental results show that as compared to clustering based skew assignment technique, our approach provides improvements of up to 22.6% (8.9% on average) in the mean and up to 24.1% (10% on average) in the standard deviation of the delay, with a very small power overhead (less than 3%). Next four chapters model the layout dependence of mechanical stress and explore techniques to exploit the layout dependencies of mechanically stressed silicon through mechanical stress aware design and optimization. Chapter 3 uses mechanical stress aware standard cell library design in conjunction with dual threshold voltage (V_{th}) assignment to achieve optimal power-performance tradeoff, and decrease leakage power consumption by ~24%. Chapter 4 discusses a standard cell

library design technique called STEEL, which provides average delay improvements of 11% over equivalent single-V_{th} implementations, while consuming 2.5X less leakage than the dual-V_{th} alternative. The next two chapters focus on modeling the layout dependence of mechanical stress. Chapter 5 discusses compact closed-form models for layout dependence of process induced stress, and its impact on carrier mobility. Experimental results based on simulation and measured data show that the proposed models accurately capture the layout dependence of mechanical stress. Chapter 6 proposes a technique to model non-rectangular gates (NRG) with non-uniform carrier mobility to enable accurate prediction of both device drive current and leakage. Next chapter studies the impact of Rapid Thermal Anneal (RTA) temperature variation on circuit timing and leakage, by proposing a new local anneal temperature variation aware analysis framework, and proposes techniques to minimize the impact of anneal temperature variation. Chapters 8 and 9 show significant impact of different Double Patterning Lithography (DPL) techniques on Static random-access memory (SRAM) robustness through measurement and simulation, and propose DPL-aware sizing optimization of SRAM cell. Experimental results based on 45nm industrial models show that using the best DPL option for each layer, along with the sizing optimization presented, can achieve single exposure robustness together with improved DPL printability at nearly no overhead (less than 0.2% increase in write energy). Finally, a framework that captures through-silicon via (TSV) induced mechanical stress and its impact on device mobility is discussed. It is used to study the impact of TSV stress on circuit delay, and TSV stress is shown to cause delay variations of up to 6.9%.

Chapter 1
Introduction

Extending Moore's law through aggressive process scaling has been the driving force behind semiconductor industry. However, scaling of semiconductor technologies to sub-45nm nodes has greatly expanded the number and complexity of sources of variation. Variations due to thermal, lithographic, mechanical stress (layout dependent), and doping sources need to be modeled to enable accurate analysis and optimization of very-large-scale-integration (VLSI) circuits [1, 2]. Since the underlying variation causing mechanism is different for each of these sources, separate modeling and analysis is required to consider these more complex aspects of variation outside the normal corner considerations of process, voltage, and temperature [3]. At the device level, these sources of variation affect device properties such as gate length (L), threshold voltage (V_{th}), mobility (μ), oxide thickness (t_{ox}), etc.

1.1 Variation in Device Parameters

This section discusses important sources of variation in advanced semiconductor technology nodes.

1.1.1 Gate Length Variation

In nanometer CMOS optical lithography is being pushed to new extremes. The smallest printable feature size is defined by the Raleigh criterion to be $k_1\lambda/NA$ [4], where k_1 is the process difficulty factor, λ is the wavelength of the light source, and NA is the

numerical aperture determined by lens size. Currently, 193nm is the shortest wavelength in use for semiconductor production and is expected to continue its dominance for several technology nodes in the future. At 65nm technology node and below, the minimum feature size is much smaller than the optical wavelength, thereby causing the printed shapes to deviate significantly from the drawn rectilinear shapes [5, 6]. Since gate length (L) is typically the critical dimension (CD), it is extremely susceptible to variation. Variation in L has a strong impact on the performance and leakage of a circuit, by altering the drain current (I_D), V_{th} through drain induced barrier lowering (DIBL), and gate-to-channel capacitance (C_{gc}), which loads the previous logic stage. Several prior works have proposed approaches to characterizing, modeling, analyzing, and managing/avoiding CD variation [7, 8, 9, 10, 11]. Non-uniform gates (NRG) are typically modeled by breaking them up into a set of parallel transistors with constant gate lengths. By summing up the current for each slice, one can obtain drive current for the transistor, and this can be mapped to one value of representative gate length for the device based on a current versus gate length look-up table.

Today's most aggressive single exposure production processes with off axis illumination have a k_1 factor of 0.36 - 0.4 for logic, and 0.29-0.30 for memory [12], which are quite close to the theoretical limit of 0.25. Using immersion lithography at 193nm (NA = 1.2), k_1 is required to be <0.2 to print 32nm pattern, which is lower than its theoretical limit. As a result, traditional lithography using 193nm wavelength light cannot print sub-32nm patterns. With significant technical hurdles delaying implementation of new lithography techniques, such as extreme ultraviolet (EUV) [13] and immersion ArF (IArF) [14], double patterning is the only viable solution to adhere to Moore's Law,

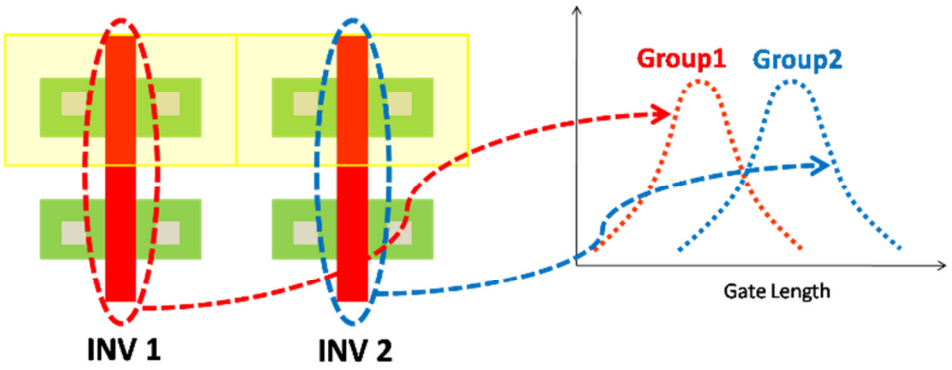


Figure 1.1 Layout of two inverters showing DPL based length variation.

despite the increased cost due to lower throughput and higher process complexity [15]. Double Patterning Lithography (DPL) [16, 17] partitions a critical-layer layout into two mask layouts and exposures, such that each individual exposure step takes place at a robust 0.35-0.4 k_1 factor, which is much more favorable for manufacturing compared to single exposure, ultra-low k_1 lithography. However, DPL incurs added complexity: more processing steps, throughput overhead, and tight overlay control between the two exposures. Several DPL schemes have been proposed in the past [18, 19], however the two most popular techniques are litho-etch-litho-etch (LELE) and the sacrificial self-aligned spacer with trim [20]. In pitch-split DPL either lines or spaces between lines are printed in two sequential processes. Thus, DPL is characterized by the existence of dual populations for critical dimension (CD), with uncorrelated variance and distinct means. So, for a fixed polysilicon gate pitch, devices on alternate poly tracks are correlated while devices on adjacent tracks are not. This violates the assumption of spatial correlation between gate lengths of devices placed close to each other that has always been used to reduce pessimism in corner-based timing analysis. For example, Figure 1.1 shows two inverters laid out next to each other, which are printed with different exposures under DPL. As a result, they will have uncorrelated gate length distributions, so that their

electrical characteristics (delay, power, etc.) can be extremely different from each other despite being adjacent in the same die. While such variation in gate length presents significant challenges to timing analysis and optimization of logic [21], it will have a much stronger negative impact on SRAM robustness where a mismatch between devices (e.g., access and pull-down devices) can cause significant yield loss. On the other hand, spacer double patterning [20] provides excellent variability control, but it restricts the entire layout to one critical dimension. Each DPL implementation has a different impact on line space and linewidth variation, and this creates a need for DPL aware analysis and optimization of VLSI circuits.

1.1.2 Threshold Voltage Variation

For 90nm technology node and earlier, the main cause of V_{th} variation was a purely probabilistic phenomenon (which is independent of other types of variation) known as random dopant fluctuation (RDF). RDF occurs in MOSFET devices because of the random nature of ion implantation [22, 23]. Several past works have addressed threshold voltage variation and many variation models have been proposed to model this effect [24, 25, 26]. However, with process scaling, other factors, such as rapid thermal annealing (RTA) temperature variation across chip, have a significant effect on device V_{th} variation [27]. Higher local anneal temperature drives the junctions both longitudinally and vertically, and causes a higher activation of dopants. This results in lower V_{th} by a combination of short channel effects and compensation of halo doping. V_{th} variation has a significant effect on delay and leakage of VLSI circuits. Device drive current (I_{on}) has a super linear dependence on V_{th} [28], while leakage varies exponentially with V_{th} [29]. With scaling, leakage power consumption is now on the same order as

dynamic power consumption [30], so any variation in leakage power could lead to significant variation in total circuit power.

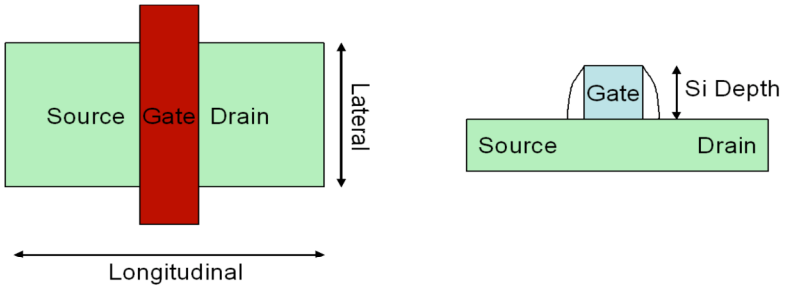
1.1.3 Gate Oxide and Dielectric Layer Thickness Variation

In state-of-the-art process nodes, the equivalent gate oxide thickness, t_{ox}, is on the order of 1nm [31], which is less than five silicon atoms thick. Thus, atomic scale roughness introduced at the gate-to-oxide and oxide-to-silicon interfaces can cause significant amounts of oxide thickness variation [32]. These variations are probabilistic in nature and can lead to variability in mobility, gate tunneling leakage current, and threshold voltage, among other parameters [33]. Several modeling works have focused on accurate modeling and analysis of t_{ox} variation [34, 35, 36].

Aside from the gate oxide, dielectric material between each metal layer in a process also experiences thickness variation. This material is often referred to as the inter-layer dielectric, or ILD. ILD thickness variation is a spatially correlated (systematic) variation that is caused during the Chemical-Mechanical Polishing (CMP) manufacturing step used for planarization of the dielectric material. Since the ILD thickness due to CMP is dependent on topology (regions with higher density interconnect polish slower than sparse regions), ILD thickness is spatially correlated based on the interconnect density [37]. Thus, the impact of ILD variation can be predicted and many publications have sought to provide techniques that ensure uniform metal density and, therefore, reduce the systematic variation in ILD due to CMP [38, 39].

1.1.4 Sub-90nm Induced Variation

Maintaining performance and reliability while facing fundamental scaling limitations (i.e. gate oxide thickness) is a major challenge for semiconductor industry. We can no longer scale certain device parameters such as t_{ox}, V_{th}, V_{DD} as aggressively as gate length (L)



	NMOS	PMOS
Longitudinal	Tensile	Compressive
Lateral	Tensile	Tensile
Si Depth	Compressive	Tensile

Figure 1.2 Desired stress types for NMOS and PMOS [40].

without significantly degrading reliability and exponentially increasing leakage current. Effects like well proximity [31] and mechanical stress due to shallow trench isolation (STI) [40] have emerged in the last decade and now contribute to device variability. Additionally, as MOSFET's continue to scale below 100nm, higher effective fields cause mobility degradation, leading to decreasing drive currents. In order to battle mobility degradation and achieve higher drive currents, modern-day fabrication processes use special means to induce mechanical stress in MOSFET's, which enhances carrier mobility. Mobility enhancement has emerged as an attractive alternative to device scaling because it can achieve similar device performance improvements with reduced effects on reliability and leakage. Mechanical stress in Silicon leads to band splitting and alters the effective mass, which results in carrier mobility changes [41, 42]. Induced stress in the channel can be either tensile or compressive. As illustrated in Figure 1.2, NMOS and PMOS devices have different desired stress types (compressive or tensile) in the longitudinal, lateral and Si-depth (vertical) dimensions. By providing the correct type of stress for a device (in one or more dimensions), we can achieve higher drive currents.

There are four major sources of stress in a modern CMOS technology: eSiGe (generates compressive stress, used only for PMOS) [43], Shallow Trench Isolation (generates compressive stress)[44], compressive, and tensile nitride liners [45], and stress memorization technique (SMT) [46]. Since the size of these stress sources present in the vicinity of device channel depends on the layout, stress induced in the channel of a device has a very strong dependence on the device layout, and the layout of neighboring devices. This creates a layout dependent variation in device mobility across the chip, which has a significant effect on drive current and leakage [47]. More recently, three-dimensional (3D) stacking has emerged as a solution to meet the scaling targets on performance, power dissipation, and packaging form factor [126, 127]. Through-Silicon vias (TSVs) are used to connect dies in the vertical direction achieving higher density, and wirelength reduction. However, they also induce stress in the neighboring devices due to thermal mismatch with silicon. Typically, Copper is used as TSV material due to its low resistivity, and it exerts tensile stress in the longitudinal direction due to a higher coefficient of thermal expansion as compared to Silicon.

Additionally, local anneal temperature variation across the chip causes variation in device properties such as extrinsic resistance (R_{ext}), and gate to source capacitance (C_{gs}) [27]. As anneal temperature increases, higher dopant activation and increased gate overlap of source and drain together result in lower extrinsic transistor resistance (R_{ext}). Similarly, C_{gs} increases with increase in temperature as gate/drain and gate/source overlap increases upon increasing local anneal temperature due to increased dopant diffusion. These in turn affect the performance and leakage of VLSI circuits. Measurement results from a 65nm chip show that ring oscillator frequency can vary by as

much as ~20% based on the position in the die due to local anneal temperature variation [27].

1.2 Main Contributions of Dissertation

This dissertation focuses on addressing the challenges posed by the continuing scaling of semiconductor devices through modeling and optimization. Parameter variations (in L, V_{th}, etc.) cause high yield losses due to the strong dependence of the circuit delay on them. We propose the use of soft-edge flip-flops as an effective way to mitigate yield losses due to parameter variations. Soft-edge flip-flops have a small window of transparency instead of a hard edge for capturing data, allowing limited cycle stealing on critical paths, and thus compensating for delay variations. By enabling time borrowing, soft-edge flip-flops essentially allow random delay variations to average out across multiple logic stages. In addition, it addresses small amounts of delay imbalance between logic stages further maximizing the frequency of operation.

Another important source of variation in modern processes is mechanical stress based device mobility enhancement. Process-induced mechanical stress is used to enhance carrier transport and achieve higher drive currents in current CMOS technologies. However, mechanical stress induced in the device channel, and hence the device drive current and leakage, has a very strong dependence on layout parameters such as active area length, distance of device from the well edge, etc. We study this dependence and propose guidelines to optimize a layout while considering the layout dependence of mechanical stress. Stress is then used as a means to achieve optimal power-performance trade-off by combining stress-based, performance-enhanced standard cell assignment with dual-V_{th} assignment. Based on stress-optimized layouts, we develop a circuit-level, block-based, stress-enhanced optimization algorithm including all layout-

dependent sources of mechanical stress. We also propose a novel standard cell library methodology, entitled STEEL that strives to fully exploit the layout dependencies of mechanical stress. Finally, we propose compact closed-form models for layout dependence of process induced stress, and its impact on carrier mobility. We analyze the physics behind stress inducing process steps, and solve relevant equations describing the stress distribution, in order to develop the models. Since the derivation is based on underlying physics, the derived models are scalable for future technology nodes. We also propose a technique to model non-rectangular gates (NRG) with non-uniform carrier mobility (stress induced) through simultaneous extraction of effective gate length (EGL), and effective carrier mobility (ECM), to enable accurate prediction of both device drive current and leakage.

Next, we focus on Rapid Thermal Annealing (RTA) induced variation in the performance and leakage of VLSI circuits. Suppressing device leakage while maximizing drive current is the prime focus of semiconductor industry. RTA drives process development on this front by enabling fabrication steps such as shallow junction formation that require a low thermal budget. However, decrease in junction anneal time for more aggressive device scaling has reduced the characteristic thermal length to dimensions less than the typical die size. Also, the amount of heat transferred, and hence the local anneal temperature, is affected by the layout pattern dependence of optical properties in a region. This variation in local anneal temperature causes a variation in performance and leakage across the chip by affecting the threshold voltage (V_{th}) and extrinsic transistor resistance (R_{ext}). We propose a new local anneal temperature variation aware analysis framework which incorporates the effect of RTA induced

temperature variation into timing and leakage analysis. We solve for chip level anneal temperature distribution, and employ TCAD based device level models for drive current (I_{on}) and leakage current (I_{off}) dependence on anneal temperature variation, to capture the variation in device performance and leakage based on its position in the layout. We also propose techniques to minimize the impact of anneal temperature variation, and examine their effectiveness and implementation cost.

DPL-aware analysis and optimization of SRAM is the focus of next two chapters. Pitch-split DPL decomposes and prints the critical layout shapes in two exposures, leading to mismatch between adjacent devices due to systematic offsets between the two exposures. This results in adjacent devices with different mean critical dimension (CD), and uncorrelated CD variation. We study the impact of this mismatch on SRAM robustness, and propose a DPL-aware SRAM sizing scheme to mitigate yield loss. We then extend the analysis to include self-aligned spacer DPL, and compare the layerwise impact of different DPL choices on SRAM robustness, density, and printability, to decide the best DPL choice for each layer. We then perform a sizing optimization that accounts for increased variability due to DPL for each layer.

Finally, we propose compact closed-form models for TSV-induced mechanical stress, and its impact on carrier mobility. Model derivation is based on the physics behind stress inducing process step, and it accounts for layout features like STI width, active area length, and neighboring devices. We also propose a new TSV-aware timing analysis framework which embodies transistor level models for TSV stress sensitivity, to incorporate TSV induced stress/mobility variation into traditional timing analysis.

1.3 Organization of Dissertation

The remainder of this thesis is organized as follows. Chapter 2 discusses the use of soft-edge flip-flops for improved timing yield. Chapter 3 focuses on exploring the layout dependence of process induced mechanical stress, and using stress as a means for optimal power-performance tradeoff in conjunction with dual-V_{th} assignment. Chapter 4 focuses on a new stress based standard cell design methodology, STEEL. Chapter 5 proposes closed-form models for layout dependence of mechanical stress, while Chapter 6 discusses simultaneous extraction of effective gate length and effective carrier mobility for non-uniform devices. In Chapter 7, we discuss RTA-aware analysis framework for accurate estimation of delay and leakage. Chapters 8 and 9 study the impact of Double Patterning Lithography (DPL) on SRAM robustness and explore DPL-aware SRAM design to improve robustness. In Chapter 10 we propose a TSV-aware timing analysis framework to study the impact of TSV-induced stress on circuit and device level timing. Finally, Chapter 11 concludes the dissertation with a summary of completed work and a brief discussion of future work.

Chapter 2
Use of Soft-edge Flip-flops for Improved Timing Yield

Modern CMOS processes suffer from large variations in transistor parameters such as length and threshold voltage [48]. As the circuit delay has a strong dependence on these parameters, it shows a significant spread due to process variations. This can result in significant yield losses due to timing failures in the circuit, as nominally sub-critical paths may now become critical. Scaling trends point to an overall increase in the variations with technology scaling. This increase can be attributed to a number of factors, such as increasing difficulty of manufacturing control, increase in atomic scale randomness like variation in dopant profile of the transistor channel, and introduction of new systematic variation generating mechanisms [49]. Hence, there is a need for extensive design and modeling in order to address the variations.

A significant amount of research has focused on process variation aware analysis. One such approach is corner based static timing methodology [50]. However, traditional corner based design approach mostly leads to overly pessimistic guard banding [51]. Moreover, while global variations can be approximated by considering appropriate corner cases, it does not provide a statistical way of modeling the variations across dies. Also, as variations are increasing, number of process corners to be considered for true accuracy is

becoming too large for computational efficiency. A second approach, namely, statistical static timing analysis (SSTA) has emerged as an alternative analysis approach. While much work has been done on the analysis using SSTA [52], there has been a limited work to account for variability in circuit optimization [53, 54, 55, 60]. In [55], the authors extend deterministic optimization using gate sizing approach to the statistical domain, while in [56], authors used geometric programming to address the problem. Most of the optimization in this area uses gate sizing as a solution.

Retiming and so-called useful clock skew assignment has been considered as an effective way to balance out the path delays and maximize the frequency of operation. In the past, this has been formulated as a linear programming problem [57]. However, with the exception of [58], these solutions are deterministic in nature and do not consider the effect of process variations. Useful skew assignment also results in a large number of paths pushed to the very edge of satisfying the timing requirements. Hence, even a slight variation in delay on these critical paths can cause a timing based failure. This makes useful skew assignment more effective in improving nominal delay than for improving yield. Furthermore, useful skew assignment is typically performed on an individual flip-flop basis, which is difficult in practice. In the results of this chapter, we show that performing useful skew assignment for clusters of flip-flops can address clock skew, but is ineffective for addressing process variation and circuit imbalance.

Another approach to address delay variations in the circuit is to use latch based design. Latches have no hard boundary and are transparent for half a clock period. This means that there can be variations in the data arrival time and still the correct data would be captured by the latch. In [59], the authors present clock scheduling for latches in order

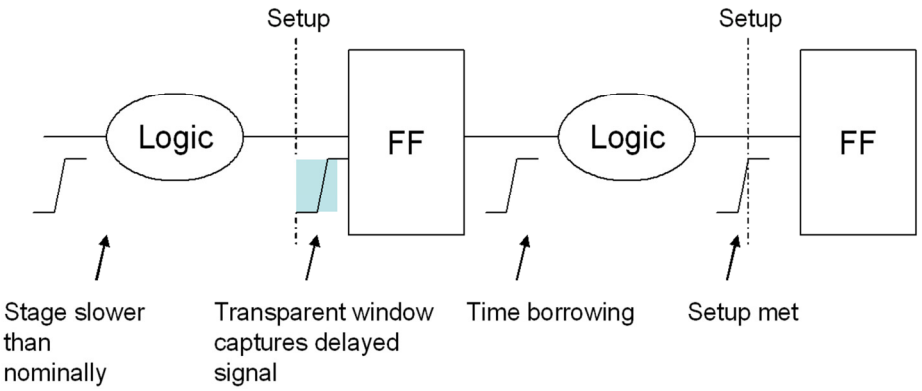


Figure 2.1 Transparency window compensating for delay variation.

to improve yield considering the delay variations. However, latch based design has its own limitations. The most important problem with latch based design is the need for two separate clocks, which means significant power and area overhead. Furthermore, generating two non-overlapping clocks can be difficult in high performance design. The goal of our work is to use the concept of averaging and cycle stealing, while maintaining the conventional flip-flop based paradigm.

We propose the use of soft-edge flip-flops as a useful method to address process variation as well as limited circuit imbalance through time borrowing and averaging across stages. The key idea is to delay the clock edge of the master latch so as to create a window of transparency instead of a hard boundary for capturing the data. As shown in Figure 2.1, the transparency window allows cycle stealing in order to compensate for the difference in delays of the two paths resulting from either circuit imbalance or from local random process variations. Soft-edge flip-flops utilize the fact that random variations average out along a circuit path, and by allowing averaging across stages they reduce the sensitivity of the design towards process variations. By creating a window of transparency, we increase the hold time of the flip-flop. However, as the window is small, the resulting increase in hold time is not large enough to cause hold time violations. This

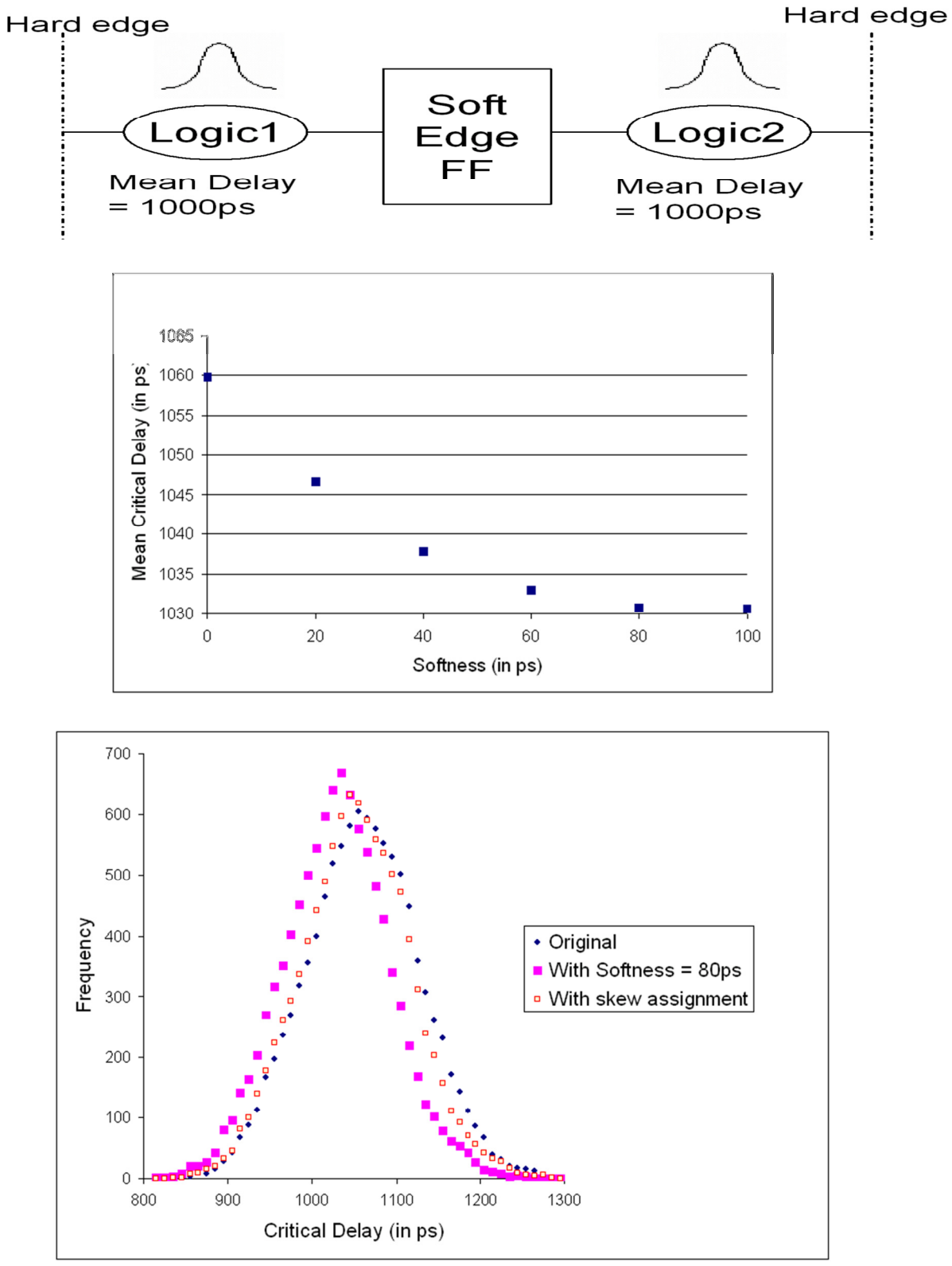


Figure 2.2 Setup for demonstrating the effectiveness of soft-edge flip-flops and the corresponding plots of mean v/s softness and delay distribution for softness of 0 and 60 ps and for skew assignment.

is verified for all the test circuits as discussed later in the results section. The softness comes at the cost of power. Hence, we designed a library of soft-edge flip-flops with varying amounts of softness in order to analyze the power overhead involved. A variation aware algorithm was then used to statistically optimize yield with minimum power overhead.

The rest of the Chapter is organized as follows. Section 2.1 discusses the proposed methodology while Section 2.2 describes the soft-edge flip-flop assignment technique. The design of the soft-edge flip-flops is discussed in Section 2.3. Sections 2.4 and 2.5 discuss the experimental results and summary, respectively.

2.1 Proposed Methodology

Figure 2.2 shows a sample setup to demonstrate the effectiveness of soft-edge flip-flops in addressing process variations. The setup consists of two stages of logic(Logic1 and Logic2) separated by a soft-edge flip-flop whose softness is varied from 0 to 100ps in steps of 20 ps. Logic1 has a nominal delay of 1050 ps, and Logic2 has a nominal delay of 1000 ps. In order to simulate delay variations, each of them has a delay distribution with a variation of 5%(/) and another 5% of random component. In the absence of softness the critical delay is the maximum of the two delays. However, on introducing softness in the flip-flop, for cases where the path preceding the flip-flop (Logic1) is critical, the softness is used to borrow time from the next stage (Logic2) to average out the delays. This leads to the shift in the mean delay as softness is increased. If skew assignment was done, it would have assigned the flip-flop a skew of 25 ps, so as to balance out the nominal delays. And for each sample, the critical delay would have been the maximum of the two delays.

With standard D flip-flop, the distribution of critical delay has a mean of 1059.8 ps, and a standard deviation of 68.4 ps. As shown in the 2.2, the mean decreases as softness is increased and becomes almost constant after a softness of 80 ps. This is because almost all the cases involving the path preceding the soft-edge flip-flop being critical, have been averaged out by time borrowing to the maximum limit allowed by the slack available. Beyond this point, increasing softness has little effect on the mean. The value of mean for softness of 80 ps is 1030.7 ps with a standard deviation of 63.6 ps. This means that the mean of the delay improved by 29.1 ps which is fairly large fraction of the initial standard deviation (about 42.5%). For the case of skew assignment, the mean delay is 1053.1 for an improvement of only 6.7 ps with a standard deviation of 66.7 ps. While skew assignment obtains a 25ps improvement in nominal delay, the statistical mean of the delay under process variation is improved by much less. This is caused by the balancing of delay paths in skew assignment, which makes the design more sensitive to process variation and counter acts the improvement in nominal delay. The delay distribution curves plotted for softness of 0 and 80ps along with the distribution for the case of skew assignment is shown in Figure 2.2. Note that skew assignment shifts the delay distribution curve slightly to the left, while the shift is much more significant in the case with softness of 80 ps. This shift corresponds to the improvement in mean. The curve for the case of 80 ps softness is narrower than that for no softness, and this shows up as improvement in the standard deviation.

As seen in this example, there is little improvement in mean if we increase the softness beyond 80 ps. Softness comes at a cost of power overhead, thus, there is a need for an algorithm to intelligently assign softness so as to get best possible improvements in

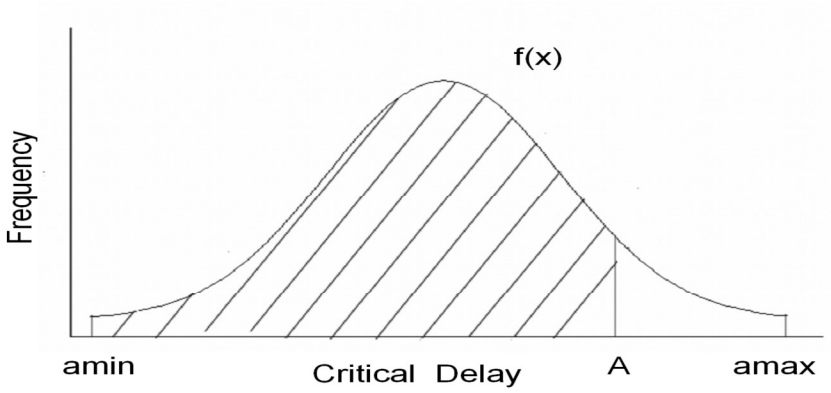


Figure 2.3 Calculation of yield for a given clock frequency from the critical delay distribution curve.

mean for minimum power. We discuss the soft-edge flip-flop assignment technique to minimize power overheads in the next section. In order to analyze the power overhead involved, a library of soft-edge flip-flops was designed. The method to design a soft edge flip-flop is to delay the master's clock. This can be done in different ways, and this is described in Section 2.4.

2.2 Soft-edge Flip-flop Assignment Technique

Soft-edge flip-flops have a power overhead associated with them due to the additional delaying elements used to delay the clock to the master latch. The flip-flop assignment problem can thus be formulated as the minimization of power for a given yield constraint. In theory, for a custom design, it is possible to construct a soft-edge flip-flop with any arbitrary amount of softness, by proper sizing. However, for a typical standard cell based design, there will be a library of such flip-flops with varying amounts of softness, and so our goal in this case is to assign flip-flops available in a library. Thus, the design variable which is the amount of softness, is a discrete variable, while the yield constraint is statistical in nature.

The calculation of yield from the statistical data is illustrated in Figure 2.3. For a given clock period, there is a certain value of critical delay beyond which a circuit fails to

meet the timing. This value is shown as A in the figure. So for all values of delay between amin and A, the circuit meets the timing requirements and for delays beyond that it fails. The yield in this case is given by the area of the shaded portion divided by the total area under the curve. Thus, the yield Y is given by equation 1.

$$Y = \frac{\int_{amin}^{A} f(x)dx}{\int_{amin}^{amax} f(x)dx} \tag{1}$$

Clearly, any yield constraint of the form $Y>Y_o$ can be translated into an equivalent constraint on the critical delay A for that given clock period expressed as $A > A_0$. Now, we can express the flip-flop assignment problem as a simple robust integer linear programming (ILP) problem as shown below:

$$Minimize\ power\ P = \sum_{f=1}^{F}\sum_{k=1}^{n} P_k v_{fk}\ \ subjet\ to:$$

$$Y(x_1, x_2, \dots, x_F) > Y_o \tag{2}$$

$$\sum_{k=1}^{n} v_{fk} = 1 \tag{3}$$

$$\sum_{k=1}^{n} S_k v_{fk} - x_f = 0 \tag{4}$$

$$v_{fk} \epsilon \{0,1\} \tag{5}$$

where F is the total number of flip-flops in the circuit, and n is the total number of quantization levels of softness that a flip-flop can have (softness is a discrete design variable and can be varied in steps). v_{fk} is the k_{th} quantization level variable associated with the f_{th} flip-flop, S_k is the softness associated with the k_{th} quantization level, and x_f is the value of softness for the f_{th} flip-flop. Note that softness associated with the first quantization level is 0, and this corresponds to the standard D flip-flop from the library. P_k is the power consumed by a flip-flop with softness S_k. The power for the rest of the

circuit remains the same, and the only overhead is due to the assignment of softness to the flip-flops. Constraint (2) is the simple yield constraint while constraints (3), (4) and (5) together make sure that each flip-flop has only one value of softness assigned to it out of the n possible values corresponding to the n levels of quantization. Setup and hold time constraints are essentially captured in the yield constraint and hence they need not be expressed independently. The yield constraint makes the problem variation aware, as yield depends on the delay distribution caused due to process variations. There have been many advancements in efficiently solving robust integer linear programming problems. However, a typical industrial circuit consists of several thousands of flip-flops and the runtimes for solving the corresponding ILP will most likely be very large. Hence, we use a less complex greedy heuristic to assign softness, while the ILP is considered for possible future extensions.

We use a greedy algorithm 'GREEDY_SOFTNESS( )' to solve this problem of soft-edge flip-flop assignment. The idea is to search for the most critical path in the circuit (considering process variations), and increase the softness of the flip-flop following the path by one step. This is to be repeated till the yield requirements are just met, or when there ceases to be any appreciable improvement upon increasing softness. The second criterion ensures that we stop assigning further softness if the improvement gained by that step is not appreciable, and that we stop in case the yield target is not achievable for the given constraint. The pseudo code for GREEDY_SOFTNESS( ) is as follows:

GREEDY_SOFTNESS( )

1 continue <-- true

2 while continue == true

3 find most critical path P

4 find current yield Y

5 check <-- false

6 if P ends at a flip-flop f's input

7 then increase softness of the flip-flop f by one step

8 check <-- true

9 find new yield Ynew

10 if Ynew > Y0

11 then continue <-- false

12 else if (Y - Ynew) is not appreciable

13 then continue <-- false

14 if check == true

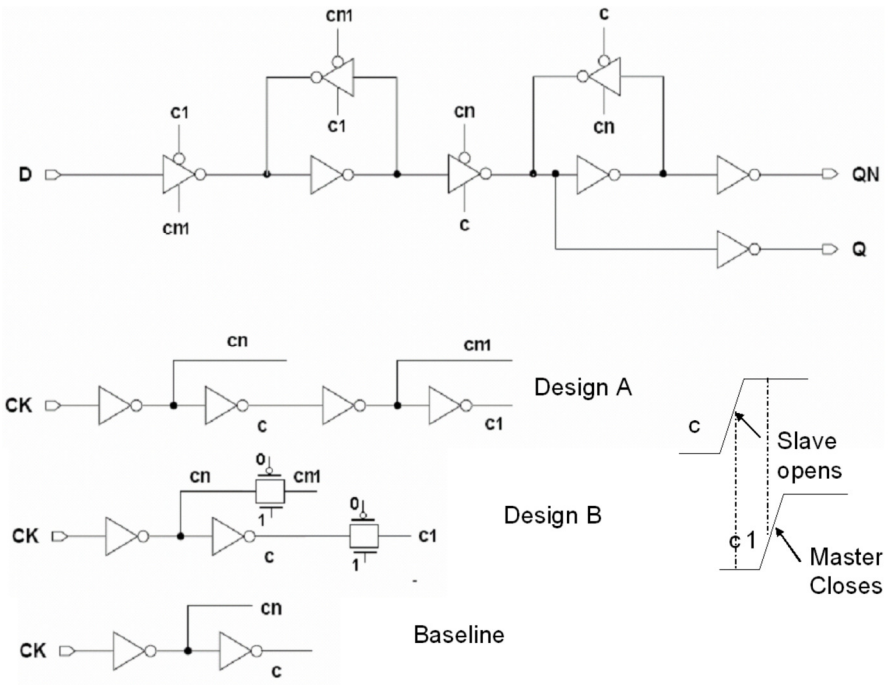


Figure 2.4 Designs for Soft-edge Flip-flops.

15 then decrease the softness of flip-flop f by one step.

As we show later in the results section, we obtain near maximum delay improvements with small power overheads (less than 3%), by using this greedy algorithm for assigning soft-edge flip-flops.

2.3 Soft-edge Flip-flop Design

Figure 2.4 shows the design of a soft-edge flip-flop. Like a standard D flip-flop, it is constructed using back to back master/slave latches. Each latch consists of a tristate inverter feeding into the latch followed by a cross-coupled inverter pair. For a standard D flip-flop, at the positive edge of the clock, the tristate feeding the master closes while the tristate forming the feedback loop opens thereby latching the value into the master latch. At the same time, the tristate feeding the slave opens, while the feedback tristate closes, making the slave transparent. This allows the outputs Q and QB to be evaluated based on the value latched in the master. So, in order to be displayed correctly at the output, the data should arrive sometime before the rising edge of the clock. This time is called the setup time and it defines the hard edge before which the data should arrive for a standard D flip-flop.

In order to create a window of transparency (soft edge), we need to delay the clock supplied to the master latch by a small amount. The amount of delay is the period for which both the master and the slave are transparent together. This is because when the slave's clock makes a transition causing it to become transparent, the tristate feeding the master is also open as its clock is a delayed version of the slave clock. In this way, even though slave is transparent, a change in data value can still change the state of master

which, in turn, will change the value of output. As shown in Figure 2.4, signal c1 which is fed to the master latch is the delayed version of signal c which controls the slave latch. For the period between the two vertical lines, slave has opened while the master has not closed and this is the period of transparency or the amount of softness. This softness allows time borrowing if the stage following the flip-flop has a slack.

As in the original D flip-flop c was one inverter delayed version of cn, in our new design c1 and cn1 (supplied to the master) should have be such that c1 is more delayed than cn1. This ensures that our design is as close to the original as possible, if we match the rise time for the new signals to the original ones. Keeping this in mind, there are two ways of delaying the clock. One is to simply insert two more inverters after c as shown in design A. Here, we need to size the inverter chain such that cn1 and cn have the same rise time, while c and c1 have the same rise time and this should be close to the rise time of the corresponding signal in the standard D flip-flop. Along with the risetime we need to ensure that the delay between cn1 (c1) and cn (c) is the same as the value of softness required. For higher values of softness, more inverters might be added to the chain of inverters.

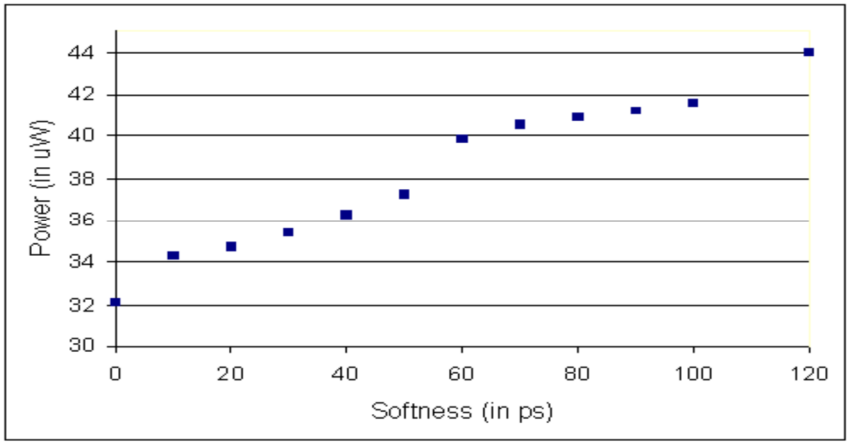


Figure 2.5 Power consumption for the designed soft-edge flip-flops.

Table 2.1 Area and power overheads for soft-edge flip-flops.

Softness	Percentage power overhead	Percentage area overhead
10	6.8	18
20	8.2	18
30	10.2	18
40	12.9	18
50	16	18
60	24.1	32
70	26.2	32
80	27.2	32
90	28.3	32
100	29.5	32
120	36.8	45

However, for design A, cn1 (c1) is two inverter delayed version of cn (c). This means that there is a lower limit to the softness that we can assign which is determined by the minimum delay of two inverters. For lower values of softness, we use design B, which uses pass gates for delaying the signals. In design B as shown in the Figure 2.4, c1 and cn1 are delayed versions of c and cn respectively, which have been delayed using pass gates. By appropriate sizing much lower values of softness are possible. In our designed library, we use design B for getting the softness of 10 ps and 20 ps, while design A is used for all other values.

Figure 2.5 shows the power values for flip-flops with different values of softness. Note that softness of 0 corresponds to the standard D flip-flop. Power has been measured using Hspice, assuming the switching activity of the data to be 0.1 . The sudden increase in power when we go from softness of 50 ps to 60 ps, is due to the fact that two more inverters have to be added to the inverter chain in order to get values of softness greater than 50 ps. A similar increase can be seen when increasing the softness from 100ps to 120ps. Table 2.1 gives the power and area overheads for the library of soft-edge flip-flops.

Area overheads are computed by laying out the soft-edge flip-flops and comparing it to the area of standard cell. The power and area overheads for the flip-flop with the largest softness (120ps) are 36.8% and 45%. However, we find that only a small fraction of flip-flops need this level of softness. Hence, the overall power overhead was found to be a very small percentage (less than 3%) of the overall circuit power, as reflected in the experimental results which are discussed in the next section.

2.4 Experimental Results

The proposed soft-edge flip-flop assignment technique was implemented based on IBM 0.13µm technology. All the test circuits were synthesized, placed and routed using commercial tools. The test circuits ranged in size from 119-36544 gates. The physical placement details were used to cluster the flip-flops for the purpose of clustering based skew assignment. We compare our results against skew assignment for cluster sizes of 30 (large), 15 (moderate), and 5 (small). Based on industrial estimates [61], we use a switching activity of 0.1 while measuring power.

Table 2 summarizes the main results for the proposed approach and compares them with clustering based skew assignment for ISCAS89 benchmark circuits and two DSP circuit implementations ('Viterbi1' and 'Viterbi2'). The first three columns give the name and size of each test circuit. The fourth and fifth columns give the mean and standard deviation of delay for the original test circuit without any optimization. The next three pairs of columns give the percentage improvement in mean and standard deviation of delay, by using clustering based skew assignment technique for cluster sizes of 30, 15 and 5. The next pair of columns gives the percentage improvement in mean and standard deviation by using soft-edge flip-flops. Note that this is the best possible improvement which can be achieved by using soft edge flip-flops. The last two columns provide an

Table 2.2 Improvement in mean delay for several circuits by soft-edge flip-flop assignment as compared to clock skewing.

Delay (ps)	No. of gates	No. of flip-flops	Values for Original Circuit		Percentage Improvement by skew assignment (Cluster Size = 30)		Percentage Improvement by skew assignment (Cluster Size = 15)		Percentage Improvement by Skew assignment (Cluster Size = 5)		Percentage Improvement by Softness		Improvement by softness compared to clustering (size=3)	
			μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
s298	119	14	345.5	15.2	0.0	0.0	0.0	0.0	1.6	2.8	14.3	6.3	8.8X	2.2X
s344	160	15	567.5	30.8	0.0	0.0	0.0	0.0	5.6	3.5	17.1	11.1	2.9X	3.2X
s349	161	15	573.5	31.3	0.0	0.0	0.0	0.0	7.4	4.4	17.2	14.0	2.3X	3.1X
s386	159	6	351.5	15.6	0.6	3.2	0.6	3.2	0.6	3.2	2.6	4.1	4.6X	1.2X
s400	164	21	490.5	25.4	0.0	0.0	0.0	0.0	0.0	0.0	11.5	15.6	NA	NA
s420	218	16	731.4	42.4	0.0	0.0	4.8	7.4	5.9	7.5	6.8	15.2	1.2X	2.0X
s510	211	6	521.5	27.6	1.2	0.4	1.2	0.4	2.0	0.9	6.5	8.7	3.3X	9.9X
s526	194	21	488.5	25.3	0.0	0.0	0.0	0.0	15.6	13.3	22.6	22.8	1.4X	1.7X
s820	289	5	531.2	28.3	1.1	4.2	1.1	4.2	1.1	4.2	4.1	5.5	3.8X	1.3X
s832	287	5	537.5	28.7	0.0	0.0	0.0	0.0	0.0	0.0	3.9	5.1	NA	NA
s838	446	32	1144.4	71.5	4.4	0.9	4.4	0.9	4.8	1.1	6.7	3.6	1.4X	3.4X
s1196	529	18	644.5	36.3	4.9	2.8	6.4	8.1	6.8	10.7	16.9	24.1	2.5X	2.2X
s1238	508	18	683.4	39.0	3.7	4.6	3.7	4.6	3.7	4.6	9.5	10.3	2.5X	2.3X
s1423	657	74	1688.5	99.2	0.3	0.1	0.4	0.3	0.5	0.3	3.1	1.9	6.8X	5.9X
s1488	653	6	579.4	31.7	0.0	0.0	0.0	0.0	0.0	0.0	2.1	2.7	NA	NA
s15850	9812	534	1456.0	92.5	0.7	0.6	1.8	1.9	1.8	1.9	6.0	5.9	3.4X	3.0X
s35932	16065	1728	480.4	24.0	3.7	5.1	3.7	5.1	3.7	5.1	8.9	12.2	2.4X	2.4X
s38417	15106	1636	1466.0	62.8	1.8	2.2	3.1	4.6	5.0	6.4	8.1	11.4	1.6X	1.8X
Viterbi1	12024	4215	812.3	46.8	0.3	0.3	0.3	0.3	2.3	3.5	6.1	6.1	2.7X	1.8X
Viterbi2	36544	5788	1744.3	70.0	0.2	0.7	0.3	1.7	0.5	2.8	3.3	12.8	6.4X	4.5X
Average					**1.1**	**1.2**	**1.6**	**2.1**	**3.4**	**3.8**	**8.9**	**10.0**	**2.6X**	**2.6X**

estimate of how this improvement compares with that for skew assignment technique for a cluster size of 5. Essentially, it is the ratio of the improvement achieved by using soft-edge flip-flops, to that achieved by using clustering based skew assignment for cluster size of 5. Cluster size of 5 is difficult to achieve. This is due to the fact that several flip-flops are typically run by a single delay generator.

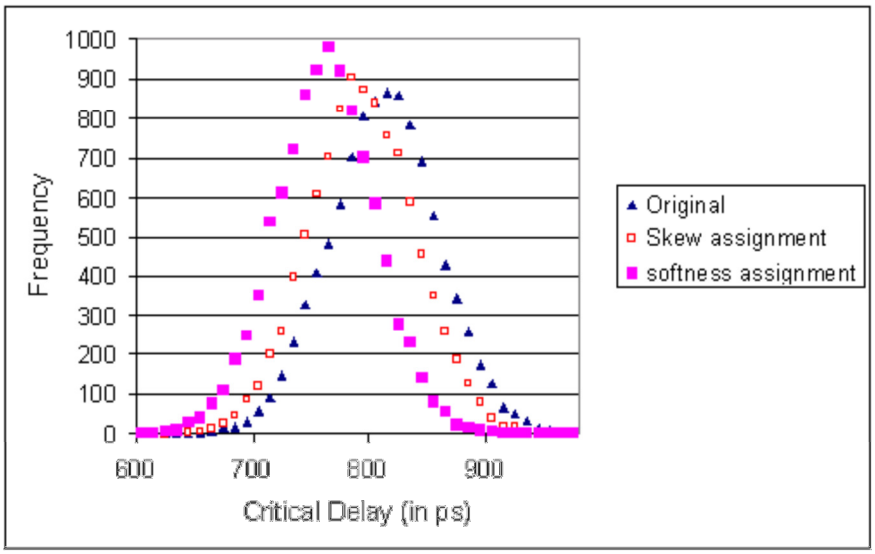


Figure 2.6 Delay distributions for the circuit Viterbi1- for the original, skew assignment and softness assignment cases.

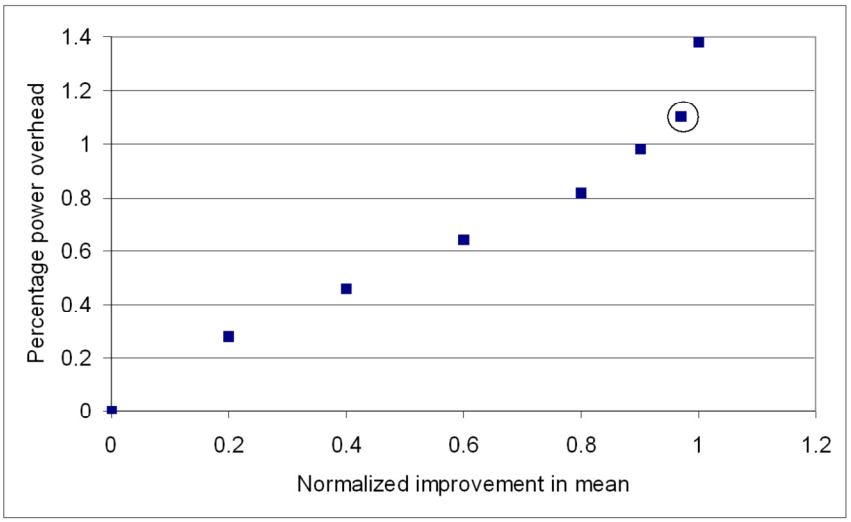


Figure 2.7 Power overhead versus improvement in mean delay for the circuit Viterbi1.

The results clearly show that our approach gives significantly better improvements in mean and standard deviation even when compared to skew assignment for a cluster size of 5, which is difficult to achieve. We get improvements of up to 22.6% (8.9% on average) in the mean and up to 24.1% (10% on average) in the standard deviation of the delay, over the original circuit. As compared to skew assignment with a

cluster size of 5, our approach gives improvements of up to 8.8X (2.6X on average) in the mean and up to 9.9X (2.6X on average) in the standard deviation of the circuit delay. Based on the results, we can conclude that performing useful skew assignment for clusters of flip-flops can address clock skew, but is ineffective for addressing process variation and circuit imbalance.

Figure 2.6 shows sample delay distribution curves of one of the larger circuits (Viterbi1) for the original circuit without any optimization, for the case of skew assignment (cluster size =5), and for the case of using soft-edge flip-flops. As shown in Table 2, the mean delay improves by 2.3% for skew assignment, and by 6.1% for the case in which soft-edge flip-flops are used. This change is depicted in the shifting of the curves towards left. The shift is less for skew assignment (lesser improvement), and a much larger shift is observed for soft-edge flip-flops. Also, the curves become narrower as we move from the original circuit to skew assignment and to soft-edge flip-flop assignment. This corresponds to the improvement in the standard deviation by using the two approaches.

Figure 2.7 shows the plot of power overhead versus improvement in mean for the circuit Viterbi1. The point corresponding to the improvement obtained by greedy algorithm has been circled in the figure. Note that beyond the circled point, increase in power is much more as compared to the further improvement in mean that we can obtain. Figure 2.8 shows the

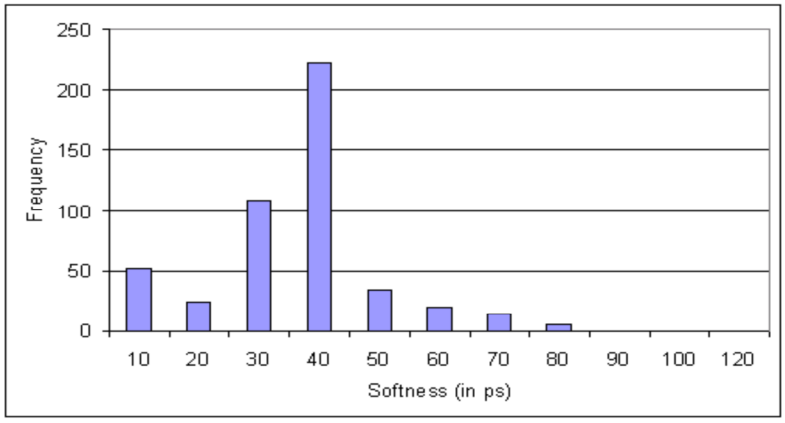


Figure 2.8 Distribution of softness for the circuit Viterbi1.

Table 2.3 Results for softness assignment using greedy algorithm.

Circuit	percentage improvement in μ	Improvement as a fraction of best possible improvement	percentage overhead in power	Percentage of flip-flops assigned softness
s298	13.8	0.96	1.2	64.3
s344	16.9	0.98	1.8	20.0
s349	16.5	0.96	2.2	33.3
s386	2.4	0.96	1.9	50.0
s400	11.4	0.98	1.8	23.8
s420	6.4	0.94	2.0	25.0
s510	6.2	0.96	0.9	33.3
s526	22.3	0.99	2.8	66.7
s820	3.9	0.94	2.6	80.0
s832	3.7	0.95	2.6	80.0
s838	6.4	0.95	2.1	60.0
s1196	16.6	0.98	1.1	27.8
s1238	9.3	0.98	2.7	27.8
s1423	3.1	0.98	2.6	23.0
s1488	1.9	0.93	0.3	16.7
s15850	5.9	0.99	2.7	32.2
s35932	8.7	0.98	1.2	9.3
s38417	8.0	0.99	1.0	9.5
Viterbi1	6.0	0.98	1.1	11.4
Viterbi2	3.3	0.98	1.1	9.6
Average	**8.6**	**0.97**	**1.8**	**35.2**

corresponding distribution of softness after running the greedy algorithm on the circuit Viterbi1. Maximum softness used in this case is 80 ps.

Table 2.3 shows results for soft-edge flip-flop assignment using the greedy algorithm. Columns two and three give improvement achieved in mean and express it as a fraction of the best possible improvement that we can get using soft-edge flip-flop assignment (reported in Table 2). Next two columns give the percentage power overhead and the percentage of flip-flops that were assigned some softness. Power overhead is computed based on the extra power used for making the flip-flops soft, as compared to the power for the original circuit, assuming a data switching activity of 0.1. On an average, we get 0.97 of the best possible improvement in mean by using the greedy algorithm for a power overhead of 1.8%. The results show that we get very close to the best possible improvement by using greedy algorithm, with a small power overhead. The power overhead decreases for the larger circuits. This is because the only overhead is due to the assignment of soft-edge flip-flops while the rest of the circuit power remains the same before and after the optimization. For larger circuits, this overhead forms a much smaller fraction of the overall power. For a more complex approach, the small gains are likely to be overweighed by the runtime overheads.

Soft-edge flip-flops have a window of transparency instead of a hard edge. So, it is necessary to check for possible hold time violations. Figure 2.9 shows the short path slacks for all test circuits after soft-edge flip-flop assignment. Short path slack is the amount of time by which the minimum path delay meets the hold time violation constraint. As shown in the figure, the hold time constraints are met comfortably, and there is substantial slack for the circuit to meet the constraints after variations. This is

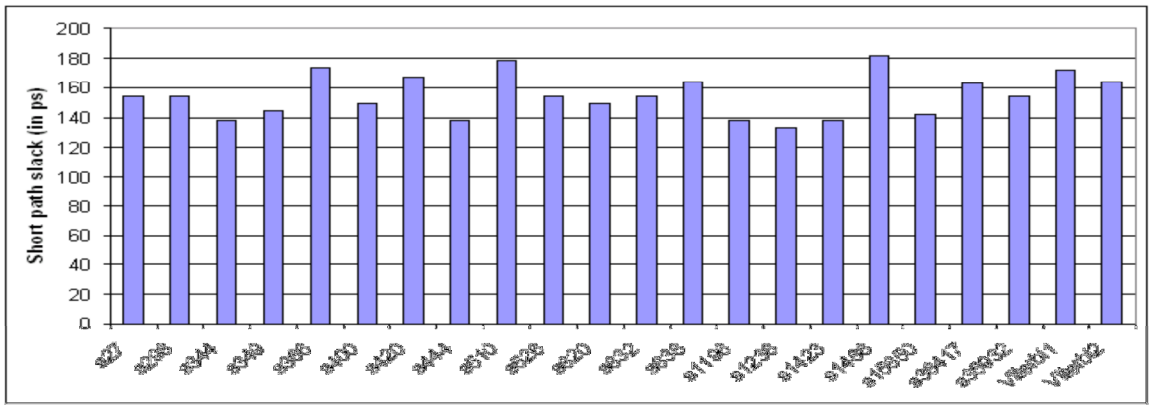


Figure 2.9 Short path slack after soft-edge flip-flop assignment.

expected because the window of transparency is very small even for the largest value of softness (about 3FO4), and so the resulting increase in holdtime is not large enough to cause hold time violations.

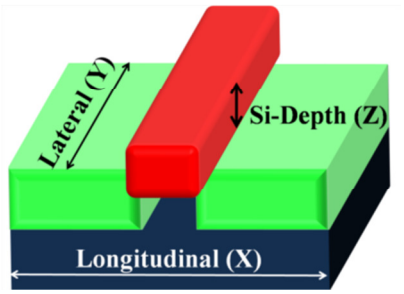
2.5 Summary

In this chapter, we proposed the use of assigning soft-edge flip-flops as a process variation tolerant technique for fine grained balancing of a circuit, in order to improve the timing yield. Soft edge flip-flops allow time borrowing and averaging across stages, making the design less sensitive to process variations by taking advantage of the fact that random variations average out. We constructed a library of soft-edge flip-flop variants, with softness of up to 120 ps (about 3FO4). As soft-edge flip-flops have a power overhead associated with them, we used a statistically aware greedy algorithm to intelligently assign these flip-flops in order to minimize the power overhead. Experimental results show that as compared to clustering based skew assignment technique, our approach provides improvements of up to 22.6% (8.9% on average) in the mean and up to 24.1% (10% on average) in the standard deviation of the delay, with a very small power overhead (less than 3%).

Chapter 3
Mechanical Stress Aware Optimization for Leakage Power Reduction

Maintaining integrated circuit (IC) performance and reliability in modern-day semiconductor processes, while continuing aggressive process scaling, is becoming increasingly difficult because of fundamental scaling limitations. Device parameters like oxide thickness (t_{ox}), threshold voltage (V_{th}), and supply voltage (V_{dd}) can no longer be scaled as aggressively as gate length (L) without significantly degrading reliability and exponentially increasing leakage current. Furthermore, as MOSFET's scaled below 100nm, process engineers sought to battle the mobility degradation caused by larger effective electric fields. To ameliorate mobility degradation and the subsequent drive current reduction, several process techniques have been developed which induce mechanical stress in a device's channel. Mobility enhancement has emerged as an attractive alternative to voltage and oxide thickness scaling because it can obtain similar device performance improvement, with reduced effects on reliability and leakage.

Mechanical stress in silicon breaks crystal symmetry and removes the 2-fold and 6-fold degeneracy of the valence and conduction bands, respectively [41, 42]. This leads to changes in the band scattering rates and/or the carrier effective mass, which in turn affects carrier mobility. Mechanical stress induced in a CMOS channel can be either tensile or compressive. As illustrated in Figure 3.1, NMOS and PMOS devices have



	NMOS	PMOS
X	Tensile	Compressive
Y	Tensile	Tensile
Z	Compressive	Tensile

Figure 3.1 Desired stress types for NMOS and PMOS devices.

different desired stress types (compressive or tensile) in the longitudinal, lateral, and Si-depth (vertical) dimensions. By providing the correct type of stress for a device (in one or more dimensions), we can achieve higher drain currents. However, since carrier mobility affects the drain current in all MOSFET operation regimes, increased carrier mobility not only increases saturation current, it also increases subthreshold current. Specifically, short-channel MOSFET saturation drain current, $I_{D,sat}$, has a sub-linear dependence on mobility, μ_0, while the subthreshold drain current ($I_{D,sub}$) dependence on mobility is linear [28, 29]. These two relationships between drain current and mobility make mobility enhancement an interesting alternative to other power/delay optimization techniques.

One of the most popular power/delay optimization techniques that has been researched considerably in both academia and industry is the dual-V_{th} optimization scheme [62, 63]. This technique typically uses gate sizing and two choices of threshold voltage to optimize a given circuit for some metric (usually delay or power). Since $I_{D,sat}$ and $I_{D,sub}$ are super-linearly and exponentially dependent on V_{th}, respectively, V_{th} can potentially be a powerful optimization parameter. However, since incorporating different threshold voltages adds significant design and process complexity, practical implementations typically restrict the number of threshold voltages to ~2 [64].

One of the main disadvantages of using a dual-V_{th} scheme is, coincidentally, also one of its strengths: each gate in the design can either be high-performance or low-leakage. Dual-V_{th} provides for a wide range of performances (due to the super-linear and exponential dependencies of $I_{D,sat}$ and $I_{D,sub}$ on Vth, respectively), but the approach has only coarse granularity in its selection. Mobility enhancement induced by mechanical stress, however, is layout dependent and can therefore provide much finer delay-versus-leakage control without adding to process complexity/cost. This granularity, coupled with the fact that leakage is only linearly dependent on mobility, makes stress-induced mobility enhancement an interesting research topic that can either be directly compared to dual-V_{th} assignment, or used concurrently to provide additional gains in either leakage or delay. Since the leakage penalty incurred by mobility enhancement is significantly less than V_{th} assignment, we focus on leakage reduction in this work. However, for completeness, we also show that our joint optimization framework can be used to reduce circuit delay for iso-leakage.

To date, there has been limited research on the layout dependence of stress-based current improvement. Most of the published work has focused on the effects of Shallow Trench Isolation (STI) [65-68] or limited their analysis to only include the PMOS sources of mechanical stress [69-72]. Reference [73] studies variability in CMOS circuits for a low power 45nm test chip featuring STI and tensile nitride liner as sources of stress (NMOS only). One key result is that NMOS devices show 5% higher performance as source/drain diffusion lengths are increased by 75%, which is qualitatively similar to our results for a process with added stress sources for both PMOS and NMOS. In the last few years, researchers have begun exploring layout optimization techniques involving stress.

In [66], the authors presented an active-layer fill insertion technique which optimized circuit delay by exploiting STI stress. However, in the 65nm industrial technology used in this research, we discovered that the STI stress contribution was <10% of the total channel stress, making STI optimization less effective. The first optimization scheme developed to exploit the source/drain length dependency was published in [74], which described a timing closure technique that utilized stress enhanced versions of standard cells to improve path delays. While the authors in [74] do report average delay savings of ~5%, they do not disclose the additional leakage power consumed, nor do they discuss possible leakage versus delay tradeoffs.

This work described in this chapter differs from previously published research in that it incorporates all of the layout dependent sources of stress and, consequently, exploits a larger number of layout properties that affect stress (e.g., source/drain lengths, contact placement, distance from STI, etc.). Additionally, unlike [74], our optimization algorithm is not a one-sided approach that only optimizes delay. The proposed optimization accounts for the tradeoff between leakage and delay and it achieves the largest improvement in leakage power (delay) for identical delay (leakage power). Thus, to our knowledge, this is the first work to use stress-enhanced standard cells in a new, circuit-level, block-based, joint optimization framework that improves either leakage power consumption for iso-delay performance or circuit delay for iso-leakage-power consumption.

In this chapter, we begin by addressing the layout dependency of stress-based performance enhancement. We perform a comprehensive study in order to determine how various layout parameters affect device stress, and then analyze their impact on device

performance. From this study we then extract the main layout properties that impact mechanical stress in our industrial, 65nm process. Next, these layout properties allow us to create "high-Stress" and "low-Stress" versions of a subset of standard cells from an industrial 65nm CMOS library (analogous to "low-V_{th}" and "high-V_{th}" cells in a dual-V_{th} library). Finally, we propose a stress-aware optimization algorithm and generate two comparisons: 1) stress-based performance enhancement versus dual-V_{th} assignment, and 2) combined stress-based enhancement with dual-V_{th} versus only dual-V_{th}.

The rest of the chapter is organized as follows. Background for this work is discussed in Section 3.1. Section 3.2 presents a study on the layout dependence of stress-based performance enhancement, while Section 3.3 outlines stress-dependent layout properties for our 65nm technology. Results obtained by modifying these properties in 65nm industrial standard cells is discussed in Section 3.4. Section 3.5 includes details on the proposed optimization methodology. The experimental setup and results for the optimization algorithm are reported in Section 3.6, and Section 3.7 concludes with a brief summary.

3.1 Proposed Methodology

This section discusses the two main topics that are the foundation of this work: the sources of mechanical stress (and their dependency on layout properties) and how mobility and V_{th} affect drain current.

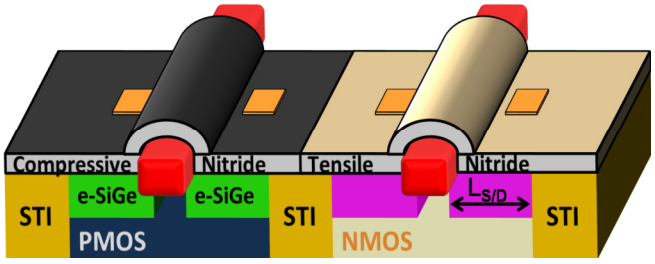


Figure 3.2 Sources of stress for NMOS and PMOS devices.

3.1.1 Mechanical Stress Sources and their Layout Dependence

Mechanical stress in silicon can be generated by either thermal mismatch or lattice mismatch. Thermal mismatch stress is caused by differences in the thermal expansion coefficient, while lattice mismatch stress is caused by differences in lattice constants. Figure 3.2 shows the major sources of stress for one of the latest 65nm CMOS technologies [75]. The sources are Shallow Trench Isolation (STI), embedded SiGe (only in PMOS devices), tensile/compressive nitride liners (in NMOS/PMOS devices, respectively), and the Stress Memorization Technique (SMT).

Shallow Trench Isolation (STI): STI creates compressive stress longitudinally and laterally due to thermal mismatch [66,68-70] and volume expansion [70]. From Figure 3.1, it is apparent that this compressive stress degrades the electron mobility in NMOS devices (in both the longitudinal and lateral directions) [76] and degrades hole mobility in PMOS devices in the lateral direction. However, STI stress that is induced longitudinally (e.g., at the left and right boundaries of standard cells) actually improves hole mobility in PMOS devices.

Embedded SiGe (eSiGe): For PMOS transistors, an eSiGe process is implemented where SiGe is epitaxially grown in cavities that have been etched into the source/drain (S/D) areas [77]. Lattice mismatch between Si and SiGe creates a large compressive stress in the PMOS channel, resulting in significant hole mobility improvement.

Dual-stress Nitride Liners: As shown in Figure 3.2, mechanical stress can also be transferred to the channel through the active area and polysilicon gate by depositing a permanent stressed liner over the device [78]. Tensile liners improve electron mobility in NMOS devices, while compressive liners improve hole mobility in PMOS devices. The latest high performance process nodes have simultaneously incorporated both tensile and

compressive stressed liners into a single, high performance CMOS flow, called the Dual-Stress Liner technique. In this process, a highly tensile Si3N4 liner is uniformly deposited over the entire wafer. The film is then patterned and etched from the PMOS regions. Next, a highly compressive Si3N4 liner is deposited, patterned and etched from the NMOS regions.

Stress Memorization Technique (SMT): In addition to the permanent tensile liner shown in Figure 3.2, the Stress Memorization Technique (SMT) is also used to increase the stress in n-type MOSFETs [79]. In this technique, a stressed dielectric layer is deposited over all of the NMOS regions, thermally annealed, and then completely removed. The stress effect is transferred from the dielectric layer to the channel during the anneal and is "memorized" during the re-crystallization of the active area and gate polysilicon.

A closer examination of these stress sources shows that the amount of stress transferred to the channel, and, consequently, the drive current enhancement, has a strong dependence on certain layout properties. The amount of eSiGe (and, hence, the stress), for example, depends upon the length of the active area. Longer active area also means that the STI will be pushed further away from the channel, which will lower its effect on the total channel stress. Therefore, the drive current of a transistor depends not only upon the gate length and width (L and W), but also upon the exact layout of the individual transistor and its neighboring transistors. This means that the performance of two transistors with identical gate lengths and widths can actually differ significantly, depending on their layouts.

Beginning in Section 3.2, we study the layout dependence of stress-based performance enhancement for different device configurations and identify simple layout properties in our 65nm process that allow us to maximize the performance gains due to stress. The idea is to determine the key layout parameters that a layout designer can change to affect transistor performance. Since we are interested in optimizing the layout, uniform techniques such as SMT can be ignored because SMT involves a uniform film deposition, anneal and removal over all of the NMOS regions, which leads to a uniform shift in NMOS drive current that is relatively independent of layout [80].

3.1.2 Drain Current Dependence on Stress and V_{th}

Modifying carrier mobility directly affects the amount of current that flows between the source and drain terminals of a transistor. Increased carrier mobility increases the drain current, I_D, in all regimes of MOSFET operation, which improves transistor performance (in terms of delay) but increases leakage power. In order to study the delay-versus-leakage tradeoffs involved in stress enhancement, we examine the saturation and subthreshold current equations in order to determine their dependency on carrier mobility. This also allows us to compare mobility enhancement to other performance enhancement techniques, such as V_{th} reduction. Equations (1) and (2) below give the expressions for drain current when the transistor is operating in the saturation and subthreshold regimes, respectively [28, 29].

$$I_{D,sat} = \frac{\mu_0}{[1 + U_0(V_{GS} - V_{th})]} \frac{C_{ox}}{2aV} \frac{W}{L_{eff}} (V_{GS} - V_{th})^2$$
$$V = \frac{1 + v_c + \sqrt{1 + 2v_c}}{2} \quad v_c = U_1((V_{GS} - V_{th})/a) \tag{1}$$

$$I_{D,sub} = A \cdot e^{\frac{1}{\eta v_T}\cdot(V_G - V_S - V_{th0} - \gamma' V_S + \eta V_{DS})} \cdot \left(1 - e^{(-V_{DS})/v_T}\right)$$
$$A = \mu_o C_{ox} \frac{W}{L_{eff}} v_T^2 e^{1.8} e^{\frac{\Delta V_{th}}{\eta v_T}} \quad (2)$$

From (1) and (2), it is evident that the saturation drain current ($I_{D,sat}$) has a sub-linear dependence on mobility, μ_0 (due to the vertical field mobility degradation coefficient, U0) while the subthreshold drain current ($I_{D,sub}$) dependence on μ_0 is linear. The drain current dependence on V_{th}, however, is almost linear in saturation, but is exponential in the subthreshold regime. Therefore, if we obtain identical saturation current improvement using two separate enhancement techniques: 1) stress-based mobility enhancement, and 2) V_{th} reduction, then the corresponding increase in leakage current for the reduced-V_{th} case will be much higher (due to the exponential dependence of $I_{D,sub}$ on V_{th}). Consequently, the reduced increase in leakage current makes mobility enhancement a more attractive option than its V_{th} counterpart.

The benefits of using mobility enhancement over V_{th} reduction is illustrated in Figure 3.3, which shows the normalized I_{on} versus I_{off} curves for stress-based and V_{th}-

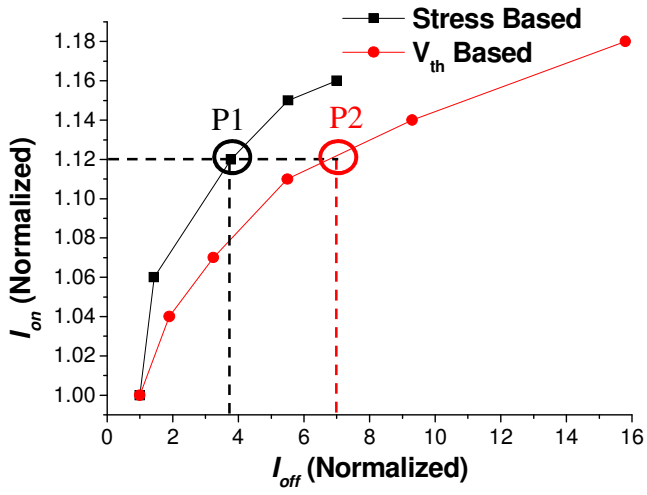


Figure 3.3 I_{on} vs. I_{off} for V_{th} & stress-based enhancement in a 65nm PMOS device.

based performance enhancement for an isolated, 65nm PMOS device. The device has three sources of stress: STI, acompressive nitride liner, and eSiGe source/drain regions. Stress is varied by changing the active area length, while the n-channel doping is changed to vary V_{th}. The curves clearly show that the tradeoff is better for stress variation. For a 12% improvement in I_{on}, the leakage for the V_{th} case is nearly twice as large as that for the stress-based improvement (shown in Figure 3.3 as points P1 and P2), and the difference is only amplified for higher values of improvement. Also, stress-based improvement allows for more fine-grain improvement control than V_{th} assignment, given that only two or three V_{th} values are typically allowed. Therefore, a designer would prefer to achieve performance improvements through stress-enhancement whenever possible, due to the reduced leakage penalty and increased granularity. The superiority of the stress-based performance improvement technique makes it an appealing option for further investigation.

Thus, the next two sections study the layout dependence of stress, and identify the primary layout properties that can be modified so that stress-induced enhancements are maximized.

3.2 Layout Dependence of Stress-based Enhancement

In order to study the layout dependence of stress-based performance enhancement, we used the DaVinci 3D TCAD tool [81], which has an extensive set of stress-related features. Additionally, we followed the layout rules from an industrial 65nm CMOS technology and the device fabrication was simulated in Tsuprem4 [82] (in order to capture the process-induced stress). The stress values were then imported into DaVinci, which simulated the device and solved for the stress-based mobility enhancement equations. The resulting values for drive current and leakage were verified

against experimental test chip data, which was consistent with previously published 65nm technology data for minimum sized NMOS and PMOS devices [75]. Furthermore, the simulated values of stress were in close agreement with previously reported data for PMOS channel stress while considering all of the layout dependent sources of stress [77]. Due to the absence of any previously published data on the layout dependence of stress or drive-current (due to stress), measured test chip results were used to quantify the impact of layout diversity on device performance. The fabrication process used for this test-chip employs all the known stress enhancement techniques. The hardware data was used to verify the accuracy of our TCAD setup, and the TCAD-based simulation results were found to be in close agreement with the measured data. Our consistency with these fabricated measurements can be attributed to the fact that we model all of the layout dependent sources of stress in the industrial 65nm technology. For a PMOS device, the sources of stress that are layout dependent include the compressive nitride liner, eSiGe, and STI. The NMOS sources, on the other hand, only include the tensile nitride liner and STI. We have ignored the Stress Memorization Technique (SMT) in our simulations, since it involves a uniform deposition and eventual removal of a dielectric layer over all NMOS devices (as discussed previously in Section 1.1). SMT, therefore, does not depend on layout properties and can be accurately treated as a uniform increase in NMOS drive current, independent of layout [80].

Previously, Figure 3.2 showed the 3D cross-section of an isolated PMOS device surrounded by STI. For the device shown, we increase the active area length ($L_{S/D}$) and examine the corresponding changes in drive current.[1] Increasing active area length has a number of effects: 1) it increases the amount of eSiGe, causing more stress to be

transferred to the channel; 2) it increases the distance between the channel and the STI, decreasing the effect STI has on channel stress; and 3) it allows more nitride over the active area. The nitride layer actually transfers stress in two ways – vertically through the gate and longitudinally through the active area. Since active contacts create openings in the nitride layer, the longitudinal component of nitride stress can be increased by moving the contacts away from the channel. Similarly, a source/drain region that does not have any contacts (or has a smaller number of contacts) will have higher channel stress than one that has a high contact density.

Figure 3.4a shows the longitudinal stress (S_{xx}) in the same isolated PMOS device for two normalized $L_{S/D}$ values of 1 and 1.58 (the values are normalized to the length of a minimum-sized, contacted S/D region). Figure 3.5 shows the PMOS drive current, I_{on}, and leakage current, I_{off}, plotted against $L_{S/D}$, while Figure 3.6 shows the normalized PMOS longitudinal stress plotted against $L_{S/D}$. Results show that for a 12% performance increase, leakage current only increases by 3.78X. This I_{on} versus I_{off} tradeoff is much better than the tradeoff produced by the alternative, V_{th}-based enhancement technique, as predicted in Section II-B. Additionally, Figure 3.5 shows the saturation point for extending $L_{S/D}$. Increasing the S/D length beyond 1.58 (normalized) yields minimal performance gains, even when active area length and leakage current are increased substantially. Finally, the performance enhancement is also sensitive to contact placement. Moving the contacts away from the channel accounts for nearly 2.6% of the drive current improvement and a device with a non-contacted drain (typically seen in

1. The authors would like to note that in this work, $L_{S/D}$ is equivalent to both the $L_{S/D}$ and $L_{p/p}$ used in previous works (such as[18]). Thus, in the remainder of the chapter, $L_{S/D}$ can refer to any longitudinal S/D dimension.

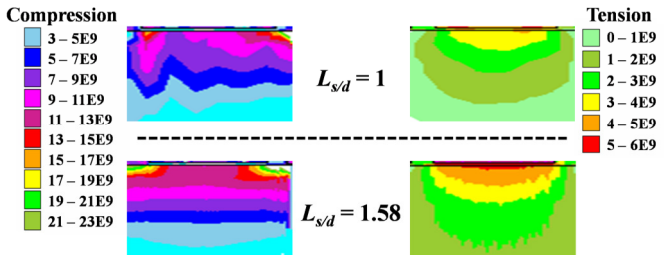


Figure 3.4 Longitudinal stress component S_{xx} (in Pascals) for normalized $L_{S/D}$ of 1 and 1.58 for (a) PMOS (b) NMOS.

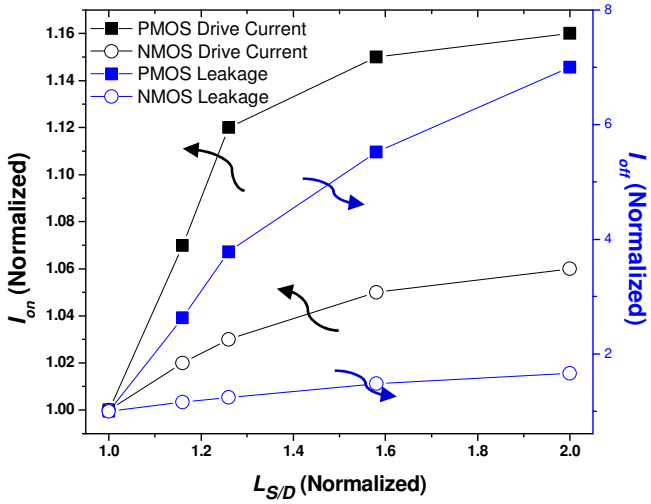


Figure 3.5 I_{off} and I_{on} vs. $L_{S/D}$ curves for stress-based performance enhancement in isolated PMOS and NMOS devices.

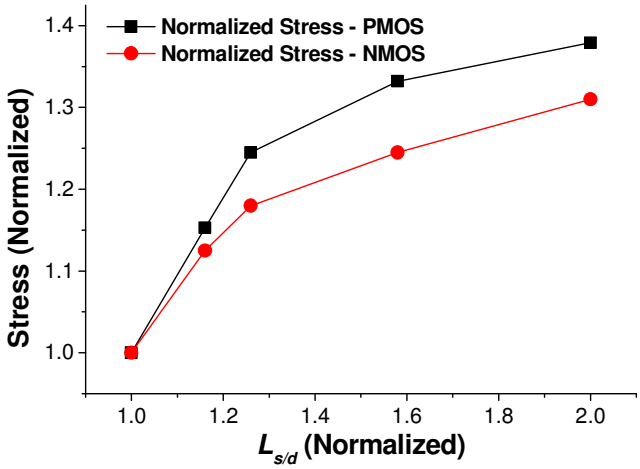


Figure 3.6 Longitudinal Stress vs. $L_{S/D}$ for isolated PMOS and NMOS devices.

series devices) has ~4% higher performance.

Unlike its PMOS counterpart, NMOS device performance is actually degraded by STI since STI induces compressive stress in the channel. Thus, increasing NMOS $L_{S/D}$ not only pushes away the compressive STI, but it also allows for more contact separation from the channel. Figure 3.4b shows the longitudinal stress in an isolated NMOS device for normalized $L_{S/D}$ values of 1 and 1.58. In addition to PMOS I_{on} and I_{off}, Figure 3.5 also shows NMOS I_{on} and I_{off} while Figure 3.6 shows its normalized longitudinal stress versus $L_{S/D}$. For NMOS devices, a 5% performance gain can be achieved for a 1.48X increase in leakage current. NMOS devices also have the same (normalized) upperbound for $L_{S/D}$ extension as their PMOS counterparts, 1.58. Beyond this value, the area and leakage current penalties do not warrant the minimal gains in I_{on}. The increase in performance in NMOS devices, however, is limited by the fact that we are only increasing the nitride's longitudinal stress through the active area (about 35% of the total stress due to the nitride liner), and pushing away the STI (which has a relatively smaller contribution to the overall channel stress). Experimental results show that almost 80% of the total NMOS improvement is due to moving the contacts and a device with a non-contacted drain has ~2% higher performance.

Next, we studied transistor performance in denser layouts. Figure 3.7 shows the channel stress and the corresponding layout view for three PMOS transistors in a 3-input NAND gate. The device in the center (device 2) has higher stress than the two corner transistors because it is surrounded by more eSiGe (its own S/D regions as well as its neighbors' S/D regions). This difference in stress is reflected in their drive current performance, and simulations show that the drive currents for the center and edge devices

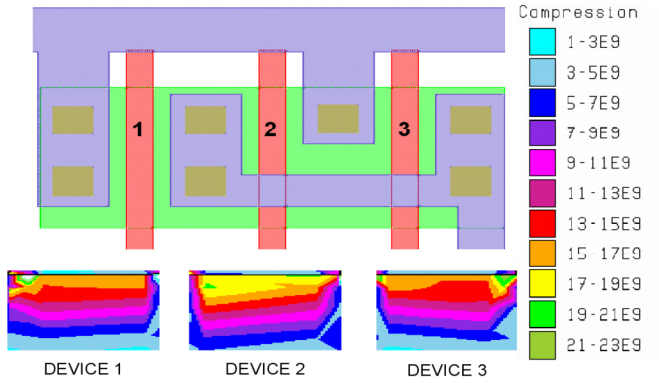


Figure 3.7 PMOS devices for a 3-input NAND gate and the corresponding channel stress distribution (in Pa).

differ by 8.2%. Furthermore, if there were five devices side-by-side instead of three, the difference would increase to 14.8%. This means that the drive current of a transistor is not only layout-dependent, but it is also location-dependent. Similar experiments for NMOS devices show differences of 7.4% and 12.2% for the case of three and five side-by-side transistors, respectively.

3.3 Layout Properties that Impact Mechanical Stress and Performance

Based on the intuition developed in the previous section, we now identify 3 simple layout properties in our 65nm technology that can be used to optimize a given layout for stress-induced performance enhancement. Once the properties are presented, the end of this section discusses one other important stress effect: the position-dependency of stress-induced performance enhancement. When mechanical stress is present in MOSFETs, matching W and L does not guarantee similar transistor performance even when neglecting process variation. Apart from W and L, the drive current is also affected by the layout parameters that influence stress: active area length, placement and number of contacts, and device context (i.e., whether the device is surrounded by other transistors or isolated by STI on one or both sides). In this chapter,

we have already discussed the first two parameters in great detail, while the third parameter (device context) has only been briefly mentioned (at the end of Section 3,2). However, since the device context or position of a transistor within a layout also affects performance, it must be accounted for by the designer, so this phenomenon is discussed in more detail at the end of the section.

Upon finishing the layout dependency study in Section 3.2, we determined that in our 65nm industrial process, the following 3 properties had the largest impact on improving performance (without modifying existing cell boundaries).

Layout Property #1: Active Area or Source/Drain Lengths

Using the length of a transistor's source or drain regions (or, equivalently, changing the amount of active/diffusion area) to modify stress-enhancement is well known technique and has been studied in a number of works [69,72-74]. Increasing the active area moves the STI regions away from the channels and increases the amount of eSiGe in PMOS devices. Moving the STI farther from the channel improves the performance of NMOS devices since STI exerts a compressive stress in the longitudinal direction, which degrades the NMOS electron mobility. For PMOS devices, on the other hand, compressive STI stress is actually beneficial and improves hole mobility. However, increasing the active area for PMOS devices still results in higher stress due to the relatively small contribution of STI compared to the other sources of stress. Measurements show that the stress due to STI represents <10% of the total channel stress. Therefore, the increase in eSiGe and its resulting contribution to PMOS channel stress dominates the stress due to STI and provides a significant increase in hole mobility.

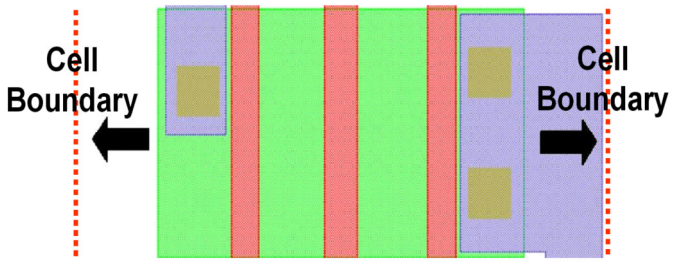


Figure 3.8 Application of Layout Property #1 to PMOS stack in 3-input NOR.

Increasing the active area can most readily be accomplished in a compact pull-up or pull-down network (often containing an NMOS or PMOS stack) that does not use the full width of a cell (Figure 3.8 shows the scope for increasing the active area of a PMOS stack in a 3-input NOR gate). In the case of stacked transistors, the layout does not require contacts between intermediate nodes. Thus, their spacing can be significantly tighter because nodes that contain contacts need larger spacing to satisfy the technology's design rules. In the absence of stressors, it is best to minimize the active area in order to reduce the capacitance. However, in the presence of stressors, increasing active area length also results in higher stress in the channel (and, hence, higher drive current), in addition to increasing the source/drain capacitances. In a given CMOS layout, increased S/D capacitance for transistors closer to the output will directly affect the output capacitance, while transistors closer to the V_{DD} and V_{SS} rails will have a smaller effect. Hence, this layout property should be increased in cells with larger output loads, so that the change in capacitance is a small fraction of the total output capacitance. The authors would like to note that the mechanical stress dependence on active area can also be exploited to create high performance versions of standard cells which incur some area penalty, but are assigned optimally within a design.

Layout Property #2: Contact Placement

Moving the contacts away from the channel allows more stress to be transferred by the nitride layer. For isolated devices, pulling the contacts as far away from the gate polysilicon as the design rules permit maximizes the stress-enhancement. Contacts between two gates, on the other hand, can either be placed midway for identical performance enhancement of both transistors, or placed closer to the non-critical transistor (increasing stress in the critical device). Moving the contacts away will also result in a small increase in the source/drain resistance, but, in our 65nm study, this increase was typically less than 5Ω (based on sheet resistance calculations for the maximum S/D displacement obtained while creating the stress-aware optimized library), and the resulting gain in drive current outweighed the increase. The maximum S/D contact displacement observed was 60nm.

Layout Property #3: Lateral Active Area Placement

From Figure 3.1, we know that the desired stress in the lateral direction is tensile for both NMOS and PMOS devices. Figure 3.9a shows the lateral stress behavior near the interface of the two nitride layers (cross-section across the poly going from PMOS to NMOS over STI). Figure 3.9b shows the plot of normalized lateral stress (normalized to the stress value at the point farthest from the nitride liner interface) at a depth of 1nm below the Si surface versus the distance from the tensile/compressive liner interface, under the tensile nitride layer. The behavior is interesting in the sense that there is a region of compressive stress under the tensile nitride (the NMOS side) and there is a region of tensile stress under the compressive nitride(the PMOS side). This behavior follows from the physics involved behind the stress-inducing process step. At the

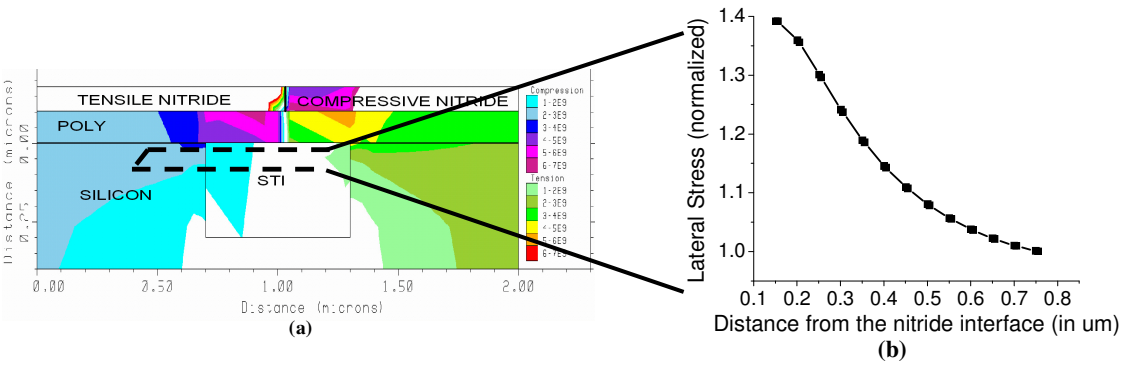


Figure 3.9 Stress (in Pascals) at nitride interface for NMOS and PMOS: (a) 2D view across lateral STI (b) Behavior under tensile nitride at channel depth.

compressive/tensile nitride liner interface, each nitride layer exerts an equal and opposite force on the other nitride layer, which imposes the opposite type of stress under the adjacent layer. Therefore, if possible, it is beneficial to move the PMOS active area into this region of tensile stress and the NMOS away from the region of compressive stress. The space for this movement is most readily available when the transistor widths are small but the cell pitch (lateral size) is large (due to pitch uniformity across standard cells). This combination of properties, for example, is common in minimum sized, simple gates (e.g., minimum size inverters, buffers, or 2-input NAND/NOR's).

It should be noted that the lateral active area placement will slightly alter the V_{th} of the shifted devices, due to well edge proximity effects [83]. However, since the amount of lateral shift applied to the 65nm standard cells was <0.205mm for the NMOS cells and <0.12mm for the PMOS cells, the corresponding shift in V_{th} was found to be <0.32mV (in both HSPICE and TCAD simulations, independently) for all devices. Since this V_{th}

shift is relatively small, the reported results described in the remainder of the chapter do not include the well edge proximity change induced by Layout Property #3. However, if this shift in threshold voltage becomes appreciable in future processes, our experimental setup can easily be modified to include a well edge proximity model, such as the ones described in [84], which will capture the corresponding change in V_{th}.[2]

Apart from these three layout properties, a designer must also be aware of how the channel stress is affected by the position of a device within the layout. Stress in the channel of a device depends not only upon its S/D lengths and contact placement, but also upon its surroundings. As we have shown in the previous section, devices that share their source/drain regions with other transistors have significantly higher stress (and hence drive current enhancement) than those at the edges of an active region (which are therefore bordered by STI), even for identical $L_{S/D}$ and contact placement. This difference in stress can be attributed to the effects of STI, as well as the fact that stressors for a device also affect its neighbors.

Ignoring the position-dependence of stress could lead to a number of design issues. First of all, the location of a transistor could result in an unexpected increase in drive current, resulting in smaller delay and possible hold-time violations, as some gates might be faster than expected. Secondly, the position-dependent current offset could modify the noise margins of a circuit. Hence, for circuits that are sensitive to noise margins (e.g., SRAM cells, Sense Amplifiers, etc.), these deviations must be accounted for either during the design phase (for example, by guardbanding against position-dependent offsets), or during the layout phase (e.g., by modifying the $L_{S/D}$'s to cancel the offsets). Finally, in certain circuits, if the strength of a transistor (in terms of drive

current) is increased beyond the expected value, it could cause a substantial drop in performance. A detailed example of context-sensitive design is included in Section V. All in all, designers need to be aware of the effect that position has on performance, especially if pin-to-pin delay, noise margins, or transistor strength are essential to a particular design.

There are three main ways that a designer could capture the position dependence of stress within a particular design: fabrication, TCAD simulation, and electrical circuit simulation. The first solution, fabrication, is an expensive and time consuming endeavor, especially during the early stages of a process's lifetime. The second alternative – using TCAD tools to simulate the position dependence of stress – can be costly in terms of runtime, and convergence becomes extremely difficult when simulating more than 10 devices at once. The final solution, electrical circuit simulation (e.g., HSPICE simulation), promises to be the most efficient in terms of both cost and runtime. Unfortunately, to our knowledge, there has been little research dedicated towards electrical models that capture the layout dependence of stress. Furthermore, of the few that have been published (such as [71]), none have been implemented within an electrical circuit model (e.g., BSIM). The problems associated with each of these solutions make modeling the position dependence of stress an important and interesting research topic that remains largely unexplored.

3.4 Modifying 65nm Standard Cell Layouts

This section discusses the effectiveness of modifying the layout properties from

[2] HSPICE well-edge proximity was captured during Calibre PEX parasitic extraction, and then fed into our industrial BSIM models to calculate the effect on V_{th}. Note that the 0.32mV shift reported can be viewed as the shift in ΔV_{th} (the change in Vth due to well proximity), not total ΔV_{th} itself.

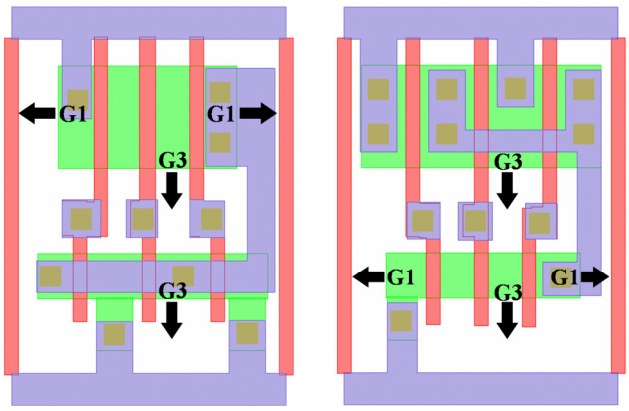


Figure 3.10 Two Layouts – (a) 3-input NOR gate and (b) 3-input NAND gate – showing the scope for layout-based stress improvement.

Section IV in standard cells from an industrial 65nm CMOS technology library. For a given layout, as shown in Section III, a basic tradeoff always exists between the source/drain length, $L_{S/D}$, and the improvement in drive current. By exploiting this tradeoff, we can make faster, but leakier, versions of the standard cells with varying area increments and assign them intelligently to the critical paths in order to optimize performance. The performance enhanced versions all use a combination of the three properties discussed in Section 3.3: increased $L_{S/D}$, larger poly-to-contact spacing, and stress-aware lateral placement.

For example, Figure 3.10a shows the layout for a 3-input NOR gate. It consists of three PMOS transistors in series (a 3-PMOS stack) and three NMOS transistors in parallel. This means that the source and drain of each NMOS is connected to the ground and the output, respectively, necessitating contacts at each node. The PMOS stack on the other hand, only needs one contact to V_{DD} (at the source of the leftmost PMOS) and one contact to the output (at the drain of the rightmost PMOS). Using the classical layout methodology (where stress is ignored and capacitance is minimized), we can shrink the non-contacted S/D regions to lower the parasitic PMOS capacitance. As shown in Figure

3.10a (labeled “G1”), the PMOS region has the capability of increasing the source/drain lengths (Layout Property #1) by ~22% without affecting the overall cell area. While increasing the source/drain lengths, we simultaneously shift the contacts away from the gates (Layout Property #2), maximizing performance enhancement. If we increase the active area uniformly for all transistors, drive current improves by ~12% for each PMOS device. Also, there is lateral room to move the NMOS and PMOS active area and exploit the stress dependence of Layout Property #3 (labeled “G3” in Figure 3.10a). This leads to further improvements of about 3% and 1.5% for NMOS and PMOS devices, respectively. Therefore, for the 3-input NOR gate, we observe overall improvements in drive current of ~13.5% for PMOS devices and ~3% for NMOS devices. Similarly, by modifying Layout Properties 1–3 in a 2-input NOR gate, we can achieve drive current improvements of 7.5% and 3% for the PMOS and NMOS devices, respectively.

Similarly, Figure 3.10b shows the layout for a 3-input NAND gate. Instead of a PMOS stack, there is an NMOS stack in the NAND gate, so there is a potential to increase the NMOS active area length without affecting the cell area. While altering Layout Properties 1 and 2, we obtain an improvement of ~4% for each of the NMOS drive currents. Also, there is space for moving the active areas to exploit the mobility dependence of Layout Property #3. This leads to further improvements in NMOS and PMOS devices of ~3% and ~1.5%, respectively. Overall, we can achieve a ~7% NMOS performance enhancement and a ~1.5% PMOS performance enhancement. Similarly, by modifying Layout Properties 1–3 of a 2-input NAND, we can obtain drive current improvements of 4.5% and 1.5% for the NMOS and the PMOS devices, respectively. Scope for such layout-based improvements is found in most of the standard cells in our

Table 3.1 Percentage contribution of layout properties 1–3 to the overall drive current improvement for pmos/nmos stacks.

	Property 1	Property 2	Property 3
NOR3 PMOS	69.60%	19.30%	11.10%
NAND3 NMOS	20.10%	37.80%	42.10%
NOR2 PMOS	53.30%	26.60%	20.10%
NAND2 NMOS	10.10%	27.20%	62.70%

Table 3.2 Summary of stress-aware layout optimization drive current improvement and tradeoffs in 65nm standard cells

Cell Name	Percentage drive current improvement by layout optimization		Increase in leakage current by layout optimization		Increase in leakage current for identical drive current improvement by V_{th} reduction		Percentage increase in output capacitance with a FO4 output loading
	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS	
3-input NOR	3%	13.50%	1.22X	4.02X	1.31X	9.20X	2.74%
2-input NOR	3%	7.50%	1.22X	2.24X	1.31X	3.52X	1.92%
3-input NAND	7%	1.50%	1.98X	1.10X	2.36X	1.53X	1.85%
2-input NAND	4.50%	1.50%	1.45X	1.10X	1.68X	1.53X	1.30%
Iso Area INV	3%	1.50%	1.21X	1.10X	1.31X	1.53X	0%
Incr. Area INV	6%	13%	1.86X	3.88X	2.22X	7.04X	2.40%

library.

Table 3.1 shows the percentage contribution of each layout property to the total drive current improvement achieved for PMOS and NMOS stacks in 2- and 3-input NOR and NAND gates, respectively. The relative contribution of the properties varies between the four cases. This is due to the presence of eSiGe in PMOS which is a major contributor to the overall stress in the channel. As a result, for PMOS devices, altering Layout Property #1 (increasing the active area) results in the maximum improvement as compared to the improvement achieved by modifying the other two properties. However, in the case of NMOS devices, increasing active area results in pushing away the STI, whose contribution to the overall channel stress is relatively smaller. Longitudinal stress

due to nitride is increased upon the alteration of Layout Property #2, and Layout Properties 2–3 are major contributors to drive current improvement in NMOS devices.

Table 3.2 summarizes the results of changing Layout Properties 1–3 in a few standard cells. It reports the percentage drive current improvement, leakage current increase, and the percentage increase in the output capacitance (assuming an FO4 output loading). It also reports the leakage current increase for identical drive current improvements through V_{th} reduction. Comparing the leakage current increase for stress-aware layout optimization to V_{th} reduction re-establishes the superiority of the stress-aware layout optimization. For a 3-input NOR gate, the PMOS leakage current increased by 4X when the layout was optimized to exploit stress dependencies, while the corresponding increase for the V_{th} reduction case was 9.2X. The increase in NMOS leakage for a 3-input NAND gate was found to be 2X for stress-based layout optimization, and 2.4X for the case of V_{th} reduction. Application of Layout Property #1 increased the S/D capacitance since $L_{S/D}$ was increased, but, as shown in Table II, this increase was very small (<3% if we assume an FO4 output loading).

In this same manner, we modified the layout properties from Section IV in ~25 standard cells in a 65nm industrial library, creating a stress-enhanced version of each cell. For the majority of standard cells, the stress-enhanced versions are the same area as the original cells, thus, there is no area penalty. However, since there are no series/stacked devices in inverter layouts, there is negligible space to modify Layout Property #1. The capacitance increase for the “Iso Area INV” is 0% as reported in Table 3.2, because there is only space for the application of Layout Property #3, which does not affect capacitance. Therefore, we decided to create a second, slightly larger, stress-enhanced

version of each inverter cell (with ~20% area increase per cell) that achieved larger drive currents (13% increase for PMOS and 6% increase for NMOS). Since the inverters, however, only make up a small subset of our standard cell library, the overall impact on circuit area is <0.5% (as shown later in Table IV). The final stress-enhanced standard cell library is comprised of different sized inverters (iso-area and increased-area versions) as well as 2- and 3-input NAND and NOR gates of varying strengths.

As mentioned in Section 3.2, the position of a device within a layout also affects its stress, and, therefore, its drive current. This position-dependent drive current enhancement can significantly hurt the performance of some circuits. This fact was verified using the circuit shown in Figure 3.11, which contains the schematic and partial layout of a basic domino implementation of a 2-input OR gate. Keeper device P2 is a weak PMOS that is used to hold the high state at node N during the evaluation period of the clock, so that N is not discharged by the NMOS leakage currents. The keeper, P2, should be sized large enough to replace the NMOS leakage current and sustain a high voltage at N, but, at the same time, it should be small enough so that the pull-down network can discharge N quickly to minimize the short-circuit current.

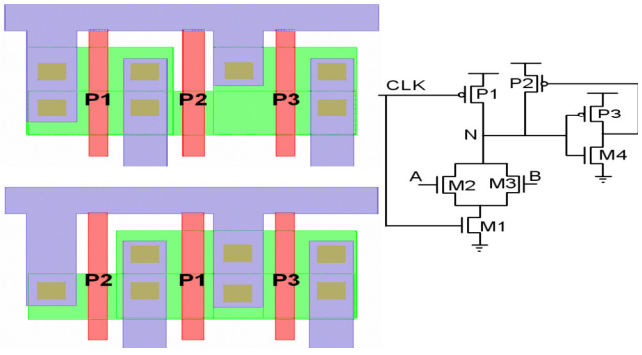


Figure 3.11 Basic Domino gate and two possible layouts for the PMOS devices.

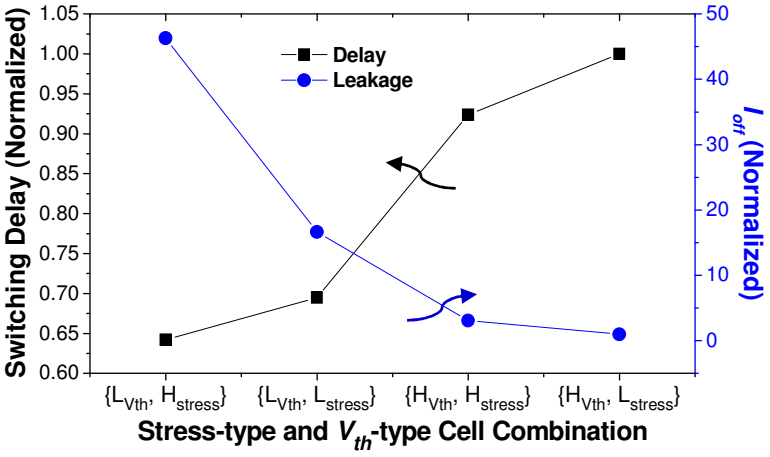


Figure 3.12 Leakage and switching delays for various combinations of V_{th} and stress-based optimization for 3-input NOR gate.

Figure 3.11 shows two possible layout scenarios for the three PMOS transistors. In one case P2 is located between P1 and P3, while in the other case P1 is in the middle. As shown in Section III, for the two scenarios the drive current for P2 differs by ~8%. This means that the first scenario has higher drive current for keeper P2 than the expected value. As the keeper fights against the pull-down stage, there is a performance loss. HSPICE simulations show that the time taken to discharge node N increases by ~12%. This performance loss can worsen for more aggressively sized cases. For these HSPICE simulations, we approximated the drive current increase due to stress by changing the relevant mobility numbers in the transistor models.

3.5 Optimization Methodology

Stress-based performance enhancement provides a better leakage versus performance tradeoff than V_{th} assignment (as discussed previously in Section 3.1.2). However, when the standard cell area is fixed (i.e., the stress-enhanced version occupies the same/slightly higher amount of area as the original version), we can only obtain limited average drive current improvement through stress-aware layout optimization

(<10%). Therefore, we combine stress-optimized assignment with dual-V_{th} assignment to simultaneously achieve a larger range of current improvement and more fine-grained control over the performance enhancement (and, consequently, the increase in leakage). Figure 3.12 shows the leakage and switching delays for various combinations of V_{th} and stress-based optimization for a 3-input NOR gate. Low stress (L_{stress}) optimization corresponds to a standard cell in the library that has not been optimized for stress enhancement (by altering the layout properties), while high stress (H_{stress}) optimization corresponds to the layout optimized version of the standard cell. For the dual-V_{th} approach, a gate has only two options to choose from, high-V_{th} (HV_{th}) or low-V_{th} (LV_{th}). Introducing stress-based, layout-optimized cells provides an additional reduced leakage option (when performed on a high-V_{th} cell) for gates that require moderate improvements in performance, thereby saving leakage power. Additionally, it also provides a higher performance option when combined with low-V_{th} to further reduce delay.

For simultaneous V_{th}/stress optimization level selection and sizing optimization, we use an iterative approach similar to [62] that can be divided into two main parts:

1. A certain number of gates in each iteration are assigned to the low-V_{th} or high stress optimization level.

2. The circuit is then rebalanced by reducing the size of the affected gates and other gates are re-sized to compensate for the area reduction (the objective is iso-area).

Initially, all gates are set to their {HV_{th},L_{stress}} version, to maximize leakage savings. Then, in each iteration, a merit function is evaluated for all gates in a circuit. This merit function rates the increase in total leakage with respect to the performance

gain of the circuit. Gates with the highest merit are selected first and set to the next highest performance level. The performance levels for our library are shown in the x-axis of Figure 3.12, and, from left to right, are ordered from highest performance (and leakage) to lowest performance (and leakage). This order holds for all standard cells in our library. The merit function is shown below in (3):

$$\text{Merit}(G) = \frac{\Delta I_{off}(G)}{\Delta D(G)}$$
$$\text{where } \Delta D(G) = \sum_{arcs}^{\alpha} \Delta d_{\alpha}(G) \cdot \frac{1}{k + Slack_{min} - Slack_{\alpha}} \quad (3)$$

Here, $\Delta d_{\alpha}(G)$ is the impact that increased gate performance has on a particular timing arc, α; k is a small negative number; and $Slack_{min}$ is the worst slack seen in the circuit. This weighting function takes the value 1/k for timing arcs on the critical paths, and approaches zero for less critical timing arcs.

Once the merit function is evaluated, a circuit's gate sizes are no longer optimal since one or more gates have been assigned to a higher performance level. The resulting decrease in delay creates excess area which can be recovered from the now oversized gates. By shifting this excess area to undersized regions, we can improve performance without increasing area (or only increasing it by a small amount). The candidates for reduction include the modified gate itself along with any gates sharing a timing path with the modified gate. Because modifying a gate has a greater effect on nearby gates, we can identify a modified gate's core of influence to a predetermined logic depth based on the distance of gates (sharing a timing arc with the modified gate) from the changed gate. This depth was experimentally determined to be three levels of logic [62]. For the purpose of resizing, we use a delay-sensitivity-based sizing optimization algorithm [85].

The pseudo code for a given value of target critical delay (T_T) is shown below. Note that Lines 2 and 3 merely provide one set of initial values for T_C and T_N such that the conditions of the while loop are satisfied in the first iteration.

Algorithm 1 STRESS_OPT(T_T) // T_T = Target Delay

```
1:  Set all cells in netlist to {H_Vth,L_stress} version
2:  Run Initial STA and baseline sizing
3:  T_N = T_T + 1 // T_N = new critical path (CP) delay
4:  T_C = T_N + γ + 1 // T_C = current CP delay
5:  // γ = small constant, checks for >minimal changes in TC
6:  while ( (T_N > T_T) and ((T_C - T_N) > γ) )
7:        T_C = T_N
8:        Evaluate Merit(G) for all gates, G // see (3)
9:        Move gates with highest Merit(G) to next highest
      performance level
10:   Rebalance circuit through sizing
11:   Update STA, find new critical delay, T_N
12:  end while
```

The next section discusses the experimental results obtained when applying this optimization algorithm to benchmark circuits.

3.6 Experimental Setup and Results

The following section describes the library characterization used within our experimental setup, as well as the results obtained from using the proposed optimization scheme on a number of benchmark circuits.

3.6.1 Library Characterization

To implement our optimization methodology, we first had to characterize our stress-enhanced standard cell library and determine the decrease/increase in propagation-delay/leakage- power, respectively, that the standard cells achieved while exploiting the layout dependencies of stress. The characterization flow is illustrated in Figure 3.13 and captures the relative change in propagation delay and leakage power, as compared to the "unstressed" version of a particular standard cell. While characterizing one standard cell,

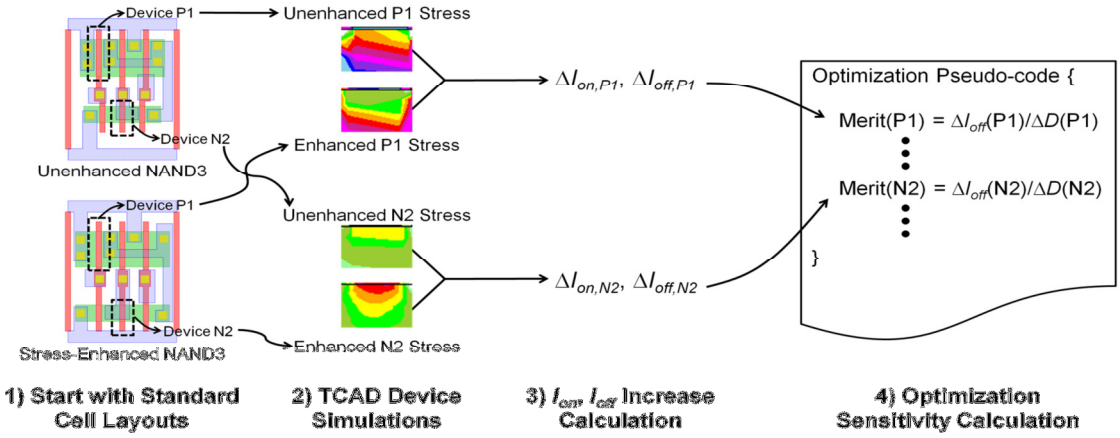


Figure 3.13 Stress-enhanced library characterization for stress-aware optimization.

we simulated both the stress-enhanced version and its unstressed counterpart in Tsuprem4 and DaVinci, as discussed in Section 3.2. From these simulations, we were able to calculate the relative increase in I_{on} and I_{off} (referred to as $\Delta I_{on}(X)$ and $\Delta I_{off}(X)$, respectively) for each device, X, within the standard cell. These $\Delta I_{on}(X)$ and $\Delta I_{off}(X)$ values for every PMOS and NMOS device (in every standard cell in our library) were then input directly into the optimization engine. Within the optimization algorithm, $\Delta I_{on}(X)$ is translated to decreasing propagation delay by using an inverse relationship fit:$\Delta d_{\alpha}(X) \propto \frac{1}{\Delta I_{on}(X)}$. Finally, these values, $\Delta d_{\alpha}(X)$ and $\Delta I_{off}(X)$, are used directly in the merit function described in (3).

In order to examine the effect that neighboring cells had on the channel stress of a device, we conducted a simple experiment where the value of I_{on} for a minimum-sized inverter in isolation was compared to the same minimum-sized inverter which had inverters as neighbors on both sides (representing a more "dense" context). We chose the min-sized inverter because of all of the standard cells, it was the most sensitive to changes in context. For the stress-enhanced inverter cell, we observed a 0.8% higher I_{on}

and a 2.0% higher I_{off} in the case where neighboring cells were included. However, the corresponding gains in I_{on} and I_{off} (ΔI_{on} and ΔI_{off}) for the stress-enhanced version (compared to the unoptimized version) decreased by <0.1% and <1%, respectively, while considering neighbors. Since the I_{on}/I_{off} gains achieved for stress-enhanced layouts showed little sensitivity to changes in context and because circuit level TCAD simulations were not possible (due to runtime and convergence issues), we used the library characterization of isolated cells to drive the circuit-level analysis in this chapter. In the proposed circuit-level optimization (discussed in Section 3.3), critical cells are iteratively exchanged with their stress-enhanced (or dual-V_{th}) counterparts. While considering the optimization of one particular cell within one iteration, only the type of enhancement is modified. All other parameters like neighborhood, size, and cell type (NAND, NOR, etc.) are held constant

3.6.2 Experimental Results

The algorithm described in Section 3.5 was implemented in C and tested on ISCAS85 benchmark circuits, two DSP circuit implementations ("Viterbi1" and "Viterbi2"), and a USB 2.0 controller implementation. The benchmarks vary in size from 166 to 37560 gates. The circuits were synthesized using an industrial 65nm CMOS technology with the following specifications:

- $V_{DD,nominal}$ = 1V
- HVT, NMOS V_{th} = 334mV
- HVT, PMOS V_{th} = -391mV
- LVT, NMOS V_{th} = 243mV
- LVT, PMOS V_{th} = -280mV

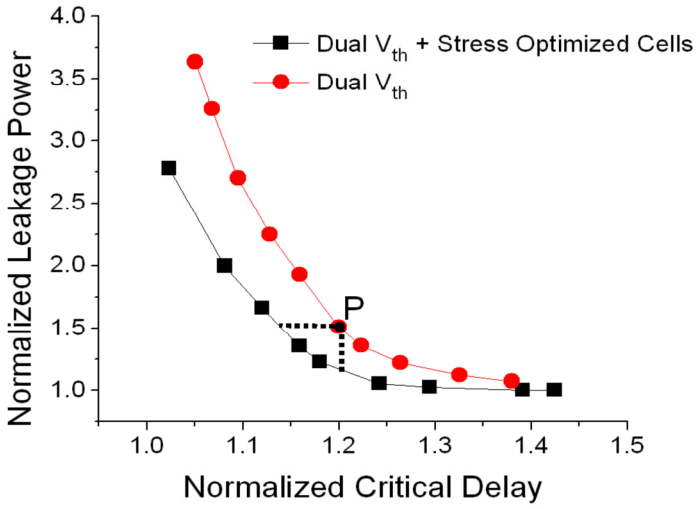


Figure 3.14 Leakage power versus delay tradeoff curve for the circuit c7552 for dual-V_{th} and proposed approach.

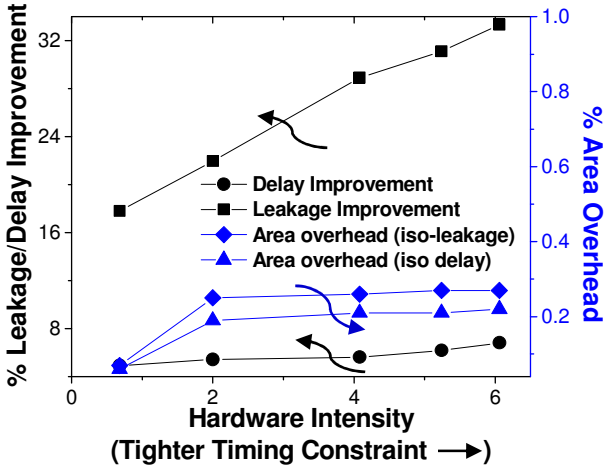


Figure 3.15 Delay and power improvement and the corresponding area overhead plotted against hardware intensity.

Table 3.3 Stress and V_{th} Combinations

	Cell Combinations
(1) Combined stress-enhancement and dual-V_{th}	$\{L_{Vth}, H_{stress}\}$, $\{L_{Vth}, L_{stress}\}$, $\{H_{Vth}, H_{stress}\}$, $\{H_{Vth}, L_{stress}\}$
(2) Only dual-V_{th}	$\{L_{Vth}, L_{stress}\}$, $\{H_{Vth}, L_{stress}\}$
(3) Only stress-enhancement	$\{H_{Vth}, H_{stress}\}$, $\{H_{Vth}, L_{stress}\}$

The resulting spread in I_{on} and I_{off} (between HVT and LVT) was 1.24X/1.32X and 16X/29X, respectively, for NMOS/PMOS transistors. All of the standard cells (both the original and the stress-enhanced versions) in our library were characterized (using HSPICE) at both the high- and low-V_{th} values. The layout-dependent characteristics (e.g., rise/fall delay, rise/fall power, etc.) and parasitics (such as junction capacitance and S/D resistance) for each cell were captured during the HSPICE characterization. All of the improvements discussed in this section use a dual-V_{th} optimization (using simultaneous V_{th} selection and gate sizing) as the basis for comparison.

Figure 3.14 shows the leakage power versus critical delay curves for the two techniques: dual-V_{th} assignment and dual-V_{th} assignment combined with stress-aware layout optimization, for one of the larger circuits, c7552. As mentioned earlier, combining stress-based layout optimization with V_{th} assignment provides a better range and more fine-grained control of performance enhancement as compared to the dual-V_{th} based assignment (see Table 3.3 for the cell combinations used in each optimization scheme). This is clearly seen in Figure 3.14 while comparing both the critical delay for the two techniques at the same value of leakage (iso-leakage), as well as the leakage power at the same value of critical delay (iso-delay). The key metric that we use in our comparisons is known as hardware intensity (η), which was proposed in [86] for quantifying the tradeoff between power and delay of a design. A hardware intensity of x means that a 1% decrease in delay leads to an x% increase in power. The hardware intensity for the majority of blocks in a microprocessor design is between 2 and 3 [87]. Thus, for a fair evaluation of the proposed approach, we present results for points on the power-delay curve that correspond to a hardware intensity value between 2 and 3. One

such point is shown as "P" in the leakage-power-delay tradeoff curve ($\eta = 2$) in Figure 3.14. For the circuit, c7552, our proposed optimization results in 22% lower leakage power for iso-delay, and 5.4% lower delay for iso-leakage, when compared to dual-V_{th} based assignment at point P.

Figure 3.15 shows how the percentage improvement (of our combined method over dual-V_{th}) in leakage power and critical delay, as well as the corresponding area overhead varies with hardware intensity for c7552. Percentage improvement in leakage power increases with increasing hardware intensity because the leakage-power-delay curves for our approach and dual-V_{th} assignment move further apart as delay decreases (or hardware intensity increases). The improvement in critical delay also increases with increasing hardware intensity. The area overhead, however, shows an initial increase as more gates require higher performance, but then becomes fairly constant at higher values of hardware intensity. For the remainder of this section, we report power and delay improvement numbers for points on the leakage-power-delay curves that correspond to a hardware intensity of 2.

Table 3.4 summarizes the improvements seen in two comparisons: 1) combined stress-enhancement and dual-V_{th} (which uses the cell combinations shown in (1) in Table 3.3) versus only dual-V_{th} (see (2) in Table 3.3), and 2) stress-enhancement (see (3) in Table 3.3) versus only dual-V_{th}. The first two columns state the name of the test circuit and its size. The next four columns report the percentage improvement in leakage over the dual-V_{th} case and the corresponding area overhead for iso-delay (for both comparisons). The last four columns show the percentage improvement in critical delay and the corresponding area overhead for iso-leakage-power (for both comparisons). The small

Table 3.4 Improvement in leakage and delay as compared to dual-V_{th} based assignment.

Circuit	Number of gates	Comparison for iso-delay against only dual-Vth assignment				Comparison for iso-leakage against only dual-Vth assignment			
		Stress + V_{th} based assignment		Only Stress based assignment		Stress + V_{th} based assignment		Only Stress based assignment	
		Improvement In leakage	Area overhead	Improvement in leakage	Area overhead	Improvement in delay	Area overhead	Improvement in delay	Area overhead
c432	166	38.50%	0.30%	5.40%	0.50%	5.00%	0.50%	3.60%	0.60%
c499	962	20.40%	0.90%	5.10%	0.90%	4.60%	0.90%	3.40%	1.00%
c880	390	33.70%	0.10%	12%	0.20%	5.80%	0.30%	2.30%	0.30%
c1908	432	22.50%	0.60%	7.40%	0.70%	4.70%	0.90%	3.00%	0.90%
c2670	964	14.70%	0.10%	5.10%	0.20%	5.20%	0.30%	3.60%	0.30%
c3540	962	23.90%	0.20%	4.70%	0.30%	4.70%	0.30%	2.50%	0.30%
c5315	1750	22.90%	0.20%	4.90%	0.30%	4.90%	0.20%	2.60%	0.20%
c6288	2470	20.10%	0.90%	5.90%	0.90%	4.60%	0.90%	3.00%	0.90%
c7552	1993	22.00%	0.30%	4.80%	0.20%	5.40%	0.20%	3.10%	0.30%
Viterbi 1	14503	21.50%	0.30%	4.90%	0.40%	5.30%	0.30%	2.90%	0.50%
Viterbi 2	34082	22.60%	0.30%	5.10%	0.40%	5.20%	0.20%	2.70%	0.40%
USB	37560	22.40%	0.30%	5.20%	0.30%	5.20%	0.40%	2.80%	0.30%
Average		**23.80%**	**0.40%**	**5.90%**	**0.40%**	**5.10%**	**0.50%**	**3.00%**	**0.50%**

value of area overhead occurs because of the increased area variants of the layout-optimized inverter cells (mentioned in Section 3.4).

The results clearly show that our combined approach significantly improves the leakage power for iso-delay, and also improves critical delay for iso-leakage, when compared to dual-V_{th} based assignment. We get up to a 38.5% (23.8% on average) improvement in leakage for iso-delay, and up to a 5.8% (5.1% on average) improvement in delay for iso-leakage. The area overhead is very small for both the cases – less than 0.5% on average across all 12 circuits. It is worth noting that while our delay improvements are similar to those published in [74], our proposed technique provides the 5.1% delay improvement (on average) for iso-leakage.

As mentioned previously, Table 3.4 also includes a one-to-one comparison of stress-enhancement versus dual-V_{th}, where stress-enhancement achieves up to a 7.4% (5.9% on average) improvement in leakage for iso-delay, and up to a 3.6% (3% on average) improvement in delay for iso-leakage (compared to dual-V_{th}). The discrepancy between the leakage improvement of the combined approach (stress + dual-V_{th}) versus dual-V_{th} (23.8% on average) compared to only stress-enhancement versus dual-V_{th} (5.9% on average) arises because the point on the stress-enhancement leakage/delay curve where hardware intensity equals 2 ($\eta = 2$) occurs at a larger delay (e.g., a point to the right of P in Figure 3.14). This is explained by the fact that stress-enhancement alone can only achieve <1/2 of the performance enhancement of dual-V_{th}. Thus, the leakage comparison between stress-enhancement and dual-V_{th} occurs in the region of leakage-versus-delay where stress does not have as large of an advantage over dual-V_{th} (note the smaller gap between the two curves in Figure 3.14 as you move towards larger delays). However, at the new comparison point, for this framework and technology, stress-enhancement still outperforms dual-V_{th} both in leakage optimization as well as delay optimization. This is noteworthy because using stress-enhancement by itself eliminates the extra masks and processing steps required by dual-V_{th} designs, which reduces process complexity and cost. Furthermore, the stress-enhancement versus dual-V_{th} improvement numbers are limited by the fact that we require small or no area overhead for the redesigned standard cells. Using more advanced techniques, we could further improve the stress-enhanced tradeoff between area and performance, which will increase the performance gap between stress-enhancement and dual-V_{th}.

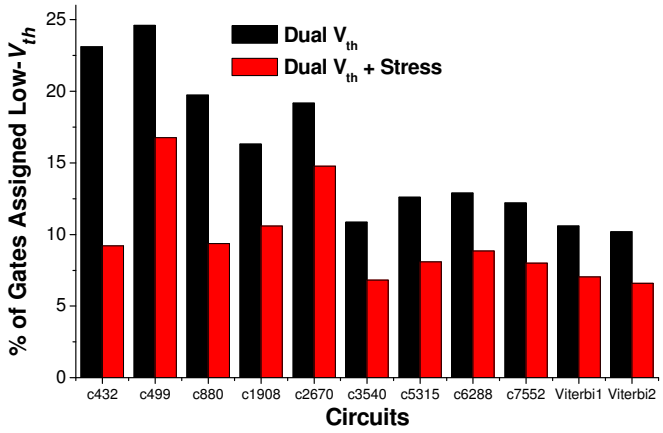


Figure 3.16 Percentage of gates assigned to low-V_{th} for dual-V_{th} and the combined dual-V_{th} and stress based approach.

Figure 3.16 shows the percentage of gates assigned to low-V_{th} for the dual-V_{th} assignment, as well as the combined "stress enhancement + dual-V_{th}" approach. These numbers are reported for iso-delay points on the leakage-delay curves corresponding to a hardware intensity of 2. As expected, for the combined approach, a lesser number of gates are assigned to low-V_{th} as compared to dual-V_{th} assignment. This is because for the dual-V_{th} assignment, not all gates assigned to low-V_{th} need such a large performance improvement. Combining stress-optimized cell assignment with dual-V_{th} assignment provides an additional lower leakage option for the cells that require moderate improvements. This reduces the number of cells that are assigned to low-V_{th}, which, in turn, results in lower leakage current. Typically, the number of gates assigned to low-V_{th} for the combined approach is ~35% lower than the number for dual-V_{th} assignment.

To further investigate the tradeoff that exists between leakage power savings and area overhead, we performed another experiment using a richer library comprised of higher area, stress-enhanced versions of all the cells. The area overhead for the higher area versions was ~20% per cell, and every cell in the richer library had three variants: an

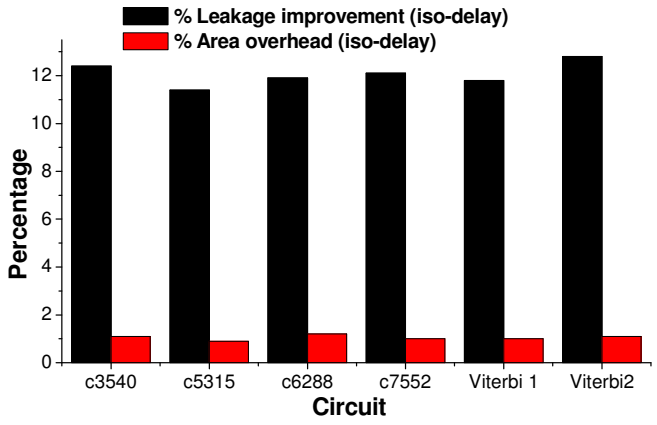


Figure 3.17 Delay and power improvement and the corresponding area overhead for the richer library over the original library.

original unoptimized version; an iso-area, stress-enhanced version; and an increased area, stress-enhanced version. The richer library provided more intermediate, low-leakage options (in addition to the low-V_{th} cell) for gates requiring moderate improvements. By providing these intermediate performance alternatives, the overall leakage power (for iso-delay) is further reduced as compared to dual-V_{th} assignment. Figure 3.17 shows the comparison between the "stress-enhancement + dual-V_{th} assignment" optimization for the richer library and the original, stress-optimized library (with increased area versions for inverters only). It plots the leakage power improvement (for iso-delay) and the corresponding area overhead obtained by using the richer library (compared to the original stress-enhanced library) for six of the larger circuits. On average, using the richer library further improved the leakage power (at iso-delay) by ~12% for an area overhead of ~1% over joint assignment using the original library. This experiment shows that there is scope for further improvement using the richer library. However, the richer library also incurs a higher characterization cost due to the large number of variants for each cell. One approach to minimize this cost would be to only create multiple versions of cells that

are used most often (typically the smaller gates such as inverters, NAND's, NOR's, etc.).

3.7 Summary

In this chapter, we explored the modification of standard cell layouts in order to optimize the stress-based performance enhancement, and proposed a block-based optimization algorithm that combined stress-enhancement with dual-V_{th} assignment to achieve performance gains in leakage or delay. We studied the dependence of drive current improvement on layout parameters like source/drain length and contact placement, and found that the performance of any given layout could be enhanced by increasing the active area length. Based on our observations, we exploited a set of layout properties which maximized the performance improvement of a standard cell without increasing area. When these properties were modified in standard cells from a 65nm industrial library, PMOS and NMOS drive currents attained an average performance enhancement of 6% and 4.4%, respectively, without increasing the cell area. The corresponding average increase in leakage was found to be 2.2X and 1.5X for PMOS and NMOS devices, respectively. Next, we combined the assignment of these stress-optimized cells with V_{th} assignment in order to optimally tradeoff leakage power and performance. When compared to the traditional dual-V_{th} based assignment technique, the new approach reduced leakage current by 23.8% on average for identical delay, and improved critical delay by 5.1% on average for identical leakage, with a very small area overhead (<0.5%).

Chapter 4
STEEL: A Technique for Stress-enhanced Standard Cell Library Design

As discussed in Chapter 3, three of the four main mechanical stress sources in today's processes – STI, nitride, and eSiGe – are all dependent on common layout parameters in modern standard cells. The two most dominant layout properties that affect mechanical stress and are customizable within standard cell design are source/drain (S/D) active area and contact placement. Larger S/D areas allow for greater amounts of eSiGe (in PMOS devices) and nitride (in both types of devices), which enhances mechanical stress in the channel. Contact placement, however, disrupts the continuity of the nitride layer and, consequently, lowers the contribution of the nitride layer to channel stress. Hence, contacts placed farther away from the channel will increase the amount of nitride adjacent to the channel, enhancing channel stress. Overall, the layout dependencies of stress are well documented [69, 71, 72], but little research has been dedicated to developing new standard cell library design techniques that exploit these dependencies.

Thus, in this chapter we propose a new standard cell design methodology that strives to fully exploit the layout dependencies of mechanical stress. Our library design methodology differs from previous mechanical stress work in that it employs a cell-level, library-wide enhancement technique that not only increases within-cell stress, but also increases cell-to-cell stress. Since most standard cells in a typical library have

source/drain VDD and VSS ties adjacent to one or both edges of the cell, our new, stress-enhanced libraries share these ties across cell placement and route boundaries as illustrated in Figure 4.1. By sharing the V_{DD} and V_{SS} nodes, stress is enhanced in both the edge devices as well as their neighbors, increasing Ion and Ioff by up to ~20% and ~3.5X, respectively for PMOS devices, and 7.5% and ~2X, respectively for NMOS devices.

The remainder of the chapter is organized as follows. Section 4.1 describes the technique used in our proposed standard cell design methodology. Section 4.2 describes our standard cell design and its ease of integration within state-of-the-art VLSI design flows. Finally, Section 4.3 discusses our results and Section 4.4 concludes the chapter with a brief summary.

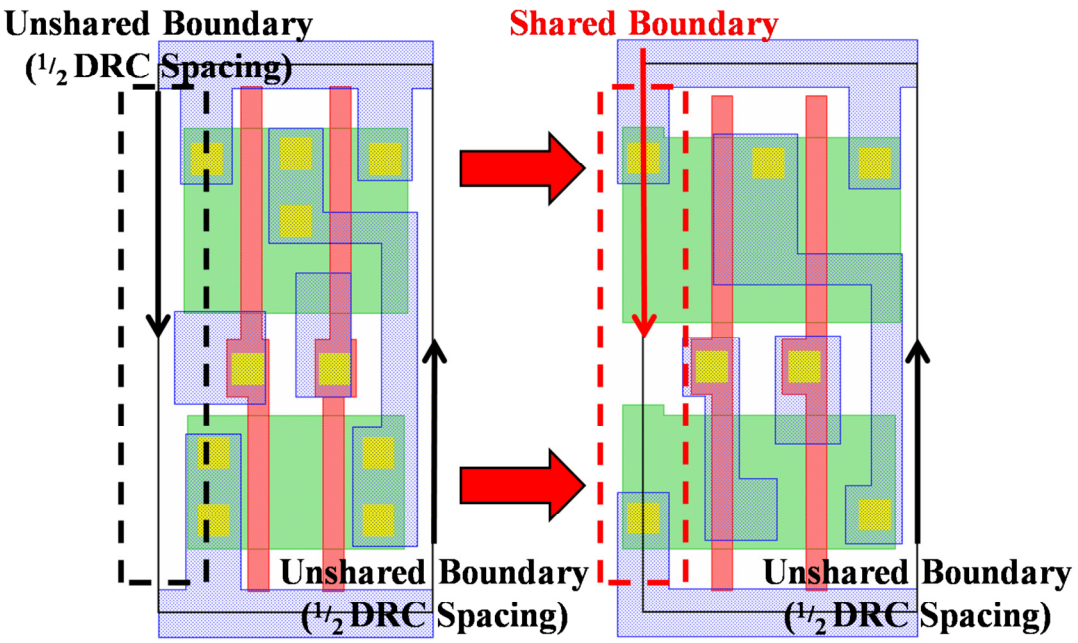


Figure 4.1 Traditional standard cell layout (a) versus proposed shared source/drain approach (b) for a 2-input NAND.

4.1 A Technique for Enhancing Stress in Standard Cell Layouts

As stated in Chapter 3, mechanical stress in MOSFET channels depends on a number of layout parameters. However, the amount of mechanical stress in a typical CMOS device is not only a function of its own layout parameters (S/D area, contact placement, etc.), but also of its neighbors' parameters. Thus, NMOS and PMOS devices that share their S/D regions with other transistors have significantly higher channel stress (and, hence, drive current enhancement) than those at the edges of an active region (which are therefore bordered by STI), even for identical active area length and contact placement. For NMOS devices, this is mainly due to the fact that STI has a negative impact on the amount of tensile stress induced in the longitudinal direction, resulting in lower values of tensile stress in edge devices compared to devices towards the center. For PMOS devices, stress due to STI enhances channel stress, however, since eSiGe has a much stronger contribution than STI, "center" PMOS devices also exhibit considerably higher channel stress as they are surrounded by more eSiGe. Therefore, in the presence of mechanical stress, two devices with identical layout parameters (W, L, $L_{s/d}$, contact placement, etc.) may differ significantly in drive current, depending upon their positions in the layout (even when neglecting process variation).

From a standard cell design perspective, one would ideally avoid these stress-based variations and move to a more uniformly stressed standard cell to minimize context dependencies and performance uncertainty. By sharing the V_{DD} and V_{SS} source/drain ties across standard cell boundaries, we can effectively increase the number of "center" devices (devices with at least one other transistor on both sides) in a given standard cell. This results in higher channel stress in the devices of such cells, since all of the affected devices will have more neighbors (which means more eSiGe, smaller STI regions, more

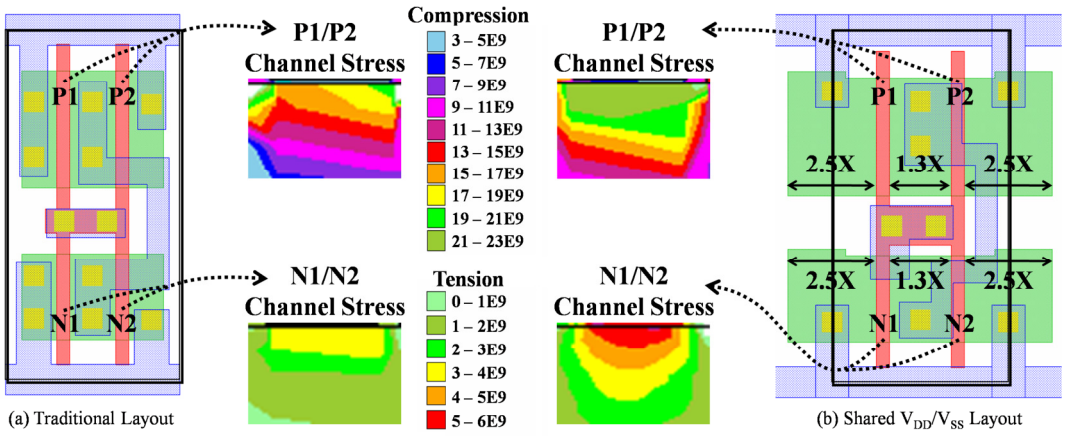


Figure 4.2 Traditional standard cell layout (a) versus proposed shared source/drain approach (b) for a 2-input NAND.

nitride, etc.). Figures 4.2 (a) and (b) illustrate our shared VDD and VSS source/drain connection technique (referred to as the STEEL – STrEss Enhanced Library – technique for the remainder of the chapter). Figure 4.2a depicts the traditional standard cell layout (for an inverter with two fingers) where the active area edge is placed at a location >=1/2 the design rule space from the standard cell boundary (the black rectangle that encapsulates the cell). However, since most standard cells in a typical library have at least one cell edge that is adjacent to a V_{DD} and V_{SS} S/D, we can share the connection between cells, effectively doubling the S/D active area and eliminating STI between the two cells. The edge devices achieve the largest increase using this approach – typically $L_{s/d}$ increases by >2X – and their induced channel stress now becomes more comparable to the stress in the "center" devices. Therefore, sharing the VDD and VSS connections between standard cells will not only lead to a more uniform distribution of channel stress, but will also improve the overall drive current of the standard cells (shown in the channel stress contour plots in the center of Figure 4.2). The actual "sharing" occurs in Figure 4.2b where the Metal-1 connections from VDD and VSS have been moved to the cell boundary. In this case, PMOS and NMOS drive currents increase by 13.5% and 6.3%,

respectively, while leakage current increases by 2.8X and 1.6X. Furthermore, one of the strengths of STEEL is that it achieves these improvements in stress uniformity and drive current with no cell area increase (i.e., the area encapsulated by the black place and route boundaries in Figure 4.2 is identical for both cells (a) and (b)).

4.2 Implementation of STEEL in Standard Cell Design

In order to develop a 65nm STEEL standard cell library that accurately captured stress effects and ensured compatibility within existing VLSI design tools (e.g., synthesis tools, place and route tools, etc.), we created a design flow which is described below and illustrated in Figure 4.3. This design flow is executed on a cell-by-cell basis, and begins by capturing the effects of stress for each device within a cell. We use Tsuprem4 to simulate the fabrication steps and Davinci 3D TCAD to capture the stress-enhanced device parameters. Then, we calibrate our TCAD model with an HSPICE model and extract the effects of stress into one device-specific multiplication factor: the low-field mobility multiplier ($\mu_{0,STRESS\_MULT}$). This modified HSPICE model is then used within Signalstorm (a library characterization tool) to calculate the propagation delays and

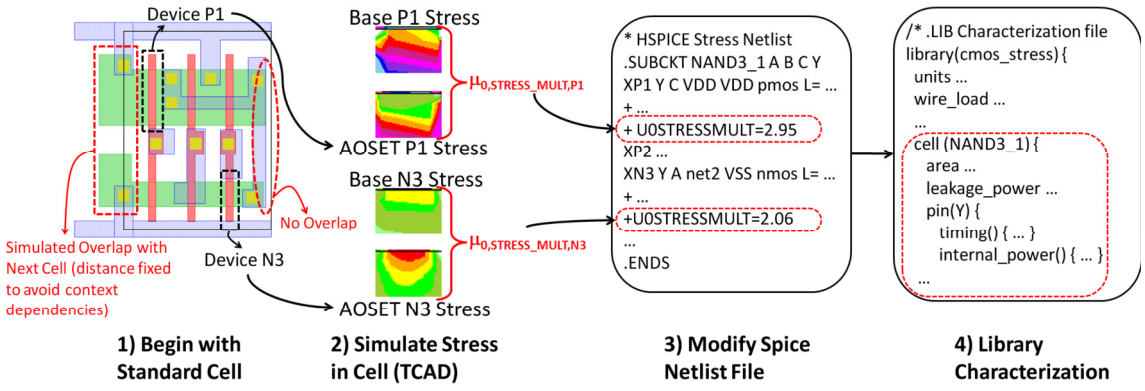


Figure 4.3 STEEL characterization flow.

power consumption for a given cell, which is eventually output in Synopsys's Liberty file format. This .LIB file can be used in a number of industry standard synthesis and/or automated place and route (APR) tools.

The remainder of this section describes the STEEL standard cell design flow in more detail and concludes by describing common issues encountered and how they were resolved. We implemented our design flow on a reduced set of the most commonly used standard cells – 33 standard cells in total.

4.2.1 Tsuprem4 and Davinci Device Simulation

Our design flow begins by using Tsuprem4 to simulate the fabrication of a particular device and capture the process-induced stress. Davinci 3D TCAD tool is then used to capture device behavior under stress by solving for stress-based mobility enhancement equations. We used a TCAD device simulator for this work because currently, to our knowledge, there are no industry-standard device models that capture all of the layout-dependent effects of stress. BSIM4 captures only the STI-related stress impact on effective mobility (μ_{eff}), saturation velocity (v_{sat}), and threshold voltage (V_{th}). However, Chapter 3 showed that other layout parameters also play a critical role in determining the amount of mechanical stress induced in a channel. Therefore, to capture these effects we simulate each standard cell in Tsuprem4 and Davinci, and extract the new, stress-enhanced low-field mobility (μ_0) at $V_{GS} = V_{DD} = 1V$ and $V_{DS} = 50mV$. By comparing a device's stress-enhanced mobility to its mobility without stress (the same TCAD simulation with the stress-analysis disabled), we can determine a device-specific scalar multiplier for μ_0: $\mu_{0,STRESS\_MULT}$. This multiplier is then used in our BSIM4 HSPICE model, described next.

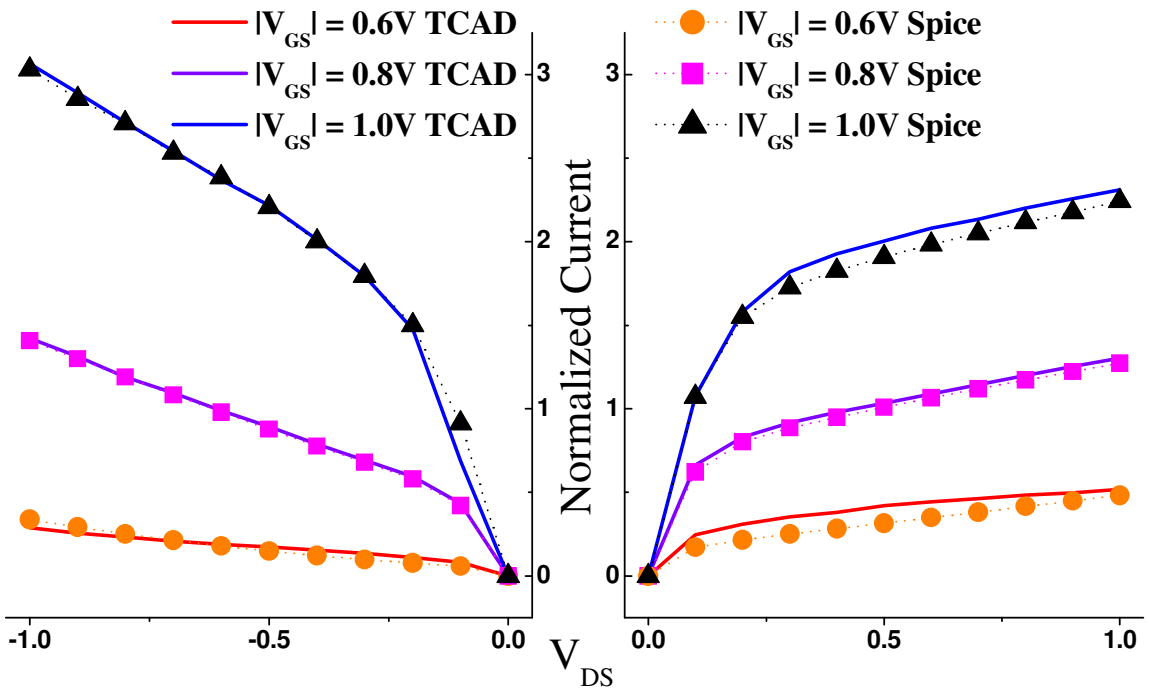


Figure 4.4(a) PMOS and (b) NMOS I-V plots: Davinci vs. HSPICE.

4.2.2 Stress-Enhanced BSIM4 HSPICE Model

After calibrating Davinci device simulations to 65nm industrial HSPICE models (by matching Ion and Ioff), we adjust the BSIM4 model so that the low-field mobility multiplier, $\mu_{0,STRESS\_MULT}$, is included as a possible input parameter for both PMOS and NMOS devices. We simply scale the old value of μ_0 by the multiplier: $\mu_0 = \mu_{0,OLD} \cdot \mu_{0,STRESS\_MULT}$. Simultaneously, since our Davinci models already capture all of the sources of mechanical stress, we temporarily turn off the BSIM4 stress models for μ_{eff}, v_{sat}, and V_{th} by setting the stress effect parameters for mobility degradation/enhancement (KU0), saturation velocity degradation/enhancement (KVSAT), and threshold voltage shift (KVTH0) to zero. The resulting I-V fit for minimum-sized NMOS and PMOS devices is shown in Figure 4.4, which verifies the accuracy of our model. For example, in these minimum-sized devices we find that our modified HSPICE device models incur an average root mean square error in saturation current of ~3mA and ~0.7mA for the NMOS

and PMOS devices, respectively. These HSPICE device models eventually serve as the basis of our standard cell library characterization.

4.2.3 Standard Cell Library Characterization

To make our new standard cell library compatible with existing digital, integrated circuit (IC) design flows, it is essential to be able to characterize the new standard cells and determine typical gate level parameters such as pin capacitance, propagation delay, dynamic and leakage power consumption, etc. To achieve this, we input our modified HSPICE models into Cadence's Signalstorm delay calculator. Signalstorm then simulates our stress-enhanced gates over a number of output-loading and input-slew combinations and finally generates a LIBERTY characterization file (.LIB). The .LIB file generation is the last step in the STEEL standard cell design flow and it enables the use of these new libraries within synthesis and APR tools with minimum additional overhead (described in more detail in Section 4.3.1).

4.2.4 Implementation Decisions in STEEL

There were several design decisions that needed to be resolved while creating a STEEL standard cell library. The first decision addressed the number of variants that could exist at an abutted boundary. These variants occur because many of the standard cells in a typical library cannot share the V_{DD} and V_{SS} connections at both edges of the cell. Instead, the adjacent S/D node is connected to some other net (e.g. the output node in a minimum-sized Inverter or NAND gate). For instance, refer to the 2-input NAND layout in Figure 4.1b. The NMOS drain on the right-hand side is tied to the output, Y. Therefore, this drain cannot be shared at the boundary with any arbitrary cell in a design whose left NMOS S/D is not connected to the same net. In this case, the PMOS source tied to VDD could be shared, but only with a cell that has the same configuration (shared

PMOS, unshared NMOS) or a custom "Filler" cell designed for the "shared PMOS, unshared NMOS" case. Therefore, to keep the number of edge variants small, we implemented two types of standard cell edges: shared or unshared. If either the NMOS or PMOS S/D is not connected to V_{SS}/V_{DD}, respectively, then that edge of the cell is designed to be completely unshared. STEEL consequently has three different types of cells:

- Cells with both edges "shared" (such as the one in Figure 4.2b).
- Cells with one "shared" edge and one "unshared" edge (previously discussed and illustrated in Figure 4.1b).
- Cells with both edges "unshared" (similar to the layout shown in Figure 4.1a).

Each standard cell in the library corresponds to only 1 of these 3 types, with the exception of inverters and buffers. To ease APR we designed two versions of inverter and buffer cells, one with the maximum number of shared connections and one with zero shared connections (both edges "unshared"). The "unshared" inverter and buffer cells reduce the placement/routing complexity involved during buffer insertion. For additional details of using STEEL libraries within APR, refer to Section 4.3.1.

The second design decision made was that a cell edge of a certain type (either shared or unshared) could only be abutted with an edge of the same type. In our implementation, we chose to let the APR tool handle this by passing it an additional set of constraints:

- Only abut "shared" edges with "shared" edges.
- Only abut "unshared" edges with "unshared" edges.

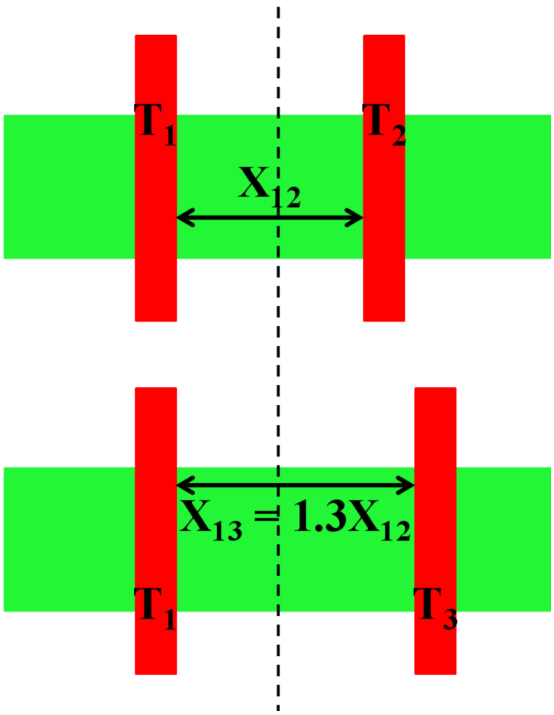


Figure 4.5 Context dependency within STEEL designs.

Details regarding the additional overhead needed to use STEEL within APR is included in Section 4.3.1.

The final implementation detail is a by-product of the layout dependency of stress. Since we are essentially extending the active area between standard cells, differing amounts of active overlap for different combinations of cells could significantly change the Ion and Ioff currents for a given device. Therefore, context dependencies could easily arise if the STEEL library is not carefully designed. To illustrate this problem, consider the example in Figure 4.5, which shows two overlap cases for transistor, T_1. In the first case, the standard cell containing T_1 is placed next to a cell whose nearest device is T_2. The distance, X_{12}, between these two transistors corresponds to the active area length, $L_{s/d}$, of this source/drain region and directly affects the amount of stress induced in both T_1 and T_2. However, in the same design, the same cell type that contains T_1 is used again, but this time is placed next to T_3 and the S/D length increases by 1.3X. In this simple

example, this 30% change will increase the drive current by ~10% (if we assume T_1, T_2, and T_3 are PMOS devices), which is substantial.

One way to handle this context dependency is to characterize the particular device, T_1 for every possible $X_{1,N}$ that could exist by abutting it next to any other "shared" edge in the library. However, since an industrial library typically has many hundreds of cells, this leads to an infeasibly large number of characterizations. Instead, we chose to fix the distance $_{XM,N}$, such that each device T_M and T_N are placed $0.5X_{M,N}$ away from the boundary. We selected a value for $X_{M,N}$ that achieved ~20% and ~8% increases in PMOS and NMOS Ion (for the edge devices) and increased Ioff by ~4X and ~2X, respectively.

4.3 Experimental Results

In order to determine the strengths of the STEEL design methodology, we compared it to two industry design flows: single-V_{th} (using regular-V_{th}, or RVT, cells) and dual-V_{th} (using both RVT and low-V_{th}, or LVT, cells). These comparisons are included in Sections 4.3.2 and 4.3.3, respectively. We also describe a simple assignment technique in Section 4.3.4 which only applies the advantages of STEEL to critical cells, improving leakage at slower delay points or in unbalanced circuits. However, before we examine our results, we begin by briefly discussing how our place and route tools were configured to handle the STEEL library.

4.3.1 APR using STEEL Libraries

As mentioned previously in Section 4.2.4, the various standard cell edge types (either "shared" or "unshared" in our implementation) in the STEEL library add a small amount of complexity to cell placement. To minimize this complexity, we enforced a few additional constraints within the APR tool (discussed in Section 4.2.4). We accomplished this through a custom Tool Command Language (TCL) script that was designed and run

within Cadence's APR tool, Encounter. Essentially, the script steps through each placed standard cell in the design, starting with the top, leftmost cell, and continues from left to right across a single core row before proceeding to the next row down. As the script traverses the standard cell row (from left to right), it checks the adjacent cell edges. If the edges match, the TCL script moves to the next cell. However, if the edges do not match, the script checks if the opposite side of the right cell matches the current cell edge. If it does, the script flips the cell and continues. If neither sides match, then a filler cell is placed in between the cells, to ensure that design rules are satisfied. The penalty incurred is typically minimal, and we found that even with row utilizations of up to ~85%, the STEEL library can be placed and routed using the same floorplan and dimensions as the traditional standard cell libraries.

4.3.2 STEEL versus Regular-V_{th} Results

We begin our analysis by comparing the area, leakage power, and delay of STEEL designs to their traditional, single-V_{th}-based equivalent. The basis of our comparison was an industrial 65nm RVT library. Both libraries were characterized using the stress-enhancement models and flow described in Section 4.2 and pictured in Figure 4.3. With the new .LIB files, we were able to synthesize and place and route a variety of benchmarks using both libraries. In total, we implemented the physical design of 10 benchmarks whose gate count ranged anywhere from ~100 to ~60,000 standard cells. Each benchmark was synthesized at a number of different constraints to determine both the area-versus-delay tradeoff, as well as the leakage-power-versus-delay tradeoff.

For example, Figures 4.6 and 4.7 illustrate these tradeoffs for a Viterbi Decoding circuit (with ~25,000 gates). There are a few interesting points to notice from these plots. First of all, the STEEL version has a better area/delay tradeoff characteristic. Hence, for

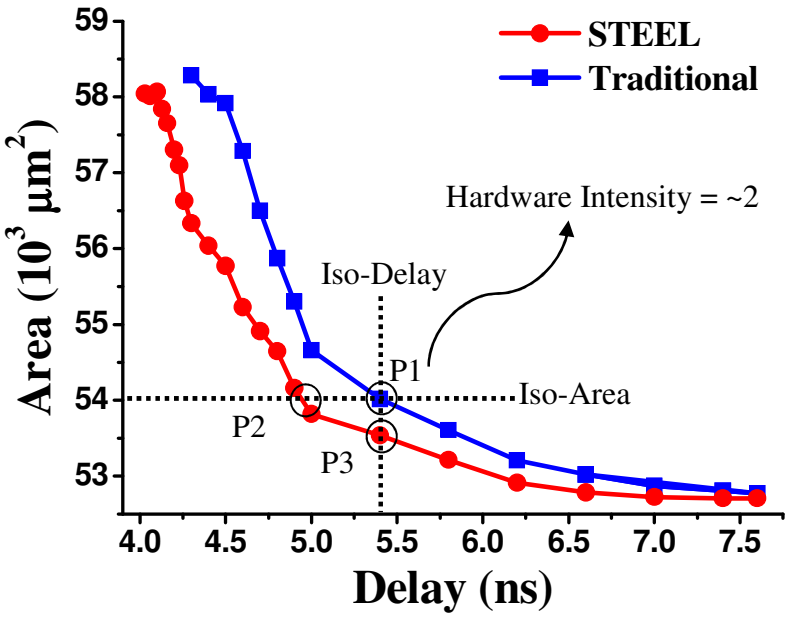


Figure 4.6 Area versus delay for the Viterbi decoder benchmark.

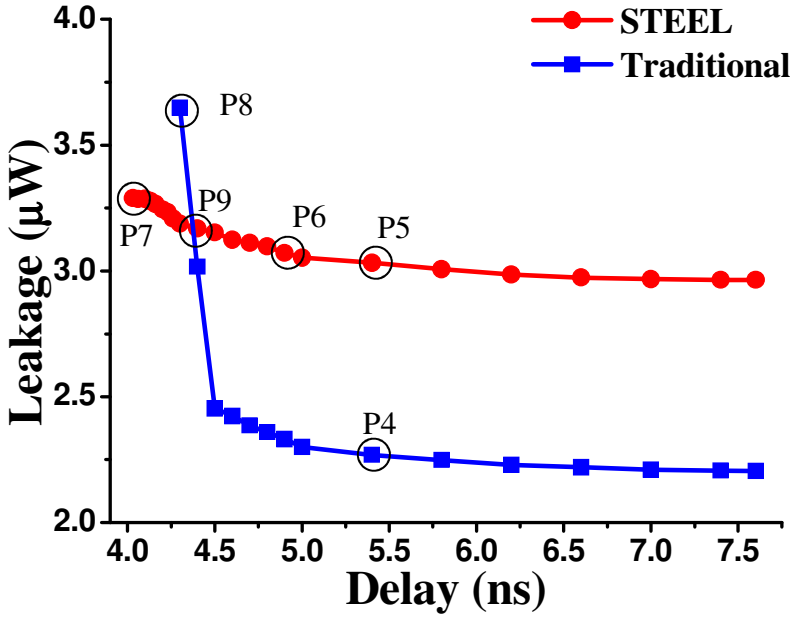


Figure 4.7 Leakage versus delay for the Viterbi decoder benchmark.

the same critical path delay, the STEEL implementation will consume less area. This improvement occurs because the STEEL cells are identical in area to the traditional cells, but have reduced propagation delays (due to the stress-enhancement achieved through

Table 4.1 Design improvement obtained using STEEL (compared against single-V_{th} and dual-V_{th} implementation)

Circuit	Number of Gates	% Delay Improvement (Iso-area	% Area Improvement (Iso-delay)	Leakage Increase (Iso-delay)	Leakage Increase (Iso-area)	% Delay Improvement Beyond Min. Critical Path	$\frac{Dual-Vth\ Leakage}{STEEL\ Leakage}$†
c432	143	18.60%	2.40%	1.41	1.46	12.50%	2.95
c1908	265	6.00%	6.70%	1.11	1.22	9.40%	4.88
c880	291	16.50%	2.60%	1.34	1.39	8.10%	2.37
c2670	489	9.20%	1.10%	1.35	1.34	4.40%	0.85
c3540	921	9.00%	2.10%	1.33	1.36	9.00%	2.08
c7552	1264	11.10%	0.90%	1.27	1.28	12.50%	2.97
c5315	1275	15.50%	1.50%	1.33	1.34	13.30%	2.78
c6288	1703	7.10%	0.40%	1.27	1.28	8.20%	3.52
Viterbi Dec.	25287	8.00%	1.10%	1.33	1.35	6.30%	2.06
Ethernet	66310	8.60%	0.10%	1.50	1.50	7.50%	0.79
AVERAGE		**11.00%**	**1.90%**	**1.32**	**1.35**	**9.10%**	**2.53**

† The dual-V_{th} leakage increase over STEEL is calculated at iso-delay for the minimum critical path delay of the STEEL design.

active-area overlap). Consequently, the physical design tools do not have to size a given STEEL path as aggressively as its corresponding traditional path implementation, leading to reduced area consumption.

Alternatively, if you analyze the circuits at the same value of area (iso-area), STEEL typically reduces delay by 11% (again, due to the stress-enhancement achieved without increasing area). Notice that even at the minimum delay point on the traditional curve, the STEEL library still provides ~9% improvement. Furthermore, if you examine the leakage tradeoff in Figure 4.7, leakage power in the Viterbi decoder increases rapidly on the left side of the plot (toward smaller values of delay). This is due to the fact that to meet these tight timing constraints, the synthesis tool must size up the majority of the

gates in the design, which increases leakage dramatically. Since stress-enhanced gates are designed to primarily give improvements in Ion (and therefore, delay), this region of the curve is where the STEEL library prefers to operate.

The full set of benchmark results compared to the single-RVT library is included in the seven leftmost columns of Table 5.1. This table was constructed using the following procedure. For each benchmark, we analyzed the area/delay tradeoff curve for the traditional 65nm implementation to determine the delay where hardware intensity was ~2. Hardware intensity was originally proposed in [87] as a power versus delay metric. In this work we use a modified version of hardware intensity that compares area and delay. Thus, for the remainder of the chapter, hardware intensity is defined as the percentage change in area over the percentage change in delay. Next, the corresponding values of area and delay (whose hardware intensity is ~2) were used to determine the iso-area and iso-delay comparisons made against the STEEL implementation. For example, in the Viterbi decoder benchmark, the point on the area/delay curve (for the traditional implementation) where the hardware intensity was equal to 2 is labeled point "P1" in Figure 5.6. The corresponding delay improvement that we achieve using STEEL is given in Column 3 of Table 5.1. For the Viterbi decoder, this value is calculated by comparing the delays at "P1" and "P2" (in Figure 5.6). Similarly, area improvement – Column 4 in Table 5.1 – is calculated by comparing the areas at "P1" and "P3". Next, Columns 5 and 6 include the leakage power increase incurred by the STEEL implementation. These values are calculated for the Viterbi circuit by comparing the leakage values at "P4" and "P5" (from Figure 5.7) for the iso-delay case, and comparing "P4" with "P6" for the iso-area column. Finally, the decrease in the minimum critical path delay is noted in Column

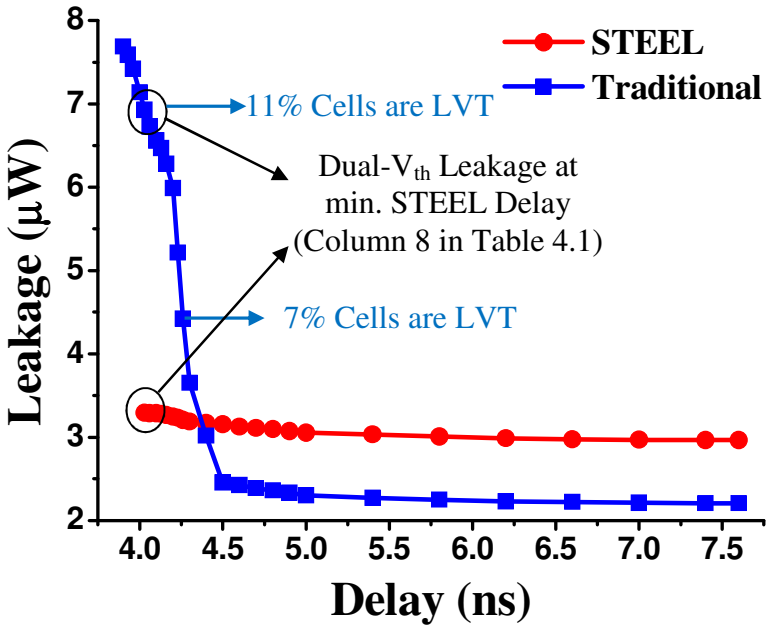


Figure 4.8 Viterbi decoder leakage vs. delay for dual-V_{th} case.

7. This value for the Viterbi decoder is determined by comparing the delay at points "P7" and "P8" in Figure 5.7. The remainder of Table 5.1 is discussed in Section 5.3.3.

Generally, we discovered that for iso-area, the STEEL implementation achieves average delay improvements of 11% while leakage only increases by 35% on average. Additionally, we found that the STEEL-based benchmarks successfully synthesized at a minimum delay value that was, on average, 9.1% less than the traditional minimum delay.

4.3.3 STEEL versus Dual-V_{th} Results

In addition to a significantly improved area-delay tradeoff for STEEL versus a single-V_{th} standard library, we now demonstrate that STEEL provides superior performance with a single-V_{th} over a traditional dual-V_{th} library for the majority of operating points where dual-V_{th} would be of interest. This arises due to the improved Ion vs. Ioff tradeoff using stress enhancement compared to using low-V_{th} devices and indicates that STEEL simultaneously offers a better power/performance envelope and lower manufacturing costs over dual-V_{th}. Figure 4.8, for example, illustrates the

leakage/delay curve for the dual-V_{th} implementation of the Viterbi decoder (notice its similarity to Figure 4.7). The slower part of the curve (delay > 4.26ns) is actually identical to Figure 4.7, due to the fact that only RVT cells are used in the design until the delay constraint becomes less than or equal to 4.26ns. In the region of interest for STEEL, we found that the leakage crossover point (where dual-V_{th} leakage becomes greater than STEEL) typically occurred between the most tightly constrained RVT design (with zero LVT cells) and the dual-V_{th} implementation that used the minimum number of LVT cells needed to satisfy timing. Since the LVT cells in our industrial library typically increased leakage by ~20X, the minimum leakage for the dual-V_{th} case occurred at the timing constraint that used the minimum number of LVT cells. Even at this minimum leakage point for dual-V_{th} (where the number of LVT cells is only a small percentage of the total number of cells, <5%), the substantial leakage increase per low-V_{th} cell caused this minimum-leakage, dual-V_{th} implementation to almost match the leakage increase incurred by STEEL. Over all of the benchmarks, we found that even at the minimum dual-V_{th} leakage, dual-V_{th} only showed a 2.9% average savings in leakage over STEEL. Furthermore, by the time the STEEL implementations reached their minimum delay, the dual-V_{th} leakage had increased to ~2.5X the average value of STEEL leakage (displayed in the last column of Table 5.1). An example point for the Viterbi decoder circuit for this value is shown in Figure 4.8.

Since the STEEL implementations can typically provide up to ~10% delay improvements over single-V_{th} designs while consuming only a fraction of the leakage power of dual-V_{th}, STEEL can provide more optimal designs in two ways. First, for designs that only need moderate delay improvements – less than 10% – STEEL can be

used to achieve these improvements. By utilizing the STEEL standard cells, the designer would not only reduce leakage (as compared to the dual-V_{th} implementation), but would also dramatically reduce manufacturing costs, since the second threshold voltage mask would not be needed. Alternatively, STEEL could also be used in conjunction with the dual-V_{th} approach to achieve more optimal designs (in terms of area and power). Since typical dual-V_{th} processes only provide coarse-grain threshold voltage values, some standard cells in a path might be sub-optimally assigned if they do not need the full performance enhancement provided by moving to a lower V_{th} value. For these cells, the STEEL versions would be more appropriate, since they can obtain more fine-grained performance improvements and will fill some of the performance space between V_{th} values. Additionally, by designing LVT STEEL cells, delay improvement can be extended beyond the performance of dual-V_{th}.

4.3.4 Intelligent STEEL-Cell Assignment

One interesting discrepancy that we found during this work was the fact that in our largest circuit, an ethernet controller, the STEEL design did not outperform the dual-V_{th} implementation. In fact, out of the 10 benchmarks, the ethernet circuit was the only case where we did not obtain improvements in leakage versus dual-V_{th}. To understand this phenomenon, we analyzed the structure of the ethernet controller and made some interesting observations:

- Even though the ethernet controller used a large number of standard cells, its paths were not balanced and the number of critical paths only represented a small fraction of the total number of paths.

- Out of ~66,000 standard cells, the dual-V_{th} design only used 285 LVT cells (<1% of the total) to meet the minimum timing constraint achieved using STEEL.

With this knowledge, it was clear why the STEEL implementation did not improve upon the dual-V_{th} case. Since we had not previously employed any delay/leakage optimization in our approach, the ~1.3X STEEL average leakage increase per standard cell occurred in each of the ~66,000 standard cells, whereas the ~20X leakage increase per LVT cell only occurred in <1% of the total cells. Therefore, while the STEEL designs outperformed dual-V_{th} in the majority of our experiments, it was clear that exploring intelligent assignment schemes would be beneficial to our work, both to improve the STEEL leakage performance in unbalanced designs (as compared to dual-V_{th}), as well as achieve leakage values closer to the RVT-based designs.

So far, we have reported the STEEL results for the case where we use our stress-enhanced library uniformly across a given design (i.e, every gate in the circuit is assigned to its stress-enhanced version). However, not all of the gates in a circuit need performance enhancement to meet timing for a given delay constraint. These non-critical gates only add to the leakage overhead, and as a result we observed that the STEEL designs had larger leakage than their single-V_{th} counterpart, even at larger values of delay (more relaxed delay constraints). Thus, there is ample scope for intelligent assignment of stress-enhanced cells, where the traditional RVT library is used in conjunction with STEEL, and the STEEL cells are only assigned to timing critical gates. An intelligent cell assignment scheme will substantially reduce the leakage overhead but maintain similar improvements in delay. The benefits of this technique derive from

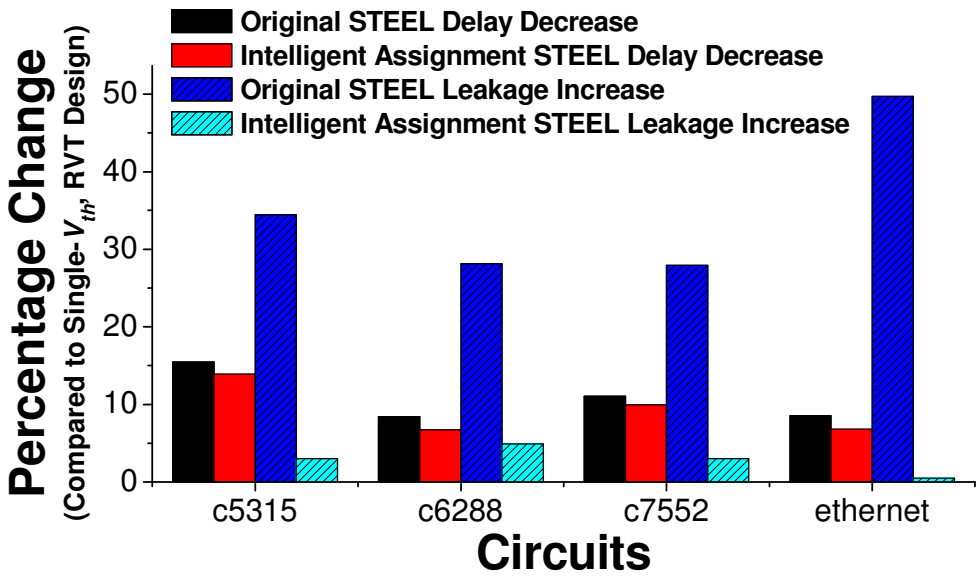


Figure 4.9 Impact of intelligent STEEL assignment on delay and leakage.

the fact that only a fraction of total number of gates in a circuit are timing critical. Replacing only the critical gates with the leakier, higher-performance versions will result in significantly lower leakage increases, as compared to the case where all of the gates are replaced.

As a further investigation into the scope of intelligent assignment, we perform a simple experiment where we replace only the top ~10%, timing critical gates in a circuit with their stress-enhanced versions. We perform this experiment at the same hardware intensity point (discussed previously) on the area-versus-delay curve for the traditional RVT library, and compare the delay improvement and leakage overhead numbers to the case where stress enhancement was used in every cell (Column 3 and Column 6 of Table 4.1, respectively). Figure 4.9 shows the percentage improvement that we observe using intelligent assignment, as compared to the uniform-replacement ("Original" STEEL) scheme. Ideally, we would prefer to obtain all of the delay improvement achieved in the previous section (i.e., achieve 100% of the typical 11% delay improvement over RVT),

while reducing the percentage leakage increase to 0% (i.e., matching the RVT leakage). As shown in the figure, we can get >80% of the "Original" delay improvement through selective replacement, while incurring a much smaller increase in leakage. The selective scheme typically reduces the uniform STEEL leakage increase by ~90%. From Figure 4.9, observe that the leakage number for the ethernet benchmark is exceptionally small because, despite its large size (~66,000 gates), the number of timing critical gates is very small (as mentioned previously). Thus, to achieve 80% of the "Original" improvement, only 625 gates need to be replaced with their stress-enhanced version (less than 1% of the total gates), which results in substantial leakage savings that is comparable with dual-V_{th}.

Intelligent replacement schemes like this approach allow STEEL to maintain its advantage over dual-V_{th}, even for designs that are extremely unbalanced (such as the ethernet benchmark). Additionally, this approach can be used to improve leakage power consumption within any STEEL design (especially for relaxed delay constraints). This means that the leakage for the STEEL technique will approach that of the traditional RVT library, especially at delay constraints located to the right of the leakage crossing point (e.g., all of the STEEL leakage values to the right of point "P9" in Figure 4.7 will be much closer to RVT).

4.4 Summary

In this chapter, we proposed STEEL, a new standard cell library design technique for modern stress-enhanced semiconductor processes. STEEL fully exploits the layout dependencies of stress. By designing the STEEL standard cells to share the V_{DD} and V_{SS} source/drain connections across cell boundaries, one can achieve drive current improvements of up to 20%. While implementing the proposed standard cell approach in a number of benchmark circuits, we demonstrated average delay reductions of 11% with

only a 35% average increase in leakage, compared to single-V_{th} implementations. Additionally, STEEL-based circuits typically achieved a ~2.5X reduction in leakage when compared to dual-V_{th} designs. This implies that for designs requiring an 11% delay improvement (or less) beyond a single-V_{th} implementation, STEEL can provide this improvement for a smaller leakage penalty as well as much lower manufacturing costs compared to dual-V_{th}. Orthogonally, STEEL can also be used in conjunction with dual-V_{th} (similar to the work in Chapter 3) to provide more optimal designs (in terms of both leakage and delay).

Chapter 5
Closed-Form Modeling of Layout-Dependent Mechanical Stress

Mechanical stress inducing layout features are used by modern CMOS processes in order to enhance carrier mobility, for higher performance. Mechanical stress breaks the crystal symmetry of Silicon, causing changes in the band scattering rates, and/or the carrier effective mass, which in turn affects carrier mobility [41, 42]. Application of the correct type of stress (tensile or compressive) results in significantly higher carrier mobility, and improves transistor performance [75]. There are three major layout dependent sources of mechanical stress: Shallow Trench Isolation (STI) generates compressive stress due to thermal mismatch with Silicon [43], embedded SiGe is epitaxially grown in the S/D regions of PMOS devices to induce high compressive stress due to lattice mismatch [44], and tensile/compressive nitride liner layers are integrated into a single, high performance process flow called the Dual Stress Liner (DSL) approach [45]. However, stress introduced in the channel, and hence carrier mobility, show a strong dependence on the device layout and its neighboring features [47]. As a result, layout properties such as active area length, number of contacts, distance of the device to the well edge, etc. become important in determining the mechanical stress induced in the channel of a device. Figure 5.1 shows the layout view for the three PMOS devices in a 3-input NAND gate, along with the corresponding longitudinal stress distribution under the

channel, for a selected cross-section. Although the three devices have identical gate width and length, the channel stress is different in the three cases depending on other layout features such as active area length, and contact placement. The device in the center (device 2) has higher stress than the two corner transistors because it is surrounded by more SiGe. This difference in stress is reflected in their performance, and simulations show that the drive currents for the center and edge devices differ by 8.2%. Such dependence can result in significant variation in the performance and leakage of devices, based on their context and layout.

Technology computer-aided design (TCAD) tools have been used to simulate device fabrication in order to capture process induced mechanical stress, and calculate its impact on device performance and leakage. However, TCAD tools based simulation frameworks involve time consuming computational steps, and have severely limited capacity in terms of the number of devices that can be accurately simulated in a single run. Hence, there is an urgent need to develop scalable, closed-form models for calculating process induced stress as a function of the device layout, and its neighboring

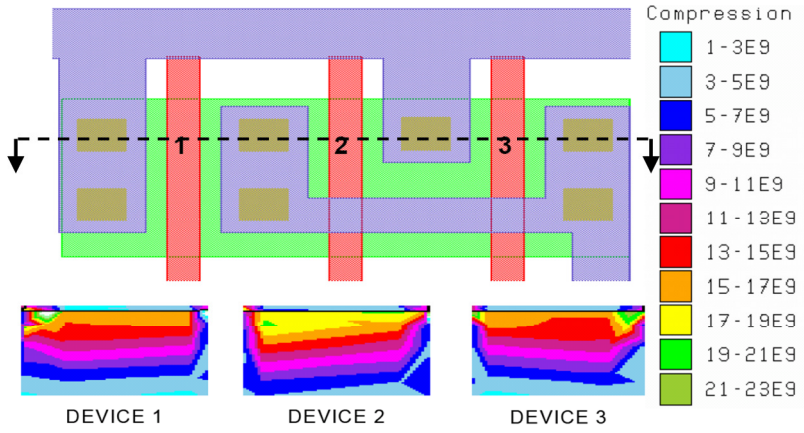


Figure 5.1 Channel stress distribution for PMOS devices in a 3-input NAND for a selected cross-section.

features, to enable fast and accurate modeling and simulation of strained devices. In the past, [68, 69] have studied this layout dependence for different sources of stress, for both NMOS and PMOS devices. However, there has been very little work on comprehensive closed-form models of the layout dependence of process induced stress, and its impact on carrier mobility. Authors in [67] focused mainly on modeling mobility changes due to STI stress. Reference [71] presented a very good method at modeling layout dependence of process induced stress through non process specific analytic models. However, while these models show a good fit for isolated device level stress simulation, they do not account for layout features such as distance of device from the well edge (tensile/compressive liner interface), presence of contacts, dummy poly, and neighboring devices. The paper also does not account for the transverse/lateral stress dependence on layout. So, while these models provide a good fit for simple device level experiments, they fail to account for key neighboring features which are critical for accurate stress simulation, when focusing on the complete circuit layout.

In this work, we propose compact closed-form models for layout dependence of process induced stress, and its impact on carrier mobility. We analyze the physics behind stress inducing process steps, and solve relevant equations describing the stress distribution, in order to develop the models. Since the derivation is based on underlying physics, the derived models are scalable. We model stress due to Shallow Trench Isolation (STI), tensile/compressive nitride liners, and embedded SiGe S/D layers (used only in PMOS devices). In order to quantify the impact of stress on mobility, we use the piezoresistive model [88]. Since longitudinal stress varies across the device width; we

propose partitioning the gate into segments, such that each segment has almost constant stress, based on measured, stress-critical, layout parameters. We calculate the stress based mobility enhancement, in terms of mobility multipliers, for each of these segments, and take a weighted average of these multipliers based on the slice widths to derive one mobility multiplier for each device.

The rest of the chapter is organized as follows. Section 5.1 discusses the derivation of stress models for the different stress inducing process steps, along with the translation of stress into impact on device mobility. Experimental results are discussed in Section 5.2, and Section 5.3 concludes the chapter.

5.1 Modeling Stress-enhanced Carrier Mobility

For model based simulation of strained devices, we need to calculate the mechanical stress induced in the device channel, and then translate the stress into impact on carrier mobility. This impact is quantified in terms of mobility multipliers, which can then be used in circuit simulators such as SPICE to capture the stress effect. In this section, we first present our closed-form stress models to enable fast and accurate stress modeling, and the second part of the section discusses translating these stress numbers into mobility multipliers to calculate the impact on performance and leakage by using SPICE.

5.1.1 Stress Models

We develop our stress models by analyzing the physics behind various stress inducing process steps, and solving relevant equations. We analyze each source of stress separately, and add up the stress due to each source, to obtain overall stress in the device channel. Since the models are based on the physics behind each process step, they are scalable for future technology generations. The sources of stress modeled are: embedded

SiGe S/D layer (for PMOS devices), tensile/compressive nitride liners, and Shallow Trench Isolation (STI). The models represent a very simple combination of transverse and longitudinal direction 1D spring approximations. The physics based derivation is done under multiple simplifying assumptions and is supposed to provide a general form for the model, while the actual parameter values come from rigorous calibration-optimization. For each device, we consider all the features within a certain window of influence (of length L_W), to calculate the resulting stress.

Embedded SiGe source/drain: For PMOS devices, SiGe is epitaxially grown in cavities that have been etched into the source/drain areas. A large compressive stress is created in the PMOS channel due to lattice mismatch between Si and SiGe, thereby resulting in significant hole mobility improvement. In this process, NMOS is protected by a capping layer to prevent Si recess, and SiGe epitaxial growth. The key to modeling the magnitude of induced stress is to identify the physics behind generation of compressive stress, and solve relevant equations by applying simple spring approximations. We assume that the widths of all structures are much bigger than their lengths (quasi 1D case).

Figure 5.2 shows a very simple layout used to explain the derivation of 1D models for compressive stress generated due to embedded SiGe. The layout is composed of two simple devices separated by STI, one with embedded SiGe in S/D regions (device 0), and the other without it (device 1). Ge has a lattice constant larger than Si and hence it occupies more volume than Si would occupy. The gray areas (SiGe) can be seen as trying to expand in all the directions. The scenario after epitaxial growth of SiGe is depicted in the bottom picture of Figure 2. If χ is an atomic ratio of Ge in Si and Ω_{Si} and Ω_{Ge} are the atomic volumes of Si and Ge, respectively, then it is easy to show that an initial volume

V_0 (volume without introduction of SiGe in the S/D) would try to expand by $\Delta V = \left(\frac{\Omega_{Ge}}{\Omega_{Si}} - 1\right)\chi V_0$, which translates to a linear expansion of $\Delta L_F = \sqrt[3]{\left(\frac{\Omega_{Ge}}{\Omega_{Si}} - 1\right)\chi L_0}$ in all the three dimensions. As shown in the middle picture of Figure 2, in the absence of any confinement (neighboring features), SiGe would have expanded by this amount. The expansions for the left and right SiGe regions can therefore be expressed as:

$$\Delta L_{F\_L0} = \sqrt[3]{\left(\frac{\Omega_{Ge}}{\Omega_{Si}} - 1\right)\chi L_{SG\_L0}} \qquad \Delta L_{F\_R0} = \sqrt[3]{\left(\frac{\Omega_{Ge}}{\Omega_{Si}} - 1\right)\chi L_{SG\_R0}} \tag{1}$$

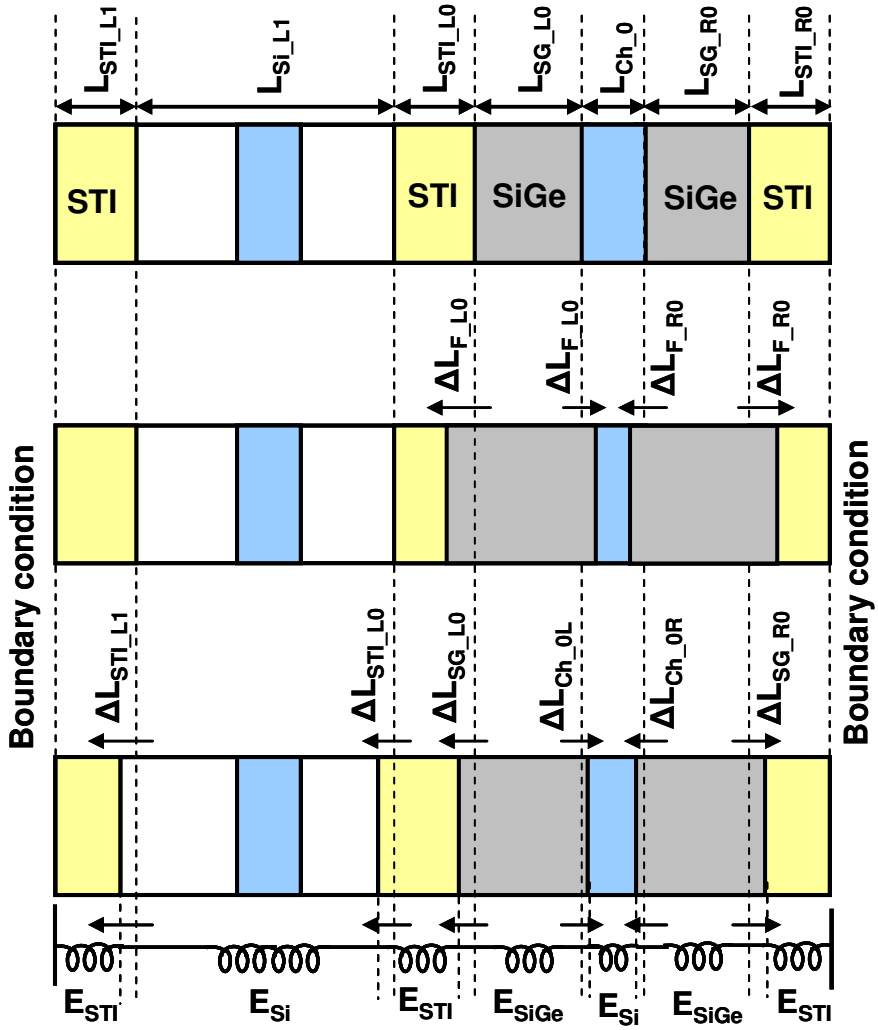


Figure 5.2 Before SiGe expansion (top), after non-confined SiGe expansion (middle), and after deformation of all segments due to SiGe expansion (bottom).

In reality, the presence of neighboring features opposes such an expansion, thereby creating compressive stress in the device channel. The deformation of the SiGe sub-segment as compared to the non-confined case can be expressed as the difference between the non-confined and the actual confined case expansion.

The bottom picture in Figure 5.2 shows the deformation of different segments after SiGe expansion. We consider each layout segment as being represented by a spring (or an elastic beam) characterized by different elasticity. It is assumed that displacements at ends of the considered segment (leftmost and rightmost edges) are equal to zero. This might be treated as the symmetry boundary conditions. At equilibrium, the forces acting from one sub-segment on another at the points of contact are equal. It provides us, in the frame of the accepted approximation, with the condition of equal stress along the entire line of cross-section. This stress value will depend on the layout composition in the region of interest. So, we can express the generated longitudinal stress in different segments with following equations.

$\sigma_{STI\_L1} = \frac{-\Delta L_{STI\_L1}}{L_{STI\_L1}} E_{STI}$	$\sigma_{Si\_L1} = \frac{\Delta L_{STI\_L1} - \Delta L_{STI\_L0}}{L_{Si\_L1}} E_{Si}$
$\sigma_{STI\_L0} = \frac{\Delta L_{STI\_L0} - \Delta L_{SG\_L0}}{L_{STI\_L0}} E_{STI}$	$\sigma_{SG\_L0} = -\frac{\Delta L_{F\_L0} - \left(\Delta L_{SG\_L0} + \Delta L_{Ch\_0L}\right)}{L_{SG\_L0}} E_{SiGe}$
$\sigma_{Ch\_0L} = -\frac{\Delta L_{Ch\_0R} + \Delta L_{Ch\_0L}}{L_{Ch\_0}} E_{Si}$	$\sigma_{SG\_R0} = -\frac{\Delta L_{F\_R0} - \left(\Delta L_{SG\_R0} + \Delta L_{Ch\_0R}\right)}{L_{SG\_R0}} E_{SiGe}$
$\sigma_{STI\_R0} = \frac{\Delta L_{STI\_R0} - \Delta L_{SG\_R0}}{L_{STI\_R0}} E_{STI}$	

(2)

Here E_{SG}, E_{Si} and E_{STI} are the elasticity constants of the $Si_{1-x}Ge_x$, silicon, and STI, various ΔL and L are the deformations and nominal dimensions as shown in the figure, and σ_a is the stress generated in segment a due to SiGe expansion.

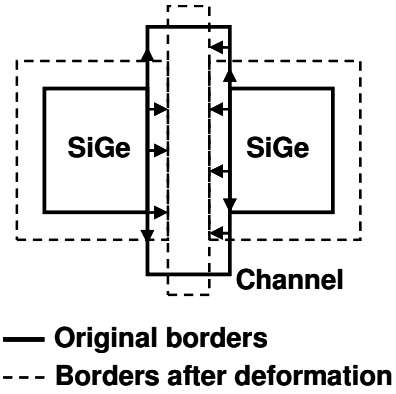


Figure 5.3 Sample device layout showing generation of transverse stress.

Using the condition of equal stress, we can set up the system of equations for determination of unknown deformations of the segments. The deformation numbers for each segment can then be used to determine the value of stress generated. Upon solving these equations, we obtain the longitudinal stress in the channel:

$$\sigma^{L}_{Ch\_0} = -\frac{\beta\left(L_{SG\_L0} + L_{SG\_R0}\right)}{d_{STI} + d_{Si} + (1+\beta)d_{SiGe}} \tag{3}$$

$$d_{STI} = \frac{L_{STI\_L1} + L_{STI\_L0} + L_{STI\_R0}}{E_{STI}} \qquad d_{\mathrm{Si}} = \frac{L_{Ch\_0} + L_{Si\_L1}}{E_{Si}}$$

$$d_{SiGe} = \frac{\left(L_{SG\_L0} + L_{SG\_R0}\right)}{E_{SiGe}} \qquad \beta = \sqrt[3]{\left(\frac{\Omega_{Ge}}{\Omega_{Si}} - 1\right)\chi}$$

In general, for any given layout we can write the longitudinal stress in a channel as:

$$\sigma^{L}_{SiGe} = -\frac{\beta\left(\sum_j L_{SG\_j} + \sum_i L_{neigh\_L(R)i}\right) + \left(\Delta L_{BC\_L} + \Delta L_{BC\_R}\right)}{\frac{\sum_k L_{STI\_k}}{E_{STI}} + \frac{\sum_n L_{Si\_n}}{E_{Si}} + \frac{(1+\beta)\left(\sum_j L_{SG\_j} + \sum_i L_{neigh\_L(R)i}\right)}{E_{SG}}} \tag{4}$$

Here L_{SG_j} is the length of the j-th $Si_{1-x}Ge_x$ S/D segment on the same active area as the device while L_{neigh_Lj} and L_{neigh_Rj} are the length of the neighboring SiGe active areas. L_{STI_k} is the k-th STI width, L_{Si_n} is the length of the n-th gate or non-SiGe source/drain (NMOS active) area in the longitudinal direction. ΔL_{BC_L} and ΔL_{BC_R} are the boundary

conditions at the left and right window edges representing stress-induced edge displacements.

In addition to the generation of compressive longitudinal stress due to SiGe growth, transverse strain/stress is also generated because of traction between channel segment and adjacent SiGe structures. The expansion of these SiGe drain structures in transverse direction causes the adjacent silicon (channel area) to expand as well. This is illustrated in Figure 3. Hence, in order to estimate SiGe induced transverse stress in the device channel, we need to account for stress caused by the traction with adjacent SiGe areas due to SiGe expansion.

The transversal stress can be calculated as

$$\sigma_{Ch\_0}^{T} = \frac{E_{Si}}{W_{Ch}}\left(\frac{\sigma_{SG\_L0}^{T} + \sigma_{SG\_R0}^{T}}{2E_{SG}} + \frac{\beta}{1+\beta}\right) \tag{5}$$

Here W_{Ch} is the width of the channel. $\sigma_{SG\_L0}^{T}$ and $\sigma_{SG\_L0}^{T}$ are the stress in adjacent left and right S/D $Si_{1-x}Ge_x$ structures, which can be calculated in a manner similar to Equation 4 by replacing all *L* (horizontal distances) by *W* (vertical distances). Indexes *T* and *B* are for top and bottom, respectively.

$$\sigma_{SiGe}^{T} = -\frac{\beta\left(\sum_j W_{SG\_j} + \sum_i W_{neigh\_T(B)i}\right) + \left(\Delta L_{BC\_T} + \Delta L_{BC\_B}\right)}{\frac{\sum_k W_{STI\_k}}{E_{STI}} + \frac{\sum_n W_{Si\_n}}{E_{Si}} + \frac{(1+\beta)\left(\sum_j W_{SG}^{j} + \sum_i W_{neigh\_T(B)i}\right)}{E_{SG}}} \tag{6}$$

Nitride Liner: Capping stressed layer technology is one of the most important techniques employed to generate a desirable stress in device channel. Traditionally, a silicon nitride based contact etch stop layer (CESL) is used as the source of the tensile stress. In this technology, a $Si_xN_yH_z$ layer is deposited followed by a special type of anneal to release hydrogen. This results in volume shrinking, which generates strong

tensile stress in the surrounding confinement that gets transferred into the channel region of NMOS devices. In order to avoid tensile stress generation in PMOS devices, different technological steps were introduced. The most effective way was to dope the CESL in the PMOS regions with a *Ge* implant that results in volume expansion, and compressive stress generation in the confinement. Latest high performance process nodes have simultaneously incorporated both tensile and compressive nitride liners into a single high performance CMOS flow, called the Dual Stress Liner approach. Nwell mask is generally used while defining the compressive and tensile regions and nwell edges can be seen as the interface of compressive and tensile nitride.

We define α as the coefficient of proportionality between the as-drawn length (L_{CESL}) of a CESL segment (stress effect is not accounted), and the confinement-free length (L_{CESL}^{*}) of the same segment if the nitride layer was allowed to expand/contract without any confinement imposed by neighboring features: $L_{CESL}^{*} = \alpha.L_{CESL}$. Having defined that, we can then proceed to calculate the stress generated due to nitride in a manner similar to embedded SiGe. The quasi 1D approximation yields the following expression for capping layer induced longitudinal stress as a function of layout geometry.

$$\sigma_{CESL}^{L} = \frac{(1-\alpha)\sum_{i} L_{CESL\_i} + \left(\Delta L_{BC\_L} + \Delta L_{BC\_R}\right)}{\frac{\alpha\sum_{i} L_{CESL\_i}}{E_{CESL}} + \frac{\sum_{j} L_{Poly\_j}}{E_{Si}} + \frac{\sum_{k} L_{Contact\_k}}{E_{Contact}}} \tag{7}$$

Here, L_{CESL_i} is the length of *i*-th stress layer segment either between two neighboring poly, or between poly and contact, or poly and border of the chosen window, L_{Poly_j} is the length of the *j*-th gate (channel length), and $L_{Contact\_k}$ is the contact size, all in the longitudinal direction. Similar to the SiGe case, ΔL_{BC_L} and ΔL_{BC_R} are the boundary conditions at the left and right window edges representing stress-induced edge

displacements and E_{CESL}, E_{Si} and $E_{Contact}$ are the elasticity constants of the capping layer, silicon, and the contact material, respectively. In the absence of contacts, $L_{Contact}$ is taken as 0.

Stress in the transverse direction can be obtained by replacing all the longitudinal measurements with transverse measurements and left and right boundary conditions with the corresponding top and bottom limits. The traverse stress can then be expressed as:

$$\sigma_{CESL}^{T} = \frac{(1-\alpha)\sum_{i} W_{CESL\_i} + \left(\Delta L_{BC\_T} + \Delta L_{BC\_B}\right)}{\frac{\alpha\sum_{i} W_{CESL\_i}}{E_{CESL}} + \frac{\sum_{j} W_{Poly\_j}}{E_{Si}} + \frac{\sum_{k} L_{Contact\_k}}{E_{Contact}}} \tag{8}$$

Figure 5.4 shows a set of relevant layout parameters for CESL stress calculation. As predicted by the proposed model, the presence of polysilicon gates and contacts decreases the stress due to nitride liner by breaking the continuity of the deposited nitride liner layer. Contacts create holes in the liner layer, while polysilicon gates cause a bump in the deposited liner layer to

bring down the stress. As a result, an isolated device with no contacts will have the highest stress due to nitride. These effects are included in the models expressed in Equations 7 and 8.

Shallow Trench Isolation: Shallow Trench Isolation (STI) creates stress due to thermal mismatch between silicon and STI. The difference in the thermal expansion coefficients causes compressive stresses to develop in the device once the wafer is cooled down post annealing. We can quantify the magnitude of generated stress using the expression for linear contraction that causes the stress to develop. For a given silicon segment, contraction upon cooling can be quantified as:

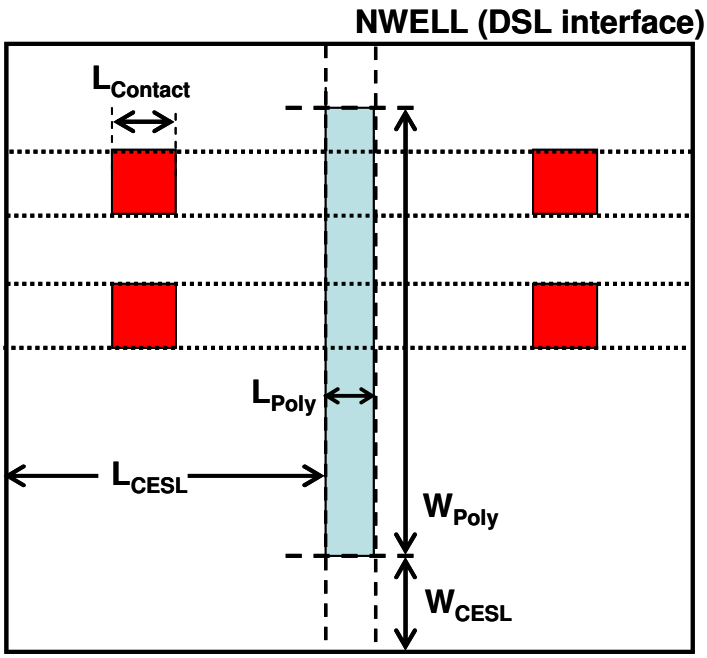


Figure 5.4 Sample layout parameters for CESL stress calculation.

$$\Delta L_{x,y} = \left(\alpha_{Si} \times \Delta T \times L_{x,y}\right) \tag{9}$$

Here $\Delta L_{x,y}$ is change in length upon cooling, α_{Si} is the thermal expansion coefficient of Silicon, ΔT is the difference between the anneal temperature and the final temperature, and $L_{x,y}$ is the as-drawn length of the considered segment. This is the contraction that would occur in the absence of any confinement. We can then proceed to calculate STI stress for a given layout segment, by following an approach similar to that used for calculating stress due to SiGe, and nitride.

The longitudinal stress can be expressed as:

$$\sigma_{STI}^{L} = -\frac{(\alpha_{Si} - \alpha_{STI})\Delta T \sum_i L_{STI\_i} + \left(\Delta L_{BC\_L} + \Delta L_{BC\_R}\right)}{\frac{(1-\alpha_{STI}\Delta T)}{E_{STI}} \sum_i L_{STI\_i} + \frac{(1-\alpha_{Si}\Delta T)}{E_{Si}} \sum_j L_{Si\_j}} \tag{10}$$

Here L_{STI_i} is the length of the i-th STI segment, L_{Si_j} is the length of the j-th silicon segment, α_{Si} and α_{STI} are the coefficients of thermal expansion for silicon, and STI,

respectively. Replacing longitudinal measurements by lateral (transverse) measurements and left and right boundary conditions by top and bottom edges, we get the following expression for transverse stress:

$$\sigma_{STI}^{T} = -\frac{(\alpha_{Si} - \alpha_{STI})\Delta T \sum_i W_{STI\_i} + \left(\Delta L_{BC\_T} + \Delta L_{BC\_B}\right)}{\frac{(1-\alpha_{STI}\Delta T)}{E_{STI}} \sum_i W_{STI\_i} + \frac{(1-\alpha_{Si}\Delta T)}{E_{Si}} \sum_j W_{Si\_j}} \tag{11}$$

It should be noted that all the derived formulas which describe the stress generated by different stress sources ((4), (6), (7), (8), (10), (11) contain the window edge displacements terms ΔL_{BC}. These displacements generally should be equal to zero, in accordance with the assumption of symmetry boundary condition. However, in some specific cases, when the effect of global load, such as packaging, chip mounting or 3D integration, on the variation of transistor-to-transistor characteristics is of interest, these terms should come from the *global* finite element based simulation. Also note that these models provide a general form for functions to estimate stress, the values for parameters such as E, α, etc. are obtained by calibration optimization and might be different from the actual physical values.

5.1.2 Converting Stress to Mobility

The layout dependence of process induced stress leads to gates with non uniform stress, and, hence, non uniform mobility, in the device channel across the width of the device. Based on the closed-form models, we know the layout parameters that affect the stress induced in the channel (such as number of contacts, distance of device from well-edge, active area length, etc.). This knowledge can used to partition the device gate into segments, such that these stress-critical geometrical parameters for a given segment are constant throughout the segment width. We can then calculate stress, and its impact on

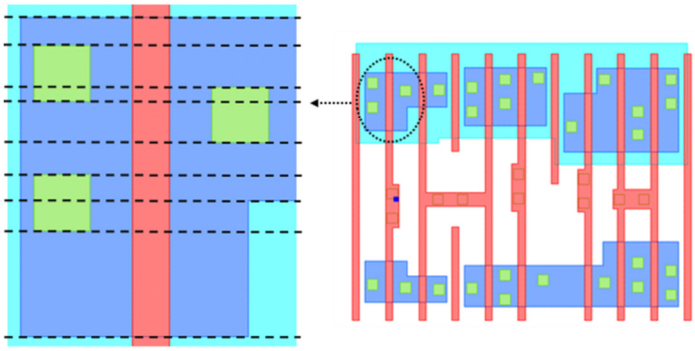


Figure 5.5 MUX layout showing stress based partitioning of a random PMOS device.

mobility, for each of these segments independently, and take a weighted average of mobility multipliers for different segments (based on segment width), to determine one single value of mobility multiplier for the strained device. Figure 5.5 shows a sample device layout (selected from a larger MUX circuit layout) partitioned into segments. For each segment, we can then proceed to calculate stress due to different sources, and sum it up to obtain the overall stress in each direction. However, in accordance with Poisson's Effect, layout generated longitudinal strain also produces a transverse strain which is given by $\varepsilon^T = -\nu\varepsilon^L$; where ν is the Poisson's factor, ε^T is the transverse strain, and ε^L is the longitudinal strain. Similar relationship exists for longitudinal stress caused by layout generated transverse stress/strain. A complete stress distribution in the *j*-th segment can then be expressed as following:

$$\begin{aligned} \sigma_j^{L(tot)} &= \sigma_j^L - \nu\sigma_{Ch}^T \\ \sigma_j^{T(tot)} &= \sigma_{Ch}^T - \nu\sigma_j^L \end{aligned} \tag{12}$$

where, $\sigma_j^{L(tot)}$ and $\sigma_j^{T(tot)}$ are the total longitudinal and the total transverse stress in the segment; σ_j^L and σ_{Ch}^T are the longitudinal and transverse stress values calculated based on the model, and *v* is the Poisson factor. As shown in Figure 5, the longitudinal stress is different for different segments of the device based on the longitudinal layout parameters

while the traverse stress is same for the entire channel. Finally, we use piezoresistive coefficients to convert from stress to mobility [88]. Mobility multiplier (μ_{mult}) for a given segment is expressed as:

$$\mu_{mult} = 1 + \frac{\Delta\mu}{\mu} = 1 + \pi_L \sigma_j^{L(tot)} + \pi_T \sigma_j^{T(tot)} \tag{13}$$

Here, π_L and π_T are the longitudinal and transverse piezoresistive coefficients, respectively. Since the piezoresistive coefficients have a strong dependence on the doping concentrations [88], we assume that these coefficients come from calibration optimization as well. Finally, we can take a width based weighted average of these multipliers to obtain an overall device mobility multiplier, which can then be used in a circuit simulator such as SPICE for accurate simulation of strained devices.

5.2 Experimental Results

In order to verify the accuracy of proposed stress models, we used TCAD simulation based stress and on-current (Ion) data for NMOS and PMOS devices in various configurations. We also validated our models against ring oscillator frequency data from an experimental test chip fabricated in a process that contains both nitride liner and SiGe stress enhancement techniques. The models were separately calibrated for each case by setting up a system of equations in terms of the unknown model coefficients (π_L, π_T, α, etc.) using measured layout parameters. We wrote a simple layout editor script to measure layout distances, and segment the device gate into regions with equal stress. As discussed in the previous section, this segmentation is done such that the stress-critical layout parameters such as active area length, etc. are constant for each segment. SPICE based simulations were used to generate tables for dependence of Ion on mobility

multipliers. Finally, MATLAB code based on least squares fitting is used to solve for model coefficients using these equations.

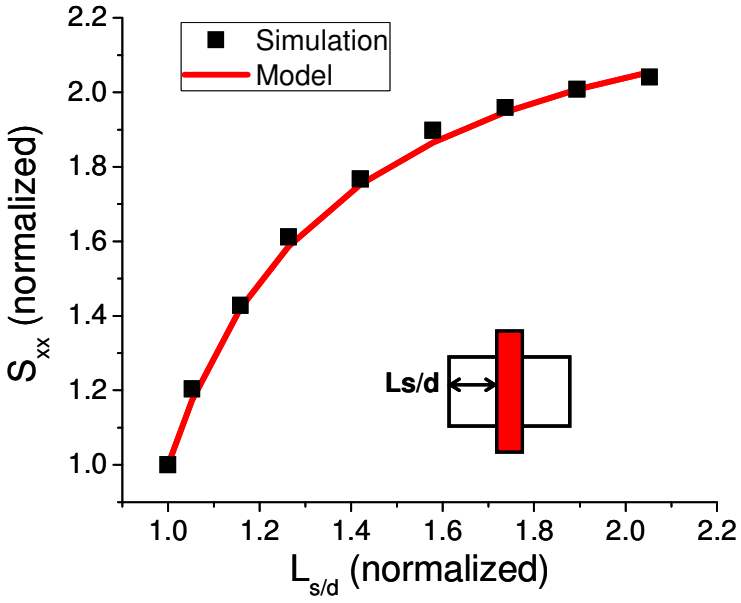


Figure 5.6 Longitudinal channel stress as a function of active area length as obtained by TCAD simulations and after proposed model fitting.

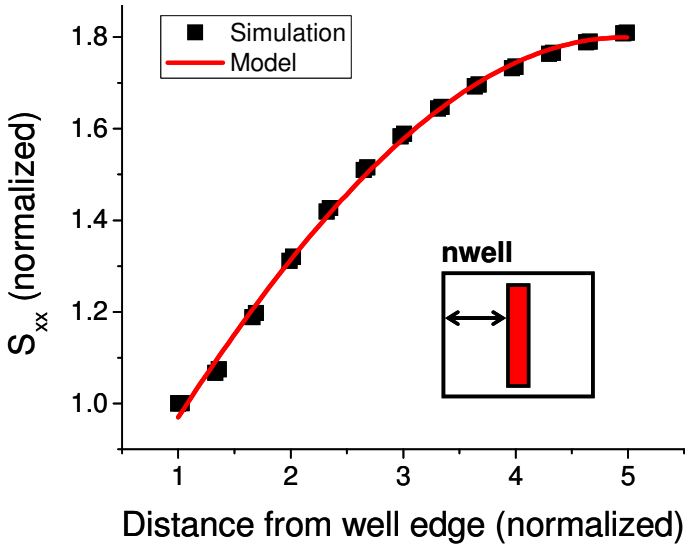


Figure 5.7 Longitudinal channel stress as a function of distance from well edge as obtained by TCAD simulations and after proposed model fitting.

5.2.1 TCAD Experiments

In this set of experiments we use a setup comprising of Tsuprem4 (for simulating fabrication process to generate stress data), and Davinci (for simulating the on current values using Tsuprem4 generated stress data) to generate on current values for different layout configurations of 65nm NMOS and PMOS devices. The TCAD setup is accurately calibrated to the SPICE models for the 65nm technology. Once calibrated, the models are used to generate mobility multipliers which are then used in SPICE based simulation of the devices, and the result is then compared to the TCAD simulation data for each configuration.

We first look at the impact of active area length on device stress. Active area length is one of the most important layout parameters that impacts channel stress quite significantly by increasing the SiGe region around the channel(for PMOS). Figure 5.6 shows the variation of longitudinal channel stress with source/drain length ($L_{s/d}$) (normalized to minimum value of Ls/d) of an isolated 65nm PMOS device as simulated in Tsuprem4 and Davinci TCAD tool. Also shown in the figure is the stress predicted by the proposed model. Stress values are normalized to the value of stress at minimum $L_{s/d}$ for the technology. The figure shows that increasing $L_{s/d}$ increases stress in the channel and this dependence is captured quite accurately by the proposed stress model.

Next we focus on the CESL stress and predict the TCAD results with the proposed model. The most critical layout parameter for CESL is the distance to well edge which serves as an interface between the compressive and the tensile nitride liners. Figure 5.7 shows TCAD based simulation for dependence of PMOS channel stress due to nitride liner as a function of distance from the well edge in the longitudinal direction. As the distance from the well edge increases, so does the compressive stress [7]. The stress

values are normalized to the value at minimum allowed distance from well edge for the 65nm technology.

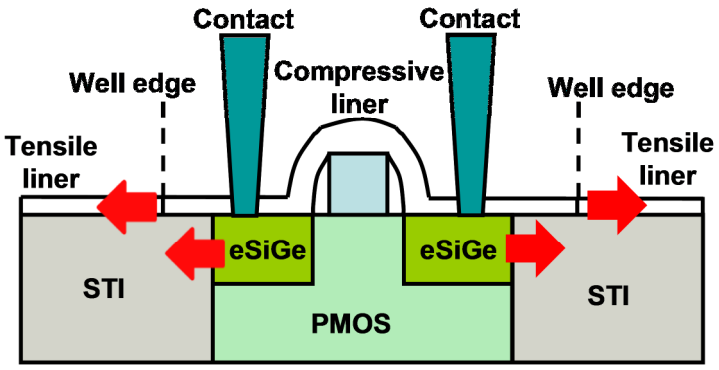


Figure 5.8 Layout permutations in TCAD experiments for model verification.

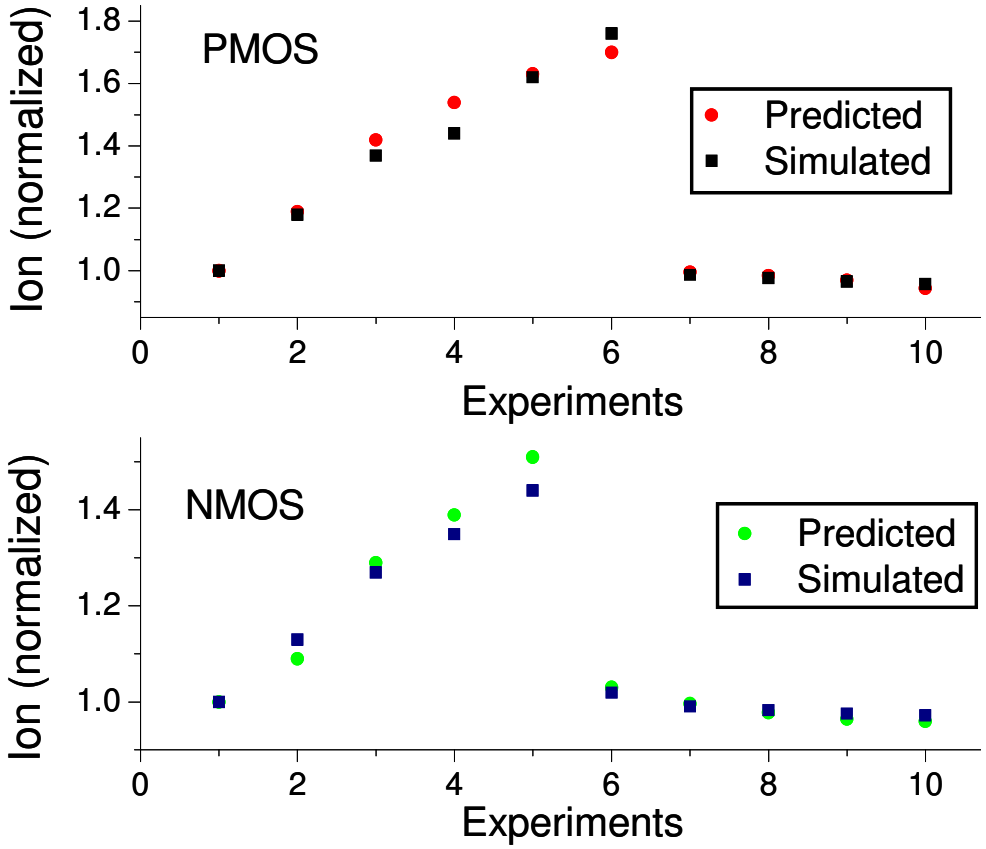


Figure 5.9 Experimental (TCAD) and predicted on current values for NMOS and PMOS devices.

We then analyze on-current predictions for NMOS and PMOS obtained from TCAD simulations and the proposed models. For this we generate a set of layout experiments by varying critical layout parameters. Different combinations of various layout parameters, as shown in Figure 5.8, are varied to generate several different experiments. The first few experiments try to increase the stress based mobility by a combination of increasing the active area length, moving the device away from well edge (in the longitudinal direction), sharing the active area with other devices, etc., while the last few experiments try to decrease the stress based mobility by moving the devices closer to the well edge, introducing more contacts, and decreasing active area length.

Figure 5.9 shows the predicted and simulated on current values (normalized to the on current for isolated NMOS and PMOS devices with one contact) for various TCAD experiments. The proposed model accurately predicts the current values, and the root mean square error in predicted on current value is less than 0.8% for both PMOS and NMOS experiments.

5.2.2 Hardware Experiments

In this set of experiments, the proposed stress models are calibrated and verified using ring oscillator frequency data from an experimental test chip. The ring oscillator data is measured and averaged over several dies to reduce the impact of random and die-to-die systematic variations. For the purpose of calibration, we assume that the frequency of oscillation is directly proportional to average drive current for the ring oscillator, which was confirmed to be a valid assumption using SPICE based simulations of the ring oscillator circuit. Once calibrated, the models are used to calculate impact of stress in terms of mobility multipliers for different ring oscillator layout configurations. Figure 5.10 shows the comparison between the measured frequency data and the predicted

frequency (normalized) for various layout experiments. The plot is divided into three distinct regions corresponding to three different set of layout configurations constituting the hardware experiments. In the nwell experiments, we vary the distance between nwell edge and device in both lateral and longitudinal directions. Since nwell mask is used to define the interface between compressive and tensile nitride liners, such changes have an impact on longitudinal and traverse stress due to the nitride layer. In the second set of experiments, active area layout and length was varied to change the amount of embedded SiGe next to the channel (only for PMOS devices), and the distance between STI edge and gate. In the contact experiments, we varied the number of contacts in the devices constituting the ring oscillator. The plot shows that the models exhibit a very good fit to the hardware data with the root mean square error between simulated and measured data to be only 0.9%.

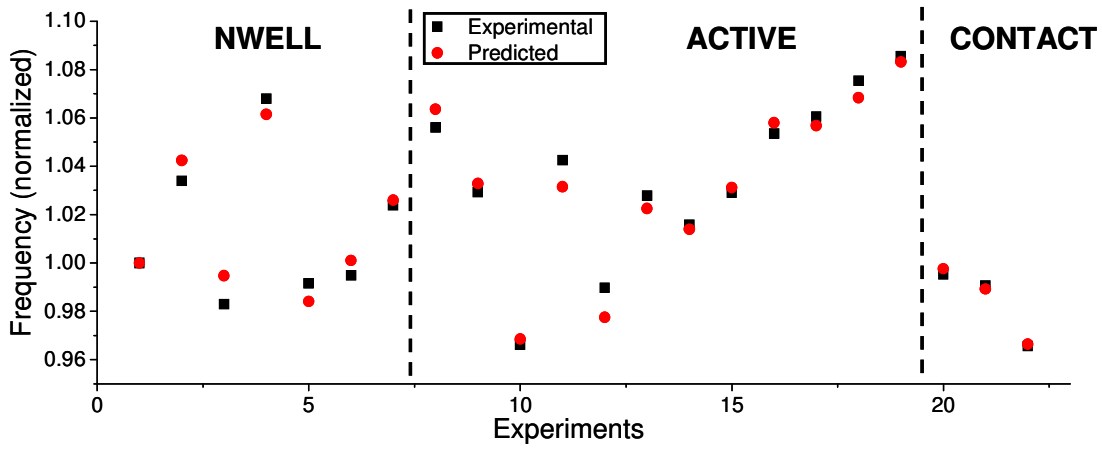


Figure 5.10 Experimental (hardware) and predicted ring oscillator frequencies for different layout configurations.

5.3 Summary

In this chapter, we propose compact, closed-form models for layout dependence of process induced stress. We partition each device channel into segments with equal stress in order to calculate the impact on mobility in terms of mobility multipliers. We extensively verify our models against hardware and TCAD simulation data for a large number of layout permutations. The models enable fast and accurate stress prediction for a device in a given layout environment. The root mean square error in the predicted behavior is observed to be less than 1% for the different experiments, thereby, verifying the accuracy of the models.

Chapter 6
Simultaneous Extraction of Effective Gate Length and Low-field Mobility in Non-uniform Devices

Aggressive CMOS process scaling makes it increasingly difficult to maintain performance and reliability of integrated circuits. At 45nm technology node and below, the minimum feature size is much smaller than the optical wavelength, thereby causing the printed shapes to deviate significantly from the drawn rectilinear shapes. In order to achieve higher performance, modern CMOS processes use special features to induce mechanical stress in the channel of a device, to enhance carrier mobility. However, stress introduced in the channel, and hence carrier mobility, has a strong dependence on device layout and its neighboring features. This results in non rectangular gates (NRG) with non uniform mobility distribution across the device width.

Figure 6.1 shows an nmos device with non rectangular gate, and non uniform mobility distribution. Based on layout induced stress, the gate can be divided into three distinct regions (marked as L-low, M-medium, H-high) with different stress profiles. In case of nmos, larger active area increases the induced tensile stress, and presence of contacts decreases the stress. Hence the stress induced is highest in the region with no contacts and longest active area length, and lowest in the region with shortest active area and contacts. In addition, the figure shows resulting contours for the non rectangular gate.

It is critical to communicate this processing knowledge to the design phase in the form of compact transistor models, for the designer to be able to characterize these non uniformities.

Several prior works have proposed approaches to modeling non rectangular gates, by breaking them up into a set of parallel transistors with constant gate lengths [89, 90]. By summing up the current for each slice, one can obtain drive current for the transistor, and this can be mapped to one value of representative gate length for the device based on a current versus gate length look-up table. However, such a mapping to effective gate length can be done to match either the drive current (Ion) or the leakage (Ioff), and the chosen Effective Gate Length (EGL) mispredicts the other current value. To address this, [91] proposed using device length and width as modeling parameters, and [92] proposed modeling an NRG as two parallel devices with different lengths. Although using two different EGLs can accurately model the device well in its two working states (ON/OFF), this method is inaccurate for intermediate states since it is hard to predict which devices

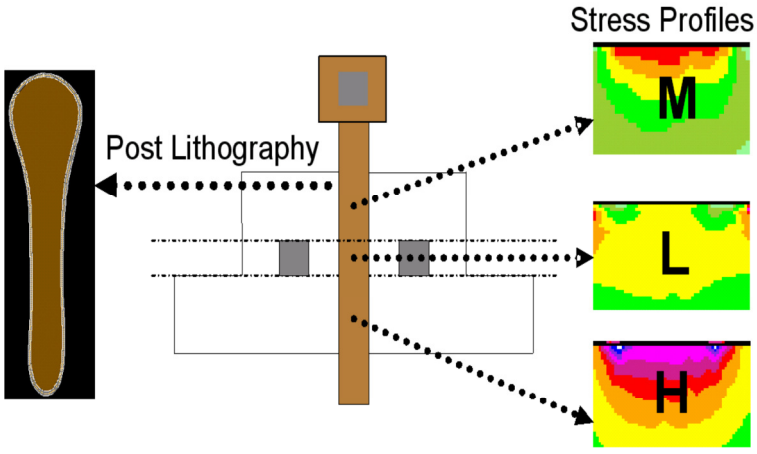


Figure 6.1 NMOS device with non-uniform stress and non-rectangular gate.

are absolutely on or off for complicated cell schematics, and device EGL has a dependence on gate-to-source voltage (V_{gs}). Other improvements have proposed considering threshold voltage variation across device width during slicing [93], and modeling EGL as a function of gate-to-source voltage (V_{gs}) [94].

However, none of these approaches consider the variation of layout induced stress and hence the carrier mobility and device current across the device width. As a result, slice current values considered are incorrect; and hence, the calculated EGL, when used in conjunction with the stress enhanced mobility for the device, mispredicts the drive and leakage current. The problem of modeling non-uniform carrier mobility is similar in principle to the problem of modeling a non-rectangular gate. In this chapter, we propose and EGL type of approach (slicing, summing, and mapping back based on a look-up table) to calculate effective carrier mobility for a device. We also propose simultaneous extraction of EGL and effective carrier mobility (ECM) for each gate, where we consider both gate length and stress enhanced mobility for each slice to calculate drive current and leakage, sum up the slice currents to obtain device drive and leakage currents, and finally map these currents values to EGL and ECM for the device. Since our proposed method derives only one value of EGL and ECM to match both drive current and leakage for a device, it is also expected to be more accurate for intermediate values of V_{gs} as compared to using separate EGLs for on and off states. The rest of the chapter is organized as follows. Background and methodology for concurrent calculation of EGL and ECM is discussed in Section 6.1. Experimental results are discussed in Section 6.2, and Section 6.3 concludes the chapter.

6.1 Background and Proposed Methodology

As discussed in Chapter 3, there are four major sources of stress in a modern CMOS technology: eSiGe (generates compressive stress, used only for PMOS), Shallow Trench Isolation (generates compressive stress, compressive, and tensile nitride liners, and stress memorization technique (SMT). Since the size of these stress sources present in the vicinity of device channel depends on the layout, stress induced in the channel of a device has a very strong dependence on the device layout, and the layout of neighboring devices. The layout dependence of generated stress in the channel is very well studied in the literature, but none of the past works addresses modeling the variation of stress, and hence carrier mobility, across device width. A naïve approach can be to break up the layout into slices based on the layout, such that stress enhanced carrier mobility is almost constant for each slice, and then take a weighted average of slice mobilities based on the slice width. However, such an approach does not result in accurate prediction of drive and leakage current upon using the calculated mobility in a device simulator such as SPICE. As an example, the nmos device shown in Figure 6.1 was simulated using Tsuprem4 process simulator to obtain stress profiles for the three slices, and lithographic simulations were used to obtain non rectangular gate contours. When the EGL obtained by slicing, and the carrier mobility obtained by taking a weighted average of mobilities in different regions (based on width of segments) are used in a SPICE simulation, the error in predicted Ion is 7.4%, and the predicted Ioff shows an error of 62.2%. Thus, there is a need for more accurate modeling of effective mobility.

We observed that the problem of modeling non-uniform carrier mobility is similar in principal to modeling non-rectangular gates. Hence, we can use slicing and summing based approach to calculate one representative value of mobility for the device. Figure

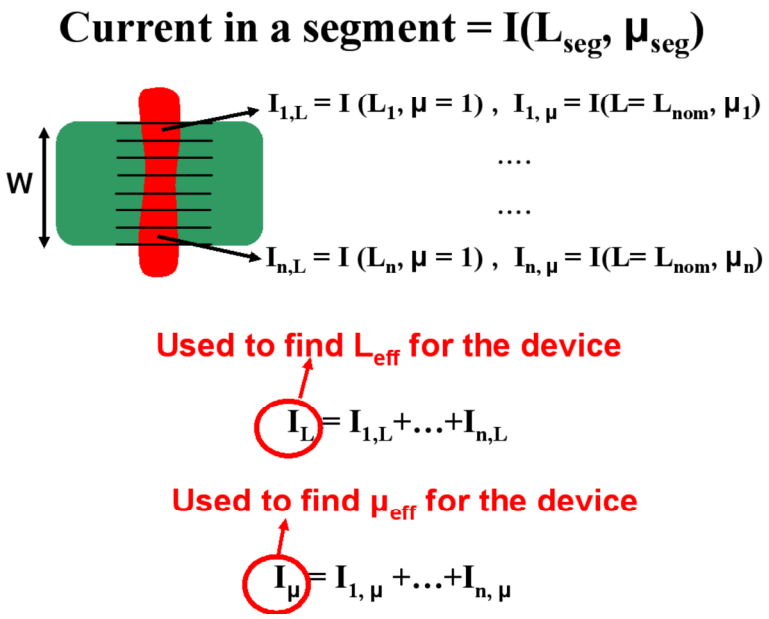


Figure 6.2 Independent calculation of EGL and ECM.

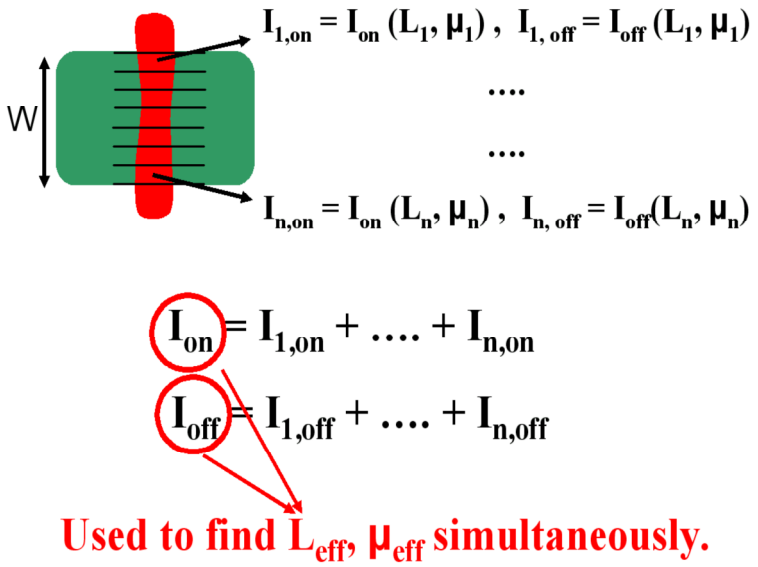


Figure 6.3 Simultaneous calculation of EGL and ECM.

6.2 shows the flow for such an approach to calculate EGL and ECM independently. When EGL and ECM calculations are done independently, a nominal value for the other parameter (mobility or gate length) is assumed for each slice (denotes nominal mobility). Such a calculation, while better than the naïve approach for mobility

calculation, is still inaccurate for two reasons: a) the current calculated for each slice, and hence the device current used for mapping to EGL and ECM is incorrect, as each effect is considered in isolation, and b) while mapping back from current to EGL and ECM, only one of Ion or Ioff can be used for mapping, and the other value is still mispredicted.

In order to enable more accurate simulation, we propose simultaneous extraction of EGL and ECM. Figure 6.3 depicts the proposed approach. It involves concurrently using the lithographic contours, and stress profiles to calculate Ion and Ioff for each slice, summing up the slice currents to obtain drive and leakage currents for the device, and finally mapping it to values of EGL and ECM based on a look-up table based approach. The values of EGL and ECM so obtained, accurately predict device drive current and leakage, when used in conjunction with the SPICE models for the device. Look-up table is prepared by simulating devices at different values of gate length and mobility using the device simulator SPICE for large width devices to counter edge effects, as suggested in [90]. Slice current calculation also takes into account the variation of threshold voltage,as suggested in [93]. Such a simultaneous extraction ensures that there are two parameters to model (EGL and ECM), and hence, both on and off currents can be matched. By eliminating both the accuracy related problems with independent calculation of EGL and ECM, simultaneous extraction ensures that the resulting simulations are much closer to TCAD/hardware data. Fig. 6.4 shows flowcharts for the independent and simultaneous extraction of EGL and ECM to further explain the methodology.

Figure 6.5 shows the distribution of normalized Ion for nmos, as a function of mobility and gate length. Gate length of 0 denotes nominal gate length, and mobility of 1

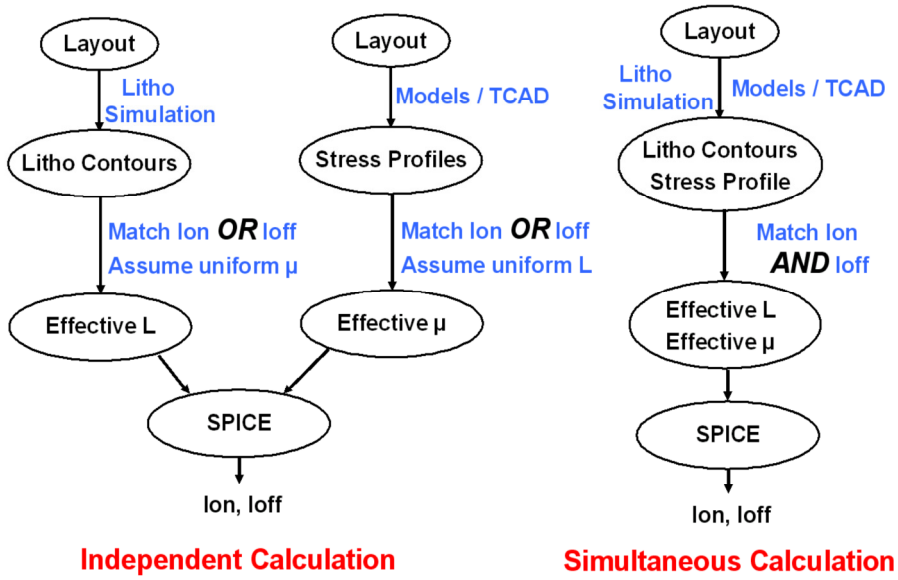


Figure 6.4 Flowcharts depicting independent and simultaneous extraction of EGL and ECM.

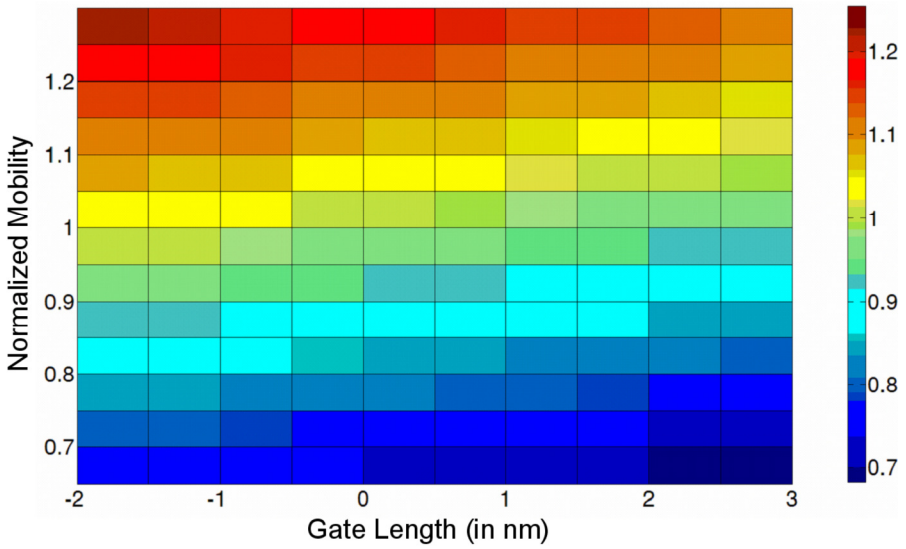


Figure 6.5 Ion variation with mobility and gate length for an nmos.

denotes mobility for an isolated minimum sized device. Mobility distribution across the channel can be summed up according to the following equation:

$$\frac{\mu_0}{\mu_e} = \frac{1}{L}\int_0^L \frac{\mu_0}{\mu(x)}\,dx \qquad (1)$$

where μ_0 is the nominal unstrained mobility, μ_e is the stress enhanced mobility, and L is the channel length. The mobility values so generated, when used in SPICE, generate drive and leakage current values which are in close agreement with the TCAD data. The TCAD setup comprises of Tsuprem4 [82] for simulating the device fabrication process and generating the stress distribution, and Davinci [81] for device simulation based on the stress profile imported from Tsuprem4 to generate final current and mobility values. TCAD setup was closely matched to the 45nm cell library used for experiments. Sources of mechanical stress considered are: eSiGe, compressive nitride liner, and Shallow Trench Isolation (STI), for pmos transistors, and STI, and tensile nitride liner, for nmos transistors. Mapping from device Ion and Ioff to EGL and ECM involves a simple search of the Ion and Ioff distributions as a function of gate length and mobility. The following section discusses device and gate level results to verify the accuracy of the proposed simultaneous extraction approach.

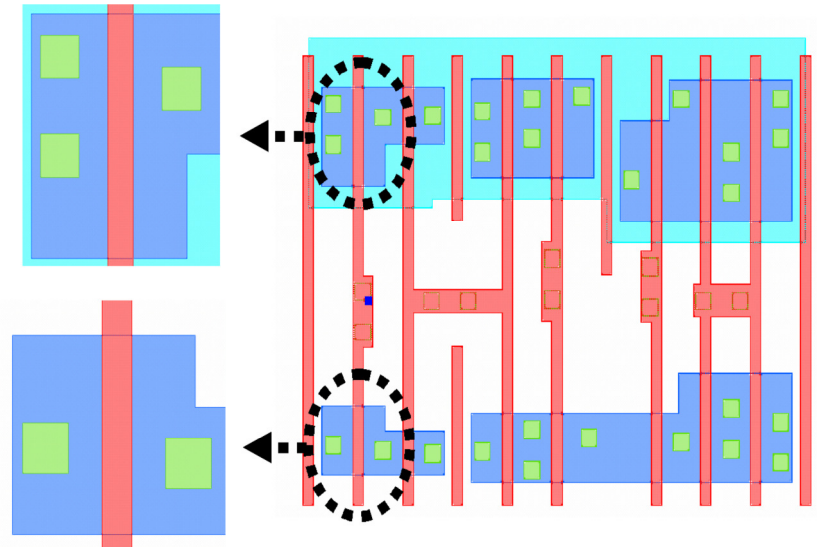


Figure 6.6 MUX layout showing two randomly selected devices.

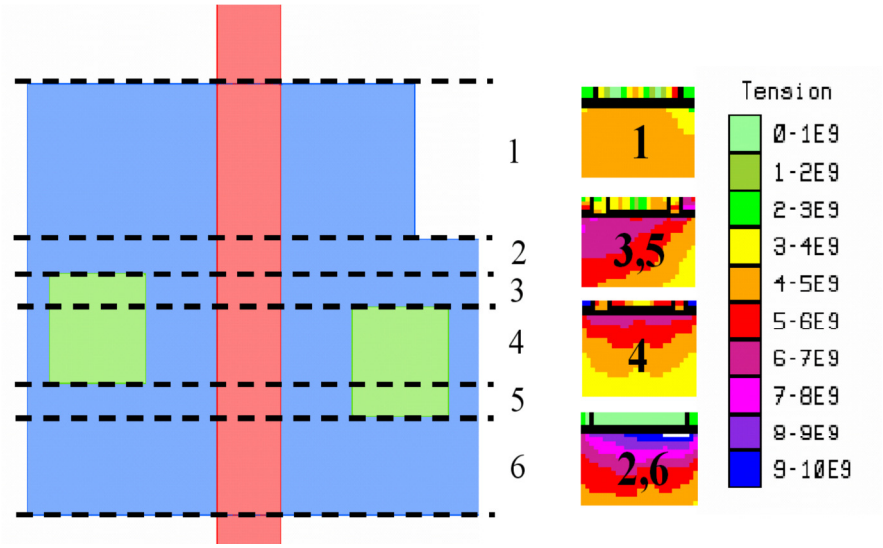


Figure 6.7 Stress based partitioning of the NMOS gate.

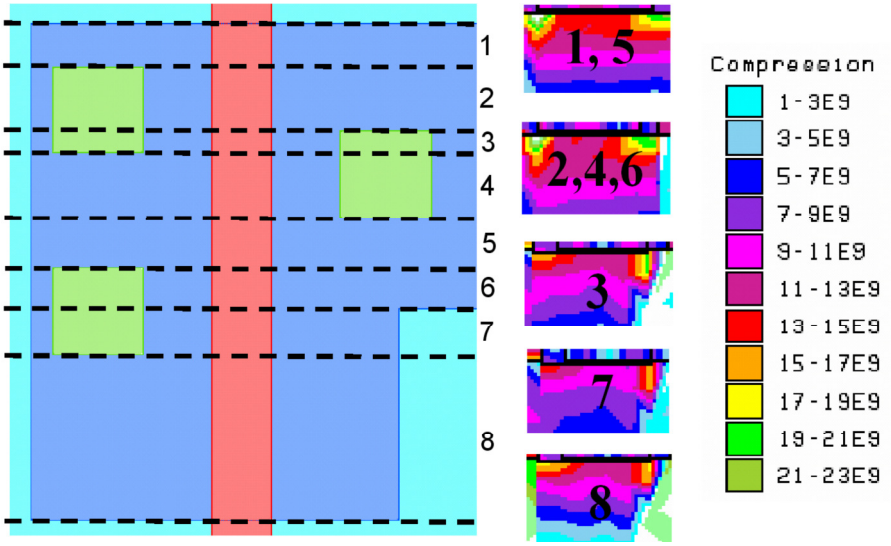


Figure 6.8 Stress based partitioning of the PMOS gate.

6.2 Experimental Results

This section verifies the effectiveness of the proposed technique for simultaneous EGL and ECM extraction, by analyzing device level and gate level results for error in predicted drive current and leakage by using the naïve approach of taking weighted average of mobility, and for independent EGL and ECM correction. All the errors are calculated with respect to the golden values obtained by running TCAD simulations for

stress and lithography simulation for gate length for each slice, and summing up the slice currents, while also taking into account the variation in threshold voltage in accordance with [93]. As discussed in section 6.1, we consider all the layout dependent sources of stress in our simulations: embedded SiGe, tensile/compressive nitride liner, and Shallow Trench Isolation (STI).

Figure 6.6 shows the layout of a 2-input MUX, and provides a closer view of one nmos and one pmos device chosen from the layout. Figures 6.7 and 6.8, show the stress based partitioning of each gate along with the corresponding stress, for the nmos and the pmos device respectively. The error in predicted currents, when calculating EGL (for a fixed nominal value of mobility for each slice) and taking a weighted average of mobility was found to be 6.4% for Ion, and 56.3% for Ioff, for the pmos device. The corresponding error in the case of nmos was found to be 5.9% and 58.1%, in Ion and Ioff, respectively. Next, we calculated the EGL and ECM independently, and the error observed for pmos device were 4.1% in Ion, and 38.2% in Ioff. In case of NMOS device, the errors observed were 4.0% and 35.2%, for Ion and Ioff values, respectively. For both nmos and pmos devices, independent calculation of EGL and ECM leads to less error in predicted currents, as compared to the naïve approach of calculating EGL and calculating a weighted average of mobility across the device width. However, the errors are still very high (particularly in Ioff), and hence the need for a more accurate prediction by concurrent calculation of EGL and ECM. No error is observed in the case of the proposed concurrent calculation, because the correct values of Ion and Ioff are used for the mapping back to EGL and ECM. In all the independent calculations of EGL and ECM, Ion was matched while mapping back to effective values.

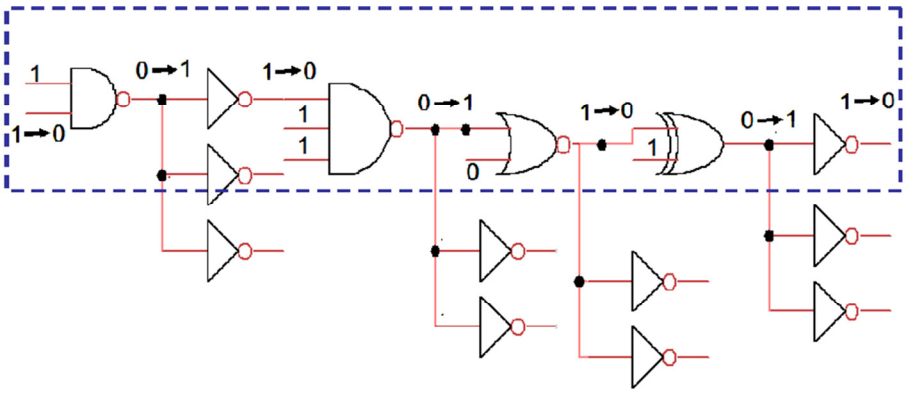


Figure 6.9 Circuit path showing an input transition.

Table 6.1 Delay and leakage errors for independent calculation of EGL and ECM (relative to simultaneous extraction)

Cell Name	Independent calculation of EGL/ECM				Naïve averaging of Mobility			
	Match Ion		Match Ioff		Match Ion		Match Ioff	
	Delay	Leakage	Delay	Leakage	Delay	Leakage	Delay	Leakage
Inverter	4.80%	42.10%	5.40%	31.40%	6.50 %	57.40%	6.80 %	51.20%
2-input NAND	4.90%	29.10%	5.10%	27.60%	6.70 %	54.30%	6.90 %	52.30%
3-input NOR	4.40%	31.30%	4.70%	28.70%	6.20 %	55.60%	6.70 %	51.80%
Average	**4.70%**	**34.20%**	**5.10%**	**29.20%**	**6.50 %**	**55.80%**	**6.80 %**	**51.80%**

Next, we present gate level results to further establish the effectiveness of proposed simultaneous extraction approach. Errors in average delay and gate leakage are calculated for the naïve approach of calculating EGL (for a fixed nominal value of mobility for each slice) and taking a weighted average of mobility, and for independent calculation of EGL and ECM, as compared to the concurrent calculation. As discussed earlier, in case of independent EGL and ECM calculations, mapping back from current to EGL and ECM can be done to match either Ion or Ioff. We analyze the errors in both the

scenarios. Table 6.1 shows the gate level results for error in delay and gate leakage, for the two different ways of independently calculating EGL and ECM (matching Ion, and matching Ioff), and for the naïve approach. While matching Ion, average percentage error observed is 4.7% for delay and 34.2% for leakage. The corresponding average errors for matching Ioff are 5.1% and 29.2% for Ion and Ioff, respectively. For the naïve approach, average error observed while matching Ion is 6.5% in delay, and 55.8% in leakage, and matching Ioff results in 6.8% and 51.8% errors in leakage and delay, respectively.

We also simulated a sample circuit path to analyze the error in circuit level delay for using independent calculation of EGL and ECM, as well as the naïve approach. We simulated the delay of the circuit path shown in Figure 6.9, for the input transition shown in the figure. The naïve averaging approach results in an error of 4.6% in the predicted delay, while independent EGL and ECM calculation resulted in an error of 3.2%. As mentioned earlier, no error is incurred by our proposed simultaneous extraction method. This set of results clearly establishes that the proposed simultaneous calculation approach is considerably more accurate as compared to the independent EGL and ECM calculation.

Finally, we present device level simulation results to demonstrate the accuracy of proposed approach in simulating current for intermediate values of V_{gs}. Using two different EGLs can accurately model the device well in its two working states (ON/OFF), while assuming uniform mobility across the device width. However, even for constant device mobility, this method is inaccurate for intermediate states since it is hard to predict which devices are absolutely on or off for complicated cell schematics, and device EGL has a dependence on gate-to-source voltage (V_{gs}). As a result, some works proposed

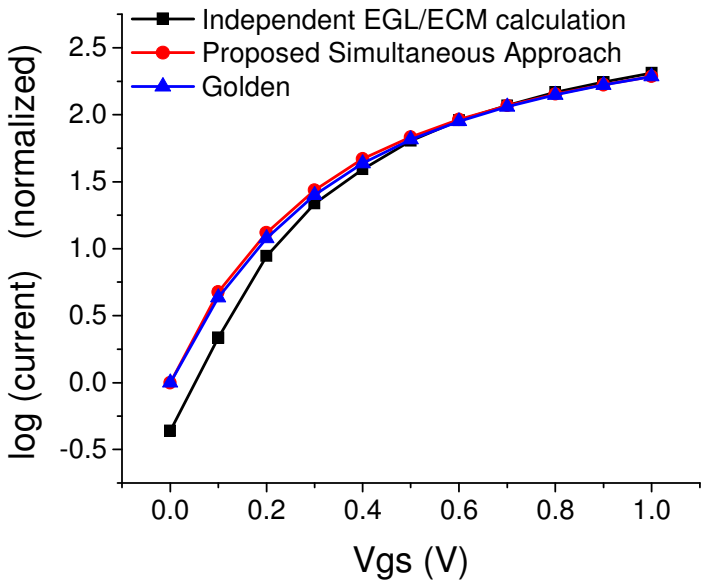


Figure 6.10 Drain current as a function of V_{gs} for V_{ds} =V_{DD} for an NMOS device.

modeling EGL as a function of V_{gs} for more accurate simulation, but this incurred additional characterization overhead. Our approach derives only one combination of EGL and ECM to match both drive current and leakage. Since there is no switching from one set of values to another for ON/OFF states (as in the case of separate EGLs), the proposed approach is expected to simulate the current for intermediate states with a good accuracy, thereby potentially eliminating the need to store effective values as a function of V_{gs} to enable more accurate simulation of intermediate Vgs values.

The proposed simultaneous extraction approach is shown to accurately simulate the intermediate states, as illustrated in Figure 6.10. It shows log of normalized current (normalized to the value at V_{gs} = 0) as a function of V_{gs}, at V_{ds} = V_{DD}, for the nmos device chosen from the minimum sized inverter. As expected, our proposed method provides a good fit to the "golden" I-V curve (obtained by slice based summing currents for each voltage point), and the root mean square error observed is less than 1% of the

saturation current. Also, plotted is the I-V curve for independent calculation of EGL and ECM (while matching on current), and as expected it shows large deviation from the "golden" curve. Root mean square error for the independent calculation is ~3.3% of the "golden" saturation current. Since, our proposed method provides a good fit for intermediate values of V_{gs}, it is also ideal for simulating low voltage circuits which operate at low values of V_{gs} and V_{ds}, and are being extensively used for low power applications.

6.3 Summary

In this chapter, we propose simultaneous extraction of effective gate length (EGL) and effective carrier mobility (ECM) for a device, where we break a gate into parallel slices, consider both gate length and stress enhanced mobility for each slice to calculate drive current and leakage, sum up the slice currents to obtain device drive and leakage currents, and finally map these currents values to EGL and ECM for the device. We performed device and gate level simulations to establish the need for such an approach, by calculating the error in predicted on and off currents, as well as the average delay and leakage, for a naïve approach that does not consider mobility variation while calculating EGL, and another approach which calculates EGL and ECM independently. Gate level results for independent calculation of EGL and ECM show an average error in predicted delay of 4.7% and that in predicted leakage of 34.2%, thereby confirming that the simultaneous extraction is considerably more accurate as compared to the other approaches. The proposed approach also provides good accuracy in predicting current values for intermediate values of V_{gs}.

Chapter 7
Analyzing electrical effects of RTA-driven local anneal temperature variation

The semiconductor industry faces significant challenges as it strives to extend Moore's law through aggressive process scaling. The most important challenge lies in maximizing the device on-current while suppressing the leakage. Progress in this goal is driven by advances in the engineering of ultra-thin gate insulators, high-mobility channels, ultra-shallow junctions and low-resistance contacts. RTP (Rapid Thermal Processing) is a key process step in providing the essential capabilities for both process and material development on this front [95]. Figure 7.1 illustrates the important role of RTP in an advanced fabrication process.

RTP is employed in fabrication steps that require the wafer to be heated and cooled quickly within a low thermal budget (a small value of temperature-time product) [96]. For example, shallow p+-n junctions are difficult to fabricate due to high Boron diffusivity and formation of Boron channeling tail. Rapid Thermal Anneal (RTA) has been successfully used to address this problem [97, 98]. RTA typically involves a spike anneal, where the wafer is ramped to a high temperature and then allowed to cool immediately [99]. Spike annealing allows the use of high temperatures for higher dopant activation and ion implantation damage annealing, while restricting dopant diffusion by minimizing effective anneal time (low thermal budget).

However, decrease in anneal times to achieve shallower junctions for more aggressively scaled transistors has resulted in a reduction of the characteristic thermal length (the length over which thermal equilibrium can be reached for a given time) to dimensions less than the typical die size [100]. In addition, since radiation is the primary source of heat transfer, the layout pattern dependence of optical properties (emissivity, reflectivity, etc.) also affects the amount of heat absorbed and hence the local anneal temperature of a region in the layout [101]. This leads to variation in the local anneal temperature across the chip, which in turn affects transistor performance and leakage [102].

Higher local anneal temperature drives the junctions both longitudinally and vertically, and causes a higher activation of dopants. This results in lower threshold voltage (V_{th}) by a combination of increased short channel effects and compensation of halo doping. Also, higher dopant activation and increased gate overlap of source and drain together result in lower extrinsic transistor resistance (R_{ext}). This correlation in the

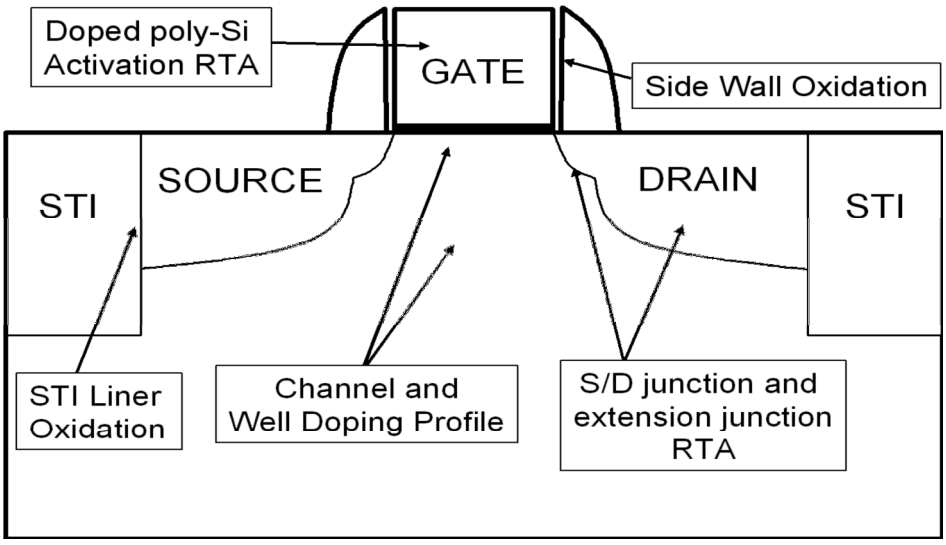


Figure 7.1 Role of RTP in advanced fabrication process.

across-chip variation of V_{th} and R_{ext} results in a pronounced effect on the drive current and leakage. Since the characteristic thermal length is quite large, entire circuit blocks may be systematically faster or slower depending on their position in the layout. Thus, neglecting RTA-induced variations can result in significant misinterpretation of circuit timing. Experiments in [103] showed that ring oscillator frequency can vary by as much as ~20% based on position in the die due to local anneal temperature variation.

There has been recent work focused on obtaining a rigorous solution to local anneal temperature variation and analyzing its effect on rapid thermal processes such as oxidation [104]. However this analysis was very involved and has not been extended to a framework that can be used efficiently for any given layout. To the best of our knowledge, no work in the past has addressed the problem of obtaining a quick and efficient solution to anneal temperature distribution, which can then be used for RTA-aware timing analysis. This chapter proposes a new RTA-aware timing framework that embodies transistor-level models for anneal temperature sensitivity, to incorporate RTA-induced temperature variation into traditional timing/leakage analysis. This is achieved by modeling the dependence of drive current (Ion), and leakage current (Ioff) on local anneal temperature, using device-level TCAD simulations. Next, the wafer is meshed into a rectangular grid, and local anneal temperature is solved for using the finite difference method. This involves discretization of the second spatial derivative of temperature as a finite difference in rectangular coordinates. Once the local anneal temperatures are known, Ion and Ioff multipliers enable accurate timing and leakage analysis for a device (based on its position in the wafer). We also discuss techniques to minimize the electrical

impact of anneal temperature variation, and analyze each of the proposed techniques for effectiveness and cost of implementation. We examine filler insertion, film deposition, and gate length biasing, and conclude that a hybrid approach, comprising of gate length

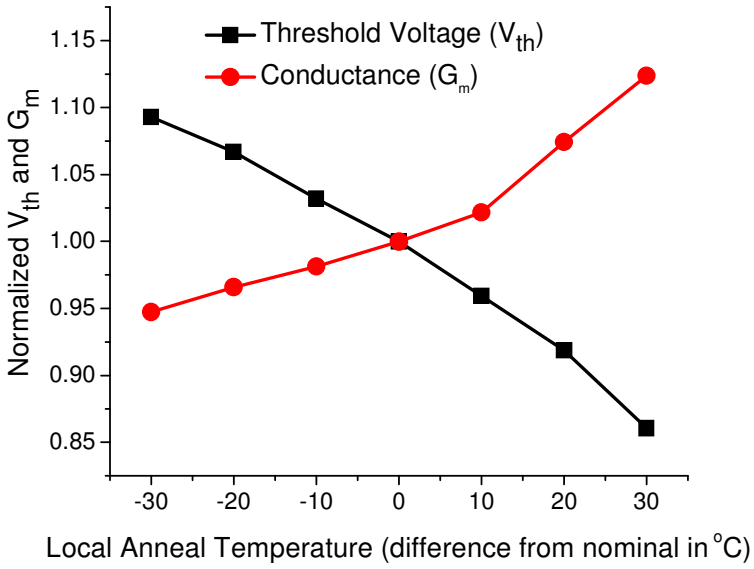


Figure 7.2 PMOS V_{th} and G_m variation with anneal temperature.

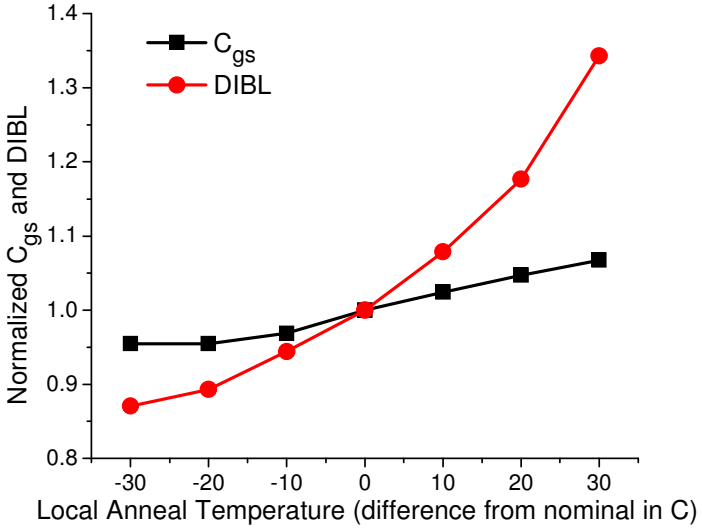


Figure 7.3 PMOS C_{gs} and DIBL variation with anneal temperature.

biasing followed by filler insertion, provides the best solution.

The rest of the chapter is organized as follows. Section 7.1 motivates and provides background for this work. Section 7.2 presents the methodology and simulation flow for RTA-aware timing analysis. Section 7.3 describes results for chip-level anneal temperature simulation, and the corresponding distributions for on and off currents. Section 7.4 discusses techniques to minimize the impact of anneal temperature variation, and Section 7.5 concludes the chapter.

7.1 Device Level Analysis of Electrical Properties

As discussed earlier in this chapter, local anneal temperature varies with position on a die. This variation in turn affects device properties such as vertical and longitudinal position of the S/D junctions, dopant activation, and compensation of the halo doping. Higher local anneal temperature results in lower V_{th} and R_{ext} due to a combination of these effects, and hence faster devices. Prior works have reported the intra-die variation of R_{ext} and V_{th} to be highly correlated [102, 103], and this makes the strength of this variation particularly strong.

Figure 7.2 illustrates the temperature dependence of threshold voltage (V_{th}) and conductance (G_m) for a 45nm PMOS device. As expected, V_{th} decreases, while G_m increases with a rise in local anneal temperature. Figure 7.3 shows the effect of temperature variation on gate to source capacitance (C_{gs}) and drain induced barrier lowering (DIBL). Both C_{gs} and DIBL increase as temperature rises since gate/drain and gate/source overlap lengths grow due to increased dopant diffusion. To examine the effect on performance and leakage, drive current (Ion) and leakage current (Ioff) are also plotted as a function of temperature in Figure 7.4. In all of these plots, 0 denotes nominal anneal temperature, while other temperature values signify a deviation from the nominal

value. All the other values plotted as a function of temperature, are normalized to the corresponding values at nominal anneal temperature. The plots clearly show a significant dependence of performance and leakage on local anneal temperature, due to the strong dependencies of V_{th}, G_m, C_{gs}, and DIBL. A modest increase of 20°C over the nominal anneal temperature yields an 11.8% change in Ion and ~4X change in Ioff. Decreasing the anneal temperature by 20°C results in 7.4% and 2.6X decreases in Ion and Ioff, respectively.

Figure 7.5 shows V_{th}, G_m, and DIBL as a function of anneal temperature for an NMOS device. A 20°C increase in anneal temperature increases Ion by 9.4% and Ioff by ~5X. Previous work [104] reported anneal temperature differences of up to 50°C between highest and lowest anneal temperature on the wafer, for regular patterns. Such a large

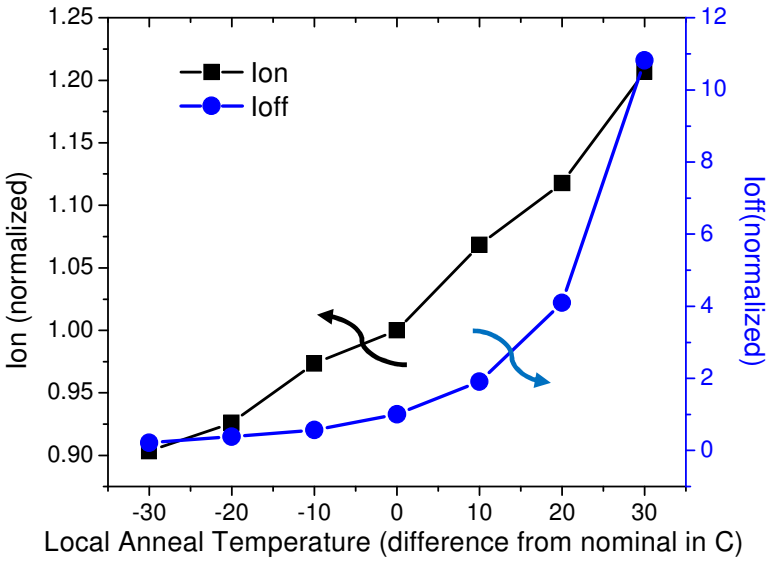


Figure 7.4 PMOS Ion and Ioff variation with anneal temperature.

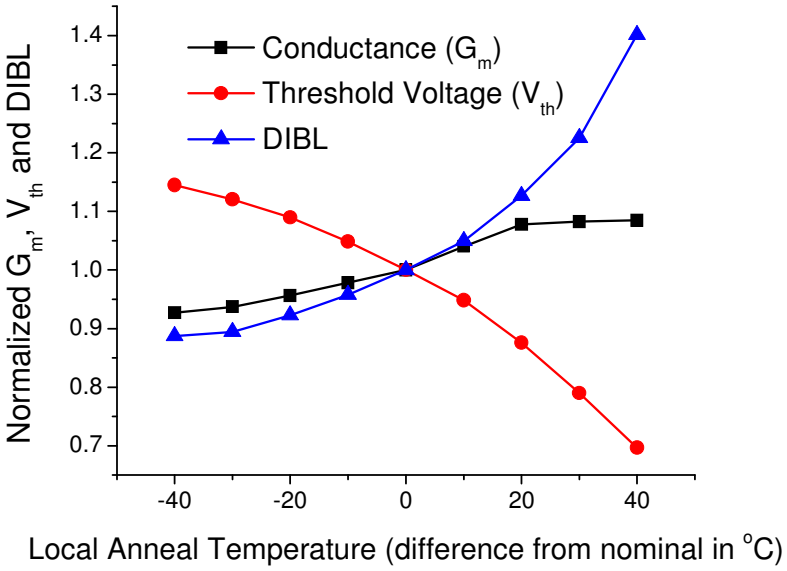


Figure 7.5 V_{th}, G_m and DIBL variation with anneal temperature for NMOS device.

difference can cause significant variation in performance and leakage that must be accounted for while analyzing performance and leakage. The plots also suggest that anneal temperature dependence of Ion and Ioff can be modeled as polynomial functions of temperature. The next section develops these models, discusses the methodology for calculating local anneal temperature, and describes a simulation flow for RTA-aware timing analysis.

7.2 Simulation Methodology

There are two major components of the RTA-aware simulation framework: models to capture the impact of local anneal temperature on leakage and drive currents, and a methodology to solve for local anneal temperature. Figure 7.6 illustrates the simulation flow. Device-level TCAD simulations are performed using TSUPREM4 to model the dependence of leakage current and drive current on local anneal temperature, in terms of on and off current multipliers. The layout is meshed into a rectangular grid

and the finite difference method is used to solve for temperature as a function of position on the wafer. This temperature distribution is used to determine local anneal temperature for a gate using the layout definition file to determine its position. Once the local anneal temperature is known, the Ion and Ioff models are used to modify the values for leakage and delay (read from the characterized library) at the time of final simulation. In this work, we assume that gate delay is inversely proportional to Ion, and hence we scale the gate delay by the inverse of the Ion multiplier. The remainder of this section describes the development of these models in detail, along with the method to solve for local anneal temperature, and concludes by discussing the RTA-aware simulation flow.

7.2.1 Modeling performance and leakage variation with local anneal temperature variation

As discussed earlier, variation in local anneal temperature affects V_{th} and R_{ext} by a combination of changes in dopant activation, effective channel length, halo doping, and gate overlap of source and drain. There are two approaches to consider these effects during timing and leakage analysis:

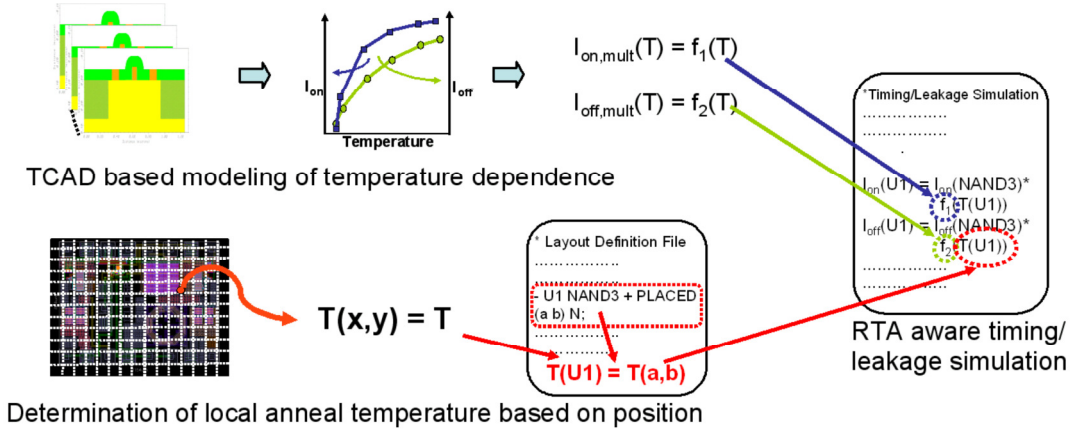


Figure 7.6 RTA aware performance/leakage analysis flow.

- Model temperature effects on basic device properties (Vth and Rext or doping profile, effective channel length, and halo doping), then use these models to modify the HSPICE model files and characterize standard cell library at different anneal temperatures. For a given temperature, interpolate between the known values from library files characterized at certain fixed values of local anneal temperature.
- Model temperature effects in terms of Ion and Ioff multipliers as a function of local anneal temperature. Characterize the standard cell library once for nominal anneal temperature, and superimpose these multipliers at the time of timing/leakage analysis when reading in the characterized values for a given standard cell.

Since the focus of this work is to perform accurate delay and leakage simulation, the second approach provides more accurate results than relying on interpolation. Also, the second approach is easier to implement and has a lower characterization cost than the first approach. We therefore use the second approach to construct the simulation framework in this work.

We use TCAD (TSUPREM4 [82] for process simulation, and MEDICI [81] for device simulation) for device-level simulation and I-V characteristics of the devices are matched to the 45nm technology used for this work. Figure 7.7 shows normalized values of Ion and Ioff (normalized to the values at nominal anneal temperature) plotted against the anneal temperature for a PMOS device, and Figure 7.8 shows the corresponding plot for an NMOS device. Also shown are our polynomial models for Ion and Ioff as a function of temperature. Since these models are fitted to the normalized values, they

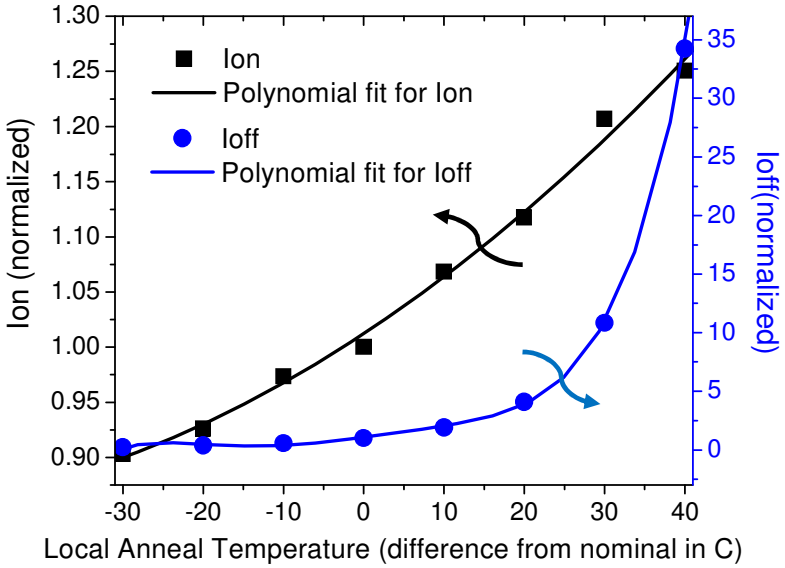


Figure 7.7 PMOS models for Ion and Ioff variation with temperature.

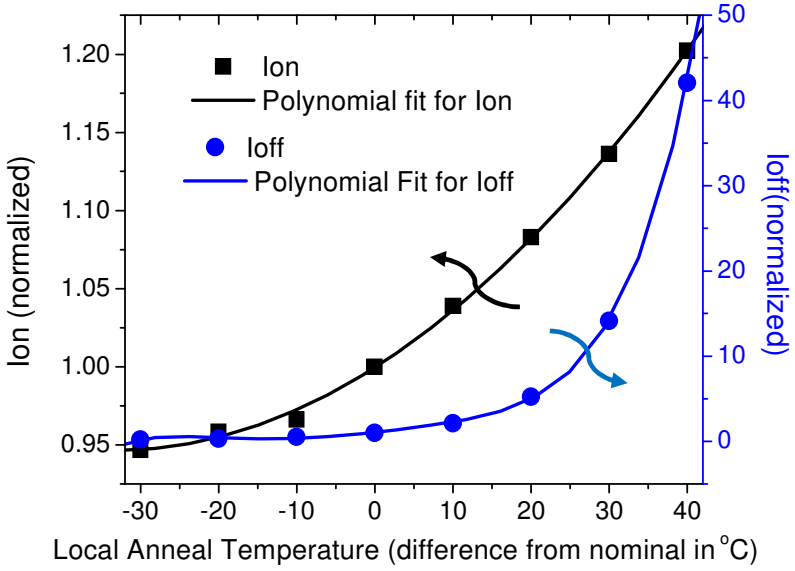


Figure 7.8 NMOS models for Ion and Ioff variation with temperature.

represent multipliers by which nominal Ion and Ioff should be multiplied to obtain their correct values for a given temperature. The exact dependency of leakage and drive

currents on anneal temperature is a complex non-linear function. However, we see that device currents can be modeled with good accuracy using polynomial functions of the anneal temperature. Using MATLAB for accurate curve fitting, we observe that a quadratic polynomial can be used to model the Ion dependence with good accuracy, while a polynomial of degree 5 was needed to model the off current dependence, for both NMOS and PMOS. It is evident from the figures that polynomials predict the Ion and Ioff dependence accurately.

7.2.2 Chip level anneal temperature variation analysis

Solving for local anneal temperature involves setting up differential equations describing the heat flow, and then discretizing the chip area into rectangular grid and approximating the partial derivatives. Since radiation is the dominant mechanism of heat transfer, we assume that conduction and convection have negligible contribution. We also assume that the temperature variation across the thickness of the chip is negligible compared to the variations in the plane of the wafer. This assumption is based on the fact that the typical duration of RTA processes (on the order of a few seconds) is large enough to allow the temperature distribution across the thickness of the wafer to reach study state. These assumptions are valid, and have been used in the past to make accurate predictions for RTP processes [104]. This allows us to solve for the temperature in the plane of the wafer (x-y plane), and in particular we can assume the partial derivative with respect to the wafer thickness (z) to be zero.

In the steady state, we can write the heat balance equation as Poisson's equation

$$\ddot{d}k\nabla^2\mathrm{T}(\mathrm{x},\mathrm{y}) = -\left(\mathrm{P_{ABS}}(\mathrm{x},\mathrm{y}) - \mathrm{P_{EMI}}(\mathrm{x},\mathrm{y})\right) \quad (1)$$

where d is the wafer thickness, k is the in plane thermal conductivity, T(x,y) is the temperature distribution in the wafer plane, PABS is the radiative power absorbed per unit area, and PEMI is the radiative power per unit area emitted by the wafer [104]. The emitted and absorbed power perms vary with position on the wafer due to the layout pattern and hence position dependence of optical properties (emissivity, absorptivity, etc.), and can be expressed as

$$\mathrm{P_{ABS}(x,y)} = \alpha(x,y,T)P_{incident} \tag{2}$$

$$\mathrm{P_{EMI}(x,y)} = \varepsilon(x,y,T)\sigma T^4(x,y) \tag{3}$$

where $P_{incident}$ is the heater power per unit area incident on the wafer, σ is the Stefan-Boltzmann constant, $\alpha(x,y,T)$ is the position and temperature dependent effective total absorptivity, and $\varepsilon(x,y,T)$ is the effective wafer emissivity. The effective emissivity and absorptivity depend on optical properties of the layer structure, and upon the temperature of the region. They also depend on the wavelength of the radiations incident on the wafer, which in turn depends on the heater temperature and material (known for a given fabrication process).

The steady-state heat balance equation (1) can be discretized by writing the second derivative of temperature as a finite difference term. We use a grid based approach, where we discretize the wafer surface into a rectangular grid structure with lengths Δx, and Δy in the x and y direction respectively. This gives us one discritized node equation for each node on the grid. The spatial derivative of temperature in equation (1) can be written as

$$\nabla^2 T(x,y) = \frac{\partial^2}{\partial x}T(x,y) + \frac{\partial^2}{\partial y}T(x,y) \tag{4}$$

and we can discritize the individual second spatial derivative terms. Let $T_{a,b}$ represent the anneal temperature at a node with co-ordinates (a,b). Now, in the x direction, we can write

$$\frac{\partial^2 T_{a,b}}{\partial x} \approx \frac{\frac{T_{a-1,b}-T_{a,b}}{\Delta x} - \frac{T_{a,b}-T_{a+1,b}}{\Delta x}}{\Delta x} \quad (5)$$

$$\frac{\partial T_{a,b}}{\partial x} \approx \frac{T_{a-1,b}-2T_{a,b}+T_{a+1,b}}{(\Delta x)^2} \quad (6)$$

Discretizing the heat balance equation yields the following system of non-linear equations (one equation for each node on the grid):

$$\frac{T_{a-1,b}-2T_{a,b}+T_{a+1,b}}{(\Delta x)^2} + \frac{T_{a,b-1}-2T_{a,b}+T_{a,b+1}}{(\Delta y)^2} = -\left(\alpha_{a,b}\left(T_{a,b}\right)P_{incident} - \varepsilon_{a,b}\left(T_{a,b}\right)\sigma {T^4}_{a,b}\right) \quad (7)$$

where $\alpha_{a,b}\left(T_{a,b}\right)$, and $\varepsilon_{a,b}\left(T_{a,b}\right)$ are the position-dependent functions of temperature, describing the average behavior of absorptivity and emissivity in a rectangle with sides Δx, and Δy, centered at the node point. The optical properties at a given point depend on the layer structure of the wafer. There are five different kinds of layer structures possible at the time of RTA: N+ source/drain, P+ source/drain, polysilicon over isolation, polysilicon over transistor, and shallow trench isolation (STI). We use a tool called Rad-Pro [105] to calculate the temperature dependent functions of emissivity and absorptivity for each of these configurations. Interfering optical reflections at the interface of different layers cause a dependence on the wavelength and the exact layer structure. Rad-Pro uses universally accepted and extensively calibrated models to predict the directional, spectral, and temperature dependence of radiative properties for multilayer structures consisting of materials like silicon (doped/undoped), silicon dioxide, silicon nitride, and polysilicon.

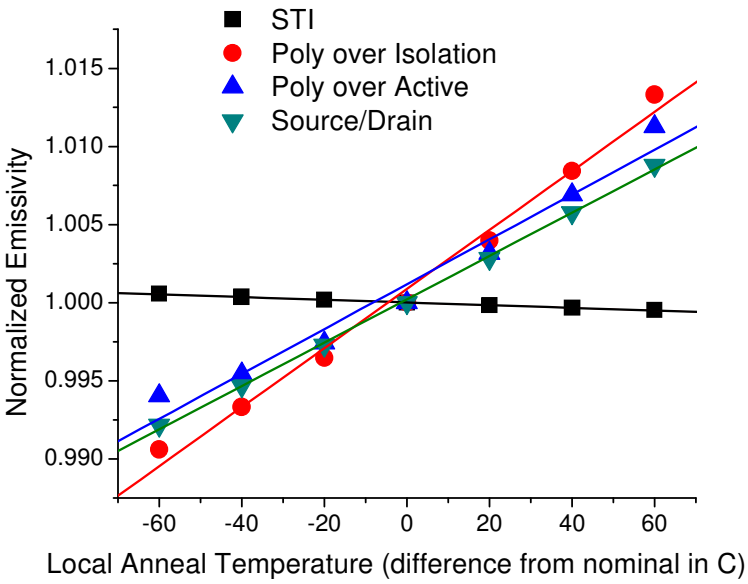


Figure 7.9 Model for emissivity variation with anneal temperature.

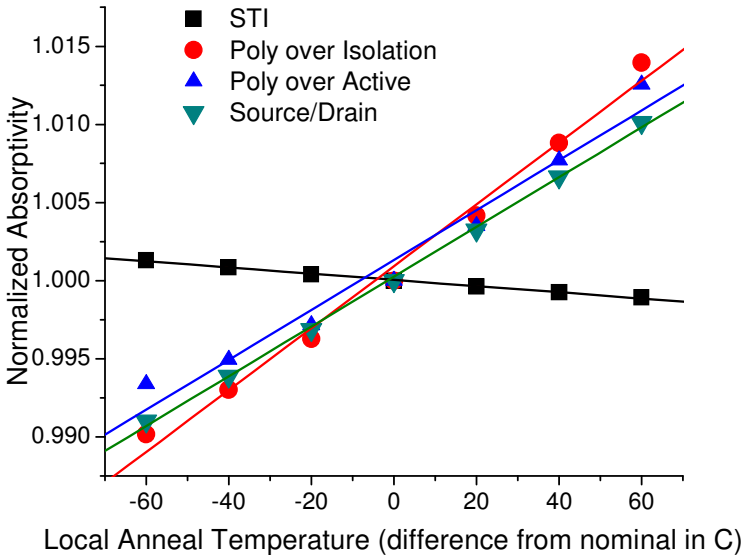


Figure 7.10 Model for absorptivity variation with anneal temperature.

Figures 7.9 and 7.10 show the plots of normalized emissivity and absorptivity, respectively, as a function of temperature for the different layer structures. In the figures, 0 represents nominal anneal temperature for the 45nm technology used for this project,

and other temperature points are expressed as difference from nominal. We observe that the variation of these optical properties with temperature can be modeled accurately by using linear functions of temperature. Hence they are modeled in the form a(bT+c), where a is the value of the property at nominal anneal temperature, and b and c are coefficients modeling the linear dependence on temperature. For example, the emissivity for STI is expressed as $\varepsilon_{STI}(b_{STI}T+c_{STI})$, where ε_{STI} is the emissivity for STI at nominal anneal temperature. Figures 9 and 10 also show the corresponding linear fit for each of the functions.

The Calibre layout verification tool [106] calculates relative densities of different layer types for each node point, in a rectangle with sides Δx, and Δy, centered at the node point. These density values are used to calculate density based weighted average of the linear fit coefficients, to yield final average temperature-dependent functions for absorptivity, and emissivity. Finally, these simultaneous non-linear equations are solved for using MATLAB to yield chip level temperature maps for local anneal temperature. Local anneal temperature at any point can be determined by interpolating between the values at node points for the rectangle on the grid that contains the point. These temperature values are used in conjunction with TCAD-based models for accurate performance and leakage analysis, by changing the delay/leakage based on the position of the gate in the layout. The next section discusses the result of such an analysis for several test chips.

7.3 Experimental Results

To demonstrate the importance of anneal temperature variation aware analysis, the flow described in the previous section is applied to two 45nm test chips, and one 65nm test chip. There are 40X40 rectangular grid cells of equal size located on the top

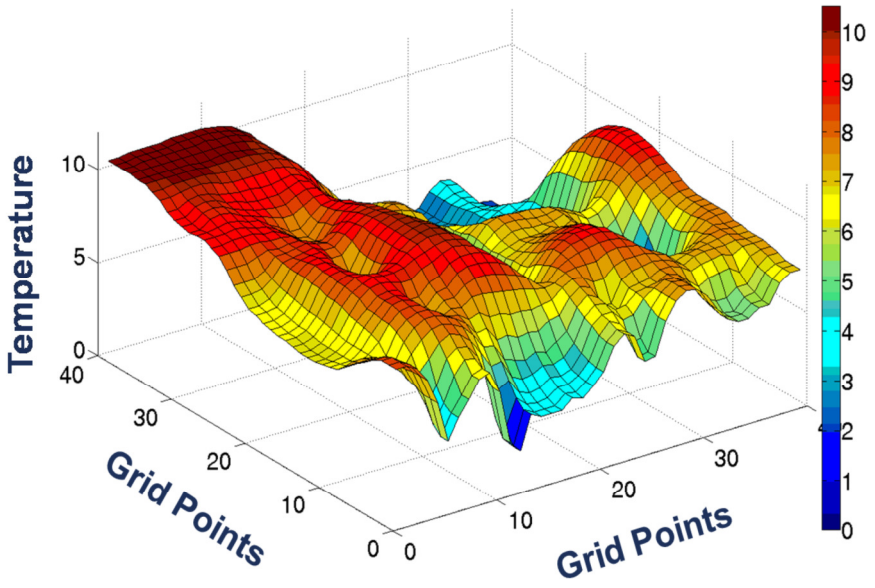


Figure 7.11 Local anneal temperature distribution for the 45nm chip 1.

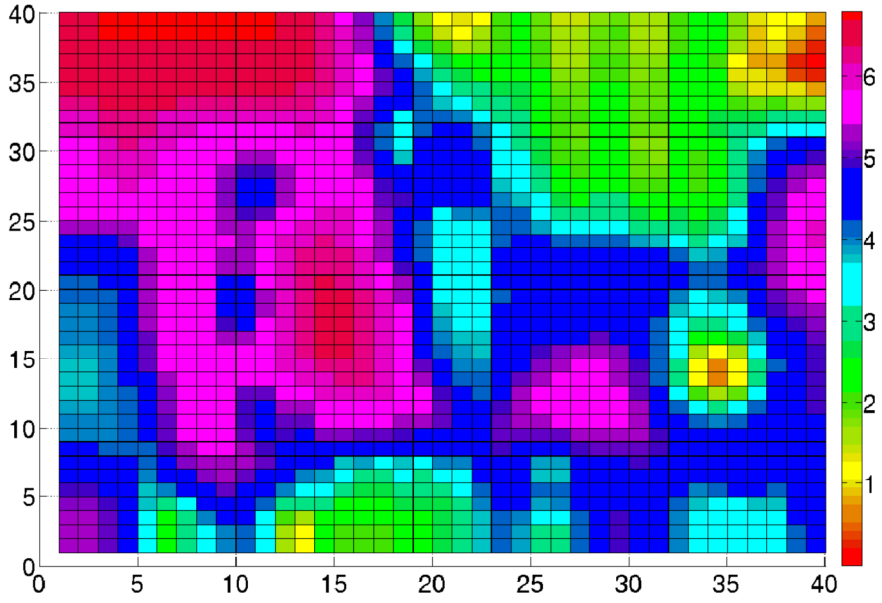


Figure 7.12 Ion map for 45nm Chip 1.

surface of the chips, and 40X40 temperature distribution maps are calculated. Calibre was used for processing the layout and to obtain the relative densities of different layer structures. MATLAB was used for solving the non-linear system of equations, and a

trust-region based technique was used, which solves a linear system of equations to find the search direction [107, 108].

Figure 7.11 shows the temperature map for Chip 1 (45nm). The X and Y axes represent the grid points on the 40X40 grid, while temperature is plotted as a function of position on the grid. The temperature map shows the presence of two high temperature regions on the chip, which result in the two peaks. The difference between maximum and minimum local anneal temperature on the chip was found to be 10.5°C. We observed the temperature map correlates well with the STI density distribution, because STI has the lowest reflectivity among all the layer structures. Lower reflectivity translates into higher absorptivity, and a high density of STI results in higher temperature in the region due to increased absorption of incident power. However, there are other long range effects related to characteristic thermal length that make the correlation less exact.

To examine the electrical effects of the local anneal temperature, TCAD-level models for temperature dependence of Ion and Ioff are employed. For both Ion and Ioff, the values reported are the average values for PMOS and NMOS. Figure 7.12 shows the Ion map for the chip. Ion values are reported as the percentage deviation from the slowest location on the die. Deviation of up to 6.8% from the slowest location are observed. The resulting deviation in inverter delay was found to be 7.3%. A high deviation in performance/delay establishes the need for such a local anneal temperature aware performance analysis. Figure 7.13 shows the corresponding plot for Ioff. The effects observed here are substantial as well. The plot shows Ioff as a function of position on the grid, and all the values are normalized to the lowest leakage point in the die. We observe

that device leakage at the fastest point (highest Ion) is 2.45X higher than the slowest point.

Next we examined another 45nm experimental test chip for chip level anneal temperature variations, using 40X40 rectangular grid for analysis. Figure 7.14 shows the

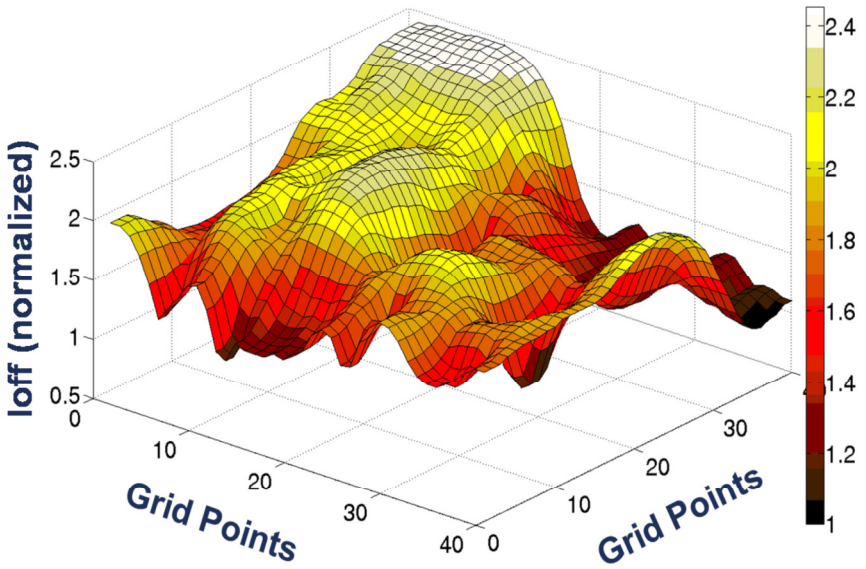


Figure 7.13 Ioff map for the 45nm Chip 1.

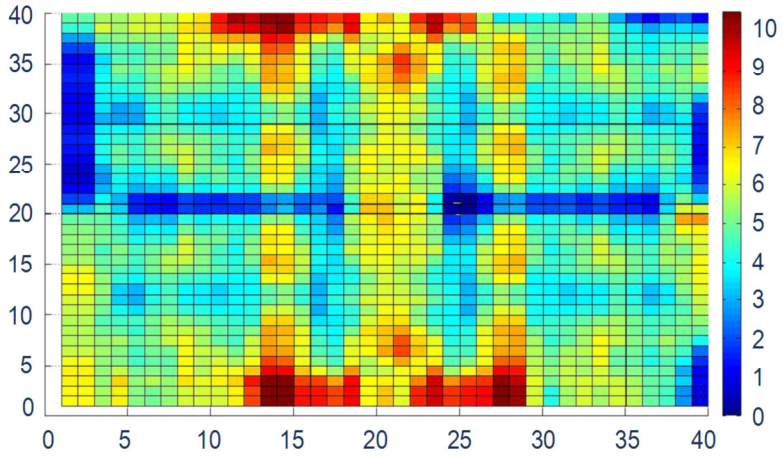


Figure 7.14 Local anneal temperature distribution for 45nm Chip 2.

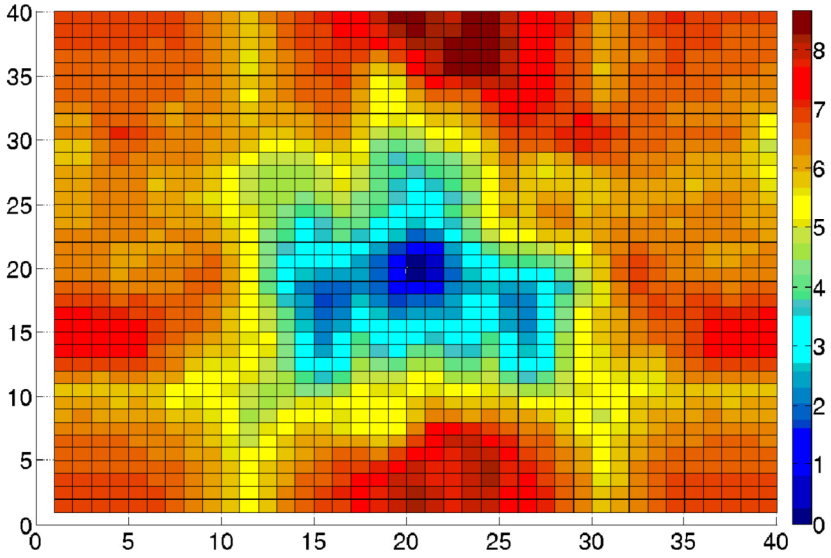


Figure 7.15 Local anneal temperature distribution for the 65nm chip.

chip level anneal temperature distribution for this chip. The difference between maximum and minimum local anneal temperature on the chip was found to be 10.5°C. This translates into 6.6% variation in Ion, and a 2.4X variation in the off current. These numbers are very close to Chip 1, but the actual temperature distributions are very different (layout pattern dependent).

To examine how the effect scales with technology, we performed full chip thermal simulations for a 65nm test chip. We again use a 40X40 rectangular grid for the analysis. Figure 7.15 shows the chip-level temperature distribution. The difference between maximum and minimum local anneal temperature on the chip was found to be 8.5°C, which is slightly smaller than the 45nm test case. The temperature distribution, again, shows a good correlation with STI density distribution. Although the magnitude of chip-level temperature variation is smaller than the 45nm test case, it remains large enough to have a reasonable impact on performance and leakage.

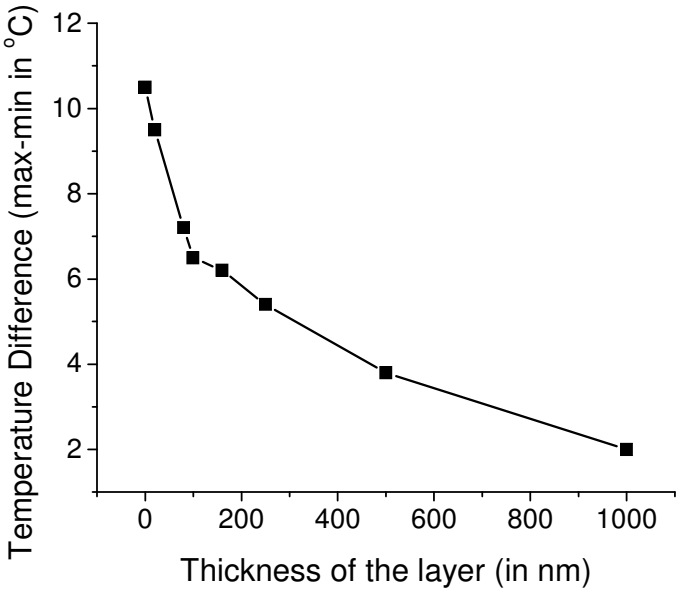


Figure 7.16 Temperature variation as a function of film thickness.

7.4 Minimizing Anneal Temperature Variation

As established in the previous section, RTA-driven local anneal temperature variation significantly impacts device drive current and leakage, and there is a need to minimize this variation. In this section, we discuss three approaches to minimize the impact of anneal temperature variation: one modifies the process flow by introducing a film deposition step, and the other two approaches use anneal temperature variation aware design to mitigate the impact of anneal temperature variation across chip. These approaches are analyzed for effectiveness and cost of implementation.

7.4.1 Film deposition to minimize the difference in reflectivities

There are five distinct layer structures at the time of RTA, with different optical properties (absorptivity, emissivity, and reflectivity). This difference leads to anneal temperature variation across chip, based on the relative densities of these different layer structures. Such a variation can be mitigated by minimizing the difference between these

reflectivities. One way to accomplish this goal is to deposit a film of uniform thickness across the entire chip area prior to the RTA step, followed by film removal post RTA. With the rest of the layer structure remaining the same, this deposited film makes the different layer structures similar to each other, thereby

reducing the difference in their optical properties. Increasing the thickness of the deposited layer, decreases the difference in optical properties, and reduces the anneal temperature variation across chip.

Figure 7.16 shows the magnitude of chip level anneal temperature variation as a function of deposited layer thickness for 45nm Chip 1. Anneal temperature variation decreases as thickness of the film is increased, and for a film thickness of 1000nm, anneal temperature variation is as low as 2°C. Thus, film deposition effectively mitigates anneal temperature variation, without any design phase optimization. However, it incurs two additional process steps (film deposition and removal), and is very sensitive to the exact values of incident radiation wavelength. This means that the film deposition thickness needs to be finely tuned in accordance with the heater material and temperature, which in turn decides the incident radiation wavelength.

7.4.2 Filler Insertion

As seen in the results section, anneal temperature has a dependence on the relative densities of different layer structures, and in particular upon the STI density. This is because STI has the lowest reflectivity amongst all the layer structures, which translates into higher absorptivity. Thus, high STI density results in higher temperature in the region due to increased absorption of incident power. Filler insertion can be used to achieve more uniform STI density across the chip, thereby decreasing the magnitude of

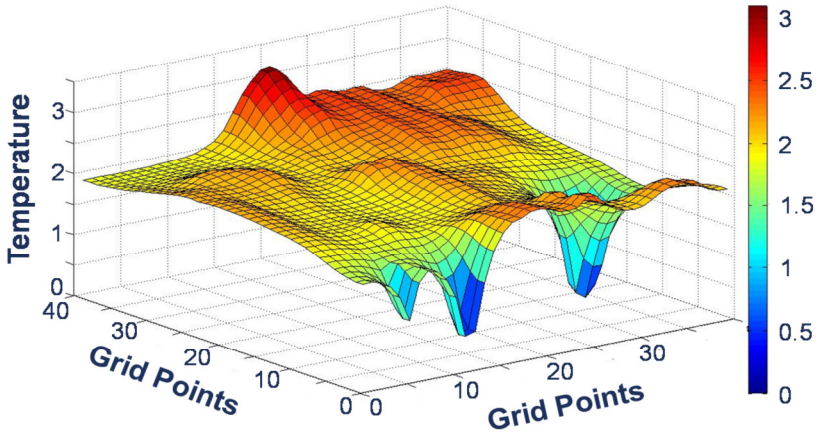


Figure 7.17 Anneal temperature distribution for Chip 1, post active filler insertion.

local anneal temperature variation. To explore the effectiveness of this approach we performed a simple filler insertion experiment. We try to achieve uniform STI density across Chip 1, through active fill insertion, without rearranging layout blocks (only through filler insertion). Figure 7.17 shows the anneal temperature distribution for Chip 1 after dummy fill insertion. Magnitude of anneal temperature variation is decreased to 3.1°C, from the original variation of 10.5°C. This shows that filler insertion can effectively mitigate anneal temperature variation. In [109], authors attempt to reduce impact of anneal temperature variation through a two-step procedure, which involves rearranging layout blocks, and polysilicon fill insertion. They report almost zero RTA variation in the final optimized chip. However, this work uses a simple STI density based linear fit to calculate R_s, which does not consider the densities of other layers and becomes inaccurate as the density of STI becomes uniform (i.e., as STI density becomes uniform, differences in density between other layers become more important in determining the local anneal temperature).

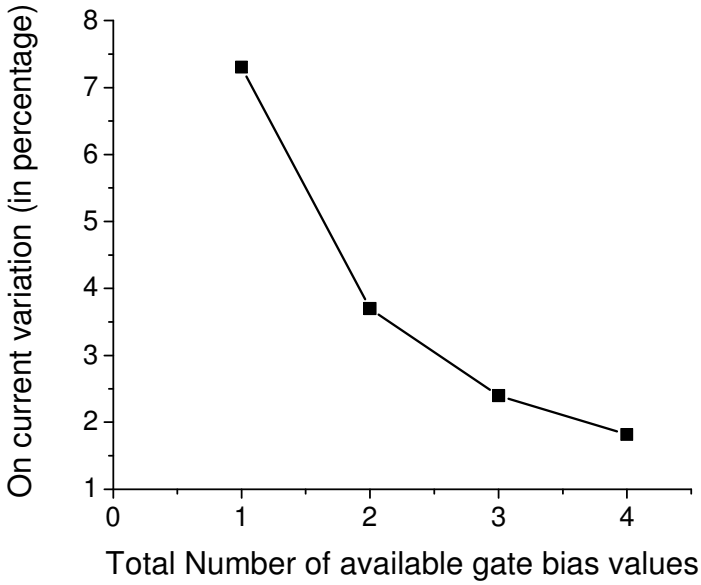


Figure 7.18 Ion variation as a function of number of gate bias values.

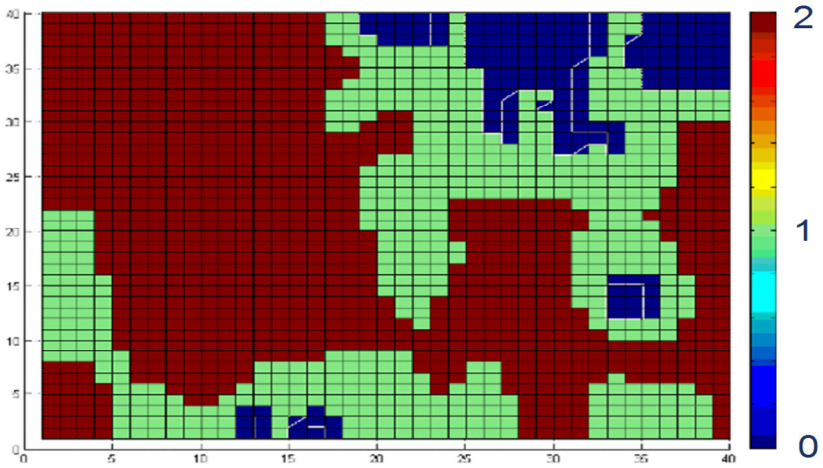


Figure 7.19 Gate bias distribution map for three gate bias values.

Dummy fill can either be active area or polysilicon. Active area fill alters the distribution of STI, thereby affecting the mechanical stress induced in the channel of device (and hence the device mobility and performance). STI exerts compressive stress, which is beneficial for PMOS performance but detrimental for NMOS. Thus, active area fill can have unwanted effects on circuit performance and leakage. Polysilicon fill is

electrically inactive, but can couple capacitively with lines in neighboring layers. Although dummy polysilicon lines are left floating, they can still degrade performance through coupling. As a result it makes sense to consider minimizing or restricting the amount of fill inserted.

7.4.3 Gate-length Biasing

Gate-length biasing has been used in past to reduce leakage power [110]. This involves providing small biases (<10%) of transistor gate lengths to reduce leakage. Gate-length biases smaller than 10% ensure pin-compatibility with un-sized version of the cell, and helps retain the same polysilicon pitch as the unbiased version. We propose the use of gate-length biasing to mitigate the electrical impact of anneal temperature variation. NMOS and PMOS devices are biased independently, and the bias values are fixed for a chip. As the number of available bias values increases, greater reduction in RTA-induced variation is achieved. Figure 7.18 shows how the magnitude of on current variation for 45nm chip 1 decreases as the number of available gate bias values is increased. For PMOS (NMOS) the value of largest gate bias is less than 8% (6%)

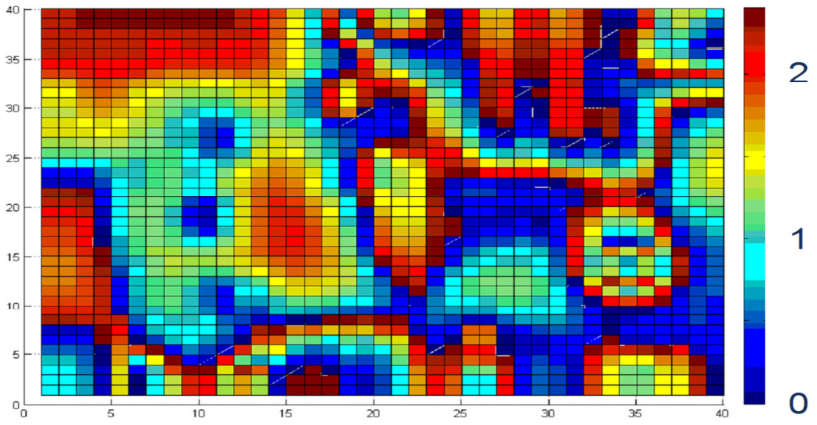


Figure 7.20 Ion distribution for Chip 1 for three available values of gate bias.

of the nominal gate length. On current variation can be reduced to 2.4% by using three values of gate bias. Figure 7.19 shows the gate length bias distribution for three available values of gate length. Here 0 indicates nominal gate length (zero bias), 2 corresponds to the highest value of gate length bias, and 1 corresponds to a middle value. Figure 7.20 shows the corresponding chip level on current distribution.

While gate-length biasing does not have any direct implementation cost, it is impossible to approach zero RTA induced variation through this technique without an unreasonable number of required gate bias values, which would incur high characterization costs. A hybrid approach is therefore suggested, where most of the improvement is derived from gate-length biasing, while filler insertion is used to achieve almost zero variation with a very small amount of dummy fill being inserted. Such an approach would best balance quality of results with implementation costs.

7.5 Summary

In this work, we proposed a new local anneal temperature variation aware performance and leakage analysis framework which embodies transistor level models for anneal temperature sensitivity, to incorporate RTA induced temperature variation into traditional timing/leakage analysis. We solve for chip-level anneal temperature distribution by dividing the wafer surface into rectangular grids, and employ TCAD-based device-level models for drive current (Ion) and leakage current (Ioff) dependence on anneal temperature variation, to capture the variation in device performance and leakage based on its position in the layout. Experimental results based on a 45nm test chip shows anneal temperature variations of up to 10.5°C, which results in 6.8% variation in device performance and 2.45X variation in device leakage across the chips. The corresponding variation in inverter delay was found to be 7.3%, thereby establishing the

importance of such a local anneal temperature variation aware performance/leakage analysis. We also analyze techniques to minimize anneal temperature variation, and examine their effectiveness in mitigating RTA induced variation. Based on the analysis, it was concluded that a hybrid approach with gate length biasing and filler insertion could provide the best solution, at lowest implementation cost.

Chapter 8
Analysis and Optimization of SRAM Robustness for Double Patterning Lithography

In Chapter 1, we discussed briefly how Double Patterning Lithography (DPL) further increases the lithography based gate length variation due to the existence of dual populations for critical dimension (CD). Pitch-split DPL decomposes and prints critical layout shapes in two exposures, and systematic offsets between the two masks leads to mismatch between adjacent devices. So, devices on alternate poly tracks are correlated while devices on adjacent tracks are not. While this affects the timing analysis and optimization of logic circuits, it has a much more severe negative impact on SRAM robustness where a mismatch between devices can cause significant yield loss. Thus, there is a need to study the design impact of DPL-induced variability, to enable DPL aware design and optimization.

Figure 1 shows the schematic and conventional layout of a typical six transistor SRAM cell. The access transistor and pull up/pull down (PU/PD) transistors for a given side lie on different poly tracks (e.g., PG1 lies on a different track than PU1/PD1) and will be printed with different exposures under DPL. As a result the access and PU/PD

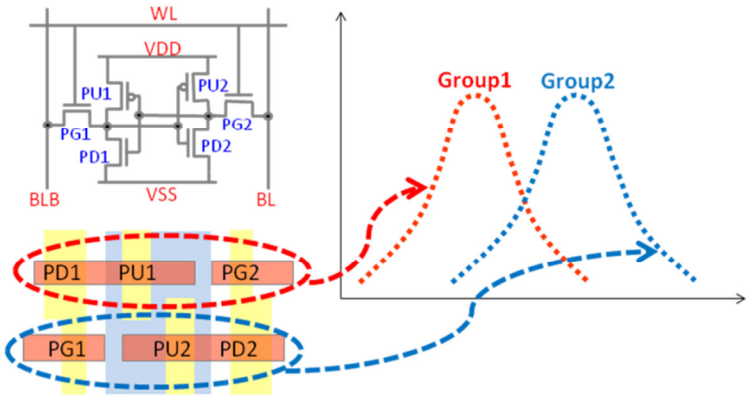


Figure 8.1 SRAM schematic and layout showing DPL based variation.

transistors on a given side of the symmetric circuit structure will have uncorrelated gate length distributions, with such mismatch severely impacting the SRAM cell robustness by increasing functional failures. For example, if the access transistor becomes stronger than the PD transistor, the SRAM cell will be more prone to read failures. Reference [111] presents modeling of SRAM failures, and statistical optimization to minimize yield loss, for single exposure lithography. In [112], the authors analyze the impact of lithographic variation on electrical yield of 32nm SRAM, for single patterning and cut-mask double patterning [113], where one exposure is used to print the polysilicon tracks and the other is used for cut-mask to print line ends. However, with technology scaling, the polysilicon pitch will go below the resolution limit of single exposure, and double patterning will be required to print the adjacent polysilicon tracks. To the best of our knowledge, this is the first work to analyze SRAM robustness under pitch-splitting double patterning, and propose a DPL-aware sizing scheme to mitigate yield loss due to DPL.

In this chapter, we use measurement (from a 45nm test chip) and simulation to analyze the impact of DPL-based variation on SRAM robustness, as compared to traditional single exposure lithography. We show that the DPL impact on cell robustness is substantial, and there is a need for DPL variation aware SRAM robustness analysis and

optimization framework. We propose a DPL-aware SRAM sizing technique that iteratively sizes the SRAM cell to achieve desired robustness while changing the read and write energies by a very small amount. The rest of the chapter is organized as follows. Section 8.1 discusses the background and analysis of DPL impact on SRAM cell robustness, while Section 8.2 introduces the proposed DPL-aware SRAM sizing technique. Experimental results are discussed in Section 8.3, and Section 8.4 concludes the chapter.

8.1 Background and Analysis

As discussed earlier, device mismatch due to DPL can result in increased failure probability of the SRAM cell. This mismatch depends on the mean (μ) and standard deviation (σ) of the two line width distributions, for the two exposures used to print adjacent polysilicon tracks. Based on hardware results, [114] reported 3σ/μ numbers as high as ~16.5% for DPL line width distributions in 32nm technology. Parametric failures in SRAM cell are principally due to:

1. Destructive Read/Read Failure – flipping of the stored data in the cell while reading. Flipping occurs when bump in the read voltage is higher than the trip point of the other inverter (e.g., in Figure 8.1 when the bump in the output of inverter PU2-PD2 (V_{read}) > V_{trip} for inverter PU1-PD1, while reading out a 0).
2. Write failure – failure to write to a cell within the time when wordline (WL) is high.
3. Access time failure – an increase in the access time of the cell violating the delay requirements.
4. Hold failure – destruction of the cell content in standby mode due to the application of lower supply voltage (in order to suppress leakage in standby mode).

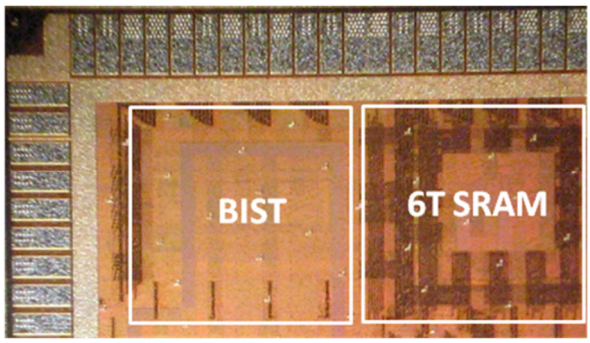


Figure 8.2 Die-shot of the 45nm test chip showing the SRAM array and built-in self-test (BIST) structure.

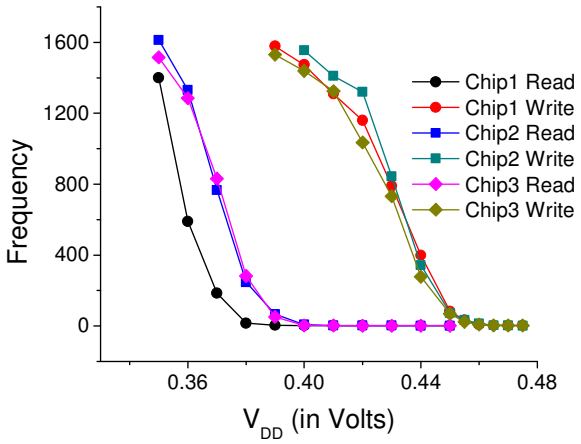


Figure 8.3 Number or read and write failures as a function of V_{DD}.

In this section, we analyze the impact of DPL on SRAM variability and robustness through measurement and simulation.

8.1.1 Test Chip Measurement-based Analysis

We implemented a 45nm test chip with 32kb 6T SRAM arrays to observe the effect of double patterning on SRAM failure counts. Figure 8.2 shows the die shot of the chip with SRAM arrays and built-in self-test (BIST) structures. We measured 75 such chips to obtain data on read and write failures. In order to obtain statistically significant failure data from a 32kb memory, we lower the operating voltage to induce failures. For a given

sample of transistor lengths, the mismatch has a different impact on read and write errors, and hence it is necessary to differentiate between them. Read errors are examined by writing to the memory at nominal V_{DD} (1.1V) and then reading out at a lower voltage. Similarly, capturing write errors involves writing the cells at a reduced voltage, followed by a read at full V_{DD}.

It is necessary to observe how the failures scale with voltage to determine the appropriate operating voltage for measurement. Figures 8.3 shows the increase in read and write errors as supply voltage is reduced in three test chips. The peripheral circuits are designed such that they are not failure critical at lower values of voltage, and the failures occur only in the SRAM cells. In the case of a read operation, the number of failures increases abruptly at voltages close to 0.36V, whereas in the case of write operation the number of errors becomes sufficient for statistical analysis at approximately 0.45V. Eventually, as the operating voltage approaches the threshold voltage, nearly all cells fail. The difference in behavior between read and write operations matches results from SPICE simulation of a nominally sized SRAM cell. Write operation is more stable at nominal V_{DD} (write margin is higher than read margin), however as V_{DD} is lowered it becomes less robust than read operation. Thus, more write failures will occur at lower V_{DD}, which means that significant write failures are observed before reaching the threshold voltage where nearly the full array abruptly fails. Based on these observations,

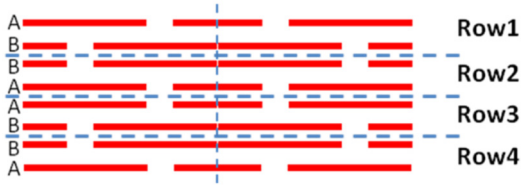


Figure 8.4 Stick diagram (polysilicon only) showing how rows of SRAM cells are laid out.

we focus on write 0 and write 1 operations, and the V_{DD} used for measurement is 0.45V.

Figure 8.4 shows the stick diagram of polysilicon layer depicting how rows of SRAM cells are laid out. As shown in the figure, adjacent rows of SRAM cells are mirror images of each other. Hence, if gates on polysilicon track A (shown in Figure 8.4) are stronger (smaller channel length) than those on track B in one row of SRAM cells, gates on track B will be stronger in the adjacent rows. If in fact DPL has a major impact on SRAM stability, even rows should behave differently than odd rows for a given operation (write 1 or write 0). This difference should result in significantly different error counts for the two rows.

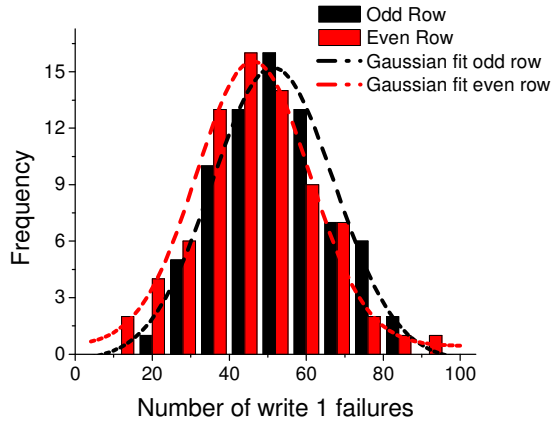


Figure 8.5 Write 1 failure count distribution for even and odd rows.

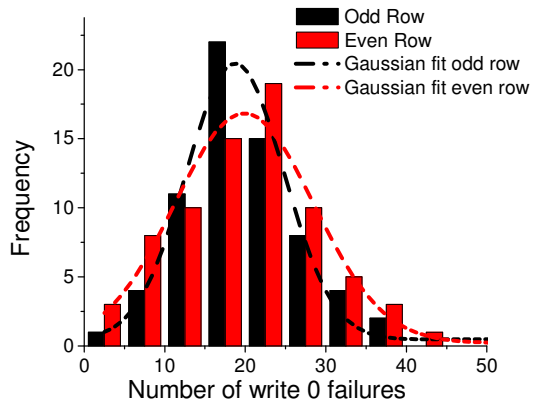


Figure 8.6 Write 0 failure count distribution for even and odd rows.

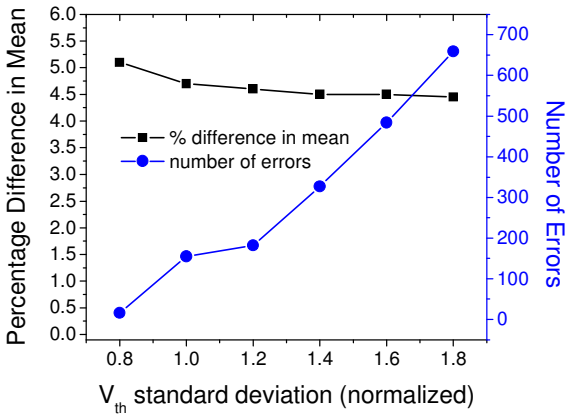


Figure 8.7 Difference in mean and number of errors for write 0 operation as a function of V_{th} standard deviation.

Figure 8.5 shows the error count distribution for write 1 operations across the 75 test chips, for even and odd rows, along with the Gaussian fit for the two distributions (total number of failures analyzed is ~7700). The distributions for even and odd rows are distinct, and the difference in mean for the two distributions is ~14.5%. Figure 8.6 shows the corresponding distributions for write 0 operation, and the difference in mean between even and odd rows is ~25% in this case (total number of failures analyzed is ~3400). To show through simulation that most of this difference in mean failure counts is caused by double patterning and not random V_{th} variation, we plot the number of write 0 errors as well as the percentage difference between the mean number of errors of even and odd rows as a function of V_{th} standard deviation for 75 length samples (to model 75 dies measured) under the assumption of single exposure lithography. Under a single exposure lithography assumption both even and odd rows use the same length sample, but include random V_{th} variation, and the V_{th} standard deviation is increased in steps. Figure 8.7 shows this plot; all the V_{th} standard deviation values are normalized to σ_{VT0}, where σ_{VT0} is the standard deviation of intra-die V_{th} variation specified for the 45nm technology. As

V_{th} standard deviation increases, the number of errors and the difference in the number between even and odd rows both go up. As a result, the percentage difference in error remains almost constant at around~4.5%, which is significantly lower than the observed difference in the silicon measurements. Hence, random V_{th} variation is unlikely to have caused the high difference in mean observed between even and odd rows for the test chip measurements.

Next, we performed Student's t-test [115] on the two sets of data for write 0 and write 1 operations, to more conclusively reject the possibility that the observed difference in means is due to random variation rather than DPL. Student's t-test is a small-sample statistical hypothesis test, in which two sets of data are tested to determine if their mean difference is due to chance/ random variation, or if there is indeed a difference in the two sets of data. The data is said to follow the null hypothesis if there is no effective difference between the observed sample means for the two sets, and any measured difference is due only to chance/random variation. For write

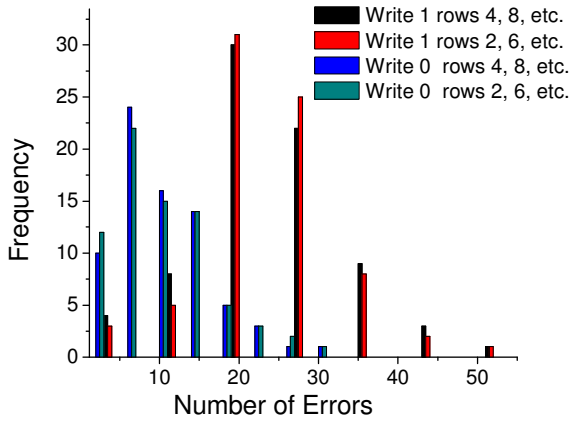


Figure 8.8 Write 0 and write 1 failures for two subsets of even rows.

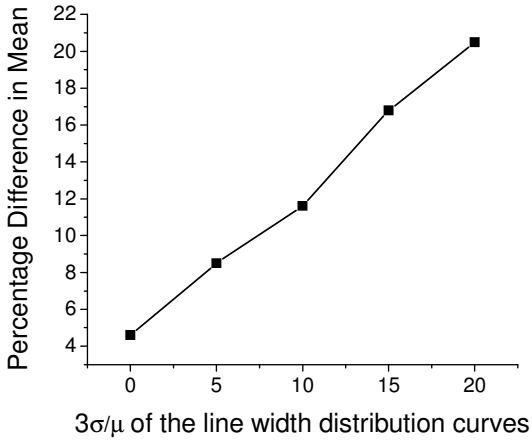


Figure 8.9 Write 1 difference in number of failures for even and odd rows as a function of 3σ/μ of line width distribution curves for DPL.

1 operation, the probability of the result assuming the null hypothesis was found to be 0.025, and the corresponding probability for the write 0 operation was found to be 0.0033. Both these values suggest the improbability of the null hypothesis (probability < 0.05) and point to the conclusion that the differences in measured error count are due to double patterning lithography.

Another way to validate the t-test results is to compare the failure counts between subsets of even/odd rows. Since there is no added variability due to DPL, two subsets of even rows should show similar error counts for the 75 test chips within the bounds of random variation. We break the even rows into two subsets: subset 1 comprising of rows 2, 6, 10, etc., and subset 2 comprising of rows 4, 8 , 12, etc. Figure 8.8 shows the error count distribution for the two subsets, for write 1 and write 0 operations. As expected, the distributions are very similar for the two subsets, with the difference in mean failure count being ~3% for write 1, and ~1% for write 0 operation. On performing Student's t-test upon the two subsets, we obtain the probability of the result assuming null hypothesis to be 0.91 for write 0 operation, and 0.62 for write 1 operation. Thus, t-tests can

successfully determine that the two sets of data are subsets of same kind of row (even in this case). This experiment further validates the t-test results suggesting difference in behavior between even and odd rows is due to DPL.

In order to estimate the DPL line width distribution curves for the two separate exposure steps, we use simulation to find 3σ/μ value for the two curves that can generate such a difference between failure counts for even and odd rows. To simplify the problem, we assume that the two curves have same mean and standard deviation, and sweep the 3σ/μ values to generate difference in error counts between even and odd rows as a function of 3σ/μ for the line-width distribution. Figure 8.9 shows the resulting plot for write 1 operation. The plot gives a rough estimate of 3σ/μ for the two curves to be ~12.8%. Such a difference can potentially lead to appreciable degradation in SRAM robustness compared to the single exposure case. We now perform simulation based analysis to quantify this impact.

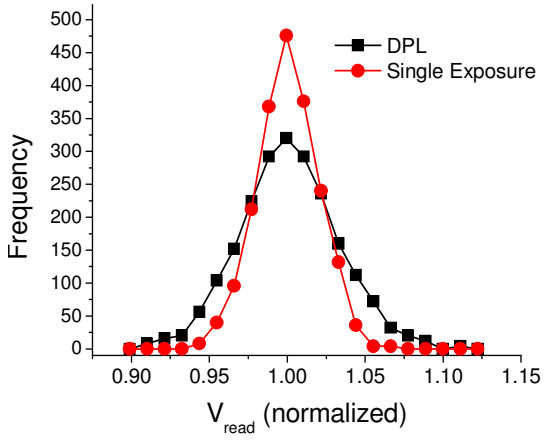


Figure 8.10 V_{read} distribution for DPL and single exposure system.

8.1.2 Simulation based Analysis

In order to analyze the effect of DPL on SRAM parametric failures, we must begin with the analysis of the failure triggering mechanisms under DPL based mismatch. First, we study the effect of DPL based mismatch on V_{read} and write time of a 45nm SRAM, and compare the effect to that of a single exposure based system. V_{read} is defined as bump in the output voltage of inverter PU2-PD2, while reading out a 0 from SRAM. Higher the value of V_{read}, more prone is the cell to read failure under random V_{th} variation. Similarly, a cell with higher write time will be more likely to experience write failure under random V_{th} variation. For cell level DPL based analysis, we assume that gate length distribution of PD1, PU1, and PG2 (mean μ_1, standard deviation σ_1), is uncorrelated with the gate length distribution of PD2, PU2, and PG1 (μ_2, σ_2), which lie on separate polysilicon track. Our analysis focuses on a 45nm industrial SRAM cell which is optimized for single exposure based patterning. Based on V_{th} corner analysis, the nominal cell experiences read failure at a V_{th} σ value of $4.23\sigma_{VT0}$, and write failure σ value is $6.36\sigma_{VT0}$, where σ_{VT0} is a specified number for the technology. These numbers establish that, in general, write operation is much more robust for the industrial SRAM being analyzed, which provides the designer an opportunity to make the read operation more robust at the cost of degrading write robustness by a small amount. A similar opportunity will exist in case the read operation is more robust than write. We exploit this property later on in our DPL-aware sizing optimization.

Figure 8.10 shows the V_{read} distribution for the simple case of equal means ($\mu_1 = \mu_2$) and standard deviations ($\sigma_1 = \sigma_2$), with $3\sigma_1/\ \mu_1 = 3\sigma_2/\ \mu_2 = 10\%$, for the two line width distribution curves. Also shown in Figure 2, is the distribution of V_{read} for single exposure case assuming $3\sigma/\ \mu$ is 10%. All the V_{read} values are normalized to V_{read} for the nominal

cell (V_{read0}). The mean and variance of V_{read} distribution for DPL is found to be ~1 and 0.035 (normalized to nominal V_{read0}), while the mean and variance for single exposure are ~1 and 0.018, respectively. Standard deviation in the case of DPL based technology is almost twice the standard deviation for single exposure system. DPL $\mu + 3\sigma$ (1.11 Vread0) is higher than the single exposure case (1.06 Vread0). Hence, it is important to consider both gate length and Vth variation for accurate variability/robustness analysis in DPL systems. Figure 8.11 shows similar plots for write time analysis. DPL based distribution has a mean value of ~1 and a standard deviation of 0.024, while the single exposure mean and standard deviation are ~1 and 0.014, respectively (normalized to nominal write time). Again, the standard deviation of DPL case is higher than the single exposure, suggesting that there is a need for DPL variation aware analysis. DPL almost doubles the standard deviation observed in the case of single exposure system, when the means of the two gate length distribution curves are identical. In case there is a difference in means, impact of DPL increases even further due to increase in mismatch between transistors on adjacent polysilicon tracks.

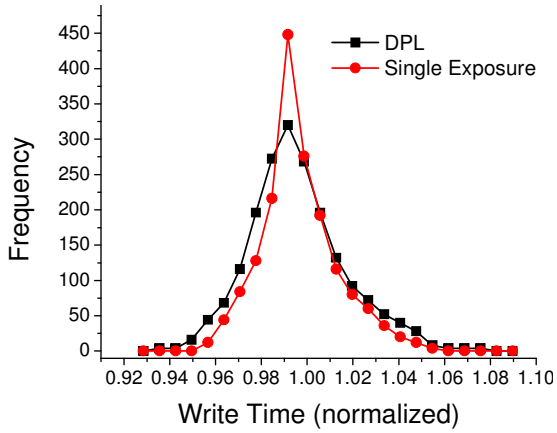


Figure 8.11 Write time distribution for DPL and single exposure system.

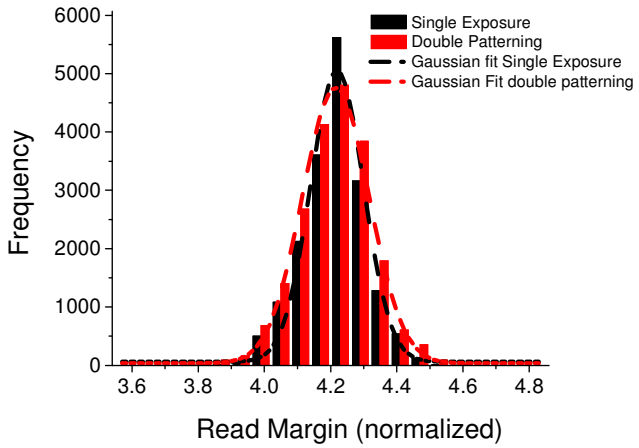


Figure 8.12 Read V_{th} σ failure number distributions for DPL and single exposure lithography.

A convenient method to analyze SRAM robustness is through corner-based failure analysis, where the failure is characterized in terms of the smallest multiple of sigma V_{th} at which the cell experiences functional failure, and this value is called read/write margin (or the V_{th} σ failure numbers). Figure 8.12 shows the read V_{th} failure distributions for both the cases. These distributions are generated by performing V_{th} corner based failure analysis at each gate length sample, to find the smallest V_{th} σ number at which the cell experiences functional failure. As expected based on V_{read} analysis, double patterning leads to much worse V_{th} failure numbers (or the V_{th} failure σ distribution has higher variance). Mean of the read V_{th} failure curve for DPL is $4.20\sigma_{VT0}$, with a standard deviation of 0.2 σ_{VT0}. Single exposure read mean is $4.23\sigma_{VT0}$, and standard deviation is $0.1\sigma_{VT0}$, and the standard deviation is half of that in the case of DPL. For the μ-3σ point in the distribution, the probability of failure increases by ~3.3X due to DPL, as compared to single exposure lithography.

We look at another way to analyze the read stability instead of the computationally intensive analysis involving V_{th} corner analysis on every length sample.

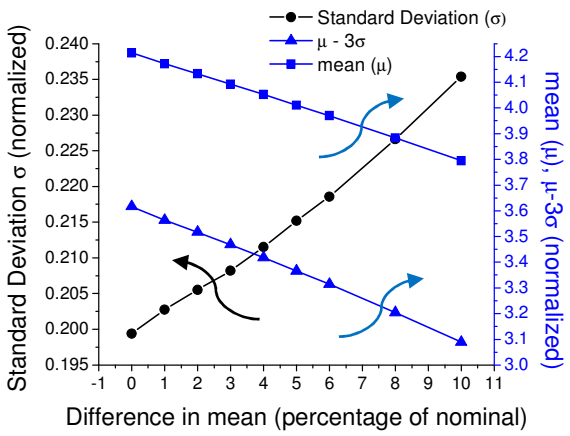


Figure 8.13 Read V_{th} standard deviation, mean and μ-3σ as a function of difference in means of the line width distribution curves for DPL.

We pick the worst few (1%) length sample points for read operation, by using the V_{read} distribution curve, and the knowledge that high values of V_{read} make the sample point more vulnerable to V_{th} variation based failure. On these selected points, we run V_{th} corner analysis and take an average of the failure numbers. The average roughly corresponds to μ - 2.8σ for the case of read failure. This shows that the corner cases of the V_{read} distribution are the ones which generate lowest (worst) V_{th} σ failure numbers. In other words, V_{th} corner analysis on the worst cases of DPL based V_{read} distribution captures most of the worst cases (lowest V_{th} σ failure values) of the complete analysis involving finding V_{th} failure numbers at each length sample. So, if the aim of an analysis is to capture the worst case, then length-based analysis and the V_{th} corner analysis can be decoupled while still capturing most of the bad cases (more prone to functional failure).

Figure 7 shows how the mean (μ), standard deviation (σ), and μ-3σ points of the read V_{th} σ failure distribution vary if a difference in mean is introduced between the two curves ($\mu 1 \neq \mu 2$), for read operation. All the values are plotted against the difference in means of the two DPL length distributions (expressed as a percentage number of the

nominal value). As the difference in mean of the two length distributions increases, mean of the V_{th} σ failure distribution decreases, and variance increases which means that the cell becomes less robust. For difference in mean of 4%, μ-3σ value of the V_{th} failure distribution goes down to as low as $3.41\sigma_{VT0}$, which ~13% smaller than the value for single patterning, and the probability of failure for the μ-3σ point increases by ~7X. Hence, the impact of DPL on SRAM cell robustness greatly increases with the increase in the difference between the mean of two gate length distribution curve. This is expected since increasing the difference in mean of the two length curves increases the mismatch between SRAM devices, thereby making them more prone to failure.

An interesting analysis examines the mean and variance of the V_{read} distribution if PG1, PU1, and PD1 were on the same poly track, assuming that the layout could be changed in such a manner. Now the access transistor and PU/PD transistors will have identical lengths and there would be no mismatch there due to DPL. As a result, we would expect the V_{read} distribution to be much closer to the single exposure case. For the simple case of equal means ($\mu_1 = \mu_2$), and variances, with $3\sigma_1/\ \mu_1 = 3\sigma_2/\ \mu_2 = 10\%$, V_{read} distribution has a mean of ~1 and standard deviation of 0.02 (normalized to nominal V_{read}). These values are very close to single exposure case as expected ($\mu = 1$, $\sigma = 0.018$). However, the actual V_{th} σ failure numbers are higher than the single exposure case. This is because of the mismatch between the two inverters due to different distributions. For example, in case of a reading a zero, although V_{read} is not affected much (as PG1, PU1, and PD1 lie on the same polysilicon track), the trip voltage (V_{trip}) of the other inverter (PU2-PD2) depends on the other uncorrelated length distribution (for PU2, PD2, and PG2). This mismatch between V_{read} and V_{trip} can cause samples with high V_{read} and low

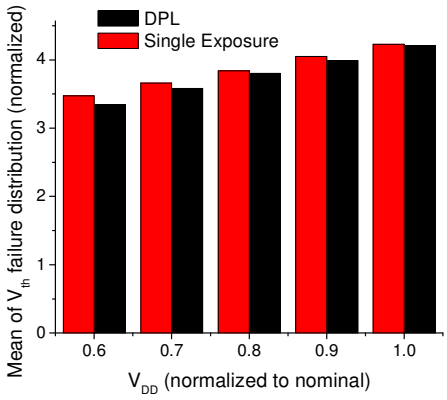


Figure 8.14 Mean of V_{th} failure distribution as a function of V_{DD} scaling for DPL and single exposure techniques.

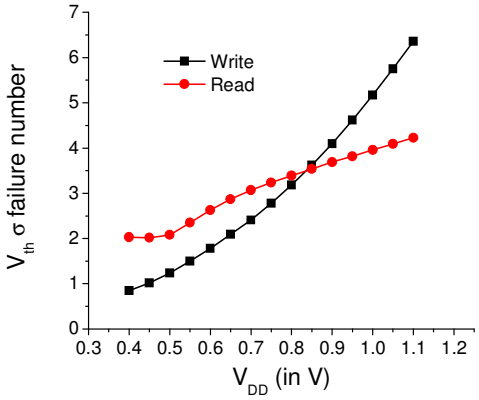


Figure 8.15 Read and write V_{th} σ failure numbers as a function of V_{DD}.

V_{trip}, which are very vulnerable to random V_{th} variation based failure. Even despite this mismatch, the failure numbers are much better than DPL based results for the original layout, and this could potentially be a useful design optimization to mitigate yield losses due to DPL. However, such a change in layout leads to very high area penalty (~2×), making this an unattractive design change.

Finally, we look at the effect of DPL on voltage scaling. With the SRAM cell fixed at the nominal sizing, V_{DD} is scaled to analyze the impact of DPL on V_{th} σ failure number. Figure 8.14 shows how the mean of read V_{th} σ failure distribution varies with V_{DD} for DPL and single exposure systems. V_{DD} values are normalized to the nominal

VDD. Supposing we require the mean of the V_{th} failure distribution to be greater than 3.5 σ_{VT0}, we find that the supply voltage can be scaled down to 0.62 of the nominal V_{DD} under single exposure. However, degraded robustness under DPL leads to less favorable voltage scalability, requiring a supply voltage of at least 0.68 of the nominal V_{DD}, leading to approximately 20% energy penalty. Figure 8.15 shows how the V_{th} σ failure number changes for read and write operations as V_{DD} is lowered, at nominal value of gate length. Write operation is more stable at nominal V_{DD} (higher value of V_{th} σ failure number), however as V_{DD} is lowered it becomes less robust than read operation. At nominal V_{DD}, DPL-aware sizing optimization can make read operation more robust at the cost of degrading write robustness by a small amount. However, at lower values of V_{DD}, write operation needs to be optimized and made more robust, and this could be done at the cost of read robustness.

8.2 DPL-aware SRAM Sizing

Based on the intuition developed through analyzing the impact of DPL on SRAM cell robustness, we now propose a DPL-aware SRAM sizing scheme to mitigate the negative impact of DPL on SRAM robustness. Key points to remember from the analysis section are:

- Typically read and write robustness numbers are very different for SRAM, providing the designer an opportunity to trade the robustness of one operation off for the other. For the SRAM cell under consideration write is more robust than read at nominal V_{DD}, which is the common case in modern SRAMs (read is more stable at lower V_{DD}).
- The length-based analysis and the V_{th} corner analysis can be decoupled, and the DPL sizing optimization can focus on optimizing the worst cases (say μ+3σ) of the V_{read} and write time distributions.

- Given a range in which the means and variances of the two gate length distributions could lie, there is a worst case combination that creates maximum mismatch (highest values of mean and standard deviation for line width distribution curves). Any sizing optimization should be directed at this worst case, and the other intermediate cases are expected to improve by using the resulting sizes. This fact is verified in the experimental results.

The DPL-aware SRAM sizing optimization problem can be viewed as that of shifting the V_{th} σ failure number distribution (Eg. Figure 8.12) to the right for the less robust operation, while meeting the constraints on read and write times (to avoid access failures), and read/write energies. Shifting the distribution to the right would increase the value of V_{th} σ failure number, and hence decrease the probability of failure, for any given point on the distribution (higher value of failure σ means lower probability of failure). We can choose a representative point on the distribution (of the form μ-aσ), and try to shift it to the right (or increase its V_{th} σ failure number) to achieve this goal. This is based on the assumption that variance of the V_{th} σ failure number distribution would not change drastically during the sizing optimization, and so increasing the V_{th} σ failure number for one point is the same as shifting the entire curve to the right. This assumption is validated by the experimental results discussed in the next section, where the variance of the curve is almost the same before and after the optimization. For our experiments, we choose this representative point as the mean (μ) of the failure number distribution curve. Hence, the problem can now be seen as maximizing the mean (μ) of V_{th} σ failure distribution for the less robust operation, while meeting the constraints on read and write times (to avoid access failures), and read/write energies. Hold failures were demonstrated to have much

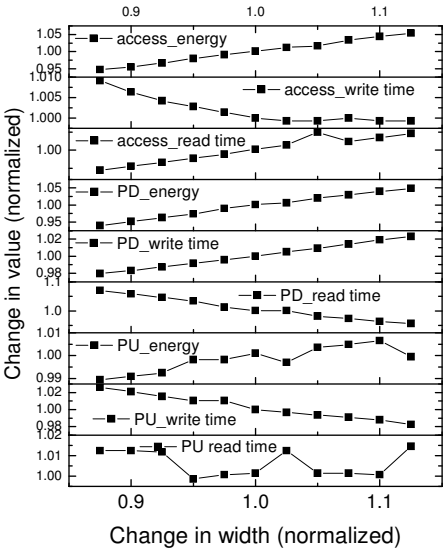


Figure 8.16 Variation in read, write times, and average energy with change in w_{PU}, w_{PG}, or w_{PU} for nominal gate lengths.

lower occurrence, and they can be further controlled by appropriately choosing standby mode supply voltage [10], so they were not included in the sizing optimization as constraints. Only the widths of the SRAM devices were used as optimization variables. The problem can then be stated as:

$$\begin{aligned} &Maximize \quad f(w_{PU}, w_{PD}, w_{PG}) = \min(\mu_{Vth,read}, \mu_{Vth,write}) \\ &subject\ to: \\ &\quad E_{read} + E_{write} < E_0 \\ &\quad T_{read} < T_{r,0} \\ &\quad T_{write} < T_{w,0} \end{aligned} \tag{1}$$

w_{PD}, w_{PG} are the widths of pull up, pull down, and access device, respectively, $\mu_{Vth,read}$, $\mu_{Vth,write}$ are the mean of V_{th} σ failure number distribution for read and write, respectively, E_{read}, E_{write} are the read and write energies, while T_{read} and T_{write} are read and write times, respectively. The function f chooses the less robust of the two operations (read/write) to optimize as the one with lower value of mean of V_{th} σ failure number distribution, while E_0, $T_{r,0}$, and $T_{w,0}$ are the constraints on total energy, read time, and

write time, respectively. In order to solve this problem, we can use the intuition developed during analysis, and decouple the length based analysis and V_{th} corner analysis. As a result, we can try to minimize the worst case values of V_{read} distribution if write is more robust than read, or write time distribution if read is more robust than write, while considering only DPL based gate length (line width) variation. This can be seen as shifting the V_{read} (Figure 8.10) or write time (Figure 8.11) distribution to the left, thereby making the cell more robust. E.g. if we shift the V_{read} curve to the left, we decrease the V_{read} value for any given point on the curve. Lower the value of V_{read}, less prone is the cell to read failure under random V_{th} variation. Thus shifting the V_{read} curve to the left would increase the robustness which results in higher value of V_{th} σ failure numbers. Again, we can choose a representative point on the $V_{read/write}$ time curve and minimize it (shift it to the left). This minimization can be seen as minimizing the value of a point say $P=\mu+a\sigma$ on the $V_{read/write}$ time curve (for our analysis a=3).

Now if we try to size the SRAM cell iteratively where we can change one width value by one step (say 1% of the nominal size) at a time; at each step of the iteration we will have the choice to change either w_{PU}, w_{PG}, or w_{PD}. But changing each of them by one step has a different effect on the read/write times, energies, and V_{read}. Figure 8.16 shows the variation of read and write times, and average energy ($(E_{read}+E_{write})/2$) with change in w_{PU}, w_{PG}, or w_{PU}, for nominal value of gate length. For each sub-plot, width of one of access (w_{PG}), pull up(w_{PU}), or pull down(w_{PD}) transistors is varied, while keeping the other two values fixed at nominal, and one of the values out of read time, write time, or average energy is plotted as a function of the width being varied. All the values are normalized to their value for the nominal cell. The key conclusion from the figure is that,

the decision on which transistor to size at a given step in the optimization iteration, depends on the actual values of w_{PU}, w_{PG}, and w_{PD}. For example in order to increase read robustness, we can choose increase either of w_{PU} or w_{PD}, or reduce the size of the access transistor w_{PG} by one step, but the best choice depends upon the actual values of w_{PU}, w_{PG}, and w_{PD} at that point. In order to choose the best width value to change, we define a sensitivity metric G, based on the decrease in the value of point P (ΔP) and change in the value of a constraint function C (ΔC), where point P is the representative point chosen on the $V_{read/write}$ time curve.

$$G = \frac{\Delta C}{\Delta P}, \qquad P = \mu + a\sigma \quad (a = 3)$$
$$C = w_1 \frac{E_{read}}{E_{r,nom}} + w_2 \frac{E_{write}}{E_{w,nom}} + w_3 \frac{T_{read}}{T_{r,nom}} + w_4 \frac{T_{write}}{T_{w,nom}} \tag{2}$$

where $E_{r,nom}$, $E_{w,nom}$, $T_{r,nom}$, and $T_{w,nom}$ are the nominal values of read energy, write energy, read time and write time, respectively. $w_{1,2,3,4}$ are positive numbers less than 1, such that

$$w1+w2+w3+w4 = 1 \tag{3}$$

At each step we calculate G for a single step change in each of w_{PU}, w_{PG}, and w_{PD}, and accept the change that yields the minimum value of G. Using different weights, we can define the relative importance of constraints. To calculate P, we need the mean and variance of V_{read} or write time distribution curve, given the mean and variances of the two length distribution curve (μ_1, σ_1, μ_2, σ_2), and a set of device widths. We use a Taylor series expansion to calculate the mean and variance of a function $y = f(l_1, l_2)$, where l_1 and l_2 are the two independent random variables representing the two length distributions.

$$\mu_y = f(\mu_1, \mu_2) + \frac{1}{2}\sum_{i=1}^{2} \frac{\partial^2 f(l_1, l_2)}{\partial l_i^2}\Big|_{\mu_i} \sigma_i^2$$
$$\sigma_y^2 = \sum_{i=1}^{2} \left(\frac{\partial f(l_1, l_2)}{\partial l_i} \right)^2 \sigma_i^2 \tag{4}$$

To verify the accuracy of this expression in our analysis setup, we calculated the value of mean and variance for the V_{read} distribution curve of Figure 8.10 for the nominal cell using (4). The mean was calculated to be 1, and the variance was 0.036 (normalized to nominal value of V_{read}), which is very close to the simulated values for the curve (mean = 1.0, variance = 0.035). The mean and variance values from the Taylor Series expansion are merely used to guide the iterative optimization in the right direction (through the sensitivity metric), and so the fact that these values are slightly inaccurate (based on approximation) does not affect the final result of the sizing optimization significantly. Hence, using Taylor series expansion is a reasonable approximation to make. A flowchart outlining the proposed SRAM sizing optimization algorithm is shown in Figure 8.17. The next section discusses the experimental results for improvement in SRAM robustness using the proposed sizing algorithm.

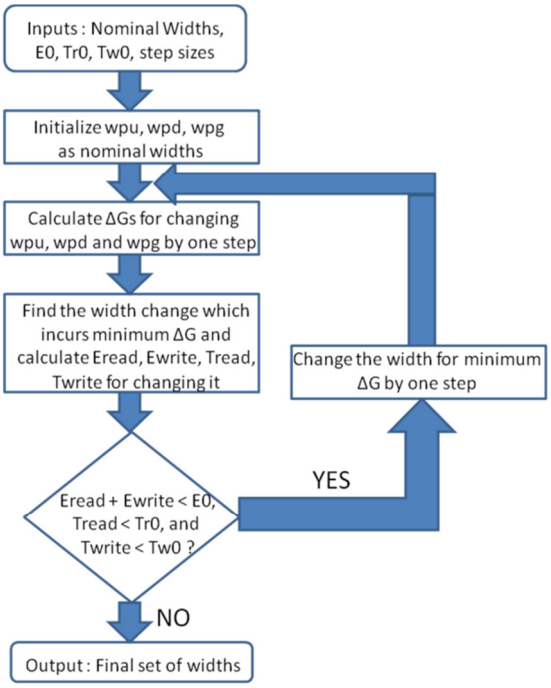


Figure 8.17 Proposed SRAM sizing optimization algorithm.

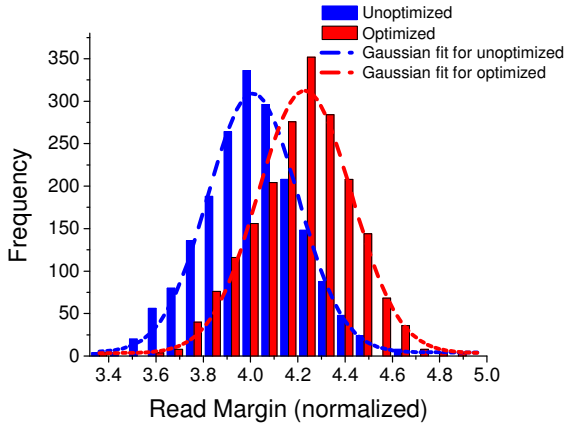


Figure 8.18 V_{th} σ failure distribution for read operation before and after the proposed optimization.

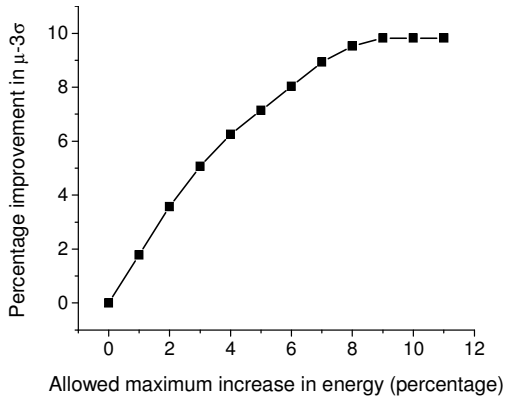


Figure 8.19 Variation of percentage improvement in the μ-3σ point of the V_{th} σ failure distribution with maximum allowed change in energy for the optimization algorithm.

8.3 Experimental Results

We use our proposed technique to optimize an industrial 45nm SRAM bitcell optimized for single patterning lithography, considering DPL based variation. For the purpose of analysis, we assume that the two gate length distribution curves (for the adjacent polysilicon tracks) have the same standard deviation, with $3\sigma_1/\ \mu_1 = 3\sigma_2/\ \mu_2 =$ 10%, and their means can differ by up to 5% ($|\mu_1-\mu_2| <=5\%$). As discussed in Section 8.2, we run our DPL-aware sizing optimization algorithm for worst case combination of mean and standard deviation that creates maximum mismatch ($3\sigma_1/\ \mu_1 = 3\sigma_2/\ \mu_2 = 10\%$, and

difference in mean is maximum = 5%), and then examine at the V_{th} σ failure improvement at intermediate values to examine the effectiveness of the algorithm in mitigating yield losses at intermediate values.

The read operation is less robust for the analyzed SRAM. Figure 8.18 shows the V_{th} σ failure number distribution for read operation before and after the application of the proposed approach. Sizing optimization shifts the curve to the right, thereby making the SRAM cell more robust (higher value of V_{th} σ failure number means lower probability of failure). The mean of the V_{th} failure number distribution for optimized SRAM is 4.22 σ_{VT0} (very close to the mean in the case of single exposure for $3\sigma/\mu = 10\%$), while the standard deviation observed is ~0.21σVT0. The absolute value of variance remains almost the same before and after the optimization (standard deviation = ~$0.21\sigma_{VT0}$ for unoptimized case), just the curve is shifted to the right through optimization. This validates the assumption made during the sizing optimization that variance of the V_{th} σ failure number distribution would not change drastically during the sizing optimization. The μ-3σ of V_{th} distribution for application of proposed sizing is $3.57\sigma_{VT0}$, which is a 6.2% improvement over the μ-3σ of the unoptimized DPL curve. This corresponds to a 2.17X reduction in failure probability of the μ-3σ point in the distribution. These values are for a maximum allowed change of 5% in $E_{read}+E_{write}$, compared to the nominal value ($E_0 = 1.05(E_{r,nom} + E_{w,nom})$). Figure 8.19 shows the variation of the percentage improvement in the μ-3σ of V_{th} σ failure number distribution achieved by the proposed optimization over unoptimized SRAM, as a function of maximum allowed normalized value of $E_{read}+E_{write}$ ($E_0/(E_{r,nom} + E_{w,nom})$). The improvement number goes up with increase in maximum allowed change in dynamic energy, and saturates for an allowed

change of ~9%. Maximum improvement in μ-3σ point is ~9.8%, which corresponds to 3.6X reduction in cell failure probability, for an SRAM cell area overhead of only 1.6%. Beyond this value, increasing the allowable energy penalty gives no further improvement because any further sizing violates the write time constraint ($T_{write} < T_{w,0}$). For this maximum improvement sizing point, μ-3σ for the write V_{th} σ failure number distribution decreases to ~$5.19\sigma_{VT0}$ from its value of ~$5.43\sigma_{VT0}$ for the unoptimized case. Even after the sizing optimization, write operation remains the more robust operation, but read robustness improves significantly.

Table 1 summarizes the percentage improvement in mean and μ-3σ points for V_{th} failure distribution curve of the optimized case over non optimized SRAM, for intermediate variation numbers ($3\sigma_1/\mu_1 <= 10\%$, $3\sigma_2/\mu_2 <= 10\%$, $|\mu_1-\mu_2| <= 5\%$), and maximum allowed change in dynamic energy of 5% ($E_0 = 1.05(E_{r,nom} + E_{w,nom})$). The improvement numbers obtained decrease as the mismatch is decreased (low variances and difference in mean). This is because there is less room for optimization, as the mean and μ-3σ values are closer to the nominal case. However, the proposed technique does ensure that we get almost all of the possible improvement given the constraints. Hence, SRAM cell optimized for worst case variation provides good improvement in SRAM robustness at intermediate points.

Next, we compare our approach to an approach where SRAM cell is over-optimized under single exposure based variation, in order to achieve better robustness under DPL based variation. For this purpose we use an algorithm similar to our proposed algorithm, but with a single length distribution curve instead of two (as in DPL). We find that in order to achieve similar robustness as the DPL aware sizing scheme, the

constraints on energy and access times have to be relaxed. Such a technique results in higher energy and slower access times as compared to DPL-aware sizing optimization for the same value of improvement over the unoptimized case. For iso-robustness, such a technique results in 7.9% higher energy (E_{read}+E_{write}), and 4.6% larger access times as compared to the proposed technique.

Table 8.1 Robustness improvement numbers for intermediate values of mean and standard deviation for DPL length distributions

DPL Length Distribution			Improvement in V_{th} failure numbers	
$3\sigma_1/\mu_1$	$3\sigma_2/\mu_2$	Mean difference	Mean	$\mu - 3\sigma$
10%	10%	5%	5.2%	6.2%
10%	10%	3%	3.1%	4.3%
10%	10%	1%	1.0%	2.4%
10%	10%	0%	0.1%	1.5%
9%	10%	3%	3.2%	4.8%
10%	8%	2%	2.1%	3.5%

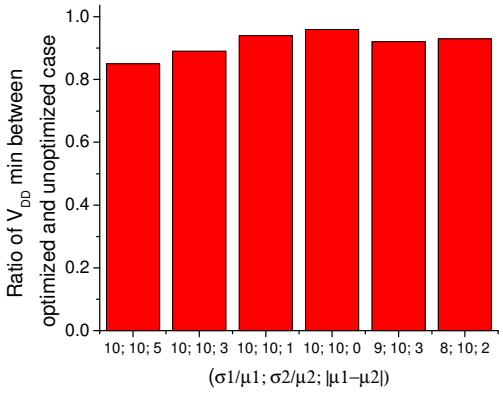


Figure 8.20 Ratio of minimum V_{DD} allowed in optimized and unoptimized case for a variety of mean and standard deviation combinations for the gate length distributions.

Finally, we analyze the improvement in voltage scalability of the SRAM using the proposed technique. We fix the desired mean of the V_{th} σ failure number distribution to be greater than $3.5\sigma_{VT0}$, and calculate the V_{DD} to which the SRAM can be scaled given the two length distribution curves before the mean goes below the desired value, with and without the application of proposed sizing optimization. Figure 14 plots the ratio of minimum V_{DD} allowed in the optimized and unoptimized case for a variety of mean and standard deviation combinations of the two gate length distribution curves. On an average, the proposed technique improves V_{DD} scalability by 9.5% (18.1% reduction in energy). Thus, the proposed DPL-aware sizing optimization approach is shown to effectively mitigate the yield loss at a small penalty.

8.4 Summary

Pitch-split double patterning lithography results in adjacent devices with different mean critical dimension (CD), and uncorrelated CD variation. Such a variation can increase functional failures in SRAM cells, which are very sensitive to mismatches, and degrade yield. In this chapter, we analyze the impact of DPL on parametric functional failures in SRAM cells and propose a DPL-aware SRAM sizing scheme to effectively mitigate the yield losses for a very small energy and area overhead. Experimental results based on 45nm models show that DPL can significantly impact the SRAM cell robustness and hence there is a need for DPL aware analysis and optimization of SRAM cells. The proposed technique is very effective in mitigating the negative impact of DPL on SRAM robustness, and can improve V_{th} failure numbers by up to 9.8%, which translates to a 3.6X reduction in SRAM cell failure probability, for a very small area penalty of 1.6%.

Chapter 9
Design-Patterning Co-optimization of SRAM Robustness for Double Patterning Lithography

As discussed in the previous chapter, double patterning lithography (DPL) provides an attractive optical lithography solution for 32nm and subsequent technology nodes. There are two primary DPL techniques: pitch-split double patterning (PSDP) and self-aligned double patterning (SADP), which can be implemented using a positive tone or a negative tone process [16,17,18,19,20]. Each DPL implementation has a different impact on line space and linewidth variation, and by analyzing the impact of these different DPL options the best overall process flow can be achieved. Figure 9.1 shows the process flows for these two DPL techniques. PSDP partitions a critical-layer layout into two mask layouts and exposures, each having to resolve only half the ultimate pattern pitch. Figure 1 (top) depicts the process flow for pitch split: a layout is decomposed by distributing alternating features onto two photo-masks and then exposing these two masks sequentially with an optical isolation process step, such as a transfer etch (litho-etch-litho-etch [LELE] [17]) or a resist freeze (litho-freeze-litho-etch [LFLE] [9]). A major limitation of the pitch-splitting approach to double-patterning is the inevitable overlay error between the two exposures. Since the two optically isolated images that form the final wafer pattern are exposed independently, mask placement, alignment, and

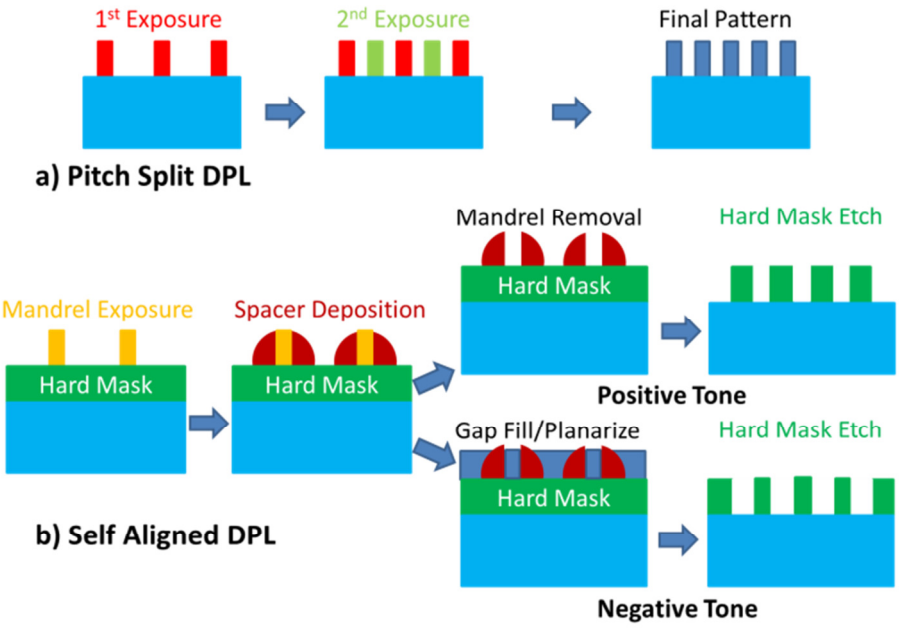


Figure 9.1 Double Patterning Process Flow.

magnification errors will cause the space between neighboring features to be adversely affected by potentially a significant amount [116]. Additionally, in pitch-split DPL either lines or spaces between lines are printed in two sequential processes. Thus, pitch-split DPL is characterized by the existence of dual populations for critical dimension (CD), with uncorrelated variance and distinct means.

SADP, on the other hand, exhibits excellent variability control by using only one lithographic patterning process coupled with the deposition of a spacer to double the pattern frequency, thereby making the critical dimension immune to system overlay [20]. As shown in Figure 9.1 (bottom), pitch doubling is achieved by depositing a sidewall spacer onto a core mandrel shape; since spacer is deposited on both sides of the mandrel, pitch of the deposited sidewalls is half that of the mandrels from which they are formed. Since the critical dimension is defined by sidewall deposition and not the lithography step, SADP provides better CD control. However, this also limits the entire layout to one

critical dimension. Hence, it may be difficult to print irregular patterns using SADP. PSDP can be applied with either negative (N-PSDP) or positive (P-PSDP) tone process, each having a different impact on line space and linewidth. In case of SADP, generated spacers act analogously to photoresist and can define either spaces or lines. As shown in Figure 9.1, positive tone SADP (P-SADP) is defined as the process where trenches are generated in the area not under spacers, and negative tone SADP (N-SADP) is defined as the process where trenches are generated in the area under spacers.

Previous chapter showed that pitch split DPL based gate length variation can have a significant negative impact on SRAM robustness, and presented a DPL aware sizing technique to mitigate the yield loss. This chapter extends the analysis and optimization framework discussed in the previous chapter to address the case when multiple DPL choices are available for printing each layer, and an optimal approach is desired for increased SRAM robustness. This chapter presents a comprehensive analysis and optimization framework which compares the layer-wise impact of each of these patterning choices on SRAM robustness, density, and printability; and performs a sizing optimization that accounts for increased variability due to DPL for each layer.

We use extensive simulation to analyze the impact of DPL-based variation on SRAM robustness, area and printability. We show that it is important to compare the impact of using different DPL techniques in order to choose the best option for each layer. It then performs a sizing optimization that accounts for increased variability due to DPL for each layer.

The rest of the chapter is organized as follows. Section 9.1 discusses relevant background, while Section 9.2 presents a layerwise analysis of DPL impact on SRAM

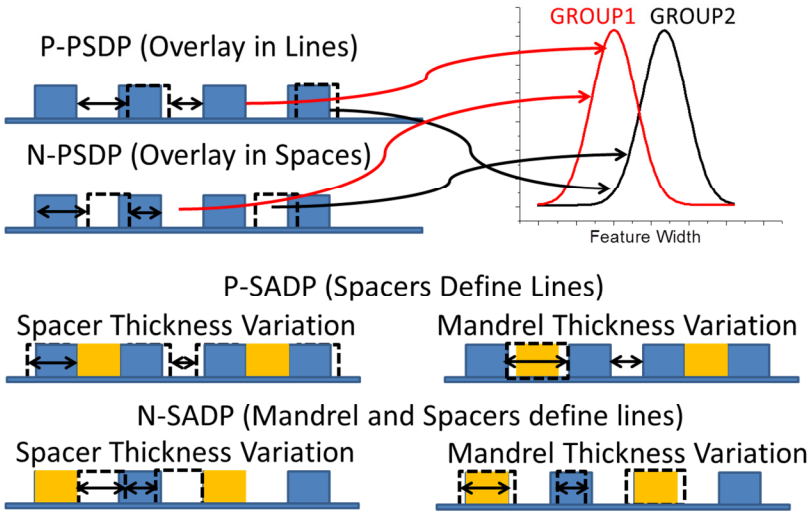


Figure 9.2 Impact of DPL techniques on linewidth and line space.

cell robustness, area and printability. Section 9.3 presents the DPL-aware SRAM sizing technique, and Section 9.4 concludes the chapter.

9.1 Background

Figure 9.2 shows the impact of different DPL techniques on linewidth and line space variation, for a given layer. For any given layer, there are four DPL options to choose from:

P-PSDP: As shown in Figure 9.2, overlay affects pitch and line space, but not the linewidth. However, since the lines are printed in two separate exposures, systematic offsets between the two exposures cause the existence of dual CD populations with uncorrelated variances and distinct means (μ). Based on hardware results, [114] reported 3σ/μ numbers as high as ~16.5% for P-PSDP CD distributions in 32nm technology, with an 8% difference in mean CD for the two linewidth distributions.

N-PSDP: Figure 9.2 shows how overlay impacts linewidth, but not line space and pitch in the case of N-PSDP. This coupling of overlay to linewidth variation increases the overall CD variation, and makes N-PSDP a less attractive DPL option for layers that

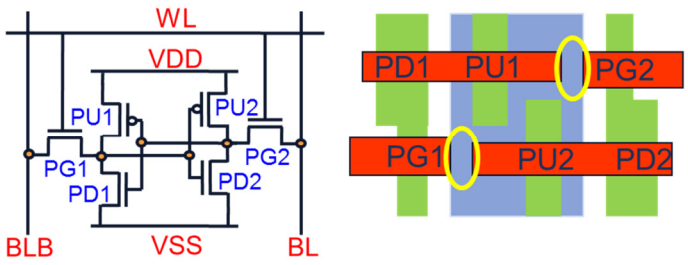


Figure 9.3 Schematic and layout of a six transistor SRAM cell.

require tight CD control. N-PSDP prints spaces in two separate exposures, and that causes the existence of dual line space (and hence CD) populations with uncorrelated variances and distinct means.

P-SADP: SADP combines a lithography step with a deposition step to double the pattern frequency; therefore the critical dimensions are immune to system overlay. However, it experiences some CD variability due to spacer thickness variation as shown in Figure 9.2. In case of P-SADP, spacer thickness variation affects both linewidth and line space. Mandrel thickness variation impacts line space only. Since, CD is defined by spacers grown in a single deposition step, no dual CD population exists in the case of P-SADP. However, as mentioned earlier, P-SADP limits the entire layout to one critical dimension, and that might not be desirable for certain layers.

N-SADP: N-SADP defines alternate lines using mandrel, while the rest are defined using mandrel and spacer. As shown in Figure 9.2, spacer thickness variation changes linewidth of alternate lines, and line space. Mandrel thickness variation affects two adjacent lines oppositely; an increase in mandrel thickness will increase the width of one set of lines defined by the mandrel, while it decreases the printed width of the other set of lines. This leads to the existence of dual CD populations in the case of N-SADP, similar to PSDP.

N-SADP allows multiple linewidths unlike P-SADP, thereby providing more design flexibility.

9.2 Layerwise DPL based Analysis of SRAM

In this section, we present a layerwise analysis of the impact of different DPL-techniques on SRAM robustness, printability, and area to help decide the best technique for each layer. The analysis presented is based on an industrial 45nm SRAM cell, originally optimized for single exposure lithography.

9.2.1 Polysilicon Layer

Figure 9.3 shows the schematic and conventional layout of a typical six transistor SRAM cell. The access transistor and pull up/pull down (PU/PD) transistors for a given side lie on different poly tracks (e.g., PG1 lies on a different track than PU1/PD1). A convenient method to analyze SRAM robustness is through corner-based failure analysis where each device is skewed in its worst direction to induce a particular failure mode, e.g., strong access devices lead to read failures. SRAM robustness is characterized in terms of the smallest multiple of sigma V_{th} (specified for a given technology) at which the cell experiences functional failure, and this value is called read/write margin. Higher values of read/write margin imply that a cell can tolerate higher levels of random V_{th} variation before experiencing functional failure.

As studied in Chapter 8, using PSDP for the polysilicon layer causes the existence of dual (bimodal) CD populations, which can have a significant negative impact on SRAM robustness due to increased mismatch between devices (e.g., PG1 lies on a different track than PU1/PD1). For example, if the access transistor (PG) becomes stronger than the pull down transistor, the SRAM cell becomes more prone to read failures. N-PSDP can potentially have a bigger negative impact on robustness as

compared to P-PSDP, since it couples overlay to linewidth variation. Another printability concern to address is printing the line ends (marked in Figure 9.3), since using PSDP alone the space between tip to tip and tip to line should be at least 80nm to ensure printability [117]. To achieve smaller spacing (for improved SRAM density), additional techniques such as cut-mask [118] might be required to define line ends. This incurs higher cost due to additional process steps.

Figure 9.4 shows the layout decomposition using P-SADP. Since the polysilicon gates are defined by sidewall deposition and spacer trim mask, no dual CD population is observed unlike PSDP. Spacer thickness variation can cause CD variation, while mandrel

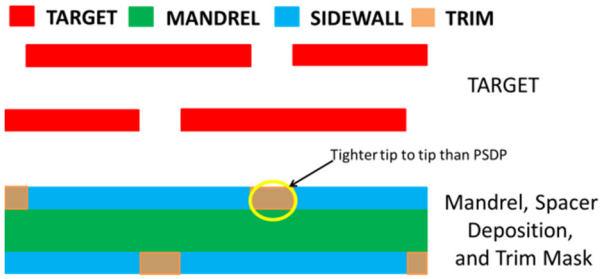


Figure 9.4 Polysilicon decomposition using P-SADP.

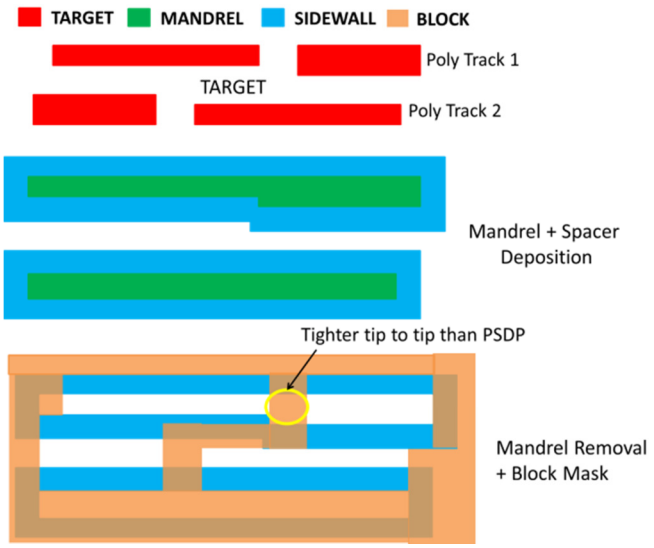


Figure 9.5 Polysilicon decomposition using N-SADP.

thickness variation has no impact on CD and only affects the line space. However, since spacer thickness defines gate length, all the devices are constrained to a single gate length value. This can cause some negative impact on robustness, since SRAM cells optimized for single exposure typically use longer gate lengths for the access transistor to improve read/write margins. As shown in Figure 9.4, line ends are printed using the original mandrel and trim mask, and no additional process step is needed (cut mask) for tip to tip distance more than ~40nm [117]. Thus, tip to tip values can shrink to almost half of that permitted by PSDP alone without any additional process steps.

Figure 9.5 shows the layout decomposition using N-SADP. With the use of block mask and mandrel, multiple gate lengths can be printed using N-SADP. Similar to P-SADP, here line ends can be printed using block mask, and tip to tip distance of ~40nm can be printed at no additional cost. However, unlike P-SADP, there is some mismatch between devices on adjacent polysilicon tracks. As shown in Figure 9.5, devices on polysilicon Track 1 are defined by the mandrel, while devices on Track 2 are defined jointly by the mandrel and the spacers. So, if devices on Track 1 become stronger due to mandrel thickness variation (smaller gate length); devices on Track 2 will be weaker because a decrease in mandrel thickness will increase their printed gate length. Additionally, devices on Track 2 might experience added variability due to spacer thickness variation. This inverse correlation in gate length between adjacent polysilicon tracks will negatively impact the SRAM robustness. One final concern with using N-SADP for polysilicon layer is based on the fact that the same technique would be used for printing the polysilicon in logic. Typical polysilicon width to space ratio for logic is

about 1:3 [119], and that would mean that the sidewall may have to be thrice the thickness of the mandrel. This might be a challenge from the process point of view.

In order to analyze the effect of different DPL techniques on SRAM parametric failures, we begin with an analysis of the failure triggering mechanisms under DPL based mismatch. A read failure is defined as flipping of the stored data in the cell while reading. Flipping occurs when the bump in stored '0' voltage is higher than the trip point of the opposite inverter (e.g., when the bump in the output of inverter PU2-PD2 (V_{read}) > V_{trip} for inverter PU1-PD1, while reading out a '0'). Write failure is defined as a failure to write to a cell within the time when wordline (WL) is high. As discussed earlier, PSDP results in two linewidth distributions with uncorrelated standard deviations (σ_1, σ_2) and distinct means (μ_1, μ_2). For P-PSDP, we assume a conservative scenario of equal means ($\mu_1 = \mu_2$), with $3\sigma_1/\mu_1 = 3\sigma_2/\mu_2 = 10\%$, for the two linewidth distribution curves. Based on ITRS estimates for PSDP overlay [119], we assume a similar scenario of equal means for N-PSDP, but with an increased variability $3\sigma_1/\mu_1 = 3\sigma_2/\mu_2 = 15\%$, due to coupling of CD variation to overlay. To simulate P-SADP, we use sidewall thickness 3σ of 1.6nm (based on ITRS [119]), and constrain the cell to have a single gate length. Finally, N-SADP is simulated by assuming sidewall thickness 3σ of 1.6nm, mandrel thickness variation 3σ/μ of (equal to the value for individual PSDP exposures).

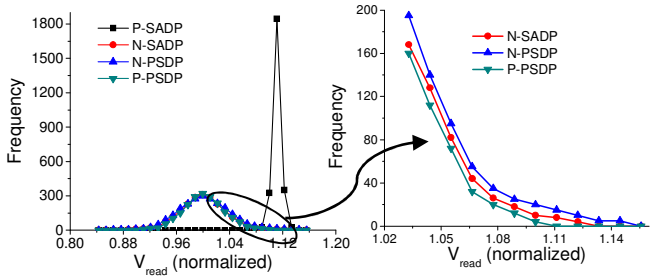


Figure 9.6 V_{read} distributions for different DPL techniques.

Table 9.1 Read distribution characteristics for different DPL techniques

DPL Technique	μ (normalized)	σ (normalized)	Failure probability of μ-3σ (normalized)
P-PSDP	4.20	0.20	1
N-PSDP	4.22	0.31	3.04
N-SADP	4.21	0.24	1.46
P-SADP	3.53	0.03	1.79

V_{read} is defined as bump in the output voltage of inverter PU2-PD2, while reading out a '0'. Higher values of V_{read} indicate that a cell is more prone to read failure under random V_{th} variation. Similarly, a cell with longer write time will be more likely to experience write failure under random V_{th} variation. Figure 9.6 shows V_{read} distributions for the four DPL techniques, along with a zoomed in view of the distribution tail for N-SADP, N-PSDP, P-PSDP. All V_{read} values are normalized to V_{read} for the nominal cell. N-PSDP shows maximum variability in V_{read} and write time distributions, due to dual CD populations, along with coupling of overlay to CD variation. The mean and variance of V_{read} distribution for P-PSDP is found to be ~1 and 0.035 (normalized to nominal V_{read}), while the mean and variance for N-PSDP are ~1 and 0.053, respectively. For the write time distribution, P-PSDP has a mean value of ~1 and a standard deviation of 0.017, while for N-PSDP mean and standard deviation are ~1 and 0.025, respectively (normalized to nominal write time). In both cases, N-PSDP increases the variability by ~1.5× compared to P-PSDP. For SADP techniques, P-SADP based distributions shows lower variance than N-SADP, since only spacer thickness variation impacts CD, unlike N-SADP where a mismatch is observed between devices on adjacent poly tracks due to mandrel and spacer thickness variation. However, P-SADP shows a positive shift in V_{read} mean (negative impact on read robustness) due to the gate lengths being constrained to a

single CD value; the write time mean on the other hand decreases (positive impact on write robustness). The mean and variance of V_{read} distribution for P-SADP are 1.11 and 0.004 (normalized to nominal Vread), while the mean and variance for N-SADP are ~1 and 0.039, respectively. For the write time distribution, P-SADP has a mean value of 0.96 and a standard deviation of 0.004, while for N-SADP mean and standard deviation are ~1 and 0.02, respectively (normalized to nominal write time). N-SADP shows higher variability than P-PSDP because of the inverse correlation between gate lengths on adjacent polysilicon tracks, which increases variability.

Based on a V_{th} corner analysis, the nominal cell experiences read failure at a V_{th} σ value of $4.23\sigma_{VT0}$, and write failure σ value is $6.36\sigma_{VT0}$, where σ_{VT0} is the standard deviation of intra-die Vth variation specified for the technology. These numbers establish that, in general, write operation is much more robust for the industrial SRAM being analyzed, and so we focus on the read margin distribution for our analysis. Table 9.1 summarizes the impact of different DPL techniques on the read margin distribution. The mean (μ) and standard deviation (σ) values are normalized to σ_{VT0}. Also reported are the failure probability numbers for the μ-3σ point on the distribution (normalized to failure probability for P-PSDP). Even though P-SADP is the lowest variability solution, it has a negative impact on read robustness due to the CD being constrained to one value. The write robustness on the other hand is improved. However, the overall SRAM robustness is read constrained (nominal write margin > read margin). So, P-PSDP provides the best (most robust) solution for nominal SRAM. However, due to its lower variability, P-SADP may potentially provide a more robust solution by resizing the cell to improve read robustness at the cost of write robustness. The sizing optimization discussed in Section

9.3, increases the mean of the P-SADP read margin distribution. Post sizing optimization P-SADP is clearly the better technique for printing polysilicon layer, as it provides lower variability (higher robustness) and tighter tip to tip distance for line ends (lower cell area), as compared to PSDP.

9.2.2 Metal 1 Layer

DPL based overlay in Metal 1 (M1) layer has been shown to have a very small impact on logic cell delay. In [120], a DPL aware M1 analysis showed less than 2% impact of overlay on cell delay. This is because the coupling capacitance between poly gate and contact or metal is very small compared with gate capacitance. We conduct a similar analysis on SRAM cells by introducing maximum overlay and CD variation (based on N-PSDP numbers) in the M1 layer, and perform SPICE-based analysis on the circuit after parasitic extraction. We measure both read/write delay and read/write margin of the SRAM cell for all possible combinations of CD and overlay variation. The maximum impact on read/write delay was found to be 0.7%, while maximum impact on read/write margin was 0.3%. Hence, we can conclude that overlay and CD variation should not be deciding factors for choosing one DPL technique over another. Instead, the focus should be on ease of layout decomposition, and minimization of cell area. Since overlay has been shown to have a small impact on delay and robustness, we can consider using PSDP techniques that exhibit higher overlay and variability than SADP. Reference [121] presents the M1 layout for a typical 6-transistor SRAM cell in 65nm, and the M1 layout for subsequent technologies is considered to be a scaled identical version.

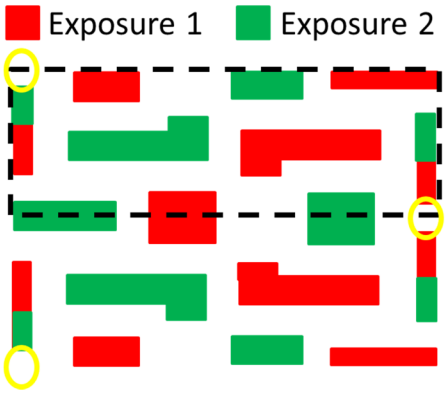


Figure 9.7 M1 decomposition using P-PSDP.

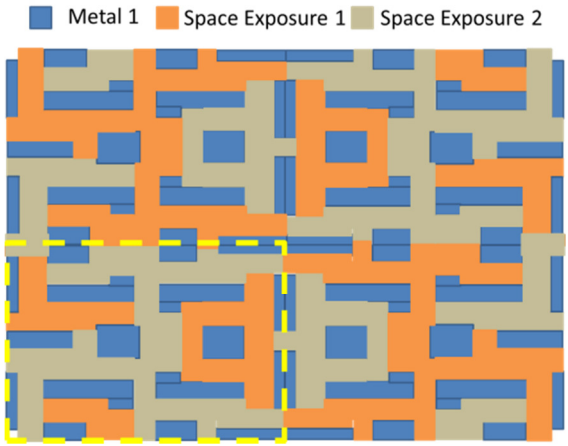


Figure 9.8 M1 decomposition using N-PSDP.

Figure 9.7 shows the layout decomposition of the M1 layer using P-PSDP for two SRAM cells placed side by side. Layout decomposition appears trivial at the cell level (a single cell is highlighted in the figure) and can be achieved by introducing a few stitches, however the analysis is complicated when we consider how the cells are abutted. Each cell is a mirror image of the cell next to it. As circled in the figure, there will be a tip to tip distance violation when we consider the neighboring cells. This can be addressed by area relaxation on the edges, such that the tip to tip distance is increased enough to be printed reliably using single exposure. Based on references [117] and [122], it can be concluded that single exposure can print tip to tip distances greater than ~80nm. Relaxing

the area such that the tip to tip distance is 90nm, results in an bitcell area penalty of ~5.6% for decomposition using P-PSDP.

Figure 9.8 shows the layout decomposition using N-PSDP for four SRAM cells, where we try to decompose spaces instead of lines. One single cell is highlighted in the layout, but similar to P-PSDP, the analysis is complicated when we consider the neighboring cells. However, no additional area increase is required for layout decomposition using N-PSDP. An important point here is the fact that since we are decomposing spaces which merge with each other so as to create isolated metal lines, there will be a lot more stitches introduced in N-PSDP based decomposition. Stitches are considered as an overhead for decomposition of lines using P-PSDP, as they can cause yield loss due to overlay between the two masks, and increase the manufacturing cost due to the resulting requirements for tight overlay control [123,124]. However, in the case of N-PSDP, this is not such a major concern since there is ample space for overlap between the spaces for Exposure 1 and Exposure 2, thereby making the layout more immune to overlay and negating the negative impact of stitches. Hence, N-PSDP is the better PSDP solution for M1 layer since it provides lower area decomposition solution, and the SRAM is not sensitive to overlay and CD variation in the M1 layer.

Next we look at possible decomposition using SADP options. Looking at the layout, we can rule out decomposition using P-SADP, due to the existence of multiple linewidths, and the existence of irregular patterns in the layout. P-SADP constrains the linewidths to one value determined by the spacer width. Constraining the M1 shapes to one width value such that the layout is decomposable using P-SADP (increase area to help address irregularity), will result in significant area increase (~24% increase in cell

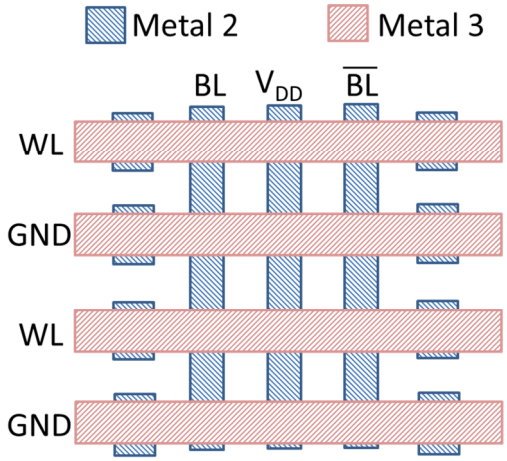


Figure 9.9 Metal 2 and Metal 3 layout in the SRAM array.

area). Hence, P-SADP is not an attractive option for M1 decomposition. N-SADP on the other hand permits multiple linewidths using block mask in addition to the spacers. However, the irregularity in the M1 layout still requires significant area increase before the layout can be decomposed using N-SADP (~13% increase in cell area). Most significant area increase comes from the need to space out the two layout features with stitches in Figure 7, due to the fact that features cannot be 'stitched' as in pitch-splitting. Since, CD uniformity and overlay are not important for M1, PSDP is clearly the best technique to use for printing M1 patterns, with N-PSDP being the better PSDP solution.

Figure 9.9 shows the layout of SRAM cell showing Metal 2 (M2) and Metal 3 (M3) layers. Bit lines (BL and BL_bar), VDD, are laid out using M2, along with blocks of M2 used to make contacts with M3 layer running in the perpendicular direction. Since the layout is fairly regular, it can be implemented using any of the four DPL techniques. Bit line capacitance will have a direct impact on read time, hence the technique which ensures minimum variability in capacitance and hence read delay will be desired. Another consideration will be the number of linewidths allowed. P-SADP will permit

only one linewidth (defined by spacer thickness), which might be undesirable for M2 since VDD is routed in this layer and is typically a long wire. It is preferable to have the option of wider wires to lower resistance and improve electromigration properties for long wire such as VDD, while employing minimum wire widths for other shorter signals to minimize capacitance. As a result, P-SADP is less suited for M2 and higher metal layers. However, as discussed earlier, both SADP techniques provide almost half the tip to tip (and tip to line) spacing as compared to PSDP, at no extra cost. Hence, SADP can decrease the layout area.

We use Predictive Technology Models (PTM) [20] to assess the impact of line space and linewidth variation due to DPL on wire capacitance and wire resistance. These values were then used in SPICE-based simulation of SRAM cell to determine the impact on speed and robustness of SRAM. For our analysis, we use interconnect technology parameters based on ITRS numbers for 45nm technology, also used in [13] which focuses on interconnect analysis for logic. Based on SPICE simulations, read delay is shown to

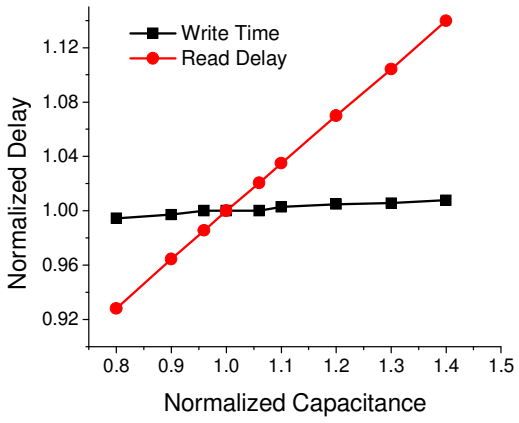


Figure 9.10 Read and write delays as a function of bit line capacitance.

Table 9.2 Bit line capacitance and read delay variation for different Double Patterning Techniques.

DPL Technique	C_{max} (normalized)	C_{min} (normalized)	Max. Delay (normalized)	Min. Delay (normalized)
P-PSDP	1.05	0.96	1.02	0.98
N-PSDP	1.12	0.89	1.04	0.96
P-SADP	1.03	0.98	1.01	0.99
N-SADP	1.02	0.99	1.01	1.00

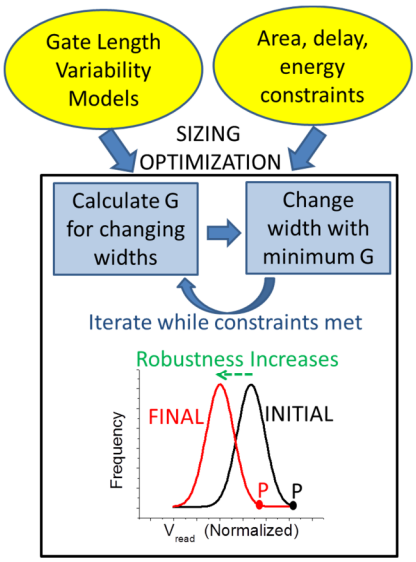


Figure 9.11 DPL-aware SRAM sizing Framework.

have a near linear dependence on bit line capacitance, as depicted in Figure 9.10. Write time on the other hand is less sensitive to variation in bit line capacitance. Table 9.2 summarizes the maximum and minimum value of bit line capacitance (normalized to the nominal value), along with the impact on read delay (normalized to nominal read delay), for different DPL techniques. For each DPL technique, we consider the best and worst case combination of overlay and interconnect CD variation to determine the impact on total capacitance. As expected, N-PSDP has the maximum variability in bit line capacitance, and hence read delay. Both SADP techniques show very small variability in both capacitance and read delay. However, since N-SADP allows multiple linewidths, it

is the best candidate for printing M2 layer. Since the bit line is interdigitated with VDD, dual CD population in P-PSDP has a smaller impact on capacitance variation [124].

9.2.3 Metal 3 Layer and Contact/Via

Typically, a checkerboard PSDP has been used to print contacts/via reliably using DPL [125]. Via area enclosed by metal layer can be reduced due to overlay, increasing via resistance. However, impact of via resistance increase has been shown to be negligible on timing in [120], where a 2× increase in resistance was shown to affect timing by 0.1%. Based on these works, we can conclude that any technique that reliably print via/contacts suffices, and no SRAM specific analysis is required. As shown in Figure 9.9, ground (GND) and word line (WL) are laid out using Metal 3 (M3). Capacitance variation observed is similar to the case of M2, due to interdigitated GND and WL minimizing the impact of overlay in P-PSDP. In the case of M3, we would also like to have the option to print multiple wire widths since ground line runs in M3. So, P-SADP is not desirable for M3 layer. Amongst the other techniques, we can use N-SADP since the layout of M3 layer is regular, and N-SADP results in lowest variability impact on capacitance. The next section discusses SRAM sizing optimization which uses layer wise DPL variability information to optimize the read robustness.

9.3 DPL-aware SRAM Sizing Framework

We use a DPL-aware SRAM sizing approach, similar Chapter 8, to optimize the nominal SRAM cell for better robustness. As shown in Figure 9.11, inputs to the sizing optimization include DPL technology specific gate length variability models, and constraints on read/write energy, read/write delay, total cell area. These constraints can be updated based on the DPL technique used for other layers, to incorporate DPL based variability information for all the layers. For example, if P-PSDP is used to print M1

layer, it has an additional bitcell area penalty of ~5.6% as compared to using N-PSDP. This information can be used to tighten the area constraint accordingly. Since nominal write margin ($6.36\sigma_{VT0}$) is much higher than nominal read margin ($4.23\sigma_{VT0}$), the sizing optimization focuses on improving read margin. As discussed in Section 9.2, read failure occurs when bump in the read voltage (V_{read}) is higher than the trip point of the other inverter. V_{read} is defined as bump in the output voltage of inverter PU2-PD2 (Figure 9.2), while reading out a 0 from SRAM. Higher the value of V_{read}, lower is the value of read margin. We improve read margin by shifting the V_{read} distribution to the left, as shown in Figure 9.11. This is achieved by iteratively sizing the SRAM cell to shift a candidate point P ($\mu+3\sigma$) on the V_{read} distribution to the left, as long as the constraints are met. Only the widths of the pull up, pull down and access transistors are used as optimization variables. We define a sensitivity metric G ($\Delta C/ \Delta P$), based on the decrease in the value of point P (ΔP) and change in the value of a constraint function C (ΔC). At every iteration step, the width change which yields the minimum value of G is chosen.

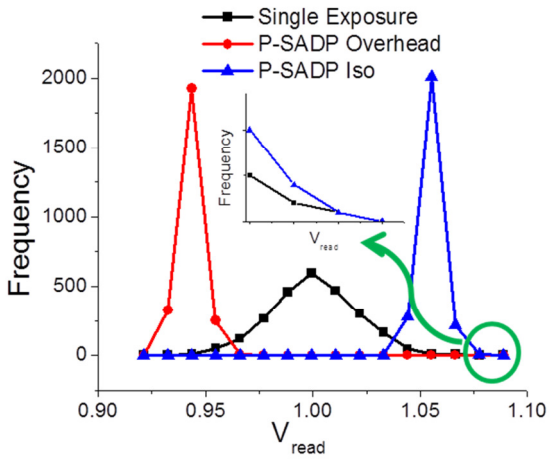


Figure 9.12 V_{read} distributions for optimized P-SADP and single exposure lithography.

Table 9.3 Post sizing overhead and robustness numbers for different DPL implementations of polysilicon gates in SRAM.

Lithography Technique	μ-3σ (normalized)	Failure Probability	Area Overhead	Energy Overhead
Single Exposure	3.90	1.0X	0%	0%
P-SADP iso	3.86	1.2X	0%	0.2%
P-SADP overhead	4.30	0.2X	0%	6.3%
P-PSDP	3.75	1.9X	1.6%	6.9%
N-SADP	3.63	3.0X	2%	6.8%
N-PSDP	3.50	4.8X	2%	7%

As seen in Section 9.2.1, P-SADP provides the lowest variability solution for polysilicon gates, but it has negative impact on read robustness due to the CD being constrained to one value. So, while the standard deviation of V_{read} distribution is very small for P-SADP, it has a high value of mean for the nominal case. We perform a sizing optimization for P-SADP to shift the V_{read} distribution to the left, in order to exploit the low variability in P-SADP to achieve a more robust solution. Figure 9.12 shows the V_{read} distributions for two optimized versions of the SRAM cell, along with the distribution for single exposure case assuming 3σ/μ for gate length variation is 10%. The first optimized curve (P-SADP iso) is for the iso-area and iso-energy (less than 0.2% impact on energy) optimized cell, while the second curve (P-SADP overhead) allows a 6.3% increase in write energy and 2.5% increase in write time. While the P-SADP iso curve very closely matches the μ+3σ point of the single exposure case (shown in the zoomed plot, ~1% difference in the μ+3σ points), P-SADP overhead provides higher robustness than single exposure for no area penalty (10.2% improvement in the Vread μ+3σ point). We characterize read robustness in terms of the failure probability of the μ-3σ point on the read margin distribution. P-SADP iso has a failure probability of 1.2X (normalized to single exposure failure probability), while P-SADP overhead has a normalized failure

probability of 0.2X. Thus, by choosing the optimal combination of DPL techniques, and performing the proposed sizing optimization of the SRAM cell, we can achieve the same robustness as single exposure lithography, with the improved printability of DPL. For a small overhead, we can achieve a more robust solution as compared to single exposure.

Table 9.3 summarizes read robustness along with area and energy overheads for all the possible DPL implementations of polysilicon, where μ-3σ values are normalized to σ_{VT0}, failure probabilities are normalized to single exposure failure probability, and all overhead numbers are expressed as percentage increase from the nominal cell optimized for single exposure. The maximum allowed change in energy is 7%. P-SADP provides much better robustness than any other DPL solution, none of which can improve the failure probability of the μ-3σ point to less than 1.9X that of the single exposure case even after incurring a 2% bitcell area penalty. So, based on sizing results and the analysis presented in Section 9.2, the optimal DPL approach is to print polysilicon layer using P-SADP, M1 using N-PSDP, M2 and M3 using N-SADP. Any deviation from this optimal assignment will result in tighter sizing optimization constraints and hence lower robustness. As shown in the results, such an assignment can achieve single exposure robustness, and improved DPL printability at almost no overhead.

9.4 Summary

DPL can be implemented as positive/negative tone spacer patterning or pitch splitting, with each DPL implementation having different impact on line space and linewidth variation. In this chapter, we use extensive simulation to analyze the layer-wise impact of DPL-based variation on SRAM robustness, area and printability, and show that it is important to compare the impact of using different DPL techniques in order to choose the best option for each layer. A DPL-aware SRAM sizing technique is presented

that incorporates DPL based variability information for each layer into one sizing flow. Experimental results based on 45nm industrial models show that using the best DPL option for each layer, and performing the sizing optimization presented, we can achieve single exposure robustness, for improved DPL printability at almost no overhead. Cell failure probability can be further improved to 0.2X the single exposure failure probability, for a small overhead.

Chapter 10
Modeling TSV-Induced Mechanical Stress to Enable TSV-Aware Timing Analysis

Three-dimensional (3D) stacking is an emerging integrated circuit (IC) integration technique that presents a viable solution to meet the scaling targets on performance, power dissipation, functionality, and packaging form factor [126,127,128]. 3D integration stacks multiple dies interconnected in the vertical direction using through-silicon vias (TSVs) to achieve wirelength reduction and increased density. However, 3D stacking requires modifications to the electronic design automation (EDA) tools to enable 3D IC design. For example, some key challenges that need to be addressed are floorplanning, thermal modeling, parasitic extraction, stress modeling (timing analysis), etc [129]. TSV materials and silicon have different coefficients of thermal expansion generating mechanical stress in the devices, which in turn affects the device mobility and hence performance. Figure 1 shows the map of longitudinal stress (S_{xx}) for a single isolated copper TSV, along with a plot of stress as a function of distance along the x-direction. As shown in the figure, longitudinal stress can vary from +100 MPa (tensile) to -100MPa (compressive), which in turn can change hole mobility by ~12%, and significantly impact gate delay. Hence, accurate, closed-form TSV stress modeling is needed to enable TSV-aware timing analysis for 3D ICs.

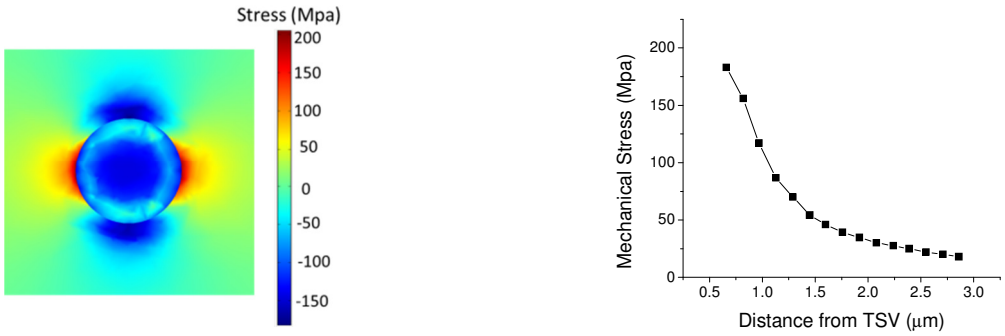


Figure 10.1 Longitudinal stress for an isolated Copper TSV.

In the relevant literature, [130] focused on 3D placement, routing, and floorplanning, while [131] studied the impact on TSV stress on reliability. However, there has been limited work on comprehensive closed-form modeling of TSV-induced mechanical stress, and its impact on circuit timing. Reference [132] presented analytical maps of TSV stress that were used for TSV-aware timing analysis. However, the proposed analytical models do not account for important layout features such as active area length, shallow trench isolation (STI), and the presence of neighboring devices. Additionally, it does not consider separate piezoresistive coefficients for transverse and longitudinal direction, while converting stress to mobility, which further degrades the accuracy of simulation-based results.

This chapter proposes compact closed-form models for TSV-induced mechanical stress, and its impact on carrier mobility. The model derivation is based on the physics behind stress inducing process steps, and accounts for layout features such as STI width, active area length, and neighboring devices. Comparison with finite element analysis based simulations shows that the proposed models accurately capture TSV stress. We then propose a new TSV-aware timing analysis framework that embodies transistor-level models for TSV stress sensitivity, to incorporate TSV induced stress/mobility variation

into final circuit timing sign-off. Section 10.1 presents the proposed stress models, while Section 10.2 presents the proposed TSV-aware timing analysis flow, and gate/circuit level timing results are discussed in Section 10.3.

10.1 Modeling TSV Stress and Impact on Device Mobility

To enable TSV-aware timing analysis, the mechanical stress induced in the device channel must be known, which can then be translated into impact on carrier mobility. This section first presents closed-form TSV stress models to enable fast and accurate stress modeling. The second part of the section provides model verification by comparing to finite element analysis based simulations.

10.1.1 Stress Models

TSVs generate stress due to thermal mismatch with silicon. The difference in the thermal expansion coefficients causes stress to develop in the device once the wafer is cooled down post electroplating. Tungsten (W) and Copper (Cu) have been considered as TSV fill materials, however Cu has emerged as the more widely used option due to its low resistivity. Copper has a higher coefficient of thermal expansion than silicon, which results in the generation of tensile stress in the longitudinal direction upon cooling. Due to high stress and possible thermo-mechanical reliability issues (e.g., cracks), a region around each TSV is typically defined as a Keep Out Zone (KOZ) where cells cannot be placed. We model thermal mismatch stress due to TSV and STI. For each device, we consider all features within a certain window of influence (of length LW) to calculate the resulting stress. The model is derived under multiple simplifying assumptions and the actual parameter values come from calibrating the model to simulation or measurement data.

For a given segment (material M) of length $L_{x,y}$, the change in length ($\Delta L_{x,y}$) upon heating/cooling can be quantified as:

$$\Delta L_{x,y} = \alpha_M \Delta T L_{x,y} \tag{1}$$

where α_M is the coefficient of thermal expansion for material M, and ΔT is the change in temperature. This is the change in length that would occur when the material segment is in isolation. Such a scenario does not cause any mechanical stress to develop. However, the presence of neighboring features with different coefficients of thermal expansion restricts the amount by which a given segment length can change, and this deviation from the stress free (isolated) case leads to the development of mechanical stresses. We consider each layout segment as a spring (or an elastic beam) characterized by different elasticity. We can then express stress generated in each segment as a function of deviation from the stress-free isolated scenario. It is assumed that displacements at the ends of the considered window (leftmost and rightmost edges) are equal to zero. At equilibrium, the forces acting from one segment on another at the points of contact are equal. Stress in each segment can be expressed in terms of deformation of the segment and segment elasticity. Finally, we solve for the unknown deformations of the segments, to obtain the final expression for longitudinal stress:

$$\begin{aligned}
\sigma_L &= -\frac{A_{STI} + A_{TSV} + \left(\Delta L_{BC_L} + \Delta L_{BC_R}\right)}{\frac{(1-\alpha_{STI}\Delta T)}{E_{STI}}\sum_i L_{STI_i} + \frac{(1-\alpha_{Si}\Delta T)}{E_{Si}}\sum_j L_{Si_j} + \frac{(1-\alpha_{Cu}\Delta T)}{E_{Cu}}\sum_k D_{TSV_k}} \\
A_{STI} &= (\alpha_{Si} - \alpha_{STI})\Delta T \sum_i L_{STI_i} \\
A_{TSV} &= (\alpha_{Si} - \alpha_{Cu})\Delta T \sum_i D_{TSV_i}\left(|x_i| - |y_i|\right)/\sqrt{x_i^2 + y_i^2}
\end{aligned} \tag{2}$$

Here L_{STI_i} is the length of the *i*-th STI segment, L_{Si_j} is the length of the *j*-th silicon segment, D_{TSV_k} is the diameter of the *k*-th TSV, α_{Si}, α_{Cu} and α_{STI} are the

coefficients of thermal expansion for silicon, Copper, and STI, respectively. E denotes the elasticity constants of different materials, and ΔL_{BC_L} and ΔL_{BC_R} are the boundary conditions at the left and right window edges representing stress-induced edge displacements, respectively. Stress generated by TSV is in the radial direction, and this needs to be transformed into longitudinal and lateral components. Terms x_i and y_i used for this transformation denote the relative co-ordinates of the center of the device while considering the center of the *i*-th TSV as the origin. Replacing longitudinal measurements by lateral (transverse) measurements and left and right boundary conditions by top and bottom edges, we obtain the following expression for transverse stress:

$$\begin{aligned}
\sigma_T &= -\frac{A_{STI} + A_{TSV} + \left(\Delta L_{BC_T} + \Delta L_{BC_T}\right)}{\frac{(1-\alpha_{STI}\Delta T)}{E_{STI}}\sum_i W_{STI_i} + \frac{(1-\alpha_{Si}\Delta T)}{E_{Si}}\sum_j W_{Si_j} + \frac{(1-\alpha_{Cu}\Delta T)}{E_{Cu}}\sum_k D_{TSV_k}} \\
A_{STI} &= (\alpha_{Si} - \alpha_{STI})\Delta T \sum_i W_{STI_i} \\
A_{TSV} &= (\alpha_{Si} - \alpha_{Cu})\Delta T \sum_i D_{TSV_i}(|y_i| - |x_i|)/\sqrt{x_i^2 + y_i^2}
\end{aligned} \tag{3}$$

These models correctly predict the stress due to irregular active area shape, neighboring devices, varying STI width, and other layout variations. The proposed models capture stress due to thermal mismatch. In case of additional sources of stress (e.g., embedded Silicon Germanium, nitride liners, etc.), total stress can be calculated by superposition. The window edge displacement terms (ΔL_{BC}) in (2) and (3) should generally be equal to zero according to the assumption of symmetric boundary conditions. However, in some specific cases when the effect of global load, such as packaging, chip mounting or 3D integration, on the variation of transistor-to-transistor characteristics is of interest, these terms should come from the global finite element based

simulation. We use piezoresistive coefficients to convert from stress to mobility [88]. The mobility multiplier for a given segment is expressed as:

$$\mu_{mult} = 1 + \frac{\Delta\mu}{\mu} = 1 + \pi_L \sigma_L + \pi_T \sigma_T \tag{4}$$

In case of irregular active area shape or neighboring features, a gate can be partitioned into segments of equal stress such that stress critical layout parameters are constant for each segment (as discussed in Chapter 5). The final mobility multiplier is then obtained by calculating a width based weighted average of these multipliers to obtain an overall device mobility multiplier.

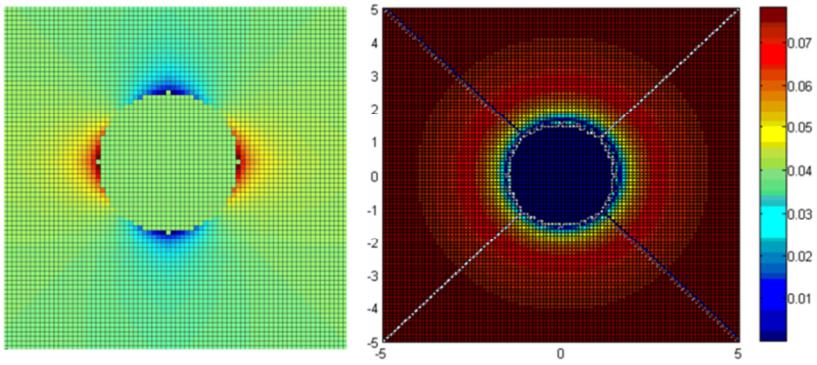


Figure 10.2 Longitudinal stress as predicted by the model (left) and percentage error compared to finite element based simulation (right).

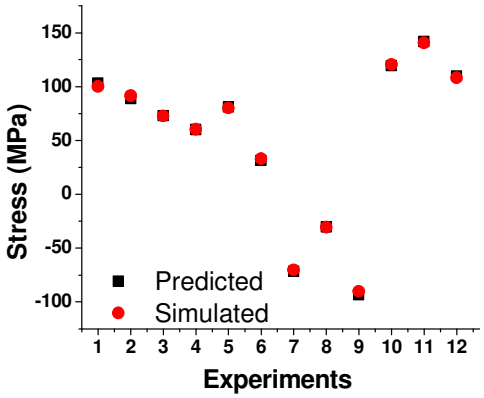


Figure 10.3 Simulated and modeled longitudinal stress for different layout configurations.

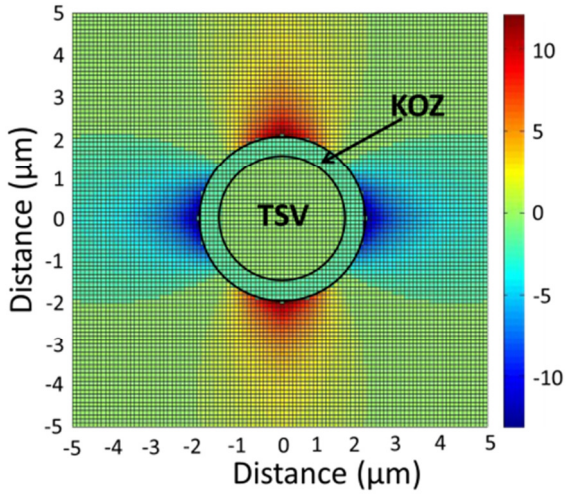


Figure 10.4 Hole mobility variation for an isolated TSV.

10.1.2 Model Verification

To verify the accuracy of proposed model, we compare the model based prediction with finite element based analysis. Figure 10.2 shows the longitudinal stress map for an isolated TSV as predicted using the proposed models, along with the percentage error in the model prediction as compared to finite element based simulation. The error in predicted stress is less than 1% for the entire region. Next, we test the accuracy of the model for more complex combinations of layout parameters. We vary the number and position of TSVs, shape and width of STI and neighboring transistors, and the active area length of device, to generate a set of experiments. As shown in Figure 10.3, the proposed models exhibit a very good fit to the simulation results. Finally, Figure 10.4 shows the hole mobility variation for an isolated TSV. As shown in the figure, significant hole mobility variation is observed for a single TSV, and hole mobility can vary by as much as ~20% depending on the position of the device relative to the TSV. This variation will increase as the number of TSVs increases, due to superposition of stress as expressed in Equations (2) and (3). Impact of TSV stress on device mobility

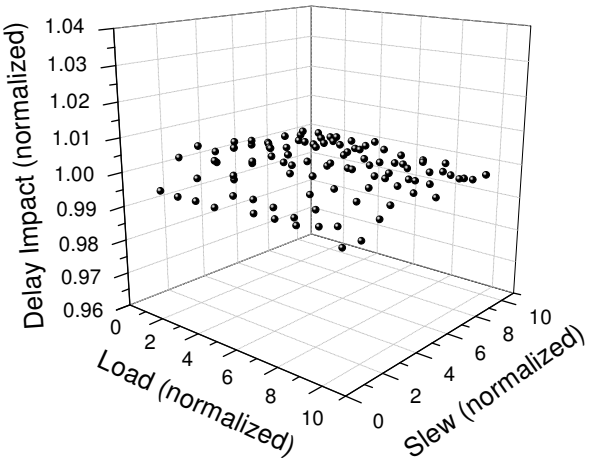


Figure 10.5 Impact of 10% mobility change on fall delay across different input slew and load capacitance values.

is found to be significant, supporting the need for a TSV-aware timing analysis framework.

10.2 TSV-aware Timing Analysis

There are two possible approaches to TSV-aware timing analysis:

- Characterize standard cell libraries at different device mobility values. For a given device, calculate the TSV stress based mobility change, and interpolate between values from library files characterized at fixed values of electron and hole mobilities.
- Model the dependence of gate delay on mobility change. Characterize standard cell library once, and superimpose these gate delay model during timing analysis based on the TSV stress based mobility change.

Figure 10.5 shows the impact of a 10% increase in electron mobility on the output falling delay of an industrial 45nm inverter, for different output capacitance load and input slew combinations. Delay impact is expressed as the ratio between nominal and higher mobility case, for a given load and slew value. All delay values are normalized to

the delay impact for the nominal case with FO4 load and a nominal input slew. As shown in the figure, the impact of mobility on delay is very uniform across different load and slew values (maximum deviation from the nominal case is ~0.6%). Hence, we can model dependence of gate delay on mobility change for the nominal case, and the same function can be used to predict delay impact for any given slew and load combination with reasonable accuracy. This approach has a lower characterization cost as compared to the first approach where multiple standard cell libraries have to be characterized. Calibrated models are used to generate stress maps and predict the impact on device mobility based on TSV configuration and position of device in the layout. This mobility multiplier is used to evaluate the final impact of TSV-induced stress on rise and fall delays. Finally, we perform timing analysis on the circuit to obtain circuit level delay values.

10.3 Experimental Results

To demonstrate the importance of TSV-aware analysis, we applied the flow described in the previous section to a variety of circuits using industrial commercial 45nm technology. We present gate level and circuit level results, based on TSVs that are 3µm in diameter with a nominal KOZ of 0.5µm, which represents the current state of the art [119,132,133].

10.3.1 Gate-level Analysis

Figure 10.6 shows the variation in gate delay of an inverter based on its position, for an isolated TSV. As shown, two identical inverters can vary by as much as 12% in rise delay, and 4% in fall delay based on their position relative to the TSV. Table 10.1 summarizes the maximum and minimum values of normalized rise and fall delay for a variety of gates. All delay values are normalized to the nominal FO4 gate delay.

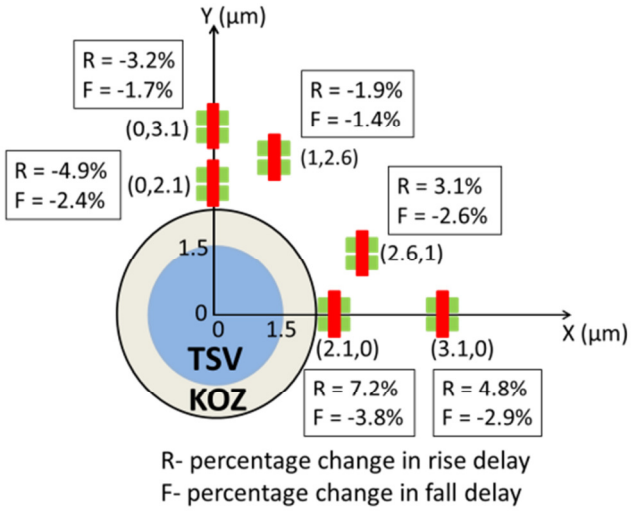


Figure 10.6 Inverter gate delay variation based on its position.

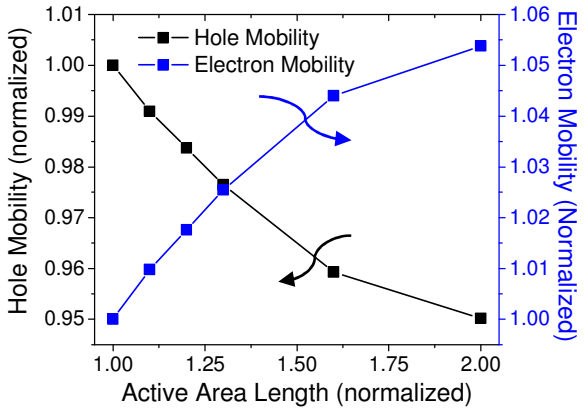


Figure 10.7 Electron and hole mobility variation with active area length.

Table 10.1 Impact on cell delay based on layout position.

DPL Technique	Rise max (normalized)	Rise min (normalized)	Fall Max (normalized)	Fall Min (normalized)
Inverter	1.07	0.95	1.00	0.96
NAND2	1.06	0.95	1.00	0.95
NOR2	1.07	0.96	1.00	0.96
NAND3	1.06	0.95	1.00	0.94
NOR3	1.06	0.96	1.00	0.95

Significant impact on gate delay is observed in all cases, and the impact will increase when more than one TSV is considered due to superposition of stress. Figure

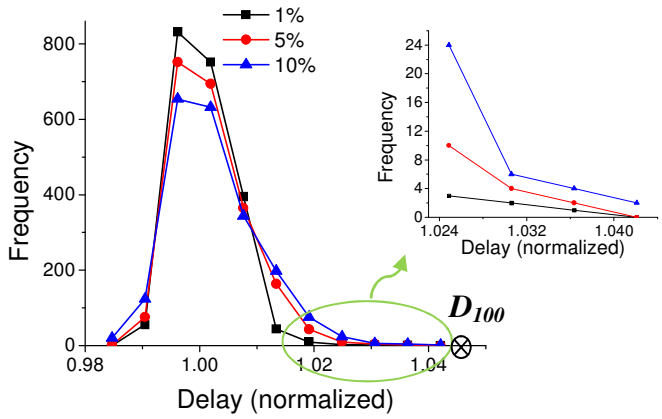


Figure 10.8 Delay distribution for different TSV densities (16x16 multiplier).

10.7 shows the impact of changing the active area length on electron and hole mobility for an inverter at a distance of 0.5µm from an isolated TSV. Increasing active area length enhances electron mobility while degrading hole mobility. Similar results can be obtained by filler insertion next to the transistors. These gate-level results establish the potential for 1) TSV-aware optimization that can be achieved through TSV-aware placement where critical devices can be placed in TSV-stress based mobility enhancement zones to enhance performance, and 2) TSV-aware active area change/filler insertion where TSV stress maps are used to change the active area length and active fill around timing critical devices to improve overall circuit delay.

10.3.2 Circuit-level Analysis

For the circuit-level timing analysis, we first create a fine grained TSV grid, and then for a given TSV density, TSVs are randomly assigned to the grid points. For each TSV density, we run 2000 simulations to capture potential cases where TSVs are adjacent to critical paths, thereby significantly impacting the circuit delay. Figure 10.8 shows the delay distribution (normalized to nominal delay) for 2000 runs at TSV densities of 1%, 5%, and 10%, along with the delay for 100% TSV density (D100) for a 16x16 multiplier

benchmark circuit. Maximum impact on delay was found to be 4.2% relative to nominal delay. This impact manifests as error if timing analysis is not able to model the impact of TSV stress on channel mobility. The figure shows that certain configurations of TSV have significant impact on circuit delay for even low densities. Even though the distribution spread increases as we increase TSV density, there are outliers with high impact on delay for lower densities as well. For example, even in the case of TSV density as low as 1%, some configurations lead to a change in delay close to the 100% density case (which is the maximum possible impact). Table 10.2 summarizes the impact of TSV-induced stress on delay for different benchmark circuits. All the delay values are normalized to the nominal case, which is not TSV-aware. Significant impact on cell delay is observed for all TSV densities, and TSV-aware cell delay varies from 4.2%-6.9%. Based on these results we conclude that TSV-aware timing analysis is required to enable accurate analysis of 3D circuits.

Finally, we study the impact of KOZ on TSV stress induced delay variation. One possible way to decrease circuit delay sensitivity to TSV stress is to increase the KOZ. This reduces the stress generated in the device channel at the cost of increased chip area. We first focus on the simple case of an inverter and an isolated TSV shown in Figure 10.6. Figure 10.9 shows the maximum rise delay and KOZ area (normalized to nominal KOZ area) as a function of the KOZ length. Upon increasing the KOZ to 3µm (equal to the TSV diameter), there remains a 2% impact of TSV stress on inverter delay (while incurring 9.6X increase in KOZ area). As shown in the figure, significant area increase is observed for intermediate delay values as well. This implies that, for layouts with high TSV density, increasing KOZ might not be a very effective way to minimize the impact

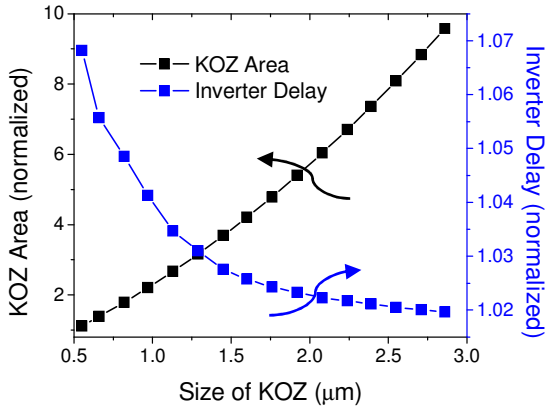


Figure 10.9 Inverter delay as a function of KOZ dimension.

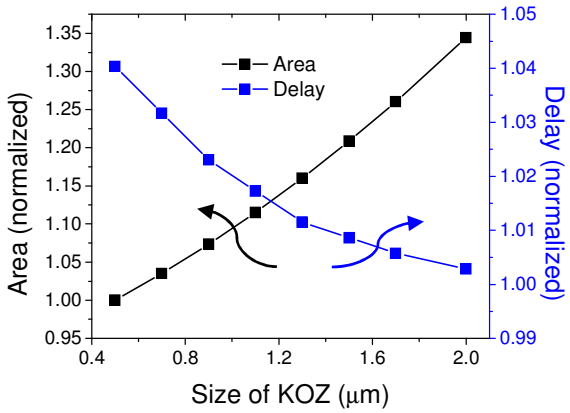


Figure 10.10 Delay and area as a function of KOZ size for 16x16 multiplier.

Table 10.2 Impact of TSV stress on circuit delay.

Benchmark	# of Gates	% Delay Impact (normalized)
8-bit ALU	1163	5.4%
16x16 multiplier	3834	4.2%
32-bit adder	2152	6.3%
Viterbi Decoder 1	14503	6.8%
Viterbi Decoder 2	36529	6.9%

of TSV stress on circuit delay. Figure 10.10 shows how the area and delay vary with increase in the KOZ (from the nominal value of 0.5μm) for the 16x16 multiplier, assuming a TSV density of 10%. As expected, while the area penalty increases to 12%, there is still a finite impact (~2%) of TSV stress on circuit timing. Thus, there is a need

for TSV-aware optimization framework to best exploit TSV stress, instead of less effective solutions such as larger KOZ.

10.4 Summary

In this chapter, comprehensive closed-form models for TSV-induced mechanical stress and its impact on device mobility are presented, enabling TSV-aware timing analysis. TSV-aware timing analysis shows significant impact of TSV stress on both gate and circuit-level delay (up to 6.9% impact on circuit delay). Based on the analysis results, it is concluded that there is a need for TSV-aware optimization to exploit TSV stress for balancing circuit performance and area.

Chapter 11
Conclusion and Future Work

Semiconductor device scaling trends indicate an overall increase in variation with technology scaling. This can be attributed to increased difficulty in controlling device fabrication process, increase in atomic scale randomness (e.g. doping profile), and emergence of new variation causing mechanisms. Over the course of the last decade, new process steps have been introduced to aid device manufacturing (e.g. new lithography techniques) and to help meet scaling targets (e.g. inducing mechanical stress in device channel to enhance mobility). These process techniques, while critical to continued device scaling, also act as new sources of systematic variation. As a result, DFM has emerged as an important semiconductor research field, to improve device manufacturability in latest technology nodes. This dissertation presents models, design techniques, analysis and optimization frameworks to enable optimal design of advanced semiconductor devices. Rest of the chapter summarizes our contributions to the evolving field of DFM, and concludes with a discussion of future work.

11.1 Conclusion – Summary of Our Contributions

This dissertation focuses on new sources of variation for advanced technology nodes. Chapter 2 presents soft-edge flip-flops as an effective method to address process variation through time borrowing and averaging across stages. This time borrowing is enabled by delaying the clock edge of the master latch in a flip-flop, so as to create a

window of transparency instead of a hard boundary for capturing the data. We develop a library of soft-edge flip-flops with varying amounts of softness. We then propose a statistically aware flip-flop assignment algorithm that maximizes the gain in timing yield while minimizing the incurred power overhead. Experimental results on a wide range of benchmark circuits show that the proposed approach improves the mean delay by 1.9-22.3% while simultaneously reducing the standard deviation of delay by 1.9-24.1% while increasing power by a small amount (0.3-2.8%).

The next four chapters focus on modeling, analysis, and optimization techniques for mechanical stress based enhancement of device mobility. In Chapter 3, we proposed stress as a means to achieve optimal power-performance trade-off by combining stress based, performance-enhanced standard cell assignment with dual-V_{th} assignment. We studied how stress-induced performance enhancements are affected by layout properties and improved standard cell layouts so that performance gains are maximized. We then developed a circuit-level, block-based, stress-enhanced optimization algorithm including all layout-dependent sources of mechanical stress. By combining the two performance enhancement techniques (stress-based and dual-V_{th}) for a set of benchmark circuits, we found that our stress-aware optimization, decreased leakage by ~24% on average, for iso-delay, when compared to dual-V_{th} assignment. Similarly, for iso-leakage, our optimization algorithm reduced delay on average by 5%. In both cases, the proposed method only incurred a small area penalty (< 0.5%). In Chapter 5, we proposed a new library design methodology, called STEEL, which shared the V_{DD} and V_{SS} (power and ground) source/drain connections across standard cell boundaries and, consequently, increased mobility and performance (due to the strong active area dependency of

mechanical stress). Overall, this standard cell performance improvement led to circuit delay reductions of 11% while only increasing leakage by 35% – a 2.5X reduction from equivalent DVT implementations.

Chapter 5 presented compact closed-form models that capture the layout dependence of mechanical stress induced in the device channel while considering all relevant sources of stress (STI, tensile/compressive nitride liners, and embedded SiGe). The models were calibrated using ring oscillator frequency data obtained from an experimental test chip to verify their accuracy. Results indicated that the models accurately capture the layout dependence of stress and carrier mobility for a variety of layout permutations and the root mean square error in the predicted ring oscillator frequency was less than 1% for the different layout experiments. These models can help drive layout optimization and timing/power analysis without the use of technology computer-aided design (TCAD) tools, which are slow and very limited in capacity. Chapter 6 we presented a novel method to effectively model non rectangular gates with non-uniform carrier mobility. First, we proposed a slicing and summing based approach to calculate effective carrier mobility for a device. We then developed a methodology for simultaneous extraction of effective gate length (EGL), and effective carrier mobility (ECM), to enable accurate prediction of both device drive current and leakage. This method was shown to more accurate than previously proposed approaches which result in in errors of up to 4.1% and 38.2% in device drive current and leakage, respectively.

Chapter 7 proposed a new local anneal temperature variation aware analysis framework which incorporated the effect of RTA induced temperature variation into timing and leakage analysis. We solved for chip level anneal temperature distribution,

and employed TCAD based device level models for drive current and leakage dependence on anneal temperature variation, to capture the variation in device performance and leakage based on its position in the layout. Experimental results based on a 45nm experimental test chip showed anneal temperature variation of up to ~10.5^{o}C, which can result in ~6.8% variation in device performance and ~2.45X variation in device leakage across the chip.

Chapters 8 and 9 focused on SRAM design for different DPL techniques. Chapter 8 analyzed the impact of DPL on functional failures in SRAM bitcells, and proposed a DPL-aware SRAM sizing scheme to effectively mitigate yield losses. Experimental results based on 45nm industrial models and test chip measurements showed that DPL can significantly impact SRAM cell robustness. Using the proposed DPL-aware sizing scheme, the SRAM cell failure probability was reduced by up to 3.6X. Chapter 9 extends the framework, to also analyze layers other than polysilicon across different DPL options (positive and negative SADP, and PSDP). It presented a comprehensive analysis and optimization framework that compared the layerwise impact of different Double Patterning Lithography (DPL) choices on SRAM robustness, density, and printability. It then performed a sizing optimization while accounting for increased variability due to DPL for each layer. Experimental results based on 45nm industrial models showed that using the best DPL option for each layer, along with the sizing optimization presented, we can achieve single exposure robustness together with improved DPL printability at nearly no overhead (less than 0.2% increase in write energy).

Finally, Chapter 10 presented an analysis of the impact of TSV induced mechanical stress on device, gate, and circuit level delay. We proposed a new TSV-aware

timing analysis framework, which performed circuit level timing using proposed closed-form models of TSV stress and its impact on device mobility. The framework was used to study the impact of TSV stress on circuit delay, and TSV stress was shown to cause delay variations of up to 6.9%.

11.2 Future Work

There are numerous problems within the scope of DFM that need to be addressed at the current and future technology nodes. This section discusses future explorations related to the work discussed in this dissertation.

11.2.1 DFM-friendly Placement and Routing

A lot of the work discussed in this dissertation focuses on analysis and circuit/gate level optimization techniques, to help mitigate the negative impact of variation. While these techniques have been shown to be extremely effective, intelligent placement and routing can potentially provide significant improvements in some specific cases. A few examples are:

RTA induced variation: Chapter 7 analyzes filler insertion, film deposition, and gate length biasing as techniques to mitigate RTA induced variation. However, an intelligent RTA-aware placement can exploit local anneal temperature by placing timing critical devices in layout positions with high local anneal temperature (and hence higher drive current and lower delay). Such a technique can potentially improve circuit timing by exploiting RTA induced variation.

TSV Stress: As shown in Chapter 10, TSV stress can significantly impact circuit timing, and gate level delay impact has a strong dependence on the position of the gate relative to the TSV grid. Moreover, increasing the size of KOZ was shown to be very costly and less effective for layouts with a high TSV density. Similar to the case of local anneal

temperature variation, a TSV-aware placement scheme can potentially exploit TSV stress induced variation to improve circuit timing, while incurring lower penalty than a larger KOZ.

Double Patterning Lithography: A key issue in DPL from the layout point of view is decomposition of the layout for multiple exposure steps, such that two features are assigned to separate mask exposures if their spacing is less than the specified minimum spacing (single exposure resolution limit). Layout segments that cannot be decomposed in this manner, are called native conflicts, and need to be resolved by altering the layout [111, 112]. Post placement decomposition approaches do not achieve great results, and some native conflicts cannot be resolved. An intelligent decomposition friendly placement and routing can potentially improve the decomposability and manufacturability of the layout significantly by enabling better conflict resolution.

11.2.2 Exploring the Impact of New Lithography Techniques on Logic and Memory

As mentioned earlier, DPL is the only viable lithography solution for current technology nodes, due to technical hurdles delaying commercial implementation of new lithography techniques such extreme ultraviolet (EUV), immersion ArF (IArF) lithography, and e-beam lithography. A lot of research is focusing on enabling the implementation of these new lithography processes, and EUV, for example, is on the ITRS roadmap for technology nodes beyond 22nm [119]. However, there is a need to develop modeling and simulation framework to enable analysis and design for these new lithography techniques. Such a framework can help determine the design level impact of a lithography choice on both logic and memory. The design level impact will be an important metric essential to a better understanding of the tradeoffs involved when choosing one lithography option over another.

11.2.3 Designing Test Structures for Model Calibration

Several analysis and modeling frameworks have been proposed in this thesis to help capture variation due to new sources for advanced technology nodes. Silicon based data is required to verify and calibrate these models and improve their accuracy. However, it is not trivial to isolate many of the discussed effects, and research is needed to design test structures and measurement techniques which isolate the effect of these different sources of variation. For example, it is extremely difficult to decouple the impact of mechanical stress on device drive current from random threshold voltage variation. Similarly, isolating the impact of local anneal temperature variation on device leakage and drive current is a difficult challenge. Each of these problems needs a unique approach based on the underlying variation causing mechanism. One such example was discussed in Chapter 8, where measurement based analysis of DPL impact on SRAM required special processing of SRAM failure data (breaking up the failure data into even and odd rows), to distinguish the impact of DPL from random V_{th} variation. Such techniques will be critical to model verification and calibration in future technology nodes.

RELATED PUBLICATIONS

- V. Joshi, D. Sylvester, "Modeling TSV-Induced Mechanical Stress to Enable TSV-Aware Timing Analysis," *Submitted to IEEE Custom Integrated Circuits Conference (CICC)*, 2011.

- V. Joshi, K. Agarwal, D. Sylvester, "Design-Patterning Co-optimization of SRAM Robustness for Double Patterning Lithography," *Submitted to IEEE Custom Integrated Circuits Conference (CICC)*, 2011.

- V. Joshi, K. Agarwal, D. Blaauw, D. Sylvester, "Analysis and Optimization of SRAM Robustness for Double Patterning Lithography," *ACM/IEEE International Conference on Computer-Aided Design (ICCAD)*, pp.25-31, 2010.

- V. Joshi, M. Wieckowski, G. Chen, D. Blaauw, D. Sylvester, "Analyzing the Impact of Double Patterning Lithography on SRAM Variability in 45nm CMOS," *Proc. IEEE Custom Integrated Circuits Conference (CICC)*, 2010.

- V. Joshi, V. Sukharev, A. Torres, K. Agarwal, D. Sylvester, D. Blaauw, "Closed-Form Modeling of Layout-Dependent Mechanical Stress," *ACM/IEEE Design Automation Conf (DAC)*, pp. 673-678, 2010.

- V. Joshi, K. Agarwal, D. Sylvester, "Simultaneous Extraction of Effective Gate Length and Low-field Mobility in Non-uniform Devices," *International Symposium on Quality Electronic Design (ISQED)*, pp. 158-162, 2010.

- V. Joshi, K. Agarwal, D. Sylvester, D. Blaauw, "Analyzing Electrical Effects of RTA-driven Local Anneal Temperature Variation," *Asia and South Pacific Design Automation Conference (ASPDAC)*, pp. 739-744, 2010.

- V. Joshi, B. Cline, D. Sylvester, D. Blaauw, and K. Agarwal, "Mechanical Stress Aware Optimization for Leakage Power Reduction," *IEEE Trans. on CAD of Integrated Circuits and Systems*, pp. 722-736, 2010.

- B. Cline, V. Joshi, D. Sylvester, D. Blaauw, "Stress Enhanced Standard Cell Library Design," *ACM/IEEE International Conference on Computer-Aided Design (ICCAD)*, pp. 691-697, 2008.

- V. Joshi, B. Cline, D. Sylvester, D. Blaauw, K. Agarwal, "Leakage Power Reduction using Stress-Enhanced Layouts," *ACM/IEEE Design Automation Conf (DAC)*, pp. 912-917, 2008.

- V. Joshi, B. Cline, D. Sylvester, D. Blaauw, K. Agarwal, “Stress Aware Layout Optimization,” *International Symposium on Physical Design (ISPD)*, pp. 168-174, 2008.

- V. Joshi, D. Blaauw, D. Sylvester, “Soft-edge Flip-flops for Improved Timing Yield: design and optimization,” *ACM/IEEE International Conference on Computer-Aided Design (ICCAD)*, pp. 667-673, 2007.

BIBLIOGRAPHY

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

[2] D. Frank, "Parameter variations in sub 100 nm CMOS technology," *ISSCC—Advanced Solid State Circuits Forum: Managing Variability on Sub-100nm designs*, 2004.

[3] S. Borkar, T. Kamik, S. Narendra, J. Tschanz, A. Keshavarsi, and V. De,"Parameter variations and impact on circuits and microarchitecture," *Proc. Des. Autom. Conf.*, pp. 338–342, 2003.

[4] A. K. K. Wong, *Resolution Enhancement Techniques in Optical Lithography*, SPIE PRESS, 2001, p. 28.

[5] J. A. Croon et al., "Line edge roughness: Characterization, modeling, and impact on device behavior," *Proc. of IEDM*, pp. 307-310, 2002.

[6] N. Gunther, E. Hamadeh, D. Niemann, I. Pesic, M. Rahman, "Modeling intrinsic fluctuations in decananometer MOS devices due to gate line edge roughness (LER)," *Proc. of ISQED*, pp. 510-515, 2005.

[7] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos, "Modeling Within-Die Spatial Correlation Effects for Process-Design Co-Optimization", *Proc. Int. Symp. on Quality Electronic Design*, pp. 516 – 521, Mar. 2005.

[8] R. Singhal, A. Balijepalli, A. Subramaniam, F. Liu, S. Nassif, and Y. Cao, "Modeling and Analysis of Non-Rectangular Gate for Post-Lithography Circuit Simulation," *Proc. 44th Annual Conf. on Design Automation*, pp. 823 – 828, June 2007.

[9] M. Orshansky, L. Milor, and C. Hu, "Characterization of Spatial Intrafield Gate CD Variability, Its Impact on Circuit Performance, and Spatial Mask-Level Correction," *IEEE Trans. on Semiconductor Manufacturing*, vol. 17, no. 1, pp. 2 – 11, Feb. 2004.

[10] M. Palusinski, A. J. Strojwas, and W. Maly, "Regularity in Physical Design," *GSRC Workshop*, June 2001.

[11] A. J. Strojwas, "Process-Design Interaction Modeling Based Design for Manufacturability," *Tutorial, Proc. 40th Annu. Conf. on Design Automation*, June 2003.

[12] Sematech High Index Workshop, Oct. 2006, Kyoto.

[13] H. H. Solak et al., "Sub-50 nm period patterns with EUV interference lithography," *Microelectronic Engineering*, vol. 67-68 , Issue 1, pp. 56-62, 2003.

[14] R. H. French et al., "High Index Immersion Lithography With Second Generation Immersion Fluids To Enable Numerical Apertures of 1.55 For Cost Effective 32 nm Half Pitches," *Proc. SPIE Optical Microlithography XX*, 6520, 2007.

[15] G. Capetti et al.," Sub k1 = 0.25 Lithography with Double Patterning Technique for 45nm Technology Node Flash Memory Devices at 193nm," *Proc. SPIE Optical Microlithography, vol. 6520*, pp. 65202K-1 - 65202K-12.

[16] J. Finders, et. al, "Double Patterning Lithography: The Bridge Between Low k1 ArF and EUV," *Microlithography World*, Feb. 2008.

[17] M. Drapeau, et. al, "Double Patterning Design Split Implementation and Validation for the 32nm Node," *Proc. SPIE Design for Manufacturability through Design-Process Integration*, vol. 6521, 2007.

[18] W.-Y. Jung, et. al, "Patterning with amorphous carbon spacer for expanding the resolution limit of current lithography tool," *Proc. SPIE 6156*, 6156J1, 2006.

[19] C.-M. Lim, et. al, "Positive and negative tone double patterning lithography for 50-nm flash memory," *Proc. SPIE 6154*, 615410, 2006.

[20] M. Maenhoudt, et.al, "Double Patterning Scheme for Sub-0.25 k1 Single Damascene Structures at NA=0.75, λ=193nm," *Proc. SPIE Conference on Optical Microlithography*, pp. 1508-1518, 2005.

[21] K. Jeong et. al, " Timing analysis and optimization implications of bimodal CD distribution in double patterning lithography," *Proc. ASPDAC*, pp. 486-491, 2009.

[22] H. Tuinhout, F. Widdershoven, P. Stolk, J. Schmitz, B. Dirks, K. van der Tak, P. Bancken, and J. Politiek, "Impact of Ion Implantation Statistics on VT Fluctuations in MOSFETs: Comparison between Decaborane and Boron Channel Implants," *Symp. on VLSI Technology*, pp. 134 – 135, 2000.

[23] Y. Li, S. M. Yu, and H. M. Chen, "Process-variation- and Random-dopants-induced Threshold Voltage Fluctuations in Nanoscale CMOS and SOI Devices," *Microelectronic Engineering*, vol. 84, pp. 2117 – 2120, Sept. 2007.

[24] A. Asenov, "Random Dopant Induced Threshold Voltage Lowering and Fluctuations in Sub-0.1 mm MOSFETs: A 3-D 'Atomistic' Simulation Study," *IEEE Trans. on Electron Devices*, vol. 45, no. 12, pp. 2505 – 2513, Dec. 1998.

[25] C. Millar, D. Reid, G. Roy, S. Roy, and A. Asenov, "Accurate Statistical Description of Random Dopant-Induced Threshold Voltage Variability," *IEEE Electron Device Letters*, vol. 29, no. 8, pp. 946 – 948, Aug. 2008.

[26] P. A. Stolk, F. P. Widdershoven, and D. B. M. Klaassen, "Modeling Statistical Dopant Fluctuations in MOS Transistors," *IEEE Trans. on Electron Devices*, vol. 45, no. 9, pp. 1960 – 1971, Sept. 1998.

[27] I. Ahasan et al., "RTA-driven intra-die variations in stage delay, and parametric sensistivities for 65nm technology," VLSI *Symp. Tech. Digest*, pp. 170-171, 2006.

[28] S. Wolf, *Silicon Processing for the VLSI Era*, Lattice Press, 1995, p. 273.

[29] A. Chandrakasan, W. J. Bowhill, and F. Fox, *Design of High-Performance Microprocessor Circuits*, IEEE press, 2001, p. 49.

[30] A. Chandrakasan, et al., "Design considerations and tools for low-voltage digital system design," *Proc. 33rd. Design Automation Conference*, pp. 113-118, 1996.

[31] *Front End Processes – International Technology Roadmap for* Semiconductors (2007 Edition)[Online].Available:http://www.itrs.net/Links/2007ITRS/2007_Chapters/2007_FEP.pdf

[32] A. Asenov, S. Kaya, and J. H. Davies, "Intrinsic Threshold Voltage Fluctuations in Decanano MOSFETs Due to Local Oxide Thickness Variations," *IEEE Trans. on Electron Devices*, vol. 49, no. 1, Jan. 2002.

[33] J. H. Stathis, "Physical and predictive models of ultrathin oxide reliability in CMOS devices and circuits," *IEEE Trans. Device and Mater. Rel.*, pp. 43–59,March 2001.

[34] E. Y. Wu et al., "CMOS scaling beyond the 100-nm node with silicon-dioxide-based gate dielectrics," *IBM J. Res. Dev.*, pp. 287–298, 2002.

[35] H. Wang et al., "Impact of random soft oxide breakdown on SRAM energy/delay drift," *IEEE Trans. Device and Mater.Rel.*, pp. 581–591, December 2007.

[36] K. Chopra et al., "A statistical approach for full-chipgate-oxide reliability analysis," *Proc. ICCAD*, pp. 698–705, November 2008.

[37] V. Mehrotra, S. Nassif, D. Boning, and J. Chung, "Modeling the Effects of Manufacturing Variation on High-Speed Microprocessor Interconnect Performance," *IEEE Int'l Electron Devices Meeting Technical Digest*, pp. 767 – 770, Dec. 1998.

[38] Y. Chen, A. B. Kahng, G. Robins, and A. Zelikovsky, "Area Fill Synthesis for Uniform Layout Density," *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 21, no. 10, pp. 1132 – 1147, Oct. 2002.

[39] Y. Chen, A. B. Kahng, G. Robins, A. Zelikovsky, and Y. Zheng, "Compressible Area Fill Synthesis," *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 24, no. 8, pp. 1169 – 1187, Aug. 2005.

[40] V. Chan et al., "Strain for PMOS performance Improvement," Proc. CICC, pp. 667-674, 2005.

[41] F. Andrieu et al., "Experimental and Comparative Investigation of Low and High Field Transport in Substrate- and Process-Induced Strained Nanoscale MOSFETs," *Proc. VLSI Technol. Symp. Tech. Dig.*, pp. 176-177, 2005.

[42] K. Mistry et al., "Delaying Forever: Uniaxial Strained Silicon Transistors in a 90nm CMOS Technology," *Proc. VLSI Technol. Symp. Tech. Dig.*, pp. 50-51, 2005.

[43] Z. Luo et al., "Design of high performance PFETs with strained Si channel and laser anneal," *Proc. of IEDM*, pp. 489-492, 2005.

[44] R. A. Bianchi et al., "Accurate modeling of trench isolation induced mechanical stress effects on MOSFET electrical performance," *Proc. of IEDM*, pp. 117-120, 2002.

[45] H. S. Yang et al., "Dual stress liner for high performance sub- 45nm gate length SOI CMOS manufacturing," *Proc. IEDM*, pp. 1075-1077, 2004.

[46] K. Ota et al., "Novel locally strained channel technique for high performance 55nm CMOS," *Proc. IEDM*, pp. 27-30, 2002.

[47] V. Moroz et al., "The Impact of Layout on Stress-Enhanced Transistor Performance," *Proc. of SISPAD*, pp. 143-146, 2005.

[48] S. Nassif, "Delay variability: sources, impacts and trends," *Proc. of ISSCC*, pp. 368-369, 2000.

[49] Y Taur et al., "CMOS scaling into nanometer regime," *Proc. of the IEEE*, no. 4, pp. 486-504, 1997.

[50] S.R. Nassif, A.J. Strojwas and S.W. Director, "A methodology for worst-case analysis of integrated circuits," *IEEE Trans. Computer-Aided Design*, vol. CAD-5, pp.104-113, Jan. 1986.

[51] C. Visweswariah, "Death, taxes and failing chips," *Proc of DAC*, pp. 343-347, June 2003.

[52] Hongliang Chang and Sachin Sapatnekar, "Statistical timing analysis considering spatial correlations using a single pert-like traversal," *Proc. of ICCAD*, pp. 621-625, 2003.

[53] E. Jacobs and M. Berkelaar, "Gate sizing using a statistical delay model," Proc. of DAC, pp. 283-290, 2000.

[54] P Seung et al., "Novel sizing algorithm for yield improvement under process variation in nanometer technology," *Proc. of DAC*, pp. 454-459, 2004.

[55] M. Mani and M. Orshansky, "A new statistical optimization algorithm for gate sizing," *Proc. of ICCD*, pp. 272-277,2004.

[56] Jaskirat Singh, Vidyasagar Nookala, Zhi-Quan Luo, and Sachin Sapatnekar, "Robust gate sizing by geometric programming," *Proc. of DAC* , pp. 315–320, 2005.

[57] X. Liu, M.C. Papaefthymiou, and E.G. Friedman, " Maximizing performance by retiming and clock skew scheduling ," *Proc. of DAC*, pp. 231-236, 1999.

[58] V. Nawale and T.W. Chen, "Optimal useful skew scheduling in the presence of variations using robust ILP formulation," *Proc. of ICCAD*, pp 27-32, 2006.

[59] Aaron P. Hurst and Robert K. Brayton, "Latch based design under process variation," *IWLS* 2006.

[60] A. Srivastava, D. Sylvester and D. Blaauw, "Statistical optimization of leakage power considering process variations using dual Vth and sizing," *Proc. of DAC*, pp. 773-778,2004.

[61] N. Magen, A. Kolodny, U. Weiser, N. Shamir, "Interconnect power dissipation in a microprocessor," *SLIP* 2004.

[62] S. Sirichotiyakul et al., "Duet: An Accurate Leakage Estimation and Optimization Tool for Dual-Vt Circuits," *IEEE Trans. on VLSI Systems*, Vol. 10, No. 2, pp. 79-90, April 2002.

[63] L. Wei et al., "Design and optimization of low voltage high performance dual threshold CMOS circuits," *Proc. 35th Design Automation Conference*, pp. 489-494, June 1998.

[64] D. Sylvester and A. Srivastava, "Computer-Aided Design for Low-Power Robust Computing in Nanoscale CMOS," *Proc. of the IEEE*, Vol. 95, pp. 507-529, March 2007.

[65] R. A. Bianchi, G. Bouche, and O. Roux-dit-Buisson, "Accurate modeling of trench isolation induced mechanical stress effects on MOSFET electrical performance," *Proc. of IEDM*, pp. 117-120, 2002.

[66] A. Kahng, P. Sharma, and R.O. Topaloglu, "Chip Optimization Through STI-Stress-Aware Placement Perturbations and Fill Insertion," *IEEE Trans. on CAD*, Vol. 72, pp. 1241-1252, July 2008.

[67] K. Su et al., "A Scaleable Model for STI Mechanical Stress Effect on Layout Dependence of MOS Electrical Characteristics," *in Proc. of CICC*, pp. 245-248, Sept. 2003.

[68] K. Yamada, et al., "Layout-Aware Compact Model of MOSFET Characteristics Variations Induced by STI Stress," *in IEICE Trans. on Elect*, Vol. E91-C, No. 7, pp. 1142-1150, July 2008.

[69] V. Moroz et al., "The Impact of Layout on Stress-Enhanced Transistor Performance," *in Proc. SISPAD*, pp. 143-146, Sept. 2005.

[70] Y.M. Sheu et al., "Modeling Mechanical Stress Effect on Dopant Diffusion in Scaled MOSFETs," *in IEEE Trans. on Electron Devices*, Vol. 52, pp. 30-38, Jan. 2005.

[71] M. V. Dunga et al., "Modeling Advanced FET Technology in a Compact Model," *IEEE Trans. on Elect. Dev.*, Vol. 53, pp. 1971-1978, Sept. 2006.

[72] G. Eneman et al., "Layout Impact on the Performance of a Locally Strained PMOSFET," *in Proc. of Symp. on VLSI Technology*, pp. 22-23, June 2005.

[73] L. T. Pang et al., "Measurement and Analysis of Variability in 45 nm Strained-Si CMOS Technology," *in IEEE Journal of Solid-State Circuits*, Vol. 44, pp. 2233-2243, Aug. 2009.

[74] A. Chakraborty, S. Shi, and D. Pan, "Layout Level Timing Optimization by Leveraging Active Area Dependent Mobility of Strained-Silicon Devices," *in Proc. of DATE*, pp. 849-855, March 2008.

[75] W. H. Lee et al., "High performance 65 nm SOI technology with enhanced transistor strain and advanced-low-K BEOL," *in Proc. IEDM*, Dec. 2005.

[76] G. Scott et al., "NMOS Drive Current Reduction Caused by Transistor Layout and Trench Isolation Induced Stress," *in Proc. of IEDM*, pp. 827-830, 1999.

[77] Z. Luo et al., "Design of high performance PFETs with strained si channel and laser anneal," *in Proc. of IEDM*, pp. 489-492, Dec. 2005.

[78] H. S. Yang et al., " Dual stress liner for high performance sub-45nm gate length SOI CMOS manufacturing," *in Proc. of IEDM*, pp. 1075-1077, Dec. 2004.

[79] K. Ota et al., "Novel locally strained channel technique for high performance 55nm CMOS," *in Proc. of IEDM*, pp. 27-30, 2002.

[80] A. Eiho et al., "Management of Power and Performance with Stress Memorization Technique for 45nm CMOS," *in Proc. IEEE Symposium on VLSI Technology*, pp. 218-219, June 2007.

[81] Manual, *DaVinci 3D TCAD*, Version 2005.10.

[82] Manual, *Synopsys TSUPREM4*, Version 2007.03.

[83] T. B. Hook et al, "Lateral Ion Implant Straggle and Mask Proximity Effect," *in IEEE Trans. on Electron Devices*, Vol.50, pp.1946-1951, Sept. 2003.

[84] Manual, *BSIM4 Spice Model*, Version 4.6.1, pp. 115-116.

[85] A. Dharchoudhury et al., "Transistor-level sizing and timing verification of domino circuits in the powerPC™ microprocessor," *in Proc. ICCD*, pp. 143-148, Oct. 1997.

[86] V. Zyuban et al., "Unified Methodology for Resolving Power-Performance Tradeoffs at the Microarchitectural and Circuit Levels," *in Proc. ISLPED*, pp. 166-171, Aug. 2002.

[87] S. Burns et al., "Comparative Analysis of Conventional and Statistical Design Techniques," *in Proc. 44th Design Automation Conference*, pp 238-243, June 2007.

[88] Y. Kanda, "A Graphical Representation of the Piezoresistance Coefficients in Silicon," *IEEE Transactions on Electron Devices*, Ed. 29, No. 1, 1982, pp. 64-70.

[89] Balasinski A. "A methodology to analyze circuit impact of process-related MOSFET geometry," *Proc. of SPIE*, vol.5379, no.1, pp.85-92, 2004.

[90] W.J.Poppe, L. Capodieci, J.Wu, and A. Neureuther, "From Poly Line to Transistor: Building BSIM Models for Non-Rectangular Transistors," *Proc. of SPIE*, Vol. 6156, 2006.

[91] Jen-Yi Wuu, Fedor G. Pikus, Malgorzata Marek-Sadowska, "Fast and simple modeling of non-rectangular transistors," *Photomask Technology 2008. Edited by Kawahira, Hiroichi; Zurbrick, Larry S. Proceedings of the SPIE*, Volume 7122, pp. 71223S-71223S-10, 2008.

[92] A Sreedhar, S. Kundu, "On modeling impact of sub-wavelength lithography on transistors," *Proc. of ICCD*, pp. 84-90, 2007.

[93] P.Gupta, A.Kahng, Y.Kim, S.Shah, and D.Sylvester, "Modeling of non-uniform device geometries for postlithography circuit analysis," *Proc. of SPIE*,Vol. 6156, 2006.

[94] R.Singhal, A.Balijepalli, A.Subramaniam, F.Liu, S.Nassif, and Y.Cao, "Modeling and Analysis of Non-Rectangular Gate for Post-Lithography Circuit Simulation," *Proc. of DAC*, 2007.

[95] P. J. Timmans et al., "Rapid Thermal Processing," in *Handbook of Semiconductoer Manufacturing Technology*, Y. Nishi and R. Doering (eds.), Marcel Dekker, Inc., New York (2000), pp. 201-286.

[96] R. B. MacKnight et al., "RTP application and technology options for the sub-45 nm nodes," *in Proc. 12th IEEE International Conference on Advanced Thermal Processing of Semiconductors*, pp. 3-36, 2004.

[97] A. Agarwal, "Ultra-shallow junction formation using conventional ion implantation and rapid thermal annealing," *in Proc. Conference on Ion Implantation Technology*, pp. 293-299, 2000.

[98] P. J. Timmans et al., "Challenges for ultra-shallow junction formation technologies beyond the 90 nm node," *in Proc. 11th IEEE International Conference on Advanced Thermal Processing of Semiconductors*, pp. 17-33, 2003.

[99] Woo Sik Yoo et al., "Comparative study on implant anneal using single wafer furnace and lamp-based rapid thermal processor," *in Proc. 9th IEEE International Conference on Advanced Thermal Processing of Semiconductors*, pp. 240-245, 2001.

[100] T. Feudel et al., "Junction scaling for next generation microprocessor technologies," *in Proc. SEMI Technology Sessions*, SEMICON Europa, 2005.

[101] Y. H. Zhou et al., "A Monte Carlo Model for Predicting the Effective Emissivity of the Silicon Wafer in Rapid Thermal Processing Furnaces," *in International J. Heat Mass Transfer*, vol. 45, pp. 1945-1949, 2002.

[102] S. K. Springer et al., "Modeling of Variation in Submicrometer CMOS ULSI Technologies," *in IEEE Transactions on Electron Devices*, vol. 53, no. 9, pp. 2168-2178, 2006.

[103] I. Ahasan et al., "RTA-driven intra-die variations in stage delay, and parametric sensistivities for 65nm technology," *in VLSI Symp. Tech. Digest*, pp. 170-171, 2006.

[104] M. Rabus et al., "Rapid thermal processing of silicon wafers with emissivity patterns," *in Journal of Electronic Materials*, vol. 35, no. 5, pp. 877-891, 2006.

[105] B. J. Lee et al., "Rad-Pro: effective software for modeling radiative properties in rapid thermal processing," *in Proc. 13th IEEE International Conference on Advanced Thermal Processing of Semiconductors*, pp. 275-281, 2005.

[106] Manual, *Mentor Graphics CALIBRE*, Version 2008.03.

[107] J. J. More et al., "Computing a trust region step," *SIAM Journal on Scientific and Statistical Computing*, vol. 3, pp. 553-572, 1983.

[108] R. H. Byrd et al., "Approximate solution of the trust region problem by minimization over two-dimensional subspaces," *Mathematical Programming*, vol. 40, pp. 247-263, 1988.

[109] Y. Wei, J. Hu, F. Liu, S. Sapatnekar, "Physical Design Techniques for Optimizing RTA-induced Variations," *ACM/IEEE Asia/South Pacific Design Automation Conference (ASPDAC)*, pp. 745 - 750, 2010.

[110] P. Gupta, A. Kahng, P. Sharma, D. Sylvester, "Selective gate-length biasing for cost-effective runtime leakage control," *ACM IEEE Design Automation Conference (DAC)*, pp. 327 - 330, 2004.

[111] S. Mukhopadhayay et. al, "Modeling of Failure Probability and Statistical Design of SRAM Array for Yield Enhancement in Nanoscaled CMOS," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Volume 24, Issue 12, pp. 1859-1880, December 2005.

[112] S. Verhaegen et. al, "Litho variations and their impact on the electrical yield of a 32nm node 6T SRAM cell," *Proc. SPIE*, vol. 6925, 69250R, 2008.

[113] C. Sarma et. al, "Double exposure double etch for dense SRAM: a designer's dream," *Proc. SPIE*, vol. 6924, pp. 692429-692429-9, 2008.

[114] M. Dusa et al., "Pitch Doubling Through Dual-Patterning Lithography: Challenges in Integration and Litho Budgets," *Proc. SPIE Conference on Optical Microlithography*, pp. 65200G-1 - 65200G-10, 2007.

[115] R. R. Wilcox, "Introduction to robust estimation and hypothesis testing," Elsevier academic press, 2005, p. 155.

[116] K.-J. R. Chen, et.al, "Resist Freezing Process for Double-Exposure Lithography", *Proc. SPIE Advances in Resist Materials and Processing Technology XXV*, Vol. 6923, 2008, pp. 69230G-1–69230G-10.

[117] Y. Ma et. al, "Decomposition Strategies for Self-Aligned Double Patterning," *Proc. SPIE*, vol. 7641, 76410T, 2010.

[118] S. Verhaegen et. al, "Litho variations and their impact on the electrical yield of a 32nm node 6T SRAM cell," *Proc. SPIE*, vol. 6925, 69250R, 2008.

[119] International Technology Roadmap for Semiconductors 2010 update.

[120] K. Jeong et. al, "Assessing Chip-Level Impact of Double Patterning Lithography," *Proc. ISQED 2010*, pp. 122-130.

[121] F. Arnaud et. al, "Low Cost 65nm CMOS Platform for Low Power & General Purpose Applications," *Proc. VLSI Technology Symposium*, pp. 10-11, 2004.

[122] C. Bencher et. al, "Gridded Design Rule Scaling: Taking the CPU toward the 16nm node," *Proc. SPIE*, vol. 7274, 72740G, 2009.

[123] T.-B. Chiou, et. al, "Full-Chip Pitch/Pattern Splitting for Lithography and Spacer Double Patterning Technologies", *Proc. SPIE Lithography Asia*, Vol. 7140, 2008, 71401Z-1–71401Z-12.

[124] R. Ghaida et. al, "Design-Overlay Interactions in Metal Double Patterning," *Proc. SPIE*, vol. 7275, 727514, 2009.

[125] M. Burkhardt et. al, "Overcoming the challenges of 22-nm node patterning through litho-design co-optimization," *Proc. SPIE*, vol. 7274, 727404, 2009.

[126] M. Koyanagi et al., "Three-Dimensional Integration Technology Using Through-Si Via Based on Reconfigured Wafer-to-Wafer Bonding," *Proc. CICC*, 2010.

[127] R. Tummala et al., "Trend from ICs to 3D ICs to 3D Systems," *Proc. CICC*, pp. 439 – 444, 2009.

[128] G. V. der Plas et al., "Design Issues and Considerations for Low-Cost 3D TSV IC Technology," *Proc. ISSCC*, 2010.

[129] C. Chiang and S. Sinha "The Road to 3D EDA Tool Readiness," *Proc. ASPDAC*, pp. 429-436, 2009.

[130] D. H. Kim et al, "A Study of Through-Silicon-Via Impact on the 3-D Stacked IC Layout," *Proc. ICCAD*, pp. 674-680, 2009.

[131] K. H. Lu et al., "Thermo-Mechanical Reliability of 3-D ICs containing Through Silicon Vias," *Proc. Electronic Components and Technology Conference*, pp. 630-634, 2010.

[132] J. S. Yang et al., "TSV Stress Aware Timing Analysis with Applications to 3D-IC Layout Optimization," *Proc. DAC*, pp. 803-806, 2010.

[133] S. Gupta et al, "Techniques for Producing 3D ICs with High-Density Interconnect," *Proc. 21st International VLSI Multilevel Interconnection Conference*, September 2004.