UNIVERSITY OF CALIFORNIA, SAN DIEGO

Variability Assessment and Mitigation in Advanced VLSI Manufacturing Through Design-Manufacturing Co-Optimization

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Electrical Engineering (Computer Engineering)

by

Kwangok Jeong

Committee in charge:

Professor Andrew B. Kahng, Chair
Professor Chung-Kuan Cheng
Professor Larry Larson
Professor Lawrence Saul
Professor Yuan Taur

2011

The dissertation of Kwangok Jeong is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Chair

University of California, San Diego

2011

DEDICATION

- *To my loving parents, without whose love and encouragement this thesis would not have started.*
- *To my wife, Sumi, without whose love and sacrifice this thesis would not have finished.*
- *To my son, Hoejin, for always being there smiling even though I could not always take care of him.*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank to my parents (my father Jaechoon Jeong and my mother Imsoon Lee), my wife Sumi Hong, her mother Heungyun Ryu, my son Hoejin Jeong, and my sister Yongok Jeong and her family for their patience of my absence and endless love throughout the years. Their continuous support and belief on me have been the most important contribution to my course.

I certainly feel privileged and grateful to work under my advisor Professor Andrew B. Kahng's guidance, and I thank him for all his invaluable support and advice. Especially, his enthusiasm always stimulates me, and his attitude on research and logical ways of thinking will remain in the rest of my life as great lessons.

I am thankful to my thesis committee members, Professor C. K. Cheng, Professor Larry Larson, Professor Lawrence Saul, and Professor Yuan Taur, for taking time out of their busy schedules to review and evaluate my research work. I am grateful for their encouragement and valuable feedback.

I certainly feel lucky to work with a great group of friends and collaborators in VLSI CAD Laboratory – former members Professor Puneet Gupta, Professor Bao Liu, Dr. Swamy Muddu, Dr. Kambiz Samadi, Dr. Chul-Hong Park, Dr. Puneet Sharma, Dr. Rasit O. Topaloglu, and Professor Hailong Yao, and current members Tuck-Boon Chan, Seokhyeong Kang, Jingwei Lu and Vaishnav Srinivas. We had wonderful projects and spent great times together, and I would like to especially thank all of the current members for proofreading this thesis. I wish continuous cooperation and correspondence with all of them.

I also miss the times with Mohit Gupta, Sung Kyu Han, Neelmani Kumar, Yen-Kuan Wu and Ruixin Yang. I wish them the best in their future.

I would also like to thank VLSI CAD Laboratory administrator Virginia McIlwain, ECE graduate program coordinator Shana Slebioda, and payroll managers Yuka Nakanishi and M'Lissa Michelson for their support and cooperation which many times went above and beyond their job responsibilities.

Last, but not least, I would like to express my deepest gratitude to Dr. Jeong-Taek Kong and Dr. Kyu-Myung Choi, and to all members of the Design Technology Team (former CAE Team) at Samsung Electronics, for their support and encouragement.

The material in this thesis is based on the following publications.

- Chapter 2 is based on the following publications:
  - **Kwangok Jeong**, Andrew B. Kahng and Kambiz Samadi, "Impacts of Guardband Reduction on Design Process Outcomes: A Quantitative Approach", *IEEE Transactions on Semiconductor Manufacturing* 22(4) (2009), pp. 552–565.
  - **Kwangok Jeong**, Andrew B. Kahng and Kambiz Samadi, "Quantified Impacts of Guardband Reduction on Design Process Outcomes", *Proc. International Symposium on Quality Electronic Design*, 2008, pp. 790–897.
- Chapter 3 is based on the following publication:
  - **Kwangok Jeong** and Andrew B. Kahng, "Variation Mapping From Timing Path Delay Measurements Using Compressed Sensing", *Proc. ACM International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems*, 2011, pp. 1–6.
- Chapter 4 is based on the following publications:
  - **Kwangok Jeong**, Andrew B. Kahng and Rasit O. Topaloglu, "Assessing Chip-Level Impact of Double Patterning Lithography", *Proc. International Symposium on Quality Electronic Design*, 2010, pp. 122–130.
  - **Kwangok Jeong**, Andrew B. Kahng and Rasit O. Topaloglu, "Is Overlay Error More Important Than Interconnect Variations in Double Patterning?", *Proc. ACM International Workshop on System-Level Interconnect Prediction*, 2009, pp. 3–10.
  - **Kwangok Jeong** and Andrew B. Kahng, "Timing Analysis and Optimization Implications of Bimodal CD Distribution in Double Patterning Lithography", *Proc. Asia and South Pacific Design Automation Conference*, 2009, pp. 486–491.

- Robert T. Greenway, Rudolf Hendel, **Kwangok Jeong**, Andrew B. Kahng, John S. Petersen, Zhilong Rao and Michael C. Smayling, "Interference Assisted Lithography for Patterning of 1D Gridded Design", *Proc. SPIE Symposium on Advanced Lithography*, Vol. 7271, 2009, pp. 72712U-1–72712U-11.
- Robert T. Greenway, **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and John S. Petersen, "32nm 1-D Regular Pitch SRAM Bitcell Design for Interference-Assisted Lithography", *Proc. SPIE BACUS Symposium on Photomask Technology and Management*, Vol. 7122, 2008, pp. 71221L-1–71221L-12.

- Chapter 5 is based on the following publications:
  - **Kwangok Jeong**, Andrew B. Kahng and Christopher J. Progler, "Cost-Driven Mask Strategies Considering Parametric Yield, Defectivity and Production Volume", submitted to *SPIE Journal of Microlithography, Microfabrication and Microsystems*, 2011.
  - **Kwangok Jeong**, Andrew B. Kahng and Christopher J. Progler, "New Yield-Aware Mask Strategies", to appear in *Proc. Photomask and Next-Generation Lithography Mask Technology*, 2011.
  - **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and Hailong Yao, "Dose Map and Placement Co-Optimization for Improved Timing Yield and Leakage Power", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(7) (2010), pp. 1070–1082.
  - **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and Hailong Yao, "Dose Map and Placement Co-Optimization for Timing Yield Enhancement and Leakage Power Reduction", *Proc. ACM/IEEE Design Automation Conference*, 2008, pp. 516–521.

- Chapter 6 is based on the following publications:
  - Mohit Gupta, **Kwangok Jeong** and Andrew B. Kahng, "Timing Yield-Aware Color Reassignment and Detailed Placement Perturbation for Bimodal CD

Distribution in Double Patterning Lithography", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(8) (2010), pp. 1229–1242.

- **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and Hailong Yao, "Dose Map and Placement Co-Optimization for Improved Timing Yield and Leakage Power", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(7) (2010), pp. 1070–1082.
- Mohit Gupta, **Kwangok Jeong** and Andrew B. Kahng, "Timing Yield-Aware Color Reassignment and Detailed Placement Perturbation for Double Patterning Lithography", *Proc. ACM/IEEE International Conference on Computer-Aided Design*, 2009, pp. 607–614.
- Robert T. Greenway, Rudolf Hendel, **Kwangok Jeong**, Andrew B. Kahng, John S. Petersen, Zhilong Rao and Michael C. Smayling, "Interference Assisted Lithography for Patterning of 1D Gridded Design", *Proc. SPIE Symposium on Advanced Lithography*, Vol. 7271, 2009, pp. 72712U-1–72712U-11.
- **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and Hailong Yao, "Dose Map and Placement Co-Optimization for Timing Yield Enhancement and Leakage Power Reduction", *Proc. ACM/IEEE Design Automation Conference*, 2008, pp. 516–521.
- Robert T. Greenway, **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and John S. Petersen, "32nm 1-D Regular Pitch SRAM Bitcell Design for Interference-Assisted Lithography", *Proc. SPIE BACUS Symposium on Photomask Technology and Management*, Vol. 7122, 2008, pp. 71221L-1–71221L-12.

• Chapter 7 is based on the following publications:

- Puneet Gupta, **Kwangok Jeong**, Andrew B. Kahng and Chul-Hong Park, "Electrical Assessment of Lithographic Gate Line-End Patterning", *SPIE Journal of Microlithography, Microfabrication and Microsystems* 9(2) (2010), pp. 023014-1–023014-19.

- Puneet Gupta, **Kwangok Jeong**, Andrew B. Kahng and Chul-Hong Park, "Electrical Metrics for Lithographic Line-End Tapering", *Proc. Photomask and Next-Generation Lithography Mask Technology*, 2008, pp. 70238A-1–70238A-12.

My coauthors (Mr. Robert T. Greenway, Mr. Mohit Gupta, Professor Puneet Gupta, Dr. Rudolf Hendel, Professor Andrew B. Kahng, Dr. Chul-Hong Park, Mr. John S. Petersen, Dr. Christopher J. Progler, Dr. Zhilong Rao, Dr. Kambiz Samadi, Dr. Michael C. Smayling, Dr. Rasit O. Topaloglu and Professor Hailong Yao, listed in alphabetical order) have all kindly approved the inclusion of the aforementioned publications in my thesis.

VITA

1974	Born, Seoul, South Korea
1997	B.Sc., Electrical Engineering, Hanyang University, Seoul, Korea
1999	M.Sc., Electrical Engineering Hanyang University, Seoul, Korea
2010	C.Phil., Electrical Engineering (Computer Engineering), University of California, San Diego
2011	Ph.D., Electrical Engineering (Computer Engineering), University of California, San Diego

All papers coauthored with my advisor Prof. Andrew B. Kahng have authors listed in alphabetical order.

- Tuck-Boon Chan, **Kwangok Jeong** and Andrew B. Kahng, “Performance and Variability Driven Guidelines for BEOL Layout Decomposition with LELE Double Patterning”, to appear in *Proc. SPIE Photomask Technology*, 2011.
- **Kwangok Jeong**, Andrew B. Kahng and Christopher J. Progler, “New Yield-Aware Mask Strategies”, to appear in *Proc. SPIE Photomask and Next-Generation Lithography Mask Technology*, 2011.
- **Kwangok Jeong** and Andrew B. Kahng, “Toward PDN Resource Estimation: A Law of Power Density”, to appear in *Proc. ACM International Workshop on System-Level Interconnect Prediction*, 2011.
- Sung Kyu Han, **Kwangok Jeong**, Andrew B. Kahng and Jingwei Lu, “Stability and Scalability in Global Routing”, to appear in *Proc. ACM International Workshop on System-Level Interconnect Prediction*, 2011.
- **Kwangok Jeong** and Andrew B. Kahng, “Variation Mapping From Timing Path Delay Measurements Using Compressed Sensing”, *Proc. ACM International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems*, 2011, pp. 1–6.

- **Kwangok Jeong**, Andrew B. Kahng and Seokhyeong Kang, “Toward Effective Utilization of Timing Exceptions in Design Optimization”, *Proc. International Symposium on Quality Electronic Design*, 2010, pp. 54–61.

- **Kwangok Jeong** and Andrew B. Kahng, “Methodology from Chaos in IC Implementation”, *Proc. International Symposium on Quality Electronic Design*, 2010, pp. 885–892.

- **Kwangok Jeong**, Andrew B. Kahng and Rasit O. Topaloglu, “Assessing Chip-Level Impact of Double Patterning Lithography”, *Proc. International Symposium on Quality Electronic Design*, 2010, pp. 122–130.

- **Kwangok Jeong**, Andrew B. Kahng, Bill Lin and Kambiz Samadi, “Accurate Machine Learning-Based On-Chip Router Modeling”, *IEEE Embedded Systems Letters* 2(3) (2010), pp. 62–66.

- Puneet Gupta, **Kwangok Jeong**, Andrew B. Kahng and Chul-Hong Park, “Electrical Assessment of Lithographic Gate Line-End Patterning”, *SPIE Journal of Microlithography, Microfabrication and Microsystems* 9(2) (2010), pp. 023014-1–023014-19.

- Mohit Gupta, **Kwangok Jeong** and Andrew B. Kahng, “Timing Yield-Aware Color Reassignment and Detailed Placement Perturbation for Bimodal CD Distribution in Double Patterning Lithography”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(8) (2010), pp. 1229–1242.

- **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and Hailong Yao, “Dose Map and Placement Co-Optimization for Improved Timing Yield and Leakage Power”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(7) (2010), pp. 1070–1082.

- **Kwangok Jeong** and A. B. Kahng, “A Power-Constrained MPU Roadmap for the International Technology Roadmap for Semiconductors (ITRS)”, *Proc. International SoC Design Conference*, 2009, pp. 49–52.

- **Kwangok Jeong**, Andrew B. Kahng and Kambiz Samadi, "Architectural-Level Prediction of Interconnect Wirelength and Fanout", *Proc. International SoC Design Conference*, 2009, pp. 53–56.

- **Kwangok Jeong**, Andrew B. Kahng and Kambiz Samadi, "Impacts of Guard-band Reduction on Design Process Outcomes: A Quantitative Approach", *IEEE Transactions on Semiconductor Manufacturing* 22(4) (2009), pp. 552–565.

- Mohit Gupta, **Kwangok Jeong** and Andrew B. Kahng, "Timing Yield-Aware Color Reassignment and Detailed Placement Perturbation for Double Patterning Lithography", *Proc. ACM/IEEE International Conference on Computer-Aided Design*, 2009, pp. 607–614.

- **Kwangok Jeong**, Andrew B. Kahng and Rasit O. Topaloglu, "Is Overlay Error More Important Than Interconnect Variations in Double Patterning?", *Proc. ACM International Workshop on System-Level Interconnect Prediction*, 2009, pp. 3–10.

- Robert T. Greenway, Rudolf Hendel, **Kwangok Jeong**, Andrew B. Kahng, John S. Petersen, Zhilong Rao and Michael C. Smayling, "Interference Assisted Lithography for Patterning of 1D Gridded Design", *Proc. SPIE Symposium on Advanced Lithography*, Vol. 7271, 2009, 72712U-1–72712U-11.

- **Kwangok Jeong**, Andrew B. Kahng and Hailong Yao, "Revisiting the Linear Programming Framework for Leakage Power vs. Performance Optimization", *Proc. International Symposium on Quality Electronic Design*, 2009, pp. 127–134.

- **Kwangok Jeong** and Andrew B. Kahng, "Timing Analysis and Optimization Implications of Bimodal CD Distribution in Double Patterning Lithography", *Proc. Asia and South Pacific Design Automation Conference*, 2009, pp. 486–491.

- **Kwangok Jeong**, Andrew B. Kahng and Hailong Yao, "On Modeling and Sensitivity of Via Count in SOC Physical Implementation", *Proc. International SoC Design Conference*, 2008, pp. 125–128.

- Robert T. Greenway, **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and John S. Petersen, "32nm 1-D Regular Pitch SRAM Bitcell Design for Interference-Assisted Lithography", *Proc. SPIE BACUS Symposium on Photomask Technology and Management*, Vol. 7122, 2008, 71221L-1–71221L-12.

- **Kwangok Jeong**, Andrew B. Kahng, Chul-Hong Park and Hailong Yao, "Dose Map and Placement Co-Optimization for Timing Yield Enhancement and Leakage Power Reduction", *Proc. ACM/IEEE Design Automation Conference*, 2008, pp. 516–521.

- Puneet Gupta, **Kwangok Jeong**, Andrew B. Kahng and Chul-Hong Park, "Electrical Metrics for Lithographic Line-End Tapering", *Proc. Photomask and Next-Generation Lithography Mask Technology*, 2008, pp. 70238A-1–70238A-12.

- **Kwangok Jeong**, Andrew B. Kahng and Kambiz Samadi, "Quantified Impacts of Guardband Reduction on Design Process Outcomes", *Proc. International Symposium on Quality Electronic Design*, 2008, pp. 790–897.

ABSTRACT OF THE DISSERTATION

Variability Assessment and Mitigation in Advanced VLSI Manufacturing Through Design-Manufacturing Co-Optimization

by

Kwangok Jeong

Doctor of Philosophy in Electrical Engineering (Computer Engineering)

University of California, San Diego, 2011

Professor Andrew B. Kahng, Chair

Increasing variability in today's manufacturing processes causes parametric yield loss that increases manufacturing cost. In spite of the tremendous effort and enhancement from both manufacturing and design sides, problematic systematic variations still remain uncompensated. In addition, while new manufacturing techniques have been adopted to reduce variability by improving pattern fidelity in the subwavelength lithography regime, new techniques continually introduce new sources of variabilities. To mitigate any remaining or emerging variabilities, accurate modeling and assessment of the variabilities through detailed analyses of underlying physical mechanisms is essential. Appropriate optimizations in both design and manufacturing must be developed, based on comprehensive understanding of the benefits and costs of such new measures.

This thesis first quantifies impact of guardband reduction on design outcomes, and resulting yield and cost, to objectively evaluate the true benefits of various guardband reduction techniques. Cost-effective guardband reduction techniques are then presented for both design and manufacturing. The proposed measures span multiple stages of design, manufacturing and implementation: (1) from basic circuit elements such as device, interconnect, logic gates and memory bitcells, to high-level design implementation phases such as logic synthesis, placement and routing, and (2) from mask generation and lithography, to post-silicon variation measurement.

The innovative techniques proposed in this thesis can be grouped into three main thrusts: (1) variability modeling and mapping, (2) variation assessment, and (3) variability mitigation.

In the *variability modeling and mapping* thrust, this thesis reviews various variation modeling techniques and proposes a novel variation mapping framework (based on compressed sensing theory) that reconstructs the details of multiple, simultaneously occurring systematic variation maps from measurements of a small number of naturally-occurring timing paths within the design.

In the *variability assessment* thrust, this thesis proposes techniques to quantify variability in advanced lithography techniques, such as double patterning lithography and interference-assisted lithography, and provides useful observations for designers and manufacturers to tradeoff quality of results versus design and manufacturing cost.

In the *variability mitigation* thrust, this thesis presents three distinct approaches to explicitly mitigate variations and enable principled tradeoffs between design cost and yield. First, design-aware manufacturing process optimization provides optimal mask strategies considering parametric and defect yields, and optimal exposure dose maps considering design timing and leakage power. Second, manufacturing-aware design optimizations include a cell swapping-based placement optimization, timing yield-aware detailed placement optimization to mitigate impact of bimodal CD distribution in double patterning lithography, and development of a new 1-D regular pitch SRAM bitcell for interference-assisted lithography. Finally, design-manufacturing co-optimization includes a first-ever elucidation of the tradeoff between electrical performance and manufacturing cost for modern transistor fabrication.

Chapter 1

Introduction

Semiconductor markets continually demand more functional diversity in integrated-circuit (IC) products, and the semiconductor industry must continually lower costs in order to satisfy this demand. The most significant trend is decreasing cost-per-function, which has led to significant improvements in economic productivity and overall quality of life through proliferation of computers, communication, and other industrial and consumer electronics. Through a trajectory often referred to as "Moore's Law", the semiconductor industry has sustained the rapid pace of improvement in its products by exponentially decreasing the minimum feature sizes used to pattern and fabricate ICs. However, as minimum feature sizes approach physical limits, many hitherto neglected or unnoticed phenomena have recently emerged as critical challenges (in terms of variability, performance, power and yield) to further advances in semiconductor-based integration. Although the challenges may be solved or relaxed with the advancement of manufacturing techniques, it is difficult for the semiconductor industry to come to grips with rapidly increasing costs of processing equipment and manufacturing in advanced technologies. Innovation with respect to devices, materials, fabrication processes, and other traditional levers for technology scaling has become extremely expensive, and the industry's technology roadmap is now fraught with uncertainty and risk. This thesis seeks to enable the continuation of cost-effective integrated-circuit innovation with leading-edge manufacturing technologies. In this context, the co-optimization of design and manufacturing offers a lifeline – a "design-based equivalent scaling" path forward – for the semiconductor industry.

1.1 Traditional Design and Manufacturing Optimization Techniques

Higher production yield, which directly reduces the cost of the manufacturing, has been one of the primary goals for manufacturers and designers to survive in today's competitive IC world. Yield is defined as the number of chips that function correctly and satisfy timing and power specifications, expressed as a percentage of the total number of chips manufactured. Yield is low at the process development and ramp-up stages, and increases, in general, to over 90% for a mature process. Yield is commonly classified into the following two categories.

- *Functional yield* or *catastrophic yield* is the percentage of chips that are functional. Examples of functional failures that limit functional yield are shorts and opens in wires and vias, as well as line-end shortening.

- *Parametric yield* is the percentage of the functionally-correct chips that satisfy given delay and power specifications.

Many types of process variations and defects cause yield loss. Functional yield loss is usually caused by misprocessing and random contaminant-related defects. Parametric yield loss is typically due to process variations. However, process variations can also cause functional failures (e.g., line-end shortening leading to an always-on device), while defects can cause parametric yield loss (e.g., particle contamination that causes interconnect narrowing but not a complete open).

While yield loss due to functional failures is significant, parametric failures have gained significance and now dominate functional failures. Arguably, measures to improve parametric yield are more challenging to develop and adopt. A variety of techniques have been introduced to improve parametric yield from both the manufacturing side and the design side.

1.1.1 Manufacturing-Side Approaches

Photolithography has been a key IC manufacturing technique. Photolithography is the process of transferring circuit patterns on a layout to the surface of a silicon

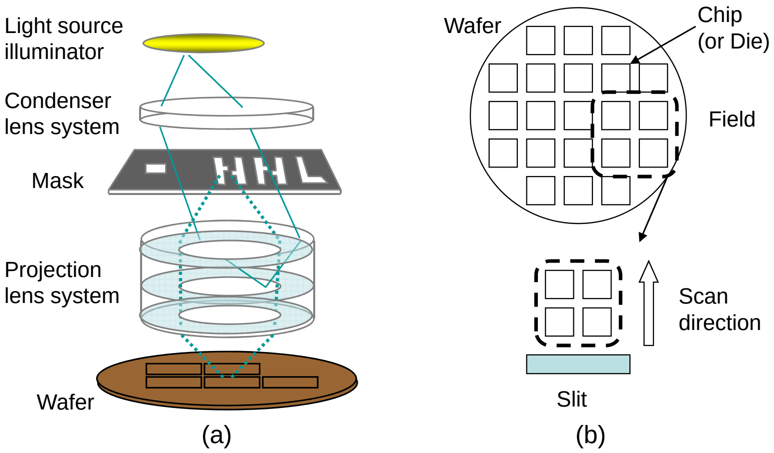


Figure 1.1: Schematic of photolithography system: (a) a step-and-scan system and (b) exposure field scanned by slit.

wafer. It uses light to transfer a geometric pattern from a photomask (also referred to as a reticle, or simply "mask") to a light-sensitive chemical (photoresist, or simply "resist") on a silicon substrate. Photolithography involves a complex series of steps, e.g., *resist coating*, *soft bake*, *exposure*, *post-exposure bake* (PEB) and *development*. The series of chemical treatments ultimately engrave the exposed patterns into the material underneath the photoresist.

Among the photolithography steps, *optical lithography* (i.e., exposure and development) is the most important step to create nanoscale circuit patterns. The modern lithography tool is a step-and-scan system which is a hybrid of scanner and stepper systems. A scanner projects a slit of light from a mask onto the wafer through the optical lens system. During the scanning operation, a small portion of the wafer, called a *field*, is exposed through a mask. Multiple copies of a chip on the mask can be printed onto the wafer. Then, the wafer is stepped to a new location and the scanning operation is repeated. A simple schematic of an optical lithography system with its main components – light source, condenser lens, mask, projection (or objective) lens, and the wafer – is shown in Figure 1.1.

Optical lithography typically uses deep-ultraviolet excimer (or exciplex) lasers that have different wavelengths according to technology nodes; krypton fluoride lasers (KrF: 248nm wavelength) have been used in 180nm $\sim$ 130nm technology nodes, and argon fluoride lasers (ArF: 193nm wavelength) have been applied to 90nm $\sim$ 32nm

nodes. The condenser lens serves to deliver uniform light with adequate intensity to the mask. The projection lens captures some portion of the diffraction order through the mask, and then delivers the image onto the wafer. However, due to the finite size of lenses and the higher diffraction orders of light, pattern information generated by the *Fourier transform* of the mask is not completely captured. The loss of diffraction information leads to limited image quality and resolution. This resolution limit of optical lithography is described by the Rayleigh equation [128]

$$R = k_1 \tfrac{\lambda}{NA}. \tag{1.1}$$

Here, R is the minimum half pitch that can be implemented on the silicon wafer; λ is the exposure wavelength of the illumination source; k_1 is a process-dependent factor determined mainly by the resist capability, the tool control, the reticle pattern adjustments and the process control; and NA is the *numerical aperture* of the projection lens, which is a measure of the ability to capture diffraction orders.

To achieve smaller and denser patterns, smaller k_1 and higher NA are required. k_1 has a fundamental lower limit of 0.25. With the adoption of double patterning lithography, it is in the range of 0.18 $\sim$ 0.28 for the 32nm technology node [33]. NA equals $n \sin\theta$ where θ is the diffraction angle, and n is the minimum index of refraction of the image medium (1.0 for air, 1.33 for pure water, and up to 1.56 for oils). The sine of the maximum half-angle of light determines the light that passes through a lens to the wafer [128].

However, higher NA can degrade the pattern quality. The *depth of focus* (DOF) is one of the key measures to assess printing quality and robustness (i.e., *process margin*). The DOF is given as

$$DOF = k_2 \tfrac{\lambda}{(NA)^2} \tag{1.2}$$

where k_2 is an empirical constant. Because of the inverse dependency with NA^2, depth of focus with high NA is extremely shallow. For this reason, techniques to minimize wafer topography, notably *chemical-mechanical planarization* (CMP), are required [179]. Ideally, the top of the wafer plane must coincide with the focal plane of the objective lens, resulting in formation of the image at the best focus. If the wafer surface deviates from the focal plane of the lens, then the aerial image is transferred out of focus. DOF is defined as the maximum shift in focus that maintains tolerable deviation between the achieved image and its intended shape.

Scaling of physical dimensions, faster than the advances in wavelengths, materials and equipments, has led to increased lateral dimension variability in photolithography. In the *front-end-of-line* (FEOL), variations in gate polysilicon (poly) length (i.e., device gate length) and diffusion region size cause transistor delay and leakage variations. In the *back-end-of-line* (BEOL), variations in wire width cause variations in parasitics and consequently in wire delays and signal integrity.

To achieve high fidelity of silicon shapes to "drawn" shapes, several *resolution enhancement techniques* (RETs) have been developed [156]. RETs are methods used in lithography to enhance the printability of mask features. RETs are typically applied after design signoff, and before or during the mask data preparation stage. Commonly used RETs are as follows.

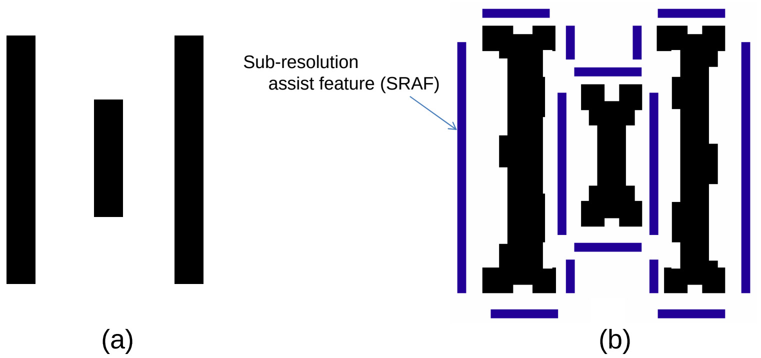


Figure 1.2: Examples of resolution enhancement techniques: (a) drawn layout and (b) mask patterns after OPC and SRAF insertion.

- *Optical proximity correction* (OPC) [140] [172] [61] selectively alters the shapes of the mask patterns as shown in Figure 1.2 to compensate for patterning imperfections from subwavelength lithography. Rule-based OPC designs the mask aperture based on predefined rules for layout configurations such as corners or notches. Model-based OPC uses a lithography simulator in the loop as it optimizes the mask pattern. While OPC is very effective at reducing patterning variation, it requires large runtime and significantly increases the mask complexity.

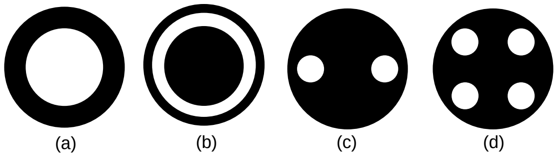


Figure 1.3: Examples of aperture shapes for OAI: (a) circular, (b) annular, (c) dipole and (d) quadrupole apertures.

- *Subresolution assist features* (SRAFs) or *scattering bars* (SB) [54] are layout features that are inserted between layout features to improve their printability through contrast-enhancing interference. SRAFs have narrow widths as shown in Figure 1.2(b) and do not print on the wafer.

- *Off-axis illumination* (OAI) [137] [106] generally refers to illumination which intentionally has an off-axis component, i.e., which includes light that is not normally incident to the mask. Off-axis illumination uses circular, annular, dipole or quadrupole apertures as shown in Figure 1.3. OAI improves the DOF for certain pitches while worsening the DOF for other pitches, which can become *forbidden pitches*.

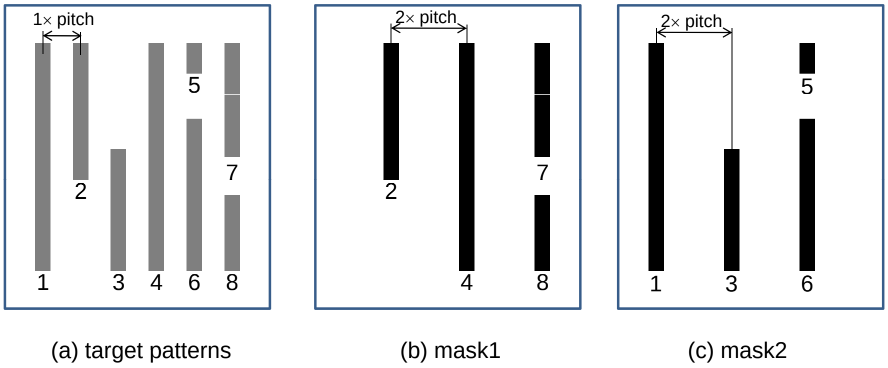
- *Double dipole exposure* [92] [32] is another attractive technique due to the relatively low cost of binary and attenuated phase shifting masks, both of which can be used with dipole illumination. Double dipole exposure splits the design of two-axis patterns into two separate (horizontal segments, and vertical segments) layers, so that two different apertures (i.e., x-axis parallel and y-axis parallel poles) filter out high diffraction orders for each axis.

- *Phase shift mask* (PSM) [117] [122] changes the depth of clear (transmissive) regions of the mask in certain locations to shift the phase of transmitted light and therby induce destructive interference at feature edges; this enhances pattern contrast and resolution.

New patterning techniques, enabled by new types of new lithography equipment, have also been introduced to overcome subwavelength lithography challenges.

- *Immersion ArF* (IArF) [173] [141] uses a liquid medium between lenses and the resist stack. The Rayleigh equation can be rewritten as $R = k_1 \frac{\lambda/n}{\sin\theta}$. *Effective wavelength* can be defined as λ/n for a given diffraction angle θ [141]. The liquid medium gives rise to smaller effective wavelength with the same or even larger-wavelength light, due to its larger refraction index (e.g., n = 1.44 for water). For instance, the effective wavelength of 193nm dry lithography is just 193nm, while that of 193nm water immersion lithography is 134nm. This smaller effective wavelength has enabled 45nm technology with traditional 193nm ArF.

- *Double patterning lithography* (DPL) [63] [55] [91] [167] [93] achieves two times higher resolution than traditional single exposure lithography. In double patterning lithography (DPL), pitch is effectively managed by *pitch splitting*: two patterns with less than a given spacing are separated and assigned to different masks as shown in Figure 1.4. Patterns in the two masks are transferred sequentially to a single layer on the wafer. While double dipole exposure splits patterns according to pattern direction, DPL improves effective imaging resolution by splitting a desired pitch into two portions, each with a lower spatial frequency than the original.



(a) target patterns (b) mask1 (c) mask2

Figure 1.4: DPL partitions target patterns in a critical layer (with 1× pitch) into two masks (with 2× pitch).

- *Interference lithography* (IL) [66] generates patterns without any mask. Controlled interference of two or four beams produces high-contrast, regular and high-resolution dense grating patterns. Since mask cost for conventional projection lithography increases continuously due to complex RETs, the maskless process of IL is attractive for future lithography. A drawback of IL is that it is restricted to producing periodic patterns only. A *hybrid optical maskless* (HOMA) lithography approach [66] uses a second exposure step with traditional projection lithography to implement final complex patterns.

- *Extreme ultraviolet lithography* (EUVL) [34] uses significantly smaller wavelength ($13.5nm$) than the traditional deep-ultraviolet lithography. All matter absorbs EUV radiation. Hence, EUVL is performed in a vacuum. In addition, to solve the absorption problem through lenses, EUVL uses several mirrors that reflect the light, and finally a multi-layer (ML) Mo-Si reflective mask is used to implement patterns on photoresist. Given that EUVL significantly reduces wavelength compared to the traditional lithography, EUVL is expected to have very high resolution. However, many technical hurdles still delay early adoption of EUVL in production. These include fabrication of low-defectivity mask blanks, development of reliable EUV sources with high output power and sufficient lifetime for surrounding collector optics, controlling contamination of mirrors, development of resists with sufficiently low linewidth roughness and low exposure dose, and protection of masks from defects without pellicles [11].

- *Nanoimprint lithography* [151] [74] is based on the concept that a mold or template with nanostructures on its surface can be pressed against a substrate that has been coated with resist material, so as to replicate patterns by physical or chemical methods. Imprint lithography has the potential to be a cost-effective solution, but there are a number of problems that need to be solved, such as difficulties associated with the requirement of a $1\times$ template, defects, template lifetime, overlay, etc.

Another effort from the manufacturing side has been to take on variation modeling. As manufacturing techniques evolve, the measurable magnitude of variation decreases. However, due to smaller and faster devices, sensitivity to manufacturing variation has increased. This requires more accurate process variation measurement and modeling. *Systematic process variations* can be deterministically modeled and compensated with measurable parameters such as *critical dimension* (CD) of polysilicon gate, saturation current ($I_{d,sat}$), off-state current (I_{off}), threshold voltage (V_{th}), ring oscillator (RO) frequency, etc. To accurately model the variation within a die, field or wafer, customized test structures, e.g., arrays of measurement structures, or on-chip testing/sensing units, have been used. Agarwal et al. [31] use several array structures and test methodologies to measure I_{off} and V_{th}. Friedberg et al. [65] design CD test structures to capture variations in gate length. Given measured data, a number of variation modeling techniques have been studied. Stine et al. [170] use various fitting techniques, such as least-squares regression, cubic spline, etc. Friedberg et al. [65] use pointwise averaging of initial data, and model systematic variation using least-squares regression.

1.1.2 Design-Side Approaches

Traditionally, guardbanding has been the only available knob by which designers can trade off design cost and production yield. Overdesign assuming worst-case impact of variations has been widely accepted, although large guardband makes the final chip signoff tougher than it needs to be, incurring significant design turnaround time and cost increase. Furthermore, guardbands may be applied incorrectly due to lack of understanding of the systematic nature of the variations. More explicit approaches have been developed as *design for manufacturing* (DFM) from the design side. DFM is a set of techniques by which designers can improve electrical characteristics and yield, based on improved understanding of process variations.

Variation assessment. Many works address the need for accurate analysis of manufacturing variabilities, and quantification of the impact of such variabilities on design. Balasinski et al. [37] propose a methodology of manufacturability qualification for ultra-deep submicron circuits, based on optical simulation of the layout, integrated with device simulation; see also [164]. Pack et al. [143] propose to incorporate advanced models of lithographic printing effects into the design flow to improve yield and performance verification accuracy. Gupta et al. [82] observe that lithography simulation permits post-OPC (optical proximity correction) estimation of on-silicon feature sizes at different process conditions. Yang et al. [192] address post-lithography based analysis and optimization, proposing a timing analysis flow based on residual OPC errors (equivalent to lithography simulation output) for timing-critical cells and their layout neighborhoods. Cao et al. [49] propose a methodology for standard-cell characterization considering litho-induced systematic variations. In [49], the objective is to enable efficient post-litho analysis by running litho-aware characterization. Furthermore, to minimize the difference between isolated and actual placement contexts of a given standard cell, vertical dummy poly patterns are inserted at the cell boundary. Finally, it is noteworthy that Gupta and Heng [75] perform "iso-dense aware" timing analysis (based on modeling of systematic through-focus L_{eff} variation) to achieve up to 40% reduction of the best-case/worst-case guardband in static timing analysis. Also, Sylvester et al. [174] observe that up to 60% of BEOL guardband can be eliminated by use of realistic BEOL variation models.

Variation mitigation. Many design techniques to mitigate manufacturing variabilities have been proposed. An example of a traditional variation mitigation technique is *fill insertion*. Fill insertion has been an important knob to improve pattern uniformity and thus reduce variability. *Chemical-mechanical polishing* (or *planarization*) (CMP) is performed between lithography steps to attain the designed layer height and to planarize the layer for succeeding process steps. Unfortunately, CMP is imperfect and cannot completely eliminate topography variation. Topography variation changes the metal height in BEOL layers which affects the wire resistance and capacitance. CMP for FEOL is used to planarize the oxide that is deposited for *shallow trench isolation* (STI). Imperfect FEOL CMP leads to defocus during polysilicon patterning and poor inter-device isolation. To improve CMP performance, layouts with uniform pattern density are required.

Other examples of variation mitigation are seen in the optical lithography context. As noted above some pitches, especially with the use of OAI, have poor printability; these pitches are known as *forbidden pitches*. Scattering bars reduce the occurrences of forbidden pitches and enhance printability. Etch dummies are non-functional geometries added to the active layer to protect devices near the active edges from ion scattering during etching. Gupta et al. [84] propose optimal *scattering bar* and *etch dummy* insertion techniques using dynamic programming-based detailed placement algorithms to reduce or eliminate the number of forbidden pitches. Scattering bar insertion interferes with etch dummy insertion because of spacing rules between specific etch dummies and scattering bars. A scattering bar-aware etch dummy insertion flow is also proposed in [84] to make the layout more conducive to scattering bar insertion after etch dummy insertion. Finally, a detailed placement approach for etch dummy insertion is also proposed. The reported results show substantial reduction in the number of forbidden pitches and in the *edge placement error* (EPE) due to exposure and etch non-idealities.

Kahng et al. [102] propose the use of *auxiliary patterns* which are similar in function to scattering bars but wider and hence more effective at shielding critical patterns and their OPC treatments from proximity effects. The disadvantage of auxiliary patterns is that, unlike scattering bars, they are printed on the wafer and may require whitespace for their insertion. A detailed placement approach is proposed to apply auxiliary patterns to a design with no area overhead. The approach is proposed in the context of cell-based OPC to reduce OPC runtime, but can be used to reduce CD errors that arise due to optical proximity effects (i.e., through-pitch CD variation).

Variation exploitation. Exploitation of the unavoidable systematic variations is another research direction in design for manufacturing. Mechanical stress on active regions of devices, arising due to the proximity and width of STI wells, is significant in existing technologies. Stress due to STI is compressive and typically enhances the mobility of PMOS while degrading the mobility of NMOS. Consequently, delay and leakage increase for PMOS while decreasing for NMOS. Several techniques have been proposed to reduce STI stress-induced variation. Kahng et al. [104] present timing-driven optimization of STI stress in standard cell designs, using detailed placement perturbation and active-layer fill insertion to improve CMOS performance.

Gupta et al. [81] propose a timing optimization approach that exploits the opposite lithography-induced gate length variations experienced by dense and isolated pitches to compensate for each other. In their process, gate lengths of dense devices (i.e., devices with small spacings from neighboring devices) increase with defocus, while those of isolated devices decrease. Gupta et al. [81] construct isolated and dense variants for all cells in the library. An optimizer is then used to map each of the cell instances to either a dense or an isolated variant from the library. The objective of the optimizer is to use a mix of isolated and dense variants, such that the delay and leakage variabilities due to defocus become essentially " self-compensated".

Regular layout. In addition, new layout styles aiming at more regular design have been suggested to achieve reliable printability of subwavelength features. Gupta and Kahng [78] point out that full-chip layouts may need to be assembled as a collection of regular printable patterns for technologies at 90nm and beyond. Lavin et al. [116] propose "*layout using gridded glyph geometry objects*" (L3GO) with points, sticks and rectangle glyphs, to improve manufacturability. Using the glyph-based layout methodology, a circuit may avoid manufacturing challenges that arise from design irregularity. Liebmann et al. [120] propose a rule-based layout optimization methodology based on *restrictive design rules* (RDRs) to control linewidth on the poly layer. Having a limited number of linewidths along with single orientation of features, RDRs present new challenges to automatic design migration. Wang et al. [184] study the impact of grid-placed contacts on *application-specific integrated circuit* (ASIC) performance. A grid-based layout scheme allows layout to be partitioned for double exposure illumination [185]. Jhaveri et al. [99] introduce the concept of a regular design fabric for defining the underlying silicon geometries of a circuit. They also discuss the benefits of using extremely regular designs constructed from a limited set of lithography-friendly patterns. Using a "*pushed rule*", the area penalty which has been one of the drawbacks of grid-based layouts is reduced. Maly et al. [130] propose "*lithographer's dream patterns*" (LDP), a methodology that incorporates extremely regular and uniform layout patterns with a large number of dummy patterns.

1.2 Problems: Left and Emerging on the Table

In spite of the tremendous effort and innovation from both manufacturing and design sides, problematic systematic variations still remain uncompensated. In addition, while new manufacturing techniques have been adopted to reduce variability (and thus guardband) by improving pattern fidelity in the subwavelength lithography regime, new techniques continually introduce new variabilities.

Inefficiency of variation modeling. Figure 1.5 illustrates various types of traditional variation measurement structures: (a) device array, (b) on-chip sensors, (c) on-chip critical paths, and (d) test element group (TEG). Massive test structures such as device array [31] may continue to be important to measure variation profiles and major characteristics of manufacturing process. However, measuring CD or electrical characteristics of individual transistors requires a large amount of test time and cost, as well as valuable silicon area or additional processing complexity. A test element group (TEG) in scribelines (i.e., the gaps between dies) can contain various measurement circuits, preserving the silicon area for actual designs. However, variation observed at the scribelines may not be well-correlated with the variation in actual products, so that the use of scribeline TEG may be limited to monitoring process abnormality.

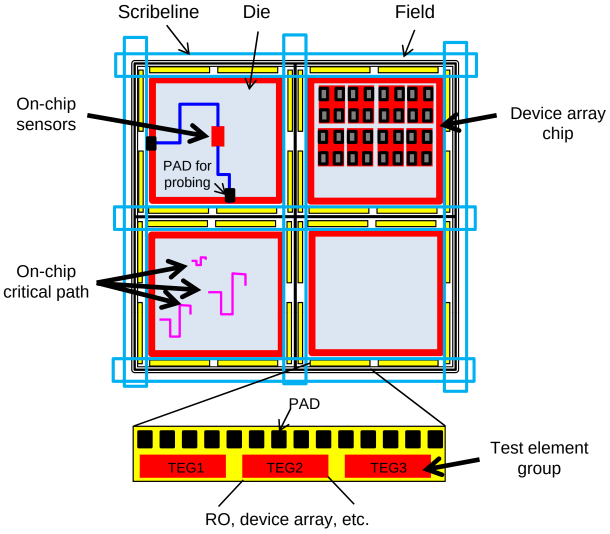


Figure 1.5: Examples of variation measurements.

On-chip sensing circuits, e.g., temperature and/or voltage sensors, ring oscillators, and either actual or mimicked timing paths [125], can also be used to model variability of actual products. However, due to the disruptive nature of on-chip measurement circuits – constraining optimization of actual products as placement or routing blockages, it may be difficult to increase the number of sensing circuits enough to model detailed variations. Efficient modeling techniques can help reduce the overhead of embedding and measuring the test structures. Regression-based modeling approaches have been used to find a simple closed-form representation of variation with *a priori* knowledge or assumption of the typical shape of underlying variations. However, its accuracy is limited since regression-based approaches rely on *a priori* knowledge or assumption of the typical shape of underlying variations, and for more detailed analysis high-order model function is necessary with sufficiently large number of samples.

Inaccuracy of modeling pattern imperfection. In the low-k_1 patterning regime ($k_1 < 0.3$), gate shape is no longer a perfect rectangle. Linewidths vary, corners are rounded, and small features disappear as shown in Figure 1.6. Current circuit analysis tools assume that transistor gate and diffusion shapes are perfect rectangles, and are unable to handle complicated geometries. Large discrepancies can be observed between the simulated and measured values of transistor parameters such as current and threshold voltage. Moreover, such discrepancies are likely to become more significant as overlay becomes a more critical issue in modern technologies.

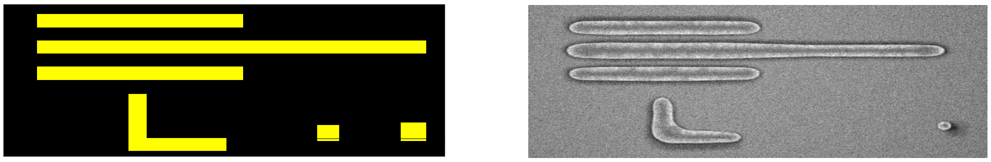

(a) drawn patterns (b) patterns on silicon

Figure 1.6: Examples of the pattern imperfection. Drawn patterns do not appear exactly on silicon.

Need for new cost-driven mask strategies. Traditionally, a large mask consisting of multiple copies of a die has been used to maximize lithography throughput, since a large mask can print many dies at a time as shown in Figure 1.7. However, as reticle size increases, the mask cost (write, inspection, defect disposition, repair, etc.) increases. For high-volume products, mask cost can be disregarded, but for small-volume products – in light of shuttle-based prototyping, design revisions and respins, market competition, and other reasons – mask cost can significantly impact overall cost per die. Mask writing cost, lithography cost, and mask yield all vary with reticle size. However, in today's IC manufacturing, the maximum possible reticle size is traditionally used. As photomask cost excessively increases with the adoption of new patterning techniques, we now need new cost-effective reticle strategies. To this end, a complete cost model comprehending mask generation, lithography, and yield due to various reticle strategies is required.

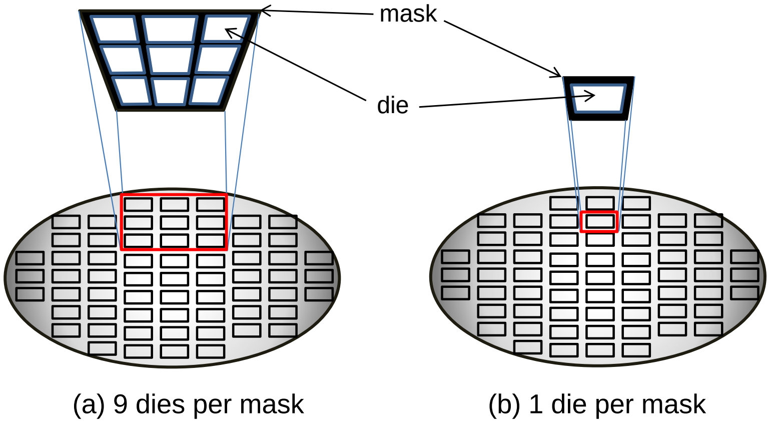


Figure 1.7: A typical large-sized mask (left) and a small-sized mask (right).

Need for adaptive process control. A recent technology from ASML [1], called *Dose Mapper* [196] [94] [113], allows for minimization of *across-chip linewidth variation* (ACLV) and *across-wafer linewidth variation* (AWLV) using an exposure dose (or, simply, dose) correction scheme. ACLV is primarily caused by the mask and scanner, while AWLV is affected by the track and etcher [159]. Dose Mapper in the ASML tool parlance exercises two degrees of control, *Unicom* and *Dosicom* [145], which respectively change dose profiles along the lens slit and the scan directions of the step-and-scan exposure tool. Figure 1.8 from [113] shows three methods of dose correction. Exposure dose can be changed per field to reduce AWLV (left) and also be changed in both x- and y-directions within a field to reduce ACLV (center and right).

The Zeiss/Pixer critical dimension control (CDC) technique [29] also enables adaptivity in the manufacturing flow to meet the required CD specifications. The CDC technology modifies the local mask transmissivity (which translates into local CD changes on the wafer during the lithography process) without removing the pellicle, thus allowing for CD manipulations either at the mask manufacturing site, or at the fab line [39]. When there exists CD variation as shown in Figure 1.9(a), CDC adds shading elements on a specific area of a mask to cause larger CD as shown in Figure 1.9(b).

The goal of such new equipments is to improve global CD uniformity. However, considering different timing criticality of transistors in a design, a uniform CD across a design is neither necessary nor optimal – timing-critical transistors may need smaller

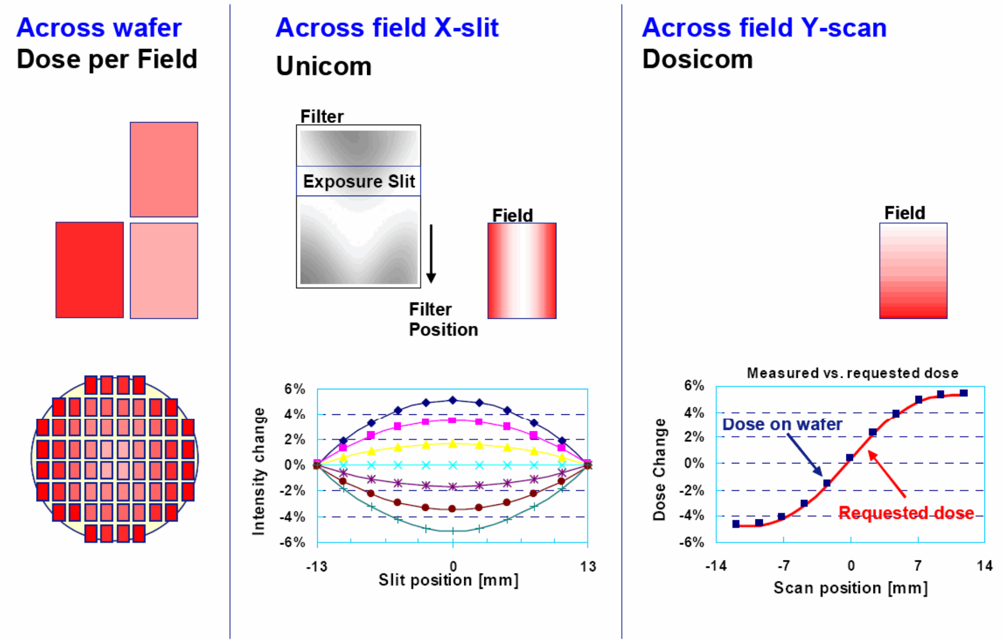


Figure 1.8: Three methods of dose correction using the Dose Mapper. Exposure dose can be changed per field to reduce AWLV (left) and can be changed within a field by Unicom (center) and Dosicom (right) to reduce ACLV. Figure reproduced from [113].

CD to increase speed, while non-timing critical transistors need larger CD to reduce leakage current. Further manufacturing optimization is possible if design information is reflected to those equipments. To this end, accurate and efficient variation modeling techniques as well as new design optimization techniques are essential.

New problems from new subwavelength lithography techniques. Projection optical lithography at $193nm$ with advanced RETs and immersion is expected to satisfy the needs of the $45nm$ node. However, for $32nm$ node patterning, the availability of options such as EUVL remains unclear. An EUV imaging system is composed of mirrors coated with multilayer structures designed to have high reflectivity at $13.5nm$ wavelength. There are many technical hurdles for implementing EUV lithography in terms of mask blank fabrication, high output power source, resist material, etc. In addition, although EUVL can successfully generate sub-$20nm$ patterns, economic cost must be considered in its adoption.

Double patterning lithography (DPL) partitions a critical-layer layout into two

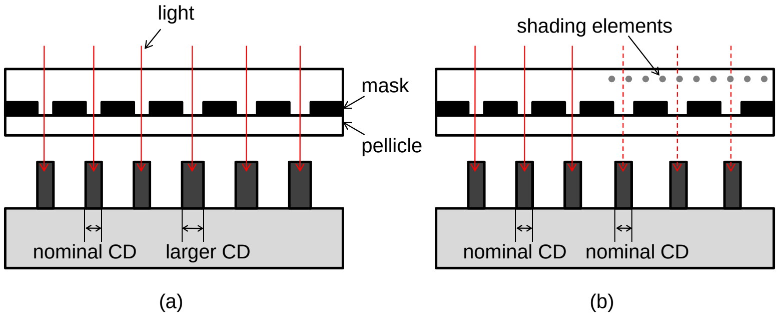


Figure 1.9: An illustration of the Zeiss/Pixer critical dimension control (CDC) technique. (a) CD variation with an initial mask. (b) CD correction by locally adding shading elements on the mask.

mask layouts, each with relaxed critical pitch and spacing. DPL provides an attractive alternative or a supplementary method to enable the $32nm$ and $22nm$ process nodes, relative to costlier technology options such as high refractive index materials, EUVL, or e-beam lithography. However, two lithography steps with overlay of two masks introduce additional variability in both FEOL and BEOL, as illustrated in Figure 1.10(a). In BEOL, overlay introduces additional linewidth and linespace variation, and results in capacitance variation. In FEOL, overlay results in two distinct distributions of gate CDs, and uncorrelated CD variations as shown in Figure 1.10(b); this introduces a new set of 'bimodal' challenges for timing analysis and optimization.

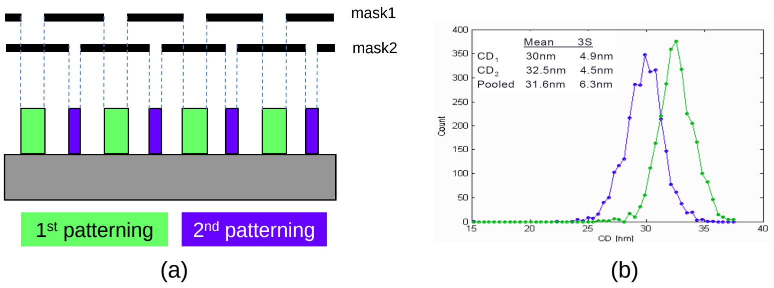


Figure 1.10: Two lithography steps result in bimodal CD distribution: (a) CD variation in DPL and (b) bimodal CD distribution (reproduced from [64]).

1.3 This Thesis

To mitigate the remaining or emerging variabilities, accurate modeling and assessment of the variabilities are essential through detailed analyses of underlying physical mechanisms. Appropriate optimizations in both design and manufacturing must be developed, based on comprehensive understanding of the benefits and costs of such additional measures. Figure 1.11 illustrates the scope and organization of this thesis to these ends.

This thesis first quantifies the impact of guardband reduction on design outcomes, as well as resulting yield and cost, to objectively evaluate the true benefits of various guardband reduction techniques. Cost-effective guardband reduction techniques are then presented for both design and manufacturing. The proposed measures span multiple stages of design, manufacturing and implementation: (1) from basic circuit elements such as device, interconnect, logic gates and memory bitcells, to high-level design implementation phases such as logic synthesis, placement and routing, and (2) from mask generation and lithography, to post-silicon variation measurement. The innovative techniques proposed in this thesis can be grouped into three main thrusts:

- Variability modeling and mapping,
- Variation assessment, and
- Variability mitigation.

In the **variability modeling and mapping** thrust, this thesis reviews various variation modeling techniques and proposes a novel variation mapping framework (based on compressed sensing theory) that reconstructs the details of multiple, simultaneously-occurring systematic variation maps from measurements of a small number of naturally-occurring timing paths within the design.

In the **variability assessment** thrust, this thesis provides quantified analyses of new interconnect and device variations that are emerging with advanced lithography techniques. For instance, for double patterning lithography (DPL), which is regarded as the most promising next-generation patterning technique for $20nm$ technology and below, this thesis develops variation analysis frameworks based on production signoff

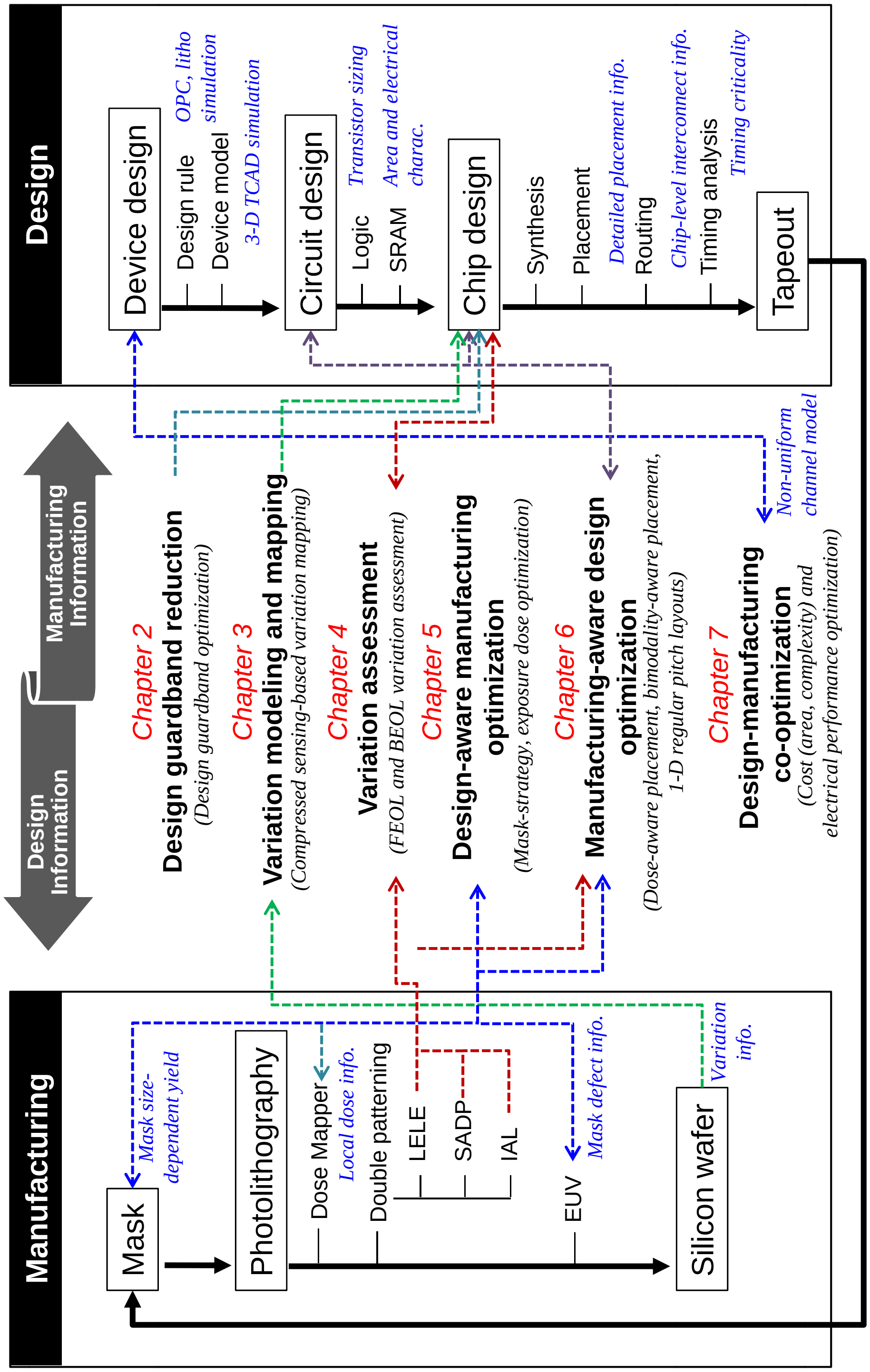

Manufacturing
Mask
Mask size-dependent yield
Photolithography
Dose Mapper
Local dose info.
Double patterning
LELE
SADP
IAL
EUV
Mask defect info.
Silicon wafer
Variation info.
Design Information
Manufacturing Information
Chapter 2
Design guardband reduction
(Design guardband optimization)
Chapter 3
Variation modeling and mapping
(Compressed sensing-based variation mapping)
Chapter 4
Variation assessment
(FEOL and BEOL variation assessment)
Chapter 5
Design-aware manufacturing optimization
(Mask-strategy, exposure dose optimization)
Chapter 6
Manufacturing-aware design optimization
(Dose-aware placement, bimodality-aware placement, 1-D regular pitch layouts)
Chapter 7
Design-manufacturing co-optimization
(Cost (area, complexity) and electrical performance optimization)
Non-uniform channel model
Design
Device design
Design rule
Device model
OPC, litho simulation
3-D TCAD simulation
Circuit design
Logic
SRAM
Transistor sizing
Area and electrical charac.
Chip design
Synthesis
Placement
Detailed placement info.
Routing
Chip-level interconnect info.
Timing analysis
Timing criticality
Tapeout

Figure 1.11: Scope and organization of this thesis.

tools and 3-D TCAD (“technology computer-aided design”) tools, considering all possible process options and scenarios. Exhaustive studies with the proposed frameworks, from a small representative interconnect structure to chip-level designs, afford new insights to designers and manufacturers regarding how to trade off quality of results versus design and manufacturing costs, across various double patterning process technology options. With respect to devices, interconnects, and electrical performance, the thesis gives both analytic and empirical assessments for the significance of ‘bimodal’ dimensional distributions that arise from the DPL approach.

Finally, in the **variability mitigation** thrust, this thesis presents three distinct approaches to explicitly mitigate variations and enable principled tradeoffs between design cost and yield. First, *design-aware manufacturing process optimizations* provide optimal mask strategies considering parametric and defect yields by integrating mask size-dependent variation and parametric yield models into a cost model that incorporates mask, wafer, and processing costs, along with throughput, yield, and manufacturing volume. This aspect of the thesis also analyzes impact of defects on parametric yield with understanding of design context (i.e., timing and electrical-functional criticality of each pattern in the layout design). This thesis also proposes design-aware local optimizations of exposure dose in the photolithography process, to improve timing yield of circuits as well as reduce leakage power.

Second, *manufacturing-aware design optimizations* include a cell swapping-based placement optimization algorithm that improves timing yield as well as reduces leakage power, in light of systematic variations of exposure dose in the manufacturing process. A new bimodal-aware timing analysis methodology is proposed in the context of double patterning lithography; this significantly reduces pessimism of traditional timing analysis approaches, and provides optimization techniques to improve timing yield of designs. The bimodal-related research also devises a novel metric to quantify the delay variation of timing paths due to bimodal distribution of pattern variations, and develops efficient, optimal cell-based timing-aware DPL mask assignment and detailed placement algorithms. Other work develops new 1-dimensional regular pitch SRAM (static random-access memory) bitcell layouts which are amenable to interference-assisted lithography (IAL). This part of the thesis devises required design rules for a $32nm$ 6-T bitcell, and designs a family of IAL-friendly bitcell layouts. The quality of the proposed bitcell layouts has been verified through lithography and circuit simulations.

Third, *design-manufacturing co-optimization* includes a first-ever elucidation of the tradeoff between electrical performance and manufacturing cost for modern transistor fabrication. This thesis introduces a novel shape-based (i.e., a general superellipse-based) transistor model, which includes (1) capacitance modeling of line-end extension and consequent current density changes in the transistor channel, and (2) on- and off-current modeling from the new capacitance model. The new transistor model enables fast evaluation of electrical characteristics of complicated post-lithography gate patterns. Then, through assessment of impacts of various layout design rules, mask design optimizations, and lithography process parameters on design area and electrical characteristics, this thesis derives simple rules of thumb for electrically safe and lithographically robust, yet cost-effective and area-conserving, transistor design rules.

The remainder of this thesis is organized as follows.

- In Chapter 2, to motivate the production impact of this thesis, we quantify the true benefit of variation reduction. We give detailed assessments on impact of guardband reduction with respect to a number of metrics of design productivity (iterations, CPU times in synthesis, clock tree synthesis (CTS) and place-and-route (P&R) phases, total design flow turnaround time (TAT), etc.), design closure (final timing fixes, etc.), and design quality (standard-cell area, routed wirelength, critical-path delay, dynamic and leakage power, etc.). We also quantify the true value of the guardband reduction in terms of design yield and the number of good dies per wafer.

- Chapter 3 proposes a novel variation mapping framework that reconstructs the details of multiple, simultaneously occurring systematic variation maps from measurements of natural timing paths in a design. Using the sparsity in DCT coefficients of a variation map, we formulate delay of natural timing paths as a function of the DCT coefficients, and find the DCT coefficients using a linear programming solver, using only a small number of measured delay values from natural timing paths. We also suggest potential useful applications of the proposed variation mapping technique: (1) modeling of multiple variation maps for the multiple IC layers in 3-D integration, and (2) decomposition of the effects of multiple variation sources on timing delay variation.

- Chapter 4 introduces various types of double patterning lithography (DPL) techniques that aim to reduce patterning variability, and discusses new challenges that arise from such advanced lithography techniques. We analyze impact of DPL on both front-end-of-line (FEOL) and back-end-of-line (BEOL) variabilities. For BEOL variability due to DPL, we provide a variational interconnect analysis framework, taking overlay into account. We apply the proposed framework to testcases ranging from a small representative interconnect structure (using 3-D TCAD-based tools) to chip-level designs (using golden extraction and timing analysis tools). For FEOL *bimodal CD distribution* variability due to DPL, we give both analytic and empirical assessments of the potential impact of DPL on timing analysis and guardbanding, and propose potential solutions for each step of the design process to mitigate impacts of this additional variability.

- Chapter 5 proposes two design-aware manufacturing process optimization techniques. First, we find optimal reticle strategies considering parametric and defect yields. We analyze CD variability with respect to reticle size, and quantify its impact on parametric yield. We integrate the parametric yield depending on the field size in a cost model that incorporates mask, wafer, and processing cost considering throughput, yield, and manufacturing volume. We then assess various reticle strategies (e.g., single-layer reticle (SLR), multiple-layer reticle (MLR), and small and large size) considering field-size dependent parametric yield. We also analyze defect-induced parametric yield in extreme ultraviolet lithography (EUVL), and show that sensitivity of parametric yield to defect parameters, i.e., defect density, height, distribution and influence distance. We then compare parametric yields of various reticle strategies. Second, we propose a novel method to improve the timing yield of as well as reduce total leakage power, using advanced manufacturing techniques, such as ASML Dose Mapper [1], and Zeiss/Pixer CDC [29]. We propose a design-aware dose map optimization, based on the fact that the exposure dose in the exposure field can change the lengths and widths of transistor gates in a circuit.

- Chapter 6 presents novel manufacturing-aware design optimization techniques for recent advanced manufacturing techniques. First, we propose a placement optimization technique that complements the *Dose Mapper* optimization discussed in Chapter 5. Second, we propose new bimodal-aware timing analysis and optimization methods to improve timing yield of standard cell-based designs that are manufactured using DPL. To mitigate the timing variability in double patterning, we introduce a new metric that quantifies the delay variation of timing paths, and implement an optimal cell-based timing-aware color assignment technique for double patterning that reduces both timing delay as well as timing variation. To address the increased coloring conflicts due to this intentional timing-aware coloring, we also propose a dynamic programming-based detailed placement algorithm that minimizes coloring conflicts by perturbing placement and exploiting whitespace in the given placement. Third, we present new 1-D regular pitch SRAM bitcell layouts which are amenable to *interference-assisted lithography* (IAL), which has been proposed as a low-cost maskless double patterning. We derive required design rules for a $32nm$ 6-T bitcell, and propose a family of IAL-friendly bitcell layouts. We confirm through lithography and circuit simulations that the proposed bitcell layouts can be successfully printed by IAL and that their electrical characteristics are comparable to those of existing bitcell layouts.

- Chapter 7 describes a design-manufacturing co-optimization approach. We propose a novel modeling framework which includes (1) capacitance modeling of a line-end extension and consequent current density changes in the device channel, and (2) I_{on} and I_{off} modeling from the new capacitance model. We define a new electrical metric for a line-end shape as the $expected$ change in I_{on} or I_{off} under a given overlay error distribution. We further apply a *superellipse* form to parameterize line-end shapes; we then use this to generate a large variety of line-end shapes. We evaluate the electrical metric on these line-end shapes to come up with simple rules of thumb that the lithographer can use to quickly evaluate the quality of a combined lithography and OPC solution with respect to line-end shaping. We also evaluate post-litho line-end shapes while varying OPC, lithography and design rule parameters, and find a tradeoff between cost and electrical characteristics.

Chapter 2

Impact of Guardband Reduction

In sub-90nm process technologies, there has been increased interest in design for manufacturability (DFM) techniques that address mounting variability and leakage power challenges. As we review below, several recent works attempt to 'close the loop' from systematic or deterministic variability sources (litho, etch and CMP) back to design analysis (SPICE models of devices and gates, RC extraction of interconnects, etc.). However, DFM tools and methodologies that bring process awareness into design analysis and optimization will be of limited interest to design teams unless the signoff design attributes (quality-of-result or QOR) and/or the design cycle (turnaround time or TAT) actually improve. In particular, design teams require promising financial return to go through the extra tool adoption, flow integration and design efforts that lead to more manufacturable tapeouts to the foundry. The challenge for the foundry and EDA sectors today is to collaboratively deliver opportunities for design-side customers to realize potential financial benefits in return for deploying DFM approaches. To this end, quantified return on investment (ROI) analyses are required.

Another motivation for this work comes from the semiconductor technology roadmapping (ITRS) [11] community, which spans lithography, process integration, front-end process, interconnect, and other technologies. In the ITRS effort, it has never been clear 'how much variability can design tolerate?' For example, the 2005 edition of the ITRS increased the lithography critical dimension (CD) 3-sigma tolerance from its historical 10% value up to 12%. While this relaxation of the ITRS CD control requirement enables continuation of the foundry process roadmap, it was obtained without any rigorous analysis of net impact on the extractable design value per wafer. Future balancing between process scaling and design technology 'equivalent scaling' on the Moore's Law roadmap must be guided by more quantitative analyses.

Today, in the $65nm$ and early $45nm$ nodes, particularly for high-performance process flavors, silicon providers are likely to consider providing variant guardbands at the level of device model or interconnect RCX models, corresponding to different regimes of manufacturing-friendliness or "DFM score" in the tapeout. The first example might be the reduction of *worst-case-best-case* (WC-BC) guardband for RC extraction, which is enabled by the deployment of new golden models for *chemical-mechanical planarization* (CMP), and which lead to new process-aware extraction and timing analysis (as well as process-driven dummy fill) flows. The second example might be the application of a different (narrower) SPICE model guardband for, e.g., a multi-fingered device that is laid out with optimal (restricted) pitch and poly dummy layout choices.

With respect to the preceding discussions and examples, significant overheads to the silicon provider are associated with this nascent paradigm shift in the foundry-designer business model. Among these overheads are commitment to additional model-to-silicon fidelity constraints, increased process technology characterization effort, and opening up of another dimension of competition with other foundries. Yet, the benefits to the foundry are clear: incentive for design customers and EDA partners to 'do the right thing' for the manufacturing process, and the opportunity to offer differentiated value to customers. Clearly, a missing element for the concept of layout-specific design guardbanding to go forward is a *framework* to quantify the impact of guardband change on design QOR and TAT. The work in this chapter seeks to fill this gap.

In this chapter, we develop an experimental framework, and then experimentally quantify the impact of model guardband reduction on outcomes of the synthesis, place and route (SP&R) implementation flow. We make the following contributions.

- We study small open-source standard-cell cores in $90nm$ and $65nm$ foundry technologies (ARM/TSMC) as well as an *industrial* embedded processor core implemented in $45nm$, and separately evaluate the impacts of guardband reductions in the FEOL (Liberty timing models) and in the BEOL (RCX in golden extraction such as with STAR-RCXT).

- We assess impact of guardband reduction with respect to three metrics: (1) design productivity (iterations, CPU times in synthesis, CTS and P&R phases, total design flow TAT, etc.), (2) design closure (final timing fixes, etc.), and (3) design quality (standard-cell area, routed wirelength, critical-path delay, dynamic and leakage power, etc.).

- We observe that the value of guardband reduction can be very significant. For example, we find that the 40% guardband reduction obtained by [75] with a 'iso-dense' variational timing analysis methodology leads to typical reductions of 13% in standard-cell area, 12% in routed wirelength, 13% in dynamic power, 19% in leakage power and 28% in SP&R turnaround time for open-source designs in both $90nm$ and $65nm$. We also observe reductions of 8% in standard-cell area, 7% in routed wirelength, 5% in dynamic power, and 10% in leakage power for the embedded processor core in $45nm$ at 30% guardband reduction.

- We decompose each separate impact of P, V, and T on delay. We observe that each axis of PVT has different delay impact. If any of P, V, and T are fixed for reasons such as test specifications (low V_{cc} margin) or customer requests, it will limit the guardband reduction.

- We quantify the impact of the guardband reduction on design yield. Our analysis shows up to 4% increase in the number of good dies per wafer with 27% guardband reduction. However, we notice a reduction in the number of good dies per wafer after 40% guardband reduction.

2.1 Related Literature

We are not aware of any previous literature that quantifies the impact of guardband reduction in a modern IC implementation flow. However, we note two related literature that respectively addresses (1) the taxonomies of variation sources and guardbanding in the modeling and analysis chain and (2) systematic process variation-aware design analyses.

Taxonomies of Variation Sources and Guardbanding. It is well-understood that variation can arise from environmental parameters (temperature, supply voltage, etc.), manufacturing processes that lead to device and interconnect changes, and reliability effects (hot-carrier degradation, NBTI, etc.). Scheffer [154, 155] gives a taxonomy of uncertainty and variation sources, with emphasis on the back end of the line (BEOL), i.e., the interconnect stack. This is in a similar spirit to the work of Nassif [136], which reviews the sources and the impact of parameter variability across inter-die and intra-die sources. While these works taxonomize and quantify individual variation sources, they do not make any connections back to the quantified impact within the chip implementation flow.

Systematic Process Variation-Aware Design Analyses. Prediction and compensation of systematic variations has traditionally been done by the manufacturing process, with only simple guardbanded abstractions (e.g. design rules) being passed on to the designers. However, the increasing magnitude and 2-D pattern dependence of these variations, their impact on design metrics, and the inability of manufacturing equipment and process techniques to fully mitigate them, are causes for serious concern in sub-100nm technologies. If modeling and design guardbands used for timing and power signoff include compensatable systematic variations, the result is overdesign and a more difficult design closure task. With this in mind, a number of recent works have proposed systematic process variation-aware design analyses to 'close the loop' from manufacturing simulation back to the design flow.

Balasinski et al. [37] propose a methodology of manufacturability qualification for ultra-deep submicron circuits, based on an optical simulation of the layout which is integrated with device simulation; see also [164]. Pack et al. [143] propose to incorporate advanced models of lithographic printing effects into the design flow to improve yield and performance verification accuracy. Gupta et al. [82] observe that lithography simulation permits post-OPC (optical proximity correction) estimation of on-silicon feature sizes at different process conditions. Yang et al. [192] address post-lithography based analysis and optimization, proposing a timing analysis flow based on residual

OPC errors (equivalent to lithography simulation output) for timing-critical cells and their layout neighborhoods. Cao et al. [49] propose a methodology for standard-cell characterization considering litho-induced systematic variations. In [49], the objective is to enable efficient post-litho analysis by running litho-aware characterization. Furthermore, to minimize the difference between isolated and actual placement contexts of a given standard cell, vertical dummy poly patterns are inserted at the cell boundary. Finally, it is noteworthy that Gupta and Heng [75] perform "iso-dense aware" timing analysis (based on the modeling of systematic through-focus Leff variation) to achieve up to 40% reduction of the BC-WC guardband in static timing analysis. Also, Sylvester et al. [174] observe that up to 60% of BEOL guardband can be eliminated by using a realistic BEOL variation model.

Despite such vigorous research activities in this arena, some fundamental questions remain open. For example, what is the impact of the guardband on design quality? And, what is the specific return that we can expect to be realized by the design team from availability of, e.g., iso-dense aware timing analysis [75], post-lithography based analysis and optimization [192], or any other potential path to reduced guardband? The following sections describe the efforts toward a quantified answer to these questions.

2.2 Model Guardband Reduction

2.2.1 Impact of PVT on Circuit Delay

To quantify the impact of guardband reduction on design process outcomes, we first quantify the existing guardband in foundry delay models. Guardband exists in the form of delay for each process, voltage, and temperature (PVT) corner (i.e., delay tables in Liberty model files). Since each axis of PVT will have a different delay impact, we quantify the impact due to each P, V, and T corner separately.

Standard cells. To assess the impact of each P, V, and T parameter on standard cell delay, we run SPICE simulations for a simple inverter cell across 8 possible combinations of PVT, i.e., $\{P_{slow}, P_{fast}\} \times \{V_{low}, V_{high}\} \times \{T_{low}, T_{high}\}$.[1] Table 2.1 shows the

[1] Slew (39.2ps) and load capacitance (4.9fF) values are selected from the third row and third column entry of the 7 × 7 Liberty delay table for a $65nm$ technology.

delay values of $65nm$ inverter cell for all PVT combinations. Of the $1.8\times$ difference between worst-/best-case PVT corners, $1.46\times$ is from process, $1.25\times$ is from voltage and $0.97\times$ is from temperature (i.e., due to a reverse temperature effect).

Table 2.1: Inverter delay for different P, V and T corners.

Process		Voltage (V)	Temperature (^oC)	Delay (ps)
NMOS	PMOS			
Fast	Fast	1.0	-40	22.17
Fast	Fast	1.0	125	22.54
Fast	Fast	0.9	-40	27.21
Fast	Fast	0.9	125	26.16
Slow	Slow	1.0	-40	31.44
Slow	Slow	1.0	125	30.63
Slow	Slow	0.9	-40	42.78
Slow	Slow	0.9	125	38.89

If any of the P, V, and T parameters are fixed for reasons such as test specifications or customer requests, this will limit the actual achievable guardband reduction. Figure 2.1 shows worst-/best-case delay changes *with only* process guardband reduction. To determine this, we perform a set of fine-grained SPICE simulations with fixed V and T. We create 100 SPICE models by interpolating between FF (fast NMOS and fast PMOS) and SS (slow NMOS and slow PMOS) models with step size of 1% (i.e., corresponding to 1% guardband reduction).

We then measure rise and fall delays of four standard cells including an inverter cell (INV), a 2-input NAND gate (NAND2), a 2-input AND gate (AND2), and a 4-input AND-OR gate (AO22), using the corresponding interpolated SPICE models. Figure 2.1 shows normalized worst- and best-case delay values of the above cells. We take average of rise and fall delays, and normalize the worst- and best-case delays of each cell to the delay value of the cell at the original best-case process corner (i.e., 0% RGB), respectively. We observe that delay at worst-case (best-case) decreases (increases) with reducing process guardband. We observe that the decreasing (increasing) rate of delay

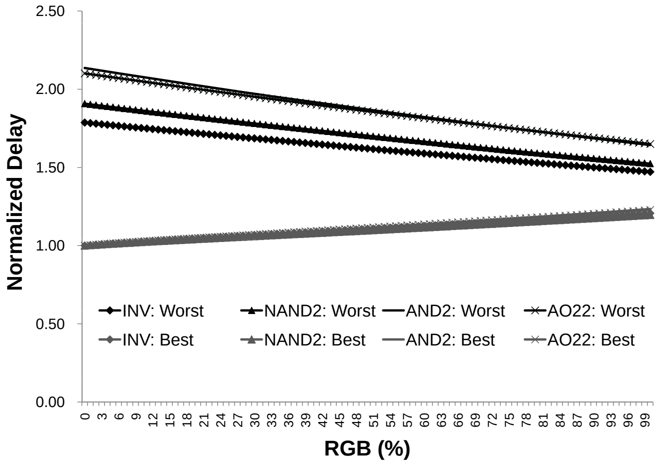


Figure 2.1: Worst-/best-case delay changes of an inverter (INV), a 2-input AND gate (AND2), a 2-input NAND gate (NAND2), and a 2-input AND-OR gate (AO22) versus the process guardband.

change does not have a significant relationship with the functional complexity of the cell. At 100% guardband reduction, FF and SS have the same SPICE model. Hence, the delay difference in Figure 2.1 is due only to temperature and voltage guardband.

Also, Figure 2.2 shows the delay change percentage, for worst- and best-case corners, of the above four cells when the process guardband reduces from 0% to 100%. We observe that the worst-case delay change of complex cells, e.g., AO22, is larger than that of an inverter. The best-case delay change of a NAND2 is the smallest among the four cells and is within 1.07% of that of the inverter.

Memory cells. Since SRAM occupies a significant portion of today's SOC designs, we also assess the impact of guardband reduction on SRAM performance. A 6-T SRAM bitcell is composed of 6 transistors, 2 bitlines (BL and BLb) and one wordline (WL). A bit of data is stored in the complementary internal nodes nl and nr, when WL is '1'. The transistors are classified as pass (or access) transistors ($C1$ and $C2$), pull-down transistors ($B1$ and $B2$), and pull-up transistors ($A1$ and $A2$) as shown in Figure 2.3.

During a read operation, one of the pre-charged bitlines is discharged through a pass transistor and its associated pull-down transistor (i.e., $C1$-$B1$ or $C2$-$B2$), and the sense amplifier detects the voltage difference between the two bitlines. I_{cell} is the maximum current that flows during the read operation, and can be used as an SRAM

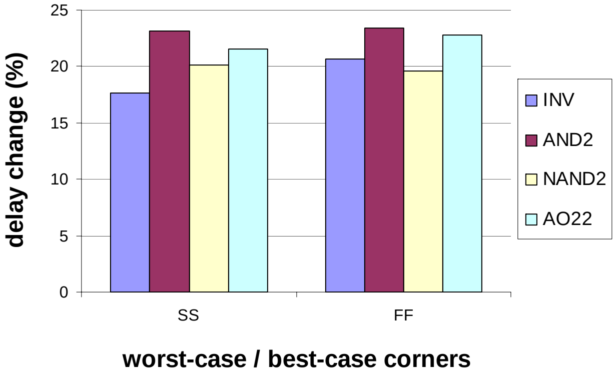


Figure 2.2: Worst-/best-case delay change percentage across 0%-100% RGB.

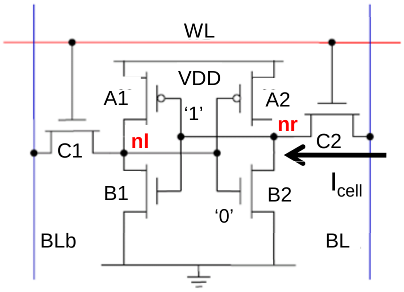


Figure 2.3: Schematic circuit diagram for a 6-T SRAM bitcell.

performance metric. We measure worst-/best-case I_{cell} of a $65nm$ SRAM bitcell with the interpolated 100 SPICE models used for inverter delay simulation. Figure 2.4 shows worst-/best-case I_{cell} changes *with only* process guardband reduction. According to the figure, best-case (worst-case) I_{cell} decreases (increases) with reducing process guardband. Since I_{cell} is inversely proportional to SRAM delay, the increase of I_{cell} at the worst-case corner decreases SRAM delay. However, the performance of SRAM depends not only on the I_{cell}, but also on the sense amplifier's reaction speed and digital logic signal propagation speed in the peripherals of SRAM. Figure 2.5 shows the normalized delay of an SRAM bitcell, which is derived from I_{cell} simulation results, and the normalized delay of an inverter. We observe that the delay of an SRAM bitcell is more sensitive to the guardband reduction than that of an inverter. Hence, we can conclude that the logic delay improvement from the worst-case guardband reduction can speed up both standard logic cells and embedded SRAMs.

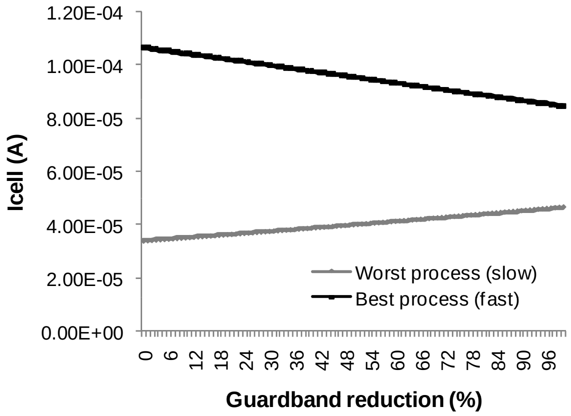


Figure 2.4: Worst-/best-case I_{cell} changes of a $65nm$ SRAM bitcell versus process guardband.

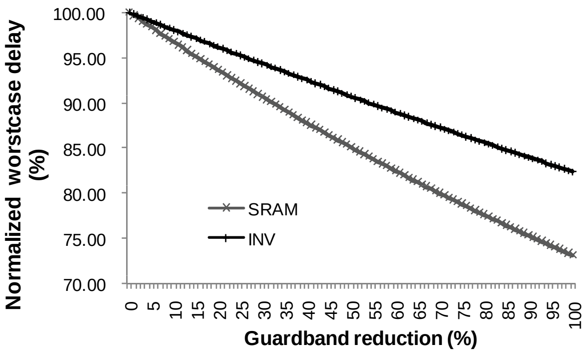


Figure 2.5: Normalized worst-case delay of $65nm$ inverter (INV) and SRAM bitcell versus process guardband.

2.2.2 Liberty Model Scaling

In corner-based design and signoff methodologies, there are best-case and worst-case design behaviors for which cells are characterized, and which are captured in respective Liberty (.lib) format libraries [24]. In the Liberty format, each standard cell *master* has several attributes, such as pin type, loads, stimuli and lookup-table indices. The data available in the Liberty format include capacitance, thresholds/switching points, rise time, fall time, and power values of each cell in the library. Static timing analysis operates independently of characterization, reading both a Verilog netlist and multiple timing libraries. To use the delay changes from the guardband reduction, new characterization must be performed for each guardband value. However, the cell characterization process is very time-consuming. Instead, for SP&R, we can directly reduce the delay guardband by linear scaling of timing libraries, since delay varies linearly with guardband reduction as shown in Figures 2.1 and 2.5. In the experiments, we run through a traditional timing-driven SP&R flow; hence, we scale only the input pin capacitances and timing tables, and we do not modify the power tables of the .lib files.

It is well-known that one can specify PVT scaling factors in the technology library environment, using so-called k-factors. These k-factors (so-called because they are attributes with names starting with $k\_$) are multipliers that scale defined library values, allowing consideration of the effects of changes in process, voltage and temperature [24]. However, in the used methodology, we do not use k-factors since they cannot correctly capture guardband reduction. Instead, we apply an entry-by-entry library scaling methodology in which (1) the difference between values of a certain table entry in two libraries (e.g., worst-case and best-case) is computed, and then (2) the amount of required guardband reduction is applied to this difference and the corresponding (e.g., best- and worst-case) table values are modified accordingly.

Figure 2.6 illustrates the steps required to scale timing tables within the Liberty files.

- *Goal: Entry-by-entry BC-WC guardband reduction.* Figure 2.6(a) shows an example of timing tables within best- and worst-case Liberty files.[2] We seek a uniform percentage of guardband reduction to each entry-by-entry difference (i.e., the amount of guardband associated with each delay value) between best-case and worst-case delay values, which are characterized under the corresponding PVT conditions.[3] Note that we cannot simply reduce values of worst-case delays, and increase values of best-case delays, by fixed percentages; this will not result in a uniform guardband reduction.

- *Index matching step.* In a production timing library, it is common for, e.g., the input slew time indices of the best-case library to be different from the indices

[2] The tables shown in Figure 2.6 are for illustrative purposes. Neither their indices nor their entries represent realistic values.

[3] PVT condition for best (worst) case is fast (slow) transistors, high (low) supply voltage and low (high) temperature.

of the worst-case library. Hence, before we can scale entry-by-entry guardband values, we must first match up the indices of corresponding tables in the best-case and worst-case libraries. We achieve this by interpolation/extrapolation from the original index values of both tables, as illustrated by the "index-matched best-case" table in Figure 2.6(b).

- *Calculation of entry-by-entry guardband reduction.* After unifying the library table indices, we can compute the entry-by-entry difference (i.e., original amount of guardband) and apply the necessary guardband reduction. For example, in Figure 2.6(b), we see that for input slew time = 2 and capacitive load = 1, the best-case and worst-case delay values are 2 and 4, respectively. To reduce the guardband by 10%, we first find the difference between corresponding values (i.e., 4 - 2 = 2). Then, we add 5% of this difference to the best-case value, and subtract 5% of this difference from the worst-case value. The resulting guardband-reduced BC/WC values are seen in Figure 2.6(c). We more formally describe the index-matching and guardband reduction procedures in Figures 2.7 and 2.8.[4]
- *Scaling of pin capacitance guardband.* Note that input pin capacitance values can be considered as 1×1 tables. Hence, the same guardband reduction methods are applied to them as well.

2.2.3 Interconnect Model Scaling

It is commonly accepted that interconnect has become a dominant factor in determining circuit performance. In sub-100nm processes, litho- and CMP-induced variations in conductor width, conductor thickness, and inter-layer dielectric (ILD) height within the BEOL stack can cause significant variation of interconnect parasitics.

In the corner-based design methodology, extreme values of resistance and capacitance are used to obtain worst-case and best-case corners in timing analysis. For example, in best-case analysis we use the smallest capacitance value, and in worst-case

[4] In Figure 2.8, the factor 1/200 arises because half of the $x\%$ guardband reduction is applied to each of the best-case and worst-case values.

load slew	1	2	3	4
1	1	1	1	1
2	**2**	2	2	2
3	3	3	3	3
4	4	4	4	4

load slew	1	2	3	4
2	**4**	4	4	4
3	6	6	6	6
4	8	8	8	8
5	10	10	10	10

original best-case (a) original worst-case

load slew	1	2	3	4
2	2	2	2	2
3	**3**	3	3	3
4	4	4	4	4
5	5	5	5	5

load slew	1	2	3	4
2	**4**	4	4	4
3	6	6	6	6
4	8	8	8	8
5	10	10	10	10

index-matched best-case (b) original worst-case

load slew	1	2	3	4
2	**2.1**	2.1	2.1	2.1
3	3.15	3.15	3.15	3.15
4	4.2	4.2	4.2	4.2
5	5.25	5.25	5.25	5.25

load slew	1	2	3	4
2	**3.9**	3.9	3.9	3.9
3	5.85	5.85	5.85	5.85
4	7.8	7.8	7.8	7.8
5	9.75	9.75	9.75	9.75

reduced best-case (c) reduced worst-case

Figure 2.6: Illustration of steps in guardband reduction for timing tables of the Liberty (.lib) files. (a) Original best-/worst-case tables. (b) New best-case table with input slew time indices matched up with those of the worst-case table. (c) 10% guardband reduction, computed on an entry-by-entry basis, across all the table entries.

analysis we use the largest capacitance value. Resistance behaves inversely to capacitance, hence minimum resistance is used in worst-case analysis and maximum resistance is used in best-case analysis. In addition to process variations, operating conditions such as temperature affect resistance and capacitance values. In 90nm copper technology, large temperature variation (e.g., from -40^{o}C to 125^{o}C) can lead to 50% increases in resistance. From Table 2.2, which includes the process and temperature effects, we see that at the worst interconnect corner, the values of capacitance and resistance are greater than those at the best interconnect corner by 17% and 13%, respectively.

We implement a model guardband reduction for interconnect resistance and capacitance as follows.

```
Input: best-/worst-case libraries.
Output: index-matched best-case library.

for all the cells in the best-case library:
    find the corresponding cell in the worst-case library.
    interpolate/extrapolate the new best-case table entries using the best-/worst-case values.
    copy the slew rate index of the worst-case table on to that of the best-case table.
```

Figure 2.7: Index matching procedure.

```
Input: index-matched best-/worst-case libraries and $x\%$ guardband reduction.
Output: guardband reduced best-/worst-case libraries.

for all the common cells in the best-/worst-case libraries:
    for each entry in a best-case table ($value_{best}$):
        $value_{best} = value_{best} + \frac{x}{200}(value_{worst} - value_{best})$.
    for each entry in a worst-case table ($value_{worst}$):
        $value_{worst} = value_{worst} - \frac{x}{200}(value_{worst} - value_{best})$.
```

Figure 2.8: Guardband reduction procedure.

- We first extract resistance and capacitance from a sample design for best and worst corners using a signoff extractor (Synopsys Star-RCXT).
- We compare the mean of the worst-corner values with that of the best-corner values.
- Finally, for a given percentage reduction in guardband, we find proper scaling factors for each corner by a method similar to that described above for Liberty scaling. The scaling equations and the relative values of interconnect capacitance and resistance for $90nm$ technology are summarized in the Table 2.2.[5]

[5] Note that since the P&R tool (*Cadence SOC Encounter* [7]) and the signoff extraction tool (*Synopsys Star-RCXT* [28]) have discrepancies in their computed interconnect resistance and capacitance values, we compute separate scaling factors for each. Analogous scaling factors are separately computed for P&R and signoff extraction in the $65nm$ technology.

Table 2.2: R and C comparison and scaling method for $90nm$ interconnect.

		Best corner	Worst corner
P&R	R ratio	1	1.11
	R scaling factor for RGB($x\%$)	$1 + \frac{x}{200} \cdot (1.11 - 1)$	$1 - (1 - \frac{1}{1.11}) \cdot \frac{x}{200}$
	C ratio	1	1.15
	C scaling factor for RGB($x\%$)	$1 + \frac{x}{200} \cdot (1.15 - 1)$	$1 - (1 - \frac{1}{1.15}) \cdot \frac{x}{200}$
Signoff	R ratio	1	1.13
	R scaling factor for RGB($x\%$)	$1 + \frac{x}{200} \cdot (1.13 - 1)$	$1 - (1 - \frac{1}{1.13}) \cdot \frac{x}{200}$
	C ratio	1	1.17
	C scaling factor for RGB($x\%$)	$1 + \frac{x}{200} \cdot (1.17 - 1)$	$1 - (1 - \frac{1}{1.17}) \cdot \frac{x}{200}$

2.3 Analysis Flow and Testcases

2.3.1 Timing-Driven Implementation Flow

Figure 2.9 shows the traditional SP&R flow that we have scripted for "push-button" use in the experiments. The steps in Figure 2.9 represent the major physical design steps. At each step, we require that the design must meet the timing requirements before it can pass on to the next step. (This is standard practice, since the later in the design flow, the harder it is to fix a given timing violation.) In other words, in the event of any timing violation, the implementation flow goes back to the previous step through a return path and fixes the violation.

In the flow, we first synthesize RTL netlists with worst-corner libraries. This synthesis step, when different reduced-guardband libraries are used, produces initial netlists with different total standard-cell area. We use a fixed utilization ratio in all testcases at the floorplan stage. We optimize timing inside the P&R tool using its embedded RCX and delay calculation engines. Since the designer's concern is generally to obtain the best performance within given environments and constraints, we concentrate on fixing any setup violations at this stage of the implementation flow. Once all setup violations are cleared, it is necessary to fix any hold violations using the best-case library. While attempting to fix the hold violations, sometimes new setup violations are created, and iteration over the above steps is required until all violations are cleared at both the best and worst timing corners.

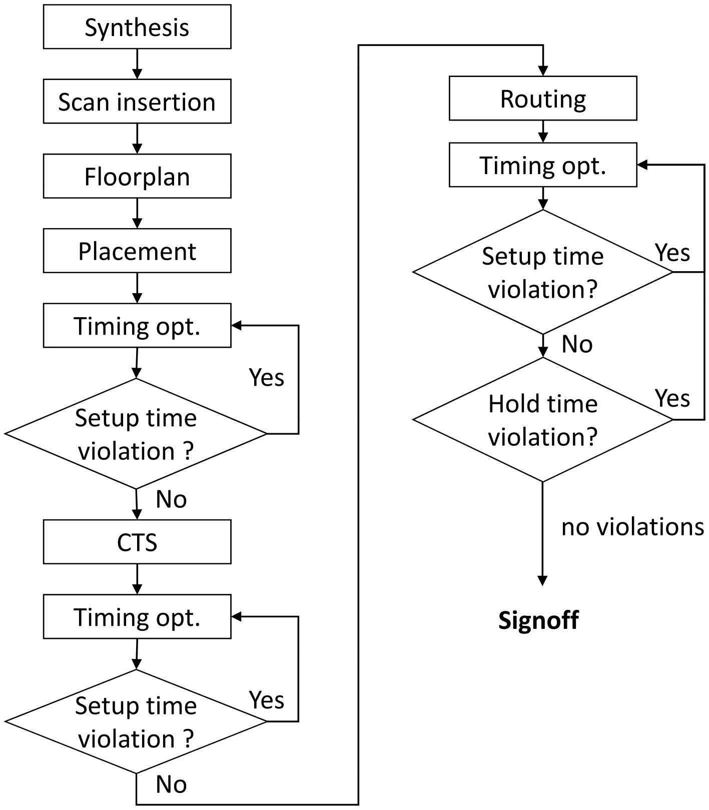


Figure 2.9: Implementation (synthesis, placement and routing) flow.

2.3.2 Testcases and Tools

We use four benchmark designs in the experiments. The first two are the AES and $JPEG$ cores, obtained as RTL from the open-source site $opencores.org$ [15]. The third testcase is $5XJPEG$, which is composed of 5 copies of the $JPEG$ core. The fourth is an embedded processor core provided by Qualcomm, Inc. [18]. For the first three testcases we perform experiments using front-end libraries in TSMC 90nm and 65nm technologies. For the fourth testcase we use 45nm libraries obtained from a foundry. The AES core typically synthesizes to approximately 16K instances; target clock frequency is 400MHz in 90nm and 600MHz in 65nm. The $JPEG$ (resp.

$5XJPEG$) core typically synthesizes to approximately 64K (resp. 320K) instances; target clock frequency is 300MHz in 90nm and 500MHz in 65nm. The embedded processor has approximately 67K instances; target frequency is 500MHz in 45nm. We use *Cadence RTL Compiler v05.20-s009_1* [6] to synthesize the open-source designs and use *Synopsys Design Compiler v2007.12-SP4* [21] to synthesize the embedded processor. We use *Cadence SOC Encounter v5.2* and *Cadence SOC Encounter v7.1 usr2* [7] to execute the P&R flow on open-source and embedded processor testcases, respectively. Initial row utilizations are 40%, 60%, 60% and 65% for the AES, $JPEG$, $5XJPEG$ and embedded processor designs, respectively. Note that final row utilizations may change depending on timing optimization steps (e.g., buffering, sizing, etc.) that are executed during the P&R flow. We use *Synopsys Design Compiler v2006.06-SP3* [21] for scan insertion and *Synopsys Star-RCXT v2006.06-SP1* [28] for RCX. Finally, *Synopsys PrimeTime v2005.12-SP3* [25] is used for static timing analysis.

2.4 Analysis on the Impact of Guardband Reduction

In the experiments for AES, $JPEG$, and $5XJPEG$ testcases, we run the entire implementation flow with six sets of libraries corresponding to model guardband reductions of 0%, 10%, 20%, 30%, 40% and 50%. We do this for each of three cases: (1) only back-end-of-line (BEOL) guardband reduction, (2) only front-end-of-line (FEOL) guardband reduction, and (3) both BEOL and FEOL guardband reduction – in order to separately observe the impact of FEOL and BEOL guardband reduction. Last, we do this for each of 90nm and 65nm technologies. As a result, each testcase is implemented with the scripted flow of Figure 2.9 for a total of 6 $\times$ 3 $\times$ 2 = 36 times, 18 times in each technology. However, for the embedded processor testcase, we only consider Case (2), hence we only implement the testcase 6 times in 45nm.

In the following sections, we use “F” or “FE” as shorthand for FEOL; “B” and “BE” are shorthand for BEOL. We also give detailed tables of numerical data for the 90nm $JPEG$ core implemented with 300MHz target frequency and for the 45nm embedded processor core with 500MHz target frequency. Other results are presented more compactly in graphical form.

2.4.1 Impact on Quality of Results

We assess impact of guardband reduction with respect to design quality metrics of area, routed wirelength, and dynamic and leakage power. Table 2.3 shows the impact of guardband reduction on the area (i.e., the sum of all standard cell areas within the design) for the $90nm$ $JPEG$ core implemented with $300MHz$ target frequency. Table 2.4 shows the impact of guardband reduction on total wirelength. For power estimation, we consider two different scenarios: (1) foundries reduce the guardband through process enhancement or (2) foundries simply reduce their guardband without process enhancement or changing operating condition. Tables 2.5 and 2.6 show the impact of guardband reduction on dynamic and leakage power, respectively, for Scenario (2). We note that the power reduction comes from the reduced area. Power values, especially leakage power, cannot be obtained by linear interpolations/extrapolations as used in delay scaling. Although we did not re-characterize cell power, we expect worst-case power to increase and best-case power to decrease since power and delay typically change in opposite directions. We also expect that power reduction from the area reduction will still be valid for Scenario (1).

Figures 2.10, 2.11, 2.12 and 2.13 show the impact of guardband reduction of both FE and BE on area, routed wirelength, dynamic power and leakage power for AES, $JPEG$ and $5XJPEG$ designs using $90nm$ and $65nm$ technologies, respectively. We observe that the area, wirelength, power metrics are "well-behaved"; they improve (decrease) as the percentage of the guardband reduction increases. At the level of 40% guardband reduction achieved by the variational timing approach from IBM [75], reductions of nearly 18% area, over 21% wirelength, 20% dynamic power and 29% leakage power are achieved, on average.[6] Somewhat surprisingly, guardband reduction for interconnect (i.e., BEOL) parasitics has much less impact on design quality than guardband reduction for FEOL models. In addition, Tables 2.7, 2.8, and 2.9 show the impact of guardband reduction with respect to area, and dynamic and leakage power,

[6] In [101], Kahng and Mantik observed the existence of 'inherent noise' in IC implementation tools, and documented up to 12% change in quality of result (e.g., total post-route wirelength) due to the tools' sensitivity to such noise sources as input renaming, randomization, scaling, etc. We note this previous work because it implies a limit to cleanliness of experimental data as we trace the impact of guardband reduction through the tool flow. Also, inherent tool noise may swamp any benefits of guardband reduction in certain design regimes (e.g., with respect to tightness or looseness of timing and area constraints).

for the $45nm$ embedded processor core. We observe that at 30% guardband reduction, area, dynamic power, and leakage power reduce by 8%, 5% and 10%, respectively.

Table 2.3: Area versus guardband reduction for $90nm$ $JPEG$ at $300MHz$.

Area	0%		10%		40%		50%	
	mm^2	$\%$	mm^2	$\%$	mm^2	$\%$	mm^2	$\%$
F	0.367	100	0.356	97.0	0.339	92.3	0.331	90.3
B	0.367	100	0.367	100.1	0.357	97.5	0.355	96.7
F+B	0.367	100	0.355	96.9	0.339	92.4	0.331	90.3

Table 2.4: Total wirelength versus guardband reduction for $90nm$ $JPEG$ design at $300MHz$.

WL	0%		10%		40%		50%	
	mm	$\%$	mm	$\%$	mm	$\%$	mm	$\%$
F	1609.2	100	1608.6	99.9	1544.3	96.0	1512.2	94.0
B	1609.2	100	1617.2	100.5	1586.6	98.6	1590.1	98.8
F+B	1609.2	100	1593.3	99.0	1539.0	95.6	1514.8	94.1

Table 2.5: Dynamic power versus guardband reduction for $90nm$ $JPEG$ design at $300MHz$.

P_{dyn}	0%		10%		40%		50%	
	mW	$\%$	mW	$\%$	mW	$\%$	mW	$\%$
F	114.1	100	102.8	90.02	93.6	82.00	91.5	80.19
B	114.1	100	111.2	97.46	106.1	93.01	107.5	94.21
F+B	114.1	100	101.0	88.52	94.5	82.81	90.8	79.59

Table 2.6: Leakage power versus guardband reduction for 90nm $JPEG$ design at 300MHz.

	0%		10%		40%		50%	
P_{leak}	mW	%	mW	%	mW	%	mW	%
F	0.250	100	0.210	84.00	0.175	69.89	0.167	66.56
B	0.250	100	0.243	97.04	0.226	90.15	0.229	91.88
F+B	0.250	100	0.204	81.61	0.178	71.06	0.166	66.50

Table 2.7: Area versus guardband reduction for 45nm embedded processor core at 500MHz.

	0%		10%		30%		50%	
Area	mm^2	%	mm^2	%	mm^2	%	mm^2	%
F	0.175	100	0.174	99.48	0.163	92.81	0.155	88.79

Table 2.8: Dynamic power versus guardband reduction for 45nm embedded processor core at 500MHz.

	0%		10%		30%		50%	
P_{dyn}	μW	%	μW	%	μW	%	μW	%
F	112.29	100	110.12	98.07	107.58	95.81	100.98	89.93

Table 2.9: Leakage power versus guardband reduction for 45nm embedded processor core at 500MHz.

	0%		10%		30%		50%	
P_{leak}	μW	%	μW	%	μW	%	μW	%
F	2.063	100	2.064	100.05	1.867	90.50	1.685	81.68

2.4.2 Impact on Critical Paths

It is instructive to look more closely at the effect of guardband reduction on timing modeling and analysis. Table 2.10 shows the average cell delays in a critical path

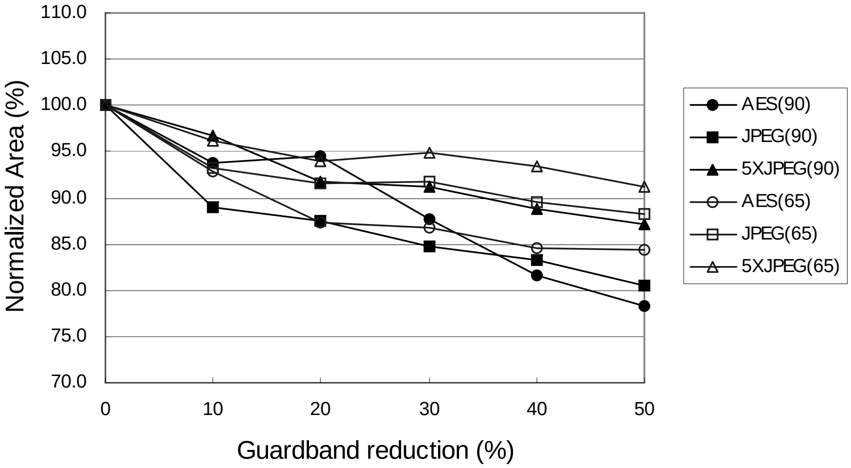


Figure 2.10: Area versus guardband reduction.

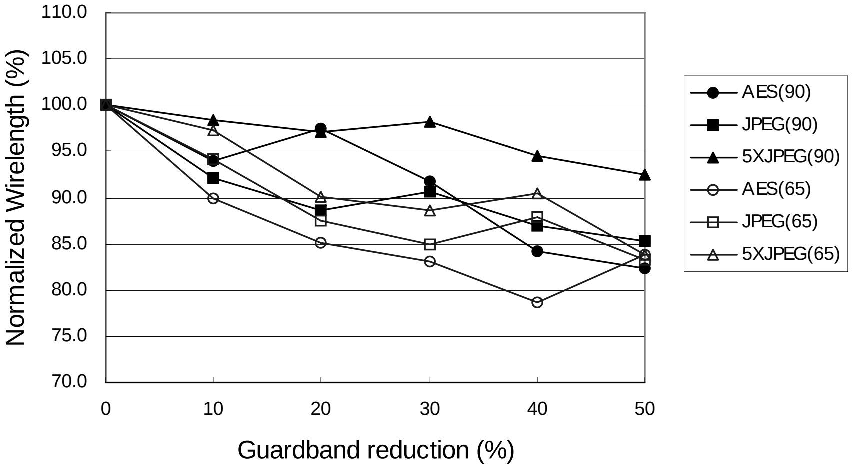


Figure 2.11: Total wirelength versus guardband reduction.

of the $90nm$ $JPEG$ implementation, for both best-case and worst-case corners, across different guardband reductions. We see that a 10% reduction in guardband increases (decreases) the best (worst) average stage delay by only $4ps$ (3% of the average stage delay). Also, the delay differences across different guardband reductions in the BEOL are very small compared to the differences in the FEOL. Possibly, the impact of BEOL guardband reduction (despite being expected and evident from the resulting data) will not always be visible due to inherent noise in EDA implementation tools [101]. The

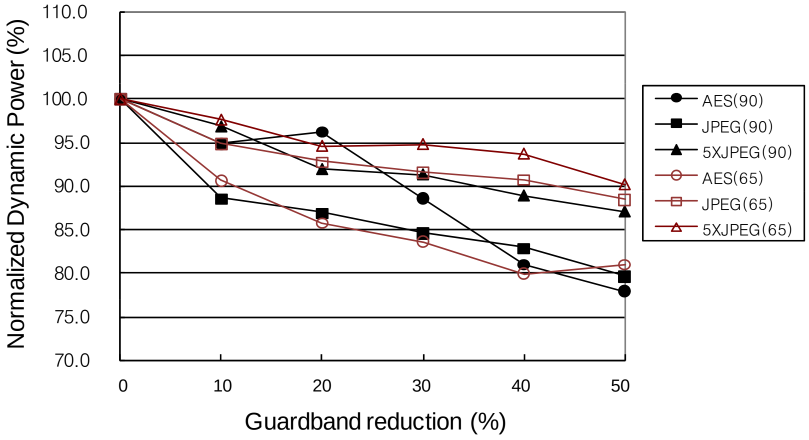


Figure 2.12: Total dynamic power versus guardband reduction.

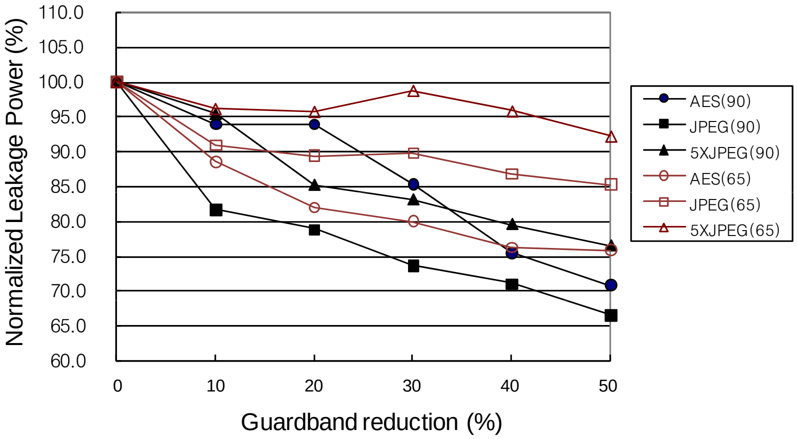


Figure 2.13: Total leakage power versus guardband reduction.

results of Table 2.10 are in alignment with the trends we observe for area and wirelength versus guardband reduction above.

2.4.3 Impact on Design Turnaround Time

Table 2.11 shows the substantial impact of guardband reduction on total SP&R flow runtime for the 90nm $JPEG$ testcase. Also, Figure 2.14 shows the impact of guardband reduction on total SP&R flow runtime for AES, $JPEG$ and $5XJPEG$ designs using 90nm and 65nm technologies. The data shows up to 41% reduction in

Table 2.10: Critical-path delay variations across different guardband reductions.

Cases	GB reduction	Timing corner	Total path delay (ns)	Average stage delay (ns)
	0%	WC	3.520	0.147
		BC	1.435	0.060
F	10%	WC	3.406	0.142
		BC	1.525	0.064
	40%	WC	3.069	0.128
		BC	1.813	0.076
	50%	WC	2.960	0.123
		BC	1.910	0.080
B	10%	WC	3.515	0.146
		BC	1.437	0.060
	40%	WC	3.502	0.146
		BC	1.443	0.060
	50%	WC	3.497	0.146
		BC	1.445	0.060
F+B	10%	WC	3.410	0.142
		BC	1.523	0.063
	40%	WC	3.085	0.129
		BC	1.804	0.075
	50%	WC	2.979	0.124
		BC	1.899	0.079

SP&R flow runtime with a 40% guardband reduction. Table 2.12 shows that total SP&R flow runtime decreases by 7% with 30% guardband reduction for the 45nm embedded processor core. In real-world design contexts, such a substantial reduction in SP&R runtime can, at a minimum, reduce tapeout schedule risk, and permit additional optimization iterations and design space explorations. A substantial reduction in SP&R

flow runtime can also reduce time to market for an IC product.

Another very clear benefit from guardband reduction can be seen from analysis of violations in signoff analysis. Recall that if there are violations at the signoff stage, then it is necessary to go back to the P&R stage and fix them. The number of design iterations required to fix violations is reflected by a variety of ‘figure of merit’ parameters that are often tracked by designers, e.g., total number of violations, worst negative slack (WNS), and total negative slack (TNS). These three metrics represent different views of the design timing characteristics:

- The total number of violations represents how many violating points the designer needs to worry about.
- WNS represents the largest timing violation.
- TNS indicates how difficult fixing all the current violations in a design can be.

From these numbers, we can estimate the difficulty of meeting timing constraints, and how many iterations will be required. For example, from the total number of violations and TNS of hold time analysis, the designer can estimate how many buffers are needed to fix the violations, and indirectly estimate how much the standard-cell area will increase as a result. Or, the designer can use the WNS value to see how close a design is to becoming feasible with respect to timing constraints.

Table 2.13 shows various figures of merit for the $90nm\ JPEG$ post-P&R result obtained with a 0% guardband reduction, when evaluated using other (10%, 40%, 50%) guardband reductions. The table gives number of violations, worst negative slack, and total negative slack, with respect to both setup and hold constraints using signoff flow. Here, we can see very substantial benefits from guardband reduction. For example, with a 40% guardband reduction, the vast majority of timing violations are erased, and the WNS and TNS metrics are also reduced substantially (by up to 100%). This will clearly improve timing convergence by reducing design iterations.

Table 2.11: Guardband reduction versus SP&R flow runtime for 90nm $JPEG$ design at 300MHz.

Runtime	0%		10%		40%		50%	
	sec	%	sec	%	sec	%	sec	%
F	7129	100	5653	79.3	4068	57.1	4061	57.0
B	7208	100	7327	101.7	7507	104.1	5755	79.8
F+B	6950	100	5729	82.4	4366	62.8	4061	58.4

Table 2.12: Guardband reduction versus SP&R flow runtime for 45nm embedded processor core at 500MHz.

Runtime	0%		10%		30%		50%	
	sec	%	sec	%	sec	%	sec	%
F	13032	100	12155	93.27	12353	94.79	10662	81.81

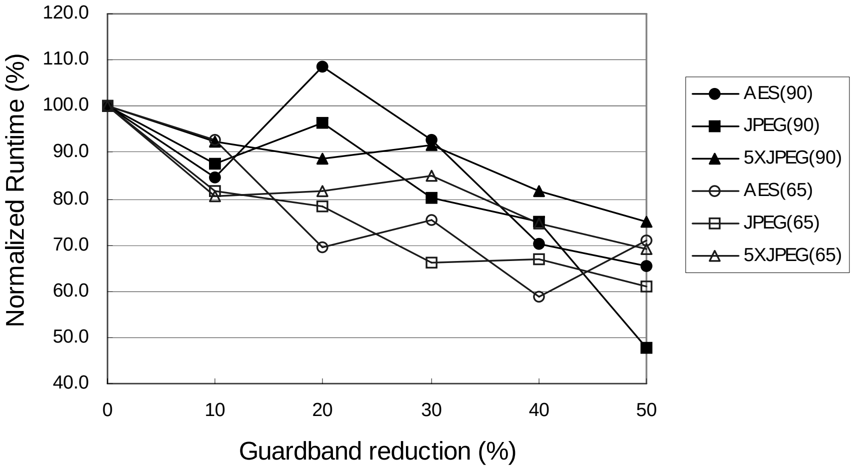


Figure 2.14: Guardband reduction versus total SP&R flow runtime.

2.4.4 Impact on Design Yield

Guardbanding exists in today's design methodologies to help guarantee high yield in spite of process variability. In this subsection, we quantify the impact of guard-

Table 2.13: Guardband reduction versus number of violations, worst negative slack (WNS) and total negative slack (TNS).

			Guardband reduction			
			0%	10%	40%	50%
F	Setup	# of viols	235	3	0	0
		WNS (ns)	-0.126	-0.016	0	0
		TNS (ns)	-9.95	-0.03	0	0
	Hold	# of viols	4414	675	526	287
		WNS (ns)	-0.116	-0.045	-0.030	-0.028
		TNS (ns)	-259.68	-15.19	-4.20	-1.06
B	Setup	# of viols	235	231	203	198
		WNS (ns)	-0.126	-0.121	-0.11	-0.10
		TNS (ns)	-9.95	-8.97	-6.29	-5.43
	Hold	# of viols	4414	4410	4404	4400
		WNS (ns)	-0.116	-0.116	-0.116	-0.116
		TNS (ns)	-259.68	-259.39	-259.59	-258.34
F+B	Setup	# of viols	235	3	0	0
		WNS (ns)	-0.13	-0.011	0	0
		TNS (ns)	-9.95	-0.02	0	0
	Hold	# of viols	4414	676	524	298
		WNS (ns)	-0.116	-0.045	-0.030	-0.034
		TNS (ns)	-259.68	-15.24	-4.30	-1.11

band reduction on design yield. We believe that such quantification will be an essential part of manufacturing-aware design methodologies in the future.

Overall yield is modeled as the product of *random* defect yield, which depends on die area, and *parametric* yield, which is independent from die area.

$$Y = Y_r \cdot Y_p \tag{2.1}$$

Random defect yield (Y_r). A variety of models exist for the spatial distribution of random electrical faults across a wafer, and random defect yield Y_r. The fundamental difference between these models is the assumed distribution of the random defects [112]. Commonly, random defects are characterized by defect density parameter d, and clustering parameter α. The average number of defects on a chip of area A is Ad. The number of defects x in a random chip is an integer-valued random variable, and the observed phenomenon of defect clustering is effectively modeled by assuming a negative binomial probability density function for x [46]:

$$\begin{aligned} p(x) &= Prob(\text{number of defects on chip} = x) \\ &= \frac{\Gamma(\alpha + x)}{x!\Gamma(\alpha)} \frac{(Ad/\alpha)^x}{(1 + Ad/\alpha)^{\alpha+x}} \end{aligned} \tag{2.2}$$

where $\Gamma(x)$ is the Gamma function. The yield with respect to random defects is obtained as the probability $p(0)$ of having no defect on a chip. Substituting $x = 0$ in Equation (2.2):

$$Y_r = (1 + Ad/\alpha)^{-\alpha} \tag{2.3}$$

If we use $\alpha = \infty$, which corresponds to the case of unclustered defects, Equation (2.3) gives a *Poisson* density function with mean Ad, and the yield with respect to random defects is pessimistically estimated as:

$$Y_r = e^{-Ad} \tag{2.4}$$

From Equation (2.4), we conclude that random defect yield (Y_r) will increase with decreasing area (A) accomplished by guardband reduction. Other widely used random defect yield models are Murphy and Bose-Einstein as shown in Equations (2.5) and (2.6), respectively [112].

$$Y_r = (\frac{1 - e^{-Ad}}{Ad})^2 \tag{2.5}$$

$$Y_r = \frac{1}{(1 + Ad)^n} \tag{2.6}$$

In the Bose-Einstein model, n is the complexity factor. A comparison of the above yield models shows that for small defect densities (0.2 /cm^2), all three models predict similar yield results. Even for larger defect densities (i.e., 1 and 2 /cm^2), for die areas less than $100mm^2$, the deviations are within 5% [112].[7]

In addition, hypothetically, reduced chip area could decrease wire spacing which would then increase the likelihood of short defects. Hence, we perform random defect yield analysis using Edinburgh Yield Estimator - Sampling (EYES) [16]. The EYES uses a sampling technique to estimate the properties of the IC layout as a whole.

We use a Poisson yield model and account for both open and short faults in the same layer. We do not consider inter-layer faults such as dielectric and pinhole faults. In the experiments, we analyze random defect yield of GDSII for $5XJPEG$ implemented with each reduced guardband. Table 2.14 shows the random defect yield values from Equations (2.4), (2.5), (2.6), and EYES for 65nm $5XJPEG$ design. In Bose-Einstein model we use the physical chip area, A, and an average defect density, d.[8] We use defect density of 0.2 (/cm^2) for all the equations as well as EYES experiments.

Table 2.14: Random defect yield for 65nm $5XJPEG$ design.

RGB (%)	Chip area (cm^2)	Y_r from Eq. (2.4)	Y_r from Eq. (2.5)	Y_r from Eq. (2.6)	Y_r from from EYES
0	0.014562	0.99709	0.99709	0.99709	0.9850
10	0.014205	0.99716	0.99716	0.99716	0.9855
20	0.014084	0.99719	0.99718	0.99719	0.9867
30	0.014093	0.99719	0.99718	0.99718	0.9832
40	0.014219	0.99716	0.99716	0.99716	0.9865
50	0.013992	0.99722	0.99720	0.99720	0.9880

Due to the small size of the sample design, the resulting yield values are not significantly different for each guardband. However, it is clear that random defect yield does not decrease with the guardband reduction.[9]

[7] In this work, we assume a Poisson model for random defect yield estimation.

[8] If chip area and a general defect density are used instead of critical area and specific defect density per critical area, then the complexity factor of Bose-Einstein equation is equal to 1 [112].

[9] Chip size is determined by the resulting standard cell area after synthesis. Due to the inherent noise of optimization, the chip size trend shows some glitches, e.g., at 40% guardband reduction.

In the simple approach, the critical area, to be used in the above models, is equal to the die area. However, there needs to be a refinement by adding up the active area of the logic, memory, and IO cells and assigning different defect density values to each of these components. Assuming that the wafer fab provides a single, average d, we can use a simple approach that assigns a 30% addition to d for memory blocks and a 20% reduction to d for IO cells. Indeed, the proper way is to get yield information from chips with logic only, and memory only, and then calculate defectivities for each [112]. Therefore, Equation (2.4) is modified as follows.

$$Y_r = e^{-(A_{memory}d_{memory}+A_{logic}d_{logic}+A_{IO}d_{IO})} \tag{2.7}$$

where, A_{memory}, A_{logic}, A_{IO} denote memory, logic, and IO cell physical area, and d_{memory}, d_{logic}, d_{IO} denote memory, logic, and IO cell defect density values, respectively.

Parametric yield (Y_p). Yield with respect to *parametric variation*, Y_p, can be estimated by considering a normal distribution with best-case and worst-case corners being set at -3σ and 3σ, respectively. The 3σ window can be taken to define the original guardband (i.e., 0% guardband reduction, with range 6σ).[10] Then, assuming no change in manufacturing variability, a K% design guardband reduction would result in a reduced range of (6σ)(100-K)/100. To calculate the parametric yield impact of design guardband reduction with no change of manufacturing variability, we may use the error function (erf, i.e., cumulative distribution of the normal distribution) for the appropriate range. For example, $Y_p(RGB\%)$ with respect to 0% guardband reduction can be computed as

$$Y_p(0) = \frac{1}{2}(1+erf(\frac{3}{\sqrt{2}})) - \frac{1}{2}(1+erf(\frac{-3}{\sqrt{2}})) = 0.9973 \tag{2.8}$$

Number of good dies. To assess the impact of guardband reduction on design yield, we track the change in the number of good dies per wafer as we reduce the design guardband. To calculate the number of good dies per wafer, we first compute the gross

[10] We understand that these assumptions are appropriate to the current practice. The present discussion can be easily modified to use a different $k\sigma$ window.

number of dies per wafer as described in [188]:

$$N_{gross} = \pi(\frac{r^2}{A} - \frac{2r}{\sqrt{2A}}) \tag{2.9}$$

where A represents the die area which is fabricated on a wafer with radius r. In the above equation the second term accounts for wasted area around the edges of a circular wafer. Using Equations (2.1) and (2.9), the number of good dies per wafer is:

$$N_{good} = Y \cdot N_{gross} \tag{2.10}$$

There are two main scenarios for the guardband reduction.

1. We are able to improve the process so as to reduce the amount of guardbanding. This scenario corresponds to performing "iso-dense" timing analysis [75].

2. We simply apply a reduced guardband during the design process, even though the actual variability of the manufacturing process remains the same. This scenario corresponds to the $Y_p(RGB\%)$ calculation above.

Scenario (1) implies that Y_p remains at 0.9973, while overall yield increases because we benefit from decreased random defect yield loss due to decreased die area as well as from the reduced die area itself. Table 2.15 shows the number of good dies per wafer for each guardband reduction. For this analysis, we assume that a typical 65nm SOC design that has 0.85cm^2 die area and is composed of 0.48cm^2 of standard logic cells and 0.37cm^2 of fixed blocks, i.e., embedded SRAM, analog cores and IO cells. We use a 300mm wafer diameter to calculate the number of dies, 0.2 /cm^2 and 0.21 /cm^2 defect density values (for logic cells and fixed blocks, respectively) to calculate random defect yield. Area reduction values are the average results from 65nm testcases of the experiments.[11] The use of an average area reduction is justified since all testcases across 90nm and 65nm show results that are monotone in guardband reduction value, and that have standard deviation (for any given guardband reduction value) of less than 5% (see Figure 2.10). The table shows that 40% guardband reduction results in approximately 10% increase in the number of good dies.

[11] If redesigned circuits with guardband reduction do not result in chip area reduction, the random defect yield improvement will decrease.

Table 2.15: The number of good dies per wafer for Scenario (1) guardband reduction.

RGB	Expected area (cm^2)		Y_p	Y_r	Y	#gross	#good
(%)	Logic	Fixed	(%)	(%)	(%)	dies/wafer	dies/wafer
0	0.480	0.370	99.7	82.5	82.3	759	624
10	0.449	0.370	99.7	83.0	82.8	789	653
20	0.438	0.370	99.7	83.2	83.0	801	665
30	0.430	0.370	99.7	83.3	83.1	809	672
40	0.417	0.370	99.7	83.6	83.3	823	686
50	0.408	0.370	99.7	83.7	83.5	833	695

Scenario (2), which is the focus of the discussion henceforth, changes $Y_p(RGB\%)$ as described above and is more pessimistic because no process improvement is assumed: the design guardband reduction increases random defect yield Y_r due to reduced die area, but this trades off against decreased Y_p.[12] Table 2.16 shows the number of good dies per wafer for each guardband reduction with the same assumptions used for Scenario (2). We observe that Y_p keeps decreasing as we reduce guardband as shown in Column 4 in the table, but we observed that decreased die area increases the number of good dies per wafer even without process enhancement.

Figure 2.15 shows the change in the number of good dies per wafer over the guardband reduction for different defect clustering. From these plots, we can see that the number of good dies per wafer is maximized at around 20% of guardband reduction. In this figure, the assumption is that the entire design consists of logic cells. This trend will not be changed by the clustering of defects. Figure 2.16 shows level curves of the number of good dies per wafer, plotted against guardband reduction (y-axis) and area reduction (x-axis). The dashed trace shows (area reduction, guardband reduction) points

[12] There is a third scenario, where the design floorplan is fixed so that standard-cell area reduction (due to reduced design guardbanding) does not result in any die area reduction. In this third scenario, wirelength reduction in the standard-cell blocks will result in lower metal density, which will reduce particle defect yield loss (since critical area is a function of wire density [88]). Hence, even when there is no change in die area with guardband reduction (e.g., with fixed-floorplan or pad-limited designs), we can expect a certain amount of Y_r improvement which increases the number of good dies per wafer. However, we do not currently have the tool infrastructure or foundry critical-area analysis decks needed to study this scenario.

Table 2.16: The number of good dies per wafer for Scenario (2) guardband reduction.

RGB	Expected area (cm^2)		Y_p	Y_r	Y	#gross	#good
(%)	Logic	Fixed	(%)	(%)	(%)	dies/wafer	dies/wafer
0	0.480	0.370	99.7	82.5	82.3	759	624
10	0.449	0.370	99.3	83.0	82.4	789	651
20	0.438	0.370	98.4	83.2	81.9	801	656
30	0.430	0.370	96.4	83.3	80.3	809	650
40	0.417	0.370	92.8	83.6	77.5	823	638
50	0.408	0.370	86.6	83.7	72.5	833	604

that we have realized experimentally. We see that the number of good dies increases by up to 4.1%, then starts to decrease, until the onset of yield degradation beyond 40% reduction in guardband.

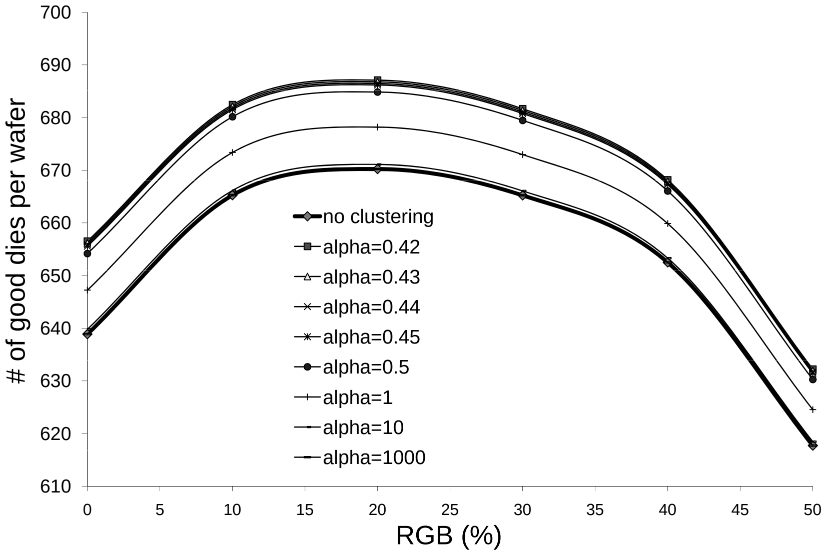


Figure 2.15: Change in the number of good dies per wafer, versus guardband reduction (%) and defect clustering.

We also estimate the impact of reduction of only the process guardband, since operating voltage and temperature can be fixed due to the design's requirements, as

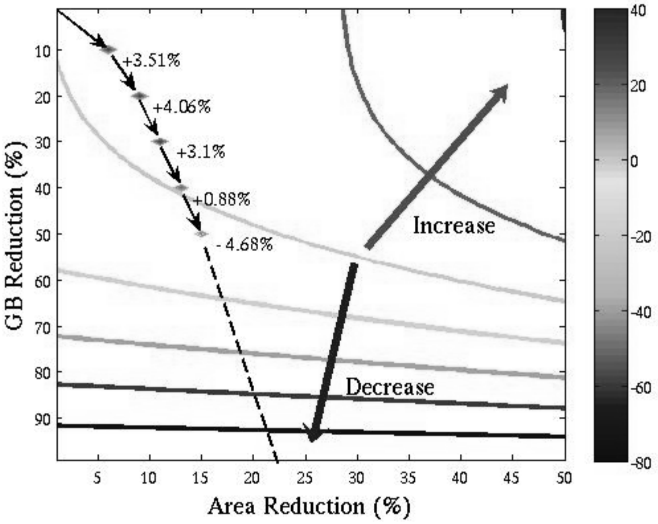


Figure 2.16: Change (%) in the number of good dies per wafer, versus guardband reduction (%) and area reduction (%).

mentioned earlier in Section 2.2.1. To calculate the area reduction from the process guardband reduction, we map the delay reduction percentages to the area reduction percentages from the experimental results on the logic area reduction in Figure 2.10 and worst-case delay reduction in Figure 2.1. Figure 2.17 shows the simple linear regression results for area reduction versus guardband reduction.

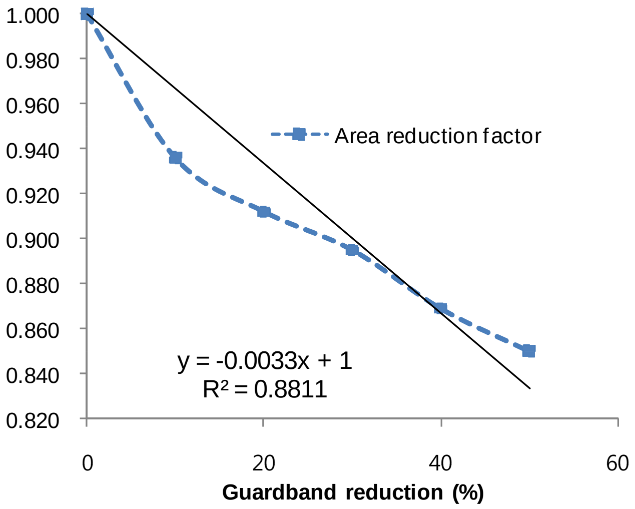


Figure 2.17: Linear fit for area reduction (%) versus guardband reduction (%).

We then compute Y_p and Y_r. Figure 2.18 shows the change in the number of good dies per wafer over the guardband reduction for two different assumptions: (1) with fixed blocks of which size are not changed with guardband reduction, and (2) without fixed blocks which implies that hard macros are newly designed corresponding to the guardband reduction or a design without hard macros. This plot reflects again a typical SOC in 90nm and 65nm, with die area 0.85cm^2, and 0.48cm^2 of the die being logic that is affected by the guardband reduction and 0.37cm^2 of hard macros that may or may not be affected by guardband reduction. We observe that the number of good dies per wafer is maximized at around 24$\%$ process guardband reduction which results in 3.6$\%$ increase in the number of good dies per wafer,[13] even with over half of the design's area being fixed. The number of good dies per wafer can increase up to 10$\%$ at 38$\%$ process guardband reduction, if we redesign hard macros according to the guardband reduction or if a design is composed of pure logic cells.

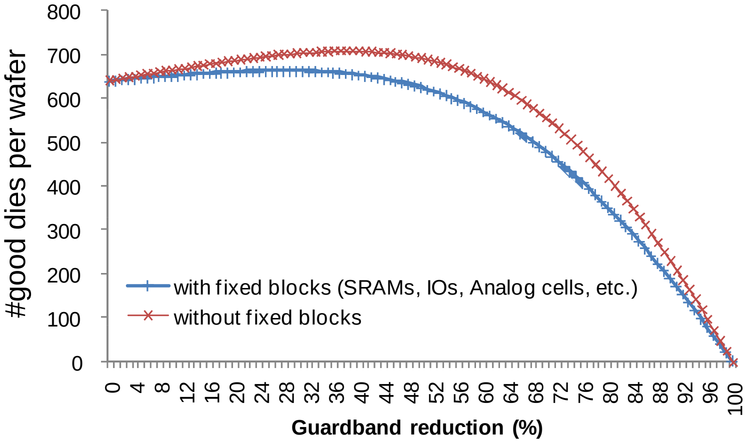


Figure 2.18: Change in the number of good dies per wafer only for the process guardband reduction ($\%$).

[13] 4$\%$ increase in the number of good dies is significant. For example, if a design needs 50K wafers to produce 30M good units, and the cost per wafer is $3K, the 4$\%$ represents a reduction of 2K wafers for the same number of good units, and the cost saving is $6M.

2.5 Conclusions and Research Directions

In this chapter, we establish an experimental framework and then experimentally quantify the impact of model guardband reduction on outcomes of the synthesis, place and route (SP&R) implementation flow. We assess the impact of model guardband reduction on various metrics of design cycle time and design quality, using open-source and industrial embedded processor core with production 90nm, 65nm, and 45nm technologies and libraries. We observe typical (i.e., average) reductions of 13%, 12%, 13%, and 19% in standard-cell area, total routed wirelength, dynamic power, and leakage power metrics from a 40% reduction in library model guardband (i.e., open-source testcases) and observe up to 8%, 7%, 5%, and 10% reductions in standard-cell area, total routed wirelength, dynamic power, and leakage power for the embedded processor core at 30% guardband reduction. We also observe 100% reduction in the number of timing violations for a netlist that is synthesized with original library and extraction guardbands; this improvement can prove to be a significant factor in timing closure and design cycle turnaround time. Last, we quantify the impact of the guardband reduction on design yield. For yield assessment, Scenario 2 with fixed blocks shows up to 4% increase in the number of good dies per wafer with 27% guardband reduction. Interestingly, this increase in the number of good dies comes without any assumption of improved manufacturing capability (i.e., variability reduction). In addition, statistical analysis and optimization methodologies may not provide, by themselves, sufficient improvement of circuit metrics (e.g., [45] cites a 2% power reduction from statistical optimization; see also [134]). Therefore, the results suggest that there is justification for the design, EDA and process communities to enable guardband reduction as an economic incentive for manufacturing-friendly design practices.[14]

We have two directions for future research: (1) to assess the impact of RGB on memory embedded designs, and (2) to assess the feasibility of simultaneous guardband reduction and voltage lowering to find the best combination of guardband and supply voltage which optimizes for the area, yield, and power.

[14] As we have noted above: Although there exist clear decreasing trends in area and wirelength with respect to guardband reduction, due to the noise in the commercial tools, small guardband reductions (e.g., by $< 10\%$) may not always change flow outcomes as noticeably or consistently.

2.6 Acknowledgments

Chapter 2 is in part a reprint of “Impacts of Guardband Reduction on Design Process Outcomes: A Quantitative Approach”, *IEEE Transactions on Semiconductor Manufacturing* 22(4) (2009) and “Quantified Impacts of Guardband Reduction on Design Process Outcomes”, *Proc. International Symposium on Quality Electronic Design*, 2008. I would like to thank my coauthors Kambiz Samadi and Professor Andrew B. Kahng.

I also would like to thank Dr. Riko Radojcic and Mr. Durodami Lisk of Qualcomm CDMA Technologies, Inc. for providing the industrial testcase as well as for useful conversations that have strengthened the experimental results of the work.

Chapter 3

Variation Modeling

In this chapter, we review previous works on variation modeling and propose a novel variation modeling framework that can reconstruct the details of *multiple*, *simultaneously occurring* systematic variation maps (e.g., across-field or across-die variations) from a small number of measurements of natural timing paths in a design. The proposed method avoids the cost of additional silicon area to embed test structures, as well as the cost of extra design and test efforts to sensitize a large number of timing paths. Motivated by recent work on the theory and applications of *compressed sensing*, and the fact that the set of coefficients of the *discrete cosine transform* (DCT) of a specific variation is unique and sparse, we formulate circuit delay (specifically, in natural timing paths) as a function of the DCT coefficients, and find the DCT coefficients using a linear programming solver, starting from the measured delay of timing paths. The DCT coefficients thus obtained are then used to build variation maps that can be fed to advanced manufacturing optimization tools or to various post-silicon design optimizations.

3.1 Classification of Variation

Process variation can be generally classified into two types of components, *systematic* and *random*.

- *Systematic* components are sources of systematic variation that are known and correlated across the field. This variation can be modeled and compensated.

- *Random* components are sources of random variation that are unknown, or known as pure random noise. This form of variation is difficult to model or compensate accurately.

Many sources and parameters contribute to systematic variation. Systematic variation can be classified as either *pattern-dependent* or *location-dependent* with respect to the variation source and its characteristic appearance.

- *Pattern-dependent* (high-frequency) variation results when different layout patterns have varying density, neighboring patterns and/or orientations.
- *Location-dependent* (*spatial*) (low-frequency) variation refers to spatial-location variation, which occurs despite having identical layout patterns over all locations.

Systematic variation can also be classified according to different levels of spatial hierarchy.

- *Wafer-to-wafer* variation is caused by drift during the equipment operation process from one wafer to the next, as a result of different positioning of wafers in a boat (lot) during a batch furnace step [170], etc. The magnitude of this variation can be quite large. However, this is expected to be compensated by equipment tuning as well as across-wafer or across-field variation compensation techniques.
- *Across-wafer* (*intra-wafer*) variation appears as strong, radial (concentric circles) spatial variation across the wafer. This type of variation is exacerbated by single-wafer processing for 300mm wafers. Overlay error, non-uniform etch rate over the wafer [110], temperature non-uniformity in post-exposure bake (PEB) [195], and resist thickness variation during spin-on resist coating are common sources.
- *Across-field* (*intra-field*) variation is caused by, e.g., lithographic sources such as focus and exposure variations within a field [42], lens aberrations [43], mask errors [189] and variations in etch loading [87].
- *Across-die* (or *within-die*) variation is due to sources such as pattern-dependent variations [186] (due to OPC residuals, or well proximity effects [187]) and dy-

namic operational variations (due to temperature and voltage variations, or crosstalk noise).[1] Here we assume that location-dependent variations are only from wafer- and field-level sources.

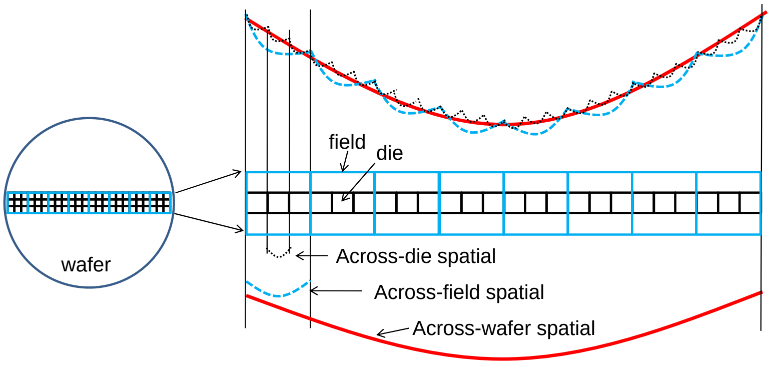


Figure 3.1: Conceptual decomposition of 1-D systematic spatial variations at different levels of hierarchy.

3.2 Previous Approaches to Variation Modeling

In this section, we briefly review the previous variation modeling techniques.

3.2.1 Regression: Closed-Form Modeling

Regression is a popular method to describe systematic variation as a closed-form equation.

Parametric Regression. Linear, quadratic [170] [65] [56] or higher-degree [138] polynomial functions are used to model systematic spatial variation using standard linear regression. Least-squares regression is a powerful method to find a unique fitting function in overdetermined systems with minimum (sum of squared) error. Systematic process variation can be modeled as a function or a set of functions in three dimensions.[2]

[1] Other terminology includes *die-to-die* variation, which can be regarded as combined variation due to wafer-to-wafer, across-wafer and across-field variation sources.

[2] The x- and y-axes are used to specify the location of measurement points, and the z-axis is used to describe the measured data.

Matrix algebra is used for least-squares regression. An overdetermined system is defined as

$$\mathbf{AX} = \mathbf{T}$$

where $\mathbf{A}$ is an $M \times N$ matrix that stores evaluated values of each term in the assumed function, $\mathbf{X}$ is an $N \times 1$ matrix for unknown coefficients that uniquely determine the function, and $\mathbf{T}$ is an $N \times 1$ matrix for target or measured values, with $M > N$. When M = N, $\mathbf{X}$ can easily be calculated as $\mathbf{A}^{-1}\mathbf{T}$. However, when $M \neq N$, $\mathbf{X}$ is calculated as $(\mathbf{A}^{\mathbf{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathbf{T}}\mathbf{T}$. After determining coefficients of the regression function, we can calculate expected values at every point by function evaluation.

For example, for first-order polynomial ($V(x, y)$ = ax + by + c) with four measured values t_1, t_2, t_3, and t_4, at locations (x,y) = { (1,1), (1,2), (2,1), (2,2) }, the $\mathbf{A}$, $\mathbf{X}$, and $\mathbf{T}$ matrices are defined as:

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 2 & 1 & 1 \\ 2 & 2 & 1 \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \text{ and } \quad \mathbf{T} = \begin{pmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{pmatrix}$$

The underlying assumption of parametric regression is that systematic properties of location-dependent, across-field variation can be described by a polynomial function. Parametric regression has its advantages: results are compact and the model coefficients for different wafers and lots are useful for tracking lot-to-lot and wafer-to-wafer variations. However, the primary drawback of regression techniques is that they assume a large degree of global analytic regularity in the wafer-level variation, which is often not the case [170].

Nonparametric regression (piecewise polynomial). If the actual systematic variation has complex form, parametric regression requires higher-degree polynomial functions to reduce estimation error. However, Runge's phenomenon [153] points out that higher-degree polynomials can lead to large undulations at the model boundary, causing a large interpolation error near the boundary.

The easiest way to mitigate Runge's phenomenon with parametric regression methods is to use a non-uniform sampling that samples more data near the boundary. For example, along the range of the model, e.g., -1 $\leq x \leq 1$, sampling density can be $1/\sqrt{1-x^2}$. In any case, a higher-degree polynomial requires more data points. Another way to mitigate Runge's phenomenon is to use nonparametric regression. For example, spline regression determines a set of polynomial functions where each is fitted only for a specific subregion of the entire modeling space. Several different spline methods are discussed in Appendix B of [170]. In general, when trying to decrease the interpolation error, one can increase the number of polynomial pieces which are used to construct the spline, rather than increasing the degree of the polynomials used.

For three-dimensional surface modeling, the spline concept can be extended by triangulation of the surface. In [170], a cubic function is used to model each triangular surface subject to a smoothness constraint of matching second derivatives of polynomials at each boundary between triangles.

3.2.2 Set of Expected Values Modeling

Regression gives a closed-form equation for the variation. Using the equation, we can find the expected value of any points on die, field and wafer. Separately from the closed-form equations, a method is needed to model the value of each location itself. Such a method can be used as a preprocessing step before applying regression or other modeling methods. Two methods to find a representative value for each point are as follows.

Pointwise average. Average values of the same locations within die, field or wafer can be used to represent the value of specific points. Pointwise average can be performed on a per-die, per-field, or per-wafer basis. We define the following notations.

- Levels of hierarchy:
  - Wafer: wafer-related terms have the subscript w
  - Field: field-related terms have the subscript f

  - Die: die-related terms have the subscript d

- Quantities
  - n_w: # wafers
  - n_f: # fields per wafer
  - n_d: # dies per field
  - n_m: # measurement points per die

- Inferred variation maps
  - $V_{w,s}(x,y)$: inferred across-wafer systematic variation map
  - $V_{f,s}(x,y)$: inferred across-field systematic variation map
  - $V_{d,s}(x,y)$: inferred across-die systematic variation map

- Actual variation maps (measured)
  - $V^*_{w,s}(x,y)$: actual across-wafer systematic variation map
  - $V^*_{f,s}(x,y)$: actual across-field systematic variation map
  - $V^*_{d,s}(x,y)$: actual across-die systematic variation map

Given all measured data (= $n_w \times n_f \times n_d \times n_m$), we can calculate pointwise average on wafer, field, or die by averaging values of corresponding points within targets.

$$
\begin{aligned}
V_{w,s}(x,y) &= \left(\sum_{i=1}^{n_w} V^*_{w(i)}(x,y) \right) / n_w \\
V_{f,s}(x,y) &= \left(\sum_{i=1}^{n_w} \sum_{j=1}^{n_f} V^*_{f(i,j)}(x,y) \right) / \left(n_w n_f \right) \\
V_{d,s}(x,y) &= \left(\sum_{i=1}^{n_w} \sum_{j=1}^{n_f} \sum_{k=1}^{n_d} V^*_{d(i,j,k)}(x,y) \right) / \left(n_w n_f n_d \right)
\end{aligned}
$$

where i, j and k indicate the indices for wafers, fields, and dies, respectively, e.g., $d(i,j,k)$ represents the k^{th} die in j^{th} field in i^{th} wafer under testing.

Expectation Maximization (EM) [150]. During the fabrication process, process engineers can fabricate some experimental wafers to assess any proposed process changes. In addition, not all of the measured points are fabricated correctly. The collected data from the former can show aberrant values (outliers) and those from the latter can result in missing data. Reda et al. [150] propose to use an expectation maximization algorithm to estimate any missing values based on multivariate statistical techniques, and propose to use a chi-square distribution of $Mahalanobis$ distance to distinguish the outliers.

The EM algorithm consists of two steps:

- 1. *E-step* estimates the expected values of the missing measurements, given the current *Maximum Likelihood Estimator (MLE)*, μ and Σ.
- 2. *M-step* re-estimates the distribution of parameters (μ, Σ) to maximize the likelihood of the data, based on the estimated expected values from the E-step.

The parameter μ of the first iteration is the same as pointwise average and will be updated through a number of iterations. Upon completion of EM, μ denotes systematic variation, and Σ denotes the amount of random variation.

3.2.3 Smoothing: Moving Average versus Meshed Spline

Since there exists random noise in both processes and measurements, raw data may needs to be preprocessed to remove local abrupt variation. After this smoothing, we can apply regression techniques to model systematic spatial variation. *Downsampled moving average* (DSMA) and *meshed spline method* (MSM) are used in [170] for this purpose.

Downsampled moving average. DSMA uses the intuitive notion of a moving average to smooth rapidly varying features arising due to die-level, pattern-dependent effects. DSMA first defines an $n \times n$ grid on a target and downsamples by taking every x rows or columns and constructing a sparser grid. The value for each point in the downsampled grid is calculated by averaging values within a certain distance of that point. During downsampling and averaging, high-frequency variation terms are filtered (smoothed). However, with insufficient data points, downsampling may lose important variation components, and moving average can smooth out actual high-frequency variation.

Meshed spline. Splines with downsampling can also be used to smooth input data, in addition to the regression as discussed above. In the MSM approach, the input data is interpolated onto a regular grid whose size is determined by the number of measurement points per field. The procedure then selects grid lines (every x lines), and smooths input data on the selected grid using spline method. This process is performed in horizontal and vertical directions and can be repeated in other directions such as along 45° and 135° lines. The results of MSM in one direction and the others are then averaged together at each grid intersection.

3.2.4 Hierarchical Decomposition of Variation

To model across-wafer variation in a closed-form equation, any of the regression methods can be used on the given data. However, due to the different sources of variations in each physical hierarchy level, such as wafer, field, and die, the variation at each level can appear differently. For instance, periodic variation across wafer can be observed due to the existence of across-field variation as shown in Figure 3.1.

Freidberg et al. present a method to decouple across-wafer and across-field variation when using parametric regression. Figure 3.2 shows original wafer-level CD measurements. The measured values from each field are pointwise-averaged within a field as shown in Figure 3.3. A second-order polynomial is then used to find the systematic variation across field. Figure 3.4 shows the fitted map for the pointwise-averaged variation in Figure 3.3.

$$fit = \beta_1 x^2 + \beta_w y^2 + \beta_3 x + \beta_4 y$$

After the across-field variation is obtained, across-wafer variation is modeled after subtracting across-field variation (in Figure 3.4) from the original raw data (in Figure 3.2), which is shown in Figure 3.5. Finally, across-wafer variation is again modeled using a second-order polynomial as shown in Figure 3.6.

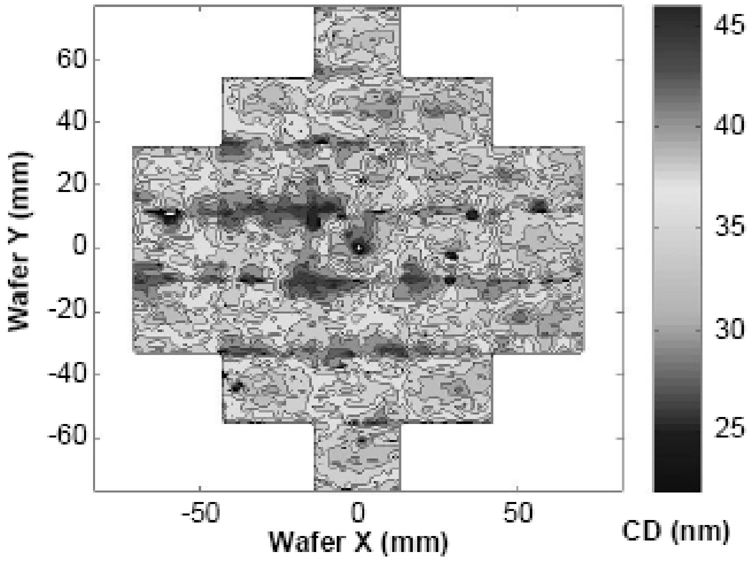


Figure 3.2: Original full-wafer CD measurements. Figure reproduced from [65].

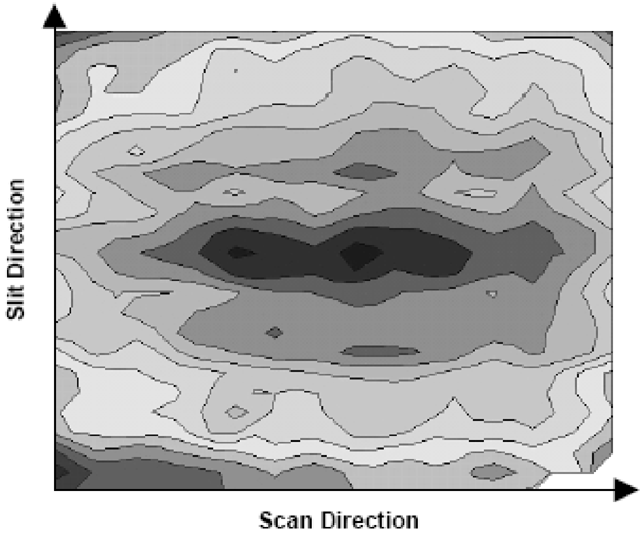


Figure 3.3: Pointwise-averaged variation within a field. Figure reproduced from [65].

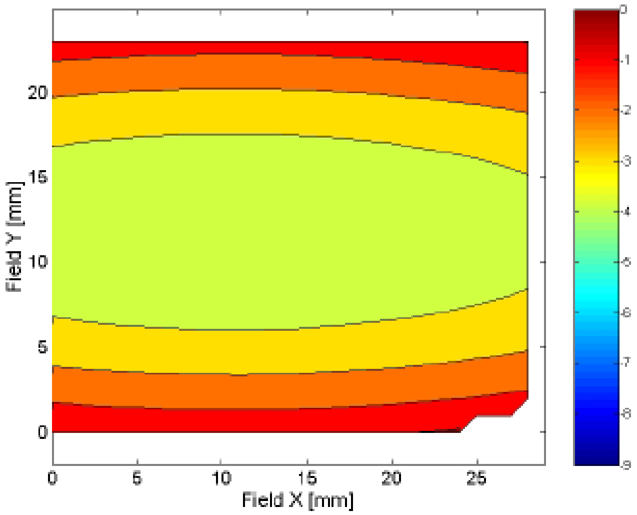


Figure 3.4: Modeled within-field variation. Figure reproduced from [65].

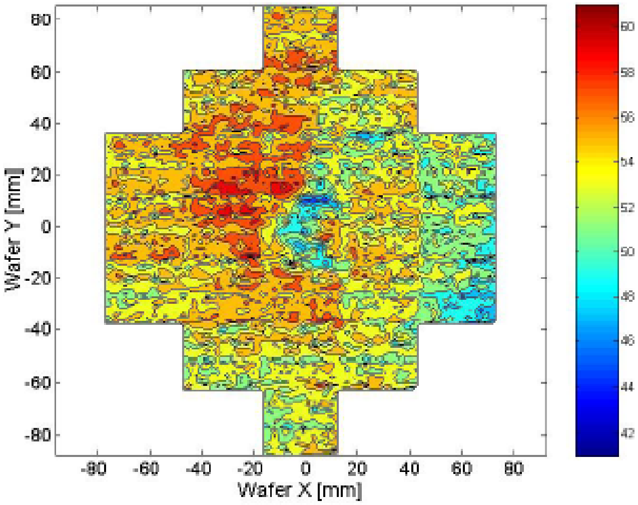


Figure 3.5: Full-wafer CD map after average within-die (field) CD fingerprint removed. Figure reproduced from [65].

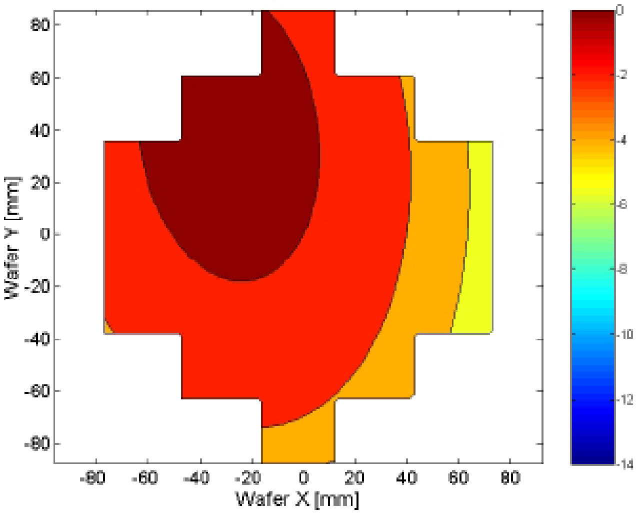


Figure 3.6: Modeled across-wafer variation with die (field) effects removed. Figure reproduced from [65].

3.2.5 Combined Techniques

We have discussed several different techniques for smoothing and regression. However, the accuracy of each technique strongly depends on the measured data. A linear combination of different types of variation models is presented in [170]. Given variation estimators E_1, E_2, ..., E_k, the combined variation model, E_{eff}, can be formulated as

$$E_{eff} = \sum_i w_i E_i, \qquad \sum_i w_i = 1$$

where the w_i are normalized weights for each respective model. Without a priori knowledge, all weights w_i are equal. However, if the models have known degrees of efficiency or accuracy, higher-rank models may be given higher weight. Assuming that the errors associated with each model are independent and normally distributed with roughly the same variance, the combined effective variation model will have smaller variance by approximately a factor of $\sum_i^k {w_i}^2$.

3.2.6 Experimental Results

We apply several variation modeling techniques to real, measured data. We obtain measured saturation current ($I_{d,sat}$) for a $28nm$ NMOS from an IC foundry. Our goal is to extract systematic variations which can be used to inform process variation compensations based on advanced manufacturing techniques such as Dose Mapper [2] and CDC Pixer [29]. We also compare various modeling techniques and flows with respect to the accuracy of variation modeling.

The data set consists of ∼2,300 data points, i.e., (17 wafers) × (17 fields per wafer) × (8 measured data per field). Figure 3.7 shows a baseline variation map constructed from pointwise average for all wafers (i.e., blue circles in the figure), followed by an interpolation using Matlab's *griddata* function [12].[3] Mean (μ_{orig}) and sigma (σ_{orig}) over all data points are respectively 902.5 and 62.4, and the sigma-to-mean ratio (σ/μ) is 6.9%.

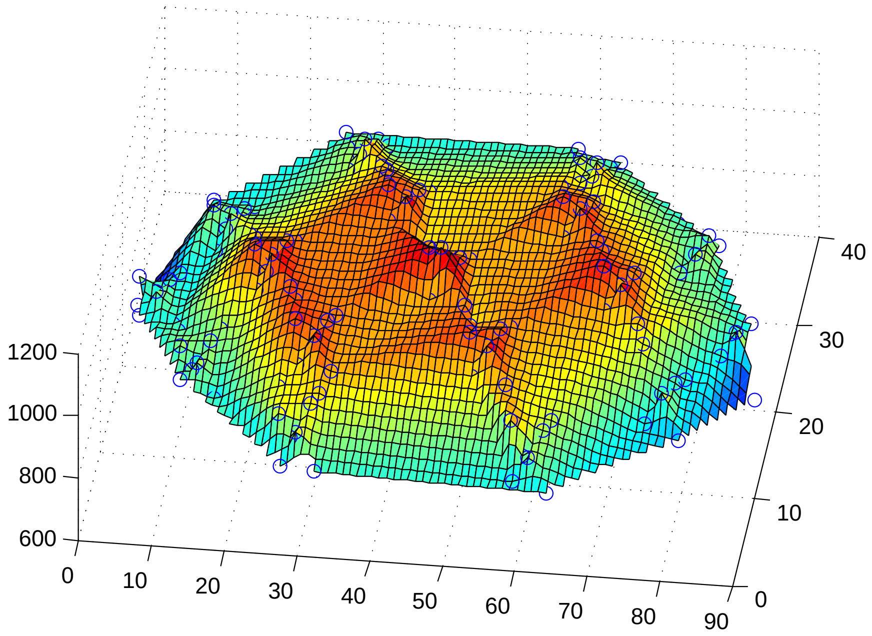


Figure 3.7: A baseline variation map of the measured data in $28nm$ technology, constructed from pointwise average for all wafers (i.e., blue circles in the figure), followed by an interpolation using Matlab's *griddata* function [12].

We estimate the relative magnitudes of across-field and across-wafer variations.

[3] Interpolation is used to fill the empty, i.e., unmeasured, area on the wafer.

For across-wafer variation, we calculate standard deviation of the measured values for the same location in each field, across all wafers. We then find the arithmetic average of the standard deviations of all locations in a field. The average standard deviation across wafer is computed as 59.7. For across-field variation, we calculate standard deviation of the measured values for all locations in each field, and then take an arithmetic average for the standard deviations of all fields in all wafers. The average standard deviation across field is computed as 37.4. From this result, we can conclude that across-wafer variation is more severe than across-field variation. This suggests that it would be difficult to find a "common" systematic across-field variation model applicable to all fields, since the across-field variation varies significantly according to the location of the field in a wafer. Hence, we find the systematic variation for a wafer without separately modeling systematic across-field variation. We evaluate the following modeling techniques with the given data set.

- Method 1. A quadratic function (i.e., $V_{w,s}(x, y) = a_1x^2 + a_2y^2 + a_3x + a_4y + a_5$) is used to fit the variation of the wafer.

- Method 2. A quadratic function (i.e., $V_f(x, y) = a_1x^2 + a_2y^2 + a_3x + a_4y + a_5$) is used to fit the variation of each field in the wafer. Least-squares regression is performed for each field, and the variation of the wafer is modeled as a set of across-field variation models.

- Method 3. *Virtual probe* (VP) [119] is used to fit the variation of each field in the wafer. VP is a compressed sensing-based modeling technique (see the detailed discussion in Section 3.3 below). The variation of the wafer is modeled as a set of across-field variation models.

- Method 4. Matlab *griddata* function is used to fit the variation of the wafer. Like spline interpolation, the function *griddata* can produce smooth and continuous surfaces for non-uniformly distributed data using cubic functions. The entire wafer can be modeled.

For all methods, we first find a "common" representative wafer by pointwise averaging. Each method is then applied on the (single) pointwise-averaged wafer. We then use k-fold cross-validation [133], a technique for assessing how the results of a statistical analysis will be generalized to an independent data set, to compare the effectiveness of the modeling techniques. In each round of cross-validation, $k - 1$ groups are used to train a model and one group is used to validate the model. The cross-validation continues until each group has been used as the validation set. Model quality for each round is measured by the *mean squared error* (MSE) $\sum (V_i^* - V_i)^2/N$ where V_i^* is the measured value in the validation data, V_i is the modeled value, and N is the number of measured values in the validation data. The average MSE for a given modeling technique across (over) all rounds is taken as a quality measure of that technique.

Table 3.1 shows the average MSE values for each modeling technique from the k-fold cross-validation, as well as the standard deviation (STDEV) of the remaining variation, assuming that the modeled variation is completely removed from the original data.

Table 3.1: Average mean squared error (MSE) from k-fold cross-validation, and standard deviation (STDEV) of remaining variations after removing the modeled variation from the original data.

Method	Average MSE	STDEV
Original	-	62.7
1	1418.2	37.7
2	981.8	31.3
3	672.1	25.9
4	631.3	25.1

From the results, we have the following conclusions.

- Systematic variation modeling and compensation can significantly reduce process variation. The original variation (σ_{orig}) is reduced from 59.7 to 37.7, even if we use a simple across-wafer variation model (Method 1).
- In Method 2, across-wafer variation should not matter in modeling, since each field is separately modeled. However, Method 2 still shows standard deviation

of 31.3 for the remaining variation, which is not much less than the 37.4 average standard deviation for the original field data. This may imply that the quadratic model is not sufficient to model across-field variation.

- Among the various modeling techniques, *griddata* shows the best accuracy, and VP also shows comparable accuracy. Using these two methods, the sigma-to-mean ratio can be reduced from 6.9% to 2.8%, which can reduce the process guardband by half assuming that systematic variation is compensated.

3.3 Variation Mapping Using Compressed Sensing

Many works address variation measurement and analysis as discussed in the previous section. Measurable parameters include critical dimension (CD) of polysilicon gate, saturation current ($I_{d,sat}$), off-state current (I_{off}), threshold voltage (V_{th}), ring oscillator (RO) frequency, etc. To accurately model the variation within a die, field or wafer, customized test structures, e.g., arrays of measurement structures, or on-chip testing/sensing units, have been used. For example, Agarwal et al. [31] use several array structures and test methodologies to measure I_{off} and V_{th}. Friedberg et al. [65] design CD test structures to capture variations in gate length. Although such controlled test structures may continue to be important to measure variation profiles, and will be used to capture major characteristics of manufacturing process, measuring CD or electrical characteristics of individual transistors requires a large amount of test time and cost, as well as valuable silicon area.

By contrast, variation information is naturally available in integrated circuits. Speedpath test measures the delay of a particular subcircuit (e.g., a timing-critical path) in a design, and is used to bin fabricated products based on measured speed or to diagnose potential timing failures [38] [47] [190]. The drawback of the speedpath test is high test cost. The number of paths measured by speedpath test per die is, in general, limited to small numbers, e.g., tens or hundreds [47]. However, for the purpose of modeling and compensation of the systematic variation, not every die needs to be tested, and the number of paths to measure for one or a few pilot dies can be increased without affecting test throughput. Chang et al. [51] introduce an efficient statistical timing-analysis algorithm that predicts the probability distribution of circuit delay considering both inter-die and intra-die variations. Visweswariah et al. [183] propose a canonical first-order delay model that takes into account both correlated and independent randomness. Mogal et al. [132] propose a technique to compute the statistical criticality information in a digital circuit under process variations. The advantage of using actual timing paths of the design to monitor systematic spatial variation is that there is essentially no additional design effort, and no waste of silicon area.

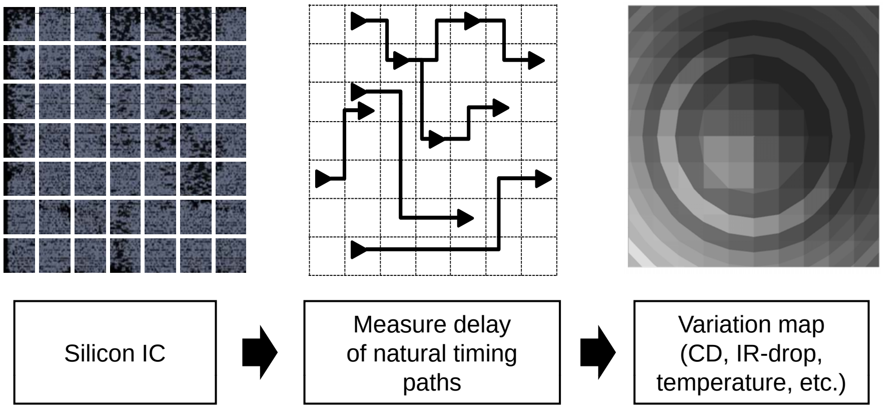

Silicon IC
Measure delay of natural timing paths
Variation map (CD, IR-drop, temperature, etc.)

Figure 3.8: Conceptual high-level flow of variation mapping from measured delay of "natural" timing paths to physical parameter variation.

Compressed sampling (Candes et al. [48]) and compressed sensing (Donoho et al. [62]) have been proposed for reconstruction of variations based on a substantially reduced number of measurements. Koushanfar et al. [111] estimate gate delay variation from timing path measurements using the wavelet transform. They show the effectiveness of their method on small-sized test circuits. Shamsi et al. [163] estimate leakage power based on compressed sensing. The concept of using natural timing paths and the modeling methods in [111] are well-matched to our work to be dicussed in this section. However, Koushanfar et al. directly model delay variation of individual gates, and do not consider mapping to a physical parameter variation; since one variable is assigned to each gate, their work requires a large number of measurements, e.g., a larger number of timing paths than the number of gates in the design, which is often impractical. Li et al. [119] propose another application of compressed sampling using discrete cosine transform (DCT) of variation to estimate the maximum frequency and leakage power of ICs placed across a wafer.

In this section, we propose a variation mapping technique that models systematic spatial variation of physical parameters from measured delay of “natural” timing paths; the concept of the proposed method is illustrated in Figure 3.8. We exploit the idea of compressed sensing with timing path measurement data, and we find a method for mapping measured delay variation to physical parameter variation. Based on the theory and application of compressed sensing, and the fact that the set of coefficients of the DCT of a specific variation is unique and sparse, we formulate circuit (e.g., natural timing path) delay as a function of the DCT coefficients, and find the DCT coefficients using a linear programming solver, starting from the measured delay values of the timing paths.

The DCT coefficients thus obtained can then be used to build variation maps that can be fed to advanced manufacturing optimization tools (cf. the Dose Mapper approach of [96] or the Zeiss CDC-Pixer technology [39]).[4] We verify the effectiveness of the proposed method with real circuits as well as artificial circuits under a range of variation assumptions with a linear delay model. We also discuss practically useful applications of the proposed method, including simultaneous modeling of multiple variation maps for the multiple IC layers in 3-D integration, and decomposition of the effects of multiple variation sources on timing delay variation.

3.3.1 Variation Mapping and Least-Squares

Given a 2-D gridded region and an assumption of correlated physical variation within the region, the objective is to find the specific variation within each grid. When measured values are available for each grid, the variation can be directly modeled using the measured values. When there are more measured data points than unknowns, least-squares regression can be used to find a unique fitting function with minimum (sum of squared) error. The accuracy of modeling highly depends on the number of samples taken as well as the degree of the polynomial used to represent the data. Lin-

[4] In our variation mapping work, we focus on CD variation modeling since CD is a physical parameter that strongly affects electrical performance, and CD variation exhibits systematic spatial characteristics due to periodic step-and-scan-based exposure systems. By contrast, e.g., V_{th} variation due to random dopant fluctuation may have weaker systematic dependence at wafer- and field-scales. Further, once the delay variation from unknown sources is mapped to equivalent CD variation, it is directly compensatable using recent CD control techniques such as Dose Mapper.

ear, quadratic [170] [65] [56] or higher-degree [138] polynomial functions are used to model systematic spatial variation. These methods directly model variation using measured values.

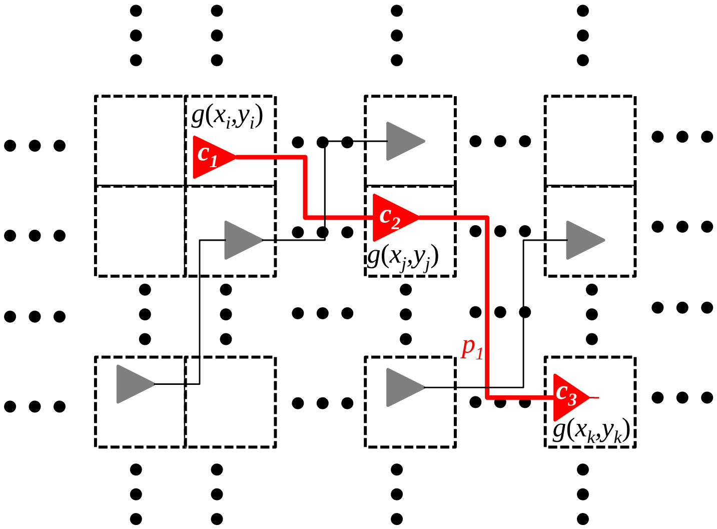


Figure 3.9: Example of natural timing paths in 2-D array. Path p_1 consists of three cells c_1, c_2, and c_3.

Least-squares can effectively determine the variation map with high accuracy. For a $P \times Q$ grid array and timing paths on grids as shown in Figure 3.9, one can express timing path delay using the variation value $g(x_i, y_i)$ at each grid $(x_i,\ y_i)$. For example, if we know that the variation follows a second-order polynomial, i.e., $g(x_i, y_i) = ax_i^2 + by_i^2 + cx_iy_i + dx_i + ey_i + f$, given at least six measured values $g(x_i, y_i)$ at (x_i, y_i), we can determine the six model coefficients. If we are instead given sums of variations of multiple grids, as with the delay of timing path p_1 in Figure 3.9, we can reformulate the linear system using linear combinations of rows. The delay variation of timing path p_1 is a sum of variation values of $g(x_i, y_i)$, $g(x_j, y_j)$, and $g(x_k, y_k)$. Hence, we can construct a new linear system $\mathbf{C} \times \mathbf{X} = \mathbf{D}$, for path p_1.

$$\mathbf{C}^T = \begin{pmatrix} x_i^2 + x_j^2 + x_k^2 \\ y_i^2 + y_j^2 + y_k^2 \\ x_i y_i + x_j y_j + x_k y_k \\ x_i + x_j + x_k \\ y_i + y_j + y_k \\ 1 + 1 + 1 \end{pmatrix}, \mathbf{X} = \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \end{pmatrix}$$

$$\mathbf{D} = \begin{pmatrix} g(x_i, y_i) + g(x_j, y_j) + g(x_k, y_k) \end{pmatrix}$$

With sufficiently many timing path measurements, the model coefficients can be found using least-squares, and from the model coefficients, we can determine the variation of each grid.

To verify the effectiveness of variation mapping using least-squares, we generate artificial timing paths on a $P \times Q$ grid, and apply an artificial CD variation to each grid. From the measured delay values of a set of timing paths under a given CD variation map, we test the ability to restore the given CD values for all grids. To this end, we have implemented a parameterized artificial circuit generator with the following input parameters.

- k: number of measured timing paths within the $P \times Q$ array
- n_{stage}: number of stages in each timing path
- r: radius of the bounding box of timing paths, in grid units
- o: order of the polynomial equation

On an 11×11 grid, we generate k timing paths that span different sets of grids. Each timing path consists of n_{stage} stages of logic cells. Delay values of the logic cells are chosen randomly from a discrete set of delay values, e.g., 100ps, 200ps, and 300ps, to mimic different cell delay values that result from different drive strength and load capacitance values. For each timing path, the location of the center of the path's bounding box is chosen randomly, and logic cells on the timing path are placed randomly within distance r of this center. When r = 1, all cells in the timing path are located in a single grid. The artificial circuit generator enables analysis of the sensitivity of model accuracy to the number of timing paths (k) and the model order (o). We perform experiments five times with different random seeds for each parameter, and find average root mean squared error (RMSE) and maximum error (ME) for the five runs. Experiments with a second-order CD variation assumption show that the variation can be successfully restored using only six timing paths – there are six unknowns in the second-order polynomial. Mean (RMSE) and maximum CD errors are negligibly small: 0.002528nm and 0.004nm, respectively.

Unfortunately, without *a priori* knowledge of the underlying variation model, or when the variation shape is too complex to be expressed using a simple polynomial functional form, the least-squares approach cannot accurately restore the CD variation map. Figure 3.10(a) shows a complex variation map with convex second-order variation in one half of the die, and concave second-order variation in the other half of the die. We evaluate the accuracy with increasing polynomial order and increasing number of timing paths. However, abrupt variation at the center is not correctly captured by the least-squares regression as shown in Figure 3.10(b). The maximum CD error in the reconstructed variation map does not go below 3nm even with a fifth-order polynomial model and 100 timing paths.

3.3.2 DCT-Based Compressed Sampling

We introduce an implementation of the compressed sensing proposed in [119] for *direct modeling* of variation to overcome the limitations of parametric regressions.

Compressed sampling is used for underdetermined systems, e.g., when we try to fit higher-degree models with very few data points. Using substantially fewer samples, virtual probe (VP) [119] shows significantly higher accuracy than naive 2-D interpolation. VP is based on properties of the discrete cosine transform (DCT). A finite sequence of data points can be expressed in terms of a sum of cosine functions oscillating at different frequencies, i.e., a DCT. Conversely, given the DCT of a variation map, we can find

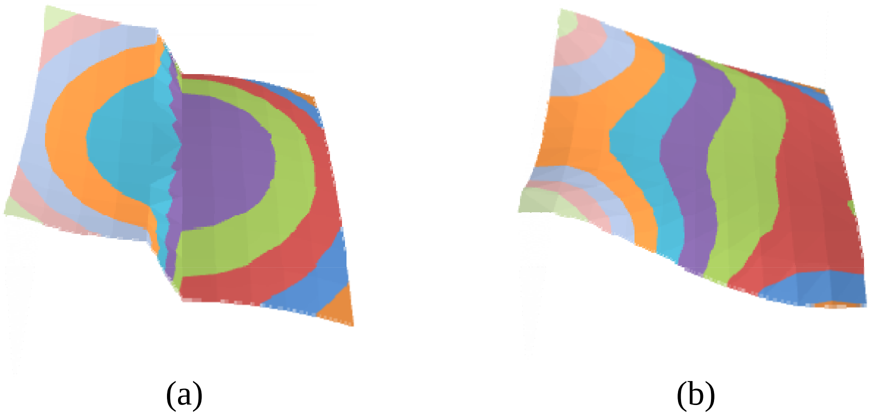


Figure 3.10: (a) Example of a complex variation map. (b) Restored variation map using least-squares regression.

values at any location using the inverse discrete cosine transform (IDCT). From sampled values in 2-D, 2-D images such as CD variation map in silicon or IR-drop/temperature variation map in a die can be restored using 2-D DCT and IDCT. We use the following notations in the rest of this section.

- P: number of rows for a uniformly-divided variation map
- Q: number of columns for a uniformly-divided variation map
- (x,y): a point in a variation map with $x \in \{1, 2, ..., P\}$ and $y \in \{1, 2, ..., Q\}$
- (u,v): a point in a frequency-domain map for the given variation map
- $g(x, y)$: a value within a variation map at a point (x, y)
- $G(u, v)$: a DCT coefficient at a frequency-domain point (u, v)
- M: number of measured values

DCT is defined as:

$$\begin{aligned} G(u,v) = \sum_{x=1}^{P}\sum_{y=1}^{Q}\Bigg\{ & \alpha_u \cdot \beta_v \cdot g(x,y) \cdot \cos\frac{\pi(2x-1)(u-1)}{2P} \\ & \cdot \cos\frac{\pi(2y-1)(v-1)}{2Q}\Bigg\} \end{aligned} \tag{3.1}$$

where α_u and β_v are defined as:

$$\alpha_u = \begin{cases} \sqrt{1/P}, & u = 1 \\ \sqrt{2/P}, & 2 \leq u \leq P \end{cases}$$

$$\beta_v = \begin{cases} \sqrt{1/Q}, & v = 1 \\ \sqrt{2/Q}, & 2 \leq v \leq Q \end{cases}$$

IDCT is defined as:

$$\begin{aligned} g(x,y) = \sum_{u=1}^{P} \sum_{v=1}^{Q} \Big\{ & \alpha_u \cdot \beta_v \cdot G(u,v) \cdot \cos \frac{\pi(2x-1)(u-1)}{2P} \\ & \cdot \cos \frac{\pi(2y-1)(v-1)}{2Q} \Big\} \end{aligned} \tag{3.2}$$

For a $P \times Q$ array, with PQ values of $g(x,y)$, the DCT coefficients can be uniquely determined by solving the following linear system, which corresponds to the set of IDCT equations in Equation (3.2).

$$\mathbf{A} \times \eta = \mathbf{B} \tag{3.3}$$

$$\mathbf{A} = \begin{pmatrix} A_{1,1,1} & A_{1,1,2} & \dots & A_{1,P,Q} \\ A_{2,1,1} & A_{2,1,2} & \dots & A_{2,P,Q} \\ \vdots & \vdots & \vdots & \vdots \\ A_{PQ,1,1} & A_{PQ,2,1,} & \dots & A_{PQ,P,Q} \end{pmatrix}$$

$$\begin{aligned} A_{m,u,v} = \alpha_u \cdot \beta_v \cdot \cos \frac{\pi(2x-1)(u-1)}{2P} & \qquad (3.4) \\ \cdot \cos \frac{\pi(2y-1)(v-1)}{2Q} & \\ \eta = [G(1,1)\ G(1,2)\ \dots\ G(P,Q)]^T & \qquad (3.5) \\ \mathbf{B} = [g(x_1,y_1)\ g(x_2,y_2)\ \dots\ g(x_m,y_m)\ \dots\ g(x_{PQ},y_{PQ})]^T & \qquad (3.6) \end{aligned}$$

However, with only M values of $g(x,y)$ when $M \ll PQ$, the linear system is underdetermined, so that there exists an infinite number of solutions. VP solves this underdetermined system using a maximum posterior estimation technique, based on the observation that DCT coefficients $G(u, v)$ tend to be zero with very high probability, so that the DCT coefficient matrix is sparse. This enables DCT coefficients to be found using a linear programming technique, with a much smaller number of samples required.

By solving the following linear program, $G(u, v)$ (= η_i) can be determined minimizing the sum of slack variable θ, and finally, VP can restore the original variation $g(x, y)$ using IDCT.

$$\begin{aligned} \text{minimize:} \quad & \theta_1 + \theta_2 + \cdots + \theta_{PQ} \\ \text{subject to:} \quad & \mathbf{A} \times \eta = \mathbf{B} \\ & -\theta_i \leq \eta_i \leq \theta_i, \quad (i = 1, 2, \ldots, PQ) \end{aligned} \tag{3.7}$$

The results in [119] show that VP can estimate variation using 4× fewer samples than traditional 2-D interpolation, while achieving the same accuracy.

3.3.3 CD Variation Modeling Using Timing Path Delays

VP is effective for direct variation modeling with a small number of variation measurements. However, VP requires a measurement point per grid basis. Hence, it is difficult to apply VP to measured delay values of timing paths that pass through multiple grids in a plane. To use VP with timing path delay data, each timing path should be enclosed within a grid. In reality, however, finding such timing paths is neither easy nor practical. Furthermore, the delays of such paths would in general be small, so that higher measurement precision would be required. To use measured delays of arbitrarily spanning timing paths, we utilize the concepts of VP, i.e., DCT, IDCT and linear programming, but formulate the linear equations differently.

Suppose that there is a $P \times Q$ array of grids with several timing paths (in the grids) as shown in Figure 3.9. We determine variation of all grids using a small number of measured delay values of timing paths. For example, we find the CD values of all $P \times Q$ grids from the measured delay values of k timing paths $\{p_i \mid 1 \leq i \leq k\}$.

Since $g(x, y)$ in VP is the variation map in a 2-D grid, without loss of generality, we assume $g(x, y)$ to be a CD error ΔCD, i.e., the difference between actual $CD_{x,y}$ at a grid point (x, y) and a nominal CD value CD_{nom}:

$$g(x, y) = \Delta CD_{x,y} = CD_{x,y} - CD_{nom} \tag{3.8}$$

Once we find $g(x, y)$, we can calculate the CD values of each grid point. To map measured delay values to CD values, we define a delay model with respect to a CD value. As shown in Figure 3.11, gate delay changes linearly with CD in a reasonable range of CD variation, e.g., from $60nm$ to $70nm$. Hence, we use a linear delay model as:

$$t_i = \gamma_i(CD_{x,y} - CD_{nom}) + t_{i,nom}, \tag{3.9}$$

where t_i and $t_{i,nom}$ represent actual and nominal delays of cell i that is placed at a grid (x, y), and γ_i is a coefficient of delay sensitivity to CD value.[5]

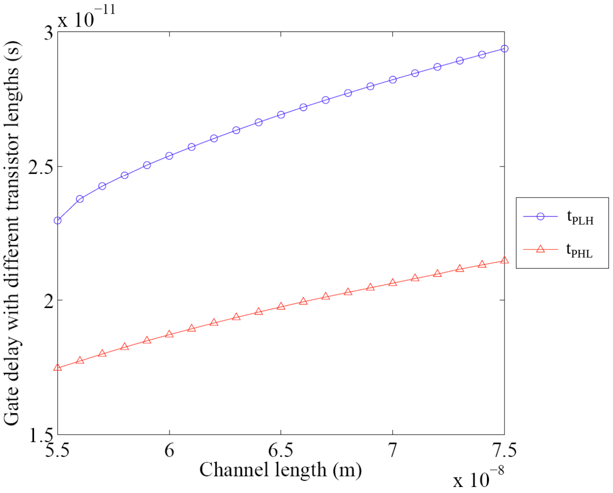


Figure 3.11: SPICE-calculated delay values versus CD in an inverter implemented in $65nm$ technology.

From Equations (3.8) and (3.9), we can rewrite the cell delay t_i in terms of CD ratio $g(x, y)$ as:

$$t_i = \gamma_i g(x, y) + t_{i,nom} \tag{3.10}$$

[5] However, different timing arcs can have different sensitivities, as shown in Figure 3.11. It is not trivial to find accurate sensitivity coefficients on a per-instance basis considering different input slews and output loads. Detailed processes are discussed in Section 3.4.5.

For the path p_1 in Figure 3.9, we have a measured delay value t_{p_1} of the path p_1, and t_{p_1} can be expressed as:

$$\begin{aligned} & t_{p_1} - t_{c_1,nom} - t_{c_2,nom} - t_{c_3,nom} \\ & = \gamma_{c_1} g(x_i, y_i) + \gamma_{c_2} g(x_j, y_j) + \gamma_{c_3} g(x_k, y_k) \end{aligned} \tag{3.11}$$

where $t_{c_1,nom}$, $t_{c_2,nom}$, and $t_{c_3,nom}$ are the known nominal delay values for cells c_1, c_2, and c_3, and γ_{c_1}, γ_{c_2}, and γ_{c_3} represent (different) precharacterized delay sensitivities to CD for each cell.

Linear combinations of rows in the matrix $\mathbf{B}$, weighted by γ_i of each cell, result in Equation (3.11). At the same time, the linear combinations of the rows in the matrix $\mathbf{A}$ with the same weights as used in the corresponding rows in $\mathbf{B}$ is also performed. We define a new linear equation, $\mathbf{C} \times \eta = \mathbf{D}$. Matrices $\mathbf{C}$ and $\mathbf{D}$ are defined as follows. Here we only show the elements for path p_1.

$$\mathbf{C}^T = \begin{pmatrix} \cdots & \gamma_{c_1} A_{r_1,1,1} + \gamma_{c_2} A_{r_2,1,1} + \gamma_{c_3} A_{r_3,1,1} & \cdots \\ \cdots & \gamma_{c_1} A_{r_1,1,2} + \gamma_{c_2} A_{r_2,1,2} + \gamma_{c_3} A_{r_3,1,2} & \cdots \\ \vdots & \vdots & \vdots \\ \cdots & \gamma_{c_1} A_{r_1,P,Q} + \gamma_{c_2} A_{r_2,P,Q} + \gamma_{c_3} A_{r_3,P,Q} & \cdots \end{pmatrix}$$

$$\mathbf{D}^T = \begin{pmatrix} \cdots & t_{p_1} - t_{c_1,nom} - t_{c_2,nom} - t_{c_3,nom} & \cdots \end{pmatrix}$$

where r_1, r_2, and r_3 denote the row indices of $g(x_i, y_i)$, $g(x_j, y_j)$, and $g(x_k, y_k)$ in matrix $\mathbf{B}$.

We calculate all elements in matrix $\mathbf{C}$ with the SPICE-characterized delay sensitivity and Equation (3.4). All elements in matrix $\mathbf{D}$ are known from measurements and from timing analysis with nominal CD. Hence, with a reasonable number of timing paths, the vector η (i.e., $G(u, v)$) can be calculated using the same linear programming approach in Equation (3.7), and we can restore $g(x, y)$ and thus $CD_{x,y}$ using IDCT.

Using the artificial circuit generator discussed in Section 3.4.1, we analyze the sensitivity of the model accuracy to the number of timing paths (k), the number of logic stages (n_{stage}), and the radius (r) of the bounding box of a timing path. Table 3.2 shows maximum error (ME) values, with respect to k, n_{stage}, and r. Observations from the experiments are summarized as follows.

- The number of timing paths is the main factor that affects the accuracy of the proposed model. We can see that maximum CD error decreases rapidly and saturates at less than $1nm$ from k = 20, for most cases. This implies that the proposed method can have less than $1nm$ CD error with 16.5% (= 20/121) of the number of timing paths required when solving the naive linear equations.
- Larger radius r of timing paths shows smaller error when the number of paths is small (less than 20). This implies that timing paths that are widely spread should be chosen when we can measure only a small number of timing paths.
- Larger number of stages may be preferred when the number of measurable timing paths is relatively small. However, when the number of timing paths is large, the number of stages of timing paths has little relationship with estimation accuracy.

We also apply the proposed method to the complex variation map in Figure 3.10(a). Figure 3.12 shows the restored CD variation map with 50 timing paths. ME and RMSE are $0.788nm$ and $0.31nm$, respectively.

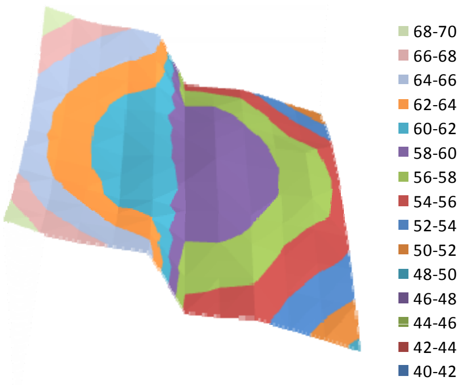


Figure 3.12: Restored variation map for the complex CD variation shown in Figure 3.10(a).

Table 3.2: Maximum CD error (nm) with respect to k, n_{stage}, and r for an artificial CD variation map of a single die.

n_{stage}=5	r					
k	1	3	5	7	9	11
5	7.22E+01	1.84E+01	1.67E+01	1.40E+01	1.09E+01	1.80E+01
10	1.52E+01	6.41E+00	4.32E+00	3.31E+00	5.45E+00	5.18E+00
15	9.29E+00	2.04E+00	1.42E+00	8.12E-01	9.00E-01	8.37E-01
20	8.21E+00	1.53E+00	3.56E-01	1.70E-01	6.29E-01	3.12E-01
25	2.94E+00	2.60E-01	1.08E-01	4.07E-02	7.94E-02	5.21E-02
30	1.21E+00	2.09E-08	2.02E-08	2.13E-08	2.46E-08	2.64E-08
35	5.68E-01	2.05E-08	2.04E-08	2.06E-08	2.14E-08	2.28E-08
40	5.28E-01	2.00E-08	2.06E-08	2.16E-08	2.20E-08	2.51E-08
n_{stage}=10	r					
k	1	3	5	7	9	11
5	7.22E+01	1.78E+01	1.53E+01	9.73E+00	1.15E+01	7.63E+00
10	1.52E+01	5.89E+00	3.84E+00	4.75E+00	3.72E+00	3.34E+00
15	9.29E+00	2.42E+00	1.11E+00	1.67E+00	1.28E+00	8.62E-01
20	8.21E+00	1.16E+00	6.23E-01	8.37E-01	2.86E-01	4.15E-01
25	2.94E+00	2.49E-01	2.10E-01	2.45E-08	2.33E-08	8.58E-02
30	1.21E+00	2.34E-08	2.27E-08	2.24E-08	2.48E-08	2.42E-08
35	5.68E-01	2.25E-08	2.56E-08	2.17E-08	2.70E-08	2.36E-08
40	5.28E-01	2.20E-08	2.36E-08	2.41E-08	2.18E-08	2.59E-08

3.3.4 Other Variation Modeling Applications

The proposed framework for the variation mapping discussed in the previous subsection can be extended to various applications. In this subsection, we provide two applications: (1) simultaneous multiple variation map modeling in 3-D stacking, and (2) decomposition of multiple variation sources.

CD variation modeling for 3-D die stacking. From a manufactured 3-D-stacked design, it is difficult to measure the variations of individual dies using direct probing or SEM. In addition, during stacking and through-silicon via generation, additional electrical variation occurs that cannot be monitored by inspection of each die. We can apply our proposed method to a 3-D design which consists of more than one stacked die. From the measured delay of timing paths that arbitrarily span across multiple dies, we find a set of DCT coefficients for each die, and restore the variation map of each die simultaneously.

For two stacked dies, we generate a new matrix $\mathbf{A}2$ that is a concatenation of two $\mathbf{A}$ matrices in Equation (3.3).

$$\mathbf{A2} = [\mathbf{A}\ \ \mathbf{A}] = \begin{pmatrix} A_{1,1,1} & \ldots & A_{1,P,Q} & A_{1,1,1} & \ldots & A_{1,P,Q} \\ A_{2,1,1} & \ldots & A_{2,P,Q} & A_{2,1,1} & \ldots & A_{2,P,Q} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{PQ,1,1} & \ldots & A_{PQ,P,Q} & A_{PQ,1,1} & \ldots & A_{PQ,P,Q} \end{pmatrix}$$

We also generate a vector matrix $\eta 2$ that contains all DCT coefficients of both dies by vertically concatenating DCT coefficient vectors η_1 and η_2.

$$\eta\mathbf{2} = [\eta_\mathbf{1}^\mathbf{T}\ \ \eta_\mathbf{2}^\mathbf{T}]^\mathbf{T} = [G_1(1,1)\ \ldots\ G_1(PQ)\ G_2(1,1)\ \ldots\ G_2(PQ)]^T$$

Then, linear combinations of rows of $\mathbf{A}2$ with delay sensitivity (γ) values results in new matrices $\mathbf{C}$ and $\mathbf{D}$. Finally, a linear programming instance as in Equation (3.7) is formulated and solved.

We verify the accuracy of individual die variation mapping for 3-D stacking. Figure 3.13 shows assumed underlying variation maps of the two stacked dies. We use second-order polynomials for each variation map: the variation map for the first die has a convex shape with a minimum value of 50nm at the center and a maximum value of 60nm at the four corners, and the variation map for the second die has a concave shape with a minimum value of 60nm at the four corners and a maximum value of 70nm at the center. We assume that each die is discretized into $P\times Q$ grids, and generate k timing paths that span different subsets of the two grid planes. Each timing path consists of n_{stage} stages of logic cells placed within r distance from the center of each path bounding box. Each cell instance is assigned at random to either the first or the second die. Table 3.3 shows maximum CD error of both dies with respect to different k and r values, with n_{stage} = 5. Using 50 timing paths with r = 5 and n_{stage} = 5, mean (RMSE) and maximum CD errors from both dies are 0.259nm and 0.739nm, respectively. We can conclude that the proposed approach successfully reconstructs each of the given CD maps of the two dies using randomly placed timing paths.

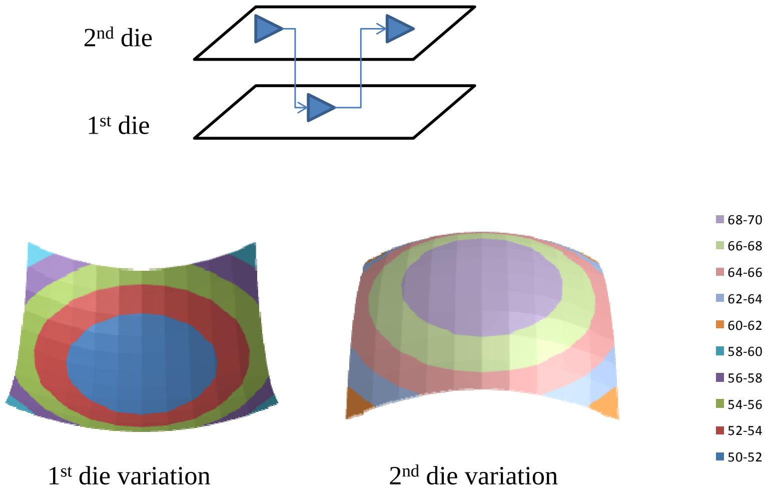


Figure 3.13: 3-D stacking with two dies (upper). Second-order CD variation map on an 11×11 grid is assumed for each die (lower).

CD variation modeling of multiple underlying variations. The measured delay variation results from a mixture of many model parameters. For example, suppose that there are independent CD and interconnect variations in a design. The measured delay is a single value; however, the delay of timing paths in the design is affected by both the CD and interconnect variations. Delay from larger CD and smaller interconnect capacitance can be similar or exactly the same as the delay from smaller CD and larger interconnect capacitance. Hence, it is difficult to determine whether the delay variation

Table 3.3: Maximum CD error (nm) with respect to k, n_{stage}, and r for artificial CD variation maps of two stacked dies.

n_{stage}=5	r					
k	1	3	5	7	9	11
10	2.68E+01	2.91E+01	2.50E+01	1.31E+01	1.15E+01	3.37E+01
20	1.06E+01	7.78E+00	5.59E+00	8.50E+00	6.35E+00	7.87E+00
30	8.78E+00	5.05E+00	3.71E+00	2.42E+00	3.64E+00	3.51E+00
40	7.64E+00	2.42E+00	2.01E+00	1.74E+00	2.38E+00	2.33E+00
50	7.00E+00	1.31E+00	9.21E-01	1.26E+00	1.30E+00	1.25E+00
60	5.50E+00	4.27E-01	2.64E-01	2.84E-01	3.23E-01	6.51E-01
70	3.69E+00	1.78E-01	3.97E-02	3.29E-02	4.39E-02	3.51E-01
80	2.76E+00	9.45E-02	5.86E-09	6.14E-09	5.96E-09	2.25E-01
90	2.31E+00	6.94E-09	5.30E-09	1.15E-08	8.54E-09	2.25E-01
100	1.74E+00	4.26E-09	5.30E-09	1.25E-08	1.21E-08	2.25E-01

comes from CD or interconnect capacitance variation. When we assume a linear delay model, where length-based interconnect delay is added to the path delay calculation, the proposed method can decompose the CD and interconnect variation from the measured path delay values.

In this application, the formulation is similar to that of the two-die example. We generate the $\mathbf{A}2$ matrix (resp. $\eta\mathbf{2}$) by concatenating $\mathbf{A}$ matrices (resp. η) for each CD and interconnect capacitance. However, linear combination of rows to form $\mathbf{C}$ is complicated. For each grid through which an interconnect passes, we use a π model as shown in Figure 3.14. For clarity of the presentation of the delay expression in the following example, we assume a cell is placed at the center of a grid, so that the length of the interconnect in the grid connected to the cell is assumed to be half of the grid size. In practice, it would be straightforward to use actual cell locations and to segment routing segments at grid boundaries.

For a two-stage path p where a driver cell c_1 at grid I is connected to a cell c_2 at grid J using an interconnect passing through grids I, J, and K, the element of the row in $\mathbf{C}$ corresponding to this path is calculated using the Elmore delay model as:

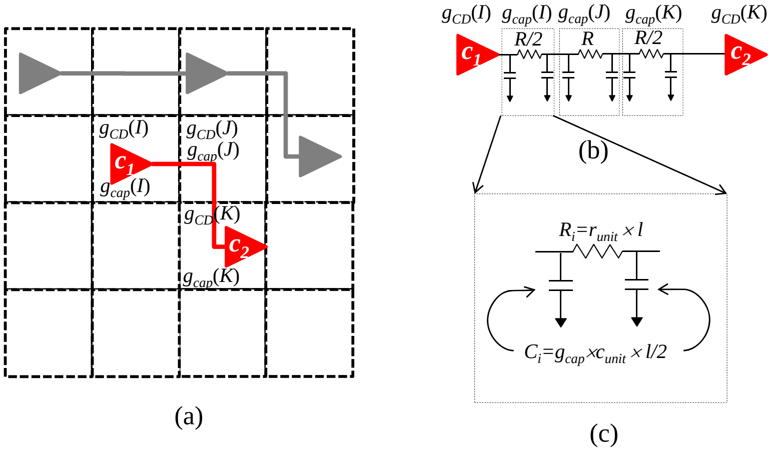


Figure 3.14: Interconnect model: (a) a timing path, (b) grid-based segmentation of interconnect, and (c) interconnect model for a grid.

$$
\begin{aligned}
C_{p,u,v} &= \gamma_{c_1} A_{I,u,v} + \left(r_{unit}\frac{l}{2}\right)\left(A_{I,u,v}\frac{c_{unit}}{2}\frac{l}{2}\right) \\
&+ \left(r_{unit}\frac{l}{2}\right)\left(A_{J,u,v}\frac{c_{unit}}{2}l\right) + \left\{\left(\frac{r_{unit}}{2} + r_{unit}\right)l\right\}\left(A_{J,u,v}\frac{c_{unit}}{2}l\right) \\
&+ \left\{\left(\frac{r_{unit}}{2} + r_{unit}\right)l\right\}\left(A_{K,u,v}\frac{c_{unit}}{2}\frac{l}{2}\right) \\
&+ \left\{\left(\frac{r_{unit}}{2} + r_{unit} + \frac{r_{unit}}{2}\right)l\right\}\left(A_{K,u,v}\frac{c_{unit}}{2}\frac{l}{2}\right) \\
&+ \gamma_{c_2} A_{K,u,v} \qquad (3.12)
\end{aligned}
$$

where r_{unit} and c_{unit} are unit-length interconnect resistance and capacitance, length l is the grid size (unit step), and $A_{I,u,v}$, $A_{J,u,v}$ and $A_{K,u,v}$ are elements in rows I, J, and K in the $\mathbf{A}$ matrix, respectively.

We verify the accuracy of the CD and interconnect variation decomposition. We generate k timing paths by randomly choosing the placement locations of cell instances. We also apply interconnect variation between cell instances. The interconnect variation is assumed as capacitance variation in a slanted shape as shown in Figure 3.15(a). CD variation is assumed as a second-order polynomial with a minimum of $55nm$ and a maximum of $65nm$ within a die. For the capacitance variation, we assume $\pm 10\%$ from the nominal value. We use R_{unit} = 70Ω and C_{unit} = $0.4pF$, respectively, for $50um$-length interconnect in a typical $65nm$ technology. We observe maximum errors of 1.48% and

1.06% for CD and interconnect capacitance, respectively, with 35 paths spread on 5 grids (r = 5). The restored capacitance variation map is shown in Figure 3.15(b).

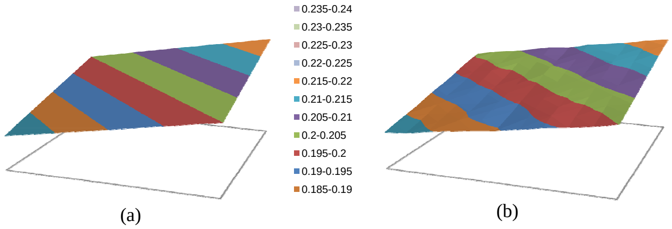


Figure 3.15: Given interconnect capacitance variation map in (a) and restored interconnect variation in (b). Minimum and maximum variations from the nominal capacitance of $0.2pF$ are assumed to be -10% at location (1,1) and +10% at location (11,11), respectively.

3.3.5 Experiments with Real Design and Libraries

We have applied the proposed method to a real design with an industry library. We implement an open-source core, *AES*, obtained as an RTL netlist from *opencores.org* [15]. We synthesize, place and route the core using *Cadence RTL Compiler v5.2* and *Cadence SOC Encounter v7.2* with a subset (50 combinational cells + 2 sequential cells) of an TSMC 90nm library characterized with nominal CD of 90nm. The number of cell instances in the design is $22K$, and the size of the design is 500$\mu m \times$500μm.

To model timing variation due to CD variation, we characterize timing by changing the gate length of the original 90nm transistors in the CDL SPICE netlist of each given cell master, using *Cadence Signal Storm v6.1*. The prepared CD variation libraries have the naming convention $N05$, ... $P00$, ... $P05$ with respect to ΔCD = -5nm, ... 0nm, ..., +5nm; suffixes of master cell names in each library use the same naming convention. For instance, the master cell name of the two-input NAND gate characterized with -3nm CD variation is NAND2_N03.

From the SPICE-characterized libraries, we calculate precise delay sensitivity (γ) to CD for each timing arc in each master cell, for each slew and load combination in the timing table, since the sensitivities to CD are different for the cells, timing arcs, input slew and load capacitance combinations. Each timing arc delay for each slew and load combination is modeled as a linear function of ΔCD as Equation (3.9). The sensitivity (γ) and the nominal delay for each timing arc are obtained using linear regression across all CD variation libraries, and stored in 2-D tables indexed by input slew and load capacitance. This sensitivity calculation for the 50 combinational cells takes 3.12 seconds on an Intel Xeon 3.33GHz Linux system.

We discretize the design into $P \times Q$ grids. Under a CD variation assumed on the grid, i.e., a second-order polynomial having minimum 85nm and maximum 95nm,[6] we assign CD values for all grids according to the CD variation model, assuming perfect spatial correlation within each grid. Then, according to the assigned CD value in each grid, the corresponding master cell characterized at the assigned CD variation value is instantiated for each cell instance within the grid. For instance, if a NAND2 instance is placed in a grid that has -3nm CD variation, the master of that cell instance is set to NAND2_N03.

From static timing analysis using *Synopsys PrimeTime vC-2009.06-SP2*, we obtain critical paths that span different sets of grids,[7] and the logic cell instances in the timing paths with corresponding signal direction, input slew and load capacitance values are stored. The nominal delay values and sensitivity (γ) values of timing arcs in the timing paths are calculated from the precharacterized sensitivity and nominal delay tables using linear interpolation between the nearest slew and load indices in the tables from the stored slew and load values.[8] The linear interpolation is performed using *GNU octave*. The surface consisting of values at four slew-load corners is fitted as Ax + By + Cxy + D where x and y are slew and load values, and A, B, C, and D are model coefficients.

From the nominal delay and sensitivity values, we formulate each path delay as a function of DCT coefficients, and find the DCT coefficients using linear program-

[6] Koushanfar et al. [111] assume total of 12% random variation consisting of correlated intra-die variation (60%), uncorrelated intra-die variation (20%), and inter-die variation (20%). Those variations are summed and appear as an overall variation within a die. The given variation in the experiments approximates the overall variation.

[7] The paths are not necessarily independent from each other. Components of timing paths, i.e., cell instances and nets, can be shared by the selected paths.

[8] For the stored slew (s) and load (l) values of a timing arc, the four nearest slew-load indices (s_i, l_i), (s_{i+1}, l_i), (s_i, l_{i+1}), and (s_{i+1}, l_{i+1}), where $s_i \leq s \leq s_{i+1}$ and $l_i \leq l \leq l_{i+1}$, are used for purposes of interpolation.

ming. The linear programming instance is solved using *ILOG CPLEX* [10]. Finally, we compare the given CD variation and model CD variation from DCT coefficients.

Table 3.4: Root mean squared error (RMSE) (nm) and maximum error with respect to the number of timing paths (k) in AES design.

k	Grids			
	11×11		21×21	
	RMSE (nm)	ME (nm)	RMSE (nm)	ME (nm)
10	1.13	3.53	2.23	7.59
20	0.92	2.88	1.00	3.93
30	0.50	2.02	0.87	3.48
40	0.36	1.15	0.74	2.81
50	0.27	0.98	0.70	2.48
60	0.20	0.78	0.45	1.41
70	0.18	0.78	0.41	1.41
80	0.10	0.32	0.46	1.77
90	0.09	0.28	0.36	1.19
100	0.00	0.00	0.34	1.07

Table 3.4 shows model accuracies in terms of RMSE and ME with respect to the number of used timing paths. We use two different grids, 11×11 and 21×21, to see the dependency of the model accuracy on granularity of grids. The levels of RMSE and ME for an 11×11 grid are similar to those with artificial circuits using an 11×11 grid. For the finer-grain 21×21 grid, the CD error is larger than that of an 11×11 grid. However, the error decreases to $\sim 1nm$ as the number of timing paths reaches 100. From the table, we can conclude that the proposed modeling method successfully maps the measured delay to the actual CD variation under real designs and libraries, given a reasonable number ($\sim$100) of measurements.

3.4 Conclusions and Research Directions

We have reviewed recent variation modeling techniques and proposed a novel variation mapping framework that reconstructs the details of multiple, simultaneously occurring systematic variation maps from measurements of natural timing paths in a design. Using the sparsity in DCT coefficients of a variation map, we formulate delay of natural timing paths as a function of the DCT coefficients, and find the DCT coefficients using a linear programming solver, with a small number of measured delay of the timing paths. We have also explored potential useful applications of the proposed method: (1) modeling of multiple variation maps for the multiple IC layers in 3D integration, and (2) decomposition of the effects of multiple variation sources on timing delay variation.

We have verified the effectiveness of the proposed method with artificial circuits under a range of variation assumptions and with a linear delay model. We have also applied the proposed method to a real design, with detailed timing tables considering different slew and load conditions, characterized with a range of discrete CD values.

Future research seeks to (1) use more accurate delay models considering slew and load propagation that have not been completely comprehended in the reported delay model, (2) provide a solution to decompose various levels of variation hierarchies, e.g., decomposition of intra-die variation from intra-wafer variation using timing path measurements of multiple dies, and (3) develop manufacturing recipes and design/design process optimization methodologies that incorporate the accurately modeled variation.

3.5 Acknowledgments

Chapter 3 is in part a reprint of “Variation Mapping From Timing Path Delay Measurements Using Compressed Sensing”, *Proc. ACM International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems*, 2011. I would like to thank my coauthor Professor Andrew B. Kahng.

I also thank Yorick Trouiller, Alexandre Villaret and Thierno Tahir Diallo of STMicroelectronics for a number of motivating discussions, and Professor Farinaz Koushanfar of Rice University for clarification of the spatial variation modeling in [111].

Chapter 4

Variability Assessment in Advanced Lithography

As Moore's Law continues to drive higher performance with smaller circuit features, lithography is being pushed to new extremes. Projection optical lithography at $193nm$ with advanced *resolution enhancement techniques* (RETs) and immersion is expected to satisfy the needs of the $45nm$ node. However, for $32nm$ node patterning, the availability of options such as *extreme ultraviolet* (EUV) and *immersion ArF* (IArF) [53] at $157nm$ remains unclear. An EUV imaging system is composed of mirrors coated with multilayer structures designed to have high reflectivity at $13.5nm$ wavelength. Hence, there are many technical hurdles for implementing EUV lithography in terms of mask blank fabrication, high output power source, resist material, etc. IArF requires truly high-index fluids (NA = 1.55 $\sim$ 1.6), with corresponding advances in high-index resists and optical materials. In addition, although EUV and IArF can successfully generate $32nm$ patterns, economic cost must be considered in the adoption of these technologies.

Double patterning lithography (DPL) [63] partitions a critical-layer layout into two mask layouts and exposures, each with relaxed critical pitch and spacing. DPL incurs a throughput overhead, and necessitates tight overlay control between the two exposures. However, DPL provides an attractive alternative or a supplementary method to enable the $32nm$ and $22nm$ process nodes, relative to costlier technology options such as high refractive index materials, EUV, or e-beam lithography.

In this chapter, we first introduce various types of DPL techniques in Section

4.1. We then analyze the impact of DPL on back-end-of-line (BEOL) and front-end-of-line (FEOL) variabilities in Sections 4.2 and 4.3, respectively. We also describe potential courses of action for the industry as standards and flows that address the additional variabilities in DPL. Finally, Section 4.4 presents focus-exposure process window comparison between a traditional single-exposure lithography and a DPL technique *interference-assisted lithography* (IAL) through lithography simulations.

4.1 Double Patterning Lithography (DPL)

In this section, we first discuss various options in implementing DPL and present potential variability issues that arise with adoption of DPL.

4.1.1 DPL Options

The ITRS [11] cites three general DPL options:

- **Double exposure (DE).** DE consists of two successive exposures followed by a single etch. To suppress interference between exposures, a ‘freezing’ step may be used, resulting in a so-called litho-freeze litho-etch (LFLE) process [55] [91].

- **Double patterning (DP).** DP also requires two exposure steps, but each exposure is followed by an etch step, resulting in a *litho-etch litho-etch* (LELE) process. The main difference with respect to DE is the use of an additional etch step for a hardmask. This increases the number of process steps, but reduces interference between two exposures.

- **Self-aligned double patterning (SADP).** After one exposure and etch on a hard mask, pattern doubling is accomplished by spacer formation and a second etch. The main idea of SADP is to utilize the space between a first set of printed features, dimensions of which are bloated by spacers. SADP is regarded as a viable option for regular poly gates [167] and bitlines [93] in recent memory processes.

Double exposure (DE) and double patterning (DP) can be applied with either positive or negative photoresist processes. With positive photoresist in copper dual-damascene interconnect, trenches are patterned on the area exposed by light following the mask image, while with negative photoresist, trenches are patterned on the area not exposed by light. In this thesis, ‘P-DE’ or ‘P-DP’ denotes the former and ‘N-DE’ or ‘N-DP’ denotes the latter, following the type of photoresist.

With an SADP, mask patterns are transferred onto the hard mask, but the hard mask itself does not convey the target patterns directly. Generated spacers, which define the drawn patterns (space or line), act analogously to photoresist. Hence, similar to the DE/DP processes, we define ‘P-SADP’ as the process that generates trenches in the area not under the spacers, and ‘N-SADP’ as the process that generates trenches in the area under the spacers.

Double exposure lithography. Double exposure (DE) with negative-tone resist creates trenches at twice the resolution of normal lithography, using two successive exposure steps. Double exposure DPL prints spaces rather than target line shapes, and is hence called a *negative dual trench process* [64]. One edge of a target line is formed with the first exposure, and the other edge is generated with the second exposure, as shown in Figure 4.1. Both edges of two adjacent lines facing each other are formed at the same time. While an exposure dose variation can result in an edge placement error, both lines will be affected by the same amount, and critical dimensions (CDs) of adjacent lines remain identical, as shown in Figure 4.1(b). However, in the presence of overlay, CDs of adjacent lines can differ by the amount of the overlay error, as shown in Figure 4.1(c).

We note that while double exposure DPL entails a relatively simple process, the fact that CDs are determined by overlay error reduces the technique’s viability. This is because the roadmap for overlay control capability is significantly looser than the general CD control requirement (e.g., the 2007 ITRS specifies overlay tolerance at the 45nm node to be as large as 9nm [11]).

Double patterning lithography. Double patterning (DP) with positive-tone resist creates lines at twice the normally achievable resolution, using a LELE process. At the first etch step, the patterns of the first resist layer are transferred to an underlying hard mask. Photoresist is then coated onto the surface remaining after the first process, and exposed in the second exposure step. The flow finishes with the hard mask that prints one line and the resist of the second exposure that prints the other line. In double patterning, the two edges of a line that are printed by the first etch and the second exposure, and the two

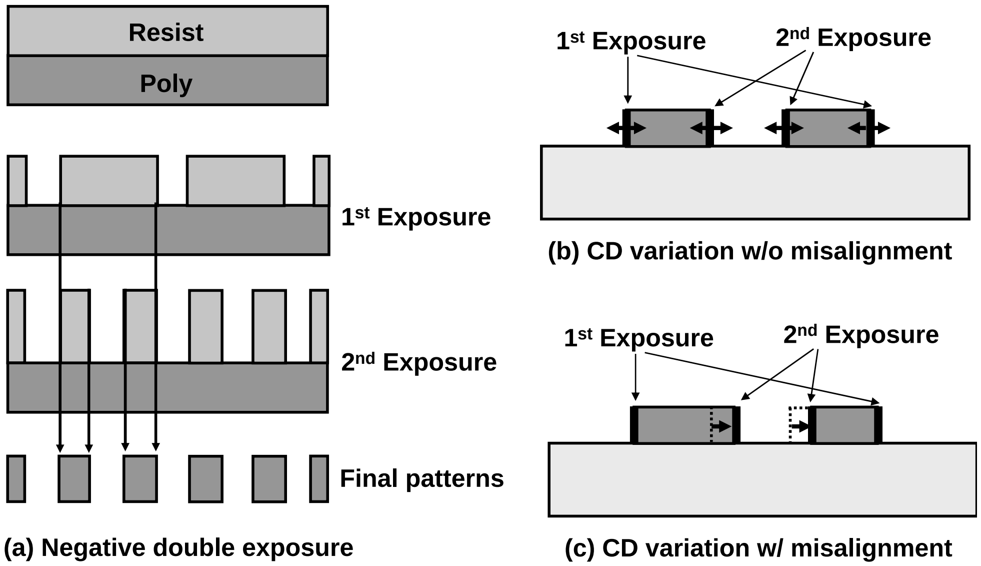


Figure 4.1: An implementation of double exposure (DE) lithography.

edges of the adjacent line that are printed by the second exposure and etch process, can be different, as shown in Figure 4.2(a). While the first patterns are made on a perfectly flat wafer, the second resist is coated onto a topography that is a result of overetch of the first patterning step. The topography implies greater depth of focus (DOF) variation, so that the CDs between the first and the second patterns can differ. Plasma exposure of the first line during the second etch could additionally cause CD change [64].

Unlike the double exposure, double patterning will have two different CD populations due to the CD control error, as shown in Figure 4.2(b). Since overlay just shifts the line, without changing the linewidth, overlay itself does not matter; this is illustrated in Figure 4.2(c).

Self-aligned double patterning lithography. The third DPL technology option [108] [129] uses sacrificial spacer technology. First, mandrel (or core) patterns are implemented on a hard mask. Second, spacers are formed at all sides of the mandrel patterns. Then, block or cut mask patterns modify the spacer patterns to the actual target patterns on a silicon wafer. Depending on whether spacers define lines or dielectric, SADP has two possible implementation options.

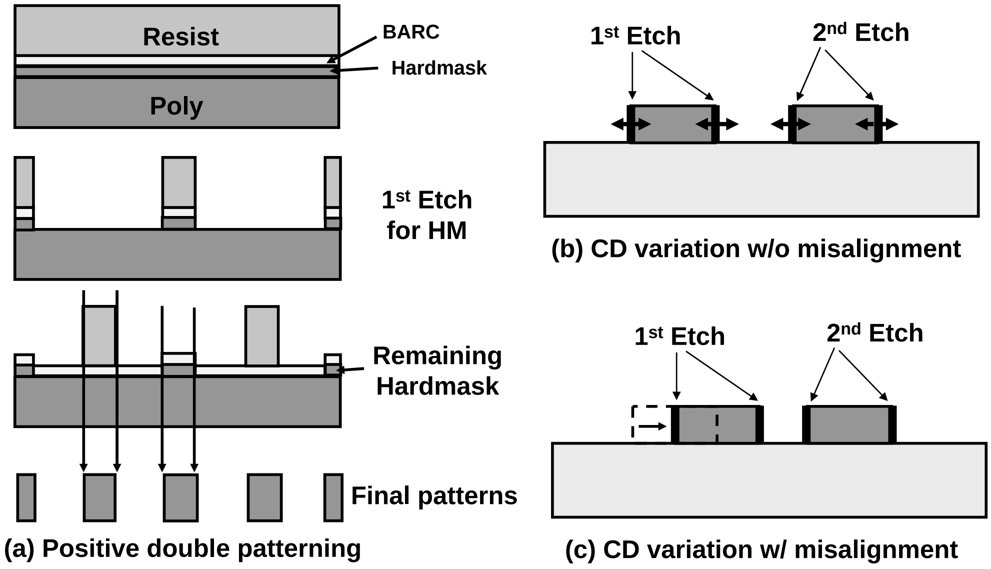


Figure 4.2: An implementation of double patterning (DP) lithography.

- *Spacer is dielectric (SID).* Figure 4.3 illustrates an SID-type SADP process. For BEOL layers, spacers define dielectric, and the region not under the spacers is filled with metal.

- *Spacer is metal (SIM).* Figure 4.4 illustrates an SIM-type SADP process. For BEOL layers, spacers define metal lines, and the region under the spacers is filled with metal.

Given a well-controlled spacer formation and etch process, the CD difference between adjacent lines can be maintained to be as small as the CD control capability. Since SIM-type SADP process is more complex due to additional processes and material combinations, i.e., tone reversal, gap filling, etch back, oxide spacer removal, etc., SID-type SADP is expected to be adopted. With the use of second block (or cut) mask patterns, SADP can implement various pitches and spaces as well as complex 2-D patterns.

Interference-assisted lithography. Interference-assisted lithography (IAL) is a kind of double exposure lithography consisting of two exposures – maskless two-beam or four-beam interference lithography (IL), and projection lithography (PL). IL produces high contrast, regular and high-resolution dense grating patterns, but this step does not capture detailed circuit geometries. PL is then used to erase a subset of the IL grat-

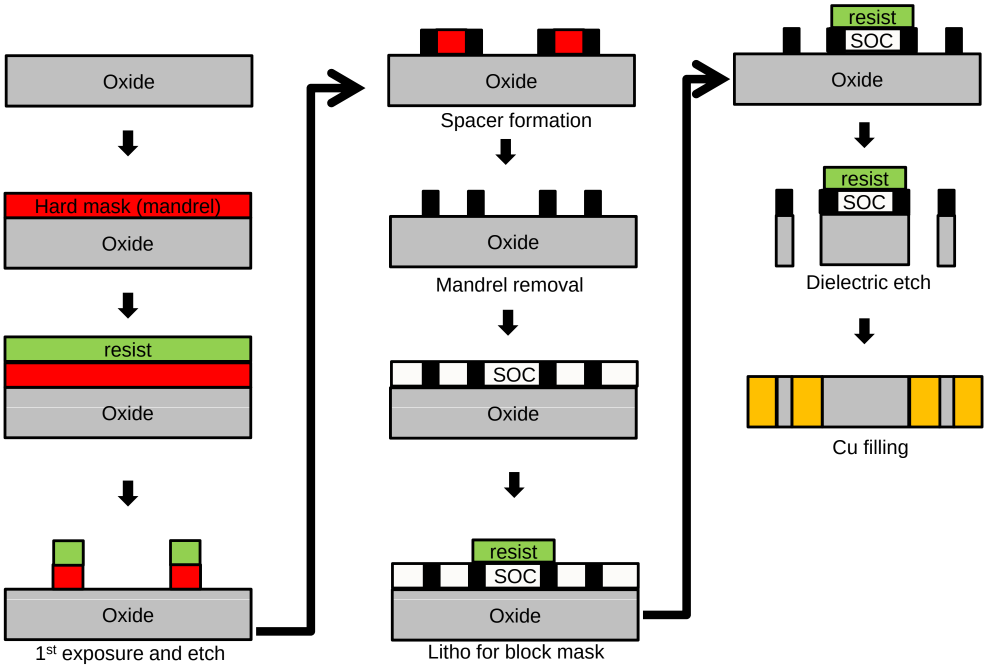


Figure 4.3: An implementation of SID-type SADP.

ing patterns so that only actual design geometries remain on the resist. Figure 4.5 shows the 32nm design patterns obtained by Fritze et al. [66] from combining the high-resolution interference with lower-resolution projection lithography. The key advantage of IAL is the generation of high-contrast images with a maskless step. Since the high-resolution patterns can be generated without a mask, mask cost can be significantly reduced. The second exposure that trims unnecessary parts of the grating patterns can use older-generation projection systems and photomasks, which have significantly lower cost due to lower complexity of optical proximity correction (OPC).

4.1.2 Taxonomy of Sources for Additional Variability in DPL

According to semiconductor equipment and materials international (SEMI) standard P19 [161], *linewidth* is defined as, at a given cross-section of the line, the distance between the airline material boundaries at some specified height above the interface between the patterned layer in which the line is formed and the underlying layer. *Linespace* can be defined similarly, except the distance is measured between two lines. In this

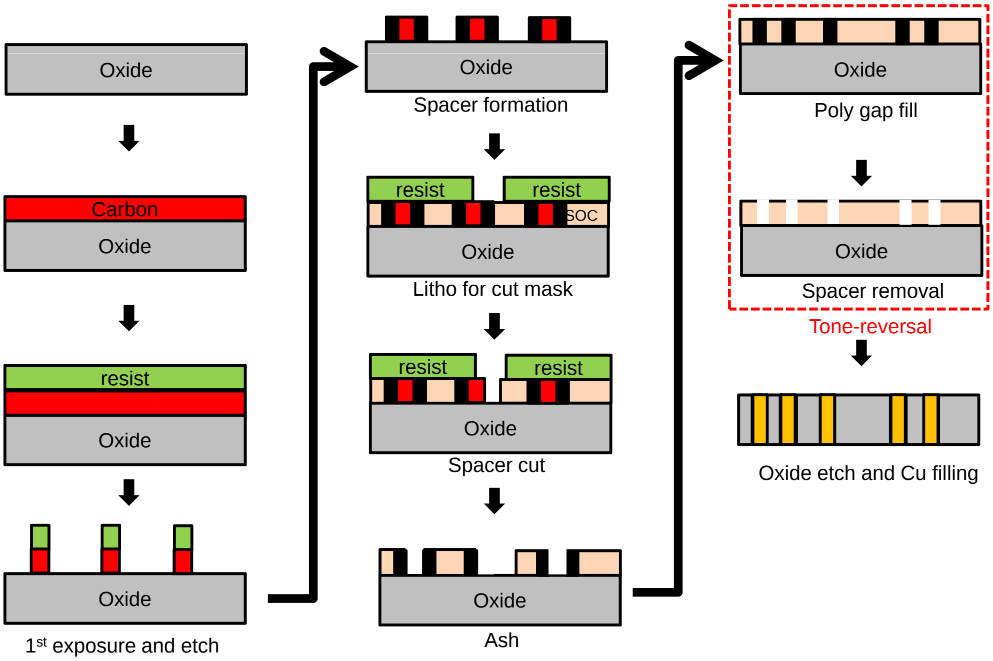


Figure 4.4: An implementation of SIM-type SADP.

subsection, we discuss sources of the linewidth and linespace variations in DE/DP and SADP.

Sources of linewidth and linespace variation in DE/DP. In DE and DP, *overlay* causes linespace and linewidth variations. The sources of overlay are alignment error due to poor optics in the alignment system, including reticle-to-tool alignment error and reticle-to-wafer alignment error; stepper-induced field errors including lens distortion, magnification and reticle rotations; wafer expansion or contraction; mask error; and

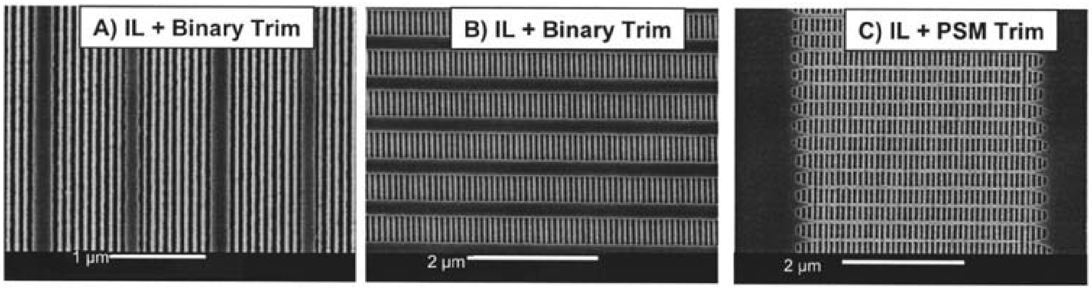


Figure 4.5: Experimental results of a dual-resist IAL reported by Fritze et al. [66].

nonlinear wafer deformation due to high temperature or stress in film deposition [71].

According to SEMI standard P18 [160], *overlay* is a vector quantity defined at every point on the wafer; it is the difference between a vector position in a substrate geometry and a vector position in an overlaying pattern. Linewidth and linespace variations due to overlay result in reliability and defect yield problems (open/short faults or electromigration-induced defects due to narrower overlaps between contacts and under-/overlying layers, etc.), as well as performance variations that cause loss of parametric yield.

One of the major sources of overlay is misalignment. SEMI standard P18 [160] defines *alignment* as the mechanical positioning of reference points on the wafers ("alignment targets") to the corresponding points on the reticles. The measure of alignment is the overlay at the position on the wafer where the alignment targets are placed. Overlay in DPL can be measured and controlled in two ways, according to the alignment reference point: (1) indirect alignment (IA), and (2) direct alignment (DA). With DE or DP, (1) IA aligns the two masks for a given layer to a reference point in the underlying layer, while (2) DA aligns the second mask to the first mask [115]. IA and DA are illustrated in Figure 4.6, where dashed lines indicate alignment, and the reference layer may be a layer that has already been manufactured, such as a shallow trench isolation layer, or an inter-layer dielectric layer. IA is expected to have $\sqrt{2}$ times larger pattern shifts within a layer due to the two independent alignments.

Sources of linewidth and linespace variation in SADP. In addition to overlay, existing CD variation of mandrel, spacer and block (trim) patterns plays an important role in linewidth and linespace variation. A BEOL line (i.e., metal) can be implemented in four ways using SID-type SADP as shown in Figure 4.7.

Let 3σ CD variation of mandrel, spacer, and block patterns be σ_M, σ_S, and σ_B respectively, and let 3σ overlay between mandrel and block mask be σ_{M-B}. Depending on which pattern defines the edge of a line, linewidth variation changes as:

- Line by only a mandrel: $\sigma_{linewidth} = \sigma_M$
- Line by only spacers: $\sigma_{linewidth} = \sqrt{\sigma_M^2 + (2\sigma_S)^2}$
- Line by a mandrel and a block: $\sigma_{linewidth} = \sqrt{(0.5\sigma_M)^2 + \sigma_{M-B}^2 + (0.5\sigma_B)^2}$

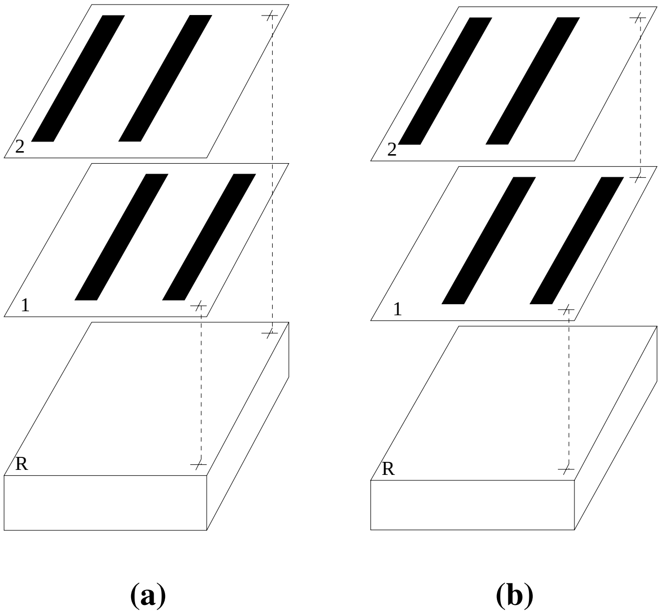


Figure 4.6: Two masks (1 and 2) and a printed reference layer (R). (a) Indirect alignment. (b) Direct alignment.

- Line by a spacer and a block: $\sigma_{linewidth} = \sqrt{(0.5\sigma_M)^2 + \sigma_S^2 + \sigma_{M-B}^2 + (0.5\sigma_B)^2}$

ITRS Lithography Chapter [11] predicts CD variations and overlay requirements of DPL for $32nm$ technology node and below. Table 4.1 shows the requirements of 3σ CD and overlay variations for $32nm$, $28nm$ and $22nm$ technology nodes. Using the variation requirements in Table 4.1, we estimate 3σ linewidth variations in SADP with respect to the four implementation methods. Table 4.2 shows the estimated linewidth variations. From the table, we observe that linewidth variation can significantly increase when a line edge is defined by a block mask pattern. We also observe that a line defined by only mandrel has the smallest linewidth variation. These observations can be used to quantify the quality of SADP layout decomposition solutions from different algorithms. As an example, Figure 4.8 shows two different layout decompositions for the same target layout and resulting final patterns on silicon. With overlay of block mask patterns, solution in (b) suffers from significant linewidth variation, while solution in (c) has no linewidth variation.

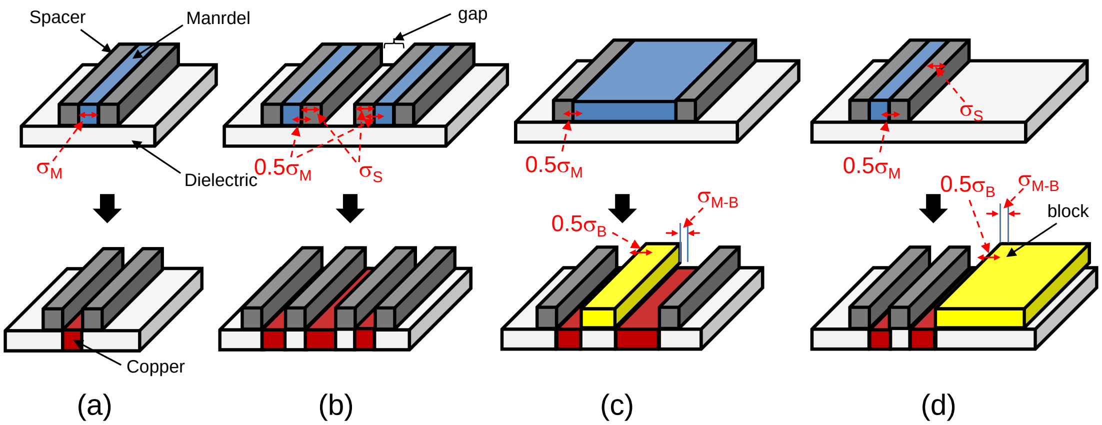


Figure 4.7: A BEOL line in various implementations of SID-type SADP: (a) both line edges are defined by only mandrel edges, (b) both line edges are defined by spacer edges, (c) one line edge is defined by a mandrel edge and the other line edge is defined by a block edge, and (d) one line edge is defined by a spacer edge and the other line edge is defined by a block edge.

4.2 Impact of DPL-Induced BEOL Variation

Interconnect variation due to process variation has been analyzed in a number of references. Mehrotra et al. [131] conduct a simulation-based study of the impact of manufacturing variation on interconnect performance. Lu et al. [127] provide a set of interconnect corner models using Monte Carlo simulations. Stine et al. [171] propose a practical methodology for determining the impact of interconnect pattern-dependent variation without using TCAD tools, and study the impact on simple circuit such as balanced clock networks and an SRAM array. Liu et al. [124] present a model order reduction technique for RLC interconnects including variational analysis based on matrix perturbation expansion theory.

Interconnect variation contributes to circuit delay uncertainty. Lin et al. [123] analyze circuit delay variation due to interconnect parameter variation using efficient experimental designs and sensitivity analysis. Narasimha et al. [135] study the effect of interconnect process variations induced by lithography and etch processes on crosstalk delay and noise. Venkatraman et al. [181] investigate the impact of process-induced parameter variation on global interconnects that require multi-level signaling with variational sensitivity parameters.

Table 4.1: CD and overlay requirements (3σ variation) for DE/DP and SADP processes in ITRS 2010 [11].

DPL option	Parameter	$32nm$	$28nm$	$22nm$
DE/DP	CD (nm)	3.0	2.5	2.1
	Overlay (nm)	5.5	3.8	2.6
	Spacer CD (nm)	1.9	1.4	1.1
SADP	Mandrel CD (nm)	3.0	2.3	1.8
	Overlay (nm)	11.9	8.9	7.1

Table 4.2: 3σ linewidth variations in SID-type SADP due to different implementation methods for $32nm$, $28nm$ and $22nm$ technology nodes.

Method	3σ linewidth variation (nm)		
	$32nm$	$28nm$	$22nm$
Line by only a mandrel	3.0	2.3	1.8
Line by only spacers	4.8	3.6	2.8
Line by a mandrel and a block	12.1	9.0	7.2
Line by a spacer and a block	12.2	9.2	7.3

To relax the design constraints introduced by interconnect variations, Shigyo [166] evaluates a tradeoff between capacitance and RC delay variation caused by fringing capacitances, and suggests a set of design guidelines for the interconnect structures that are insensitive to the process fluctuations. Kahng et al. [105] develop additional matching rules to relax design pessimism via field solver analysis.

Laidler et al. [114] identify the sources of pattern distortions in FinFET technology and investigate overlay sources in [115]. Rigolli et al. [152] present the overlay budget for a double patterning lithography and propose an efficient overlay metrology. Yamamoto et al. [191] propose multi-layer reticle techniques with a single mask to reduce mask-to-mask overlay and mask cost. Sezginer et al. [162] develop a graphi-

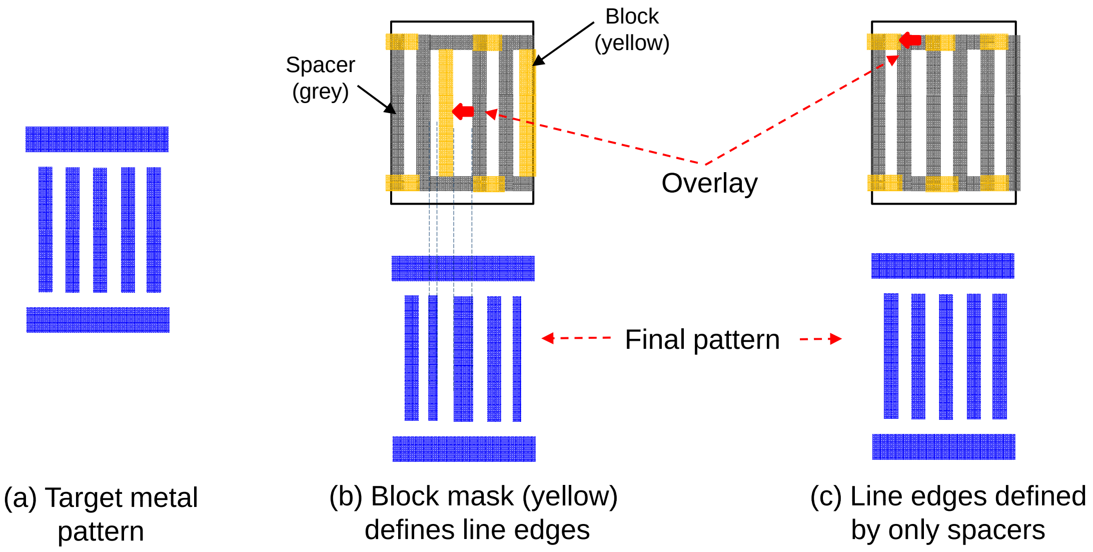

Spacer (grey)
Block (yellow)
Overlay
Final pattern
(a) Target metal pattern
(b) Block mask (yellow) defines line edges
(c) Line edges defined by only spacers

Figure 4.8: Different layout decompositions result in different linewidth variations: (a) a target layout, (b) a layout decomposition solution (top) and resulting final patterns due to overlay (bottom), and (c) another layout decomposition solution (top) and resulting final patterns due to overlay (bottom).

cal method of visualizing the many-dimensional process window for double patterning lithography considering width and space variation from overlay.

A number of works have sought to quantify the impact of overlay in double patterning lithography, either analytically or empirically. Jeong et al. [95] identify the impact of poly linewidth variation from DPL on design timing, and introduce the bimodal linewidth distribution problem in DPL. Ghaida et al. [69] quantify the impact on capacitance and RC delay of individual overlay components, and discuss the relative impact of each component. The impact of overlay in the coupling and total capacitance in BEOL double patterning lithography is addressed with TCAD simulations in [175]. Yang et al. [193] present capacitance and delay variation from overlay in double patterning lithography with analytical modeling of overlay and capacitance variation. However, different impacts from different double patterning lithography options have not been analyzed. In this section, we investigate the detailed mechanisms of linewidth and linespace variation for known DPL techniques with different options, and thoroughly assess the impact on the electrical characteristics of interconnect, using simple structures to chip-level testcases.

4.2.1 TCAD-Based BEOL Analysis

Various interconnect structures in a given interconnect layer interact with each other and form complicated electric fields. To account for these different interconnect structures during circuit design, capacitance tables for each pattern are utilized by designers or RC extractors. The capacitance tables are generated using two- or three-dimensional field solvers for various combinations of width and spacing to neighbors per interconnect layer. Due to the different metal density and patterns, each pattern can have different process variations, so that widths and heights vary based on the context of patterns. Therefore, variational capacitance tables are required. We generate worst-case corners for each capacitance between interconnect pairs using statistical variation information from the semiconductor foundry per each width-height combination. We describe a methodology to generate a variational capacitance table for traditional single-patterning lithography.

There are four major parameters in the traditional interconnect variational analysis, i.e., interconnect width (W), height, space, and dielectric height. An interconnect has intra-layer coupling capacitances C^{intra} with neighbor nets in the same layer, and inter-layer coupling capacitances C^{up} and C^{down} with upper and lower layers.

Figures 4.9 and 4.10 show the variation impact due to overlay or spacer thickness variation for known DPL technologies. In the figures, interconnects are decomposed into *mask1* and *mask2*, and are marked with '1' and '2' correspondingly. S is a parameter for overlay in DE/DP, for which the 3σ value is specified by lithography tool suppliers. In Figure 4.9(a), we shift interconnects printed on *mask2* by a positive value of S to account for overlay in P-DE/DP. A negative value of S implies a shift in opposite direction for the edges of interconnects. Due to the shifting, intra-layer coupling increases on one side of the interconnect but decreases on the other side. We shift a mask by S which varies from -3σ to 3σ with 1σ increments. Figure 4.9(b) shows the impact of overlay on N-DE/DP processes.

In Figures 4.10(a) and (b), we show the printed interconnects in P-SADP and N-SADP processes, respectively. For an SADP process, S is a parameter for the spacer thickness variation determined by manufacturers. In the rest of experiments in this section, we do not consider linewidth variation due to overlay of block mask in SADP.

Figure 4.11(a) shows the impact of rotational overlay on printed features with positive photoresist in DE/DP processes. Negative photoresist would not result in spacing errors due to a rotational overlay component. Figure 4.11(b) shows worst-case rotational impact, where features printed by both masks are inclined towards each other. Below, we conduct 3-dimensional TCAD analysis and compare the impact due to the rotational component of the overlay.[1]

In assessing these double patterning lithography options, we may vary the width and pitch of the interconnects as necessary to simulate impact of overlay. We focus on the translational overlay component which appears to have the largest electrical impact [69].[2]

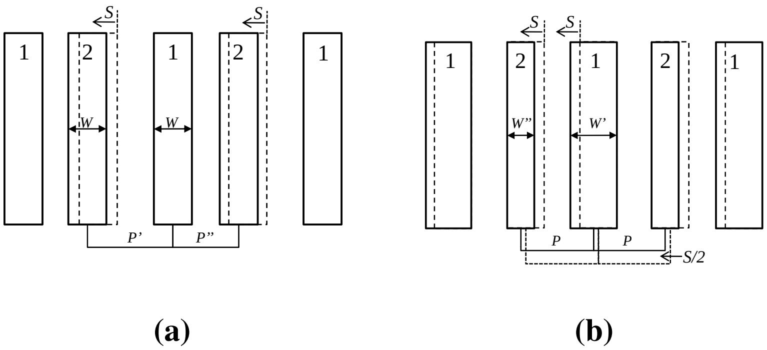


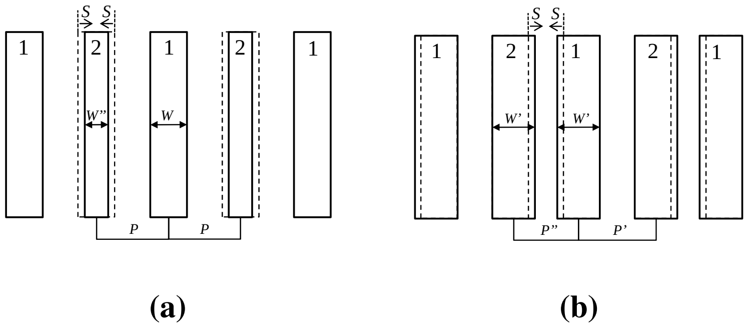
Figure 4.9: (a) P-DE/DP process. Patterns printed using *mask2* are shifted by S due to overlay, which causes pitch and space variation between patterns ($P" \leq P \leq P'$). (b) N-DE/DP process. Overlay varies linewidth ($W" \leq W \leq W'$), but does not affect pitch and space.

4.2.2 Signoff Tool-Based Chip-Level BEOL Analysis

Today, the industry is nearing a critical juncture for choosing among various DPL technology options and process control capabilities. Accordingly, a rigorous, efficient framework is needed for variational performance analyses at chip level, and across many DPL technology options.

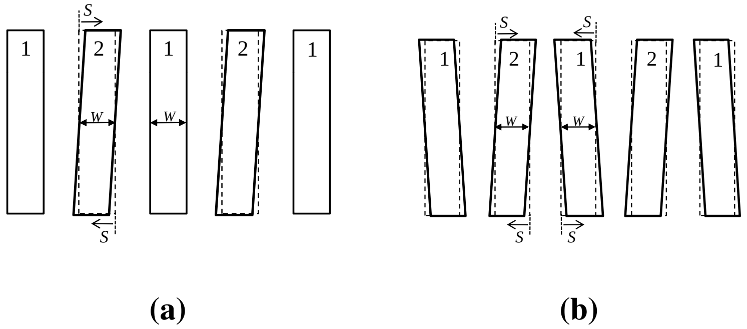
[1] While rotational error is already in the overlay budget, we separately study it in case it requires its own specification in the future.

[2] Although the translational overlay can be reduced by enhanced overlay control, its complete elimination is not possible [11].



(a) **(b)**

Figure 4.10: (a) P-SADP process. Patterns printed after spacer formation can differ in width due to spacer thickness variations S (W" $\leq W$). Spacer thickness variation does not affect pitch but varies space. (b) N-SADP process. Linewidth, space and pitch are varied due to spacer thickness variation. (P" $\leq P \leq P$' and $W \leq W$'.)



(a) **(b)**

Figure 4.11: (a) Rotational overlay due to *mask1* in DE/DP. (b) Worst-case rotational overlay due to *mask1* and *mask2* in different directions.

We describe the BEOL variation analysis flow for different DPL options. We assume direct alignment (DA) for design-level overlay analysis, since IA is expected to have $\sqrt{2}$ times larger pattern shifts within a layer. In DE/DP with DA, we assume that 3σ overlay is S between two DPL masks. Although both masks can be shifted in arbitrary directions and by different amounts, shifts of *mask1* and *mask2* in opposite directions, orthogonal to the preferred routing direction, will induce worst-case space variation between patterns.[3] In SADP, we assume 3σ spacer thickness variation to be $S/2$, so that the maximum CD variation from nominal is set to S as in DE/DP. From the TCAD studies, we have observed that inter-layer coupling and via capacitances are insignificant. Therefore, we decrease the number of combinations by excluding the inter-layer overlay.

[3] The shift of the first mask is with respect to a reference layer similar to a traditional process. We apply necessary computations to ensure that overlay between *mask1* and *mask2* meets ITRS guidelines.

Since the width and space variation differ for each double patterning lithography option, we use different design of experiments (DOE) for each.[4] As inputs of the DOEs, '$layer_{mask1}$' and '$layer_{mask2}$' denote two DPL masks for each DPL-applied BEOL '$layer$' from interconnect layers M2 to M5.

DOE for P-DE/DP. Assuming the width variation from resist or etch variation is sufficiently smaller than the overlay,[5] we perform simulations to analyze impact of space variation due to overlay. From the overlay S, one space increases by S while the other space decreases by S as shown in Figure 4.9(a). The DOE is given as:

```
01: for each $layer \in$ {M2, M3, M4, M5}
02:   for each $S \in$ {-3$\sigma$, -2$\sigma$, -1$\sigma$, 0$\sigma$, 1$\sigma$, 2$\sigma$, 3$\sigma$}
03:     shift $layer_{mask1}$ by +$S/2$
04:     shift $layer_{mask2}$ by -$S/2$
05:     merge $layer_{mask1}$ and $layer_{mask2}$ with remaining layers
06:     RC parasitic extraction and timing analysis
```

DOE for N-DE/DP. Overlay S contributes to width increase for patterns in DPL *mask1* by S and width decrease of patterns in DPL *mask2* by S. However, the space between patterns in different DPL masks remains nominal as shown in Figure 4.9(b). The DOE is given as:

```
01: for each $layer \in$ {M2, M3, M4, M5}
02:   for each $S \in$ {-3$\sigma$, -2$\sigma$, -1$\sigma$, 0$\sigma$, 1$\sigma$, 2$\sigma$, 3$\sigma$}
03:     shift $layer_{mask1}$ by +$S$/2
04:     resize $layer_{mask1}$ by +$S$
05:     shift $layer_{mask2}$ by +$S$/2
06:     resize $layer_{mask2}$ by -$S$
07:     merge $layer_{mask1}$ and $layer_{mask2}$ with remaining layers
08.     RC parasitic extraction and timing analysis
```

DOE for P-SADP. Due to the spacer thickness variation by $S/2$, the width of the even patterns that are generated by the space between spacers in Figure 4.10(a) can

[4] We use the term "DOE" to indicate a set of experiments to evaluate a process variation scenario.

[5] CD control requirement in DRAM at the 32nm half-pitch technology node is 3.3nm, which is around half of the overlay control requirement 6.4nm in ITRS 2008 [11].

be changed by S (two times $S/2$). Since there is no pitch change in P-SADP, when the width of the even patterns increases (decreases) by S, space between adjacent patterns decreases (increases) by $S/2$. The DOE is given as:

```
for each $layer \in \{$M2, M3, M4, M5$\}$
  for each $S \in \{$-3$\sigma$, -2$\sigma$, -1$\sigma$, 0$\sigma$, 1$\sigma$, 2$\sigma$, 3$\sigma\}$
    resize $layer_{mask1}$ by 0
    resize $layer_{mask2}$ by $S$
    merge $layer_{mask1}$ and $layer_{mask2}$ with remaining layers
    RC parasitic extraction and timing analysis
```

DOE for N-SADP. In N-SADP, spacer thickness variation results in width variation of all lines, and in pitch variation as shown in Figure 4.10(b). We resize adjacent lines by $S/2$ to represent the global width variation, and then shift adjacent lines that are facing each other with the varying edges by $S/4$ to represent the pitch variation. The DOE is given as:

```
for each $layer \in \{$M2, M3, M4, M5$\}$
  for each $S \in \{$-3$\sigma$, -2$\sigma$, -1$\sigma$, 0$\sigma$, 1$\sigma$, 2$\sigma$, 3$\sigma\}$
    resize $layer_{mask1}$ by +$S$/2
    shift $layer_{mask1}$ by +$S$/4
    resize $layer_{mask2}$ by +$S$/2
    shift $layer_{mask2}$ by -$S$/4
    merge $layer_{mask1}$ and $layer_{mask2}$ with remaining layers
    RC parasitic extraction and timing analysis
```

With the output RC parasitic files, we analyze timing and capacitance variations for individual nets, and the timing of the whole design.

4.2.3 Experiments

Interconnect analysis setup. We use the interconnect lateral and vertical dimensions in Table 4.3 for interconnect variation analysis. The values are obtained from the ITRS [11].

Table 4.3: Nominal dimensions of an intermediate layer interconnect for 45nm, 32nm, 28nm and 22nm technology nodes.

Technology node	45nm	32nm	28nm	22nm
Interconnect width (nm)	68	61	43	31
Interconnect height (nm)	122	110	77	55
Dielectric thickness (nm)	122	110	77	55
Dielectric constant	3.3	3.3	2.9	2.8

Interconnect capacitance tables. We conduct experiments across various DPL options to be able to compare these options in terms of capacitance values. The first experiment compares impact of overlay in various DPL options, and also compares the impact of overlay with that of width, height, or all width, height and dielectric thickness variations. For overlay, we do not consider CD variations from a single lithography step in DPL, i.e., first and second patternings in DE/DP, and mandrel patterning in SADP. In this experiment, we use 45nm node interconnect parameters in Table 4.3.

Tables 4.4 and 4.5 compare capacitance values extracted using *Synopsys Raphael* [26]. C^{down}, C^{top}, and C^{intra} are the capacitances to the lower, upper, and intra-layer interconnects, respectively. C^{total} is the total capacitance. Subscripts min, nom, and max indicate minimum, nominal and maximum cases, respectively.[6] From the results, we observe that positive photoresist process with direct alignment (DA) results in similar capacitance impact due to overlay or width variations. We also observe that IA results in larger capacitance changes compared to DA cases of the positive photoresist case.

The second experiment compares overall capacitance variations, including CD variations of all lithography steps, i.e., first and second patternings in DE/DP, and mandrel and block patternings in SADP. In this experiment, we use interconnect nominal parameters of 32nm, 28nm and 22nm technology nodes in Table 4.3, and 3σ CD variation and overlay of those technology nodes in Table 4.1.

Tables 4.6, 4.7 and 4.8 show minimum, nominal and maximum values of intra-

[6] Since each value is found as a maximum value among all the DOE results, the summation of partial coupling capacitance may not match with the total capacitance.

Table 4.4: Inter-layer capacitance ($aF/\mu m$) comparison for a $45nm$ technology.

Configuration	C^{down}_{min}	C^{down}_{nom}	C^{down}_{max}	C^{top}_{min}	C^{top}_{nom}	C^{top}_{max}
P-DE/DP, DA, Overlay Only	31	31	31	31	31	31
P-DE/DP, DA, Width Only	30	31	31	31	31	32
P-DE/DP, DA, All Variations	25	31	39	26	31	40
P-DE/DP, IA, Overlay Only	31	31	31	31	31	31
P-DE/DP, IA, Width Only	30	31	31	31	31	32
P-SADP, Spacer Thickness Only	29	31	31	30	31	31
P-SADP, All Variations	25	31	40	25	31	41
N-DE/DP, DA, Overlay Only	27	31	33	28	31	34
N-DE/DP, DA, Height Only	31	31	31	31	31	32
N-DE/DP, DA, All Variations	22	31	43	23	31	44
N-DE/DP, IA, Overlay Only	26	31	34	27	31	35
N-DE/DP, IA, Height Only	31	31	31	31	31	32
N-SADP, Spacer Thickness Only	30	31	31	31	31	31
N-SADP, All Variations	26	31	38	26	31	39

and inter-layer capacitances from various DPL options for $32nm$, $28nm$ and $22nm$, respectively. Intra-layer (coupling) capacitance correlates to noise or coupling-induced delay at the design level. Total capacitance, on the other hand, correlates with circuit delay. For SADP, we also compare the capacitances from different line implementation methods discussed in Section 4.1.2. In the table, SADP1, SADP2, SADP3 and SADP4 denote the different line generation methods: “line by only mandrel”, “line by only spacers”, “line by a mandrel and a block”, and “line by a spacer and a block”, respectively.

From the tables, we can observe that P-/N-SADP1 and P-/N-SADP2 show smaller capacitance variations (i.e., difference between minimum and maximum values), compared to P-/N-DE/DP. This is because the linewidth or space of P-/N-SADP1 and P-/N-SADP2 are defined by only spacer thickness or mandrel CD, but those of DE/DP are involved with overlay. We also observe that P-/N-SADP3 and P-/N-SADP4 show

Table 4.5: Intra-layer capacitance ($aF/\mu m$) comparison and total capacitance ($aF/\mu m$) comparison for a $45nm$ technology.

Configuration	C_{min}^{intra}	C_{nom}^{intra}	C_{max}^{intra}	C_{min}^{total}	C_{nom}^{total}	C_{max}^{total}
P-DE/DP, DA, Overlay Only	56	68	85	200	200	205
P-DE/DP, DA, Width Only	56	68	86	174	200	222
P-DE/DP, DA, All Variations	36	68	135	152	200	283
P-DE/DP, IA, Overlay Only	52	68	95	200	200	210
P-DE/DP, IA, Width Only	56	68	86	174	200	222
P-SADP, Spacer Thickness Only	61	68	76	189	200	200
P-SADP, All Variations	49	68	90	165	200	236
N-DE/DP, DA, Overlay Only	68	68	68	192	200	204
N-DE/DP, DA, Height Only	57	68	79	178	200	214
N-DE/DP, DA, All Variations	44	68	102	145	200	271
N-DE/DP, IA, Overlay Only	68	68	68	189	200	206
N-DE/DP, IA, Height Only	57	68	79	178	200	214
N-SADP, Spacer Thickness Only	56	68	68	187	200	200
N-SADP, All Variations	44	68	99	163	200	242

even larger variations than DE/DP, although linewidth or space of all these options are affected by overlay. This is due to larger overlay spec for SADP, as shown in Table 4.1. Among all SADP cases, N-SADP3 and N-SADP4 show significantly larger variation than P-SADP3 and P-SADP4. This is because overlay causes space variation in N-SADP3 and N-SADP4, while overlay causes linewidth variation in P-SADP3 and P-SADP4. Due to the high aspect ratio of interconnect, space variation appears more significant than linewidth variation.

We analyze the trend of capacitance variation with respect to technology nodes. Table 4.9 shows the ratio of the total capacitance (C^{intra} + C^{inter}) variation to its nominal value for each technology node. We can observe that the ratio of capacitance variation slowly increases as technology advances.

We also conduct 3-dimensional TCAD field solver analysis for rotational over-

Table 4.6: Intra- and inter-layer capacitance ($aF/\mu m$) comparison for a $32nm$ technology.

Configuration	C_{min}^{intra}	C_{nom}^{intra}	C_{max}^{intra}	C_{min}^{inter}	C_{nom}^{inter}	C_{max}^{inter}
P-DE/DP	132	139	147	62	64	65
N-DE/DP	131	139	146	59	64	69
P-SADP1	134	138	143	61	64	66
P-SADP2	134	138	143	61	64	67
P-SADP3	132	139	145	56	64	72
P-SADP4	132	139	145	56	64	72
N-SADP1	134	139	144	64	64	64
N-SADP2	134	139	144	64	64	64
N-SADP3	119	140	164	60	64	67
N-SADP4	126	141	157	60	63	66

lay, using the setup in Figure 4.11. Upper and lower layers contain orthogonal interconnects with same width and spacing and no overlay. We use 500,000 grid points, and use the same technology and dimensions with $1\mu m$ lines. In Figure 4.11(a), coupling between the middle interconnect and one of its immediate neighbors increases by only 0.3% due to rotational overlay. Figure 4.11(b) provides the worst case, where the impact rises to 2.82%. We conclude that rotational overlay is not as significant, as one section of a line would get closer to, while the remaining section would move away from, an intra-layer neighbor. The magnification component is similar to width variations; hence, we focus on the translational component of overlay.

Chip-level analysis setup. Traditional parasitic extraction tools directly read a design database (e.g., design exchange format (DEF), GDS, etc.), and use capacitance tables that contain width or height variations of metal or dielectric layers. To account for overlay in extraction, we present a new RC extraction flow for double patterning lithography as shown in Figure 4.12. Details of the flow are as follows.[7]

[7] Although we perform exhaustive analysis for the purpose of technology selection, use of the proposed framework for DPL variability analysis targets worst-case corners only, thereby reducing the number of simulations.

Table 4.7: Intra- and inter-layer capacitance ($aF/\mu m$) comparison for a $28nm$ technology.

Configuration	C^{intra}_{min}	C^{intra}_{nom}	C^{intra}_{max}	C^{inter}_{min}	C^{inter}_{nom}	C^{inter}_{max}
P-DE/DP	115	123	131	54	56	58
N-DE/DP	115	122	130	52	56	61
P-SADP1	118	122	126	54	56	59
P-SADP2	118	122	126	53	56	59
P-SADP3	116	122	128	49	56	64
P-SADP4	116	122	128	49	56	64
N-SADP1	118	122	127	56	56	56
N-SADP2	118	122	127	56	56	56
N-SADP3	104	124	146	53	56	59
N-SADP4	111	125	141	53	56	58

- *Step 1. Initial GDS.* We stream out GDS from a routed design.

- *Step 2. Split GDS.* We generate a base GDS that only has all front-end-of-line (FEOL) layers, i.e., n-well, active, p-implant, along with larger-dimension interconnect layers that do not use double patterning lithography, and sub-GDS files for double patterning-applied layers. We assume that double patterning lithography is applied to local interconnect layers which use stricter design rules.

- *Step 3. Pattern decomposition for double patterning lithography.* For local interconnect layers, we generate two sub-GDS files from a decomposition of the original layout using integer linear programming-based min-cost coloring [103]. Finally, patterns in each local layer are split into two masks, $layer_{mask1}$ and $layer_{mask2}$.

- *Step 4. Shift and merge.* To model interconnect parameter variations due to overlay, each sub-GDS in each layer is overlaid with a different origin point on top of the base GDS. For instance, to model a -10nm translational overlay for M2 layer's first double patterning mask ($M2_{mask1}$), we locate the sub-GDS containing

Table 4.8: Intra- and inter-layer capacitance ($aF/\mu m$) comparison for a $22nm$ technology.

Configuration	C^{intra}_{min}	C^{intra}_{nom}	C^{intra}_{max}	C^{inter}_{min}	C^{inter}_{nom}	C^{inter}_{max}
P-DE/DP	109	118	127	52	54	56
N-DE/DP	109	118	127	49	54	59
P-SADP1	113	117	122	51	54	57
P-SADP2	113	117	122	51	54	57
P-SADP3	111	117	124	46	54	62
P-SADP4	111	117	124	46	54	62
N-SADP1	113	118	123	54	54	54
N-SADP2	113	118	123	54	54	54
N-SADP3	99	120	144	51	54	57
N-SADP4	106	121	140	51	54	56

the $M2_{mask1}$ at (-10nm, 0nm) in the coordinate system of the base GDS. To shift and merge GDS files, we use SKILL scripts with the *Cadence Virtuoso Layout Design Environment IC6.1.0.243* [8].

- *Step 5. Resize and extraction.* We use the $SIZE$ command in *Synopsys Hercules v2006.12-8* [22] to expand or shrink original patterns to account for width variation from overlay. After width change, the $BOOLEAN\ OR$ command is used to merge two double patterning mask layers. Finally, we use *Synopsys Star-RCXT v2007.06* [28] for RC extraction.

We implement from RTL the open-source core AES, obtained from $opencores.org$ [15]. With 4ns clock cycle time, we synthesize, place and route the testcase with *Nangate* 45nm open cell library [14] using *Cadence RTL Compiler v5.2* [6] and *Cadence SOC Encounter v7.2* [7]. The final implemented AES has 86% placement utilization with 26,069 standard cell instances, and average 10% (maximum 14%) metal density with no metal fill insertion.[8] We also implement another testcase with floating

[8] Metal density is calculated for only signal routing layers. The maximum metal density value for signal nets is 50%, when all routing tracks are occupied.

Table 4.9: Ratio of variation to nominal value ($\%$) for total capacitance (i.e., $C^{intra} + C^{inter}$) with respect to technology nodes.

Configuration	$32nm$	$28nm$	$22nm$
P-DE/DP	4.5	5.5	6.6
N-DE/DP	6.1	6.8	7.8
P-SADP1	3.4	3.8	4.3
P-SADP2	3.8	3.9	4.4
P-SADP3	6.8	7.2	8.5
P-SADP4	6.9	7.3	8.6
N-SADP1	2.7	2.8	3.3
N-SADP2	2.7	2.8	3.3
N-SADP3	13.1	14.2	15.9
N-SADP4	9.3	9.9	12.0

track-type dummy fill to observe the impact of overlay in a manufacturing-ready design. Average and maximum metal density with metal fill are 37$\%$ and 46$\%$ for all routing layers, respectively.

For the BEOL stack of the chip-level design, we use five small-dimension metal layers (M1-M5), and two large-dimension metal layers (M6 and M7) as shown in Table 4.10. The values reflect a representative $45nm$ technology, from the ITRS [11]. We use 20$\%$ of nominal interconnect width as 3σ variation of overlay, and use 3.3 as the effective dielectric constant for all dielectric materials. We do not include the variation in FEOL, since the impact of overlay in FEOL needs to be included in cell characterization and library generation. Due to the minimum precision of $1nm$ for the layout editor, we use $12nm$ for 3σ of overlay or spacer thickness variability S for design-level analysis, instead of the $10.4nm$ that is 20$\%$ of M1 width.

Chip-level analysis results. The first analysis compares the coupling-induced delay variation due to overlay. We use *Synopsys PrimeTime-SI vB-2008.12-SP2* [25] as a standard coupling-aware delay calculator which takes into account the amount of cross-coupled capacitance and relative arrival times. This tool also considers slew rates of

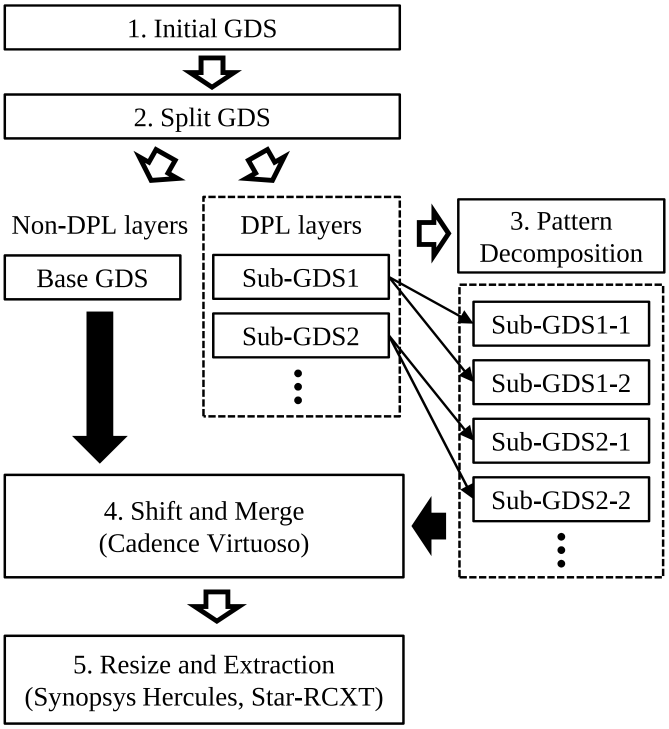


Figure 4.12: Extraction flow for double patterning lithography considering overlay.

all signal transitions, switching directions, and combined effects of all aggressors on a victim net. After logical and electrical filtering using functional checking and timing window comparisons, coupling capacitances greater than a specific threshold value are considered during the coupling noise analysis.

We identify the net with largest coupling-induced delay in the nominal design. This net consists of three interconnect segments: 1.604μm of M2, 0.78μm of M3 and 14.788μm of M4 segments. The M2 segment has a same-layer neighbor with minimum spacing on the right-hand side. Two M1 nets and three M3 nets cross the M2 segments. The M3 segment does not have any neighbor with minimum spacing. The M4 segment has neighbors at minimum spacing on both sides. 26 M5 nets and 31 M3 nets cross the M4 segment. Figure 4.13 illustrates simplified configurations of the selected nets, with negative variation of S, for each DPL technique. Black boxes and white boxes with solid boundaries denote the selected net (victim) and neighbors (aggressors) in the same layer, respectively. The boxes with dotted boundaries represent the original patterns

Table 4.10: Technology stack parameters for a 45nm technology.

Layer	W_{nom}	H_{nom}	D_{nom}
poly	45nm	80nm	160nm
M1	52nm	94nm	94nm
M2	52nm	94nm	94nm
M3∼M5	68nm	122nm	122nm
M6, M7	104nm	188nm	188nm
3σ variation	$W_{3\sigma}$	$H_{3\sigma}$	$D_{3\sigma}$
M2∼M5	13.6nm	24.4nm	24.4nm

without variations, and the orthogonal gray boxes represent the aggressors in upper or lower layers. From the layout configurations, we expect that intra-layer coupling will dominate for the M2 segment, and that both inter-layer and intra-layer coupling will affect coupling-induced delay variation.

The number of aggressors after filtering is five, and the aggressors are connected to the victim net via 27 coupling capacitances. Coupling-induced delay change without metal fill ('w/o metal fill') and with metal fill ('w/ metal fill') at the nominal corner (S = 0) is 364ps and 292ps, respectively. After metal fill insertion, total capacitance of the net increases from 2.946fF to 3.023fF. However, in 'w/ metal fill', ground capacitance increases from 1.087fF to 1.385fF, but coupling capacitance decreases.

We now discuss in detail, by way of example, the coupling-induced delay variation with M4 overlay error.[9] Figure 4.14 shows the coupling-induced delay variation with different M4 overlay bounds.

- **P-DE/DP.** For the selected victim net, linewidth does not change but the space between aggressors and the victim changes. For M4 overlay, with both negative and positive S, the coupling-induced delay increases. This is because the capacitance increase due to neighbor nets on one side is larger than the capacitance decrease with neighbors on both sides.

[9] Coupling-induced delay variation due to M2 overlay is smaller than that due to M4 overlay, and the impact of overlay in M3 and M5 layers contributed by inter-layer coupling variation is around ±1%.

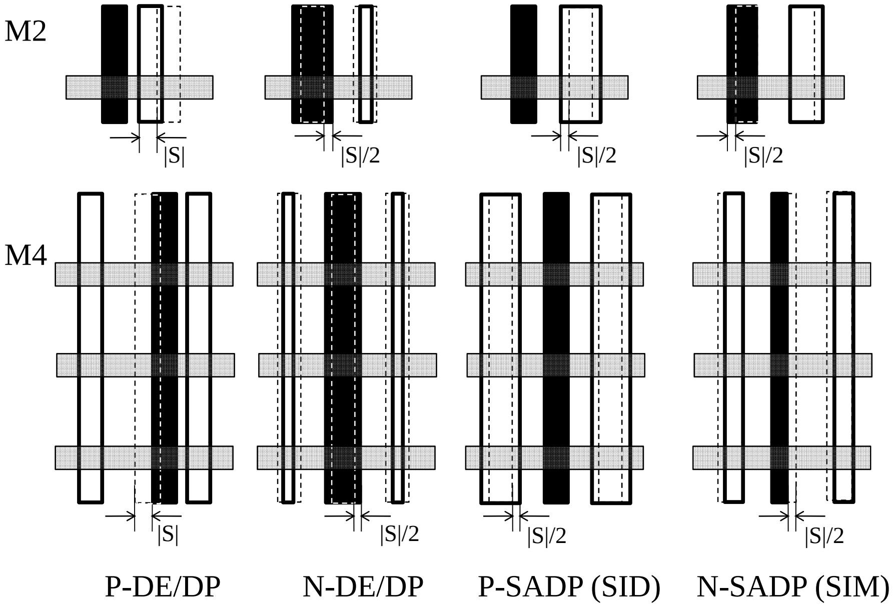


Figure 4.13: Simplified configurations of a net having the largest delay variation due to coupling in the testcase before metal fill.

- **N-DE/DP.** Since the space between intra-layer aggressors and the victim does not change, coupling capacitance variation with respect to intra-layer aggressors is small. However, linewidth increase (decrease) of the victim net amplifies (decreases) ground capacitance and inter-layer coupling. For the M4 segment, negative (positive) S results in linewidth increase (decrease) of the victim net as shown in Figure 4.13. Coupling-induced delay due to the large number of neighbors on upper and lower layers, increases (decreases) with linewidth increase (decrease).

- **P-SADP (SID-type).** For this specific victim net, the M4 segment consists of the patterns underlying the primary patterns in the first litho-etch step of the SADP process. Negative (positive) S leads to smaller (larger) spacer thickness which results in smaller (larger) space between the victim and aggressors. Therefore, coupling-induced delay increases with negative S and decreases with positive S. Since the spaces on both sides of the M4 segment are increased or decreased at the same time, the impact of overlay in P-SADP is larger than that of DE or DP.

- **N-SADP (SIM-type).** The space on one side of the M4 segment changes, causing intra-layer coupling variation. The width of the victim also changes, causing

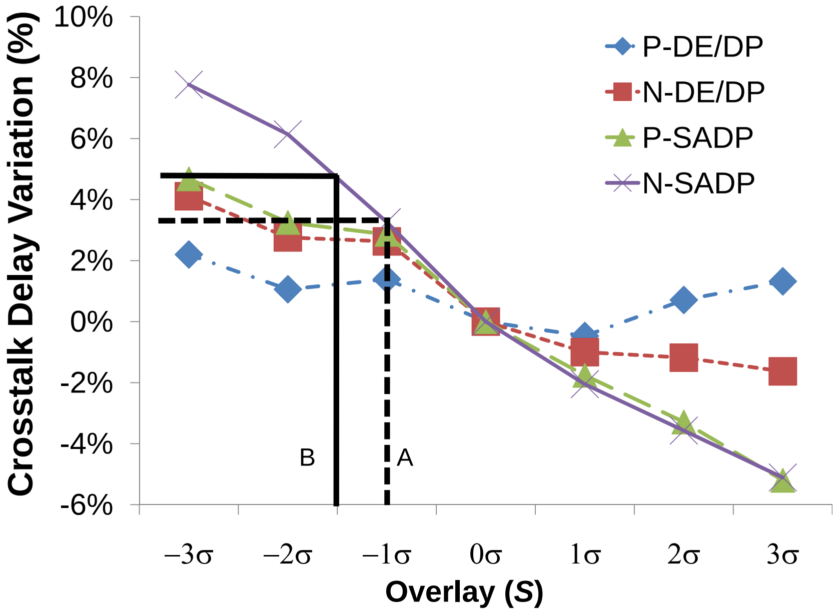


Figure 4.14: Coupling-induced delay variation (%) (y-axis) due to M4 overlay (x-axis).

inter-layer coupling variation. Positive S leads to linewidth increase and spacing decrease, so that the coupling-induced delay increases. Negative S leads to linewidth decrease as well as spacing increase as shown in Figure 4.13, so that the coupling-induced delay variation decreases. Note that in Figure 4.14, we inversely plot N-SADP results against S, to juxtapose delay variations against those of other options. Since intra-layer and inter-layer couplings vary together in the same direction, overlay impact in N-SADP can be larger than in P-SADP.

From Figure 4.14, we can observe the relative significance of the overlay control requirement for each option.

1. P-DE/DP has the smallest variation from overlay. With the same 3σ overlay control, the variations in P-DE/DP, N-DE/DP, P-SADP and N-SADP are 2.20%, 4.11%, 4.68% and 7.77%, respectively. This implies that the overlay control requirement for P-DE/DP can be relaxed compared to the other technology options.

2. If overlay control in N-SADP is relatively easy, such that it can be controlled

within 1σ,[10] then the overlay control requirement for P-SADP and N-DE/DP must be within 2σ to have a similar level of variation in N-SADP, e.g., 3.26% variation, as indicated by the dotted line 'A'.

3. The overlay control requirement for N-SADP should be twice as tight as for the others. For instance, if we target 3σ overlay for N-DE/DP or P-SADP, the overlay control for N-SADP must be within 1.5σ to have variation similar to N-DE/DP or P-SADP, as indicated by the solid line 'B'.

The second analysis compares capacitance variation due to overlay. Figure 4.15 shows interconnect capacitance changes of the top 5,307 high-capacitance nets ($\geq 2fF$) of the 'w/o metal fill' testcase. We measure maximum increase, maximum decrease and mean variation, by comparing extracted parasitic files. In most cases, we observe more than 10% increase or decrease of capacitances from the nominal capacitance values. Such increases and decreases of capacitances will contribute to larger on-chip variations in timing analysis.

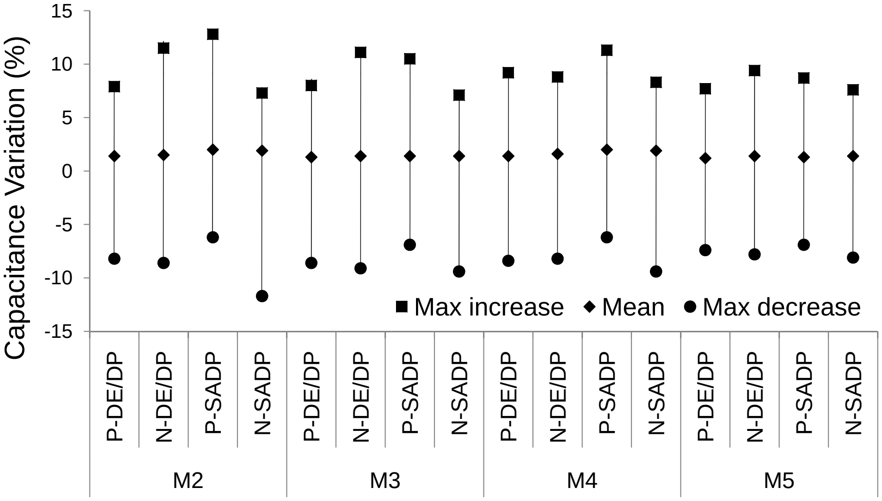


Figure 4.15: Capacitance changes (%) of high-capacitance nets ($\geq 2fF$) from 3σ overlay.

[10] Spacer thickness variation in SADP can be much less than overlay in DE/DP, e.g., 1/3 of the overlay control spec in DE/DP, since spacer thickness is controlled by well-controlled oxide growth, deposition and thinning rates.

The third analysis compares the impact of overlay on design timing. We use total negative slack (TNS), which is the sum of timing slack values at all endpoints in static timing analysis, as a metric to quantify design timing. Figure 4.16 shows the normalized TNS of the worst-case corner ($S = \pm 3\sigma$) with respect to the TNS of the nominal corner (S = 0). Values on the y-axis give the relative variation from the TNS value of the nominal corner (1.0). The total negative slack in the nominal corner is -63ns without metal fill and -83ns with metal fill.

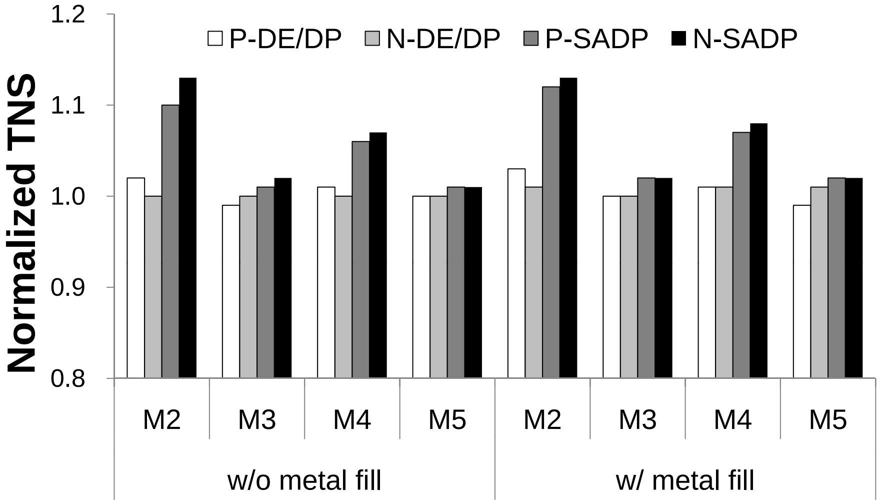


Figure 4.16: Normalized total negative slack (y-axis) due to 3σ overlay in each layer (x-axis) for each double patterning lithography option.

From the TNS variation analysis, we observe that P-SADP and N-SADP options have greater sensitivity to the overlay than DE/DP, since both linewidths and spaces are varied in P-SADP and N-SADP. Figure 4.17 shows the relative sensitivity of double patterning lithography options with respect to overlay. We observe that both P-SADP and N-SADP are more sensitive than DE/DP with the same overlay, and that the lower layer (M2), which uses smaller-dimension design rules, is more sensitive than higher layers with larger-dimension design rules.

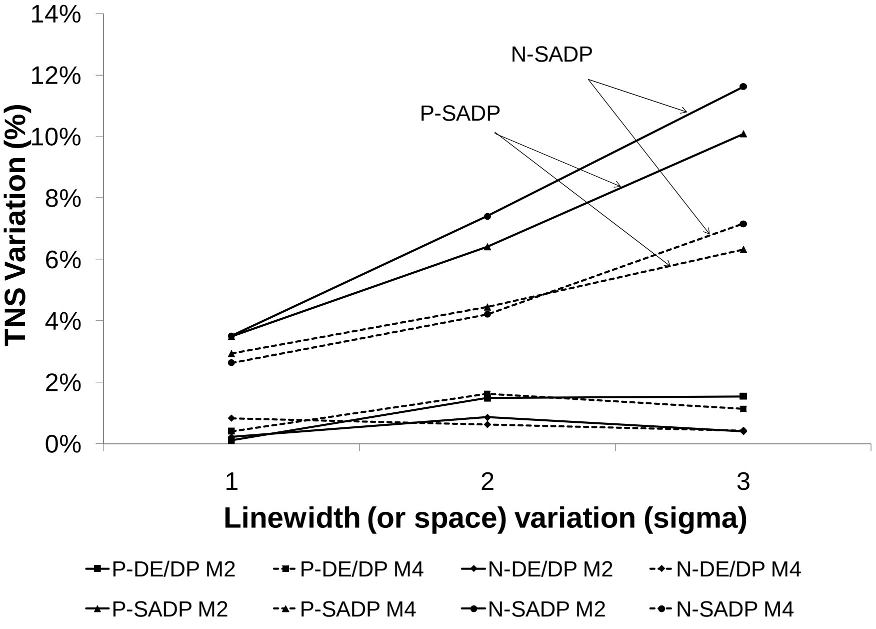


Figure 4.17: Total negative slack variation (%) from the nominal value in each double patterning lithography option with respect to overlay S variation.

4.3 Impact of DPL-Induced FEOL Variation

4.3.1 Impact on Coupling Capacitance in FEOL Layers

To evaluate the impact of overlay in front-end-of-line (FEOL) double patterning, we introduce overlay to poly and contact layers with respect to M1 layer, and measure delay variation.

We select the five most commonly instantiated standard cells in the used testcase as implemented using the *Nangate 45nm Open Cell Library* [14], i.e., INV_X2, INV_X4, NOR2_X2, NOR2_X4 and NAND2_X2. These account for more than 60% of the instances in the used testcase. We decompose each cell layout into five sub-layouts; '$BASE$' including diffusion, P+/N+ masks and n-well; '$P1$' and '$P2$' for odd and even poly lines counting from the leftmost cell boundary; 'C' for contact layer; and 'M' for M1 metal layer as shown in Figure 4.18. We then merge sub-layouts shifting each sub-layout by the amount $\{$-3,-2,-1,0,1,2,3$\}\sigma$ of overlay of each layer. We shift patterns in the horizontal direction only, since the most important patterns for delay are the poly gates, which are drawn in the vertical direction and that have coupling to contacts and neighboring poly gates.[11]

We measure both rise and fall delay of the cells for all possible combinations of overlay between layers. The maximum impact of overlay on cell delay is as small as 2%, since coupling capacitance between poly gate and contact or metal is small compared with gate capacitance; this is a consequence of gate oxide thickness (less than $2nm$ in $45nm$ bulk technology). Note that this delay variation is only from the parasitic capacitance variation due to the overlay. Overlay can amplify the poly and diffusion shape rounding after lithography, and channel-stress variation from contact overlay can have larger impact on delay than coupling capacitance variation. Yet, such effects are supposed to be embedded in silicon measurement data and the characterized library.

Let the x-coordinate of $P1$, $P2$, C and M after overlay be x_{P1}, x_{P2}, x_C, and x_M, respectively, which are the distance measured from the original x-coordinate, and let 3σ overlay error for each layer be d_{P1}, d_{P2}, d_C and d_M. We use $10nm$ for d_{P1} and d_{P2}, and $13nm$ for d_C and d_M. We measure both rise and fall delay of the cells for all 81 combinations of overlay between layers. However, the impact of overlay error on cell delay is small. We use a design of experiment (DOE) for FEOL analysis as follows.

```
for each $x_{P1} \in \{-d_{P1}, 0, d_{P1}\}$
  for each $x_{P2} \in \{-d_{P2}, 0, d_{P2}\}$
    for each $x_C \in \{-d_C, 0, d_C\}$
      for each $x_M \in \{-d_M, 0, d_M\}$
        shift $P1$ by $x_{P1}$
        shift $P2$ by $x_{P2}$
        shift $C$ by $x_C$
        shift $M$ by $x_M$
        merge $P1$, $P2$, $C$, and $M$
        Netlist and parasitic extraction
        Run SPICE and measure delay
```

Table 4.11 shows the fall delay variation from the nominal case (x_{P1} = x_{P2} =

[11] Shifting patterns in the y-direction does not change space between the poly gates and neighboring contacts; thus, gate capacitance variation from y-direction overlay is expected to be smaller than that from x-direction overlay.

$x_C = x_M = 0$) of INV_X4 for each corner of the DOE that shows the largest variation among the five selected cells. The delay values of other cells are smaller than INV_X4. However, the impact of overlay error on cell delay is small. The delay variation can be significant with overlay error when the actual litho images of poly and diffusion are used in the analysis.[12]

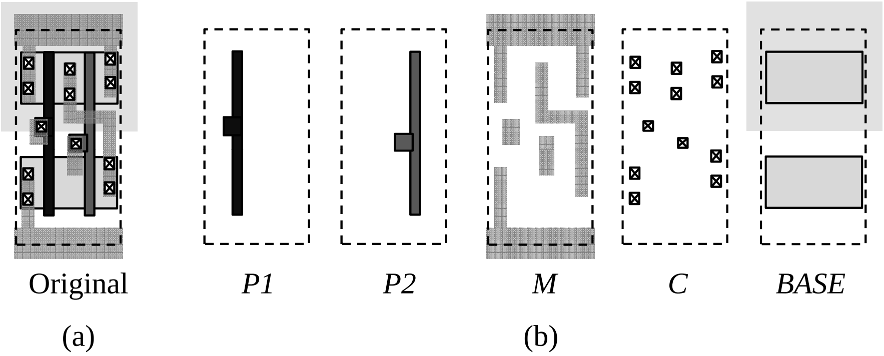


Figure 4.18: Mask decomposition for FEOL overlay simulation. (a) Original NAND2_X2 layout. (b) Five decomposed sub-layouts.

4.3.2 Impact on Via Resistance Variation

Due to the overlay between metal layers and via layers, via area enclosed by metal layer can be reduced, causing resistance increase as well as reliability problems such as electromigration. Thus, we also evaluate the impact on design timing of via resistance variation due to overlay.

In design-level analysis, traditional RC extractors use a resistance table defined in a technology file, e.g., Interconnect Technology File (ITF). We increase via resistance values, extract RC values from the used testcase, and analyze timing. We make the pessimistic assumption that via resistance can vary up to 2× from its original specification solely due to overlay. Table 4.12 summarizes timing variation due to this pessimistic, overlay-specific via resistance variation. We measure the critical-path delay as well as total negative slack (TNS) of the testcases. From the table, we can observe that the im-

[12] Overlay error can amplify the poly and diffusion shape rounding after lithography, and channel-stress variation from contact overlay can have larger impact on delay than coupling capacitance variation. Yet, such affects are supposed to be embedded in silicon measurement data and the characterized library.

Table 4.11: Fall delay variation (%) of INV_X4 for each overlay scenarios.

			x_C = -d_C	x_C = 0	x_C = d_C
x_{P1} = -d_{P1}	x_{P2} = -d_{P2}	x_M = -d_M	0.91	1.19	0.89
		x_M = 0	0.89	1.15	0.87
		x_M = d_M	0.87	1.13	0.83
	x_{P2} = 0	x_M = -d_M	0.06	0.23	0.45
		x_M = 0	0.04	0.19	0.40
		x_M = d_M	0.04	0.17	0.38
	x_{P2} = d_{P2}	x_M = -d_M	-1.45	-0.68	-0.64
		x_M = 0	-1.47	-0.72	-0.66
		x_M = d_M	-1.47	-0.74	-0.70
x_{P1} = 0	x_{P2} = -d_{P2}	x_M = -d_M	0.72	1.02	0.70
		x_M = 0	0.70	0.98	0.68
		x_M = d_M	0.68	0.96	0.64
	x_{P2} = 0	x_M = -d_M	-0.15	0.04	0.23
		x_M = 0	-0.15	0.00	0.21
		x_M = d_M	-0.17	-0.02	0.17
	x_{P2} = d_{P2}	x_M = -d_M	-1.68	-0.89	-0.87
		x_M = 0	-1.68	-0.94	-0.89
		x_M = d_M	-1.70	-0.96	-0.91
x_{P1} = d_{P1}	x_{P2} = -d_{P2}	x_M = -d_M	0.51	0.85	0.51
		x_M = 0	0.49	0.81	0.49
		x_M = d_M	0.47	0.79	0.47
	x_{P2} = 0	x_M = -d_M	-0.38	-0.15	0.02
		x_M = 0	-0.38	-0.19	0.00
		x_M = d_M	-0.40	-0.21	-0.02
	x_{P2} = d_{P2}	x_M = -d_M	-1.94	-1.11	-1.09
		x_M = 0	-1.94	-1.15	-1.11
		x_M = d_M	-1.96	-1.15	-1.13

pact of via resistance variation on timing is as small as 0.1%. Of course, these results are highly technology- and circuit-specific. If via resistance variation substantially impacts performance in a given technology and circuit, via resistances may be updated using enclosed areas as input in via resistance formulas.

Table 4.12: Critical-path delay (ns) and total negative slack (ns) with original via resistance and with $2\times$ larger via resistance.

		Original	$2\times$
Via resistance (Ω)	V1, V2	3.3	6.6
	V3, V4, V5	2.5	5.0
	V6	1.7	3.4
Critical Path Delay (ns)		4.994	4.999
Total Negative Slack (ns)		-140.995	-141.185

4.3.3 Impact of Bimodal CD Distribution

In traditional single-exposure lithography, adjacent identical layout features will have identical mean critical dimension (CD), and spatially correlated CD variations. However, with DPL, adjacent features can have distinct mean CDs, and uncorrelated CD variations. This introduces a new set of ‘bimodal’ challenges for timing analysis and optimization.

In this subsection, we assess the potential impact of DPL on timing analysis error and guardbanding, and find that the traditional ‘unimodal’ characterization and analysis framework may not be viable for DPL. For example, using $45nm$ models, we find that different DPL mask layout solutions can cause $50ps$ skew in clock distribution that is unseen by traditional analyses. Different mask layouts can also result in 20% or more change in timing path delays. Such results lead to insights into physical design optimizations for clock and data path placement and mask coloring that can help mitigate the error and guardband costs of DPL. Figure 4.19(a) shows a bimodal CD distribution for $32nm$ technology measured from 24 wafers processed by DPL, as reported in [64].

Figure 4.19(b) shows a simplified illustration of the bimodal CD distribution, in which two CD groups have independent mean and sigma values. The bimodal CD distribution affects design timing as follows.

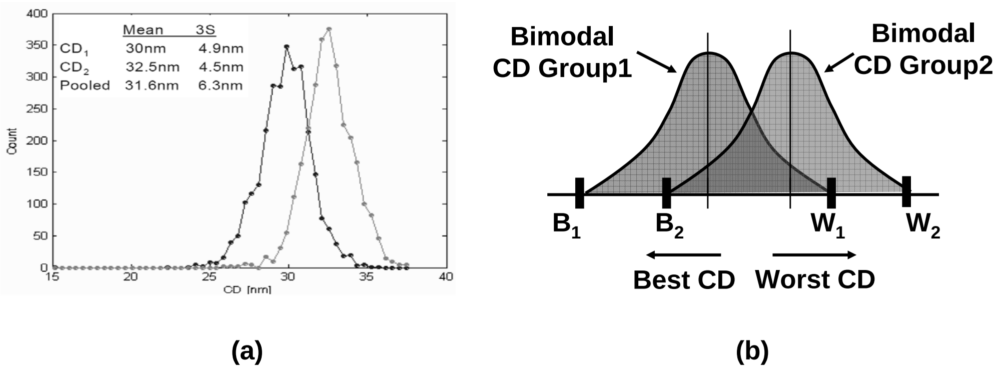


Figure 4.19: Bimodal CD distribution. Left figure reproduced from [64].

Loss of spatial correlation. The existence of two independent CD populations in a design takes away the presumptions of spatial correlation that has always been used to reduce pessimism in corner-based timing analysis. For example, consider two closely placed, identical inverters made with different steps of double patterning lithography – i.e., one inverter is made by the first litho-etch step and the other is made by the second litho-etch step. These two inverters can have different gate CDs, so that their electrical characteristics, such as delay and power, can also be extremely different from each other despite being adjacent in the same die.

In general, within-die variations are taken into account by on-chip variation (OCV) models or by statistical timing analysis flows. Bimodal CD distribution can also be treated as an additional variation source. However, the important problem that we address in this work is that the size of the variation from the bimodal CD distribution can be very large, e.g., over 8% of mean CD difference between the groups, as shown in Figure 4.19(a); therefore, designers must consider more extreme within-die variations during timing optimization as a direct consequence of DPL.

Increase of overall CD variation. Unless the two CD populations have the same mean values, overall CD variation must be increased with DPL. Dusa et al. propose the use of a unimodal representation pooled from the bimodal CD distribution [64], specifically,

$$\begin{aligned}(3\sigma_{CD,pooled})^2 &= \frac{(3\sigma_{CD,G1})^2}{2} + \frac{(3\sigma_{CD,G2})^2}{2} \\ &+ \left(\frac{3}{2}\left(\mu_{CD,G1} - \mu_{CD,G2}\right)\right)^2\end{aligned} \tag{4.1}$$

where $G1$ and $G2$ are the two different groups of CD populations. Dusa et al. observed about 20% of 3σ CD variation to the mean CD from the pooled CD model for $32nm$ DPL process. Table 4.13 shows, for various CD mean differences between $G1$ and $G2$, the CD mean and sigma values for the bimodal distribution, and for the corresponding unimodal distributions as calculated using Equation (4.1) for $50nm$ target CD.

Table 4.13: Mean and sigma of bimodal and pooled unimodal CD distributions.

		$G1$		$G2$	
		Mean (nm)	3σ (nm)	Mean (nm)	3σ (nm)
Mean Diff.	Unimodal	50.00	2.00	-	-
$0nm$	Pooled uni.	50.00	2.00	-	-
	Bimodal	50.00	2.00	50.00	2.00
$1nm$	Pooled uni.	50.00	2.50	-	-
	Bimodal	49.50	2.00	50.50	2.00
$2nm$	Pooled uni.	50.00	3.61	-	-
	Bimodal	49.00	2.00	51.00	2.00
$3nm$	Pooled uni.	50.00	4.92	-	-
	Bimodal	48.50	2.00	51.50	2.00
$4nm$	Pooled uni.	50.00	6.32	-	-
	Bimodal	48.00	2.00	52.00	2.00
$5nm$	Pooled uni.	50.00	7.76	-	-
	Bimodal	47.50	2.00	52.50	2.00
$6nm$	Pooled uni.	50.00	9.22	-	-
	Bimodal	47.00	2.00	53.00	2.00

As seen in the table, overall CD variation of the unimodal representation in Column 4 increases with the increasing mean difference between CD groups. This increased variation will necessarily increase the guardband of the design process, and in turn will worsen optimization and design closure runtime, as well as standard design metrics such as area, wirelength, violations, etc., as recently reported by Jeong et al. [97] (cf. the discussion of Figure 4.26 below).

Path delay variation in DPL. We refer to the different CD distributions as corresponding to the different *colorings* (i.e., mask exposures) of the gate polys in a cell layout. In DPL coloring, adjacent minimum-pitch poly lines must be colored differently. Thus, a cell can have (at least) two basic versions according to its coloring sequence, as shown in Figure 4.20. To distinguish between these different colorings, when the cell is instantiated in standard, "North" orientation. we use C_{12} (respectively, C_{21}) to refer to a cell in which the first or leftmost poly is colored by CD group1 (respectively, CD group2), the second poly is colored by CD group2 (CD group1), and so on. It is important to note that regardless of whether a cell has an odd number of polys or an even number of polys, and cells' placement locations and orientations, there will exist two different colorings for the cell, based on which color is assigned to the first (leftmost) poly. We discuss the key impacts of the bimodal CD distribution: on path delay variation, on timing slack variation, and on the design process.

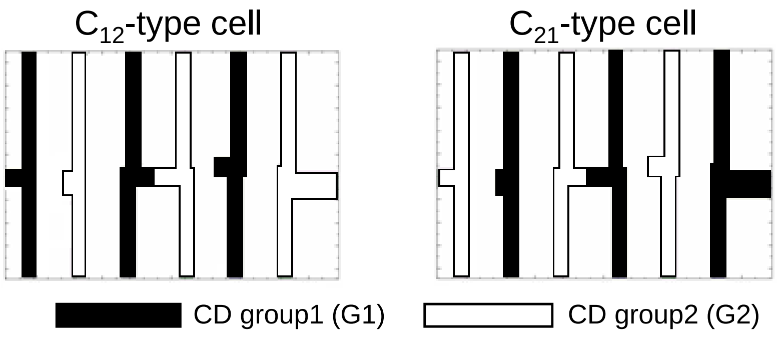


Figure 4.20: Example of two different DPL colorings for a NOR3 cell.

Every cell instance in a design can be colored differently according to its location and the surrounding cell instances. Therefore, instances of the same master cell in a timing path can be differently colored, and can have different electrical behaviors. Due to the loss of the spatial correlation between differently colored cells, delays across cell types (C_{12} and C_{21}) in a path can vary randomly or with less correlation, even while cells of the same type coloring have strong correlation. Finding the path delay variation of a timing path in the presence of bimodal CD distribution requires solution of the following problem formulation.

Bimodal Path Delay Variation Analysis. Given m cells g_i of C_{12} type and n cells q_j of C_{21} type in a timing path, determine the delay variation of the timing path, subject to the constraints:

$$(a)\ \ Min_{i,j}cov(g_i, g_j) > Max_{i,j}cov(g_i, q_j)$$
$$(b)\ \ Min_{i,j}cov(q_i, q_j) > Max_{i,j}cov(g_i, q_j)$$

According to the constraints, the covariance between cells in the same group is larger than the covariance between cells in different groups.

The delay variation of a delay path is:

$$\begin{aligned} \sigma^2(d(path)) &= \sigma^2(\sum_i(d(g_i)) + \sum_j(d(q_j))) \\ &= \sum_i \sigma^2(d(g_i)) + \sum_j \sigma^2(d(q_j)) \\ +2\sum_{i_1,i_2} cov(g_{i_1}, g_{i_2}) &+ 2\sum_{j_1,j_2} cov(q_{j_1}, q_{j_2}) + 2\sum_{i,j} cov(g_i, q_j) \end{aligned} \tag{4.2}$$

From Equation (4.2), since $cov(g_i, q_j)$ is small (e.g., zero in the case of no correlation), the path delay variation for a path composed of uncorrelated different types of cells is smaller than that of a path composed of only correlated cells.

Recall that for the DPL process, patterns are first partitioned into two groups, and that the two groups are each assigned a distinct color. The constraint is that same-color patterns should not be placed within the minimum distance that is permitted by the litho and etch equipment. According to the placement locations, orientations and the neighboring cells, a cell can be colored in different ways. Figure 4.21 shows the delay variation of 4-stage inverter chains and buffer chains for all possible colorings of cells for two different CD mean differences, 0nm in (a) and 3nm in (b). We measure the delay of the timing paths across the four combinations of extreme CD corners (Min and Max CD values for each CD group). The x-axis shows the different path coloring sequences, and the legends in the figure show the combinations of the extreme corners of each CD group. Note that even for such a simple timing path, the number of required timing analyses in the DPL regime increases exponentially with the number of stages.

For this experiment, we use the $45nm$ bulk CMOS SPICE model from the University of Arizona's Predictive Technology Model website [17] and $45nm$ circuits from Nangate Open Cell Libraries [14]. We assume that the CD values of each CD group have perfect spatial correlation, so as to isolate the impact of bimodality as well as to reduce the number of experiments. The number of configurations of each path, accounting for different colorings and process corners, is $4 \cdot 2^4 = 64$. Table 4.14 shows all possible CD corners (Column 1), and all possible coloring sequences (Column 3), in the 4-stage inverter and buffer chains. Right arrows (→) imply the logical signal propagations, and cells can be placed anywhere in a die.

We assume the CD variation within each CD group to be $2nm$, which is comparable to the ITRS predicted value for CD control in the $45nm$ node, i.e., $1.9nm$ [11].[13] Finally, we measure the delay of the 64 different path configurations while sweeping the difference of means between CD group1 and CD group2 from $0nm$ to $6nm$. We also compare the delay estimated from the pooled unimodal CD model (ref. Table 4.13) with that estimated from the more realistic bimodal CD model.

From this study, we observe that for most cases, the delay values are within the boundary of the delay at the MAX-MAX and MIN-MIN corners, and that most results from bimodal analysis are within the window established by the pooled unimodal model. However, not all cases are covered by the pooled model when the mean CD difference between the two groups is $0nm$. In addition, delay variation increases when the mean difference between the two CD groups increases. Note that the delay variation of pooled unimodal cases becomes significantly larger than that for the bimodal cases when the mean CD difference becomes nonzero, as shown in Figure 4.21(b). This immediately raises the question as to whether the pessimism of a pooled unimodal delay model (i.e.,

[13] As noted in the earlier review of double exposure DPL technology, since overlay control in the $45nm$ node is $9nm$, it is difficult to use the negative double exposure process in light of the CD variation requirement. Hence, we do not consider the negative correlation between CD groups that would result with double exposure DPL, and we assume that CD variation is determined only by CD control capability.

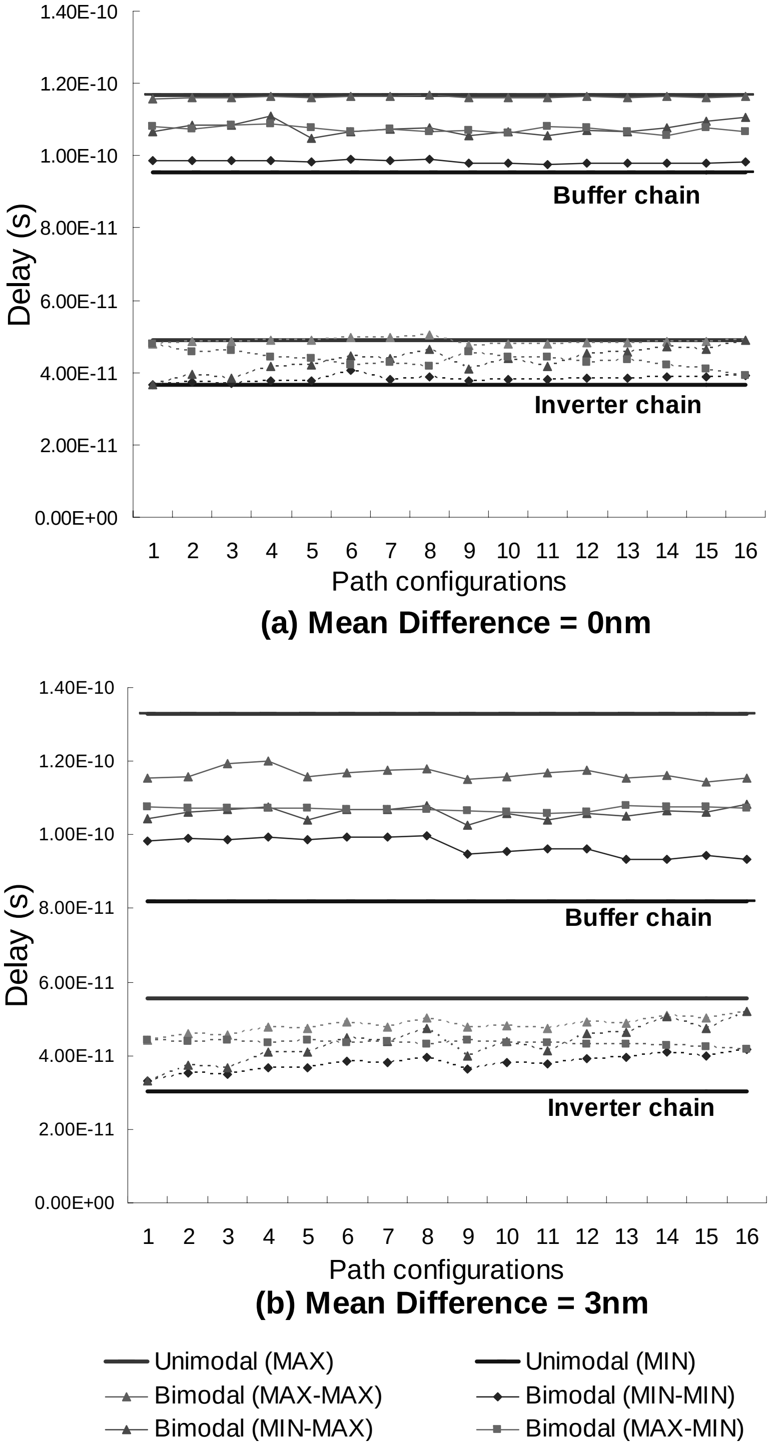
Table 4.14: Path configurations for 4-stage inverter and buffer chains.

CD corner	Path coloring configuration	
$G1$ group - $G2$ group		
MAX - MAX MAX - MIN MIN - MAX MIN - MIN	1	$C_{12} \rightarrow C_{12} \rightarrow C_{12} \rightarrow C_{12}$
	2	$C_{12} \rightarrow C_{12} \rightarrow C_{12} \rightarrow C_{21}$
	3	$C_{12} \rightarrow C_{12} \rightarrow C_{21} \rightarrow C_{12}$
	4	$C_{12} \rightarrow C_{12} \rightarrow C_{21} \rightarrow C_{21}$
	5	$C_{12} \rightarrow C_{21} \rightarrow C_{12} \rightarrow C_{12}$
	6	$C_{12} \rightarrow C_{21} \rightarrow C_{12} \rightarrow C_{21}$
	7	$C_{12} \rightarrow C_{21} \rightarrow C_{21} \rightarrow C_{12}$
	8	$C_{12} \rightarrow C_{21} \rightarrow C_{21} \rightarrow C_{21}$
	9	$C_{21} \rightarrow C_{12} \rightarrow C_{12} \rightarrow C_{12}$
	10	$C_{21} \rightarrow C_{12} \rightarrow C_{12} \rightarrow C_{21}$
	11	$C_{21} \rightarrow C_{12} \rightarrow C_{21} \rightarrow C_{12}$
	12	$C_{21} \rightarrow C_{12} \rightarrow C_{21} \rightarrow C_{21}$
	13	$C_{21} \rightarrow C_{21} \rightarrow C_{12} \rightarrow C_{12}$
	14	$C_{21} \rightarrow C_{21} \rightarrow C_{12} \rightarrow C_{21}$
	15	$C_{21} \rightarrow C_{21} \rightarrow C_{21} \rightarrow C_{12}$
	16	$C_{21} \rightarrow C_{21} \rightarrow C_{21} \rightarrow C_{21}$

today's standard practice) will be too costly in the DPL regime. We also observe that for skewed processes (MAX-MIN or MIN-MAX), delay variation across all the path configurations is larger than for MAX-MAX or MIN-MIN.

Figure 4.22 shows delay variations of a 16-stage inverter chain, normalized to mean values. Here, only four (out of 2^{16}) path colorings are studied: (i) C_{12}-only, (ii) C_{12}-C_{21}-C_{12}-... alternation, (iii) C_{21}-C_{12}-C_{21}-... alternation, and (iv) C_{21}-only. Corresponding to the analytical solution in 4.2, alternative coloring of the timing path shows smaller delay variations.

Timing slack variation in DPL. While path delay variation can be reduced by the bimodal CD distribution, we find a very different situation with variation of timing slack



(a) Mean Difference = 0nm

(b) Mean Difference = 3nm

Figure 4.21: Delay variations of 4-inverter and 4-buffer chains. Path configurations are as given in Table 4.14.

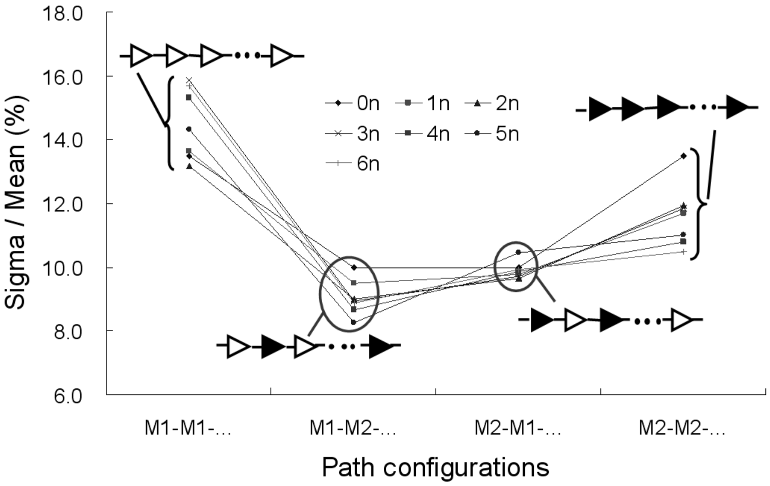


Figure 4.22: Relative delay variation σ/μ ($\%$) over all process corners.

– which is the most important parameter for design timing. Timing slack (t_{slack}) of the design is defined from clock-path delay (t_{clock}), clock cycle time (t_{cycle})) and data-path delay (t_{data}) as

$$t_{slack} = t_{clock} + t_{cycle} - t_{data}. \tag{4.3}$$

The variation of the timing slack is calculated by

$$\sigma^2_{t_{slack}} = \sigma^2_{t_{clock}} + \sigma^2_{t_{data}} - 2cov\left(t_{clock}, t_{data}\right). \tag{4.4}$$

For a traditional single-exposure process, if we assume that spatial correlation is high, the covariance term in Equation (4.4) will reduce the slack variation. However, in DPL, since cells in the clock path can be colored in a different way from cells in the data path, the covariance term will be reduced to zero, so that timing slack variation becomes a sum of clock path and data path variations. To meet signoff timing constraints with this increased slack variation in DPL, designs will require more stringent and difficult timing optimization.

We illustrate this concept with Figure 4.23, which portrays the slack calculation for the traditional single-exposure process in (a), and for the DPL process in (b). In this simple example, we assume that nominal delay of both clock and data path are $10ns$, and, following the analysis of path delay variation in Equation (4.2), we assume that DPL has smaller delay variation than the single exposure, e.g., $\pm 5ns$ for single exposure and $\pm 2ns$ for DPL.

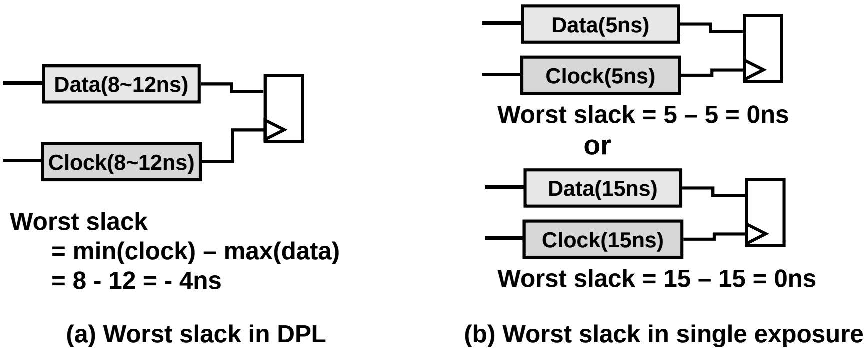


Figure 4.23: Worst timing-slack calculation in the DPL and (traditional) single-exposure regimes.

In the single-exposure case, due to the strong spatial correlation between the clock path and the data path, process variation does not make timing slack worse. However, in the DPL case, although the delay variation is small, we can see large negative slack, due to the weak correlation between clock and data path – that is, each path delay can be varied independently.

To see more explicitly and realistically the impact of bimodal CD distribution on the timing slack, we extract a topmost critical path from the AES core, obtained as RTL from the open-source site $opencores.org$ [15], which synthesizes to 40K instances, and is placed and routed with a reduced set of $45nm$ library cells. Both the launching and capturing clock paths are composed of 14 stages of inverters, respectively. Also, the launching and capturing clock paths share the initial 4 stages of inverters, but differ from each other in the latter 10 stages of each path. The data path is composed of 30 logic stages, e.g., 2-input NAND, NOR, OR and AND logic cells, and 1-input BUF and INV cells. An exhaustive design of experiments (DOE) would require $4 \cdot 2^{54}$ cases. We reduce the DOE complexity by restricting alternatives for the clock paths, the combinational data path, and registers.

First, we assume that the colorings of all cells in the data path are fixed. This allows us to evaluate the impact of bimodal CD distribution only on the clock design. Second, the number of clock path configurations still remains very large ($4 \cdot 2^{24}$), so we further limit the experiments to the 5 extreme cases shown in Table 4.15.

For a design to operate correctly, data signals must be carried from one (launch-

Table 4.15: Coloring configurations of the critical path example.

	Data path	Launching clock path	Capturing clock path
Case 1		$C_{12} \rightarrow C_{12} \rightarrow$...	$C_{12} \rightarrow C_{12} \rightarrow$...
Case 2		$C_{21} \rightarrow C_{21} \rightarrow$...	$C_{21} \rightarrow C_{21} \rightarrow$...
Case 3	$C_{12} \rightarrow C_{12}$...	$C_{12} \rightarrow C_{12} \rightarrow$...	$C_{21} \rightarrow C_{21} \rightarrow$...
Case 4		$C_{21} \rightarrow C_{21} \rightarrow$...	$C_{12} \rightarrow C_{12} \rightarrow$...
Case 5		$C_{12} \rightarrow C_{21} \rightarrow$...	$C_{12} \rightarrow C_{21} \rightarrow$...

ing) register to the next (capturing) register once per clock cycle. The timing slacks for setup and hold time are defined by[14]

- Setup timing slack:

$$\begin{aligned} t_{slack,setup} &= t_{RAT,setup} - t_{AAT,setup} \\ &= (t_{capture} - t_{launch}) + t_{cycle} - t_{setup} - t_{data} \geq 0 \end{aligned} \tag{4.5}$$

- Hold timing slack:

$$\begin{aligned} t_{slack,hold} &= t_{AAT,hold} - t_{RAT,hold} \\ &= (t_{launch} - t_{capture}) + t_{data} - t_{hold} \geq 0 \end{aligned} \tag{4.6}$$

The difference of delays between launching and capturing clock paths, i.e., *clock skew*, plays an important role in both the setup and hold timing slacks. If $t_{capture}$ is greater (resp. smaller) than t_{launch}, this increases (decreases) setup time slack but decreases (increases) hold time slack regardless of data path delay. Therefore, however well one optimizes the circuit to have zero slack, an unbalanced clock network can create clock skew and cause timing problems by either setup or hold time violations. Figure 4.24 shows the maximum skew that occurs as a result of the bimodal CD distribution, across the path coloring sequences shown in Table 4.15. Note that the clock skew is originally designed to be zero. Intuitively, we can expect that there is no clock skew

[14] We use the standard acronyms of AAT for actual arrival time, and RAT for required arrival time.

when the coloring sequences of both clock paths are the same, i.e., Cases 1, 2 and 5. However, even when the mean difference between two CD groups is zero, Cases 3 and 4 show substantial clock skew due to the different coloring sequences of launching and capturing clock paths, and the skew increases when the mean CD difference increases. The maximum clock skews of Cases 3 and 4 with $0nm$ mean CD difference are $22.7ps$ for each, and these skews increase up to $52.2ps$ and $53.4ps$, respectively, with $6nm$ mean difference. Another implication of Figure 4.24 is that the pooled unimodal CD representation cannot discern the potential skew-related timing problems in DPL designs, even though the pooled model accounts for the physical distribution of CDs, and is very pessimistic with respect to CD corners. This is because the pooled CD model cannot distinguish the colorings of paths.

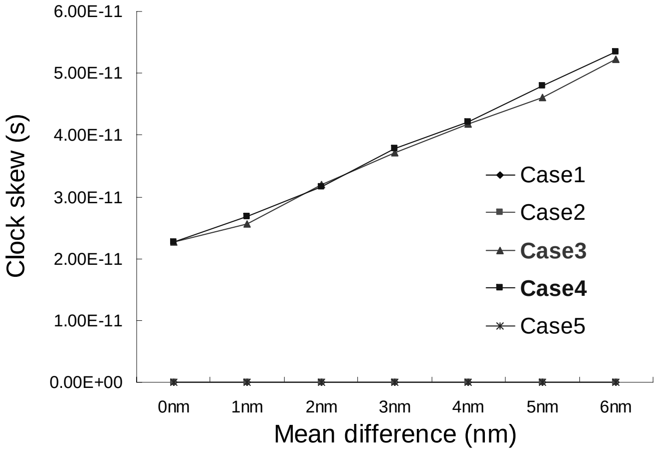


Figure 4.24: Clock skew versus CD mean difference between CD groups, across combinations of process corners. Cases 1, 2 and 5 are superposed on the x-axis.

Figure 4.25 shows the slack changes of each coloring sequence of clock paths versus the mean difference of the CD groups at the worst CD corner combination (MAX-MAX). The timing path originally has zero slack when the CD mean difference is zero (i.e., two color groups have same CD mean). For Case 4, since the delay of the capturing (resp. launching) clock path decreases (increases), the slack becomes negative[15]; this will worsen when the number of stages of the clock network increases. For Cases 1, 2, 3 and 5, delay of the capturing clock path is greater than that of the launching clock

[15] With $6nm$ mean CD difference, -$18ps$ of slack violation occurs. This value is about 10% of the clock path delay of the used testcases.

path, so that the slack is still positive or even improved. However, since the improved slack on this path is only from clock skew, there can easily be a resulting timing problem for the next timing path that starts with this path's capturing register, or increased hold time violations per Equation (4.6). We also notice again that the pooled unimodal CD representation shows unnecessarily pessimistic setup timing slack values.

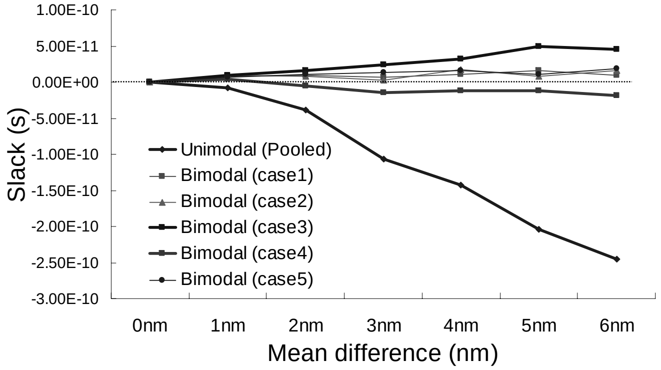


Figure 4.25: Timing slack versus CD mean difference between CD groups across combinations of process corners.

Guardband and design process in DPL. The simplest way to consider the bimodal CD distribution in the design process is to model bimodal as unimodal. The already-cited pooled unimodal CD model from [64] can be useful, and today's conventional flow can still be used. However, the pooled unimodal model gives a too-pessimistic guardband, which can lead to significant overdesign.

Figure 4.26 shows the best-case and worst-case delays of the $45nm$ INV cell for each of pooled unimodal and bimodal, with mean difference between the two CD groups in x-axis. Delay difference between worst and best shows the size of the guardband. As seen in the figure, simple unimodal modeling will lead to more than $2\times$ increase of guardband, even for the small mean difference cases; according to the recent study on guardband impact in [97], this will lead to over 15%, 39% and 14% of area, runtime and wirelength increase, respectively.

To reduce such pessimism in unimodal representation, separate timing models for each CD group are required. However, this increases the difficulty of circuit optimization. Placement location and surrounding patterns will determine the timing model of a cell instance, since these factors affect the DPL coloring. Consequently, even slight

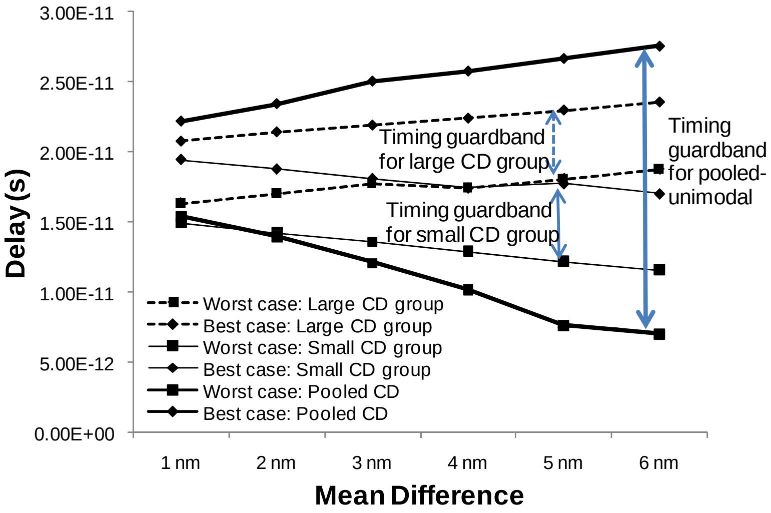


Figure 4.26: Timing guardband for each characterization method.

cell movement or resizing can give large and non-obvious changes in delay values under skewed process combinations, i.e., MIN-MAX or MAX-MIN. This may lead to more physical design iterations, since at every ECO placement step, cells' timings can be changed by the applied DPL patterning and coloring solution.

From the above results and discussions, we can conclude that a pooled unimodal representation with pessimistic corner values does not suffice in the future of DPL, and furthermore, as we demonstrated above, the pooled unimodal model cannot capture the potential timing problems caused by uncorrelated data and clock delay variations. To deal with the challenges presented by the bimodal CD distribution, novel timing analysis and optimization methodologies are required.

4.4 IAL Focus-Exposure Process Window Analysis

The challenge for interference-assisted lithography (IAL) is to convert existing designs to 1-D regular pitch and regular linewidth. This challenge has a number of aspects, including resolution limits for the required minimum pitch to avoid CD variation during the second (trim) exposure. Requirements for device sizing, reliability and power are also key factors in migration to IAL.

4.4.1 Considerations in IAL

CD variation of neighboring line. CDs of patterns generated by IL can be affected during trim mask exposure, i.e., through exposure dose and pattern distortion. An optical proximity correction (OPC) approach such as biasing the CD of a neighboring line is not feasible to reduce the impact of such CD variation in IAL. Fritze et al. [67] show that the second PL exposure produces 20% CD variation when removing a single line, and 5% variation when removing two lines. Fundamentally, the greater the second exposure dose, the greater its impact on CD of the neighboring patterns.

Overlay sensitivity. IAL implementation requires tight overlay control. Fritze et al. [66] simulated double exposure with $15nm$ overlay range. It is reported in [66] that $15nm$ overlay when removing two lines results in a very significant $1.3\times$ dose variation.

Variety of pattern shapes. The drawback of IL is that it can print only one-dimensional (1-D) regular pitch and linewidth patterns on a layer. By contrast, today's typical logic layouts employ a multiplicity of linewidths, pitches and shapes. On metal layers, chemical-mechanical planarization (CMP) dummy fill shapes are required to meet minimum density rules as shown in Figure 4.27. However, IAL cannot generate such irregular metal shapes. Moreover, two-dimensional layouts are required for via doubling; the via landing pad at a metal line-end may require various linewidths, and non-preferred direction routings (e.g., jogs and bendings) are used to minimize the number of vias or wirelength. The ubiquity of such metal patterns in modern layouts can only delay the application of IAL to real devices.

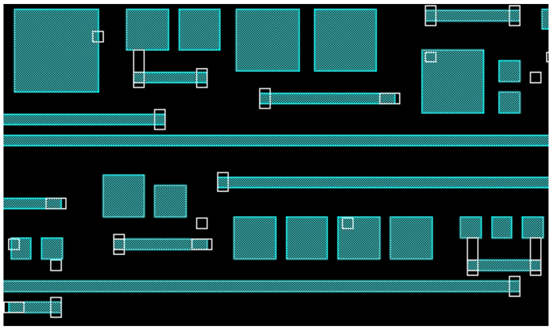


Figure 4.27: Problems of IAL application due to dummy metal fill.

4.4.2 Process Window Analysis

We compare the process window of IAL to a single exposure process for poly layer. In this experiment, the single exposure is simulated with a 6%-attenuated 180°-phase mask with numerical aperture of 1.2. An ASML-type cross-quad illuminator shape is used with x-y azimuthal-like polarization optimized for x- and y-oriented pitches with $\sigma_{xCenter} = 0.27$, $\sigma_{yCenter} = 0.89$ and $\sigma_{Width} = 0.15$. We study critical dimension (CD) tolerance of $\pm 10\%$ CD without considering actual function of the pattern. As shown in Figure 4.28, CD at the diffusion edge near the line-end (CD$_{Taper}$) is typically narrower than CD at the other diffusion edge (CD$_{Inner}$). We allow CD$_{Inner}$ to be in the range from 29nm to 35nm, and the gap between the two opposing ends (CD$_{Gap}$) to be in the range from 10nm to 45nm to not scum on the resist between the opposing poly ends. We then find the condition that maximizes the ratio of the CD$_{Taper}$ to CD$_{Inner}$. Table 4.16 summarizes minimum and maximum boundary conditions for each CD parameter in Figure 4.28. Within these boundaries, we compare the image quality of IAL with that of the single exposure.

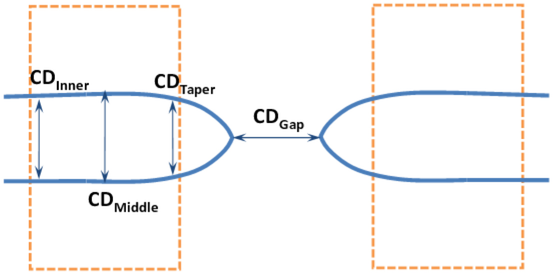


Figure 4.28: Lithographic metrics for critical dimension (CD).

Table 4.16: Critical dimension targets.

Parameter	Minimum (nm)	Maximum (nm)
CD$_{Gap}$	10	45
CD$_{Inner}$	22	27
CD$_{Middle}$	22.5	27.5
CD$_{Taper}$	10	35

Figure 4.29 (resp. 4.30) compares CD_{Gap} (resp. CD_{Taper}) of single exposure with that of IAL from various focus-exposure combinations. We observe that CD variation of IAL is significantly smaller than that of single exposure. Figure 4.31 compares process windows of single exposure and IAL with CD_{Gap} = $30nm$. IAL results in 1.7× exposure latitude at $0.10\mu m$ depth of focus (DOF), and 2.5× maximum DOF compared to the single exposure process. From the figure, we can conclude that IAL provides significantly greater focus and exposure tolerance compared to the single exposure process, for the same minimum CD_{Taper}, which is the most important factor limiting the overall process window of IAL.

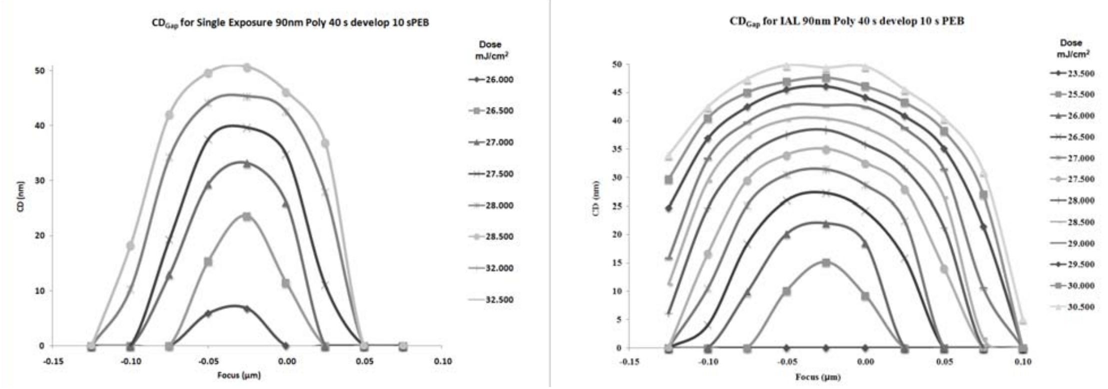


Figure 4.29: Focus-exposure simulation results of the reference single exposure (left) and IAL (right) for CD_{Gap}.

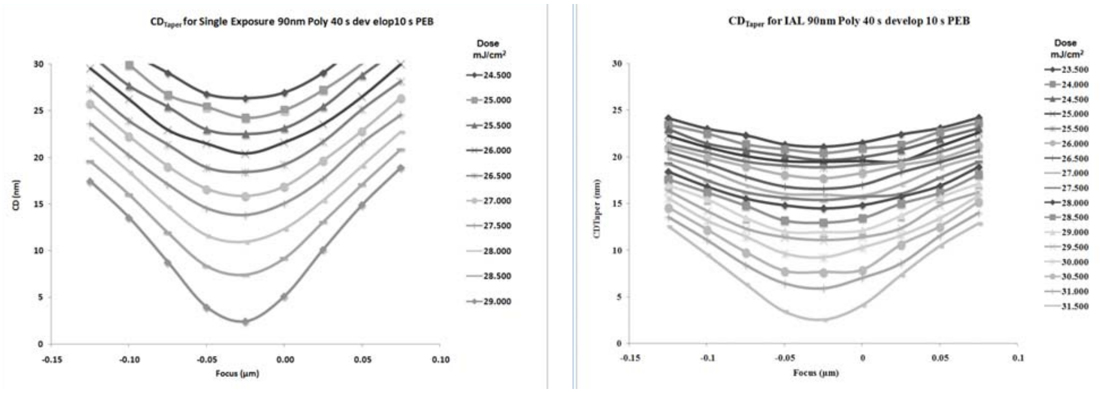


Figure 4.30: Focus-exposure simulation results of the reference single exposure (left) and IAL (right) for CD_{Taper}.

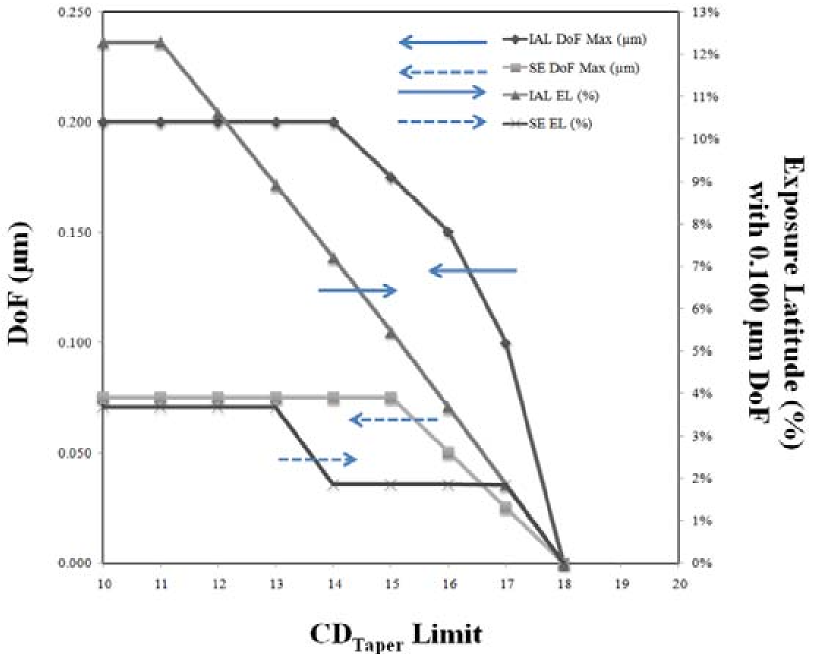


Figure 4.31: Process window comparison for IAL and single exposure process with respect to the minimum allowable CD$_{Taper}$.

4.5 Conclusions and Research Directions

We have addressed various double patterning options, and analyzed detailed mechanisms of additional variations in double patterning for BEOL as well as FEOL layers.

For BEOL, we have provided a variational interconnect analysis framework for double patterning lithography, taking overlay into account. We have applied the proposed framework to testcases ranging from a small representative interconnect structure to chip-level designs based on a $45nm$ technology, with golden extraction and timing analysis tools. We obtain the following conclusions, which may help process technology developers to assess double patterning lithography options in terms of chip-level performance and variability.

1. Overlay with indirect alignment (IA) results in higher capacitance variations than direct alignment (DA) in DE or DP.

2. For all DPL techniques, more than 10% interconnect capacitance variation can occur due to overlay or spacer thickness variation.

3. Design timing can be significantly degraded due to the large capacitance variation, e.g., up to 13% worse total negative slack in N-SADP with 3σ of spacer thickness variation.

4. SADP may require track-type metal fills for chemical-mechanical polishing constraints. Hence, performance degradation due to fill may be larger for SADP in production designs. Furthermore, mask coloring and design will be difficult. SADP has tighter variability control, but is an expensive option in terms of design rules and restrictions.

5. Given the potential disadvantages of SADP, P-DE/DP may be the most favorable option for BEOL double patterning lithography based on performance. With the same 3σ variation control ($12nm$), the coupling-induced delay variation in P-DE/DP is half that of N-DE/DP.

6. When variation specifications differ, e.g., 3σ for DE/DP and 1σ for SADP, the amounts of coupling-induced delay variation can be similar. Designers and lithographers must then consider design cost and cost of ownership associated with these technology options.

The study of overlay impacts may shed light onto which technology should be preferred, at least from a performance-oriented perspective. Furthermore, the framework we provide for DPL variability analysis can be used in the analysis and optimization of interconnects once a particular DPL method is chosen as a technology.

For FEOL, we have shown that 'bimodal' CD distribution and loss of spatial correlation between differently colored (exposure) cells have far-reaching impacts on circuit properties that are neither well-defined nor well-studied. We have given both analytic and empirical assessments of the potential impact of DPL on timing analysis error and guardbanding. We observe that the traditional 'unimodal' characterization and analysis framework may not be viable for DPL. For example, experimental analyses demonstrate that different mask layouts can result in 20% or more change in timing path delays. As shown in Figure 4.26 and Table 5.8, design guardband and timing slack in double patterning can each degrade by up to $2\times$, and this will significantly hinder the

13% per year of device performance improvement expected in [11] (cf. studies of 'cost of guardband' in [97]).

We have also shown that a new maskless lithography technique IAL has a larger process window compared to traditional single exposure lithography, through focus-exposure process window simulations.

4.6 Acknowledgments

Chapter 4 is in part a reprint of "Assessing Chip-Level Impact of Double Patterning Lithography", *Proc. International Symposium on Quality Electronic Design*, 2010, "Is Overlay Error More Important Than Interconnect Variations in Double Patterning?", *Proc. ACM International Workshop on System-Level Interconnect Prediction*, 2009, "Timing Analysis and Optimization Implications of Bimodal CD Distribution in Double Patterning Lithography", *Proc. Asia and South Pacific Design Automation Conference*, 2009, "32nm 1-D Regular Pitch SRAM Bitcell Design for Interference-Assisted Lithography", *Proc. SPIE BACUS Symposium on Photomask Technology and Management*, 2008, and "Interference Assisted Lithography for Patterning of 1D Gridded Design", *Proc. SPIE Symposium on Advanced Lithography*, 2009. I would like to thank my coauthors Mr. Robert T. Greenway, Dr. Rudolf Hendel, Professor Andrew B. Kahng, Dr. Chul-Hong Park, Mr. John S. Petersen, Dr. Zhilong Rao, Dr. Michael C. Smayling and Dr. Rasit O. Topaloglu.

I also thank Dr. Harry J. Levinson, Dr. Jongwook Kye and Dr. Yuansheng Ma of GLOBALFOUNDRIES for discussions on double patterning lithography, and Prof. Puneet Gupta and Mr. Rani S. Ghaida of UCLA for discussions on overlay.

Chapter 5

Design-Aware Manufacturing Process Optimization

In this chapter, we discuss two design-aware manufacturing process optimizations. The first optimization focuses on reticle generation, and encompasses mask cost, lithography cost and yield. We provide new yield-aware mask strategies to mitigate emerging variability and defectivity challenges. To address *variability*, we analyze CD variability with respect to reticle size, and its impact on parametric yield. With a cost model that incorporates mask, wafer, and processing cost considering throughput, yield, and manufacturing volume, we assess various reticle strategies (e.g., single-layer reticle (SLR), multi-layer reticle (MLR), and small and large reticle size) considering field size-dependent parametric yield. To address *defectivity*, we compare parametric yield due to EUVL mask blank defects for various reticle strategies in conjunction with reticle floorplan optimizations such as shifting of the mask pattern within a mask blank to avoid defects being superposed by performance-critical patterns of a design.

The second optimization focuses on lithography dose optimization using advanced dose control in manufacturing equipment. We propose to exploit the recent availability of fine-grain dose control in the step-and-scan tool to achieve manufacturing-time (yield-aware dose mapping) optimizations of timing yield and leakage power. We formulate the placement-aware dose map optimization as quadratic and quadratically constrained programs which are tractable to efficient solvers.

5.1 Cost-Driven Reticle Strategy Optimization

Photomask cost is a highly critical, non-recurring component of manufacturing cost. Semiconductor manufacturers have long sought cost-effective photomask strategies. Multiple copies of a single layer of one product IC are patterned in a full-size mask blank to obtain a *single-layer reticle* (SLR), used in most high-volume products. In a *multi-project reticle*, the same layer (e.g., M3) of several different products ICs is implemented on a single reticle; this allows sharing of mask costs between individual product owners. Beyond a “single-layer-per-reticle” strategy, multi-layer and multi-product strategies are also implemented on a single reticle [35], and an algorithm to enable the layer placement and quality check procedure according to a parameterized cost function is proposed in [36]. In addition, reticle size and number of dies per reticle are other knobs that can be tweaked by manufacturers or designers.

IC manufacturing, traditionally uses the maximum possible reticle size. This is commonly believed to maximize litho tool throughput and minimize manufacturing cost. However, as reticle size increases, the mask cost (write, inspection, defect disposition, repair, etc.) also increases. For high-volume products, mask cost can be disregarded, but for low-volume products – in light of shuttle-based prototyping, design revisions and respins, market competition, and other factors – mask cost can significantly impact overall cost per die. Mask writing cost, lithography cost, and mask yield all vary with reticle size. Also, larger reticles can result in larger CD variation in silicon, leading to parametric yield loss that potentially increases manufacturing cost, even for high-volume products. Hence, a new cost model is required to comprehend reticle size-dependent cost changes.

Besides the issue of variability, defectivity (notably, mask blank defects in extreme ultraviolet lithography (EUVL)) looms as a critical issue for mask generation and product yield. EUVL uses reflective masks instead of the traditional optical transmission masks. EUVL mask blanks contain a stack of 40 to 50 Mo-Si alternating layers, to maximize reflection at 13.5nm wavelength. Each of these layers requires a discrete processing step, hence defects at each layer can accumulate [44]. Defects in multi-layer EUVL blanks are difficult to detect and repair, and manifest as distortions of image placement [90]. An EUVL buried mask defect is known to cause critical dimension

(CD) change [59]. Such CD changes may not cause catastrophic defects in the IC product, but can cause parametric yield loss through timing failures. Since EUVL mask blanks are not anticipated to be completely defect-free, a new reticle floorplan method is required to deal with defective mask blanks. Burns et al. [44] propose mask pattern translation and rotation in a mask blank to avoid the placement of critical mask patterns on defects. Such freedom in reticle floorplanning also depends on reticle size.

5.1.1 Reticle Strategies

A reticle contains one or more dies, and all dies in a reticle are printed at the same time. We study the following strategies.

- *Single-layer reticle on large field (SLR-L):* a reticle contains one processing layer for many copies of a die as shown in Figure 5.1(a). This is the traditional mask strategy.
- *Single-layer reticle on small field (SLR-S):* a reticle contains one processing layer for one or a small number of dies as shown in Figure 5.1(b). Lithography throughput may be reduced, but mask cost can also be reduced.
- *Multi-layer reticle on large field (MLR):* a reticle contains multiple layers (e.g., M1, M2, etc.) of a design as shown in Figure 5.1(c). When printing one layer, the other regions (i.e., other layers) of the reticle are blocked using a mechanism called *blading* [35]. The number of reticles for a design can be reduced.

5.1.2 Cost Model

SEMATECH has for many years provided guidance on mask costs and their potential effects on product cost. The 1997 SEMATECH mask cost of ownership (COO) model [177] included actual manufacturing process steps. A revision in the year 2000 added mask processing time to the cost model. In 2001, the mask COO model was revised to reflect technology acceleration, i.e., a 2-year cycle of technology improvement, instead of the previously assumed 3-year cycle. Mask set cost is obtained as the sum of costs for all masks in the set; mask set costs are rising due to the increase in individual

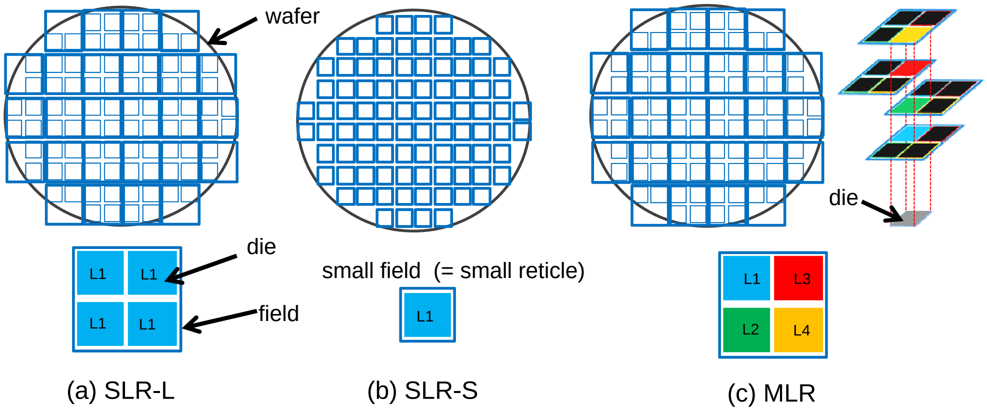


Figure 5.1: Examples of mask strategies: (a) single-layer reticle on a traditional large field, (b) single-layer reticle on a small field, and (c) multi-layer reticle on a large field.

mask cost as well as an increase in the total number of masks in a mask set. The work of Trybula [176] reviews the SEMATECH methodology to ensure that projected mask costs reflect planned geometries. Grenon [72] observes that the largest mask cost improvements come from higher defect repair yields, and proposes mask cost projections considering new or improved mask repair technologies such as focused ion beam (FIB), nano machining, and femtosecond laser repair.

The work of Pramanik et al. [147] analyzes the cost of various reticle strategies based on the SEMATECH cost of ownership model. Although Pramanik et al. model the costs of mask and lithography with respect to the field size, they mainly focus on mask generation cost (including mask yield) and stepper cost; they do not consider the parametric yield variation of silicon dies. Our present work extends that of Pramanik et al. by integrating the impact of field size on CD variation in silicon observed from recent 65nm and 45nm foundry data, and by then reevaluating the manufacturing cost of various reticle strategies with 45nm mask and lithography costs scaled from 90nm technology values.

90nm mask cost model [147]. Each reticle strategy is differentiated by the number of dies per field. To represent mask cost considering the number of dies per field, we use the following parameters.

- w_f: field width on wafer, in mm
- h_f: field height on wafer, in mm

- M: mask reduction factor (in general, 4)
- n_{row}: number of rows of dies per field
- n_{col}: number of columns of dies per field
- $n_{m,vc}$: number of masks for very critical layers (e.g., $193nm$)
- $n_{m,c}$: number of masks for critical layers (e.g., $248nm$)
- $n_{m,nc}$: number of masks for noncritical layers (e.g., I-line)
- n_m (= $n_{m,vc} + n_{m,c} + n_{m,nc}$): total number of masks

The key contributors to mask costs are time-dependent cost (i.e., mask writing/inspection time) and yield-dependent cost. Mask writing/inspection time is proportional to the mask area and the mask resolution. Mask area is calculated based on how many dies are in a mask. To reflect cost differences due to mask resolution, scaling factors are used. Writing and inspection times for very critical (resp. critical) layers are assumed to be $4\times$ (resp. $2\times$) larger than corresponding times for noncritical layers. The combined time-dependent cost is calculated as

$$cost_{time} = r_{res} \cdot t_{min} \cdot A$$

where r_{res} is the cost scaling factor for mask resolution, t_{min} is the writing/inspection time for noncritical layers normalized to a unit area, and A is the mask field area calculated as $w_f \cdot h_f$.

Mask yield is affected by critical dimension (CD) (Y_{cd}), image placement error (Y_{pl}), random defects (Y_{def}), and some other uncertainties (Y_{misc}). The overall yield of a mask layer is calculated as

$$Yield = Y_{cd} \cdot Y_{pl} \cdot Y_{def} \cdot Y_{misc}. \tag{5.1}$$

The *baseline mask yields* (Y^*) of full-size reticle for $90nm$ technology are assumed as $Y^*_{cd} = 90\%$, $Y^*_{pl} = 90\%$, $Y^*_{def} = 80\%$, and $Y^*_{misc} = 90\%$, with these values obtained from the third-year production yield of a typical $180nm$ node technology [147]. From Equation (5.1), the cumulative baseline mask yield is 58%. From the baseline yield values, yields for various reticle sizes are calculated, considering corner protrusion impacts p from different reticle sizes and a yield correction factor b. Corners of a square mask suffer from resist film thickness non-uniformity, which causes CD and image placement errors. Corner protrusion is the extension of a square field beyond the circular “stable region” in a mask, and is proportional to the diagonal of the mask field (i.e., $\sqrt{w_f^2 + h_f^2}$). The yield correction factor b is based on the idea of “bucketing” of yield-loss sources. Pramanik et al. [147] assume that a third of mask CD yield loss is from field size-dependent random variation, and another third from the corner protrusion effect. Each component of mask yield is then calculated from the baseline yield as

$$Y_{cd} = (Y^*_{cd})^{\left(1+w_f/w^*_f+p/p^*\right)/b}, \ Y_{def} = \left(Y^*_{def}\right)^{A/A^*}, \text{ and } Y_{pl} = \left(Y^*_{pl}\right)^{p/p^*}$$

where A^*, w^*_f, and p^* are the area, mask field width and corner protrusion of a $100{\times}100mm^2$ reference mask, respectively.

From time-dependent cost and yield-dependent cost, overall mask cost is calculated. Let the calculated cost of very critical, critical and noncritical layers in a mask set be $cost_{m,vc}$, $cost_{m,c}$, and $cost_{m,nc}$, respectively, and let the number of masks for corresponding mask layers be $n_{m,vc}$, $n_{m,c}$, and $n_{m,nc}$, respectively. The total mask set cost $Cost_{maskset}$ is calculated as

$$Cost_{maskset} = cost_{m,vc} \cdot n_{m,vc} + cost_{m,c} \cdot n_{m,c} + cost_{m,nc} \cdot n_{m,nc}.$$

Table 5.1 summarizes 90nm mask cost with respect to the field sizes shown in Table 4 of Pramanik et al. [147]. The numbers of very critical, critical and non-critical layers for 90nm were assumed as 8, 8 and 12, respectively.

Scaled 45nm mask cost. We use the 90nm cost model to estimate mask set cost for 45nm technology, based on the following assumptions.

- Mask cost doubles at the introduction year of each successive technology node.
- Mask cost for a given technology decreases at the rate of 20% per year.
- The introduction years of 90nm, 65nm and 45nm are 2003, 2005, and 2007, respectively.
- The number of mask layers for 45nm is 33 as predicted in the 2007 ITRS [11].
- The proportions of very critical, critical and non-critical layers are equal (i.e., 11 layers for each).

Table 5.1: 90nm mask cost from Pramanik et al. [147].

Field in mask ($mm\times mm$)	100×100	64×96	64×64	32×64	32×32
Field on wafer ($mm\times mm$)	25×25	16×24	16×16	8×16	8×8
Die in mask ($mm\times mm$)	32×32	32×32	32×32	32×32	32×32
Die on wafer ($mm\times mm$)	8×8	8×8	8×8	8×8	8×8
Number of dies per field	9	6	4	2	1
Mask cost per layer					
Very critical ($)	112,000	59,000	41,000	24,000	19,000
Critical ($)	28,000	20,000	15,000	11,000	9,000
Non-critical ($)	10,000	8,000	7,000	6,000	6,000
Mask set cost					
Very critical ($)	896,000	472,000	328,000	192,000	152,000
Critical ($)	224,000	160,000	120,000	88,000	72,000
Non-critical ($)	120,000	96,000	84,000	72,000	72,000
Overall mask set cost ($)	1,240,000	728,000	532,000	352,000	296,000

These assumptions imply a 45nm mask cost that is $4\times(0.8)^{(2011-2003)}$ times the 90nm initial mask cost; the factor 4 is from the two technology generations, and the mask cost is continuously reduced by 20% since the 90nm technology introduction year 2003. Table 5.2 shows the calculated mask set cost for 45nm. We observe that this cost is similar to the 90nm mask set cost, a conclusion that matches mask cost trends across several recent technology nodes.

Litho cost model. Total manufacturing cost depends on throughput. A smaller field is expected to cause lower throughput, since it requires a greater number of exposures. We calculate lithography cost as a function of mask field size.

Parameters that affect lithography cost are number of exposures per wafer n_e, cost of a single exposure $cost_e$, and number of mask layers n_m. The number of exposures is inversely proportional to the mask field size, and is calculated as the total number of dies per wafer divided by the number of dies per field. Then, litho cost per wafer ($cost_e \cdot n_e \cdot n_m$) is multiplied by the number of wafers developed n_w. Finally, the total

Table 5.2: 45nm mask cost scaled from 90nm mask cost.

Mask field size ($mm \times mm$)	100×100	64×96	64×64	32×64	32×32
Mask die size ($mm \times mm$)	32×32	32×32	32×32	32×32	32×32
Number of dies per field	9	6	4	2	1
Mask cost per layer					
Very critical ($)	75,162	39,594	27,515	16,106	12,751
Critical ($)	18,790	13,422	10,066	7,382	6,040
Non-critical ($)	6,711	5,369	4,698	4,027	4,027
Mask set cost					
Very critical ($)	826,781	435,537	302,661	177,167	140,258
Critical ($)	206,695	147,640	110,730	81,202	66,438
Non-critical ($)	73,820	59,056	51,674	44,292	44,292
Overall mask set cost ($)	1,107,296	642,232	465,064	302,661	250,987

lithography cost $Cost_{litho}$ is calculated as

$$Cost_{litho} = n_w \left(cost_{e,vc} \cdot n_{e,vc} \cdot n_{m,vc} + cost_{e,c} \cdot n_{e,c} \cdot n_{m,c} + cost_{e,nc} \cdot n_{e,nc} \cdot n_{m,nc} \right)$$

where subscripts vc, c, and nc denote very critical, critical and noncritical layers, respectively.

For 45nm lithography cost, we study three scenarios.

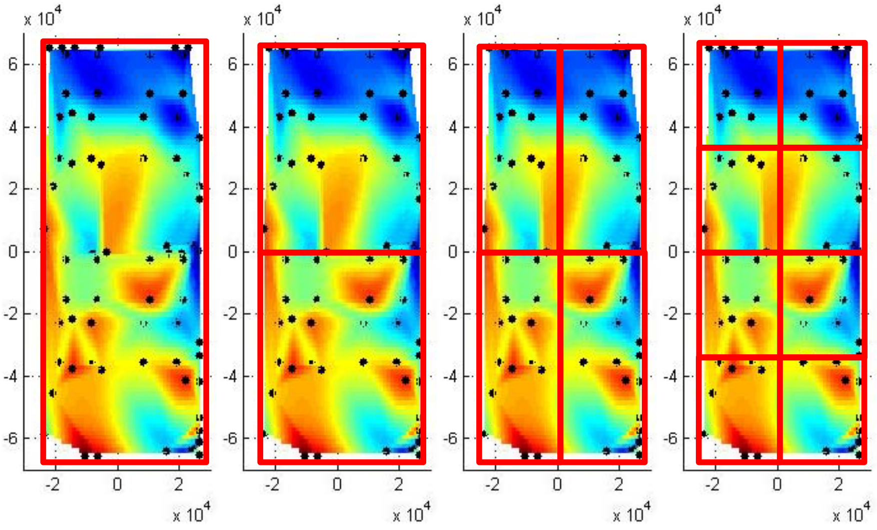
- *Scenario 1: constant lithography cost.* Cost of an exposure for very critical ($cost_{e,vc}$), critical ($cost_{e,c}$) and non-critical ($cost_{e,nc}$) layers is assumed as $2.5, $1.5 and $0.5, respectively, based on 90nm lithography cost estimation [147].

- *Scenario 2: scaling by the lithography tool cost ratio.* Lithography tool cost is assumed as $40M, $49M, and $52M, for 45$nm$, 32$nm$ and 22nm technologies [148]. From curve-fitting of lithography tool cost with respect to technology generation, the 90nm tool cost is estimated as $29M. Then, scaling 90$nm$ exposure cost by 1.38 (= $40M / $29M) gives $3.45, $2.07 and $0.69 as the 45$nm$ exposure cost for critical, critical and non-critical layers, respectively.

- *Scenario 3: doubling at every technology generation.* We also study a pessimistic lithography cost scenario to see the impact of high lithography cost on mask strategy. 45nm exposure cost for very critical, critical and non-critical layers is assumed as \$13.79, \$8.28, and \$2.76, respectively.

5.1.3 Parametric Yield Cost

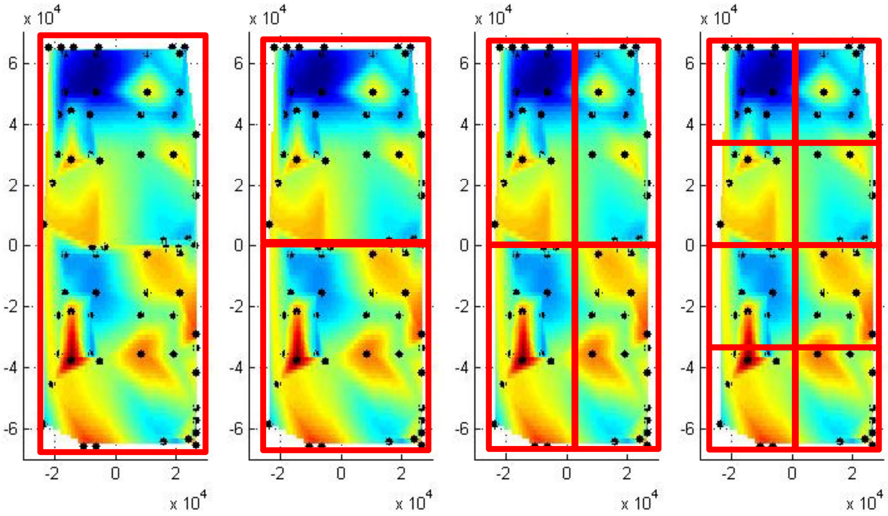
Mask size affects not only mask yield, but also the parametric yield of the manufactured dies. We analyze how CD variation changes with respect to mask size. Figures 5.2(a) and 5.3(a) respectively show mask CD variation maps for 90nm and 65nm industry products. The original mask size is approximately 52mm×132mm for both masks.



(a) 52mm x 132mm (b) 52mm x 66mm (c) 26mm x 66mm (d) 26mm x 33mm

Figure 5.2: Mask CD variation map for a 90nm product.

From the given CD measurement data, we analyze CD variations while decreasing field size, as shown in parts (b), (c), and (d) of Figures 5.2 and 5.3. Figures 5.4 and 5.5 respectively show 3σ CD variations with respect to field size for the 90nm and 65nm masks. As field size decreases from 52mm×132mm to 26mm×33mm, the average of 3σ CD variations of the small subfields is reduced from 2.04nm (resp. 2.21nm) to 1.37nm (resp. 1.77nm) for the 90nm (resp. 65nm) mask. Furthermore, if we are allowed to choose the subfield with minimum CD variation out of all subfields, 3σ CD



(a) 52mm x 132mm (b) 52mm x 66mm (c) 26mm x 66mm (d) 26mm x 33mm

Figure 5.3: Mask CD variation map for a $65nm$ product.

variation can be reduced to $1.10nm$ (resp. $1.08nm$) for $90nm$ (resp. $65nm$) mask, as shown in the red-dotted traces in Figures 5.4 and 5.5.

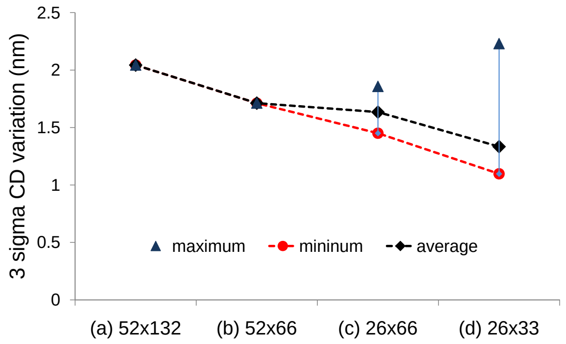


Figure 5.4: 3σ CD variation (nm) versus field size (mm^2) for the $90nm$ mask CD map in Figure 5.2.

Reduced variation in mask CD from a small-field strategy would contribute to reduced variation in electrical characteristics on the manufactured wafer. Although we did not have access to a unified CDU data set for both mask and wafer of a single design, we have been able to analyze variations of (on-wafer) electrical characteristics in two industry data sets with respect to the mask size.

The first data set consists of measured ring oscillator delay in a $65nm$ test chip from Foundry A. There are 14 measurement points regularly placed in a $20 \times 20 mm^2$

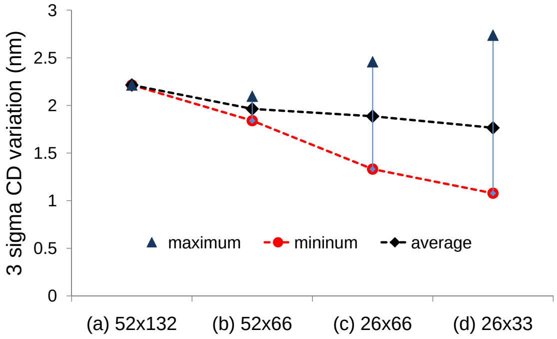


Figure 5.5: 3σ CD variation (nm) versus field size (mm^2) for the $65nm$ mask CD map in Figure 5.3.

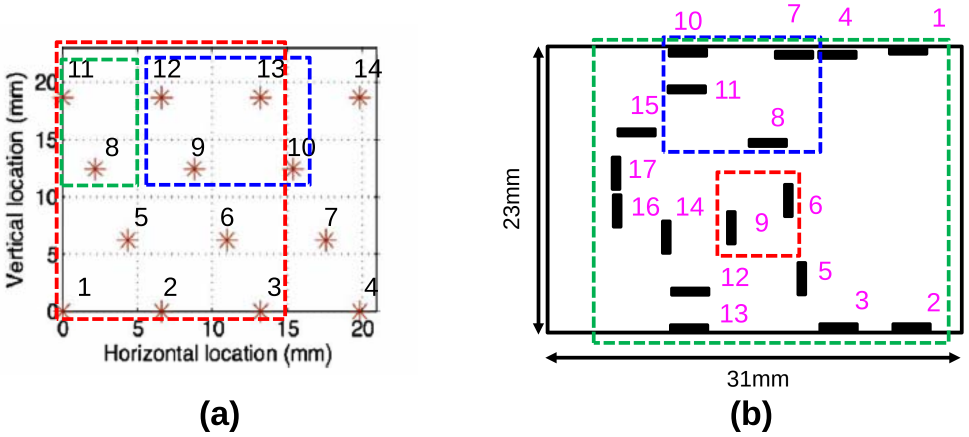


Figure 5.6: Measurement point locations in fields from Foundry A data in (a) and Foundry B data in (b).

field as shown in Figure 5.6(a). The number of measured fields is 36,727. From this data, we calculate delay variation (σ/μ) while changing the size of the sampling window to account for the impact of small field, as shown by the dotted boxes in Figure 5.6(a). The size and location of a sampling window together determine the measurement points included. Table 5.3 summarizes σ/μ with respect to the sampling window height and width. Given the yield of the full-size field, we normalize the delay variations of different field sizes to that of the full-size field, and calculate corresponding parametric yields. For instance, the number of standard deviations resulting in 90% yield is 1.645. The delay variation of the $400mm^2$ full-size field of Foundry A is 2.995 as shown in Table 5.3. The 3.126 delay variation from the $266.66mm^2$ field is equivalent to 1.576 (= 1.645×(2.995/3.126)) standard deviations, which gives 88.5% yield. Column 4 (resp. 5) of Table 5.3 shows the parametric yield, assuming that the parametric yield of a full-size field is 90% (resp. 80%). The parametric yield improves as window size decreases.

Table 5.3: Delay variation and parametric yield with respect to field size in $65nm$ test chip from Foundry A.

Width	Height	Area	Delay variation σ/μ	$Y_{p,90}$	$Y_{p,80}$
(μm)	(μm)	(mm^2)	(%)	(%)	(%)
20000	20000	400.00	2.995	90.0	80.0
13333	20000	266.66	3.126	88.5	78.0
8888	6500	57.77	2.749	92.7	83.7
2222	6500	14.44	1.897	99.1	95.7

The second data set consists of measured $I_{d,sat}$ in a $45nm$ test chip from Foundry B. There are 17 measurement points in a $23{\times}31mm^2$ field, as shown in Figure 5.6(b). We again calculate $I_{d,sat}$ variation while changing sampling window size. Table 5.4 summarizes σ/μ of $I_{d,sat}$ variation with respect to the sampling window height and width. We again assume that the yield of a full-size field is 90% (resp. 80%), then normalize delay variation of different field sizes to that of the full-size field and calculate parametric yields.

Figure 5.7 shows the relationship between parametric yield and field area for both data sets. From linear regression, we obtain a parametric yield model with respect to the normalized field area f_{area}. The obtained parametric yield model is reflected in the final cost model as a denominator in the lithography cost, assuming that more wafers will be processed as parametric yield decreases. The linear parametric yield model is

$$Y_p(f_{area}) = (1 - \alpha f_{area})$$

where α is 0.1296 (resp. 0.2657) when the yield of full-size mask is assumed to be 90% (resp. 80%).

Table 5.4: $I_{d,sat}$ variation and parametric yield with respect to field size in $45nm$ test chip from Foundry B.

Width (μm)	Height (μm)	Area (mm^2)	$I_{d,sat}$ variation σ/μ (%)	$Y_{p,90}$ (%)	$Y_{p,80}$ (%)
22941	20418	468.42	3.209	90.0	80.0
16088	12937	208.13	2.945	92.7	83.7
6881	9239	63.57	2.421	97.1	91.0
7548	8312	62.74	1.687	99.8	98.5
6222	4855	30.20	2.266	98.0	93.0
5999	4385	26.31	2.368	97.4	91.7
5617	3698	20.77	1.501	100.0	99.4
3640	2994	10.90	2.153	98.6	94.4
2360	4385	10.35	1.085	100.0	100.0
3172	75	0.24	1.042	100.0	100.0

5.1.4 Overall Manufacturing Cost Comparison

Finally, the overall manufacturing cost considering parametric yield is calculated as

$$Cost_{all} = n_{regen} \cdot Cost_{maskset} + Cost_{litho}/Y_p$$

where n_{regen} is the number of mask regenerations considering mask wearout.[1]

Figure 5.8 shows overall manufacturing cost with respect to varying numbers of dies per field, as the number of wafers processed is increased. This comparison assumes 90% parametric yield for a full-size field. All values are normalized to the cost of processing 10 wafers with a $100\times100mm^2$ field.

For Scenario 1 shown in Figure 5.8(a), we can observe that below 100 wafers, fewer dies per field can have lower cost than the full-size field (i.e., 9 dies per field) case: up to 20 wafers, 2 dies per field has best cost; between 20 and 40 wafers, 4 dies per field has best cost; and between 50 wafers and 100 wafers, 6 dies per field has best

[1] We assume that the mask set must be regenerated every 86,000 exposures, which is the number of exposures for 1,000 $300mm$ wafers with a $25\times25mm^2$ full-size field. With a small field, n_{regen} increases due to the increase in the number of exposures per wafer.

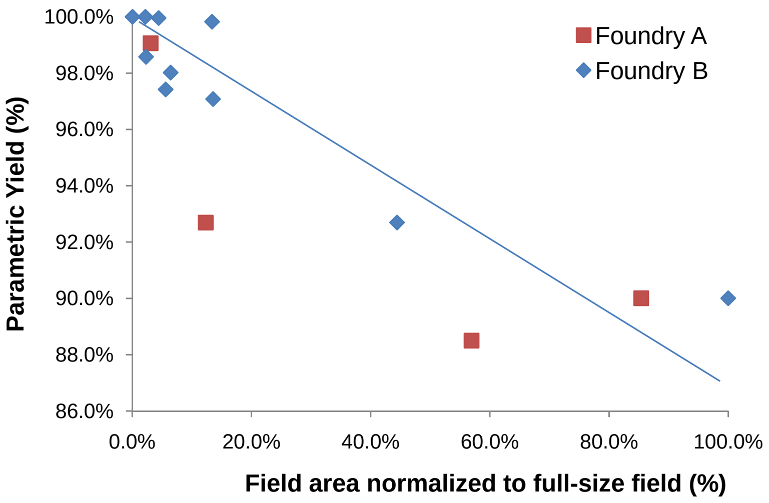
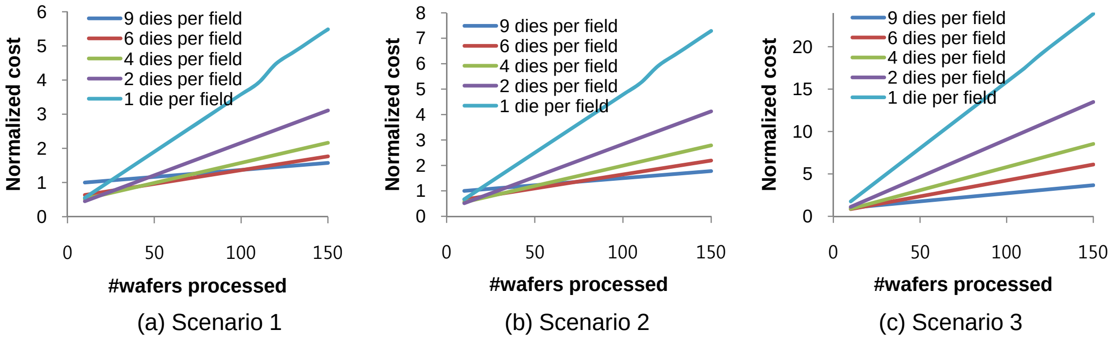


Figure 5.7: Field size normalized to a full-size field in x-axis, versus parametric yield assuming 90% yield for full-size field in y-axis.



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Figure 5.8: Overall manufacturing cost in y-axis versus the number of wafers processed. Cost values are normalized to the cost of processing 10 wafers with a $100 \times 100mm^2$ field.

cost. The benefit of small-size field (i.e., SLR) is reduced as lithography cost increases. This is seen in Figures 5.8(b)-(c): for Scenario 2, the full-size field has best cost when more than 70 wafers are processed; and for Scenario 3, the full-size field has best cost when more than 10 wafers are processed.

5.1.5 Defect-Aware Parametric Yield for EUVL

In this subsection, we compare parametric yield due to EUVL mask defects for various reticle strategies. To calculate defect-aware parametric yield, we first randomly distribute defects on a mask blank. At the same time, we extract timing-critical regions from a design using signoff timing analysis and placement information. We then check whether any defect in a mask blank overlaps with any timing-critical region. The overlapping of defects and timing-critical regions varies with the reticle strategy and the location of the field on a mask blank. We estimate the yield from Monte Carlo simulation.

Defect density and distribution. Burns et al. [44] assume 10 to 55 defects per mask; Heuvel et al. [90] use a mask with 0.72 known defects/cm^2 in their experiments, and find around 200 defects from inspection. Early EUVL mask blanks contain thousands of defects. With steady improvements in blank generation, the detectable defect count, with first-generation mask-blank inspection tools limited to detecting 80nm defect size, was reduced to hundreds in 2007. However, the number of defects increases again by more than an order of magnitude when detectable defect size is reduced from 80nm to 50nm by advances in inspection technology [44]. Among the detectable defects, defects that change feature size by more than 10$\%$ are regarded as critical defects in the ITRS [11]. Burns et al. [44] assume defect sizes of 146nm to 3,690nm in their defect-avoiding mask alignments, while the ITRS specifies that the critical defect size for EUVL masks is 41nm in 2009 and reduces to 16nm in 2024 [11].

We focus on substrate defects, which are the majority (e.g., 75$\%$ in [149]) of EUVL mask defects. These substrate defects are randomly placed in a typical 150$mm \times$ 150mm mask blank. The used testcase has 8$mm \times$8mm area, and we assume that 16 (4×4) dies can be fit into the full-size reticle. Assumed defect densities are summarized in Table 5.5. Up to 2.222 defects/cm^2 in a mask blank in Table 5.5 may be realistic, but we also examine much larger defect densities to account for future inspection technology improvements and/or early stages of technology introduction.

For a given defect density, we distribute the defects in two ways.

- *Uniform random.* The number of defects per mask blank is calculated from the given defect density, and defect location coordinates are determined by uniformly random number generation between 0 and mask blank size in x and y respectively.
- *Decentered Gaussian.* The number of defects per mask blank is calculated from the given defect density, and defect locations are sampled from a decentered Gaus-

Table 5.5: Assumed defect densities.

Field size (1×) ($cm \times cm$)	#Defects per mask blank	Mask blank size (4×) ($cm \times cm$)	Defect density (/cm^2) in a mask blank (4×)
3.2 × 3.2	10	15.0 × 15.0	0.044
3.2 × 3.2	50	15.0 × 15.0	0.222
3.2 × 3.2	100	15.0 × 15.0	0.444
3.2 × 3.2	500	15.0 × 15.0	2.222
3.2 × 3.2	1000	15.0 × 15.0	4.444
3.2 × 3.2	5000	15.0 × 15.0	22.222

sian distribution. The decentered Gaussian distribution is composed of two Gaussian distributions: for x- (y-)coordinates, one mean is located at the left (bottom) boundary of the mask blank, and the other mean at the right (top) boundary of the mask blank. We take one-sixth of the mask blank width (height) as the sigma of the Gaussian distribution.

Defect and impact on circuit timing. Clifford et al. [59] show that square defects at the substrate with widths varying from 60nm to 90nm all result in around 50-60nm defect widths at the final multi-layer (ML) EUVL mask surface, and that defect heights can vary from 1.5nm to 5.5nm. CD on wafer varies mainly with the defect height at the top of the ML surface; Clifford et al. also propose a simple linear equation to calculate CD variation (ΔL) from the surface defect height as

$$\Delta L = \frac{\sqrt{I_{NoDefect}}\,(m_{Defect} \cdot h_{SurfaceDefect} + b_{Defect})}{ImageSlope} \tag{5.2}$$

where $h_{SurfaceDefect}$ is the defect height at the top of the ML surface, $I_{NoDefect}$ is the image intensity without defects, $ImageSlope$ is the slope of the aerial image, and m_{Defect} and b_{Defect} are fitting parameters. According to the equation, different volume sizes of defects change heights of surface defects, and thus result in different CD and circuit timing. Table 5.6 summarizes the defect heights assumed in this work, and their respective impacts on CD and timing. To calculate ΔL, we use Equation (5.2) with the same parameters used by Clifford et al. [59]. To quantify the impact of ΔL on timing, we measure delay variation (ΔT) from the nominal worst-case delay of a most frequently used cell (i.e., 2-input NAND gate) in the used testcase with respect to transistor gate length variation, using a $45nm$ open-source design kit [14].

Table 5.6: Surface defect height, CD variation (ΔL), and resulting timing variation (ΔT) from a $45nm$ open-source design kit [14].

Height (nm)	ΔL (nm)	ΔT (ps)
1	1.03	2.00
2	3.06	5.87
4	7.11	13.41
8	15.22	28.27

From the delay variation due to defects, we can estimate parametric yield. When a defect is located on a timing-critical cell whose slack is less than ΔT of the defect, the die will fail due to timing errors and is counted as a yield loss.[2]

Reticle strategies. We consider various reticle strategies as illustrated in Figure 5.9.

- Case 1: *SLR-L*
- Case 2-A: *MLR* with defects on every layer (i.e., region) having the same impact on timing
- Case 2-B: *MLR* with defects only on critical layers (e.g., poly) affecting timing
- Case 3-A: *SLR-S* with mask location selected randomly in a 2-D lattice of available locations in a mask blank
- Case 3-B: *SLR-S* with mask location selected as the lowest defect-density region in a 2-D lattice of available locations in a mask blank
- Case 4: *SLR-S* with mask generated at the lowest defect-density region with no restriction in the location[3]

[2] We ignore the fact that multiple defects on a timing path (i.e., the sum of timing variations from multiple defects) can cause a timing failure even though none of the defects individually causes a timing failure.

[3] Case 3-B maximizes the number of available masks per mask blank (e.g., 9 fields), but Case 4 may not.

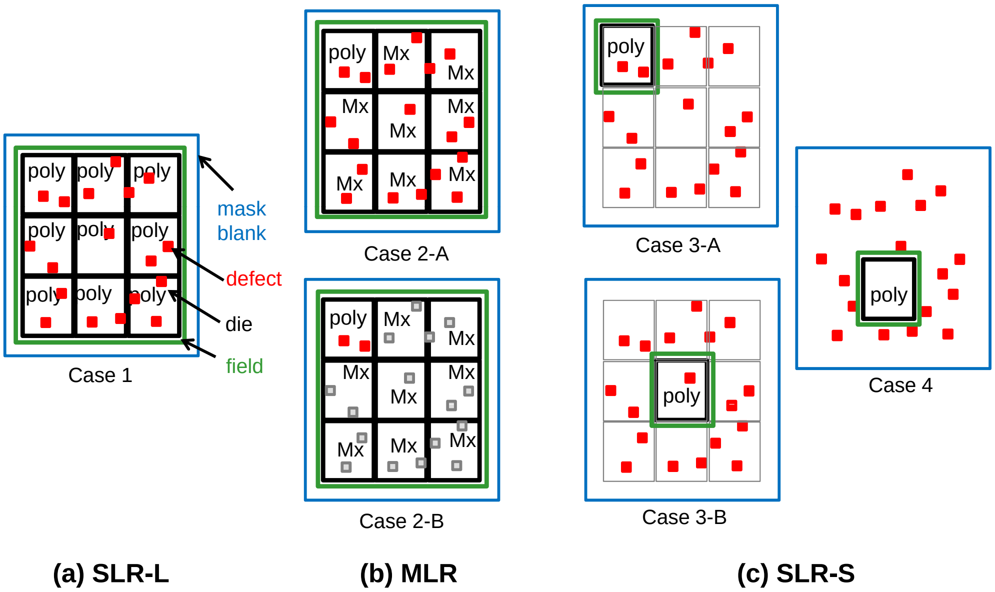


Figure 5.9: Reticle strategies: (a) SLR-L, (b) MLR with same weight for all layers (top) and different weights for different layers (bottom), (c) SLR-S with random location (top-left) and lowest defect location in a gridded mask blank (bottom-left), and with optimal location (right).

Intuitively, one can expect that Case 1 and Case 2-A should have the same yield when all layers have the same sensitivity to defects, since overall yield is a cumulative yield of all layers for both cases. Case 1 and Case 2-B should have the same yield when only one critical layer (e.g., poly) is sensitive to defects. In both cases, yield only depends on the yield of the critical layer. Additionally, Case 2-B and Case 3-A should have the same yield, since both cases use the same region in the mask blank. Case 4 will clearly have better yield than Case 3-B, since Case 4 has no constraints for the location of the mask. Case 3-B will have better yield than Case 3-A, since Case 3-B can use the region with lowest defect density out of nine available regions in a mask blank. Hence, assuming that only defects on critical layers have impact, there are four distinct cases: Case 1, Case 2-B, Case 3-B and Case 4.

Yield calculation. We calculate timing-critical regions in a design from a signoff static timing analysis. We find a list of timing-critical cells whose timing slack is less than the timing variation due to defects (ΔT), and obtain a list of bounding boxes of timing-critical cells from placement information (e.g., Design Exchange Format (DEF) [4]). Using the timing-critical regions in a die and randomly-placed defect regions in a mask blank, we check whether any defect region overlaps with any timing-critical regions of dies in a field. If there is an overlap, the die is regarded as failed. This geometric manipulation saves simulation time that would otherwise be required to perform actual timing analysis with defect-induced linewidth variation.

We note that Case 4 shows zero yield loss with the reasonable defect densities that we assumed. Although Case 4 can have a yield loss with very high defect densities, the runtime for the overlap checking increases excessively. Hence, for Case 4, we calculate a lower bound for defect density which incurs a yield loss, instead of performing the overlap checking.

To calculate a lower bound of defect density, we define the following sets of regions.

- S_C: set of timing-critical regions in a field, i.e., a list of bounding boxes of timing-critical cells that would result in parametric failure if intersected with a defect location

- S_F: set of forbidden regions in a mask blank where mask origin should not be located, to avoid overlap of defect regions with S_C

- S_P: set of feasible regions in a mask blank where mask origin can be located with no overlaps between S_F and S_C[4]

Figure 5.11 illustrates a simple example of the forbidden region calculation for a single point defect and a timing-critical cell. When a defect p is located at (p_x, p_y) in a mask blank, and there is one timing-critical region r at (r_x, r_y) with width of r_w and height of r_h, the mask origin should not be placed in the red region defined by the lower-left corner at ($p_x - r_x - r_w$, $p_y - r_y - r_h$) and the upper-right corner at ($p_x - r_x$, $p_y - r_y$) as shown in Figure 5.11. If the mask origin is placed in the red region, timing-critical region r must be overlapped by the defect.[5] Figure 5.10 shows the procedure to calculate

[4] S_P is calculated by subtracting S_F from the bounding box of the entire mask blank

[5] For defects with nonzero area, the calculation method is similar, with the dimensions of a forbidden region expanded by the width and height of defects.

S_F for a single defect. Each defect defines $|S_C|$ rectangular regions in S_F, and we iterate the procedure for all defects in the mask blank to obtain S_F.

Algorithm: FORBIDDEN_REGION
Input: defect p at (p_x, p_y)
Output: forbidden region $S_F(p)$

```
$S_F(p) \leftarrow Ø$
for each timing-critical region $r \in S_C$
    calculate a defect region $f$ ($x_1$, $y_1$, $x_2$, $y_2$) by
        $x_1 \leftarrow p_x$ - ($r_x$ + $r_w$)
        $y_1 \leftarrow p_y$ - ($r_y$ + $r_h$)
        $x_2 \leftarrow p_x$ - $r_x$
        $x_2 \leftarrow p_y$ - $r_y$
    $S_F(p) \leftarrow S_F(p) \cup f$
end
```

Figure 5.10: Procedure to calculate forbidden regions due to a single defect.

With a pessimistic assumption that no forbidden regions due to different defects intersect each other, the area of S_F, which is the union of all forbidden regions, is simply calculated as the area of S_C multiplied by the number of defects. As the number of defects increases, the area of S_F increases and the area of S_P decreases. When the area of S_F is equal to the area of the mask blank, the area of S_P reaches zero and Case 4 must have a yield loss regardless of the choice of the mask location. Hence, a lower bound on the number of defects needed to cause yield loss is calculated as $Area(S_{P0})/Area(S_C)$ where S_{P0} is the area of feasible region without defects. S_{P0} is calculated as $(width_{field} - width_{die}) \times (height_{field} - height_{die})$. If the number of defects does not exceed the lower bound, then there must exist a nonempty subset of the feasible region within which a die can be located, and hence Case 4 would have 100% yield.

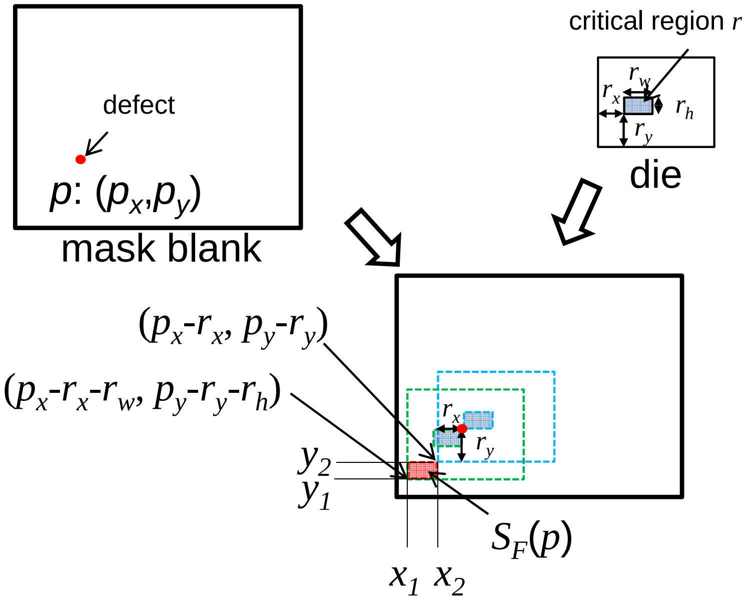


Figure 5.11: An example of forbidden region calculation for a single defect.

5.1.6 EUVL Parametric Yield Comparison

We calculate parametric yield due to EUVL defects for a given number of mask sets, i.e., 1,000 sets. We furthermore evaluate the parametric yield sensitivity to defect parameters, such as defect density d, defect height h, defect influence distance r, and defect distribution method m.

Defect density versus parametric yield. The first experiment compares the parametric yield changes due to defect density. For this experiment, other parameters are fixed in reasonable ranges. Defect height is assumed as 4nm and defect influence distance is assumed as 30nm in wafer (120nm in reticle), which is 2× *full width at half maximum* (FWHM) of typical surface defects reported by Clifford et al. [59]. Defects are assumed to have a uniform random distribution. Figure 5.12 compares parametric yields of various reticle strategies. Case 2-B has the worst yield, since it assumes that a possible problematic mask for a critical layer in MLR is used for all dies. Case 1 has better yield than Case 2-B, but still has lower yield than other two cases, since several of the dies in a field can be affected by defects. Case 4 shows perfect yield, since there is large flexibility to place a critical layer on a mask blank avoiding defects,[6] and Case 3 shows the

[6] $Area(S_C)$ of the testcase is 22,443.4μm^2 with 250nm defect influence distance in wafer (1000nm

second best yield. While the yield trends are clear, we note that the differences between cases are not significant in the range of reasonable defect densities.

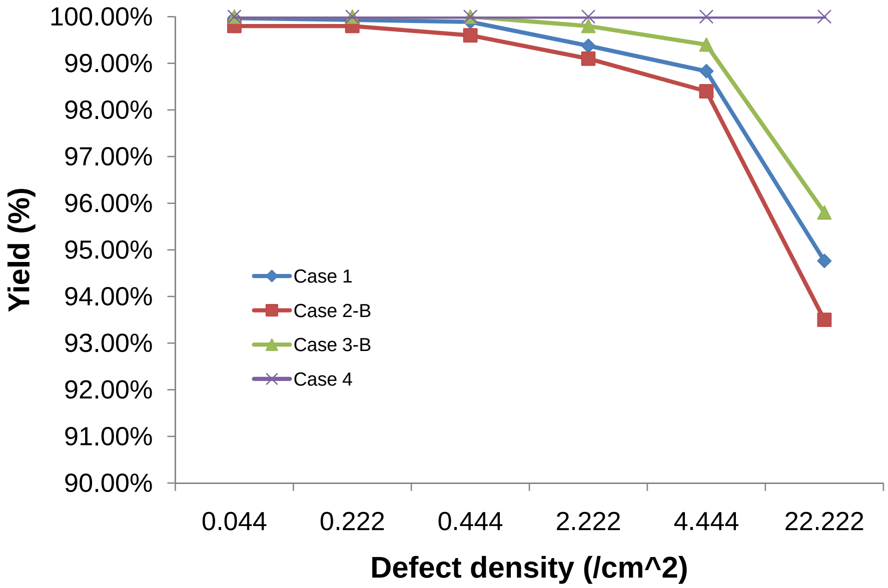


Figure 5.12: Defect density versus yield for various reticle strategies. $4nm$ defect height and $120nm$ (4×) defect influence distance are assumed, with defects uniformly distributed.

Defect height versus parametric yield. The second experiment assesses parametric yield changes due to defect height. Since defect height determines the CD variation, it will affect timing and hence the timing-critical area in a design (i.e., S_C). For this experiment, defect density is fixed in the range of 0.444-2.222 defects/cm^2, defect influence distance is assumed as $30nm$, and defects are assumed to have a uniform random distribution. Figure 5.13 compares parametric yields of various reticle strategies. We observe that parametric yield is not significantly changed due to the defect height. The reason is that the timing-critical region is relatively small compared to the entire field area, and this swamps even the assumption of a pessimistic defect height, e.g., $8nm$.

Defect influence distance versus parametric yield. The third experiment assesses parametric yield impact of the defect influence distance. We examine zero influence defect distance (i.e., point defect), a reasonable influence distance (i.e., 2× FWHM of

in reticle), and $Area(S_{P0})$ is $576mm^2$ (= $(32mm$ - $8mm)^2$). The lower bound of the number of defects $Area(S_{P0})/Area(S_C)$ is 25,665. As long as the number of defects is less than 25,665, Case 4 has 100% yield.

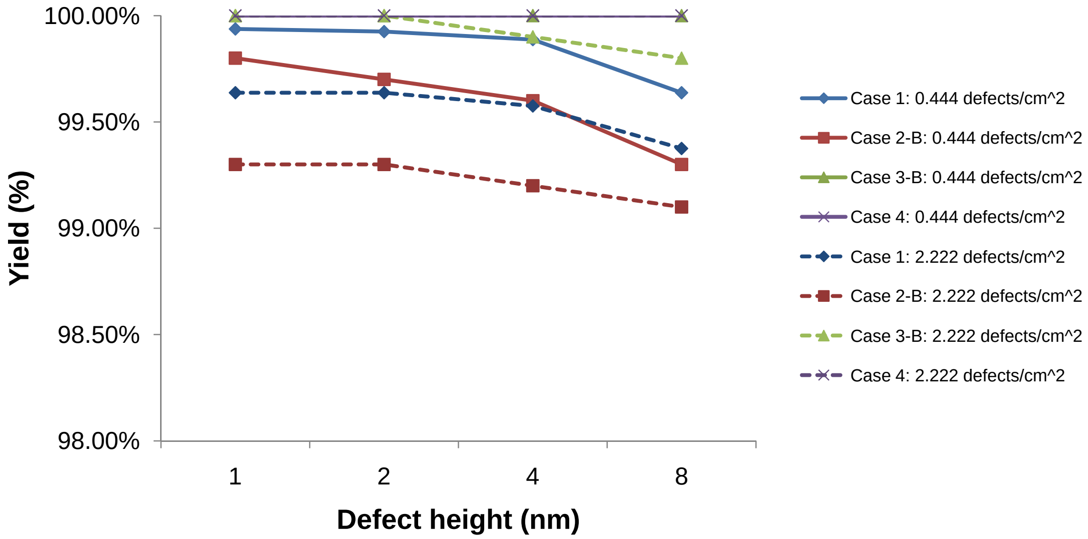


Figure 5.13: Defect height versus yield for various reticle strategies. 0.444-2.222 defects/cm^2 defects are uniformly distributed and defect influence distance is assumed as 120nm (4×).

typical surface defects), and a very large influence distance (i.e., 1,000nm in reticle (250nm in wafer)), with 4nm defect height and uniform defect distribution.[7] Figure 5.14 compares parametric yields for various reticle strategies. We see that the yield sensitivity to defect influence distance is negligibly small. For 0 and 120nm distance, there is almost no difference. With a larger defect influence of 1,000nm in reticle, yield is reduced, but the yield loss is still insignificant. This again may be attributed to the relatively small timing-critical region in a design.

Defect distribution versus parametric yield. Finally, Figure 5.15 assesses the yield difference between uniform and decentered-Gaussian defect distributions. Case 4 still shows perfect yield. In addition, Case 3-B with decentered Gaussian distribution also shows perfect yield, since the dies near the center of 16 (= 4 × 4) possible locations have low defect probability due to the construction of the decentered Gaussian distribution. However, the worst case of Case 2-B, where field location is chosen along the boundary of the mask blank, shows a sharp yield loss. Except for Case 2-B, yield with the decentered Gaussian defect distribution is higher than yield with the uniform defect

[7] Although typical mask defect size is as small as $< 100nm$, the map of defect locations produced by the inspection process may not be accurate (e.g., around 500nm resolution in x- and y-coordinates, respectively). Hence, the case of 1,000nm defect influence distance may not be overly pessimistic.

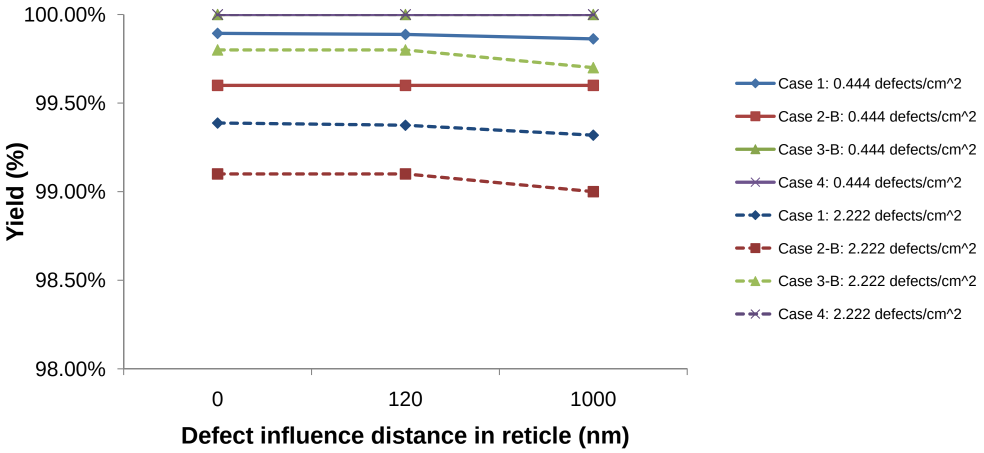


Figure 5.14: Defect influence distance versus yield for various reticle strategies. 0.444-2.222 defects/cm^2 defects with 4nm height are uniformly distributed.

distribution.

Significance of EUVL defectivity. From the experiments, the major observations are summarized as follows.

- As defect density increases, parametric yield decreases.
- As defect height increases, parametric yield decreases.
- As defect influence distance increases, parametric yield decreases.
- A decentered Gaussian random distribution of defect locations reduces the parametric yield loss. In particular, when we are looking for a best location for a critical layer to be placed in a mask blank, the decentered Gaussian assumption gives lower defect density near the center of the mask blank.

These observations are fairly intuitive, and they support the notion that defects should be accurately identified and cleaned up as much as possible to mitigate potential defect-induced parametric yield loss. Interestingly, however, experimental results indicate that the parametric yield loss due to mask blank defects may not be as significant as has been recently thought by most EUVL researchers. The main reason is that in typical designs the timing-critical region that can be affected by mask blank defects is quite small relative to the entire design area. Table 5.7 shows the relative size of the timing-critical region of several real designs implemented in 65nm and 45nm technologies. (The testcase used for yield calculation is based on an MPEG2 core in Row 3 of Table

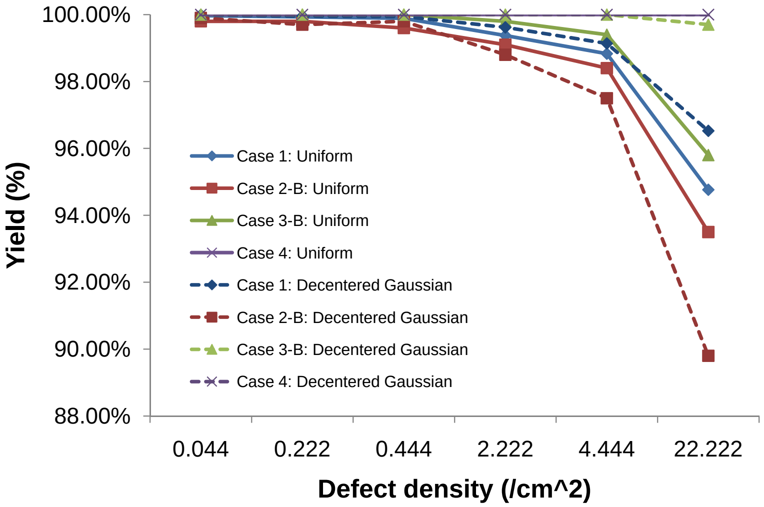


Figure 5.15: Defect distribution methods versus yield for various reticle strategies. Defects with $4nm$ height and $120nm$ ($4\times$) influence distance are distributed.

5.7.) Hence, as long as the relative size of the timing-critical region does not increase significantly, mask blank defectivity may not be the most critical issue for near-term EUVL adoption, and more attention and investment can be directed to other technical hurdles for EUVL.

Table 5.7: Proportion of timing-critical regions in real designs. Area of timing-critical region is calculated as the sum of areas of cells for which timing slack is less than $20ps$.

$65nm$		$45nm$	
Design	Timing-critical area (%)	Design	Timing-critical area (%)
MPEG2	1.077	*AES45*	2.068
AES65	1.746	*JPEG45*	0.187
JPEG65	0.442		

5.2 Timing Yield-Aware Dose Map Optimization

Critical dimension (CD) variation is a dominant factor in the variation of delay and leakage current of transistor gates in integrated circuits. With advanced manufacturing processes, CD variation is worsening due to a variety of systematic variation sources at both within-die and reticle- or wafer-scale; the latter sources include radial bias of spin-on photoresist thickness, etcher bias, reticle bending, uniformity of wafer starting materials, etc [100]. A statistical leakage minimization method is proposed in [40], which obtains significant improvement in total leakage reduction by simultaneously varying the threshold voltage, gate sizes and gate lengths. Gupta et al. [85] proposed to apply gate-length (CD) biasing only on the devices in non-critical paths for leakage power control without negative effects on timing.

A recent technology from ASML, called *Dose Mapper* [196, 94], allows for minimization of ACLV (Across-Chip Linewidth Variation) and AWLV (Across-Wafer Linewidth Variation)[8] using an exposure dose (or, simply, dose) correction scheme. Dose Mapper in the ASML tool parlance exercises two degrees of control, *Unicom* and *Dosicom* [145], which respectively change dose profiles along the lens slit and the scan directions of the step-and-scan exposure tool.

Today, the Dose Mapper technique is used solely (albeit very effectively – e.g., [157]) to reduce ACLV or AWLV metrics for a given integrated circuit during the manufacturing process. However, to achieve optimum device performance (e.g., clock frequency) or parametric yield (e.g., total chip leakage power), not all transistor gate CD values should necessarily be the same. For devices on setup timing-critical paths in a given design, a larger than nominal dose on the poly layer (causing a smaller than nominal gate CD) will be desirable, since this creates a faster-switching transistor. On the other hand, for devices that are on hold timing-critical paths, or in general that are not setup-critical, a smaller than nominal dose on the poly layer (causing a larger than nominal gate CD) will be desirable, since this creates a less leaky (although slower-switching) transistor. What has been missing, up to now, is any connection of such "design awareness" – that is, the knowledge of which transistors in the integrated-circuit product are setup or hold timing-critical – with the calculation of the Dose Mapper solution.[9] The Zeiss/Pixer Critical Dimension Control (CDC) technology [29] also enables adaptivity in the manufacturing flow to meet the required CD specifications. The CDC technology

[8] ACLV is primarily caused by the mask and scanner, while AWLV is affected by the track and etcher [159].

[9] Optimization of gate CDs according to setup or hold timing (non-)criticality has been used by [85]. What we propose below uses a coarser knob (i.e., the dose map) for design-aware gate CD control, but has the advantage of not requiring any change to the mask or OPC flows.

modifies the local mask transmissivity (which translates into local CD changes on the wafer during the lithography process) without removing the pellicle, thus allowing for tool installations either at the mask manufacturing site, or at the fab line. In this section, we focus on the Dose Mapper technology for tuning of transistor gate dimensions.

5.2.1 Preliminaries of Dose Map Optimization

Dose mapper fundamentals. Figure 5.16 shows the intrafield Dose Mapper concept. In Figure 5.16, the slit exposure correction is performed by Unicom. The actuator is a variable-profile gray filter inserted in the light path. The default filter has a second-order (quadratic) profile, and ASML [1] recommends use of a quadratic slit profile to model data in the slit direction. It is also possible to obtain a customized profile: lithography systems with Unicom (e.g., the ASML XT:1700i machine) support a slit profile represented by polynomials of up to the 6^{th} order in the dose recipe. Overall, a correction range of $\pm 5\%$ can be obtained with Unicom for the full field size of $26mm$ in the X-direction.

Scan exposure correction is realized by means of Dosicom, which changes the dose profile along the scan direction. The dose generally varies only gradually during scanning, but the dose profile can contain higher-order corrections depending on the exposure settings. The dose set, $D_{set}(y)$, is used to model parameters for a dose recipe formed of Legendre polynomials (Legendre functions of the first kind) as

$$D_{set}(y) = \sum_{n=1}^{8} L_n P_n(y) \tag{5.3}$$

where y is a floating variable ($|y| \leq 1$) related to the scan position, L_n are Legendre coefficients, and $P_n(y)$ are Legendre polynomials of variable y. Up to eight Legendre coefficients can be supported. The correction range for the scan direction is $\pm 5\%$ (10% full range) from the nominal energy of the laser. When the requested x-slit and y-scan profiles are sent to the lithography system, they are converted to system actuator settings (one Unicom shift for all fields, and a dose offset and pulse energy profile per field).

Dose sensitivity is the relation between dose and critical dimension, measured as CD [nm] per percentage [%] change in dose. Increasing dose decreases CD as shown in

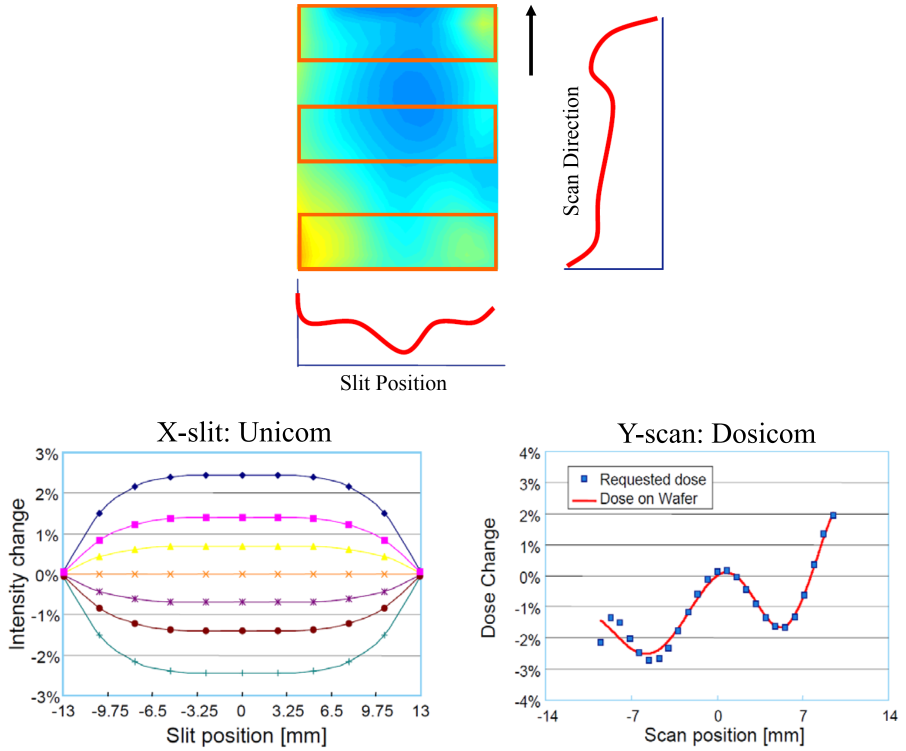


Figure 5.16: Unicom and Dosicom, which respectively change dose profiles in slit and scan directions. Figure reproduced from [2].

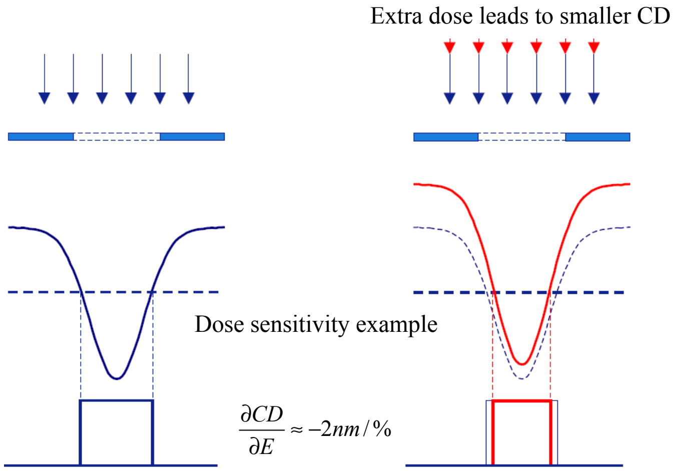

Extra dose leads to smaller CD
Dose sensitivity example
$\frac{\partial CD}{\partial E} \approx -2nm/\%$

Figure 5.17: Dose sensitivity: increasing dose (red color) decreases the CD [1].

Figure 5.17, i.e., the dose sensitivity has negative value. To calculate the dose sensitivity ($\triangle CD/\triangle E$, [nm/%]), a Focus-Exposure Matrix (FEM) must be exposed on a product wafer for each product layer using standard production settings for reticle (e.g., 6% attPSM), resist and illumination.

Dose map optimization problem. The design-aware dose map problem, for the objective of timing yield and leakage power, can be stated as follows. *Given placement P with timing analysis results, determine the dose map to improve timing yield as well as reduce total device leakage.* We have studied two dose map optimization problems with different objectives: the first seeks to minimize total leakage power under a clock period upper bound constraint, and the second seeks to minimize clock period under a leakage power upper bound constraint. These two optimizations are respectively formulated as a quadratic program (QP) and as a quadratically constrained program (QCP), and are solvable using efficient commercial solvers [10].

In the following, for simplicity of exposition we assume that the reticle area taken up by a single copy of the integrated circuit is the same as the area of the exposure field. In practice, the exposure field will contain one or more copies of the integrated circuit being manufactured. It is simple to extend the proposed algorithms to the case where the exposure field contains multiple copies of the integrated circuit being manufactured: smoothness or gradient constraints are scaled, and multiple copies of the dose map solution are tiled horizontally and vertically.

For the dose map optimization problem, we partition the exposure field into a set of rectangular grids $R = |r_{i,j}|_{M\times N}$ on both active and poly layers, where the (uniform) width and height of rectangular grid $r_{i,j}$ are both less than or equal to a user-specified parameter G. G controls the granularity of the dissected rectangular grids: a smaller value of G corresponds to a larger number of rectangular grids, along with a more precisely specified new dose map and better timing yield and/or leakage power improvement. However, G cannot be set too small, due to Dose Mapper equipment limitations. In general, G can be determined so as to balance between Dose Mapper equipment constraints and timing yield and/or leakage power improvement. While different values of G may be used for different layers, we assume in the following that the same G values are used for both active and poly layers. Dose map optimization using different granularities of the partitioned rectangular grids is tested and discussed in Section 5.2.4.

Our discussion will focus on dose map optimization for the poly layer, i.e., for modulation of gate length. We have also tried dose map optimization on both the active and poly layers to simultaneously modulate gate width and length when optimizing timing and leakage power. We now state circuit delay and leakage power estimation equations, as well as our problem formulations, considering both gate width and gate length variations.

Circuit delay and leakage power calculation. We assume that dose sensitivity D_s has the typical value of -2nm/% [157]. Gate length and gate width change linearly with dose tuning, i.e., $\Delta L_p = D_s \times d^P_{i,j}(p)$ and $\Delta W_p = D_s \times d^A_{i,j}(p)$, where ΔL_p is the change in gate length of gate p, ΔW_p is the change in gate width of gate p, and $d^P_{i,j}(p)$ and $d^A_{i,j}(p)$ are percentage values which specify the relative changes of dose for poly and active layers in the rectangular grid $r_{i,j}$ wherein gate p is located.

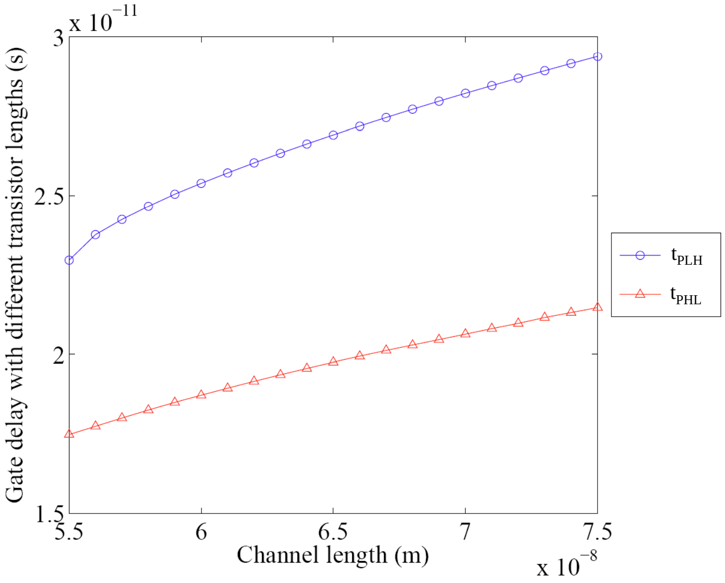


Figure 5.18: Delay of an inverter versus gate length.

Figure 5.18 shows SPICE-calculated delay values as gate lengths are varied in an inverter that is implemented in 65nm technology with equal channel lengths of the PMOS and NMOS devices. Figure 5.19 shows SPICE-calculated inverter delay values as gate widths of the PMOS and NMOS devices are changed by the same delta value. In Figures 5.18 and 5.19, t_{PLH} and t_{PHL} respectively denote the low to high propaga-

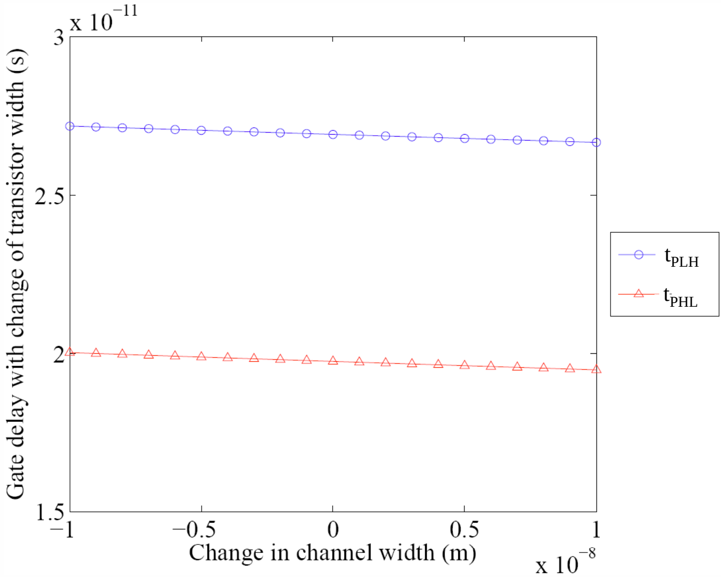


Figure 5.19: Delay of an inverter versus change in gate width.

tion delay and the high to low propagation delay. From the two figures, the gate delay varies linearly with both gate length and gate width around the nominal feature size (i.e., $65nm$) and the original transistor widths. Background experiments test Liberty [24] nonlinear delay model tables of 36 different $65nm$ standard cell masters, and confirm in all cell masters such an approximate linear relationship at each pair of input slew and load capacitance values. Similar studies at $90nm$ are conducted in [96].

When gate length and/or gate width changes in a small range, the effects of the change on other topologically adjacent gates are typically small.[10] Hence, we assume that the gate delay decreases linearly as the gate width increases, and increases linearly as the gate length increases. Since gate length (resp. width) changes linearly when the dose on the gate for poly (resp. active) layer varies, there is a linear relationship between the change of gate delay and the change of exposure dose on the gate for both poly and active layers, i.e., $\Delta t_p = t'_p - t_p = A_p \times \Delta L_p + B_p \times \Delta W_p = A_p \times D_s \times d^P_{i,j}(p) + B_p \times D_s \times d^A_{i,j}(p)$. Here, t_p and t'_p are the delay values of gate p before and after the percentage dose changes $d^P_{i,j}(p)$ and $d^A_{i,j}(p)$ on poly and active layers in the rectangular

[10] We recognize that off-path loading, slew propagation, and crosstalk timing windows can all change, and will be eventually accounted for precisely by golden signoff analysis. However, we assume in the proposed optimization framework – as is fairly standard in the sizing literature – that these effects are negligible, and we validate our experimental results below with golden signoff analysis.

grid $r_{i,j}$ where gate p is located, ΔL_p and ΔW_p are the changes in gate length and gate width of gate p, and A_p and B_p are fitted parameters that are dependent on input slew and load capacitance of each gate. In other words, for each distinct standard cell, and for each combination of input slew and load capacitance, different values of A_p and B_p are obtained from processing of Liberty nonlinear delay model tables. Total runtime of this procedure for a subset of a $65nm$ production standard-cell library (36 combinational cells and 9 sequential cells) is less than 1 minute on a single processor using our Liberty processing and curve-fitting utility. The fitted parameters can also be used to compute the change in gate delay when only the dose on poly layer changes (i.e., with only gate length modulation), in which case the dose change on active layer ($d^A_{i,j}(p)$) is 0.

For circuit delay calculation, without loss of generality we consider a combinational circuit with n gates as in [52]. Sequential circuits may be addressed similarly, e.g., by 'unrolling' them into combinational circuits that traverse from primary inputs and sequential cell outputs, to sequential cell inputs and primary outputs. For a given combinational circuit, we add to the corresponding circuit graph one fictitious source node which connects to all primary inputs, and one fictitious sink node which connects from all primary outputs. Nodes are indexed by a reverse topological ordering of the circuit graph, with the source and sink nodes indexed as $n + 1$ and 0, respectively.

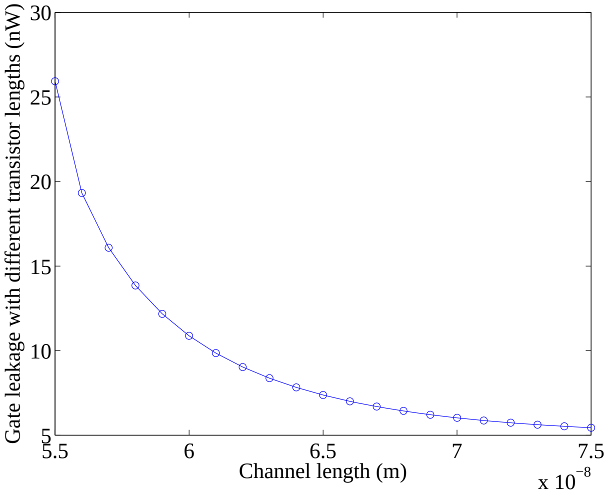


Figure 5.20: Average leakage of an inverter (INV_X1) versus gate length (VDD = 1.0V, Temperature = $25°C$, Process = TT).

Figure 5.20 shows SPICE-calculated average transistor leakage values with sim-

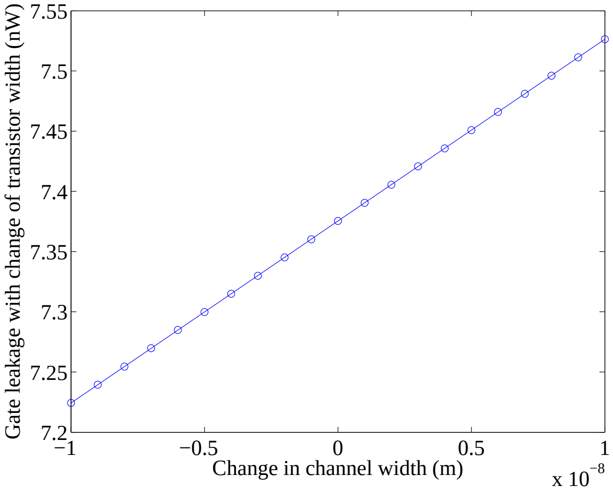


Figure 5.21: Average leakage of an inverter (INV_X1) versus the change in gate width (VDD = 1.0V, Temperature = $25°C$, Process = TT).

ulation condition (VDD = 1.0V, Temperature = $25°C$, Process = TT (i.e., typical corner of NMOS and PMOS)) as gate lengths are varied in a minimum-size inverter that is implemented in $65nm$ technology, where channel lengths of the PMOS and NMOS devices are equal. Figure 5.21 shows SPICE-calculated average transistor leakage values with the same inverter and simulation condition, as all channel widths of the PMOS and NMOS devices are changed by the same delta value. The figures show that leakage varies exponentially with gate length and linearly with the change in gate width, around the nominal feature size (i.e., $65nm$) and the original transistor widths. We have also performed background experiments on Liberty leakage values of 36 different standard cell masters, and confirmed these exponential (linear) relationships between leakage and gate length (width). Similar analyses for the $90nm$ technology node can be found in [96]. In the optimization, we assume that the change of leakage power of a gate is a quadratic function of the change in gate length[11] and a linear function of the change in gate width, i.e., $\Delta Leakage(\Delta L_p, \Delta W_p) = \alpha_p \times (\Delta L_p)^2 + \beta_p \times \Delta L_p + \gamma_p \times \Delta W_p$ for gate p. The calculation of the change in total leakage power of the gates in the circuit is given by Equation (5.4). Note that the parameters α_p, β_p and γ_p are gate-specific, i.e., different values of the parameters are used for different types of gates. Similar to the

[11] We recognize that leakage power is exponential in gate length. We use a quadratic approximation to facilitate the problem formulation and solution method.

computation of gate delay, the fitted parameters can also be used to compute the change in leakage power when only the dose on poly layer changes, in which case there is no dose change on active layer (i.e., ΔW_p = 0).

$$\begin{aligned} \Delta Leakage &= \textstyle\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p\in r_{i,j}} \alpha_p \times D_s^2 \times d_{i,j}^P(p)^2 \\ &\quad +\beta_p \times D_s \times d_{i,j}^P(p) + \gamma_p \times D_s \times d_{i,j}^A(p) \end{aligned} \tag{5.4}$$

5.2.2 Problem Formulation of Dose Map Optimization

For simplicity, we do not include dose-dependent change of wire delay in the proposed problem formulation; note that a dose map optimization on the poly and active layers will not affect wire layout patterns, and thus will not affect golden wire parasitics. In our proposed implementation, wire delay is obtained from golden static timing analysis reports and added in between gates.

Assume that the original dose in the chip area is uniform. The goal of the design-aware dose map optimization (*DMopt*) is to tune the dose maps on poly and active layers simultaneously to adjust the channel lengths and widths of the gates and thereby optimize circuit delay and/or total leakage power, subject to upper and lower bounds on delta dose values per grid, and a dose map smoothness bound to reflect the fact that exposure dose must change gradually between adjacent grids. In the following problem formulations, we use delta leakage instead of total leakage power to facilitate the computation. By minimizing (or, constraining) delta leakage, i.e., the change in total leakage power, the total leakage power will be minimized (or, constrained). To compute delta leakage power, three fitted parameters (i.e., α_p, β_p and γ_p) are needed as in Equation (5.4). However, to compute the total leakage power, four fitted parameters are needed (i.e., a constant item is needed besides coefficients α_p, β_p and γ_p) because we assume a quadratic relation between the change in leakage power and the change in doses on active and poly layers. Since delta leakage is sufficient for the following problem formulations, we use delta leakage rather than total leakage power to avoid the constant item in the estimation.

Design-aware dose map optimization on the poly layer. The design-aware dose map optimization for poly layer can be formulated as a quadratic program or a quadratic constraint program based on different types of constraints (i.e., linear or quadratic).

First, we optimize dose map on the poly layer for improved leakage under timing constraints. The optimization problem on poly layer is formulated as a quadratic program as follows.

- **Objective:** minimize $\Delta Leakage$
- **Subject to:**

$$L \leq d^P_{i,j} \leq U \quad \forall\, i \in [1, M],\ j \in [1, N] \tag{5.5}$$

$$\begin{cases} |d^P_{i,j} - d^P_{i+1,j+1}| \leq B \,\forall\, i \in [1, M-1],\ j \in [1, N-1] \\ |d^P_{i,j} - d^P_{i,j+1}| \leq B \quad \forall\, i \in [1, M],\ j \in [1, N-1] \\ |d^P_{i,j} - d^P_{i+1,j}| \leq B \quad \forall\, i \in [1, M-1],\ j \in [1, N] \end{cases} \tag{5.6}$$

$$\begin{cases} a_q \leq T \qquad\quad \forall\, q \in fanin(0) \\ a_r + t'_q \leq a_q \quad \forall\, r \in fanin(q) \quad (q = 1, \cdots, n) \\ 0 \leq a_{n+1} \\ t'_p = t_p + A_p \times D_s \times d^P_{i,j}(p) \qquad (p = 1, \cdots, n) \end{cases} \tag{5.7}$$

$$T \leq \tau_L \tag{5.8}$$

Equation (5.5) specifies the correction ranges on the dose for the poly layer, where L and U are user-specified or equipment-specific lower and upper bounds on the dose change. Equations (5.6) specify smoothness constraints on the dose for the poly layer, i.e., that the doses in neighboring rectangular grids should differ by a bounded amount.[12] Equation (5.7) specifies the delay constraint when the delays of the gates are scaled during the dose adjustment process. In Equation (5.7), a_q represents the arrival time at node q, which is the maximum delay from source node 0 to node p; $d^P_{i,j}(p)$ is the change in percentage of dose in rectangular grid $r_{i,j}$ on the poly layer in which gate p

[12] The dose generally varies gradually. To reflect the gradual property of dose profiles, the smoothness constraint is specified.

is located. The parameter A_p is gate-specific, and different values of the parameters are used for different types of gates as well as for gates of the same type that have different input slews and load capacitances. Equation (5.8) captures the user-specified upper bound (i.e., τ_L) on the delay of the longest path in the circuit. The calculation of the change in total leakage power of the gates $\Delta Leakage$ in the circuit is given by Equation (5.4), where only poly layer related leakage (i.e., $\alpha_p \times D_s^2 \times d_{i,j}^P(p)^2 + \beta_p \times D_s \times d_{i,j}^P(p)$) is computed. Since the constraints are linear and the objective is quadratic, this gives a quadratic program instance.

Second, we optimize dose map on the poly layer for improved timing under leakage constraints. The optimization problem on the poly layer is formulated as a quadratically constrained program as follows.

- **Objective:** minimize T
- **Subject to:** Equations (5.5), (5.6), (5.7), and

$$\Delta Leakage \leq \xi_L \tag{5.9}$$

Equations (5.5), (5.6), and (5.7) are as discussed in the previous problem formulation. Equation (5.9) specifies the constraint on the change in the total leakage power of all cell instances, where ξ_L is a user-specified parameter for the constraint. Since the constraint in Equation (5.9) is quadratic and the objective is linear, this yields a quadratically constrained program instance.

Design-aware dose map optimization on both poly and active layers. First, we optimize dose map on both poly and active layers for improved leakage under timing constraints. The optimization problem on both poly and active layers is formulated as a quadratic program as follows.

- **Objective:** minimize $\Delta Leakage$
- **Subject to:** Equations (5.5), (5.6), and

$$L \leq d_{i,j}^A \leq U \quad \forall\, i \in [1, M],\ j \in [1, N] \tag{5.10}$$

$$\begin{cases} |d^A_{i,j} - d^A_{i+1,j+1}| \leq B \, \forall \, i \in [1, M-1], \; j \in [1, N-1] \\ |d^A_{i,j} - d^A_{i,j+1}| \leq B \quad \forall \, i \in [1, M], \; j \in [1, N-1] \\ |d^A_{i,j} - d^A_{i+1,j}| \leq B \quad \forall \, i \in [1, M-1], \; j \in [1, N] \end{cases} \tag{5.11}$$

$$\begin{cases} a_q \leq T \qquad \forall \, q \in fanin(0) \\ a_r + t'_q \leq a_q \quad \forall \, r \in fanin(q) \quad (q = 1, \cdots, n) \\ 0 \leq a_{n+1} \\ t'_p = t_p + A_p \times D_s \times d^P_{i,j}(p) \\ \qquad + B_p \times D_s \times d^A_{i,j}(p) \qquad (p = 1, \cdots, n) \end{cases} \tag{5.12}$$

$$T \leq \tau_{WL} \tag{5.13}$$

Similar to Equations (5.6) for the poly layer, Equations (5.10) specify the correction ranges on the dose for the active layer. Equations (5.11) specify smoothness constraints on the dose for the active layer, and Equations (5.12) specify the delay constraint when the delays of the gates are scaled during the dose adjustment process on both poly and active layers. The variables a_p and $d^P_{i,j}(p)$ are defined as in Equation (5.7), and $d^A_{i,j}(p)$ is the change in percentage of dose in grid $r_{i,j}$ on the active layer wherein gate p is located. The parameter B_p is gate-specific, similar to the parameter A_p used in Equation (5.7). Equation (5.13) specifies the constraint on the delay of the longest path in the circuit, where τ_{WL} is a user-specified parameter for the constraint. The calculation of $\Delta Leakage$, the change in total leakage power of the gates in the circuit, is given by Equation (5.4) which considers the impact of both gate length and gate width variations on leakage power.

Second, we optimize dose map on both poly and active layers for improved timing under leakage constraints. The optimization problem on both poly and active layers is formulated as a quadratically constrained program as follows.

- **Objective:** minimize T
- **Subject to:** Equations (5.5), (5.6), (5.10), (5.11), (5.12), and

$$\Delta Leakage \leq \xi_{WL} \tag{5.14}$$

Equations (5.5), (5.6), (5.10), (5.11), and (5.12) are as discussed in previous problem formulations. Equation (5.14) specifies the constraint on the change in the total leakage power of all cell instances, where ξ_{WL} is a user-specified parameter for the constraint. Again, since the constraint in Equation (5.14) is quadratic and the objective is linear, we have an instance of quadratically constrained program.

The above problem formulations[13] result in either a quadratic program or a quadratically constrained program, which can be solved using classic quadratic programming methods. In particular, we use *ILOG CPLEX* [10] in the experimental platform described below.

5.2.3 Timing and Leakage Power Optimization Flow

Overall optimization flow. Figure 5.22 shows the whole flow integrating *DMopt* together with *dosePl* (discussed in Section 6.1 as an example of manufacturing-aware design optimization) for timing and leakage optimization. Note that the timing and leakage optimization flow is carried out after V_{th} and V_{dd} assignment processes. For the timing and leakage related dose map optimization problem, the input consists of (i) the original dose maps (i.e., those calculated to minimize ACLV and AWLV metrics, based on in-line metrology) for both poly and active layers, (ii) the characterized standard-cell timing libraries (or, other timing models that comprehend the impact of dose on transistor gate lengths and widths) for different gate lengths and gate widths, and (iii) the circuit with placement and routing information. By "placement and routing information", we also include implicit information that is necessary for timing and power analyses, e.g., extracted wiring parasitics. With the nominal gate-length cell timing and power libraries, and the circuit itself with its placement, routing and parasitic data, timing analysis can be performed to generate the input slews and output load capacitances of all the cell instances. With the input slews and output load capacitances of all the cell instances, the original dose maps, and characterized cell libraries of different gate lengths and gate widths, the dose map optimization is executed to determine doses that adjust gate lengths and gate widths of the cells for timing and leakage optimization,

[13] The optimization result is feasible for the Dose Mapper equipment, as a consequence of the constraints (5.5), (5.6), (5.10) and (5.11).

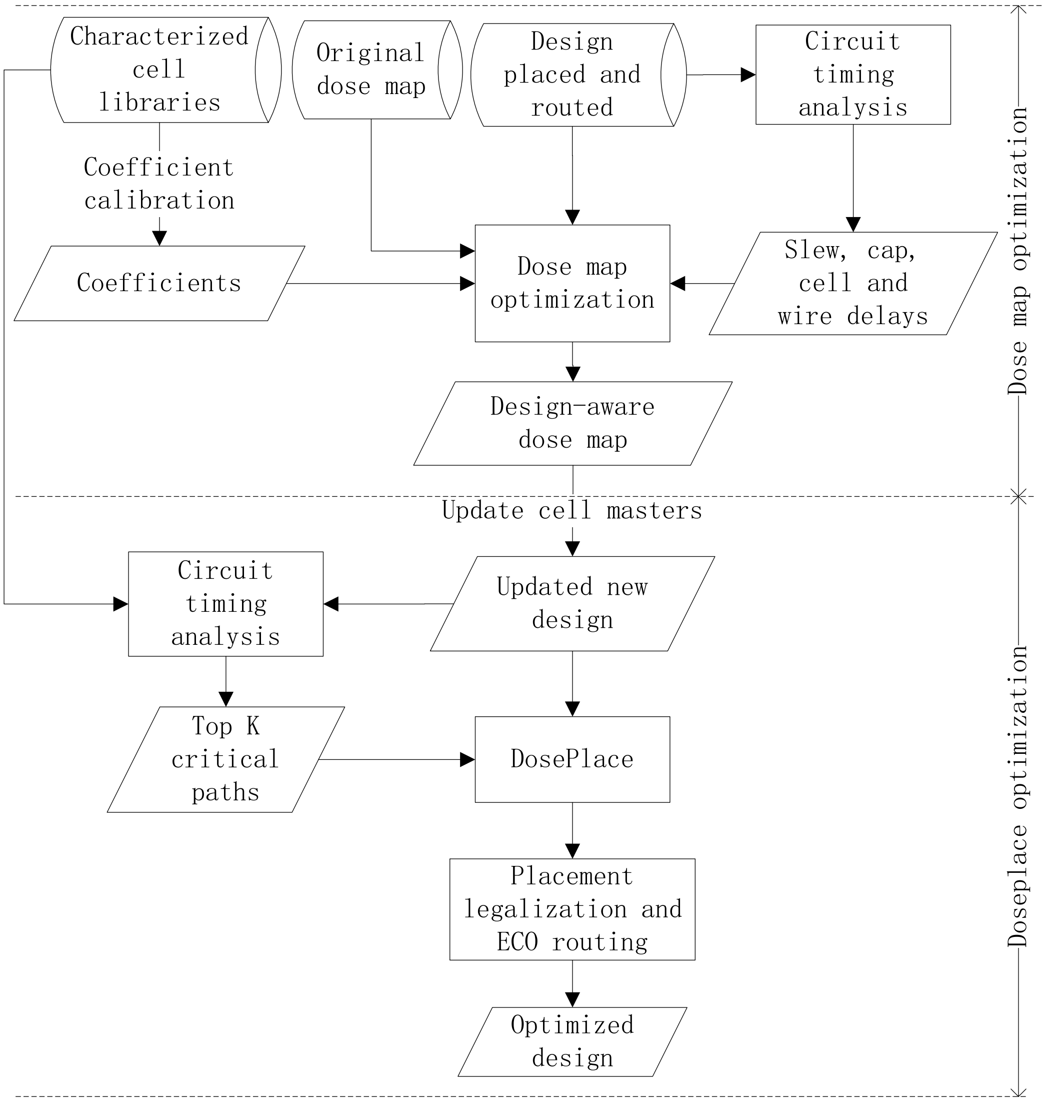


Figure 5.22: Flow of the timing and leakage power optimization with integrated *DMopt* and *dosePl* (in Section 6.1).

subject to dose map constraints. Finally, the optimized design-aware dose maps on both layers are generated.

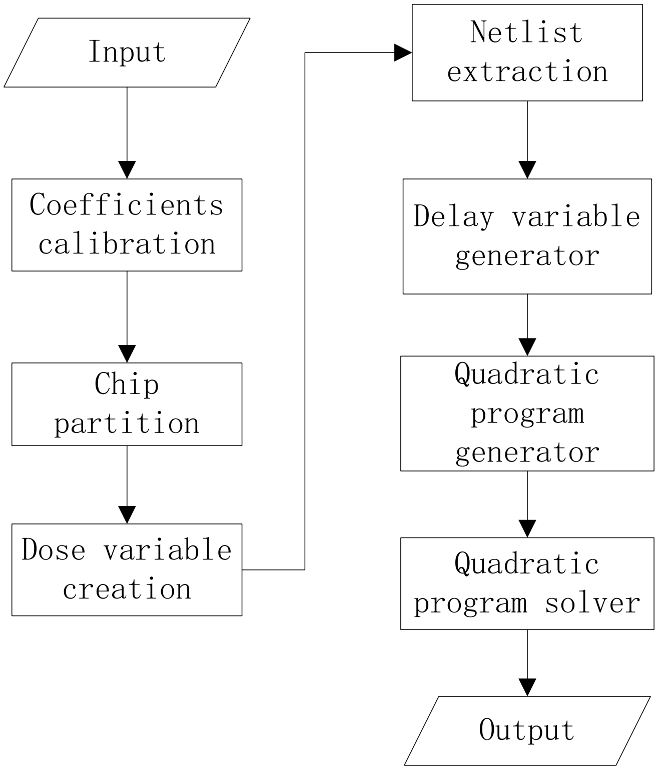


Figure 5.23: Detailed view of design-aware dose map optimization flow.

According to the optimized design-aware dose maps on both poly and active layers, the cell instances in different grids of the dose maps will have different gate lengths and widths as well as different cell masters in the characterized cell libraries.[14] Thus, the design's netlist representation must be updated according to the dose maps. Using the characterized cell libraries, timing analysis is performed on the new design with the updated cell masters to identify the top-k (e.g., $k = 10,000$) critical paths for a complementary *dosePl* optimization process (see Section 6.1). The *dosePl* is a manufacturing-aware design optimization based on a cell swapping strategy, which may introduce an illegal placement result. Therefore, a legalization process is invoked to

[14] When the gate lengths and widths are computed from the optimized dose maps, it is possible that the computed values do not exactly match the available drive strengths of the cell masters in the characterized cell libraries. Thus, a rounding step is needed to snap the computed gate lengths and widths to the cell masters that have most-similar drive strengths.

legalize the swapped cells. ECO routing is then executed for the affected wires to refine the design with optimized timing yield.

Summary of the dose map optimization flow. The dose map optimization in Figure 5.23 is summarized as follows. The input consists of the original dose maps on both layers, the characterized cell libraries of different gate lengths and widths, and the input slews and output capacitances of all the cells in the circuit. From the characterized 65nm cell libraries of different gate lengths and widths and 90nm cell libraries of different gate lengths[15], the coefficients in the linear function of delay, and the quadratic function of leakage power, are calibrated. Note that when gate delay calculation in the cell libraries adopts a lookup table method, where the entries are indexed by input slews and output capacitances, the coefficients of the delay functions may be calibrated for each entry in each delay table. Then, according to the input slew and output capacitance values that were obtained for each cell in the previous step, the coefficients associated with the nearest entry (or, entries with interpolation) in the table are applied to calculate the delay of the cell.

The exposure fields on both poly and active layers are then partitioned into rectangular grids. For each grid on the poly (active) layer, a variable $d^P_{i,j}$ ($d^A_{i,j}$) represents the amount of dose change in the grid. Maximum circuit delay is captured using variable a_p that represents the arrival time at the output of cell p. When all the variables are obtained, a quadratic program (resp. quadratically constrained program) problem instance is generated by introducing the dose map correction range constraints, dose map smoothness constraints, and delay constraints, as well as the objective of minimizing the total leakage power of all the cells under timing constraints (resp. minimizing the timing of the circuit under leakage constraints). Finally, a quadratic program (resp. quadratically constrained program) solver finds the optimal dose change in each grid based on the original dose maps; this yields optimal design-aware dose maps.

[15] We focus on the dose map optimization methods for 65nm testcases. However, 90nm testcases are also used in dose map optimization for gate length modulation to provide supporting experimental data.

5.2.4 Experimental Results

To assess the effectiveness of the proposed dose map optimization algorithms, we first sweep the dose change on the poly layer from -5% to $+5\%$ for all the cell instances in the $65nm$ design *AES65* and the $90nm$ design *AES90* (shown in Table 5.8); we perform timing analysis using *Synopsys PrimeTime version Z-2006.12* [25] and leakage power estimation using *Cadence SOC Encounter version 7.10* [7]. The timing analysis and leakage power estimation are based on pre-characterized $65nm$ and $90nm$ cell libraries with gate length and gate width variants.

Table 5.8: Characteristics of $65nm$ and $90nm$ designs implemented with Artisan TSMC library.

Design	Chip size (mm^2)	#Cell instances	#Nets
AES65	0.058	16187	16450
JPEG65	0.268	68286	68311
AES90	0.25	21944	22581
JPEG90	1.09	98555	105955

Delay and leakage power results are given in Table 5.9 and Table 5.10, where "MCT" refers to minimum cycle time and "P_{leak}" refers to the total leakage power of all the cells. The extreme cases of dose change on the poly layer correspond to maximum timing yield improvement ($d^P_{i,j} = +5$) or leakage power reduction ($d^P_{i,j} = -5$). The results show that timing yield improvement can be obtained at the cost of leakage power increase, and leakage power reduction can be obtained at the cost of timing yield degradation. Uniform dose change in all the cell instances cannot obtain timing yield improvement without leakage power increase. However, the proposed dose map optimization algorithms can obtain substantial timing yield improvement without increase in total leakage power, as well as leakage power reduction without degradation in timing yield.

The timing and leakage optimization flow is implemented in C++ and tested on industrial testcases as given in Table 5.8. In Table 5.8, there are two different classes of testcases. *AES65* and *JPEG65* are $65nm$ designs, and *AES90* and *JPEG90* are $90nm$

Table 5.9: Delay and leakage values of $65nm$ design *AES65* when dose change $d^P_{i,j}$ is swept from 0% to -5% and from 0% to $+5\%$ on the poly layer. The simplistic, uniform increase of dose cannot obtain delay improvement without incurring leakage increase.

Dose change	$d^P_{i,j}$ = 0	$d^P_{i,j}$ = -1	$d^P_{i,j}$ = -2	$d^P_{i,j}$ = -3	$d^P_{i,j}$ = -4	$d^P_{i,j}$ = -5
MCT (ns)	1.638	1.677	1.715	1.750	1.786	1.824
imp. (%)	–	-2.38	-4.70	-6.84	-9.04	-11.36
P_{leak} (μW)	448.0	397.7	356.9	324.8	299.9	279.6
imp. (%)	–	11.23	20.33	27.50	33.06	37.59
Dose change	$d^P_{i,j}$ = 0	$d^P_{i,j}$ = +1	$d^P_{i,j}$ = +2	$d^P_{i,j}$ = +3	$d^P_{i,j}$ = +4	$d^P_{i,j}$ = +5
MCT (ns)	1.638	1.601	1.557	1.517	1.474	1.427
imp. (%)	–	2.26	4.95	7.39	10.01	12.88
P_{leak} (μW)	448.0	513.4	600.4	722.2	893.5	1142.2
imp. (%)	–	-14.60	-34.02	-61.21	-99.44	-154.96

Table 5.10: Delay and leakage values of $90nm$ design *AES90* when dose change $d^P_{i,j}$ is swept from 0% to -5% and from 0% to $+5\%$ on the poly layer. The simplistic, uniform increase of dose cannot obtain delay improvement without incurring leakage increase.

Dose change	$d^P_{i,j}$ = 0	$d^P_{i,j}$ = -1	$d^P_{i,j}$ = -2	$d^P_{i,j}$ = -3	$d^P_{i,j}$ = -4	$d^P_{i,j}$ = -5
MCT (ns)	1.990	2.031	2.078	2.115	2.155	2.188
imp. (%)	–	-2.08	-4.40	-6.30	-8.28	-9.95
P_{leak} (μW)	2430.2	2225.1	2054.5	1914.5	1796.6	1699.8
imp. (%)	–	8.44	15.46	21.22	26.08	30.06
Dose change	$d^P_{i,j}$ = 0	$d^P_{i,j}$ = +1	$d^P_{i,j}$ = +2	$d^P_{i,j}$ = +3	$d^P_{i,j}$ = +4	$d^P_{i,j}$ = +5
MCT (ns)	1.990	1.950	1.905	1.868	1.818	1.758
imp. (%)	–	2.03	4.26	6.16	8.65	11.66
P_{leak} (μW)	2430.2	2678.1	2995.0	3404.1	3939.8	4619.0
imp. (%)	–	-10.20	-23.24	-40.07	-62.12	-90.07

Table 5.11: Results of dose map optimization on poly layer, i.e., gate length (L_{gate}) modulation with smoothness bound $B = 2$, dose correction range $\pm 5\%$ and $5 \times 5um^2$ grids.

AES65	Nom L_{gate}	$5 \times 5um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	1.638	1.631	0.44	1.607	1.89
P_{leak} (μW)	448.0	409.7	8.54	441.3	1.49
Runtime (s)	–	72	–	108	–

JPEG65	Nom L_{gate}	$5 \times 5um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	2.179	2.174	0.25	2.081	4.52
P_{leak} (μW)	2915.5	2312.7	20.67	2922.3	-0.23
Runtime (s)	–	490	–	891	–

AES90	Nom L_{gate}	$5 \times 5um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	1.990	1.975	0.75	1.861	6.47
P_{leak} (μW)	2430.2	1823.2	24.98	2386.1	1.82
Runtime (s)	–	176	–	227	–

JPEG90	Nom L_{gate}	$5 \times 5um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	2.906	2.894	0.41	2.667	8.23
P_{leak} (μW)	4354.2	3422.5	21.40	4244.4	2.52
Runtime (s)	–	2157	–	3644	–

Table 5.12: Results of dose map optimization on poly layer, i.e., gate length (L_{gate}) modulation with smoothness bound $B = 2$, dose correction range $\pm 5\%$ and $10 \times 10um^2$ grids.

AES65	Nom L_{gate}	$10 \times 10um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	1.638	1.632	0.35	1.626	0.71
P_{leak} (μW)	448.0	434.3	3.05	445.4	0.57
Runtime (s)	–	18	–	335	–

JPEG65	Nom L_{gate}	$10 \times 10um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	2.179	2.178	0.04	2.102	3.54
P_{leak} (μW)	2915.5	2480.9	14.91	2913.4	0.07
Runtime (s)	–	292	–	558	–

AES90	Nom L_{gate}	$10 \times 10um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	1.990	1.981	0.44	1.872	5.91
P_{leak} (μW)	2430.2	1901.6	21.75	2370.2	2.47
Runtime (s)	–	85	–	145	–

JPEG90	Nom L_{gate}	$10 \times 10um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	2.906	2.901	0.16	2.689	7.45
P_{leak} (μW)	4354.2	3453.6	20.68	4273.5	1.85
Runtime (s)	–	1194	–	2068	–

Table 5.13: Results of dose map optimization on poly layer, i.e., gate length (L_{gate}) modulation with smoothness bound $B = 2$, dose correction range $\pm 5\%$ and $30 \times 30um^2$ grids.

AES65	Nom L_{gate}	$30 \times 30um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	1.638	1.637	0.07	1.637	0.07
P_{leak} (μW)	448.0	447.9	0.01	447.1	0.19
Runtime (s)	–	9	–	46	–

JPEG65	Nom L_{gate}	$30 \times 30um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	2.179	2.172	0.31	2.159	0.91
P_{leak} (μW)	2915.5	2843.1	2.48	2909.8	0.19
Runtime (s)	–	61	–	929	–

AES90	Nom L_{gate}	$30 \times 30um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	1.990	1.989	0.05	1.927	3.19
P_{leak} (μW)	2430.2	2172.4	10.61	2406.0	1.00
Runtime (s)	–	16	–	92	–

JPEG90	Nom L_{gate}	$30 \times 30um^2$ grids			
		QP	imp. (%)	*QCP*	imp. (%)
MCT (ns)	2.906	2.887	0.65	2.757	5.11
P_{leak} (μW)	4354.2	3822.2	12.22	4308.3	1.06
Runtime (s)	–	243	–	2545	–

designs. In the experiments, the dose sensitivity D_s is -2nm/%. The parameters A_p, B_p, α_p, β_p and γ_p are calibrated using *Synopsys PrimeTime* [25] and *Cadence SOC Encounter* [7] based on the pre-characterized cell timing and leakage libraries. Since different libraries (i.e., 90nm and 65nm) are used for different designs, two sets of parameters are calibrated from the different libraries and used in the dose map optimization for the corresponding testcases. Tables 5.11, 5.12, and 5.13 show the dose map optimization results for the poly layer. In the tables, "QP" refers to the quadratic program for improved total leakage under timing constraint, and "QCP" refers to the quadratically constrained program for improved timing under leakage constraint. Different sizes of rectangular grids are used in the dose map optimization, i.e., $5 \times 5um^2$, $10 \times 10um^2$, and either $30 \times 30um^2$ (for 65nm cases) or $50 \times 50um^2$ (for 90nm cases). The dose smoothness bound is $B = 2$,[16] and the dose correction range is $\pm 5\%$. From the results, the finer the rectangular grids, the greater the improvement in the timing of the circuit or in the total leakage power.

Table 5.14: Percentage of critical timing paths in testcases.

Design	$95 \sim 100\%$ MCT (%)	$90 \sim 100\%$ MCT (%)	$80 \sim 100\%$ MCT (%)
AES65	16.54	28.98	41.98
JPEG65	4.80	9.89	30.23
AES90	0.91	4.54	22.84
JPEG90	0.12	0.35	3.92

We observe different optimization quality between 90nm testcases (*AES90* and *JPEG90*) and 65nm testcases (*AES65* and *JPEG65*). Average leakage reduction for 90nm testcases under timing constraints with $5 \times 5um^2$ grids is 23.2%, but for 65nm testcases is 14.6%. Average MCT reduction for 90nm testcases under leakage constraints with $5 \times 5um^2$ grids is more than 7.4%, but that of 65nm testcases shows 3.4%. There are two reasons for the above optimization discrepancy between 90nm and 65nm designs. The first reason is that $5 \times 5um^2$ grids have different granularities for the different designs. From Table 5.8, the average number of cell instances in a grid of $5 \times 5um^2$

[16] Different smoothness bounds may be specified in the slit and scan directions (see Section 5.2.1). Here, we use an average example value for both directions.

is 2.2 for the 90nm testcases and 6.3 for the 65nm testcases. As discussed above, the finer the rectangular grids, i.e., the fewer cell instances in one grid, the better the optimization quality. The smaller average number of cell instances per grid for the 90nm testcases permits larger improvements. The second reason is the difference in timing criticality (slack distribution) of the testcases before optimization. Table 5.14 shows the timing criticality of each testcase as the number of critical paths within a specific range of timing. More paths in the 65nm testcases have delay values near the MCT, which makes it difficult for the dose map optimization to remove all those paths to improve timing. However, in the 90nm testcases, the number of such critical paths is small, making it easier for the dose map optimization to improve timing. For these reasons, more substantial leakage and timing improvements are observed for the 90nm testcases.

Table 5.15: Results of dose map optimization on both poly and active layers using quadratic program for improved leakage power, i.e., gate length (L_{gate}) and gate width (W_{gate}) modulation, with smoothness bound $B = 2$ and dose correction range $\pm5\%$.

AES65	Nom	Grids	L_{gate}	imp. (%)	*both*	imp. (%)
	L_{gate}&W_{gate}	(μm^2)				
MCT (ns)	1.638	5 × 5	1.631	0.44	1.635	0.18
		30 × 30	1.637	0.07	1.631	0.45
P_{leak} (μW)	448.0	5 × 5	409.7	8.54	383.8	14.33
		30 × 30	447.9	0.01	444.9.0	0.69
Runtime (s)	–	5 × 5	72	–	110	–
		30 × 30	9	–	13	–

JPEG65	Nom	Grids	L_{gate}	imp. (%)	*both*	imp. (%)
	L_{gate}&W_{gate}	(μm^2)				
MCT (ns)	2.179	5 × 5	2.174	0.25	2.177	0.09
		30 × 30	2.172	0.31	2.179	0.01
P_{leak} (μW)	2915.5	5 × 5	2312.7	20.67	2301.1	21.07
		30 × 30	2843.1	2.48	2763.2	5.22
Runtime (s)	–	5 × 5	490	–	1232	–
		30 × 30	61	–	93	–

Table 5.16: Results of dose map optimization on both poly and active layers using quadratically constrained program for improved timing, i.e., gate length (L_{gate}) and gate width (W_{gate}) modulation, with smoothness bound $B = 2$ and dose correction range $\pm 5\%$.

AES65	Nom	Grids	L_{gate}	imp. (%)	*both*	imp. (%)
	L_{gate}&W_{gate}	(μm^2)				
MCT (ns)	1.638	5×5	1.601	1.89	1.586	3.17
		30×30	1.647	0.07	1.630	0.48
P_{leak} (μW)	448.0	5×5	441.3	1.49	447.0	0.22
		30×30	447.9	0.01	446.5	0.32
Runtime (s)	–	5×5	108	–	179	–
		30×30	46	–	141	–

JPEG65	Nom	Grids	L_{gate}	imp. (%)	*both*	imp. (%)
	L_{gate}&W_{gate}	(μm^2)				
MCT (ns)	2.179	5×5	2.081	4.52	2.090	4.10
		30×30	2.159	0.91	2.153	1.21
P_{leak} (μW)	2915.5	5×5	2922.3	-0.23	2922.0	-0.22
		30×30	2909.8	0.19	2907.9	0.26
Runtime (s)	–	5×5	891	–	1561	–
		30×30	929	–	3184	–

Table 5.15 shows the dose map optimization results using the quadratic program for improved leakage power on both poly and active layers $65nm$ designs. From the results, slightly better leakage improvement is obtained using simultaneous modulation of both gate length and gate width. Table 5.16 shows the dose map optimization results using the quadratically constrained program for improved timing on both poly and active layers. Again, only the $65nm$ designs are tested, and again slightly better timing improvement is obtained using simultaneous modulation of gate length and gate width than only using gate length modulation. The maximum change in gate width is $10nm$ according to the dose sensitivity -2nm/% and dose correction range $\pm 5\%$; this is relatively small when compared with the transistor widths of cells in the $65nm$ standard

cell library (the minimum transistor width in 65nm cells is around 200nm, while the maximum width is more than 650nm). As a result, there is only slight impact of gate width modulation on the cell's delay and leakage, and the related timing and/or leakage improvements are not significant.

In one case (JPEG-65 with $5 \times 5um^2$ grids in Table 5.16), the dose map optimization using simultaneous gate width and gate length modulation obtains slightly worse results than using only gate length modulation. We attribute this to the use of more fitted parameters (i.e., B_p and γ_p for gate width related delay and leakage) in estimation of cell delay and leakage, which can introduce more estimation errors. From the Liberty delay model tables of 36 different 65nm standard cell masters, for all the arcs (i.e., rise and fall) with all the slew/load combinations, we perform curve fitting for cell delay versus gate length using the least square method. When only gate length changes, 21 different characterized libraries are needed corresponding to the 21 different dose values for poly layer in Table 5.9. In this case, the maximum sum of squares of the residuals for all the fitted curves is 0.0005. When both gate length and gate width change, a total of 441 (i.e., 21 $\times$ 21) characterized libraries are needed, which is a combination of 21 different dose values for the poly layer (i.e., the change in gate length) and 21 different dose values for the active layer (i.e., the change in gate width). In this case, the maximum sum of squares of the residuals for all the fitted curves is 0.0101, which is much larger than 0.0005. The increased error in curve fitting may be caused by the increased number of variables (i.e., gate width) and the increased number of characterized libraries.

From the results in Table 5.9 and Table 5.10, we see that smaller dose change results in smaller timing improvement, e.g., Table 5.9 shows $d^P_{i,j} = +1$ corresponding to 2.05% timing improvement versus $d^P_{i,j} = +5$ corresponding to 10.42% improvement. Therefore, tighter smoothness bounds (i.e., $B < 2$) will result in smaller timing improvement by enforcing smaller available dose changes within each rectangular grid. By testing different sizes of the rectangular grids, the smoothness bounds are also elaborated, i.e., the effective smoothness bound of a given smoothness value is different for different rectangular grids. For example, the effective smoothness bound of smoothness value $B = 2$ over $50 \times 50\mu m^2$ grids (i.e., 2%/50μm) is tighter than that over $10 \times 10mum^2$ grids (i.e., 2%/10μm). As mentioned earlier in Section 5.2, the

Zeiss/Pixer Critical Dimension Control (CDC) technology also enables adaptivity in the manufacturing flow to meet the required CD specifications. We note that the proposed methods can be used for any emerging technology that enables the fine-grain tuning of CD (i.e., along with relaxed effective smoothness bound) during manufacturing. Moreover, the sizes of used testcases (Table 5.8) are very small, with the largest area ($90nm$ *JPEG90*) being only a little over $1mm^2$. For designs of larger sizes, we anticipate that the proposed methods will obtain better timing and leakage improvements.

5.3 Conclusions and Research Directions

In this chapter, we have proposed two design-aware manufacturing process optimizations. Our first study provides new yield-aware mask strategies to mitigate emerging variability and defectivity challenges. We have analyzed CD variability with respect to reticle size, and quantified its impact on parametric yield. We have also integrated parametric yield depending on field size with a cost model that incorporates mask, wafer, and processing cost considering throughput, yield, and manufacturing volume. This enables assessment of various reticle strategies (e.g., single layer reticle (SLR), multiple layer reticle (MLR), and small and large size) considering field-size dependent parametric yield. Another aspect of our study addresses defect-induced parametric yield in EUVL, where we assess the sensitivity of parametric yield to several defect parameters, i.e., defect density, height, distribution and influence distance. We then compare parametric yields of various reticle strategies. The analysis results confirm a clear cost benefit from use of small-field rather than traditional full-field reticles when the volume size is small. Furthermore, we have shown that small-size field in EUVL can have significantly higher parametric yield in light of EUVL mask blank defectivity.

Future research seeks to update the cost model for future technologies with various mask and patterning technologies (DPL, EUVL, imprint, etc.), and to include more data for various design types (SOC, MPU, ASIC, etc.) and design sizes in order to derive realistic design-dependent defectivity requirements.

The second study proposes a method to improve the timing yield of the circuit and reduce total leakage power, using design-aware dose map and dose map-aware placement optimization. We focus mainly on the placement-aware dose map optimization. The complementary dose map-aware placement optimization (see Section 6.1) takes as input a placement-aware timing and leakage optimized dose map. As an extension for dose map optimization, we seek to minimize the delay variation of different chips across the wafer or the exposure field.

5.4 Acknowledgments

Chapter 5 is in part a reprint of “New Yield-Aware Mask Strategies”, to appear in *Proc. Photomask and Next-Generation Lithography Mask Technology*, 2011, “Dose Map and Placement Co-Optimization for Improved Timing Yield and Leakage Power”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(7) (2010), and “Dose Map and Placement Co-Optimization for Timing Yield Enhancement and Leakage Power Reduction”, *Proc. ACM/IEEE Design Automation Conference*, 2008. I would like to thank my coauthors Professor Andrew B. Kahng, Dr. Chris Progler, Dr. Chul-Hong Park, and Professor Hailong Yao.

Chapter 6

Manufacturing-Aware Design Optimization

In this chapter, we present novel design techniques that are targeted towards recent advanced manufacturing techniques. First, we propose a placement optimization technique to be used with *Dose Mapper* (Section 5.2). Second, we propose new bimodal-aware timing analysis for standard cell-based designs manufactured using DPL (Section 4.3.3). Third, we present new 1-D regular-pitch SRAM bitcell layouts which are amenable to patterning by interference-assisted lithography (IAL) (Section 4.1.1).

6.1 Dose Map-Aware Placement Optimization

Given the discussion of a placement-specific dose map in Section 5.2, it is natural to ask whether a dose map-specific placement can further improve the result. In this section, we describe a simple cell swapping-based dose map-aware placement (*dosePl*) optimization.

6.1.1 Dose Map-Aware Placement

The *dosePl* problem can be stated as follows. Given the original placement result and a timing- and leakage-aware dose map, determine cell pairs to swap for timing yield improvement. We define the *bounding box of a cell* as the bounding box of all the cell's fanin cells and fanout cells, as well as the cell itself. Figure 6.1 shows the bounding box of a 3-input NAND (NAND3) cell, denoted by the dashed line.

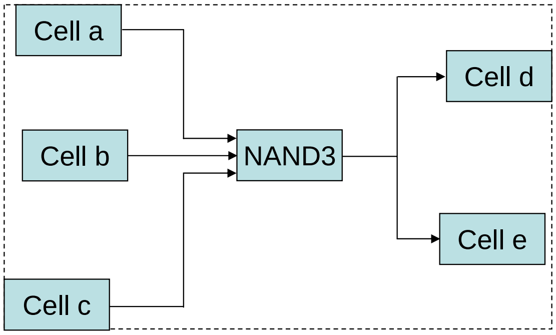


Figure 6.1: Bounding box of a 3-input NAND (NAND3) cell: moving the cell within its bounding box has a lower likelihood of increasing total wirelength.

The basic idea behind the cell swapping-based optimization method is to swap cells on timing-critical paths (referred to as *critical cells* henceforth) to high-dose regions and non-timing critical cells to low-dose regions, to further enhance the circuit performance subject to a leakage constraint. The underlying intuition is that moving a cell within its bounding box has a lower likelihood of increasing total wirelength or timing delay than moving it outside its bounding box. Thus, we seek pairs of cells l with bounding box b_l and cell m with bounding box b_m in different dose regions, such that cell l is in b_m and cell m is in b_l. With this restriction, we filter out candidate cell swaps that can be too disruptive to wirelength and timing.

Additional heuristics to avoid wirelength increase. When two cells satisfy the condition that they are located in each other's bounding boxes, it is still possible for total wirelength to increase. We thus adopt the following heuristics to further filter out unpromising cell pairs. For the filtering, we use technology- and design-specific tunable parameters γ_1, γ_2, γ_3, γ_4 and γ_5.

(1) Distance between the two cells to be swapped. When the distance between two cells is very large, the impact of cell swapping on total wirelength is potentially large. Therefore, we avoid considering swaps of cells that are farther apart than a predefined distance threshold (γ_2).[1]

[1] In the experimental results below, this threshold is chosen proportionally to the gate pitch, which is computed as the chip dimension divided by the square root of gate count in the chip.

(2) Changes in half-perimeter wirelength (HPWL) comparison. We also filter cell swaps by computing updated HPWL estimates; only if the estimated wirelength increase for all incident nets (e.g., the four nets incident to the NAND3 cell in Figure 6.1) is below a predefined threshold (γ_3) (e.g., 20% in the experiments reported below) will the cell swap be attempted.

On the number of swaps and cell priority. For a given critical path, a few cell swaps may suffice to reduce the path delay, and excessive cell swapping may introduce unnecessary wirelength and leakage increase. So, an upper bound on the number of cells swapped (γ_1) for each critical path is specified in the proposed heuristic's implementation (e.g., one cell per critical path in the experiments below). The priority for a critical cell during swapping is decided according to the following two factors.

(1) Number of critical paths that pass through the cell. The more critical paths that pass through a given cell, the more beneficial it is to swap the cell to a higher-dose region. Higher priorities are assigned to cells that are on a larger number of critical paths.

(2) Slack of critical paths. The larger the total path delay of a given critical path, the more important it is to swap cells on the path to achieve delay reduction. Therefore, higher priorities are assigned to cells on paths with greater timing criticality (i.e., smaller slack).

Based on the above two heuristic factors, critical cells are assigned weights as calculated in Equation (6.1), where C_l is the set of critical paths on which cell l is located. In the proposed implementation, cells are processed path by path (obtained from golden timing analysis), in order from most timing-critical to least timing-critical. Therefore, cells on more-critical paths always have higher priorities than cells on less-critical paths. Cells in the same critical path are prioritized (processed) in non-increasing order of weights that are computed as

$$W(cell_l) = \sum_{cell_l \in C_l} e^{-slack(C_l)}. \tag{6.1}$$

On the allowable leakage power increase. When cells l and m are to be swapped, the increase in their combined leakage power $\Delta I_{off}(l,m)$ is estimated beforehand. If $\Delta I_{off}(l,m)$ is less than a given fraction (γ_4) (e.g., 10%) of the original leakage power

$I_{off}(l, m)$ of the two cells, they will be swapped. Otherwise, no swapping will be performed, so as to avoid large leakage increase. Because one cell is swapped to a higher-dose region (i.e., leakage increases) and the other one is swapped to a lower-dose region (i.e., leakage decreases), it is not always the case that cell swapping will decrease leakage power.

6.1.2 Cell-Swapping Heuristic

The pseudocode of one round of the proposed cell-swapping heuristic is given in Figure 6.2. In each round of cell swapping, a maximum of γ_5 swaps are allowed (e.g., one swap for each round of cell swapping in the experiments). The cell-swapping process is based on the critical paths, which are first sorted in non-decreasing order of their timing slack. A cell in a timing-critical path are then swapped with another cell in a non-timing critical path. Since it is not necessary to swap all the cells in a critical path to improve its timing, the swapping process for a path is terminated when the number of swapped cells reaches a user-defined parameter γ_1 (in experiments, up to one cell is swapped on each path). For a given candidate swapping pair, the swapping process checks the bounding-box constraint, the dose constraint and the distance, then computes HPWL increase and leakage increase when the pair is swapped. If a candidate pair passes all the checks, its cells are swapped and we update the number of swapped cells for the affected critical paths. The cell-swapping process continues until all critical paths are processed, or the number of swaps reaches γ_5. When one round of the swapping process finishes, the perturbed placement is legalized and routed by *engineering change order* (ECO) placement and routing processes. After the final ECO routing, golden timing analysis is performed with updated parasitics to evaluate the timing improvement of the circuit. If the circuit delay is improved, the swapping is accepted. Otherwise, the swapped cell instances are rolled back to their previous locations and another round of cell swapping is performed with those swapped cells marked as fixed (i.e., those cells cannot be swapped again in the following cell-swapping process). In the experiments reported below, the total number of rounds of cell swapping is 10. A larger number of swapping rounds can achieve higher timing improvement. However, the improvement cannot be guaranteed because only swapping cells on worst-slack paths can improve timing yield, and such cells may not be swappable due to the swapping constraints (distance, leakage increase, etc.).

```
Algorithm:
Find cells in top-$k$ critical paths by golden timing analysis;
Compute weights for critical cells as in Equation (6.1);
Sort critical paths in non-decreasing order according to their slacks;
Set $numSwaps \leftarrow 0$;
for $i = 1$ to $k$ do
  Sort the cells in critical path $c_i$ in non-increasing order according to their weights;
  for each cell $l \in$ critical path $c_i$ do
    if # swapped cells in path $c_i$ $n(c_i) > \gamma_1$ then break; end if
    Compute bounding box $b_l$ of cell $l$ in path $c_i$;
    Compute the set $R$ of rectangular grids that intersect with $b_l$;
    Sort the grids $r \in R$ in non-increasing order according to their doses $d(r)$;
    Set $flag \leftarrow$ FALSE;
    for each $r \in R$ do
      if $d(r) < d(l)$ then break; end if // $d(l)$ is the dose on cell $l$
      Sort the non-critical cells $NC$ in grid $r$ in non-decreasing order by Manhattan distance from cell $l$;
        for each cell $m \in NC$ do
          if $dist(l, m) > \gamma_2$ then break; end if
          if $l \in b_m$ and $m \in b_l$ and $\Delta HPWL(l) < \gamma_3$ and $\Delta HPWL(m) < \gamma_3$ and $\Delta I_{off}(l,m) < I_{off}(l,m) \cdot \gamma_4$
          then
            Swap $(l, m)$;
            Update the number of swapped cells $n(c_s)$ for each critical paths $c_s$ such that cell $l \in c_s$;
            Set $flag \leftarrow$ TRUE;
            $numSwaps$ ++;
            if $numSwaps \geq \gamma_5$ then return; end if
            break;
          end if
        end for
      if $flag =$ TRUE then break; end if
    end for
  end for
end for
```

Figure 6.2: One round of cell swapping heuristic in *dosePl*.

6.1.3 Experimental Results

We perform the proposed placement optimization with the dose map optimization (Section 5.2). The experimental results of dose map-placement co-optimization are given in Table 6.1. Testcases are partitioned into rectangular grids of size $5 \times 5 \mu m^2$, the *dose correction range* is $\pm 5\%$, and *dose smoothness bound* is $B = 2$ as the experiments in Section 5.2.4. Dose change $d^P_{i,j}$ is a percentage value which specifies the relative changes of dose for poly in the rectangular grid $r_{i,j}$. MCT refers to minimum cycle time. *DMopt* (QCP; see Section 5.2.2) first improves the timing yield under leakage-power constraint. Cell swapping-based *dosePl* is used to further improve the results.

Table 6.1: Results of dose map optimization on poly layer using quadratically-constrained programming (Section 5.2.2) for improved timing, followed by incremental placement optimization (*dosePl*).

Testcase	$AES65$			$JPEG65$		
	Nom L_{gate}	*QCP*	*dosePl*	Nom L_{gate}	*QCP*	*dosePl*
MCT (ns)	1.638	1.607	1.601	2.179	2.081	1.847
P_{leak} (μW)	448.0	441.4	441.4	2915.5	2922.3	2922.3
Runtime (s)	–	108	40	–	924	149

Figure 6.3 shows four slack profiles of $AES65$ (used in Section 5.2.4), including (i) the original design (*Orig*), (ii) the design after dose map optimization on poly layer for improved timing with dose correction range $\pm 5\%$, smoothness bound $B = 2$, and rectangular grid $5 \times 5 \mu m^2$ (*DMOpt*), (iii) the design after further placement optimization (*dosePl*), and (iv) the design when all the gates in the top 10,000 critical paths are given the maximum possible dose of +5% relative to the original dose (*Bias*). The purpose of applying the maximum possible exposure dose to timing-critical cells is to assess the optimization headroom left after *DMopt*. From Figure 6.3, the worst slack of the original design is first improved by dose map optimization process, and then further improved by the proposed placement optimization (*dosePl*). However, it is difficult for the dose map and placement optimization to improve the slacks for all the paths on the "hill" around the critical slack value of $0ns$. Besides, as shown by Table 6.2, though there is seemingly headroom left between the optimized design and the "optimal" (*Bias*) design, it is impossible to reach the "optimal" design without dramatically increasing the total leakage power. We have also tried to follow the dose map-specific placement ECO with a second dose map optimization, i.e., applying the *DMopt* and *dosePl* optimizations in alternation. However, this did not result in any further improvement.

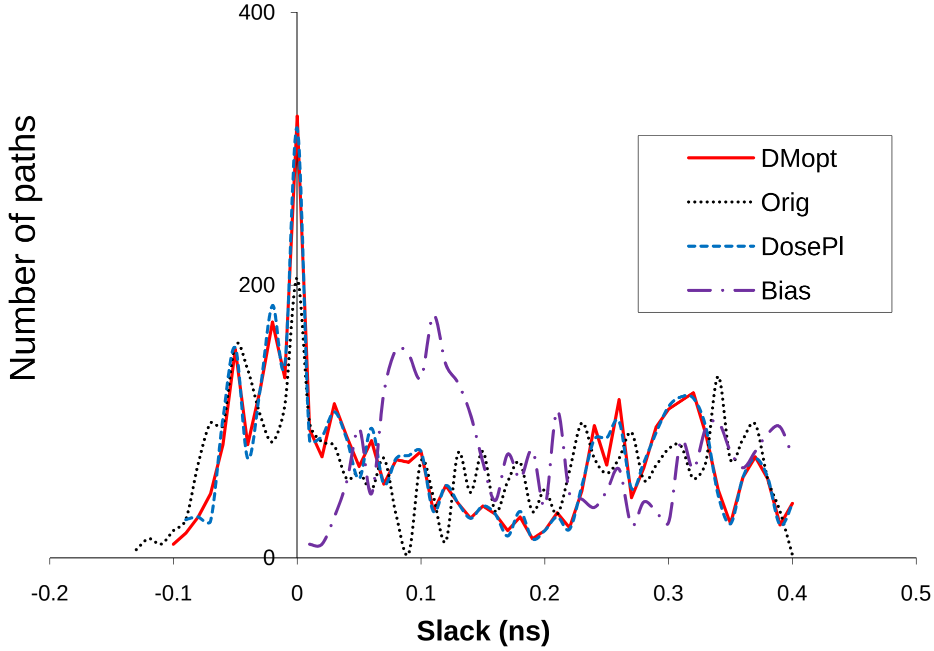


Figure 6.3: Slack profiles of $AES65$ (used in Section 5.2.4) before *DMopt*, after *DMopt*, after *dosePl*, and after biasing where all of the gates in the top 10,000 critical paths receive maximum possible exposure dose (+5%).

Table 6.2: Delay and leakage values of $AES65$ when dose change $d^P_{i,j}$ is swept from 0% to -5% and from 0% to $+5\%$ on poly layer.

Dose change	$d^P_{i,j}$ = 0	$d^P_{i,j}$ = -1	$d^P_{i,j}$ = -2	$d^P_{i,j}$ = -3	$d^P_{i,j}$ = -4	$d^P_{i,j}$ = -5
MCT (ns)	1.638	1.677	1.715	1.750	1.786	1.824
imp. (%)	–	-2.38	-4.70	-6.84	-9.04	-11.36
P_{leak} (μW)	448.0	397.7	356.9	324.8	299.9	279.6
imp. (%)	–	11.23	20.33	27.50	33.06	37.59
Dose change	$d^P_{i,j}$ = 0	$d^P_{i,j}$ = +1	$d^P_{i,j}$ = +2	$d^P_{i,j}$ = +3	$d^P_{i,j}$ = +4	$d^P_{i,j}$ = +5
MCT (ns)	1.638	1.601	1.557	1.517	1.474	1.427
imp. (%)	–	2.26	4.95	7.39	10.01	12.88
P_{leak} (μW)	448.0	513.4	600.4	722.2	893.5	1142.2
imp. (%)	–	14.60	-34.02	-61.21	-99.44	-154.96

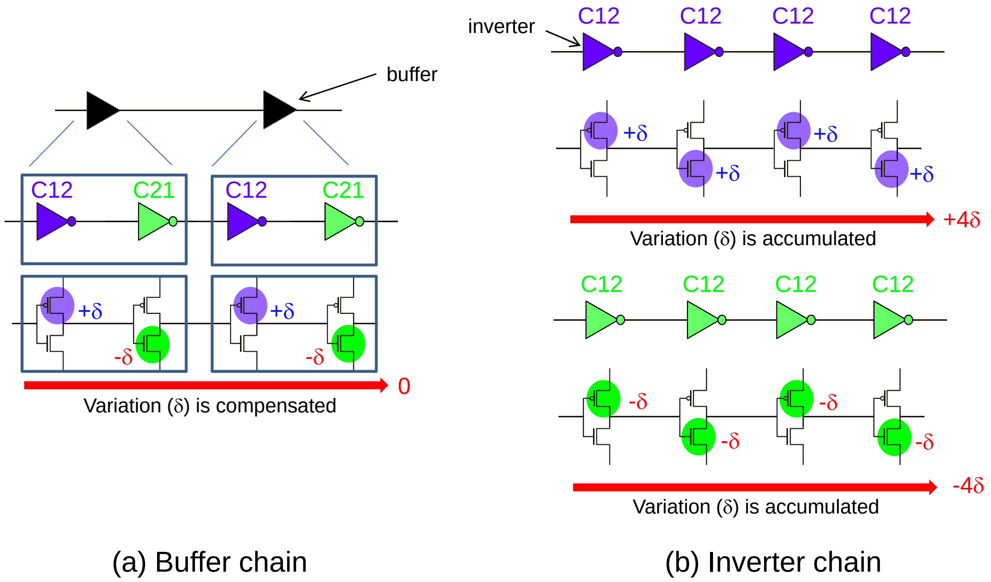
6.2 Bimodality-Aware Mask Assignment and Detailed Placement for Double Patterning Lithography

In Section 4.3.3, we have assessed the potential impact of bimodal CD distribution in DPL on timing analysis and guardbanding. Bimodal CD distribution and loss of spatial correlation between differently colored (exposure) cells have far-reaching impacts on circuit properties. We observe that the traditional 'unimodal' characterization and analysis framework may not be viable for DPL. For example, experimental analyses demonstrate that different mask layouts can result in 20% or more change in delays of timing paths. Given the implications discussed in Section 4.3.3, we propose new bimodal-aware optimization methods to improve timing yield of standard cell-based designs that are manufactured using DPL. We present two optimization techniques:

- an ILP-based maximization of 'alternate' mask coloring of instances in timing-critical paths to minimize harmful covariance and performance variation; and
- a dynamic programming-based detailed placement algorithm that solves mask coloring conflicts and can be used to ensure "double patterning correctness" after placement or even after detailed routing while minimizing the displacement of timing-critical cells with manageable ECO impact.

6.2.1 A New Metric: Coloring Sequence Cost

We now propose a methodology for optimal coloring of timing paths. We begin by quantifying "balance" in the coloring of timing paths. The impact of DPL's bimodal CD distribution on cell delay varies according to the number of poly lines in a cell, the topology of the circuit, the assigned color for each poly gate, and the specific transistors that are activated during signal transitions. A path consisting of only buffers (each buffer comprising cascaded two inverters) experiences small impact from the bimodal CD distribution, since the CD change of the first inverter can be compensated by that of the second inverter as shown in Figure 6.4(a). On the other hand, a path consisting of only one-stage inverters can have two different worst-case delay values when all inverters are assigned the same color: the inverters will have either all positive CD changes or all negative CD changes, so that there is no compensation as shown in Figure 6.4(b).


buffer
C12
C21
C12
C21
+δ
-δ
+δ
-δ
0
Variation (δ) is compensated
inverter
C12
C12
C12
C12
+δ
+δ
+δ
+δ
+4δ
Variation (δ) is accumulated
C12
C12
C12
C12
-δ
-δ
-δ
-δ
-4δ
Variation (δ) is accumulated

(a) Buffer chain

(b) Inverter chain

Figure 6.4: Delay variations of timing paths due to bimodal CD distribution in DPL. (a) In a path consisting of only buffers, variation of each inverter stage is compensated. (b) In a path consisting of only inverters, variation of each inverter stage accumulates.

For cells that are more complex than inverters or buffers, the impact on delay of bimodal CD distribution is complicated. Table 6.3 shows circuit simulation results with SPICE for the NAND2 shown in Figure 6.5(b), with respect to switching input (A1 or A2), switching direction (rise and fall), and bimodal CD variation. Comparing the second and fifth rows, we observe that the CD of A2 has negligible impact on the rise delay due to the transition of A1 ($t_{plh,A1}$). Similarly, comparing the second and sixth rows, the CD of A1 does not affect rise delay due to the transition of A2 ($t_{plh,A2}$). We can conclude that rise delay of the NAND2 depends only on the CD of the transitioning PMOS transistors MP1 or MP2.

However, both fall delays – due to the transition of A1 ($t_{phl,A1}$) or of A2 ($t_{phl,A2}$) – are affected by both CD values of the NMOS transistors MN1 and MN2. For the series transistors MN1 and MN2, the coloring of MN2 affects cell delay triggered by MN1, and vice versa. We observe that fall delay values in the fifth and sixth rows are in between the values in the second and third rows, which are respectively the slowest and fastest delays when both MN1 and MN2 have smaller or larger CD. Therefore, the

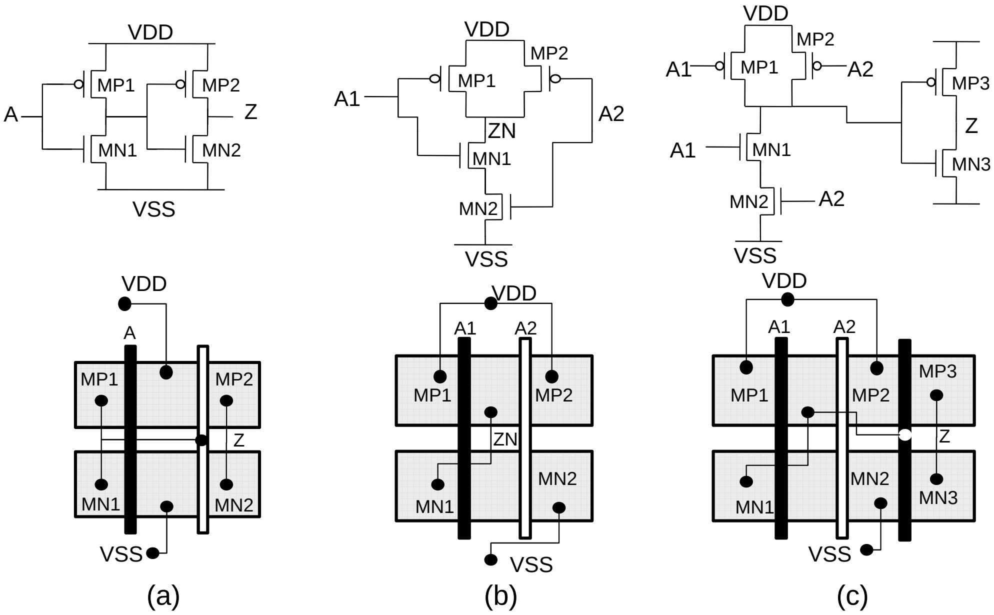


Figure 6.5: Schematic and layout of C_{12}-type (a) BUF, (b) NAND2, and (c) AND2 cells.

average CD change of MN1 and MN2 can be used to represent the fall delay change.[2]

Coloring sequence cost (*CSC*) for a timing arc. To account for the different impact of bimodal CD changes on the cell delay, we define a *coloring sequence cost* (*CSC*) that scores how poly lines are colored alternately from input to output (i.e., we use *CSC* as a quantitative measure of the alternate coloring of timing paths). The smaller the *CSC* value, the more alternate coloring is in the signal propagation path in a cell, which implies smaller delay variation due to the bimodal CD distribution. For a single NMOS or PMOS device, we assign *CSC* value of either 1 or -1 according to the color of the transistor. *CSC* for a network of transistors is calculated as follows.

- Parallel transistors: 1 or -1 of the transitioning input poly
- Series transistors: average *CSC* of all series transistors
- Fingered transistors: average *CSC* of all fingered transistors

[2] Using the average of MN1 and MN2 may not be accurate, since delay impacts of MN1 and MN2 are different due to different charge-sharing effects for MN1 and MN2. We can extend the methods presented here to obtain and use accurate delay values via cell characterization with different CD combinations.

Table 6.3: Delay changes due to the CD changes of the transitioning input gates of a 2-input NAND gate which has two poly lines corresponding to two input pins A1 and A2.

A1 CD (nm)	A2 CD (nm)	$t_{phl,A1}$ (s)	$t_{plh,A1}$ (s)	$t_{phl,A2}$ (s)	$t_{plh,A2}$ (s)
51	51	4.979e-11	9.730e-11	5.465e-11	1.131e-10
49	49	4.823e-11	8.825e-11	5.148e-11	1.021e-10
50	50	4.830e-11	9.290e-11	5.308e-11	1.076e-10
51	49	4.905e-11	9.726e-11	5.232e-11	1.022e-10
49	51	4.889e-11	8.828e-11	5.379e-11	1.130e-10

- Cascaded transistors: sum of the *CSC* values of all stages

Based on the above rules, examples of the *CSC* calculation for buffer (BUF), 2-input NAND (NAND2), and 2-input AND (AND2) cells are shown below. We calculate *CSC* for each timing arc for each coloring version of a given master cell. For example, $CSC_{C_{12},rise_{A1}}$ denotes the coloring sequence cost for rise delay due to a transitioning input A of a coloring version C_{12}. We use ‘1’ and ‘-1’ for the *CSC* of the transistors formed by black and white poly lines in Figure 6.5, respectively. For C_{21} cells, the colors of poly lines are inverted.

• BUF: There are two poly lines which are cascaded inverters (INV followed by INV) as shown in Figure 6.5(a).

- $CSC_{C_{12},rise_A}$ (= MN1: on, MP2: on) = 1 + (-1) = 0
- $CSC_{C_{12},fall_A}$ (= MP1: on, MN2: on) = 1 + (-1) = 0
- $CSC_{C_{21},rise_A}$ (= MN1: on, MP2: on) = -1 + 1 = 0
- $CSC_{C_{21},fall_A}$ (= MP1: on, MN2: on) = -1 + 1 = 0

• NAND2: There are two poly lines and one logic stage as shown in Figure 6.5(b).

- $CSC_{C_{12},rise_{A1}}$ (= MP1: on) = 1

- $CSC_{C_{12},fall_{A1}}$ (= MN1: on, MN2: on) = (1 + (-1)) / 2 = 0
- $CSC_{C_{12},rise_{A2}}$ (= MP2: on) = -1
- $CSC_{C_{12},fall_{A2}}$ (= MN1: on, MN2: on) = (1 + (-1)) / 2 = 0
- $CSC_{C_{21},rise_{A1}}$ (= MP1: on) = -1
- $CSC_{C_{21},fall_{A1}}$ (= MN1: on, MN2: on) = (-1 + 1) / 2 = 0
- $CSC_{C_{21},rise_{A2}}$ (= MP2: on) = 1
- $CSC_{C_{21},fall_{A2}}$ (= MN1: on, MN2: on) = (-1 + 1) / 2 = 0

• AND2: AND2 consists of a NAND2 and an INV as shown in Figure 6.5(c). The *CSC* values of NAND2 and INV are added. We show only example *CSC* calculations for the rise and fall delay of the C_{12}-type AND2 by A1.

- $CSC_{C_{12},rise_{A1}}$ (= (MN1: on, MN2: on) + MP3: on)
  = (1 + (-1)) / 2 + 1 = 1
- $CSC_{C_{12},fall_{A1}}$ (= MP1: on + MN3: on) = 1 + 1 = 2

In our experiments, we analyze the schematic and layout of all cell masters used in our testcases, and calculate the *CSC* value for each timing arc of each coloring version.

Coloring sequence cost for a path (*CSCP*). Given the *CSC* values of all timing arcs, we define the coloring sequence cost of a path (*CSCP*) as a weighted sum of the *CSC* values of its timing arcs. Since the impact of *CSC* is relative to each timing arc delay, the weight is given by the delay value t_l of the timing arc l, so that

$$CSCP_i = \sum_{l \in i} CSC_l \cdot t_l \tag{6.2}$$

To verify the correlation between *CSCP* and the actual delay variation, we extract a timing path with 22 stages of logic cells from an actual design.[3] We assign a color to

[3] The timing path consists of two buffers, three inverters, three 2-input NOR, two 2-input OR, nine 2-input NAND, one 3-input NAND, one 3-input OR-AND, and one 3-input AND-OR gates.

each cell in the path randomly, then calculate *CSCP* for each path coloring, and measure the path delay using two bimodal-aware timing libraries, *G1L-G2S* and *G1S-G2L*.[4]

Figure 6.6 shows the correlation between the calculated *CSCP* and the delay difference for 1,300 random colorings of the timing path. We observe that *CSCP* and timing variation have a strong positive correlation, i.e., correlation coefficient is 0.902, and rank correlation is 0.900.

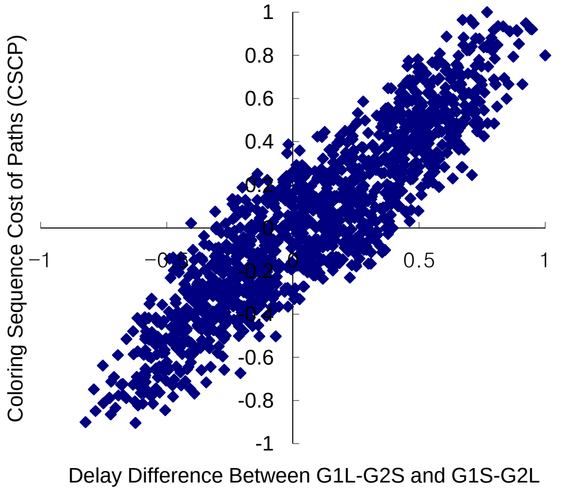


Figure 6.6: Correlation between *CSCP* and the delay difference between the two bimodal-aware timing libraries *G1L-G2S* and *G1S-G2L*, for 1,300 different colorings of the timing path.

6.2.2 Optimal Color Assignment: OPT-COLOR

Due to the high correlation between *CSCP* and delay variation, our approach is to minimize delay variation by minimizing *CSCP* of timing-critical paths.

Optimal timing path coloring problem:

- Given: a set P of timing-critical paths.
- Assign coloring of each cell in the timing paths so as to minimize $Max_{i \in P} \mid CSCP_i \mid$, where $CSCP_i$ is the coloring sequence cost of path i.

[4] Suppose each group of poly lines in a cell has CD value either of CD1 or of CD2, according to the color of the poly. *G1L-G2S* (*G1S-G2L*) represents the case that CD1 (CD2) is larger than CD2 (CD1).

For the top-k timing-critical paths in a design, this can be formulated as an integer linear program (ILP) using an indicator variable M for the maximum magnitude of any *CSCP*, and binary variables x_j and y_j to capture the color of a cell j.

Minimization of maximum *CSCP*:

- **Objective:** Minimize M
- **Subject to:**

$$\begin{aligned} M \geq CSCP_i, & \qquad 1 \leq i \leq k \\ M \geq -CSCP_i, & \qquad 1 \leq i \leq k \\ CSCP_i = & \ \textstyle\sum_{l \in i} CSC_{C_{12},l}(j) \cdot x_j + CSC_{C_{21},l}(j) \cdot y_j \\ x_j + y_j = 1, & \qquad x_j \in \{0,1\}, \quad y_j \in \{0,1\} \end{aligned}$$

where P is a set of k timing-critical paths, and a path i is a set of timing arcs. *CSC*$_{C_{12},l}(j)$ and *CSC*$_{C_{21},l}(j)$ are the two different *CSC* values of timing arc l of cell j with respect to the two coloring versions C_{12} and C_{21} of the cell.

An ILP solver (*ILOG CPLEX v10.110* [10]) is used to solve this problem, and returns the optimal color values for cells in the top-k timing paths. As shown in Table 6.7, ILP runtimes are reasonable and scale well – e.g., 5.23 sec for 1,000 timing-critical paths.

6.2.3 Coloring Conflict Removal: DPL-CORR

Coloring conflict denotes the case that two adjacent lines with minimum spacing are assigned to a single mask. In DPL, two adjacent lines with minimum spacing must be assigned to different masks, since they cannot be printed by a single exposure. The proposed timing optimization by alternate color assignment within timing-critical paths (OPT-COLOR), presented in Section 6.2.2, can introduce coloring conflicts between neighbors in a row for which colors have already been determined.

Given a coloring solution that maximizes the alternate coloring of timing-critical paths, we solve the resulting coloring conflicts by placement perturbation within available whitespace using a dynamic programming (DP) formulation [83]. In other words, we exploit the whitespace available in a standard-cell row to solve any coloring conflicts that result from the OPT-COLOR.

We further note that the placement perturbation approach as a post-processing step after cell coloring may not converge to complete conflict removal in designs with high utilization of layout area. To counter this, we introduce a recoloring approach which recolors the cells during whitespace optimization to remove additional coloring conflicts. We make sure that this recoloring introduces minimum change to the colors of the cells determined from OPT-COLOR. We describe these dynamic programming formulations in the rest of this section.

Dynamic programming formulation for DPL-CORR (SHIFT). We use the following notation.

- l_j^{PS} (r_j^{PS}) denotes the space between the leftmost (rightmost) poly and the cell outline of a cell j.
- l_j^{PC} (r_j^{PC}) represents the color of the leftmost (rightmost) poly for cell j.
- The sites in a standard-cell row are indexed from left to right. A cell occupies multiple placement sites in a standard-cell row. s_j denotes the leftmost placement site index for cell j.
- x_j denotes the left x-coordinate of cell j. If the unit for x_j is equal to the *placement site width*, then x_j = s_j.
- w_j represents the width of cell j.
- δ_j denotes the displacement of cell j from its original left x-coordinate.
- Res_{min} denotes the minimum resolution of a traditional single-exposure lithography.

Figure 6.7 illustrates this notation for two adjacent cells a and $a-1$ in a standard-cell row. We consider only the boundary poly lines in the cells (i.e., those with extremal x-coordinates), as the internal poly lines in a cell are assumed to have been colored alternately and therefore have no bearing on the neighboring cells.

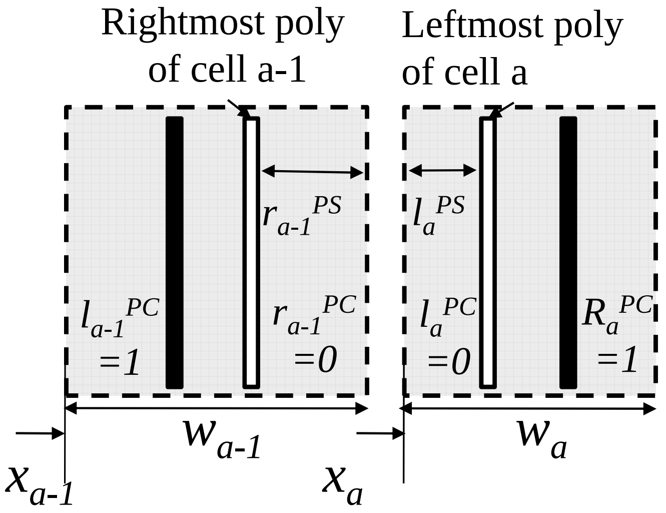


Figure 6.7: Variables used in the DP problem formulation.

Cells can be shifted in multiples of the *placement site width*, which is the finest positional granularity in the standard-cell row. For a given cell a, we formulate the minimum-perturbation placement problem for removing coloring conflicts as follows.

- **Objective:** Minimize $\sum \mid \delta_i \mid$
- **Subject to:** $x_a + \delta_a - x_{a-1} - \delta_{a-1} - w_{a-1} + l_a^{PS} + r_{a-1}^{PS} \geq Res_{min}$ whenever l_a^{PC} and r_{a-1}^{PC} are equal.

We solve this problem via a DP recurrence. The cost function for placing a cell a at placement site b is as follows.

SHIFT:

$$Cost(a,b) = \lambda_a \mid s_a - b \mid +$$
$$Min_{i=x_{a-1}-SRCH}^{x_{a-1}+SRCH}\{Cost(a-1,i) + HCost(a,b,a-1,i)\}$$
$$Cost(1,b) = \lambda_1 \mid s_1 - b \mid$$

λ_a (= $e^{-\alpha \cdot slack_a}$) defines the weight of cell a according to its timing criticality using its slack value $slack_a$. This weight determines the relative importance of preserving the initial placement as opposed to displacing the cell to placement site b. The value of α

is chosen such that it allows positive-slack cells to move while restricting the movement of timing-critical cells. *SRCH* is the range over which the cell may be displaced.[5]

HCost denotes the cost of displacing cell a to site b, relative to a's immediate left neighbor in the row and depending on the distance between the corresponding boundary poly lines and their colors. The method for computing the *HCost* is shown in Figure 6.8. When cell a is the leftmost cell in a cell row, *HCost* for placing cell a to placement site b is zero (Line 01). Otherwise (Line 02), the *HCost* is calculated considering the locations of cell a and its left-neighbor cell $a-1$. The *spacing* between the leftmost poly in cell a and the rightmost poly in cell $a-1$ is calculated (Line 03). When the *spacing* between the two polys is less than a given minimum resolution Res_{min} and the two polys have the same color, *HCost* is inifinity (Line 04). Otherwise, *HCost* is zero (Line 05).

SHIFT computes the cost of displacement of cell a and the minimum of the sum of displacement cost of its left neighbor ($a-1$) and the corresponding *HCost* over a set of displacements in the search range (*SRCH*).

We take care of the flipped orientation of cells in the calculation of *HCost*. For instance, if cell a is placed in flip-south (FS) or flip-north (FN) orientation – that is, if cell a is mirrored about the y-axis – then l_a^{PS} corresponds to r_a^{PS} and vice-versa. Hence, r_a^{PC} is used in cost calculation instead of l_a^{PC}.

SHIFT with a different objective (MINMAX). We have also applied another objective for whitespace management to remove coloring conflicts. The objective seeks to minimize the maximum displacement of a cell in a row rather than the sum of displacement costs of all cells. We consider this objective since the previous one can have high standard deviation of the displacements of individual cells from the mean displacement. The objective is defined as follows.

- **Objective:** Minimize $\{Max \mid \delta_i \mid\}$
- **Subject to:** $x_a + \delta_a - x_{a-1} - \delta_{a-1} - w_{a-1} + l_a^{PS} + r_{a-1}^{PS} \geq Res_{min}$ whenever l_a^{PC} and r_{a-1}^{PC} are equal.

[5] As noted above, the displacement is made in multiples of site width, and so the runtime for the proposed algorithm is contained using this *SRCH* parameter. High-utilization designs may require a large range of displacements to utilize the whitespace available in selective pockets, with increased runtime due to a larger *SRCH* value.

```
*HCost*($a$,$b$,$a-1$,$i$) of cell $a$
Input:
  Displacement of cell $a$ corresponding to site index $b$: $\delta_b$
  Displacement of cell $a-1$ corresponding to site index $i$: $\delta_i$
  $x$-coordinate and width of cell $a$: $x_a$ and $w_a$
  $x$-coordinate and width of cell $a-1$: $x_{a-1}$ and $w_{a-1}$
Output:
  Value of *HCost*
Algorithm:
  Case $a = 1$: *HCost*$(1, b) = 0$
  Case $a > 1$
  /* Calculate the spacing between cell $a$ and $a-1$
     according to new $x$-coordinates */
  $spacing = x_a + \delta_b - x_{a-1} - \delta_i - w_{a-1} + l_a^{PS} + r_{a-1}^{PS}$
  if $(spacing - Res_{min} < 0)$ && $(l_a^{PC} == r_{a-1}^{PC})$
      *HCost*$(a, b, a-1, i) = \infty$
  else
      *HCost*$(a, b, a-1, i) = 0$
```

Figure 6.8: *HCost* algorithm for coloring-conflict removal.

This objective leads to a minor modification in the DP formulation stated in normal **SHIFT**.

DP with recoloring (SHIFT+RECOLOR). To maintain the timing goals of the design, the timing-critical cells need to be locked in their position while performing detailed placement perturbation. This can lead to very little or no reduction in the number of coloring conflicts as these timing-critical cells block the movement of non-timing critical cells. The lack of whitespace available for perturbation in very high-utilization designs may also lead to non-convergence of the above-stated DP approach. The only alternate available to remove coloring conflicts is the recoloring of the cells which are not fixed by **SHIFT**. Therefore, we need to include the recoloring of the cells into the proposed DP recurrence and assign a cost for recoloring the cell in the DP formulation. We believe that removing coloring conflicts is far more important and should be achieved (even if there is slight degradation in timing) since that is needed to ensure printability of the layout patterns. Therefore, we allow recoloring of fixed color cells, but they are given high weights as compared to other cells (as done for timing-critical cells in **SHIFT**). We include the color of the cell as a new dimension for formulating DP for this approach. Without loss of generality, we assume that cell a has original color C_{12} and is recolored to C_{21}. Therefore, the cost of placing cell a at placement site b when recoloring of cells is allowed is as follows.

SHIFT+RECOLOR:

$$\begin{aligned}
Cost(a,b,C_{12}) &= \lambda_a \mid s_a - b \mid + Min[\\
&\quad \{Min_{i=x_{a-1}-SRCH}^{x_{a-1}+SRCH}\{Cost(a-1,i,C_{12})\\
&\qquad + HCost(a,b,C_{12},a-1,i,C_{12})\}\},\\
&\quad \{Min_{i=x_{a-1}-SRCH}^{x_{a-1}+SRCH}\{Cost(a-1,i,C_{21})\\
&\qquad + HCost(a,b,C_{12},a-1,i,C_{21})\}\}]
\end{aligned}$$

$$\begin{aligned}
Cost(a,b,C_{21}) &= \lambda_a \mid s_a - b \mid + \lambda_c^a + Min[\\
&\quad \{Min_{i=x_{a-1}-SRCH}^{x_{a-1}+SRCH}\{Cost(a-1,i,C_{12})\\
&\qquad + HCost(a,b,C_{21},a-1,i,C_{12})\}\},\\
&\quad \{Min_{i=x_{a-1}-SRCH}^{x_{a-1}+SRCH}\{Cost(a-1,i,C_{21})\\
&\qquad + HCost(a,b,C_{21},a-1,i,C_{21})\}\}]\\
Cost(1,b,C_{12}) &= \lambda_1 \mid s_1 - b \mid\\
Cost(1,b,C_{21}) &= \lambda_1 \mid s_1 - b \mid + \lambda_c^1
\end{aligned}$$

$Cost(a,b,C_{12})$ denotes the minimum cost of placing the cell a with original color C_{12} at placement site b when the color of cell $a-1$ can be either C_{12} or C_{21}. Similarly, $Cost(a,b,C_{21})$ denotes the minimum cost of placing the recolored cell a with color C_{21} at placement site b when the color of cell $a-1$ can be either C_{12} or C_{21}. The recoloring weight of cell a is defined as λ_c^a (= $e^{-\beta slack_a}$) where β can take different values to assign different recoloring weights to the cells. Fixed-color cells have high recoloring weight for high values of β because they also happen to be timing-critical cells with negative slack values. As demonstrated below, this **SHIFT+RECOLOR** DP formulation can achieve a conflict-free design by combining (i) displacement of cells ("**SHIFT**") to add spaces between coloring-conflicting cells with (ii) inverting the given coloring of cells ("**RECOLOR**") to remove coloring conflicts without additional spaces.

Applicable design stages. The proposed approach is aware of timing-critical cells and seeks to minimize perturbation of these cells to preserve timing goals of the design. Table 6.4 summarizes applicable design stages of the proposed methodology; △ and ◯ represent applicable design stages, while × represents inapplicable design stages. The alternate coloring (**OPT-COLOR**) and conflict removal (**DPL-CORR**) optimizations can be used either separately or together. OPT-COLOR can be performed at every timing optimization stage in the design implementation flow, even before placement, while DPL-CORR is applicable after placement. However, timing can change significantly at every design optimization stage, so that timing-critical paths can continually change, and new coloring conflicts can occur whenever placement locations or master cells of cell instances change. Consequently, OPT-COLOR may optimize non-timing critical paths if used at early design stages, and DPL-CORR may be performed unnecessarily if used before any significant timing optimizations, while failing to guarantee DPL-correctness at signoff. Hence, we suggest using the proposed methodology at near-final timing signoff stages – in particular, after detailed routing (denoted by ◯ in Table 6.4), when timing improvement has 'saturated' and most timing-critical paths have been fixed or otherwise clearly comprehended by the design team.

Table 6.4: Applicable design stages.

Design Stage	OPT-COLOR	DPL-CORR
Pre-placement	△	×
Post-placement	△	△
Post-clock-synthesis	△	△
Post-routing	◯	◯

6.2.4 Experimental Setup

Library preparation. We use *G1L-G2S* and *G1S-G2L* as the names of bimodal-aware timing libraries corresponding to the scenarios $MAX_{G1} \geq MAX_{G2}$ and $MAX_{G1} \leq MAX_{G2}$, respectively. Each scenario can further be split based on the CD mean difference values between the two groups.

We choose the most commonly used 36 standard cells from *Nangate* $45nm$ *Open Cell Library* [14]. We create two coloring versions for each standard cell, e.g., NAND2_C_{12} and NAND2_C_{21} for a $NAND2$ cell. We characterize delay of all coloring versions of cells using Predictive Technology Model (PTM) [17], with respect to the two scenarios *G1L-G2S* and *G1S-G2L* and the three cases of CD mean difference equal to 2, 4 and $6nm$. Table 6.5 summarizes timing libraries we generate for bimodal-aware timing analysis. Unimodal CD values corresponding to the bimodal CD values follow the calculation used in [64]. With the unimodal timing models, we require one timing signoff, while with the bimodal timing models, we need two timing signoffs for each *G1L-G2S* or *G1S-G2L* case.

Table 6.5: Bimodal-aware timing libraries.

CD Mean Diff. (→)	$2nm$		$4nm$		$6nm$	
	CD (nm) of group $G1$ and $G2$					
Corner Name (↓)	*G1*	*G2*	*G1*	*G2*	*G1*	*G2*
Unimodal	53.61		56.32		59.22	
G1L-G2S	53	51	54	50	55	49
G1S-G2L	51	53	50	54	49	55

We define the minimum spacing between same-colored poly lines (Res_{min}) to be $330nm$, which is calculated by subtracting the poly width ($50nm$) from twice the defined poly pitch ($2 \times 190nm$) in the *Nangate* $45nm$ library. Separately, all 72 (36 × 2) standard cells are analyzed and *CSC* values are recorded.

Testcase preparation. We implement four open-source cores, AES and $JPEG$, obtained from the open-source site $opencores.org$ [15], and two sub-blocks of *OpenSparc*

T1 design, i.e., LSU (load and store unit) and EXU (execution unit), obtained from the *Sun OpenSPARC Projects* site [19]. We synthesize the cores using *Cadence RTL Compiler v5.2* [6] with the original (non-bimodal) timing library which does not have coloring information and which assumes worst-case CD values of $50nm$ for all transistors. We use *Cadence SOC Encounter v7.2* [7] to place and route with three different placement utilizations (70%, 80% and 90%), to compare the difficulty of coloring-conflict removal.

We then assign color C_{12} or C_{21} to all cell instances by replacing the original master cell names, e.g., NAND2_C_{12} or NAND2_C_{21} for NAND2. For this initial coloring assignment, the only objective is to not create coloring conflicts, which is the only constraint for the traditional DPL pattern decomposition. We first assign a color to the leftmost cell in each cell row and assign a color of the next cell so as to not create a coloring conflict with the first cell, and then iterate this assignment method to the end of the cell row. For each initially colored design, we extract RC parasitics from *SOC Encounter v7.2* [7] and then perform timing analysis with *Synopsys PrimeTime vB-2008.12-SP2* [25].

Table 6.6 summarizes design and timing information of the used testcases when the $2nm$ CD mean difference library is used. The number (e.g., 70) in the name of testcases (e.g. $AES70$) denotes the initial placement utilization (e.g., 70%). WNS and TNS respectively represent the worst negative slack of the design and the total negative slack (which is the sum of all negative slacks) over all the end points of timing paths. WNS can be regarded as the feasibility of timing closure at the given clock cycle time, and TNS can be regarded as the required effort to fix all timing violations of the design. Timing with the original (single CD distribution) timing library is met at the given clock cycle times. However, due to the bimodal CD distribution, timing of the double patterning-applied designs is significantly degraded. It must be noted that the unimodal timing analysis is significantly more pessimistic than the bimodal-aware timing analysis. We also observe that the use of the bimodal-aware timing library can by itself directly improve timing significantly, due to the ‘intrinsic’ alternate coloring within a cell, as we observe in the fall delay of a NAND2 in which CD variation of MN1 is compensated by opposite CD variation of MN2 (refer to Figure 6.5).

Table 6.6: Testcase information.

	#instance		Area	Util.	Unimodal		Bimodal	
	#-C_{12}	#-C_{21}	(μm^2)	(%)	WNS (ns)	TNS (ns)	WNS (ns)	TNS (ns)
$AES70$	21350	4676	44848	69.10	-0.428	-73.8	-0.190	-14.9
$AES80$	19300	7336	38735	81.26	-0.460	-79.6	-0.197	-19.1
$AES90$	13388	7396	29765	91.15	-0.489	-67.4	-0.239	-12.8
$JPEG70$	77807	15091	175742	68.85	-0.613	-208.3	-0.331	-23.1
$JPEG80$	66928	25742	152430	79.47	-0.641	-191.9	-0.232	-16.1
$JPEG90$	60136	32483	137571	88.31	-0.613	-192.7	-0.229	-20.6
$EXU70$	23764	5933	68066	68.71	-0.446	-173.0	-0.235	-25.1
$EXU80$	19669	9898	58705	79.40	-0.548	-130.5	-0.251	-14.6
$EXU90$	18008	10007	51663	88.37	-0.449	-149.9	-0.199	-15.4
$LSU70$	30831	4673	106385	70.11	-0.448	-89.9	-0.136	-4.8
$LSU80$	27444	7638	93154	79.97	-0.486	-108.4	-0.207	-5.0
$LSU90$	23165	11926	82909	90.20	-0.466	-120.2	-0.212	-8.5

Experimental flow. Figure 6.9 shows the proposed design optimization framework for double patterning. Major steps are in the left-hand side and output data are in the right-hand side. Solid arrows show the design flow and dashed arrows show the data flow.

Step 1: Initial design. For the initial testcase preparation, we use a traditional timing-driven design implementation flow. The design starts with RTL netlists and timing constraints, and is synthesized, placed and routed with the original (traditional) worst-case timing library from single CD distribution.

Step 2: Initial coloring. The framework performs initial coloring for double patterning, in which no coloring conflicts are allowed. The output is a design exchange format (initial_colored.def).

Step 3: Timing analysis. Based on the coloring information and bimodal-aware timing libraries, a static timer analyzes timing, and generates an ILP problem instance for top-k timing-critical paths.

Step 4: Optimal coloring (OPT-COLOR). The ILP solver finds the optimal alternate coloring solution for the selected timing paths, and at the same time, a pre-defined color is assigned to all clock buffers.[6]

Step 5: Conflict removal (DPL-CORR). DPL-CORR solves the coloring conflicts caused during *Step 4*, subject to the coloring constraints (*keep_color.list*) and timing constraints (*slack.list*). Partially disconnected nets due to the placement perturbation in DPL-CORR are ECO-routed, and a final design (*opt.def*) that does not have coloring conflicts is generated.

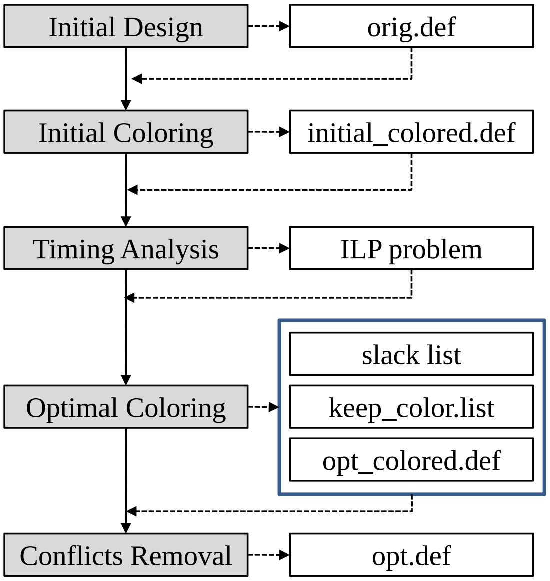


Figure 6.9: Design framework for bimodal-aware timing optimization.

The *orig.def* is not suitable for double patterning. After Step 2, *initial_colored.def* is applicable for double patterning, since this design does not have coloring conflicts.

[6] Although we suggested to use the same color for all clock buffers, this can introduce a large number of coloring conflicts when the number of clock buffers is large (indeed, in some complex SOCs with many disjoint clock trees, more than 10% of total cell instances can be clock buffers). In this case, a methodological approach of *intrinsic* alternate coloring within block buffer and inverter cells can be exploited: if clock buffers/inverters have only an even number of fingers in every transistor, then the *CSC* values of those cells become zero. Hence, we can use either C_{12} or C_{21} cells arbitrarily in the clock network without causing any bimodality-induced clock skew problem. This methodological approach would change the designs of library cells.

We use this *initial_colored.def* as the reference double patterning-applied design. The *opt_colored.def* is used to show the pure effect of the proposed optimal coloring method, since the placement locations of cells and the routing are not disturbed from the reference design. We finally compare the timing quality of the final design (*opt.def*) with the reference design (*initial_colored.def*). We note that the proposed optimization framework does not increase the given die area at all, since it perturbs the placement using existing whitespace between placed cells.

6.2.5 Experimental Results

The first experiment is to verify the quality of the alternate color assignment (OPT-COLOR), varying the number of timing-critical paths taken into account. We apply the alternate color assignment to the timing-critical paths and uniform color assignment to the clock paths, but do not apply DPL-CORR that may introduce other timing uncertainty from placement perturbation and ECO-routing. Table 6.7 shows *CSCP* and TNS reduction from the alternate color assignment on different top-k critical paths of $AES70$. Runtime is listed in Row 2. Average |*CSCP*| of the top-k critical paths (Avg. *CSCP*) of the initially colored design ("Initial Coloring") and of the alternately colored design ("Alternate Coloring") is given in Rows 3 and 7, respectively. TNS is calculated using the two worst-corner bimodal-aware timing libraries *G1L-G2S* and *G1S-G2L*. Between the two timing results, the worse one is reported. We observe that as the number of timing paths increases, the average *CSCP* of the optimized design decreases. As a result, the total negative slack decreases.

The second experiment is to verify the quality of the alternate color assignment method (OPT-COLOR) on different testcases. Tables 6.8 and 6.9 summarize the WNS and TNS reduction from the alternate color assignment for different testcases.[7] We observe that WNS is reduced by more than 150ps and TNS is improved by around 10ns (from -23.1ns to -13.2ns) in $JPEG70$, even with 2nm CD mean difference libraries.

The third experiment is to measure the performance of the proposed DPL-CORR technique. Tables 6.11 and 6.12 show the results of DP-based coloring-conflict removal

[7] For this experiment, we consider top-400 critical paths for AES testcases, top-1000 critical paths for EXU and LSU testcases, and top-2000 critical paths for $JPEG$ testcases.

Table 6.7: Average *CSCP* of the top-k critical paths and TNS (ns) reduction via the alternate color assignment (OPT-COLOR).

#Timing-critical paths (k)		100	200	300	400	1000
Runtime (s)		0.46	4.20	19.38	6.19	5.23
Initial Coloring	Avg. *CSCP*	2.019	2.010	2.016	2.012	2.000
	TNS @2nm	-14.91	-14.91	-14.91	-14.91	-14.91
	TNS @4nm	-24.38	-24.38	-24.38	-24.38	-24.38
	TNS @6nm	-36.09	-36.09	-36.09	-36.09	-36.09
Alternate Coloring	Avg. *CSCP*	1.992	1.962	1.974	1.959	1.957
	TNS @2nm	-14.43	-13.40	-12.97	-12.48	-11.91
	TNS @4nm	-23.37	-21.68	-20.51	-19.51	-17.08
	TNS @6nm	-34.48	-32.87	-31.07	-29.45	-26.29

algorithm on different testcases. The tables show the performance statistics for DP implementation on testcases with “Random” coloring and “Alternate” coloring respectively. The experiments are performed for different values of α to highlight the performance of the algorithm as the movement of timing-critical cells is restricted progressively from unrestricted movement to no movement. We also compare the performance of **SHIFT** and **MINMAX** on testcase $AES70$ with “Random” coloring in Table 6.10. The value of α is 0. The value of the sum of maximum displacements of over all rows (SOMR) is lower for **MINMAX** than that for **SHIFT**, as expected, but the improvement is not significant. On the contrary, the sum of displacements of timing-critical cells (SDTC) and the sum of displacements of non-timing critical cells (SDNTC) increases substantially for **MINMAX** as compared to the values reported for **SHIFT**. Therefore, we have chosen **SHIFT** for **SHIFT+RECOLOR** which seeks complete conflict removal.

In Tables 6.11 and 6.12, performance data are reported for DP variants, i.e., **SHIFT** and **SHIFT+RECOLOR**, respectively. The value of *SRCH* is taken as 50 and the value of β for determining recoloring weight is 30. For **SHIFT**, we report the number of conflicts after placement optimization (#Confl.), the sum of displace-

Table 6.8: WNS (ns) comparison before (Init.) and after (Alt.) alternate coloring (OPT-COLOR) for different testcases.

CD diff.	$2nm$		$4nm$		$6nm$	
Coloring	Init.	Alt.	Init.	Alt.	Init.	Alt.
$AES70$	-0.189	-0.154	-0.257	-0.192	-0.388	-0.293
$AES80$	-0.197	-0.190	-0.234	-0.220	-0.309	-0.299
$AES90$	-0.239	-0.186	-0.301	-0.250	-0.430	-0.386
$JPEG70$	-0.331	-0.173	-0.498	-0.229	-0.663	-0.391
$JPEG80$	-0.232	-0.207	-0.371	-0.337	-0.561	-0.484
$JPEG90$	-0.229	-0.210	-0.334	-0.274	-0.532	-0.393
$EXU70$	-0.235	-0.206	-0.266	-0.197	-0.348	-0.265
$EXU80$	-0.251	-0.247	-0.308	-0.276	-0.379	-0.351
$EXU90$	-0.199	-0.173	-0.255	-0.211	-0.358	-0.297
$LSU70$	-0.136	-0.057	-0.230	-0.093	-0.330	-0.169
$LSU80$	-0.207	-0.206	-0.245	-0.220	-0.307	-0.273
$LSU90$	-0.212	-0.161	-0.288	-0.251	-0.405	-0.343

ments of non-timing critical cells (SDNTC), and the sum of displacements of non-timing critical cells (SDNTC) respectively in Columns 5, 6, and 7 in Table 6.11. For **SHIFT+RECOLOR**, we report the number of conflicts after placement optimization (#Confl.), the sum of displacements of non-timing critical cells (SDNTC), the sum of displacements of non-timing critical cells (SDNTC) and the number of recolorings of fixed-color cells (FCCD), respectively, in Columns 4, 5, 6 and 7 in Table 6.12.

The sum of displacements of timing-critical (non-critical) cells, SDTC (SDNTC), decreases (increases) with increase in the value of α as the movement of timing-critical cells is restricted and movement of non-critical cells is needed to compensate for that restriction to achieve the same results. Runtime in Column 8 in Tables 6.11 and 6.12 shows linear scalability of the proposed algorithms with respect to instance count.

The first DPL-CORR algorithm **SHIFT** is able to solve all the coloring conflicts for testcases with a given random coloring at 70% and 80% placement utilizations, when α values are 0 and 20. For high-utilization testcases ($AES90$ and $JPEG90$), the

Table 6.9: TNS (ns) comparison before (Init.) and after (Alt.) alternate coloring (OPT-COLOR) for different testcases.

CD diff.	$2nm$		$4nm$		$6nm$	
Coloring	Init.	Alt.	Init.	Alt.	Init.	Alt.
AES70	-14.91	-12.48	-24.38	-19.51	-36.09	-29.45
AES80	-19.07	-16.10	-26.41	-22.65	-36.52	-32.53
AES90	-12.83	-11.17	-21.68	-18.52	-32.47	-28.09
$JPEG$70	-23.08	-13.20	-50.76	-22.34	-105.15	-49.29
$JPEG$80	-16.15	-10.68	-33.03	-15.50	-68.52	-34.27
$JPEG$90	-20.56	-12.12	-38.77	-18.49	-78.97	-40.26
EXU70	-25.11	-18.31	-42.99	-22.38	-75.46	-37.88
EXU80	-14.57	-12.04	-21.49	-15.99	-35.00	-26.72
EXU90	-15.38	-14.75	-23.69	-21.41	-39.31	-34.00
LSU70	-4.78	-0.91	-13.33	-2.78	-30.31	-7.32
LSU80	-4.98	-4.22	-10.51	-6.59	-21.04	-13.71
LSU90	-8.52	-6.94	-15.58	-10.34	-26.52	-17.95

algorithm is able to remove around 33% and 39%, respectively, of conflicts without any recoloring. Understandably, the algorithm performance suffers drastically when α is taken as ∞, since this means that all timing-critical cells are locked in their positions. The number of conflicts can be reduced by increasing the *SRCH* range for displacements since the whitespace is often not distributed evenly over the entire standard-cell row. This approach can reduce the number of conflicts but the runtime can increase substantially. It should be noted that whitespace management alone cannot guarantee complete conflict removal in high-utilization designs because the algorithm is restricted by the lack of whitespace needed for complete conflict removal.

The goal of complete conflict removal can be realized fully with the second DPL-CORR algorithm, **SHIFT+RECOLOR**. We apply this DP algorithm to the alternate coloring results since this is the case of interest from a flow and methodology standpoint. The number of conflicts reduces to 0 for all values of α, albeit with a slight penalty incurred in recoloring the fixed color cells obtained from alternate coloring results. The

Table 6.10: Performance comparison of **MINMAX** and **SHIFT** for $AES70$.

	MD	SOMR	SDTC	SDNTC
SHIFT	3.42	134.14	880.08	2835.94
MINMAX	3.42	115.90	1131.26	3401.95

sum of displacements of timing-critical cells (SDTC) and the sum of displacements of non-timing critical cells (SDNTC) are both reduced considerably with this approach compared to those from **SHIFT**, implying less effort in subsequent ECO-routing. We also observe that there is a slight degradation in timing when DP-based recoloring is applied; this is due to the fact that the colors of some fixed cells are changed. The runtime is of the order of several minutes, since the number of conflicts is smaller. In general, we observe that DP with recoloring can handle all of the testcases with ease and can achieve complete conflict removal with practical runtime.

Finally, we compare timing slack and timing-slack variation before and after coloring optimization, including the effect of the placement perturbation and ECO routing due to the conflict removal. Table 6.13 shows timing of the initial coloring and timing after coloring optimization with DPL-CORR **SHIFT**, for $JPEG70$ testcase. We use timing criticality weight α = 20. 'Worst WNS' and 'Worst TNS' represent the worse timing slack between the two corner libraries, and 'WNS diff.' and 'TNS diff.' represent the slack difference between the two corner libraries. Due to the placement perturbation and ECO routing, the worst negative slack is degraded by at most $1ps$ from the results of the OPT-COLOR only. However, we still observe up to $157ps$, $268ps$ and $271ps$ of WNS reduction, and $9.81ns$, $28.36ns$ and $55.75ns$ of TNS reduction for $2nm$, $4nm$ and $6nm$ CD mean difference in bimodal CD distribution, respectively, in the $JPEG70$ testcase. In addition, the maximum variation of the worst (total) negative slack between two corner libraries is reduced from $439ps$ ($98.58ns$) to $132ps$ ($27.23ns$) for $6nm$ CD mean difference libraries. This implies more robust timing of the design with respect to CD-distribution changes of $G1$ and $G2$.

In the case that the design is very congested, the second DPL-CORR algorithm **SHIFT+RECOLOR** can completely solve the coloring conflicts without significant timing degradation. Table 6.14 summarizes timing-slack changes due to OPT-COLOR and DPL-CORR, including ECO routing and recoloring. Again, the worse WNS and TNS values are chosen from the results of the two bimodal-aware timing analyses. We report the highest utilization cases for each design.

6.3 1-D Gridded Design for IAL

We have briefly discussed interference-assisted lithography (IAL) in Section 4.1.1. In this section, we propose a 1-D gridded (i.e., regular pitch) design that is amenable to IAL.

6.3.1 Concept of Gridded Design Rule (GDR)

Conventional random-logic design has been done for many years using random two-dimensional (2-D) shapes. 2-D shapes have edges which are placed on a “design grid” which may be ten to twenty times smaller than the feature size of a technology node. For example, 45nm logic design uses a 1nm grid. Such a very fine grid allows edges to have a very large number of spacings relative to other edges in the same layer or different layers in the design. As a result, *complex design rules* (CDRs) taking hundreds of pages in a design manual are given to the physical design team. Unfortunately, even these complex rules have not been enough and *restricted design rules* (RDRs) have been introduced to try to catch “hotspots” and other marginally manufacturable patterns.

One dimensional (1-D) *gridded design rules* (GDRs) refer to a layout style in which critical layers are drawn only with 1-D lines on a coarse grid [168] [169] [121]. Since the shapes are straight lines, they can be described with fewer variables such as width, space and line-end gap. The grid is typically one or half pitch of the layer or a related layer. Figure 6.10 shows an example of two functionally equivalent layouts with the left side drawn with 1-D GDR and the right side drawn with 2-D CDR. Three problematic areas are highlighted in the 2-D CDR case. Site 1 points out a transistor which is isolated in the x-direction from other lines; it will have a reduced process window as compared to the same poly line in the 1-D GDR. Site 2 indicates a poly line in a dense environment; it will also have a reduced process window since one side is

Table 6.11: DP-based coloring-conflict removal using **SHIFT**. SDTC and SDNTC denote sum of displacements (μm) of timing-critical cells and of non-timing critical cells, respectively. (Results for other testcases are available at [30].)

	Testcase	#Confl.	α	**SHIFT**			
				#Confl.	SDTC	SDNTC	CPU (s)
Random	$AES90$	6574	0	4394	995.22	8961.73	52.22
			20	4394	971.09	9638.51	52.05
			∞	6516	0	88.16	53.76
	$JPEG90$	27296	0	16668	4785.34	49532.05	347.23
			20	16668	4660.13	53842.2	347.73
			∞	27296	0	0	348.04
	$EXU90$	8841	0	5344	5990	12101	69.79
			20	5344	6098	12914	69.89
			∞	8841	0	0	72.67
	$LSU90$	10114	0	3275	10603	17623	84.79
			20	3275	10601	18723	84.80
			∞	10112	0	0.85	91.44
Alternate	$AES90$	158	0	42	276.26	323.76	49.98
			20	42	303.43	382.47	51.57
			∞	151	0	5.89	53.43
	$JPEG90$	2036	0	0	2523.20	4751.52	213.88
			20	0	2664.94	5568.14	213.78
			∞	2026	0	12.16	241.39
	$EXU90$	443	0	0	535.61	969.00	66.80
			20	0	580.26	1214.1	66.73
			∞	439	0	3.61	72.86
	$LSU90$	411	0	0	523.64	399.19	80.55
			20	0	605.91	530.67	80.28
			∞	403	0	11.21	90.45

Table 6.12: DP-based coloring-conflict removal using **SHIFT+RECOLOR**. SDTC and SDNTC denote sum of displacements (μm) of timing-critical cells and of non-timing critical cells, respectively. FCCD represents the number of recolored cells during DPL-CORR. (Results for other testcases are available at [30].)

	Testcase	α	**SHIFT+RECOLOR**				
			#Confl.	SDTC	SDNTC	FCCD	CPU (s)
Random	$AES90$	0	0	12.35	13.68	27	205.76
		20	0	11.59	42.18	29	209.42
		∞	0	0	4824.67	32	218.02
	$JPEG90$	0	0	64.98	71.25	152	607.88
		20	0	54.72	213.75	158	607.86
		∞	0	0	22005.04	199	678.69
	$EXU90$	0	0	909	690	11	191.02
		20	0	510	589	63	191.26
		∞	0	0	1559	138	210.74
	$LSU90$	0	0	3329	2413	0	229.77
		20	0	2994	2630	44	229.81
		∞	0	0	939	136	264.91
Alternate	$AES90$	0	0	94	78	2	145.12
		20	0	21	27	15	145.25
		∞	0	0	69	25	158.59
	$JPEG90$	0	0	594	446	86	609.3
		20	0	245	334	298	609.05
		∞	0	0	373	689	693.12
	$EXU90$	0	0	211	99	47	193.54
		20	0	139	99	78	192.75
		∞	0	0	264	148	210.61
	$LSU90$	0	0	289	136	44	231.18
		20	0	193	124	90	231.48
		∞	0	0	124	169	266.11

Table 6.13: Comparisons of WNS and TNS, before and after OPT-COLOR and DPL-CORR **SHIFT** optimizations.

Stage	Timing slack	Mean CD difference		
		$2nm$	$4nm$	$6nm$
Initial	WNS (ps) w/ $G1L - G2S$	-331	-498	-663
Coloring	WNS (ps) w/ $G1S - G2L$	-122	-155	-224
	Worst WNS (ps)	-331	-498	-663
	WNS diff. (ps)	209	343	439
	TNS (ns) w/ $G1L - G2S$	-23.08	-50.76	-105.15
	TNS (ns) w/ $G1S - G2L$	-4.03	-3.44	-6.57
	Worst TNS (ns)	-23.08	-50.76	-105.15
	TNS diff. (ns)	19.05	47.32	98.58
OPT-COLOR	WNS (ps) w/ $G1L - G2S$	-165	-230	-392
+	WNS (ps) w/ $G1S - G2L$	-174	-186	-260
DPL-CORR	Worst WNS (ps)	-174	-230	-392
	WNS diff. (ps)	9	44	132
(**SHIFT**)	TNS (ns) w/ $G1L - G2S$	-13.27	-22.40	-49.40
	TNS (ns) w/ $G1S - G2L$	-9.45	-12.89	-22.17
	Worst TNS (ns)	-13.27	-22.40	-49.40
	TNS diff. (ns)	3.82	9.51	27.23

Table 6.14: Comparisons of WNS and TNS before and after OPT-COLOR and DPL-CORR **SHIFT+RECOLOR** optimizations.

Testcase	Stage	Timing	Mean CD difference		
		slack	$2nm$	$4nm$	$6nm$
AES90	Initial Coloring	WNS (ns)	-0.239	-0.301	-0.430
		TNS (ns)	-12.83	-21.68	-32.47
	OPT-COLOR Only	WNS (ns)	-0.186	-0.250	-0.386
		TNS (ns)	-11.17	-18.52	-28.09
	DPL-CORR	WNS (ns)	-0.209	-0.316	-0.398
	(**SHIFT+RECOLOR**)	TNS (ns)	-11.19	-8.58	-28.14
JPEG90	Initial Coloring	WNS (ns)	-0.229	-0.334	-0.532
		TNS (ns)	-20.56	-38.77	-78.97
	OPT-COLOR Only	WNS (ns)	-0.210	-0.274	-0.393
		TNS (ns)	-12.12	-18.49	-40.26
	DPL-CORR	WNS (ns)	-0.198	-0.231	-0.357
	(**SHIFT+RECOLOR**)	TNS (ns)	-12.45	-18.43	-29.70
EXU90	Initial Coloring	WNS (ns)	-0.199	-0.255	-0.358
		TNS (ns)	-15.38	-23.69	-39.31
	OPT-COLOR Only	WNS (ns)	-0.173	-0.211	-0.297
		TNS (ns)	-14.75	-21.41	-34.00
	DPL-CORR	WNS (ns)	-0.173	-0.212	-0.296
	(**SHIFT+RECOLOR**)	TNS (ns)	-14.44	-20.75	-32.72
LSU90	Initial Coloring	WNS (ns)	-0.212	-0.288	-0.405
		TNS (ns)	-8.52	-15.58	-26.52
	OPT-COLOR Only	WNS (ns)	-0.161	-0.251	-0.343
		TNS (ns)	-6.94	-10.34	-17.95
	DPL-CORR	WNS (ns)	-0.176	-0.252	-0.344
	(**SHIFT+RECOLOR**)	TNS (ns)	-6.57	-9.43	-16.00

relatively more isolated than the other. Finally, Site 3 shows a poly line in a congested 2-D environment; this site is susceptible to necking and bridging hotspots in addition to showing a reduced process window. As illustrated by the left side of Figure 6.10, the poly lines are on a uniform pitch with dummy lines as needed. The horizontal metal lines are also on a uniform pitch with other line segments separated by uniform gaps. Since the poly and metal lines are on perpendicular grids, the diffusion and gate contacts can be automatically located at intersections of the grid lines.

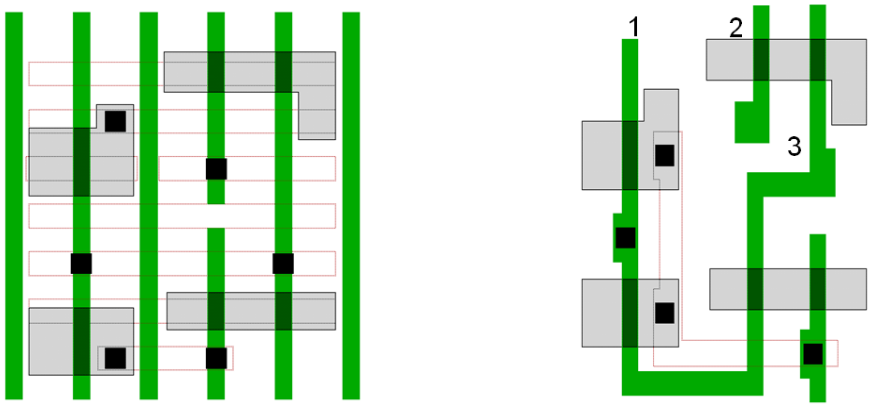


Figure 6.10: 1-D GDR layout (left) compared to 2-D CDR layout (right).

6.3.2 IAL-Friendly SRAM Bitcell Layout

The 6-T SRAM bitcell shown in Figure 6.11 is composed of six transistors, two bitlines (BL and BLb) and one wordline (WL). Both bitlines carry complementary data. A bitcell can read or write a bit of data when WL is high. Bitlines are used for both input and output terminals. The internal six transistors consist of two pass-gate (PG) transistors ($PG1$ and $PG2$), two pull-down (PD) driver transistors ($PD1$ and $PD2$) and two pull-up (PU) transistors ($PU1$ and $PU2$).

Operation of 6-T SRAM bitcell is as follows.

- **Read operation.** Before WL goes high, both bitlines are precharged to the supply voltage. When WL goes high, one of the two bitlines is discharged through a drive transistor. This creates a voltage difference between the bitlines, which is captured by a sense amplifier attached to the bitlines. Due to the RC timing delay

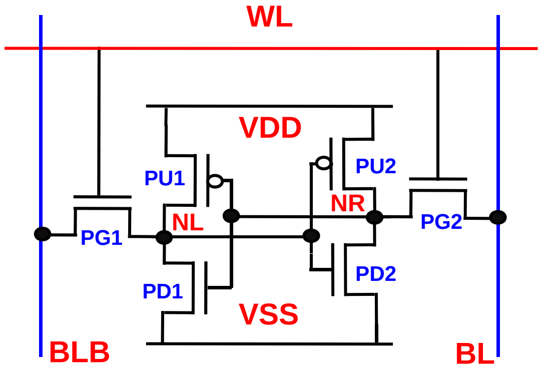


Figure 6.11: Schematic of 6-T SRAM bitcell.

from a bitline through a drive transistor to the ground, and to compensate for the mismatch in the sense amplifier, the discharging should be fast enough for the sense amplifier to realize the voltage difference within a specified time.

- **Write operation.** For write-low operation (i.e., BL is low), NR is high and WL goes to high, $PG2$ is turned on and the current flows from VDD through $PU2$ and $PG2$ to BL. At this time, the voltage of NR decreases from the current ratio of $PU2$ to $PG2$, and finally goes to ground when $PD2$ is turned on by the other internal node, NL.

Reliable operation is one of the major concerns of the bitcell design. A bitcell must provide stable read, write and data retention abilities. The two aspects of area and stability are interdependent since designing a bitcell for improved stability invariably requires a large bitcell area.

- **Read stability.** Read stability means that the data inside a bitcell must not change during the read operation. Data retention of the SRAM bitcell, both in standby mode and during a read operation, is an important functional constraint. As explained above, during a read operation discharging must be fast enough and an internal node (either NL or NR) must be low enough to turn on the drive transistor connected to that internal node. Furthermore, once the low-state node goes above the inverter's logical threshold voltage, internal data would be changed. The level of voltage increase at the internal node is decided by the ratio of the current of the drive transistor $PD1$ ($PD2$) to that of the access transistor, $PG1$ ($PG2$). The current ratio between drive and access transistors is called the *cell ratio* of the SRAM bitcell. Large cell ratio means a large drive transistor, so that the voltage on internal node NL (NR) can be kept as low as the output of the voltage divider of $PD1$ ($PD2$) and $PG1$ ($PG2$). For stable read operation, large cell ratio is preferred, but increases the bitcell area. Usually, a cell ratio of 1.5 $\sim$ 2 minimizes bitcell area while guaranteeing some level of read stability. So, if the gate lengths of $PD1$ ($PD2$) and $PG1$ ($PG2$) are the same, the gate width of $PD1$ ($PD2$) must be greater than that of $PG1$ ($PG2$).

- **Writeability.** Writeability refers to the requirement that data on bitlines must change the internal nodes of the bitcell within a specified time. This depends on the ratio of currents of the $PU1$ ($PU2$) and $PG1$ ($PG2$) transistors. To easily switch the high-state internal node to ground, the current of $PG1$ ($PG2$) should be larger than that of $PU1$ ($PU2$). This ratio is called the *pull-up ratio*, and indicates how easily the data can be changed by the low-state bitline. Pull-up ratio depends on both PMOS and NMOS currents, hence mobility must be considered during the transistor sizing. If NMOS mobility is twice of PMOS mobility, the same transistor widths of $PU1$ ($PU2$) and $PG1$ ($PG2$) result in 0.5 of pull-up ratio. So, $PU1$ ($PU2$) and $PG1$ ($PG2$) are sized as small as possible and can have the same size.

The cell ratio and pull-up ratio are determined not only by layout dimensions but also by the amount of current driven by each transistor. In a traditional 6-T SRAM bitcell, widths and lengths of transistors are all different to improve static noise margin (SNM) and to minimize bitcell area. However, in an IAL-friendly design, all six transistors must have the same gate length. Table 6.15 shows an example of transistor sizing for a 32nm bitcell scaled from an industry 90nm bitcell. Columns 4 and 5 give the length and width of transistors obtained by naive geometric scaling from 90nm to 32nm. Columns 6 and 7 give the target transistor sizes for an IAL-friendly design.

Typically, bitcell's electrical characteristics are measured by the following static metrics: Butterfly curve [158], N-curve [73], read current (I_{read}), leakage current ($I_{leakage}$),

Table 6.15: Example of 6-T SRAM bitcell device sizing.

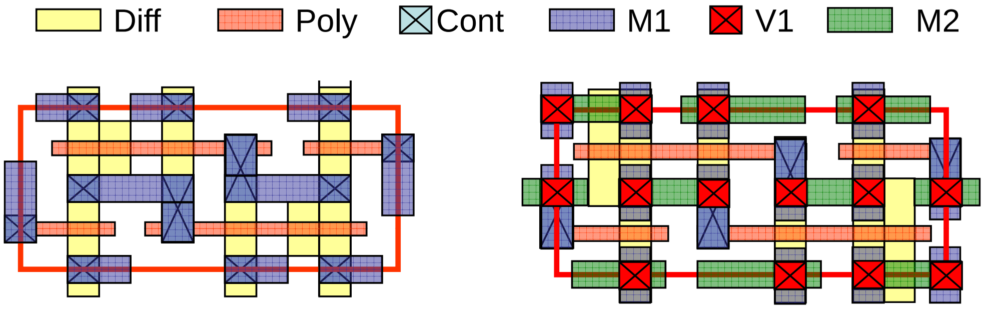
	Transistors	L (nm)	W (nm)
TSMC 90nm bitcell	Pull-Up (PU)	100	100
	Pull-Down (PD)	100	175
	Pass-Gate (PG)	115	120
32nm bitcell scaled from 90nm	Pull-Up (PU)	32	32
	Pull-Down (PD)	32	56
	Pass-Gate (PG)	115	120
32nm 1-D regular bitcell	Pull-Up (PU)	32	44
	Pull-Down (PD)	32	88
	Pass-Gate (PG)	32	44

minimum voltage to hold data (VDD_{hold}), etc. Detailed measurement methods, along with measured results of the proposed bitcells, are discussed in Section 6.3.4.

We generate an initial grid-based bitcell layout as shown in Figure 6.12(a). We assume that the feature size and minimum spacing of all layers are each equal to two drawing grids, so that all patterns can have the same linewidth. However, this grid-based bitcell is still not IAL-friendly, as there exist 2-D patterns on the M1 layer; furthermore, not all patterns are placed on an 1-D pitch, except for poly patterns. All circuit nodes are completely connected using the M1 layer.

From the initial layout, we can decompact all patterns so that they are placed on the 1-D pitch. However, the 'L' shapes on the M1 layer cannot be made IAL-friendly through decompaction. To make all geometries 1-D, the 'L' shapes must be split into two rectangles, and then one of the rectangles must be moved to another layer, e.g., M2. We use the vertical direction for M1 patterns and the horizontal direction for M2 patterns. Finally, the new 1-D regular-pitch layout is generated as shown in Figure 6.12(b).

To implement a bitcell with 32nm gate length, we must define relevant design rules. In Figure 6.12(b), we observe that the IAL-friendly bitcell layout requires at least two poly pitches in the vertical direction, and five contact pitches in the horizontal direction. The required size of the pitch in each direction is calculated as follows.



(a) Initial on-grid layout

(b) 1-D regular pitch layout

Figure 6.12: IAL-friendly bitcell layout.

- **Vertical pitch.** Vertical pitch is decided by either minimum interference lithography (IL) pitch or the minimum distance that embeds all constituent materials. Figure 6.13 shows a vertical cut-view of the proposed bitcell. Poly pitch must be greater than the sum of poly width, contact width and twice poly-to-contact spaces. Poly-to-contact space is determined by the sum of the thickness of the spacer and the strain layer. Considering process variations such as overlay and CD error, poly-to-contact space is then calculated as

$$s_{pc} \geq W_{spacer} + W_{strain\_layer} + E_{overlay} + E_{CD}$$

where s_{pc} is poly-to-contact space, W_{spacer} and W_{strain_layer} are the widths of the spacer and strain layer, and $E_{overlay}$ and E_{CD} are the amounts of overlay error and CD control error. We assume that W_{spacer} and W_{strain_layer} are $8nm$ and $10nm$ according to Verhaegen et al. [182], and $E_{overlay}$ and E_{CD} are $6.4nm$ and $2.6nm$ according to the ITRS [11]. The minimum poly-to-contact space results in $27nm$. The required poly pitch is calculated by a simple summation of poly width, contact width and twice the poly-to-contact space. We also assume that minimum poly width is $32nm$ and minimum contact width for $32nm$ technology is $45nm$, following [182]. From this, we see that the poly pitch must be greater than $131nm$ (= $27 \times 2 + 32 + 45$). Since the expected IL pitch for NA = 1.2 and λ = $193nm$ is much less than $131nm$, the vertical pitch is determined by the above calculation.

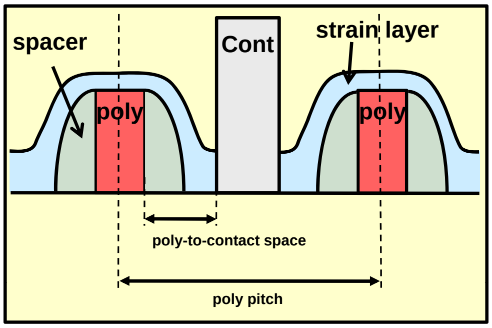


Figure 6.13: Vertical cut-view of SRAM bitcell.

- **Horizontal pitch.** In the horizontal direction, there are no constraints analogous to those we saw for the vertical pitch (strain layer, spacer, etc.).[8] Therefore, in the horizontal direction we can use either the same poly pitch or a reduced pitch.

- **Drawing grid.** Finally, we define the minimum size for a *drawing grid*. Once we determine the number of grids per pitch, we define the size of a drawing grid by simple arithmetic division and rounding. In the present study, we consider 5 grids per pitch and 6 grids per pitch. If we use 5 grids per pitch, the size of the minimum drawing grid is $26nm$, and if we use 6 grids per pitch, the size of the minimum drawing grid is $22nm$. Using the minimum drawing grids, we define design rules to draw layouts. Table 6.16 summarizes two sets of design rules corresponding to the two drawing-grid sizes.

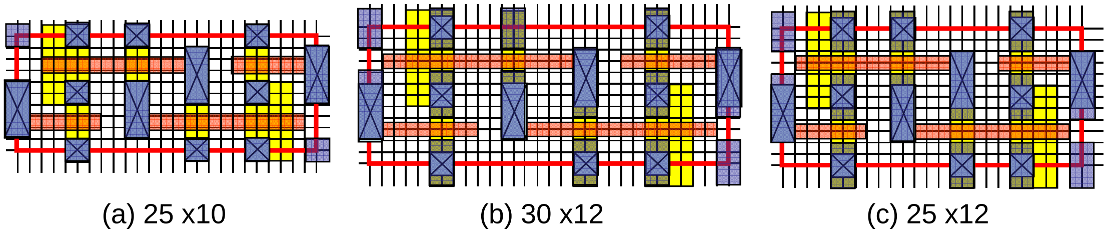
Based on the design rules, we develop three different bitcell layouts. Figure 6.14(a) shows a bitcell layout with the 5-grid rule and Figure 6.14(b) show a bitcell layout with the 6-grid rule. Figure 6.14(c) is also generated with the 6-grid rule but its horizontal pitch is five times the 6-grid rule's drawing grid. The bitcell areas are $0.169\mu m^2$, $0.174\mu m^2$ and $0.145\mu m^2$, respectively.

Among the bitcells, Figure 6.14(c) is the best candidate considering area, poly-to-contact space margin, and electrical characteristics. Figure 6.15 shows the complete layout of each layer of the 25×12 bitcell (Figure 6.14(c)). All layers except the diffusion layer have 1-D regular pitch and width.

[8] Horizontal pitch is constrained by the diffusion-to-nwell spacing, diffusion-to-diffusion spacing, and minimum contact/metal/via pitches (which are in turn determined by IL resolution), etc.

Table 6.16: Two sets of design rules for IAL-friendly layout.

Design rule item	5-grid rule	6-grid rule
Unit drawing grid size	26nm (= 1 grid)	22nm (= 1 grid)
Metal min. width/space	52nm (= 2 grid)	44nm (= 2 grid)
Contact/Via width/space	52nm (= 2 grid)	44nm (= 2 grid)
Diffusion width/space	52nm (= 2 grid)	44nm (= 2 grid)
poly-to-contact space	23nm	28nm
poly pitch	130nm	132nm


(a) 25 x10

(b) 30 x12

(c) 25 x12

Figure 6.14: Three candidate bitcell layouts.

6.3.3 Manufacturability Study

We evaluate the feasibility of the proposed bitcells in terms of lithography and circuit simulations.

Lithography simulation setup. For lithography simulation, we use *Fraunhofer Institute IISB's Dr. LiTHO version 0.10.5*. We use full-vector models on thin mask and high-contrast positive 50nm resist model parameters. We apply a simple optical proximity correction (OPC). We use a *crossquad* (XQUAD) setting for block exposure with NA = 1.2 of 66nm or 90nm IL pitches. We develop solutions for all layers except the diffusion layer. We use a positive-tone resist for poly layer and negative-tone resist for all other layers. Table 6.17 summarizes the simulation settings for all layers. Consistent with the previous work [144], *chromeless phase assist* (CPA) masks provide best-quality for block exposure.

Lithography simulation result. Mask patterns, second exposure images, final exposure images and final resist patterns for poly layer are shown in Figures 6.16(a), (b),

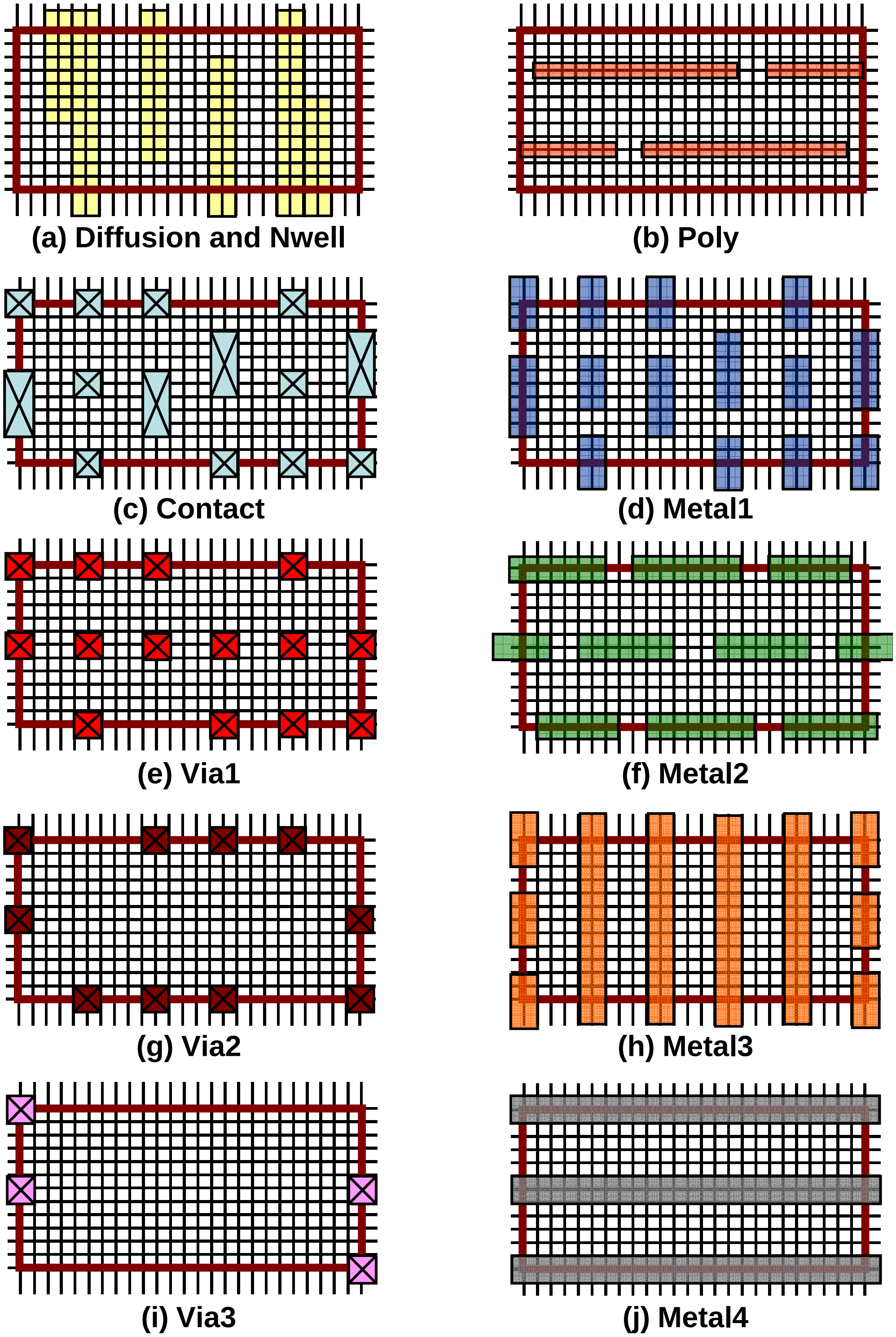


Figure 6.15: Layout for each layer.

Table 6.17: Lithography simulation conditions.

Layer	IL pitch	Mask type				Resist type
		Binary chrome mask		CPA mask		
		σ_{Center}	σ_{Radial}	σ_{Center}	σ_{Radial}	
Poly	$66nm$	0.625	0.15	-	-	positive
Contact	$90nm$	0.890	0.15	0.94	0.15	negative
M1	$90nm$	0.890	0.15	0.94	0.15	negative
V1	$90nm$	0.410	0.15	-	-	negative
M2	$90nm$	0.940	0.15	0.94	0.15	negative
V2	$90nm$	0.410	0.15	-	-	negative
M3	$90nm$	-	-	0.94	0.15	negative
V	$90nm$	0.410	0.15	-	-	negative
M4	$90nm$	-	-	0.94	0.15	negative

(c) and (d), respectively. Figure 6.17 shows the resist patterns for contact, M1 and V1 layers. We observe that all patterns including rectangular contacts and very high-dense hole arrays are successfully printed through the IAL process.

6.3.4 Electrical Characteristics

We verify electrical characteristics of the proposed bitcells according to the following metrics.

Butterfly curve. Butterfly curve is generally used to qualify the static noise margin. To generate the butterfly curve, we measure the voltage-transfer curve between internal nodes, NL and NR. Figure 6.18 shows the butterfly curves of the proposed bitcell and the reference bitcell for which transistor sizes are shown in Table 6.15. We observe that the static noise margin of the proposed bitcell is similar to that of the reference.

N-Curve. To obtain more insight into static characteristics, the N-curve can be analyzed [158]. The N-curve is generated by plotting the current consumed by an internal node as the node switches from low to high. SINM (SVNM) is the static current (volt-

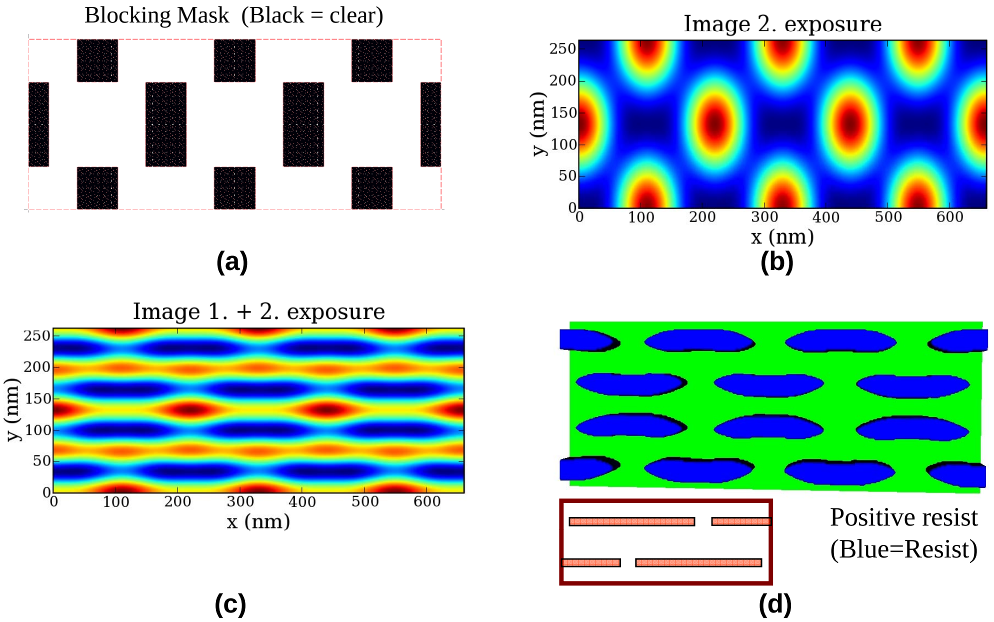


Figure 6.16: Lithography simulation results on poly layer using a binary mask.

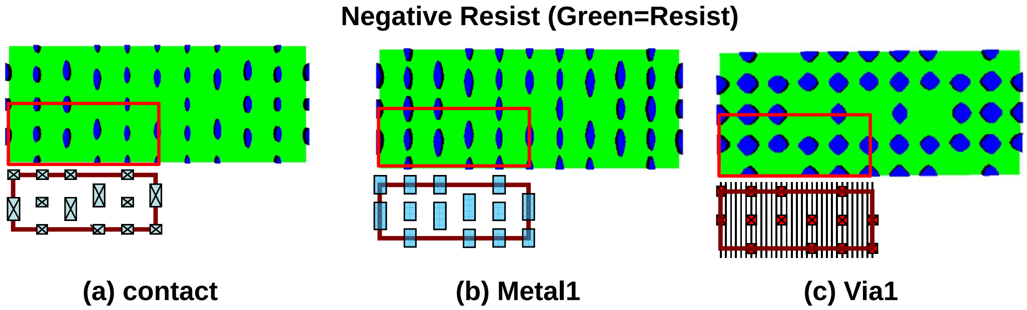


Figure 6.17: Resist (negative-tone) patterns for (a) contact layer, (b) M1 layer using a CPA mask and 2-beam IL, and (c) V1 layer using CPA mask and 4-beam IL.

age) noise margin, which is strongly related to read stability; WTI (WTV) is write trip current (voltage), which is strongly related to writeability. SINM increases and WTI decreases, as the transistor width increases. Larger SINM implies better read ability and smaller WTI represents worse writeability. Figure 6.18 compares N-curves of the proposed bitcell (SINM_reg and WTI_reg) and the reference bitcell (SINM_TSMC and WTI_TSMC). According to the figure, the proposed bitcell has better read stability but worse writeability.

I$_{read}$. I_{read} is the measured current at a bitline when wordline is switched to high. Large I_{read} implies better read stability. We observe again that the proposed bitcell has better read stability as shown in the first row of Table 6.3.4.

I$_{leakage}$. $I_{leakage}$ is the measured current from the supply node when a bitcell is in stable steady state. $I_{leakage}$ is important not only as power consumption itself, but as a metric of stable data retention. Smaller $I_{leakage}$ is preferred. However, we observe that the proposed bitcell has slightly larger $I_{leakage}$, as shown in the second row of Table 6.3.4.

VDD$_{hold}$. VDD_{hold} is the minimum supply voltage required to hold a bit of data, and is measured by lowering supply voltage and monitoring the internal nodes. When the voltage difference between NL and NR becomes less than a sensing margin, the internal data cannot be captured by the sense amplifier and the data will be lost. Comparison of VDD_{hold} in the third row of Table 6.3.4 does not show any significant difference.

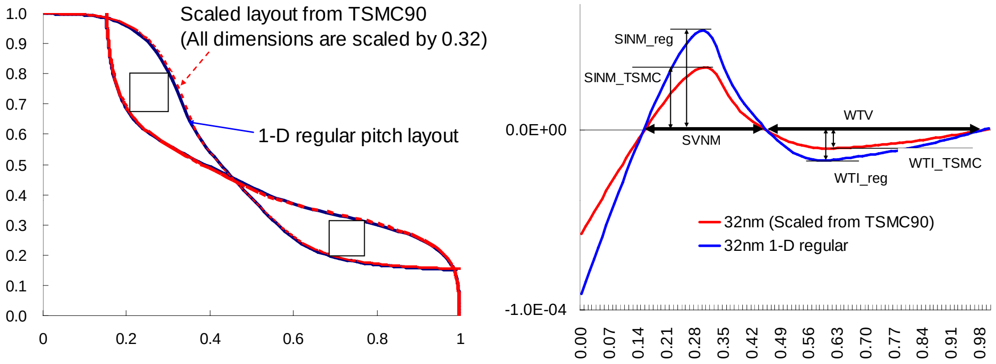


Figure 6.18: Butterfly curve and N-curve for the reference and proposed bitcells.

From the simulation results, we conclude that the proposed bitcell has better read stability but worse writeability. However, we note that since there exists flexibility with

Table 6.18: Comparison of I_{read}, $I_{leakage}$ and VDD_{hold}.

	Reference (Scaled TSMC)	1-D regular (30×12 and 25×12)
I_{read}	$41.2\mu A$	$66.7\mu A$
$I_{leakage}$	$85.4nA$	$142.7nA$
VDD_{hold}	$110mV$	$118mV$

respect to diffusion-layer patterning, we can further improve writeability by adjusting diffusion sizes.

Figures 6.19, 6.20 and 6.21 compare the butterfly curve, N-curve, I_{read}, $I_{leakage}$ and VDD_{hold} of the three proposed bitcells across three operating conditions. Figures 6.19 and 6.20 do not show significant degradation of electrical characteristics, across all operating conditions. We observe that the 25×10 bitcell has the best read stability, but worst writeability, as we expect from the larger transistor sizes used in the bitcell. From Figure 6.21, we again see that the 25×10 bitcell has highest I_{read} and highest $I_{leakage}$; Results on VDD_{hold} do not show any difference. We also observe that in all the simulation results, graphs of the 25×12 bitcell are near-perfectly overlaid by those of the 30×12 bitcell. Therefore, we conclude that the 25×12 bitcell is the best candidate when considering electrical stability as well as area.

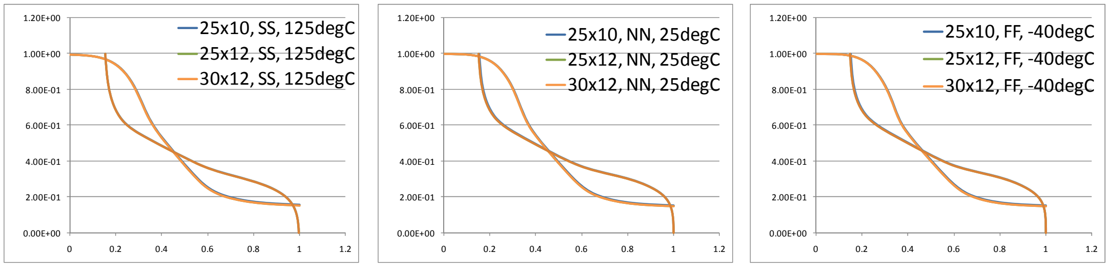


Figure 6.19: Butterfly curve comparison at different operating conditions.

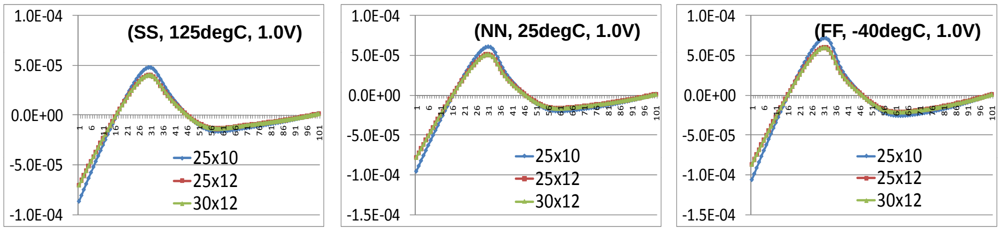
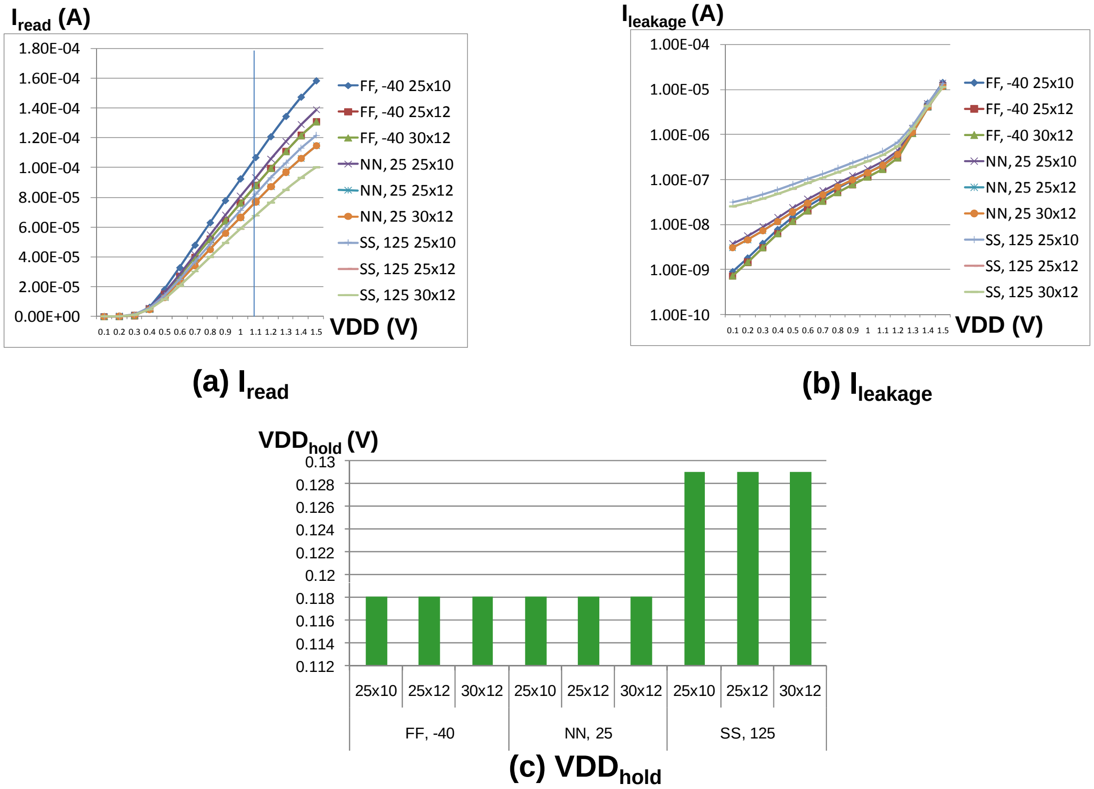


Figure 6.20: N-curve comparison at different operating conditions.



(a) I_{read}

(b) $I_{leakage}$

(c) VDD_{hold}

Figure 6.21: I_{read}, $I_{leakage}$ and VDD_{hold} comparison at different operating conditions.

6.4 Conclusions and Research Directions

Double patterning lithography is an inevitable solution and is being adopted for $32nm$ and below technologies. However, due to bimodality, i.e., two CD populations within a die, on-chip timing variability increases substantially beyond the variability that occurs with traditional single-exposure lithography.

To mitigate the timing variability in double patterning, we have proposed a new metric that quantifies the delay variation of timing paths, and implemented an optimal cell-based timing-aware color assignment technique for double patterning that reduces both timing delay as well as timing variation. To address the increased coloring conflicts due to this intentional timing-aware coloring, we have also proposed a dynamic programming-based detailed placement algorithm that minimizes coloring conflicts by perturbing placement and exploiting whitespace in the given placement. With this new methodology, we effectively reduce the timing delay as well as timing variation for DPL-patterned designs. We achieve maximum $271ps$ ($55.75ns$) reduction in the worst (total) negative slack and 70% (72%) reduction in the worst (total) negative slack variation in double patterning-applied designs.

Further research beyond this work seeks (i) to analyze the net benefits of adopting double patterning with consideration of bimodality, so that designers and lithographers can best trade off design and process margins, (ii) to seek more accurate metrics (objective functions) for further enhancement of timing quality through timing path balancing, (iii) to explore different objectives for the placement perturbation, (iv) to investigate the tradeoff between recoloring and displacement in terms of impact on timing quality, and finally, (v) to develop a simultaneous timing-aware coloring and conflict removal methodology as a golden timing and placement optimizer for double patterning lithography in the presence of bimodality.

We have also proposed a design methodology for 1-D regular-pitch SRAM bitcell layouts which are amenable to an interference-assisted lithography (IAL) manufacturing process. We derive required design rules for a 6-T bitcell to have $32nm$ gate length, and propose a family of IAL-friendly bitcell layouts. Through lithography and circuit simulations, we confirm that the proposed bitcell layouts can be successfully printed by IAL and that their electrical characteristics are comparable to those of existing bitcell layouts.

As next goals for IAL, we also seek (1) to provide stronger electrical circuit validation with statistical analysis and dynamic circuit analysis, including comparison with production SRAM bitcells; (2) to develop a full 1-D regular-pitch bitcell including diffusion layer; and (3) to report measured data obtained from tapeout.

6.5 Acknowledgments

Chapter 6 is in part a reprint of “Timing Yield-Aware Color Reassignment and Detailed Placement Perturbation for Bimodal CD Distribution in Double Patterning Lithography”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(8) (2010), “Dose Map and Placement Co-Optimization for Improved Timing Yield and Leakage Power”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(7) (2010), “Timing Yield-Aware Color Reassignment and Detailed Placement Perturbation for Double Patterning Lithography” *Proc. ACM/IEEE International Conference on Computer-Aided Design*, 2009, “Interference Assisted Lithography for Patterning of 1D Gridded Design”, *Proc. SPIE Symposium on Advanced Lithography*, 2009, “Dose Map and Placement Co-Optimization for Timing Yield Enhancement and Leakage Power Reduction”, *Proc. ACM/IEEE Design Automation Conference*, 2008, and “32nm 1-D Regular Pitch SRAM Bitcell Design for Interference-Assisted Lithography”, *Proc. SPIE BACUS Symposium on Photomask Technology and Management*, 2008. I would like to thank my coauthors Mr. Robert T. Greenway, Mr. Mohit Gupta, Dr. Rudolf Hendel, Professor Andrew B. Kahng, Dr. Chul-Hong Park, Mr. John S. Petersen, Dr. Zhilong Rao, Dr. Michael C. Smayling, and Professor Hailong Yao.

I also thank Mr. Andy Hoover at Freescale Semiconductor for suggesting the usage of even number of fingers in all devices of the clock distribution (buffers, inverters) to mitigate bimodality, and Mr. Kevin Liu (formerly of Applied Materials) and Mr. Scott Mackay of Petersen Advanced Lithography for their support and many helpful interactions. I note that research at UCSD was supported by the Semiconductor Technology Academic Research Center (STARC).

Chapter 7

Design-Manufacturing Co-Optimization

In this chapter, we propose a novel modeling framework which includes (1) capacitance modeling of a line-end extension and consequent current density changes in channel, and (2) I_{on} and I_{off} modeling from the new capacitance model. We define a new electrical metric for a line-end shape as the $expected$ change in I_{on} or I_{off} under a given overlay error distribution. We further apply a *superellipse* form to parameterized line-end shapes, and we then generate a large variety of line-end shapes. We evaluate the electrical metric on these line-end shapes to come up with simple rules of thumb that the lithographer can use to quickly evaluate the quality of a lithography + OPC solution with respect to line-end shaping. We also evaluate post-litho line-end shapes while varying OPC, lithography and design rule parameters, and find a tradeoff between cost and electrical characteristics.

7.1 Introduction

In the low-k_1 patterning regime ($k_1 < 0.3$), gate shape is no longer a perfect rectangle. Current circuit analysis tools assume that transistor gate and diffusion shapes are perfect rectangles, and are unable to handle complicated geometries. Large discrepancies can be observed between the simulated and measured values of such transistor parameters as current and threshold voltage. Moreover, such discrepancies are likely to become more significant as overlay becomes a more critical issue in future technologies.

Several previous works electrically model non-rectilinear geometries [37] [143] [89] [70] [109] [146] [107]. All of these works consider the threshold voltage and hence the current density to be uniform along the device width. As a result, variations including that of gate length are treated the same, irrespective of the location of the variation. It is known that the fringing capacitance[142] due to line-end extension and dopant scattering significantly affect the device threshold voltage. These effects are more pronounced near the device edges and roll off sharply toward the center of the device. Several previous works have accounted for this effect via non-rectangular gate models [77] [79] [165]. Most of these works slice non-rectangular gates along the device width at a certain level of granularity, then sum up I_{on} (or I_{off}) of all slices to model I_{on} (or I_{off}) of the non-rectangular device. For each slice, the current density model corresponding to its length is used. The total current of the device is the integral of the current density over its width. The total current can be used to provide an equivalent length for the rectangular device, so that it can be modeled using SPICE-like tools. Gupta et al. [80] have also used TCAD simulation to investigate the impact of the non-rectangular shape of diffusion on circuit performance.

The primary concerns of lithographic patterning have been *line-end pullback* and *linewidth*. Traditionally, lithographers have measured line-end printing quality by (1) line-end gap (space between two facing line-ends), (2) CD at the gate edge (*LW0*), and (3) non-existence of line-end shortening (i.e., the condition where poly fails to cover active completely). Though these geometric metrics have served as good indicators, the ever-rising contribution to the layout area of line-end extension – defined as the extension of polysilicon shape beyond the active edge – strongly motivates the reduction of pessimism in qualifying line-end patterning. The quality of line-end patterning depends on the rounded shape of the line-end extension as well as on the linewidth at device edge (and, to a negligible extent, on the line-end gap).

Electrically-aware metrics for line-end extension can be helpful in this regard. The device threshold voltage is, with nominal patterning, a weak function of line-end shapes. However, the electrical impact of line-end shapes can increase with overlay errors, since displaced line-end extensions can be enclosed in the transistor channel, and non-ideal line-end shape will manifest as an additional gate CD variation. We employ 3-D TCAD simulators [26] [27] to investigate the changes of gate capacitance, I_{on}, and I_{off}, according to various line-end shapes and line-end extension lengths. We observe that I_{on} and I_{off} have strong relationships with line-end shapes. For example, preliminary experiments using the 3-D TCAD tool, *Synopsys DaVinci* [20], indicate that line-end extension length can affect I_{on} and I_{off} by as much as 4.5% and 30%, respectively, as shown in Figure 7.1. Moreover, the electrical impact of the line-end extension can vary significantly with overlay.

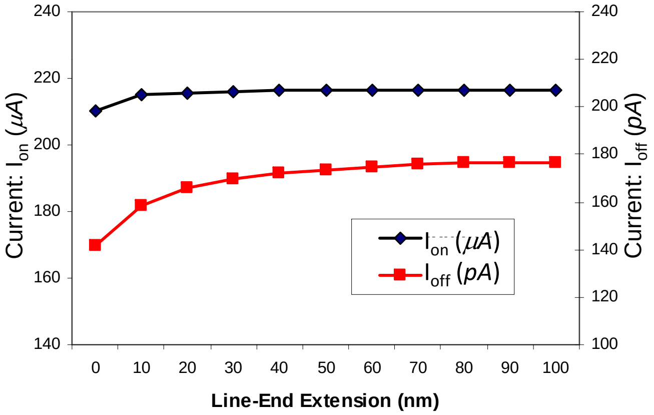


Figure 7.1: I_{on} and I_{off} change due to line-end extension length.

7.2 Non-Uniform Gate Model

An electrical model for the line-end extension must capture change in power and performance characteristics of a given device. For line-end modeling, we convert a lithography contour into several sliced rectangles as shown in Figure 7.4. For each slice, we use a current density model corresponding to its length l_i. The sum of the currents of all slices is the total current of the device. The total current can be used to calculate the gate length of an equivalent rectangular device, so that the current can be evaluated by SPICE-like tools. This line-end model, along with a non-uniform channel model similar to Gupta et al.'s work [79], is used to model the device under overlay error. We calculate the probability of each slice being placed at a given location from the overlay error distribution. Using location-dependent fringe capacitance and current models for line-end extension and channel as well as the overlay error probability model, we predict I_{on} or I_{off} considering a given overlay error.

Line-end extension affects the fringe capacitance to the channel of a MOS gate, which in turn affects the threshold voltage of the gate. Hence, I_{on} and I_{off} models accounting for line-end impact can be developed in terms of line-end capacitance. Figure 7.2 shows the overall flow of the line-end modeling.

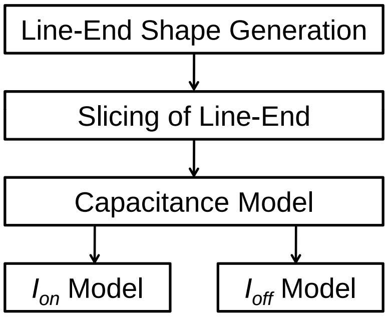


Figure 7.2: Line-end extension modeling flow. I_{on} and I_{off} can be modeled as functions of line-end extension capacitance.

7.2.1 Superellipse

We propose a line-end shape generation method using the *superellipse* equation. A superellipse is defined as the set of all points (x, y) that satisfy

$$\left|\frac{x}{a}\right|^n + \left|\frac{y-k}{b}\right|^n = 1,$$

where $n > 0$, a and b are the semi-minor and semi-major axes of the superellipse, and k represents line-end shift in the y-axis. For a given line-end shape, a and b represent gate length and length of line-end extension, respectively. The exponent n determines the curvature, or corner rounding, of the line-end extension. For example, n = 2 yields an ordinary ellipse, and increasing n beyond 2 yields shapes with sharper corners, increasingly resembling a rectangle. The center o of a superellipse represents an overlay error value where 3σ is considered to be the worst-case overlay error.

To capture asymmetric line-end shapes, the superellipse can be rotated about its center using the transform x = $x'cos\theta$ - $y'sin\theta$ and y = $x'sin\theta$ + $y'cos\theta$ (or x = $x'cos\theta$ + $y'sin\theta$ and y = $-x'sin\theta$ + $y'cos\theta$), where x' and y' are the coordinates of the original superellipse shape. The quantity $b + k$ represents the new line-end extension length (LEE) after line-end shift. In this work, we focus on symmetric line-end shapes only.

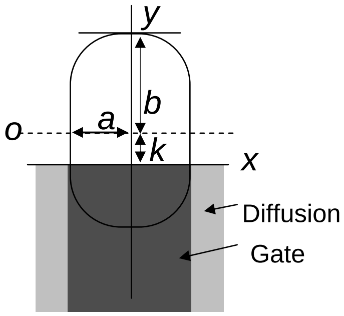


Figure 7.3: An example of a line-end shape represented by superellipse equation.

7.2.2 Capacitance Model

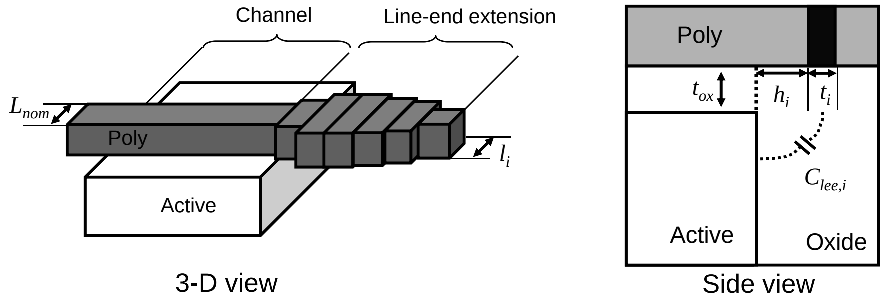
Gate capacitance is a sum of capacitance of the gate channel ($C_{channel}$) and capacitance of the line-end extension (C_{lee}). C_{lee} is the fringe capacitance between line-end extension and gate channel. We can simply model the capacitance of the line-end extension as the sum of the fringe capacitance of each slice of the line-end extension, as illustrated in Figure 7.4:

$$C_{lee} = \sum_{i=1}^{N} C_{lee,i} \tag{7.1}$$

$$\text{where} \qquad C_{lee,i} = l_i^{\alpha} \left(\frac{t_i}{h_i + t_i/2 + t_{ox}} \right)^{\beta}$$

Capacitance of each line-end slice or segment can be modeled as a function of its length (l_i), thickness (t_i), distance from the gate edge (h_i) and gate oxide thickness (t_{ox}). Intuitively, the fringe-capacitance effect increases with larger length, larger thickness and smaller distance from the gate edge.

We simulate capacitance changes using a 3-D RC extraction tool, *Synopsys Raphael* [26], while varying line-end extension length. In this simulation, we assume STI oxide depth of $100nm$, t_{ox} of $1.5nm$, gate thickness of $100nm$, and gate length of $45nm$, consistent with the 3-D device simulation setup used to characterize I_{on} and

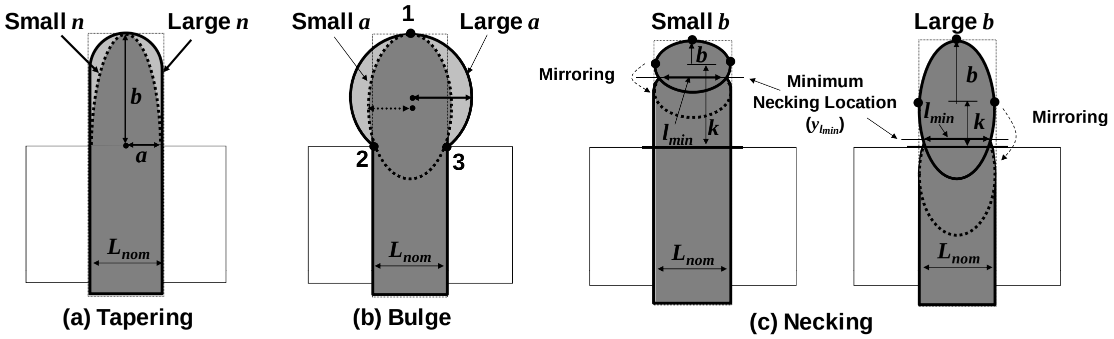


3-D view

Side view

Figure 7.4: Modeling line-end capacitance.

I_{off}. We find model coefficients α and β using the Matlab nonlinear regression function (*nlinfit*) [12]. The fitted model shows 1.19% of average magnitude error on 150 different line-end shapes, with α = 0.1389 and β = 0.4253. All dimensional parameters of the model are in units of nm, and the calculated capacitance is in units of aF.



(a) Tapering

(b) Bulge

(c) Necking

Figure 7.5: Line-end shapes represented by the superellipse equation.

Figure 7.5 (a)-(c) show three representative shapes of line-end extension, for which l_i can be calculated directly from the superellipse parameters a, b, k, and n, as follows.

Tapering: Figure 7.5(a) is the case of *tapering*, in which the center of the superellipse is on the gate edge, and l_i can be calculated as

$$l_i = 2a\left(1 - \left|\frac{h_i - k}{b}\right|^n\right)^{\frac{1}{n}}$$

Bulge: Figure 7.5(b) represents a *bulge* line-end shape, in which the minor axis is greater than the nominal linewidth (L_{nom}) and the minimum linewidth between the center of the superellipse, and the gate edge is greater than or equal to L_{nom}. The corresponding y-coordinate, when the linewidth is L_{nom}, is calculated by

$$y_{L_{nom}} = k - b\left(1 - \left|\frac{L_{nom}/2}{a}\right|^{n}\right)^{\frac{1}{n}}$$

The value of l_i for the bulge shape is then computed as

$$\begin{aligned} l_i &= 2a\left(1 - \left|\tfrac{h_i - k}{b}\right|^{n}\right)^{\frac{1}{n}}, && h_i \geq y_{L_{nom}} \\ l_i &= L_{nom}, && 0 \leq h_i \leq y_{L_{nom}} \end{aligned}$$

Necking: Figure 7.5(c) gives two examples of *necking* shapes. It is difficult to ensure smooth changes in linewidth for necking cases by using one superellipse. Therefore, we apply a mirroring transform where the mirroring axis has the minimum linewidth (l_{min}). The corresponding y-coordinate of the mirroring axis $y_{l_{min}}$ is calculated by

$$y_{l_{min}} = k - b\left(1 - \left|\frac{l_{min}/2}{a}\right|^{n}\right)^{\frac{1}{n}} \tag{7.2}$$

The value of l_i for the necking shape is then

$$\begin{aligned} l_i &= 2a\left(1 - \left|\tfrac{h_i - k}{b}\right|^{n}\right)^{\frac{1}{n}}, && h_i \geq y_{l_{min}} \\ l_i &= 2a\left(1 - \left|\tfrac{h_i - 2y_{l_{min}} + k}{b}\right|^{n}\right)^{\frac{1}{n}}, && 2y_{l_{min}} - k \leq h_i \leq y_{l_{min}} \\ l_i &= L_{nom}, && h_i \leq 2y_{l_{min}} - k \end{aligned}$$

7.2.3 I_{on} Model

Using the capacitance model for line-end extension, we propose a new model for I_{on}. Inverse narrow width effect (iNWE) due to the line-end fringe capacitance is modeled in the BSIM4 SPICE model [3] as an exponentially decaying function of gate width. We assume that the impact of line-end capacitance decreases exponentially from the gate edge to the channel, to account for the iNWE model in BSIM4. Figure 7.6 illustrates the modeling approach, where i_{on} is on-current of an individual gate segment s, and the segment index s represents the distance from the gate edge. C_{lee_top} and C_{lee_bottom} represent line-end capacitances at the top and bottom sides of a gate, respectively. Thus, total current I_{on} is expressed as a sum of segment currents i_{on} over all segments:

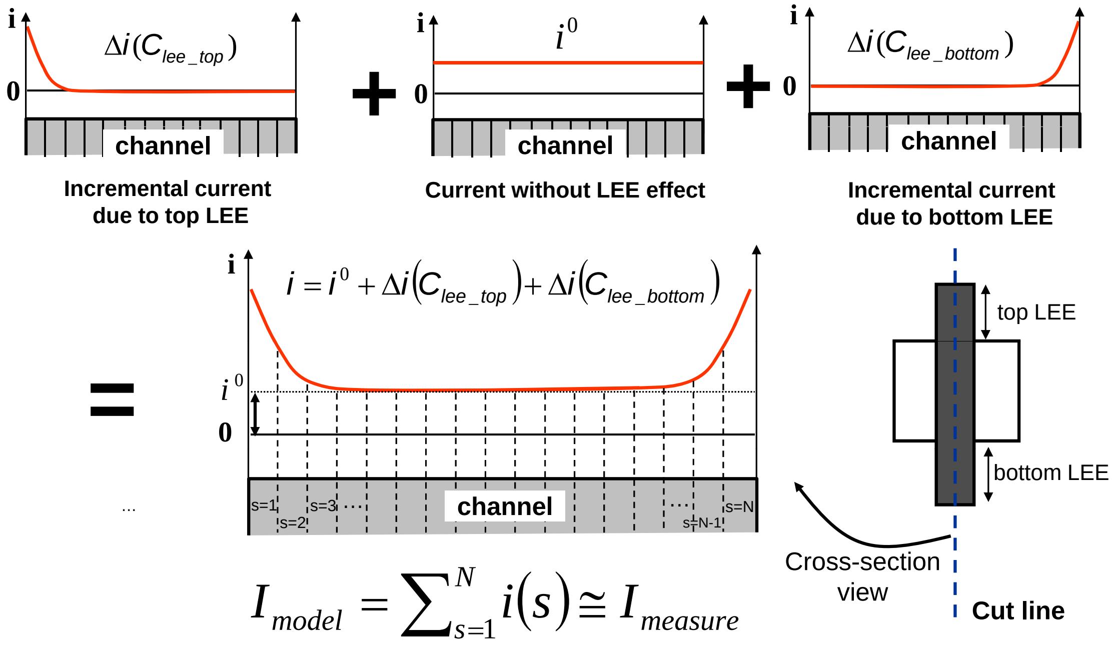

i
0
$\Delta i(C_{lee\_top})$
channel
Incremental current due to top LEE
+
i
0
i^0
channel
Current without LEE effect
+
i
0
$\Delta i(C_{lee\_bottom})$
channel
Incremental current due to bottom LEE
=
i
$i = i^0 + \Delta i(C_{lee\_top}) + \Delta i(C_{lee\_bottom})$
i^0
0
channel
s=1
s=2
s=3
s=N-1
s=N
$I_{model} = \sum_{s=1}^{N} i(s) \cong I_{measure}$
top LEE
bottom LEE
Cross-section view
Cut line

Figure 7.6: Non-uniform channel modeling procedure.

$$
\begin{aligned}
I_{on} &= \sum_{s=1}^{N} i_{on}(C_{lee\_top}, C_{lee\_bottom}, s, L) \\
i_{on}(C_{lee\_top}, C_{lee\_bottom}, s, L) &= i^0_{on}(L_s) + \\
&\quad \Delta i_{on}(C_{lee\_top}, s, L_s) + \Delta i_{on}(C_{lee\_bottom}, N - s + 1, L_s)
\end{aligned}
$$

Here, $i^0_{on}(L_s)$ is the on-state current of a gate segment which is not affected by line-end extension. The additive on-state current (Δi_{on}) for each segment of a gate is modeled as a function of the line-end capacitance (C_{lee}), segment index (s), and length (in gate length direction) of the segment (L_s). More precisely, $i^0_{on}(L_s)$ and Δi_{on} are defined as

$$
\begin{aligned}
i^0_{on}(L_s) &= h(L_s) \cdot i^0_{on\_nom} \\
\Delta i_{on}(C_{lee}, s, L_s) &= f(C_{lee}) \cdot g(s) \cdot h(L_s) \\
&f(C_{lee}) = (C_{lee})^{\alpha} \\
&g(s) = \gamma e^{-\beta(s-1)} \\
&h(L_s) = (\frac{L_{nom}}{L_s})^{k}
\end{aligned}
$$

where $i^0_{on\_nom}$ is the baseline current of a segment with a nominal gate length L_{nom}, as measured from the current value difference between two large-width devices that have the same line-end shape. Functions f and g account for the size and the exponential decay rate of the impact of line-end capacitance. Function h linearly scales the calculated current based on the gate length of a gate segment, since on-state current is an inverse-linear function of gate length.

The model accuracy using Matlab nonlinear regression function (*nlinfit*) [12] is 0.24$\%$ average magnitude of error for 38$nm \le L_s \le$ 52nm. Here, α, β and γ are 0.1616, 0.030 and 0.1349, respectively, and k is 1.035. I_{on} is given in units of uA.

7.2.4 I_{off} Model

I_{off} is similarly modeled as a sum of segment currents i_{off}. Again, C_{lee_top} and C_{lee_bottom} represent line-end capacitances at the top and bottom sides of a gate. Finally, total off-state current I_{off} is expressed as

$$
\begin{aligned}
I_{off} &= \sum_{s=1}^{N} i_{off}(C_{lee\_top}, C_{lee\_bottom}, s, L_s) \\
i_{off}(C_{lee\_top}, C_{lee\_bottom}, s, L_s) &= i^0_{off}(L_s) + \\
i_{off}(C_{lee\_top}, s, L_s) &+ \Delta i_{off}(C_{lee\_bottom}, N-s+1, L_s)
\end{aligned}
$$

where $i^0_{off}(L_s)$ is the off-state current of a gate segment which is not affected by line-end extension. The additive off-state current (Δi_{off}) for each segment is modeled as a function of the line-end capacitance (C_{lee}), segment index (s), and its length (L_s). More precisely:

$$
\begin{aligned}
i^0_{off}(L_s) &= h_1(L_s) \cdot i^0_{off\_nom} \\
\Delta i_{off}(C_{lee}, s, L_s) &= f(C_{lee}) \cdot g(s) \cdot h_2(L_s) \\
& f(C_{lee}) = (C_{lee})^{\alpha} \\
& g(s) = \gamma e^{-\beta(s-1)} \\
& h_1(L_s) = k_1 e^{k_2(L_s - L_{nom})} \\
& h_2(L_s) = k_3 e^{k_4(L_s - L_{nom})}
\end{aligned}
$$

where $i^0_{off\_nom}$ is the baseline current of a segment with nominal gate length L_{nom}, as measured from the current value difference between two large-width devices that have the same line-end shape. Functions f and g again account for the size and the exponential decay rate of the impact of line-end capacitance. Functions h_1 and h_2 exponentially scale the calculated current based on the gate length of a gate segment, since off-state current is an exponential function of gate length.

We find the coefficients using Matlab nonlinear regression function (*nlinfit*) [12]. Here, α, β and γ are 0.045, 0.012 and 667.2, respectively, and k_1, k_2, k_3 and k_4 are -0.5129, 0.6118, -0.2739 and 1.971, respectively. The model shows 1.02% average magnitude error compared to TCAD simulation for $38nm \leq L_s \leq 52nm$. I_{off} is given in units of nA.

7.2.5 Overlay Error Model

With overlay error, the segments near the channel edge change. Since segments in the channel affect I_{on} and I_{off} differently compared to the segments in the line-end extension, we first determine whether the segment belongs to the channel or the line-end extension. Overlay error is a vector component quantity in x and y directions. We assume that the minimum poly-to-diffusion spacing is larger than the overlay error, so that overlay error does not cause any spurious transistor channels. Therefore, x-direction (i.e., perpendicular to the poly direction) overlay error is neglected. Given an overlay error, we can calculate the I_{on} and I_{off} of the entire gate by summing up the current values (i_{on} and i_{off}) of segments that are in the channel.

Overlay error is assumed to have a normal distribution. To account for the different probabilities for different magnitudes of overlay error and corresponding current changes, we calculate expected current I_{exp} based on the normal distribution assumption of overlay error, with mean and 3σ assumed to be zero and $10nm$, respectively.[1] Due to the segmentation-based current calculation, we discretize the range of magnitudes of overlay error. N_{sites} denotes the number of possible sites of poly placement due to overlay error. $P(S)$ is the probability of poly being placed at the S^{th} site, where $1 \leq S$

[1] The ITRS sets 3σ overlay error for the MPU $45nm$ half-pitch node as $11nm$ [11]. We use $10nm$ for simplicity in calculations. All results in the rest of this section assume $10nm$ 3σ overlay error.

$\leq N_{sites}$. In the modeling, we use $5nm$ for segmentation size and 5 different sites (i.e., N_{sites} = 5) for overlay error. The third site represents no (i.e., $0nm$) overlay error and the others represent movement of poly segments by $\pm 5nm \sim \pm 10nm$. Each site S has probability $P(S)$ calculated by integrating the normal distribution between the limits of the site S, as shown in Figure 7.7(b). The current $I(S)$ of a gate poly placed at site S is calculated according to where each segment of poly belongs. Finally, we can calculate the expected current I_{exp} by integrating the product of $P(S)$ and $I(S)$ over the range of possible overlay error values.

$$I_{exp} = \sum_{S=1}^{N_{sites}} P(S)I(S)$$

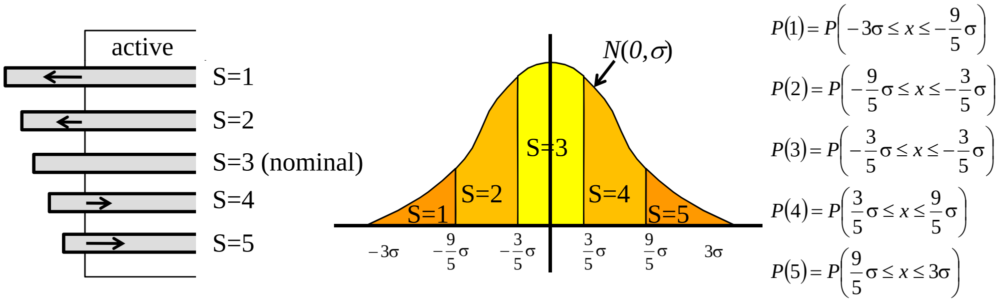

active
S=1
S=2
S=3 (nominal)
S=4
S=5
N(0,σ)
S=1
S=2
S=3
S=4
S=5
−3σ
$-\frac{9}{5}\sigma$
$-\frac{3}{5}\sigma$
$\frac{3}{5}\sigma$
$\frac{9}{5}\sigma$
3σ
$P(1) = P\left(-3\sigma \leq x \leq -\frac{9}{5}\sigma\right)$
$P(2) = P\left(-\frac{9}{5}\sigma \leq x \leq -\frac{3}{5}\sigma\right)$
$P(3) = P\left(-\frac{3}{5}\sigma \leq x \leq -\frac{3}{5}\sigma\right)$
$P(4) = P\left(\frac{3}{5}\sigma \leq x \leq \frac{9}{5}\sigma\right)$
$P(5) = P\left(\frac{9}{5}\sigma \leq x \leq 3\sigma\right)$

Figure 7.7: Overlay error model: (a) five discretized overlay errors, and (b) probability $P(S)$ calculation for each overlay error S.

7.3 Electrical Assessment of Line-End Shapes

In this section, we evaluate the accuracy of the models and assess the electrical characteristics of the various line-end shapes generated from the superellipse equation.

7.3.1 Model Accuracy

We apply the proposed model to an ideal rectangular line-end shape. Table 7.1 shows the comparisons of the model and the TCAD simulation. We measure I_{on} and I_{off}, changing the line-end extension length. Columns 1 and 2 show the drawn transistor width and line-end extension length, respectively. Columns 3 and 4 show the I_{on} and I_{off} values without considering the line-end effects. Comparing column 5 with 7, and 6 with 8, shows the accuracy of the model. ***Maximum errors of the I_{on} and I_{off} models are 0.66% and 2.50%, respectively.***

Table 7.1: Model accuracy and impact of overlay error on rectangular line-end extension.

Width (nm)	LEE (nm)	Drawn		Model w/o overlay		Sentaurus		Model w/ overlay	
		I_{on} (μA)	I_{off} (nA)	I_{on} (μA)	I_{off} (nA)	I_{on} (μA)	I_{off} (nA)	I_{on} (μA)	I_{off} (nA)
100	100	105.0	50.5	105.3	51.3	105.0	50.5	105.2	51.3
	70			105.2	51.0	104.9	50.3	105.1	51.0
	50			105.0	50.7	104.8	50.0	105.0	50.7
	30			104.9	50.3	104.6	49.4	104.8	50.3
	10			104.6	49.3	103.9	48.1	104.4	48.8
200	100	209.3	96.9	209.1	97.0	209.3	96.9	208.9	97.0
	70			208.9	96.5	209.2	96.8	208.8	96.4
	50			208.7	96.0	209.1	96.5	208.6	95.9
	30			208.5	95.2	208.9	95.8	208.4	95.2
	10			208.0	93.5	208.2	94.1	207.7	92.5
300	100	311.4	136.7	312.1	138.3	311.4	136.7	311.9	138.2
	70			311.9	137.6	311.3	136.7	311.7	137.5
	50			311.7	136.9	311.3	136.6	311.5	136.8
	30			311.4	135.9	311.2	136.3	311.2	135.8
	10			310.8	133.6	310.2	134.3	310.5	132.2

Columns 9 and 10 show the impact of overlay error with 3σ = $10nm$. When we reduce the line-end extension, we can see the decreasing trends of I_{on} and I_{off}, since small line-end extension results in small gate capacitance and hence higher threshold voltage. This result implies that an unnecessarily large line-end rule is not desirable from the electrical point of view. Note that since the shape is perfectly rectangular, overlay error does not cause linewidth variation in the channel. Hence, the impact of overlay error is negligibly small for the ideal rectangular line-end shape.

7.3.2 Evaluation of Line-End Shapes

We also evaluate the line-end shapes generated by the proposed superellipse model. Tapering is a typical shape in the post-OPC silicon image. As noted above, corner rounding is represented by the superellipse parameter n. Larger n results in less corner rounding, but increases mask cost in terms of mask writing time and mask inspection since aggressive OPC needs to be applied. Bulging may be caused by inaccurate OPC and may be amplified under defocus. The degree of bulge shape is determined by a and by having positive k. Necking is a reduction in linewidth that is caused by an excessive OPC hammerhead, i.e., the hammerhead results in narrow linewidth at the channel edge under defocus, even if the hammerhead can compensate for corner rounding error at a best-focus condition.

For each shape generated from a superellipse as shown in Figure 7.5, we change the line-end extension length by shifting the entire poly shape, and calculate I_{off} for each shape.[2] When we reduce the line-end extension length, since the line-end part of the poly gate becomes enclosed by the diffusion, segments in the line-end extension turn into gate segments in the channel.

Table 7.2 shows the dependence of I_{on} and I_{off} on the superellipse exponent and the line-end extension length, i.e., on the *sharpness* of the line-end extension. In this case, the superellipse semi-minor and semi-major axes are fixed at 22.5nm and 100nm, respectively. The bold italic entries in the table show the cases where I_{off} remains within 10% of the 100nm LEE cases. ***As we increase n, the tapering becomes more rectangular, so that the $\mathbf{I}_{off}$ variation due to line-end extension length is reduced. As a result, LEE can be reduced further with larger n when 10% $\mathbf{I}_{off}$ increase is allowed.*** For example, for n = 2.5, LEE can be reduced from 100nm to 60nm, but for n = 4, LEE length can be reduced from 100nm to 40nm. As noted above, increased n requires more complex OPC and can increase OPC and mask costs. Note that in the table, some cases

[2] We limit the minimum line-end extension length to 20nm, to avoid line-end shortening by overlay error.

are out of the boundary of the model, but it is obvious that those cases must be avoided in design, so as to avoid excessive leakage current.

Table 7.2: I_{on} and I_{off} changes with line-end extension length and *sharpness* for 200nm width NMOS.

	Superellipse exponent (n)							
	2.5		3.0		4.0		5.0	
LEE (nm)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)
100	208.91	***96.91***	208.92	***96.92***	208.92	***96.93***	208.92	***96.94***
90	208.87	***96.80***	208.87	***96.77***	208.87	***96.78***	208.87	***96.78***
80	208.85	***96.99***	208.82	***96.72***	208.81	***96.62***	208.81	***96.61***
70	208.93	***98.15***	208.82	***97.05***	208.76	***96.52***	208.75	***96.43***
60	209.20	***101.70***	208.91	***98.39***	208.73	***96.66***	208.69	***96.31***
50	209.77	111.53	209.18	***102.18***	208.76	***97.47***	208.64	***96.40***
40	210.85	141.91	209.77	112.71	208.93	***99.97***	208.66	***97.14***
30	212.72	274.60	210.89	148.55	209.36	107.41	208.80	***99.70***
20	-	-	212.95	372.79	210.29	137.34	209.29	109.01

Table 7.3 shows the I_{on} and I_{off} dependence on the *fatness* of the bulge shape and the line-end extension length. The superellipse exponent is fixed at $n = 3.0$. Since we use a contour that passes through three points in Figure 7.5(b), if we change the semi-minor axis a, the other parameters b and k are determined automatically by solving the superellipse equation. For the bulge shape line-end extension, I_{off} variation is small compared to the tapering (sharpness) case. We also observe that I_{on} and I_{off} are reduced by 7% and 38% when we reduce the line-end length from 100nm to 20nm, with semi-minor axis length of 28nm. ***Typically, I_{on} and I_{off} decrease when line-end extension length decreases, since large-width line-end segments due to the bulge shape are turned into channel segments.***

Table 7.4 shows the I_{on} and I_{off} dependence on the location of necking and

Table 7.3: I_{on} and I_{off} changes with line-end extension length and *fatness* for 200nm width NMOS.

	Superellipse semi-minor axis (a) (nm)							
	25		26		27		28	
LEE (nm)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)
100	208.90	96.74	208.89	96.69	208.88	96.65	208.87	96.61
90	208.50	94.47	208.30	94.00	208.16	93.64	208.01	93.35
80	207.56	90.69	207.13	89.77	206.72	89.14	206.33	88.68
70	206.47	86.54	205.70	85.24	204.97	84.39	204.28	83.82
60	205.32	82.31	204.18	80.63	203.12	79.60	202.11	78.92
50	204.16	78.07	202.65	76.01	201.24	74.79	199.92	74.02
40	203.05	73.89	201.15	71.41	199.38	69.98	197.73	69.10
30	202.13	70.11	199.81	66.96	197.67	65.23	195.68	64.20
20	201.69	69.07	198.91	63.61	196.36	60.94	194.00	59.48

the line-end extension. For this simulation, we use a superellipse with 100nm line-end extension length. By changing the semi-major axis, we control the necking location where the linewidth is minimized. We shift the entire poly shape downward to model the reduction of the line-end design rule. The table shows that necking makes the device leaky, and that leakage current increases or decreases with line-end extension length. In particular, when the necking occurs near the channel edge, e.g., y_{min} = 0 or 10nm, I_{off} increases substantially for all line-end extension lengths. This is because the minimum linewidth of the necking is already enclosed by the channel as a result of overlay error.

The bold italic entries in Table 7.4 show the cases where the I_{off} increase remains within 10% of the 100nm LEE cases. As the necking location moves farther from the channel edge, we can reduce LEE further. Table 7.4 implies that ***if we cannot avoid necking shapes, the necking location must be placed at least as far as the maximum overlay error from the channel edge.***

Table 7.4: I_{on} and I_{off} changes with line-end extension length and *necking* for $200nm$ width NMOS.

	Necking location ($y_{l_{min}}$ in Eq. 7.2) (nm) for $100nm$ LEE. $l_{min} = 40nm$							
	0		10		30		50	
LEE (nm)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)	I_{on} (uA)	I_{off} (nA)
100	210.13	***115.79***	209.23	***100.30***	208.92	***96.93***	208.92	***96.94***
90	210.89	130.13	209.95	113.58	208.88	***96.88***	208.87	***96.78***
80	211.14	133.49	210.67	126.81	208.98	***98.37***	208.81	***96.61***
70	211.15	133.88	210.86	129.48	209.58	109.07	208.75	***96.41***
60	211.08	133.63	210.83	129.59	210.17	119.70	208.74	***96.84***
50	211.00	133.33	210.75	129.26	210.24	121.02	209.17	***104.33***
40	210.96	133.54	210.66	129.09	210.14	120.72	209.58	111.73
30	211.12	136.49	210.71	130.52	210.04	120.53	209.50	111.88
20	211.82	157.95	211.21	143.61	210.16	124.05	209.37	111.79

From all experiments, we observe that as line-end extension length is reduced from $100nm$, I_{on} and I_{off} are also reduced, due to reduced line-end capacitance. However with tapering or necking effects, if the small linewidth of the line-end segments is situated within the channel area due to overlay error, I_{off} increases significantly. We also observe that the impact of line-end extension itself is negligibly small due to the electrical characteristics, but that in combination with line-end pull-back and overlay error, small-linewidth line-end segments lead to large variation in I_{on} and I_{off}.

From the observations, the desirable attributes of line-end shapes can be summarized as follows.

- Larger n is preferred to suppress I_{off} variation. With larger n, we can further reduce the line-end extension length.

- Bulge may be the best line-end extension shape for I_{off}, since it can reduce I_{off} of the most leaky part (near gate edge) of a gate.

- Necking shape always increases I_{off}. Hence, necking should be avoided in line-end shaping. If we cannot avoid necking, the necking location must be away from the channel edge, although it increases the line-end extension length.

7.4 Tradeoff: Design Rule versus Manufacturing Cost versus Electrical Characteristics

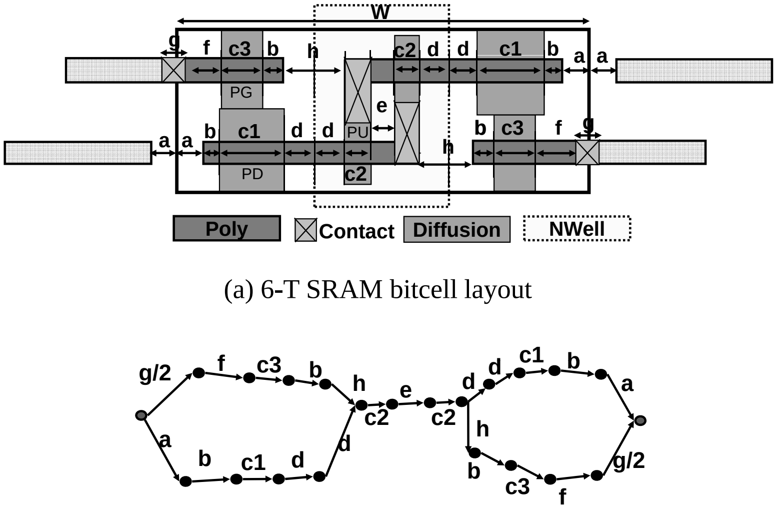
In this section, we present case studies for the tradeoffs between design rule, manufacturing cost and electrical characteristics, using the proposed non-uniform gate models.

7.4.1 Co-Optimization 1: Area versus Leakage Tradeoff

SRAM Bitcell. Figure 7.8(a) shows an example of a 6-T SRAM bitcell layout, while (b) shows the corresponding layout constraint graph that defines the width of the bitcell. In the figure, a is half the line-end gap, b is the length of the line-end extension, and $c1$, $c2$ and $c3$ are the respective widths of pull-down (PD), pull-up (PU) and pass-gate (PG) transistors. Also, d is the space rule between diffusion and N-well, e is the space between diffusion patterns, f is the space between contact and diffusion, g is the width or height of a contact, and h is the space between poly and contacts of a different net. Since the line-end extension (b) occurs twice in the critical path of this width constraint graph, when we reduce the length of the line-end extension by x, the bitcell width decreases by $2x$, and this will reduce the bitcell area.

We evaluate the area and leakage current of a bitcell by changing the sharpness as well as the length of the line-end extension. Table 7.5 shows the design rules for a $45nm$ technology that we use [14], along with the assumed transistor width values. Figure 7.9 shows the tradeoff curve under the given design rules.

The I_{off} value in Figure 7.9 is the total leakage current of all transistors, i.e., two PD, two PG and two PU transistors, in the bitcell. To calculate PU (PMOS) leakage current, we assume that unit-width leakage of the PMOS is half that of NMOS. We also assume that the line-end extension length of PUs is fixed, since it is determined by other



(a) 6-T SRAM bitcell layout

(b) Width-constraint graph

Figure 7.8: SRAM layout and width constraint graph.

Table 7.5: Design rules from the *Nangate 45nm Open Cell Library* [14], and the width of transistors in an SRAM bitcell.

Rule name	Minimum rule
Half line-end gap (a)	50nm
Diffusion-to-N-well space (d)	55nm
Diffusion-to-diffusion space (e)	80nm
Contact-to-diffusion space f	60nm
Contact width (height) (g)	50nm
Poly-to-contact space (h)	90nm

Transistor	Minimum width
Pull-up $(c2)$	60nm
Pull-down $(c1)$	120nm
Pass-gate $(c3)$	60nm

fixed design rules, i.e., e and d, and cannot be reduced further without disconnecting electrodes. Table 7.6 shows SRAM bitcell area reduction due to the line-end extension reduction under given leakage power constraints. ***If we permit a factor of 2 leakage increase (i.e., 100% in Column 5), we can reduce the line-end design rule to approximately*** $40nm \sim 20nm$***, and reduce the bitcell size by*** $10.42\% \sim 16.67\%$***, depending on the superellipse exponent.***

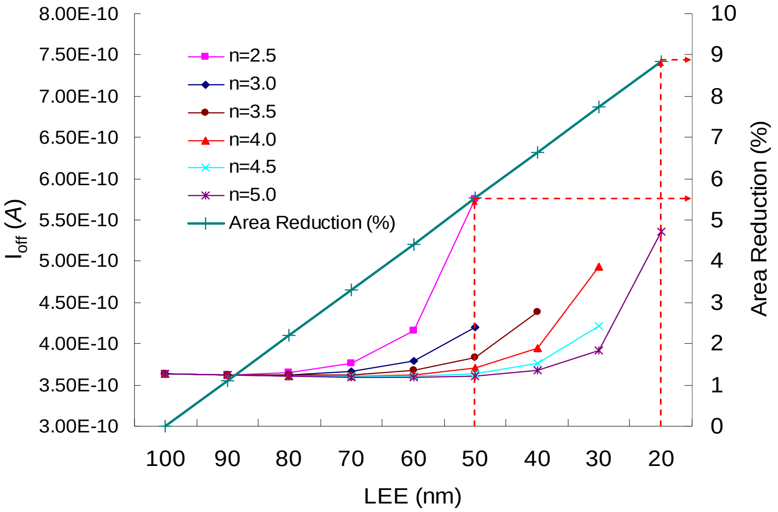
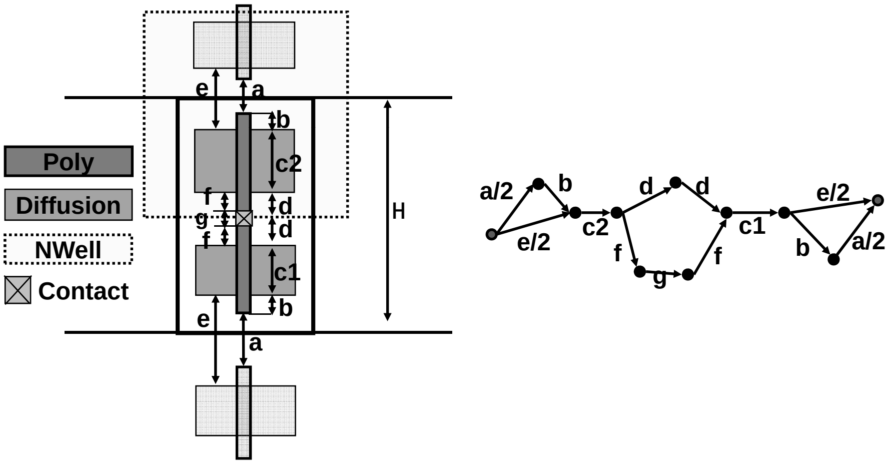


Figure 7.9: Area-leakage tradeoff for an SRAM bitcell.

Table 7.6: SRAM bitcell area reduction (%) with respect to allowed leakage increase (%).

n	Allowed leakage increase (%)					
	10	30	60	100	200	300
2.5	6.25	8.33	10.42	10.42	12.50	12.50
3.0	8.33	10.42	12.50	12.50	14.58	14.58
3.5	10.42	12.50	14.58	14.58	14.58	16.67
4.0	12.50	14.58	14.58	16.67	16.67	16.67
4.5	12.50	14.58	16.67	16.67	16.67	16.67
5.0	14.58	14.58	16.67	16.67	16.67	16.67

Standard Cell Logic. Similar to the SRAM bitcell, we analyze the standard cell logic area and leakage current based on the line-end extension length and the sharpness of tapering. We take an inverter cell as being representative of standard cells. Figure 7.10(a) shows the basic layout structure of a standard inverter cell; (b) shows the corresponding height constraint graph. The notation is the same as that given for the SRAM bitcell except that $c1$ and $c2$ are the gate widths of NMOS and PMOS transistors, respectively. Figure 7.11 shows the tradeoff curve under the given design rules for $45nm$ technology. We assume NMOS and PMOS widths of $400nm$ and $600nm$, respectively, and that unit-width leakage current of PMOS is half that of NMOS. Unlike in the SRAM case, line-end extension length of PMOS devices can also be reduced. Due to the relatively large transistor sizes in a logic cell, impact of line-end extension length is smaller than in a bitcell. Table 7.7 shows standard cell area reduction due to the line-end extension length reduction under given leakage power constraints. In general, each logic cell has its own width but shares a common cell height with all other cells. Hence, the area reduction due to cell height reduction observed in the inverter example applies equally to all standard cells. From Figure 7.11 and Table 7.7, ***if a factor of 2 leakage increase is allowed,*** $9.52\% \sim 10.88\%$ ***of logic area can be reduced by line-end design rule relaxation.***



(a) Standard logic layout (b) Height-constraint graph

Figure 7.10: Inverter cell layout and height constraint graph.

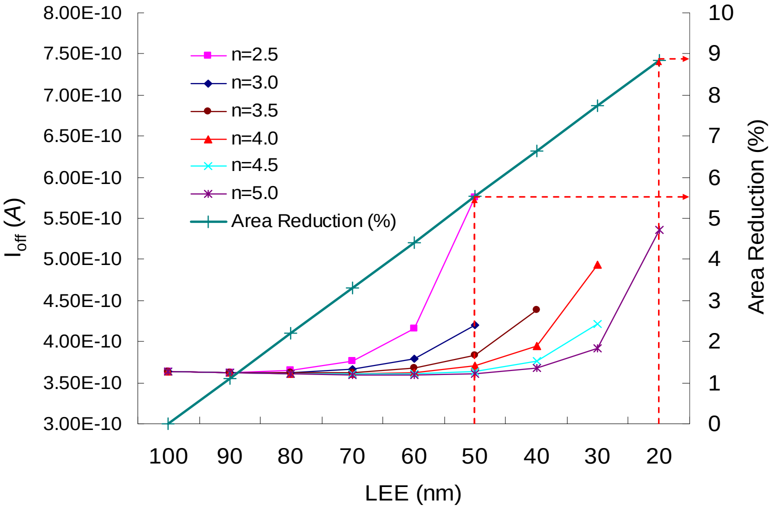


Figure 7.11: Area-leakage tradeoff for a logic cell.

Table 7.7: Logic area reduction (%) under allowed leakage increase (%).

n	Allowed leakage increase (%)					
	10	30	60	100	200	300
2.5	6.80	8.16	8.16	9.52	9.52	9.52
3.0	8.16	9.52	9.52	9.52	10.88	10.88
3.5	9.52	9.52	10.88	10.88	10.88	10.88
4.0	9.52	10.88	10.88	10.88	10.88	10.88
4.5	10.88	10.88	10.88	10.88	10.88	10.88
5.0	10.88	10.88	10.88	10.88	10.88	10.88

7.4.2 Co-Optimization 2: Design-Rule versus OPC/Litho Cost versus Leakage Tradeoff

The proposed electrical models also enable fast analysis of the post-litho line-end shapes, and thus can be used to evaluate various design rules and OPC/litho parameters in terms of the resulting area and leakage current.

Experimental Setup. OPC cost can be measured by the runtime and data size resulting from the number of fragmentations. The following parameters from *Calibre Model-Based OPC User's Manual* [13], shown in Figure 7.12, control the fragmentation of the line-end extension OPC treatment.

- **lineEndLength.** This parameter defines the distance criteria used to determine whether or not a fragment is a line-end. A line-end is defined as an edge that is shorter than or equal to lineEndLength and between two convex corners, each of which is longer than or equal to the lineEndLength parameter. Any line-end edge will be treated differently than others.

- **lineEndAdjDist.** This parameter specifies the distance away from the line-end edge determined by lineEndLength. Part of edges within the distance to the line-end specified by this parameter will be fragmented differently from other parts of the edges.

- **cornedge.** This parameter specifies detailed fragmentation locations via options "**lea** *lead1 ... leadN*". *lead1 ... leadN* specify the fragmentation locations from line-end adjacent convex corners. The values *lead1 ... leadN* are the distances to a vertex from the previous vertex, as shown in Figure 7.12.

We use the following optical models and process corners for OPC and lithography simulation to produce $38nm$ and $52nm$ CD values at the best- and worst-case corners, respectively.

- **Optical model.** We use λ = $193nm$, NA = 1.2, and an annular-type illuminator with 0.7 and 0.5 for sigma and inner-sigma, respectively. We use a constant threshold (CTR) model of 0.25 for both OPC and lithography.

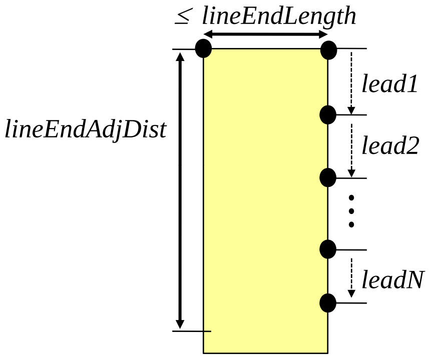


Figure 7.12: OPC parameters for line-end fragmentation.

- **Process corner.** We set +10nm DOF and +2$\%$ higher dose for the best-case corner, and -10nm DOF and -3$\%$ lower dose for the worst-case corner.

We permute the following parameters for OPC/litho simulations.

- **Number of fragmentations** (N_f)**.** This directly affects the cost of OPC and lithography. We evaluate five different numbers (i.e., 0, 1, 2, 3, and 4) of fragmentations with 100nm *lineEndAdjDist*.

- **Fragmentation locations.** We permute all possible fragmentation locations for each number of fragmentations with 10nm minimum fragment length. For N_f = 1, we evaluate ten different fragmentation locations, from 10nm to 100$nm.$ For N_f = 2, we sweep the location of the first fragmentation ($lead1$) from 10nm to 90nm and we sweep the location of the second fragmentation ($lead2$) from 10nm to '100nm -$lead1$' from the first fragmentation location. Similarly, we examine all possible different combinations of fragmentation locations for N_f = 3 or 4. The number of different cases are 1, 10, 45, 120 and 210, for N_f = 0, 1, 2, 3 and 4, respectively.

We implement a layout that contains various cases of line-end extension length (LEE) and line-end gap (LEG) values, and apply different OPC as parameters explained above. The layout contains 100 (10 $\times$ 10) different combinations of line-end extension length and line-end gap. The distance between patterns from different LEE and LEG combinations is approximately 10μm to suppress interference between them. Each pattern consists of two groups of 11 parallel poly lines as shown in Figure 7.13, and the shape of the center line among the 11 lines is checked. The separation distance between the two groups follows specified line-end gap values varying from $10\mu m$ to $100\mu m$. Patterns are designed based on the design rules used in *Nangate 45nm Open Cell library* [14]: poly-to-poly pitch is $190nm$, gate length (CD) is $45nm$, and poly length is determined by the sum of $1200nm$ ($400nm$ NMOS, $600nm$ PMOS, and $200nm$ of diffusion space between NMOS and PMOS) and 2× LEE.

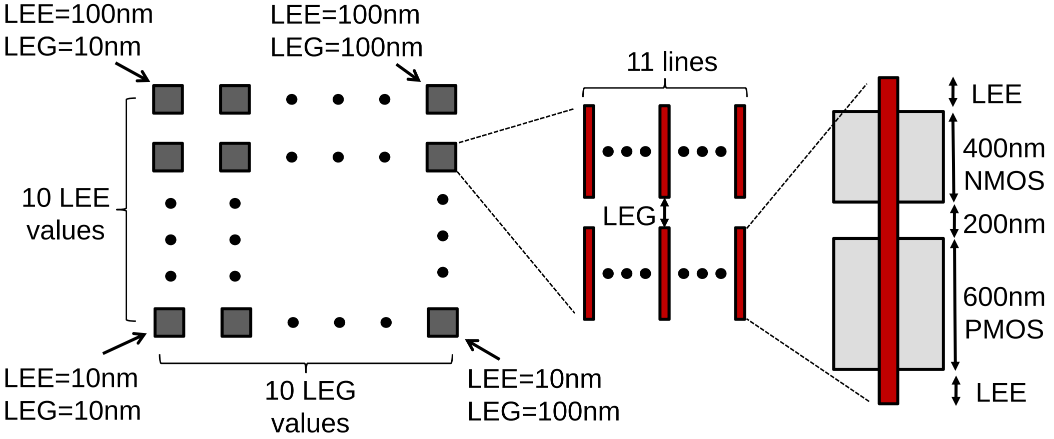


Figure 7.13: Layout of the test patterns for OPC/litho simulation.

We analyze the results from a total of 115,800 cases (= 3 process corners × 100 different design rules × 386 different fragmentations) in this section.

Evaluation of Traditional Line-End Shape Metrics. The first analysis evaluates the electrical characteristics of traditional line-end shape metrics, such as linewidth at the gate edge (LW0) and corner rounding of line-end. LW0 has long served as the most important parameter in the traditional line-end metric. However, LW0 is not a sufficient metric to estimate electrical characteristics of a device. Table 7.8 shows how I_{off} can vary for the same LW0 value, i.e., $45nm$. The large I_{off} variation in Table 7.8 is caused by the different shapes, especially necking location, due to the different design rules and fragmentations in OPC. Figure 7.14 shows the shapes of poly lines corresponding to Cases 2 and 7 in Table 7.8. ***Necking in the channel can significantly increase $\mathbf{I}_{off}$, even when LW0 matches the target linewidth.***

We note that I_{off} of Case 7 is larger than that of Case 2, although Case 7 has higher OPC cost due to the larger number of fragmentations and has better corner round-

Table 7.8: I_{off} variation for the same $45nm$ linewidth at the gate edge (LW0).

Case	Design rule		#fragmentations	LW0	I_{off} (nA)
	LEE (nm)	LEG (nm)			
1	80	30	0	45	151
2	40	80	0	45	115
3	90	100	1	45	153
4	100	90	1	45	164
5	100	80	2	45	144
6	100	80	3	45	152
7	80	100	3	45	191
8	90	90	4	45	162
9	100	100	4	45	121

ing compared to Case 2. In other words, ***less tapering, which is regarded as good in the traditional line-end metric, does not necessarily correspond to better electrical characteristics.***

I_{off} **Variation versus Process Variation.** We also evaluate I_{off} variation at different process corners. Figure 7.15 shows I_{off} variations at best-/nominal-/worst-case process corners with respect to LEG design rules. I_{off} values increase by an order of magnitude from worst-case to nominal-case, and from nominal-case to best-case corner. This I_{off} increase can be explained from Figure 7.16 which shows the litho contours for best-case (red), nominal-case (yellow), and worst-case (green) process corners, for $10nm$, $50nm$ and $100nm$ LEG rules. We can observe that litho contours shrink when the process corner changes from worst-case to best-case, and this results in continuous I_{off} increase.

We observe $2\times$ increase in I_{off} at the best-case corner for $50nm$ LEG rule in Figure 7.15, although nominal- or worst-case corners do not show significant I_{off} variation. Again, even though $100nm$ LEG shows the smallest LW0, $50nm$ LEG results in the largest I_{off}. Ignorable necking for $50nm$ LEG at the nominal- and worst-case corners becomes severe at the best-case corner as shown in Figure 7.16.

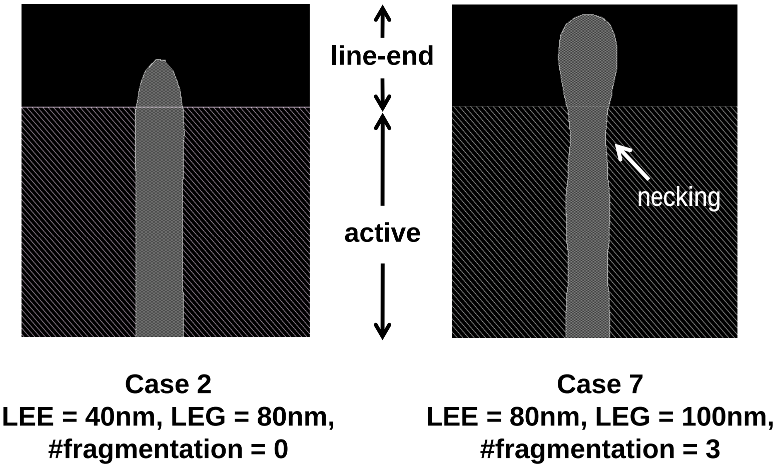


Figure 7.14: Litho images for Cases 2 and 7 in Table 7.8.

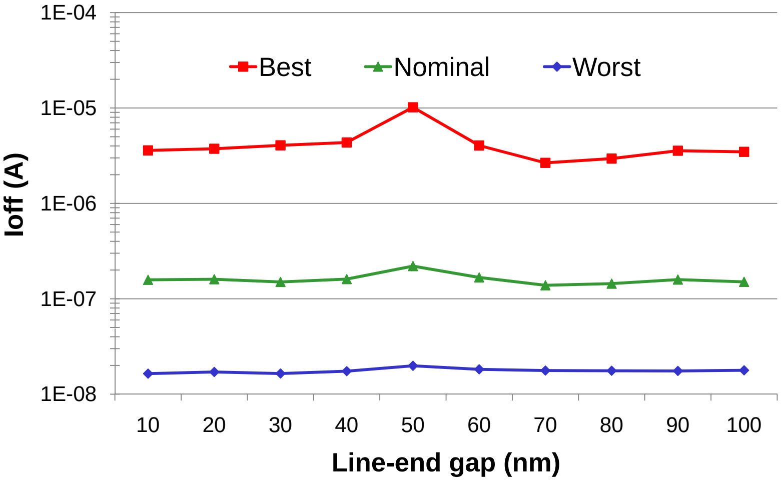


Figure 7.15: I_{off} variations at best-/nominal-/worst-case process corners with respect to LEG design rules.

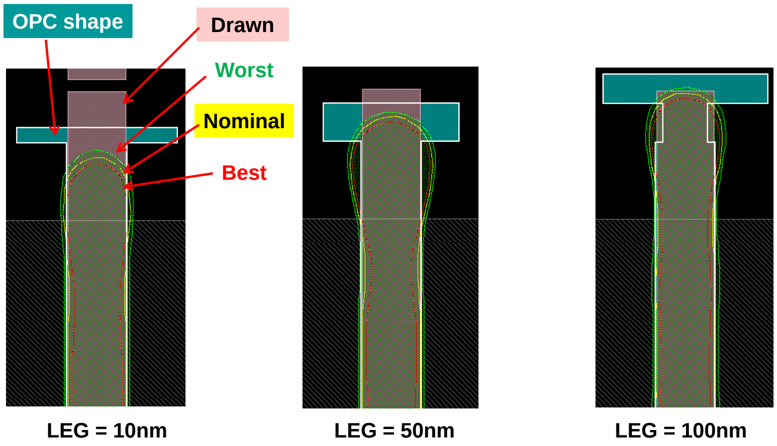


Figure 7.16: Litho contours at best-case (red), nominal-case (yellow), and worst-case (green) corners, for $10nm$, $50nm$ and $100nm$ LEG rules.

Optimal OPC Setup. Traditional line-end metrics do not correctly represent the electrical characteristics of transistors. For better electrical characteristics, we evaluate various OPC parameters, and seek to find the OPC setup that has the best electrical performance, e.g., least I_{off}.

For each number of fragmentations (N_f), we find the optimal fragmentation location that has the minimum I_{off} under given design rules, i.e., LEE = $100nm$ and LEG = $100nm$. Table 7.9 shows the best fragmentation locations – i.e., with smallest I_{off} – for different N_f. We observe that ***a larger number of fragmentations result in smaller*** $\boldsymbol{I_{off}}$ ***for a given design rule.*** We can also observe that ***fragmentations near the gate edge are better to minimize*** $\boldsymbol{I_{off}}$ ***for 100nm LEE and LEG design rules.***

However, a larger number of fragmentations lead to larger OPC runtime (as well as larger post-OPC data) as shown in Figure 7.17. In the experiments for Figure 7.17, we generate a testcase that contains 100K poly lines, and perform OPC/litho simulations by changing the number of fragmentations for only the line-end extension from one to ten. We do not introduce any fragmentation for edges that are not in the line-end extension. From Table 7.9 and Figure 7.17, designers can explicitly trade off OPC cost and I_{off}.

Optimal Design Rules and OPC Setup. Finally, to quantify the cost of design rule parameters, i.e., LEE and LEG, we introduce as a metric, the normalized area of a logic cell, parameterized with LEE and LEG values, as

$$C_{v_1,v_2} = H_{v1,v_2}/H_{100,100}$$

Table 7.9: Best fragmentation locations when I_{off} for LEE = 100nm and LEG = 100nm.

	Best fragmentation locations.				
#Frag.	$lead1$	$lead2$	$lead3$	$lead4$	I_{off}
	(nm)	(nm)	(nm)	(nm)	(nA)
0	-	-	-	-	151
1	80	-	-	-	148
2	90	10	-	-	140
3	70	20	10	-	133
4	60	10	10	20	121

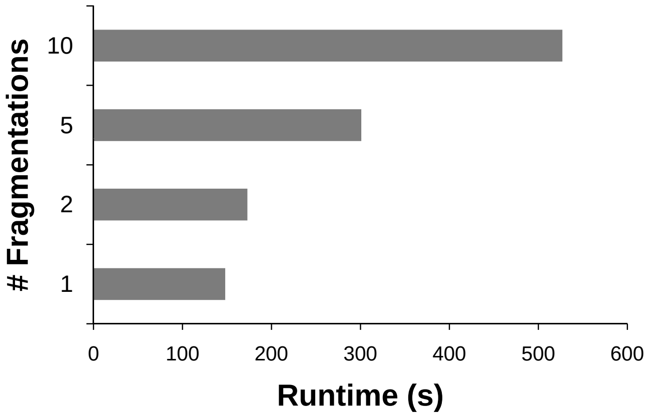


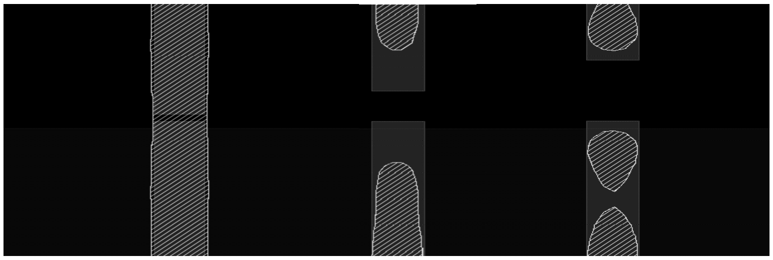
Figure 7.17: OPC/litho simulation runtime due to the number of fragmentations.

where H_{v_1,v_2} represent the height of a single logic cell when LEE and LEG are v_1 and v_2. Logic cell height is a function of LEE and LEG; the height is calculated as a sum of NMOS width, PMOS width, space between NMOS and PMOS, two times LEE, and one LEG. $H_{100,100}$ is 1,500 (= 400 + 600 + 200 + 2×100 + 100) which is used as a reference area value. According to the LEE and LEG ranges in the experiment, C_{v1,v_2} is calculated in Table 7.10. From the table, we can easily obtain the area reduction from LEE and LEG design rule changes.

Table 7.10: Normalized area (%), $C_{LEE,LEG}$ according to LEE and LEG design rules, relative to the area when both LEE and LEG are $100nm$.

	LEG									
LEE	10	20	30	40	50	60	70	80	90	100
100	94	95	95	96	97	97	98	99	99	100
90	93	93	94	95	95	96	97	97	98	99
80	91	92	93	93	94	95	95	96	97	97
70	90	91	91	92	93	93	94	95	95	96
60	89	89	90	91	91	92	93	93	94	95
50	87	88	89	89	90	91	91	92	93	93
40	86	87	87	88	89	89	90	91	91	92
30	85	85	86	87	87	88	89	89	90	91
20	83	84	85	85	86	87	87	88	89	89
10	82	83	83	84	85	85	86	87	87	88

Although we find the best OPC parameters for given LEE and LEG values in the previous section, the best OPC parameters can vary with the applied design rules. We evaluate the best combinations of design rules and OPC parameters. From all the simulation results, we again find the best OPC parameters (i.e., fragmentation options) that result in the smallest I_{off} in any combination of LEE and LEG design rules. Tables 7.11, 7.12, 7.13, 7.14, and 7.15 show arrays of I_{off} values that contain smallest I_{off} value among all different fragmentation locations for different number of fragmentations (N_f). In the tables, 'B', 'S', and 'F' denote bridging, line-end shortening, and broken lines, respectively. 'O' is used when a too-small linewidth that is out of the modeling boundary is introduced in the channel. I_{off} values are obtained at the nominal-case corner, but the best fragmentation locations do not change at worst-case or best-case corners. We do not calculate I_{off} for catastrophic error cases, such as bridging, line-end shortening, and broken lines, when those errors appear in any of the best-/nominal-/worst-case process corners and in any of 11 parallel lines in the test block, as shown in the center of Figure 7.13. Figure 7.18 shows examples of these catastrophic errors. In general, bridging error occurs when LEG is very small and not enough fragmentations are performed. Line-end shortening is mainly due to small LEE. Broken lines are an extreme case of necking which occurs when large hammerhead OPC serifs are generated.[3]



(a) Bridging (b) Line-end shortening (c) Broken

Figure 7.18: Lithographic errors at the line-end.

Important observations are summarized as follows.

- Minimum I_{off} value is 115, 108, 102, 108, and $103nA$, for N_f = 0, 1, 2, 3, and 4, respectively.
- ***Considering OPC cost, larger number of fragmentations does not effectively reduce the I*** $_{off}$***.*** Specifically, when the number of fragmentations is 4 and LEG is $20nm$, I_{off} is 8× the minimum I_{off}. This is due to very narrow tapering as shown in Figure 7.19.
- ***The minimum I*** $_{off}$ ***is found when the number of fragmentations is 2 with*** $lead1$ ***= 10*** nm ***and*** $lead2$ ***= 30*** nm***.***

[3] However, it is difficult to categorize the exact mechanisms of these errors, since the errors do not occur consistently with monotonic variation of OPC/litho and design rule parameters.

Table 7.11: I_{off} (nA) with respect to the LEE and LEG design rules for the best fragmentation location cases that lead to smallest I_{off} for N_f = 0.

	LEG									
LEE	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
100	B	153	156	B	B	B	159	121	141	151
90	B	150	153	B	B	B	155	120	140	150
80	B	148	151	B	B	B	152	120	137	149
70	B	165	154	B	B	B	149	119	134	147
60	B	S	O	B	B	B	147	117	131	144
50	B	S	S	B,S	B,S	B	145	116	128	141
40	B	S	S	B,S	B,S	B,S	145	**115**	125	138
30	B	S	S	B,S	B,S	B,S	O	O	O	O
20	B	S	S	B,S	B,S	B,S	S	S	O	O
10	B	S	S	B,S	B,S	B,S	S	S	S	S

- ***Optimal LEE and LEG design rules corresponding to the minimum I$_{off}$ are 20nm and 70nm, respectively, which can result in 13% area reduction*** according to Table 7.10. However, it may be risky to adopt a design rule that is near values resulting in catastrophic errors. If we add 20nm of margin to the LEE design rule to avoid risky design rules, we still reduce area by around 10%.

- ***Larger LEE and LEG do not always result in smaller I$_{off}$.*** The LEG values that produce smallest I_{off} are 80, 70, 70, 60, and 100nm for N_f = 0, 1, 2, 3, and 4, respectively.

7.5 Conclusions and Research Directions

We have proposed a novel modeling framework to model the electrical impact of line-end shapes. We model a line-end shape by a general superellipse equation. We model the capacitance between the line-end and the gate channel, and derive I_{on} and I_{off}

Table 7.12: I_{off} (nA) with respect to the LEE and LEG design rules for the best fragmentation location cases that lead to smallest I_{off} for N_f = 1 ($lead1$ = 60nm).

	LEG									
LEE	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
100	B	224	220	376	189	181	145	162	164	152
90	B	220	212	340	185	176	146	159	162	153
80	B	216	213	370	180	172	140	156	159	150
70	B	209	207	330	175	168	140	151	156	148
60	B	204	200	353	172	164	136	147	153	145
50	B,S	O	O	310	162	160	132	142	148	142
40	B,S	S	S	347	161	151	**108**	138	144	139
30	B,S	S	S	342	O	O	O	135	O	O
20	B,S	S	S	S	O	O	O	O	O	O
10	B,S	S	S	S	S	S	S	S	S	S

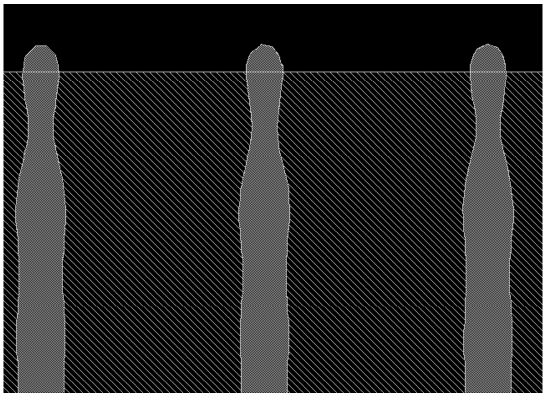


Figure 7.19: Significant tapering when the number of fragmentations is 4 and LEG is 20nm.

Table 7.13: I_{off} (nA) with respect to the LEE and LEG design rules for the best fragmentation location cases that lead to smallest I_{off} for N_f = 2 ($lead1$ = $10nm$, $lead2$ = $30nm$).

	LEG									
LEE	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
100	B	160	150	161	220	167	139	144	159	150
90	B	157	148	159	216	165	136	142	158	156
80	B	152	149	154	208	164	135	138	156	157
70	B	147	145	151	203	161	131	136	154	162
60	B	O	O	O	198	156	128	132	148	158
50	B,S	S	S	S	194	145	123	132	148	161
40	B,S	S	S	S	187	130	114	122	144	149
30	B,S	S	S	S	S	126	109	117	138	145
20	B,S	S	S	S	S	O	**102**	O	130	140
10	B,S	S	S	S	S	S	S	O	S	O

Table 7.14: I_{off} (nA) with respect to the LEE and LEG design rules for the best fragmentation location cases that lead to smallest I_{off} for N_f = 3 ($lead1$ = 10nm, $lead2$ = 10nm, $lead3$ = 20nm).

	LEG									
LEE	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
100	B	155	160	173	181	135	149	152	156	166
90	B	154	157	169	179	135	146	150	167	183
80	B	149	154	165	178	132	145	149	159	191
70	B	145	147	161	173	129	142	147	163	194
60	B	352	186	186	168	125	138	142	160	191
50	B,S	S	S	S	163	119	133	137	155	188
40	B,S	S	S	S	S	116	123	134	146	182
30	B,S	S	S	S	S	110	117	130	142	178
20	B,S	S	S	S	S	**108**	113	122	137	173
10	B,S	S	S	S	S	O	O	O	O	O

Table 7.15: I_{off} (nA) with respect to the LEE and LEG design rules for the best fragmentation location cases that lead to smallest I_{off} for N_f = 4 ($lead1$ = 60nm, $lead2$ = 10nm, $lead3$ = 10nm, $lead4$ = 20nm).

	LEG									
LEE	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
100	B	831	F	F	F	370	207	175	160	121
90	B	821	F	F	F	403	223	181	162	122
80	B	831	F	F	F	412	223	180	162	119
70	B,S	858	F	F	F	416	222	177	163	116
60	B,S	S	F	F	F	416	216	174	160	112
50	B,S	S	F,S	F	F	408	209	170	160	109
40	B,S	S	F,S	F,S	F	404	206	164	157	107
30	B,S	S	F,S	F,S	F	410	207	162	151	**103**
20	B,S	S	F,S	F,S	F,S	O	552	204	155	109
10	B,S	S	F,S	F,S	F,S	S	S	S	S	S

models from it, considering overlay error in the manufacturing process. The proposed model accuracy is within 0.47% and 1.28% for I_{on} and I_{off}, respectively, compared to 3D TCAD simulation. Experimental results show that different line-end extension lengths can affect I_{on} and I_{off} by 4.5% and 30%, respectively, and that different line-end shapes, combined with overlay error, can increase I_{off} by several times compared to the ideal line-end shape. The proposed electrical model enables fast and accurate evaluations of various line-end shapes, given the results of large sized design of experiments. Applying the model to SRAM bitcell and inverter cell layout, we observe that the traditional line-end extension design rule can be reduced further without affecting electrical characteristics of circuits. We also evaluate the tradeoffs among design rules and the resulting area, OPC/litho parameters, and I_{off}. From the analyses, we show the potential for optimal design rules and OPC/litho parameters that can minimize I_{off} and reduce layout area by more than 10%.

Next goals beyond this study seek (1) to find a systematic methodology for small-sized design of experiments to derive the optimal OPC and design rules, and (2) to provide rules of thumb for the optimal line-end shaping, so that designers and lithographers can easily find optimal solutions according to their own OPC/lithography/design rules and device characteristics.

7.6 Acknowledgments

Chapter 7 is in part a reprint of “Electrical Assessment of Lithographic Gate Line-End Patterning”, *SPIE Journal of Microlithography, Microfabrication and Microsystems* 9(2) (2010) and “Electrical Metrics for Lithographic Line-End Tapering”, *Proc. Photomask and Next-Generation Lithography Mask Technology*, 2008. I would like to thank my coauthors Professor Puneet Gupta, Professor Andrew B. Kahng and Dr. Chul-Hong Park.

Chapter 8

Conclusions

Increasing variability in today's manufacturing processes causes parametric yield loss that increases manufacturing cost. In spite of the tremendous effort and enhancement from both manufacturing and design sides, problematic systematic variations still remain uncompensated. For uncompensated variabilities, guardbanding has been the only available knob to trade off design cost and production yield. Overdesign assuming worst-case impact of variations has been widely accepted, although large guardband makes the final chip signoff tougher than it needs to be, incurring significant design turnaround time and cost increase. In addition, while new manufacturing techniques have been adopted to reduce variability (and thus guardband) by improving pattern fidelity in the subwavelength lithography regime, new techniques introduce new variabilities that can again increase guardband. To mitigate the remaining or emerging variabilities, accurate modeling and assessment of the variabilities are essential through detailed analyses of underlying physical mechanisms. Appropriate optimizations in both design and manufacturing must be developed, based on comprehensive understanding of the benefits and costs of such additional measures.

In this thesis, we have first quantified the impact of guardband reduction on design outcomes, and resulting yield and cost, to objectively evaluate the true benefits of various guardband reduction techniques. We have then presented cost-effective guardband reduction techniques for both design and manufacturing. The proposed techniques span multiple stages of design, manufacturing and implementation: (1) from basic circuit elements such as device, interconnect, logic gates and memory bitcells, to high-level design implementation phases such as logic synthesis, placement and routing, and (2) from mask generation and lithography, to post-silicon variation measurement.

The innovative techniques proposed in this thesis can be grouped into three main thrusts: (1) variability modeling and mapping, (2) variation assessment, and (3) variability mitigation.

In the **variability modeling and mapping** thrust, the main outcomes are as follows.

- We have reviewed various variation modeling techniques and proposed a *variation mapping* problem as a new variation modeling technique.
- We have proposed a novel variation mapping framework (based on compressed sensing theory) that reconstructs the details of multiple, simultaneously occurring systematic variation maps from measurements of a small number of naturally-occurring timing paths within the design.

In the **variability assessment** thrust, we have provided quantified analyses of new interconnect and device variations that are emerging with advanced lithography techniques. The major contributions are summarized as follows.

- For BEOL variation, we have developed variation analysis frameworks based on production signoff tools and 3-D TCAD ("technology computer-aided design") tools, considering all possible process options and scenarios.
- Exhaustive studies with the proposed frameworks, from a small representative interconnect structure to chip-level designs, afford new insights to designers and manufacturers regarding how to trade off quality of results versus design and manufacturing costs, across various double patterning process technology options.
- For FEOL variation, we have given both analytic and empirical assessments of the potential impact of DPL on timing analysis error and guardbanding. Using $45nm$ models, we have found that different DPL mask layout solutions can cause $50ps$ skew in clock distribution that is unseen by traditional analyses, and that different mask layouts can also result in 20% or more change in timing path delays.

- Motivated by these observations, we have proposed potential solutions spanning every step of the design implementation process, i.e., bimodal-aware timing analysis, alternate coloring of timing paths, and placement perturbation for coloring-conflict removal.
- We have also shown the manufacturability of IAL through process window analysis with varying design rules.

In the **variability mitigation** thrust, we have presented three distinct approaches to explicitly mitigate variations and enable principled tradeoffs between design cost and yield. The proposed techniques are summarized as follows.

Design-aware manufacturing process optimization:

- We have provided optimal mask strategies considering parametric and defect yields. We integrate mask size-dependent variation and parametric yield models into a cost model that incorporates mask, wafer, and processing costs, along with throughput, yield, and manufacturing volume. This aspect of the thesis also analyzes impact of defects on parametric yield with understanding of design context (i.e., timing and electrical-functional criticality of each pattern in the layout design).
- We have also proposed novel design-aware local optimizations of exposure dose in the photolithography process, to improve timing yield of circuits as well as reduce leakage power.

Manufacturing-aware design optimization:

- We have proposed a cell swapping-based placement optimization algorithm that improves timing yield while also reducing leakage power, given a context of systematic or intentional variations of exposure dose in the manufacturing process.
- To mitigate timing variability in double patterning, we have proposed a new bimodal-aware timing analysis methodology in the context of double patterning lithography; this significantly reduces pessimism of traditional timing analysis

approaches, and provides optimization techniques to improve timing yield of designs. Our bimodality-related research has also devised a novel metric to quantify the delay variation of timing paths due to bimodal distribution of pattern variations, and has developed efficient, optimal cell-based timing-aware DPL mask assignment and detailed placement algorithms.

- We have developed new 1-dimensional regular pitch SRAM bitcell layouts which are amenable to interference-assisted lithography (IAL). In this research, we have devised required design rules for a $32nm$ 6-T bitcell, and designed a family of IAL-friendly bitcell layouts. The quality of the proposed bitcell layouts has been verified through lithography and circuit simulations.

Design-manufacturing co-optimizations:

- We have introduced a novel shape-based (i.e., a general superellipse-based) transistor model, which includes (1) capacitance modeling of line-end extension and consequent current density changes in the transistor channel, and (2) on- and off-current modeling from the new capacitance model. The new transistor model enables fast evaluation of electrical characteristics of complicated post-lithography gate patterns.

- Through assessment of impacts of various layout design rules, mask design optimizations, and lithography process parameters on design area and electrical characteristics, we have derived simple rules of thumb for electrically safe and lithographically robust, yet cost-effective and area-conserving, transistor design rules.

Bibliography

[1] *ASML*, http://www.asml.com/ .

[2] *ASML Dose Mapper*, http://wps2a.semi.org/cms/groups/public/documents/membersonly/van_schoot_presentation.pdf .

[3] *BSIM4.6.4*, http://www-device.eecs.berkeley.edu/∼bsim3/BSIM4/BSIM464/BSIM464_Manual.pdf .

[4] *Cadence Design Exchange Format*, http://openeda.si2.org/projects/lefdef .

[5] *Cadence Library Exchange Format*, http://openeda.si2.org/projects/lefdef .

[6] *Cadence RTL Compiler*, http://www.cadence.com/eu/Pages/rtl_compiler.aspx .

[7] *Cadence SOC Encounter*, http://www.cadence.com/products/di/soc_encounter .

[8] *Cadence Virtuoso Layout Design Environment*, http://www.cadence.com/products/cic/pages .

[9] *GNU OCTAVE*, http://www.gnu.org/software/octave .

[10] *ILOG CPLEX*, http://www.ilog.com/products/cplex .

[11] *International Technology Roadmap for Semiconductors*, http://www.itrs.net .

[12] *Mathworks MATLAB*, http://www.mathworks.com .

[13] *Mentor Graphics Calibre Model-Based OPC*, http://www.mentor.com .

[14] *Nangate 45nm Open Cell Library*, http://www.nangate.com .

[15] *OPENCORES.ORG*, http://www.opencores.org .

[16] *Predictions Software EYES*, http://www.icyield.com/eyes.html .

[17] *Predictive Technology Model*, http://www.eas.asu.edu/∼ptm .

[18] *Qualcomm, Inc.*, http://www.qualcomm.com .

[19] *Sun OpenSPARC Project*, http://www.sun.com/processors/opensparc .

[20] *Synopsys DaVinci*, http://www.synopsys.com/Tools/TCAD/DeviceSimulation/Pages/default.aspx .

[21] *Synopsys Design Compiler*, http://www.sysnopsys.com/Tools/Implementation/RTLSynthesis/Pages/default.aspx .

[22] *Synopsys Hercules*, http://www.synopsys.com/Tools/Implementation/PhysicalVerification/Pages/Hercules.aspx .

[23] *Synopsys HSPICE*, http://www.synopsys.com/Tools/Verification/AMSVerification/CircuitSimulation/HSPICE/Pages/default.aspx .

[24] *Synopsys Liberty File Format*, http://www.synopsys.com/community/interoperability/pages/libertylibmodel.aspx .

[25] *Synopsys PrimeTime*, http://www.synopsys.com/Tools/Implementation/SignOff/Pages/PrimeTime.aspx .

[26] *Synopsys Raphael*, http://www.synopsys.com/Tools/TCAD/InterconnectSimulation/Pages/Raphael.aspx .

[27] *Synopsys Sentaurus*, http://www.synopsys.com/Tools/TCAD/DeviceSimulation/Pages/default.aspx .

[28] *Synopsys STAR-RCXT*, http://www.synopsys.com/Tools/Implementation/SignOff/Pages/Star-RCXT.aspx .

[29] *Zeiss*, http://www.smt.zeiss.com .

[30] *DP-based coloring conflict removal for all testcases (full version)*, http://vlsicad.ucsd.edu/DPLCorr-Results/TableX.pdf .

[31] K. Agarwal and S. Nassif, “Characterizing Process Variation in Nanometer CMOS”, *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2007, pp. 396–399.

[32] C. Babcock, Y. Zou, I. Matthew, D. Dunn, Z. Baum, Z. Zhao and P. LaCour, “Double Dipole RET Investigation for 32 nm Metal Layers”, *Proc. SPIE Photomask Technology*, Vol. 7122, 2008, pp. 71220R-1–71220R-9.

[33] G. E. Bailey, A. Tritchkov, J.-W. Park, L. Hong, P. Xie, V. Wiaux, E. Hendrickx, S. Verhaegen and J. Versluijs, “Double Pattern EDA Solutions for 32nm HP and Beyond”, *Proc. SPIE Design for Manufacturability through Design-Process Integration*, Vol. 6521, 2007, pp. 65211K-1–65211K-12.

[34] V. Bakshi, *EUV Lithography*, SPIE Press, 2008.

[35] A. Balasinski, “Multi-Layer and Multi-Product Masks: Cost Reduction Methodology”, *Proc. SPIE Photomask Technology*, Vol. 5567, 2004, pp. 351–359.

[36] A. Balasinski, J. Cetin, A. B. Kahng and X. Xu, “A Procedure and Program to Calculate Shuttle Mask Advantage”, *Proc. SPIE Photomask Technology*, Vol. 6349, 2006, pp. 63492B-1–63492B-8.

[37] A. Balasinski, L. Karklin and V. Axelrad, “Impact of Subwavelength CD Tolerance on Device Performance”, *Proc. SPIE Design, Process Integration and Characterization for Microelectronics*, Vol. 4692, 2002, pp. 361–368.

[38] P. Bastani, N. Callegari, L.-C. Wang and M. S. Abadir, “Statistical Diagnosis of Unmodeled Systematic Timing Effects”, *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2008, pp. 355–360.

[39] G. Ben-Zvi, E. Zait, V. Dmitriev, S. Labovitz, E. Graitzer, K. Böhm, R. Birkner and T. Scherulebl, “Mask CD Control (CDC) Using AIMS as the CD Metrology Data Source”, *Proc. SPIE Photomask and Next-Generation Lithography Mask Technology XV*, Vol. 7028, 2008, pp. 70281C-1–70281C-9.

[40] S. Bhardwaj, Y. Cao and S. Vrudhula, “Statistical Leakage Minimization Through Joint Selection of Gate Sizes, Gate Lengths and Threshold Voltage”, *Proc. IEEE Asia and South Pacific Design Automation Conference*, 2006, pp. 953–958.

[41] A. M. Biswas, J. Li, J. A. Hiserote and L. S. Melvin III, “Extension of 193nm Dry Lithography to 45-nm Half-Pitch Node: Double Exposure and Double Processing Technique”, *Proc. SPIE Photomask Technology*, Vol. 6349, pp. 63491P-1–63491P-9.

[42] Y. Borodovsky, “Impact of Local Partial Coherence Variations on Exposure Tool Performance”, *Proc. SPIE Optical/Laser Microlithography VIII*, Vol. 2440, 1995, pp. 750–770.

[43] T. A. Brunner, “Impact of Lens Aberrations on Optical Lithography”, *IBM Journal of Research and Development* 41(1-2) (1997), pp. 57–67.

[44] J. Burns and M. Abbas, “EUV Mask Defect Mitigation Through Pattern Placement”, *Proc. SPIE Photomask Technology*, Vol. 7823, 2010, pp. 782340-1–782340-5.

[45] S. M. Burns, M. Ketkar, N. Menezes, K. A. Bowman, J. W. Tschanz and V. De, “Comparative Analysis of Conventional and Statistical Design Techniques”, *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2007, pp. 238–243.

[46] M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing: for Digital, Memory and Mixed-Signal VLSI Circuits*, Kluwer Academic Publishers, 2000.

[47] N. Callegari, L.-C. Wang and P. Bastani, "Speedpath Analysis Based on Hypothesis Pruning and Ranking", *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2009, pp. 346–351.

[48] E. Candes, "Compressive Sampling", *Proc. International Congress of Mathematicians*, 2006, pp. 1433–1452.

[49] K. Cao, S. Dobre and J. Hu, "Standard Cell Characterization Considering Lithography Induced Variations", *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2006, pp. 801–804.

[50] G. Capetti, P. Cantu, E. Galassini, A. V. Pret, C. Turco, A. Vaccaro, P. Rigolli, F. D'Angelo and G. Cotti, "Sub k1 = 0.25 Lithography with Double Patterning Technique for 45nm Technology Node Flash Memory Devices at λ = 193nm", *Proc. SPIE Optical Microlithography XX*, Vol. 6520, 2007, pp. 65202K-1–65202K-12.

[51] H. Chang and S. S. Sapatnekar, "Statistical Timing Analysis under Spatial Correlations", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 24(9) (2005), pp. 1467–1482.

[52] C.-P. Chen, C. C. N. Chu and D. F. Wong, "Fast and Exact Simultaneous Gate and Wire Sizing by Lagrangian Relaxation", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 18(7) (1999), pp. 1014–1025.

[53] J.-H. Chen, L.-J. Chen, T.-Y. Fang, T.-C. Fu, L.-H. Shiu, Y.-T. Huang, N. Chen, D.-C. Oweyang, M.-C. Wu, S.-C. Wang, J. C. Lin, C.-K. Chen, W.-M. Chen, T.-S. Gau, B. J. Lin, R. Moerman, W. G. van Ansem, E. van der Heijden, F. de Jong, D. Oorschot, H. Boom, M. Hoogendorp, C. Wagner and B. Koek, "Characterization of ArF Immersion Process for Production", *Proc. SPIE Optical Microlithography XVIII*, Vol. 5754, 2005, pp. 13–22.

[54] J. F. Chen, T. Laidig, K. E. Wampler and R. Caldwell, "Optical Proximity Correction for Intermediate-Pitch Features using Sub-Resolution Scattering Bars", *Journal of Vacuum Science and Technology B: Microelectronics and Nanometer Structures* 15(6) (1997), pp. 2426–2433.

[55] K.-J. R. Chen, W.-S. Huang, W.-K. Li and P. R. Varanasi, "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.

[56] L. Cheng, P. Gupta, C. Spanos, K. Qian and L. He, "Physically Justifiable Die-Level Modeling of Spatial Variation in View of Systematic Across Wafer Variability", *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2009, pp. 104–109.

[57] T.-B. Chiou, R. Socha, H.-Y. Kang, A. C. Chen, S. Hsu, H. Chen and L. Chen, "Full-Chip Pitch/Pattern Splitting for Lithography and Spacer Double Patterning Technologies", *Proc. SPIE Lithography Asia*, Vol. 7140, 2008, 71401Z-1–71401Z-12.

[58] M. Cho, Y. Ban and D. Z. Pan, "Double Patterning Technology Friendly Detailed Routing", *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2008, pp. 506–511.

[59] C. H. Clifford and A. R. Neureuther, "Smoothing Based Model for Images of Isolated Buried EUV Multilayer Defects", *Proc. SPIE Emerging Lithography Technologies XII*, Vol. 6921, 2008, pp. 692119-1–692119-10.

[60] B. Cline, K. Chopra, D. Blaauw and Y. Cao, "Analysis and Modeling of CD Variation for Statistical Static Timing", *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2006, pp. 60–66.

[61] N. B. Cobb, A. Zakhor, M. Reihani, F. Jahansooz and V. N. Raghavan, "Experimental Results on Optical Proximity Correction with Variable-Threshold Resist Model", *Proc. SPIE Optical Microlithography X*, Vol. 3051, 1997, pp. 458–468.

[62] D. Donoho, "Compressed Sensing", *IEEE Transactions on Information Theory* 52(4) (2006), pp. 1289–1306.

[63] M. Drapeau, V. Wiaux, E. Hendrickx, S. Verhaegen and T. Machida, "Double Patterning Design Split Implementation and Validation for the 32nm Node", *Proc. SPIE Design for Manufacturability through Design-Process Integration*, Vol. 6521, 2007, pp. 652109-1–652109-15.

[64] M. Dusa, J. Quaedackers, O. F. A. Larsen, J. Meessen, E. van der Heijden, G. Dicker, O. Wismans, P. de Haas, K. van Ingen Schenau, J. Finders, B. Vleeming, G. Storms, P. Jaenen, S. Cheng and M. Maenhoudt, "Pitch Doubling Through Dual-Patterning Lithography: Challenges in Integration and Litho Budgets", *Proc. SPIE Optical Microlithography XX*, Vol. 6520, 2007, pp. 65200G-1–65200G-10.

[65] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey and C. J. Spanos, "Modeling Within-Field Gate Length Spatial Variation for Process-Design Co-Optimization", *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing III*, Vol. 5756, 2005, pp. 178–188.

[66] M. Fritze, T. M. Bloomstein, B. Tyrrell, T. H. Fedynyshyn, N. N. Efremow, D. E. Hardy, S. Cann, D. Lennon, S. Spector, M. Rothschild and P. Brooker, "Hybrid Optical Maskless Lithography: Scaling Beyond the 45nm Node", *Journal of Vacuum Science and Technology B: Microelectronics and Nanometer Structures* 23(6) (2005), pp. 2743–2748.

[67] M. Fritze, B. Tyrrell, T. Fedynyshyn, M. Rothschild and P. Brooker, "High-Throughput Hybrid Optical Maskless Lithography: All-Optical 32-nm Node Imaging", *Proc. SPIE Emerging Lithographic Technologies IX*, Vol. 5751, 2005, pp. 1058–1068.

[68] M. Garg, A. Kumar, J. van Wingerden and L. Le Cam, "Litho-Driven Layouts for Reducing Performance Variability", *Proc. IEEE International Symposium on Circuits and Systems*, 2005, pp. 3551–3554.

[69] R. S. Ghaida and P. Gupta, "Design-Overlay Interactions in Metal Double Patterning", *Proc. SPIE Design for Manufacturability through Design-Process Integration III*, Vol. 7275, 2009, pp. 727514-1–727514-10.

[70] R. Giacomini and J. A. Martino, "Modeling Silicon on Insulator MOS Transistors with Nonrectangular-Gate Layouts", *Journal of the Electrochemical Society* 153(3) (2006), pp. G218–G222.

[71] W. B. Glendinning and J. N. Helbert, *Handbook of VLSI Microlithography: Principles, Technology, and Applications*, 2nd ed., Noyes Publications; William Andrew Publishing, 2001.

[72] B. J. Grenon, "Mask Costs, A New Look", *Proc. SPIE European Mask and Lithography Conference*, Vol. 6281, 2006, pp. 628101-1–628101-7.

[73] E. Grossar, M. Stucchi, K. Maex and W. Dehaene, "Read Stability and Write-Ability Analysis of SRAM Cells for Nanometer Technologies", *IEEE Journal of Solid-State Circuits* 41(11) (2006), pp. 2577–2588.

[74] L. J. Guo, "Recent Progress in Nanoimprint Technology and Its Applications", *Journal of Physics D: Applied Physics* 37(11) (2004), pp. 123–141.

[75] P. Gupta and F.-L. Heng, "Toward a Systematic-Variation Aware Timing Methodology", *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2004, pp. 321–326.

[76] M. Gupta, K. Jeong and A. B. Kahng, "Timing Yield-Aware Color Reassignment and Detailed Placement Perturbation for Double Patterning Lithography", *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2009, pp. 607–614.

[77] P. Gupta, K. Jeong, A. B. Kahng and C.-H. Park, "Electrical Metrics for Lithographic Line-End Tapering", *Proc. Photomask and Next-Generation Lithography Mask Technology XV*, Vol. 7028, 2008, pp. 70283A-1–70283A-12.

[78] P. Gupta and A. B. Kahng, "Manufacturing-Aware Physical Design", *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2003, pp. 681–687.

[79] P. Gupta, A. B. Kahng, Y. Kim, S. Shah and D. Sylvester, “Modeling of Non-Uniform Device Geometries for Post-Lithography Circuit Analysis”, *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing IV*, Vol. 6156, 2006, pp. 61560U-1–61560U-10.

[80] P. Gupta, A. B. Kahng, Y. Kim, S. Shah and D. Sylvester, “Investigation of Diffusion Rounding for Post-Lithography Analysis”, *Proc. IEEE Asia and South Pacific Design Automation Conference*, 2008, pp. 480–485.

[81] P. Gupta, A. B. Kahng, Y. Kim and D. Sylvester, “Self-Compensating Design for Focus Variation”, *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2005, pp. 365–368.

[82] P. Gupta, A. B. Kahng, S. Nakagawa, S. Shah and P. Sharma, “Lithography Simulation-Based Full-Chip Design Analyses”, *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing IV*, Vol. 6156, 2006, pp. 61560T-1–61560T-8.

[83] P. Gupta, A. B. Kahng and C.-H. Park, “Detailed Placement for Improved Depth of Focus and CD Control”, *Proc. IEEE Asia and South Pacific Design Automation Conference*, 2005, pp. 343–348.

[84] P. Gupta, A. B. Kahng and C.-H. Park, “Detailed Placement for Enhanced Control of Resist and Etch CDs”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 26(12) (2007), pp. 2144–2157.

[85] P. Gupta, A. B. Kahng, P. Sharma and D. Sylvester, “Gate-Length Biasing for Runtime-Leakage Control”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 25(8) (2006), pp. 1475–1485.

[86] B. Hargreaves, H. Hult and S. Reda, “Within-Die Process Variations: How Accurately Can They Be Statistically Modeled?” *Proc. IEEE Asia and South Pacific Design Automation Conference*, 2008, pp. 524–530.

[87] C. Hedlund, H.-O. Blom and S. Berg, “Microloading Effect in Reactive Ion Etching”, *Journal of Vacuum Science and Technology* 12(4) (1994), pp. 1962–1965.

[88] H. T. Heineken and W. Maly, “Interconnect Yield Model for Manufacturability Prediction in Synthesis of Standard Cell Based Designs”, *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 1996, pp. 368–373.

[89] F.-L. Heng, J.-F. Lee and P. Gupta, “Toward Through-Process Layout Quality Metrics”, *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing III*, Vol. 5756, 2005, pp. 161–167.

[90] D. van den Heuvel, R. Jonckheere, J. Magana, T. Abe, T. Bret, E. Hendrickx, S. Cheng and K. Ronse, "Natural EUV Mask Blank Defects: Evidence, Timely Detection, Analysis and Outlook", *Proc. SPIE Photomask Technology*, Vol. 7823, 2010, pp. 78231T-1–78231T-12.

[91] M. Hori, T. Nagai, A. Nanamura, T. Abe, G. Wakamatsu, T. Kakizawa, Y, Anno, M. Sugiura, S. Kusumoto, Y. Yamaguchi and T. Shimokawa, "Sub-40-nm Half-Pitch Double Patterning with Resist Freezing Process", *Proc. SPIE Advances in Resist Materials and Processing Technology XXV*, Vol. 6923, 2008, pp. 69230H-1–69230H-8.

[92] S. D. Hsu, J. F. Chen, N. Cororan, W. T. Knose, R. J. Socha, D. J. van den Broeke, T. L. Laidig, K. E. Wampler, X. Shi, M. Hsu, M. Eurlings, J. Finders, T.-B. Chiou, W. Conley, Y. W. Hsieh, S. Tuan and F. Hsieh, "65-nm Full-Chip Implementation Using Double Dipole Lithography", *Proc. SPIE Optical Microlithography XVI*, Vol. 5040, 2003, pp. 215–231.

[93] B. Hwang, N. Lim, J.-H. Park, S. Jin, M. Kim, J. Jung, B, Kwon, J. Hong, J. Han, D. Kwak, J. Park, J.-D. Choi and W.-S. Lee, "Development of 38nm Bit-Lines Using Copper Damascene Process for 64-Giga Bits NAND Flash", *Proc. IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, 2008, pp. 49–51.

[94] N. Jeewakhan, N. Shamma, S.-J. Choi, R. Alvarez, D. H. Son, M. Nakamura, V. Pici, J. Schreiber, W.-S. Tzeng, S. Ang and D. Park, "Application of Dosemapper for 65-nm Gate CD Control: Strategies and Results", *Proc. SPIE Photomask Technology*, Vol. 6349, 2006, pp. 63490G-1–63490G-11.

[95] K. Jeong and A. B. Kahng, "Timing Analysis and Optimization Implications of Bimodal CD Distribution in Double Patterning Lithography", *Proc. IEEE Asia and South Pacific Design Automation Conference*, 2009, pp. 486–491.

[96] K. Jeong, A. B. Kahng, C.-H. Park and H. Yao, "Dose Map and Placement Co-Optimization for Timing Yield Enhancement and Leakage Power Reduction", *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2008, pp. 516–521.

[97] K. Jeong, A. B. Kahng and K. Samadi, "Quantified Impacts of Guardband Reduction on Design Process Outcomes", *Proc. International Symposium on Quality Electronic Design*, 2008, pp. 790–797.

[98] K. Jeong, A. B. Kahng and R. O. Topaloglu, "Is Overlay Error More Important Than Interconnect Variations in Double Patterning?", *Proc. International Workshop on System Level Interconnect Prediction*, 2009, pp. 3–10.

[99] T. Jhaveri, L. Pileggi, V. Rovner and A. J. Strojwas, “Maximization of Layout Printability/Manufacturability by Extreme Layout Regularity”, *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing IV*, Vol. 6156, 2006, pp. 615609-1–615609-15.

[100] A. B. Kahng, “Key Directions and a Roadmap for Electrical Design for Manufacturability”, *Proc. European Solid-State Device Research Conference*, 2007, pp. 83–88.

[101] A. B. Kahng and S. Mantik, “Measurement of Inherent Noise in EDA Tools”, *Proc. International Symposium on Quality Electronic Design*, 2002, pp. 206–211.

[102] A. B. Kahng and C.-H. Park, “Auxiliary Pattern for Cell-Based OPC”, *Proc. SPIE Photomask Technology*, Vol. 6349, 2006, pp. 63494S-1–63494S-10.

[103] A. B. Kahng, C.-H. Park, X. Xu and H. Yao, “Layout Decomposition for Double Patterning Lithography”, *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2008, pp. 465–472.

[104] A. B. Kahng, P. Sharma and R. O. Topaloglu, “Chip Optimization Through STI-Stress-Aware Placement Perturbations and Fill Insertion”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 27(7) (2008), pp. 1241-1252.

[105] A. B. Kahng and R. O. Topaloglu, “Generation of Design Guarantees for Interconnect Matching”, *Proc. International Workshop on System Level Interconnect Prediction*, 2006, pp. 29–34.

[106] K. Kamon, T. Miyamoto, Y. Myoi, M. Fujinaga, H. Nagata and M. Tanaka, “Photolithography System Using Modified Illumination”, *Japanese Journal of Applied Physics* 32(1R) (1993), pp. 239–243.

[107] S.-D. Kim, H. Wada and J. C. S. Woo, “TCAD-Based Statistical Analysis and Modeling of Gate Line-Edge Roughness Effect on Nanoscale MOS Transistor Performance Scaling”, *IEEE Transactions on Semiconductor Manufacturing* 17(2) (2004), pp. 192–200.

[108] S.-M. Kim, S.-Y. Koo, J.-S. Choi, Y.-S. Hwang, J.-W. Park, E.-K. Kang, C.-M. Lim, S.-C. Moon and J.-W. Kim, “Issues and Challenges of Double Patterning Lithography in DRAM”, *Proc. SPIE Optical Microlithography XX*, Vol. 6520, 2007, pp. 65200H-1–65200H-7.

[109] K. Koike, K. Nakayama, K. Ogawa and H. Ohnuma, “Optimization of Layout Design and OPC by Using Estimation of Transistor Properties”, *Proc. SPIE Photomask and Next-Generation Lithography Mask Technology XIII*, Vol. 6283, 2006, pp. 62830O-1–62830O-11.

[110] A. Koshiishi, Y. Araki, S. Himori and T. Iijima, “Investigation of Etch Rate Uniformity of 60 MHz Plasma Etching Equipment”, *Japanese Journal of Applied Physics* 40(1-11) (2001), pp. 6613–6618.

[111] F. Koushanfar, P. Boufounos and D. Shamsi, “Post-Silicon Timing Characterization by Compressed Sensing”, *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2008, pp. 185–189.

[112] R. Kumar, *Fabless Semiconductor Implementation*, 1st ed., McGraw-Hill Professional Publishing, 2008.

[113] H. van der Laan, R. Carpaij, J. Krist, O. Noordman, Y. van Dommelen, J. van Schoot, F. Blok, C. van Os, S. Stegeman, T. Hoogenboom, C. Hickman, E. Byers and T. Gugel, “Etch, Reticle, and Track CD Fingerprint Correction with Local Dose Compensation”, *Proc. SPIE Data Analysis and Modeling for Process Control II*, Vol. 5755, 2005, pp. 107–118.

[114] D. Laidler, “Identifying Sources of Overlay Error in FinFET Technology”, *Proc. SPIE Metrology, Inspection, and Process Control for Microlithography XIX*, Vol. 5752, 2005, pp. 80–90.

[115] D. Laidler, P. Leray, K. D’Havé and S. Cheng, “Sources of Overlay Error in Double Patterning Integration Schemes”, *Proc. SPIE Metrology, Inspection, and Process Control for Microlithography XXII*, Vol. 6922, 2008, pp. 69221E-1–69221E-11.

[116] M. Lavin, F.-L. Heng and G. Northrop, “Backend CAD Flows for Restrictive Design Rules”, *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2004, pp. 739–746.

[117] M. D. Levenson, N. S. Viswanathan and R. A. Sympson, “Improving Resolution in Photolithography with a Phase-Shifting Mask”, *IEEE Transactions on Electronic Devices* 29(12) (1982), pp. 1812–1846.

[118] H. J. Levinson, *Principles of Lithography*, 2nd ed., SPIE Press, 2005.

[119] X. Li, R. R. Rutenbar and R. D. Blanton, “Virtual Probe: A Statistically Optimal Framework for Minimum Cost Silicon Characterization of Nanoscale Integrated Circuits”, *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2009, pp. 433–440.

[120] L. Liebmann, D. Maynard, K. McCullen, N. Seong, E. Buturla, M. Lavin and J. Hibbeler, “Integrating DFM Components Into a Cohesive Design-To-Silicon Solution”, *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing III*, Vol. 5756, 2005, pp. 1–12.

[121] B. J. Lin, “Immersion Lithography and Its Impact on Semiconductor Manufacturing”, *Journal of Microlithography, Microfabrication, Microsystems* 3(3) (2004), pp. 377–396.

[122] B. J. Lin, “The Attenuated Phase-Shifting Mask”, *Journal of Solid State Technology* 35(1) (1992), pp. 43–47.

[123] Z. J. Lin and C. J. Spanos, “Sensitivity Study of Interconnect Variation Using Statistical Experimental Design”, *Proc. International Workshop on Statistical Metrology*, 1998, pp. 68–71.

[124] Y. Liu, L. T. Pileggi and A. J. Strojwas, “Model Order-Reduction of RC(L) Interconnect Including Variational Analysis”, *Proc. ACM/EDAC/IEEE Design Automation Conference*, 1999, pp. 201–206.

[125] Q. Liu and S. S. Sapatnekar, “Synthesizing a Representative Critical Path for Post-Silicon Delay Prediction”, *Proc. ACM International Symposium on Physical Design*, 2009, pp. 183–190.

[126] Q. Liu and S. S. Sapatnekar, “Capturing Post-Silicon Variations Using a Representative Critical Path”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29(2) (2010), pp. 211–222.

[127] N. Lu, “Statistical and Corner Modeling of Interconnect Resistance and Capacitance”, *Proc. IEEE Custom Integrated Circuits Conference*, 2006, pp. 853–856.

[128] C. A. Mack, *Fundamental Principles of Optical Lithography: The Science of Microfabrication*, John Wiley & Sons, 2007.

[129] M. Maenhoudt, J. Versluijs, H. Struyf, J. van Olmen and M. van Hove, “Double Patterning Scheme for Sub-0.25 k1 Single Damascene Structures at NA=0.75, λ=193nm”, *Proc. SPIE Optical Microlithography XVIII*, Vol. 5754, 2005, pp. 1508–1518.

[130] W. Maly, Y.-W. Lin and M. Marek-Sadowska, “OPC-Free and Minimally Irregular IC Design Style”, *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2007, pp. 954–957.

[131] V. Mehrotra, S. Nassif, D. Boning and J. Chung, “Modeling the Effects of Manufacturing Variation on High-Speed Microprocessor Interconnect Performance”, *Proc. IEEE International Electron Devices Meeting*, 1998, pp. 767–770.

[132] H. D. Mogal, H. Qian, S. S. Sapatnekar and K. Bazargan, “Fast and Accurate Statistical Criticality Computation under Process Variations”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 28(3) (2009), pp. 350–363.

[133] F. Mosteller and J. W. Tukey, *Handbook of Social Psychology*, Addison-Wesley, 1968.

[134] F. N. Najm, “On the Need for Statistical Timing Analysis”, *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2005, pp. 764–765.

[135] U. Narasimha, B. Abraham and N. S. Nagaraj, “Statistical Analysis of Capacitance Coupling Effects on Delay and Noise”, *Proc. International Symposium on Quality Electronic Design*, 2006, pp. 795–800.

[136] S. Nassif, “Modeling and Forecasting of Manufacturing Variations”, *Proc. International Workshop on Statistical Metrology*, 2000, pp. 2–10.

[137] T. Ogawa, M. Uematsu, T. Ishimaru, M. Kimura and T. Tsumori, “Effective Light Source Optimization with the Modified Beam for the Depth-of-Focus Enhancements”, *Proc. SPIE Optical Microlithography VII*, Vol. 2197, 1994, pp. 19–30.

[138] S. Ohkawa, M. Aoki and H. Masuda, “Analysis and Characterization of Device Variations in an LSI Chip Using an Integrated Device Matrix Array”, *IEEE Transactions on Semiconductor Manufacturing* 17(2) (2004), pp. 155–165.

[139] U. Okoroanyanwu, A. Tchikoulaeva, P. Ackmann, O. Woord, B. La Fontaine, K. Bubke, C. Holfeld, J. H. Peters, S. Kini, S. Watson, I. Lee, B. Mu, P. Lim, S. Raghunathan and C. Boye, “Assessing EUV Mask Defectivity”, *Proc. SPIE Extreme Ultraviolet Lithography*, Vol. 7637, 2010, pp. 76360J-1–76360J-12.

[140] O. W. Otto, J. G. Garofalo, K. K. Low, C.-M. Yuan, R. C. Henderson, C. Pierrat, R. L. Kostelak, S. Vaidya and P. K. Vasudev, “Automated Optical Proximity Correction: A Rules-Based Approach”, *Proc. SPIE Optical/Laser Microlithography VII*, Vol. 2197, 1994, pp. 278–293.

[141] S. Owa and H. Nagasaka, “Immersion Lithography: Its Potential Performance and Issues”, *Proc. SPIE Optical Microlithography XVI*, Vol. 5040, 2003, pp. 724–733.

[142] C. Pacha, M. Bach, K. von Arnim, R. Brederlow, D. Schmitt-Lansiedel, P. Seegebrecht, J. Berthold and R. Thewes, “Impact of STI-Induced Stress, Inverse Narrow Width Effect and Statistical Vth Variations on Leakage Current in 120nm CMOS”, *Proc. European Solid-State Device Research Conference*, 2004, pp. 397–400.

[143] R. C. Pack, V. Axelrad, A. Shibkov, V. V. Boksha, J. A. Huckabay, R. Salik, W. Staud, R. Wang and W. D. Grobman, “Physical and Timing Verification of Subwavelength-Scale Designs: I. Lithography Impact on MOSFETs”, *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing*, Vol. 5042, 2003, pp. 51–62.

[144] J. S. Petersen, M. J. Maslow and R. T. Greenway, "An Integrated Imaging System for the 45-nm Technology Node Contact Holes Using Polarized OAI, Immersion Weak PSM and Negative Resists", *Proc. SPIE Optical Microlithography XVIII*, Vol. 5754, 2005, pp. 488–497.

[145] I. Pollentier, S. Y. Cheng, B. Baudemprez, D. Laidler, Y. van Dommelen, R. Carpaij, J. Yu, J. Uchida, A. Viswanathan, D. Chin, K. Barry and N. Jakatdar, "In-Line Lithography Cluster Monitoring and Control Using Integrated Scatterometry", *Proc. SPIE Data Analysis and Modeling for Process Control*, Vol. 5378, 2004, pp. 105–115.

[146] W. J. Poppe, L. Capodieci, J. Wu and A. Neureuther, "From Poly Line to Transistor: Building BSIM Models for Non-Rectangular Transistors", *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing IV*, Vol. 6156, 2006, pp. 61560P-1–61560P-9.

[147] D. Pramanik, H. Kamberian, C. Progler, M. Sanie and D. Pinto, "Cost Effective Strategies for ASIC Masks", *Proc. SPIE Cost and Performance in Integrated Circuit Creation*, Vol. 5043, 2003, pp. 142–152.

[148] C. Progler, *Personal Communication*, 2011.

[149] A. Rastegar, "Overcoming Mask Blank Defects in EUV Lithography", *SPIE Newsroom*, 2009, http://spie.org/x34659.xml?pf=true&ArticleID=x34659 .

[150] S. Reda and S. Nassif, "Analyzing the Impact of Process Variations on Parametric Measurements: Novel Models and Applications", *Proc. DATE Test for Variability, Reliability and Circuit Marginality*, 2009, pp. 375–380.

[151] D. J. Resnick, W. J. Dauksher, D. P. Mancini, K. J. Nordquist, T. C. Bailey, S. C. Johnson, N. A. Stacey, J. G. Ekerdt, C. G. Willson, S. V. Sreenivasan and N. E. Schumaker, "Imprint Lithography: Lab Curiosity or the Real NGL", *Proc. SPIE Emerging Lithographic Technologies VII*, Vol. 5037, 2003, pp. 12–23.

[152] P. Rigolli, C. Turco, U. Iessi, G. Capetti and P. Canestrari, "Double Patterning Overlay Budget for 45nm Technology Node Single and Double Mask Approach", *Journal of Vacuum Science and Technology B: Microelectronics and Nanometer Structures* 25(6) (2007), pp. 2461–2465.

[153] C. Runge, "Über Empirische Funktionen und Die Interpolation Zwischen Äquidistanten Ordinaten", *Zeitschrift für Mathematik and Physik* 46 (1901), pp. 224–243.

[154] L. Scheffer, "Why Are Timing Estimates So Uncertain? What Could We Do About This?", *ACM International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems*, 2006, http://www.lscheffer.com/Uncertain.pdf .

[155] L. Scheffer, "An Overview of On-Chip Interconnect Variation", *Proc. International Workshop on System Level Interconnect Prediction*, 2006, pp. 27–28.

[156] F. M. Schellenberg and L. Capodieci, "Impact of RET on Physical Layouts", *Proc. International Symposium on Physical Design*, 2001, pp. 52–55.

[157] J. B. van Schoot, O. Noordman, P. Vanoppen, F. Blok, D. Yim, C.-H. Park, B.-H. Cho, T. Theeuwes and Y.-H. Min, "CD Uniformity Improvement by Active Scanner Corrections", *Proc. SPIE Optical Microlithography XV*, Vol. 4691, 2002, pp. 304–314.

[158] E. Seevinck, F. J. List and J. Lohstroh, "Static-Noise Margin Analysis of MOS SRAM Cells", *IEEE Journal of Solid-State Circuits* SC-22(5) (1987), pp. 748–754.

[159] R. Seltmann, R. Stephan, M. Mazur, C. Spence, B. La Fontaine, D. Stankowski, A. Poock and W. Grundke, "ACLV-Analysis in Production and its Impact on Product Performance", *Proc. SPIE Optical Microlithography XVI*, Vol. 5040, 2003, pp. 530–540.

[160] SEMI P18-92 (Reapproved 1104), "Specification for Overlay Capabilities of Wafer Steppers", http://www.semi.org .

[161] SEMI P19-92 (Reapproved 0707), "Specification for Metrology Pattern Cells for Integrated Circuit Manufacture", http://www.semi.org .

[162] A. Sezginer, B. Yenikaya and W. Staud, "Double Patterning Technology: Process-Window Analysis in a Many-Dimensional Space", *Proc. SPIE Photomask and Next-Generation Lithography Mask Technology XIV*, Vol. 6607, 2007, pp. 66072S-1–66072S-9.

[163] D. Shamsi, P. Boufounos and F. Koushanfar, "Noninvasive Leakage Power Tomography of Integrated Circuits by Compressive Sensing", *Proc. ACM/IEEE International Symposium on Low Power Electronics and Design*, 2008, pp. 341–346.

[164] A. Shibkov and V. Axelrad, "Integrated Simulation Flow for Self-Consistent Manufacturability and Circuit Performance Evaluation", *Proc. International Conference on Simulation of Semiconductor Processes and Devices*, 2005, pp. 127–130.

[165] 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. ACM/EDAC/IEEE Design Automation Conference*, 2007, pp. 823–828.

[166] N. Shigyo, “Variations of Interconnect Capacitance and RC Delay Induced by Process Fluctuations”, *Proc. International Workshop on Statistical Metrology*, 2000, pp. 68–71.

[167] W. Shiu, W. Ma, H. W. Lee, J. S. Wu, Y. M. Tseng, K. Tsai, C. T. Liao, A. Wang, A. Yau, Y. R. Lin, Y. L. Chen, T. Wang, W. B. Wu and C. L. Chiang, “Spacer Double Patterning Technique for Sub-40nm DRAM Manufacturing Process Development”, *Proc. SPIE Lithography Asia*, Vol. 7140, 2008, pp. 71403Y-1–71403Y-8.

[168] M. C. Smayling, C. Bencher, H. D. Chen, H. Dai and M. P. Duane, “APF Pitch Halving for 22nm Logic Cells Using Gridded Design Rules”, *Proc. SPIE Design for Manufacturability through Design-Process Integration II*, Vol. 6925, 2008, pp. 69251E-1–69251E-8.

[169] M. C. Smayling, H. Y. Liu and L. Cai, “Low k_1 Logic Design Using Gridded Design Rules”, *Proc. SPIE Design for Manufacturability through Design-Process Integration II*, Vol. 6925, 2008, pp. 69250B-1–69250B-7.

[170] B. E. Stine, D. S. Boning and J. E. Chung, “Analysis and Decomposition of Spatial Variation in Integrated Circuit Processes and Devices”, *IEEE Transactions on Semiconductor Manufacturing* 10(1) (1997), pp. 24–41.

[171] B. E. Stine, V. Mehrotra, D. S. Boning, J. E. Chung and D. J. Ciplickas, “A Simulation Methodology for Assessing the Impact of Spatial/Pattern Dependent Interconnect Parameter Variation on Circuit Performance” *Proc. IEEE International Electron Devices Meeting*, 1997, pp. 133–136.

[172] J. Stirniman and M. Rieger, “Fast Proximity Correction with Zone Sampling”, *Proc. SPIE Optical/Laser Microlithography VII*, Vol. 2197, 1994, pp. 294–301.

[173] M. Switkes and M. Rothschild, “Immersion Lithography at 157 nm” *Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures* 19(6) (2001), pp. 2353–2356.

[174] D. Sylvester, O. S. Nakagawa and C. Hu, “Modeling the Impact of Back-End Process Variation on Circuit Performance”, *Proc. International Symposium on VLSI Technology, Systems and Applications*, 1999, pp. 58–61.

[175] R. O. Topaloglu, “Interconnect Variability Analysis for Double Patterning Lithography”, *Proc. International VLSI Multilevel Interconnection Conference*, 2008, pp. 267–270.

[176] W. J. Trybula, “A Common Base for Mask Cost of Ownership”, *Proc. SPIE Photomask Technology*, Vol. 5256, 2003, pp. 318–323.

[177] W. J. Trybula and D. L. Dance, “Cost of Mask Fabrication”, *Proc. SPIE Emerging Lithographic Technologies*, Vol. 3048, 1997, pp. 211–215.

[178] D. Tsien, C.K. Wang, Y. Ran, P. Hurat and N. Verghese, “Context-Specific Leakage and Delay Analysis of a 65nm Standard Cell Library for Lithography-Induced Variability”, *Proc. SPIE Design for Manufacturability through Design-Process Integration*, Vol. 6521, 2007, pp. 65210F-1–65210F-10.

[179] T. Tugbawa, T. Park, D. Boning, T. Pan, P. Li, S. Hymes, T. Brown and L. Camilletti, “A Mathematical Model of Pattern Dependence in Cu CMP Process”, *Proc. International Chemical-Mechanical Polishing Symposium*, 1999, pp. 605–615.

[180] A. Vanleenhove and D. Van Steenwinckel, “A Litho-Only Approach to Double Patterning”, *Proc. SPIE Optical Microlithography XX*, Vol. 6250, 2007, pp. 65202F-1–65202F-10.

[181] V. Venkatraman and W. Burleson, “Impact of Process Variations on Multi-Level Signaling for On-Chip Interconnects”, *Proc. International Conference on VLSI Design*, 2005, pp. 362–367.

[182] S. Verhaegen, S. Cosemans, M. Dusa, P. Marchal, A. Nackaerts, G. Vandenberghe and W. Dehaene, “Litho Variations and Their Impact on the Electrical Yield of a 32nm Node 6T SRAM Cell”, *Proc. SPIE Design for Manufacturability through Design-Process Integration II*, Vol. 6925, 2008, pp. 69250R-1–69250R-12.

[183] C. Visweswariah, K. Ravindran, K. Kalafala, S. Walker and S. Narayan, “First-Order Incremental Block-Based Statistical Timing Analysis,” *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2004, pp. 331–336.

[184] J. Wang and A. K. Wong, “Effects of Grid-Placed Contacts on Circuit Performance”, *Proc. SPIE Cost and Performance in Integrated Circuit Creation*, Vol. 5043, 2003, pp. 134–141.

[185] J. Wang, A. K. Wong and E. Y. Lama, “Standard Cell Design with Regularly-Placed Contacts and Gates”, *Proc. SPIE Design and Process Integration for Microelectronic Manufacturing III*, Vol. 5379, 2005, pp. 56–66.

[186] J. Watts, N. Lu, C. Bittner, S. Grundon and J. Oppold, “Modeling FET Variation Within a Chip as a Function of Circuit Design and Layout Choices”, *Nanotech Workshop on Compact Modeling*, 2005, pp. 87–92.

[187] J. Watts, K.-W. Su and M. Basel, “Netlisting and Modeling Well-Proximity Effects”, *IEEE Transactions on Electronic Devices* 53(9) (2006), pp. 2179–2186.

[188] N. H. E. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, Addison Wesley, 2005.

[189] A. K. Wong, R. A. Ferguson and S. Mansfield, "The Mask Error Factor in Optical Lithography", *IEEE Transactions on Semiconductor Manufacturing* 13(2) (2000), pp. 235–242.

[190] J. Xiong, C. Visweswariah and V. Zolotov, "Statistical Ordering of Correlated Timing Quantities and Its Application for Path Ranking", *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2009, pp. 122–125.

[191] Y. Yamamoto, R. Rigby and J. Sweis, "Multi-Layer Reticle (MLR) Strategy Application to Double-Patterning/Double-Exposure for Better Overlay Error Control and Mask Cost Reduction", *Proc. SPIE Photomask Technology*, Vol. 6730, 2007, 67302X-1–67302X-12.

[192] J. Yang, L. Capodieci and D. Sylvester, "Advanced Timing Analysis Based on Post-OPC Extraction of Critical Dimensions", *Proc. ACM/EDAC/IEEE Design Automation Conference*, 2005, pp. 359–364.

[193] J.-S. Yang and D. Z. Pan, "Overlay Aware Interconnect and Timing Variation Modeling for Double Patterning Technology", *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2008, pp. 488–493.

[194] K. Yuan, J.-S. Yang and D. Z. Pan, "Double Patterning Layout Decomposition for Simultaneous Conflict and Stitch Minimization", *Proc. ACM International Symposium on Physical Design*, 2009, pp. 107–114.

[195] Q. Zhang, K. Poolla and C. J. Spanos, "One Step Forward From Run-to-Run Critical Dimension Control: Across-Wafer Level Critical Dimension Control Through Lithography and Etch Process" *Journal of Process Control* 18(10) (2008), pp. 937–945.

[196] G. Zhang, M. Terry, S. O'Brien, R. Soper, M. Mason, W. Kim, C. Wang, S. Hansen, J. Lee and J. Ganeshan, "65nm Node Gate Pattern Using Attenuated Phase Shift Mask with Off-Axis Illumination and Sub-Resolution Assist Features", *Proc. SPIE Optical Microlithography XVIII*, Vol. 5754, 2005, pp. 760–772.