

Design Methodologies and CAD for Emerging Nanotechnologies

THÈSE N° 5797 (2013)

PRÉSENTÉE LE 4 JUILLET 2013

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS

LABORATOIRE DES SYSTÈMES INTÉGRÉS (IC/STI)

PROGRAMME DOCTORAL EN MICROSYSTÈMES ET MICROÉLECTRONIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Shashi Kanth BOBBA

acceptée sur proposition du jury:

Prof. D. Atienza Alonso, président du jury

Prof. G. De Micheli, directeur de thèse

Prof. Y. Leblebici, rapporteur

Prof. I. O'Connor, rapporteur

Dr V. F. Pavlidis, rapporteur



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2013

Abstract

With technology scaling reaching the fundamental limits of Si-CMOS in the near future, the semiconductor industry is in quest for innovation from various disciplines of *integrated circuit* (IC) design. At a fundamental level, technology forms the main driver for innovation, where emerging nanotechnologies based on new transistor material are being investigated. For instance technologies based on nanowires and nanotubes are promising contenders for Si-CMOS due to their high energy efficiency and improved channel properties. The second driver for innovation in IC design is *three dimensional* (3D) integration. 3D technologies are proven to be cost effective and are being adopted by all the leading fabs. One key driver for the future of IC design is the shift in the design paradigm. Computing paradigms based on new capabilities offered by nanodevices open up new venues for innovation in the field of IC design.

This thesis aims at bridging paths between technology and design for exploring new nanotechnologies. This thesis is organized across three different nanotechnologies with an aim to provide novel circuits, architectures and design methodologies in order to leverage the new capabilities offered by these technologies. The considered nanotechnologies are: *3D monolithic integration* (3DMI), *Silicon nanowire FET* (SiNWFET), and *Carbon nanotube FET* (CNFET). The novelty and contributions of this thesis consists of proposing design methodologies and developing *computer aided design* (CAD) tools for these nanotechnologies by taking into account the technology constraints.

This thesis has an interdisciplinary vision involving process, design and CAD for emerging nanotechnologies. In the first part of the thesis, a physical design tool (CELONCEL) is developed for ultra fine-grain 3DMI circuits, whereby the main aim is to evaluate the performance of 3DMI technology for ASIC design. The second part of the thesis deals with layout technique for *double-gate silicon nanowire FET* (DG-SiNWFET) when applied to ambipolar logic circuits. Novel layout synthesis algorithm is proposed for complex Boolean functions with embedded XOR/XNOR functionality. In the final part of the thesis, robust design techniques for CNFET circuits are presented, whereby the goal is to improve the yield while considering CNT imperfections.

Two layout techniques are proposed which take into account mispositioned-immune CNTs and CNT-correlation. For the first time CNFET circuits are benchmarked at a system-level with their respective CMOS counterparts.

Keywords:

nanotechnology, emerging, 3D monolithic integration, silicon nanowires, carbon nanotubes, design, process, layout, ambipolarity, robust design techniques, placement, physical design, synthesis, algorithms.

Résumé

Avec la future atteinte des limites fondamentales de la technologie Si-CMOS, l'industrie des semi-conducteurs est en quête d'innovations dans les différentes disciplines se rattachant à la conception de *circuits intégrés* (CI). De manière générale, le moteur principal et fondamental de l'innovation est la technologie. En particulier, des nanotechnologies émergentes, basées sur de nouvelles structures de transistor, sont étudiées. Par exemple, les technologies faisant appel aux nanofils et nanotubes sont des sérieuses prétendantes au remplacement de la technologie Si-CMOS, du fait de leur haute efficacité énergétique et de leurs propriétés de canal améliorées. Le second moteur de l'innovation dans le domaine de la conception des CIs est l'intégration *tridimensionnelle* (3D). Les technologies 3D sont rentables et sont en passe d'être adoptées par les principaux acteurs industriels. Un des moteurs clés pour la conception des CIs est l'adoption de nouveaux paradigmes de conception. Ces paradigmes, qui sont basés sur les possibilités offertes par ces nanotechnologies, offrent de nouvelles voies dans le domaine de la conception des CI.

Cette thèse vise à explorer les nouvelles nanotechnologies en en faisant le lien avec la conception de circuits. Cette thèse est articulée autour de trois nanotechnologies différentes, dans l'objectif de proposer de nouveaux circuits, de nouvelles architectures et méthodes de conception, tout en s'appuyant sur les possibilités offertes par ces technologies. Les nanotechnologies considérées: l'intégration 3D monolithique (*3D monolithic integration* - 3DMI), les transistors à nanofils silicium (*Silicon nanowire FET* - SiNWFET) et les transistors à nanotubes de carbone (*Carbon nanotube FET* - CNFET). Les nouveautés et contributions de cette thèse se résument à proposer des méthodes de conception et à développer des outils de conception assisté par ordinateur (*computer aided design* - CAD) intégrant les contraintes inhérentes à ces technologies.

Cette thèse a une portée interdisciplinaire, incorporant le processus de fabrication, la conception et les outils de CAD pour ces nouvelles nanotechnologies. En effet, les outils de conceptions qui sont présentés dans cette thèse ont été mis au point en étroite collaboration avec les contraintes

technologiques. Dans la première partie de la thèse, un outil de conception physique bas niveau (CELONCEL) pour des circuits 3DMI est présenté, offrant la possibilité d'évaluer les performances de cette technologie dans le contexte de la conception d'ICs. La seconde partie de la thèse est dédiée aux techniques de layout pour les *SiNWFET à double grille* (DG-SiNWFET) dans le cadre de leur application aux circuits ambipolaires. De nouveaux algorithmes permettant l'automatisation du dessin de fonctions booléennes complexes incluant le support des fonctions XOR/XNOR sont également présentés. Dans la partie finale de la thèse, des techniques robustes de conception pour des circuits à base de CNTFETs sont proposées, avec pour objectif d'améliorer le rendement de fabrication en considérant les imperfections dues aux CNTs. Deux techniques de layout intégrant le mauvais positionnement des CNTs et leur CNT-correlation sont détaillées. Pour la première fois, une évaluation au niveau système entre des circuits CMOS et CNTFET est proposée.

Keywords:

nanotechnologies, 3D monolithique, nanofils silicium, nanotubes de carbone, conception pour des circuits, layout, ambipolaires, techniques robustes de conception, placement, conception physique, synthèse, algorithmes.

Acknowledgments

I am truly grateful to my thesis advisor, Prof. Giovanni De Micheli, for his constant guidance, help, and encouragement over the course of my PhD. Many of my publications would not have been possible without his final ‘push’, and his commitment to constantly improve the quality of my work. I also thank him for giving me ample freedom in choosing my research topics and making me a part of collaborative projects with Stanford University and CEA-LETI.

I thank my thesis committee members: Prof. Ian O’Connor, Prof. Yusuf Leblebici, and Dr. Vasilis F. Pavlidis for reading this dissertation and providing constructive feedback. I would like to particularly thank Prof. Yusuf Leblebici for the technical interaction within the nanosys project, and Dr. Vasilis F. Pavlidis for the fruitful collaboration on 3D integration, and for his detailed comments on my thesis. I would like to thank Prof. David Atienza for serving as president of my jury, and also for being my mentor during the first year of my PhD.

I thank my research collaborators at EPFL - Pierre-Emannuelle, Davide, Michele, Jian, Luca and Vasilis for their contributions to some key publications, which form the core of this thesis. Special thanks to Haykel for sharing his thesis template and also for sharing his wisdom during the early days of my research career. I would like to thank my colleagues at CEA-LETI, Perrine Batude and Olivier Thomas for their fruitful research collaboration on 3D monolithic integration. A special thanks goes to Ashutosh Chakraborty, from UT Austin, for introducing me to physical design and for his help in developing CAD tools. I would like to thank Prof. Subhasish Mitra (firstly, for hosting me at Stanford University) and Jie Zhang for their fruitful collaboration on imperfection-immune CNFET circuits.

A special thanks to all those who are responsible for the underlying academic and administrative work. Firstly, I am very grateful to Christina Govoni for taking care of me right from the first day in the lab, with issues starting from french translation to *oh-you-lost-your-office-key-again!!*. I thank Marie Halm for timely updates on the required administrative work and

providing me with needed information for successfully finishing my PhD. I thank Rodolphe Buret for the technical assistance and Anil Leblebici for organizing seminars.

A very special thanks to all my colleagues and dear friends from Integrated Systems Laboratory: Andrea, Jack, Cristina, Srini, Antonio, Federico, Chip, Pierre-Emmanuel, Jaume, Vasilis, Sandro, Alena, Camilla, Michele, Julien, Irene, Hu, Wenqi, Sara, Somayyeh, Hassan, Gozen, Francesca, Jian, Luca and all the others that I may have missed. Special thanks to Srini for being my *partner-in-fun* at LSI and in organizing many cultural events on campus. I would like to thank Jack, Cri, Jaume, Andrea, Cami, Michele for their constant efforts in stretching the lunch break with their entertaining/adventurous stories. Special thanks to Federico for initiating *dolce-dopo-la-voyage* culture in the cafeteria which led to good stock of sweets during our coffee breaks. I thank Antonio (*hmmmm*) for introducing *lifestyle-without-limits* to the lab. I have to confess, without you guys my life in Switzerland would have been dead boring. I would also like to thank Ashoka, for amazing loads of fun we had together in the last 10 years (travel, cricket, *bheja-fry* ...), but more importantly for inspiring me towards pursuing a PhD.

I am grateful to all my friends for making my life exciting in Lausanne: especially friends from EPFL-Toastmasters club for helping me hone my leadership skills and soft skills and for making my tuesday evenings very entertaining. I thank my friends from TEDxLausanne for bringing the creative spirit to everything we did together, and for rooting in me the power of synergizing technology, entertainment and design. A special thanks to all my dear friends from Yuva (Indian student association), with whom I played, sang (esp. *point-to-point antakshri*), danced, cooked, hiked, organized events ... - *Truly Great Time*.

I would like to thank Ivan and Jack for their legendary entertainment at home and for the positive vibes they bring to Chemin des Cotes (#4). I am truly lucky to share my personal space with these two awesome guys. A special thanks goes to Namrata for all her travel adventures with me and her animated stories which were a great relief during the stressful phase of my PhD.

I owe my deepest gratitude to my parents, for their constant love and uninterrupted support. Especially, I thank my mother for being a driving force in pursuing my dreams. I thank my brother and family-friends for their support. I would like to specially thank my uncle, Dr. Vijay Kumar, for being a huge inspiration to me since my school days.

Contents

Abstract	i
Résumé	iii
Acknowledgments	v
Contents	vii
1 Introduction	1
1.1 Roadmap for Nanotechnology: <i>More-Moore</i> to <i>More-than-Moore</i>	3
1.2 Nanotechnology: Advanced CMOS	4
1.2.1 Advancement in planar transistor	5
1.2.2 Multiple gate transistor structures	7
1.2.3 Gate-All-Around transistor architectures	9
1.3 Nanotechnology: Beyond CMOS	10
1.3.1 Carbon Electronics	10
1.3.2 Single-Electron Transistors	13
1.4 Three Dimensional Integration	14
1.5 EDA for Emerging Nanotechnologies	15
1.6 Research Contributions	17
1.6.1 Contributions to 3D Monolithic Integration	18
1.6.2 Contributions to Silicon Nanowire FETs	18
1.6.3 Contributions to CNFET Circuits	19
1.7 Thesis Organization	20
2 Design Techniques for 3D Monolithic Integration	23
2.1 Technology Background	25
2.1.1 State of the Art	26
2.2 3DMI Technology at CEA-LETI	28
2.2.1 High quality top film	30
2.2.2 High Performance Transistors in Top and Bottom Active Layer	32
2.2.3 3D Contacts	33

2.2.4	Alignment	36
2.3	Standard Cell Transformation Techniques	37
2.3.1	<i>Intra-Cell</i> Stacking Transformation	37
2.3.2	<i>Intra-Cell</i> Folding Transformation	38
2.3.3	<i>Cell-On-Cell</i> Stacking Transformation	40
2.4	Planar-to-3D Library Mapping	41
2.5	Chapter Contribution and Summary	43
3	Physical Synthesis Tool for 3DMI Circuits	45
3.1	State of the Art	45
3.2	Design Flow for Various <i>Cell-Transformation</i> Techniques . . .	47
3.3	Placement Tool: CELONCELPD	47
3.3.1	Initial Transformation: DEFLATE	48
3.3.2	Second Transformation: INFLATE	49
3.3.3	Active Layer Assignment	50
3.3.4	Legalization	53
3.4	Experimental Setup	55
3.5	Results and Discussion	57
3.5.1	Area Comparison	57
3.5.2	Wirelength-driven Placement	58
3.5.3	Timing-driven Placement	58
3.5.4	Timing-driven with <i>in-place</i> Optimization	60
3.5.5	Runtime of CELONCEL Placer	60
3.6	Chapter Contribution and Summary	63
4	3.5D Integration: A Cost Effective Scheme for Future MP-SoCs	65
4.1	3.5D Integration	65
4.2	Multi-Processor System-on-Chip	67
4.3	3.5D Integration for MPSoCs	68
4.4	Cost Analysis	70
4.5	Simulation Framework and Results	71
4.5.1	Performance Improvement of the Core	71
4.5.2	Performance Improvement of the NoC	73
4.6	Chapter Contribution and Summary	75
5	Design Techniques for Nanowire FETs with Controllable Polarity	77
5.1	Transistors with Controllable Polarity: Ambipolar Transistors	79
5.1.1	Double-Gate SiNWFET Technology	79
5.1.2	Device Operation	80
5.1.3	Design Techniques for Ambipolar Circuits	81
5.2	Ambipolar Logic Circuits	81
5.2.1	Terminology	81
5.2.2	Unate, Binate, and Mixed Boolean Functions	81

5.2.3	Ambipolar Logic Gates	82
5.3	Layout Techniques for Ambipolar Logic Gates	85
5.3.1	Dumbbell-Stick Diagrams	85
5.3.2	Layout Techniques for 2-input Unate Functions	86
5.3.3	Layout Techniques for 2-input Binate Functions	87
5.3.4	Layout Techniques for XNUnmixed Functions	87
5.3.5	Procedure for Generating Layout of XNUnmixed Functions	88
5.3.6	Examples	93
5.4	Gate-level Technology Mapping	94
5.4.1	TCAD Model of DG-SiNWFET	95
5.4.2	FO4 delay of Basic Logic Gates	96
5.4.3	Gate-level Mapping of Arithmetic Circuits	98
5.5	Chapter Contribution	101
6	Sea-of-Tiles Fabric for DG-SiNWFET Circuits	103
6.1	Logic Tiles as Building Blocks	105
6.2	Area Optimal Tiles	107
6.3	Case Studies	109
6.3.1	Mapping 3-input Boolean Functions onto SoT of TileG2	109
6.3.2	Mapping Various Blocks onto Sea-of-Tiles of TileG2 and TileG1h2	111
6.4	Sizing the Tiles with Circuit-level Benchmarking	113
6.4.1	Experimental Setup	114
6.4.2	Tile sizing of an Inverter	115
6.4.3	Tile sizing with Circuit-level Benchmarking	116
6.4.4	Synthesis of Data path Circuits	118
6.5	Discussion	120
6.6	Chapter Contribution	124
7	Robust Design Techniques for Carbon Nanotube FET Circuits	125
7.1	Challenges of CNFET Technology	127
7.2	CNT correlation	129
7.3	Yield of CNFET with respect to CNT-Correlations	130
7.3.1	Model for CNT Count Limited Yield	130
7.3.2	Circuit-Level Yield Model	131
7.3.3	CNT Correlation for Enhancing the Yield of CNFET Circuits	134
7.4	Mispositioned-CNT Immune Circuits	135
7.4.1	Layout Technique Based on Euler Paths	136
7.4.2	Mispositioned-CNT Immune Layouts with respect to CNT Correlation	138
7.5	Yield Enhanced CNFET Cell Library	140
7.6	System-Level Benchmarking	142
7.6.1	Experimental Setup	145

7.6.2	Results and Discussion	145
7.7	Chapter Contribution and Summary	151
8	Conclusions and Future Work	153
8.1	Thesis Summary and Contribution	153
8.2	Future Work	156
	Bibliography	159
	List of Figures	177
	List of Acronyms	183
	Curriculum Vitae	185

Introduction

For the last five decades, transistor scaling has been the primary workhorse leading to stellar advancement of modern electronic systems. An empirical observation by Gordon E. Moore on the exponential growth in the number of transistors per die every year [1] has become the mission statement, providing timeline for innovation, to the semiconductor industry. With technology scaling, the performance of the transistor continuously improved, which resulted in reduction of the overall cost. However, ever since the process technology has moved to nanometric regime (i.e. for physical gate length below 100 nm), several challenges have cropped up, such as short channel effects, increase in device parasitics and gate leakage current.

Despite these challenges, transistor scaling continued with timely innovation of the age old Si-transistor. For instance, in order to improve the mobility of the device, strain in the silicon channel was introduced with embedded Silicon Germanium and strained Silicon Nitride for 90nm and 65nm nodes [2, 3]. High-k metal-gate technology is incorporated at 45nm and 32nm node in order to combat the problem of gate leakage current [4, 5, 6, 7], and FinFET technology is introduced at 22nm node to improve the electrostatic control of the channel [8, 9]. State of the art fabrication technology has gate length in the range of few tens of nanometers [10]. Scaling down to such small geometries, we are approaching the fundamental limits of planar Si-CMOS.

In order to continue the trend dictated by Moore's law, while addressing the key challenges posed by planar CMOS, the semiconductor industry is in a quest for innovation from various disciplines of *integrated circuit* (IC) design. Firstly, the focus is on identifying new materials and devices that can potentially replace the traditional Si transistor. This involves emerging

nanotechnologies based on advanced-CMOS approaches, which consists of new channel materials and multigate fully depleted device structures; and beyond-CMOS approaches based on carbon electronics, single electron devices, spintronics, and molecular computing. Secondly, the focus is on a paradigm shift towards heterogeneous integration with *More than Moore* (MtM), where the emphasis is on realizing the complete system (comprising of both the digital and non-digital functionalities like analog, RF, sensors..etc) in a power efficient manner [11]. *Three dimensional* (3D) integration is a key solution offering cost-effective fabrication of MtM products [12]. Thirdly, the focus is on finding new computing paradigms that can leverage on the new capabilities offered by emerging nanotechnologies.

While all the three paths are being explored independently, this thesis aims at bridging paths between technology and design for exploring various emerging nanotechnologies. Abiding to the existing CMOS design flow, we do not reap the full potential of these new technologies which have very attractive properties. A good example of this argument is our affinity towards “Von Neumann” architecture [13]. Despite all its disadvantages, Von Neumann architecture is preferred due to its compatibility with CMOS technology. For instance Von Neumann architecture advocates separation of computation and storage information. CMOS technology inherited this abstraction by optimizing computation (leading to fast, small and expensive) and storage (large and cheap) individually resulting in maximizing the performance.

Many alternatives have been proposed in the last 50 years, none of them succeeded mainly because of the technological CMOS evolution. Emerging nanotechnologies might open up new avenues for adopting different computer architectures for realizing efficient electronic systems. A hint on this could be taken from neuromorphic computing which could lead to a new computing paradigm. Neuromorphic computing offers a very powerful way to process information, similar to natural intelligence, which is much faster than the ordinary CMOS approach based on the “Von Neumann” architecture. Furthermore, emerging nanodevices are prone to uncertainties leading to increase in failure rate, hence new architectures should take into account the possibility of occurring errors. This would lead to more powerful computing, as compared to the deterministic approach of CMOS.

Computer Aided Design (CAD) tools are inherent to today’s complex IC design process involving millions of transistors. The need for a new technology will trigger the development for appropriate new tools and methods. Hence, design methodologies and tools needed to be customized to the specific technology. Therefore, design tools will definitely be the discriminating factor for the success of one specific technology.

1.1. Roadmap for Nanotechnology: More-Moore to More-than-Moore3

This thesis is organized across three different nanotechnologies with an aim to provide novel circuits, architectures and design methodologies in order to leverage the new capabilities offered by these technologies. The considered nanotechnologies are: 3D monolithic integration, Silicon nanowire FET, and Carbon nanotube FET. The novelty and contributions of this thesis consists in proposing design methodologies and developing CAD tools for these nanotechnologies by taking into account the technology constraints. This close interaction with technology is needed in order to fully exploit the tremendous potential of nanodevices.

1.1 Roadmap for Nanotechnology: More-Moore to More-than-Moore

In formulating his famous law, Gordon E. Moore made an empirical observation on the doubling of the number of transistors per CPU every year [14]. Moore's law provided a clear direction and timeline for innovation in the semiconductor industry for the last five decades. Relentless focus on Moore's Law, guided by the scaling rules set by Dennard [15], has provided ever-increasing transistor performance and density (Figure 1.1). According to Dennard scaling, the oxide thickness (T_{ox}), transistor length (L_g) and transistor width (W) are scaled by a constant factor ($1/k$) in order to provide a delay improvement of $1/k$ at constant power density [15].

As transistor scaling entered the nanometric regime (physical gate length below 100 nm), the classical transistor scaling could not meet the scaling

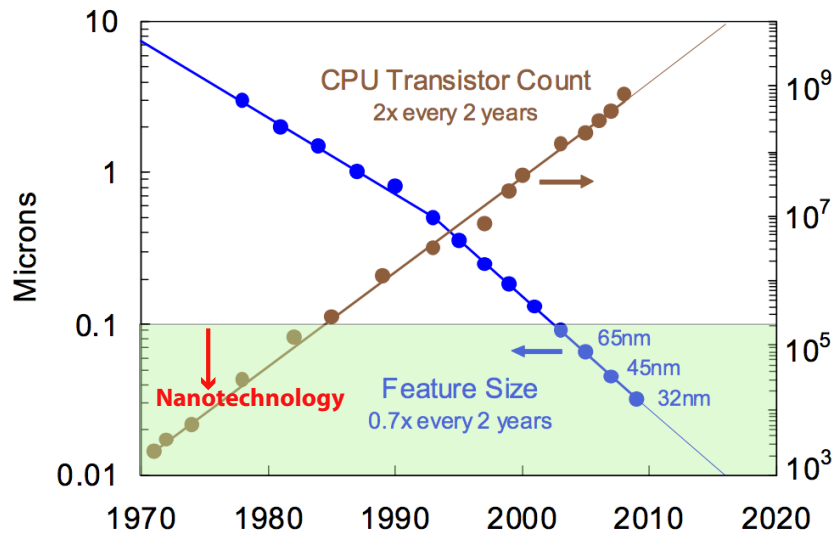


Figure 1.1: Moore's Law: CPU transistor count has increased by 2X and feature size has decreased by 0.7X every two years.

rules set by Dennard. *Short Channel Effects* (SCE) pose a major challenge to the current transistor architecture. For instance, by reducing the length of transistor, the off-state leakage current (I_{off}) is increased due to the *Drain-Induced Barrier Lowering* (DIBL) and degraded *Subthreshold Slope* (SS). Furthermore, scaling of gate oxide thickness of (T_{ox}) is needed to improve the electrostatic control of the channel. A T_{ox} of 1 nm is required for the 10 nm node in order to attain sufficient electrostatic control over the channel. On the other hand, such thin atomic layers of gate oxide comes at a cost of increase in the gate leakage current. Practical considerations on leakage limit the physical gate length to $\sim 15\text{-}20$ nm [2]. Moreover, decreasing the gate pitch decreases the stress enhancement for N- and P- MOSFETs thereby decreasing the mobility of the carriers.

Various advanced CMOS techniques (discussed in Sec. 1.2) are applied to enhance the quality of the transistors in order to achieve the performance targets set by Moore's law. In addition, various new technologies based on entirely new materials (nanowires, nanotubes..etc) are being explored as a replacement for Si-CMOS for sub-5 nm node.

In the recent years, *International Technology Roadmap for Semiconductors* (ITRS) shift the focus from More-Moore (Moore's law scaling) to *More-than-Moore* (MtM). Instead of focusing strictly on the performance of CPU, MtM emphasizes on the overall integration and on the efficient implementation of every component [16, 11]. The concept of MtM evangelizes heterogeneous integration of digital and non-digital (analog, RF, MEMS, sensors...) functionalities into compact systems (system-in-package) that will be the key driver for a wide range of applications, such as communication, automotive, and health-care. The main aim of MtM scaling is to increase system-level power efficiency and capabilities, and to provide a roadmap for future nanotechnologies. Fig. 1.2 illustrates the two roadmaps that will be crucial for deployment of future nanotechnologies. On one hand, transistor scaling is continued as this leads to high performance CPU, memory and logic. On the other hand, MtM aims at providing power efficient systems. One of the key enabler for realizing MtM systems is *Three dimensional* (3D) integration (discussed in Sec. 1.4).

1.2 Nanotechnology: Advanced CMOS

Scaling the transistor in the nanometric regime worsens short channel effects, increases device parasitics and increases gate leakage current. In order to mitigate these drawbacks, various transistor structures are being investigated for advanced technology nodes. These transistor structures can be broadly classified based on the method of electrostatic confinement over the channel [17].

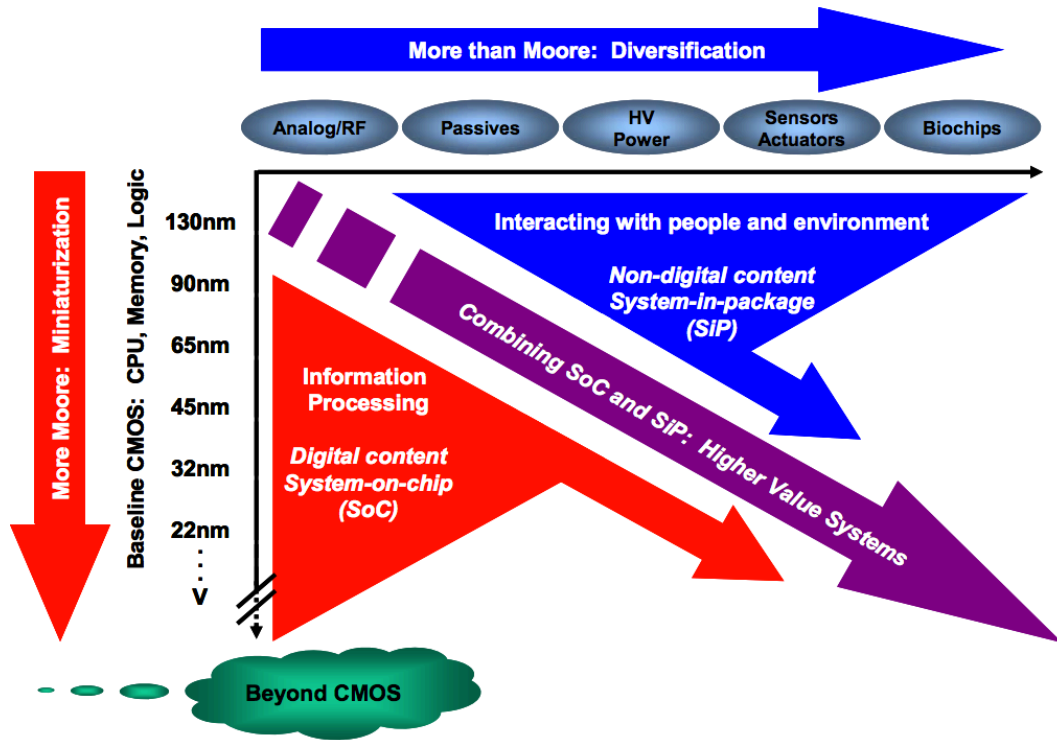


Figure 1.2: More than Moore. “Whereas More Moore may be viewed as the brain of an intelligent compact system, ‘More-than-Moore’ refers to its capabilities to interact with the outside world and the users.” [11]

- Planar transistor structure with enhanced electrostatics (*e.g.* Strained silicon, High-k, ultra-thin body).
- Multiple gate transistor structure (*e.g.* double-gate, tri-gate, FinFET).
- *Gate-All-Around* (GAA) transistor structure (*e.g.* nanowire FET).

1.2.1 Advancement in planar transistor

In this section, process techniques in order to mitigate the transistor scaling issues of a planar Si-CMOS transistor are discussed. Strained silicon is introduced in order to improve the mobility of the device, where as high-k metal-gate technology is employed to reduce the gate leakage current. With the introduction of *fully-depleted silicon on insulator* (FDSOI), parasitics and leakage current in the substrate are reduced.

Strained Silicon

This technique has been widely adopted by the industry as the performance of the transistor is improved without any further shrinking of the transistor

gate length by introducing lattice strain into the Si channel. By inducing mechanical strain in the channel region, the carrier transport properties of the NMOS and PMOS transistors are enhanced. By placing an active silicon region on a substrate layer with a larger lattice constant, strain is induced. Fig.1.3 illustrates a silicon layer on Silicon Germanium lattice. This modifies the band-structure within the active silicon region. This modification leads to a lower scattering probability and thus, to a higher mobility in the channel. By using silicon with 20% Ge portion, the electron mobility can be enhanced by 70% leading to a speed improvement around 30% [18].

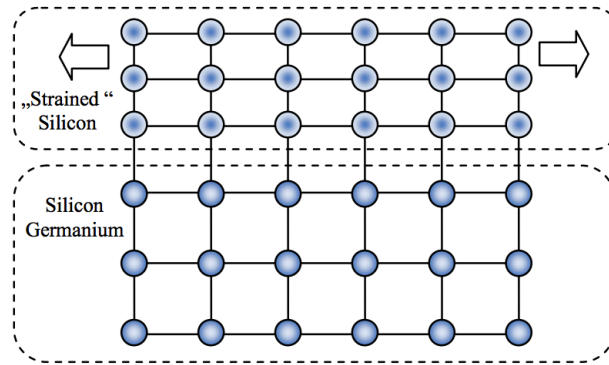


Figure 1.3: Illustration of straining of silicon by means of silicon germanium.

Strained Si is being implemented in nearly all 90 nm, 65 nm, and 45 nm technology nodes [2, 3]. Though the gate length is kept constant with this technique, the transistor density can be increased by scaling down the transistor pitch. However, beyond few technology nodes, this will be limited by parasitic resistance and capacitance between the gate and source/drain contacts.

High-K Metal-Gate

Scaling the thickness of the gate oxide (T_{ox}) improves the electrostatic control of the channel. A T_{ox} of 1 nm is required for the 10 nm node in order to attain sufficient electrostatic control over the channel. On the other hand, such thin atomic layers of T_{ox} comes at a cost of exponential increase in the gate leakage current [9]. Typically, FETs with 20 nm gate length have leakage current densities in the order of $10^{-2} - 10^{-1} A/cm^2$. Gate leakage current is mitigated by incorporating gate oxides with high dielectric constant (called as high-k), thereby allowing to increase the oxide thickness while ensuring good electrostatic control of the channel. The thicker the gate oxide, the lower is the gate tunneling current. Among the various high-k dielectrics being investigated, HfO_2 is widely adopted by the industry for process nodes below 45nm. For instance, the gate leakage current is reduced by a factor of more than 10^4 for a 20-nm device [19].

Ultrathin body MOSFET

This approach replaces the bulk silicon of a normal transistor with a thin layer of silicon built on an insulating layer, creating a device that is often called an *ultrathin body silicon-on-insulator* (UTB SOI), also known as a fully depleted SOI. Fig. 1.4a illustrates an UTB SOI transistor. An SOI isolation layer separates the thin active device layer and the main substrate. Due to the full depletion of the body, there is no room for unwanted current paths to form under the channel like in a traditional bulk device. This results in reduced parasitic and leakage currents in the substrate. The roots of UTB SOI technology for planar electrostatic confinement dates back to 1980s [20]. UTB SOI devices are quite similar to conventional planar CMOS transistors, thereby making them easier to manufacture.

UTB SOI transistors are ideal for low-power applications as there is a possibility for body biasing with the thin *buried oxide* (BOX). By applying a small voltage to the silicon substrate below the BOX, we can alter the channel properties, reducing the electrical barrier that stops current flowing from the source to the drain. As a result, less voltage needs to be applied to the transistor gates to turn the devices on. When the transistors are not needed, the bias voltage can be removed, which restores the electrical barrier thereby reducing the leakage current of the device. The main challenges of UTB SOI include variation in the thickness of the thin silicon film and also the difficulties in inducing strain in the channel.

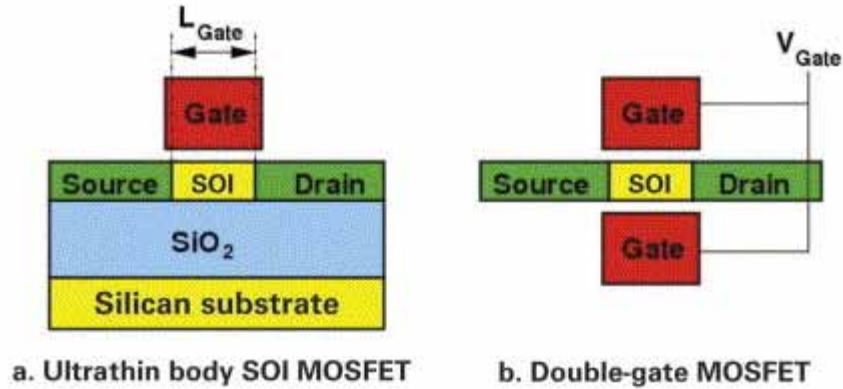


Figure 1.4: (a) Ultrathin body SOI MOSFET (b) Double-gate SOI MOSFET.

1.2.2 Multiple gate transistor structures

In order to minimize the increasing *Short Channel Effects* (SCE) beyond 22nm process, a number of multiple gate FETs have been developed. The trend towards multiple gate transistor started from the idea of a double-gate device based on SOI material [21] (see Fig.1.4b). Fig. 1.5 depicts various

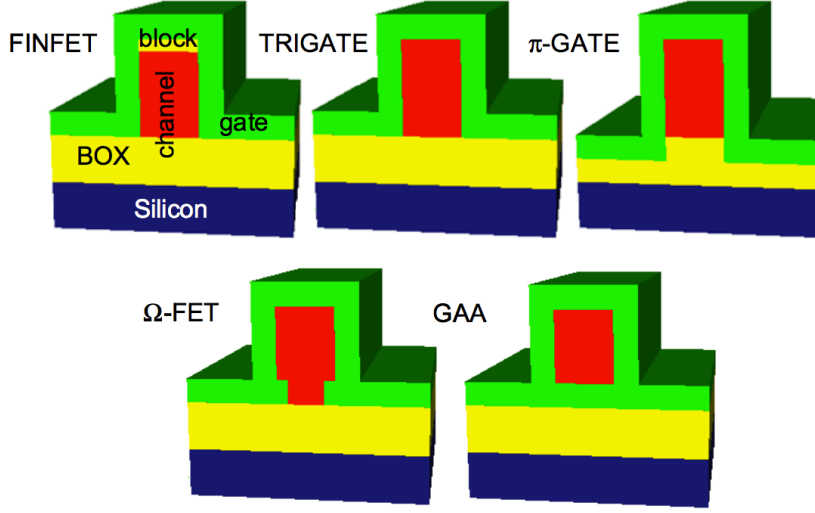


Figure 1.5: Types of multiple gate architectures [17]

multiple gate transistor structures that have been demonstrated over the last two decades.

FinFET : The FinFET device shown in the Fig. 1.5 is a manufacturable and cost-effective version of a double gate device [8] with which we can realize double gate devices on a bulk-Si substrate. Unlike traditional transistor channel, the channel of a FinFET device is vertical to the plane of the substrate.

Trigate : The Trigate device shown in the Fig. 1.5 is an extension of a FinFET device (with a double gate) to a three gate structure. Trigate devices have gates around three sides of the device which improves the electrostatic control over the channel thereby reducing the short channel effects [9]. When compared to FinFETs, there is no gate-blocking layer on the top of the gate.

π -Gate and Ω -Gate : Both π -gate and Ω -gate are extensions of Trigate structure for further improving the electrostatic control of the channel. In the case of π -gate, the gate is extended below the channel region which creates a virtual back gate thereby reducing the leakage current from the drain [22]. In the case of Ω -FET, in addition to the trigate structure, the Ω -gate underlaps also the fourth side of the transistor channel. Similar to π -gate structure, this device has a similar effect in reducing the channel leakage thereby improving the electrostatic control of the channel [23].

Gate-All-Around (GAA) FET : When compared to the rest of the multi-gate devices, *Gate-All-Around* (GAA) devices are comprised of a gate which wraps entirely around the channel, thereby providing full two

dimensional confinement over the channel. A GAA device based on nanowire transistors is presented in section 1.2.3.

One of the most popular multiple-gate FETs is a Trigate transistor. Intel employed this transistor at 22nm process node. When compared to UTB SOI device, a Trigate device provides better electrostatic control over the channel as well as near ideal sub-threshold slope. One of the main challenges in making a Trigate device is manufacturing the fins so that they are uniform.

1.2.3 Gate-All-Around transistor architectures

Nanowire transistors

Gate-All-Around (GAA) devices offer the best potential solution to electrostatic confinement challenges. Nanowires are an extreme case of GAA devices. One possibility is to extrapolate the FinFET concept by using a vertically stacked nanowire device that is completely surrounded by a cylindrical gate. Fig.1.6 illustrates a possible extension of a tri-gate FinFET to *Silicon Nanowire FET* (SiNWFET) device structure with a vertical stack of *Silicon Nanowires* (SiNWs) suspended between the source and drain pillars. The superior performance of these devices comes from a high I_{on}/I_{off} , due to the gate-all-around structure, which improves the electrostatic control of the channel, thereby reducing the leakage current of the device.

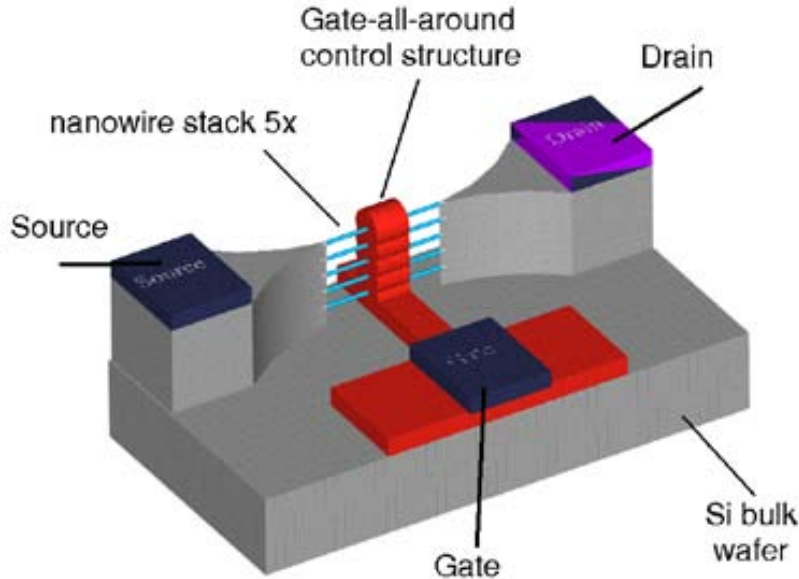


Figure 1.6: Concept drawing of vertically stacked gate-all-around silicon nanowire field effect transistor. [24]

Furthermore, SiNWFETs can exhibit ambipolar conduction (i.e. both electrons and holes have the same contribution to the total drive current), which can be electrically controlled. This feature of controllable polarity of SiNWFET exhibit one very interesting property, where by means of an additional independent gate it is possible to change one hard-wired transistor from n-type to p-type, and vice versa. This improves the computing complexity per transistor, also referred as high expressive power. The in-field polarizability of using ambipolar SiNWFET enables the development of new logic architectures, which are intrinsically not implementable in CMOS in a compact form [25, 26]. Connor *et. al.* have shown a reconfigurable circuit made up of 7 ambipolar transistors which can be configured to any of the 8 logic functions [26]. Design techniques for DG-SiNWFET will be discussed more in detail in Chapter 5 and 6.

1.3 Nanotechnology: Beyond CMOS

While silicon based nanodevices will continue to dominate consumer electronics for this decade, it is well-understood that conventional Moore's Law scaling must come to an end sometime by the next decade, due to a combination of on-chip power dissipation and speed limitations. It is therefore highly probable that new materials and devices will take the place of Si-CMOS and related devices, which have dominated the market for the past 40 years. In this section, few of the most promising nanotechnologies, beyond conventional CMOS, are reviewed.

1.3.1 Carbon Electronics

Two of the most promising contenders for carbon electronics, as a future replacement for Si-CMOS, are Graphene and Carbon nanotube based transistors. Both these technologies are compatible with the current CMOS process flow.

Graphene

Two-dimensional graphene films have generated a huge interest recently as an alternative for channel replacement material in MOSFET structures. Graphene films are well known to behave as high-mobility zero band-gap semiconductors with high carrier mobilities. Researchers have already demonstrated high-speed devices in the range of 300 GHz and are expected to go up to 1 THz. This opens applications in the RF-analog range.

From an integration point of view, graphene devices are planar and compatible with CMOS process. However one of the main drawbacks for VLSI compatibility is the "Zero band gap". Because of the zero bandgap,

devices implemented on large-area graphene channels cannot be switched off and therefore are not suitable for logic applications. Bandgap can be induced in graphene by cutting the material into thin ribbons or applying an electric field to bilayer graphene. When patterned to sufficiently small ribbon widths, the graphene ribbons begin to display a finite band gap resulting from quantum confinement. Opening a band gap requires nanoribbons with sub 5 nm width coupled with very well-defined edges. Variation in edge roughness has a huge impact on the mobility of the device. On the other hand, bilayer graphene requires applied voltages of around 100 V to create a bandgap of about 0.25 eV which is simply not feasible for IC applications. However, Graphene is highly desirable in other venues like Optoelectronics, NEMS, and Spintronics.

One of the close contenders for Graphene is a mono-layered material called Molybdenite (MoS_2). MoS_2 has a direct bandgap (1.8 eV) from the start and does not need to be made into nanoribbons for semiconductor applications [27]. Single-layer molybdenite is a direct-bandgap semiconductor, unlike silicon, which has an indirect gap. It is easier to make devices like LEDs, solar cells and photodetectors and any other photonic devices with direct rather than indirect gap semiconductors.

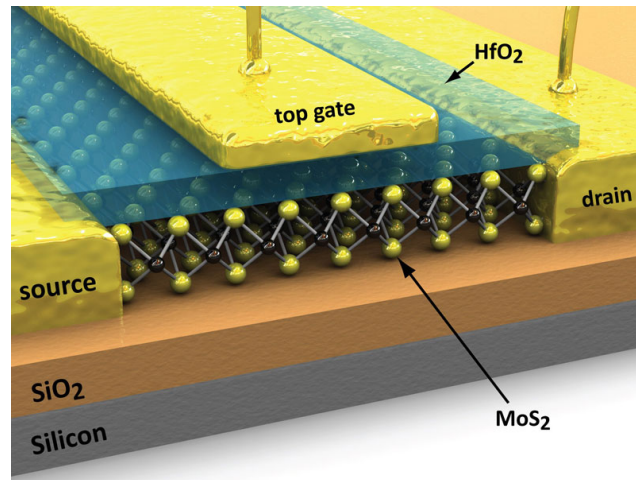


Figure 1.7: Single layer Molybdenite (MoS_2) transistor. [27]

Carbon Nanotubes

Since the discovery of *Carbon Nanotubes* (CNTs) by Iijima in 1991 [28, 29], they have captured the attention of researchers worldwide. CNTs are made from graphene sheet, with one or more than one layers forming *single-walled nanotubes* (SWNTs) or *multi-walled nanotubes* (MWNTs) respectively. Figure 1.8 shows a graphene sheet, when folded into a cylinder forms a SWNT.

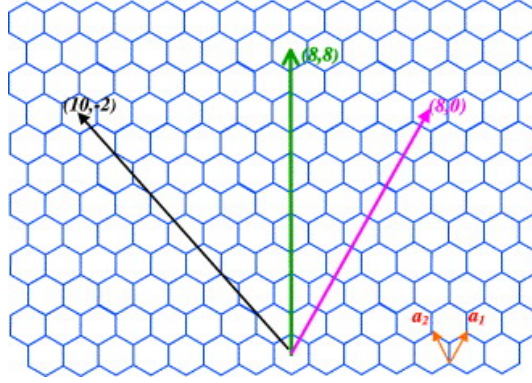


Figure 1.8: Schematic honeycomb structure of a graphene sheet. Carbon atoms are at the vertices. SWNTs can be formed by folding the sheet along lattice vectors. The two basis vectors a_1 and a_2 , and several examples of the lattice vectors are shown.

The electrical properties of the CNT depends on how the cylinder is made from the one-dimensional graphene sheet. An (m,n) nanotube is formed by folding a graphene sheet into a cylinder connecting the ends of a (m,n) lattice vector. The (m,n) indices determine the diameter of the nanotube and the chirality, which determines the electrical characteristics of the CNT. Fig. 1.9 demonstrates various examples of the nature of CNTs based on the chirality of the nanotube.

CNFET devices fabricated with ideal CNT synthesis can potentially provide more than an order of magnitude benefit in energy-delay product over Silicon CMOS at 16 nm technology node [30, 31]. Franklin et al., have demonstrated a sub- 10 nm CNFET, which outperforms its competing Si devices by more than four times in terms of normalized current density at low operating voltages of 0.5 V [32], thereby making them ideal for both high performance and low power applications. However, significant challenges in CNT synthesis prevent CNFETs today from achieving such ideal benefits [33]. CNFET technology is expected to have higher variability, as compared to CMOS, because of the following CNT-specific imperfections related to CNT-synthesis: 1. The presence of metallic CNTs (m-CNTs, versus the useful semiconducting or s-CNTs); 2. CNT diameter variations; 3. Mispositioned-CNTs; and 4. CNT density variations.

Imperfection-immune design techniques for CNFET technology will be discussed in Chapter 7.

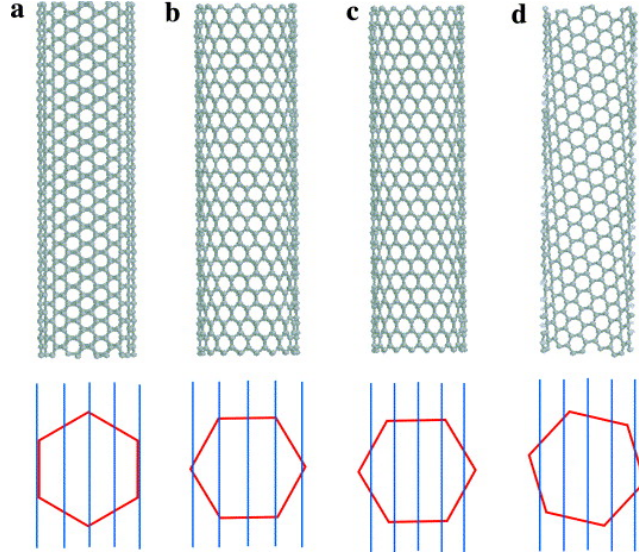


Figure 1.9: (a) Armchair, (b,c) zig-zag and (d) chiral tube; (a) metallic, (b) small gap semiconductor, and (c,d) semiconductor. [34]

1.3.2 Single-Electron Transistors

A *Single-Electron Transistors* (SET) is a three terminal device based on Coulomb blockade. The channel of the transistor is comprised of a single quantum dot, which connects the source and drain of the transistor through tunnel junctions (Fig. 1.10). If $\mu_l > \mu_{N+1} > \mu_r$, then empty states may be populated in the island and single electrons may tunnel through the island. A gate may be used to change the Fermi level of the island and therefore switch the single electron current *on* or *off*. The number of electrons in the quantum dot is controlled by the gate. Depending on the size and material, the quantum dot may have up to thousands of electrons.

Though SETs have high switching speeds, in the order of 0.1 ps, the delay of SET based circuits is limited by the RC time constant (which includes the transistor and interconnect delays). In order to take advantage of SETs, the circuit architecture would have to be local so that the SETs would not have to drive a high capacitance line across the chip. Hence architectures based on *Binary Decision Diagram* (BDD) logic or cellular automata are favorable [35].

On the other hand, due to the high impedance required for Coulomb blockade, SET devices are ideal for memory structures [36]. However, it has to be noted that a SET device driving an external load, such as a word or bit line in a memory cell, will limit the access time to/from the memory cell. Hence, novel architectures are envisaged which bridge the best of SET technology with the standard MOS technology [37].

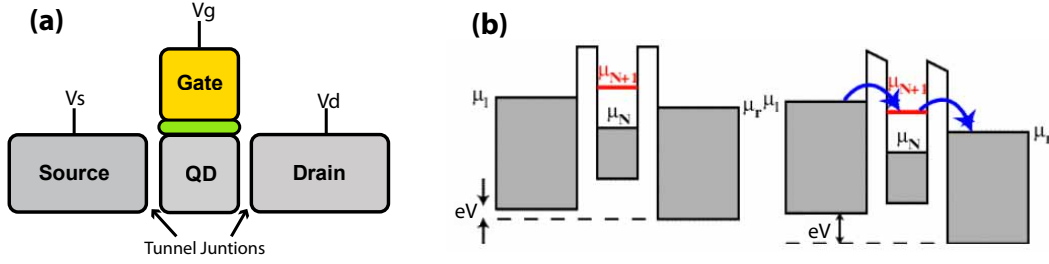


Figure 1.10: (a) Schematic of a SET device (b) Single electron tunneling based on Coulomb blockade.

1.4 Three Dimensional Integration

Three dimensional (3D) integration emerges as a promising solution for realizing future gigascale circuits by integrating multiple layers of active devices vertically [12]. 3D integration increases the density of devices, shortens the interconnect delays, thereby enhancing the performance of circuits. 3D fabrication technologies can be broadly classified into two groups according to the used integration scheme:

- 3D parallel integration with *Through Silicon Vias*, (TSV).
- 3D monolithic integration.

In 3D parallel integration with TSVs, each active layer, along with its respective interconnect metal layers, is fabricated separately and is subsequently stacked via TSVs [38, 39]. Fig. 1.11a illustrates 3D parallel integration with three dies connected with TSVs forming the vertical interconnections. TSVs are built by drilling holes in the silicon die and filling them up with metal. Due to the alignment issues of the stacked dies, the size of the TSVs is kept large (1000nm) in order to ensure electrical connection between the desired points of the dies. Since the size of the TSVs are relatively high when compared to the size of the transistors, they are only feasible for coarse-grain (block-level) integration.

On the other hand *3D monolithic integration* (3DMI) involves processing sequentially thin silicon wafers on top of already processed wafers. Fig. 1.11b shows the cross-section of a wafer manufactured by 3D monolithic process having NMOS devices in the bottom active layer and PMOS devices on the top active layer [40]. The two active layers are connected using a 3D contact which is similar to the conventional inter-layer vias. Since the density of the connection is high, 3DMI technology is suitable from fine-grain (or transistor-level) to coarse-grain (block-level) integration.

Most of the design techniques studied for 3D technology are related to 3D TSV technology with an emphasis on block-level partitioning of circuit in 3D

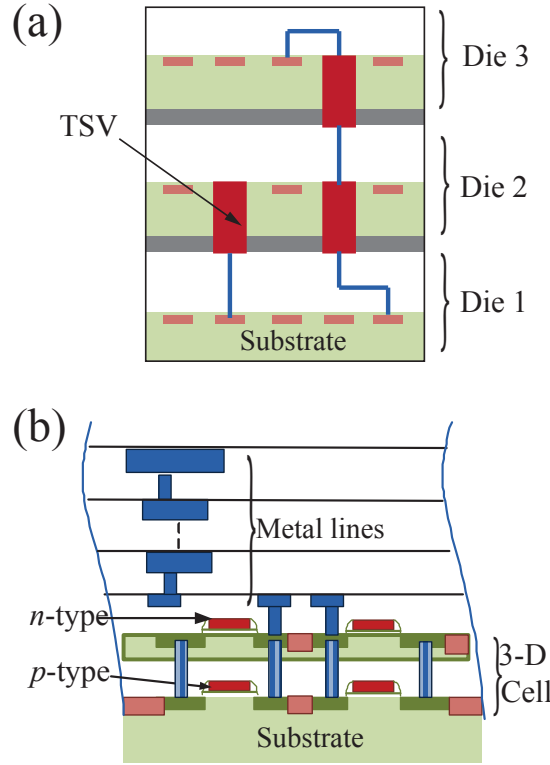


Figure 1.11: (a) 3D TSV integration, (b) 3D Monolithic integration

[41, 42, 43]. 3D monolithic integration has seen substantially less research effort at the CAD level. In order to study the feasibility of 3DMI technology for ASIC design, there is a need for new physical design tool. Existing 3D CAD tools cannot be extended to 3DMI as they do not take into account the technology constraints provided by 3DMI technology. This thesis takes the first step towards providing a complete ASIC design flow for 3DMI technology. Novel cell design techniques (see Chapter 2) along with the placement tool (see Chapter 3) are proposed for evaluating the prospects of 3DMI technology.

1.5 EDA for Emerging Nanotechnologies

It took 5 decades to go from a few transistor to a billion transistor complex ICs. As of 2012, the highest transistor count in a commercially available IC is over 7.1 billion transistors (Nvidia's Kepler-based GK110 GPU). In order to realize such a complex system, designers follow a series of steps starting from computational model at a high level of abstraction, then go through a sequence of synthesis and optimization (technology mapping), followed by physical synthesis flow, and formal verification, before it is finally manufactured via advanced lithography processes. In order to improve the efficiency of the design

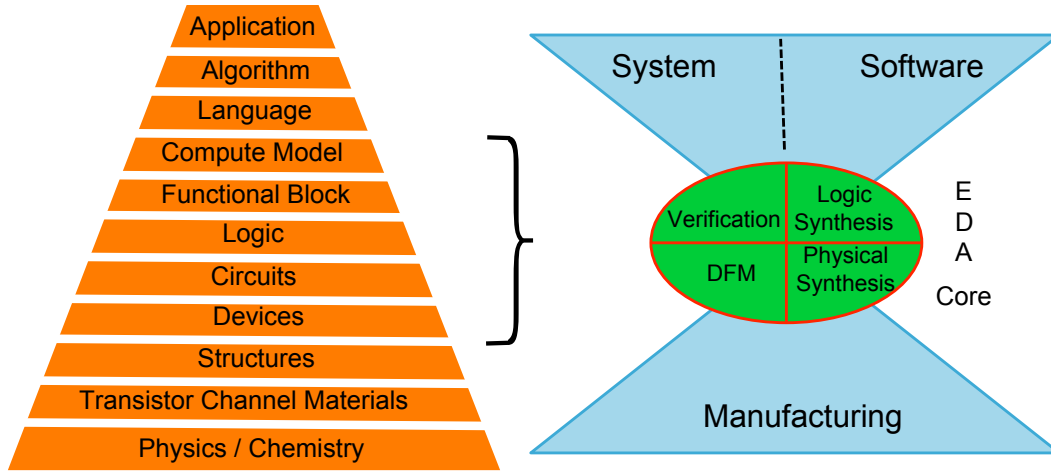


Figure 1.12: Abstraction Levels of the (CMOS) Design Process (left) and the appropriate tools (right). [44]

flow, *Electronic Design Automation* (EDA) has been established. The field of EDA is one of the earliest inter-disciplinary collaborations between computer scientists and electrical engineers. The collaboration yielded a complete design flow from system level specification to silicon implementation. Fig. 1.12 (left) illustrates the various abstraction levels of the current design process [44].

Today's design flow involves hundreds of *Computer Aided Design* (CAD) tools. Fig. 1.12 (right) depicts the number of CAD tools we have at various abstraction levels [44]. Current EDA flow is quite mature when it comes to the design flow starting from application to circuit. However, the main challenge is at the top (between system and application) and at the bottom (between circuit and device).

To find a path between new nanodevices, which show magnificent opportunities, and the possibility to be built into a useful system, an interaction between the communities of design and technology has to be established. While emerging devices have very attractive properties, the design tools need to enable their use in large scale circuits in order to compete with classic circuit design. Novel circuits, architectures and design methodologies are going to be needed for a full exploitation of nanodevices.

To design with emerging nanotechnologies there is a need for:

- Models and abstractions at all levels of the design flow.
- Compatibility to existing industry design standards of high-level behavioral languages.

- New system-level design methodologies at the top of the design flow.
- Robust design techniques at the interface between circuit and device of the design flow.
- Design for manufacturing at the back end in order to obtain high-yield, thereby assuring the feasibility of new nanotechnologies.

Approaching the end of the CMOS roadmap the need for a new technology will trigger the development for appropriate new tools and methods. Hence, a design methodology and tools for a specific technology is becoming a reality. Design tools will definitely be the discriminating factor for the success of one specific technology.

1.6 Research Contributions

This thesis contributes to the design techniques and tools for three emerging nanotechnologies. The design tools presented in this thesis are developed in close collaboration with the technology. The first part of the thesis is on design techniques and CAD tools for 3D monolithic integration. As discussed in Sec. 1.4, the key motivation for 3DMI technology is in enabling fine-grain vertical stacking of transistors, thereby increasing the density of transistors on the chip. The main aim of this work is to evaluate the performance of 3DMI technology for ASIC design. The second part of the thesis deals with design methodologies for ambipolar circuits based on *double-gate silicon nanowire FET* (DG-SiNWFET). Ambipolar circuits are promising due to their higher expressive power [26, 25]. However, the need to independently route the two gates for every transistor adds to the overall routing complexity and this might eventually degrade the benefits of high expressive nature of these transistors. In this part of the thesis, new abstraction at the physical level of the device is proposed, and based on this new layout synthesis algorithm is proposed for complex Boolean functions with embedded XOR/XNOR functionality. The third part of the thesis is on design techniques for CNFET circuits. State-of-the-art CNT synthesis techniques are prone to CNT imperfections and in order to realize functional CNFET circuits, robust design techniques are needed. With the main focus on two important CNT imperfection, mispositioned-immune CNTs and CNT-correlation, layout techniques are proposed in order to improve the yield of CNFET circuits.

The design techniques presented in this thesis focus on a unique aspect that are common to all three nanotechnologies (3DMI, DG-SiNWFET, and CNFET). Hence, some of the techniques presented for each of these technologies can be extended to the other. For example, Wei *et. al.* have experimentally demonstrated 3D monolithic integrated circuit with CNFET technology [31].

Thus for 3D monolithic CNFET technology, we can envisage employing 3D placement tool (presented in the first part of the thesis) for fine-grain partitioning of the circuits across two active layers, while applying the robust layout techniques in order to improve the yield of CNFET circuits (presented in the third part of the thesis). Similarly, the techniques presented in the second and third part of this thesis can be synergized when considering double-gate CNFET technology with controllable polarity [26, 25].

1.6.1 Contributions to 3D Monolithic Integration

The key feature of 3DMI technology is the size of the 3D contacts which is in the order of traditional metal vias in conventional Si-CMOS technology. This opens up venues for fine-grain stacking of transistors vertically and gives a knob to extend Moore's law scaling for over two process nodes. This thesis contributes to design space exploration of this novel technology at various levels (design, CAD and architecture). This work is closely linked to 3DMI technology from CEA-LETI.

- At the design level, this thesis contributes to various standard cell transformation techniques (*intra-cell* stacking, *cell-on-cell* stacking, and *intra-cell* folding). All the three cell transformation techniques are analyzed to study the improvement in performance for each of these techniques. The regularity of the standard cell design flow is ensured for all the transformations, thereby abiding to the conventional ASIC design flow. Though the proposed transformation techniques assume only two active layers, they can be extended for multiple-active layers.
- At the CAD level, this thesis presents a physical design tool (called CELONCEL) for 3DMI technology with fine-grain partitioning. As a matter of fact, at the time at which this thesis is written, CELONCEL is the only physical synthesis flow for realizing 3DMI circuit. CELONCEL places cells in two active layers for improved area, wirelength, and delay. CELONCEL is a pre-/post-processor for existing 2D placement engines which focuses on partitioning across two active layers and the detailed placement for each active layer.
- At the architectural level, this thesis presents the concept of 3.5D integration for future MPSoCs. With 3.5D, i envisage hybridization of fine-grain 3D monolithic integration with the traditional back-end 3D integration (with TSVs).

1.6.2 Contributions to Silicon Nanowire FETs

Double gate transistors with controllable polarity open up new opportunities for efficient implementation of XOR-dominated circuits. On the scientific side, this part of the thesis presents novel layout abstractions for double-gate devices

and layout algorithm for complex functions with embedded XOR gate. On the engineering side, this thesis contributes to design of regular logic tile for DG-SiNWFET technology.

- This thesis addresses gate-level routing issue of DG-SiNWFETs, which is fundamental to all double-gate devices with controllable polarity. In order to facilitate this study, novel symbolic layouts are proposed for ambipolar logic with *Dumbbell-Stick* diagrams. Compact layout techniques are proposed for complex gates with an embedded XOR/XNOR function.
- An efficient regular layout brick (called as tile) is proposed, which forms the basic building block for the *Sea-of-Tiles* (SoT) design methodology. After determining the optimized SoT fabric, physical mapping of various logic functions are studied.
- For the first time a circuit-level benchmarking is carried to study the benefits of DG-SiNWFET when compared to FinFET circuits, with the help of a TCAD simulation of the device at various corners. It has to be noted that, at the time at which this thesis is written, there are no compact models for DG-SiNWFET.

1.6.3 Contributions to CNFET Circuits

Though CNFETs show superior performance when compared to Si-CMOS, they are prone to imperfections coming from the CNT synthesis process. On the scientific side, this part of the thesis addresses layout techniques to enhance the yield of CNFET circuits under the influence of few CNT imperfections. On the engineering side, a system-level benchmarking is carried to study the performance benefits of CNFET versus CMOS. The proposed design techniques are carried in collaboration with Stanford university.

- I propose physical design technique to improve the yield of the CNFET circuit by taking advantage of CNT correlations. With aligned-active layout style, i demonstrate improvement in yield by correlating the critical transistors.
- I present a novel layout technique that is immune to mispositioned CNTs. Various mispositioned-CNT immune layout schemes are studied with respect to CNT correlation and cell routing.
- In order to improve the overall yield of CNFET circuits, I apply robust layout techniques to design the basic building blocks (standard cells) for CNFET circuits. A standard cell library is designed by applying both the active-aligned and mispositioned-CNT immune layout styles.
- By incorporating yield-enhanced standard cell library in the *Integrated Circuits* (IC) design flow, I perform system level benchmarking of CNFET circuits when compared to CMOS circuits. This part of the research

addresses system-level benchmarking, for the first time, comparing CN-FET and CMOS at various technology nodes.

1.7 Thesis Organization

This thesis is divided into three main parts. In the first part of the thesis, I present novel design techniques and a placement tool for realizing high-performance integrated circuits based on 3D monolithic integration. The second part of the thesis deals with the design techniques for DG-SiNWFET technology with the emphasis on realizing ambipolar logic circuits onto regular layout fabrics. In the final part of the thesis, I present physical design techniques for enhancing the yield of CNFET circuits. Description of previous and current work, as well as relation of this dissertation to the work of others is presented in various chapters.

Chapter 2 focuses on design techniques for 3DMI technology. After introducing the state-of-the-art 3DMI technology, a survey of existing design techniques for 3DMI circuits is presented. Then, various standard cell transformation techniques are studied which set the basis for developing a targeted CAD tool as well as for system-level exploration.

Chapter 3 presents the physical synthesis flow for 3DMI circuits. A placement tool, CELONCEL, for *fine-grain* partitioning for 3DMI circuits is developed. The algorithms used by the CELONCEL design tool are scalable to gates with more than million gates. Then, performance gain achieved by the proposed methodology is compared with respect to the existing design methodologies for 3DMI circuits.

Chapter 4 presents a novel 3D integration scheme, called 3.5D integration, which synergizes 3D monolithic integration with the traditional back-end 3D integration (with TSVs). The effectiveness of 3.5D integration is studied by carrying system-level benchmarking of a 288-core MPSoC, based on which hypothesis on the manufacturing and test cost are made.

Chapter 5 presents design techniques for ambipolar logic circuits with a focus on DG-SiNWFET with controllable polarity. After giving a background on DG-SiNWFET technology, novel symbolic-layouts (dumbbell-stick diagrams) for double-gate transistors are proposed. Then, various implementation of ambipolar logic circuits from simple to complex logic functions are studied, followed by a novel layout technique for ambipolar logic gates with embedded XOR/XNOR functionality.

Chapter 6 brings the idea of regular fabrics to DG-SiNWFET technology. In order to improve the yield, a regular layout fabric (called as *tile*) for DG-

SiNWFET technology is proposed. With the help of *Sea-of-Tiles* methodology, various logic functions are mapped onto an array of tiles. With the help of a compact model of the device, a circuit-level benchmarking is carried in order to compare the DG-SiNWFET technology to Si-CMOS at 22nm technology node.

Chapter 7 presents yield-enhancing design techniques for CNFET circuits. After introducing the state-of-the-art CNFET technology, a survey on various CNT-imperfections is studied. In order to improve the yield of CNFET circuits, layout techniques are proposed to mitigate circuit failures caused by mispositioned-CNTs as well as to benefit from CNT-correlation. With the help of yield-enhanced CNFET cell library, a system-level benchmarking of CNFET circuits is carried out.

Chapter 8 concludes the dissertation by highlighting the contribution of this research as well as presents some possible extensions of this work applied to other emerging technologies.

Design Techniques for 3D Monolithic Integration

2

The performance of ICs in advanced technology nodes is dominated by the interconnect delay [45]. Migrating to 3D ICs, we can envisage reduced interconnect delay and chip area which is achieved by placing the logic gates, on the critical path, very close to each other using multiple active layer. Loh et al. have shown the benefits of 3D ICs in terms of wirelength, latency and power depending on the granularity level at which various processing elements are partitioned across multiple active layers [46]. Figure 2.1 illustrates the circuit partitioning of a processor at various granularities. For example, at a coarse-grain level, we can have cache on top of cores, or cores on top of cores, as presented in Figure 2.1a. At a finer level of granularity we can realize functional blocks on top of each other (Figure 2.1b). Going at even finer level, we can perform 3D stacking at the gate and standard cell level, as illustrated in Figure 2.1(c,d). Care should be taken while realizing fine-grain partitioning for routing intense designs, as the routing complexity is further increased.

In this work, I address design techniques for fine-grain partitioning of circuits with 3D integration. *3D monolithic integration* (3DMI) is an ideal choice for ultra-high density 3D circuits, as it provides 3D contacts in the size of few 100nm [47]. In the case of TSV technology, due to low precision of the alignment capability of the equipment and the relatively large size of TSVs, 1000nm [48], circuit integration at the transistor/gate level cannot be done [49].

Figure 2.2 shows the cross-section of a wafer manufactured by 3DMI technology having *N-type* (NMOS) devices in the bottom active layer and *P-type* (PMOS) devices on the top active layer [40]. The two active layers are connected using a 3D contact which is similar to the conventional inter-layer vias. With the latest advancement in 3DMI technology (low temperature top

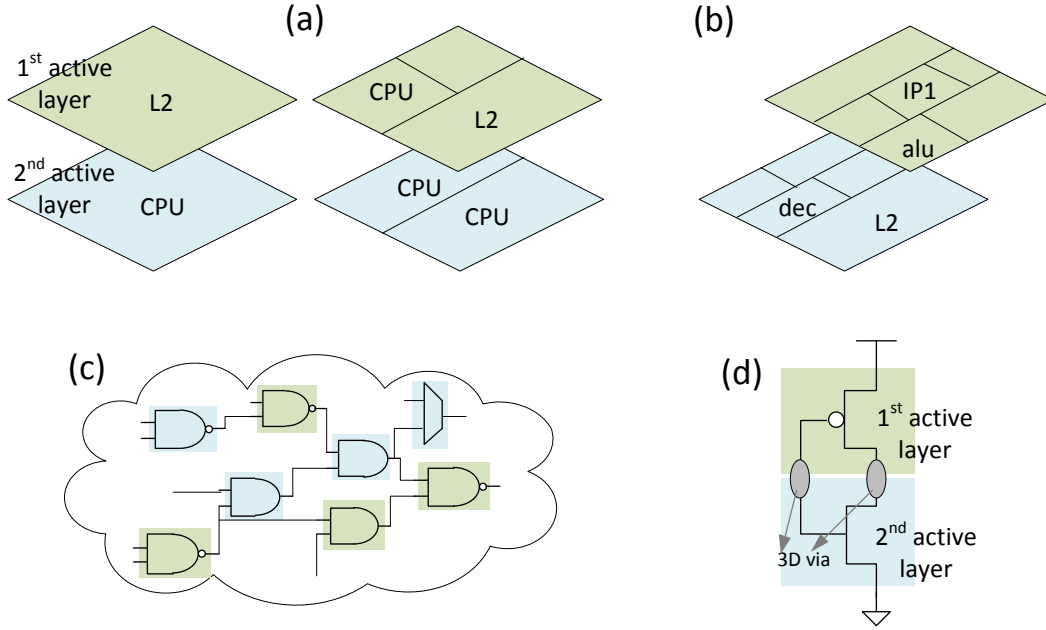


Figure 2.1: Coarse-grain to fine-grain circuit partitioning for 3D circuits [46] (a) Memory/Core on a core, (b) Functional unit blocks on top of each other, (c) Logic gates distributed across different layers, and (d) Transistor scale partitioning.

FETs, intermediate metal layer between the active layers, and high quality bonding) [47], we can build complex 3DMI circuits in the near future.

In this chapter, I provide the overview of the state of the art *3D Monolithic Integration* (3DMI) technology and analyze all the possible standard cell transformation techniques for fine-grain stacking of transistors in 3D. One of the simplest techniques is *intra-cell* transformation, where a standard cell is partitioned across multiple layers [47, 50]. However, with this technique, gates on the critical path cannot be placed close to each other in the third-dimension. In this work I propose a novel *cell-on-cell* transformation technique, where planar (2D) cells are placed on top of each other.

This chapter is organized as follows. The necessary technology background is first introduced with an overview of 3DMI technology from the last two decades. Next, the state of the art 3DMI technology from LETI is presented, which sets the technology assumptions taken in this work. Furthermore, various standard cell design techniques for 3DMI circuits are analyzed followed by planar-to-3D library mapping.

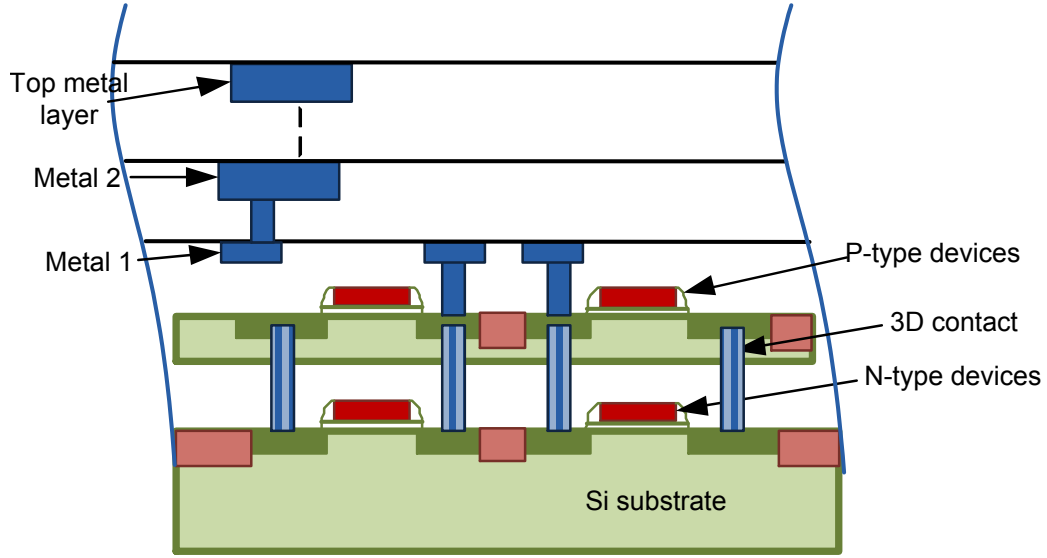


Figure 2.2: Cross-section of a 3D monolithic die with two active layers.

2.1 Technology Background

The seminal work on 3D monolithic integration dates back to early 1980's, when the semiconductor industry for the first time approached the technology scaling problem with the length of the transistor entering the sub-micron dimensions. Foreseeing an end to transistor scaling, which was limited by the basic lithographic resolution, one of the viable solution picked up by the technologists was to stack transistors in 3D. Various articles, from leading industries and research labs, demonstrated stacked transistors between 1980 and mid-1990. The main references are summarized in Fig. 2.3. This fine grain stacking of transistors in 3D is referred as 3D monolithic integration.

The main technological challenge was to fabricate the top semiconductor layer. The first demonstration of 3D monolithic integrations (from Stanford, MIT, TI, Fujitsu..) were based on polycrystalline or amorphous top silicon active area. Polysilicon *Laser Epitaxial Growth* (LEG) and laser recrystallization were the most widely employed techniques by the research groups shown in Fig. 2.3. Some groups also focused on solid phase epitaxy regrowth of amorphous silicon [66]. In any case, the top active area was polycrystalline at the wafer scale and grains had random orientations. From an electrical point of view, this leads to very strong device-to-device variations (threshold voltage, gate oxide thickness...) attributed to grain boundaries and differences in grains orientation.

In order to overcome the grain orientation control challenge, few research groups (IMS) worked on *Epitaxial Lateral Overgrowth* (ELO) where the information of the crystalline structure of the bottom layer was directly

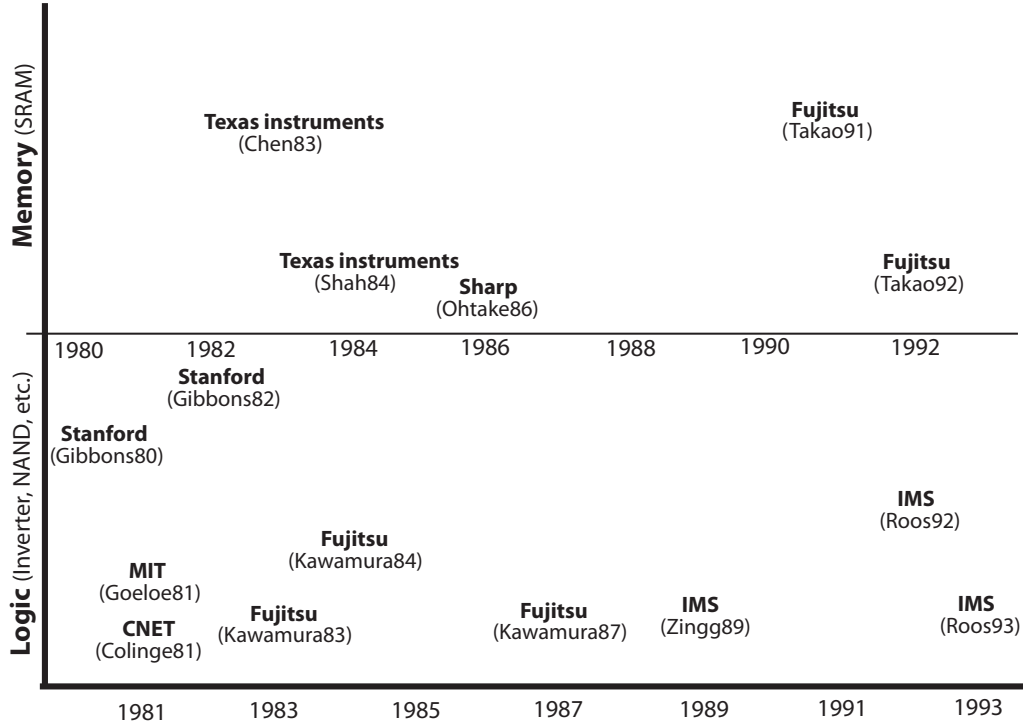


Figure 2.3: Main references related to monolithic 3D integration before 1993. Chen83 [51], Shah84 [52], Gibbons80 [53], Gibbons82 [54], Goeloe81 [55], Colinge81 [56], Kawamura83 [57], Kawamura84 [58], Kawamura87 [59], Ohtake86 [60], Zingg89 [61], Takao91 [62], Takao92 [63], Roos92 [64], Roos93 [65].

transferred to the top layer during the growth process [64], [65]. All these techniques were named “seed window techniques” as they relied on the opening of a window in the bottom layer to transfer the crystalline-structure information. With the advancement in lithography techniques, transistor scaling advanced for the next two decades without the need to adopt 3D monolithic integration.

2.1.1 State of the Art

In nanometer regime, semiconductor industry faces new challenges. In addition to the technological challenges, one has to cope up with short channel effects, variability, interconnect delay..etc. In this context, 3D monolithic integration is attaining the necessary limelight as it is a key enabler in increasing the circuit density and decreasing the interconnect delay. Fig. 2.4 details the monolithic 3D demonstrations showcased since 00’s both for logic and memory applications. In this phase of 3DMI technology, current scaling concerns such as short channel effects control, performance optimization, mobility boost, variability are addressed.

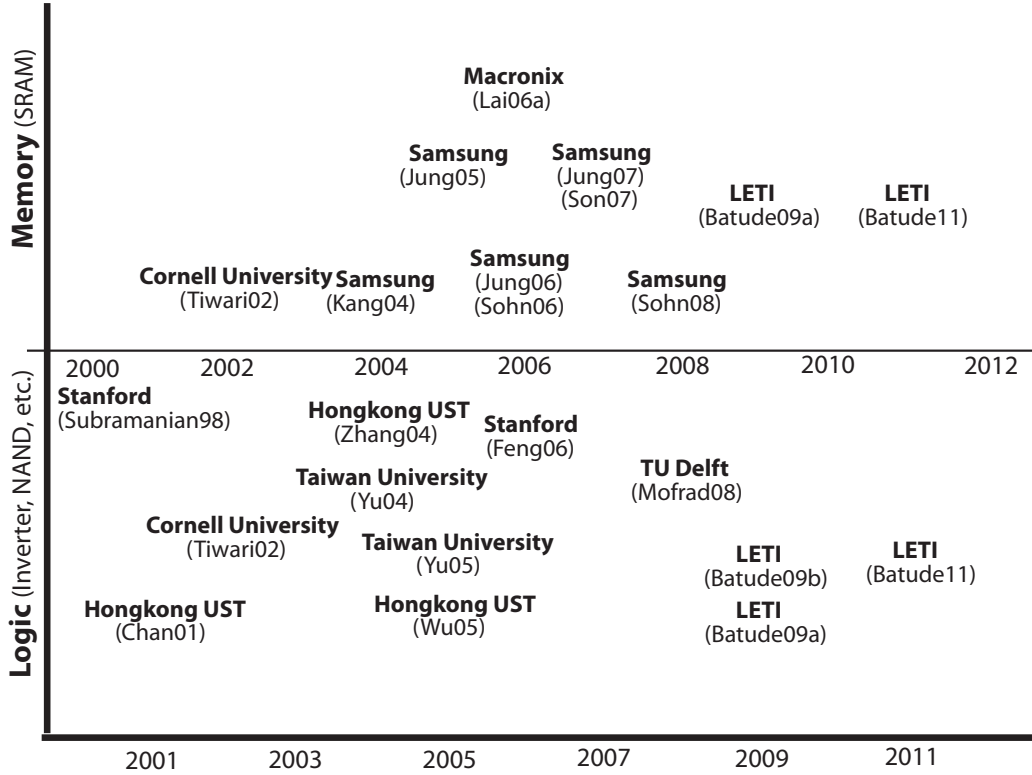


Figure 2.4: Main references of 3D monolithic integration since 2000. Subramanian98 [67], Chan01 [68], Tiwari02 [69], Zhang04 [70], Yu05 [71], Wu05 [72], Feng06 [73], Mofrad08 [74], Batude09b [40], Batude09a [47], Batude11 [75], Kang04 [76], Jung05 [77], Jung06 [78], Sohn06 [79], Jung07 [80], Son07 [81], Sohn08 [82].

3DMI based logic devices

Realizations from single transistors to small circuits have shown the ability of 3DMI technology to design high performance reliable devices. In the early 00's, Honk Kong University demonstrated small circuits based on *Metal Induced Lateral Crystallization* (MILC) [83], [70], [72]. With the introduction of molecular bonding for 3DMI technology, huge advancement have been done to realize a high quality top layer. For the first time, the performance of the top and bottom transistors are matched. Most of the integrated demonstrations however target relaxed dimensions but recently inverters with gate length down to 50nm were fabricated as shown in Figure 2.5 [75].

In the state of the art planar CMOS, the quest for mobility boost resulted in independent optimization of n- and p- FETs with different channel materials or orientations. 3DMI is considered as a choice of privilege to optimize

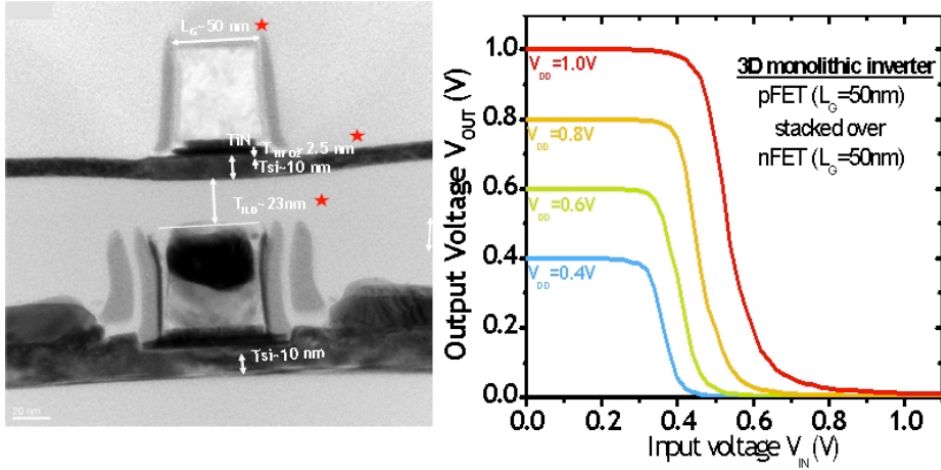


Figure 2.5: SEM cross-section of stacked transistors with $L_G = 50 \text{ nm}$ and ultra thin interlayer dielectric $T_{\text{ILD}} = 23 \text{ nm}$, $T_{\text{Si}} = 10 \text{ nm}$ (morphological structure). Inverter transfer voltage characteristic with pFET stacked over nFET ($L_G, P = L_G, N = 50 \text{ nm}$). [34]

CMOS cells. Many studies have been dedicated to Ge-Si co-integration where the top layer contains only Ge-pMOS FETs and the bottom one contains only Si-nMOS FETs [40]. In addition to independent device type optimization, this integration scheme alleviates the high thermal budget associated to dopant activation in the top layer. Finally for the sake of transport improvement, 3D monolithic integration has also demonstrated its ability to stack different channel orientation [47].

3DMI based Memories

Memories are probably one of the best candidates for 3DMI technology, due to their high density and regularity. Batude et al., have proposed a compact and robust 4T SRAM bit cell which leverages the dynamic coupling offered by 3DMI technology with a thin inter-layer dielectric [84]. Most advance demonstrations were shown by Samsung through the use of their *Single-crystal Si layer Stacking* (S3) technology [85]. From 2004 through to 2007, they kept showing improved performance of scaled SRAM cell with aggressive cell size. They have demonstrated Flash memories and SRAM stacked up to three layers [80].

2.2 3DMI Technology at CEA-LETI

The basic integration principle is described in Fig. 2.6. Firstly devices are fabricated on the bottom layer level where any substrate (bulk or SOI) can

possibly be used (step 1). Then the *Inter Layer Dielectric* (ILD) is deposited and planarized (step 2). In step 3, intermediate metal layer is realized on top of the bottom active layer. Afterwards (step 4) the high quality top active layer is realized either by seed window techniques or by low temperature molecular bonding (discussed in detail in Sec. 2.2.1). In step 5, the top devices are sequentially built on top of the first layer. With an important caveat though, in not to degrade the lower layer properties while fabricating the top device. Finally BEOL is processed to ensure electrical connections between the different layers (step 6). It has to be noted that before BEOL, one can insert more than one layer on top of the bottom layer in order to stack more than two layers. On a similar note, multiple layers of intermediate metal can be realized in between two vertically stacked active layers.

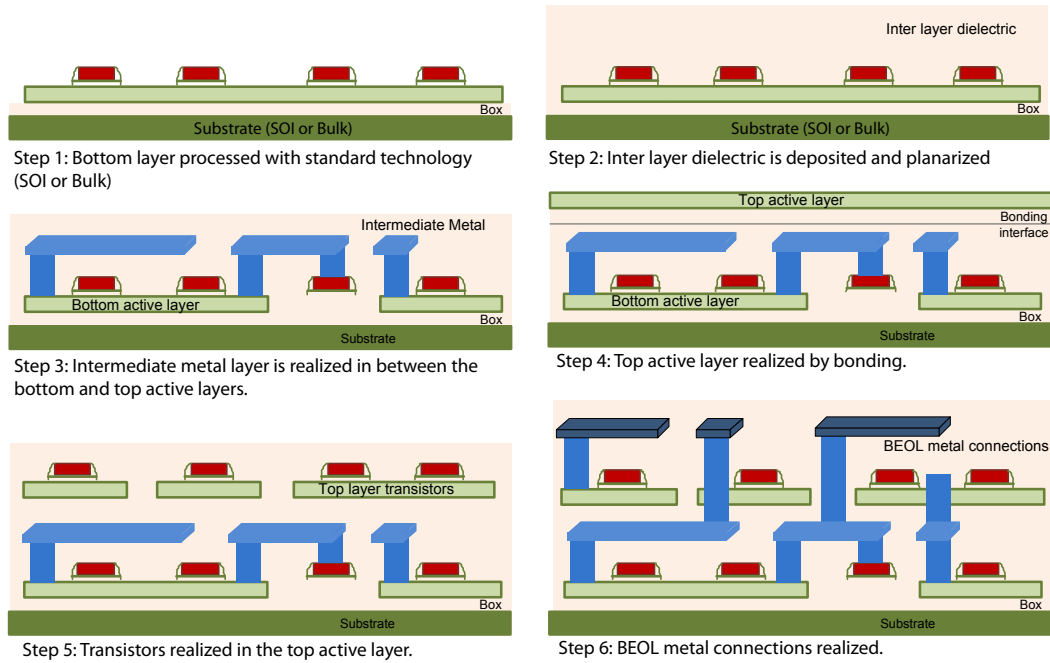


Figure 2.6: Monolithic 3D fabrication.

Within this 3D monolithic integration scheme, for industrial adoption, several technological challenges have to be addressed:

- Fabrication of high quality top film.
- High performance bottom active layer (preservation of bottom FET performance during top FET fabrication).
- High performance top layer taking into account the constraints on thermal budget.
- 3D contacts.

- Alignment between the layers.

2.2.1 High quality top film

To achieve similar performance for the bottom and the top transistor layers, the top semiconductor (active) layer has to be of pristine surface quality (crystalline structure with a very low thickness variation) at the wafer scale. The two main approaches to build the top layers, seed window techniques and molecular bonding, are explained below.

Seed-window technique

The two most widely employed seed window techniques are *Laser Epitaxial Growth* (LEG) and *Epitaxy Lateral Overgrowth* (ELO). Both these techniques are based on recrystallization of top silicon film with the help of crystalline information coming from the seed windows. The seeds are contact-like holes patterned in the *Inter Layer Dielectric* (ILD) and epitaxially filled with single *crystalline-Silicon*.

In the case of LEG, molten amorphous-Silicon is vertically and laterally solidified from the seed to obtain crystalline Silicon. Son et. al., have shown that LEG leads to protrusions in the top films located in the center of neighbor seeds [81]. These protrusions are morphologically defective. Some layout tricks are however expected to avoid them or at least decrease their density. CMP process is eventually required and carried out in order to obtain the flat surface. It was also demonstrated that there is a critical laser energy (located around 800mJ/cm) required to obtain perfect crystalline silicon [81].

On the other hand, ELO brings the crystal-structure information from the bottom layer to the top layer. First, the seed contacts are formed on the bulk Si after ILD planarization using CMP. As the 2nd step, Damascene patterns are formed in the ILD to be filled with Si epitaxial layers grown from the seed contacts. The epitaxial Si layers are grown from the bulk Si in the seed holes and extend laterally when they arrived at the top end of the seed contact. After the growth, the Si layers have hills and valleys due to the difference of growth rates between the growing directions. Such morphology has to be further flatten with CMP process. The Si CMP is expected to stop when it touches the oxide layers, thereby allowing a relative control in the thickness of the Si layer.

With the scaling of lithography pitch, seed window techniques face major challenges to address sub-45 nm node:

- The seed windows themselves are detrimental for density optimization.

- The silicon channel thickness in the recrystallization techniques is defined by CMP which is not yet matured to provide a ± 1 nm control at the wafer scale.

Molecular Wafer Bonding

The use of Molecular Wafer Bonding provides crucial advantages compared to previously employed techniques for top active layer realization:

- Top substrate presents a perfect quality (crystalline quality and thickness control) at the wafer scale (supplier dependent).
- No need for seed-windows which limits density.
- Independent optimization of strain, channel & surface orientations and channel material independently for each layer.

A handle wafer is used to transfer the top active layer onto the bottom active layer. Low temperature bonding of SOI substrate followed by etch back of handle wafer is used to transfer the stacked thin film. A low temperature annealing (200°C) is performed to strengthen the bonding interface between the ILD and the top active layer. The ILD is usually around 100nm between the top of the bottom gate and the bottom of the top silicon channel. The bonding interface is located roughly in the middle of the ILD. Its thickness can be further decreased by using a thin thermal oxide on the handle wafer instead of a deposited one, to eventually be as thin as 25nm [75]. However to achieve high quality molecular bonding, several challenges are raised. Firstly, the thickness of the bottom wafer hosting the bottom transistor layer must be suppressed to enable full film transfer. Secondly, the ILD thickness has to be minimized in order to allow dense 3D contacts.

When compared to the recrystallization technique, molecular wafer bonding fairs well in multiple disciplines (see Table 2.1).

Features	Recrystallization techniques	Molecular bonding
<i>Crystalline orientation (Top vs. Bottom)</i>	Same	Independent
<i>Thermal budget</i>	High	Low
<i>Defect density</i>	High	Supplier dependent
<i>Channel thickness control</i>	CMP	Supplier dependent
<i>Channel material</i>	Same	Independent
<i>Density</i>	Losses due to seed windows	High

Table 2.1: Comparison between seed window techniques and molecular bonding for 3D monolithic integration.

2.2.2 High Performance Transistors in Top and Bottom Active Layer

One of the key challenges to mainstream 3DMI technology for sub-22nm node, is to obtain high performance transistors both on the bottom and top active layers. In order to preserve the bottom layer transistors, dopant diffusion has to be avoided and the access resistance should not be degraded when fabricating the top layer. These two conditions can be met by minimizing the top layer thermal budget and by designing a thermally resistant bottom layer. The challenges associated to each layer are detailed in the following paragraphs.

Low temperature process for top active layer

Low temperature processing of top active layer is crucial in order to retain the performance of bottom layer transistor. Neither additional dopant diffusion, nor salicide degradation can be tolerated. The most expensive step in terms of thermal budget is dopant activation. For Silicon integration a promising way to achieve excellent activation levels is the *Solid Phase Epitaxy* (SPE) technique. SPE is based on the low temperature recrystallization of amorphous Si that results in an above equilibrium activation. This technique can be used at low temperatures (typically below 600°C) which suppress dopant diffusion and facilitate ultra-shallow junction formation. Additionally to dopant activation, other steps such as gate dielectric, spacers, and passivation layers realization have to keep these thermal budgets as low as possible. For gate dielectric realization, thermally grown SiO₂ (at 1000°C) is prohibited and logically replaced by high K dielectric with HfO₂ deposited at 350°C. This allows designing a whole transistor fabrication process in line with the target thermal budget of 600°C.

Thermally robust bottom layer with optimized silicides

Silicides are greatly sensitive to thermal budget. For 600°C thermal budget, the classical NiSi agglomerates, leading to strong increase in sheet resistance. To stabilize NiSi, an original treatment based on platinum associated with fluorine and tungsten implantation has been proposed [75]. Figure 2.7 shows the benefits of this NiSi treatment which ensures its stability up to (650°C, 40 min) whereas NiSi is agglomerated in less than 1 min at this temperature. The electrical results were confirmed by *Scanning Electron Microscopy* (SEM) observations of the silicide layer.

Figure 2.8 and Figure ??, shows the C(V) and Id-Vg characteristics of the top and bottom layer transistors with 3DMI technology. Red curves corre-

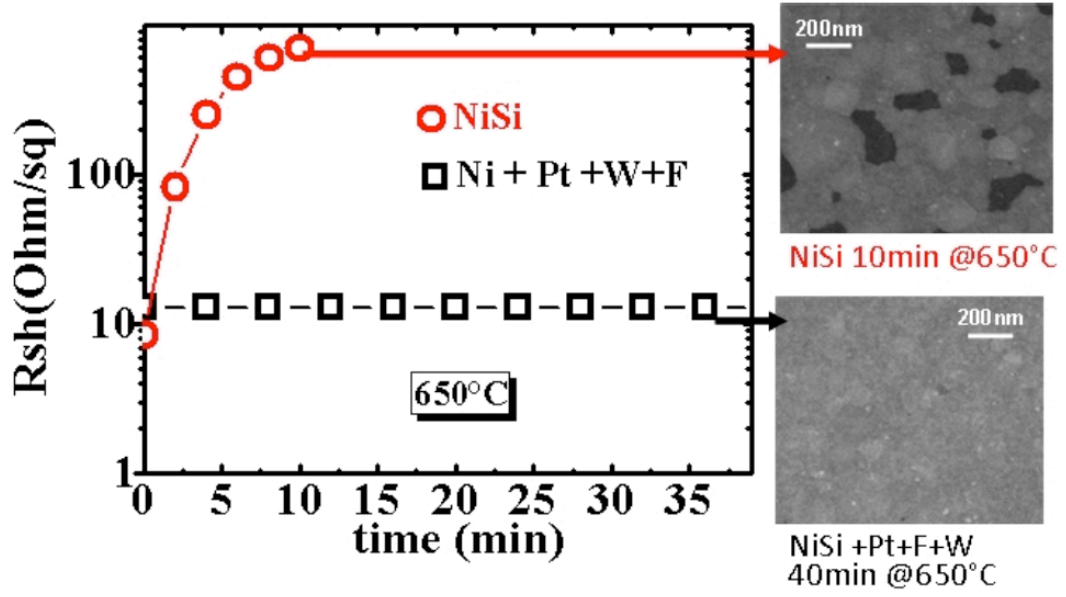


Figure 2.7: Sheet resistance of NiSi and NiSi + Pt + F + W as a function of the annealing time at 650°C. The sheet resistance of NiSi alone exhibits a dramatic increase as soon as 1 minute annealing is performed. Stabilized silicide does not show any change neither in morphology nor in electrical properties for annealing for as long as 40min. [86]

spond to the bottom transistors processed at regular high temperature. Blue curves correspond to the top FETs processed in a cold process [47].

In a summary, for bottom MOSFET, the access silicidation is mandatory for reaching ITRS values in terms of series resistance. The development of a stabilized silicide up to 650°C is an essential breakthrough. For top MOSFET, a 650°C thermal budget allows epitaxy for raised source-drain which is compulsory for advanced nodes with fully depleted SOI structures. This stabilized silicide together with the low temperature process enables to obtain high performance top and bottom MOSFETs for the sub-22nm nodes.

2.2.3 3D Contacts

Contacting top and bottom layer is a problem specific to 3D monolithic integration scheme. Firstly, contacts have to land on two layers at different height (attention has to be paid to the contact aspect ratio) and secondly, contacts have to connect the different layers, see Figure 2.10. Two options are possible to design these 3D contacts:

- Two different lithography steps can be used to design monolayer contact and “thru layers contacts”. In that case, the multilayer contact connecting the top layers will be a lateral contact [80].

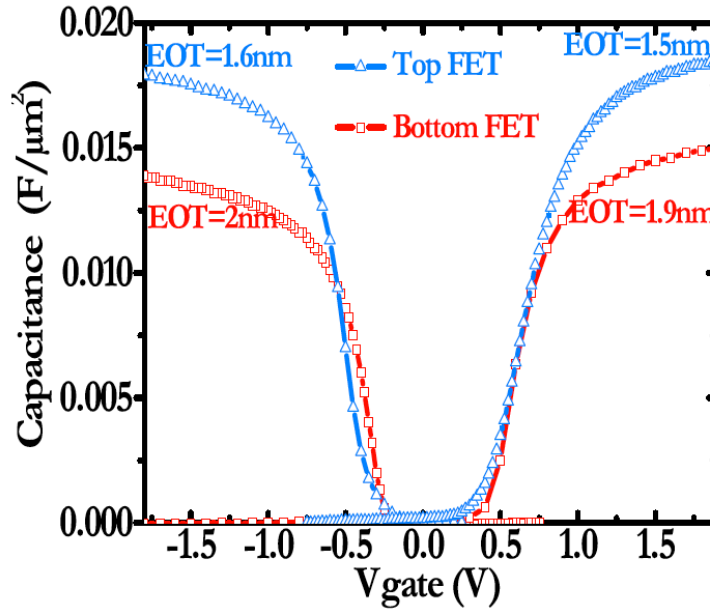


Figure 2.8: $C(V)$ characteristics of bottom and top FETs having the same HfO_2/TiN gate stack but respectively processed at 1050°C and 600°C . Red curves correspond to the bottom transistors processed at regular high temperature. Blue curves correspond to the top FETs processed in a cold process. [47]

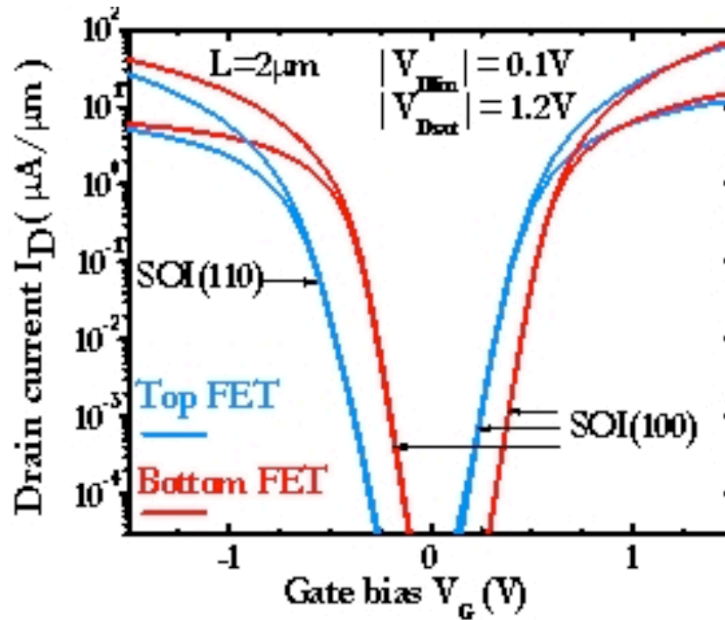


Figure 2.9: Drain current as a function of gate voltage for transistors stacked on the same wafers. Red curves correspond to the bottom transistors processed at regular high temperature. Blue curves correspond to the top FETs processed in a cold process. [47]

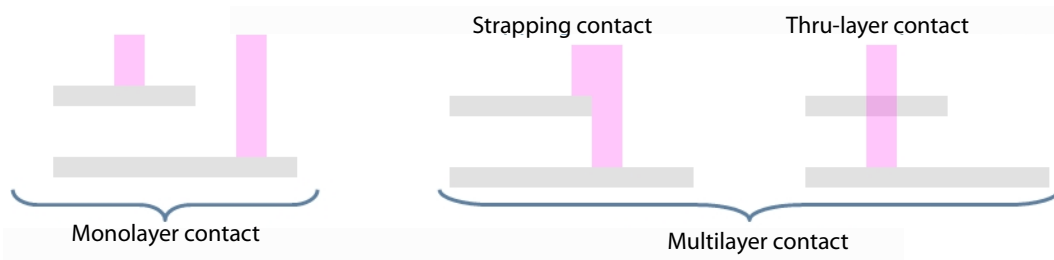


Figure 2.10: 3D contacts for 3DMI technology. Monolayers contacts land either on the top or on the bottom layer whereas multilayers contacts land on several layers. In the case of “thru-layer contact” the contact drills the top layer and further digs into the ILD until reaching the bottom layer whereas in the case of the “strapping contact” a highly selective etching allows lying on both layers at the same time.

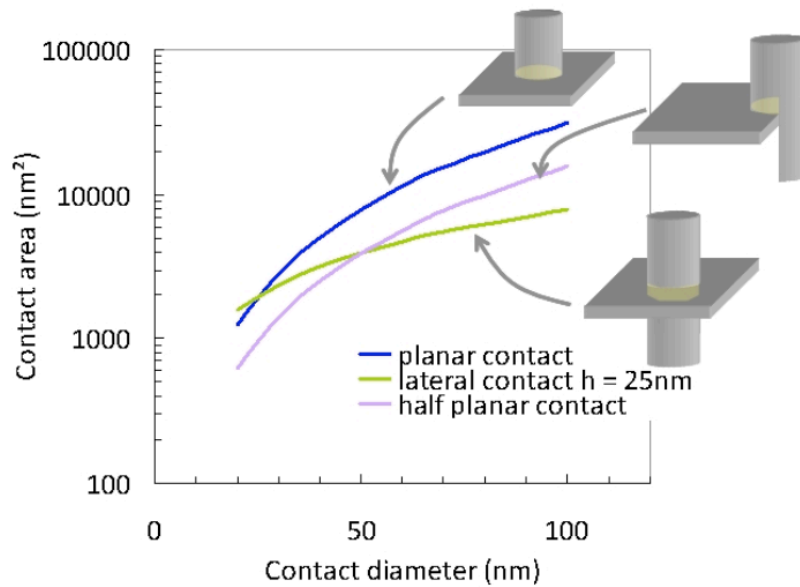


Figure 2.11: Contact area as a function of contact diameter for planar contact (corresponding to monolayer contact), half planar contact (corresponding to “strapping contact”) or lateral contact (corresponding to “thru contact”).

- To save one lithography step, a single lithography can be used for both type of contacts. In that configuration, a highly selective etch is needed to open contact stretching down to bottom layer without passing trough the upper layer. It was demonstrated that the bottom layer can be reached without passing trough the silicide of the top active layer, thereby giving a low contact resistivity [86].

The choice between the different options does not only depend on number of extra lithography steps needed, but also on the physical and electrical attributes (i.e. contact resistance as well as density). Figure 2.11 shows the

area of the various 3D contacts. As the contact diameter is decreased, the lateral contact becomes more and more interesting as it allows larger contact area which is fundamental to decrease the contact resistance (tends to be limiting in device performance). On the contrary, the strapping contact, although interesting from a lithography point of view, tends to have an area that becomes prohibitive for advanced nodes. An alternative option to design the strapping contact is to increase the contact area and design it as a rectangle whose dimensions will be twice the one of a single contact in addition to the possible misalignment between the two connected layers. In that case the area is kept constant on each layer but at the expense of a loss of density.

2.2.4 Alignment

Density of 3D structure is mainly linked to alignment performance. The huge advantage of monolithic integration compared to other 3D integration schemes mainly relies on the fact that the alignment performance between layers only depends on lithographic alignment capability. The overlays of bottom active level with bottom gate level and top active layer are displayed in Table 2.2. We clearly observe that the alignment performance is not degraded for the upper level location. This enables connections at the transistor scale.

	σ bottom gate / bottom active	σ top active / bottom active
X	7 nm	7 nm
Y	10 nm	7 nm

Table 2.2: Alignment performance with a 248 nm stepper for bottom gate on bottom active and top active on bottom active. σ is the standard deviation of the overlay measurement. [86]

2.3 Standard Cell Transformation Techniques

In this section, I discuss three methods of modifying a traditional planar (2D) standard cell to a 3D standard cell. Using the first method (*intra-cell* stacking transformation), a planar cell is mapped into a 3D cell by realizing the pull-up network (all p-type devices) of the cell on the top active layer and pull-down network (all n-type devices) on the bottom active layer. In the second method (*intra-cell* folding transformation), the planar cell is folded across different layers by realizing parts of the cell in different layers while keeping the cell height similar to the planar case. The third method (*cell-on-cell* stacking transformation) modifies the layout of the standard cells in a way that they can be stacked on top of each layer. For all the transformations discussed in this section, I define the height of the cell (standard cells) as the vertical distance between the power and ground rails (see Fig. 2.12), and the width of the cell as the space occupied in the horizontal direction when considering the footprint of the standard cell.

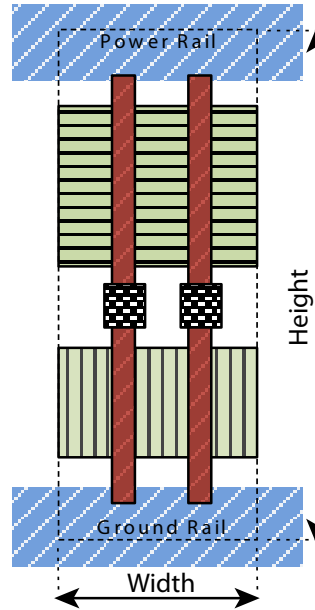


Figure 2.12: Example of a standard cell illustrating the height and width of the cell.

2.3.1 Intra-Cell Stacking Transformation

Standard cells implement pre-defined logic functions (for example, NAND gates, NOR gates, and flip-flops) and have fixed height but varying widths. The structure of a typical 2D standard cell layout is shown in Figure 2.13a. The power and ground rails are located at the top and bottom end of the cell.

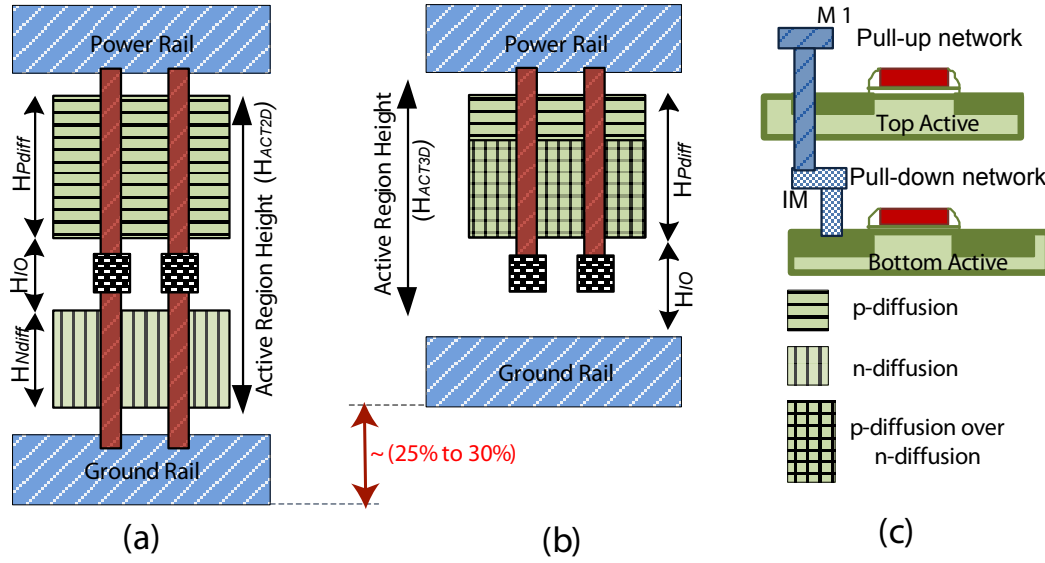


Figure 2.13: (a) Typical cell in 2D configuration (b) *intra-cell* transformation, in two active layers, by realizing pull-up network on the top layer and pull-down network at the bottom layer (c) Cross-sectional view of the two active layers with the metals (IM and M1) for realizing PUN and PDN of the cell.

Active region height (H_{ACT}) of the cell is where the transistors are fabricated. The distance between two diffusion regions is called diffusion gap region, where the input pins are placed. Since 3DMI technology offers multiple active layers adjacent to each other, the layout of the standard cell can be folded in multiple layers [87], [40]. For instance, as illustrated in Figure 2.13b, *p-type* devices are realized on the top active layer and *n-type* devices on the bottom active layer. Since the *p-diffusion* is typically wider than the *n-diffusion*, the active region height for a 3D cell (H_{ACT3D}) is limited by the height of the *p-diffusion* (H_{Pdiff}).

In the above transformation, the reduction in height of a 3D cell is due to the *n-diffusion* region. Moreover, there can be a slight increase in the space needed for *input-output* (IO) pins in the 3D layout, as the design rules should be followed, considering the close proximity of wide power rails. The active region (in green) with horizontal stripes represents a *p-active* region, whereas the green vertical stripes represent the *n-active* region. The overlap between these two active regions, realized in two different layers, has a gridded pattern.

2.3.2 Intra-Cell Folding Transformation

In the previous transformation, we have seen that the height of the standard cell was altered in both the cases. However, we can also envisage a 3D cell, built across multiple active layers, by folding the width of the standard cell. Unlike realizing the *p-type* and *n-type* devices in two different layers, in this

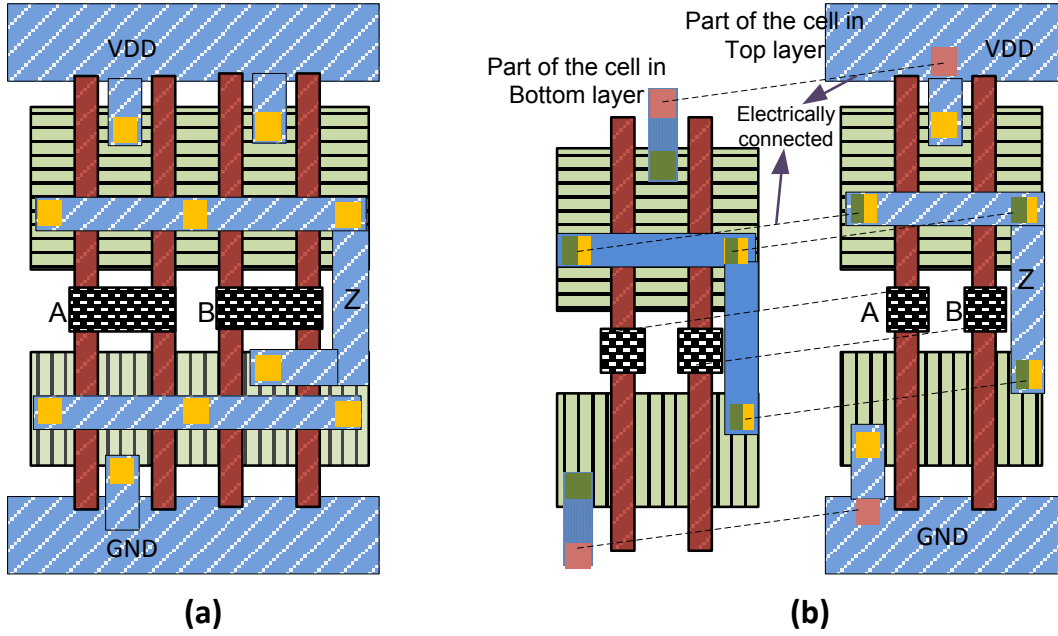


Figure 2.14: (a) Two input NAND cell with a high drive strength having finger transistors (b) corresponding cell built in 3D with *intra-cell* folding transformation where the fingers are realized in the bottom/top active layer.

transformation I fold the gates and fingers of the cells across two different layers. For example, consider a two input (inputs A , B) NAND gate. The width of the cell can be folded by realizing the gate A in the bottom layer and gate B in the top layer. The benefits of this transformation can be maximized when applied to cells with high driving strength, where the large transistors are implemented by multiple fingers.

Figure 2.14a shows a 2-input NAND with high driving strength. With *intra-cell* folding transformation, I realize the fingers in the top active layer thereby resulting in a compact cell as shown in Fig. 2.14b. The layout to the left of Fig. 2.14b is a part of the NAND gate placed at the bottom layer. The dotted line represents the electrical connections between the top cell part and the bottom cell part. Both of the parts are connected to form a 3D cell. Unlike the *intra-cell* stacking transformation, where the gain in height is constant through out the library, the gain in width with *intra-cell* folding transformation varies depending on the type of the cell, number of inputs (*fan-in*) to the cell and the driving strength (*fan-out*) of the cell.

2.3.3 Cell-On-Cell Stacking Transformation

To achieve truly stacked cells, I propose the method of *cell-on-cell* stacking. In *cell-on-cell* stacking, instead of distributing the diffusion regions of the cell in two active layers, the cells are realized with one active layer and one metal layer, but such cells can be placed on top of each other. One of the main challenges for this approach is to access the IO pins of the bottom cell from the top metal layers (for instance metal 2) without short-circuiting the IO pins of the cell placed on the top active layer. Figure 2.15 illustrates an example of *cell-on-cell* stacking of two cells on top of each other. Figure 2.15(a,b) show a 2-input NAND gate, realized in the top active layer and the bottom active layer such that pin access can be maintained. The *intra-cell routing* (ICR) of the bottom cell is realized with the intermediate metal layer in between the active layers. Tungsten is used for ICR of the bottom cell, whereas copper is used for the top cell. I did not observe any considerable delay degradation of the bottom cell when compared to the similar cell realized in the top layer. This attributes to the fact that transistor delay plays a leading role, when compared to the local interconnect delay coming from intra-cell routing, in determining the overall delay of the cell.

In order to access the IO pins of the bottom cell to the top metal layers, extra space is allocated in the top active layer. For instance, the IO pins

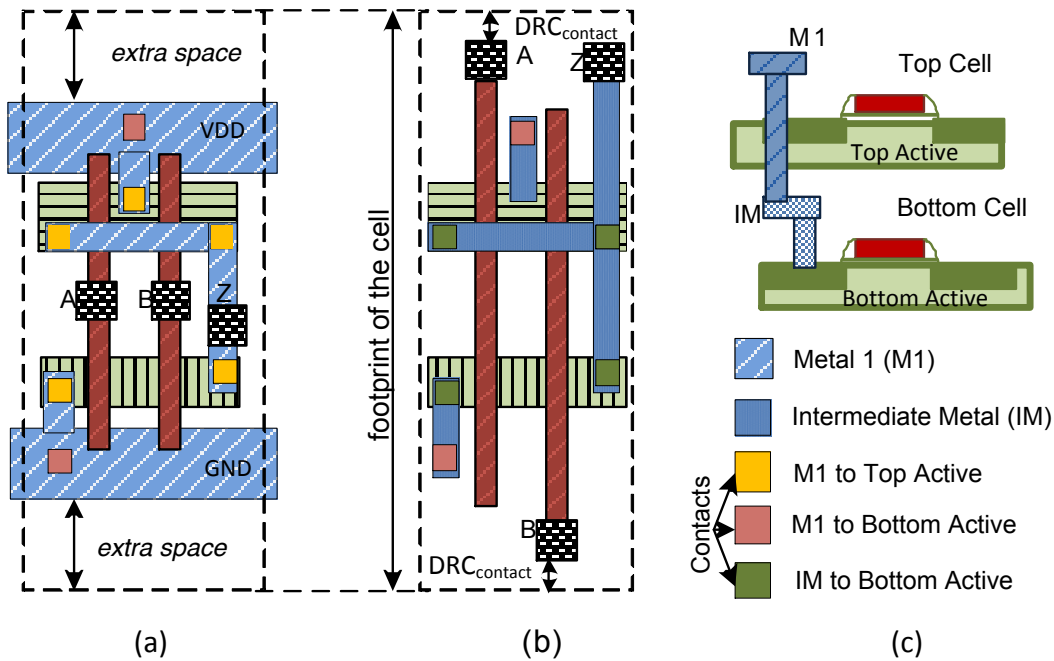


Figure 2.15: CELONCEL NAND2 layout, (a) cell realized in the top active layer (b) corresponding cell in the bottom active layer, and (c) Cross-sectional view of the two active layers with the metals (IM and M1) to realize the cells.

of the top cell are placed in between the power and ground rails (VDD and GND rails in Fig. 2.15). Whereas, the IO pins of the bottom cell are placed beyond the rails. Hence the cell height (or footprint) has to consider the additional space for the IO pins coming from the bottom cell and also the respective design rule for avoiding conflicts (DRC_{contact}) with the IO pins of the neighboring cell. This leads to an increase in the standard cell height.

2.4 Planar-to-3D Library Mapping

Until now I have explained various cell-transformation techniques. In this section, I focus on the implementation details of these transformations. Figure 2.16 shows the three approaches to realize a 3D standard-cell library from a 2D library. Table 2.3 compares the standard-cell height of existing 2D standard cell libraries before and after the cell transformation. I have benchmarked across three important cell libraries at 45 nm and 65 nm technology node. The *intra-cell* folding transformation does not have any impact on the height of the standard cell, however only affects the width of the cells. Table 2.4 presents the percentage improvement in width of the folded cells when compared to the 2D cells, while mapping the 45 nm Nangate 2D cell library [88].

Few key observations from 2D to 3D cell transformation:

- By *intra-cell* stacking, all the cells are spread across two active layers, thereby making a 3D cell library. On average, I observe 30% reduction in the standard cell height with *intra-cell* stacking. The height of the standard cell is directly related to the footprint of the circuit. Hence a 30% reduction of the cell height leads to almost 30% reduction in the overall area. However, in current technologies the performance of a circuit is more important than the area. With 3D cells, I envisage significant decrease in the interconnect length as an outcome of the reduced footprint. One of the primary advantages of this transformation is the ease of integration with existing design flows, since the only design effort required is building the 3D cell library. The CAD part for realizing the logic-to-layout (RTL-to-GDSII) flow does not need any alteration, as the physical design tool when solving the placement problem models the cells as rectangular boxes with the IO pins located at the center of the box. In other words the placement tool does not differentiate a 2D cell from a 3D cell.
- With *intra-cell* folding, the cells are built in a 3D manner by folding the width of the cell while keeping the height constant. From Table 2.4 we can see that the gain in width depends on the type of cell, fan-in and fan-out of the cell. Consequently this transformation cannot be justified for small cells (e.g., inv, nand2, nor2...etc.). However, maximum area gain can be achieved for complex gates (e.g., flipflops) and cells with

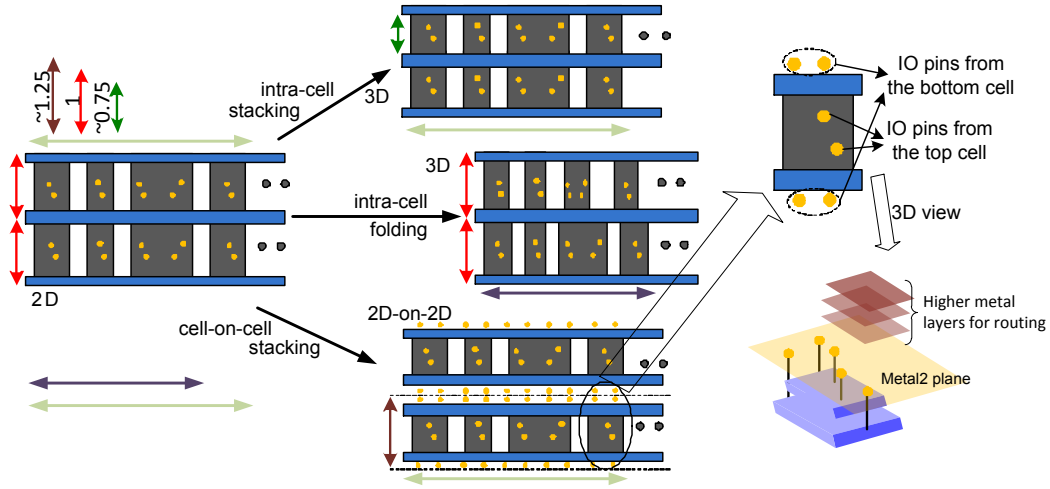


Figure 2.16: 2D to 3D Cell Transformation.

Cell Height	45nm Nangate library	45nm commercial library	65nm commercial library
Planar (2D)	100 %	100 %	100 %
Intra-cell stacking (3D)	71.43 %	71.61 %	69.05 %
Cell-on-cell (2D-on-2D)	125.71 %	125.93 %	125.0 %
Intra-cell folding (3D)	100 %	100 %	100 %

Table 2.3: Normalized height of existing standard cell libraries before and after cell transformation.

high driving strength. Hence, the total area gain (the sum of the device area and the metal routing area for a benchmark circuit) is not uniform unlike with *intra-cell* stacking transformation. The physical design flow for handling these cells is similar to the *intra-cell* stacking case, where traditional 2D placement tools can be employed.

- *Cell-on-cell* stacking leads to 25% increase in the cell height. However, in this case all the cells occupy one active layer and, therefore, one cell can be placed on top of the other. Hence, with 25% increase in the footprint of the cell, we can accommodate 2X the number of cells in the two active layers. Moreover, the number of the neighboring cells is doubled as compared to the 2D or *intra-cell* implementations. Figure 2.17 shows the cells with their immediate and next neighboring cells for all the

Standard Cells	Width gain (%)	
	<i>Low drive (2X, 4X)</i>	<i>High drive ($\geq 4X$)</i>
inv	0	33%
nand2 / nor2	33%	40.62%
nand3 / nor3	25%	42.86%
and3/or3	40%	40%
aoi21 / oai21	25%	42.86%
aoi22 / oai22	40%	44.44%
SD-flipflop	50%	50%

Table 2.4: Percentage improvement in width of the standard cells before and after the folding transformation.

cases. The design effort for *cell-on-cell* stacking is higher as the number of cells is doubled, for the top and bottom active layers. Moreover, a new physical design tool is needed to place the cells in multiple active layers.

- For all the above cell transformation techniques we can observe that the IO pin density is increased per unit area. Hence designs with low to moderate routing needs can benefit from these techniques. On the other hand, for design requiring high routing resources, coarse-grain (block-level) partitioning is advisable.

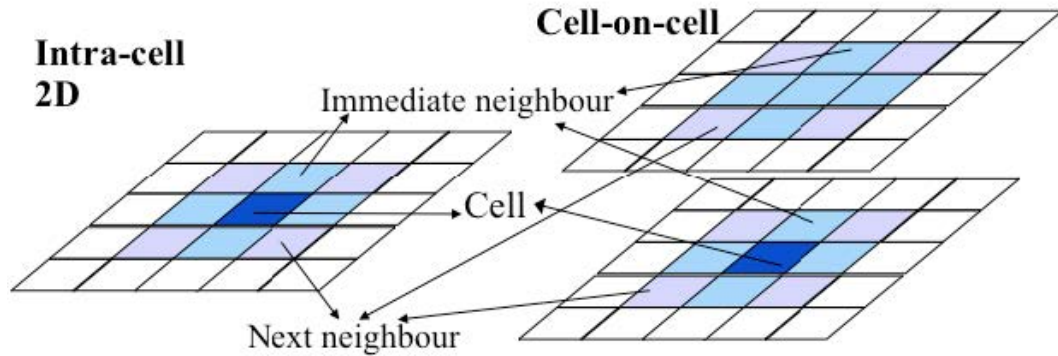


Figure 2.17: Neighboring cells in the case of planar, *intra-cell* and *cell-on-cell*.

2.5 Chapter Contribution and Summary

This chapter sets the background with the state of the art 3DMI technology. 3DMI technology offers 3D contacts in the range of ~ 100 nm, thereby enabling

fine-grain circuit partitioning across multiple layers. At the design level, this thesis sets its focus on 3DMI applied to ASIC designs.

This chapter explores for the first time the cell-transformation techniques specific to fine-grain 3DMI circuits. A novel design technique *cell-on-cell* stacking is proposed, which enables overlapping of planar standard cells on top of each other without any pin conflicts. In addition, this chapter also contributes to studying various other standard cell transformation techniques (*intra-cell* stacking and *intra-cell* folding). All the three cell transformation techniques are analyzed to study the improvement in performance for each of these techniques.

At the standard cell abstraction, the area benefit achieved by all the three cell-transformation techniques are fairly comparable. However, in order to understand the true benefits of these techniques, a complete physical synthesis flow studying big benchmark circuits is needed. Existing 2D placement tools can be employed for both the *intra-cell* design techniques. Nevertheless for *cell-on-cell* stacking, a 3D placement tool is required. The following chapter deals with this aspect, where a novel physical synthesis tool is proposed for *cell-on-cell* stacking.

Physical Synthesis Tool for 3DMI Circuits

3

Chapter 2 proposed various standard cell transformation techniques, which can be employed to realize ASICs. While, intra-cell design technique is compatible with the existing 2D physical synthesis flow, cell-on-cell stacking needs a new placement tool to partition the circuit across two active layers. A novel placement tool, CELONCELPD, targetted to *cell-on-cell* stacking, is proposed in this work. With the help of both INTRACEL and CELONCEL physical synthesis flows, various cell transformation techniques are studied and benchmarked with planar CMOS technology.

The chapter is organized as follows. After discussing the state of the art of 3D placement tools for 3DMI technology, an overview of the complete physical synthesis flow for various cell transformation techniques (see Section 2.3) is presented. Next, our new placement tool for 3DMI circuits (called CELONCELPD) is studied in depth. Then, with the help of INTRACEL and CELONCEL design flows, various cell transformation techniques are benchmarked with planar Si-CMOS technology at 45 nm node. Finally, the chapter is concluded by discussing the results and by overviewing the contributions from this part of the thesis.

3.1 State of the Art

In recent years, there has been extensive work in developing new physical design tools for 3D IC design [41], [42], [43]. However, all these tools are mainly linked to the 3D TSV technology. 3D monolithic integration has seen substantially less research effort at the CAD level. In this thesis, I bring 3DMI technology to ASIC design.

Many authors have solved the placement problem for 3D circuits by incorporating the width of the 3D contact into their wirelength optimization formulation [89]. Hence for wide 3D contacts, as in the case of TSVs (500 - 1000 nm), apart from minimizing the average wirelength of the circuit there is also a need to reduce the usage in the number of 3D contacts. At a CAD level, the only difference between 3D TSV and 3DMI technology is the size of the 3D contact. Hence, one might argue that existing 3D placement tools can be applied to solve the placement problem for 3DMI circuit. However, the idea of one placement tool for all 3D circuits is not practical, as technology details needs to be taken into account. For instance, in 3DMI technology (see Section 2.2) the intermediate metal layer between the active layers is Tungsten, as it has a high thermal coefficient when compared to Copper. However, Tungsten is three times more resistive than Copper, thereby making it more suitable for local routing (for example routing within the standard cell) than for general routing connecting neighboring cells. Additional technology features add to the design complexity, hence we need new CAD tools, especially physical synthesis tools, to bridge the time gap for designers. With this work, I take the first step towards providing a complete design flow for 3D monolithic technology. CELONCEL design flow, comprising of CELONCELPD and CELONCELLIB, can be integrated into the traditional 2D design flow.

Previous research work on 3D physical design adopts existing 2D placement tools for placing cells across multiple active layers [89], [90], [91], [42]. However, researchers have mainly focused on placement for 3D TSV technology with an objective of reducing the estimated wirelength of the placed netlist with an additional constraint on minimizing the number of TSVs. In the work by Deng et al., the authors have adopted CAPO [90] to partition the circuit across multiple layers [89]. By reducing the weight of the TSVs in their problem formulation the authors briefly cover the placement problem for 3DMI technology.

This work differentiates from the existing work in many folds. First, the design technique proposed is closely linked to the current technology. Second, the CELONCELPD presented in this work does not modify the 2D placement engine; however it acts as a wrapper around the 2D placement engine to place standard cells in 3D. For instance, the state of the art physical design tools have been developed and tuned over a decade [91], [92], [90], [93] and separate customization of the tools for different technologies can be very expensive, if not adapted carefully. Compared to academic placers, industrial placers (e.g. Cadence Encounter, IC Compiler etc.) offer complete physical synthesis flow (with steps such as buffer insertion, gate sizing, fanout optimization, repeater insertion etc.) for advanced timing closure. Hence, in this work I build CELONCELPD as a wrapper around the industrial placement engine [93] (Cadence Encounter) to study timing benefits of various cell transformations

techniques related to 3DMI technology. However, the proposed placement technique is also compatible with other existing 2D placement engines.

3.2 Design Flow for Various Cell-Transformation Techniques

Figure 3.1 presents the IC design flow, from logic-to-layout, for *cell-on-cell* and *intra-cell* transformations. Both *intra-cell stacking* and *intra-cell folding* transformation map a 2D cell to a 3D cell. “INTRACEL*stack* Library” and “INTRACEL*fold* Library” shown in the figure correspond to the 3D libraries designed by *intra-cell* stacking and *intra-cell* folding transformations. One of the key advantages of these techniques is the usability of the existing physical synthesis tools. INTRACEL design flow presented in Fig. 6.3b is similar to the conventional 2D design flow, with an extra design effort in building the 3D cell libraries. In the rest of this chapter I refer to INTRACEL*stack* and INTRACEL*fold* as design flows for the two *intra-cell* techniques.

On the other hand, for *cell-on-cell stacking* transformation, new blocks are incorporated into the existing physical synthesis design flow. The CELONCEL design flow is presented in Figure 3.1a. CELONCELLIB is a novel standard library with the cells designed by *cell-on-cell stacking* (see Section 2.3). CELONCELPD has four main steps in the flow. The details of each step are described in the following section. The first two steps, DEFLATE and INFLATE transformations, help in employing existing 2D placement engines as a core placement tool. The physical information of the standard cell library (e.g. LEF file for Cadence tool flow) is altered with DEFLATE transform. The width of the cells is reduced by half. At this stage most commercial/academic placement tools can be used to generate a virtual seed placement without any overlap among the transformed cells. With the INFLATE transform, the width of the cells is doubled in the seed placement result. This generates overlaps among the neighboring cells. The next step is ACTIVEASSN that performs the active layer assignment of the cells. This step reduces the overlap among cells by an order of magnitude. Finally, minimum perturbation legalization is done to remove rest of the overlaps in the step LEGALIZE thus completing the placement.

3.3 Placement Tool: CELONCELPD

From Section 2.4, we observe that existing 2D physical synthesis tools are sufficient for both the *intra-cell* transformations. In this section, I explain the various steps of CELONCELPD, a novel physical synthesis tool for *cell-on-cell* transformation. The key assumption I take forward with the *cell-on-cell* stacking is that the footprint and the delay of the cell, when placed in the top or bottom active layer, does not alter. Based on this, I conjecture that during physical synthesis the choice of active layer for each cell can be abstracted as a purely overlap issue without any

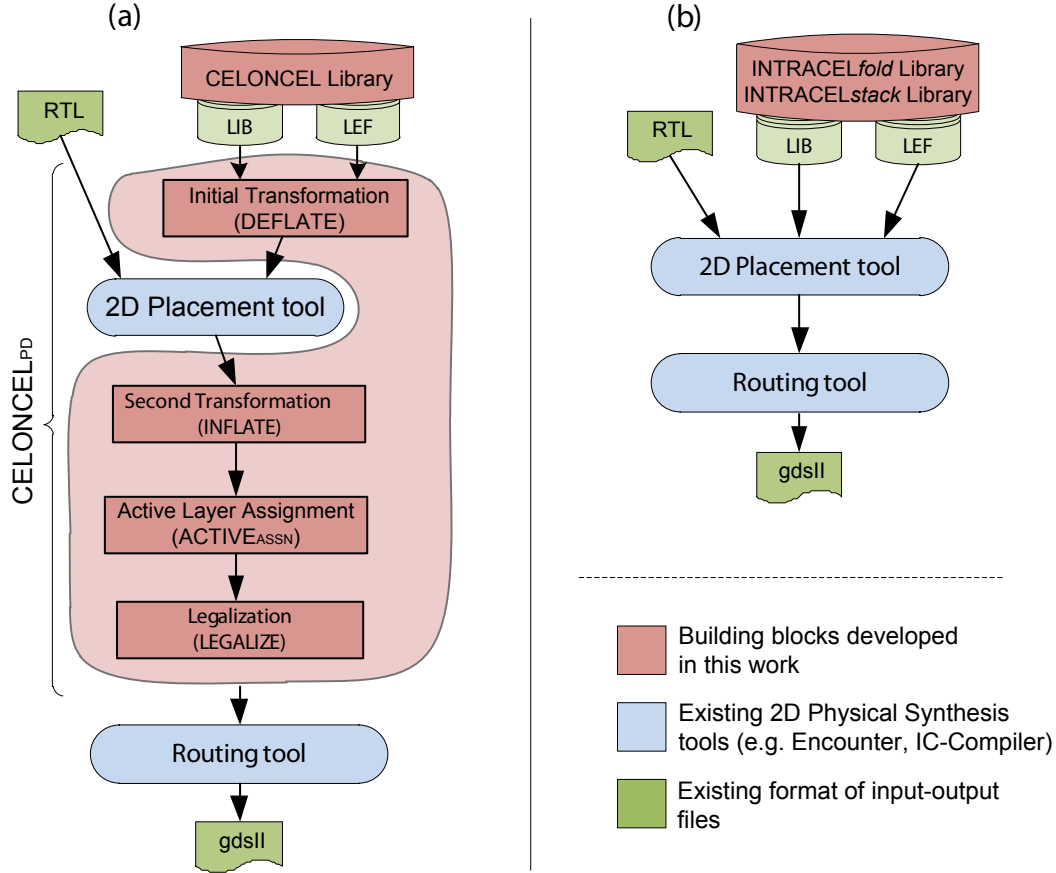


Figure 3.1: Logic-to-Layout Design Flow for (a) *cell-on-cell* and (b) *intra-cell* transformations.

impact on timing of the design. Once the active layer oblivious layout is obtained, the choice of active layer is made by a dedicated step. One of the critical benefits of isolating layer assignment and placement is that several physical synthesis steps that run during in-place timing optimization within placement can be performed transparently. These steps include aggressive buffer insertion, gate sizing, cell replication, clock tree generation, clock buffer placement, latch resizing, etc.

3.3.1 Initial Transformation: DEFLATE

The DEFLATE transformation generates a virtual cell library from a given real cell library such that cell dimension and pin location are modified. Since I consider two active layers in our work, I shrink the width of each cell by half. Note that, to avoid placement errors, I need to scale down the x-coordinates of the pin geometry defined for such a cell. Figure 3.2 shows an example of a 2D cell undergoing this initial transformation. At this stage, I run any 2D placement engine to generate legalized placement consisting of transformed cells. Previous works such as [94], [95] have used the concept of cell expansion/deflation for congestion alleviation and

transforming placement with blockages to contiguous placement respectively.



Figure 3.2: DEFLATE transformation applied to all the library cells.

Algorithm: DEFLATE

Input: *Celoncel.lib*, *Celoncel.lef*

Output: *Virtual.lib*, *Virtual.lef*

for each cell SC in *Celoncel.lib* and *Celoncel.lef*

Scale down the width of SC by 50%

Scale down the pin coordinates of SC by 50%

end

Write modified cells as *Virtual.lef* and *Virtual.lib*

Update verilog to use modified cell variant

/ Virtual.lib and Virtual.lef are employed by the 2D placement engines to do the initial placement of the benchmark circuits */*

3.3.2 Second Transformation: INFLATE

The INFLATE transform takes the placement information from the solution of a commercial placer on the virtual library and applies an inverse transform such that the width of the cells is expanded back to their original size. While doing this expansion, I assume that the *center* of the cell remains fixed. Due to expansion of the width of the cells, it is possible that part of some cells may lie outside the floorplan area. INFLATE also snaps such cells inside the placement area. Once the width of all the 2D cells is doubled, the placement has a huge number of overlaps. All the cells are now placed in only one active layer oblivious of the availability of another active layer. Figure 3.3 shows an example of few cells, placed in two neighboring standard cell rows (i and j), undergoing INFLATE transform. The center of all the cells (for example $O1$, $O2$ and $O3$ in the figure) remain fixed while undergoing the INFLATE transformation. The corresponding overlap and whitespace for both the rows are shown in the Figure 3.3.

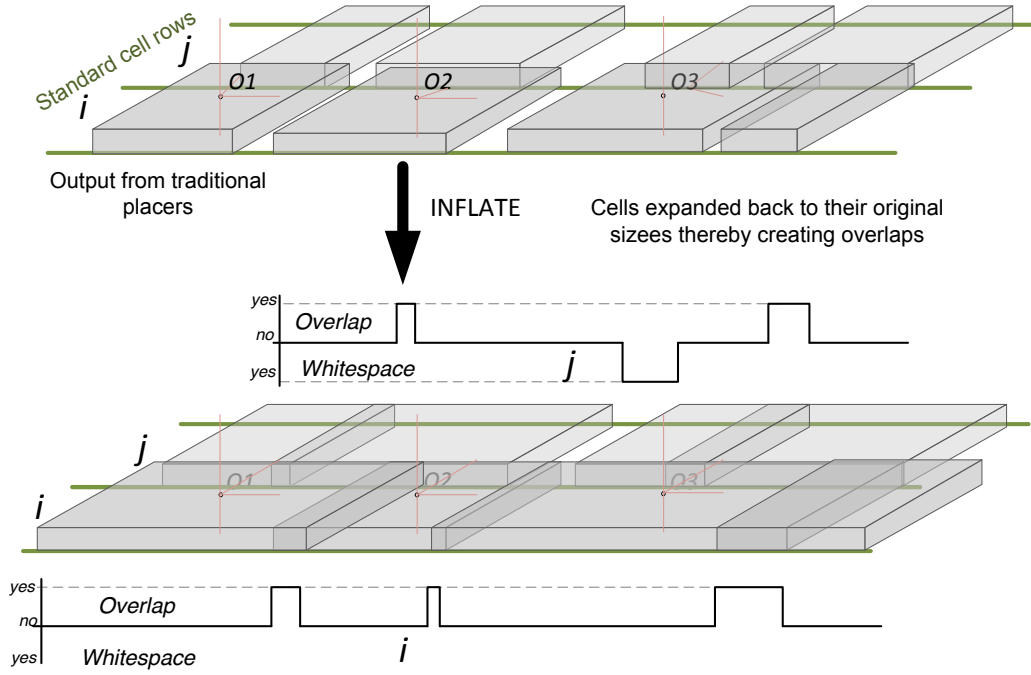


Figure 3.3: INFLATE transformation shown for neighboring standard cell rows (i and j). The width of the cells is doubled, while keeping their centers (e.g. $O1$, $O2$ and $O3$) fixed. Morphing the cell width leads to overlaps and whitespace between the cells.

Algorithm: INFLATE

Input: *initial placement* /* initial placement is the layout from 2D Placement tool for a chosen benchmark */

Output: *inflated placement*

for each cell C in *initial placement*

Scale up the width by 50% keeping its center of gravity fixed

if C not entirely in the die area /* Fix expanded cells protruding the die area */

snap C to be inside the die

end

end

Write modified cells as *inflated placement*

3.3.3 Active Layer Assignment

This step assigns the active layer of each cell with the objective of minimizing the overlap with the neighboring cells. During this stage, I assume that all cells are

fixed in their active area plane at locations determined by the placer and only their z dimension (i.e. active layer) can be modified. This problem can be formulated as a *zero-one linear program* (ZOLP). Solving one large ZOLP for the entire chip is impossible due to runtime issues. However, owing to the structure of the placement and the type of overlaps resulting due to INFLATE transform, I decompose the active layer assignment of all the cells as sequence of active layer assignment of each circuit row independently without sacrificing the optimality of the solution.

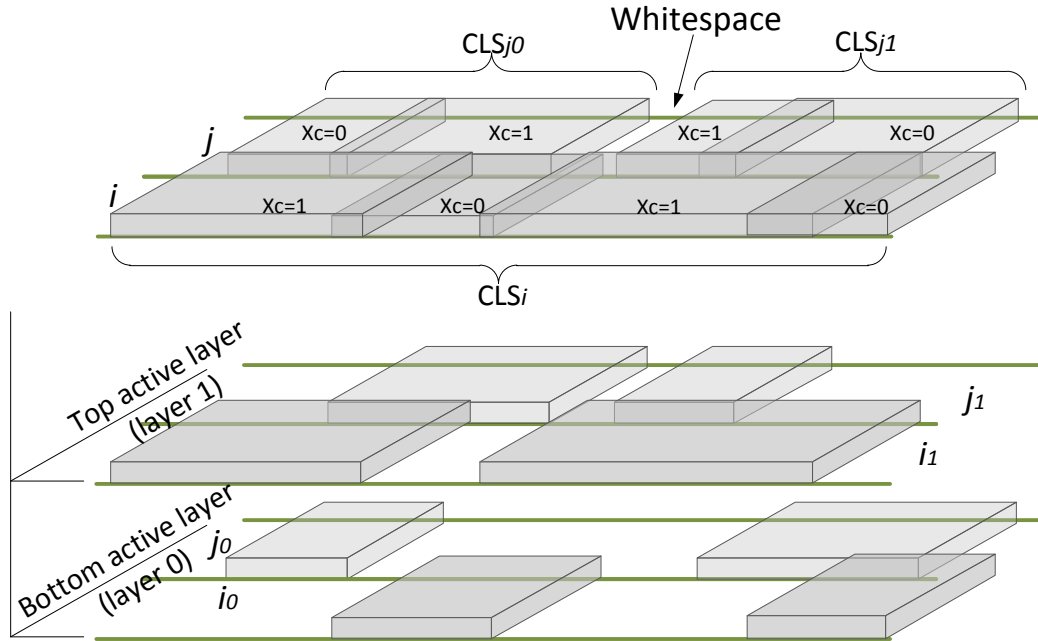


Figure 3.4: Active layer assignment shown for neighboring standard cell rows (i and j). Overlap between the cells is removed by assigning the cells to different active layers with the help of the ZOLP formulation. Whitespace between the cells helps in forming small clusters to speed up the ILP.

Algorithm: *ACTIVE*_{ASSIGN}

Input: *inflated placement* /* initialPlacement is the layout from the INFLATE transform */

Output: *Place_layer0, Placer_layer1*

for each row *R* in *inflated placement*

let CELLS be the cells in *R*

Scan CELLS from left to right creating nonoverlapping clusters *C*

end

for each cluster *C* of independent cells

solve ZOLP_minimize_Overlap(*C*) to get active layer coordinates for the CELLS in *C*

end

for each active layer *L* /*2 in our example */

Write the (x, y) coordinates of the Cells assigned to *L*

end

The objective function to minimize is the remaining overlap after active layer assignment is performed. A small remaining overlap directly means less movement of cells from their optimal location, determined by the placer, during the legalization step. Consider a floorplan with N standard cell rows of width W_{row} . Let us denote the set of cells laying in a circuit row i by C_i . Further, let $OV(a, b)$ denote the 2D overlap between two cells a and b in the row. For each cell a , let X_a be the binary variable whose value determines the active layer in which the cell a will reside in the 3D layout, and W_a be the width of the cell a . With this terminology, the ZOLP can be formulated as:

$$\begin{aligned}
 \min \quad & \sum_{i=1}^N \left(\sum OV(c1, c2) (X_{c1} \bar{\oplus} X_{c2}) \right) & (c1, c2 \in C_i) \\
 \text{s.t.} \quad & \sum X_c \times W_c \leq W_{row} & c \in C_i \\
 & \sum (1 - X_c) \times W_c \leq W_{row} & c \in C_i \\
 & X_c \in (0, 1)
 \end{aligned}$$

The possible overlap between two cells is multiplied by the XNOR ($\bar{\oplus}$) of the binary variables associated with their layer assignment. Thus, only when the two cells are assigned to the same active layer, the corresponding overlap value adds to the cost function. The two set of constraints of the above formulation are to bound the cells within the footprint of the standard cell row in which they are placed. Figure 3.4 shows the active layer assignment of the inflated placement from Figure 3.3. The overlap between the neighboring cells is removed by spreading the cells across the bottom and top active layers. The cells in the row $i0$ and $j0$ are assigned to the bottom active layer and cells in the row $i1$ and $j1$ are placed in the top active layer. Note that XNOR implies multiplication of two variables thus the formulation is no longer linear but quadratic in nature. However, by virtue of the variables being binary, each quadratic term can be decomposed into linear terms by adding an auxiliary binary variable as follows. Let X_A and X_B be the two binary variables whose product (i.e. $X_A \times X_B$) appears in the cost function expression. Introduce a

new binary variable X_{AB} such that:

$$\begin{aligned} C1 : X_A + X_B &\leq 1 + X_{AB} \\ C2 : (1 - X_A) + (1 - X_B) &\leq 2 - 2 \times X_{AB} \end{aligned}$$

By replacing $X_A \times X_B$ by X_{AB} , and adding the above constraints to the ILP, the new problem formulation avoids multiplication of binary variables. For example, when $X_A = 0$ and $X_B = 0$; $X_A \times X_B = 0$. Constraint C1 leads to $0 \leq 1 + X_{AB}$ i.e. $-1 \leq X_{AB}$. This does not force X_{AB} to a unique value, both $X_{AB} = 0$ and $X_{AB} = 1$ satisfy the equation $-1 \leq X_{AB}$. With the constraint C2, when $X_A = 0$ and $X_B = 0$, we have $X_{AB} \leq 0$. Hence with the two constraints, C1 and C2, binary variable X_{AB} is similar to $X_A \times X_B$. A truth table with the various combinations of X_A and X_B is presented in Table 3.1.

X_A	X_B	$X_A X_B$	X_{AB}		
			C1	C2	$C1 \cap C2$
0	0	0	0, 1	0	0
0	1	0	0	0	0
1	0	0	0	0	0
1	1	1	1	1, 0	1

Table 3.1: Truth table

ZOLP Speed Up: The number of binary variables in the ZOLP above is equal to the number of cells in a circuit row. For big benchmarks and real world designs, this number can be in the order of several thousands. To alleviate this problem, I decompose the ZOLP problem by finding independent clusters as follows. I scan the layout of a row from left to right. Any time a whitespace is encountered, the ZOLP problem of the cells on left of the whitespace is solved independently to the ZOLP problem of the cells on the right. This is because during the active layer assignment cells cannot move in the 2D plane thus the cells on both sides of a whitespace cannot generate new overlaps between them and can be treated independently. For example, in Figure 3.4, two independent clusters (CLS_j0 and CLS_j1) can be identified in the row j (CLS_j) formed by the whitespace separating both the clusters.

3.3.4 Legalization

Major overlaps are minimized in the layer assignment phase. However some overlap may still remain, mainly due to different sizes of the cells. I perform legalization to remove these overlaps by minimizing the cost function that is the total displacement of all cells in their own active layer from the optimal location determined by the placement tool (note that ACTIVEASSN maintains the location of the cell). For this objective, the problem can be decomposed into solving each row independently without loss of optimality of the overall solution. For each row, legalization can be cast as a linear program as described next. Let us denote the set of cells lying in a circuit row on active layer 0 by CLS_0 and active layer 1 as CLS_1 . Further, let the original and post-legalization x-location of cell a be denoted by $XO(a)$ and $X(a)$

respectively. Thus, the magnitude of movement of the cell is $(X(a) - XO(a))$ due to legalization. Note that during legalization no cell changes its circuit row or active layer therefore the y and z coordinate of each cell do not change due to legalization. I also denote the width of cell a by $W(a)$ and the cell on its right side on the same active layer by $RT(a)$. The leftmost and the rightmost cell in the row are denoted by $L0$ and $R0$ for the bottom active layer, $L1$ and $R1$ for the top active layer. The x-coordinate of the left and right extreme of the span of the row is represented by $START$ and END . With this terminology, the LP for legalization can be written as:

$$\begin{array}{ll}
\min & \sum |X(a) - XO(a)| \quad \forall a \in CLS_0 \cup CLS_1 \\
\text{s.t.} & X(a) + W(a) \leq X(RT(a)) \quad \forall a \in CLS_0 \\
& X(L0) \geq START \\
& X(R0) + W(R0) \leq END \\
& X(a) + W(a) \leq X(RT(a)) \quad \forall a \in CLS_1 \\
& X(L1) \geq START \\
& X(R1) + W(R1) \leq END
\end{array}$$

The cost function is the sum of the displacement of all cells. The formulation can be easily changed to minimizing the largest displacement (instead of current form to minimize total displacement). There are two sets of constraints for the LP, one for each active layer. Though the function $(X(a) - XO(a))$ is non linear, the above LP can still be solved by replacing the function by a variable $MOVE_a$ and the following constraints can be added:

$$\begin{array}{ll}
X(a) - XO(a) & \leq MOVE_a \\
X(a) - XO(a) & \geq -MOVE_a
\end{array}$$

Adding the above constraints forces the variable $MOVE_a$ to behave like the absolute distance between $X(a)$ and $XO(a)$ when the objective is to minimize $X(a) - XO(a)$.

After the LP has been solved and the critical cells enlarged, there may be overlaps between the expanded critical cells and their neighboring non-critical cells. This overlap needs to be removed through legalization. In view of the general philosophy of perturbing the minimum amount of interconnects and cells, I propose Algorithm (Legalize) which remove overlaps while shifting the least number of cells by minimum displacement to get a legalized placement.

Algorithm: Legalization

Input: *Place_layer0*, *Place_layer1*
Output: *legalPlace_bot*, *legalPlace_top*
for each each active layer *L*
 Write LP for Legalization of *L*
 Solve LP to get new coordinates
end
 Run legality checker to ascertain legal layout
 Write the *def* file for bottom and top active layer

3.4 Experimental Setup

The core components shown in Figure 6.3 were written in C++ and compiled with g++ 4.4.4. I used open source MILP solver Gurobi [96] as our ZOLP and LP solver engine. Synopsys Design Compiler (A-2007.12-SP4) [97] was used for mapping the RTL of the benchmarks onto the target standard cell library. Cadence SOC Encounter (v8.1) [93] was used as the physical synthesis engine to generate the virtual seed placement. Timing analysis was performed with Synopsys PrimeTime (D-2009.12-SP2) using the capacitance table of the standard cell library. In this study I have mapped the open-source 45nm Nangate (v1.3) [88] library to different 3D libraries by changing only the physical attributes of the cell. For a fair comparison to study the interconnect delay for all the four cases (2D and all the three 3D variants) I assume similar delay characteristics for all the cells while the physical attributes vary depending on the layout style. INTRACELstack has cells, built in 3D by *intra-cell* stacking transformation, with 30% less height. INTRACELfold has cells built in 3D where the width of the cells is altered as per the discussion presented in Section 3.3. CELONCELLIB has cells, which are 2D cells with a capability of either accommodating a cell on the top or below, that span 25% more in height.

To evaluate the performance of the various cell transformations for 3DMI technology, I used a broad range of designs, from interconnect dominated circuits, such as *low density parity check* (LDPC) decoder, to the complex synthetic design b19, comprising of almost 100K nets. The majority of the designs are obtained from the open-source design library [98], while the big synthetic design b19 is taken from the ITC99 suite [99]. The design parameters are given in Table 3.2, which report the number of nets, cells, and pins in the input RTL of the benchmarks.

The last column (Dmin) indicates the minimum possible delay achievable if no changes in the circuit netlist are allowed during placement. This value was obtained by performing timing analysis of the benchmark with the value of interconnect resistance and capacitance set to zero. In the absence of any netlist change (i.e. resizing, buffering, logic duplication etc), the virtue of a placement can be gauged

by observing how closely the post-placement timing, tracks D_{min} for the circuit. Note that, if netlist changes are allowed, the physical synthesis engine can achieve delays lower than D_{min} . However, in that case the number of nets, cells, and pins can change.

CELONCELPD is configured in three modes: in the first mode, *wirelength-driven placement* is run, in the second mode, *timing-driven placement* is run, and in the third mode, *timing-driven optimization* along with *in-place optimization* is run which performs various optimizations such as buffer insertion, gate sizing, cell replication, etc. Note that D_{min} sets the starting seed value for timing optimization. In order to check the runtime of the complete CELONCELPD design flow, I have created bigger benchmarks (around 2M gates) with multiple instances of the existing benchmarks (Table 3.5).

<i>Benchmark</i>		#Nets	#Cell	#Pins	D_{min} (ns)
Circuit	function				
LDPC	Low Density Parity Check decoder	48K	44K	4100	6.904
WbC	Wishbone Interconnect Matrix IP core	29K	27K	2546	2.382
B19	Synthetic design	99K	87K	77	4.305
Ethernet	Ethernet	43K	42K	210	14.738
Des	Data Encryption Standard	59K	56K	298	2.532

Table 3.2: Characteristics of benchmarks used in our experiments. The columns denote the number of nets, cells, and pins. D_{min} gives the delay of the circuit under ideal interconnect conditions (with Resistance and Capacitance set to zero).

Cell characterization

In this work, I assume similar device characteristics for all the active layers. For example, consider a 45nm thin-box silicon-on-insulator technology for the top and bottom active layers. This assumption facilitates us to evaluate the impact of interconnect parasitic on the cell delay. In the case of *intra-cell* transformations, the intermediate-metal is not employed for routing the cell. Hence the delay of the 3D cell is similar to the 2D cell. However, the physical attributes of the 3D cell differ from the respective 2D cell depending on the applied *intra-cell* transformation technique (either *intra-cell* stacking or *intra-cell* folding). In the case of *cell-on-cell* transformation, every cell (X) in the standard cell library has two versions, X_{top} (cell X placed in the top layer) and X_{bottom} (cell X placed in the bottom layer). Hence the number of cells in the standard cell library is doubled. All the bottom version of the cells employ highly resistive intermediate-metal (Tungsten) layer for cell-routing, whereas the top version of cells employ regular metal (Copper) for *intra-cell* routing. I characterized few complex cells by taking into account the

layout parasitics (using Calibre xRC [100]). I observe negligible delay degradation (less than 0.1%) for the bottom cell compared to the top version of the cell because of the high resistive intermediate metal layer. This agrees with the fact that the impact of local interconnect on the delay of the circuit is minimal. Hence in this work, I assume similar delay characteristics for the top and bottom cells. It has to be noted that the footprint of the top and bottom cell is kept the same (see Section 2.3).

3.5 Results and Discussion

In this section, I present the performance improvement when mapping a 2D circuit to 3DMI technology with three cell transformation techniques discussed in this thesis, namely INTRACELstacking, INTRACELfolding and CELONCELstacking. Firstly, I study the performance gain in terms average wirelength after placing the circuit. Similar to the existing 3D placement tools, based on TSV technology, I run the CELONCELPD in wirelength-driven placement. Secondly, I study the improvement in the timing of the circuits, when driven in timing-driven placement mode. With the help of in-place optimization mode, I also study the timing improvement after physical synthesis techniques like buffer insertion, gate sizing, repeater insertion, and cell replication.

3.5.1 Area Comparison

In the first instance, I compared the area gain of the various 3D transformations with the 2D case. Figure 3.5 shows the percentage improvement in total chip area. Note that the area analysis presented here is carried under wirelength-driven optimization mode. *Intra-cell* stacking decreases the cell height by 30%. The reduced cell height reflects in the increase of the number of standard cell rows for a given floorplan. Hence we can observe a 30% area gain. In the case of *cell-on-cell* stacking, the cell area is increased by 25%, however, we have twice the floorplan area to stack the cells on top of each other. Hence, an overall chip area improvement of 37.5% is achieved with *cell-on-cell* stacking. On the other hand for *intra-cell* folding transformation, the decrease in area depends on the type of the cell and its respective fan-in and fan-out (see Section 3.3). Reduction in the area of the benchmark circuit depends on the number of area-efficient cells in the synthesized netlist. For example the best transformation technique for Ethernet circuit is *intra-cell* folding as reflects in the circuit depends on the number of area-efficient cells in the synthesized netlist. For reduced wirelength as well as improved timing. From Figure 3.5, I observe that with the CELONCEL flow we achieve better area gain when compared to the INTRACEL flow for most of the designs, with an only exception of Ethernet benchmark. Among the two *intra-cell* techniques, *intra-cell* folding fairs to be a better choice.

3.5.2 Wirelength-driven Placement

In the *wirelength-driven* placement mode, the physical design tool places the cells of the given netlist in such a way that the average interconnect length is minimized. Table 3.3 reports the average wirelength for the various benchmark circuits. When comparing all the benchmark circuits, it can be inferred from the table that interconnect plays a dominant role in LDPC decoder. Though the number of cells and nets are similar for LDPC and ethernet circuits (see Table 3.2), the average wirelength for LDPC circuit after placement phase is 3.5% higher than ethernet. Percentage improvement in wirelength, for all the benchmarks, when employing *intra-cell* and *cell-on-cell* design techniques is plotted in Figure 3.6. In general, the results for *cell-on-cell* are better than *intra-cell* techniques. The average improvement in wirelength over a 2D case employing CELONCEL, INTRACELstack, and INTRACELfold are 16.2%, 10.5%, and 13.9% respectively.

3.5.3 Timing-driven Placement

In the timing driven placement mode, the placer is allowed to move the cells to reduce timing without changing the netlist in any manner. Simulation results for timing-driven optimization are summarized in Table 3.4. In this table I report total wirelength, total power, and critical path delay of different benchmarks. All numbers are reported using Cadence Encounter (EDI) v9.1 (2010 release). The power numbers include all components of the power dissipation namely leakage power, switching power, and internal power. Due to smaller die sizes when CELONCEL or INTRACEL flows are used, I conjecture that critical path delay should also decrease accordingly. Averaged over all benchmarks, the critical path delay of the circuit using the CELONCEL flow is 6.1% smaller than the 2D circuit. However, the INTRACELstack does not exhibit any consistent trend compared to the 2D case with the average improvement in the critical path delay by less than 1%. On the other hand, INTRACELfold shows consistent gains similar to CELONCEL case with an average improvement of 5%. For this set of experiments,

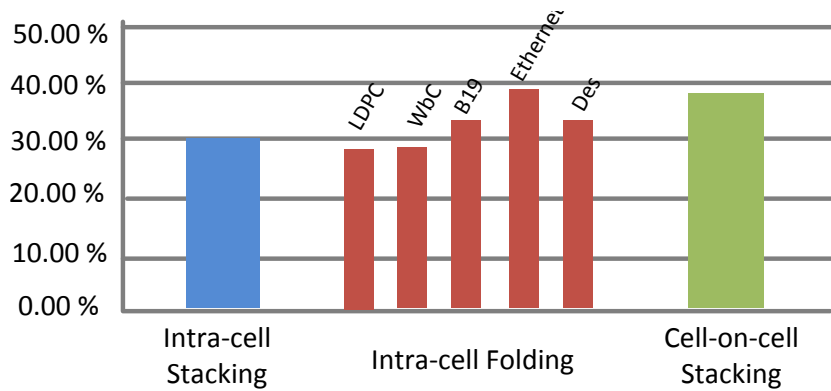


Figure 3.5: Percentage improvement in the total area for all the cases.

CIRCUITS	PLANAR (2D)	INTRA-CELL STACKING	INTRA-CELL FOLDING	CELL-ON-CELL
LDPC	15.4E+05UM	13.8E+05UM	15.0E+05UM	13.7E+05UM
WbC	3.70E+05UM	3.53E+05UM	3.31E+05UM	3.24E+05UM
B19	8.29E+05UM	7.24 E+05UM	6.99E+05UM	6.89 E+05UM
ETHERNET	4.21 E+05UM	3.72 E+05UM	3.30E+05UM	3.43 E+05UM
DES	5.84 E+05UM	5.08 E+05UM	4.74+05UM	4.54 E+05UM

Table 3.3: Improvement in Wirelength of the benchmark circuits subjected to *wirelength-driven* optimization.

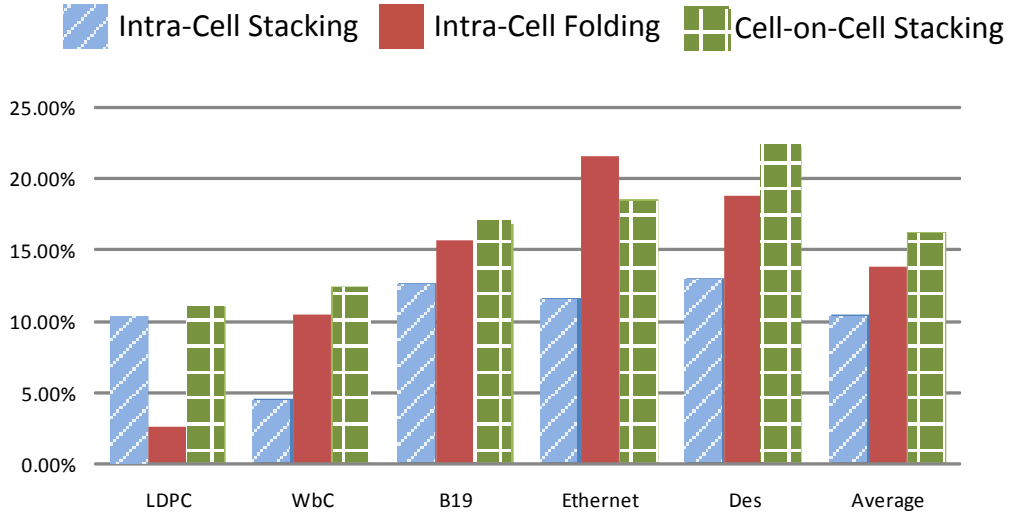


Figure 3.6: Performance improvement in wirelength of various benchmark circuits when subjected to *wirelength-driven* placement.

the timing constraint for each benchmark was set to be equal to the theoretical maximum performance that can be achieved. The maximum performance is obtained by setting interconnect resistance and capacitance equal to zero and running the timing analysis.

The percentage improvement in performance (wirelength, delay, and power) for all the benchmarks is plotted in Figure 3.7 and Figure 3.8. From Figure 3.8, I observe that the timing optimization has almost no impact in the case of LDPC decoder. The 2D and 3D cases achieve very similar delays. This could be attributed to the dominance of interconnect for LDPC circuit. Though the delays are similar, the overall power consumption is reduced for all the 3D cases when compared to the 2D case (shown in Figure 3.8). For instance with *cell-on-cell* transformation, LDPC decoder consumes 10.5% less power compared to the 2D case.

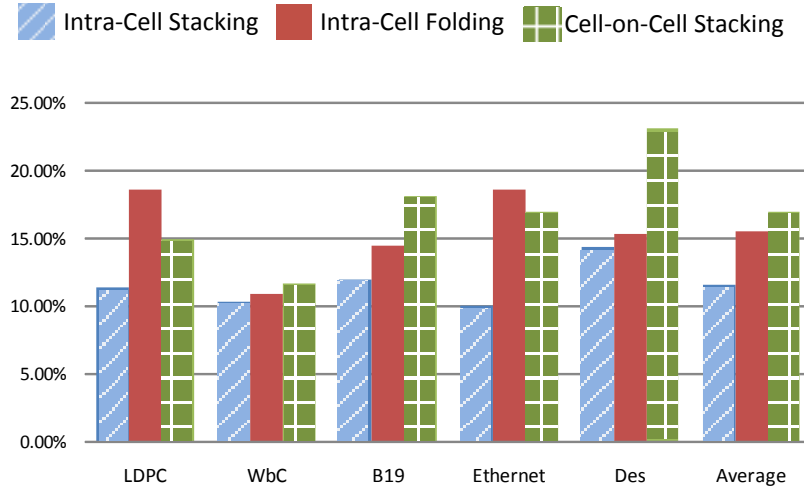


Figure 3.7: Performance improvement in wirelength of various benchmark circuits when subjected to *timing-driven* placement.

3.5.4 Timing-driven with in-place Optimization

For completeness, I have also looked into the timing driven placement with *in-place optimization*. This case study showcases the adaptability of the CELONCEL design flow with the existing 2D placement engines. With *in-place optimization*, the placer has the flexibility to apply any synthesis or timing optimization transforms to the netlist on the fly to improve the timing. For these set of experiments, I set the timing constraint corresponding to an unachievable speed (10 GHz). In this manner, I test the best performance that each of the techniques can produce. Compared to the 2D case, employing CELONCEL can reduce the critical path delay even further by 2.75%. Similarly, by using INTRACEL, the critical path delay can be reduced by approximately 2.7%. Note that this improvement in critical path delay is additional to the best solution obtained using the 2D case, thus hard to obtain.

Figure 3.9 shows the reduced delay of LDPC circuit with in-place optimization in all the cases (2D and 3D cases). With *cell-on-cell* transformation the minimum possible delay of 2.129 ns is realized. Though the 2D case has a slight better delay over *cell-on-cell* in the timing driven mode, I see better delay characteristics with *in-place optimization*. The reason could be related to the double the amount of neighboring slots with *cell-on-cell* stacking. Since more neighboring cells can be accommodated next to each other (see Figure 2.17). With *cell-on-cell*, I speed up the circuit by 13.49% when compared to the 2D case.

3.5.5 Runtime of CELONCEL Placer

All benchmark are run on an Intel Xeon CPU X5650 Linux workstation running at 2.67GHz. The runtime of the CELONCELPD, running on a single thread, for various benchmarks in timing driven mode is shown in Table 3.5. The total

Circuits	Metrics	Timing Driven Optimization					Timing Driven In-Place Optimization				
		Planar (2D)	IntraCell Stacking	IntraCell Folding	Cell-on-Cell		Planar (2D)	IntraCell Stacking	Intra-Cell Folding	Cell-on-Cell	
LDPC	Wirelength(um)	1.67E+06	1.48E+06	1.36e+06	1.42E+06		1.83E+06	1.60E+06	3.02E+06*	1.54E+06	
	Delay (ns)	6.877	6.904	6.866	6.904		2.461	2.421	4.692 *	2.129	
	Power(mW)	1147	1064	1092	1018		1554	1461	2058 *	1470	
WbC	Wirelength(um)	3.77 E+05	3.38E+05	3.36e+05	3.33E+05		3.76E+05	3.64E+05	3.34E+05	3.33E+05	
	Delay (ns)	4.661	4.628	4.342	4.449		1.039	1.041	1.096	1.083	
	Power(mW)	100.4	99.93	99.45	99.64		70.63	71.14	70.69	72.05	
B19	Wirelength(um)	8.60E+05	7.57E+05	7.36E+05	7.04E+05		7.93E+05	7.00E+05	6.77E+05	6.49E+05	
	Delay (ns)	4.723	4.691	4.694	4.691		4.224	4.219	4.184	4.185	
	Power(mW)	434.5	429.7	429.1	425.2		337	312.7	325.3	314.7	
Ethernet	Wirelength(um)	4.3E+05	3.87E+05	3.50E+05	3.57E+05		4.94E+05	4.38E+05	3.91E+05	3.97E+05	
	Delay (ns)	30.598	29.063	26.194	26.427		1.252	1.281	1.223	1.336	
	Power(mW)	175.3	170.6	166	165.6		133.2	132.7	137.3	130.7	
Des	Wirelength(um)	6.06E+05	5.19E+05	5.13E+05	4.66E+05		6.71E+05	5.81E+05	22.9E+5*	5.45E+05	
	Delay (ns)	3.854	3.944	3.731	3.39		1.132	0.971	4.021 *	1.016	
	Power(mW)	535.8	525.3	524.8	517.5		620.2	608.2	920.2 *	580.5	

Table 3.4: Wirelength, Delay, and Power information of benchmark circuits for various cases when subjected to *timing-driven* placement with and without *in-place optimization*.

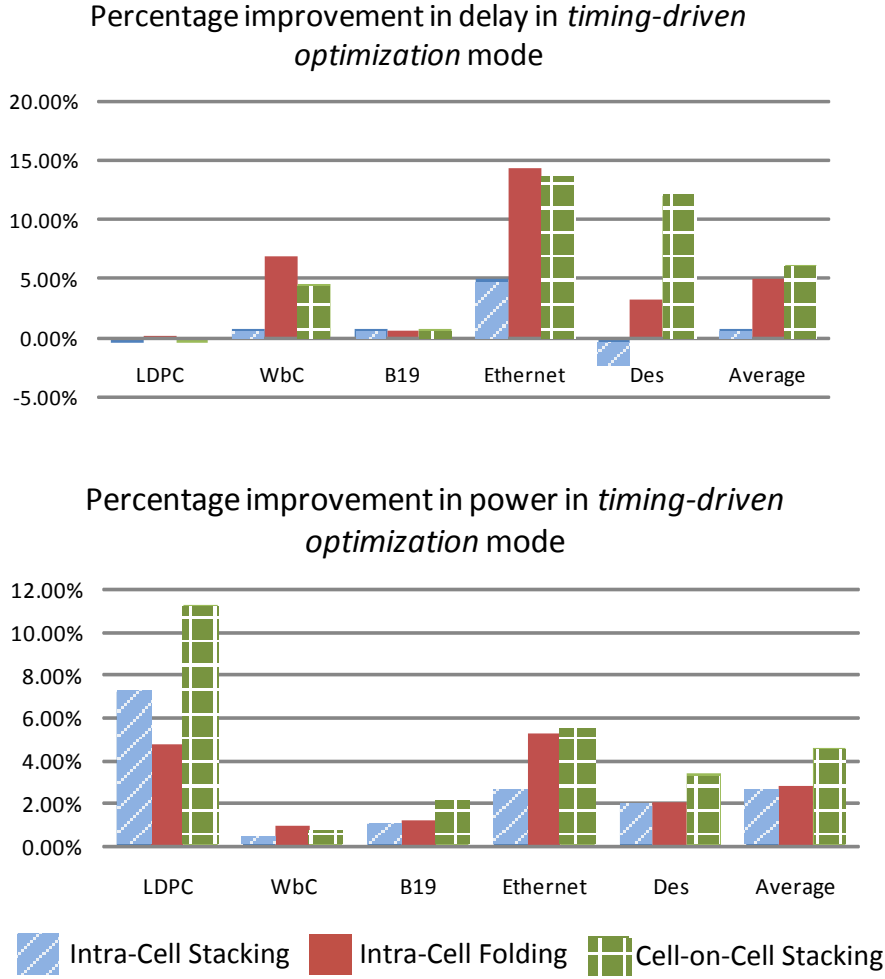
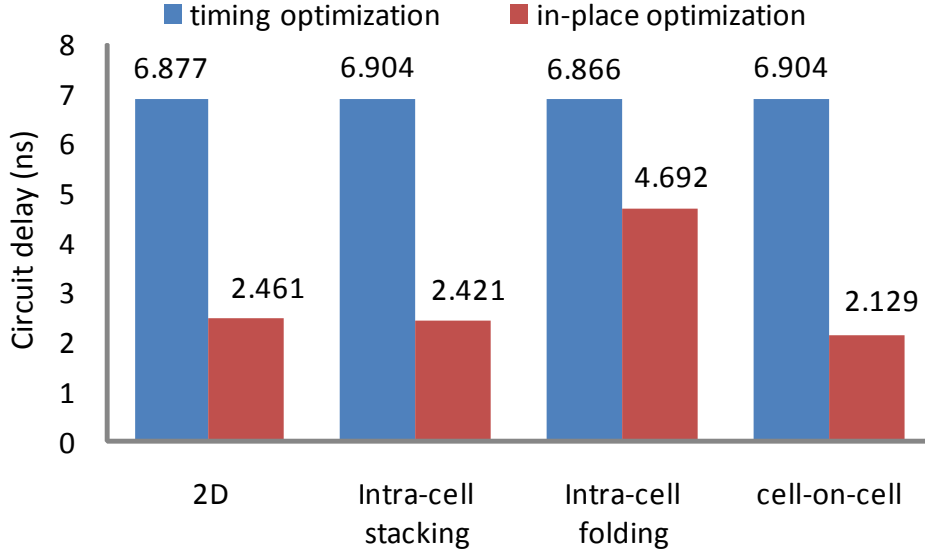


Figure 3.8: Performance improvement of various benchmark circuits when subjected to *timing-driven* placement.

time taken by the CELONCELPD is the sum of the time taken by the 2D engine (Encounter in our experiment) as well as the time taken for solving our ILP formulation for active-layer assignment and legalization steps.

On an average, active-layer assignment and legalizer takes 11.4% of the total time taken for 3D placement. Our flow has also been tested for bigger benchmarks which were created by instantiating many modules of the basic blocks. For the largest benchmark, DES.LDPC.B19_10X (2M), our ILPs were solved in 8 minutes. The runtime benefit comes from clustering only the overlapping cells in each row and solving their respective ILP formulation for active layer assignment and legalization.

Figure 3.9: Delay reduction of a LDPC decoder with *in-place* optimization.

Benchmarks	# of Gates	CELONCELPD		Δt_{3D} %
		2D Engine	ACTIVE _{Assign} + Legalize	
WbC	27K	101.295s	12.96s	11.35
Ethernet	42K	175.623s	23.434s	11.77
LDPC	44K	170.219s	19.892s	10.46
DES	56K	198.331s	24.314s	10.97
B19	87K	285.581s	48.015s	14.39
B19_10X*	870K	3786.416s	515.864s	11.99
LDPC_20X*	880K	6020.149s	445.53s	14.51
LDPC_40X*	1.76M	12506.88s	1189.49s	8.69
DES_LDPC_B19_10X*	~2M	5274.994s	491.493s	8.52

Avg. 11.4%

Table 3.5: Total runtime with CELONCEL placer, which includes the time taken by the 2D placement engine and the time taken by active-layer assignment and legalizer step. Benchmarks with * are made up by instantiating multiple modules into a bigger block. For example B19_10X has 10 instances of B19. DES_LDPC_B19_10X has 10 instances of DES, LDPC and B19.

3.6 Chapter Contribution and Summary

In this chapter, a novel physical synthesis flow (CELONCEL) for fine-grain partitioning of 3D circuits is presented. With CELONCEL design technique, two planar standard cells can be placed on top of each other without any conflicts in the input-output pins of the standard cell. However, this needs two variants of each standard cell, one for the bottom active layer and one for the top layer.

CELONCELPD is a pre-/post-processor for existing 2D placement engines which focus on partitioning the circuits across two active layers. CELONCELPD transforms the monolithic 3D placement problem into a virtual 2D problem solved using existing 2D placers. A zero-one linear program formulation is used for assigning planar cells to multiple active layers. It also encompasses a legalizer for removing the overlap between the cells for each active layer, which allows minimum perturbation in the location of the cells, thereby giving high quality 3D layout.

This chapter also explores circuit level benchmarking of various circuits mapped with planar (technology mapping) CMOS and 3DMI standard cell libraries at 45 nm node. As compared to traditional 2D physical synthesis flow, with CELONCEL (compared to planar implementation) I reduce the wirelength, critical path delay, and the die area by 15%, 6.1%, and 37.5% respectively.

In the near future, co-integration of both 3DMI and 3D-TSV technologies can be envisaged. The design methodology proposed in this chapter studies the physical design technique for fine-grain partitioning of circuits, which is feasible with 3DMI technology and cannot be extended to 3D-TSV technology as the size of the TSVs is large (around 1000 nm). Hence it is beneficial to apply 3DMI technology to leverage the benefits from fine-grain partitioning of the circuit and 3D-TSV technology for benefitting from coarse-grain partitioning. This is the topic of the next chapter.

3.5D Integration: A Cost Effective Scheme for Future MPSoCs

4

Two diverse manufacturing techniques for fabricating 3D integrated systems are vertical integration with *through-silicon-vias* (3D TSV) and *monolithic integration* (3DMI). In this chapter, a hybrid integration scheme is presented that combines these two approaches, taking into account their existing technology limits, into a disruptive paradigm called 3.5D integration. This novel integration supports circuit-partitioning both at the gate and block level showing benefits in performance and cost. To demonstrate the effectiveness of 3.5D integration, a 288-core multi-processor system-on-chip is studied and hypothesis are made on the manufacturing and test cost.

The organization of this chapter is as follows. First, the idea of 3.5D integration is presented, which is followed by a brief overview of *Multi-Processor System-on-Chip* (MPSoC). Next, I extend the concept of 3.5D integration for future MPSoCs and demonstrate various integration schemes considering a 288-core MPSoC as a case study. Then, cost analysis for various integration schemes is studied. Next, performance improvement of an MPSoC is studied by custom technology mapping of the cores and the communication network. Finally, the chapter is concluded by over-viewing the contributions of this chapter.

4.1 3.5D Integration

3.5D integration is a hybrid integration scheme which synergizes the two diverse 3D fabrication technologies based on 3D TSV and 3D monolithic integration. 3.5D integration benefits from fine-grain partitioning of the circuit, with the help of 3DMI technology, as well as the coarse-grain partitioning provided by the 3D TSV technology.

In 3D TSV technology, each active layer, along with its respective interconnect

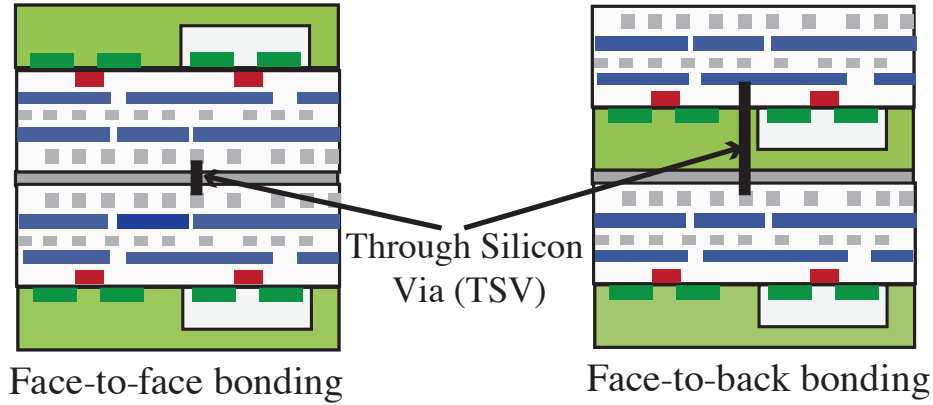


Figure 4.1: 3D TSV Integration [102].

metal layers, is fabricated separately and is subsequently stacked via TSVs [38, 39]. Due to the alignment issues of the stacked dies, the size of the TSVs is kept large (1000 nm) in order to ensure electrical connection between the desired points of the dies. Since the size of the TSVs are relatively high when compared to the size of the transistors, they are only feasible for coarse-grain (block-level) integration. For 3.5D integration, I consider die stacking employing TSVs [101]. Fig. 4.1 illustrates two die-stacking techniques for 3D TSV circuits, where multiple dies are stacked either by *face-to-face* or *face-to-back* bonding [102].

On the other hand, 3D monolithic integration, though in the early stage, is getting attention from various researchers as it promises to provide high-density 3D circuits [47, 103]. Fig. 4.2a illustrates the cross-sectional view of a first-generation industrial 3D monolithic technology, in which p-type devices are realized in the bottom active layer and n-type devices on the top active layer [40]. The two active layers are connected using a 3D contact which is similar to the conventional inter-layer vias. Key benefit of this integration scheme is the reduced active footprint due to small vertical contacts in the range of few 100nm, when compared to TSV sizes in the order of few micrometers. High-density vertical connection is a key feature of 3D monolithic integration as it enables fine-grain (i.e., gate-level) circuit partitioning. In addition, processing n-type and p-type devices in two different layers adds flexibility for separate technological optimization to boost their performance.

3.5D integration leverages the key benefits of both 3D monolithic and TSV integrations. Fig. 4.2b illustrates the synergy between 3D monolithic and TSV integration, where 3 dies are stacked on top of each other. The connection between these three dies is realized with TSVs. On the other hand, each of these dies are realized with 3DMI technology with two active layers. In the rest of this chapter, I study the benefit of this novel integration scheme when applied to multi-processor system-on-chip.

The following is the nomenclature used in this chapter:

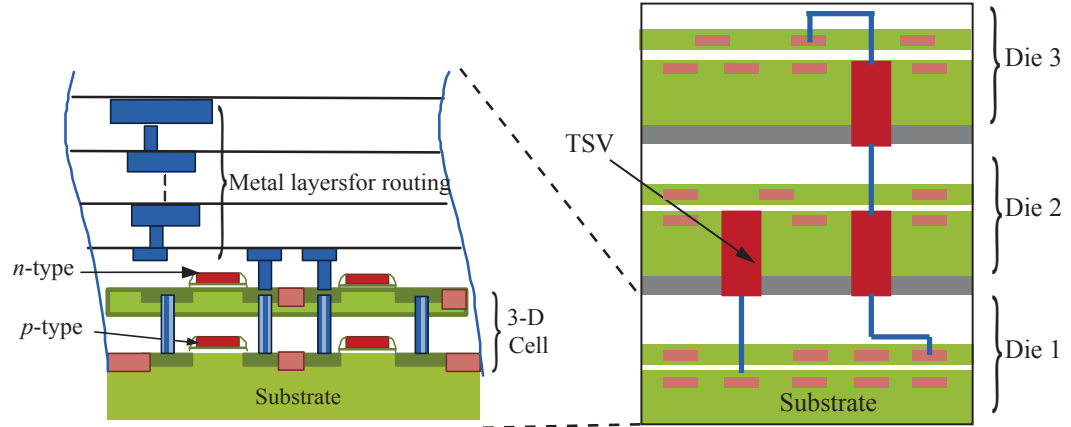


Figure 4.2: (a) Transistor stacking with 3D monolithic. (b) Potential 3.5-D Integration.

- **2D** - Planar technology
- **3D TSV** - Die-stacking with TSVs
- **3DMI n/p** - 3D monolithic integration with n-active layer over p-active layer
- **3.5D** - Synergy of 3D TSV and 3DMI n/p

4.2 Multi-Processor System-on-Chip

A *Multi-Processor System-on-Chip* (MPSoC) is a *system-on-chip* (SoC) with multiple processors among possibly other building blocks. MPSoCs, which inherently provide application-level parallelism, evolved as an architectural choice to overcome the power density limits of single processor systems. In traditional single processor systems, computational power increased for each new generation due to the raise in operating frequency as the transistors in new technology nodes were faster. This leads to increased power densities, which was managed by voltage scaling techniques. However, as supply voltage scaling has a limited range in new technology nodes, computation power can only be increased at manageable power densities. By parallelizing the application onto multiple processors of the MPSoC and by independently controlling the voltage of each processor, MPSoCs yield high performance systems when compared to single processor systems.

MPSoCs usually contain multiple *Processing Elements* (PE) linked by an interconnection network. The PEs of an MPSoC are related to the applications and the type of MPSoC. In the case of *homogeneous* MPSoCs, the PE is a unique tile which is instantiated several times to form a multiprocessor SoC. On the other hand, heterogeneous MPSoCs consist of different PEs (processors, memories, I/O components, and hardware accelerators).

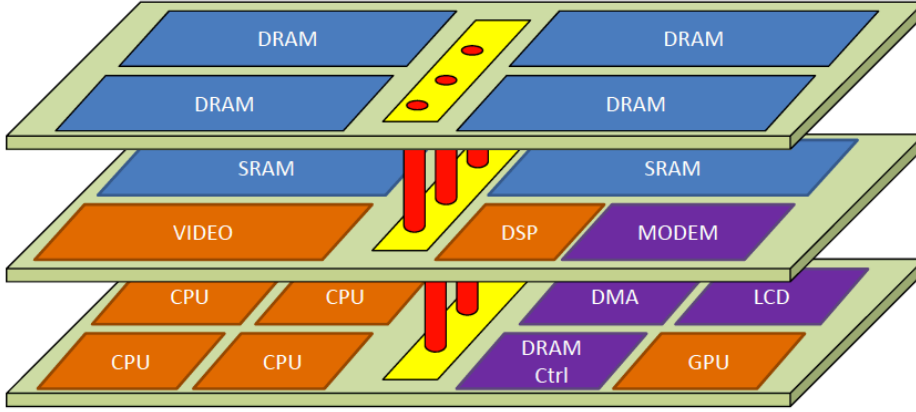


Figure 4.3: Example of a 3D MPSoC with three stacked layers [108].

MPSoCs are communication centric as the processing elements communicate over a global interconnect in order to run the application in parallel. *Network-on-Chips* (NoCs) have been proposed as a solution, and widely adopted, to the interconnect design challenge for MPSoCs [104, 105, 106]. A NoC is composed of *Network Interfaces* (NI), router and links. The NI decouples computation from communication functions by forming an interface between the interconnection environment and the PE. Routers are in charge of arbitrating the data between the PEs through the links. More details on various NoC topologies can be found here [107].

In order to mitigate the interconnect delay at advanced technology nodes, future MPSoCs will integrate multiple layers of active devices on a single 3D chip [12]. Fig. 4.3 illustrates a 3D MPSoC with three stacked layers where the connection between the different active layers is provided by *Through Silicon Vias* (TSVs).

4.3 3.5D Integration for MPSoCs

In this section, I consider mapping a generic MPSoC to various vertical integration schemes described in Section 4.1. For the sake of scalability, all the processing cores of the MPSoC are interconnected by a homogeneous *Network-on-Chip* (NoC) [109] as illustrated in Fig. 4.4. Planar homogeneous MPSoC (depicted to the left of Fig. 4.4) is mapped to 3D TSV, 3DMI n/p and 3.5D integration schemes. Let us assume that when mapping with 3D TSV technology, the planar MPSoC is partitioned into K layers (see Fig. 4.4). In order to interconnect the processing cores across various layers, a vertical extension of the NoC, a 3D NoC [110] is required.

Alternatively, 3DMI n/p integration “folds” the cores by fine-grain stacking of the n-type on top of the p-type transistors, thereby reducing the overall area of the planar MPSoC by 30% (see Section 4.3.1). Though the transistors are stacked in 3D, the architecture of the MPSoC is similar to the planar MPSoC, where a 2D

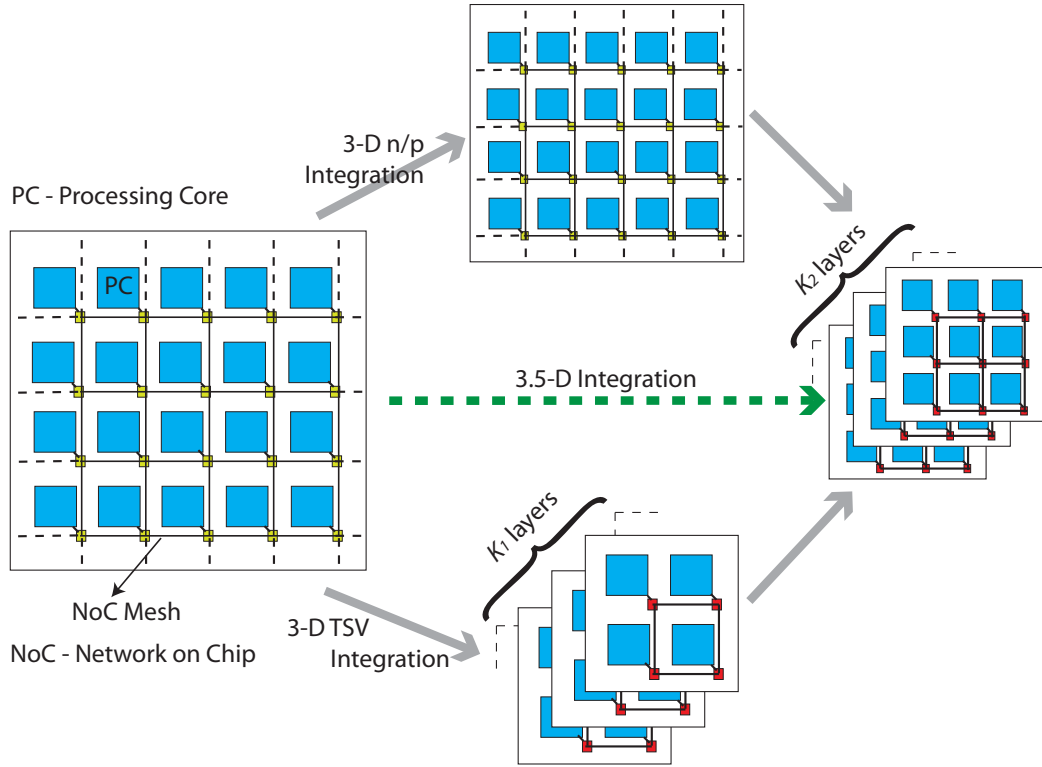


Figure 4.4: Technology mapping from planar 2D to 3D TSV, 3DMI n/p and novel 3.5D integrations.

NoC is still employed to connect the cores.

In 3.5D integration, I envisage technology mapping with both 3D TSV and 3DMI n/p. First, the planar MPSoC is mapped with 3DMI n/p technology, which is further mapped with 3D TSV technology. The 3DMI n/p integration increases the integration density, thereby reducing the footprint of the processing cores. Consequently, more cores can be placed for a given die area. Several of the 3DMI n/p dies are stacked with TSVs to produce a 3.5D multi-processor SoC. Let us assume that mapping with 3.5D integration results in vertical stacking of K_2 layers. As depicted in Fig. 4.4, with the same die size for 3D TSV and 3.5D, we observe 9-cores per die for 3.5D when compared to 4-cores with 3D TSV. This results in fewer layers in 3.5D ($K_2 < K_1$), which affects both cost and system performance and are subsequently studied in the following sections.

Case study of a 288-core MPSoC

In order to study the cost and performance benefits of various 3D integration schemes, I consider a synthetic case study of a 288-core MPSoC. I target the homogeneous cores presented in [111] (suited for a telecommunication system) as a good vector for scalability. Fig. 4.5 illustrates the fully homogeneous processor

array called GENEPEY. A homogeneous platform is defined with a single tile instantiated several times and connected to a homogeneous NoC. Each tile manages its processing resources along with its configuration and its scheduling. SMEP for each tile is a smart memory engine, where as the control processor is a 32-bit MIPS processor which manages dynamic reconfigurations, real-time scheduling and synchronizations. The NI is the network interface which connects the tile to the homogeneous NoC.

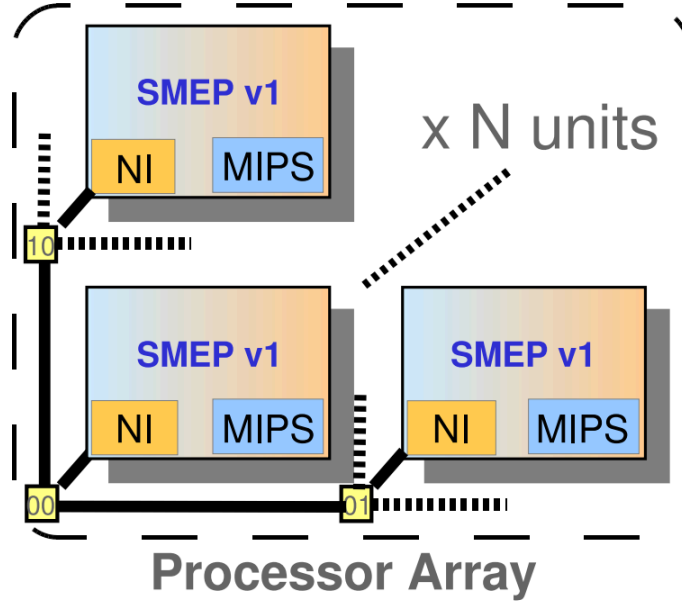


Figure 4.5: Fully homogeneous processor array : GENEPEY v1 [111]

4.4 Cost Analysis

To demonstrate the effectiveness of 3.5D integration, the manufacturing cost is analyzed for different technologies. I investigate the cost to implement a 288-core MPSoC for various 3D integration schemes.

For any vertical integration approach, the bonding process contributes significantly to the overall cost. Among the possible bonding approaches wafer-to-wafer, die-to-wafer, and die-to-die, I consider the die-to-wafer as a good compromise between manufacturing throughput and yield [112]. In order to estimate the overall cost of the MPSoC, I consider appropriate costs, reported in literature, for TSV die-to-wafer process [112] and 3DMI n/p integration [47]. In the case of 2D integration, 5 mask sets are needed for the active layer in which both the n-type and p-type transistors are patterned. Whereas in the case of 3DMI n/p integration, 6 mask sets (i.e. 3 mask sets for each active layer), as well as an additional *Silicon-On-Insulator* (SOI) layer, are needed. Taking into account the extra processing steps, the authors of [47] have reported 26%

increase in the total cost when compared to planar SOI (8 metal levels 22-nm process), with an assumption of producing 30000 wafers per month. It has to be noted that, with 3DMI n/p integration, I reduce the footprint of the active circuit, thereby decreasing the overall cost of the chip. The partition of the 288-core MPSoC with various integration schemes (shown in Fig 4.4) is reported in Table 4.1.

Integration	Cores/Die	Stacked dies	Die area (mm ²)
2-D	16 x 18	1	318
3-D TSV	6 x 8	6	59
3-D n/p	16 x 18	1	229
3.5-D	8 x 9	4	64

Table 4.1: Partition of a 288-core MPSoC with various Integration schemes.

The increase in the number of cores per die for 3.5D is supported by the decrease in the active footprint of the core, offered by 3DMI n/p integration. From Table 4.1, we can observe for similar die area for 3.5D and 3D TSV, I obtain 72 cores/die with 3.5D when compared to 48 cores in the case of 3D TSV. Hence by packing more cores onto a given die area, I reduce the number of stacked dies from 6 to 4. By considering the cost for 3DMI n/p integration [47] and the cost of TSV process for Die-to-Wafer stacking [112], I conjecture the manufacturing cost of the MPSoC to be reduced by 20% for 3.5D integration when compared to a corresponding 3D TSV implementation.

In addition to the manufacturing cost, testing cost plays a vital role in determining the overall cost of the 3D (vertically stacked) ICs. Figure 4.6 depicts test flows for 2D and 3D ICs [113]. A 2D test flow has two phases, wafer test (performed on the fabricated wafer) and final test (to detect packaging faults).

4.5 Simulation Framework and Results

In this section, I study the performance improvement of the blocks within the core of the MPSoC as well as the system-level performance improvement of the network-on-chip connecting various blocks of the MPSoC.

4.5.1 Performance Improvement of the Core

Since 3DMI n/p technology offers multiple active layers adjacent to each other, the layout of the planar standard cell can be folded in two layers thereby forming a 3D cell. For this case study, I employ *intra-cell stacking* techniques (discussed in 2.3.1). Fig. 4.7 shows the 2D and the 3D standard cells. As illustrated in Fig. 4.7b, p-type devices (forming the pull-up network) are realized at the top active

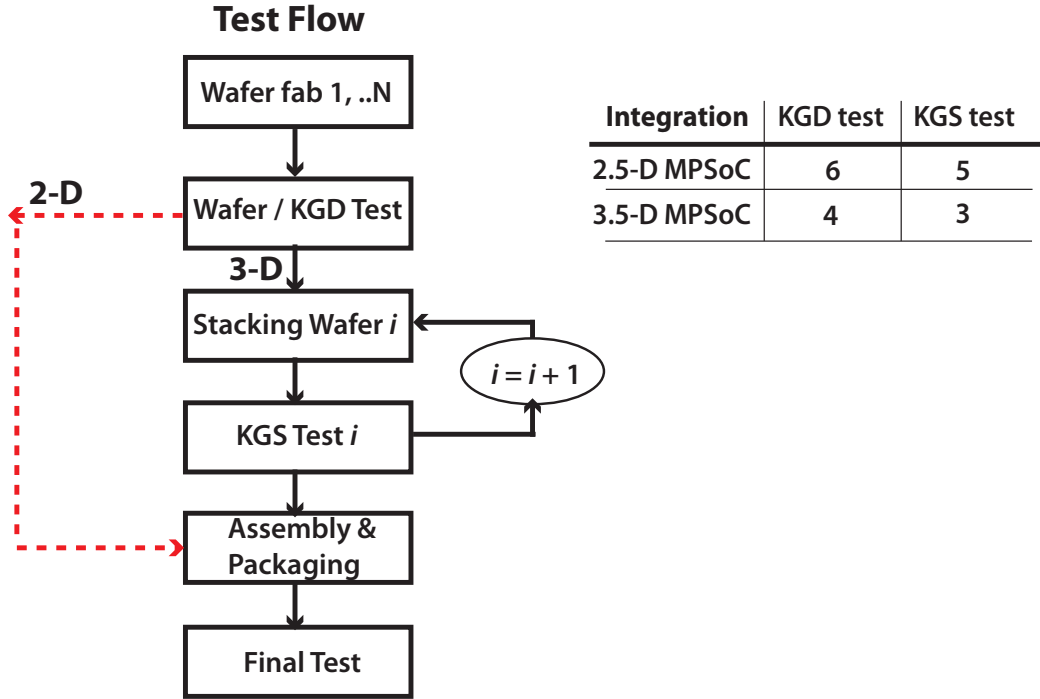


Figure 4.6: 2D and 3D Test flows [113].

layer and n-type devices (forming the pull-down network) at the bottom active layer. By assuming the same design rules of the backend of the line (metal lines), I mapped various existing 2D standard cell libraries to 3D cell libraries. One of the primary advantages of this cell transformation is the ease in integration with the conventional design flow, as the design effort consists of developing only the 3DMI n/p cell library [114].

The standard cells of the 45 nm Nangate Open Cell Library [13] are mapped to the corresponding 3D equivalent by changing the physical-attributes of the cells. For instance, the height of the cells is reduced by 30% without modifying any width. The size of the I/O pins is retained as in the case of a 2D cell while the location is altered. Since the drive strength of the gates is not altered, I assume similar delay characteristics as in the planar case. In this study I did not take into account the parasitic extraction of the standard cells. This assumption is valid at the current technology nodes, as the overall delay is dominated by interconnect and transistor delays.

Various benchmark circuits within the core of the MPSoC are considered [14]. Synopsys Design Compiler is used for mapping the RTL of the benchmarks onto target 3D standard cell library. Cadence Encounter is used as the physical synthesis engine to generate the virtual seed placement in wirelength driven mode. In Fig. 4.8, I show the improvement in wirelength, delay, and power of various benchmark circuits after placement is performed using the 2D and 3DMI n/p cell libraries. The power numbers include all components of the power dissipation namely leakage

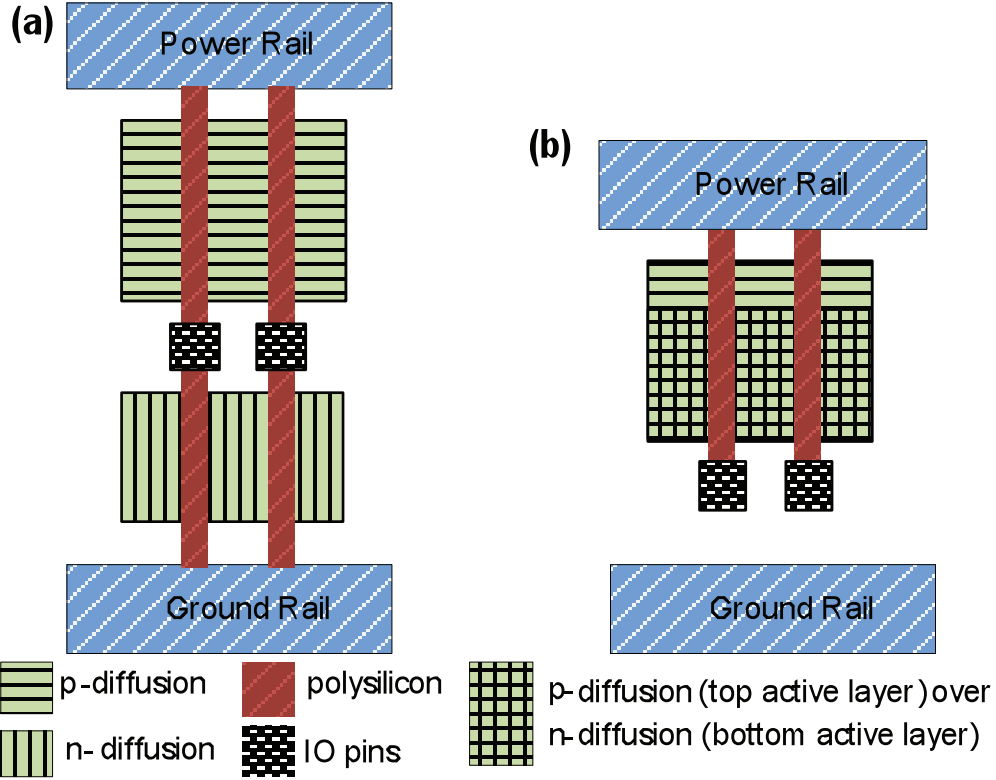


Figure 4.7: (a) Typical standard cell in 2D (planar) configuration and (b) Standard cell designed in 3DMI n/p by realizing the PUN on the top active layer and the PDN in the bottom active layer.

and switching power. By partitioning the circuit at the gate-level, with 3DMI n/p integration, the active-area footprint is reduced thereby leading to reduction in the average interconnect wirelength of the circuit. The performance improvement from 2D to 3DMI n/p integration is due to the reduction in the average interconnect wirelength. I observe 11.5% delay improvement and 3.4% power improvement when averaged across various benchmark circuits shown in Fig. 4.8.

4.5.2 Performance Improvement of the NoC

To assess the system-level performance of the different designs, from the communication point of view, I use a cycle accurate NoC [115]. The simulator assumes best effort NoC architecture similar to that described in the xpipes library [116]. I assume wormhole flow control with input buffered switches, that use round-robin arbitration and ON/OFF flow control [117]. Without loss of generality the arbitration and crossbar switching are done in one cycle. There are no output buffers in the switches, but pipeline stages are placed on long links in order to achieve the required operating frequency. I inject packets that are 10 flow control units (flits) long.

For our case study, I generate different mesh and 3D-mesh topologies that

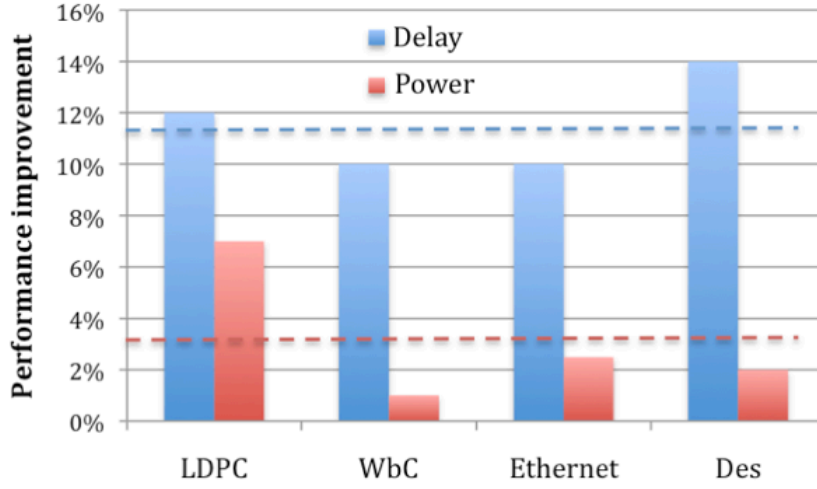


Figure 4.8: Performance improvement in terms of delay and power of various blocks of the core.

correspond to the different partitioning of the cores according to the various integration schemes as presented in Table 4.1. I simulate for different injection rates to assess the latency of the packets and the actual possible injection rates for the different NoC configurations. I inject uniform random traffic and for each configuration I perform simulations for 100000 cycles. First, I study the latency of the NoC at a constant injection rate of 0.1. When compared to planar implementation, with a 2D NoC (mesh size 16x18), the latency is reduced by 57% and 68% for the 3D NoC connecting the 288-cores in 3D TSV configuration (with mesh size 6x8x6) and 3.5D configuration (with mesh size 8x9x4) respectively. Given a 60% reduction in latency of a 3D NoC when compared to a 2D NoC, in Fig. 4.9a, I only show the latency for the two relevant cases of the 3D NoC. I observe 24% decrease in latency of the NoC for 3.5D configuration when compared to 3D TSV configuration.

Next, I study the maximum injection rate possible for various configurations. The injection rates are the values that actually affect the end-to-end NoC latency, hence the maximum injection rate gives the best performance of the NoC. In Fig. 4.9b, I show 44% improvement in injection rate from 3D TSV to 3.5D MPSoC implementation.

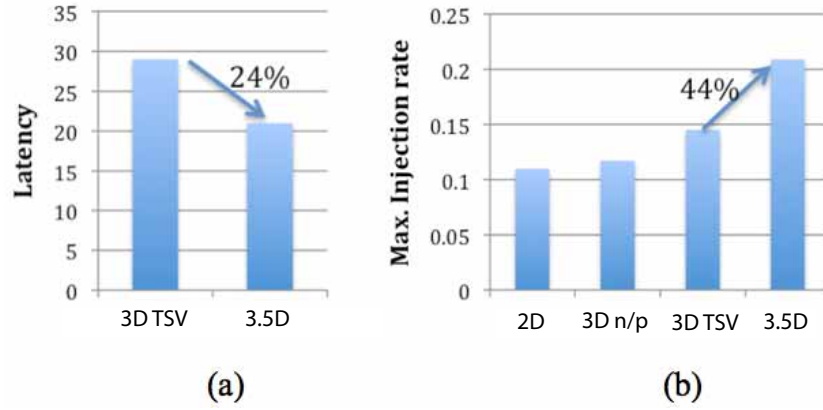


Figure 4.9: Performance improvement in terms of delay and power of various blocks of the core.

4.6 Chapter Contribution and Summary

In this chapter, a novel vertical integration scheme, called 3.5D integration, is proposed which synergizes existing 3D TSV and 3DMI technologies. The feature of gate-level stacking with 3DMI n/p integration is leveraged to stack more cores onto a die, when compared to a straightforward 3D TSV integration.

I consider a synthetic case study of a 288-core MPSoC to get insight into the advantages and disadvantages of the proposed integration scheme. Applying 3.5D integration to a 288-core MPSoC, the number of stacked dies is reduced by 30% when compared to a 3D TSV implementation. Based on existing cost models for various technologies, I conjecture that the overall manufacturing cost can be reduced by 20% and the test cost can be reduced by 30% for a 288-core MPSoC, with 3.5D technology when compared to a 3D TSV implementation.

Our simulation results show a performance improvement of 11.5% (on an average) for various benchmarks comprised in the core. In the case of interconnection network, I observe large improvement in the latency of the 3D NoC (average of 24%) for 3.5-D implementation when compared to 3D TSV implementation of the MPSoC.

Design Techniques for Nanowire FETs with Controllable Polarity

5

Following the trend to *one-dimensional* (1D) structures, *silicon nanowire field-effect transistors* (SiNWFETs) are a promising extension to the tri-gate FinFETs [118]. The superior performance of these 1D channel devices comes from a high I_{on}/I_{off} ratio, due to the gate-all-around structure, which improves the electrostatic control of the channel, thereby reducing the leakage current of the device. The advantage of SiNWFETs over other 1D devices such as carbon nanotube transistors is that SiNWs can be fabricated with a top-down silicon process [119]. Moreover, SiNWs can be built in vertical stacks, thereby giving very dense array of nanowire transistors [120]. Figure 5.1(a, b) illustrates a possible extension of a FinFET to SiNWFET device structure with SiNWs suspended between source and drain pillars. In addition, SiNWFETs exhibit enhanced electrostatics properties, such as polarity control, which are electrically hard to achieve in planar- and Fin- FETs.

The design methodology presented in this work takes advantage of the electrostatics of these devices, which can be fabricated to be ambipolar, i.e. to exhibit both n- and p-type characteristics. By engineering the source and drain contacts and by constructing independent double-gate structures, the device polarity can be electrostatically forced to either n- or p- type by polarizing one of the two gates. Figure 5.1c illustrates a *double-gate* (DG) SiNWFET device structure with *control gate* (CG) and *polarity gate* (PG). The in-field polarizability of these devices enables the development of new logic architectures, which are intrinsically not implementable in CMOS in a compact form [121]. However, the routing complexity at the device level increases due to the presence of an extra gate, the polarity gate [122].

Typical CMOS layout techniques involve transistors with a single gate. In the traditional approach for CMOS, compact layouts are realized by optimal transistor chaining of p- and n-type transistors [123, 124, 125]. However, in the case of ambipolar gates, the polarity of the transistor (p-type or n-type) changes with the input signals. Motivated by these observations, I propose compact layout techniques

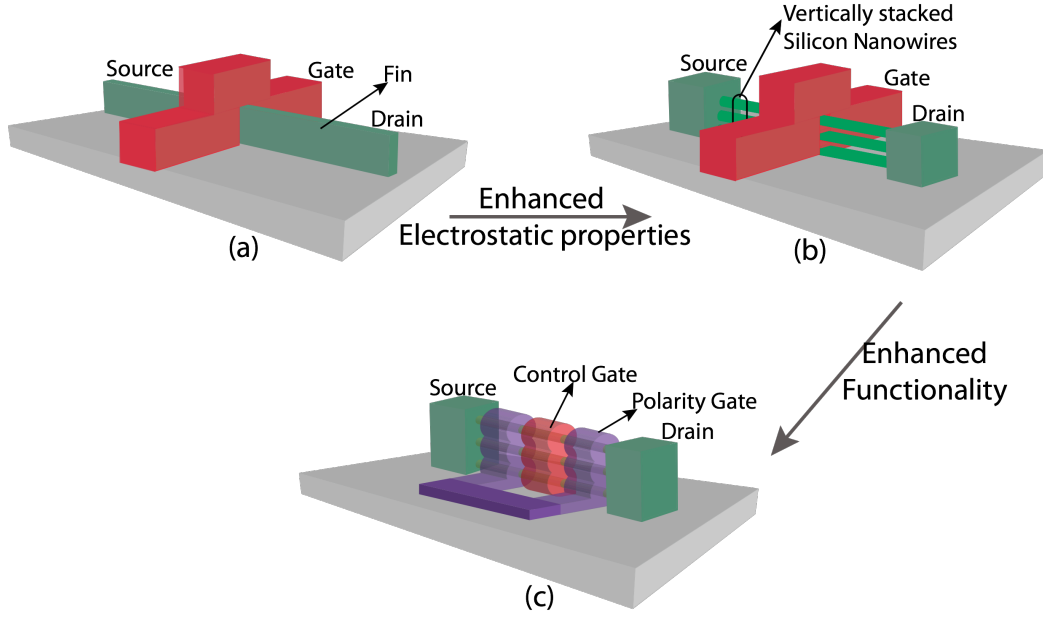


Figure 5.1: (a) FinFET providing increase in controllable channel area between the source and drain regions (b) Vertically-stacked SiNWFET with multiple parallel nanowire channels, each with Gate-All-Around control (c) Double-Gate SiNWFET with control and polarity gates.

for *Double Gate Silicon Nanowire FET* (DG-SiNWFET). In order to facilitate this, I propose novel symbolic layouts for ambipolar logic with Dumbell-Stick diagrams.

In this chapter, I address layout techniques to mitigate gate-level routing congestion caused by two independent gates of a DG-SiNWFET. The main contributions of this chapter are:

- A compact layout technique for complex gates with embedded XOR/XNOR functions. In order to facilitate this, a novel symbolic layout (called Dumbell-Stick diagrams) is proposed for ambipolar logic gates.
- With the help of TCAD simulation of the SiNWFET, gate-level simulations were performed to study the benefits of DG-SiNWFET when compared to CMOS at 22nm node.

The remainder of this chapter is organized as follows. First, a background on DG-SiNWFET technology and various design approaches based on ambipolar devices is presented. Next, I discuss how to realize various Boolean functions with ambipolar logic style (i.e. by employing double-gate transistors with controllable polarity). Then, I introduce novel layout techniques for ambipolar circuits by mitigating gate-level routing overhead caused by an extra gate for every transistor. With the help of TCAD model of the basic device, I compare DG-SiNWFET with CMOS at 22nm node. Finally, the chapter is concluded by discussing the results and by summarizing the contributions of this part of the thesis.

5.1 Transistors with Controllable Polarity: Ambipolar Transistors

This section surveys previous works related to ambipolar technologies with a main focus on DG-SiNWFET. It also summarizes various new design techniques, which leverage ambipolarity at the circuit level.

5.1.1 Double-Gate SiNWFET Technology

Various new technologies show an inherent behavior towards controllable polarity, including silicon nanowires FETs [126], carbon nanotube FET [127], and graphene nanoribbons [128]. In this section, I focus on SiNWFETs to illustrate the layout technique for ambipolar logic circuits. The advantage of SiNWFETs over other one-dimensional devices such as carbon nanotube transistors is that SiNWs can be fabricated with a conventional silicon process as an extension to traditional CMOS technology [119]. Moreover, SiNWs can be built in vertical stacks, thereby giving dense arrays of nanowire transistors [120].

Figure 5.2 shows a DG-SiNWFET device structure with SiNWs suspended between source and drain pillars. This SiNW is divided into three sections, which are in turn polarized by two gate-all-around gate regions. The center gate region works as in a conventional MOSFET, switching conduction in the device channel by means of a potential barrier. The side regions are instead polarized by a polarity gate, which controls Schottky barrier thickness at the S/D junctions, thus forcing the device to be either n- or p-type.

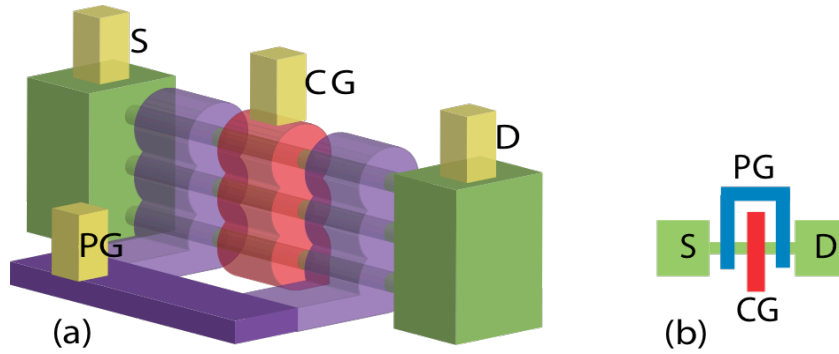


Figure 5.2: Conceptual structure of the ambipolar DG- SiNWFET: a) 3D view of the device. b) Top view of the device showing one stack of nanowires forming the channel.

A SEM image of an array of vertically stacked SiNWs, suspended between pillars, before patterning the gates, is shown in Fig. 7.3a. Figure 7.3b shows the double-gate SiNWFET after patterning the control and polarity gates [122].

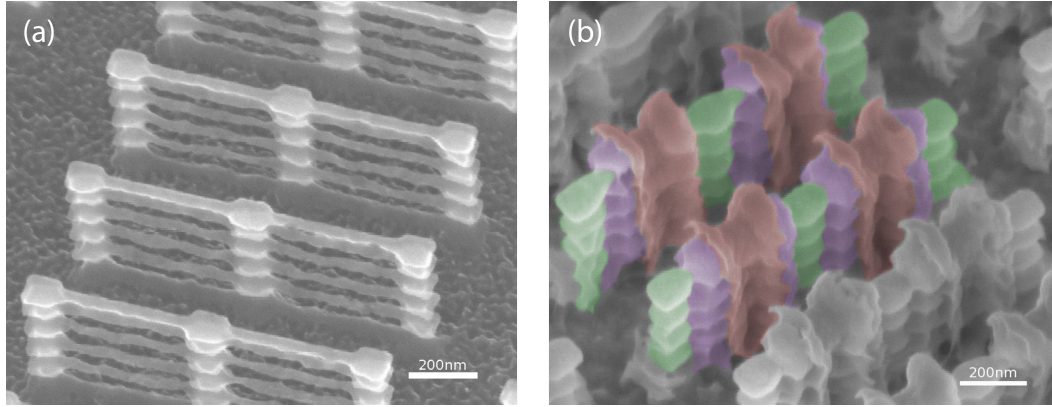


Figure 5.3: Conceptual structure of the ambipolar DG- SiNWFET: a) 3D view of the device. b) Top view of the device showing one stack of nanowires forming the channel.

5.1.2 Device Operation

A fabrication technique to manufacture programmable DG-SiNWFETs has been proposed in [122]. Figure 5.4(a, b) illustrates the top view of the DG-SiNWFET and its corresponding symbol. As the name suggests, the device has two gates CG and PG. The *control gate* (CG) is similar to the regular gate of a MOSFET, which turns the device on or off. On the other hand, *polarity gate* (PG) sets the majority carriers of the device channel to either p-type or n-type. As depicted in Fig. 5.4c, if the PG is set to high (logic 1), the device behaves as a n-type transistor, and by setting the PG to low (logic 0), we obtain a p-type transistor.

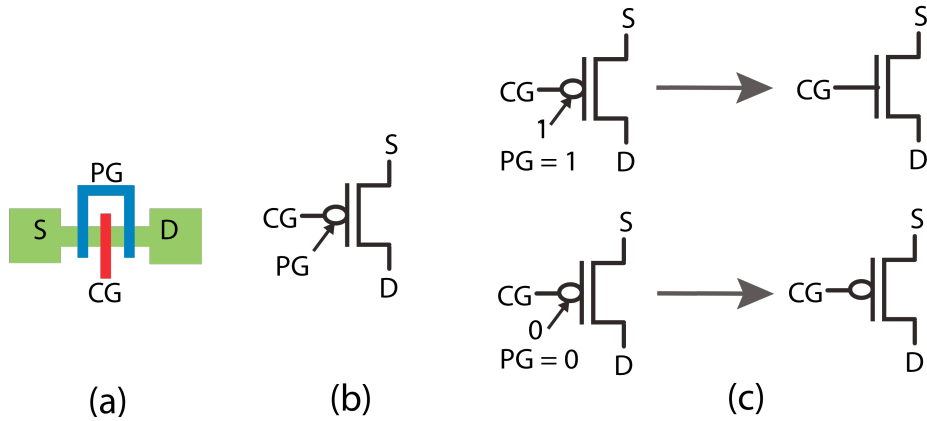


Figure 5.4: Double-gate SiNWFET (a) Layout (top view). (b) Symbol of an ambipolar FET (c) Configuration as n-type and p-type by setting the PG.

Though I focus on double-gate SiNWFETs, the proposed design methodology holds relevant for other ambipolar FETs (CNFETs [127], and GNR-FETs [128]) with two independent gates for in-field programmability.

5.1.3 Design Techniques for Ambipolar Circuits

New design methodologies are proposed for exploiting the controllable polarity, unique to double-gate devices, which leads to a very compact realization of XOR function [129, 130, 131]. In [129], a reconfigurable logic gate that maps eight different 2-input logic functions in dynamic logic was presented. In [130], a library of static ambipolar gates based on generalized NOR-NAND-AOIs is proposed which efficiently implements XOR-based functions. Various novel reconfigurable blocks with embedded XOR blocks have been proposed which leverage upon embedded XOR functionality [132]. In [131], authors propose universal logic modules that leverage ambipolar transistors. In this work, I abstract the physical design issues that are common to all the new design methodologies comprising of double-gate ambipolar transistors. I also propose a procedure for constructing the symbolic layout of ambipolar logic circuits.

5.2 Ambipolar Logic Circuits

In this section, I discuss various logic implementations with double-gate transistors with controllable polarity. First, I introduce the basic terminology for ambipolar transistors along with the simple classification of Boolean functions. Then, ambipolar logic style for various Boolean functions is illustrated with relevant examples.

5.2.1 Terminology

A controllable polarity transistor, αt , is denoted as a quadruple (D, CG, PG, S) , where D, CG, PG and S represent the drain, the control-gate, the polarity-gate and the source signals that αt connects to, respectively. The voltage signal applied to the PG determines the type (p-type or n-type) of transistor. A transistor, αt , operates as a p-type device (αt_p) by connecting the PG to 0, and an n-type device (αt_n) by connecting the PG to 1.

5.2.2 Unate, Binate, and Mixed Boolean Functions

A function $f(x_1, x_2, \dots, x_i, \dots, x_n)$ is positive unate in x_i if, $\forall x_j, j \neq i$:

$$f(x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) \geq f(x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n)$$

Similarly, the function is negative unate in x_i if, $\forall x_j, j \neq i$:

$$f(x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \geq f(x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$$

A function f is binate in variable x_i if it is neither positive nor negative unate on variable x_i . A function is (positive/negative) *unate* if it is either positive or negative unate for all x_i , where $i \in [1, n]$. Similarly, a function is *binate* if it is binate for all the variables. A function is *mixed*, if it contains both binate and unate variables. Table 5.1 gives few examples of unate, binate, and mixed functions.

Boolean function	Type
INV, NAND2...	unate (negative)
AND2, OR3...	unate (positive)
XOR (a, b)	binate
$sa + \bar{s} \bar{b}$	mixed (binate + postive unate + negative unate)
$sa + \bar{s} b$	mixed (binate + positive unate)
$(sa + \bar{s} b)'$	mixed (binate + negative unate)
$\overline{a(b \text{ XOR } c)}$	mixed (XNUMixed)

Table 5.1. Example of *unate*, *binate*, and *mixed* functions.

There are various flavors of mixed functions, according to how binate and positive/negative unate variables are combined. I consider also here a subclass of mixed functions that is common in design libraries. I call XNUMixed those functions that are conjunctions/disjunction of XOR/XNOR with negative unate functions. De Morgan's law [133] can be used to map into this class those functions that combine positive unate functions with XOR/XNOR.

5.2.3 Ambipolar Logic Gates

In this section, I describe circuit level implementation of negative unate, positive unate, binate and XNUMixed logic functions realized with controllable-polarity transistors.

Negative unate functions

Negative unate functions are obtained by biasing the polarity gates of the *pull-up-network* (PUN) to *Gnd* and *pull-down-network* (PDN) to *Vdd*. This is similar to complementary CMOS style where the PUN and PDN are comprised of p-FETs and n-FETs respectively. Fig. 5.5a illustrates a 2-input NAND gate. Since the ambipolar transistors are configurable, just by swapping the *Vdd* and *Gnd* terminals, along with the connection to the PGs, of the NAND schematic (Fig. 5.5a), I generate a NOR function as shown in Fig. 5.5b. This technique applies to all the negative unate function.

Positive unate functions

In the case of positive unate functions, various design approaches are considered. Fig. 5.6a shows an implementation of a 2-input OR gate from the same schematic of a NAND gate. By interchanging the voltage applied to the PGs in the PUN and PDN, we obtain n-type and p-type transistors in the PUN and PDN respectively. Though this gives a straightforward implementation of positive unate logic, we have

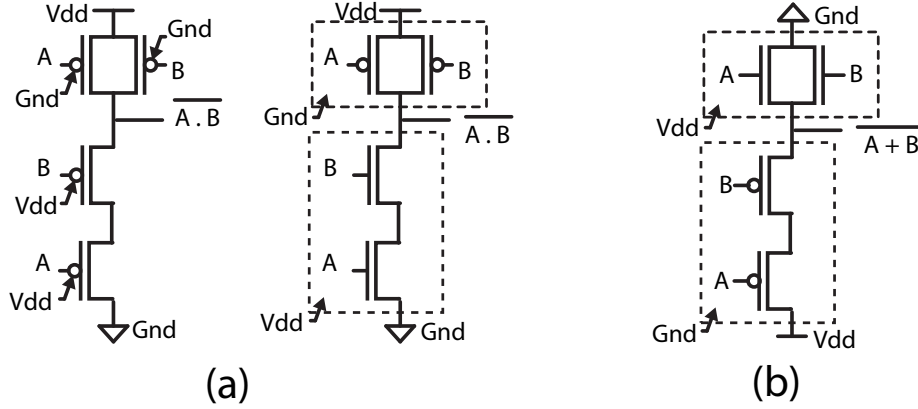


Figure 5.5: Negative unate logic function (a) NAND gate (b) NOR gate implementation by swapping the Vdd and Gnd of a NAND gate (a).

to consider the degraded output signal (*e.g.* $(A+B)_d$). By adding a buffer at the output we can realize full swing at the output. On the other hand, a positive unate function can be obtained by inverting the output of an equivalent negative unate function. An example of a 2-input AND gate is shown in Fig. 5.6b. In addition, a positive unate function can be obtained by applying De Morgan's law to the function as shown in the Fig. 5.6c. Since we prefer not to use a configuration that degrades output signals and requires buffer (*e.g.* Fig. 5.6a), a rule of thumb to implement positive unate functions is by biasing the PGs of the PUN and PDN to Gnd and Vdd respectively (see Fig. 5.6(b, c)). Between the two configurations of Fig. 5.6(b, c), we can observe that the implementation in Fig. 5.6b is better as it requires fewer numbers of inverters (*e.g.*, input inversion has to be accounted for).

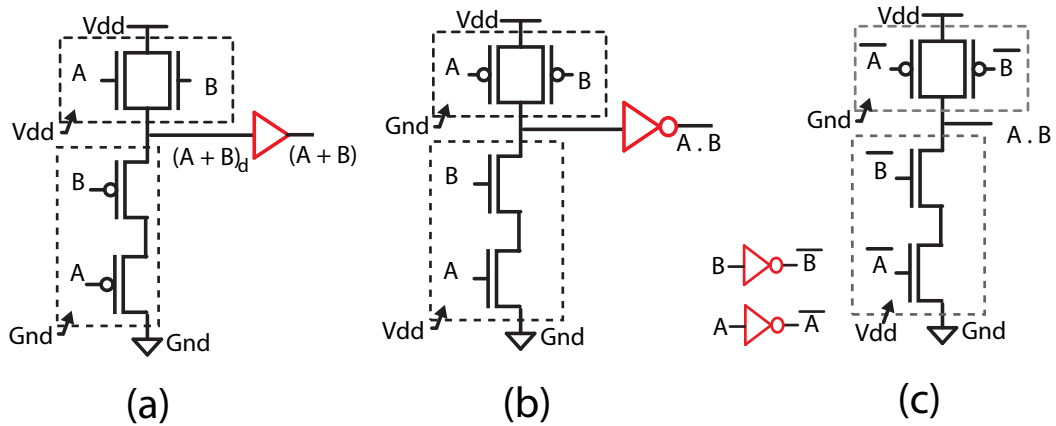


Figure 5.6: Positive unate logic function (a) OR gate implementation by interchanging the voltage of the PGs in the PUN and PDN. (b) AND (positive unate) gate implemented with NAND (negative unate) gate followed by Inverter. (c) AND gate implemented by applying De Morgans rule.

Binate functions

Double-gate transistors are efficient in implementing binate functions. An example of a 2-input XOR gate with only two ambipolar transistors is shown in Fig. 5.7a. When compared to unate logic style, we notice that the PGs are connected to the input logic signals (*e.g.* logic signal B in the Fig. 5.7). From the truth table shown in Fig. 5.7a, I observe that output is degraded when the ambipolar transistor is configured to be p-type in the PDN and n-type in the PUN. The degraded output signal can be recovered by placing a buffer at the output. In order to obtain full swing at the output, an alternative approach using transmission-gates (*e.g.*, two parallel transistors) is proposed [130], where a 2-input XOR gate can be constructed using only 4 ambipolar transistors. An example of a 2-input XOR gate is shown in Fig. 5.7b, where all the polarity gates are either connected to logic input B or $\bar{B}$. For any given configuration, the output is either pulled-up (or pulled-down) by both n-type and p-type transistors. Fig. 5.7c demonstrates the case where B is assigned to logic 1. The transmission gates with complemented inputs in the PUN and PDN assure a full swing at the output. When compared to static CMOS implementation of an XOR2 (which needs 12 transistors), the transmission-gate XOR2 with ambipolar transistors needs only 8 transistors (transistors shown in Fig. 5.7b along with the two inverters for generating $\bar{A}$ and $\bar{B}$).

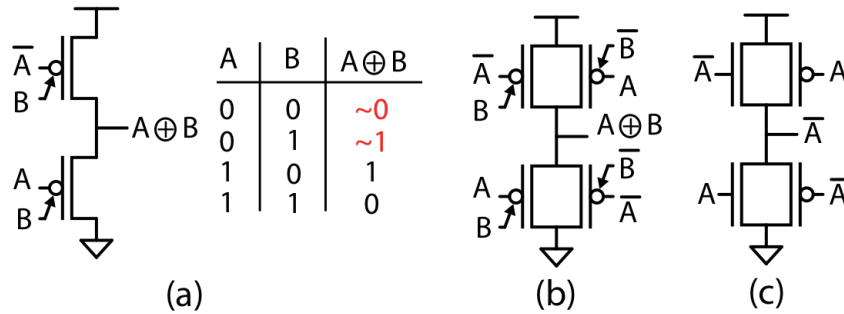


Figure 5.7: 2-input logic gates (a) NAND gate with PGs connected to V_{dd} and G_{nd} (b) XOR gate with PGs connected to input signals (B or $\bar{B}$) (c) XOR gate with B assigned to logic 1.

Mixed functions

Within the mixed function class, XNUMixed functions can be effectively laid out. As an example, In Fig. 5.8, I show the implementation of function $Y = \overline{(A \oplus B)C}$ in both static-CMOS logic style (Fig. 5.8b) and ambipolar logic style (Fig. 5.8a). From the figure, we can observe that the number of transistors is reduced by half with ambipolar logic style when compared to CMOS implementation. I incorporated transistor pairs only for the XOR combination of the logic, where the PGs are biased to input logic signals (B and $\bar{B}$). For the variables, which are negative unate (*e.g.* logic function Y is negative unate in C), the PGs are connected to the V_{dd} and G_{nd} as shown in the Fig. 5.8a.

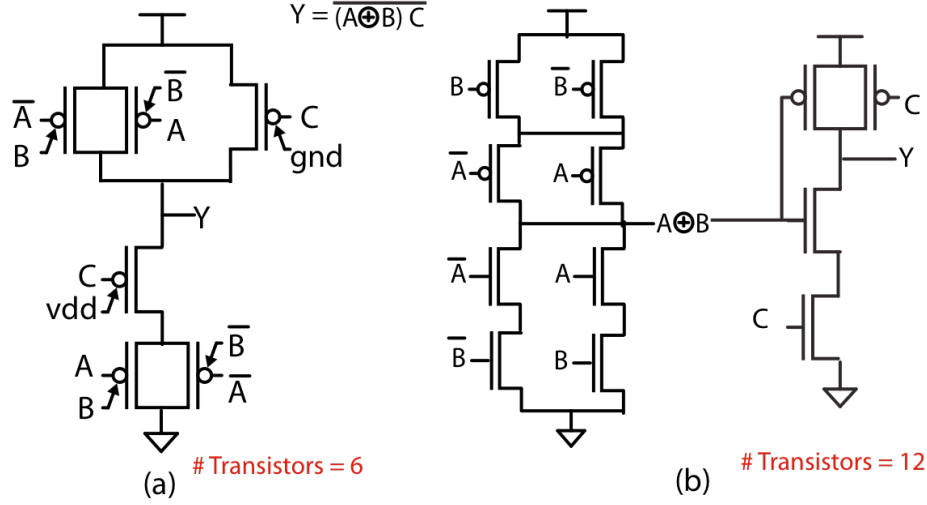


Figure 5.8: Mixed function $Y = \overline{(A \oplus B)C}$ (a) Ambipolar logic style, where the PGs of the binate logic are connected to logic inputs (B or $\bar{B}$) and PGs of unate variables are connected to V_{dd} and Gnd (b) Static CMOS implementation.

5.3 Layout Techniques for Ambipolar Logic Gates

One of the caveats of ambipolar design style is the increase in the intra-cell routing complexity. Since every transistor has two gates to connect to the logic signals, care should be taken to mitigate the gate-level routing complexity. In this section, I first propose novel symbolic-layouts for controllable-polarity logic gates, called *dumbell-stick diagram* (DSD). With the help of DSDs, I propose a novel layout technique for ambipolar design style. In addition to transistor pairing I also leverage on transistor grouping, thereby obtaining layouts which are compact, regular and easy to route. I start with simple examples of 2-input logic gates and then propose a generic procedure for arbitrary complex XNUnmixed gates.

5.3.1 Dumbell-Stick Diagrams

Similar to the CMOS stick diagrams [134], dumbell-stick diagrams denote ambipolar devices (in our case DG-SiNWFET) with a simplified layout abstraction in order to study the cell-routing complexity. Fig. 5.9a shows the top view of a simple DG-SiNWFET (see Fig. 5.2). Similar to FinFETs, a large transistor is obtained by increasing the number of nanowire-stacks (Fins in the case of FinFET) in parallel as shown in the Fig. 5.9b. In Fig. 5.9c, I show the dumbell-stick representation of the transistor, with suspended silicon nanowires between the source and drain contacts forming the basic dumbell, and the control gate and the polarity gate constituting the sticks. It has to be noted that DSDs do not take into account the size of the transistor, but just the topology of the interconnect.

Transistor pairing, shown in Fig. 5.9d, is an important transistor placement

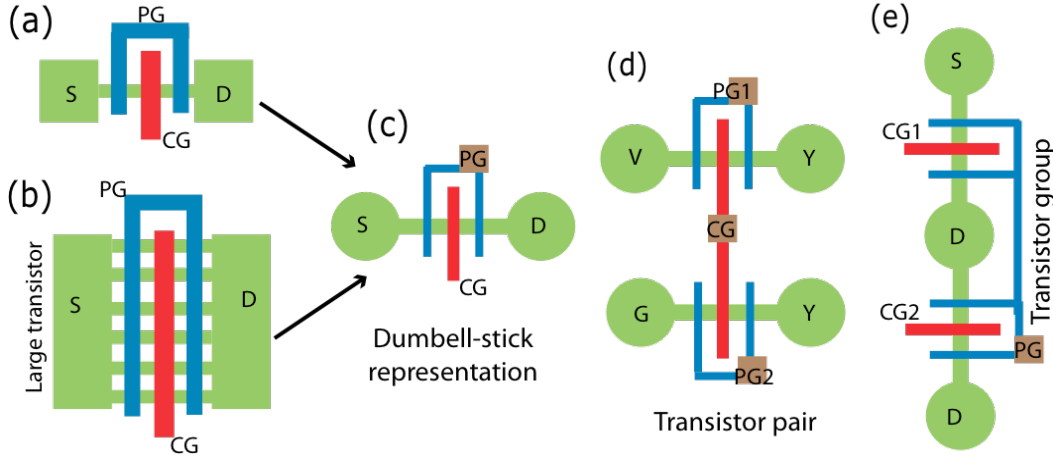


Figure 5.9: (a) A top view of the DG-SiNWFET shown in Fig. 5.2. (b) Large transistor. (c) Equivalent dumbbell-stick diagram. (d) Dumbbell-stick diagram of an Inverter with a transistor pair. (e) Grouping transistor with similar polarity gates.

technique used for layout area reduction. By transistor-pairing, control gates of two transistors (connected to the same signal) are vertically aligned by a single stick segment to minimize the routing complexity as well as to ensure more layout regularity. Two transistors, $\alpha t1$ and $\alpha t2$, are paired together if their control gates are connected to the same signal, i.e. $CG(\alpha t1) = CG(\alpha t2)$.

In Fig. 5.9e, I show transistor grouping. Two transistors, $\alpha t1$ and $\alpha t2$, belong to the same group if their polarity gates are connected to the same voltage, i.e. $PG(\alpha t1) = PG(\alpha t2)$. Hence by transistor-grouping, transistor's with similar PGs are grouped together. Transistor grouping is unique to ambipolar double-gate devices. In the following section I show the importance of grouping transistors for minimizing the routing overhead introduced by polarity gates.

5.3.2 Layout Techniques for 2-input Unate Functions

From Section 5.2.3, I have seen that negative unate logic gates (*e.g.* NAND, NOR, INV,...) can be obtained by biasing the polarity gates of the PUN to Gnd and PDN to Vdd . Hence, all the transistors in the PUN (and PDN) can be grouped together (i.e. PGs of the stacked transistors are connected together), thereby forming one PG for each PUN and PDN. With fixed biasing for the PGs, CMOS layout techniques with optimal transistor chaining [125, 123] can be employed in order to obtain area-efficient layout. The transistors are placed in two parallel rows where all transistors in the PUN are in one row while all the transistors in the PDN are in the other. The main objective is to place transistors in such a way that the gate signals are aligned and drain/source regions of adjacent transistors are abutted. Fig. 5.10 shows an example of a 2-input NAND gate with an Euler path approach [124]. From the Euler path, the optimal transistor alignment chain is obtained.

For positive unate logic gates (*e.g.* AND, OR, BUF,...), one of the techniques presented in Fig. 5.6(a, b, c) can be employed. In all the three cases, we can observe that the transistors in the PUN and PDN can be grouped together and tied to either V_{dd} or G_{nd} . The layout technique for unate logic gates is similar to CMOS style.

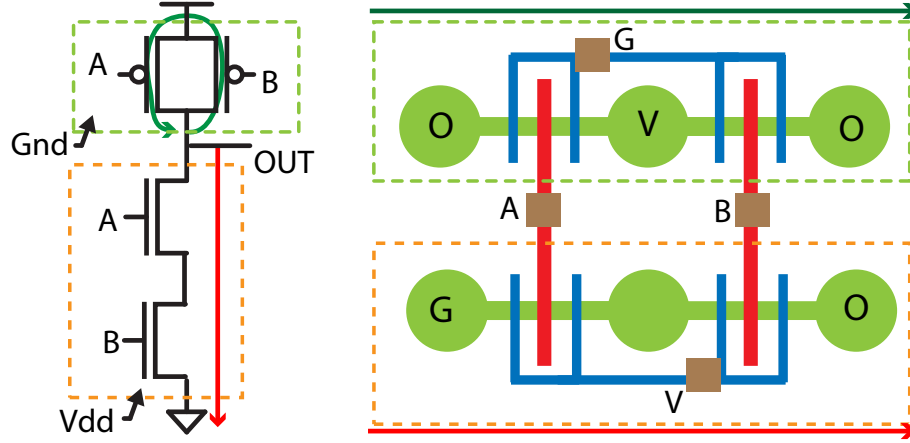


Figure 5.10: Dumbbell-stick diagram for 2-input NAND gate with the PGs grouped together in the PUN (and PDN) and connected to GND (and VDD).

5.3.3 Layout Techniques for 2-input Binate Functions

The main application of ambipolar devices is in implementing binate logic functions. From Section 5.2.3, we have seen that a 2-input XOR gate can be constructed using only 4 transistors. Fig. 5.11 shows an example of a 2-input XOR gate along with the two possible dumbbell-stick representations. In case-1, i illustrate a layout technique, similar to conventional CMOS style, where all the transistors in the PUN (and PDN) are placed together. It has to be noted that the polarity gates in the PUN (and PDN) cannot be grouped, unlike in the case of unate logic gates. Since the adjacent transistors cannot be grouped, extra routing effort is needed to connect polarity gates together (case-1 of Fig. 5.11). An efficient implementation is shown in the case-2 of Fig. 5.11, where polarity gates are grouped together irrespective of the transistor being a part of PUN or PDN. The circuit is no more seen as PUN and PDN, but partitioned based on the signals assigned to the PGs. From the dumbbell-stick diagram, i can observe that the PUN and PDN are placed next to each other. Unlike CMOS, DG-SiNWFET technology does not impose any process challenges (which lead to design rules) when placing p-type next to n-type transistors.

5.3.4 Layout Techniques for XNUnmixed Functions

Several novel circuit designs and architectures have been proposed which leverage upon ambipolar logic with embedded XOR functionality [121, 132, 131]. In [132],

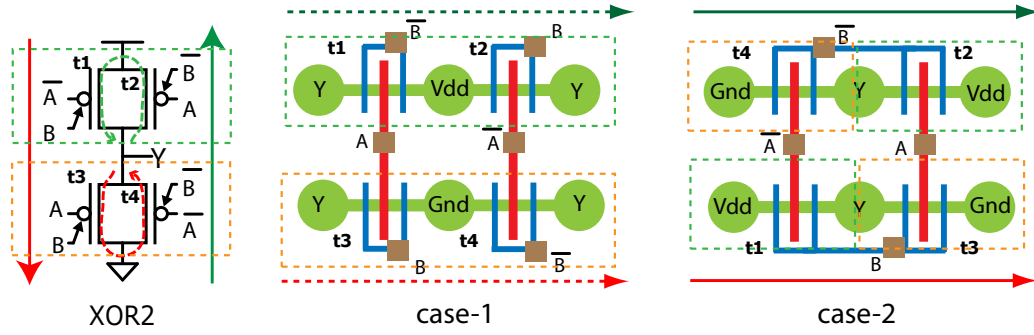


Figure 5.11: Dumbbell-stick diagram for 2-input XOR gate – (case-1) conventional approach by placing the transistors in the pull-up (pull-down) together so that they share the diffusion contacts (case-2) efficient layout technique where the transistors are grouped together, irrespective if they are located in the pull-up or pull-down networks, as well as share the same diffusion contacts.

authors have presented the idea of regular logic fabrics and evaluated various complex gates (combination of AND-XOR-OR-INV) based on the number of sub-functions each gate can implement. A key observation is that 2-input XOR/XNOR gates form the main building block of most logic cells, especially used in data-path design and within arithmetic building blocks. Recall that XNUMixed functions are conjunctions/disjunction of XOR/XNOR with negative unate functions.

From the example of XOR2 (Fig. 5.11), I observe that efficient layouts are obtained by placing the transistors together with similar PGs. In order to facilitate grouping, a specific transistor ordering is needed for XNUMixed logic functions Fig. 5.12 illustrates two different transistor arrangements for the function $\overline{(A \oplus B)C}$. In Fig. 5.12a, the binate logic part $(A \oplus B)$ is realized close to the *Gnd* and *Vdd* terminals, whereas, in Fig. 5.12b it is realized close to the output (*Y*) terminal. From the DSDs of the two cases, I infer that the circuit implementation in Fig. 5.12b is more efficient when compared to the one in Fig. 5.12a, as it reduces the routing needed by the polarity gates. As a rule of thumb, in the case of XNUMixed logic gates, placing the binate logic close to the output node leads to efficient layout.

5.3.5 Procedure for Generating Layout of XNUMixed Functions

In this section, I present a generic procedure to generate layout for XNUMixed functions. Our objective is to achieve a regular layout with:

- Transistor pairs aligned to give the least number of breaks in the active regions, which lead to realizing compact layouts (like in CMOS and NMOS logic).
- The least number of transistor groups, in order to reduce intra-cell routing complexity.

It has to be noted that if we refer to the first goal only, procedures for CMOS layout [124, 123, 125] are widely applicable. In particular the algorithm by Hwang

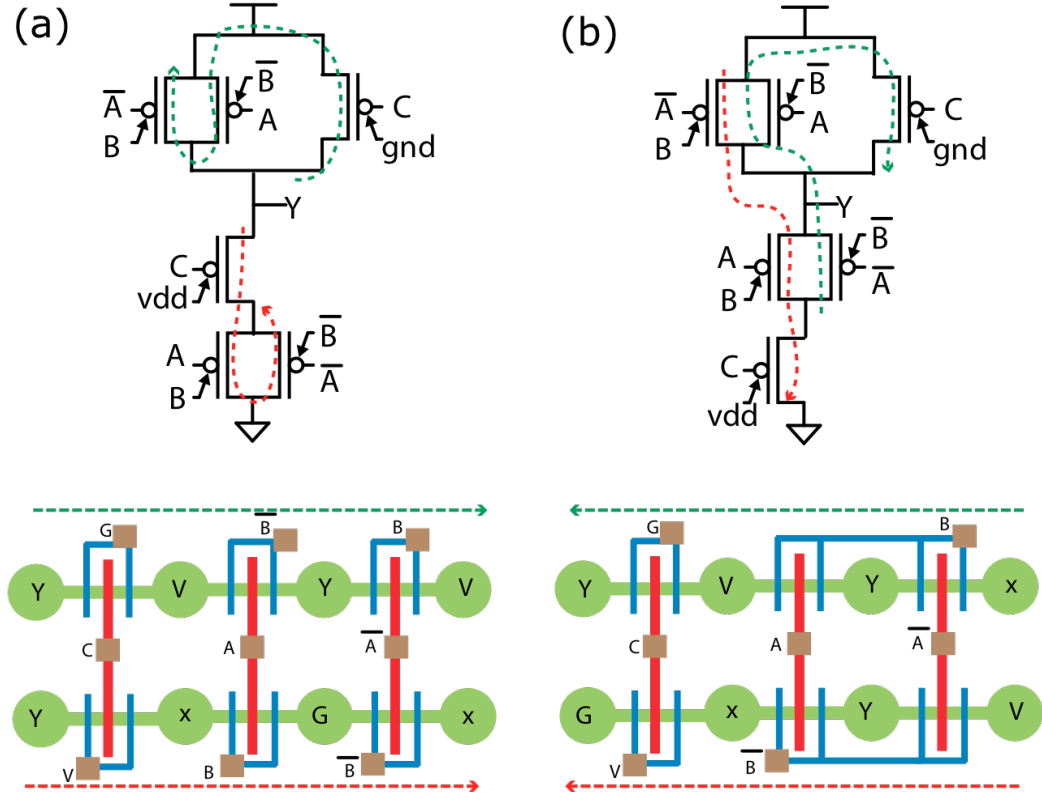


Figure 5.12: Transistor ordering for XNUMixed logic function.

[123] gives near-optimum solution with short computing time. In our case, it is important to address both aforementioned goals, and thus I adapt Hwang's algorithm, which I summarize below and exemplify in action.

Our procedure, to meet both objectives, consists of six steps: re-ordering, grouping, pairing, generating unate- and binate-bipartite graphs, chaining, and DSD construction. Input to the procedure is a XNUMixed circuit schematic with a complementary logic style (i.e., equal number of transistors in the pull-up and pull-down network with dual topology graphs).

The first step is transistor re-ordering, with the binate inputs placed close to the output node as explained in Fig. 5.12. By means of transistor grouping, various sub-graphs are formed by clustering the transistors sharing similar PGs. I model the circuit schematic as a list of graphs $G = \{G_{PG-1}, G_{PG-2}, \dots, G_{PG-i}, \dots\}$, where $G_{PG-i} = (V, E)$, in which V represent the nodes (source/drain contacts of the transistors) and E represent the edges (CG of αt) of all the transistors whose PG is connected to i . Applying transistor grouping to circuit schematic shown in Fig. 5.13a, I obtain $G = \{G_{PG-v}, G_{PG-g}, G_{PG-B}, G_{PG-\bar{B}}\}$ for the four transistor groups with PGs connected to Vdd , Gnd , B and $\bar{B}$ respectively. As an example I list the graph related to the transistors whose PGs are connected to B , $G_{PG-B} = [A, \bar{A}], [(a3, a4), (b2, b1)]$.

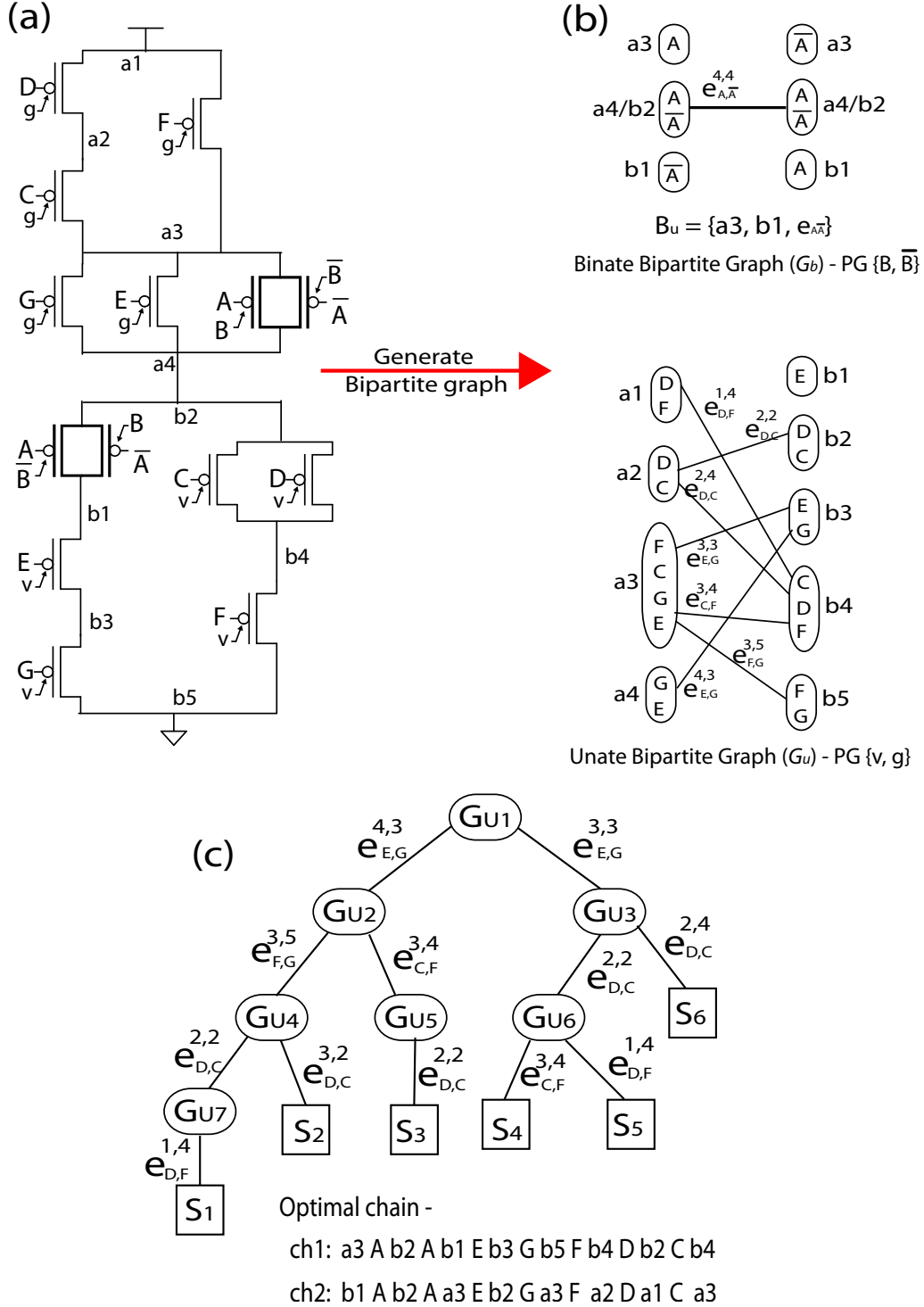


Figure 5.13: Logic-to-layout procedure (a) Complex logic function. (b) Separate bipartite graph representation for binate and unate part of the logic. (c) Search tree of unate-bipartite graph in (b).

Transistor pairing is performed next. In this step transistors with similar CGs are paired together. For complementary logic style, each pair consists of a transistor in the PUN and PDN. This step ensures the control gates are well aligned with minimum routing resources.

I differentiate from Hwang's approach by generating separate bipartite graphs for the unate and binate parts of the function. The unate and binate logic part of the circuits can be determined from the transistor-grouping step. I represent the possible abutments between the dual graphs as a bipartite graph G_x . An unate-bipartite graph (G_u) corresponds to the dual sub-graphs G_{PG-v} and G_{PG-g} , whereas, a binate-bipartite graphs (G_b) corresponds to the dual sub-graphs G_{PG-B} , and $G_{PG-\bar{B}}$. In the bipartite graph, nodes with only one transistor constitute to the list of *essential abutments* (e.g. nodes $a3$ and $b1$ of the graphs G_u and G_b). The main objective of this step is to find a unique transistor chain for the PUN and PDN with minimum number of breaks in the adjacent PGs and the diffusion area.

A pseudo-code description of the proposed procedure is shown below:

Algorithm: ambipolar_logic_to_layout()

Step 1: re-ordering();
Step 2: grouping();
Step 3: pairing();
Step 4: $G_b = \text{binate_bipartite_graph}()$;
Step 5: $G_u = \text{unate_bipartite_graph}()$;
Step 6: $B_b = \text{NULL}$;
Step 7: $E_{abu} = \text{essential_binate_abutments}(G_b, B_b)$;
Step 8: $B_u = E_{abu}$;
Step 9: $\text{opt_chain} = \text{Chaining}(G_u, B_u)$;
Step 10: $\text{dumbell_stick_diagram}(\text{opt_chain})$;

In the algorithm, B_b and B_u represents the set of essential abutments (i.e. nodes with only one transistor) of the bipartite graphs G_b and G_u respectively. Since the XOR2 part of the logic constitutes mainly for G_b , finding the essential abutments (B_b) is simple as shown in the Fig. 5.13b. Once I have the set of essential abutments from the binate logic part of the circuits, I continue to find the essential edges from the remaining part of the circuit. Optimal transistor chaining is obtained by a depth-first search on G_u while B_u is set to B_b . Fig. 5.13c shows how the procedure works on the example circuit. The search process starts from the root, where $G_{u1} = G_u$ and $B1 = a3, b1, eA\bar{A}$. From G_u (see Fig. 5.13b), I see $b1$ as an essential edge, hence I form an edge set, which consists of $eEG43$ and its mutually exclusive member, $eEG33$. Traversing to the left branch of the search tree node, G_{u1} , I add $eEG43$ to B_{u1} to form a new set B_{u2} . Similarly G_{u2} is derived from G_{u1} by removing edge set corresponding to $eEG43$. The matrix representations of G_{ui} and B_{ui} at various nodes of the search tree are shown in the

G_{U1} :

	b2	b3	b4	b5
a1	-	-	e_{DF}^{14}	-
a2	e_{DC}^{22}	-	e_{DC}^{24}	-
a3	-	e_{EG}^{33}	e_{CF}^{34}	e_{FG}^{35}
a4	-	e_{EG}^{43}	-	-

$$B_1 = \{e_{AA}, a3, b1\}$$

 G_{U2} :

	b1	b2	b3	b4
a1	-	-	e_{DF}^{14}	-
a2	e_{DC}^{22}	-	e_{DC}^{24}	-
a3	-	-	e_{CF}^{34}	e_{FG}^{35}
a4	-	-	-	-

$$B_2 = \{e_{AA}, e_{EG}^{4,3}, a3, b1\}$$

 G_{U4} :

	b1	b2	b3	b4
a1	-	-	e_{DF}^{14}	-
a2	e_{DC}^{22}	-	e_{DC}^{24}	-
a3	-	-	-	-
a4	-	-	-	-

$$B_4 = \{e_{AA}, e_{EG}^{4,3}, e_{FG}^{3,5}, a3, b1\}$$

 G_{U7} :

	b1	b2	b3	b4
a1	-	-	e_{DF}^{14}	-
a2	-	-	-	-
a3	-	-	-	-
a4	-	-	-	-

$$B_2 = \{e_{AA}, e_{EG}^{4,3}, e_{FG}^{3,5}, e_{DC}^{2,2}, a3, b1\}$$

 G_{U3} :

	b2	b3	b4	b5
a1	-	-	e_{DF}^{14}	-
a2	e_{DC}^{22}	-	e_{DC}^{24}	-
a3	-	-	e_{CF}^{34}	-
a4	-	-	-	-

$$B_2 = \{e_{AA}, e_{EG}^{3,3}, a3, b1\}$$

 G_{U6} :

	b2	b3	b4	b5
a1	-	-	e_{DF}^{14}	-
a2	-	-	-	-
a3	-	-	e_{CF}^{34}	-
a4	-	-	-	-

$$B_2 = \{e_{AA}, e_{EG}^{3,3}, e_{DC}^{2,2}, a3, b1\}$$

 G_{U5} :

	b1	b2	b3	b4
a1	-	-	-	-
a2	e_{DC}^{22}	-	-	-
a3	-	-	-	-
a4	-	-	-	-

$$B_2 = \{e_{AA}, e_{EG}^{4,3}, e_{CF}^{3,4}, a3, b1\}$$

Figure 5.14: The matrix representation of the nodes in Fig. 5.13c.

 Fig. 5.14. The leaves of the search represent possible transistor chains (S_i).

A graphical representation of the optimum transistor chain for the example is shown in Fig. 5.15a. It can be noticed that, unlike in CMOS layouts, the Euler path spans both the PUN and PDN for obtaining minimum number of breaks in the diffusion region as well as polarity gates. Fig. 5.15b illustrates the dumbbell-stick diagram of the circuit.

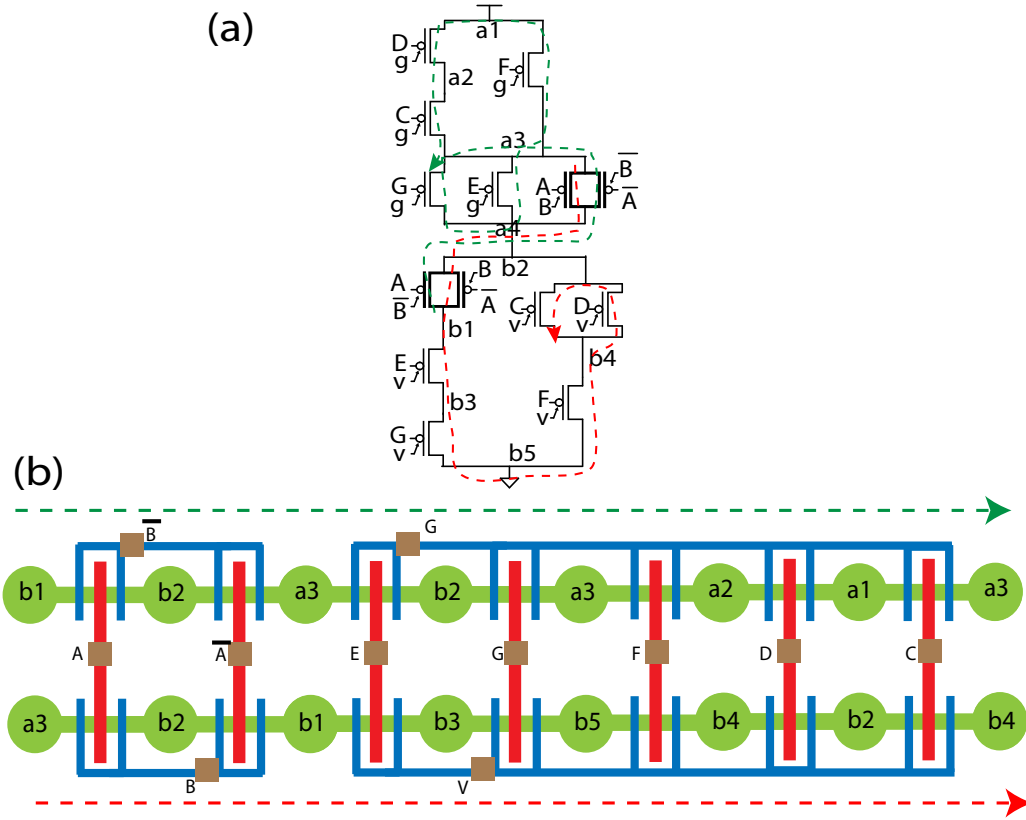


Figure 5.15: (a) Graphical representation of transistor chains derived from Fig. 5.13c. (b) Dumbbell-stick diagram of the circuit.

5.3.6 Examples

Here I illustrate the technique presented in the previous section to few logic blocks with ambipolar logic. Fig. 5.16 illustrates an efficient reconfigurable logic block (F1) with ambipolar transistors, which can implement 12 different sub-functions [132]. Once the dumbbell-stick diagram is extracted from optimal transistor ordering, the final layout of the circuit is done by taking into account the sizing of the transistors for uniform delay caused by the transistors in the PUN and PDN.

In Fig. 5.17, I show the carry-out logic implementation with ambipolar logic. An equivalent implementation with conventional static CMOS logic requires 22 transistors, whereas with ambipolar logic we need 16 transistors (10 shown in the figure along with 3 inverters). By applying the layout procedure I obtain 4 transistor chains, thereby leading to a break in the diffusion region of the dumbbell-stick diagram. Similar to above example, the complete layout of the circuit can be extracted from this DSD after considering the sizing of the transistors.

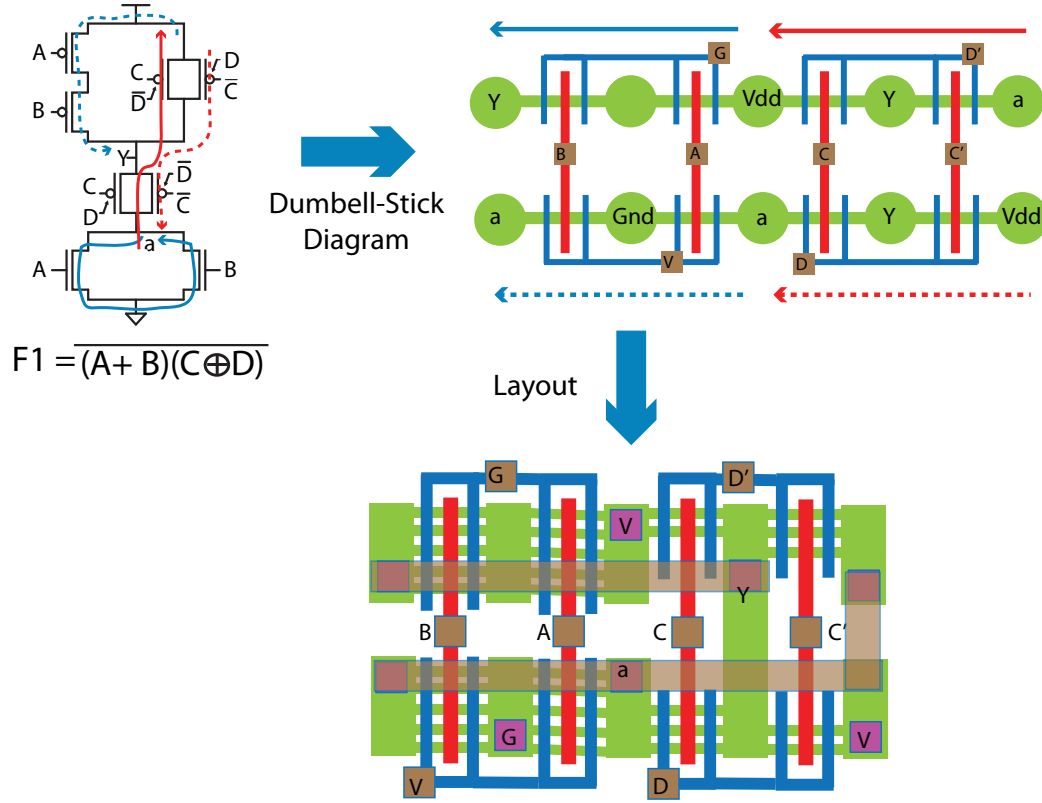


Figure 5.16: Reconfigurable logic block based on ambipolar logic style [132].

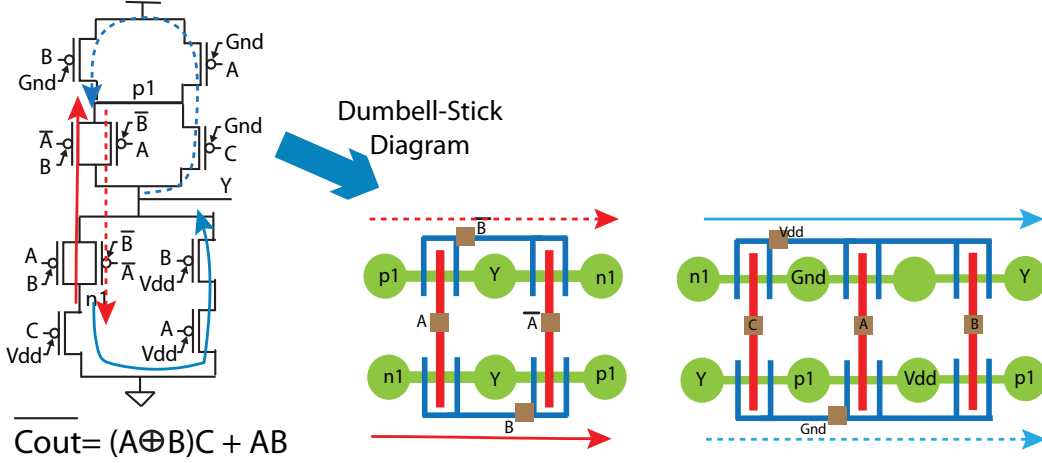


Figure 5.17: Carry-out function of a full-adder.

5.4 Gate-level Technology Mapping

In this section, I present simulation results comparing DG-SiNWFET technology against planar CMOS and FinFET technologies at 22 nm node. First, I develop a compact TCAD model of the device from which I simulate the FO4 delay of various

basic logic gates. Second, I propose an arithmetic cell library comprising of gates with an embedded XOR/XNOR function, in order to study area and performance metrics of various circuits employed in data path design.

5.4.1 TCAD Model of DG-SiNWFET

For the TCAD model of DG-SiNWFET, a single silicon nanowire with 24 nm gate length (i.e., 22nm technology node) is simulated using Synopsys Sentaurus. Metal gates with mid-gap work function are used on the HfO_2 high-k dielectric layer as shown in Fig. 5.18. The Schottky barrier height for electron is set to around 0.35eV (0.75eV for holes) in the simulation, which is achievable in standard process by using barrier height modulation technology, such as selective phase modulation of NiSi [135] or interfacial dielectric dipole [136]. By independently biasing the central gate and the polarity gates to ground or Vdd, the device shows 4 possible operation modes: two ON states and two OFF states. Fig. 5.19 illustrates their corresponding band diagrams. When $V_{ds}=V_{dd}$ (i.e., $S='0'$ and $D='1'$), one of the Schottky barriers is thin enough to allow hole tunneling from drain (p-type) or electron tunneling from source (n-type). Thus, majority carriers can either go through the device (Fig. 5.19(a,b)) or be shut off due to the barrier induced by opposite biasing of central gate and polarity gates as shown in Fig. 5.19(c,d).

The device is simulated based on drift-diffusion transport model. The symmetric characteristics obtained from TCAD simulation are demonstrated in Fig. 5.20, where comparisons are drawn with respect to the predictive technology models of CMOS at 22 nm node [137]. Controllable polarity behavior, with symmetric characteristics, is obtained at a nominal voltage of +1.2V. From the Fig. 5.20, we can observe similar characteristics between CMOS and SiNWFET. In order to further improve the performance, the silicide contacts should locate close enough to the gate-controlled region, and spaces between central gate and polarity gates are helpful for reducing the off-state leakage.

Verilog-A model

To enable a first-order evaluation at the circuit level, a small signal model for the ON state of the NWFET has been written in Verilog-A. The equivalent circuit of a single wire NWFET is described in Fig. 5.21. The core of the model is based on a current source emulating the drain/source current as a function of the voltage applied to the polarity gate and the control gate, in addition to the voltage at the source and drain terminals. The current source is described by a table model extracted from TCAD simulations by sweeping V_{CG} -s, V_{PG} -s and V_{DS} between -1.2V and +1.2V with a step of 0.2V, 0.2V, and 0.05V respectively. The access resistance corresponds to the pillar at drain and source contacts. Each capacitance extraction has been carried out from TCAD simulations as an average value under all possible bias conditions. This model is able to capture the basic behavior of a single wire transistor. In a first order, a stack of several wires might be seen as the parallel interconnection of several NWFETs. Then, a

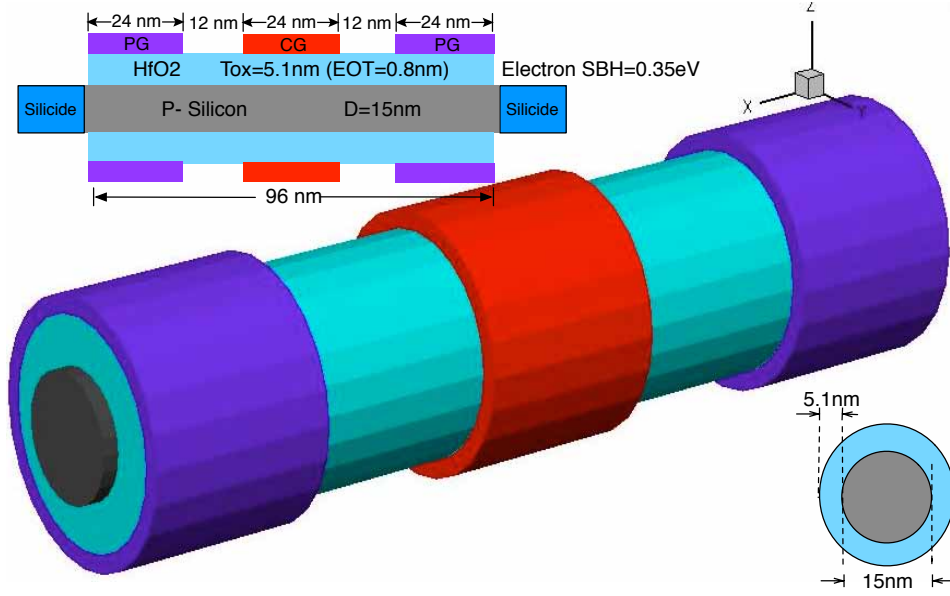


Figure 5.18: The schematic of the ambipolar silicon nanowire used in TCAD.

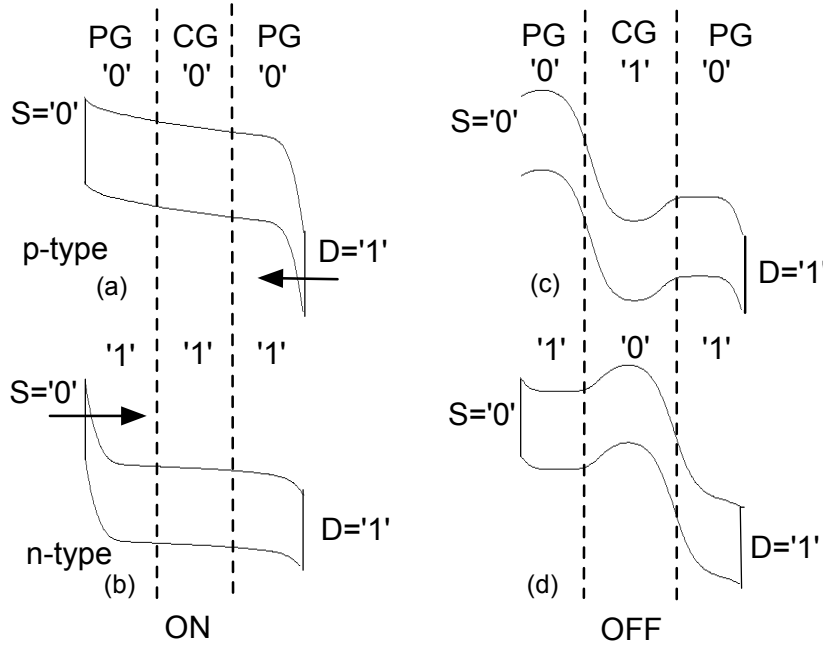


Figure 5.19: Band diagram of the SiNWFET.

stack of wires is modeled by the parallel arrangement of single transistor model.

5.4.2 FO4 delay of Basic Logic Gates

In the previous sections, I demonstrated the effectiveness of DG-SiNWFET, in terms of transistor count, for realizing various types of Boolean functions.

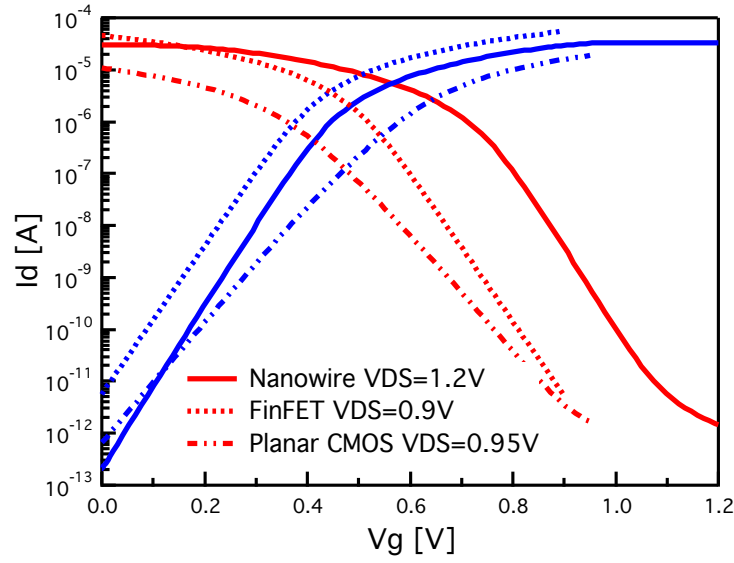


Figure 5.20: Symmetric characteristics of ambipolar SiNWFET from TCAD simulation.

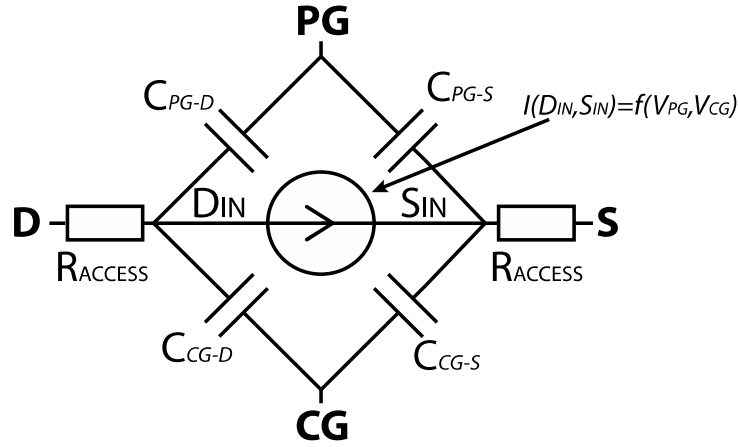


Figure 5.21: Single NWFET equivalent circuit.

Here I show the performance improvement by simulating the FO4 delay for various logic gates and comparing them to traditional static logic style. Table 5.2 reports the FO4 delays of various logic gates. I report the average delay by considering both the raise and fall delays. Clearly logic gates realized with FinFETs and SiNWs fare better when compared to planar Si-CMOS due to the improved electrostatics of the device. In the case of DG-SiNWFET, I report delays $CG \rightarrow Out$ (input fed through control gate) and $PG \rightarrow Out$ (input fed through polarity gate). For unate functions, PGs are connected to V_{dd} and Gnd , hence $PG \rightarrow Out$ is listed as X in the Table 5.2. On the other

Logic gates	FinFET (ns)	DG-SiNW (ns)		Si-CMOS (ns)
		CG -> Out	PG -> Out	
INV	0.056	0.043	X	0.139
NAND2	0.066	0.05	X	0.148
NOR2	0.062	0.072	X	0.172
XOR2	0.081	0.046	0.108	0.191
XOR3	0.198	0.079	0.11	0.38

FinFET: L=24 nm, W=15 nm, H=28 nm, N=1

DG-SiNW: L=24 nm, 6 NWs per stack

Planar CMOS: L=22 nm

Table 5.2. FO4 delay of basic logic gate.

hand for XOR2 and XOR3, the PGs are connected to the input signals.

For unate logic gates (INV, NAND2, NOR2), I observed similar delay characteristics for both FinFET and DG-SiNWFET. In the case of XOR2 with DG-SiNWFET, I observe a faster switching time for the input connected to the CG (0.046 ns), when compared to the input connected to the PG (0.108 ns). For the overall delay of the XOR2, I consider the average delay with inputs fed through both CG and PG. Compared to a FinFET implementation, an XOR2 with DG-SiNWFET shows 6% improvement in delay. A compact implementation of an XOR3 [130] is possible by employing the transistor pairs of an XOR2 as transmission gates (see Fig. 16). Since I keep the same transistor count for realizing the XOR3 operation, the delay of an XOR3 gate is comparable to XOR2 gate, unlike static CMOS logic. From Table 5.2, I can observe that the average delay of an XOR3 gate with DG-SiNWFET is only 16% higher when compared to XOR2 gate. On the other hand, for static CMOS the delay of an XOR3 is twice to that of XOR2 gate.

5.4.3 Gate-level Mapping of Arithmetic Circuits

Arithmetic circuits are one of the most important applications for double-gate transistors due to the dominant presence of XOR/XNOR based functions. In this section, I leverage on the performance study of basic gates (Table 5.2) in order to evaluate the effectiveness of ambipolar logic on arithmetic circuits. I propose an arithmetic cell library comprising basic arithmetic building blocks (Table 5.3). Various design schemes with minimum number of transistors are employed, for both ambipolar logic and conventional CMOS logic (unipolar transistors), while ensuring full voltage-swing at the output.

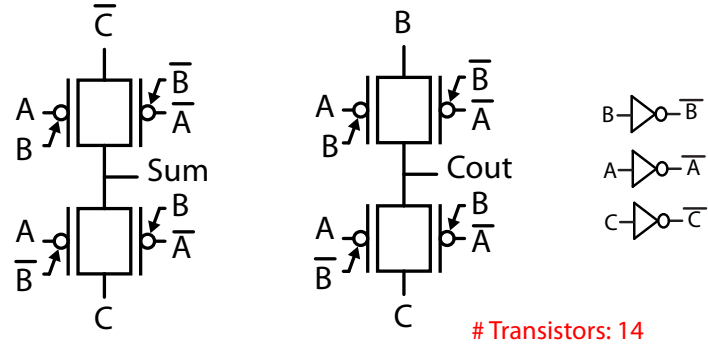
Gate	Ambipolar Logic		CMOS logic	
	Area*	Delay*	Area*	Delay*
XOR2	8	1	12	1
XOR3	10	1.19	24	2
Half adder	12	1	16	1
Full adder	14	1.19	18	2
4-2 Compressor	26	2	42	4
5-3 Compressor	38	3	64	4

* Area normalized to the number of transistors employed.

** Delay normalized to the XOR2 delay.

Table 5.3. Arithmetic cell library.

Double-gate FETs with controllable polarity



Conventional FETs

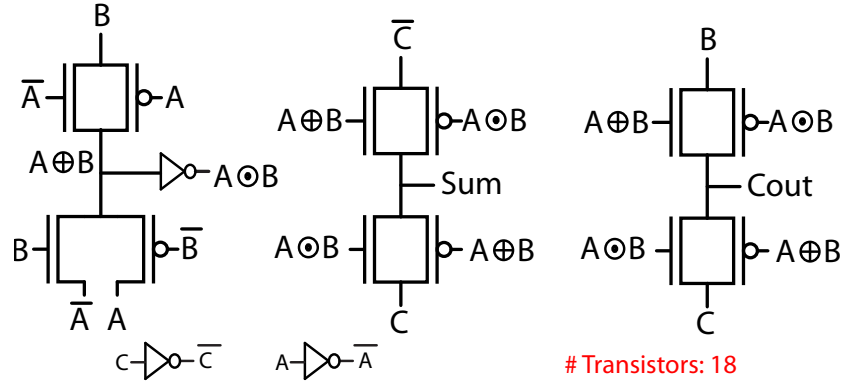


Figure 5.22: Full-adder implementation with DG-SiNW transistors and conventional CMOS transistors.

In Table 5.3, I report area and delay of various gates. The area is normalized to the number of transistors employed to realize the function, whereas the delay is normalized to the delay of an XOR2 gate (listed in Table 5.2). A compact full-adder (FA) realized with DG-SiNWFET, is compared to its equivalent FinFET implementation in Fig. 5.22. It has to be noted that the FA forms a fundamental building block for many arithmetic circuits. A FA

with DG-SiNWFETs shows 22% improvement in area and 40% improvement in normalized delay when compared to its implementation in CMOS logic style. I observe similar improvement in area and delay of the compressors, as they are composed of multiple FAs.

By employing the arithmetic cell library, I study various industry standard benchmark circuits comprising of adders, multipliers, compressors and counters as listed in Table 5.4. From the table below, I observe that the ambipolar logic consistently fares well when compared to the conventional CMOS logic in both area (32% on average) and delay (38% on average). In Fig. 5.23, I show the improvement in performance for different categories of arithmetic circuits. I observe major performance gain in the reduction trees of the multipliers.

Benchmarks	Ambipolar Logic		CMOS logic	
	Area*	Delay*	Area*	Delay*
5-bit ripple-carry adder	68	5.76	88	9
16-bit carry-select adder	392	6.76	504	10
(29, 3)-compressor	408	12	680	16
(11, 2)-reduction tree (16 x 16 MAC)	124	6	226	10
Wallace tree (54 x 54 multiplier)	338	8	546	16
(31, 5) Parallel counter	364	8.33	468	14

Table 5.4. Arithmetic benchmark circuits.

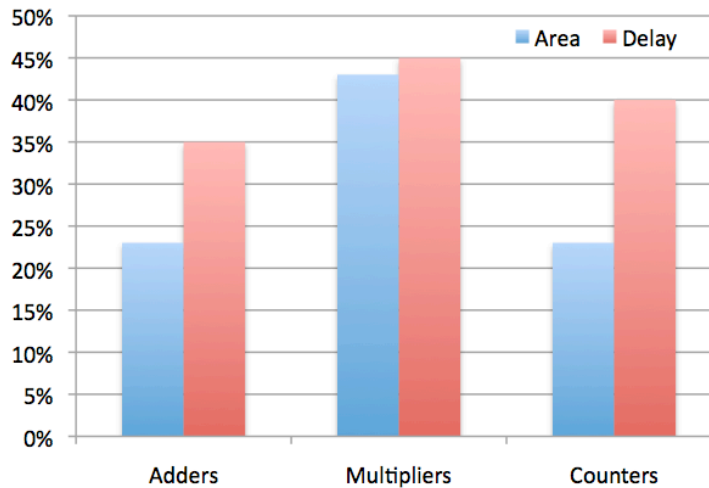


Figure 5.23: Average performance improvement for various components of data path circuits.

5.5 Chapter Contribution

In this chapter, I introduce DG-SiNWFET technology and its prospects in realizing ambipolar logic circuits. Among the various nanotechnologies, based on carbon nanotubes or graphene, SiNWFET is favorable due to their top-down silicon process [119]. With two independent gates, a control gate and a polarity gate, a DG-SiNWFET can be field programmed to either p- or n-type transistor. Though this feature of controllable polarity shows huge potential for novel design methodologies in circuit design, one of the fundamental problem at a physical level lies in mitigating the gate-level routing congestion caused by the need to access the two independent gates of each and every transistor.

This chapter contributes to this fundamental physical design problem for ambipolar logic circuits. In order to facilitate this, I propose novel symbolic layouts for ambipolar logic with Dumbell-Stick diagrams. Ambipolar logic outperforms conventional CMOS static logic for Boolean functions with embedded XOR/XNOR functions. The main contribution of this chapter is a layout methodology and algorithm for complex functions with embedded XOR/XNOR block.

Furthermore, in order to study the effectiveness of DG-SiNWFET technology when compared to CMOS technologies, I present a first-order model of the device at 22 nm node. By mapping various arithmetic circuits, I obtain 32% average improvement in area and 38% improvement in delay.

In the following chapter, I extend the proposed layout technique to design the fundamental building block for ambipolar logic circuits. With the help of technology mapping, an optimal layout fabric is found which is further optimized for delay by varying the number of stacked silicon nanowires.

Sea-of-Tiles Fabric for DG-SiNWFET Circuits

6

In the previous chapter, I presented a layout technique for mitigating the routing congestion caused by two independent gates of DG-SiNWFET circuits. In this chapter, I propose regular layout fabric for DG-SiNWFET technology based on the layout technique presented in Section 5.3.

Layout regularity is one of the key features required to increase the yield of ICs at advanced technology nodes [138]. Hence, design styles based on regular layout fabrics have the advantage of higher yield as they maximize the layout manufacturability. Various regular fabrics have been proposed throughout the evolution of semiconductor industry, where some recent approaches are discussed in [139, 140, 141]. In gate-array fabric style, a sea of prefabricated transistors is customized to obtain a desired logic gate. The flexibility of building generic logic gates comes at a cost of area as well as routing overhead, thereby increasing the performance gap between ASICs and gate arrays. With the advent of via-programmable gate arrays [140] and logic-bricks [141], the performance gap is reduced. On the other hand, strict design rules, at 22 nm technology node and beyond, has led to cell layouts with arrays of gates with a constant gate pitch, which resemble a sea-of-gates layout style.

In this chapter, I propose an efficient regular layout fabric (called as *tile*), which forms the basic building block for DG-SiNWFET based circuits. With *Sea-of-Tiles* (SoT) design methodology I envisage a regular arrangement of an array of tiles (6.1). Any desired logic function can be mapped onto an array of logic tiles. Hence, there is a need to find the optimal tile. In this study, I optimize for area and regularity (which improves the overall yield). Technology mapping, with logic synthesis tools, on various tiles helped us in choosing an efficient tile for realizing SoT. With the optimal tile as a

basic building block for a SoT fabric, I demonstrate mapping any 3-input NPN-equivalent function [142, 143] as well as several other building blocks proposed for ambipolar logic circuits [132, 131]. However, since a unique tile is replicated in the SoT approach, correct tile sizing is crucial for the overall circuit performances. By tile sizing, the minimum width of all the transistors in the tile is set uniformly. The minimum width of the transistor is determined by the number of vertically stacked SiNWs.

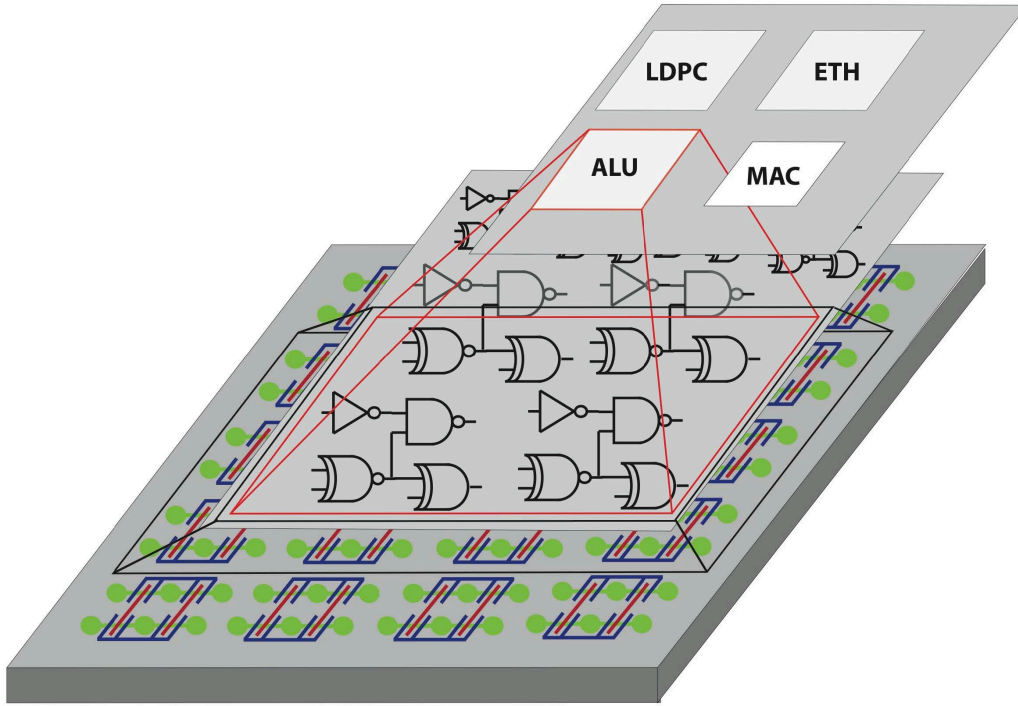


Figure 6.1: *Sea-of-Tiles* (SoT) design methodology.

The main contributions of this chapter are:

- Design of an efficient regular layout fabric (*tile*), which forms the basic building block for a *Sea-of-Tiles* (SoT) design methodology. I show how various Boolean functions can be mapped onto the SoT fabric of the optimum tile.
- Circuit-level benchmarking study by sizing the tiles with respect to the number of vertically-stacked SiNWs of a DG-SiNWFET. The information on the number of stacked nanowires is important to the technologists in order to optimize the fabrication of the layout fabric.

The chapter is organized as follows. In the first part of this chapter, I present the idea of logic tiles and choose a set of tiles for our study. Then, I investigate the area optimal tile with the help of technology mapping with traditional synthesis tools. Next, various 3-input NPN-equivalent function along

with various building blocks for ambipolar logic circuits are mapped onto the array of area optimal logic tiles. In the second part of this chapter, I study technology mapping of various benchmark circuits by varying the number of stacked SiNWs of the SoT fabric. Finally, the chapter is concluded by discussing the results and by summarizing the contributions of this part of the thesis.

6.1 Logic Tiles as Building Blocks

A logic *tile* is defined as an array of transistors, which are paired and grouped together. By grouping the polarity gates of the adjacent transistors, I reduce the number of input pins of the tile. Moreover, the technology facilitates in realizing these tiles with a high yield as the silicon nanowires are fabricated in groups. In this work, I limit my study to a maximum of three transistors in series for noise margin reasons. However, the proposed design methodology can be employed to tiles with higher number of series connected transistors.

A Tile_{Gn} , is an array of n transistor-pairs grouped together. Figure 6.2 shows four tiles that I consider for the *sea-of-tiles* (SoT) architecture. Any Boolean logic function can be mapped on to an array of tiles. Tile_{G1} (Fig. 6.2a) is the simplest tile with only one pair of transistors. Mapping a generic logic function onto an array of Tile_{G1} (also called as SoT of Tile_{G1}), leads to larger layouts with a large number of diffusion breaks and increase in the number of interconnections per tile. Tile_{G2} and Tile_{G3} include two and three transistor pairs, respectively, grouped together. In the example of carry-out logic gate of a full-adder (see Fig. 5.17), Tile_{G2} and Tile_{G3} are employed to realize the gate. Similarly in the case of NAND (Fig. 5.10) and XOR (Fig. 5.11), Tile_{G2} forms the basic building block. A hybrid tile, Tile_{G1h2} (Fig. 6.2c), is a combination of Tile_{G1} and Tile_{G2} , whose polarity gates are not connected. This gives the flexibility of utilizing a part of a tile, when remained un-mapped, by functions with low area utilization. For example, a NAND2 gate when mapped onto a Tile_{G1h2} requires only the segment of a tile with gates G1 and G2. The unmapped part of the tile with gate G3 can be employed either to map an inverter or to increase the drive strength of the gate.

The Tile_{G2} , shown in the Fig. 6.2b, can be configured to various logic functions by connecting the nodes (n1-n6) and gates (g1, g2, G1 and G2) to appropriate inputs. Table 6.1 lists various logic functions that can be realized with a single Tile_{G2} . However, any complex logic function can be obtained by considering an array of Tile_{G2} . In Table 6.2, I report various logic gates that can be configured with the 4 tiles I have considered. The number of tiles required for each gate and their respective area utilization is also presented. The extra logic needed for generating the inverted inputs is considered in the area evaluation. For example the 2-input XOR gate, shown in case-2 of Fig.

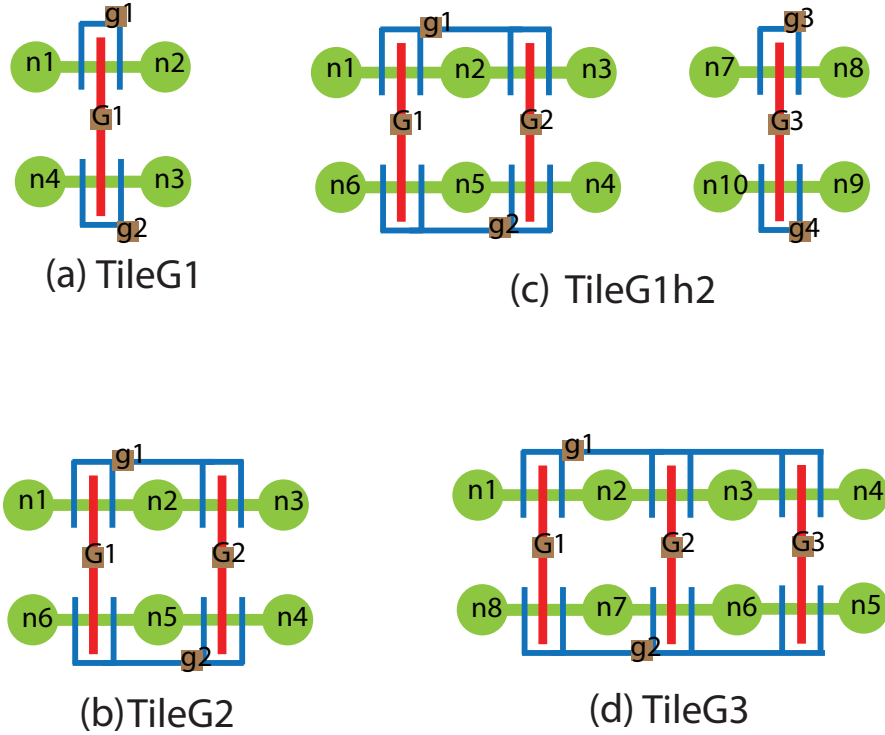


Figure 6.2: Dumbbell-stick diagrams of various logic tiles considered for SoTs (a) Tile_{G1} (b) Tile_{G2} (c) Tile_{G1h2} (d) Tile_{G3} .

5.11, employs only one Tile_{G2} as I assumed the availability of complimented input signals. In our technology mapping, I assume single-rail logic; hence I need to generate the complimented signals when needed. In Table 6.2, I report 2 tiles of Tile_{G2} for XOR2 implementation, where one of the Tile_{G2} 's is considered for generating the two negated input signals ($\overline{A}$ and $\overline{B}$).

Logic	n1	n2	n3	n4	n5	n6	G1	G2	g1	g2
XOR2	Gnd	Out	Vdd	Gnd	Out	Vdd	A	A'	B'	B
XNOR2	Gnd	Out	Vdd	Gnd	Out	Vdd	A	A'	B	B'
NAND2	Out	Vdd	Out	Out	-	Gnd	A	B	Gnd	Vdd
NOR2	Vdd	-	Out	Out	Gnd	Out	A	B	Gnd	Vdd
INV	Vdd	Out	Vdd	Gnd	Out	Gnd	A	A	Gnd	Vdd
BUF	01	Vdd	Out	Out	Gnd	01	A	01	Gnd	Vdd

Table 6.1. Various logic gates that can be realized by configuring the Tile_{G2} .

Gates	Tile _{G1}		Tile _{G2}		Tile _{G1h2}		Tile _{G3}	
	#N	#UF	#N	#UF	#N	#UF	#N	#UF
AND2	3	0.6	2	0.6	1	0.75	1	1
AND3	4	0.57	2	0.8	1.38	0.67	2	0.57
AOI21	3	0.6	2	0.6	1	0.75	1	1
AOI221	5	0.56	3	0.625	1.62	0.71	2	0.71
AOI222	6	0.54	3	0.75	2	0.67	2	0.86
AOI22	4	0.57	2	0.8	1.38	0.67	2	0.57
AOI321	6	0.54	3	0.75	2	0.67	2	0.86
BUF	2	0.66	1	1	0.62	1	1	0.67
INV	1	0.66	1	1	0.38	1	1	0.67
NAND2	2	0.66	1	1	0.62	1	1	0.67
NAND3	3	0.6	2	0.6	1	0.75	1	1
NAND4	4	0.57	2	0.8	1.38	0.67	2	0.57
NOR2	2	0.66	1	1	0.62	1	1	0.67
NOR3	3	0.6	2	0.6	1	0.75	1	1
NOR4	4	0.57	2	0.8	1.38	0.67	2	0.57
OAI21	3	0.6	2	0.6	1	0.75	1	1
OAI22	4	0.57	2	0.8	1.38	0.67	2	0.57
OR2	3	0.6	2	0.6	1	0.75	1	1
OR3	4	0.57	2	0.8	1.38	0.67	2	0.57
XNOR2	8	0.57	2	0.8	1.38	0.67	2	0.57
XNOR3	9	0.56	3	0.625	1.62	0.71	2	0.71
XOR2	8	0.57	2	0.8	1.38	0.67	2	0.57
XOR3	9	0.56	3	0.625	1.62	0.71	2	0.71

Table 6.2. Various logic gates that can be mapped by configuring the contacts and the input signals of the four tiles (#N – Number of tiles, and #UF – Utilization factor).

6.2 Area Optimal Tiles

In this work I compare four tiles for an efficient implementation of the SoT architecture. Our main objective is to find the best tile, which gives highest area utilization for various benchmarks. Though the techniques presented in this work are linked to the ambipolar DG-SiNWFETs, the concepts can be extended to all the technologies contending for ambipolar logic circuits with top-gated double-gate transistors (i.e. where the all the gate contacts are accessed from the top of the transistor channel). For instance the concept of tiles can be extended to double-gate Carbon nanotube FET presented in [130].

Figure 6.3 shows our design flow. As a first step, for every tile (Tile_{G_i}) I generate a list of logic gates that can be mapped on to it (*TileG_i.lib*) and their respective utilization factor (*TileG_i.util*). Utilization factor takes only the active area into account. For example NAND2 when mapped onto a Tile_{G1} has a utilization factor of 0.66, whereas when mapped onto a Tile_{G2} it has a utiliza-

tion factor of 1. It has to be noted that the number of logic gates that can be mapped to different tiles vary. For technology mapping, I used Synopsys design compiler [97] and ABC [144] synthesis tools to benchmark various circuits.

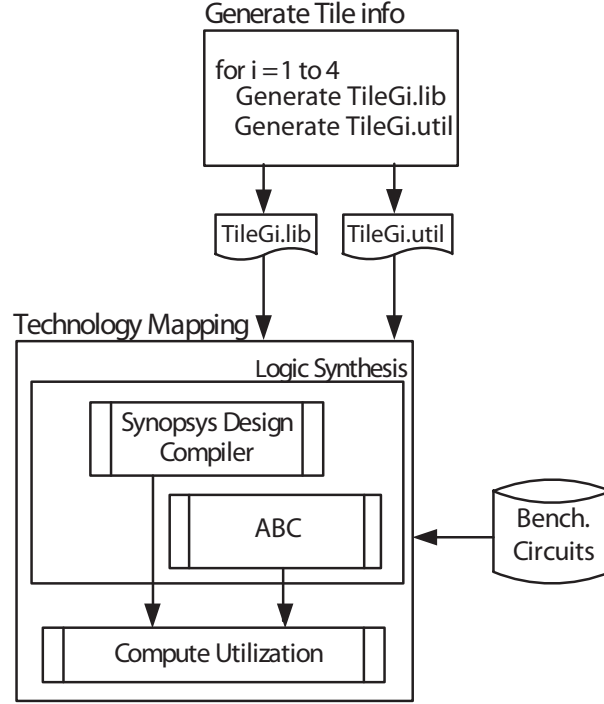


Figure 6.3: Design flow for finding the best Tile for SoT.

Table 6.3 summarizes the results of various benchmark circuits after technology mapping. I report total area utilization for each benchmark when mapped onto four different tiles (Tile_{G1} , Tile_{G2} , Tile_{G1h2} , and Tile_{G3}). Technology mapping only uses the cells that are associated with each tile (shown in Table 6.2). Both the synthesis tools were run with different delay constraints. Area utilization for a benchmark circuit is calculated from the total count of each cell and their respective utilization factors.

Examining the results for the four logic tiles, I observe that SoT with tiles Tile_{G1h2} (and, Tile_{G2}) have a higher area efficiency, 10% (8%) and 16% (14%), when compared to SoT with Tile_{G1} and Tile_{G3} , respectively. Though Tile_{G3} and Tile_{G1h2} have the same number of transistors per tile, the hybrid tile outperforms Tile_{G3} with 10% improvement in area efficiency. The main reason behind this is due to the high utilization factor of Tile_{G1h2} in realizing fundamental logic gates like INV, BUF, NAND2, and NOR2 (see Table 6.2).

In this study, I determined the best possible tile based on the area employed by an array of tiles after mapping various circuits. Based on our simulations,

Bench.	Tile _{G1}		Tile _{G2}		Tile _{G1h2}		Tile _{G3}	
	DC	ABC	DC	ABC	DC	ABC	DC	ABC
Dalu	1968	2558	1728	2235	1689	2115	1808	2548
Add64	3946	3004	3693	2664	3483	2483	3560	2740
C5315	4072	5404	3465	4791	3422	4477	3984	5088
C7552	4914	5606	4188	5001	4150	4653	4752	5456
i10	5964	6350	5034	5634	4790	5286	5452	6232
C1908	1132	1778	936	1518	942	1469	1116	1692
C3540	2940	3436	2517	3033	2486	2859	2756	3184
C6288	8462	9336	7227	8253	7373	7744	7580	8000
Des	9392	12482	8142	10623	7910	10323	9016	11912
Average	1	1	0.86	0.87	0.85	0.83	0.94	0.94

Table 6.3. Normalized area of various benchmarks when mapped onto a SoT with Tile_{G1}, Tile_{G2}, Tile_{G1h2}, and Tile_{G3}.

both Tile_{G1h2} and Tile_{G2} yield high area efficiency for logic functions upto 4-input Boolean variables. The sizing of these tiles, by varying the number of vertically stacked SiNWs, is studied in Section.6.4.

6.3 Case Studies

In this section, I map various logic functions onto an array of tiles comprising Tile_{G2} and Tile_{G1h2}. First, I map all 3-input Boolean functions onto SoT with Tile_{G2}, followed by mapping various blocks unique to ambipolar logic circuits proposed in the literature. I show how Tile_{G2} and Tile_{G1h2} can be the basic building block for the future ambipolar logic circuits with double independent gates.

6.3.1 Mapping 3-input Boolean Functions onto SoT of Tile_{G2}

I study the mapping of 3-input Boolean functions by considering the matching compatibility graph for 3-input Boolean space [142] (illustrated in Fig. 6.4). Each vertex V_i in the graph, is annotated with one function F_i , which belongs to the corresponding NPN-equivalence class [145, 146] of V_i . All the functions (F_i), listed in Table 6.4 are representative of a NPN-equivalence class. In other words, all (e.g., 256) 3-input functions can be obtained from the 13 representative functions (in Fig. 6.4) by input complementation and/or permutation and/or output complementation. The type of the function along with the number of transistors needed for implementing in static CMOS and ambipolar logic styles are listed in Table 6.4. Type of the function corresponds to F_i being unate (U), binate (B), mixed (M) and mixed with embedded XOR/XNOR (XM) (refer Section. 5.2.1). I compare the transistor count for realizing the functions with both static CMOS and ambipolar logic implementation. I do

not take into account the inverters needed for input and output negations, as they are similar for both the logic styles. From the table I infer that 30% of the total NPN-equivalent functions have embedded XOR/XNOR (F_3 , F_7 , F_{12} and F_{13}). Hence, ambipolar logic is ideal for realizing these logic functions.

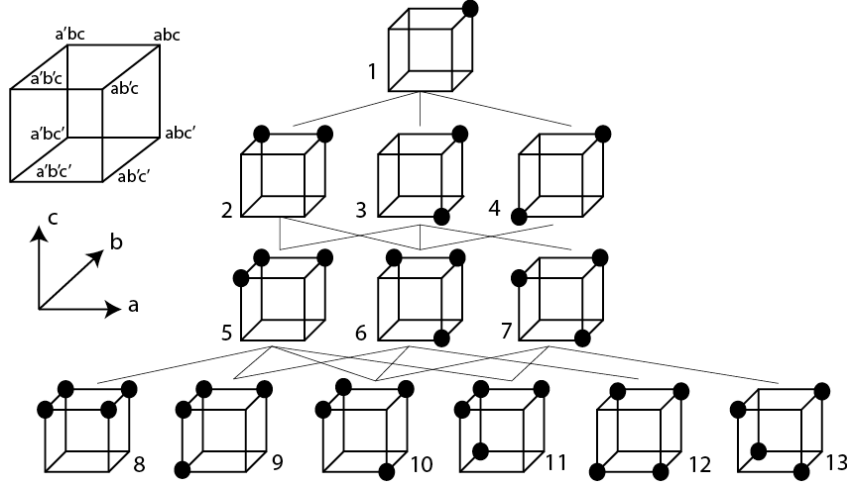
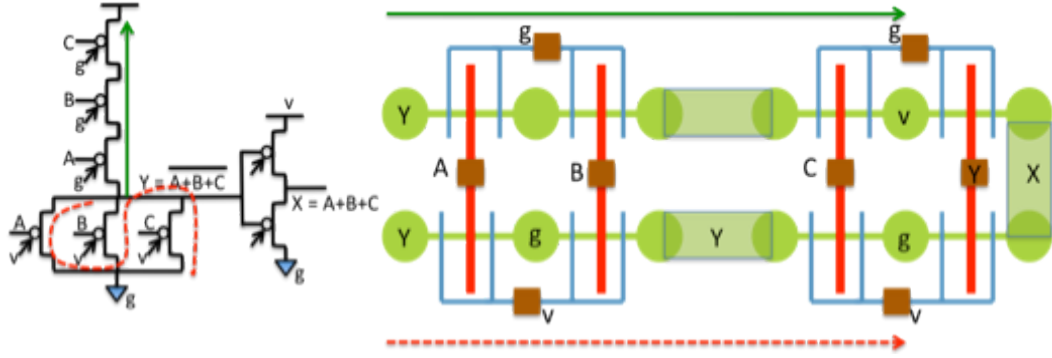


Figure 6.4: Matching compatibility graph for 3-input Boolean space.

Boolean space	Representative functions	Type	Transistor count	
			Static CMOS	Ambipolar Logic
F_1	abc	U	6	6
F_2	ab	U	4	4
F_3	$a(b^{\wedge}c)$	XM	12	6
F_4	$abc+a'b'c'$	B	12	12
F_5	$c(b+a)$	U	6	6
F_6	$bc+ab'c'$	M	10	10
F_7	$abc+b'(c^{\wedge}a)$	B	18	12
F_8	c	U	2	2
F_9	$bc+a'b'$	M	8	8
F_{10}	$c(b+a') + ab'c'$	B	12	12
F_{11}	$cb + ca' + a'bc'$	M	12	12
F_{12}	$(b^{\wedge}c)'$	XM	8	4
F_{13}	$a^{\wedge}b^{\wedge}c$	XM	18	10

Table 6.4. NPN-equivalent functions, of a 3-input Boolean space, implemented in static CMOS and ambipolar logic styles. Type of the function corresponds to unate (U), binate (B), mixed (M), and mixed with embedded XOR/XNOR (XM).

Figure 6.5: Mapping of 3-input OR function (F_1).

Layout synthesis technique, presented in Section 5.3, is applied to the functions listed in Table 6.4 for mapping them onto a SoT of Tile_{G2} . In the case of unate functions, the layout technique is similar to the traditional CMOS style. As a generic example for unate functions, I map a 3-input OR function on to a pair of adjacent tiles (see Fig. 6.5). Mapping of all the representative functions with embedded XOR/XNOR (F_3 , F_7 , F_{12} and F_{13}) is depicted in Fig. 6.6. Embedded XOR functionality is one of the key features of ambipolar logic gates. With a transmission-gate transistor structure [130], a 2-input and a 3-input XOR/XNOR gate can be constructed using only 4 transistors. In Fig. 6.6 (refer F_{12} and F_{13}) Tile_{G2} can be configured to be XOR2 and XOR3 by changing the connection to the source/drain contacts of the transistors comprising the tile. It has to be noted that extra tiles are needed to generate the complemented input signals.

6.3.2 Mapping Various Blocks onto Sea-of-Tiles of Tile_{G2} and Tile_{G1h2}

Several novel reconfigurable blocks have been proposed which leverage upon embedded XOR functionality of ambipolar logic. In Figure 6.7, I demonstrate how a computational fabric (F1) [132] and a universal logic module (3,2-ULM) [131] can be mapped onto a SoT of Tile_{G2} . Inverted inputs, for a 2-input XOR functions, are generated with a single tile ($\text{Tile}-(i,j)$ for 3,2-ULM and $\text{Tile}-(i,j+2)$ for F1).

In Figure 6.8, I show dumbbell-stick diagrams of both the sum (Sum) and carry-out (Cout) logic of a full-adder, mapped onto a SoT (an array of $n \times n$) with Tile_{G1h2} . The layout synthesis procedure, explained in Sec. 5.4, is applied to obtain the optimal transistor chaining of the Cout logic. Both the Sum and Cout logic blocks are mapped onto 3 adjacent tiles of the $n \times n$ array. $\text{Tile}-(i, j)$ in the figure refers to the location of the tile in i^{th} row and j^{th} column. The Sum , which is a 3-input XOR of inputs A , B and C , is mapped on to a $\text{Tile}-(i+1,j)$ of the entire array. The unmapped part of the

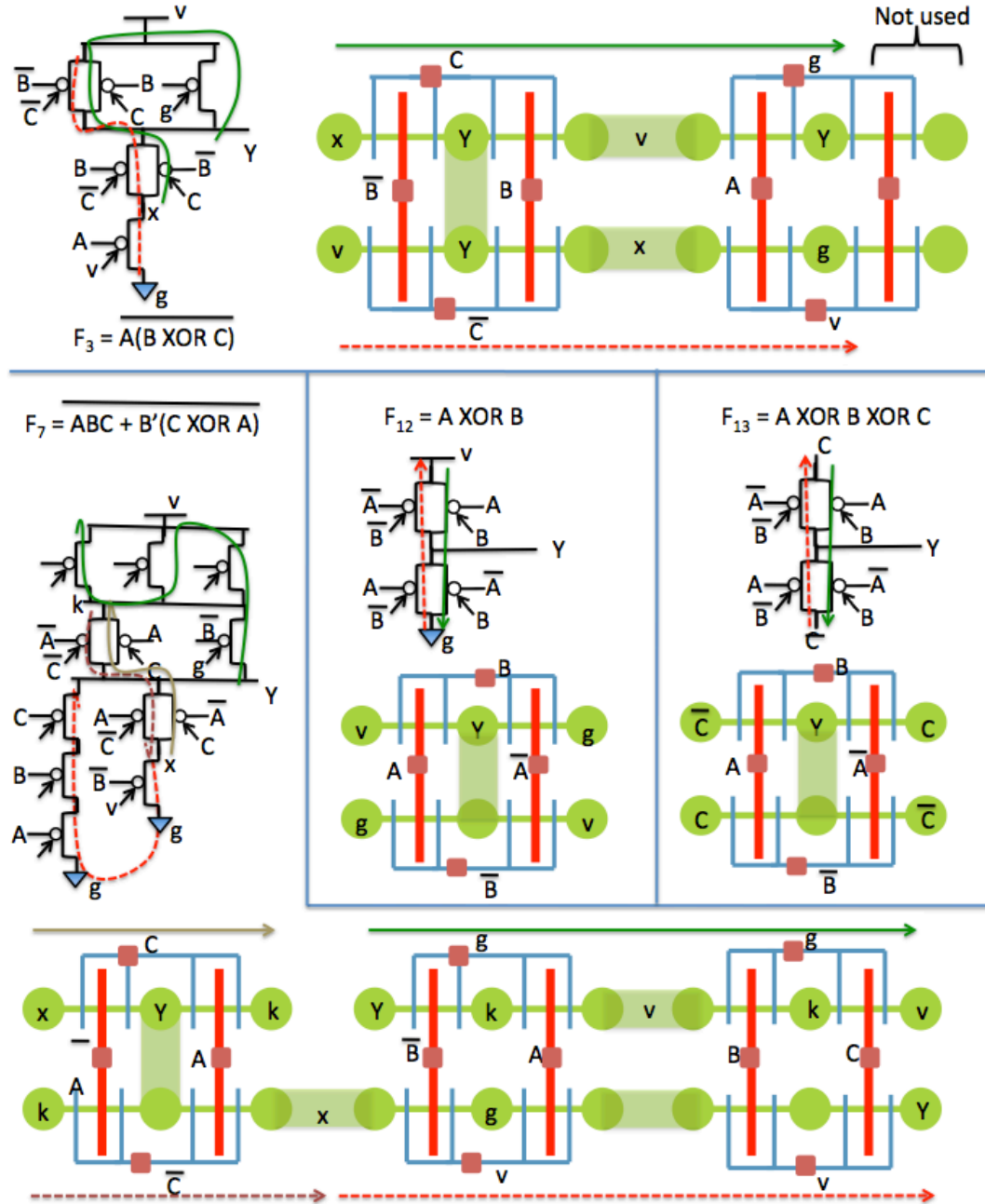


Figure 6.6: Mapping of 3-input NPN equivalent functions with embedded XOR/XNOR (F_3 , F_7 , F_{12} and F_{13}).

Tile-(i+1,j) can be employed for realizing either an inverter logic gate or can be a part of the neighboring logic gate. Similarly the C_{out} is mapped on to 2 tiles Tile-(i,j) and Tile-(i,j+1).

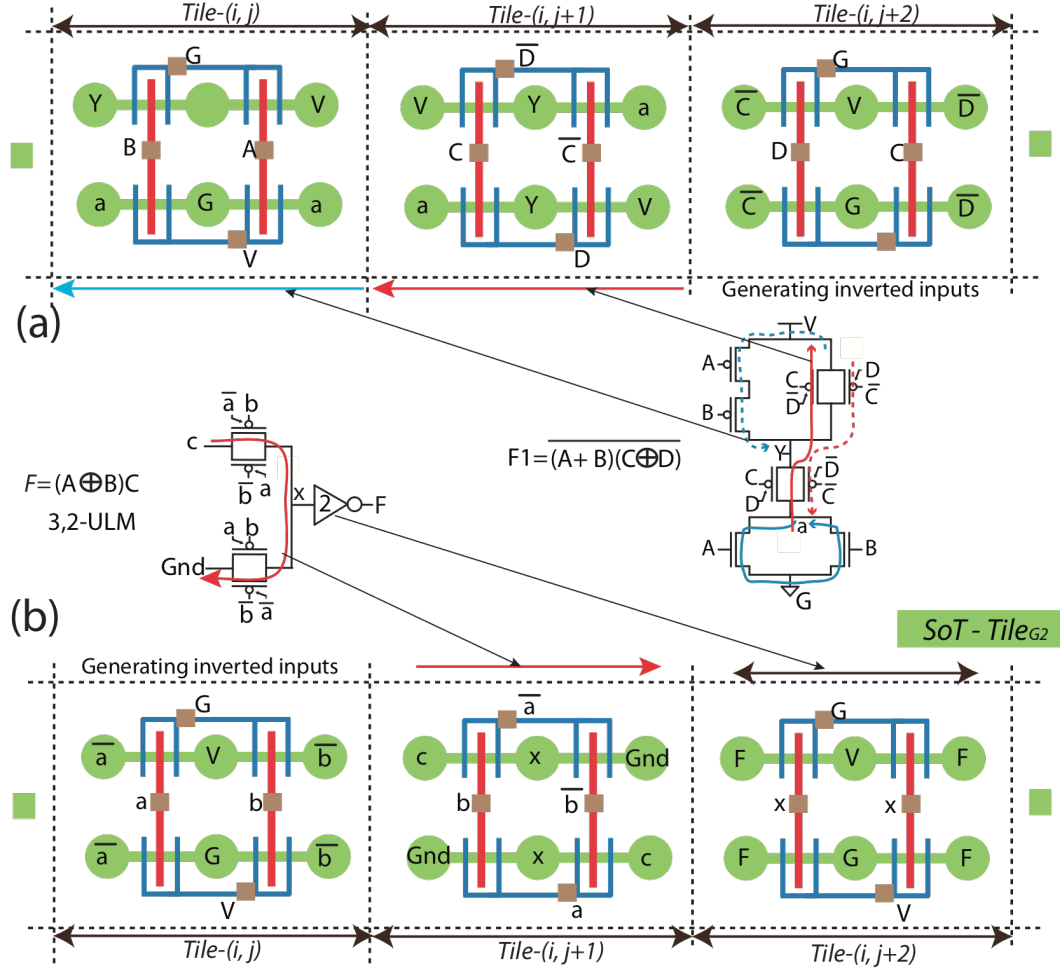


Figure 6.7: Reconfigurable fabrics mapped on to SoT with Tile_{G2} (a) Regular computation fabric [132] (b) Universal logic module (3,2-ULM) [131].

6.4 Sizing the Tiles with Circuit-level Benchmarking

In this section, I study the sizing of regular logic tiles (for DG-SiNWFET technology). By sizing I mean the number of silicon nanowires that are vertically stacked to form the basic transistor. In this study, I consider a uniform array of Tile_{G2} with the same number of vertically stacked nanowires for all the transistors on the wafer. However, the proposed design methodology can be employed for other tiles. First, I present the experimental setup with which I leverage this study starting from gate-level simulation to circuit-level benchmarking of DG-SiNWFET technology.

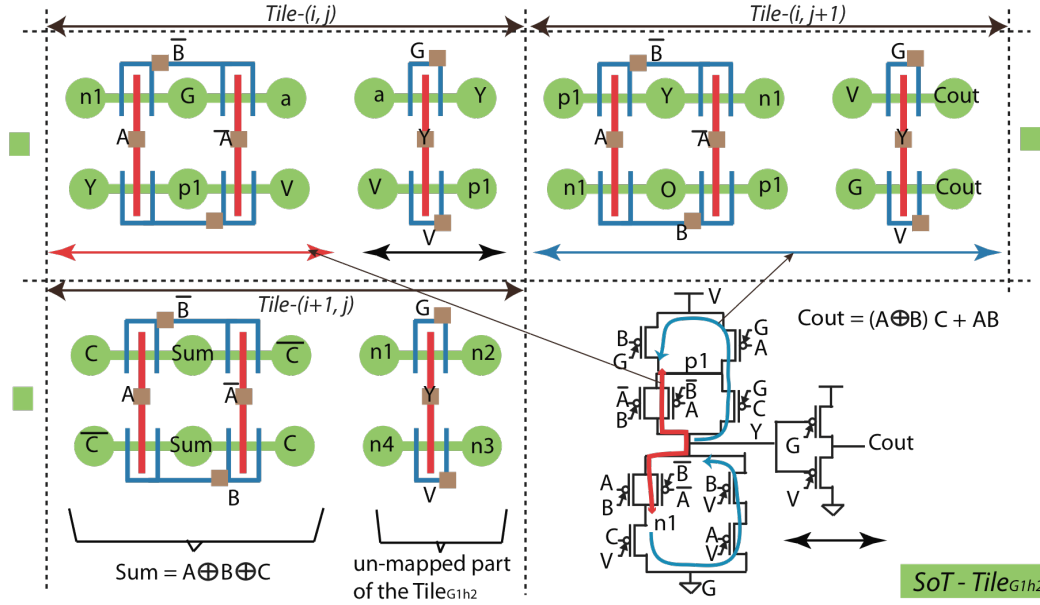


Figure 6.8: A Full-adder mapped on to a Sea-of-Tiles with the hybrid tile Tile_{G1h2} as the basic building block.

6.4.1 Experimental Setup

The overall design flow to size the transistors of the Tile_{G2} , by varying the number of vertically stacked nanowires, is shown in the Fig. 6.9. Various cell libraries for Tile_{G2} were generated with a varying set of vertically stacked silicon nanowires (from 1 to 6). With the help of TCAD modeling (explained in Section 5.4.1) of the nanowire FET, I characterized the electrical performances of the DG-SiNWFET transistors. From the physical simulations, a basic compact Verilog-A model is derived (see Section 5.4.1), in order to enable fast electrical circuit simulation. Different flavors of the library were generated based on the number of vertically stacked nanowires to form the channel (with stacks of 2, 4, and 6 nanowires). The set of logic cells consists of 16 combinational logic cells such as NAND2, NAND3, NOR2, AOI21, ... and one D flip-flop with asynchronous reset and preset. Characterization was performed with Encounter Library Characterizer tool [93]. With the generated *lib* file, I synthesize various benchmark circuits [98] using Synopsys Design Compiler [97]. I consider timing, leakage power and area reports to compare the performance of logic tiles (Tile_{G2} with varying stacked SiNWs) to traditional CMOS at various technology nodes. CMOS counterpart libraries have been generated using PTM models [137]. The nominal voltages for the different technologies have been used, such as 0.95V for CMOS at 22nm node and 1.2V for SiNWFETs. The gate sizing respects the Nangate library [88] sizing, and ideal scaling have been applied between the different technology nodes (45nm and 22nm). In addition to the gate characterization, a simple ideally scaled model for the wire load is added to the libraries.

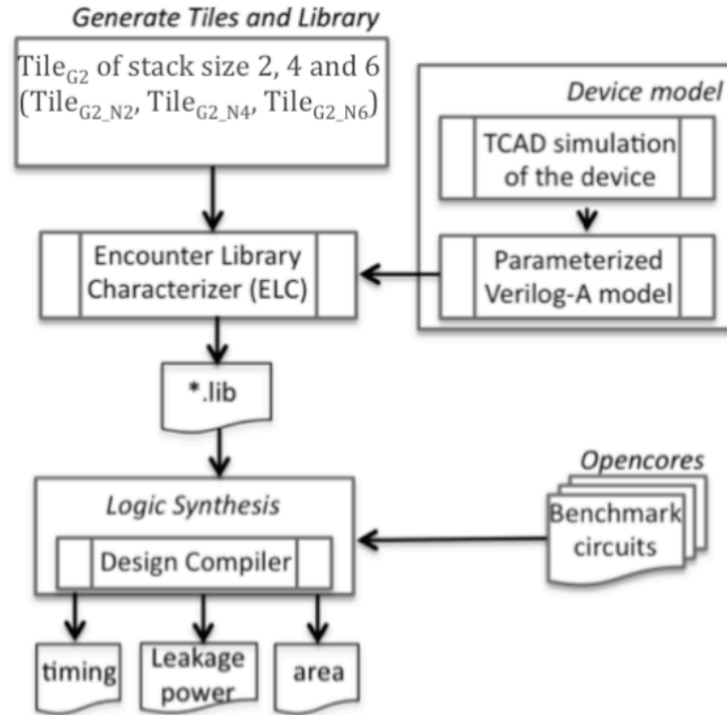


Figure 6.9: Design flow for sizing the tiles.

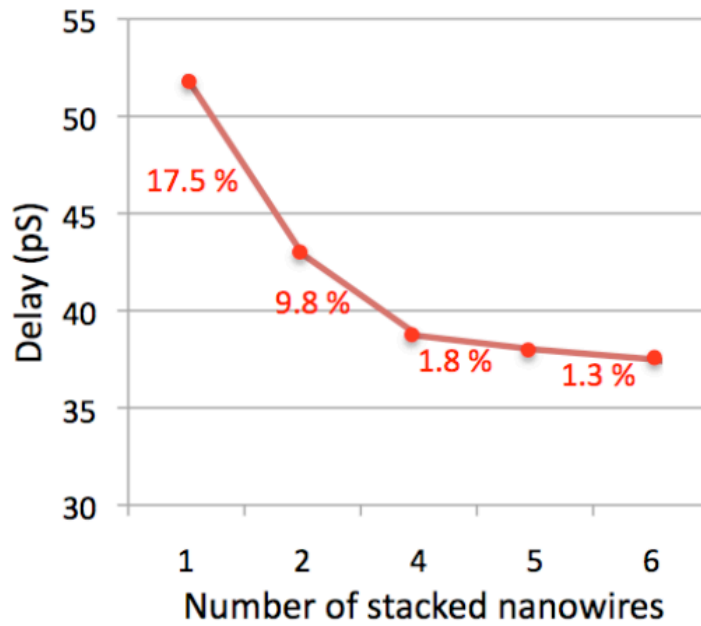


Figure 6.10: Delay characteristics of an Inverter, driving a constant load, with varying stack size.

6.4.2 Tile sizing of an Inverter

First I start with a gate-level simulation by studying the performance metrics of a simple inverter. I size the transistors of the tile by studying the per-

formance of an inverter with varying number of stacked nanowires. Fig. 6.10 illustrates the delay characteristics of an inverter, driving a constant load, with stack size varying from 1 to 6. The width and the length of both the n- and p-type DG-SiNWFETs of the inverter are set to 80nm and 22nm respectively. Though I observe an overall decrease in delay with increase in the stack size, the percentage decrease in delay (shown next to the delay curve in Fig. 6.10) saturates for stack size above 4 nanowires. For instance, I observe 17.5% difference in delay between the stack size of 1 and 2. I note that the percentage improvement in delay is less than 2% for the stack size above 4 nanowires. Ideally, we would like to have a stack size as high as possible, as this leads to overall increase in the I_{ON} of the transistor per unit area. However, increasing the stack size also reflects on the form factor of the stack, thereby inducing variations in the nanowires in the vertical direction. In addition to the technology driven limit on the form factor, which limits the maximum number of stacked nanowires, I study the maximum limit of nanowire stack by carrying circuit-level benchmarking.

6.4.3 Tile sizing with Circuit-level Benchmarking

I study delay, area, and leakage power of various benchmark circuits when mapped with DG-SiNWFET and CMOS technologies at 22nm node. For our analysis, we consider Tile_{G_2} with stack size of 2, 4 and 6 nanowires, and the simulations results linked to each of these libraries are referred to DG-SiNWFET_2X, DG-SiNWFET_4X, and DG-SiNWFET_6X respectively. In order for a fair comparison, I considered SoT architecture for both CMOS and SiNWFET.

Delay:

The performance of all the benchmark circuits when mapped with various libraries is presented in Fig. 6.11. Averaged across all the benchmarks, I observe 1.8x improvement in delay with DG-SiNWFET technology when compared to CMOS. However, the delay improvement varies across different benchmarks based on the application as well as on the logic synthesis tool. For instance, I note a major difference in performance gain when comparing *ethernet* (Eth_top) to *aes encryption* (Aes). In the case of Aes benchmark, I observe 2x improvement in performance from CMOS to SiNWFET technology. Whereas in the case of Eth_top, I observe 1.3x improvement in performance as the interconnect delay plays a dominant role in the overall performance. Comparing the results of technology mapping with SiNWFET technology, I observe minimal improvement in performance by increasing the stack size from 4 nanowires to 6 nanowires. For instance, considering Aes and Wbconmax benchmarks, I observe similar delay characteristics for DG-SiNWFET_X4 and DG-SiNWFET_X6. This observation also agrees with the results from the gate-level simulation carried-out in the previous section.

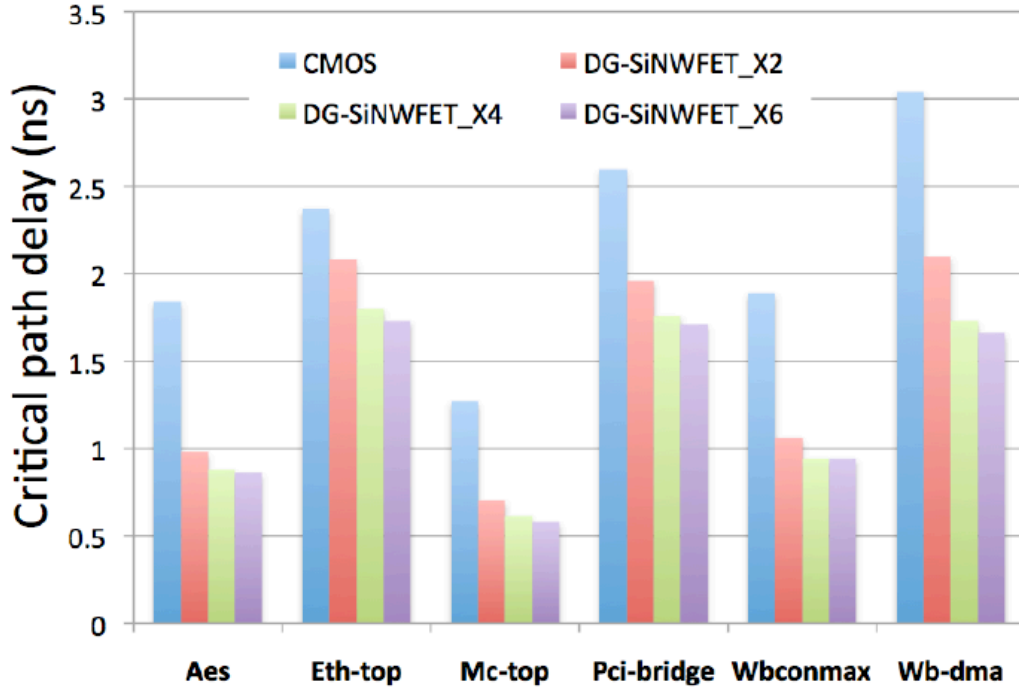


Figure 6.11: Critical path delay of various benchmark circuits when mapped with DGSiNWFET and CMOS technologies.

Area:

The combinational area of the various benchmark circuits after technology mapping with Synopsys design compiler is illustrated in Fig. 6.12. In this study, I considered SoT architecture for both CMOS and SiNWFET. The area of the tile (Tile_{G2}) designed with DG-SiNWFETs is 1.4x higher when compared to its equivalent tile designed with CMOS technology. This is due to the space occupied by the extra polarity gates for each transistor forming the tile, and their respective design rule. The difference in the area of the CMOS and DG-SiNWFET tile directly reflects in the overall area of the synthesized circuit. On an average, I observe 1.7x (70%) increase in combinational area with DGSiNWFET when compared to CMOS technology. Further improvement in area can be envisaged by optimizing SiNW technology. On the other hand, when comparing the area of various DG-SiNWFET implementations, I observe minor reduction in area (less than 1% on an average) as I increase the stack size from 4 to 6 nanowires.

Leakage power:

DG-SiNWFETs are promising contenders for the next generation transistors as they provide better electro-static control over the channel, due to the gate all-around implementation of both the gates. In order to gauge the

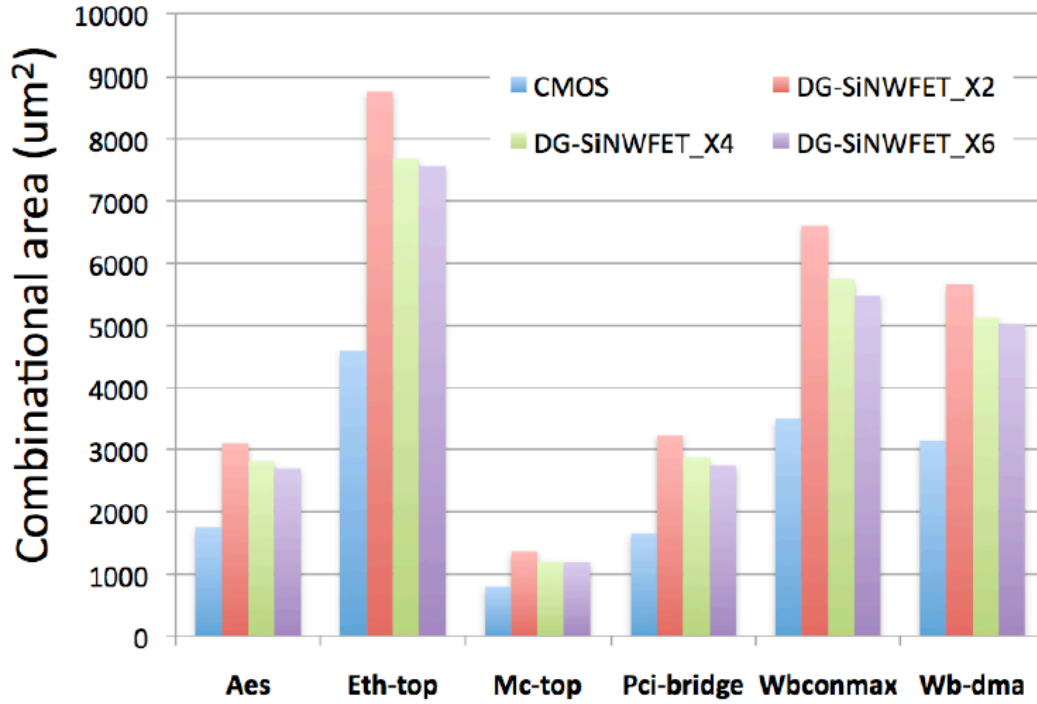


Figure 6.12: Combinational area of various benchmark circuits when mapped with DGSiNWFET and CMOS technologies.

impact of this at the circuit level, I have studied the leakage power for various benchmarks. Fig. 6.13 compares the leakage power of CMOS with all the three implementation of DG-SiNWFETs. We observe drastic reduction (14X) of leakage power with DG-SiNWFETs. When compared to simulation at 45nm node [147], I observe more than 1 order of magnitude reduction in leakage power. This can be attributed to the exponential decrease in the I_{OFF} of the SiNWFET when scaled from 45nm to 22nm node.

Fig. 6.14 compares the leakage power for the various benchmarks mapped with DG-SiNWFET technology. By increasing the stack size, I increase the number of nanowires for each transistor of the tile uniformly, thereby increasing the overall leakage power. For all the benchmarks, I observe a linear increase in the overall leakage power.

6.4.4 Synthesis of Data path Circuits

Datapath circuits are critical in today's ASIC design, as they widely employ XOR/XNOR operations. Double-gate transistors with controllable polarity can be employed to reduce the criticality of datapath circuits, with their unique ability to efficiently implement XOR/XNOR-based logic gates. I showcase here this advantage by synthesizing selected datapath circuits with DG-SiNWFETs technology and we compare them with traditional CMOS

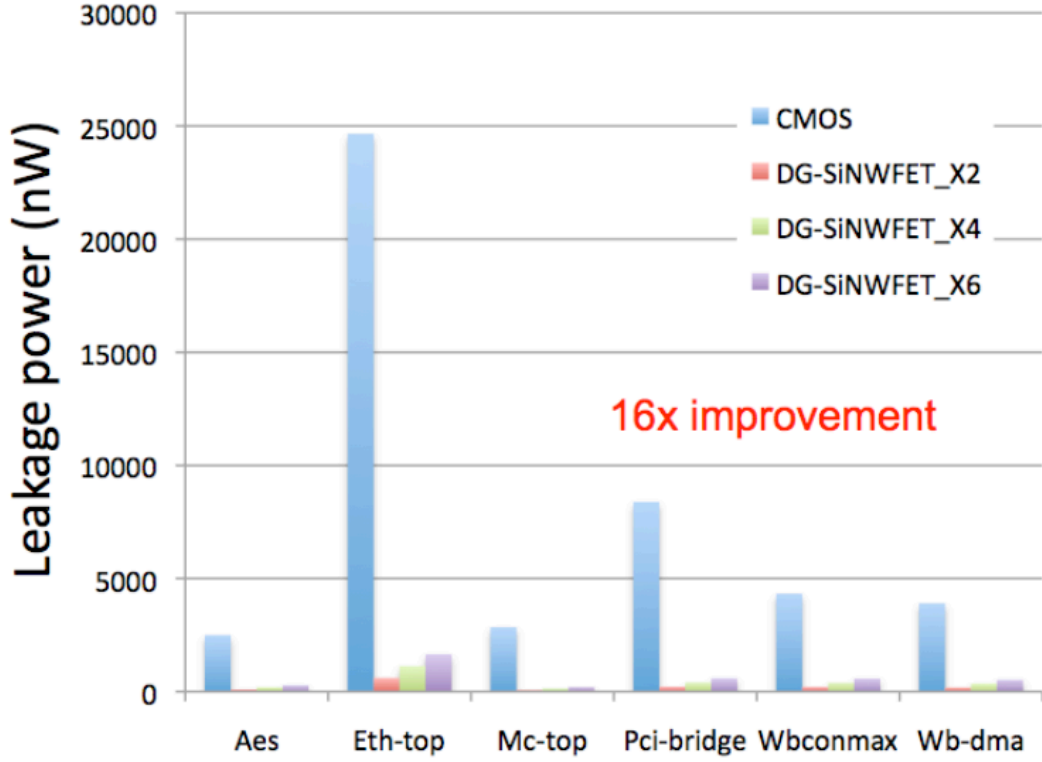


Figure 6.13: Leakage power of various benchmark circuits when mapped with DG-SiNWFET and CMOS technologies.

implementations. The standard cell libraries, for 22nm node, employed in the previous section are considered in this study. The datapath circuits employed to benchmark the advantage of DG-SiNWFETs over CMOS are (i) a Brent Kung adder, (ii) a fast column compressor multiplier, (iii) a square root unit and (iv) a 6 operand multiply and accumulate block. The bit widths considered range from 16 to 64 bits as in real life ASICs. Synopsys Design Compiler is used to synthesize the given datapath benchmarks.

Table 6.5 summarizes the synthesis results. Fig. 6.15 illustrates the normalized area for the considered datapath circuits. On an average, DG-SiNWFET implementations have 17% more area when compared to equivalent CMOS implementation. It has to be noted that the SiNW tile occupies 40% more area when compared to CMOS tile. When compared to the average area increase of 70% for standard benchmarks (see Fig. 6.12), I observe considerable improvement in area for datapath circuits.

The performance improvement, in terms of normalized delay, is presented in Fig. 6.16. I observe, 2.1x improvement (on an average) in delay when compared to CMOS implementations. Clearly I observe better area and delay metrics for datapath circuits. This can be attributed to the efficient

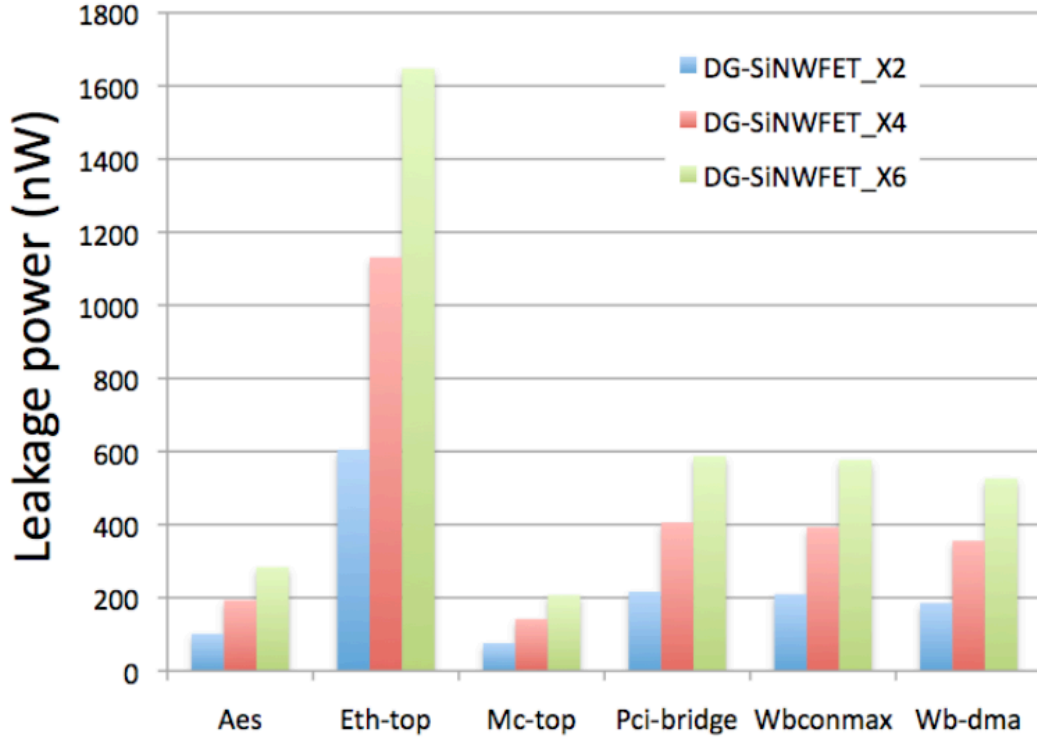


Figure 6.14: Leakage power of various benchmark circuits when mapped with DGSiNWFET with varying stack size.

implementation of XOR/XNOR gates with DG-SiNWFET.

Further improvement in performance can be envisaged by improving the state of the art logic synthesis tools. In this study, I employ commercial logic synthesis tool (Design compiler), which is effective for unate logic functions as it uses AND/OR representations, while XOR operations are partially exploited. Hence, novel logic synthesis tools, which can efficiently manipulate both AND/OR and XOR operations, can fully harness the potential of DG transistors with controllable polarity. Preliminary attempts in [148, 149] highlight the interest for this study.

6.5 Discussion

This chapter introduces a regular layout fabric (called *tiles*) for DG-SiNWFET. A logic tile is essentially an array of prefabricated transistor-pairs grouped together. The motivation in this chapter is to find the basic building block for future ambipolar logic circuits. I study four different logic tiles, which individually form the basic building block for *sea-of-tiles* (SoT) fabric. By running extensive comparisons of mapping standard benchmarks on to

Datapath Circuits	DG-SiNWFET			CMOS		
	Area (μm^2)	Gate Count	Delay (ns)	Area (μm^2)	Gate Count	Delay (ns)
Brent Kung Adder 64 bit	188.26	1049	0.55	167.01	1080	1.24
Column Compressor Multiplier 32 bit	1759.33	7780	1.07	1554.71	7738	2.65
Square Root Unit 32 bit	251.84	1256	12.04	191.23	1115	21.53
MAC 6 operand 16 bit	1186.12	5850	0.98	1082.22	5865	2.75
Average	846.39	3983.75	3.66	748.79	3949.50	7.04

Table 6.5. Synthesis results for selected datapath circuits with DG-SiNWFET and CMOS at 22nm Technology node.

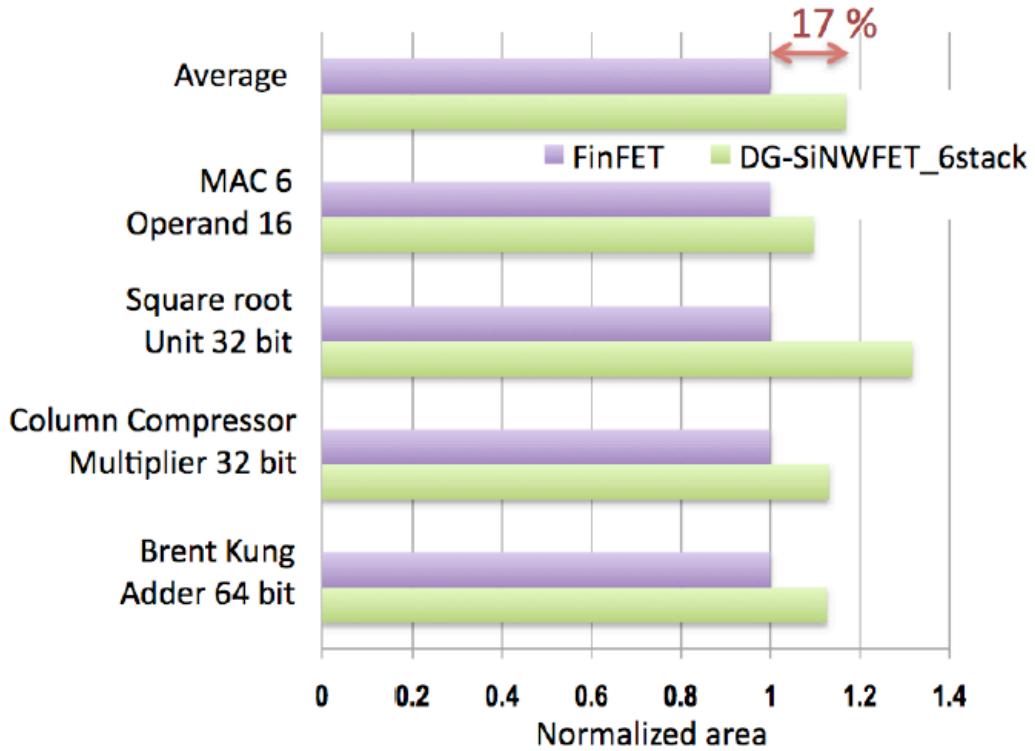


Figure 6.15: Normalized area comparing DG-SiNWFET with stack size of 6 nanowires, with equivalent CMOS implementation.

the SoT fabric I find Tile_{G2} and Tile_{G1h2} to be optimal with respect to area utilization.

In order to study the effectiveness of DG-SiNWFET technology, I carried gate-level simulation to circuit-level benchmarking. At a gate-level, mapping various arithmetic circuits, I obtain 32% average improvement in area and 38% improvement in delay. In this chapter, I evaluate the performance of regular logic tiles for DG-SiNWFETs by varying the number of stacked nanowires. Starting from a TCAD model of DG-SiNWFET, which exhibits p-type and n-type characteristics by controlling the polarity of the second

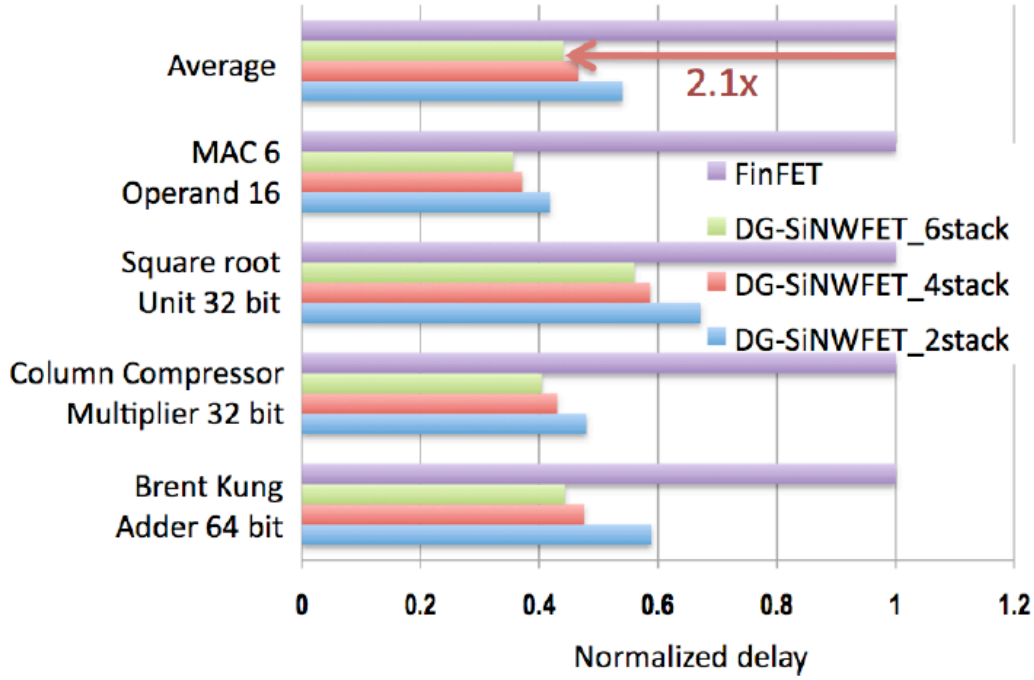


Figure 6.16: Normalized delay comparing DG-SiNWFET, with various stack size, with equivalent CMOS implementation.

gate, I optimize the device performance for achieving a balanced p- and n-type behavior. I show the layout fabric, Tile_{G2} , with 6 vertically stacked nanowires achieves the best performance for a given area constraint. SoT with Tile_{G2} , outperform Si-CMOS, averaged across various benchmark circuits, with 1.8x improvement in delay and 16x improvement in leakage power. For datapath circuits, our simulation results show 2.1x improvement in delay for an area overhead of 17% when compared to CMOS at 22 nm node.

Fabrication of vertically-stacked SiNWFET has many challenges. Technologists have to take into account the variations in the diameter of nanowires placed on top of each other. Increasing the number of stacked nanowires increases variations, hence there is an interest to keep the number of stacked nanowires to a minimal number. On the other hand, increasing the number of nanowires improves the drive current of the SiNWFET. In this study, the device is optimized for performance by varying the number of stacked silicon nanowires as well as the transistor width. Benchmarking at the design level we show best performance for a vertical stack of 6 nanowires. A Tile_{G2} with all the DG-SiNWFETs comprising of 6 vertically-stacked nanowires, provides a good starting point for technologists to realize the SiNWFET and also in studying the diameter variations.

From the design perspective, this chapter adopts the idea of logic

tiles which have been employed to realize semi-custom circuits with SoT architecture. However, it is noteworthy that the logic tiles are inherently reconfigurable. The in-field configurability opens novel opportunities to build reconfigurable logic operators with a very limited amount of transistors [26, 129]. Hence, I envisage using the SoT fabric to efficiently build reconfigurable circuits such as Field Programmable Gate Arrays (FPGAs). However, specific architectural organization should be used in order to keep the wiring complexity minimal, such as in [150] where a matrix arrangement with fixed interconnection pattern was proposed. Such organization can also be extended to semi-custom circuits, with matrices of logic tiles with a reduced wiring complexity between the building gates.

In this study, I employed commercial logic synthesis tool (Design compiler) during the technology-mapping phase with DG-SiNWFET technology. It has to be noted that ambipolar logic gates are efficient in implementing XOR dominated circuits. State-of-the-art logic synthesis tools are effective for unate logic functions, as the Boolean function is decomposed into *And-Inverter* graphs. Hence, I envisage better performance with novel logic synthesis tools specifically designed for XOR dominated circuits. A major aim for future automated synthesis tools is to efficiently manipulate both AND/OR and XOR operations in order to fully harness the potential of novel nanotechnologies featuring transistors with controllable-polarity. Preliminary attempts in [148, 149] highlight the interest for this study.

6.6 Chapter Contribution

This chapter proposes a novel layout fabric, called as logic tiles, for ambipolar logic circuits based on DG-SiNWFETs. With *Sea-of-Tiles* (SoTs) design methodology, I envisage an array of tiles with a constant pitch spread across the chip. Technology mapping with various tiles have been performed in order to find the tile with maximum area efficiency. I show SoT with tiles Tile_{G2} and Tile_{G1h2} , on an average, outperform the one with Tile_{G1} and Tile_{G3} by 16% and 10% in area utilization, respectively. In this chapter, I demonstrate the mapping of 3-input NPN-equivalent function along with various building blocks for ambipolar logic circuits onto SoT of Tile_{G2} and Tile_{G1h2} . From the case studies carried, I conjecture Tile_{G2} to be the fundamental building block for future ambipolar circuits which employ top-gated transistors with two independent gates.

The second part of the chapter studies the sizing of the Tile_{G2} with respect to the number of vertically-stacked SiNWs. The information regarding the number of vertically stacked nanowires is important for technologists in order to optimize the fabrication of the basic building block (layout fabric). Circuit-level benchmarking is performed in order to study the benefits of DG-SiNWFET circuits when compared to CMOS circuits at 22nm node. The performance of DG-SiNWFET with varying number of vertically-stacked SiNWs is extracted from TCAD simulations and introduced into a simple SPICE table model. Different cell libraries are then characterized by electrical simulations. Benchmark circuits are mapped onto SoT to compare the performance (timing, leakage power and area) of logic tiles with CMOS technology at 22 nm technology node. When compared to Si-CMOS, averaged across regular benchmark circuits, I observe 1.8x improvement in delay and 16x decrease in the leakage power with an area overhead of 58%. Finally, I evaluated the performance of datapath circuits, which are dominated by XOR/XNOR gates. Comparing DG-SiNWFET to CMOS at 22nm, I observe 2.1x improvement in delay with an area overhead of 17%.

This chapter of the thesis concludes the second part of this thesis, which deals with design methodologies for silicon nanowire FETs. In the following chapter, robust design techniques for *carbon nanotube FET* (CNFET) based circuits are addressed, and the yield of the CNFET circuits is improved by layout techniques immune to CNT-imperfections and by leveraging inherent properties of CNT technology.

Robust Design Techniques for Carbon Nanotube FET Circuits

7

Foreseeing the trends dictated by Moore's law and anticipating the fundamental limits of CMOS technology in the near future [151, 17], the semiconductor industry is in a quest for a successor technology to CMOS. Among the technologies being considered, *Carbon Nanotube Field Effect Transistors* (CNFETs) appear to be one of the promising successors to MOSFETs due to their superior device characteristics [152, 153, 154]. CNFETs can be classified based on the operation of the device as, Schottky Barrier CNFET, MOSFET-like CNFET, and Band-To-Band-Tunneling CNFET [155]. In this work, I consider top-gated MOSFET-like CNFETs (MOS CNFETs) [153]. For the sake of simplicity, I will refer to MOS CNFETs as just CNFETs from here on.

A representative CNFET structure is shown in Fig. 7.2. Multiple semi-conducting *Single-Walled Carbon Nanotubes* (SWCNTs, or simply CNTs) are grown on or transferred onto a substrate. The CNTs in the device act as transistor channels whose conductivity can be modulated by the gate [156]. The source and the drain regions of CNTs are heavily doped. During the doping process the gate is self-aligned, thereby leaving the CNT region under the gate undoped (intrinsic CNT region). The current carriers in the CNT channel are controlled by the electric field applied to the gate and the type of doping realized on both sides of the un-doped region. The gate, source and drain contacts, and interconnects are defined by conventional lithography, whereas the inter-CNT spacing is not limited by lithography.

CNFET devices fabricated with ideal CNT synthesis can potentially provide more than an order of magnitude benefit in *Energy-Delay Product* (EDP) over Silicon CMOS at 16 nm technology node [157, 158]. Franklin

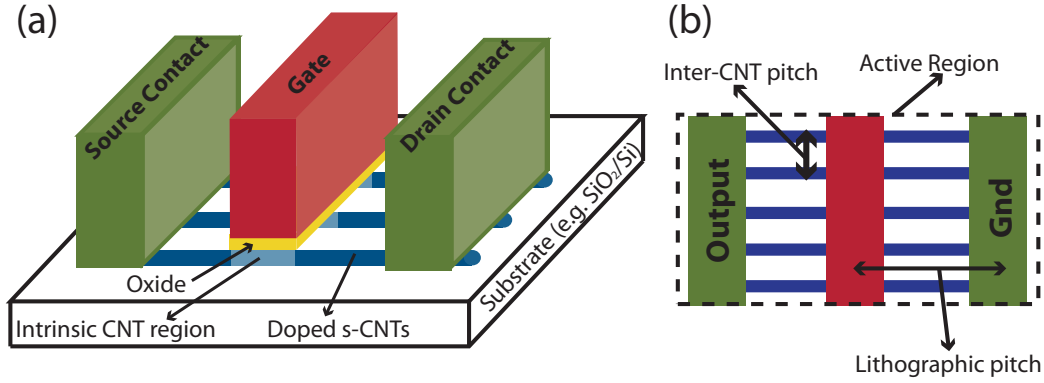


Figure 7.1: (a) CNFET structure. (b) Top view of CNFET.

et al., have demonstrated a sub-10 nm CNFET, which outperforms its competing Si devices by more than four times in terms of normalized current density at low operating voltages of 0.5 V [159], thereby making them ideal for both high performance and low power applications. However, significant challenges in CNT synthesis prevent CNFETs today from achieving such ideal benefits [160]. CNFET technology is expected to have higher variability, as compared to CMOS, because of the following CNT-specific imperfections related to CNT-synthesis: 1. The presence of metallic CNTs (m-CNTs, versus the useful semiconducting or s-CNTs); 2. CNT diameter variations; 3. Mispositioned-CNTs; and 4. CNT density variations.

All of the above imperfections cause variations in the drive currents of CNFETs, which lead to delay variations and/or logic failure. Logic failures can be abstracted as stuck-open and bridging faults. The former case corresponds to having no CNTs, or no continuous CNTs, in a channel region. The latter corresponds to having either m-CNTs in a channel region or mispositioned CNTs. For VLSI circuits with billions of transistors, CNT failures can substantially reduce the overall circuit yield.

In this chapter, I address physical design techniques to minimize failure of CNFET circuits. Based on these techniques, I design a yield-enhanced standard cell library for realizing the complete IC design flow, in order to study the system-level performance of CNFET circuits at advanced technology nodes. Our work contributes in three folds:

- Aligned-active layout technique, which takes into account CNT-correlation to improve the yield of CNFET circuits.
- A novel mispositioned-CNT immune layout technique, which ensures immunity to mispositioned-CNTs.
- System-level benchmarking of CNFET circuits.

This chapter is organized as follows. First, a survey on various CNT challenges and relevant design techniques to handle them is presented. Next, I study the impact of CNT-correlation to improve the yield of the CNFET circuits. Then, various mispositioned-CNT immune layouts styles are presented, followed by designing a yield-enhanced standard cell library. With the help of yield-enhanced cell library, I perform circuit-level benchmarking to compare CNFET technology to CMOS at various technology nodes. Finally, the chapter is concluded by discussing the results and by summarizing the contribution of this part of the thesis.

7.1 Challenges of CNFET Technology

An ideal CNFET device can potentially provide more than an order of magnitude benefit in Energy-Delay Product (EDP) over Silicon CMOS at 16 nm technology node [157, 158]. However, state-of-the-art CNFET technology faces several challenges:

- CNT density:** In order for CNFET device to match the performance of state-of-the-art silicon CMOS, we require high CNT density with more than 100 CNTs/ μm [158]. Current CNT synthesis techniques are far from this target. Improvement in CNT density has been demonstrated using techniques such as multiple-growth [161] or multiple-transfer [162]. In order to achieve optimal energy-delay tradeoffs, CNFETs with CNT density of about 250 CNTs/ μm are needed [156]. Apart from the stringent need to increase the overall CNT density, one has to note the inherent nature of CNT density variations and its effect on circuit performance. CNT density variations are caused by non-uniform spacing between the CNTs on the substrate, which are caused during the CNT growth process using chemical synthesis techniques. This results in CNFETs, though with a fixed width, having a variable number of CNTs. Zhang et. al., have presented a model for characterizing the CNT count in a CNFET as a function of CNFET width (W) and calibrated to experimental data [163].
- metallic CNTs:** A CNT can be either *metallic* (m-CNT) or *semiconducting* (s-CNT), depending on the CNT chirality [164]. Since it is very difficult to precisely control the chirality during CNT synthesis, a CNT can turn out be either m-CNT or s-CNT. State-of-the-art CNT synthesis techniques typically produce 4% to 50% m-CNTs [165, 166]. Presence of m-CNTs lead to CNFET circuit failure as m-CNTs can create source-drain shorts causing excessive leakage and reduced noise margins in CNFET circuits. To tackle the m-CNT challenges, researchers have found out ways to either lower the fraction of m-CNTs during growth

[165, 166], or to remove them before circuits are patterned [167, 168]. Layout techniques such as [169, 170] have also been developed to build CNFETs highly tolerant to m-CNTs even in the absence of m-CNT removal. Though techniques have been developed to remove m-CNTs [171, 167], the process does not guarantee 100% removal of all the m-CNTs, and moreover, it involves removing few of the s-CNTs.

- ***mispositioned* CNTs:** CNT growth techniques are able to produce CNT arrays in which most of the CNTs are aligned in a single direction [172, 173]. However, a small fraction of the CNTs can be misaligned. These misaligned CNTs can cause changes in actual CNT length in the CNFET channel and also introduces CNT-to-CNT junctions. In the extreme cases, poor alignment of CNTs can cause functional failures of logic gates ([174]). In this thesis, I propose special layout technique to ensure immunity to these mispositioned-CNTs [175]. The proposed design technique is compatible with existing CMOS design flow.
- **CNT diameter variations:** CNT diameter variations are also caused by CNT chirality variations, as the diameter of a CNT is a function of its chirality [164]. Current CNT synthesis techniques can produce CNTs with diameters ranging from 0.5 to 3 nm. While this range of variation in CNT diameter can introduce considerable variations in the drive current of the device, few CNT synthesis techniques have reported CNT diameter variation below 10% [176, 173].

All of the above challenges related to CNT-synthesis cause variations in the drive currents of CNFETs, which lead to delay variations and/or logic failure. For example, presence of metallic CNTs lead to device failure, and has to be avoided by all means. For VLSI circuits with billions of transistors, failures caused by these non-idealities can substantially limit the overall circuit yield.

A few previous publications have analyzed the impact of some of these CNT specific variations. Paul *et. al.* compared CNT diameter variations to conventional variations in Si-CMOS such as channel length and oxide thickness, and concludes that CNFETs are less sensitive to conventional variations than to CNT-specific variations (such as CNT diameter variations) [177]. Good electrostatic control and near ballistic transport in CNFETs can significantly minimize the impact of conventional variation sources. It is also suggested that CNT diameter variations can be very important because CNT diameter directly modulates the band gap of a CNT, and therefore affects the threshold voltage of CNFET.

However, for logic circuit applications, single-CNT FETs have their limitations. CNFETs that contain multiple CNTs (multi-CNT FETs) are often used instead because these CNTs can conduct in parallel to provide required drive current. Such configuration is also considerably helpful for reducing variations because multiple CNTs can often behave independently in a CNFET, leading to statistical averaging effects. Previous publications (e.g., [178, 179]) have shown that statistical averaging can significantly alleviate the impact of CNT diameter variations and alignment variations.

Most of these previous analyses assume that a CNFET has a known number of CNTs (CNT count) to start with, and focus on discussing the impact of CNT diameter and alignment variations. However, CNT count can vary substantially in a CNFET. I refer to these variations as CNT count variations. CNT count variations in a CNFET can be caused by both grown CNT density variations, and m-CNT-induced count variations, i.e., the loss of s-CNTs after m-CNT removal.

In this thesis, I focus on two important CNT imperfections: (a) mispositioned-CNTs; (b) CNT count variation. On one hand, mispositioned-CNTs affect the functionality of the transistor, eventually leading to errors. On the other hand, CNT count variation lead to variations in the drive current of the transistors. We introduce CNT correlation in order to minimize delay variations caused by CNT count variation and based on this propose layout technique to improve the overall yield of CNFET circuits.

7.2 CNT correlation

Correlation of CNTs is a very unique feature of CNFET technology. CNFETs are correlated if they share the same CNTs forming their channel region. Fig. 7.2(a, b) shows the top view of complementary logic inverter with p-CNFET (p-type) and n-CNFET (n-type). In Fig. 7.2a, we observe that both the n-CNFET and p-CNFET have the same CNTs forming their channel region. This is referred to as CNFET correlation. On the other hand, in the inverter shown in Fig. 7.2c, the p-CNFET and n-CNFET are uncorrelated as they are formed by different CNTs. In Fig. 7.2(c, d) we illustrate the impact of CNT count variation on the two inverters. For CNFETs which are correlated (Fig. 7.2c), the impact of CNT count variation is uniform on both the CNFETs. On the other hand, for uncorrelated CNFETs (Fig. 7.2d), the drive current of each CNFET is independent to CNT count variation. In the example illustrated, we observe that the inverter in Fig. 7.2d has a stronger p-CNFET. The impact of CNT correlation on CNFET circuits will be studied in the following section.

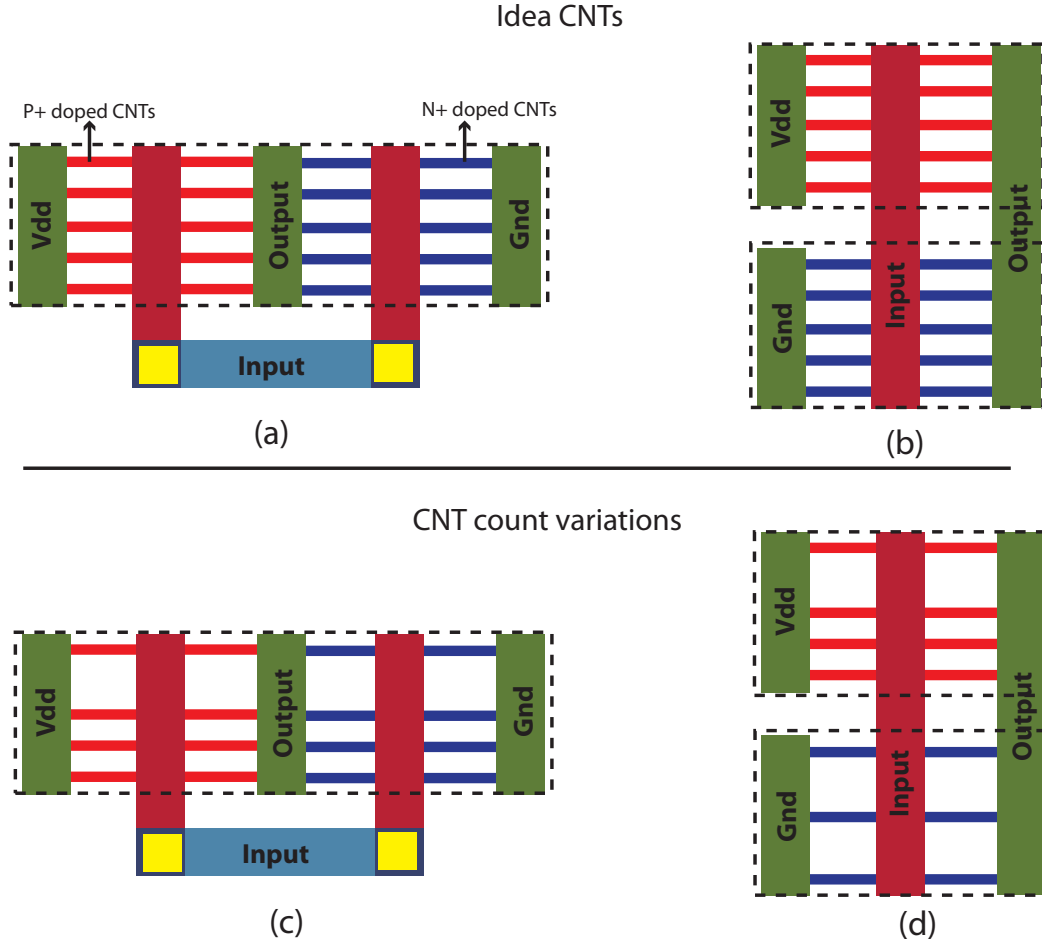


Figure 7.2: (a, c) Top view of an inverter with CNFETs having the same CNTs (referred to as correlated CNFETs); (b, d) Top view of an inverter with un-correlated CNFETs; (a, b) Ideal CNT; (c, d) CNT count variation

7.3 Yield of CNFET with respect to CNT-Correlations

In this section, the impact of CNT correlations on the overall yield of CNFET circuits is studied. The work presented in this section is based on the work of Jie Zhang from Stanford university [180, 181], with whom I collaborated to develop imperfection-immune layouts for CNFET circuits.

7.3.1 Model for CNT Count Limited Yield

CNT count failure can be caused due to m-CNTs, CNT density variations and mispositioned CNTs. The effect of mispositioned CNTs within a CNFET has been found to be very limited [174], especially when the channel length is small or if directional CNT growth is adopted. Therefore, our model focuses

on CNT count failure caused by m-CNTs and CNT density variations.

During CNT growth, assume each CNT has a probability p_m of being metallic and $p_s (=1-p_m)$ being semiconducting. Consider an m-CNT removal process [168], where p_{Rm} stands for the conditional probability of a CNT being removed given it is an m-CNT. For practical VLSI circuit applications, p_{Rm} of greater than 99.99% is required [182]. For most of the discussions in this work, we assume that $p_{Rm} \rightarrow 1$. As a side effect, m-CNT removal processes may also inadvertently remove some fraction of s-CNTs, and the conditional removal probability of a s-CNT is denoted by p_{Rs} . A single CNT can contribute to CNT count failure of a CNFET if it is an m-CNT or if it is an s-CNT but is removed inadvertently. Let p_f stand for this probability, we have

$$p_f = p_m + p_s p_{Rs} \quad (7.1)$$

Consider a CNFET designed with width W , that has $N = N(W)$ CNTs prior to m-CNT removal. In the presence of CNT density variations, $N(W)$ has a statistical distribution, denoted by $\text{Prob}N(W)$. A model for the probability distribution of $N(W)$ as a function of W , and the mean and standard deviation of inter-CNT pitch (denoted by μ_S and σ_S), is presented in [182]. This model is utilized by keeping the σ_S / μ_S ratio as reported in [182]. In our analysis the mean of inter-CNT pitch (μ_S) is assumed to be an optimized value of 4 nm [156].

The probability of CNT count failure (or simply failure probability) of a CNFET is denoted by p_F . Assuming CNT failures are independent of each other, the CNFET fails only if all the $N(W)$ CNTs fail. Applying the law of total probability, p_F is found to be

$$P_F(W) = \sum_{N_i} p_f^{N_i} \text{Prob}\{N(W) = N_i\} \quad (7.2)$$

Figure 7.3 illustrates the relationship of p_F vs. W for different processing conditions. For each case, p_F decreases exponentially with W , as can be seen from equation (7.2). Hence the p_F of a CNFET can be reduced by increasing its width. However, this approach is expensive as it increases the gate parasitics.

7.3.2 Circuit-Level Yield Model

To evaluate yield at the circuit level, a chip consisting of M transistors (CNFETs) that are independent of each other is considered, with W_i representing the width of the i^{th} CNFET. The circuit-level yield is given by

$$\text{Yield} = \prod_{i=1}^M \{1 - p_F(W_i)\} \approx 1 - \sum_{i=1}^M p_F(W_i) \quad (7.3)$$

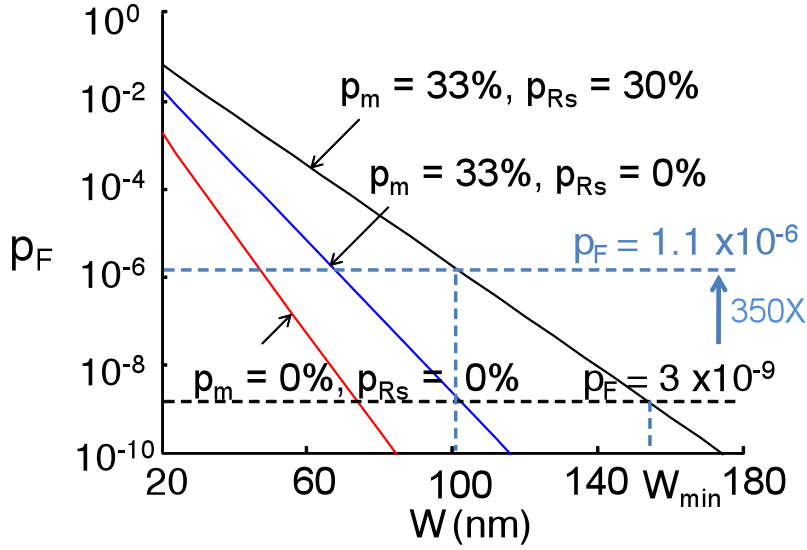


Figure 7.3: CNFET failure probability vs. CNFET width ($p_{Rm} = 1$). [180]

where $p_F(W_i)$ can be found using (7.2) or equivalently from Figure 7.3. Because $p_F(W_i)$ is sensitive to CNFET width W_i , most of the yield loss in (7.3) is due to small-width CNFETs. To optimize an existing circuit design to meet a certain yield, a simple strategy is to increase the sizes of the small-width CNFETs according to a threshold width (W_t). We further define W_{min} as the minimum possible W_t , such that a chip level yield requirement ($Yield_{desired}$) is met. Formally, W_{min} can be found by solving the following optimization problem

$$\begin{aligned}
 & W_{min} = \min(W_t) \\
 & s.t. \text{ Yield} = \prod_{i=1}^M \{1 - p_F(U_{W_t}(W_i))\} \geq Yield_{desired}
 \end{aligned} \tag{7.4}$$

where $U_{W_t}(W_i) = \max(W_i, W_t)$ is an “upsizing” function. Finding the exact optimal solution to (7.4) can be tedious, but the problem can be substantially simplified by neglecting the yield loss in (7.3) due to non-minimum-sized transistors. That is, if there are M_{min} transistors with minimum size (W_t), then problem (7.4) can be re-written as

$$\begin{aligned}
 & W_{min} = \min(W_t) \\
 & s.t. \text{ Yield} = \prod_{i=1}^M \{1 - p_F(W_t)\} \approx 1 - M_{min}p_F(W_t) \geq Yield_{desired}
 \end{aligned} \tag{7.5}$$

The procedure for finding W_{min} according to (7.5) is straightforward: take a device-level p_F vs. W curve such as Figure 7.3, draw a horizontal

line corresponding to $(1 - Yield_{desired})/M_{min}$ and the x-coordinate of the intersection gives W_{min} . Although estimating W_{min} for (7.5) can be iterative in nature, it is simple in practice especially for discrete sizing schemes adopted in standard cell based designs.

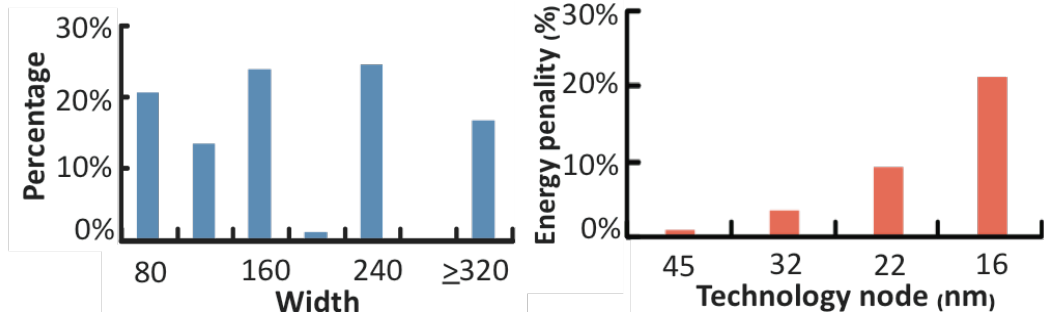


Figure 7.4: Case study: (a) Transistor width distribution of an OpenRISC core synthesized using Nangate 45nm Cell Library. (b) Gate capacitance increase (penalty) vs. technology node associated with upsizing the small transistors to W_{min} . [180]

As a case study, a transistor sizing distribution (shown in Figure 7.4a) extracted from an OpenRISC processor design (cache not included) [98] synthesized with the Nangate 45nm Open Cell Library [88] is considered. M_{min} can be estimated to contain the two left-most bins in Figure 7.4a, which gives 33% of the total number of transistors M . If $M = 100$ million and the desired circuit yield is 99% (assuming $p_{Rs} = 5\%$), the W_{min} in this example is about 155 nm (illustrated in Figure 7.3). This result verifies the initial choice of W_{min} for containing only the first two bins.

Next, area and power penalties associated with upsizing small-width CNFETs is discussed. For standard cell-based designs, there is little area penalty for up-sizing the smallest cells, since there is enough free space available as the distance between the rails is fixed. For example, none of the cells in our library requires an area increase to accommodate the upsized CNFET. Energy and power penalties, on the other hand, are unavoidable due to the capacitance increase. Figure 7.4b shows the energy penalty (%) associated with such upsizing calculated based on the power reports generated by Synopsys Design Compiler. A scaling analysis is also performed for different technology nodes beyond 45 nm by assuming that the CNFET dimensions scale linearly with technology node by 0.7x per generation, while the inter-CNT pitch (μ_S) remains constant at 4 nm. Analysis is not performed beyond the 16 nm node due to the limitations of the CNFET Spice model [183]. Placement of the circuits is performed using Capo [184] and wire parasitics are estimated using FLUTE [185] combined with parameters from [186]. Note that, because the

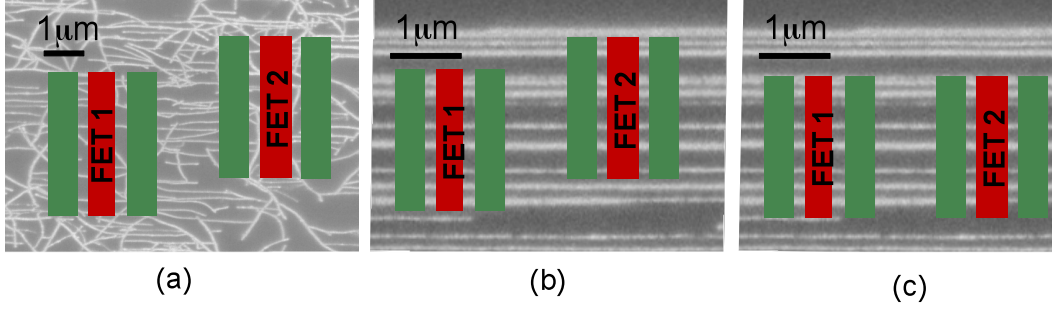


Figure 7.5: a) Non-aligned layout style on uncorrelated CNT growth. (b) Non-aligned layout style on directional CNT growth. (c) Aligned-active layout style on directional CNT growth. [180]

value of W_{min} does not scale with technology, the amount of energy penalty is expected to increase significantly as technology scales down.

7.3.3 CNT Correlation for Enhancing the Yield of CNFET Circuits

The circuit-level yield (and therefore W_{min}) calculation in previous section is based on the assumption that failure probabilities (p_F) of all CNFETs are independent of each other. This assumption is close to reality if the CNFET circuit is fabricated using a growth that produces uncorrelated CNTs (e.g., Figure 7.5a). However, if directional CNT growth (Figure 7.5b) is used, this assumption is overly pessimistic. If two CNFETs have the same size and are aligned in the CNT direction (Figure 7.5c), large correlation can be observed in both CNT count ([187]) and CNT type (i.e., metallic or semiconducting [170]) of the CNTs contained in the two CNFETs. To simplify the analysis, we assume that all CNTs have a fixed length L_{CNT} . Perfect correlation between CNFETs can be achieved if they are spaced within the CNT length, and CNFETs are completely uncorrelated when spaced beyond L_{CNT} .

To find a less pessimistic value of W_{min} for directional CNT growth, we assume that the whole circuit (consisting of W_{min} small-width CNFETs, as defined in Sec. 7.3) is distributed in K_R rows. CNFETs taken from different rows do not share common CNTs and are therefore independent with each other. The yield expression of (7.3) can be re-written as

$$Yield = \prod_{i=1}^{K_R} (1 - p_{RF_i}) \approx 1 - \sum_{i=1}^{K_R} p_{RF_i} = 1 - K_R p_{RF} \quad (7.6)$$

where p_{RF_i} is the failure probability of row i , and p_{RF} is the chip-level average value of the p_{RF_i} 's.

Calculating p_{RF} in a general case (allowing arbitrary positioning of the CNFETs) requires numerical methods. However, we realize that the minimum value for p_{RF} is achieved in the special case where all the minimum-sized CNFET active regions are strictly aligned to each other (as shown in Figure 7.5c). This layout style is defined as aligned-active layout. Because all the CNT counts and types are correlated in this case, the probability of having a failing row is the same as the probability of having one failing CNFET in this row, i.e., $p_{RF} = p_F$. Comparing it with the fully independent case (7.4), the circuit failure probability (i.e., $1 - Yield$) is reduced by W_{min}/K_R times. This ratio of W_{min}/K_R represents the average number of minimum-sized CNFETs in a row, which we denote by M_{min}^R . M_{min}^R is largely determined by L_{CNT} and the average pitch between the small-width CNFETs (denoted by $p_{min-CNFET}$):

$$M_{min}^R = L_{CNT} / p_{min-CNFET} \quad (7.7)$$

Hence, CNT growth with large L_{CNT} or designs with small $p_{min-CNFET}$ are both beneficial to the yield improvement. With the improved yield expression (7.6), the requirement that determines W_{min} in (7.5) can be relaxed by the exact same amount as the reduction in p_{RF} . A much lower W_{min} can therefore be expected.

7.4 Mispositioned-CNT Immune Circuits

Various research groups have shown highly-aligned CNTs by growing them on single-crystal quartz [188] [173]. Nevertheless, a small percentage of CNTs tends to be mispositioned (mispositioned-CNTs). Mispositioned-CNTs affect the functionality of logic gates, by causing CNT short failures. Figure 7.6 shows a 2-input NAND gate with transistors A and B, connected in series in the *pull-down-network* (PDN) and connected in parallel in the *pull-up-network* (PUN). The mispositioned-CNTs in the PUN do not pass under the gate region, hence are completely doped with p+ dopants, thereby creating an unnecessary short circuit between the supply (Vdd) and output (out). CNT short failures rise with the increase in distance between the gates (A and B in our example), leading to undesired logic errors.

A design technique, called mispositioned-CNT immune layout, to handle the errors caused by mispositioned-CNTs was presented in [174], where etched regions are realized to avoid unnecessary short circuits. Figure 7.7 illustrates an example of an *And-Or-Inv* (AOI21). Etched region is introduced between the gates A and B in the pull-up network (PUN), thereby breaking the CNTs that are not aligned to the gate.

A general rule of thumb for mapping a generic schematic to mispositioned-immune layout presented by [174] (shown in Figure 7.7) is given below:

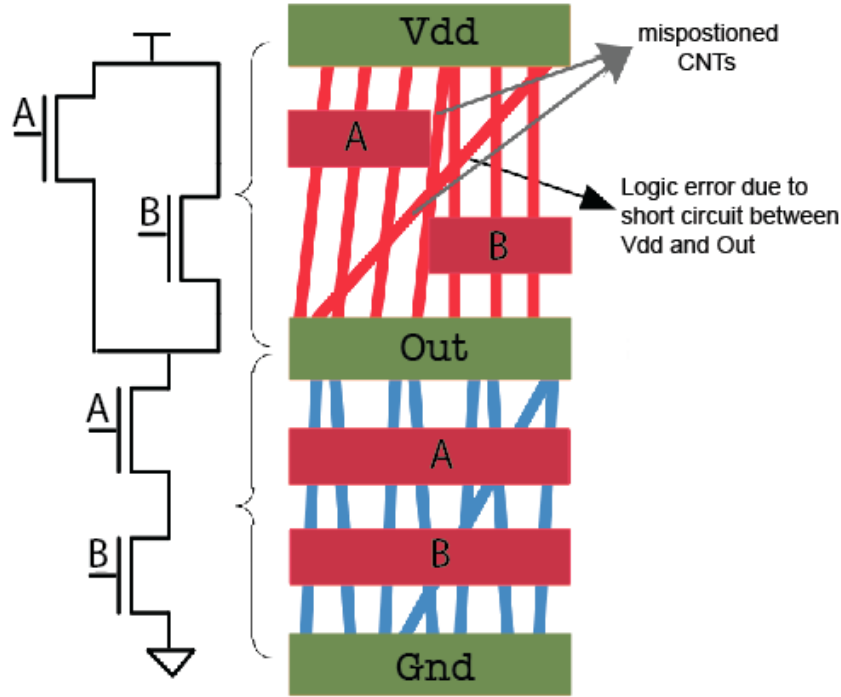


Figure 7.6: Logic errors caused by mispositioned-CNTs. [174]

- A node is mapped to a metal contact. In Figure 7.7, nodes Vdd, x, Out and Gnd are realized with a metal contact in green.
- CNTs between the parallel transistors are etched away. In the example shown in Figure 7.7, etched regions are introduced between transistors A and B in the PUN and transistors A-B and C in the PDN.
- Transistors in series have the same CNTs running under the gate region, hence not affected by mispositioned-CNTs.

7.4.1 Layout Technique Based on Euler Paths

In this work, I propose mispositioned-CNT immune layout techniques based on Euler paths [124] [189], which is similar to traditional CMOS layouts. In order to realize an Euler path for the circuit, a logic graph of the schematic is first constructed, where each transistor is represented by an edge in the graph and the connection between the transistors is denoted by a node. Euler path is defined as a trail in the graph which traverses every edge only once. The path is defined by the order of each transistor name.

Figure 7.8 illustrates the two possible layout schemes of an AOI21 gate. In Figure 7.8a, one Euler path is realized for each PUN and PDN. This technique is reminiscent to the existing CMOS layouts, where Euler paths are chosen

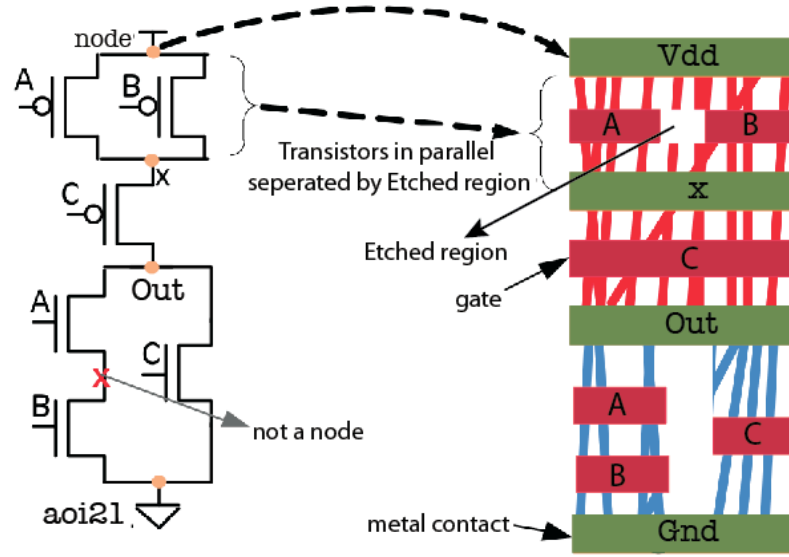


Figure 7.7: Mispositioned-CNT immune layout [174].

with similar transistor ordering [189]. With optimal transistor ordering, the intra-cell routing complexity is minimized, thereby leading to regular layouts. Figure 7.8a shows the layout of the AOI21. Since the PUN and PDN are realized with Euler paths, we can observe that the layout of networks is immune to mispositioned-CNTs. The CNTs between the PUN and PDN are etched away in order to avoid the logic errors caused by mispositioned-CNTs.

In an alternative approach, a simple layout can be realized by drawing one Euler path for the entire circuit covering both the PUN and PDN (see Figure 7.8b). I observe that mispositioned-CNTs have no effect on this layout style, as at any given point CNTs are either connected to a metal contact (node) or passing under a gate. As a further step, the sizes of the transistors can be varied for balancing the drive strength of the PUN and the PDN.

In the case of layout scheme presented in Figure 7.8a, conventional CMOS layout techniques can be applied for generating the layout. On the other hand, for the novel layout scheme in Figure 7.8b, the following heuristic can be applied for mapping a generic schematic.

- Create a graph of the circuit, with the contacts mapped as nodes and gates as edges, which connect the nodes.
- Draw an Euler path traversing all the nodes and edges of the entire network (no concept of PUN and PDN). Since I consider only one Euler path, transistor ordering need not be taken into account. In case if an Euler path does not exist, a break in the active region can be realized. However, this will not affect the immunity towards mispositioned-CNTs.

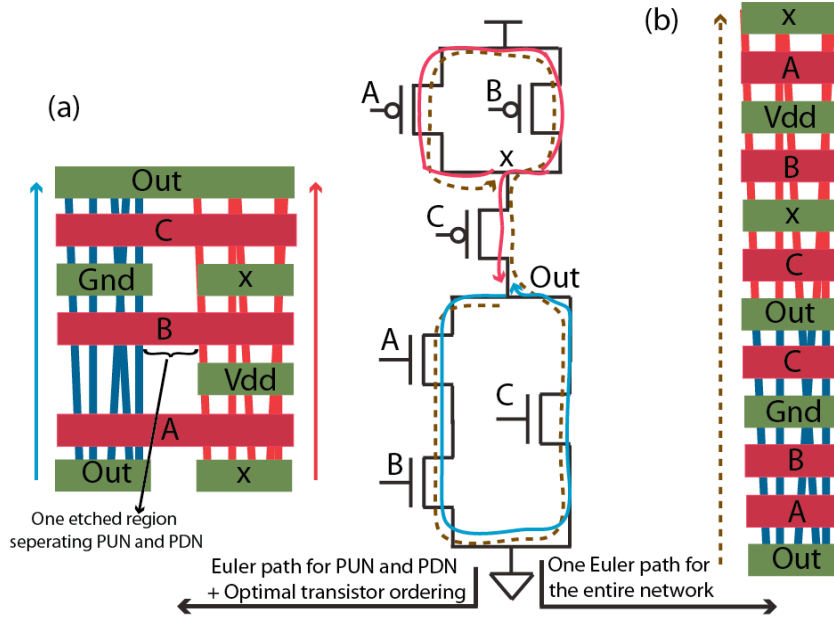


Figure 7.8: Misaligned-CNT-immune layout based on Euler paths. (a) One Euler path for each PUN (red line) and PDN (blue line). (b) One Euler path for the entire schematic (dotted line).

7.4.2 Mispositioned-CNT Immune Layouts with respect to CNT Correlation

From Section 7.2 I infer that yield of circuit is improved by maximizing the correlation between the transistors. Hence maximum yield is obtained by correlating all the CNFETs comprising the circuit. All the three layout techniques presented in Section 7.3 are analyzed here with respect to CNT correlation and cell routing complexity. This gives us the optimal choice of layout for designing the standard cell library.

As an example, I study a 3-input NAND gate realized with all the three layout styles. The schematic of the NAND gate is shown in Figure 7.9a. In Scheme-1, etched regions are realized between the transistors in parallel [174]. In Figure 7.9b two etched regions are realized between gates A and B and gates B and C of the PUN. Transistors in the PDN are in series, hence not affected by mispositioned-CNTs. It can be observed that all transistors in the PDN are correlated as they have the same CNTs forming the channel for all the three transistors. However, in the PUN all the three transistors are uncorrelated, thereby affecting the yield of the logic gate (see Section 7.2).

Scheme-2 and Scheme-3, shown in Fig. 7.9, are obtained by employing the layout technique presented in Section 7.4.1. The layout style presented in Scheme-2 (Figure 7.9c) is similar to the standard cell CMOS style layouts

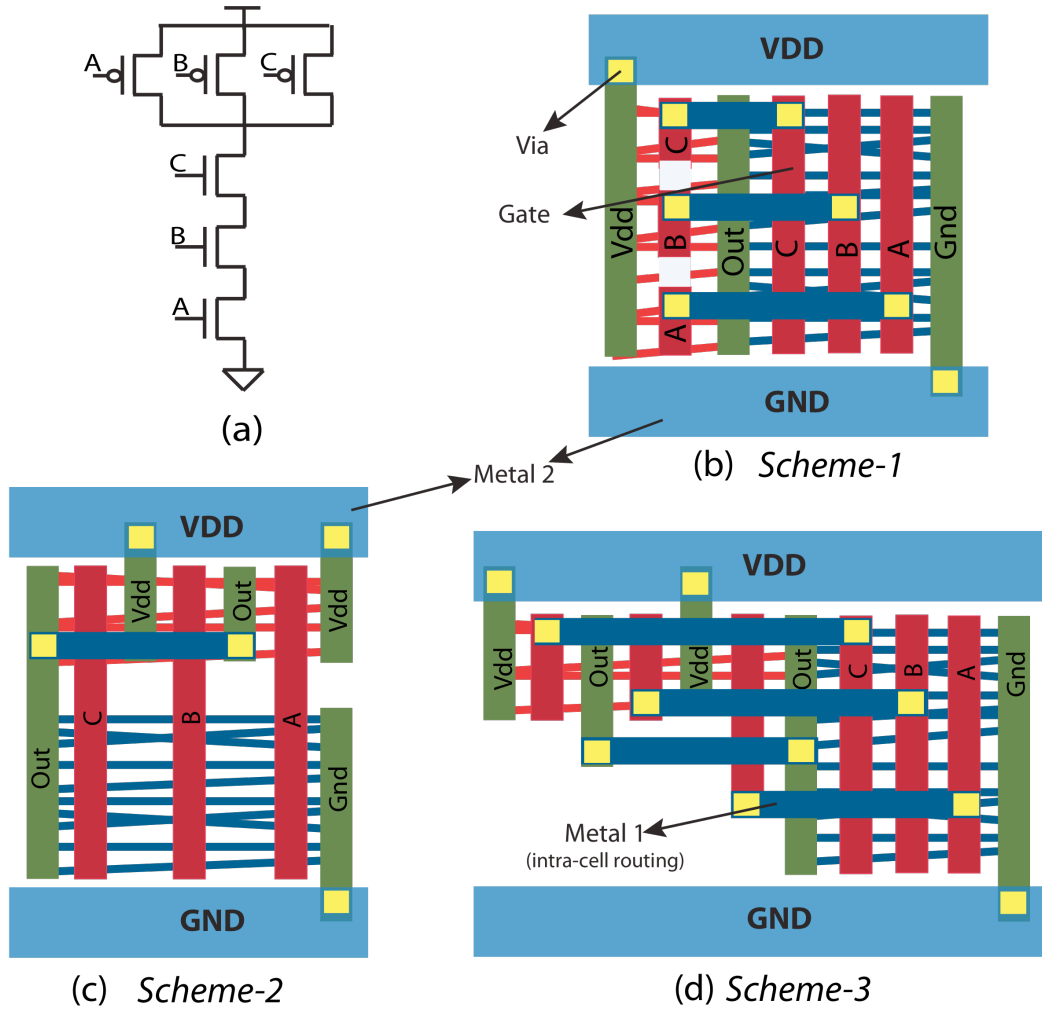


Figure 7.9: NAND3 gate. (a) Circuit schematic. (b, c, d) Mispositioned-CNT-immune layouts.

[124]. The PUN and PDN are separated by an etched region, thereby gaining immunity to mispositioned-CNTs. We can observe from Figure 7.9c that all the CNFETs in the PUN (and also in the PDN) are correlated. However, the PUN and PDN are not correlated. Hence any logic gate can be realized with just two aligned-active grids. The main advantage of this layout scheme is the intra-cell routing.

On the other hand, Scheme-3 is the ideal layout style for correlating all the transistors of the network. Realizing a layout with only one Euler path inherently makes all the transistors correlated. Hence we can obtain maximum yield with Scheme-3 layout style. However, the intra-cell routing to connect the gates in the PUN and PDN is complex, as an extra metal layer is needed. Hence, the regularity at the gate stack is compromised for making all the transistors correlated. Moreover, the layouts tend to be wider and shorter

GATES	SCHEME 1				SCHEME 2				SCHEME 3			
	AREA*	# VIAS	ICRA*	# AAG	AREA*	# VIAS	ICRA*	# AAG	AREA*	# VIAS	ICRA*	# AAG
INV	168	4	33	1	216	2	6	2	168	4	33	1
NAND2	378	6	78	2	546	3	42	2	544	8	105	1
NAND3	812	8	135	3	1020	5	63	2	1128	11	279	1
NAND4	1254	10	204	4	1638	7	100	2	1920	15	510	1
AOI12	1156	8	180	2	1020	7	161	2	832	11	390	1

Area* – Area of the standard cell in λ^2
Vias – Number of Vias

ICRA* – Intra-cell routing area (Metal 1 routing area in λ^2)
AAG – Number of *aligned-active* grids

Table 7.1: Area, Routing complexity (minimum # Vias and ICRA), and AAGs for Mispositioned-CNT immune layout schemes.

as we place the PUN next to the PDN [175].

Table 7.1 reports various performance metrics (cell routing complexity, active area and the number of *Aligned-Active Grids* (AAG)) of the three layout schemes applied for various logic gates. In order to avoid technology dependency, I employ λ -based rules [186] for calculating the area of the cell and *Intra-Cell Routing Area* (ICRA). The length of the transistor is set to 2λ with the minimum transistor width of 8λ .

Transistor sizing is taken into account for all the gates. The number of AAGs varies for Scheme-1 based on the function and fan-in of the logic gate. Intra-cell routing complexity for Scheme-1 increases for complex gates and gates with high fan-in. In the case of Scheme-2, the intra-cell routing is simplified with the minimum number of AAGs set to two. On the other hand, for Scheme-3, extra resources in terms of cell area and intra-cell routing is needed for achieving minimum number of AAGs. Among the three layout schemes presented, I observe that Scheme-2 is preferable when considering all the performance metrics (cell area, intra-cell routing area, and the number of AAGs).

7.5 Yield Enhanced CNFET Cell Library

Layout techniques to improve the yield of CNFET circuits (presented in Sections 7.3 and 7.4) are employed here to realize the desired standard cell library. From Section 7.3, I infer that Scheme-2 is an ideal choice for realizing standard cells due to its simplified intra-cell routing as well as ease in aligning the critical transistors of the PUN and PDN. However, in order to improve the overall yield of the CNFET circuit, the aligned-active layout style requires the active regions not only within each individual cell, but also between different cells to be aligned to each other.

Hence, for designing a new standard cell library an aligned-active technique is added to the new set of design-rules. The aligned-active design rule ensures that an active-area grid is virtually marked where all the transistors of the PUN (and also of the PDN) are aligned. In this section, I study the impact on cell area and gate capacitance by applying the aligned-active design rule on existing libraries. I applied the aligned-active restriction to an existing standard cell library [88] by the following heuristic:

- Estimate W_{min} according to Equations (7.5) and (7.6).
- Find active regions corresponding to all the CNFETs with width smaller than W_{min} and perform upsizing. These active regions are called critical active regions.
- Place the n-type (same for p-type) critical active regions of all cells in the cell library in such a way that their y-coordinates match with each other.
- Modify the intra-cell routing as necessary.

Note that, although non-critical active regions have not been explicitly mentioned in the above heuristics, it is still beneficial to align them with the critical active regions as much as possible.

The standard cells in the Nangate Open Cell Library [88] were modified according to the aforementioned procedure for the enforcement of aligned-active restriction. Figure 7.10 illustrates one of the standard cells (AOI222_X1) before (a) and after (b) enforcing this restriction. An example of DFFS_X2 flipflop with aligned-active restriction is shown in Fig. 7.11. The critical n-type active regions in this cell are highlighted in dashed yellow lines. After the modification, all the n-type active regions in the cell are aligned according to a globally defined grid. The cell width has increased by 9% as a result of this change.

I now discuss the area costs of strictly aligning the critical transistors ($W < W_{min}$) of an existing CMOS standard cell libraries. Altering the positions of active regions in the critical cells will have an impact on the intra-cell as well as inter-cell routing. However, in order to minimize the penalty on inter-cell routing, I retained the location of the I/O pins as much as possible while modifying the cells.

Aligning to the optimal grid has an area impact on 4 cells (out of a total of 134 cells) from the Nangate Open Cell Library, including AOI222_X1 and DFFS_X2 cells shown in Fig. 7.10 and Fig. 7.11. I have further extended my analysis to a commercial 65 nm standard cell library, having 775 cells.

About 20% of the library cells have an impact on area while aligning the active regions. Table 7.2 presents the area penalty on standard cell libraries for enforcing aligned-active layout style. Overall I observe that aligning active regions becomes complex for gates with high fan-in as well as flip-flops and latches, thereby leading to area penalty. However, the area penalty can be minimized by increasing the number of aligned-active regions of the standard cells. For example by doubling number of AAR for both p-type and n-type CNFETs, instead of one, results in zero area penalty. However, the p_{RF} benefit is reduced by 2x (described in Section 7.2), which corresponds to less than 5% increase in W_{min} .

7.6 System-Level Benchmarking

In this section, I perform system level evaluation of CNFET circuits. Physical design techniques presented in the previous sections are employed to design a yield-enhanced standard cell library, based on which I synthesize various benchmark circuits. Comparison with CMOS circuits is drawn considering area, delay, and power at various technology nodes. A snapshot on the overall design methodology considered in this work is presented in Fig. 7.12. The proposed flow starts from CNT synthesis and leads to complete IC design flow. A brief overview of the design methodology is explained below:

- **CNT Synthesis:** Carbon nanotubes are grown using chemical synthesis and the exact positioning and chirality of CNTs is very

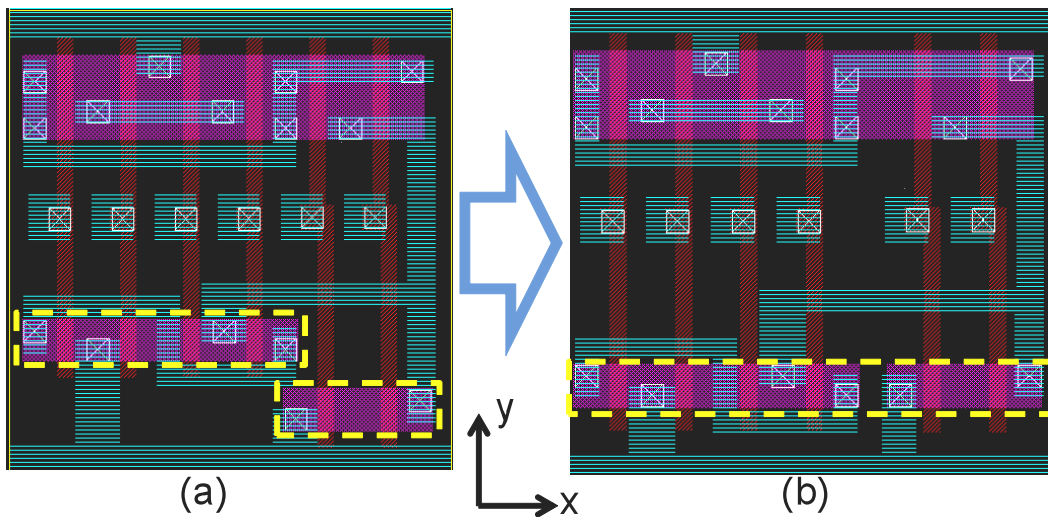


Figure 7.10: Enforcing aligned-active layout style to the AOI222_X1 cell from the Nangate 45nm Open Cell Library.

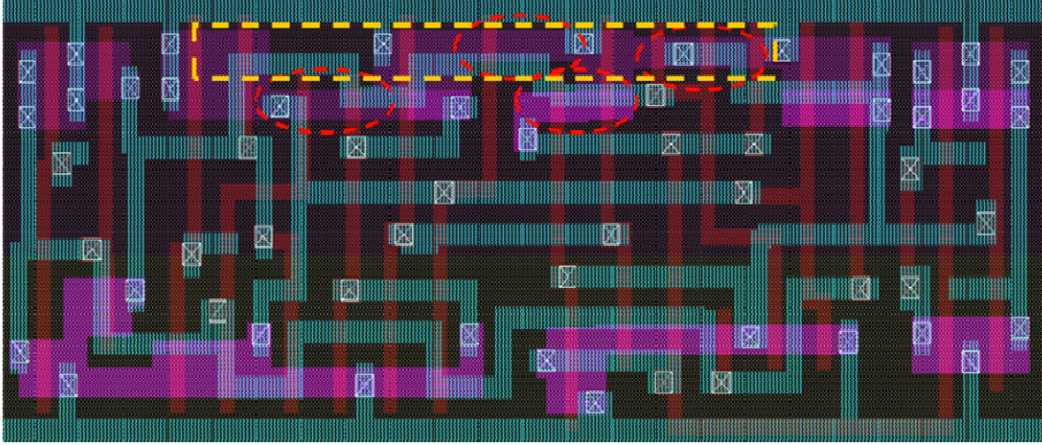


Figure 7.11: Enforcing aligned-active layout style to the DFFS_X2 cell from the Nangate 45nm Open Cell Library.

	65nm commercial library		45nm Nangate Open Cell Library
	one aligned active region	two aligned active regions	
# std. cells	775	775	134
Cells with area penalty	~ 20%	0	4
Min penalty	10%	0%	8%
Max penalty	70%	0%	8%

Table 7.2: Area penalty on standard cell libraries for enforcing aligned-active layout style.

difficult to control. As a result, we have a mixture of semiconducting and metallic CNTs (5% to 50% m-CNTs). By removing the m-CNTs [168], we are remained with s-CNTs for building reliable CNFET circuits.

- **CNT correlation:** Correlation of CNTs is a very unique feature of CNFET technology. CNFETs are correlated if they are aligned in the CNT direction, i.e., CNFETs with similar CNTs forming their channel region. For example in Figure 7.12, large correlation can be observed in both CNT count [187]) and CNT type (i.e., metallic or semiconducting [170]) for CNFET-1 and CNFET-2. On the other hand, CNFETs are uncorrelated if they have different CNTs forming their channel (e.g., CNFET-2 and CNFET-3). In Section 7.2, we quantitatively showed how CNTs correlation can improve the yield

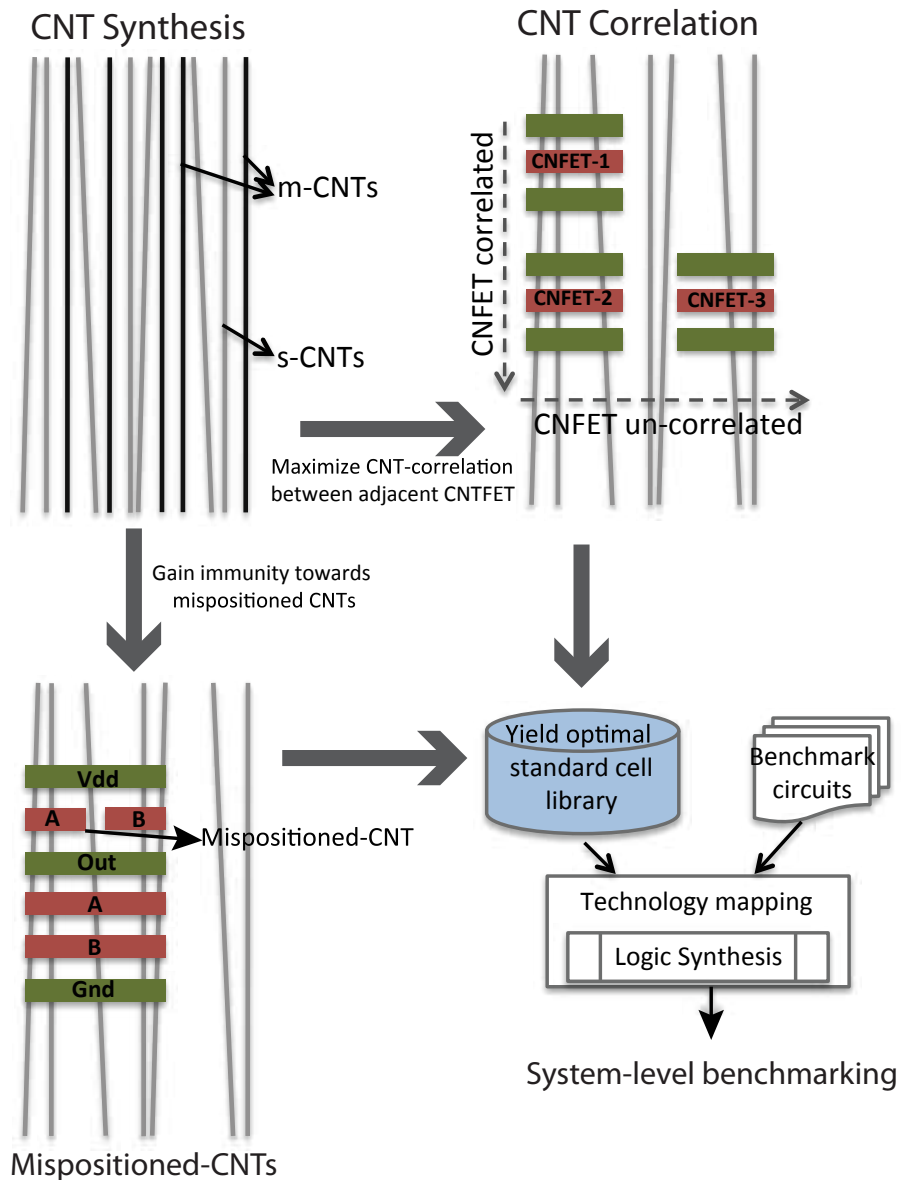


Figure 7.12: Design methodology.

of the circuit. The key idea is to maximize correlation between CNFETs.

- **Mispositioned CNTs:** During the CNT fabrication process, some CNTs are mispositioned due to the lack of control on CNTs position. As discussed in Section. 7.4, mispositioned-CNTs can cause logic failures. With the help of mispositioned-CNT immune layout techniques, these logic errors are avoided, thereby improving the yield of CNFET circuits.
- **IC design flow with yield-enhanced standard cell library:** By

maximizing CNT correlation and with the help of mispositioned-CNT immune layout technique a standard cell library is designed in order to improve the overall yield of CNFET circuits (see Section. 7.5). By incorporating the yield-enhanced cell library in the overall IC design flow, I study the system level performance of CNFET circuit at various technology nodes (32 nm, 22 nm, and 16 nm).

7.6.1 Experimental Setup

A detailed design flow for system-level benchmarking of CNFET circuits is presented in Fig. 7.13. For the design of yield-enhanced CNFET cell library, CNT synthesis parameters like (p_{Rm} , p_{Rs} , and p_m) are needed in order to find the minimum width of the transistor. For improving the yield of CNFET circuits, I employ both the aligned-active and mispositioned-immune layout styles to design various standard cells. The set of standard cells consists of 32 combinational logic cells such as NAND2, NAND3, NOR2, AOI21, ... and a D flip-flop with asynchronous reset and preset. Electrical characterization of standard cells is done with Encounter Characterizer tool [93] using the Stanford's CNFET compact model [190]. To enable our performance evaluation of CNFET circuits, I generate libraries (*.lib files) for various technology nodes (32 nm, 22 nm and 16 nm) at a nominal voltage of 0.8 V. CMOS counterpart libraries have been generated using PTM models [137]. The gate sizing respects the Nangate library [88] sizing and ideal transistor scaling have been applied for both logic and memory scaling between the different technology nodes. In addition to the gate characterization, a simple and ideally scaled model of the wire load is considered.

I then use a set of logic circuits taken from the OpenCores repository [98]. These benchmarks illustrate various applicative constraints from simple gate dominated circuits (e.g. memory controller) and interconnection dominated circuits (e.g ethernet) to complex blocks (OpenRISC processor). Synopsys Design Compiler [97] does the synthesis of these circuits. Timing, power and area reports are considered to evaluate the impact of CNFET implementation when compared to CMOS counterparts.

7.6.2 Results and Discussion

CNFET vs. CMOS at various technology nodes

In this section, I study the critical path delay and dynamic power of CNFET and CMOS circuits. For a fair comparison, each benchmark is constrained with the same clock frequency for CMOS and CNFET at each technology node. The clock frequency is set to the maximum frequency achieved by

Benchmarks	Nodes	Clock	Critical path delay (ns)		Dynamic power (mW)	
			CMOS	CNFET	CMOS	CNFET
mem_ctrl # Cell = 26K # FF = 194	32 nm	0.34	0.34	0.2	8.88	8.7
	22 nm	0.32	0.32	0.13	8.61	2.9
	16 nm	0.3	0.3	0.1	6.7	2.01
eth # Cell = 51K # FF = 10K	32 nm	0.37	0.37	0.33	77.6	72.2
	22 nm	0.33	0.33	0.24	64.5	23.97
	16 nm	0.35	0.35	0.24	49.33	15.31
wb_conmax # Cell = 6279 # FF = 773	32 nm	0.43	0.43	0.14	8.78	7.66
	22 nm	0.41	0.41	0.1	6.66	3.55
	16 nm	0.37	0.37	0.09	5.9	2.73
wb_dma # Cell = 24K # FF = 578	32 nm	0.46	0.46	0.2	5.14	5.07
	22 nm	0.41	0.41	0.15	4.63	1.51
	16 nm	0.34	0.34	0.13	4.19	1.19

mem_ctrl : Memory Controller **eth** : Ethernet IP core # Cells = number of logic cells
wb_conmax : Wishbone IP core **wb_dma** : Wishbone DMA IP core # FF = number of flip-flops

Table 7.3: Critical path delay and Dynamic power at various process nodes for CMOS and CNFET technologies.

with CNFETs when compared to CMOS. Dynamic power reported in Table 7.3 includes both the internal power and the net switching power. In our simulations, I observe decreasing trend in dynamic power with scaling (from 32 nm to 16 nm) for both the technologies. Maximum reduction in dynamic power is achieved at lower technology nodes (22 nm and 16 nm) for all the benchmarks. Averaged across all the benchmarks and nodes, I observe 2.3x improvement in dynamic power with CNFET technology over CMOS technology for the same frequency of operation.

Energy-delay-product (EDP) is an attractive metric to compare designs, as one can trade increased delay for lower energy per operation (e.g., by scaling down the supply voltage, we can trade the increase in delay with the decrease in overall energy consumption). Fig. 7.16 shows the improvement in EDP with CNFET gates when compared to CMOS. I observe maximum EDP gains for gate-dominated circuits ranging from 2x to 8x. Averaged across all the benchmarks, CNFET circuits show 5.7x improvement in EDP when compared to CMOS circuits.

Maximum performance

In the previous section, I studied the CMOS circuits and CNFET circuits operating at the same frequency. I observed that CNFET circuits meet the CMOS delay requirements with ease, due to the superior device characteristics of CNFETs. Here, I study the maximum frequency achieved by benchmark

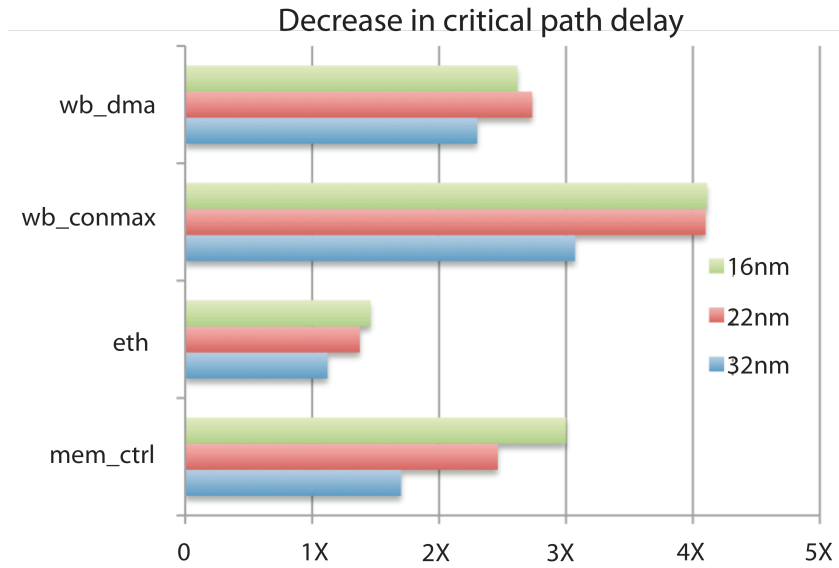


Figure 7.14: Critical path delay improvement of CNFET circuits when compared to CMOS circuits.

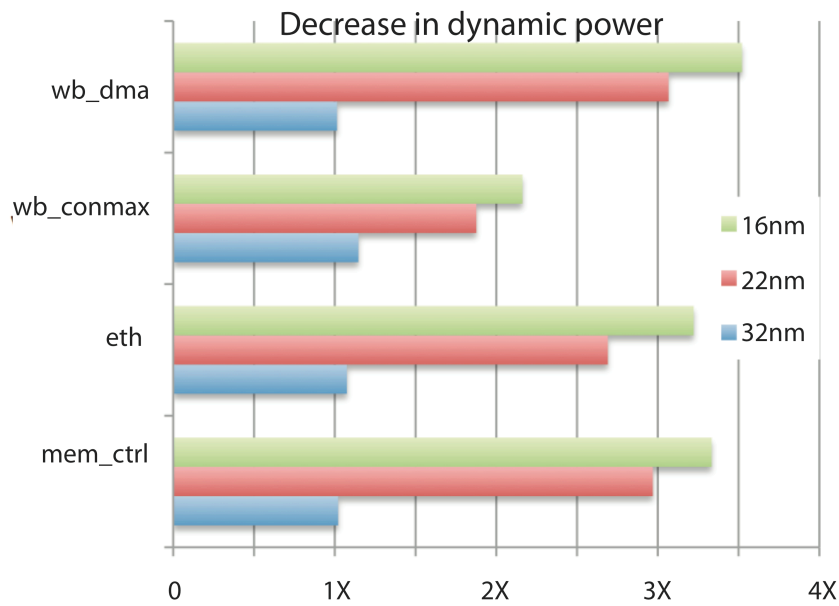


Figure 7.15: Dynamic power improvement of CNFET circuits when compared to CMOS circuits.

circuits with CNFETs. This study will shed some light on the impact of CNFET technology on high performance computational blocks, which are desired to operate at maximum possible frequencies. In order to maximize the performance, I synthesized the benchmarks with very low delay constraints. Fig. 7.17 shows the maximum frequency improvement for various benchmark circuits. At 16nm I observe a maximum frequency gain of 8.5x, averaged

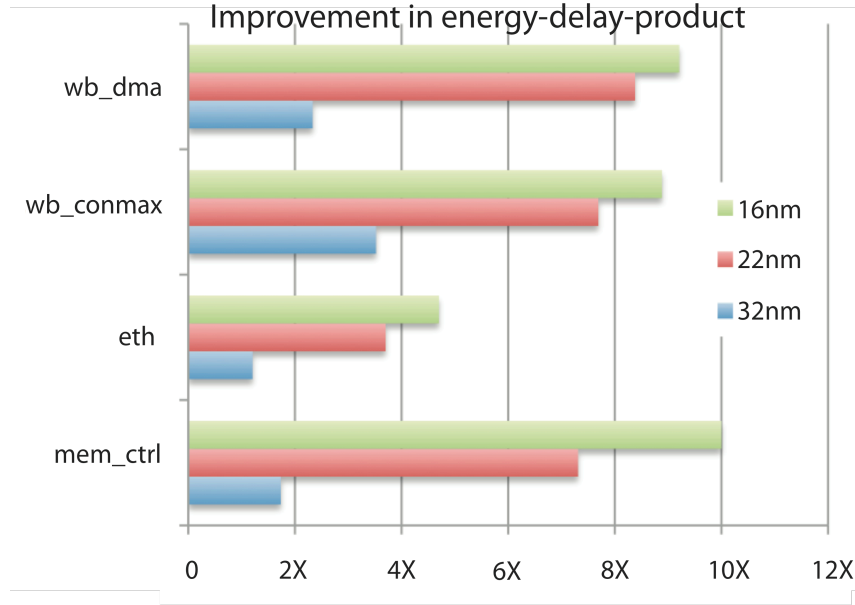


Figure 7.16: EDP of various benchmarks with CNFETs when compared to their implementation with planar CMOS technology.

across all the benchmark circuits. However, it has to be noted that the dynamic power increases with frequency. The increase in power can be kept under control by applying voltage scaling design technique. An optimal operating condition can be found by applying low power design techniques, considering various frequencies and supply voltages, in our system-level simulation framework. Finding the optimal voltage for each benchmark is beyond the scope of this work.

Case study: OpenRISC processor

In the previous two sections, I evaluated the performance improvement of various benchmarks circuits (gate-dominated as well as interconnect-dominated). Here, I study the impact of CNFET technology at a higher abstraction by mapping an OpenRISC processor with CNFET technology. I synthesized the OpenRISC 1200 processor [98] at various lithography nodes for both CMOS and CNFET technologies. Our main motive is to find the maximum frequency achievable at each of the technology nodes. Fig. 7.18a depicts the maximum frequency of an OpenRISC core for each node. Large memory banks have been used for the different nodes. I evaluated the performance by assuming the same ideal memory bank for both CMOS and CNFET processor. Optimized design with memory realized with CNFETs is out of scope of this work. CNFET technology outperform CMOS with a gain of up to 2.1x at 16 nm node, leading to a maximal performance of 4 GHz. The performance improvement of CNFET processor does not match the results

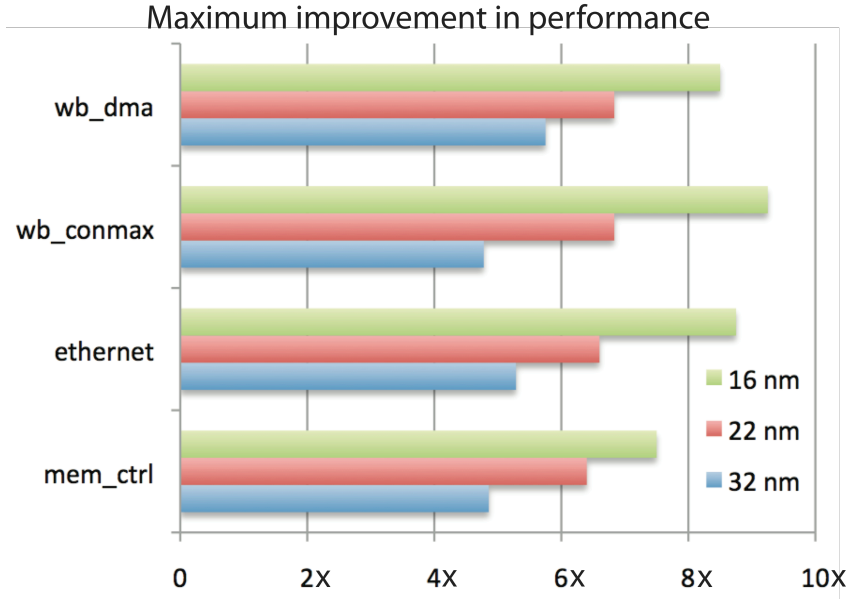


Figure 7.17: Maximum frequency improvement with CNFETs when compared to CMOS technology.

presented in the previous section, where I showed 8.5x improvement at 16 nm node. The main reason comes for limited performance gain coming from the critical path delay to access the data from the memory bank. Further improvement in performance of CNFET processor can be envisaged by realizing the interconnect with CNTs [191].

Fig. 7.18b illustrates the energy-delay-product for the OpenRISC processor. EDP is extracted by assuming same clock constraints for CMOS and CNFET. Averaged across all the nodes, I observe 1.5x improvement in EDP with CNFET processor when compared to equivalent CMOS implementation.

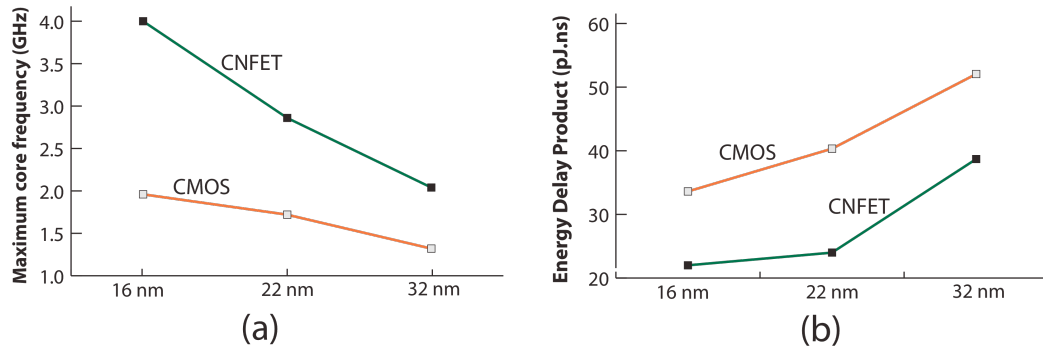


Figure 7.18: OpenRISC 1200 casestudy at various technology nodes. (a) Maximal frequency achievable. (b) Energy-delay-product.

7.7 Chapter Contribution and Summary

In this chapter, robust design techniques to improve the yield of CNFET circuits are presented. With aligned-active layout style, the yield of the CNFET circuits is improved by taking the advantage of CNT correlations. In the context of standard cell based design, two aligned-active grids (one for each PUN and PDN) are needed in order to ensure that all the CNFETs (either in PUN or PDN) placed in a standard cell row have high probability of sharing the same CNTs forming their channel. This chapter also contributes towards a novel mispositioned-immune layout style which ensure immunity towards circuits failures caused by mispositioned-CNTs. Various mispositioned-immune layouts schemes are studied with respect to CNT correlation and cell routing.

In order to improve the overall yield of CNFET circuits, the proposed layout techniques which take into account the imperfections of the state-of-the-art CNT synthesis process, are employed to design the standard cells for CNFET circuits. A yield-enhanced standard cell library is designed by applying both the aligned-active and mispositioned-CNT immune layout styles.

With the yield-enhanced standard cell libraries, system-level benchmarking is carried in order to compare CNFET circuits to their equivalent CMOS circuits at various technology nodes. Averaging across various benchmark circuits, at different technology nodes, a 5.7x improvement in energy-delay-product of CNFET circuit over CMOS circuit is observed. When simulating an OpenRISC heterogeneous processor, processor with CNFET gates is sped up by 2.1x when compared to CMOS at 16 nm technology node.

Conclusions and Future Work

8

This thesis addresses design techniques and CAD tools for three emerging nanotechnologies: 1) the first part presents physical design methodologies for fabricating circuits based on *3D monolithic integrated* (3DMI) technology, 2) the second part deals with *double-gate silicon nanowire FET* (DG-SiNW FET) and 3) the third part deals with imperfection-immune layout techniques for *carbon nanotube FET* (CNFET) circuits. This thesis aims at bridging paths between technology and design for exploring new nanotechnologies. All the design tools presented in this thesis are developed in close collaboration with our technology partners.

The design techniques presented in this thesis focus on unique aspects that are common to all three nanotechnologies (3DMI, DG-SiNWFET, and CNFET). Hence, some of the techniques presented for each of these technologies can be extended to the other.

In the following section, a summary of every chapter is highlighted. Then possible future works are proposed.

8.1 Thesis Summary and Contribution

After introducing this thesis with a general background on emerging nanotechnologies, the second and third chapter deal with physical design techniques for 3D monolithic integrated circuits. The fourth chapter proposes a novel integration scheme for 3D integration combining both 3D monolithic and 3D TSV based technologies. Then, the fifth and sixth chapters are on layout techniques for DG-SiNW FET technologies. The seventh chapter deals with robust layout techniques for imperfection immune CNFET circuits.

In Chapter 2, I present standard cell design techniques for fine-grain 3D circuits based on 3DMI technology. This chapter explores for the first time various cell-transformation techniques for 3DMI circuits. I propose a novel *cell-on-cell* stacking, which enables overlapping of planar standard cells on top of each other without any pin conflicts. I also study the area improvement of *cell-on-cell* stacking compared to the planar as well as *intra-cell* stacking and *intra-cell* folding design techniques. Both *intra-cell* - stacking and folding techniques map a planar (i.e. 2D) standard cell to a 3D standard cell. Whereas *cell-on-cell* stacking enables placing two planar cells on top of each other while maintaining the regularity of ASIC design. When compared to planar and *intra-cell* stacking configuration, I demonstrate improved area efficiency with *cell-on-cell* stacking. Overall performance study comparing the various cell transformations is studied after developing physical synthesis tool for various cell configurations.

In Chapter 3, I study the performance metrics of various 3D cell transformations by realizing a complete physical synthesis flow (i.e. Logic-to-Layout) for *intra-cell* and *cell-on-cell* design techniques. Existing placement tools are employed for *intra-cell* stacking and folding techniques, as they require efforts only in designing the 3D cells. This chapter presents for the first time a new physical synthesis tool (CELONCEL) for *cell-on-cell* stacking. CELONCEL design technique comprises of CELONCELLIB and CELONCELPD. CELONCELLIB comprises of two sets of standard cells, one for the bottom active layer and one for the top layer. CELONCELPD is a pre-/post-processor for existing 2D placement engines which partitions the circuits across two active layers. CELONCELPD transforms the monolithic 3D placement problem into a virtual 2D problem solved using existing 2D placers. This chapter also explores circuit level benchmarking of various circuits mapped with planar (technology mapping) CMOS and 3DMI standard cell libraries at 45 nm node. As compared to traditional 2D physical synthesis flow, with CELONCEL (compared to planar implementation) I demonstrate reduction in the average wirelength, critical path delay, and the die area. Compared to both *intra-cell* - stacking and folding, *cell-on-cell* stacking fares well in wirelength and delay reduction for majority of the benchmark circuits.

In Chapter 4, I propose a novel vertical integration scheme, called 3.5D integration, which synergizes existing 3D TSV and 3DMI technologies. I chose *intra-cell* stacking for realizing gate-level integration with 3DMI technology, thereby increasing the number of cores on a die. I consider a synthetic case study of a 288-core MPSoC to get insight into the advantages and disadvantages of the proposed integration scheme. By applying 3.5D integration to a 288-core MPSoC, I conjecture 30% reduction in number of stacked dies, 20% reduction in the overall manufacturing cost, and 30% reduction in test cost when compared to a 3D TSV implementation. From

technology mapping, our simulation show 11.5% improvement in performance of various benchmarks comprised in the core. I also study the interconnection network, where we observe large improvement in the latency of the 3D NoC (average of 24%) for 3.5-D integration over 3D TSV implementation of the MPSoC.

In Chapter 5, I address for the first time the physical design challenges of ambipolar logic circuits based on DG-SiNW FET technology. With the help of two independent gates, a control gate and a polarity gate, a DG-SiNW FET can be field programmed to either p- or n-type transistor. This unique feature of DG-SiNW FETs opens up new avenues for innovation in the way we do circuit design. This chapter deals with fundamental physical design problem of mitigating the gate-level routing congestion caused by the need to access the two independent gates of each and every transistor. This problem is unique to all the technologies contending for ambipolar logic circuits which employ top-gated transistors with two independent gates. I propose a novel symbolic layout for ambipolar logic called Dumbbell-stick diagram. Since the XOR functionality is inherent to DG-SiNW FET, ambipolar logic is ideal for Boolean functions with embedded XOR/XNOR functions. The main contribution of this chapter is a layout methodology and algorithm for complex functions with embedded XOR/XNOR block. In this chapter, I also study the effectiveness of DG-SiNW FET technology when compared to CMOS technologies, with the help of a first-order model of the device at 22 nm node. From our simulations at the gate level, I demonstrate the effectiveness of DG-SiNW FET in realizing the fundamental arithmetic circuits.

In Chapter 6, I propose a novel layout fabric, called logic *tiles*, for ambipolar logic circuits based on DG-SiNW FETs, which can be configured to various logic gates. With the idea of *Sea-of-Tiles* (SoTs) methodology, I envisage an array of tiles with a constant pitch spread across the chip. I perform technology mapping with various tiles in order to find the tile with maximum area efficiency. I show SoT with tiles Tile_{G2} and Tile_{G1h2} are optimal when considering the intra-cell routing and the respective area utilization of the active area. In this chapter, I demonstrate the mapping of 3-input NPN-equivalent function along with various building blocks for ambipolar logic circuits onto SoT of Tile_{G2} and Tile_{G1h2} , and I conjecture Tile_{G2} to be the fundamental building block. I also study the impact on the performance of various benchmarks by varying the number of vertically stacked nanowires of Tile_{G2} . Circuit-level benchmarking is performed in order to study the benefits of DG-SiNW FET circuits when compared to CMOS circuits at 22nm node. Benchmark circuits are mapped onto SoT to compare the performance (timing, leakage power and area) of logic tiles with CMOS technology at 22 nm technology node. I evaluated the performance of datapath circuits, which are dominated by XOR/XNOR gates. Comparing DG-SiNW FET to CMOS at 22nm, I observe 2.1x improvement in delay with

an area overhead of 17%.

In Chapter 7, I investigate new design techniques for CNFET technology. Current CNFET technology is prone to many CNT-imperfections which affect the yield of CNFET circuits. In this chapter, I address two important CNT imperfections to improve the overall yield of CNFET circuits. First, I propose a novel mispositioned-immune layout style which ensures immunity towards circuits failures caused by mispositioned-CNTs. Second, I present aligned-active layout style which improves the yield of CNFET circuits by taking the advantage of CNT correlations. Then I employed both the layout techniques in order to obtain a yield-enhanced standard cell library. In the second part of the chapter, I address system-level benchmarking of CNFET circuits and compare to their equivalent CMOS circuits at various technology nodes. I observe 5.7x improvement in energy-delay-product of CNFET circuit over CMOS circuit averaged across various benchmarks.

8.2 Future Work

Among the three emerging technologies considered, 3D monolithic integration is a near viable solution for industrial realization. On the other hand, technologies based on silicon nanowires and carbon nanotubes have few fundamental technological limitations to be overcome before industrial adoption. In this section, I highlight future research directions in design methodologies and CAD tools for all the three technologies.

In this first part of the thesis, I presented a novel *cell-on-cell* design for standard cells and a placement tool (CELONCEL) for realizing ultra fine-grain 3D circuits by considering an ideal 3DMI technology. An ideal 3DMI technology features similar transistor characteristics for both top and bottom layer transistors. However, this is not achievable with state-of-the-art technology, as a cost effective 3DMI technology is realized by sequentially stacking a SOI wafer on top of a standard bulk-Si bottom active layer. Taking this into account, one possible extension to CELONCEL placement tool, is to place high performance standard cells (which fall in the critical path) in the bottom active layer and other cells in the top layer. Since the cells placed in the top layer are realized on SOI wafer, I can improve the overall performance. One more design-technology-CAD problem that can be explored is to find the optimal number of intermediate metal layers between the two active layers for 3DMI technology. State-of-the-art 3DMI technologies employ high-thermal resistive metal layers in between the two active layers, so that they can withstand the high temperature when the top active layer is processed. For the 3DMI technology I considered in this thesis, tungsten is considered as an intermediate metal layer. However, these metals are also highly resistive,

thereby increasing the interconnect delay. One optimization problem that can be looked at is to find the optimal number of intermediate metal layers such that it has a minimal impact on the overall interconnect delay. While 3D TSV technology has been widely adopted by the industry, 3DMI technology is still in search for a right application. Hence there is a huge research potential in evaluating 3DMI technology to various designs styles.

In the second part of this thesis, I proposed novel layout synthesis algorithms along with a regular layout fabric for ambipolar logic circuits, realized with DG-SiNW FETs. From the design methodology perspective, this work adopts the idea of regular logic tiles which can be employed to realize semi-custom circuits with *sea-of-tiles* (SoT) architecture. However, it is noteworthy that the logic tiles are inherently reconfigurable. The in-field configurability opens novel opportunities to build reconfigurable logic operators with a very limited number of transistors. Hence, we can envisage using the SoT fabric to efficiently build reconfigurable circuits such as Field Programmable Gate Arrays. However, specific architectural organization should be used in order to keep the wiring complexity minimal. In this study, I employed commercial logic synthesis tool (Design compiler) during the technology-mapping phase with DG-SiNW FET technology. It has to be noted that ambipolar logic gates are efficient in implementing XOR dominated circuits. State-of-the-art logic synthesis tools are effective for unate logic functions, as the Boolean function is decomposed into *And-Inverter* graphs. Hence, we envisage better performance with novel logic synthesis tools specifically designed for XOR dominated circuits. A major aim for future automated synthesis tools is to efficiently manipulate both AND/OR and XOR operations in order to fully harness the potential of novel nanotechnologies featuring transistors with controllable-polarity. On the fabrication side, vertically-stacked SiNW FETs has many challenges. Technologists have to take into account the variations in the diameter of nanowires placed on top of each other. Increasing the number of stacked nanowires increases variations, hence there is an interest to keep the number of stacked nanowires to a minimal number. On the other hand, increasing the number of nanowires improves the drive current of the SiNWFET. This arises a process-design co-optimization problem that can help the technologists to fabricate devices with the optimal number of nanowires. This is a fairly new technology and there is an interest in studying the device characteristics by taking into account the variations linked to the nanowires.

In the final part of the thesis, I proposed layout techniques for improving the yield of CNFET circuits by considering CNT specific non-idealities. I considered two aspects of current CNT-synthesis, mispositioned-CNTs and CNT correlation. Now that we proposed active-aligned layout technique, specific to CNT correlations, it will be interesting to find if CNT correlation can mitigate delay variations in CNFET circuits. On the CAD side, we can envisage

developing new placement techniques by taking into account CNT correlations.

Most of the design techniques presented in this thesis are specific to standard ASIC design. It will be interesting to see how these ideas and techniques can be applied to other important applications such as memories and reconfigurable circuits.

Bibliography

- [1] G. Moore, "Cramming more components onto integrated circuits, electronics 38," 1965.
- [2] S. Thompson, M. Armstrong, C. Auth, M. Alavi, M. Buehler, R. Chau, S. Cea, T. Ghani, G. Glass, T. Hoffman, *et al.*, "A 90-nm logic technology featuring strained-silicon," *Electron Devices, IEEE Transactions on*, vol. 51, no. 11, pp. 1790–1797, 2004.
- [3] P. Bai, C. Auth, S. Balakrishnan, M. Bost, R. Brain, V. Chikarmane, R. Heussner, M. Hussein, J. Hwang, D. Ingerly, *et al.*, "A 65nm logic technology featuring 35nm gate lengths, enhanced channel strain, 8 cu interconnect layers, low-k ild and 0.57 μm sup_i 2_i/sup_i sram cell," in *Electron Devices Meeting, 2004. IEDM Technical Digest. IEEE International*, pp. 657–660, IEEE, 2004.
- [4] S. Natarajan, M. Armstrong, M. Bost, R. Brain, M. Brazier, C. Chang, V. Chikarmane, M. Childs, H. Deshpande, K. Dev, *et al.*, "A 32nm logic technology featuring 2 nd-generation high-k+ metal-gate transistors, enhanced channel strain and 0.171 μm 2 sram cell size in a 291mb array," in *Electron Devices Meeting, 2008. IEDM 2008. IEEE International*, pp. 1–3, IEEE, 2008.
- [5] K. Mistry, C. Allen, C. Auth, B. Beattie, D. Bergstrom, M. Bost, M. Brazier, M. Buehler, A. Cappellani, R. Chau, *et al.*, "A 45nm logic technology with high-k+ metal gate transistors, strained silicon, 9 cu interconnect layers, 193nm dry patterning, and 100% pb-free packaging," in *Electron Devices Meeting, 2007. IEDM 2007. IEEE International*, pp. 247–250, IEEE, 2007.
- [6] C. Auth, A. Cappellani, J. Chun, A. Dalis, A. Davis, T. Ghani, G. Glass, T. Glassman, M. Harper, M. Hattendorf, *et al.*, "45nm high-k+ metal gate strain-enhanced transistors," in *VLSI Technology, 2008 Symposium on*, pp. 128–129, IEEE, 2008.
- [7] P. Packan, S. Akbar, M. Armstrong, D. Bergstrom, M. Brazier, H. Deshpande, K. Dev, G. Ding, T. Ghani, O. Golonzka, *et al.*, "High perfor-

- mance 32nm logic technology featuring 2_i sup_i nd_i/sup_i generation high-k+ metal gate transistors,” in *Electron Devices Meeting (IEDM), 2009 IEEE International*, pp. 1–4, IEEE, 2009.
- [8] D. Hisamoto, W. Lee, J. Kedzierski, E. Anderson, H. Takeuchi, K. Asano, T. King, J. Bokor, and C. Hu, “A folded-channel mosfet for deep-sub-tenth micron era,” *IEDM Tech. Dig.*, vol. 1998, pp. 1032–1034, 1998.
 - [9] B. Doyle, S. Datta, M. Doczy, S. Hareland, B. Jin, J. Kavalieros, T. Linton, A. Murthy, R. Rios, and R. Chau, “High performance fully-depleted tri-gate cmos transistors,” *Electron Device Letters, IEEE*, vol. 24, no. 4, pp. 263–265, 2003.
 - [10] K. Ahmed and K. Schuegraf, “Transistor wars,” *Spectrum, IEEE*, vol. 48, no. 11, pp. 50–66, 2011.
 - [11] W. Arden, M. Brillouët, P. Cogez, M. Graef, B. Huizing, and R. Mahnkopf, “More-than-moore white paper,” *Version*, vol. 2, p. 14, 2010.
 - [12] V. Pavlidis and E. Friedman, *Three-dimensional integrated circuit design*. Morgan Kaufmann Pub, 2009.
 - [13] J. Von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior (Commemorative Edition)*. Princeton university press, 2007.
 - [14] G. Moore, “Progress in digital integrated electronics,” in *Electron Devices Meeting, 1975 International*, vol. 21, pp. 11–13, IEEE, 1975.
 - [15] R. Dennard, F. Gaensslen, V. Rideout, E. Bassous, and A. LeBlanc, “Design of ion-implanted mosfet’s with very small physical dimensions,” *Solid-State Circuits, IEEE Journal of*, vol. 9, no. 5, pp. 256–268, 1974.
 - [16]
 - [17] K. Kuhn, M. Liu, and H. Kennel, “Technology options for 22nm and beyond,” in *Junction Technology (IWJT), 2010 International Workshop on*, pp. 1–6, IEEE, 2010.
 - [18] K. Rim, S. Koester, M. Hargrove, J. Chu, P. Mooney, J. Ott, T. Karnarsky, P. Ronsheim, M. Jeong, A. Grill, *et al.*, “Strained si nmosfets for high performance cmos technology,” in *VLSI Technology, 2001. Digest of Technical Papers. 2001 Symposium on*, pp. 59–60, IEEE, 2001.
 - [19] M. Akbar, H. Cho, R. Choi, C. Kang, C. Kang, C. Choi, S. Rhee, Y. Kim, and J. Lee, “Optimized nh_i sub_i 3_i/sub_i annealing process for high-quality hfsion gate oxide,” *Electron Device Letters, IEEE*, vol. 25, no. 7, pp. 465–467, 2004.

- [20] H. Lim and J. Fossum, "Threshold voltage of thin-film silicon-on-insulator (soi) mosfet's," *Electron Devices, IEEE Transactions on*, vol. 30, no. 10, pp. 1244–1251, 1983.
- [21] F. Balestra, S. Cristoloveanu, M. Benachir, J. Brini, and T. Elewa, "Double-gate silicon-on-insulator transistor with volume inversion: A new device with greatly enhanced performance," *Electron Device Letters, IEEE*, vol. 8, no. 9, pp. 410–412, 1987.
- [22] J. Park, J. Colinge, and C. Diaz, "Pi-gate soi mosfet," *Electron Device Letters, IEEE*, vol. 22, no. 8, pp. 405–406, 2001.
- [23] F. Yang, H. Chen, F. Chen, C. Huang, C. Chang, H. Chiu, C. Lee, C. Chen, H. Huang, C. Chen, *et al.*, "25 nm cmos omega fets," in *Electron Devices Meeting, 2002. IEDM'02. International*, pp. 255–258, IEEE, 2002.
- [24] D. Sacchetto, M. H. Ben-Jamaa, G. De Micheli, and Y. Leblebici, "Fabrication and characterization of vertically stacked gate-all-around si nanowire fet arrays," in *Solid State Device Research Conference, 2009. ESSDERC'09. Proceedings of the European*, pp. 245–248, IEEE, 2009.
- [25] M. De Marchi, M. Jamaa, and G. De Micheli, "Regular fabric design with ambipolar cntfets for fpga and structured asic applications," in *Proceedings of the 2010 IEEE/ACM International Symposium on Nanoscale Architectures*, pp. 65–70, IEEE Press, 2010.
- [26] I. O'Connor, J. Liu, D. Navarro, I. Hassoune, S. Burignat, and F. Gaffiot, "Ultra-fine grain reconfigurability using cntfets," in *Electronics, Circuits and Systems, 2007. ICECS 2007. 14th IEEE International Conference on*, pp. 194–197, IEEE, 2007.
- [27] B. Radisavljevic, A. Radenovic, J. Brivio, V. Giacometti, and A. Kis, "Single-layer mos2 transistors," *Nature nanotechnology*, vol. 6, no. 3, pp. 147–150, 2011.
- [28] S. Iijima *et al.*, "Helical microtubules of graphitic carbon," *nature*, vol. 354, no. 6348, pp. 56–58, 1991.
- [29] S. Iijima and T. Ichihashi, "Single-shell carbon nanotubes of 1-nm diameter," 1993.
- [30] N. Patil, A. Lin, J. Zhang, H. Wong, and S. Mitra, "Digital vlsi logic technology using carbon nanotube fets: Frequently asked questions," in *Design Automation Conference, 2009. DAC'09. 46th ACM/IEEE*, pp. 304–309, IEEE, 2009.

- [31] H. Wei, N. Patil, A. Lin, H. Wong, and S. Mitra, "Monolithic three-dimensional integrated circuits using carbon nanotube fets and interconnects," in *Electron Devices Meeting (IEDM), 2009 IEEE International*, pp. 1–4, IEEE, 2009.
- [32] A. Franklin, M. Luisier, S. Han, G. Tulevski, C. Breslin, L. Gignac, M. Lundstrom, and W. Haensch, "Sub-10 nm carbon nanotube transistor," *Nano letters*, vol. 12, no. 2, pp. 758–762, 2012.
- [33] J. Deng and H. Wong, "A compact spice model for carbon-nanotube field-effect transistors including nonidealities and its application—part ii: full device model and circuit performance benchmarking," *Electron Devices, IEEE Transactions on*, vol. 54, no. 12, pp. 3195–3205, 2007.
- [34] H. Dai, "Carbon nanotubes: opportunities and challenges," *Surface Science*, vol. 500, no. 1, pp. 218–241, 2002.
- [35] S. Kasai and H. Hasegawa, "A single electron binary-decision-diagram quantum logic circuit based on schottky wrap gate control of a gaas nanowire hexagon," *Electron Device Letters, IEEE*, vol. 23, no. 8, pp. 446–448, 2002.
- [36] M. Lutwyche and Y. Wada, "Estimate of the ultimate performance of the single-electron transistor," *Journal of applied physics*, vol. 75, no. 7, pp. 3654–3661, 1994.
- [37] H. Inokawa, A. Fujiwara, and Y. Takahashi, "A multiple-valued logic and memory with combined single-electron and metal-oxide-semiconductor transistors," *Electron Devices, IEEE Transactions on*, vol. 50, no. 2, pp. 462–470, 2003.
- [38] S. Koester, A. Young, R. Yu, S. Purushothaman, K. Chen, D. La Tulipe, N. Rana, L. Shi, M. Wordeman, and E. Sprogis, "Wafer-level 3d integration technology," *IBM Journal of Research and Development*, vol. 52, no. 6, pp. 583–597, 2008.
- [39] N. Sillon, A. Astier, H. Boutry, L. Di Cioccio, D. Henry, and P. Leduc, "Enabling technologies for 3d integration: From packaging miniaturization to advanced stacked ics," in *Electron Devices Meeting, 2008. IEDM 2008. IEEE International*, pp. 1–4, IEEE, 2008.
- [40] P. Batude, M. Vinet, A. Pouydebasque, C. Le Royer, B. Previtali, C. Tabone, L. Clavelier, S. Michaud, A. Valentian, O. Thomas, *et al.*, "Geoi and soi 3d monolithic cell integrations for high density applications," in *VLSI Technology, 2009 Symposium on*, pp. 166–167, IEEE, 2009.

- [41] S. Das, A. Chandrakasan, and R. Reif, "Design tools for 3-d integrated circuits," in *Design Automation Conference, 2003. Proceedings of the ASP-DAC 2003. Asia and South Pacific*, pp. 53 – 56, jan. 2003.
- [42] J. Cong, G. Luo, J. Wei, and Y. Zhang, "Thermal-aware 3d ic placement via transformation," in *Design Automation Conference, 2007. ASP-DAC'07. Asia and South Pacific*, pp. 780–785, IEEE, 2007.
- [43] L. Zhou, C. Wakayama, and C.-J. Shi, "Cascade: A standard super-cell design methodology with congestion-driven placement for three-dimensional interconnect-heavy very large-scale integrated circuits," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, pp. 1270 –1282, july 2007.
- [44] W. R. R. R. Cavin, W. Joyner, "The brave new old world of design automation research," 2009.
- [45] R. Havemann and J. Hutchby, "High-performance interconnects: An integration overview," *Proceedings of the IEEE*, vol. 89, no. 5, pp. 586–601, 2001.
- [46] G. Loh, Y. Xie, and B. Black, "Processor design in 3d die-stacking technologies," *Micro, IEEE*, vol. 27, no. 3, pp. 31–48, 2007.
- [47] P. Batude, M. Vinet, A. Pouydebasque, C. Le Royer, B. Previtali, C. Tabone, J. Hartmann, L. Sanchez, L. Baud, V. Carron, *et al.*, "Advances in 3d cmos sequential integration," in *Electron Devices Meeting (IEDM), 2009 IEEE International*, pp. 1–4, IEEE, 2009.
- [48] "Tezzaron semiconductor," 2011.
- [49] D. Kim, K. Athikulwongse, and S. Lim, "A study of through-silicon-via impact on the 3d stacked ic layout," in *Proceedings of the 2009 International Conference on Computer-Aided Design*, pp. 674–680, ACM, 2009.
- [50] S. Bobba, A. Chakraborty, O. Thomas, P. Batude, V. Pavlidis, and G. De Micheli, "Performance analysis of 3-d monolithic integrated circuits," in *3D Systems Integration Conference (3DIC), 2010 IEEE International*, pp. 1–4, IEEE, 2010.
- [51] C. Chen, H. Lam, S. Malhi, and R. Pinizzotto, "Stacked cmos sram cell," *Electron Device Letters, IEEE*, vol. 4, no. 8, pp. 272–274, 1983.
- [52] A. H. Shah, L. R. Hite, S. S. M. Shetti, P. K. Chatterjee, H. E. Davis, R. K. Hester, S. D. S. Malhi, R. Karnaugh, C. D. Gosmeyer, R. S. Sundaresan, C. E. Chen, H. W. Lam, and R. A. Haken, "A 2 um stacked cmos 64k sram," in *VLSI Technology, 1984. Digest of Technical Papers. Symposium on*, pp. 8 –9, sept. 1984.

- [53] J. Gibbons and K. Lee, "One-gate-wide cmos inverter on laser-recrystallized polysilicon," *Electron Device Letters, IEEE*, vol. 1, no. 6, pp. 117–118, 1980.
- [54] J. Gibbons and K. Lee, "A folding principle for generating three-dimensional mosfet device structures in beam-recrystallized polysilicon films," in *Electron Devices Meeting, 1982 International*, vol. 28, pp. 111–114, IEEE, 1982.
- [55] G. Goeloe, E. Maby, D. Silversmith, R. Mountain, and D. Antoniadis, "Vertical single-gate cmos inverters on laser-processed multilayer substrates," in *Electron Devices Meeting, 1981 International*, vol. 27, pp. 554–556, IEEE, 1981.
- [56] J. Colinge and E. Demoulin, "A high density cmos inverter with stacked transistors," *Electron Device Letters, IEEE*, vol. 2, no. 10, pp. 250–251, 1981.
- [57] S. Kawamura, N. Sasaki, T. Iwai, M. Nakano, and M. Takagi, "Three-dimensional cmos ic's fabricated by using beam recrystallization," *Electron Device Letters, IEEE*, vol. 4, no. 10, pp. 366–368, 1983.
- [58] S. Kawamura, N. Sasaki, T. Iwai, R. Mukai, M. Nakano, and M. Takagi, "3-dimensional gate array with vertically stacked dual soi/cmos structure fabricated by beam recrystallization," in *VLSI Technology, 1984. Digest of Technical Papers. Symposium on*, pp. 44–45, IEEE, 1984.
- [59] S. Kawamura, N. Sasaki, S. Kawai, T. Shirato, N. Aneha, and M. Nakano, "3-d high-voltage cmos ics by recrystallized soi merged with bulk control-unit," in *Electron Devices Meeting, 1987 International*, vol. 33, pp. 758–761, IEEE, 1987.
- [60] K. Ohtake, K. Shirakawa, M. Koba, K. Awane, Y. Ohta, D. Azuma, and S. Miyata, "Triple layered soi dynamic memory," in *Electron Devices Meeting, 1986 International*, vol. 32, pp. 148–151, IEEE, 1986.
- [61] R. Zingg, B. Hofflinger, and G. Neudeck, "Stacked cmos inverter with symmetric device performance," in *Electron Devices Meeting, 1989. IEDM'89. Technical Digest., International*, pp. 909–911, IEEE, 1989.
- [62] Y. Takao, H. Shimada, N. Suzuki, Y. Matsukawa, Y. Kobayashi, and N. Sasaki, "A low-power sram utilizing high on/off ratio laser-recrystallized soi pmosfet load," in *VLSI Circuits, 1991. Digest of Technical Papers. 1991 Symposium on*, pp. 95–96, 1991.
- [63] Y. Takao, H. Shimada, N. Suzuki, Y. Matsukawa, and N. Sasaki, "Low-power and high-stability sram technology using a laser-recrystallized p-channel soi mosfet," *Electron Devices, IEEE Transactions on*, vol. 39, no. 9, pp. 2147–2152, 1992.

- [64] G. Roos and B. Hoefflinger, "Complex 3d cmos circuits based on a triple-decker cell," *Solid-State Circuits, IEEE Journal of*, vol. 27, no. 7, pp. 1067–1072, 1992.
- [65] G. Roos and B. Hoefflinger, "Three-dimensional cmos nand with three stacked channels," *Electronics Letters*, vol. 29, no. 24, pp. 2103–2104, 1993.
- [66] Y. Uemoto, E. Fujii, A. Nakamura, and K. Senda, "A high-performance stacked-cmos sram cell by solid phase growth technique," in *VLSI Technology, 1990. Digest of Technical Papers. 1990 Symposium on*, pp. 21–22, IEEE, 1990.
- [67] V. Subramanian and K. Saraswat, "High-performance germanium-seeded laterally crystallized tfts for vertical device integration," *Electron Devices, IEEE Transactions on*, vol. 45, no. 9, pp. 1934–1939, 1998.
- [68] V. Chan, P. Chan, and M. Chan, "Multiple layers of cmos integrated circuits using recrystallized silicon film," *Electron Device Letters, IEEE*, vol. 22, no. 2, pp. 77–79, 2001.
- [69] S. Tiwari, H. Kim, S. Kim, A. Kumar, C. Liu, and L. Xue, "Three-dimensional integration in silicon electronics," in *High Performance Devices, 2002. Proceedings. IEEE Lester Eastman Conference on*, pp. 24–33, IEEE, 2002.
- [70] S. Zhang, R. Han, X. Lin, X. Wu, and M. Chan, "A stacked cmos technology on soi substrate," *Electron Device Letters, IEEE*, vol. 25, no. 9, pp. 661–663, 2004.
- [71] D. Yu, A. Chin, C. Liao, C. Lee, C. Cheng, M. Li, W. Yoo, and S. McAlister, "Three-dimensional metal gate-high- κ -goi cmosfets on 1-poly-6-metal 0.18- μm si devices," *Electron Device Letters, IEEE*, vol. 26, no. 2, pp. 118–120, 2005.
- [72] X. Wu, P. Chan, S. Zhang, C. Feng, and M. Chan, "A three-dimensional stacked fin-cmos technology for high-density ulsi circuits," *Electron Devices, IEEE Transactions on*, vol. 52, no. 9, pp. 1998–2003, 2005.
- [73] J. Feng, Y. Liu, P. Griffin, and J. Plummer, "Integration of germanium-on-insulator and silicon mosfets on a silicon substrate," *Electron Device Letters, IEEE*, vol. 27, no. 11, pp. 911–913, 2006.
- [74] M. Mofrad, R. Ishihara, J. Derakhshandeh, A. Baiano, J. van der Cingel, and C. Beenakker, "Monolithic 3d integration of single-grain si tfts," *Proceedings of Mat. Res. Soc (San Francisco, USA)*, 2008.

- [75] P. Batude, M. Vinet, C. Xu, B. Previtali, C. Tabone, C. Le Royer, L. Sanchez, L. Baud, L. Brunet, A. Toffoli, *et al.*, “Demonstration of low temperature 3d sequential fdsoi integration down to 50 nm gate length,” in *VLSI Technology (VLSIT), 2011 Symposium on*, pp. 158–159, IEEE, 2011.
- [76] Y. Kang, S. Jung, J. Jang, J. Moon, W. Cho, C. Yeo, K. Kwak, B. Choi, B. Hwang, W. Jung, *et al.*, “Fabrication and characteristics of novel load pmos sstft (stacked single-crystal thin film transistor) for 3-dimensional sram memory cell,” in *SOI Conference, 2004. Proceedings. 2004 IEEE International*, pp. 127–129, IEEE, 2004.
- [77] S. Jung, Y. Rah, T. Ha, H. Park, C. Chang, S. Lee, J. Yun, W. Cho, H. Lim, J. Park, *et al.*, “Highly cost effective and high performance 65nm s_i sup_i 3_i/sup_i(stacked single-crystal si) sram technology with 25f_i sup_i 2_i/sup_i, 0.16 um_i sup_i 2_i/sup_i cell and doubly stacked sstft cell transistors for ultra high density and high speed applications,” in *VLSI Technology, 2005. Digest of Technical Papers. 2005 Symposium on*, pp. 220–221, IEEE, 2005.
- [78] S. Jung, J. Jang, W. Cho, H. Cho, J. Jeong, Y. Chang, J. Kim, Y. Rah, Y. Son, J. Park, *et al.*, “Three dimensionally stacked nand flash memory technology using stacking single crystal si layers on ild and tanos structure for beyond 30nm node,” in *Electron Devices Meeting, 2006. IEDM’06. International*, pp. 1–4, IEEE, 2006.
- [79] K. Sohn, H. Mo, Y. Suh, H. Byun, and H. Yoo, “An autonomous sram with on-chip sensors in an 80-nm double stacked cell technology,” *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 4, pp. 823–830, 2006.
- [80] S. Jung, H. Lim, C. Yeo, K. Kwak, B. Son, H. Park, J. Na, J. Shim, C. Hong, and K. Kim, “High speed and highly cost effective 72m bit density s_i sup_i 3_i/sup_i sram technology with doubly stacked si layers, peripheral only cosix layers and tungsten shunt w/l scheme for standalone and embedded memory,” in *VLSI Technology, 2007 IEEE Symposium on*, pp. 82–83, IEEE, 2007.
- [81] Y. Son, J. Lee, P. Kang, M. Kang, J. Kim, S. Lee, Y. Kim, I. Jung, B. Lee, S. Choi, *et al.*, “Laser-induced epitaxial growth (leg) technology for high density 3-d stacked memory with high productivity,” in *VLSI Technology, 2007 IEEE Symposium on*, pp. 80–81, IEEE, 2007.
- [82] K. Sohn, Y. Suh, Y. Son, D. Yim, K. Kim, D. Bae, T. Kang, H. Lim, S. Jung, H. Byun, *et al.*, “A 100nm double-stacked 500mhz 72mb separate-i/o synchronous sram with automatic cell-bias scheme and adaptive block redundancy,” in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pp. 386–622, IEEE, 2008.

- [83] H. Liu, M. Kumar, and J. Sin, "A novel 3-d bicmos technology using selective epitaxy growth (seg) and lateral solid phase epitaxial (lspe)," *Electron Device Letters, IEEE*, vol. 23, no. 3, pp. 151–153, 2002.
- [84] P. Batude, M. Jaud, O. Thomas, L. Clavelier, A. Pouydebasque, M. Vinet, S. Deleonibus, and A. Amara, "3d cmos integration: Introduction of dynamic coupling and application to compact and robust 4t sram," in *Integrated Circuit Design and Technology and Tutorial, 2008. ICICDT 2008. IEEE International Conference on*, pp. 281–284, IEEE, 2008.
- [85] S. Jung, J. Jang, W. Cho, J. Moon, K. Kwak, B. Choi, B. Hwang, H. Lim, J. Jeong, J. Kim, *et al.*, "The revolutionary and truly 3-dimensional 25fj supj 2i/supj sram technology with the smallest sj supj 3i/supj (stacked single-crystal si) cell, 0.16 umj supj 2i/supj, and sstft (atacked single-crystal thin film transistor) for ultra high density sram," in *VLSI Technology, 2004. Digest of Technical Papers. 2004 Symposium on*, pp. 228–229, IEEE, 2004.
- [86] P. Batude, M. Vinet, A. Pouydebasque, L. Clavelier, C. LeRoyer, C. Tabone, B. Previtali, L. Sanchez, L. Baud, A. Roman, *et al.*, "Enabling 3d monolithic integration," *ECS Transactions*, vol. 16, no. 8, pp. 47–54, 2008.
- [87] M. Jeong, K. Guarini, V. Chan, K. Bernstein, R. Joshi, J. Kedzierski, and W. Haensch, "Three dimensional cmos devices and integrated circuits," in *Custom Integrated Circuits Conference, 2003. Proceedings of the IEEE 2003*, pp. 207–213, IEEE, 2003.
- [88] "Nangate opencell library," 2011.
- [89] Y. Deng and W. Maly, "Interconnect characteristics of 2.5-d system integration scheme," in *Proceedings of the 2001 international symposium on Physical design*, pp. 171–175, ACM, 2001.
- [90] J. Roy, D. Papa, S. Adya, H. Chan, A. Ng, J. Lu, and I. Markov, "Capo: robust and scalable open-source min-cut floorplacer," in *Proceedings of the 2005 international symposium on Physical design*, pp. 224–226, ACM, 2005.
- [91] T. Chan, J. Cong, J. Shinnerl, K. Sze, and M. Xie, "mpl6: enhanced multilevel mixed-size placement," in *Proceedings of the 2006 international symposium on Physical design*, pp. 212–214, ACM, 2006.
- [92] Z. Jiang, T. Cheny, T. Hsuy, H. Chenz, and Y. Changyz, "Ntuplace2: A hybrid placer using partitioning and analytical techniques," in *Proceedings of the 2006 international symposium on Physical design*, pp. 215–217, ACM, 2006.

- [93] “Cadence encounter.”
- [94] X. Yang, M. Wang, R. Kastner, S. Ghiasi, and M. Sarrafzadeh, “Congestion reduction during placement with provably good approximation bound,” *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 8, no. 3, pp. 316–333, 2003.
- [95] A. Chakraborty, A. Kumar, and D. Pan, “Regplace: a high quality open-source placement framework for structured asics,” in *Design Automation Conference, 2009. DAC’09. 46th ACM/IEEE*, pp. 442–447, IEEE, 2009.
- [96] “Gurobi optimization tool.”
- [97] “Synopsys design compiler (a-2007.12-sp4).”
- [98] “Open source benchmarks.”
- [99] “Itc99 benchmark suite.”
- [100] “Mentor calibre xrc.”
- [101] J. Van Olmen, A. Mercha, G. Katti, C. Huyghebaert, J. Van Aelst, E. Seppala, Z. Chao, S. Armini, J. Vaes, R. Teixeira, *et al.*, “3d stacked ic demonstration using a through silicon via first approach,” in *Electron Devices Meeting, 2008. IEDM 2008. IEEE International*, pp. 1–4, IEEE, 2008.
- [102] Y. Deng and W. Maly, “2.5 d system integration: a design driven system implementation schema,” in *Design Automation Conference, 2004. Proceedings of the ASP-DAC 2004. Asia and South Pacific*, pp. 450–455, IEEE, 2004.
- [103]
- [104] M. Taylor, J. Kim, J. Miller, D. Wentzlaff, F. Ghodrat, B. Greenwald, H. Hoffman, P. Johnson, J. Lee, W. Lee, *et al.*, “The raw microprocessor: A computational fabric for software circuits and general-purpose programs,” *Micro, IEEE*, vol. 22, no. 2, pp. 25–35, 2002.
- [105] “Tilera. 100 core commercial processor tile-gx,”
- [106] S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, A. Singh, T. Jacob, S. Jain, *et al.*, “An 80-tile sub-100-w teraflops processor in 65-nm cmos,” *Solid-State Circuits, IEEE Journal of*, vol. 43, no. 1, pp. 29–41, 2008.
- [107] G. De Micheli and L. Benini, *Networks on chips: technology and tools*. Morgan Kaufmann, 2006.

- [108] C. Seiculescu, “Design methods and tools for application-specific predictable networks-on-chip,”
- [109] L. Benini and G. De Micheli, “Networks on chips: A new soc paradigm,” *Computer*, vol. 35, no. 1, pp. 70–78, 2002.
- [110] C. Seiculescu, S. Murali, L. Benini, and G. De Micheli, “Sunfloor 3d: a tool for networks on chip topology synthesis for 3-d systems on chips,” *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 29, no. 12, pp. 1987–2000, 2010.
- [111] C. Jalier, D. Lattard, A. Jerraya, G. Sassatelli, P. Benoit, and L. Torres, “Heterogeneous vs homogeneous mpsoc approaches for a mobile lte modem,” in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 184–189, European Design and Automation Association, 2010.
- [112] X. Dong and Y. Xie, “System-level cost analysis and design exploration for three-dimensional integrated circuits (3d ics),” in *Design Automation Conference, 2009. ASP-DAC 2009. Asia and South Pacific*, pp. 234–241, IEEE, 2009.
- [113] E. Marinissen and Y. Zorian, “Testing 3d chips containing through-silicon vias,” in *Test Conference, 2009. ITC 2009. International*, pp. 1–11, IEEE, 2009.
- [114] S. Bobba, A. Chakraborty, O. Thomas, P. Batude, T. Ernst, O. Faynot, D. Pan, and G. De Micheli, “Celoncel: Effective design technique for 3-d monolithic integration targeting high performance integrated circuits,” in *Proceedings of the 16th Asia and South Pacific Design Automation Conference*, pp. 336–343, IEEE Press, 2011.
- [115] C. Seiculescu, S. Volos, N. Pour, B. Falsafi, and G. De Micheli, “Ccnoc: On-chip interconnects for cache-coherent manycore server chips,” in *Workshop on Energy-Efficient Design (WEED 2011)*, 2011.
- [116] S. Stergiou, F. Angiolini, S. Carta, L. Raffo, D. Bertozzi, and G. De Micheli, “ $\times$ pipes lite: A synthesis oriented design library for networks on chips,” in *Design, Automation and Test in Europe, 2005. Proceedings*, pp. 1188–1193, IEEE, 2005.
- [117] W. Dally and B. Towles, *Principles and practices of interconnection networks*. Morgan Kaufmann, 2003.
- [118] S. Suk, S. Lee, S. Kim, E. Yoon, M. Kim, M. Li, C. Oh, K. Yeo, S. Kim, D. Shin, *et al.*, “High performance 5nm radius twin silicon nanowire mosfet (tsnwfet): fabrication on bulk si wafer, characteristics, and reliability,” in *Electron Devices Meeting, 2005. IEDM Technical Digest. IEEE International*, pp. 717–720, IEEE, 2005.

- [119] R. M. Ng, T. Wang, and M. Chan, "A new approach to fabricate vertically stacked single-crystalline silicon nanowires," in *Electron Devices and Solid-State Circuits, 2007. EDSSC 2007. IEEE Conference on*, pp. 133–136, IEEE, 2007.
- [120] D. Sacchetto, M. Ben-Jamaa, G. De Micheli, and Y. Leblebici, "Fabrication and characterization of vertically stacked gate-all-around si nanowire fet arrays," in *Solid State Device Research Conference, 2009. ESSDERC'09. Proceedings of the European*, pp. 245–248, IEEE, 2009.
- [121] M. H. Ben Jamaa, D. Atienza, Y. Leblebici, and G. De Micheli, "Programmable logic circuits based on ambipolar cnfet," in *Proceedings of the 45th annual Design Automation Conference*, pp. 339–340, ACM, 2008.
- [122] M. De Marchi, D. Sacchetto, S. Frache, J. Zhang, P. Gaillardon, Y. Leblebici, and G. De Micheli, "Polarity control in double-gate, gate-all-around vertically stacked silicon nanowire fets,"
- [123] C.-Y. Hwang, Y.-C. Hsieh, Y.-L. Lin, and Y.-C. Hsu, "A fast transistor-chaining algorithm for cmos cell layout," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 9, no. 7, pp. 781–786, 1990.
- [124] T. Uehara and W. VanCleemput, "Optimal layout of cmos functional arrays," *Computers, IEEE Transactions on*, vol. 100, no. 5, pp. 305–312, 1981.
- [125] R. L. Maziasz and J. P. Hayes, "Layout optimization of static cmos functional cells," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 9, no. 7, pp. 708–719, 1990.
- [126] A. Colli, S. Pisana, A. Fasoli, J. Robertson, and A. Ferrari, "Electronic transport in ambipolar silicon nanowires," *physica status solidi (b)*, vol. 244, no. 11, pp. 4161–4164, 2007.
- [127] Y.-M. Lin, J. Appenzeller, J. Knoch, and P. Avouris, "High-performance carbon nanotube field-effect transistor with tunable polarities," *Nanotechnology, IEEE Transactions on*, vol. 4, no. 5, pp. 481–489, 2005.
- [128] A. K. Geim and K. S. Novoselov, "The rise of graphene," *Nature materials*, vol. 6, no. 3, pp. 183–191, 2007.
- [129] I. O'Connor, L. Junchen, F. Gaffiot, F. Prégaldiny, C. Lallement, C. Maneux, J. Goguet, S. Frégonèse, T. Zimmer, L. Anghel, *et al.*, "Cntfet modeling and reconfigurable logic-circuit design," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 54, no. 11, pp. 2365–2379, 2007.

- [130] M. H. Ben Jamaa, K. Mohanram, and G. De Micheli, "Novel library of logic gates with ambipolar cntfets: Opportunities for multi-level logic synthesis," in *Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE'09.*, pp. 622–627, IEEE, 2009.
- [131] A. Zukoski, X. Yang, and K. Mohanram, "Universal logic modules based on double-gate carbon nanotube transistors," in *Design Automation Conference (DAC), 2011 48th ACM/EDAC/IEEE*, pp. 884–889, IEEE, 2011.
- [132] M. De Marchi, S. Bobba, M. Ben Jamaa, and G. De Micheli, "Synthesis of regular computational fabrics with ambipolar cntfet technology," in *Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on*, pp. 70–73, IEEE, 2010.
- [133] R. H. Katz and G. Borriello, "Contemporary logic design," 2005.
- [134] C. Mead and L. Conway, "Introduction to vlsi systems," *Textbook in preparation*, 1978.
- [135] W.-Y. Loh, P. Hung, B. Coss, P. Kalra, I. Ok, G. Smith, C.-Y. Kang, S.-H. Lee, J. Oh, B. Sassman, *et al.*, "Selective phase modulation of nisi using n-ion implantation for high performance dopant-segregated source/drain n-channel mosfets," in *VLSI Technology, 2009 Symposium on*, pp. 100–101, IEEE, 2009.
- [136] B. Coss, W.-Y. Loh, J. Oh, G. Smith, C. Smith, H. Adhikari, B. Sassman, S. Parthasarathy, J. Barnett, P. Majhi, *et al.*, "Cmos band-edge schottky barrier heights using dielectric-dipole mitigated (ddm) metal/si for source/drain contact resistance reduction," in *VLSI Technology, 2009 Symposium on*, pp. 104–105, IEEE, 2009.
- [137] "Predictive technology model."
- [138] T. Jhaveri, L. Pileggi, V. Rovner, and A. J. Strojwas, "Maximization of layout printability/manufacturability by extreme layout regularity," in *SPIE 31st International Symposium on Advanced Lithography*, pp. 615609–615609, International Society for Optics and Photonics, 2006.
- [139] Y.-W. Lin, M. Marek-Sadowska, and W. Maly, "Transistor-level layout of high-density regular circuits," in *Proceedings of the 2009 international symposium on Physical design*, pp. 83–90, ACM, 2009.
- [140] Y. Ran and M. Marek-Sadowska, "Designing via-configurable logic blocks for regular fabric," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 14, no. 1, pp. 1–14, 2006.

- [141] B. Taylor and L. Pileggi, “Exact combinatorial optimization methods for physical design of regular logic bricks,” in *Design Automation Conference, 2007. DAC’07. 44th ACM/IEEE*, pp. 344–349, IEEE, 2007.
- [142] F. Mailhot, *Technology Mapping for VLSI Circuits Exploiting Boolean Properties and Operations*. PhD thesis, to the Department of Electrical Engineering. Stanford University, 1991.
- [143] S. L. Hurst, J. C. Muzio, and D. M. Miller, *Spectral techniques in digital logic*. Academic Press, Inc., 1985.
- [144] B. L. SYNTHESIS, “Abc: A system for sequential synthesis and verification,” *Berkeley Logic Synthesis and Verification Group*, 2011.
- [145] S. Muroga, *Threshold logic and its applications*, vol. 13. Wiley-Interscience New York, 1971.
- [146] R. K. Brayton, G. D. Hachtel, C. McMullen, and A. L. Sangiovanni-Vincentelli, *Logic minimization algorithms for VLSI synthesis*, vol. 2. Springer, 1984.
- [147] S. Bobba, P.-E. Gaillardon, J. Zhang, M. De Marchi, D. Sacchetto, Y. Leblebici, and G. De Micheli, “Process/design co-optimization of regular logic tiles for double-gate silicon nanowire transistors,” 2012.
- [148] P.-E. D. M. G. Amaru. L., Gaillardon, “A new canonical bdd for logic synthesis targeting ambipolar transistors,” 2013.
- [149] “An efficient logic synthesis methodology for mixed xor-and/or dominated circuits,” 2013.
- [150] P.-E. Gaillardon, M. H. Ben-Jamaa, F. Clermidy, and I. O’Connor, “Ultra-fine grain fpgas: A granularity study,” in *Nanoscale Architectures (NANOARCH), 2011 IEEE/ACM International Symposium on*, pp. 9–15, IEEE, 2011.
- [151] M. Horowitz, E. Alon, D. Patil, S. Naffziger, R. Kumar, and K. Bernstein, “Scaling, power, and the future of cmos,” in *Electron Devices Meeting, 2005. IEDM Technical Digest. IEEE International*, pp. 7–pp, IEEE, 2005.
- [152] H. Wong, S. Mitra, D. Akinwande, C. Beasley, Y. Chai, H. Chen, X. Chen, G. Close, J. Deng, A. Hazeghi, *et al.*, “Carbon nanotube electronics-materials, devices, circuits, design, modeling, and performance projection,” in *Electron Devices Meeting (IEDM), 2011 IEEE International*, pp. 23–1, IEEE, 2011.
- [153] Y.-M. Lin, J. Appenzeller, J. Knoch, and P. Avouris, “High-performance carbon nanotube field-effect transistor with tunable polarities,” *Nanotechnology, IEEE Transactions on*, vol. 4, no. 5, pp. 481–489, 2005.

- [154] P. Avouris, Z. Chen, and V. Perebeinos, “Carbon-based electronics,” *Nature Nanotechnology*, vol. 2, no. 10, pp. 605–615, 2007.
- [155] A. Raychowdhury, A. Keshavarzi, J. Kurtin, V. De, and K. Roy, “Carbon nanotube field-effect transistors for high-performance digital circuits—dc analysis and modeling toward optimum transistor structure,” *Electron Devices, IEEE Transactions on*, vol. 53, no. 11, pp. 2711–2717, 2006.
- [156] J. Deng, N. Patil, K. Ryu, A. Badmaev, C. Zhou, S. Mitra, and H.-S. Wong, “Carbon nanotube transistor circuits: Circuit-level performance benchmarking and design options for living with imperfections,” in *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pp. 70–588, IEEE, 2007.
- [157] N. Patil, A. Lin, J. Zhang, H. Wong, and S. Mitra, “Digital vlsi logic technology using carbon nanotube fets: Frequently asked questions,” in *Design Automation Conference, 2009. DAC’09. 46th ACM/IEEE*, pp. 304–309, IEEE, 2009.
- [158] L. Wei, D. J. Frank, L. Chang, and H.-S. Wong, “A non-iterative compact model for carbon nanotube fets incorporating source exhaustion effects,” in *Electron Devices Meeting (IEDM), 2009 IEEE International*, pp. 1–4, IEEE, 2009.
- [159] A. Franklin, M. Luisier, S. Han, G. Tulevski, C. Breslin, L. Gignac, M. Lundstrom, and W. Haensch, “Sub-10 nm carbon nanotube transistor,” *Nano letters*, vol. 12, no. 2, pp. 758–762, 2012.
- [160] J. Deng and H.-S. Wong, “A compact spice model for carbon-nanotube field-effect transistors including nonidealities and its application —part i: Model of the intrinsic channel region,” *Electron Devices, IEEE Transactions on*, vol. 54, pp. 3186–3194, dec. 2007.
- [161] S. W. Hong, T. Banks, and J. A. Rogers, “Improved density in aligned arrays of single-walled carbon nanotubes by sequential chemical vapor deposition on quartz,” *Advanced materials*, vol. 22, no. 16, pp. 1826–1830, 2010.
- [162] M. M. Shulaker, H. Wei, N. Patil, J. Provine, H.-Y. Chen, H.-S. Wong, and S. Mitra, “Linear increases in carbon nanotube density through multiple transfer technique,” *Nano letters*, vol. 11, no. 5, pp. 1881–1886, 2011.
- [163] J. Zhang, N. P. Patil, and S. Mitra, “Probabilistic analysis and design of metallic-carbon-nanotube-tolerant digital logic circuits,” *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 28, no. 9, pp. 1307–1320, 2009.

- [164] R. Saito, G. Dresselhaus, M. S. Dresselhaus, *et al.*, *Physical properties of carbon nanotubes*, vol. 35. World Scientific, 1998.
- [165] Y. Li, D. Mann, M. Rolandi, W. Kim, A. Ural, S. Hung, A. Javey, J. Cao, D. Wang, E. Yenilmez, *et al.*, “Preferential growth of semi-conducting single-walled carbon nanotubes by a plasma enhanced cvd method,” *Nano Letters*, vol. 4, no. 2, pp. 317–321, 2004.
- [166] L. Qu, F. Du, and L. Dai, “Preferential syntheses of semiconducting vertically aligned single-walled carbon nanotubes for direct use in fets,” *Nano letters*, vol. 8, no. 9, pp. 2682–2687, 2008.
- [167] G. Zhang, P. Qi, X. Wang, Y. Lu, X. Li, R. Tu, S. Bangsaruntip, D. Mann, L. Zhang, and H. Dai, “Selective etching of metallic carbon nanotubes by gas-phase reaction,” *Science*, vol. 314, no. 5801, pp. 974–977, 2006.
- [168] N. Patil, A. Lin, J. Zhang, H. Wei, K. Anderson, H. Wong, and S. Mitra, “Vmr: Vlsi-compatible metallic carbon nanotube removal for imperfection-immune cascaded multi-stage digital logic circuits using carbon nanotube fets,” in *Electron Devices Meeting (IEDM), 2009 IEEE International*, pp. 1–4, IEEE, 2009.
- [169] R. Ashraf, R. K. Nain, M. Chrzanowska-Jeske, and S. G. Narendra, “Design methodology for carbon nanotube based circuits in the presence of metallic tubes,” in *Nanoscale Architectures (NANOARCH), 2010 IEEE/ACM International Symposium on*, pp. 71–76, IEEE, 2010.
- [170] A. Lin, N. Patil, H. Wei, S. Mitra, and H. Wong, “Accnt—a metallic-cnt-tolerant design methodology for carbon-nanotube vlsi: Concepts and experimental demonstration,” *Electron Devices, IEEE Transactions on*, vol. 56, no. 12, pp. 2969–2978, 2009.
- [171] P. G. Collins, M. S. Arnold, and P. Avouris, “Engineering carbon nanotubes and nanotube circuits using electrical breakdown,” *Science*, vol. 292, no. 5517, pp. 706–709, 2001.
- [172] C. Kocabas, S. J. Kang, T. Ozel, M. Shim, and J. A. Rogers, “Improved synthesis of aligned arrays of single-walled carbon nanotubes and their implementation in thin film type transistors,” *The Journal of Physical Chemistry C*, vol. 111, no. 48, pp. 17879–17886, 2007.
- [173] N. Patil, A. Lin, E. R. Myers, K. Ryu, A. Badmaev, C. Zhou, H.-S. Wong, and S. Mitra, “Wafer-scale growth and transfer of aligned single-walled carbon nanotubes,” *Nanotechnology, IEEE Transactions on*, vol. 8, no. 4, pp. 498–504, 2009.

- [174] N. Patil, J. Deng, A. Lin, H. Wong, and S. Mitra, "Design methods for misaligned and mispositioned carbon-nanotube immune circuits," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 27, no. 10, pp. 1725–1736, 2008.
- [175] S. Bobba, J. Zhang, A. Pullini, D. Atienza, and G. De Micheli, "Design of compact imperfection-immune cnfet layouts for standard-cell-based logic synthesis," in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 616–621, European Design and Automation Association, 2009.
- [176] H. Ago, S. Imamura, T. Okazaki, T. Saito, M. Yumura, and M. Tsuji, "Cvd growth of single-walled carbon nanotubes with narrow diameter distribution over fe/mgo catalyst and their fluorescence spectroscopy," *The Journal of Physical Chemistry B*, vol. 109, no. 20, pp. 10035–10041, 2005.
- [177] B. C. Paul, S. Fujita, M. Okajima, T. H. Lee, H.-S. Wong, and Y. Nishi, "Impact of a process variation on nanowire and nanotube device performance," *Electron Devices, IEEE Transactions on*, vol. 54, no. 9, pp. 2369–2376, 2007.
- [178] S. Y. Borkar, A. Keshavarzi, J. K. Kurtin, and V. K. De, "Statistical circuit design with carbon nanotubes," Dec. 29 2005. US Patent App. 11/323,369.
- [179] A. Raychowdhury, V. K. De, J. Kurtin, S. Y. Borkar, K. Roy, and A. Keshavarzi, "Variation tolerance in a multichannel carbon-nanotube transistor for high-speed digital circuits," *Electron Devices, IEEE Transactions on*, vol. 56, no. 3, pp. 383–392, 2009.
- [180] J. Zhang, S. Bobba, N. Patil, A. Lin, H. Wong, G. De Micheli, and S. Mitra, "Carbon nanotube correlation: promising opportunity for cnfet circuit yield enhancement," in *Proceedings of the 47th Design Automation Conference*, pp. 889–892, ACM, 2010.
- [181] J. Zhang, A. Lin, N. Patil, H. Wei, L. Wei, H. Wong, and S. Mitra, "Carbon nanotube robust digital vlsi," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 31, no. 4, pp. 453–471, 2012.
- [182] J. Zhang, N. Patil, and S. Mitra, "Probabilistic analysis and design of metallic-carbon-nanotube-tolerant digital logic circuits," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 28, no. 9, pp. 1307–1320, 2009.
- [183] J. Deng and H. Wong, "A compact spice model for carbon-nanotube field-effect transistors including nonidealities and its application—part

- ii: full device model and circuit performance benchmarking,” *Electron Devices, IEEE Transactions on*, vol. 54, no. 12, pp. 3195–3205, 2007.
- [184] J. Roy and I. Markov, “Seeing the forest and the trees: Steiner wirelength optimization in placement,” *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 4, pp. 632–644, 2007.
- [185] C. Chu, “Flute: fast lookup table based wirelength estimation technique,” in *Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design*, pp. 696–701, IEEE Computer Society, 2004.
- [186] C. Mead and L. Conway, “Introduction to vlsi systems,” *Textbook in preparation*, 1978.
- [187] J. Zhang, N. Patil, A. Hazeghi, and S. Mitra, “Carbon nanotube circuits in the presence of carbon nanotube density variations,” in *Design Automation Conference, 2009. DAC’09. 46th ACM/IEEE*, pp. 71–76, IEEE, 2009.
- [188] S. J. Kang, C. Kocabas, T. Ozel, M. Shim, N. Pimparkar, M. A. Alam, S. V. Rotkin, and J. A. Rogers, “High-performance electronics using dense, perfectly aligned arrays of single-walled carbon nanotubes,” *Nature Nanotechnology*, vol. 2, no. 4, pp. 230–236, 2007.
- [189] J. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital integrated circuits*, vol. 996. Prentice-Hall, 1996.
- [190] “Stanford cnfet model.”
- [191] N. Srivastava and K. Banerjee, “Performance analysis of carbon nanotube interconnects for vlsi applications,” in *Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on*, pp. 383–390, IEEE, 2005.

List of Figures

1.1	Moore's Law: CPU transistor count has increased by 2X and feature size has decreased by 0.7X every two years.	3
1.2	More than Moore. "Whereas More Moore may be viewed as the brain of an intelligent compact system, 'More-than-Moore' refers to its capabilities to interact with the outside world and the users." [11]	5
1.3	Illustration of straining of silicon by means of silicon germanium.	6
1.4	(a) Ultrathin body SOI MOSFET (b) Double-gate SOI MOSFET.	7
1.5	Types of multiple gate architectures [17]	8
1.6	Concept drawing of vertically stacked gate-all-around silicon nanowire field effect transistor. [24]	9
1.7	Single layer Molybdenite (MoS_2) transistor. [27]	11
1.8	Schematic honeycomb structure of a graphene sheet. Carbon atoms are at the vertices. SWNTs can be formed by folding the sheet along lattice vectors. The two basis vectors a_1 and a_2 , and several examples of the lattice vectors are shown.	12
1.9	(a) Armchair, (b,c) zig-zag and (d) chiral tube; (a) metallic, (b) small gap semiconductor, and (c,d) semiconductor. [34]	13
1.10	(a) Schematic of a SET device (b) Single electron tunneling based on Coulomb blockade.	14
1.11	(a) 3D TSV integration, (b) 3D Monolithic integration	15
1.12	Abstraction Levels of the (CMOS) Design Process (left) and the appropriate tools (right). [44]	16
2.1	Coarse-grain to fine-grain circuit partitioning for 3D circuits [46] (a) Memory/Core on a core, (b) Functional unit blocks on top of each other, (c) Logic gates distributed across different layers, and (d) Transistor scale partitioning.	24
2.2	Cross-section of a 3D monolithic die with two active layers.	25

2.3	Main references related to monolithic 3D integration before 1993. Chen83 [51], Shah84 [52], Gibbons80 [53], Gibbons82 [54], Goeloe81 [55], Colinge81 [56], Kawamura83 [57], Kawamura84 [58], Kawamura87 [59], Ohtake86 [60], Zingg89 [61], Takao91 [62], Takao92 [63], Roos92 [64], Roos93 [65].	26
2.4	Main references of 3D monolithic integration since 2000. Subramanian98 [67], Chan01 [68], Tiwari02 [69], Zhang04 [70], Yu05 [71], Wu05 [72], Feng06 [73], Mofrad08 [74], Batude09b [40], Batude09a [47], Batude11 [75], Kang04 [76], Jung05 [77], Jung06 [78], Sohn06 [79], Jung07 [80], Son07 [81], Sohn08 [82].	27
2.5	SEM cross-section of stacked transistors with LG=50nm and ultra thin interlayer dielectric TILD=23nm, TSi=10nm (morphological structure). Inverter transfer voltage characteristic with pFET stacked over nFET (LG,P=LG,N=50nm).[34]	28
2.6	Monolithic 3D fabrication.	29
2.7	Sheet resistance of NiSi and NiSi + Pt + F + W as a function of the annealing time at 650°C. The sheet resistance of NiSi alone exhibits a dramatic increase as soon as 1 minute annealing is performed. Stabilized salicide does not show any change neither in morphology nor in electrical properties for annealing for as long as 40min. [86]	33
2.8	C(V) characteristics of bottom and top FETs having the same HfO ₂ /TiN gate stack but respectively processed at 1050°C and 600°C. Red curves correspond to the bottom transistors processed at regular high temperature. Blue curves correspond to the top FETs processed in a cold process. [47]	34
2.9	Drain current as a function of gate voltage for transistors stacked on the same wafers. Red curves correspond to the bottom transistors processed at regular high temperature. Blue curves correspond to the top FETs processed in a cold process. [47]	34
2.10	3D contacts for 3DMI technology. Monolayers contacts land either on the top or on the bottom layer whereas multilayers contacts land on several layers. In the case of “thru-layer contact” the contact drills the top layer and further digs into the ILD until reaching the bottom layer whereas in the case of the “strapping contact” a highly selective etching allows lying on both layers at the same time.	35
2.11	Contact area as a function of contact diameter for planar contact (corresponding to monolayer contact), half planar contact (corresponding to “strapping contact”) or lateral contact (corresponding to “thru contact”).	35
2.12	Example of a standard cell illustrating the height and width of the cell.	37

2.13	(a) Typical cell in 2D configuration (b) <i>intra-cell</i> transformation, in two active layers, by realizing pull-up network on the top layer and pull-down network at the bottom layer (c) Cross-sectional view of the two active layers with the metals (IM and M1) for realizing PUN and PDN of the cell.	38
2.14	(a) Two input NAND cell with a high drive strength having finger transistors (b) corresponding cell built in 3D with <i>intra-cell</i> folding transformation where the fingers are realized in the bottom/top active layer.	39
2.15	CELONCEL NAND2 layout, (a) cell realized in the top active layer (b) corresponding cell in the bottom active layer, and (c) Cross-sectional view of the two active layers with the metals (IM and M1) to realize the cells.	40
2.16	2D to 3D Cell Transformation.	42
2.17	Neighboring cells in the case of planar, <i>intra-cell</i> and <i>cell-on-cell</i>	43
3.1	Logic-to-Layout Design Flow for (a) <i>cell-on-cell</i> and (b) <i>intra-cell</i> transformations.	48
3.2	DEFLATE transformation applied to all the library cells.	49
3.3	INFLATE transformation shown for neighboring standard cell rows (<i>i</i> and <i>j</i>). The width of the cells is doubled, while keeping their centers (e.g. <i>O1</i> , <i>O2</i> and <i>O3</i>) fixed. Morphing the cell width leads to overlaps and whitespace between the cells.	50
3.4	Active layer assignment shown for neighboring standard cell rows (<i>i</i> and <i>j</i>). Overlap between the cells is removed by assigning the cells to different active layers with the help of the ZOLP formulation. Whitespace between the cells helps in forming small clusters to speed up the ILP.	51
3.5	Percentage improvement in the total area for all the cases.	58
3.6	Performance improvement in wirelength of various benchmark circuits when subjected to <i>wirelength-driven</i> placement.	59
3.7	Performance improvement in wirelength of various benchmark circuits when subjected to <i>timing-driven</i> placement.	60
3.8	Performance improvement of various benchmark circuits when subjected to <i>timing-driven</i> placement.	62
3.9	Delay reduction of a LDPC decoder with <i>in-place</i> optimization.	63
4.1	3D TSV Integration [102].	66
4.2	(a) Transistor stacking with 3D monolithic. (b) Potential 3.5-D Integration.	67
4.3	Example of a 3D MPSoC with three stacked layers [108].	68
4.4	Technology mapping from planar 2D to 3D TSV, 3DMI n/p and novel 3.5D integrations.	69
4.5	Fully homogeneous processor array : GENEPI v1 [111]	70
4.6	2D and 3D Test flows [113].	72

4.7	(a) Typical standard cell in 2D (planar) configuration and (b) Standard cell designed in 3DMI n/p by realizing the PUN on the top active layer and the PDN in the bottom active layer.	73
4.8	Performance improvement in terms of delay and power of various blocks of the core.	74
4.9	Performance improvement in terms of delay and power of various blocks of the core.	75
5.1	(a) FinFET providing increase in controllable channel area between the source and drain regions (b) Vertically-stacked SiNWFET with multiple parallel nanowire channels, each with Gate-All-Around control (c) Double-Gate SiNWFET with control and polarity gates.	78
5.2	Conceptual structure of the ambipolar DG- SiNWFET: a) 3D view of the device. b) Top view of the device showing one stack of nanowires forming the channel.	79
5.3	Conceptual structure of the ambipolar DG- SiNWFET: a) 3D view of the device. b) Top view of the device showing one stack of nanowires forming the channel.	80
5.4	Double-gate SiNWFET (a) Layout (top view). (b) Symbol of an ambipolar FET (c) Configuration as n-type and p-type by setting the PG.	80
5.5	Negative unate logic function (a) NAND gate (b) NOR gate implementation by swapping the V_{dd} and G_{nd} of a NAND gate (a).	83
5.6	Positive unate logic function (a) OR gate implementation by interchanging the voltage of the PGs in the PUN and PDN. (b) AND (positive unate) gate implemented with NAND (negative unate) gate followed by Inverter. (c) AND gate implemented by applying De Morgans rule.	83
5.7	2-input logic gates (a) NAND gate with PGs connected to V_{dd} and G_{nd} (b) XOR gate with PGs connected to input signals (B or $\overline{B}$) (c) XOR gate with B assigned to logic 1.	84
5.8	Mixed function $Y = \overline{(A \oplus B)C}$ (a) Ambipolar logic style, where the PGs of the binate logic are connected to logic inputs (B or $\overline{B}$) and PGs of unate variables are connected to V_{dd} and G_{nd} (b) Static CMOS implementation.	85
5.9	(a) A top view of the DG-SiNWFET shown in Fig. 5.2. (b) Large transistor. (c) Equivalent dumbbell-stick diagram. (d) Dumbbell-stick diagram of an Inverter with a transistor pair. (e) Grouping transistor with similar polarity gates.	86
5.10	Dumbbell-stick diagram for 2-input NAND gate with the PGs grouped together in the PUN (and PDN) and connected to G_{ND} (and V_{DD}).	87

5.11	Dumbbell-stick diagram for 2-input XOR gate – (case-1) conventional approach by placing the transistors in the pull-up (pull-down) together so that they share the diffusion contacts (case-2) efficient layout technique where the transistors are grouped together, irrespective if they are located in the pull-up or pull-down networks, as well as share the same diffusion contacts.	88
5.12	Transistor ordering for XNUMixed logic function.	89
5.13	Logic-to-layout procedure (a) Complex logic function. (b) Separate bipartite graph representation for binate and unate part of the logic. (c) Search tree of unate-bipartite graph in (b).	90
5.14	The matrix representation of the nodes in Fig. 5.13c.	92
5.15	(a) Graphical representation of transistor chains derived from Fig. 5.13c. (b) Dumbbell-stick diagram of the circuit.	93
5.16	Reconfigurable logic block based on ambipolar logic style [132]. .	94
5.17	Carry-out function of a full-adder.	94
5.18	The schematic of the ambipolar silicon nanowire used in TCAD. .	96
5.19	Band diagram of the SiNWFET.	96
5.20	Symmetric characteristics of ambipolar SiNWFET from TCAD simulation.	97
5.21	Single NWFET equivalent circuit.	97
5.22	Full-adder implementation with DG-SiNW transistors and conventional CMOS transistors.	99
5.23	Average performance improvement for various components of data path circuits.	100
6.1	<i>Sea-of-Tiles</i> (SoT) design methodology.	104
6.2	Dumbbell-stick diagrams of various logic tiles considered for SoTs (a) Tile _{G1} (b) Tile _{G2} (c) Tile _{G1h2} (d) Tile _{G3}	106
6.3	Design flow for finding the best Tile for SoT.	108
6.4	Matching compatibility graph for 3-input Boolean space.	110
6.5	Mapping of 3-input OR function (F_1).	111
6.6	Mapping of 3-input NPN equivalent functions with embedded XOR/XNOR (F_3 , F_7 , F_{12} and F_{13}).	112
6.7	Reconfigurable fabrics mapped on to SoT with Tile _{G2} (a) Regular computation fabric [132] (b) Universal logic module (3,2-ULM) [131].	113
6.8	A Full-adder mapped on to a Sea-of-Tiles with the hybrid tile Tile _{G1h2} as the basic building block.	114
6.9	Design flow for sizing the tiles.	115
6.10	Delay characteristics of an Inverter, driving a constant load, with varying stack size.	115
6.11	Critical path delay of various benchmark circuits when mapped with DGSiNWFET and CMOS technologies.	117
6.12	Combinational area of various benchmark circuits when mapped with DGSiNWFET and CMOS technologies.	118

6.13	Leakage power of various benchmark circuits when mapped with DG-SiNWFET and CMOS technologies.	119
6.14	Leakage power of various benchmark circuits when mapped with DGSiNWFET with varying stack size.	120
6.15	Normalized area comparing DG-SiNWFET with stack size of 6 nanowires, with equivalent CMOS implementation.	121
6.16	Normalized delay comparing DG-SiNWFET, with various stack size, with equivalent CMOS implementation.	122
7.1	(a) CNFET structure. (b) Top view of CNFET.	126
7.2	(a, c) Top view of an inverter with CNFETs having the same CNTs (referred to as correlated CNFETs); (b, d) Top view of an inverter with un-correlated CNFETs; (a, b) Ideal CNT; (c, d) CNT count variation	130
7.3	CNFET failure probability vs. CNFET width ($p_{Rm} = 1$). [180] . .	132
7.4	Case study: (a) Transistor width distribution of an OpenRISC core synthesized using Nangate 45nm Cell Library. (b) Gate capacitance increase (penalty) vs. technology node associated with upsizing the small transistors to W_{min} . [180]	133
7.5	a) Non-aligned layout style on uncorrelated CNT growth. (b) Non-aligned layout style on directional CNT growth. (c) Aligned-active layout style on directional CNT growth. [180]	134
7.6	Logic errors caused by mispositioned-CNTs. [174]	136
7.7	Mispositioned-CNT immune layout [174].	137
7.8	Misaligned-CNT-immune layout based on Euler paths. (a) One Euler path for each PUN (red line) and PDN (blue line). (b) One Euler path for the entire schematic (dotted line).	138
7.9	NAND3 gate. (a) Circuit schematic. (b, c, d) Mispositioned-CNT-immune layouts.	139
7.10	Enforcing aligned-active layout style to the AOI222_X1 cell from the Nangate 45nm Open Cell Library.	142
7.11	Enforcing aligned-active layout style to the DFFS_X2 cell from the Nangate 45nm Open Cell Library.	143
7.12	Design methodology.	144
7.13	Design flow.	146
7.14	Critical path delay improvement of CNFET circuits when compared to CMOS circuits.	148
7.15	Dynamic power improvement of CNFET circuits when compared to CMOS circuits.	148
7.16	EDP of various benchmarks with CNFETs when compared to their implementation with planar CMOS technology.	149
7.17	Maximum frequency improvement with CNFETs when compared to CMOS technology.	150
7.18	OpenRISC 1200 casestudy at various technology nodes. (a) Maximal frequency achievable. (b) Energy-delay-product.	150

List of Acronyms

Acronym	Definition
1D	One Dimension
3D	Three Dimension
3DMI	3D Monolithic Integration
AAG	Aligned-Active Grids
AOI	And-Or-Inv
BDD	Binary Decision Diagram
CAD	Computer Aided Design
CG	Control Gate
CNFET	CNT Field Effect Transistor
CNT	Carbon Nanotube
DG	Double Gate
DG-SiNWFET	Double-Gate SiNWFET
DIBL	Drain-Induced Barrier Lowering
DSD	Dumbell-Stick Diagram
EDA	Electronic Design Automation
EDP	Energy-Delay Product
ELO	Epitaxial Lateral Overgrowth
FDSOI	Fully-Depleted Silicon on Insulator
FET	Field Effect Transistor
GAA	Gate-All-Around
IC	Integrated Circuit
ICR	Intra-Cell Routing
ICRA	Intra-Cell Routing Area
ILD	Inter Layer Dielectric
IO	Input-Output
ITRS	International Technology Roadmap for Semiconductors
LEG	Laser Epitaxial Growth
m-CNT	metallic-CNT
MILC	Metal Induced Lateral Crystallization
MPSoC	Multi-Processor System-on-Chip
MtM	More than Moore
MuGFET	Multiple Gate Field Effect Transistor

MWNT	Multi-Walled Nanotubes
NoC	Network-on-Chip
PDN	Pull-Down-Network
PE	Processing Elements
PG	Polarity Gate
PUN	Pull-Up-Network
s-CNT	semiconducting-CNT
S3	Single-crystal Si layer Stacking
SCE	Short Channel Effects
SEM	Scanning Electron Microscopy
SET	Single-Electron Transistors
SiNW	Silicon Nanowire
SiNWFET	Silicon Nanowire Field Effect Transistor
SoC	System-on-Chip
SoT	Sea-of-Tiles
SPE	Solid Phase Epitaxy
SWNT	Single-Walled Nanotubes
TSV	Through-Silicon-Vias
UTB SOI	Ultrathin Body Silicon on Insulator

CURRICULUM VITEA

Shashikanth Bobba

email: shashikanth.bobba@epfl.ch;

LinkedIn: <http://www.linkedin.com/in/shashibobba>

Snail-mail: Chemin Des Cotes -4; Renens 1020; Switzerland

Work: +41216930920, Mobile: +41787748887



PROFILE

- Strong research experience in design methodologies and CAD for CMOS and Emerging nanotechnologies.
- Experience in development and usage of commercial EDA tools for front-end/back-end design with 45nm to 16nm technologies both under commercial and predictive models.
- Worked on various aspects of IC design, spanning from modeling of on-chip transformers, design of network-on-chip interfaces for asynchronous and GALS design, to physical design techniques for 3D integration and emerging nanotechnologies.
- Broad knowledge on design techniques bridging process, design, and CAD for early performance evaluation of emerging nanotechnologies.
- Good team player with a broad international work experience in semiconductor industry/research labs across seven countries (Belgium, France, Italy, India, Sweden, Switzerland, and USA).
- Carried out collaboration with various research labs and industries resulting in 3 Patents and more than 20 publications in reputed international proceedings and journals, including a *Best paper award* at IEEE/ACM NanoArch Symposium.

EDUCATION

EPFL (École Polytechnique Fédérale de Lausanne)	Lausanne, Switzerland
• PhD. Candidate, Institute of Electrical Engineering - Advisor: Prof. Giovanni De Micheli - PhD Thesis: "Design Methodologies and CAD for Emerging Nanotechnologies"	Jun 2013
Politecnico di Torino	Torino, Italy
• Post Graduate Masters in Wireless Systems	<i>Summa Cum Laude</i> Nov 2006
Lund University (LTH)	Lund, Sweden
• M.Sc in System-on-Chip design	<i>GPA 4.4/5</i> Jan 2005

PROFESSIONAL EXPERIENCE

Stanford University Palo Alto, CA, USA
Visiting Research Scholar at Robust Systems Laboratory (Prof. Subhasish Mitra) (Jun 2011 – Oct 2011)
Worked on physical design techniques for improving the yield of CNFET circuits. With the help of robust layout techniques, various standard cell libraries are developed and characterized to study system-level benchmarking of CNFET circuits.

CEA-LETI, France Grenoble, France
Visiting Research Scholar at Innovative Device Laboratory, Minatec (Oct 2009 – May 2010)
Worked on design methodologies for 3D monolithic integration. Developed novel standard cell transformation techniques and a placement tool (CELONCEL) for ultra-high dense 3D monolithic integrated circuits. The result of this research work has been patented [P1] and published at various international conferences [C6, C8, C9, J1].

EPFL Lausanne, Switzerland
IC Design Engineer at Integrated Systems Laboratory (May 2007 – Dec 2009)
FP7 EU Project: Globally Asynchronous Locally Synchronous (GALS) Network-on-Chips (GALAXY)
GALAXY aimed at employing GALS design methodology for realizing a Network-on-Chips (NoC) for a complex multimedia SoC with multiple voltage and frequency islands.

- Designed a pausable clock GALS wrapper for the X-Pipes NoC architecture. Port controllers employing Asynchronous Finite State Machines were designed using *petrify*. The final implementation of the pausable clock network initiator and network target can be plugged into asynchronous network without any metastability.
- Within the scope of this project, I was solely responsible for 3 deliverables and represented EPFL in three annual workshops in Newcastle (2007), Ferrara (2008) and Munich (2009).

Telecom Italia Labs

Hardware Designer

Designed a switched parasitic antenna for Zigbee motes, which involved extensive 3-D EM simulations using ANSYS HFSS tool.

Torino, Italy

(Nov 2006 – Feb 2007)

Agilent Technologies

EDA Software Engineer at Solution Services Group

Developed a tool in C++, which visualizes Surface-currents of passive components, thereby enriching the functionality of state-of-the-art EM/Circuit co-simulators. This tool has been integrated into the Agilent Momentum 2.5D simulator with which the designer can visualize the surface currents on the layouts of the passive components.

Ghent, Belgium

(Jul 2006 – Oct 2006)

Ericsson AB

RFIC Engineer at MMIC Design group

- Developed a design kit in ADS for generating layout macros for designing Marchund Baluns for MMICs.
- As a verification engineer, I carried out phase noise measurements of different VCO's for MMIC application.

Molndal, Sweden

(Feb 2005 – Jul 2005)

Ericsson Mobile Platforms (now ST-Ericsson)

Mixed Signal IC Designer at Circuit Design Group, Research Department

Worked on high-frequency modeling and design of on-chip inductors and transformers for RF applications. Developed a design kit in ADS for automatic layout generation of various geometries of inductors and transformers and analyzed them using EM solvers like Momentum and Fasthenry. Different high frequency models were developed for circuit simulations.

Lund, Sweden

(Feb 2004 – Dec 2004)

PATENTS

[P1] **S. Bobba** and O. Thomas, "Multi-level integrated circuit, Device and method for modelling multi-level integrated circuits". US 2012/0161329 A1, Jun. 28, 2012.

[P2] D. Sachetto, **S. Bobba**, P.-E. Giallardon, Y. Leblebici, and G. De Micheli, "Generic Memristive Structure and Complementary Resistive Programming". EP 12179317.8, 3 August 2012.

[P3] D. Sachetto, **S. Bobba**, P.-E. Giallardon, Y. Leblebici, and G. De Micheli, "TaOx/CrOy ReRAM element" EP 12179314.5.

INTERNATIONAL PUBLICATIONS

Conferences:

[C15] P.-E. Gaillardon, M. De Marchi, D. Sacchetto, **S. Bobba**, L. Amaru, Y. Leblebici, G. De Micheli, "Towards Structured ASICs using Polarity-Tunable Si Nanowire Transistors", *IEEE/ACM Design Automation Conference 2010*.

[C14] P.-E. Gaillardon, **S. Bobba**, L. Amaru, M. De Marchi, D. Sacchetto, Y. Leblebici, G. De Micheli, "Vertically Stacked Double Gate Nanowires FETs with Controllable Polarity: From Devices to Regular ASICs", *IEEE Design, Automation & Test in Europe*.

[C13] **S. Bobba**, P.-E. Giallardon, C. Seiculescu, V. Pavlidis, and G. De Micheli, "3.5-D Intergration: A Case study", *IEEE International Symposium on Circuits and Systems 2013*.

[C12] P.-E. Giallardon, D. Sachetto, **S. Bobba**, Y. Leblebici, and G. De Micheli, "GMS: Generic Memristive Structure Concept for 3-D FPGAs", *IEEE VLSI System on Chip 2012*.

[C11] **S. Bobba**, P.-E. Giallardon, J. Zhang, M. De. Marchi, D. Sachetto, Y. Leblebici, and G. De Micheli, "Process/Design Co-optimization of Regular Logic Tiles for Double-Gate Silicon Nanowire Transistors", *IEEE/ACM International Symposium on Nanoscale Architectures 2012*. **[BEST PAPER AWARD]**

- [C10] **S. Bobba**, M. De. Marchi, Y. Leblebici, and G. De Micheli, “Physical Synthesis onto Sea-of-Tiles with Double-gate Silicon Nanowire FET”, *IEEE/ACM Design Automation Conference 2012*.
- [C9] **S. Bobba**, A. Chakraborty, O. Thomas, P. Batude, T. Ernst, O. Faynot, D. Z. Pan, and G. De Micheli. “CELONCEL: Effective Design Technique for 3-D Monolithic Integration targeting High Performance Integrated Circuits”. In *Proceedings of the 16th Asia and South Pacific Design Automation Conference*.
- [C8] P. Batude, ..., **S. Bobba**, et. al. “Advances, Challenges and Opportunities in 3D CMOS Sequential Integration”. *IEEE International Electron Device Meeting*.
- [C7] M. De Marchi, **S. Bobba**, H. Ben Jamaa, and G. De Micheli. “Synthesis of regular computational fabrics with ambipolar CNTFET technology”. *17th IEEE International Conference on Electronics, Circuits and Systems*.
- [C6] **S. Bobba**, A. Chakraborty, O. Thomas, P. Batude, V. Pavlidis, and G. De Micheli. “Performance Analysis of 3-D Monolithic Integrated Circuits”. *IEEE International 3D System Integration Conference 2010*.
- [C5] J. Zhang, **S. Bobba**, N. Patil, A. Lin, H.-S.P. Wong, G. De Micheli and S. Mitra, “Carbon Nanotube Correlation: Promising Opportunity for CNFET Circuit Yield Enhancement,” *IEEE/ACM Design Automation Conference*.
- [C4] **S. Bobba**, S. Carrara, and G. De Micheli., “Design of a CNFET Array for Sensing and Control in P450 based Biochips for multiple drug detection”, *IEEE International Symposium on Circuits and Systems 2010*.
- [C3] **S. Bobba**, J. Zhang, A. Pullini, D. Atienza, and G. De Micheli, “Design of Compact Imperfection-Immune CNFET Layouts for Standard-Cell-Based Logic Synthesis” *IEEE Design, Automation & Test in Europe*.
- [C2] D. Atienza, **S. Bobba**, M. Poli, G. De Micheli, and L. Benini. “System-Level Design for Nano-Electronics”, *IEEE International Conference on Electronics, Circuits and Systems*.
- [C1] **S. Bobba**, A. Sathanur, T. Mattsson and S. Nilsson, “Design of a novel type of on-chip transformer suitable for baluns in customary BICMOS/CMOS technologies”, *IEEE European Conference on Circuit Theory and Design*.

Journals:

- [J1] **S. Bobba**, A. Chakraborty, O. Thomas, P. Batude, and G. De Micheli, “Novel Cell Transformation and Placement Technique for 3D Monolithic Integrated Circuits”. *ACM Journal on Emerging Technology in Computing Systems*, 2012.
- [J2] **S. Bobba** and G. De Micheli, “Layout Technique for Double-Gate Silicon Nanowire FETs with an efficient Sea-of-Tiles Architecture”. *IEEE Transaction on VLSI*, 2013.
- [J3] **S. Bobba**, J. Zhang, P-E. Gaillardon, H-S P. Wong, S. Mitra, and G. De Micheli, “System Level Benchmarking with Standard Cell Library Optimization for Carbon Nanotube VLSI Circuits”. *ACM Journal on Emerging Technology in Computing Systems*, 2013.
- [J4] P.-E. Gaillardon, L. G. Amarù, **S. Bobba**, M. De Marchi and D. Sacchetto et al. “Nanosystems: Technology and Design”, *Philosophical Transactions of the Royal Society of London*, 2013.

Book Chapters:

- [Chapter 1] M. Vinet, P. Batude, and **S. Bobba**. “3D Monolithic Integration”. In *Future Intelligent Integrated Systems: Advanced Silicon CMOS Technologies*. WSPC-Pan Stanford (Singapore), October 2011.
- [Chapter 2] D. Sacchetto, H. Ben Jamaa, **S. Bobba**, and F. Sun. “Emerging Interconnect Technologies”. In *Communication Architectures for Systems-On-Chip*. CRC Press Taylor & Francis Group, 6000 Broken Sound Parkway NW, 2011.

SKILLS

- Languages: C, C++, Python, VHDL, Verilog, SystemC, Tcl, shell scripting.
- Tools: Synopsys: Design compiler, Physical Compiler, PrimeTime, HSPICE. Cadence: SoC Encounter, Virtuoso. Mentor: Calibre, ModelSim.
- Other: MATLAB, LATEX, FastHenry, Momentum.

AWARDS AND ACTIVITIES

- **Best paper award** at the IEEE/ACM Nanoarch 2012 conference held in Amsterdam.
- Awarded “*Summa Cum Laude*” (Honors) for the PG Masters program at Politecnico di Torino.
- Awarded a scholarship (22000 Euros) by the Italian government to pursue Postgraduate Masters.
- Travel grant (1000\$) for DAC PhD Forum – (by SIGDA 2011).
- Reviewer for IEEE Transactions on Computer Aided Design (TCAD), Transactions on VLSI (TVLSI), Design Automation and Test in Europe (DATE), and Microelectronics Journal.

Extra-Curricular

- President of EPFL Toastmasters club (www.epfltoastmasters.com) for the year 2012-2013.
- Member of TEDxLausanne team (<http://tedxlausanne.org/>).
- Vice president of YUVA (Indian Student Association at EPFL) for the term 2011-2012.
- Co-founder of Marketing to Minds group (www.marketingtominds.com).
- Hobbies: Skiing, Badminton, Travelling, Photography, and Hiking.