Circuit Optimisation using Device Layout Motifs

Yang Xiao

Ph.D.

University of York
Electronics

July 2015

Abstract

Circuit designers face great challenges as CMOS devices continue to scale to nano dimensions, in particular, stochastic variability caused by the physical properties of transistors. Stochastic variability is an undesired and uncertain component caused by fundamental phenomena associated with device structure evolution, which cannot be avoided during the manufacturing process. In order to examine the problem of variability at atomic levels, the '*Motif*' concept, defined as a set of repeating patterns of fundamental geometrical forms used as design units, is proposed to capture the presence of statistical variability and improve the device/circuit layout regularity. A set of 3D motifs with stochastic variability are investigated and performed by technology computer aided design simulations.

The statistical motifs compact model is used to bridge between device technology and circuit design. The statistical variability information is transferred into motifs' compact model in order to facilitate variation-aware circuit designs. The uniform motif compact model extraction is performed by a novel two-step evolutionary algorithm. The proposed extraction method overcomes the drawbacks of conventional extraction methods of poor convergence without good initial conditions and the difficulty of simulating multi-objective optimisations. After uniform motif compact models are obtained, the statistical variability information is injected into these compact models to generate the final motif statistical variability model.

The thesis also considers the influence of different choices of motif for each device on circuit performance and its statistical variability characteristics. A set of basic logic gates is constructed using different motif choices. Results show that circuit performance and variability mitigation can benefit from specific motif permutations. A multi-stage optimisation methodology is introduced, in which the processes of optimisation are divided into several stages. Benchmark circuits show the efficacy of the proposed methods. The results presented in this thesis indicate that the proposed methods are able to provide circuit performance improvements and are able to create circuits that are more robust against variability.

Contents

List of Figures

List of Tables

Acknowledgments

I would like to take the opportunity to express my gratitude towards many people who have assisted me in this work. Firstly, I must give the greatest of thanks to my supervisor, Prof. Andy Tyrrell and co-supervisor, Dr. Martin Trefzer, for their valuable ideas, countless suggestions, support and encouragements throughout the last four years. Particularly, he and Martin devoted a lot of time and patience to the reading and correction of this thesis. I also must thank my colleagues and mentors Dr. James A. Walker and Dr. Simon Bale, for their fruitful suggestions, discussions and progress of my research. I also would like thank to Dr. Scott Roy for his valuable advice and ideas on device research, discussions, support and for helping with my publications reading and correcting.

I would then like to thank everybody else on the PAnDA project and everybody in the Intelligent Systems Group at University of York, for their knowledge and support. Last but not least, I want to thank my family, especially my dad and mum, who have always been there, without their endless support, I could not have finished this.

This work would not have been possible without funding from the EPSRC PAnDA project.

Declaration

The work presented in this thesis is the author's own work, unless it is stated otherwise. The following items have been previously published during this research.

- Y. Xiao, M. A. Trefzer, J. A. Walker, S. J. Bale and A. M. Tyrrell, Two Step Evolution Strategy for Device Motif BSIM Model Parameter Extraction, *15th IEEE Congress on Evolutionary Computation (CEC14)*, pp. 2877-2884, Beijing, China, Jul, 2014.
- Y. Xiao, M. A. Trefzer, S. Roy, J. A. Walker, S. J. Bale and A. M. Tyrrell, Circuit Optimization using Device Layout Motifs, *5th European Workshop on CMOS Variability (VARI)*, pp.1 - 6, Palma, Spain, Sep, 2014.
- Y. Xiao, J. A. Walker, S. J. Bale, M. A. Terfzer and A. M. Tyrrell, Circuit Design Optimisation Using a Modified Genetic Algorithm and Device Layout Motifs, *11th IEEE International Conference on Evolvable Systems (ICES)*, pp.1-8, Orlando, USA, Dec, 2014.
- Y. Xiao, S. Roy, S. J. Bale, J. A. Walker, M. A. Terfzer and A. M. Tyrrell, Variation-aware Circuit Performance Optimisation Using Multi-Stages Evolutionary Algorithm and Motifs, *IEEE Transactions on VLSI* (Under review).

Hypothesis

Circuit performance improvement and variability mitigation can be achieved through a combination of evolutionary algorithms and device layout motifs.

Chapter 1

Introduction

Contents

1.1 Motivation

Since the first integrated circuit flip-flop with two transistors was built by Jack Kilby in 1958 at Texas Instruments, there has been an unprecedented growth of the semiconductor industry. In 1971, Intel's 4004 microprocessor integrated 2300 transistors [1]. This number increased to 5 billion transistors in Microsoft's Xbox One Main SoC in 2013 [2]. On the economic side, the semiconductor industry business has grown worldwide in sales from $1 billion in 1970 to $250 billion in 2007, and has reached $335.8 billion in 2014 [3]. The driving force behind this incredible growth comes from continuous miniaturisation of transistors and improvements in manufacturing techniques (Figure 1.1 shows that microprocessors' evolution obeys Moore's law [4]). As transistor feature size gets smaller, device become faster, have higher packing density, lower power dissipation and are cheaper to manufacture. However, as process technology scales down beyond *100-nm* feature size, the traditional VLSI design approach needs to be modified to cope with the new challenges arising related to process variation, interconnect processing difficulties, and other newly exacerbated physical effects.

Many circuit designers spend much of their effort specifying function with hardware description languages and seldom look at actual transistor characteristics. This results in the isolation of device engineers from circuit design engineers who lack an understanding of the

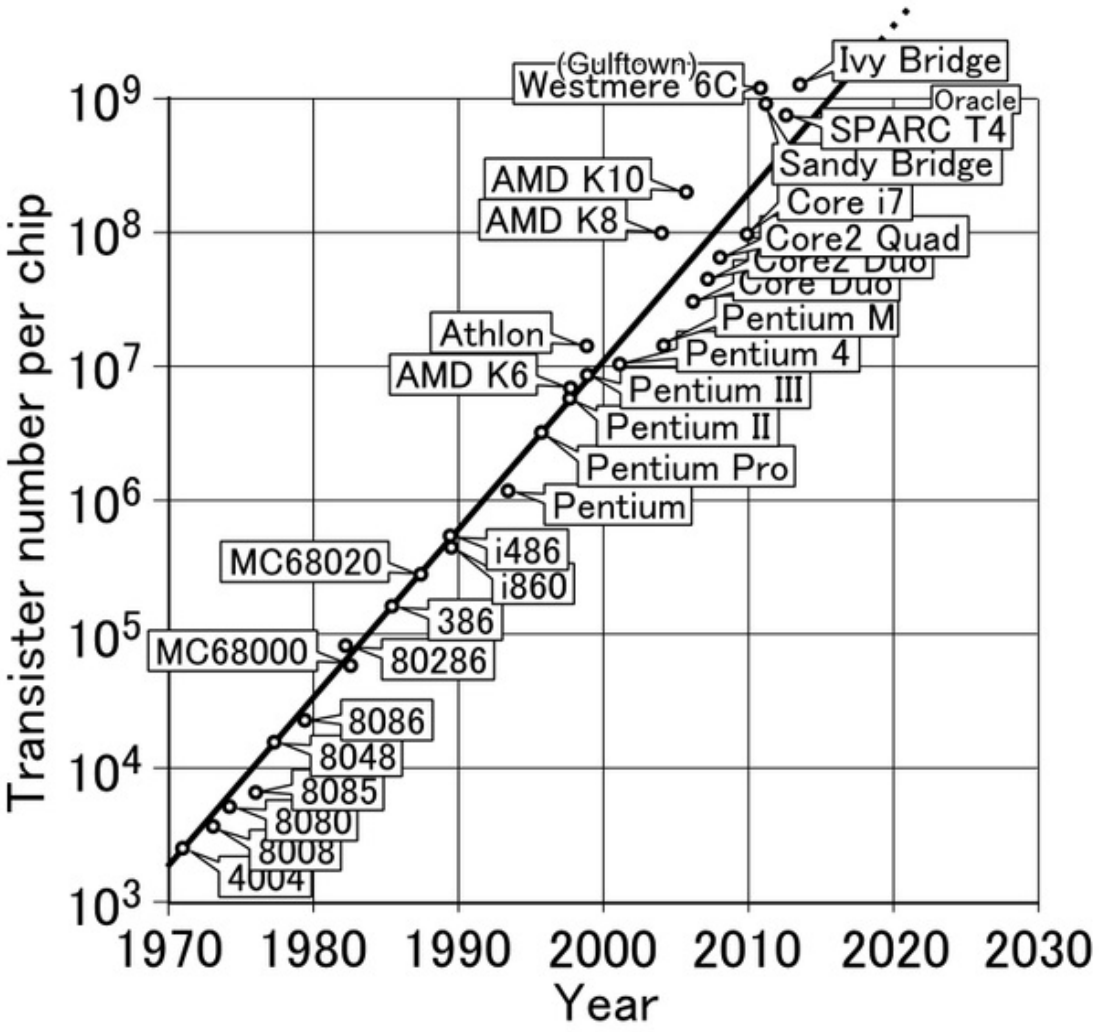


Figure 1.1: A comparison of the transistor counts in various microprocessors, with a curve showing the projected counts following Moore's predictions of a doubling every 24 months [4].

impact of design upon performance, yield, and manufacturability owing to the fundamental limitations of technology and device physics at the nanometer scale. Thus, it is necessary to propose practical guidelines and methodologies to bridge from physical processing to circuit design to build an understanding of integrated circuits from the bottom up, helping circuit designers to avoid some of the pitfalls due to these technology constraints.

This thesis is concerned with these issues and considers a novel concept based on, '*motif*', fundamental geometrical forms that uses these as design units to investigate device/circuit performance and its variability characteristic. The proposed methodology optimises circuits' performance through automatic optimisation of circuits constructed using motif permutations and adjustment of motif geometry. A set of motifs are proposed that provide different layout

choices for transistor. Evolutionary algorithms are automatically selected appropriate motif choices for each transistor in the circuit. The whole circuit layout can be seen as different motifs' combination. Various motif permutations lead to the difference in circuit performance. Additionally, motifs' geometry sizes are adjusted by evolutionary algorithms in order to further improve circuit performance and decrease the influence of device variability on circuit performance.

1.2 Aims and Objectives

The aim of this thesis is to propose a methodology to optimise circuit performance and examine the problem of variability at atomic levels. In order to achieve this aim, the thesis has the following objectives:

- To propose various novel motifs based on fundamental geometrical form and investigate their characteristic through 3D device layout structure modelling, and atomistic simulation techniques (in Chapter 3);
- To simulate large ensembles of device layout motifs including a major variability source, line edge roughness, in deep sub-micron devices, and perform statistical analysis to find out the influence of intrinsic parameter fluctuations on motifs (in Chapter 3);
- To develop a novel tool based on bio-inspired approach to extract the atomistic device simulation results to generate corresponding compact model so as to bridge device simulation and circuit simulation (in Chapter 5);
- To develop a novel methodology, combining motifs and evolutionary optimisation algorithm, in order to achieve both circuit performance optimisation and variability-aware design (in Chapter 6).

1.3 Organisation of The Thesis

The outline of this thesis is organised as follows: Chapter 2 reviews the background associated with device scaling and corresponding challenges that come from lithography limitation, physical effects, and transistors variability. The transistor variability classification, sources of variability, and the impact of variability on device/circuit performance are comprehensively described.

Chapter 3, presents the novel concept, '*motif*', and how to investigate its characteristics at the device level. Motif is a generic scheme based on fundamental reused geometrical forms that prove convenient for increasing design regularity and examining the device/circuit problem of atomistic variability. A statistical motifs simulation methodology is introduced to investigate motif characteristics and their relationship with statistical variability. Finally, statistical analysis is carried out to show the distribution of motif statistical parameters.

Chapter 4 gives an outline of the field of evolutionary algorithms, incorporating basic knowledge of evolutionary mechanisms and the main operations in evolutionary algorithms. Additionally, some of the issues related to the application of evolutionary algorithm on VLSI design are described.

Chapter 5 introduces statistical motif compact model generation processes. The extraction process is divided into two stages. The first stage is uniform model extraction that is performed by a novel bio-inspired tool, which uses an incremental evolutionary algorithm to extract model parameters from device simulation results. The second stage is statistical variability generation, where parameter variations following a specific distribution are injected into uniform models to achieve variability-enhanced motif compact model.

Chapter 6 proposes a methodology to combine evolutionary algorithm and motif to achieve circuit performance optimisation. A set of possible motif permutations is tested to build benchmark circuits to find the "appropriate" permutations to achieve the best circuit performance.

Finally, concluding remarks and observations are given in Chapter 7 in addition to suggestions for future work. In order to easily understand the thesis structure, an organisation diagram is drawn in Figure 1.2.

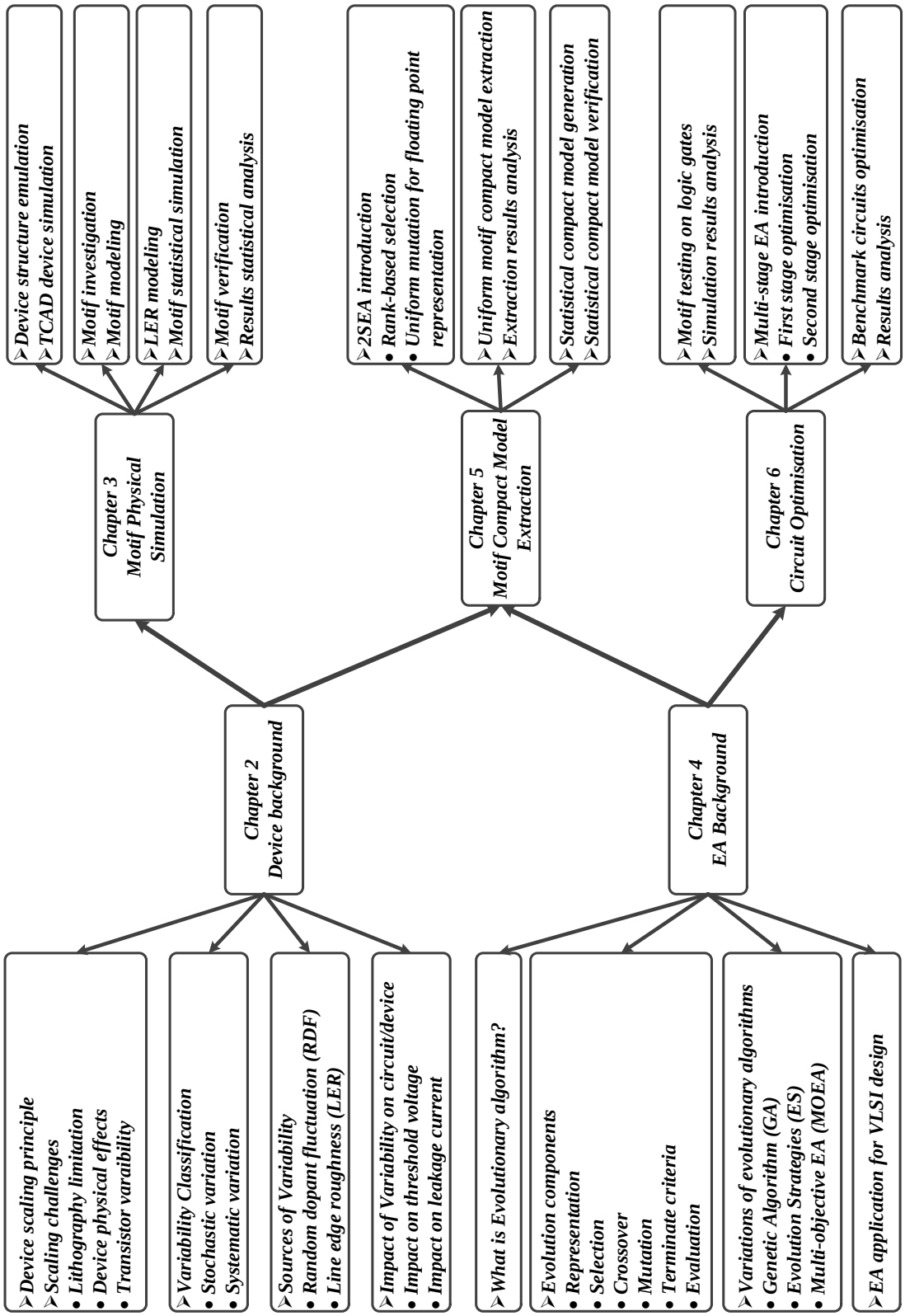


Figure 1.2: The thesis organisation diagram, which summaries main chapters' topics, key observations (represented by ▷) and points of interest (represented by •).

Chapter 2

Transistor Scaling and Variability

Contents

The semiconductor industry was born in 1960s. Pursuing circuit design cost efficiency and functional density benefits remain a key driver of the semiconductor industry development. New semiconductor technologies (often obtained by scaling) are introduced every 2-3 years that allowed to define "ICs", consisting of millions of transistors, to be produced cost effectively with product performance enhancements. The implementation of new technologies has successfully supported the pursuing year-by-year requirements of the semiconductor industry

driven somewhat by Moore's Law [5][10]. As the 2013 international technology roadmap of semiconductor (ITRS) predicts [6], device dimensional and functional scaling of CMOS will continue to strongly affect device cost and performance. Although scaling offers improvements in at least one feature of size, power, speed, or cost benefits, the uncertainty of the physical electrical properties of transistors caused by natural variation along with scaling to the atomic level have significantly affected IC design performance. Additionally, transistor fabrication limitation and imperfection due to processes variation at atomic scales has also further aggravated the variability of device performance. These uncertainty and imperfection factors double the risk of IC design [7].

This chapter reviews the basic concepts of semiconductor device physics and variability issues to prepare the necessary background on device scaling and its challenges regarding intrinsic parameter fluctuations due to scaling. Starting with the concept of device scaling in Section 2.1, the scaling principle directly supports device dimensions linearly scaling that contribute effectively to the performance and density improvements in VLSI evolution. Several challenges due to atomistic variations are addressed, including lithography due to manufacture limitation and short-channel effect (SCE) occurring as device features reach atomic levels. For example, the threshold voltage (V_{th}) is further decreased due to drain induced barrier lowering (DIBL) resulted in SCE, particularly at high drain voltage; and the sub-threshold leakage current is increased. On the other hand, the variation in channel length also further leads to the SCE increasing [8]. Sections 2.2- 2.4 focus on the basic knowledge of the characteristics of variability. These include variability classification (Section 2.2), the sources of variability (Section 2.3) and the impact of variation on device/circuit performance (Section 2.4). Generally, the variability is classified into systematic and intrinsic variation that are attributed to many reasons, including manufacture imperfection, the operating environment variation and intrinsic device parameter fluctuation. The work reported in this thesis only considers the variability caused by intrinsic fluctuation. Two primary variability sources (random dopant fluctuation and line edge roughness) are considered to address these intrinsic variations. Finally, Section 2.4 cover the impact of metal-oxide-semiconductor field effect transistor (MOSFET) variability on device/circuit delay and leakage current.

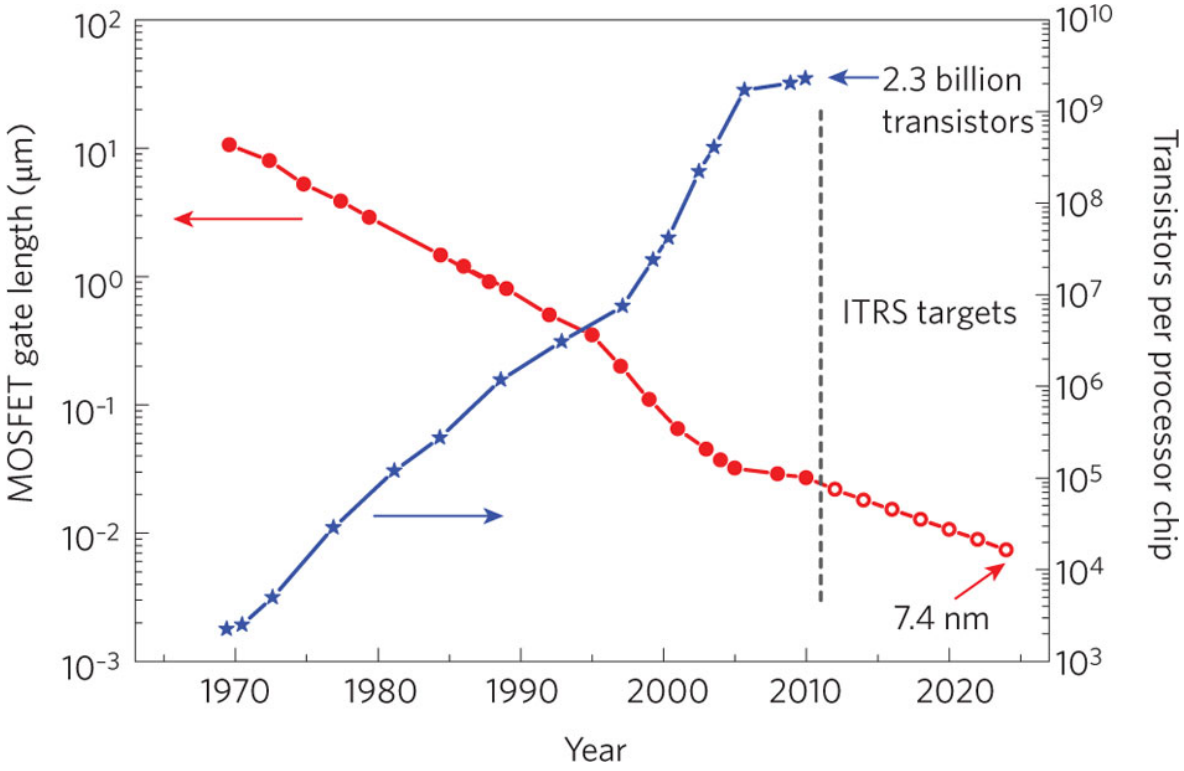


Figure 2.1: Trends in device electronics. The number of transistor per processor chip has increased (blue star line) as MOSFET device feature size scaling (red dot line). Evolution of MOSFET gate length is represented by filled red circles and ITRS predictions is shown by open red circles [9].

2.1 Scaling of Transistor Device

2.1.1 Scaling Principle

The aim of scaling to shrink MOSFET transistor dimensions is to obtain larger transistor densities to increase system functionality and enhance performance benefits. The essence of Moore's Law observed a trend in 1965 that the number of transistor density on a single chip will double every two years [5]. This trend has continued development never halt until today. The driving force behind this trend is the continuous miniaturization of the device and relative scaling theory [10]. The characteristics of MOSFET device preserved and maintained along with the critical parameters of device scaled is supported and governed by the era of classical geometrical-driven scaling principle (Dennard's scaling guidelines in 1975) [11]-[12]. An accurate reflection of transistor density growth on the single chip following Moore's prediction and evolution of MOSFET device technology is illustrated in Figure 2.1 [9]. These curves showing the expected doubling of transistor density through transistor dimension shrunk about 30% every two years result in a chip area reduction 50%. Transistor scaling not

only increases transistor density on the single chip but also enables transistor performance improvement (faster speed because of the scaling of the gate capacitance which reduces the delay and less energy due to power supply scaling). In general, every generation of device technology produces 50% more transistor on the chip and increases speed by about 40% though consuming nearly the same amount of energy as the previous generation technology [10].

To explain geometry driven scaling principles, an example of a scaled MOS device is demonstrated here. The cross-sectional structure of this scaled MOS device is depicted in Figure 2.2 [11]. The first characteristic influenced by scaling is transistor dimension and chip area. Both

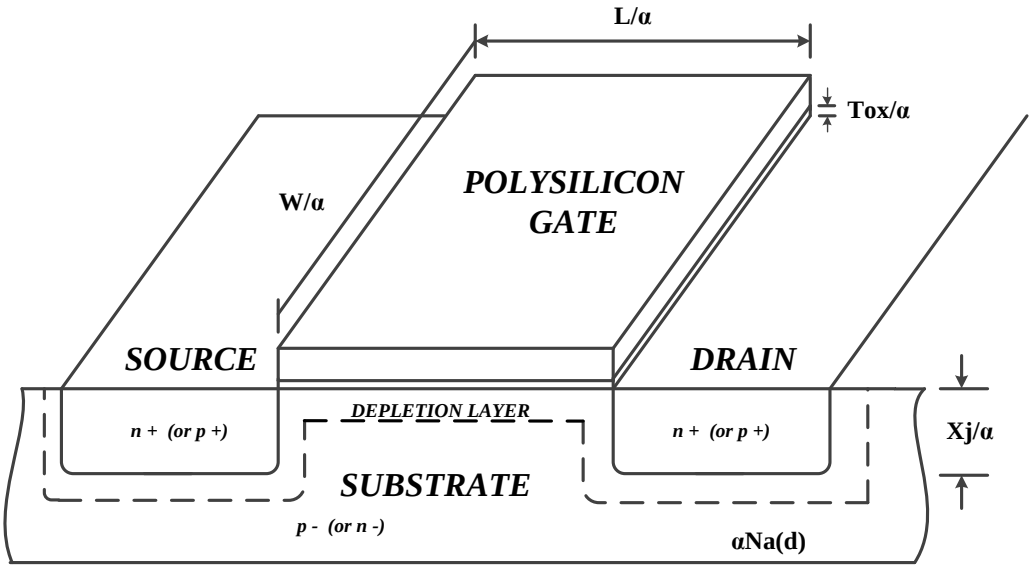


Figure 2.2: A cross-sectional structure of scaled MOS device [11].

lateral and vertical device dimensions have been affected as device scaling. The transistor density in the chip increases as the result of reduction of the lateral dimensions. For instance, the equivalent oxide thickness is reduced scaling factor ($\frac{1}{\alpha}$) that has reduced from 100 nm to around 1 nm in modern CMOS technology. The device channel size determines the depletion regions associated with the *pn* junction of source and drain. In order to control the conductance of the channel this requires the sum of the widths of the depletion layer to be smaller than the source-drain distance [11]. The reduction of the channel length relates

Table 2.1: Influence of scaling on MOS device characteristics

	Parameters	Scaling Factor
Device Parameters	Length (L)	$1/\alpha$
	Width (W)	$1/\alpha$
	Junction depth (X_j)	$1/\alpha$
	Substrate doping N_a	α
	Gate oxide thickness (t_{ox})	$1/\alpha$
	Depletion layer thickness (d)	$1/\alpha$
	Gate Delay (VC/I)	$1/\alpha$
Resultant Influence	Dynamic power dissipation (P_d)	$1/\alpha^2$
	DC power dissipation (P_s)	$1/\alpha^2$
	Gate area	$1/\alpha^2$
	Power density	1
	Current density	α

to reduction in the width of the depletion layers, which corresponds to the doping concentration of the substrate silicon increasing. For example, the substrate doping concentration is increased $N_{anew} = \alpha N_{aold}$, accomplished by the device width, the gate oxide thickness is linearly reduced by the same scaling factor. Thus, it brings the second characteristic that the corresponding electrical parameters are also scaled. For instance, the supply voltage is reduced by the scaling factor α. At the same time, the drain-to-source current (I_{ds}) is also decreased as the supply voltage is reduced. The other obvious parameter is power density. The power dissipation of the single device decreases by $\frac{1}{\alpha}$ as result of scaling. This results in the total power density of a chip remaining constant while the number of devices per unit area increasing by α^2. Finally, Table 2.1 lists the device scaling parameters and the corresponding resultant effect [11] so as to summary the influence of scaling on device characteristics.

2.1.2 Scaling Challenges

Lithography Challenges

From the viewpoint of market demand for faster, smaller, and cheaper electronic products, the possibility to pattern devices with ever smaller features on silicon has promoted the integration of millions or billions of transistors on a chip. The lithography processes is a key factor in the progress of device scaling. The development of optical lithography technologies

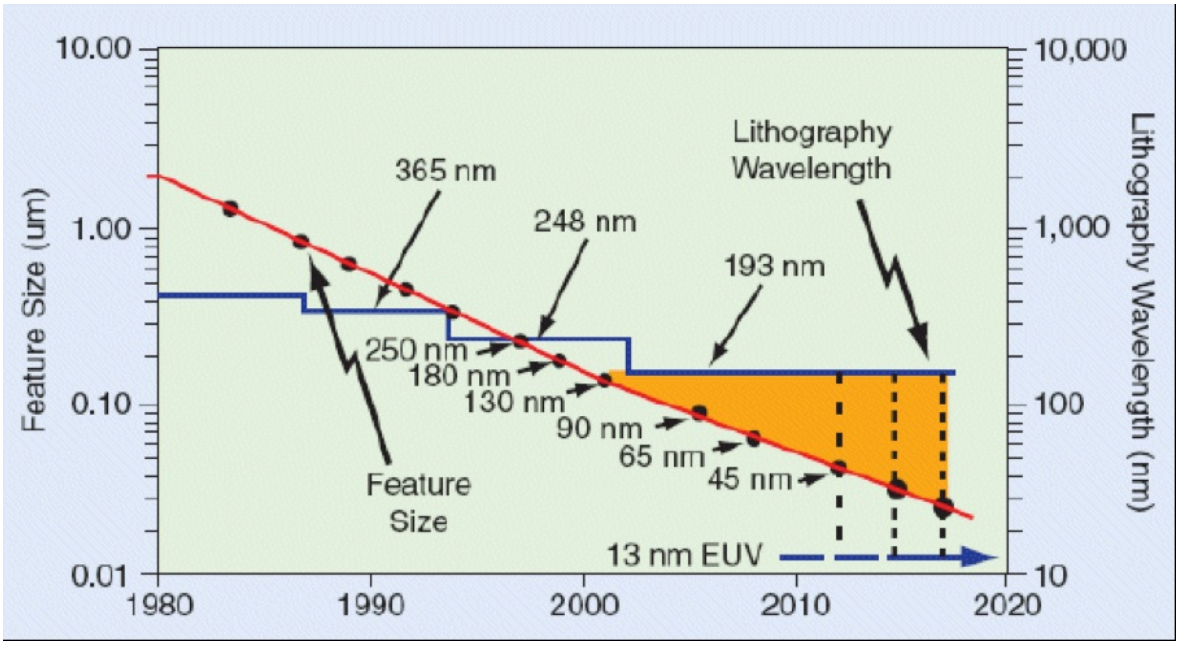


Figure 2.3: Evolution of lithography wavelength (blue line) as silicon feature size (red dot line) scaling. As devices scale to nanometer regions, manufacturability challenges "subwavelength gap" continue to grow. The illumination wavelength (193 nm) is much greater than the feature size (*45 nm* and below) [14].

provides the necessary manufacture support to keep pace with device technology improvement in the semiconductor industry [13]. The trend of optical lithography has continued from 435 nm (g-line of mercury lamp) wavelength in the 1980s to 193 nm (ArF laser) wavelength in 2000. Figure 2.3 shows the evolution of wavelength as silicon feature size scaling [14]. However, this evolving pace is broken while device scaled into nanometer regime. The pattern of device at *130nm* CMOS technology is created from a 248 nm wavelength the krypton fluoride (KrF) laser. The wavelength of light is larger than the minimum device feature size to be printed. This meant that the physical geometry patterns cannot be reliably printed for a given lithography step [15]. The patterns suffer higher distortions and no longer "what you see is what you get". For example, an example of lithography simulation of flip-flop is shown in Figure 2.4 [16]. This lithography simulation shows that two points in the layout are shorted due to the pattern distortion.

In order to keep pace of technology node development and print reliable devices with minimum feature size, designers have introduced and explored various techniques to reducing failures that occur because of lithography such as chemically mechanically polished (CMP), etch-loading, and other complex mechanical interactions. One of most cost-efficient techniques and powerful patterning tool is resolution enhancement techniques (RETs), including optical proximity correction (OPC), phase shift mask (PSM) and off axis illumination (OAI) to

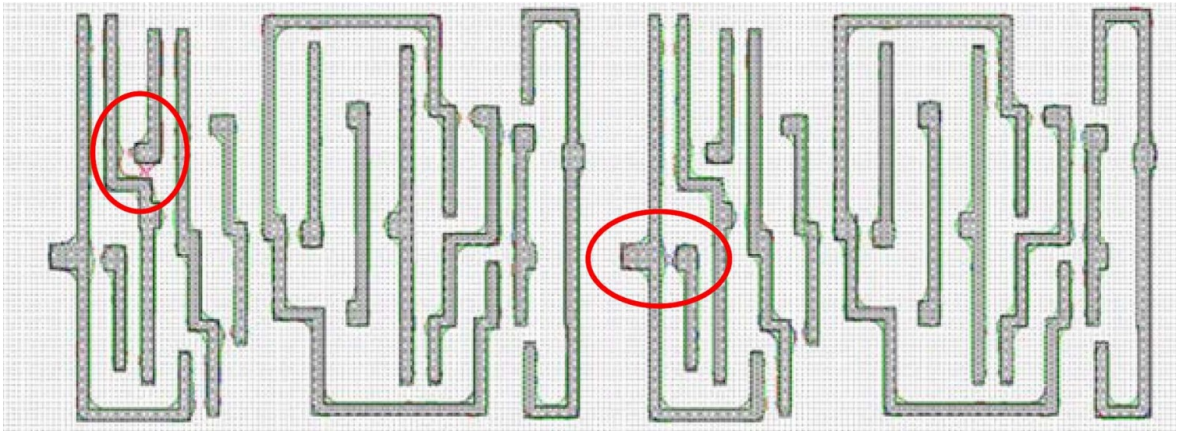


Figure 2.4: An example of lithography simulation of flip-flop. Because of pattern distortion, some of points in the layout are shorted and highlighted by red circle [16].

modify the chip mask database (GDSII) and achieve better printability, higher yield, and less variability [17]-[23]. A comparison of device layout pattern using OPC and no OPC is illustrated in Figure 2.5 [17].

Besides the challenges on lithography and manufacture process, the natural properties of devices deriving from the small-geometry effects also limit device scaling, including weakened gate controllability over the channel, threshold voltage variation, increased sub-threshold leakage current and S/D resistances.

Direct Tunneling Gate Leakage Current

In order to maintain the gate controllability over the channel while the feature size scales, gate oxide thickness needs to be reduced. However, the direct tunnelling gate leakage increases exponentially due to quantum mechanical tunnelling for gate oxide thickness less than 2 nm, a major contributor to static power dissipation [8][24]. Facing this serious problem, one of solution is to replace the SiO_2 material by alternative gate dielectric materials ($SiON$) [8] with higher dielectric permittivity (high-k), which allows the reduction of the equivalent oxide thickness (EOT) to 1 nm. Additionally, several other high-k dielectrics based on Hafnium (HfO2 with dielectric constants beyond 20) have also been introduced to improve device performance [15].

Threshold Voltage Variations

Although many design benefits, such as the decreasing threshold voltage and lower intrinsic capacitance are gained from decreasing channel length, the channel length cannot be arbi-

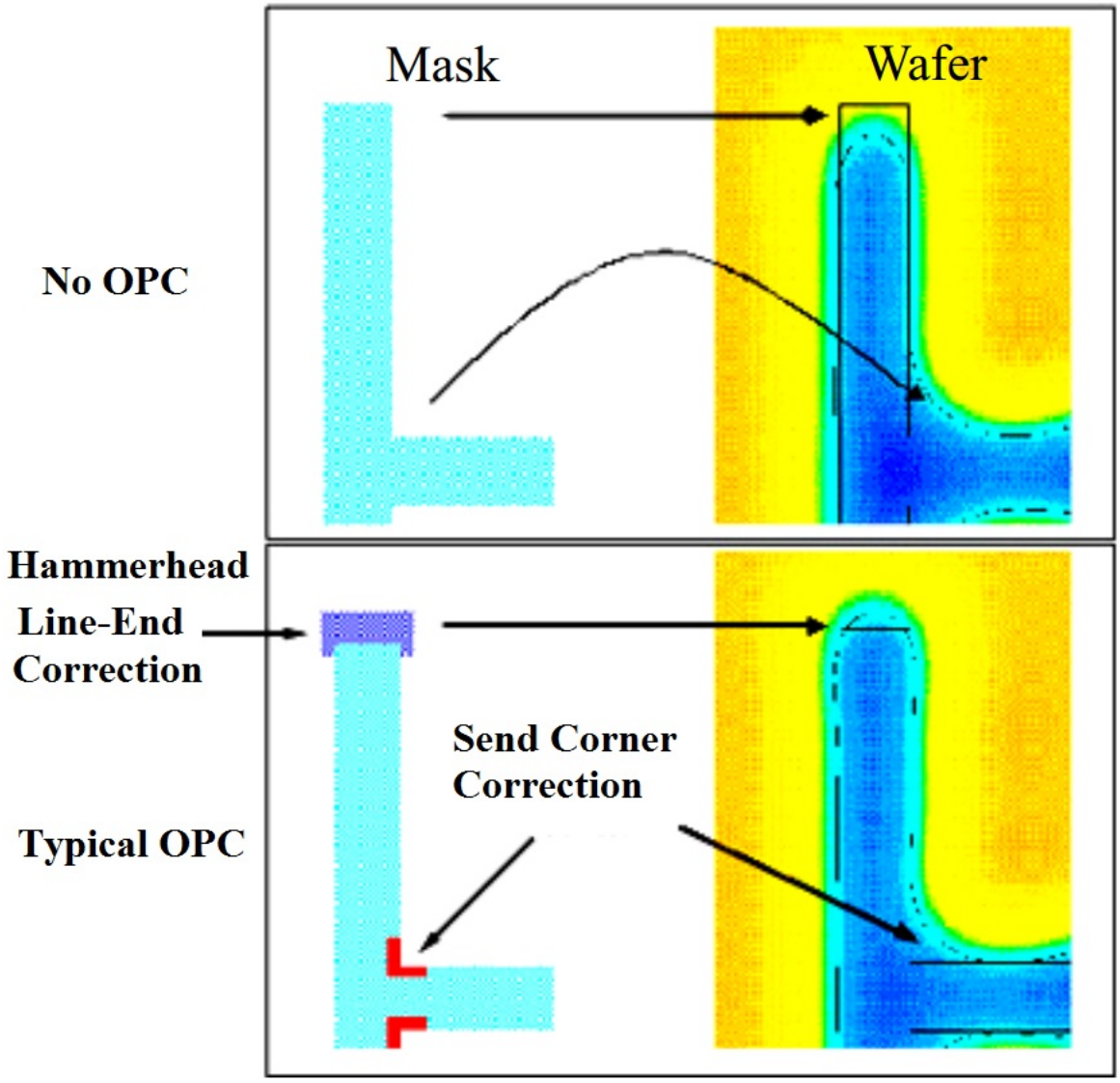


Figure 2.5: The comparison of device layout pattern using OPC or no OPC [17]. The case where OPC correction is used obviously has higher yield than no OPC design.

trarily reduced even if it is achievable through the lithography [8], due to the short channel effects (*SCE*). The obvious effect of the SCE on the device is the threshold voltage (V_{th}) *roll-off* as a result of the channel length decreasing and the drain induced barrier lowering (*DIBL*) leading to a further reduction of V_{th}. Figure 2.6 shows the DIBL and V_{th} *roll-off* [25]. Compared to long channel, the shorter channel length has the lower lateral potential barrier between the drain and source, as shown in Figure 2.7 [10]. This means that devices require less gate voltage to deplete the substrate beneath the gate dielectric.

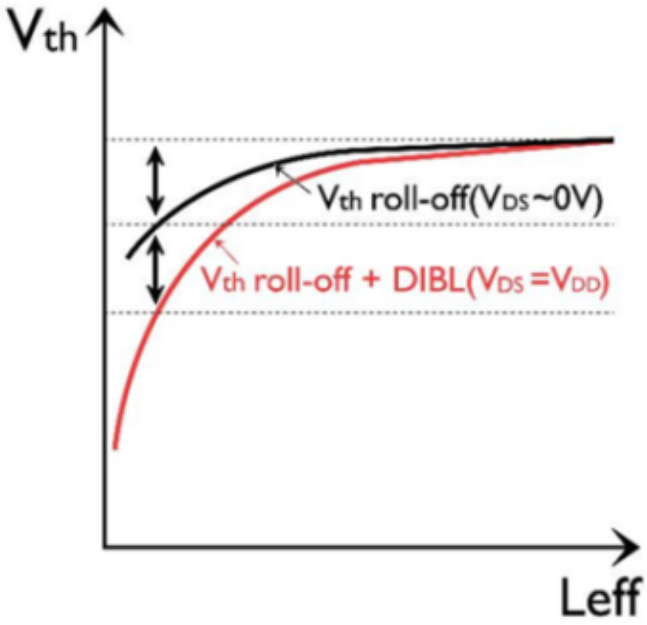


Figure 2.6: Threshold voltage V_{th} roll-off and drain induced barrier lowering (DIBL). V_{th} decreases as the channel length reduction. In addition to this, V_{th} further reduced when drain voltage increases is caused by DIBL [25].

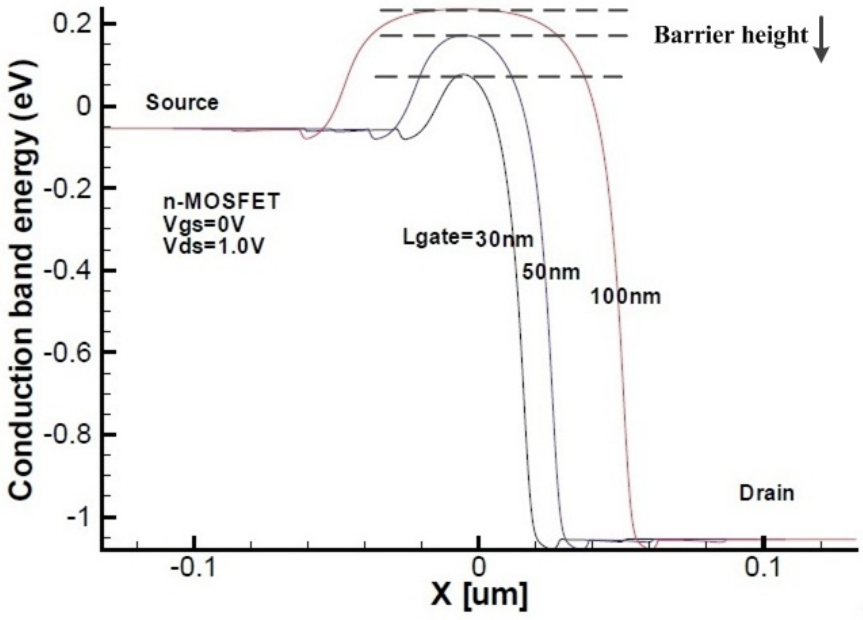


Figure 2.7: Barrier height lowering due to channel length reduction [8].

2.2 Transistor Variability Classification

Variation is the deviation of realized values from its original design intent. As the semiconductor device continued scaling to the nanometer scale, undesired and uncertainty variation result in the performance of devices and circuits degrading, even yield loss, caused by the limitation of the modern manufacture processing and device intrinsic parameter variations at atomic levels. Variability problem becomes a critical challenge meaning that designers had to face and attempt to design different methodologies to ensure robust device and circuit performance. The variation can be characterized in many different ways based on the device characteristics. Generally, variations fall into two categories: random/stochastic and systematic variations [26]-[28].

2.2.1 Random/Stochastic Variation

Random/stochastic variation is an uncertain or undesired component caused by fundamental phenomena associated with modern device structures and manufacturing technologies [27]-[28]. Examples of such variation incorporate random dopants fluctuation, line edge roughness, poly-silicon grain boundaries, surface roughness and gate oxide thickness induced variation. These variations are shown in Figures 2.8, 2.9 and 2.10. In general, random variability is an uncorrelated variation that depends on independent random variables, where the information about characteristics of individual device cannot provide any additional information to predict the characteristics of other devices [29]. However, random variations can be analyzed and predicted by a set of statistical variability results that has been addressed by the literature [30]-[32][35]-[36]. An example of investigation the impact of statistical process variations on a NMOS device using *35 nm* technology is shown in Figure 2.11 [8].

2.2.2 Systematic Variation

Systematic variation usually occurs at die, lot and wafer level resulting from imperfections caused within the lack of manufacturing control or the lack of fidelity in reproducing mask patterns onto the wafer [27][33]-[34]. Systematic variation is correlation variation that can be predicted. Common examples including transistor dimension and gate oxide thickness variation across chip, from die to die and from wafer to wafer and can be evaluated through calculating the impact of layout effects and corresponding spatial parameters [33]. An example of investigating effect of intra-die correlation on statistical performance analysis of 1-bit

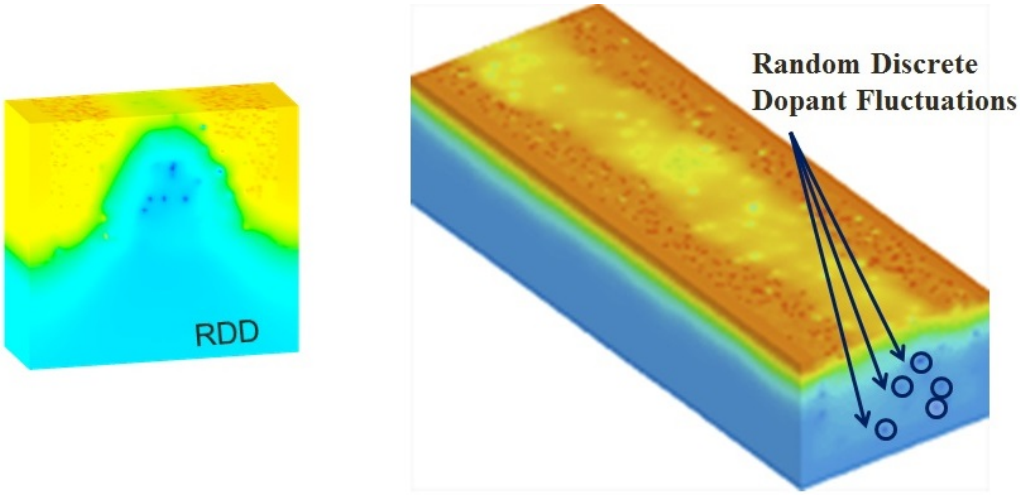


Figure 2.8: Random discrete dopant fluctuations illustrated in 3D device structure [22]. It is impossible to perfectly control the exact quantity and position of dopant atom while dopant atoms are being implanted into the silicon. These results in exact differences in the electronic characteristics of two transistors, even both are fabricated under the same conditions.

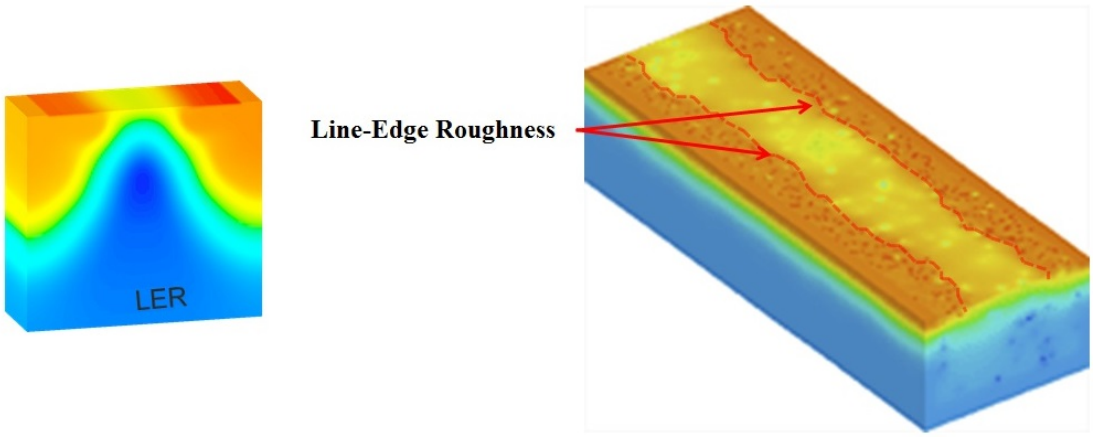


Figure 2.9: Line edge roughness (LER) in 3D device structure [22]. The random deviation between actual gate edges and ideal definition of gate edges is not avoided due to the limitation of resolution in optical lithography. LER results in enhanced lateral diffusion and effective channel length reduction.

full adder is addressed to illustrate how to analyse and predict the systematic variation [34]. 1-bit full adder circuit is laid on various spatial positions. These layouts are assumed to have different intra-die correlation due to systematic spatial variation. The histogram of full adder delay measured from 500 samples with different correlation coefficient is shown in Figure 2.12 [34]. Based on simulation results analysis, the delays standard deviation on the layout design with low correlation has decreased about 7.8% comparing to that with high correlation.

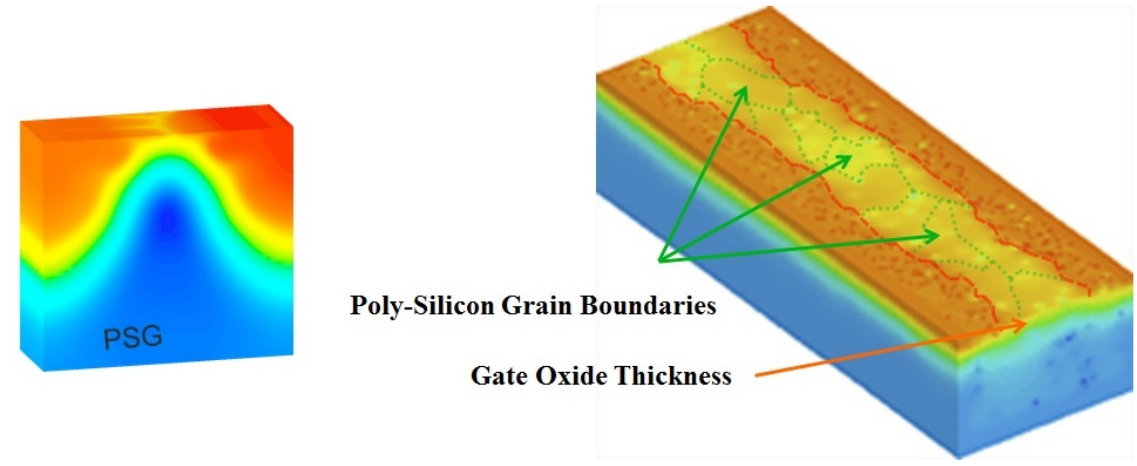


Figure 2.10: Poly-silicon grain boundaries (PSG) and gate oxide roughness (TOX) in 3D device structure [22]. PSG is another variability sources to affect on the channel current due to the random arrangement of grains within the gate material. Additionally, with the gate oxide thickness scaled, TOX induced variation is not ignored, especially TOX variation induced threshold voltage variation as much as RDF below *30 nm* technology [36].

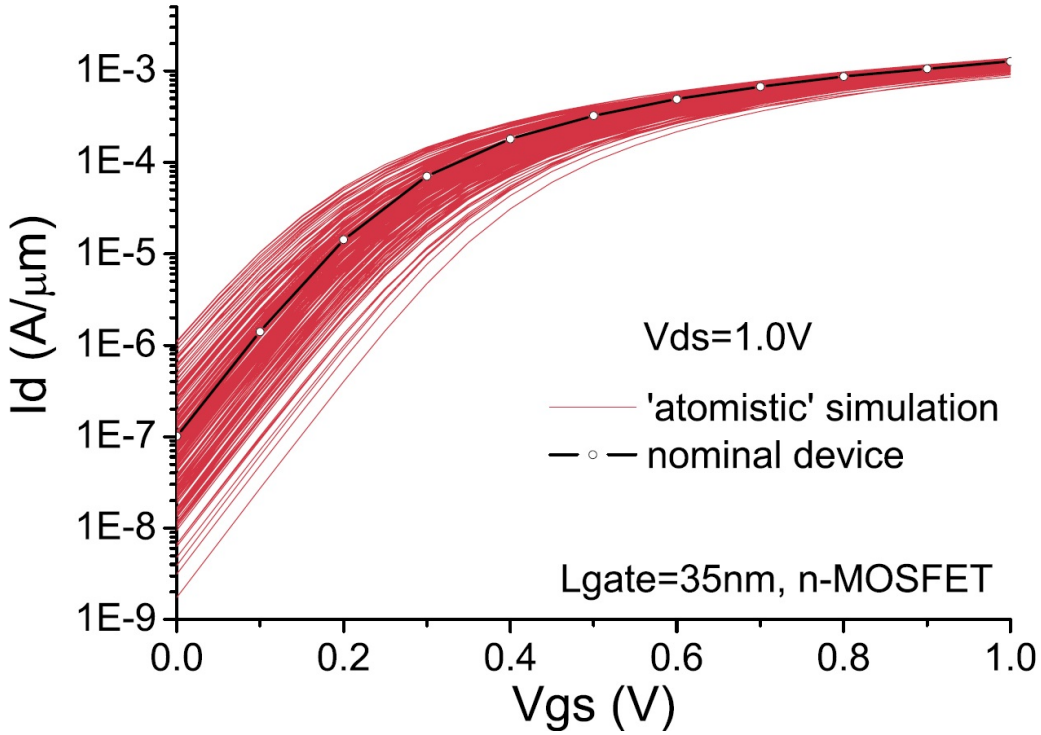
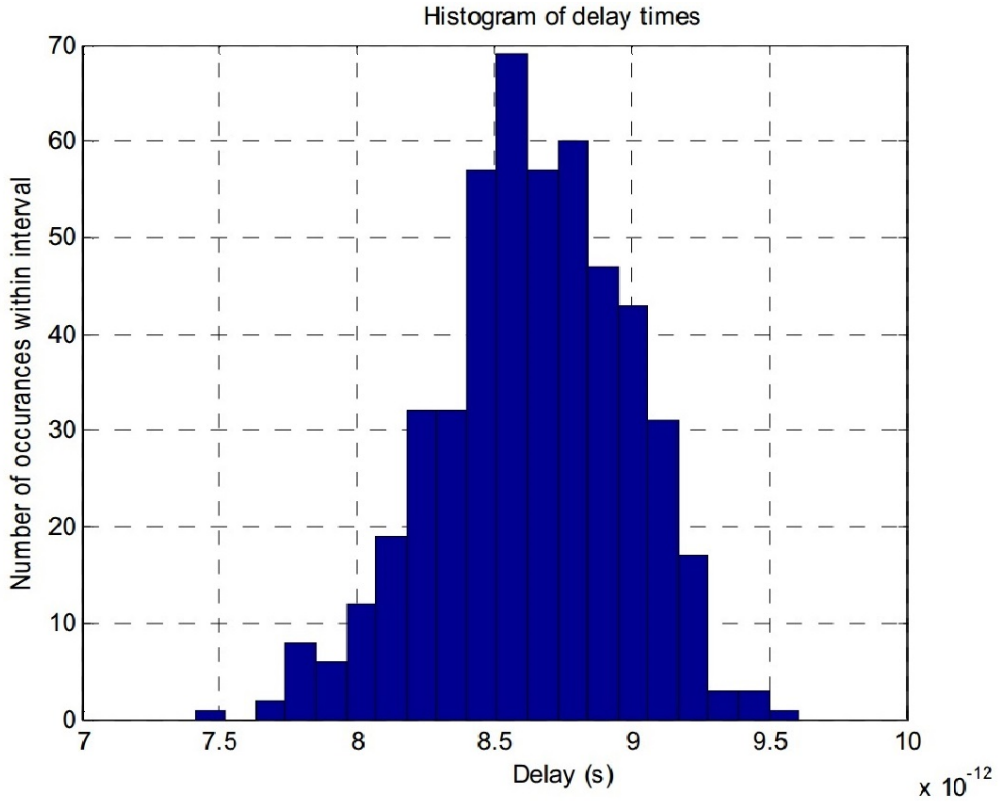


Figure 2.11: I_d-V_g characteristics for *35 nm* gate length NMOS subject to RDD, LER and PSG induced statistical variability[8]

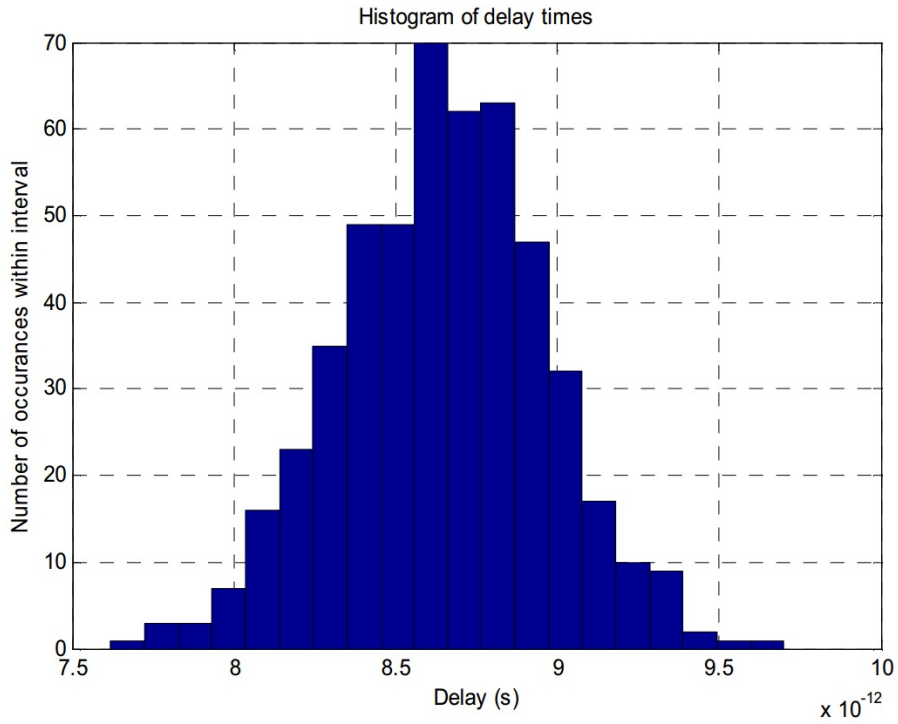
2.3 Sources of Stochastic Variability

2.3.1 Random Dopant Fluctuation

Random dopant fluctuation (RDF), which is the dominant source of stochastic variability, is an unavoidable form of process variation resulting from fluctuations of the implanted im-



(a)



(b)

Figure 2.12: The histogram of layout with different correlation delay measured from 500 randomised instances [34]. Sub-figure (a) shows (the mean is 8.64ps and standard deviation is 0.345ps) results from layout with higher correlation ($\lambda = 0.07$); and Sub-figure (b) design (the mean is 8.647ps and standard deviation is 0.318ps) with low correlation ($\lambda = 10$). Although mean of both cases remain similar, the standard deviation obviously reduces by 7.8% [34].

purity (dopants) concentration. The basic structure of a device with discrete impurities is shown in Figure 2.8. The dopants are implanted into the silicon with very high energies; the subsequent thermal annealing process results in the silicon atoms within the crystal lattice being substituted by the implanted atoms. Because of the spread of atoms position it is difficult to accurately control the quantity and positions of the individual dopant atoms; this means that every device has its own unique distribution of dopants [23]. As devices scaled into nanometer region, the device electrical characteristics and behaviour are further affected and degraded by the sporadic implanted impurity concentration and stochastic placement of impurity. Traditionally, based on assumptions of statistical averaging of the dopant concentrations, transistors should have smooth carrier concentration and potential profiles. Practically, the implanted impurity concentration within active region of a device is relatively low while scaling to nanometer, which depends on the actual number and specific placement of dopant atoms [23]-[24].

The significant effect of RDF is a random shift in the threshold voltage (V_{th}) of a device [25]. Based on a set of devices simulations and the corresponding literature published by Asenov *et al*, the number of dopant atoms present within the active region of a minimum-sized device is typically around one hundred that of the number of dopant atoms within the channel region followed a Poisson distribution around a mean value, $N_d = 130$ [25]-[26]. The threshold voltage is significantly affected by the influence of dopant atoms on the potential. Besides threshold voltage variations, a number of other major device characteristics are also affected, such as sub-threshold swing, sub-threshold leakage current and drain current [23]. Although new device structures such as FinFETs [67] and ultra-body-thin silicon-on-insulator (UBT SOI) device [15], reduce the variability problem, this problem is not entirely solved [27]-[29].

2.3.2 Line Edge Roughness

Line edge roughness (LER) is the second dominant source of intrinsic variability, derived from the discrete molecular structure of the photo-resist used in the lithography process [30]-[32]. Slower dissolution rate of large polymer aggregates in the transistor formation process is the primary cause of the LER [11]. Figure 2.13 depicts a SEM image of line edge roughness patterns.

Although RDF is primary source of intrinsic variations in bulk CMOS device, the effect of LER on device performance is much stronger even overtake RDF below *30 nm* technology node and become the dominant variability source, as shown in Figure 2.14 [74]. Therefore, it

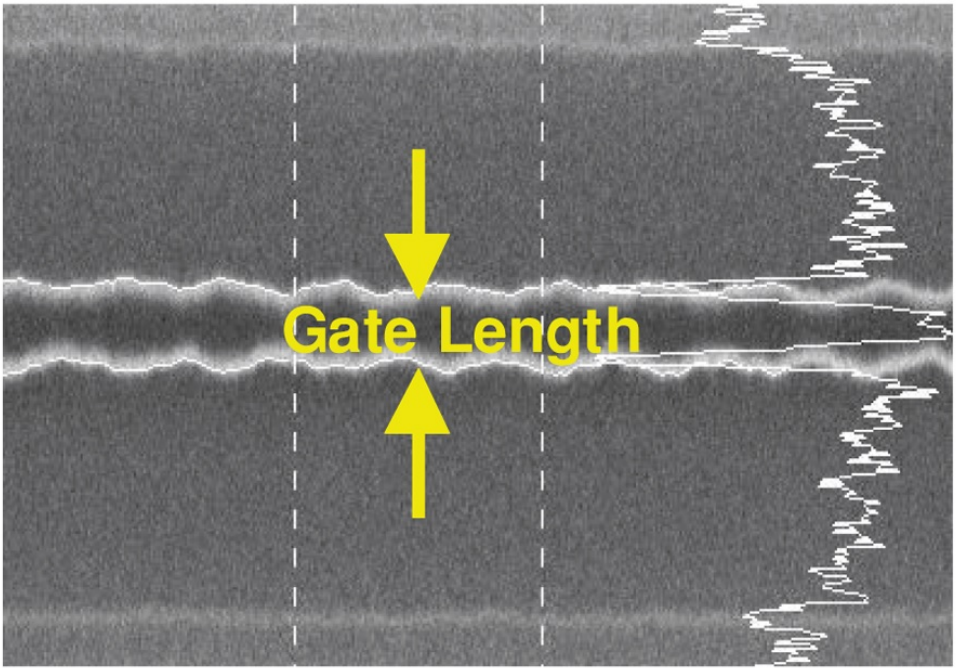


Figure 2.13: SEM image of line edge roughness pattern [33]

is necessary to understand the effect of LER on device performance. Many modelling methods such as 2D gate slicing method [71] and 3D LER modelling using power spectrum method [74], have been proposed to model and characterise LER. The details of these methods will be addressed in Chapter 3.

2.3.3 Interface Roughness

An interface (dielectric layer) separates the substrate and the transistor gate. SiO_2 is usually used to form this interface due to its excellent interface properties [29]. This thinner layer with higher dielectric constant gives better transistor performance. As the device feature size scaled into atomistic levels, gate oxide thickness becomes equivalent to several atomic layers. The conventional way of assuming smooth boundaries and interfaces is no longer valid [38]. A SEM image of interface roughness is shown in Figure 2.15 [40]. When the oxide thickness is only a few atomic layers the interface roughness steps will result in significant oxide thickness variations (T_{ox}) [38]. The variation in gate oxide thickness makes each transistor microscopically different, which unique oxide roughness will affect the device characteristics such as mobility, gate tunnelling current and threshold voltage; and further having an impact on the device performance, yield, and reliability.

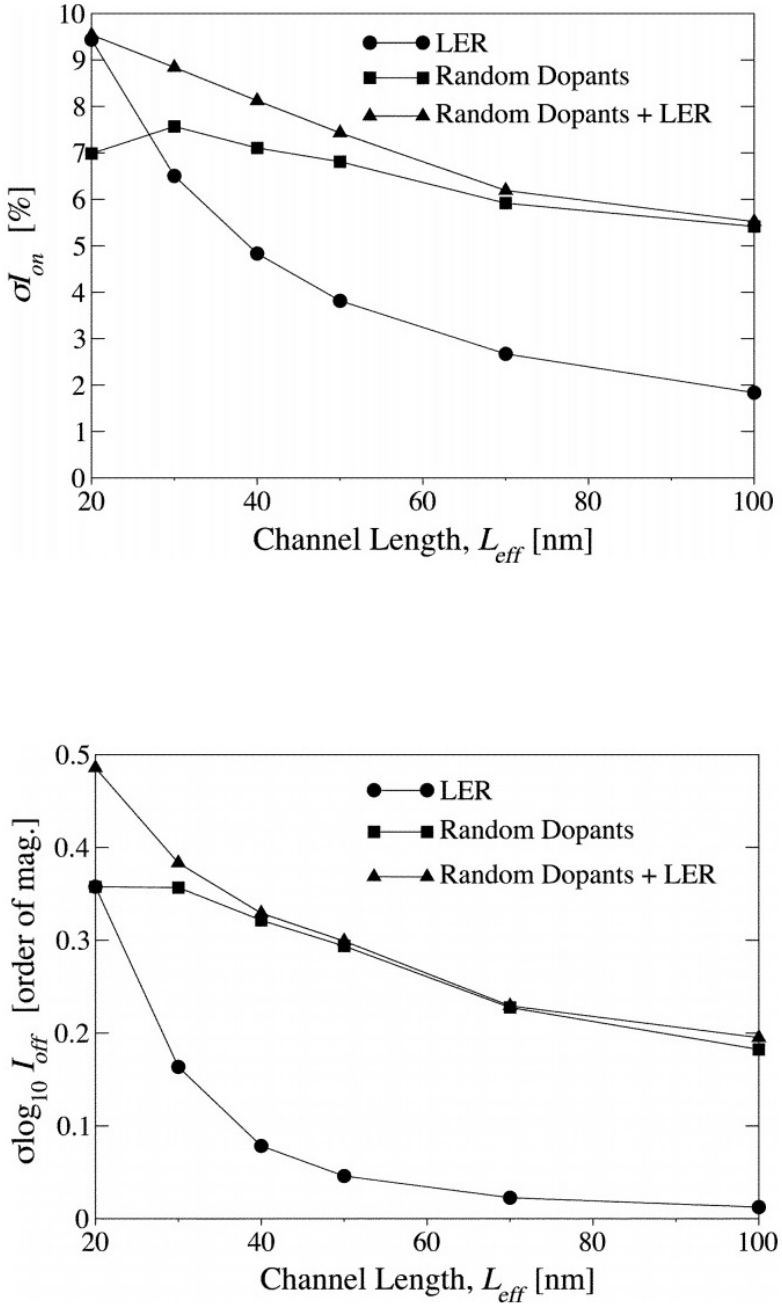


Figure 2.14: Dependence of σI_{on} (represented by upper sub-figure) and $\sigma log(I_{off})$ (represented by bottom sub-figure) on the channel length for a set of device with channel width 50 nm. As channel length reduction, the effect of LER on device performance is much stronger [74].

2.3.4 Polysilicon Grain Boundaries

The polycrystalline grain structure of the polysilicon gate is another source of intrinsic param-

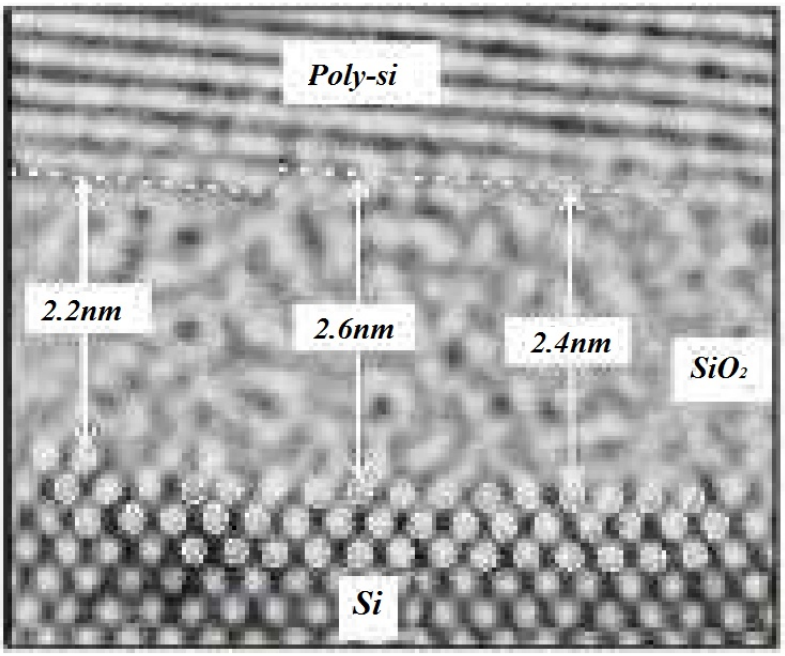


Figure 2.15: SEM image of interface roughness [40]

eter fluctuations, which polysilicon grain boundaries (PSG) variation (shown in Figure 2.16 [40]) occurs due to the random arrangement of grains with gate material due to this polycrystalline structure [15]. Polycrystalline silicon is rich in defects at the grain-boundary regions which can act as charge-traps [15]. Enhanced diffusion along the grain boundaries result in nonuniformity in the doping within the polysilicon gate. The dopants penetrate through the polysilicon and insulator into the channel region that results in localised random variations on the lower surface of the polysilicon[15].

2.4 Impact of Transistor Variability on Device/Circuit Performance and Monte Carlo Simulation

According to literatures [28][35][37] introduced, intrinsic variability has significant negative impact on the performance of device and circuit such as causing timing uncertainty logic circuits and exacerbating the power dissipation problems. Figure 2.17 illustrates the impact of intrinsic variability on the performance of XOR and XNOR gate design using 35 nm devices [15]. The delay (timing) and power has been significantly influenced by device variability, especially variations in delay (approximately 12%). In order to demonstrate how intrinsic variability affects the performance of a device/circuit, two important performance features

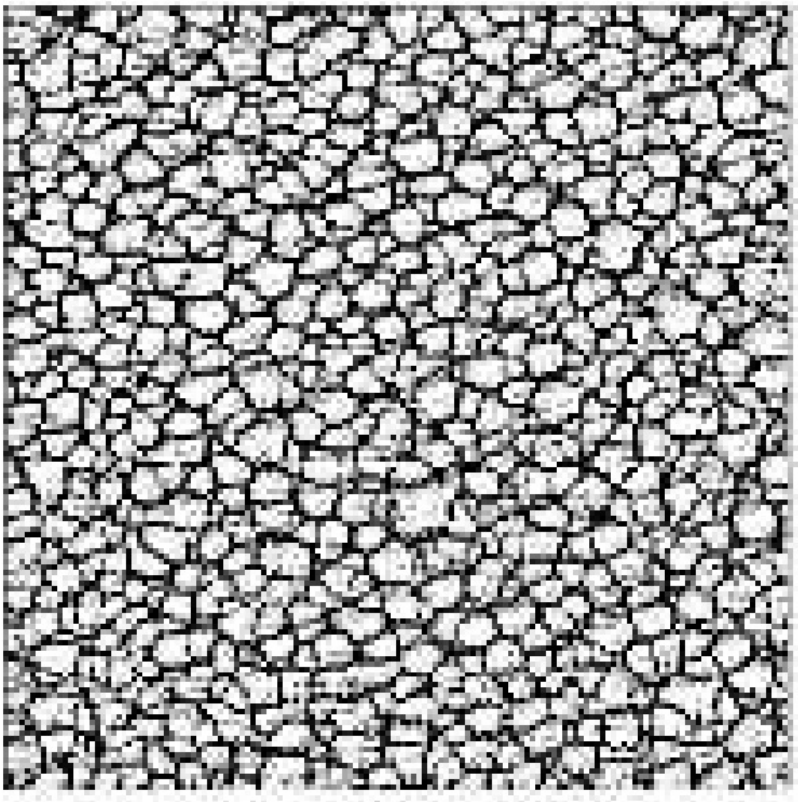


Figure 2.16: SEM micrograph of typical PSG from bottom [40]

delay and power consumption are used. In addition, Monte-Carlo simulation method is introduced that is commonly used to measure the impact of transistor-level statistical variations on circuit performance.

2.4.1 Impact on Delay Time

The delay variation is closely related to threshold voltage variability. The delay of logic gate depends on the geometrical size of transistors in the gate and the capacitance of load that must be driven [11]. According to literature [15], higher drain current results in faster capacitor charging time and Equation 2.1 illustrates the delay time for a switching MOSFET.

$$T_{Delay} = \frac{C|V_{DD}|}{I_{DD}} = \frac{2LC}{W\mu_{eff}C_{ox}(V_{DD} - V_{th})^2} \tag{2.1}$$

Where C is device capacitance, V_{DD} is the supply voltage and I_D is the drain current. C_{ox} is the gate oxide capacitance per unit area, μ_{eff} is the mobility of charge carries. W and L are device's width and length. Based on Equation 2.1 delay variation is decided by the threshold voltage (V_{th}) variation while keeping other parameters constant. This means that

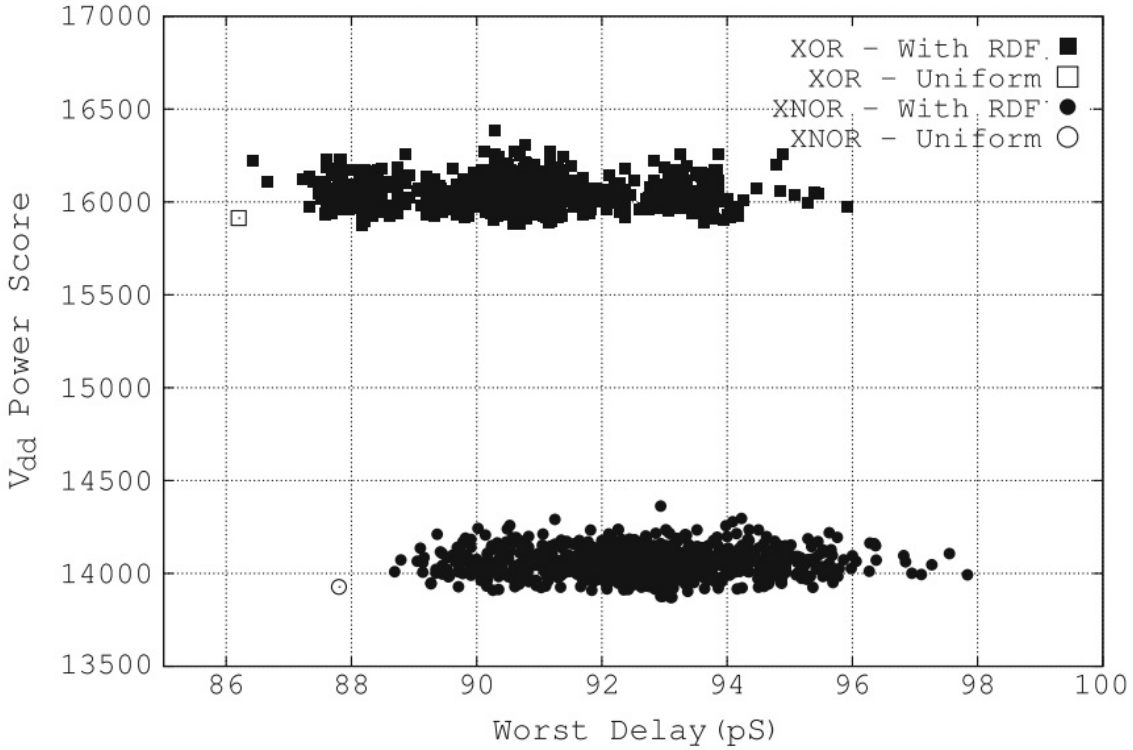


Figure 2.17: Effect of intrinsic variability within 35 nm XOR and XNOR logic designs [93].

larger distribution of threshold voltage leads to a wider distribution of delay time and vice versa.

2.4.2 Impact on Leakage Current

From *65 nm* and below, both average and standard deviation of leakage power consumption increase with device scaling [37]. Figure 2.18 compares the device power consumption *vs.* various technologies [39]. Obviously, the Leakage power consumption takes account for 40%∼50% in *65 nm* technology that has become a dominant contributor to the total power consumption [37]. Thus, here only discusses the impact of intrinsic variability on leakage power consumption. In order to investigate the effect of intrinsic variation on device leakage power consumption, sub-threshold leakage current is analysed here. The sub-threshold leakage occurs when gate-source voltage (V_{gs}) is below transistor V_{th} and has exceeded the weak inversion point [11]. The leakage current defined by Equation 2.2.

$$I_{Sub-leakage} = KV_T^2(W/L)\mathrm{e}^{\frac{V_{gs}-Vth}{nV_T}}(1-\mathrm{e}^{-V_{ds}/V_T}) \tag{2.2}$$

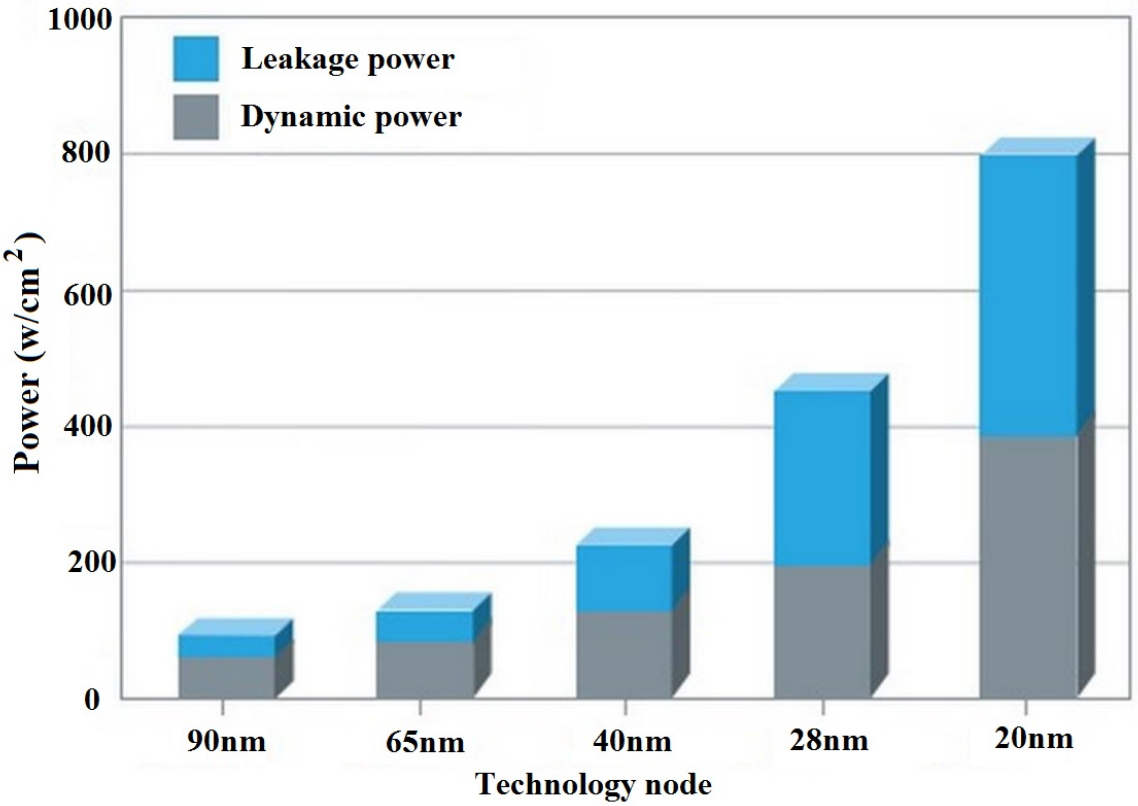


Figure 2.18: Total power ratio evolution *vs.* technology node [39].

Where K and n are technology parameters, W and L are device's width and length. V_T and V_{ds} are the thermal voltage and the drain-source voltage. From Equation 2.2, the leakage current increases exponentially as V_T increasing or V_{th} decreasing. Similarly, the leakage current variation also determined by V_T and V_{th} variations while other parameters keep as the constant.

2.4.3 Monte Carlo Simulation (Statistical Simulation)

Monte Carlo (MC) simulation is an important way to capture statistical information of a design by allowing designers to run a simulation several times, each times using different randomly generated values for a set of process parameters. As the literature [40] introduced, MC simulation is used to investigate and study statistical variability in transistor technology at the levels of physical transistor simulation, compact model and circuit simulation. The use of MC method in physical device simulation to obtain the variability information is discussed within [29]. In order to assist variability-aware circuit design, the obtained fluctuation information is transferred into statistical compact models to build circuit, and perform circuit level simulation. Similarly, the MC simulation method also provides one of the easiest ways

Table 2.2: Influence of random/systematic variation on MOS device parameter

Parameters	Random	Systematic
Gate Length (L)	Line edge roughness (LER)	Lithography and etching: proximity effects, orientation
Gate Oxide Thickness (T_{ox})	Si/SiO_2, $SiO_2/Poly\text{-}Si$ Interface roughness	Non uniformity in the process of oxide growth
Channel doping concentration	Random dopant fluctuation (RDF)	
Threshold Voltage	RDF, LER, Interface roughness, PSG, etc.	Non-uniform annealing temperature [28]
Leakage current	RDF, LER, Interface roughness, PSG, etc.	poly corner rounding

to predict circuit performance to assist circuit design and performance improvement.

2.5 Summary

In this chapter, an overview of some fundamental principles of transistor scaling and the key concepts of stochastic variability issues were introduced. The scaling principle relies on device dimensions linearly scaling so that IC design achieves the performance and density improvements in VLSI evolution. However, some challenges associated with devices scaled to the atomic scale become significant, particularly the impact of intrinsic variation on device/circuit performance. Several major intrinsic variability sources, random dopant fluctuation (RDF) and line edge roughness (LER), interface roughness and poly-silicon grain boundaries were explained in detail. Table 2.2 summaries the influence of random and systematic variation on MOS device parameters. The delay and leakage current are more sensitive to these variability sources and would require more care during the design stage so that some of the possible pitfalls can be avoid such as failure of timing analysis due to variation. Meanwhile, the understanding of variability aids proposed methodologies to mitigate and make device/circuit performance insensitive to variation. This chapter provides that the necessary background knowledge for an understanding of device concepts and the characteristic of statistical variation of fundamental physical parameters discussed in Chapter 3. Chapter 3 will discuss proposed motif concept and corresponding motif simulation in order to investigate the impact of LER-induced variability on various motifs.

Chapter 3

Motifs Physical Simulation

Contents

Two extreme approaches are usually used to study the design of integrated circuits. First a bottom-up approach begins with understanding semiconductor device physics and device modelling, and finally the circuit designs [41]. The other approach is top-down which treats semiconductor device as a black box and only uses its terminal electrical behaviour to design circuits with little attention to the device internal characteristics [41]. Here, we adapt the

first design methodology to investigate the impact of motifs on circuit performance variation. A solid understanding of transistor layout at device level is essential because many of layout features directly impact the device and circuit performance and these impacts are not easily explained from a circuit level viewpoint.

In this chapter, a brief review of the CMOS device fabrication flow and the relevance of device simulation methodology is given in Section 3.1. Following a general description of layout consideration in Section 3.2, '*motifs*' notion is introduced an intuitive technique that proves useful in understanding the link between transistor layout and its effect on device variability. In order to investigate this link a statistical physical device simulation methodology has been developed and is described in the Section 3.3, where a large sample of 3D motif structures with microscopically different LER pattern (different variability information) is emulated and simulated using the Synopsys technology computer-aided design (TCAD) tool. Finally, detailed analysis of the motif simulation results provide in depth understanding of transistor layout affected by LER statistical variability and is reported in Section 3.4.

3.1 Device Simulation and Methodology

3.1.1 CMOS Device Fabrication Flow

In order to understand how to use TCAD tool to predict device's electrical, thermal and other properties under different operating conditions without actual fabrication, it is necessary to have a detailed knowledge of the device fabrication process. In reality, the need for a series of steps, including oxidation, etching, polysilicon deposition and ion implantation, to achieve the circuit layout geometries are transferred onto silicon [41]. For the sake of conciseness and clarity, a bulk CMOS device fabrication process description is introduced by a set of illustrations and the following section.

Well preparation

The process begins with a p-type silicon wafer roughly 0.72 mm to 0.77 mm thick and with a diameter of either 200 or 300 mm (typically for *180 nm* technology) that carries an epitaxial layer of very moderate p-type doping. Figure 3.1 show cross sections of the wafer processed following steps to form the n-well. The first step in processing is oxidized the wafer in a high-temperature (typically $900°C - 1200°C$) furnace that forms an oxide layer (SiO_2) through the

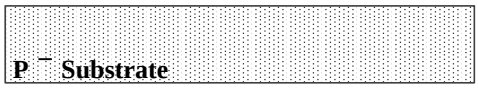
reaction of Si and O_2 becoming SiO_2 on the wafer surface, which is shown in Figure 3.1(b) [42]. Subsequent to these step, an organic photoresist is deposited on the top of the SiO_2 layer surface (Figure 3.1(c)), and then the n-well mask is aligned above the photoresist material layer so as to soften the uncovered photoresist through exposing under UV light. Then the n-well mask and the softened photoresist are removed for further processing (Figure 3.1(d)). The exposed oxide is removed through etching with hydrofluoric acid (HF) (Figure 3.1(e)) the rest of the photoresist material is stripped away using a mixture of acids (piranha etch [42]), shown in Figure 3.1(f). Next, the n-well is formed from where is the exposed wafer. In general, two ways are used, diffusion or ion implantation, to form the n-well. In the diffusion process, the gas containing the dopants is heated so that the dopant atoms diffuse into the exposed wafer. Because of lateral diffusion the well region is larger than the exposed region (Figure 3.1(g)). With ion implantation, n-type impurities are accelerated by an electric field and blasted into the uncovered wafer to form n-well. The remaining wafer is protected by the covered SiO_2 layer that prevent dopant atoms to enter the substrate. Finally, in order to obtain the bare wafer with wells for further processing the remaining oxide layer is stripped with HF (Figure 3.1(h)).

Transistor formation

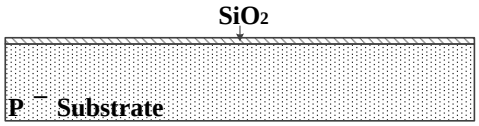
After the well is prepared, the transistor formation is introduced. Active areas are firstly defined so as to distinguish those areas that are to become body ties, diffusion areas and MOSFET region. In order to achieve isolation between p-type active area and n-type active area, the bulk material with the nitride layer protecting the silicon underneath from etching is isolated by shallow trench isolation (STI, approximately 300 to 400 nm deep). Then, this open trench is filled into the oxide material. In order to obtain flat surface, the wafer surface is polished via chemical mechanical polishing (CMP) since a perfectly flat surface is crucial for the subsequent steps (Figure 3.2(a)).

The next step is gate dielectric formation that grows a thin layer of oxide (3-5 nm typically for *180 nm* technology) on the top of the wafer (Figure 3.2(b)). This oxide thickness, designated t_{ox}, is an important process parameter that can be scaled as the technology scaling. Next, the transistor gate is grown followed by a polysilicon deposition step. The polysilicon film (roughly 200 nm) is deposited on the entire surface. The polysilicon is then patterned and etched to form the transistor gates and interconnect lines (Figure 3.2(b)).

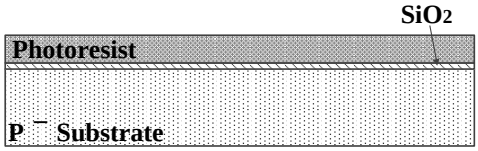
The wafer is now ready for the spreading of ion to form source/drain areas and body ties.



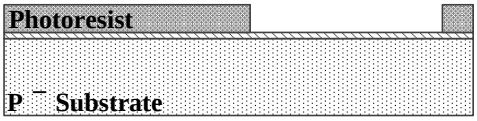
(a) Initial wafer. Processing is assumed to start from a p-type wafer.



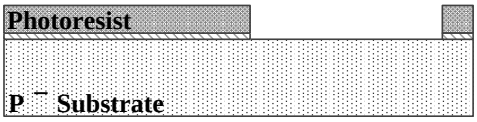
(b) Wafer oxidation that an oxide layer is deposited on the wafer surface.



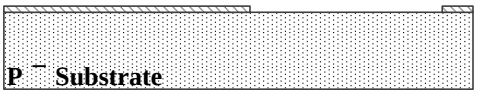
(c) Organic photoresist is deposited on the top of the oxide layer.



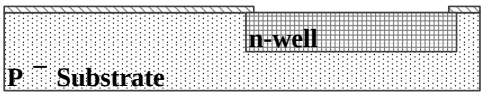
(d) Removing the softened photoresist.



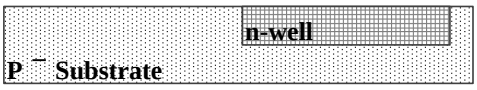
(e) Etching the exposed oxide so as to form n-well.



(f) Stripping away the rest of the photoresist on the oxide layer.

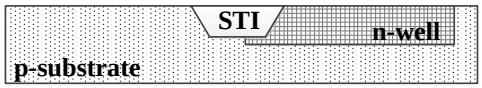


(g) N-well formation using diffusion or ion implantation.

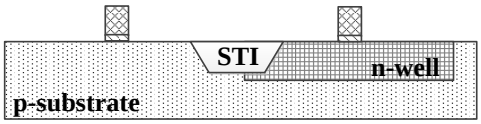


(h) Completing n-well formation and removing the rest of oxide layer.

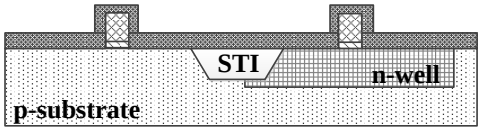
Figure 3.1: Cross section of the wafer involved in forming the n-well[42].



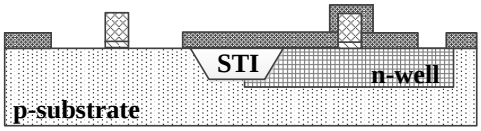
(a) Shallow trench isolation (STI) formation to achieve isolation between p-type and n-type active area.



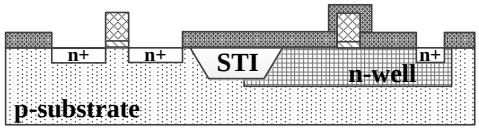
(b) Thin layers of oxide and polysilicon film are successively deposited to gate oxide layer and transistor gate.



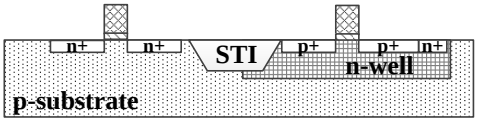
(c) A protective layer of photoresist is deposited and patterned with the n-diffusion mask.



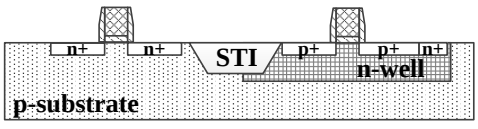
(d) The photoresist layer is patterned with the n-diffusion mask.



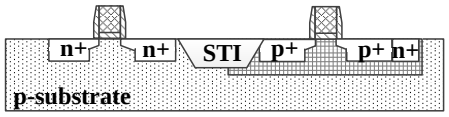
(e) N+ implantation to form n-channel device source/drain and p-channel device body tied.



(f) P+ implantation to form p-channel device source/drain.



(g) Sidewall or oxide spacer formation.



(h) A heavier and deeper implantation.

Figure 3.2: Cross section of the wafer while forming ploysilicon and diffusion region[42].

A protective layer of photoresist is formed and patterned with the n-diffusion mask (Figure 3.2(c)). The exposed regions are bombarded with donors. The polysilicon gate acting as a shield blocks the diffusion, called *self-aligned* process. Consequently, the edges of the source and drain areas are separated by a channel under the gate (Figure 3.2(d) and (e)). In this step, lightly doped drain/source diffusions (LDD) is employed to avoid excessive lateral fields and to better control SCE, specially occurring in submicron devices [42]. In order to build LDD structure, a spacer oxide or side-well is deposited through an anisotropic oxide etch leaving wells on the each side of the polysilicon gate, as shown in Figure 3.2(g). The process, only with opposite photoresist and opposite doping, is repeated to produce the p-channel source/drain extension, resulting in the cross section illustrated in Figure 3.2(f).

At this point, a heavier and deeper implant (roughly 180 to 200 nm) is performed to completely bring forth the n-channel source/drain and body ties for the n-wells (shown in Figure 3.2(h)). The lightly doping implants next to the channel area are preserved, attributed to the spacer protection. The same steps are repeated again for p-channel source/drain but with everything reversed. Finally, in order to lower the electrical resistivity of the source/drain and gate regions, a thin but conductive silicide film (Ni, Co or Ti film) is covered on the top of these regions, so-called salicide approach [42]. At this point, a CMOS device is completed.

3.1.2 3D Device Structure Simulation Methodology in TCAD

Technology computer-aided design (TCAD) tools are widely used by semiconductor device engineers to emulate process flows and simulate various kinds of device. TCAD tools help designers to predict device's electrical, thermal and other properties under different operating conditions without actual fabrication, providing that proper calibration has been done [43]. TCAD simulates the electron-hole transportation in the device using physical models [44]. The advantage of TCAD simulation is that avoids effects of limited factors due to the characteristics of basic physical models, further improving accuracy of results [45]. Simultaneously, a fully 3D process and device simulation is easy to emulate and perform using TCAD tools. The commercial TCAD package Sentaurus from Synopsys [46] was used to perform 3D device emulation and simulation in the work reported in this thesis to obtain detailed physical insights into their operation. Sentaurus, a suite of TCAD tools (shown in Figure 3.3 [81]), simulates the fabrication, operation and reliability of semiconductor devices. The physical models are used in the Sentaurus simulators to represent the wafer fabrication steps and device operation, providing a way to explore and optimise the new semiconductor devices.

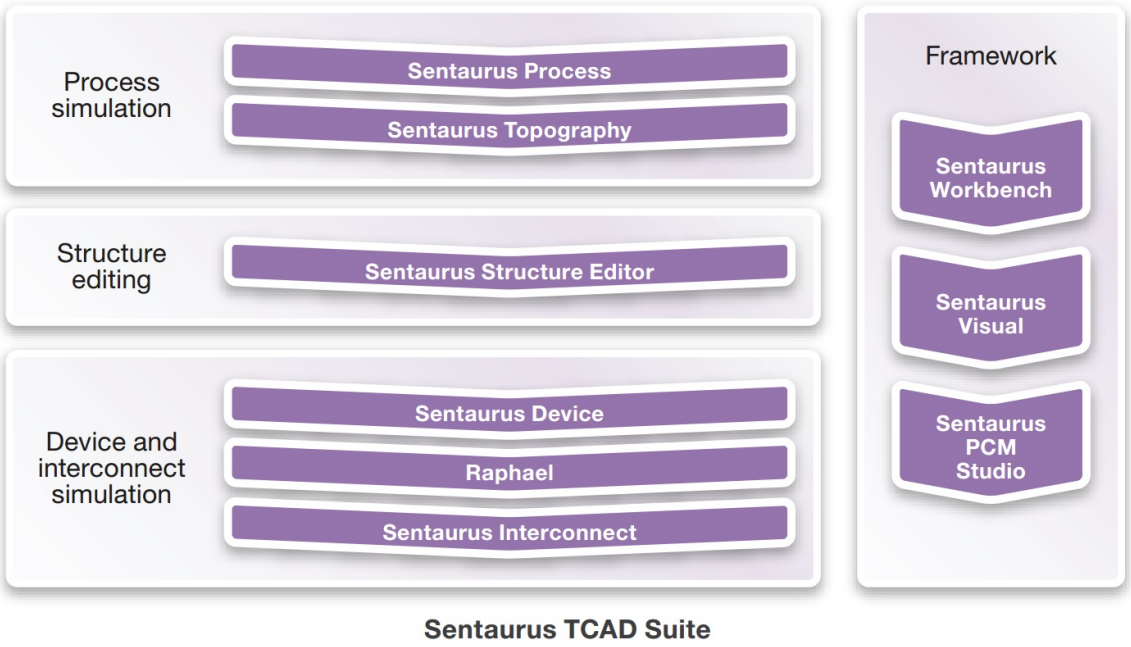


Figure 3.3: The framework of Sentaurus TCAD Suite [81].

The Sentarus TCAD provides full-flow 2D and 3D process and device simulation flows, with advanced structure generation, meshing and numeric analysis. The Sentarus TCAD supports silicon and compound semiconductor technologies, covering a broad range of semiconductor applications [81].

- Sentaurus Process (SP): Sentaurus Process simulates the fabrication steps in silicon process technologies in 2D and 3D [81]. A predictive framework to simulate a broad spectrum of technologies from nanoscale CMOS to high-voltage power device is provided by SP. This tool is equipped with a set of advanced process models, which include default parameters calibrated with data from equipment vendors. With Sentaurus Process, users can easily simulate process modules and integrate them into complete front end of line process flows. An advanced set of oxidation, diffusion, implantation, and mechanics models, combined with robust mesh generation and structure-editing capabilities, cover important process modules such as ultrashallow junction formation, high-k/ metal gate, and strained silicon. In addition, SP also offers a kinetic Monte Carlo (KMC) simulator for atomistic simulations of the interactions of dopants with point defects and extended defects in silicon. The more details of SP can be found in SP manual [85].

- Sentaurus Structure Editor (SSE): Sentaurus Structure Editor is a 2D/3D device editor which builds and edits device structures using geometric operations [81]. SSE has two

ways to edit the device structure. One way uses the graphical user interface (GUI) of SSE which features a command-line window in which script commands corresponding to the GUI operations are displayed. Alternatively, a user can enter script commands directly at the command-line. 2D and 3D device structure are created geometrically using 2D or 3D primitives such as rectangles, polygons, cuboids, cylinders, and spheres. 3D regions can also be created by extruding 2D objects or sweeping 2D objects along a path. A set of operations are provided to device structure process such as filleting, 3D edge blending, chamfering, and so on. These operations allow greater flexibility in structure generation. In addition, complex shapes are also generated by performing Boolean operations (union, subtract, intersect) between elements. After completing device structure creation, it needs to generate a virtual device structure through the meshing tool. This virtual device structure is described in TCAD tool by a TDR file, which contains the grid (or geometry) of the device information and the data field information. The grid of the device information consists of a description of the various regions, including boundaries, material types, and the locations of any electrical contacts [44]. The data field information mainly contains the properties of the device, such as the doping profiles, in the form of data associated with the discrete nodes[44]. For example, continuous properties such as doping profiles are represented on a sparse mesh and defined at a finite number of discrete points in space [44]. The doping at any point between nodes can be obtained by interpolation.

The mesh tools are composed of two mesh generators: Sentaurus mesh (SM) and Noffset3D [86]. Firstly, the SM generates high-quality spatial discretizations for complex 2D and 3D device. Two mesh generation engines are contained in SM: an axis-aligned mesh generator and a tensor-product mesh generator. The axis-aligned mesh generator produces Delaunay meshes. The meshes contain triangles only in 2D and tetrahedral in 3D [86]. The tensor-product mesh generator is used to generate meshes for Sentaurus Device Electromagnetic Wave Solver and Sentaurus Device Monte Carlo, which the meshes contain rectangular elements in 2D and hexahedral elements in 3D. Secondly, Noffset3D constructs Delaunay meshes using surface-adapted, anisotropic, mesh layers [86]. This mesh generator creates triangles and rectangles in 2D, and tetrahedral in 3D. It uses an advancing front method to create layers of elements from designated surfaces and interfaces, and fills the remainder with an isotropic mesh [86]. The more details of mesh tool can be found in mesh tool manual [86].

- Sentaurus Device (SDEVICE): Sentaurus Device simulates the electrical, thermal, and optical characteristics of silicon and compound semiconductor devices in 2D and 3D [81]. SDEVICE supports the design and optimisation of various semiconductor technologies including nanoscale CMOS, FinFET, thin film transistors (TFTs), flash memory, SiGe heterojunction bipolar transistors (HBTs), large-scale power devices, compound semiconductors, CMOS image sensors, and solar cells. SDEVICE incorporates an extensive set of physical models and material parameters, and supports DC, AC, transient and harmonic balance analysis. The electrical behaviour of a single semiconductor device in isolation or several physical devices combined in a circuit is simulated numerically by SDEIVCE. Terminal currents, and charges are computed based on a set of physical device equations/models that describes the carrier distribution and conduction mechanisms [44]. Generally, user can specify a series of equations (or numeric methods) to control device simulations so as to obtain different solutions and simulations, which include box method algorithm [44], Harmonic balance (HB) analysis [44], Backward Euler Method [44], etc. For example, in order to perform transient simulation, the discretization of transient equations, simple Backward Euler (BE) method and composite trapezoidal rule/backward differentiation formula (TRBDF) are used in SDEVICE [44]. A typical tool flow with device simulation using SDEVICE is shown in Figure 3.4 [44].

- Sentaurus Visual (SV): Sentaurus Visual provides users with a interactive 1D, 2D, and 3D visualization and enable the postprocessing of output data to generate new curve and extract parameters [81].

3D Device Structure Creation

In order to intuitively understand 3D device structures, an example of 3D NMOS device structure (shown in Figure 3.5) created from SSE is demonstrated here, which the steps required to arrive at the fully 3D formed structure is described in the following (see Figures 3.7 and 3.8). A 2D cross section of this NMOS device, depicted in Figure 3.6, cuts from the 3D structure. Since the device structure is fully parameterized, various device geometry sizes and doping profiles in the device are easily emulated through the modification of these parameters (the definition of these parameters are listed in the Table 3.1).

The 3D device structure in SSE is generated by a series of geometry-editing operations. These geometry generation steps are typical for CMOS fabrication and include:

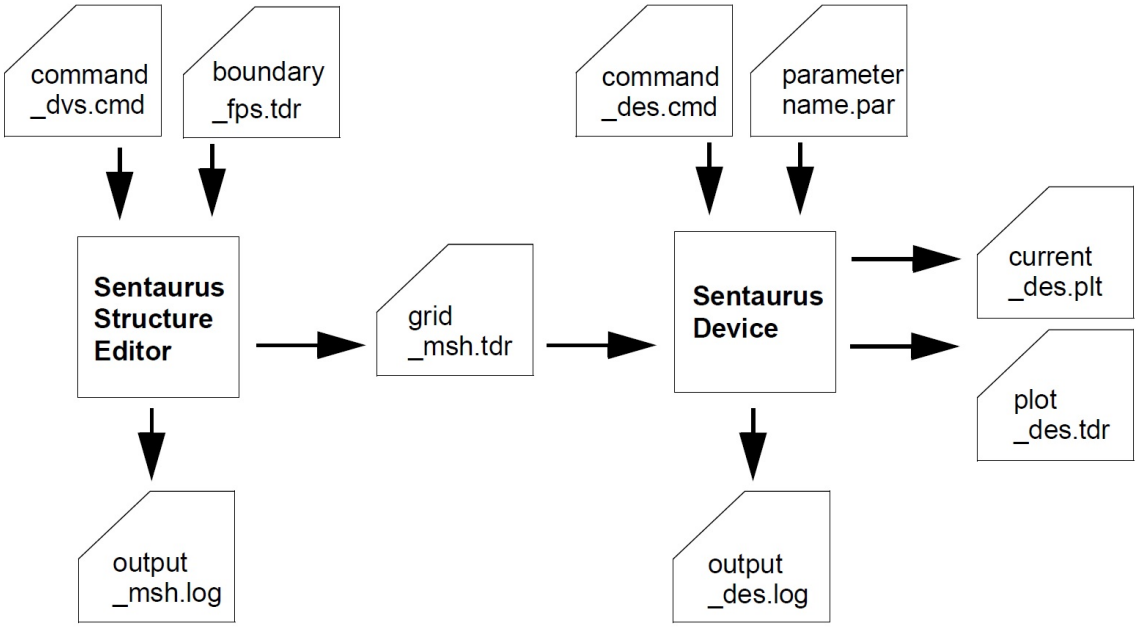


Figure 3.4: Typical tool flow with device simulation using Sentaurus Device [44]. The creation of a device structure by process simulation (SP) or structure emulation (SSE) is followed by meshing using SM. Control of mesh refinement is handled automatically via the file _dvs.cmd. Then, SDEVICE is used to perform the electrical characteristics of the device simulation. Finally, the output from device simulation in 2D and 3D is visualized and the electrical characteristics are plotted by SV.

Table 3.1: TCAD Device Parameters

Parameter	Description
L_g $[nm]$	Length
W $[nm]$	Width
L_{sp} $[nm]$	Width of the device spacer
L_{sd} $[nm]$	Width of the device Source/Drain
H_s $[nm]$	Substrate thickness
T_{ox} $[nm]$	Thickness of the gate oxide
N_a $[cm^{-3}]$	Concentration of the body/channel doping
Nd_po $[cm^{-3}]$	Concentration of polygate doping
Nd_sd $[cm^{-3}]$	Concentration of Source/Drain doping
Nd_ex $[cm^{-3}]$	Concentration of extension doping
Xj_sd $[\mu m]$	Depth of the Source/Drain doping junction
Xj_ex $[\mu m]$	Depth of the extension doping junction

- Substrate definition
- Gate oxide formation

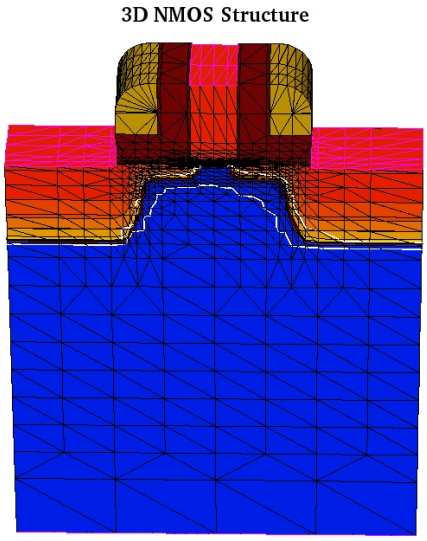


Figure 3.5: A virtualized 3D NMOS transistor structure with meshing information, created by TCAD tool.

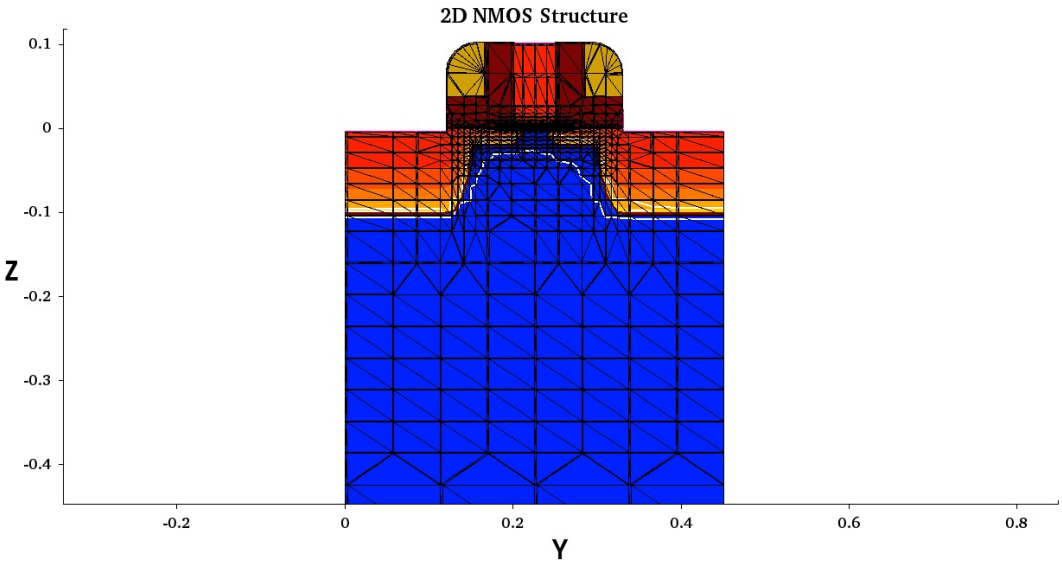
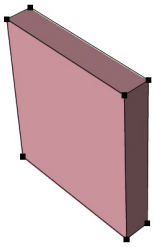


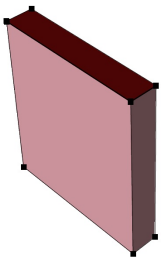
Figure 3.6: 2D cross-section of front view NMOS transistor cut from Figure 3.5 shown NMOS 3D structure.

- Polygate formation
- Extension spacer formation
- Nitride spacer formation
- Contacts formation

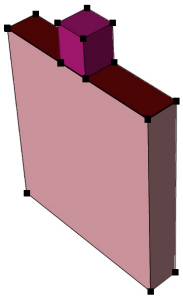
Each one of the abovementioned operations typically is composed of one or more steps, including basic geometry objects and operations with them. A step-by-step outline is given



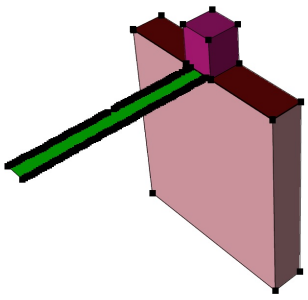
(a) The first step of the process is substrate definition.



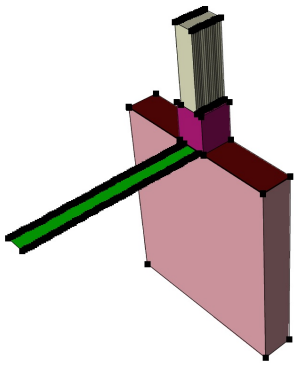
(b) The next step is the growth of SiO_2 oxide layer on the surface.



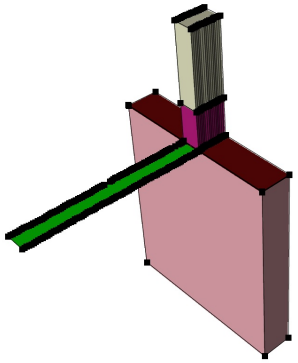
(c) The poly-silicon is pattern and generated to form transistor gate.



(d) This step defines a mask preparing for LER generation.

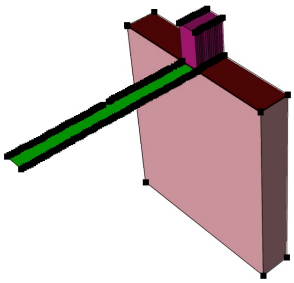


(e) A layer of photoresist is deposited over the gate surface for next step process.

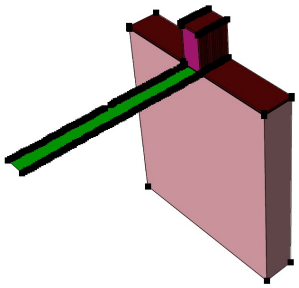


(f) Gate roughness is created by first depositing and developing photoresist above the gate surface to be masked, then etching gate poly silicon to form roughness.

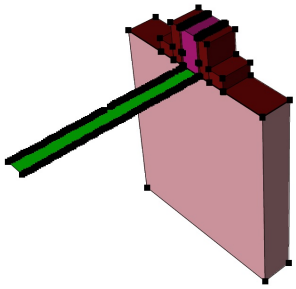
Figure 3.7: 3D NMOS device with LER emulation steps (continued on Figure 3.8).



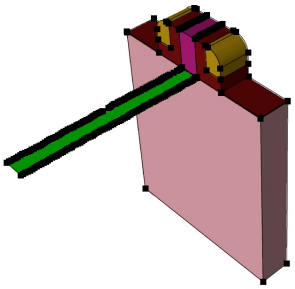
(g) The photoresist is removed.



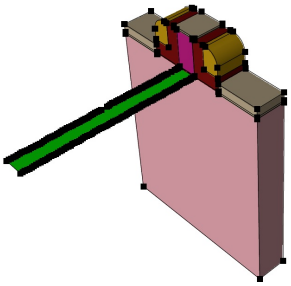
(h) A thin SiO_2 spacers on either side of gate is grown.



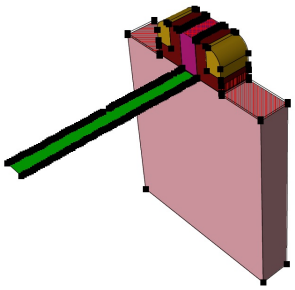
(i) Extension spacer formation continuing



(j) The grown of nitride spacers and spacer rounding



(k) Contacts setting through creating cuboid metal region on the source/drain and gate surface.



(l) Removing the newly created metal region and finishing contacts defining.

Figure 3.8: Continued 3D NMOS device with LER emulation steps.

in Figures 3.7 and 3.8 to illustrate how to emulate the device structure in SSE. In addition to defining the geometry of a structure, the step of defining doping profiles and generating the mesh for device simulation are also crucial for device emulation due to their direct affect on simulation results to reflect the accuracy of device characteristics. The corresponding doping concentration parameters in the structure are listed in Table 3.2. After the various parts of the structure have been specified by the corresponding doping profile and refinement parameters, the meshing engine (Sentaurus Mesh) is called to generate a virtual structure to prepare for the device simulator. This virtual structure is an approximation of a real device

[47]. The final device structure with the doping information is given in Figure 3.9.

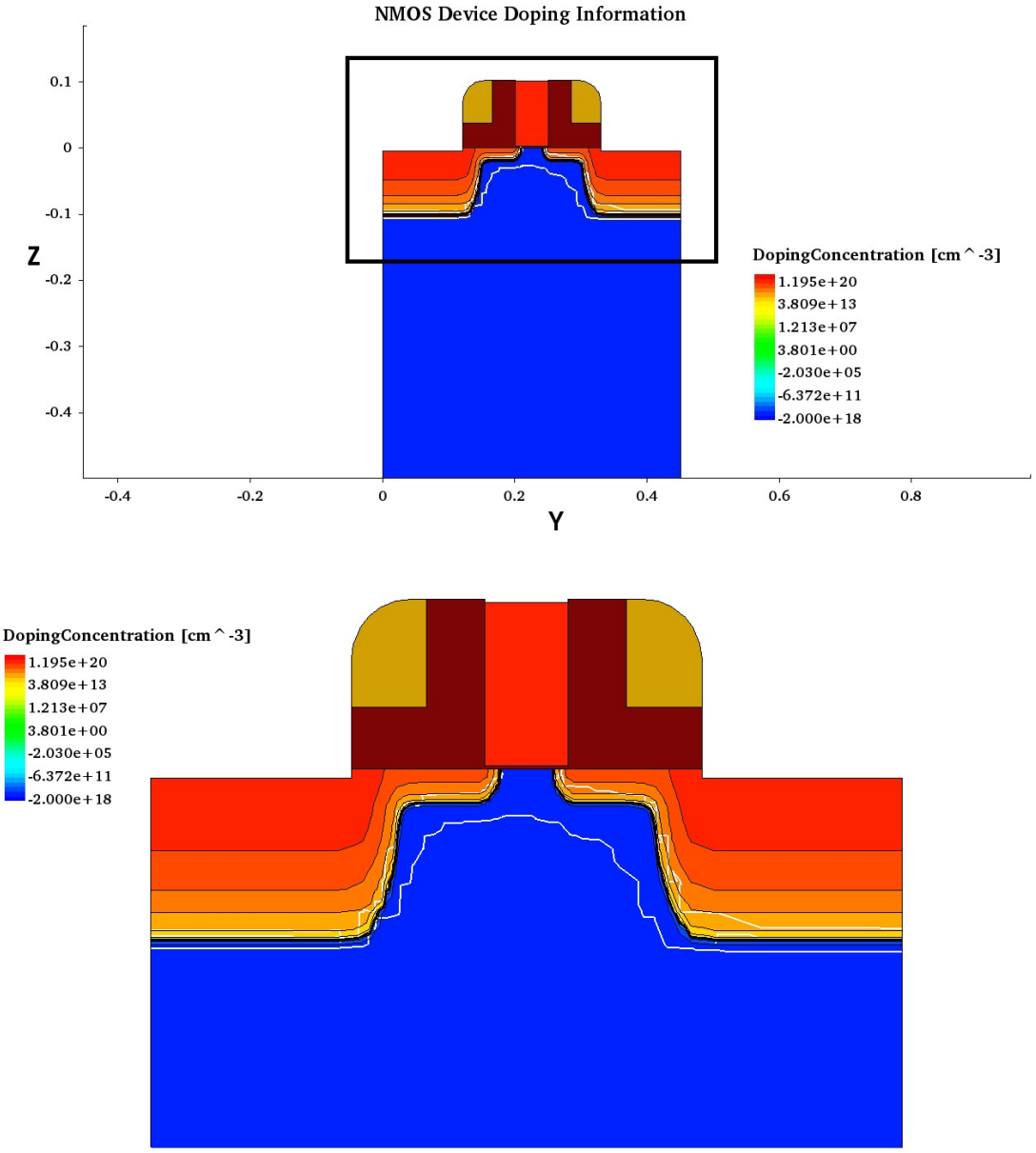


Figure 3.9: The upper sub-figure is front view of 2D NMOS device with doping information. The bottom sub-figure clearly shows the channel and source/drain doping concentration.

Device Physical Simulation

For device simulation, the virtual device structure obtained from mesh processing is passed to the device simulator (Sentaurus device) to perform device simulation. Sentaurus device (SDEVICE) is a fully featured 2D and 3D device simulator that incorporates a set of advanced physical models and robust numeric methods for the simulation of most types of semicon-

Table 3.2: TCAD Device Doping Concentration Parameters

Parameter	Description	Value
N_a $[cm^{-3}]$	Concentration of the body/channel doping	2×10^{18}
Nd_po $[cm^{-3}]$	Concentration of polygate doping	1×10^{20}
Nd_sd $[cm^{-}3]$	Concentration of S/D doping	1×10^{20}
Nd_ex $[cm^{-3}]$	Concentration of extension doping	2×10^{19}
Xj_sd $[\mu m]$	Depth of the S/D doping junction	0.1
Xj_ex $[\mu m]$	Depth of the extension doping junction	0.02

ductor devices [47]. The SDEVICE is used to simulate the drain current as a function of the gate voltage at a low drain bias (drain bias of 0.05 V) and a high drain bias (drain bias of 1.2 V), and then device properties (such as threshold voltage, leakage current) are extracted from these simulation results. The following sets of models are used in the SDEVICE.

- Drift-diffusion transport model

  The drift-diffusion transport model is the default carrier transport model in the SDEVICE which is suitable for low-power density devices with long active regions. This model is derived from the Boltzmann transport equation [44][48]. The current densities for electrons and holes in the drift-diffusion model are represented by Equations 3.1 and 3.2.

$$\vec{J_n} = \mu_n(n\nabla E_C - 1.5nkT\nabla \ln m_n) + D_n(\nabla n - n\nabla \ln \gamma_n) \tag{3.1}$$

$$\vec{J_p} = \mu_p(p\nabla E_V - 1.5nkT\nabla \ln m_p) + D_p(\nabla p - p\nabla \ln \gamma_p) \tag{3.2}$$

  Where n and p is the electron and hole density, respectively; μ_n and μ_p is the electron and hole mobility; D_n and D_p respectively, diffusion coefficient, are given by the *Einstein relation*, $D_n = kT\mu_n$ and $D_p = kT\mu_p$. m_p and m_n is the spatial variation of the effective masses. For Boltzmann statistics, γ_p=γ_n=1.

- Philips unified mobility model

  The Philips unified mobility model, proposed by Klaassen [49], unifies the descriptions of majority and minority carrier bulk mobility. The temperature dependence of the mobility, screening of the ionized impurities by charge carriers, electron-hole scattering and clustering of impurities are considered and incorporated in this model [49]- [50].

3.2 Motifs Design Methodology

3.2.1 Layout Consideration and Variability

To understand the impact of different transistor layouts on device performance and variability, several layout-dependant factors are analysed for characterizing active-extension-induced variation due to STI stress, poly-pitch-induced variation due to impact of lithography and the impact of stack layout and channel segmented configuration on transistor performance.

Shallow Trench Isolation (STI) Effect

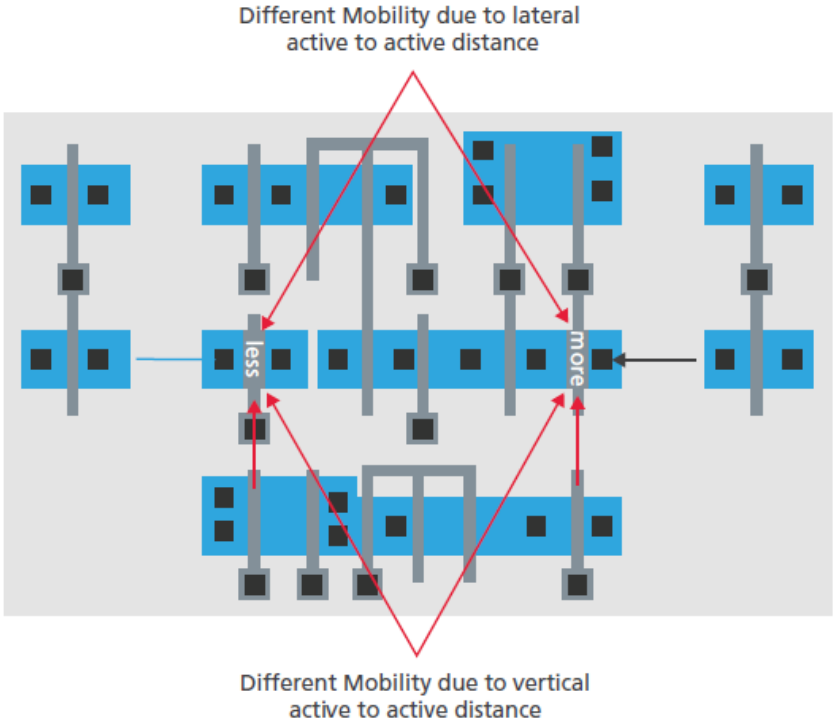


Figure 3.10: This standard cell layout demonstrates how characteristic of device in the cell is affected by stress due to STI width. STI width is context dependent which determined by the distances of devices in neighbouring cells (active-to-active spacing). Wider STI leads to greater distances and more compressive stress, results in PMOS performance increase [51].

Since *250 nm* process node STI (shown in Figure 3.8 (a)) has become the standard device isolation scheme, which trenches are cut into the bulk material and filled with SiO_2 to separate transistors and NMOS/PMOS regions [8][42]. Due to the thermal expansion mismatch between Si and SiO_2 mechanical compressive stress induced by STI changes the lattice spacing and affects transistor electrical behaviour such as mobility, saturation velocity and threshold

voltage. Therefore, STI-induced stress becomes an inevitable concern even if STI unintentionally induce stress. The STI-induced stress decreases with the distance between a transistor and the nearest STI edge [51]-[53]. This distance is called the length of diffusion (LOD). The width (called active-to-active spacing) of the STI also determines how much stress is applied (illustrated in Figure 3.10 [51]). When the amount of STI is small, transistors should be laid out very close, and vice versa. Due to the impact of compressive stress, wider STI will enhance PMOS performance and degrade NMOS performance, while narrower STI will work enhance NMOS and degrade PMOS [51].

Length of Diffusion (LOD)

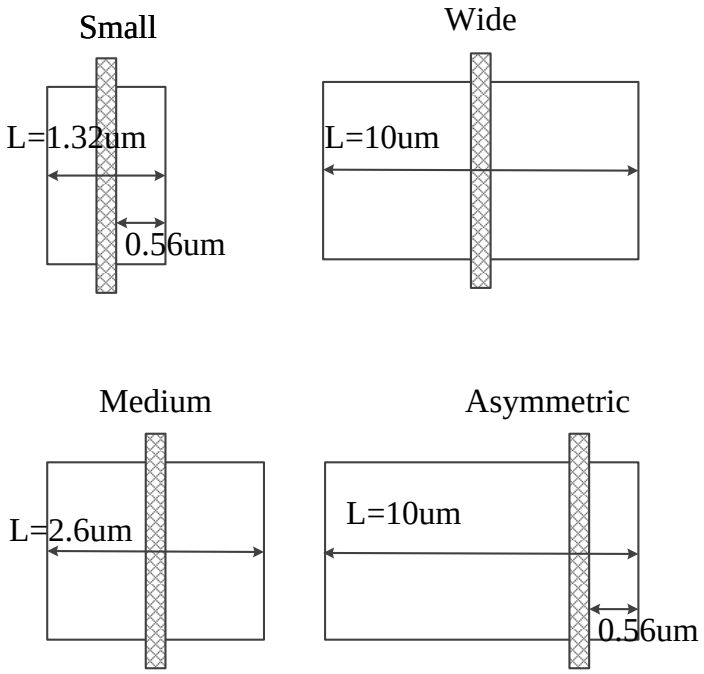


Figure 3.11: In order to investigate the impact of LOD on device characteristics, four test layout structures with different LOD (gate length 0.2 μm and width 20 μm) are used (These test layout structures are taken from [54]). These layouts are classified two types: symmetric and asymmetric placement; And symmetric placement includes three layouts with different LOD.

LOD is another issue related to STI. As previously stated stress placed on the transistor gate depends on the LOD or source/drain regions. To study the impact of LOD on device performance a set of test layouts with different LOD (shown in Figure 3.11) were tested and

measured in literature [54].

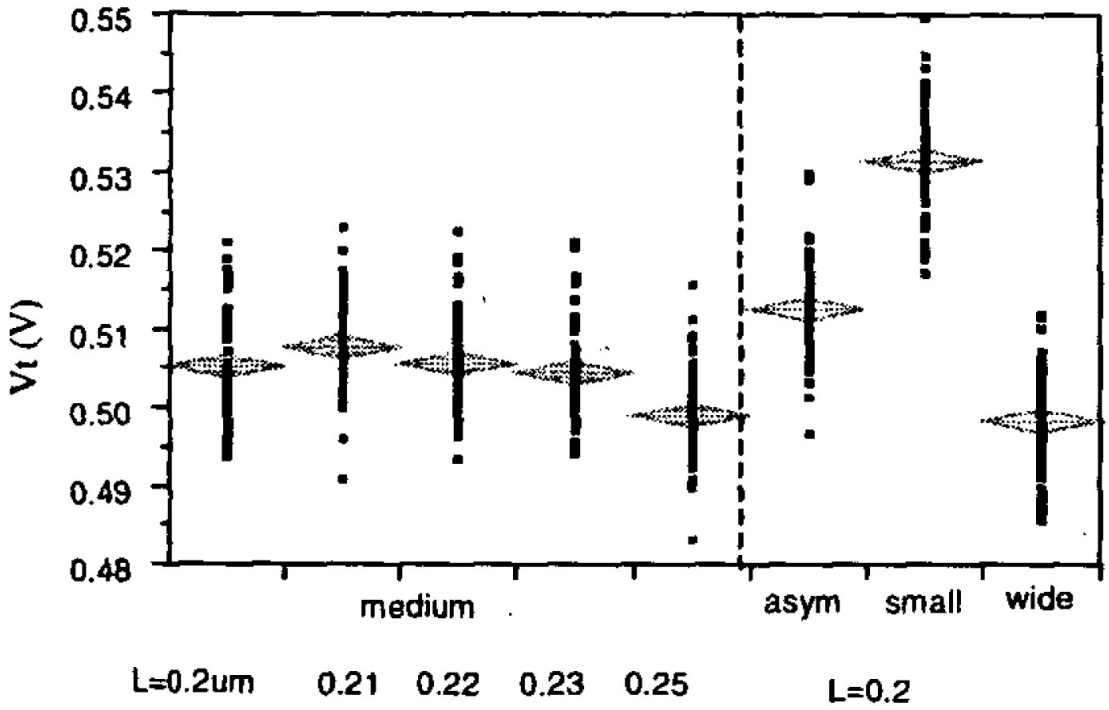


Figure 3.12: NMOS V_{th} vs. device layout (device layout as shown in [54]). The V_{th} of device with small size diffusion has a 25 mV increase over the medium test layout. The V_{th} of device is significantly sensitivity to layout placement.

These test layouts produced with *0.2* μm process. Based on LOD these test devices are classified as small device, medium device, wide device and wide device with asymmetrical placement. Additionally, medium devices with different gate length were also tested. Figures 3.12 and 3.13 show the impact of different LOD on NMOS/PMOS device threshold voltage (V_{th}). Comparing the NMOS devices and PMOS devices, the threshold voltage of NMOS device is significantly sensitive to device layout. In contrast, PMOS devices exhibit little sensitivity to layout [54]. According to literature [54] analysis, compressive stress in silicon due to the piezoresistive effect [55] causes a reduction in electron mobility, while it causes a minimal effect on hole mobility [56]-[57]. Therefore, NMOS device characteristic will be significantly affected by trench stress than PMOS [54].

Shared Source/Drain Diffusion

A diffusion region shared in stack or cascade configuration is a common layout style in standard cell design, depicted in Figure 3.14. The shared diffusion is usually used to merge transistors for circuit layout geometry optimisation. The transistors merged result in sub-

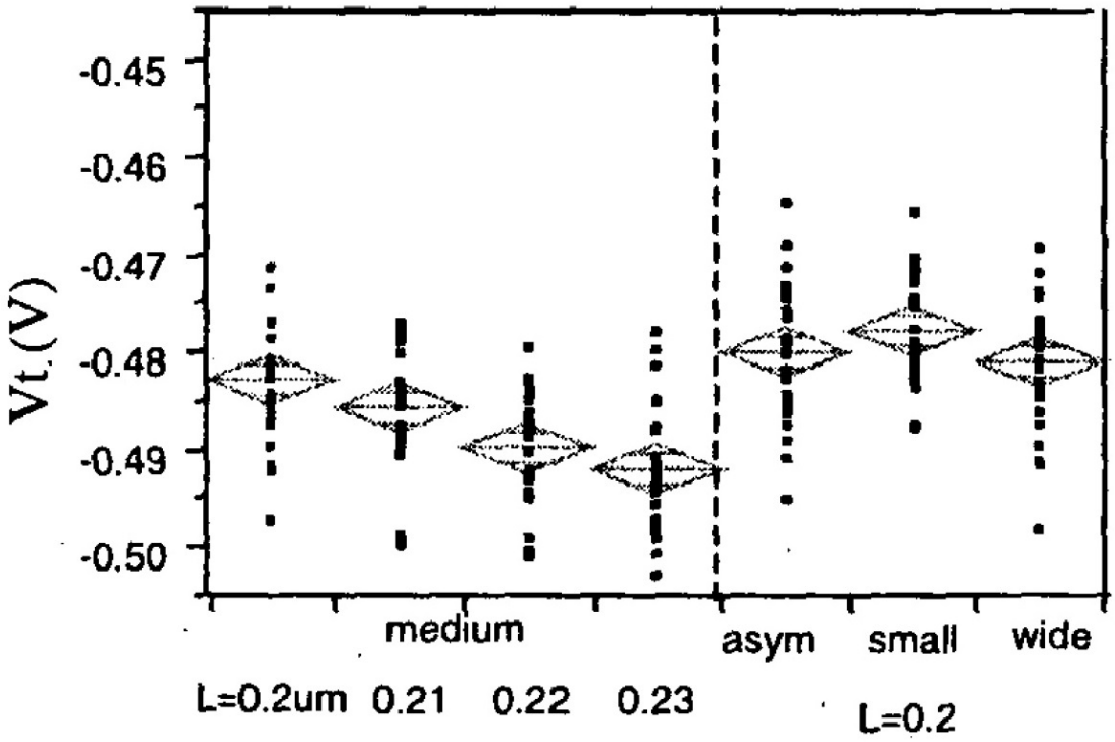


Figure 3.13: PMOS V_{th} vs. device layout (device layout as shown in [54]). Comparing to NMOS devices, PMOS exhibits little sensitivity to layout placement variations since stress has a minimal effect on hole mobility [56]-[57].

stantially reduced parasitic source/drain capacitance in this area due to reduction of the minimum distance between poly gate wires [58]. In addition, this layout style is also useful to investigate mismatch in two identical devices due to variability induced by random variability sources [59].

Segmented Channel

In order to improve the short-channel effect due to enhanced electrostatic integrity, a transistor with a continuous width can be segmented into multiple stripes of equal width [60][61]. Figure 3.15 depicts an example of a continuous channel separated to segment. The continuous channel is isolated by a set of very shallow trench isolation (VSTI) dielectric material. The segmented channel brings the benefit of device performance improvement. From the device level view, because of fringing electric fields via VSTI the gate control of the channel potential is enhanced and the device is more robust to the short-channel effects [58][61]-[64]. At the circuit level, transistor stacking has capability for dynamic threshold voltage control that can effectively reduce the leakage current in high performance logic and memory design

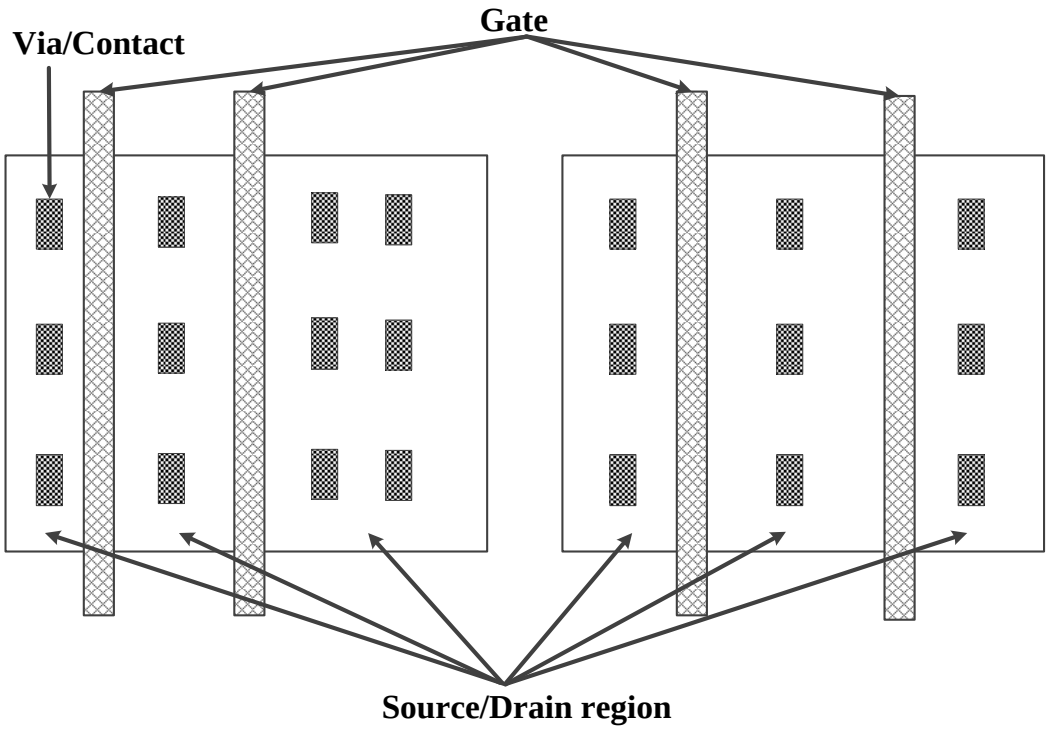


Figure 3.14: Test structures with shared source/drain region is proposed. The left test layout is designed to investigate the mismatch in two identical devices changing as two gates distance and the right layout placement is used for evaluating the relationship of parasitic capacitance in shared source/drain area.

[58][64]. The idea of a proposed O shaped layout placement is derived from the benefits of device and circuit performance improvement due to the channel segmented. The details of the O shaped layout placement (shown in Figure 3.16) is introduced in the next section.

3.2.2 Motif Concept

A recent suggestion to allow circuit to be more effectively optimised in the presence of transistor variation and improve the design layout regularity is through the use of layout motif. As stated in [82]-[83], a layout motif that is a block of transistors on a shared region of active (shown in Figure 3.17 [82]), was proposed to reduce the effect of stochastic variability on current variation and increase the layout regularity so as to improve resistance to systematic variability. Parallel and series configurations of motifs, with one to five NMOS transistors, were simulated with different gate length and transistor widths. Layout motifs provide a way to examine the effect of atomistic variations at the circuit level and allow us to see how atomistic variations affect the logic circuits, without having to simulate entire standard cell

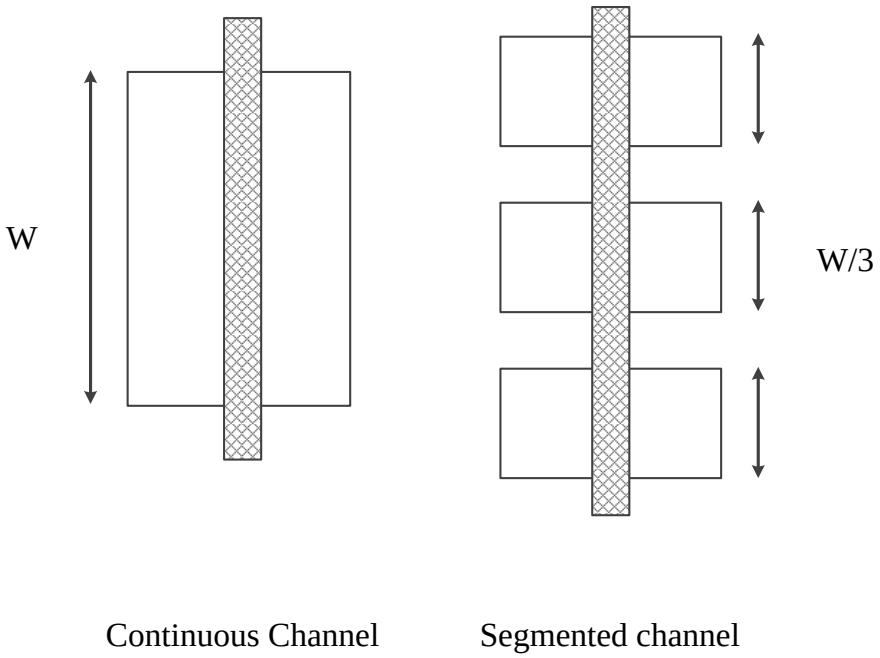


Figure 3.15: Test structures with segmented channel [59]. In this case, a continuous channel device is separated into three devices with the same small width channel. The segment structure inspires one of layout placement structure used in this thesis, which *O* shaped layout structure will be described in next section.

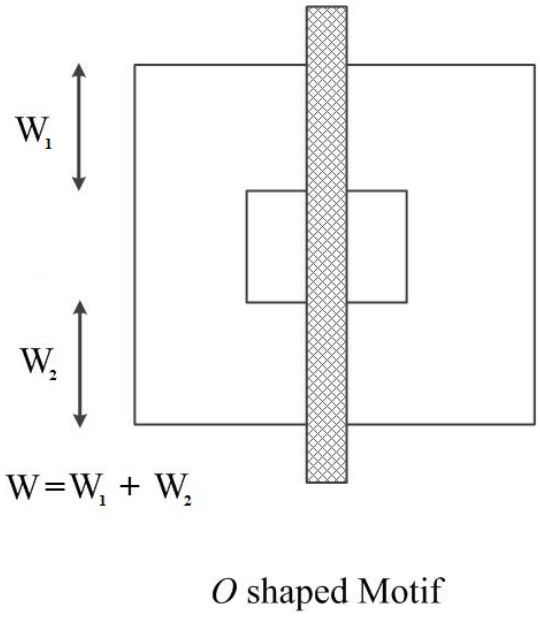


Figure 3.16: The proposed *O* shaped layout structure (Top view).

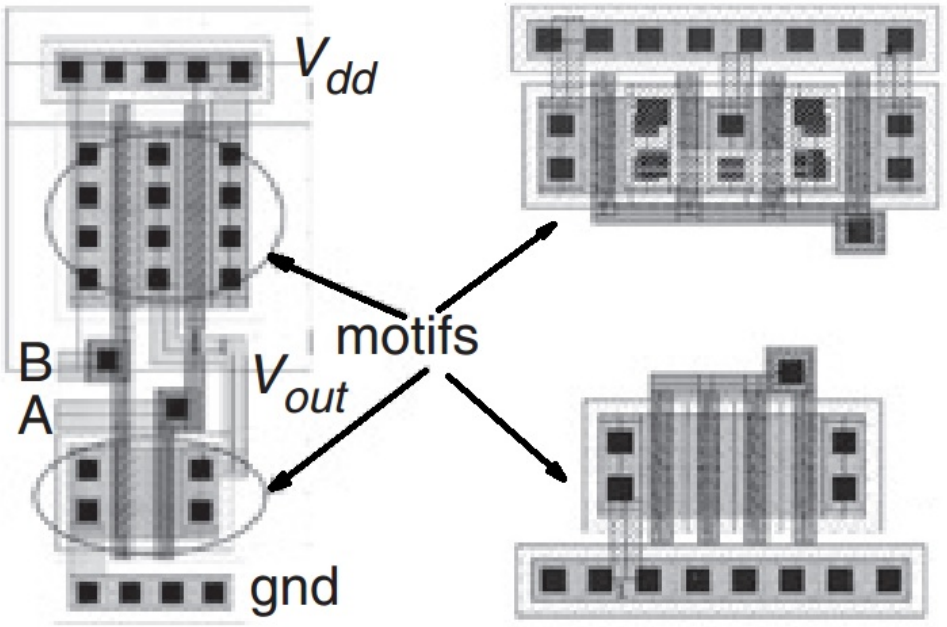


Figure 3.17: Examples for layout motif. The left two layout motifs come from NAND2 gate compact layout.

logic. In addition to motif used for digital circuit optimisation, layout motif is also used in analogue design to minimize device mismatch problem. As [84] introduced, the mismatch between transistor was dependent on their relative channel orientation. The appropriate use of layout motif in analogue circuit (shown in Figure 3.18 [84]) could be used to find a good compromise between matching and parasitic effects.

Motifs are a generic scheme that are defined as repeating patterns of fundamental geometrical forms, which are used as design units which often appear as sub-portions of different cells or the same cell. Motifs sit between single transistor/portion of transistor structure and custom standard cell layouts in the circuit hierarchy, and reflect the increasing regularity that is required in the state-of-the-art CMOS layout to facilitate practical fabrication at the nanoscale, especially often used by foundry manufacturing. The layout motif can be classified into device layout motif and circuit layout motif. Generally, a motif can be as small as a single transistor or portion of a transistor repeating structure (device layout motif), such as single transistor and part of single fin structure in FinFET device (illustrated in Figure 3.19 [65]). Alternatively, a motif can be as large as coupled clusters of transistors-groupings from which wider devices and more complex circuits can be constructed (circuit layout motif), such as an optimised 6T SRAM cell (shown in Figure 3.20 [66]). This thesis only focuses on device layout motif study and investigates the influence of device layout motif on circuit

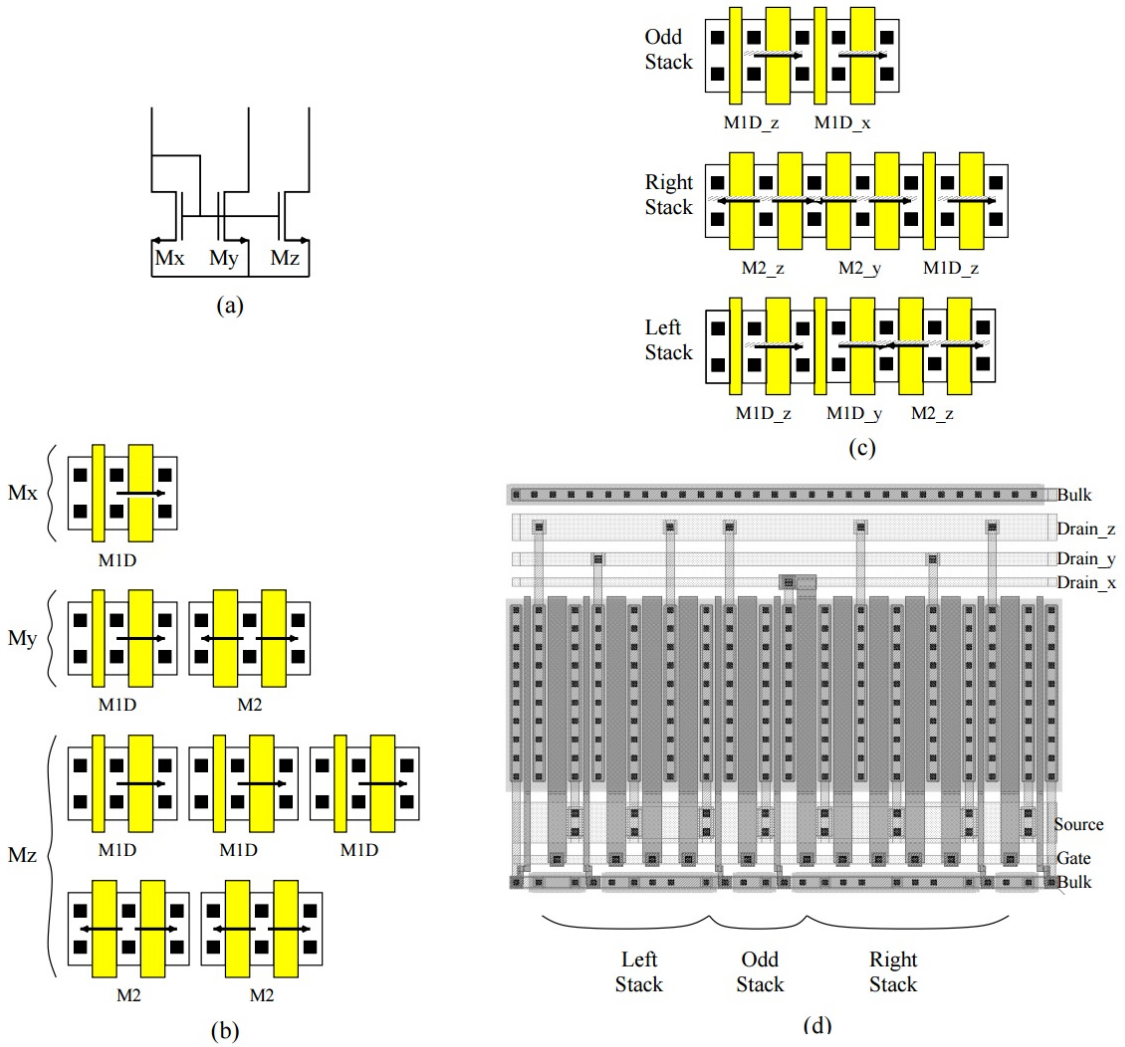


Figure 3.18: As an example, consider a current mirror composed of three transistors M_x, M_y, M_z shown in (a) [84]. The assigned motifs of each transistor are shown in (b). Based on literature [84] introduced optimisation algorithm, one motif from each is placed in the odd stack; and the other two stacks are then composed by taking one motif alternatively from each transistor as shown in (c). (d) shows the physical layout of the current mirror stack after abutting the three elementary stacks shown in (c) [84].

optimisation.

Optimisation of CMOS circuitry at a single transistor level has always been computationally expensive, although industry is increasingly reliant on fine grained SPICE simulation to adequately model critical delay paths in modern designs. However, when the possible variability of each transistor must be taken into consideration, the computational cost for critical path analysis becomes prohibitive: such variations can be local (e.g. caused by transistor random dopant distributions or line edge roughness) or non-local (e.g. caused by coupled stress induced effects between devices) in nature. Characterising motifs and using them as

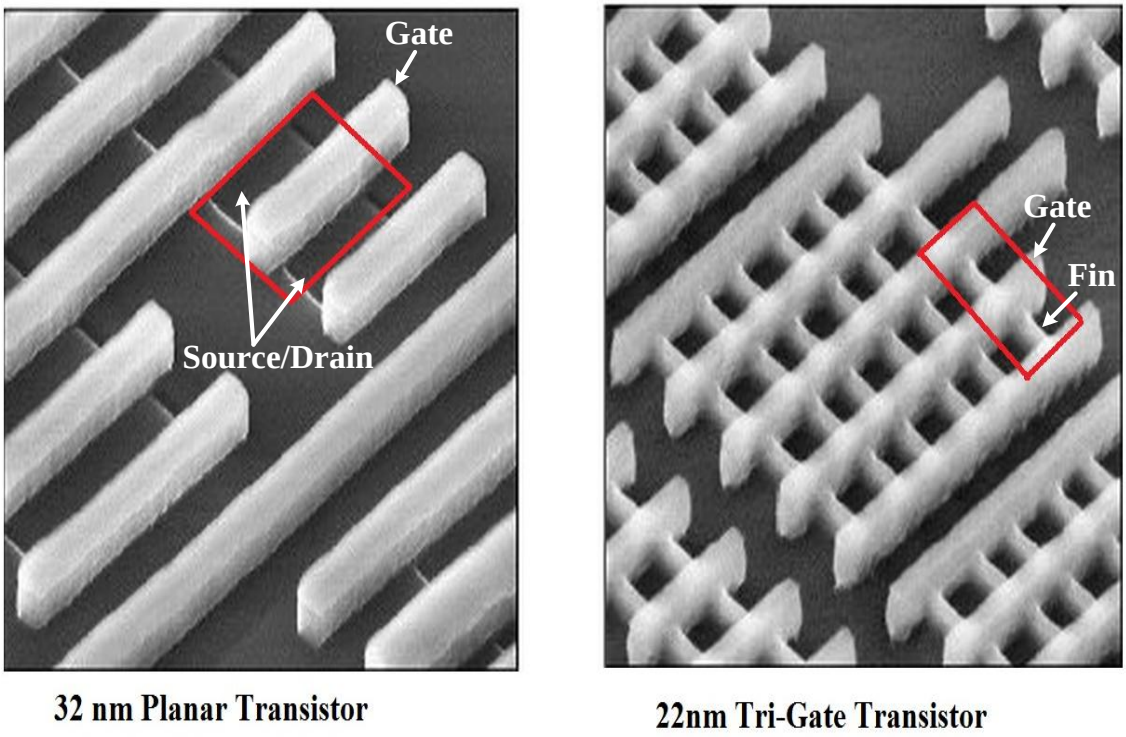


Figure 3.19: Two examples of motif (shown in red rectangle) are taken from SEM image of 32nm planar transistors and 22nm tri-gate transistors [65], respectively. As motif definition described, motif can be as small as a part of unit in tri-gate transistor or a single transistor.

larger grained fundamental units in the design process reduces the computational overhead of designs to more manageable levels. It also allows for some key non-local effects to be encapsulated within the motif parameterisation. Critically, the granularity of motif design is still below that of custom cells, meaning that critical delay path calculations of sufficient accuracy can still be made. An example of a 6T SRAM suffering mismatch as result of the performance degrade due to a poor layout and printability is shown in Figure 3.21 [67]. The repeating units in SRAM are considered as various motifs analysed that provides an efficient way to quickly identify these motifs (repeating units) characteristic with low computational expenses, and also further improving the whole design layout regularity.

3.2.3 Motifs Design

Taking into account Section 3.2.1 mentioned layout factors, several motif structures, incorporating long diffusion area motif, multi-finger motif and O shaped motif, (shown in Figures 3.22, 3.23, and 3.24) were constructed to study and evaluate the effects of motif structure

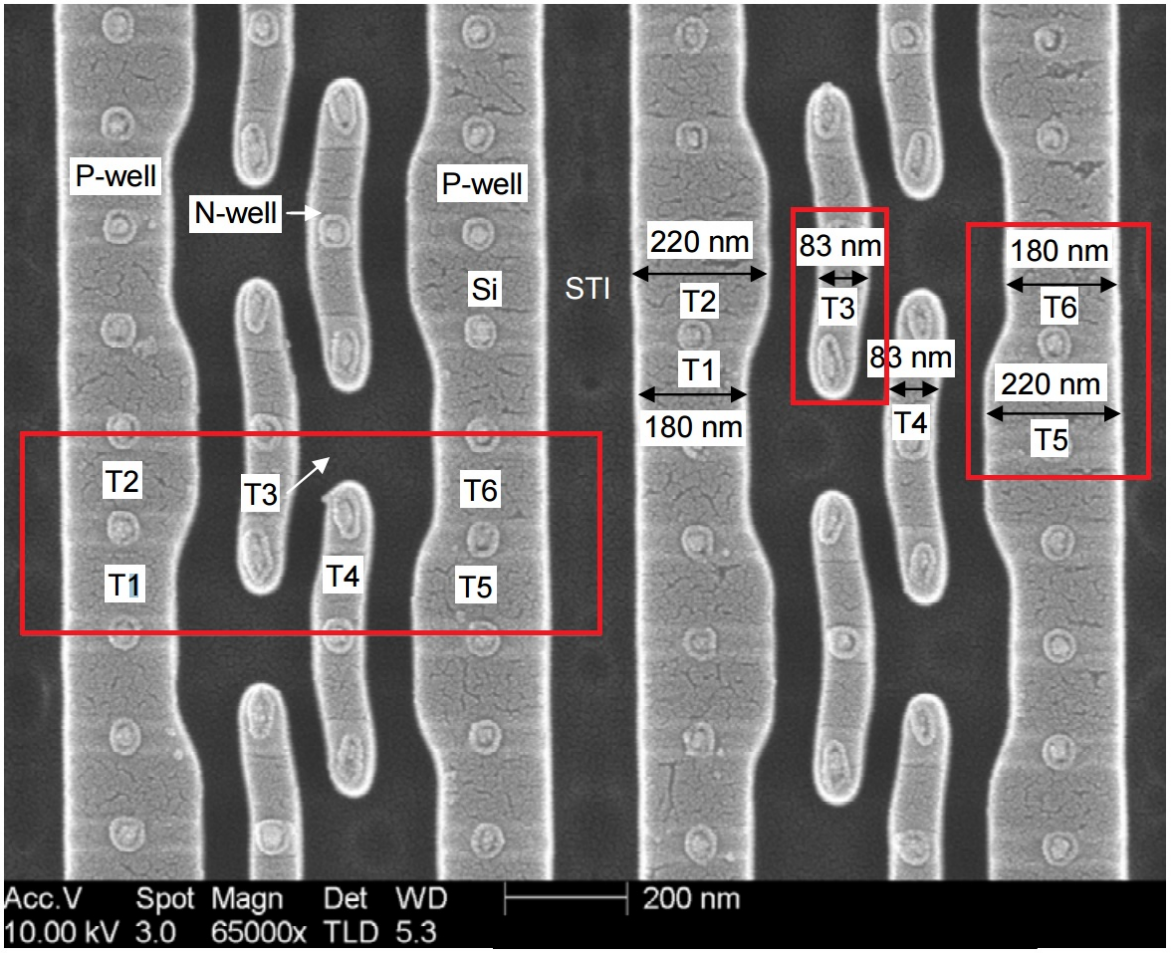


Figure 3.20: Several motifs are illustrated by red rectangles from SRAM SEM image [66]. Comparing to motifs illustrated in Figure 3.19, motifs can be as large as the whole SRAM or the transistors-groupings from SRAM layout.

on device performance and variation. These motifs were developed in a general-purpose *50 nm* CMOS technology due to the fact that *50 nm* technology is widely used in device research and is well understand. These motif structures were emulated in fully 3D process and performed TCAD device simulation. Table 3.3 lists these motifs geometry parameters used for TCAD simulation.

The long diffusion area motif has a longer S/D diffusion area than standard transistor, which has been observed to cause different STI stress placed on transistor gates. In addition to different strain in the transistor, longer S/D diffusion area leads to the series resistance of S/D increasing, which also further affects device performance (Figure 3.25 shows the comparison of the long diffusion motif with different LOD). The multi-finger motif is a common layout in circuit layout design. Here, the multi-finger motif is mainly for characterizing poly-pitch effects. The *O* shaped motif is a relatively complex structure, which is for characterizing the

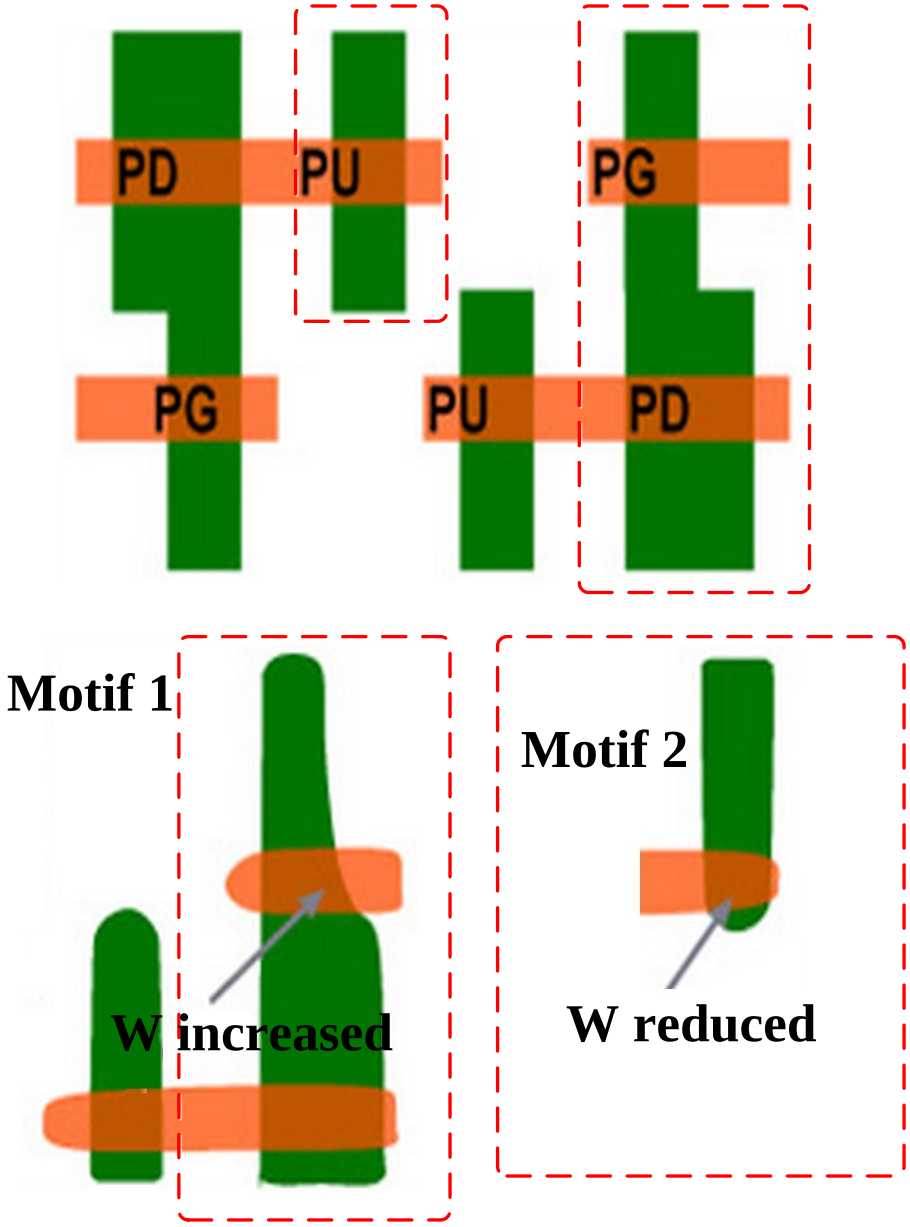


Figure 3.21: Two motifs are proposed based on the fundamental repeating geometrical forms from SRAM print image [67]. Some of unexpected layout placements caused the SRAM performance degradation (as the bottom sub-figure shown motifs layout) can be easily investigated by motif.

impact of segmented channel. In order to easily understand this extremely complex motif structure, the *O* shaped motif emulation process is outlined in Figure 3.26. The *O* shaped motif structure starts with a substrate definition (Figure 3.26(a)). Active areas are defined by photolithography and oxide/silicon etching (Figure 3.26(b)). The surrounding trenches and central hole are filled by oxide to form STI (Figure 3.26(c) and Figure 3.26(d)). Additionally, gate oxide is formed together with trenches fill processes. After gate oxide formation, the

Table 3.3: TCAD Device Geometry Parameters

Parameter	Description	Value
L_g $[nm]$	Length	50
W $[nm]$	Width (Long diffusion area motif)	80
	Width (Multi-finger motif)	160 (two small devices (W=80nm) parallel connection
	Width (O shaped motif)	160
$ActiveArea$ $[\mu m^2]$	Active area (Long diffusion area motif)	0.068
	Active area (Multi-finger motif)	0.072
	Active area (O shaped motif)	0.096
L_{sp} $[nm]$	Width of the device spacer	80
L_{sd} $[nm]$	Width of the device Source/Drain (Except Long diffusion area motif)	200
	Width of the device Source/Drain (Long diffusion area motif)	400
H_s $[nm]$	Substrate thickness	500
T_{ox} $[nm]$	Thickness of the gate oxide	2

poly gate is formed across the central STI (Figure 3.26(e)). The following steps are the same as standard device fabrication steps described in Section 3.1.2, form extension spacer (Figure 3.26(f)), nitride sidewall spacer (Figure 3.26(g)) and defining contacts (Figure 3.26(h)).

3.3 TCAD Statistical Variability Simulation Methodology

3.3.1 Line Edge Roughness Modelling

LER is strongly related to the layout patterning fidelity and layout proximity [68]. Every device in the standard cells has different LER due to different layout proximity. Standard cells are widely used in traditionally digital VLSI design, in particular repeating cells in standard cell design that any smaller changes to mitigate variation can result in significant improvement at the circuit level [68]. Thus, it is necessary to analyse the impact of LER on device performance. Many methodologies for modelling LER are described in [68]-[71]. Traditionally, the use of 2D device "slices" technique splits a 3D device with rough edges into multiple 2D devices with different gate lengths. For a given device width, the original 3D device is broken up into the N 2D segments with the appropriate width equals to LER

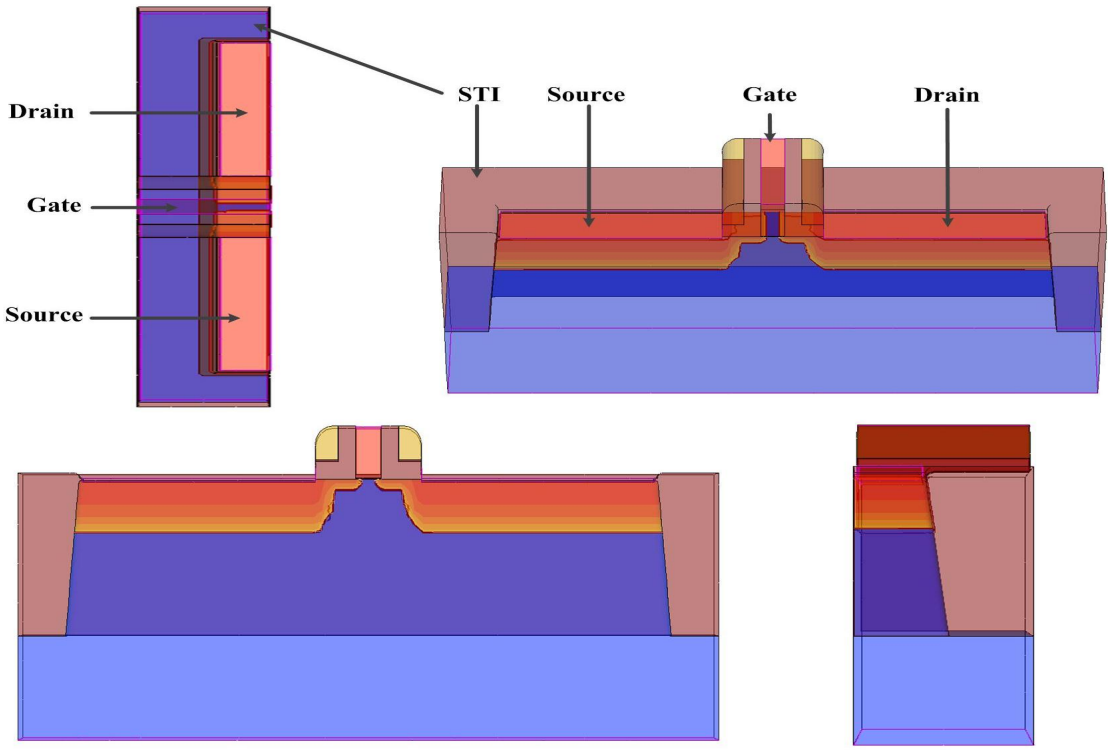


Figure 3.22: In order to investigate the impact of series resistance and stress placing on gate due to LOD on transistor performance, long diffusion area motif is proposed. Top view (upper left), Front view (low left), Side view (lower right), 3D view (upper right).

characteristic spatial period, which is shown in Figure 3.27 [27][70]. Each segment is modelled as a sub-transistor with correct assignment of narrow-width and different gate length [72]. A set of 2D segments simulation are simulated to capture the statistical characteristics of small 2D devices, which is easily implemented and the low computational burden through comparison to full 3D device simulation. The threshold voltage (V_{th}) of the 3D device is assigned to different segments. The drive current of the 3D device is calculated by adding up all of drain currents (I_d) and leakage current (I_{off}) for each piece together. Although gate slices technique is easily to model the effect of LER on 3D device with less computation expensive, this method has some limitations which affect the result accuracy. Firstly, due to the total output characteristics based on the linear superposition of currents, it requires that the drive current should be a linear function of V_{th}. However, the leakage has an exponential dependence on V_{th} that means the slicing method is not appropriate to apply in subthreshold region [72]. Additionally, since the gate slices method requires that each slice maintains the same direction from source to drain. Once the electrical field along the channel direction has significantly distortion, this method cannot provide a correct prediction under LER [72]-[75].

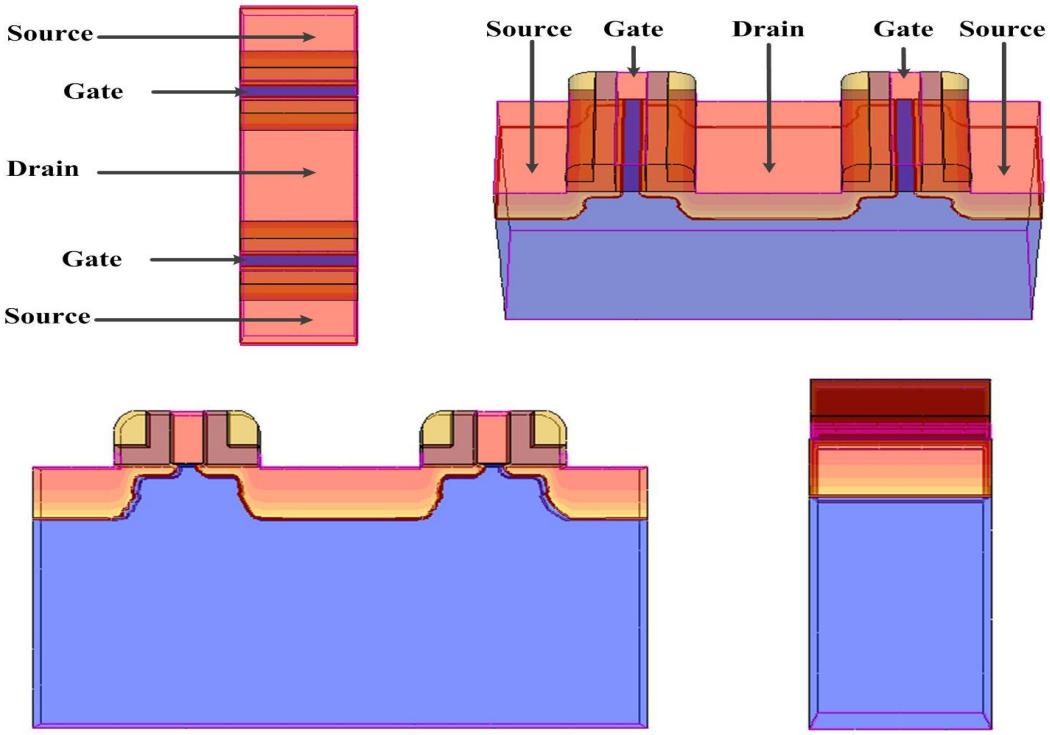


Figure 3.23: Multi-finger motif (parallel gates) is a widely used the common unit in circuit design. As previously described, this motif provides a way to investigate the impact of LER-induced variation on two close gates. Top view (upper left), Front view (low left), Side view (lower right), 3D view (upper right).

In order to reveal the actual shape of the resulting statistical distribution and to understand the physical reasons behind LER-induced variations, comprehensive 3D LER modelling method based on the use of 1D Fourier synthesis technique and power spectrum is carried out in this experiment [73]. The flow chart of modelling LER method is shown in Figure 3.28. LER is described by the power spectral density (PSD) that is the Fourier transform of the autocorrelation function [73]-[75]. Two important parameters, the standard deviation (RMS amplitude) Δ and the correlation length Λ, are applied to characterize the autocorrelation function. The standard deviation of the transversal magnitude to the gate edge is represented by Δ. The fitting of a particular type of autocorrelation function to the gate edge is represented by the correlation length, Λ. $S_G(k)$ and $S_E(k)$ are the power spectra for Gaussian and exponential autocorrelation functions, respectively [73].

$$S_G(k) = \sqrt{\pi}\Delta^2 e^{-(\frac{k^2\Lambda^2}{4})} \tag{3.3}$$

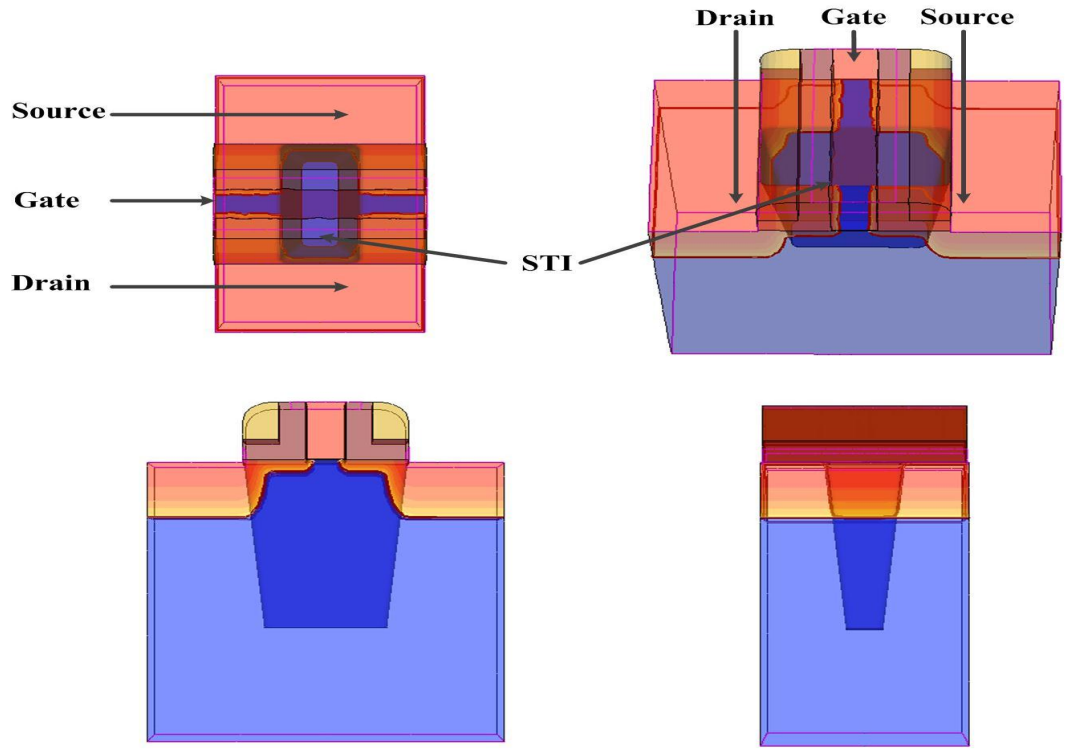


Figure 3.24: *O* shaped motif is a complex and irregular layout placement that is inspired from the channel segment. Top view (upper left), Front view (low left), Side view (lower right), 3D view (upper right).

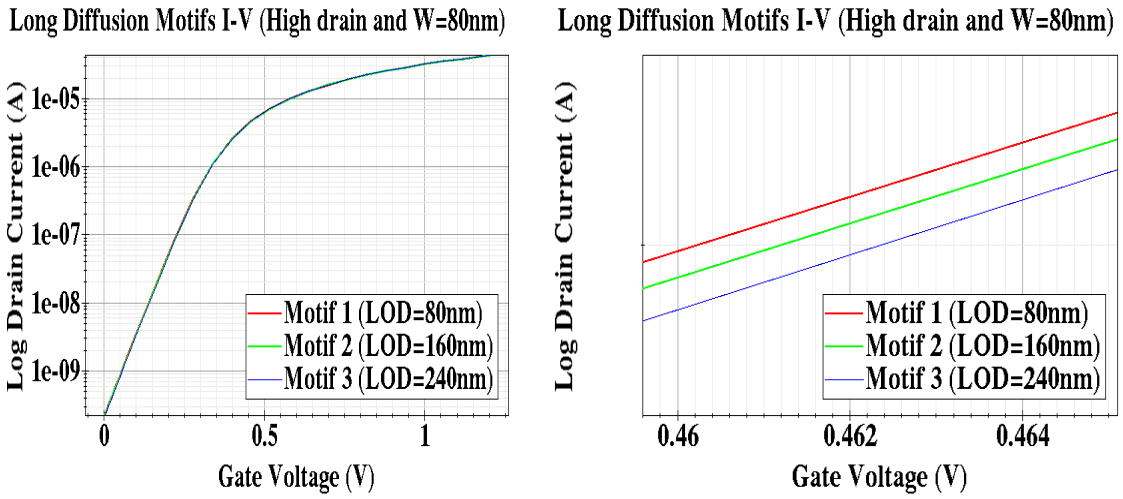
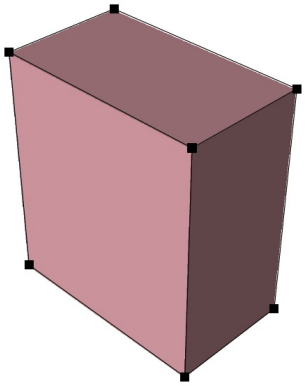


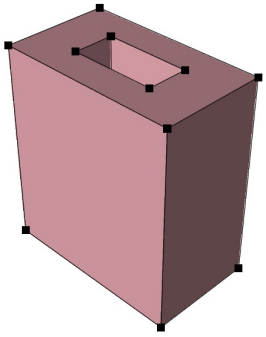
Figure 3.25: The comparison of the long diffusion motif with different LOD. As the S/D diffusion area increasing, the drain current deceases due to the S/D resistance value changed.

$$S_E(k) = \frac{2\Delta^2\Lambda}{1 + k^2\Lambda^2} \tag{3.4}$$

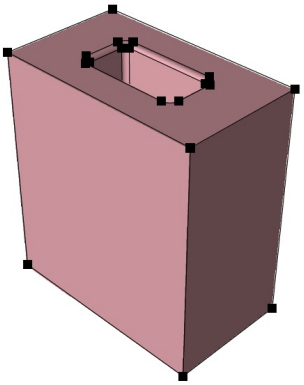
$$k = i(2\pi/Ndx) \tag{3.5}$$



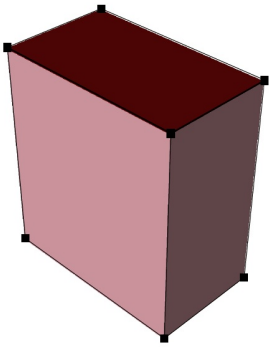
(a) First step is substrate definition



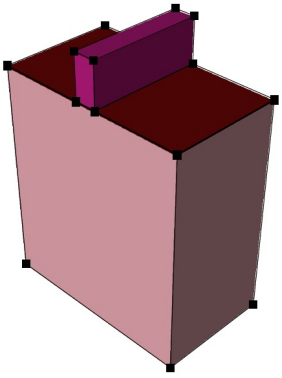
(b) Active area definition and central trench etching



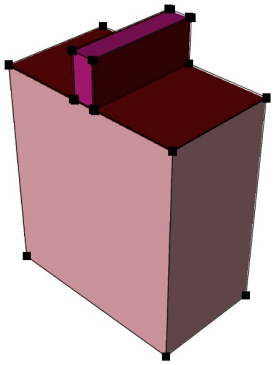
(c) Edges of central trench surrounding



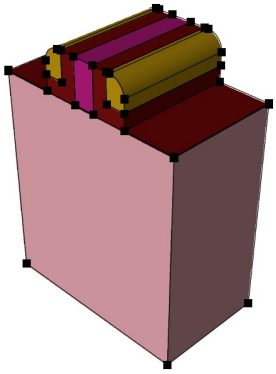
(d) Central STI formation and gate oxide formation



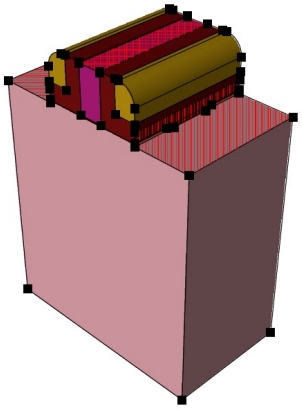
(e) Polysilicon gate formation



(f) Oxide extension spacer formation



(g) Nitride sidewall space formation



(h) Contacts definition

Figure 3.26: *O* shaped Motif 3D structure emulation processing without LER

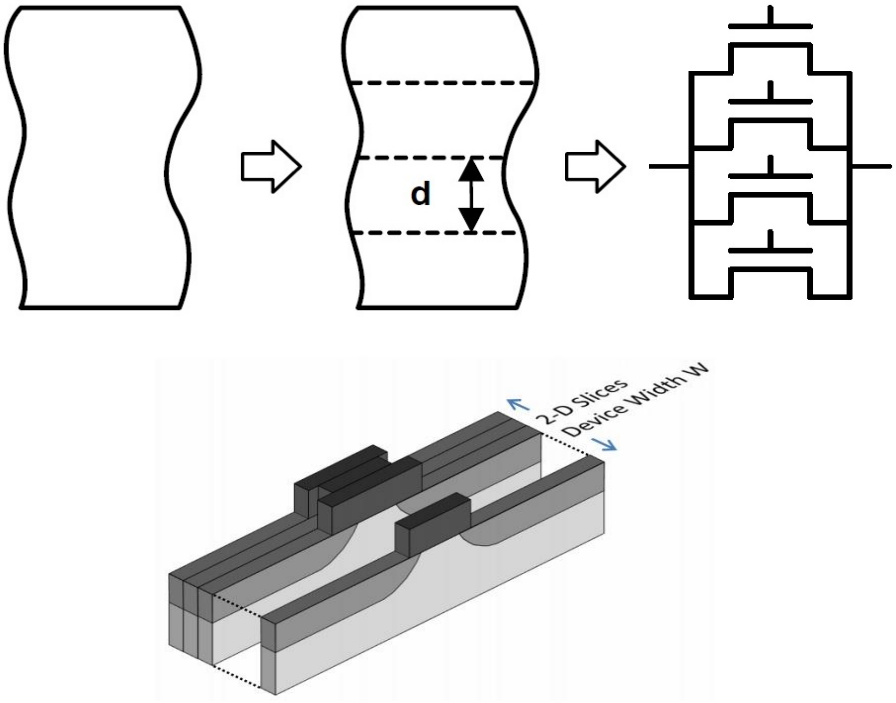


Figure 3.27: The flow to divide a gate roughness into a set of narrow slices, where each slice is assumed to be ideal with no roughness [27]-[72]. Each slice has a unique gate length due to LER and the same width.

Where dx is the discrete spacing used for the gate edge and $0 < i < N/2$. These N discrete elements are used to represent the gate line $L(k)$ in the Fourier space, with amplitude generated from the above power spectra equation and the phase of the elements being selected randomly, making each line unique. In order to obtain the corresponding real space line $H(x)$, $L(k)$ is dealt with the inverse Fourier transform, shown in Figure 3.28. Comparing to the gate edges created by the exponential autocorrelation function, lines generated from the Gaussian autocorrelation function are smoother due to a lack of high frequency components. More details of this method are described in [74]-[75]. The correlation length is set to $\Delta =$ 12 nm and the RMS is $\Lambda = 2$ nm, as this is typical for *50 nm* technology [45]. Figure 3.29 gives an example of random line generated by this method.

3.3.2 TCAD Statistical Simulation Approach

An ensemble of motifs (300 samples) was simulated using Sentaurus TCAD tool in order to obtain motif structures variability information. In order to obtain a good estimate for 1σ,

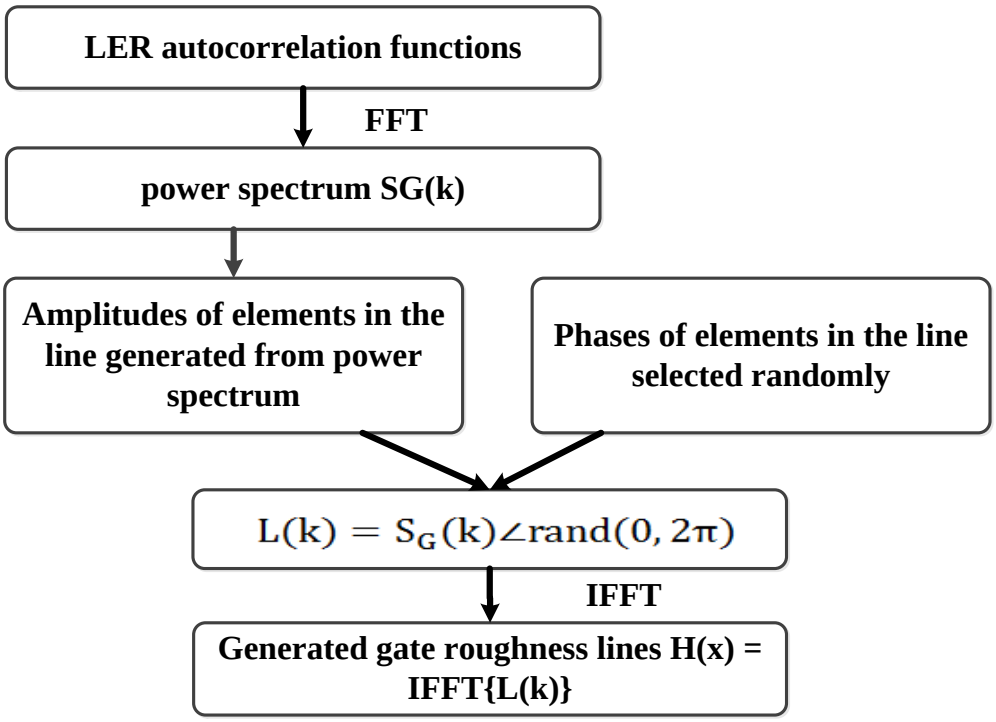


Figure 3.28: The flow chart of LER generation.

at least 300 samples are required. Figure 3.30 illustrates the flow of the TCAD statistical simulation method. The flow is divided into three main parts. The LER-induced variation information is first generated from the previously introduced method in Section 3.3.1. This variation information is then passed into TCAD structure editor to generate the gate pattern mask. The second part is 3D motif structure emulation and simulation. Massive motif structure ensembles with unique patterns of LER are created and simulated. Finally, these motif samples simulation provide statistical simulation result so that it is possible to extract and assess the impact of LER induced variability on each motif structure. As a generic methodology for investigating the effect of variability on device layout structure, this procedure is obviously generic and valid for any technology and device such as FinFET and UBT SOI device.

3.4 Experiment Results and Discussion

Experiments are processed on a 20 cores, 40G RAM and Linux OS server. The edge roughness is generated by our LER modelling tool, written in Matlab. The device structure with the

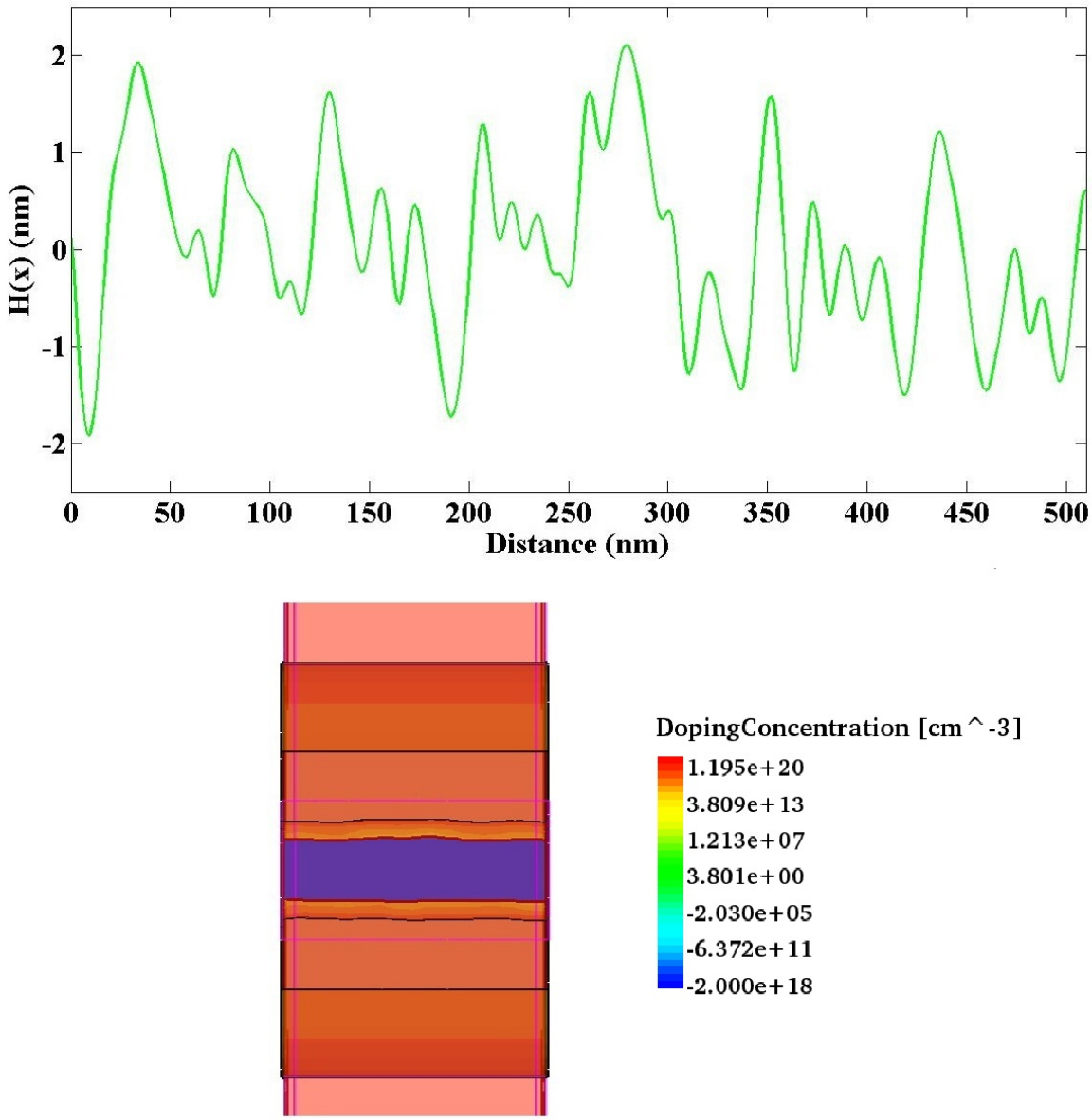


Figure 3.29: The roughness of gate edge is generated using PSD method. The top subfigure shows the mathematical calculation gate edge line. And the application of the mathematical calculated line edge result in TCAD device structure is depicted in the bottom subfigure.

edge roughness emulation and simulation performed by Sentaurus TCAD software described in Section 3.1.2. A screen capture of the TCAD simulation process showing simulation information is shown in Figure 3.31.

3.4.1 Motif Verification

For guaranteeing and providing a reliable metric to verify motifs simulation result, a reference device with regular layout (seen in Figure 3.32 and the corresponding device geometry param-

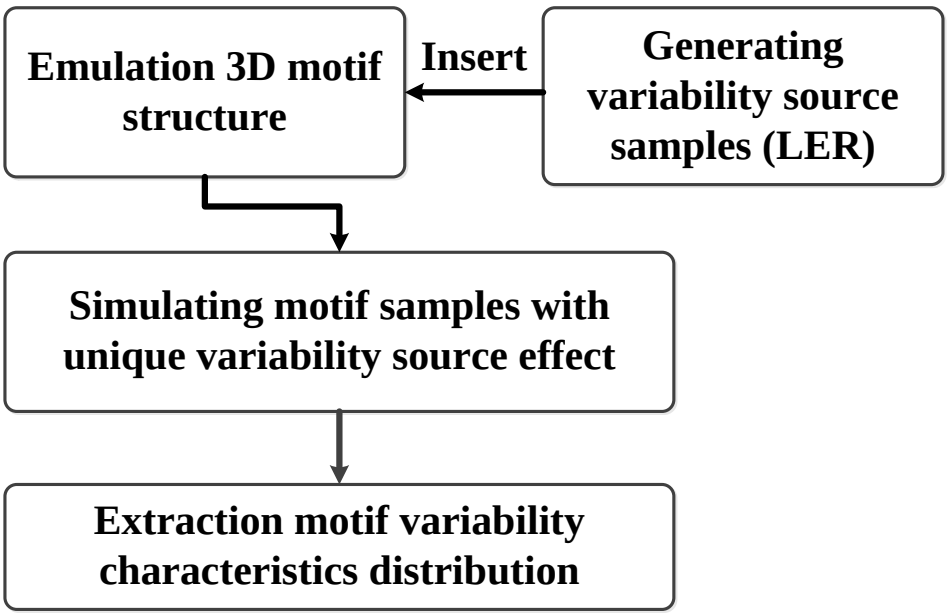


Figure 3.30: The flow chart of motifs TCAD statistical simulation.

eters is listed in Table 3.4) is emulated and simulated. The reference device (regular/standard device) with two different geometry sizes (W=80nm and W=160nm) were tested, gate voltage vs. drain current results are illustrated in Figure 3.33. Equation 3.6 is current calculation formulation.

Table 3.4: TCAD Device Geometry Parameters

Parameter	Description	Value
L_g $[nm]$	Length	50
W $[nm]$	Width	80/160
$ActiveArea$ $[\mu m^2]$	Active area (W=80nm/160nm)	0.036/0.072
L_{sp} $[nm]$	Width of the device spacer	80
L_{sd} $[nm]$	Width of the device Source/Drain	200
H_s $[nm]$	Substrate thickness	500
T_{ox} $[nm]$	Thickness of the gate oxide	2

$$I_D = \frac{\mu_0 C_{ox} W}{L}[(v_{gs} - V_T - (\frac{v_{ds}}{2})]v_{ds} \tag{3.6}$$

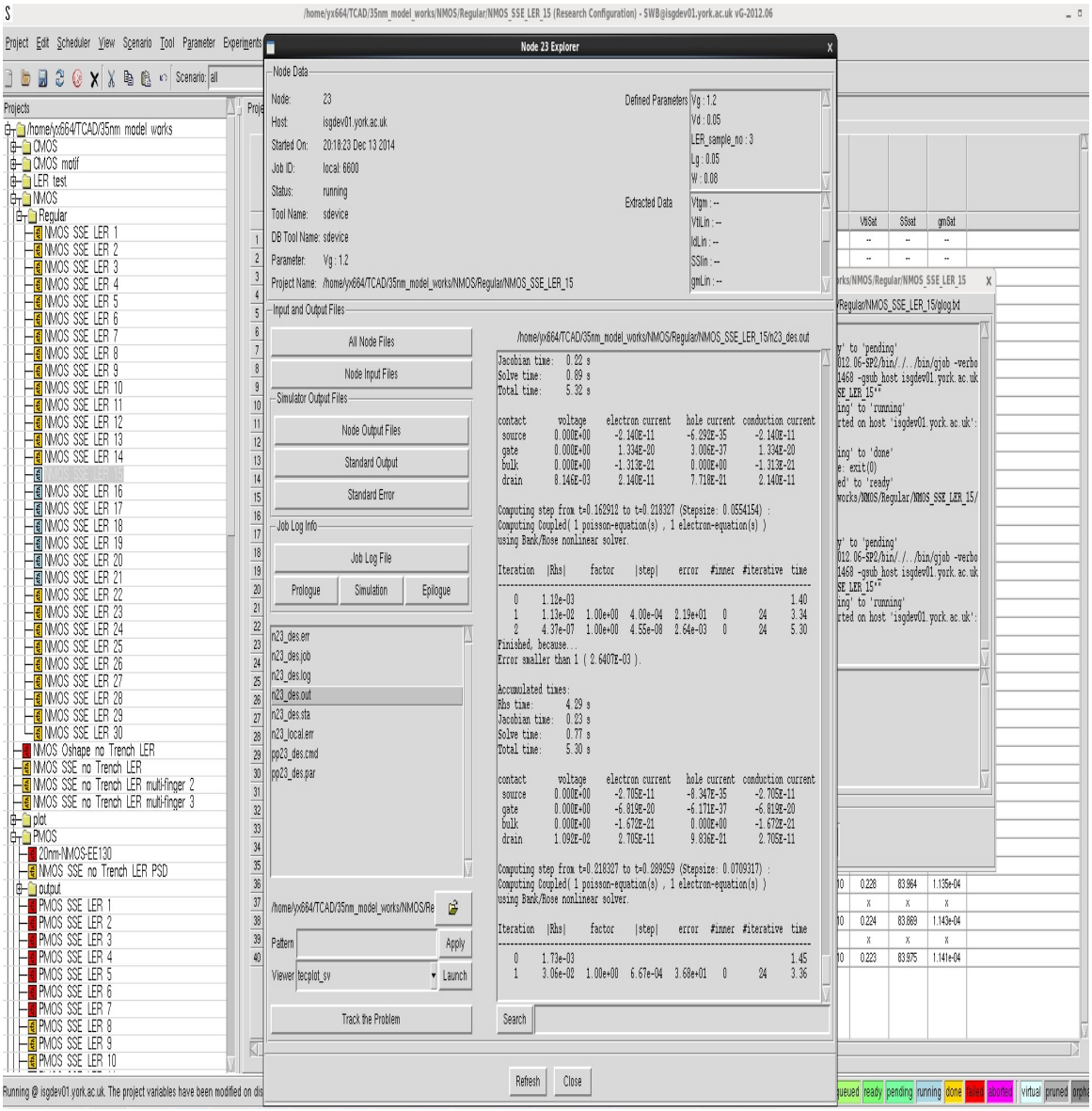


Figure 3.31: TCAD Simulation tool GUI and screen capture of simulation information.

Where C_{ox} is the gate oxide capacitance per unit area, μ_0 is the mobility of charge carries. W/L is the aspect ratio. The current value becomes doubled while device width is doubled and the other parameters keep constant. According to simulation result, the current value of device with W=160nm is nearly double of the drain current of device with W=80nm. Reference device simulations are in agreement within Equation 3.6 the theoretical prediction. Consequently, this reference device is considered reliable and suitable for further experiments. Once the correctness of this reference device characteristic is proved, all motifs can be verified through comparing with the reference device.

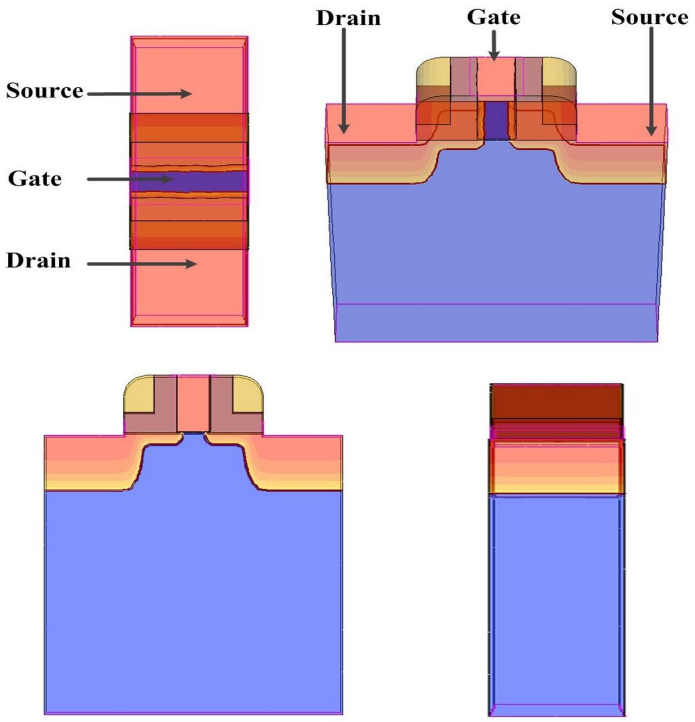


Figure 3.32: Reference device with standard transistor layout is constructed by TCAD tools, illustrated by different views. Top view (upper left), Front view (low left), Side view (lower right), 3D view (upper right).

The comparison of *I-V* characteristic between *O* shaped motif, long diffusion, multi-finger device and reference device (regular/standard device, W=160nm) is depicted in Figure 3.34. The current values of these motifs are approximately the same. This means that these motifs and reference device have the similar *I-V* characteristics. Simultaneously, the slightly difference of their current values also prove that the device performance is indeed affected by layout factors.

3.4.2 Statistical Motif Simulation Results

As previously stated, the motif technique is a generic methodology for investigating the impact of different layout structure on device performance and variability. In this section, the aim is to verify whether the motif technique is reasonable and reliable for exploring the link between motif and device performance and variability. Thus, only *O* shaped motif, multi-finger motif and reference device are used to perform statistical simulation to verify this methodology. Gate folding configuration (multi-finger motif) is a common layout in digital and analogue design where the large size transistor is split into several smaller transistors

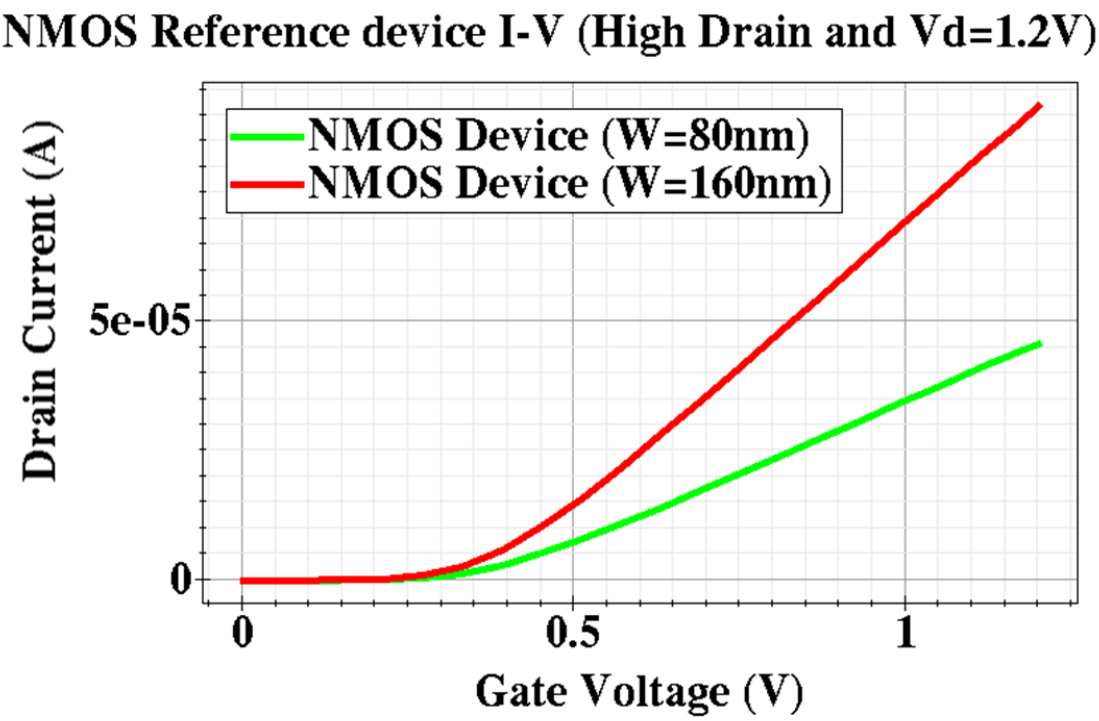


Figure 3.33: Reference device I-V characteristic with different width. The device with w=160nm is nearly double of the drain current of device with w=80nm. These simulations are agreement within the device characteristic theory analysis.

with parallel connections. This motif is very useful for random variability study. The results of O shaped motif, multi-finger motif and reference device simulation are illustrated in Figures 3.35, 3.36 and 3.37 showing drain current *vs.* gate voltage ($I_d - V_g$) electrical transfer characteristics of 300 microscopically motifs with unique LER different devices, respectively. These devices have been simulated at a high drain voltage of 1.2 V (-1.2 V) and low drain voltage of 0.05 V (-0.05 V) for n-channel (p-channel) motif. The distribution of $I_d - V_g$ traces shows the noticeable impact of the sources of statistical variability. Two important device parameters, threshold voltage and leakage current, are extracted from the $I_d - V_g$ characteristics to evaluate the impact of statistical variability on motif structure. The standard deviation and average value of V_{th} and I_{off} for motifs are summarized in Table 3.5.

Figures 3.38 and 3.39 show the histogram of distributions of threshold voltage and off-current for O shaped motif and multi-finger motif, respectively. In order to intuitively observe the distribution of the extracted parameters, the comparison of motifs and reference device are illustrated by *box*-and-*whisker* plots, shown in Figures 3.40 and 3.41. The threshold voltage standard deviation for n-channel multi-finger motif is 5.03 mV and the mean value is 385.9

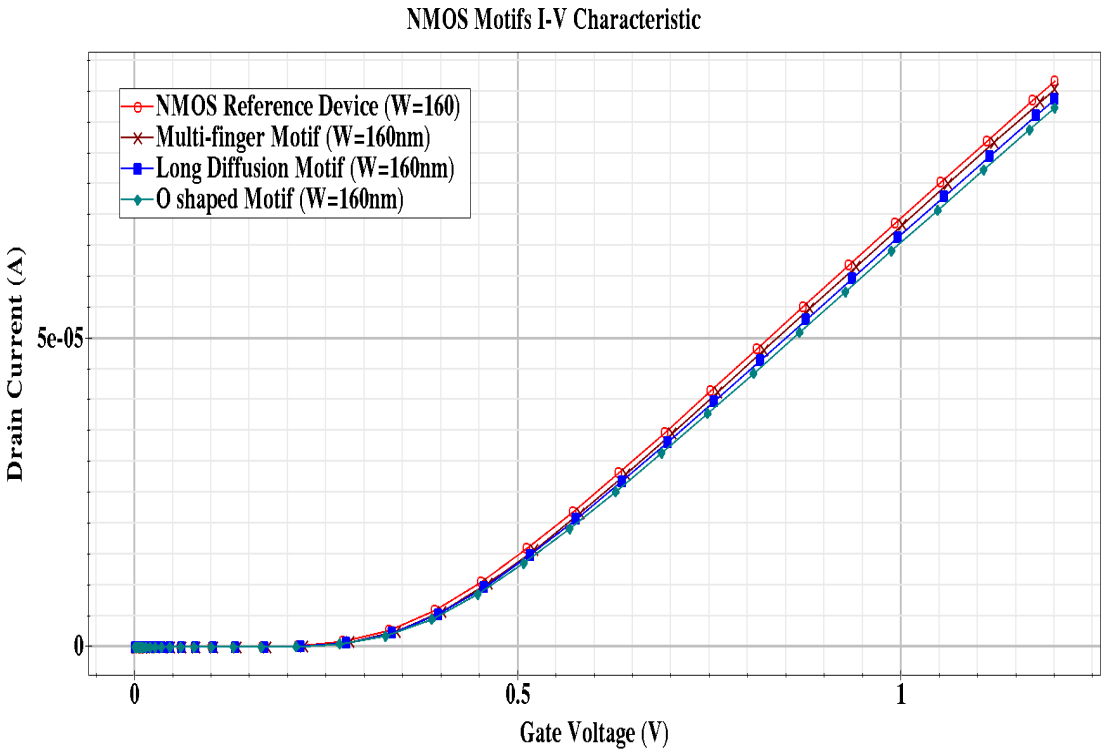


Figure 3.34: The comparison of different motifs' I-V characteristic. Although these motifs I-V characteristic are approximately the same, the various layouts result in these motifs existing the slightly difference on their performance.

Table 3.5: Motifs extraction parameters at high drain voltage

Motifs	Parameter	Value (NMOS)	Value (PMOS)
Reference device	$StdV_{th}$	3.25 mV	2.56 mV
O shaped motif		3.26 mV	2.49 mV
Multi-finger motif		5.03 mV	2.57 mV
Reference device	$MeanV_{th}$	385.3 mV	-445.9 mV
O shaped motif		388.3 mV	-452.1 mV
Multi-finger motif		385.9 mV	-446.6 mV
Reference device	$StdI_{off}$	8.75e-11 A	1.47e-11 A
O shaped motif		9.23e-11 A	1.43e-11 A
Multi-finger motif		1.38e-10 A	1.45e-11 A
Reference device	$MeanI_{off}$	8.02e-10 A	-1.97e-10 A
O shaped motif		7.95e-10 A	-1.88e-10 A
Multi-finger motif		8.12e-10 A	-1.94e-10 A

mV. For n-channel reference device, it has the threshold voltage standard deviation of 3.25 mV and average of 385.3 mV. Although the average threshold voltage of multi-finger motif is nearly equal to the reference device, the threshold voltage standard deviation is larger than the reference device due to poly pitch affecting on variation and small size sub-transistor associated with more variation than large transistor. Comparing the threshold voltage of n-channel O shaped motif and the reference device, the O shaped motif has a slightly higher average threshold voltage (388.3 mV) than the reference device (385.3 mV). It is noticeable that the higher threshold voltage value is beneficial for the O shaped motif performance than the reference device. The reason for this is difficult to explain based on simple analytical assumptions but might be related to the complex centre SiO_2 STI structure affects on channel potential, mobility and other intrinsic parameter, which changes device performance. According to these results analysis, a possible way achieving circuit optimisation is inspired from motifs simulation result that selects different motifs to construct circuit to meet specific design requirements. Additionally, Figure 3.42 shows $V_{th} - I_{off}$ scatter plots and histogram of distributions of V_{th}/I_{off} for reference device with different width. Simulation result illustrate that the variability is well suppressed with device geometry size increasing (supported by Figure 3.43), which conclusion is in agreement with the literature [76]-[80].

3.5 Summary

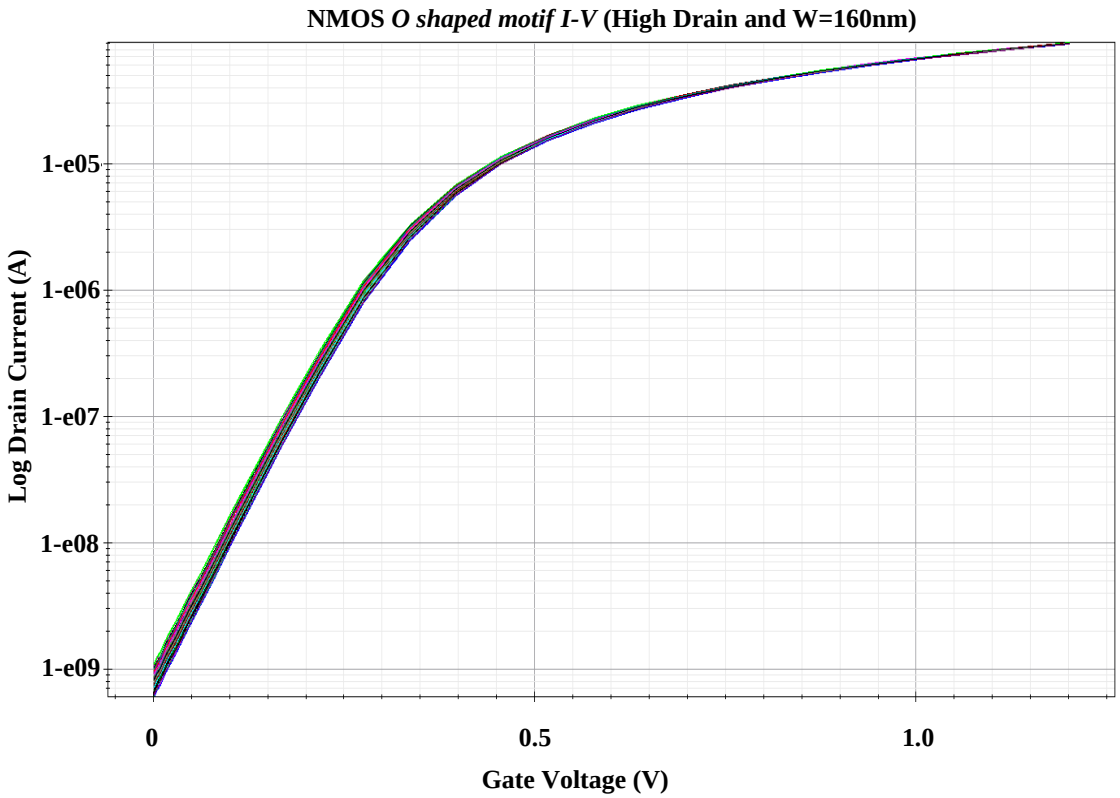
In this chapter, the concept of motif has been described and studied in detail, which motif is proposed for improving the device/circuit layout regularity. Motif is also a promising and systematic way to investigate impact of variability on the device layout at atomic levels. In order to investigate the effects of intrinsic parameters (threshold voltage and off-current) fluctuation on motifs performance, the motif structure with LER-induced variation has been modelled. A statistical motif TCAD simulation methodology has been developed for calculating motif's intrinsic parameters fluctuation and evaluating their performance that is also shown to be equally applicable to other device structures.

The distribution of V_{th}/I_{off} has been accurately characterized at low and high drain voltages with ensembles of 300 samples. Simulation results indicate that the device performance and intrinsic parameter variability due to LER is actually influenced by motif structures. For example, PMOS O-shaped motif has smaller V_{th} standard deviation and lower I_{off} than reference device based on Table 3.5 shown results. This gives an inspiration that the use of different motif constructing circuit achieves circuit optimisation that is discussed in Chapter 6. Ta-

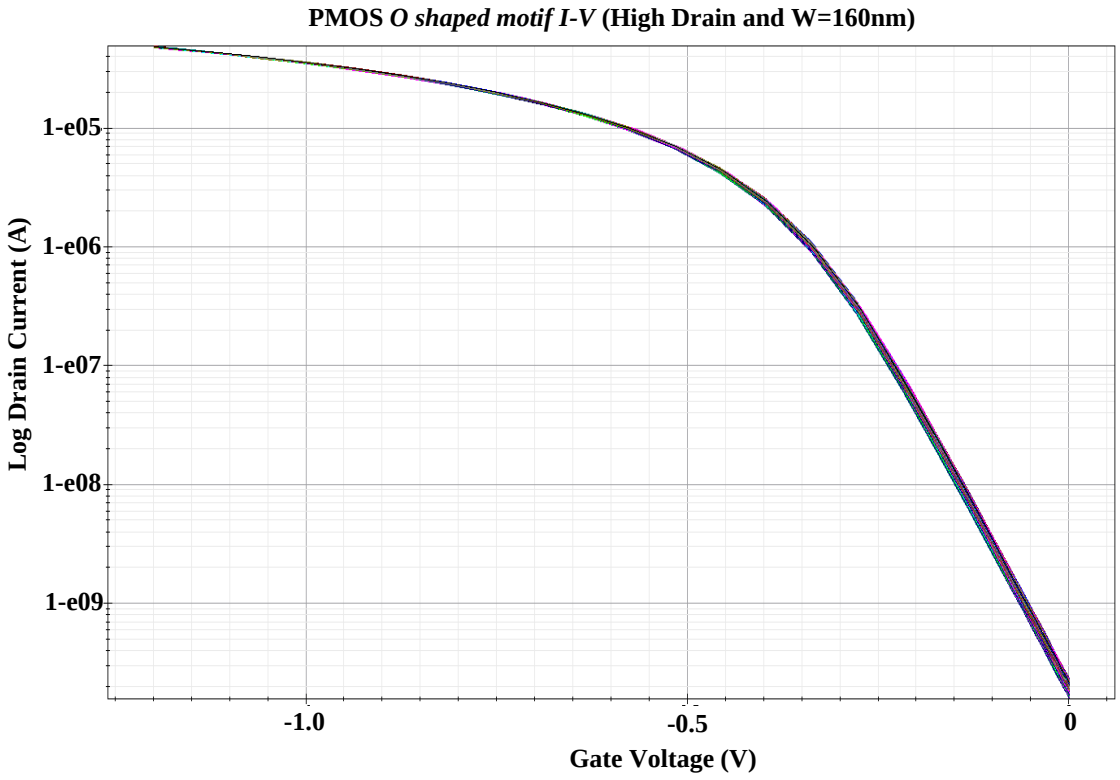
ble 3.6 summaries these motifs' advantages and disadvantage. Additionally, statistical motif simulation results in this chapter are essential for BSIM extraction (described in Chapter 5) that builds the bridge between device simulation and circuit simulation so that circuit can be constructed using motifs. Chapter 4 covers the basics of evolutionary algorithms (EAs) knowledge to prepare the reader with the necessary background on EA operation to follow the discussion in the Chapters 5 and 6.

Table 3.6: Summary of motifs Characteristics

Motifs	Advantage	Disadvantage
O shaped motif	lower PMOS $StdV_{th}$ and lower PMOS I_{off} than PMOS reference device	larger layout area, larger NMOS I_{off}
Multi-finger motif	reduce parasitic S/D capacitance, more compact layout area	larger $StdV_{th}$ (NMOS) than reference device and PMOS
Long diffusion device	increase diffusion area to decrease S/D resistance	larger layout area

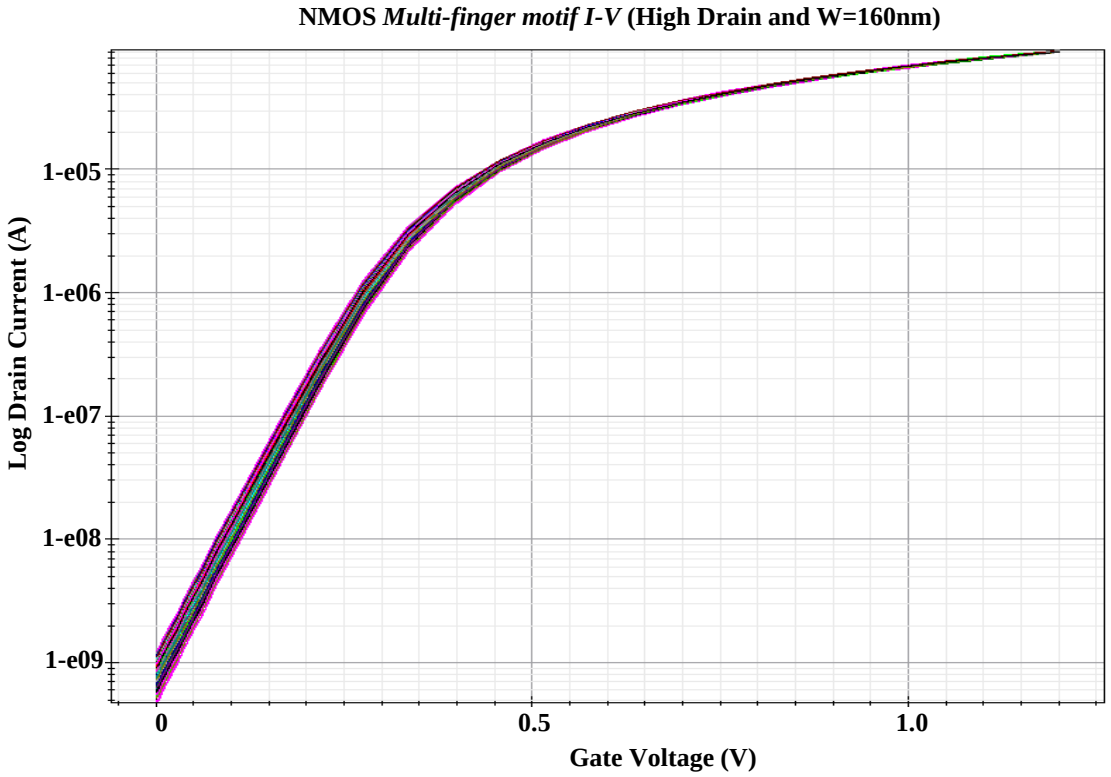


(a)

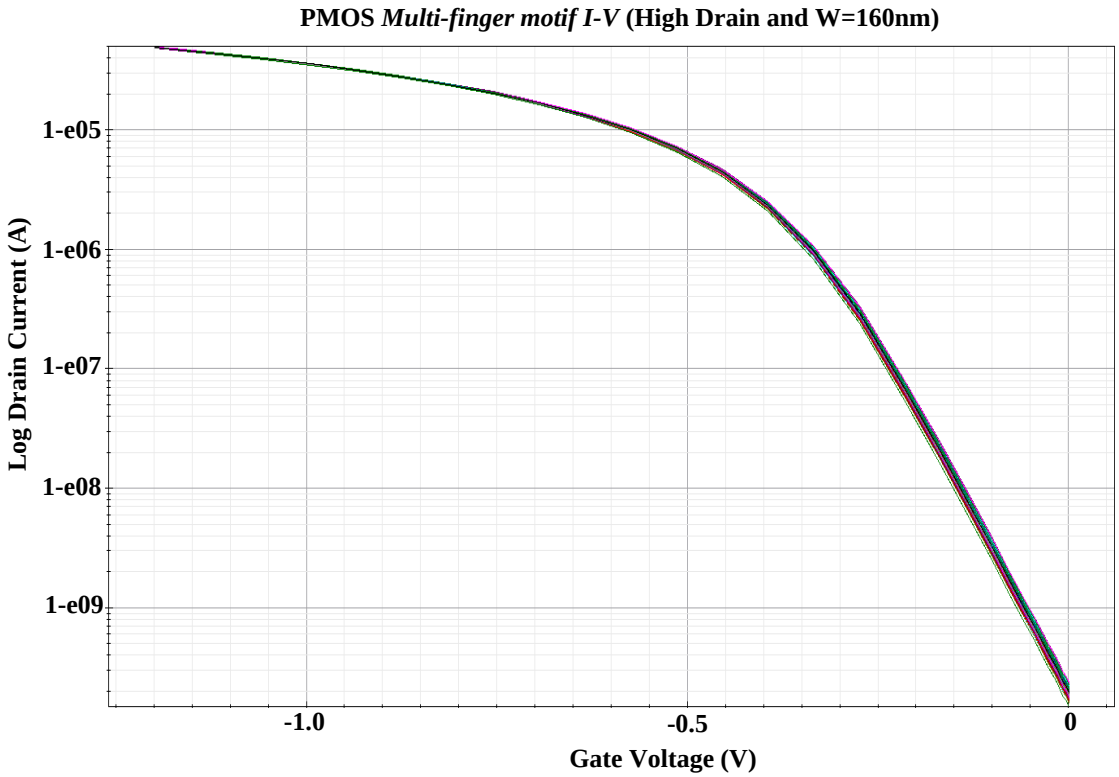


(b)

Figure 3.35: The complete ensemble of 300 gate transfer characteristics in saturation and linear regimes for O shaped motif with $50\ nm$ gate length under high drain voltage condition, subjected purely to LER are illustrated in n-channel motif (a) and p-channel motif (b), respectively. LER has a relatively impact on on-current variation and the subthreshold region where the variation in the leakage current increases. .

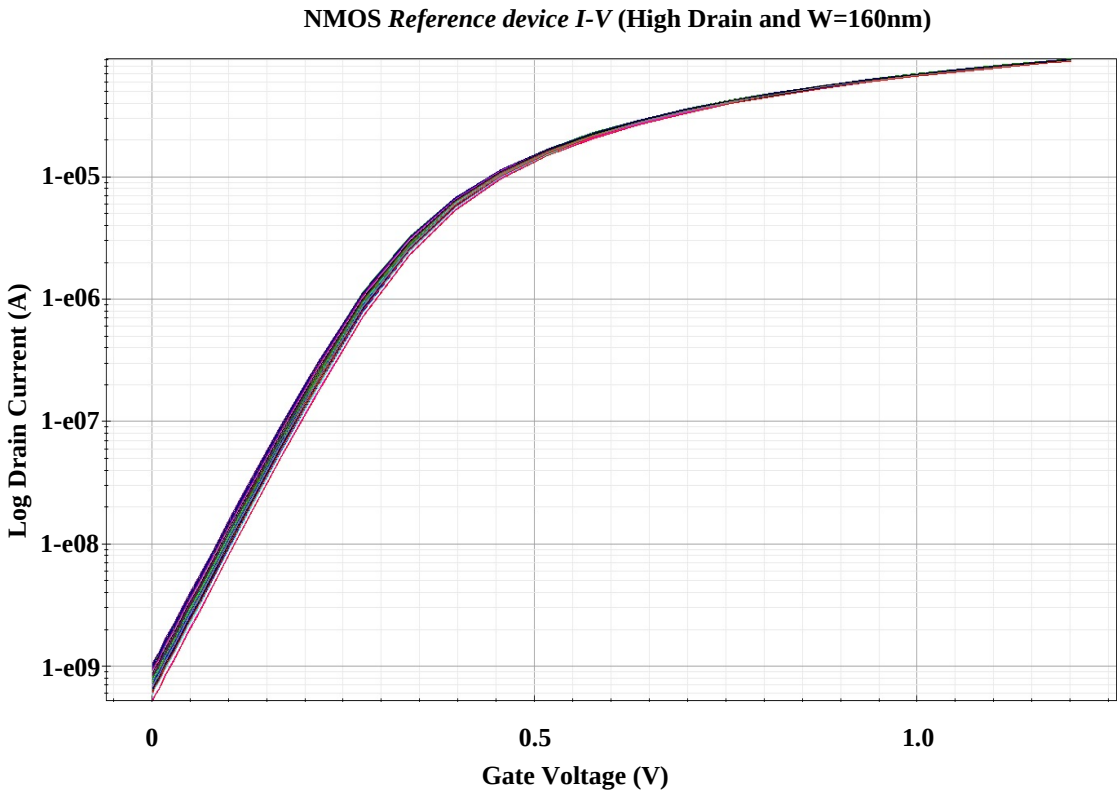


(a)

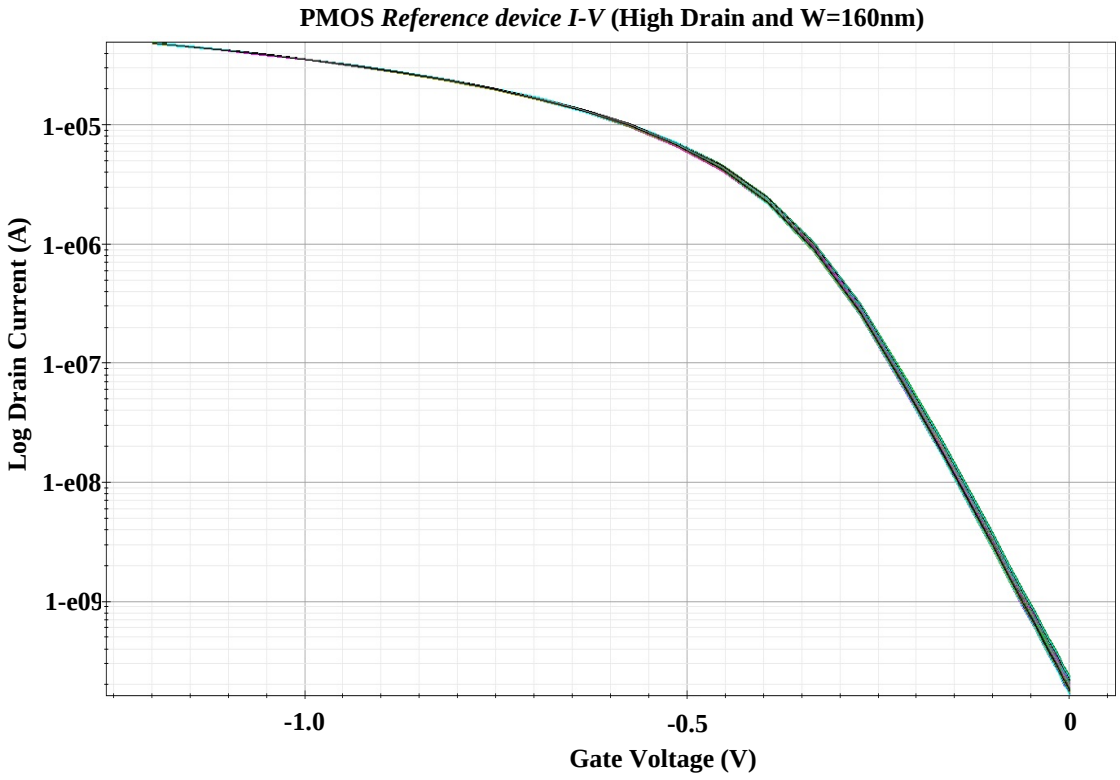


(b)

Figure 3.36: Large ensemble of 300 $I_d - V_g$ characteristics in saturation and linear regimes for multi-finger motif with *50 nm* gate length under high drain voltage condition, subjected purely to LER are illustrated in n-channel motif (a) and p-channel motif (b), respectively.

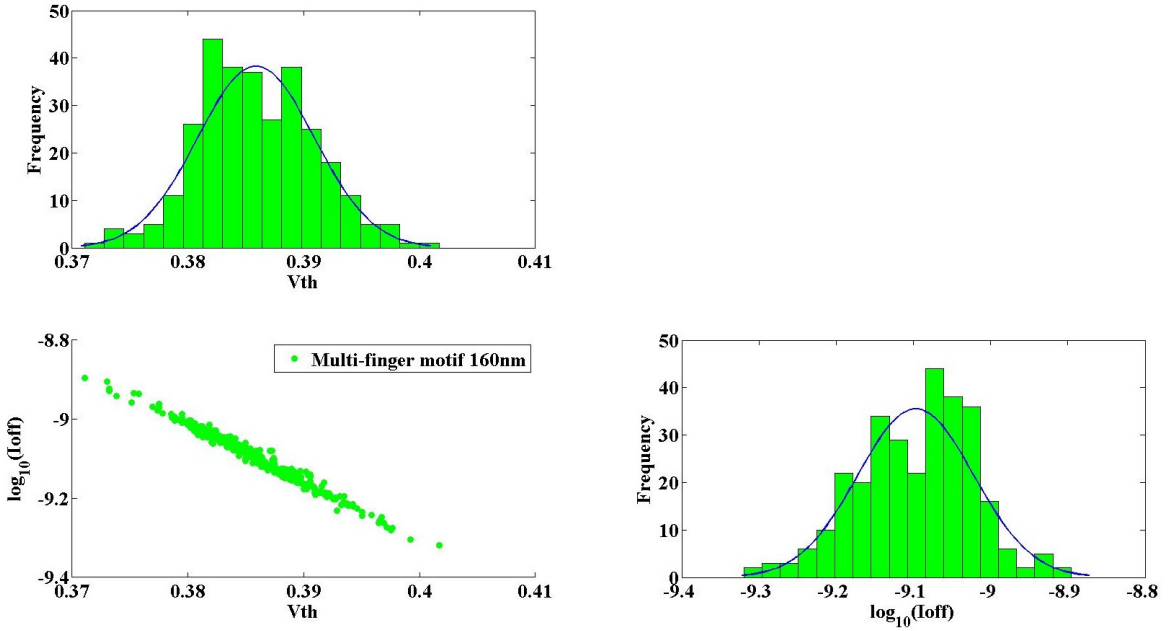


(a)

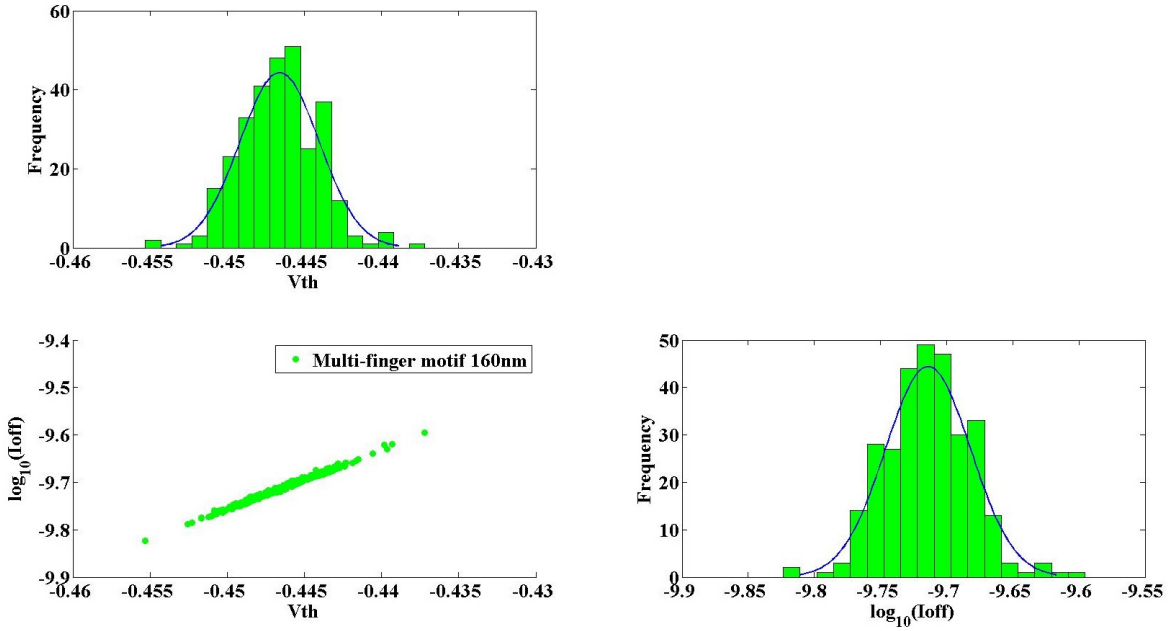


(b)

Figure 3.37: Large ensemble of 300 $I_d - V_g$ characteristics in saturation and linear regimes for reference device with *50 nm* gate length under high drain voltage condition, subjected purely to LER are illustrated in n-channel motif (a) and p-channel motif (b), respectively.



(a)



(b)

Figure 3.38: The scatter plot and histogram of distributions of V_{th} and I_{off} for multi-finger motif, (a) n-channel motif. (b) p-channel motif. The scatter cloud contains 300 points obtained from TCAD statistical simulation of multi-finger motif, which highlights the affect of LER-induced variability. The density plots (above and to the right of the scatter plot) show the distribution for the V_{th} and I_{off}.

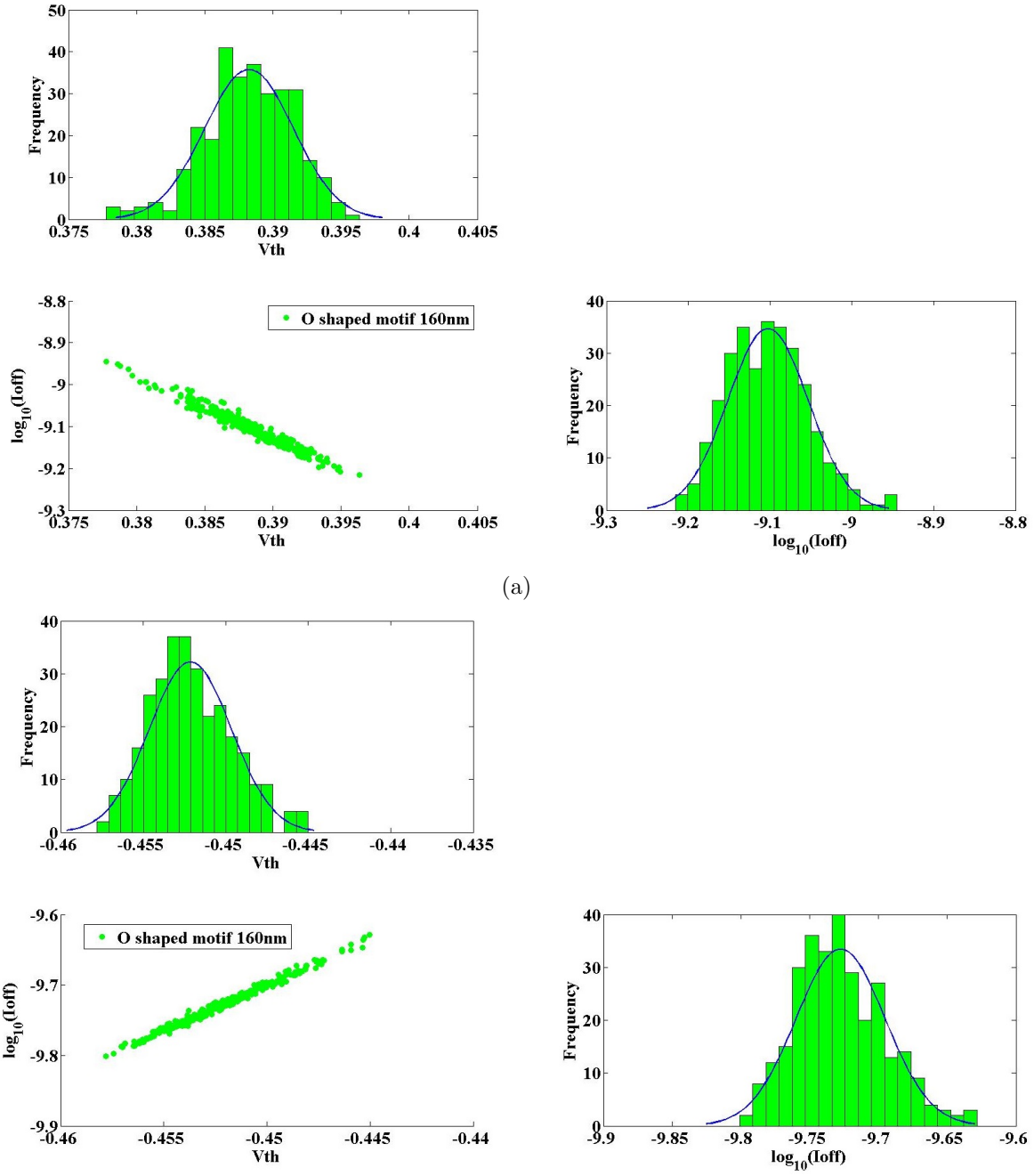
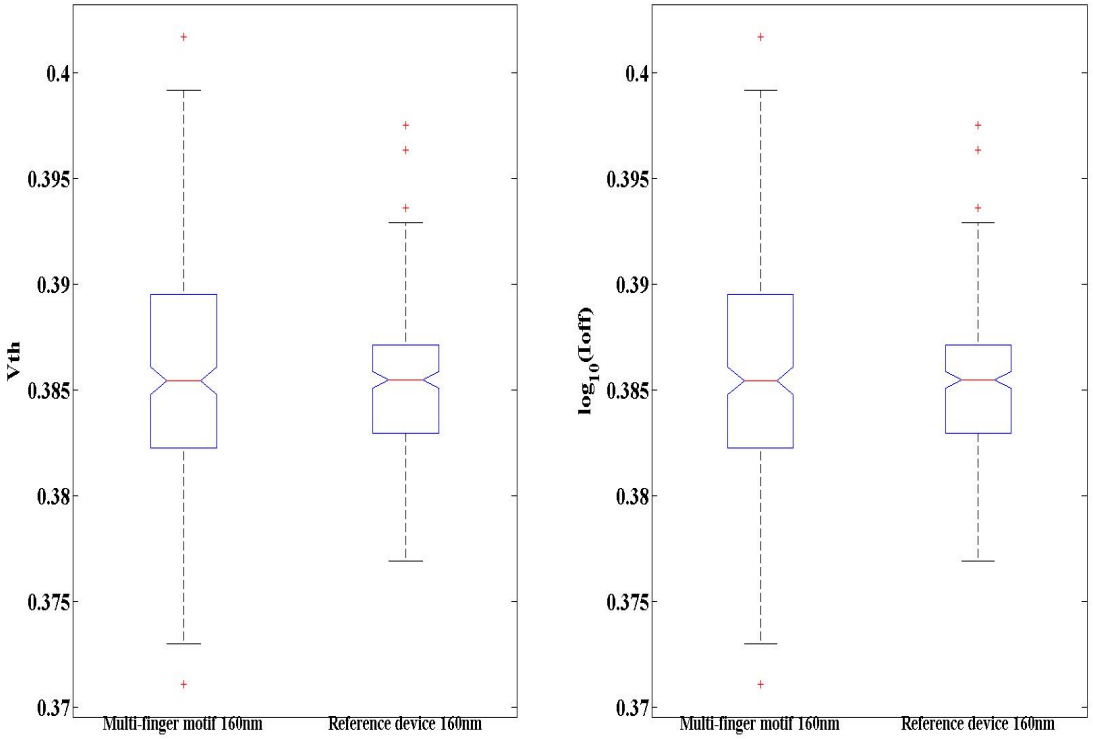
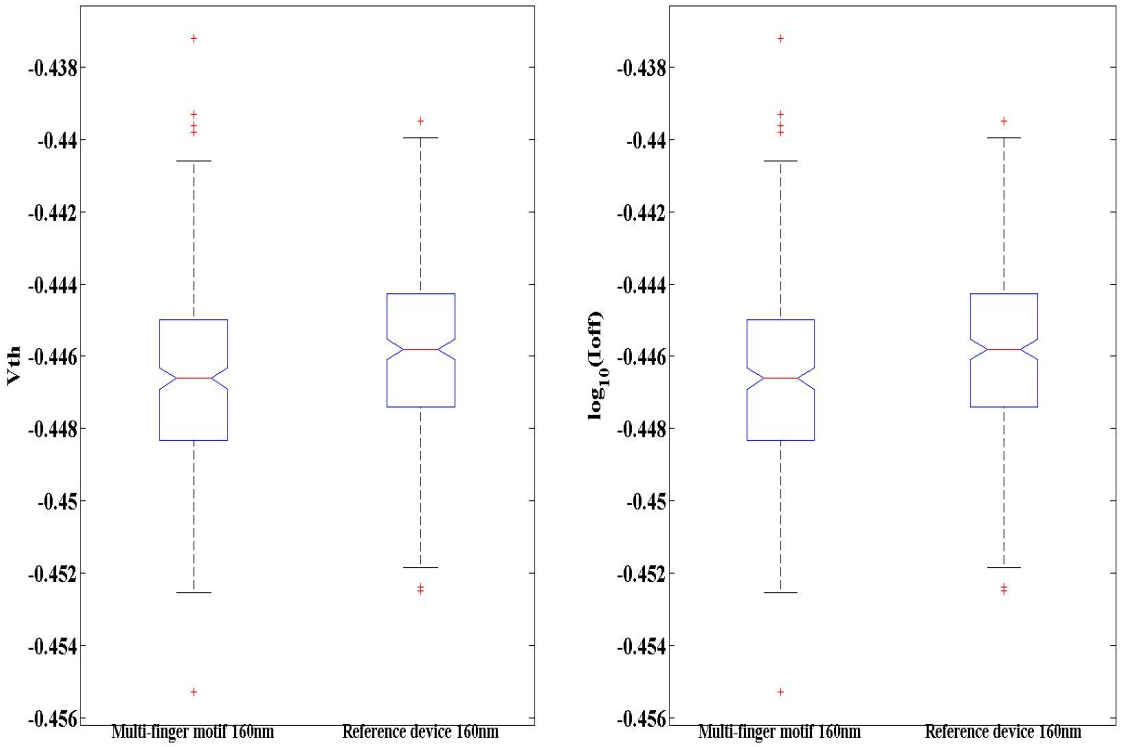


Figure 3.39: The scatter plot and histogram of distributions of V_{th} and I_{off} for O shaped motif, (a) n-channel motif. (b) p-channel motif. The scatter cloud contains 300 points obtained from TCAD statistical simulation of O shaped f motif, which highlights the affect of LER-induced variability. The density plots (above and to the right of the scatter plot) show the distribution for the V_{th} and I_{off}.

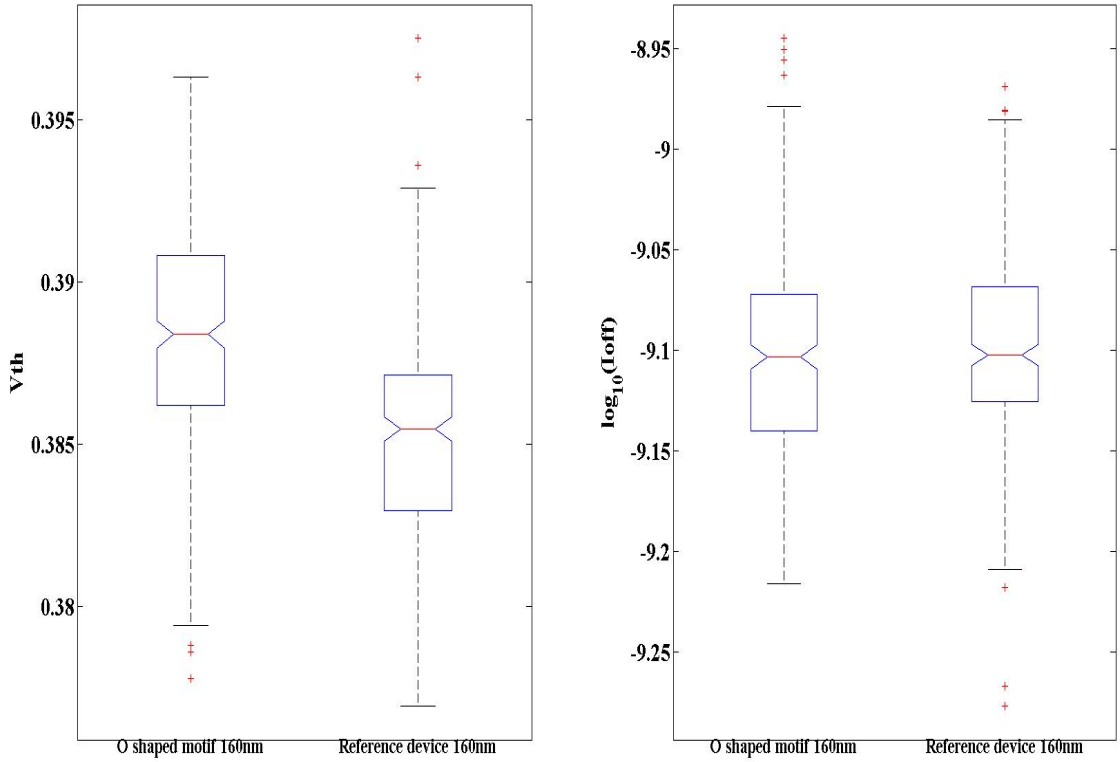


(a)

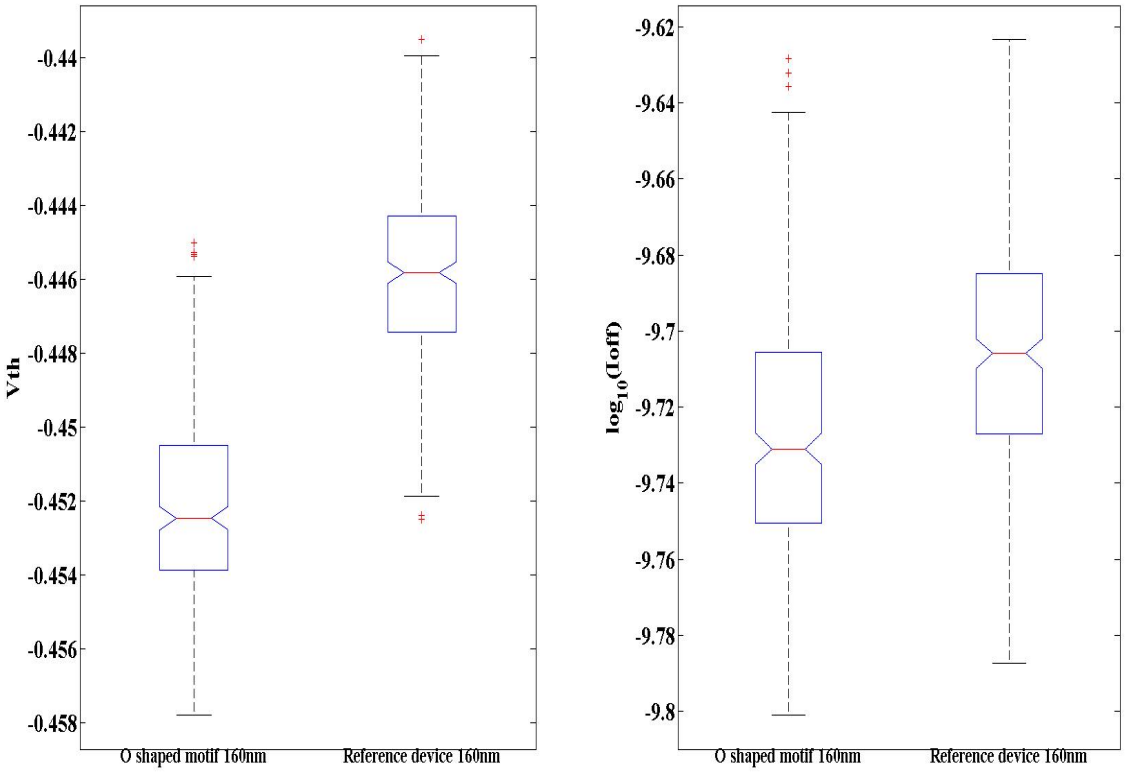


(b)

Figure 3.40: Comparison of multi-finger motif and reference device at high drain voltage for V_{th} and I_{off}, (a) n-channel motif. (b) p-channel motif. The line in the centre of the box denotes the median, the *box* denotes the inter-quartile range (IQR).

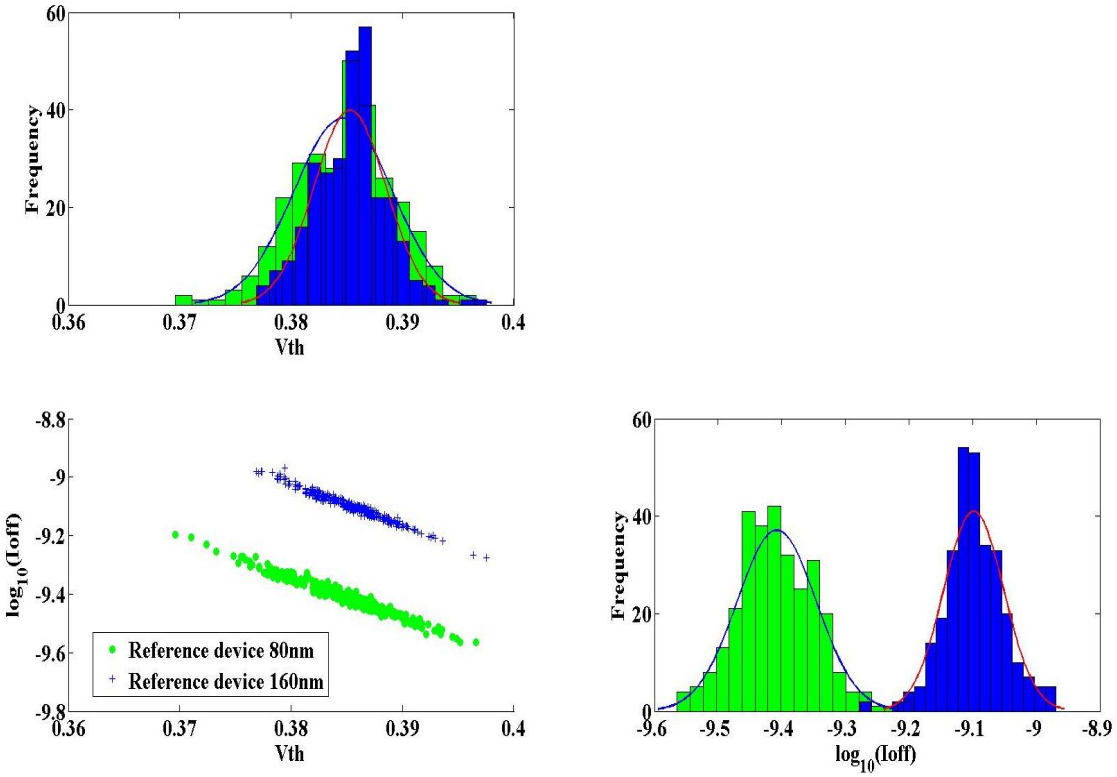


(a)

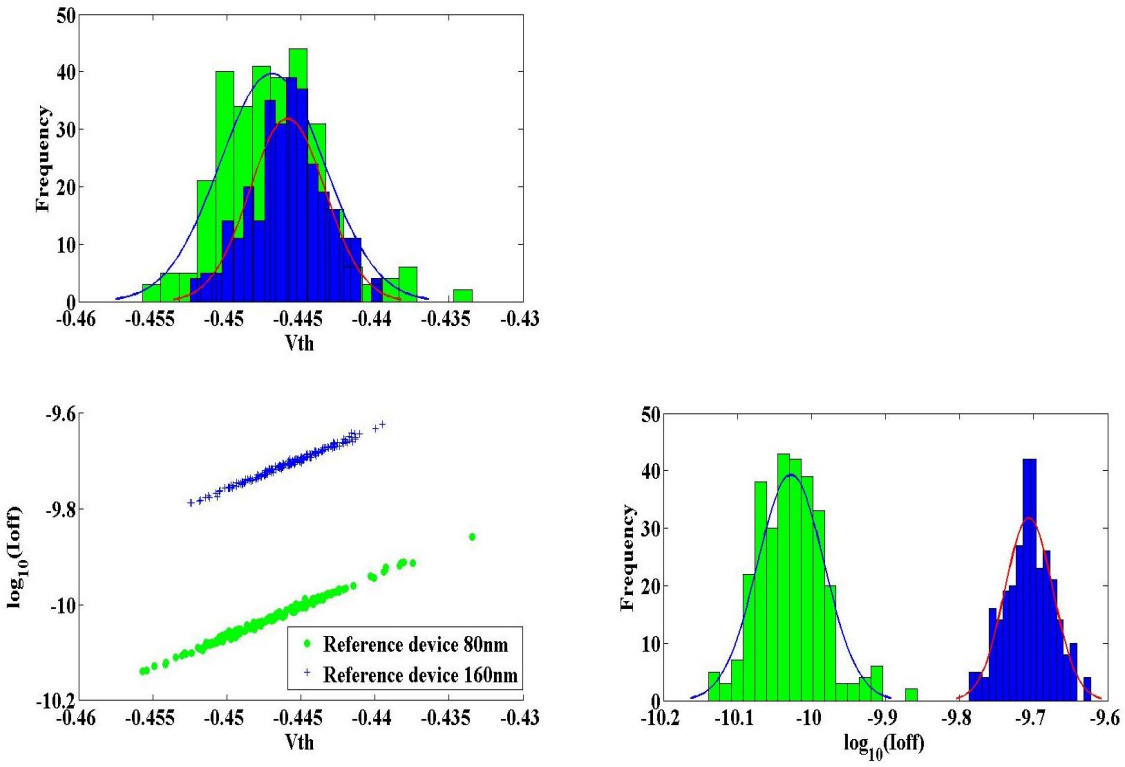


(b)

Figure 3.41: Comparison of O shaped motif and reference device at high drain voltage for V_{th} and I_{off} represented by *Box-and-whisker* plots, (a) n-channel motif. (b) p-channel motif.

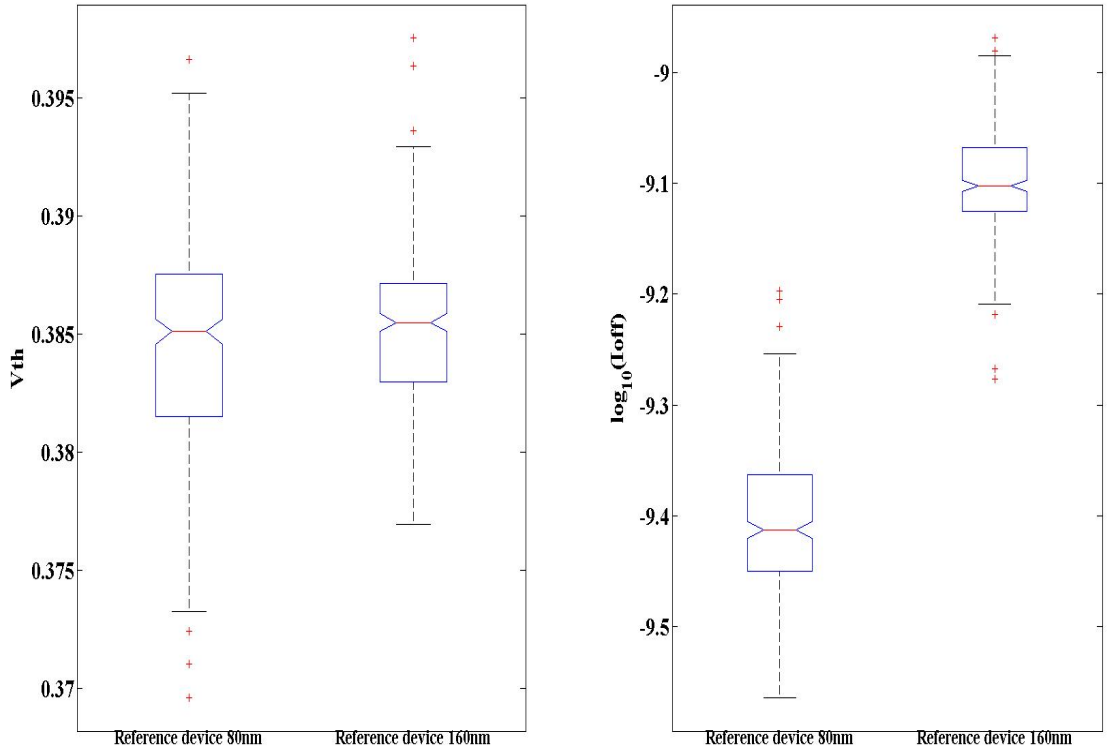


(a)

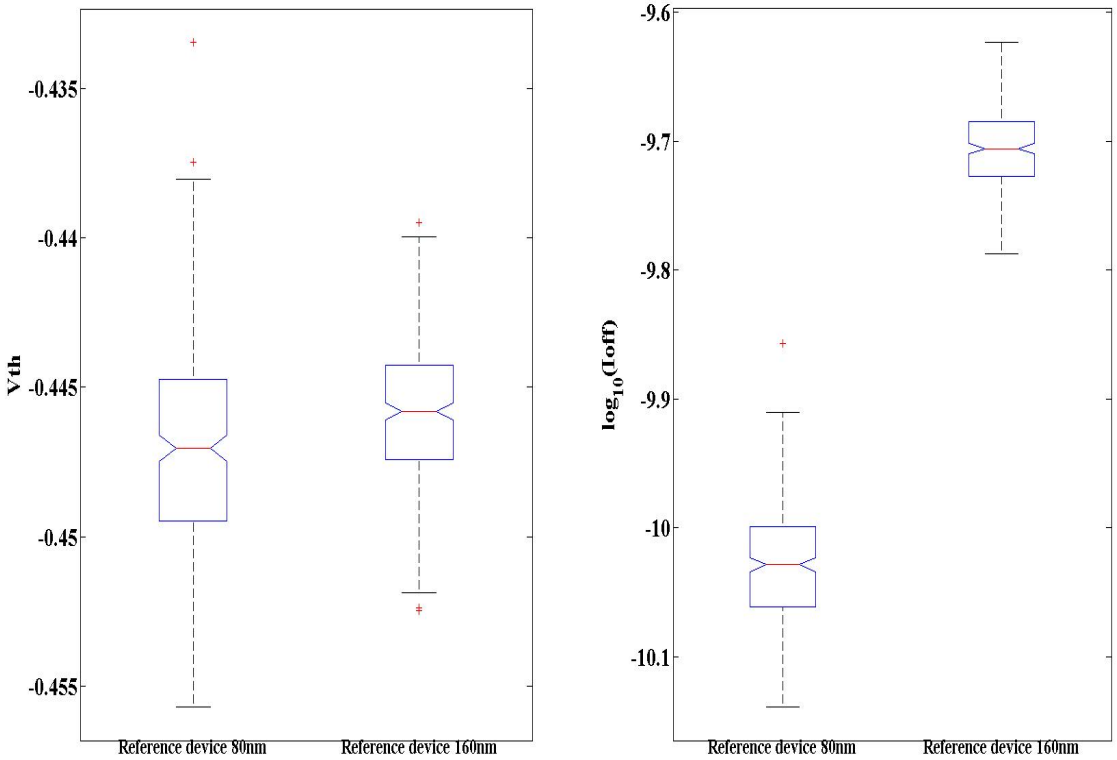


(b)

Figure 3.42: Comparison of reference device with different width at high drain voltage for V_{th} and I_{off} illustrated by the scatter plot and the density plots, (a) n-channel device. (b) p-channel device.



(a)



(b)

Figure 3.43: Comparison of reference device with different width at high drain voltage for V_{th} and I_{off} represented by *Box-and-whisker* plots, (a) n-channel device. (b) p-channel device.

Chapter 4

Evolutionary Algorithms

Contents

As single chip integrated circuits grow to comprise millions of transistors, the complexity of circuit design problems, and therefore design optimisation, become exponentially more difficult. This increase in design complexity has meant that it is impossible to achieve appreciate

designs without the assistance of computer programs. In order to handle the complexity of circuit designs satisfying design specifications and optimisation within reasonable time, modern IC design depends on electronic design automation (EDA) software, which cover virtually every phase of the VLSI design flow. However, most VLSI physical design problems are NP hard combinatorial optimisation and mutually dependent problems that are governed by a large number of constraints and multiple criteria [87]-[89]. The capabilities and limitations of EDA tools have crucial impact on the performance and outcome of the circuit designs as well as on the resources required to design a circuit, with the output being limited by deterministic algorithms and search strategies. Alternatively, stochastic search algorithms, e.g. *Evolutionary Algorithms* (EAs), could be considered to assist EDA tools and provide a possible way to overcome the limitations of conventional algorithms as a result of EAs ability to search large problem spaces comprising of many local minima [90]-[96].

This chapter covers the appropriate level of concepts and principles of EAs to prepare the basic background for the application of EAs on the field of VLSI physical design and optimisation. This chapter is divided into two parts: Section 4.1 describes what an Evolutionary Algorithm is and its evolutionary mechanism flow. Related evolution operations are described and demonstrated as the foundation of background knowledge in order to give an understanding of the proposed bio-inspired extraction algorithm and circuit optimisation approach in Chapters 5 and 6. Several EA operations such as selection, crossover, mutation and fitness evaluation, are discussed in Section 4.2. The details of three variations of EAs widely used are described independently in Section 4.3: *Genetic Algorithm* by Holland [97]-[99], *Evolutionary Strategies* by Rechenberg [100] and Schwefel [100], and Multi-Objective Evolutionary Algorithm [101]-[104].

Although physical design optimisation problems are generally combinatorial in nature, they are often more complex than traditional combinatorial optimisation problems due to the difficulty of managing a set of mutually exclusive or contradicting constraints and searching a large solution space. The second part of this chapter (Section 4.4) introduces the "restricting" factors of evolutionary algorithm applications for the VLSI physical design process, including problem instances characteristics analysis, problem decomposition and application of EA for VLSI designs.

4.1 What is an Evolutionary Algorithm?

EAs are a population-based stochastic search algorithms inspired by the Darwinian concepts of natural evolution [105], and have been widely used to solve complex optimisation problems [94]-[96]. Although many different variants of EAs have been proposed and addressed in EAs techniques development history, the common underlying idea behind these variants of EAs is the same: a population of individuals encode possible solutions to the problem, natural selection (survival of the fittest) is driven by the environment and this should result in a better adapted solution such as an improvement in the fitness of the population [105].

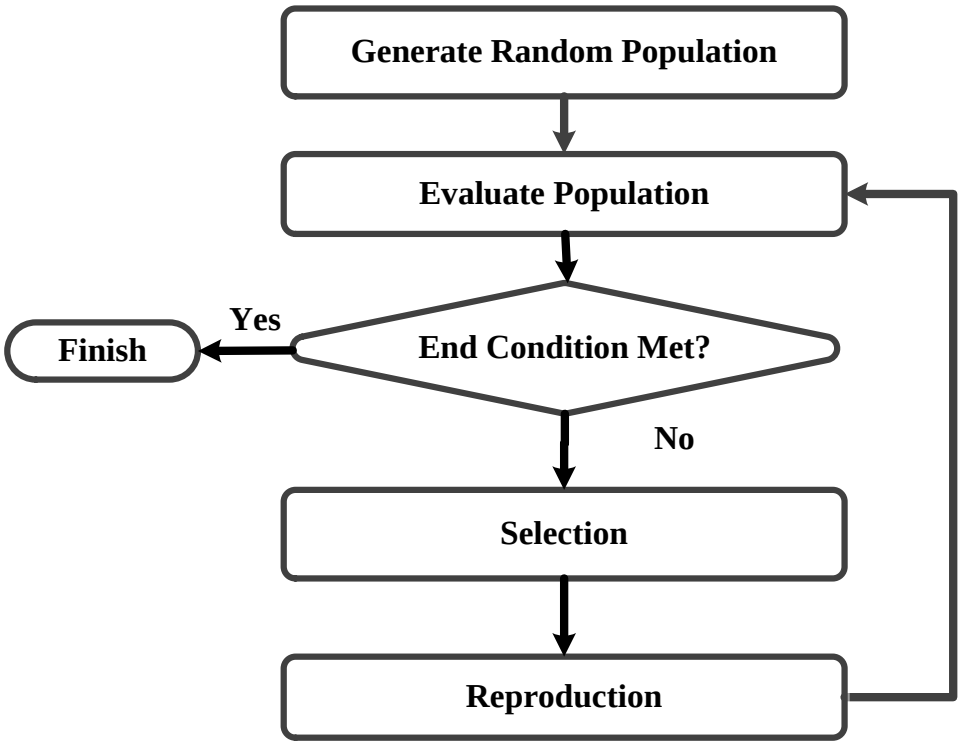


Figure 4.1: Block diagram of the main operations of a generic evolutionary loop [99].

There are two fundamental forces, selection and variation, which constitute the basis of evolutionary systems. A quality (fitness) function is given as measuring criterion so that high quality solutions are selected to seed the next generation by applying a set of evolution variation operations to them, this process is the so-called selection that acts as a force pushing for improving quality [106]. The variation operations incorporate crossover (recombination) and mutation. Generally, one or more new offspring result from crossover (recombination) operating on two or more selected candidates (the parents). The second variation operation is mutation which can also produce new offspring from parents. The marked difference between

crossover and mutation is that mutation operates on one candidate and generates one new offspring. These variation operations (crossover and mutation) not only inherit and keep the high quality features from parents, but also facilitate the necessary diversity in children. A generic evolutionary loop is shown in Figure 4.1 [99] that produces the candidate's whose fitness values improve in consecutive populations [107]-[109]. The evolutionary process is iterated until a solution with sufficient quality is found or a previously stop criteria is reached or triggered.

4.2 Operation of an Evolutionary Algorithm

4.2.1 Representation

The first step in EAs is to set up a mapping between the original problem and potential solutions space where evolution takes place, which is commonly called representation. An individual in an EA consists of a genotype (chromosome) and represents a potential solution for the given problem. Each genotype is constructed of a number of genes. Although variation operations associate separately with representation, a suitable representation actually improves the complexity of variation operation manipulating a solution. For example, given an optimisation problem on integers, these set of integers would form a set of genotype. The integer can be represented by a binary representation. One candidate value is 28; then this integer is encoded as a binary representation, 11100, which can be easily performed variation operations such as bit-flipping mutation, one point/N point crossover. Hence, individuals within an EA made the encoded of candidate solutions to a specific format are easier to be accepted and implemented by variation operations such as binary, integer, floating-point and permutation representation [110]-[112].

4.2.2 Selection

A good procedure for selection, inspired by natural evolution [99], is at the heart of any evolutionary algorithm. The selection process determines which candidates in the current generation survive to be allowed to participate in reproduction and form the new generation, and which ones die out [113]-[115]. Although selection has many variants, a common underlying aim of selection is to promote population members which receive high rewards from the fitness function [99][116]. Elitist strategies are typically used so that only the better solutions are preserved and selected to ensure the fitness of individual in every future generation is at

least as strong that of the parent generation. Some of the selection strategies are described in the following.

Roulette Wheel Selection

Roulette Wheel Selection is the most common fitness-proportionate selection technique. All the individuals in the population are assigned a segment of roulette wheel based on their fitness value and [117]. The ratio of individual's segment in the roulette wheel is ruled "the bigger the value is, the larger the segment is", which means that the size of segment is proportional to the value of the fitness of the individual [99][117]. Then, the process, which this virtual roulette wheel is spun and stopped for individual selection, is repeated until all offspring is selected. Although Individuals with higher fitness have more probability of selection, this algorithm is no guarantee that all best individual will be selected in each time.

Tournament Selection

Tournament selection is widely used selection strategies in evolutionary algorithms due to its extreme simplicity. Tournament selection provides selection pressure by holding a tournament among k individuals, with k being the tournament size. The selection pressure is the degree to which the better candidates are selected: the higher the selection pressure, the more the better individual are selected. The winner of the tournament with the best fitness of the k tournament competitors is inserted into the mating pool for reproduction. The tournament winners comprise of mating pool which has a higher average fitness than the average population fitness. The selection pressure is provided by the fitness difference, which drives the EA to improve the fitness of each succeeding generation [99]. Finally, the tournament selection strategy can easily adjust and control the selection pressure through varying the tournament size k [99].

Rank Selection

Rank selection strategy sorts the population of candidates into a queue and ranks them according to their fitness value; then individuals with high fitness values are selected from this rank queue. Each individual has been allocated the selection probability with respect to its rank [117]. However, unlike in proportional selection, the rank selection controls selective pressure by same selection probability which is assigned to the fittest candidates. In addition,

it also reduces the risk of premature convergence in terms of selection pressure [117]. The advantage of the rank selection is for easily implementation and behaves in a more robust manner than other methods [117]-[119]. The drawback of this strategy is increasing the computational expense as a result of the individual ranking process [99].

(μ, λ) Evolution

Evolution strategies is an iterative procedure relying on a birth surplus [120]-[121], which strictly holds $\lambda \geq \mu$. λ offspring in the current generation are created from their μ parents through crossover and/or mutation, whereas the parents will be replace by μ offspring no matter whether the parents' fitness is better than offspring. Therefore, the fitness value exists fluctuations due to a non-elitist strategy, which means that the best individuals of future generations may perform worse than the best of the current generation [119].

($\mu + \lambda$) Evolution

In order to achieve an alternative elitist selection strategy, the new individuals in the future generation are composed by the union of μ parents and λ offspring. This strategy guarantees the population quality improvement generation by generation.

4.2.3 Reproduction

Crossover

The principle of crossover is to produce new individuals with desirable features from parents in evolutionary cycles [76]. The new offspring genotype is assembled from random parts of the parents genotypes, merging the genetic information from the parents into the offspring genotype. Although crossover, as a stochastic operator, provides a reasonable way for offspring to combine both parents' features, the generated offspring may have undesirable combinations of traits due to random recombination. Different variations of crossover are implemented due to a variety of representation [99]. For example, three standard forms of crossover are generally used for binary representation, which are single-point, multi-point and uniform crossover. For floating-point representation, except an analogous operation to these used for binary representation being perform floating representation, genotype represented by floating-point representation can be typically performed by arithmetic recombination, which the details of this operation are described in literature [77]. In addition, these variations also affect the

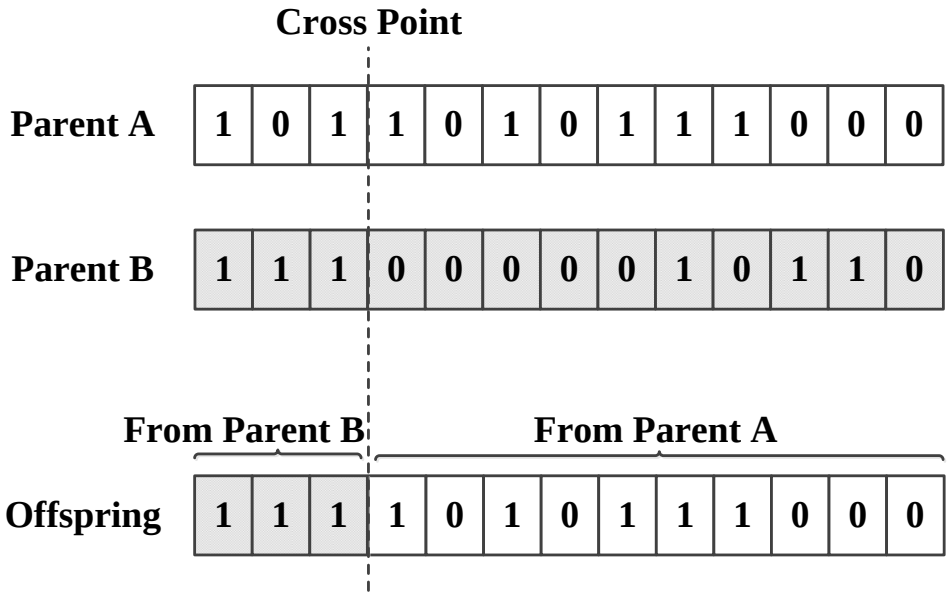


Figure 4.2: Example of single-point crossover operation used in evolutionary algorithms

offspring's genetic features in terms of information context inherited from the parents. Three crossover techniques are discussed here to demonstrate this.

Single-point Crossover Single-point crossover is the most simple crossover operation. The genotype of offspring is combined from portion of two parent genotype. The single-point crossover normally chooses a random crossover point in the gene string of parents. The left of the crossover point with the segment of parent A is copied to offspring, it scans the right of the crossover point with segment of parent B, gene by gene, as illustrated in Figure 4.2. The principal behind the single-point crossover is that offspring can be produced by swapping the portions of two parent genotype divided by crossover point.

Multi-point Crossover Multi-point crossover processing mechanism is similar to the single-point one. The marked differences is that the genotype of parents are split into N segments (substrings) by $N\text{-}1$ randomly selected crossing points, swapping these segments [124]. Figure 4.3 shows an example of two-point crossover, in which segments between the two crossing points are swapped between the two parent genotype.

Uniform Crossover Unlike single-point and multi-point crossover, which directly use

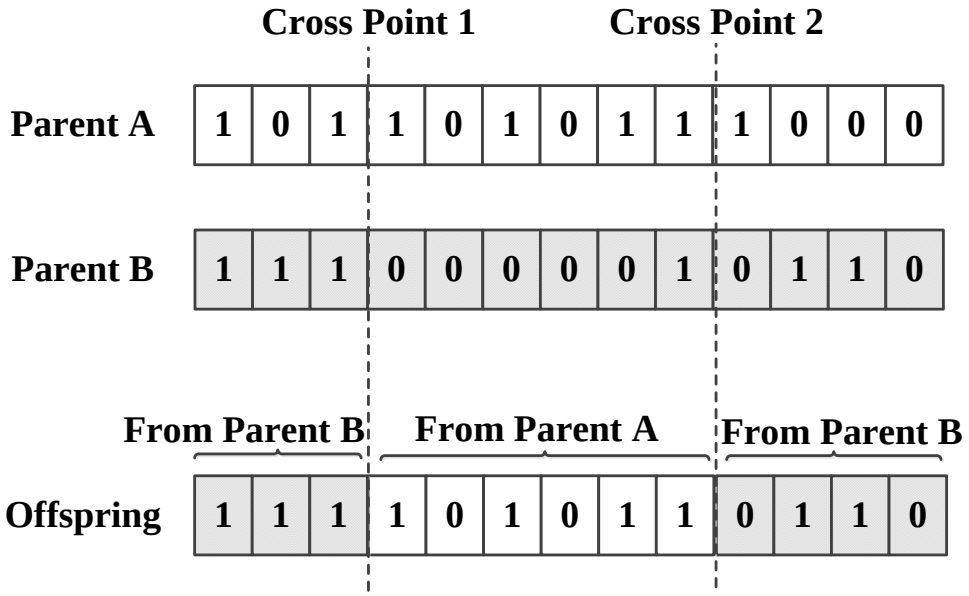


Figure 4.3: An example of the multi-point crossover

crossing points to divide/split the genotype of parents at the segment level, uniform crossover works at the gene level using a predetermined parameter to operate crossover instead of fixed points [99][124]. Uniform crossover works by treating each gene independently and makes a choice through the predetermined parameter to which genes from parent A or parent B participate in the exchange. A set of random variables are randomly generated from a uniform distribution [0, 1]. Each random variable relates to one position in offspring genotype. The offspring's genotype has a corresponding string constituted by a set of random variables from a uniform distribution over [0, 1]. In each gene position, If the corresponding random value is below the predetermined parameter, the gene is inherited from the parent A; otherwise, from parent B. Therefore, the evolved gene from either parent A or parent B is randomly selected [124]. Then, the inverse mapping uses to generate the second offspring. An example of uniform crossover is illustrated in Figure 4.4. The random variable string is first generated from uniform distribution [0, 1] and the predetermined parameter is set to 0.5 to make a choice to select genes from parent A or parent B.

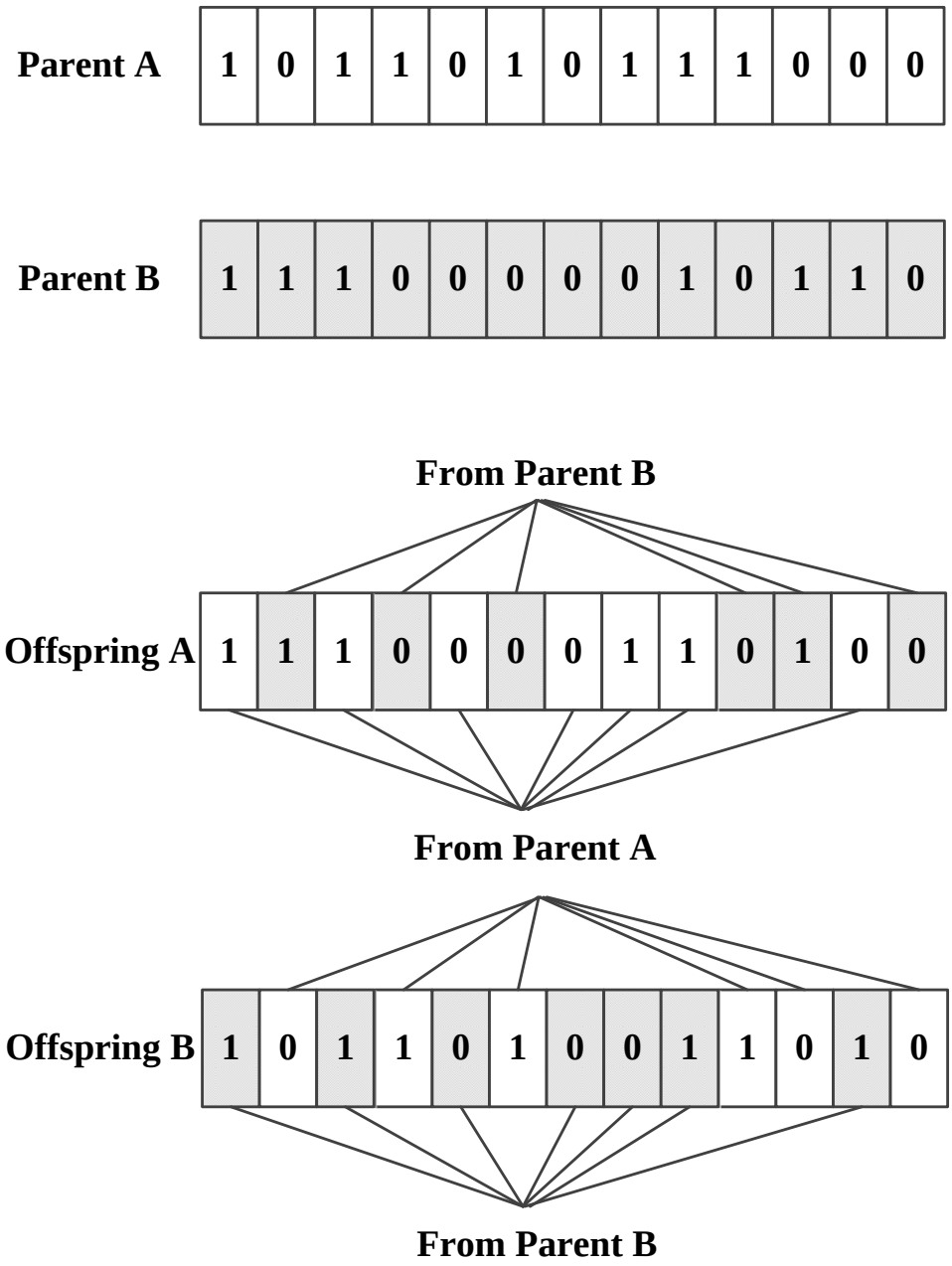


Figure 4.4: Uniform-point crossover. In this case, the predetermined parameter is 0.5. The corresponding random variables string is [0.20, 0.56, 0.41, 0.75, 0.33, 0.62, 0.11, 0.28, 0.39, 0.67, 0.87, 0.11, 0.75]. For offspring A, if random variable in each gene position is below 0.5, the gene come from parent A; otherwise from parent B. The inverse mapping is used to create offspring B.

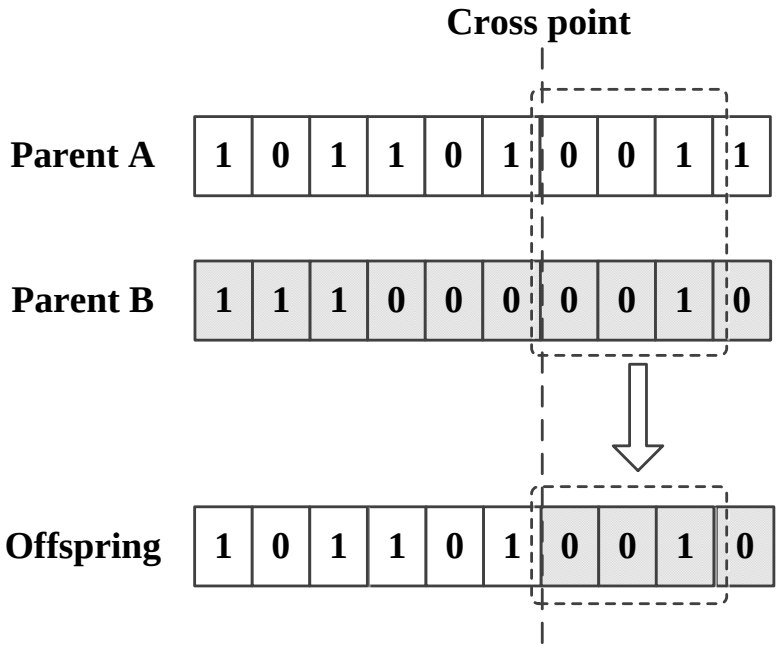


Figure 4.5: An offspring is generated from one-point crossover. The drawback of crossover is that offspring genotype is very close to the genotype of parent A because portion of genotype segment in both parents is same, leading to population diversity reductions. In order to solve this problem, the need introduces new gene into mating pool to keep the population diversity.

Mutation

Mutation is an important variation operator for increasing population diversity that has been seen as supplying "fresh blood" to the gene pool [125]. Although crossover is an efficient genetic rebuild mechanism, by itself, the reachability of the entire potential solutions in search space cannot be guaranteed owing to the fact that it only comprises the inheritance mechanism from the exiting genes [108][126]. As an example, consider the genotype of individual A and B with binary-coded are both composed by 10 bits. Figure 4.5 shows two individual A and B performing crossover. The new individual handled by crossover in particular bit positions, bit 8, 9, 10, have the same value. In such a case, crossover cannot create an individual with different values in these positions, without mutation. Thus, mutation fills in this gap. The mutation provides more solution choices in search space as a series of random variations are delivered into the new population [126]-[130]. The search space has been enlarged in terms of increasing the chance of finding the best solution.

Different genotype encoding methods (representations) determine which ones of the various mutation strategies can be applied and work best. One of the simple mutations is bit-

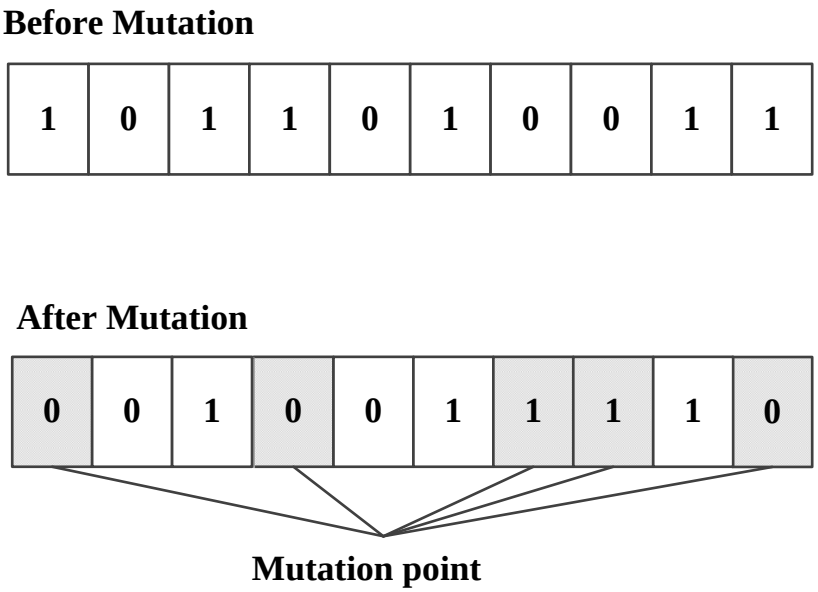


Figure 4.6: A common and simple mutation operation in binary representation. The bit-wise mutation randomly selects a set of mutation positions to invert gene value, for example, "0" becomes "1" and "1" becomes "0".

wise mutation, where a randomly selected portion of genes from the genotype is flipped. Figure 4.6 depicts the operation of the bit-wise mutation. Mutation can also be used for integer or floating-point representations. There are two common methods to generate a new gene value. One is to add a new random value following a specific distribution with regard to the original value. Alternatively, a completely new value generated randomly from a specific distribution but within the specific range replaces the old one [99].

During mutation operation, one basic genetic parameter, mutation probability (P_m) [131], is used to govern and control how parts of genotype will be mutated. Mutation probability is defined as the probability of mutating each gene. For example, if mutation probability is 100%, whole genotype is changed, if this parameter is 0%, nothing is changed [132]. This parameter controls the rate at which new gene values are involved into the population in terms of determining the reachability of the algorithm. However, a poor choice of mutation probability causes that certain gene values would never be tried out such as if it is set either too low or too much random perturbation would appear resulting in loss of inheritance from their parents.

When only mutation used in algorithms, without crossover, the mutation with high mutation probability is equivalent to random search, random modification of the potential solution and

acceptance if there is improvement [133]. In contrast, a low level of mutation or mutation with appropriate mutation probability will promote to prevent and control premature convergence [134]. Based on specific problem analysis, the proposed bio-inspired approaches for model extraction and circuit performance optimisation only use mutation with appropriate mutation probability to perform evolution operation; and the details of these algorithms are introduced in Chapters 5 and 6.

4.2.4 Evaluation

Evaluation forms the basis of selection, which provides a "ruler" to assess individuals and check whether they meet the optimisation requirements [87][99]. A quality function, called fitness function or cost function, is used to transfer the real world problem to a specific description that can be easily handled to assess the individuals within EAs. In other words, the fitness function defines what improvement means and identifies how close the features of an evaluated candidate solution are to that of the desirable solution. In addition, a fitness function with various constraints would guide EAs to find different solutions with a number of independent desirable characteristics, particularly handling multi-objectives optimisation problems [135]-[136]. For example, when evolving logic circuit designs, the fitness function may also reward acceptable solutions with minimal influence of the device variability, faster solutions with a minimal gate delay, or tradeoff solution that of appropriate speed and variability tolerance.

4.2.5 Termination Criteria

How to terminate EAs, i.e. when and how the algorithm should be stopped, is the last consideration in EA design. Generally, two things are used as termination criteria: One criterion depends on reaching the predetermined limit, for example, a pre-defined maximum generation number; the other criterion is where a perfect solution is found that meets the specification requirements, at which point the algorithm can terminate [137]. For certain problems, particularly optimisation problem, there is a risk that using the second criterion as stop condition; the condition will never be satisfied and the algorithm may never stop [101]. Therefore, in many cases the termination criterion is often a combination of both time and correctness.

4.3 Type of Evolutionary Algorithm

This section outlines the principle operation and differences between several variations forms of evolutionary algorithm, including genetic algorithm, evolutionary, evolutionary programming, genetic programming and multi-objective evolutionary algorithm.

4.3.1 Genetic Algorithm

The Genetic Algorithms (GAs) was first introduced in John Holland's book "*Adaptation in Natural and Artificial Systems*" [97]-[99], in 1975, which provides a simple framework for attempting to solve complex optimisation problem. The first task when designing GAs, or EAs in general is to decide how best to represent candidate solutions in the algorithm [101]. Although their representations and associate evolutionary operators can be considered separately, different types of representations need a set of variation operators to suite them. Thus, the "right" representation is important for effective evolutionary search in GAs. In addition, the range of gene value and form of gene in representation can be restricted by priori information and problem constraints. For example, if optimisation targets are a set of floating point value array, the floating-point representation is selected to represent genotype. If each optimisation targets have a limitation range, the gene value also can be limited into specific interval according to problem requirements/constraints. Amongst others, four different representations are widely used in GAs: binary, integer, real-value, and permutation representation. The candidate solutions are usually called individual comprising "chromosomes" in GAs, and a group of individuals forms a population. As previously described, two basic evolution processes are handled in EA: *inheritance* and *survival*. Inheritance, including crossover and mutation, involves passing down of features from parent to offspring [137]. Survival refers to elite individuals selected from a population to weed out individuals representing bad solutions.

The advantages of GAs include: adaptation and learning from past information, intrinsic parallelism of the population-based approach with little computational overhead, and efficient handling of complex problems [108]. Of course, GA is a heuristic procedure; hence, there is no guarantee to find the optimum solution in a set amount of time. However, GAs are suitable algorithm to find good solutions for the most of optimisation problems without requiring specialist knowledge domain or an strictly analytical description of a problem. A simple GA (SGA) is summarised in Table 4.1 [99].

Table 4.1: Sketch of the simple GA [99]

Representation	Bit-string
Parent selection	Fitness proportional
Recombination	1-point crossover
Mutation	Bit flip
Survivor Selection	Generational

4.3.2 Evolutionary Strategies

Evolutionary Strategies (ES), were firstly developed independent if other EA algorithm families by Rechenberg, Hans-Paul Schwefel and Peter Bienert, in the early 60s [99]-[100]. The candidate solution encoding used in ES is a series of real-value numbers because ES were historically designed for continuous parameter optimisation tasks, *e.g.,* the design of a 3D convergent-divergent hot water flashing nozzle [99][138]. Early ES work on (1+1)-ES with a deterministic step size control to achieve linear convergence order [103][139]. Here, one parent and one offspring formed by mutating the parent make up the population. Only if the offspring fitness is at least as good as the parent, it will become the parent in the next generation. Otherwise the parent continues to the next generation.

Schwefel extended the (1+1)-ES towards using multi-parent strategies employing self-adapting mechanisms [140]-[141]. The marked difference between early ES and other EAs is the use of self-adaption of strategy parameters in the individual representation that are used to control the behaviour of the mutation operator, *e.g.,* by introducing small random changes from a Gaussian distribution into mutation so as to bypass the local optimum [99][101][103]. The self-adaption mechanism, as a useful feature so that some of parameters of ES can be changed in a specific manner, is still included in modern ESs. For reference, A sketch of ES is given in Table 4.2 [101].

Table 4.2: Sketch of ES [101]

Representation	Real-value vectors
Recombination	Discrete or intermediary
Mutation	Gaussian perturbation
Selection strategy	$(\mu\ ,\ \lambda)$ or $(\mu\ +\ \lambda)$
Speciality	Self-adaption of mutation step sizes

4.3.3 Evolutionary Programming

Evolutionary Programming (EP) is another variation form of evolutionary computation paradigms, which is first defined by Fogel in 1964 for performing a learning process aiming to generate artificial intelligence [15]. The EP uses finite state machines (FSM) as predictors and evolved them. FSM operate on a finite set of input symbols, possess a finite number of internal states, and produce symbols from a finite set [154]. The corresponding input-output symbol pairs and set of state transitions specify the behaviour of the FSM [154]. EP performs a population of FSM. One of five mutation modes is randomly used to perform existing FSMs in the population as parent solutions to reproduce the children FSMs. These five mutation modes are change an output symbol, change the initial state, change a state transition, add a state and delete an existing state [15]. A sketch of EP is given in Table 4.3 [99].

Table 4.3: Sketch of EP [99]

Representation	Real-value vectors
Parent Selection	Deterministic (each parent creates one offspring via mutation)
Recombination	None
Mutation	Gaussian perturbation
Survivor Selection	Probabilistic ($\mu + \mu$)
Speciality	Self-adaption of mutation step sizes

4.3.4 Genetic Programming

Genetic Programming (GP) is an evolutionary algorithm-based methodology, which optimises computer programs based on a fitness landscape to perform a user-defined task [15]. GP can be considered as specialisation of GA where each individual is a computer program. An automatic system is created by the objectives, accepting a high level statement of a problem as its input and creating some form of computer program to solve the problem without human-intervention in the programming [15]. The initial population is generated by a large number of random programs. Generally, GP evolves computer programs, represented in memory as tree structure. Each tree node is an operator function and terminal node is an operand. Comparing to the representation of GAs or ES, GP representation has two different aspects. The first aspect is chromosomes in GP are nonlinear structure rather than linear vectors in

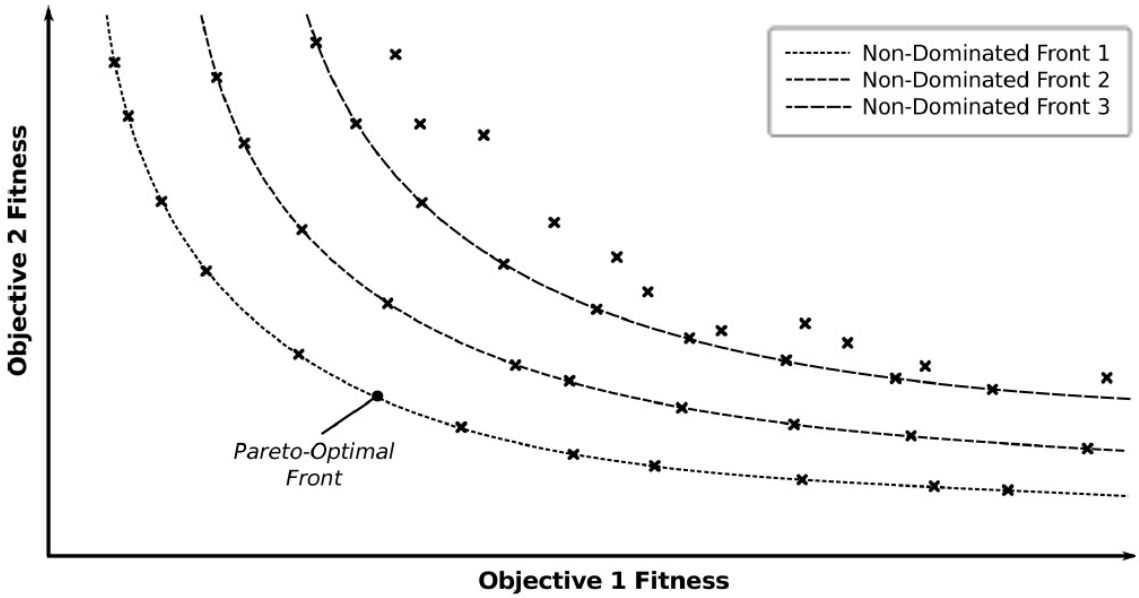


Figure 4.7: An example of individuals in a multi-objective problem. The bottom line is the Pareto-optimal front [103]. If none of the objective functions can be improved in value without degrading some of the other objective values, which solution is so-called non-dominated [144].

GAs or ES. The second aspect is that chromosomes in GP can differ in size, whereas the chromosome length in GAs or ES is usually fixed. Although GP also includes crossover and mutation operations, crossover and mutation applied in GP has slightly different from these used in GA due to tree-based representation. The details how to perform crossover and mutation in GP are introduced in literature [99]. A Sketch of GP is shown in Figure 4.3 [99].

Table 4.4: Sketch of GP [99]

Representation	Tree structure
Parent selection	Fitness proportional
Recombination	Exchange of subtree
Mutation	Random change in tree
Survivor Selection	Generational replacement

4.3.5 Multi-Objective Evolutionary Algorithm (MOEA)

MOEA use a population based approach that is developed from traditional single-objective evolutionary algorithms, where more than one solution participates in an evolutionary loop and all solutions are evaluated by trade-offs of performance between different objective func-

tions that many mutually conflict [103][142]. Although many algorithms and application studies involve performing multiple objectives, the majority of these methods avoid the complexities performing a true multi-objective optimisation problem and transform multiple objectives into a single objective function by using some user-defined parameters [155]. In fact, multi-objective optimisation is treated as several single-objective optimisations for handling multiple objectives, respectively. In order to balance these multiple objectives or achieve specific optimisation targets, these multiple objectives are weighted and combined into a single measure [156]. When two objectives mutually conflict, there is no single optimum solution. The common method trade-off between conflicting objectives usually sacrifices one objective and gains in the other one [155]. However, the true multi-objective optimisation method based on several specific optimisation techniques (such as Pareto-based algorithm [155], average ranking algorithm[192], grid-based EA [192] etc.) is to find a set of solutions which are all optimal and no solutions from the set of optimal solutions can be said to be better than any other [155]. Generally, with the multi-objective optimisation algorithms it is difficult to find a suitable solution to guarantee all objectives are being simultaneously improved in cases where objectives are mutually conflicting. In order to solve this problem and intuitively reflect the set of optimal solutions, one of commonly used method, Pareto-based fitness assignment method, exhibits all solutions on a large N-dimensional (N= No. of objectives) trade-off surface so that the algorithm finds a satisfactory solution in which fitness is improved in one objective without causing detriment in another [103][143]. All such solutions form part of the Pareto-optimal front and are considered to be non-dominated, *e.g.*, at least no worse than or equal to (high-fitness in at least one objective) any other population member. Figure 4.7 shows an example of Non-dominate fronts highlighting the Pareto-optimal fronts [103]. MOEA have many different variations such as Fonseca and Fleming's MOGA [143], Srinivas and Deb's NSGA II [144], and Horn *et al.'s* NPGA [145]. Although each variation has a different taken on multi-objective optimisation, there are two common features: first, the population fitness assignment based on non-dominated sorting; Second, to preserve diversity among solutions of the same non-dominated front.

4.4 Evolutionary Algorithm for VLSI Design

4.4.1 Charateristics of Problem Instances

An EA is applied to solve a complex problem of VLSI optimisation and overcome the limita-

tion of EDA tools due to its ability of effective global search. Since EAs are population-based approaches, solution quality depends heavily upon the population size and the number of generations parameters are evolved for to decide whether sufficient diversity and convergence has been achieved [96]. As population size increases, the solution space is more accurately reflecting the domain of real-world problem; and high quality solutions can possibly be found more easily. However, the more elements that have to be evaluated, the more simulation time will be consumed, particularly if several constraints, like area, delay, power and robustness, are considered simultaneously [94]. Searching a huge solution space using a large population is a computationally expensive task. Thus, parameters of EAs need to be set appropriately so as to balance solution quality and simulation time within reasonable runtimes of the algorithm.

4.4.2 Problem Instances Decomposition

Most VLSI optimisation problems are very complex and are restricted by many constraints. It is difficult to find solutions that are satisfactory with respect of a number of competing or contradicting criteria, which poses great challenges to designers. To decrease the complexity of these optimisation problems, one possibility is to decompose the optimisation process into several steps (stages), or divided optimisation problem into a series of simply sub-problems, so that these sequential steps (or sub-problems) can be more easily handled by an EA. The idea of problem decomposition is to reduce the search space to simplify the complexities of VLSI optimisation problems to a manageable level [146] and to improve EAs performance [147].

4.4.3 Application of EA on VLSI Design

The most VLSI optimisation problems are complex problem with a large number of non-trivial constraints, which belongs to NP-complete and NP-hard categories [157]. Evolutionary algorithms demonstrate several potential advantages to find optimal solutions to VLSI optimisation problems [15], These advantages of EA include natural automatically searching solutions space, inherently parallel performing optimisation process and effectively solving NP-hard problem [15][87][157]. In order to comprehensively know the history of EA widely used in digital and analogue design and optimisation problems, Table 4.5 lists several synopsis of notable examples.

In addition, two examples as demonstration how to apply EA in VLSI design and optimisa-

Table 4.5: Examples of Evolutionary Algorithms in Electronic Circuit Design

Author	Notes
Koza, (1992)	GP used to evolve boolean functions, multiplexers & parity functions [158].
Roper, (1995)	GAs used to perform the Automatic Generation of Test Data [159], which result found that genetic parameter values (such as mutation rate, crossover rate and population size) the greatly affect the speed at which a solution is found.
Koza, (1997)	Use of GP to evolve analogue computational circuits. The functions were square, cube, square-root and cube root. The circuits used BJ transistors and passive components [160].
Miller, (1998)	Evolution of binary multiplexers, adders and multipliers using CGP. The results included a design for 3-bit multiplier that is 20% more efficient in terms of gates than the best known conventional design [120].
Mazumder and Shahookar, (1999)	GAs used to solve the circuit partitioning problem [157].
Langeheine, (2002)	Optimisation on actual logic gates on FPTA chip [161].
Koziel, (2003)	Use of an adaptive EA to lower the power consumption of CMOS circuits during high-level synthesis, with the goal of reducing average and peak temperatures [162].
Wu, (2005)	CGP used to evolve low-pass filter topologies using SPICE simulation [163].
Trefzer, (2006)	Used multi-objective approach on FPTA with 11 different fitness criteria [164].
Li, (2007)	Extraction of CMOS device models parameters using an adaptive sampling based GA [165].
Xiao, (2009)	MOGA is developed to optimize the soft error tolerance of standard cell circuits. Results decrease the soft error rate (SER) 74.25% with very limited delay overhead (0.28%) and reduce the area for most of the circuit under test by average 5.23% [166].
Hilder, (2010)	CGP used to evolve standard cell library to achieve variability-aware design [15].
Walker, Terfzer, Tyrrell, (2013)	MOCGP used to design function decoders for the PAnDA architecture [167].

tion to produce good quality outputs are introduced in the following. One example is the application EA on a typical transistor model parameter extraction procedure. Generally, the transistor model is described by large number of equations with many parameters. Adjusting these parameters to fit the actual curves of the expected device is a difficult task for

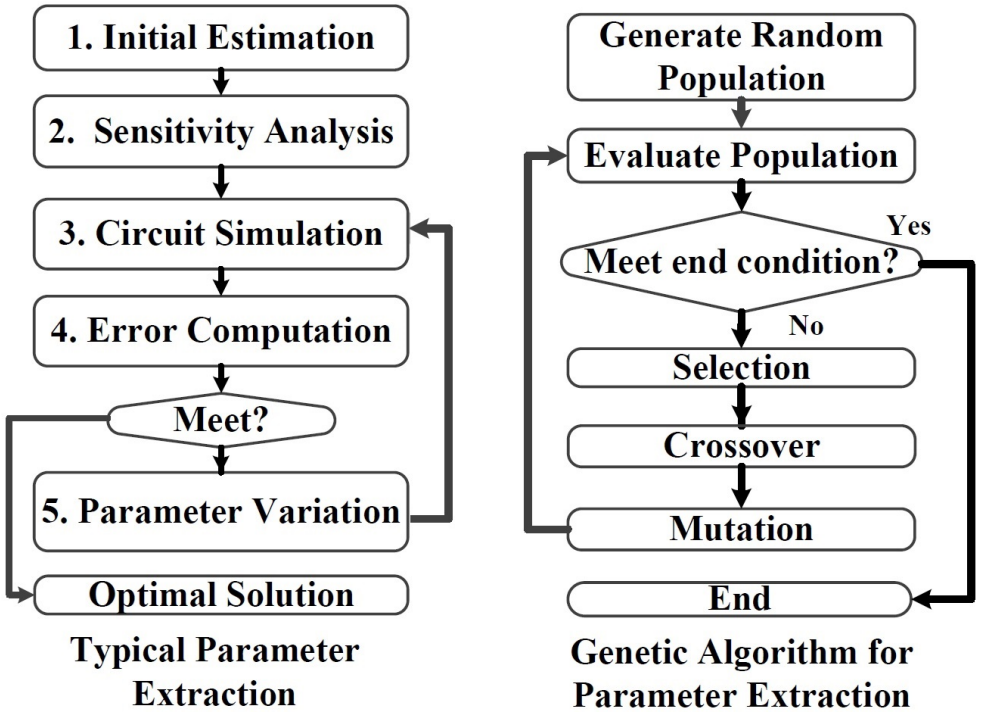


Figure 4.8: Comparison of a typical parameter extraction procedure (left subfigure) and genetic algorithm (GA) (right subfigure). The circuit simulation and error computation in parameter extraction is analogous with fitness evaluation in GA. The parameter variation can be performed through evolutionary operations.

designers, which not only requires designers to test the large number of parameter and their combinations, but also requires designers to have a lot of experience and technical knowledge of parameter prediction and estimation, especially the initial model parameter estimation. Therefore, alternative automatic method, using EA, is considered to solve the increase of the demand for automatic parameter extraction method. In order to easily understand the advantage of EA on the model extraction, a brief compact model extraction process is first introduced. Figure 4.8 shows a comparison of a typical parameter extraction procedure and genetic algorithm (GA). The first step in extraction process is to give the initial model parameter estimation from measurements and technology process data [149]-[150]. However, if these initial parameters are poor, the extraction result will be poor and find it difficult to converge. Thus, this step heavily depends on the designer's expertise. Compared to conventional method, the advantage of using EAs for parameter extraction is its natural automatic tuning abilities, even starting from poor initial parameters estimation. Therefore, EA provide good tolerance to initial parameter setting, while requiring less specialist knowledge of model extraction or related parameter extraction experiences. Once the initial starting point is completely set, a sensitivity analysis will be performed to determine how these param-

eters influence a given target. Next steps in a loop are parts of the optimisation process through simulation, error calculation and parameter variation to achieve parameters fitting. A conventional extraction and optimisation method is used in this procedure that performs a succession of optimisations of one characteristic at a time, until convergence is reached [149]. Although conventional methods have low computational cost and can easily perform single objective optimisation, the results often fail to converge when either two or more characteristics are conflicting or poor initial parameters estimation is chosen, further resulting in the bonus from low computational time also eliminating [149]-[151]. However, EAs, due to intrinsic parallelism and multi-objective optimisation mechanisms, can efficiently perform the model extraction problem and overcome convergence issues seen in conventional method. Thus, a novel two step EA extraction approach has been proposed and it is discussed in Chapter 5.

Another example of the application of EAs on VLSI design optimisation is circuit performance optimisation. Circuit's operating speed and power consumption are two primary performance characteristics. The delay/speed of components in a circuit will change as the prior-state conditions, the output load, and the path between input and output varies. Components can be single devices or consist of a whole circuit. The overall circuit performance is influenced by these components due to their different delay characteristics. The same situation occurs in the case of power consumption. Thus, in order to comprehensively understand circuit performance, it is necessary to exhaustively test every possible combination of different components. However, exhaustive testing is not a realistic option, especially for large scale circuit designs where the number of components/transistors exponentially increases. Thus, EAs are considered a convenient way to handle these optimisation problems, especially multi-objective optimisation problems. The circuit is encoded into a string of genes. Each component (*e.g.,* transistor, resistor, capacitor, and so on) or component features (e.g. transistor size, resistor/capacitor value, and so on) can be represented by different gene [152]-[153]. Then, the circuit is optimised through manipulating these genes in the evolutionary optimisation process.

4.5 Summary

This chapter has introduced and studied what an evolutionary algorithm (EA) is, their components and related variants of EAs and EA application to VLSI design. The description of EAs is based on a general scheme that gives rise to other forms of EA variants. These are

explained in Section 4.1. In order to further comprehend the principle of EA, several general concepts and evolutionary operators have been addressed in Section 4.2, including representation, selection, crossover, mutation, evaluation, and termination criteria. Three examples of EAs: GA, ES and MOEA, have been described in Section 4.3. Due to the complexity of VLSI designs and limitation of current EDA tools, EAs are applied for solving these problems due to the global search ability of EAs. The characteristics of VLSI design problems and their decomposition for effective used in EAs are introduced in Section 4.4. This chapter introduced the necessary basic concepts and principles of EAs forming the basis of understanding the application of EAs in Chapters 5 and 6. The following chapter introduces the detail of motif model extraction using a novel two step EA BSIM model extraction methodology. Additionally, a statistical variability modelling methodology will be addressed.

Chapter 5

Extraction and Modelling Statistical Motifs Compact Model

Contents

In modern circuit design, the expense of fabrication and testing means that accurate modelling and simulation are crucial to the continued progress of the industry. However, the modelling and simulation of semiconductor devices is a difficult and computationally intensive task. Typically two simulations, TCAD simulation and Simulation Program with

Integrated Circuit Emphasis (SPICE) simulation, are used to investigate the characteristics of device/circuit. The advantage of TCAD simulation was discussed in Chapter 3 such as improvement accuracy of device simulation result and easily achievement of fully 3D device process emulation and simulation. However, computational time has been sacrificed for the sake of obtaining accuracy, especially for circuit level simulations. On the other hand, SPICE simulation also provides accurate device simulation by using many approximation equations (so-called device compact model) rather than fundamental physical models. The advantage of SPICE simulation based on compact models is that the computational time is much shorter than the TCAD method, particularly convenient for performing circuit simulation rather than simple transistor simulation. However, the compact models consist of a large number of equations with many parameters, explicitly determining these parameter values that is a trivial process.

In this chapter, to accurately reflect the device behaviour so that it efficiently predicts the circuit outcome, the device simulation moves from TCAD to SPICE simulation, where a set of device (mainly device layout motifs) compact models are extracted. An overview of the compact model and its analytical equations is introduced in Section 5.1 so that readers can intuitively understand the related compact model knowledge and model extraction process. Because of the limitation of conventional mathematical model extraction and optimisation algorithm on poor convergence without good initial conditions and the difficulty of simultaneous multi-objective optimisation [168]-[169], a novel two steps evolutionary algorithm (2SEA) performing motifs model extraction is proposed in Section 5.2; and the motifs model extraction results are discussed in Section 5.3. In order to explore the issue of circuit performance variability at the transistor level, statistical variability motifs modelling method is discussed in Section 5.4.

5.1 Device Compact Model and Extraction Method

5.1.1 Device Compact Model

Device compact models are a key interface between device technology and circuit design, which incorporate many parameters to accurately model circuit components in circuit designs, which are extensively used in transistor-level circuit design and verification, particularly in standard-cell characterization procedures [170]. The compact model describes the device's current-voltage characteristics and accounts for all the major physical effects in state-of-art

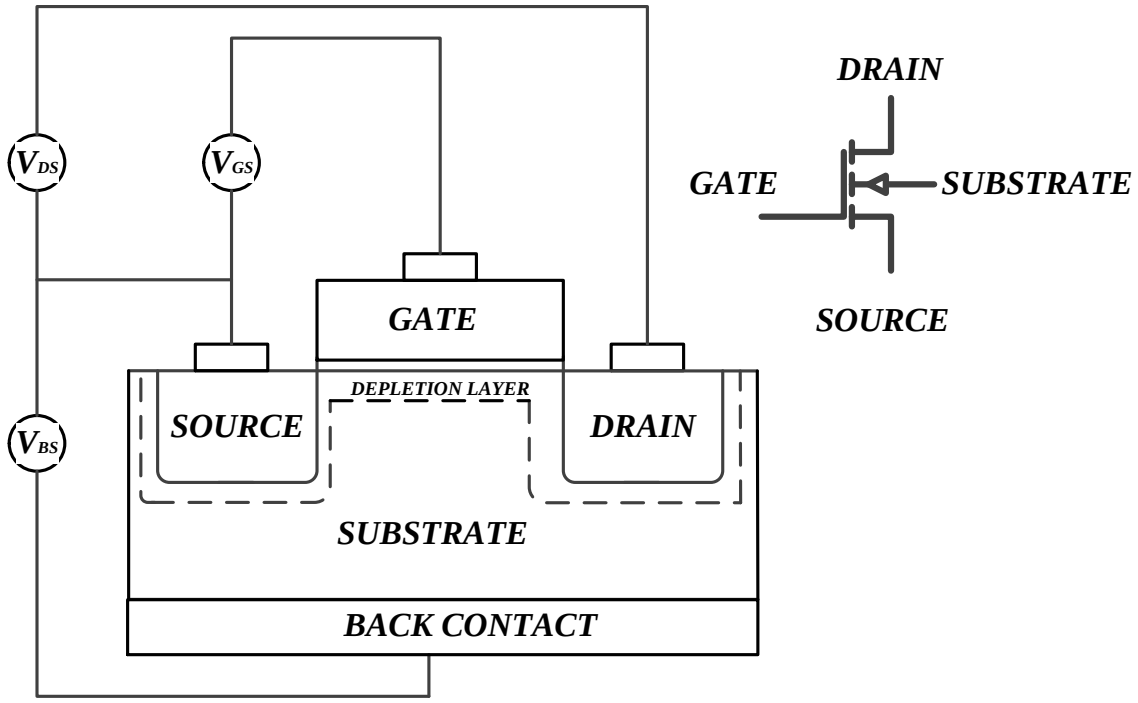


Figure 5.1: Cross-section and circuit symbol of n-channel MOSFET [170]. When the gate voltage V_{GS} is positive and over the threshold voltage, the NMOS is on and an n-type conductive channel between the drain and source below the oxide is formed. When the gate voltage is below the threshold voltage, the NMOS is off. In this case, a four terminal NMOS circuit symbol is displayed.

MOSFET device. The device model can be seen as an equivalent circuit model for a typical single device (Cross-section and circuit symbol of n-type MOSFET is shown in Figure 5.1 [170]).

Complex but accurate models from UC Berkeley were developed to model device properties, which can be divided into different classes: First generation models (level 1, level 2 and level 3 models), second generation models (Berkeley Short-channel IGFET Model (BSIM), HSPICE level 28 and BSIM2), third generation models (BSIM3, level 7, level 48, etc.) and four generation models (BSIM4, etc.). BSIM4 model, as extension of BSIM3 model, mainly addresses the MOSFET physical effects into *sub-100nm* regime [171]. Comparing to previous generation models, BSIM4 has a better models of short channel effects, local stress, transistor operating in the sub-threshold region, gate leakage, temperature variations, etc [171]. In addition, the equations used in BSIM4 have better convergence ability than previous generation models during circuit simulation. A comparison of the performance and SPICE model

Table 5.1: A summary of different SPICE compact models performance and its worked on minimum gate length and oxide thickness [15].

Model	Minimum L (μm)	Minimum T_{ox} (μm)	Subthreshold Accuracy	Small Signal Accuracy
MOS Level 1	5	50	Not modelled	Poor
MOS Level 2	2	25	Poor	Poor
MOS Level 3	1	20	Poor	Poor
BSIM1	0.8	15	Fair	Poor
BSIM2	0.35	7.5	Good	Fair
BSIM3v2	0.25	5	Good	Good
BSIM3v3	0.15	4	Good	Good
BSIM3v4	0.13	3	Good	Good
BSIM4v6	<0.09	<3	Good	Good
PSP103	0.01	0.001	Good	Good

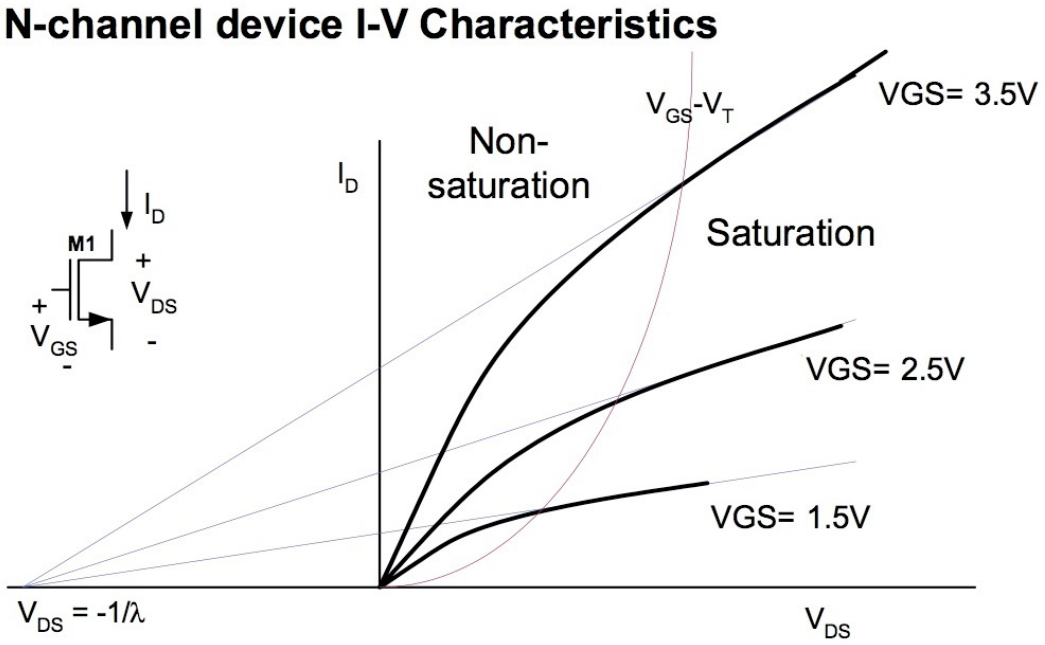
worked minimum gate length and the oxide thickness is summarised in Table 5.1 [15].

BSIM4, which consists of a set of equations with over 300 parameters, is chosen to model the motifs' properties and physical effects of motifs developed in *50 nm* technology, because BISM4 models provide precise device behaviour modelling at *sub-100 nm* regime. In order to promise reliable performance and precisely reflect the device characteristics on the circuit design, understanding the link between the properties of device and device model parameters is critical, which serves as the foundation of the beginning of device modelling.

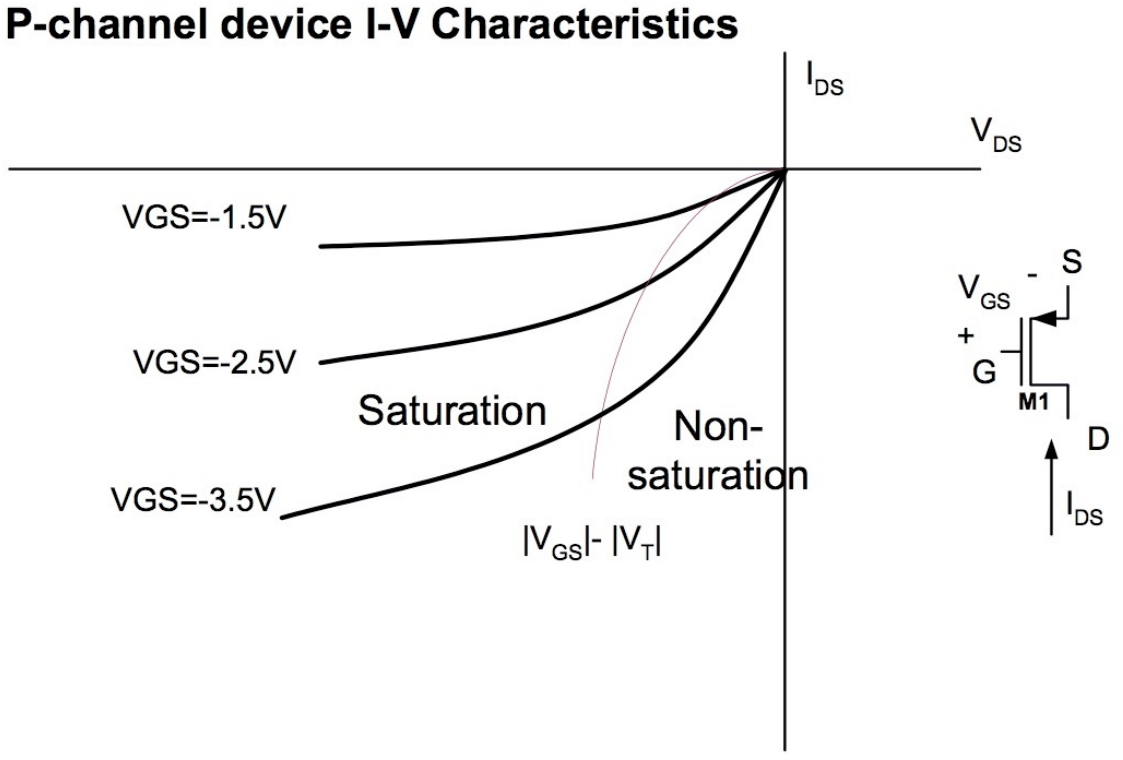
The device electrical characteristics are usually represented by the current-voltage (I-V) characteristics. The I-V characteristics of n- and p-channel device in the linear and saturated regions are illustrated in Figure 5.2 [172]. The boundary between the linear and saturation regions is distinguished by the Equation 5.1. Various regions of operation for these two different type devices are outlined in the Table 5.2 [172].

$$V_{ds} = |V_{gs} - V_t| \tag{5.1}$$

A set of model parameters are configured to fit the device's physical simulation results. The accurate model parameter value extraction provides the more explicit description of device characteristics, particularly the core parameters. This drives to analyse and understand of



(a)



(b)

Figure 5.2: The I-V characteristics of n- and p-channel device under different gate voltage condition. The boundary between the linear and saturated regions is indicated by the upward curving parabola (sub-figure (a) NMOS) and downward curving parabola (sub-figure (b) PMOS) [172].

Table 5.2: Various regions of operation for MOSFETs

Type	Region	Boundary	Current
N-channel MOSFET	Cut off	$V_{GS} \leq V_T$	$I_{DS} = 0$
	Linear	$V_{GS} > V_T$, $V_{DS} \leq V_{SG} - V_T$	$I_{DS} = \mu_n C_{ox} \frac{W}{L}[(V_{GS} - V_T)V_{DS} - \frac{V_{DS}^2}{2}](1 + \lambda V_{DS})$
	Saturation	$V_{GS} > V_T$, $V_{DS} > V_{SG} - V_T$	$I_{DS} = \frac{1}{2}\mu_n C_{ox} \frac{W}{L}(V_{GS} - V_T)^2(1 + \lambda V_{DS})$
P-channel MOSFET	Cut off	$V_{SG} \leq \lvert V_T \rvert$	$I_{SD} = 0$
	Linear	$V_{SG} > \lvert V_T \rvert$, $V_{SD} \leq V_{SG} - \lvert V_T \rvert$	$I_{SD} = \mu_p C_{ox} \frac{W}{L}[(V_{SG} - \lvert V_T \rvert)V_{SD} - \frac{V_{SD}^2}{2}](1 + \lambda V_{SD})$
	Saturation	$V_{SG} > \lvert V_T \rvert$, $V_{SD} > V_{SG} - \lvert V_T \rvert$	$I_{SD} = \frac{1}{2}\mu_p C_{ox} \frac{W}{L}(V_{SG} - \lvert V_T \rvert)^2(1 + \lambda V_{SD})$

how these core parameters influence the transistor behaviour under DC analysis such as threshold voltage roll-off, non-uniform doping effect, sub-threshold conduction, mobility reduction due to the vertical field, carrier velocity saturation, drain induced barrier lowering (DIBL) and the parasitic resistance effect [171]. A sensitivity relationship between the BSIM4 parameters and *I-V* curve fitting is listed in Figure 5.3. It is important to know the sensitivity of parameters on the curve fitting target to assist the extraction process and improve optimisation time. For example, the threshold voltage, one of the most important electrical parameter when modelling MOSFETs, represents the onset of significant drain current flow that is a function of a number of parameters [171] such as gate material, gate insulator thickness, channel doping, impurities at silicon-insulator interface. In addition, the threshold voltage is a function of temperature that decreases with an increase in temperature [171]. A complete threshold voltage model equation (shown in Equation 5.2 [173]) is, therefore, described as a set of BSIM4 model parameters. These parameter values determine the accuracy of the description of the device characteristics. All of parameters used in the following equations are listed in Appendix B.

$$\begin{aligned} V_{th} = {} & VTH0 + K1 \cdot [Part1] - K2 \cdot [Part2] + (K3 + K3B \cdot V_{bseff}) \cdot \frac{TOXE}{W_{eff} + W_0} \cdot \Phi_s \\ & - (\frac{1}{2} \cdot [Part3] \cdot (V_b i - \Phi_s)) - \frac{1}{2} \cdot [Part4]) \end{aligned} \tag{5.2}$$

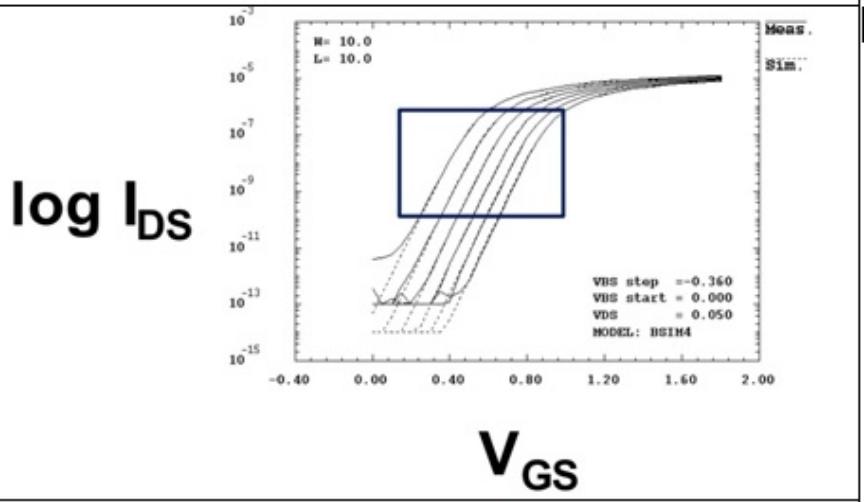
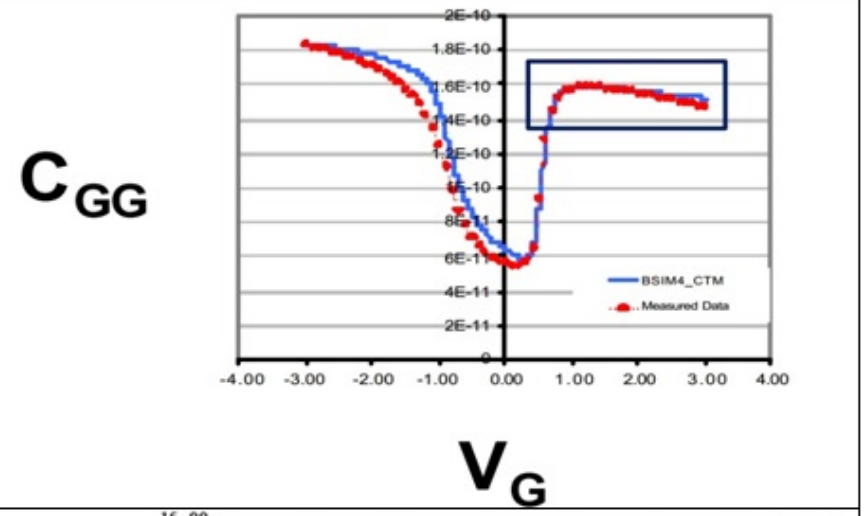
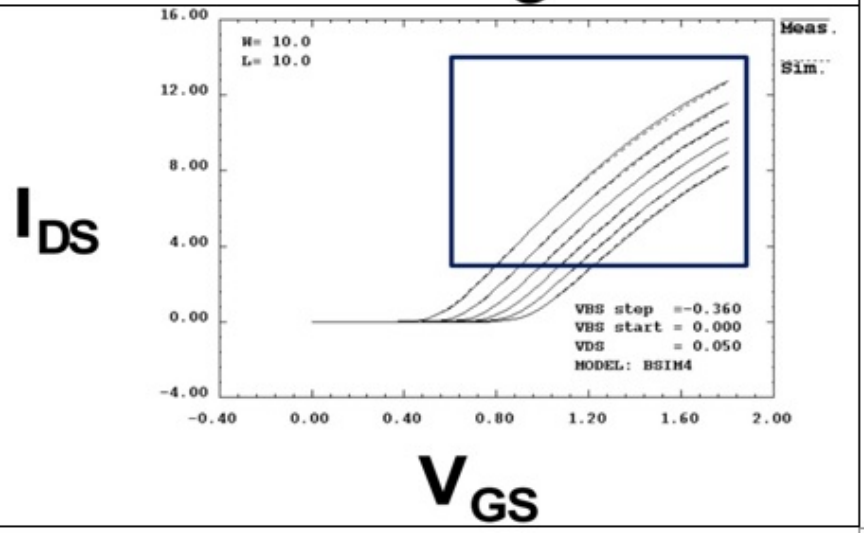
BSIM4 Parameter	Fitting Target
VTH0, K1, K2, DVTP0 ... (Threshold voltage) NFACTOR, VOFF, ... (Subthreshold Slop)	
TOXP, NGATE, (Quantum-mechanical charge layer thickness and poly gate depletion effect)	
U0, UA, UB, UC,(Mobility)	

Figure 5.3: A sensitivity relationship between the BSIM4 parameters and I-V curve fitting target [182]. Different I-V regions are governed by various BSIM4 parameters. For example, the moderate inversion region can be adjusted by controlling parameter, 'MINV'. The drain current in the subthreshold region can be controlled by 'NFACTOR'.

$$\begin{aligned} Part1 = K1 \cdot \sqrt{1 + \frac{LPEB}{L_{eff}}} \cdot (\frac{TOXE}{TOXM} \cdot \sqrt{\Phi_s - V_{bseff}}) \\ + (\frac{TOXE}{TOXM} \cdot \sqrt{\Phi_S} \cdot (\sqrt{1 + \frac{LPE0}{L_{eff}}} - 1)) \end{aligned} \tag{5.3}$$

$$Part2 = \frac{TOXE}{TOXM} \cdot V_{beff} \tag{5.4}$$

$$Part3 = [\frac{DVT0W}{cosh(DVT1W \cdot \frac{L_{eff} \cdot W_{eff}}{l_{tw}})} + \frac{DVT0}{cosh(DVT1 \cdot \frac{L_{eff}}{l_t}) - 1}] \tag{5.5}$$

$$Part4 = \frac{V_ds}{cosh(DSUB) \cdot \frac{L_{eff}}{l_{t0}} - 1} \cdot (ETA0 + EATB \cdot V_{bseff}) \tag{5.6}$$

Besides threshold voltage, the sub-threshold behaviour is also critical to the accuracy of device modelling. The drain current in sub-threshold region is modelled by:

$$I_d = \mu \cdot \frac{W}{L} \cdot \sqrt{\frac{q \cdot \xi_{si} \cdot NDEP}{2 \cdot \Phi_s}} \cdot V_t^2 \cdot exp(\frac{(V_{gs} - V_{th}VOFF(VOFF/L_{eff}))}{n \cdot V_t}) \tag{5.7}$$

$$V_t = \frac{K_B \cdot T}{q} \tag{5.8}$$

The sub-threshold swing parameter, n, is determined by channel length and interface state density, which is calculated using Equation 5.9. The errors in calculating the depletion width capacitance are compensated by parameter *NFACTOR*.

$$\begin{aligned} n = 1 + NFACTOR \cdot \frac{C_{dep}}{C_{oxe}} \\ + [\frac{\frac{1}{2}(VDSV + CDSCD \cdot V_{ds} + CDSCB \cdot V_{bseff})}{cosh(DVT1 \cdot \frac{L_{eff}}{l_t}) - 1}]/C_{oxe} \end{aligned} \tag{5.9}$$

In the drain current model, mobility modelling is significant to the accuracy of a MOSFET model. The carrier mobility in the device channel is lower than that in the bulk substrate because of scattering mechanisms. Mobility of carriers, therefore, depends on many process parameters and bias conditions. Three different models of the effective mobility (*mobMod* = 0, 1, and 2) are provided in BSIM4. The model in this work only gives the unified formulation of mobility on *mobMod* = 0. Appendix B lists threshold voltage modelling, sub-threshold swing modelling, and mobility modelling with the relevant parameters in BSIM4 models.

Additionally, in order to achieve a better fitting of the physical device simulation result, the saturated region is further refined, which are separated into three sub-regions depicted in Figure 5.4 [171]. The device behaviour is governed by three different physical mechanisms

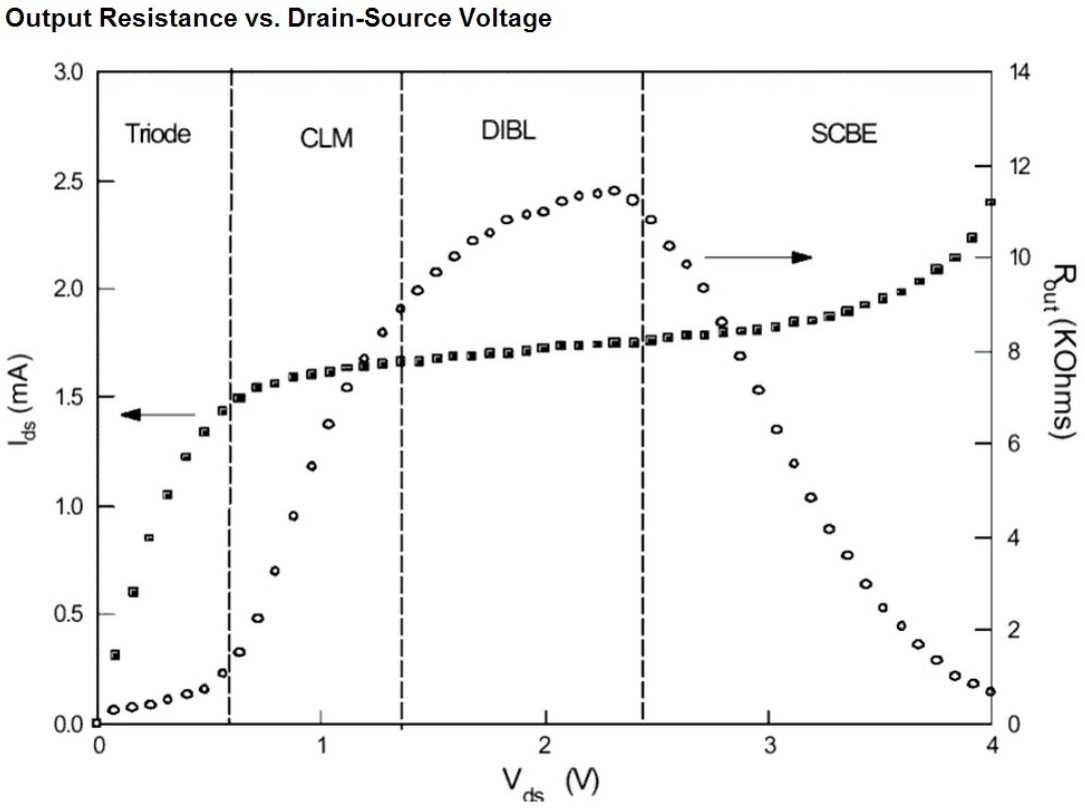


Figure 5.4: General behaviour of MOSFET output resistance [171][173]. The curve marked by rectangle points is drain current *vs.* drain voltage. The curve with cycle points is output resistance *vs.* drain voltage. The device operation region is divided into two main regions, triode (linear) region that the current increase quickly with drain voltage and saturation region that current has a weaker dependence on the drain voltage [171]. The saturation region are further divided several small regions based on channel length moderation (CLM), drain-induced barrier lowering (DIBL) and substrate current induced body effect (SCBE).

and each of those mechanisms dominates the *I-V* curve in a specific place. Those mechanisms are channel length modulation (CLM), drain-induced barrier lowering (DIBL), and substrate-current induced body effect (SCBE). Each of mechanisms are described by the relevant parameters that are listed in Appendix B.

5.1.2 Model Extraction and Optimisation Methodology

Parameter extraction methodology is the key part of model development. Generally, two different extraction strategies are available: group device extraction and single device extraction [171]. In group device extraction, this strategy extracts physical data from a set of devices with different geometries (long channel and short channel) under the same bias conditions. The group device extraction extracts parameters in long channel device, which are independent of short channel effects, and then perform short channel device for parameters related to short channel effects [171]. Thus, this method is beneficial for the group of devices with arbitrary geometries but might not be perfect for a particular device. For single device extraction, experiment data from the single device is applied extract a complete set of model parameters [171]. This method provides an accurate resulting fit with specific geometries; but fitting devices with different geometries are likely not to fit well. In this thesis, single device extraction is used as it is sufficient for motifs compact model extraction requirement due to motifs with fixed geometry size. Thus, single device extraction strategy is used in this study.

Additionally, optimisation strategy is also crucial for extraction results, which treats each model parameters as a "fitting" parameter and gives the minimum average error between measured and simulated data points. However, conventional mathematical model extraction optimisation algorithms have the limitation of a poor convergence without good initial conditions that heavily depends on the designer's expertise and also the difficulty of simultaneous such many parameters optimisation [168]-[169]. The device model usually refers to several hundred *I-V* curve points. The extraction and optimisation is a time consuming task to achieve a set of accurate parameter values with reasonable physical meaning. The advantage of evolutionary algorithms handling complex problems using multi-objective optimisation makes them an interesting candidate to solve this complex parameter extraction and optimisation problem [174]-[177] is discussed in Chapter 4. The following section introduced a novel 2SEA extraction methodology based on EA approach to perform motif compact model extraction.

5.2 Two Step Evolutionary Algorithm Extraction Method

In an attempt to improve the accuracy and efficiency of the model extraction method a Two Step Evolutionary Algorithm (2SEA) with chromosome resizing mechanism, written in the C language, is proposed. The details of this algorithm are introduced as the following.

5.2.1 Overview of Two Step Evolutionary Algorithm Model Extraction Method

The properties of an EA are beneficial for achieving multi-objective optimisation important in this work [178]-[180]. To solve the problem of the traditional method achieving poor convergence without a good initial condition, the two step strategy is proposed to assist in obtaining good initial conditions (step 1), and then further optimising the candidate solutions (step 2). Additionally, when good initial conditions are obtained, it also provides an optimisation guide for later optimisation. As Section 5.1.1 introduced, the device's I-V curve is divided into several regions. Fitting curve target can be trimmed by adjusting chromosome parameter size.

The steps in the process of motifs model extraction using 2SEA is illustrated in Figure 5.5. Originally the initialisation of the algorithm follows two steps but the algorithm can be easily extended to more steps according to experiment requirements. The two steps method demonstrates a technique of general applicability via incremental evolution to obtain reliable motifs compact model extraction results without the need of deeply extraction expertise.

First Step

Since conventional method limits the poor convergence without a good initial value, the aim of the first step of the evolution loop is to find out better initial model parameters value match between the BSIM parameter and the physical device characteristics (from TCAD simulation results) for the later finer extraction processing. These good initial model parameter values will be finely extraction in later step. Before the algorithm loop begins, a set of core model parameters are chosen involving the first step evolution which key parameters relate to threshold voltage, subthreshold swing, mobility, saturation velocity, etc.

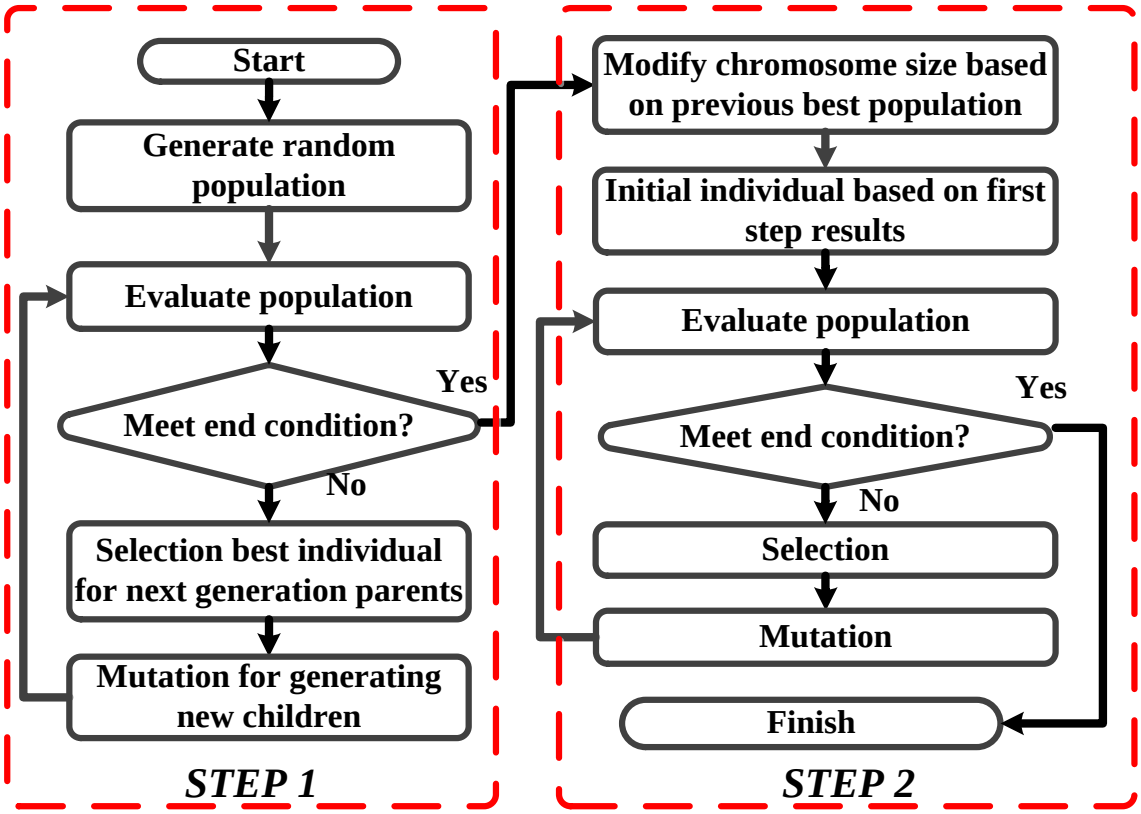


Figure 5.5: The flow chart of 2SEA. Here only displays two steps but the algorithm can be easily extending to more steps. At the beginning of each step optimisation except first step, the chromosome parameter size will be adjusted based on previous step optimisation results. The aim of step 1 is to find a good initial condition for later steps so as to the drawback of the conventional method achieving poor convergence without good initial condition. The step 2 is a finer extraction and optimisation process according to the step 1 extraction results.

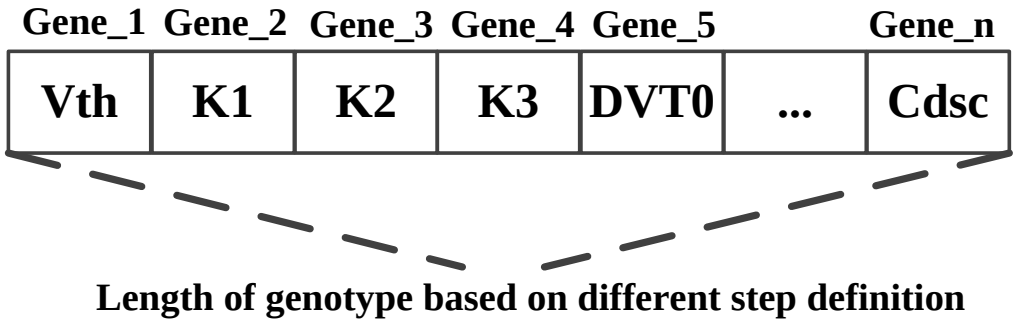


Figure 5.6: The outline of the genotype structure. Each gene represents one optimised BSIM model parameters and each gene is a floating-point value. Although the length of genotype is fixed in every step, the different steps have the variety of length of genotype due to optimisation direction changing.

Second Step

The next steps of evolution loop are a finer model extraction based on first step extraction result that purposefully optimises the compact model to match physical device simulation results. In this step, some of new model parameters are added into the algorithm to finely optimise the mismatch between BSIM compact model parameter and the physical device characteristics. Although the first step and second step optimisation perform the same EA processing to extract the model, the difference between first step and second step is that the second step optimisation purposely adjusts optimised parameters to further reduce the existed mismatch in first step optimisation results. Thus, what these new parameters are selected and optimised is decided by the first step extraction result. The first generation population in the second step optimisation depends on the previous step results. In additional, although two step extractions are applied in this thesis, this increment algorithm can be extended many steps until the satisfied result found.

5.2.2 Representation

The genotype in the proposed algorithm employs an array consisting of floating-point numbers to encode genotype. Each floating-point number corresponds to a BSIM4 model parameter. Figure 5.6 illustrates the outline of the genotype structure. The design of the genotype encoding strategy strongly depends on the properties of motifs. This means that one genotype needs at least 100 floating point numbers to characterize the BSIM4 model. If all the pa-

rameters are encoded in the genotype, there is no doubt that the huge search space puts a heavy burden on the algorithm, leading to a reduction of search efficiency. Consequently, only dominant core parameters are employed to build genotype in these experiments. In order to avoid potential numerical problems, warnings or fatal errors being reported from SPICE simulation due to parameter falling outside the range limitations, the BSIM model parameter is restricted to specific ranges [181]. This means that each gene in the genotype corresponding to BSIM parameter generated by 2SEA is also restricted to a specific range. Since each parameter in the BSIM model has a different unit and range, the normalized parameters are chosen on the interval $[0, 1]$, and are then mapped to their respective fields. This gene generation mechanism is calculated by the follow Equation 5.10:

$$P_n = P_{nmin} + (P_{nmax} - P_{nmin}) \cdot \beta \tag{5.10}$$

Where β is a random floating-point number in the interval $[0, 1]$, and P_{nmax} and P_{nmin} are the maximum and minimum value of n-th parameter respectively.

5.2.3 Selection Process

All individuals are evaluated for breeding and the whole population is sorted according to fitness. In the selection operation, an offspring always has a higher selection priority than the parents when both of them have same fitness value. To inherit elite solutions from the previous generation to the next generation, the individual with the best fitness value is directly copied into the next generation, with each producing λ offspring, which carry on the mutation process, described follow section. A $(1 + \lambda)$ strategy is used in this experiment and illustrates how the 2SEA is used to extract model parameters. The algorithm can easily be extended to a $(\mu + \lambda)$ strategy, where μ is the number of parents and λ is the number of children.

5.2.4 Mutation Process

Beginning with an initial parent generation in the each evolution iteration, a child is produced by randomly modifying the parent gene values [182]. Owing to candidate genotype is the floating-point representation and the allele value of each gene value is located in its specific

domain defined by a lower L_m and upper U_m bound, resulting in the following transformation:

$$< x_1, x_2, \ldots, x_m > \rightarrow < x'_1, x'_2, \ldots, x'_m >, where\ x_m, x'_m \in [L_m, U_m]. \quad (5.11)$$

In general, uniform and non-uniform mutation are available for floating-point representation. Both of them reproduce the new gene values according to the probability distribution. The uniform mutation is the most straightforward way to perform float-point encodings, analogous to bit-flipping for binary representation and the random resetting sketched above for integer genes. Thus, single-gene uniform mutation is used in this experiment [99], which is described by Equation 5.12.

$$< x_1, x_2, \ldots, x_n, \ldots, x_m > \rightarrow < x_1, x_2, \ldots, x'_n, \ldots, x_m >, where\ x_n, x'_n \in [L_n, U_n]. \quad (5.12)$$

The new gene values of x'_n are generated uniformly randomly from [L_n, U_n]. This ensures the new gene values will not be beyond their range limitation and hence avoids fatal errors or warning reported from simulation due to improper model parameter values.

5.2.5 Motif Netlist Simulation

The steps in the process of candidate solutions evaluation through SPICE simulation are illustrated in Figure 5.7. The evaluation process follows a number of steps: The first step forms motifs BSIM model card, in which a number of important model parameters are encoded and evolved by genotype. These optimised parameters within the genotype are inserted into a predefined partial motif netlist that forms a complete motif BSIM model card for each individual. An example of the complete reference device netlist is shown in Appendix C. The following steps are the most time consuming, involving the use of the NGSPICE simulator [183] performing motif model cards simulation, and the post-processing of the SPICE output data. NGSPICE is called successively for a number of motif netlist for every individual in population, with the output data processed conveniently for fitness calculation. The detailed fitness calculation is introduced in the following.

5.2.6 Fitness Evaluation

In this case, the difference between the TCAD measured curve and the 2SEA generated curve from SPICE simulation is the essence of this problem, which intuitively reflects the error of extracted parameters close to the TCAD measured data. Figure 5.8 shows the outline of

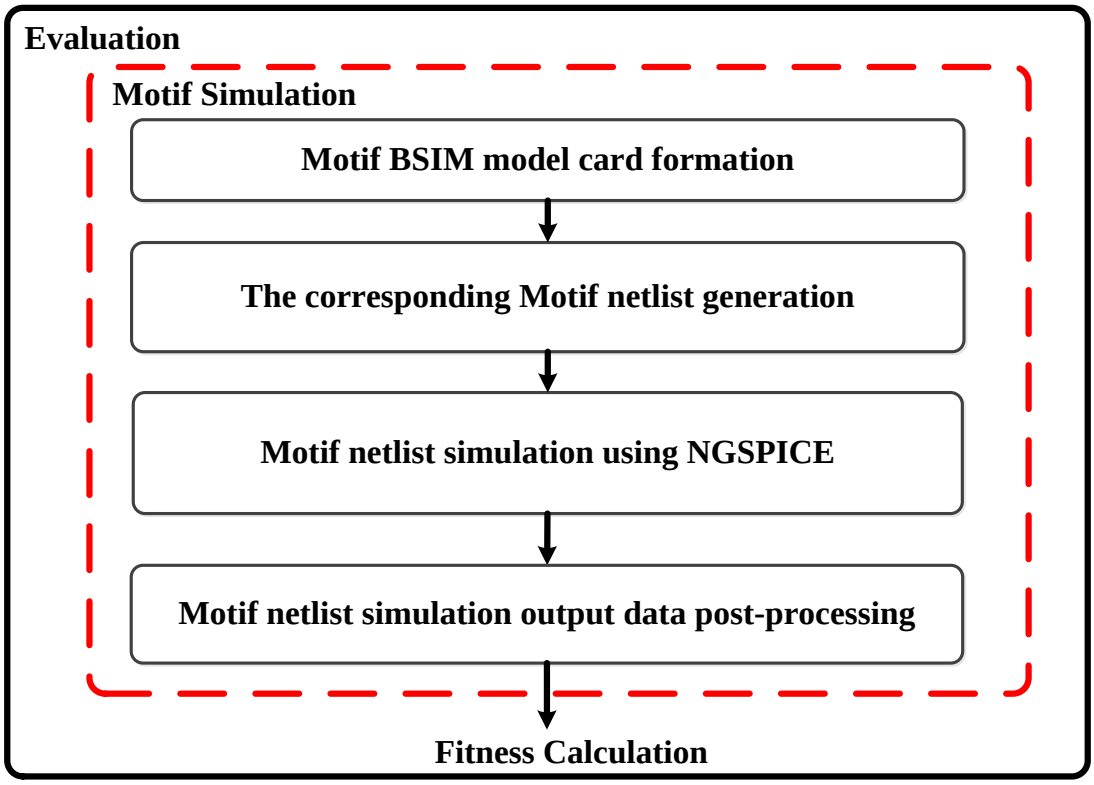


Figure 5.7: The steps in the process of candidate solutions evaluation

evaluation in the case of I-V curves. In general, root means square (RMS) is used for evaluation in such problems. However, RMS makes the fitting accuracy reduce drastically when the two evaluated values have different magnitude levels [181]. Drain current (I_d), especially, has no significant difference when gate voltage (V_g) is at low bias condition, but I_d is increased significantly as V_g becomes large [181]. Instead of RMS, the sum of absolute errors is used for evaluation. The sum of absolute errors between measured data and 2SEA generated data is calculated using the following Equation 5.13:

$$fitness = \sum_{m} \left| \frac{P_{2SEA}(m) - P_{TCAD}(m)}{P_{TCAD}(m)} \right| \tag{5.13}$$

Where m means the m-th measure point on the curve line. The actual drain current value at the m-th point is given by $P_{TCAD}(m)$. $P_{2SEA}(m)$ represents the simulated drain current value at the m-th point. In addition, model parameters extraction involves 10 I-V curve lines based on different V_d bias condition, each line has at 13 sampling point.

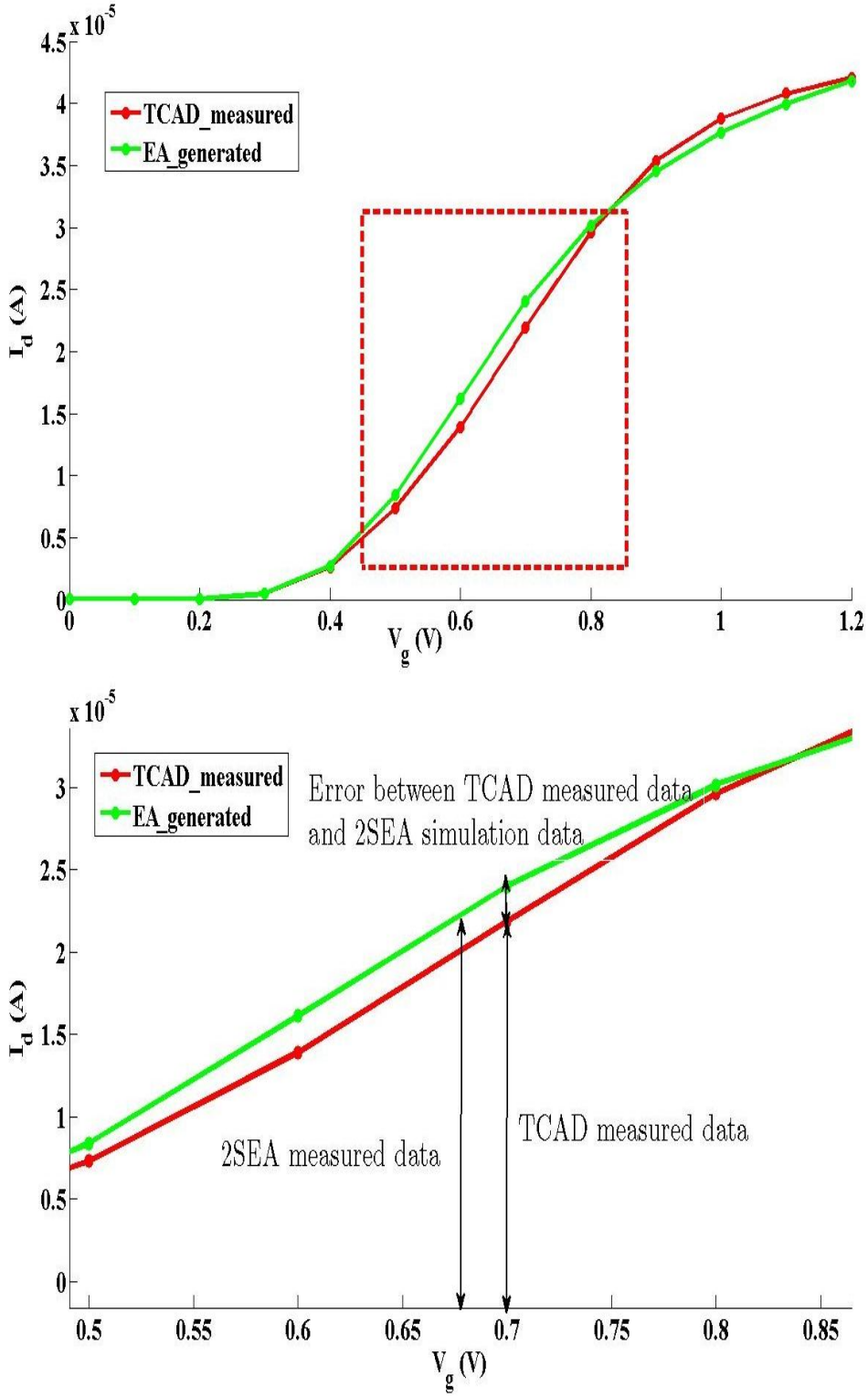


Figure 5.8: Fitness calculation using error rate. Two points with the same gate voltage are taken from TCAD measured curves and EA candidate generated curves based on SPICE simulation, respectively. Then, the drain current absolute error of both points are calculated as fitness. In the fitness calculation, 12 sample point with different gate voltages are applied.

5.3 Experiment Results and Analysis

5.3.1 Experiment Setting-up

In this experiment, the extraction is processed on an Intel i5 (4 cores), 8G RAM and Windows OS machine. Measured data of the motif was obtained from TCAD *50 nm* device simulation result. Before extracting TCAD measured data of a device, a known SPICE model is extracted in a control experiment in order to verify that the methodology of 2SEA algorithm is effective. Based on extraction result, the error between 2SEA extracted model and SPICE reference model result is less than 1%. This validates that the 2SEA can be seen as an effective method for device motifs extraction. In order to explore characteristic of O shape device motifs and reference device, these models for each were extracted as illustrated in the following.

5.3.2 Reference Device Extraction

Before extraction begins, a portion of parameters in compact model need no fitting due to them being directly obtained from experimentally measured or process simulated structural values, typically parameters related to gate oxide thickness, doping concentration in the channel, junction depth, and dielectric constant. These prerequisite parameters are listed in Table 5.3. Basic device parameters are extracted first, such as threshold voltage parameters *Vth0*, *k1*, *k2*, sub-threshold region parameters, *nfactor*, *voff*, mobility parameters, *U0*, *Ua*, *Ub*, short channel effect parameters, *dvt0*, dvt1, *dvt2*, and so on. In the second step, a portion of other parameters, based upon the first step extraction result, are added into genotype in order to further extract and optimise model parameter. The NMOS and PMOS reference

Table 5.3: Prerequisite process parameters prior to extraction process

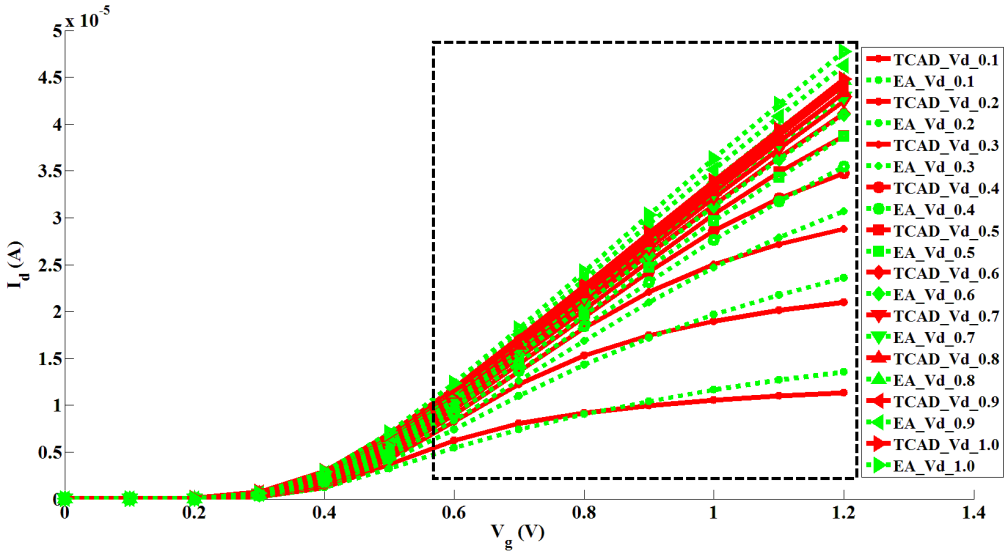
Parameter	Physical Meaning	Value
T_{ox}	Gate oxide thickness	2 [nm]
N_a	Body doping	2e18 [cm-3]
N_d_po	Polygate Doping	1e20 [cm-3]
N_d_sd	Source/Drain Doping	1e20 [cm-3]
N_d_ex	Source/Drain Extension Doping	2e19 [cm-3]
X_j_sd	Source/Drain Junction Depth	0.1 [μm]
X_j_ex	Source/Drain Extension Junction Depth	0.02 [μm]

Table 5.4: Two Step EA Parameters (Reference Deivce)

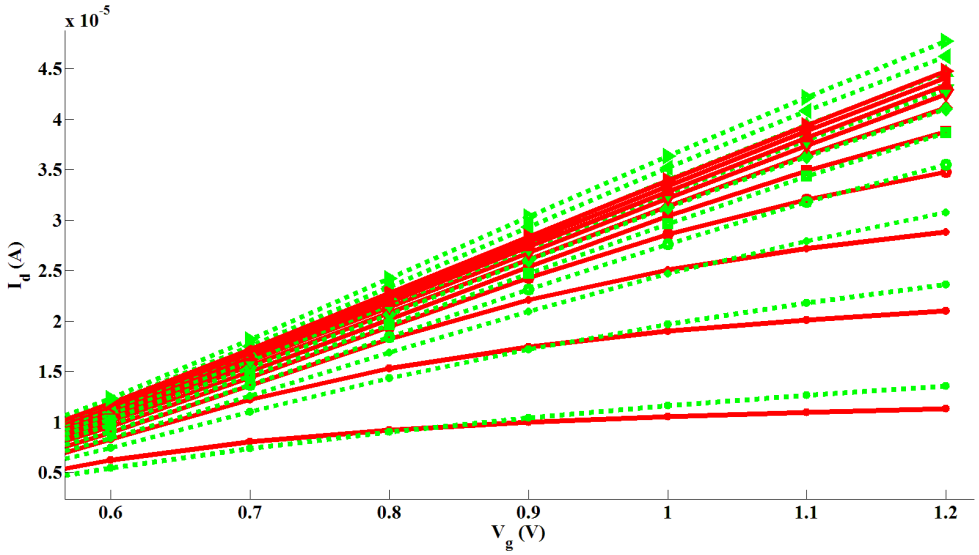
EA Parameter	Value
Step	1
Generation number	10,000
Population size	5 (1 + 4)
Mutation rate	1/24 (Mutation rate = 1/ n, n is number of genes in genotype)
Genotype size	24 (Vth, k1, k2, k3, dvt0, dvt1, dvt2, dvt0w, dvt1w, dvt2w, u0, ua, ub, vsat, a0, a1, a2, keta, voff, eta0, etab, nfactor, cit, cdsc)
Step	2
Generation number	10,000
Population size	5 (1 + 4)
Mutation rate	1/36 (Mutation rate = 1/ n, n is number of genes in genotype)
Genotype size	36 (Vth, k1, k2, k3, dvt0, dvt1, dvt2, dvt0w, dvt1w, dvt2w, u0, ua, ub, vsat, a0, a1, a2, keta, voff, eta0, etab, nfactor, cit, cdsc, pdits, pditsd, pdiblc1, pdiblc2, drout, pclm, pdiblcb, fprout, delta, minv, rdsw, dsub)

device with fixed geometry size (*L=50 nm*, and *W=80 nm*) are extracted by 2SEA. The evolutionary parameters are listed in Table 5.4. Both types of reference device extraction procedure are separately executed 10 times. The worst final generation fitness value for NMOS device extraction at the second step optimisation is 14.14% and the best fitness value is 6.4%. Low fitness values are good in this case. Figure 5.9 shows that the best solution of ES generated simulation curves in 10 runs (green dash lines) on the first step extraction. The sum of errors of all curves between TCAD measured curves and 2SEA generated simulation curves in the first step extraction is 9.02%. As can be seen from Figure 5.9, 2SEA generated simulation curves fit well with TCAD measured curve at low drain region and have less matching at saturation region. This result suggests the need to add relevant saturation region parameters on the second step optimisation to achieve better fitting at saturation region. In the second step optimisation, the number of genes in a chromosome is raised to 36 from 24 genes. The second step optimisation result is illustrated in Figure 5.10. After the second step optimisation, the sum of error of all curves shown in Figure 5.10 reduces to 6.4%. PMOS reference device extraction has similar situation that needs to adjust the number of optimisation parameters to achieve better extraction results. The final worst fitness value of

PMOS model extraction is 12%, and best value is 6.2%. Figure 5.11 and 5.12 shows the first step extraction results and second step extraction results, respectively.



(a)

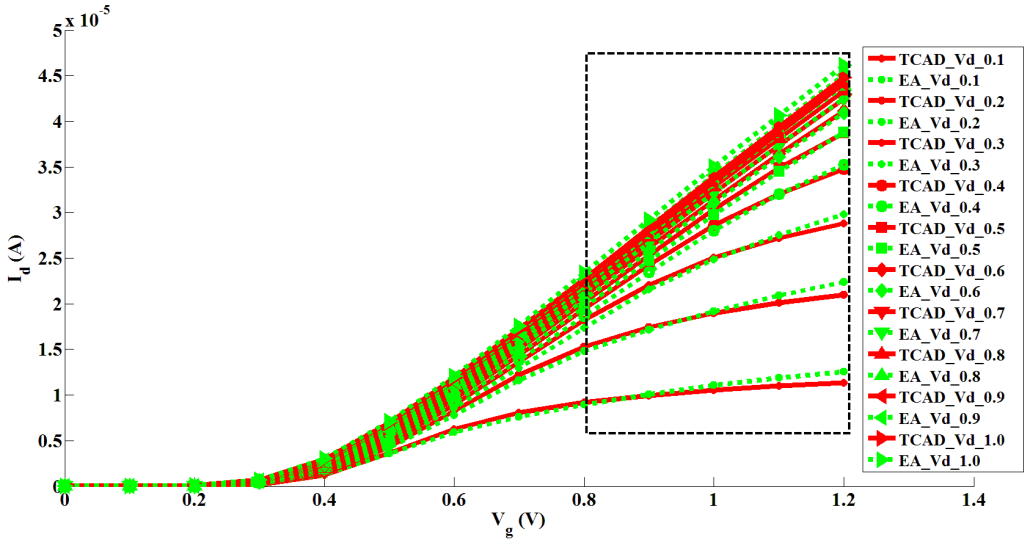


(b)

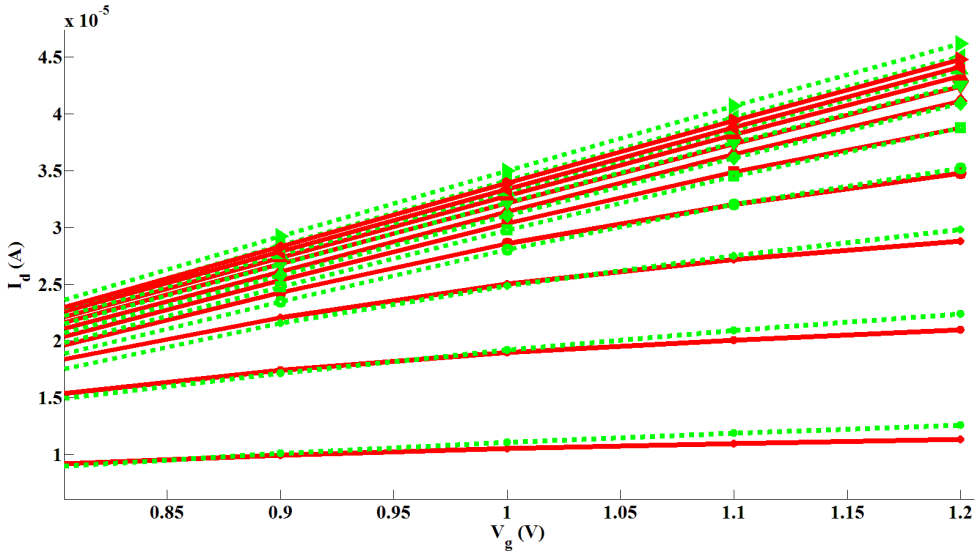
Figure 5.9: The comparison between NMOS reference device TCAD measured curves (red) and ES generated simulation curves (green/dash) for the best solution in 10 runs on the first step extraction. Two curves fit well in the triode region and have less match in the saturation region (shown in bottom sub-figure). Suggesting that the optimisation parameter size for the second step extraction require adjustment.

5.3.3 *O* Shaped Device Extraction

In a similar manner, the measured data of the *O* shaped motif (*L=50 nm* and *W=160 nm*)



(a)



(b)

Figure 5.10: The comparison between NMOS reference device TCAD measured curves (red) and ES generated simulation curves (green/dash) for the best solution in 10 runs on the second step extraction. After the second step extraction, two curves in the saturation region also have better match. The bottom sub-figure zooms the optimised region.

also came from 3D TCAD simulation. The evolutionary parameters for the O shaped motif are listed in Table 5.5. To help with convergence, the generation number in this experiment was enlarged from 10,000 to 20,000, simultaneously; the population size is also extended to 20, $(1 + 19)$. The O shaped motif extraction was executed 10 runs. The worst solution of NMOS O shaped motif error is 11.19% and the best solution error is 6.22%. Figure 5.13 shows the best NMOS O shaped motif extracted result in 10 runs after first step extraction. The sum of error between TCAD measured data of O shaped motif and evolutionary generated data is

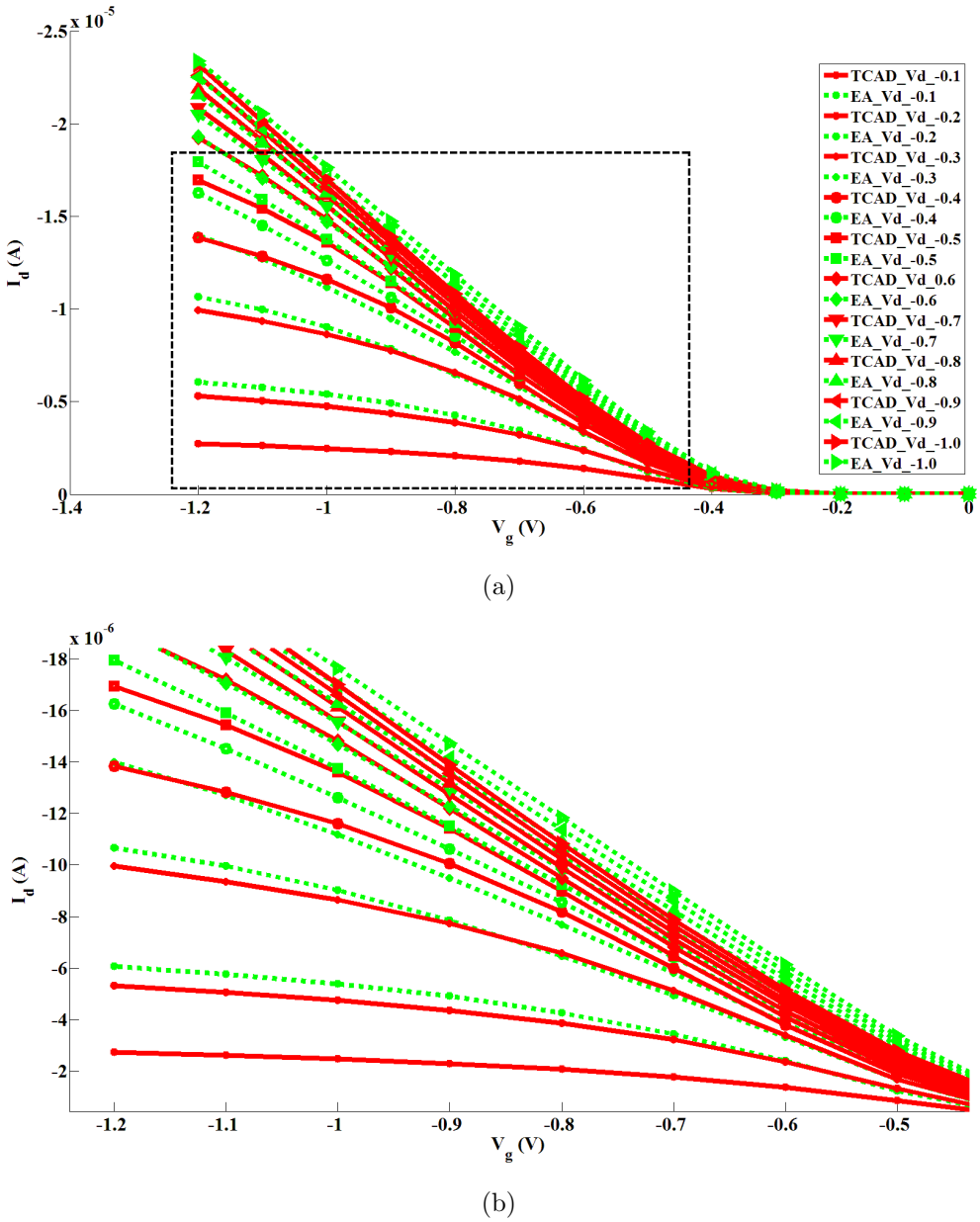
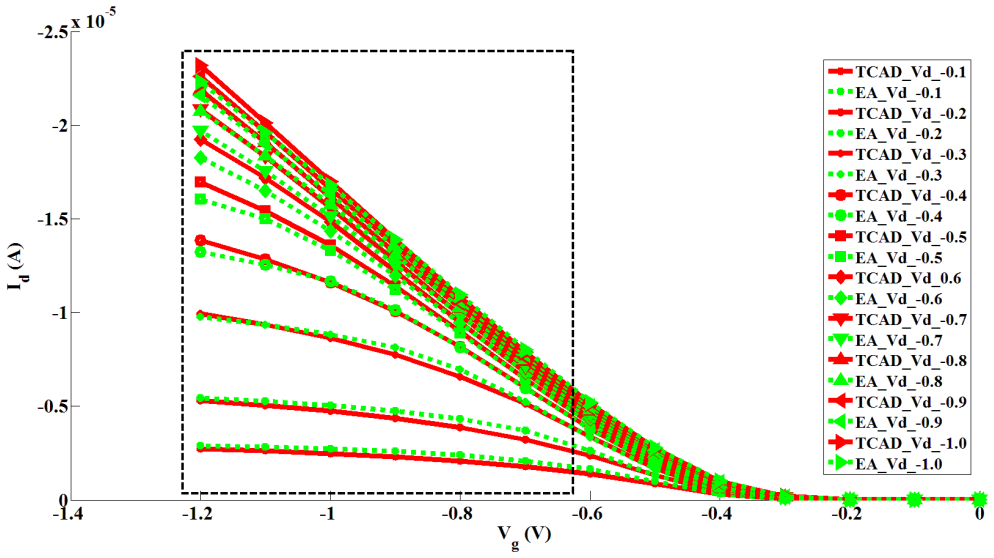


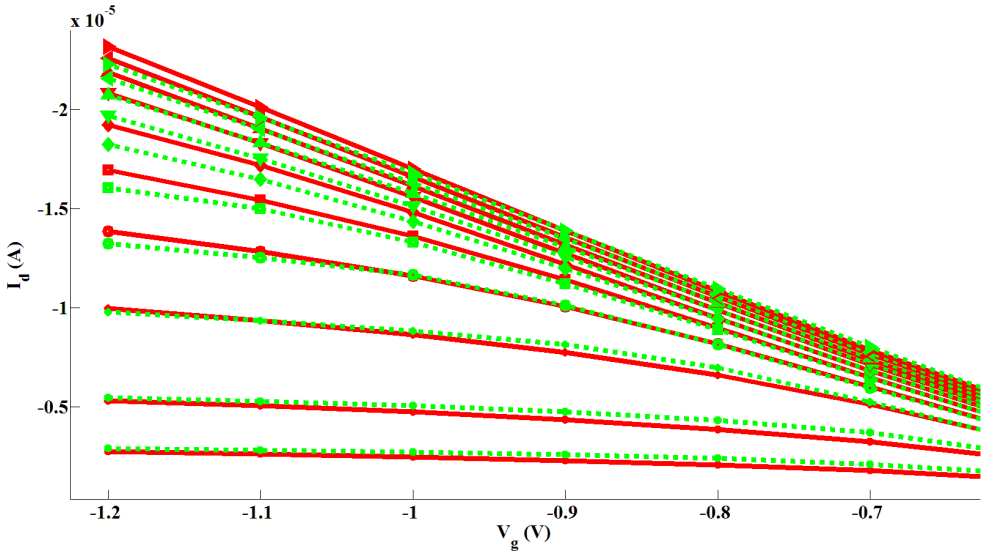
Figure 5.11: The comparison between PMOS reference device TCAD measured curves (red) and ES generated simulation curves (green/dash) for the best solution in 10 runs on the first step extraction. Larger error region is zoomed in the bottom sub-figure.

13.76%. This has similar properties to the reference device, in that it has better curve fitting in the triode region and mismatch in the saturation region, especially, curves with low drain bias. The second step extraction results show better accuracy between the TCAD simulated results and BSIM evolutionary extraction results (depicted in Figure 5.14), where the fitness value finally converges to 6.22 %. Figures 5.15 and 5.16 show the PMOS O shaped motif extraction results that the final worst fitness value is 10%, and best value is 6% in 10 runs.

In summary, the first step reference device and O shaped motif extraction results illustrate



(a)



(b)

Figure 5.12: The comparison between PMOS reference device TCAD measured curves (red) and ES generated simulation curves (green) for the best solution in 10 runs on the second step extraction. After the second step extraction, two curves in the saturation region also have better match. The bottom sub-figure zooms the optimised region.

that proposed increment algorithm can actually find the better initial model parameters value that provide a good start for next step extraction. The mismatch region is shown in the first step results, for example, mismatch part is local at saturation region. Based on this analysis some of new parameters related to saturation region are considered and optimised in the second step optimisation. Comparing the second step optimisation results and first step extraction results, the mismatch region in I-V curve in first step results is actually improved through second step finely optimisation. Although the final reference device and

Table 5.5: Two Step EA Parameters (O Shaped Motif)

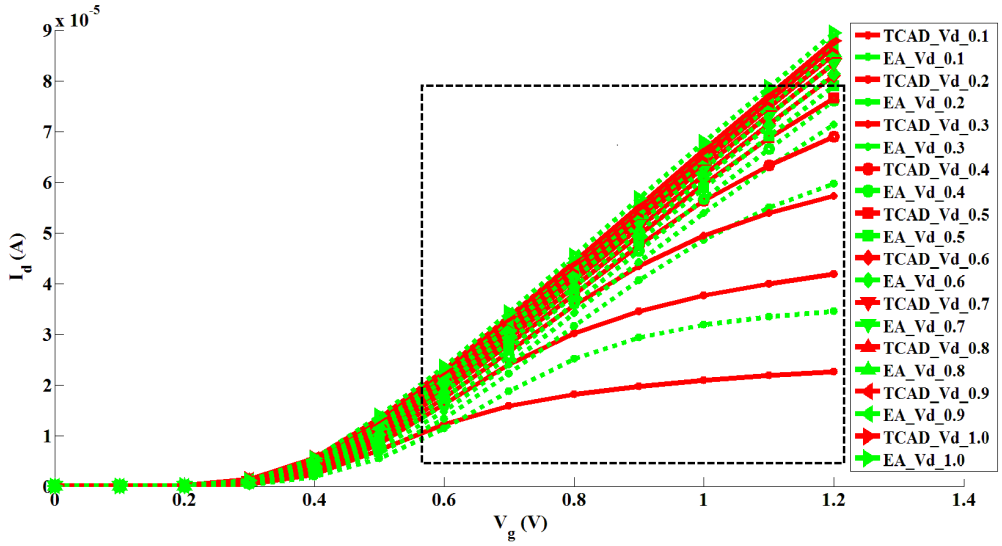
EA Parameter	Value
Step	1
Generation number	20,000
Population size	20 (1 + 19)
Mutation rate	1/24 (Mutation rate = 1/ n, n is number of genes in genotype)
Genotype size	24 (Vth, k1, k2, k3, dvt0, dvt1, dvt2, dvt0w, dvt1w, dvt2w, u0, ua, ub, vsat, a0, a1, a2, keta, voff, eta0, etab, nfactor, cit, cdsc)
Step	2
Generation number	20,000
Population size	20 (1 + 19)
Mutation rate	1/36 (Mutation rate = 1/ n, n is number of genes in genotype)
Genotype size	36 (Vth, k1, k2, k3, dvt0, dvt1, dvt2, dvt0w, dvt1w, dvt2w, u0, ua, ub, vsat, a0, a1, a2, keta, voff, eta0, etab, nfactor, cit, cdsc, pdits, pditsd, pdiblc1, pdiblc2, drout, pclm, pdiblcb, fprout, delta, minv, rdsw, dsub)

O shaped motif extraction results (exist about 6% error to the TCAD simulation result) is not very perfect than model extracted by industry extraction tool (about 3% error [29]), only 36 parameters are optimised in this experiment. When the more parameters are optimised, the accuracy of final extraction result will further increase. In addition, although extracted model has about 6% error, this level accuracy is sufficient for the experiments in this thesis.

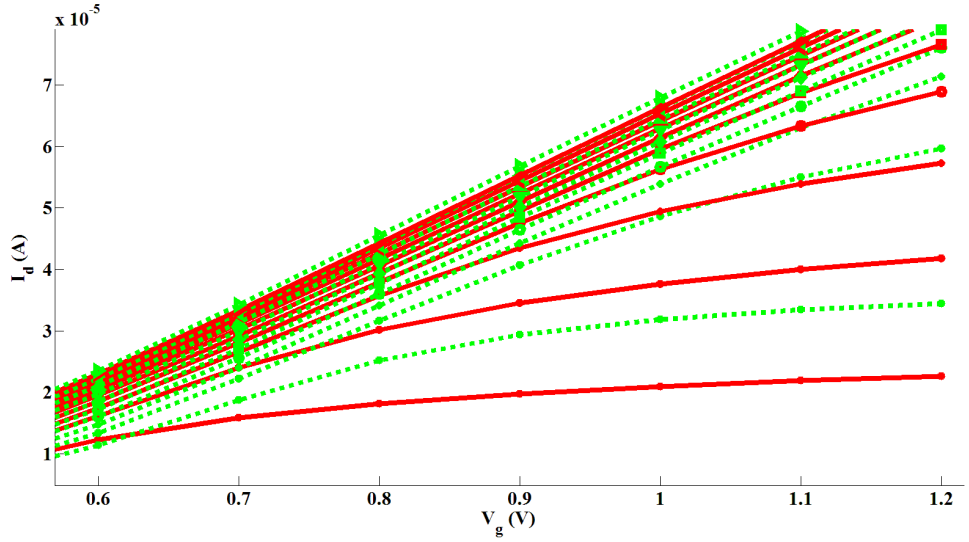
5.4 Statistical Device Layout Motif Compact Model Modelling

5.4.1 Direct Statistical Compact Modelling methodology

The intrinsic variability of electrical properties of aggressively scaled MOSFET has become a significant challenge in integrated circuit design. Statistical compact model act as a bridge between IC designer and foundry, which is an effective way to communicate statistical variability information to designers so that designers can mitigate the impact of device parameter fluctuations and improve circuit performance and stability [184]. Thus, statistical variability



(a)

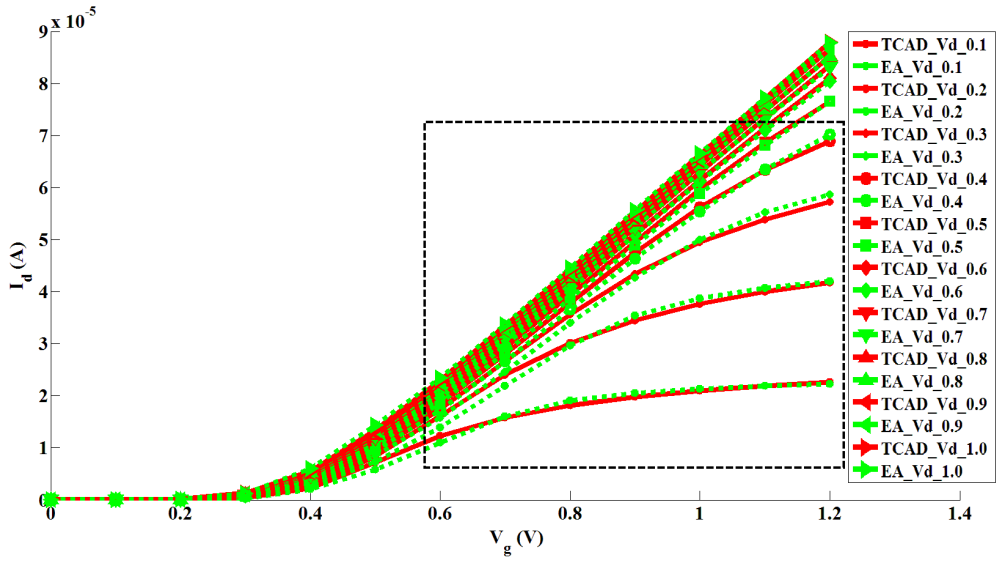


(b)

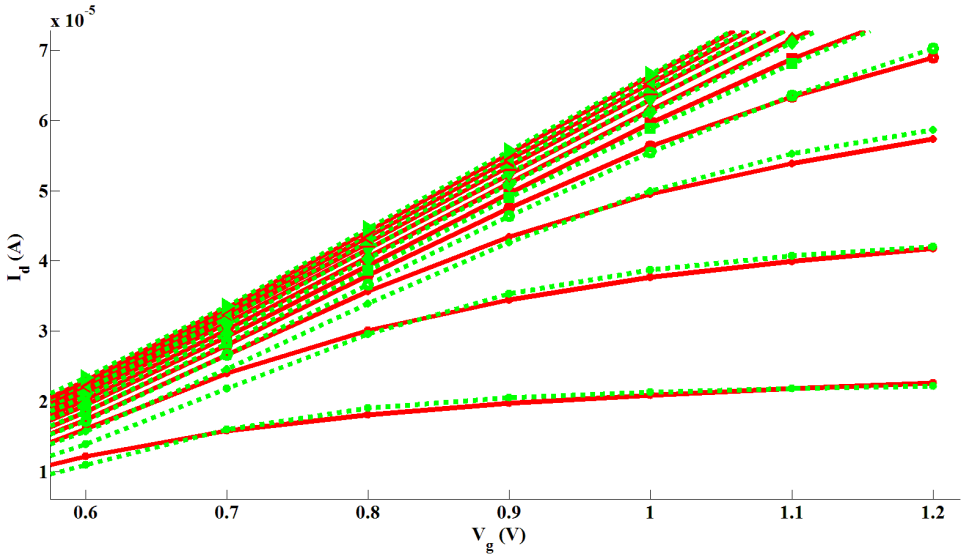
Figure 5.13: The comparison between NMOS *O* shaped motif TCAD measured curves (red) and ES generated simulation curves (green/dash) for the best solution in 10 runs on the first step extraction. Larger error region is zoomed in the bottom sub-figure.

models become a key factor for variation-aware design. Generally, two typically statistical parameter modelling approaches are used by designers: The first approach is so called "naïve approach" that the variation of statistical compact model parameter is generated base upon assuming independent normal (Gaussian) distribution for each extracted parameter [185]. The second method preserves the correlation between extracted parameters based on principle component analysis (PCA) that is called "PCA approach" [185].

Two industry-standard compact models, threshold-voltage-based BSIM4 and surface-potential-based (PSP) model, are provided to produce the statistical variability model. Due to motifs



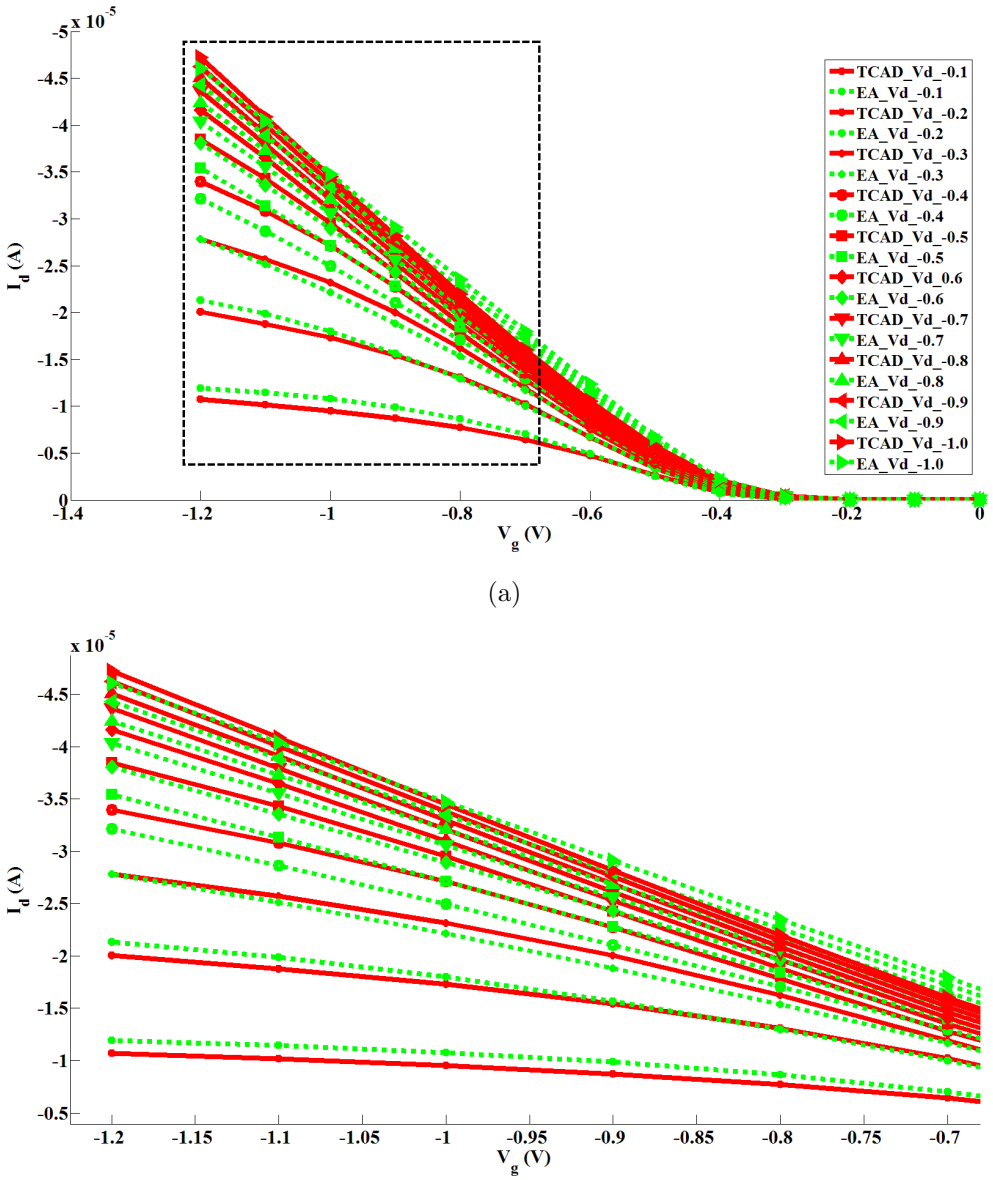
(a)



(b)

Figure 5.14: The comparison between NMOS O shaped motif TCAD measured curves (red) and ES generated simulation curves (green/dash) for the best solution in 10 runs on the second step extraction. The bottom sub-figure zooms the optimised region.

compact model extraction based on BSIM model, the corresponding statistical variability models are also built upon BSIM model. Chapter 3 discussed the simulated I_d *vs.* V_g characteristics of 300 microscopically different motifs under the influence of the LER sources, the I-V curves have significantly spread, indicating that statistical variability strongly impacts the device operation. In this case, the naïve approach is applied to produce the statistical variability compact model that adapts several variability parameters, including threshold voltage, offset voltage, low-field mobility, to produce a set of variation values following a Gaussian

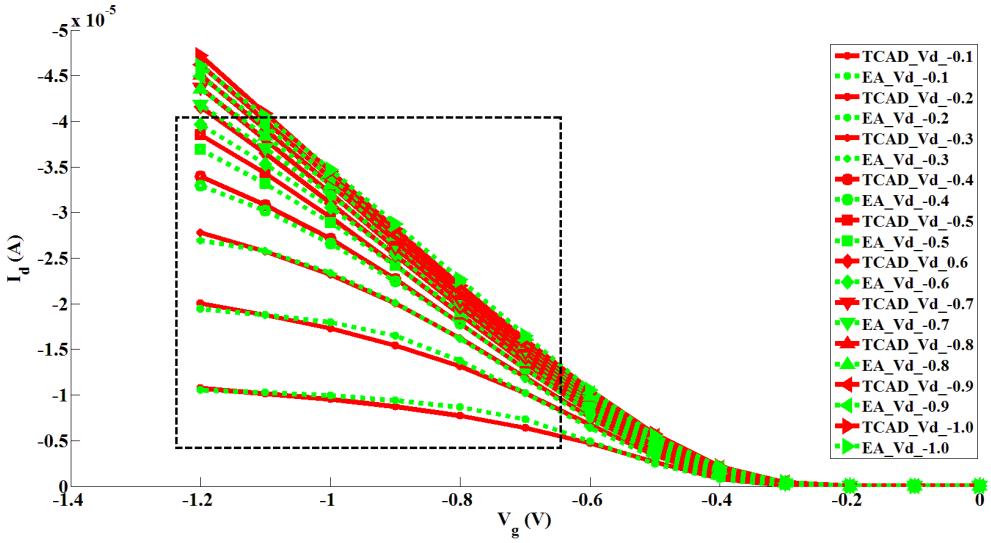


(a)

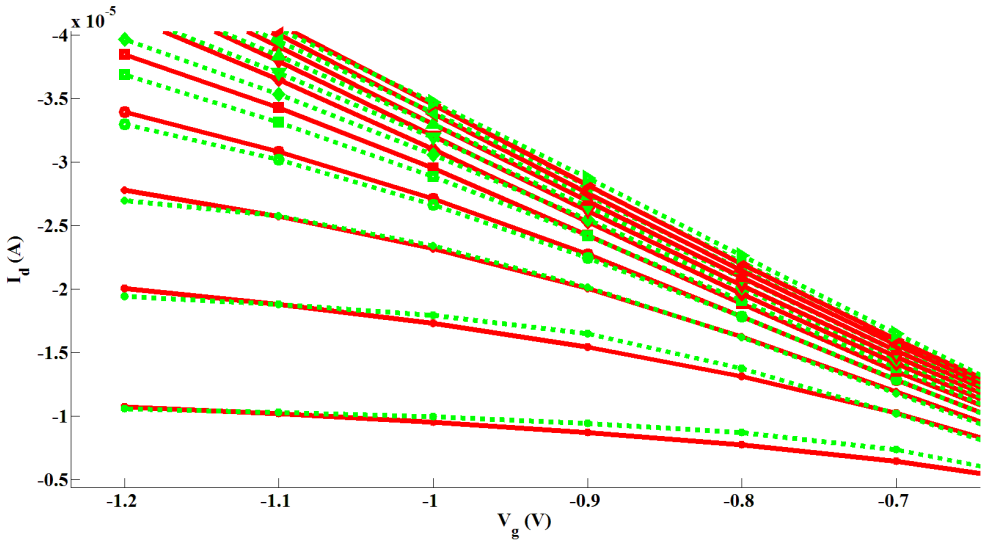
(b)

Figure 5.15: The comparison between PMOS *O* shaped motif TCAD measured curves (red) and ES generated simulation curves (green/dash) for the best solution in 10 runs on the first step extraction. Larger error region is zoomed in the bottom sub-figure.

distribution and injects these variations into a uniform model that has no variability effects [186][187]. As discussed in [188], a fully accurate seven parameter 50000 card models are used for the real yield prediction on industrial problems. For the sake of the simplicity and saving computational time, only 1000 card models with one simple variation parameter, threshold voltage, are used to capture the intrinsic variation information here. The device uniform model can be obtained through 2SEA extraction method. A variation value with Gaussian distribution is randomly generated and added to mean value of threshold voltage and directly



(a)



(b)

Figure 5.16: The comparison between PMOS *O* shaped motif TCAD measured curves (red) and ES generated simulation curves (green/dash) for the best solution in 10 runs on the second step extraction. The bottom sub-figure zooms the optimised region.

injected into the uniform BSIM model enables variability to be included in the motifs. The threshold voltage Gaussian distribution parameter can be directly captured from the TCAD device simulation results. Figure 5.17 shows the flow chart of motifs statistical variability model generation.

5.4.2 Statistical Compact Modelling Results and Verification

Figure 5.18 shows that an example of statistical NMOS reference device *I-V* curves present-

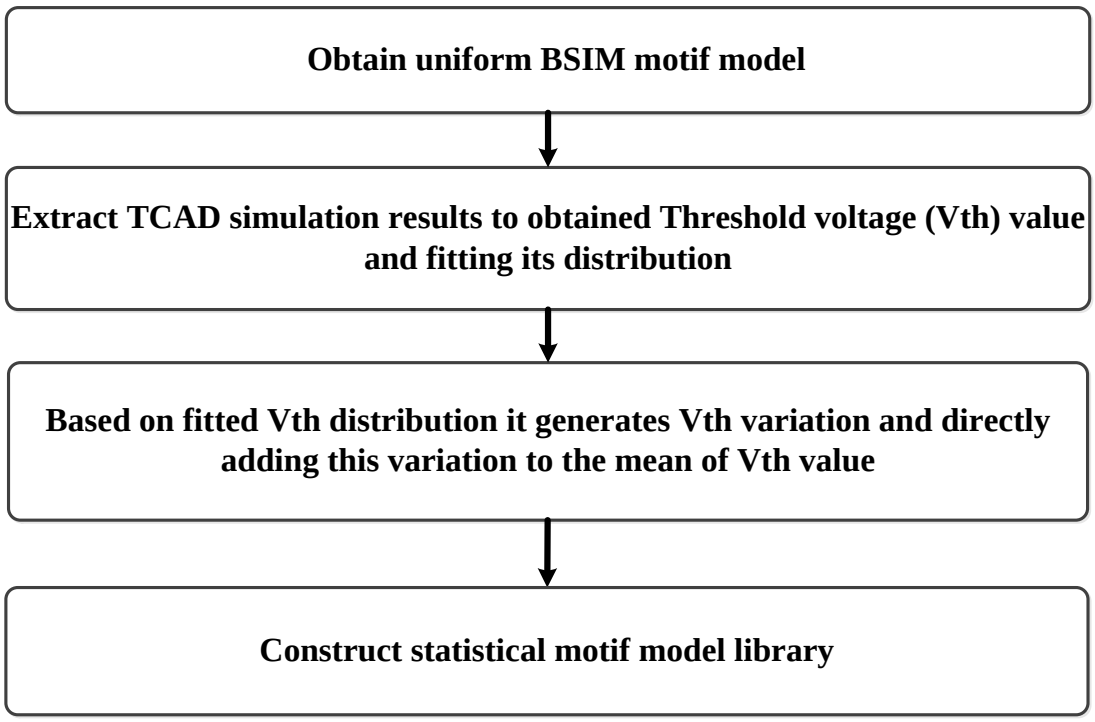


Figure 5.17: The flow chart of motifs statistical variability model generation. In this case, only one parameter, threshold voltage, is used to generate variation. However, the described method is a generic approach that can also be adapted to other parameters.

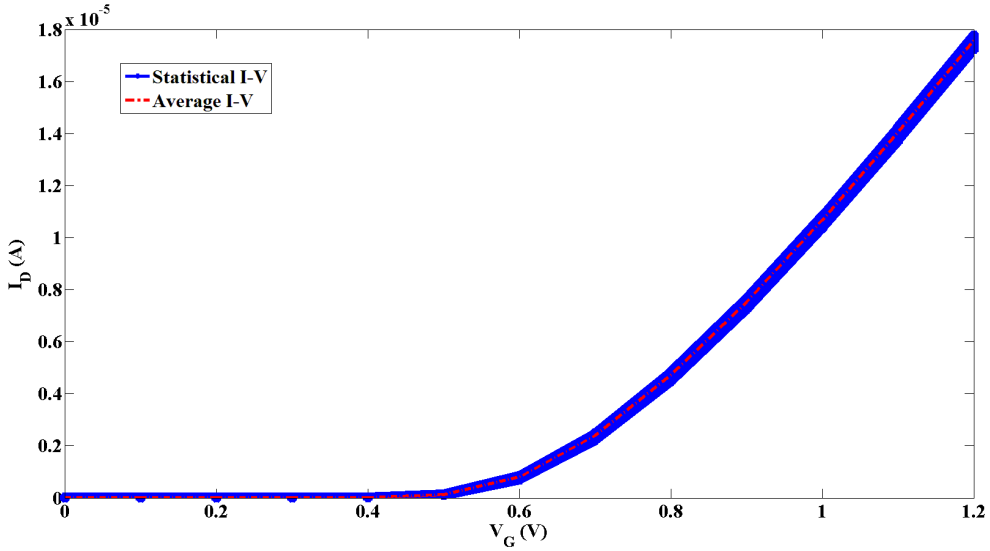


Figure 5.18: Large ensembles of NMOS reference device I-V curves presenting different instances with LER variability effect. These curves are generated by proposed motifs statistical variability modelling approach.

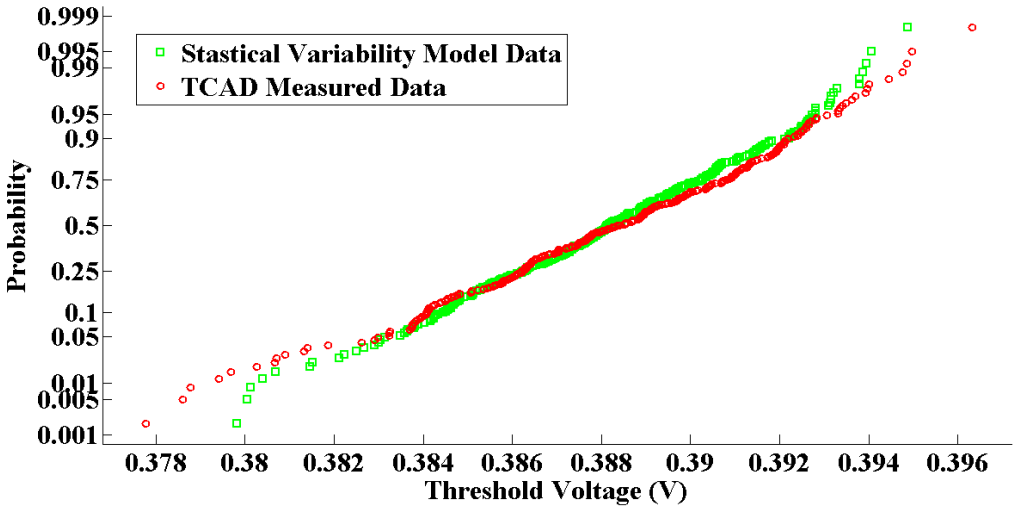


Figure 5.19: The probability plots of typical threshold voltage in compact model, which clearly demonstrated that approximation between the TCAD measured data and data produced by the above mentioned approach is appropriate.

ing different instances with LER variability effect is generated by the previously described statistical modelling method. In order to verify how the statistical variability model can fit well with reality, threshold voltage produced by the naïve approach is compared to TCAD simulation results. Figure 5.19 shows the probability plots of typical threshold voltage in compact model, which clearly demonstrated that the TCAD measured data and data produced by the above mentioned approach are appropriate. These statistical variability models are robust and promising. This approach can be widely used to generate other motifs statistical variability model.

5.5 Summary

In this chapter, an overview of equations with many parameters used in BSIM4 compact model are discussed, which include related threshold voltage parameters, sub-threshold region parameters and source/drain resistance. In order to link between motifs (described in Chapter 3) and the use of motif performing circuit optimisation (described in Chapter 6), the uniform motif compact model is extracted by the proposed 2SEA method that the detailed and principle of this extraction algorithm is given in this chapter. In order to explore the influence of motifs on circuit performance variability, a statistical motif compact model library is established through introduced statistical modelling methodology. The comparison of distribution of TCAD measured data and data generated from the proposed statistical mo-

tifs model indicates the statistical variability model using the proposed statistical modelling approach is robust and reliable. The following chapter describes an important methodology that combines motifs and evolutionary algorithm to achieve circuit performance and its variability optimisation. Results in next chapter directly prove the feasibility of this thesis's hypothesis.

Chapter 6

Circuit Optimisation using Motifs and Evolutionary Algorithm

Contents

Chapter 5 discussed motif compact model extraction and statistical modelling methodology transferring motifs' characteristics to compact model cards, which bridges motifs and circuit design. Since motifs capture the fabrication advantages of layout regularity and allow appropriate investigation of the influence of transistors layout on circuit performance, a circuit

performance optimisation methodology combing evolutionary algorithm and motifs is introduced in this chapter. Testing and evaluating the influence of various motif permutations on circuit optimisation is a curial part of this thesis. Optimisation results not only verify the practicability of the proposed circuit optimisation methodology, but also directly provide evidence to support the thesis' hypothesis.

Several basic logic gates such as NAND, AND, OR and NOR gate are constructed using different motif permutations and are evaluated through exhaustive testing in Section 6.1. However, exhaustive testing is not a realistic and efficient option when motif permutations exponentially increase as the number of transistors in the circuit increases. Testing circuits constructed from hundreds and thousands of motif permutations is a heavy computational burden. The potential advantage of applying EAs to circuit optimisation (introduced in Chapter 4) benefits from the EA's ability to effectively search large solution spaces whilst not being trapped in local optima. The flow of the multi-stage evolutionary algorithm with dynamic mutation mechanism and its components are addressed in Section 6.2. The algorithm improves the efficiency of motif permutation testing for circuit optimisation. In order to verify the practicability and correctness of the proposed EA on circuit optimisation, a benchmark circuit, XOR gate constructed by 12 transistors are tested, which is described in Section 6.3. This XOR gate has moderate size scale with little computation either for the exhaustive testing or EA verification. Finally, two additional benchmark circuits, half adder and full adder, are tested and discussed in Section 6.4. The aim of this chapter is to verify the proposed motifs and multi-stage optimisation methodology, and prove their efficiently in performing circuit optimisation. Although only circuit speed and its variability are considered as optimisation targets, the proposed optimisation methodology can be extended and applied to additional cost functions, including power, stability and sub-circuit yield.

6.1 Applying Motifs to Basic Logic Gates

6.1.1 Logic Gates Test

In order to verify the influence of motifs on circuit performance, four basic logic gate circuits NAND, NOR, AND, and OR are constructed using every possible permutation of various motifs and are considered as benchmark circuits to be tested. Transistor level schematics of these basic logic gates are illustrated in Figure 6.1. Because only O shaped motif and standard (reference) device are extracted using the extraction process described in Chapter 5, all logic

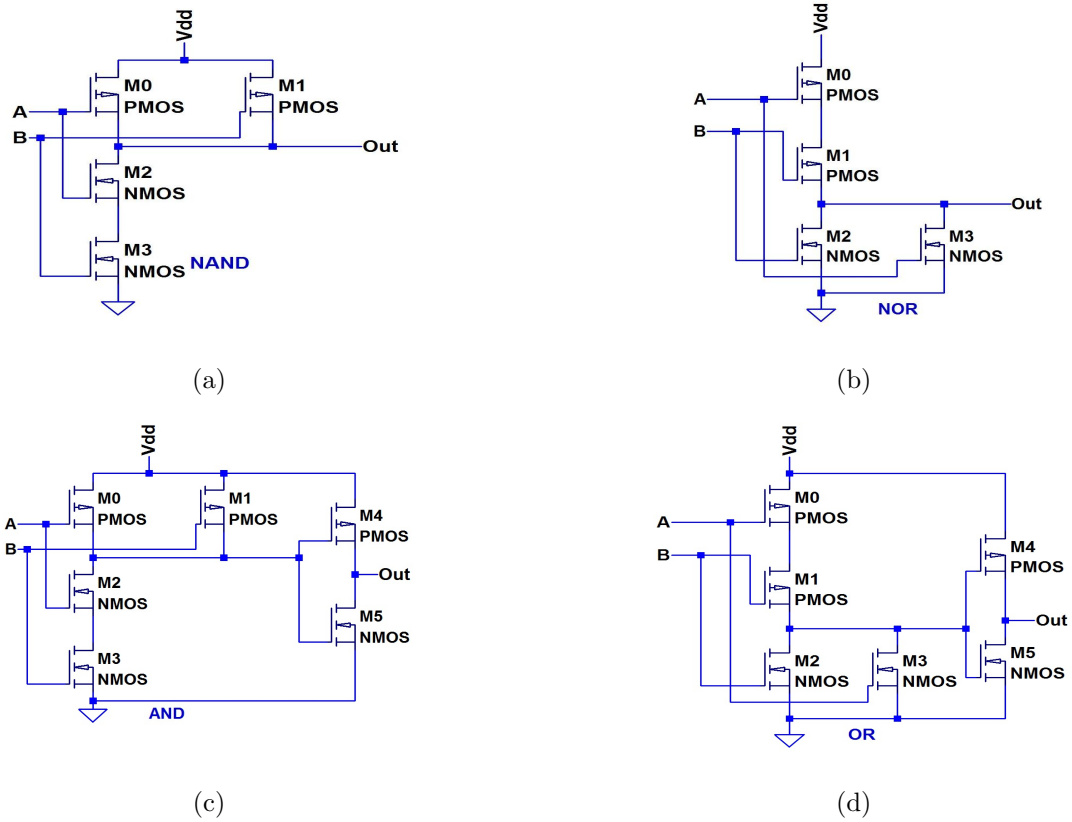


Figure 6.1: Transistor level schematic of four basic logic gates, NAND gate (left upper), NOR (right upper), AND (left bottom), OR (right bottom).

gates are built through the combination of these two motifs. In order to investigate the effects of motifs structure on circuit performance variability, the statistical variability motifs model is used here.

6.1.2 Exhaustive Testing

The candidate basic logic gates are constructed using a small number of transistors. This means that every possible motif permutations that could make up the circuit can be exhaustively tested. As a pair of complementary logic rules, NAND gates and NOR gates have the same set of input test combinations. Table 6.1 lists the NAND gate and NOR gate exhaustive circuit combinations using different motifs, where M_x is a specific transistor name used in circuit. The standard (reference) device is signified by '0', and O shaped motif is signified by '1'. Similarly, AND gate and OR gate have the same test combinations. Since AND and OR gates incorporate more transistors than NAND and NOR logic gates, the corresponding exhausting combinations are more than the NAND and NOR motif permutations, and are listed in Table 6.2. To accurately evaluate the influence of various motif permutations on the

circuit performance, the worst-case delay is used as the assessment metric for overall performance. The worst-case delay of the candidate circuit is defined by the slowest output signal transient response propagation delay caused by changing input signal. A 2-input NAND gate is shown here as an example of how to calculate the worst-case delay. Six propagation delays of NAND gate are measured in accordance with six different input scenarios, which are listed in Table 6.3. Then, the slowest propagation delay is selected as the worst-case delay. Figure 6.2 illustrates these six situations and indicates how to measure the worst propagation delay.

Table 6.1: The exhaustive set of device combinations for the NAND and NOR logic gates. Mx refers to a specific transistor position in each logic gate (see Figure 6.1). At each position, the use of a standard device layout is signified by '0', and the use of an O shaped motif is signified by '1'.

Index	M0:M1:M2:M3	Index	M0:M1:M2:M3
1	0 : 0 : 0 : 0	9	1 : 0 : 0 : 0
2	0 : 0 : 0 : 1	10	1 : 0 : 0 : 1
3	0 : 0 : 1 : 0	11	1 : 0 : 1 : 0
4	0 : 0 : 1 : 1	12	1 : 0 : 1 : 1
5	0 : 1 : 0 : 0	13	1 : 1 : 0 : 0
6	0 : 1 : 0 : 1	14	1 : 1 : 0 : 1
7	0 : 1 : 1 : 0	15	1 : 1 : 1 : 0
8	0 : 1 : 1 : 1	16	1 : 1 : 1 : 1

6.1.3 Results and Observations

Figures 6.3, 6.4, 6.5 and 6.6 show a set of box and whisker plots of the benchmark circuits' statistical simulation results and indicate the influence of the benchmark logic gate circuits built by various motif combinations on circuit speed and its variability. In order to spot interesting effects of the various motif permutations more easily, the results for these motif permutations are sorted into different groups based on their worst-case delay. Tables 6.4 to 6.9 summarise the statistical variability analysis results for the basic logic gates for each group. It is worth noting that circuits constructed by a mixture of motifs have shorter delay and smaller variations than circuits with a single type of motif. This result gives an indication that the performance of larger circuits can significantly benefit from an appropriate mix of different motifs constructing the circuit.

Table 6.2: The exhaustvie AND gate and OR gate circuit combinations

Index	M0:M1:M2:M3:M4:M5	Index	M0:M1:M2:M3M4:M5
1	0 : 0 : 0 : 0 : 0 : 0	33	1 : 0 : 0 : 0 : 0 : 0
2	0 : 0 : 0 : 0 : 0 : 1	34	1 : 0 : 0 : 0 : 0 : 1
3	0 : 0 : 0 : 0 : 1 : 0	35	1 : 0 : 0 : 0 : 1 : 0
4	0 : 0 : 0 : 0 : 1 : 1	36	1 : 0 : 0 : 0 : 1 : 1
5	0 : 0 : 0 : 1 : 0 : 0	37	1 : 0 : 0 : 1 : 0 : 0
6	0 : 0 : 0 : 1 : 0 : 1	38	1 : 0 : 0 : 1 : 0 : 1
7	0 : 0 : 0 : 1 : 1 : 0	39	1 : 0 : 0 : 1 : 1 : 0
8	0 : 0 : 0 : 1 : 1 : 1	40	1 : 0 : 0 : 1 : 1 : 1
9	0 : 0 : 1 : 0 : 0 : 0	41	1 : 0 : 1 : 0 : 0 : 0
10	0 : 0 : 1 : 0 : 0 : 1	42	1 : 0 : 1 : 0 : 0 : 1
11	0 : 0 : 1 : 0 : 1 : 0	43	1 : 0 : 1 : 0 : 1 : 0
12	0 : 0 : 1 : 0 : 1 : 1	44	1 : 0 : 1 : 0 : 1 : 1
13	0 : 0 : 1 : 1 : 0 : 0	45	1 : 0 : 1 : 1 : 0 : 0
14	0 : 0 : 1 : 1 : 0 : 1	46	1 : 0 : 1 : 1 : 0 : 1
15	0 : 0 : 1 : 1 : 1 : 0	47	1 : 0 : 1 : 1 : 1 : 0
16	0 : 0 : 1 : 1 : 1 : 1	48	1 : 0 : 1 : 1 : 1 : 1
17	0 : 1 : 0 : 0 : 0 : 0	49	1 : 1 : 0 : 0 : 0 : 0
18	0 : 1 : 0 : 0 : 0 : 1	50	1 : 1 : 0 : 0 : 0 : 1
19	0 : 1 : 0 : 0 : 1 : 0	51	1 : 1 : 0 : 0 : 1 : 0
20	0 : 1 : 0 : 0 : 1 : 1	52	1 : 1 : 0 : 0 : 1 : 1
21	0 : 1 : 0 : 1 : 0 : 0	53	1 : 1 : 0 : 1 : 0 : 0
22	0 : 1 : 0 : 1 : 0 : 1	54	1 : 1 : 0 : 1 : 0 : 1
23	0 : 1 : 0 : 1 : 1 : 0	55	1 : 1 : 0 : 1 : 1 : 0
24	0 : 1 : 0 : 1 : 1 : 1	56	1 : 1 : 0 : 1 : 1 : 1
25	0 : 1 : 1 : 0 : 0 : 0	57	1 : 1 : 1 : 0 : 0 : 0
26	0 : 1 : 1 : 0 : 0 : 1	58	1 : 1 : 1 : 0 : 0 : 1
27	0 : 1 : 1 : 0 : 1 : 0	59	1 : 1 : 1 : 0 : 1 : 0
28	0 : 1 : 1 : 0 : 1 : 1	60	1 : 1 : 1 : 0 : 1 : 1
29	0 : 1 : 1 : 1 : 0 : 0	61	1 : 1 : 1 : 1 : 0 : 0
30	0 : 1 : 1 : 1 : 0 : 1	62	1 : 1 : 1 : 1 : 0 : 1
31	0 : 1 : 1 : 1 : 1 : 0	63	1 : 1 : 1 : 1 : 1 : 0
32	0 : 1 : 1 : 1 : 1 : 1	64	1 : 1 : 1 : 1 : 1 : 1

In NAND and NOR logic gates, M0 and M1 are PMOS transistors and located in the pull-up network/part in CMOS circuit. This means that circuit performance improvements can be

Table 6.3: NAND gate output delay measurement scenario based on different inputs transient changing scenarios.

Scenario	AB $<---->$ AB	Output response changing
1	01 $---->$ 11	1 $-->$ 0
2	01 $<----$ 11	1 $<--$ 0
3	10 $---->$ 11	1 $-->$ 0
4	10 $<----$ 11	1 $<--$ 0
5	00 $---->$ 11	1 $-->$ 0
6	00 $<----$ 11	1 $<--$ 0

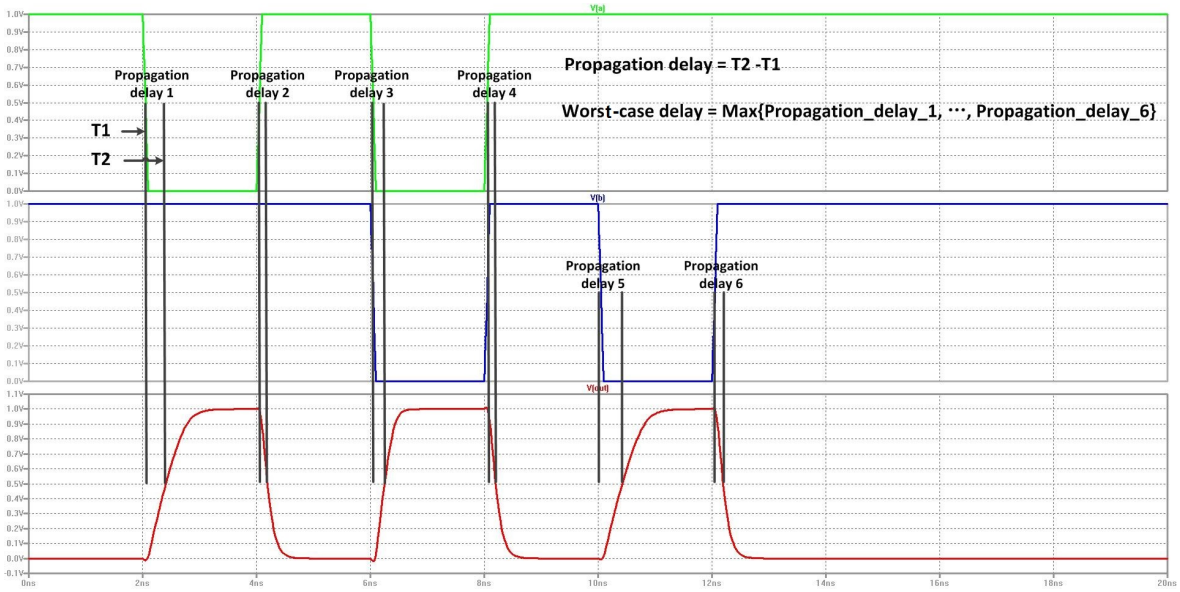


Figure 6.2: NAND gate output delay measurement scenario. Six propagation delays are measured in accordance with six various input scenarios. The slowest delay in these six propagation delays is selected as the worst-case delay.

attributed to pull-up network performance improvements. To analyse the effects of the pull-up network on propagation delay, all the PMOS transistors in the NAND and NOR gates are replaced by a corresponding equivalent resistor. The original NAND and NOR circuits have been transformed to NMOS logic circuits (shown in Figure 6.7). In general, the pull-up delay increases with the value of the load resistor increasing [189]. To verify the circuits composed of the O shaped motif and the standard device also follow this principle, the output resistance of O shaped and standard device were measured and are shown in Figure 6.8. The standard device has a larger output resistance than the O shaped motif. When the standard device is used in the pull-up network, it results in the pull-up delay increasing compared to the use of the O shaped motif. AND and OR gate simulation results can also explain through motifs'

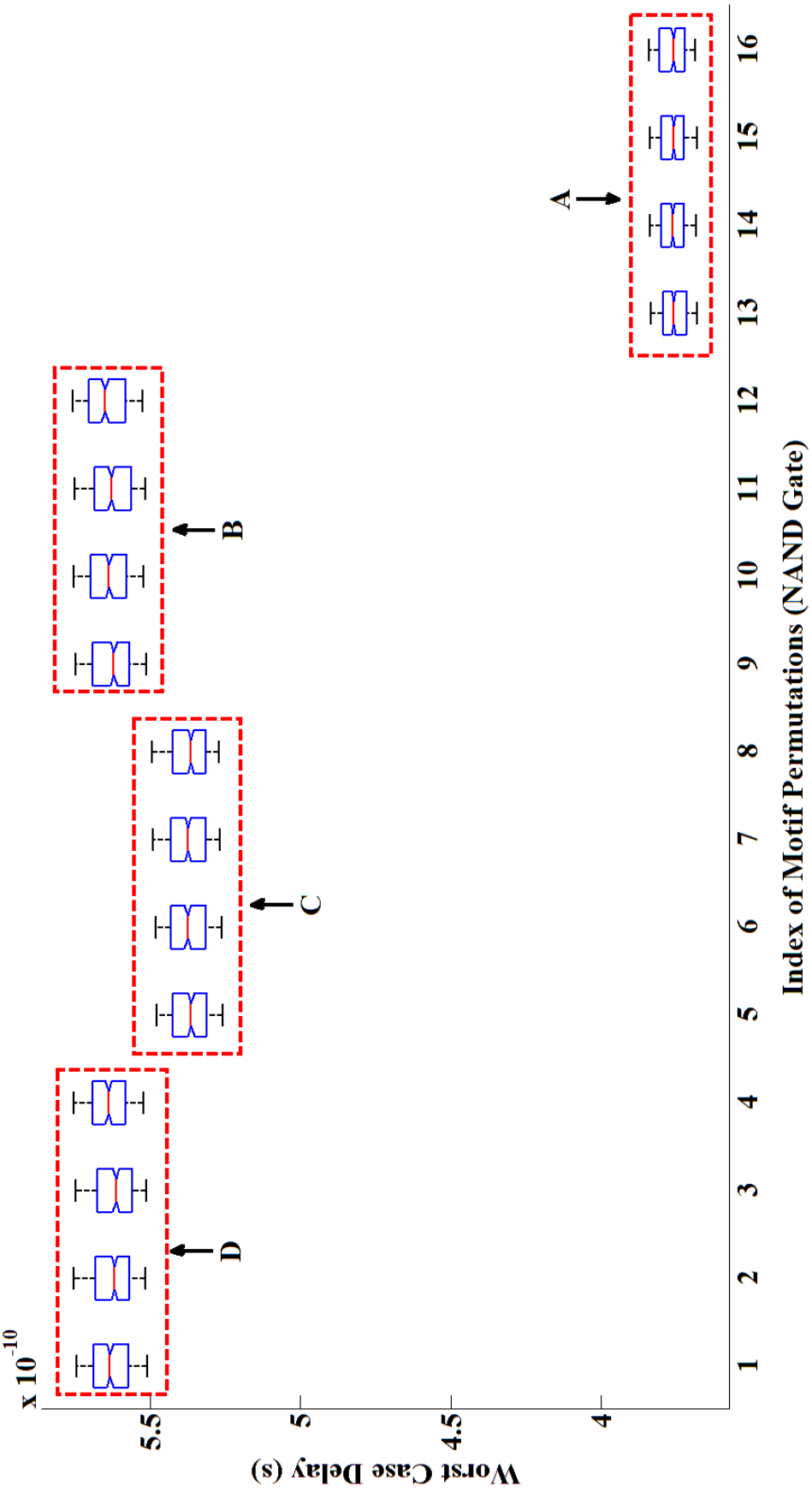


Figure 6.3: The relationship of worse case delay and motif permutations (NAND gate). Only *O* shaped motif and standard (reference) device are used to constitute various motif permutations. These motif permutations are sorted into different groups (A, B, C, D) based on measured worst-case delays.

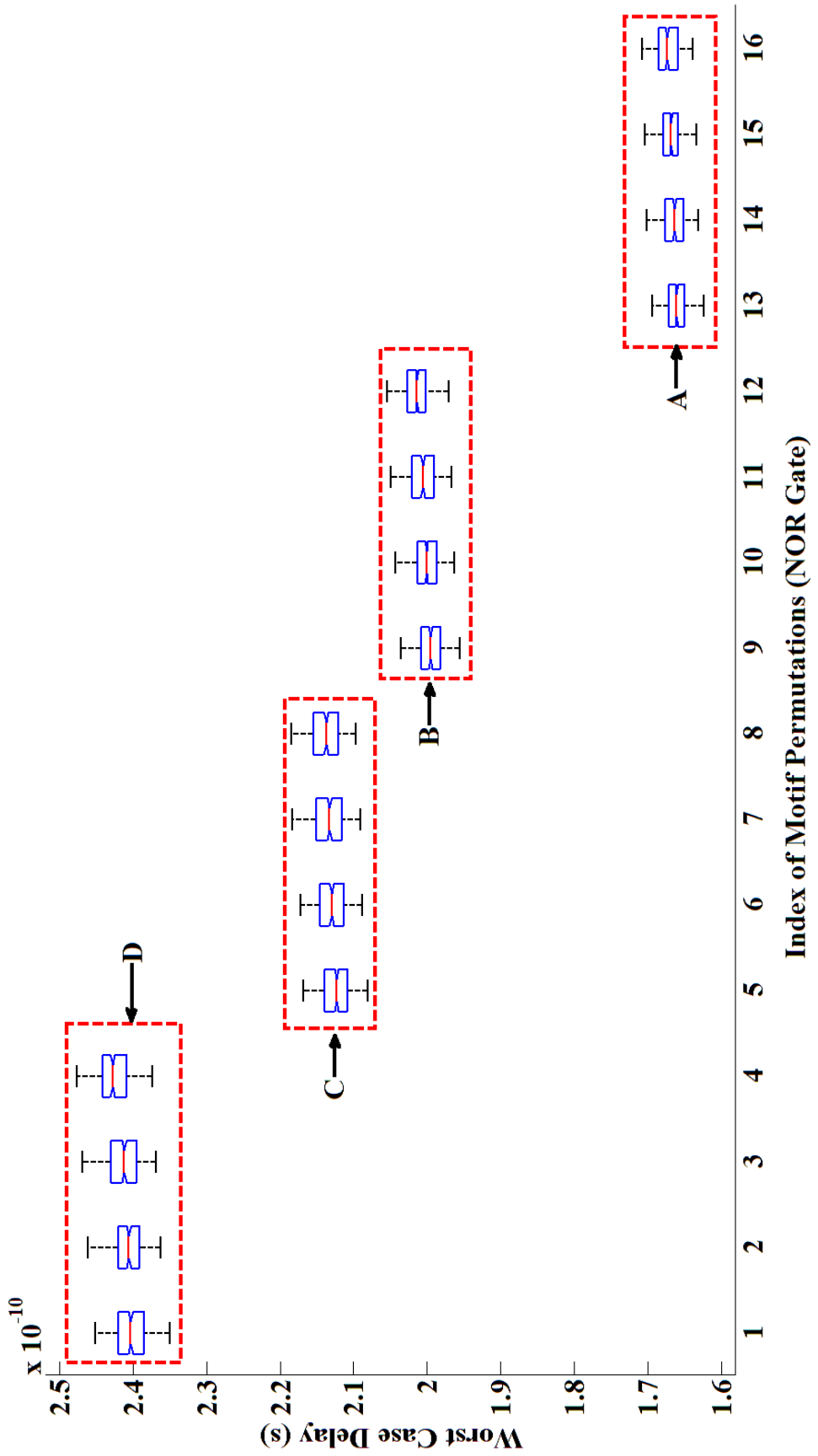


Figure 6.4: The relationship of worse case delay and motif permutations (NOR gate). Only O shaped motif and standard (reference) device are used to constitute various motif permutations. These motif permutations are sorted into different groups (A, B, C, D) based on measured worst-case delays.

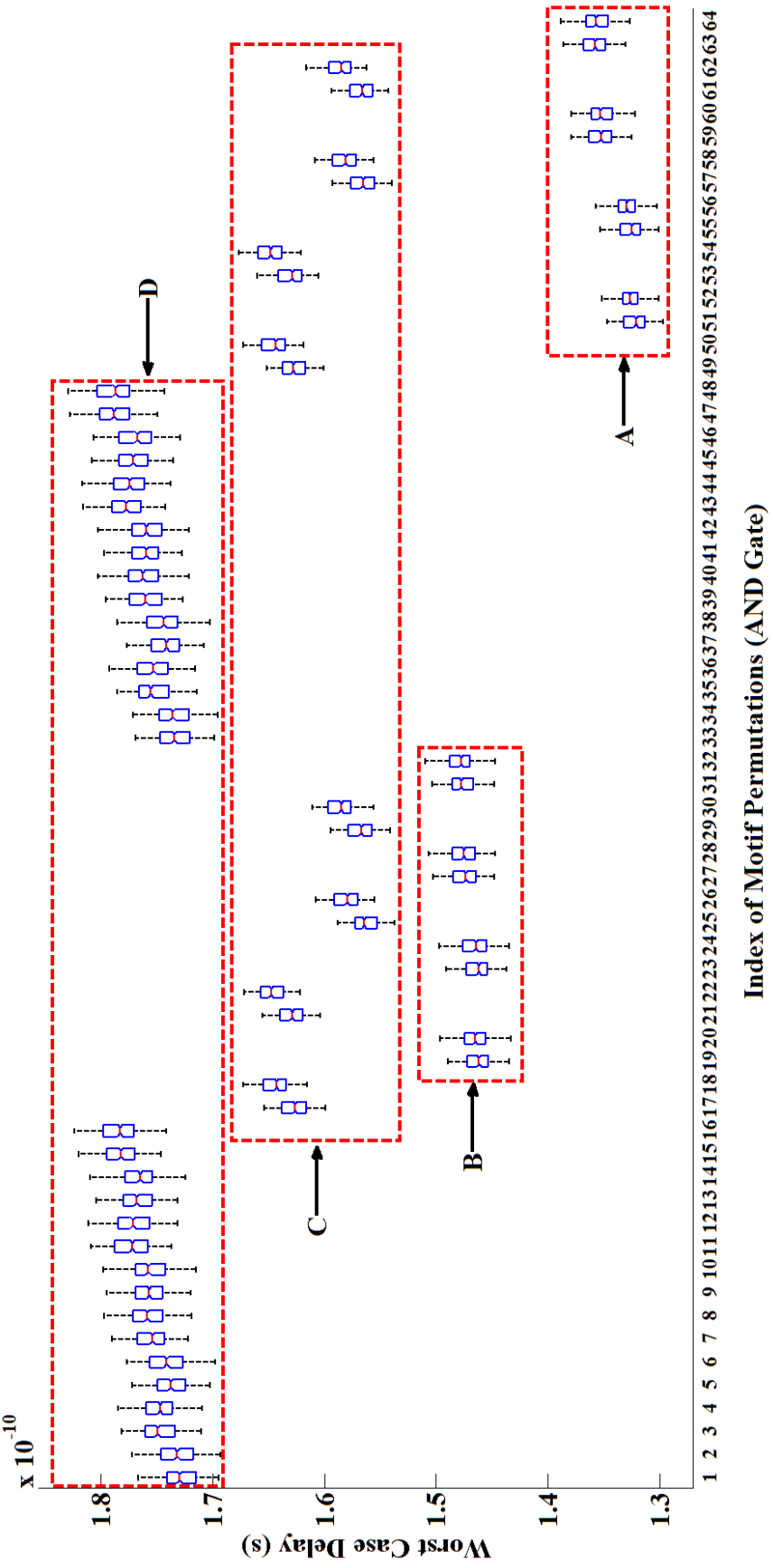


Figure 6.5: The relationship of worse case delay and motif permutations (AND gate). Only O shaped motif and standard (reference) device are used to constitute various motif permutations. These motif permutations are sorted into different groups (A, B, C, D) based on measured worst-case delays.

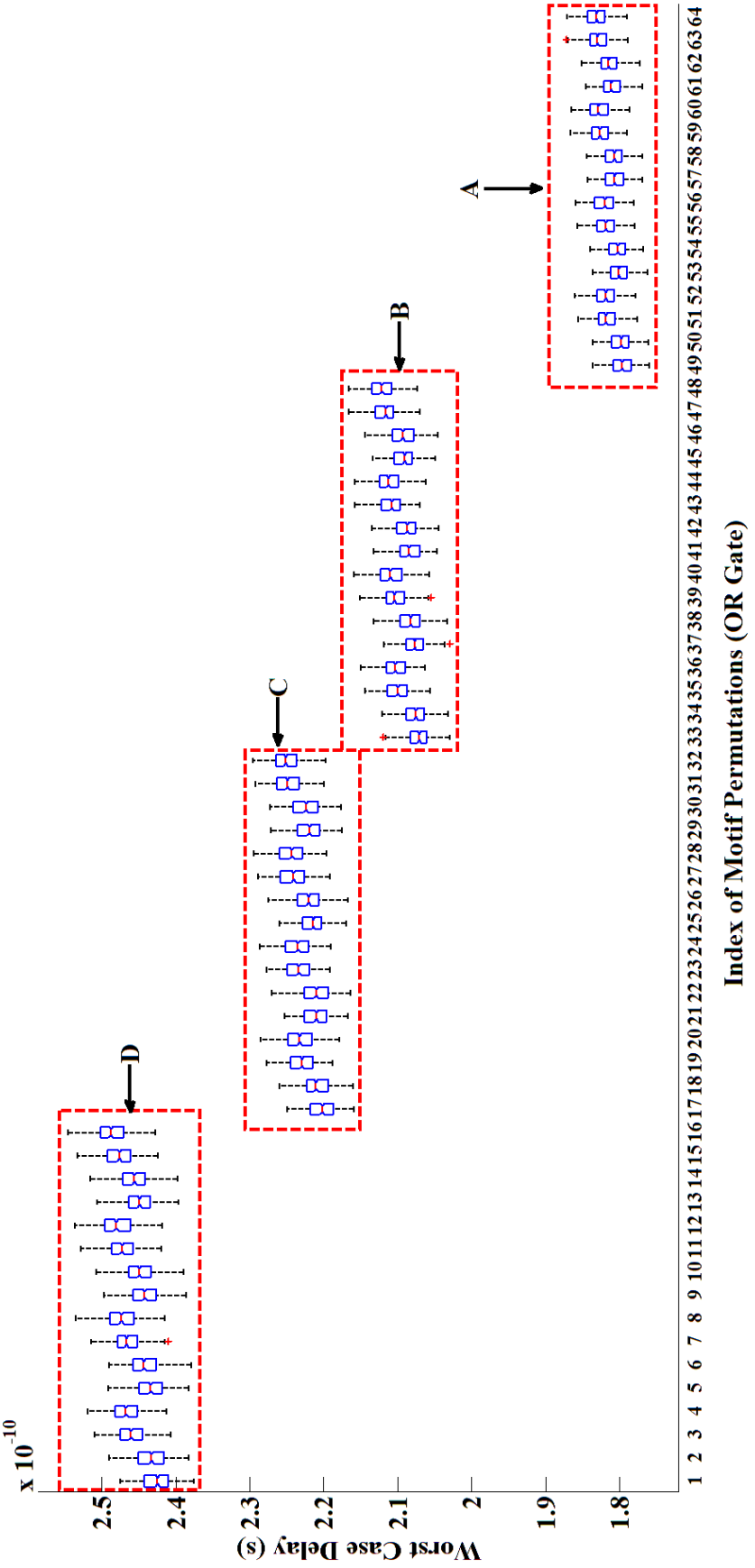


Figure 6.6: The relationship of worse case delay and motif permutations (OR gate). Only O shaped motif and standard (reference) device are used to constitute various motif permutations. These motif permutations are sorted into different groups (A, B, C, D) based on measured worst-case delays.

Table 6.4: The statistical variability results of the motif permutation in a NAND gate (Groups are shown in Figure 6.3)

Gate	Group	Index	Mean (ps)	Standard deviation (ps)
NAND	A	16	375.6	4.485
		15	376.2	4.478
		14	375.3	4.573
		13	376.3	4.489
	C	8	538.1	6.567
		7	538.0	6.662
		6	537.1	6.470
		5	536.6	6.438
	B	12	564.0	6.803
		11	562.6	6.760
		10	563.9	6.690
		9	562.6	6.777
	D	4	563.6	6.843
		3	562.4	6.714
		2	563.4	6.703
		1	562.4	6.823

Table 6.5: The statistical variability results of the motif permutation in a NOR gate (Groups are shown in Figure 6.4)

Gate	Group	Index	Mean (ps)	Standard deviation (ps)
NOR	A	16	167.3	1.524
		15	166.8	1.621
		14	166.4	1.561
		13	165.9	1.555
	B	12	201.3	1.794
		11	200.5	1.833
		10	200.1	1.819
		9	199.3	1.818
	C	8	214.1	2.062
		7	213.5	2.171
		6	212.9	2.106
		5	212.3	2.068
	D	4	242.4	2.191
		3	241.5	2.183
		2	240.9	2.223
		1	240.1	2.219

Table 6.6: The statistical variability results of the motif permutation in an AND gate (Groups are shown in Figure 6.5)

Gate	Group	Index	Mean (ps)	Standard deviation (ps)
AND	A	51	132.3	1.130
		52	132.7	1.009
		55	132.6	1.185
		56	133.0	1.076
		59	135.3	1.242
		60	135.2	1.262
		63	135.9	1.208
		64	135.7	1.307
	B	19	146.3	1.274
		20	146.5	1.345
		23	146.3	1.250
		24	146.5	1.372
		27	147.4	1.328
		28	147.6	1.371
		31	147.7	1.271
		32	147.8	1.328
	C	17	162.7	1.259
		18	164.4	1.242
		21	162.9	1.257
		22	164.7	1.222
		25	156.3	1.203
		26	158.1	1.203
		29	156.7	1.282
		30	158.6	1.294
		49	162.7	1.237
		50	164.5	1.243
		53	163.0	1.266
		54	164.8	1.289
		57	156.5	1.308
		58	158.2	1.283
		61	156.7	1.227
		62	158.6	1.227

Table 6.7: The continued statistical variability results of the motif permutation in an AND gate (Groups are shown in Figure 6.5)

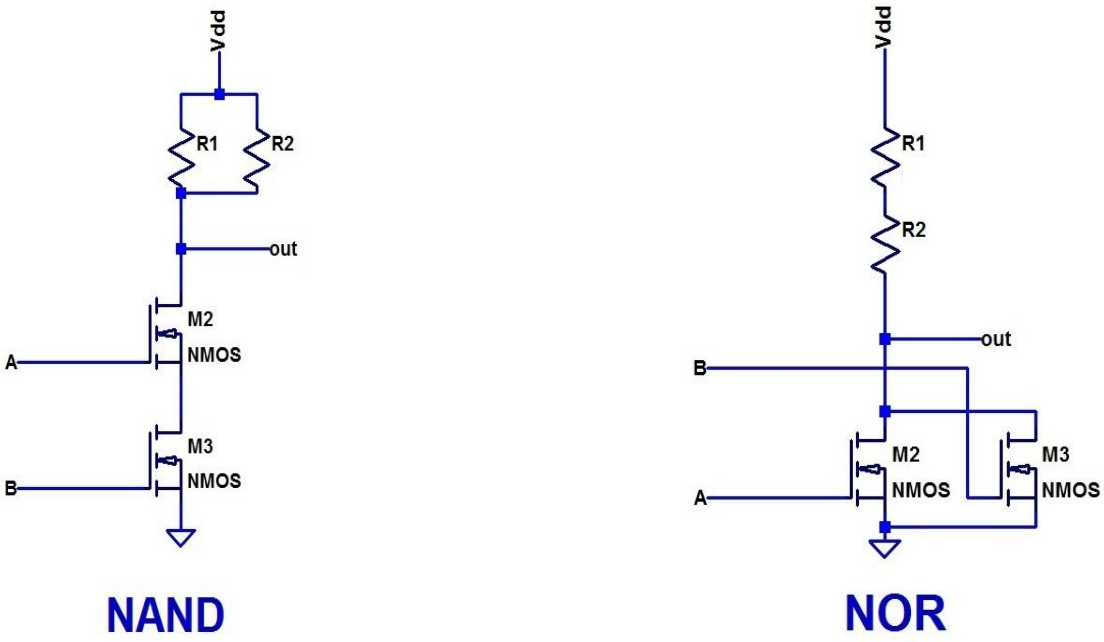
Gate	Group	Index	Mean (ps)	Standard deviation (ps)
AND	D	1	172.8	1.726
		2	173.1	1.825
		3	174.7	1.652
		4	174.6	1.643
		5	173.7	1.642
		6	174.1	1.789
		7	175.4	1.566
		8	175.8	1.713
		9	175.6	1.588
		10	175.6	1.909
		11	177.2	1.796
		12	177.1	1.878
		13	176.6	1.667
		14	176.5	1.821
		15	178.2	1.765
		16	178.3	1.856
		33	173.3	1.702
		34	173.5	1.768
		35	175.3	1.669
		36	175.4	1.751
		37	174.1	1.647
		38	174.4	1.762
		39	176.0	1.650
		40	176.2	1.828
		41	176.0	1.539
		42	176.0	1.766
		43	178.0	1.695
		44	177.5	1.729
		45	177.1	1.665
		46	176.8	1.818
		47	178.8	1.670
		48	178.7	1.836

Table 6.8: The statistical variability results of the motif permutation in an OR gate (Groups are shown in Figure 6.6)

Gate	Group	Index	Mean (*ps*)	Standard deviation (*ps*)
OR	A	49	179.6	1.534
		50	180.1	1.547
		51	182.0	1.603
		52	181.8	1.621
		53	180.0	1.487
		54	180.2	1.524
		55	182.0	1.557
		56	182.0	1.667
		57	180.6	1.544
		58	180.7	1.497
		59	182.6	1.517
		60	182.7	1.605
		61	181.0	1.549
		62	181.4	1.502
		63	182.9	1.513
		64	183.1	1.713
	B	33	207.2	1.721
		34	207.6	1.836
		35	210.0	1.778
		36	210.3	1.747
		37	207.7	1.720
		38	208.3	1.852
		39	210.3	1.749
		40	210.9	2.004
		41	208.4	1.673
		42	208.8	1.8124
		43	210.9	1.818
		44	211.2	1.880
		45	209.1	1.638
		46	209.3	1.951
		47	211.7	1.817
		48	212.1	1.929

Table 6.9: The continued statistical variability results of the motif permutation in an OR gate (Groups are shown in Figure 6.6)

Gate	Group	Index	Mean (ps)	Standard deviation (ps)
OR	C	16	248.7	1.259
		17	220.3	1.969
		18	220.8	1.970
		19	223.0	1.953
		20	223.2	2.100
		21	221.1	1.953
		22	221.1	2.182
		23	223.4	1.954
		24	223.7	2.034
		25	221.5	1.829
		26	222.0	2.067
		27	224.1	2.041
		28	224.4	2.086
		29	222.0	1.980
		30	222.4	2.060
		31	224.8	2.037
	D	1	242.6	2.144
		2	243.3	2.274
		3	246.0	2.054
		4	246.5	2.096
		5	243.4	2.134
		6	244.2	2.195
		7	246.5	1.968
		8	247.3	2.200
		9	244.2	2.144
		10	244.9	2.235
		11	247.3	2.055
		12	247.8	2.404
		13	244.9	2.111
		14	245.6	2.275
		15	247.7	2.196
		16	248.7	2.273



(a) (b)

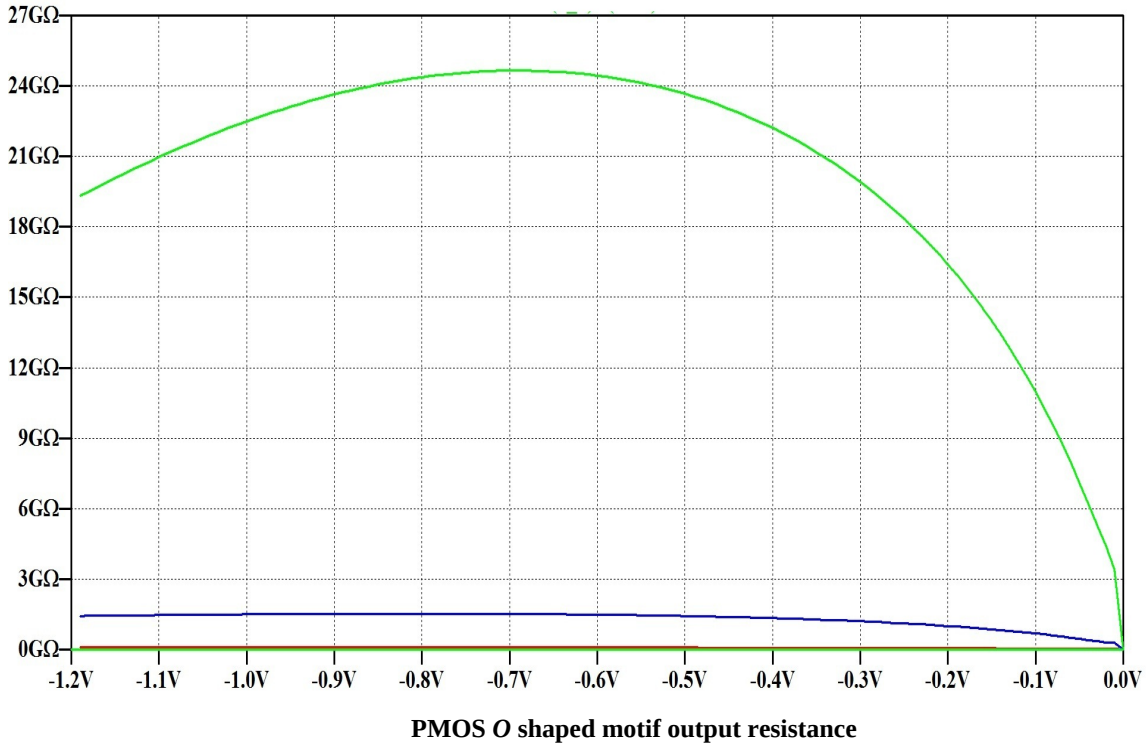
Figure 6.7: NMOS logic circuits of NAND and NOR gate transistor level schematics. The original PMOS transistors are replaced by the equivalent resistor. These circuits are tested using the equivalent resistor value which is equal to the standard device and O shaped motif, respectively.

output resistance analysis.

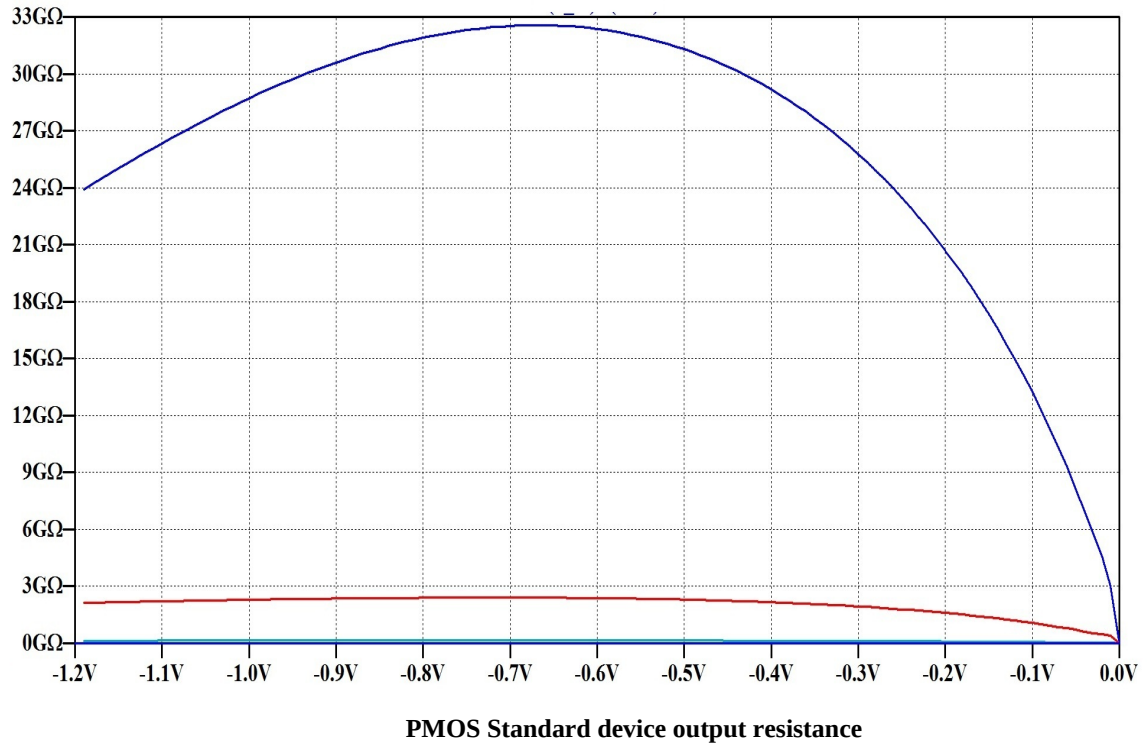
6.2 Multi-Stage Evolutionary Algorithm for Circuit Optimisation

6.2.1 Methodology

For the small scale circuits in the previous section, exhaustive testing is an efficient and fast method to capture the best motif permutation for specific circuit designs. However, as the circuit functionality becomes more complex, increasing the number of transistors in the circuit results in the number of permutations to increase exponentially. Exhaustive testing then becomes infeasible. To rapidly and efficiently test every possible motif permutation to achieve circuit performance optimisation and mitigate the effect of variability on the circuit performance, a multi-stage evolutionary algorithm methodology is proposed and described, which, for the first time, develops a bottom-up methodology to combine device layout motif at device level and EA to sort out circuit variability issues at circuit level.



(a)



(b)

Figure 6.8: The PMOS *O* shaped motif and standard device output resistance value is measured under different drain voltages. Y-axis is output resistance; and X-axis is gate voltage.

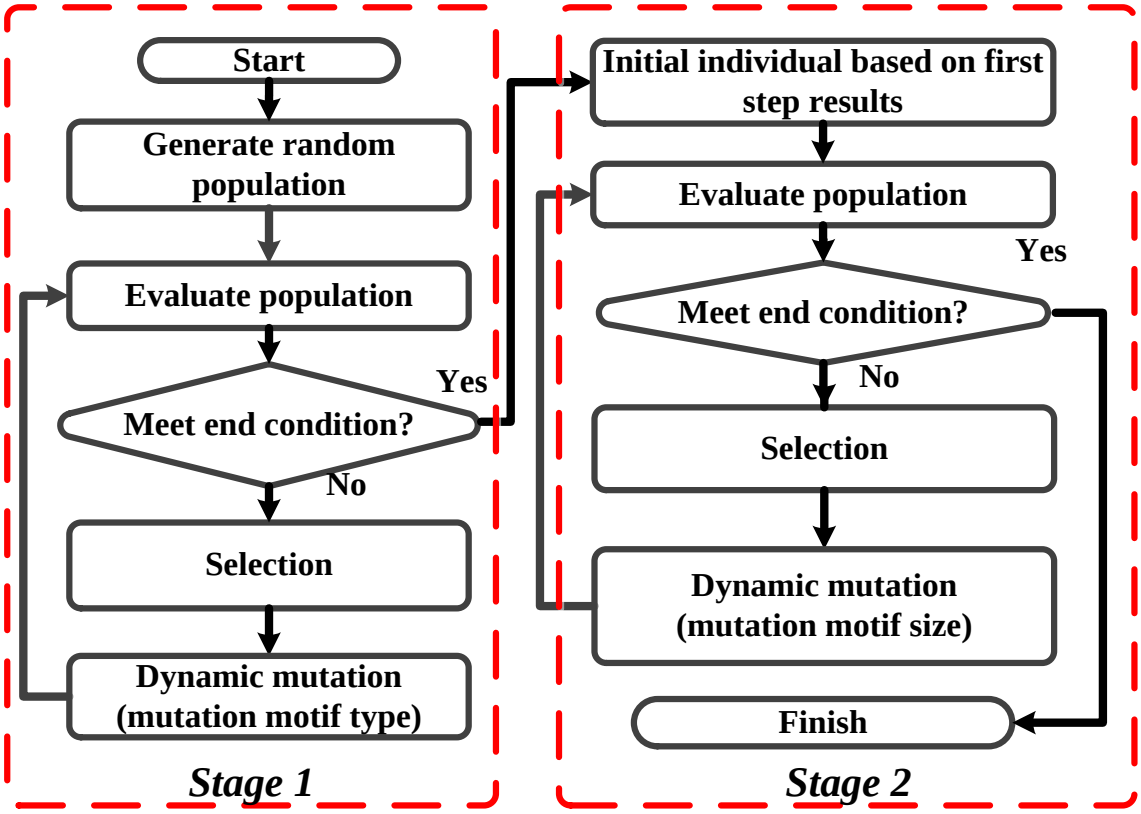


Figure 6.9: The flow chart of multi-stage evolutionary algorithm

Firstly, circuit worst-case propagation delay is optimised by evolving the circuit design on different motif permutations. Then, the best evolved solution generated in stage one is passed to a second stage of evolution to optimise motif size/drive current. The fitness function is based on both the mean and the standard deviation of the worst-case delay, thus accounting for both circuit performance and variability. As previously described, a circuit's performance can be improved by an optimal choice of motif permutations. Therefore, the first stage of EA optimisation seeks to evolve the best choice of motifs for small worst-case delay without considering circuit variability.

In the second optimisation stage, performance variability (mainly propagation delay variation) is optimised, where the mean and standard deviation of the worst-case delay are minimised with equal weighting through evolution of device sizes/drive currents. In the second stage, only motifs geometry sizes are evolved. The circuit used motif permutation keeps the same as the best evolved solutions obtained from the first optimisation stage. The flow chart of multi-stage evolutionary algorithm is shown in Figure 6.9.

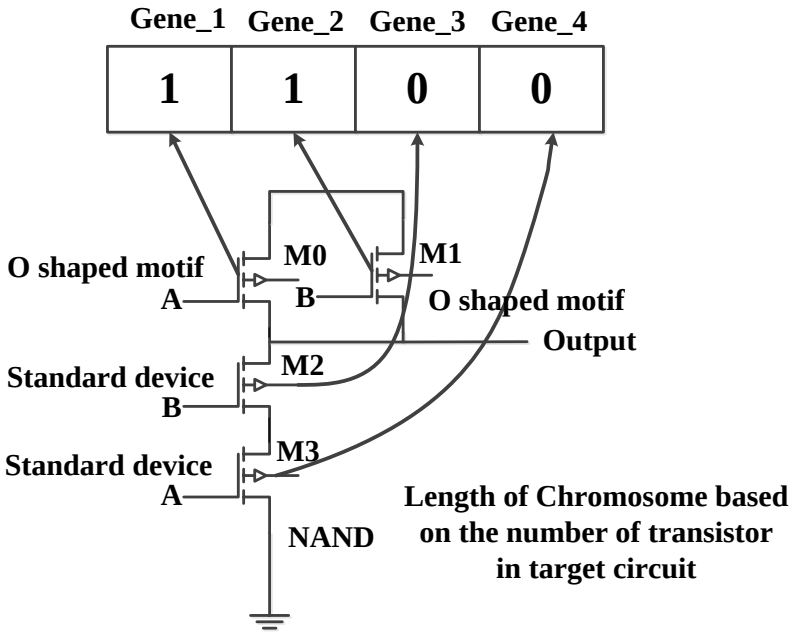


Figure 6.10: An example of NAND gate is represented by a genotype structure in first stage. Each transistor is represented a gene in the genotype. The genotype size depends on the number of transistor in circuit.

6.2.2 First Stage EA Optimisation

Representation

The first step of building an evolutionary algorithm is to decide on a genetic representation of candidate solutions. To handle genotype easily, a binary string is employed to represent the candidate circuit topology, where the binary representation acts as the EA is genetic encoding (an example of NAND gate circuit is represented by a chromosome structure in the first stage, which is illustrated in the Figure 6.10). Each transistor in the circuit is represented by a gene in the genotype. The length of the genotype equals to the number of transistor in the circuit. Hence, genotype size varies according to the circuit size, but is unchanged during an optimisation run of a specific circuit. Two types of motif/device are available at this stage: standard device signified by a gene '0' and *O* shaped motif, signified by a gene '1'.

Genetic Operation

The evolutionary techniques used in this two stage approach consider an EA which uses

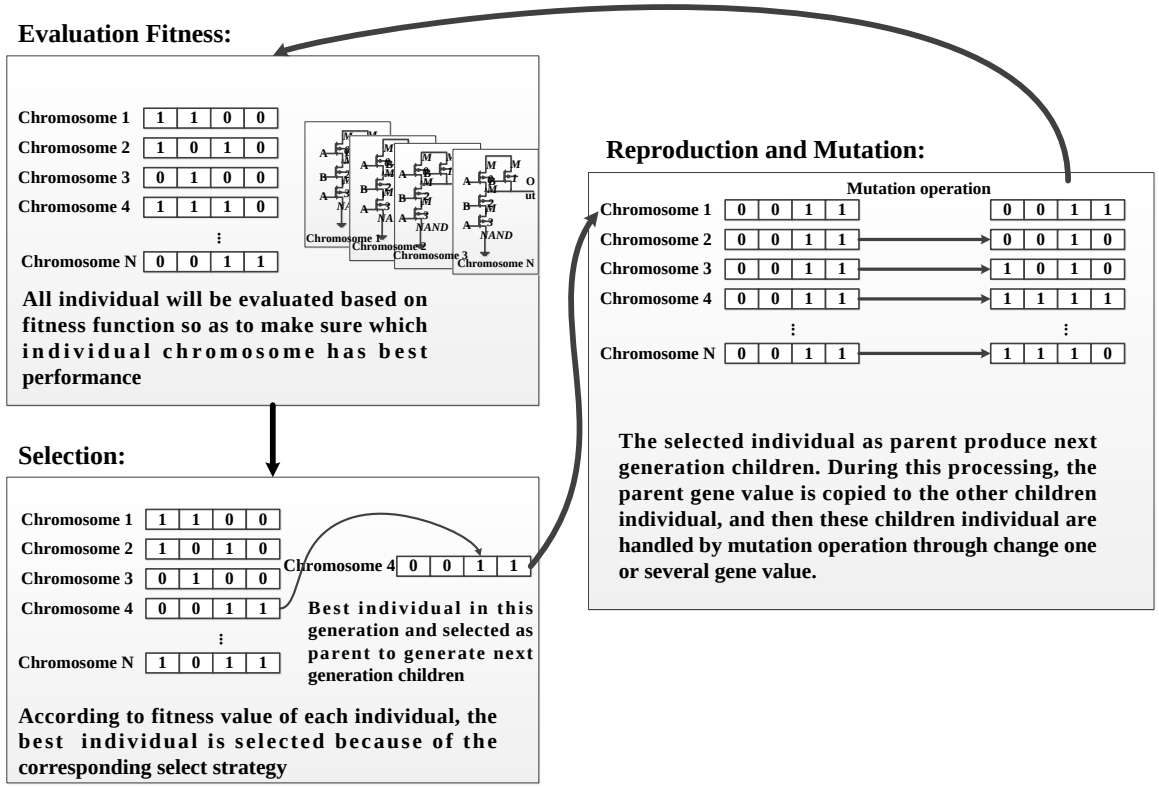


Figure 6.11: An example of the evolutionary loop, demonstrating the genetic operations.

mutation and selection. The variation operation in this problem is to swap the motif type and change device size. Comparing to mutation, crossover cannot efficiently perform these variations, thus no crossover is used. Figure 6.11 depicts an evolutionary loop example to briefly illustrate these genetic operations. Offspring is generated through randomly selecting and mutating the parent gene values. In order to escape local optimum [89][190], a dynamic gene mutation is used for this experiment, which randomly determines how many genes are selected to perform mutation. The drawback of the use of single gene mutation in EA and the corresponding simulation result are addressed in Section 6.4.3. Figure 6.12 shows the dynamic mutation operation process flow chart. The maximum number of mutated genes is five and the minimum number is one. Based on literature [134] introduced a low level of mutation will promote to prevent and control premature convergence, here the maximum number of mutated gene is set to five. By dynamically controlling the mutated gene in the individual genotype, individuals have been improved which prevents individual genotype prematurely converging to a local optimum.

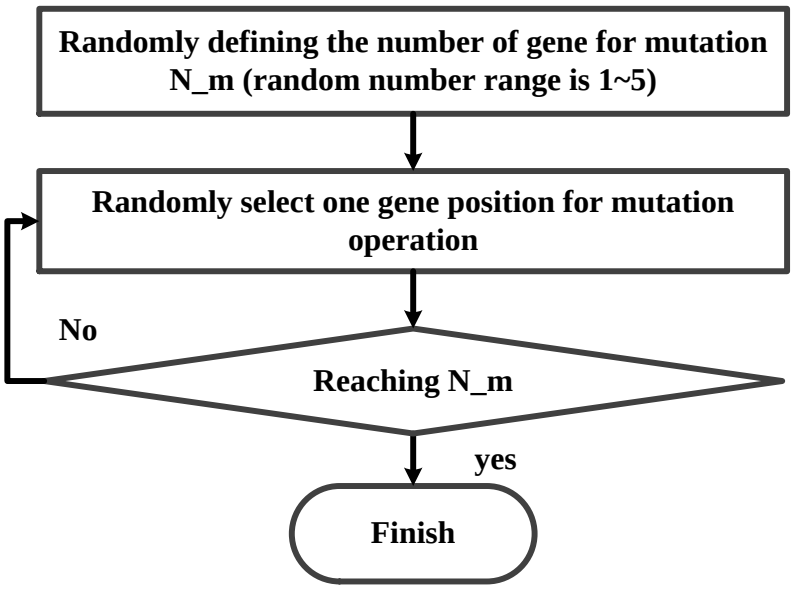


Figure 6.12: The flow chart of the dynamic genen mutation. The maximum number of gene for mutation is 5; and the minimum number of gene for mutation is 1.

Evaluation

The aim of first stage optimisation is to find the best circuit combination by selecting motif permutations to achieve the minimum worst-case delay. The worst-case delay of an individual is the slowest output signal transient response propagation delay caused by an input signal change. Propagation delays and the calculation of the worst-case delay are similar to that mentioned in the previous section. Candidate solutions make up a population; each candidate has mutually independent worst-case delay. Then, these candidate worst-case delays are ranked; and the shortest delay from these worst-case delays is selected as the fitness value to evaluate each individual. The fitness function is calculated by using Equation 6.1:

$$Fitness = min\{T_{worstdelay}(1), \ldots, T_{worstdelay}(n)\} \tag{6.1}$$

Where $T_{worstdelay}(n)$ is the *n-th* individual's worst-case delay, where n is the number of the individual.

Table 6.10: Active gate widths of two device types (having same gate length *L=50 nm*). Wider devices are constructed from minimal devices connected in parallel.

Type	Width
Standard device	80 nm (m=0)
	160 nm (m=1)
	240 nm (m=2)
	320 nm (m=3)
	400 nm (m=4)
	480 nm (m=5)
O shaped motif	160 nm (m=0)
	320 nm (m=1)
	480 nm (m=2)

6.2.3 Second Stage EA Optimisation

Representation

The purpose of the second stage optimisation is to minimize circuit variability by reconfiguring device widths. Each type of motif has several different device widths available. All of available device widths are listed in Table 6.10. Moreover, the gene value in the binary string only has two optional values '0' and '1'. The binary string is not appropriate to represent this problem because the least of device width options is three. Two optional values in the binary string are not enough to represent these device widths. Therefore, an integer string is used to represent the genotype. As the circuit topology being optimised in stage two is identical to that of stage one, the number of gene in the second stage genotype is the same as in first stage optimisation. Each gene position is fixed to a specific motif type inherited from the first stage evolution. In stage two the gene value represents the corresponding device width. The example of NAND gate chromosome in second stage is depicted in Figure 6.13.

Genetic Operation

The number of optimisation objectives increases in the second stage. In order to optimise circuit delay variability, statistical ensembles of 300 samples are simulated to obtain the standard deviation of the mean worst-case propagation delay. Because of these changes the selection and evaluation process undergoes necessary modifications. The total number of

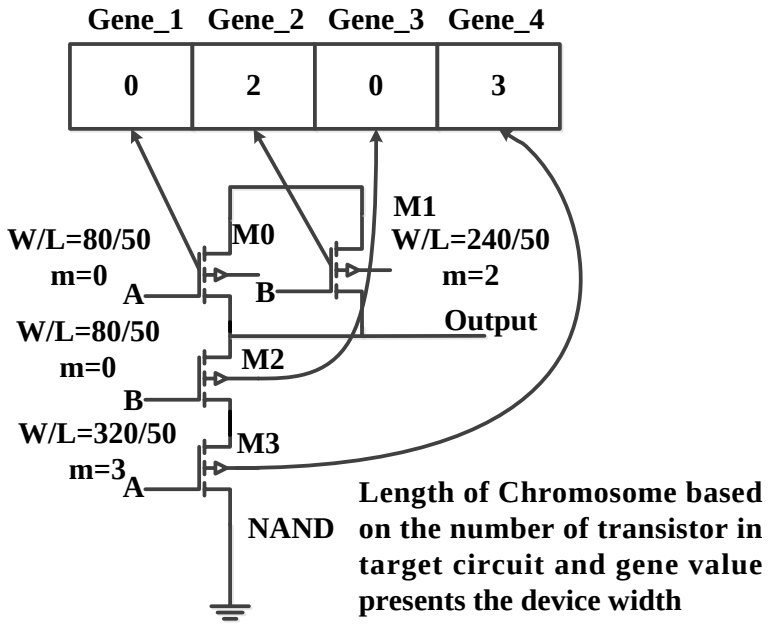


Figure 6.13: The genotype structure of NAND gate in second stage. In the second stage, the genotype size is the same as in the first stage. Because the gene in genotype represents the motif width, the genotype adapts integer representation.

generations in the second stage decreases to 50 in order to reduce the computational demand when performing the statistical samples. Mutation operation is used to evolve motif size in second stage. The dynamic gene mutation technique is adapted to avoid individuals prematurely converging to a local optimum; the principle of this dynamic technique is described in the previous Section 6.2.2. Due to the optimisation objectives increasing in the second stage, it needs an "appropriate" population size for a better chance to explore the search space and to discover possible good solutions. However, the population size cannot be set too large to avoid suffering from high computational cost. Hence, the population size increases from 20 to 40. A (20 +20) selection strategy is used in the second stage. For each parent, one offspring is produced.

Evaluation

The second stage optimisation involves the multi-objective optimisation that is the reduction of the mean and standard deviation of the worst-case propagation delay through evolving device size keeping the previously evolved motif configuration the same. An average ranking (AR) method, proposed by Bentley and Wakefield [191], is adapted for evaluation so that

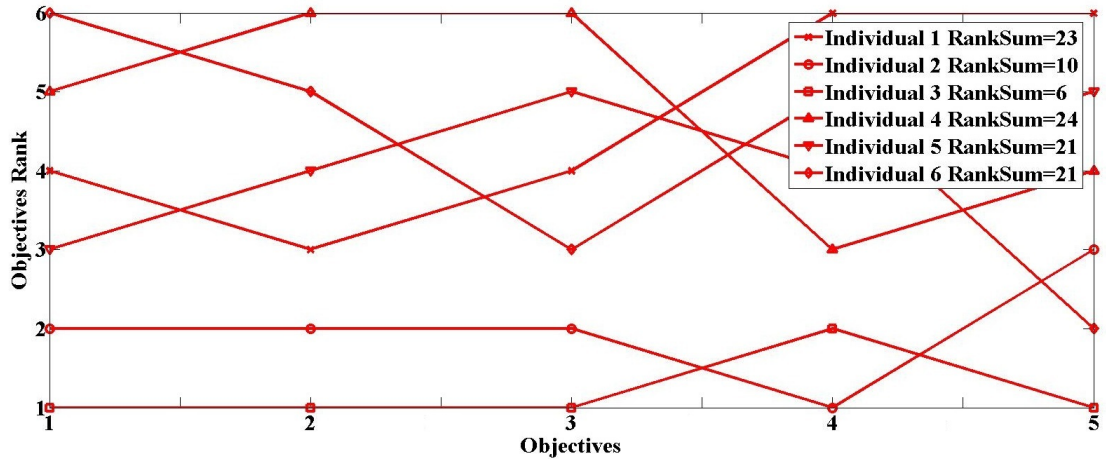


Figure 6.14: An example of AR method calculation. In this example, six individuals are evaluated and five optimisation objectives are optimised. The fitness is calculated as the sum of these ranking positions of the solutions of all objectives, and ranking these obtained fitness of solutions. The best fitness is selected from the smallest rank value. Individual 3 has smallest ranking sum value which means that the individual 3 has the best fitness value.

the algorithm selects elite offspring for reproduction. A set of ranking list of fitness values for each objective are built and extracted from the fitness of every solution by AR. These objective ranking lists are individually sorted in the order of fitness, resulting in a set of different ranking positions for every solution for each objective. The ranking positions of a solution X_n for all objectives are defined the follow the vector $R(X_n) = (Objective1(X_n), Objective2(X_n), \ldots, ObjectiveM(X_n)))$. Where $ObjectiveM(X_n)$ is the rank of X_n for the *M-th* objective.The fitness calculation is defined as the sum of these ranking positions of the solutions of all objectives, and then ranking these obtained fitness of solutions so that best solutions can be selected. Hence, the higher fitness rank a solution has, the greater its chance of being selected for producing offspring. Because all objectives are treated separately, AR is a range-independent method [191]. Of course, although AR method risks a loss of diversity and can lead to solutions falling into a sub-area of the Pareto front, this method has successfully handled convergence for many-objective problems [192]-[196]. In the second stage evaluation, the mean and standard deviation of the worst-case delay are used to calculate the solution's fitness value. The fitness function is calculated by the following equations.

$$Fitness(i) = \sum_{0<j<n} Rank(i,j) \tag{6.2}$$

Where m is the individual in the population, n is the number of optimisation objectives (in this case only two). Figure 6.14 illustrates this fitness calculation process.

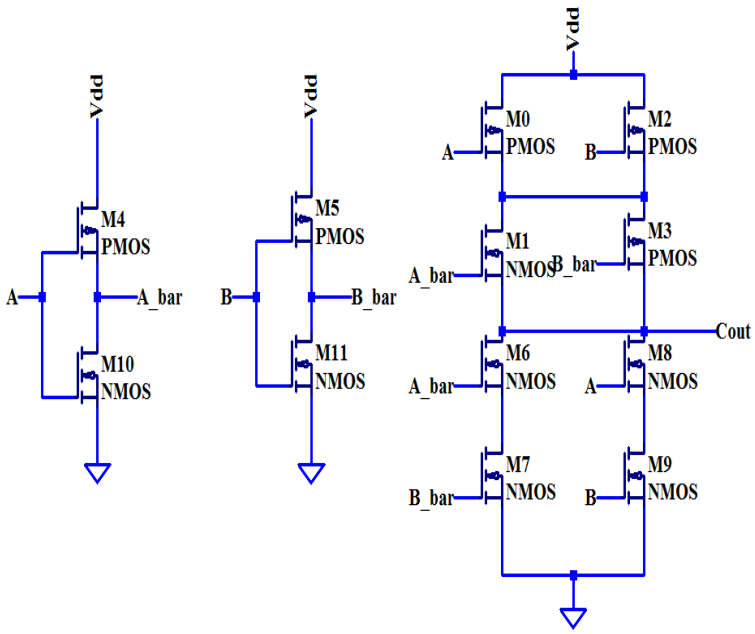


Figure 6.15: The XOR gate transistor level schematic. XOR gate consists of 12 transistors.

6.3 Evolutionary Algorithm Verification

6.3.1 Benchmark Circuit I: XOR

In order to guarantee the correctness of circuit optimisation results obtained from the proposed EA algorithm, an XOR gate is used as a verification benchmark performing exhaustive testing and EA optimisation procedure to find best motif permutations. XOR gate circuit consists of 12 transistors (Figure 6.15 depicts the transistor level circuit schematic). XOR gate is the same as other logic gates constructed by two types of motifs: standard device and O shaped motif. Hence, the circuit combination has 4096 motif permutations that is moderate-sized scale without costing significant amounts of computation and simulation time for either the exhaustive tests or evolutionary algorithm. The exhaustive test results provide references of the best motif solution to verify whether EA can find the same best solutions. XOR gate exhaustive tests are the same as other logic gate simulations, where each motif permutation has 300 samples with different LER information performs statistical variability simulation. Figure 6.16 illustrates the XOR gate exhaustive test results. The index of motif permutations from 3840 to 4095 have relatively better mean value and standard deviation

of propagation delay than other motif permutation. Figure 6.17 indicates the best solutions found by the EA. All transistors of the XOR gate have fixed device size in the exhaustive test. Therefore, only various motif permutations with fixed device size are evolved by the EA. A predetermined number of evolution parameters are listed in Table 6.11. Best solutions found by the exhaustive test and EA locate in the same range of index of motif permutation. This conclusion proves that the EA is a reliable approach for finding appropriate solutions, particularly with the exhaustive test being no longer efficient for large circuits. In addition, although both exhaustive testing and EA approach use around five days to complete solutions searching, the use of EA approach finding out best solution is faster than exhaustive testing because the fitness value of EA has been already convergence before reaching the maximum generation.

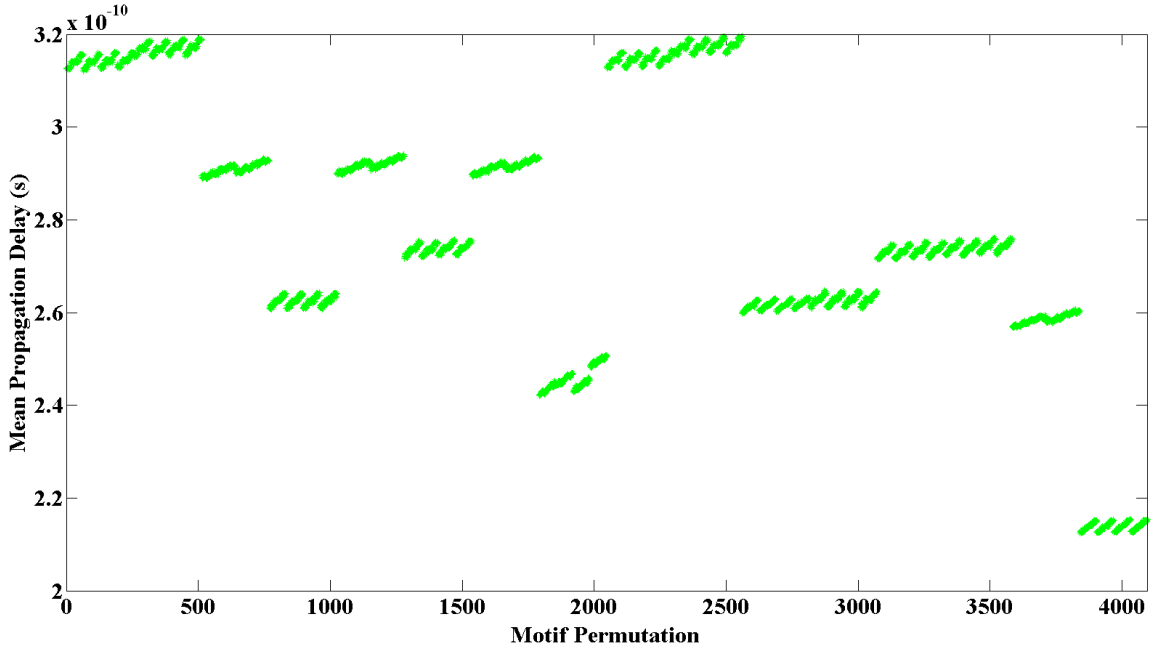
Table 6.11: EA evolution parameter in XOR gate optimisation. Although two objectives (mean value of propagation delay and standard deviation of propagation delay) are optimised in this experiment, only motif permutation is evolved in order to fairly compare the exhausting test results because the exhausting test only test different motif permutations not considering different device size.

Parameter	Value
Population Size	20 (1 + 19)
Chromosome Length	12
Number of generations	250
Objectives	2 (mean & Standard deviation)

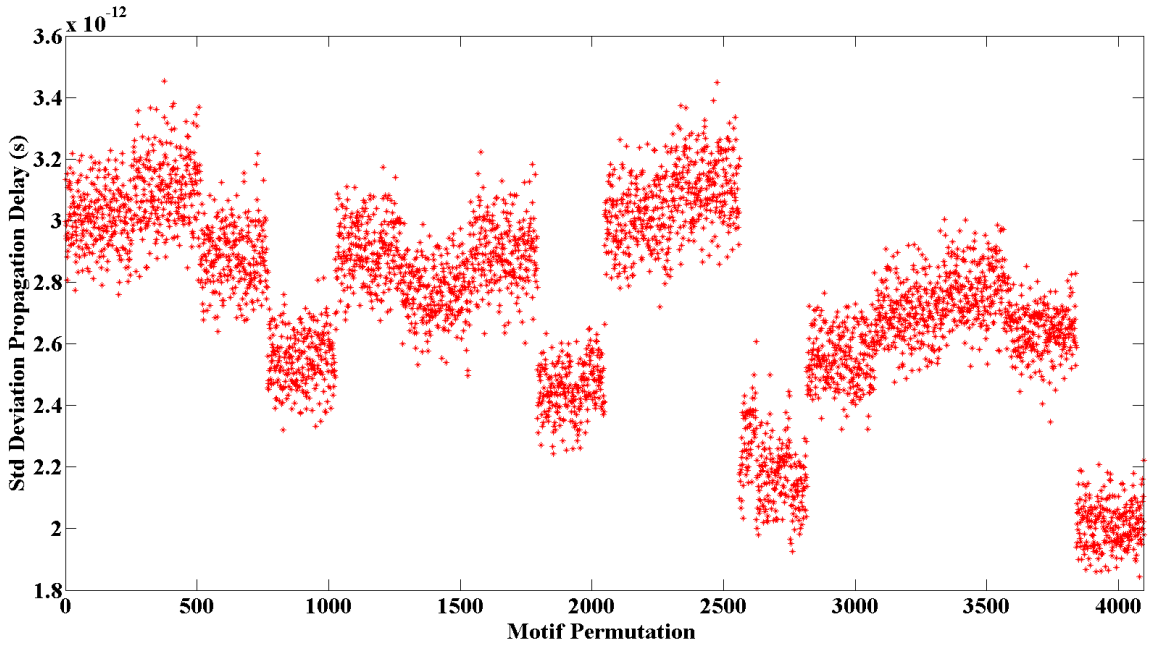
6.4 Benchmark Circuits Evaluation

6.4.1 Benchmark Comparison Circuits

Two more complex circuits, a half adder and full adder (transistor level circuit schematics are shown in Figures 6.18 and 6.19) are used as benchmark circuits to be optimised. In order to investigate the effect of motifs on circuit performance and verify whether different motif permutations can bring benefits for circuit optimisation, two reference solutions are built separately in the first stage. One reference circuit is constructed using only standard devices, and another only uses O shaped motif. The evolved results obtained from first stage optimisation can be compared with these reference circuits, and are also directly passed to

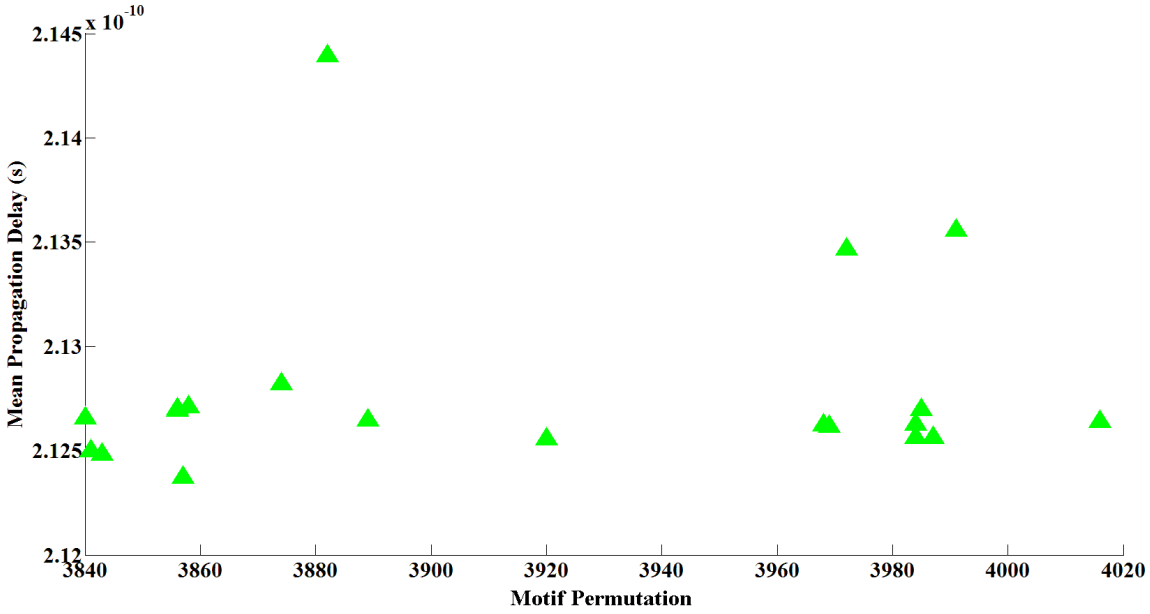


(a)

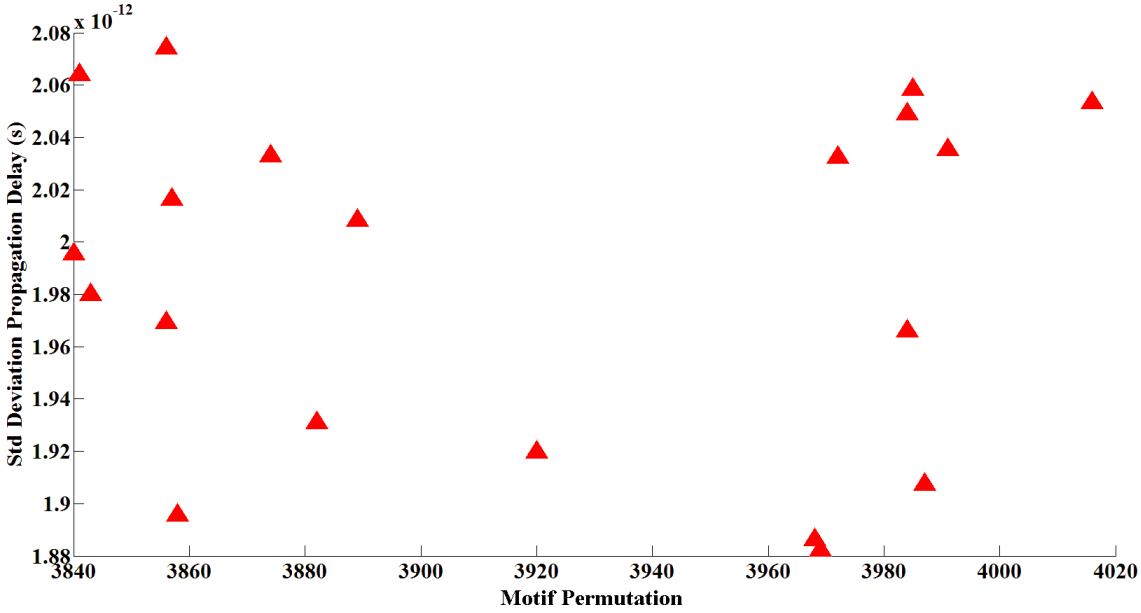


(b)

Figure 6.16: XOR gate exhaustive test results. (a) mean value of propagation delay vs. motif permutations. (b) standard deviation of propagation delay vs. motif permutations. Compared sub-figure (a) and (b) the motif permutation's index from 3840 to 4095 have both better the mean value and standard deviation than other motif permutations.



(a)



(b)

Figure 6.17: XOR gate EA optimised results. (a) mean value of propagation delay vs. motif permutations. (b) standard deviation of propagation delay vs. motif permutations. Compared to results found by exhaustive test shown Figure 6.16, best solutions found by the EA are located in the same range of index motif permutation.

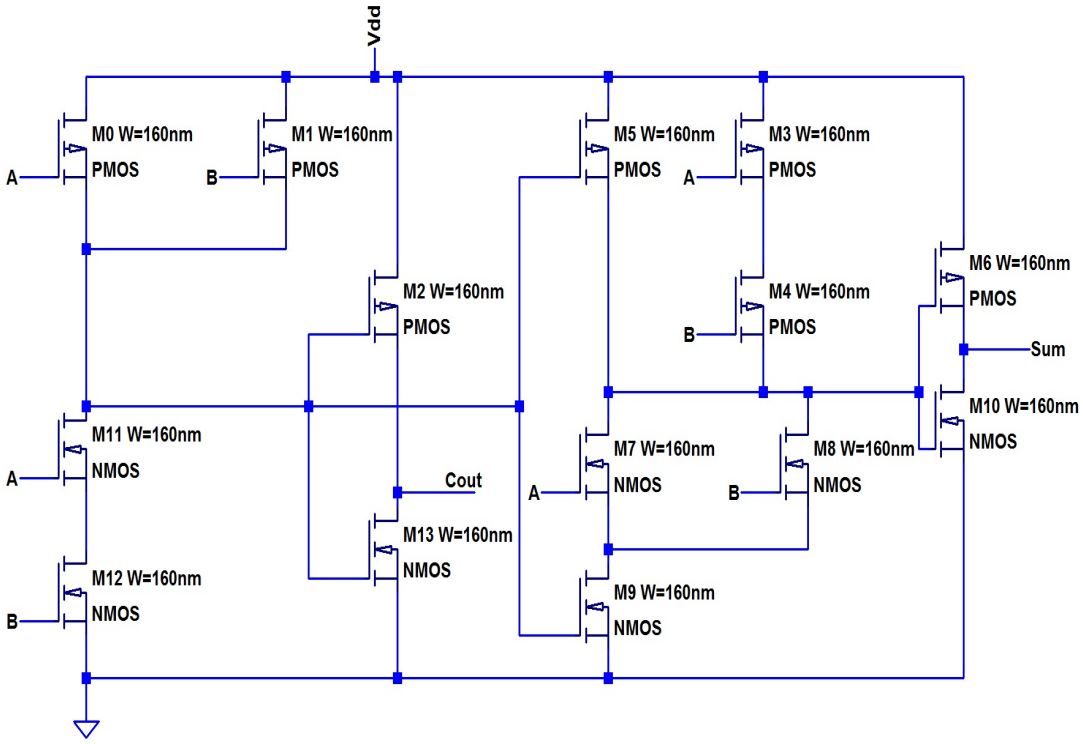


Figure 6.18: Standard half adder cell. A transistor level circuit schematic of complementary CMOS logic half adder that consists of 14 transistors.

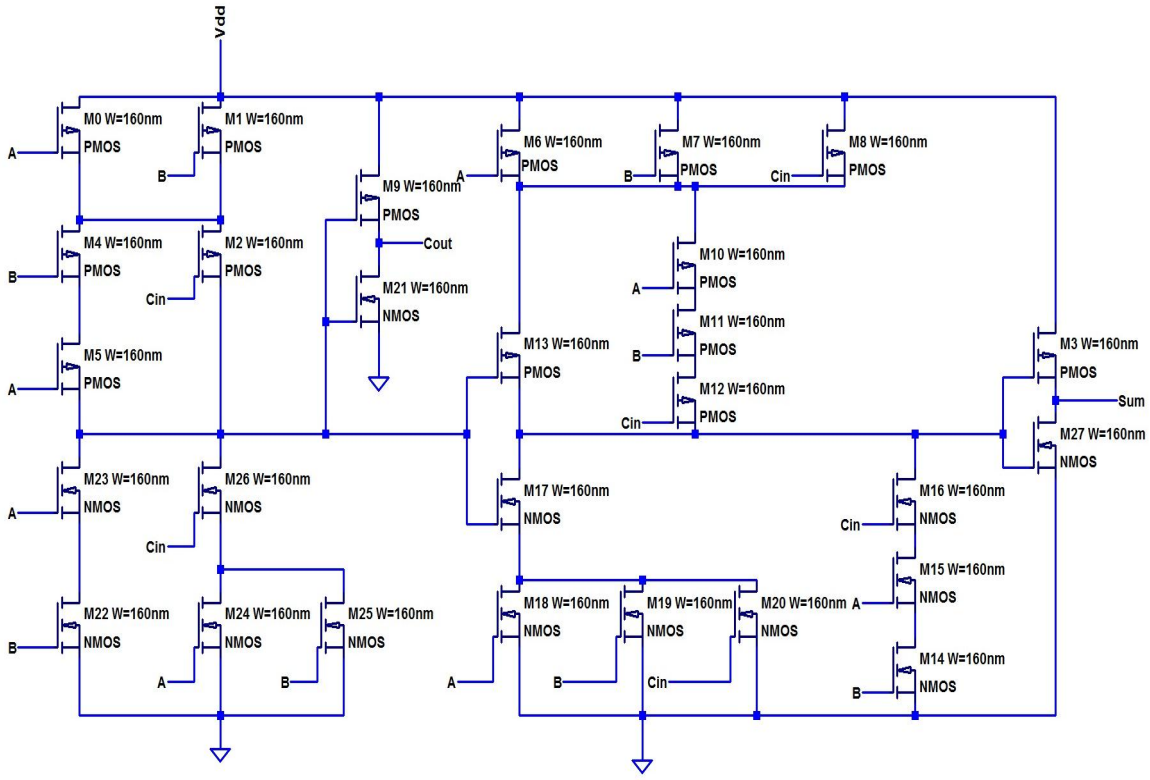


Figure 6.19: Standard full adder cell. A transistor level circuit schematic of 1-bit complementary CMOS logic full adder that consists of 28 transistors.

the second stage optimisation as a first generation genotype. The algorithm was run and the number of generations and corresponding evolution parameters are listed in Table 6.12. Additionally, because reference device and *O* shaped motif compact models are extracted with fixed width (reference device W=80nm and *O* device W=160nm), motifs with large size is generally replaced by the parallel small size motif structure in the circuit level design. In order to the large size motif is also reliable, the reference device with large size is verified in TCAD simulation result (W=160nm) and extracted small size device compact model with parallel connection (W=80nm, m=2). The error between TCAD simulation and extracted model is about 7%. This extracted error is acceptable for this experiment.

Table 6.12: Proposed EA Evolution parameter. Due to the computational overhead of performing the a large number of SPICE statistical sample simulation and multi-objective calculation, the corresponding evolution parameter is performed for a considerably larger number of the population size and shorter number of generations.

Stage	Parameter	Value
One	Population Size	20 (1 + 19)
	Genotype Length	14 (half adder) & 28 (full adder)
	Number of generations	250
	Objectives	1 (propagation delay)
Two	Population Size	40 (20 + 20)
	Genotype Length	14 (half adder) & 28 (full adder)
	Number of generations	50
	Objectives	2 (mean value propagation delay & standard deviation propagation delay)

6.4.2 Benchmark Circuit II: Half Adder

The comparison of the worst-case delay of half adder produced by the two reference solutions and first stage evolved solution are listed in Table 6.13. Evolved half adder benchmark circuit has obviously improved upon the reference designs. The worst-case delay of the half adder has been improved by 26.5% over the reference solution which only uses regular devices, and by 5.5% over the *O* shaped motif solution. The evolved benchmark circuit is depicted in Figure 6.20. These results illustrate that circuit performance improvement can be achieved through appropriate selecting motif permutations to construct a circuit.

In order to optimise circuits which have both good worst-case propagation delays, and are also

Table 6.13: Comparison of evolved half adder solution and reference solutions

Design	Circuit combination	Worst-case delay
Ref 1 (Standard device)	0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0	3.585609e-10
Ref 2 (O shaped motif)	1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1	2.788188e-10
Evolved design	1 : 1 : 0 : 1 : 0 : 0 : 1 : 1 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0	2.633979e-10

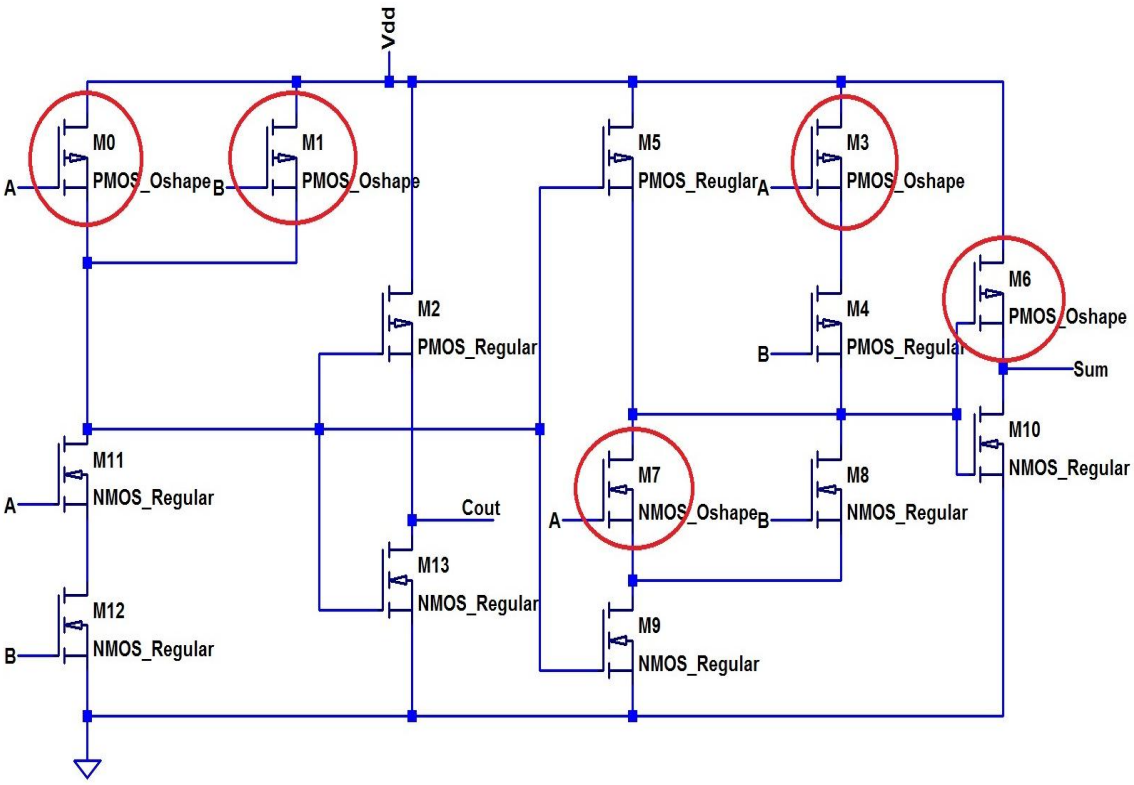


Figure 6.20: Transistor level evolved circuit schematic of half adder after the first stage optimisation. the evolved transistors replaced by O shaped motif are highlighted by red circle in the circuit.

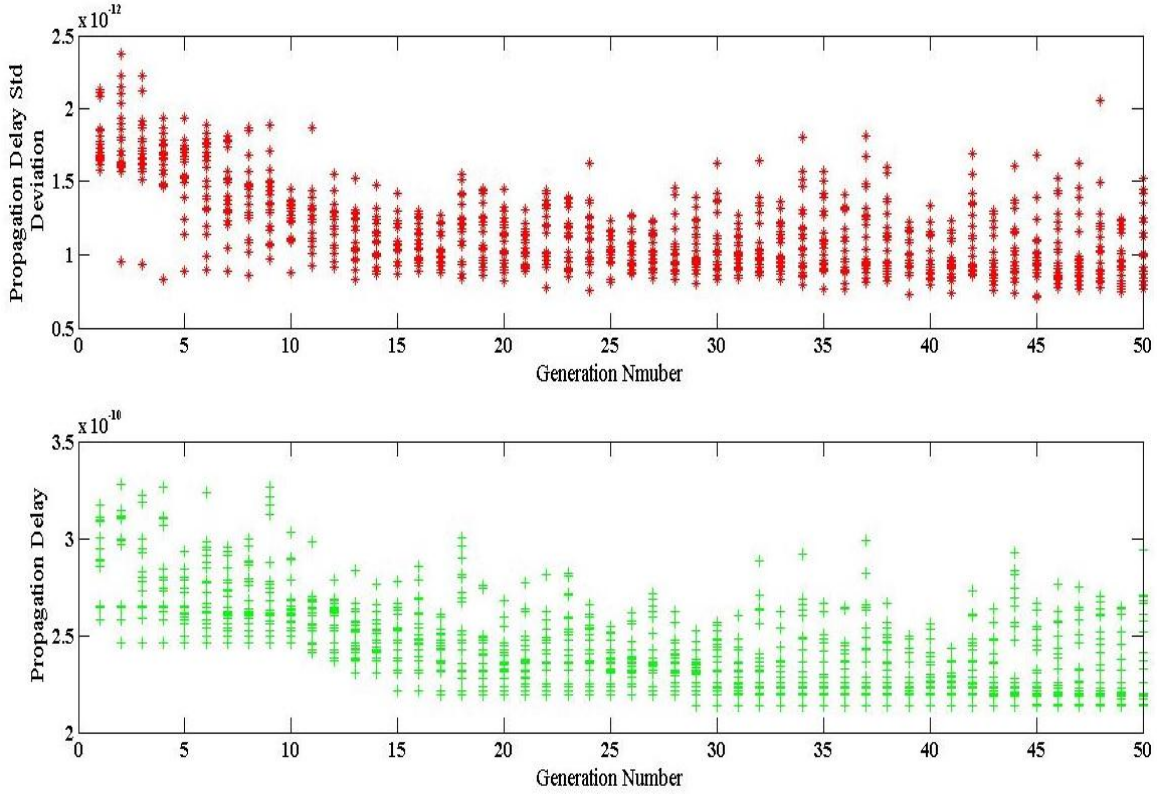


Figure 6.21: The trend of the effect of device variability on the evolved half adder circuit performance. Each star in figures represents an individual in the population. The upper sub-figure is the standard deviation of propagation delay vs generation number. The lower sub-figure is the mean value of propagation delay vs generation number.

robust in the presence of device atomistic variability, the best evolved solutions obtained from first stage optimisation are further optimised at the second stage based on motif width/drive current, and with fitness functions which consider both the mean value and standard deviation of the worst-case propagation delay. Figure 6.21 illustrates the second stage optimisation results obtained. The final half adder performance variability has been reduced comparing with solutions at the beginning of optimisation. Figure 6.22 shows that the best evolved results with highest rank sum value from final best solutions in the second stage optimisation.

6.4.3 Benchmark Circuit III: Full Adder

Table 6.14 shows the comparison of the evolved full adder design and reference designs, the performance of the full adder also has a significant improvement, increasing 30.1% over the reference solution only using standard device, and 5.8% while using O shaped motif. The corresponding evolved full adder design are shown in Figure 6.23. Similarly, for the sake of

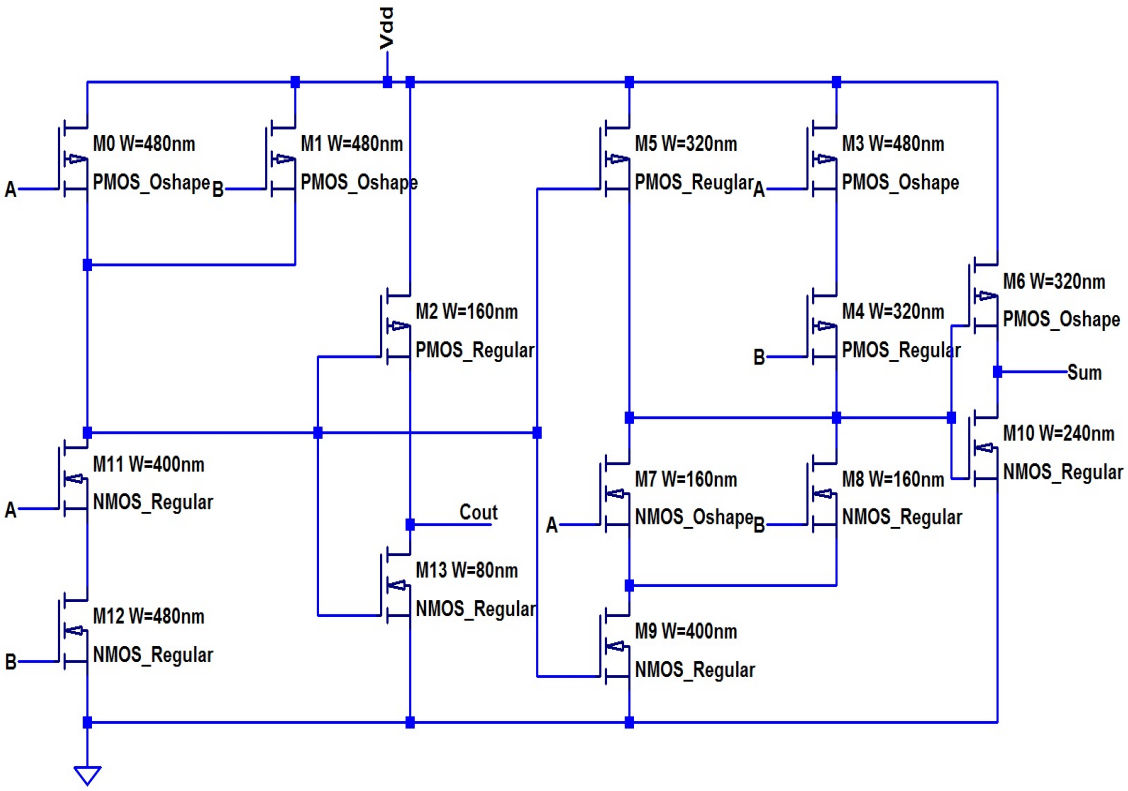


Figure 6.22: The final evolved half adder circuit has minimum effect of device variability on the circuit performance. The evolved circuit shows that the appropriate motif is applied in the specific position with evolved geometry size.

Table 6.14: Comparison of evolved 1-bit full adder solution and reference solutions

Design	Circuit combination	Worst-case delay
Ref 1 (Regular device)	0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0	6.984966e-10
Ref 2 (O shape device)	1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1	5.178464e-10
Evolved design	1 : 1 : 0 : 1 : 1 : 1 : 1 : 0 : 0 : 0 : 1 : 1 : 0 : 0 : 1 : 0 : 0 : 1 : 1 : 0 : 0 : 0 : 1 : 0 : 0 : 0 : 0 : 0	4.879261e-10

Table 6.15: A group of the final best results based on single gene mutation in 50 simulation runs

No.	Final Best Circuit Combinations	Worst-Case Delay	Rank
1	1 : 1 : 0 : 1 : 1 : 1 : 1 : 0 : 0 : 0 : 1 : 1 : 0 : 0 : 1 : 0 : 0 : 1 : 1 : 0 : 0 : 0 : 1 : 0 : 0 : 0 : 0 : 0	4.879261e-10	1
2	1 : 1 : 0 : 1 : 1 : 1 : 1 : 0 : 0 : 0 : 1 : 1 : 0 : 0 : 1 : 0 : 0 : 1 : 0 : 1 : 0 : 0 : 1 : 0 : 0 : 0 : 0 : 0	4.879283e-10	2
3	1 : 1 : 0 : 1 : 1 : 1 : 1 : 0 : 0 : 0 : 1 : 0 : 1 : 0 : 1 : 0 : 0 : 1 : 0 : 1 : 0 : 0 : 1 : 0 : 0 : 0 : 0 : 0	4.882526e-10	3

minimizing the effect of variation on circuit performance, the best full adder evolved solution is optimised again in the second stage optimisation. Figure 6.24 depicts the optimised results. The circuit propagation delay variability has obviously improved, especially, as the scale of circuit enlarging, the improvement become obviously. The final best evolved solution of the full adder with highest rank sum value is depicted in Figure 6.25.

Additionally, an EA using single gene mutation has also been simulated and investigated in the full adder of first stage optimisation in order to demonstrate the advantage of dynamic gene mutation escaping local optimum. Table 6.15 lists a group of the final best solutions found by EA based on single gene mutation that come from different runs. Each of solutions is given a rank according to its worst-case delay. The hamming distance of the structure to both the individual on rank 1 and 2 is two. This means motif permutation does not get from the second individual to the first individual through changing one gene value. The drawback is easily overcome by the dynamic gene mutation as result of its naturally performing multiple genes. Hence, dynamic gene mutation is adopted in proposed multi-stage EA.

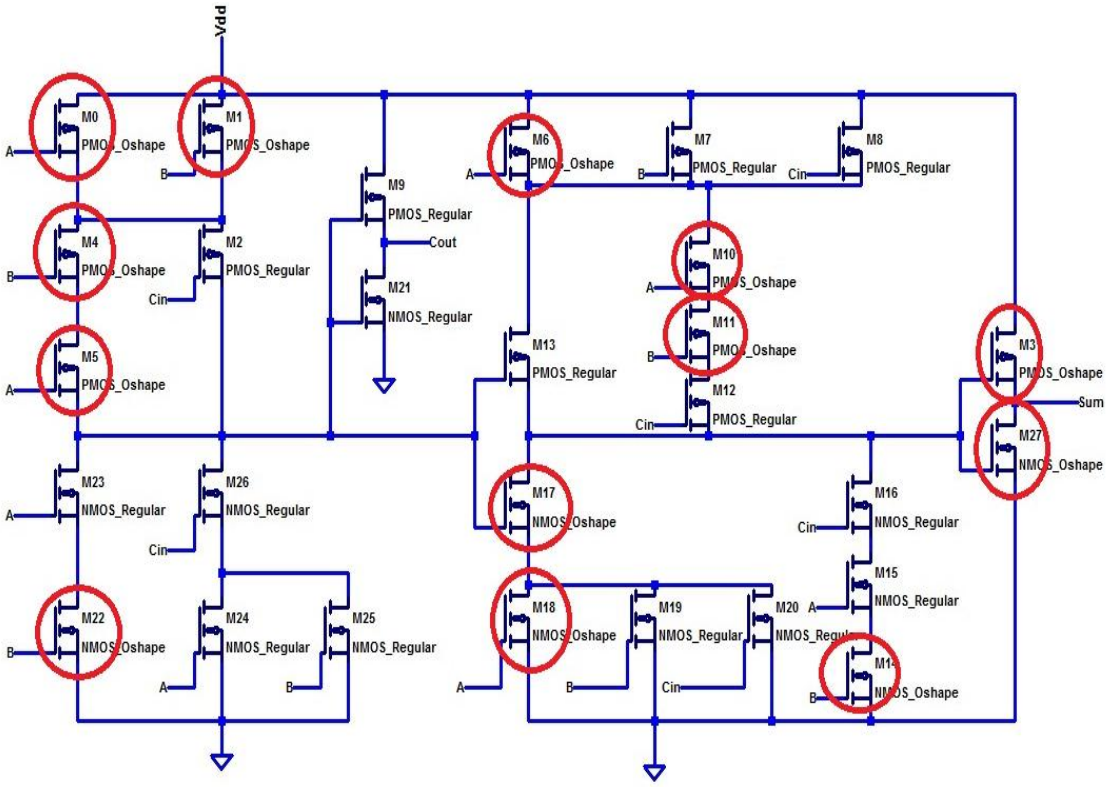


Figure 6.23: Transistor level evolved circuit schematic of full adder after the first stage optimisation. The evolved transistors replaced by O shaped motif in the circuit are highlighted by red circle.

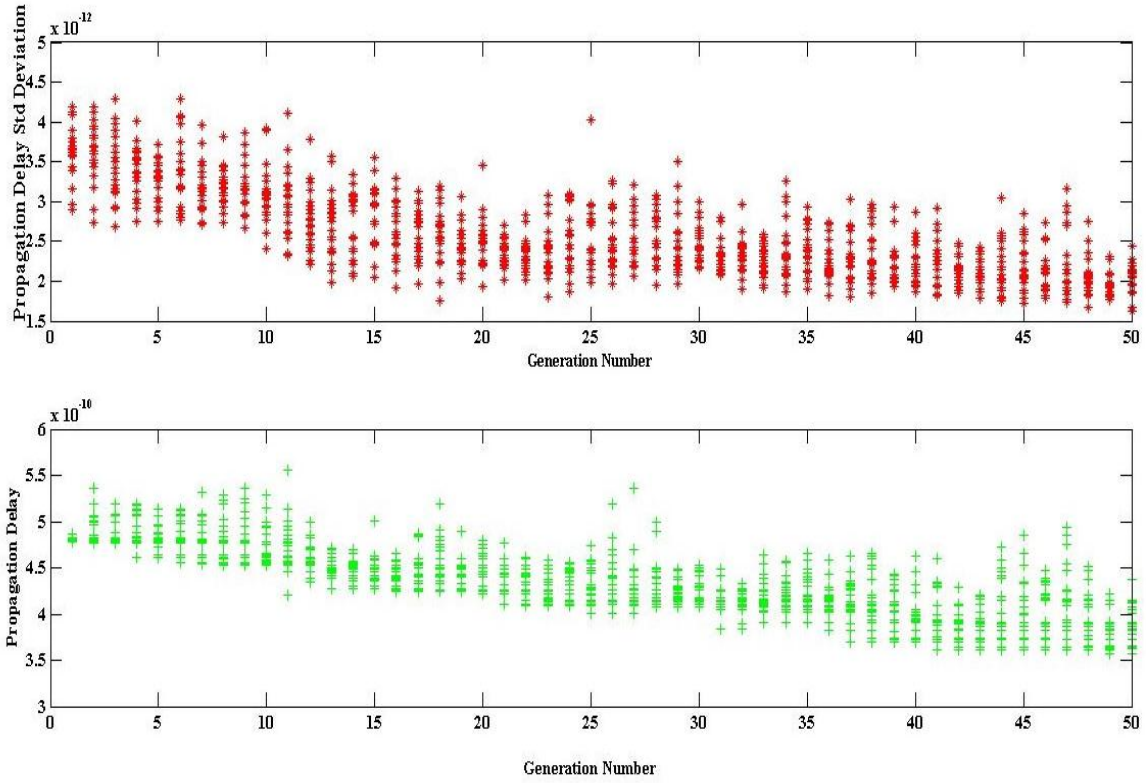


Figure 6.24: The trend of the effect of device variability on the evolved full adder circuit performance. Each point represents an individual in the population. The upper sub-figure is the standard deviation of propagation delay vs generation number. The lower sub-figure is the mean value of propagation delay vs generation number. Both the standard deviation and mean value of propagation delay have been decreased as generation number increasing. These results prove that proposed optimisation method effectively minimise the effect of device variability on test circuit performance.

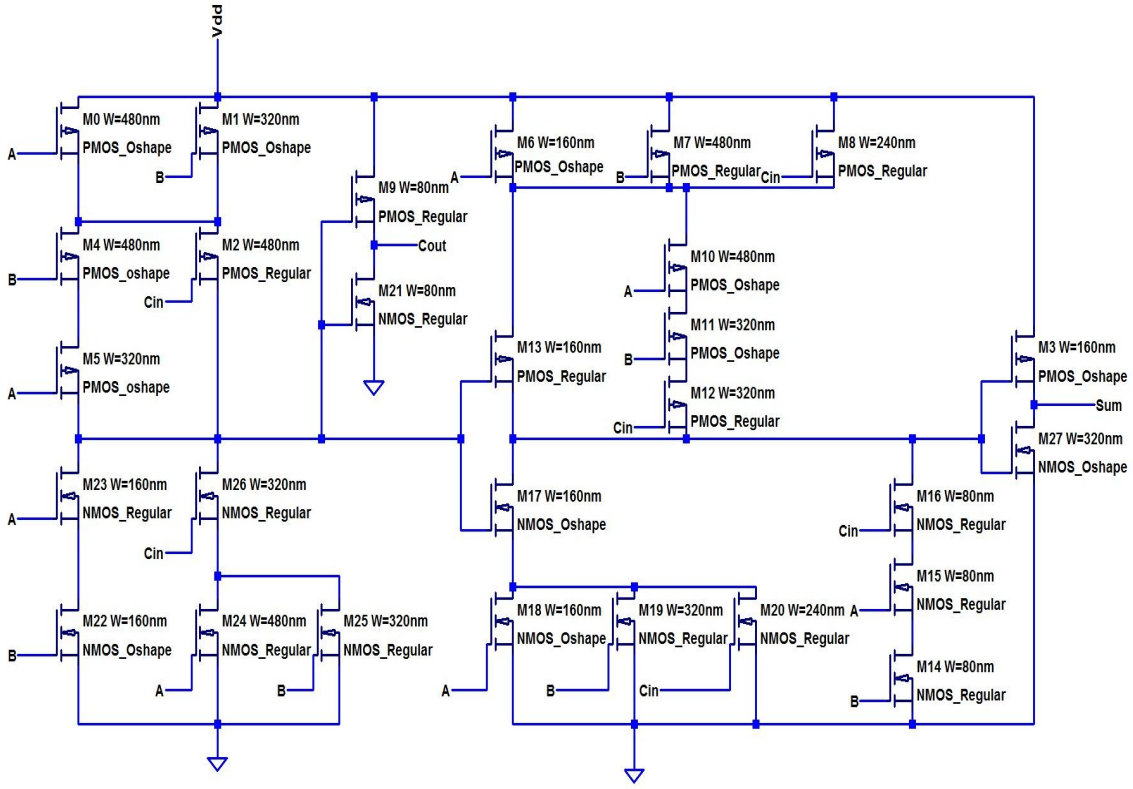


Figure 6.25: The final evolved full adder circuit has minimum the effect of device variability on the full adder circuit performance. As the number of transistor in the test circuit increasing, the evolved full adder circuit shows more complex circuit motif combination. This suggests that the proposed optimisation methodology will be advantages when performing large scale circuit performance optimisation.

6.5 Summary

In this chapter, a multi-stage EA approach for circuit performance optimisation using various motif permutations has been discussed. Firstly, a set of logic gate NAND, NOR, OR, and AND constructed by different motif permutations were exhaustively tested to investigate the influence of various motif permutations on the circuit performance and its variability. The logic gate simulation results show that circuit performance and its variability is improved through using specific motif permutations. However, as the number of transistors in the circuit increases, the corresponding motif permutations also exponentially increase, resulting in exhaustive tests becoming infeasible. In order to resolve this drawback of the exhaustive test and improve the efficiency of finding the best motif permutation to achieve circuit optimisation, a multi-stage evolutionary algorithm with dynamic mutation mechanism has been proposed in this chapter. The proposed optimisation algorithm handles the circuit performance and its variability optimisation at different stages. The idea behind this approach is based on step-wise optimisation (described in Chapter 4) that attempts to decompose the circuit performance optimisation and its variability minimisation into different stages so that such problems can be performed more easily. A XOR gate is used as benchmark circuit to verify whether the EA can find optimal solutions by comparing with solutions found by the exhaustive tests. The comparison results suggest that the EA is a promising approach. Finally, two additional circuits: a half adder and a full adder, are assessed for circuit speed improvement and variability tolerance, with both evolved results demonstrating a significant improvement over the original design. Additionally, benefits of using dynamic gene mutation rather single gene mutation are confirmed. The following final chapter of this thesis contains concluding remarks about the experiments conducted and potential future work.

Chapter 7

Conclusions and Future Work

Contents

This thesis developed a methodology for combining evolutionary algorithms with motif-based circuit simulations, to optimise the design and performance of digital circuits. The methodology was demonstrated on bulk CMOS standard cells a technology that continues to have considerable importance to the semiconductor industry. However, the methodology has applicability for other technologies, simply dependant on the choice of circuit motifs and models used. This novel approach automatically searches for the best permutation of motifs in order to optimise circuit speed (worst-case delay); and further enhancing to mitigate against the effects of atomistic device variability through evolving motif widths/drive currents. The delay measurements are performed using an open source SPICE circuit simulator in the evolution loop. These methodology exhibits significant optimisation capability in minimising worst-case delay and diminishing the effects of device variability on circuit performance.

7.1 Conclusions

At the beginning of this thesis, the hypothesis was stated: "circuit performance improvement and variability mitigation can be achieved through a combination of evolutionary algorithms and device layout motifs."

A novel methodology was developed using novel layout motifs to tackle the problem of atomistic variability at circuit level and combining them with evolutionary algorithms to achieve

circuit performance optimisation. From the evidence provided in Chapter 6 of this thesis, it is possible to conclude that the hypothesis has been supported, as the performance of the half adder and full adder using mixed motif permutation have been improved by at least 25% improved. The standard deviation of propagation delay has been decreased by at least 19% via the proposed optimisation algorithm (based on Figures 6.22 and 6.25 shown results), which achieved the goal for variability mitigation. The benchmark circuit experiments proved that EA is a reliable and effective way to find out optimal solutions.

The primary focus of the research in this thesis has focussed on developing a bottom-up methodology, which can directly reflect the device atomistic information at circuit level. Thus, the device layout and its atomistic information at the device level, device layout motif was proposed to know relationship between the device layout and its atomistic information at the device level. However, it is impossible to test and investigate every device layouts due to endless layout configuration. Thus, only several device layout motifs were investigated; and we found different layout geometry configuration actually affect device characteristics (which results addressed in Chapter 3). Further work in investigating various device layout motifs need to continue. The more device layout motifs were investigated, the more details of various device layout motifs were obtained. These layout motifs will provide more choices when used to build circuits.

Additionally, the compact model brings the device layout motif into circuit level design. The two-step EA extraction algorithm proposed to extract compact model and overcome the conventional method easily fell in poor convergence without good initial value. Because the number of optimised model parameters is less than these parameters optimised in industry extraction tool, the outcome extracted by proposed method are not better than industry extraction tool. However, the accuracy of extraction results will further improve and increase as the more model parameters are considered and optimised. The more accuracy compact model can be extracted in future work.

7.2 Summary

Chapter 2 gives a comprehensive review of the basic concepts of scaling and the challenges of MOSFET device evolution. Starting with the concept of device scaling, generic rules of geometrical-driven scaling principle applying to MOSFET physical dimensions, doping concentration and electrical features were introduced. Although devices achieve the performance

and density bonus due to device dimensions linearly scaling, some challenges, associated with the lithography limitation, short-channel effects due to small device features and stochastic variability caused by the fundamental discrete natural of charge and matter, have to be faced as devices scale to atomic levels. Stochastic variations are inevitable as is the ever increasing process complexity. Hence, this work mainly focuses on mitigating the effects of stochastic variability on circuit performance. In depth understanding of variability classification, sources of variability, and their influence on circuit design performance is crucial for an understanding of the use of motifs on variation-aware circuit design in this thesis.

In Chapter 3, the concept of motifs and corresponding details of statistical 3D simulation and analysis were introduced. Motifs, defined as fundamental geometrical forms used as design units, provide the advantage to examine the problem of atomistic variability on fine-grained common repeating design units, whereby motifs are generic schemes that suite any repeating sub-structure in device structure and circuit layout. In this work, motifs specify a set of layout patterns of either single transistors or closely coupled clusters transistor-groupings from which wider devices and more complex circuits can be constructed. Since full scale 3D simulation is critical to the analysis of realistic devices, device fabrication steps were reviewed at the beginning of this chapter which provides the necessary background knowledge for an understanding of device structure emulation through TCAD tools. A group of full scale 3D simulations for each type of motif were carried out in this work via TCAD tools. Simultaneously, in order to investigate the interplay between statistical variability and various motif structures, line edge roughness was introduced into 3D motif simulations. Finally, motif simulations employed statistical analysis through the extraction of the distribution of threshold voltage and leakage current.

Chapter 4 covers an introduction to Evolutionary Algorithms (EAs) and evolutionary mechanism and related evolution operations to prepare for the application of EAs on VLSI design. Generally, VLSI design and its optimisation problems are complex and mutually dependent problems governed by a set of constraints. Evolutionary algorithms were considered to effectively handle these optimisation problems due to their ability to search large complex problem spaces and perform multi-objective optimisation. Special attention was paid to problem decomposition through incremental evolution and multi-stage optimisation process so that the complex and computationally heavy problems can be transformed to a set of simple sub-problems with lower computational requirements.

Based on the atomistic motifs simulation results obtained from the large scale simulation

of LER-induced performance variability in 300 samples in Chapter 3, the successful statistical motif compact model extraction process was introduced in Chapter 5. The statistical variability model extraction was divided into two parts: uniform model extraction without considering variability and statistical variability model library generation. Due to the drawback of conventional extraction methods related to poor convergence without a good initial value and high computational cost for multi-objective optimisation, a novel two step evolutionary algorithm (2SEA) is proposed to assist in obtaining good uniform model extraction results even with not optimal initial conditions. The details of evolutionary operations used in this approach were described, including selection, mutation and evaluation. The error of final motifs model fitting to TCAD device simulation is 6%∼7% for both n- and p-channel devices. Subsequently, statistical variability compact models are generated based on the previously obtained uniform compact models. For the sake of simplicity only threshold voltage is considered to generate variations in these experiments. The threshold voltage parameter is assumed to have a Gaussian distribution. A variation value, based on a Gaussian distribution is injected into the uniform motif compact model, enabling variability to be included in the motifs. The corresponding Gaussian distribution parameters were extracted from the distribution of TCAD device statistical simulation results. While the statistical compact models obtained, these statistical motifs compact models were used to construct circuit for performance optimisation.

In Chapter 6, the application of motifs to circuit performance optimisation was investigated. A set of basic logic gate circuits constructed by using different motifs were examined through exhaustive tests. Results show that specific motif permutations benefit circuit performance improvement. Two additional benchmark circuits, half adder and 1-bit full adder, were tested to prove that circuits can generally benefit from appropriate motif permutations. Exhaustive test is an efficient and fast method to examine small circuits. However, exhaustive test becomes infeasible, especially testing every possible motif permutation when motif permutations increase exponentially as the number of transistors in the circuit increases. Thus, a multi-stage evolutionary algorithm is proposed to solve this problem. In order to verify whether the EA can find the "right" solutions, an XOR gate was tested by EA and exhaustive test. Results show that the best motif permutations found by EA are the same as those found by the exhaustive test. This suggests that the proposed optimisation approach is a promising method. Half adder and 1-bit full adder evolved results showed that the approach of combining motifs and evolutionary algorithm exhibits significant optimisation capability in improving circuit performance (mainly circuit speed optimisation) and in minimising the

effects of device variability on circuit performance. Additionally, results with a mixture of motif permutations are shown to consistently achieve better performance than those of motif permutation using a single type motif. These results provide the evidence to support this thesis's hypothesis.

7.3 Future Work

There are several areas where the work presented here could be extended in the future. Firstly, the motifs investigation results and analysis within this thesis was obtained from *50 nm* process bulk CMOS device layout motifs. The *50 nm* technology was chosen as it is widely used in device research, and is well understood. However, advanced technologies such as *14nm/16nm* process technologies have already been used in the latest industry design. Thus, future work will be required by using these technologies to confirm suitability and effectiveness of motifs on the advanced technologies. According to motifs definition, any repeating pattern of fundamental geometrical forms in device/circuit can be seen as motifs. In this thesis, only device layout motifs were investigated and so future work will extend motifs study. Standard cell motifs with repeating sub-units could be planned and investigated. The accuracy of the influence of intrinsic-variability effects on motifs characteristic depends on the number of various variability sources they include, with this study only including the LER-induced variability. Many other variation sources in realistic devices such as RDF-induced variability, lithography-induced variations and gate oxide thickness fluctuation will be worthy of considering and studying in the future for further increasing the accuracy of the simulated effects and revealing the influence of these sources on devices characteristics.

In this research, the naïve approach was applied to generate the statistical motif compact model. Only one key statistical parameter, threshold voltage, is used to perform statistical motifs model extraction, and this statistical variability model library consisted of 1000 model cards for each type of motif. Although this simple method saves computational expense and simplifies extraction process, the accuracy of statistical model decreases. Particularly, reliable variability-aware circuit design requires the support of accurate statistical compact model. Two useful extensions to this work in the future will be to increase the accuracy of statistical compact model. One improvement is to increase the predetermined statistical compact model library size to overcome the limitation on the statistical sample size. The second improvement is to select an optimum set of statistical parameters and inject variation based on the sensitivity analysis of I-V characteristic.

Whilst the focus of the optimisation objectives discussed in this thesis has been concerned primarily with propagation delay and its variability, there are other optimisation objectives such as power consumption, circuit layout area and yield which are also critical to circuit design. However, as the number of optimisation objective increases, the optimisation efficiency will become a major challenge used in this optimisation methodology. This requires further to study many-objective evolutionary technologies with modest changes to optimisation algorithm so as to find an appropriate manner to trade-off pursing perfect optimisation results and optimisation efficiency. Additionally, as this work described that the speed of logic gate can be determined by various motif permutations, this feature will provide a new optimisation branch in static timing analysis (STA). Applying various motifs in the critical path to modify the propagation delay of logic gates in the critical path achieves timing problem optimisation.

Special attention will be given to novel device structure, new standard cell design and new EDA tool in the future. Novel device structure/standard cell design methodology can be beneficial from motifs technique, which provides a detailed analysis of simple, repeating sub-units, and that the reusable datasets can then be used to infer the robustness and high-performance of designs at a higher level in the hierarchy. In future, device/circuit design may simultaneously work with standard motifs. Additionally, motif technique and different optimisation algorithms will be combined to develop new EDA tools to tackle the problems caused by semiconductor development.

Appendix A

NMOS Device TCAD Script

```
(sde:clear)
;=========================================
; PARAMETER DEFINITION
;=========================================
; Geometrical parameters:
(define Lg @Lg@) ; Gate length [um]
(define Lsp 0.08) ; Spacer width [um]
(define Lsd 0.2) ; Source/Drain width [um]
(define W @W@) ; (full) Device width [um]
(define Wt 0.12) ; Trench width [um]
(define Ro 0.01) ; Oxidation rounding radius [um]
(define Ht 0.3) ; Trench depth [um]
(define trench_angle 85) ; Trench angle [degrees]
(define Hs 0.5) ; Substrate height [um]
(define Hd 0.01) ; Divot height [um]
(define Hp 0.1) ; Poly height [um]
(define Tox 20e-4) ; Gate oxide thickness [um]
(define TextSpa 0.005) ; Extension spacer thickness [um]
(define TNiSpa 0.045) ; Nitride spacer thickness [um]
(define Hc 0.02) ; Contacts thickness [um]
;-----------------------
; Derived Dimensions:
```

```
(define ymax (+ (* Lg 0.5) Lsd Wt))
(define xmax W)
(define active_xmin Wt)
(define active_ymin Wt)
(define ygate (+ Wt Lsd))
(define TOxSpa (- Lsp (+ TNiSpa TextSpa)))
(define Ysp (- ygate Lsp))
(define Zc 0.0)
;————————————————
; Doping parameters:
(define Na 2e18) ; Body doping [cm-3]
(define Nd_po 1e20) ; Poly Doping [cm-3]
(define Nd_sd 1e20) ; Source/Drain Doping [cm-3]
(define Xj_sd 0.1) ; Source/Drain Junction Depth [um]
(define Nd_ex 2e19) ; SD Extension Doping [cm-3]
(define Xj_ex 0.02) ; SD Extension Junction Depth [um]

;=========================================
; PROCESS EMULATION
;=========================================
; Substrate
(sdegeo:create-cuboid (position 0 0 (* -1.0 Hs)) (position xmax ymax 0) "Silicon" "RSub-
strate" )
(sdegeo:create-cuboid (position 0 ymax (* -1.0 Hs)) (position xmax (* 2 ymax) 0) "Silicon"
"RSubstrate1" )

; Trench
(sdegeo:set-default-boolean "ABA")
(define GAS1 (sdegeo:create-cuboid (position 0 0 (* -1.0 Ht)) (position active_xmin ymax 0)
"Gas" "Dummy1" ))
(define GAS2 (sdegeo:create-cuboid (position 0 0 (* -1.0 Ht)) (position xmax active_ymin 0)
"Gas" "Dummy2" ))
(define GAS3 (sdegeo:create-cuboid (position 0 ymax (* -1.0 Ht)) (position active_xmin (* 2
```

```
ymax) 0) “Gas” “Dummy3” ))
(define GAS4 (sdegeo:create-cuboid (position 0 (* 2 ymax) (* -1.0 Ht)) (position xmax (- (* 2 ymax) active_ymin) 0) “Gas” “Dummy4” ))
(sdegeo:delete-region (list GAS1 GAS2 GAS3 GAS4))
(sdegeo:taper-faces (list (car (find-face-id (position active_xmin (- ymax 0.001) (* Ht -0.5)))) (car (find-face-id (position (- xmax 0.001) active_ymin (* Ht -0.5))))) (position active_xmin active_ymin 0.0) (gvector 0 0 1) (- 90 trench_angle))
(sdegeo:taper-faces (list (car (find-face-id (position active_xmin (+ ymax 0.001) (* Ht -0.5)))) (car (find-face-id (position (- xmax 0.001) (- (* 2 ymax) active_ymin) (* Ht -0.5))))) (position active_xmin (- (* 2 ymax) active_ymin) 0.0) (gvector 0 0 1) (- 90 trench_angle))

; Trench rounding
(sdegeo:fillet (list (car (find-edge-id (position active_xmin (- ymax 0.01) 0))) (car (find-edge-id (position (- xmax 0.01) active_ymin 0))) (car (find-edge-id (position (+ active_xmin (/ (* -0.5 Ht) (tan (degrees->radians trench_angle)))) (+ active_ymin (/ (* -0.5 Ht) (tan (degrees->radians trench_angle)))) (* -0.5 Ht ) ))) ) Ro)
(sdegeo:fillet (list (car (find-edge-id (position active_xmin (+ ymax 0.01) 0))) (car (find-edge-id (position (- xmax 0.01) (- (* 2 ymax) active_ymin) 0))) (car (find-edge-id (position (+ active_xmin (/ (* -0.5 Ht) (tan (degrees->radians trench_angle)))) (- (- (* 2 ymax) active_ymin) (/ (* -0.5 Ht) (tan (degrees->radians trench_angle)))) (* -0.5 Ht ) ))) ) Ro)
(sdegeo:bool-unite (find-material-id “Silicon”))

; Trench Fill + Gate Oxide
(sdegeo:set-default-boolean “BAB”)
(sdegeo:create-cuboid (position 0 0 (* -1.1 Ht)) (position xmax (* 2 ymax) Tox) “Oxide” “RTrench” )
(define PHOTO (sdegeo:create-cuboid (position active_xmin active_ymin (* -0.5 Hs)) (position xmax (- (* 2 ymax) active_ymin) 0.5) “Photoresist” “RMask1” ))
(sdegeo:create-cuboid (position 0 0 (* -0.5 Hs)) (position xmax (* 2 ymax) (+ Hd Tox)) “Oxide” “RTrenchTop” )
(sdegeo:delete-region PHOTO)
(sdegeo:bool-unite (find-material-id “Oxide”))
```

; Trench Oxide Rounding

(sdegeo:fillet (list (car (find-edge-id (position active_xmin (- ymax 0.01) (+ Hd Tox)))) (car (find-edge-id (position (- xmax 0.01) active_ymin (+ Hd Tox)))) (car (find-edge-id (position (- xmax 0.01) (- (* 2 ymax) active_ymin) (+ Hd Tox))))) Ro)

; Poly Gate

(sdegeo:create-cuboid (position 0.0 ygate (* -0.5 Hs)) (position xmax (- (* 2 ymax) ygate) (+ Hp Tox)) "PolySilicon" "RGate")

; Extension Spacer

(sdegeo:create-cuboid (position 0.0 (- ygate TextSpa) (* -0.5 Hs)) (position xmax (- (+ TextSpa (* 2 ymax)) ygate) (+ Hp Tox)) "Oxide" "RExtSpacer")
(sdegeo:bool-unite (find-material-id "Oxide"))

; Spacer

(sdegeo:create-cuboid (position 0.0 (- ygate TextSpa TOxSpa) (* -0.5 Hs)) (position xmax (- (+ TextSpa TOxSpa (* 2 ymax)) ygate) (+ Hp Tox)) "Oxide" "ROxSpacer1")
(sdegeo:create-cuboid (position 0.0 (- ygate TextSpa TOxSpa TNiSpa) (* -0.5 Hs)) (position xmax (- (+ TextSpa TOxSpa TNiSpa (* 2 ymax)) ygate) (+ TextSpa TOxSpa Tox)) "Oxide" "ROxSpacer2")
(sdegeo:bool-unite (find-material-id "Oxide"))
(sdegeo:create-cuboid (position 0.0 (- ygate TextSpa TOxSpa TNiSpa) (* -0.5 Hs)) (position xmax (- (+ TextSpa TOxSpa TNiSpa (* 2 ymax)) ygate) (+ Hp Tox)) "Nitride" "RNiSpacer")

; Spacer rounding

(sdegeo:fillet (list (car (find-edge-id (position (- xmax 0.001) (- ygate TextSpa TOxSpa TNiSpa) (+ Hp Tox)))) (car (find-edge-id (position (- xmax 0.001) (- (+ TextSpa TOxSpa TNiSpa (* 2 ymax)) ygate) (+ Hp Tox))))) (* TNiSpa 0.8))

; Contacts

(sdegeo:set-default-boolean "ABA")
(sdegeo:create-cuboid (position (+ active_xmin 0.005) (+ active_ymin 0.005) -0.005) (position

```
xmax (- ygate TextSpa TOxSpa TNiSpa) (+ Hc Tox)) "Metal" "RSourceContact" )
(sdegeo:create-cuboid (position (+ active_xmin 0.005) (- (+ TextSpa TOxSpa TNiSpa (* 2
ymax)) ygate) -0.005) (position xmax (- (* 2 ymax) active_ymin 0.005) (+ Hc Tox)) "Metal"
"RDrainContact" )
(sdegeo:create-cuboid (position 0.0 ygate Hp) (position xmax (- (* 2 ymax) ygate) (+ Hp
Tox +0.005)) "Metal" "RGateContact" )

;=========================================
; CONTACTS DEFINITION
;=========================================

(sdegeo:define-contact-set "gate" 4.0 (color:rgb 1.0 0.0 0.0 ) "##" )
(sdegeo:define-contact-set "source" 4.0 (color:rgb 1.0 0.0 0.0 ) "==" )
(sdegeo:define-contact-set "drain" 4.0 (color:rgb 1.0 0.0 0.0 ) "||" )
(sdegeo:define-contact-set "bulk" 4.0 (color:rgb 1.0 0.0 0.0 ) "<><>" )

(sdegeo:set-current-contact-set "bulk")
(sdegeo:set-contact-faces (find-face-id (position (- xmax 0.01) (- ymax 0.01) (* -1 Hs)))
"bulk")

(sdegeo:set-current-contact-set "gate")
(sdegeo:set-contact-boundary-faces (find-region-id "RGateContact"))
(sdegeo:delete-region (find-region-id "RGateContact"))

(sdegeo:set-current-contact-set "source")
(sdegeo:set-contact-boundary-faces (find-region-id "RSourceContact"))
(sdegeo:delete-region (find-region-id "RSourceContact"))

(sdegeo:set-current-contact-set "drain")
(sdegeo:set-contact-boundary-faces (find-region-id "RDrainContact"))
(sdegeo:delete-region (find-region-id "RDrainContact"))
```

```
;=========================================
; DOPING INFORMATION
;=========================================
(sedr:define-constant-profile "Bbody" "BoronActiveConcentration" Na)
(sedr:define-constant-profile-material "Bbody_PL" "Bbody" "Silicon")
(sedr:define-constant-profile "AsPoly" "ArsenicActiveConcentration" Nd_po)
(sedr:define-constant-profile-material "AsPoly_PL" "AsPoly" "PolySi")

(sedr:define-refinement-window "SImp" "Rectangle" (position 0.0 0.0 Zc) (position (* 1.5 xmax) Ysp Zc) )
(sedr:define-refinement-window "SextImp" "Rectangle" (position 0.0 0.0 Zc) (position (* 1.5 xmax) ygate Zc) )
(sedr:define-refinement-window "DImp" "Rectangle" (position 0.0 (- (+ TextSpa TOxSpa TNiSpa (* 2 ymax)) ygate) Zc) (position (* 1.5 xmax) (* 2 ymax) Zc) )
(sedr:define-refinement-window "DextImp" "Rectangle" (position 0.0 (- (* 2 ymax) ygate) Zc) (position (* 1.5 xmax) (* 2 ymax) Zc) )
(sedr:define-gaussian-profile "SDext" "ArsenicActiveConcentration" "PeakPos" 0 "PeakVal" Nd_ex "ValueAtDepth" Na "Depth" Xj_ex "Gauss" "Factor" 0.3)
(sedr:define-gaussian-profile "SD" "PhosphorusActiveConcentration" "PeakPos" 0 "PeakVal" Nd_sd "ValueAtDepth" Na "Depth" Xj_sd "Gauss" "Factor" 0.3)
(sedr:define-analytical-profile-placement "Source_PL" "SD" "SImp" "Symm" "NoReplace" "Eval")
(sedr:define-analytical-profile-placement "SourceExt_PL" "SDext" "SextImp" "Symm" "NoReplace" "Eval")
(sedr:define-analytical-profile-placement "Drain_PL" "SD" "DImp" "Symm" "NoReplace" "Eval")
(sedr:define-analytical-profile-placement "DrainExt_PL" "SDext" "DextImp" "Symm" "NoReplace" "Eval")

;=========================================
; MESHING INFORMATION
;=========================================
(sedr:define-refinement-window "Sub_RW" "Cuboid" (position 0.0 0.0 -1.0) (position xmax
```

(* 2 ymax) -0.4)) (sdedr:define-refinement-size "Sub_RD" (/ W 2) (/ ymax 2) 0.1 (/ W 4) (/ ymax 4) 0.05)
(sdedr:define-refinement-placement "Sub_PL" "Sub_RD" "Sub_RW")
(sdedr:define-refinement-window "Tr_RW" "Cuboid" (position 0.0 0.0 -0.4) (position xmax (* 2 ymax) (* 1.1 Hp))) (sdedr:define-refinement-size "Tr_RD" (/ W 2) (/ ymax 2) 0.05 (/ W 4) (/ ymax 4) 0.025)
(sdedr:define-refinement-placement "Tr_PL" "Tr_RD" "Tr_RW")
(sdedr:define-refinement-function "Tr_RD" "MaxLenInt" "PolySilicon" "Oxide" 0.001 4.0)
(sdedr:define-refinement-window "SD_RW" "Cuboid" (position (- active_xmin 0.01) active_ymin (* -1.2 Xj_sd)) (position xmax (- (* 2 ymax) active_ymin) 0.01))
(sdedr:define-refinement-size "SD_RD" (/ W 4) (/ ymax 4) 0.025 (/ W 64) (/ ymax 32) 0.0125)
(sdedr:define-refinement-placement "SD_PL" "SD_RD" "SD_RW")
(sdedr:define-refinement-function "SD_RD" "DopingConcentration" "MaxTransDiff" 1.0)
(sdedr:define-refinement-window "Ext_RW" "Cuboid" (position (- active_xmin 0.01) Ysp (* -2 Xj_ex)) (position xmax (- (* 2 ymax) Ysp) 0.01))
(sdedr:define-refinement-size "Ext_RD" (/ W 4) (/ ymax 32) 0.0125 (/ W 64) (/ ymax 128) 0.0025)
(sdedr:define-refinement-placement "Ext_PL" "Ext_RD" "Ext_RW")
(sdedr:define-refinement-function "Ext_RD" "DopingConcentration" "MaxTransDiff" 1.0)
(sdedr:define-refinement-window "Ch_RW" "Cuboid" (position (- active_xmin 0.01) (- ygate (* Lg 0.1)) (* -1.2 Xj_ex)) (position xmax (- (+ (* 2 ymax) (* Lg 0.1)) ygate) 0.01))
(sdedr:define-refinement-size "Ch_RD" (/ W 8) (/ ymax 32) 0.005 (/ W 64) (/ ymax 256) 0.001)
(sdedr:define-refinement-placement "Ch_PL" "Ch_RD" "Ch_RW")
(sdedr:define-refinement-function "Ch_RD" "MaxLenInt" "Silicon" "Oxide" 0.00025 2.0)

; Offset-Global
(sdenoffset:create-global "usebox" 2 "maxangle" 150 "maxconnect" 1000000 "background" "" "options" "" "triangulate" 0 "recoverholes" 0 "hlocal" 0 "factor" 1.5 "subdivide" 0 "terminateline" 3 "maxedgelength" 0.05 "maxlevel" 5)
(sdenoffset:create-noffset-interface "material" "Silicon" "Oxide" "hlocal" 0.0001 "factor" 1.5 "window" 0.0 Ysp -0.1 xmax (- (* 2 ymax) Ysp) 0.0)
(sdenoffset:create-noffset-interface "material" "Oxide" "Silicon" "hlocal" 0.0005 "factor" 2.0

“window” 0.0 Ysp -0.1 xmax (- (* 2 ymax) Ysp) 0.0)

(sdenoffset:create-noffset-interface “material” “PolySilicon” “Oxide” “hlocal” 0.0005 “factor” 2.0)

;————————————————————————————-

; Build Mesh

(sde:build-mesh “snmesh” “ ” “n@node@_msh”)

Appendix B

Model Parameters Definition

Table B.1: Variables used in Modeling Threshold Voltage

Equation Variable	Description
VTH0	long channel threshold voltage at Vbs = 0
K1	first-order body effect coefficient
K2	second-order body effect coefficient
K3	narrow width coefficient
K3B	Body effect coefficient of K3
W0	narrow width parameter
LPE0	lateral non-uniform doping parameter at Vbs = 0
LPEB	lateral non-uniform doping effect on K1
TOXE	electrical gate equivalent oxide thickness
TOXM	Gate oxide thickness at which parameters are extracted
DSUB	DIBL coefficient exponent in subthreshold region
ETA0	DIBL coefficient in the subthreshold region
ETAB	body-bias for the subthreshold DIBL effect
Φ_s	surface potential
DVT0	first coefficient of short-channel effect on VTH
DVT0W	first coefficient of narrow-width effect on VTH for small channel length
DVT1	second coefficient of short-channel effect on VTH
DVT2	body-bias coefficient of short-channel effect on VTH
DVT2W	body-bias coefficient of narrow-width effect on VTH for small channel length
NDEP	channel doping concentration at X_{dep0} the depletion edge at $V_{bs} = 0$
V_{bseff}	effective body bias
V_{ds}	Drain-Source voltage
L_{eff}	effective channel length
W_{eff}	effective width
l_t	characteristic length

Table B.2: Threshold voltage modelling, sub-threshold swing modelling, and mobility modelling in BSIM model parameters

VTH0	VTH0	long channel threshold voltage at Vbs = 0
K1	K1	first-order body effect coefficient
K2	K2	second-order body effect coefficient
K3	K3	narrow width coefficient
A0	A0	Coefficient of channel-length dependence of bulk charge effect
A1	A1	First non-saturation effect parameter
A2	A2	Second non-saturation factor
VSAT	VSAT	Saturation velocity
DVT0	DVT0	first coefficient of short-channel effect on VTH
DVT0W	DVT0W	first coefficient of narrow-width effect on VTH for small channel length
DVT1	DVT1	second coefficient of short-channel effect on VTH
DVT1W	DVT1W	second coefficient of narrow-width effect on VTH for small channel length
DVT2	DVT2	body-bias coefficient of short-channel effect on VTH
DVT2W	DVT2W	body-bias coefficient of narrow-width effect on VTH for small channel length
NFACTOR	NFACTOR	subthreshold swing factor
VOFF	VOFF	Offset voltage in subthreshold region for large W and L
DSUB	DSUB	DIBL coefficient exponent in subthreshold region
ETA0	ETA0	DIBL coefficient in the subthreshold region
ETAB	ETAB	body-bias for the subthreshold DIBL effect
CIT	CIT	Interface trap capacitance
CDSC	CDSC	Drain-Source to channel coupling capacitance
CDSCB	CDSCB	Body-bias coefficient of CDSC
CDSCD	CDSCD	Drain-bias coefficient of CDSC
U0	U0	Low-field mobility
UA	UA	First-order mobility degradation coefficient due to vertical field
UB	UB	Second-order mobility degradation coefficient
MINV	MINV	V_{gsteff} fitting parameter for moderate inversion condition

Table B.3: Saturation Region Output Conductance Parameters

Saturation Region Variable	BSIM4 Parameter	Description
DROUT	DROUT	Channel-length dependence coefficient of the DIBL effect on output resistance
PSCBE1	PSCBE1	First substrate current induced body-effect parameter
PSCBE2	PSCBE2	Second substrate current induced body-effect coefficient
FPROUT	FPROUT	Effect of pocket implant on Rout degradation
PDITS	PDITS	Impact of drain-induced Vth shift on Rout
PDITSL	PDITSL	Channel-length dependence of drain-induced Vth shift on Rout
PDITSD	PDITSD	Vds dependence of drain-induced Vth shift on Rout
PCLM	PCLM	hannel length modulation parameter
PDIBLC1	PDIBLC1	First output resistance DIBL effect parameter
PDIBLC2	PDIBLC2	Second output resistance DIBL effect parameter
PDIBLCB	PDIBLCB	Body bias coefficient of output resistance DIBL effect
DELTA	DELTA	Parameter for DC V_{dseff}
RDSW	RDSW	Zero bias LDD resistance per unit width for RDSMOD=0

Appendix C

NMOS reference device netlist

```
* NMOS Motif Extraction Test
.INCLUDE ./NMOS.lib

VDS v_id 0
VIDS v_id v_d
VGS v_g 0

* Measurement transistors

Mmeas_1 v_d v_g 0 0 N_50n L=5e-08 W=8e-08

.OPTIONS NOPAGE NOMOD NOACCT METHOD=GEARS MAXORD=2 INGOLD=2
.DC VDS 0 1.2 0.01 VGS 0 1.2 0.1
.PRINT DC v(v_d) v(v_g) i(VIDS)
.END
```

Bibliography

[1] Computer history Museum. "1971 - Microprocessor Integrates CPU Function onto a Single Chip" *www.computerhistory.org. [Online]. Available: http://www.computerhistory.org/semiconductor/timeline/1971-MPU.html.* [Accessed: Mar. 29, 2015].

[2] Wasson, S. (2013, Aug. 28). "Xbox One SoC rivals AMD's Tahiti for Size, Complexity" *The Techreport News [Online]. Available: http://techreport.com/news/25288/xbox-one-soc-rivals-amd-tahiti-for-size-complexity.* [Accessed: Mar. 29, 2015].

[3] Rosso, D. (2015, Feb. 2). "Global Semiconductor Industry Posts Record Sales in 2014" *The Semiconductor Industry Association (SIA) [Online]. Available: http://www.semiconductors.org/news/2015/02/02/global_sales_report_2014/global_semiconductor_industry_posts_record_sales_in_2014/.* [Accessed: Mar. 29, 2015].

[4] "AMD Plans for 25x Efficiency Gains by 2020" *www.futuretimeline.net. [Online]. Available: http://www.futuretimeline.net/blog/computers-internetblog.htm#.VSVYK_lC9n0.* [Accessed: Mar. 29, 2015].

[5] Taur, Y. and Ning, T. H. "Fundamentals of Mordern VLSI Devices." *Cambridge University Press,* 2009.

[6] "2013 International Technology Roadmap for Semiconductors Process Integration, Device, and Structure Summary" *The International Technology Roadmap for Semiconductors [Online]. Available: http://public.itrs.net/ITRS%201999-2014%20Mtgs,%20Presentations%20&%20Links/2013ITRS/2013Chapters/2013PIDS_Summary.pdf.* [Accessed: Mar. 29, 2015].

[7] Wong, B.P.; Mittal, A.; Cao, Y. and Starr, G. "Nano-CMOS Circuit and Physical Design." *John Wiley & Sons, Inc., Hoboken, New Jersey.* 2005.

[8] Wang, X.S. "Simulation Study of Scaling Design, Performance Characterization, and Statistical Variability and Reliability of decananometer MOSFETs." *PhD Thesis, University of Glasgow,* 2010.

[9] Schwierz, F. "Graphene Transistors." *Nature Nanotechnology,* Vol. 5, pp. 487-496, 2010.

[10] Teubner, J. and Woods, L. "Data Processing on FPGAs." *Morgan & Claypool Publishers,* 2013.

[11] Weste, N. and Eshraghian, K. "Principles of CMOS VLSI Design A System Perspective" *Addison-Wesley,* 1985.

[12] Allen, P.E. and Holberg, D.R. "CMOS Analog Circuit Design (3rd edition)." *Oxford University Press,* 2011.

[13] Borkar, S. "Design Reliable Systems from Unreliable Components: The Challenges of Transistor Variability and Degradation." *IEEE Micro,* Vol. 25, pp. 10-16, 2005.

[14] Yamamoto, T.; Ooishi, N. and McKay, K. "Using In-design Physical Verification to Reduce Tapeout Schedules." *Technical report, Toshiba Corp. & Synopsys Inc.,* 2010.

[15] Hilder, J.A. "Evolving Variability Tolerant Logic." *PhD Thesis, University of York,* 2010.

[16] Calhoun, B.H.; Cao, Y.; Li, X.; Mai, K.; Pileggi, L.T.; Rutenbar, R.A. and Shepard, K.L. "Digital Circuit Design Challenges and Opportunities in the Era of Nanoscale CMOS." *Proceedings of the IEEE,* Vol. 96, No. 2, pp. 343-365, Feb. 2008.

[17] Cheung, N. "Mircofabrication Technology Lecture Notes." *University of California at Berkeley, [Online]. Available: http://www-inst.eecs.berkeley.edu/∼ee143/fa10/lectures/Lec_05.pdf.* [Accessed: Mar. 29, 2015].

[18] Cobb, N. B. "Fast Optical and Process Proximity Correction Algorithms for Integrated Circuit Manufacturing." *PhD Thesis, University of California at Berkeley,* 1994.

[19] Allgair, J.A.; Ivy, M.; Lucas, K.; Sturtevant, J.L.; Elliott, R.C.; Mack, C.A.; MacNaughton, C.W.; Miller, J.D.; Pochkowski, M.; Preil, M.E.; Robinson, J.C. and Santos, F. "Characterization of Optical Proximity Correction Features." *Journal of Metrology, Inspection, and Process Control for Microlithography XV,* Vol.4344, 2001.

[20] Levenson, M.D.; Viswanathan, N.S. and Simpson, R.A. "Improving Resolution in Photolithography with a Phase-Shifting Mask." *IEEE Transactions on Electron Devices,* Vol. ED-29, pp. 1828-1836, 1982.

[21] Rai-Choudhury, P. "Handbook of Microlithography, Micromachining, and Microfabrication: Vol 1." *SPIE Press,* 1999.

[22] Mack, C.A. "Off-Axis Illumination." *Technical report, KLA-Tencor, FINLE Division,* 2003.

[23] Wong, B.P.; Mittal, A.; Starr, G.W.; Zach, F.; Moroz, V. and Kahng, A. "Nano-CMOS Design for Manufacturability: Robust Circuit and Physical Design for Sub-65nm Technology Nodes." *Wiley-Blackwell,* 2008.

[24] Wong, H.P.; Frank, D.J.; Solomon, P.M.; Wann, C.H.J. and Welser, J.J. "Nanoscale CMOS." *Proceedings of the IEEE,* Vol. 87, pp. 537-570, 1999.

[25] Kim, Y.B. "Challenges for Nanoscale MOSFETs and Emerging Nanoelectronics." *Transcations on Electrical and Electronic Materials,* Vol. 11, pp. 93-105, 2010.

[26] Saxena, S.; Hess, C.; Karbasi, H.; Rossoni, A.; Tonello, S.; McNamara, P.; Lucherini, S.; Minehane, S.; Dolainsky, C. and Quarantelli, M. "Variation in Transistor Performance and Leakage in Nanometer-Scale Technologies." *IEEE Transactions on Electron Devices,* Vol. 55, pp. 131-144, 2008.

[27] Patel, K. "Intrinsic and Systematic Variability in Nanometer CMOS Technologies." *PhD Thesis, University of California at Berkeley,* 2010.

[28] Pang, L.T. "Measurement and Analysis of Variability in CMOS Circuits." *PhD Thesis, University of California at Berkeley,* 2008.

[29] Asenov, A. and Cheng, B. "Modeling and Simulation of Statistical Variability in Nanometer CMOS Technologies." *Springer,* 2011.

[30] "Statistical Methods For Semiconductor Chip Design." *Manual, Silicon Integration Initiative, Inc,* 2008.

[31] Cathignol, A.; Cheng, B.; Chanemougame, D.; Brown, A.R.; Rochereau, K. and Asenov, A. "Quantitative evaluation of statistical variability sources in a 45nm technological node LP N-MOSFET." *IEEE Electron Device Letters,* Vol. 29, pp. 609-611. 2008.

[32] Asenov, A.; Roy, S.; Brown, R.A.; Roy, G.; Alexander, C.; Riddet, C.; Millar, C.; Cheng, B.; Martinez, A.; Seoane, N.; Reid, D.; Bukhori, M.F.; Wang, X. and Kovac, U. "Advanced Simulation of Statistical Variability and Reliability in Nano CMOS Transistors." *IEEE International Electron Devices Meeting (IEDM),* 2008.

[33] Okada, K.; Yamaoka, K. and Onodera, H. "A Statistical Gate Delay Model for Intra-chip and Inter-chip Variabilities." *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC),* pp. 31-36, Jan. 2003.

[34] Zheng, X. and Edwards, D. "Statistical Analysis Model of Nano-CMOS Variability with Intra-die Correlation Due to Proximity." *8th EUROSIM Congress on Modelling and Simulation (EUROSIM),* Cardiff, Sept. 2013.

[35] Merrett, M. "Modelling Statistical Variability Within Circuits Using Nano-CMOS Technologies." *PhD Thesis, University of Southampton,* 2012.

[36] Wirnshofer, M. "Variation-Aware Adaptive Voltage Scaling for Digital CMOS Circuit." *Springer Netherlands,* 2013.

[37] Shauly, E.N. "CMOS Leakage and Power Reduction in Transistors and Circuits: Process and Layout Considerations." *Low Power Electronics and Applications,* Vol. 2, pp. 1-29, 2012.

[38] Asenov, A; Brown, A.R.; Davies, J.H.; Kaya, S. and Slavcheva, G. "Simulation of Intrinsic Parameter Fluctuations in Decananometer and Nanometer-Scale MOSFETs." *IEEE Transactions on Electron Devices,* Vol. 50, pp. 1837-1852, 2003.

[39] "As Nodes Advance, So Must Power Analysis." *Technical report, MENTOR GRAPHICS,* 2014.

[40] Asenov, A; "Advanced Monte Carlo Techniques in the Simulation of CMOS Devices and Circuits." *Numerical Methods and Applications, Lecture Notes in Computer Science,* Vol. 6046, pp 41-49, 2011.

[41] Razavi, B. "Design of Analog CMOS Integrated Circuits." *The Mc Graw-Hill Companies,* 2001.

[42] Kaeslin, H. "Digital Integrated Circuit Design From VLSI Architectures to CMOS Fabrication." *Cambridge University Press,* 2008.

[43] "Sentaurus Workbench User Guide." *Synopsys User Manual,* Sept.2014.

[44] "Sentaurus Device User Guide." *Synopsys User Manual,* Sept.2014.

[45] Xiao, Y.; Trefzer, M.A.; Roy, S.; Walker, J.A.; Bale, S.J. and Tyrrell, A.M. "Circuit Optimization using Device Layout Motifs." *5th European Workshop on CMOS Variability (VARI),* pp. 1-6, Spain, Sept.2014.

[46] "Sentaurus Technology Template: Simulation of 3D NMOS Transistor." *Technical report, Synopsys,* 2012.

[47] "Sentaurus Structure Editor User Guide." *Synopsys User Manual,* Sept.2014.

[48] Vasileska, D. "Drift-Diffusion Model: Introduction." *Arizona State University, [Online]. Available: https://nanohub.org/resources/1545/download/ ddmodel_introductory_part_word.pdf.* [Accessed: April. 01, 2015].

[49] Klaassen, D.B.M. "A Unified Mobility Model for Device Simulation." *International Electron Devices Meeting (IEDM),* pp. 357-360, Dec.1990.

[50] Zheng, P.; Rougieux, F.E.; Macdonald, D. and Cuevas, A. "Measurement and Parameterization of Carrier Mobility Sum in Silicon as a Function of Doping, Temperature and Injection Level." *IEEE Journal of Photovoltaics,* Vol. 4, pp. 560-565, 2014.

[51] Verghese, N.; Nachman, R. and Hurat, P. "Modeling Stress-Induced Variability Optimizes IC Timing Performance." *Technical report, Cadence,* 2011.

[52] Bianchi, R.A.; Bouche, G. and Roux-dit-Buisson, O. "Accurate Modeling of Trench Isolation Induced Mechanical Stress effects on MOSFET Electrical Performance." *InternationalElectron Devices Meeting (IEDM),* pp. 117-120, 2002.

[53] Gallon, C.; Reimbold, G.; Ghibaudo, Gerard.; Bianchi, R.A.; Gwoziecki, R.; Orain, S.; Robilliart, E.; Raynaud, C.; Dansas, H. "Electrical Analysis of Mechanical Stress Induced by STI in Short MOSFETs using Externally Applied Stress." *IEEE Transactions on Electron Devices,* Vol. 51, pp. 1254-1261, 2004.

[54] Scott, G.; Lutze, J.; Rubin, M.; Nouri, F.; Manley, M. "NMOS Drive Current Reduction Caused by Transistor Layout and Trench Isolation Induced Stress." *International Electron Devices Meeting (IEDM),* pp. 827-830, Dec.1999.

[55] Kanda, Y. "Piezoresistance Effect of Silicon." *Sensors and Actuators A: Physical,* Vol. 28, pp. 83-91. 1991.

[56] Hynecek, J. "Elastoresistance of n-type Silicon on Sapphire." *Journal of Applied Physics,* Vol. 45, pp. 2631, 1974.

[57] Tsaur, B-Y.; Fan, J.C.C.; Chapman, R.L.; Geis, M.W.; Silversmith, D.J.; Mountain, R.W. "SOI/CMOS Circuits Fabricated in Zone-melting-recrystallized Si Films on SiO2-coated Si Substrates." *IEEE Electron Device Letters,* Vol. 3, pp. 398-401, 1982.

[58] Damrongplasit, N. "Study of Variability in Advanced Transistor Technologies." *PhD Thesis, University of California at Berkeley,* 2014.

[59] Garg, R. and Khatri, S. P. "A Variation Tolerant Combinational Circuit Design Approach Using Parallel Gates.(in Chapter 9)" *Springer US,* pp. 153-171, 2011.

[60] Ho, B.; Sun, X.; Xu N.; Sako, T.; Maekawa, K.; Tomoyasu, M.; Akasaka, Y.; Liu, T.J.K. "Fabrication of Segmented-channel MOSFETs for Reduced Short-channel Effects." *International Semiconductor Device Research Symposium (ISDRS),* pp. 1-2, 2011.

[61] Nesamani I, F.P.; Sujith M.B. and Lakshmi P.V. "Segmented-Channel MOSFET Design and Optimization-A Review." *Journal of Engineering Trends and Technology,* Vol. 8, pp. 468-471, 2014.

[62] Ho, B.; Sun, X.; Xu, N.; Sako, T.; Maekawa, K.; Tomoyasu, M.; Akasaka, Y.; Bonnin, O.; Nguyen, B.-Y.; Liu, T.J. K. "First Demonstration of Quasi-Planar Segmented-Channel MOSFET Design for Improved Scalability." *IEEE Transactions on Electron Devices,* Vol. 59, pp. 2273-2276, 2012.

[63] Nam, H. and Shin, C. "The Design Optimization and Variation Study of Segmented-channel MOSFET Using HfO2 or SiO2 Trench Isolation." *International Symposium on VLSI Technology, Systems, and Applications (VLSI-TSA),* pp. 1-2, 2013.

[64] Sun, X. "Nanoscale Bulk MOSFET Design and Process Technology for Reduced Variability." *PhD Thesis, University of California at Berkeley,* 2010.

[65] Bohr, M. "Technology Insights: 22nm Tri-gate Transistor for Industry-leading Low Power Capabilites." *[Online]. Available: http://www.intel.com/idf/library/pdf/sf_2011/SF11_SPCS002_101F.pdf.* 2011.

[66] "CMOS SRAM Analysis." *Technical report, Chipworks Inc.,* 2010.

[67] Kawa, J. "FinFET Design, Manufacturability, and Reliability." *Technical report, Synopsys,* 2015.

[68] Ban, Y.C.; Sundareswaran, S.; Panda, R.; Pan D.Z. "Electrical Impact of Line-Edge Roughness on Sub-45nm Node Standard Cell." *Design for Manufacturability through Design-Process Integration III,* Mar.2009.

[69] Kim, S.D.; Wada, H.; Woo, J.C.S. "TCAD-based Statistical Analysis and Modeling of Gate Line-Edge Roughness Effect on Nanoscale MOS Transistor Performance and Scaling." *IEEE Transactions on Semiconductor Manufacturing,* Vol. 17, pp. 192-200, 2004.

[70] Ye, Y ; Liu, F. ; Chen M.; Nassif, S. ; Cao, Y. "Statistical Modeling and Simulation of Threshold Variation under Dopant Fluctuations and Line-Edge Roughness." *IEEE Transactions on Very Large Scale Integration (VLSI) Systems,* Vol. 19, pp. 987-996, 2010.

[71] Singha, R.; Balijepalli, A.; Subramaniam, A.; Liu, F.; Nassif, S. "Modeling and Analysis of Non-rectangular Gate for Post-lithography Circuit Simulation." *Design Automation Conference (DAC),* 2007.

[72] Ye, Y. "Modeling and Simulation of Variations in Nano-CMOS Design." *PhD Thesis, Arizona State University,* 2011.

[73] Ban, Y.C. and Yang, J.S. "Layout Aware Line-Edge Roughness Modeling and Poly Optimization for Leakage Minimization." *Design Automation Conference (DAC),* pp.447-452, 2011.

[74] Asenov, A.; Kaya, S. and Brown, A.R. "Intrinsic Parameter Fluctuations in Decananometer MOSFETs Introduced by Gate Line Edge Roughness." *IEEE Transactions on Electron Devices,* Vol. 50, pp. 1254-1260, 2003.

[75] Reid, D.T. "Large-Scale Simulations of Intrinsic Parameter Fluctuations in Nano-Scale MOSFETs." *PhD Thesis, University of Glasgow,* 2010.

[76] McConaghy, T.; Breen, K.; Dyck, J. and Gupta, "A. Variation-Aware Design of Custom Integrated Circuits: A Hands-on Field Guide." *Springer New York,* 2013.

[77] Graeb, H. ; Zizala, S. ; Eckmueller, J. ; Antreich, K. "The Sizing Rules Method for Analog Integrated Circuit Design." *IEEE/ACM International Conference on Computer Aided Design,* pp.343-349, 2001.

[78] Wang, X.S.; Adamu-Lema, F.; Cheng, B.; Asenov, A. "Geometry, Temperature, and Body Bias Dependence of Statistical Variability in 20-nm Bulk CMOS Technology: A Comprehensive Simulation Analysis." *IEEE Transactions on Electron Devices,* Vol. 60, pp. 1547-1554, 2013.

[79] Bernstein, K.; Frank, D.J.; Gattiker, A.E.; Haensch, W.; Ji, B.L.; Nassif, S.R.; Nowak, E.J.; Pearson, D.J.; Rohrer, N.J. "High-performance CMOS Variability in the 65-nm Regime and Beyond." *IBM Journal of Research and Development,* Vol. 50, pp. 433-449, 2010.

[80] Kuhn, K.J.; Giles, M.D.; Becher, D.; Kolar, P.; Kornfeld, A.; Kotlyar, R.; Ma, S.T.; Maheshwari, A.; Mudanai, S. "Process Technology Variation." *IEEE Transactions on Electron Devices,* Vol. 58, pp. 2197-2208, 2011.

[81] "Sentaurus TCAD Industry-Standard Process and Device Simulators." *Synopsys DataSheet,* 2012.

[82] Paluchowski, S.H.; Cheng, B.; Roy, S.; Asenov, A. and Cumming, D.R.S. "Investigation into Effects of Device Variability on CMOS Layout Motifs." *IEEE Electronics Letters,* Vol. 44, No. 10, pp. 626-627, 2008.

[83] Caldwell, S.; Cumming, D.R.S."Design for Variability in CMOS Logic Circuits: Uncommitted Motif Arrays (UMAs)." *IEEE International Symposium on Circuits and Systems,* pp. 1807-1810, Taipei, May 2009.

[84] Dessouky, M.; Louerat, M. "A Layout Approach for Electrical and Physical Design Integration of High-Performance Analog Circuits." *IEEE 2000 First International Symposium on Quality Electronic Design,* pp. 291-298, 2000.

[85] "Sentaurus Process User Guide." *Synopsys User Manual,* Sept.2014.

[86] "Mesh Generation Tools User Guide." *Synopsys User Manual,* Sept.2014.

[87] Drechsler, R. "Evolutionary Algorithms for VLSI CAD." *Springer US,* 1998.

[88] Bazylevych, R. and Bazylevych, L., "The Methodology and Algorithms for Solving the Very Large-scale Physical Design Automation Problems: Partitioning, Packaging, Placement and Routing" *2nd Mediterranean Conference on Embedded Computing (MECO),* Budva, Jun. 2013.

[89] Xiao, Y.; Walker, J.A.; Bale, S.J.; Trefzer, M.A. and Tyrrell, A.M. "Circuit Design Optimisation Using a Modified Genetic Algorithm and Device Layout Motifs." *IEEE International Conference on Evolvable Systems (ICES),* pp.1-8, USA, Dec. 2014.

[90] Lienig, J. "Physical Design of VLSI Circuits and Application of Genetic Algorithm (Part III)." *Springer Berlin Heidelberg*, pp.277-292, 1997.

[91] Binh, T. and Korn, U. "An Evolution Strategy for the Multi-Objective Optimization." *Proceedings of the Second International Conference on Genetic Algorithms,* pp.23-28, 1996.

[92] Liu, M.G. and He, J.S. "A Multi-algorithm Framework for the Evolvement of Analog Circuits." *International Conference on Systems and Informatics,* pp. 780-784, May 2012.

[93] Walker, J.A.; Hilder, J.A.; Reid, D.; Asenov, A.; Roy, S.; Millar, C. and Tyrrell, A.M. "The Evolution of Standard Cell Libraries for Future Technology Nodes." *Genetic Programming and Evolvable Machine,* Vol. 12, pp.235-256. 2011.

[94] Wankhede, M. V. ; Deshmukh, A. Y. "Optimization of Cell-based VLSI Circuit Design using a Genetic Algorithm: Design Approach." *Proceedings of the International Multi-Conference of Engineers and Computer Scientists,* pp. 1603-1608, 2009.

[95] Xia, X.W.; Li, Y.X.; Ying, W.Q.; Chen, L. "Automated Design Approach for Analog Circuit Using Genetic Algorithm." *Springer Berlin Heidelberg,* pp. 1124-1130, 2007.

[96] Koza, J.R. Bennett, F.H. ; Andre, D. ; Keane M. A. and Dunlap, F. "Automated Synthesis of Analog Electrical Circuits by Means of Genetic Programming." *IEEE Transactions on Evolutionary Computation,* Vol. 1, pp.109-128, 1997.

[97] Holland, J.H. "Adaptation in Natural and Artificial Systems." *MIT Press,* 1975.

[98] Melanie, M. "An Introduction to Genetic Algorithms." *MIT Press,* 1998.

[99] Eiben, A.E. and Smith, J.E. "Introduction to Evolutionary Computing." *Springer Berlin Heidelberg,* 2003.

[100] Beyer, H.G. and Schwefel, H.P. "Evolution Strategies-A Comprehensive Introduction," *Journal of Natural Computing,* Vol. 1, pp. 3-52, 2002.

[101] Abraham, A. and Goldberg, R. "Evolutionary Multiobjective Optimization: Theoretical Advances and Applications." *Springer-Verlag London,* 2005.

[102] Coello Coello, C. ; Lamont, G.B. ; van Veldhuizen, D.A. "Evolutionary Algorithms for Solving Multi-objective Problems." *Springer US,* 2007.

[103] Deb, K. "Multi-Objective Optimization using Evolutionary Algorithms." *Wiley,* 2009.

[104] Deb, K.; Tiwari, R.; Dixit, M.; Dutta, J. "Finding Trade-off Solutions Close to KKT Points using Evolutionary Multi-objective Optimization." *Proceedings of the Congress on Evolutionary Computation (CEC),* pp.2109-2116, 2007.

[105] Walker, J.A. "The Automatic Acquisition, Evolution and Re-use of Modules in Cartesian Genetic Programming." *PhD Thesis, University of York,* 2007.

[106] Holland, J.H. "Genetic Algorithms and Adaptation." *Springer US*, 1984.

[107] Jin, Y. "A Comprehensive Survey of Fitness Approximation In Evolutionary Computation." *Journal of Soft Computing,* Vol. 9, pp. 3-12, 2005.

[108] Karyotis, V.; Stai, E.; Papavassiliou, S. "Evolutionary Dynamics of Complex Communications Networks." *CRC Press,* 2013.

[109] Kumar, R. and Banerjee, N. "Multicriteria Network Design Using Evolutionary Algorithm." *Genetic and Evolutionary Computation Conference,* pp. 2179-2190, USA, Jun. 2003.

[110] Rothlauf, F. "Representations for Genetic and Evolutionary Algorithms." *Springer-Verlag Berlin Heidelberg,* 2006.

[111] Golub, M. "An Implementation of Binary and Floating Point Chromosome Representation in Genetic Algorithm." *Proceedings of the 18th International Conference on Information Technology Interfaces,* pp. 417-422, Croatia, Jun, 1996.

[112] Jamikow, Z.C. and Michalewicz, Z. "An Experimental Comparison of Binary and Floating Point Representations in Genetic Algorithms." *Proceedings of the 4th International Conference on Genetic Algorithms,* USA, Jul. 1991.

[113] Shi, Z. ; Cui, J. L ; Tao, D. and Zhou, Y. "Comparison of Steady State and Elitist Selection Genetic Algorithms." *International Conference on Intelligent Mechatronics and Automation,* pp. 495-499, China, Aug. 2004.

[114] Back, T. "Selective Pressure in Evolutionary Algorithms: a Characterization of Selection Mechanisms." *IEEE World Congress on Computational Intelligence (WCCI),*, pp. 57-62, USA, Jun. 1994.

[115] Back, T.; Hammel, U.; Schwefel, H.P. "Evolutionary Computation: Comments on the History and Current State." *IEEE Transactions on Evolutionary Computation,* Vol. 1, pp. 3-17. 1997.

[116] Blickle, T. and Thiele, L. "A Comparison of Selection Schemes Used in Genetic Algorithms." *Technical report, Computer Engineering and Communication Networks Lab, Swiss Federal Institute of Technology,* 1995.

[117] Kumar, R. and Jyotishree, "Blending Roulette Wheel Selection & Rank Selection in Genetic Algorithms." *International Journal of Machine Learning and Computing*, Vol. 2, pp. 365-370. 2012.

[118] Lipowskia, A.; Lipowskab, D. "Roulette-wheel Selection via Stochastic Acceptance." *Physica A: Statistical Mechanics and its Applications,* Vol. 391, pp. 2193-2196, 2012.

[119] Razali, N.M. and Geraghty, J. "Genetic Algorithm Performance with Different Selection Strategies in Solving TSP." *Proceedings of the World Congress on Engineering,* UK, July 2011.

[120] Miller, J.F. "Cartesian Genetic Programming." *Springer-Verlag Berlin Heidelberg,* 2011.

[121] Hansen, N.; Arnold, D.V. and Auger, A. "Evolution Strategies." *Springer,* 2015.

[122] Magalhaes-Mendes, J. "A Comparative Study of Crossover Operators for Genetic Algorithms to Solve the Job Shop Scheduling Problem." *WSEAS Transactions on Computers,* Vol. 12, pp. 164-173, 2013.

[123] Picek, S.; Jakobovic, D. and Golub, M. "On the Recombination Operator in the Real-Coded Genetic Algorithms." *IEEE Congress on Evolutionary Computation (CEC),* pp.3103-3110, Mexico, 2013.

[124] De Jong, K. A. ; Spears, W. M. "A Formal Analysis of the Role of Multi-point Crossover in Genetic Algorithms." *Journal of Annals of Mathematics and Artificial Intelligence,* Vol. 5, pp.1-26. 1992.

[125] Masrili Libelli, S.and Alba, P. "Adaptive Mutation in Genetic Algorithms." *Journal of Soft Computing,* Vol. 4, pp.76-80, 2000.

[126] Koenig. A.C. "A Study of Mutation Methods for Evolutionary Algorithms." *Technical report, University of Missouri-Rolla,* 2002.

[127] Mahalakshmi, M.; Kalaivani, P.; Nesamalar, E.K. "A Review on Genetic Algorithm and its Applications." *Journal of Computing Algorithm,* Vol. 2, pp. 415-423, 2013.

[128] Chen, J.W. ; Lin, K.H. ; Zhou, C.L. "The Strength Mutation Evolutionary Algorithm and Its Application in Multi-object Optimization." *International Conference on Natural Computation, (ICNC),* pp.681-685, China, Oct. 2008.

[129] Zhang, Q.F.; Sun, J. Y.; Tsang, E. "An Evolutionary Algorithm with Guided Mutation for the Maximum Clique Problem." *IEEE Transactions on Evolutionary Computation,* Vol. 9, pp.192-200, 2005.

[130] Teo, J. "Self-adaptive Mutation for Enhancing Evolutionary Search in Real-coded Genetic Algorithms." *International Conference on Computing & Informatics, (ICOCI),* pp.1-6, Malaysia, Jun. 2006.

[131] Kitamura, S. and Hiroyasu, M. "Genetic Algorithm with Stochastic Automata-controlled, Relevant Gene-specific Mutation Probabilities." *IEEE International Conference on Evolutionary Computation,* Australia, Dec.1995.

[132] Liu, D.P. "Research on Reverse Genetic Algorithms Based on Adaptive Tuning of Mutation Probability." *IEEE Conference on Industrial Electronics and Applications,* pp.1-3, Singapore, May 2006.

[133] Ceollo Coello, C.A. "A Short Tutorial on Evolutionary Multi-objective Optimization." *International Conference on Evolutionary Multi-Criterion Optimization,* pp.21-40, 2001.

[134] Chang, C.S. and Kwan, C.M. "Evaluation of Evolutionary Algorithms for Multi-objective Train Schedule Optimization." *Springer-Verlag*, pp.803-815, 2004.

[135] Suganthan, P.N.; Hansen, N.; Liang, J.J.; Deb, K.; Chen, Y.P.; Auger, A. and Tiwari, S. "Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-parameter Optimization." *Technical report, Nanyang Technological University*, 2005.

[136] Trefzer, M.A. and Walker, J.A. and Tyrrell, A.M."A programmable analogue and digital array for bio-inspired electronic design optimization at nano-scale silicon technology nodes." Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR), pp.1537-1541, USA, Nov.2011.

[137] Safe, M.; Carballido, J.; Ponzoni, I.; Brignole, N. "On Stopping Criteria for Genetic Algorithms." *17th Brazilian Symposium on Artificial Intelligence*, pp.405-413, 2004.

[138] De Jong, K.; Fogel, D.B. and Schwefel, H.P. "Handbook of Evolutionary Computation." *IOP Publishing Ltd and Oxford University Press.* 1997.

[139] Negoita, M.G. ; Reusch, B. "Real World Applications of Computational Intelligence." *Springer-Verlag Berlin Heidelberg,* 2005.

[140] Maruo, M.H. ; Lopes, H.S. ; Delgado, M.R. "Self-Adapting Evolutionary Parameters: Encoding Aspects for Combinatorial Optimization Problems." *5th European Conference, EvoCOP,* pp.154-165, Switzerland, Mar. 2005.

[141] Eiben, A.E. "Multi-parent Recombination," *IOP Publishing Ltd and Oxford University Press,* 1995.

[142] Deb, K. "Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems." *Evolutionary Computation,* Vol. 7, pp.205-230, 1999.

[143] Fonseca, C.M.; Fleming, P.J. "Multiobjective Genetic Algorithms." *IEE Colloquium on Genetic Algorithms for Control Systems Engineering,* pp.6/1-6/5, UK, 1993.

[144] Deb, K.; Pratap, A.; Agarwal, S. and Meyarivan T. "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II." *IEEE Transactions on Evolutionary Computation,* Vol. 6, pp. 182-197, 2002.

[145] Horn, J.; Nafpliotis, N.; Goldberg, D.E. "A Niched Pareto Genetic Algorithm for Multiobjective Optimization." *IEEE World Congress on Computational Intelligence,* pp. 82-87, USA, Jun. 1994.

[146] Wei, Y. ;Zhang, Q.J. ; Nakhla, M.S. "Multilevel Optimization of High Speed VLSI Interconnects Using Decomposition." *Proceedings of the IEEE Custom Integrated Circuits Conference,* USA, May 1993.

[147] Zhang, Q.F. and Li, H. "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition." *IEEE Transactions on Evolutionary Computation,* Vol. 11, pp. 712-731, 2007.

[148] Prada, D.; Bellini, M.; Stevanovic, I.; Lemaitre, L.; Victory, J.; Vobecky, J.; Sacco, R.; Lauritzen, P. "On the Performance of Multiobjective Evolutionary Algorithms in Automatic Parameter Extraction of Power Diodes." *IEEE Transactions on Power Electronics,* Vol. PP, pp. 1-13, 2014.

[149] Chevillon, N.; Tang, M.C.; Pregaldiny, F.; Lallement, C.; Madec, M. "FinFET Compact Modeling and Parameter Extraction." *International Conference Mixed Design of Integrated Circuits & Systems (MIXDES),* pp. 55-60, Poland, Jun. 2009.

[150] Paasschens, J.C.J.; Kloosterman, W.J.; Havens, R.J. "Parameter Extraction for the Bipolar Transistor Model Mextram." *Technical report, Koninklijke Philips Electronics,* 2001.

[151] Gildenblat, G. "Compact Modeling Principles, Techniques and Applications." *Springer Netherlands,* 2010.

[152] Mhlanga, S.; Ndlovu, J.; Mbohwa, C.; Mutingi, M. "Design of Comminution Circuits for Improved Productivity using a Multi-Objective Evolutionary Algorithm (MOEA)." *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM),* pp. 1680-1684, Singapore, Dec. 2011.

[153] Yuan, H.Y.; He, J.S. "Evolutionary Design of Operational Amplifier using Variable-length Differential Evolution Algorithm." *International Conference on Computer Application and System Modeling (ICCASM),* pp.610-614, China, Oct. 2010.

[154] Fogel, G.B. "Evolutionary Programming." *Springer-Verlag Berlin Heidelberg, Natural Selection, Inc.,* USA, 2012.

[155] Deb, K. "Multi-Objective Optimization using Evolutionary Algorithms." *John Wiley and Sons LTD,* England, 2001.

[156] Minella, G.; Ruiz, R.; Ciavotta, M. "A Review and Evaluation of Multi-objective Algorithms for the Flowshop Scheduling Problem." *INFORMS Journal on Computing,* Vol. 20, No. 3, pp. 451471, 2008.

[157] Mazumder, P.; Rudnick, E.M. "Genetic Algorithms for VLSI Design, Layout and Test Automation." *Prentice Hall PTR,* Upper Saddle River, NJ, USA 1999.

[158] Koza, J.R. "Genetic Programming: On the Programming of Computers by Means of Natural Selection." *MIT Press,* 1992.

[159] Roper, M.; Maclean, I.; Brooks, A.; Miller, J.; Wood, M. "Genetic Algorithms and the Automatic Generation of Test Data." *Technical report, Semin. Arthr. Rheum,* 1995.

[160] Koza, J.R.; Bennett III, F.H.; Lohn, J.; Dunlap, F.; Keane, M.A. and Andre, D. "Automated Synthesis of Computational Circuits using Genetic Programming." *In Proceedings*

of the 1997 IEEE International Conference on Evolutionary Computation, pp. 447452, 1997.

[161] Langeheine, J.; Meier, K. and Schemmel, J. "Intrinsic Evoltion of Quasi DC Solutions for Transistor Level Analog Electronic Circuits using a CMOS FPTA Chip." 2002.

[162] Koziel, S. and Szczesniak, W. "Reducing Average and Peak Temperatures of VLSI CMOS Circuits by Means of Evolutionary Algorithm Applied to High Level Synthesis." *Microelectronics Journal,* pp. 11671174, 2003.

[163] Wu, C.; "Evolutionary Design of Analogue Circuits." *Master thesis,* University of York, UK, 2005.

[164] Trefzer, M.A.; "Evolution of Transistor Circuits." *PhD thesis,* Ruperto-Carola-University of Heidelberg, Germany, 2006.

[165] Li, Y.M.; "An Automatic Parameter Extraction Technique for Advanced CMOS Device Modeling using Genetic Algorithm." *Microelectronic Engineering,* pp. 260272, Feb. 2007.

[166] Xiao, Y.L.; Sheng, W.G.; Mao, Z.G. "Soft Error Optimization of Standard Cell Circuits based on Gate Sizing and Multi-Objective Genetic Algorithm." *Design Automation Conference (DAC),* pp. 502 - 507, July 2009.

[167] Walker, J.A.; Trefzer, M.A.; Tyrrell, A.M.; "Designing Function Configuration Decoders for the PAnDA Architecture using Multi-Objective Cartesian Genetic Programming." *IEEE International Conference on Evolvable Systems (ICES),* pp. 96-103. 2013.

[168] Keser, M. and Joardar, K. "Genetic Algorithm Based MOSFET Model Parameter Extraction." *Technical Proceedings of the International Conference on Modeling and Simulation of Microsystems,* pp. 341-344, 2000.

[169] Li, Y.M. "Hybrid Intelligent Approach for Modeling and Optimization of Semiconductor Devices and Nanostructures." *Computational Materials Science,* Vol. 45, pp.41-51, 2009.

[170] Bart, J. and Zeghbroeck, V. "The MOSFET Introduction." *University of Colorado, [Online]. Available: http://ecee.colorado.edu/bart/book/mosintro.htm.* [Accessed: April. 10, 2015].

[171] Morshed, T.H.; Yang, W.W.; Dunga, M.V.; Xi, X.M.; He, J.; Liu, W.D.; Cao, M.; Jiang, X.D.; Ou, J.J.; Chan, M.S.; Niknejad, A.M. and Hu, C.M. "BSIM4.6.4 MOSFET Model User's Manual." *User Manual, University of California at Berkeley,* 2009.

[172] Nikolic, B. "Microelectronic Devices and Circuits Lecture Notes." *University of California at Berkeley, [Online]. Available: http://www-inst.eecs.berkeley.edu/∼ee105/fa05/handouts/discussions/Discussion5.pdf.* [Accessed: April. 10, 2015].

[173] "Nonlinear Device Models." *Technical report, Agilent,* 2008.

[174] Chen, C.K. "A Genetic Algorithm for Deep-submicron MOSFET Parameters Extraction and Simulation." *Master Thesis, Chiao Tung University,* 2002.

[175] Prada, D. "Multiobjective Optimization for Parameter Extraction of Power Electronics Device." *Master Thesis, Polytechnic of Milan,* 2012.

[176] Bellini, M. ; Stevanovic, I. ; Prada, D. "Improved Lumped Charge Model for High Voltage Power Diode and Automated Extraction Procedure." *IEEE Bipolar/BiCMOS Circuits and Technology Meeting (BCTM),* pp. 49-52, USA, Oct. 2011.

[177] Li, Y.M.; Chen, Y.Y.; Chen, C.Y.; Shen, C.H.; Cheng, H.W.; Lo, I.H.; Chen, C.N. "Device SimulationBased Multiobjective Evolutionary Algorithm for Process Optimization of Semiconductor Solar Cells." *Materials and Manufacturing Processes,* Vol. 28, pp. 761-767, 2013.

[178] Souil, D.; Guegan, G.; Bertrand, G.; Faynot, O.; Deleonibus S.and Ghibaudo, G. "Intelligent BSIM4 Model Parameter Extraction for Sub-100nm MOSFET Era." *International Conference on Microelectronic Test Structures,* 2002.

[179] Watts, J.; Bittner, C.; Heaberlin D. and Hoffman, J. "Extraction of Compact Model Parameters for ULSI MOSFETs Using a Genetic Algorithm." *International Conference on Modeling and Simulation of Microsystems,* pp. 176-179, 1999.

[180] Gowda, S.M. ; Sheu, B.J. ; Chang, R.C. "Effective Parameter Extraction using Multiple-objective Function for VLSI Circuits." *Analog Integrated Circuits and Signal Processing,* Vol. 5, pp.121-133, 1994.

[181] Nishiba, A.; Kawanaka, H.; Takase, H. and Tsuruoka, S. "A Proposal of Genetic Operations for BSIM Parameter Extraction Using Real-Coded Genetic Algorithm." *Advanced Computational Intelligence and Intelligent Informatics,* Vol. 15, pp. 1131-1138, 2011.

[182] Xiao, Y.; Trefzer, M.A.; Walker, J.A.; Bale, S.J.; Tyrrell, A.M. "Two Step Evolution Strategy for Device Motif BSIM Model Parameter Extraction." *IEEE Congress on Evolutionary Computation (CEC),* pp. 2877-2884, China, 2014.

[183] Nenzi, P. and Vogt, H. "Ngspice Users Manual." Sept. 2010.

[184] Asenov, P. "Accurate Statistical Circuit Simulation in the Presence of Statistical Variability." PhD Thesis, University of Glasgow, 2013.

[185] Negin, M. "Statistical Compact Model Strategies for Nano CMOS Transistors Subject of Atomic Scale Variability." *PhD Thesis, University of Glasgow,* 2012.

[186] Ajayan, K.R. and Navakanta, B. "Device Oriented Statistical Modeling Method for Process Variability in 45nm Analog CMOS Technology." *16th International Workshop on Physics of Semiconductor Devices,* 2012.

[187] Cheng, B.J.; Dideban, D.; Moezi, N.; Millar, C.; Roy, G.; Wang, X.S.; Roy, S.; Asenov, A. "Statistical-Variability Compact-Modeling Strategies for BSIM4 and PSP." *IEEE Design & Test of Computers,* Vol. 27, pp. 26-35, 2010.

[188] Reid, D.; Millar, C.; Roy, S. and Asenov, A. "Understanding LER-induced MOSFET VT Variability - Part I: Three-dimensional Simulation of Large Statistical Samples." *IEEE Electron Device Letters,* Vol. 57, pp. 2801-2807, 2010.

[189] Rabaey, J. "Digital Integrated Circuits Lecture Notes." *University of California at Berkeley, [Online]. Available: http://bwrcs.eecs.berkeley.edu/Classes/icdesign/ee141_f01/Notes/chapter6.pdf.* [Accessed: April. 10, 2015].

[190] Hong, T.P.; Wu, M.T. ; Lee, Y.C. "Using Dynamic Mutation Rates in Gene-set Genetic Algorithms." *IEEE International Conference on Systems Man and Cybernetics (SMC),* pp.4018-4022, Turkey, Oct. 2010.

[191] Bentley, P.J. and Wakefield, J.P. "Finding Acceptable Solutions in the Pareto-Optimal Range using Multiobjective Genetic Algorithms." *Soft Computing in Engineering Design and Manufacturing, Springer London,* pp. 231-240, 1998.

[192] Li, M.Q.; Yang, S.X.; Liu, X.H. "A Test Problem for Visual Investigation of High-Dimensional Multi-Objective Search." *IEEE Congress on Evolutionary Computation (CEC),* pp. 2140-2147, China, Jul. 2014.

[193] Li, M.Q.; Zheng, J.H. ; Li, K.; Yuan, Q.Z. ; Shen, R.M. "Enhancing Diversity for Average Ranking Method in Evolutionary Many-Objective Optimization." *11th International Conference on Parallel Problem Solving from Nature,* pp. 647-656, Poland, Sept. 2010.

[194] Corne, D. and Knowles, J. "Techniques for Highly Multiobjective Optimisation: Some Nondominated Points are Better than Others." *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation,* pp. 773-780, USA, 2007.

[195] Jaimes, A.L. and Coello Coello, C.A. "Study of Preference Relations in Many-objective Optimization." *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation,* pp. 611-618, Canada, Jul. 2009.

[196] Yuan, Y.; Xu, H.; Wang, B. "Evolutionary Many-Objective Optimization Using Ensemble Fitness Ranking." *Proceedings of the conference on Genetic and Evolutionary Computation,* pp. 669-676, Canada, Jul. 2014.