Algorithm Driven Transistor sizing based Power-Timing Optimization Methodology for CMOS Digital Circuits

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science
in
Electronics and Communication Engineering

Hema Sai Kalluru
201531221
kalluru.hemasai@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
July 2023

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "**Algorithm Driven Transistor sizing based Power-Timing Optimization Methodology for CMOS Digital Circuits**" by Hema Sai Kalluru, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Advisor: Dr. Zia Abbas

To my beloved parents

Acknowledgments

Foremost, I would like to express my sincere gratitude to my advisor, Dr Zia Abbas, for the continuous support in academics and research, for his patience, motivation and invaluable guidance. He has taught me the methodology to carry out the research and to present the work as clearly as possible. He taught me to be passionate at work and compassionate with people. It was a great privilege and honor to work and study under his guidance.

I would like to thank all the people who have supported me and motivated me throughout my journey at IIITH. A special thanks to my friends from IIIT family, Harshini, Sailaja, Bhavya, Harish, Harsha, Ashfakh, Shirisha, Preetham and others, who have kept my spirits alive throughout this beautiful journey and for making my college life more cherish-able. I would like to acknowledge and thank everyone in CVEST lab for helping me and and for taking part in brain storming the ideas throughout this journey. I extend my token of gratitude to all the Distinguished Faculty in the college without whose cooperation and support, this journey would not have been possible.

I can never thank my parents enough for their love, prayers, care and sacrifices for educating me and preparing me for my future. I am thankful to have the blessings of my grand-parents and constant support of my friends.

Abstract

During the past few decades, the semiconductor VLSI industry has distinguished itself both by the rapid pace of performance improvements, and by a steady path of constantly shrinking device geometries. MOSFETs upon downsizing, have met the world's growing needs for electronic devices like computing, communication, entertainment, automotive, and other applications with steady improvements in cost, speed, and area. But the device density has led to increased power consumption. Hence, Leakage, along with propagation Delay are the key metrics to evaluate the performance of any digital circuit. In nano-scale technologies(<45nm), static power occupies a significant share in the total power budget thereby prioritizing the need for leakage reduction techniques.

Performance optimization of CMOS based circuits gains more significance for nano scale technology nodes. Variations in operating parameters such as supply voltage, temperature etc. have profound effects on power-delays specifications. The non-homogeneity in process parameters at such scaled technologies hampers the yield of the final designs. Performance degradation over time is another important factor determining the lifetime and reliability of an IC. In this work we develop a framework of algorithms to optimize digital circuits for low power and high performance applications. A wide range of analysis such as sensitivity of the circuit, correlation between parameters etc has been performed to understand the functioning in response to transistor sizing of digital cells.

The first stage of this work aims to optimize basic digital cells through transistor sizing using the proposed optimization algorithms like Pareto Harris Hawk optimization algorithm, Glowworm Swarm optimization algorithm, Strength Pareto evolutionary algorithm-II and Neighbourhood Cultivation Genetic algorithm. The resultant system design becomes robust to withstand fluctuations caused by process, aging and operating parameters apart from providing highly improved leakage-delay performances.

The second stage of this work involves optimization complex circuits. We have proposed a framework to optimize the complex circuits by selectively replacing the basic cells with the optimized sizing. The proposed framework deconstructs a given circuit into its constituent basic cells. It does a wide range of analysis such as sensitivity, the correlation between process parameters, load analysis, screening of insignificant parameters, and mismatch analysis to generate robust sizing. The algorithms generate sizing that can be used at multiple instances across different circuits. Thus the total number of design variables remains within a limit even when the cell count increases across circuits.

The proposed framework then reconstructs the circuit with these optimized cells to improve the power-delay front. While replacing the nominal with the optimized sizing for each cell, the tool identifies the load it has to drive, the path (critical/ non-critical) it is present in and its driver modules in the path. The basic cells are selectively replaced with the optimized sizing using the proposed techniques: Backward traversal replacement technique and Partitioning large basic cells. Results have shown a substantial reduction in leakage power and propagation delays in addition to minimizing human effort

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Since the invention of the first Metal Oxide Semiconductor Field Effect Transistors (MOSFET), it has become the basic building block in digital circuit applications such as microprocessors and memories, resulting in low cost, high performance circuits having a high functional density. In 1965, Gordan Moore, the co-founder of Intel made an observation that the number of transistors incorporated in a chip will approximately double every 18 months or so[3]. Since 1975, this prediction, termed as Moore's law has been driving the industry towards long-term planning and set targets in research and development as shown in Figure.1.1.

The last few decades have evidenced tremendous improvements in the power density reduction while keeping and even increasing the device performance as a result of downsizing at every new technology node. However, several unintended consequences have undermined the benefits obtained from the advances in technology. Firstly, a drastic increase in power dissipation with scaling technology nodes, secondly, the growing impact of process variations in fabrication, worsening the power dissipation[4].

1.2 Problem Statement

Continuous miniaturization of complementary metal-oxide-semiconductor (CMOS) technologies has dramatically increased the speed, power efficiency and integration of electronic systems. Moore's law continues to predict the scaling and levels of integration fairly well and the rate of scaling even outperforms the prediction in recent years.

However, this down-scaling is accompanied by increased process variability, in turn, an enormous increase in leakage power. In addition, fluctuations in operating parameters (e.g.

supply voltage and temperature) always significantly deviate the circuit performance from their expected functioning. Moreover, such high-performance circuits are facing a serious threat by aging degradation effects such as Bias Temperature Instability (BTI), which degrade the circuit performance over time. Therefore, circuit optimization for high yield has become a crucial and complex task in IC design. In fact, a high degree of reliability and then the high yield is achievable only if the devices in the circuit are cooperatively invariant against the process and operating variations in conjunction with aging degradation effects.

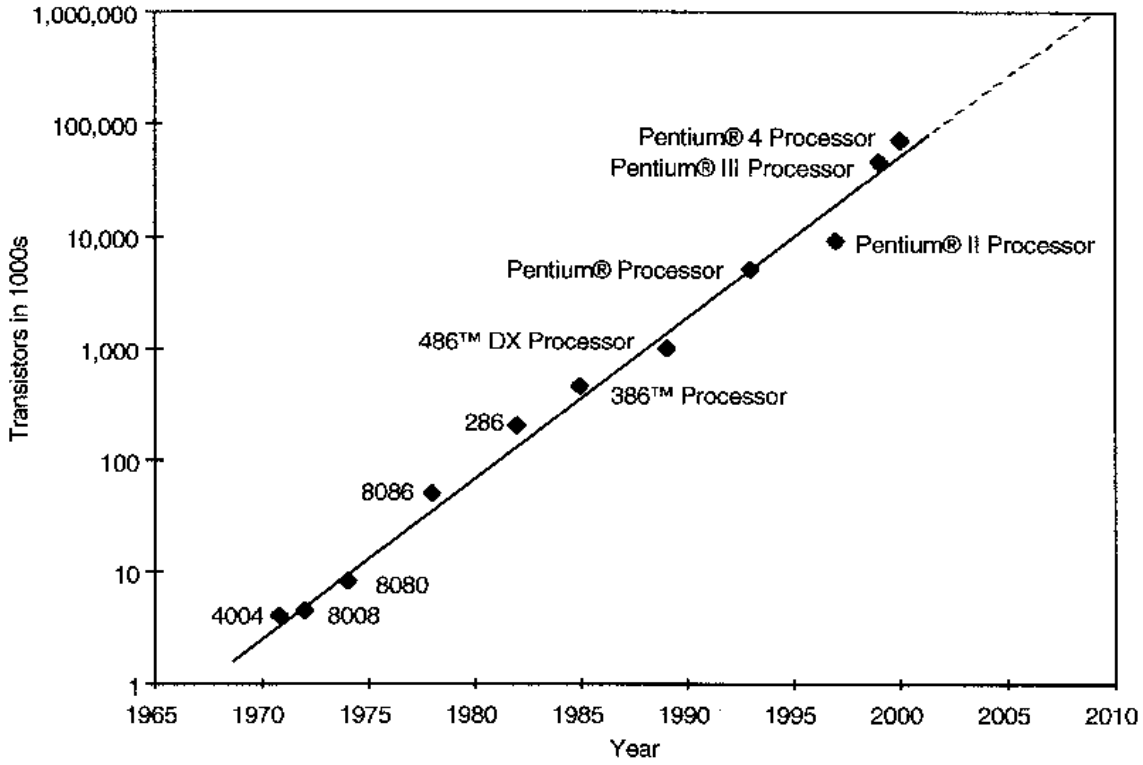


Figure 1.1: Illustration of Moore's law: The number of transistors in different microchips against production years [1]

1.3 Thesis Contribution

This thesis is mainly focused on optimizing the leakage power and propagation delay in CMOS VLSI circuits using Algorithms, to ensure ingenious working at all process and operating conditions. We mainly propose 4 optimization algorithms in this work to obtain the transistor sizing with optimized device performance. The main contributions of this research can be summarized as follows:

- We have performed Circuit Analysis to analyse the response of the circuit on changing various design parameters, local and global process parameters and correlation between process parameters

- We have proposed Novel Pareto Harris Hawk optimization algorithm, Glowwarm swarm optimization algorithm, Neighbourhood cultivation genetic algorithm and Strength pareto evolutionary algorithm - II (SPEA-II) to solve the optimization problem for basic CMOS digital cells.

- We have proposed two methods for optimizing complex circuits, by selectively replacing the basic cells with the optimized sizing obtained using the algorithms reducing the computational complexity to a great extent. The two methods proposed are: 1) Backward Traversal Replacement and 2) Partitioning large basic cells

- We have proposed a top level model for optimizing the complex circuits, a methodology that deconstructs a given circuit into its constituent basic cells. It does a wide range of analysis such as sensitivity, the correlation between process parameters, load analysis, screening of insignificant parameters, and mismatch analysis to generate robust sizing. The proposed framework then reconstructs the circuit with these optimized cells to improve the power-delay front.

- An alternative method for optimizing the Power dissipation and Area by iterative decomposition technique considering the case of the polar decoders.

1.4 Thesis Outline

The remaining of the thesis is organized as follows:

- In chapter 2, We introduce the topic of Technology scaling, impact of technology scaling on CMOS device performances. We also discuss different types of power dissipation in CMOS devices

- In chapter 3, We discuss the impact of process variations, Operating variations like Supply voltage, Temperature Fluctuations and NBTI aging variations. we have considered $\pm 3\sigma$ variation in process parameters, temperature range [-55 °C to 125 °C], $\pm 10\%$ variation in supply voltage i.e. [0.72 to 0.88] and 3 years of NBTI aging.

- In chapter 4, We have performed extensive circuit analysis to understand the behavious of the circuit. In this section, we have performed sensitivity analysis, screening, mismatch analysis and correlation between process parameters.
- In chapter 5, We put forward transistor sizing based optimization technique using algorithms to optimize the leakage power and propagational delay of basic cells. We have proposed pareto Harris Hawk optimization algorithm, Neighbourhood cultivation genetic algorithm, strength pareto optimization algorithm - II, and Glowworm swarm optimization algorithm to optimize the basic cells
- In chapter 6, We propose a methodology that deconstructs a given circuit into its constituent basic cells. It does a wide range of analysis such as sensitivity, the correlation between process parameters, load analysis, screening of insignificant parameters, and mismatch analysis to generate robust sizing. The proposed framework then reconstructs the circuit with these optimized cells to improve the power-delay front. While replacing the nominal with the optimized sizing for each cell, the tool identifies the load it has to drive, the path (critical/ non-critical) it is present in and its driver modules in the path. We have proposed two methods to selectively replace the basic cells with optimized sizing in this framework: Backward traversal replacement and Partitioning large basic cells
- In chapter 7, We propose an alternative approach, Iterative decomposition technique to optimize the power dissipated. We have implemented this technique for a particular case of polar decoders.
- In chapter 8, We conclude the thesis and with some future work

Chapter 2

Impact of Technology Scaling On Performance of Digital Circuits

2.1 Introduction

The need for affordable, high-performance electronic systems has driven significant research and innovation in the semiconductor industry. In particular, silicon-based technologies have witnessed continuous levels of advancement in order to boost performance and maintain strong economies of scale. The growing market of portable electronic devices demands lesser power dissipation for longer battery life and compact system. Advancement in technology effectively minimizes the leakage current & power and size of the cell. The leakage current in a cell is the dominating factor, which greatly affects power consumption. Optimization of power and delay is a very important issue in low-voltage and low-power applications.

2.2 Power Dissipation in CMOS Digital Circuits

The invention of CMOS digital transistors has significantly reduced power consumption as compared to previous technologies such as transistor-transistor and emitter-coupled logic. The static power i.e., the power dissipated when the circuit is not switching, is almost negligible in the early CMOS transistors. However, the power consumption has increased drastically with the increase in the transistor density and device speed.

Moore's law predicted that the number of transistors in an integrated circuit (IC) would approximately double every 2 years. As predicted, the semiconductor industry advanced through the use of dimensions and voltage scaling to create faster and more densely packed devices entering the ultra large scale integration era (ULSI).

2.2.1 Technology Scaling

There are two methods for obtaining scaling: constant voltage (CV) scaling and constant electric (CE) field scaling. The dimensions of the MOSFET are scaled by a factor 'S' in constant-field scaling, with the goal of preserving the magnitude of the internal electric fields, particularly in device channels. The power supply voltage is also proportional to the size of the device features. Constant electric field scaling is used to improve device reliability and performance. The device dimensions are reduced by a factor of 'S' in constant-voltage scaling while the power supply remains constant[5]. This scaling method is used in CMOS circuits to achieve advanced performance while maintaining transistor-transistor logic compatibility. Power and delay are much more important issues to analyze when we scale down any of the devices under the name of technology.

Constant-field scaling is an excellent framework for device scaling that does not compromise reliability. However, several parameters, such as the thermal voltage and the energy gap of silicon material, cannot be scaled with the reduced voltage and dimensions, posing design challenges. The threshold voltage is another important device parameter that does not scale well. Because a guard margin between the two parameters is required for reliable device operation, this constrains the lower limit of power supply voltage. The scaling of leakage current and the sub-threshold slope are two other parameters that pose difficulties for this method.

Constant-field scaling also results in the greatest reduction in an individual transistor's power-delay product. However, when the minimum feature size is reduced, the power supply voltage must be reduced, making scaling a very difficult task due to the external limitations of the power supply.

One issue with constant voltage scaling is that the electric field in the channel increases as the gate length decreases, resulting in velocity saturation, mobility degradation, increased leakage currents, and lower breakdown voltages, all of which can lead to serious reliability issues such as hot-carrier degradation, electromigration, and oxide breakdown.

2.3 Sources of Power consumption

Scaling of CMOS technology improved the speed nevertheless the leakage currents are left-over as an adverse effect. The problem has taken a serious turn as the scaling extends into ultra-deep-submicron (UDSM) region. The total power dissipation in a CMOS circuit can be

expressed as the sum of two main components: dynamic power (power dissipated when the circuit is switching.) and static power (power dissipated when the circuit is idle).

2.3.1 Dynamic power dissipation

Dynamic power dissipation is the power consumed when there is any switching activity in a CMOS circuit. It is the power required to charge and discharge all nodes in a CMOS circuit. This power is only dissipated when the input signals in the circuit change. Dynamic power dominates total power dissipation in CMOS circuits. This characteristic is greatly influenced by current processes or deep sub-micron processes (DSM), where the leakage power to dynamic power ratio is increasing[6]. Toatl dynamic power dissipation of a circuit with n nodes is evaluated as follows:

$$P_{dynamic} = V_{dd} \times V_{dd} \times f \times (\sum_{i=1}^{n} \alpha_i \times C_i)$$

where α_i is the switching activity of node i with capacitance C_i

2.3.2 Static Power dissipation

Since the PMOS and NMOS devices are never on together at the same time in steady-state operation, the static power consumption of static CMOS circuits is assumed to be zero. However, once the gate voltage falls below the threshold voltage, the drain current through the CMOS transistor does not drop to zero. Unfortunately, there is always a leakage current, which is primarily determined by fabrication technology. Historically, static power dissipation has made a relatively insignificant contribution to overall power dissipation and can therefore be neglected. However, with today's technologies, which use lower device threshold voltages to achieve better performance, the percentage of static power dissipation from total power has increased.

2.3.3 Short Circuit Power Dissipation

The dynamic power dissipation assumes the NMOS and PMOS are never simultaneously on i.e., the rise and fall times are zero. In reality, however, such an assumption is incorrect, and input signals have nonzero rise and fall times. As a result, a direct current path exists between Vdd and GND for a brief period of time during input switching, during which the PMOS and

NMOS devices are both conducting. This power component is consumed without contributing to circuit behaviour and is thus deemed redundant.

2.3.4 Trends in Power Dissipation

Generally, dynamic power was the dominant component and static power was negligible (Borkar, 1999). But this will totally reverse in the case as the CMOS technology scales down. Static power dissipation is increasing with each new technology node moving towards to smaller nodes. The Figure.2.1 below shows the leakage trends in CMOS devices, where the leakage power dissipation is going beyond Dynamic power dissipation[7].

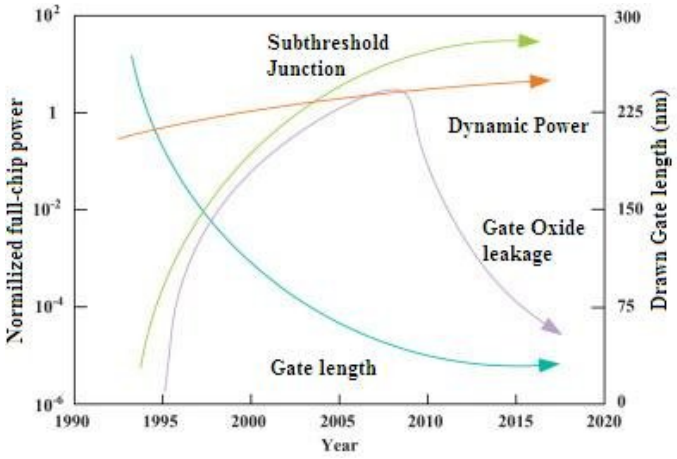


Figure 2.1: Dynamic and Static power consumption trends based on ITRS

Chapter 3

Impact of PVTA variations on Performance of Digital Circuits

3.1 Introduction

The process and operating variations greatly affect the performances in the CMOS digital circuits particularly in the smaller technology nodes (<90nm). Aging effects degrade the performance of the circuits over time. The objective is to understand the impact of the process, operating and Aging variations and optimize the performances (Leakage power and Propagational delay) of the circuit invariant to these variations.

3.2 Process Variations

Technology scaling in the nanometer era has increased the transistor's susceptibility to process variations. The effects of such variations are having a massive impact on the yield of the integrated circuits and need to be considered early in the design flow. The variations in the device and interconnect parameters such as device threshold voltage (V_{th}), oxide thickness (t_{ox}), wire width (W_M), and wire height (H) are growing at an alarming rate. Subsequently, the performance of a different die on the same wafer can also vary widely, resulting in a significant parametric yield loss.

As the MOSFET geometries continue to shrink, controlling critical device parameters becomes increasingly difficult, resulting in significant variations in device length, doping concentrations, and oxide thicknesses. These process variations are a significant issue, and the device's operation is no longer deterministic, but rather random. Process variations can be

classified as systematic or random, with systematic variations being deterministic and caused by the structure of a specific gate and its topological environment.

For instance, depending on the density of the surrounding routing, wire thicknesses will polish differently during CPM. Due to lithography limitations and the use of OPC methods, poly gate width has a deterministic dependence on the spacing of neighboring poly lines. Random variations in device length, discrete doping fluctuations, and oxide thickness variations are all unpredictable in nature. The impact of deterministic variations on circuit delay is relatively simple to analyze if accurate models of their dependence on physical topologies and the necessary layout information are available at the time of analysis.

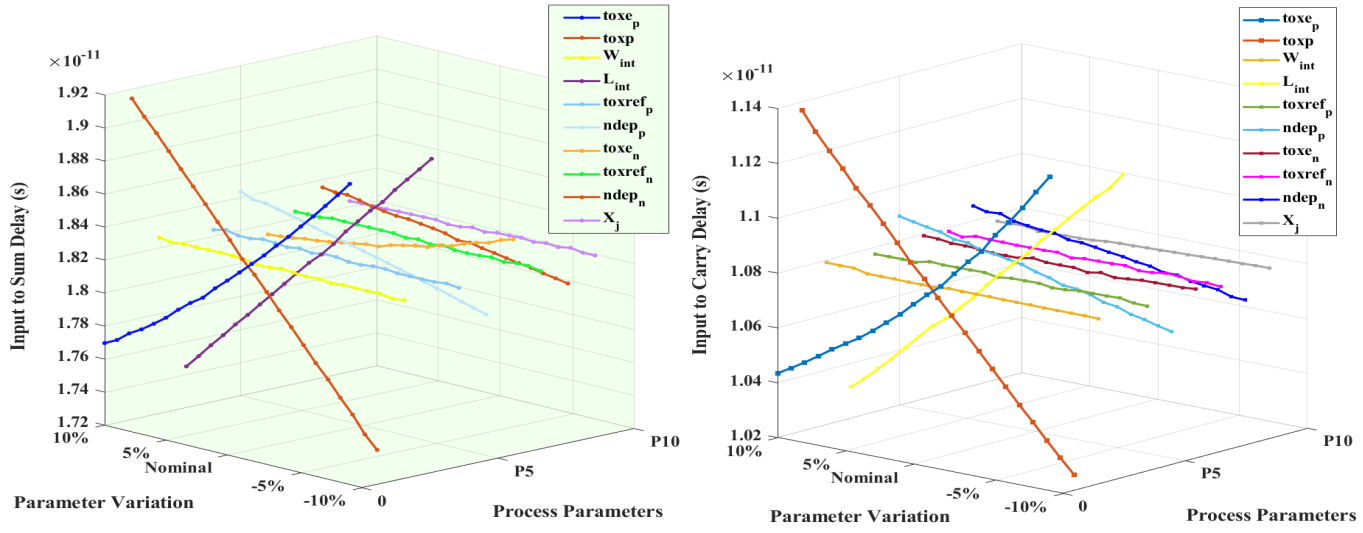
Process variations are further divided up into inter-die and intra-die variations. Intra-die variations are disparities in device features that exist within a single chip, implying that a device feature differs between different locations on the same die. Intra-chip variations frequently exhibit spatial correlations, with devices placed close to each other having a higher probability of being alike than devices placed far apart. Intra-die variations also exhibit structural correlations, which means that devices that are structurally similar have a higher likelihood of having similar device features, such as devices oriented in the same direction. Inter-chip variations are disparities that occur from one die to the next, implying that the same device on a chip has different features from one wafer to the next, wafer to wafer, and wafer lot to wafer lot. With increased process scaling, intra-chip variations are becoming a larger part of overall device feature variability, which means that devices on the same die can no longer be treated as identical copies of the same device. The impact of process parameters on Performance parameters of 1b-Full Adder circuit is represented in Fig.3.5

3.3 Operating Variations: Supply Voltage and Temperature

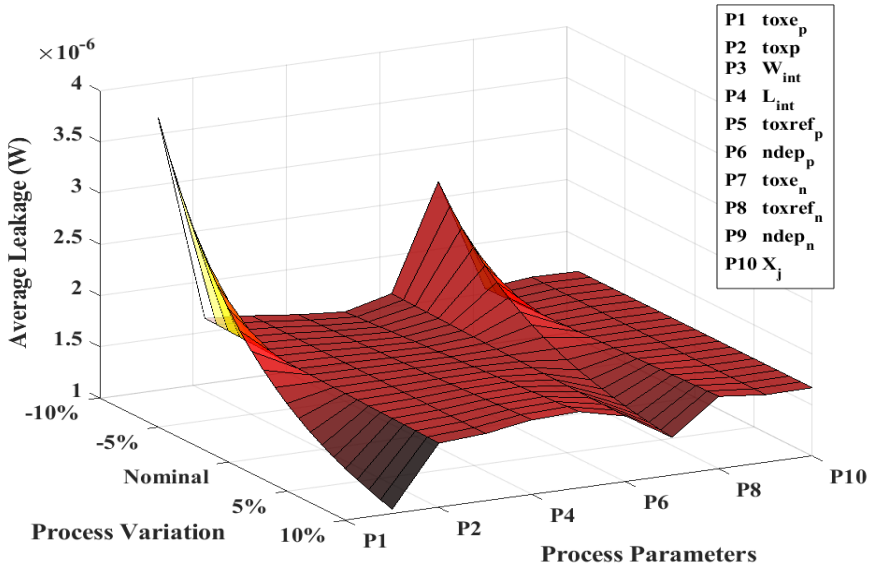
The operating variations like temperature and supply voltage cannot be neglected and have a significant impact on the performance parameters of the circuit.

3.3.1 Supply Voltage

From the previous sections, it is evident that power dissipated is linearly dependent on supply voltage i.e., reducing the supply voltage can reduce the total power dissipated. However, this reduction adversely affects the overall logic delays. Signal traces within an integrated circuit have capacitance, which is influenced by factors such as trace length. When the signal's state changes (from a logic 1 to a logic 0 or vice versa), the driving transistors must transfer



(a)



(b)

Figure 3.1: Impact of Process variations on (a) Critical Path Delay and (b) Average Leakage for a Full adder

current until the desired new state is reached. The lower the supply voltage available for the IC, the longer it takes to drive the signal [8], Hence, increasing the propagational delay of the IC.

3.3.2 Temperature

Environmental Temperature fluctuations can cause significant variations in the device performance as it alters the die temperature. Electronic systems mounted on automobile engines, for instance, operate in a temperature range of 40°C to 100°C. Temperature variations affect the device characteristics of MOSFETs, allowing the performance of the IC to vary. Leakage power increases exponentially with fluctuations in temperature [8].
The impact of the Supply voltage and Temperature on Leakage and critical path delay is depicted in Fig.3.2

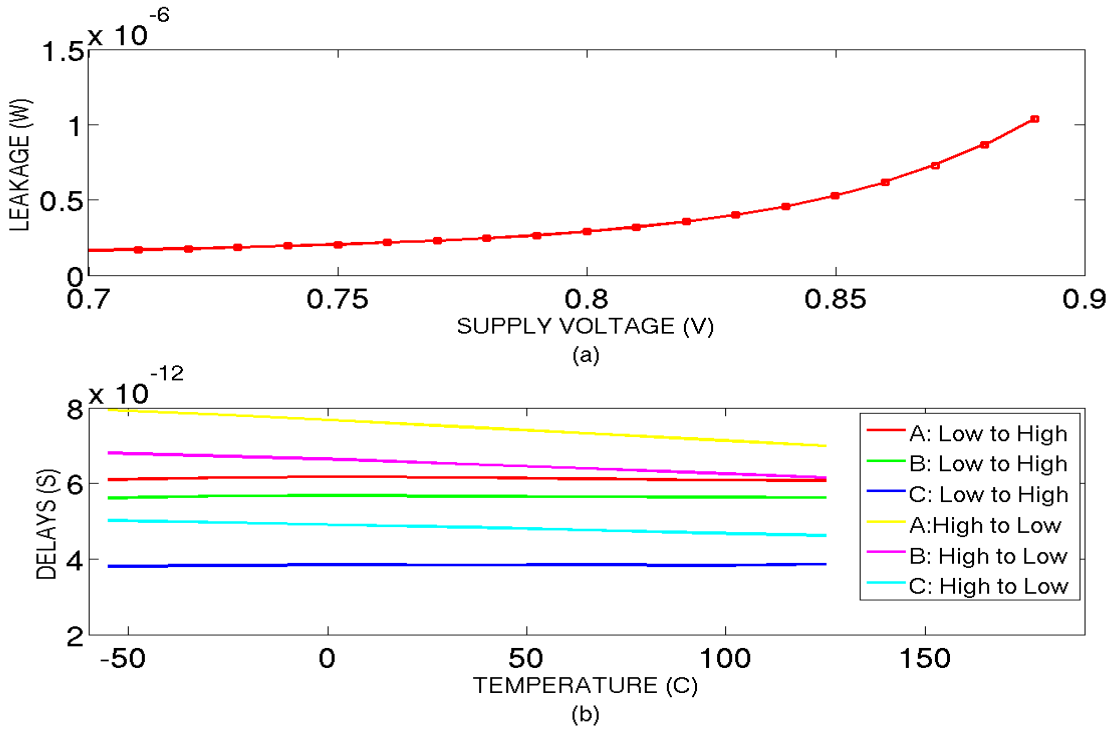


Figure 3.2: Impact of (a) Supply voltage on Leakage, (b) Temperature on critical path delay of 1b-Full Adder

3.4 Aging Variations

Aging variations(NBTI) change the initial performance parameters gradually over time. The charge carriers in the channel get trapped in the insulating dielectric (silicon dioxide), raising the electric field and interface traps are created over time. This process eventually builds electric charge in the dielectric, and in turn, increases the gate voltage required to turn on the MOSFET, i.e., Threshold voltage(V_{TH}). This increase in Threshold Voltage affects the switching of the transistor and makes the device slow over time as shown in Fig.3.3. Due to this increase, aging variations favor the leakage power while the delays are degraded.

This is worsened when the NBTI aging effects are severely impacted by the magnitude of process variations in parameters like oxide thickness(T_{ox}), the effective channel length (L_{eff}), the effective channel width (W_{eff}), and the zero-biased threshold voltage (V_{th0}) [2].

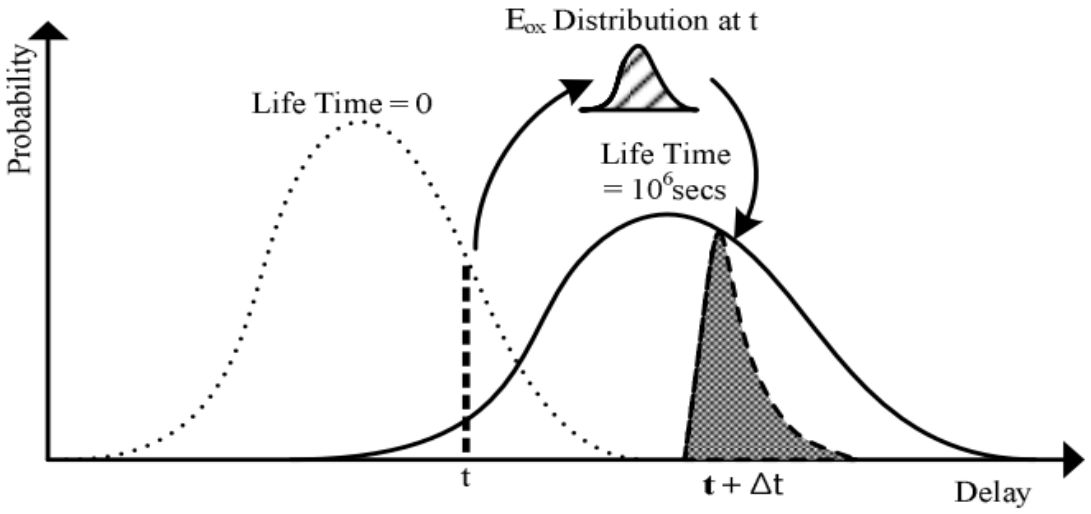


Figure 3.3: The change of gate delay due to NBTI at D=t [2]

3.5 PVTA range considered for the optimization

Subnanometer technology nodes have made possible highly complex ICs with superior integration of functionality. Nonetheless, migration to these regimes has increased the static power losses and rendered the circuits highly sensitive towards process parameters (P), supply voltage (V), temperature (T), and aging (A) variations. In this work, 10 significant process parameters are considered like channel-length and channel-width offset parameter (lint, wint), physical

gate equivalent oxide thickness (toxp), electrical gate equivalent oxide thickness ($toxe_n$, $toxe_p$), nominal gate oxide thickness for gate dielectric tunneling current ($toxref_n$, $toxref_p$), junction depth (xj) and channel doping concentration at depletion edge ($ndep_n$, $ndep_p$). The process parameters are considered at 3σ value of $\pm 10\%$ variation from the nominal value of the process parameters mentioned in Table 3.1.
The temperature range of [-55 °C to 125 °C] has been considered for optimization, with $\pm 10\%$ variation in supply voltage i.e. [0.72 to 0.88] and 3 years of NBTI aging. The impact of the Process, operating and Aging variations on Critical path delay and Average leakage power of 1b-FUll Adder circuit is represented in Fig.3.4 and Fig.3.5

Sr.no	process parameter	Lower deviation	Nominal value	Higher deviation
1	$lint$	1.2307e-009	1.35e-009	1.489e-009
2	$wint$	4.4868e-009	5e-009	5.4849e-009
3	$toxp$	3.658e-010	4e-010	4.393e-010
4	$toxe_n$	5.8436e-010	6.5e-010	7.27e-010
5	$toxe_p$	5.978e-010	6.7e-010	7.487e-010
6	$toxref_n$	5.743e-010	6.5e-010	7.251e-010
7	$toxref_p$	6.08e-010	6.7e-010	7.429e-010
8	xj	6.523e-009	7.2e-009	7.845e-009
9	$ndep_n$	1.05e+019	1.2e+019	1.304e+019
10	$ndep_p$	3.879e+018	4.4e+018	4.869e+018

Table 3.1: **Process variations considered for 22nm CMOS technology**

Throughout this thesis, we refer to

- Nominal operating conditions (NOC) as Temp=25 °C and V_{dd}= 0.8V
- Worst case operating conditions (WCO) as Temp [-55°C - 125°C] and V_{dd} = [0.72 - 0.88]V.
- PVT as $\pm 3\sigma$ variation in process parameters at WCO.
- PVTA refers to PVT with 3 years of NBTI degradation.

3.6 Literature Survey

Several approaches have previously been proposed to optimize the primary performance parameters. Channel Engineering techniques [9] like retrograde doping and halo doping reduce

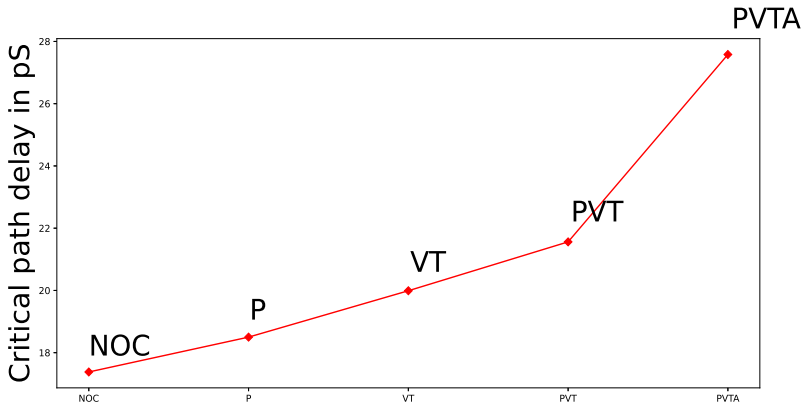
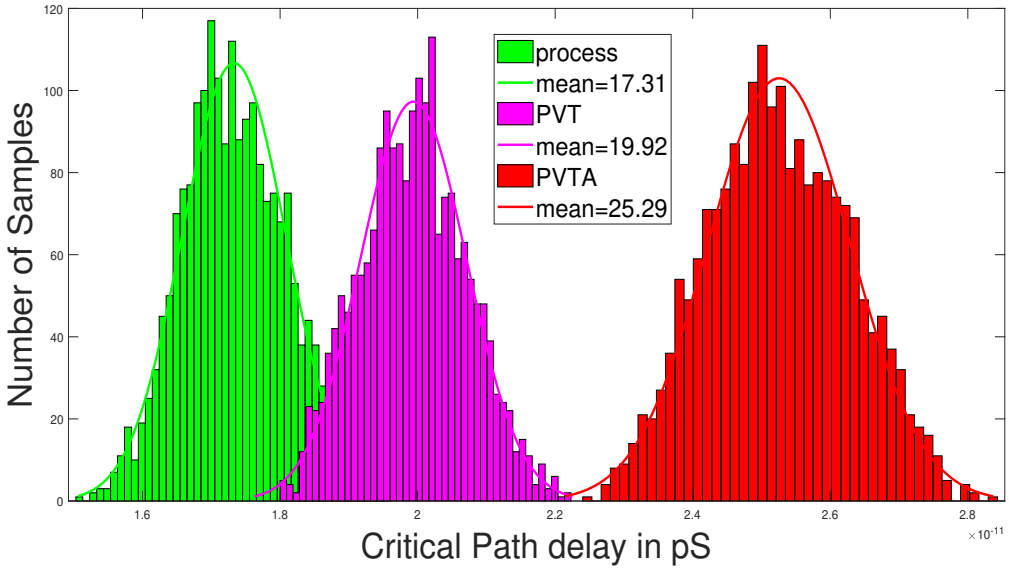


Figure 3.4: Impact of Process, Operating and Aging variations on critical path delay of Full adder at nominal sizing

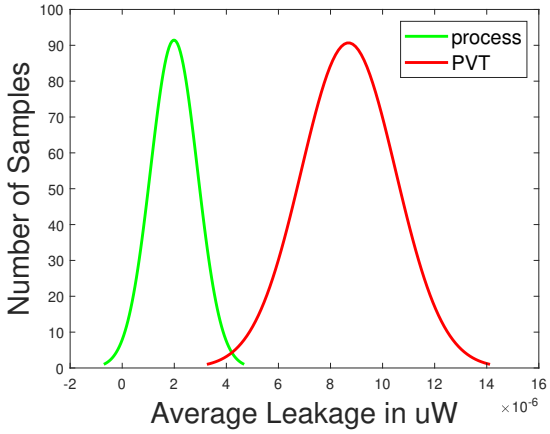
the power while, other conventional techniques, for example, dual-threshold CMOS, Multi-threshold CMOS, Power gating, Input Vector Control (IVC) [10], [11] reduce the power in standby and active modes. Leakage power analysis on nano-scale technologies is performed in [12] for low-power and high-performance applications.

However, transistor sizing is the most prominent technique and has many advantages which include reducing the glitches [13], balancing the load across multiple delay paths [14], parametric yield calculation [15], [16], and optimizing the performances [17] [18]. Ref [19] studied the impact of aging variations, while [20] considers the aging variations and optimizes the performances using ILP based algorithm.

Algorithm-based transistor sizing has been a salient technique to optimize the performances, but many of the previous works did not consider all the PVTA constraints in their work. [21] optimizes the CMOS circuit using the gradient-based algorithm and portrays a detailed study of the performance deviations and the extent of optimization with and without considering PVT variations. [22] focuses on the optimization of critical path delays but does not consider the PVTA constraints. [23] & [24] use NSGA-II and PSO algorithms to optimize analog circuits. Popular algorithms such as Genetic Algorithm (GA) [25], Particle swarm optimization (PSO) [26], NSGA-II and other swarm-based algorithms [27], [28] have been used in various scenarios obtaining appreciable optimizations but doesn't consider PVT variations. Genetic algorithms are time-consuming while swarm-based algorithms overcome the runtime issue [26].



(a)



(b)

Figure 3.5: Impact of Process variations on (a) Critical Path Delay and (b) Average Leakage for a Full adder

In [29] & [30], Process variations have been considered, but the aging effect is not considered, that too the process variations considered are for only the corner cases. [31] optimizes leakage only for standby mode. Moreover, in [29], [30] & [31]

3.7 Summary

The miniaturization of MOS devices to lower technology nodes made process variations significant in fabrication. This disparity in process parameters (e.g., tox, VT, junction depth, etc.) has become acute as it leads to an increase in leakage power as well as propagation delays. This poses a serious problem in meeting the desired timing and power criteria of the present day low power and high performance circuits. In our proposed optimization methodology, we have considered 10 significant process parameters at 3σ variation for 22nm MGK technology.

In addition to process variations, operating variations, and aging degradation are other factors that deviate the circuit's performance (leakage power and delays). Any deviation in operating parameters like supply voltage and temperature may cause a huge variation in leakage power and propagation delays. To meet the required circuit's specifications, all such variations have to be taken into account to avoid any discrepancy in the circuit's performance.

Chapter 4

Circuit Analysis

4.1 Introduction

Circuit analysis is performed for a better understanding of the response of the circuit on changing various parameters like design parameters, local and global process parameters, the correlation between process parameters, etc. These results were taken into consideration to ease the optimization process and to obtain better results. We have presented our analysis for the 28 transistor mirror full adder shown below in Fig.4.1

4.2 Sensitivity Analysis

The term sensitivity refers to the impact of changing parameter values on system functionality. Sensitivity analysis allows you to determine which system parameters have the greatest influence and which have the least. Knowledge of sensitivity allows the designer of VLSI circuits to identify the number of variables that have a critical influence on the operation of the circuit[32].

As mentioned earlier varying widths and lengths can have a significant impact on leakages and delays. The impact of each independent design parameter on all the dependent performances is nothing but sensitivity analysis. With sensitivity analysis for each parameter i.e., the Length and Width of each transistor in a digital circuit, the significant design parameters can be focused to reduce the computational time with almost similar accuracy.

We have performed a sensitivity analysis for the 28T Full adder circuit in Fig.4.1 considering the length and width of each transistor as design parameters, hence, a total of 56 design

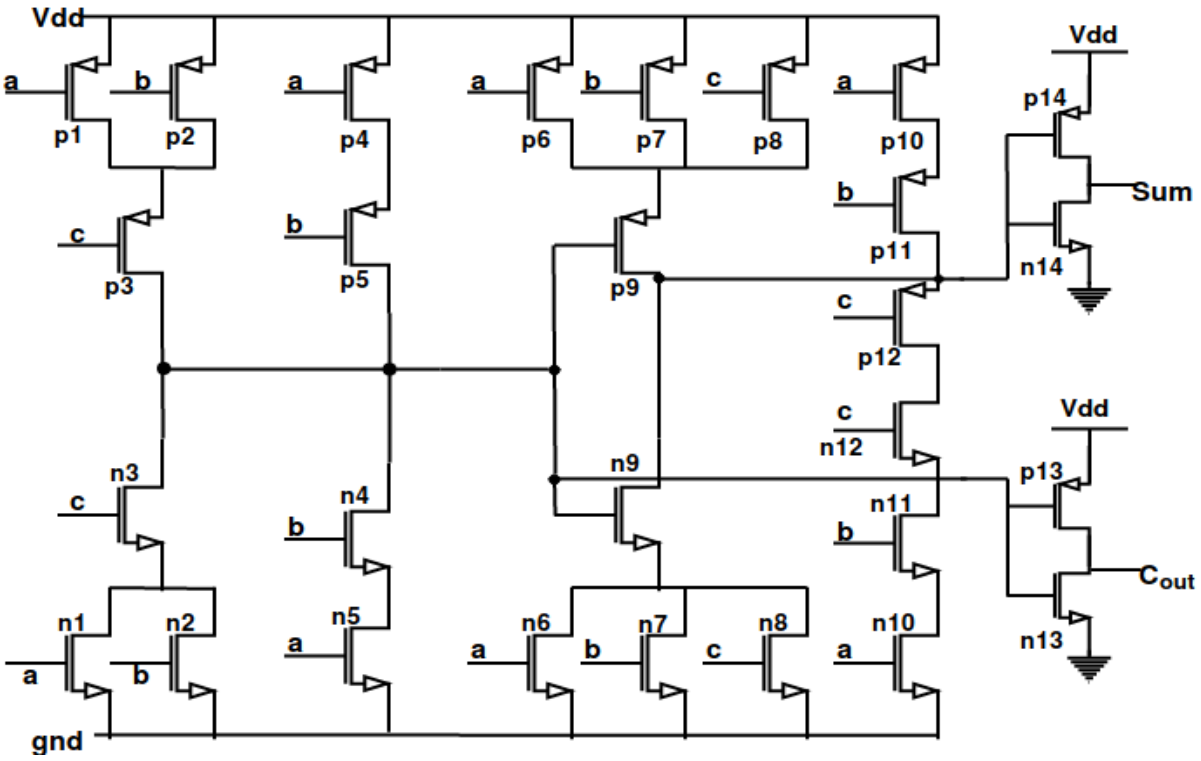


Figure 4.1: CMOS 28T Full Adder Cell

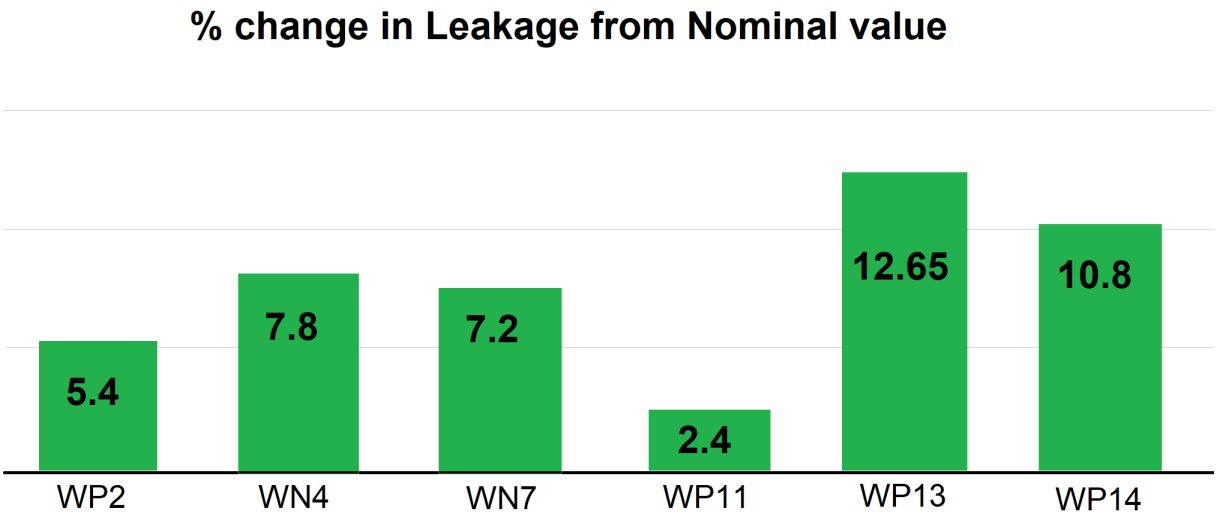
parameters.

The sensitivity of a parameter(p) is calculated by varying it by a small amount Δp, i.e., by sweeping the design parameter as shown in equation 4.1. The resulting performance difference $\Delta f = f(p + \Delta p) - f(p)$ is used to calculate the sensitivity as $\Delta f / \Delta p$.

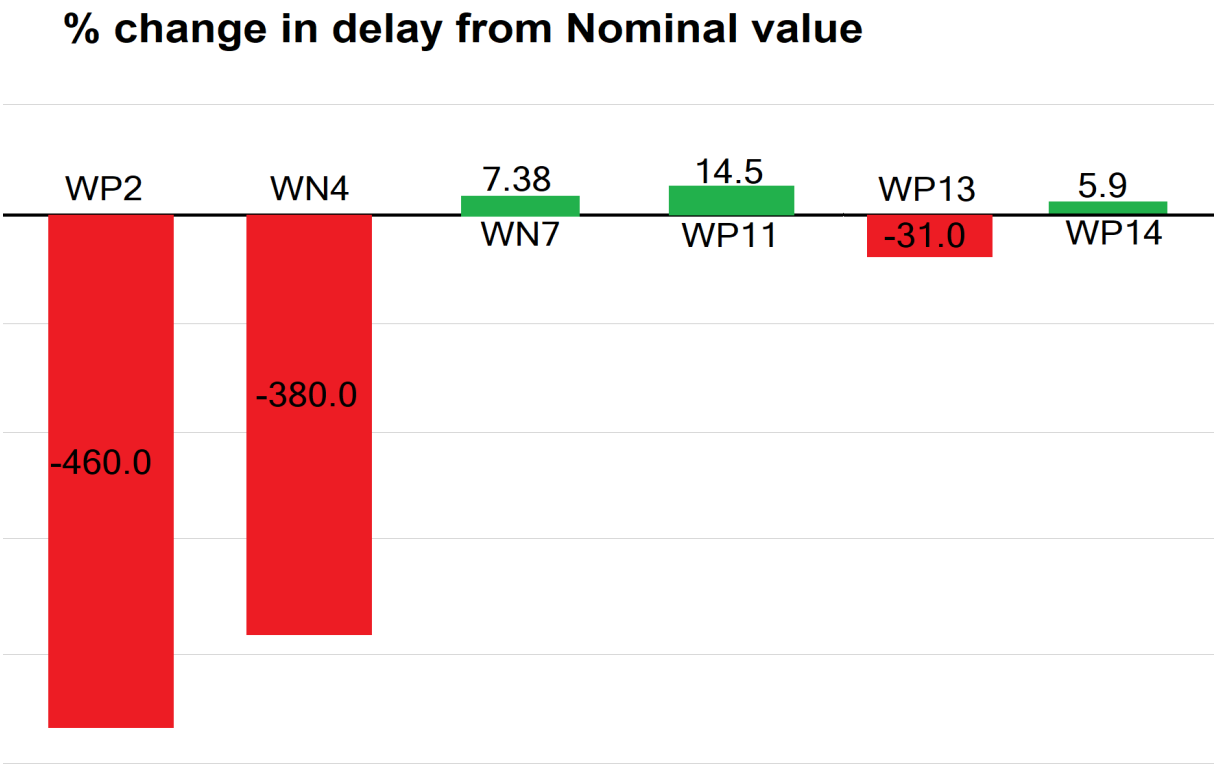
$$Sensitivity(\Delta p) = f(p + \Delta p) - f(p)/\Delta p \tag{4.1}$$

Fig(4.2a) and (4.2b) represent the sensitivity percentages of various design parameters of 28T Full adder on leakages (a) and delays (b). Table 4.1 reports the sensitivity in percentage for Widths of all the 28 transistors mentioned in Fig.4.1.

Power, area, and delay being some of the critical performance metrics, their optimization holds an integral position in the design flow. However, irrespective of the design parameters taken, their mutually contrasting feature prohibits simultaneous optimization for all three. Moreover, migration to deep sub-micron regimes enhances circuit sensitivity towards PVT variations.



(a) Average Leakage



(b) Critical Path Delays

Figure 4.2: Sensitivity of design parameters for 28T Full adder

4.3 Screening

Screening is the process of reducing or nullifying the effect of insignificant parameters. From the above sensitivity analysis, the parameters that have less impact are screened out, and only the parameters that have a significant impact are considered. This reduces the number of design variables and, in turn, the number of constraints in the algorithm, due to which the number of iterations and computational time drastically reduces. From the Table 4.1, the parameters that are screened out are W_{P10}, W_{N11}, W_{P11}, W_{N12} and W_{P12} for low power optimization considering 5% as the maximum bound to be screened. While for High performance optimization, W_{N6}, W_{P6}, W_{N7}, W_{P7}, W_{N8}, W_{P8}, W_{N10}, W_{P10}, W_{N11}, W_{P11}, W_{N12}, W_{P12}, W_{N14} and W_{P14} are screened out. We have noticed a 64% reduction in computational time without any effect on optimization.

4.4 Mismatch Analysis

Mismatch analysis is the investigation of the effect of intra-die process variations on the circuit's performance parameters. A mismatch can be thought of as spatial noise spread across the surface of a chip. The main effects of mismatch on system performance are reduced dynamic range due to increased spatial noise, precision limitation, increased area, and increased power dissipation. When designing circuits, all of these constraints must be traded off against each other.

Mismatch in CMOS circuits is induced by three primary factors. The first is the variation in physical device dimensions. The only way to minimize this effect is to use large devices that can neglect the effect of variation, which often occurs at the edges of the device. The metallurgical variation of device parameters, which primarily includes the variation of doping densities in the semiconductor, is yet another source of mismatch. Using large-size transistors can also help to reduce this type of mismatch. The device's electronic parameters are the third source of mismatch. For example, trapped charges in the gate oxide or surface states in a MOS transistor can alter the device's threshold voltage. Among these, the third is discovered to be more prevalent in MOS transistors.

Mismatch analysis can be performed by applying constant inputs to the circuit and assuming mismatch levels for each transistor's threshold voltage. We have taken into account two corners of the threshold voltage with $\pm 5\%$ variation from the nominal. We concluded from this analysis that the mismatch effect on delay and leakage is insignificant and thus negligible.

4.5 Correlation

Variations in process parameters result in the variation of performance parameters to a great extent. Since the process parameters are also inter-dependent, the correlation between process parameters has to be considered before optimizing the circuit. Pearson Correlation Coefficient(ρ) between any two variables is measured as:

$$\rho = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}} \tag{4.2}$$

Where n is the number of samples, x and y are performance parameters. The correlation percentage($\rho * 100$) for a few combinations of process parameters is shown in Fig(4.4)

According to [33], BTI and process parameters are inter-dependent. Since we have considered NBTI aging effects, the correlation between NBTI and process parameters is also taken into account. Fig(4.3) shows the scatter plot for NBTI and toxp.

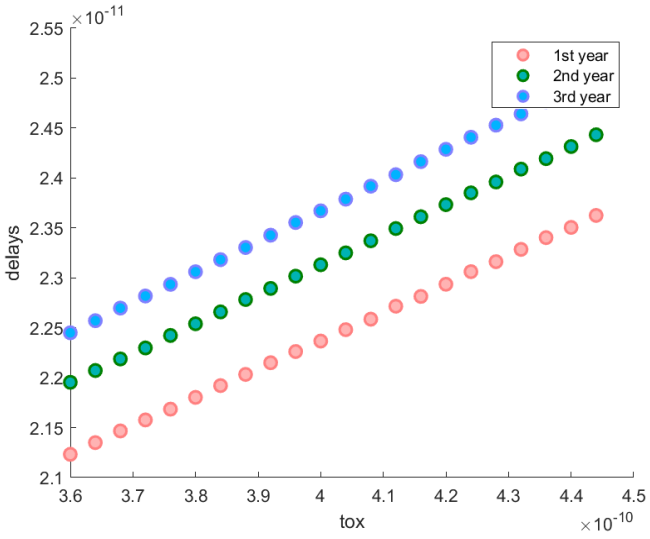


Figure 4.3: Correlation of aging and toxp

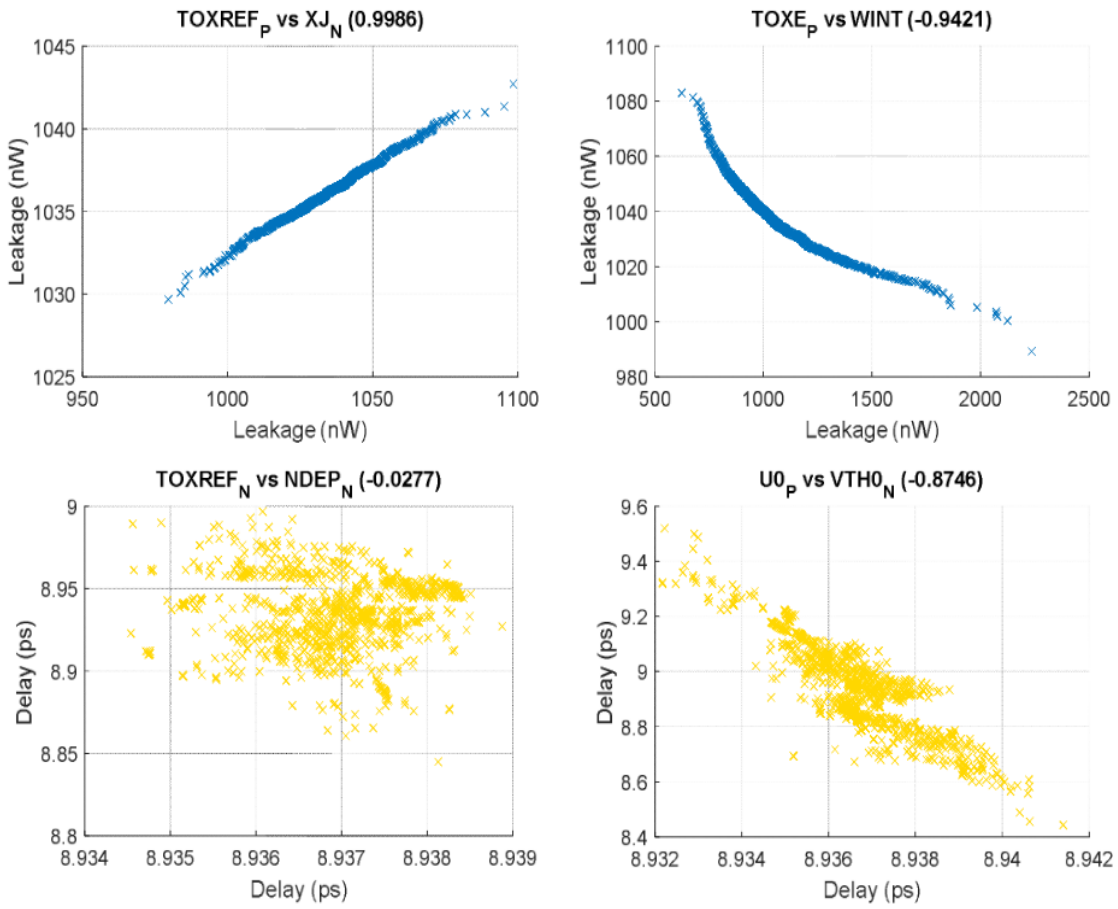


Figure 4.4: Correlation between few Process Parameters

S.no	Design parameter(Width)	Sensitivity for Leakage	Sensitivity for Delay
1	W_{p1}	5.4%	-457%
2	W_{n1}	7.2%	-100%
3	W_{p2}	5.4%	-413.8%
4	W_{n2}	7.2%	-73%
5	W_{p3}	6.6%	-349%
6	W_{n3}	8.4%	-75%
7	W_{p4}	5.4%	-467%
8	W_{n4}	7.8%	-69%
9	W_{p5}	5.4%	-380%
10	W_{n5}	7.2%	114%
11	W_{p6}	5.4%	-0.05022%
12	W_{n6}	7.2%	0.0177%
13	W_{p7}	5.4%	0.0727%
14	W_{n7}	7.2%	0.094%
15	W_{p8}	5.4%	-0.016%
16	W_{n8}	7.2%	0.0083%
17	W_{p9}	8.4%	14.5%
18	W_{n9}	9.6%	7.38%
19	W_{p10}	3.6%	0.054%
20	W_{n10}	6.02%	-0.06519%
21	W_{p11}	2.4%	-0.0366%
22	W_{n11}	3.6%	-0.085%
23	W_{p12}	3.01%	-0.269%
24	W_{n12}	4.8%	-0.0759%
25	W_{p13}	10.81%	5.9%
26	W_{n13}	12.65%	-31%
27	W_{p14}	10.81%	-0.09%
28	W_{n14}	12.65%	0.19%

Table 4.1: **Process variations considered for 22nm CMOS technology**

Chapter 5

Proposed Algorithm methodology using Transistor sizing to optimize performances of CMOS digital circuits

5.1 Introduction

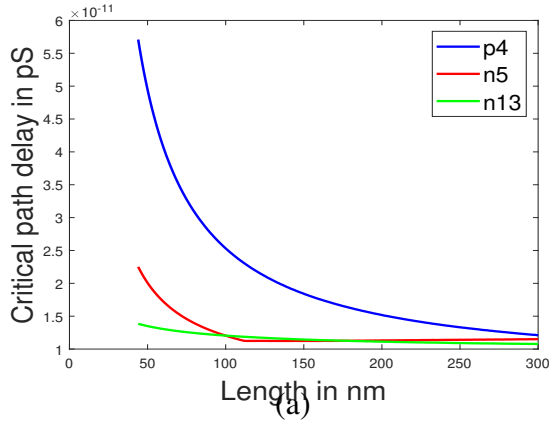
Considering all PVTA constraints, optimizing leakage power and delay becomes the bottleneck for the designers. Varying the design parameters (Widths and Lengths) is an effective way to optimize delays and leakages. However, with the increasing complexity of the circuits, it is almost impossible for a human to manually change the design parameters. Therefore, we have proposed an algorithm-based optimization scheme to find the optimal width and length values that are robust against all PVTA variations. Fig. 5.1 clearly shows that the delays and leakage may have similar/contrary dependence on the dimensions of an individual MOS device, providing the design space to find the optimal sizing of all MOS devices at which the CUT is robust against all variations and working for the intended lifetime.

5.2 Pareto Harris Hawk optimization algorithm

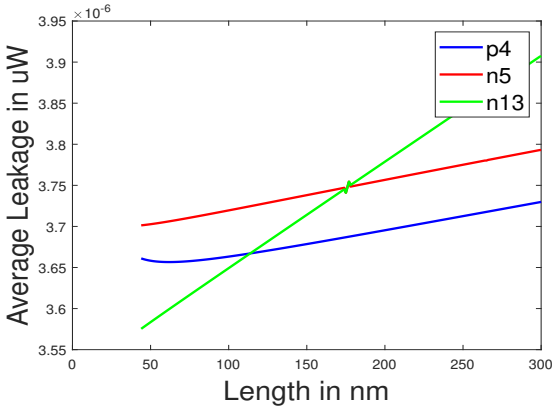
Harris hawks optimization is a single objective, population based algorithm. This algorithm is inspired by the exploration methods and pouncing strategies of harris hawks. From the basic algorithm proposed in [34], we introduce a variable c to convert our two objectives leakage and delays into a single objective function (f) as

$$f = c * max(delays) + (1 - c) * average(leakages) \tag{5.1}$$

In eq(5.1) c=1 optimizes delays ignoring leakages and c=0 optimizes leakages ignoring delays. values of c between 0 and 1 give other points. We take the logarithm of objective



(a)



(b)

Figure 5.1: Change in delay/leakage values Vs. MOS dimensions

function as the values of the objectives are very small. Leakages are in the order of 10^{-6} while delays are in the range of 10^{-12}. Taking a logarithm normalizes these values. Due to this, the linear combination tends to increase weightage for one objective creating bias. Hence we use exponentially varying weightage values to get even distribution of points to counter this effect. That is, instead of C varying as 0.1, 0.2, 0.3,... incrementation is done by eq(5.2). The value of k changes from 3-6 to get various points of optimality.

$$c_i = 1 - 1/k^i \tag{5.2}$$

This algorithm has 3 main phases. Exploration phase, Transition phase and Exploitation Phase. In Exploration Phase, they search for the Prey based on probability q. Each Hawk is a solution and the best Hawk in step is the solution of that step. Their behavior is modeled by eq(5.3) where X(t+1) is the position in the next iteration, $H = (LB + r_4(UB - LB)$ and r_1, r_2, r_3 and r_4 are random numbers inside (0,1).

$$X(t+1) = \begin{cases} X_{rand}(t) - r_1|X_{rand}(t) - 2r_2X(t)|, & \text{if } q \geq 0.5 \\ (X_{prey}(t) - X_m(t)) - r_3H), & \text{if} q < 0.5 \end{cases} \tag{5.3}$$

Energy of Harris hawks(E) is calculated as $E = 2E_0(1 - \frac{t}{T})$, where E_o changes randomly in interval (-1,1). Hence, Harris Hawks search for the Prey if $|E| \geq 1$ and when $|E| < 1$ they go to Exploitation phase.

In the Exploitation Phase, Harris Hawks perform surprise pounce. This phase has 4 cases based on the escaping probability of the prey r (r≥0.5, the prey escapes) and the Escaping energy E.

5.2.1 Case1:

When r≥0.5 and $|E| \geq 0.5$, the behaviour of Harris Hawks is modeled as shown below in eq(5.4), Where $J = 2(1 - r_5)$ and is a random number between 0 and 1.

$$\begin{aligned} X(t+1) &= \Delta X(t) - E|JX_{prey}(t) - X(t)| \\ \Delta X(t) &= X_{prey}(t) - X(t) \end{aligned} \tag{5.4}$$

5.2.2 Case2:

When r≥0.5 and $|E| < 0.5$, the prey is exhausted and has low Energy. Harris Hawks hardly encircle the prey and perform surprise pounce. This is modeled below by eq(5.5).

$$X(t+1) = X_{prey}(t) - E|\Delta X(t)| \tag{5.5}$$

5.2.3 Case3:

When r<0.5 and $|E| \geq 0.5$, they prey will have enough energy to escape and Harris Hawks perform rapid dives to correct their location with respect to the deceptive location of the prey. This is called Levy Flight(LF) and is modelled by the eq(5.6). Where D is the dimension and S is vector of Dimension 1xD, below mentioned u and v are random values between 0 and 1, β is set to 1.5

$$\begin{aligned} Y &= X_{prey}(t) - E|\Delta X(t)| \\ Z &= Y + S * LF(D) \\ LF(x) &= 0.01(\tfrac{u\sigma}{|v|^{\frac{1}{\beta}}}) \\ \sigma &= (\tfrac{\gamma(1+\beta)xsin(\frac{\pi\beta}{2})}{\gamma(\frac{1+\beta}{2})\beta 2^{\frac{\beta-1}{2}}})^{\frac{1}{\beta}} \\ X(t+1) &= \begin{cases} Y & \text{if} F(Y) < F(X(t)) \\ Z & \text{if} F(Z) < F(X(t)) \end{cases} \end{aligned} \tag{5.6}$$

5.2.4 Case4:

r<0.5 and $|E|$ <0.5, in this case the prey has low energy and the Harris Hawks hardly encircle the prey. They attack and kill the prey. This is modelled as below: eq(5.7), where $X_m(t)$ is the average of positions of hawks.

$$\begin{aligned} Y &= X_{prey}(t) - E|X_{prey} - X_m(t)| \\ X(t+1) &= \begin{cases} Y & \text{if} F(Y) < F(X(t)) \\ Z & \text{if} F(Z) < F(X(t)) \end{cases} \end{aligned} \tag{5.7}$$

Algorithm 1: Pareto Harris Hawks optimization Algorithm

```
Set Number of iterations(i) and Population Size (N);
Initialize Random Population;
for number of iters i do
    Calculate fitness of Hawks using Eq.5.1;
    Set X_prey as the best solution;
    for each hawk do
        Update E;
        if |E| ≥ 1: Exploration Phase then
            Use Eq.5.3 ;

        else
            Exploitation phase ;
            if r ≥ 0.5|E| ≥ 0.5 : Case1 then
                Use Eq.5.4 ;

            else if r ≥ 0.5|E| < 0.5 : Case2 then
                Use Eq.5.5 ;
            else if r < 0.5|E| ≥ 0.5 : Case3 then
                Use Eq.5.6 ;
            else
                Use Eq.5.7 ;
    Update X_prey with the location of Hawk with best fitness;
```

5.3 Neighbourhood Cultivation Genetic Algorithm

Neighbourhood Cultivation Genetic Algorithm (NCGA) [35] is an extension of the Genetic algorithm. This includes the mechanisms of NSGA-II [36] and SPEA-II [37] as well as the neighbourhood crossover mechanism. If the distance between two parents is more, then the crossover may have no effect on local search. Unlike the crossover in other algorithms like GA, NSGA-II and SPEA-II, NCGA selects the neighbours for crossover instead of randomly chosen individuals. Thus the resulting child individuals have more traits like their parents. This algorithm is initialized with a random population of size N, and then they are sorted according to the focused objective. The objective changes in each iteration. If there are 8 objectives to be

optimized, the first objective is chosen in the first iteration, the second objective in the second iteration and eighth objective in the eighth iteration and the first objective again in the ninth iteration and so on. From the sorted individuals, the neighbouring individuals are grouped for crossover and mutation. Crossover and mutation are similar to that in SPEA-II. The required objectives are calculated for the child individuals, and the new population of size N is chosen from the child individuals of size N and the current population of size N by environmental selection. These steps are repeated for the new population until the termination criterion is met.

Algorithm 2: Neighbourhood Cultivation Genetic Algorithm

```
Set Number of iterations(i) and Population Size (N);
Initialize the Population for number of iters i do
    for each individual do
        obtain focused obj as per the iteration;
    Sort the Individuals based on the focused objective;
    Group the Individuals with their neighbours;
    Generate the Individuals with crossover and mutation;
    Eval the focused objective;
    Replace the worst individuals with the best ones;
```

5.4 Strength Pareto evolutionary algorithm - II (SPEA-II)

SPEA-II [37] is a multi-objective optimization algorithm that is an extension of the SPEA algorithm proposed in [38]. This algorithm incorporates a fine-grained fitness assignment strategy, a density estimation technique, and an enhanced archive truncation method compared to the previous SPEA algorithm. Initial population P_0 of size N and Archive A_0 of size $N^{'}$ are initialized. In each iteration, the fitness of each individual $F(i)$ is calculated as $F(i) = D(i) + R(i)$ where $D(i)$ is the density and $R(i)$ is the raw fitness and are calculated by eq. (5.8) (5.9) and (5.10).

$$S(i) = |\{j | j \in P_t + \overline{P_t} \wedge i \phi j\}| \tag{5.8}$$

$$R(i) = \sum_{j \in P_t + \overline{P_t}, j \phi i} S(j) \tag{5.9}$$

$$D(i) = \frac{1}{\sigma_i^k + 2} \tag{5.10}$$

Each individual i in the archive $\overline{P_t}$ and the population P_t is assigned a strength value S(i) of eq. (5.8), representing the number of solutions it dominates, where ϕ corresponds to the Pareto dominance relation. On the basis of the S values, the raw fitness of individual R(i) is calculated as eq.(5.9). Density $D(i)$ of individual i is calculated as eq.(5.10), where σ_i^k is the k^{th} element for each individual i the distances to all individuals in archive and population after sorting the list in increasing order. The non-dominated individuals based on fitness from P_t and A_t are moved to A_{t+1}. If the size of A_{t+1} is less than $N^{'}$, A_{t+1} is filled with few of the dominated individuals from P_t and A_t. Whereas, If the size of A_{t+1} exceeds $N^{'}$, some individuals are removed from A_{t+1} using the truncation operator. Crossover and mutation, similar to that in SPEA [38], are performed. These steps are repeated until the termination criterion is met. In this algorithm each individual of the population corresponds to one set of sizing and the dimension of each individual is the total number of Widths and Lengths in the circuit. The focused objective is average leakage for low power applications and Critical path delay for high performance applications.

Algorithm 3: SPEA-II: Improving Strength Pareto Evolutionary Algorithm

```
Set Number of iterations(t);
Initialize the Population(P_0) of size N and create Archive(A_0) of size N';
for number of iters t do
    for each individual i do
        Compute Fitness F(i) of individual i of P_t and A_t as in eq.(5.10);
    Add non-dominated individuals from P_t and A_t to A_{t+1};
    if size of A_{t+1} is less than N' then
        Add dominated individuals from P_t and A_t to A_{t+1};

    else if size of A_{t+1} exceeds N' then
        Remove individuals from A_{t+1} using Truncation Operator;
    Perform binary selection to create mating pool;
    Generate the Individuals with crossover and mutation;
    Evaluate the focused objective;
```

5.5 Glowworm swarm optimization algorithm

Glowworm swarm optimization (GSO) [39] imitates the behaviour of glowworms. This algorithm has four main phases: Initialization, Updating luciferin value, movement and updating local decision range. In the Initialization phase, each glow worm is arbitrarily assigned to an

initial position in the feasible domain and has a random local decision range. The higher the luciferin value, more brighter the glow worm is and hence better the position of the glow worm. Brighter glow worms have a low radius for local decision range and vice versa. The brighter glow worms attract the glow worms in the local decision range. They choose different neighbours every time based on brightness. Hence the direction of movement keeps changing. Then the luciferin value is computed at the new position. The glow worms update their luciferin value based on their position using eq(5.11)

$$l_i(t) = (1-\rho)l_i(t-1) + \gamma J(x_i(t)) \tag{5.11}$$

Based on the Luciferin value the brighter glow worms attract other glow worms in their local decision range. They move in the direction of high Probability. The Probability and direction j are calculated using the below equations eq(5.12) & eq(5.13).

$$P_{ij} = \frac{l_j(t)-l_i(t)}{\sum_{k\in Neighbourhood} l_k(t)-l_i(t)} \tag{5.12}$$

$$j =_i P_{ij} \tag{5.13}$$

They update the local decision range after going to their position. If the Luciferin Value is high, they have a low radius for the local decision range and vice versa, it is also updated in this phase using the below equations. where, in eq(5.14), X_i is the position of i^{th} glow worm and in equation eq(5.15), r_i^d is the local decision range and N_i is the neighbourhood

$$X_i(t+1) = X_i(t) + s\frac{X_j(t)-X_i(t)}{||X_j(t)-X_i(t)||} \tag{5.14}$$

$$r_i^d = \min\left(r_s, \max\left(0, r_i^d(t), \beta(n_t - N_i(t))\right)\right) \tag{5.15}$$

Algorithm 4: Glow Worm Swarm Optimization

```
Set the number of iters and bounds on the position of glowworms;
Initialize the number of glow and their positions: stage-1;
Calculate the Luciferin value based on the Position of each glowworm;
for number of iters do
    for each worm: stage 2 do
        Update luciferin value based on the position of glowworm;
    for each worm: stage 3 do
        Search for the brightest;
        Calculate the probability of movement in all directions and select the best using
        5.12 & 5.13;
        Update the new location using 5.14
    for each worm: stage 4 do
        Update the local decision range using 5.15
Compute coordinates with better luciferin value
```

5.6 Implementation

All the optimization algorithms mentioned above have been developed and tested over several basic standard cells to optimize critical path delay keeping leakage power in bound (for high-performance applications) and vice versa i.e. optimizing leakage power with critical path delay in bound (for low-power applications) with the confidence, the optimized circuit will be robust against all process variations, fully functional for the targeted temperature range and supply voltage and work for the intended year. To perform PVT and aging degradation aware circuit sizing and optimization, the essential requirement is a transistor level net-list supporting both fresh as well as degraded over year device operation. The proposed work uses the MOS Reliability Analysis (MOSRA) tool for pre and post stress simulation integrated within the HSPICE simulator [40][41]. The present optimization results are limited to 22nm CMOS technology node. However, the proposed methodology is equally valid for other technology nodes.

The optimization scheme is a highly simulation intensive task. Therefore, it is always good to first optimize the CUT at nominal operating conditions (Temp=25°C, V_{dd}= 0.8V) i.e. first find the circuit safety margin which ultimately improves computational efforts when all PVTA variations are included.

After the initial part of the analysis, bounds are set on design parameters and performance parameters. In the circuit sizing and optimization of circuits, the design parameters mainly consist of channel lengths(L) and widths(W) of all devices. Such design parameters are tuned, within the bounds mentioned in Table. (5.1)

Design Parameter	Lower Bound	Initial Value (Min)	Upper Bound
L_PMOS	22nm	22nm	25nm
L_NMOS	22nm	22nm	25nm
W_PMOS	44nm	88nm [15]	800nm
W_NMOS	44nm	88nm [15]	800nm

Table 5.1: **Initial sizing and bounds of the design Parameters**

In the circuit sizing and optimization of circuits, the design parameters mainly consist of channel lengths(L) and widths(W) of all devices. Such design parameters are tuned, within specified bound (shown in Table. 5.1) during the optimization phase to fulfill the performance specifications incorporating all PVTA variations. We have also ensured the indigenous working of the final optimized circuit through the entire range of process parameters, aging variations and operating variations. An overview of the optimization process is shown in Fig.(5.2)

We presented detailed optimization results for a 1-bit CMOS mirror Full Adder(FA) circuit consisting of 28 transistors and thus 56 design parameters. The initial sizing of all MOS devices in FA are taken from [42] and the optimized results for high performance applications, i.e., critical path delays are well below the delays of initially sized FA at NOC are reported in Table. 5.2. Also reported for optimization results at NOC. The final robust sizings of all the MOS devices in FA for high performance applications are reported in Table.5.3.

Table.5.4 and Table.5.5 report results for high performance and low power applications of basic cells respectively. Fig.(5.3) shows the optimization of various basic cells for high performance applications. It is to be noted that the algorithms are modified according to the initial analysis performed. This includes determination of variable value bounds, selection of significant design variables to be modified and the range of process and operating parameters to be used to test the robustness of the design.

Performances	Initial		GSO		NCGA		SPEA2		HHO	
	NOC	PVTA	NOC	PVTA	NOC	PVTA	NOC	PVTA	NOC	PVTA
$AvgLeak$ (µW)	1.66	6.22	1.61	6.105	1.63	6.19	1.61	6.02	1.64	6.20
$T_{plh\_a\_C}$ (ps)	8.21	10.96	9.18	13.68	11.87	16.67	11.40	16.15	8.71	13.76
$T_{phl\_a\_C}$ (ps)	10.01	16.10	8.24	13.88	9.46	15.49	8.66	14.31	9.63	13.26
$T_{plh\_b\_C}$ (ps)	7.05	9.69	9.77	14.53	9.05	13.64	10.10	14.83	9.28	15.23
$T_{phl\_b\_C}$ (ps)	9.97	15.36	9.23	14.27	10.29	16.03	9.20	14.49	9.08	13.65
$T_{plh\_c\_C}$ (ps)	8.15	10.45	**10.40**	14.38	10.12	9.92	13.67	6.105	8.62	15.32
$T_{phl\_c\_C}$ (ps)	9.42	14.17	8.58	13.58	10.44	**16.76**	7.37	11.69	8.71	13.29
$T_{plh\_a\_S}$ (ps)	15.45	20.8	9.54	13.71	**12.23**	16.44	**11.49**	15.478	10.37	13.96
$T_{phl\_a\_S}$ (ps)	**17.38**	**27.58**	9.08	**15.60**	10.00	16.67	9.57	16.15	9.38	15.08
$T_{plh\_b\_S}$ (ps)	14.93	21.19	9.65	13.82	9.01	13.04	11.25	15.92	**10.66**	14.78
$T_{phl\_b\_S}$ (ps)	15.14	24.01	9.71	15.48	10.21	16.43	9.97	**16.18**	10.52	**15.296**
$T_{plh\_c\_S}$ (ps)	15.03	20.07	10.33	13.60	11.68	15.67	11.04	14.70	10.44	13.98
$T_{phl\_c\_S}$ (ps)	15.62	23.26	9.00	14.49	10.02	16.34	8.46	13.70	9.18	15.295

Table 5.2: PVTA invariant optimization results for CMOS 28T 1-bit Full Adder (for high-performance circuits)

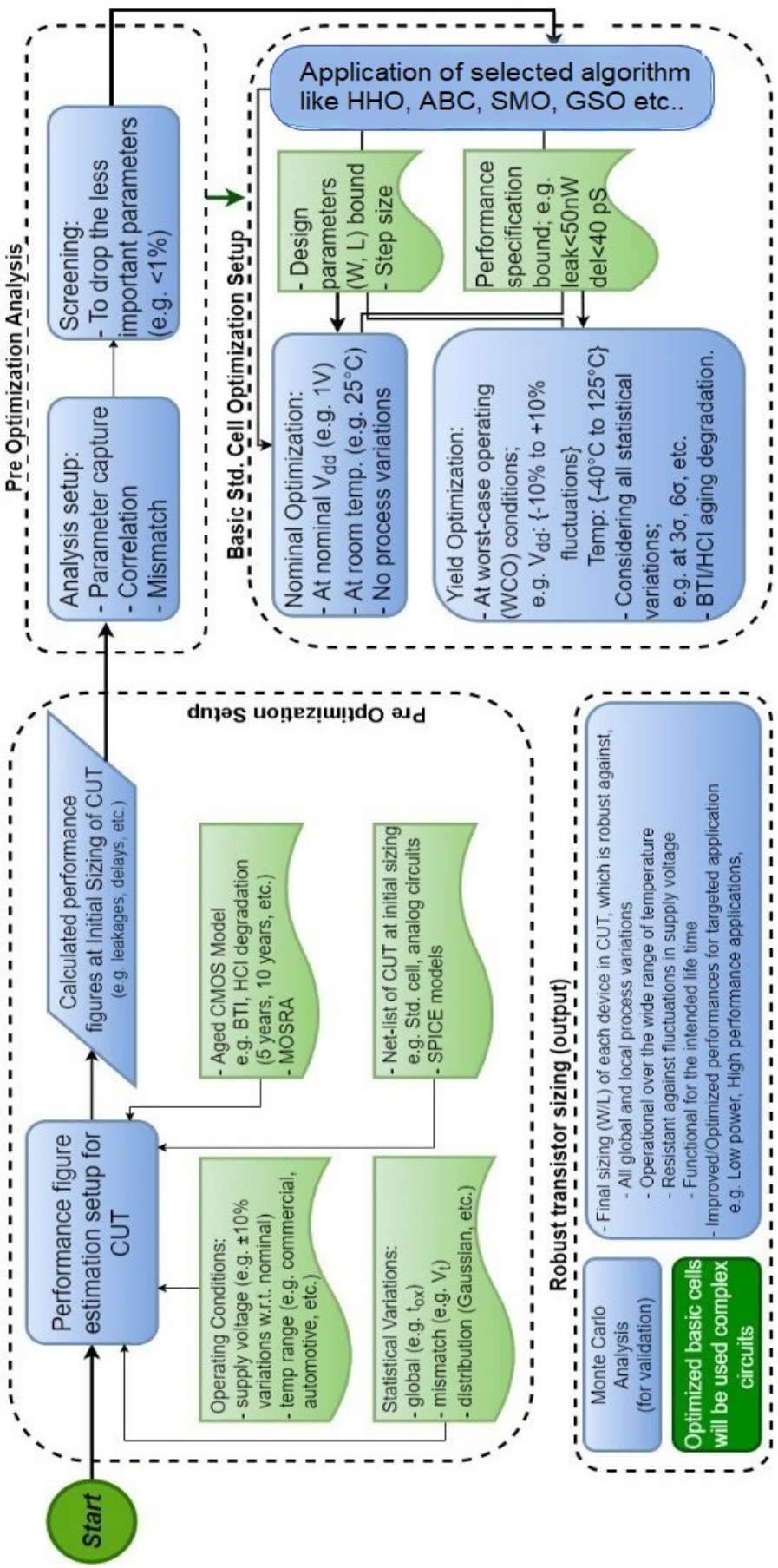


Figure 5.2: Overview of basic cell optimization

Sizings	Initial	GSO	NCGA	SPEA-II	HHO
$p1$	$352/22$	$376/22$	$623/22$	$398/22$	$565/22$
$n1$	$176/22$	$545/22$	$346/22$	$608/22$	$185/22$
$p2$	$352/22$	$259/22$	$343/22$	$497/22$	$313/22$
$n2$	$176/22$	$361/22$	$182/22$	$196/22$	$147/22$
$p3$	$352/22$	$485/22$	$262/22$	$436/22$	$344/22$
$n3$	$176/22$	$152/22$	$152/22$	$150/22$	$155/22$
$p4$	$352/22$	$498/22$	$446/22$	$380/22$	$369/22$
$n4$	$176/22$	$171/22$	$126/22$	$111/22$	$175/22$
$p5$	$352/22$	$308/22$	$229/22$	$352/22$	$341/22$
$n5$	$176/22$	$176/22$	$576/22$	$329/22$	$366/22$
$p6$	$352/22$	$900/23$	$563/22$	$451/22$	$630/24$
$n6$	$176/22$	$671/23$	$551/22$	$265/23$	$730/22$
$p7$	$352/22$	$236/22$	$377/22$	$352/23$	$900/23$
$n7$	$176/22$	$239/22$	$370/22$	$531/22$	$880/23$
$p8$	$352/22$	$333/22$	$235/22$	$799/25$	$227/23$
$n8$	$176/22$	$609/22$	$722/22$	$555/22$	$900/22$
$p9$	$352/22$	$236/22$	$377/22$	$352/23$	$900/23$
$n9$	$176/22$	$239/22$	$370/22$	$531/22$	$880/23$
$p10$	$528/22$	$145/22$	$427/22$	$897/22$	$98/23$
$n10$	$264/22$	$529/22$	$776/22$	$407/22$	$314/23$
$p11$	$528/22$	$44/22$	$44/23$	$65/22$	$112/22$
$n11$	$264/22$	$329/22$	$44/24$	$165/22$	$44/22$
$p12$	$528/22$	$44/23$	$44/22$	$44/22$	$44/23$
$n12$	$264/22$	$22/24$	$44/22$	$44/23$	$44/23$
$p13$	$352/22$	$44/22$	$44/22$	$44/22$	$44/22$
$n13$	$176/22$	$55/23$	$44/22$	$44/22$	$44/22$
$p14$	$352/22$	$265/22$	$198/22$	$379/22$	$253/22$
$n14$	$176/22$	$87/22$	$79/22$	$188/22$	$220/22$

Table 5.3: **Robust Sizing obtained for** 28**T 1-bit Full Adder cell at** $22\mathrm{nm}$ **technology node**

Circuit		Initial	HHO	NCGA	GSO	SPEA-II
AND3	PVTA	13.24	8.62	8.51	8.97	9.53
	NOC	9.24	5.4	5.37	5.63	6.18
NAND3	PVTA	10.81	6.7	6.9	6.7	7.13
	NOC	7.05	4.6	5.1	4.9	5.05
NOR3	PVTA	11.37	6.8	7.4	7.1	7.82
	NOC	7.56	6.98	5.6	5.02	5.21
MUX	PVTA	10.78	7.2	7.5	7.33	8.03
	NOC	7.86	5.5	5.8	5.4	5.8
XOR2	PVTA	8.39	5.8	6.1	6.01	6.7
	NOC	6.11	4.7	4.9	4.6	4.9
FA	PVTA	27.58	15.296	16.76	15.6	16.18
	NOC	17.58	10.66	12.23	10.40	11.49

Table 5.4: **Critical path delay**(ps) **optimization of Basic cells at 22**nm **technology at NOC and PVTA conditions**

Circuit		**Initial**	**HHO**	**NCGA**	**GSO**	**SPEA-II**
AND3	PVT	2.769	1.32	1.54	1.41	1.402
	NOC	0.312	0.144	0.151	0.148	0.167
NAND3	PVT	2.022	1.16	1.32	1.24	1.28
	NOC	0.217	0.11	0.15	0.13	0.129
NOR3	PVT	1.216	0.68	0.73	0.71	0.72
	NOC	0.282	0.15	0.18	0.17	0.14
MUX	PVT	3.1	1.31	1.47	1.33	1.26
	NOC	0.92	0.41	0.58	0.45	0.42
XOR2	PVT	7.487	3.46	3.94	3.81	3.975
	NOC	0.586	0.33	0.42	0.39	0.28
FA	PVT	15.83	5.83	6.28	6.12	5.91
	NOC	1.72	0.59	0.61	0.65	0.72

Table 5.5: **Average Leakage**$(\mu\mathrm{W})$ **optimization of Basic cells at 22**nm **technology for NOC and PVT conditions**

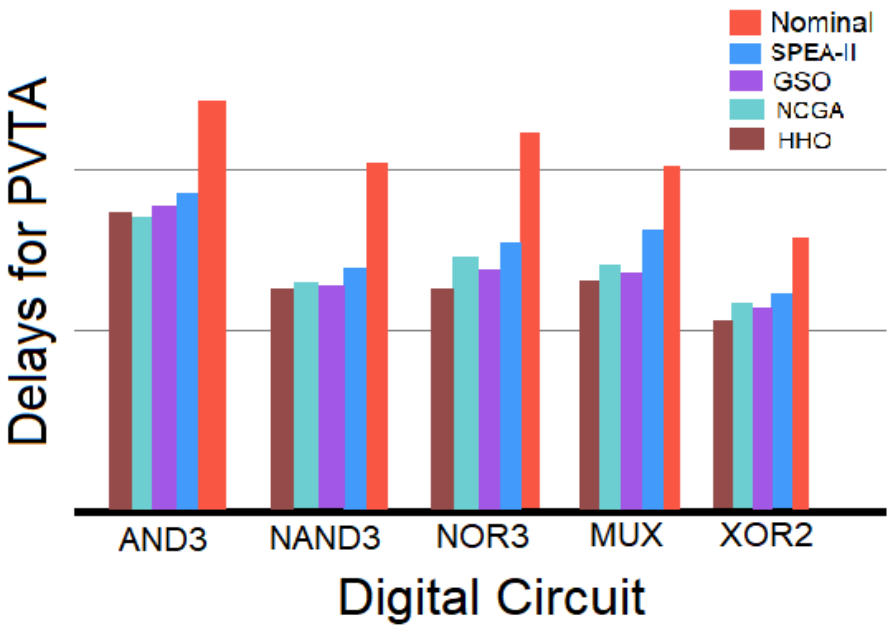


Figure 5.3: Delay optimization with PVTA variations for Basic cells

Chapter 6

Top level framework for Complex circuit optimization

6.1 Introduction

In this chapter, we present an approach to optimize multi-stage complex circuits by replacing the basic cells with preoptimized basic cell sizings using the algorithms mentioned in the previous chapter 5. In contrast to approaches that directly replace all the basic cells, we propose two methods to replace the basic cells in a complex circuit selectively. We extensively evaluated this technique on numerous ISCAS benchmark circuits.

6.2 Replacing the basic cells

We have proposed a method to optimize the complex cells by replacing the basic cells in the huge complex circuit. This reduces the computational time by great extent and always reduces leakage power, but this leads to some problems in optimization affecting delays in particular. Change in sizing alters the current through the circuit and in-turn affects the delay.

For instance, consider a 4 bit multiplier shown in Fig.6.1. Delay at initial sizing is 39.22pS and the current through the node X is 0.66μA in Nominal operating conditions(NOC). On replacing all the cells with optimized sizing obtained by, for example, SPEA algorithm, the current through node X reduces to 0.28μA and hence increasing delay to 64.19pS. Thus, it is clear that complex cells can not be optimized by directly replacing basic cells with optimized sizings as proposed in [29] and [30].

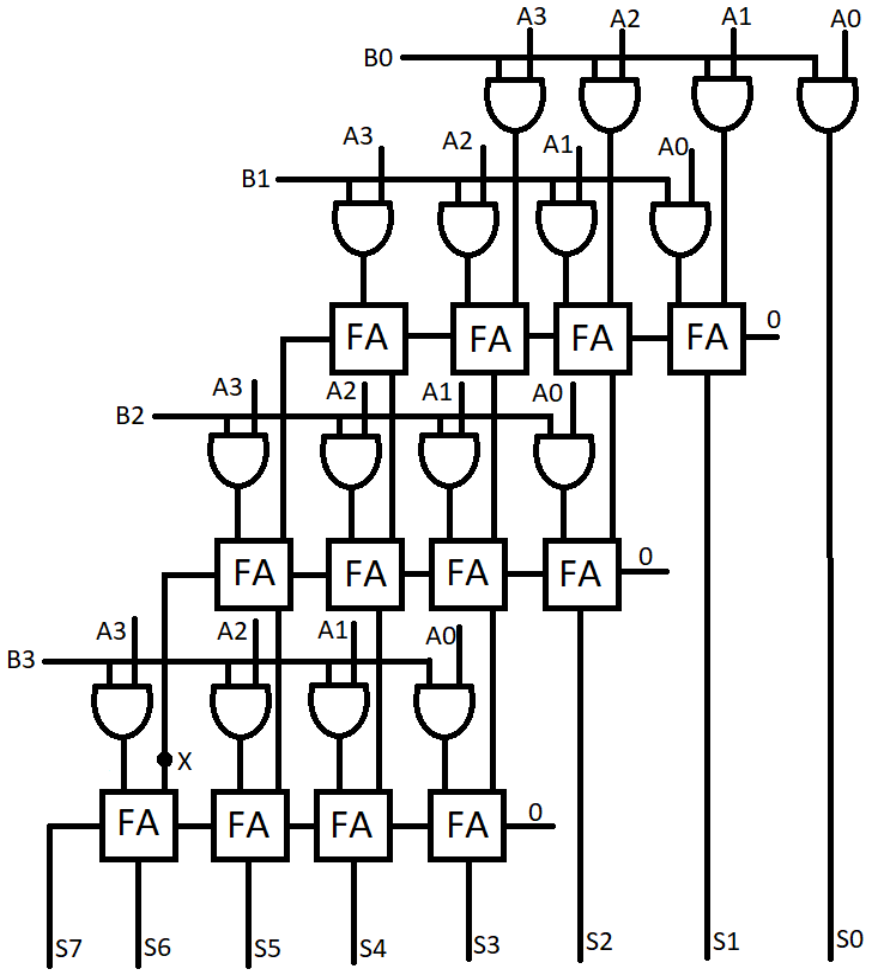


Figure 6.1: 4 bit Multiplier

6.3 Effect of cell resizing on Delay

To address the issue mentioned in the previous section, we need to first understand the effect of cell resizing. The impact of cell resizing on delay is determined by the input-output signal transitions and the sizes of the constituent transistors. The setup in Fig.6.2 is used to investigate the delay variation of a circuit path caused by cell resizing.

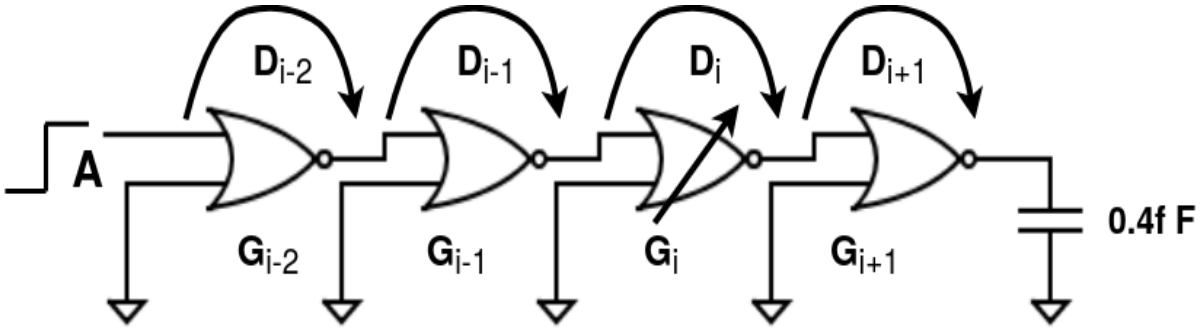
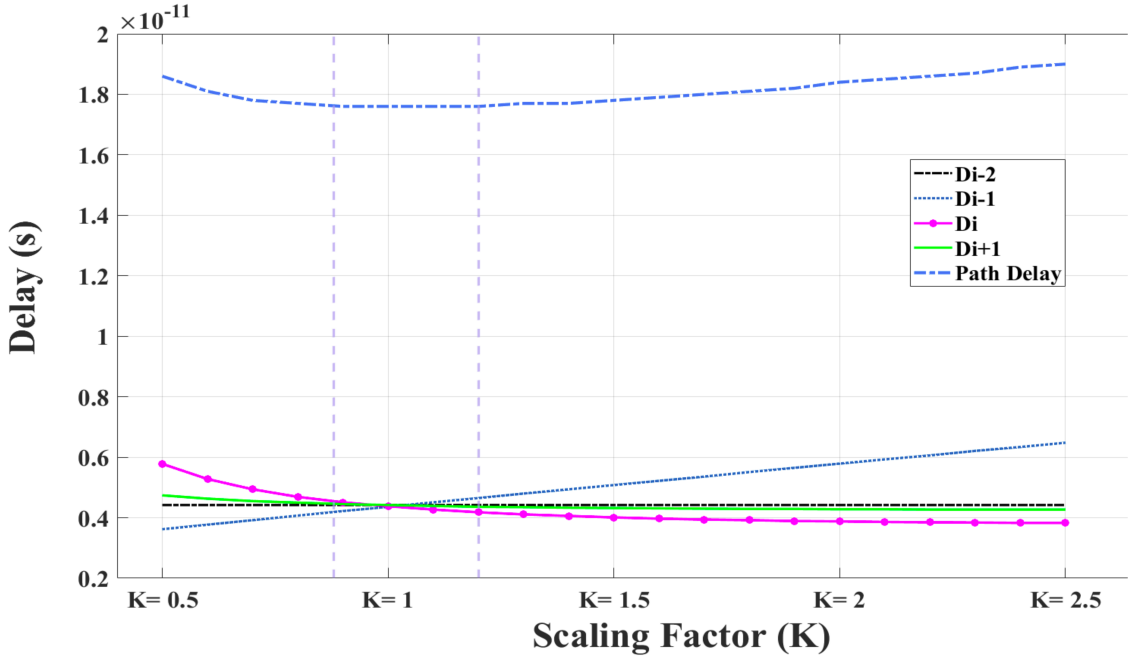


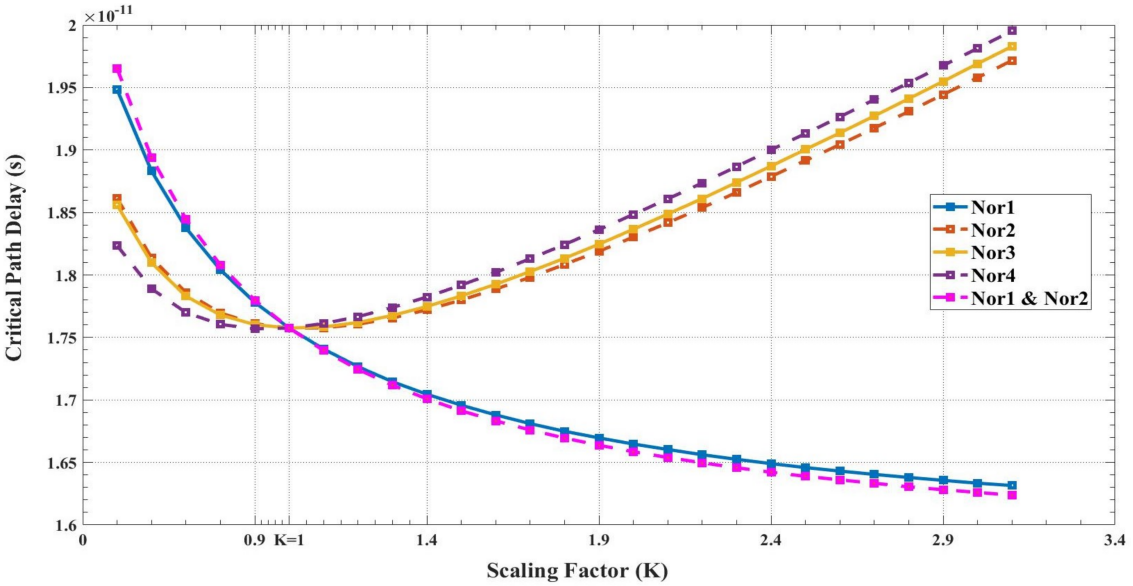
Figure 6.2: Logic path with re-sized cell

A random logic gate (G_i) is chosen, resized, and the delay (D_i) associated with it is recorded. A scaling factor K is used to resize the G_i transistors. The current pumped into the transistors to charge or discharge them is directly proportional to their widths. Faster signal transitions at the cell output node indicate higher current. However, larger transistors increase the input capacitance of the cell (which is also the load capacitance of the previous driver cell). As K increases, the driver gate (G_{i-1}) takes longer to drive G_i than G_i takes to drive G_{i+1}. Fig. 6.3 represents the impact of resizing G_i on all the cell delays (D_{i-2} to D_{i+1}) and the overall path delay.

D_{i-2} has no overall impact since the immediate load of G_{i-2} is unchanged, and thus the output transition speeds for G_{i-2}, remain constant. The input transition time for G_{i+1} decreases due to the sizing up of its driver (D_i). The linear increase in D_{i-1}, on the other hand, flattens the decreasing D_i and D_{i+1} curves. This analysis shows that path delay is minimised for a wide range of K values. Cells exhibited a significant decline in delays within this range for ($K>1$). However, Less current and consequently less leakage are obtained for ($K<1$). Circuits with a greater number of transistors have a greater number of sizing options to choose and optimize from.



(a)



(b)

Figure 6.3: Impacts of (a) Individual and Group of Gate (b) Resizing on Gate and Path Delays

6.3.1 Load analysis based optimization

According to the results of the aforementioned analysis, cell resizing influences the capacitive load provided by the cell, thereby affecting the delay. This necessitates the need for load analysis and optimization. Each *type* of constituent cell (for example, AND3, XOR2, Full Adder, and so on) is identified for a given circuit, and the amount of load they offer their preceding cell(s) in the circuit is estimated in terms of capacitive loads. This method is based on the process proposed by Olivieri and Mastrandrea in [43], in which the approximate input capacitance for a given cell is estimated by approximately matching the time required required to drive that cell to the time required to charge/ discharge some known capacitors and the rise and fall times.

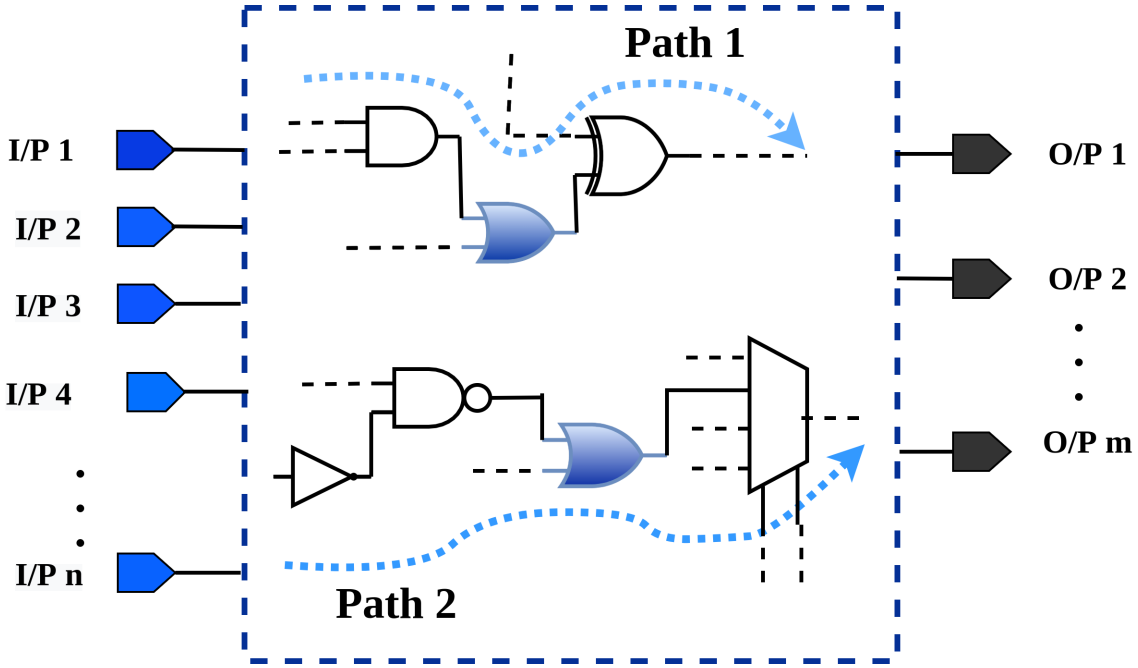
Each constituent cell must be optimised based on the load(s) (input load of the subsequent cell) that it must drive in the circuit. Because the actual input transistor capacitance varies depending on the device's operating region (accumulation, depletion, and inversion), the input load provided by a cell to its predecessor is approximated in terms of the time required to drive its input from high to low and vice versa.

As a result, the cell (for example, NOR2 in Fig. 6.5) is retained as the load to a standard output driver module. Another configuration uses a set of capacitive loads as the driver load. In both setups, the time taken by the driver to drive (or charge/ discharge) the two loads is determined in terms of the propagation delays t*pl-h* and t*ph-l* (D1 and D2 in Fig. 6.5).

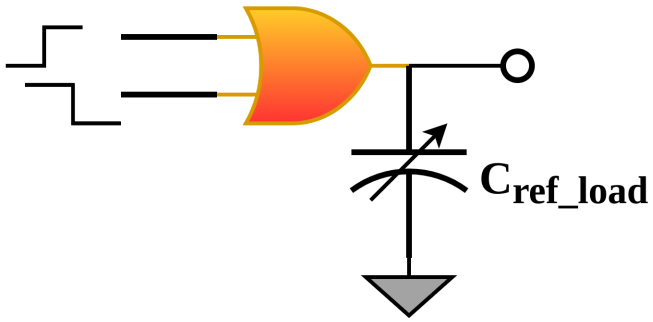
The capacitive load ($Cref$) for which both delays are nearly equal is estimated as the input load offered by the cell to any of the circuit's preceding cell(s). It should be noted that the approximate input capacitance obtained for a cell is unaffected by the driver module used. The approximate capacitance obtained for various cells is shown in Table 6.1. According to the results of the above analysis, we optimized the basic cells for a variety of loads, so that the proposed top-level model can select the optimized sizing appropriately to replace depending on the load.

6.4 Complex circuit optimization

The large complex circuits are optimized by replacing the basic cells with optimized sizing. However, straightforward replacement with optimized sizing can have many issues and



(a)



(b)

Figure 6.4: Load based Optimization

Cell	$\textbf{Delay}_{l\_h}$	$\textbf{Delay}_{h\_l}$	Cap	$\textbf{Delay}_{l\_h}$	$\textbf{Delay}_{l\_h}$
AND2	4.9ps	6.25ps	0.3fF	5ps	6.27ps
AND3	5.2ps	6.5ps	0.37fF	5.3ps	6.5ps
NAND2	4.9ps	6.25ps	0.3fF	5ps	6.27ps
NAND3	5.2ps	6.5ps	0.37fF	5.3ps	6.5ps
NOR2	5.4ps	6.4ps	0.4fF	5.4ps	6.5ps
NOR3	6.2ps	6.9ps	0.59fF	6.2ps	7.1ps
OR2	5.4ps	6.4ps	0.4fF	5.4ps	6.6ps
XOR2	6.6ps	7.4ps	0.7fF	6.6ps	7.5ps
FA	9.3ps	9.2ps	1.46fF	9.3ps	9.6ps
FO-2	5.7ps	6.7ps	0.47fF	5.7	6.8ps
FO-4	7.5ps	7.9ps	0.95fF	7.5ps	8.2ps
FO-6	9.3ps	9.3ps	1.46fF	9.3ps	9.6ps
FO-8	11.2ps	10.5ps	1.95fF	11.2ps	10.9ps

Table 6.1: Approximate Input Capacitance for Basic Cells

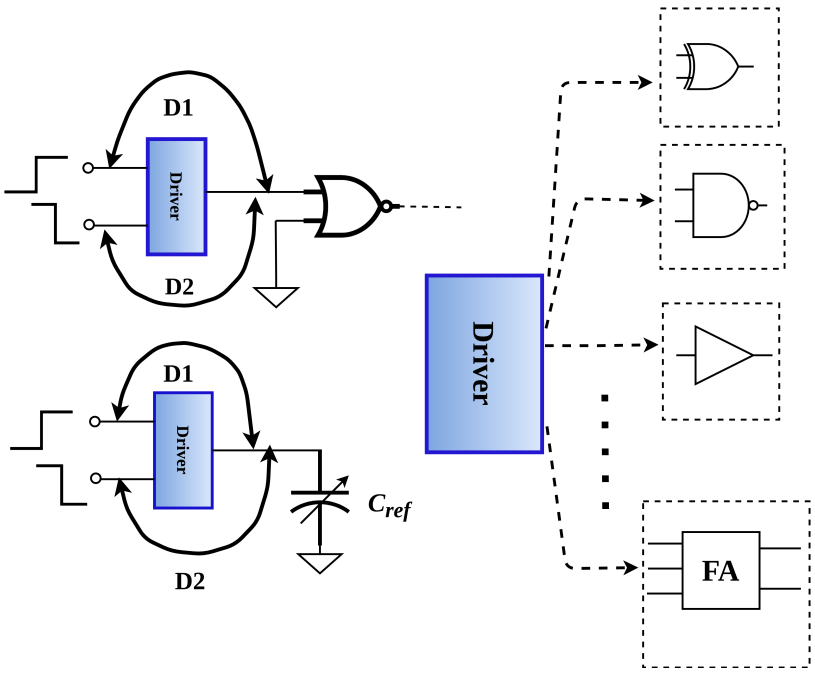


Figure 6.5: Input Load Approximation for NOR2 cell

increase the delay. Whereas replacing the basic cells always reduces the leakage.

As mentioned earlier, resizing transistors can impact the delay and can increase the delay in some instances. In other terms, on replacing the basic cell with optimized sizing, the current through the cell is affected and, in turn, the delay. To overcome these issues, we have proposed two methods. (i) Backward Traversal Replacement and (ii) Partitioning large basic cells.

6.4.1 Backward Traversal Replacement

Once the basic cells are optimized, all the cells in the non-critical paths of the circuit are replaced with their optimized instances. Since the critical path defines the overall speed of operation of the circuit, cells in it are treated differently. We propose a novel approach of *Backward Traversal and Replacement*. According to this, starting from the output end, each cell is replaced with its corresponding optimized instance. After replacement, all the circuit performances are compared with the reference performances recorded earlier. If any of the performances deteriorate as compared to their reference performances, the process is stopped, and the most recent cell replacement is rolled back. Otherwise, the replaced cell remains in the circuit, and the next cell (one cell closer to the input) is taken up for replacement.

For instance, consider the multiplier circuit in Fig.6.1 and SPEA-II optimized basic cell sizings. Full adders which are connected to S7, S6, S5, S4, and S3 are replaced with the optimized Full adder sizings. Now, the optimized delay has been reduced to 22.41pS with the SPEA-II optimized sizing. As the delay has not increased, basic cells connected to these Full adder stack i.e., "And" gates and another stack of Full Adders are also replaced with optimized sizing. Now the new optimized critical path delay becomes 20.22pS. As the delay further decreased, gates connected to the second stack of Full Adders are also optimized. But now the critical path delay has increased to 68.17pS with SPEA-II optimized sizing. As the critical path delay has taken a hit, the replacement of basic cells is stopped. For the optimized multiplier, the current through node X is 0.419μA. This is reduced when compared to that with initial sizing. But this current is capable of driving the optimized Full Adder circuit without increasing the delay. The optimized 4-bit multiplier is shown in Fig.6.6, where highlighted cells are replaced with optimized sizing and the final critical path delay is $20.22pS$. The proposed method also takes care of wire delay in complex circuits which is a major concern in 22nm or lower technology nodes. In other words, gate delay is optimized for basic cells, and the proposed Backward Traversal replacement optimization method optimizes critical path delay by handling the wire delay.

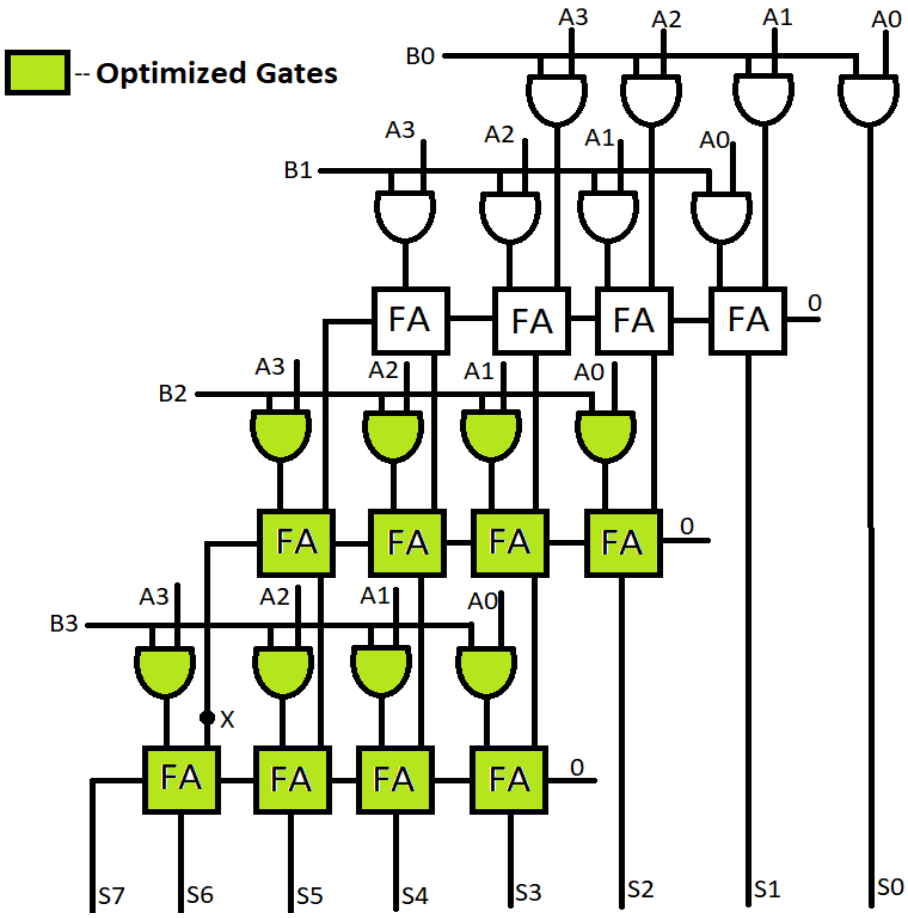


Figure 6.6: 4 bit Multiplier

6.4.2 Partitioning Large Basic cells

Depending on their architecture, large basic cells can be optimized in parts. The effects of each transistor on performance are examined using the paths from primary inputs (PIs) to primary outputs (POs). As a result, some transistors can be optimized for high performance while others can be optimized for low power applications. Such an approach makes the technique architecture dependent, but it has its advantages.

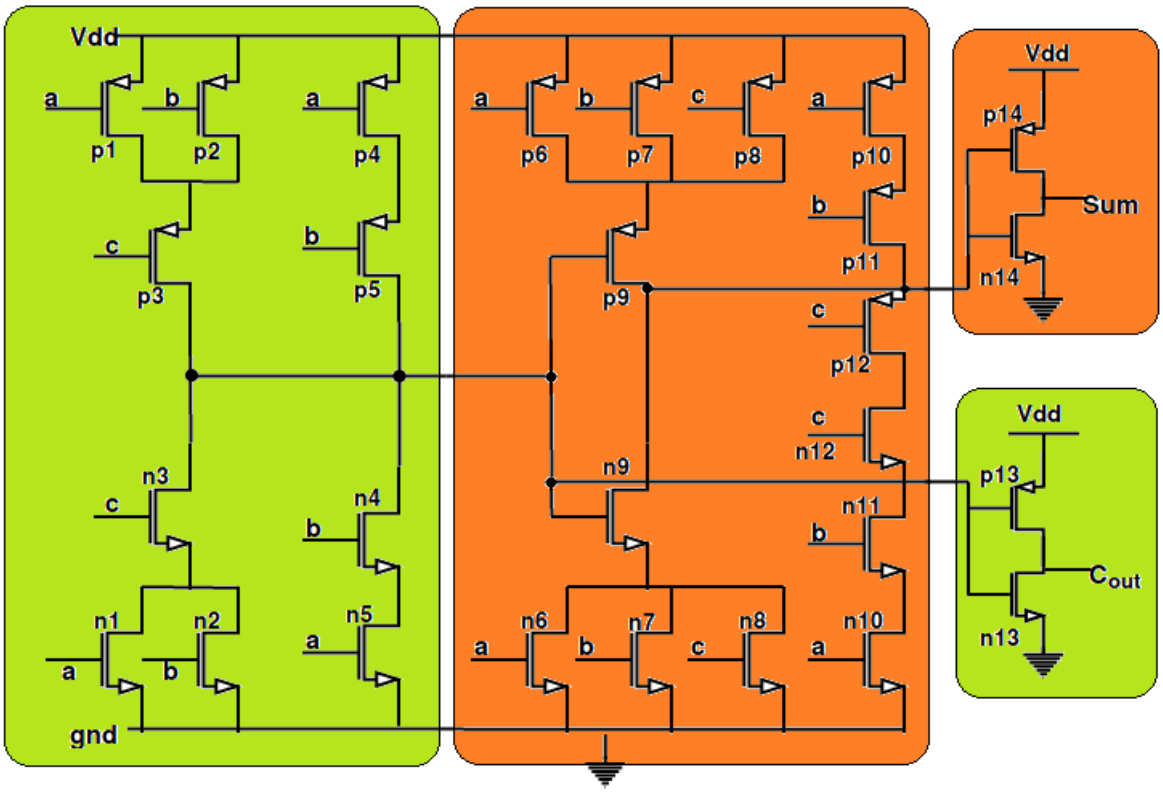


Figure 6.7: Partitioned CMOS 28T Full Adder Cell

The 28T Full adder circuit in Fig.4.1, is a classic case study. The circuit's partitioning is determined by two factors: the paths in which the transistors are located and the number of transistors. The figure depicts three primary inputs (PI) – A, B, and C – and two primary outputs (PO) – Sum and Carry. The circuit, according to the architecture, can be broadly partitioned into sets of transistor blocks, as illustrated in Fig.6.7, depending on the path they impact- the Sum block (p6-p12 and n6-n12) and the Carry block (p1-p5 and n1-n5). The PI-to-Carry (green in Fig.(6.7)) path does seem to have fewer transistors than the Sum block (orange in Fig.(6.7)). Furthermore, the last Carry stack transistors are not in charge of controlling the Sum Stack transistors.

To optimize both the conflicting objective functions, the Carry block is optimized for high performance and the Sum block for low power. The Carry block has fewer transistors, and six of the Sum transistors ($p_{10}, p_{11}, p_{12}, n_{10}, n_{11}, n_{12}$ in Fig.(6.7)) have the highest nominal sizing. The carry block being optimized for high performance, sum transistors have more windows to shrink and have a more significant impact on leakage reduction due to more number of transistors. This decreases the extent of leakage reductions as the degree of freedom is reduced here than the undivided circuit. However, the final design has better delay performances, and the total optimization time reduces to one-third of its initial value. This partitioning and trade-off analysis between the path interdependence and the number of transistors in each block changes with the performance specifications. The target specifications then determine whether a segment of the larger cell is to be optimized for low power or high speed.

6.4.3 Top level model for complex circuits

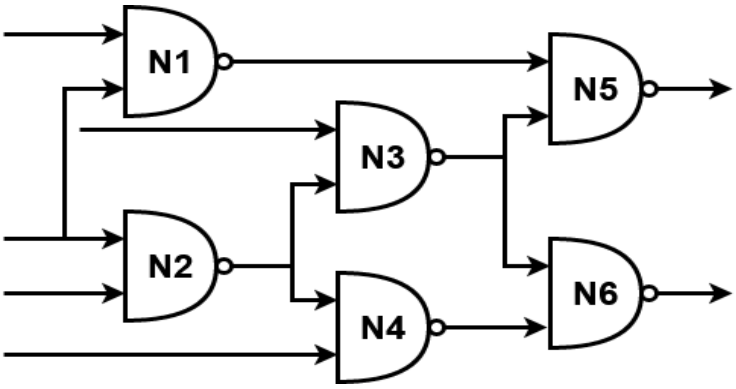
The proposed methods solve the delay concerns on replacing basic cells with optimized sizing. However, it becomes a tedious job to selectively replace the basic cells manually in circuits with a huge number of basic cells. To address this, we have proposed a top-level model that analyses and optimizes the circuit with the above proposed Backward traversal replacement and Partitioning large basic cell methods.

The optimized sizing of basic cells and the netlist of the complex circuit are fed as input to the top-level model. The model then analyses the complex circuit and splits it into individual basic cells. After the approximation of the input loads offered by each basic cell, the entire circuit is expressed as a directed graph as shown in Fig.(6.8) with one-to-one mappings between the input/ output nodes of each constituent cell. The entire circuit is traversed, taking one cell type at a time, finding all its occurrences and the various loads they are connected to across the circuit, based on the type of successive cells connected to their outputs.

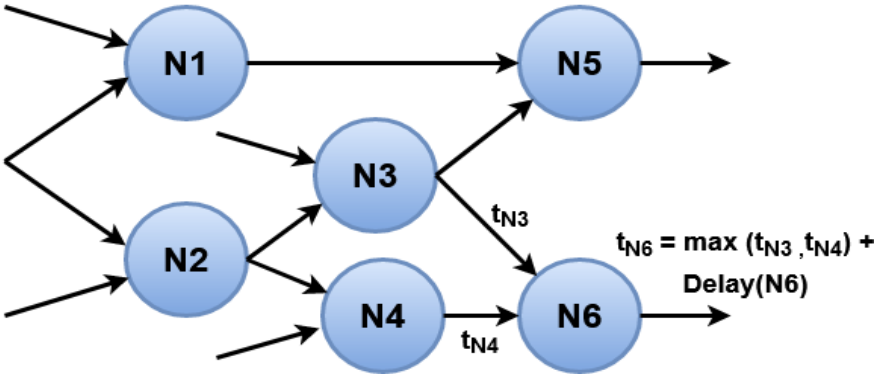
In the next stage, the tool traverses the entire graph from its primary inputs to primary outputs and attaches a weight (t_d) to all the nodes. From Fig.(6.8), the maximum delay from primary inputs to the node (N6) is represented as the sum of the maximum of the delay of inputs and propagation delay of the node as per eq. (6.1)

$$t_{N6} = max(t_{N3} + t_{N4}) + Delay(N6) \tag{6.1}$$

Algorithms vary from each other in terms of search space exploration, exploitation capabilities and speed of convergence. Based on these metrics, algorithm performances vary with the transistor count in standards cells. The model considers all these criteria and chooses the best-suited algorithm for each basic cell in the complex circuit depending on the low power or



(a) C17



(b) Directed Acyclic graph for C17

Figure 6.8: Representation of C17 as Directed Acyclic graph

high-performance applications. The proposed top-level model checks for any large basic cells that can be partitioned and replaces them by the aforementioned partitioning large basic cells method. It then proceeds with the backward traversal replacement method and finally returns the optimized complex circuit. An overview of this approach is mentioned in Fig. 6.9.

Hence this work follows an automated approach for selecting appropriate algorithms based on the cell to be optimized. The results for complex circuit optimization using the proposed model are reported in Table 6.2 and Table 6.3 for high performance and low power applications, respectively. We have achieved an average of 21% and 17% reduction in critical path delay at PVTA and NOC conditions respectively for the complex circuits mentioned in the Table6.2. Similarly it is to be noted from Table 6.3 that an average reduction of 47% and 44% reduction in average leakage is achieved at PVT and NOC conditions respectively for the mentioned complex circuits. A summary of comparison with prior work is presented in Table 6.4, where HP and LP correspond to high performance and low power optimization respectively.

Circuit		Initial	Optimized
C17	PVTA	14.7	13.9
	NOC	10.8	10.3
Parity checker	PVTA	37.35	29.21
	NOC	28.40	22.99
Multiplier	PVTA	49.59	26.1
	NOC	39.22	20.22
1b ALU	PVTA	121.34	118.2
	NOC	27.3	23.8
8b RCA	PVTA	188.16	138.1
	NOC	146.52	115.8
8b CSA	PVTA	193.2	147.2
	NOC	150.55	119.8
8b CskipA	PVTA	437.01	313.7
	NOC	327.18	204.9
C1908	PVTA	134.16	106.1
	NOC	88.79	61.5
C499	PVTA	126.8	90.84
	NOC	82.3	60.21
C1355	PVTA	528.3	437.4
	NOC	385.35	309.1

Table 6.2: **Critical path delay**(ps) **optimization of complex circuits using the developed model at 22**nm **technology at NOC and PVTA conditions**

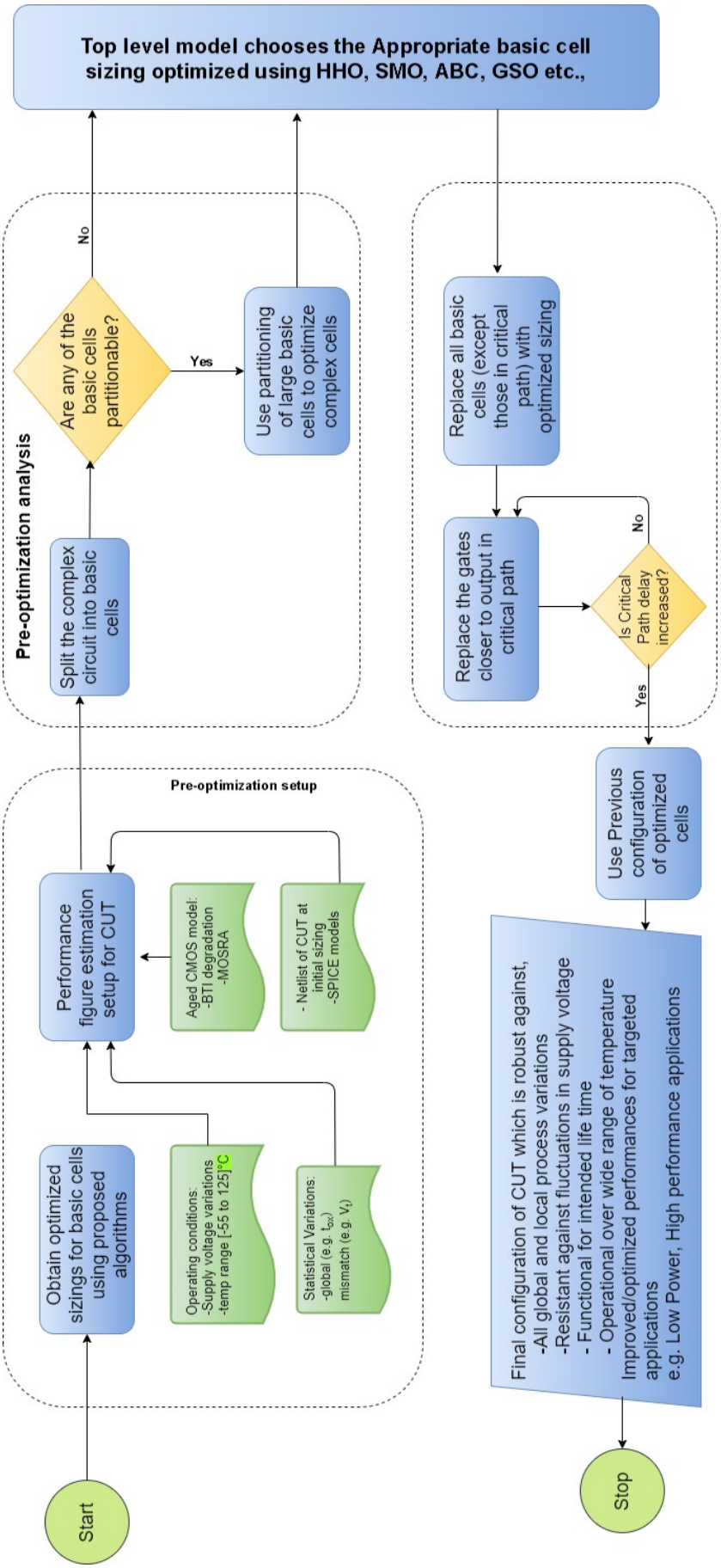


Figure 6.9: Overview of complex circuit optimization

Circuit	**Condition**	**Leakage (μW)**	
		Initial	*Optimized*
C17	NOC	1.044	0.882
	PVT	10.73	8.86
Parity checker	NOC	1.55	0.56
	PVT	4.527	1.97
Multiplier	NOC	8.09	4.20
	PVT	50.6	25.80
C432	NOC	37.7	19.17
	PVT	76.5	39.62
C880	NOC	966	705.18
	PVT	7922	5465.6
1b ALU	NOC	3.79	2.02
	PVT	46.03	32.8
8b RCA	NOC	53.3	20.33
	PVT	467.9	168.12
8b CSA	NOC	133.7	41.44
	PVT	440.6	145.26
8b CskipA	NOC	116.6	47.56
	PVT	279.9	117.9
8b 74182	NOC	9.06	6.54
	PVT	79.4	57
8b 74181	NOC	67.7	42.4
	PVT	278.6	196.6
8b 74L85	NOC	8.86	6.45
	PVT	75.4	57.3
8b 74283	NOC	178	135
	PVT	514.96	387
C1908	NOC	414	312.68
	PVT	2950.7	2128.5
C499	NOC	119.15	84.52
	PVT	1042.7	751
C1355	NOC	118.5	92.7
	PVT	980.2	773.8
C6288	NOC	499.7	275
	PVT	3285.3	1982.4
C2670	NOC	174.7	147.5
	PVT	1524	1308.9

Table 6.3: **Leakage Optimization of Complex Cells for NOC and PVT conditions**

	SA	ABC	PSO	Mem	GSO	NCGA	SPEA-II	HHO
Process variations	-	✓	✓	✓	✓	✓	✓	✓
Operating variations	✓	✓	✓	✓	✓	✓	✓	✓
Aging variations	-	-	-	-	✓	✓	✓	✓
LP optimization	✓	✓	✓	✓	✓	✓	✓	✓
HP optimization	-	-	-	-	✓	✓	✓	✓
Technology (nm)	45	45	45	22	22	22	22	22
% optimization for HP	-	-	-	-	43	39	41	47
% optimization for LP	39	45	39	52	50	44	45	52
References	[29]	[30]	[30]	[44]	-	-	-	-

Table 6.4: **Comparative study of quantitative improvements in proposed work to existing works**

Chapter 7

Alternative Approach: Case: polar decoders

7.1 Introduction

Transistor Sizing using Algorithm methodology has been proposed in the previous chapter. This chapter introduces an alternative methodology using the Iterative decomposition technique for the case of the Polar decoder. Polar codes have a very good error-correcting capacity compared to turbo and LDPC codes of similar length. They are the first to attain the channel capacity. The aim of the proposed work is to design an Area and Power optimal 2b SC polar decoder exhibiting reduced area and power without degrading the latency by iterative decomposition technique. The targeted performances in decoder architecture are achieved by the novel reformulation of F-node, G-node, and P-nodes in a polar decoder.

7.2 Polar decoders

Polar codes are the only capacity-achieving codes that attain the Shannon limit in error correction and provide the first deterministic construction of capacity-achieving codes for Binary Memoryless channels and for infinite code length. They can outperform LDPC and Turbo codes with similar code length and code rate in terms of error-correcting performance. However, there are a few concerns regarding polar codes such as high latency, high power dissipation, and of course taking high area in any Integrated Circuit (IC) design is crucial, even when the technology nodes (device dimensions) are scaled down providing room to fabricate the higher number of transistors in the same area. Throughput also decreases as the latency and power increase.

Construction of Polar codes in [45] [46] follows the principle of channel polarization. The reliability of the decoded bit is polarized according to its position in the source data. The transmitted data is divided into good positions and bad positions. The Polar Encoder for N = 8 is shown in the Fig.7.1

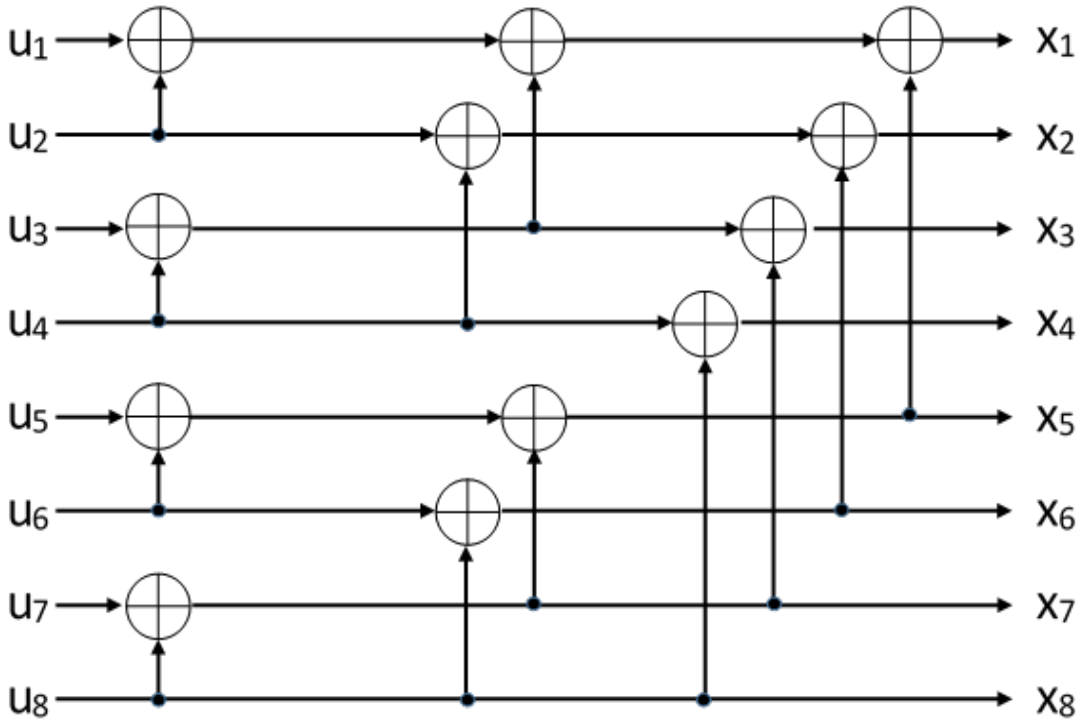


Figure 7.1: Polar Encoder for N=8

Generally, Polar codes have decoders that follow either the Successive Cancellation (SC) method, which is serial in nature, or the Belief Propagation (BP) method which is parallel in nature as cited in [47]. SC decoders exhibit capacity-achieving performance due to efficient utilization of the property of polar encoding but have a large block size. Due to their serial nature, SC decoders have high latency and hence low throughput. BP decoders, being parallel in nature solve the problem of latency but at the cost of hardware and more importantly, the error-correcting performance of the BP decoder is also not as good as SC-based decoders. Therefore, in this paper, we have reported the application of the iterative decomposition technique on SC decoders only.

The effective functioning of any Polar design mostly relies on performance parameters such as error-correcting efficiency, latency, power (dynamic and static power i.e. leakage), and of course the area has always been a crucial parameter in any design. Latency is the time interval

between the cause and the effect as the response of the system. In general, latency can occur due to delay in storing or processing or transmission or processing. In our case, the latency is due to a delay in processing the input to obtain the output. For instance, in the 2b SC decoder architecture, the time interval between the input and the output response is 10 clock cycles. Hence, the latency becomes 10 clock cycles for the respective design. This can be generalized as 1.5n-2 clock cycles for 2b SC decoder architecture where n is the code length.

7.3 Performance Parameters in Polar decoders

Power dissipation can be grouped broadly into two different components; dynamic and static (i.e. leakage) power dissipation. Dynamic power is the consequence of the charging and discharging of the load capacitances during signal switching and from short circuit current when both the pull-up and pull-down transistors are simultaneously on. Static power i.e. leakage power occurs even when there is no signal transition [48]. Leakage currents depend in a complex manner on the device structure properties such as channel dimensions, doping profile, gate oxide thickness, etc., as they are due to different physical phenomena such as gate oxide tunneling, sub-threshold conduction, and reverse bias junction conduction. As per International Technology Roadmap for Semiconductors (ITRS) for the trend of power dissipation with respect to technological progress, static power dissipation in bulk CMOS exceeds dynamic power dissipation [49], [50]. Due to the high impact of leakage power on the total power budget, instead of reporting the total power only, we have estimated the dynamic and static (leakage) power separately.

7.4 2b SC Polar decoder Architecture

The 2b SC architecture [51] [52] as depicted in Fig.7.2 can be assumed as the butterfly architecture, and it consists of F-node, G-node, and P-node as shown Fig.7.3, Fig.7.4 and Fig.7.5 respectively. The F-node has one bit 2-input XOR gate and a 7-bit comparator. The G-node has signed to 2's complement converter (S2C), adder, subtractor, and 2's complement to signed bit converter (C2S) and a MUX with the select line as one of the inputs to the G-node.

The P node has two 1-bit inputs other than two 8-bit inputs and these two 1-bit inputs determine the frozen conditions of the outputs. The other two 8-bit inputs are the LLR values obtained from F-node or G-node. This has 2- input AND gate, XOR gate, OR gate, comparator, and Inverters.

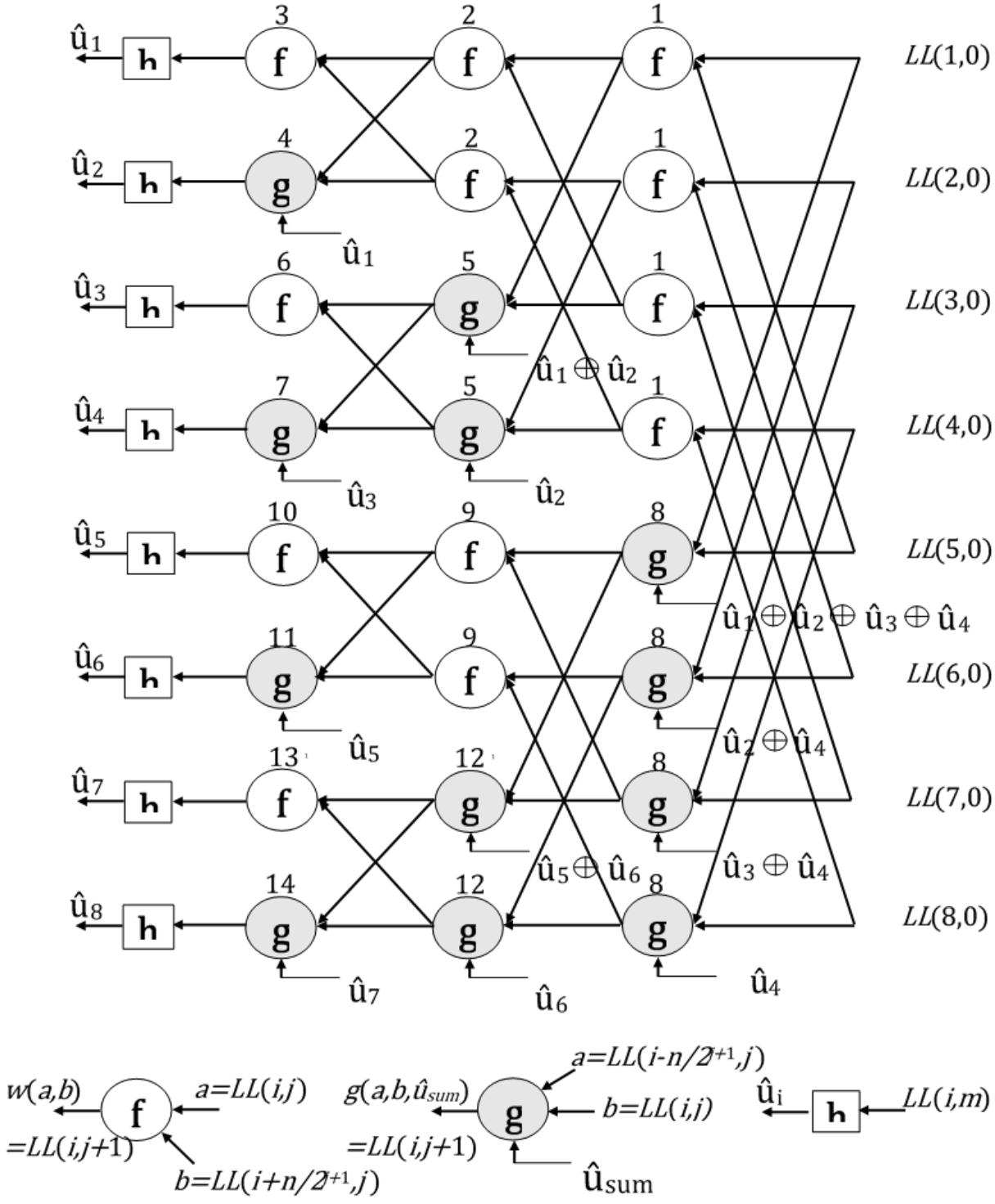


Figure 7.2: The architecture of 2b SC decoder for N=8

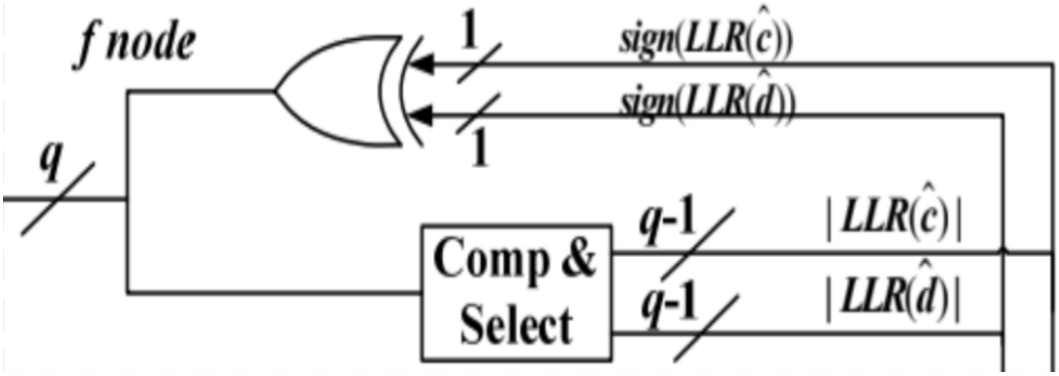


Figure 7.3: F-Node of a 2b SC Polar decoder

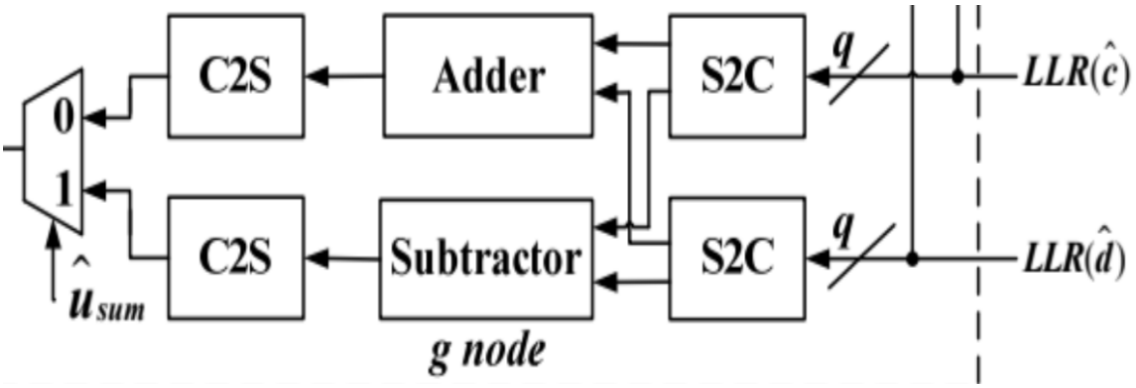


Figure 7.4: G-Node of a 2b SC Polar decoder

The 2b SC decoder architecture as shown in Fig.7.2 has 8 F-nodes, 8 G-nodes and 4 P-nodes. However, approximately half of the nodes remain idle for most of the time and therefore, can be optimized such that the maximum number of nodes are utilized i.e. not idle.

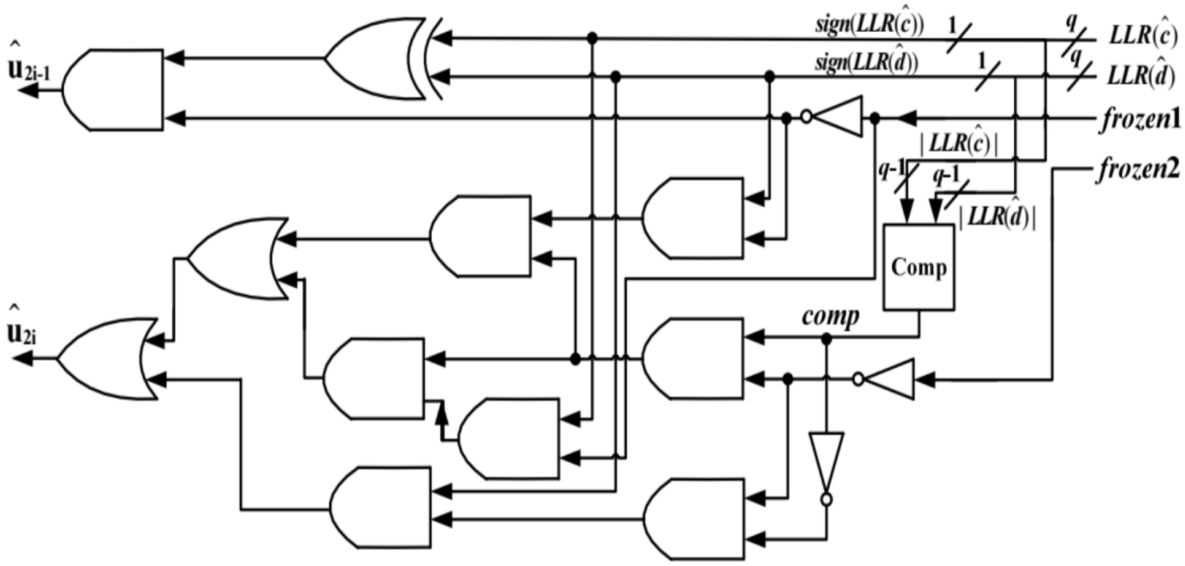


Figure 7.5: P-Node of a 2b SC Polar decoder

7.5 Background and the latencies of 2b SC architectures

In [45], the construction of polar codes was explained and the first SC decoding algorithm was explained. SC decoder has likelihood ratio (LR) based architecture as proposed in [53]. Hence it needs to perform division and multiplication tasks that increase the latency of the SC algorithm to 2n-2 clock cycles, and the complexity is O(nlog2n) for a code of length n. Later both the hardware cost and the latency are improved in the 2b-SC decoder proposed in [51]. The 2b-SC architecture uses LLRs, and hence it avoids multiplications and divisions, and this reduces the latency and hardware cost. The 2b-SC architecture is a reformulated structure of SC architecture, which computes 2 bits at the same time, unlike the SC architecture. The latency of the 2b-SC decoder is 1.5n-2 clock cycles, and the complexity is the order of n i.e., 'O(n).' Based on the 2b- SC architecture, two other architectures named overlapped 2b-SC and pre-computation 2b-SC architectures are developed.

In [51], the latency of the overlapped 2b-SC architecture is n-1 clock cycles, and for pre-computation 2b-SC architecture, the latency is 0.75n-1 clock cycles, which is the least latency. But, the hardware cost of the Pre-computation 2b-SC architecture increased drastically.

7.6 Proposed novel 2b SC Polar decoder architecture

The proposed work aims at reducing the power and area of SC and 2b SC architectures, keeping the functionality and latency of the targeted architectures unchanged.

7.6.1 Iterative Decomposition

The idea of Iterative decomposition is nothing but Resource Sharing through step-by-step execution [54]. If a function f has d similar parts, then such d parts in the function f can be replaced by a single part reducing the area, however, at the cost of latency. For instance, if a function f has three similar parts, *f1*, *f2* and *f3* as shown in Fig.7.6, the computation can be broken down and each part can be executed with the same hardware one after the other by adding an external control section as shown in Fig.7.8.

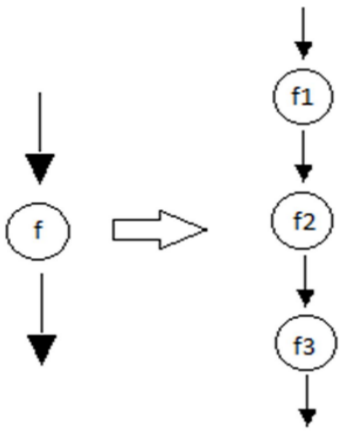


Figure 7.6: Iterative decomposition

7.6.2 Proposed 2b SC Polar decoder design

As we discussed in previous sections, latency has been reduced with the use of 2b-SC-overlapped scheduling and 2b-SC pre-computation architectures. In the proposed power-area

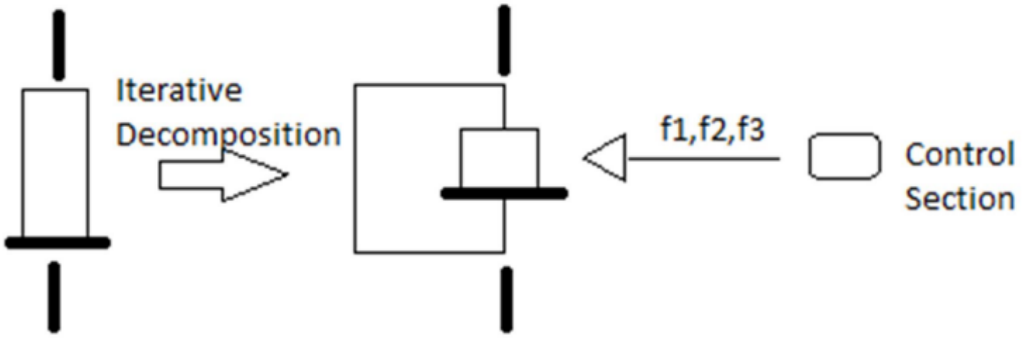


Figure 7.7: Hardware composition for d=3

efficient 2b SC decoder, the power, and area of the hardware have been reduced with the latency unaffected keeping in mind that the hardware of the SC decoders is a major concern, following the decoding procedure of 2b-SC decoder in Fig.7.2. It can be split into 10 cycles, and the outputs obtained in the previous cycles are used in the next cycle. However, we can observe in the decoding process, at any instance, the maximum number of F-nodes, G-nodes, and P-node utilized are 4, 4, and 1 respectively. Therefore, a thoughtful inspection can allow us to reduce the hardware, of course without degrading the functionality and latency. As a result, we added a control circuitry with less hardware with four F nodes, four G-nodes, and one P- node and the area can be reduced without a change in the latency of 1.5n-2 clock cycles. In the control section, Multiplexers are used to switch the inputs of the F-node, G-node, and P-nodes. The select line of the multiplexer varies according to the cycle. In the first cycle, four F- nodes are used simultaneously, and the inputs LLRs are selected through the mux. In the second cycle, two F-nodes are utilized with the F-node outputs in the previous cycle being the inputs, and in the third cycle, one P-node is used with the previous F-node outputs as inputs. In the fourth cycle, two G-nodes are used along with the outputs obtained in the third cycle and first cycles. In the fifth cycle, one P-node is used with the outputs of the G-node as inputs chosen from the corresponding multiplexer. In the sixth and seventh cycles, four G-nodes with LLRs as inputs and two F-nodes with the previous outputs of G-node as inputs are used respectively. During the eighth cycle, one P-node with the F-node outputs as inputs chosen from the corresponding multiplexer is used. In the ninth cycle, two G-nodes are used with the outputs obtained in the sixth cycle as inputs, and finally, in the tenth cycle, one P-node is used with the G-node outputs. The proposed architecture also needs some memory to store the outputs obtained in the first and sixth cycles to be used in the fourth and ninth cycles respectively.

With the optimized architecture, the hardware cost can be reduced and proportionally the power dissipation. Now, one can think of further reducing the hardware by reformulating two F-nodes, two G-nodes, and one P-node architecture or any other architecture less than the proposed four F-nodes and four G-nodes. However, such further reduced nodes based architectures will severely degrade the latency. Therefore, the proposed architecture with four F-nodes and four G-nodes and one P-node is the optimal architecture where latency is not affected. The architecture of proposed 2b-SC decoder is shown in the Fig.7.8

The mf nodes, mg nodes and mp nodes are shown in the Fig.7.8 are modified F, G, and P nodes respectively. The modified F nodes i.e., mf nodes have a 4:1 mux attached to the typical F node as shown in Fig.7.9(a). Similarly, the mg and mp nodes are also shown in Fig. 7.9(b) and Fig. 7.9(c) respectively. The mp node has a 2:1 mux in its architecture. A few nodes like mf3, mf4, mg3, and mg4 have only three inputs. In such cases, the fourth input is Considered as 8-bit 0. It doesn't make any difference as the above proposed architecture never goes to such a state requiring the fourth input.

7.7 Results

Both the above mentioned architectures, 2b SC and proposed 2b SC architectures are implemented in Verilog using Vivado using RTL coding taking N=8. Following the bottom up approach, each basic block is individually built, and later all the node blocks (F, G, and P-nodes) are built using already designed basic blocks. Thereafter, both the polar decoder architectures (existing and proposed) were designed (using node blocks), simulated, and verified for the correct functioning through the identical test bench. Finally, both architectures are synthesized to obtain the targeted performance figures of area (in terms of look-up tables(LUTs)) at the logic level, dynamic power, and latency, used to compare and validate the claimed improvement in the proposed architecture.

It is clearly evident from Table 7.1, with the proposed technique; area (in terms of LUTs) is reduced up to 46.93% for 2b SC decoder. Dynamic power reduces up to 44.55% for 2b SC decoder, without latency taking a hit.

To further validate the capability of the proposed technique, we have implemented the proposed iterative decomposition based polar decoder along with existing basic 2b-SC decoder architecture at transistor level in order to clearly demonstrate the power (especially leakage power) and area (number of transistors) improvement without degrading the latency. The area

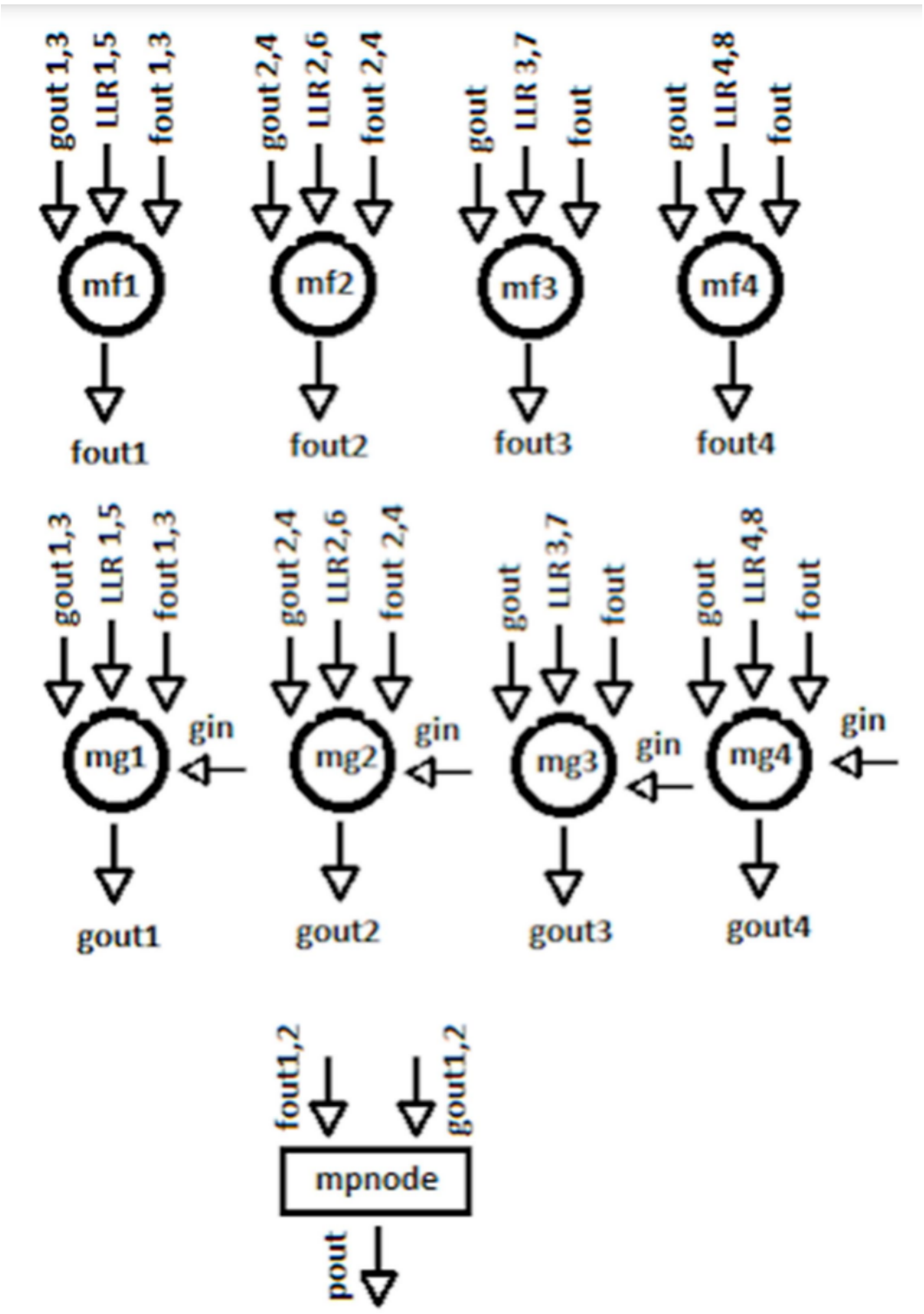
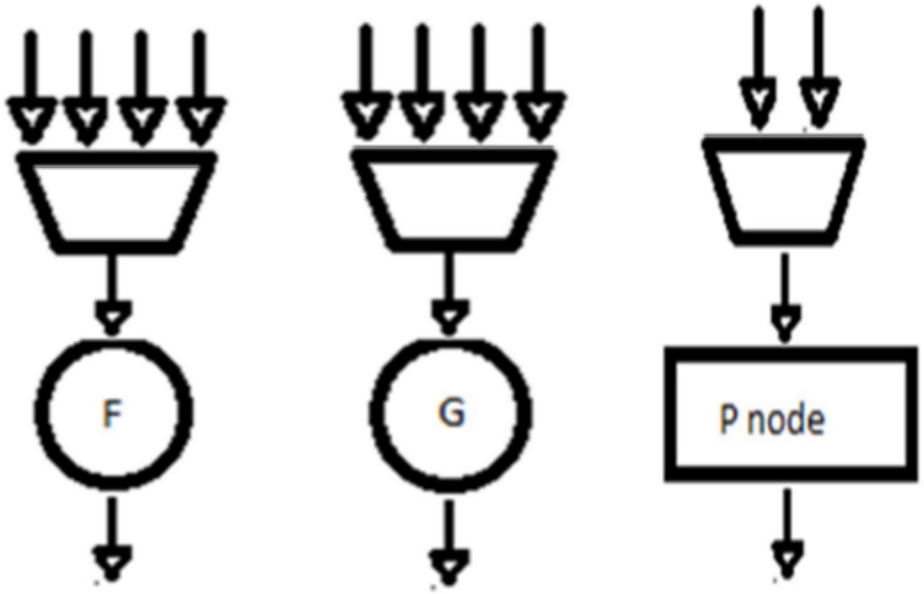


Figure 7.8: Hardware composition for d=3



(a) (b) (c)

Figure 7.9: Modified Nodes with multiplexers (a). Modified F node, (b) Modified G node, (c) Modified P node

Parameter	2b SC decoder	Proposed 2b SC decoder
Area	488	259
Dynamic Power	5.881W	3.261W
Latency	1.5n-2	1.5n-1

Table 7.1: **Comparative study of quantitative improvements in proposed work to existing works**

optimization is validated through the synopsis DC synthesis tool.

All the SPICE level simulations/characterizations of both the polar decoders are performed in HSPICE using Berkeley Short-channel IGFET Model (BSIM4) [55] with 22nm bulk CMOS Metal-Gate/High-K dielectric model parameters based on predictive technology model (PTM) [56] at the supply voltage of 1V and 0.9V.

Both the decoder architectures have eight 8-bit input signals and additionally 2 frozen bits (in 2b SC decoder). It is difficult to report the leakage power dissipation for all input patterns (approx. 2^{64} combinations). Therefore, the reported leakage data is for combination holding maximum leakage. It is again obvious from reported values in Table 7.2; leakage power is

drastically reduced up to 92.28% at the supply voltage of 1V, and 89.07% at 0.9V for the 2b SC decoder. Transistor count is reduced up to 51.5%, while the associated device area (in terms of) reduces up to 39.45% for 2b SC decoder with the proposed technique. Static Timing Analysis is also performed on these two designs in the synopsis tool to check any setup and hold violations.

Parameter	**2b SC decoder**	**Proposed 2b SC decoder**
Leakage Power (at 1V)	401.62W	30.99W
Leakage Power (at 0.9V)	244.52W5.881W	26.71W
Area ($0.0001\,\mathrm{m}^2$)	1.5n-2	1.5n-1
Number of Transistors	4680	2266

Table 7.2: **Comparative study of quantitative improvements in proposed work to existing works**

7.8 Conclusion

A novel reformulation of the F-nodes, G- nodes and P-nodes for the 2b SC decoding architectures is proposed. Based on this reformulation, the proposed architectures can outperform the existing 2b SC architectures in terms of Area and Performance with the same latency. Reported data clearly validates the proposed technique.

Chapter 8

Conclusion and Future Work

Leakage power has been a critical problem in the latest CMOS designs and when the process variations come into picture, this problem has become a Herculean task. The process, aging and operating variations like supply voltage and temperature fluctuations degrade the propagation delay affecting the speed and reliability. The objective of this work is to address these issues and optimize performance parameters of the CMOS digital circuit for indigenous working and invariant to PVTA variations.

This work presents an algorithm framework to optimize the performance parameters for low-power and high performance applications for most commonly used basic cells. Various optimization algorithms have been proposed in this thesis for the optimization purpose like Pareto Harris Hawk optimization algorithm, Glowworm Swarm optimization algorithm, Neighbourhood Cultivation Genetic Algorithm and Strength Pareto evolutionary Algorithm-II. As there is a trade off between leakage power and propagational delay, Leakage power is optimized with critical path delay in bound for low power applications and vice-versa for high performance applications i.e., critical path delay is optimized with leakage power in bound. We have achieved upto 45% optimization for high performance circuits and almost 50% reduction in leakage power for low power applications.

These optimized basic cells can be used to optimize complex circuit of any size by using the proposed techniques: Backward traversal replacement and Partitioning large basic cells reducing the computational time and overcoming the transient current issues. We have achieved upto 66% optimization in leakage power and upto 30% optimization in critical path delay for complex circuits. We have also proposed iterative decomposition technique to optimize power dissipation without impacting the latency for polar decoders case study

There is a lot of scope for improvement to this work. We list some of them here

- This optimization framework need not be limited to MOSFETs, it can be extended to advanced VLSI technologies, like SOI, SOS and FINFETs.
- This work can be extended to sequential circuits. Both, sequential basic cells and complex circuits can be optimized using the algorithm based transistor sizing methodology.
- The computational time can be greatly reduced by using accurate AI models to predict the leakages and delays avoiding the Hspice tool.

Related Publications

Conference Papers (published)

[1] H. S. Kalluru, P. Saha, A. Zahra and Z. Abbas, "Algorithm Driven Power-Timing Optimization Methodology for CMOS Digital Circuits Considering PVTA Variations," 2021 IEEE International Symposium on Circuits and Systems (ISCAS), 2021, pp. 1-5.

[2] P. Saha, H. S. Kalluru and Z. Abbas, "Transistor Sizing based PVT-Aware Low Power Optimization using Swarm Intelligence," 2021 34th International Conference on VLSI Design and 2021 20th International Conference on Embedded Systems (VLSID), 2021, pp. 234-239.

[3] P. Saha, S. Ahmed, H. S. Kalluru and Z. Abbas, "Low Power PVT-Aware Transistor Sizing and Approximate Design Generation for Standard Cells Using Swarm Intelligence," 2021 IEEE International Symposium on Circuits and Systems (ISCAS), 2021, pp. 1-5

[4] H. S. Kalluru and Z. Abbas, "Optimal Power-Area Polar Decoder Design based on Iterative Decomposition Technique," 2019 IEEE 16th India Council International Conference (INDICON), 2019, pp. 1-4, doi: 10.1109/INDICON47234.2019.9030332.

Journal (Under Review)

[1] Hema Sai Kalluru, Prasenjit Saha, Shirisha Gourishetty and Zia Abbas, “Optimization Algorithms driven Framework for High Yield Digital VLSI Designs" *(Submitted at Microelectronics Journal)*.

Bibliography

[1] John Vanston. Betterforecasts, betterplans,betterresults enhance the validity and credibility of your forecasts by structuring them in accordance with the five different ways people view the future. 07 2023.

[2] Sangwoo Han and Juho Kim. Nbti-aware statistical timing analysis framework. In *23rd IEEE International SOC Conference*, pages 158–163, 2010. doi: 10.1109/SOCC.2010.5784734.

[3] Gordon E. Moore. Progress in digital integrated electronics [technical literaiture, copyright 1975 ieee. reprinted, with permission. technical digest. international electron devices meeting, ieee, 1975, pp. 11-13.]. *IEEE Solid-State Circuits Society Newsletter*, 11(3):36–37, 2006. doi: 10.1109/N-SSC.2006.4804410.

[4] N.S. Kim, T. Austin, D. Baauw, T. Mudge, K. Flautner, J.S. Hu, M.J. Irwin, M. Kandemir, and V. Narayanan. Leakage current: Moore's law meets static power. *Computer*, 36(12): 68–75, 2003. doi: 10.1109/MC.2003.1250885.

[5] G. Baccarani, M.R. Wordeman, and R.H. Dennard. Generalized scaling theory and its application to a ¼ micrometer mosfet design. *IEEE Transactions on Electron Devices*, 31 (4):452–462, 1984. doi: 10.1109/T-ED.1984.21550.

[6] Magdy Bayoumi and Wai-Kai Chen. Introduction to vlsi systems. In WAI-KAI CHEN, editor, *The Electrical Engineering Handbook*, page 177. Academic Press, Burlington, 2005. ISBN 978-0-12-170960-0. doi: https://doi.org/10.1016/B978-012170960-0/50017-7. URL `https://www.sciencedirect.com/science/article/pii/B9780121709600500177`.

[7] Sankit Kassa and R.K.Nagaria. A review on robust low power system level digital circuit design approaches in nano-cmos technologies. 09 2015. doi: 10.1145/2818567.2818676.

[8] Trent Mcconaghy, Kristopher Breen, Jeffrey Dyck, and Amit Gupta. *Variation-Aware Design of Custom Integrated Circuits: A Hands-on Field Guide*, pages 169–186. 01 2013. ISBN 978-1-4614-2268-6. doi: 10.1007/978-1-4614-2269-3_6.

[9] K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand. Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits. *Proceedings of the IEEE*, 91(2):305–327, 2003. doi: 10.1109/JPROC.2002.808156.

[10] James Kao, Siva Narendra, and Anantha Chandrakasan. Subthreshold leakage modeling and reduction techniques. In *IEEE/ACM international conference on Computer-aided design*, pages 141–148. ICCAD, 2002.

[11] Farzan Fallah and Massoud Pedram. Standby and active leakage current control and minimization in CMOS VLSI circuits. *IEICE transactions on electronics*, 88(4):509–519, 2005.

[12] Amit Agarwal, Saibal Mukhopadhyay, Arijit Raychowdhury, Kaushik Roy, and Chris H Kim. Leakage power analysis and reduction for nanoscale circuits. *IEeE Micro*, 26(2): 68–80, 2006.

[13] Artur Wróblewski, Christian V. Schimpfle, and Josef A. Nossek. Automated transistor sizing algorithm for minimizing spurious switching activities in CMOS circuits. *2000 IEEE International Symposium on Circuits and Systems (ISCAS)*, 3:291–294, 2000.

[14] Kumar Yelamarthi and Chien-In Henry Chen. Transistor Sizing for Load balance of Multiple paths in Dynamic CMOS for Timing Optimization. *8th International Symposium on Quality Electronic Design (ISQED'07)*, pages 426–431, 2007.

[15] Zia Abbas and Mauro Olivieri. Optimal transistor sizing for maximum yield in variation-aware standard cell design. *International Journal of Circuit Theory and Applications*, 44 (7):1400–1424, 2016.

[16] Daniel K Beece, Chandu Visweswariah, Jinjun Xiong, and Vladimir Zolotov. Transistor sizing of custom high-performance digital circuits with parametric yield considerations. *Optimization and Engineering*, 15(1):217–241, 2014.

[17] John P. Fishburn and Alfred E. Dunlop. TILOS: A Posynomial Programming Approach to Transistor Sizing. In *ICCAD*, 2003.

[18] Kumar Yelamarthi and Chien-In Henry Chen. Process Variation Aware Timing Optimization through Transistor Sizing in Dynamic CMOS Logic. *9th International Symposium on Quality Electronic Design (ISQED)*, pages 143–147, 2008.

[19] Vikram G Rao and Hamid Mahmoodi. Analysis of reliability of flip-flops under transistor aging effects in nano-scale CMOS technology. In *2011 IEEE 29th International Conference on Computer Design (ICCD)*, pages 439–440, 2011. doi: 10.1109/ICCD.2011.6081439.

[20] Hao Luo and Mehrdad Nourani. Aging and Leakage tradeoff in VLSI circuits. In *10th International Design Test Symposium (IDT)*, pages 106–111, 2015. doi: 10.1109/IDT.2015.7396745.

[21] Z. Abbas, U. Khalid, and M. Olivieri. Sizing and optimization of low power process variation aware Standard cells. In *2013 IEEE International Integrated Reliability Workshop Final Report*, pages 181–184, 2013. doi: 10.1109/IIRW.2013.6804189.

[22] Kumar Yelamarthi and Chien-In Henry Chen. Process variation aware timing optimization through transistor sizing in dynamic cmos logic. In *9th International Symposium on Quality Electronic Design (isqed 2008)*, pages 143–147, 2008. doi: 10.1109/ISQED.2008.4479715.

[23] A Sasikumar and R Muthaiah. Operational amplifier circuit sizing based on nsga-ii and particle swarm optimization. In *2017 International Conference on Networks Advances in Computational Technologies (NetACT)*, pages 64–68, 2017. doi: 10.1109/NETACT.2017.8076742.

[24] R.A. Vural and T. Yildirim. Analog circuit sizing via swarm intelligence. *AEU - International Journal of Electronics and Communications*, 66(9):732–740, 2012. ISSN 1434-8411. doi: https://doi.org/10.1016/j.aeue.2012.01.003. URL `https://www.sciencedirect.com/science/article/pii/S1434841112000052`.

[25] Kim-Fung Man, Kit-Sang Tang, and Sam Kwong. Genetic Algorithms: concepts and applications [in engineering design]. *IEEE transactions on Industrial Electronics*, 43(5): 519–534, 1996.

[26] Russell Eberhart and James Kennedy. A new optimizer using Particle Swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE, 1995.

[27] A. Sasikumar and R. Muthaiah. Operational amplifier circuit sizing based on NSGA-II and Particle Swarm Optimization. In *2017 International Conference on Networks Advances in Computational Technologies (NetACT)*, pages 64–68, 2017.

[28] Revna Acar Vural and TÃ¼lay Yildirim. Analog circuit sizing via Swarm Intelligence. *AEU - International Journal of Electronics and Communications*, 66:732–740, 09 2012. doi: 10.1016/j.aeue.2012.01.003.

[29] P. Gupta, S. Gourishetty, H. Mandadapu, and Z. Abbas. PVT Variations Aware Robust Transistor Sizing for Power-Delay Optimal CMOS Digital Circuit Design. In *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5, 2019. doi: 10.1109/ISCAS.2019.8702068.

[30] P. Gupta, H. Mandadapu, S. Gourishetty, and Z. Abbas. Robust transistor sizing for improved performances in digital circuits using optimization algorithms. In *20th International Symposium on Quality Electronic Design (ISQED)*, pages 85–91, 2019. doi: 10.1109/ISQED.2019.8697718.

[31] Kyung Ki Kim and Yong-Bin Kim. A novel adaptive design methodology for minimum leakage power considering pvt variations on nanoscale vlsi systems. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 17(4):517–528, 2009. doi: 10.1109/TVLSI.2008.2007958.

[32] A. Gnudi, P. Ciampolini, R. Guerrieri, M. Rudan, and G. Baccarani. Sensitivity analysis for device design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 6(5):879–885, 1987. doi: 10.1109/TCAD.1987.1270330.

[33] Farshad Firouzi, Saman Kiamehr, and Mehdi B. Tahoori. Statistical analysis of BTI in the presence of process-induced voltage and temperature variations. In *2013 18th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 594–600, 2013. doi: 10.1109/ASPDAC.2013.6509663.

[34] Ali Asghar Heidari, Seyedali Mirjalili, Hossam Faris, Ibrahim Aljarah, Majdi Mafarja, and Huiling Chen. Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems*, 97:849–872, 2019. ISSN 0167-739X.

[35] Shinya Watanabe, Tomoyuki Hiroyasu, and Mitsunori Miki. Ncga: Neighborhood cultivation genetic algorithm for multi-objective optimization problems. In *GECCO Late Breaking Papers*, 2002.

[36] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002. doi: 10.1109/4235.996017.

[37] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. Spea2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. volume 3242, 01 2001.

[38] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999. doi: 10.1109/4235.797969.

[39] Krishnanand Kaipa and Debasish Ghose. Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions. *Swarm Intelligence*, 3: 87–124, 06 2008. doi: 10.1007/s11721-008-0021-5.

[40] Bogdan Tudor, Joddy Wang, Charly Sun, Zhaoping Chen, Zhijia Liao, Robin Tan, Weidong Liu, and Frank Lee. Mosra: An efficient and versatile mos aging modeling and reliability analysis solution for 45nm and below. In *2010 10th IEEE International Conference on Solid-State and Integrated Circuit Technology*, pages 1645–1647, 2010. doi: 10.1109/ICSICT.2010.5667399.

[41] "hspice: Mos reliability analysis (mosra)", [online]. URL `http://www.synopsys.com/products/mixedsignal/hspice/hspice.html`.

[42] Zia Abbas and Mauro Olivieri. Optimal transistor sizing for maximum yield in variation-aware standard cell design. *International Journal of Circuit Theory and Applications*, 44, 10 2015. doi: 10.1002/cta.2167.

[43] Mauro Olivieri and Antonio Mastrandrea. Logic Drivers: A Propagation Delay Modeling Paradigm for Statistical Simulation of Standard Cell Designs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 22(6):1429–1440, 2014. doi: 10.1109/TVLSI.2013.2269838.

[44] Mohammed Salman Ahmed and Zia Abbas. A memetic algorithm based pvt variation-aware robust transistor sizing scheme for power-delay optimal digital standard cell design. In *2019 IEEE 37th International Conference on Computer Design (ICCD)*, pages 385–392, 2019. doi: 10.1109/ICCD46524.2019.00060.

[45] Erdal Arikan. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory*, 55(7):3051–3073, 2009. doi: 10.1109/TIT.2009.2021379.

[46] Nadine Hussami, Satish Babu Korada, and Rudiger Urbanke. Performance of polar codes for channel and source coding. In *2009 IEEE International Symposium on Information Theory*, pages 1488–1492, 2009. doi: 10.1109/ISIT.2009.5205860.

[47] Jin Sha, Xing Liu, Zhongfeng Wang, and Xiaoyang Zeng. A memory efficient belief propagation decoder for polar codes. *China Communications*, 12(5):34–41, 2015. doi: 10.1109/CC.2015.7112042.

[48] Zia Abbas, Antonio Mastrandrea, and Mauro Olivieri. A voltage-based leakage current calculation scheme and its application to nanoscale mosfet and finfet standard-cell designs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 22(12): 2549–2560, 2014. doi: 10.1109/TVLSI.2013.2294550.

[49] Zia Abbas and Mauro Olivieri. Impact of technology scaling on leakage power in nano-scale bulk cmos digital standard cells. *Microelectronics Journal*, 45(2): 179–195, 2014. ISSN 0026-2692. doi: https://doi.org/10.1016/j.mejo.2013.10.013. URL https://www.sciencedirect.com/science/article/pii/S002626921300253X.

[50] ITRS. International technology roadmap for semiconductors, [online]. URL http://www.itrs2.net/itrs-reports.html.

[51] Bo Yuan and Keshab K. Parhi. Low-latency successive-cancellation polar decoder architectures using 2-bit decoding. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 61(4):1241–1254, 2014. doi: 10.1109/TCSI.2013.2283779.

[52] Karim H. Moussa, Ahmed H. El-Sakka, and Shawki Shaaban. Optimum polar codes encoder over binary discrete memory-less channels. *Journal of Physics: Conference Series*, 1447(1):012015, jan 2020. doi: 10.1088/1742-6596/1447/1/012015. URL https://dx.doi.org/10.1088/1742-6596/1447/1/012015.

[53] Amin Alamdar-Yazdi and Frank R. Kschischang. A simplified successive-cancellation decoder for polar codes. *IEEE Communications Letters*, 15(12):1378–1380, 2011. doi: 10.1109/LCOMM.2011.101811.111480.

[54] Rahul Shrestha and Utkarsh Rastogi. Design and implementation of area-efficient and low-power configurable booth-multiplier. In *2016 29th International Conference on VLSI Design and 2016 15th International Conference on Embedded Systems (VLSID)*, pages 599–600, 2016. doi: 10.1109/VLSID.2016.36.

[55] X. Xi J. He W. Liu M. Cao et al. M.W. Dunga, W. Yang. Bsim 4.6.1 mosfet model – user's manual, 2007, [online]. URL http://www.srware.com/xictools/docs/model_docs/bsim4.6.1/BSIM461_Manual.pdf.

[56] "predictive technology model", [online].