

OPTIMUM PHYSICAL DESIGN IMPLEMENTATION
OF BLOCK LEVEL NETWORKING ASIC
AND ITS POWER ESTIMATION

*A Dissertation Submitted in Partial Fulfillment of the Requirement for the Award of the
Degree of*

MASTER OF ENGINEERING

in Electronics and Communication Engineering

Submitted By

TEJ PAL SINGH

801561030

Under Supervision of

Er. Nitin Pant

Engineering Manager,

Intel Technology India Pvt. Ltd. Bangalore

and

Dr. Amit Kumar Kohli

Associate Professor, ECED, TU



ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT

THAPAR UNIVERSITY, PATIALA, PUNJAB, INDIA

JUNE, 2017

DECLARATION

I, **Tej Pal Singh**, hereby declare that the thesis entitled "**Optimum Physical Design Implementation of Block Level Networking ASIC and Its Power Estimation**" is a record of my own work carried out towards the partial fulfilment for the award of degree of Master of Engineering in Electronics and Communication Engineering at Thapar University, Patiala, under the guidance of **Er. Nitin Pant** (Engineering Manager, Intel Technology India Pvt. Ltd, Bangalore) and **Dr. Amit Kumar Kohli**, Associate Professor, Electronics and Communication Engineering Department, Thapar University, Patiala, India, from January 2017 – June 2017.

The matter presented in this thesis has not been submitted either in part or full to any other university or institute for the award of any other degree.

Date: 27/6/2017

Tej Pal Singh

Tej Pal Singh
Roll No. 801561030

It is certified that the above statement made by the student (Mr. Tej Pal Singh) is correct to the best of my knowledge and belief.

Nitin Pant

Mr. Nitin Pant

Engineering Manager

SDG, Networking Group

Intel Technology India Pvt. Ltd, Bangalore

INTEL TECHNOLOGY INDIA PVT. LTD.
CAMPUS-4B, ECO SPACE,
OUTER RING ROAD, BELLANDURU,
BENGALURU-560 103

Amit Kohli
27/06/2017

Dr. Amit Kumar Kohli

Associate Professor

ECED, Thapar University

Patiala-147004, (Punjab)

ACKNOWLEDGEMENT

Though only my name appears on the cover of this disquisition, a great many people have contributed to its production. I owe my gratitude to all those people who have made this dream possible and because of whom my graduate experience has been one that I will cherish forever.

To begin with, I would like to express my sincere requital towards my guide **Dr. Amit Kumar Kohli (Associate Professor, Department of Electronics and Communication Engineering, Thapar University, Patiala)**, for his consistent support and feedback, which helped me a lot in carrying out this thesis work. I would like to thank my manager, **Er. Nitin Pant (Engineering Manager, Intel Technology India Private Limited, Bangalore)** and my functional manager **Mr. Sarvesh (Digital Design Engineer, Intel Technology India Private Limited, Bangalore)** for their valuable suggestions, feedback and persuasion to do something new and challenging and to give freedom to work in my way. Their immense knowledge, guidance and support have helped me a lot during my internship time not only for technical but on non-technical part too.

I would like to thank my mentors **Mr. Amol Ram Harkare (SoC Engineer, Intel Technology)**, **Mr. Rahul Jain (SoC Engineer, Intel Technology)** and all team members for great support to learn basics and give me a proper guidance and extended support in my difficult time. I am highly indebted to my friend **Mr. Antariksh Verma (SoC Engineer, Intel Technology)** for his constant support and utmost cooperation during my internship at Intel.

I also very thankful to **Dr. R. S. Kaler (Deputy Director, Thapar University, Patiala)**, **Dr. S. S. Bhatia (DOAA, Thapar University, Patiala)** and **Dr. Alpana Aggarwal (Head of ECED)** for allowing me to do an internship and carrying out my thesis work at **Intel Technology India Pvt. Ltd. Bangalore.**

Last but not the least, I would like to thank my parents and friends for timely care and support to make me feel strong enough to complete this thesis.

Tej Pal Singh

801561030

ABSTRACT

In this thesis research work, an optimum physical design implementation technique for the block level networking application-specific-integrated-circuit (ASIC) has been presented, which significantly reduces the time for a chip to arrive in the market, and it alleviates the noise impact on critical nets. It also presents the power aware physical design flow, which helps in the reduction of the dynamic as well as leakage power.

With evolving technology nodes, the leakage power is less controllable from the designer's implementation point of view. However, the abstract level in digital design is decided mainly by the standard cell. Therefore, an approach is proposed in this research work to reduce the power consumption of the digital block. In this approach, the cell swapping/replacing technique has been utilized, which replaces the ultra-low threshold cells with the low threshold cells. Since, the power dissipation is inversely proportional to the threshold voltage level; therefore this technique is proved to be successful in the reduction of power dissipation.

In the current era, the physical implementation for any system-on-chip (SOC) is a time-consuming step; and in general, it would span from one year to two years. The main aim of the physical implementation is to optimize chip in terms of the speed, area, power, and cost. The prime concerns of any company are to reduce the time to market, the operation of chip at the desired frequency and to increase the durability of the product by reducing the power loss. Meeting the operational frequency requirement (corresponding to processing timing) is one of the major design concerns. Therefore, an algorithm has been proposed in this work indicating an optimum flow to fix the different violations observed during the physical design implementation.

To conclude, the main objective of the presented research work is to convert the register-transfer-logic (RTL) level netlist to the graphic-data-system (GDS II), which should be clean in design-rule-constraints (DRC), power efficient, timing clean, electron-migration (EM) clean. Therefore, the focus is on a design flow, which should be robust to tackle the last minute changes in the design, and it must be power aware along with the timing sign-off. The simulation results are presented to illustrate the efficacy and efficiency of the proposed algorithm for optimum physical design implementation of the block level networking ASIC and its power estimation.

Keywords: ASIC, SOC, RTL, GDS, DRC, EM.

TABLE OF CONTENTS

Sr. No	Name of the Chapters	Page No
	<i>Declaration</i>	<i>i</i>
	<i>Acknowledgement</i>	<i>ii</i>
	<i>Abstract</i>	<i>iii</i>
	<i>Table of Contents</i>	<i>iv</i>
	<i>List of Figures</i>	<i>vii</i>
	<i>List of Tables</i>	<i>ix</i>
	<i>Abbreviations and Acronyms</i>	<i>xi</i>
<i>Chapter 1</i>	Introduction	1-14
<i>1.1</i>	Libraries in physical design.....	<i>1</i>
<i>1.2</i>	Floorplanning.....	<i>2</i>
<i>1.2.1</i>	Technology Files and Library Files.....	<i>2</i>
<i>1.2.2</i>	Circuit Description.....	<i>2</i>
<i>1.2.3</i>	Design Constraints.....	<i>2</i>
<i>1.2.4</i>	Design Planning and Pad Placement.....	<i>4</i>
<i>1.2.5</i>	Macros and Standard Cell Placement.....	<i>5</i>
<i>1.2.6</i>	Power Planning and Power Mesh Design.....	<i>7</i>
<i>1.2.7</i>	Clock Planning.....	<i>8</i>
<i>1.3</i>	Placement.....	<i>9</i>
<i>1.3.1</i>	Global Placement.....	<i>9</i>
<i>1.3.2</i>	Detail Placement.....	<i>10</i>
<i>1.4</i>	Routing.....	<i>10</i>
<i>1.5</i>	Extraction.....	<i>13</i>
<i>1.6</i>	Problem Statement.....	<i>13</i>
<i>1.7</i>	Organization of Thesis.....	<i>13</i>
<i>1.8</i>	Chapter Summary.....	<i>14</i>
<i>Chapter 2</i>	Literature Survey	15-22
<i>Chapter 3</i>	Configuring the Static Timing Analysis Environment	23-34
<i>3.1</i>	Introduction to STA Environment.....	<i>23</i>
<i>3.2</i>	Specifying Clocks.....	<i>23</i>
<i>3.3</i>	Clock Uncertainty.....	<i>24</i>
<i>3.4</i>	Clock Latency.....	<i>25</i>
<i>3.5</i>	Generated Clocks.....	<i>26</i>

3.6	Timing Path Groups.....	28
3.7	Constraining Input/Output Path.....	29
3.8	Design-Rule-Check (DRC) and Crosstalk Delay Analysis.....	30
3.9	Multicycle and False Path.....	32
3.10	Chapter Summary.....	33
<i>Chapter 4</i>	<i>Methodology to Fix Timing and Power in Physical Design.....</i>	<i>35-54</i>
4.1	Setup Timing Analysis in Physical Design.....	35
4.2	Hold Timing Analysis in Physical Design.....	37
4.3	Setup and Hold Timing Analysis Timing Report.....	39
4.4	Dynamic and Static Power Dissipation in Physical Design.....	41
4.4.1	Sources of Power Dissipation.....	43
4.4.2	Dynamic Power Analysis and Its Optimization by Cell Replacing/Swapping.....	44
4.5	Leakage Power Analysis and Its Optimization by Cell Replacing/Swapping.....	46
4.5.1	Different Leakage Power Reduction Techniques.....	47
4.6	Power Dissipation and Junction Temperature.....	50
4.7	Proposed Procedural Algorithm Defining Optimum Flow in Physical Design To Fix Timing and Power Violations.....	51
4.8	Chapter Summary.....	54
<i>Chapter 5</i>	<i>Simulation Results.....</i>	<i>55-64</i>
5.1	Simulation Results of Physically Implemented Design.....	55
5.1.1	Physically Implemented Block Design Statistics.....	55
5.1.2	Physically Implemented Block Utilization Statistics.....	56
5.2	Static Timing Analysis Simulation Results.....	56
5.2.1	Different Path Groups and Clocks.....	56
5.2.2	Setup Timing and Hold Timing Violations During The First Iteration..	57
5.2.2.1	Before Fixing.....	57
5.2.2.2	After Fixing.....	58
5.2.3	Setup Timing and Hold Timing Violations During The Second Iteration.....	59
5.2.3.1	Before Fixing.....	59
5.2.3.2	After Fixing.....	59
5.2.4	Setup Timing and Hold Timing Violations During The Third Iteration.	60
5.2.4.1	Before Fixing.....	60

5.2.4.2	After Fixing.....	60
5.2.5	Setup Timing and Hold Timing Violations During The Fourth Iteration.....	61
5.2.5.1	Before Fixing.....	61
5.2.6	Graphical Representation of Violations	62
5.3	Chip Power Estimation Simulation Results.....	62
5.3.1	Ultra-Low-Threshold-Voltage (ULVT) and Low-Threshold-Voltage (LVT) Cell Statistics.....	62
5.3.2	Power Estimation Comparison Before and After Cell Swapping/Replacing (in Layout Design).....	63
5.3.3	Final Chip Power Estimation.....	63
<i>Chapter 6</i>	Conclusion and Future Scope.....	65-66
	<i>References</i>	67-70

LIST OF FIGURES

Sr. No	Figure Details	Page No.
Figure 1.1	Gate level representation	3
Figure 1.2	ASIC design initial floorplan	5
Figure 1.3	Floorplanning with standard cell and macros placement	6
Figure 1.4	Cell density map	6
Figure 1.5	Blockages	7
Figure 1.6	Ring and core's power and ground	7
Figure 1.7	Implemented power mesh	8
Figure 1.8	Clock distribution network	8
Figure 1.9	Congestion in the design due to inefficient placement	9
Figure 1.10	Cluster based cell placement	10
Figure 1.11	Channel based routing	11
Figure 1.12	Global routing	12
Figure 1.13	Congestion map	12
Figure 3.1	A synchronous design	23
Figure 3.2	A clock representation	24
Figure 3.3	Clock uncertainty	25
Figure 3.4	On-chip source latency	25
Figure 3.5	Off-chip clock source and latency	26
Figure 3.6	Generated clock in the design	27
Figure 3.7	Clock latency on generated clock	27
Figure 3.8	Different timing paths	28
Figure 3.9	Different timing path groups	29
Figure 3.10	Input port timing path	29
Figure 3.11	Output port timing path	30

Figure 3.12	Crosstalk impact	31
Figure 3.13	Multicycle path	33
Figure 4.1	Setup analysis basic architecture	35
Figure 4.2	Clock and data signal for setup analysis	36
Figure 4.3	Hold analysis basic architecture	37
Figure 4.4	Clock and data signal for hold analysis	38
Figure 4.5	CMOS inverter	41
Figure 4.6	Inverter switching voltage, energy, power and current waveforms	43
Figure 4.7	Short circuit current in CMOS inverter	44
Figure 4.8	Multi-threshold CMOS structure	48
Figure 4.9	Proposed procedural algorithm defining optimum flow in physical design for timing and power fixing	53
Figure 5.1	Setup and hold violations	62

LIST OF TABLES

Sr.No	Table Details	Page No.
Table 3.1	Delay Calculation with No Crosstalk	31
Table 3.2	Aggressor Switching in The Same Direction Negative Crosstalk	32
Table 3.3	Aggressor Switching in The Opposite Direction Positive Crosstalk	32
Table 4.1	Dependence of Sub-threshold Leakage on Device Parameters	47
Table 5.1	Design Statistics	55
Table 5.2	Chip Utilization	56
Table 5.3	Different Path Groups Present in The Design	57
Table 5.4	Different Types of Clocks and Their Time Period	57
Table 5.5	First Iteration Setup Violations Among Different Path Groups Before Fixing	58
Table 5.6	First Iteration Hold Violations Among Different Path Groups Before Fixing	58
Table 5.7	First Iteration Setup Violations Among Different Path Groups After Fixing	58
Table 5.8	First Iteration Hold Violations Among Different Path Groups After Fixing	58
Table 5.9	Second Iteration Setup Violations Among Different Path Groups Before Fixing	59
Table 5.10	Second Iteration Hold Violations Among Different Path Groups Before Fixing	59
Table 5.11	Second Iteration Setup Violations Among Different Path Groups After Fixing	59
Table 5.12	Second Iteration Hold Violations Among Different Path Groups After Fixing	60
Table 5.13	Third Iteration Setup Violations Among Different Path Groups Before Fixing	60
Table 5.14	Third Iteration Hold Violations Among Different Path Groups Before Fixing	60
Table 5.15	Third Iteration Setup Violations Among Different Path Groups After Fixing	61

Table 5.16	Third Iteration Hold Violations Among Different Path Groups After Fixing	61
Table 5.17	Fourth Iteration Setup Violations Among Different Path Groups Before Fixing	61
Table 5.18	Fourth Iteration Hold Violations Among Different Path Groups Before Fixing	61
Table 5.19	Ultra-Low and Low Voltage Threshold Cell Count Before Cell Swapping/Replacing	62
Table 5.20	Final Cell Count After Closing The Design	63
Table 5.21	Power Estimation Before and After Cell Replacing/Swapping	63
Table 5.22	Analysis Conditions	64
Table 5.23	Worst Case Power	64

ABBREVIATIONS AND ACRONYMS

AOCV	Advanced On-Chip Variation
ASIC	Application Specific Integrated Circuit
CLK	Clock
CMOS	Complementary Metal Oxide Semiconductor
CPPR	Common Path Pessimism Removal
CTS	Clock Tree Synthesis
DAC	Digital to Analog Converter
DFFC	Dynamic Flip-Flop Conversion
DRC	Design Rule Check
DUA	Design Under Analysis
FET	Field Effect Transistor
FinFET	Fin Field Effect Transistor
GBA	Group Based Approach
GRC	Global Routing Congestion
GUI	Graphical User Interface
HVT	High Voltage Threshold
LVT	Low Voltage Threshold
OCV	On Chip Variation
PBA	Path Based Approach
PNR	Placement and Route
PLL	Phase Lock Loop
RTL	Resistor Transistor Logic
STA	Static Timing Analysis
ULVT	Ultra Low Voltage Threshold
VLSI	Very Large Scale Integration
VR	Voltage Regulator
w.r.t	With Respect To

CHAPTER 1

INTRODUCTION

VLSI industry is growing by obeying Moore's law [1,2,3], where a single chip can accommodate multiple processors. To achieve scalability in the design process without sacrificing the quality of the design is becoming difficult. There is an impending requirement to do the efficient placement and physical design process that can address high-quality results in shorter time. This chapter explains the flow of the physical design process and the efficient ways to do the processes at the required stages.

1.1 LIBRARIES IN PHYSICAL DESIGN

For the physical design implementation of an application-specific-integrated-circuit (ASIC), various types of libraries are required along with different data sets. These libraries are stacks of the abstract view, transistor level circuit description, functional and simulation model and physical layout. The accuracy of libraries is very important for the physical design implementation, as it has a direct influence on the success of design. In ASIC design the standard cell libraries and input-output (IO) pads are generally utilized along with memory and custom libraries. Memory compiler provides the information regarding layout, simulation, abstract and timing views. Intellectual-property (IP), which are known as custom libraries, are the collection of manually crafted analog function layouts. The manually crafted analog functions are phase-lock-loop (PLL), digital-to-analog-converter (DAC) and voltage-regulator (VR). The basic building block of physical design implementation are standard cells and input-output pads.

In order to attain an optimum die dimension during the physical design implementation, extra care is required to be taken during standard cell libraries formation. It is a collection of desired gates in the design. Because of the advancement in the fabrication process, the routing area is dominating the transistors area. Therefore, it is necessary to reduce the routing area in comparison to transistor size.

The standard cell sizing is defined in the libraries to help the Place-and-Route (PNR) tool to do the routing. Input-output padding is one of the most difficult elements in the physical design implementation. It requires a detailed understanding and the deep circuit expertise to fabricate ASIC design padding, which is required to translate the signal level utilized in ASIC core and on the outer of core area. It also clamps the signal to power as well as ground rails. This clamping helps in protecting the chip from electrostatic discharge.

1.2 FLOORPLANNING

Floorplanning is the art of physical design implementation. An optimum floorplan leads to a high-performance ASIC design with optimal area utilization. Many important things come under floor planning, that is the placement of input-output (IO) pads, standard cells, macros and power/ground structures. Before doing the floorplanning, important data is required and one requires making sure that this information set is available. The data, which is needed to start with a physical design is as follows

1.2.1 Technology Files and Library Files

All PNR tools operate on technology file. It is used to configure structures and parameters, and it also limits the ASIC design to the specific technology. All the parameters which are mentioned in the technology file should be correct. Once all the parameters are set, several trial runs should be performed and results should be analyzed carefully. The basic technology rules are as follows

1. Manufacturing grid
2. Routing grid
3. Via definition
4. Antenna definition
5. Routing layer physical definition
6. Layer definition for placement and routing blockage
7. Layer density rule.

1.2.2 Circuit Description

In today's ASIC flow, RTL is used to design the circuits. Figure 1.1 shows a schematic of structural Verilog netlist implemented by using the verilog-hardware-description-language (VHDL) [1].

1.2.3 Design Constraints

During logic and physical design synthesis, the design constraints are ASIC design specifications. There are two major design constraints

1. Design-rule-constraints (DRC)
2. Timing constraints

Timing constraints are defined by the user based upon the speed, area, and power consumption of the design. Timing constraints, which are used during the physical design implementation, are as follows

1. False paths
2. Multiple cycle path
3. Input as well as output delays
4. Minimum and maximum path delays
5. Input transition and output load capacitance
6. System clock definition and clock delays

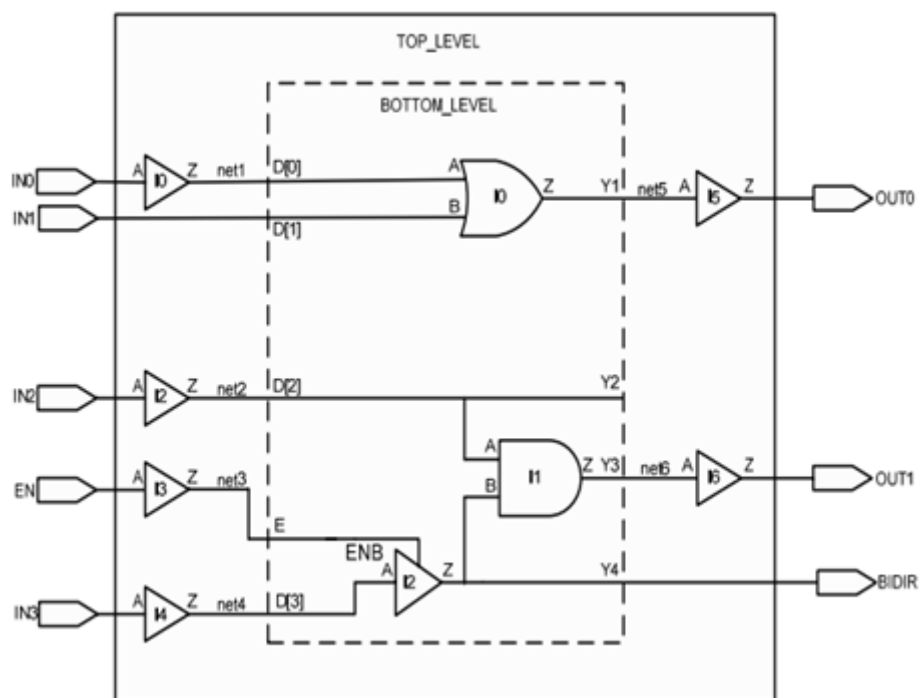


Figure 1.1 Gate level representation [1]

System clocks along with its delay are extremely important in ASIC design. All delays in the design depends upon the system clocks, which are generated inside and/or supplied externally. Design rule constraints are of higher importance in comparison to the timing constraints. These constraints need to be satisfied in an approach to meet the functional ASIC design requirements. The different important types of the design constraints are as follows

1. Maximum transition
2. Maximum capacitance
3. Maximum fan-out
4. Maximum wire-length

Maximum fan-out depicts the maximum number of cells, which could be connected to the output of a cell. Maximum transition explains the maximum amount of input transition, allowed for each cell. Maximum capacitance is similar to the maximum transition, but here the measurement is depending on total capacitance. Maximum wire length design rule constraint is used to control the maximum length of the wires to be used.

1.2.4 Design Planning and Pad Placement

Efficient design implementation depends upon the efficient style and planning approach that helps in achieving the area goals and timing requirements in the design implementation. There are two design techniques for design planning i.e., one is ASIC flat and the other one is ASIC hierarchical. For small and medium, flattening the design is most suitable, and for the larger designs, the hierarchical approach is very useful, which is portioning the design into the sub designs.

In a hierarchical design, the partitioning is of two types i.e., physical partitioning and logical partitioning. The logical partitioning occurs in early stages of the design during RTL synthesis, and physical partitioning takes place during physical design implementation. For minimizing delay in the design, the physical partitioning can be utilized. For this to be implemented at the top level, the port placement and dimensions of the partitioned block must be mentioned. The relationship between the perimeter of each partitioned block and its terminals is as follows

$$P=NS \tag{1.1}$$

where, P denotes the perimeter of the block, N indicates number of ports, and S is the spacing between ports.

For correct functioning of ASIC design, the proper placement of IO pads is very important. There are three types of IO pads i.e., power, ground and signal. These pads should be placed approximately in order to get rid of the electromigration effect and current switching noise related problems [1].

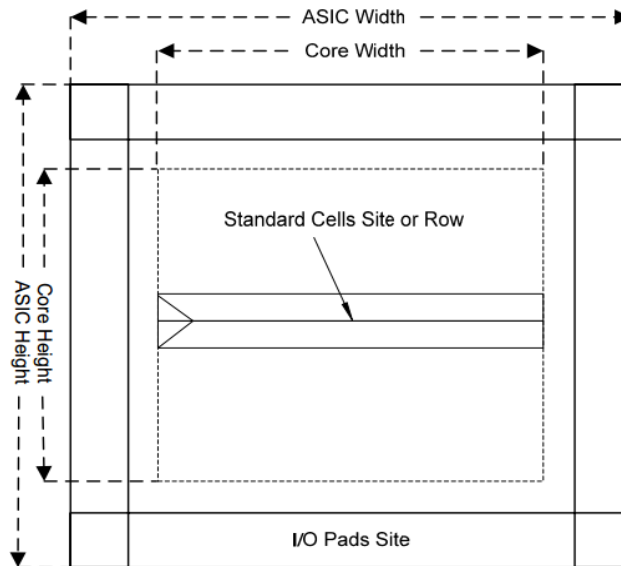


Figure 1.2 ASIC design initial floorplan [1]

1.2.5 Macros and Standard Cell Placement

As soon as the IO placement is done, the placement of macros and standard cells take place. Macros can be memories or any analog block. An appropriate placement of the macros directly influences the timing as well as the quality of the design. The placement of macros may either manual or automatic. The manual macro placement has been preferred in a case, when there are a fewer number of macros in the design and logic is also known. On the other hand, if the macros are large in number, then the automated process is more accurate to place the macros in the design. Most of the today's designs place the initial macros based upon the connectivity by using a global placer.

Optimum placement of the macros in the design not only helps in obtaining the timing goals, but it also helps to alleviate the required area. The major problem during the macro placement is that the space and timing issues need to be resolved simultaneously. Figure 1.3 shows the placement of the macros from the block implementation.

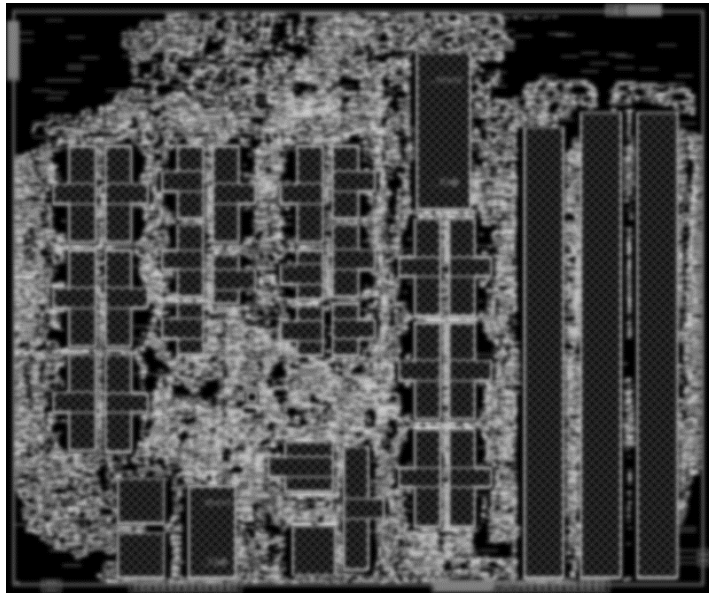


Figure 1.3 Floorplanning with standard cell and macros placement

Macros and standard cell placement can be analyzed by investigating the density map of the design. Figure 1.4 shows a cell density map.

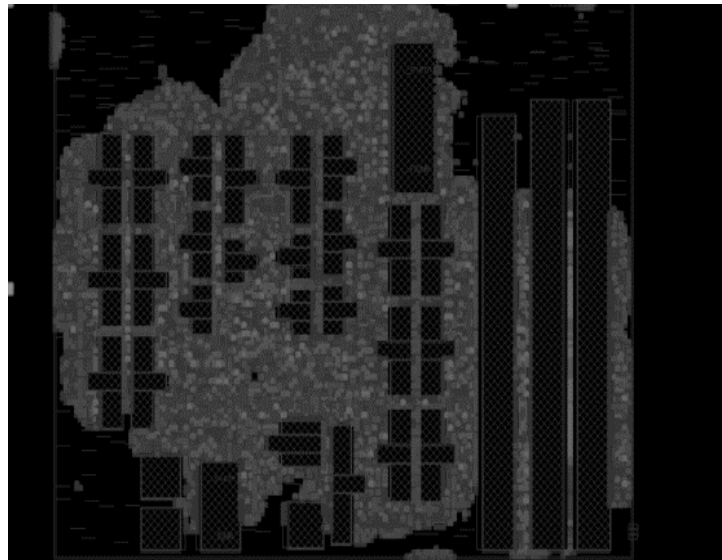


Figure 1.4 Cell density map

Blockage layers are created to prevent the placement of cells in front of the macros pin and to prevent the routing congestion around the macros corners. When macros are blocked on a number of routing layers, the wires exhibit a tendency to detour around corners, and these connect to nearby standard cells, thereby generating the routing congestions [1]. Figure 1.5 demonstrates the blockage area around the macros and corners of the die area.

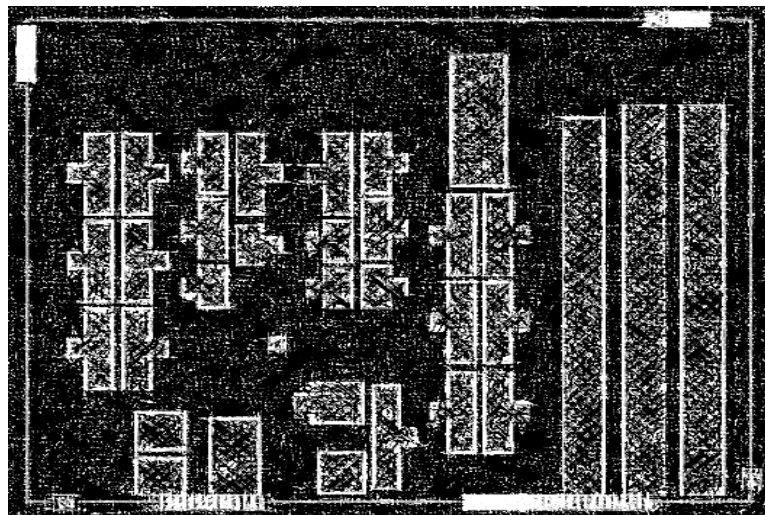


Figure 1.5 Blockages

1.2.6 Power Planning and Power Mesh Design

The next important step is to form power as well as ground structures for IO pads and core logic. The IO pads have power as well as ground buses, and these are also connected by abutment. For core logic, a core connects the core with single or with a number of ground as well as power rings.

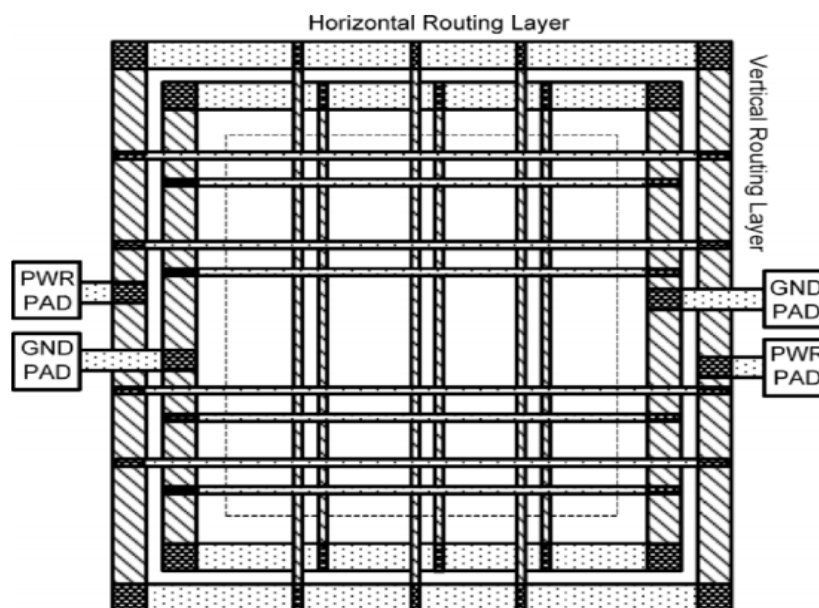


Figure 1.6 Ring and core's power and ground [1]

The horizontal and vertical metal layers are needed to define the top, bottom, left and right metal layers respectively. These horizontal and vertical metal layers are connected by using via cut. After this, the standard cell power connection is required, that is interior to core logic.

Such internal power buses composed of single or more segments of wires, which repeat after set intervals of time.

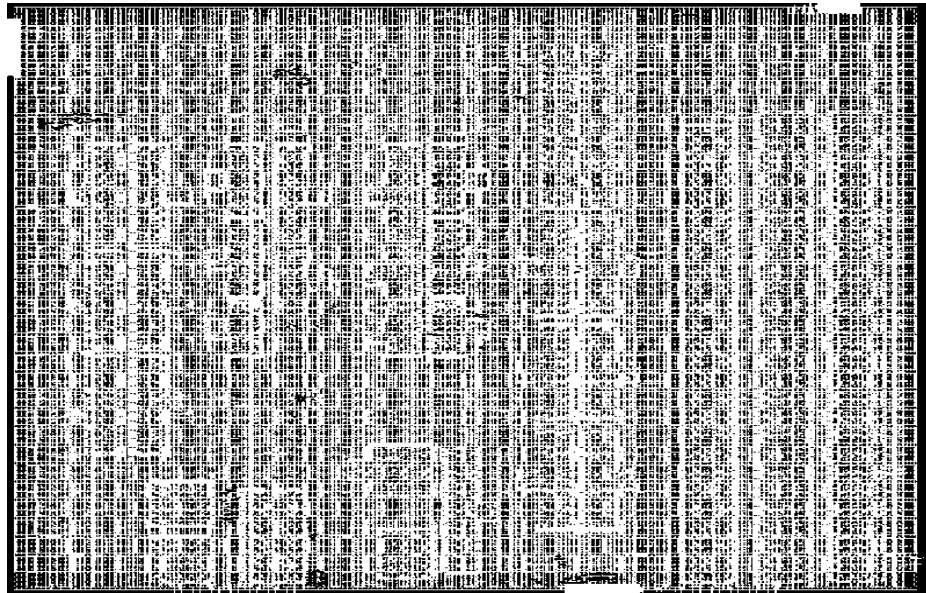


Figure 1.7 Implemented power mesh

1.2.7 Clock Planning

In order to provide a clock to all the clocked elements, an appropriate clock planning is required. Most of the ASIC designs use the clock tree synthesis, which is most suitable for high-performance chips. In such cases, the manual implementation of the distributed clock networks will help to reduce the skew between elements because of their capacitance and resistance. The key idea behind this is to build the low capacitance/resistance grid, which covers the whole logic as depicted in Figure 1.8

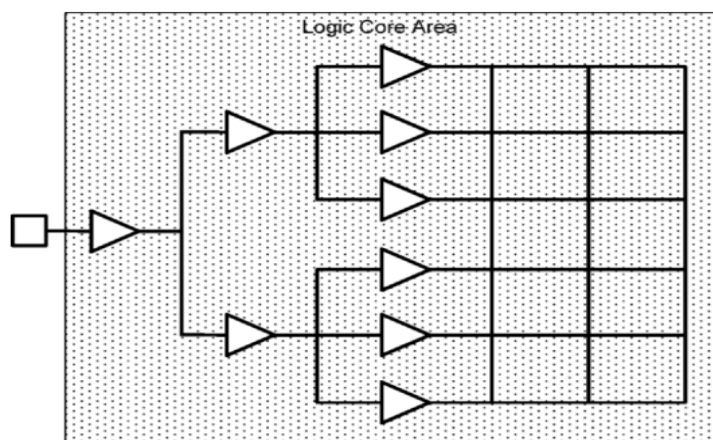


Figure 1.8 Clock distribution network [1]

This is noteworthy that the matching of components is not required in this type of clock grid. However, it presents a system clock skew to reduce the clock skew among all the leaf nodes.

1.3 PLACEMENT

The most challenging work in ASIC design synthesis is the insertion of buffers and the standard cell placement along clock path. The goal of ASIC design synthesis is to place the standard cells in the core area that is denoted using rows. This placement of standard cells should be such that there are no timing issues, and the routing should be done efficiently. The optimal placement of standard cells has direct impact on the area utilization. If standard cells are not placed optimally, it leads to the congestion in the underlying design. Figure 1.9 demonstrates a congested design.

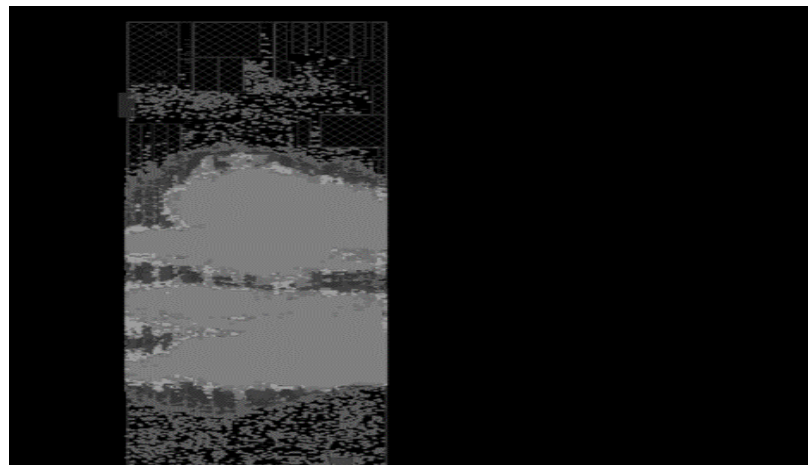


Figure 1.9 Congestion in the design due to inefficient placement

Placement is a two-step approach used by PNR tools. These steps are as follows

1. Global placement
2. Detail placement

The main aim of global placement is to decrease length of the wires; however the objective of the goal of detailed placement is to satisfy the design constraints.

1.3.1 Global Placement

Standard cells are in the floating shape before the placement. These standard cells are located arbitrarily in ASIC core; however, no particular position is assigned to it. By partitioning the standard cell area, a collection of cells can be assigned to such areas. Cluster and region options are supported by every place and route tool. Figure 1.10 illustrates a cell cluster.

During the placement, the cell cluster will try to place the group of cells in the cluster near to each other.

1.3.2 Detail Placement

Once the placement is done, the next step is detailed placement to improve their location on the basis of timing, congestion and power specifications. This placement is congestion and timing driven. For optimal minimum device dimensions, a number of routing layers can be used. Timing driven placement uses either net based algorithm or path-based algorithm.

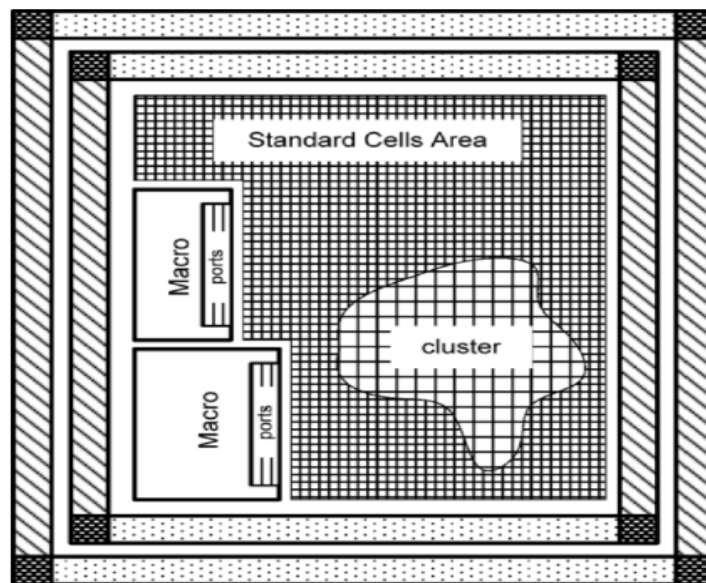


Figure 1.10 Cluster based cell placement [1]

1.4 ROUTING

After completing standard cell placement, subsequently important step is routing and extracting the parasitic for doing the static timing analysis. Because of an increase in the number of the standard cells and increase in the complexity of the ASIC design, the routing has become more difficult nowadays. Because of inefficient routing, a design may fail or may give the bad results. Besides the algorithm used for routing, the major factors which influence the routing, are the placement of standard cells, floor planning and layout of standard cells. There are different routing algorithms which are explained as follows

A). Channel Based Routing

Channel based routing has been in practice long back, when the metal layers were restricted to two or three. Routers used free space between the standard cells and the feed through to perform the routing. Figure 1.11 indicates the channel based routing.

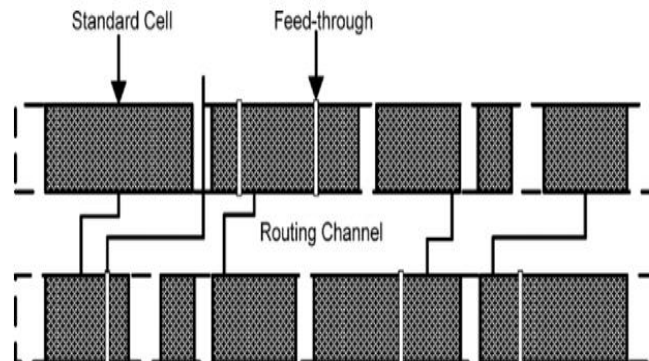


Figure 1.11 Channel based routing [1]

B). Special Routing

This type of routing is utilized in case of standard cells, macros as well as ground connections. Line probe technique utilizes the line segments for routing. It uses connectivity targets, such as port-to-port, line-to-line or port-to-line for connections. These connections don't pass through any obstruction; if they do so, they may cause a typical irreparable design rule constraint. By using the size of the ports, the probe method connects the ports of the macros with memory.

C). Global Routing

In global routing, the routing regions are defined in rectangular shape, and their corresponding density is calculated. These regions are called global-routing-cells (GRC's). The cell height, width, routing layers and spacing of wires define the maximum number of nets passing through the region. These parameters have been specified in technology file. After the global routing has been done, pin location is finalized to have the minimal connectivity of the standard cells in the core. The GRC statistics give a very good picture of the routing congestion in the design.

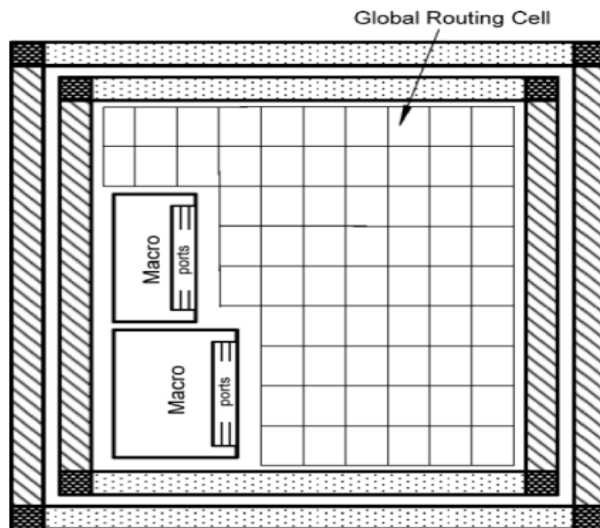


Figure 1.12 Global routing [1]

D). Detail Routing

The main goal of detail routing is to obey global routing as well as to make real physical interconnections.

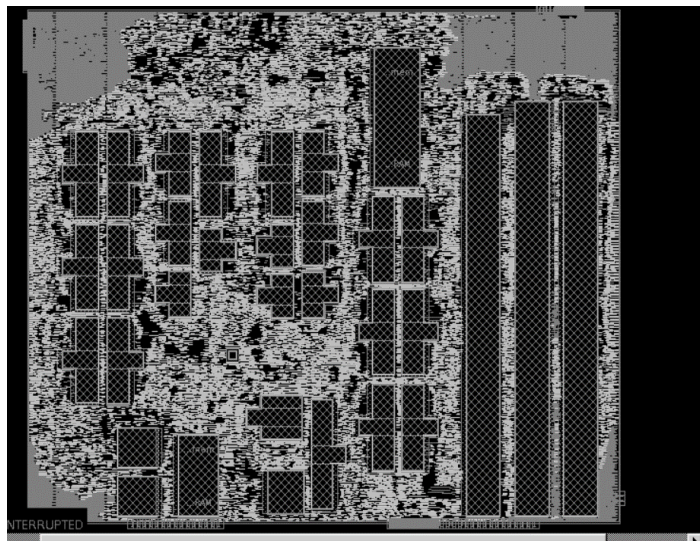


Figure 1.13 Congestion map

Therefore, once the region has been indicated by global router, then actual placement of the wires is done by the detailed router. Detail routing uses the vertical and horizontal routing grid for the actual routing.

1.5 EXTRACTION

Parasitic extraction is computation of capacitance and resistance values of the routed nets for the calculation of the delay, STA and signal integrity analysis. The most significant parameters, which are needed to retrieve the routing parasitic, are resistance, capacitance as well as inductance. Most of the design tools nowadays are using only the resistance and capacitance extraction. To extract the values of the resistance and capacitance foundries provide the sheets having the values of the resistance and capacitance defined in it based on the electrical domination.

These electrical parameters are dependent on the measurement conducted by utilizing test keys from practical silicon data in case of best, worst and typical scenarios.

1.6 PROBLEM STATEMENT

This thesis presents the following work

1. Firstly, optimum physical design implementation of a block along with its STA analysis has been done. Subsequently, an algorithm has been proposed indicating the optimum flow for fixing the violations observed in the digital design.
2. Secondly, to reduce the leakage power consumption in digital design, an approach based on the replacement/swapping of the ultra-low threshold voltage cells with the low threshold voltage cells has been reported.

1.7 ORGANIZATION OF THESIS

The presented thesis is organized as

Chapter I summarizes the basic problem statement of the research work, and gives the overview of physical design essentials.

Chapter II describes the literature survey related to the field of underlying research work.

Chapter III discusses the approach for configuring the static timing analysis. All the timing concepts have been discussed in this chapter.

Chapter IV details an algorithm to fix the timing in the design and also proposes an approach to decrease the leakage power consumption/dissipation in the design. Analytical expressions have also been developed in this chapter to express the relation between the leakage power and the threshold voltage.

In Chapter V, simulation results are presented related to the digital block.

Finally, Chapter VI concludes the research work and it also suggests the future scope of research work presented in this dissertation.

1.8 CHAPTER SUMMARY

This chapter provides a brief introduction of various important steps involved during the physical design implementation. In libraries section of this chapter, the different important libraries required during the physical design implementation has been discussed. In placement section, the placement of standard cells and macros has been discussed along with the clock tree synthesis and clock routing. In routing section, various routing techniques have been explored. Extraction section sheds light on important parameters required for the extraction purpose.

CHAPTER 2

LITERATURE SURVEY

This chapter gives an overview regarding the literature survey. Literature survey of different papers, related to VLSI chip design, has been mentioned in this chapter.

In [4], the dynamic timing simulation of complex mixed signal IPs are explained. In this paper the targeted IP is a memory, and by using its behavioral model, various checks, like IO path delays, timing checks viz. setup timing and hold timing and its impact on the IP behavior, are performed. This approach is validated on universal-verification-model (UVM) semantics. This methodology has helped in catching the systemic and strategic bugs. The proposed methodology in this paper is applicable to any analog-mixed-signal (AMS) IPs.

In [5], Wei *et al.*, have given a method to reduce the congestion issues in the physical design. As the technology is shrinking day by day, the area resources are becoming more deficient. To solve the problem of congestion due to the area, an “Always-on” buffer cluster implementation method has been proposed in [5]. The traditional method of physical design was not able to make full use of resources. Therefore, a low power physical optimization method named as “Always-on” buffer cluster implementation has been proposed. This method helps in both the power consumption and also save area resources, which further helps to resolve the congestion issues in the physical design flow.

In [6], the dynamic-flip-flop-conversion (DFFC), which is a time borrowing method, is explained. This approach helped to improve the timing slack. Older methods suffered from the false error prediction, which means even though there are no setup timing violations in the design, the older methods incorrectly issued timing error. In this paper, the authors proposed an improved DFFC method, which gives accurate timing results and consumes lesser power. This method is tested on various critical paths, and it has shown accurate results.

In [7], Luo *et al.*, proposed a fast and simple common-path-pessimism-removal (CPPR) procedure with sub-quadratic timing complexity. The time closure is always the main concern in any physical design process. The STA analysis is the way to compute the design frequency, and the design engineers spend months to detect the timing violations observed in designing. To overcome such issues observed because of on-chip-variation (OCV) and other factors, the pessimism removal plays a very important role. In this paper, an approach has been proposed for the timing closure.

In [8], a discussion regarding the leakage and dynamic power dissipation has been presented. It has been proved that with the previous technologies, the dynamic power was dominating

more than the leakage power, but in today's nanometre CMOS techniques, the leakage power has become a major concern. In [8], Singh and Tripathi have proposed an idea to simultaneously reduce the dynamic as well as leakage power consumption.

In [9], Jung *et al.*, experimentally compared single-mesh and multi-mesh designs for test circuits and proved that the multiple meshes consume smaller clock power than the single mesh, but exhibits larger skew and larger clock wire length. It has been further studied that how multiple meshes should be floor planned in the design. A practical physical design, because of hierarchical gating structure, requires more than one mesh. A single mesh can also support the same architecture, but at the cost of efficiency of gates.

In [10], Harris has explained about the clock uncertainty and its role in timing analysis. Sequential digital circuits have been built using combinational logic separated by clocked sequencing elements, like flip-flops and latches. Maximum clock frequency in the static timing analysis is estimated depending on timing specifications of combinational logic as well as sequential elements. The important parameters for timing are setup timing as well as clock-to-Q delay. But, this definition has a weakness that the definition does not include any clock uncertainty. This paper proposed the effects of uncertainty on the timing of the design.

In [11], to appraise scalability as well as performance of monolayer/multilayer 2D semiconductor-based FET's in deep micron technologies, a dissipative transport simulation utilizing no equilibrium Greens function formalism is conducted. In this, a roadmap for the semiconductor requirement is analyzed. Molybdenum disulfide is used for reduced channel length device to achieve high performance.

In [12], an overview of common-path-pessimism-removal (CPPR) is given. Transistors process parameters vary within the same die along with the variation across multiple manufacturing lots. These changes are called on-chip-variation (OCV) and it also effect wire as well as cell delays. Chip timing is set against these updated delays to make sure the proper working of chip. Excess pessimism in any clock tree network skews practical timing of circuit, which means that design may work accurately on low frequencies than the actual silicon. Therefore in order to achieve the true performance of the design, such pessimism is required to be suppressed. In this paper, author explained about the pessimism to achieve the timing goals.

In [13], Fuang and Mahyuddin have compared the implementation and parameters of advanced on-chip variation of path-based approach and conventional approach. This experiment has been performed on lower technology nodes. Advanced-on-chip-variation (AOCV) utilized to rectify the timing for on-chip-variation (OCV), based on distance and logic depth of the path. In this paper, two types of AOCV are discussed, one is group-based-

analysis (GBA) and other is path-based-analysis (PBA). Here path-based-analysis (PBA) is an improved version of the graph-based analysis. Path-based-analysis (PBA) reduces pessimism using the advanced algorithm.

In [14], Monthie *et al.*, have analyzed the influence of power supply's noise, in high-speed memory interfaces, on the clock jitter. With the scaling down of technology, the requirement of on-chip power distribution systems has substantially enhanced. With an increase in the number of transistors, there is an increase in average current density and power noise magnitude. Reduced supply voltage can cause chip failure. With these challenges, the most critical task is to achieve too small jitter to satisfy the timing budget. This paper gives a clear explanation about the power supply noise that is a major source of timing jitter.

In [15], an idea has been proposed to reduce the instantaneous voltage drop due to glitch. Glitch is an undesired transition, and a method has been proposed to compensate this. A glitch occurs, when an inertial delay is less than the differential delay at the inputs of a gate, which results in large gate switching and hence power consumption increases. These types of a large number of logic gates give rise to IR drop. Insertion of buffers at the input of the logic gates may reduce the glitch, but it results in an increase in the area and dynamic power. In the methodology proposed in [15], the transmission gate is used as a compensation circuit to reduce dynamic and extra leakage power.

In [16], Wiltgen *et al.*, have proposed a method to reduce the short circuit power in CMOS design. For estimating the ratio of change of short-circuit power to the dynamic power, many analytical expressions are developed. It has been proved that ratio of the threshold voltage to drain voltage reduces significantly with an increase in the short circuit to dynamic power ratio. In [16], authors have also suggested an approach for replacing the low-voltage-threshold (LVT) transistors with high-voltage-threshold (HVT) transistors in the same process, which helps in reducing short circuit power in design. It has been verified and has very little effect on the timing of the design.

In [17], Priya *et al.*, have explained about the leakage power reduction techniques. The leakage power has become a critical issue in the low power VLSI circuit designs. Due to a decrease in the threshold voltage and increase in the number of cells, the sub-threshold leakage current increases, which result in the leakage power. This leakage power affects the battery operated devices very badly during the standby mode. Therefore, the leakage power dissipation plays a very important role in deep submicron technology analysis. This paper gives an explanation about the various leakage power reduction techniques.

In [18], Morgenshtein has addressed the power consumption, through a study that assumes transistor network organization, in CMOS logic gates. Through electrical simulation, the

relationship between short-circuit dynamic power and charge/discharge components has been examined. Through experimental results, it has been proved that in the standard cell design, although short circuit constituent is decreasing with the advancement in the CMOS fabrication process, yet the dynamic power still acts as a major source of the power dissipation while designing. Static power, on other hand, is increasing with the advancement in the technology and becoming more critical in the VLSI physical designs.

In [19], Saini and Mehra have proposed an idea to reduce both the leakage power as well as the glitching power of the design. As the technology is shrinking, the leakage power has become the significant source of power dissipation. In the initial technology nodes, the dynamic power dominated the total power of the design, but nowadays, the leakage power is becoming one of the major concerns. An idea has been reported to alleviate leakage power as well as glitching power of lower technology nodes.

In [20], Khan *et al.*, have made an attempt to optimize the power consumption using the modular approach. Using various components, the optimization of power dissipation may be conducted. It has been proved that power usage in VLSI circuits is exclusively data dependent. Different design components exhibit conflicting effects on overall characteristics of system. However, major design issues discussed in [20] are space, speed, and power consumption.

In [21], an idea regarding breaking of clock tree root at the gate level into several pseudo clock sources has been presented. With a rapid growth of deep submicron very large scale integration circuits, many issues like power consumption and timing closure have made physical design much more difficult. In [21], an idea regarding the clock skew, that is applicable to clock tree synthesis has been addressed which helps to mitigate these issues.

In [22], an effective chip-level clock tree synthesis algorithm has been proposed to simultaneously reduce the clock divergence and multi corner skew. It is a major concern in complex system-on-chip design to synthesize effective clock tree. Balancing of the clock tree, which belong to various IPs, such that entire tree must be having smaller skew, is becoming very critical with lower technology nodes. The proposed method is expected to obtain better skews across every corner and expected to alleviate the clock divergence as well. However, 30% decrease in the clock divergence has been achieved in this and proved experimentally.

In [23], Rajaram and Pan have proposed a low power design scheme to reduce leakage power in the nanoscale CMOS VLSI design by generating an adaptive optimal reverse body-bias voltage. In this research work, the leakage monitoring circuit has been suggested to generate adaptive optimal reverse body-bias voltage. This leakage monitoring circuit compares the sub-threshold leakage current and band-to-band tunneling current. This circuit is tested on

32nm CMOS technology and the evaluations have been done in the temperature range of 25°C – 100°C. The proposed approach proved that the leakage power can be reduced significantly through body bias voltage.

In [24], the dynamic power dissipation of CMOS lower technology nodes circuit depends upon the steady state logic transition as well as glitches. In the literature, average as well as peak power dissipations are estimated by Monte-Carlo simulations, but this method is expensive. Therefore, an alternative technique has been presented in [24], in which zero delay simulation has been conducted. In this approach, the linear timing algorithm has been used to estimate steady state logic connectivity. From this information, the peak as well as average energy consumed by each vector pair can be estimated.

In [25], a new obstacle avoiding clock-tree-synthesis (CTS) technique, with slew constraints and corrected signal polarity has been proposed. As the VLSI technology is continuously scaling down towards the lower technology nodes, the CTS synthesis is becoming very much difficult. In [25], Cai *et al.*, have built a look-up table to estimate the efficient buffer delay as well as slew, which guarantees that resulting skew after the simulation must be satisfied. The CTS approach uses optimized-balanced-buffer (OBB) algorithm to meet the timing needs. The experimental results obtained in [25] showed the significant improvement in latency and skew. In this paper, the efficiency of look-up table has been proved using huge skew reduction. This OBB algorithm is an improved version of the classic balanced bipartioned algorithm.

In [26], Zhao *et al.*, explained about the principles of low power design, and then extended his research to the circuit level techniques for low power design methods of clocking systems. There are different levels to exploit the low-power design e.g., system level, architecture level, circuit level and device level. Finally, the author also discussed the low power optimization strategies at architecture as well as system level.

In [27], Salman *et al.*, have proposed the timing interdependence of setup and hold timing constraints in the static-timing-analysis (STA). The proposed idea has been divided into two parts. The first part explains sequential cells interdependence characterization, and the second part explains an efficient algorithm for STA analysis. The proposed approach minimizes the unnecessary pessimism. The results are compared and validated with proposed approach.

In [28], to attain high integration density, CMOS devices are scaled down. Due to this technology scaling, there is a drastic increase in the leakage current, which is a major contributor of design power. In [28], authors have explained different components of the leakage power and proposed different methods to reduce this. In high-performance systems leakage power contributes to a substantial portion of gross power usage, so it becomes very

difficult to work in the predefined power budget. Hence, the designers require techniques to reduce this leakage power. Hence, [28] proposes the techniques to reduce the leakage power.

In [29], Vygen has explained about the problem of slack. Meeting a slack observed in the design is very critical. Both the slacks, positive as well as negative, can affect the timing of the design. This technique details about the slack properly, and showed how to compute it along with its intended properties.

In [30], the timing analysis based on real-time flip-flops has been done. Real-time flip-flops are also known as flexible flip-flops, in which clock-pin (CP) to Q delay can change dynamically as per its setup as well as hold skews. The presented method in [30] addressed the problems related to timing, and proposed a new flexible flip-flop timing paradigm. In [30], author also explained about the scalable speedup approach to reduce the runtime with no or very little effect on the timing of the design.

In [31], the performance analysis of independent-gate FinFET's e.g., metal-insulator-gate – field-effect-transistor (MIGFET) has been analyzed using the measured data and predictions from the physics/process-based double-gate metal-oxide-semiconductor-field-effect-transistor (MOSFET) model. The MIGFET operation mode options and design are analyzed for optimizing the circuit applications. A novel design of double-balanced counterpart and signal-device RF mixer is also studied.

In [32], Aggarwal has explained about the latest development in the field of test circuits for timing checks viz. setup timing, hold timing, and access time of memory. Measurements are taken across the different voltage domain and temperature corners. Test circuits built earlier were good for the measurements in 65nm technology and older, but didn't hold good for the lower technology nodes. After undergoing numerous improvements, the proposed test circuit holds good for the lower technology nodes i.e., below 65nm technology.

In [33], two methods i.e., the construction method, and the linear programming model based method, have been proposed by Kim J.G. and Kim Y.D to do the efficient floorplanning. Floorplanning is a very important step of the physical design. However, the proper floorplanning helps to reduce the total area consumption of the chip and the timing requirement of the design. Many problems are faced while placing the blocks in the design, so as to achieve the targets. Proposed methods help to do the efficient floorplanning in the physical design implementation of the design.

In [34], a high-level leakage power reduction and analysis algorithm have been presented. For the leakage power to recharacterize a register level library, the algorithm uses device level module library. The proposed algorithm can also identify the idle modules in the data path that may be targeted for the leakage power alleviation. This leakage power can be

reduced based on the threshold voltage. This algorithm also helps to reduce and detect the leakage power consumption in the design by providing the high-level synthesis system.

In [35], Wu have presented a statistical techniques to estimate peak power dissipation in the design. This method is based on the Monte-Carlo simulation, maximum-likelihood estimation and cycle-by-cycle power consumption. It is depending on the theory of extreme order statistics. Based on input vector pairs, this method helps to predict the maximum power of the VLSI circuit. The experimental results indicate that this method typically produces the maximum power estimates.

In [36], the method has been suggested by Benini L. *et al.* for combinational circuits to reduce the glitching power by replacing the existing gates with the functional equivalent gates. By asserting a control signal, these gates level circuits can be finalized, which don't allow passing glitches through it. This proposed method has an important feature that it can be applied to employ any layout level description. It gives very predictable results with very little effect on the timing of the design.

In [37], Athas has discussed the power dissipation in battery and externally powered applications in embedded computing. This research work gives an idea about different techniques suitable for the power consumption reduction. Different circuits and architecture techniques have been presented in this paper to sustain performance level, especially when the supply voltage is reduced. Finally, stepwise charging, clock-powered logic, and reduced-swing signaling are the techniques that can outperform as energy-efficient signaling methods.

In [38], Lai *et al.*, have investigated the short circuit as well as dynamic power of CMOS gate and also proposed closed form solutions for short circuit power and output voltage for an inductor-capacitor (LC) transmission line. In [38], the ratio of short circuit power to dynamic power gets decreased for an LC transmission line. Gross power usage is focused to reduce in case of LC transmission lines because as compared to resistor-capacitor (RC) transmission lines because the inductance effect is more significant in former rather than latter.

In [39], a hinted-quad-tress (HQT) algorithm has been introduced to search for neighbors in the design. This algorithm searches neighbors directly instead of searching in the root directory. It has been proved that its performance is better than any contemporary method proposed. The design-rule-checking (DRC), its performance greatly depends on the speed of the neighbor's search. Therefore, this algorithm helps in the DRC calculation of the design.

In [40], the power consumption from different sources has been estimated. Different methods for estimating the power have been demonstrated and tested. Power consumption from off-chip driving, on-chip memories, clock distributions, and interconnections are analyzed using relevant examples. Comparisons between static as well as dynamic logic are also presented.

Comparisons are also done between the full custom design, gate array and cell library. It has been proved that power usage of all interconnects and off-chip driving are more than the power consumption of gate array.

In [41], Levitt and Abraham have presented an idea for testing the design. A method has been proposed in this paper to modify the testability by utilizing the local transformation and to analyse circuits at symbolic level. In [41], a set of the standard cell is designed by utilizing using layout level schemes. Static CMOS circuits may gives failure while working because of stuck at faults, and moreover, the test generation schemes are not enough. Therefore to check the design easily testable, a synthesis strategy has been proposed.

In [42], Kang has presented an efficient technique for computing power dissipation. For simulation, a current source and a parallel RC circuit are used. Power dissipation is one of the most significant parameters in today's deep submicron technologies. As the number of devices in VLSI is increasing, it is becoming more essential to alleviate the power consumption. This paper [42] gives an efficient technique to analyze the power consumption.

Hence, this chapter focuses on the literature survey of approximately 39 research papers related to the field of study. This chapter provides the information regarding the latest trends in the field of VLSI, and gives an idea to carry out presented research work.

CHAPTER 3

CONFIGURING THE STATIC TIMING ANALYSIS ENVIRONMENT

This chapter explains how to configure the environment for static-timing-analysis (STA). To analyze the STA results, the specification of accurate constraints is very significant. For the STA tool to identify all timing issues in designing. Moreover, design environment should be correctly and accurately specified. Configuring the STA involves the setting up clocks, specifying a multicycle path, false paths and specifying IO timing characteristics.

3.1 INTRODUCTION TO STA ENVIRONMENT

In synchronous digital design, the data computed from previous clock cycle is latched at active clock edge of the flip-flop. Figure 3.1 depicts a typical synchronous design, which is interacting with other synchronous designs. Such design-under-analysis (DUA) receives data from the clocked flip-flop, and its output is fed to another clocked flip-flop, which is external to DUA.

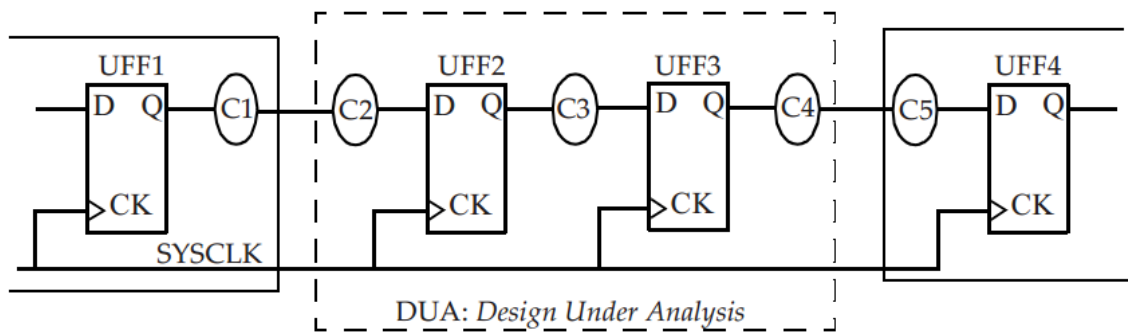


Figure 3.1 A synchronous design [2]

For static timing analysis on this design, the specification of clocks to flip-flops and the timing specification for every path is required. In above design block, there is single clock, and C1, C2, C3, C4, and C5 are the combinational blocks with C1 and C5 being outside the design, which needs to be analysed. In a typical design, there may be multiple clocks, while considering one clock domain to another.

3.2 SPECIFYING CLOCKS

To specify a clock, following information is needed

1. Clock source: It may be a port of design or any pin of a cell in the design.
2. Period: It is the time period of the clock pulse.

3. Duty cycle: It is the average time for which clock remains high or low.
4. Edges time: It is the time for rising edge or falling edge of the clock pulse.

Figure 3.2 illustrates a basic representation of a clock. By defining a clock, all the flip-flop to flip-flop timing paths are specified. It means that the flip-flop to flip-flop path must take one clock cycle and, all the internal timing paths should be analyzed with specified constraints only.

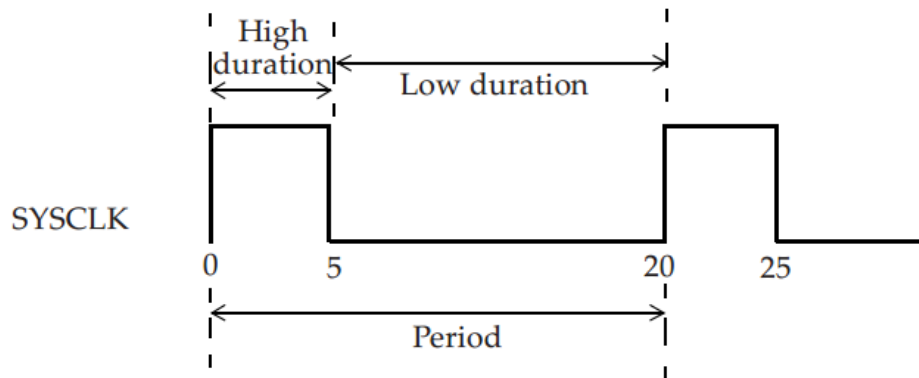


Figure 3.2 A clock representation [2]

3.3 CLOCK UNCERTAINTY

The deviation of the actual arrival time of the clock edge with respect to the idle arrival time is called as the clock uncertainty. In order to alleviate effective clock period, uncertainty may be used to change various factors like clock jitter or any other pessimism, which can be included for timing analysis. Static timing analysis of synchronous sequential circuits estimates the maximum frequency of the clock based on timing parameter of sequencing element and combinational circuits. The corresponding timing specification for flip-flop is its setup as well as clock-to-Q delay. Such specifications are approximately stated, so as to reduce their sum. Setup definition has two weaknesses. First, the definition significantly overestimates sequencing overhead in systems with typical amounts of clock uncertainty. Second, the definition is particularly pessimistic and does not uniquely define the behaviour of pulsed latches, transparent latches, and flavours of flip-flops with soft edges.

Clock uncertainty for the setup alleviates the clock duration by specified amount; and for hold, it is an additional timing margin. Figure 3.3 demonstrates the clock uncertainty for the setup and hold timing.

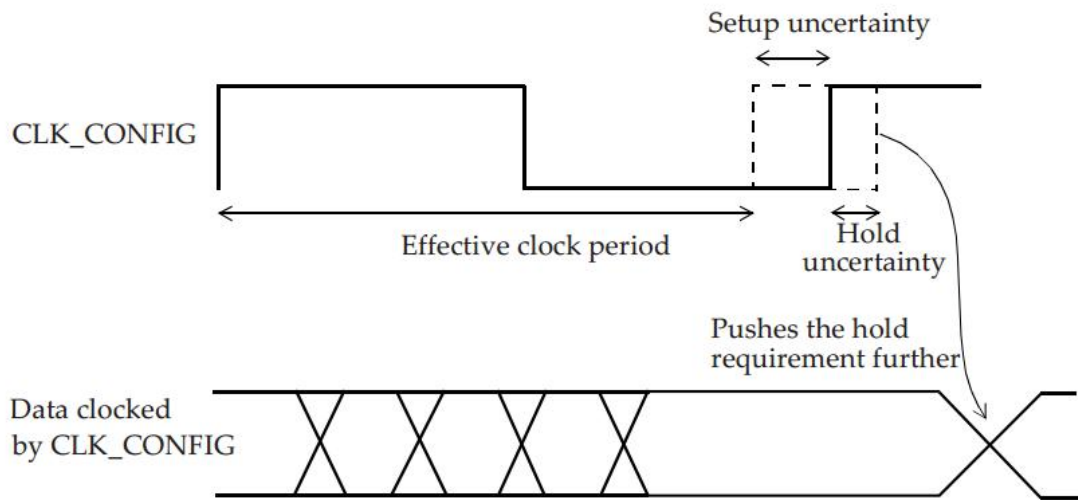


Figure 3.3 Clock uncertainty [2]

3.4 CLOCK LATENCY

The time consumed by clock to arrive at destination point from clock source point is called as clock latency. These two categories of clock latencies are

1. Network latency
2. Source latency

The delay from clock definition point to the clock pin of flip-flop, called destination point, is also known as clock network latency. The insertion delay or the source latency is the delay between clock source point and clock definition point. Source latency could either be on-chip latency or off-chip latency. Figure 3.4 and Figure 3.5 indicate on-chip and off-chip latency respectively.

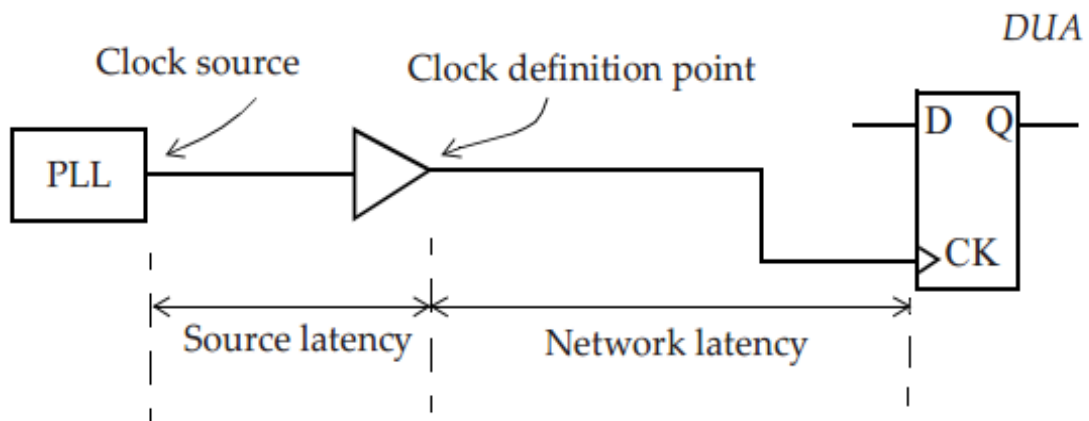


Figure 3.4 On-chip source latency [2]

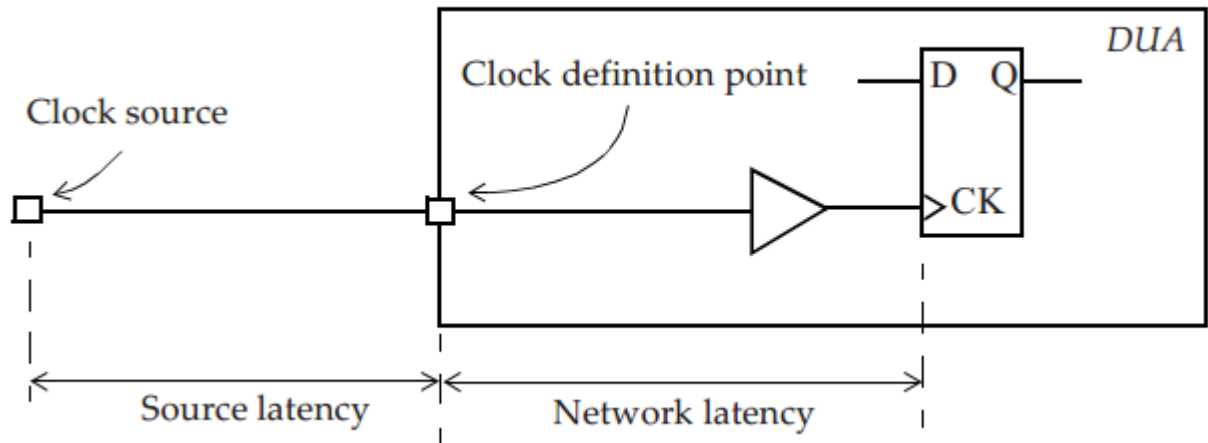


Figure 3.5 Off-chip clock source and latency [2]

Network latency is the estimate of delay of clock tree prior to the clock tree synthesis, so that once the clock tree is generated, network latency may be avoided, but the source latency exists even after clock tree is generated. After clock tree synthesis, gross clock latency is the sum of source latency plus actual delay of clock tree from the clock definition point to flip-flop [2].

3.5 GENERATED CLOCKS

Clocks can be derived from the main clock. This derived clock is called as the generated clock. A newly generated clock should be in phase with the master clock and should not generate a fresh clock domain. There is no need of updating the timing constraints for the new clock as one can define the newly generated clock as a master clock. This is one major notion of the clock origin in generated clock and master clock.

The concept of latency can also be applied on the generated clocks. There are two more points added after generating the clock, one is the generated clock definition point and the other one is the master clock definition point.

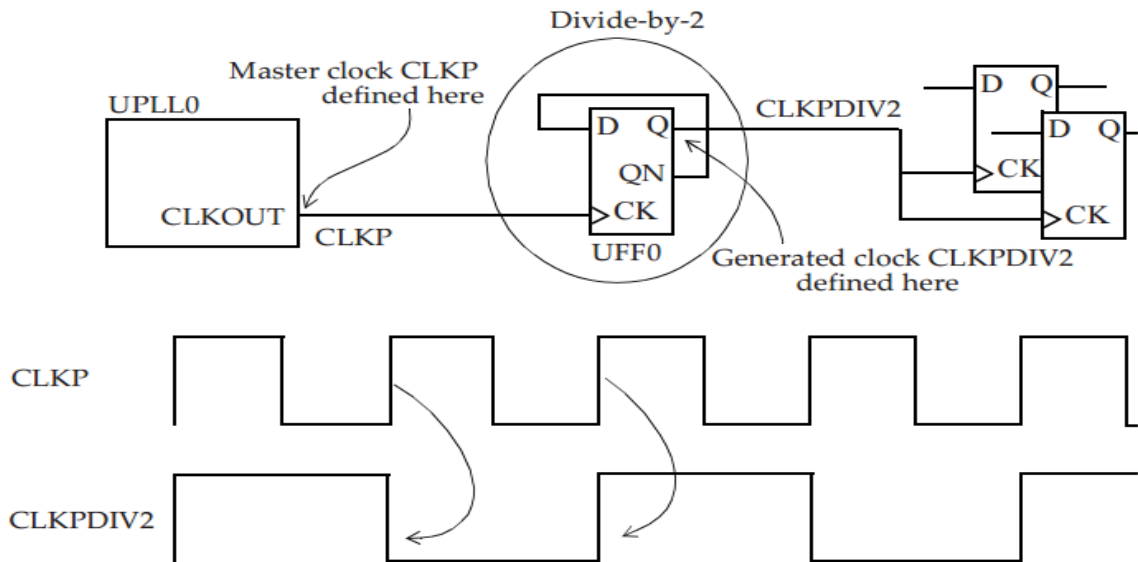


Figure 3.6 Generated clock in the design [2]

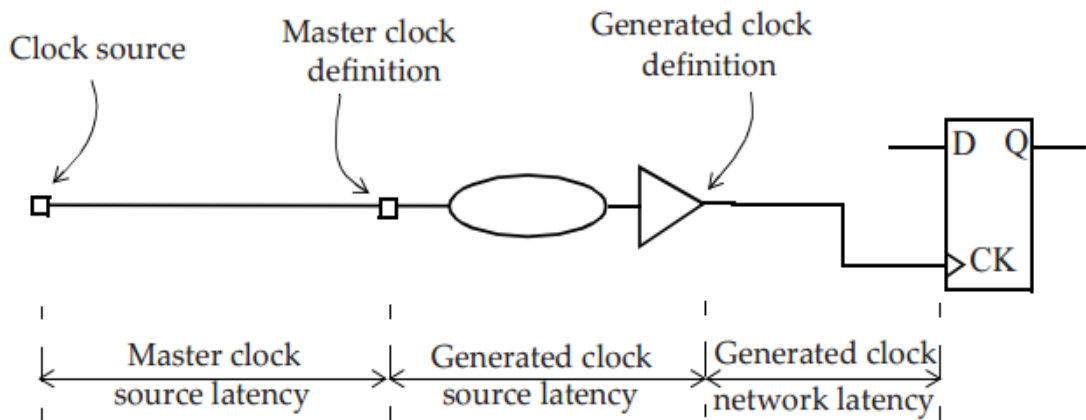


Figure 3.7 Clock latency on generated clock [2]

The latency between the master clock definition point and generated clock definition point is called as the generated clock source latency; and the timing difference between the generated clock definition point and clock pin of the flip-flop is called as generated clock network latency. The sum of source latency of master clock, the source latency of generated clock and network latency of generated clock is the total latency of network. Figure 3.7 shows the different master and generated clocks with the total latency.

3.6 TIMING PATH GROUPS

The collection of paths in given design is called timing paths. These different paths have a starting point and end point. Here, the starting points can be the input ports and clock pins of different synchronous flip-flops of the design. End points can be the data input pins and output ports. Therefore, there are different path groups present in the design and which are described as follows

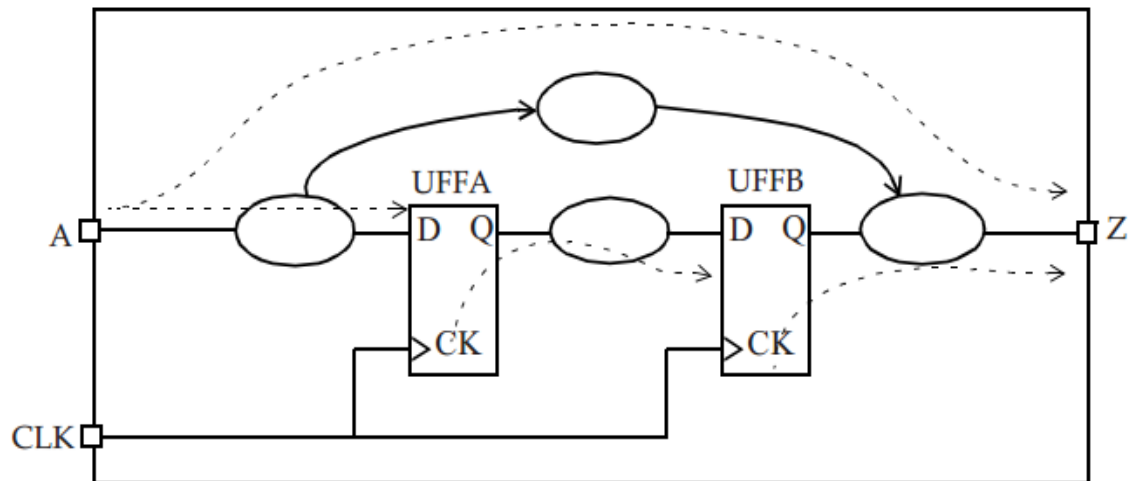


Figure 3.8 Different timing paths [2]

1. Timing path from the input port of design to the output port of design, i.e. from input port A to output port Z.
2. Timing path from the input port of design to the input port of memory device i.e., from port A to UFFA/D.
3. Timing path from the clock pin of flip-flop to data pin of flip-flop i.e., from UFFA/CLK to UFFB/D.
4. Timing path between the clock pin of port to the output port of design i.e., from UFFB/CLK to Z.

The timing paths described above are divided into timings groups. Each clock has an associated timing path associated with it and all non-clocked paths come under the default group.

The path groups in Figure 3.9 are

1. CLKA group: Path from input port A to UFFA/D.
2. CLKB group: Path from UFFA/CK to UFFB/D.
3. DEFAULT group: Input port A to output port Z, and UFFB/CK to output port Z.

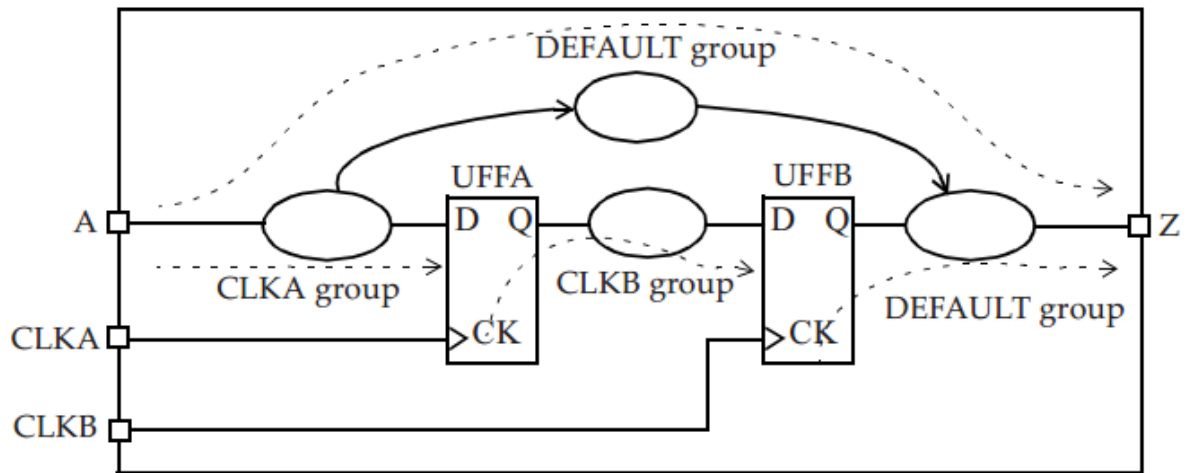


Figure 3.9 Different timing path groups [2]

3.7 CONSTRAINING INPUT/OUTPUT PATH

This section explains about the input and output paths. The STA doesn't check any path, which is not constrained. It means all the paths should be constrained to see the timings. Figure 3.10 depicts the input path of design-under-analysis (DUA). Flip-flop UFF0 provides the data to UFF1 that is internal to design under analysis. The data is connected to UFF1 via INP1, which is an input port to the DUA. The CLKA is the main clock, and the time taken before the INP1 is the external source delay. Here, the delay is calculated with respect to CLKA.

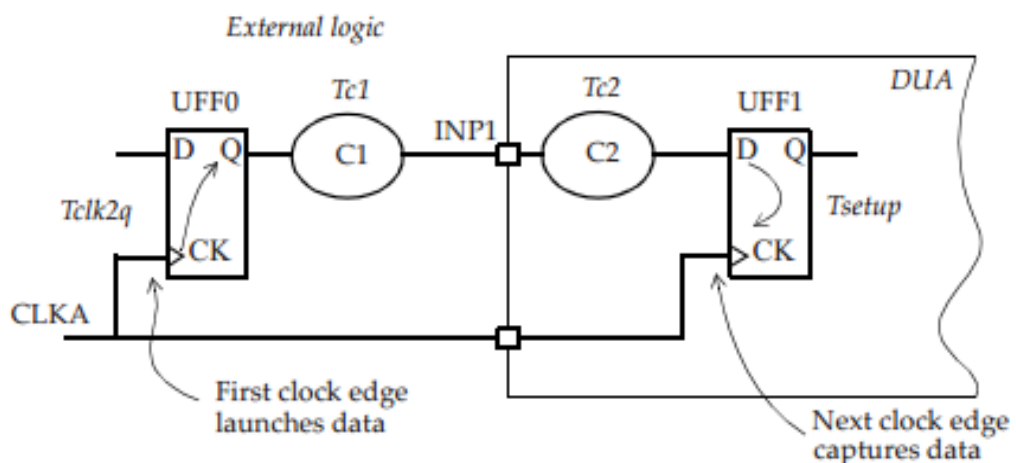


Figure 3.10 Input port timing path [2]

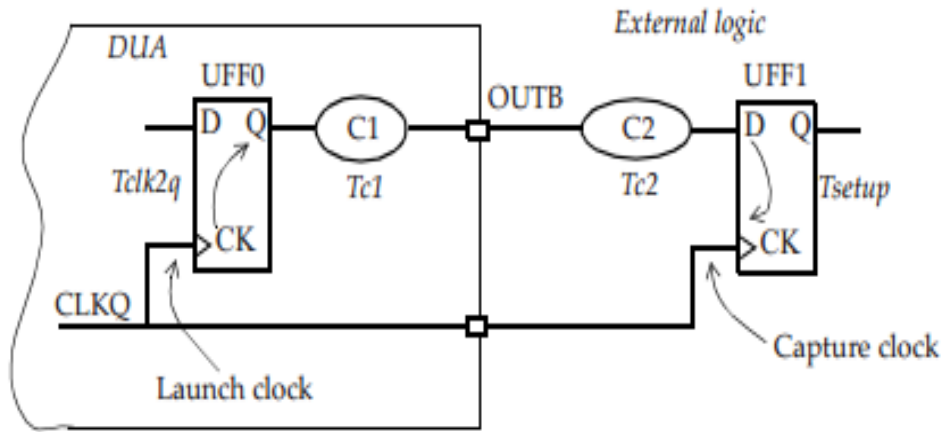


Figure 3.11 Output port timing path [2]

Figure 3.11 illustrates the output path of DUA i.e. designs under analysis. In this figure, T_{c1} and T_{c2} are combinational cells delays. The total time between UFF0 and UFF1 depends on clock period of CLKQ. The external logic of design under analysis is the sum of combinational delay T_{c2} and setup time of UFF1. Here, the output delay is calculated w.r.t. capture clock.

3.8 DESIGN-RULE-CHECK (DRC) AND CROSSTALK DELAY ANALYSIS

There are two important design rules for the static timing analysis. These are the maximum transition and maximum capacitance. Such design rules make sure that all the pins as well as ports meet specified limits for the maximum capacitance and maximum transition. As a part of STA, any violations of these parameters are reported in terms of slacks.

To extract capacitance for a typical net, the capacitance from the neighboring conductors also plays a very important role. There are different types of capacitances associated, some are grounded and others are from the traces of different signal nets. Various types of capacitances are shown in Figure 3.12. During delay calculation, all these capacitances are taken into account. There could be different cases i.e., in one case the neighboring net is steady and in other case net is switching. When the neighboring net is switching, then the timing of the net gets affected by the charging of the capacitor.

Figure 3.12 illustrates a net N1 having capacitance C_c called the coupling capacitance to a neighbouring net called the aggressor net. This example assumes different scenarios, in which the aggressor net is switching or not. The charge from driving cell in different arrangements explained below

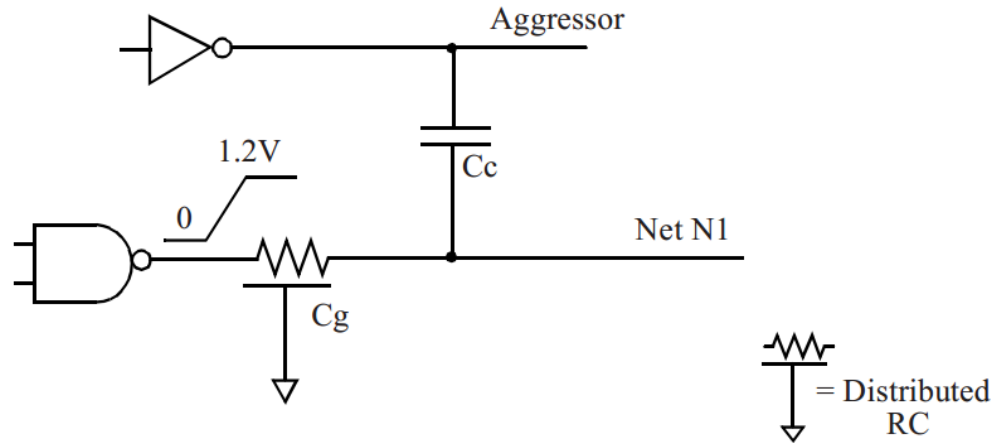


Figure 3.12 Crosstalk impact [2]

A). Aggressor Net is Steady

To charge the capacitances C_g and C_c upto V_{dd} , the driving cell provides the total charge $(C_g + C_c) * V_{dd}$. There is no crosstalk in this scenario, and the charge on the capacitors before and after the switching net is explained in the Table 3.1

Sr.No	Capacitance	Before rising transition at net N1	After rising transition at net N1
1	Grounded Capacitance, C_g	$V(C_g)=0$	$V(C_g)=V_{dd}$
2	Aggressor net steady low	$V(C_c)=0$	$V(C_c)=V_{dd}$
3	Aggressor net steady high	$V(C_g)=-V_{dd}$	$V(C_c)=0$

Table 3.1 Delay Calculation with No Crosstalk

B). Aggressor Net Switching in The Same Direction

In this case, the aggressor net is switching in same direction. If the aggressor net undergo transitions with same slew ($C_g * V_{dd}$), then it is the net charge given by driving cell. If slew is faster than net N1, then the charge required is smaller. Thus, the charge required in case of the aggressor net switching in the similar direction is lower than $C_g * V_{dd}$. Therefore, delay is smaller, and this alleviation in delay is called negative crosstalk delay.

Sr.No	Capacitance	Before rising transition at net N1	After rising transition at net N1
1	Grounded capacitance, C_g	$V(C_g)=0$	$V(C_g)=V_{dd}$
2	Coupling capacitance, C_c	$V(C_c)=0$	$V(C_c)=0$

Table 3.2 Aggressor Switching in The Same Direction Negative Crosstalk

C). Aggressor Net Switching in The Opposite Direction

In this case, the charge on coupling capacitance changes from $-V_{dd}$ to $+V_{dd}$. Therefore before and after the transition, the charge on coupling capacitance varies by $2 * C_c * V_{dd}$. Here, the additional charge is given by net N1's driver cell. Since, the delay in this case is larger, thus this delay is called positive crosstalk delay. Table given below explains the values on different capacitors.

Sr.No	Capacitance	Before rising transition at net N1	After rising transition at net N1
1	Grounded capacitance, C_g	$V(C_g)=0$	$V(C_g)=V_{dd}$
2	Coupling capacitance, C_c	$V(C_c)= -V_{dd}$	$V(C_c)=V_{dd}$

Table 3.3 Aggressor Switching in The Opposite Direction Positive Crosstalk

3.9 MULTICYCLE AND FALSE PATH

In some cases, the delay of the combinational logic is very high, and it takes more cycles to send the data. Such paths are called as a multicycle path. Data is captured at a relevant clock edge, but in the case of a multicycle path, we need to specify the edges to capture the data after a defined number of clock cycles. Figure 3.13 demonstrates a three cycle multicycle

path. There are some paths in the design, which are not real, so there is no need of checking the timing for such paths and such paths; should be disabled in the design. These paths, which are not real, are termed as a false path. The STA ignores such false paths.

The false path can be from one clock field to another, through a pin of a cell, through a pin of multiple cells, from one flip-flop's clock pin to another flip-flop. When any path designed through the pin of a cell, all such paths that pass through this pin are not taken care of while doing timing analysis. The benefit of defining false path is that underlying tool doesn't waste its time for analyzing the false paths, and it only focuses on the real paths whose timing is required.

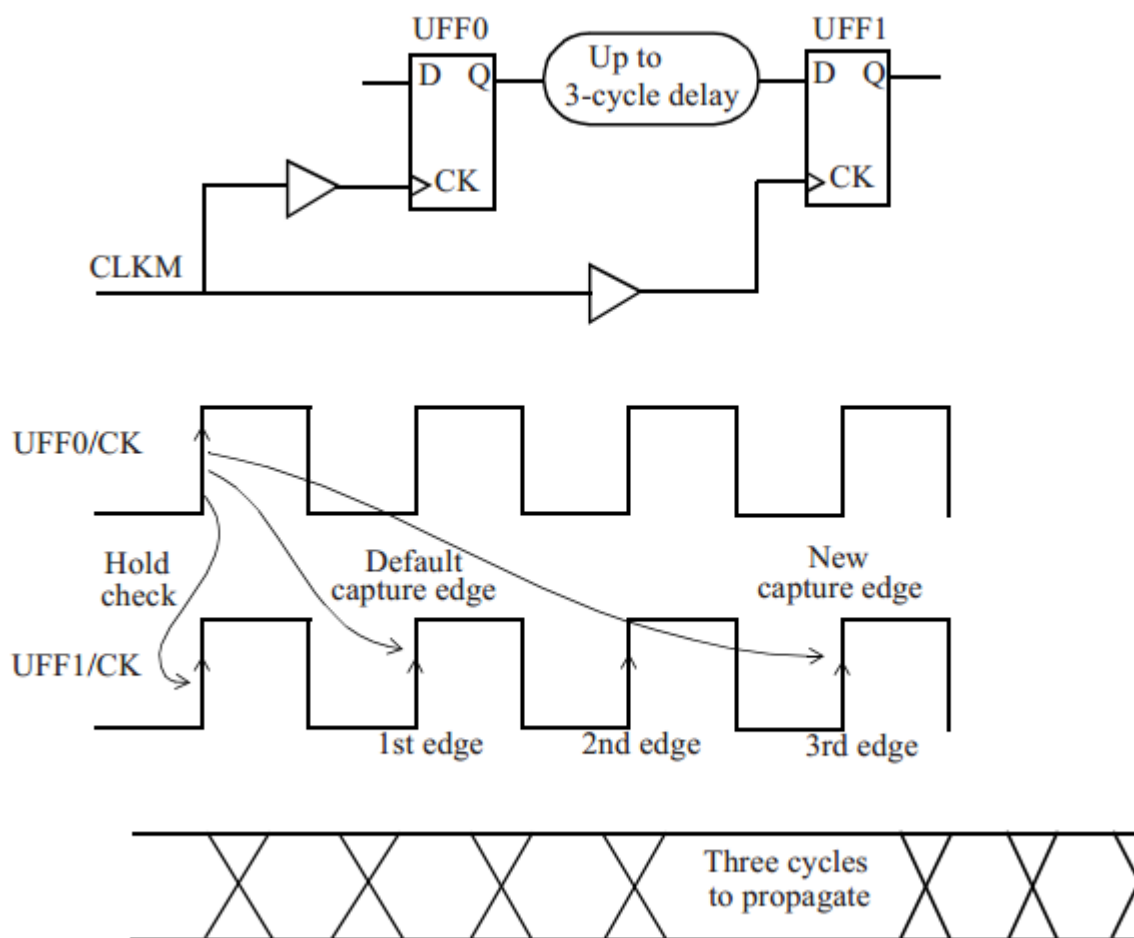


Figure 3.13 Multicycle path [2]

3.10 CHAPTER SUMMARY

Thus, this chapter provides a brief summary to configure the environment for static-timing-analysis (STA). The chapter discusses STA results with the specification of correct constraints. The important input parameters required for identification of all timing

constraints have been detailed in this chapter. Configuring for STA includes the setting up clocks, defining generated clocks, false paths and specifying IO timing characteristics.

CHAPTER 4

METHODOLOGY TO FIX TIMING AND POWER IN PHYSICAL DESIGN

In this chapter, a flow to fix the timing of the design along with leakage power reduction methodology has been proposed. During designing, there are trade-offs between the runtime, power, area and speed according to the specifications defined by a designer. Moreover, to operate at the intended clock rate, the chip must meet the timing constraints. Therefore, the timing is an important design parameter. An algorithm has been proposed in this chapter explaining the correct flow to fix the timings along with the methodology to fix the power during the design implementation.

4.1 SETUP TIMING ANALYSIS IN PHYSICAL DESIGN

Setup time is the minimum amount of time for which the data signal must be held stable prior to capture clock, so that data is captured reliably by clock. Therefore, it is the time, when the input data is stabilized and available prior to the clock edge. Figure 4.1 explains the basic architecture of the setup analysis. In Figure 4.1, there is one master clock CLKM and two paths, one is clock path and another is data path. There are two types of flip-flops, one is UFF0 called the launch flip-flop and another one is UFF1 called capture flip-flop. In order to meet the setup timing, the data path delay should be less than the clock path delay minus the setup timing of the capture flip-flop. In other words, data must be present at D-pin of capture flip-flop (UFF1) prior to its setup time.

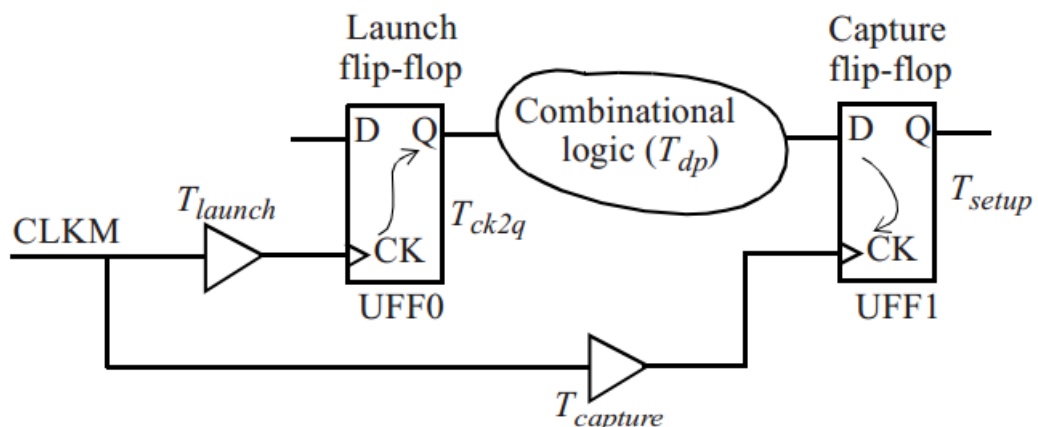


Figure 4.1 Setup analysis basic architecture [2]

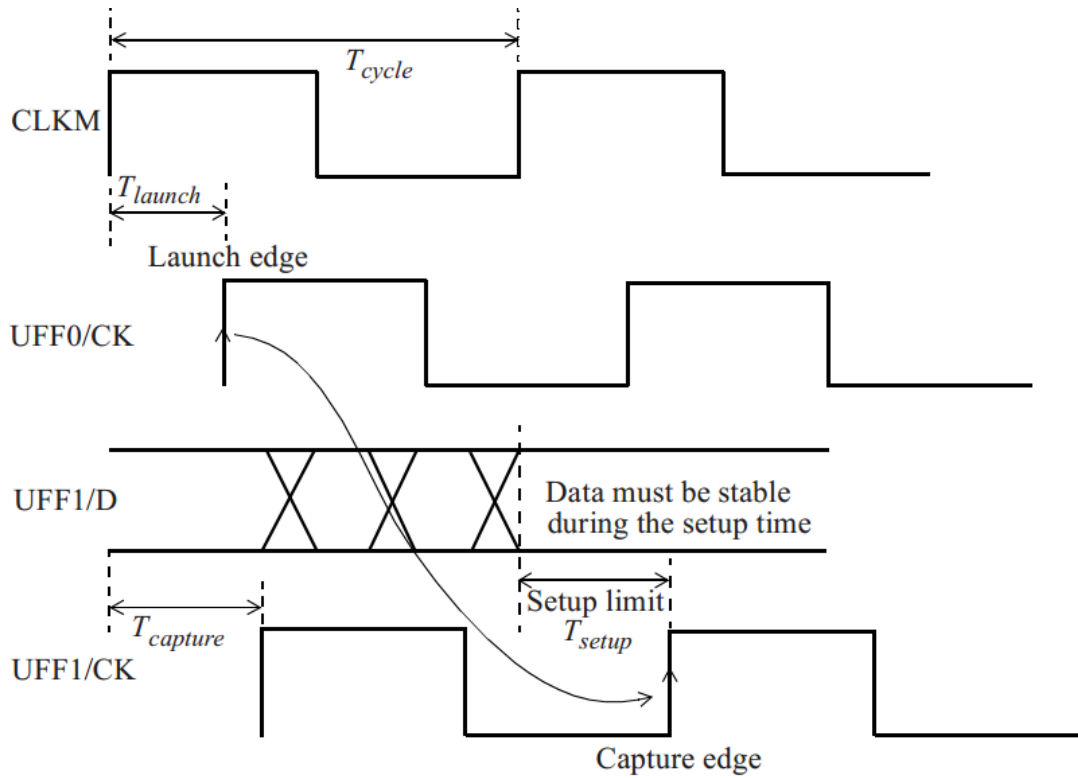


Figure 4.2 Clock and data signal for setup analysis [2]

The setup check may be analytically defined as

$$\text{Data arrival time} < \text{Data required time} \quad (4.1)$$

Here, $\text{Slack} = \text{Data arrival time} - \text{Data required time}$
and it should be positive.

$$T_{data\ path} < T_{clock\ path} - T_{setup} \quad (4.2)$$

where, $T_{data\ path}$ = Total time of the data path.

$T_{clock\ path}$ = Total time of the clock path.

T_{setup} = Setup time of capture flip-flop.

$$T_{data\ path} = T_{launch} + T_{ck2q} + T_{dp} \quad (4.3)$$

$$T_{clock\ path} = T_{cycle} + T_{capture} \quad (4.4)$$

$$\text{Therefore, } T_{launch} + T_{ck2q} + T_{dp} < T_{cycle} + T_{capture} - T_{setup} \quad (4.5)$$

where, T_{launch} is the delay of clock tree of launch flip-flop UFF0, T_{dp} is the delay of combinational logic data path, T_{ck2q} is the delay of clockpin to Q-pin of launching flip-flop, and T_{cycle} is the clock duration. $T_{capture}$ is the delay of clock tree for the capture flip-flop UFF1.

4.2 HOLD TIMING ANALYSIS IN PHYSICAL DESIGN

Hold timing is also an important timing parameter. After every clock pulse, the data should be held steady for some minimum amount of time, this minimum amount of time is called as the hold time. Figure 4.3 explains about the basic architecture of hold analysis. In Figure 4.3, there is one master clock CLKM and two paths, one is the clock path and another is the data path. There are two types of flip-flops, one is UFF0 called the launch flip-flop as well as another one is UFF1 called the capture flip-flop. In order to meet the hold timings, the data path delay should be greater than the clock path delay.

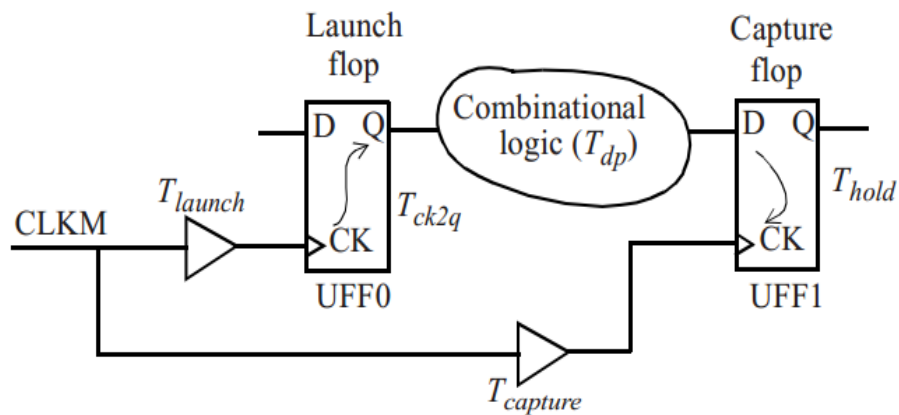


Figure 4.3 Hold analysis basic architecture [2]

The hold check may be analytically defined as

$$\text{Data arrival time} > \text{Data required time} \quad (4.6)$$

Here, Slack = Data arrival time – Data required time,
and it should be positive.

$$T_{data\ path} > T_{capture} + T_{hold} \quad (4.7)$$

where, $T_{data\ path}$ = Total time of the data path

$T_{clock\ path}$ = Total time of the clock path

T_{hold} = Hold time of capture flip-flop

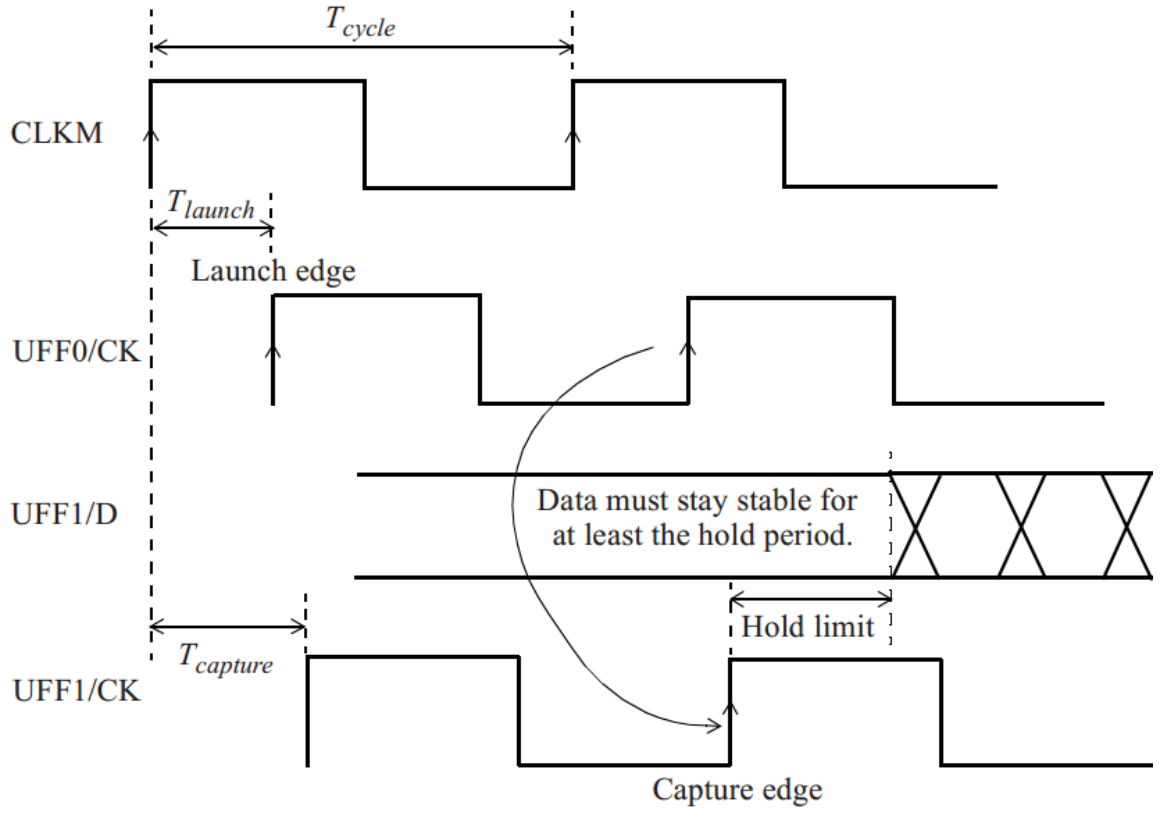


Figure 4.4 Clock and data signal for hold analysis [2]

$$T_{data\ path} = T_{launch} + T_{ck2q} + T_{dp} \quad (4.8)$$

$$T_{clock\ path} = T_{hold} + T_{capture} \quad (4.9)$$

$$\text{Therefore, } T_{launch} + T_{ck2q} + T_{dp} > T_{capture} + T_{hold} \quad (4.10)$$

where, T_{launch} is the delay of clock tree of launch flip-flop UFF0, T_{dp} is the delay of combinational logic data path, and T_{ck2q} is the delay of clockpin to Q-pin of launching flip-flop. $T_{capture}$ is the delay of clock tree for capture flip-flop UFF1.

4.3 SETUP AND HOLD TIMING ANALYSIS TIMING REPORT

Analysis of the timing of the design is required after the physical implementation of the block. The static-time-analysis (STA) tool will analyze the timings of the circuit. Following is the timing report generated by STA tool for the setup and hold analysis. The timing report for setup is showing a positive slack, which means there is no setup violation in the given path. Start point, end point, path group are also there in the report along with the different parameters [2].

Start point: UFF0 (rising edge-triggered flip-flop clocked by CLKM)

Endpoint: UFF1 (rising edge-triggered flip-flop clocked by CLKM)

Path Group: CLKM

Path Type: max

Point	Increment	Path
clock CLKM (rise edge)	0.00	0.00
clock network delay (ideal)	0.00	0.00
UFF0/CK	0.00	0.00 r
UFF0/Q (DFF)	0.16	0.16 f
UNOR0/ZN (NR2)	0.04	0.20 r
UBUF4/Z (BUFF)	0.05	0.25 r
UFF1/D (DFF)	0.00	0.25 r
clock CLKM (rise edge)	10.00	10.00
clock network delay (ideal)	0.00	10.00
clock uncertainty	-0.30	9.70 r
library setup time	-0.04	9.66
data required time		9.66
data arrival time		-0.25
slack (MET)		9.41

For hold timing analysis below report is dumped by the STA tool

Start point: UFF0 (rising edge-triggered flip-flop clocked by CLKM)

Endpoint: UFF1 (rising edge-triggered flip-flop clocked by CLKM)

Path Group: CLKM

Path Type: min

Point	Increment	Path
clock CLKM (rise edge)	0.00	0.00
clock source latency	0.00	0.00
CLKM (in)	0.00	0.00
UCKBUF0/C (CKB)	0.06	0.06 r
UCKBUF1/C (CKB)	0.06	0.12 f
UFF0/CK (DFF)	0.00	0.12 r
UFF0/Q (DFF)	0.14	0.26 r
UNOR0/ZN (NR2)	0.02	0.28 r
UBUF4/Z (BUFF)	0.06	0.34
UFF1/D (DFF)	0.00	0.34
clock CLKM (rise edge)	0.00	0.00 r
clock source latency	0.00	0.00 r
CLKM (in)	0.00	0.00 r
UCKBUF0/C (CKB)	0.06	0.06
UCKBUF2/C (CKB)	0.07	0.13
UFF1/CK (DFF)	0.00	0.13
clock uncertainty	0.05	0.18
library hold time	0.01	0.19
data required time	0.01	0.19
Data arrival time		-0.33
Slack(MET)		0.14

Thus data of the above timing can be analyzed using the above two reports. Both the reports discuss that the slack is met or not met.

4.4 DYNAMIC AND STATIC POWER DISSIPATION IN PHYSICAL DESIGN

Today, power consumption is a major factor/issue in all technologies. Products normally run on batteries in portable applications. However, battery runs down and requires replacement or recharging. Product designers have been working on increasing battery life, and key for this is to design low power IC. In this topic, fundamental theory behind different sources of power dissipation in a chip shall be explored.

The instantaneous power supplied by a circuit element is the product of voltage across circuit element and current passing through it.

Therefore,

$$\overline{P(t)} = \overline{I(t)} \overline{V(t)} \quad (4.11)$$

Over some time interval T, the energy consumed is the integral sum of the instantaneous power, and it is given by:

$$\overline{E} = \int_0^T \overline{P(t)} dt \quad (4.12)$$

Over this interval T, the average power can be written a

$$\overline{P_{avg}} = \frac{\overline{E}}{T} = \frac{1}{T} \int_0^T \overline{P(t)} dt \quad (4.13)$$

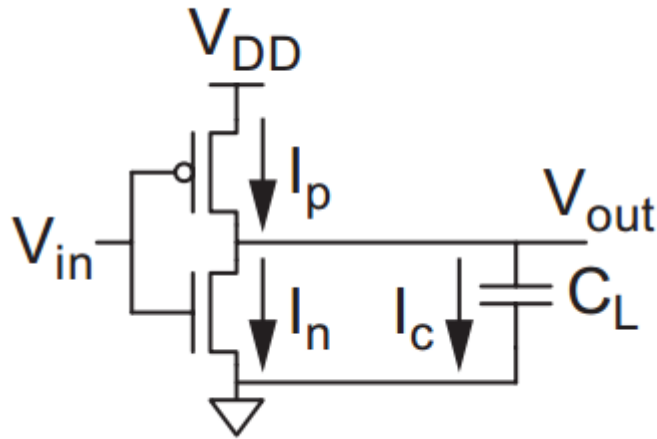


Figure 4.5 CMOS inverter [3]

Figure 4.5 demonstrates a load capacitor, when a capacitor is charged from 0 to V_c , it stores energy \overline{E}_c , and releases energy when discharges back to 0, which is given by

$$\overline{E}_c = \int_0^\infty \overline{I(t)} \overline{V(t)} dt = \int_0^\infty C \frac{dv}{dt} \overline{V(t)} dt = C \int_0^{V_c} \overline{V(t)} dv = \frac{1}{2} C V_c^2 \quad (4.14)$$

In CMOS inverter as shown in Figure 4.5, when input switches from 0 to 1, PMOS transistor turns on and changes load to V_{dd} [3]. According to Equation (4.14), the energy stored in capacitor is

$$\overline{E_c} = \frac{1}{2} C_l V_{dd}^2 \quad (4.15)$$

$$\overline{E_c} = \int_0^\infty \overline{I(t)} V_{dd} dt = \int_0^\infty C \frac{dv}{dt} V_{dd} dt = C V_{dd} \int_0^{V_{dd}} dv = C V_{dd}^2 \quad (4.16)$$

By observation, it has been found that half of the energy is dissipated as heat and other half is stored in the capacitor. When the input changes from 0 to 1, NMOS is turned on and PMOS switches off. The energy stored is dissipated in NMOS transistor. During this transition, no energy is drawn from main power source.

Figure 4.6 shows the different waveforms of the inverter. The PMOS transistor begins to turn ON when V_{in} begins to fall. Initially, it is saturated, and I_p ramps up as V_{in} falls, it levels out at I_{dsat} . The V_{out} rises to the point such that PMOS shifts to linear region, with this I_p tapers out exponentially. The PMOS starts to turn off with the rise in V_{in} . There is small blip of current during this transition and this current is called as the short-circuit current. Half of the energy is delivered to the capacitor and other half is dissipated.

Suppose $\overline{f_{sw}}$ is the average switching frequency of the gate. Load will be charged and discharged $\overline{f_{sw}} T$ times over the time interval T . Therefore, the average power dissipation according to Equation (4.13) is,

$$\overline{P_s} = \frac{E}{T} = \frac{T \overline{f_{sw}} C v_{dd}^2}{T} = C v_{dd}^2 \overline{f_{sw}} \quad (4.17)$$

It is known as dynamic power, as it arises from the switching of load of CMOS inverter. Since, most of the gates don't switch during each clock cycle; this is quiet convenient to express the switching frequency $\overline{f_{sw}}$ as $\bar{\alpha}$ times the clock frequency f which is called as the switching factor. Therefore equation for the dynamic power dissipation can be written as

$$\overline{P_s} = \bar{\alpha} C v_{dd}^2 f \quad (4.18)$$

Here, the probability that circuit node changes from 0 to 1 is called switching activity factor, because that is the only time the circuit consumes power.

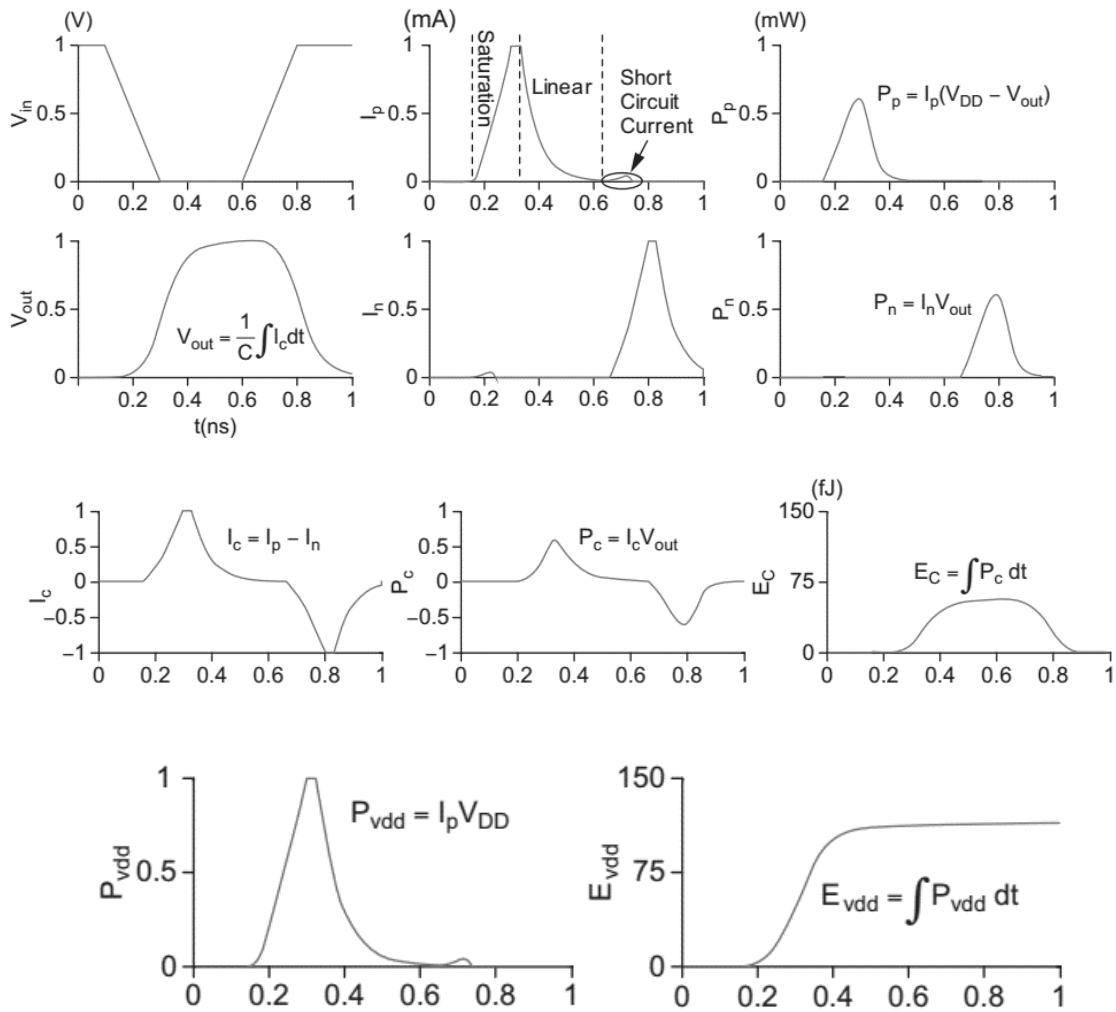


Figure 4.6 Inverter switching voltage, energy, power and current waveforms [3]

4.4.1. Sources of Power Dissipation

The major components in power dissipation are

1. Dynamic power dissipation: It is because of charging and discharging, as gates switch and short-circuit current, while both NMOS and PMOS stacks are partially ON [3].
2. Static power dissipation: It is because of sub-threshold leakage through OFF transistors, gate leakage through gate dielectric, the junction leakage from source/drain diffusions, and junction leakage through gate dielectric.

The total power dissipation in the circuit can be written as

$$P_{dynamic} = P_{short\ circuit} + P_{switching} \quad (4.19)$$

$$P_{static} = (I_{gate} + I_{sub} + I_{junc}) V_{dd} \quad (4.20)$$

$$P_{total} = P_{dynamic} + P_{static} \quad (4.21)$$

4.4.2. Dynamic Power Analysis and Its Optimization by Cell Replacing/Swapping

Dynamic power consists of two major components i.e., one is short circuit power and another one is switching power, given in Equation (4.18). To estimate the switching power in the design, the supply voltage v_{dd} and frequency f is known to the designer, the capacitance C is addition of gate, wire, and diffusion capacitance of node. An activity factor can be estimated by logic simulation. The CAD tools perform an efficient work while estimating the power. The other major component of dynamic power dissipation is because of short circuit.

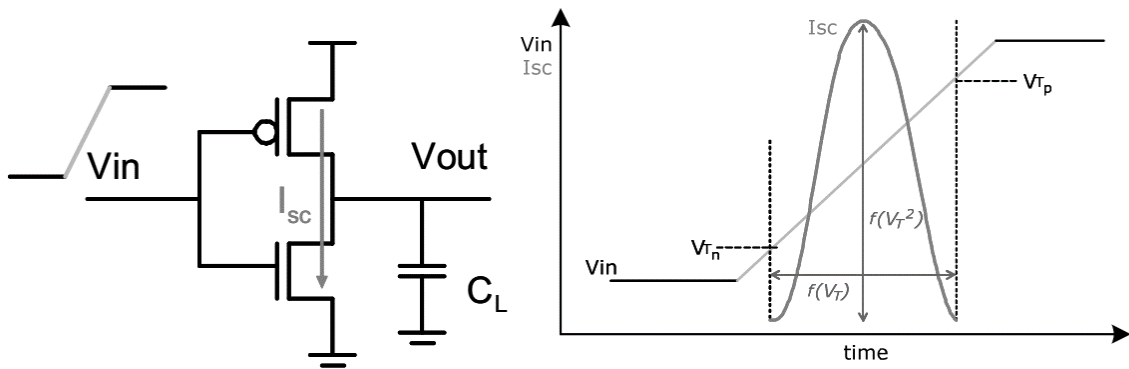


Figure 4.7 Short circuit current in CMOS inverter [18]

During transition at input of the inverter from 0 to 1, there will be timing window, in which both transistors PMOS as well as NMOS shall conduct, causing the current to flow from supply to the ground, this current is called as the short circuit current. This current flow at the time when the threshold voltage of NMOS transistor is less than input voltage; and threshold voltage of PMOS transistor is more than input voltage. This duration and magnitude of short-circuit current depend heavily on the slope of input signal. Figure 4.7 depicts the short circuit current in the inverter circuit.

Short circuit power is ignored in many scenarios during power analysis or during development of the power optimization methods, while in a few scenarios, it is considered to be some fraction of total power [16].

As shown in Figure 4.7 that during the transition of the input signals, the short circuit power is dissipated and it is dependent on the input slope duration as well as current magnitude. A paradigm for SC current estimation suggested by Veendrick has been extensively utilized as a handy paradigm of short circuit power consumption [16]; it follows that

$$P_{short\ circuit} = \frac{\bar{\beta}}{12} (V_{dd} - 2V_t)^3 \frac{\bar{\tau}}{T} \quad (4.22)$$

Here, $\bar{\beta}$ is gain factor, $\bar{\tau}$ is input rise/fall time of inverter, and T is time period of input signal. The cubic short circuit current dependency is shown in the Figure 4.7. It is apparent that short- circuit current is linearly dependent on threshold voltage.

$$\bar{\tau} = \bar{\alpha} R C \quad (4.23)$$

where, C is the input capacitance of the power measured cell, R is the effective resistance of the driver, $\bar{\alpha}$ is rise and fall time of voltage swing.

The relationship between threshold voltage and supply voltage can be written as

$$V_t = y V_{dd} \quad (4.24)$$

where, y is the ratio of V_t and V_{dd} in a given process.

By using Equation (4.23) and Equation (4.24), we may represent the dependency of short circuit power on threshold voltage

$$P_{short\ circuit} = \frac{\bar{\beta}}{12} (V_{dd} - 2V_t)^3 \frac{\bar{\tau}}{T} = \frac{\bar{\beta}}{12} ((1 - 2y)V_{dd})^3 \frac{\bar{\alpha} R C}{T} \quad (4.25)$$

$$P_{short\ circuit} = \frac{\bar{\beta}}{12} (1 - 2y)^3 V_{dd}^3 \bar{\alpha} R C f \quad (4.26)$$

The dynamic power consumption is defined as

$$P_{dynamic} = C v_{dd}^2 f \quad (4.27)$$

$$\frac{P_{short\ circuit}}{P_{dynamic}} = \frac{\bar{\beta}}{12} (1 - 2y)^3 V_{dd} \bar{\alpha} R \quad (4.28)$$

From Equation (4.28), it has been proved that there is cubic dependency of

$P_{short\ circuit} / P_{dynamic}$ on y i.e. V_t/V_{dd} .

$$\varphi = \frac{P_{short\ circuit}}{P_{dynamic}} = \frac{\bar{\beta}}{12} (1 - 2y)^3 V_{dd} \bar{\alpha} R \quad (4.29)$$

$$\varphi = \frac{P_{short\ circuit}}{P_{dynamic}} = \frac{I_{d\ sat}}{12 (V_{dd} - 2V_t)^2} (1 - 2y)^3 V_{dd} \bar{\alpha} \frac{V_{dd}}{I_{d\ sat}} = \frac{(1-2y)^3}{12.(1-y)^2} \bar{\alpha} \quad (4.30)$$

Ratio of the original with high- threshold devices

$$\frac{\varphi_{hvt}}{\varphi_0} = \frac{(1-2y_{hvt})^3}{12.(1-y_{hvt})^2} \bar{\alpha} \bigg/ \frac{(1-2y_0)^3}{12.(1-y_0)^2} \bar{\alpha} \quad (4.31)$$

$$\frac{\varphi_{hvt}}{\varphi_0} = \frac{(1-y_0)^2}{(1-y_{hvt})^2} \bigg/ \frac{(1-2y_0)^3}{(1-2y_{hvt})^3} \quad (4.32)$$

Here, 80% of reduction of the short circuit power can be achieved by setting the range $0.1 < y_0 < 0.3$ and $0.1 < y_{hvt} < 0.3$. Timing deterioration will be more significant, and component replacement needs to be performed carefully to avoid the problems of timings on critical path.

4.5 LEAKAGE POWER ANALYSIS AND ITS OPTIMIZATION BY CELL REPLACING/SWAPPING

VLSI industry is found to follow on Moore's law, according to which, on an integrated circuit, number of transistors per square inch had doubled every year; and it will continue in future as well. Due to increasing number of transistors, the expensive cooling and packaging material is required. Therefore to reduce the switching power, the supply voltage has been scaled down; and for performance trade off, the threshold voltage has also been scaled down. But due to scaling down of threshold voltage, there is an exponential increase in the leakage current causing the leakage power dissipation in the design. The main cause of static power dissipation is leakage current flowing in any circuit, when the device is active or in idle mode. There are different reasons of leakage current, but the major source of leakage current is sub-threshold leakage current.

$$P_{leakage} = \overline{I_{sub}} \cdot V_{dd} \quad (4.33)$$

Here, $P_{leakage}$ denotes leakage power and $\overline{I_{sub}}$ is the subthreshold leakage current. The other sources of the leakage current have not been taken into account for this research work, as the threshold current is dominating among all the other components.

The current-voltage (IV) model assumes that in long channel transistor, the current only flows from source to drain, when $V_{gs} > V_t$. In real transistors, the current doesn't abruptly cut off below threshold, but rather it drops off exponentially. When gate voltage is high, the transistor is ON, when V_{gs} falls below V_t , the current appears as straight line on logarithm

scale. This regime of $V_{gs} < V_t$ is called as the weak inversion and because of the drain-induced barrier lowering, the threshold leakage current increases significantly.

Leakage power constitutes a major portion of gross power of the device circuit. If no solution is found out then, it has been projected in the literature that by the end of the year 2020, the leakage power is expected to increase 32 times per device [17]. The sub-threshold leakage current is dominating component in total leakage power dissipation.

Sub-threshold current may be represented by following Equation [17]

$$\overline{I_{sub}} = I_0 e^{\frac{V_{gs} - V_{th0} - \eta V_{ds} - \gamma V_{sub}}{\eta V_\theta}} \left(1 - e^{-\frac{V_{ds}}{V_\theta}}\right) \quad (4.34)$$

$$\overline{I_0} = \mu C_{ox} \frac{W}{L} V_\theta^2 e^{1.8} \quad (4.35)$$

The sub-threshold depends on transistor parameters according to following table

Sr.No	Device Parameter	Subthreshold dependency
1	Transistor Length (L)	Inversely Proportional
2	Transistor Width (W)	Directly Proportional
3	Temperature (T) and Input Voltage (V_{gs})	Increases Exponentially
4	Threshold Voltage (V_{th0})	Increases by an order of magnitude of 100 mV decrease.

Table 4.1 Dependence of Sub-threshold Leakage on Device Parameters [17]

4.5.1 Different Leakage Power Reduction Techniques

In this section, different techniques for leakage reduction have been presented. Battery operated devices in idle mode generally suffer from the leakage power dissipation. The cellular phones remain idle most of the time, and since these devices are not turned off, the precious battery power is drained. This decreases life of the battery. Existing devices are getting modified in such a way that the leakage dissipation is very less. Battery operated devices are either in active mode or in standby mode. The leakage power reduction techniques [17] are given below.

A). Power Gating and Multi-Threshold Cells

Multi-threshold CMOS is a very powerful technique to decrease the leakage power. In this technique with power supply, ground, and existing design, a high threshold transistor is

inserted. During normal operation, the standard threshold transistors are turned ON and during inactive mode, high threshold transistors are turned ON and other gets OFF. Hence both standard and high threshold voltage are fabricated on the same chip. In this technique, high threshold voltage transistors act as power gating to standard transistors, therefore this method is also called power gating. Figure 4.8 illustrates the basic structure of a multi-threshold complementary metal oxide semiconductor.

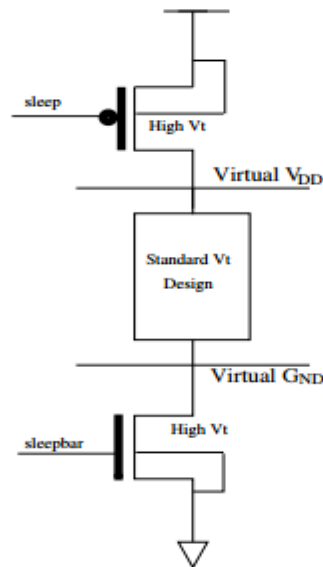


Figure 4.8 Multi-threshold CMOS structure [17]

B). *Dual-Threshold-CMOS-Transistors (DTMOS)*

This technique uses two types of the transistors, one is a high threshold and another one is low threshold transistors. In this technique, based on the path, the transistors are used. There are two types of paths in any design, one is the critical and another one is non-critical. Critical path uses the low threshold voltage transistors; and the paths, which are non-critical, use the high threshold voltage transistors. This approach works on the algorithm of finding the accurate place for inserting the high and low voltage transistors.

C). *Variable-Threshold-CMOS-Transistors (VTCMOS)*

This is a technique for the leakage power reduction during the active mode. In this scheme, threshold is increased during standby mode by connecting the voltage of substrate either higher than ground for P transistors or lower than the ground for N transistors. The significant limitation of this method is a requirement of the supplementary power supply that may not be the best in some industrial designs.

D). Proposed Leakage Power Reduction by Cell Swapping/Replacing

Application specific integrated circuits with digital logic features use the standard cell methodology for designing in the domain of semiconductor design. A standard cell is nothing, but a collection of interconnect structures and transistors, which provide a Boolean logic function e.g., AND, XOR, NOT, XNOR or storage function flip-flop and latches.

Usually, initial design of the standard cell is created in the form of netlist or schematic view. It consists of logical connectivity of all the cells, known as standard cells and macros. It is composed of a list of nets as well. A number of different electronic design automation techniques are available to generate the schematic view that gives a graphical-user- interface (GUI) for netlist generation process.

The netlist provided by the front end team consists of different threshold voltage cells. These cells or transistors can either be high threshold voltage and low threshold voltage transistors or cells. Earlier in the deep submicron technology, the LVT was used; but lately, ULVT has also been introduced. The netlist provided by the front end team in this block implementation has ULVT (ultra-low voltage threshold transistor) and LVT (Low voltage threshold). Now by Equation (4.37),

$$\overline{I_{sub}} \propto W \quad (4.36)$$

where, W is the width of the transistor.

$$\overline{I_{sub}} \propto e^{\frac{V_{gs} - V_{th0} - \eta V_{ds} - \gamma V_{sub}}{\eta V_{\theta}}} \quad (4.37)$$

$$\overline{I_{sub}} \propto (1 - e^{-\frac{V_{ds}}{V_{\theta}}}) \quad (4.38)$$

With a decrease in threshold voltage (V_{th0}), the sub threshold leakage current $\overline{I_{sub}}$ increases by an order of magnitude of 100 mV decrease. It means that the leakage power increases with a decrease in the threshold voltage i.e., ULVT cells have more leakage in comparison to the LVT cells. Although the delay of the ULVT cells is very less, which is very helpful in meeting the timing requirements in the design, yet on the other hand it dissipates more leakage power, which has adverse effect on the product yield. In this approach, an idea to reduce this leakage power has been proposed. In this method, all the high leakage cells (that is HVT cells having lesser delay have been) replaced with the LVT called, as the low leakage cells exhibit longer delay in comparison to the HVT cells. This cell swapping should be done before the placement of standard cells in the core area of the block, and after importing the netlist in the design. All the timing calculations would be done with these LVT cells, and for

any path, which is very critical to meet the timing requirements with this cell swapping, it would be fixed later in the timing fix tool by inserting the ULVT cells in place of LVT on need basis. By this approach, a substantial decrease in the leakage power has been observed, and corresponding results are presented in the next chapter.

4.6 POWER DISSIPATION AND JUNCTION TEMPERATURE

With the dissipation of power, the junction temperature of chip increases. This increase in temperature affects device both in on state and off state. When the device is off, the rise in temperature results in an increase in a number of intrinsic carriers by the following expression

$$n_i \propto e^{\frac{E_g}{kT}} \quad (4.39)$$

From this equation, it can be inferred that as the temperature rises, the number of intrinsic carriers increases in semiconductor. Majority carriers are less affected by an enhancement in temperature; therefore the device appears to be more intrinsic. The leakage current that directly depends on the intrinsic carrier concentration further elevates the temperature due to power dissipation. Ultimately, the device might break down due to increasing temperature.

An ON device will not be affected much with an enhancement in minority carrier concentration, but it depends on the mobility and threshold voltage of device, which leads to change in current.

Heat sinks can be used to take away the heat generated by the power dissipation. A heat sink has lower thermal resistance than the package, and hence draws heat from it. For heat to be removed efficiently, the rate of heat transfer from the area of heat generation to ambient must be more than the rate of heat generation. This rate of heat transfer is dependent on thermal resistance.

The thermal resistance θ is expressed as

$$\theta = \frac{l}{\sigma_c A} \quad (4.40)$$

where, l is the length, A is the area and σ_c is the thermal conductivity of heat sink.

where, θ can also be given by using the following expression,

$$\theta = \frac{\Delta T}{\Delta P} \quad (4.41)$$

Using this equation, we can state that for a given power dissipation P_D ,

$$\theta < \frac{T_j - T_a}{P_D} \quad (4.42)$$

where, T_j is the junction temperature and T_a is the ambient temperature. Heat sink materials are usually coated black to radiate high energy.

4.7 PROPOSED PROCEDURAL ALGORITHM DEFINING OPTIMUM FLOW IN PHYSICAL DESIGN TO FIX TIMING AND POWER VIOLATIONS

Continuous scaling of CMOS supports very large scale integration circuits with very high speed. Modern circuits are working on 1 GHz frequency. As the clock period is continuously decreasing, the pessimism imposed by the timing verification tool is least acceptable. Therefore, quite efficient verification as well as characterization methods may be required. To analyze the timing verification of design, the static-timing-analysis tool is required. To analyze circuit, the STA tool relies on the data provided by the cell libraries. The constraints should be very accurate and updated in order to analyze the timing in the analysis tool. An inaccurate value of the timing constraints causes STA tool results to be more pessimistic or optimistic. Both optimistic case and the pessimistic case are not desirable as the optimistic case can lead to chip failure and the pessimistic case degrades design speed.

Setup timing, hold timing, signal EM, cell EM, transition and capacitance violations are the timing violations shown by the STA tool. These timing violations need to be fixed in order to get reliable working of the design at a desired frequency. If these violations are not fixed, it could lead to the design failure. Along with timing fixing, the power fixing is also required. Leakage power is becoming one of the major causes for the device yield reduction.

Fixing of the timing and power are very important. But at which stage of the design, these fixes should be implemented, that is even more important to get the accurate timing results and for the chip to be working with the lower power. An algorithm has been proposed explaining the flow to fix these violations. The proposed algorithm is described in ten steps. Figure 4.9 details about the algorithm explaining the appropriate flow to implement the timing and power fixing.

- Step 1) In step one, the design is in the PNR tool, and the physical implementation of the block is to be completed with this tool. Once the design is ready, the timing and power checks need to be done.
- Step 2) In step two, take the database into the STA and power measurement tool for the timings and power analysis respectively.

Step 3) If timing is fixed, then move the database for fixing the power; but if the timing is not fixed, then run the STA tool to fix the timing violations.

Step 4) In step four, set the higher driving cells false, so that higher driving strength cells should not be used in the design for example x10,x12,x16 etc.

```
if {[sizeof_col [get_libs -quiet abc_*]] > 0 } {  
    # drive strength not to use  
    set_dont_use_lib_cell_abc_*/X16*  
    set_dont_use_lib_cell_abc_*/X14*  
    set_dont_use_lib_cell_abc_*/X12*  
    set_dont_use_lib_cell_abc_*/X10*  
}
```

Step 5) In step five, check for the design rule constraints (DRC's) issues and fix it in the STA tool.

```
fix_eco_drc -type max_transition -methods size_cell -setup_margin 0.01  
fix_eco_drc -type max_capacitance -methods size_cell -setup_margin 0.01
```

Step 6) In step six, fix the leakage power. Leakage power can be fixed with the cell replacement/swapping. In this, ULVT cells will be replaced/swapped with LVT cells.

```
fix_eco_leakage
```

Step 7) In step seven, the total power of digital block can be reduced by downsizing the high driving cells to the low driving strength cells.

```
fix_eco_power -verbose -setup_margin 0.045
```

Step 8) In step eight, fix the setup violations while designing by downsizing the cells in particular path group.

```
fix_eco_timing -type setup -methods size_cell
```

Step 9) Once the step violations are fixed. Fix hold timing violations. The hold timing violations can be fixed by inserting the buffers of lower driving strength as per the requirement.

```
fix_eco_timing -type hold -methods {size_cell_insert_buffer_at_load_pins}
```

Step 10) It is the final step of the algorithm, in which a patch is dumped by STA tool, and it implements the patch in the PNR tool.

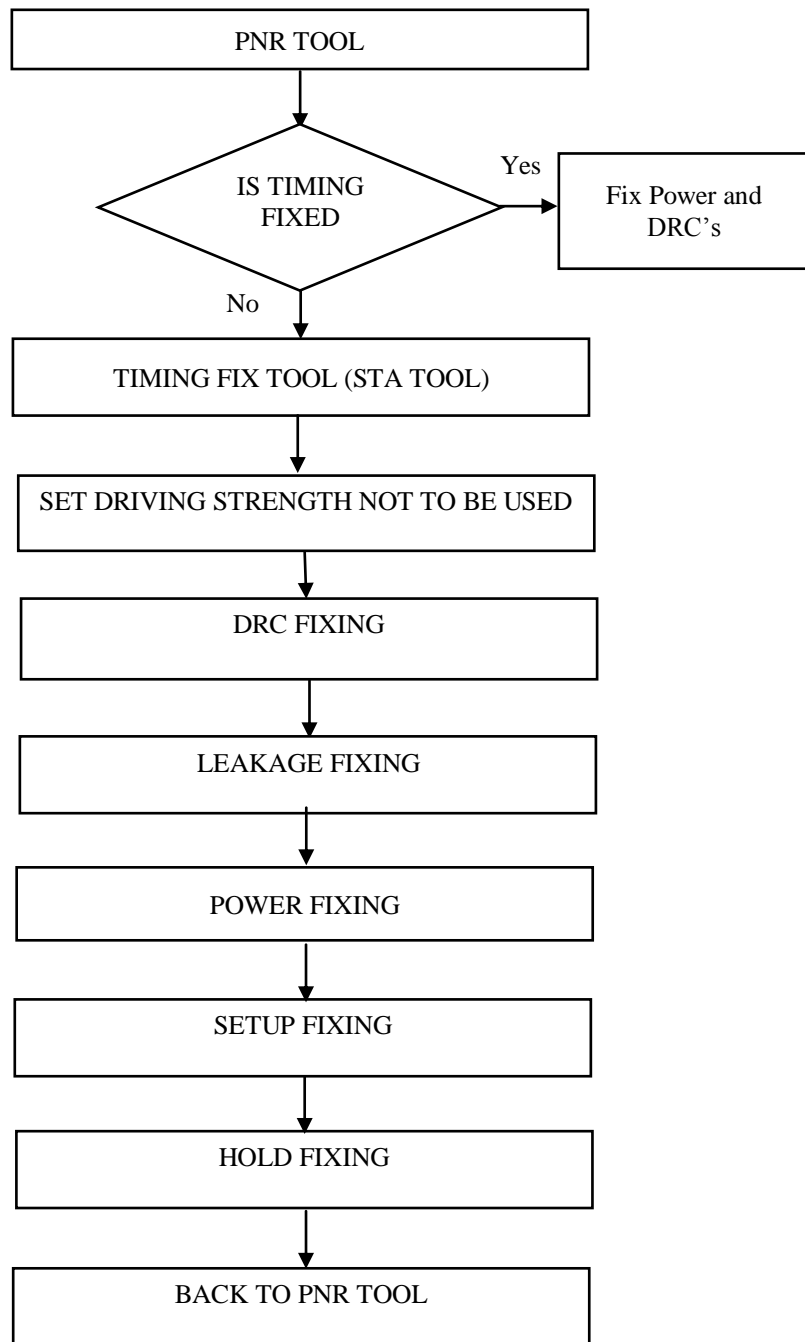


Figure 4.9 Proposed procedural algorithm defining optimum flow in physical design for timing and power fixing

4.8 CHAPTER SUMMARY

A methodology to fix the timing and power at the appropriate stage during the physical design implementation has been presented. Analysis related to setup timing and hold timing has been performed along with the analysis of power and its reduction techniques. The dependency of short circuit power as well as leakage power on threshold voltage has been analyzed, and the analytical expressions have been explored. Different methods for the reduction of the leakage power have been discussed. After analysing the leakage power alleviation methods, it has been found that there is a high degree of correlation between leakage power, dynamic power and delay of transistors. Based on the algorithm, a physical chip designer and a power consumption/dissipation analyzer can fix the power and timing of the design during the physical design implementation process.

CHAPTER 5 SIMULATION RESULTS

This chapter explains the different simulation results obtained while implementing the physical design of the block. Simulation results corresponding to a technique proposed in the previous chapter have been discussed in this chapter. This chapter discusses the design statistics, static time analysis (STA) results along with the power estimation. Results related to the reduction in the total power of the design have been discussed in this chapter.

5.1 SIMULATION RESULTS OF PHYSICALLY IMPLEMENTED DESIGN

Physical implemented block from networking ASIC has a different number of flip-flops, latches, combinational cells, test key cells, and memories. The data given below provides the information about the design statistics. With these design statistics, the physical design, STA analysis, and power analysis have been performed.

5.1.1 Physically Implemented Block Design Statistics

This block from networking ASIC has a different number of macros, flip-flops, latches, and combinational cells. Table 5.1 gives the detailed figures of the design statistics. Here, the flip-flop and latch are the circuits with two stable states; one is 1 and other is 0. These are bistable circuits and can be used to store the state information. Combinational cells can be any buffer and adder circuit etc. Test key cells are placed at a different location in the design and these can be used for testing purpose only. The design has different types of memories.

Sr.No	Name	Count
1	Latch	600+
2	Flops	20000+
3	Macros	30+
4	Test Key Cells	1+
5	Complex Cells	12000+
6	Combinational Cells	60000+
7	Memories Type	12+

Table 5.1 Design Statistics

5.1.2 Physically Implemented Block Utilization Statistics

After doing the floorplanning, the placement of the macros and standard cells in the design, the statistics regarding the area utilization of the chip has been shown in Table 5.2. The Table 5.2 indicates the area used by the standard cells, macros, and percentage of the areas from the chip used by the standard cells. Statistics regarding the core and die area of the chip have also been shown in the following table. Physically implemented block results have also been mentioned in chapter 1 with the view of all the standard cells and congestion in the design.

Sr.No	Name	Utilization (Approx.)
1	Macro Cell Area	$\sim 54041.70 \mu m^2$
2	Standard Cell Area	$\sim 52861.77 \mu m^2$
3	Core Size	$\sim 457.83 \times 372.67 \mu m^2$
4	Core Area	$\sim 170620.42 \mu m^2$
5	Chip Size	$\sim 460.00 \times 374.98 \mu m^2$
6	Chip Area	$\sim 172488.96 \mu m^2$
7	Standard Cells Utilization	$\sim 49.62\%$

Table 5.2 Chip Utilization

5.2 STATIC TIMING ANALYSIS SIMULATION RESULTS

After implementation of the block, the next important task is the Static timing analysis. The extraction done during the physical design implementation helps to do the STA analysis. To operate the chip at a particular frequency is very much important. And to meet the timings is one of the major and challenging tasks in the physical design. The design STA results are given below.

5.2.1 Different Path Groups and Clocks

A collection of paths in the given design is called as timing paths. These different paths have a starting point as well as ending point. Here, the starting points may be input ports and clock pins of different synchronous flip-flops of the design. End points can be data input pins and output ports. Therefore, there are different path groups present in the design and are described in Table 5.3.

Sr.No	Path Groups
1.	Register to Register
2.	Input to Output
3.	Input to Register
4.	Register to Out

Table 5.3 Different Path Groups Present in the Design

Table 5.4 provides information about the different clocks utilized in the design for performing the STA analysis. Period of different clocks has also been mentioned. To meet the frequency of the chip, these clock paths and clock period plays a very important role.

Sr.No	Clock Name	Period (ns)
1.	Clock A	2.5
2.	Clock B	5
3.	Clock C	2.5
4.	Clock D	5

Table 5.4 Different Types Of Clocks and Their Time Period

5.2.2 Setup Timing and Hold Timing Violations During The First Iteration

Setup timing and hold timing violations are the major violations in the physical design implementation. These violations can cause the chip to fail to work on the desired frequency. These violations need to be fixed and the chip should operate on the desired frequency. The algorithm to fix these violations at a particular stage has been discussed in chapter 4. The violations observed among different path groups and setup violations has been fixed after decreasing the delay of the flops and the combination circuits used in the design (as mentioned below).

5.2.2.1 Before Fixing

Before downsizing the cells in the design during iteration one the setup timing and hold timing violations has been observed in the block and have been shown in Table 5.5

WNS (ns) - Worst negative slack

TNS (ns) - Total negative slack

NVP-Number of violating paths

SETUP	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	-1.3494	0	-1.3494	0	0
TNS	-309.8545	0	-309.8545	0	0
NVP	443	0	443	0	0

Table 5.5 First Iteration Setup Violations Among Different Path Groups Before Fixing

HOLD	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	-0.0228	-0.0228	0	0	0
TNS	-5.0536	-5.0536	0	0	0
NVP	967	967	0	0	0

Table 5.6 First Iteration Hold Violations Among Different Path Groups Before Fixing

5.2.2.2 After Fixing

After downsizing the cells and decreasing the delays of combinational and sequential circuits, the setup violations have been fixed. For fixing the hold violation, some buffers have been inserted to increase the delay so that the hold violations should not be observed in the design.

The results given below show the setup and hold after fixing.

SETUP	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	0	0	0	0	0
TNS	0	0	0	0	0
NVP	0	0	0	0	0

Table 5.7 First Iteration Setup Violations Among Different Path Groups After Fixing

HOLD	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	-0.000	-0.000	0	0	0
TNS	-0.000	0.000	0	0	0
NVP	20	20	0	0	0

Table 5.8. First Iteration Hold Violations Among Different Path Groups After Fixing

5.2.3 Setup Timing and Hold Timing Violations During The Second Iteration

After the first round is completed, a patch is dumped by the STA tool, which contains all the information regarding the downsized cells and inserted buffers for fixing the setup timing and hold timing violations. This patch needs to be inserted in PNR tool, and again the process needs to repeat again to detect/rectify setup timing and hold timing violations. The placement of newly inserted buffers may cause the setup timing and hold timing violations in other paths. Therefore, the second iteration needs to be done.

5.2.3.1 Before Fixing

After taking the data back for STA analysis and after inserting as well as resizing the buffers, the following setup timing and hold timing violations observed.

SETUP	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	-0.0089	0	-0.0089	0	0
TNS	-0.0187	0	-0.0187	0	0
NVP	4	0	4	0	0

Table 5.9 Second Iteration Setup Violations Among Different Path Groups Before Fixing

Hold	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	-0.0226	-0.0226	0	0	0
TNS	-5.0959	-5.0959	0	0	0
NVP	979	979	0	0	0

Table 5.10 Second Iteration Hold Violations Among Different Path Groups Before Fixing

5.2.3.2 After Fixing

After fixing the setup timing and hold timing violations, the following results have been observed.

SETUP	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	0	0	0	0	0
TNS	0	0	0	0	0
NVP	0	0	0	0	0

Table 5.11 Second Iteration Setup Violations Among Different Path Groups After Fixing

Hold	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	-0.000	-0.000	0	0	0
TNS	-0.000	-0.000	0	0	0
NVP	4	4	0	0	0

Table 5.12 Second Iteration Hold Violations Among Different Path Groups After Fixing

5.2.4 Setup Timing and Hold Timing Violations During The Third Iteration

During the third round of iteration, the following results are obtained for setup and hold analysis.

5.2.4.1 Before Fixing

Once the second round of the violation fixing is over, there may be some leftover violations which need to be detected further. The results obtained after inserting the patch in STA tool are given below.

SETUP	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	0	0	0	0	0
TNS	0	0	0	0	0
NVP	0	0	0	0	0

Table 5.13 Third Iteration Setup Violations Among Different Path Groups Before Fixing

Hold	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	-0.0023	-0.0023	0	0	0
TNS	-0.0348	-0.0348	0	0	0
NVP	66	66	0	0	0

Table 5.14 Third Iteration Hold Violations Among Different Path Groups Before Fixing

5.2.4.2 After Fixing

After fixing the violations the results are shown in the next tables

SETUP	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	0	0	0	0	0
TNS	0	0	0	0	0
NVP	0	0	0	0	0

Table 5.15 Third Iteration Setup Violations Among Different Path Groups After Fixing

HOLD	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	0	0	0	0	0
TNS	0	0	0	0	0
NVP	0	0	0	0	0

Table 5.16 Third Iteration Hold Violations Among Different Path Groups After Fixing

5.2.5 Setup Timing and Hold Timing Violations During The Fourth Iteration

This is the final iteration for fixing the setup timing and hold timing violations in the implemented design. The results obtained are as follows

5.2.5.1 Before Fixing

Before fixing the setup and hold timing violations while designing, the results obtained are shown in following tables, no more violation fixing procedure is continued.

SETUP	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	0	0	0	0	0
TNS	0	0	0	0	0
NVP	0	0	0	0	0

Table 5.17 Fourth Iteration Setup Violations Among Different Path Groups Before Fixing

HOLD	Total	Reg2Reg	In2Reg	Reg2Out	In2Out
WNS	0	0	0	0	0
TNS	0	0	0	0	0
NVP	0	0	0	0	0

Table 5.18 Fourth Iteration Hold Violations Among Different Path Groups Before Fixing

5.2.6 Graphical Representation of Violations

The graphical representation of all the violations with different iterations is shown below

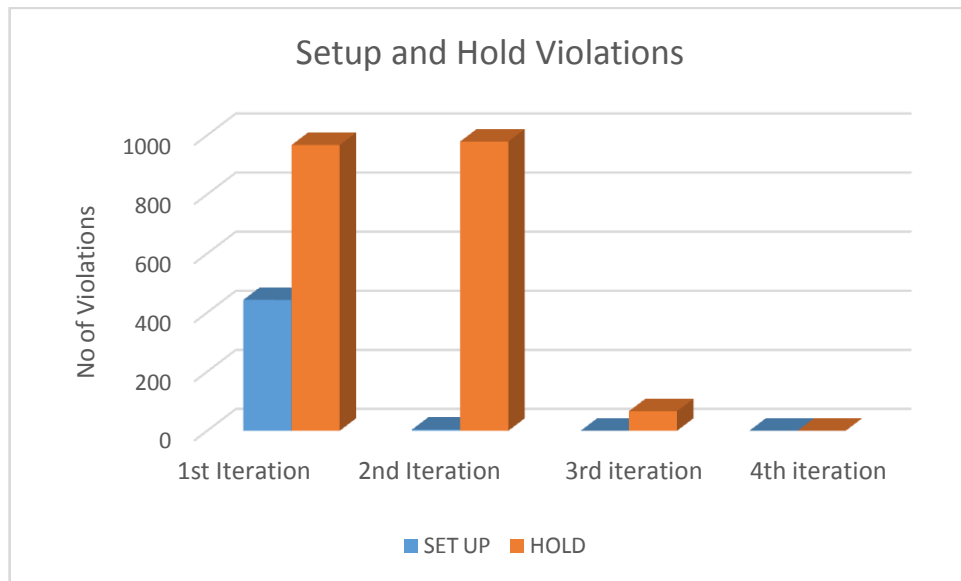


Figure 5.1 Setup and hold violations

5.3 CHIP POWER ESTIMATION SIMULATION RESULTS

Once the timing is fixed in the design, the next important factor is power estimation.

5.3.1 Ultra-Low-Threshold-Voltage (ULVT) and Low-Threshold-Voltage (LVT) Statistics

The number of ultra-low voltage and low voltage cells present in the design has been specified in the tables given below. Table 5.19 and Table 5.20 detail about the count of the cells before and after swapping/replaced.

Lib- Library of cells

Num- Total number of cells present in the design

Num %- Percentage of the cells in the design

Sr.No	Lib	Num	Num %
1.	ULVT I	1201	1.3
2.	ULVT II	1572	1.6
3.	LVT I	5064	5.4
4.	LVT II	85074	91.5

Table 5.19 Ultra-Low and Low Voltage Threshold Cell Count Before Cell Swapping/Replacing

Sr.No	Lib	Num	Num %
1.	ULVT I	69	0.1
2.	ULVT II	272	0.3
3.	LVT I	6196	6.7
4.	LVT II	86374	93

Table 5.20 Final Cell Count After Closing The Design

5.3.2 Power Estimation Comparison Before and After Cell Swapping/Replacing (in Layout Design)

Cell swapping/replacing is an effective way to reduce power usage. The following outcomes are obtained after doing the cell replacing/swapping in the design.

Sr.No	Power Component	Power Before ULVT swap (mW)	Power After ULVT swap (mW)
1	Internal Power	21.6835	20.4348
2	Leakage Power	6.0073	5.029
3	Dynamic Power	14.884	12.801
4	Total Power	42.5742	38.4648

Table 5.21 Power Estimation Before and After Cell Replacing/Swapping

5.3.3 Final Chip Power Estimation

The final power estimation statistics of the design after closing the design are demonstrated in the tables mentioned below

\

Analysis Conditions		
Sr.No	Component	Value
1	Ambient Temperature	25.00 C
2	Junction Temperature	105.00 C
3	Theta JA	15.00
4	VDD Domain :VDD Worst Voltage	0.800 V
5	VDD Domain :VDD Nominal Voltage	0.800 V

Table 5.22 Analysis Conditions

Worst Case Power				
Sr.No	Component	Static Power(mW)	Dynamic Power (mW)	Total(mW)
1	Combinational	2.40	1.11	3.51
2	Storage	3.32	5.29	8.62
3	Memory	4.12	2.18	6.30
4	Clock Tree	0.35	2.51	2.87
5	DCAP	0.00	0.00	0.00
6	Total	10.19	11.10	21.29

Table 5.23 Worst Case Power

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

This chapter concludes the research work presented in this thesis and gives an overview of the future scope of presented work.

An exponential rise in the number of transistors in latest VLSI circuits causes the physical design implementation to be exceptionally complicated and time-consuming to attain acceptable solutions, especially for the large circuits. Nowadays, the power consumption/dissipation, delay, and area are the most important factors affecting the performance in current technologies. These issues accentuate the importance of the VLSI physical design problem. Therefore, the effective and efficient electronic-design-automation (EDA) techniques are necessary to deal with these emerging challenges.

This thesis research work basically focuses on the optimum physical design and power estimation of a chip of a block level networking ASIC. The VLSI industry is growing by obeying Moore's law, where a single chip can accommodate a large number of components. To achieve scalability in the design process without sacrificing the quality of the design is becoming tedious. There is an impending requirement to do the efficient placement and physical design process that can address high-quality results in shorter time. Physical design, which generally deals with the standard cell, macros placement, and routing, has a direct impact on the static-timing-analysis (STA) while designing. The STA analysis is the way to compute the design frequency, and the design engineers spend months to meet the timing requirements while designing process. Timing is very important factor for operating the chip at a desired frequency. In this research work, an efficient placement of standard cells, macros have been done, so that there should not be any timing violations, and the chip must operate at a desired frequency. After doing the placement and routing, the STA analysis has been done and the algorithm has been proposed representing the correct and optimum flow to fix the different types of the violations in the desired design. This algorithm shows the best results in order to fix the timing violations with a minimum number of iterations.

The current era of VLSI industry is going through the phase in which designers are emphasizing on saving power, areas and reducing the time for a product (chip) to hit the market. With the progress in VLSI industry, the power consumption/dissipation is becoming a major issue. The leakage power is dominating over the dynamic power. Therefore, scheme has been suggested to fix the leakage power in desired digital design. In this approach, the ultra-low threshold voltage cells have been replaced with the low threshold voltage cells

before the placement of standard cells in the core area of a chip. A significant reduction, in the leakage power has been observed while using this approach. It is evident from the simulation results presented for the proposed cell replacement/swapping design technique that the internal power has reduced by 5.76%, the leakage power has alleviated by 16.29%, the dynamic power has got lessen by approximately 14% and the total power consumption and dissipation has decreased by approximately 9.65%. The future scope of the research work presented in this thesis is summarized as below:

1. Efficient “place-and-route” algorithms are needed, so that there should be very less utility of inserting the buffers in the design in order to fix the process timing, which has a direct impact on the leakage power of the desired design.
2. Efficient methods can be developed to control the switching activity of the clock pulse in order to save the power of the digital design.
3. Clock gating methods are efficient to reduce the power consumption. Therefore, it can give a greater edge in the field of the VLSI design industry.
4. Tunneling field effect transistors are a realistic candidate for the future low voltage logic. These transistors with steep turn-on characteristics perform faster than CMOS transistor at around 0.3V supply voltage and as a result, it can appear as an appropriate choice for very low power applications in future.
5. Different tools have been used for doing the placement, routing and static timing analysis. There is a dire need to make a single tool having better efficiency and accurate result formation capability.
6. Low threshold voltage devices are required for the designs to operate at the low input voltage. These low voltage devices are fast in operation while exhibiting minimum delay at the cost of increased leakage power, therefore the improved techniques are required to optimize the design which can maintain a balance between delay and leakage power consumption/dissipation.

REFERENCES

- [1] Golshan K. *Physical design essentials: An ASIC design implementation perspective*. USA: Springer, 2007.
- [2] Bhasker J. and Chadha R. *Static timing analysis for nanometer designs*. USA: Springer, 2009.
- [3] Weste N.H.E. and Harris D.M. *CMOS VLSI Design*. USA: Pearson Education, 2005.
- [4] Parihar K., Venkatesh M. and Patel R. (2015). Realistic dynamic timing verification for complex mixed-signal hard macro's using UVM, *IEEE International Symposium on VLSI Design and Test* [19th: Ahmadabad, India: 2015], pp.1-2.
- [5] Wei Y. *et al.* (2015). Always on buffer clustering implementation in low power physical design of 28nm process, *IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems* [Shenyang, China: 2015], pp. 1775-1780.
- [6] Ahmad M., Alizadeh B. and Forouzandeh B. (2015). A timing error mitigation technique for high-performance designs, *IEEE Symposium on VLSI* [Montpellier, France: 2015], pp. 428-433.
- [7] Luo G., Jin B. and Zhang W. (2015). A fast and simple block based approach for common path pessimism removal in static timing analysis, *IEEE International Conference on Computer-Aided Design of Integrated Circuits and Systems* [14th: Xi'an, China: 2015], pp. 234-235.
- [8] Singh R. and Tripathi A.S.M (2015). A novel approach for glitch and leakage power reduction in CMOS VLSI circuits, *International Journal of Engineering and Technical Research* [Aligarh, India: 2015], pp. 1-6.
- [9] Jung J., Lee D. and Shin Y. (2015). Design and optimization of multi mesh-clock network, *IEEE Conference on Very Large Scale Integration* [Daejeon, South Korea: 2015], pp. 1-6.
- [10] Harris D.M. (2015). Sequential element timing parameter definition considering clock uncertainty, *IEEE Transaction on Very Large Scale Integration Systems*, 23(11), 2705-2708.
- [11] Cao W. *et al.* (2015). 2D semiconductor FETs - projections and design for submicron 10nm VLSI, *IEEE Transactions on Electronic Devices*, 62 (11), 3459-3469.
- [12] Garg V. (2014). Common path pessimism removal, *IEEE International Conference on Computer-Aided Design of Integrated Circuits and Systems* [San Jose, CA, USA: 2014], pp. 592-595.

- [13] Fuang P.S. and Mahyuddin N.M. (2014). Implementation study of path-based AOCV model on pessimism reduction for 20nm technology, *IEEE International Conference on Control Systems, Computing and Engineering* [Batu Ferringhi, Malaysia: 2014], pp. 80-83.
- [14] Monthie J., Sreekumar V. and Yasgwante R. (2013). Impact of power supply noise on clock jitter in high-speed DDR memory interfaces, *International Conference on VLSI Design and Embedded Systems* [26th: Pune, India: 2013], pp. 262-266.
- [15] Kumar V.B.V.P. *et al.* (2013). Selective glitch reduction technique for minimizing peak dynamic IR drop, *IEEE Asia Pacific Conference on Postgraduate Research in Microelectronics and Solid state Electronics* [Hyderabad, India: 2013], pp. 27-32.
- [16] Wiltgen A. *et al.* (2013). Power consumption analysis in static CMOS gates, *IEEE International Symposium on Integrated Circuits and Systems Designs* [26th: Curitiba, Brazil: 2013], pp. 1-6.
- [17] Priya M.G., Baskaran K., and Krishnaveni D. (2012). Leakage power reduction techniques in deep submicron technologies for VLSI applications, *International Conference on Communication Technology and System Design* [Coimbatore, India: 2012], pp. 1163-1170.
- [18] Morgenshtein A. (2012). Short-Circuit power reduction by using high-threshold transistors, *IEEE Journal of Low Power Electronics and Applications*, 8(2), 69-78.
- [19] Saini P. and Mehra R. (2012). A novel technique for glitch and leakage power reduction in CMOS VLSI circuits, *International Journal of Advanced Computer Science and Applications*, 3(10), 161-168.
- [20] Khan Z.A. *et al.* (2011). Optimization of power consumption in VLSI circuits, *International Journal of Computer Sciences Issues*, 8(2), 648-653.
- [21] Pan J. *et al.* (2011). Physical design of YAK SoC by using an efficient clock tree synthesis method, *IEEE International Conference on Electric Information and Control Engineering* [Wuhan, China: 2011], pp. 598-601.
- [22] Rajaram A. and Pan D.Z. (2011). Robust chip level clock trees synthesis, *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, 30(6), 877-890.
- [23] Jeon H., Kim Y. and Choi M. (2010). Standby leakage power reduction technique for nanoscale CMOS VLSI systems, *IEEE Transaction on Instrumentation and Measurement*, 59 (5), 1127-1133.
- [24] Alexander J.D. and Agrawal V.D. (2009). Algorithms for estimating number of glitches and dynamic power in CMOS circuits with delay variations, *IEEE Computer*

- Society Annual Symposium on VLSI* [Tampa, Florida, USA: 2009], pp. 127-132.
- [25] Cai Y. *et al.* (2015). Obstacle avoiding and slew constrained clock tree synthesis with efficient buffer insertion, *IEEE Transaction on Very Large Scale Integration System*, 23(1), 142-155.
- [26] Zhao P., Wang Z. and Hang G. (2010). Power optimization for VLSI circuits and systems, *IEEE International Conference on Solid-State and Integrated Circuits* [10th: Shanghai, China: 2010], pp. 639-642.
- [27] Salman E. *et al.* (2007). Exploiting setup-hold time interdependence in static timing analysis, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 26(6), 1114-1125.
- [28] Aggarwal A. (2006). Leakage power analysis and reduction for nanoscale circuits, *IEEE Computer Society*, 26(2), 68-80.
- [29] Vygen J. (2006). Slack in static timing analysis, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(9), 1876-1885.
- [30] Heo J. and Kim T. (2016). Timing analysis and optimization based on flexible flip flop timing model, *IEEE Computer Society Annual Symposium on VLSI* [Pittsburgh, PA, USA: 2016], pp. 42-46.
- [31] Zhang W. *et al.* (2005), Physical insights regarding design and performance of independent gate FinFETs, *IEEE Transactions on Electronic Devices*, 52(10), 2198-2206.
- [32] Aggarwal N. (2015). Test circuit for accurate measurement of setup/hold and access time of memories, *IEEE international conference of microelectronic test circuit* [Tempe, AZ, USA: 2015], pp. 204-206.
- [33] Kim J.G. and Kim Y.D. (2003). A linear programming base algorithm for floorplanning in VLSI design, *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, 22 (5), 584-592.
- [34] Khowri K.S. and Jha N.K. (2002). Leakage power analysis and reduction during behavioural synthesis, *IEEE Transaction on Very Large Scale Integration Systems*, 10 (6), 876-885.
- [35] Wu Q., Qiu Q. and Pedram M. (2001). Estimation of peak power dissipation in VLSI circuits using the limiting distributions of extreme order statistics, *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, 20 (8), 942-956.
- [36] Benini L. *et al.*, (2000), Glitch power minimization by selective gate freezing, *IEEE Transaction on Very Large Scale Integration Systems*, 8(11), 287-298.
- [37] Athas W. (1999). Low-Power VLSI techniques for applications in embedded

- computing, *IEEE Alessandro Volta Memorial Workshop on Low-Power Design* [Como, Italy: 2012], pp. 14 -22.
- [38] Ismail Y.I., Friedman E.G. and Neves J.L. (1999). Dynamic and short-circuit power of CMOS gates driving lossless transmission lines, *IEEE Transaction on Circuits and Systems*, 46 (8), 950-961.
- [39] Lai G.G., Fussell D.S. and Wong D.F. (1996). Hinted quad trees for VLSI geometry DRC based on efficient searching for neighbours, *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, 15(3), 317-324.
- [40] Liu D. and Svensson C. (1994). Power consumption estimation in CMOS VLSI chips, *IEEE Journal of Solid-State Circuits*, 29 (6), 663-670.
- [41] Levitt M. E. and Abraham J.A. (1989). Physical design of testable VLSI: techniques and experiments, *IEEE Custom Integrated Circuits Conference* [San Diego, CA, USA: 1989], pp. 1-4.
- [42] Kang S.M. (1986). Accurate simulation of power dissipation in VLSI circuits, *IEEE Journal of Solid-State Circuits*, 21(5), 889-891.