

# **Robust Timing Analysis & Modelling of Custom High Speed Serial IO Blocks**

Major Project Report

Submitted in partial fulfillment of the requirements  
For the degree of

Master of Technology  
In  
Electronics & Communication Engineering  
(VLSI Design)

By  
**Shah Darshan Sunil**  
**(16MECV23)**



**Electronics & Communication Engineering Department  
Institute of Technology**

**Nirma University**

**Ahmedabad - 382 481**

**May, 2018**



# **Robust Timing Analysis & Modelling of Custom High Speed Serial IO Blocks**

Major Project Report

Submitted in partial fulfillment of the requirements  
For the degree of

Master of Technology  
In  
Electronics & Communication Engineering  
(VLSI Design)

By  
**Shah Darshan Sunil**  
**(16MECV23)**

Under the Guidance of  
**Dr. N. M. Devashrayee**



**Electronics & Communication Engineering Department  
Institute of Technology**

**Nirma University**

**Ahmedabad - 382 481**

**May, 2018**



# Declaration

This is to certify that

1. The thesis comprises my original work towards the degree of Master of Technology in VLSI Design at Nirma University and has not been submitted elsewhere for a degree.
2. Due acknowledgment has been made in the text to all other material used.

Shah Darshan Sunil





## Certificate

This is to certify that the Major Project entitled “**Robust Timing Analysis & Modelling of Custom High Speed Serial IO blocks**” submitted by **Shah Darshan Sunil(16MECV23)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in VLSI Design, Nirma University, Ahmedabad is the record of work carried out by him under our supervision and guidance. In our opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of our knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Dr. N. M. Devashrayee  
Internal Guide

Prof. Dr N. M. Devashrayee  
PG Coordinator (VLSI Design)

Dr D. K. Kothari  
Head, EC Dept.

Dr Alka Mahajan  
Director, IT - NU

Date :

Place : Ahmedabad







## Certificate

This is to certify that the Project entitled ”**Robust Timing Analysis & Modelling of Custom High Speed Serial IO Blocks**” submitted by **Darshan Shah (16MECV23)**, towards the submission of the Project for requirements for the degree of Master of Technology in VLSI Design, Nirma University, Ahmedabad is the record of work carried out by him under our supervision and guidance. In our opinion, the submitted work has reached a level required for being accepted for examination.

(External Guide)

Mr.Kaushik Saiprasad  
Engineering Manager  
Intel Technology India  
Bangalore

(Mentor)

Mr. Tanuj Jindal  
Digital Design Engineer  
Intel Technology India  
Bangalore

Company Seal

Intel Technology India Pvt. Ltd.(Bangalore)

Date :

Place : Bangalore



# Acknowledgment

First and foremost, sincere gratitude to my manager Mr. Kaushik Saiprasad. Also I want to thank Intel Technology India Private Limited, Bangalore for assigning me such project and guide me through.

I would like to express my gratitude and sincere thanks to my mentor Tanuj Jindal and Khushal Gondaliya, at Intel Technology India Private Limited, Bangalore for their valuable guidance during the project work, they have given me valuable advices and support which I am very lucky to benefit from.

I would also thank to my internal guide, Dr N. M. Devashrayee, Professor, VLSI Design, Institute of Technology, Nirma University, Ahmedabad for giving valuable the advice and support throughout the semester.

I would also like to express my gratitude to all faculty members of Nirma University for providing encouragement, exchanging knowledge during my post-graduate program.

- Shah Darshan Sunil (16MECV23)



# Abstract

Major design challenges of ASIC design are ultra-high speeds, power dissipation, supply rail drop, interconnect, noise, crosstalk, reliability, manufacturability and the clock distribution. On Chip Variation (OCV) is one of the barriers which contribute to these challenges and its effects are increasing with smaller process node.

With lower technology node, clock tree robustness has become an even more critical factor affecting SoC performance. Conventionally, our focus is on designing a symmetrical clock tree with minimum latency and skew. Another design challenge is the effect of crosstalk, which plays an important role in the signal integrity of the design. Crosstalk analysis are used to make the ASIC behave robustly from a timing perspective. The design functionality and its performance can be limited by noise, will impact on frequency of design and also causing functional failures.

To mitigate the above discussed issues like crosstalk and clock tree distribution it needs to devise some new methodology which provides improvement to increase robustness of the design against variations. In this thesis we focus on one such methodology which is clock mesh technology. Clock mesh technology provides uniform low skew clock distribution and offers better tolerance to on-chip variations (OCV) than conventional clock tree technology. Output slew analyzed on clock buffer clock inverter considering RC impact. Also cross talk analysis is done for various metal length and layers with different spacing, which will give the constraints on the maximum length to be used for routing.

# Table of Contents

<b>Declaration</b>	<b>i</b>
<b>Certificate</b>	<b>iii</b>
<b>Internship Certificate</b>	<b>v</b>
<b>Acknowledgment</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>List of Figures</b>	<b>i</b>
<b>List of Tables</b>	<b>ii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 CTS . . . . .	1
1.2 Crosstalk Noise . . . . .	2
1.3 Clock Buffer Vs Normal Buffer . . . . .	3
1.4 Traditional IC Physical Design Flow . . . . .	3
1.5 Motivation . . . . .	3
1.6 Objective . . . . .	4
1.7 Preface . . . . .	5
<b>2 Literature Survey</b>	<b>7</b>
2.1 Clock Mesh Variation Robustness: Benefits and Analysis . . . . .	7

2.1.1	Key Differences Between CTS And Clock Mesh . . . . .	7
2.2	Crosstalk Delay Analysis . . . . .	10
2.2.1	Timing effect of Crosstalk Delay Violations . . . . .	12
<b>3</b>	<b>CTS variation robustness</b>	<b>15</b>
3.1	Variation at Advanced Technology Nodes . . . . .	15
3.1.1	Clock Mesh versus Conventional Clock Tree Structure . . . . .	16
3.2	Understanding OCV Derating . . . . .	18
3.2.1	Simulation and Analysis of OCV Effects . . . . .	19
<b>4</b>	<b>Crosstalk Delay Analysis</b>	<b>21</b>
4.1	Crosstalk Delay Effects . . . . .	21
4.1.1	Delta Delay and Fanout Stage Effect . . . . .	22
4.1.2	Aggressor and Victim Nets . . . . .	22
4.2	Crosstalk effect on fanout . . . . .	24
4.3	Experimental Results . . . . .	24
<b>5</b>	<b>Clock Inverter Vs Clock Buffer based Clock Tree</b>	<b>29</b>
5.1	Clock Buffer Vs Normal Buffer . . . . .	29
5.1.1	Min Pulse Width . . . . .	30
5.2	Inverter Based Clock Tree . . . . .	32
5.3	Buffer Based Clock Tree . . . . .	33
5.3.1	Experimental Results . . . . .	36
<b>6</b>	<b>Conclusion and Future Work</b>	<b>41</b>
6.1	Conclusion . . . . .	41
6.2	Future Work . . . . .	41
	<b>References</b>	<b>43</b>

# List of Figures

1.2.1 Shrinking of wire geometries in the nanometer process technology leads to an increase in the amount of coupling capacitance[2]. . . . .	2
1.4.1 physical design flow . . . . .	4
2.1.1 Set of sinks addressed by each of the two clock distribution methods[4] . . . . .	8
2.2.1 Example of coupled interconnect[9] . . . . .	11
2.2.2 Crosstalk impact example[9] . . . . .	12
2.2.3 Hold Violations due to Crosstalk Effect . . . . .	13
2.2.4 Setup Violations due to Crosstalk Effect . . . . .	14
2.2.5 Setup Violation due to Crosstalk Delay . . . . .	14
3.1.1 Clock Structures - Conventional clock tree and clock mesh[4] . . . . .	17
3.1.2 SPICE waveforms showing the smoothing effect of the mesh net . . . . .	18
3.2.1 OCV tolerance - clock tree vs clock mesh[4] . . . . .	19
3.2.2 Monte Carlo simulation results[5] . . . . .	20
4.1.1 Transition slowdown or speedup caused by crosstalk . . . . .	22
4.1.2 Delta delay includes the fanout stage effect . . . . .	22
4.1.3 Transition slowdown or speedup caused by crosstalk . . . . .	23
4.2.1 Crosstalk effect modeled as delta delay for current stage and fanout . . . . .	24
4.3.1 Different cases for proposed approach . . . . .	25
4.3.2 Results for different cases having length of 200 micron for metal m3 . . . . .	25
4.3.3 Graph of stagedelay and crosstalk delay versus different sizes for different cases	26
4.3.4 Results for different cases having length of 250 micron for metal m3 . . . . .	27



4.3.5 Graph of stagedelay and crosstalk delay versus different sizes for different cases having length of 250 micron for metal m3 . . . . .	27
4.3.6 Results for different cases having length of 200 micron for metal m5 and m7 . .	28
4.3.7 Graph of stagedelay and crosstalk delay versus different sizes for different cases having length of 200 micron for metal m3 and m5 . . . . .	28
5.1.1 Pulse width for normal buffer with different rise and fall delay[10] . . . . .	31
5.1.2 Pulse width check for clock signal which is going to clock pin of flop through series of buffers with different rise and fall delay . . . . .	32
5.2.1 Inverter Based Clock Tree giving equal rise and fall times . . . . .	33
5.3.1 Buffer Based Clock Tree. Buffer is formed by connecting two invertes back to back . . . . .	33
5.3.2 Difference in high and low pulse width . . . . .	34
5.3.3 RC delay model for inverters and wire . . . . .	35
5.3.4 Different cases for proposed approach . . . . .	36
5.3.5 Result for out_tran for input_tran 20ps output_cap 40fF rc_length 20 um for metal M5 . . . . .	38
5.3.6 Result for out_delay for input_tran 20ps output_cap 40fF rc_length 20 um for metal M5 . . . . .	39

# List of Tables

- 3.1 Insertion delay and skew of clock tree and clock mesh . . . . . 17
  
- 5.1 Result for out\_tran & out\_dly for input\_tran 20ps output\_cap 40fF rc\_length 20  
um for metal M5. . . . . 37

# Chapter 1

## Introduction

Robust means ability of a system to resist change without adapting its initial stable configuration. Thus, a design implementation must be verified to be robust before timing analysis which means that it can withstand the noise without affecting the rated performance of the design. Here, we discuss some important step to improve robustness of the design, Clock tree structure and Crosstalk analysis.

### 1.1 CTS

Clock tree synthesis (CTS) is at the heart of ASIC design and clock tree network robustness is one of the most important quality metrics of SoC design. With technology advancement happened over the past one and half decade, clock tree robustness has become an even more critical factor affecting SoC performance. Conventionally, engineers focus on designing a symmetrical clock tree with minimum latency and skew.

Today, SoCs are designed to support multiple features. They have multiple clock sources and user modes which makes the clock tree architecture complex. Merging test clocking with functional clocking and lower technology nodes adds to this complexity. Due to the increase in derate numbers and additional timing signoff corners, timing margins are shrinking.

Technologies that offer variation tolerance boost design performance and productivity. Im-

proven Techniques like Clock mesh technology provides uniform, low skew clock distribution and offers better tolerance to on-chip variations (OCV) than conventional clock tree technology. The need to control OCV effects is now driving clock mesh technology to mainstream designs.

## 1.2 Crosstalk Noise

In deep sub-micron technology (i.e. 30nm) and below, the lateral capacitance between nets/wires on silicon, becomes much more dominant than the interlayer capacitance. Hence, there is a capacitive coupling between the nets, that can lead to logic failures and degradation of timing in VLSI circuits. Crosstalk is a phenomenon, by which a logic transmitted in vlsi circuit or a net/wire creates undesired effect on the neighbouring circuit or nets/wires, due to capacitive coupling.

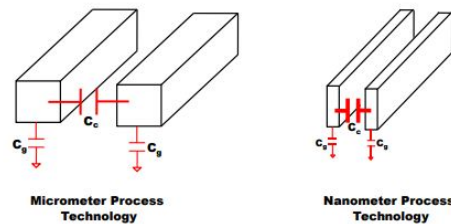


Figure 1.2.1: Shrinking of wire geometries in the nanometer process technology leads to an increase in the amount of coupling capacitance[2].

As the feature sizes have been shrinking with process-technology scaling, the spacing between adjacent interconnect wires keeps decreasing in every process technology. Also, while the lateral width of interconnect wires has been scaled down significantly their vertical height has not been scaled in proportion (as shown in Figure 1.2.1). Both these trends lead to a very rapid increase in the amount of coupling capacitance (essentially like parallel-plate capacitors) between the wires. It was reported that coupling capacitance accounts for more than 85% of the total interconnect capacitance in the 22nm technology node. More aggressive technology scaling will only lead to an increase in the overall contribution of the coupling capacitances

to the total interconnect capacitance. Therefore, signal-integrity issues such as crosstalk noise have become important when performing timing verification of VLSI chips.

### **1.3 Clock Buffer Vs Normal Buffer**

Advantages of Clock Buffer over Clock inverter.

1. Clock buffer have equal rise time and fall time compared to Normal buffers, therefore pulse width violation is avoided.
2. Clock buffer are usually designed such that an input signal with 50% duty cycle produces an output with 50% duty cycle. This usually isn't true for a normal buffer.
3. In Clock buffers Beta ratio is adjusted such that rise & fall time are matched by increasing size compared to normal buffer.
4. Clock net is one of the High Fanout Net(HFN)s. The clock buffers are designed with some special property like high drive strength and less delay. Clock buffers have equal rise and fall time. This prevents duty cycle of clock signal from changing when it passes through a chain of clock buffers.

### **1.4 Traditional IC Physical Design Flow**

Traditional IC design flow shown in above fig 1.4.1

### **1.5 Motivation**

1. Industry is transforming towards lower technology node, making everything complex.
2. To overcome this complexity, we have to make the analysis methodology robust.
3. To experienced with new methodologies to achieve design goals: clock frequency, OCV tolerance, power consumption, flow ease, and time-to-market pressure.

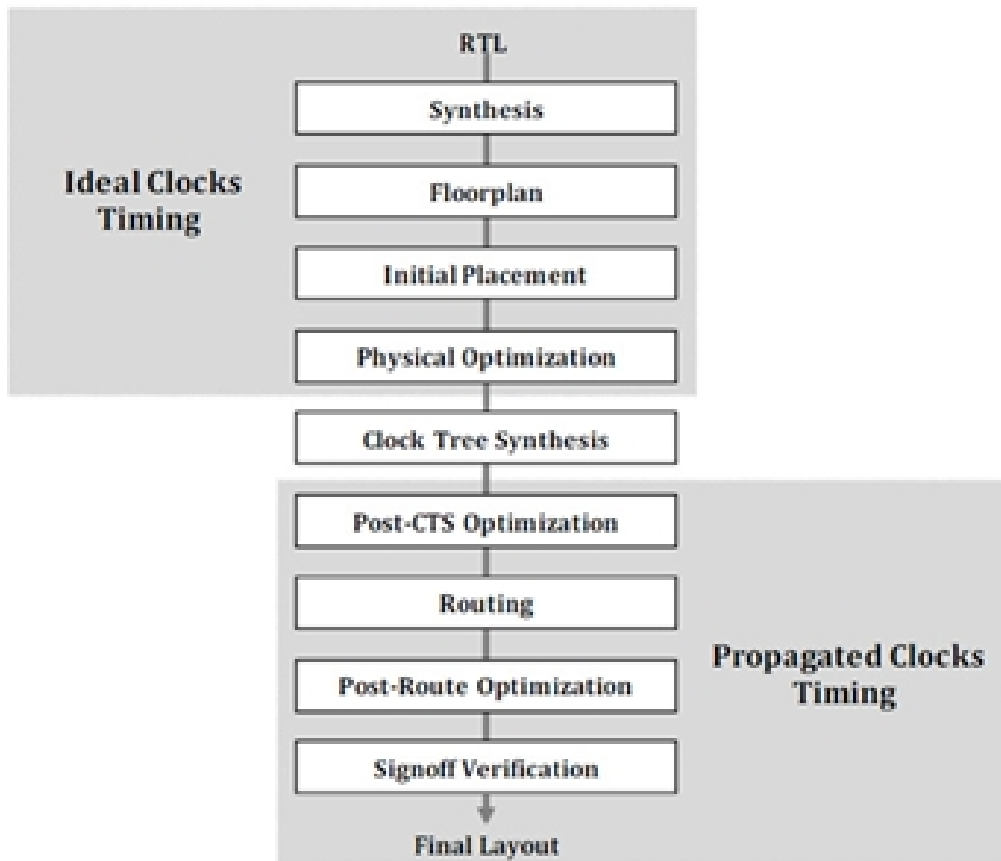


Figure 1.4.1: physical design flow

4. To improve the timing QoR of the design by designing robust clock tree considering OCV tolerance and RC impact.

## 1.6 Objective

1. Analyzing crosstalk for different metal layers of different size with different spacing between them before routing.
2. Making clock tree structure more robust to withstand the variation effect.
3. Designing a symmetrical clock tree with minimum insertion delay and 50% duty cycle
4. Offering better tolerance to RC variations.

5. Achieving high performance with equal rise and fall delay for clock tree cells.

## 1.7 Preface

The report is organized such that the basic underlying concepts are described first before delving into more advanced topics. The report starts with the basic CTS and Crosstalk, followed by commonly used clock tree structure, clock mesh, and the handling of noise and crosstalk for a nanometer design. Also, explain benefits of Clock buffer compared to Normal buffer to achieve high performance

Chapter 2 provides an literature survey of clock tree distribution and crosstalk delay analysis with advancement in technology node.

Chapter 3 This chapter gives an overview and highlights the benefits of clock mesh technology compared to conventional clock tree methods.

Chapter 4 provides an explanation about the effect of crosstalk plays an important role in the signal integrity of the design in nanometer technologies. Crosstalk analysis, namely glitch analysis and crosstalk analysis. These techniques are used to make the ASIC behave robustly from a timing perspective.

Chapter 5 This chapter gives an overview and highlights the benefits of Clock buffer compared to Normal buffer to achieve high performance with equal rise and fall delay. Also compared the results of output transition for Clock buffer and Clock inverter for given input transition and output load considering RC impact.

Chapter 6 Conclusion and Future Work





# Chapter 2

## Literature Survey

The literature survey focuses on robust clock tree distribution and to analyze and enhances the system from the noises such as crosstalk noise.

### 2.1 Clock Mesh Variation Robustness: Benefits and Analysis

*Clock Mesh Variation Robustness: Benefits and Analysis by Mallik Devulapalli and Yuichi Kawahara*

Up to now, there have been two main methods of clock distribution for large, high-performance designs: conventional clock-tree synthesis (CTS) and clock mesh. This explains the differences between CTS and clock-mesh distribution technologies.

#### 2.1.1 Key Differences Between CTS And Clock Mesh

There are three key differences between conventional CTS and clock mesh: shared path, design complexity, and timing analysis. Each subsequent section discusses each of the two clock distribution methods with respect to these key differences. At the completion of the chapter, you will know the differences and be better equipped to try a new method that may be better suited to your next design start.

## Amount Of Shared Path

The most obvious difference is the structural depth of the shared path between the clock root and the sinks. Consider an example of the same set of sinks addressed by each of the two clock distribution methods as shown in fig2.1.1.



Figure 2.1.1: Set of sinks addressed by each of the two clock distribution methods[4]

A conventional clock tree, shown at left, is characterized by an organic tree structure from the clock root that branches out to each of the sinks in the design. There is unlimited depth for both buffer and clock-gating levels. Most of the sinks in the design share very few paths back to the clock root so few, in fact, that for any two sinks in the design, the only reliably shared part of the path is the root buffer.

Clock mesh, shown on the right, is characterized by an extremely shallow logic depth below the mesh, usually just a single buffer or clock gate directly driving the sinks. Most of the insertion delay in a clock mesh design is a large, shared path from the root to the mesh.

## OCV Benefit Of Shared Clock Path

The respective logic depths (unlimited, moderate, and very shallow) are inversely related to the level of shared path between the sinks and the clock root. Path sharing reduces the impact of on-chip variation (OCV) effects on the design because when the sinks share the same clock path to the root, any process-variation occurrence in that path affects both flops equally and all

timing assumptions are preserved. In the absence of path sharing, one must increase the clock margin by a derating factor to account for the possibility that either or both the launch and capture flip flops experience a process-variation phenomenon.

We may define the extra margin by multiplying the insertion delay of the non-shared path by a derating scalar, typically between 7% and 10%. Worse yet, it is applied in a range of plus or minus the derating factor. We then add the product to the timed skew of the design and derate the clock-frequency performance of the design.

The current technology nodes encourage large designs with many different functions. As designs grow larger, the impact of OCV derating increases. Of the three clock-distribution methods, conventional CTS is the most adversely affected by OCV derating, and the growing trend is to move away from conventional CTS for high-speed designs.

On the other extreme, the sinks in a clock-mesh design share the overwhelming majority of total clock path. The result is that the measured clock skew increases very little due to OCV derating, preserving the high performance of the design. This is the main reason that clock-mesh design has long been the preferred clockdistribution method deployed by performance-oriented processor designs, whether arithmetic and logical units (ALUs) or graphical processing units (GPUs).

### **Power Tradeoff Differences**

Clock mesh consumes between 20% and 40% more power than the same design implemented with conventional CTS

### **Design Complexity**

The third area of difference among these methods is how the complexity of the clock-gating plan and the floorplan influences the effectiveness of the clock-distribution approach. Conventional CTS is the most accommodating approach for dealing with design complexity.

Clock mesh is the most rigid of the three approaches. An ideal clock mesh design has no RAMs, ROMs, or other hard blocks. Indeed, it is a flat sea of gates. This is ideal for clock mesh because there are no obstructions that prevent the placement of pre-mesh H-Tree buffers such that each H is ideal. The lack of obstructions also enables the H-Tree routes to be perfectly straight, making it easier to ensure an ideal balanced H-Tree. Clock mesh also benefits from a shallow, uniform design topology below the mesh fabric to comply with the limit of two levels of clock buffers or clock gating.

## **Timing Analysis**

In conventional CTS, we perform timing analysis with standard timing analysis tools, both the accepted signoff static timing engines and the similar timing engines embedded within the place and route tools. This makes conventional CTS the easiest method to time through every stage of the flow

In the mesh topologies, circuit simulation is required to time the multiply driven mesh fabrics. This adds a level of complexity to the clock mesh and multisource flows that may at first seem prohibitive. However, the standard is for automation within the place and route tool to launch the simulation run and then annotate the timing values onto the design for subsequent static timing reports and analyses. While this mitigates the circuitsimulation learning curve somewhat, it cannot completely obviate some exposure to the underlying simulator technology.

## **2.2 Crosstalk Delay Analysis**

*Static Timing Analysis for Nanometer Designs - A Practical Approach by J. Bhasker, Rakesh Chadha*

The capacitance extraction for a typical net in a nanometer design consists of contributions from many neighboring conductors. Some of these are grounded capacitances while many others are from traces which are part of other signal nets. The grounded as well as inter-signal capacitances are illustrated in fig 2.2.1. All of these capacitances are considered as part of the total net capacitance during the basic delay calculation (without considering any crosstalk). When the neighboring nets are steady (or not switching), the inter-signal capacitances can be treated as grounded. When a neighboring net is switching, the charging current through the coupling capacitance impacts the timing of the net. The equivalent capacitance seen from a net can be larger or smaller based upon the direction of the aggressor net switching. This is explained in a simple example below.

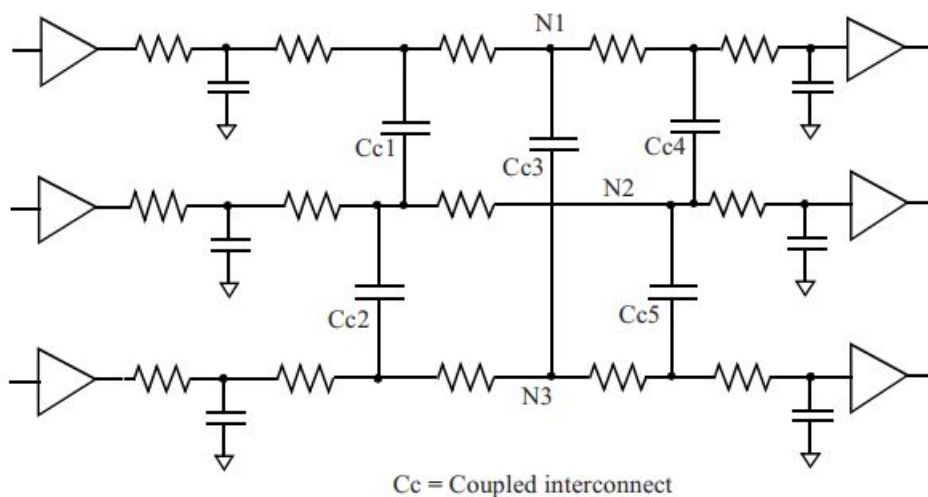


Figure 2.2.1: Example of coupled interconnect[9]

Fig 2.2.2 shows net N1 which has a coupling capacitance  $C_c$  to a neighboring net (labeled Aggressor) and a capacitance  $C_g$  to ground. This example assumes that the net N1 has a rising transition at the output and considers different scenarios depending on whether or not the aggressor net is switching at the same time.

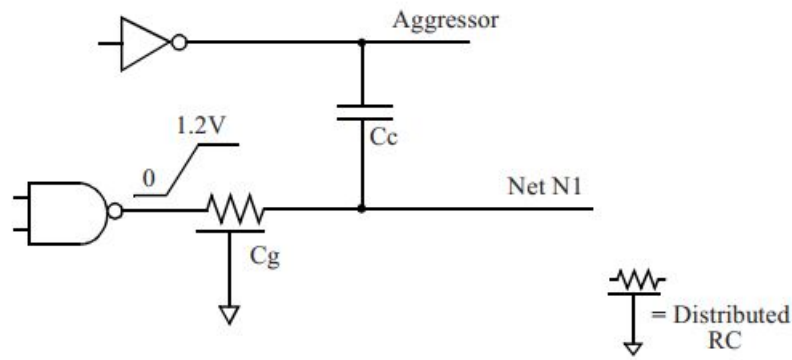


Figure 2.2.2: Crosstalk impact example[9]

## 2.2.1 Timing effect of Crosstalk Delay Violations

This section deals with various timing issues that are caused by crosstalk. Each issue is described in detail with its cause and effect.

### Hold violations

Hold violations are possible at sequential elements in the design when the data input does not respect the minimum required hold timing margin. Usually, clock networks are highly susceptible to the crosstalk issue. This is because; they are widely spread across the chip to reach all sequential elements of the design.

In the case of the sample design experiments conducted, the largest effect of crosstalk was the hold timing violations. This primarily happened because one of the clock networks in the design became the victim of a fast switching aggressor. This is depicted in the fig 2.2.3. The clock network has large coupling with another wire that is driven by a large drive strength buffer. The clock network hence becomes the victim of this aggressor as shown.

When the aggressor switches in the opposite direction of the clock, clock transitions become little slower. So, the clock transitions reach at the flip-flops little later than they should.

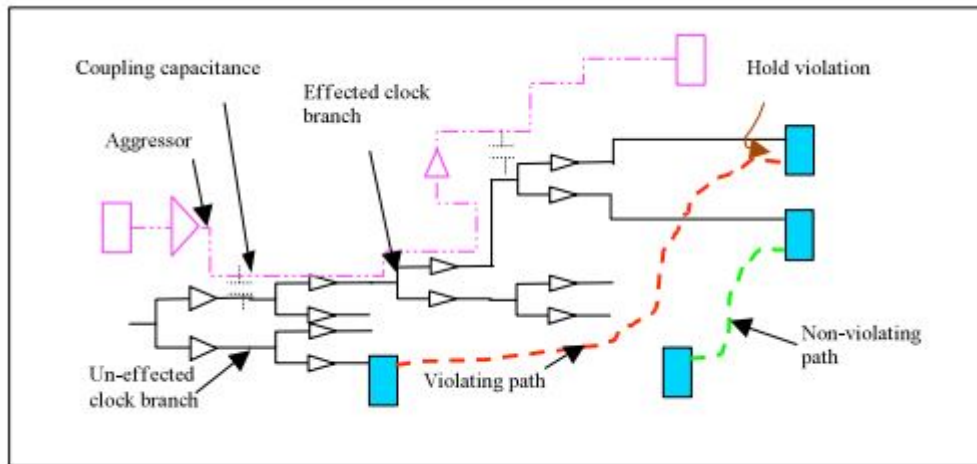


Figure 2.2.3: Hold Violations due to Crosstalk Effect

Because of this, during hold time analysis some of the timing paths, which use this clock as a capture clock might start failing due to the later arrival of the clock. (Not all paths would show violations, because the same clock might also be used as launch clock.)

### Setup violations

Similarly, setup violations are possible at sequential elements when the data inputs do not honor the setup time requirement of the sequential element. Though less number, there are setup violations observed during the crosstalk delay analysis for the experimental design taken. The root cause of these setup violations is explained in the following fig 2.2.4.

As shown in the fig2.2.4, a timing path exists between FF1 and FF2. There is a neighboring aggressor path, as shown between A and B. When the aggressor wire is switching in the opposite direction of the signal in the data path (victim), the data input of FF2 could be delayed. Because of this, a timing path that was meeting timing without crosstalk analysis would now show a violation. This is shown in the fig 2.2.5.

As shown in the fig 2.2.5, originally, the actual arrival time of the signal at the input of

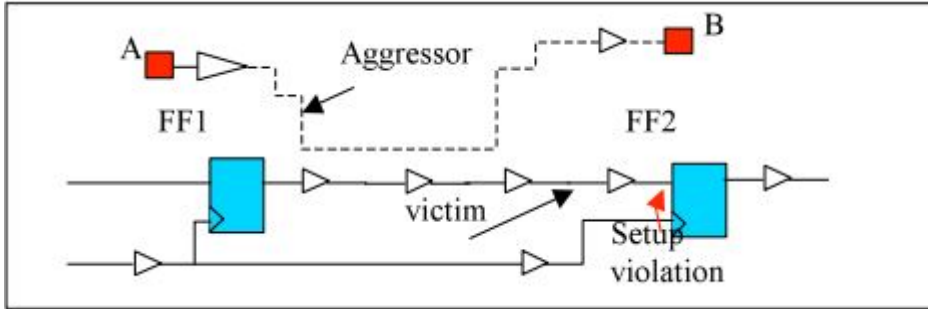


Figure 2.2.4: Setup Violations due to Crosstalk Effect

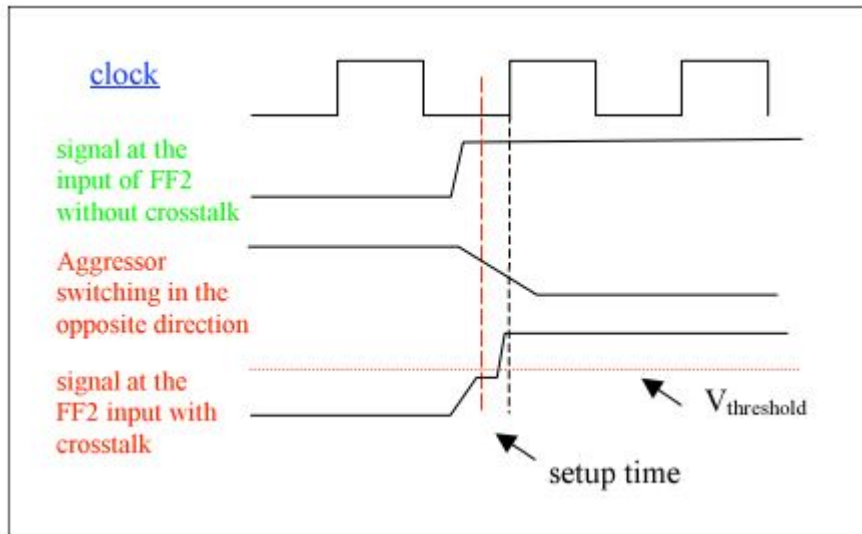


Figure 2.2.5: Setup Violation due to Crosstalk Delay

FF2 is well ahead of setup requirement of the flip-flop. If the aggressor switches in the opposite direction, the signal is delayed. The flip-flop FF2 now has a setup violation after considering the effect of crosstalk.



# Chapter 3

## CTS variation robustness

Circuit delay is increasingly affected by process variations at lower technology nodes. Global variations are in double digits now, and according to the International Technology Roadmap for Semiconductors (ITRS) the trend is rising. Variations in the manufacturing process may cause two gates that are electrically identical and in close proximity to significantly vary in delay. Consequently, designers add significant timing margin to safeguard their designs against timing violations. Technologies that offer variation tolerance boost design performance and productivity. Clock mesh technology provides uniform, low skew clock distribution and offers better tolerance to on-chip variations (OCV) than conventional clock tree technology. The need to control OCV effects is now driving clock mesh technology to mainstream designs.

### 3.1 Variation at Advanced Technology Nodes

There are two source classes of variation that must be considered in design, global and local. Global chip-to-chip variations cause performance differences among dies and are modeled as operating corners. Local on-chip variations cause performance differences among transistors within the same die and are modeled as an added derating factor to skew calculations. What are the specific causes of these local variations ?

Transistors located in close proximity on the same chip exhibit variation in their characteristics due to random manufacturing variations in :

1. the number and location of doping atoms
2. the length and width of the transistor channel
3. the thickness of oxide layers across the die

Timing derating is the universally accepted method to model the maximum OCV that the design is expected to incur. Newer technology nodes feature increased gate speeds as well as increased susceptibility to variation. Because of this, the derating factor has also increased, and today it is common to see derating between 5 percent and 10 percent. Thus, it becomes necessary to design circuit structures that are inherently variation tolerant to reduce the adverse impact of OCV derating.

Clock mesh is a clocking scheme employed by high-performance design teams to achieve low skew and high OCV tolerance. The large impact of OCV derating on conventional clock trees motivates mainstream design groups to also consider clock mesh. An examination of clocking structures explains why.

### **3.1.1 Clock Mesh versus Conventional Clock Tree Structure**

The structures of a conventional clock tree and a clock mesh are shown in figure 3.1.1. The clock tree has a clock source, clock tree cells, clock gating cells and buffers and loads. The clock mesh includes a clock source, pre-mesh drivers, mesh drivers, the mesh net, clock gates and mesh receivers, and loads.

The main difference is the presence of the mesh net. Another major difference is that the mesh drivers are connected to the mesh net as a multi-driven net. Clock mesh implementation requires an array of mesh drivers, shown in green in figure 3.1.1, to drive the massive RC network of the clock mesh.

The benefit of the mesh net is that it smoothes out the arrival time differences from the multiple mesh drivers that drive it. The smoothing effect of the mesh net is visualized with

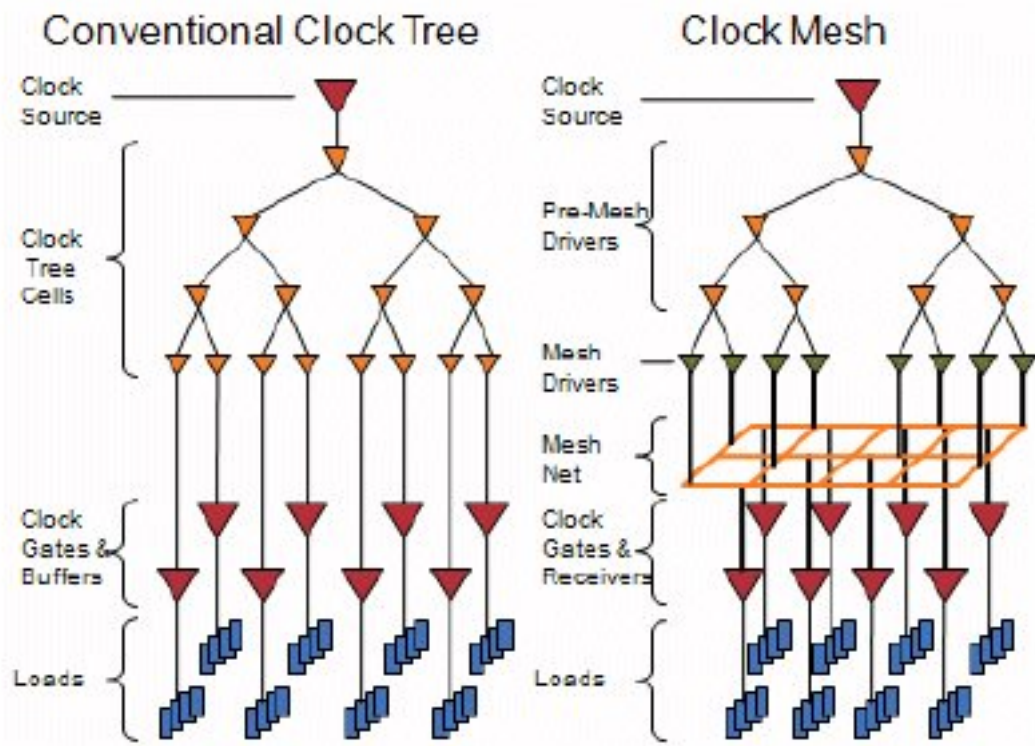


Figure 3.1.1: Clock Structures - Conventional clock tree and clock mesh[4]

circuit simulation of an actual test case shown in Figure 3.1.2. The top trace is the ideal clock, the top pair of traces shows the skew just before the mesh, and the bottom pair of traces shows the skew just after the mesh.

These timing waveforms show that the mesh receivers are switching in a very narrow timing window compared to the mesh drivers. In the analysis section, Monte Carlo simulation is used to validate over randomly varied conditions showing that the range of skews at the output

Table 3.1: Insertion delay and skew of clock tree and clock mesh

	<b>CTS</b>	<b>CTS</b>	<b>Clock mesh</b>	<b>Clock mesh</b>
-	Skew to mesh receivers	Skew to registers	Skew to mesh receivers	Skew to registers
Design1	-	323ps	14ps	136ps
Dsign2	-	81ps	14ps	28ps

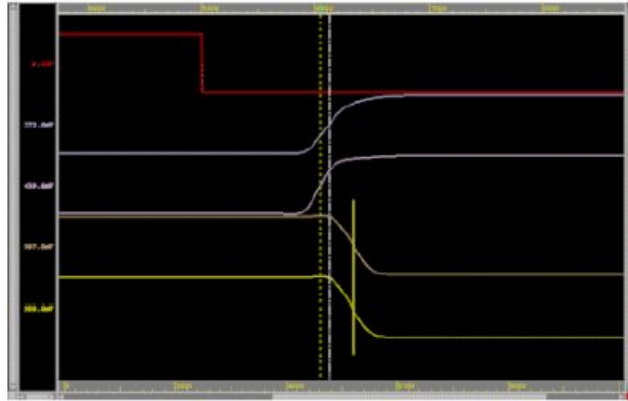


Figure 3.1.2: SPICE waveforms showing the smoothing effect of the mesh net

of the mesh is narrow compared to the range at the input of the mesh.

## 3.2 Understanding OCV Derating

For the majority of loads in a clock tree design, very little of the overall path back to the clock root is shared. The converse is true for a clock mesh design where the path from the clock root to the mesh net is shared by all loads. Thus only the paths from the mesh net through the clock gates and receivers to the loads are adversely impacted by variation effects. The variation above the mesh net is negligible.

OCV derating values range from 5 percent to 10 percent depending on the technology node and design knowledge. A typical derating factor is 7 percent. Thus, for setup-checks, the non-shared launch path is increased by 7 percent, and the non-shared capture path is reduced by 7 percent. In Figure 3.2.1, the skew is assumed to be 2 percent of the total insertion delay for both design styles. This is unlikely to occur in practice since clock mesh designs yield much better skew, but holding skew constant highlights the impact of OCV derating. Even this unrealistically conservative example shows that clock mesh has almost four times better OCV tolerance.

Since OCV derating only occurs between the unique portions of the launch and capture paths, the benefit of clock mesh OCV immunity is significant - in this example four times better.

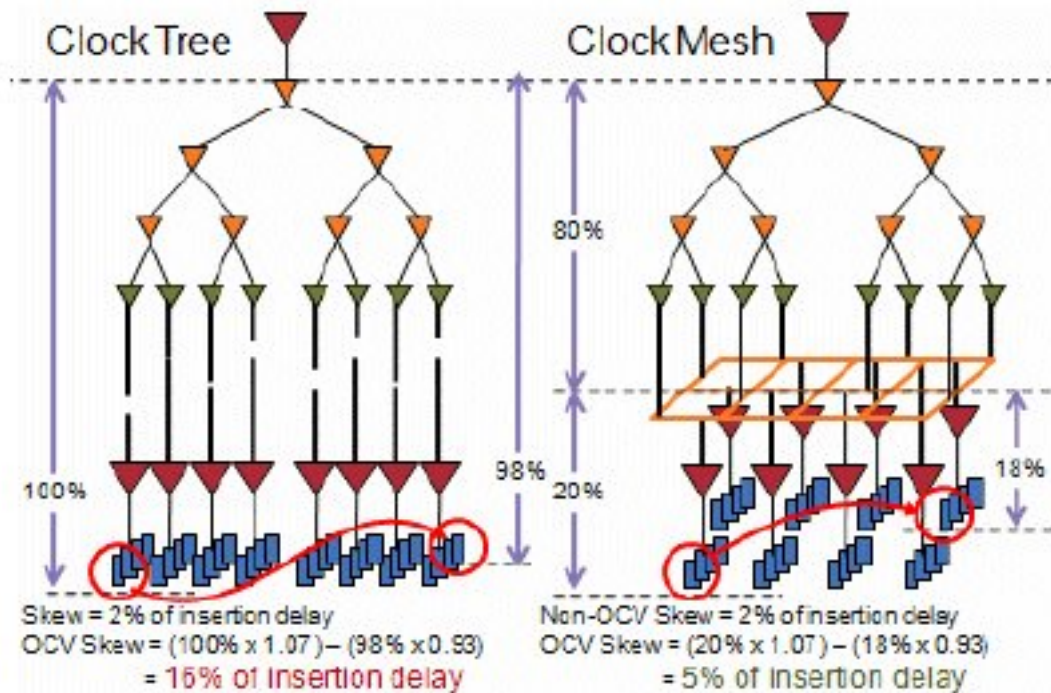


Figure 3.2.1: OCV tolerance - clock tree vs clock mesh[4]

Per ITRS variation effects increase as feature sizes decrease. As the adverse impact of OCV continues to increase, the benefits of clock mesh over clock tree become even more pronounced.

### 3.2.1 Simulation and Analysis of OCV Effects

Monte Carlo simulation is a method of applying random variations to simulate the manufacturing process. Varying the SPICE lint parameter of the NMOS/PMOS transistor model emulates the random variation of doping atom deposition in the transistors channel. During the Monte Carlo simulation, a different random value is produced for each transistor in the netlist. A Gaussian variation with zero mean and a sigma equal to  $1e-9$  is modeled. The lint parameter is varied in the range from -4 to +4 sigma, (-4nm to +4nm). Thus if the drawn length is 22nm, and the nominal lint is 0, the effective length of the device varies between 18nm and 26nm.

Fig 3.2.2 shows the results of a Monte Carlo simulation over 30 iterations. The skew variations at the receiver inputs, shown in blue, demonstrate a very small skew variation from the

clock source to the mesh receivers.

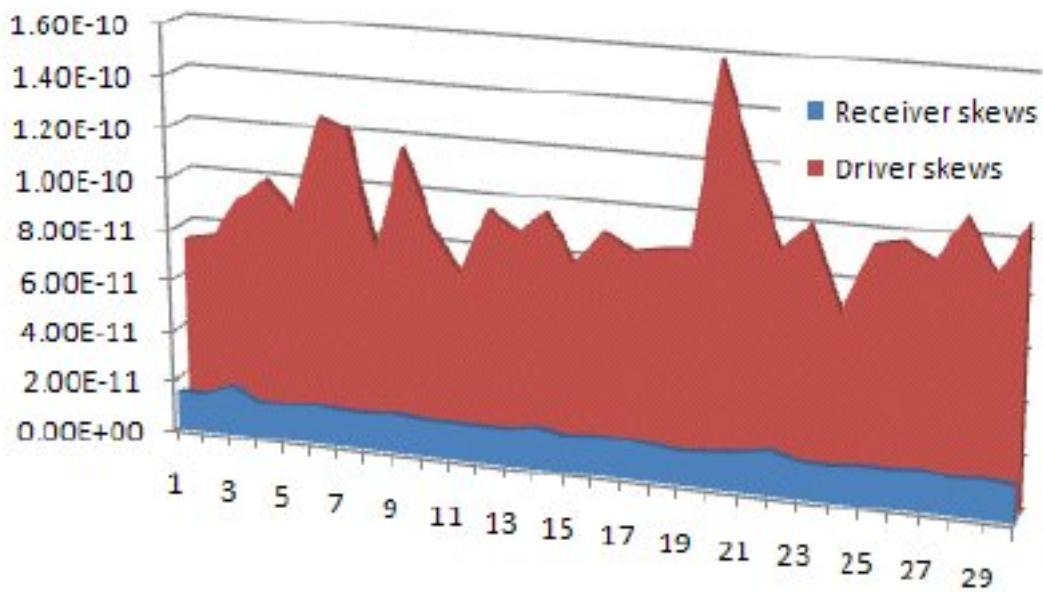


Figure 3.2.2: Monte Carlo simulation results[5]

The results shown in red are the skew variations at the mesh driver input pins. The skew variations before the mesh net are extremely large (60ps to 160ps), but the skew variations after the mesh net (14ps to 16ps) are small and validate that the multi-driven mesh net equalizes the delay.

Circuit simulation testing determines the optimum mesh spine width and pitch for a given drive capability. Monte Carlo SPICE simulation testing validates that the clock mesh produces low clock skew and has a strong immunity to on-chip variation.

# Chapter 4

## Crosstalk Delay Analysis

Crosstalk is the undesirable electrical interaction between two or more physically adjacent nets due to capacitive cross-coupling. As integrated circuit technologies advance toward smaller geometries, crosstalk effects become increasingly important compared to cell delays and net delays. Signal integrity is the ability of an electrical signal to carry information reliably and resist the effects of high-frequency electromagnetic interference from nearby signals.

As circuit geometries become smaller, wire interconnections become closer together and taller, thus increasing the cross-coupling capacitance between nets. At the same time, parasitic capacitance to the substrate becomes less as interconnections become narrower, and cell delays are reduced as transistors become smaller.

### 4.1 Crosstalk Delay Effects

Crosstalk can affect signal delays by changing the times at which signal transitions occur. For example, Figure 4.1.1 shows the signal waveforms on cross-coupled nets A, B, and C.

Because of capacitive cross-coupling, the transitions on net A and net C can affect the time at which the transition occurs on net B. A rising-edge transition on net A at the time shown in Figure 4.1.1 can cause the transition to occur later on net B, possibly contributing to a setup violation for a path containing B. Similarly, a falling-edge transition on net C can cause the

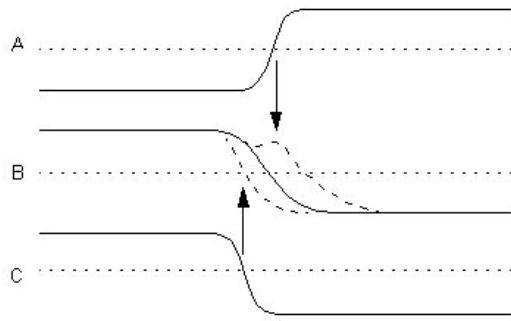


Figure 4.1.1: Transition slowdown or speedup caused by crosstalk

transition to occur earlier on net B, possibly contributing to a hold violation for a path containing B.

### 4.1.1 Delta Delay and Fanout Stage Effect

Crosstalk effects distort a switching waveform, which adds delay to the propagated waveforms of the fanout stages, as shown in Figure 4.1.2

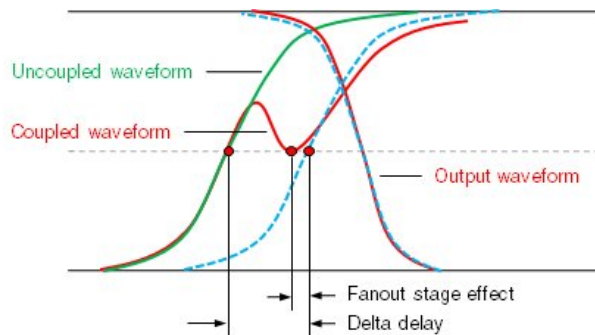


Figure 4.1.2: Delta delay includes the fanout stage effect

### 4.1.2 Aggressor and Victim Nets

A net that receives undesirable cross-coupling effects from a nearby net is called a victim net. A net that causes these effects in a victim net is called an aggressor net. Note that an aggressor net can itself be a victim net; and a victim net can also be an aggressor net. The terms



aggressor and victim refer to the relationship between two nets being analyzed.

The timing effect of an aggressor net on a victim net depends on several factors:

1. The amount of cross-coupled capacitance
2. The relative times and slew rates of the signal transitions
3. The switching directions (rising, falling)
4. The combination of effects from multiple aggressor nets on a single victim net

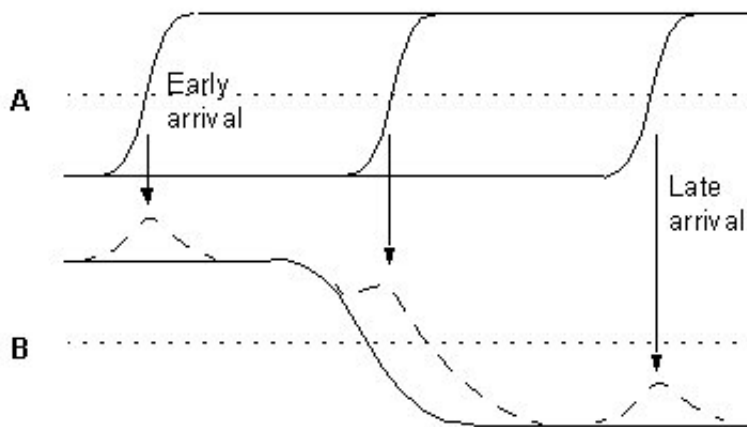


Figure 4.1.3: Transition slowdown or speedup caused by crosstalk

As shown in Figure 4.1.3, if the transition on A occurs at about the same time as the transition on B, it could cause the transition on B to occur later, possibly contributing to a setup violation; otherwise, it could cause the transition to occur earlier, possibly contributing to a hold violation.

If the transition on A occurs at an early time, it induces an upward bump or glitch on net B before the transition on B, which has no effect on the timing of signal B. However, a sufficiently large bump can cause unintended current flow by forward-biasing a pass transistor.

Similarly, if the transition on A occurs at a late time, it induces a bump on B after the transition on B, also with no effect on the timing of signal B. However, a sufficiently large bump can cause a change in the logic value of the net, which can be propagated down the timing path. Tool reports occurrences of bumps that cause incorrect logic values to be propagated.

## 4.2 Crosstalk effect on fanout

Crosstalk causes distortions in the switching waveforms and affects the delay of the victim stage and its fanouts. The circuit example in Figure 4.2.1 has a victim net with a single aggressor. Because of cross-coupling between the aggressor net and the victim net, the switching waveform at the input pin of the victim receiver is distorted. This distorted coupled waveform affects the delay of the victim net and the receiver stage. PrimeTime models the effect of the distorted coupled waveform as delta delay at the victim stage.

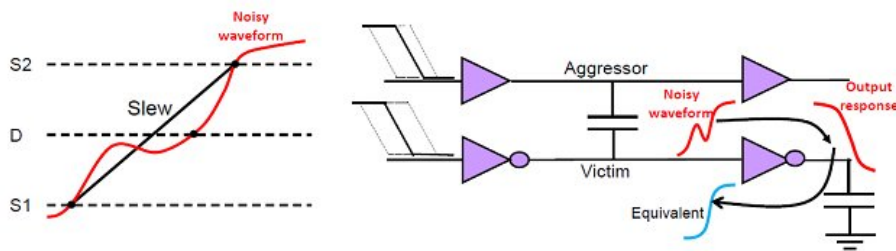


Figure 4.2.1: Crosstalk effect modeled as delta delay for current stage and fanout

## 4.3 Experimental Results

In this section, we will see experimental results that verify the accuracy and effectiveness of our proposed approach for different cases as shown in below figure 4.3.1.

In order to confirm the importance of considering coupling noise in our analysis, we find the best common interconnect cases that can be constrained before routing metal layers.

- Case1 : Three consecutive nets switching in same direction with minimum spacing.

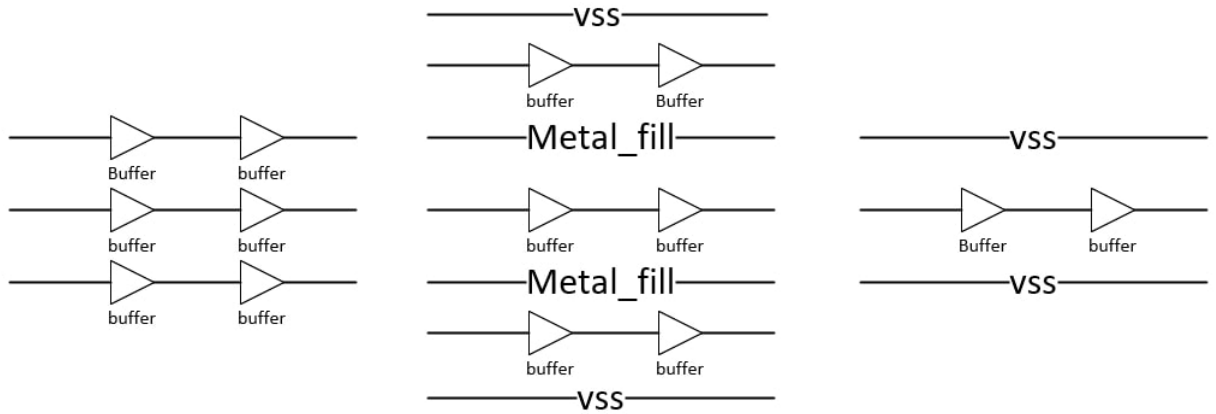


Figure 4.3.1: Different cases for proposed approach

- Case2 : Three consecutive nets switching in same direction with double spacing, shielded upper and lower net by ground.
- Case3 : One net shielded by ground.

**Result 1** : Stage delay and crosstalk delay on victim net are calculated for different cases having length 200 micron for metal m3 for different sizes of buffer as shown in below table 4.3.2.

name	2x5	8x5	16x5	32x5	64x5
stage_delay_case1	266	124	105	93	87
stage_delay_case2	113	71	63	58	58
stage_delay_case3	103	68	61	55	55
crosstalk_delay_case1	125	42	33	27	21
crosstalk_delay_case2	13	5	4	4	4
crosstalk_delay_case3	0	0	0	0	0

Figure 4.3.2: Results for different cases having length of 200 micron for metal m3

As shown in below graph 4.3.3

1. X-axis : buffer of different sizes
2. Y-axis(primary) : stage delay on victim net

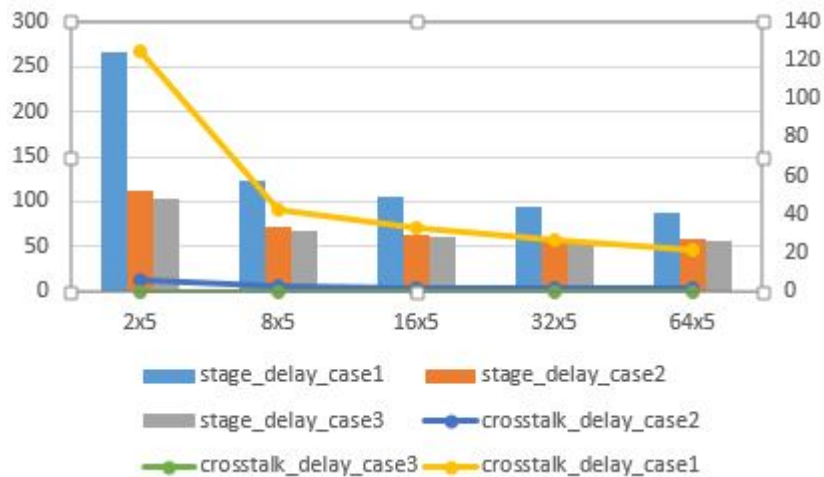


Figure 4.3.3: Graph of stagedelay and crosstalk delay versus different sizes for different cases

3. Y-axis(secondary) : crosstalk delay on victim net

As shown in above figure 4.3.3. In bar graph, different colour represented as :

1. blue colour : Case1 (Three consecutive nets switching in same direction with minimum spacing)
2. orange colour : Case2 (Three consecutive nets switching in same direction with double spacing, shielded upper and lower net by ground)
3. grey colour : Case3 (One net shielded by ground)

From the above graph 4.3.3, it was concluded that crosstalk for Case1 is very high for all different sizes of buffer, but it decreases as buffer size increases for that case. while for Case2, crosstalk decreases large as spacing between nets is doubled. For Case3, crosstalk has no impact due to the net shielded by ground on both sides. Thus effect of crosstalk reduces as spacing between net increases and driver strength increases upto certain level. Crosstalk also largely depend on the width of the metal, as metal width increases crosstalk also increases.

**Result 2** : stage delay and crosstalk delay on victim net are calculated for different cases having length 250 micron for metal m3 for different sizes of buffer as shown in below table 4.3.4.

name	2x5	8x5	16x5	32x5	64x5
stage_del	361	171	143.35	124.71	116.1
stage_del	147	86	77.22	71.75	71.69
stage_del	139	82	73.38	67.32	67.08
crosstalk_	178	70.480827	53.82562	42.136372	33.9
crosstalk_	19	7.80324	6.73939	6.956734	6.96
crosstalk_	0	0	0	0	0

Figure 4.3.4: Results for different cases having length of 250 micron for metal m3



Figure 4.3.5: Graph of stagedelay and crosstalk delay versus different sizes for different cases having length of 250 micron for metal m3

By comparing both graphs and results, it was concluded that crosstalk for second results is more as compared to the first result due to increase in length which also increases cross coupling cap.

**Result 3 :** stage delay and crosstalk delay on victim net are calculated for different cases having length 200 micron for metal m5 and m7 for different sizes of buffer as shown in below table 4.3.6.

metal	start_po	stage_02x5	cross_2x	stage_8x5	cross_8x5	stage_16x5	cross_16x5	stage_32x5	cross_32x5	stage_64x5	cross_64x5
m5	caseA_n	266.57	124.955696	124.28	42.319534	105.35	33.054337	93.03	27.142103	87.16	21.970757
m5	caseB_n	113.01	13.165759	71.19	4.826613	63.84	4.255692	58.35	4.045128	58.04	4.074451
m5	caseC_n	103.27	0	68.68	0	61.34	0	55.75	0	55.31	0
m7	caseA_n	358.68	190.107162	126.54	48.402977	91.81	27.367838	74.75	20.422466	67.63	17.968004
m7	caseB_n	144.32	17.284712	73.62	4.643559	61.26	3.282256	52.63	2.698506	48.7	2.526204
m7	caseC_n	130.08	0	69.7	0	58.42	0	50.2	0	47.38	0

Figure 4.3.6: Results for different cases having length of 200 micron for metal m5 and m7

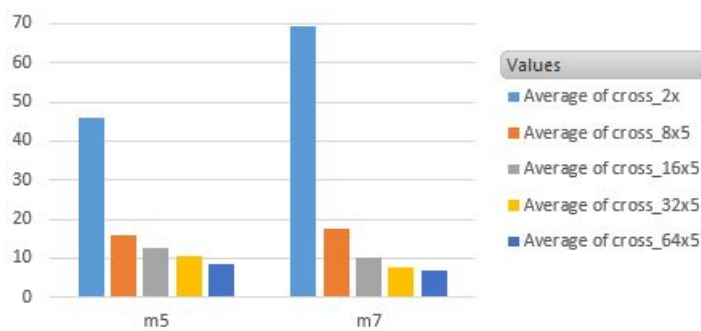


Figure 4.3.7: Graph of stagedelay and crosstalk delay versus different sizes for different cases having length of 200 micron for metal m3 and m5

From above graph 4.3.7, it was concluded that crosstalk increase with increases in metal width as going from metal m5 to m7.

# Chapter 5

## Clock Inverter Vs Clock Buffer based Clock Tree

Clock tree synthesis (CTS) plays an important role in building well-balanced clock tree, fixing timing violations and reducing the extra unnecessary pessimism in the design. The goal during building a clock tree is to reduce the skew, maintain symmetrical clock tree structure and to cover all the registers in the design.

The normal inverters and buffers are not used for building and balancing because, the clock buffers provides a better slew and better drive capability when compared to normal buffers and clock inverters provides a better balance with rise and fall times and hence maintaining the 50% duty cycle. Clock tree can be build by clock tree inverters so as to maintain the exact transition (duty cycle) and clock tree balancing is done by clock tree buffers (CTB) to meet the skew and latency requirements.

### 5.1 Clock Buffer Vs Normal Buffer

Advatages of Clock Buffer over Clock inverter.

1. Clock buffer have equal rise time and fall time compared to Normal buffers, therefore

pulse width violation is avoided.

2. Clock buffers are usually designed such that an input signal with 50% duty cycle produces an output with 50% duty cycle. This usually isn't true for a normal buffer.
3. In Clock buffers Beta ratio is adjusted such that rise & fall time are matched by increasing size compared to normal buffer.
4. Clock net is one of the High Fanout Net(HFN)s. The clock buffers are designed with some special property like high drive strength and less delay. Clock buffers have equal rise and fall time. This prevents duty cycle of clock signal from changing when it passes through a chain of clock buffers.

### **5.1.1 Min Pulse Width**

Min pulse width check is to ensure that pulse width of clock signal is more than required value for proper performance of clock. Basically it is based on frequency of operation and Technology. Means if frequency of design is 1Ghz than typical value of each high and low pulse width will be equal to  $(1\text{ns}/2)$  0.5ns if duty cycle is 50%.

Normally we see that in most of design duty cycle always keep 50% otherwise designer can face issues like clock distortion and if in our design using half cycle path means data launch at +ve edge and capturing at -ve edge and again min pulse width as rise level and fall level will not be same and if lots of buffer and inverter will be in chain than it is possible that pulse can be completely vanish.

Also we have to consider the best and worst case when clock get routed and depend on that decide that what should be the required value of Min Pulse Width. Now we know that rise delay and fall delay of combinational cells are not equal so if a clock entering in a buffer than the output of clock pulse width will be separate to input.



For example as shown in above Figure 5.1.1, if buffer rise delay is more than fall delay than output of clock pulse width for high level will be less than input.

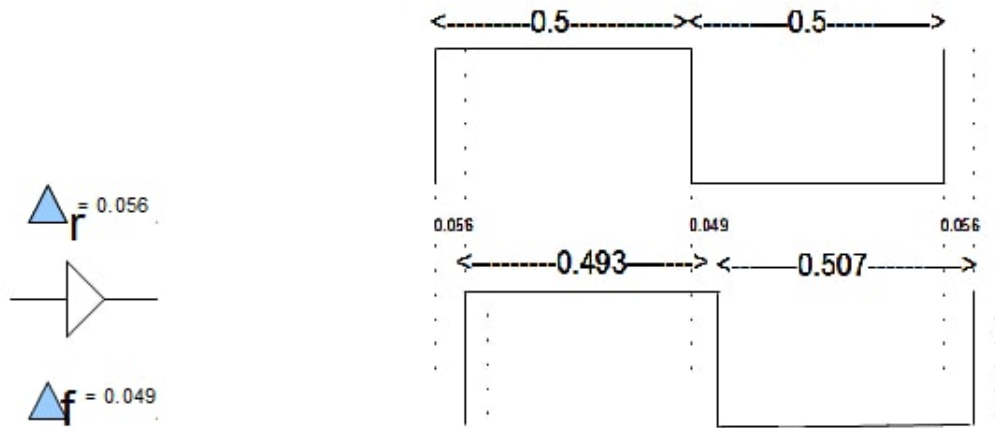


Figure 5.1.1: Pulse width for normal buffer with different rise and fall delay[10]

- High pulse :  $0.5 - 0.056 + 0.049 = 0.493$
- Low pulse :  $0.5 - 0.049 + 0.056 = 0.50$

Another real scenario example as shown in below Figure 5.1.2, lets there is a clock signal which is going to clock pin of flop through series of buffers with different rise and fall delay. We can calculate that how it effect to high or low pulse of clock.

- High pulse width =  $0.5 + (0.049 - 0.056) + (0.034 - 0.039) + (0.023 - 0.026) + (0.042 - 0.046) + (0.061 - 0.061) + (0.051 - 0.054) = 0.478\text{ns}$
- Low Pulse width =  $0.5 + (0.056 - 0.049) + (0.038 - 0.034) + (0.026 - 0.023) + (0.046 - 0.042) + (0.061 - 0.061) + (0.054 - 0.051) = 0.522\text{ns}$
- Required value of Min pulse width is 0.420ns.
- Uncertainty = 80ps
- high pulse width =  $0.478 - 0.080 = 0.398\text{ns}$

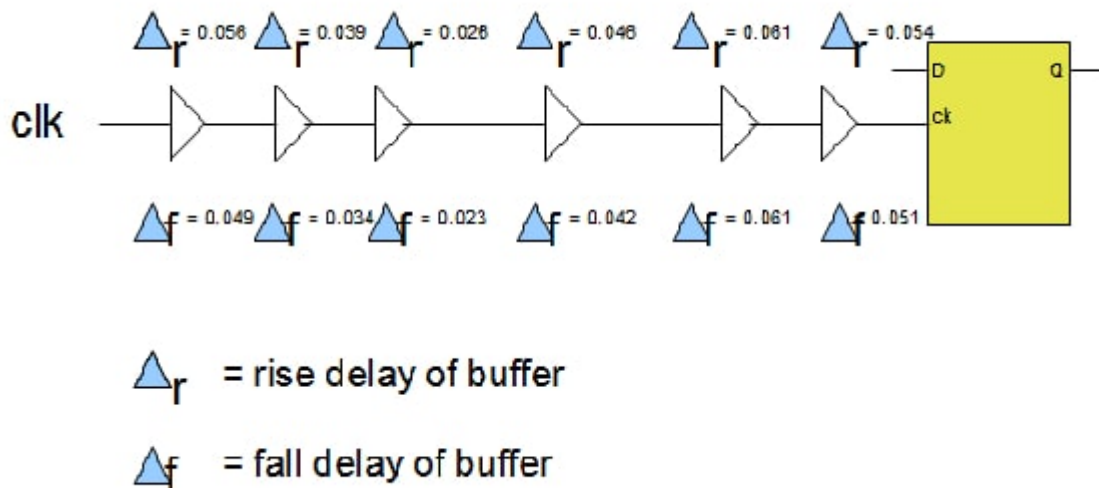


Figure 5.1.2: Pulse width check for clock signal which is going to clock pin of flop through series of buffers with different rise and fall delay

Now we can see that we are getting violation for high pulse as total high pulse width is less than Require value. So for solving this violation we can add an inverter which will change the transition and improve it.

## 5.2 Inverter Based Clock Tree

To keep things simple and pertinent to the discussion, let's assume that we are using only a single kind of inverter (i.e. of let's say drive X) to build our clock trees. And all the inverters are placed equidistant from each other. The scenario is shown in below Fig 5.2.1. Advantage of using an inverter based clock tree is that the high pulse width and the low pulse width would be symmetrical and also it cancels out jitter added due to clock. For the clock signal, this is a critical requirement, especially for SoCs which have a high interaction between the positive and negative edge triggered flip-flops.

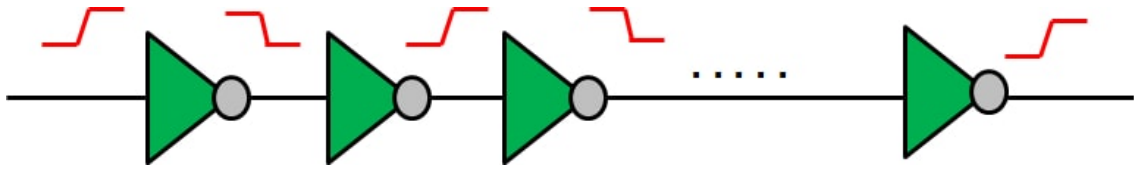


Figure 5.2.1: Inverter Based Clock Tree giving equal rise and fall times

### 5.3 Buffer Based Clock Tree

While theoretically, one can create a buffer using two identical inverters connected back to back, that is generally not the way buffers are designed while designing the standard cell libraries. To save area, the first buffer is typically of a lower drive strength and is placed very close to the second inverter as shown in below figure 5.3.1. The second inverter, however, is of higher drive strength.

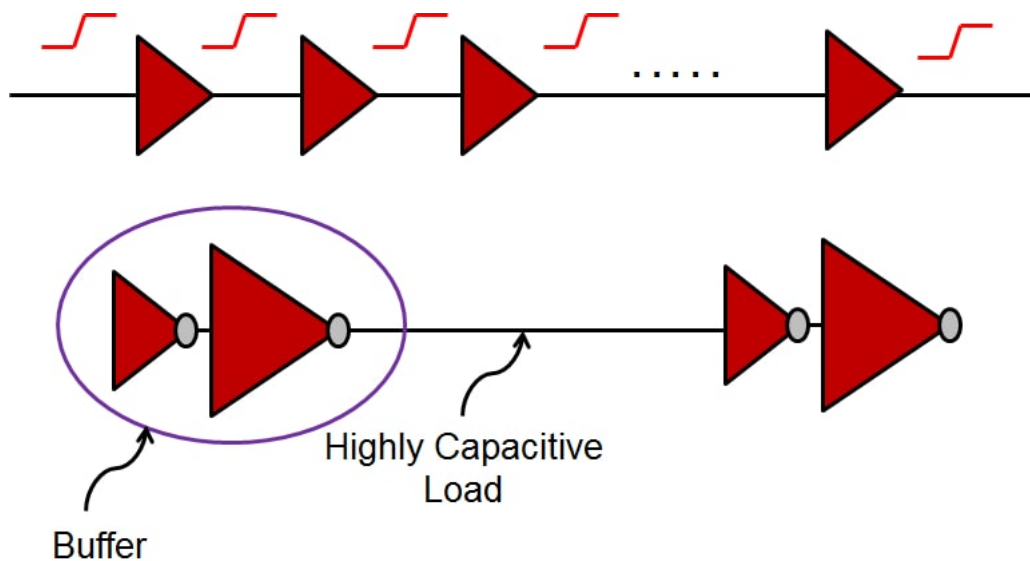


Figure 5.3.1: Buffer Based Clock Tree. Buffer is formed by connecting two invertes back to back

One must also notice that the delay of first inverter is dominated by the load of the second inverter because the wire length between these two inverters is very small, hence one can neglect the wire cap. But for the second inverter, the load comprises of the wire cap as well as the input cap of the next buffer. This introduces an asymmetry in the rise and fall delays, and hence

the high and low pulse widths of the clock signal as shown in below fig 5.3.2.

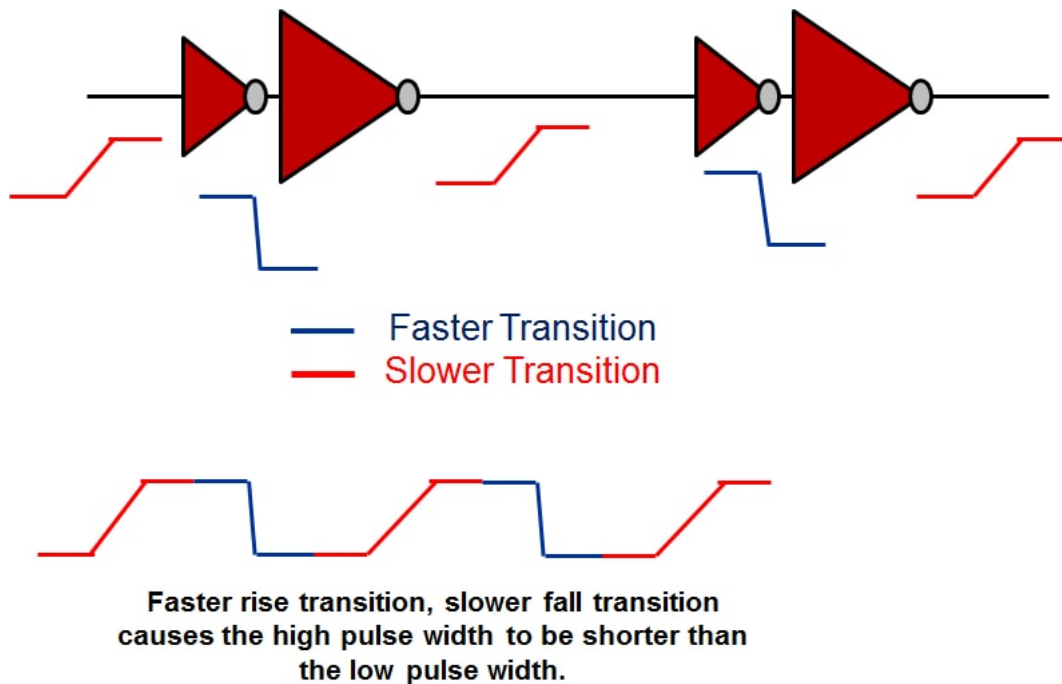


Figure 5.3.2: Difference in high and low pulse width

For applications which have a very stringent requirement on the clock high and low pulse widths, one might prefer to use an inverter based clock tree over the buffer based clock tree. Can we do something to make the buffer based clock tree work? The answer is yes! Let's take a look

If we balance the load seen by first inverter and the load seen by the second inverter, we might be able to achieve equal rise and fall times, and hence equal high and low pulse widths for the clock transition signal. In this approximation, we have modeled the wire in form of a T-model. And inverter is modeled using distributed RC model with its "on" resistance and the diffusion capacitance.

To have the equal pulse widths for high and low times, the RC delay observed by the first inverter must be equal to the RC delay of the second inverter as shown in below fig 5.3.3.

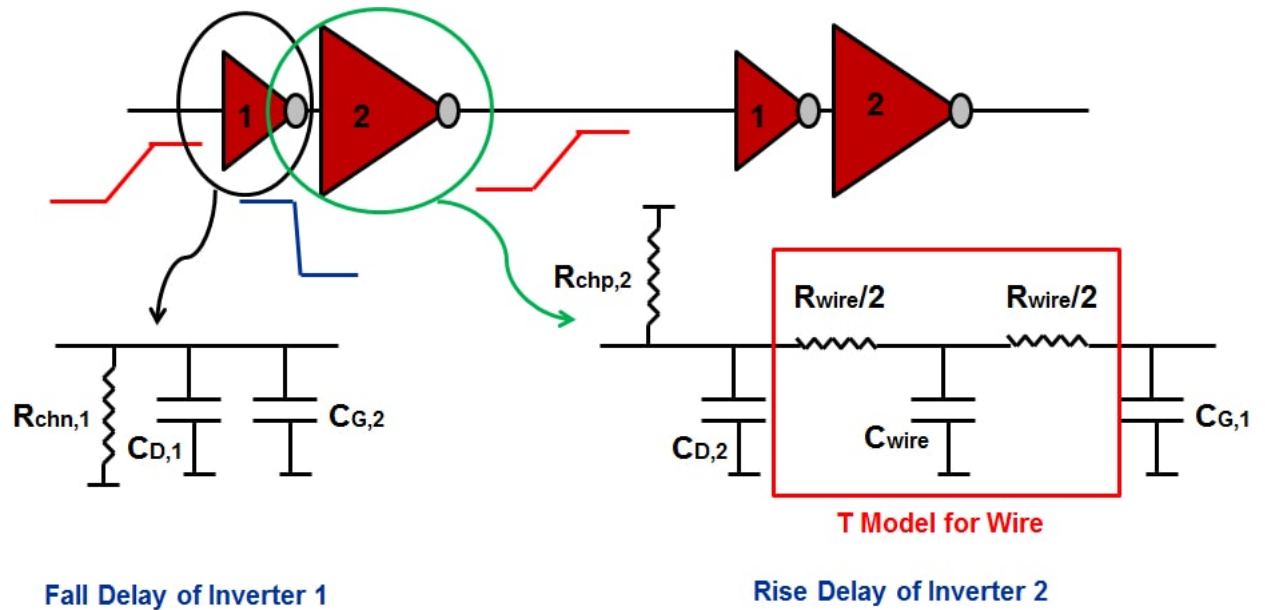


Figure 5.3.3: RC delay model for inverters and wire

$$R_{chn,1} (C_{D,1} + C_{G,2}) = R_{chp,2} (C_{D,2} + C_{wire} + C_{G,1}) + R_{wire}/2 (C_{wire} + C_{G,1}) + R_{wire}/2 (C_{G,1})$$

If this above equation is satisfied, one can say with a fair degree of confidence that the high and low pulse widths would be approximately equal. The resistance and capacitance of the wire is the function of its length and the same can be conveyed by the standard cell library designer to the backend designers.

While most standard cell library vendors provide a symmetrical buffer, there could well be a difference of a few pico-seconds in the buffer rise and fall delay, which creates a difference in the high and low pulse widths. The variation in the duty cycle increases for deeper clock trees.

A simple way to mitigate the problem is to insert an inverter in the middle point of the buffer-cased clock tree. The major challenge, however, lies in finding this middle point. This ensures that high and low pulse widths of the clock reaching at the sink pins of flip-flops is indeed the same.

### 5.3.1 Experimental Results

In this section, we will see experimental results that verify the accuracy and effectiveness of our proposed approach w.r.t clock buffer and clock inverter or different cases as shown in below figure 5.3.4.

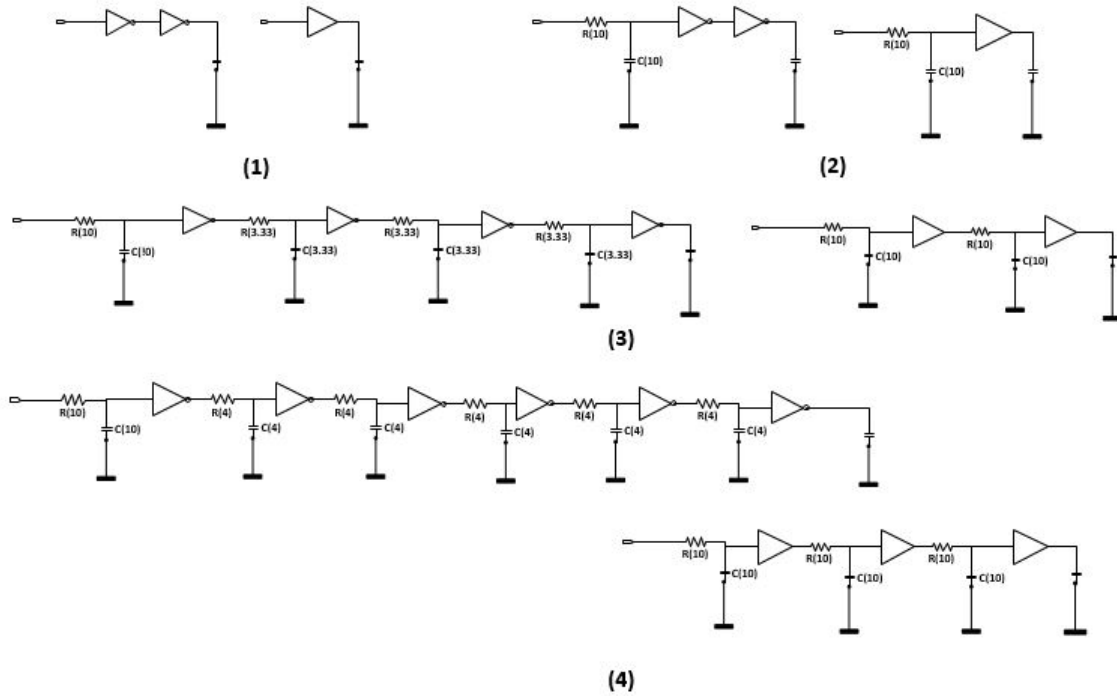


Figure 5.3.4: Different cases for proposed approach

In order to confirm the importance of considering RC impact in our analysis, we find the four different cases of buffers compared with cases of inverters having same output load and input slew.

- Case1 : Two consecutive inverter compares with one buffer without RC at input.
- Case2 : Two consecutive inverter compares with one buffer with RC of 10um length metal m5 at input.
- Case3 : Four consecutive inverter with RC of 3.33um length metal m5 in between compares with two consecutive buffer with RC of 10um length metal m5 in between and RC

Table 5.1: Result for out\_tran & out\_dly for input\_tran 20ps output\_cap 40fF rc\_length 20 um for metal M5.

Type	tr_inv	tr_buf	dly_inv	dly_buff
Case1	23ps	23.1ps	23.1ps	26.4ps
Case2	23.2ps	23.4ps	26.9ps	29ps
Case3	23.4ps	23.3ps	45ps	41.1ps
Case4	23.6ps	23.5ps	57.7ps	53.3ps

\*\*The Numbers shown here are reference numbers, not real\*\*

of 10um length metal m5 at input in both.

- Case4 : Six consecutive inverter with RC of 4um length metal m5 in between compares with three consecutive buffer with RC of 10um length metal m5 in between and RC of 10um length metal m5 at input in both.

**Result 1** : Output transition is calculated for different cases having input transition 20ps, output cap of 40fF RC length and 20u length for metal m5 for different cases of clock buffer and clock inverter as shown in above figure 5.3.5.

As shown in above graph 5.3.5

1. X-axis : different cases of buffer and inverter
2. Y-axis : output transition

As shown in above figure 5.3.5. In bar graph, different colour represented as :

1. blue colour : four different inverter cases
2. orange colour : four different buffer cases

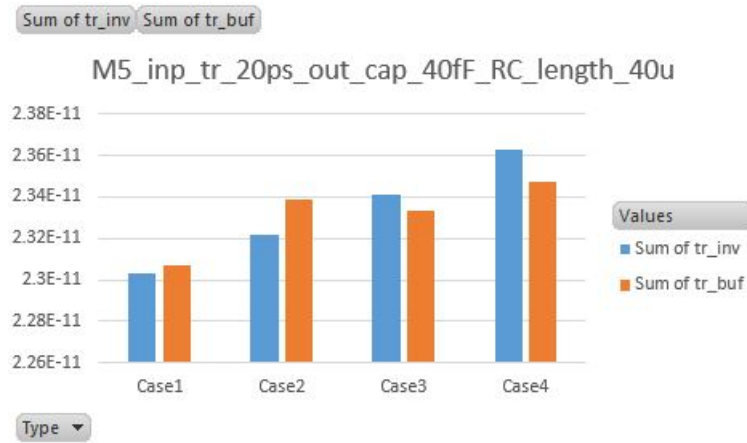


Figure 5.3.5: Result for out\_tran for input\_tran 20ps output\_cap 40fF rc\_length 20 um for metal M5

From the above graph 5.3.5, it was concluded that output transition for buffer Case1 & Case2 is high compared to inverter, but it decreases for buffer Case3 & Case4 compared to inverter. while for buffer Case3 & Case4, Output Transition decreases due to decrease in input cap for buffer as compared to inverter. Buffer is formed by connecting two inverter back to back, such that delay of first inverter is dominated by the load of the second inverter because the wire length between these two inverters is very small, hence one can neglect the wire cap. While if we don't give RC at the input then output transition for inverter Case1 & Case2 is slightly low compared to buffer.

**Result 2 :** Output delay is calculated for different cases having input transition 20ps, output cap of 40fF RC length and 20u length for metal m5 for different cases of buffer and inverters as shown in above figure 5.3.6.

From the above graph 5.3.5, it was concluded that output delay for buffer Case1 & Case2 is high compared to inverter due to less RC applied at input, but it decreases for buffer Case3 & Case4 compared to inverter. while for buffer Case3 & Case4, Output delay decreases due to decrease in input cap and delay of first inverter is dominated by the load of the second inverter because the wire length between these two inverters is very small, hence one can neglect the wire cap for buffer as compared to inverter. While if we don't give RC at the input then output



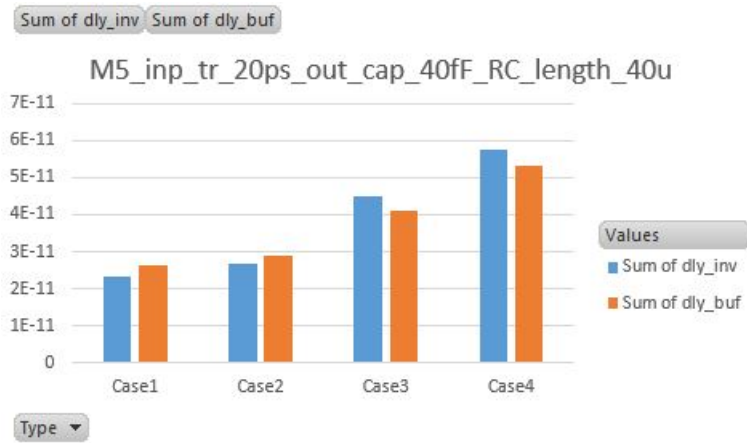


Figure 5.3.6: Result for out\_delay for input\_tran 20ps output\_cap 40fF rc\_length 20 um for metal M5

delay for inverter Case1 & Case2 is slightly low compared to buffer. Hence we can say that increase in output delay due to increase in output transition.



# Chapter 6

## Conclusion and Future Work

### 6.1 Conclusion

Clock mesh technology produces a much lower clock skew compared to a conventional clock tree and, more importantly, is inherently OCV tolerant. OCV derated clock mesh designs generally have both lower skew and higher performance than clock tree structure.

Clock Buffer having small area produces a lower output slew & delay compared to a conventional clock tree and, more importantly, is inherently RC tolerant. While Clock Inverter having Equal rise and fall used for high performance design. So, use of both Clock Inverter and Clock Buffer done to get desired Clock latency.

It is crucial to analyze result for crosstalk for various length before routing, which can be used as a constraint for routing between two pins.

### 6.2 Future Work

Future work consists of Power Optimization using vt swapping without affecting setup & hold. Also, Primetime ECO run for fixing setup and hold.



# References

- [1] Solvnet Synopsys.
- [2] Intel Library.
- [3] Process Variability at the 65nm node and Beyond Sani.R.Nassif IEEE 2008 CICC.
- [4] Mallik Devulapalli and Yuichi Kawahara, Clock Mesh Variation Robustness: Benefits and Analysis,
- [5] Haroon Gauhar, Stephanie Miller, Ashutosh Mujumdar, Dermot ODriscoll, Yuichi Kawahara, Mallik Devulapalli, Jason Binney, and Tom Chau, Structured Methods for Delay, Power, and Variation,
- [6] Harvey Toyama, Clock Mesh for Mainstream Designs,
- [7] Harvey Toyama, Multi-Source CTS Delivers Flexible High Performance and Variation Tolerance, Haroon Gauhar, Stephanie Miller, Ashutosh Mujumdar, Dermot ODriscoll, Yuichi Kawahara, Mallik Devulapalli, Jason Binney, and Tom Chau, Structured Methods for Delay, Power, and Variation,
- [8] The International technology Roadmap for Semiconductors, 2007.1.
- [9] Static Timing Analysis for Nanometer Designs - A Practical Approach by J. Bhasker, Rakesh Chadha.
- [10] <http://vlsi-soc.blogspot.in/2014/12/inverter-vs-buffer-based-clock-tree.html>