Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

PBA Based Optimization for Slew Propagation

Pragati Agarwal[1], Rohit Kumar Gupta[2], Rajat Agarwal[3], Neelu Trivedi[4]
[1]*Research Scholar, Electronics and Communication Engineering, IFTM University, Moradabad*
[2]*Associate Professor, Kalyan Singh Government Medical College, Bulandshahr*
[3]*MBBS, Shri Ram Murti Smarak Institute of Medical Sciences, Bareilly*
[4]*Associate Professor, IFTM University, Moradabad*

ARTICLE INFO

Received: 22 Dec 2024

Revised: 16 Feb 2025

Accepted: 26 Feb 2025

ABSTRACT

When we move towards the nanometer designs, the interconnections between the semiconductor devices. At this level the performance of the design is affected by the metal interconnects due to induced noise. So, in nanometer designs all these affects should also be considered. Propagation of slew in design is an important aspect while performing the static timing analysis (STA) of a design. Slew has a direct impact on delay of a timing path and could make the design pass or fail the timing closure. In conflicts of slew propagation, there are two approaches to move forward. The first approach is graph-based static timing analysis. In a graph-based approach, the worst case delays are taken into consideration by taking into the account the worst case slews (slow slews) along the timing paths, for setup analysis. While in case of hold analysis, the best case delays are taken into consideration by taking into the account the best case slews (fast slews) along the timing paths. The second approach is path-based static timing analysis. In this approach, the actual delays are taken into consideration by taking into account the actual slews along the timing paths, for setup as well as hold analysis. In path-based static timing analysis, delay is computed of the timing path so as to obtain the actual delay values. Such delay calculation takes some extra amount of time. The propagation of slew at various slew merging points in a design have been observed with the help of timing reports. When a path based approach is applied on a design, a significant improvement in the TNS (Total Negative Slack) and WNS (Worst Negative Slack) could be seen in comparison to the graph based approach. When such analysis is done on various designs, a significant improvement was observed. With this path based approach, the area required to implement the design reduced significantly. This is obtained by obtaining the number of cells added during optimization, which comes out to be less in the case of path based approach. Thus, path based approach not only reduce the TNS and WNS, but also helps in congestion reduction in a design by decreasing the area used for implementation of design.

Keywords: *Congestion, GBA, PBA, Total Negative Slack, Worst Negative Slack*

INTRODUCTION

When we move towards the nanometer designs, the interconnections between the semiconductor devices. At this level the performance of the design is affected by the metal interconnects. The phenomenon which occurs at this level and which affects the performance are due to induced noise and crosstalk between parallel metal layers. So, in nanometer designs all these affects should also be considered.

Earlier, dynamic simulations of the gate-level netlist were performed, which has become a problem for the complex designs of today. Nowadays, we have billions of transistors in a typical digital circuit. Doing verification dynamically where a stimulus is given to the input signals and the outputs are observed, could take several days and even weeks to complete the verification process. The dynamic timing simulation depends on the coverage of the used test-bench and the quality for verification. The parts on which the stimuli are applied are tested while the remaining parts remain untested. Therefore, verification of timing analysis of a digital design is done with the help of a technique known as Static Timing Analysis (or STA). With the

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

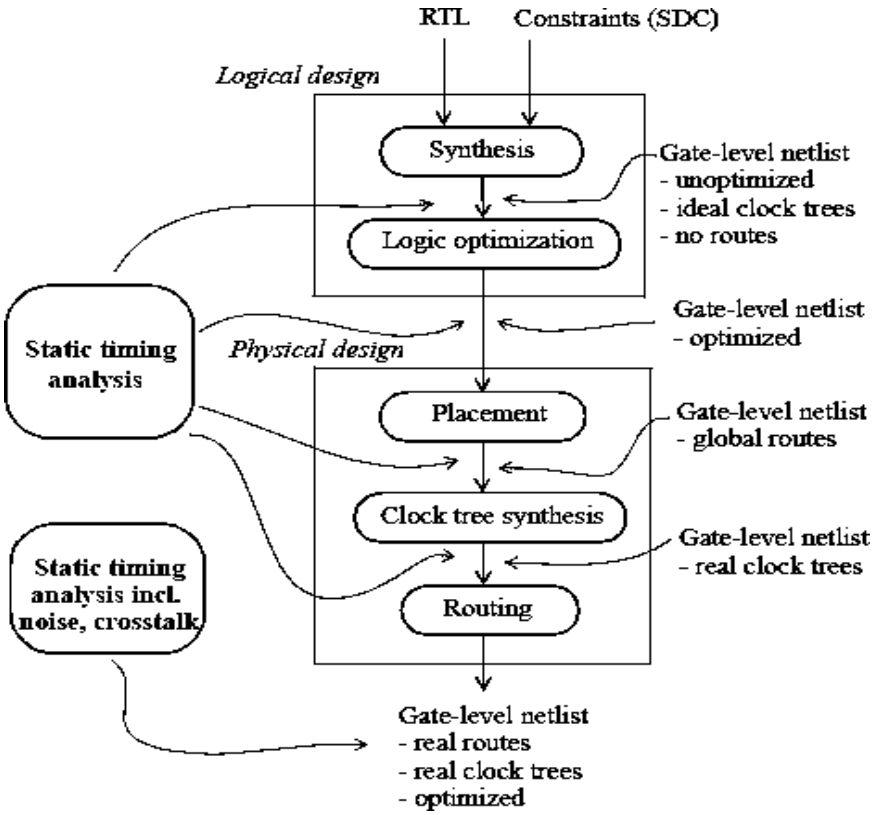
Research Article

introduction of static timing analysis (STA), use of dynamic simulation has been limited to check the functionality of the RTL Design.

In static timing analysis (STA), the timing analysis is done statically, which means that there is no dependency on the signal applied to the input pins of the design. Static timing analysis (STA) is an exhaustive technique and uses mathematical techniques. If the definitions of input clock and of the external environment are set, then static timing analysis (STA) could validate whether that design could work at the desired speed or not.

STA is performed at various steps in the design flow (Figure.1) [25]. STA after the synthesis is done with the ideal clock trees. At this stage, no routes are available. STA is done for the un-optimized as well as optimized gate-level netlist. STA is performed after every step in the physical design flow. After placement, STA is done with global routes. The resistance and capacitance values are obtained with the help of wire load models. After the clock tree synthesis (CTS), STA is performed with real clock trees. After routing, real routes are in place, the STA is done. The noise and crosstalk analysis is also taken care of at this stage. With real routes in place, the resistances and capacitances are extracted with the help of various timing models.

Figure.1 VLSI Design Flow



In [1], an algorithm has been proposed in which timing optimization is done, which is path based, by inserting the buffers. A timing graph of a logical network is made. For thin film transistor circuits (TFT), a static timing analyzer (STAF) is presented in this paper [2] by Chao-Hsuan Hsu et al. For finding the delay of longest and shortest path, the applied STA algorithm used is block based and it is done for dissimilar regions which are under bending. SPICE simulations are used for finding gate delays. This patent [3] discusses the exhaustive path-based STA algorithm. The focus of [4] is on path based approach which is in depth of AOCV model. This path based approach helps in enhancing the capability of AOCV for the reduction of pessimism. In this paper [5] Timing Arc Based Logic Analysis, which is called as TABLA is being presented for the reduction of false noise by performing temporal logic analysis for the given circuit. The used approach is based on timing arc which helps in finding the glitches. The authors in [6] Jose Luis Guntzel et al. proposed a method without using this macro expansion. As macro expansion no doubt increases the execution time but suffers from accuracy loss. So authors proposed an algorithm for functional timing analysis of the complex circuits which has a large number of gates, which is an extension to Automatic Test Generation (ATG) based

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

algorithm. For identifying the correct worst circuit delay, a new method for signal propagation was proposed by authors David Blaauw et al [7]. For selecting longest path delay in a dicey condition, multiple signal propagation is used in this paper. The proposed algorithm in [8] by David Blaauw et al. focusses on finding the delay and critical path of a timing graph as the traditional or earlier methods underrate the circuit delay which creates difficulties for the tools for further processing.

One of the cause of the silicon failure is the pessimism in crosstalk which if not resolved leads to violations which are difficult to identify as well as to fix. So to reduce the inherent pessimism authors Arvind NV et al [9]. proposed a solution for this. The results show a reduction of 70% violations using path based analysis and also the run time is not very large. In [10] for the purpose of timing validation, a path based technique is presented. The technique has two paths, first is ranking optimization and the other one is path filtering. So for characterizing the interdependent hold and setup times, authors proposed the methodology and algorithm [11]. Multiple pairs in STA can be utilized to decrease the unwanted pessimism and the interdependency reduces the optimism. This paper [12] deals with that moments of input waveform only by mapping the timing waveforms to the characterization waveform. This mapping is based on the moments of waveforms. It is required to verify all the process corners of a system, but due to the increase in PVT variations, the number of process corners have increased. So the designers are trying to decrease the number of process corners verification but this can lead to some unacceptable results for some cases, not all. In case of setup analysis, this could be possible, but not in case of hold. So the authors in their work [13] introduces a method which covers all the process corners in case of hold analysis. The authors Abdoul Rjoub et al. in [14] proposed a graph model so as to observe the CMOS circuits at transistor level and to check their behavior. BSIM4 equations is the basic on which estimation of timing analysis is done for the CMOS circuits. In [15], this inaccuracy which is caused by path delay fault is considered and for that authors present a method. Google recently introduced a programming paradigm called as MapReduce for processing the big data. So in [16] authors introduced an algorithm which is path based with MapReduce, called as fast incremental path based algorithm. Using this algorithm incremental PBA problem is transformed into tasks and then they are parallel mapped to reduce and reduce the operation.

Focus of [17] is more on graph based designs and the authors are against path based designs because of the reasons like in modern designs there are large number of paths available, which are non-scalable, so as the path based techniques. The authors proposed a graph based technique, that addresses the limitations of path based techniques. For removal of pessimism, a number of approaches were developed. In this paper another approach for the same is discussed by [18] A.B. Kanng et al. in which timing model of a flexible flip-flop is being exploited for the purpose of reducing its pessimism in setup and hold critical path timing analysis. In [19] a methodology for solving timing problems at transistor level for CMOS circuits is introduced, also methods for building the tool which consists of algorithms and methods for performing the STA. In [20], authors focus on statistical static timing analysis, its aspects and applications in VLSI systems. SSTA comes as a solution to the process variation which cause uncertainties in power and process. So authors present block based SSTA model for both global and path correlations. Common-path-pessimism is one of the factor which cause problems while doing timing analysis. So, its removal is a crucial step, which is commonly called as Common-path-pessimism removal(CPPR). So the concern
[21] is CPPR, the solution comes as a fast path based analysis. It is different from the others in the way that instead of searching the path explicitly. An implicit representation of path is performed which results in faster run time with the smaller search space. The timer used in this algorithm is ease of use, less complex, ease of code, simple.

ESTIMATION OF SLEW PROPAGATION

Propagation of slew in design is an important aspect while performing the static timing analysis (STA) of a design. Slew has a direct impact on delay of a timing path and could make the design pass or fail the timing closure. This chapter introduces the concept of slew and how it is propagated in two different ways in the design, and how this is a problem in itself to choose between the incident-slews at a slew merging point.

GRAPH-BASED STA AND PATH-BASED STA

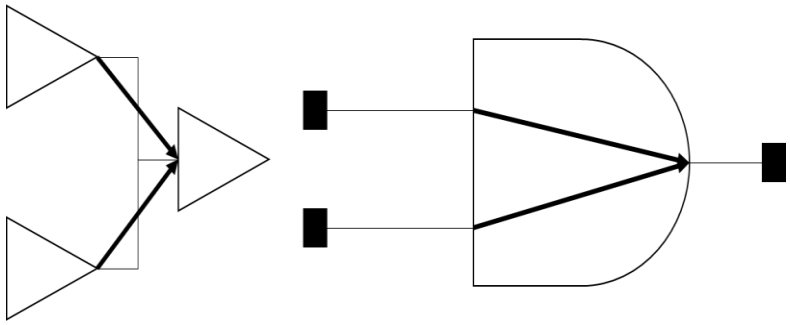
In a digital circuit design, there are points where timing arcs merge. Such points could be termed as slew merging points. Such slew merging points are shown in Figure 2. Now at these points, the question arises

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

which slew should propagate forward and be considered in the delay calculation for a particular timing path.

Figure.2 Slew Merging Points



In such conflicts, there are two approaches to move forward. The first approach is graph-based static timing analysis. In a graph-based approach, the worst case delays are taken into consideration by taking into the account the worst case slews (slow slews) along the timing paths, for setup analysis. While in case of hold analysis, the best case delays are taken into consideration by taking into the account the best case slews (fast slews) along the timing paths [22]. The second approach is path-based static timing analysis. In this approach, the actual delays are taken into consideration by taking into account the actual slews along the timing paths, for setup as well as hold analysis. In path-based static timing analysis, delay is computed of the timing path so as to obtain the actual delay values. Such delay calculation takes some extra amount of time. There is a trade-off between this extra calculation time and accurate delay calculation [23]. Most of the design engineers use graph-based static timing analysis for delay calculation in a digital design. The paths which are critical, path-based static timing analysis could be used when the tape-out is near. Generally, the runtimes of path-based static timing analysis are more as compared to the graph- based static timing analysis, for the same number of paths. Because of this only, use of path-based static timing analysis is limited. So, the design engineers need to take a call whether to fix the timing with additional run-time required for performing path-based static timing analysis and missing the deadline or to go for graph-based static timing analysis which consumes lesser amount of time [24].

PROBLEM DESCRIPTION

When different slews arrive at these slew merging points, the decision has to be made which slew should propagate forward for delay calculation of the further logic gates. In case of graph-based static timing analysis, the run-times are less but the delays which are obtained are less accurate. Graph-based static timing analysis is a pessimistic approach where worst case slack is obtained for the setup and hold analysis. Moreover, in graph-based static timing analysis, a single value of delay is considered for a timing arc, which is part of a multiple timing paths.

This is not the case in path-based static timing analysis. The paths are recalculated to obtain the actual delays. This is an optimistic approach. In path-based static timing analysis, an arc could have multiple delay values for different timing paths which makes the timing analysis more realistic. In path-based timing analysis, a path which is recalculated, is separated out from the graph and a more detailed timing analysis is carried on it. This process is called path recalculation because a graph path has been taken out and recalculated for a totally new timing behavior.

In the case of path-based static timing analysis, the slack improves a lot. This happens because the worst- case timing has already been considered in case of graph-based timing analysis. So, there is no chance of slack degradation. There are various improvements which occur in the timing of a path when a particular path is recalculated. Some the improvements are mentioned below:

a) When a path is recalculated for delays along the nets and cells, accurate timing results of slew degradation are obtained.

b) As the focus is single path, so in order to study effects like signal integrity a single edge could be propagated along the path, there is no need of any arrival window.

c) In graph-based static timing analysis, the slowest slew is propagated which is more prone to crosstalk effects. This is not the case in path-based static timing analysis where faster slews are propagated, which are

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

less prone to crosstalk effects.
d) In graph-based static timing analysis, the values used for determination of CRPR are very pessimistic but in case of path-based static timing analysis, accurate CRPR values are obtained for a timing path.

The recalculation of the timing paths is done exhaustively. There are several factors which contribute to the total number of timing paths, which are described below:
a) Total number of start-points.
b) Rising and falling edges are considered separate.
c) Existence of paths which branch apart but later converges back together.
d) Existence of gates which have both non-inverting and inverting timing arcs like XOR gates.

For a certain endpoint, multiple timing paths could exist. So all these paths to a certain endpoint are recalculated in order to obtain the timing path with the most negative slack.

In graph-search path analysis, for a certain number of endpoints, all the timing paths are analyzed. The paths with the maximum slack checked among all the checkpoints are reported. The exhaustive path-search algorithm is a much more complex process. In exhaustive path-search algorithm, all the paths leading to a certain endpoint are recalculated with their actual delay values. The path which was earlier the worst case path could no longer be the worst one, after recalculation. If a certain path is again encountered while moving from worst case slack to best case slack, those having the worst slack are reported. Approximately twenty-five thousand such iterations are performed in an exhaustive path-search process.

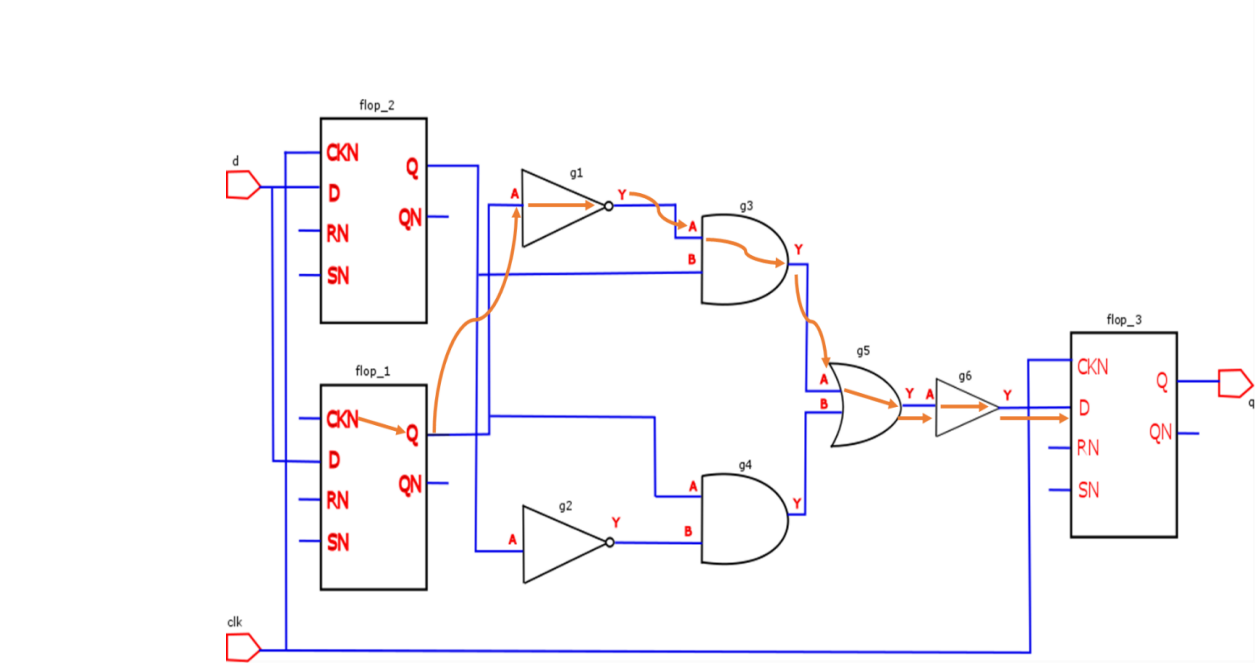
ANALYSIS AND RESULTS

Introduction

For the design shown in figure 3, there are various slew merging points. As could be seen. there could be four possible timing paths. In this section, the slew propagation along all these paths is analyzed by seeing their timing reports.

a)Path 1: flop_1/CKN to flop_3/D
b)Path 2: flop_1/CKN to flop_3/D through g4/A
c)Path 3: flop_2/CKN to flop_3/D
d)Path 4: flop_2/CKN to flop_3/D through g3/B
All the paths are shown in the subsequent figures.

Figure.3 Path 1: flop_1/CKN to flop_3/D



Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure.4 Path 2: flop_1/CKN to flop_3/D through g4/A

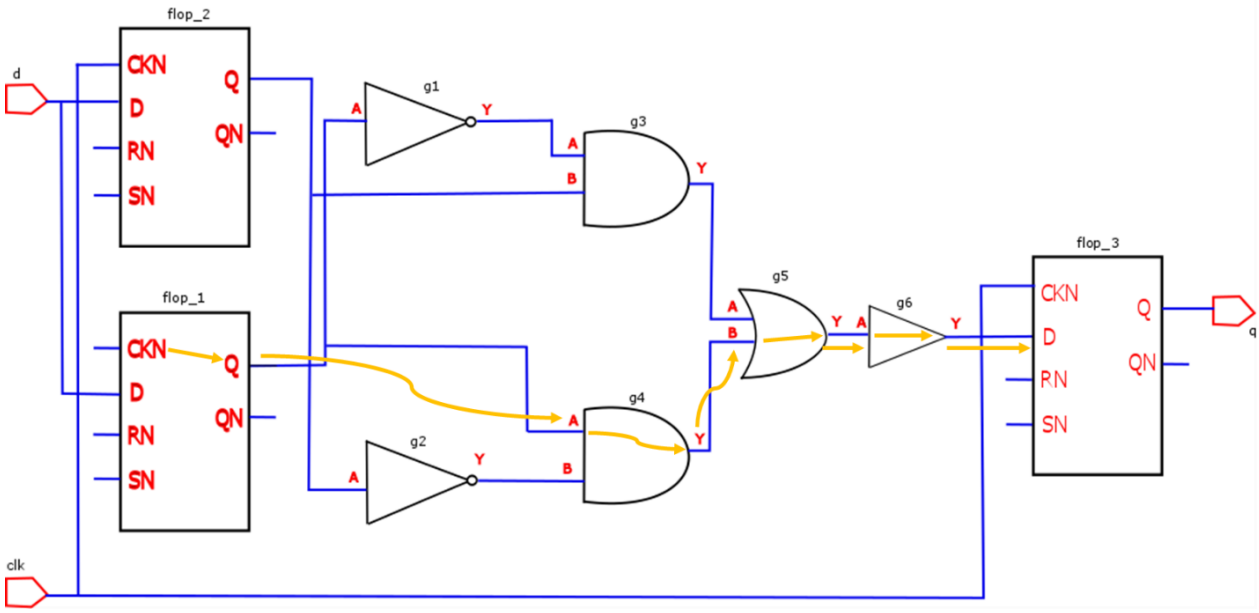
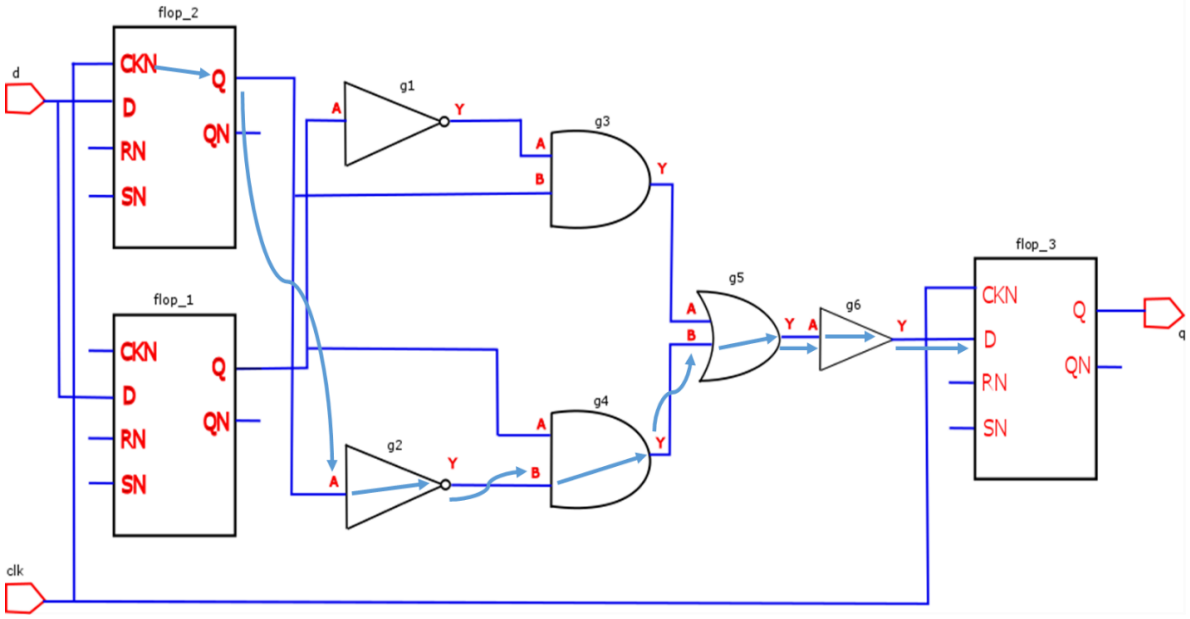


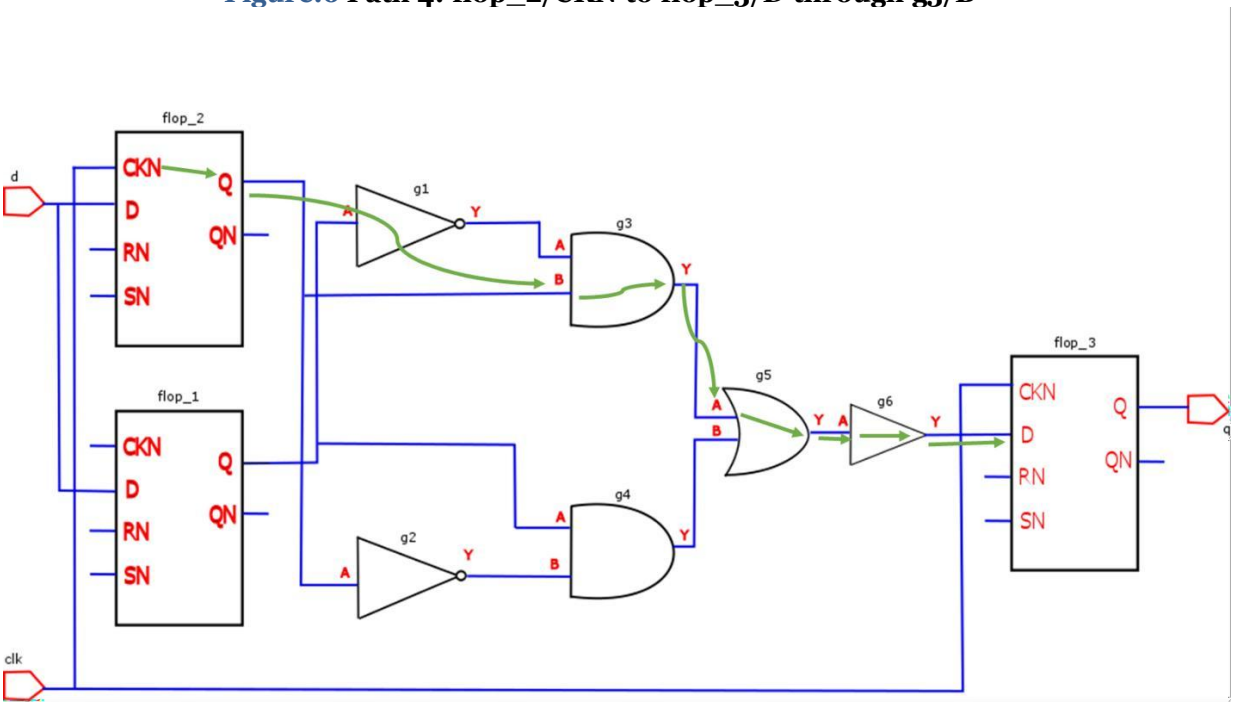
Figure.5 Path 3: flop_2/CKN to flop_3/D



Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure.6 Path 4: flop_2/CKN to flop_3/D through g3/B



Slew Propagation in Graph Based Approach

In graph-based static timing analysis approach, the worst case slew propagates at various slew merging points. The timing reports for the four timing paths are discussed in this section.

Figure.7 Timing Report for Path 1

```
Optimus> report_timing -from flop_1/CKN -to g6/Y -through g4/A

 Timing Path to flop_3/D


 Path Start Point : flop_1 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Path End Point   : flop_3 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Mode             : func1
 Corner           : slow1
 Path Type        : max
 Path Group       : clk
```

Pin	Cell	Edge	Arrival	Delay	Slew	Total Cap
-> flop_1/CKN	**DFFNSRX1**	**Fall**	**5.0000**	**5.0000**	**0.0000**	
flop_1/Q	DFFNSRX1	Fall	5.1920	0.1920	0.0360	4.3480e-03
-> g4/A	**AND2X1**	**Fall**	**5.1920**	**0.0000**	**0.0360**	
g4/Y	AND2X1	Fall	5.2600	0.0680	0.0280	1.8420e-03
g5/B	OR2X1	Fall	5.2600	0.0000	0.0280	
g5/Y	OR2X1	Fall	5.3560	0.0960	0.0350	2.6760e-03
g6/A	BUF1	Fall	5.3560	0.0000	0.0350	
-> g6/Y	**BUF1**	**Fall**	**5.3700**	**0.0140**	**0.0160**	1.7320e-03
flop_3/D	DFFNSRX1	Fall	5.3700	0.0000	0.0160	

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure.8 Timing Report for Path 2

```
Optimus> report_timing -from flop_1/CKN -to g6/Y

 Timing Path to flop_3/D


 Path Start Point : flop_1 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Path End Point   : flop_3 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Mode             : func1
 Corner           : slow1
 Path Type        : max
 Path Group       : clk
```

Pin	Cell	Edge	Arrival	Delay	Slew	Total Cap
-> flop_1/CKN	**DFFNSRX1**	**Fall**	**5.0000**	**5.0000**	**0.0000**	
flop_1/Q	DFFNSRX1	Rise	5.2190	0.2190	0.0480	4.3480e-03
g1/A	INVGCX1	Rise	5.2190	0.0000	0.0480	
g1/Y	INVGCX1	Fall	5.2330	0.0140	0.0180	1.6720e-03
g3/A	AND2X1	Fall	5.2330	0.0000	0.0180	
g3/Y	AND2X1	Fall	5.2960	0.0630	0.0270	1.7310e-03
g5/A	OR2X1	Fall	5.2960	0.0000	0.0270	
g5/Y	OR2X1	Fall	5.3830	0.0870	0.0350	2.6760e-03
g6/A	BUF1	Fall	5.3830	0.0000	0.0350	
-> g6/Y	**BUF1**	**Fall**	**5.3970**	**0.0140**	**0.0160**	1.7320e-03
flop_3/D	DFFNSRX1	Fall	5.3970	0.0000	0.0160	

Figure.9 Timing Report for Path 3

```
Optimus> report_timing -from flop_2/CKN -to g6/Y

 Timing Path to flop_3/D


 Path Start Point : flop_2 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Path End Point   : flop_3 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Mode             : func1
 Corner           : slow1
 Path Type        : max
 Path Group       : clk
```

Pin	Cell	Edge	Arrival	Delay	Slew	Total Cap
-> flop_2/CKN	**DFFNSRX1**	**Fall**	**5.0000**	**5.0000**	**0.0000**	
flop_2/Q	DFFNSRX1	Rise	5.2200	0.2200	0.0500	4.5580e-03
g2/A	INVGCX1	Rise	5.2200	0.0000	0.0500	
g2/Y	INVGCX1	Fall	5.2350	0.0150	0.0190	1.8820e-03
g4/B	AND2X1	Fall	5.2350	0.0000	0.0190	
g4/Y	AND2X1	Fall	5.3080	0.0730	0.0280	1.8420e-03
g5/B	OR2X1	Fall	5.3080	0.0000	0.0280	
g5/Y	OR2X1	Fall	5.4040	0.0960	0.0350	2.6760e-03
g6/A	BUF1	Fall	5.4040	0.0000	0.0350	
-> g6/Y	**BUF1**	**Fall**	**5.4180**	**0.0140**	**0.0160**	1.7320e-03
flop_3/D	DFFNSRX1	Fall	5.4180	0.0000	0.0160	

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure.10 TimingReport for Path 4

```
Optimus> report_timing -from flop_2/CKN -to g6/Y -through g3/B

 Timing Path to flop_3/D


 Path Start Point : flop_2 (DFFNSRX1)
                      (falling edge-triggered flip-flop clocked by clk')
 Path End Point   : flop_3 (DFFNSRX1)
                      (falling edge-triggered flip-flop clocked by clk')
 Mode             : func1
 Corner           : slow1
 Path Type        : max
 Path Group       : clk

-------------------------------------------------------------------------------
| Pin          | Cell     | Edge | Arrival | Delay  | Slew   | Total Cap  |
|--------------+----------+------+---------+--------+--------+------------|
| -> flop_2/CKN | DFFNSRX1 | Fall |  5.0000 | 5.0000 | 0.0000 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_2/Q  | DFFNSRX1 | Fall |  5.1920 | 0.1920 | 0.0370 | 4.5580e-03 |
|--------------+----------+------+---------+--------+--------+------------|
| -> g3/B      | AND2X1   | Fall |  5.1920 | 0.0000 | 0.0370 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g3/Y      | AND2X1   | Fall |  5.2680 | 0.0760 | 0.0270 | 1.7310e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/A      | OR2X1    | Fall |  5.2680 | 0.0000 | 0.0270 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/Y      | OR2X1    | Fall |  5.3550 | 0.0870 | 0.0350 | 2.6760e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g6/A      | BUF1     | Fall |  5.3550 | 0.0000 | 0.0350 |            |
|--------------+----------+------+---------+--------+--------+------------|
| -> g6/Y      | BUF1     | Fall |  5.3690 | 0.0140 | 0.0160 | 1.7320e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_3/D  | DFFNSRX1 | Fall |  5.3690 | 0.0000 | 0.0160 |            |
-------------------------------------------------------------------------------
```

It could be observed from the timing reports that at various slew merging points like g3 and g4, the worst case slew i.e. 0.027 ns and 0.028 ns is propagating, respectively.

Slew Propagation in Path Based Approach

In path-based static timing analysis approach, the accurate slew propagates at various slew merging points. The timing reports for the four timing paths are discussed in this

Figure.11 Timing Report for Path 1

```
Optimus> report_timing -from flop_1/CKN -to g6/Y -pba_mode exhaustive

 Timing Path to flop_3/D


 Path Start Point : flop_1 (DFFNSRX1)
                      (falling edge-triggered flip-flop clocked by clk')
 Path End Point   : flop_3 (DFFNSRX1)
                      (falling edge-triggered flip-flop clocked by clk')
 Mode             : func1
 Corner           : slow1
 Path Type        : max (recalculated)
 Path Group       : clk

-------------------------------------------------------------------------------
| Pin          | Cell     | Edge | Arrival | Delay  | Slew   | Total Cap  |
|--------------+----------+------+---------+--------+--------+------------|
| -> flop_1/CKN | DFFNSRX1 | Fall |  5.0000 | 5.0000 | 0.0000 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_1/Q  | DFFNSRX1 | Rise |  5.2190 | 0.2190 | 0.0480 | 4.3480e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g1/A      | INVGCX1  | Rise |  5.2190 | 0.0000 | 0.0480 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g1/Y      | INVGCX1  | Fall |  5.2330 | 0.0140 | 0.0180 | 1.6720e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g3/A      | AND2X1   | Fall |  5.2330 | 0.0000 | 0.0180 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g3/Y      | AND2X1   | Fall |  5.2960 | 0.0630 | 0.0260 | 1.7310e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/A      | OR2X1    | Fall |  5.2960 | 0.0000 | 0.0260 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/Y      | OR2X1    | Fall |  5.3830 | 0.0870 | 0.0350 | 2.6760e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g6/A      | BUF1     | Fall |  5.3830 | 0.0000 | 0.0350 |            |
|--------------+----------+------+---------+--------+--------+------------|
| -> g6/Y      | BUF1     | Fall |  5.3970 | 0.0140 | 0.0160 | 1.7320e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_3/D  | DFFNSRX1 | Fall |  5.3970 | 0.0000 | 0.0160 |            |
-------------------------------------------------------------------------------
```

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure.12 Timing Report for Path 2

```
Optimus> report_timing -from flop_1/CKN -to g6/Y -pba_mode exhaustive -through g4/A

Timing Path to flop_3/D


Path Start Point : flop_1 (DFFNSRX1)
                   (falling edge-triggered flip-flop clocked by clk')
Path End Point   : flop_3 (DFFNSRX1)
                   (falling edge-triggered flip-flop clocked by clk')
Mode             : func1
Corner           : slow1
Path Type        : max (recalculated)
Path Group       : clk

-------------------------------------------------------------------------
| Pin          | Cell     | Edge | Arrival | Delay  | Slew   | Total Cap  |
|--------------+----------+------+---------+--------+--------+------------|
| -> flop_1/CKN | DFFNSRX1 | Fall |  5.0000 | 5.0000 | 0.0000 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_1/Q  | DFFNSRX1 | Fall |  5.1920 | 0.1920 | 0.0360 | 4.3480e-03 |
|--------------+----------+------+---------+--------+--------+------------|
| -> g4/A      | AND2X1   | Fall |  5.1920 | 0.0000 | 0.0360 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g4/Y      | AND2X1   | Fall |  5.2600 | 0.0680 | 0.0260 | 1.8420e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/B      | OR2X1    | Fall |  5.2600 | 0.0000 | 0.0260 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/Y      | OR2X1    | Fall |  5.3560 | 0.0960 | 0.0350 | 2.6760e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g6/A      | BUF1     | Fall |  5.3560 | 0.0000 | 0.0350 |            |
|--------------+----------+------+---------+--------+--------+------------|
| -> g6/Y      | BUF1     | Fall |  5.3700 | 0.0140 | 0.0160 | 1.7320e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_3/D  | DFFNSRX1 | Fall |  5.3700 | 0.0000 | 0.0160 |            |
-------------------------------------------------------------------------
```

In path-based approach, for a timing path the actual slew is recalculated and is propagated forward, at the various slew merging points. This could be observed in the timing reports above. At a slew merging point g3, the timing arc from g3/A to g3/Y, the propagating slew is 0.026 ns, while for g3/B to g3/Y, the propagating slew is 0.027 ns. Similar results could be observed at other slew merging points like g4.

Figure.13 Timing Report for Path 3

```
Optimus> report_timing -from flop_2/CKN -to g6/Y -pba_mode exhaustive

Timing Path to flop_3/D


Path Start Point : flop_2 (DFFNSRX1)
                   (falling edge-triggered flip-flop clocked by clk')
Path End Point   : flop_3 (DFFNSRX1)
                   (falling edge-triggered flip-flop clocked by clk')
Mode             : func1
Corner           : slow1
Path Type        : max (recalculated)
Path Group       : clk

-------------------------------------------------------------------------
| Pin          | Cell     | Edge | Arrival | Delay  | Slew   | Total Cap  |
|--------------+----------+------+---------+--------+--------+------------|
| -> flop_2/CKN | DFFNSRX1 | Fall |  5.0000 | 5.0000 | 0.0000 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_2/Q  | DFFNSRX1 | Rise |  5.2200 | 0.2200 | 0.0500 | 4.5580e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g2/A      | INVGCX1  | Rise |  5.2200 | 0.0000 | 0.0500 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g2/Y      | INVGCX1  | Fall |  5.2350 | 0.0150 | 0.0190 | 1.8820e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g4/B      | AND2X1   | Fall |  5.2350 | 0.0000 | 0.0190 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g4/Y      | AND2X1   | Fall |  5.3080 | 0.0730 | 0.0280 | 1.8420e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/B      | OR2X1    | Fall |  5.3080 | 0.0000 | 0.0280 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/Y      | OR2X1    | Fall |  5.4040 | 0.0960 | 0.0350 | 2.6760e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g6/A      | BUF1     | Fall |  5.4040 | 0.0000 | 0.0350 |            |
|--------------+----------+------+---------+--------+--------+------------|
| -> g6/Y      | BUF1     | Fall |  5.4180 | 0.0140 | 0.0160 | 1.7320e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_3/D  | DFFNSRX1 | Fall |  5.4180 | 0.0000 | 0.0160 |            |
-------------------------------------------------------------------------
```

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure.14 Timing Report for Path 4

```
Optimus> report_timing -from flop_2/CKN -to g6/Y -pba_mode exhaustive -through g3/B

 Timing Path to flop_3/D


 Path Start Point : flop_2 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Path End Point   : flop_3 (DFFNSRX1)
                    (falling edge-triggered flip-flop clocked by clk')
 Mode             : func1
 Corner           : slow1
 Path Type        : max (recalculated)
 Path Group       : clk

-------------------------------------------------------------------------
| Pin          | Cell     | Edge | Arrival | Delay  | Slew   | Total Cap  |
|--------------+----------+------+---------+--------+--------+------------|
| -> flop_2/CKN | DFFNSRX1 | Fall |  5.0000 | 5.0000 | 0.0000 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_2/Q  | DFFNSRX1 | Fall |  5.1920 | 0.1920 | 0.0370 | 4.5580e-03 |
|--------------+----------+------+---------+--------+--------+------------|
| -> g3/B      | AND2X1   | Fall |  5.1920 | 0.0000 | 0.0370 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g3/Y      | AND2X1   | Fall |  5.2680 | 0.0760 | 0.0270 | 1.7310e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/A      | OR2X1    | Fall |  5.2680 | 0.0000 | 0.0270 |            |
|--------------+----------+------+---------+--------+--------+------------|
|    g5/Y      | OR2X1    | Fall |  5.3550 | 0.0870 | 0.0350 | 2.6760e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    g6/A      | BUF1     | Fall |  5.3550 | 0.0000 | 0.0350 |            |
|--------------+----------+------+---------+--------+--------+------------|
| -> g6/Y      | BUF1     | Fall |  5.3690 | 0.0140 | 0.0160 | 1.7320e-03 |
|--------------+----------+------+---------+--------+--------+------------|
|    flop_3/D  | DFFNSRX1 | Fall |  5.3690 | 0.0000 | 0.0160 |            |
-------------------------------------------------------------------------
```

PBA Based Optimization on Various Designs

Various designs were taken. On them graph-based and path-based optimization were carried out, and the timing reports were obtained for improvement in the total negative slack (TNS) and the worst negative slack (WNS). The results obtained are shown in the Table 1.

TABLE 1 OPTIMIZATION ON VARIOUS DESIGNS

Design	Operating Frequency (MHz)	Number of Standard Cells	TNS After GBA Optimization (ns)	TNS After PBA Optimization (ns)	WNS After GBA Optimization (ns)	WNS After PBA Optimization (ns)
1.	333.33	2044177	-457.164	-421.998	-12.576	-12.515
2.	1000	208877	-776.712	-755.417	-0.388	-0.388
3.	1754.4	633888	-7230.315	-5895.773	-2.51	-2.509
4.	48.01	70452	-1493.779	-1451.705	-50.894	-50.894

In the Table 1, it could be seen that there is significant improvement in the TNS values when PBA based optimization is done in comparison to GBA based optimization. The improvement is there because in path based approach, the timing path is recalculated to obtain accurate slews of the arcs on that path, instead of the worst case slew which is selected in case of graph based approach.

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Figure.15 Congestion Analysis Graph

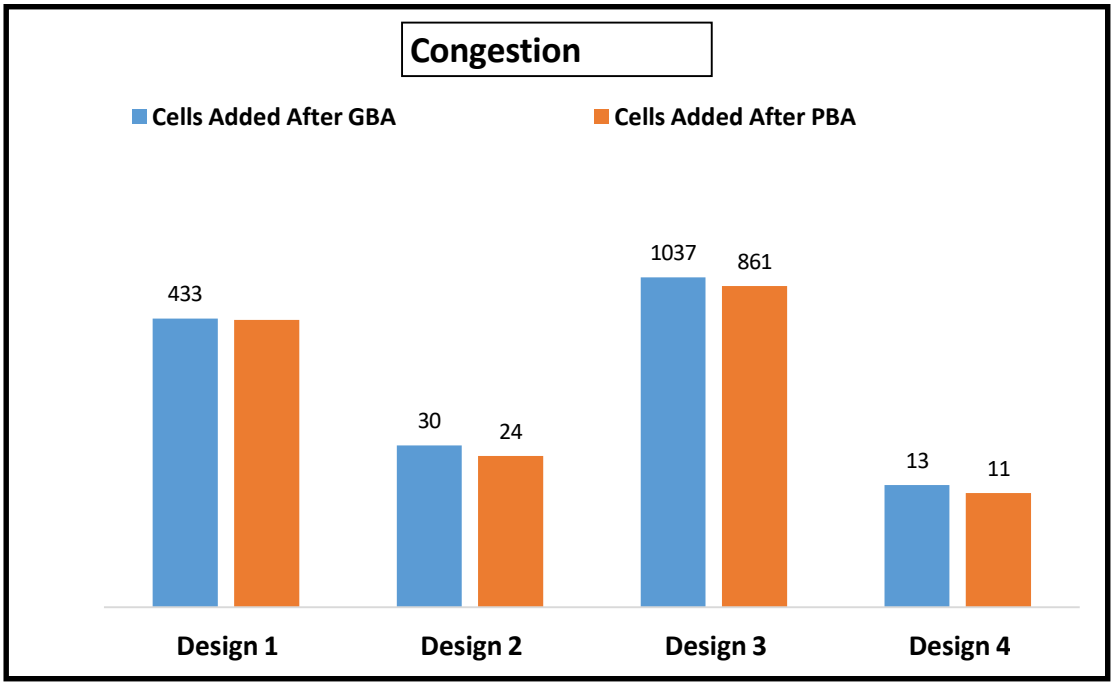


Figure.15 shows the congestion analysis graphs. The number of cells added (buffers/inverters) after optimization were obtained. This analysis was done on the same set of designs which are discussed in the Table.1. As observed in the case of every design, the number of cells added is less in case of path based optimization as compared to the graph based optimization. The reason behind this is that in case of path based optimization the total negative slack is lesser as compared to that in graph based optimization (refer Table.1), so less number of cells (buffers/inverters) are needed to remove it. Thus, lesser area is needed if path based optimization is followed in a design.

CONCLUSION

STA has been an important part of the VLSI Physical Design Process. A design which does not meets the timing requirements would not work properly. Propagation of slew in various timing paths is an important aspect which should be analyzed separately, as slew directly affects the delay of the cell or a wire. This work focusses on the slew propagation at a slew merging point. There are two aspects of looking at it. First one is the graph based approach, which is a pessimistic approach and in it the slower or the worst case slew would propagate at a slew merging point, as could be seen in the previous section. Second one is the path based approach, which is a less pessimistic approach and in it the actual slew of the timing arc for that particular path propagates forward at a slew merging point, as observed in the previous section. When PBA is applied on a design, a significant improvement in the TNS and WNS could be seen in comparison to the graph based approach. When such analysis is done on various design, a significant improvement was observed. Congestion has also been reduced as the number of buffers/inverters which got added after the PBA optimization is less as compared to GBA optimization. Thus, PBA optimization is an approach which offers lesser area implementation along with a more accurate timing closure.

REFRENCES

[1]. Yiqian Zhang, Qiang Zhou, Xian long Hong and Yici Cai, “Path-based Timing Optimization by Buffer Insertion With Accurate Delay Model,” Proc. ASIC, 21 – 24 October, 2003.
[2]. Chao-Hsuan Hsu, Chester Liu, En-Hua Ma and James Chien-Mo Li, “Static Timing Analysis for Flexible TFT Circuits”, in proc. of 47th ACM/IEEE Design Automation Conference (DAC), pp. 799-802,

June 2010.
[3]. Cristian Soviani, Rachid N. Helaihel, Khalid Rahmat, Fremont, "Efficient Exhaustive Path-Based Static Timing Analysis Using a Fast Estimation Technique" United States Patent Soviani et al., Patent No. US 8,079,004 B2, December 2011.
[4]. Pua Siaw Fuang and Nor Muzlifah Mahyuddin "Implementation Study of Path-Based AOCV Model on Pessimism Reduction for 20nm Technology" IEEE International Conference on Control System, Computing and Engineering, pp. 80-83, November 2014.
[5]. Murthy Palla1, Jens Bargfrede, Stephan Eggersglüß, Walter Anheier, Rolf Drechsler, "Timing Arc Based Logic Analysis for False Noise Reduction", IEEE/ACM International Conference on Computer-Aided Design - Digest of Technical Papers, pp. 225-230, 2009.
[6]. Jose Luis Guntzel, Ana Cristina Medina Pinto, Eduardo d'Avila and Ricardo Reis, "ATG- Based Timing Analysis of Circuits Containing Complex Gates" in proc. of 13th Symposium on Integrated Circuits and Systems Design, pp. 21-26, Sept 2000.
[7]. David Blaauw, Vladimir Zolotov, Savithri Sundareswaran, Chanhee Oh and Rajendran Panda "Slope Propagation in Static Timing Analysis" in proc. of IEEE/ACM International Conference on Computer Aided Design, ICCAD, 2000, pp. 338-343, Nov 2000.
[8]. David Blaauw, Vladimir Zolotov, and Savithri Sundareswaran, "Slope Propagation in Static Timing Analysis" IEEE transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 21, no. 10, October 2002.
[9]. Arvind NV, Rajagopal KA, Ajith HS and Suparna Das, "Path Based Approach for Crosstalk Delay Analysis", in proc. of 17th International Conference on VLSI Design, pp. 727-730, 2004.
[10]. Leonard Lee, Li-C. Wangl, T. M. Mak, and Kwang-Ting Cheng, "A Path-Based Methodology for Post-Silicon Timing Validation", in proc. of International Conference on Computer Aided Design, 2004. ICCAD-2004. IEEE/ACM, pp. 713-720, Nov 2004.
[11]. Emre Salman, Ali Dasdan, Feroze Taraporevala, Kayhan K¨uc¸ ¨ukc¸akar, and Eby G. Friedman, "Pessimism Reduction in Static Timing Analysis Using Interdependent Setup and Hold Times", Proceedings of the 7th International Symposium on Quality Electronic Design (ISQED'06) 6 pp. – 164, March 2006.
[12]. David D. Ling, Chandu Visweswariah, Peter Feldmann and Soroush Abbaspour, "A Moment-Based Effective Characterization Waveform For Static Timing Analysis", 46th ACM/IEEE Design Automation Conference, DAC '09, pp. 19-24, July 2009.
[13]. S. Onaissi, F. Taraporevala, J. Liu and F. Najm "A Fast Approach for Static Timing Analysis Covering All PVT Corners", 48th ACM/EDAC/IEEE Design Automation Conference (DAC), 2011, pp. 777-782, June 2011.
[14]. Abdoul Rjoub and Almotasem Bellah Alajlouni, "Graph Modeling for Static Timing Analysis at Transistor level in Nano-Scale CMOS Circuits", 16th IEEE Mediterranean Electrotechnical Conference, pp. 80-83, March 2012.
[15]. Bo Yao, Arani Sinha and Irith Pomeranz, "Path Selection Based On Static Timing Analysis Considering Input Necessary Assignments", 2013 IEEE 31st VLSI Test Symposium (VTS), pp.1-6, April-May 2013.
[16]. Tsung-Wei Huang and Martin D. F. Wong, "On Fast Timing Closure: Speeding Up Incremental Path-Based Timing Analysis with MapReduce", 2015 ACM/IEEE International Workshop on System Level Interconnect Prediction (SLIP), pp. 1-6, June 2015.
[17]. Hari Cherupalli and John Sartori, "Graph-based Dynamic Analysis: Efficient Characterization of Dynamic Timing and Activity Distributions", IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 729 – 735, Nov 2015.
[18]. A.B. Kanng, H. Lee, "Timing Margin Recovery With Flexible Flip-Flop Timing Model", Fifteenth International Symposium on Quality Electronic Design, pp. 496 – 503, March 2014.
[19]. Abdoul Rjoub, Almotasem Bellah Alajlouni, Hassan Almanasrah, "Graph Modeling for Static Timing Analysis at Transistor Level in Nano-Scale CMOS Circuits", 16th IEEE Mediterranean Electrotechnical Conference, pp. 80-83, March 2012.
[20]. Abu M. Baker and Yingtao Jiang, "Modeling and Architectural Simulations of the Statistical Static Timing Analysis of the Non-Gaussian Variation Sources for VLSI Circuits", International Journal of Scientific and Research Publications, Volume 3, Issue 1, January 2013.
[21]. Tsung-Wei Huang, Pei-Ci Wu, and Martin D. F. Wong, "Fast Path-Based Timing Analysis for CPPR", 2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 596 – 599, Nov 2014.
[22]. Izumi Nitta, Toshiyuki Shibuya and Katsumi Homma, "Statistical Static Timing Analysis Technology", FUJITSU Sci. Tech. J., 43, 4, October 2007.

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

[23]. Hamed Abrishami, Jinan Lou, Jeff Qin, Juergen Froessl, Massoud Pedram, "Post Sign-off Leakage Power Optimization", in proc. of Design Automation Conference (DAC), 2011 48th ACM/EDAC/IEEE, pp. 453 – 458, June 2011.
[24]. Aseem Agarwal, David Blaauw, Vladimir Zolotov, Savithri Sundareswaran, Min Zhao, Kaushik Gala, Rajendran Panda, "Path-Based Statistical Timing Analysis Considering Inter and Intra-Die Correlations", 48th ACM/EDAC/IEEE 48th ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 453 – 458, June 2011.
[25]. http://community.cadence.com/cadence_blogs_8/b/ii/archive/2013/12/02/signoff-summit-tempus-path-based-analysis-and-finfet-timing-closure
[26]. http://electronicdesign.com/eda/eda-vendors-should-improve-runtime-performance-path- based-analysis
[27]. http://vlsi-soc.blogspot.in/2014/07/timing-analysis-graph-based-vs-path.html
[28]. J. Bhasker & Rakesh Chadha, "Static Timing Analysis for Nanometer Designs – A Practical Approach," Springer, 2009.
[29]. Himanshu Bhatnagar, "Advanced ASIC Chip Synthesis," Second Edition, Kluwer Academic Publishers, 2002.