Improvement in Static Timing Analysis for Early TA-Signoff Closure in SoC flow at Very Deep Submicron Nodes

Vinod Kumar Singh

Advisors
Prof. R. N. Biswas
Mr. Mohit Verma

Submitted in partial fulfilment of the requirements for the degree of
M.Tech in Electronics and Communication Engineering,
with Specialization in
VLSI and Embedded Systems,

Indraprastha Institute of Information Technology, New Delhi

July 2014

Declaration

I hereby declare that the work presented in the report entitled "**Improvement in Static Timing Analysis for Early TA-Signoff Closure in SoC flow at Very Deep Submicron Nodes**" submitted by me for the partial fulfilment of the requirements for the degree of *Master of Technology in Electronics and Communication Engineering with specialization in VLSI & Embedded Systems* at Indraprastha Institute of Information Technology, Delhi, is an authentic record of my work carried out under guidance of Prof **R.N. Biswas** and **Mohit Verma**. Due acknowledgements have been given in the report to all material used. This work has not been submitted anywhere else for the reward of any other degree.

Vinod Kumar Singh **Place & Date**: .....

Certificate

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Prof. R.N. Biswas **Place & Date**: .....

(IIIT-Delhi)

Mr. Mohit Verma **Place & Date**: .....

(STMicroelectronics)

Abstract

An approach is proposed to improve static timing analysis for early TA-signoff closure at Very Deep Sub-Micron (V-DSM) nodes. At higher technology nodes (>100nm) the timing path delay is governed mainly by the cells, and hence the traditional corners (PVT) are sufficient to decide proper timing and functionality of SoC.

With technology scaling, the dimension of transistors and interconnects get reduced, resulting in decrease in the driving capability of cells and a simultaneous increase in the resistance of interconnects. This makes the contribution of interconnects to the path delay grow in comparison to the cell contribution. In addition, PVT variations increase with scaling, affecting the cells as well as interconnects. This results in drastic increase in timing analysis corners to achieve the desired functionality of system on chip (SoC). The growing demand of different functionalities in SoC has increased the number of modes and hence the number of analysis views (combination of modes and corners). That makes timing complex, which is further enhanced due to high frequency requirement.

Timing complexity makes STA technique time-consuming. To reduce time, multiple corners run in parallel using Distributed Multi-Mode-Multi-Corners (DMMMC) technique rather than MMMC technique which is operated sequentially. As SoC size increases (size – number of instances), it becomes more and more difficult for MMMC technique to perform the job adequately. DMMMC saves time but not so significantly as number of modes also increases simultaneously. Large SoCs require each corner to be run in parallel on multiple machines, and hence cost per corner increases.

To save host machine cost and machine run time, it is desired to reduce timing corners. This can be done by determining and exploiting the correlation between different parameters and their effects on timing. The proposed idea has been carried out by measuring slack variation across the available corners.

Three critical corners, out of the 12 corners provided by foundry, have been determined with ± 2-5 % variation: one for hold check and two for setup check. Now STA is limited to only these critical corners before final TA-signoff. By utilising these critical corners the machine license requirement as well as the machine run time can be reduced to 25%. This results in increased productivity of the organization while maintaining quality as well, by decreasing TA- cycle for early TA-signoff.

Acknowledgements

First and foremost, I would like to express my deep and sincere gratitude to my supervisor Mr. Mohit Verma for providing excellent guidance and encouragement throughout the journey of this work. Without his guidance, this work would never have been a successful one. I am also deeply grateful to Mr. Rajeev Srivastava for his valuable guidance during inception of the project. Finally, I am deeply thankful to my supervisor, Prof R. N. Biswas for provided me with lots of good advice and support with clear guidance throughout journey and encouragement for conquering every hurdle that I have encountered throughout the process. My special thanks to SoC team for the important technical discussions that helped me whenever I got stuck while working at STMicroelectronics. I am thankful to STMicroelectronics for supporting this work through an internship. My regards to all my friends here at IIITD especially Siddhartha Sankar Rout and Rahul Kumar Shah who made this journey a wonderful one. Last but not the least, I would like to thank my parents & family who have always supported me emotionally.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Background

With continuous scaling of technology node, VLSI industry boomed in last few decades. According to Moore's law, number of on-chip transistors approximately doubles in every 18 months and this trend is still continuing [1]. Exponential design evolution demanding high-end digital integrated circuits in terms of performance, speed and low power, deploy high frequency clocks. Increased clock speed in highly dense system-on-chip leads to timing complexity [2]. Process, Voltage, Temperature and Parasitic effects further enhance its complexity [3].

Timing is one of the major components to be considered while designing a semiconductor chip. It takes months in architecting and iterating the design to achieve the required timing constraint. Besides functional verification, the timing closure is a major element that decides when a chip can be released to the foundry for fabrication.

At higher technology nodes (100 nm and above) cell or gate size was sufficiently large while interconnects length was small with large cross section area [4]. Moreover, wide separation between routing of interconnects at higher technology node resulted into little crosstalk between adjacent wires. Hence, path delay was governed by cell alone whereas delay due to interconnect was ignored as shown in Figure 1.1(a).

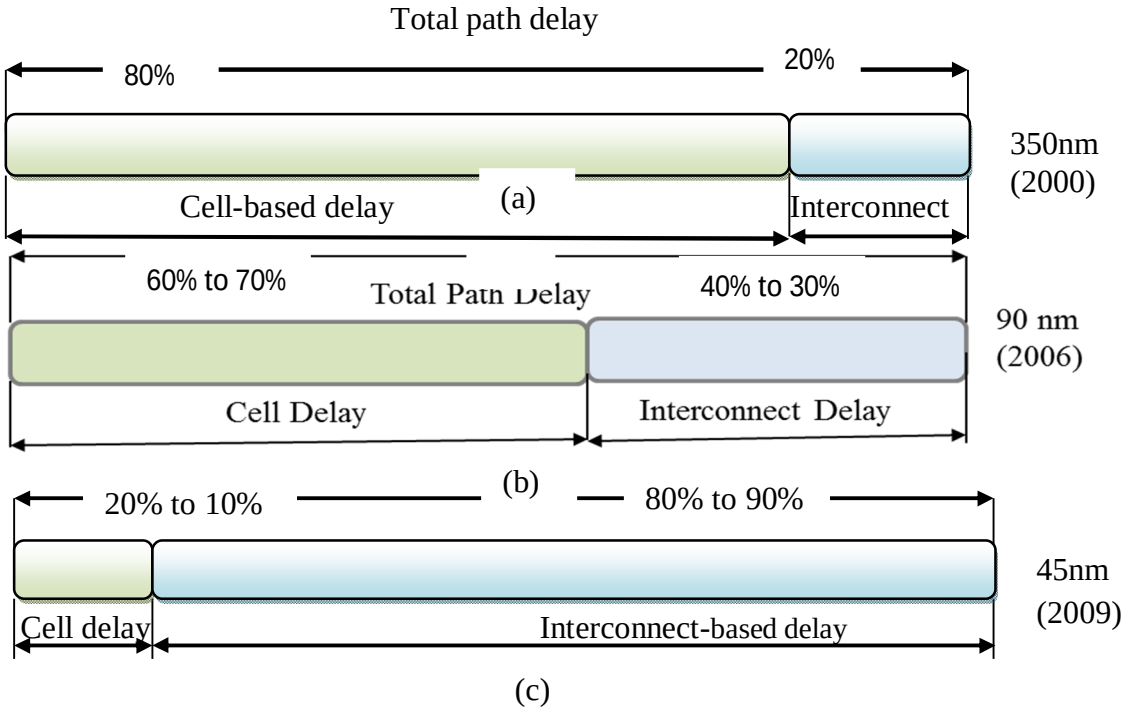


Figure 1.1: Cell and Interconnect Delay at Different Technology Nodes

With technology scaling length of interconnects increased and cross section area reduced. Thereby the interconnect delay started contributing considerably to the total path delay as shown in Figure 1.1(b) and 1.1(c). Moreover, routing is one of the important element to decide contribution of interconnect delay that is very high in Very Deep Submicron Nodes. The contribution of interconnects reaches 20% to 80% depending on length of wire and routing density.

1.2 Problem

At lower technology nodes, process, voltage and temperature variations affect circuit parameters and hence the delay. PVT variations lead to increase in the number of timing verification corners. Lowering of voltage due to scaling in very deep sub-micron (V-DSM) nodes creates a situation of temperature inversion [5] that again contributes to increase in the number of corners. Increased length and lowered cross section of interconnects in V-DSM nodes result in increased resistance of interconnects. The density of interconnects also increases in V-DSM nodes as they are placed closely, and that leads to cross talk effect and hence increased RC delay. Increased resistance of interconnects along with increased RC delay further raises the number of corners. Overall increase of analysis corners in V-DSM nodes increases the machine run time and license requirement and hence the cost.

With continuous rise in functions, the number of modes also increases [3] (Figure 1.2) which results in a drastic rise in analysis views as shown in Figure 1.3. Analysis view is combination of corners and modes which is carried out using static timing analysis (STA) tool for TA-signoff closure. Definition of corners, modes and analysis view has been represented in Table 1.1.

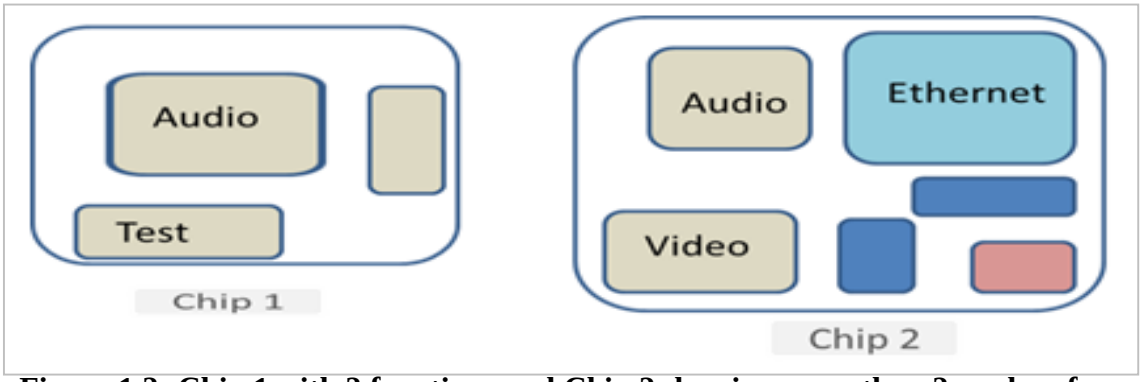


Figure 1.2: Chip 1 with 2 functions and Chip 2 showing more than 2 modes of functions

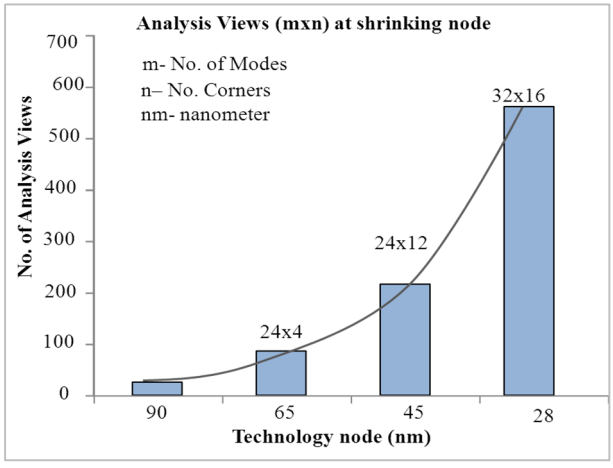


Figure 1.3: Technology node vs Number of timing analysis during STA [2] [3]

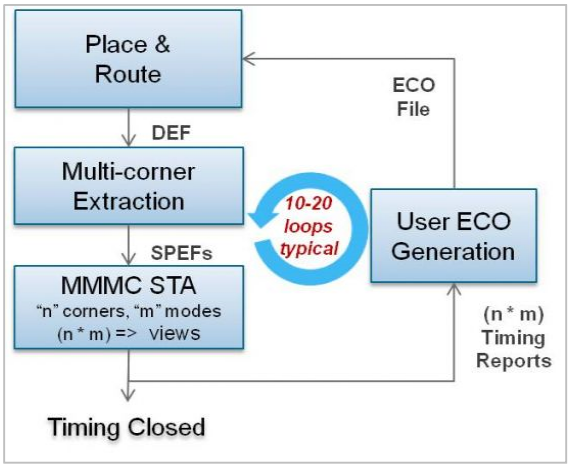


Figure 1.4: TA-Sign-off closure flow

Table 1.1: Definition of corners, modes and analysis view

Corners (n)	Each corner represents a combination of the parameters process, voltage, temperature (PVT) and parasitics (RC) that gives an extreme value (best/worst) of performance.
Modes (m)	It is the number of operations associated with the SoC such as Functionality, Scan, Test, BIST, Ethernet, Audio, Video, IO etc
Analysis View	= m*n

In addition to above raised issues, several timing iterations are being performed to achieve timing requirements as shown in Figure 1.4.

1.3 Motivation and Objective

Rapid increase in timing complexity makes it very difficult to meet the time to market requirements. All the available corners run in each mode and at least 8 iterations are required to close a single analysis view. The following illustration shows how the timing issue is getting enhanced [3]:

Foundry provides 12 corners at 45nm technology node.

If 10 functions are associated with a SoC;

Then total number of analysis view = 12x10 = 120

Machine iteration number = 120x8 = 960 ……(i)

To run all the corners in each mode independently and in parallel, the required number of machine licenses increases. Parallel distribution of corners (DMMMC) decreases run time but it increases license requirement; so tradeoff exists between run time and machine licenses. These problems motivate designers to minimize corners so that machine run time and license requirement can be reduced, thereby enabling one to meet time to market requirement and cut cost.

The objective of this work is to increase the productivity of the organization by reducing STA-cycle during early TA-signoff without affecting quality. This is done by identifying the critical corners out of many corners which can be iterated during static timing analysis at V-DSM node to meet the timing constraints.

1.4 Proposed Idea

In proposed method, timing analysis for early TA-signoff closure is limited to the critical corners, identified by measuring slack variation across all the corners available at 45nm technology node. Static Timing Analysis (STA) is carried out using industrial tools for different design using distributed multi-mode multi corners (DMMMC) analysis technique. Timing analysis is performed for several designs, where each design consists of several partitions that is categorised as simple partition or memory congestion based partition. Clock is distributed throughout the design from low frequency to high frequency.

It will be shown that out of 12 corners provided by foundry, only 3 dominant corners are sufficient for STA during pre-final signoff. Thus 75% of machine run time can be saved, and cost can be reduced to 25%. This would result in decrease of STA-cycle for early TA-signoff while quality remains unaffected, thereby increasing the productivity of the organization.

1.5 Thesis Organization

Chapter 2 start with discussion of theory and concept of STA, drain current and delay equations, effect of PVT and parasitic variation on delay calculation. This chapter also discusses some of the related previous work. Chapter 3 explains the methodology used to determine the slack variation across the available corners. Result analysis and their discussions are presented in chapter 4, followed by the conclusions in chapter 5.

Chapter 2: Theoretical Background

2.1 Introduction

This chapter describes the theory and concept behind static timing analysis along with the previous work done related to this field. The discussion includes setup, hold, required arrival time, actual arrival time and slack variation. Further it explains process-voltage-temperature variation and its effect on cell and interconnects hence on timing delay. Temperature inversion and its effect on delay at V-DSM nodes have also been discussed. Related previous work has been referred to.

2.2 Static Timing Analysis (STA)

Static Timing Analysis (STA) is standard technique available to verify the timing of the digital design. The STA is considered as static since the analysis of the design is carried out statically and does not depend upon the data values (or input vector) being applied at the input pins. An alternate approach used to verify the timing is the timing simulation that can verify the functionality as well as the timing of the design. In contrast to STA, simulation based timing analysis requires input signals for the observation of the chip behavior [6].

The purpose of static timing analysis is to validate whether the design can operate at the rated speed. That is, the design can operate safely at the specified frequency of the clocks without any timing violations.

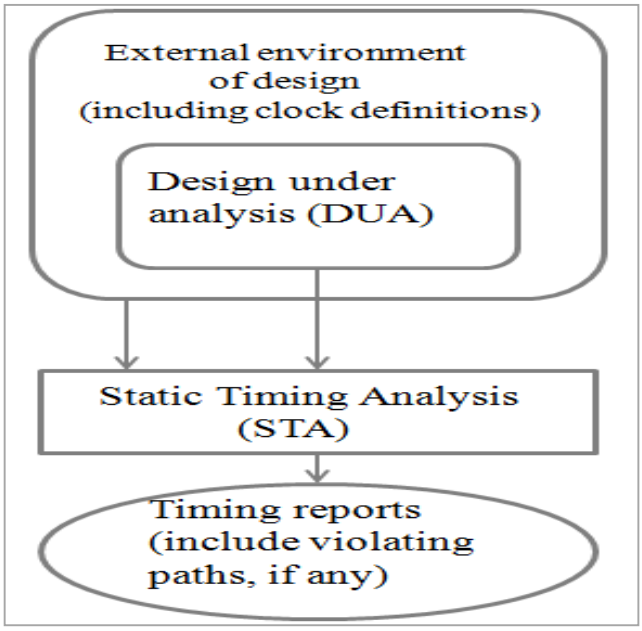


Figure 2.1: Basic functionality of Static Timing Analysis

The basic functionality of STA has been shown in Figure 2.1. DUA is the design under analysis for which timing check being carried out. Timing check such as setup check ensures that the data should arrive at a flip-flop within the given clock period. A hold check confirms that the data is held for at least a minimum time so that the flip-flop captures the intended data correctly. These checks ensure that the proper data is ready and available for capture.

Design under analysis is typically specified using a hardware description language such as VHDL or Verilog. The external environment, including the clock definitions and other constraints information using Synopsis Design Constraint (SDC) or similar format. Constraints specified in SDC are clock definition which includes information regarding generated and propagated clock, input and output delay, clock uncertainty for setup and hold check, false path (if any), latency etc. Other inputs requires to perform STA are .lib file and standard parasitic exchange format (SPEF) that contains parasitic information.

2.2.1 Why Static Timing Analysis

Static timing analysis is a complete and exhaustive verification of all timing checks of a design. Other timing analysis methods such as simulation can only verify a part of the design that get exercised by stimulus. While simulation carries out both timing as well as functionality verification, in contrary to it, STA is used for only timing analysis. As 10-100 million gates are placed on a single chip to generate input vectors, timing check becomes very difficult and complex. STA does timing check exhaustively but completely as it is independent of stimulus. Further, in a CMOS digital design flow, the static timing analysis can be performed at many different stages of the implementation as shown in Figure 2.2

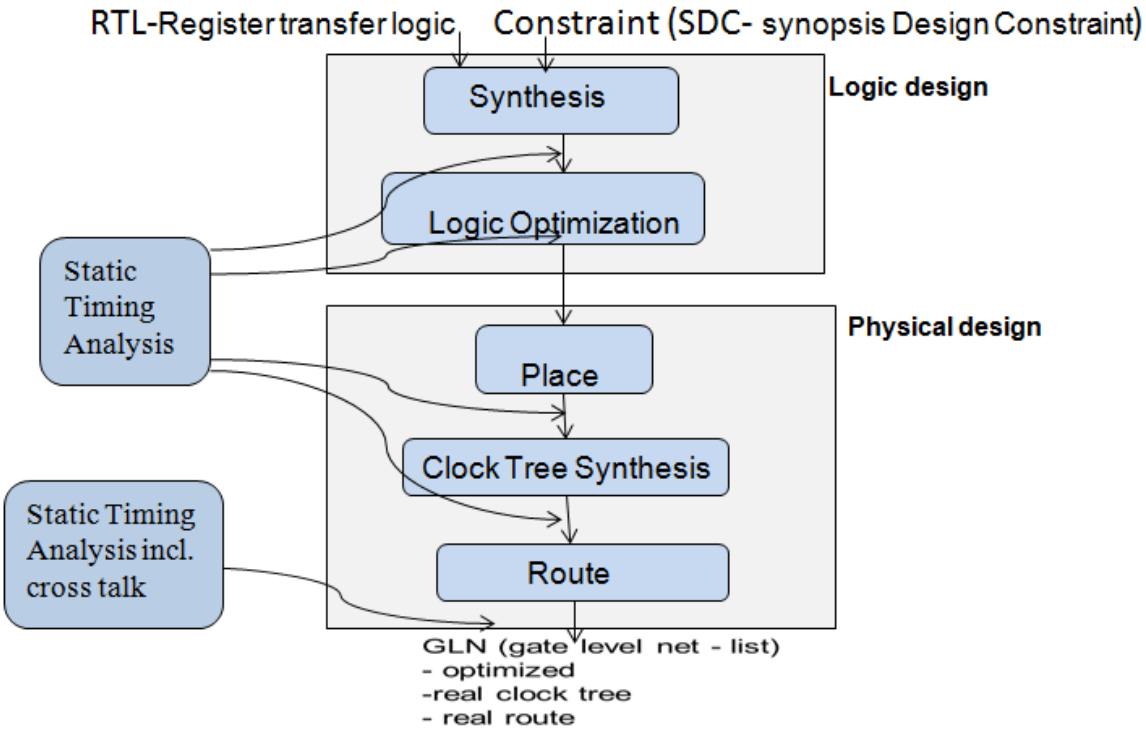


Figure 2.2: CMOS Design Flow at different level of SOC design

STA is rarely done at the RTL level as at this point, it is more important to verify the functionality of the design as opposed to timing. In gate level, again STA is performed as the full descriptions of the functional blocks are available even though all the timing information is not present. STA can also be run prior to performing logic optimization; the goal is to identify the worst or critical timing paths. STA can be rerun after logic optimization; to check whether any failing paths are still remaining that needs to be optimized [6].

At the logical design phase, ideal interconnect may be assumed since there is no physical information related to the placement and routing. The technique used at this stage is to estimate the length of the interconnect using a wire-load model. The wire-load model provides estimated RC based on the fan-out of a cell.

At the start of the physical design, clock trees are considered as ideal, that is, they have zero delay. Once the physical design start and after clock trees are built, STA can be performed to check the timing again. In fact, during physical design, STA can be performed at each and every step to identify the worst paths.

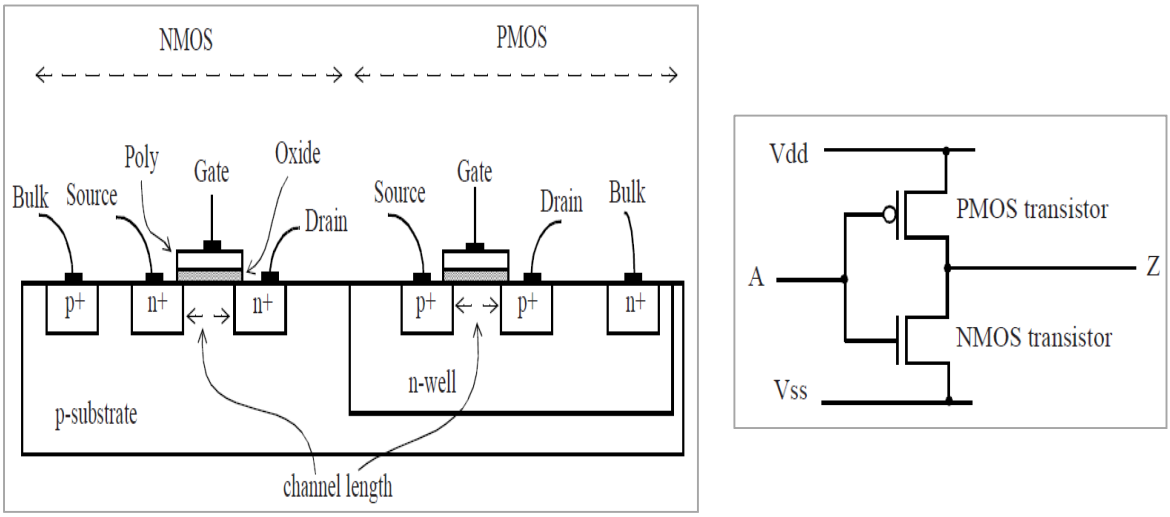
Since at VDSM nodes, contribution of cell delay to overall path delay is very small as compare to delay due to interconnect as shown in Figure 1.1 (c). Non-linear delay model (NLDM) and composite current source (CCS) are used for accurate cell delay calculation [7] [8]. For delay calculation lookup table is used which is characterized in the library.

2.3 STA Concept

2.3.1 CMOS Technology

A CMOS logic gate is built using NMOS and PMOS transistors. Figure 2.3 shows an example of (a) physical CMOS and (b) Symbolic CMOS inverter. There are two stable states of the CMOS inverter 0 or 1. When input A is low (at V_{SS} or logic-0), the NMOS transistor is off and the PMOS transistor is on, causing the output Z to be pulled to V_{DD}, which is a logic-1. When input A is high (at V_{DD} or logic-1), the NMOS transistor is on and the PMOS transistor is off, causing the output Z to be pulled to V_{SS}, which is a logic-0. In either of the two states described above, the CMOS inverter is stable at logic state low or high and no current drawn from input and V_{DD}.

For any CMOS logic gate, the pull-up (PMOS) and pull-down (NMOS) structures are complementary. CMOS technology is used to design basic cells called standard cell such as AND, OR, NAND, NOR, AND-OR-INVERT, OR-AND-INVERT and flip-flop which implement simple logic functions. These cells are basic building blocks to design simple as well as complex circuit on chip.



(a) (b)

Figure 2.3: (a) Physical representation of CMOS and (b) CMOS symbolic representation

2.3.1.1: Modelling of CMOS circuit

Figure 2.4 shows an equivalent abstract model for a CMOS cell. The main purpose of model is to extract timing behaviour of input and output regardless of internal transition. Equivalent model

is obtained by replacing cell with effective input, output capacitance and resistance. Wire also replace with its effective capacitance and resistance value.

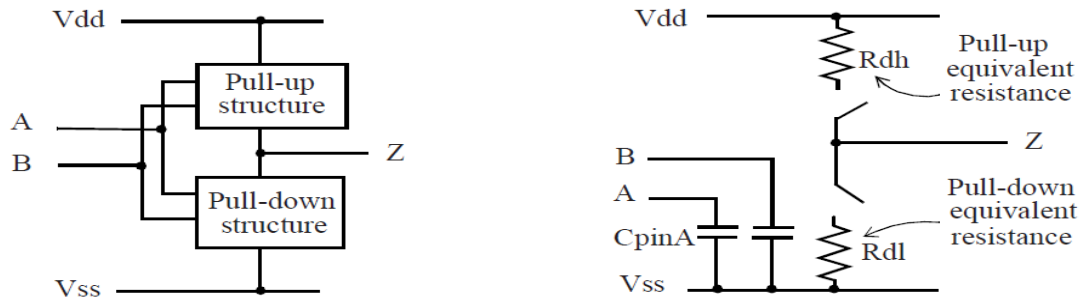


Figure 2.4: CMOS cell and its electrically equivalent model

2.3.2 Switching waveform

Figure 2.5 (a) represent equivalent waveform of CMOS inverter, (b) and (c) showing charging and discharging waveform respectively. Output transition times of the cell measured by measuring rise and fall delay.

$$V = V_{dd} * \left[1 - e^{-\frac{t}{R_{dh} * C_{load}}}\right] \ldots . (i)$$

$$V = V_{dd} * e^{-t/(R_{dl} * C_{load})} \ldots \ldots . . (ii)$$

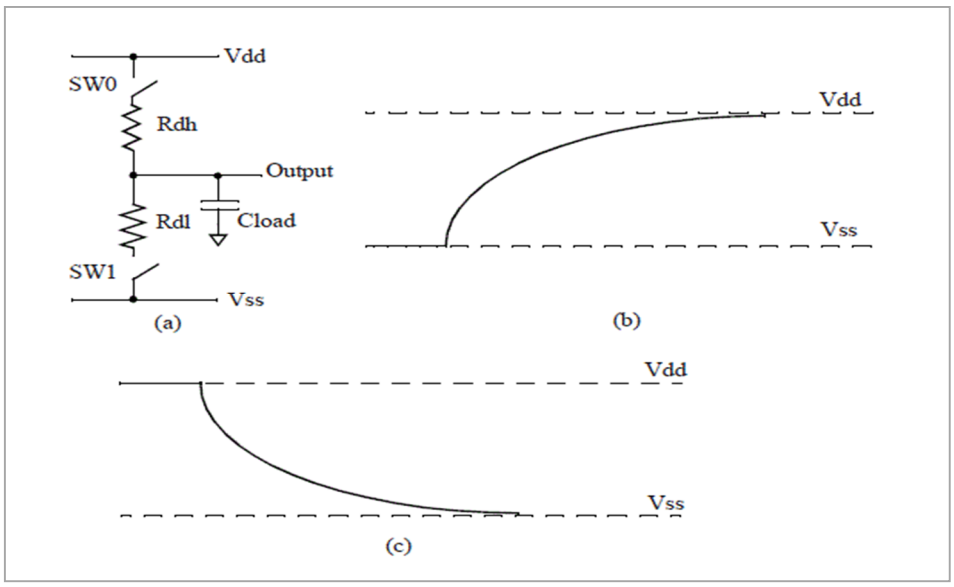


Figure 2.5: CMOS inverter charging and discharging

2.3.3 Propagation delay

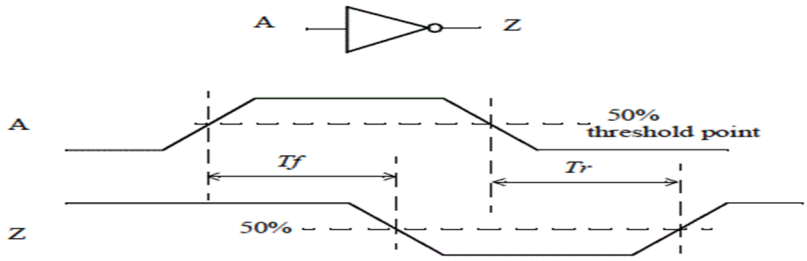


Figure 2.6: Propagation delay measurement waveforms

Propagation delay is measured by setting threshold point as shown in Figure 2.6, it provides transition from logic '0' to '1' or vice versa when data propagates from input to output.

2.3.4 Slew

Slew rate is defined as the rate of change.In static timing analysis, the rising or falling waveforms are measured in terms of whether the transition is slow or fast. **Slew** is typically measured in terms of the **transition time** that is inversely proportional to transition time.

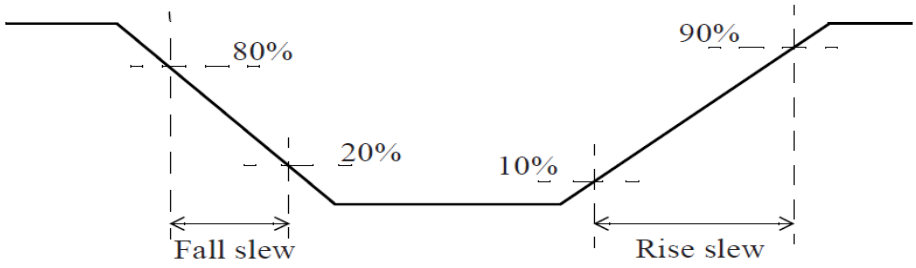


Figure 2.7: Waveform showing rising and falling edges

2.3.5 Skew

Skew is the difference in timing between two or more signals, maybe data, clock or both. For example, if a clock tree has 500 end points and has a skew of 50ps, it means that the difference in latency (source + network) between the longest path and the shortest clock path is 50ps.

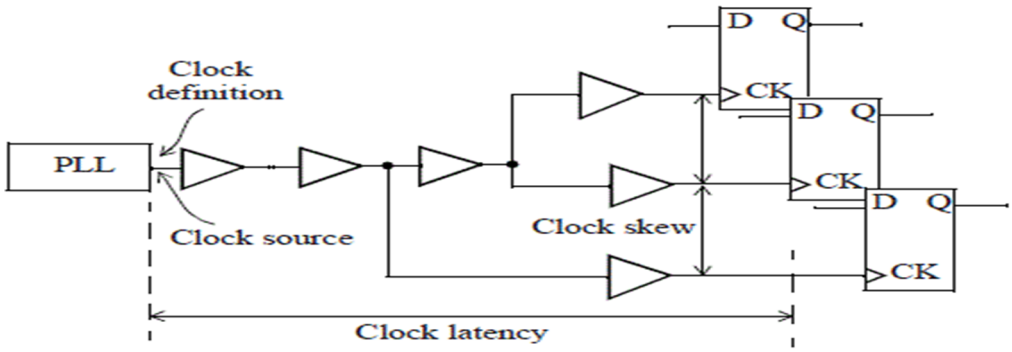


Figure 2.8: Skew representation from a small part of design

Clock latency is the total time it takes from the clock source to an end point and **Clock skew** is the difference in arrival times at the end points of the clock tree.

STA is often performed with ideal clock trees (clock can derive infinite sources with no delay) so that the focus of the analysis is on the data paths. In an ideal clock tree, clock skew is 0ps by default.

2.3.6 Uncertainty

Uncertainty measures jitters and skews in the signal at the end point of the path. Jitter is associated with clock hence it is not consider for hold check, as hold time is independent of clock.

2.3.7 Max and Min timing path

The total delay for the logic to propagate through a logic path is referred to as the path delay. This corresponds to the sum of the delays through the various logic cells and nets along the path. In general, there are multiple paths through which the logic can propagate to the required destination point. The actual path taken depends upon the state of the other inputs along the logic path. On the basis of multiple paths we can define two extreme paths.

First is the max (longest) path, which is defined the path that result in maximum delay (cell and net include combinational and sequential delay) when signal data or clock propagate through it, this is also refer as worst path delay. Second is the min (shortest) path that correspond to minimum signal delay when propagate through it, this also refer as best delay. Max path also refer as late path and min path as early path as shown in Figure 2.9.

A max path is often called a **late path**, while a min path is often called an **early path.**

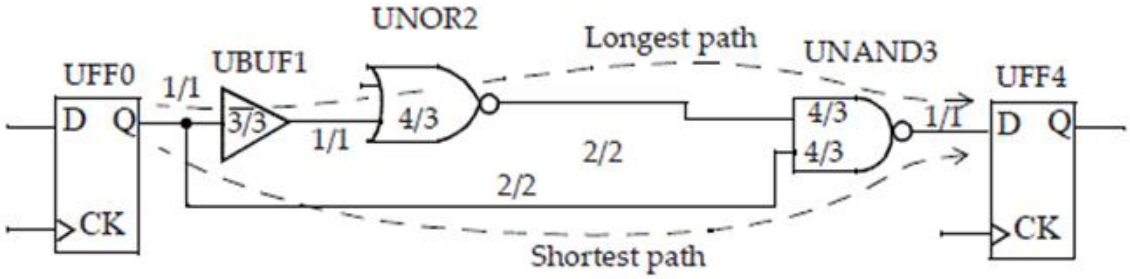


Figure 2.9: Max and Min timing paths in a design under analysis

2.3.8 Operating Condition

Static timing analysis is typically performed at a specific operating condition. An operating condition is defined as a combination of Process, Voltage and Temperature (PVT). Cell delays and interconnect delays are computed based on the specified operating condition. Operating condition governs timing complexity depending on how parameters affect cell and interconnect. Parameter variation increases with scaling consequently their effect on timing.

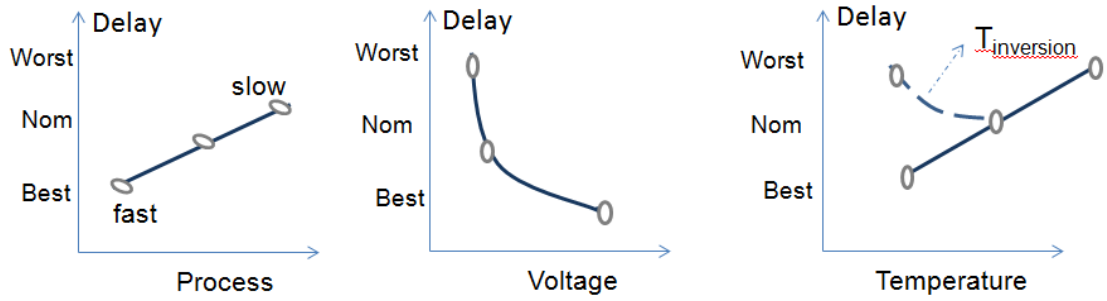


Figure 2.10: Delay variation with Process-Voltage-Temperature

2.4 Timing verification and reporting

In this chapter describes the checks that are performed as part of static timing analysis. These checks are intended to exhaustively verify the timing of the design under analysis.

2.4.1 Timing Paths

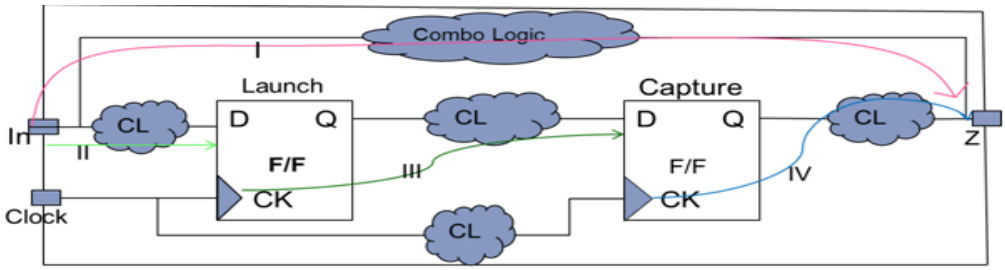


Figure 2.11: Timing paths in STA

There are four type of timing path considered during STA as shown in Figure 2.11. These are: (I) Input port to output port, (II) Input port to register, (III) Register to register and (IV) Register to output port. Various constraints such as setup and hold check carry out across any of them.

2.4.2 Required and Actual Arrival Path

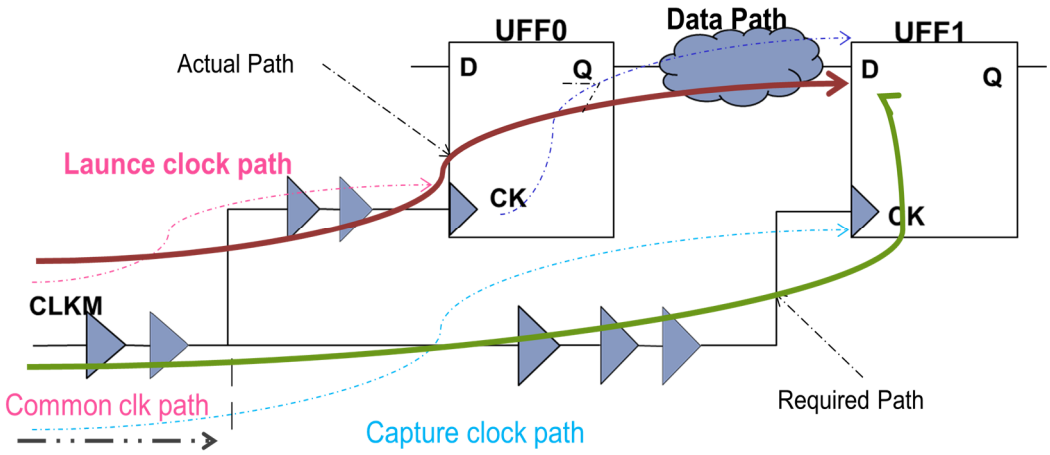


Figure 2.12: RT and AT using flip-flop pair

Required Path: Clock path give required path, as shown in Figure 2.12 and time taken by clock signal to reach end point from source defined as **required time.**

Arrival Path: Data path starting from launch path provide actual arrival path as shown in Figure 2.12. Thus time taken by data signal to reach end point defined as **arrival time.**

Note: All register must reliably capture data at desired clock edge.

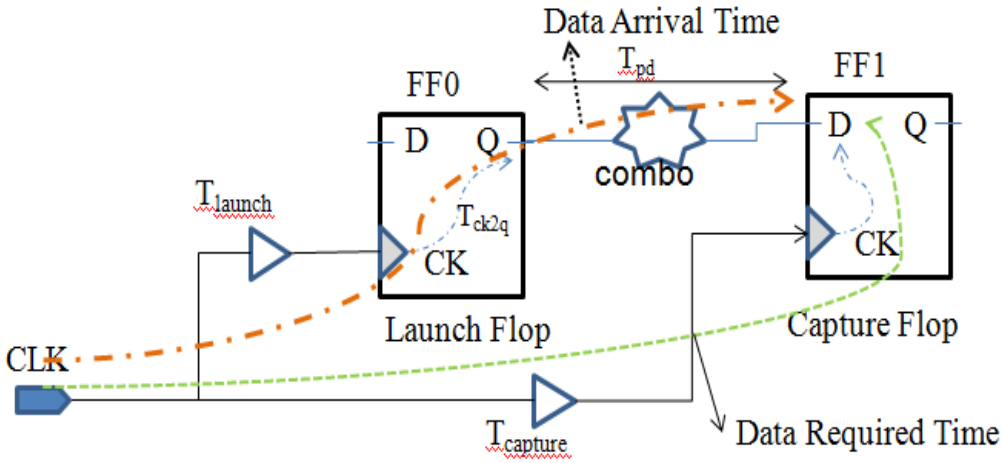


Figure 2.13: Register to register path

2.4.3 Setup Time (T_{setup})

A setup timing check verifies the timing relationship between the clock and the data pin of a flip-flop so that the setup requirement is met. In other word setup check ensures that the data is available at the input of the flip-flop before it is clocked in the flip-flop, so data should be stable for certain amount of time before arrival of active clock edge. This requirement ensures that the data is captured reliably into the flip-flop as shown in Figure 2.13 symbolic representation while 2.14 provide an example of setup timing check using waveform.

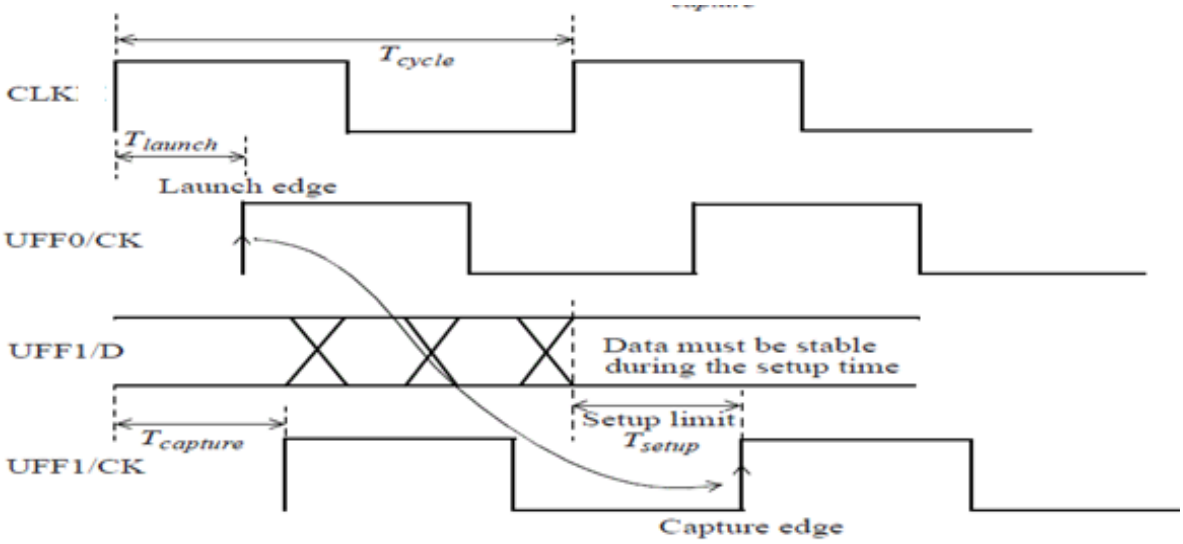


Figure 2.14: Clock waveform describing setup check

Setup Slack (S_{su})

It is defined as the difference between required arrival time and actual arrival time. Setup slack (>0 or< 0) will be positive if RT > AT and negative if RT < AT Setup check being carried out for timing check of max path (or latest path). Timing due to setup slack will be valid when $\mathbf{T_{setup} > 0}$. For worst setup slack RT_{min} and AT_{max} from the equation given below [9].

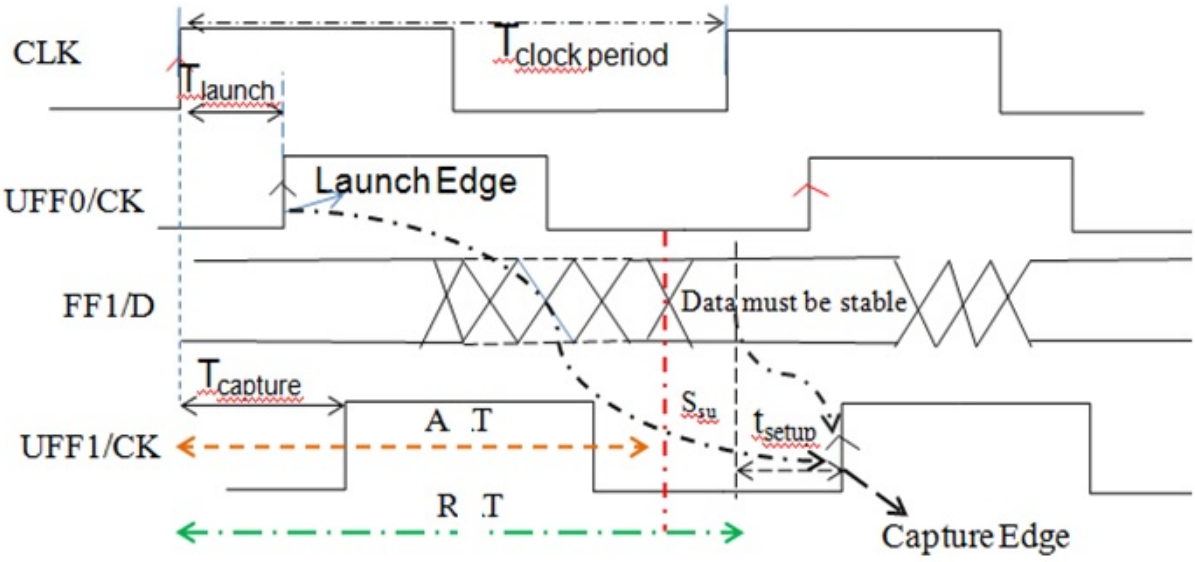


Figure 2.15: Setup slack (S_{su}) calculation

Mathematical expression for setup check

From figure 2.13 and 2.15

$$AT = T_{cycle} + T_{capture} - T_{setup}$$

$$AT = T_{Launch} + T_{ck2q} + T_{dp}$$

RT >AT ---- Valid setup path

Setup slack = RT – AT

Setup slack = $T_{cycle} + T_{capture} - T_{setup} - T_{Launch} + T_{ck2q} + T_{dp}$ ---- (2)

Equation (1) and (2) for setup slack calculation

T_{Launch} -- It is the delay of the clock tree of the launch flip-flop UFF0 (CID)

T_{dp} -- It is the delay of the combinational logic data path

T_{cycle} – It is the clock period.

$T_{capture}$ – It is the delay of the clock tree for the capture flip-flop UFF1 (OEAT)

Since the setup check poses a max constraint, the setup check always uses the longest or the max timing path. For the same reason, this check is normally verified at the slow corner where the delays are the largest.

Setup check reporting

(Path: Flip-flop to Flip-flop Path) –from figure 2.13

Table 2.1: Setup check reporting table

Setup Reporting

Header

```
Startpoint: UFF0 (rising edge-triggered flip-flop clocked by CLK )
Endpoint: UFF1 (rising edge-triggered flip-flop clocked by CLK )
Path Group: CLK
Path Type: max
```

Data Arrival

Point	Incr	Path
clock CLK (rise edge)	**0.00**	0.00
clock network delay (ideal)	0.00	0.00
UFF0/CK (DFF)	0.00	0.00 r
UFF0/Q (DFF) <-	0.16	0.16 f
UNOR0/ZN (NR2)	0.04	0.20 r
UBUF4/Z (BUFF)	0.05	0.26 r
UFF1/D (DFF)	0.00	0.26 r
data arrival time		0.26

Data Required

clock CLK (rise edge)	**10.00**	10.00
clock network delay (ideal)	0.00	10.00
clock uncertainty	-0.30	9.70
UFF1/CK (DFF)		9.70 r
library setup time	-0.04	9.66
data required time		9.66

Slack

data required time	9.66
data arrival time	-0.26
slack (MET)	9.41

The report shows that the launch flip-flop (specified by Start point) has instance name UFF0 and it is triggered by the rising edge of clock *CLK*. The capture flip-flop (specified by Endpoint) is UFF1and is also triggered by the rising edge of clock *CLK*. The Path group line indicates that it belongs to the path group *CLK*.

The Path Type line indicates that the delays shown in this report are all max path delays indicating that this is a setup check. This is because setup checks correspond to the max (or longest path) delays through the logic.

The Incr column specifies the incremental cell or net delay for the port or pin indicated. The Path column shows the cumulative delay for the arrival and the data required paths.

Clock specification used for this example.
Create_clock-name CLKM **-period** 10 **-waveform** {0 5} [**get ports** CLK] \
set_clock_uncertainty-setup 0.3 [**all clocks**] **set_clock_transition-rise** 0.2 [**all clocks**]
Set_clock_transition-fall0.15 [**all clocks**]

The launch path takes 0.26ns to get to the *D* pin of flip-flop UFF1- this is the arrival time at the input of the capture flip-flop. The capture edge (which is one cycle away since this is a setup check) is at 10ns. A clock uncertainty of 0.3ns was specified for this clock - thus, the clock period is reduced by the uncertainty margin. The clock uncertainty includes the variation in cycle time due to jitter in the clock source and any other timing margin used for analysis. The

setup time of the flip-flop 0.04ns (called library setup time), is deducted from the total capture path yielding a required time of 9.66ns. Since the arrival time is 0.26ns, there is a positive slack of 9.41ns on this timing path

2.4.4 Hold Time (T_{hold})

A **hold timing check** ensures that a flip-flop output value that is changing does not pass through to a capture flip-flop and overwrite its output before the flip-flop has had a chance to capture its original value. This check is based on the hold requirement of a flip-flop. From Figure 2.16 it is clear hold check calculated on the same edge at capture flip - flop as that of launch flip-flop.

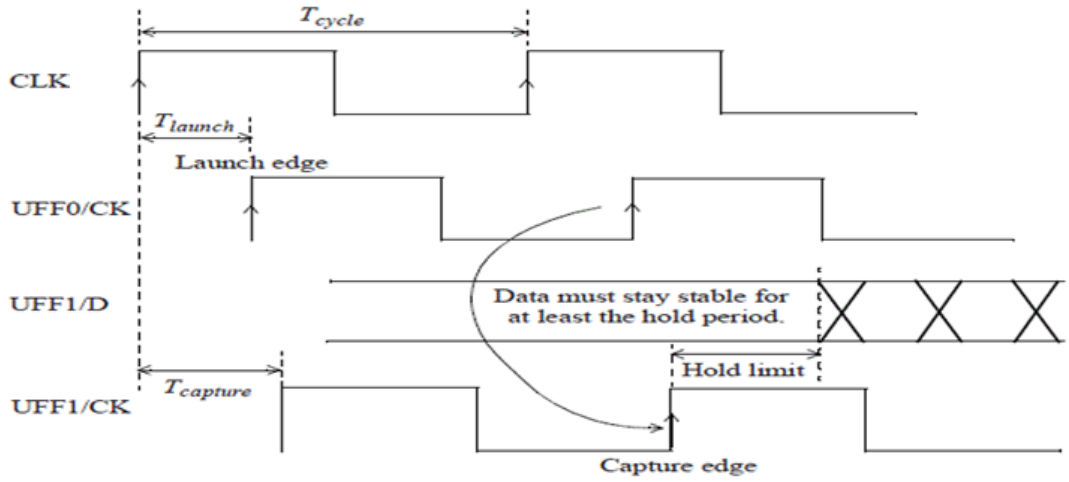


Figure 2.16: Waveform for hold check

Hold Slack (S_h)

It is defined as the difference between actual arrival time and required arrival time. Hold slack (> 0 or < 0) will be positive if AT > RT and negative if AT < RT hold check being carried out for timing check of min path (or shortest path). Timing due to hold slack will be valid when $\mathbf{T_{hold} > 0}$. For worst hold slack RT_{max} and AT_{min} from the equation given below.

Hold check applied for best case (having minimum path delay), clock path (constitute required timing) should be minimum and data path (constitute actual timing path)[9]

$$RT = T_{capture} + T_{hold}$$

$$AT = T_{launch} + T_{ck2q} + T_{dp}$$

---- (3)

$$AT > RT \text{ ---- Valid hold path}$$

$$\text{Hold slack} = AT - RT$$

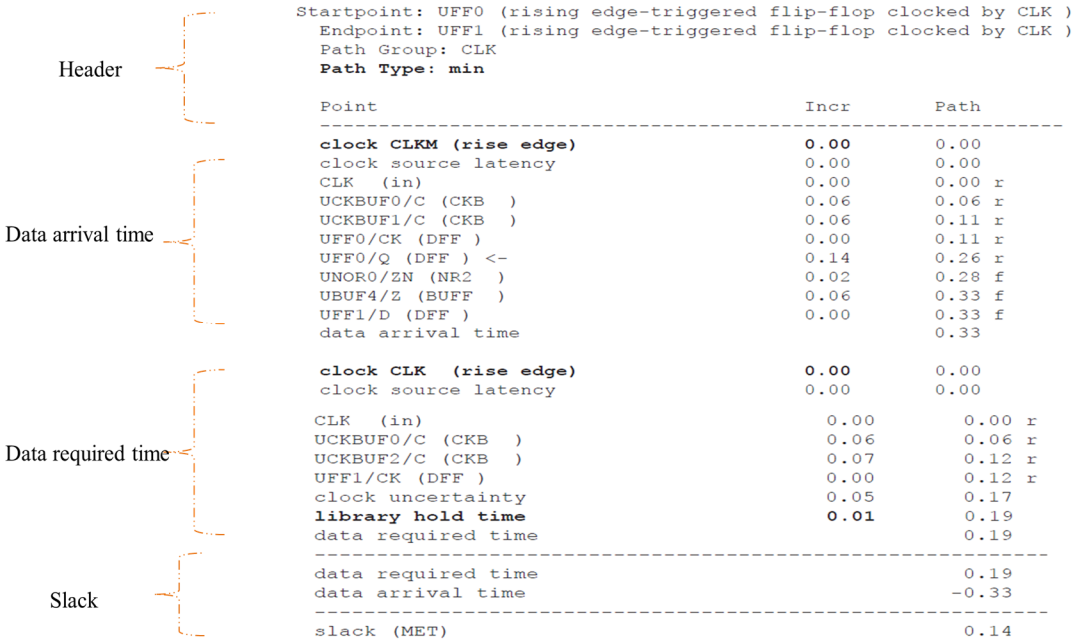
$$\text{Hold slack} = (T_{launch} + T_{ck2q} + T_{dp}) - (T_{capture} + T_{hold})$$

---- (4)

Hold check reporting

(Path: Flip-flop to Flip-flop Path) –from Figure 2.13

Table 2.2: Hold check reporting




```
Startpoint: UFF0 (rising edge-triggered flip-flop clocked by CLK )
  Endpoint: UFF1 (rising edge-triggered flip-flop clocked by CLK )
  Path Group: CLK
  Path Type: min

  Point                                              Incr       Path
  -----------------------------------------------------------------------
  clock CLKM (rise edge)                             0.00       0.00
  clock source latency                               0.00       0.00
  CLK   (in)                                         0.00       0.00 r
  UCKBUF0/C (CKB   )                                 0.06       0.06 r
  UCKBUF1/C (CKB   )                                 0.06       0.11 r
  UFF0/CK (DFF )                                     0.00       0.11 r
  UFF0/Q (DFF ) <-                                   0.14       0.26 r
  UNOR0/ZN (NR2   )                                  0.02       0.28 f
  UBUF4/Z (BUFF   )                                  0.06       0.33 f
  UFF1/D (DFF )                                      0.00       0.33 f
  data arrival time                                             0.33

  clock CLK   (rise edge)                            0.00       0.00
  clock source latency                               0.00       0.00
  CLK   (in)                                           0.00       0.00 r
  UCKBUF0/C (CKB   )                                   0.06       0.06 r
  UCKBUF2/C (CKB   )                                   0.07       0.12 r
  UFF1/CK (DFF )                                       0.00       0.12 r
  clock uncertainty                                    0.05       0.17
  library hold time                                    0.01       0.19
  data required time                                              0.19
  -----------------------------------------------------------------------
  data required time                                              0.19
  data arrival time                                              -0.33
  -----------------------------------------------------------------------
  slack (MET)                                                     0.14
```

In hold timing check path type is defined as min that indicates the cell delay values along the shortest path. The library specifies the hold time of flip-flop UFF1. The timing report shows four sections (i) Header: defines begin-end point, clock used and path type. (ii) Data arrival path: it is clock path, defines the time when active clock edge reached to capture flip-flop (iii) Data required path: it shows actual data path that include launch flop to capture flop data pin through combinational logic and (iv) Slack measurement which is obtained from difference of two different kind of path. The report shows that the ***data required time*** is 0.19ns to capture previous data safely. And new data arrives at 0.33ns which is actual arrival time, since the report shows a positive hold ***slack*** of 0.14ns.

2.5 Effects of PVT and Parasitics variations on STA

Process variations, caused by irregularities during the fabrication process, mainly affect the gate oxide thickness t_{ox} and carrier concentration, which in turn affect the threshold voltage, transistor currents and consequently path delays according to the following relationships. In addition, voltage and temperature variations during operation also affect the same parameters and hence the path delays [3][10].

2.5.1 Process

Process variation increases with scaling of technology. As a result, delay variations increase and hence the frequency performance of the design is impacted. This in turn impacts TAT (turnaround time) and timing yields.

Process variation are of following type (i) Random, (ii) Systematic, (iii) Intra- die (or within die or chip) and (iv) Inter-die (Variations between chips in the same wafer or in different wafers) as shown in Figure 2.5.1. Systematic variations involve exposure pattern variation that occurred in lithography process and silicon surface flatness variations caused by layout pattern density variation during the CMP (Chemical Mechanical Planarization).

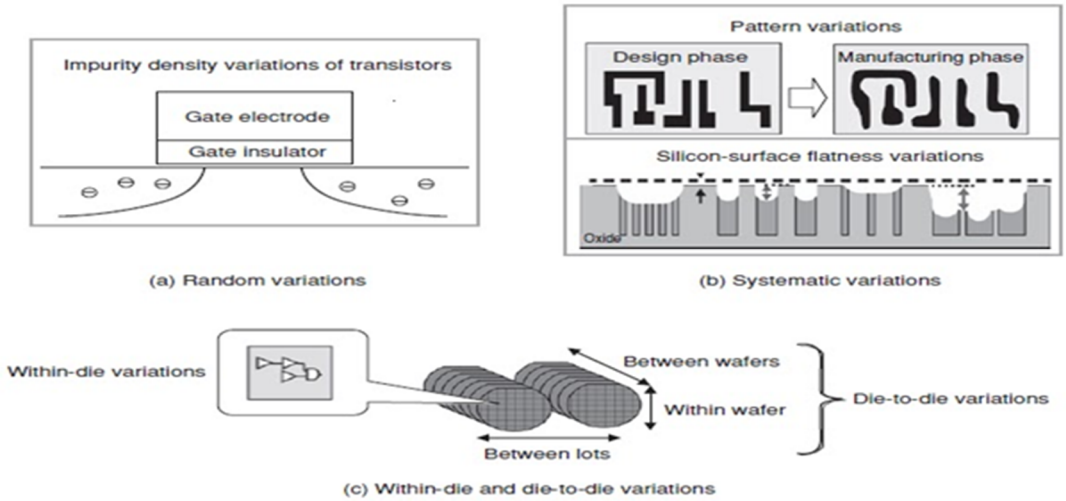


Figure 2.17: Process variation (a) Random, (b) Systematic and (c) Inter and Intra die [7]

Process parameter fluctuations in general can be organized as global variations and local variations. Local variations between identically laid-out devices arise from random microscopic processes variations. The random distributions arise from the variation of process parameters for example impurity concentration densities, oxide thickness and diffusion depths which result from varying operating or environmental conditions during the deposition or diffusion of the impurity dopants. Primary components affected by lithographic variation are; transistor gate length (L_g), and transistor width (W), which in turn affect transistor current [11]. A process is categorised as fast, normal and slow corresponding to best, nominal and worst delay respectively.

The fluctuations in the process parameters may result in the variation of sheet resistance and threshold voltage. However, the variation of threshold voltage can also be due to the variation of surface charge or change in the oxide thickness. The variations will impact the performance of device, which may exhibit wider variability leading to the degradation of yield in modern technologies and applications. This variation is due to non-uniform conditions during the deposition and/or the diffusion of the impurities.

Process variation also impacts clock skew [8]. Hence for generating a zero-skew or a bounded-skew design, routing tree is of primary importance for high-speed logic design.

Table 2.3: Process variation effect on path delay

Component / Process	L_{eff}	t_{ox}	V_{th}	Comment
Fast	↓	↓	↓	Path delay is minimum
Normal	—	—	—	
Slow	↑	↑	↑	Path delay is maximum

2.5.2 Voltage

At V-DSM nodes voltage scaling plays a significant role. Normally as V_{DD} decreases, driving capability of cell decreases. Therefore signal transition time increases resulting in large path delay. Moreover, combination of low voltage and temperature causes temperature inversion phenomenon, discussed below.

2.5.3 Temperature variations and Temperature inversion

Temperature variation affects speed, power and reliability by altering threshold voltage, mobility and saturation velocity. The resulting change in these parameter affects current that leads to timing failure, and result in communication errors between IP cores. MOSFET mobility, threshold voltage, and saturation velocity are related to temperature according to the following empirical expressions [12]-[14].

$$\mu(T) = \mu_0 (T/T_0)^{\alpha_\mu} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.1)$$

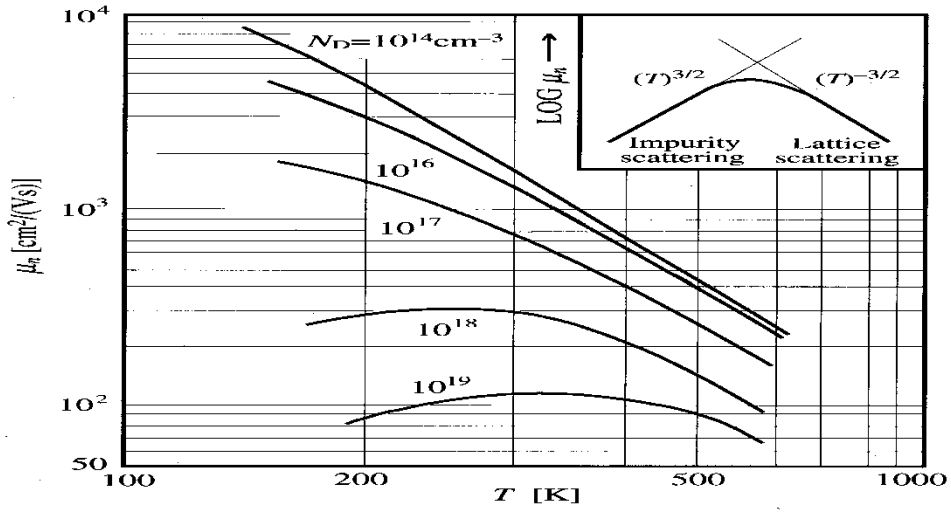
$$V_{th}(T) = V_{th0}\left[1 + \alpha_{V_{th}}(T - T_0)\right] \quad \ldots\ldots\ldots\ldots (2.2)$$

$$v_{sat}(T) = v_{sat0}\left[1 + \alpha_{v_{sat}}(T - T_0)\right] \quad \ldots\ldots\ldots\ldots\ldots (2.3)$$

Table 2.4: Various parameters

Parameter Symbol	Parameter	Comment
T ; T_0	Temperature ; Normal Temperature (25 ^{0}C)	
μ; V_{th}; v_{sat}	Mobility, Threshold voltage and Saturation velocity at temperature T	
μ_0 ; $V_{th0;}$ $v_{sat0;}$	Mobility, Threshold voltage and Saturation velocity at temperature T_0	
$\alpha_{\mu;}$ $\alpha_{V_{th}}$; $\alpha_{v_{sat0}}$	Mobility temperature coefficient (-1.3); Threshold Temperature coefficient (-3 mV/ ^{0}C); Velocity temperature coefficient (- 97 m/(s ^{0}C))	Negative Temperature coefficient

The dependence of mobility on temperature is shown in Figure 2.18. The behaviour at high temperatures is in accordance with equation 2.1 because saturation velocity is controlled mainly by **lattice scattering**. But at low temperature impurity scattering is the dominant deciding factor, resulting in a positive temperature coefficient.



(Log Scale)

Figure 2.18: Mobility vs Temperature [13]

Equations 2.1 – 2.3 can be used to obtain an expression for the drain current:

$$I_D = v_{sat}(T)W * P_s[V_{DD} - V_{th}(T)]^{\alpha} \quad \ldots\ldots\ldots\ldots\ldots\ldots (2.4)$$

where P_s is a technology-specific constant and α is a technology-specific exponent. The well-known expression given below approximate version of equation 2.4.

$$I_D = \left(\frac{1}{2}\right)\mu(T)C_{ox}\left(\frac{W}{L}\right)[V_{DD} - V_{th}(T)]^2 \quad \ldots\ldots\ldots\ldots (2.5)$$

Thus dependence of I_D on temperature has two opposite contributions – (i) a negative temperature coefficient due to the negative temperature coefficient of mobility and (ii) a positive temperature coefficient due to the negative temperature coefficient of V_{th}. The combination of these two effects leads to the phenomenon called **Inverted Temperature Dependence (ITD)**.

There exist two temperature regions: (1) the normal dependence region, where drain current I_D decreases with increasing temperature, and (2) the reverse dependence region, where I_D increases with increasing temperature. Between the two regions, there is a supply voltage where the impact of temperature on I_D is minimized. The voltage at which the temperature coefficient of drain current reverses polarity is called **crossover voltage**, **zero temperature coefficient (ZTC) voltage** or **inversion voltage.**

It is clear from equation 2.4 that the impact of the temperature coefficient of V_{th} is higher for lower value of the supply voltage V_{DD}. As the values of V_{DD} and V_{th} are linked with technology node, it is therefore not surprising that the **ITD** phenomenon is observed only at low supply voltage ($V_{DD} \leq 1V$) at 45nm node, while at 28nm technology it occurs for entire range of supply voltage.

2.6 Path Delay

The dependence of current on temperature gets translated to temperature dependence of path delay in a circuit because of the intrinsic relationship between delay and driver current. To find the cumulative delay along a path, the analysis tool adds the individual cell delays and

interconnect delays along the path, as shown in Figure 2.19. The cell delay information is contained in the library description of each cell.

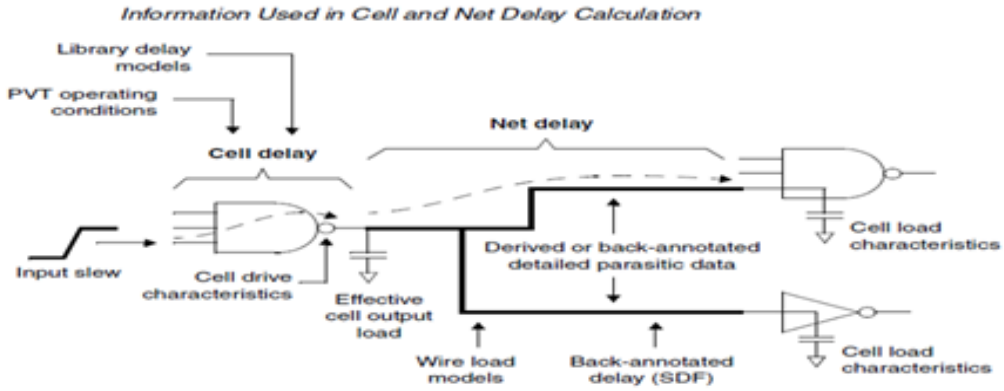


Figure 2.19: Path showing cell and net delays and components needed to calculate it

To calculate the interconnect delays, the analysis tool needs to know the characteristics of the driver cell that is driving the net, the load characteristics of the receiver cells driven by the net, and the resistance and capacitance (RC) characteristics of the wires in the net. The interconnect RC characteristics depend on the physical configuration and lengths of the wires, so these characteristics can be determined accurately only after layout has been completed.

For timing analysis performed after placement and routing, the analysis tool can extract the RC network accurately from the physical lengths of the wire connections and the known characteristics of the wire material. IC Compiler, for example, extracts RC information accurately from detail routing when available or from global routing where detail routing has not been completed.

Both cell and net delays depend inversely on I_D. Cell delay and drain current are related by equation 2.6.

$$\text{Cell delay} = \frac{C_{load} V_{dd}}{I_D} \quad \ldots\ldots\ldots\ldots\ldots\ldots (2.6)$$

Interconnect delay depends on driver current and the time constant of the equivalent RC circuit given in equation 2.7.

$$\tau_{pd} = (R_{dri} + R_{net})\,(C_{fan} + C_{net}) \quad \ldots\ldots\ldots\ldots (2.7)$$

For accurate delay calculation at V-DSM node various delay model used are Effective Capacitance Non-Linear Delay Model (NLDM) and Effective Composite Current Source (ECCS) [14]. For this library characterization being carried out that uses look-up-table (LUT) technique to find delay of the timing path. Look-Up-Table stores two variables: (i) input transition and (ii) C_{load} (Effective output capacitance of Cell). For accurate delay calculation, CCS library is used, which contains all the information of drivers as well as receivers.

At higher voltages, say 1.20V, mobility is the dominant factor deciding the temperature dependence of I_D, and the effect of change in V_{th} with temperature is negligible. So I_D has a negative temperature coefficient and hence delay increases as temperature rises as shown in Figure 2.20.

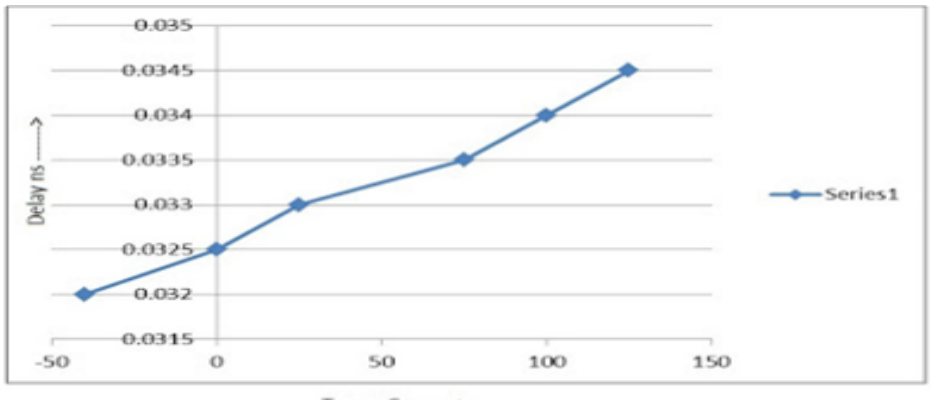


Figure 2.20: Delay Vs Temperature at high V_{dd} [15]

At low voltages ($V_{DD} \leq 1V$), V_{th} is the dominant parameter deciding temperature dependence, and hence delay decreases as temperature increases as shown in Figure 2.21.

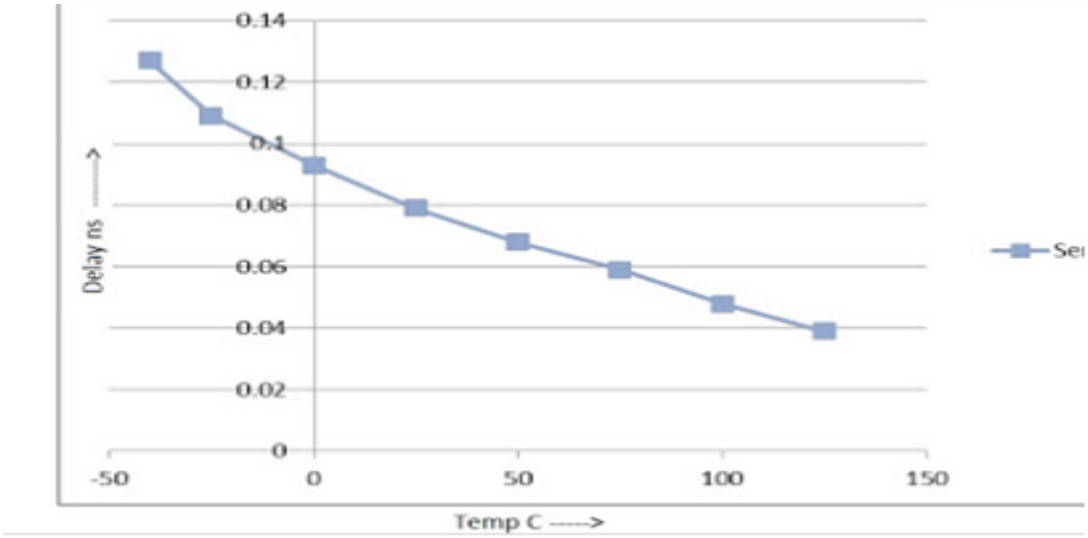


Figure 2.20: Delay vs Temperature at low V_{DD} [15]

Thus cell delay is high for low supply voltage and low temperature at V-DSM nodes, while for higher technology, cell delay highest at high temperature. This fact will be utilised in chapter 3 to arrive at an efficient strategy for STA.

2.7 Previous work and proposed work

Continuous shrinking of CMOS technology results in higher speed, higher packaging density which consequently raises complexities associated with the routing of interconnect. High operating frequencies in orders of GHz are common in modern V-DSM circuits. As the system clock period decreases, the **pessimism imposed** by timing verification tools becomes less acceptable. Therefore more accurate characterization and verification techniques are desirable [16]. The STA tools rely on data described in the cell libraries to analyse the circuit. Therefore the characterization of the individual cells in cell libraries is highly critical in terms of the accuracy of the STA results [17]–[20].

The setup and hold time constraints of the sequential cells are used to verify the timing of a synchronous circuit. Inaccurate characterization of timing constraints causes the STA results to be either highly optimistic or pessimistic. Both need to be avoided as optimistic

timing may be the cause of failure in the circuit and pessimistic timing probably degrade the operating speed.

Stojanovic and Oklobdzija [21] discuss over-optimism or pessimism in STA is primarily due to the “independent” characterization of the timing constraints, although these constraints (including CLK-to-Q delay) are “interdependent. Therefore constraints should be characterized interdependently to remove optimism or pessimism in STA. In [22], a timing-constraint characterization that minimizes the sum of the CLK-to-Q delay and the setup time is proposed by modelling CLK-to-Q delay of sequential cell. It utilizes the dependency between the CLK-to-Q delay and the setup time that helps in achieving 50–60 ps decrease in the clock period during STA.

Emre Salman et al. [16] emphasise on interdependence of setup and hold time. In this the library characterization is carried out by considering setup and hold constraints interdependency to achieve improvement in timing verification with STA tool unlike [21] and [22]. The setup and hold times are not independent [23], but rather these constraints are a function of the counterpart skews (hold skew for the setup time and setup skew for the hold time). Setup time decreases as the hold skew increases and that the hold time decreases as the setup skew increases. Thus, the smallest setup and hold times occur when the counterpart skews are the largest.

Howick, [24] investigates CLK–to-Q delay and skew relationship. He has developed an exponential relationship between CLK–to-Q delay and skew, such that skew decreases as CLK –to-Q delay increases and vice-versa.

The work discussed above, performs STA improvement by exploiting the property of inter-dependency between various parameter such as setup, hold, skew and CLK-to-Q delay [16] [21-24]. The works illustrated here are implicit to the STA which makes them corner-independent. So for proper TA-signoff all corner runs are required.

In our proposed method emphasis is on minimizing timing analysis corners for early TA- Signoff unlike the previous work. This work improves STA explicitly by reducing number of timing analysis corners, which result in reduction in machine runtime as well as cost.

CHAPTER 3: Proposed Methodology

3.1 Introduction

Formation of corners as a consequence of PVT and Parasitics variations at V-DSM nodes has been explained in the last chapter, followed by some theoretical as well as empirical considerations and discussions on expected slack variations. Finally, the experimental setup for STA has been described, followed by the procedure for STA analysis over different designs and varying frequency.

3.2 Corners Formation

Timing analysis corners are provided by foundries. Every corner is a combination of Process-Voltage-Temperature and Parasitics, which defines a specific operating condition. The classification of corners is explained in the sections below.

3.2.1 PVT Corners

Table 3. 1: PVT corners and its effect on delay

Corner corresponding to	Manufacturing Process (P)	Environmental condition		Delay
		Voltage(V)	Temperature (T)	
Best Case	Fast	High	Low	Minimum
Worst Case	Slow	Low	High	Maximum

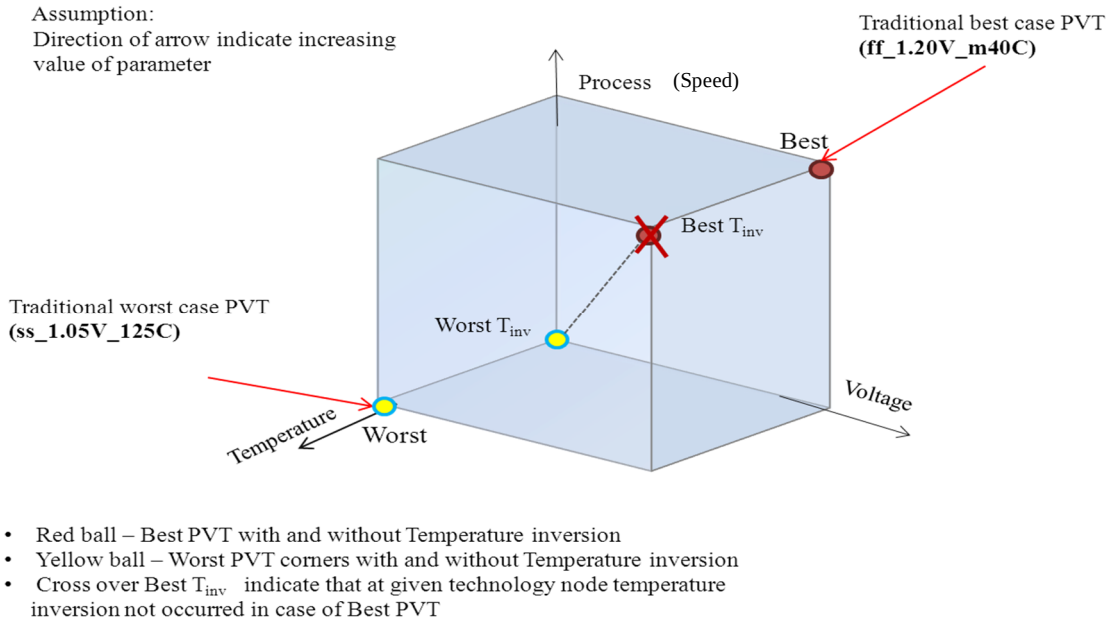


Figure 3. 1: PVT corners at 45nm technology node

As shown in Figure 3.1 axes of the cube are voltage, temperature and process, direction of arrow indicating low → high for voltage, temperature and speed (slow-slow → fast-fast for nmos-pmos) respectively. Traditionally two extreme corners are used at higher technology (> 90nm) nodes depending on PVT state or value: (i) best PVT and (ii) worst PVT corners that

correspond to minimum and maximum delay respectively. At V-DSM nodes (< 45nm) due to temperature inversion one more corner is added in each traditional corner. Fortunately in our case at 45nm technology node temperature inversion does not occur in the best PVT case. Foundry provides these corners depending on technology. Therefore, three PVT corners are sufficient to carry out STA accurately at 45nm technology node. Worst and best PVT status are shown in Table 3.1.

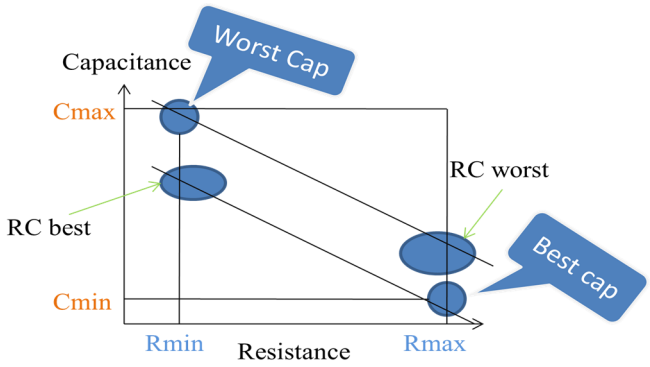
3.2.2 Parasitic Corners

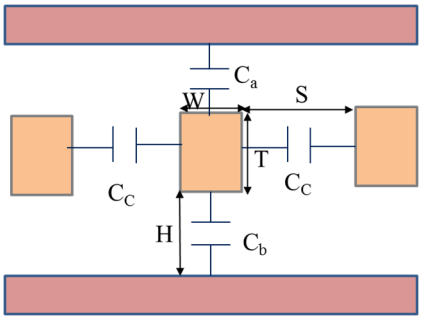
At higher process node (≥ 90nm) overall delay is governed by cell alone and delay due to interconnect is ignored. Therefore total load (depending on number of fan-out) is represented by capacitance (C_{load}) only. Consequently foundry provides two parasitic corners Cmax and Cmin which correspond to maximum and minimum delay respectively.

With technology scaling, interconnect cross-section decreases significantly, whereas length of wire and packaging density increases. Therefore it is not possible to ignore the delay offered by interconnects for correct timing analysis. Hence foundry came with two more parasitic corners as RC product, RCmax (or delay corner) and RCmin. (or Cross-talk corner). Table 3.2 explains parasitics corners and their R and C conditions, as shown in Figure 3.2 (a). C_c represents coupling capacitance with adjacent interconnects; C_a and C_b are the capacitances with top and bottom metal layers as shown in Figure 3.2 (b). Cg (ground capacitance) represents the net capacitance including the effect of C_a, C_b and fringing capacitances.

Table 3. 2: Parasitic corners

Higher Technology Nodes	Cworst (or Cmax)	maximize C and minimize drive	max-path analysis
	Cbest (or Cmin)	minimize C and maximize drive	min-path analysis
Lower Technology Nodes	RCworst (or RCmax) (Delay corner)	C_C is min, and Cg x R is max	max path analysis
	RCbest(or RCmin) (Cross-talk corner)	C_C is max, and Cg x R is min	min path analysis





(a) **(b)**

Figure 3. 2: (a) Parasitic corners and (b) Parasitic capacitance

So the parasitic corners are decided by effective capacitances only and are classified as Cmax (or Cworst) and Cmin (or Cbest) at higher technology nodes (≥ 90nm). Cbest result in

smallest delay and Cworst corresponds to largest delay and RC product of driver (or cell) is always greater than interconnect RC product. At V-DSM nodes (≤ 45nm), RCmax and RCmin add two more parasitics corners. RCmax (or RCworst) and RCmin (or RCbest) correspond to maximum and minimum delay respectively. Therefore RCmax is used for max path analysis and RCmin for min path analysis.

3.2.3 Corner classification diagram at V-DSM node (45nm)

Foundries classify 12 corners at 45nm technology depending on PVT (including temperature inversion) and parasitic (or interconnect) effect on current. Corner classification as shown in Figure 3.3 is obtained by combining PVT and parasitics corners.

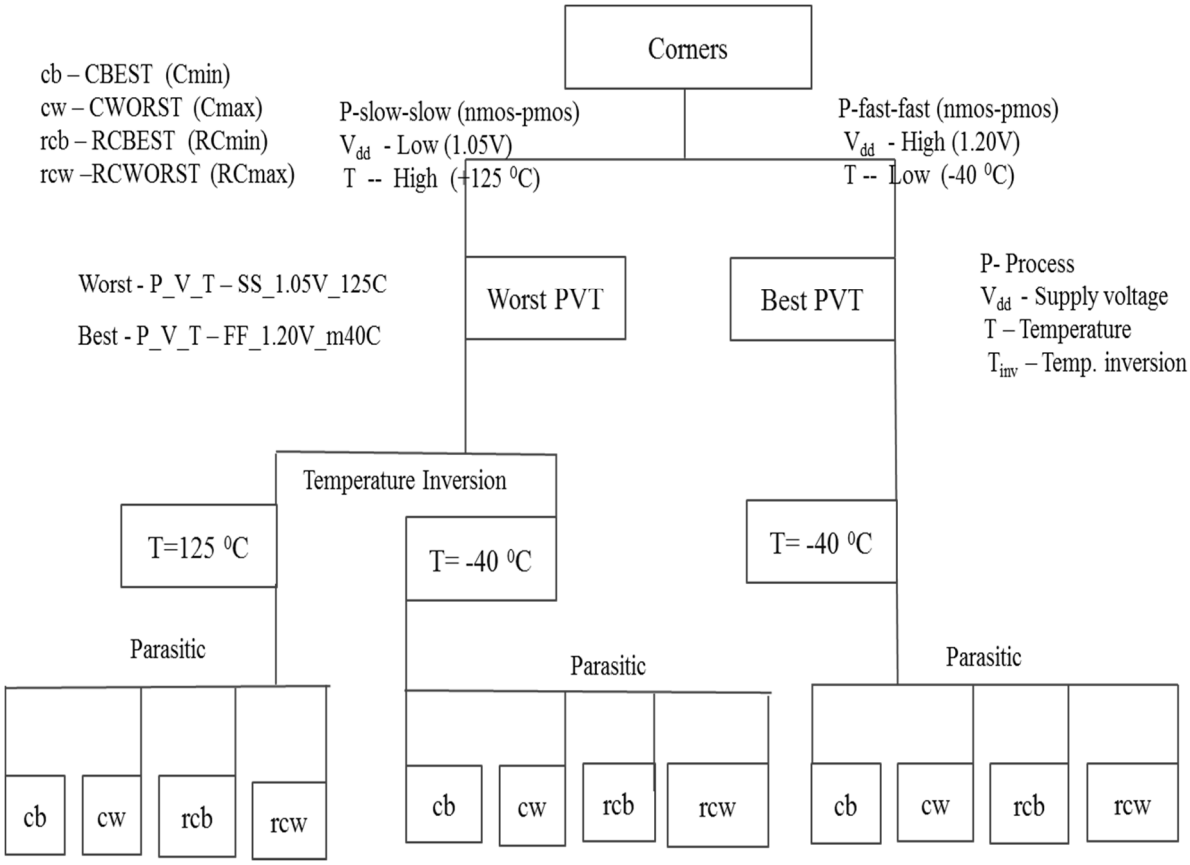


Figure 3. 3: Corners provided by foundry at 45nm technology

3.3 Assumptions, Specifications and Limitation

Table 3. 3: Experiment consideration 1

Parameter	Description
Design : Partition	D1:P1; D1:P2, D1:P3 etc.
Technology Node	45nm (CMOS)
Size of SoC	3.5 million instance
Corners by Foundry	12
Number of Modes	25
Time Taken and Memory Uses by Each Analysis View	7 hours and 35 GB
Number of Machine Runs in Parallel such that 16 Jobs Analyse in Parallel	16
Technique Used for STA	DMMMC (Distributed Multi Mode Multi Corners)
Corners (N) (PVT and Parasitic)	-- Process-Voltage-Temperature (PVT) –Environment based -- Parasitic (RC) – Interconnect based
Modes (M)	Number of Function Associated on a SoC
Analysis View	M*N

Table 3.3 explains various considerations that has been taken in experiment.

Table 3. 4: Experiment consideration 2

Process	Fast	t_{ox}– thin , L_{eff}– small, V_{th}– low, K=high
	Slow	t_{ox}– thick , L_{eff}– large, V_{th}– High, K=low
Supply Voltage		Minimum =1.05V, Maximum = 1.20V
Operating Temperature		Minimum = 40C, Maximum = 125C
Threshold Voltage		LVT=0.4 – 0.45V;RVT=0.5-0.55V; HVT= 0.7V

Table 3.4 defines key parameter values and ranges in which all the operations being carried out.

Table 3. 5: Corner representation and meanings of various terms used

Corners	Corners representation	Comments
C1	Func_ff40_1.20V_m40C_cbest	Func – Indicates mode ff and ss indicate – process fast-fast and slow-slow respectively ff40 – fast-fast (nmos - pmos) and technology node ss40 –slow-slow (nmos - pmos) and technology node
C2	Func_ff40_1.20V_m40C_cworst	
C3	Func_ff40_1.20V_m40C_rcbest	
C4	Func_ff40_1.20V_m40C_rcworst	
C5	Func_ss40_1.05V_m40C_cbest	Supply voltage (V_{dd}) –1.05V (low)--- 1.20V (high) Supply voltage – 1.20V (high) (10 to 12% above than normal V_{dd}) Supply voltage – 1.05V (high) (10 to 12% below than normal V_{dd})
C6	Func_ss40_1.05V_m40C_cworst	
C7	Func_ss40_1.05V_m40C_rcbest	
C8	Func_ss40_1.05V_m40C_rcworst	
		Temperature : (-40 ^{0}C - 125 ^{0}C) m40 ^{0}C --- minus 40 ^{0}C 125 ^{0}C ---- plus 125 ^{0}C
C9	Func_ss40_1.05V_125C_cbest	
C10	Func_ss40_1.05V_125C_cworst	
C11	Func_ss40_1.05V_125C_rcbest	Parasitic : interconnect resistance ignored Cbest or cbest (cb)– Cmax Cworst or cworst (cw) – Cmin With interconnect ($R_{net \neq 0}$) RCbest or rcbest (rcb) – RCmax (product) RCworst or rcworst (rcw) – RCmin (product)
C12	Func_ss40_1.05V_125C_rcworst	
Colour for slack comparison using excel sheet	Light Yellow – Highest slack value	
	Dark Orange – Lowest slack value	

For example a corner is elaborated as:

C1-Func_FF40_1.20V_m40C_Cbest – This corners corresponds to functional mode, fast-fast process, high voltage, low temperature and minimum capacitance (Cmin)

C2-Func_SS40_1.05V_125C_Cworst -- This corners corresponds to functional mode, slow-slow process, low voltage, high temperature and maximum capacitance (Cmax), similarly for other corners.

3.4 Proposed Methodology

Total number of corners shown in Figure 3.3 and their nomenclature with description provided in Table 3.3. Timing verification is an important parameter instead of functional verification that is responsible to release of chip to foundry for manufacturing. For SoCs timing verification carries out using STA tool.

3.4.1 Max and Min Timing Check

There are two main types of timing check; setup and hold in any design - IPs or SoCs. As discussed earlier in chapter 2, setup timing check is required for maximum path delay; on the other hand, hold timing check is required for minimum path delay. According to the definition of setup timing check for correct timing and hence setup slack, data must be available at the input pin of the capture flip flop at least by setup time (provided in the library) before arrival of active clock edge. Therefore latest data or max path arrival requires setup timing check for worst case analysis. In contrast, hold timing check is carried out for min path for worst case analysis as it corresponds to early data arrival.

If latest (slowest data arrival path or max path) data meets its setup timing then all other arrival of data is safe for setup timing. Similarly for hold check, if earliest data arrival (fastest data arrival path or min path) meets its hold timing check then all other timing paths satisfy hold check in which support minimum timing paths.

3.4.2 PVT and Parasitics

Corner classification on the basis of PVT is traditionally divided into (i) best and (ii) worst case. The best and worst case represent minimum delay and maximum delay respectively. Best PVT corresponds to fast-fast process, high voltage and low temperature and worst PVT represents slow-slow process, low voltage and high temperature traditionally for CMOS technology as shown in Table 3.1.

Figure 3.4 is a conceptual representation of a net connecting a driver output to the following buffer. C_{fan} represents the total lumped equivalent capacitance of the net. Fast-Fast process corresponds to higher driving ability of both nmos and pmos transistors, resulting in minimum delay, and vice-versa for slow-slow process. High voltage draws high current and hence increases driving capability of the transistors, and vice-versa for low voltage. In addition, low temperature further increases driving capability of the transistors because of high mobility. Therefore best PVT is defined for fast process, high voltage and low temperature, and vice versa for Worst PVT condition.

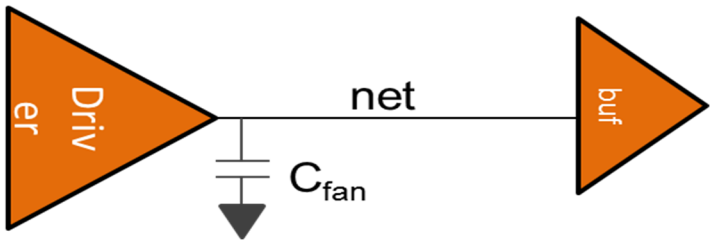


Figure 3. 4: Parasitic net capacitance

Number of corners provided by foundry at 45nm technology node is 12 as shown in Table 3.3. But the number of relevant corners required for timing checks can be minimised on the basis of the following considerations, assuming the main effect of path delay being data path delay, the clock path delay being governed by clock tree based on different design considerations:

i. The worst case scenario for ***setup*** timing check is ***maximum path delay.***
ii. The worst case scenario for ***hold*** timing check is ***minimum path delay.***
iii. Temperature inversion is effective only at ***low supply voltages.***.

Figure 3.5 explains division of setup and hold type of timing check by considering correlation among various corner deciding parameters.

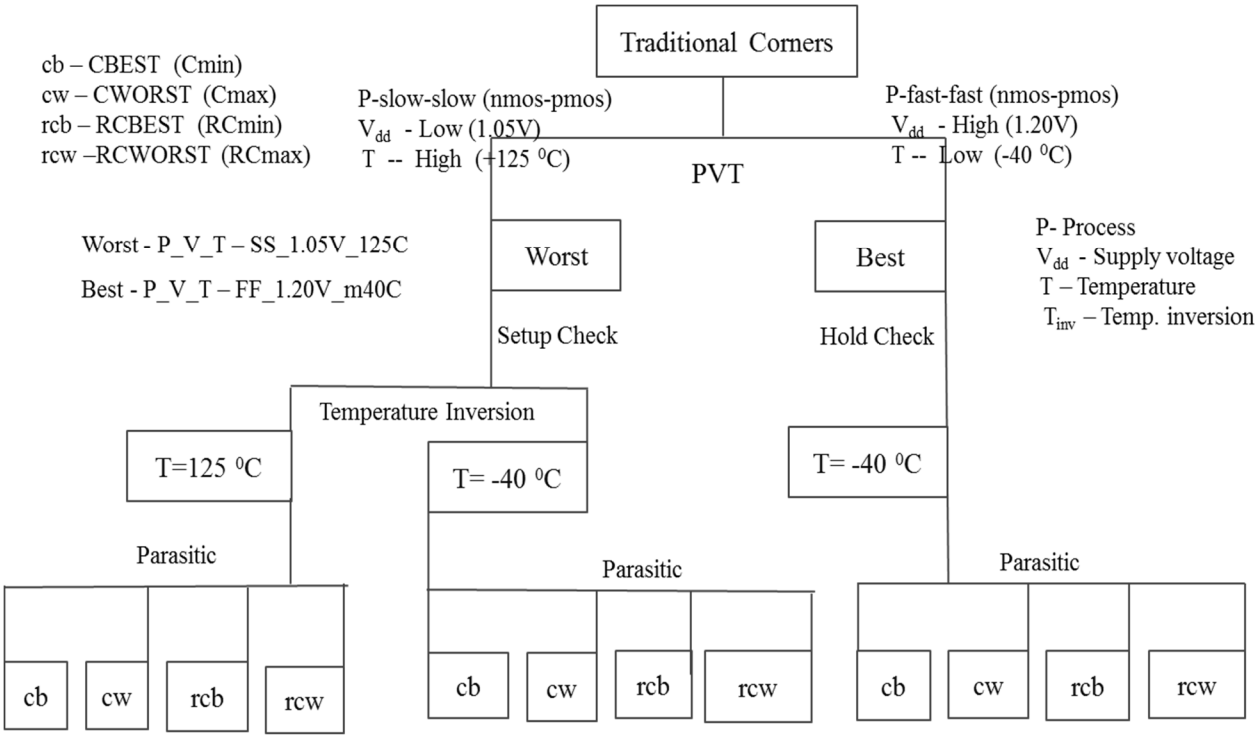


Figure 3. 5: Corners hierarchy and its division for setup and hold checks

Thus by analysing different corners during STA for setup and hold timing check few critical corners have been identified according to explained methodology.

3.5 Empirical Considerations to do STA for Limited Corners

3.5.1 Process – Voltage – Temperature

Process deviation accounts for deviations in semiconductor device parameters such as impurity concentration density, gate oxide thickness and diffusion depth, resulting in deviations in performance.

Threshold Factor Governing Current:

Higher the oxide thickness, higher the threshold voltage and hence greater effort (voltage) requires to put transistor in on state or higher transition time In case of thin gate oxide, threshold voltage is small therefore transistor switching speed is fast and hence smaller cell delay. Moreover temperature variation also affects threshold voltage. With increase in temperature, threshold voltage decreases and vice-versa as given in equation 2.2.

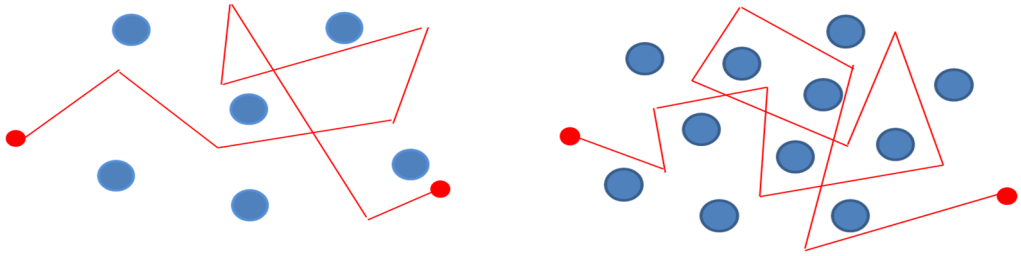
Mobility Factor Governing Current:

The ratio of saturation velocity and applied electric field is called **mobility** of the carrier, Saturation velocity depends on scattering phenomenon and hence mobility is related to mean free path of carriers and carrier density as follows:

$$\mu = \frac{v}{E} = \frac{q\tau_m}{m^*} \quad \ldots\ldots\ldots (3.1)$$

where v – Saturation velocity; E- Applied Electric Field; q – charge on carrier; τ_m – mean free path; m^* – Effective mass of particle (depends on carrier density)

As conductivity = mobility*carrier density, current depends on the mean free path and carrier density. From equation 3.1 it is clear that as mean free path decreases mobility also decreases. Mean free path depends on concentration of carrier density in the region. Higher the carrier density less will be the mean free path and hence mobility. Lower mean free path increases the chance of collision of electrons or holes as shown in Figure 3.6 and so mobility decreases.



(a) (b)

Figure 3. 6: (a) Electron Movement in low carrier density and (b) High carrier density

Impurity density distribution and carrier density decides mean free path and hence mobility of electrons and holes in the transistors. Mobility dependency on doping density shown in Figure 3.7.

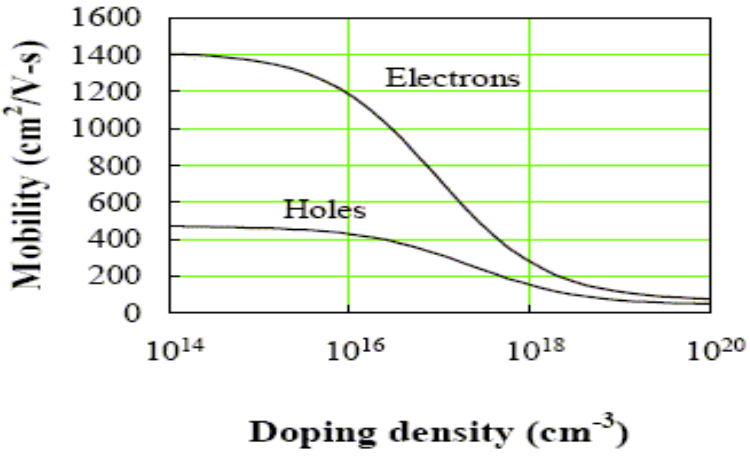


Figure 3. 7: Electron and Hole Mobility verses doping density for silicon

From Figure 3.7 it is seen that for low doping concentrations, the mobility is almost constant and is primarily limited by lattice scattering which dominates in low doping level and high temperature region. At higher doping concentrations, mobility decreases due to ionized impurity scattering as ionized doping atoms increases significantly

The dependence of mean free path and hence mobility on temperature depends on the scattering mechanism of carriers, which can be of two types: (i) Impurity scattering (ii) Lattice scattering.

Impurity Scattering:

Impurity scattering occurs due to electrostatic force between carriers and ionized impurity atoms which depends upon number of impurity atoms and time of interaction. Higher the ionized impurity concentration, greater higher the impurity scattering and hence lower mobility. The interaction time between ionized impurity and free carriers depends on relative velocity and impurity density which in turn depends on thermal velocity of carrier. So, as the temperature increases thermal velocity increases and interaction time decreases and hence mobility increases with temperature in high impurity density region.

Lattice Scattering:

As the temperature increases lattice vibration increases, vibrating atoms (called phonons) create pressure waves in the crystal lattice which reduces the mean free path among carriers and lattice atoms due to collision; therefore mobility decreases. So mobility decreases at high temperature as shown in Figure 3.8.

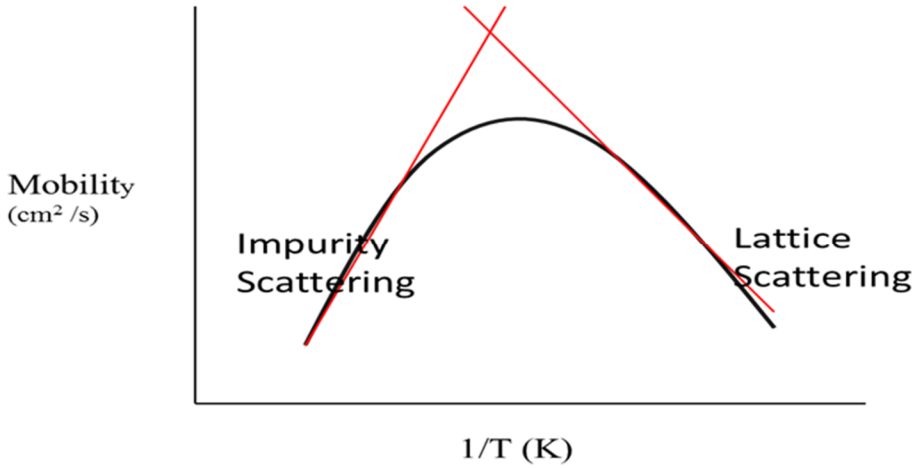


Figure 3. 8: Mobility verses Inverse of Temperature on log scale

Impurity scattering occurs in lower temperature region (nearly 200 K and below) and while lattice scattering dominant at higher temperature (nearly 400 K and above)[27]

3.5.2 Mobility and Threshold Voltage

Therefore mobility and threshold voltage are two important governing factor of drain current in transistor (CMOS technology). Mobility (μ) and threshold voltage (V_{th}) are temperature dependent. Drain current increases with increase in mobility and decrease in threshold voltage as given in equations 2.1 and 2.2. Higher the current greater will be driving capability and hence smaller delay occurs. Higher drive current corresponds to best timing condition in which delay is minimum, while lower drive current corresponds to worst timing condition in which delay is maximum. Drain current has two temperature-dependence regions: (i) Normal Dependence (ND) region (High V_{DD}) and (ii) Reverse Dependence (RD) region (≤ 45nm).

In ND region, drain current decreases and hence delay increases with ***increase*** in temperature. While both mobility and threshold voltage decreases in ND region, mobility dominates over threshold voltage. Therefore in ND region drain current decreases as mobility decreases.

In RD region, drain current decreases and hence delay increases with ***decrease*** in temperature. While both mobility and threshold voltage increase in RD region, threshold voltage (V_{th}) dominates over mobility in low voltage region (≤ 45nm). Therefore current is governed by threshold voltage and drain current decreases as temperature decreases since V_{th} increases and hence overdrive (V_{ov}) decreases (as V_{DD} is low) from equation 2.5.

Table 3. 6: Extreme PVT conditions which define worst-case and best-case delays

Process -- FF and SS (for nmos-pmos)	Fast	t_{ox}– Thin , L_{eff}– small, V_{th}– Low
	Slow	t_{ox}– Thick , L_{eff}– large, V_{th}– High
Supply Voltage		Minimum =1.05V, Maximum = 1.20V
Operating Temperature		Minimum = -40 ^{0}C, Maximum = 125 ^{0}C
Threshold Voltage		LVT=0.4 – 0.45V;RVT=0.5-0.55V; HVT= 0.7V

From all the above discussion we conclude that minimum delay occurs in best PVT condition and maximum delay occurs in worst PVT condition as presented in Table 3.6. From STA concept described in chapter 2 it is clear that for correct timing one needs to perform hold timing and setup timing that corresponds to minimum delay path and maximum delay path respectively. Therefore, by exploiting these correlation described in above paragraph an idea has been proposed in which hold timing check is done only for best PVT environment and for setup timing check analysis, to worst PVT condition. Figure 3.6 shows classification of corners division on the basis of worst and best PVT and parasitics for setup and hold timing check respectively.

3.6 Slack Trend

Slack variation for setup and hold check follow the trend as shown in Figure 3.9 (a) and 3.9 (b) respectively as discussed in proposed idea. Slack variation shown in Figure 3.9 corresponds to parasitic corners defined in worst PVT and best PVT environment for setup and hold check respectively.

Here cb- Cbest, cw- Cworst, rcb- RCbest, rcw- RCworst, AT – arrival time and RT – required time. t_{su}- setup time, t_h – hold time , SS – setup slack and HS—hold slack. Therefore SS_{cb} – setup slack for Cworst or Cmax and HS_{cb} – hold slack for Cbest or Cmin.

In setup check waveform SS_{cw} is the minimum slack value while in hold check waveform HS_{cb} is the lowest slack value. Thus these two corners are critical corners in setup and hold check respectively, in the sense that, if timing is met (no negative slack) in these corners then all others corners must satisfy timing within ±5 %.

Thus we can limit our timing analysis with STA only to such critical corners for pre-final or early TA-signoff closure at VDSM (Very Deep Submicron) nodes.

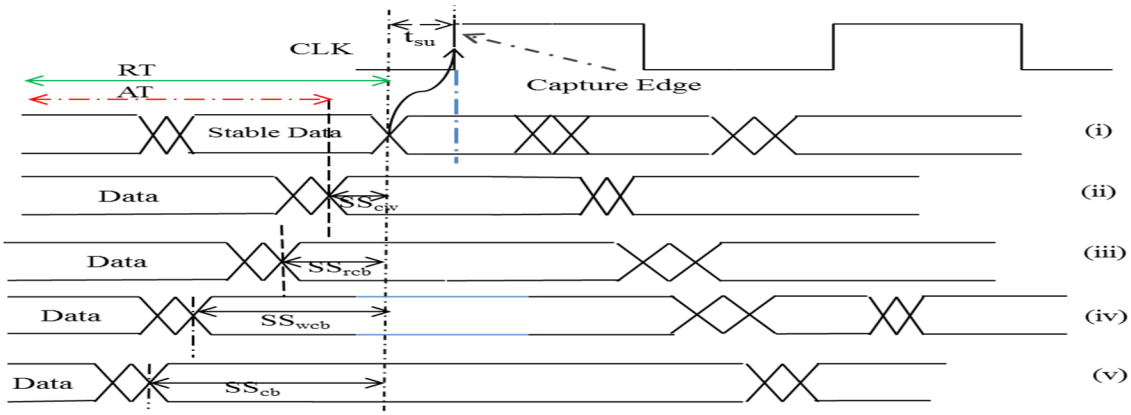


Figure : Waveform shows slack value for a path in different parasitic corners for worst PVT condition corners

(a)

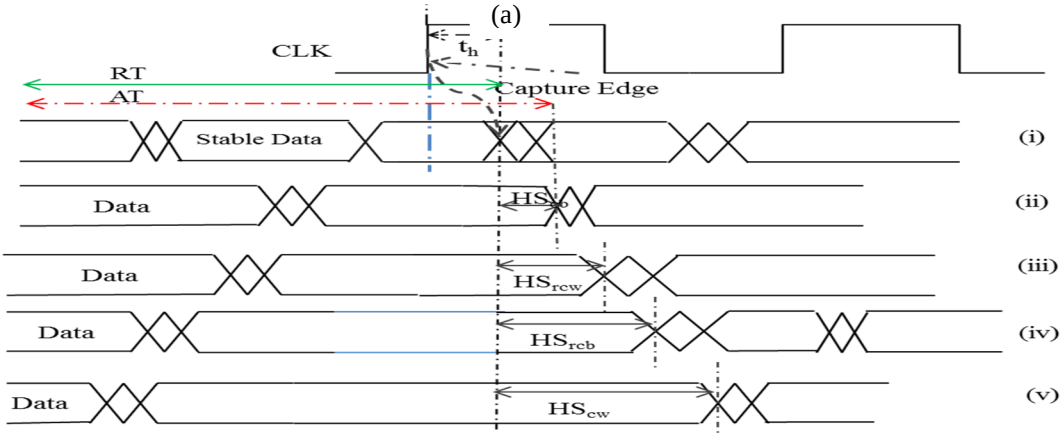


Figure : Waveform shows slack value for a path in different parasitic corners for best PVT condition corners

(b)

Figure 3. 9: Slack values and their trends in various parasitic corners

Chapter 4: Test Results and Discussion

4.1 Introduction

In this chapter the test procedure adopted and the test results obtained have been presented. Many designs have been considered for timing analysis with different frequencies. Each design consists of many partitions that may be with or without memory congestion. Frequencies are chosen in the range of 50 MHz to 550 MHz, while timing paths are chosen randomly. Slack measurement and slack comparison are carried out for setup and hold checks in various parasitic corners in each PVT condition.

4.2 Experimental Setup and Resources Used for STA

4.2.1 Steps followed to report timing paths in each corners

Step 1: Input to the STA tool. Input files required to STA are shown in Figure 3.10.

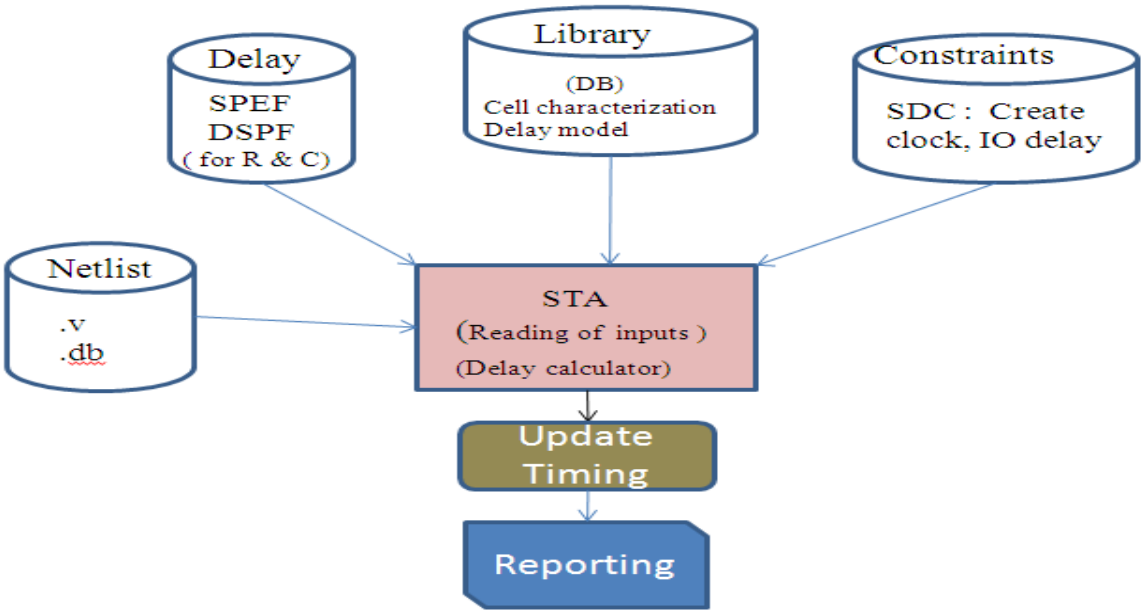


Figure 4. 1: Input to the static timing analysis tool post layout

Inputs to the STA are Net-list (or .V file – that gives information of design), Standard Delay Exchange Format (SPEF) (SPEF – gives parasitics information), .lib file (it contains information of cells, IO, macros, PVT and other cell related information.) and Synopsys design constraints (SDC – it provides information of all the constraints used in the design including clock definition).

Step 2: Distribution of analysis views

Distributed Multi-Mode-Muti-Corners technique is applied to carry out STA for multiple analysis views in parallel. Each analysis view takes individual host machine as shown in Figure 4.2. Each analysis view is distributed with threaded analysis on the remote server, enabling trade-off between time and machine license. DMMMC is used for big design (Number of instances > 3 million)

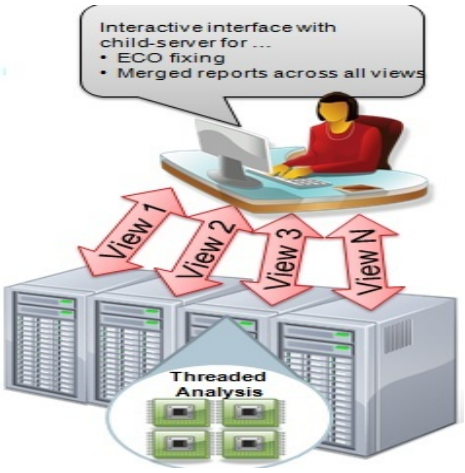


Figure 4. 2: Distributed MMMC Technique [28]

Step 3: Timing paths reported randomly for each design using industry standard tool encounter timing system (ets - Cadence). Each design consist of many partitions and each partition associated with many frequencies.

Step 4: This step concentrates on analysis of timing paths reported in step 3. Slack measurement of each timing path is measured and analysed for their variations across the corners according to the proposed method.

4.2.2 Steps followed to analyse the reports

Case 1:

In first case setup and hold timing checks are performed for all the corners at chosen frequency in each PVT. Slacks of timing paths are measured and critical corners are determined by comparing the measured slack for parasitic corners in each PVT.

Case 2:

Critical corners obtained in first case during setup timing analysis are compared for their slacks. Similarly critical corners for hold timing check are compared.

Case 3:

Both worst case critical corners are compared for their slacks.

These three steps are followed for each design, consisting of many partitions which may be memory congestion based partition or without memory congestion partition. Frequency is the basis of deciding different partitions, as different frequencies are linked with different partitions. Frequencies chosen during experiment depending on partitions in the design:

Design: Partition with memory congestion

- Clock1 – 550 MHz
- Clock2 – 450 MHz
- Clock3 – 250 MHz
- Clock4 – 50 MHz

Design: Partition without memory congestion

- Clock1 – 363 MHz
- Clock2 – 148 MHz
- Clock3 – 50 MHz

Table 4. 1: Summary of variables used to represent corners

ff40_1.20V_m40C				ss40_1.05V_m40C				ss40_1.05V_125C			
cbest	cworst	rcbest	rcworst	cbest	cworst	rcbest	rcworst	cbest	cworst	rcbest	rcworst
C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12

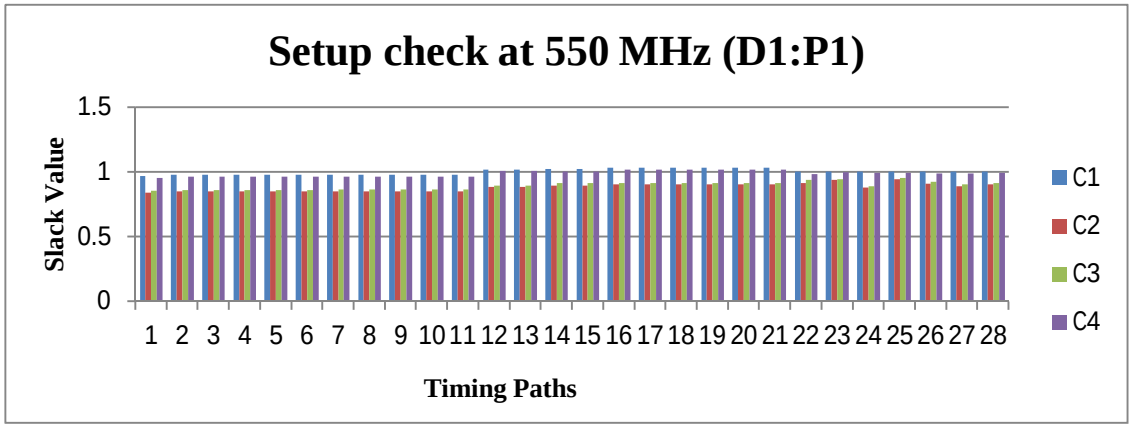
4.3 Slack variation across parasitic corners for given PVT

4.3.1: Setup timing check for partition with memory congestion

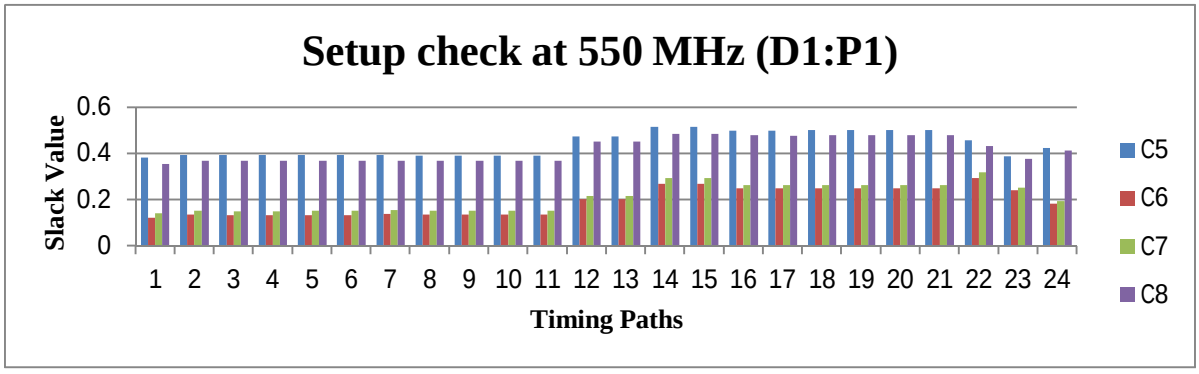
Case 1: Design 1: Partition 1 (D1:P1); Frequency 550 MHz

In this case we check setup and hold timings in each PVT corner and identify critical parasitic corners by measuring and comparing their slack values depicted in the following chart, where the vertical axis represents slack value and horizontal axis, number of timing paths chosen randomly.

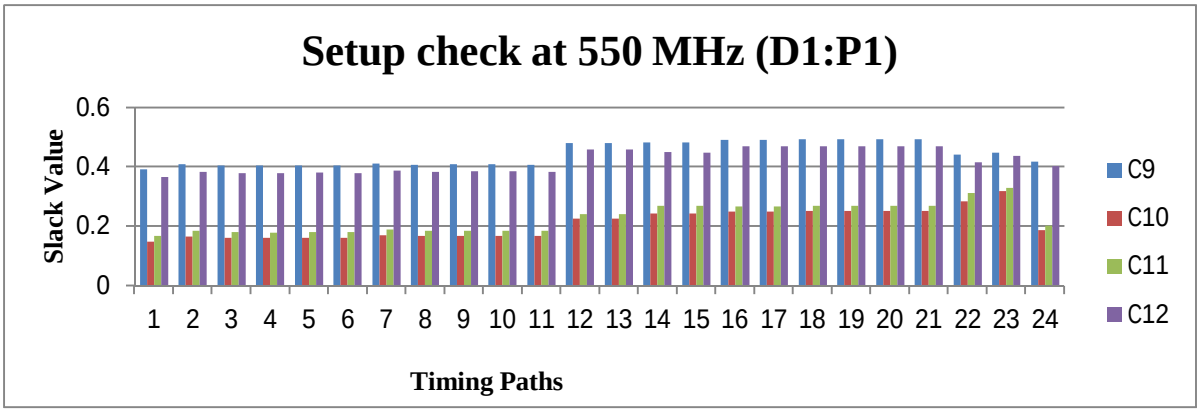
(a) Best PVT (FF40_1.20V_m40C)



(b) Worst PVT (SS40_1.05V_m40C)



(c) **Worst PVT (SS40_1.05V_125C)**

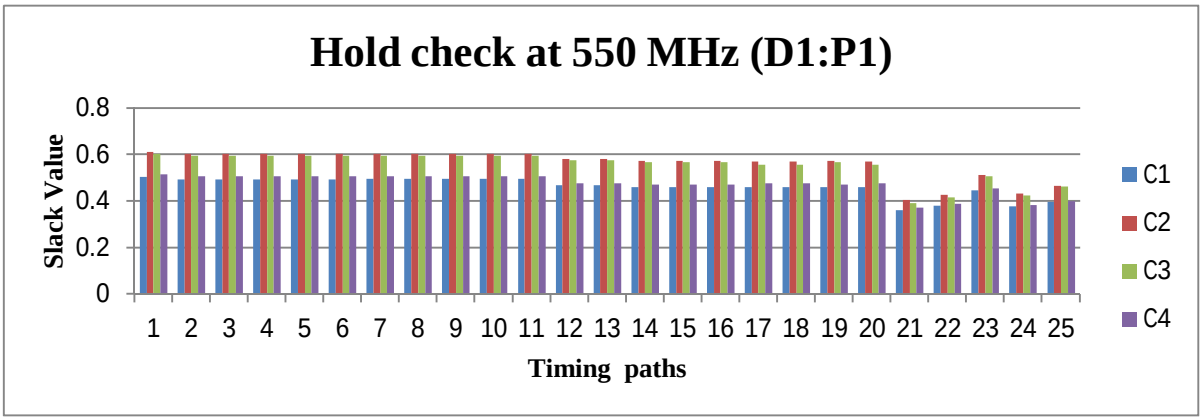


Discussion:

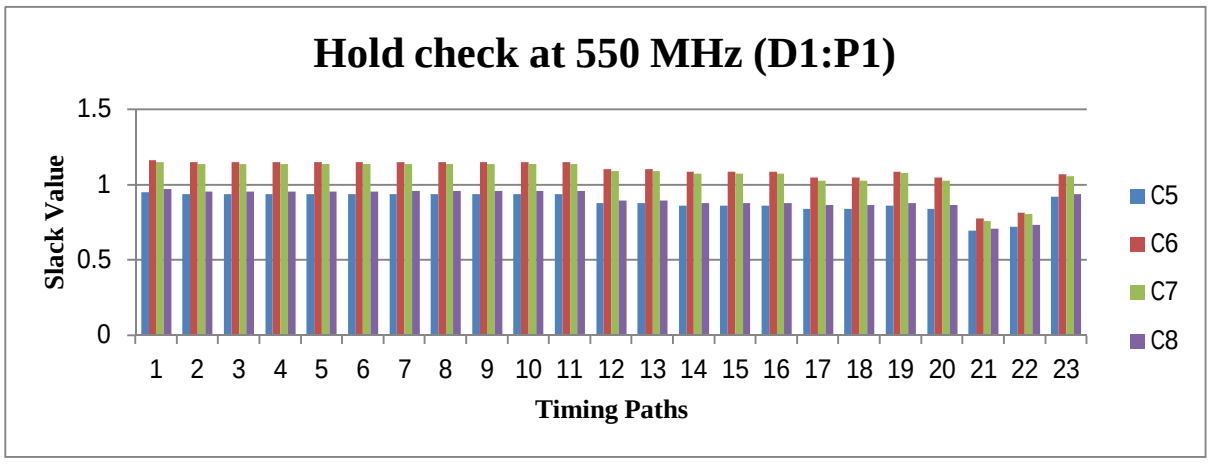
Setup slack of parasitic corners has been measured for randomly taken timing paths. Each PVT corner possesses four parasitic corners. Comparison carried out among C1-C4, C5-C8, and C9-C12 through their slack value for ***setup*** check shows that ***C2, C6*** and ***C10*** (Cworst) present minimum slack values among the values corresponding to the three PVT corners taken. So these parasitic corners are critical corners among other parasitic corners in respective PVT conditions shown in Table 4.1. These corners correspond exactly to the theoretically expected critical delay paths: maximum delay path for setup timing check, and minimum delay path for hold timing check.

4.3.2: Hold timing check for partition with memory congestion

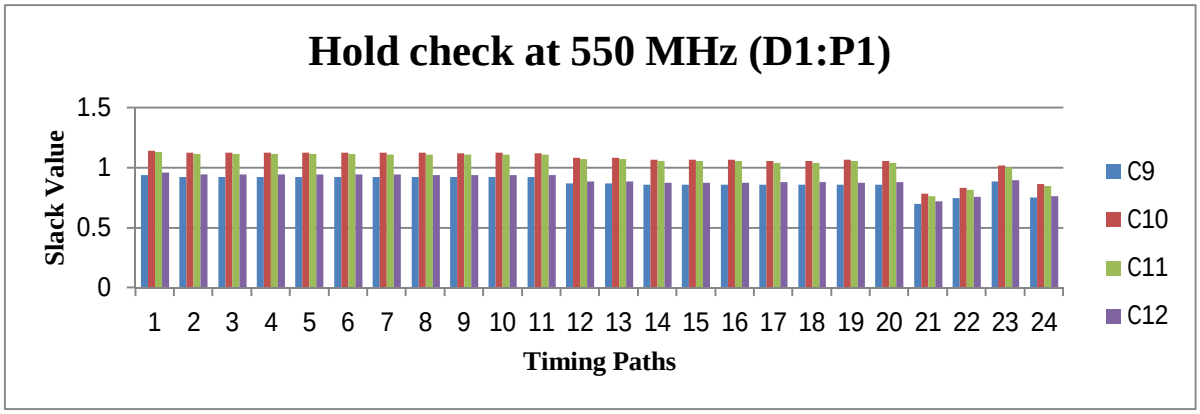
(a) **Best PVT (FF40_1.20V_m40C)**



(b) **Worst PVT (SS40_1.05V_m40C)**



(c) **Worst PVT (SS40_1.05V_125C)**



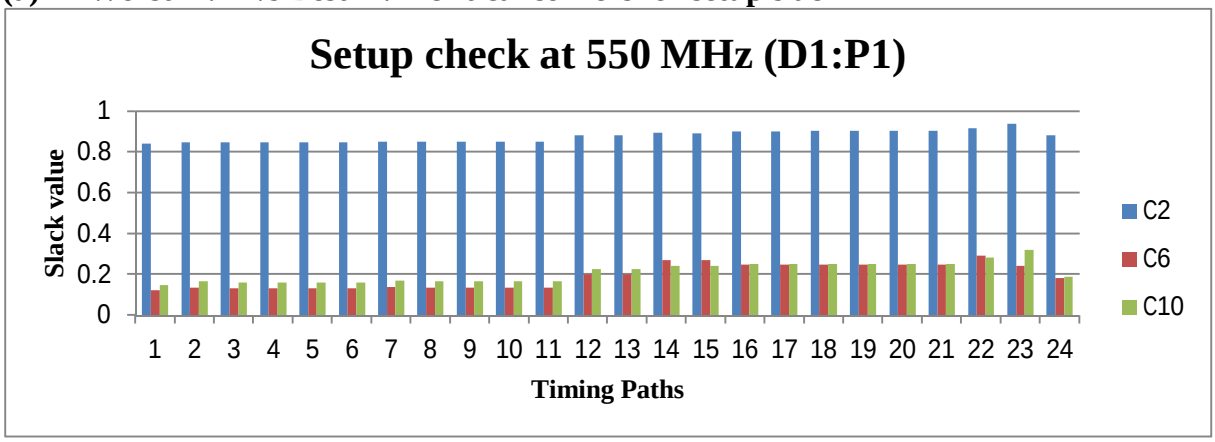
Discussion:

Hold timing check has been performed by measuring slack value of randomly chosen timing paths. Each PVT corner possesses four parasitic corners. Comparison carried out among C1-C4, C5-C8, and C9-C12 through their slack value for ***hold*** check shows that ***C1, C5*** and ***C9*** (Cbest) present minimum slack values among the values corresponding to the three PVT corners taken. So these parasitic corners are critical corners among other parasitic corners in respective PVT conditions shown in Table 4.1. These corners correspond exactly to the theoretically expected critical delay paths: maximum delay path for setup timing check, and minimum delay path for hold timing check.

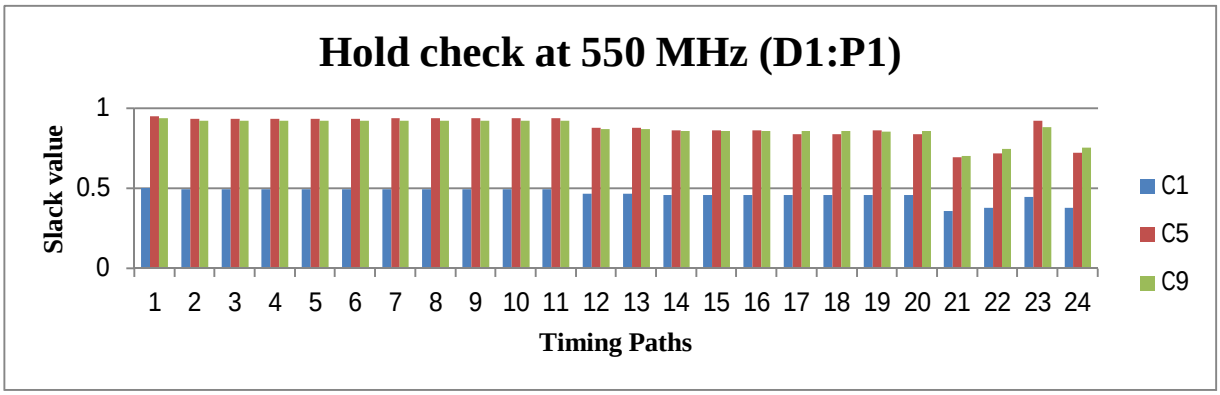
Case 2:

In this case we check setup and hold timings in the critical parasitic corners identified in case 1 for each PVT corner, and identify critical parasitic corners by measuring and comparing their slack values depicted in the following chart, where the vertical axis represents slack value and horizontal axis, number of timing paths chosen randomly.

(a) Worst PVT vs Best PVT critical corners for setup slack



(b) Worst PVT vs Best PVT critical corners for hold slack



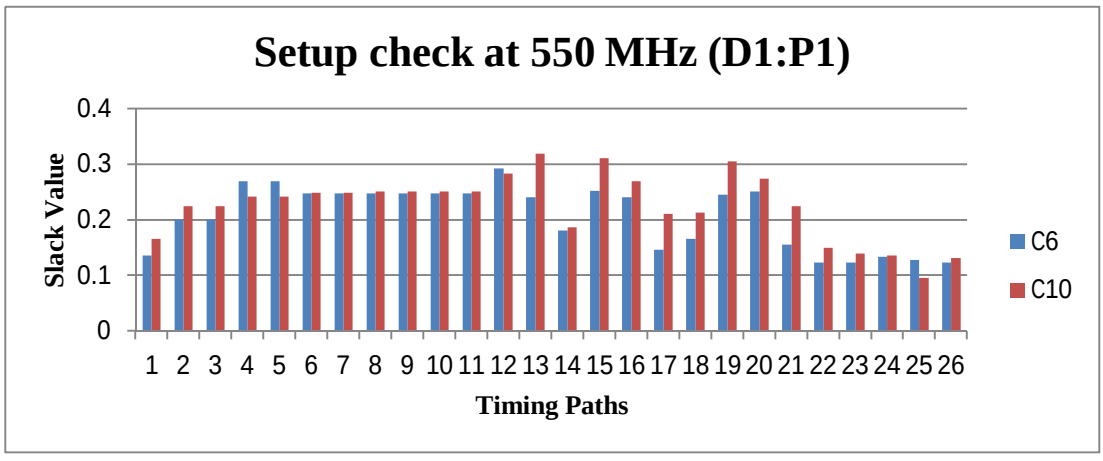
Discussion:

Slack comparisons for setup and hold slacks have been carried out at the critical corners determined in the first case. It is found that for ***setup*** check corner ***C2***, corresponding to Cworst (or Cmax) in best PVT condition, has the maximum positive slack, implying that C2 is safest corner for setup check requirement. The remaining corners C6 and C10 are therefore to be considered for identifying the final critical corners for setup check. Similarly ***C1*** gives the minimum slack value for ***hold*** timing check, and hence is the critical corner. These corners correspond exactly to the theoretically expected critical delay paths: maximum delay path for setup timing check, and minimum delay path for hold timing check

Case 3:

In this case we check setup timings in the critical parasitic corners C6 and C10 identified in case 2, and identify critical parasitic corners by measuring and comparing their slack values as depicted in the following chart, where the vertical axis represents slack value and horizontal axis, number of timing paths chosen randomly

Worst PVT critical corners for setup slack



Discussion:

Comparison of the setup slack values in worst PVT condition at the two critical corners C6 and C10 shows that none of them can be said to be clearly dominant. This analysis confirms that both these critical corners in worst PVT need to considered for timing analysis.

The numerical data obtained for the setup and hold slack values are given in Table 4.2 and 4.3.

Table 4. 2: Setup slack comparison across the corners

ff40_1.20V_m40C					ss40_1.05V_m40C					ss40_1.05V_125C				
cbest	cworst	rcbest	rcworst		cbest	cworst	rcbest	rcworst		cbest	cworst	rcbest	rcworst	
C1	C2	C3	C4	C3-C2	C5	C6	C7	C8	C7-C6	C9	C10	C11	C12	C11-C10
0.9713	0.8409	0.854	0.9554	0.0131	0.3822	0.1218	0.1402	0.3566	0.0184	0.3921	0.1475	0.1669	0.3665	0.0194
0.9796	0.849	0.8616	0.9641	0.0126	0.3938	0.134	0.1514	0.3686	0.0174	0.4095	0.1656	0.1842	0.3845	0.0186
0.9796	0.8491	0.8615	0.9642	0.0124	0.3932	0.1331	0.1506	0.3681	0.0175	0.4054	0.1609	0.1796	0.3802	0.0187
0.9796	0.8491	0.8615	0.9642	0.0124	0.3932	0.1331	0.1506	0.3681	0.0175	0.4054	0.1609	0.1795	0.3802	0.0186
0.9797	0.8492	0.8616	0.9643	0.0124	0.3933	0.1333	0.1507	0.3682	0.0174	0.4054	0.161	0.1796	0.3804	0.0186
0.9797	0.8491	0.8616	0.9642	0.0125	0.3933	0.1332	0.1507	0.3681	0.0175	0.4053	0.1609	0.1796	0.3803	0.0187
0.9799	0.8509	0.863	0.9655	0.0121	0.394	0.1388	0.1552	0.3704	0.0164	0.4113	0.1707	0.1887	0.3874	0.018
0.98	0.8509	0.8631	0.9655	0.0122	0.3924	0.1359	0.1524	0.3684	0.0165	0.4084	0.1667	0.1847	0.3844	0.018
0.9801	0.8514	0.8632	0.9659	0.0118	0.3926	0.1363	0.1525	0.3688	0.0162	0.4087	0.1673	0.1851	0.385	0.0178
0.9801	0.8512	0.8631	0.9658	0.0119	0.3925	0.1361	0.1525	0.3687	0.0164	0.4085	0.1671	0.185	0.3849	0.0179
0.9805	0.8516	0.8634	0.9662	0.0118	0.3928	0.1365	0.1528	0.3691	0.0163	0.4071	0.1666	0.1842	0.3833	0.0176
1.0206	0.8835	0.8939	1.0074	0.0104	0.4741	0.2012	0.2152	0.4523	0.014	0.4813	0.225	0.242	0.4584	0.017
1.0207	0.8836	0.894	1.0075	0.0104	0.4742	0.2013	0.2152	0.4524	0.0139	0.4816	0.2252	0.2421	0.4587	0.0169

Table 4. 3: Hold slack comparison across the corners

ff40_1.20V_m40C					ss40_1.05V_m40C					ss40_1.05V_125C				
cbest	cworst	rcbest	rcworst		cbest	cworst	rcbest	rcworst		cbest	cworst	rcbest	rcworst	
C1	C2	C3	C4	C4-C1	C5	C6	C7	C8	C8-C5	C9	C10	C11	C12	C12-C9
0.5037	0.613	0.6041	0.5161	0.0124	0.9506	1.1658	1.1507	0.9715	0.0209	0.9416	1.1452	1.1313	0.9616	0.02
0.4948	0.6041	0.5959	0.5065	0.0117	0.9377	1.1531	1.1378	0.9584	0.0207	0.9261	1.1296	1.1163	0.9454	0.0193
0.4949	0.6041	0.596	0.5064	0.0115	0.9377	1.153	1.1377	0.9583	0.0206	0.9261	1.1296	1.1163	0.9454	0.0193
0.4949	0.6041	0.596	0.5064	0.0115	0.9377	1.153	1.1377	0.9583	0.0206	0.9261	1.1296	1.1163	0.9454	0.0193
0.4948	0.604	0.5959	0.5063	0.0115	0.9376	1.1528	1.1376	0.9582	0.0206	0.926	1.1294	1.1163	0.9451	0.0191
0.4948	0.604	0.5959	0.5064	0.0116	0.9376	1.153	1.1376	0.9583	0.0207	0.926	1.1296	1.1163	0.9453	0.0193
0.4972	0.6038	0.5965	0.5078	0.0106	0.9414	1.1527	1.1387	0.9605	0.0191	0.9251	1.1253	1.1129	0.9432	0.0181
0.4971	0.6038	0.5963	0.5078	0.0107	0.9413	1.1527	1.1387	0.9605	0.0192	0.925	1.1252	1.113	0.9431	0.0181
0.4967	0.6033	0.5962	0.5072	0.0105	0.9411	1.1521	1.1385	0.96	0.0189	0.9246	1.1246	1.1126	0.9424	0.0178
0.4969	0.6035	0.5963	0.5075	0.0106	0.9412	1.1524	1.1385	0.9601	0.0189	0.9248	1.1248	1.1127	0.9427	0.0179
0.4966	0.6032	0.5961	0.5072	0.0106	0.9408	1.152	1.1383	0.9598	0.019	0.9243	1.1245	1.1124	0.9422	0.0179
0.4683	0.5818	0.5756	0.4779	0.0096	0.8792	1.1046	1.093	0.8964	0.0172	0.8713	1.0842	1.0723	0.8885	0.0172
0.468	0.5814	0.5753	0.4775	0.0095	0.8789	1.1042	1.0928	0.896	0.0171	0.871	1.0835	1.0719	0.8879	0.0169
0.4617	0.5748	0.5678	0.4722	0.0105	0.8629	1.0875	1.0757	0.8804	0.0175	0.8586	1.0703	1.0586	0.8771	0.0185

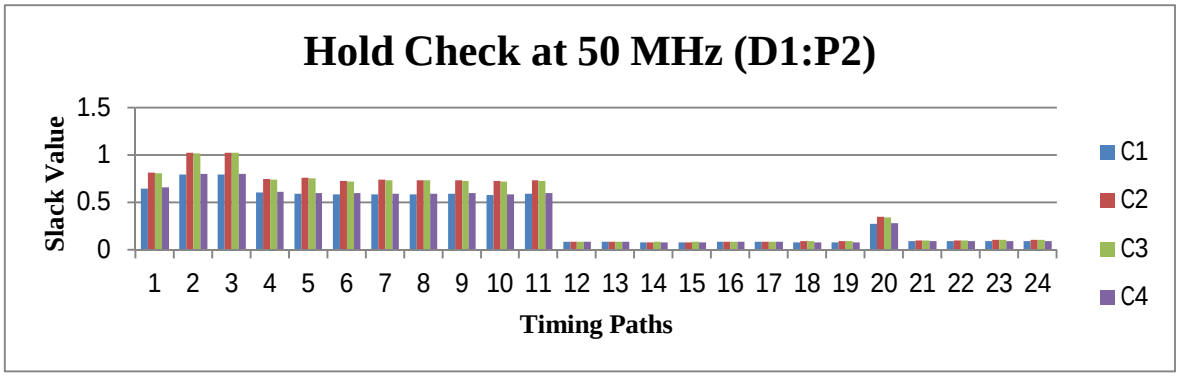
- **Row – represents timing paths;**
- **Column – represents parasitic corners and their slack values for given PVT**
- **Slack values comparison carried out using colour code scheme**
- **Yellow colour –** represents highest slack value
- **Dark Orange –** represents lowest slack value

4.3.3: Timing check for partition with memory congestion

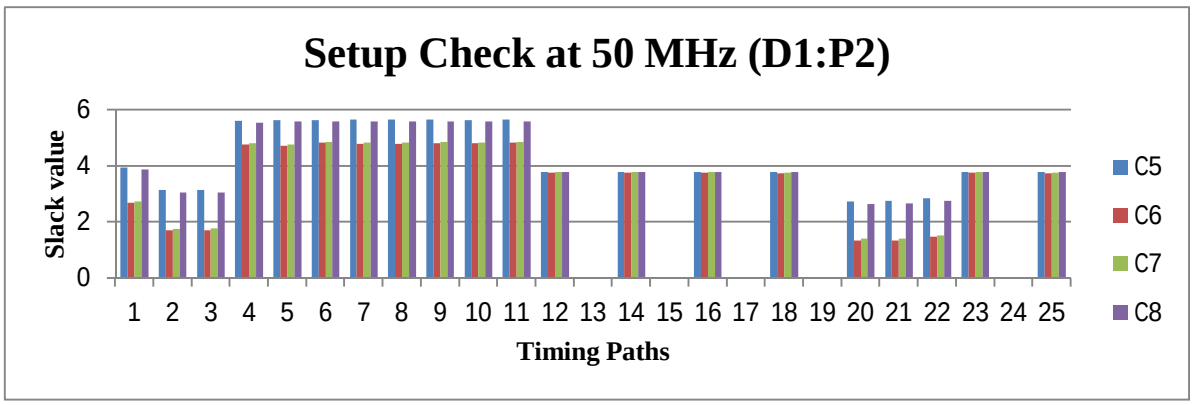
Case 1: Design 1: Partition 2 (D1:P2); Frequency 50 MHz

Comparison among parasitic corners have been done through their slack values. Best PVT is chosen for hold check and worst PVT are chosen for setup check.

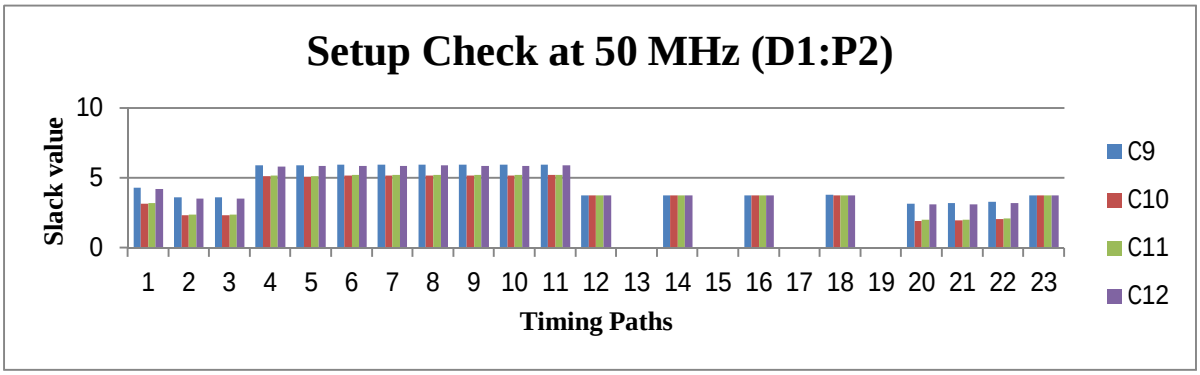
i. **Best PVT (FF40_1.20V_m40C)**



ii. **Worst PVT (SS40_1.05V_m40C)**



iii. **Worst PVT (SS40_1.05V_125C**



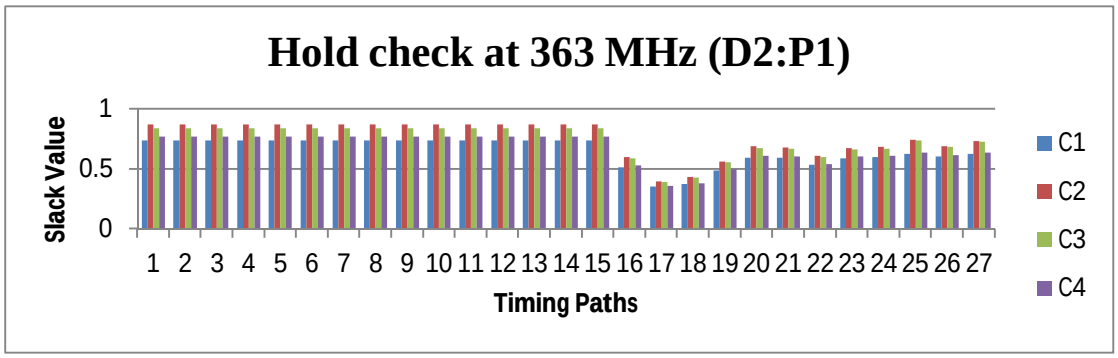
Discussion:

Setup slack and hold slack of parasitic corners have been measured for randomly taken timing paths. Each PVT corner possesses four parasitic corners. Comparison carried out among C1-C4, through their slack value for ***hold*** check and C5-C8, C9-C12 through their slack value for ***setup*** check which shows that ***C1(Cbest), C6*** and ***C10*** (Cworst) present minimum slack values among the values corresponding to the three PVT corners taken. So these parasitic corners are critical corners among other parasitic corners in respective PVT conditions shown in Table 4.1. These corners correspond exactly to the theoretically expected critical delay paths: maximum delay path for setup timing check, and minimum delay path for hold timing check

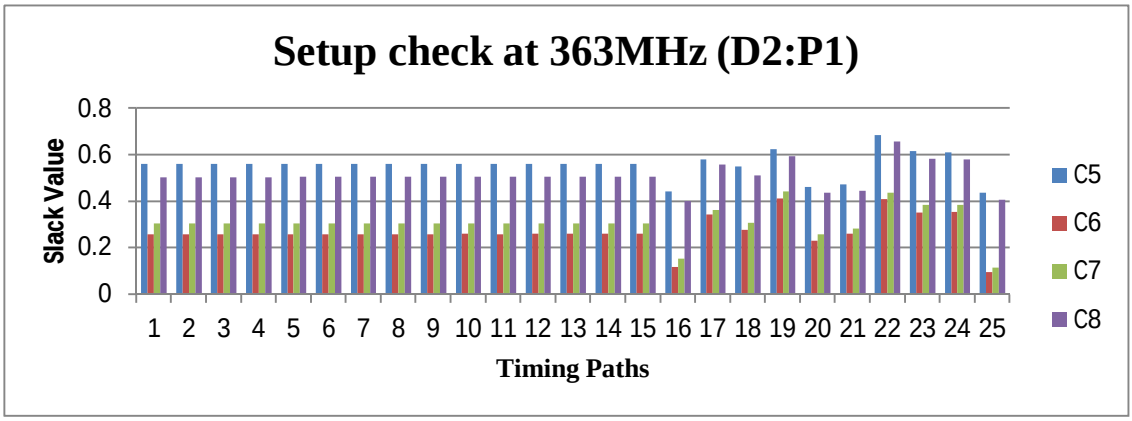
4.3.4: Timing check for partition without memory congestion

Case 1: Design 2: Partition 1 (D2:P1); Frequency 363MHz

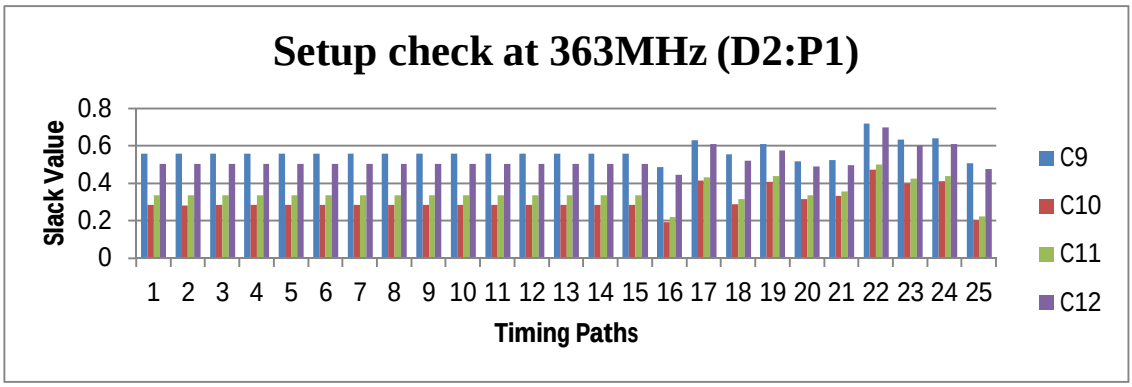
i. **Best PVT (FF40_1.20V_m40C)**



ii. **Worst PVT (SS40_1.05V_m40C)**



iii. **Worst PVT (SS40_1.05V_m40C)**



Discussion:

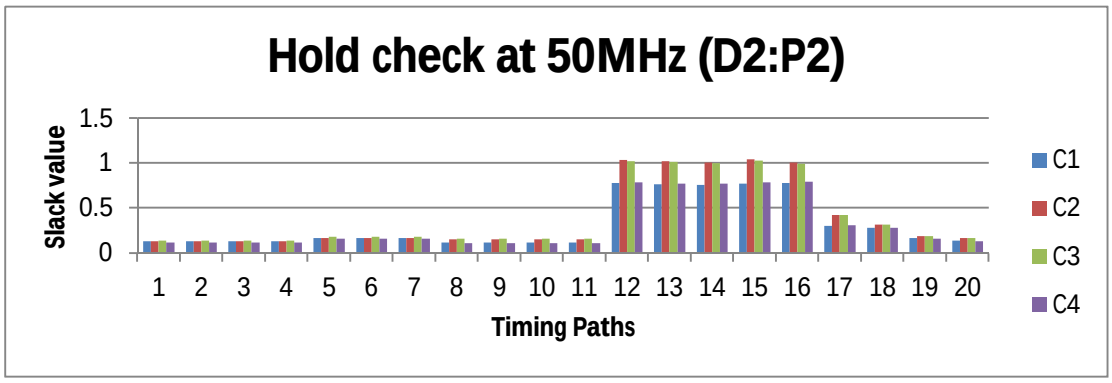
Setup slack and hold slack of parasitic corners have been measured for randomly taken timing paths. Each PVT corner possesses four parasitic corners. Comparison carried out among C1-C4, through their slack value for ***hold*** check and C5-C8, C9-C12 through their slack value for ***setup*** check which shows that ***C1(Cbest), C6*** and ***C10*** (Cworst) present minimum slack

values among the values corresponding to the three PVT corners taken. So these parasitic corners are critical corners among other parasitic corners in respective PVT conditions shown in Table 4.1. These corners correspond exactly to the theoretically expected critical delay paths: maximum delay path for setup timing check, and minimum delay path for hold timing check

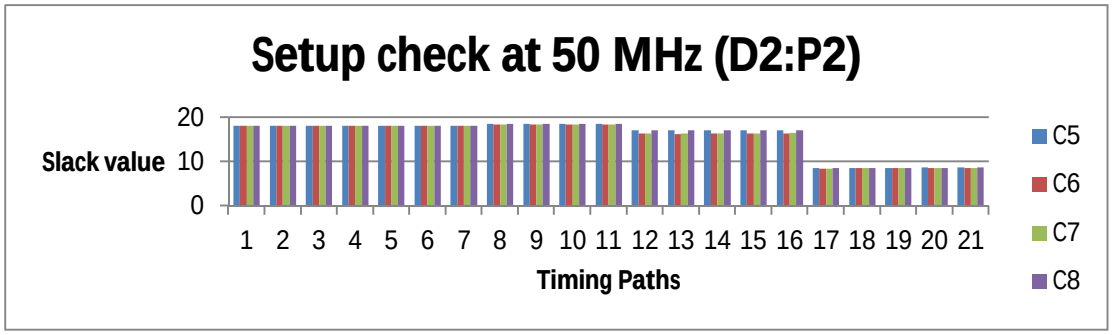
4.3.4: Timing check for partition without memory congestion

Case 1: Design 2: Partition 2 (D2:P2); Frequency 50MHz

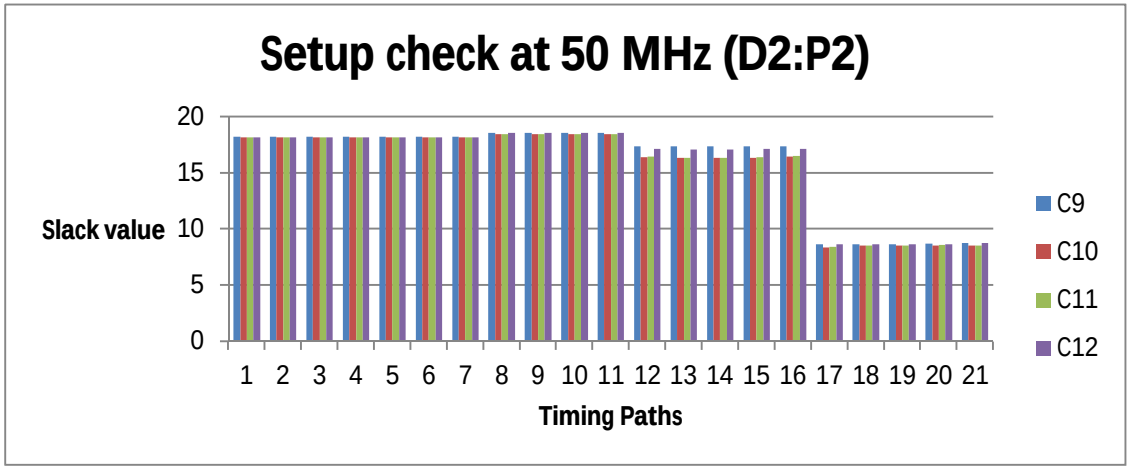
i. Best PVT (FF40_1.20V_m40C)



ii. Worst PVT (SS40_1.05V_m40C)



iii. Worst PVT (SS40_1.05V_125C)



Similar trend (as in case of D1:P1, D1:P2, D2:P1) has been observed. For ***hold*** check ***C1*** is the critical corner and for ***setup*** check ***C6, C10*** are the critical corners.

Discussion:

Similar trend (as in case of D1:P1) has been observed in different designs and partitions (with a 4% to 5% variation) D1:P2, D2:P1 and D2:P2. At higher frequencies - 100MHz and above - critical corners C1, C6 and C10 are sufficient. At lower frequencies - 50 MHz and below - slack difference is very small (in the order of 0.2 to 16.34 ps) and hence it is difficult to identify the critical corners. However, higher frequency timing paths are more critical than lower frequency timing paths. Therefore four from worst PVT and two from best PVT conditions are sufficient for setup and hold timing check respectively. By considering 4% to 6% variation we can further reduce to three critical corners, two for setup and one for hold timing check.

It is also observed that Cworst and RCbest (or Cbest and RCworst) give rise to comparable slack values. Since slack is the difference of two timing paths, so it is independent of parasitic condition.

This may be possible that the trend will be different at lower technology nodes depending upon routing and density of wire. But the number of critical corners will be same as describes in first paragraph of this section.

Chapter 5: Conclusion and Future work

It has been established that the number of critical corners get reduced to 6 (2 for hold check and 4 for setup check) out of the maximum possible 12 corners if we do not consider any slack variation. If slack of variation ± 4-5% is considered, the number of critical corners gets further reduced to 3 (1 hold and 2 setup).

Let us consider a scenario where the design has 3.5 million instances with 25 modes. In the customary approach using 12 corners, 10 machines, each capable of completing STA-analysis for 3 analysis views in a day, would require 12*25/10*3 = 10 days. By reducing the number of analysis corners to 3, the same number of machines will need 2.5 days to complete the same job. Memory usage also goes down by the same factor of 4.

If number of machines reduced to 5, the time taken for the job will be 5 days and thus there is a scope for trade-off between machines run time and machine cost.

Thus by identifying these critical corners, the time required for early TA- signoff closure can be drastically reduced, thereby enhancing the productivity of an organization significantly without compromising quality.

Future work leading to further STA improvement could involve restricting the complete timing analysis covering all possible paths for one mode only and performing timing analysis in other modes for limited timing paths and instances significant for that mode only.

References

[1] Moore's Law: past, present, future a perspective
[2] Deep-Submicron Micro Processor Design Issue
[3] www.eejournal.com/archives/articles/20121206-cadence
[4] “Complete timing sign-off nano meter era” cadence
[5] Process Variation Challenges and Solutions Approaches
[6] “Static Timing Analysis for Manometer Designs A Practical Approach” J. Bhasker, Rakesh Chadha
[7] http://www.vlsi-expert.com/2011/02/process-variation-effects-on-design.html
[8] “Analysis of the Impact of Process Variations on Clock Skew”
Stefano Zanella, Student Member, IEEE, Alessandra Nardi, Student Member, IEEE, Andrea Neviani, Member, IEEE, Michele Quarantelli, Member, IEEE, Sharad Saxena, Senior Member, IEEE, and Carlo Guardiani, Member, IEEE
[9] ieee transactions on computer-aided design of integrated circuits and systems, vol. 25, no. 9, september 2006
[10] “Incorporating Effects of Process, Voltage, and Temperature Variation in BTI Model for Circuit Design” Shreyas Kumar Krishnappa, Harwinder Singh, and Hamid Mahmoodi School of Engineering, San Francisco State University San Francisco, CA, USA{shreyas, harsingh, mahmoodi}@sfsu.edu
[11] http://www.vlsi-expert.com/2012/02/parasitic-interconnect-corner-rc-corner.html
[12] Handling Inverted Temperature Dependencein Static Timing Analysis” ALI DASDAN and IVAN HOM Synopsys, Inc.
[13] D. Wolpert and P. Ampadu, Managing Temperature Effects in Nanoscale Adaptive Systems, DOI 10.1007/978-1-4614-0748-5_2,# Springer Science+Business Media, LLC 2012
[14] “Scaling of Interconnections” EE 311 Notes/Prof Saraswat/Stanford university
[15] http://siliconsaint.blogspot.in/2012/07/temperature-inversion-in-deep-sub.html
[16] Exploiting Setup–Hold-Time Interdependence in Static Timing Analysis Emre Salman, Student Member, IEEE, Ali Dasdan, Member, IEEE, FerozeTaraporevala, Member, IEEE, Kayhan Küçükçakar, Member, IEEE, and Eby G. Friedman, Fellow, IEEE
[17] W. Roethig, “Library characterization and modeling for 130 nm and 90 nm SOC design,” Proc. IEEE Int. SOC Conf., Sep. 2003, pp. 383–386.
[18] D. Patel, “CHARMS: Characterization and modeling system for accurate delay prediction of ASIC designs,” Proc. IEEE Custom Integr. Circuits Conf., May 1990, pp. 9.5.1–9.5.6.
[19] R. W. Phelps, “Advanced library characterization for high-performance ASIC,” Proc. IEEE Int. ASIC Conf., Sep. 1991, pp. 15-3.1–15-3.4.
[20] M. A. Cirit, “Characterizing a VLSI standard cell library,” in Proc. IEEE Custom Integr. Circuits Conf., May 1991, pp. 25-7.1–25-7.4.
[21] V. Stojanovic and V. G. Oklobdzija, “Comparative analysis of masterslavelatches and flip-flops for high-performance and low-power systems,”IEEE J. Solid-State Circuits, vol. 34, no. 4, pp. 536–548, Apr. 1999.
[22] A. M. Jain and D. Blaauw, “Modeling flip flop delay dependencies intiming analysis,” presented at the ACM/IEEE Timing Issues (TAU)Workshop, Austin, TX, Feb. 2004
[23] G. Rao and K. Howick, “Apparatus for optimized constraint characterization

with degraded options and associated methods," U.S. Patent6 584 598 B2, Jun. 24, 2003.
[24] E. K. Howick, "Conquering the high-frequency domain with predictable sequential models," in Proc. Electr. and Phys. Des. Conf., Jan. 2002,CD-ROM.
[25] Milliman and Halkias "Electronics Device And Circuits"
[26] Bart J. Van Zeghbroeck, Professor Department of Electrical and Computer Engineering University of Colorado at Boulder. "Principal of Semiconductor Devices"
[27] Ali Shakauri "University of California at Santa Cruz Jack Baskin School of Engineering Electrical Engineering Department". Temperature dependence of semiconductor conductivity
[28] STMicroelectronics "Strategies for Efficient timing signoff convergence at VDSM nodes" Mohit verma