Static Timing Analysis

Difference between Dynamic Timing Analysis and Static Timing Analysis

Dynamic Timing Analysis	Static Timing Analysis
Verifies functionality of the design by applying input vectors and checking for correct output vectors	Checks Static Delay requirements of the circuit without any input or output vectors, so analysis times are relatively short and STA does not check for logical correctness of the design
Quality increases with the increase of input test vectors	Clock-related information has to be fed to the design in the form of constraints and the correctness of the constraints decides the quality
Increased Test Vectors increase Simulation Time	Timing can be analyzed for the worst case and best case simultaneously and also all timing paths areconsidered
Can be used for synchronous as well as asynchronous designs	Not suitable for asynchronous designs
Also best suitable for designs having clocks crossing multiple domains	Not suitable for designs having clocks crossing multiple domains
Computational complexity involved in finding the Input Patterns/Vectors that produce maximum delay at the output	Has more pessimism and thus gives maximum delay of the design and STA and it works with timing models

- Effective methodology for verifying the timing characteristics of a design without the use of test vectors
- Static Timing Analysis can be done only for Register-Transfer-Logic (RTL) designs
- Functionality of the design must be cleared before the design is subjected to STA
- STA approach typically takes a fraction of the time it takes to run a logic simulation

STA is a method of adding the net delays and cell delays to obtain path delays. then STA tool analyzes all paths from each and every start point to each and every endpoint and compares it against the constraint(timing specification) that exists for that path

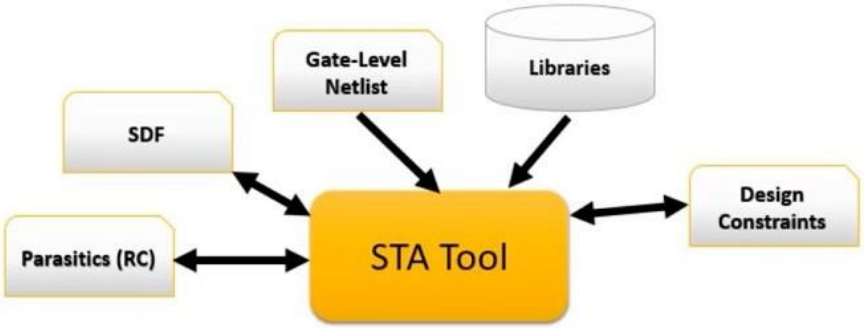
Purpose of Static Timing Analysis

- First, STA calculates the path delays for optimization tools. then based on the path delays, the optimization tool chooses cells from the timing library to create a circuit that meets your timing requirement.
- Second, STA analyzes the timing of a circuit to verify that the circuit works at the specified frequency.

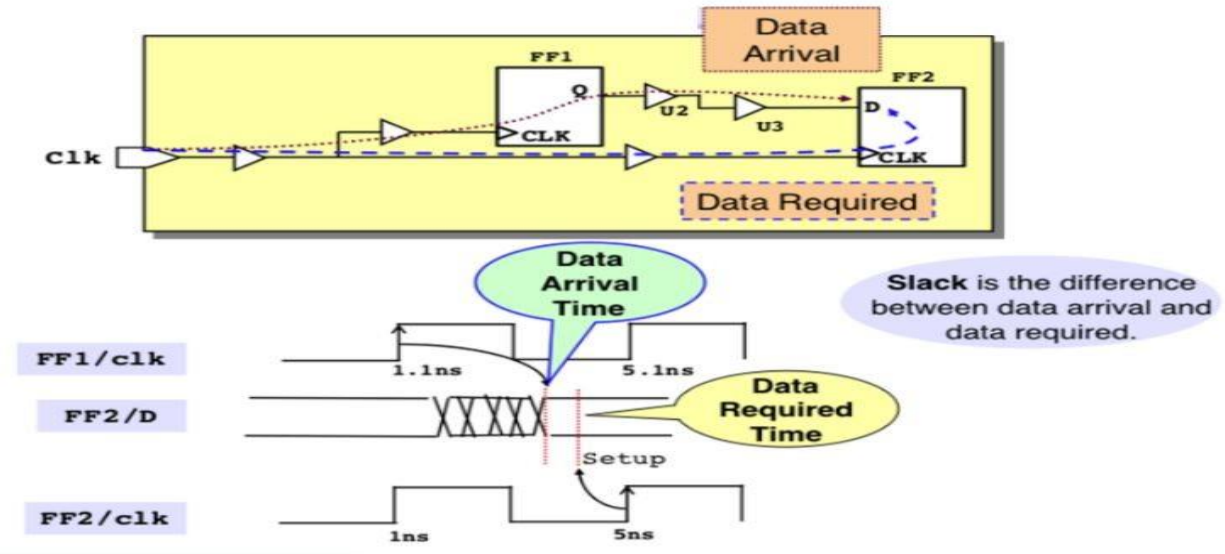
The main steps of STA

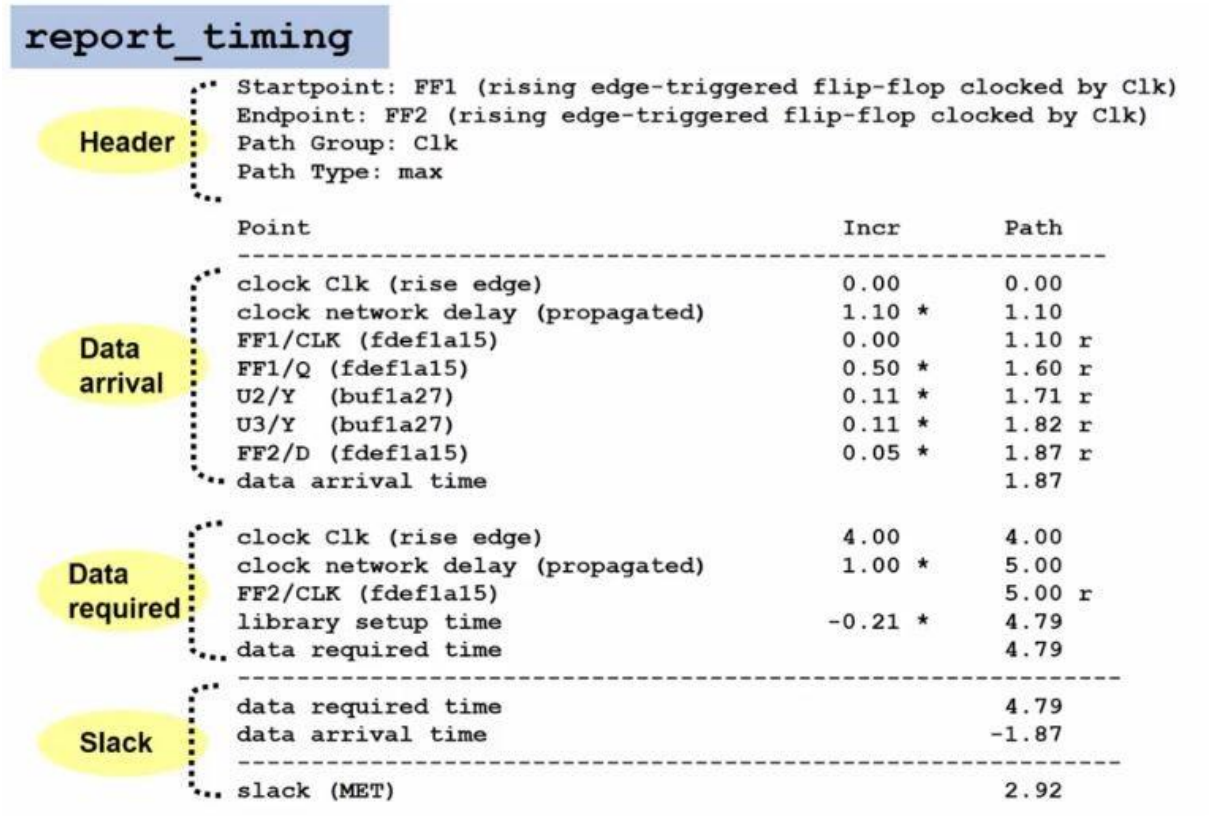
- Break the design into sets of timing paths
- Calculate the delay of each path
- Check all path delays to see if the given timing constraints are met

STA inputs and outputs



Timing Report





The above timing report is divided into 4 parts as the **Header**

- consists of a start point(FF1) and an end point(FF2)
- path group which tells for which timing path group it belongs.
- Path type: here it is max which states setup and if it was min then it is hold.

Data Arrival Section

reports the total time taken to arrive at D Pin of Flip flop 2. (ref fig just above timing report)

Data Required Section

reports the total time taken to arrive at the clock pin of FF1 minus the setup time of FF2. (ref fig just above timing report)

Slack

the timing difference between required and arrival time i.e. (RT-AT)

Typical symbols that can be seen in the PrimeTime report :

- “&” after an incremental delay number shows that the delay number is calculated with Resistor-Capacitor (RC) network back-annotation.
- “*” for Standard Delay Format (SDF) back-annotation

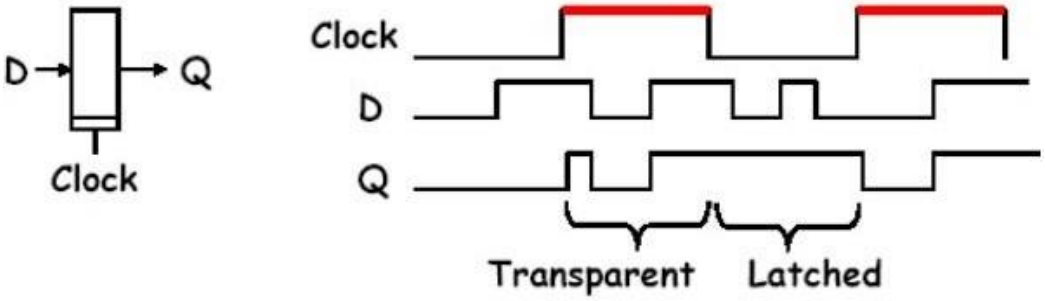
- “+” for lumped RC
- “H” for hybrid annotation
- “r” in the path column for the rising edge of the signal
- “f” in the path column for the falling edge of the signal

Most timing reports use ns for the time unit. However, you can use the PrimeTimecommandreport_unitsto report all the units, such as capacitance, resistance, time, and voltage units used by the design.

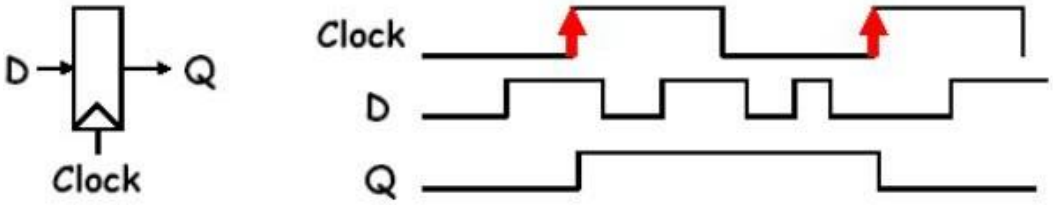
Clocked Storage Elements

Transparent Latch, Level Sensitive

Data passes through Latch when the clock is high, latched when the clock is low



D-Type Register or Flip-Flop, Edge-Triggered
Data captured on the rising edge of the clock held for the rest of the cycle



Delays

- Time taken by a signal to propagate through a Cell or Net
- Actual Path Delay is a sum of net and Cell Delays along the timing path
- Cell Delay is a function of Input Transition Time (Slew Rate), Total Output Load (NetCap + Sum of attached pin caps), and Process Parameters (Temperature, PowerLevel)

Intrinsic delay

Internal to the Cell from the Input pin to the Output pin caused by internal capacitance **propagation**

Delay

- Delay by a cell for a change of input signal to result a change in output signal as a function of Input Slew and Output load
- Propagation Delay can be Low to High (tPLH) and High to Low (tPHL)
- Maximum Propagation Delay (Clock to Q) is considered for the Setup check

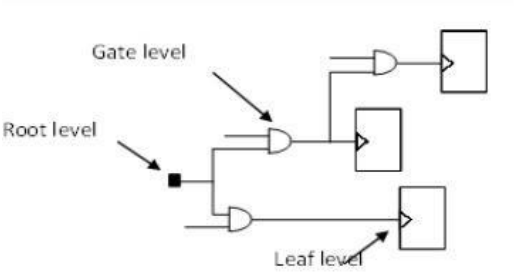
Contamination Delay

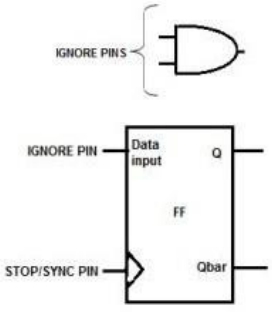
- Best case delay from valid input to output
- Minimum Propagation Delay (Clock to Q) which is called Contamination Delay is considered for Hold check

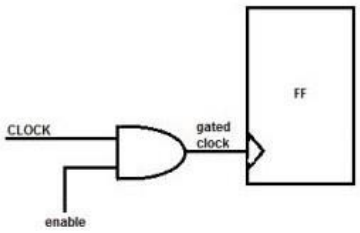
Net Delay

Total time for charging/discharging all the parasitic present in the given net

Pins related to Clock Design







Start/ Source / Root Pins
Source pin of a Clock

Stop/ Sink/ Leaf Pins
All Clock Pins of Flip Flops
The clock won't propagate after this Pin

Through pin
To make a Clock pin of a flop, not a CTS Leaf pin

Preserved Pin
If we need to preserve a pin w.r.t. location etc..

Exclude/ Ignore Pins
All non-clock pins (D pin of Flip Flops or combo logic inputs)

Not considered for Clock propagation.

Float Pins (Implicit Stop/ Macro Model)
Same as the Stop/ Sink Pin but its internal Clock Latency is considered for Clock Tree

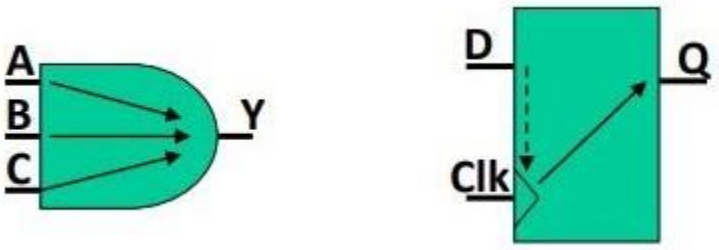
It is the entry pin of the Hard Macro.

Explicit Sync (Stop) Pin
Input of combo logic while considering Clock Tree

Important while considering Clock Gating.

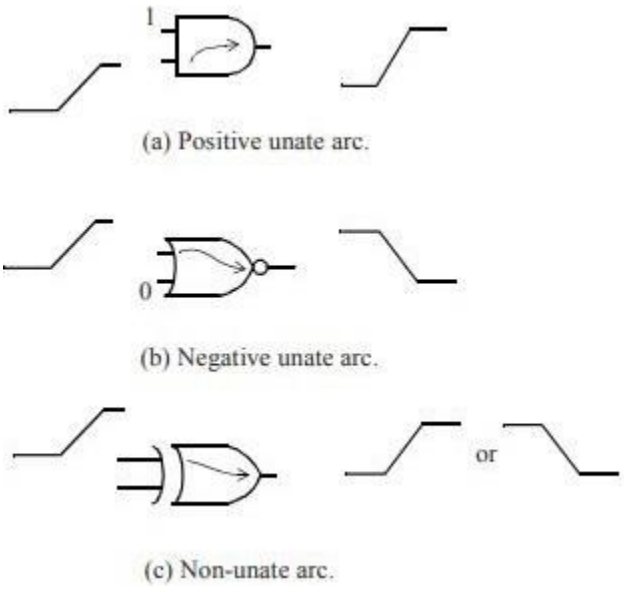
Explicit Exclude (Ignore) Sync Pin
Clock Pin of Flop is not considered a Sync/ Stop pin
This pin is due to the Clock Gating concept
In clock gating the signal will be given to AND Gate

Timing Arc



- The Timing Arc is internal to the cell
- Combinational Cells have Timing Arcs from each Input to each Output of the cell
- Flip-flops have Timing Arcs from the Clock Input pin to the Data Output Q pin(Propagation delay/ Delay Arc) and from the Clock Input pin to the Data Input D pin (setup, hold checks/ Constraint Arc)
- Latches have 2 timing arcs:
  - Clock pin to Output Q pin, when D is stable
  - Data D pin to Output Q pin when D changes (Latch is transparent)

Timing Unate



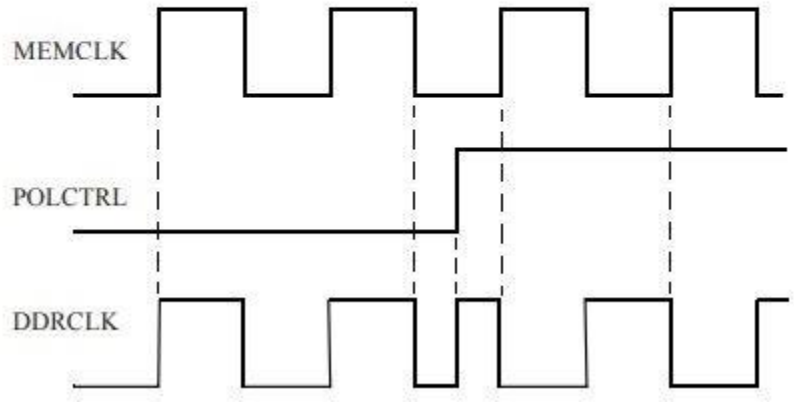
positive unate if a rising transition on an input causes the output to rise (or not to change) and a falling transition on an input causes the output to fall(or not to change). For example, the timing arcs for *AND* and *OR* type cells are positive unate. See Figure(a)

A **negative unate** timing arc is one where a rising transition on an input causes the output to have a falling transition (or not to change) and a falling transition on an input causes the output to have a rising transition (or not to change). For example, the timing arcs for *NAND* and *NOR*-type cells are negative unate.See Figure(b)

In a **non-unate timing arc**, the output transition cannot be determined solely from the direction of change of an input but also depends upon the state of the other inputs. For example, the timing arcs in an *XOR cell (exclusive-or)*are non-unate. See Figure(c).

Unateness is important for timing as it specifies how the edges (transitions)can propagate through a cell and how they appear at the output of the cell.

One can take advantage of the non-unateness property of a timing arc, such as when an xor cell is used, to invert the polarity of a clock. See the example below(figure). If inputPOLCTRL is a logic-0, the clock DDRCLK on output of the cell UXOR0 has the same polarity as the input clock MEMCLK. If POLCTRL is a logic-1, the clock on the output of the cell UXOR0 has the opposite polarity as the input clock MEMCLK.



Clock definitions in STA

Synchronous Clocks

2 clocks are synchronous w.r.t. each other

Timing paths launched by one clock and captured by another

Asynchronous Clocks
2 clocks are asynchronous w.r.t. each other
If no timing relation, STA can’t be applied, so the tool won't check the timing

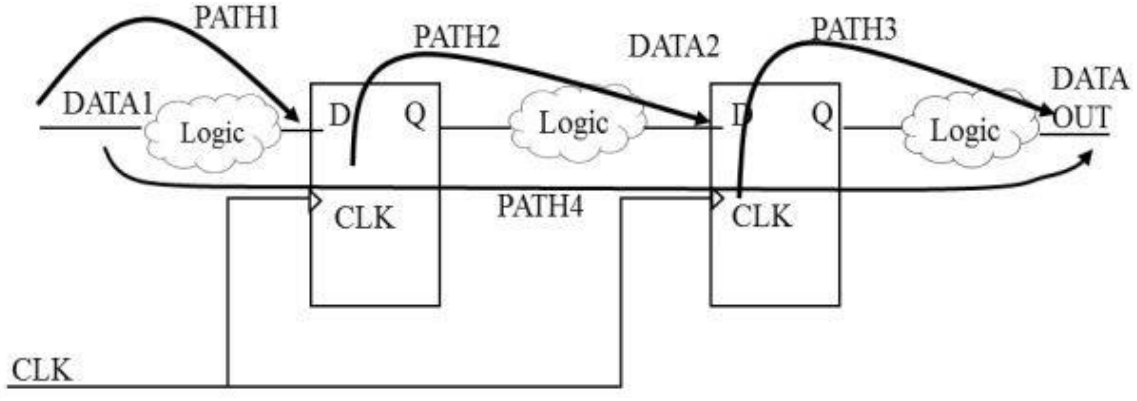
Mutually-Exclusive Clocks
Only one clock can be active at the circuit at any given time

Generated Clocks
Clock generated from a clock source as a multiple of the source clock frequency
The frequency can be a multiple or can be divided by the source clock

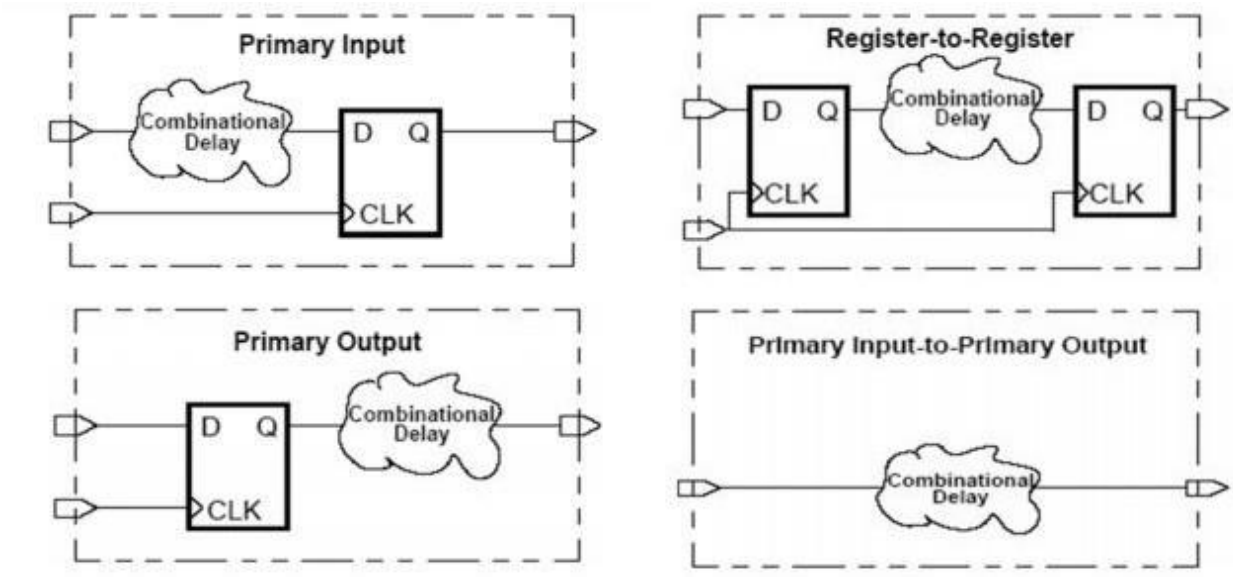
Virtual Clocks
Exists but not associated with any pin or port of the design
Used as a reference in STA to specify Input Delays and Output Loads relative to a clock (Needed to fix the Input2Reg and Reg2Output Violations)
By defining a Virtual Clock IO Constraints can be defined relative to this Virtual Clock with no specification of the source port or pin

Timing Paths
A Timing Path is a point-to-point path in a design which can propagate data from one flip-flop to another

- Each path has a start point and an endpoint
- Start point: Input ports or Clock pins of flip-flops
- Endpoints: Output ports or Data input pins of flip-flops



Timing Path Groups



Timing paths are grouped into path groups by the clocks controlling their endpoints

Input pin/port to Register
Delays off-chip + Combinational logic delays up to the first sequential device

Register to Register
Start at a sequential device
CLK-to-Q transition delay + the combinational logic delay + external delay requirements

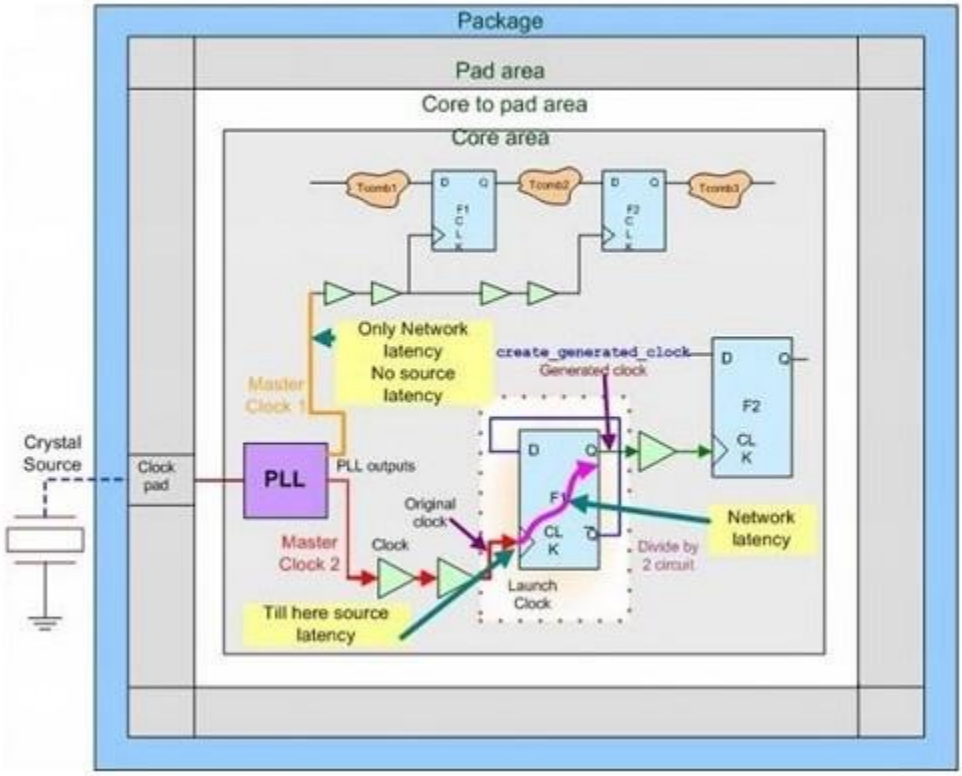
Register to Output pin/port
Delay and timing constraint (Setup and Hold) times between sequential devices for synchronous clocks + source and destination clock propagation times

Input pin/port to Output pin/port
Delays off-chip + combinational logic delays + external delay requirements

Clock Latency
Total time taken by the clock signal to reach the input of the register
Source latency is the time between clock sources to clock definition ports
Network latency is the time between clock definition ports to clock leaf cells in the design

Insertion Delay (ID)

ID is the clock latency, but after Clock Tree is synthesized.

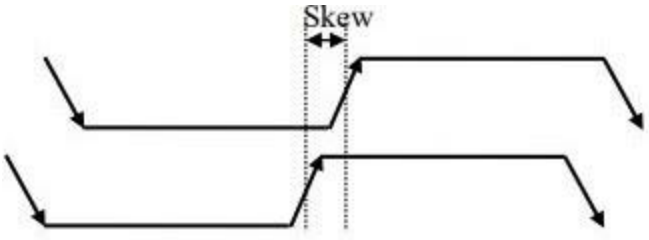


- ID is the physical delay and Clock Latency is the virtual delay
- Latency is a target given to the tool through SDC file or clock tree attribute file and Insertion Delay is the achieved delay value after CTS

Clock Uncertainty

Clock Uncertainty is the time difference between the arrivals of clock signals at registers in one clock domain or between domains

Uncertainties include Clock Skew, Clock Jitter and Clock Margin



Clock Skew refers to the absolute time difference in clock signal arrival between two points in the clock network

TLAUNCH_CLOCK - TCAPTURE_CLOCK = TSKEW

Positive Skew occurs when the Capture Clock is late w.r.t. Launch Clock

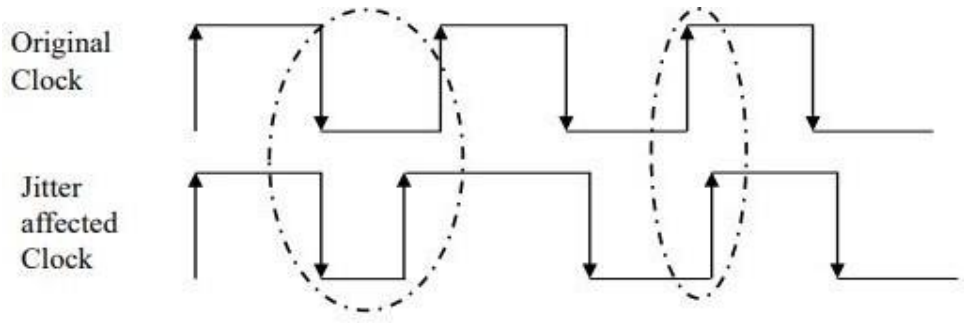
Negative Skew occurs when the Capture Clock is early w.r.t. Launch Clock

Local Skew is the Skew between the clock phase delays of two flip-flops which are the Source and Target flop of a path (Source and Destination flop)

Global Skew is the difference between the longest and shortest branch of a ClockTree (Maximum Insertion Delay – Minimum Insertion Delay)

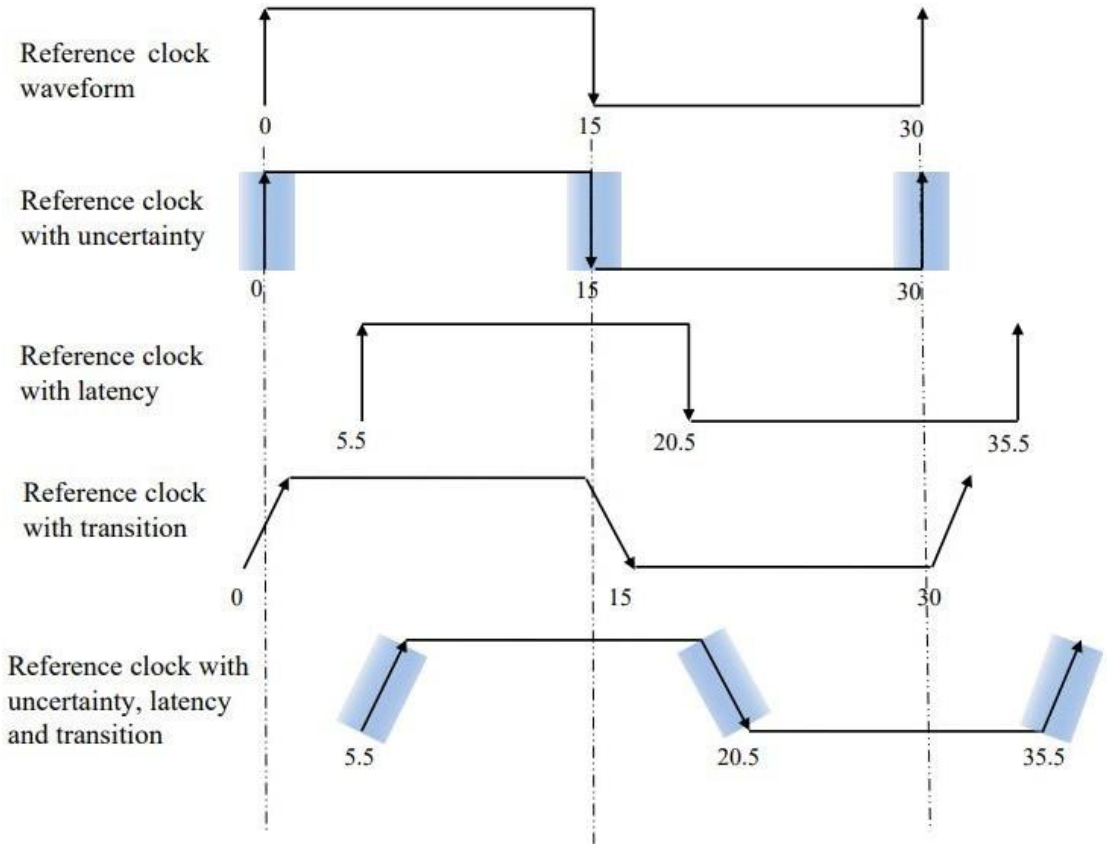
Clock Jitter

- Jitter is the short-term variations of a signal with respect to its ideal position in time
- The two major components of Jitter are random Jitter and deterministic Jitter
- Factors causing Jitter includes imperfections in Clock oscillator, supply voltage variations, Temperature variations, Crosstalk



Glitch

- Unexpected switching of any waveform
- Due to late arrival time of Gate and it is for a short period of time
- Cause extra delay and also it can cause extra power from false transitions



Pulse Width

- Pulse Width is the time between the active and inactive states of the same signal
- Minimum high pulse width is the amount of time after the rising edge of a clock, that the clock signal of a clocked device must remain stable
- Minimum low pulse width is the amount of time after the falling edge of a clock, that the clock signal of a clocked device must remain stable

Duty Cycle

- Percentage of clock period having high pulse
- Typically clock waveforms are of 50% Duty Cycle

Transition/ Slew

- Time taken by a signal to change the state (Volts/Second)
- Rise Slew (tR) is called Rise Time and Fall Slew (tF) is called Fall Time
- Minimum/ Maximum Transition is the Minimum/ Maximum slope allowed at leaf pins
- Transition affects Power Dissipation, Latency and Pulse width

Asynchronous Path

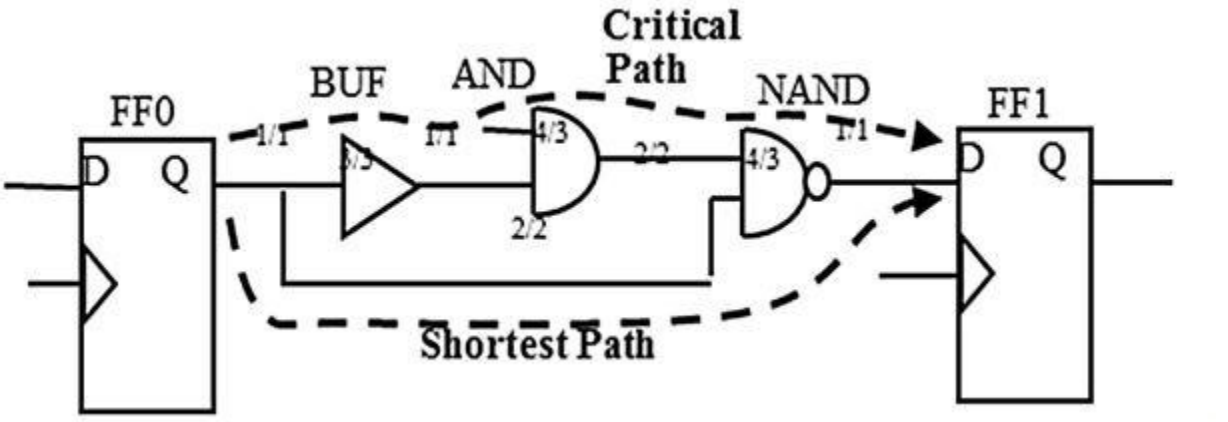
- A path from an input port to an asynchronous set or clear pin of a sequential element

Critical Path

- The path which creates longest delay
- Also called worst path/ late path/ max. path
- Timing sensitive functional paths no additional gates are allowed to be added to the path

Shortest Path

- One that takes the shortest time; this is also called the best path or early path or a min path



Clock Gating Path

- Path passed through a “gated element” to achieve additional advantages
- Clock Gating transformation does not change the state of the flops and register

Launch Path
Launch path is launch clock path which is responsible for launching the data at launch flip flop

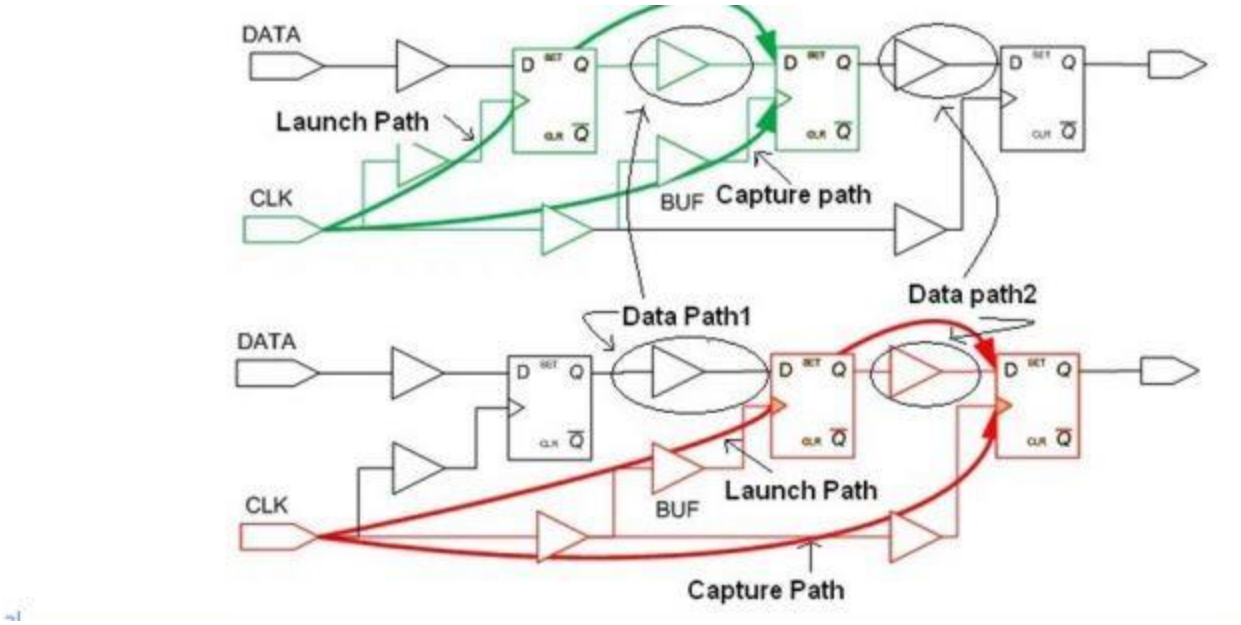
Capture Path
Capture path is capture clock path which is responsible for capturing the data at capture flip flop

Arrival Time
Launch path and data path together constitute arrival time of data at the input of capture flip-flop

Required Time
Capture clock period and its path delay together constitute required time of data at the input of capture register

Common Path Pessimism

- Same Clock Path may be a Launch Path for one Data Path and can be a Capture Path for another Data Path
- While doing OCV derating, same path may get both Min./ Max. delay
- But a path can have either as a Maximum delay or a Minimum delay (or anything in between) but never both delays at the same time
- STA tools will have techniques to remove artificially introduced pessimism between the Launch Clock Path and the Capture Clock Path



Slack

- Difference between Required Time (RT) and Arrival Time (AT)
- Positive Slack at a node implies that the arrival time at that node may be increased without affecting the overall delay of the circuit
- Negative Slack implies that a path is too slow, and the path must speed up if the whole circuit is to work at the desired speed

Setup Time

Setup time is the minimum amount of time the data signal should be held steady before the clock event so that the data are reliably sampled by the clock

TLAUNCH_CLOCK + TCLK-Q_MAX + TCOMB_MAX ≤ TCAPTURE_CLOCK - TSETUP

Hold Time

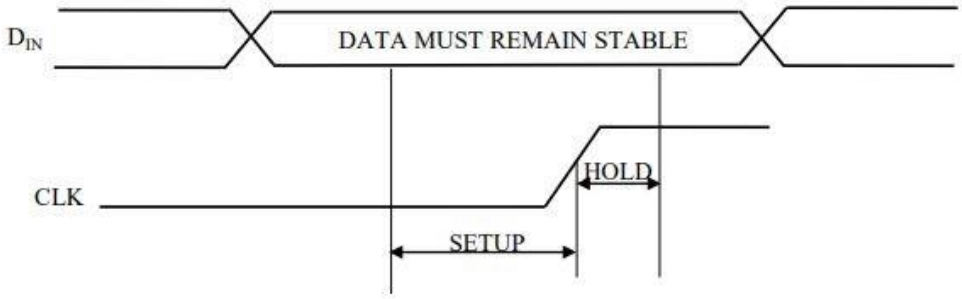
Hold time is the minimum amount of time the data signal should be held steady after the clock event so that the data are reliably sampled

TLAUNCH CLOCK + TCLK-Q_MIN + TCOMBO_MIN ≥ TCAPTURE_CLOCK + THOLD

Setup Time and Hold Time Violations

- If Setup time, T_{SETUP} for a flip-flop and if the data is not stable before T_{SETUP} from the active edge of the clock, then there is a Setup Violation at that flip-flop

- If hold time, T_{HOLD} for a flip-flop and if the data is not stable after T_{HOLD} time from the active edge of clock, then there is a hold violation at that flip-flop
- For a single-cycle circuit the signal has to propagate through Data path in one clock cycle



Recovery Time

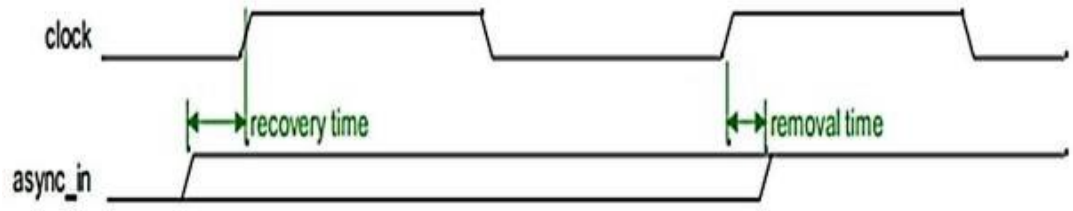
Recovery time is the minimum time that an asynchronous control input pin must be stable after being de-asserted and before the next clock transition (active edge)

Removal Time

Removal time is the minimum time that an asynchronous control input pin must be stable before being de-asserted and before the previous clock transition (active edge)

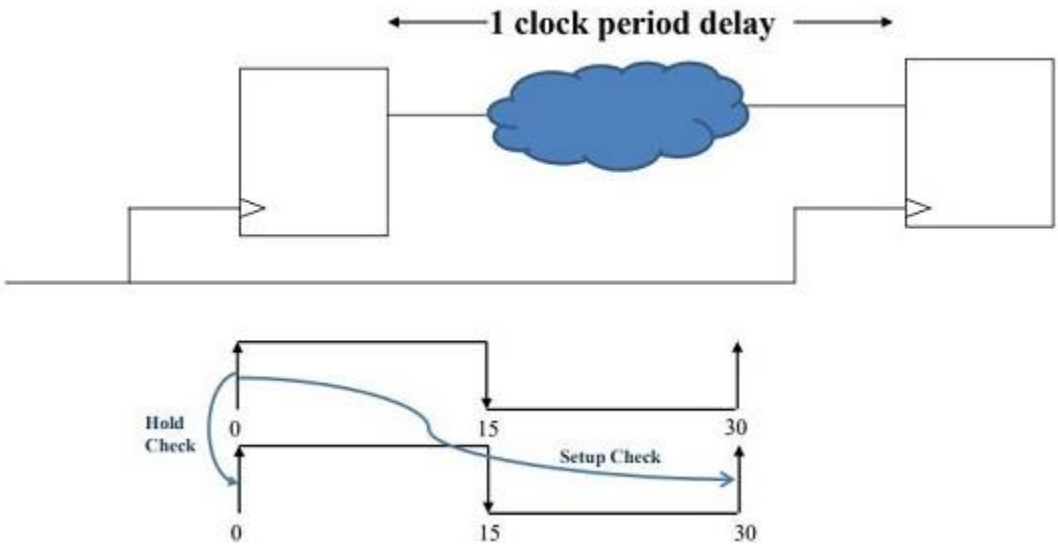
Recovery Time and Removal Time Violations

- This check is to ensure that the asynchronously signal rise/ fall edge is not occurring at the clock edge; it should be some time before or after the clock edge
- If that violates, then Recovery Time and Removal Time Violations
- Although a flip-flop is asynchronously SET or CLEAR, the negation from its RESETstate is synchronous



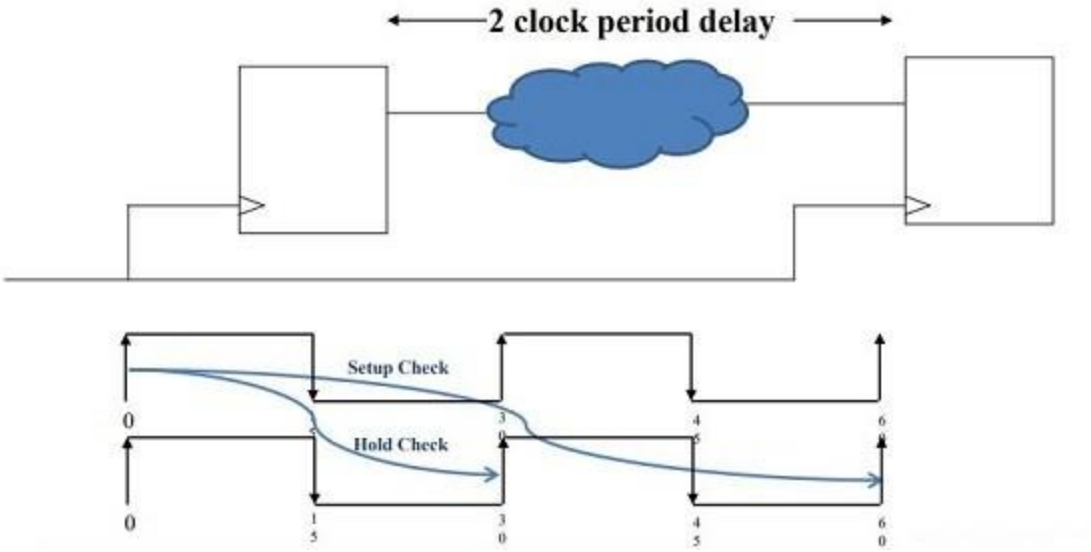
Single Cycle Path

- Timing path that is designed to take only one clock cycle for the data to propagate from the start point to the endpoint
- Start point and endpoint are flops clocked by the same clock
- By default tool will consider all timing paths as single cycle paths



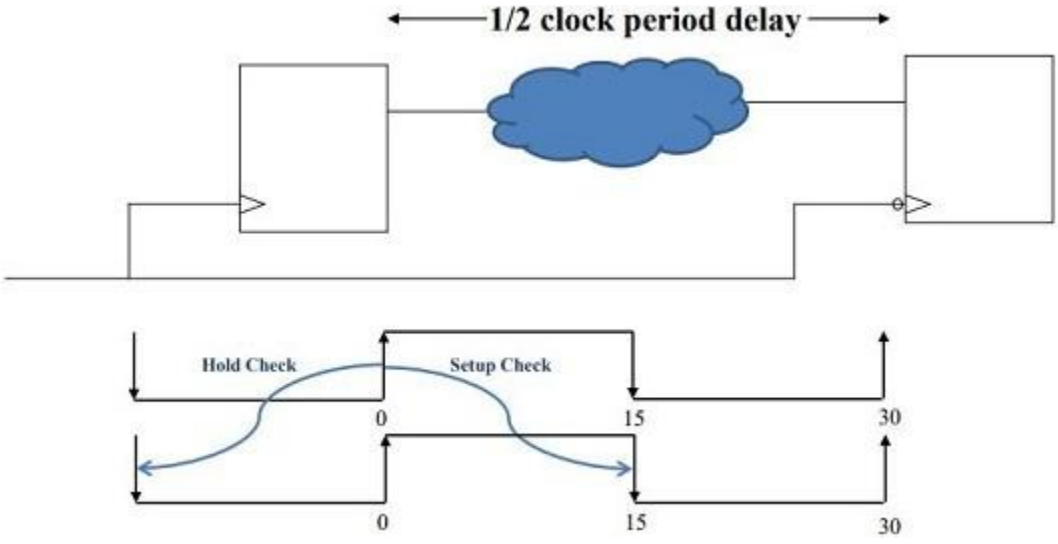
Multi-Cycle Path

- Timing path that is designed to take more than one clock cycle for the data to propagate from the start point to the endpoint
- Start point and endpoint are flops clocked by the same clock
- Need to specify the Launch edge and Capturing edge in SDC



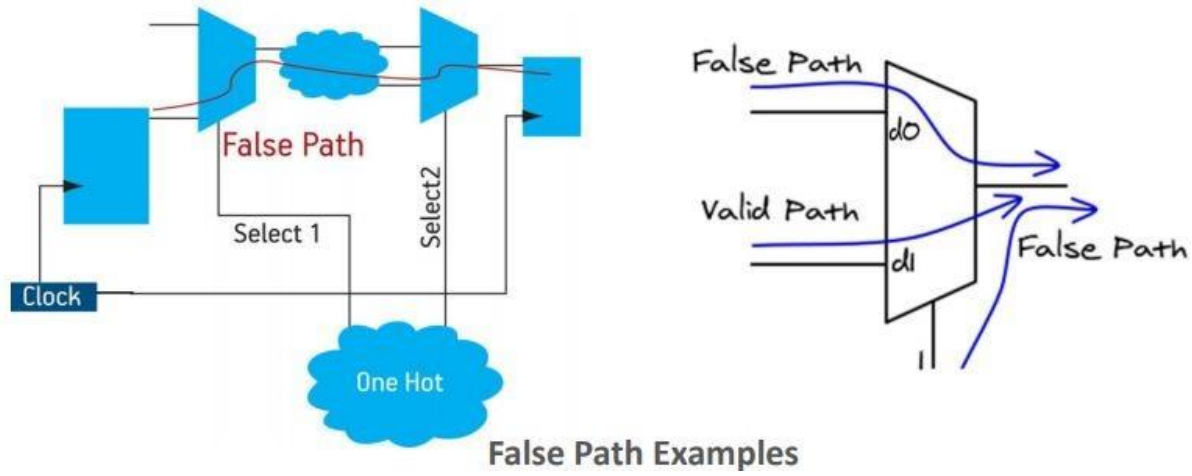
Half Cycle Path

- Timing path that is designed to take half clock cycle (both of the clock edges) for the data to propagate from the start point to the endpoint
- Start point and endpoint are flops clocked by the same clock
- No need to specify the Launch edge and Capturing edge in SDC, since the tool can identify it from the netlist



False Path

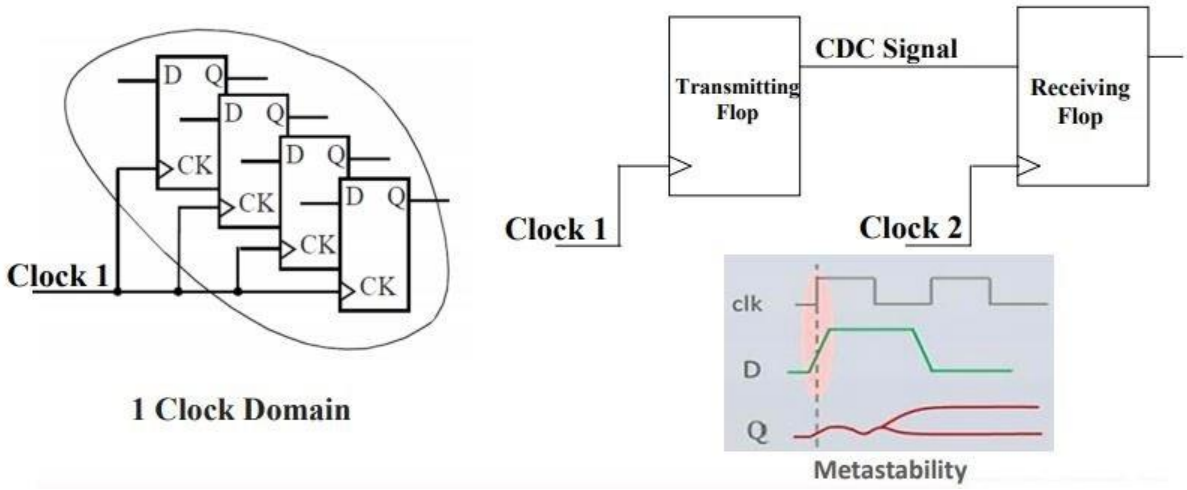
- Physically exist in the design but are Logically/ Functionally inactive/incorrect path
- Means no data is transferred from Start Point to End Point
- The goal in STA is to do timing analysis on all “true” timing paths, so these paths are excluded from timing analysis
- Similarly timing can be disabled for a pin or port or cell where the delay will be computed but won’t report it



False Path Examples

Clock Domain Crossing (CDC)

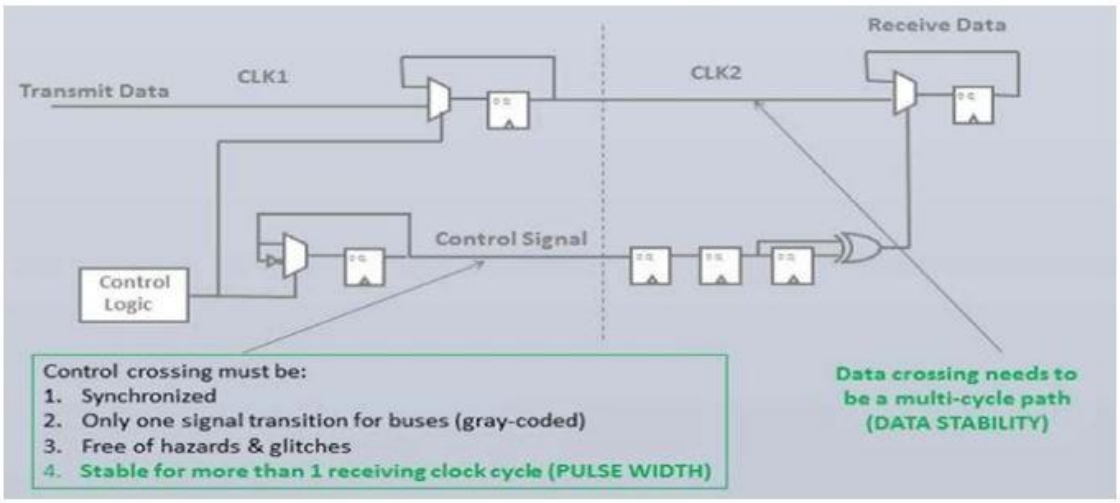
- For designs with Asynchronous Clock Domains, the CDC signal violates the Setup/Hold window of the receiving clock, resulting in metastability
- Metastability results in unpredicted values and unpredictable delays
- Those clocks has to be balanced together else, due to difference in the latency that may lead to timing violations
- Max. Delay Constraint is used to make CDC paths to get synchronized



Clock Domain Synchronization Scheme

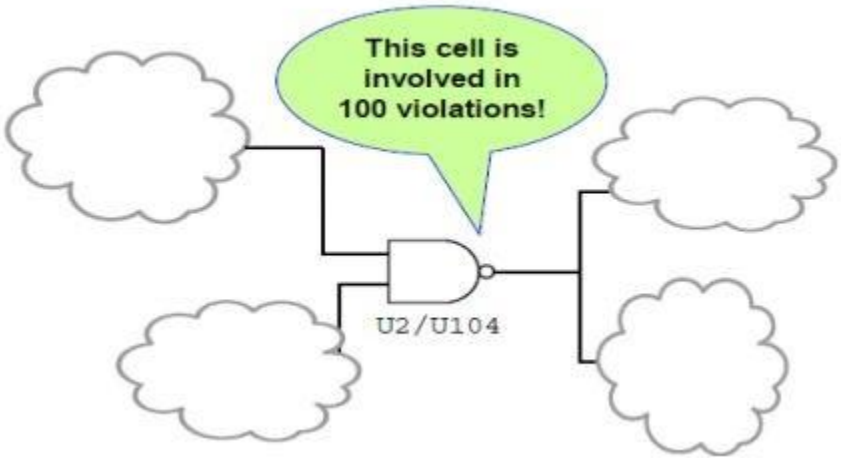
- Pulse Width check
  - The control signals is stable for longer than one receive clock period
  - Ensures that data will not be lost due to inadequate width of the control signal

- Data Stability check
  - The data updated by the transmit domain cannot be captured by the immediately following receive clock edge
  - Ensures that the captured data will not be metastable in the receive domain



Bottleneck Analysis

- Lists the cells causing the timing violations on multiple paths
- By identifying and fixing the violation caused by a Bottleneck Cell improved timing can be achieved



Multi-VT Cells

- Different threshold voltages are achieved by implanting dopants in different concentration
- Need Multi-VT Library
- Sub-threshold leakage varies exponentially with VT compared to the weaker dependency of delay over VT
- If the optimization target is power performance, first use the HVT cells library and then try LVT cells
- If the optimization target is to meet timing then first use LVT cells and then HVT cells
- If you swap the capture flop from SVT to LVT or HVT, there will be very minimal setup/hold impact in most flops, it is of zero impact for hold
- If you swap the launch flop from SVT to LVT or HVT, Setup will be improve and hold will be impacted correspondingly

High Voltage Threshold (HVT)

- Use in non-timing critical paths
- Use in power critical paths
- Has low leakage and low speed

Low Voltage Threshold (LVT)

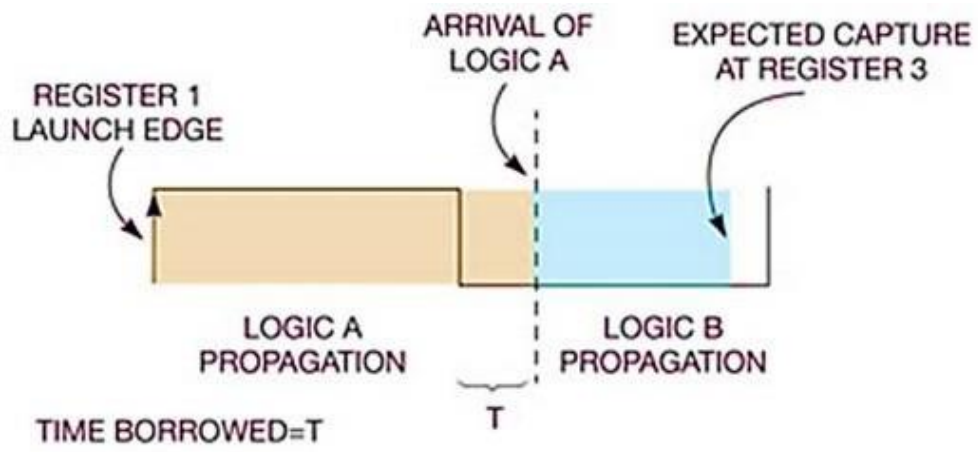
- Use in timing critical paths
- Use in non-power critical paths
- Has high leakage and high speed

Standard Voltage Threshold/ Regular Voltage Threshold (SVT/ RVT)

- Medium delay and medium power requirement

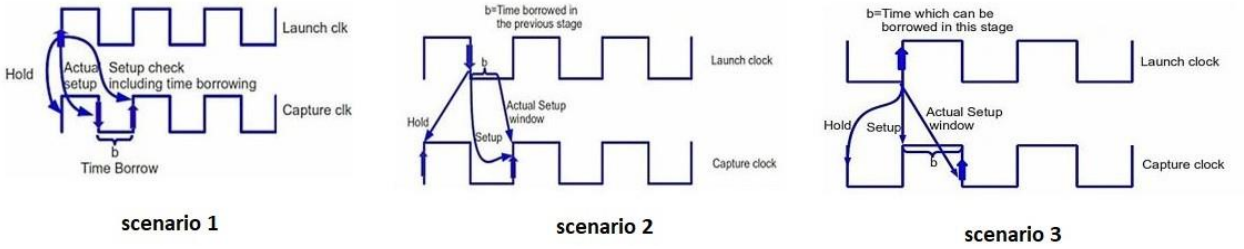
Time Borrowing

- Time Borrowing is basically for Latched based Timing Analysis
- Edge-triggered flip-flops change states at the clock edges, whereas latches change states as long as the clock pin is enabled
- In latch-based design longer combinational path can be compensated by shorter path delays in the subsequent logic stages
- The technique of Borrowing Time from the shorter paths of the subsequent logic stages to the longer path is called Time Borrowing or Cycle Stealing



- Time Borrowing typically only affects setup slack calculation since time borrowing slows data arrival times
- When the clocks of the Launching and Capturing Latches are out of phase, time borrowing is not to happen
- Timing borrowing can be multistage
- Maximum Borrow Time: Clock Pulse Width minus the library Setup Time of the Latch
- Negative Borrow Time: Arrival Time minus the clock edge is a negative number, the amount of time borrowing is negative (no borrowing)

Time Borrowing: Scenarios



scenario 1 scenario 2 scenario 3

- Scenario 1: When data is launching from a positive edge triggered flip flop and capture is to a negative level sensitive latch
- Scenario 2: When a launch is from a negative level sensitive latch and capture is to a positive edge triggered flip flop
- Scenario 3: When launch and capture are from positive level-sensitive latches

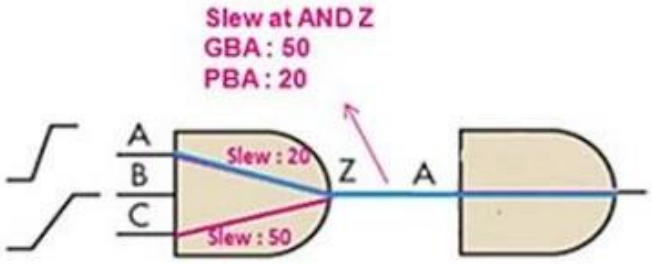
Types of Static Timing Analysis

Path Based STA (PBA)

- First, extract all possible topological paths
- Next, for each path calculate its delay and compare it with the endpoint (required)value
- Calculate the Arrival Time (AT) by adding cell delay in timing paths
- Check all path delays to see if the given Required Arrival Time (RAT) is met

Graph-Based STA (GBA)

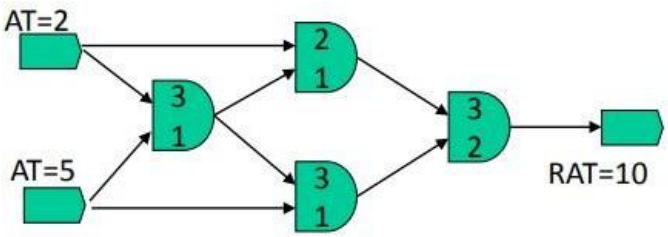
- Two types of timing data :
- Arrival times, AT (propagated forward from inputs)
- Required Arrival Times RAT (propagated from outputs)
- Slack is calculated on every design element: Slack = RT – AT



Difference between Path-Based STA(PBA) and Graphh Based STA(GBA)

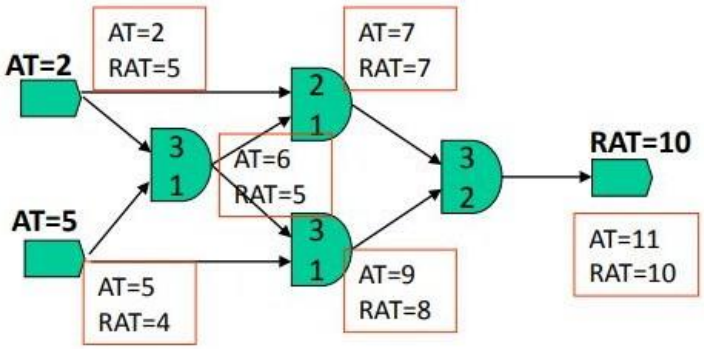
Path Based STA (PBA)	Graph Based STA (GBA)
• Path specific STA • Wont use worst skew • Intensive computation required • Less Pessimistic • More accurate • Timing constraints will be checked at end points of the timing paths • Not favorable for large no. of paths • PBA select either max. path or min. path • Timing information associated with topological paths (collections of design elements) • Traces every possible timing paths • Always done after GBA	• Parameter based STA • Wont use worst skew • Not so intensive computations • More Pessimistic • Less accurate compare to PBA • Timing constraints will be checked at each node of the timing paths • Not favorable for large no. of corners • GBA the max. path alone is selected • Timing information associated with discrete design elements (ports, pins, gates) • Its incremental; breadth based

Block-based STA vs. Path-based STA (example)



Path-based:

2+2+3 = 7 (OK)
2+3+1+3 = 9 (OK)
2+3+3+2 = 10 (OK)
5+1+1+3 = 10 (OK)
5+1+3+2 = 11 (Problem!)
5+1+2 = 8 (OK)



Block-based:

Critical path is determined as collection of gates with the same, negative slack:

Slack = RT – AT

In our case, we see one Critical path with slack = -1

STA Numericals

Problem 1:

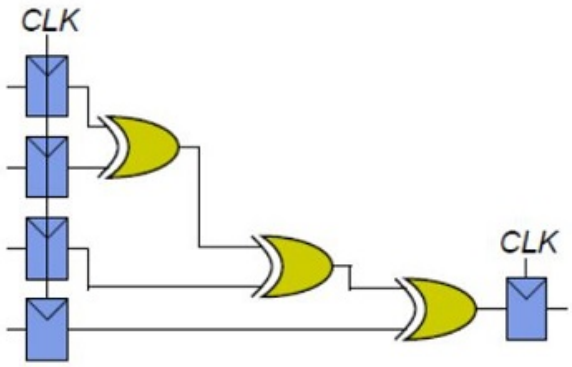
In the following circuit

Each flip flop has:

- Setup time of 60ps
- Hold time of 20ps
- Clock-to-Q maximum delay of 70ps
- Clock-to-Q minimum delay of 50ps

Each XOR gate has:

- Propagation delay of 100ps
- Contamination delay of 55ps

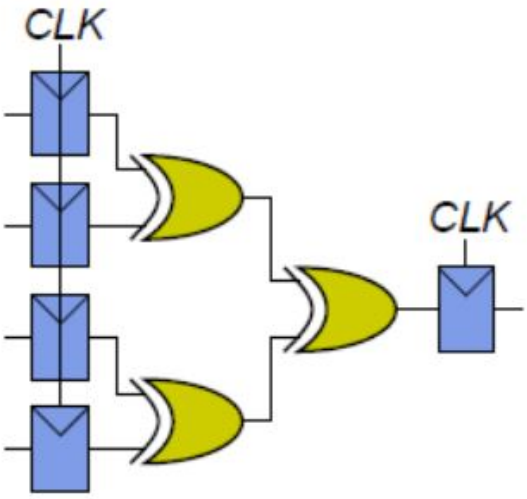


a. If there is no clock skew, what is the maximum operating frequency of this circuit?
b. How much clock skew can the circuit tolerate before it might experience a hold time violation?
c. Redesign the circuit so that it can be operated at 3GHz frequency. How much clock skew can your circuit tolerate before it might experience a hold time violation?

Solution

a.

Tc ≥ Tpcq + Tpd + Tsetup
Longest path:
Tc ≥ Tpcq + 3*Tpd + Tsetup
Tc ≥ 70 + 3*100 + 60 = 430 ps
Max Frequency = 1/Tc = 2.33 GHz



b.

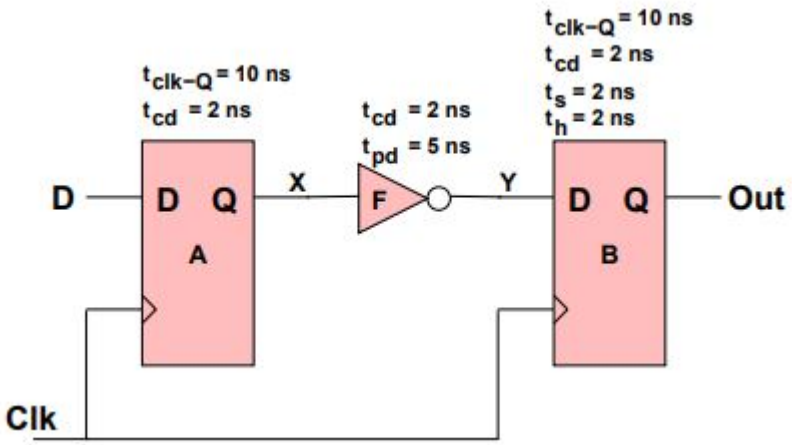
Tccq + Tcd≥Thold + Tskew
Shortest Path:
Tccq + Tcd ≥ Thold + Tskew
50 + 55 ≥ 20 + Tskew
Tskew ≤ 85 ps

c.

Tc ≥ Tpcq + 2*Tpd + Tsetup + Tskew
Tc ≥ 330 + Tskew
Tccq + 2Tcd ≥ Thold + Tskew
Tskew≤ 140 ps

Problem 2:

Q. Determining the Max. Clock Frequency for a Sequential Circuit.



Solution

Before starting timing analysis, consider the flow of data in this circuit in response to a rising clock edge, starting at flip-flop A.

1. Following the rising clock edge on Clk, a valid output appears on signal X after tClk−Q = 10 ns.

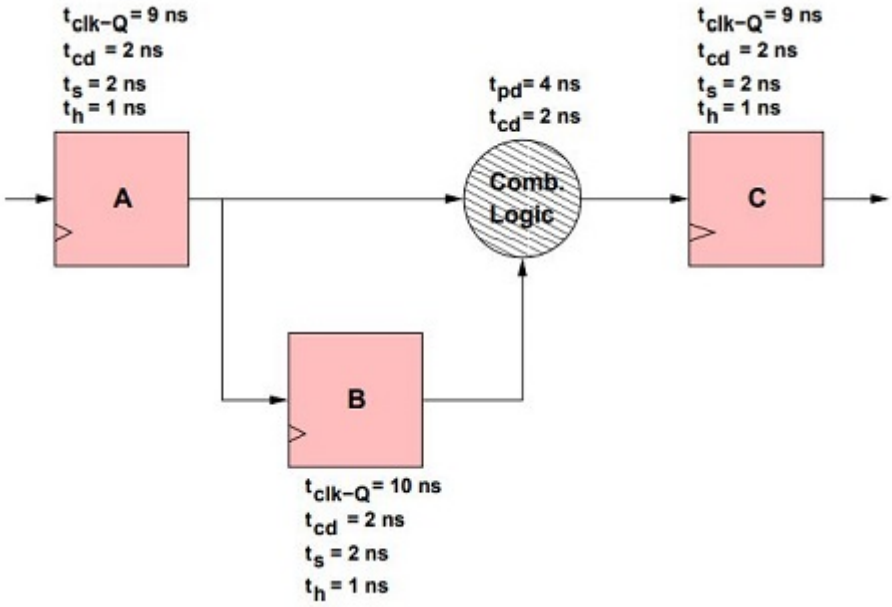
2. A valid output Y appears at the output of inverter F, tpd = 5 ns after a valid X arrives at the gate. 3. Signal Y is clocked into flip-flop B on the next rising clock edge. This signal must arrive at least ts = 2ns before the rising clock edge.

As a result, the minimum clock period, Tmin of the circuit is:
Tmin = tClk − Q(A) + tpd(F) + ts(B)
= 10ns + 5ns + 2ns = 17ns

maximum clock frequency of the circuit is 1/Tmin = 1/17ns = 58.8 MHz

Problem 3:

Q. Determining the Max. Clock Frequency the Sequential circuit shown below.



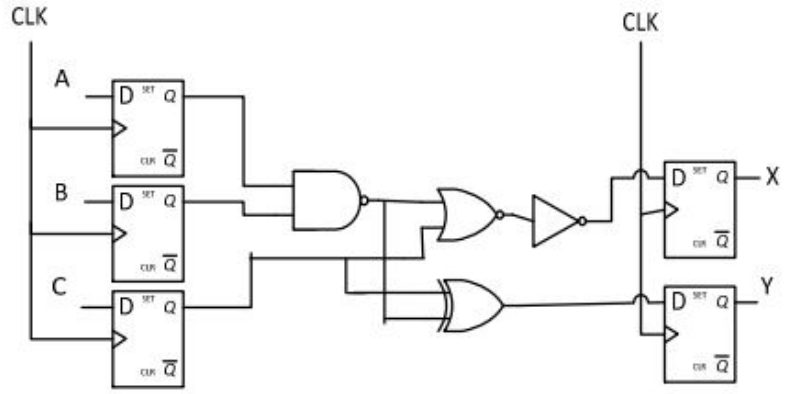
In a typical sequential circuit design there are often millions of flip-flop to flip-flop paths that need to be considered in calculating the maximum clock frequency. This frequency must be determined by locating the longest path among all the flip-flop paths in the circuit. For example, consider the circuit shown in above. there are three flip-flop to flip-flop paths (flop A to flop B, flop A to flop C, flop B to flop C),the delay along all three paths are:

- TAB = tClk−Q(A) + ts(B) = 9 ns + 2 ns = 11 ns
- TAC = tClk−Q(A) + tpd(Z) + ts(C) = 9 ns + 4 ns + 2 ns = 15 ns
- TBC = tClk−Q(B) + tpd(Z) + ts(C) = 10 ns + 4 ns + 2 ns = 16 ns

Since the TBC is the largest of the path delays, the minimum clock period for the circuit is Tmin = 16 ns and the **maximum clock frequency** is 1/Tmin = 62.5 MHz.

Problem 4:

Q. For the circuit given below calculate.
• Maximum clock frequency for reliable operation.
• The amount of clock skew the circuit can tolerate if it needs to operate at 5 Ghz.
• How much clock skew the circuit can tolerate before it experiences a hold time violation?



Gate	Tpd(ps)	Tcd(ps)
2-input NAND	20	15
2-input NOR	30	20
2-input XOR	60	40
NOT	15	10

Flip-Flop (clock-to-q) propagation delay (tpcq) = 35 ps
Flip-Flop (clock-to-q) contamination delay (tccq) = 20 ps
Flip-Flop data setup time (ts) = 30 ps
Flip-Flop data hold time (th) = 10 ps

Solution

a.

Period > (FF propagation delay) + (max combination circuit delay) + (FF Setup time) + (max clock skew)
Period > 35 + (60+20) + 30 +0 ps
Period > 145 ps
F < 1/(145 ps)
F < 6.8965 GHz.

b.

At F= 5 GHz Period = 1/(5 Ghz) = 200 ps.
Max clock skew = Clock period – (FF propagation delay + max combination circuit delay + FF Setup time)
Max clock skew = 200 – (35 + (60+20) + 30) = 200 – 145 = 55 ps.

c.

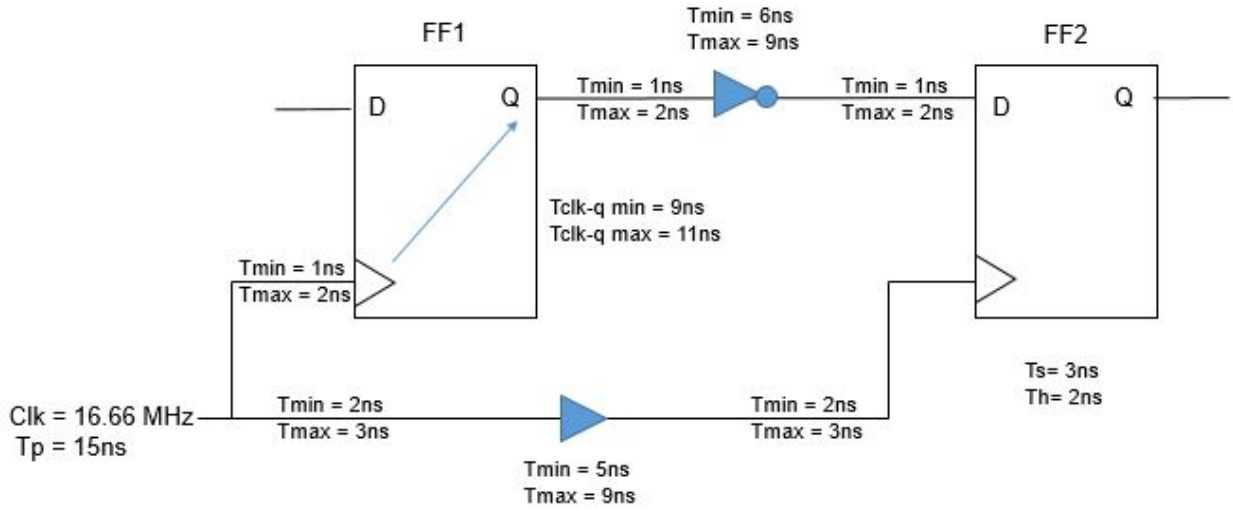
For hold time violation to NOT occur.
Hold time <= (FF contamination delay) + (min combinational circuit delay) -

(max clock skew)
So hold time will get violated when
Max clock skew > (FF contamination delay) + (min combinational circuit delay) – (Hold Time)
Max clock skew > 20 + (20+10) – 10
Max clock skew > 40 ps

Problem 5:

Q. For the circuit given below calculate.

a. setup slack
b. hold slack



Solution

before proceeding with the solution we should know :

setup slack = RTmin(minimum required time) - ATmax(maximum arrival time)
where; RTmin ≥ ATmax to satisfy setup time

Hold Slack = ATmin(minimum arrival time) - RTmax(maximum arrival time)
where; ATmax ≥ RTmin to satisfy hold time

Let's solve this..

a. Setup Slack

ATmax = 2 + 11 + 2 + 9 + 2 + 3 = 29ns
RTmin = 2+ 5 + 2 + 15(here 15ns is the time perid) = 24ns

Setup slack = ATmax - RTmin = 24 - 29 = -5ns

we can see setup is violating as ATmax is less than RTmin.

b. Hold Slack

ATmin = 1 + 9 + 1 + 6 + 1 - 2 (here Thold is considered and subtracted) = 16ns
RTmax = 3 + 9 + 3 = 15ns

Hold Slack = ATmin - RTmax = 16 - 15 = 1ns

here ATmax ≥ RTmin i.e 16 ≥ 15 (no violation)