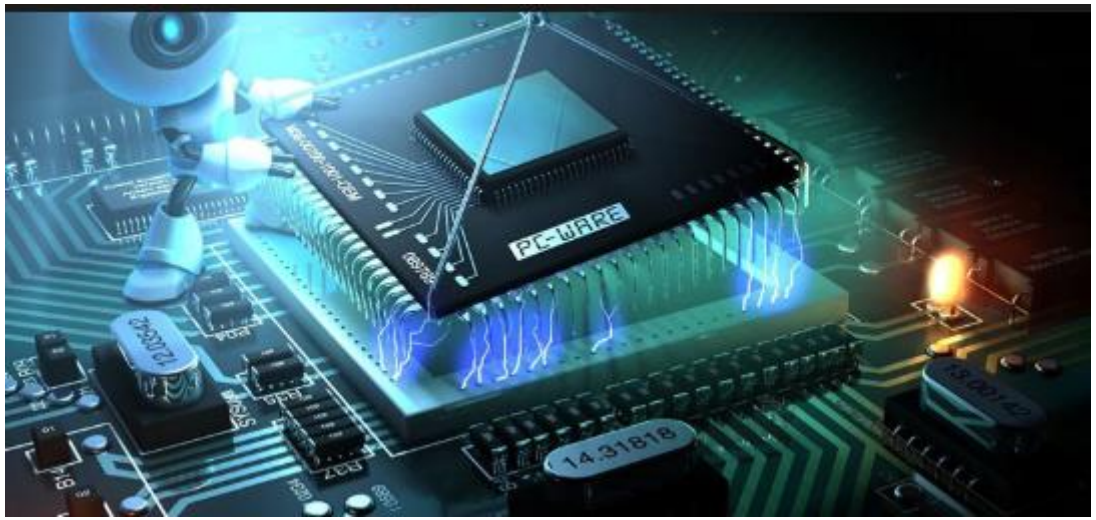
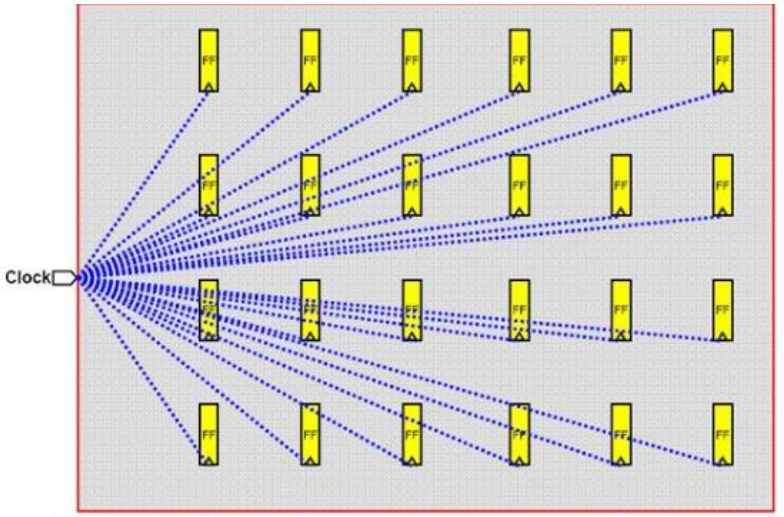
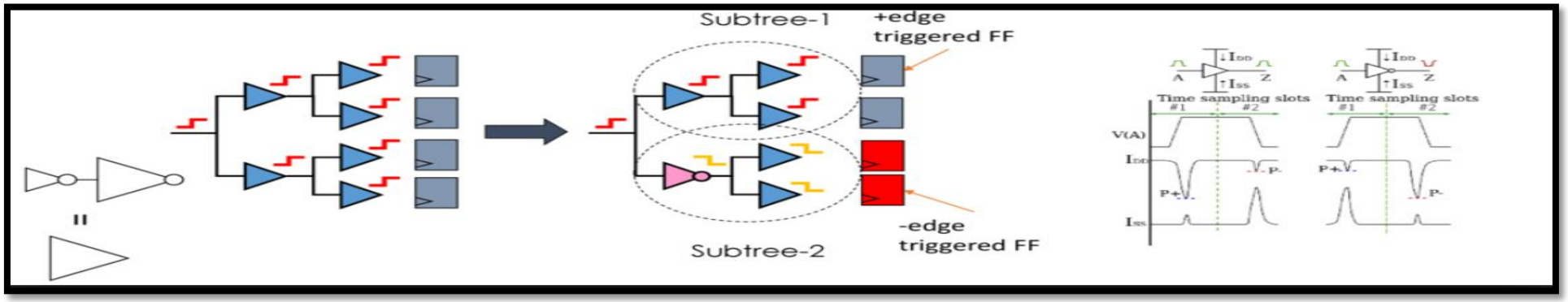




CTS & post CTS optimization



Pre cts checks (cts readiness)

Below are the few pre cts sanity checks to be performed before starting “CTS”

1. Congestion b/w macro channels and b/w standard cells – Acceptable congestion
2. Cell/Pin/Congestion density map analysis
3. Reg2Reg WNS/TNS timing -Acceptable or/not
4. Good understanding of “check_physical_design –stage pre_clock_opt” report – It checks for readiness of placement, netlist and Design constraints
5. Utilization jump from Synthesis to Place_opt (utilization growth shouldn’t be big).
6. Make sure NDR’s are applied for all the clock nets /Rechecking should be done :-p
7. LEC should be run to check functional equivalent of design intent .
8. Place_opt netlist should be passed to DFT team to cross the scan chain connectivity

Is design ready for CTS?

- **`check_physical_design -stage pre_clock_opt` checks for:**
  - Designs is placed
  - Clocks have been defined
  - Clock roots are not hierarchical pins (see below for support)
- **`check_clock_tree` checks and warns if:**
  - A clock source pin is a hierarchical pin (see below for support)
  - A generated-clock with improperly specified master-clock
  - A clock tree has no synchronous pins
  - There are multiple clocks per register

Design status before CTS

1. Floorplan – Completed
2. Power and ground nets – prerouted
3. Placement – Completed
4. Estimated congestion – acceptable Horizontal and Vertical
5. Estimated timing – acceptable WNS/TNS value (Expected 0 ~ns)
6. Estimated Logical DRC’s
7. High fanout nets are synthesized .
   - Reset , scan enable synthesized with buffers
   - Clocks are still not buffered

Method to specify the NDR rules

- **Define the NDR rules:**

```
define_routing_rule MY_ROUTE_RULES   \
  -widths {METAL3 0.4 METAL4 0.6 METAL5 0.6} \
  -spacings {METAL3 0.5 METAL4 0.65 METAL5 0.65}
```

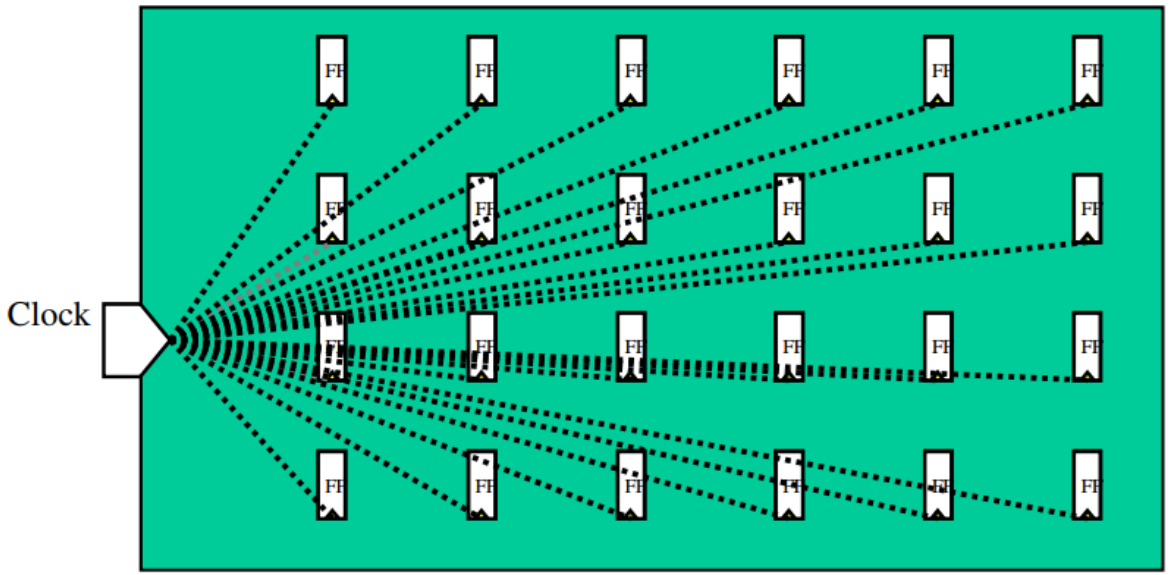
- **Configure the clock tree routing:**

```
set CLK_NETS [get_nets -of [all_fanout -flat -clock_tree]]
set_net_routing_rule -rule MY_ROUTE_RULES $CLK_NETS
set_net_routing_layer_constraint -min_layer METAL3 \
                                 -max_layer METAL5 $CLK_NETS
```

You may also specify the layers to be used for clock tree routing

CTS problem – Connectivity before CTS

- All clock pins are driven by a single clock source
- All clock pins are from a source of clock pulses in various geometrical distances



What is CTS?

- It's a kind of a tree to provide the clock to all of it's sinks (Binary tree, H-Tree etc.)
- The basic of CTS is to develop the interconnect that connect the system clocks to all the cells in the chip that uses the clock
- The primary task of CTS is vary the routing paths, placement of clocked cells and clock buffers to meet clock tree target
- When you synthesize your clock network with respect to particular clock is know as clock tree synthesis

Review of basic terminology

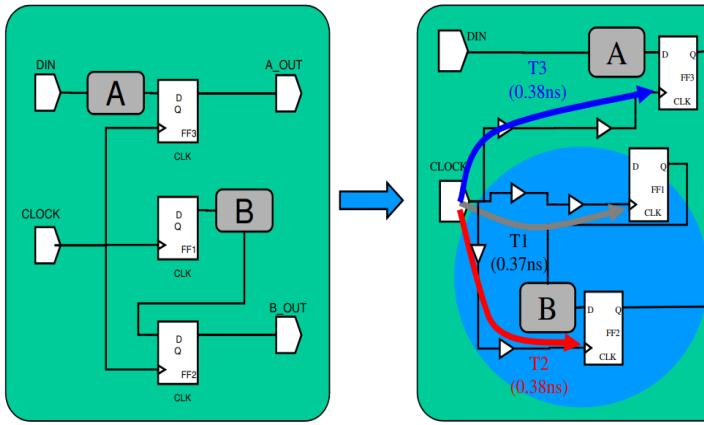
1. PLL
2. Clock period definitions
3. Clock latency
   - Source latency
   - Network latency
4. Clock uncertainty
5. Setup & Hold
6. Constraints – Max capacitance , Max fanout & mac transition
7. Skew

   Global skew, Local skew & Useful skew

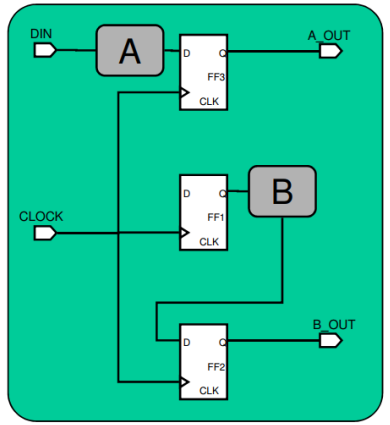
Clock skew types

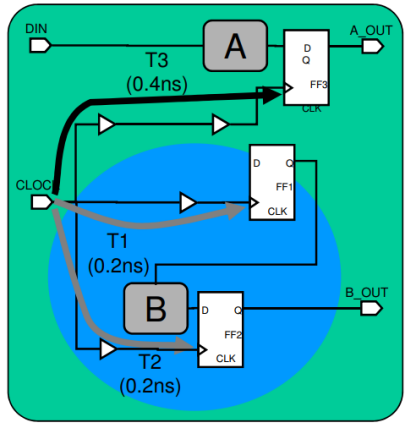
1. Global skew
2. Local skew
3. Usefull skew/Clock skewing/Push pull technique

Global skew

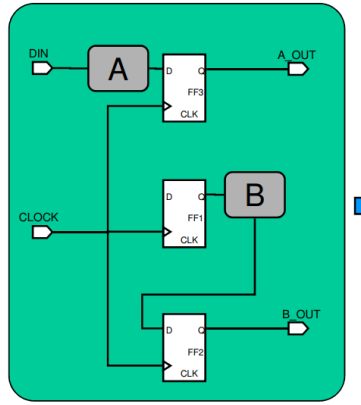


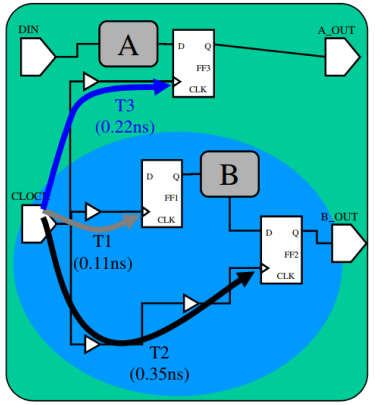
Local skew





Useful skew



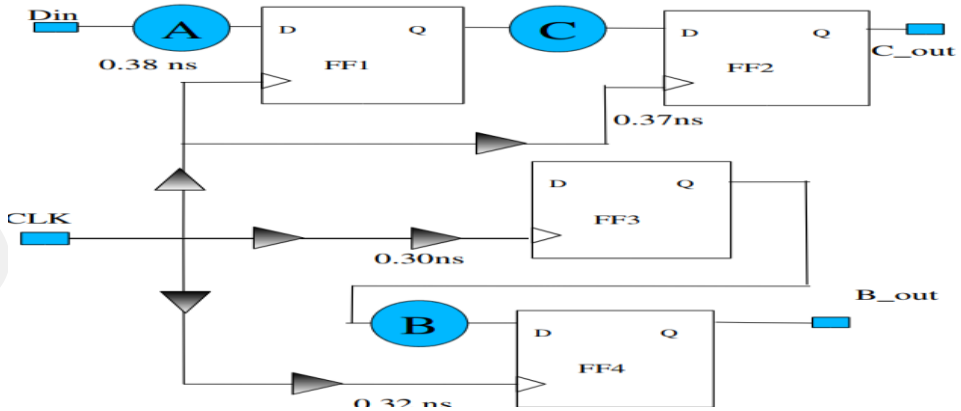


Local skew

- Local skew – It is the skew b/w two flops which were interacting logically with each other on the same clock domain.
- FF1 and FF2 are interacting with each other. Local skew = 0.38 – 0.37= 0.01ns

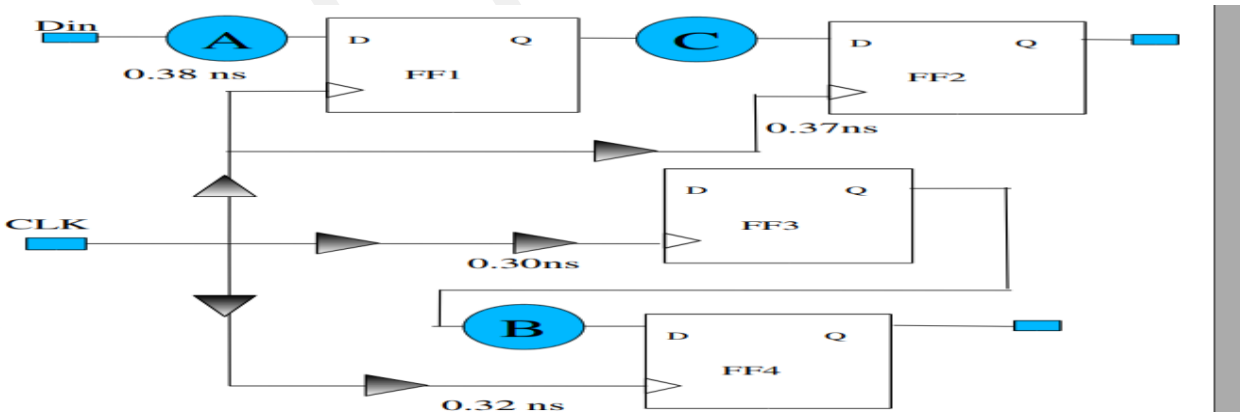
FF3 and FF4 are also interacting with each other local skew= 0.32-0.3=0.02ns

- Local skew will take lot of time and less buffers in your design



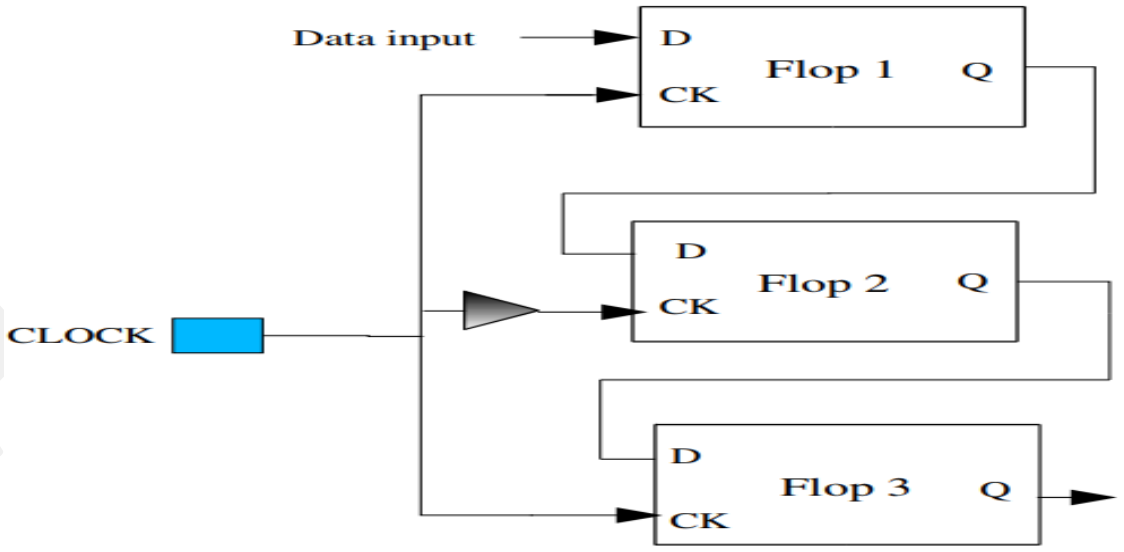
Global skew

- It is skew differences in the logest and short clock delay path from clock definition point to clock sink within the same domain .
- Global skew = 0.38-0.30=0.08ns
- CTS engine will be faster in G.Skew calculation.
- Global skew add mode buffers in your design , we should be carefull about it.



Useful skew

- Useful skew concept is used to fix the setup violations
- Lets suppose flop1 and flop2 is failing setup by -1ns and path from flop2 and flop3 is passing by +1ns. We can skew by 1ns to fix the violations on capture flow (F2).



Different types of pins attribute in CTS stage

1. Sink/Stop pin – Example Register clock pin

2. Ignore/Exclude pin – Example: Output clock port/ Input of combinational not connected any sinks further fanout

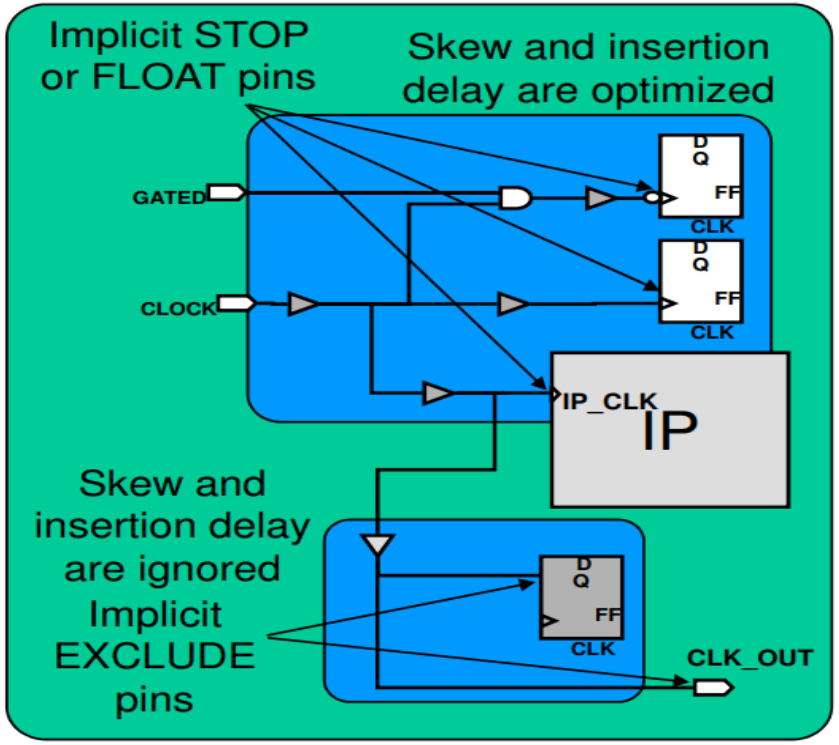
3. Through/ Non stop pin – Example : generated logic ck pin

4. Fload pin - Example : Macro clock pin

Clock Tree Exceptions: Stop, Float and Exclude Pins

Exceptions

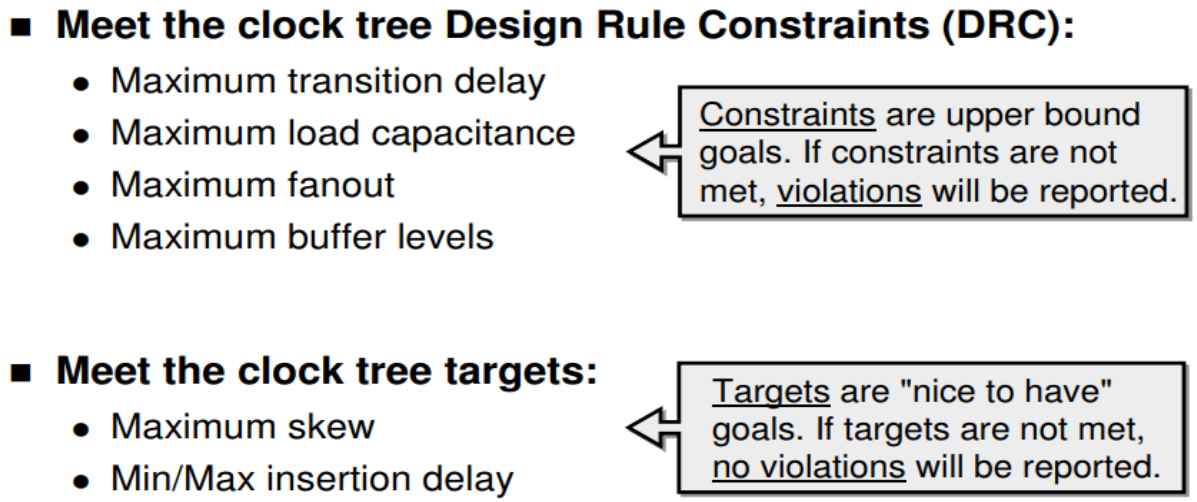
- ✓ STOP Pins
  - CTS optimizes DRC and clock tree targets (skew, insertion delay)
- ✓ FLOAT Pins
  - Like Stop pins, but with delays on clock pin
- ✓ EXCLUDE Pins
  - CTS ignores both clock tree DRCs and targets



CTS specifications/Goals

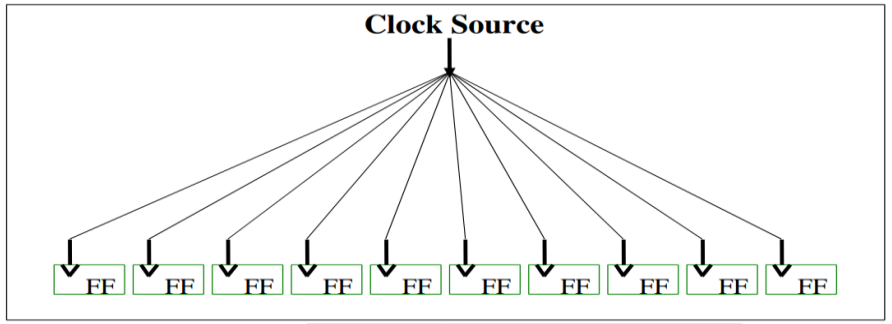
1. CTS buffering constraints
   - Max fanout
   - Max capacitance
   - Max transition

2. Meet the clock targets
   - Maximum skew
   - Maximum insertion delay

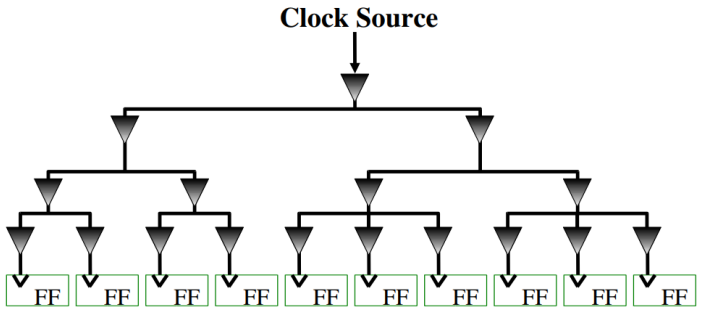


What is clock tree?

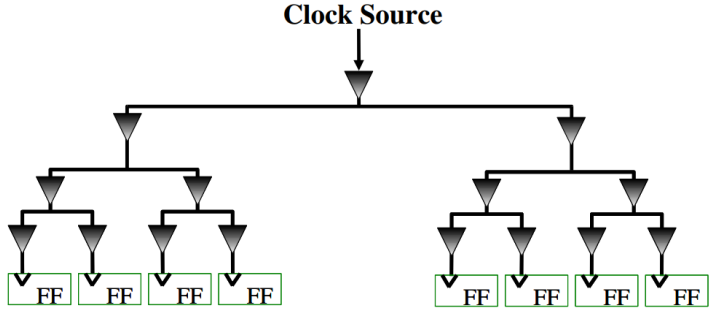
- Driving all the sink pins of a particular clock from the clock port is known as Clock tree for that particular clock.



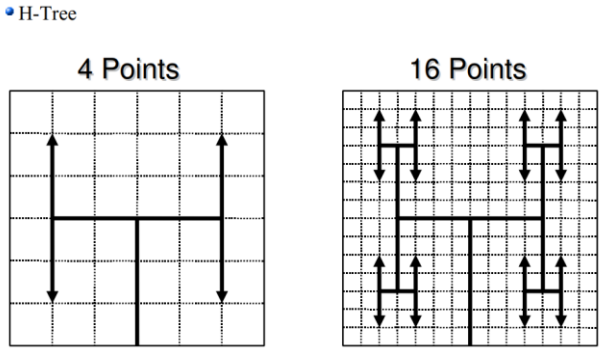
Balanced clock tree



Binary clock tree



H - clock tree



Key metrics to be checked after CTS/CTS-optimization

1. Legal placement completed – “check_legality”
2. Power and ground nets should be prerouted
3. Acceptable congestion values
4. Acceptable timing values (WNS and TNS for all the path groups in the design)
5. High fanout net synthesis is done (Nets :- reset, scan enable) . Clock network shouldn’t be buffered
6. Acceptable DRV’s

Possible ways to improve the timing QOR in cts stage in ICC

1. Cells are sitting far apart in the hierarchy and wire length is high, timing QOR can be improved by placing them together by creating the bounds

2. Assigning separate path group for the violating hierarchy (command :- group_path)

3. Assigning different weight value to different path groups depending upon the timing QOR degradations (group_path –weight)

4. Try setting the high timing effort app option and execute the “place_opt” command.

5. Incrementally running the logic optimization through (psynopt)

Tools used for placement

1 ICC2 – Synopsys

2. EDI and Innovus (CCOPT) – Cadence

3. Aprisa – Avatar integrated systems

CTS related lab exercises

1. Create different blockages and execute “create_placement” & “legalize_placement” and understand the differences in standard cell placement

2. Explore different placement GUI features

3. Create different types of “bounds” and understand the impact on “timinig” QOR and standard cell placement

4. Understand the different congestion methods in ICC

5. Explore different GUI maps need to be analyzed to decide placement quality

6. Understand the placement utilization (“report_placement_utilization”)

7. Understand the timing QOR at placement stage


THANKS