IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

NPTEL

Lecture 27: CLOCK NETWORK SYNTHESIS (PART 1)

PROF. INDRANIL SENGUPTA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Problem Formulation

- Specialized algorithms are required for clock (and power nets) due to strict specifications for routing such nets.
  - Better to develop specialized routers for these nets.
  - Do not over-complicate the general router.
  - In many designs, both these nets are manually routed.
- Sophisticated and accurate clock routing tools are a must for high-performance designs.

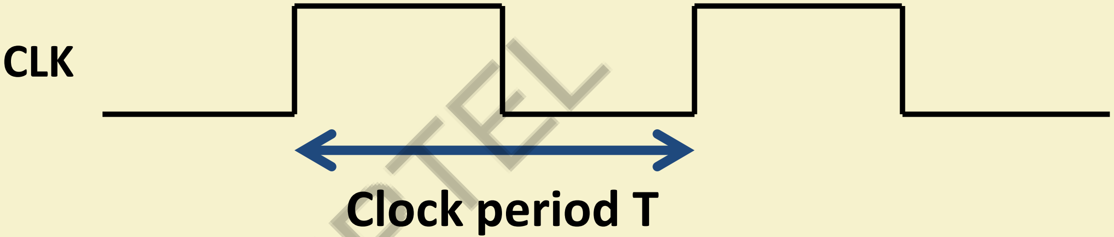
Clock Routing

- Clock synchronization is one of the most critical issues in the design of high-performance VLSI circuits.
  - Data transfer between functional elements is synchronized by the clock.
  - It is desirable to design a circuit with the fastest possible clock.
- The clock signal is typically generated external to the chip.
  - Provided to the chip through ***clock pin***.

- Each functional unit which needs the clock is connected to clock pin by the ***clock net***.
- Ideally, the clock must arrive at all the functional units precisely at the same time.
- In practice, clock skew exists.
  - Maximum difference in the arrival time of a clock at two different components.
  - Forces the designer to be conservative.
    - Use a larger time period between clock pulses, i.e. lower clock frequency.

Clocking Schemes

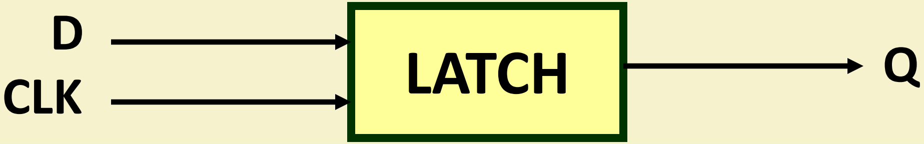
- The clock is a simple pulsating signal alternating between 0 and 1.

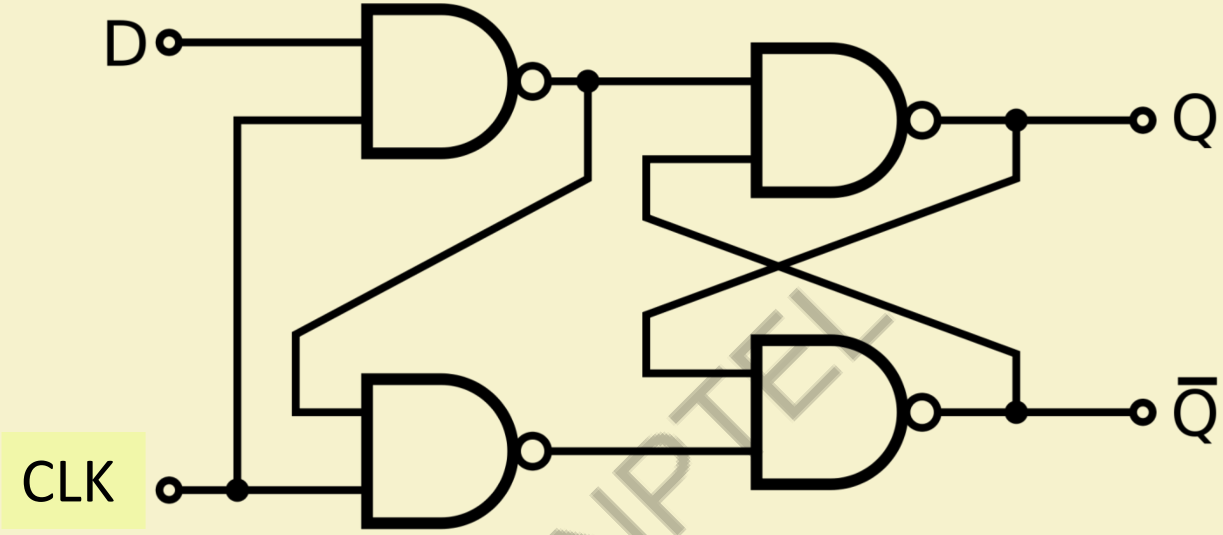


- Digital systems use a number of clocking schemes:
  1. Single-phase clocking with latches
  2. Single-phase clocking with flip-flops
  3. Two-phase clocking

Single-phase Clocking with Latches

- The latch opens when the clock goes high.
- Data are accepted continuously while the clock is high.
- The latch closes when the clock goes down.
- Not commonly used due to their complicated timing requirements.
  - Some high-performance circuits use this scheme.

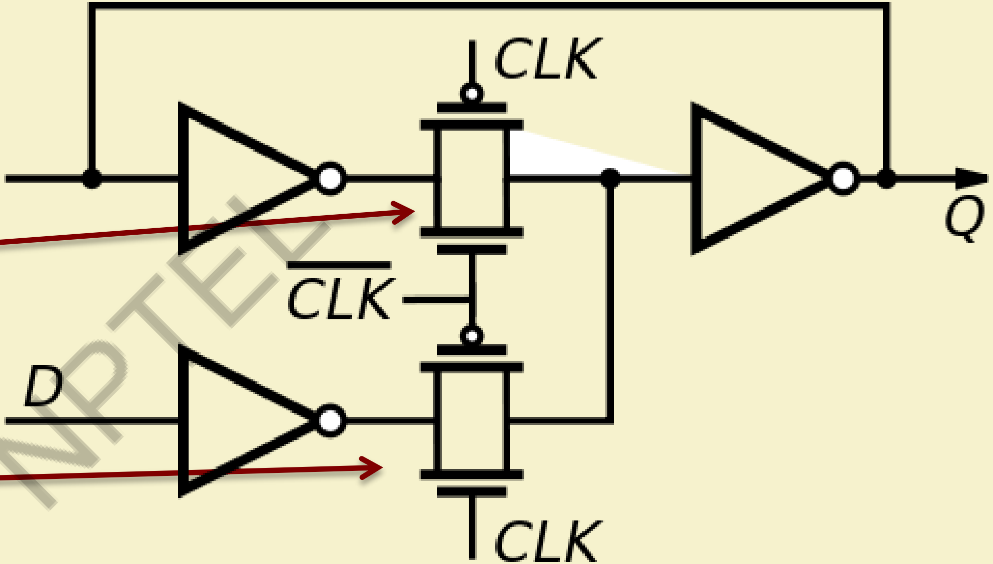




- Latch implementation using NAND gates.
  - As long as CLK is at 1, the value at D gets stored.

- Latches and flip-flops can be implemented in CMOS using inverters and switches.
- In CMOS, a switch can be implemented in two ways:
  - *Pass transistor* that requires a single n-type transistor.
    - Voltage degradation while passing high voltage.
  - *Transmission gate* that uses two back-to-back transistors, one p-type and one n-type.

CMOS Latch

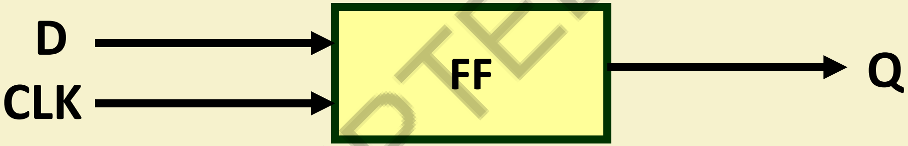


Transmission gate is conducting when CLK=1.

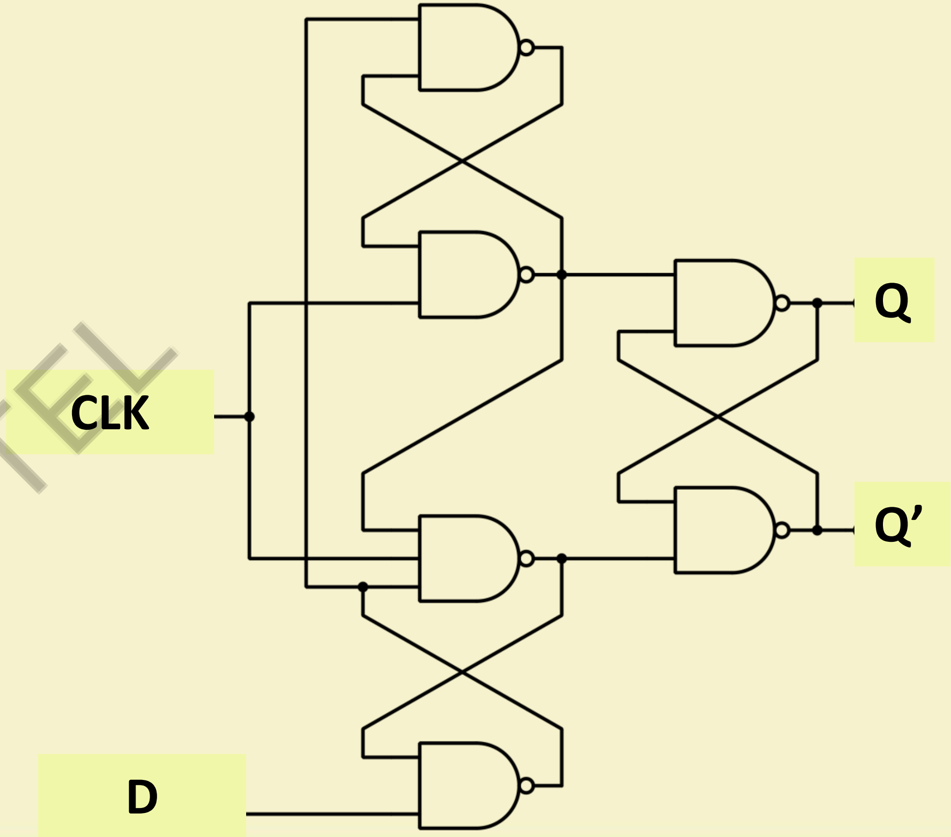
Transmission gate is conducting when CLK=0.

Single-phase Clocking with Flip-flops

- Data are accepted only on the rising or falling edge of the clock.

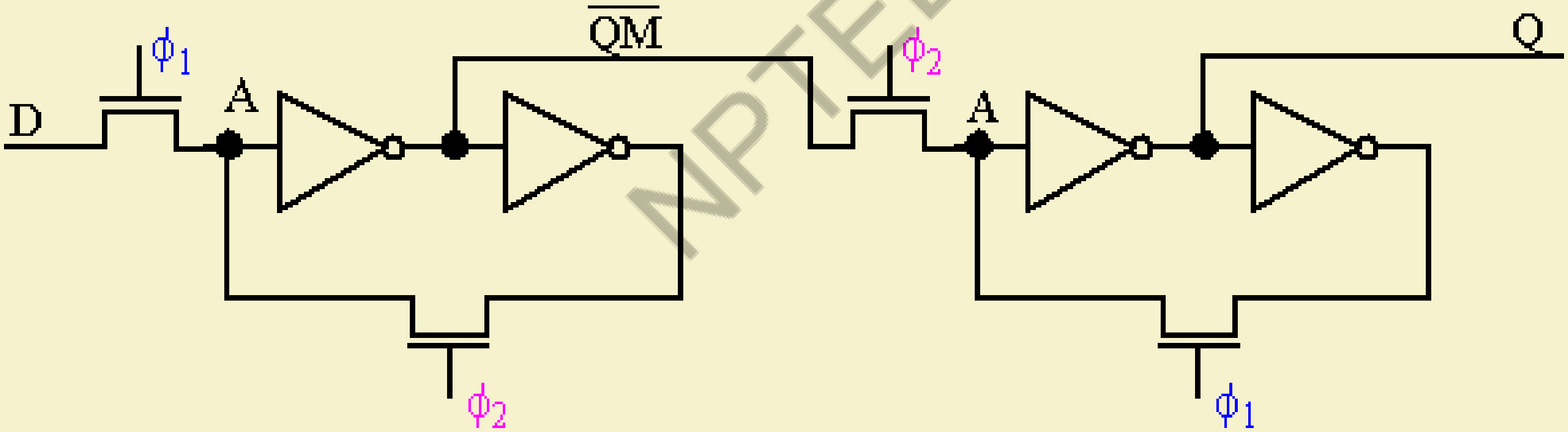


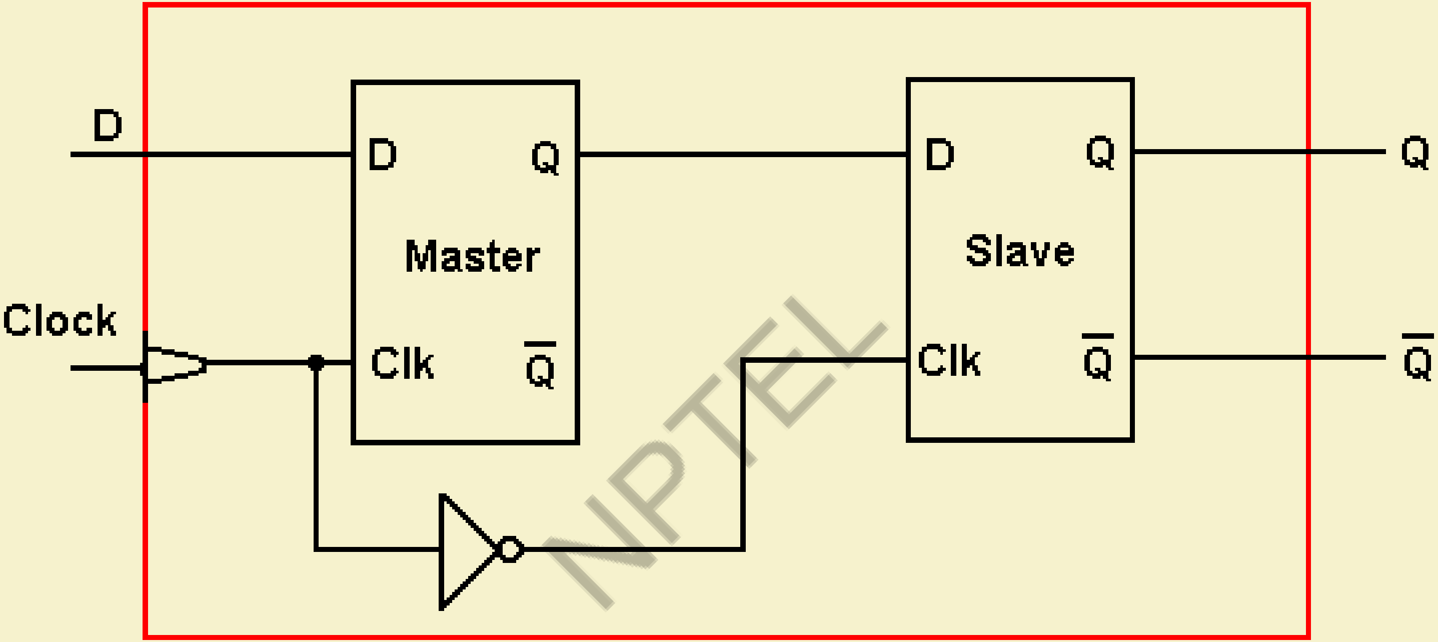
Positive Edge Triggered D Flip-flop



Two-phase Clocking

- Use two latches, one is called the ***master*** and the other the ***slave***.



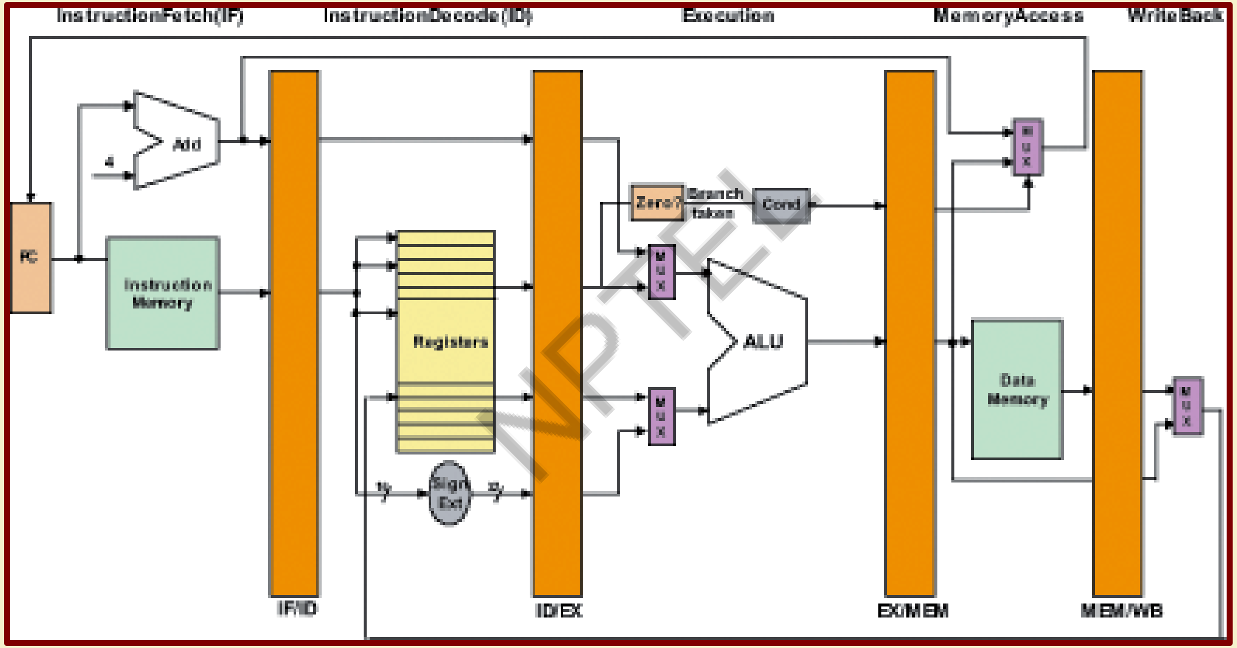


Conventional master-slave flip-flop – can also use two-phase clock

- As a rule of thumb, most systems cannot tolerate a clock skew of more than 10% of the system clock period.
  - A good clock distribution strategy is necessary.
  - Also a requirement for designing high-performance circuits.

Clocking in a Pipeline

- When successive stages are connected in a pipeline, we do not need master-slave flip-flops.
  - Use single-phase latches in the register separating states.
  - Clock alternate latch stages by the two phases Φ_1 and Φ_2 of a two-phase clock.


InstructionFetch(IF)
InstructionDecode(ID)
Execution
MemoryAccess
WriteBack
PC
4
Add
Instruction Memory
IF/ID
Registers
Sign Ext
16
32
ID/EX
Zero?
Branch taken
Cond
MUX
ALU
EX/MEM
Data Memory
MEM/WB

END OF LECTURE 27

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

NPTEL

Lecture 28: CLOCK NETWORK SYNTHESIS (PART 2)

PROF. INDRANIL SENGUPTA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Strategies to reduce clock skew

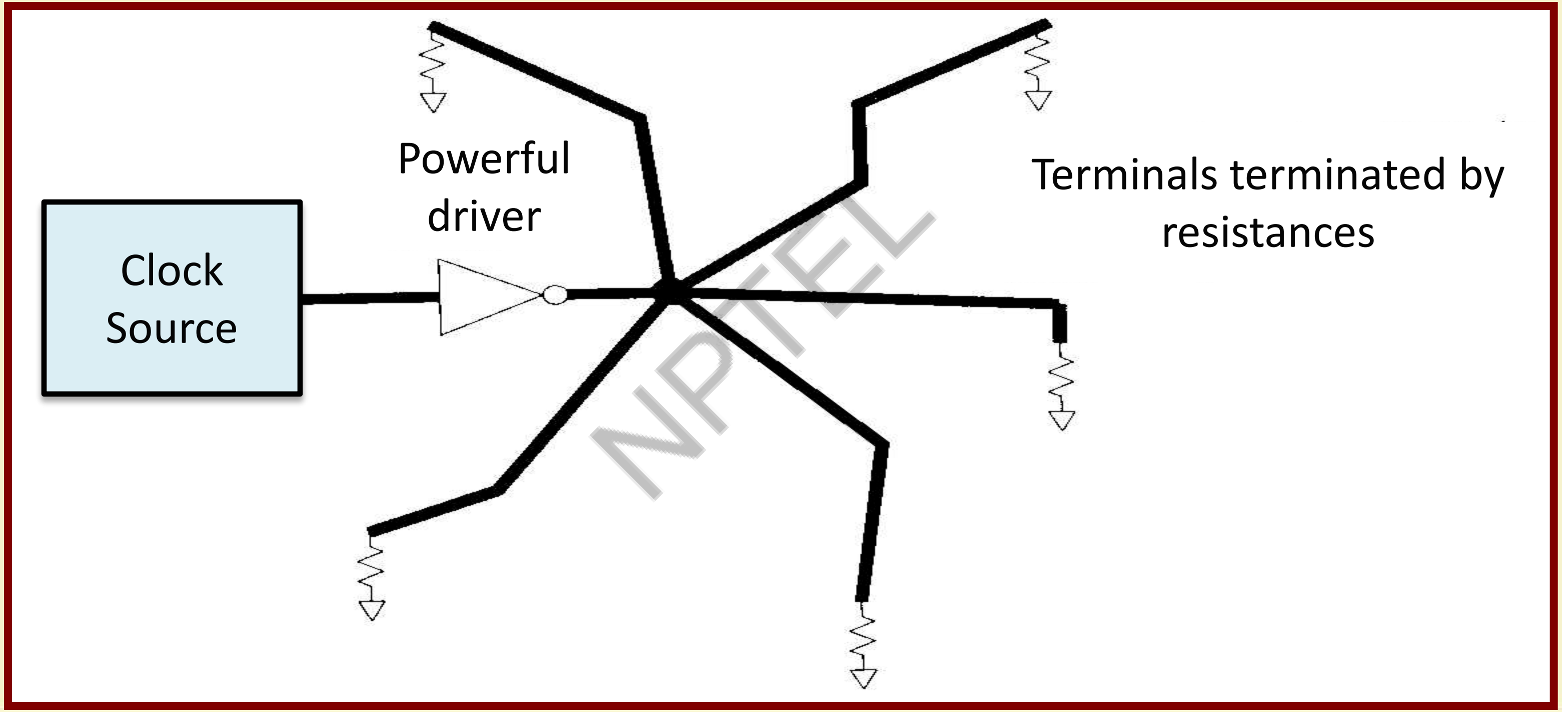
- Two main strategies:
  1. Locate all clock inputs close together; but it is difficult to implement in a large circuit.
  2. Drive them from the same source & balance the delays.
- Due to physical limitation and diverse distribution of clock sinks, strategy 2 is often used.

How to Realize Strategy 2 ?

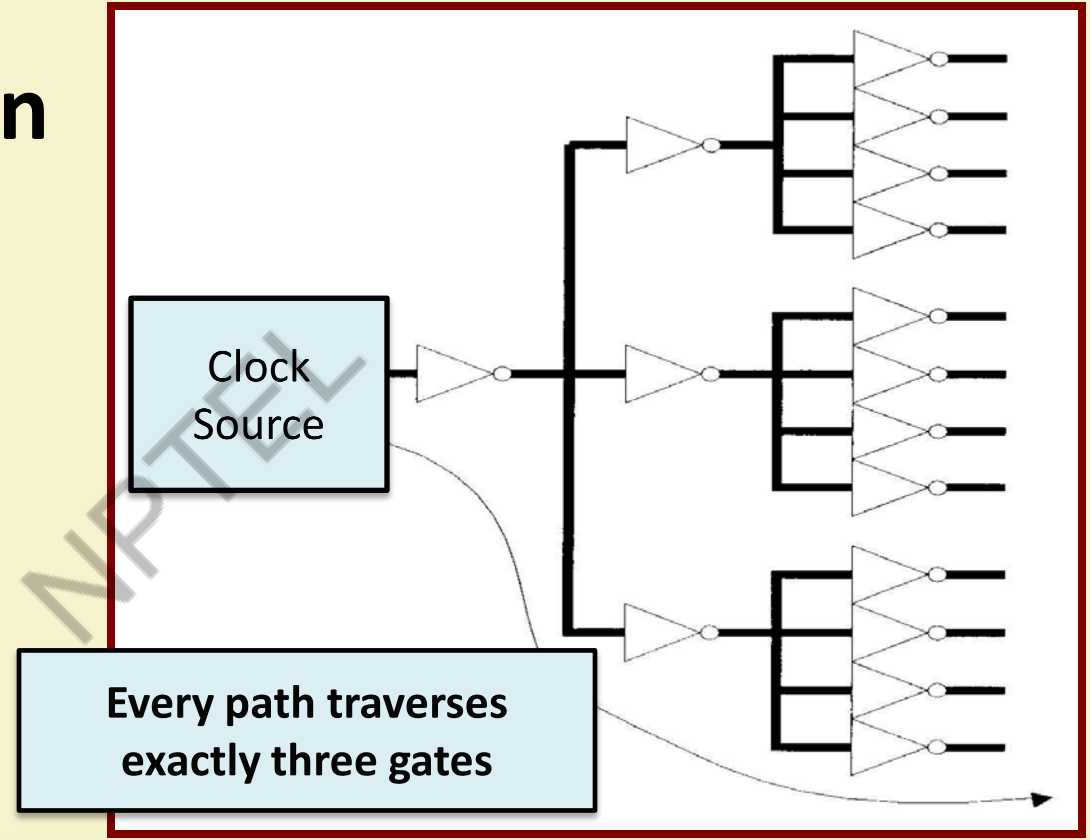
1. Spider-leg distribution network
   - Use a power driver to drive N outputs.
   - A separate wire goes to each destination.
   - Use load (R) termination to reduce reflection if the traces are long (distributed circuit). Total load = R/N.
   - For example, if line impedance=75 Ω and N=3, total load=25Ω.
   - Two or more drivers may need to be connected in parallel.

2. Clock distribution tree

Spider leg Distribution Network



Clock distribution tree

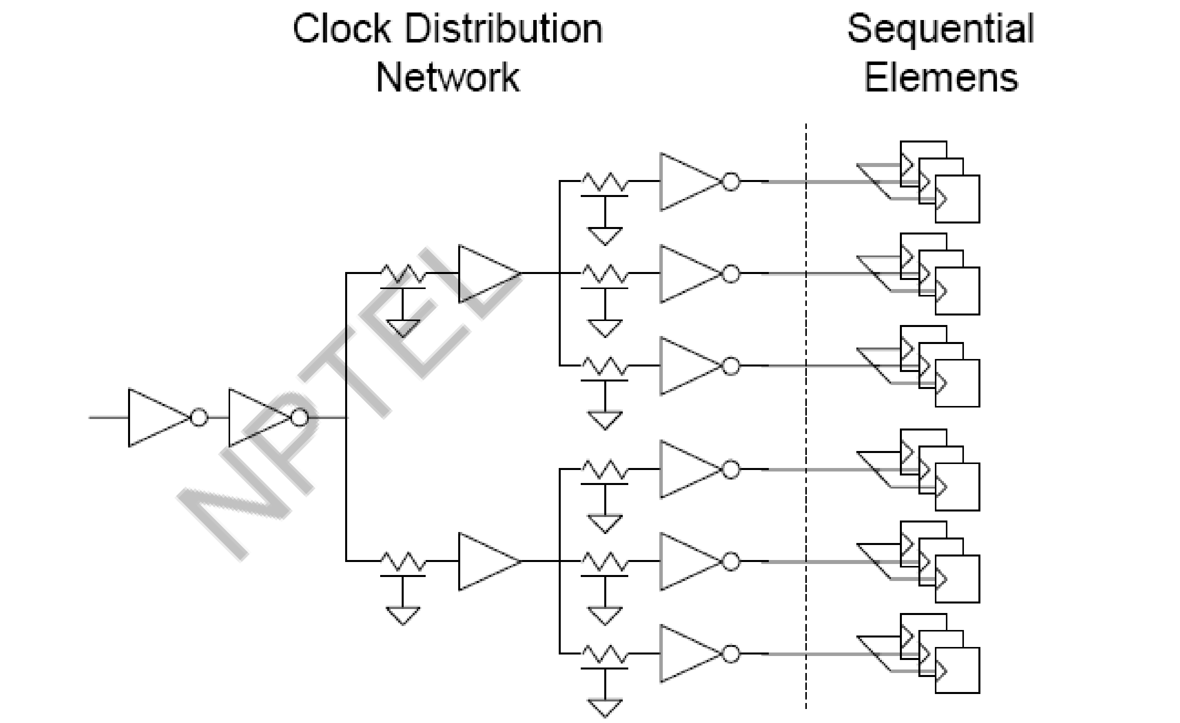


Every path traverses exactly three gates

Clock Buffering Mechanisms

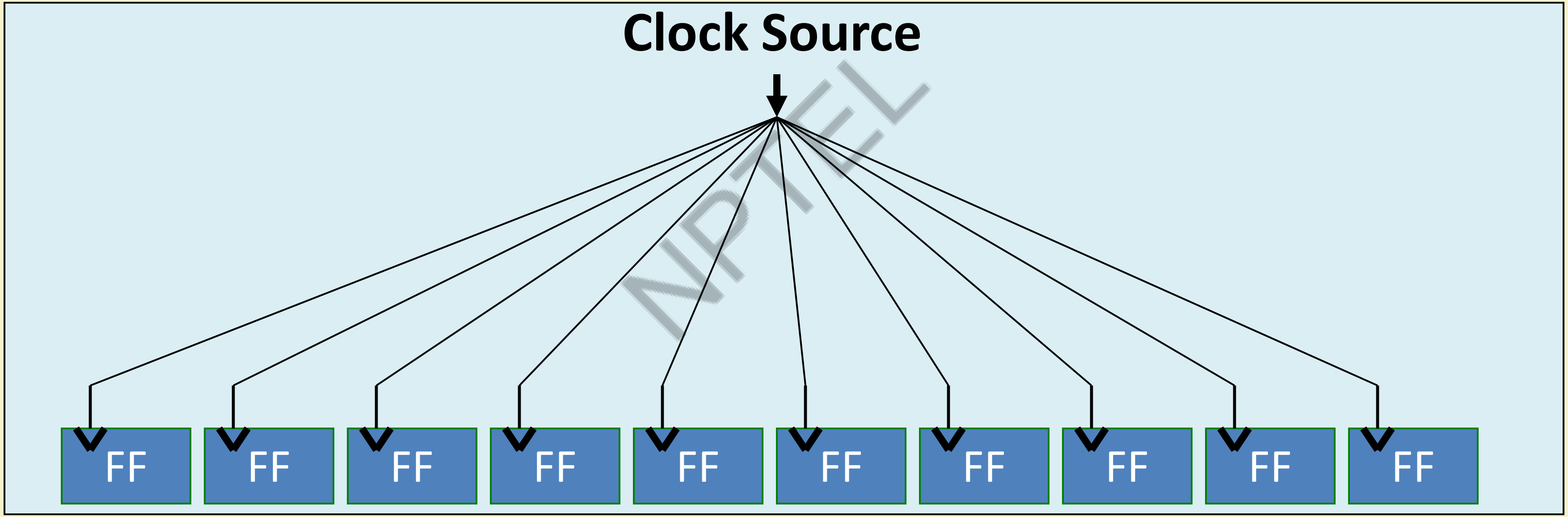
- Clock signal is global in nature.
  - Clock lines are typically very long.
  - Long wires have large capacitances, which limit the performance of the system.
  - RC delay plays a big factor.
- RC delay cannot be reduced by making the wires wider.
  - Resistance reduces, but capacitance increases.
- To reduce RC delay, buffers are used.
  - Also helps to preserve the clock waveform.
  - Significantly reduces the delay.
  - May occupy as much as 5% of the total chip area.
  - Isolate the clock net from upstream load impedances.

Use of Buffers

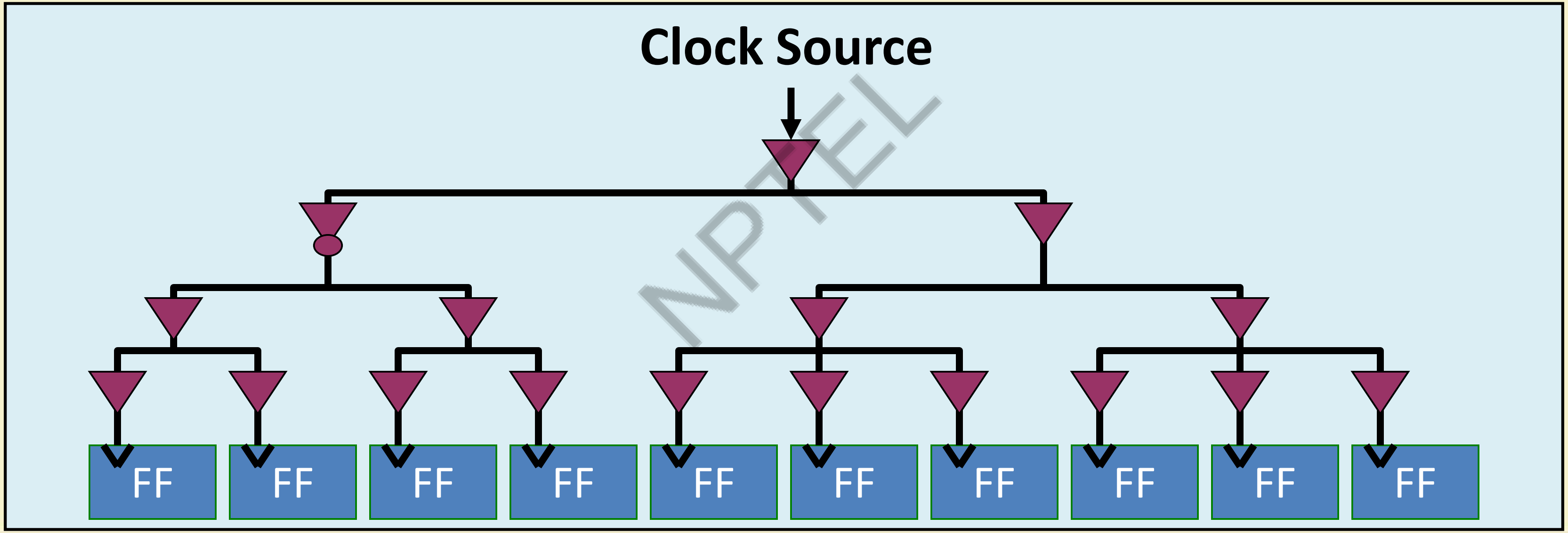


Clock tree :: to summarize

- A path from the clock source to clock sinks.

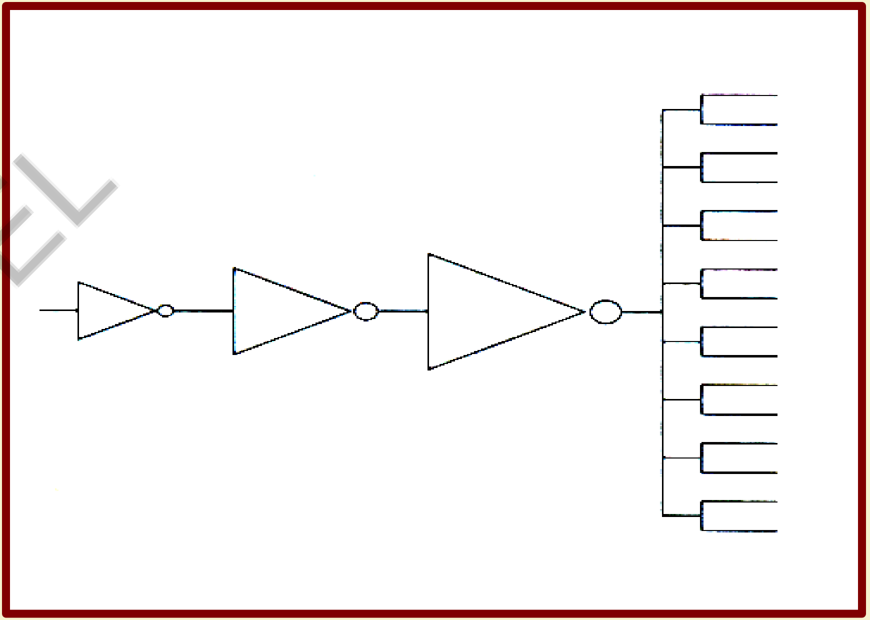


- Buffering restores the signal and reduces delay, and thus helps to guarantee the integrity of the clock signal.



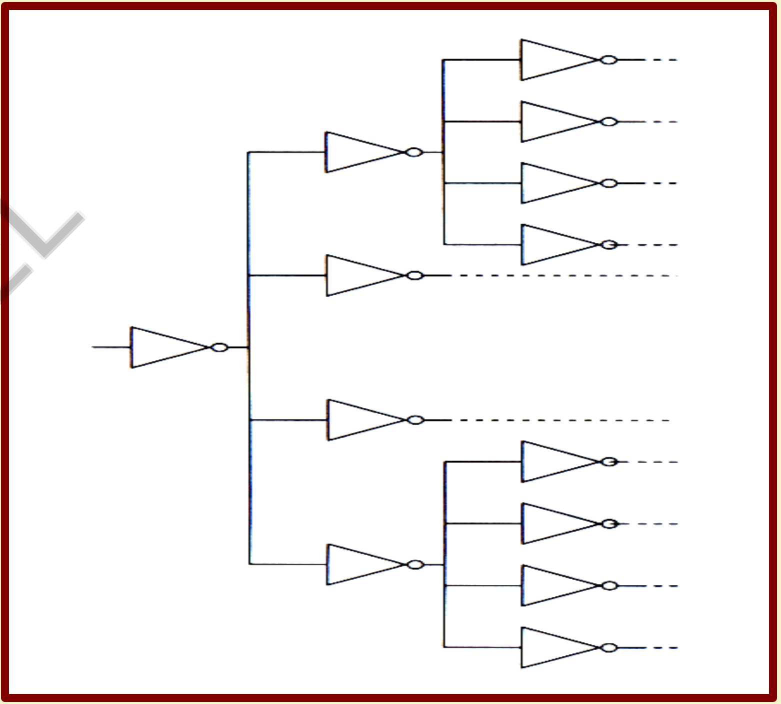
Clock Buffering :: Approach 1

- Use a big, centralized buffer.
  - Better from skew minimization point of view.
  - Only need to concentrate on equalizing the wire lengths of the tree.

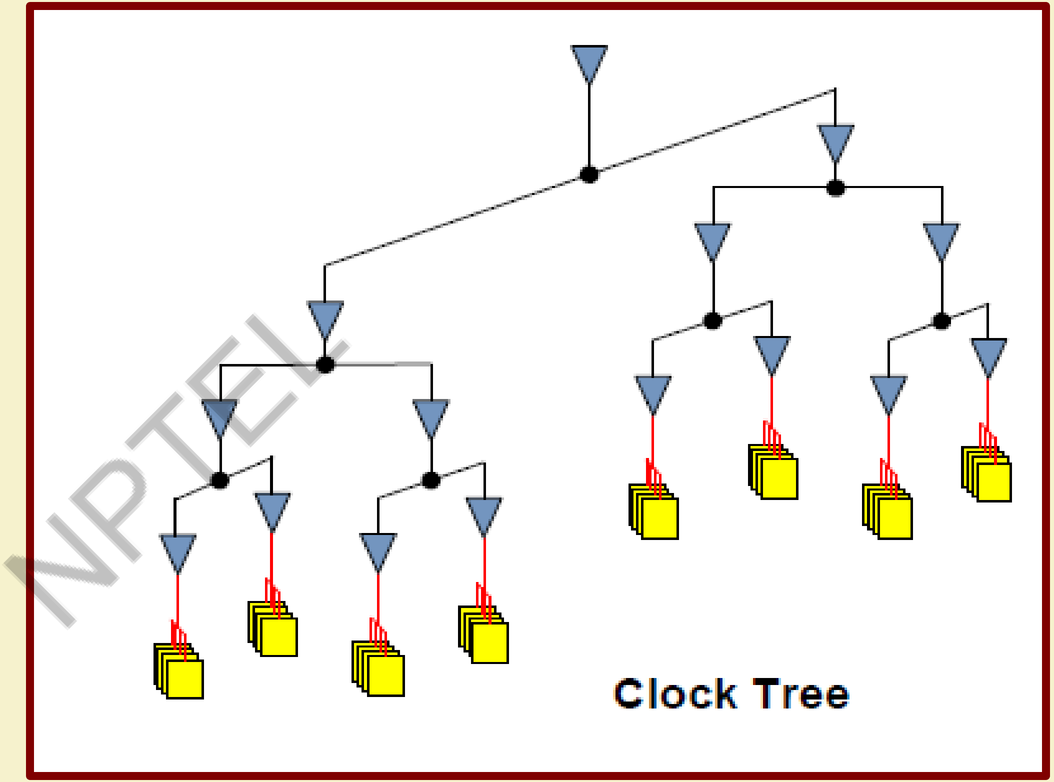


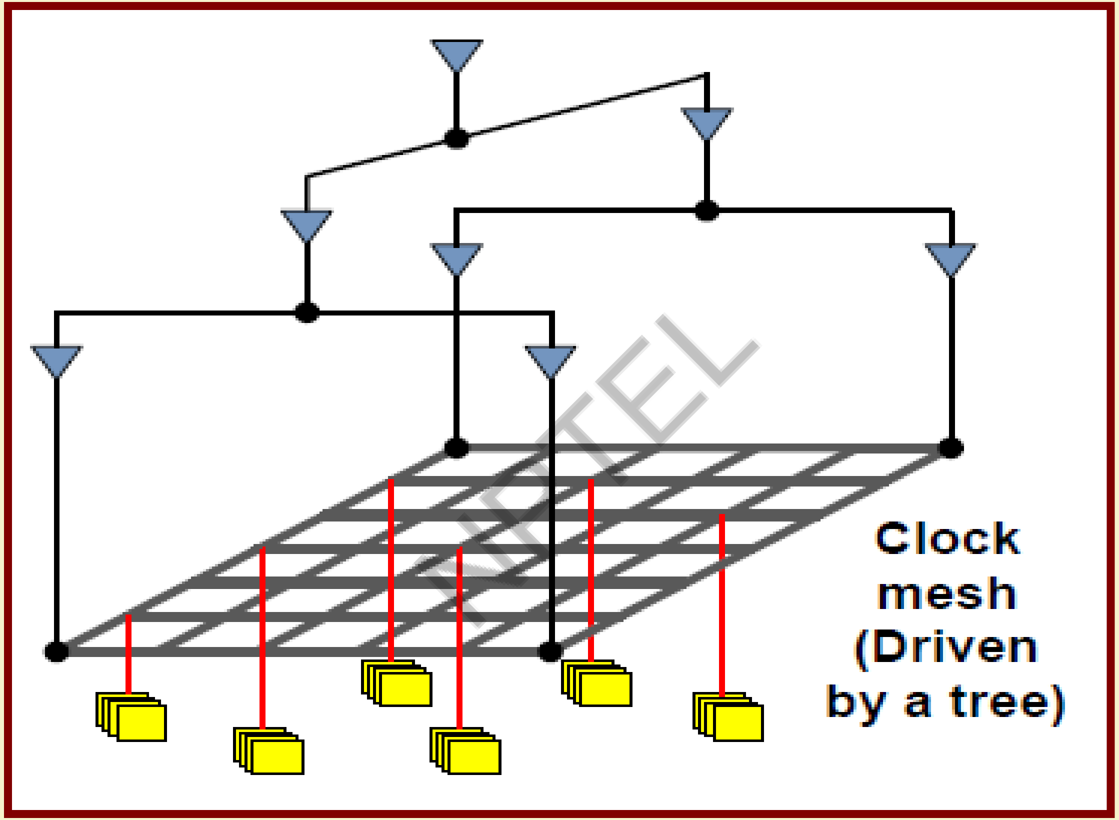
Clock Buffering:: Approach 2

- Distribute buffers in the branches of the clock tree.
  - Use identical buffers so that the delay introduced by the buffers is equal in all branches.
- Regular layout of the clock tree, and equalization of the buffer loads help to reduce clock skew.



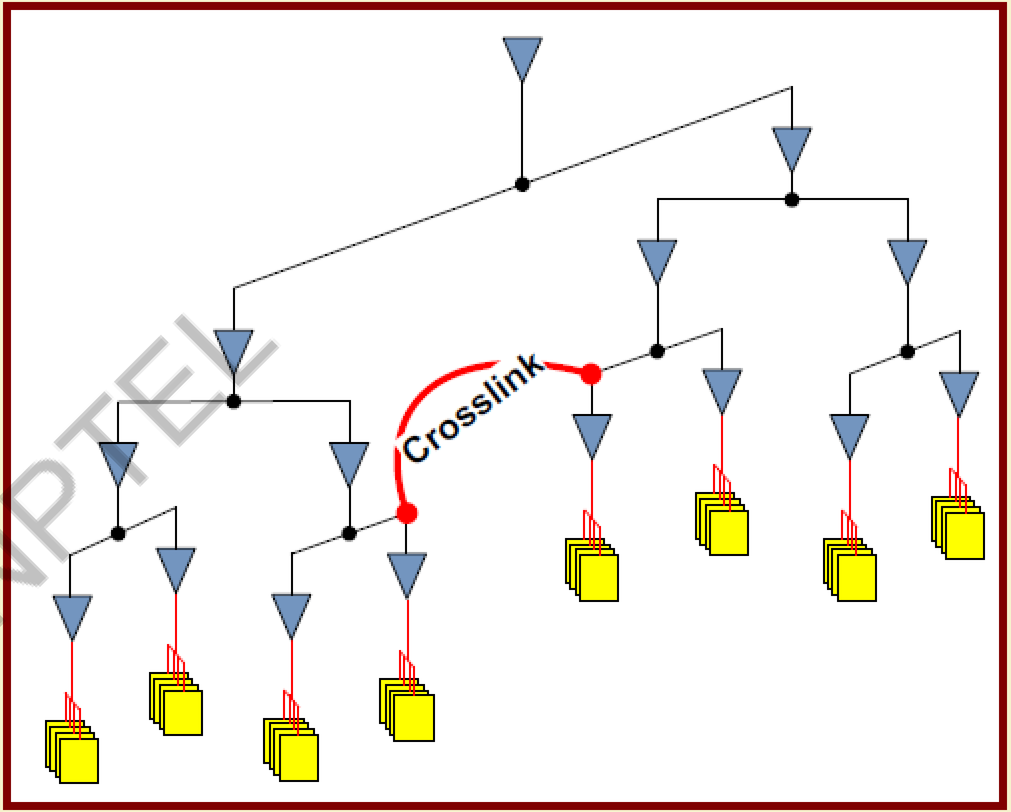
Broad Topologies



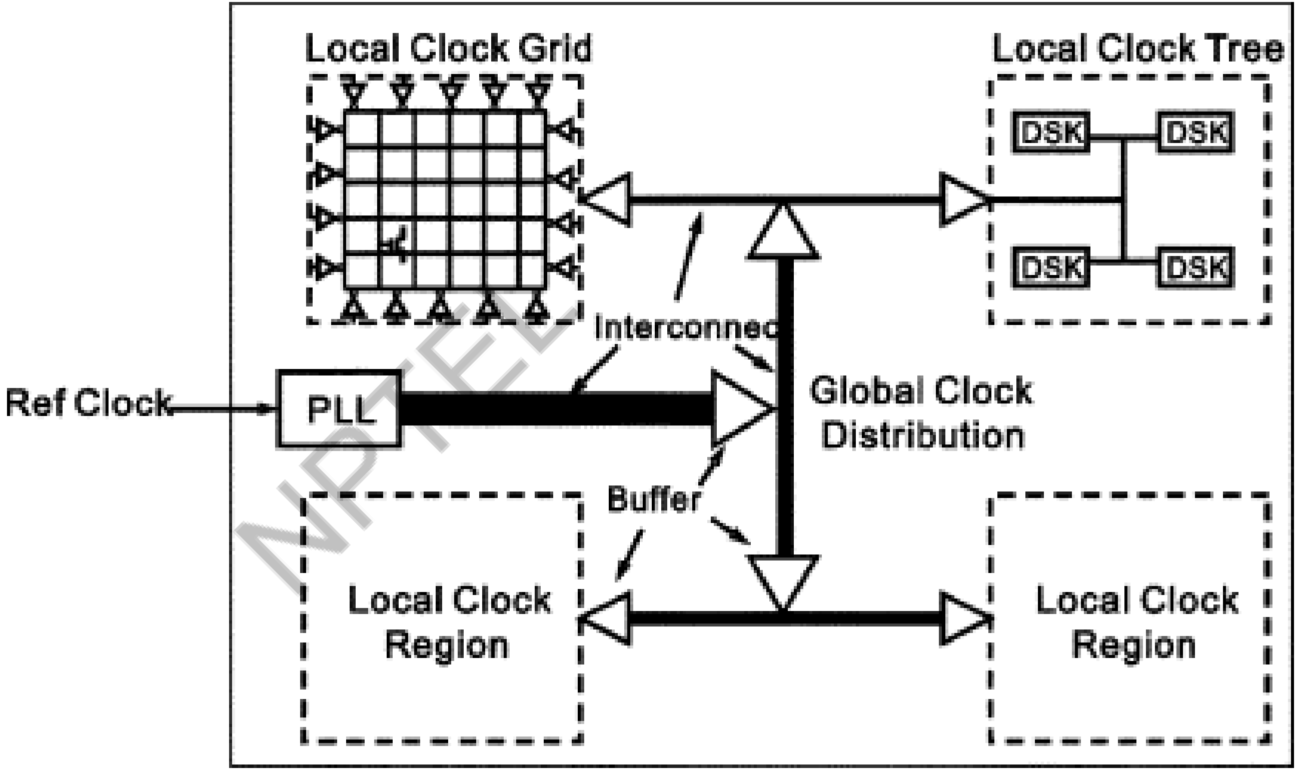

Clock
mesh
(Driven
by a tree)

Binary Tree with Crosslinks

- A specific implementation of a binary tree.
- Cross-links are inserted at specific points along the tree to equalize clock latency.



Combination of Topologies



END OF LECTURE 28

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

NPTEL

Lecture 29: CLOCK NETWORK SYNTHESIS (PART 3)

PROF. INDRANIL SENGUPTA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

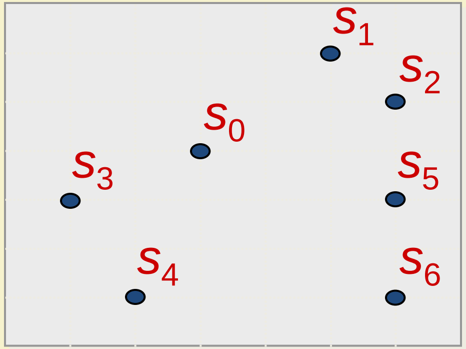
Terminology

- A clock routing instance (clock net) is represented by n+1 terminals, where s_0 is designated as the source, and $S = \{s_1, s_2, \dots, s_n\}$ is designated as sinks
  - Let s_i, $0 \le i \le n$, denote both a terminal and its location.

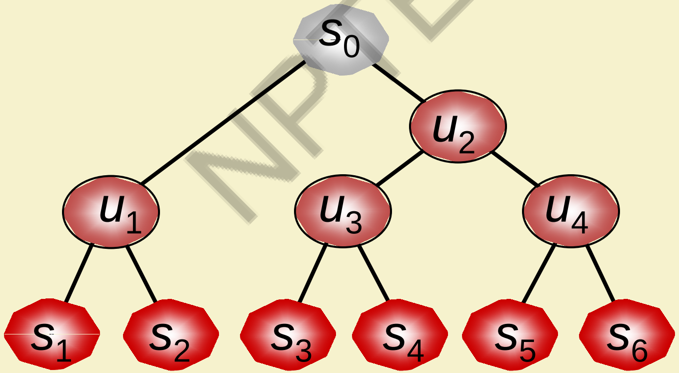
- A clock routing solution consists of a set of wire segments that connect all terminals of the clock net, so that a signal generated at the source propagates to all of the sinks.
  - Two aspects of clock routing solution: ***topology*** and ***geometric embedding***.

- The clock-tree topology (clock tree) is a rooted binary tree *G* with *n* leaves corresponding to the set of sinks.
  - Internal nodes = Steiner points

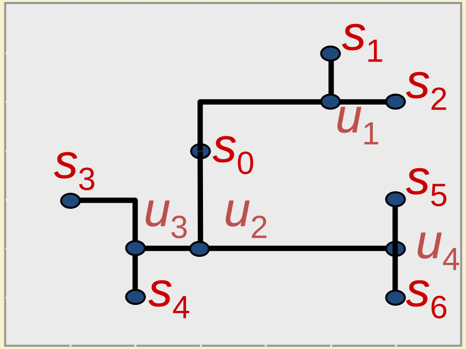
Clock routing problem instance



Connection topology



Embedding



Terminology

- Clock skew: (maximum) difference in clock signal arrival times between sinks.

$$skew(T) = \max_{s_i, s_j \in S} | t(s_0, s_i) - t(s_0, s_j) |$$

- Local skew: maximum difference in arrival times of the clock signal at the clock pins of two or more related sinks.
  - Sinks within distance *d* .
  - Flip-flops or latches connected by a directed signal path.

- Global skew: maximum difference in arrival times of the clock signal at the clock pins of any two (related or unrelated) sinks.
  - Difference between shortest and longest source-sink path delays in the clock distribution network.
  - The term “skew” typically refers to “global skew”.

Terminologies for Clock-Tree Routing

- Zero skew: zero-skew tree (ZST)
  - ZST problem
- Bounded skew: true ZST may not be necessary in practice
  - Signoff timing analysis is sufficient with a non-zero skew bound.
  - In addition to final (signoff) timing, this relaxation can be useful with intermediate delay models when it facilitates reductions in the length of the tree.
  - Bounded-Skew Tree (BST) problem.

- Useful skew: correct chip timing only requires control of the local skews between pairs of interconnected flip-flops or latches.
    - Useful skew formulation is based on analysis of local skew constraints.

Modern Clock Tree Synthesis

- Basic requirements:
  - Constructing trees with Zero Global Skew
  - Clock Tree Buffering in the presence of variation
- A clock tree should have low skew, while delivering the same signal to every sequential block.

- Clock tree synthesis is performed in two steps:

  a) Initial tree construction with one of these scenarios.

  - Construct a regular clock tree, largely independent of sink locations
  - Simultaneously determine a topology and an embedding
  - Construct only the embedding, given a clock-tree topology as input

  b) Clock buffer insertion and several subsequent skew optimizations.

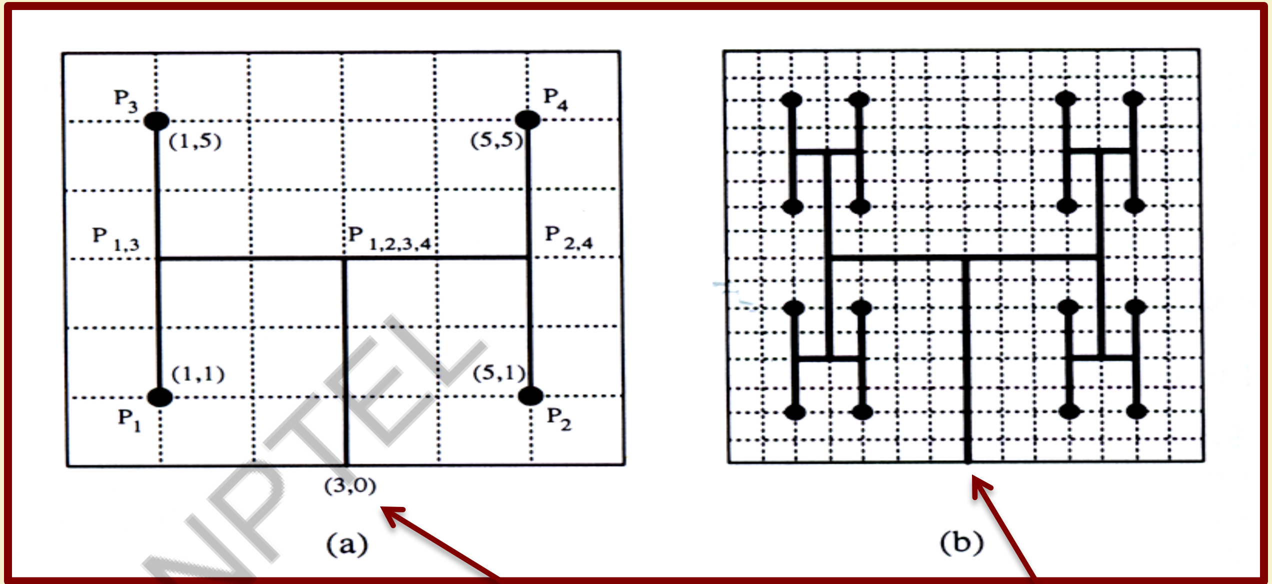
Clock Routing Algorithms

- How to minimize skew?
  - Distribute the clock signal in such a way that the interconnections carrying the clock signal to functional sub-blocks are equal in length.
- Several algorithms exist which try to achieve this goal.
  - H-tree based algorithm
  - X-tree based algorithm
  - Method of Means and Medians algorithm
  - Recursive Geometric Matching algorithm
  - Zero clock skew routing

H-tree based Algorithm

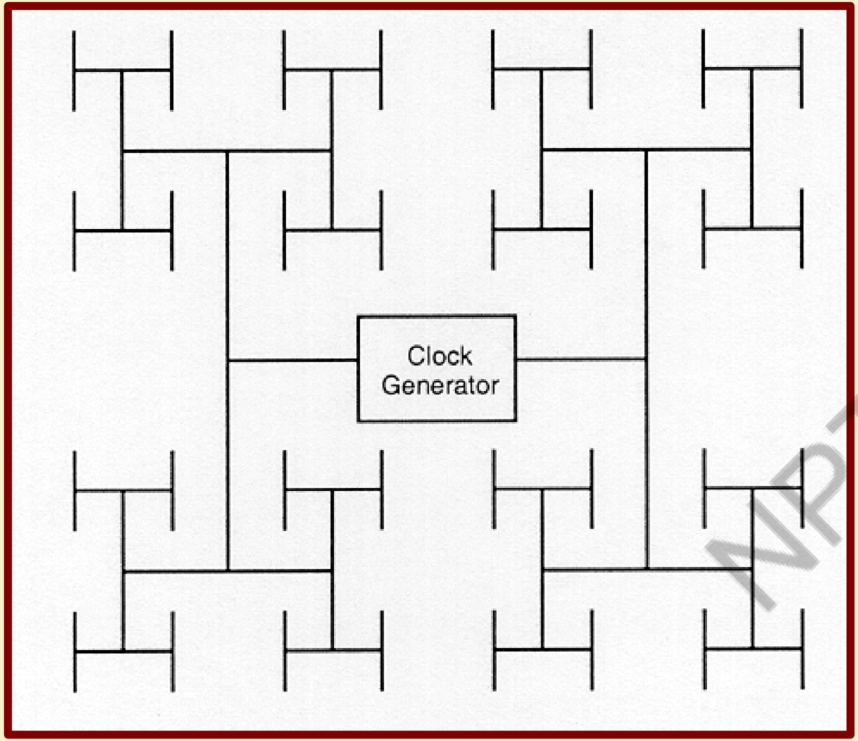
- An early approach, which is based on equalization of wire lengths.
- In H-tree based approach, the distance from clock source to each of the clock sinks is the same.
- Suitable for scenarios where all clock terminals are arranged in a symmetrical fashion, as in gate arrays or FPGAs.
  - Can also be used to carry the clock signal to various regions or zones of the chip.

- In (a), all points are exactly 7 units from the point P_0, and hence the skew is zero.
- This ensures minimum-delay routing as well.
  - P_0 and P_3 are at a distance 7 (rectilinear distance).
- Can be generalized to 4^m points, where m is an integer.

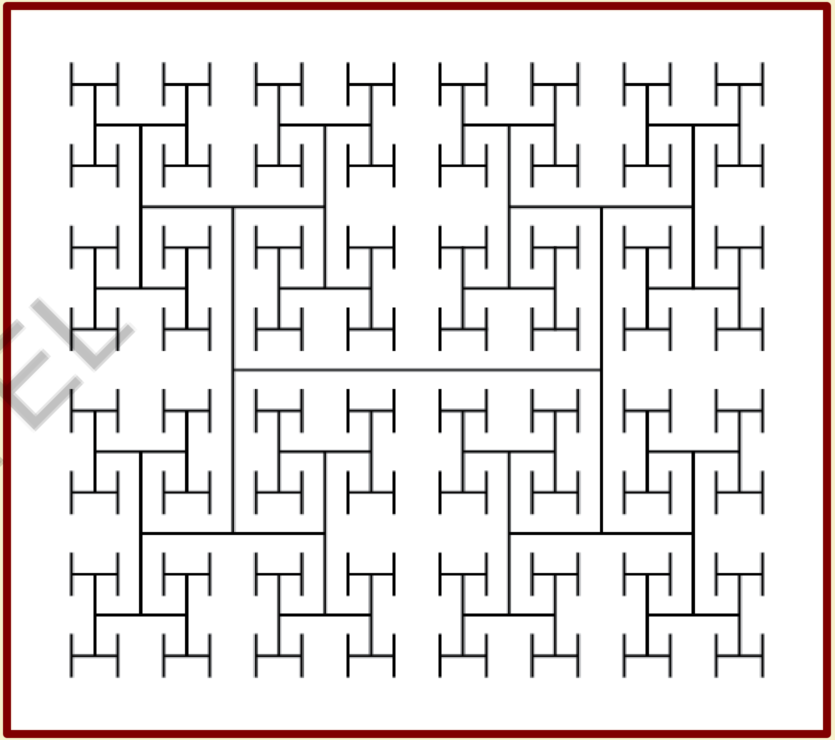


P_0 (distance = 7)

P_0 (distance = 19)



64 points

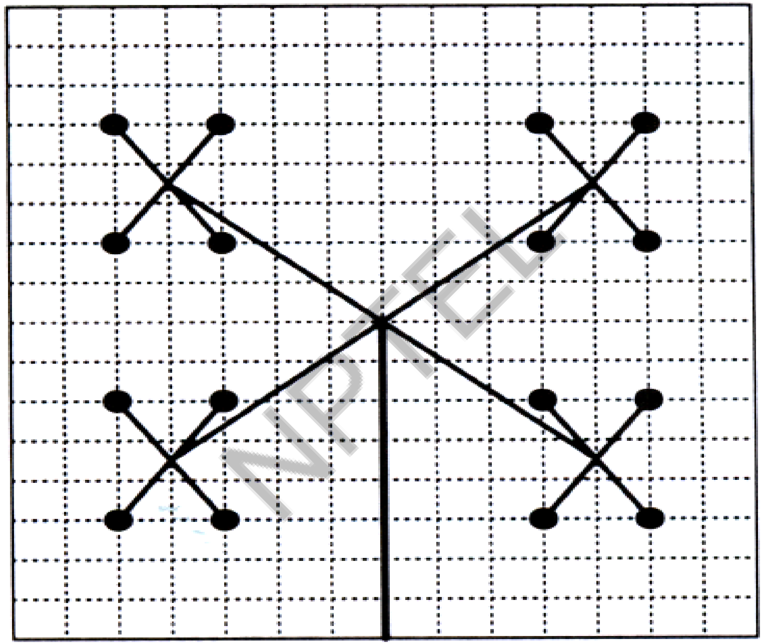


256 points

- Exact zero skew due to the symmetry of the H tree.
- Typically used for top-level clock distribution, not for the entire clock tree.
  - Blockages can spoil the symmetry of a H tree.
  - Non-uniform sink location and varying sink capacitances also complicate the design of H trees.

X-tree based Algorithm

- An alternate tree structure with a smaller delay.
  - Assuming non-rectilinear routing is possible.
- Although apparently better than H-trees, this may cause crosstalk due to close proximity of wires.
- Like H-trees, this is also applicable for very special structures.
  - Not applicable in general.



END OF LECTURE 29

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

NPTEL

Lecture 30: CLOCK NETWORK SYNTHESIS (PART 4)

PROF. INDRANIL SENGUPTA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

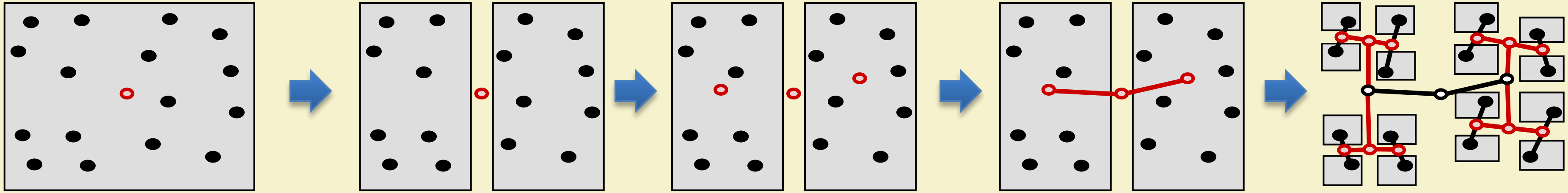
Method of Means & Medians (MMM)

- Follows a strategy very similar to the H-tree algorithm.
  - Can deal with arbitrary locations of clock sinks.
- Basic idea:
  - Recursively partition the set of terminals into two subsets of equal size (***median***).
  - Connects the center of mass of the whole set to the centers of masses of the two partitioned subsets (***mean***).

How is the partitioning done?

- Let L_x denote the list of clock points sorted according to their x-coordinates.
- Let P_x be the median in L_x.
  - Assign points in list to the left of P_x to P_L.
  - Assign the remaining points to P_R.
- Next, we go for a horizontal partition, where we partition a set of points into two sets P_B and P_T.
- This process is repeated iteratively.

- The basic algorithm ignores the blockages and produces a non-rectilinear tree. Some wires may also intersect.
  - In the second phase, each wire can be converted so that it consists only of rectilinear segments and avoids blockages.



Find the center of mass (x-median)

Partition *S* by the median

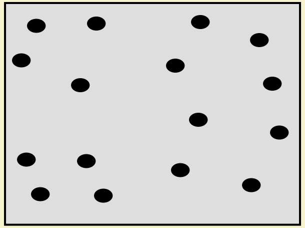
Find the center of mass for the left and right subsets of *S* (y-median)

Connect the center of mass of *S* with the centers of mass of the left and right subsets

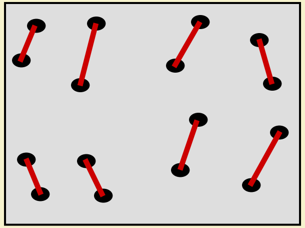
Final result after recursively performing MMM on each subset

Recursive Geometric Matching (RGM)

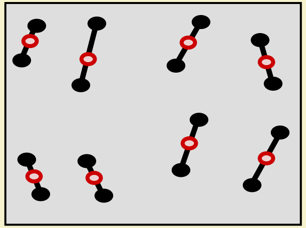
- RGM proceeds in a bottom-up fashion.
  - Compare to MMM, which is a top-down algorithm.
- Basic idea:
  - Recursively determine a minimum-cost geometric ***matching*** of *n* sinks.
  - Find a set of *n*/2 line segments that match *n* endpoints and minimize total length (subject to the matching constraint).
  - After each matching step, a balance or tapping point is found on each matching segment to preserve zero skew to the associated sinks.
  - The set of *n*/2 tapping points then forms the input to the next matching step.



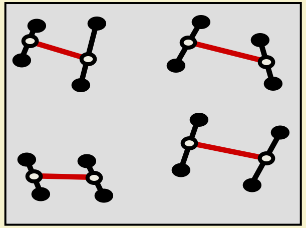
Set of *n* sinks *S*



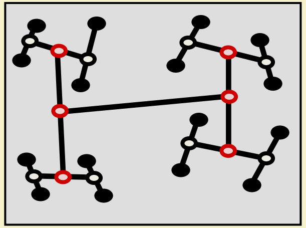
Min-cost geometric matching



Find balance or tapping points (point that achieves zero skew in the subtree, not always midpoint)



Min-cost geometric matching

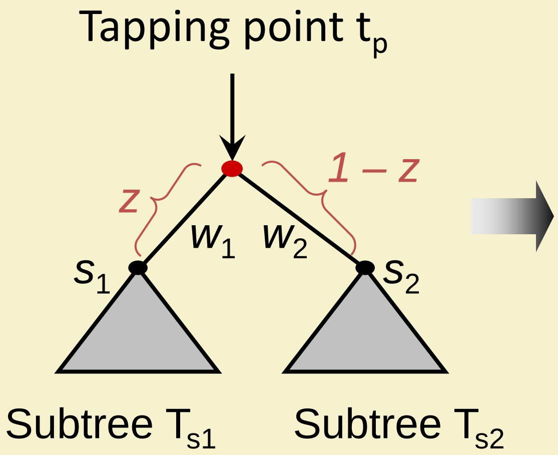


Final result after recursively performing RGM on each subset

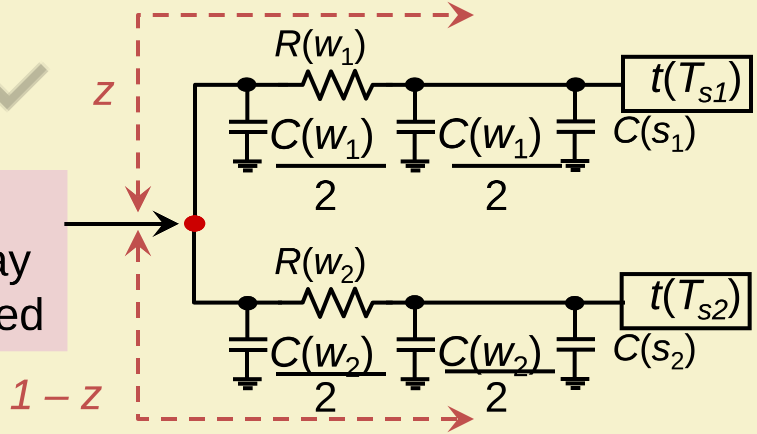
Zero Skew Clock Routing

- Based on the Elmore delay model.
  - Delay along an edge is proportional to its length.
  - However, the delay along a path is defined recursively.
- Adopts a bottom-up process of matching subtree roots and merging the corresponding subtrees, similar to RGM.
- Two important improvements:
  - Finds exact zero-skew tapping points with respect to the Elmore delay model.
  - Maintains exact delay balance even when two subtrees with very different source-sink delays are matched (by wire elongation).

- The point set is recursively partitioned into two subsets, and trees are constructed in a bottom-up manner.
  - Assume, inductively, that every sub-tree has achieved zero skew.
  - Given two zero-skew sub-trees, merge them by an edge to achieve zero skew on the new tree.
    - Necessary to decide the position of the connecting points (taps).
    - Uses Elmore delay model for the purpose.



Tapping point t_p, where Elmore delay to sinks is equalized

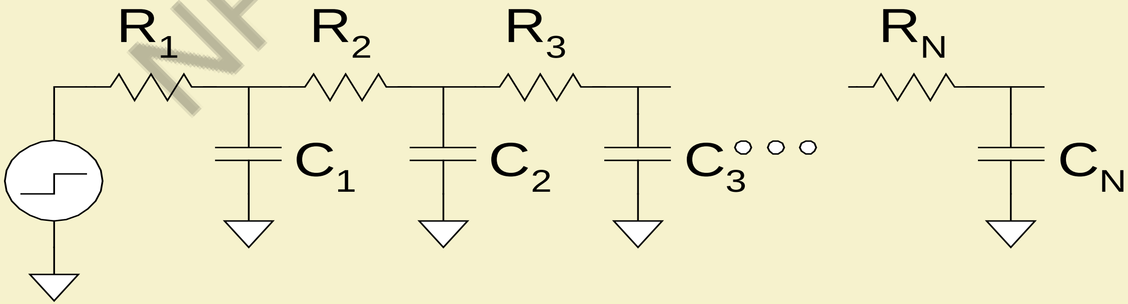


Elmore Delay

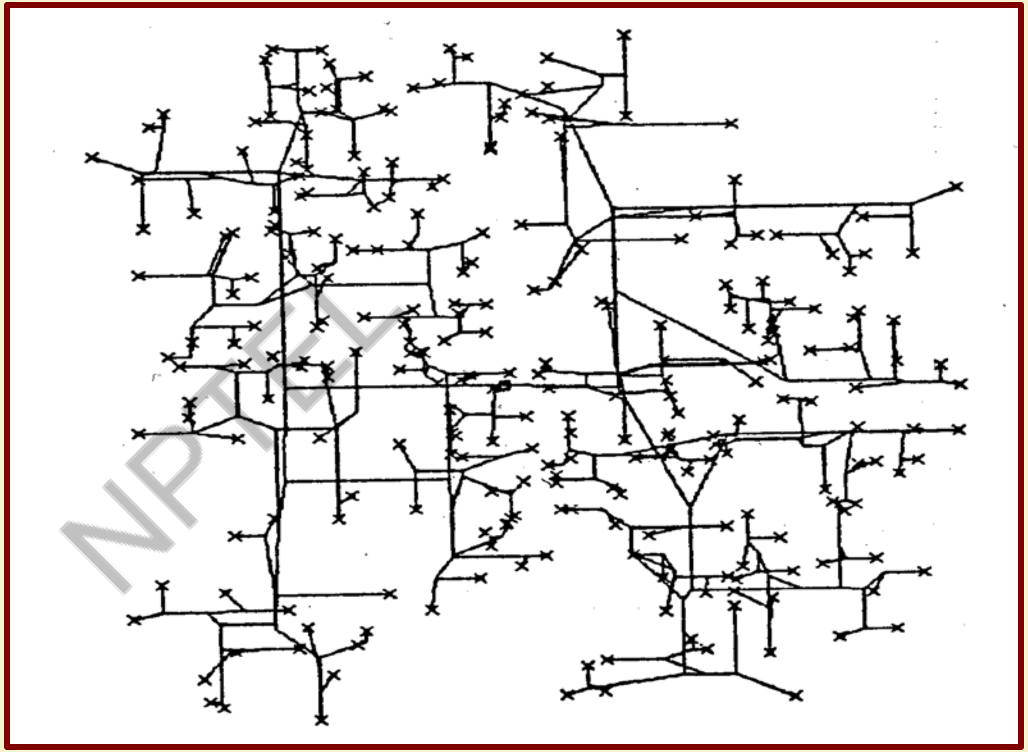
- ON transistors look like resistors.
- Pullup or pulldown network modeled as *RC ladder.*
- Elmore delay of RC ladder is shown.

$$t_{pd} \approx \sum_{\text{nodes } i} R_{i-to-source} C_i$$

$$= R_1 C_1 + \left(R_1 + R_2\right) C_2 + \ldots + \left(R_1 + R_2 + \ldots + R_N\right) C_N$$



A Zero Skew Clock Tree based on Elmore Delay Analysis



Clock Tree Buffering in the Presence of Variation

- To address challenging skew constraints, a clock tree undergoes several optimization steps:
  - Geometric clock tree construction
  - Initial clock buffer insertion
  - Clock buffer sizing
  - Wire sizing
  - Wire snaking

- In the presence of process, voltage, and temperature variations, such optimizations require modeling the impact of variations.
  - Variation model encapsulates the different parameters, such as width and thickness, of each library element.

Case Study :: IBM’s Approach

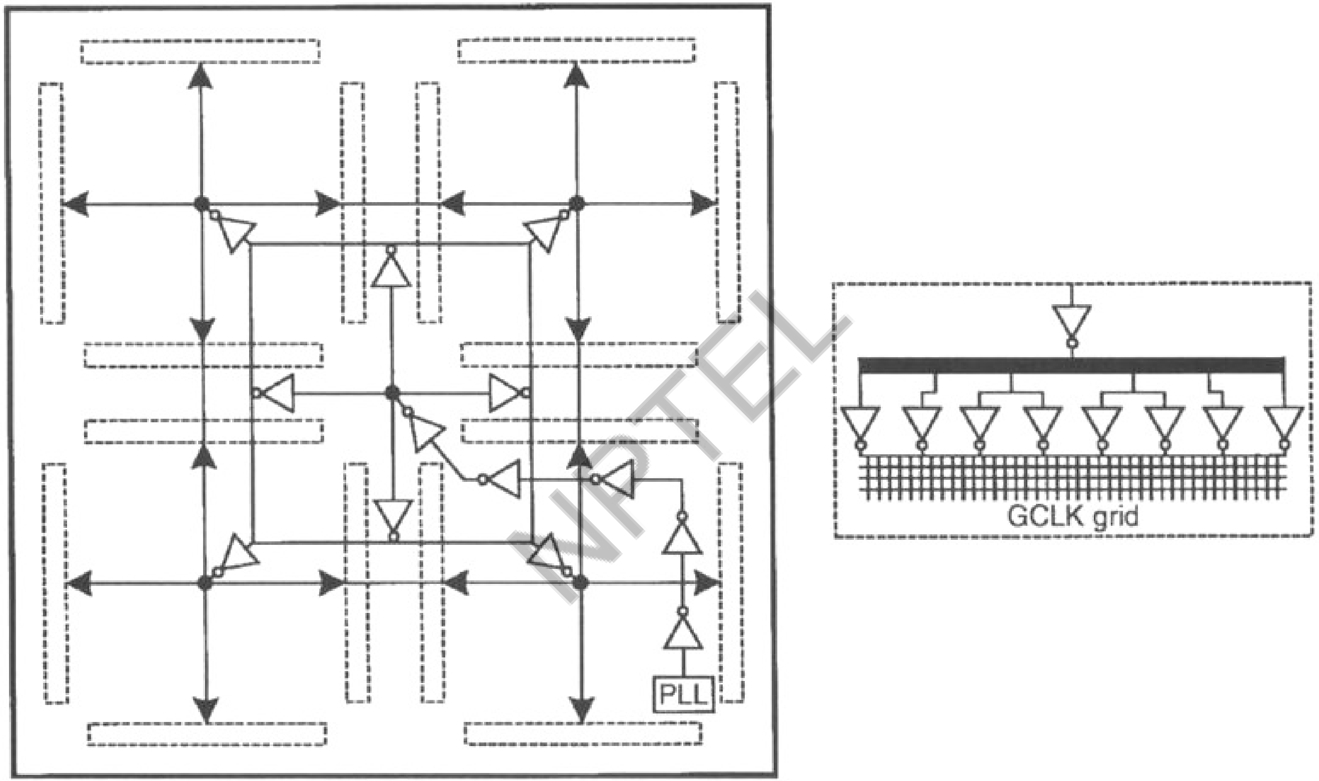
- This concept was applied to a family of IBM microprocessors.
- A central H-tree drives a set of 16 to 64 ***sector buffers***.
- Each buffer drives a ***tunable tree***.
  - Each wire width of this tree is sized.
- Finally, all the tunable trees drive a single grid that provides the clock signal to the entire chip.

- The higher levels of the network consist of trees.
  - Lower latency, lower power, lower area, better global skew.
- The lowest level consists of a regular grid.
  - Constant structure so that the clock can be distributed anywhere.
  - The regular grid allows the higher levels of the tree to be regular.
  - Better local skew.

- Another optimization:
  - The wires from the central buffer to sector buffers are length-matched.
    - Routed on top two (lowest resistance) layers.
    - Critical interconnects are split into 8 parallel wires each surrounded by VDD/GND return paths to optimize R, L, C.
    - Wire widths/spaces are further optimized.
  - The tunable trees are sized to reduce skew.
    - These trees have widely different loads.
    - The final clock grid is cut so that each leaf node drives the same load (leads to more skew than gridded network).

Case Study :: Alpha 21264 Clock Distribution

- Similar strategy of a tree-grid combination driving a global mesh called GCLK.
- The PLL clock signal is routed to the center of the die from where it is distributed by X and H trees to 16 distributed clock buffers.
- Clock buffers feed to a global clock mesh.
  - Skew is determined by grid and not gate load placement.
  - Universal availability of clock signal.
  - Good process variation tolerance.


PLL
GCLK grid

END OF LECTURE 30

Lecture 31: POWER AND GROUND ROUTING

PROF. INDRANIL SENGUPTA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Basic Problem

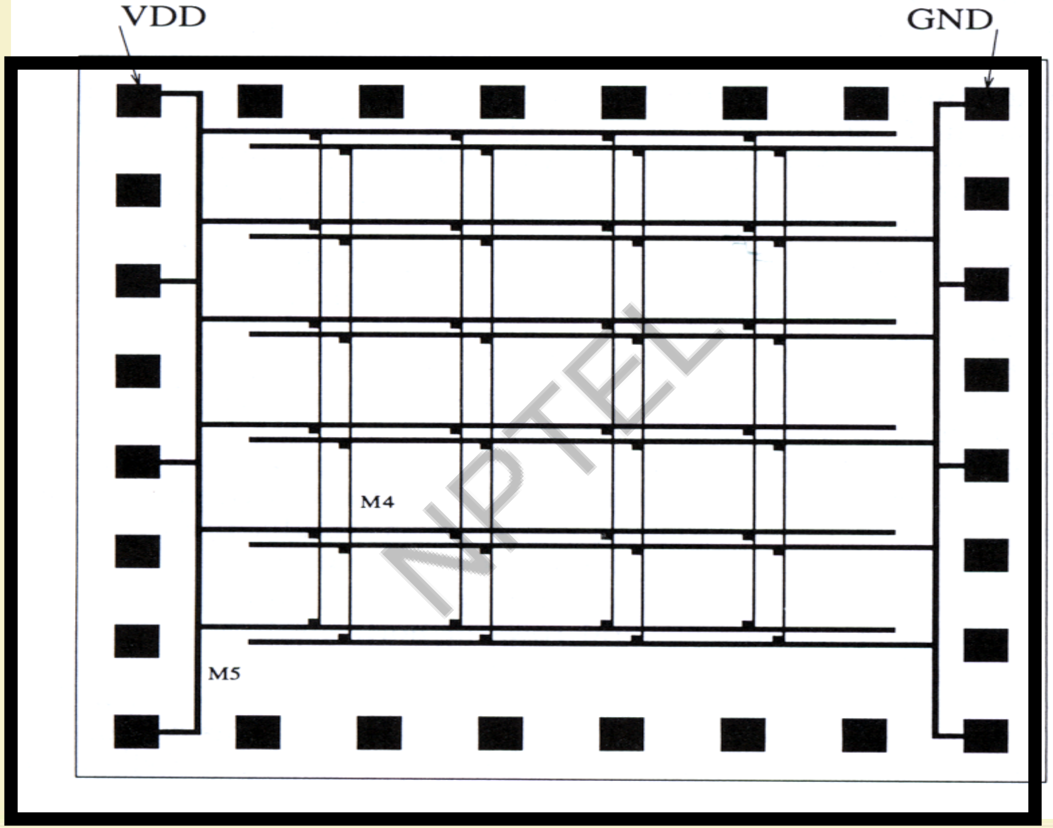
- In a design, almost all blocks require power and ground connections.
- Power and ground nets are usually laid out entirely on the metal layer(s) of the chip.
  - Due to smaller resistivity of metal.
  - Planar single-layer implementation is desirable since contacts (via' s) also significantly add to the parasitics.

- Routing of power (VDD) and ground (GND) nets consists of two main tasks:
  - Construction of interconnection topology.
  - Determination of the widths of the various segments.

- Requirement:
  - Find two non-intersecting interconnection trees.
  - The width of the trees at any particular point must be proportional to the amount of current being drawn by the points in that sub-tree.

Approach 1:: Grid Structure

- Several rows of horizontal wires for both VDD and GND run parallel to each other on one metal layer.
- The vertical wires run in another metal layer and connect the horizontal wires.
- A block simply connects to the nearest VDD and GND wire.


VDD
GND
M4
M5

Basic Steps Involved

Step 1: Creating a ring

- A ring is constructed to surround the entire core area of the chip, and possibly individual blocks.

Step 2: Connecting I/O pads to the ring

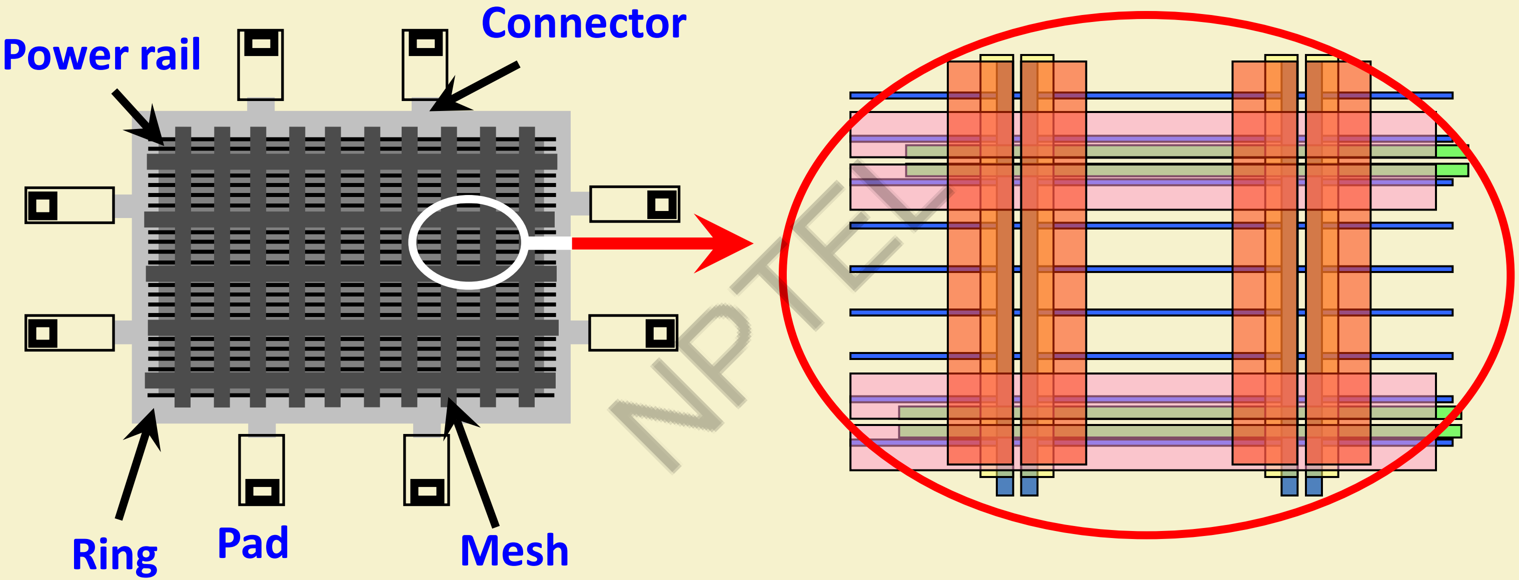
Step 3: Creating a mesh

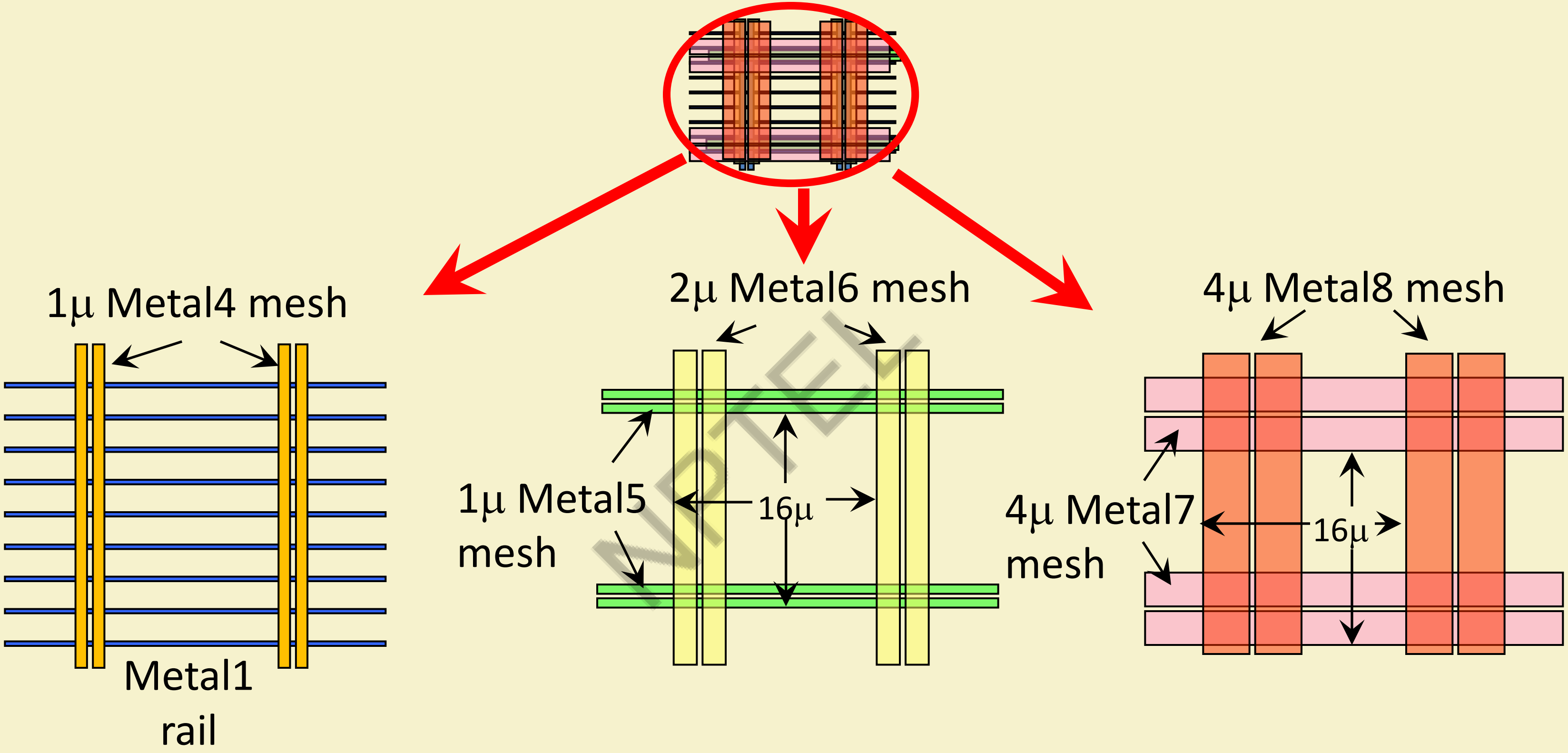
- A power mesh consists of a set of stripes at defined pitches on two or more layers.

Step 4: Creating rails on some metal layer (typically Metal1)

- Power mesh consists of a set of stripes at defined pitches on two or more layers.

Step 5: Connecting the metal rails to the mesh.


Power rail
Connector
Ring
Pad
Mesh


1μ Metal4 mesh
Metal1
rail
2μ Metal6 mesh
1μ Metal5
mesh
16μ
4μ Metal8 mesh
4μ Metal7
mesh
16μ

Approach 2:: Using Inter-digitated Trees

- Tends to route nets in an inter-digitated fashion.
- Extends one net from the left edge of the chip, and the other from the right.
  - Routing order of the connecting points is determined by the horizontal distances of the connecting points from the edge of the chip.

- Planar routing.
- Nets are determined by a combined Lee and Line Search algorithm.
  - Points of the left net which lie in the left half of the chip are routed using a fast line search algorithm.
  - Similarly, for the right net in the right half of the chip.
  - Next, all other points of the two sets are routed by Lee’s algorithm.

Basic Steps Involved

Step 1: Planarize the topology of the nets

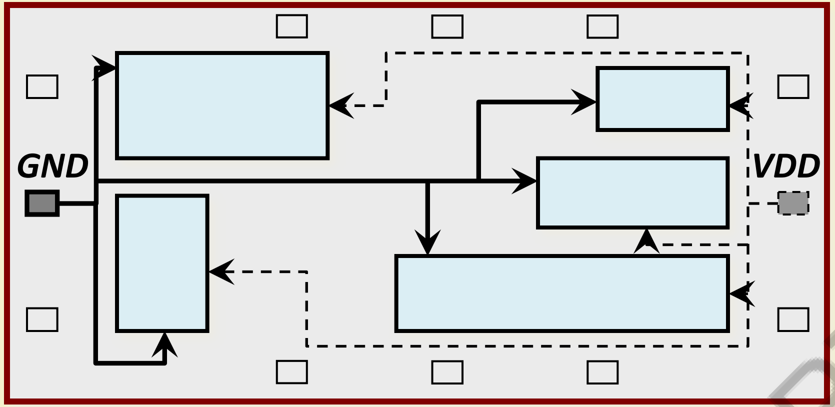
- As both power and ground nets must be routed on one layer, the design should be split using the Hamiltonian path.

Step 2: Layer assignment

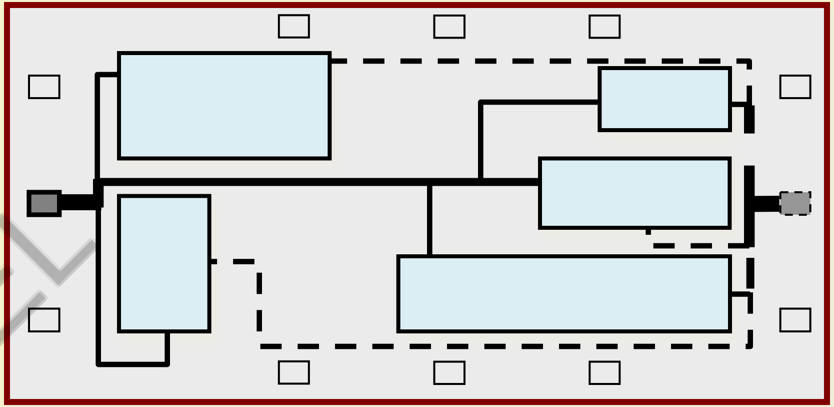
- Net segments are assigned to appropriate routing layers.

Step 3: Determining the widths of the net segments

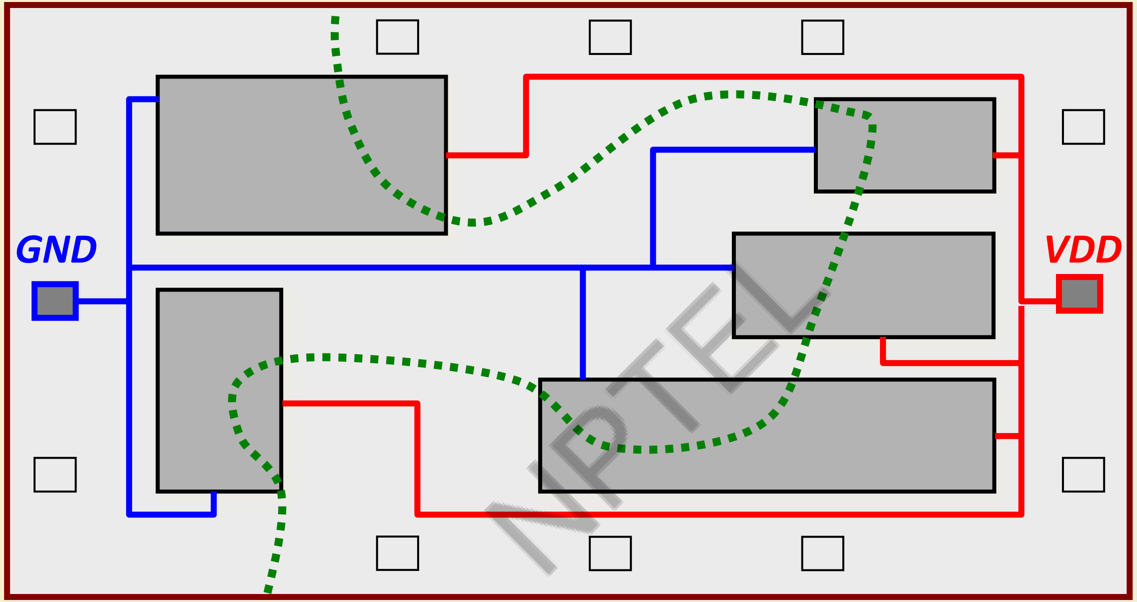
- A segment's width is determined from the sum of the currents from all the cells to which it connects



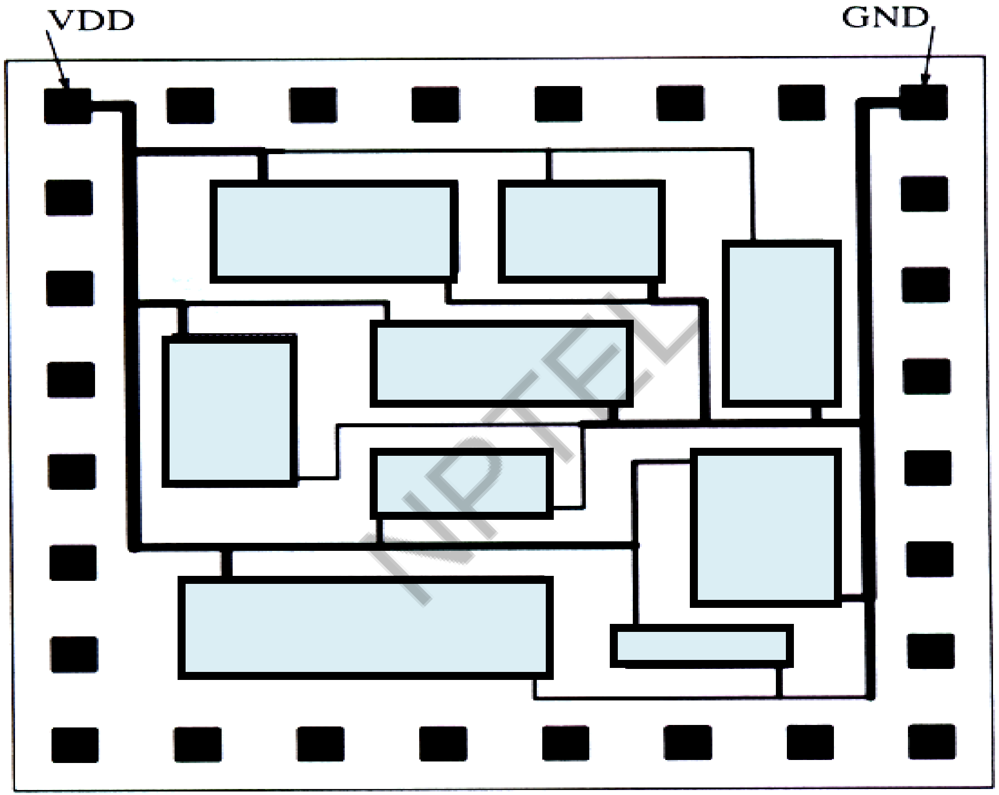
Generating topology of the two supply nets



Adjusting widths of the segments with regard to their current loads



- Determine a Hamiltonian path connecting all the terminal points.
  - A path that visits every vertex exactly once.


VDD
GND

Summary

- Power and ground routing needs special attention because of wire widths.
  - Non-uniform wire widths.
  - Careful sizing of wires is required.
- Routing of power and ground nets is often given first priority.
  - Usually laid out entirely on metal layer(s).
  - Signal nets may share the metal layer(s) with power and ground, but they change layers whenever a power or ground wire is encountered.
- Choice of layer:
  - Aluminium :: most widely used.

END OF LECTURE 31