IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

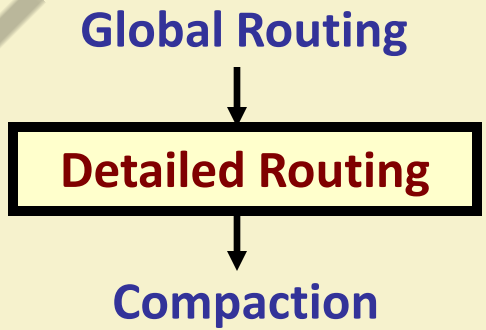
Lecture 20: DETAILED ROUTING (PART 1)

PROF. INDRANIL SENGUPTA

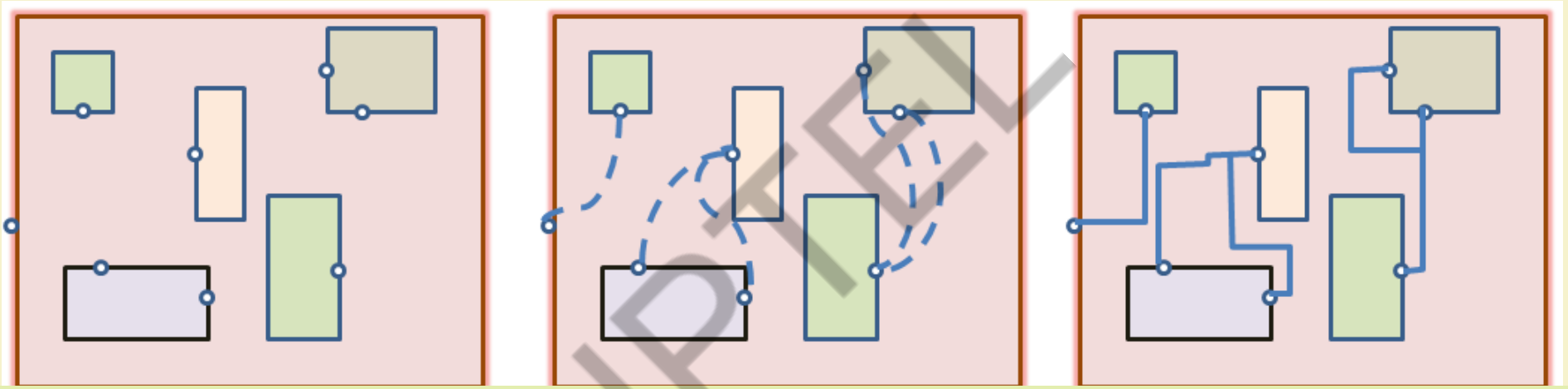
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Detailed Routing

- Find actual geometric layout of each net within assigned routing regions.
- No layouts of two different nets should intersect on the same layer.
- Problem is solved incrementally, one region at a time in a predefined order.



A Routing Example



Placement

Global Routing

Detailed Routing

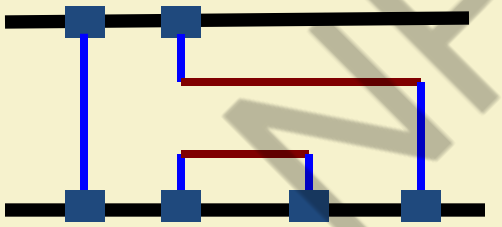
After Global Routing

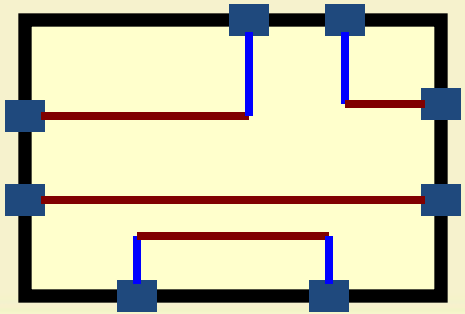
- The two-stage routing method is a powerful technique for routing.
- During the global routing stage:
  - The routing region is partitioned into a collection of rectangular regions.
  - To interconnect each net, a sequence of sub-regions to be used is determined.
  - All nets crossing a given boundary of a routing region are called *floating terminals*.
  - Once the sub-region is routed, these floating terminals become fixed terminals for subsequent regions.

Channels and Switchboxes

- There are normally two kinds of rectilinear regions.
  - Channels: routing regions having two parallel rows of fixed terminals.
  - Switchboxes: generalizations of channels that allow fixed terminals on all four sides of the region.

Channel

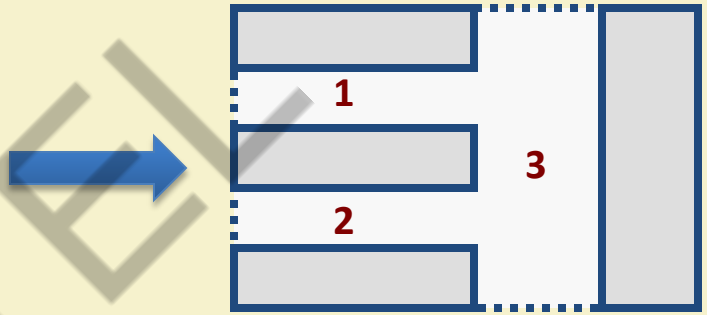




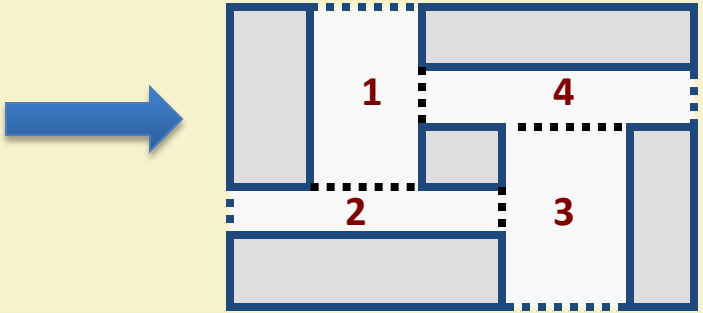
Switchbox

Order of Routing Regions

- Slicing placement topology.
  - Nets can be routed by considering channels 1, 2 and 3 in order.



- Non-slicing placement topology.
  - Channels with cyclic constraints.
  - Some of the routing regions are to be considered as switchboxes.

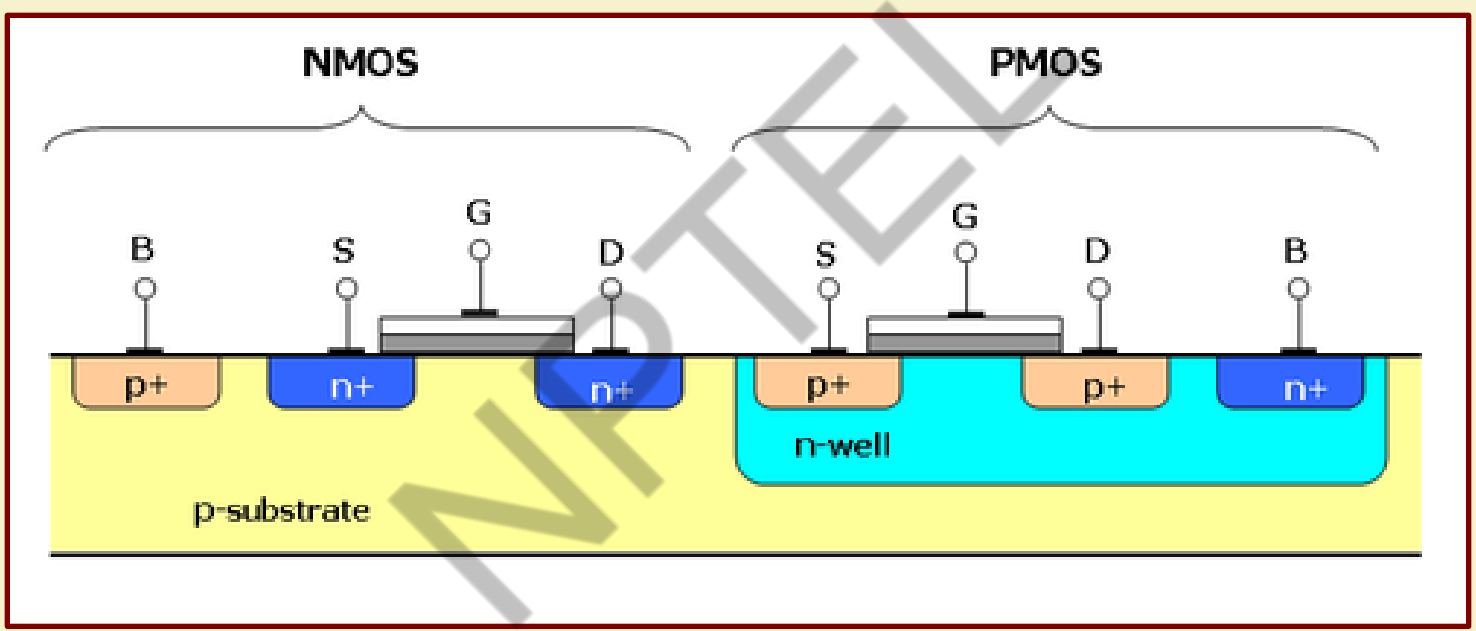


Routing Considerations

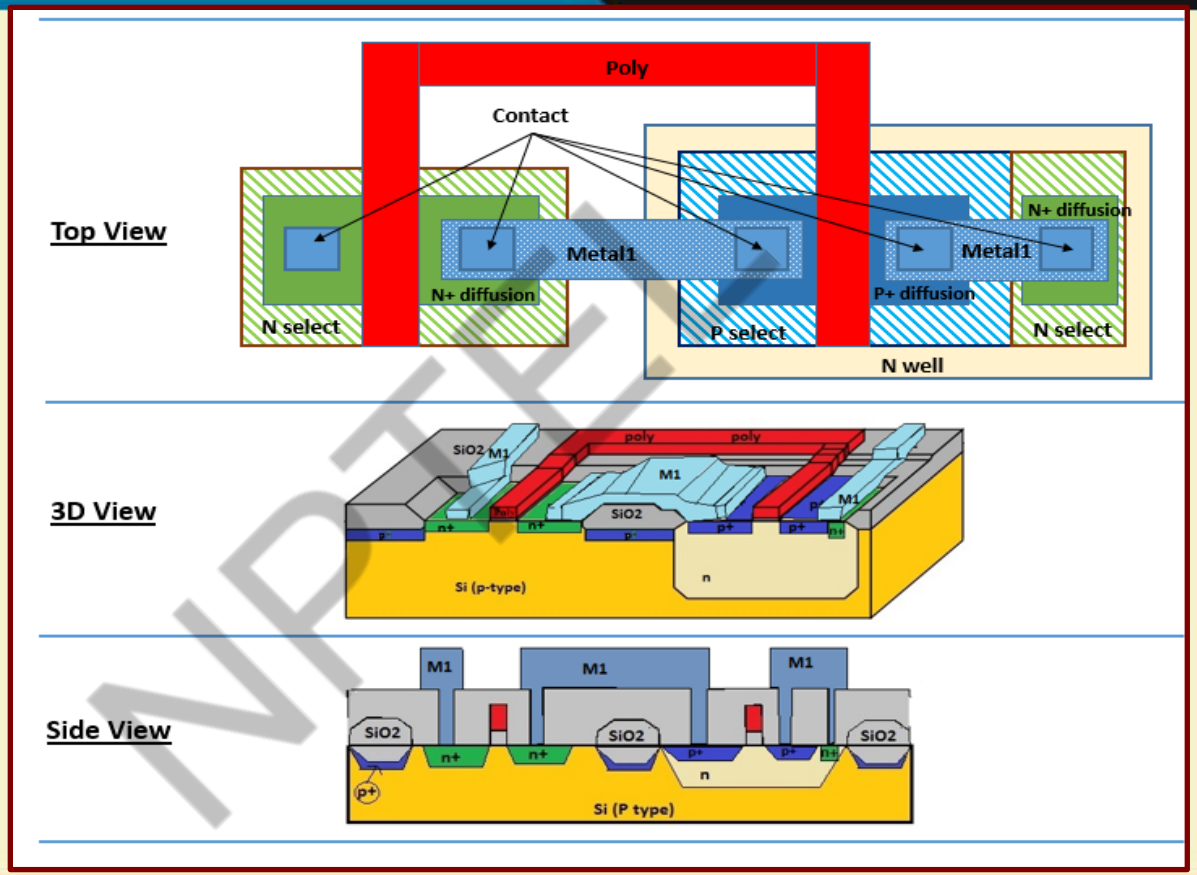
- Number of terminals
  - Majority of nets are two-terminal ones.
  - For some nets (viz. clock, power), number of terminals can be very large.
  - Each multi-terminal net can be decomposed into several two-terminal nets.
- Net width
  - Power and ground nets have greater width.
  - Signal nets have less width.

- Via restrictions
  - *Regular*: only between adjacent layers.
  - *Stacked*: passing through more than two layers.
- Boundary type
  - *Regular*: straight border of routing region
  - *Irregular*: arbitrary
- Number of layers
  - Modern fabrication technology allows at least five layers of routing.
- Net types
  - *Critical*: power, ground, clock nets
  - *Non-critical*: signal nets

CMOS Fabrication



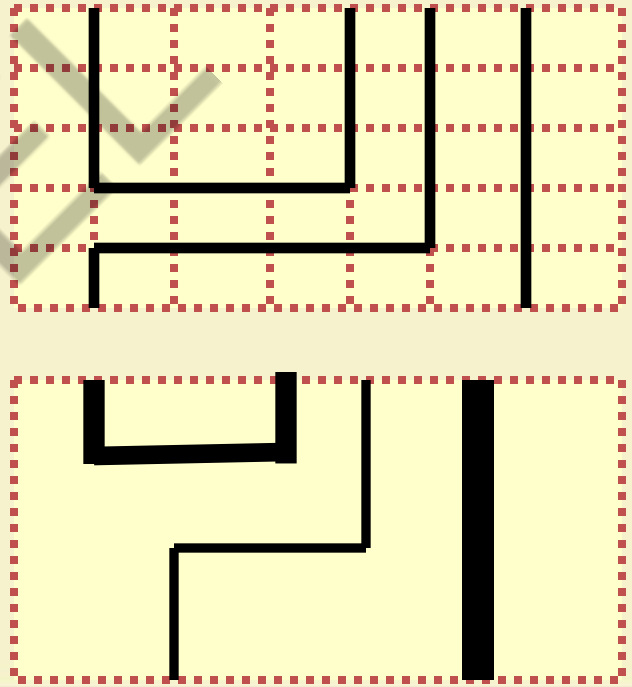
Via Connections



Routing Models

- Grid-based model
  - A grid is super-imposed on the routing region.
  - Wires follow paths along the grid lines.

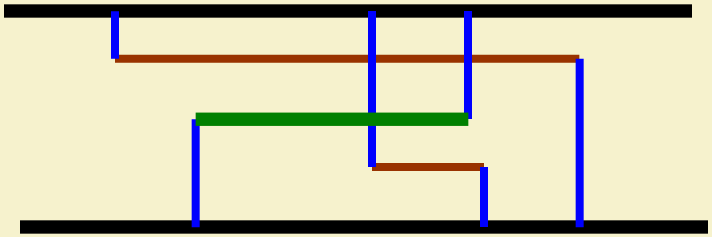
- Gridless model
  - Does not follow the gridded approach.



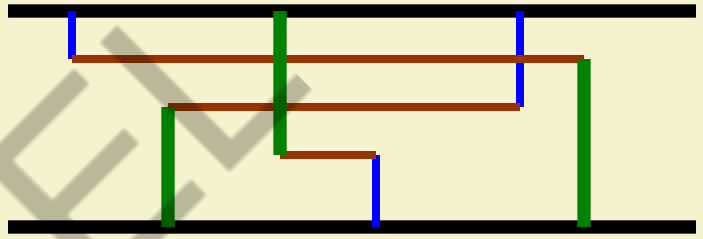
Models for Multi-Layer Routing

- Unreserved layer model
  - Any net segment is allowed to be placed in any layer.
- Reserved layer model
  - Certain types of segments are restricted to particular layer(s).
    - Two-layer (HV, VH)
    - Three-layer (VHV, HVH)

Illustration



HVH Model

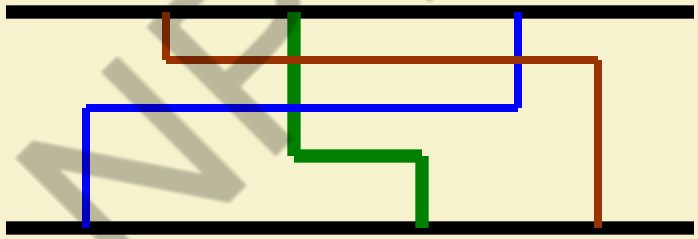


VHV Model

Layer 1

Layer 2

Layer 3



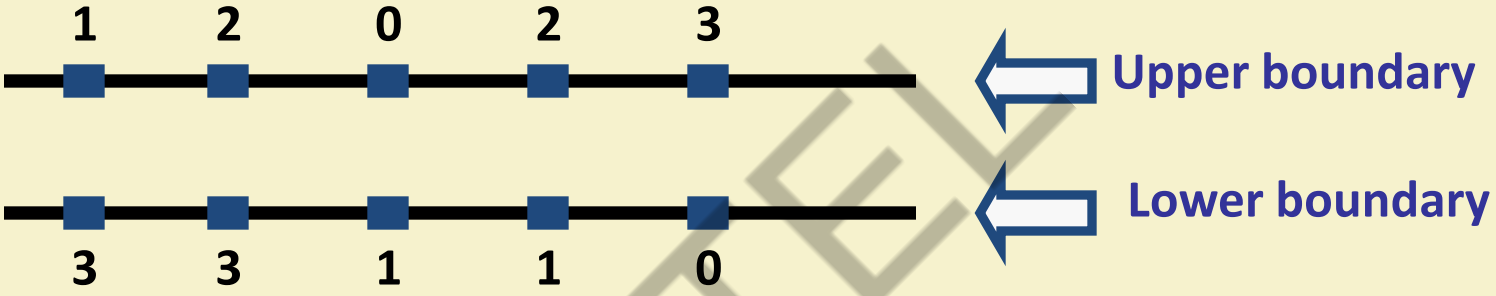
Unreserved Layer Model

Channel Routing

- In channel routing, interconnections are made within a rectangular region having no obstructions.
  - A majority of modern-day ASICs use channel routers.
  - Algorithms are efficient and simple.
  - Guarantees 100% completion if channel width is adjustable.

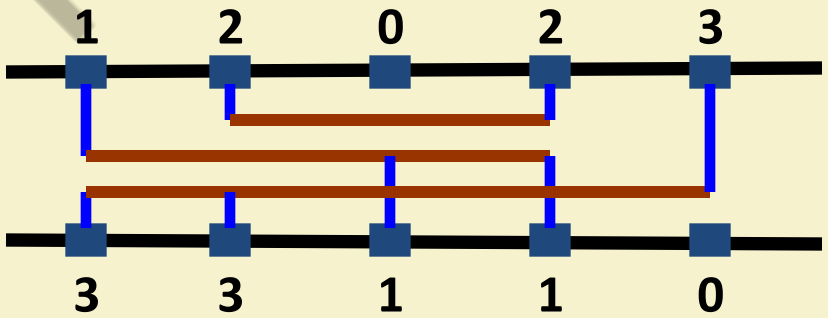
- Some terminologies:
  - Track: horizontal row available for routing.
  - Trunk: horizontal wire segment.
  - Branch: vertical wire segment connecting trunks to terminals.
  - Via: connection between a branch and a trunk.

Channel Routing Problem :: Terminologies



Net list:: TOP = [1 2 0 2 3]

BOT = [3 3 1 1 0]



Problem Formulation

- The channel is defined by a rectangular region with two rows of terminals along its top and bottom sides.
  - Each terminal is assigned a number between 0 and N.
  - Terminals having the same label i belong to the same net i.
  - A '0' indicates no connection.

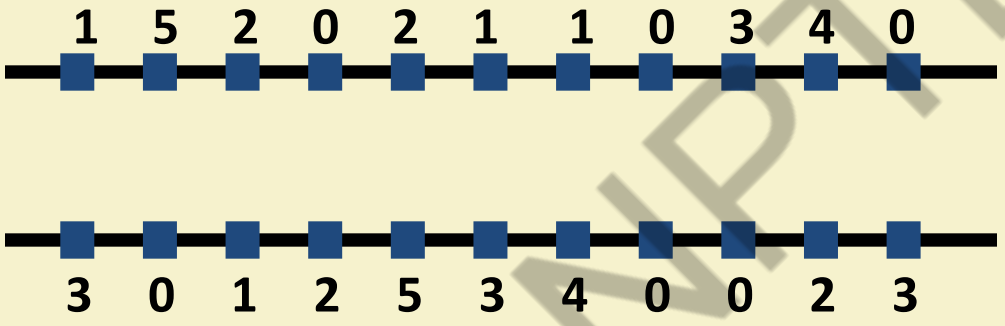
- The task of the channel router is to:
  - Assign horizontal segments of nets to tracks.
  - Assign vertical segments to connect
    a) Horizontal segments of the same net in different tracks.
    b) The terminals of the net to horizontal segments of the net.
- Channel height should be minimized.
- Horizontal and vertical constraints must be met.

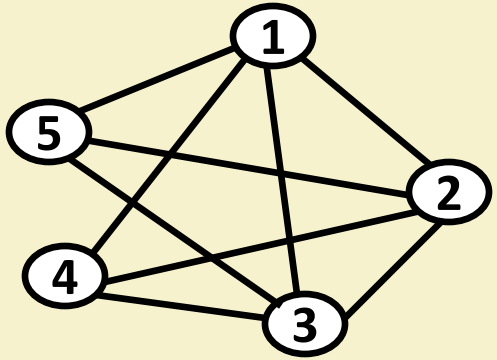
Channel Constraints

- Horizontal constraints between two nets:
  - The horizontal span of two nets overlaps each other.
  - The nets must be assigned to separate tracks.
- Vertical constraints between two nets:
  - There exists a column such that the terminal i on top of the column belongs to one net, and the terminal j on bottom of the column belongs to the other net.
  - Net i must be assigned a track above that for net j.

Horizontal Constraint Graph (HCG)

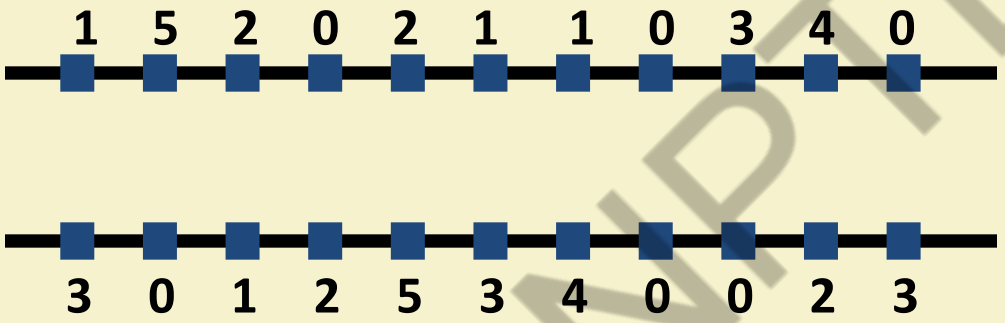
- It is a graph where vertices represent nets, and edges represent horizontal constraints.

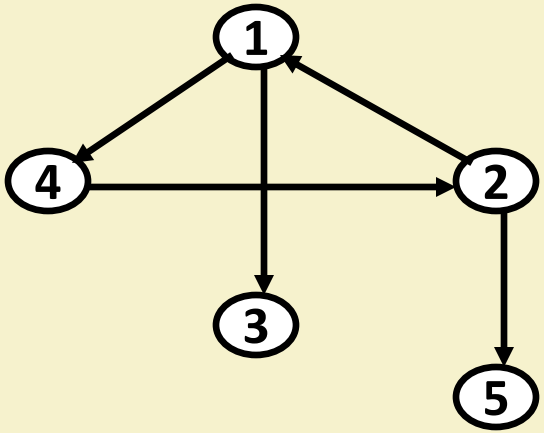




Vertical Constraint Graph (VCG)

- It is a directed graph where vertices represent nets, and edges represent vertical constraints.





END OF LECTURE 20

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

Lecture 21: DETAILED ROUTING (PART 2)

PROF. INDRANIL SENGUPTA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Two-layer Channel Routing

- Left-Edge Algorithms (LEA)
  - Basic Left-Edge Algorithm
  - Left-Edge Algorithm with Vertical Constraints
  - Dogleg Router
- Constraint-Graph Based Algorithm
  - Net Merge Channel Router
- Greedy Channel Router
- Hierarchical Channel Router

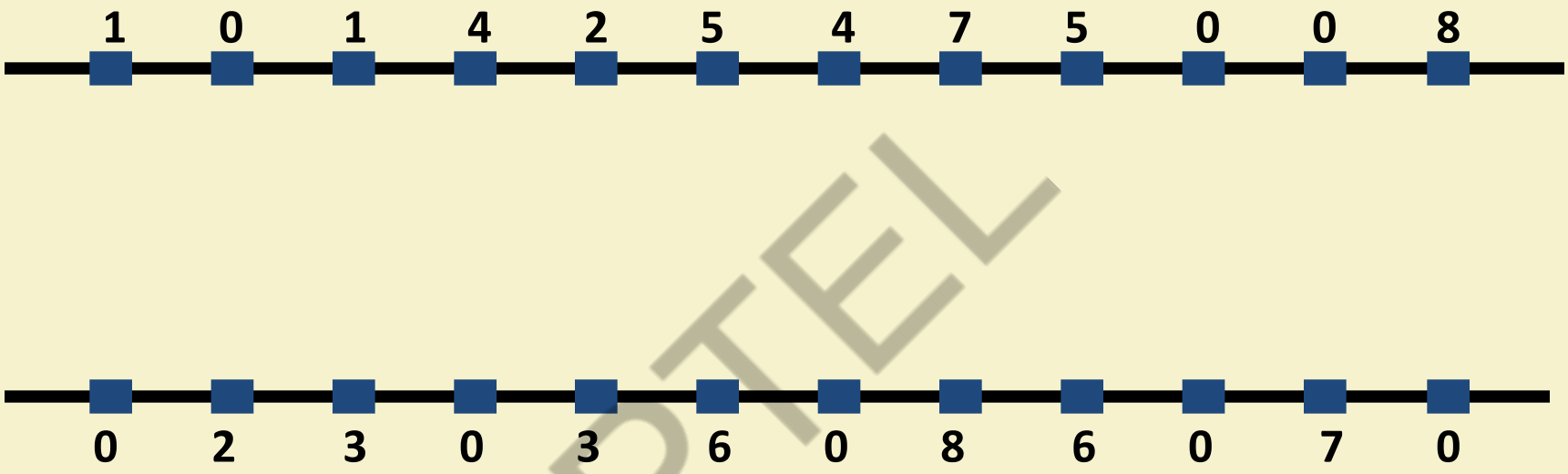
Basic Left Edge Algorithm

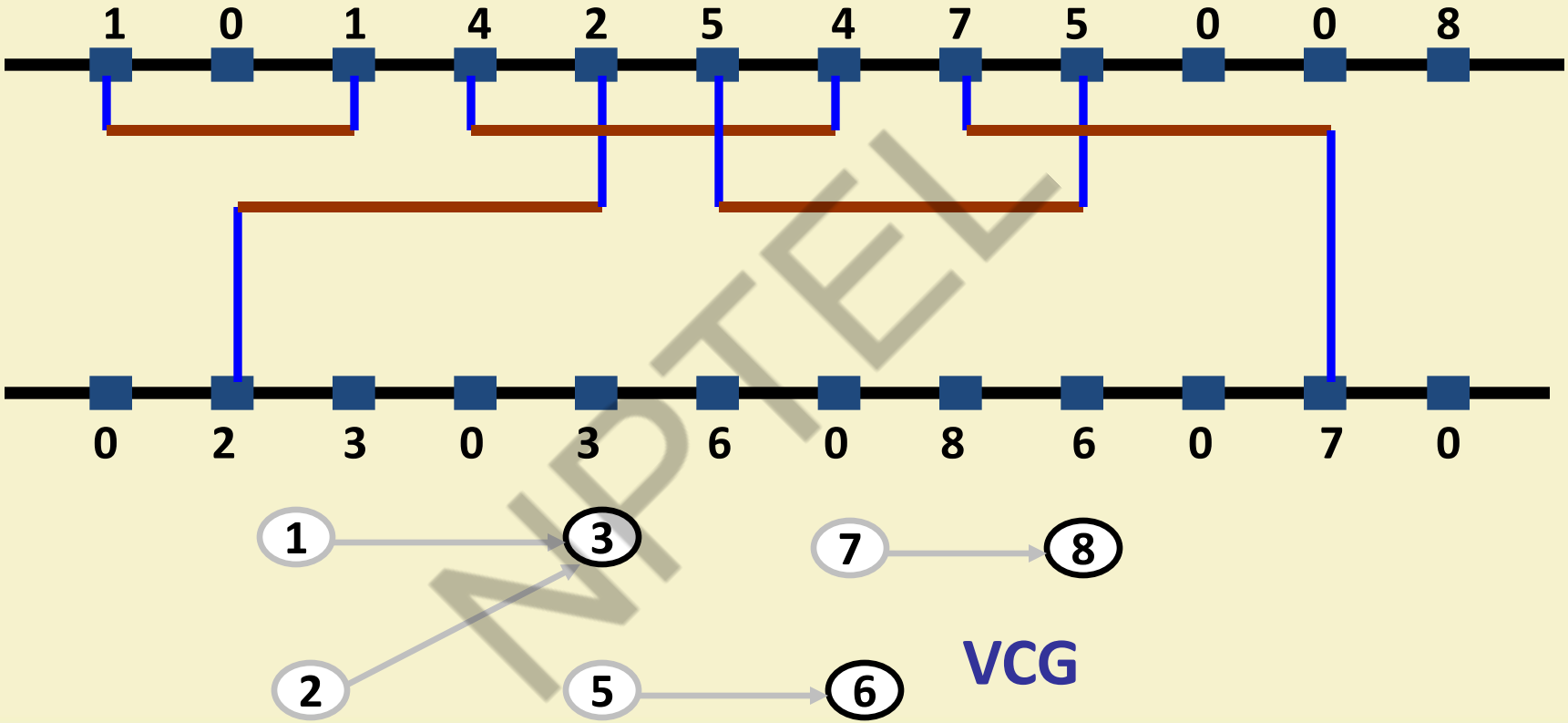
- Simplest channel routing algorithm.
- Assumptions:
  - Only two-terminal nets.
  - No vertical constraints.
  - HV two-layer model.
  - Doglegs are not allowed.

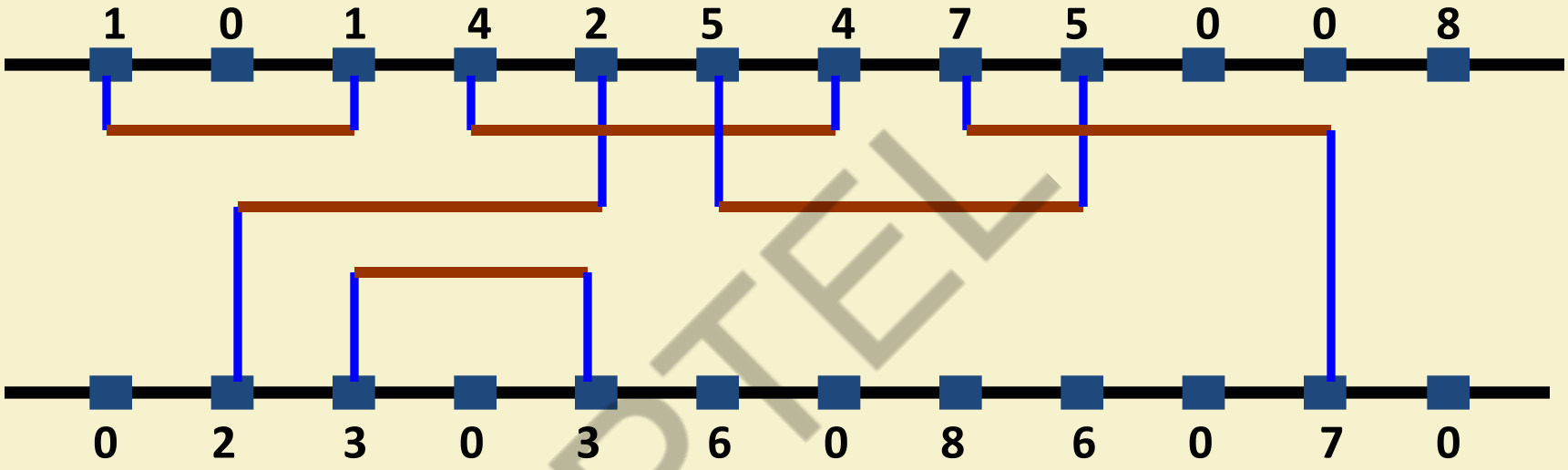
- Basic Steps:
  - Sort the nets according to the x-coordinate of the leftmost terminal of the net.
  - Route the nets one-by-one according to the order.
  - For a net, scan the tracks from top to bottom, and assign it to the first track that can accommodate it.
- In the absence of vertical constraints, the algorithm produces a minimum-track solution.

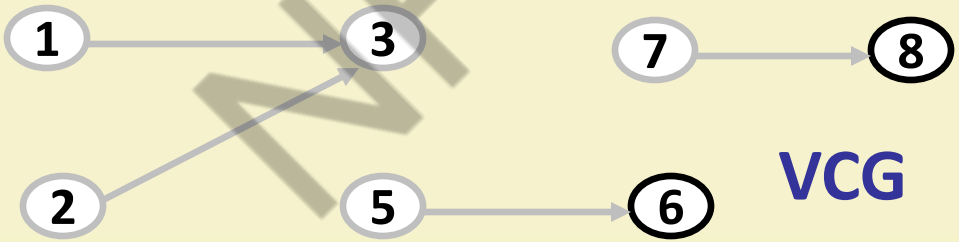
Extension to Left-Edge Algorithm

- Vertical constraints may exist, but there are no directed cycles in the VCG.
- Select a net for routing if both the following conditions are true:
  a) The x-coordinate of the leftmost terminal is the least.
  b) There is no edge incident on the vertex corresponding to that net in the VCG.
- After routing a net, the corresponding vertex and the incident edges are deleted from the VCG.
- Other considerations are the same as the basic left-edge algorithm.

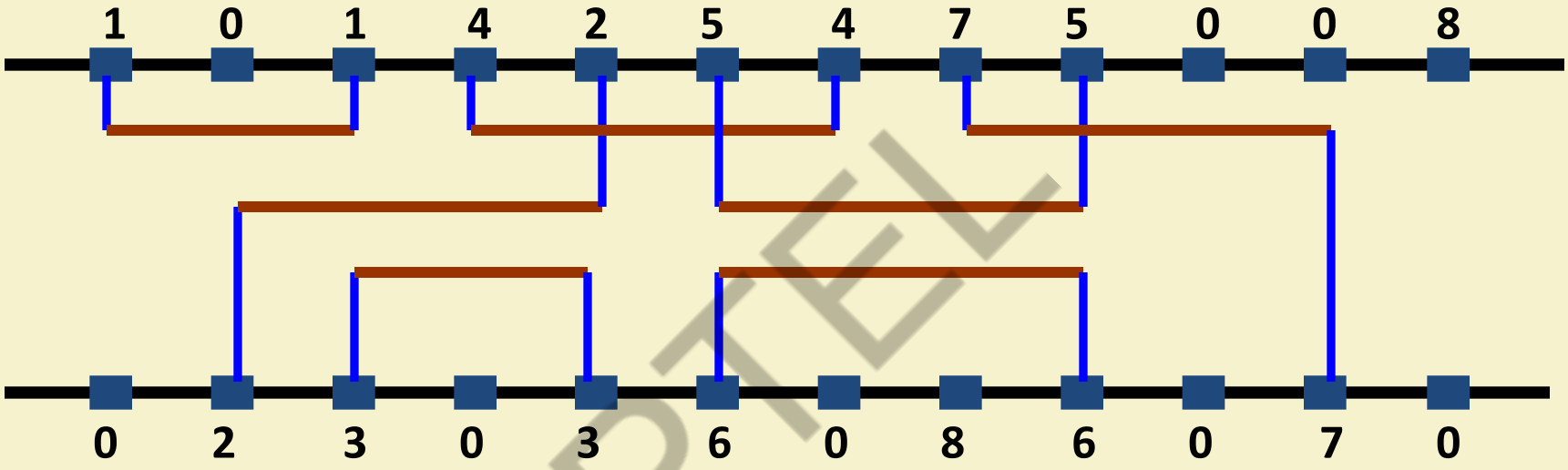


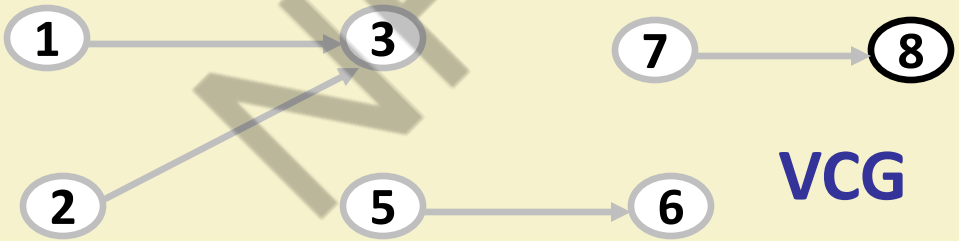




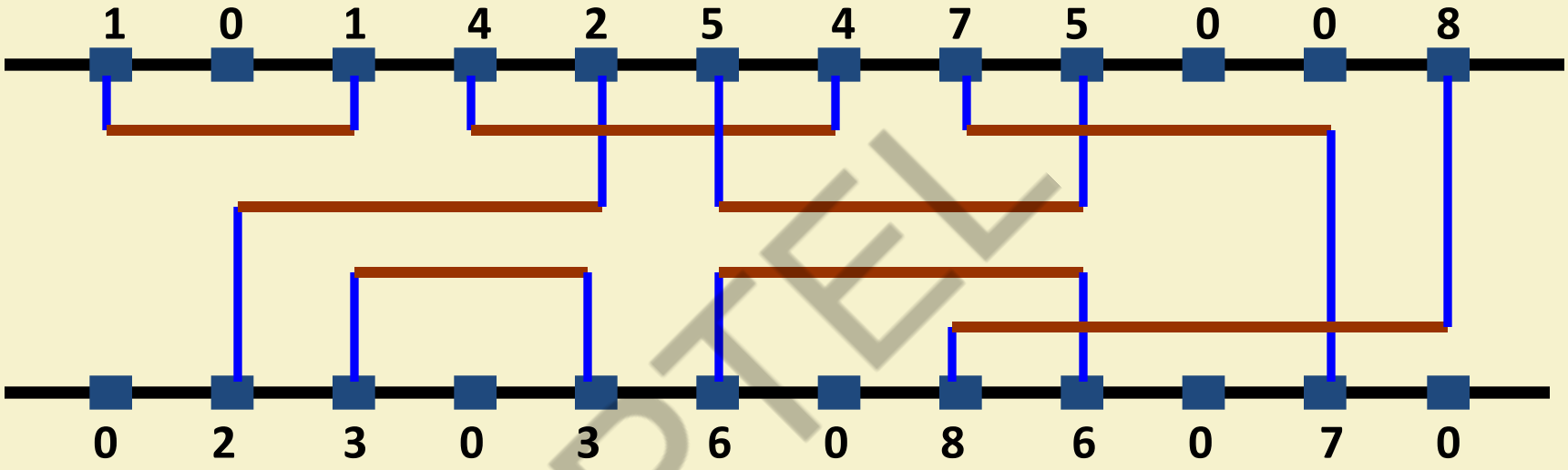


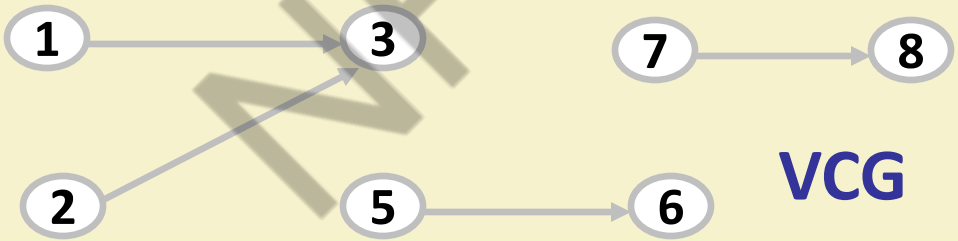
VCG





VCG





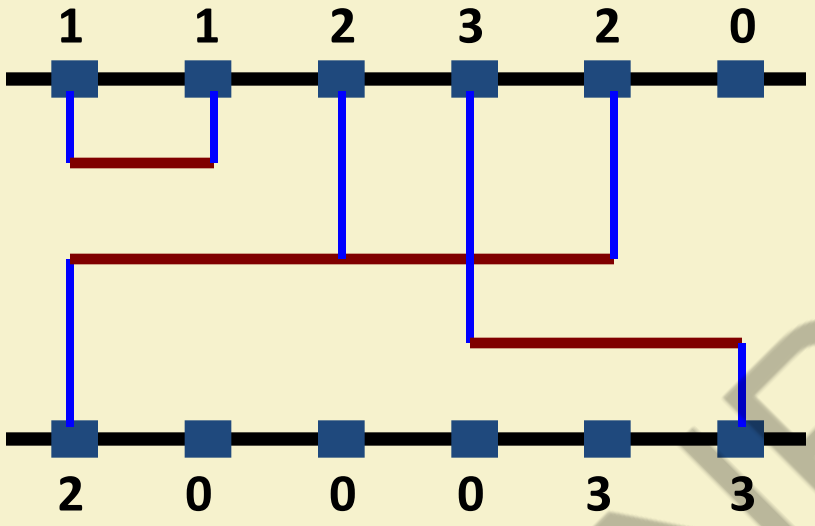
VCG

Dogleg Router

- Drawback of LEA:
  - The entire net is on a single track.
  - Sometimes leads to routing with more tracks than necessary.
- Doglegs are used to place parts of the same net on different tracks.
  - A dogleg is a vertical segment that connects two trunks located in two different tracks.
  - May lead to a reduction in channel height.

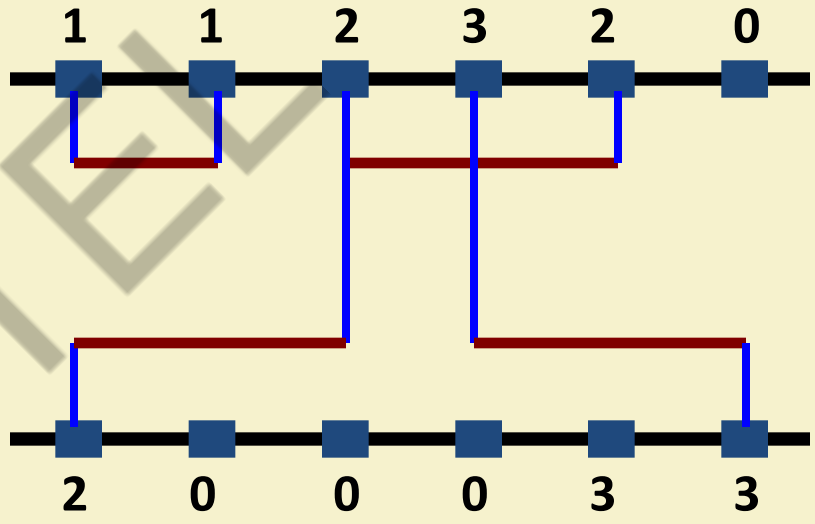
- Dogleg router allows multi-terminal nets and vertical constraints.
  - Multi-terminal nets are broken into a series of two-terminal nets.
- Cannot handle cyclic vertical constraints.

Dogleg Example



No dogleg

3 tracks

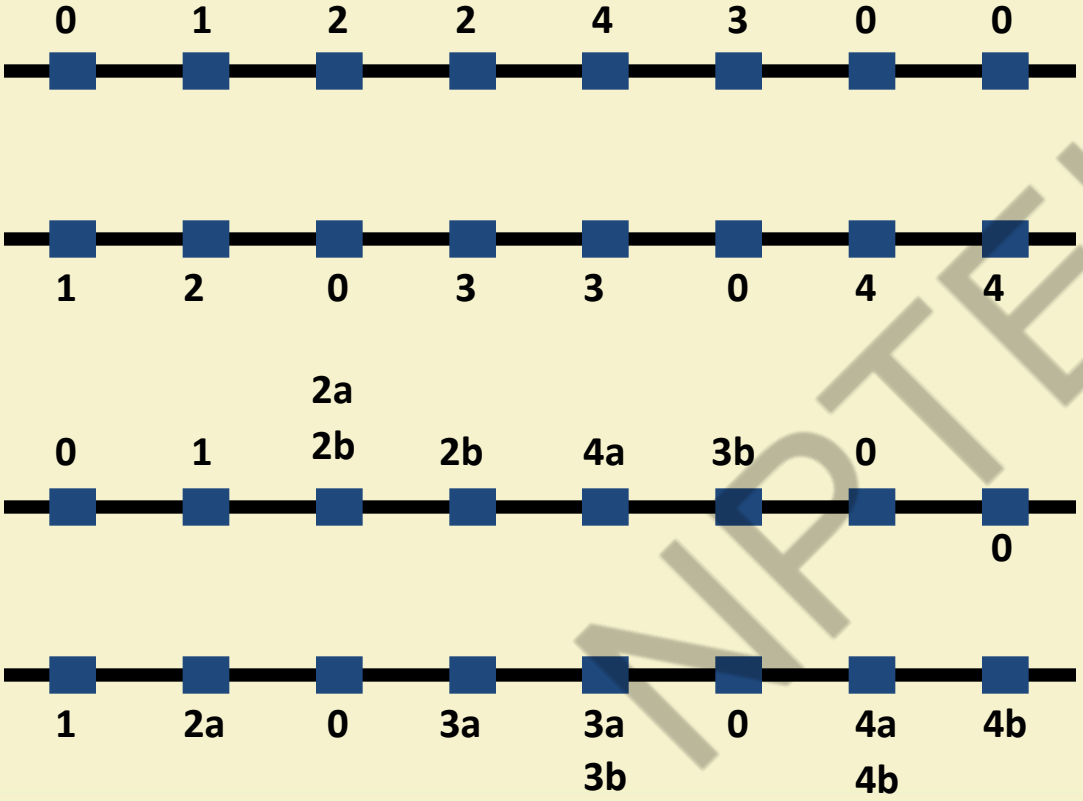


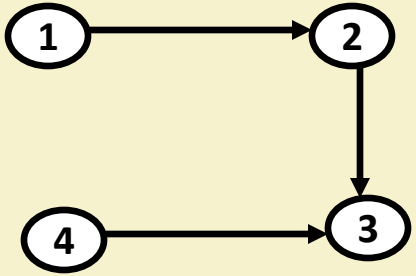
With dogleg

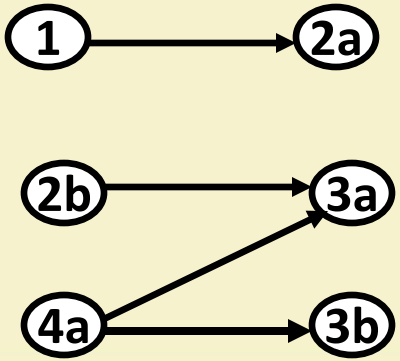
2 tracks

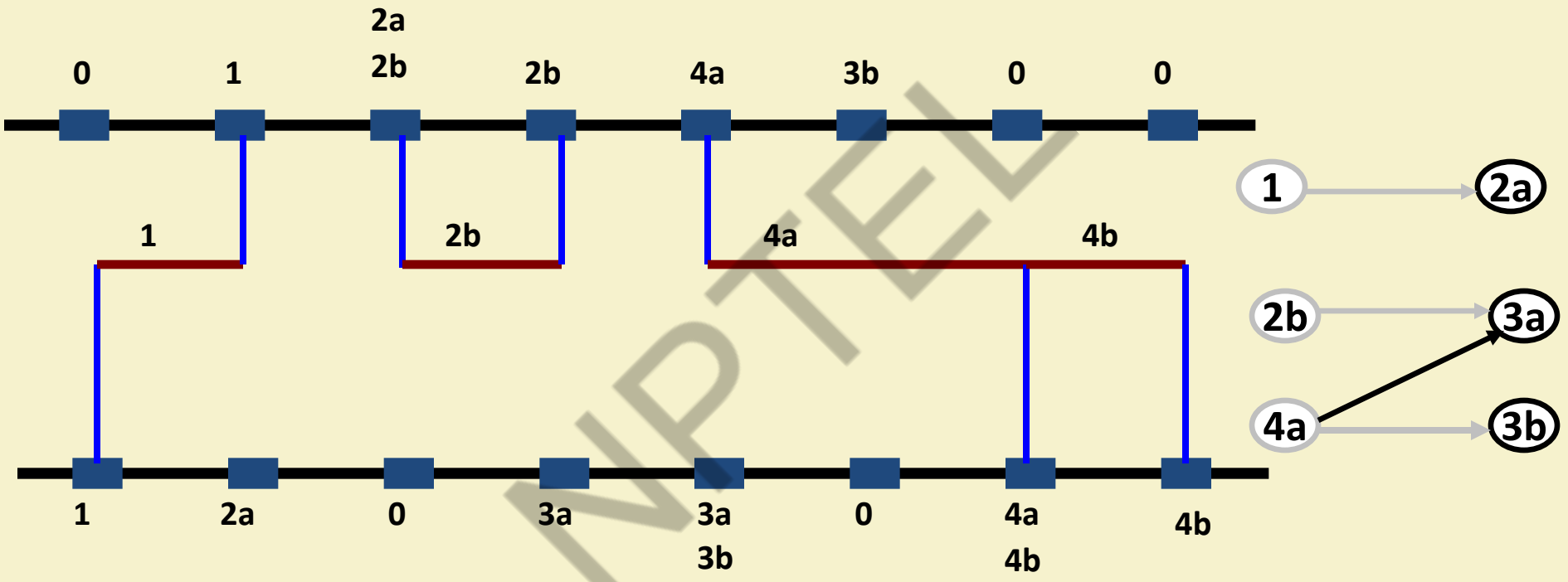
Dogleg Router: Algorithm

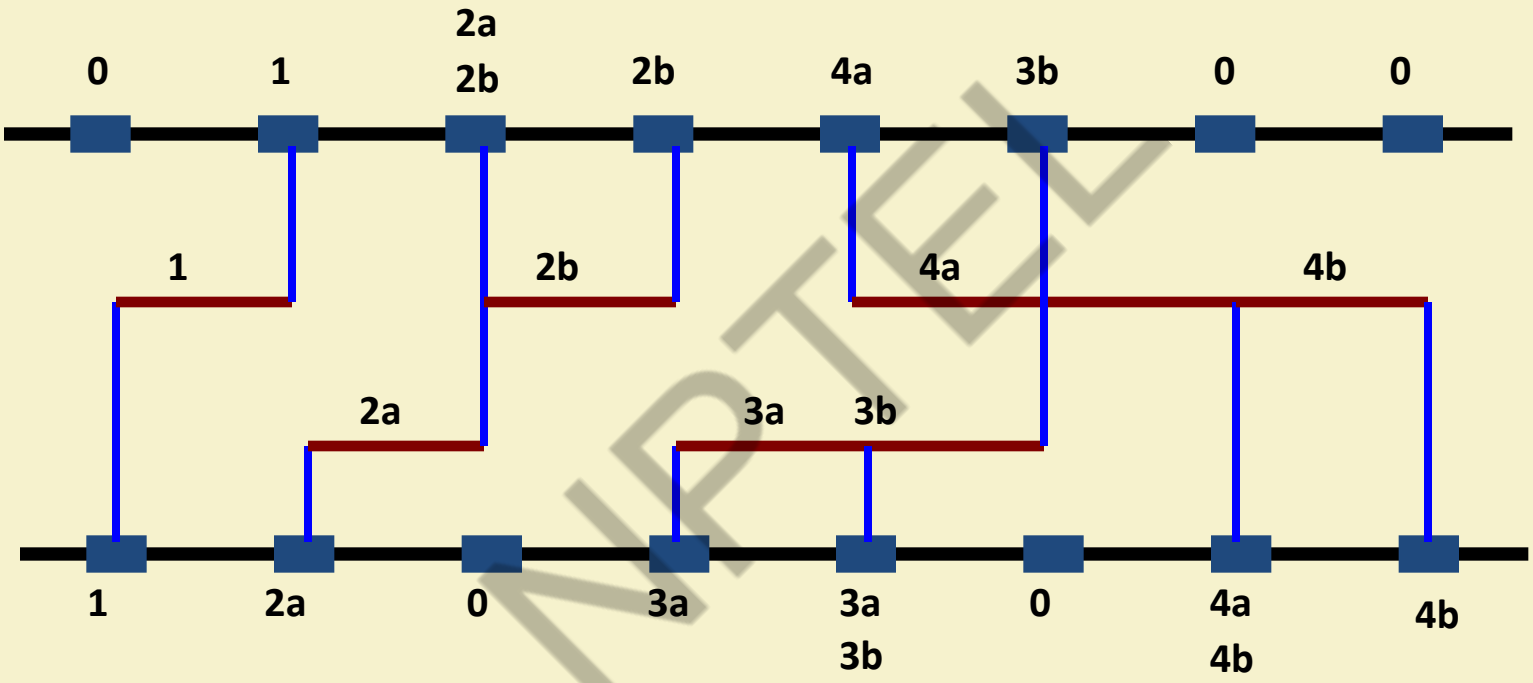
- Step 1:
  - If cycle exists in the VCG, return with failure.
- Step 2:
  - Split each multi-terminal net into a sequence of 2-terminal nets.
    - A net 2 .. 2 .. 2 will get broken as 2a .. 2a 2b .. 2b.
  - HCG and VCG gets modified accordingly.
- Step 3:
  - Apply the extended left-edge algorithm to the modified problem.


0 1 2 2 4 3 0 0
1 2 0 3 3 0 4 4
2a
2b
0 1 2b 4a 3b 0
0
1 2a 0 3a 3a 0 4a 4b
3b
4b


1
2
4
3


1
2a
2b
3a
4a
3b





END OF LECTURE 21

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

Lecture 22: DETAILED ROUTING (PART 3)

PROF. INDRANIL SENGUPTA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Net Merge Channel Router

- Due to Yoshimura and Kuh.
- Basic idea:
  - If there is a path of length *p* in the VCG, at least *p* horizontal tracks are required to route the channel.
  - Try to minimize the longest path in the VCG.
  - Merge nodes of VCG to achieve this goal.
- Does not allow doglegs or cycles in the VCG.

How does it work?

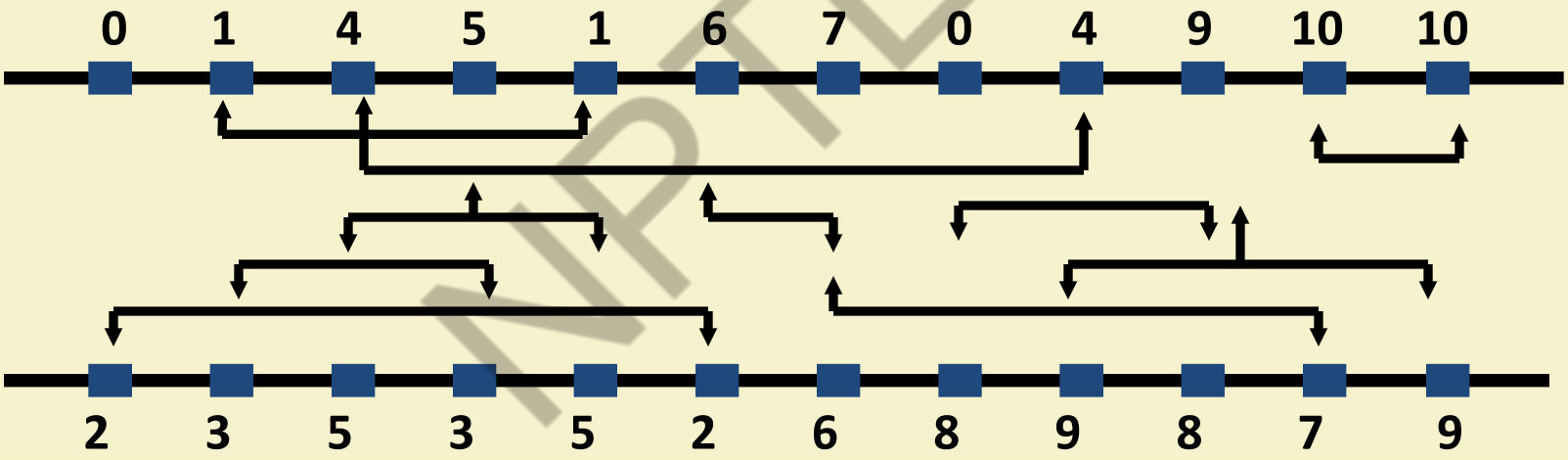
- Partition the routing channel into a number of regions called *zones*.
- Nets from adjacent zones are merged.
  - Merged nets are treated as a *composite net* and assigned to a single track.

Key Steps of the Algorithm

a) Zone representation

b) Net merging

c) Track assignment

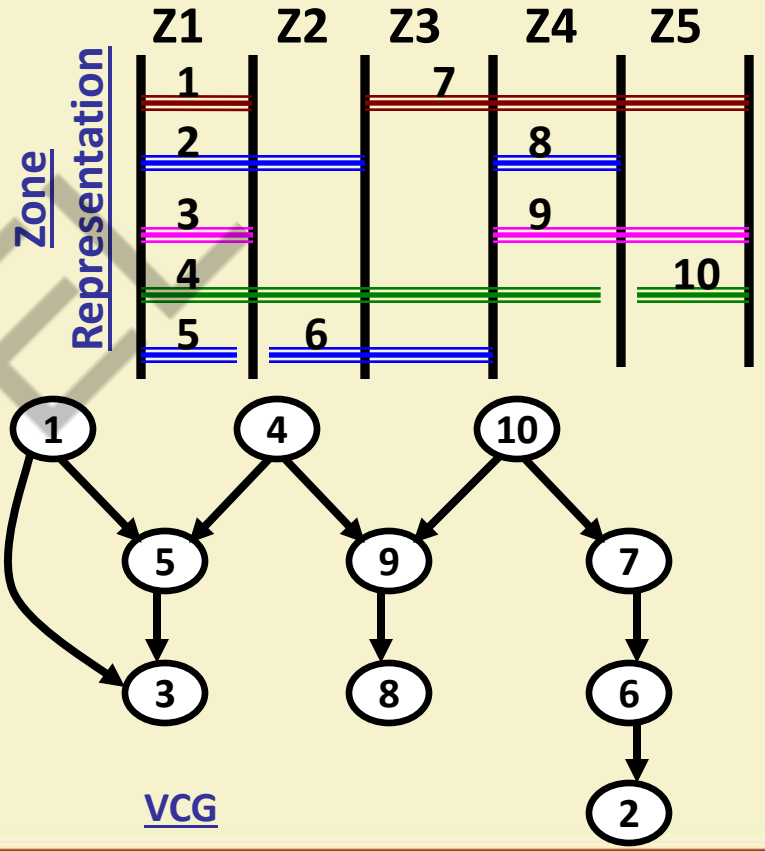


Step 1: Zone Representation

- Let *S(i)* denote the set of nets whose horizontal segments intersect column *i*.
- Take only those *S(i)* which are maximal, that is, not a proper subset of some other *S(j)*.
- Define a zone for each of the maximal sets.
- In terms of HCG / interval graph, a zone corresponds to a maximal clique in the graph.

Column	S(i)	Zone
1	{2}	
2	{1,2,3}	
3	{1,2,3,4,5}	1
4	{1,2,3,4,5}	
5	{1,2,4,5}	
6	{2,4,6}	2
7	{4,6,7}	3
8	{4,7,8}	
9	{4,7,8,9}	4
10	{7,8,9}	
11	{7,9,10}	5
12	{9,10}	

Zone Table



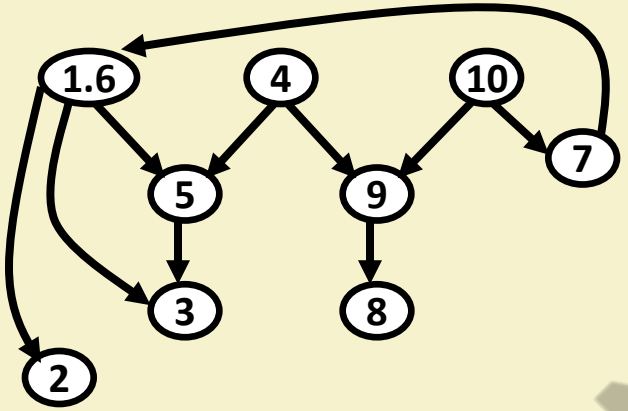
VCG

Step 2: Net Merging

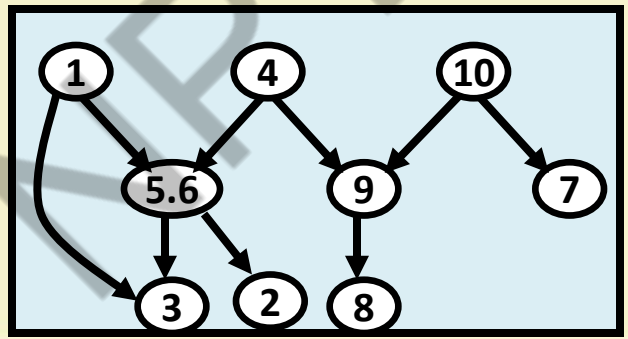
- Let N_i and N_j be two nets for which the following conditions are satisfied:
  - There is no edge between v_i and v_j in HCG.
  - There is no directed path between v_i and v_j in VCG.
- Nets N_i and N_j can be *merged* to form a new composite net.
  - Modifies VCG by merging nodes v_i and v_j into a single node $v_{i.j}$.
  - Modifies HCG / zone representation by replacing nodes v_i and v_j by a net v_{i-j}, which occupies the consecutive zones including those of nets N_i and N_j.

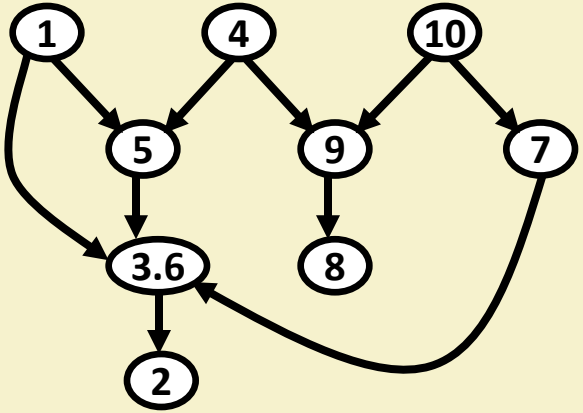
- The process is iterative:
  - Pairs of nodes are successively merged.
  - At every step of the iteration, in case of multiple choices, merge the net-pair that minimizes the length of the longest path in the VCG.
  - That is, the increase in length is minimum.
- A result:
  - If the original VCG has no cycles, then the updated VCG with merged nodes will not have cycles either.

- Iteration 1 of the example:
  - We can merge nets pairs (1,6), (3,6) or (5,6).

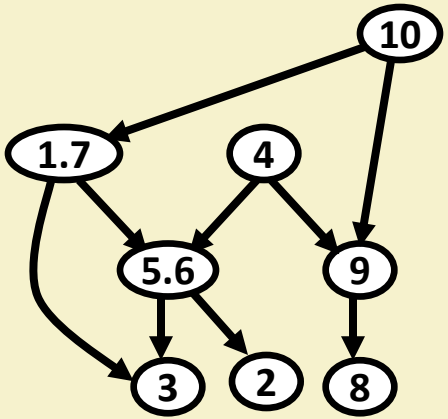


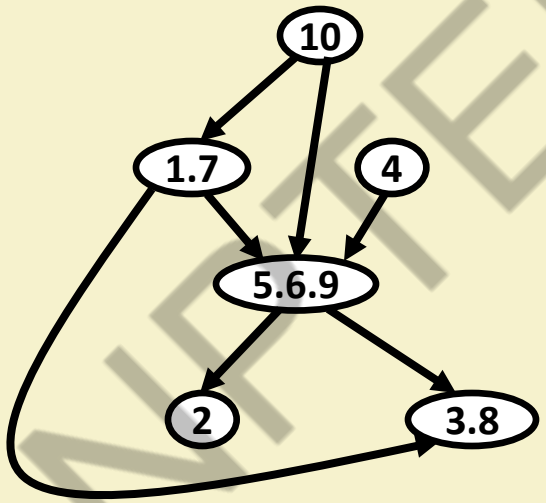
Best Choice

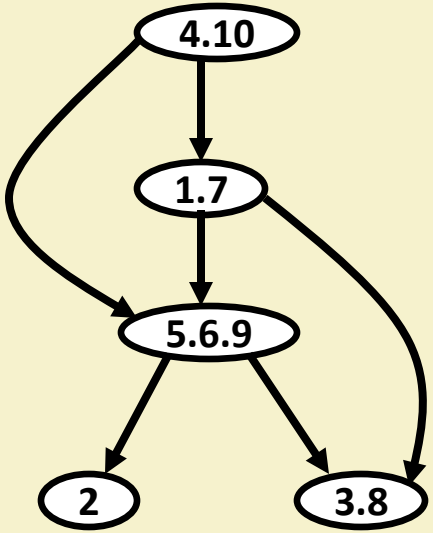




- Successive iteration steps:







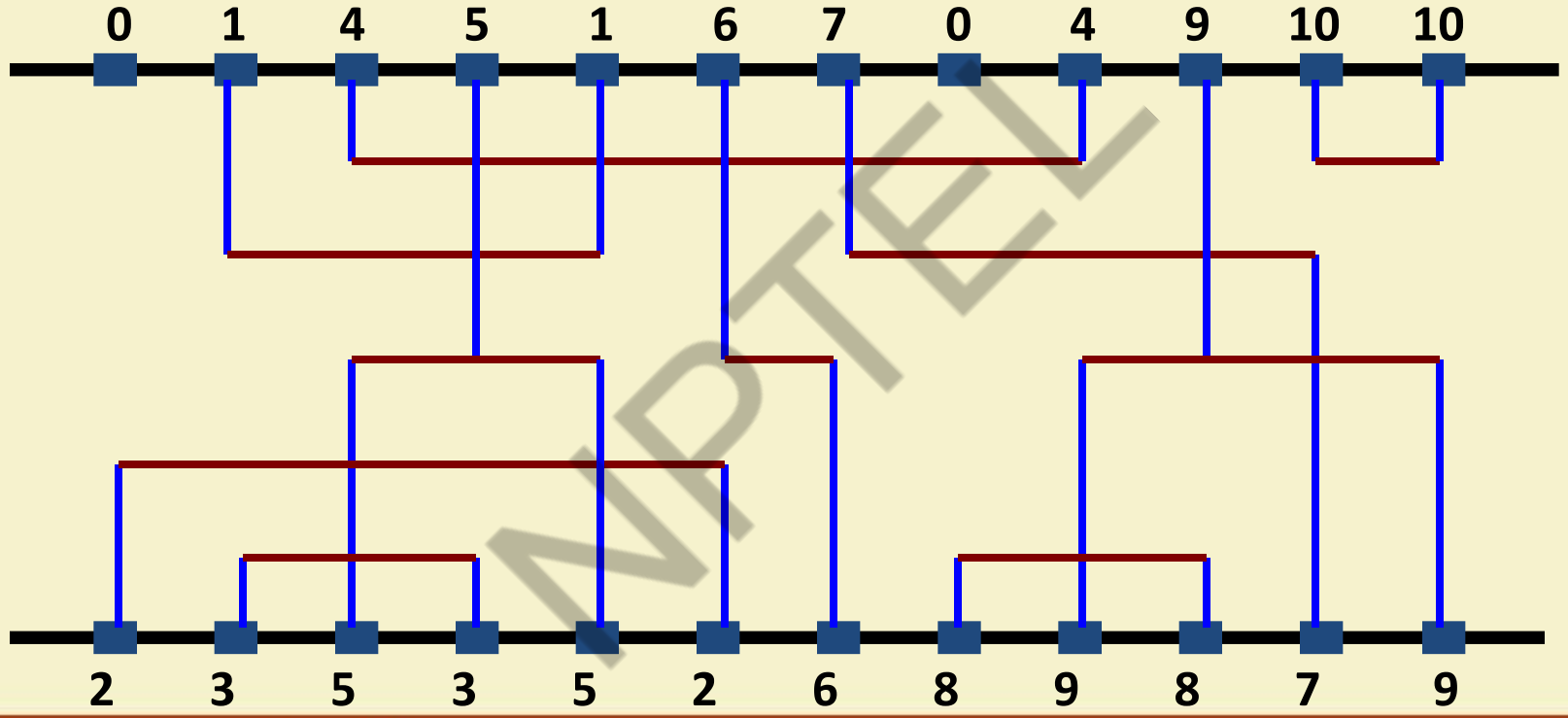
Step 3: Track Assignment

- Each node in the final graph is assigned a separate track.
- Actually we apply the left-edge algorithm to assign horizontal tracks to the merged nets.
  - The list of nets sorted on their left edges, subject to the vertical constraint, is:

    [4-10, 1-7, 5-6-9, 2, 3-8]

    Track 1: Nets 4 and 10
    Track 2: Nets 1 and 7
    Track 3: Nets 5, 6 and 9
    Track 4: Net 2
    Track 5: Nets 3 and 8

The Final Solution

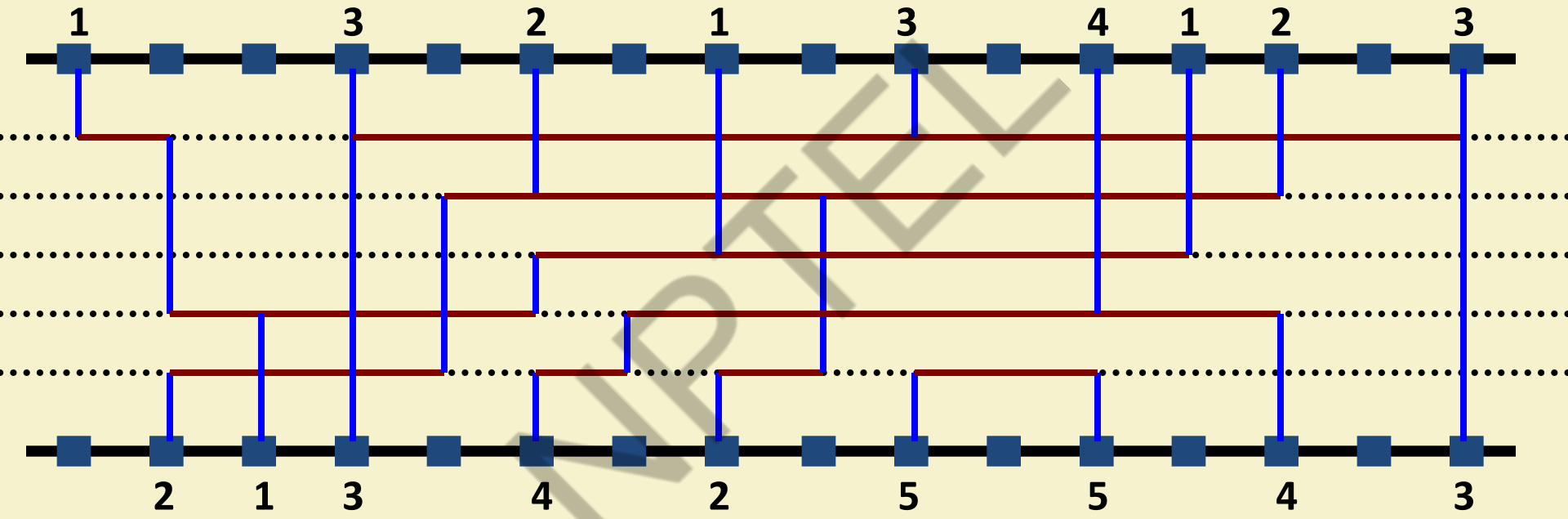


Greedy Channel Router

- The routing algorithms discussed so far route the channel one net at a time.
  - Based on left-edge algorithm or some of its variation.
- The Greedy Channel Router algorithm routes the channel column by column starting from the left.
  - Apply a sequence of greedy but intelligent heuristic at each column.
  - Objective is to maximize the number of tracks available in the next column.
- Can handle problems with cycles in VCG.
  - May need additional columns at the end of the channel.

- Some of the heuristics used:
  - Place all segments column by column, starting from the leftmost column.
  - Connect any terminal to the trunk segment of the corresponding net.
  - Collapse any split net using a vertical segment.
  - Try to reduce the distance between two tracks of same net.
  - Try to move the nets closer to the boundary which contains the next terminal of that net.
  - Add additional tracks if needed.

Channel Routed using a Greedy Router



END OF LECTURE 22

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

Lecture 23: DETAILED ROUTING (PART 4)

PROF. INDRANIL SENGUPTA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Comparison of Two-Layer Channel Routers

	LEA	Dogleg	Net Merge	Greedy
Model	Grid-based	Grid-based	Grid-based	Grid-based
Dogleg	No	Yes	Yes	Yes
Vertical constraint	No / Yes	Yes	Yes	Yes
Cyclic constraint	No	No	No	Yes

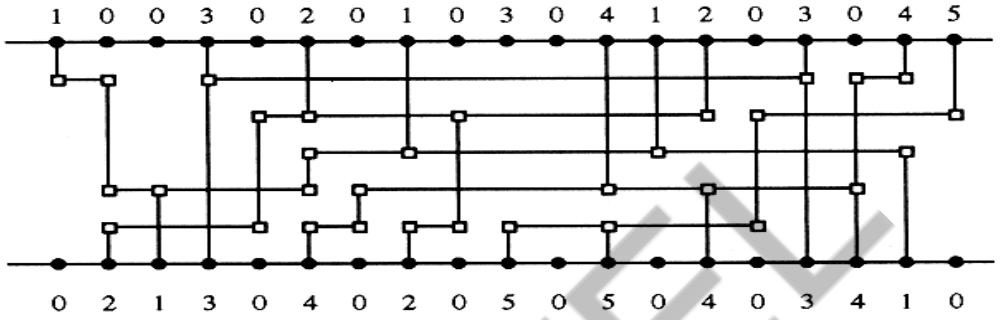
Three-Layer Channel Routing

- Several approaches:
  - Extended Net Merge Channel Router
  - HVH Routing from HV Solution
  - Hybrid HVH-VHV Router

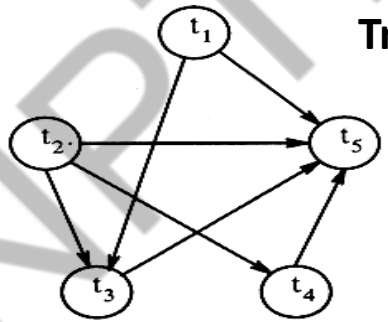
HVH Routing from HV Solution

- Very similar to the Y-K algorithm.
  - Systematically transform a two-layer routing solution into a three-layer routing solution.
  - In Y-K algorithm, nets are merged so that all merged nets forming a composite net are assigned to one track.
  - Here, the composite nets are merged to form super-composite nets.
- Objective:
  - Reduce the number of super-composite nets.

- Two composite nets in a super-composite net can be assigned to different layers on the same track.
- A track-ordering graph is used to find the optimal pair of composite nets to be merged.
  - Vertices represent the composite (tracks) in a given two-layer solution.
  - The directed edges represent the ordering restrictions on pairs of tracks.
    - Composite interval t_i must be routed above composite interval t_j, if there exists a net $N_p \in t_i$ and $N_q \in t_j$, such that N_p and N_q have a vertical constraint.



(a)



(b)

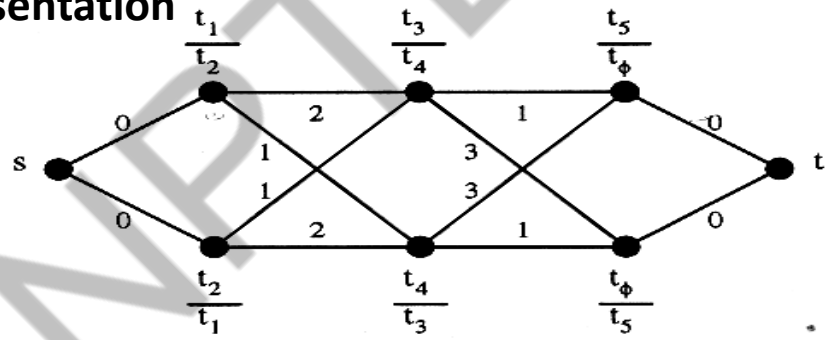
Track ordering graph

Time	P_1	P_2
1	t_1	t_2
2	t_3	t_4
3	t_5	

(c)

An optimal scheduling solution

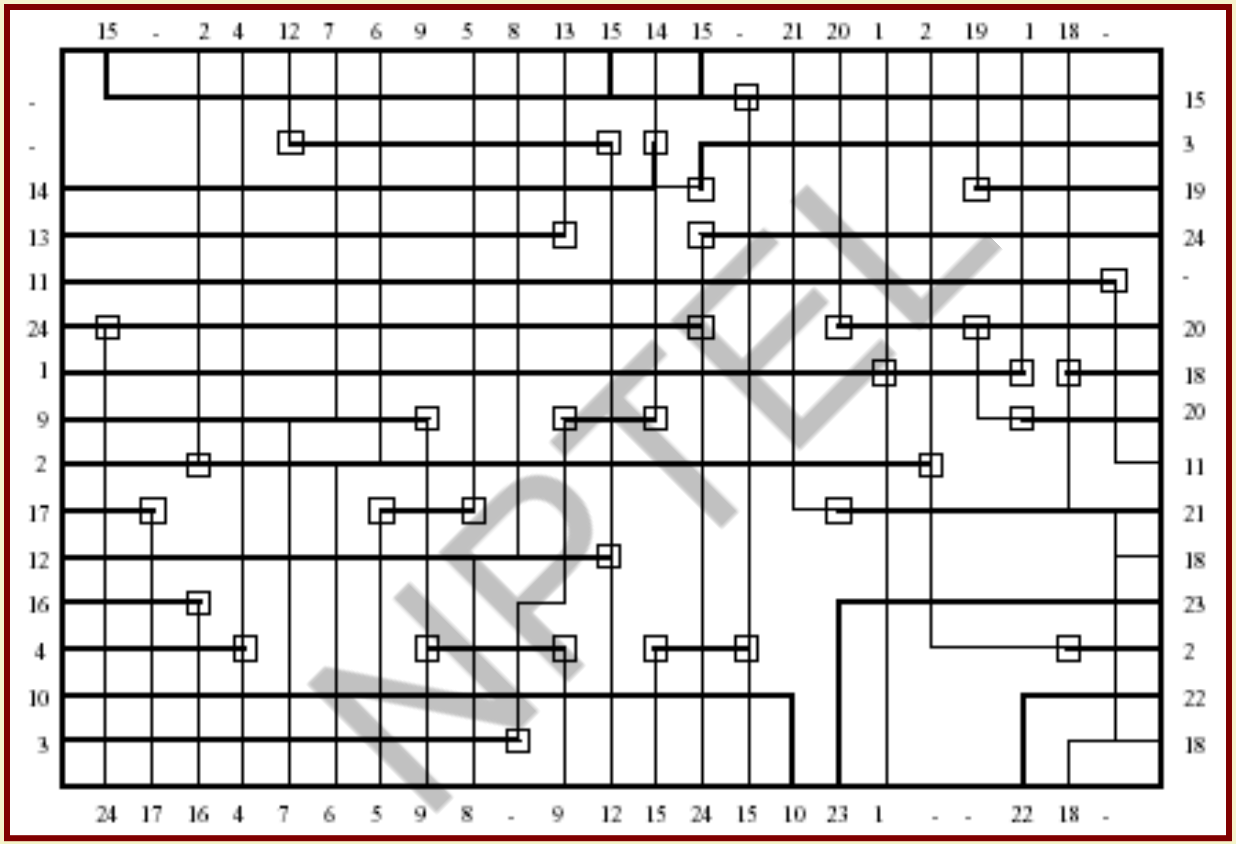
Graph representation



(d)

Switchbox Routing

- A switchbox is a generalization of a channel.
  - Has terminals on all four sides.
- More difficult than channel routing problem.
  - Main objective of channel routing is to minimize the channel height.
  - Main objective of switchbox routing is to ensure that all the nets are routed.
- Classification of algorithms:
  - Greedy router
  - Rip up and reroute routers
  - BEAVER (based on computational geometry)


15 - 2 4 12 7 6 9 5 8 13 15 14 15 - 21 20 1 2 19 1 18 -
- - 14 13 11 24 1 9 2 17 12 16 4 10 3
15 3 19 24 - 20 18 20 11 21 18 23 2 22 18
24 17 16 4 7 6 5 9 8 - 9 12 15 24 15 10 23 1 - - 22 18 -

Summary

- The detailed routing problem is solved by routing the channels and switchboxes.
- Routing results may differ based on the routing model used.
  - Grid-based.
  - Based on assigning layer of different net segments.
- The objectives for routing a channel is to minimize channel density, the length of routing nets, and the number of via's.
- The main objective of channel routing is to minimize total routing area.
- The objective of switchbox routing is to determine the routability.

Over-The-Cell Routing

Introduction

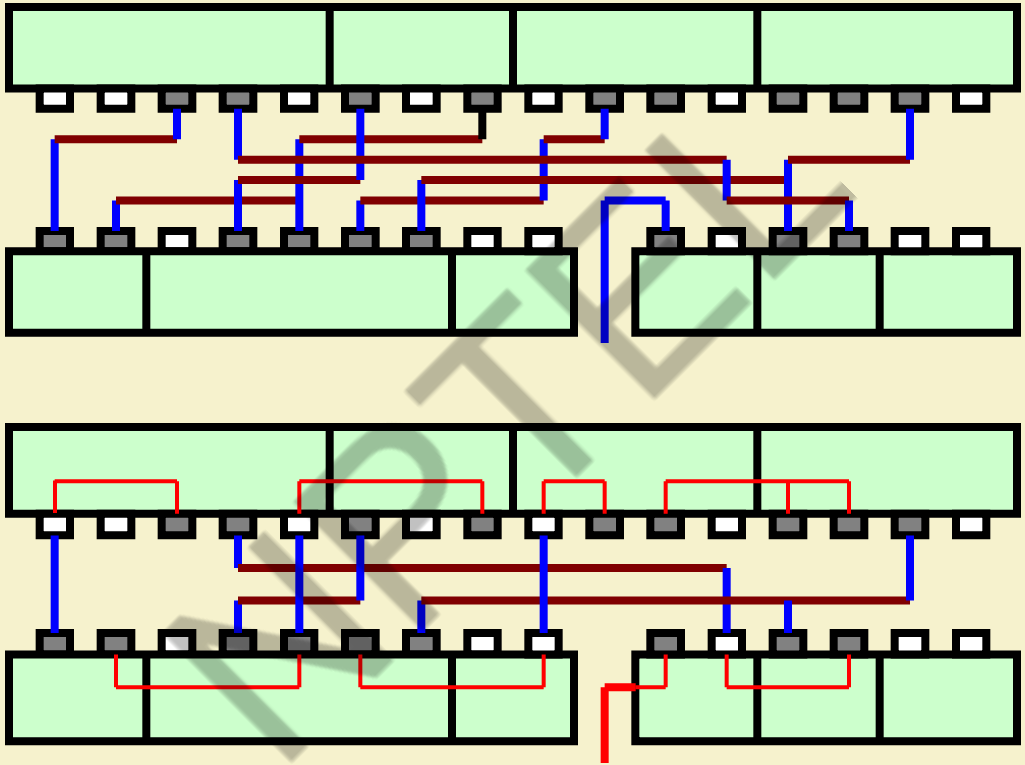
- Used in sophisticated channel routers in standard cell based designs.
- Basic idea:
  - Use of area outside the channel to obtain reduction in channel height.
  - Routing over the cell rows is possible due to limited use of the second and third metal layers.

Basic Steps in OTC Routing

- Step 1: Net decomposition
  - Each multi-terminal net is partitioned into a set of 2-terminal nets (defined based on x-coordinates of their left ends).
- Step 2: Net classification
  - Each net is classified into one of following types:
    - Type 1: There is a vacant terminal directly opposite to one of the terminals of the net.
    - Type 2: There is a vacant terminal between the two terminals of the net.
    - Type 3: None of the above.

- Step 3: Vacant terminal assignment
  - Vacant terminals are assigned to each net depending on its type and weight.
  - Weight of a net is defined as the improvement in channel congestion possible if this net can be routed over the cell.
- Step 4: Over-the-cell routing
  - The selected nets are assigned exact geometric paths for routing in the area over the cells.
- Step 5: Channel segment assignment & routing
  - Selects the best net segments to be routed in the channel.
  - Route them using any available channel router.

Example



Greedy channel router

OTC routing

END OF LECTURE 23

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

Lecture 24: CLOCK DESIGN (PART 1)

PROF. INDRANIL SENGUPTA

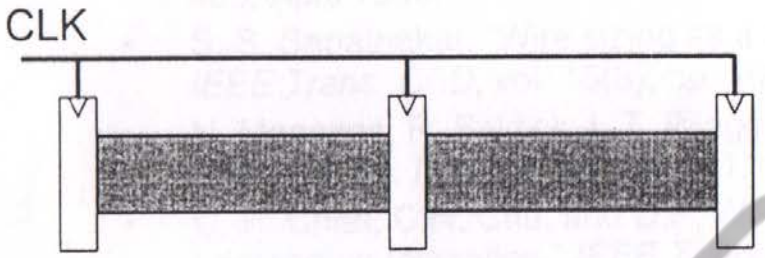
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Clock Design Issues

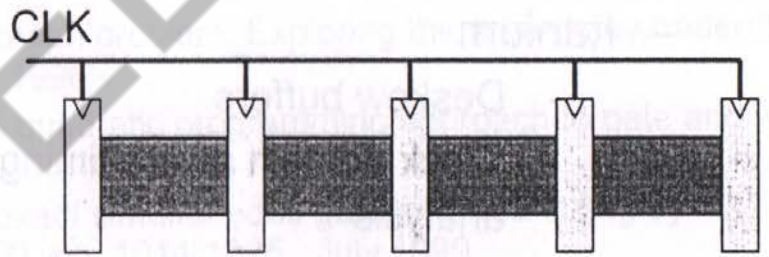
- Clocking
  - Basic issues in synchronization
  - Skew and jitter
  - Max/min delay constraints – setup and hold times
  - Factors affecting skew and jitter

Clocking

- Most chips are synchronous
  - Sequencing all activities on the chip is governed by a single reference clock.
  - The clock signal is used to synchronize the storage elements (flip-flops and latches) on the chip.
- Most chips are highly pipelined
  - A typical instruction is processed through several stages before the results are made available at the output.
  - Pipeline improves the throughput at the expense of increased latency.
  - Aggressive pipelining leads to higher frequency of operation.



2-stage pipeline



4-stage pipeline

Performance and Clocking

- Processor performance is measured by:

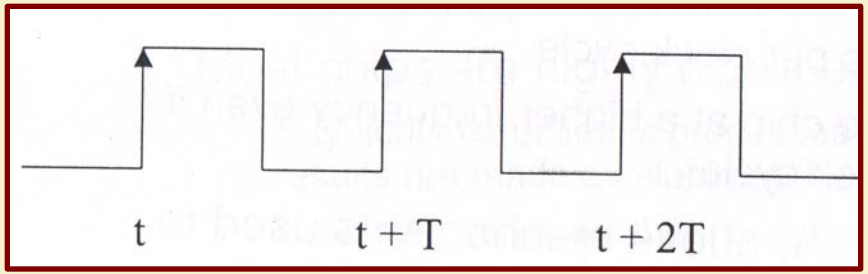
  Execution time = IC x CPI x CCT

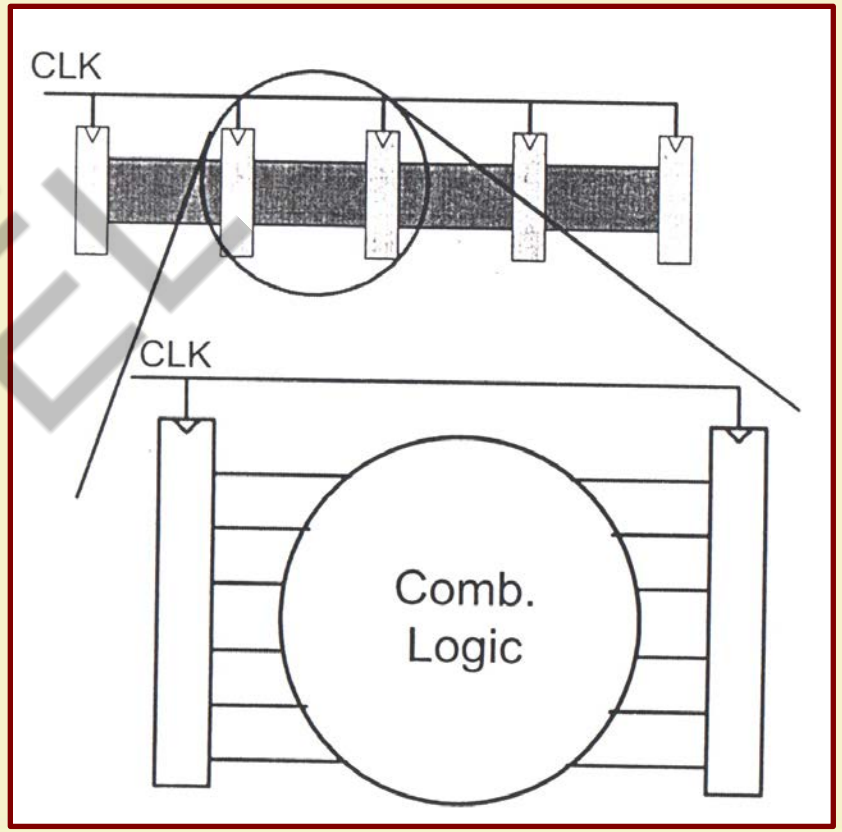
  IC: Instruction Count, CPI: Cycles per Instruction, CCT: Clock Cycle Time.

- Hence, there are three ways to improve performance:
  - *Reduce IC*: depends on instruction set architecture (ISA)
  - *Reduce CPI*: depends on ISA as well as pipeline architecture
  - *Reduce CCT*: depends on logic design as well as fabrication technology
- Here, we focus on clocking issues (i.e. reduction of CCT).

Edge-triggered Clocking

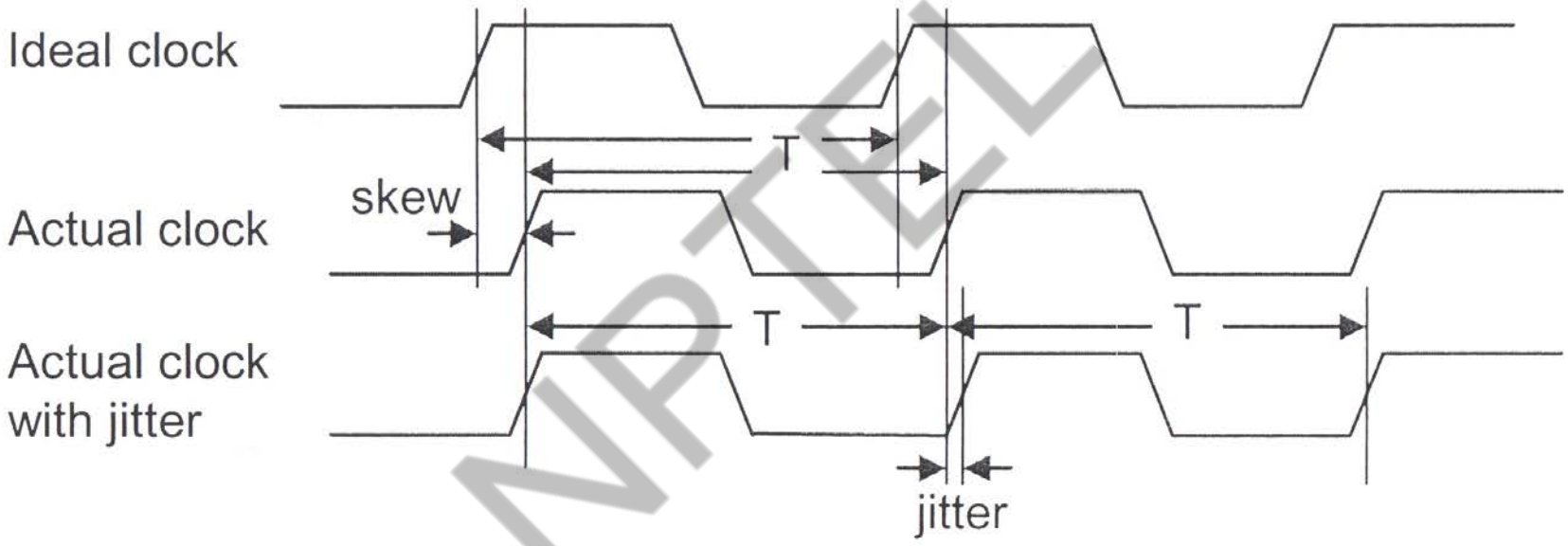
- Edge-triggered operation
  - Data leaving at time t must arrive at the next flip-flop *one setup time before* t+T, where T is the clock period.
  - May be positive or negative edge-triggered.
- Clocking relationships are relatively simple:
  - Delay from each input flip-flop to each output flip-flop of a combinational block should be less that ($T - t_{setup}$).
- Ideally, the clock signal should arrive *all* flip-flops at *exactly* the same time.





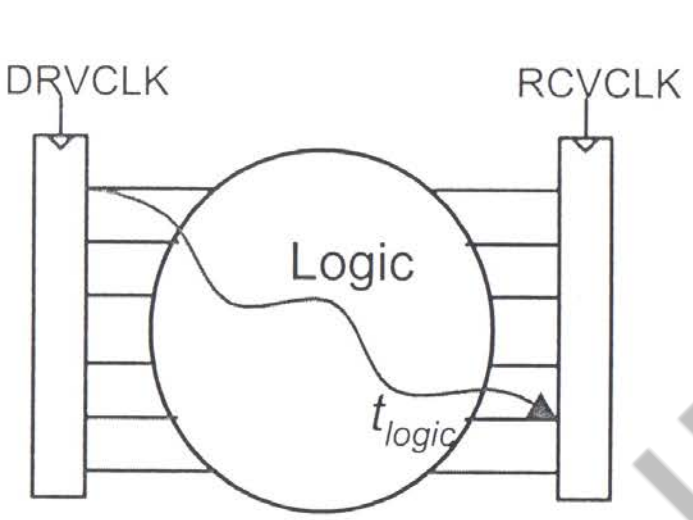
Clock Distribution: Skew and Jitter

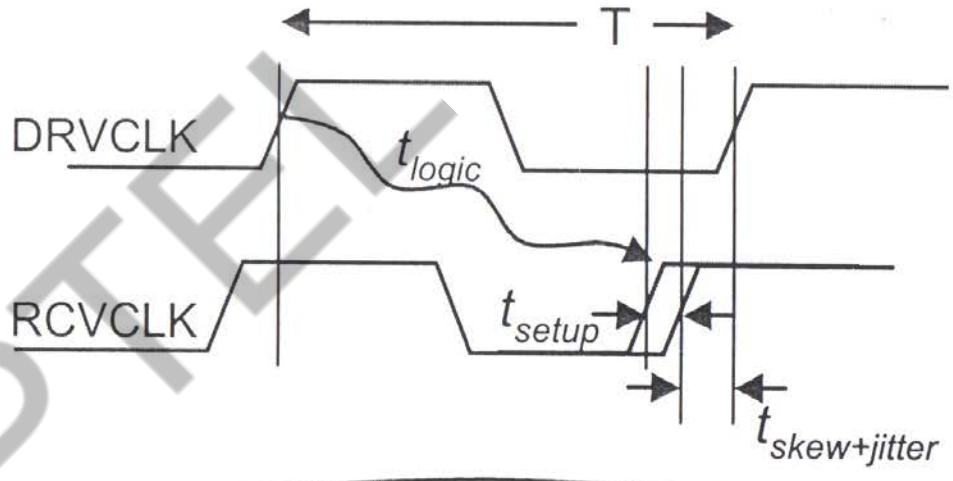
- The clock provides a common reference signal for the chip.
- The clock signal is distributed over a large area that may lead to several physical manifestations of the same signal.
- Two primary effects of this physical distribution:
  - **Skew**: Maximum delay difference between any two leaf nodes (also called clock nodes) on the clock network.
  - **Jitter**: Variation in the delay from cycle to cycle at the same clock node.


Ideal clock
T
skew
Actual clock
T
T
Actual clock
with jitter
jitter

Max Delay Constraint: SETUP Times

- Logic evaluation begins at the rising edge of DRVCLK.
- For correct operation the signal must propagate through the logic and reach the receiving flip-flop driven by RCVCLK.
- All flip-flops have a setup time requirement:
  - Minimum time the signal needs to be stable before it can be captured by the flip-flop.
- Also the skew between DRVCLK and RCVCLK as well as jitter must be taken into account.

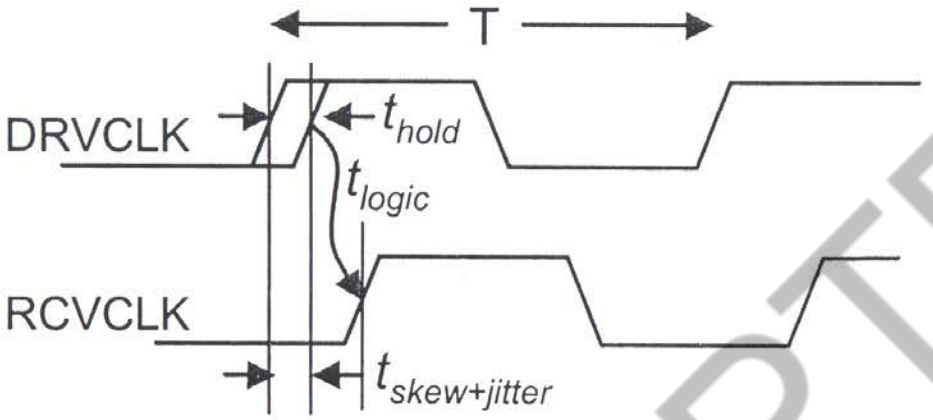




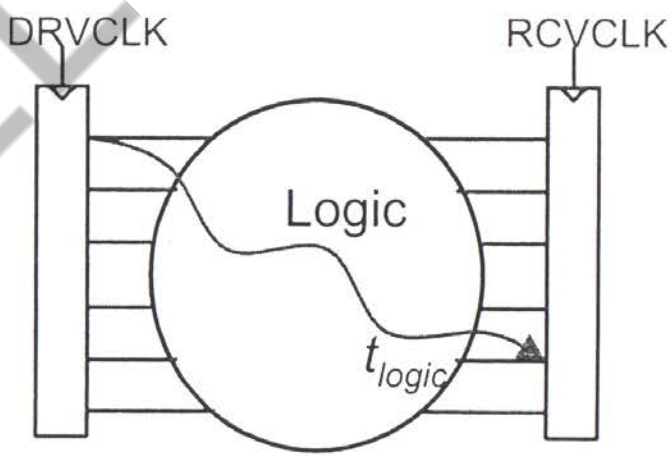
$$t_{logic} + t_{setup} + t_{skew} + t_{jitter} \leq T$$

Min Delay Constraints: HOLD times

- All storage elements have to hold their output signal for a minimum period of time.
  - If not, the following logic block may not respond correctly to this signal value.
- The signal should not race through the logic and get captured by rising edge of the receiving flip-flop for the same cycle.
  - It may be noted that this requirement does not relate to the cycle time.
- Major problem for high-performance processors where the number of gate delays in a pipeline is scaling lower.



$$t_{logic} + t_{hold} \geq t_{skew} + t_{jitter}$$



Quantitative View of Skew and Jitter

- The maximum delay constraint has to be satisfied:

$$t_{logic} + t_{setup} + t_{skew} + t_{jitter} \leq T$$

or, $t_{logic+setup} + t_{skew+jitter} \leq T$

- Hence, the maximum operating frequency with skew and jitter is

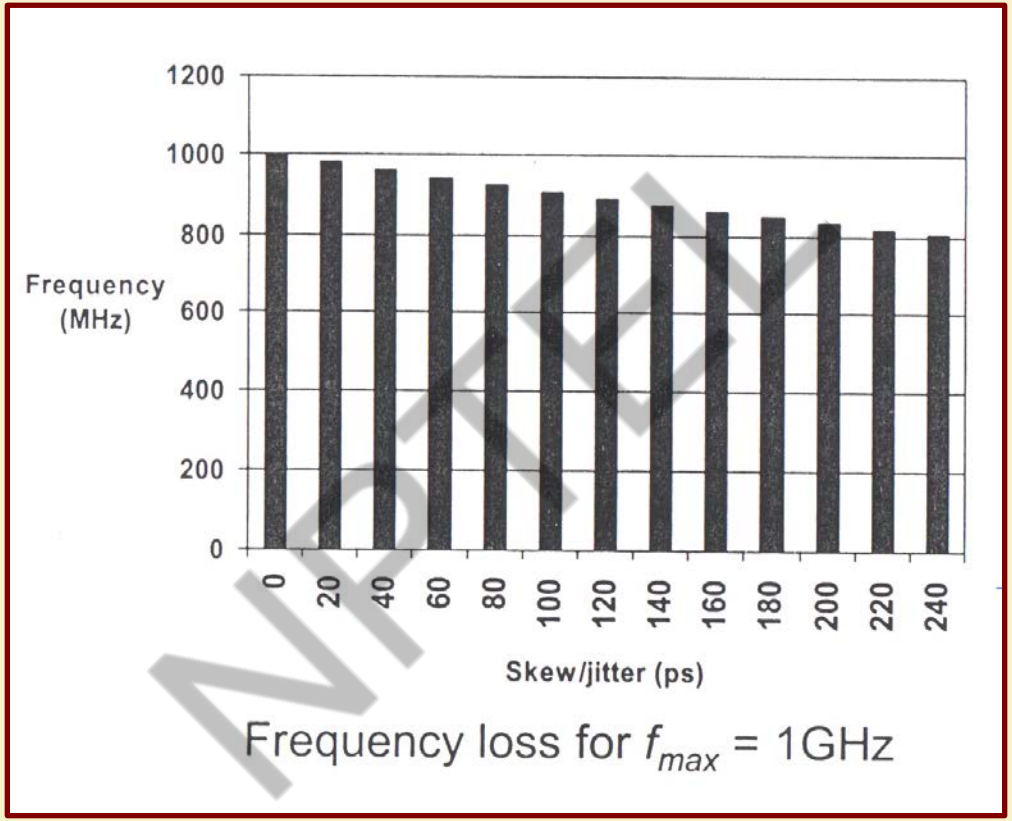
$$f_{max\text{-}skew} = 1 / (t_{logic+setup} + t_{skew+jitter})$$

- In the absence of any skew or jitter,

$$f_{max} = 1 / t_{logic+setup}$$

- Frequency cost due to skew and jitter

$$\Delta f / f_{max} = (f_{max} - f_{max\text{-}skew}) / f_{max} = t_{skew+jitter} / (t_{logic+setup} + t_{skew+jitter})$$



Frequency loss for f_{max} = 1GHz

END OF LECTURE 24

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

Lecture 25: CLOCK DESIGN (PART 2)

PROF. INDRANIL SENGUPTA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

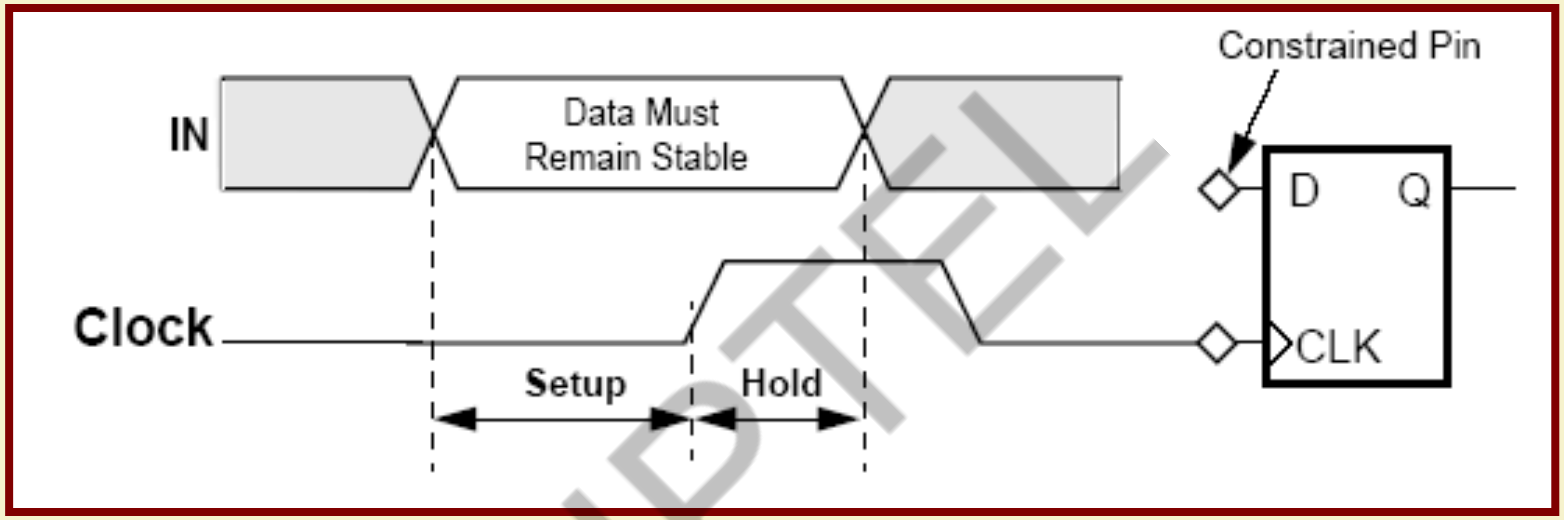
Factors Affecting Skew

- The primary factors affecting the skew of the clock signal at the leaf node:
  - Electrical symmetry of the network
  - Process variation of the clock buffers and interconnects
  - Power supply variation of the clock buffers
  - Capacitive coupling to any clock signal
  - Temperature variation across the die
- Variation effects are the most difficult to control.

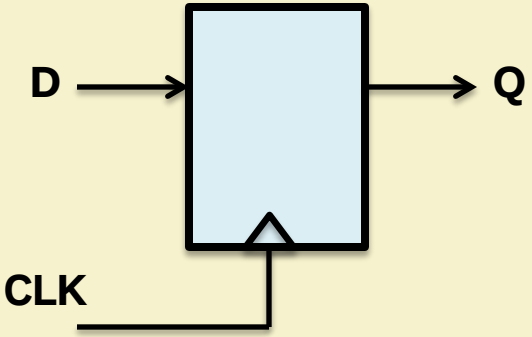
Some Terminologies

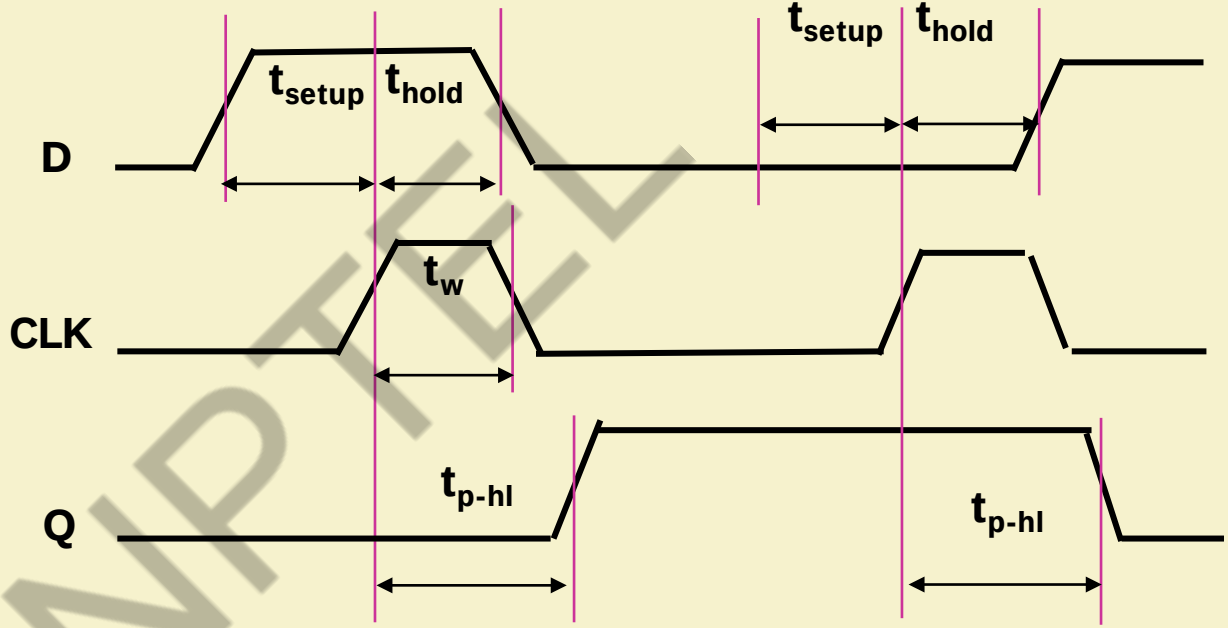
- Clock width t_w:
  - Minimum time duration for which the clock signal needs to be high in order that the flip-flops it feeds work properly.
- Setup time t_{setup} or t_{su}:
  - Amount of time the input to a flip-flop must be stable *before* the clock transitions high (for positive-edge triggered), or transitions low (for negative-edge triggered).

- Hold time t_{hold} or t_h:
  - Amount of time the input to a flip-flop must be stable *after* the clock transitions high (for positive-edge triggered), or transitions low (for negative-edge triggered).
- Propagation delays $t_{p\text{-}lh}$ and $t_{p\text{-}hl}$:
  - Delay between clocking event (low-to-high or high-to-low transition) and change in the output.


Constrained Pin
IN
Data Must
Remain Stable
D
Q
Clock
CLK
Setup
Hold

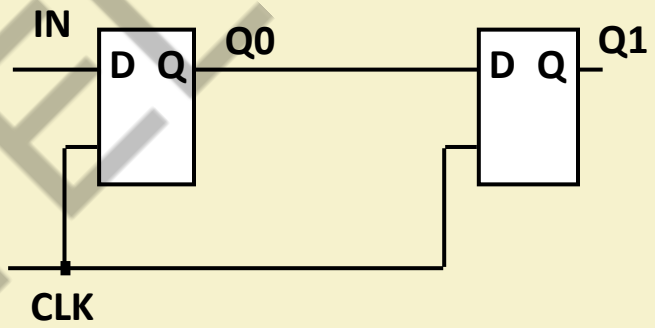
D flip-flop

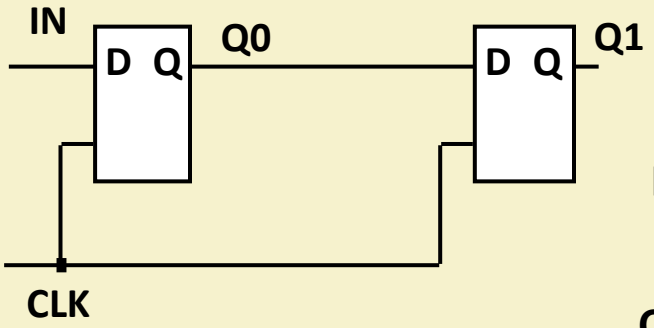




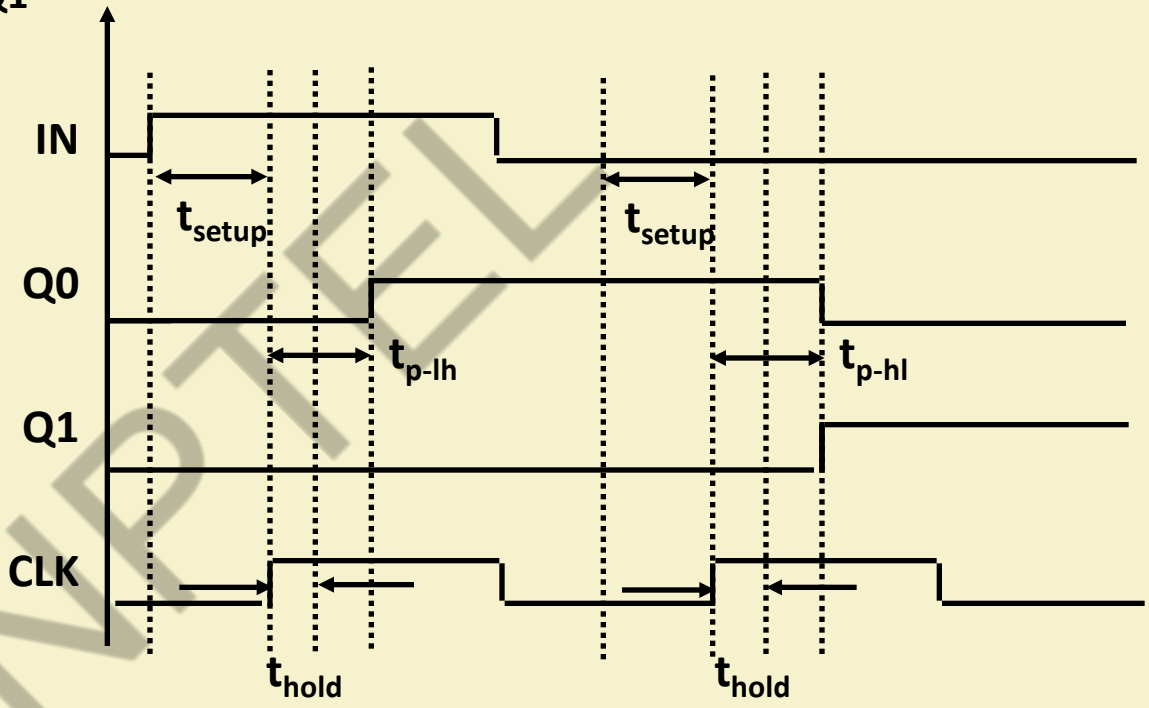
Cascading Flip-flops

- Suppose that flip-flop propagation delay exceeds hold time.
- Second stage can commit its input before Q0 changes.



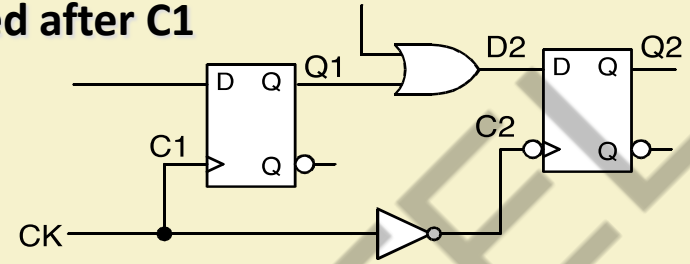


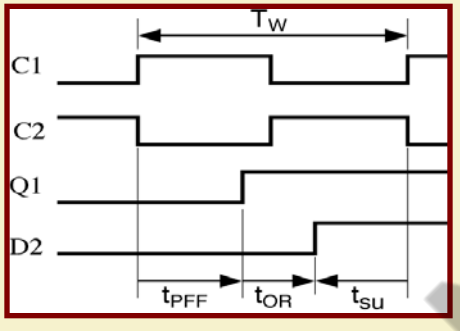
The correct scenario

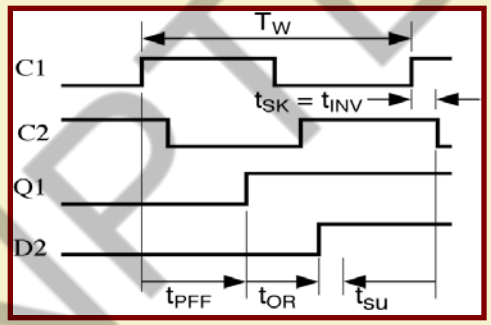


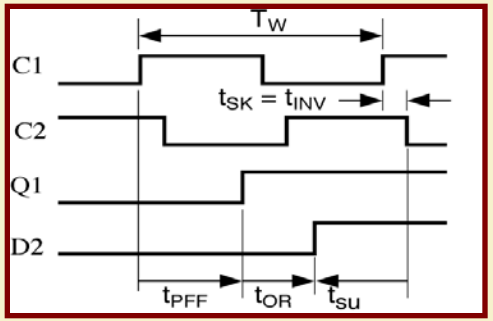
Example (Effect of clock skew on clock rate)

(a) Clock C2 skewed after C1









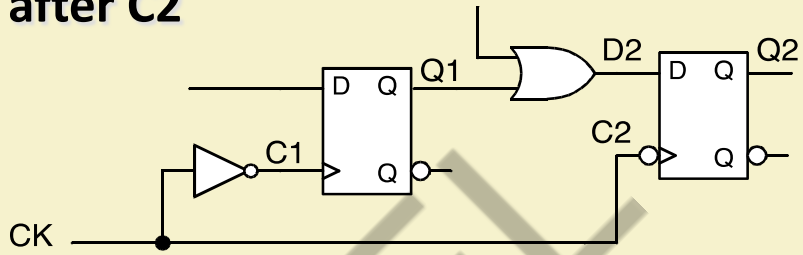
$T_W \geq \max T_{PFF} + \max t_{OR} + t_{su}$

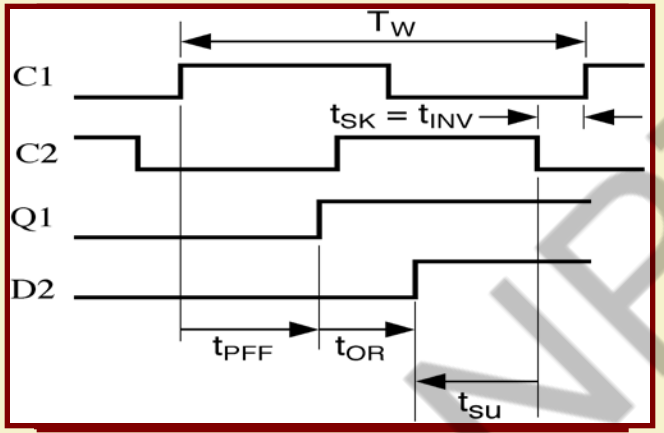
(if clock not skewed, i.e., $t_{INV} = 0$)

$T_W \geq \max T_{PFF} + \max t_{OR} + t_{su} - \min t_{INV}$

(if clock skewed, i.e., $t_{INV} > 0$)

(b) Clock C1 skewed after C2





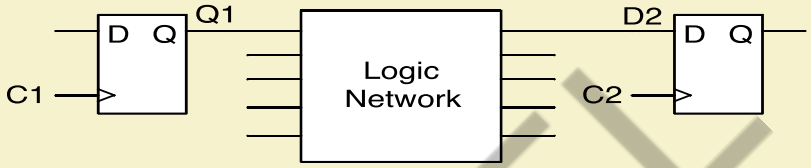
$T_W \geq \max T_{PFF} + \max t_{OR} + t_{su}$

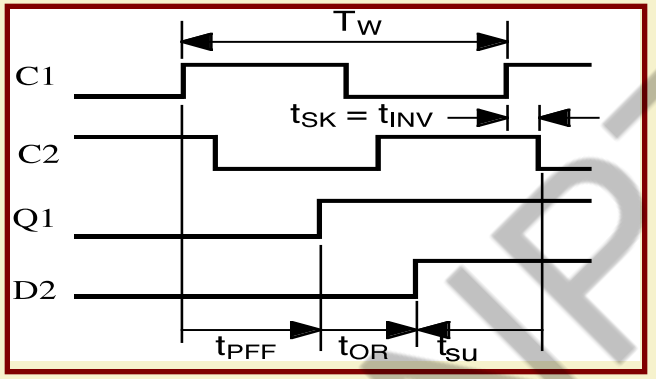
(if clock not skewed, i.e., $t_{INV} = 0$)

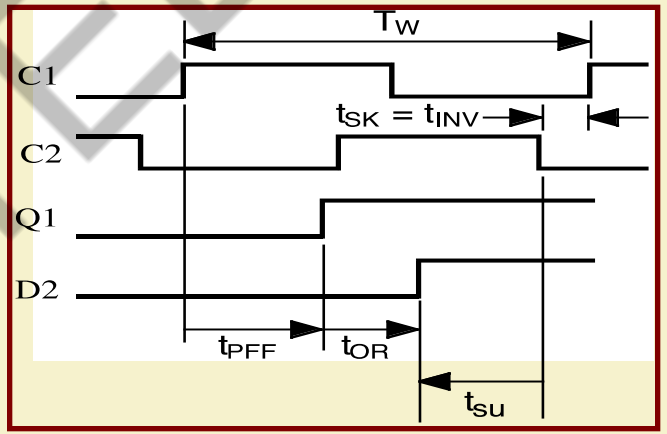
$T_W \geq \max T_{PFF} + \max t_{OR} + t_{su} + \max t_{INV}$

(if clock skewed, i.e., $t_{INV} > 0$)

Summary of maximum clock frequency calculations



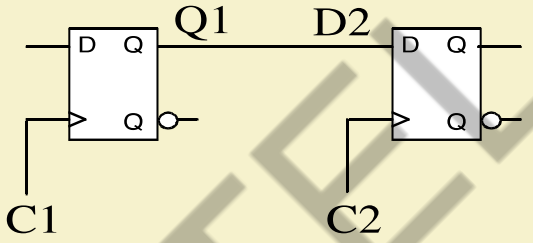




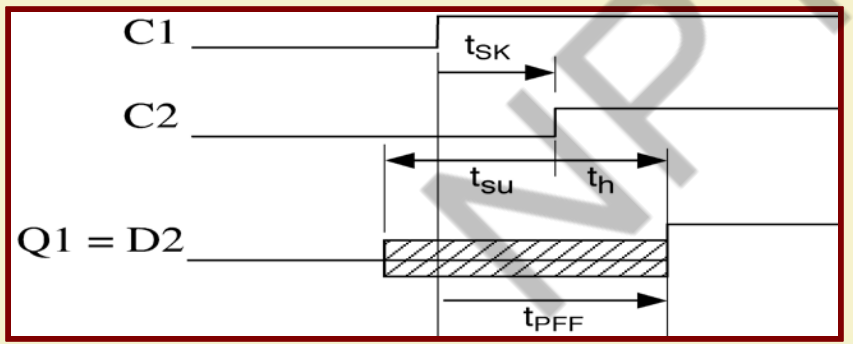
C2 skewed after C1: $T_W \geq \max T_{PFF} + \max t_{NET} + t_{su} - \min t_{INV}$

C2 skewed before C1: $T_W \geq \max T_{PFF} + \max t_{NET} + t_{su} + \max t_{INV}$

How much skew between C1 and C2 can be tolerated in the following circuit?



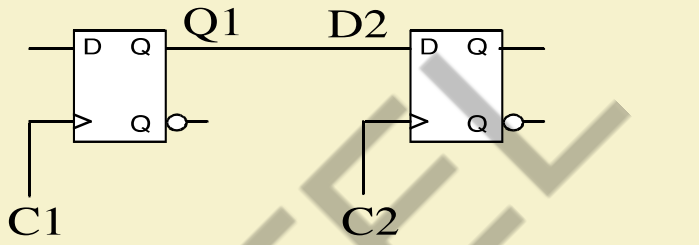
Case 1: C2 delayed after C1

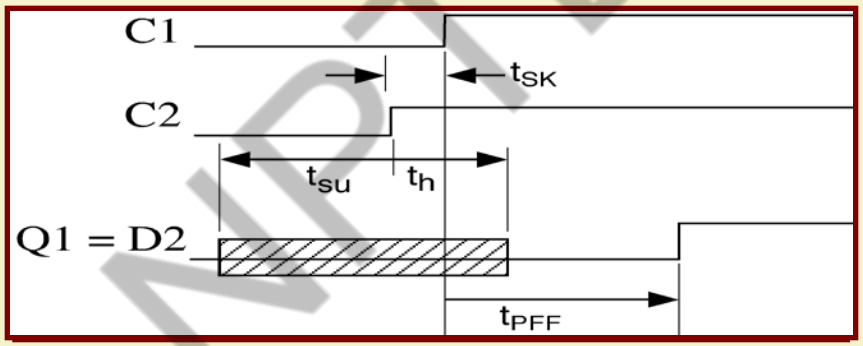


$t_{PFF} > t_h + t_{SK}$

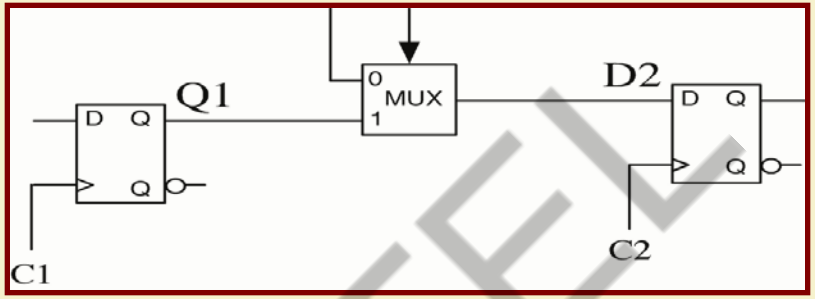
$t_{SK} < \min t_{PFF} - t_h$

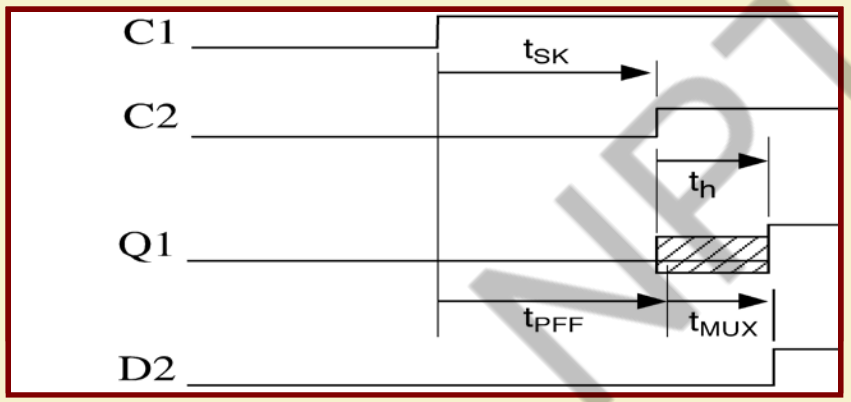
Case 2: C1 delayed from C2





How does additional delay between the flip-flops affect the skew calculations?

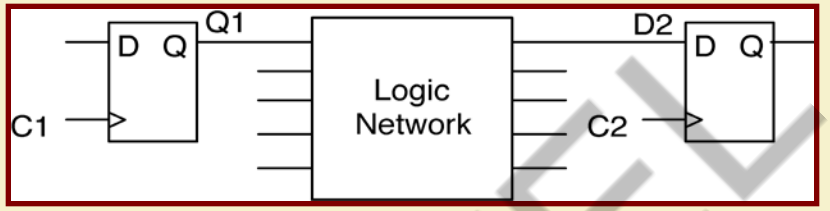


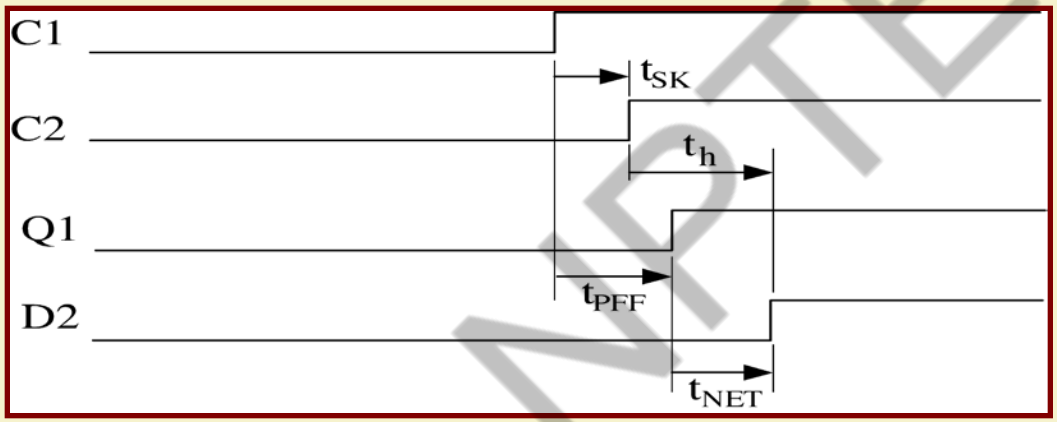


$t_{SK} \leq \min t_{PFF} - t_h$

$t_{SK} \leq \min t_{PFF} + \min t_{MUX} - t_h$

Summary of allowable clock skew calculations





$t_{SK} + t_h \leq t_{PFF} + t_{NET}$

$t_{SK} \leq \min t_{PFF} + \min t_{NET} - t_h$

END OF LECTURE 25

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

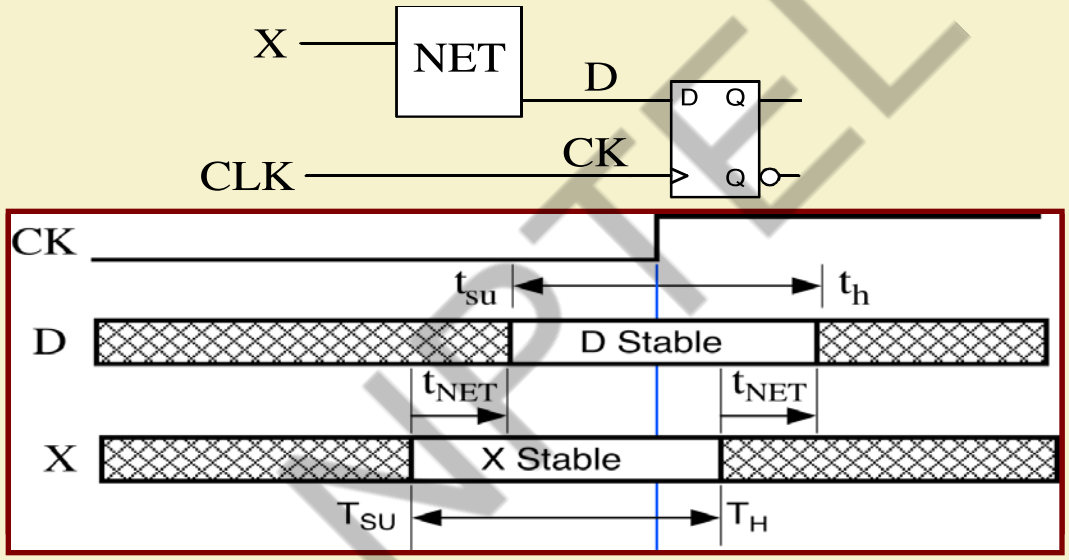
Lecture 26: CLOCK DESIGN (PART 3)

PROF. INDRANIL SENGUPTA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Global Setup Time, Hold Time, Propagation Delay

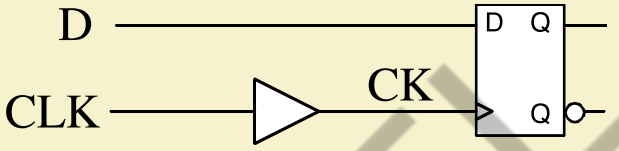
- Global setup and hold times (***data delayed***)

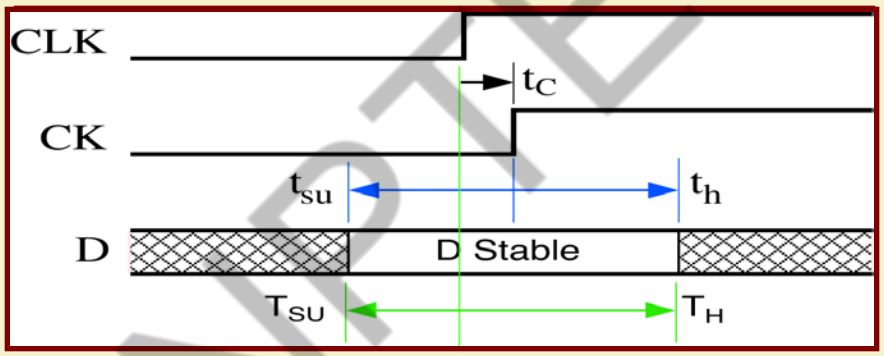


$T_{SU} = t_{su} + \max t_{NET}$

$T_H = t_h - \min t_{NET}$

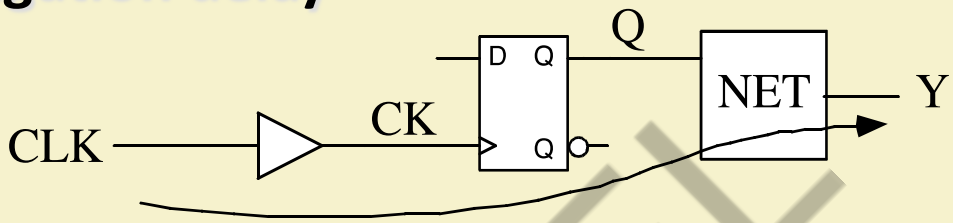
- Global setup & hold time (***clock delayed***)

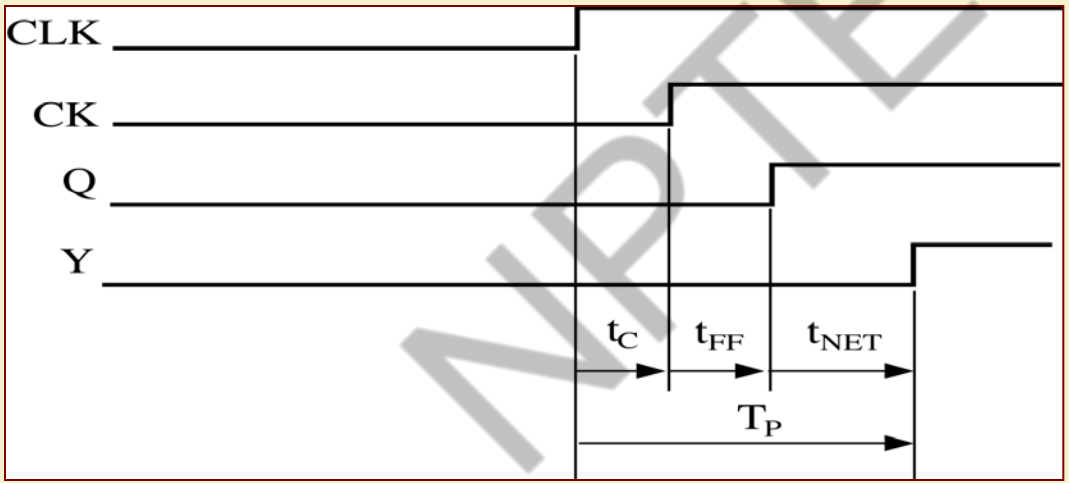




$T_{SU} = t_{su} - \min t_C$ $\quad\quad$ $T_H = t_h + \max t_C$

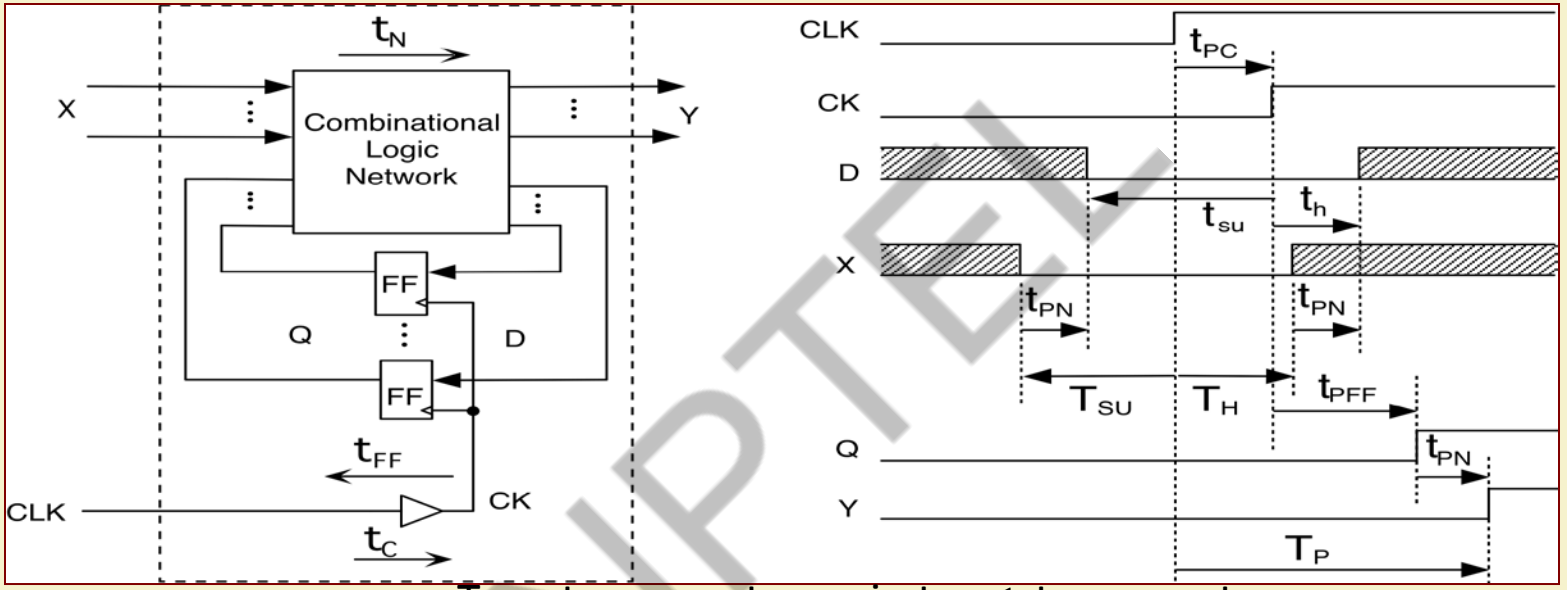
Global propagation delay





$$T_P = t_C + t_{FF} + t_{NET}$$

Summary of global timing parameters



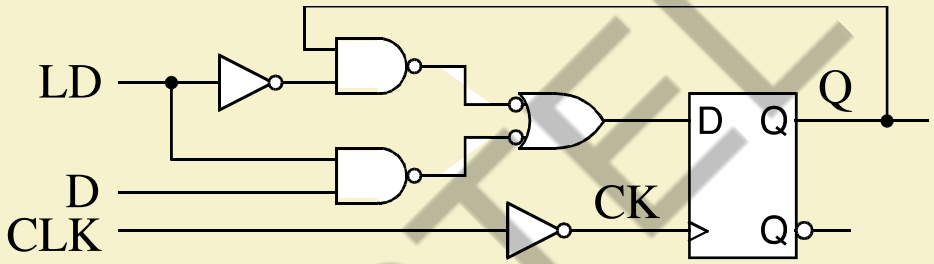
$T_{SU} = t_{su} + \max t_{PN} - \min t_{PC} \leq t_{su} + \max t_{PN}$

$T_H = t_h + \max t_{PC} - \min t_{PN} \leq t_h + \max t_{PC}$

$T_P = t_{PFF} + t_{PN} + t_{PC}$

Example:

- Find T_{SU} and T_H for input signal LD relative to CLK.

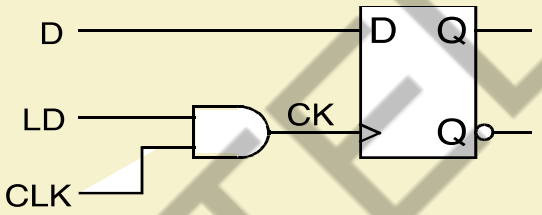


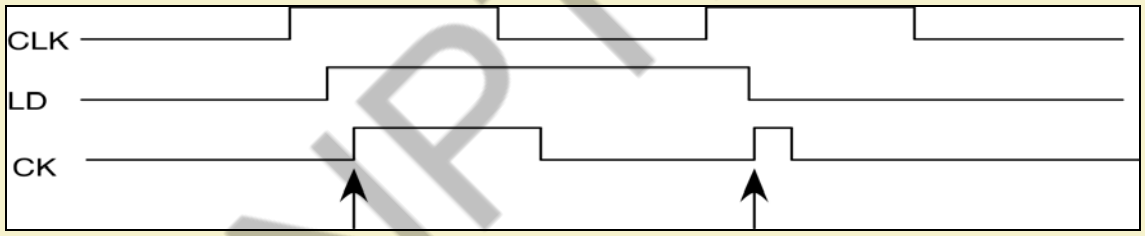
$$T_{SU} = t_{su} + \max t_{NET} - \min t_C$$
$$= t_{su} + \max t_{INV} + \max t_{NAND} + \max t_{NAND} - \min t_{INV}$$

$$T_H = t_h - \min t_{NET} + \max t_C$$
$$= t_h - \min t_{NAND} - \min t_{NAND} + \max T_{INV}$$

Register load control (gating the clock)

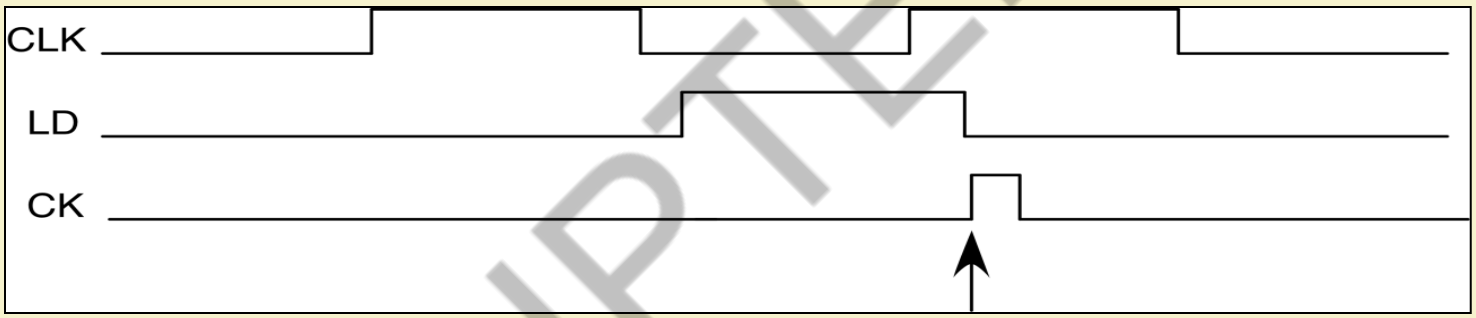
- A very bad way to add a load control signal LD to a register that does not have one is shown:



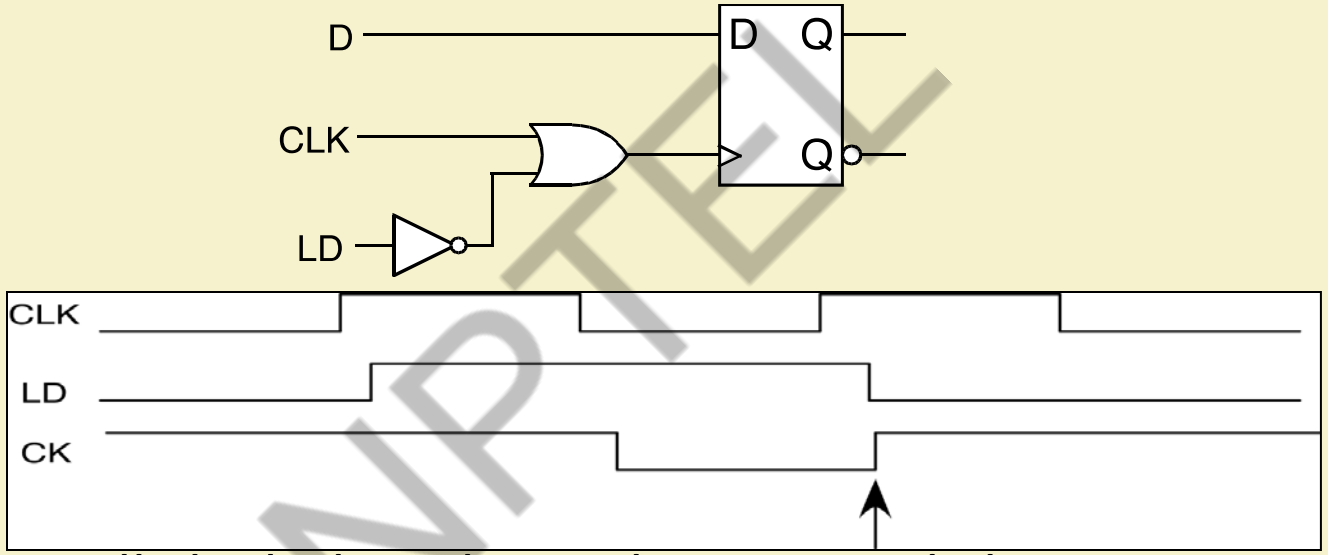


- The flip-flop sees two rising edges and will trigger twice. The only one we want is the second one.

- If LD was constrained to only change when the clock was low, then the only problem would be the clock skew.

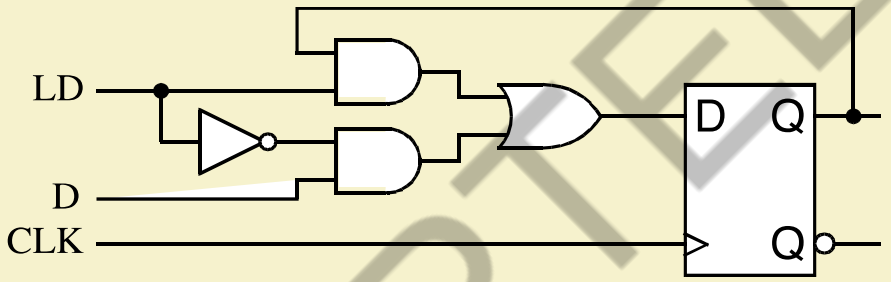


- If gating the clock is the only way to control the loading of registers, then use the following:



- There is still clock skew, but at least we only have one triggering edge.

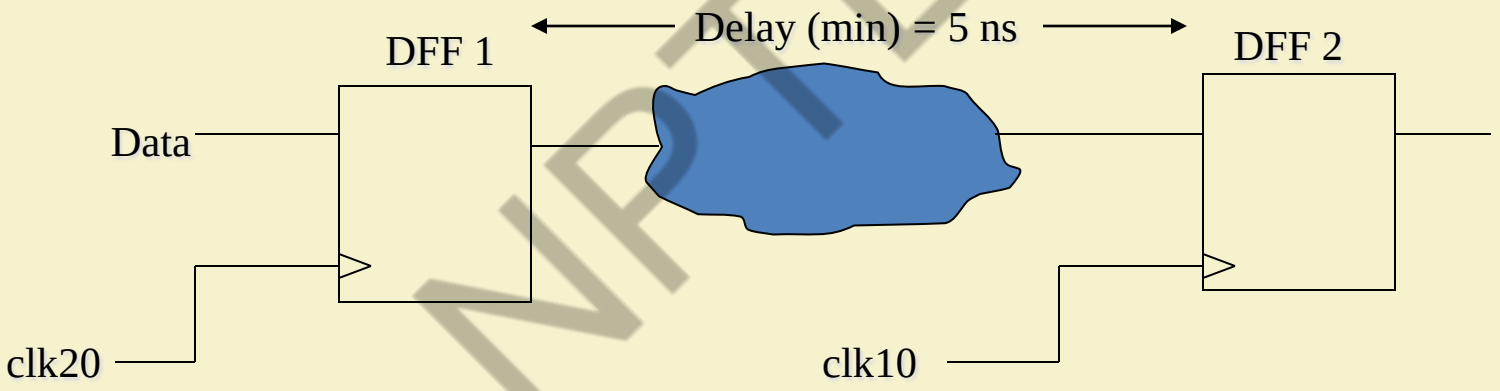
- The best way to add a LD control signal is as follows:



Detecting timing violations – CASE 1

(a) Hold time for clocks is 1.5 ns

Determine if there are any timing violations in this design

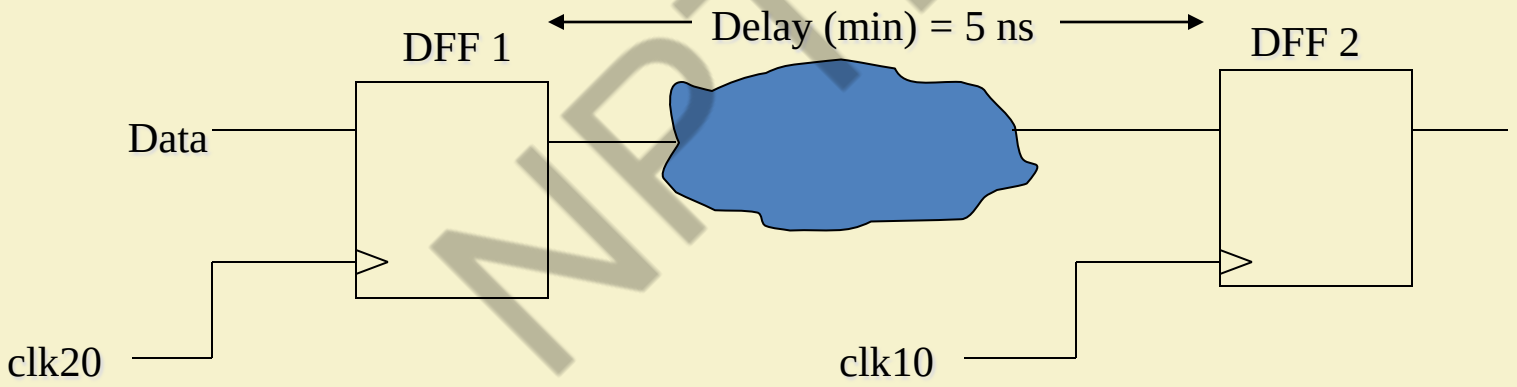


Detecting timing violations – CASE 2

(a) Hold time for clocks is 1.5 ns

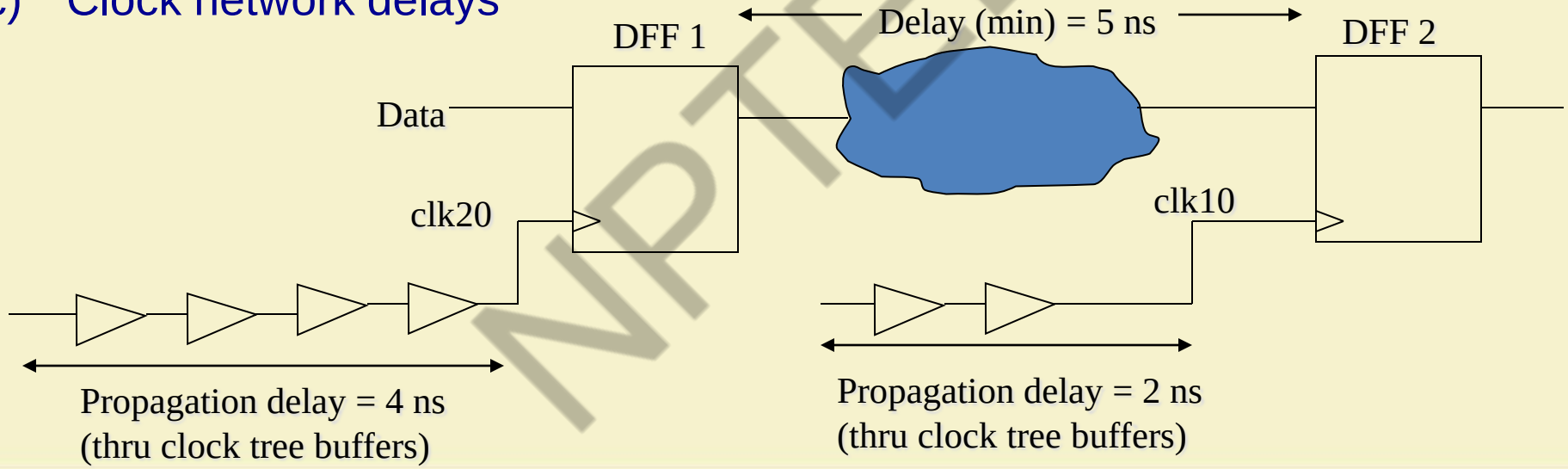
(b) Clock skew of 3.72 ns between clk20 and clk10

Determine if there are any timing violations in this design



Detecting timing violations – CASE 3

a) Hold time for clocks is 1.5 ns
b) Clock skew of 3.72 ns between clk20 and clk10
c) Clock network delays



Metrics for Clock Design

- Power
- Skew
- Jitter
- Flexibility of the distribution
  - Can clock loads be changed throughout the design cycle?
- Duty cycle variation
- Reliability
  - Electromigration due to large currents

The metrics have to be traded off against each other.

- An important part of clock design is to identify the most important metrics.

Clock Power Dissipation

- The clock signal switches twice every cycle.
  - It consumes a major portion of the power for a high-performance processor (approx. 40%).
  - A low-power clock tree implies a low-power chip.
- Power consumption has two major components:
  - The dynamic component: $C_L V_{dd}^2 f$, where C_L includes the capacitive load of the entire clock network including
    - The wires of the distribution
    - The clock buffers
    - The flip-flop loads at the leaf nodes

  - The short-circuit power when both the PMOS and NMOS transistors of the buffer are ON when a signal is in transition.
- The overall chip power can be reduced by switching off certain portions of the chip during inactive periods.
  - Called *Clock Gating*
  - Another aspect of clock design

Solving the Skew Problem

- Careful design of the clock buffers and wires.
  - Make the design as electrically symmetric as possible.
  - Should be robust to process and temperature variations, and changes in clock load.
- De-skewing circuitry
  - Use active circuitry to detect skew and compensate using feedback.
- Clock-domain aware timing synthesis
  - Timing tools that distinguish between local and global skew.

END OF LECTURE 26