ECE 6473
Lecture 9
Date: 11/16/2015

Shaloo Rakheja

Assistant Professor

ECE, NYU

Topics

- Interconnects
- Physical design
- Clocking

Reading Materials

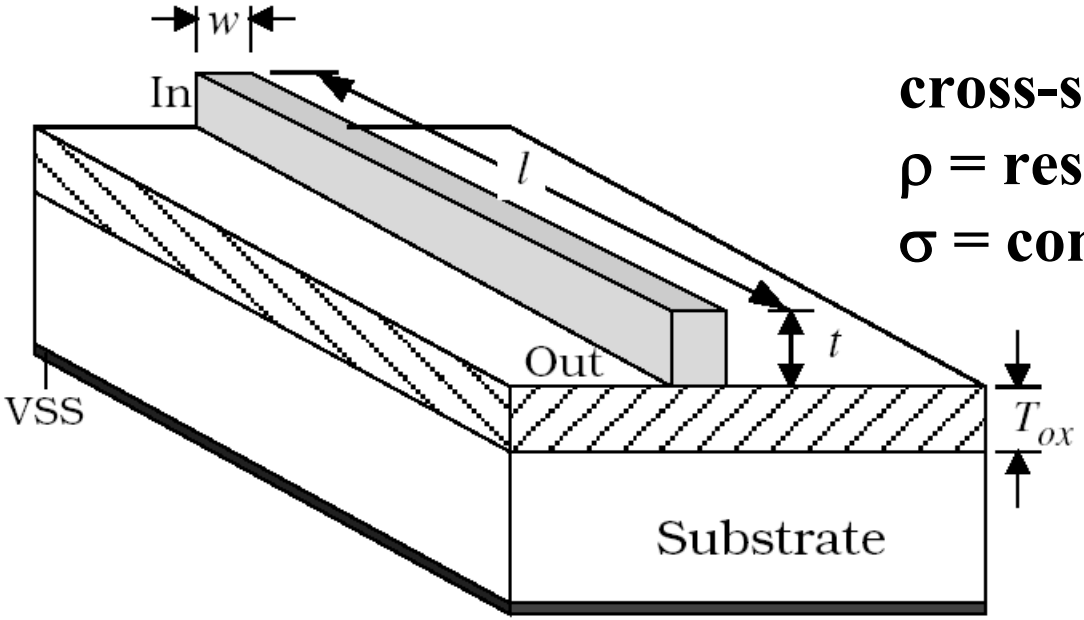
- Chapters 9 & 10: Digital Integrated Circuits, Rabaey
- Chapters 5 & 15: Introduction to VLSI Circuits and Systems, Uyemura
- Lecture + hand-written notes posted on course website

Interconnect Resistance

- **Resistance:**

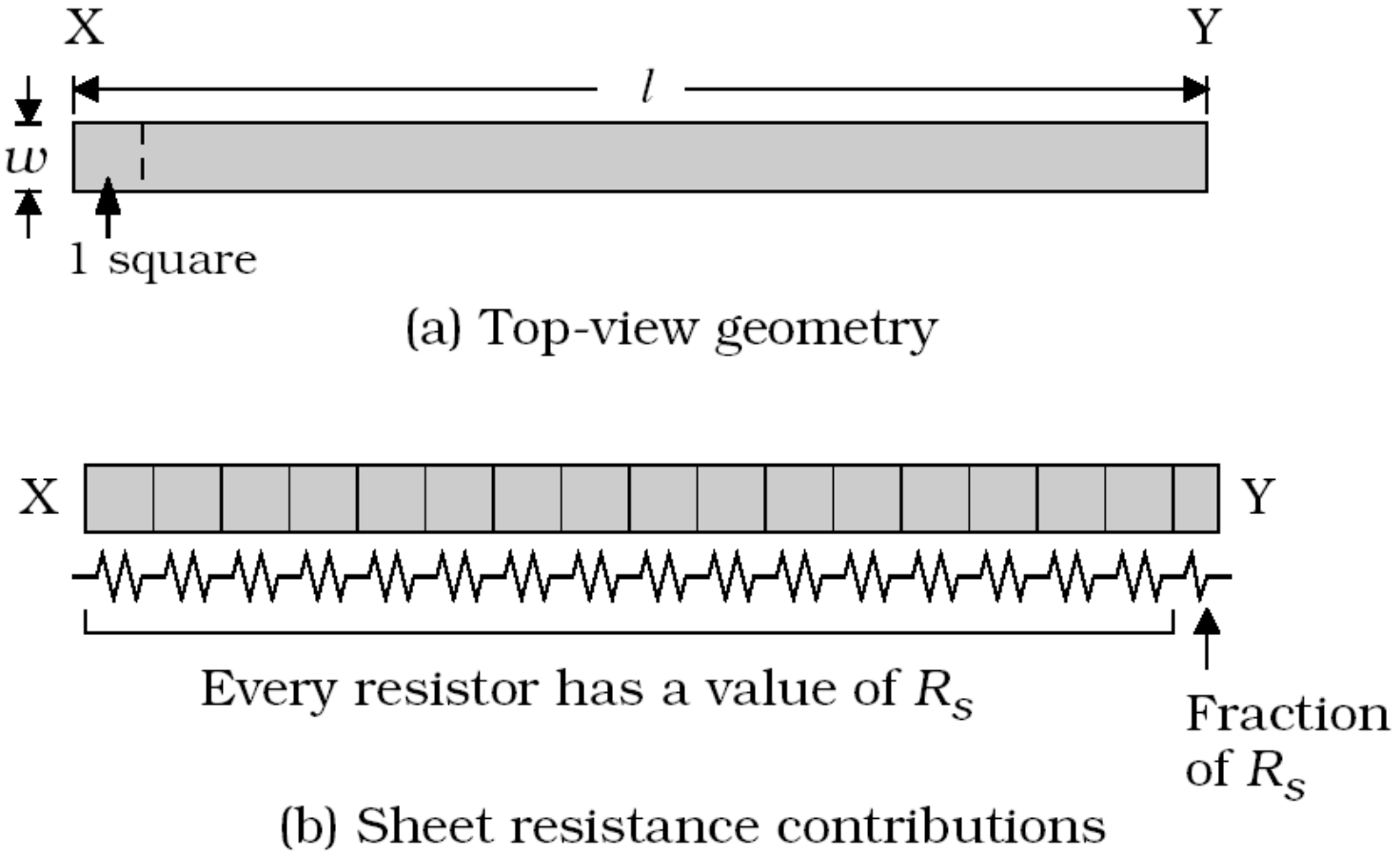
  $R_{line} = \rho(l/A) = (\rho/t) \times (l/w) = (1/\sigma t) \times (l/w) = R_s \times (l/w)$

- **R_s = sheet resistance = Resistance of a wire of $(l = w)$**
- **R_s represents the resistance of a square region with top dimensions of w by w**
- **$R_{line} = (R_s / w) \times l = r \times l$, *r = resistance per unit length***



cross-sectional area: A = t × w,
ρ = resistivity,
σ = conductivity

Interconnect Resistance – contd.

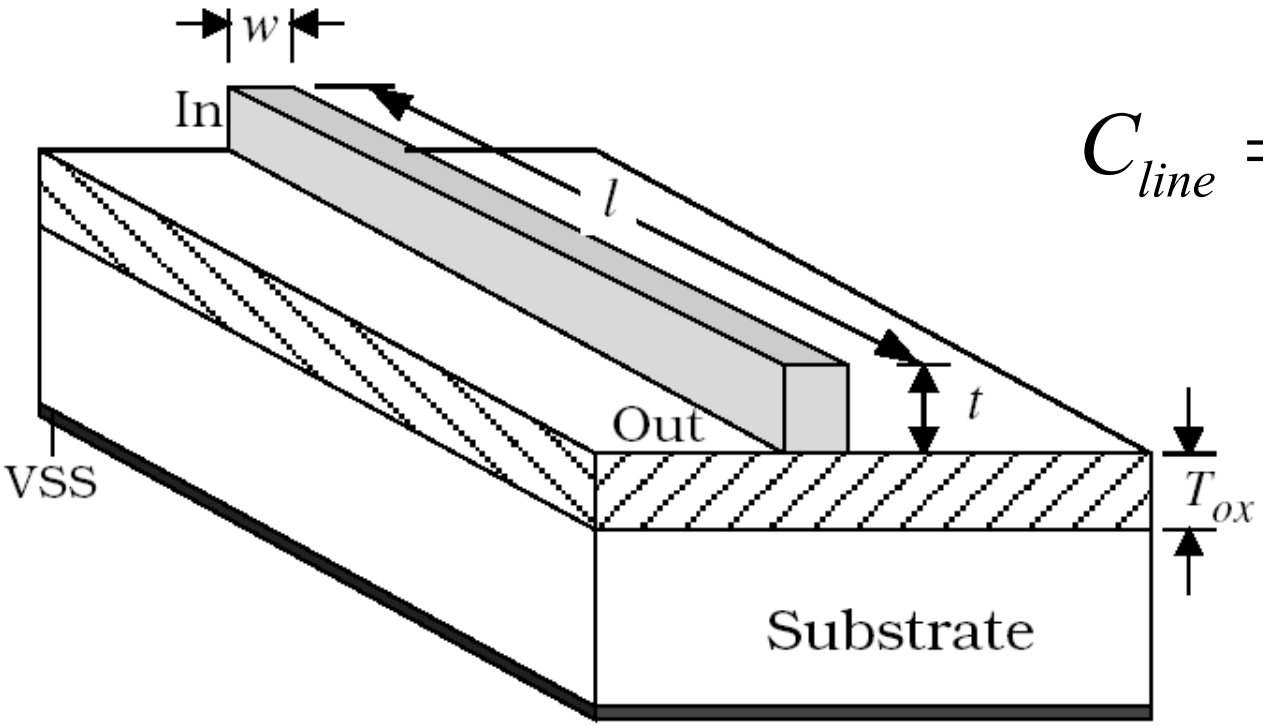


- ***$R_{line} = R_s$ x (n) = R_s x (l/w)***
  'n' = number of squares from X to Y

Capacitance Interconnect line

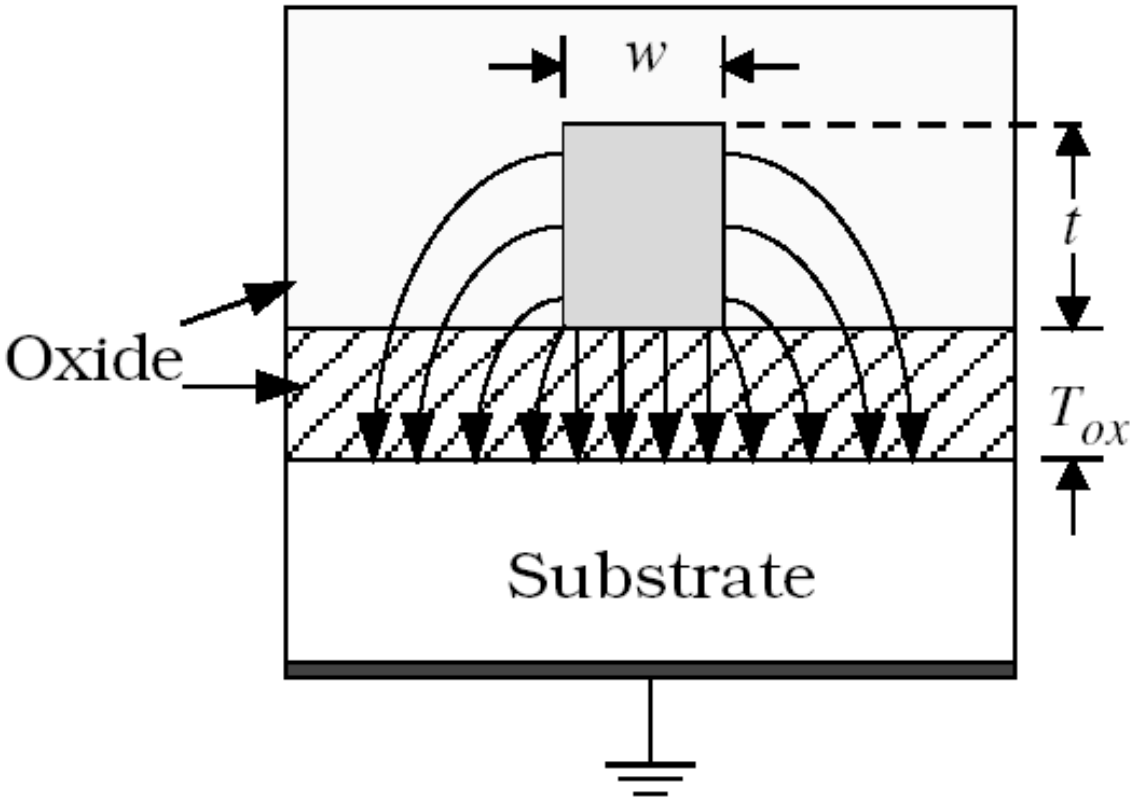
Self-capacitance:

Only parallel plate component:



$$C_{line} = \varepsilon_{ox} \frac{lw}{T_{ox}}$$

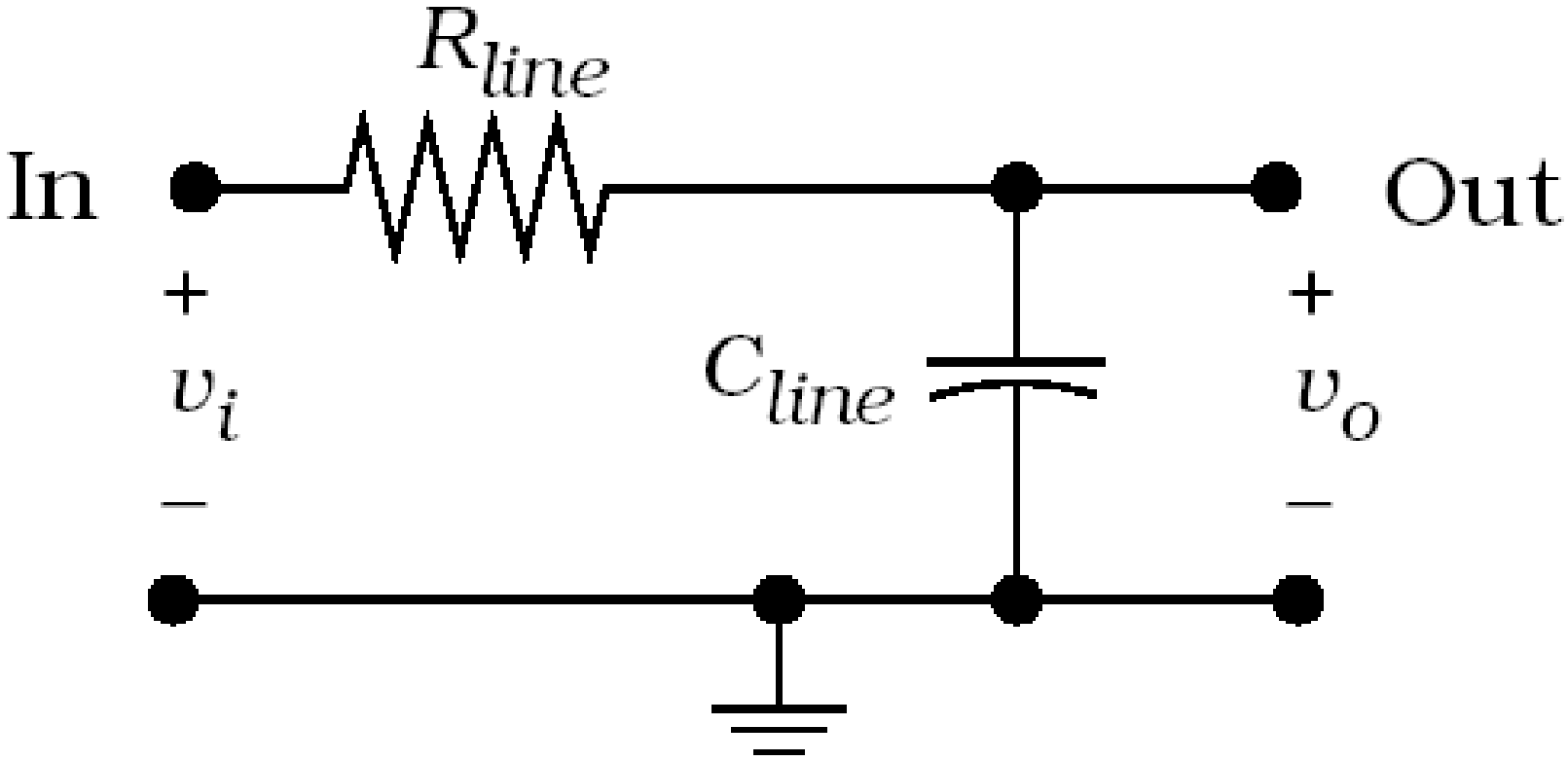
Interconnect Cap.: Fringing Field



$$c = \varepsilon_{ox}\left[1.15\left(w/T_{ox}\right) + 2.8\left(t/T_{ox}\right)^{0.222}\right] F/cm$$

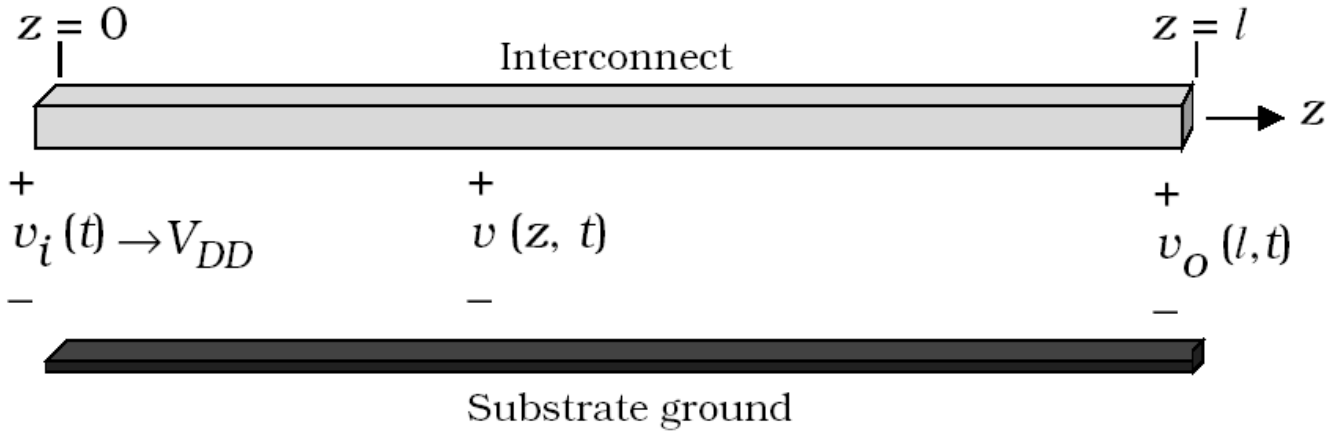
$$C_{line} = c \times l$$

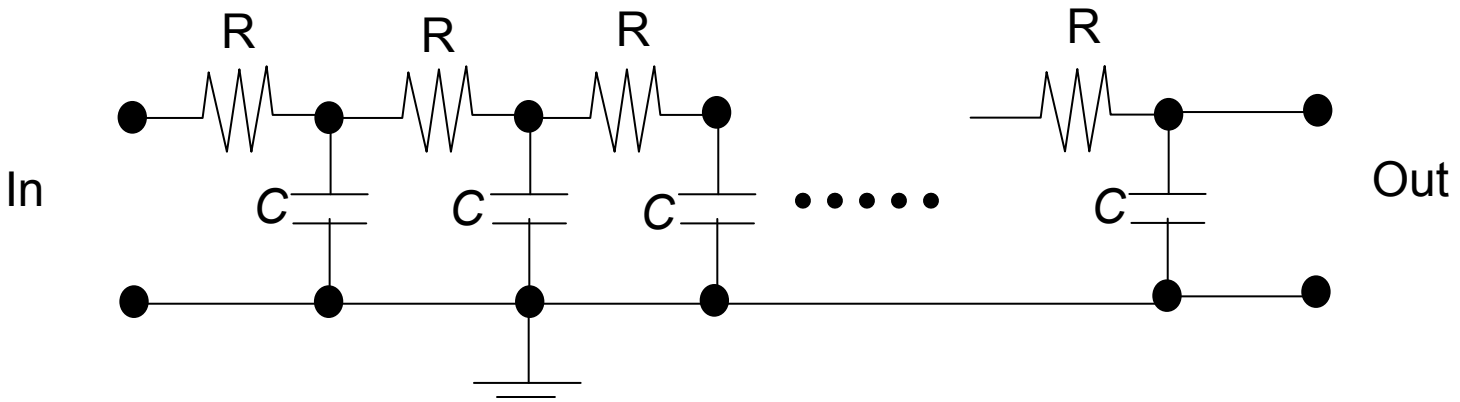
Circuit Model for Interconnects: Simplest Approach



$$wire\ delay \propto \tau = R_{line}C_{line}$$

Physical model of an interconnect line.





Physical model of an interconnect line.

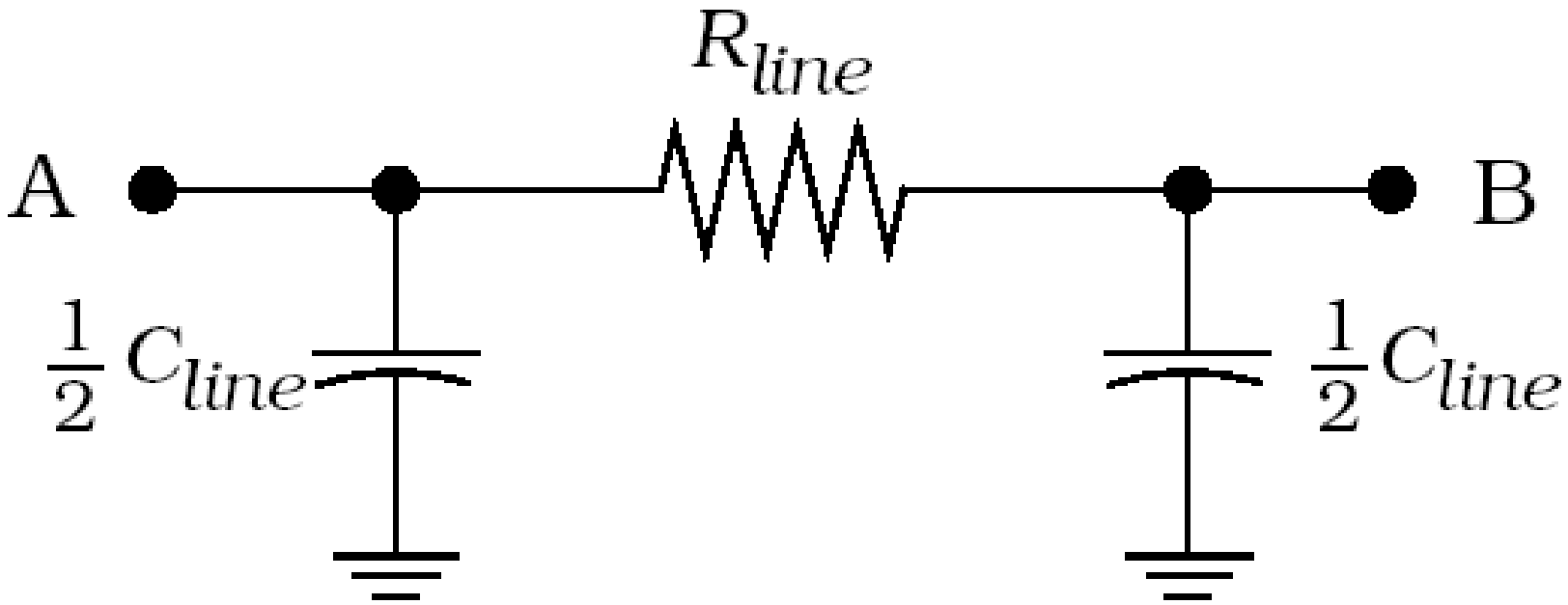
$$\tau = C\left[mR\right] + C\left[(m-1)R\right] + C\left[(m-2)R\right] + \ldots + CR$$

$$= RC\left[1 + \ldots + m\right] = RC\left[\frac{m(m+1)}{2}\right]$$

$$= \frac{R_{line}}{m}\frac{C_{line}}{m}\left[\frac{m(m+1)}{2}\right] = \left[\frac{(m+1)}{2m}\right]R_{line}C_{line}$$

$$as\ m \to \infty,\ we\ get : \tau = \frac{1}{2}R_{line}C_{line} = \frac{1}{2}rc\left(l^2\right)$$

Simple RC interconnect model.

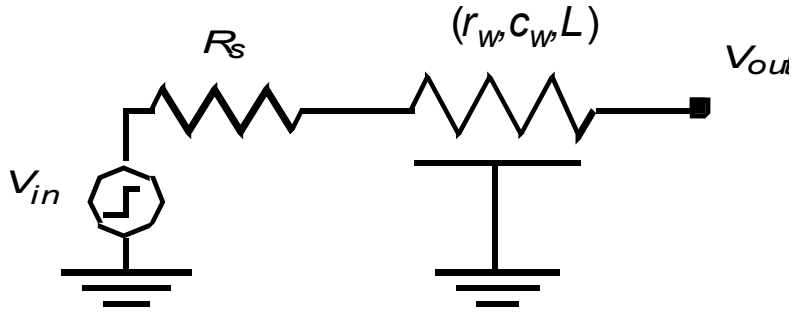


Wire Delay

- Lumped model provides a pessimistic view of wire delay
- Wire delay increases quadratically with the wire length
- Rule of thumb:
  - Wire delay need to be consider when it is larger than the delay of the gate

$$t_{RC} = \frac{1}{2}(0.69)rc\left(L_{crit}^2\right) = t_{gate} => L_{crit} = \sqrt{t_{gate}/0.38rc}$$

RC vs Lumped C wire delay



- Lumped C:

$$t_C = R_s C_w$$

- RC Model:

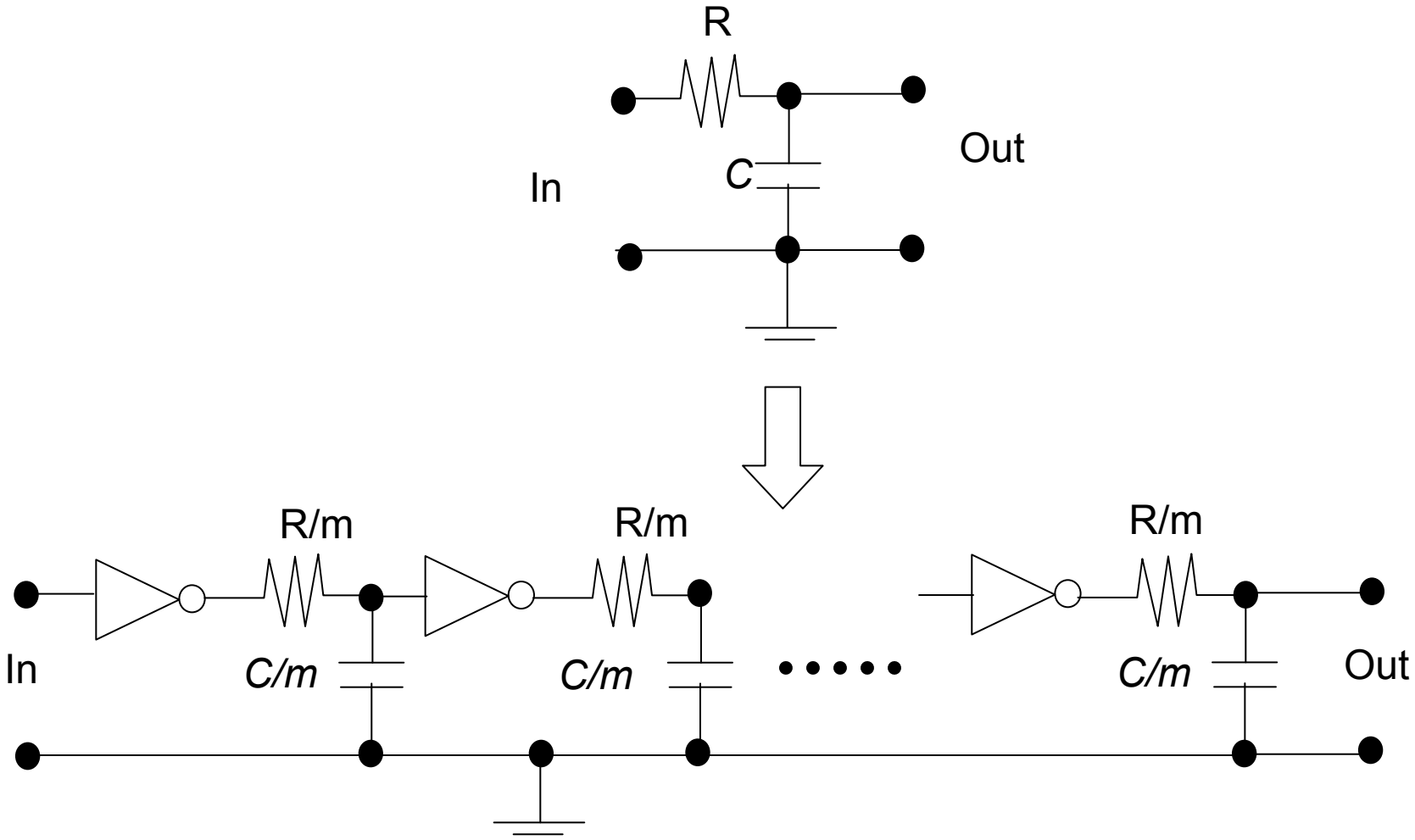
$$t_p = R_s C_w + 0.5 R_{wire} C_w = R_s C_w + 0.5 r_w c_w l^2$$

$$RC\ should\ be\ considered\ when:$$

$$0.5 r_w c_w l^2 > R_s C_w \Rightarrow l > 2R_s / r_w$$

Ex: R_s = 1KΩ, r = 0.075Ω/μm => l > 2.67cm

Reducing RC-delay: Repeater Insertion



Buffer (Repeater) Insertion

$$t_{stage} = t_{buf} + t_{RC} = t_{buf} + 0.5(0.69)\frac{R}{m}\frac{C}{m} = t_{buf} + (0.38)rc\frac{L^2}{m^2}$$

$$t = m\left(0.38rc\frac{L^2}{m^2}\right) + (m)t_{buf}$$

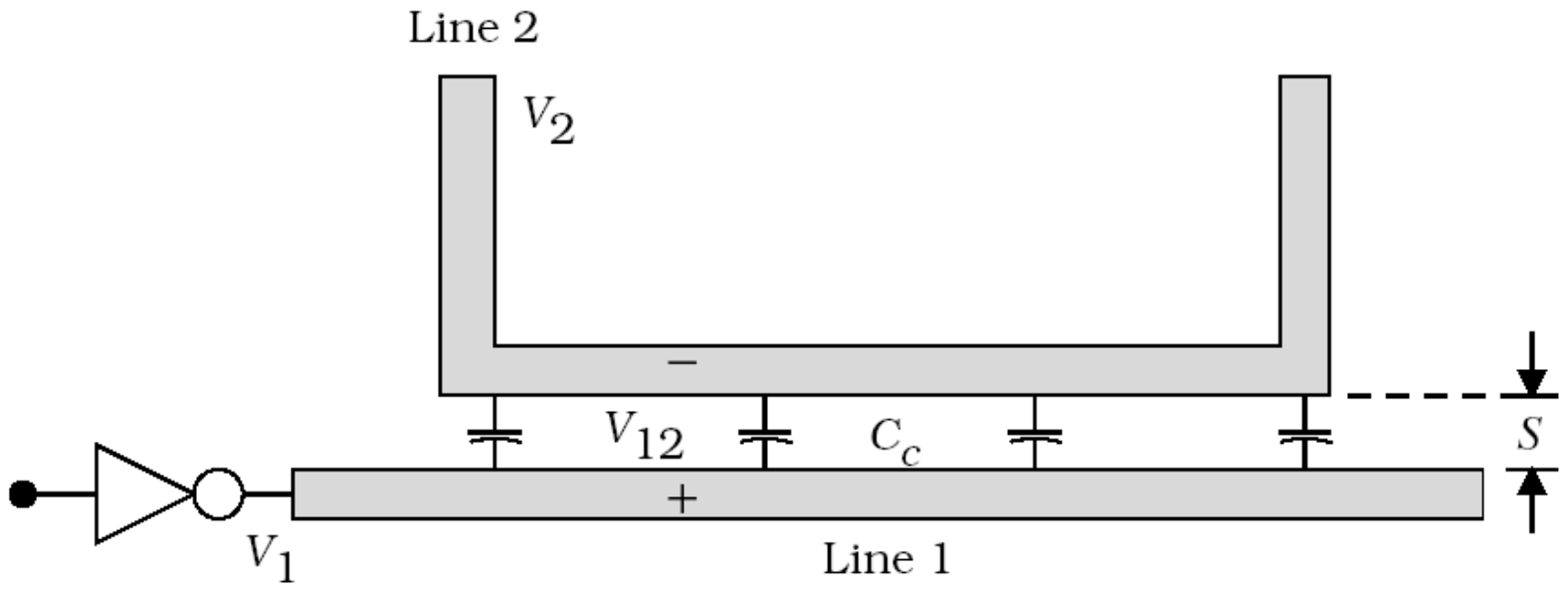
$optimal\ m = m_{opt}$:

$$\frac{\partial t}{\partial m} = -0.38rc\frac{L^2}{m^2} + t_{buf} = 0 \Rightarrow m_{opt} = L\sqrt{\frac{0.38rc}{t_{buf}}}$$

$optimal\ delay$:

$$t_{opt} = L\sqrt{\frac{0.38rc}{t_{buf}}}\left[t_{buf}\right] + L\sqrt{\frac{0.38rc}{t_{buf}}}t_{buf} = 2L\sqrt{0.38rct_{buf}} = 2\sqrt{t_{wire}t_{buf}}$$

Capacitance coupling between two lines.

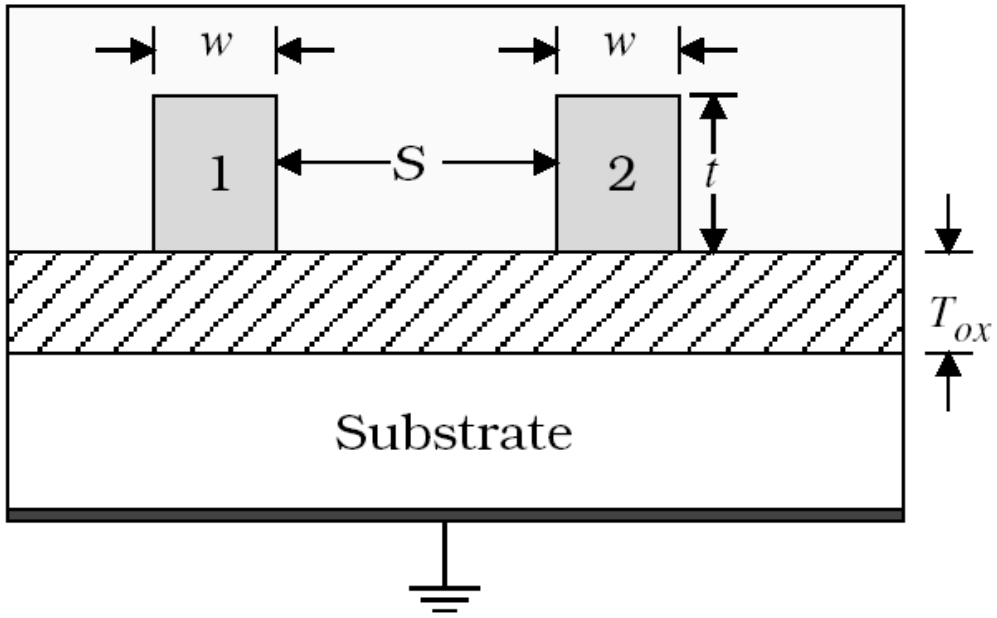


Voltage transition at line ‘1’ will get coupled to line ‘2’

Coupling capacitance

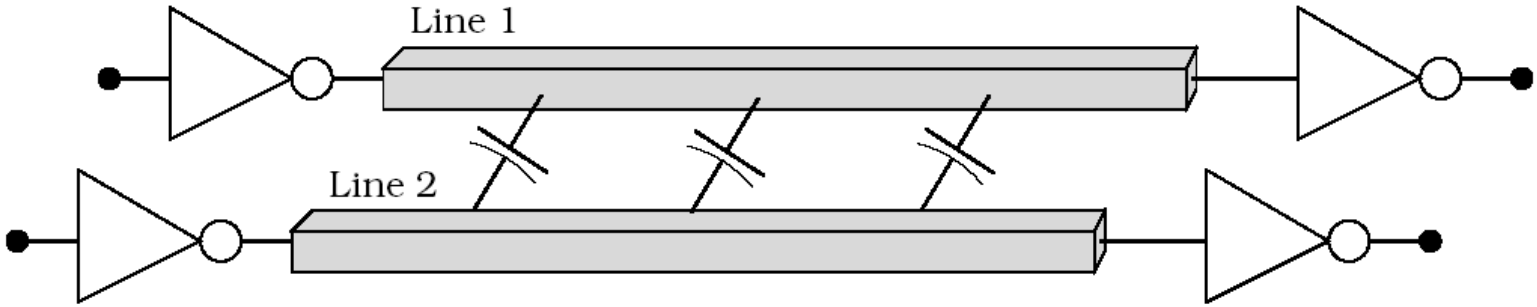
Cross-talk => An important noise source in VLSI circuits

Geometry used for coupling capacitance calculation.

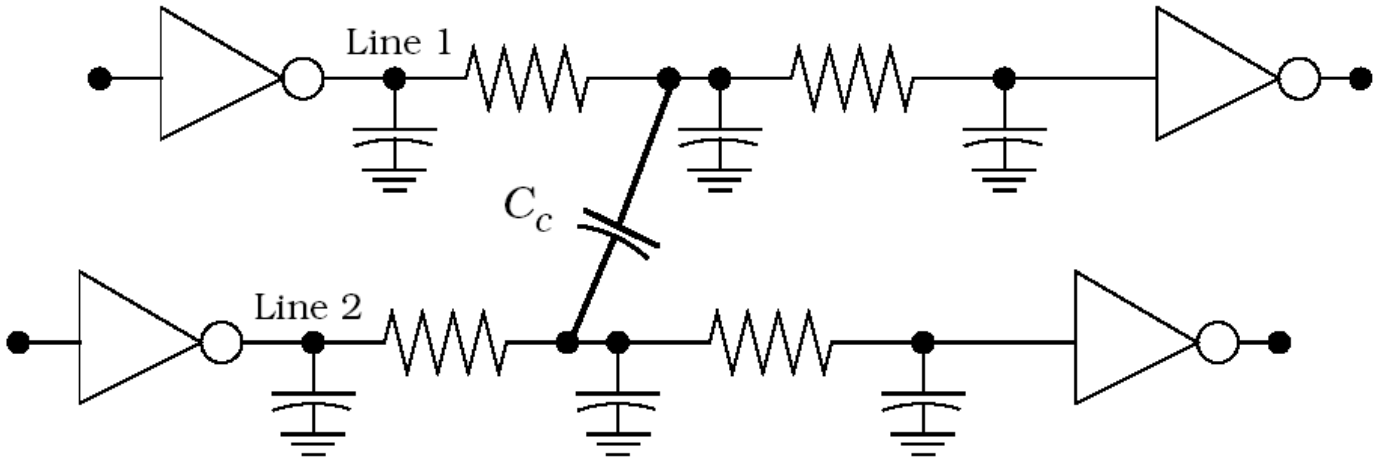


$$c_C = \varepsilon_{ox}\left[0.03\left(\frac{w}{T_{ox}}\right) + 0.83\left(\frac{t}{T_{ox}}\right) - 0.07\left(\frac{t}{T_{ox}}\right)^{0.222}\right]\left(\frac{T_{ox}}{S}\right)^{4/3}$$

Lumped-element coupling circuit model.

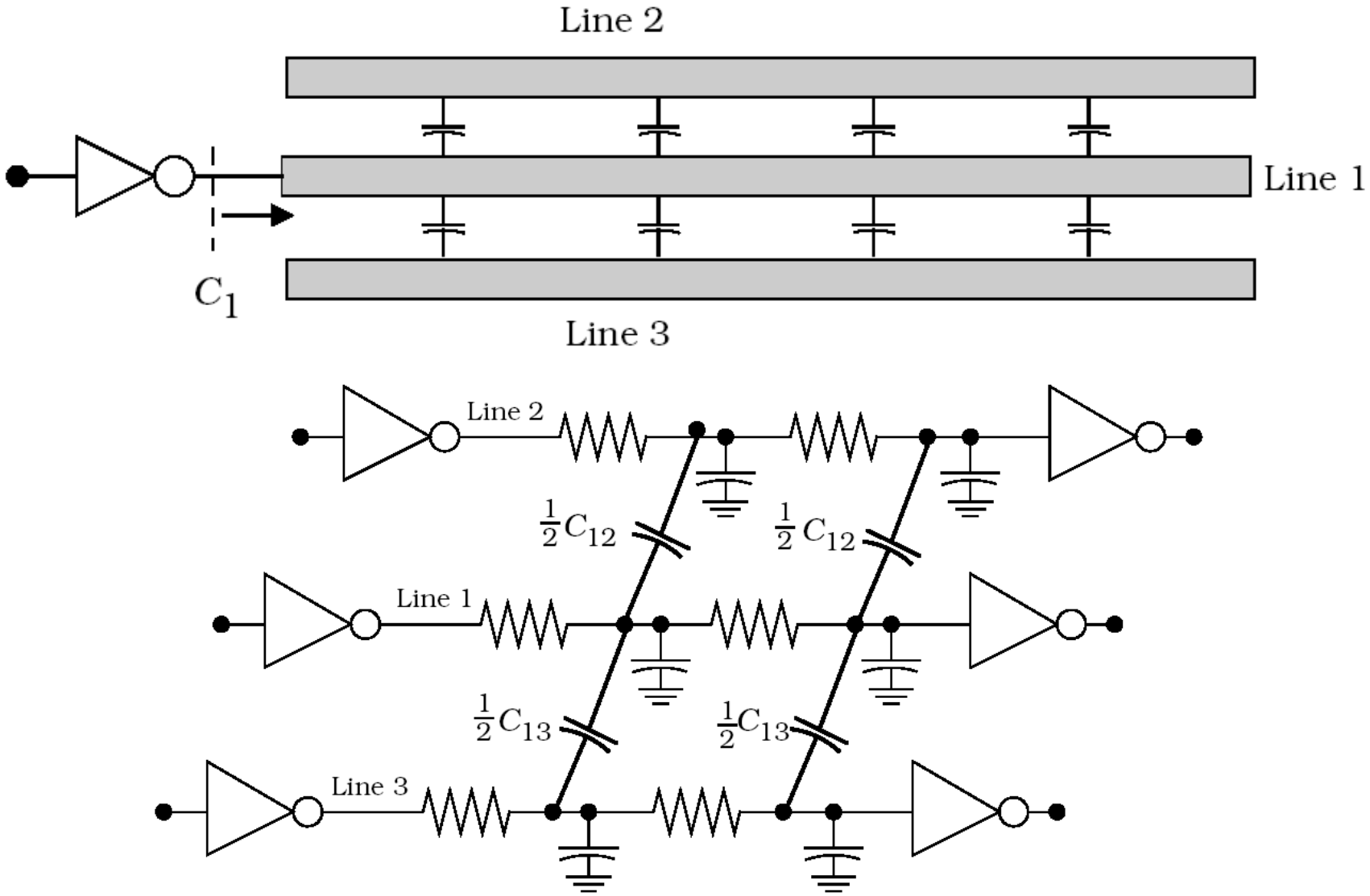


(a) Physical structure



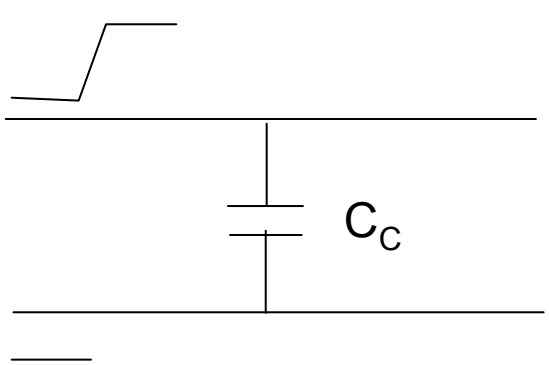
(b) RC model

Multiple-line coupling.

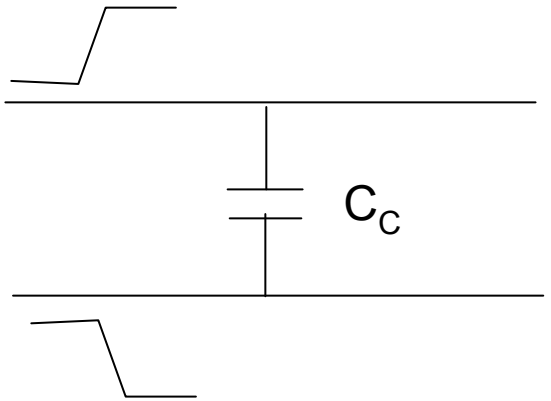


Effects of Coupling Capacitances

- **Capacitive loading on signal line increases**
  - **Higher delay**
- **Cross-Talk: It is a noise source**
  - **Aggressor : The line which is switching**
  - **Victim : The line which is experiencing a voltage change due to the switching in the aggressor**

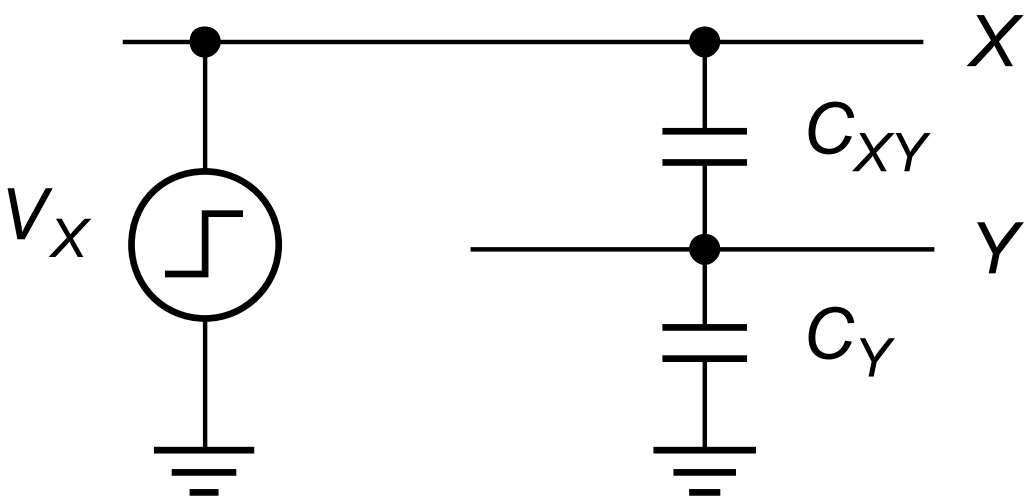


Effective capacitance: C_c



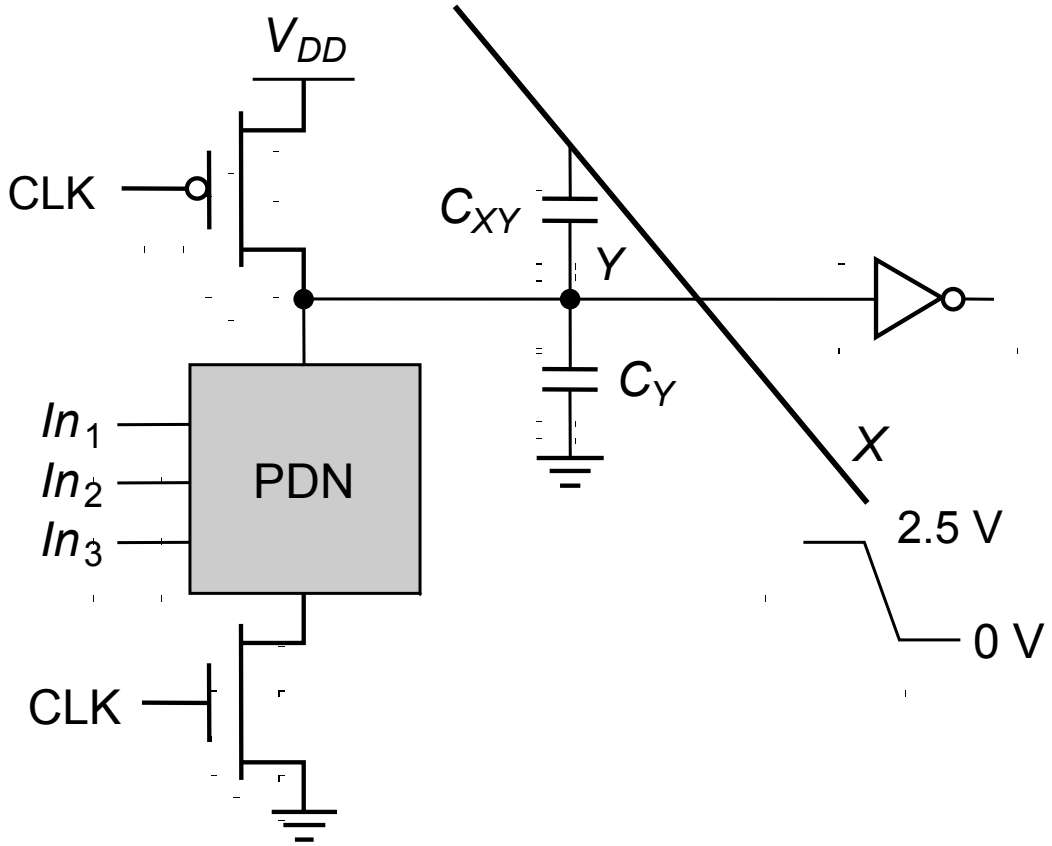
Effective capacitance: $2C_c$

Capacitive Cross Talk



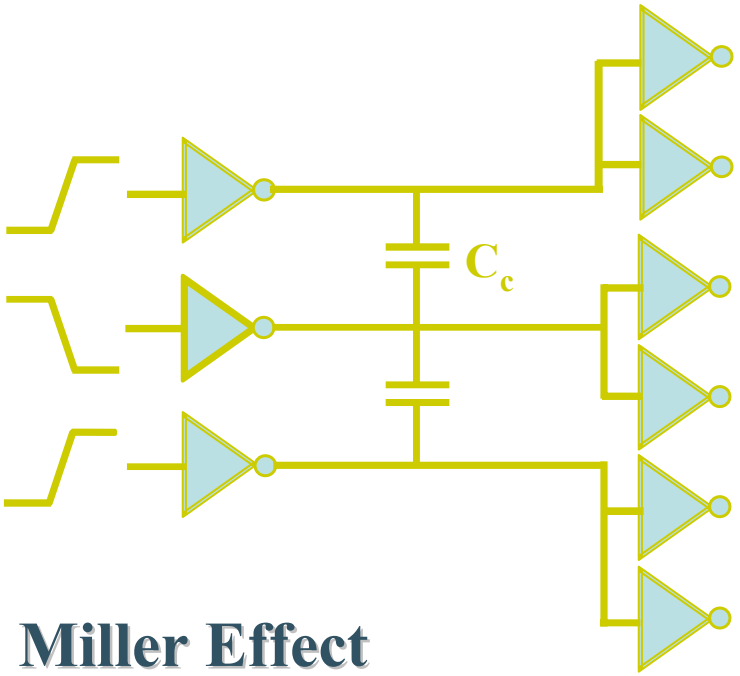
$$\Delta V_Y = \frac{C_{XY}}{C_Y + C_{XY}} \Delta V_X$$

Capacitive Cross Talk Dynamic Node



C_Y = 6fF and C_{XY} = 0.5fF : 0.19 V disturbance

Cross Talk and Performance



Miller Effect

- When neighboring lines switch in opposite direction of victim line, delay increases

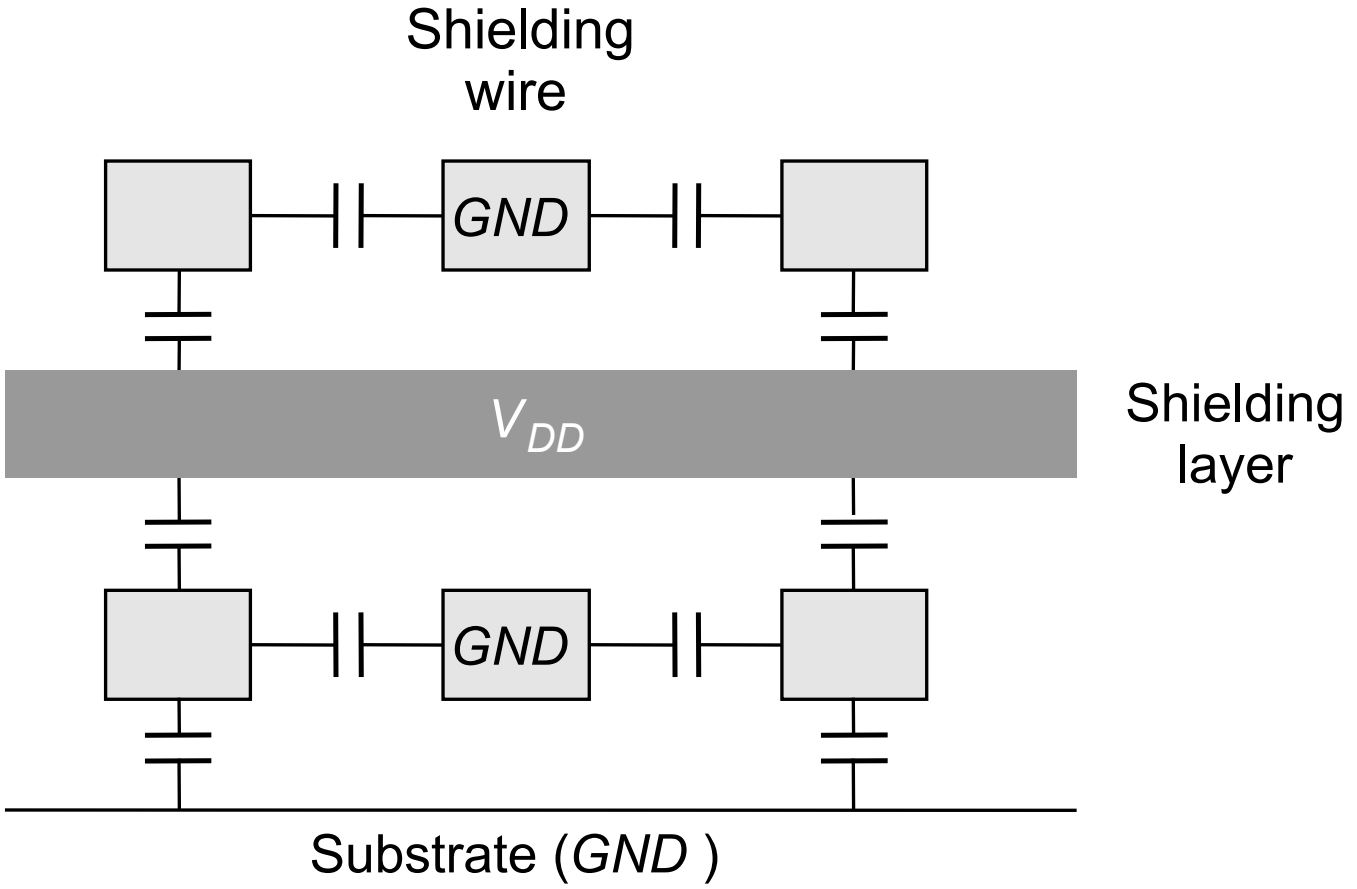
DELAY DEPENDENT UPON ACTIVITY IN NEIGHBORING WIRES

- Both terminals of capacitor are switched in opposite directions $(0 \rightarrow V_{dd}, V_{dd} \rightarrow 0)$
- Effective capacitance is doubled

Dealing with Capacitive Cross Talk

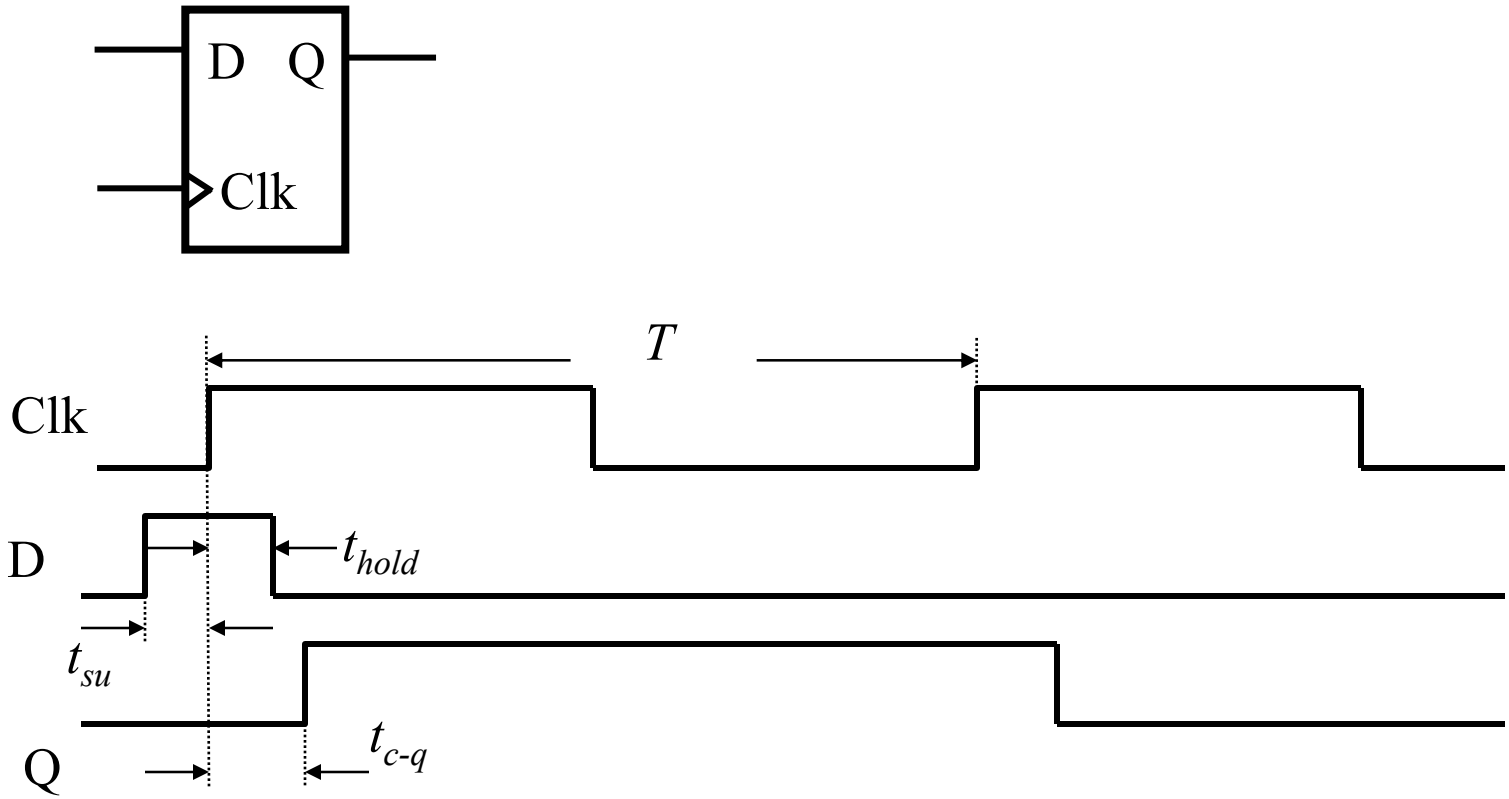
- Avoid floating nodes
- Protect sensitive nodes
- Do not run wires together for a long distance
- Use shielding wires
- Use shielding layers

Shielding



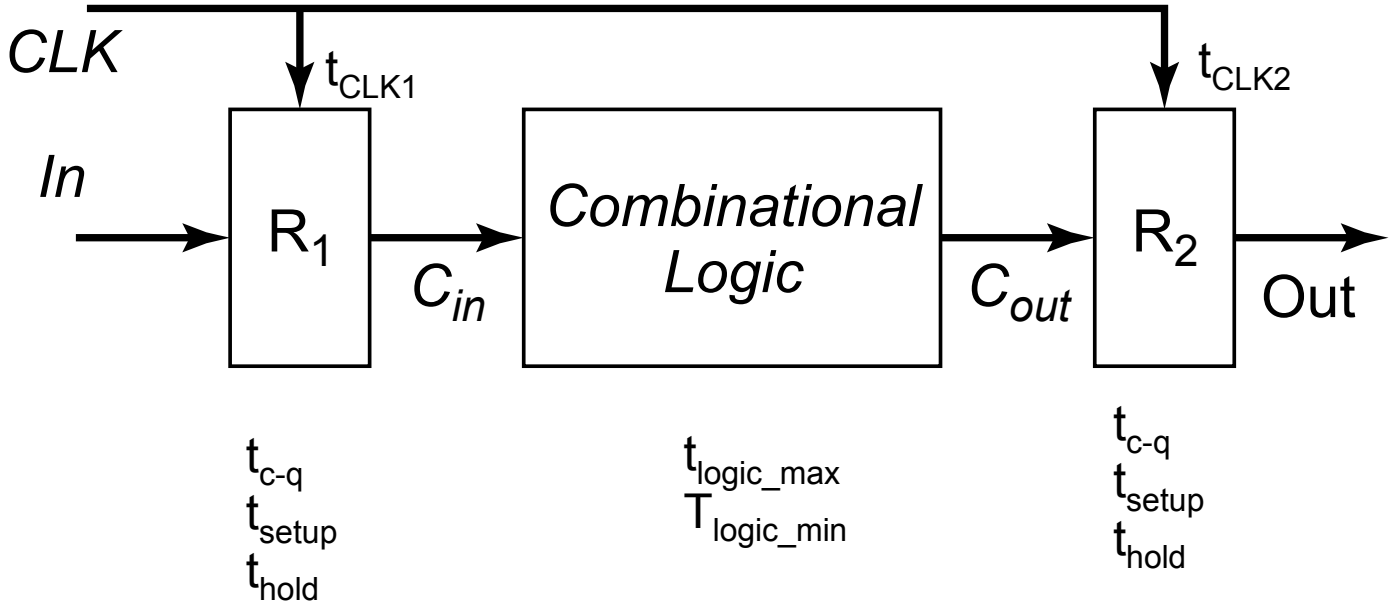
Clocking

Register Parameters



Delays can be different for rising and falling data transitions

Synchronous Timing

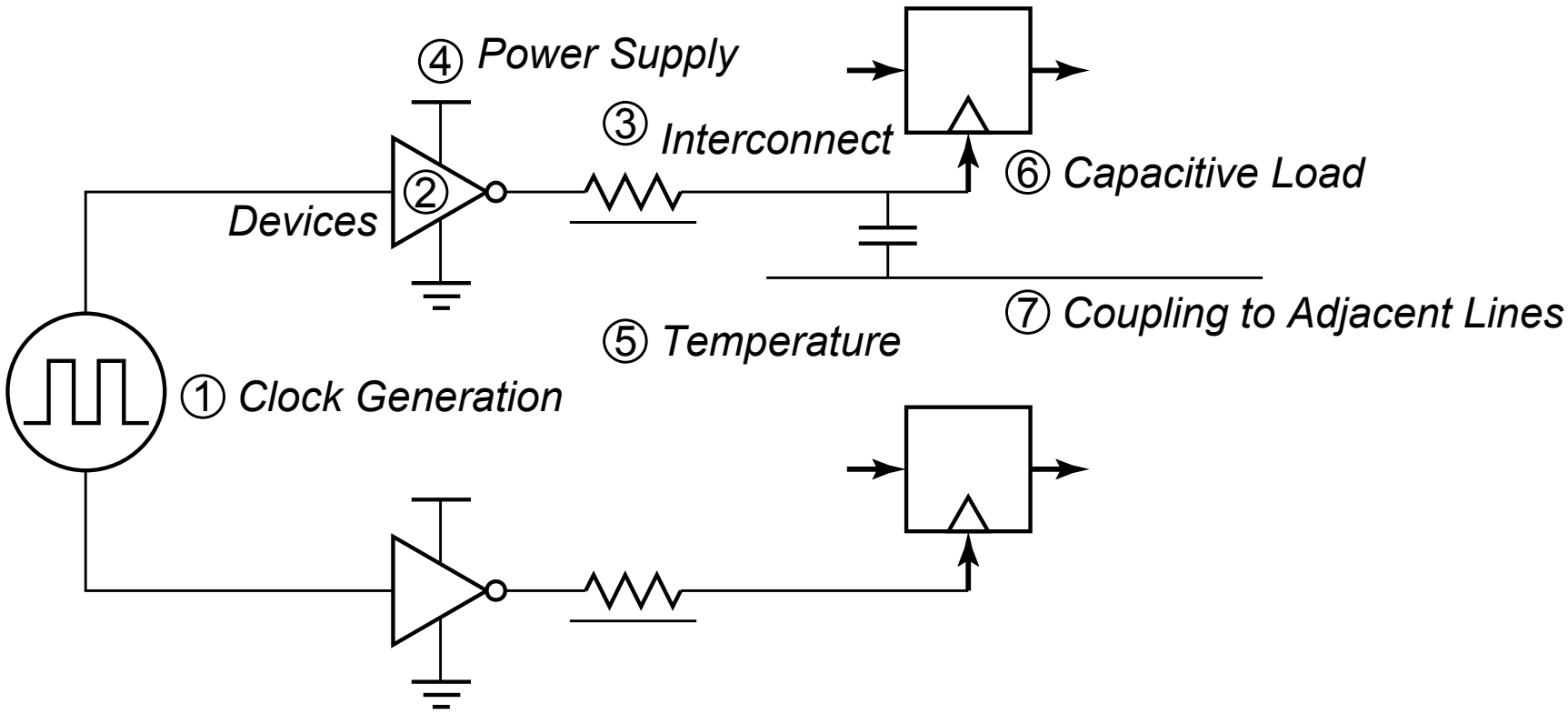


Ideally $t_{CLK1} = t_{CLK2}$

Minimum CLK Period: $T > t_{c\text{-}q} + t_{logic\_max} + t_{setup}$

To avoid race condition: $t_{hold} < t_{c\text{-}q} + t_{logic\_min}$

Clock Uncertainties

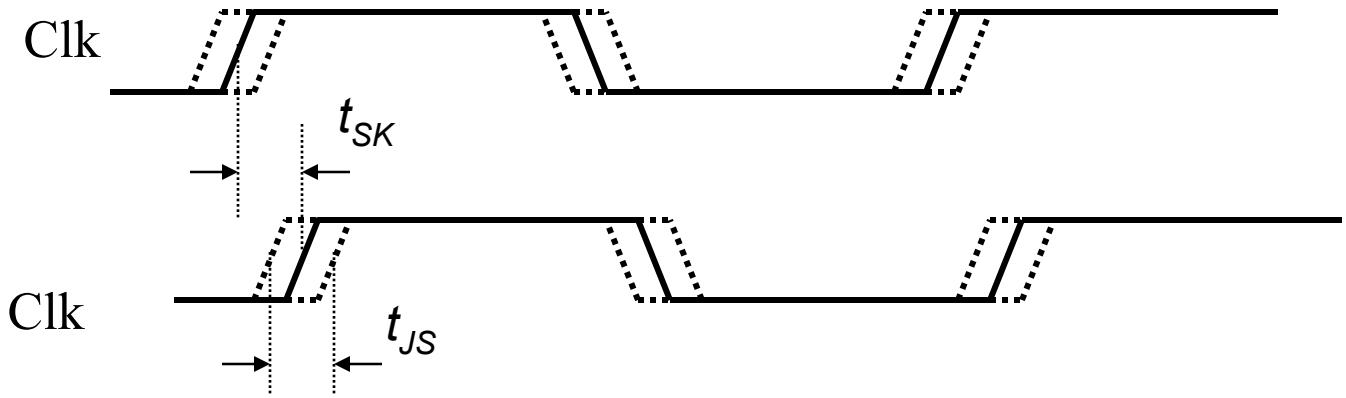


Sources of clock uncertainty

Clock Nonidealities

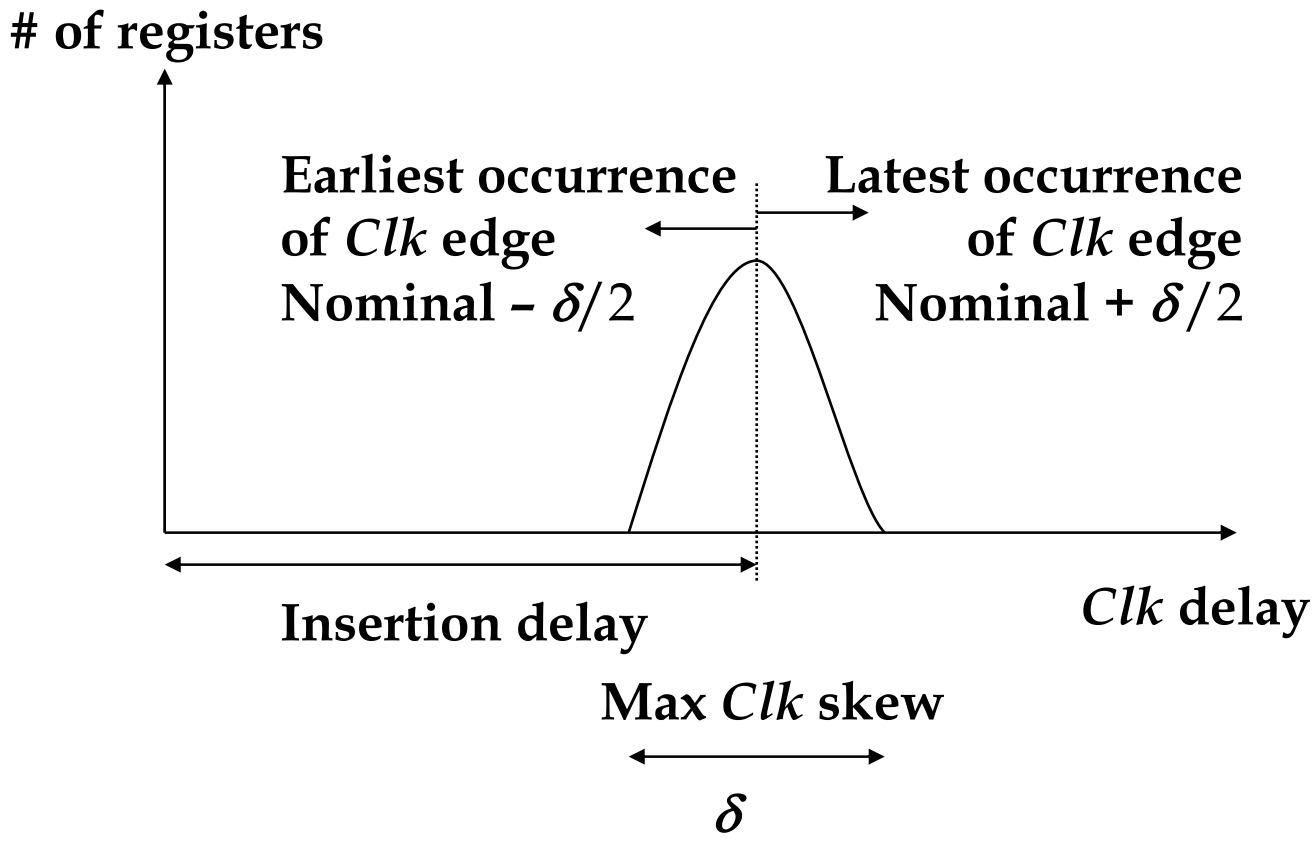
- **Clock skew**
  - Spatial variation in temporally equivalent clock edges; deterministic + random, t_{SK}
- **Clock jitter**
  - Temporal variations in consecutive edges of the clock signal; modulation + random noise
  - Cycle-to-cycle (short-term) t_{JS}
  - Long term t_{JL}

Clock Skew and Jitter

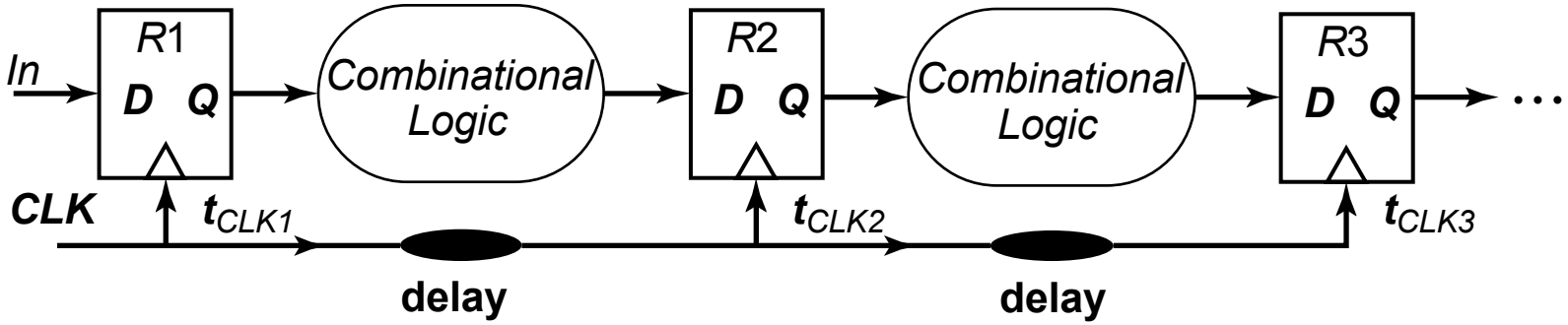


- Both skew and jitter affect the effective cycle time

Clock Skew

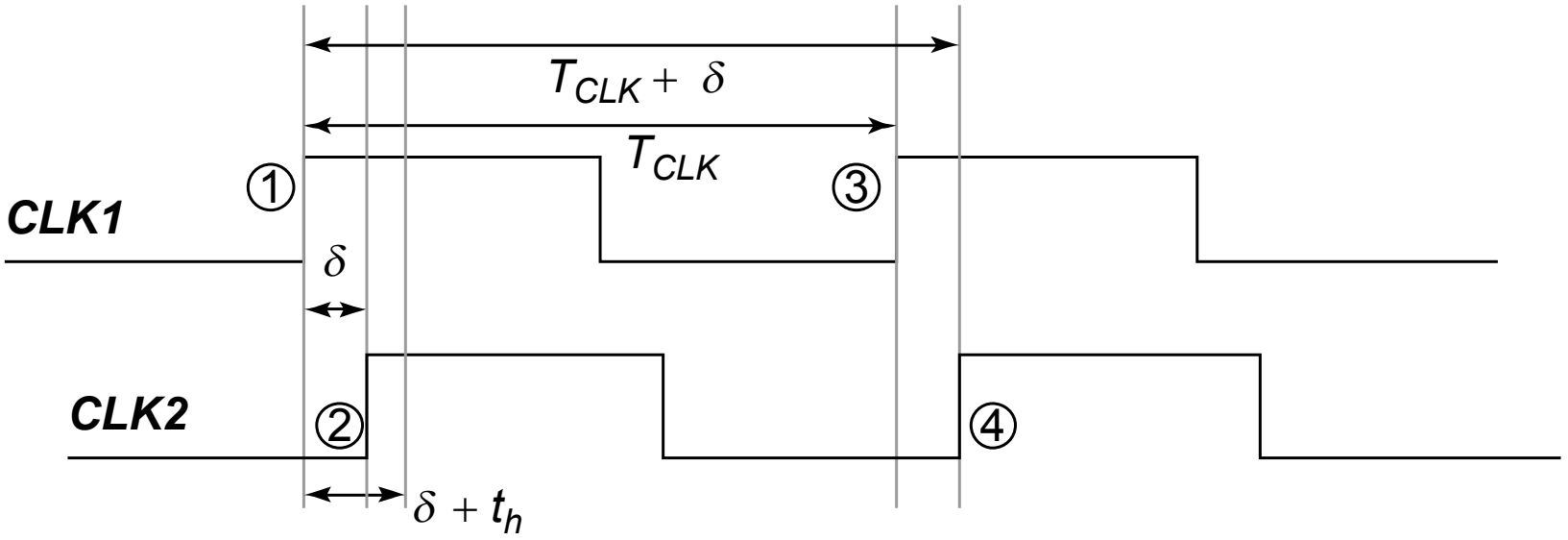

# of registers
Earliest occurrence of *Clk* edge Nominal – $\delta/2$
Latest occurrence of *Clk* edge Nominal + $\delta/2$
Insertion delay
Clk delay
Max *Clk* skew
δ

Positive Skew



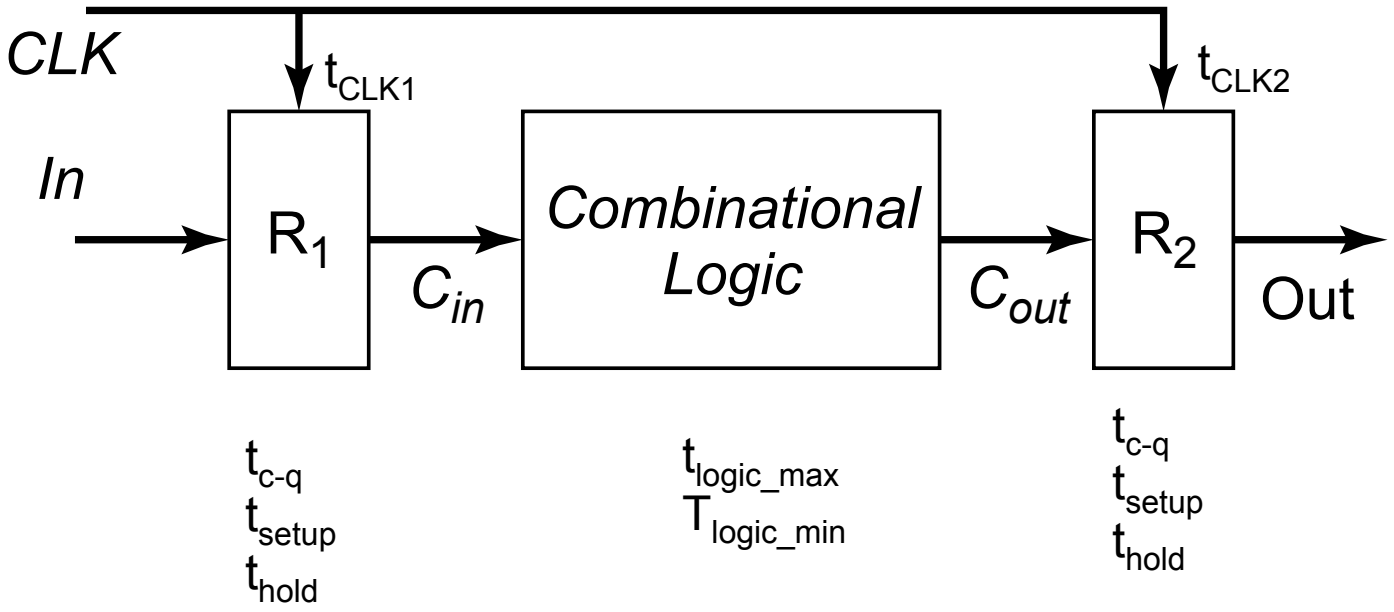
Receiving clock edge arrives late

Positive Skew



Launching edge arrives before the receiving edge

Timing Constraints: Positive Skew



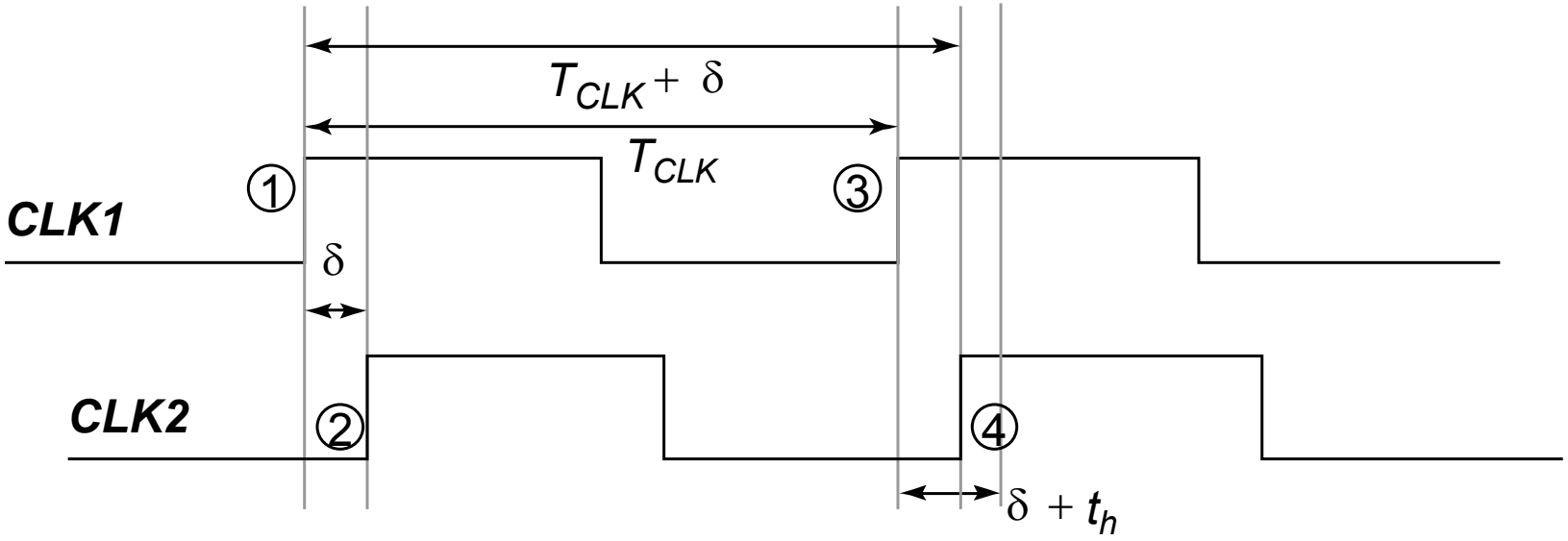
Minimum cycle time:

$$T + \delta > t_{c\text{-}q} + t_{setup} + t_{logic\_max} \Rightarrow T > t_{c\text{-}q} + t_{setup} + t_{logic\_max} - \delta$$

Apparently you can operate faster if there is a positive skew

Positive Skew: Race Condition

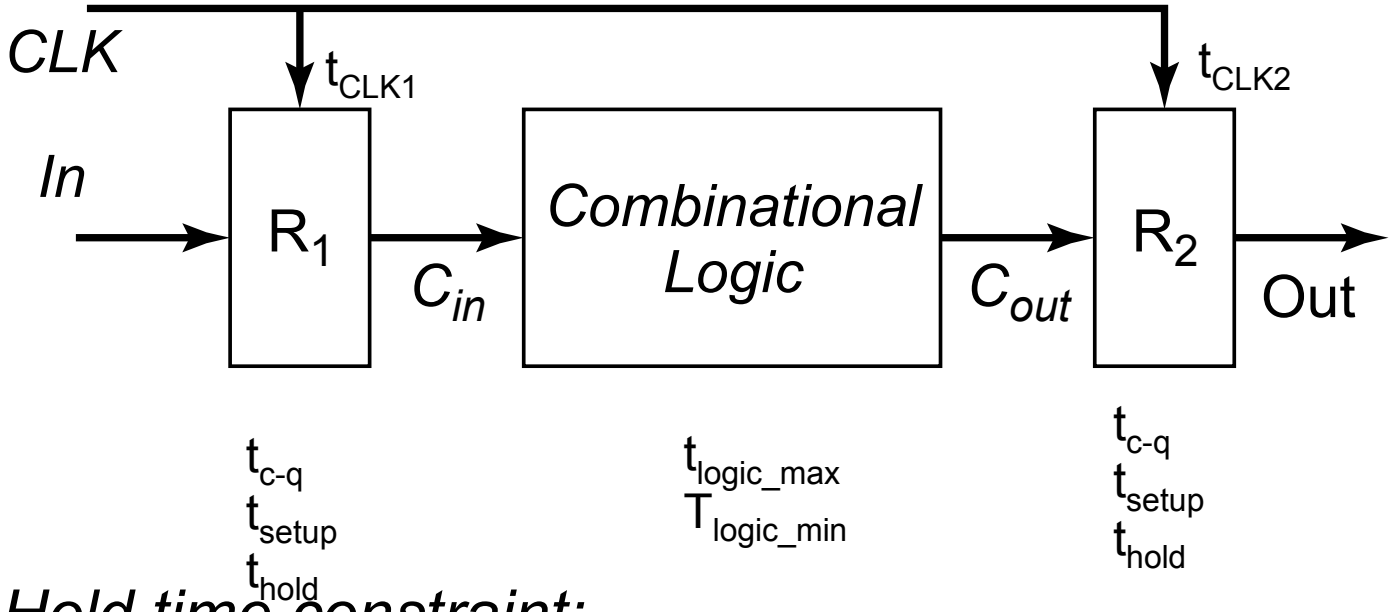
Launching edge arrives before the receiving edge



If the data at the input of R2 changes within the hold time after the arrival of edge (4), the sampled data may be wrong -> race condition

$$t_{hold} + \delta < t_{(c\text{-}q)} + t_{(logic\_min)}$$

Positive Skew: Race Condition

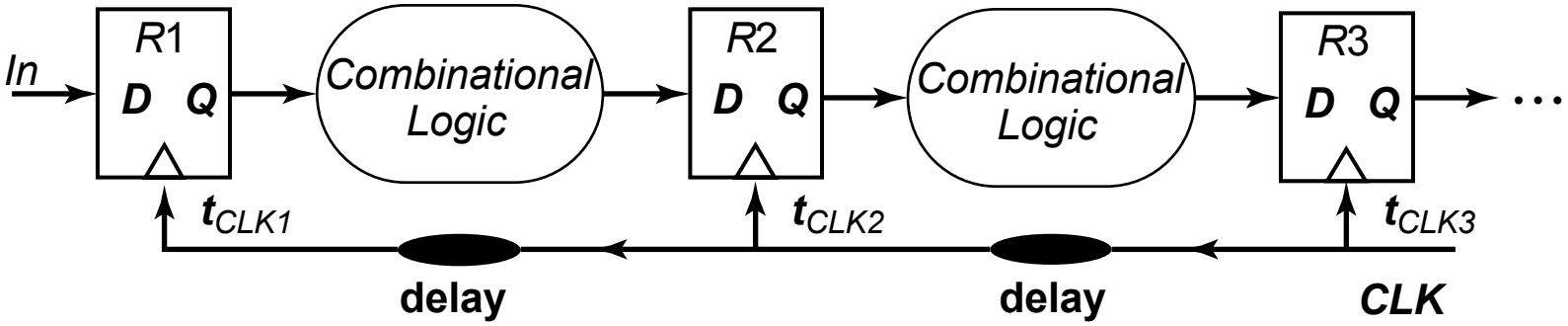


Hold time constraint:

$$t_{(c\text{-}q,\ cd)} + t_{(logic,\ cd)} > t_{hold} + \delta$$

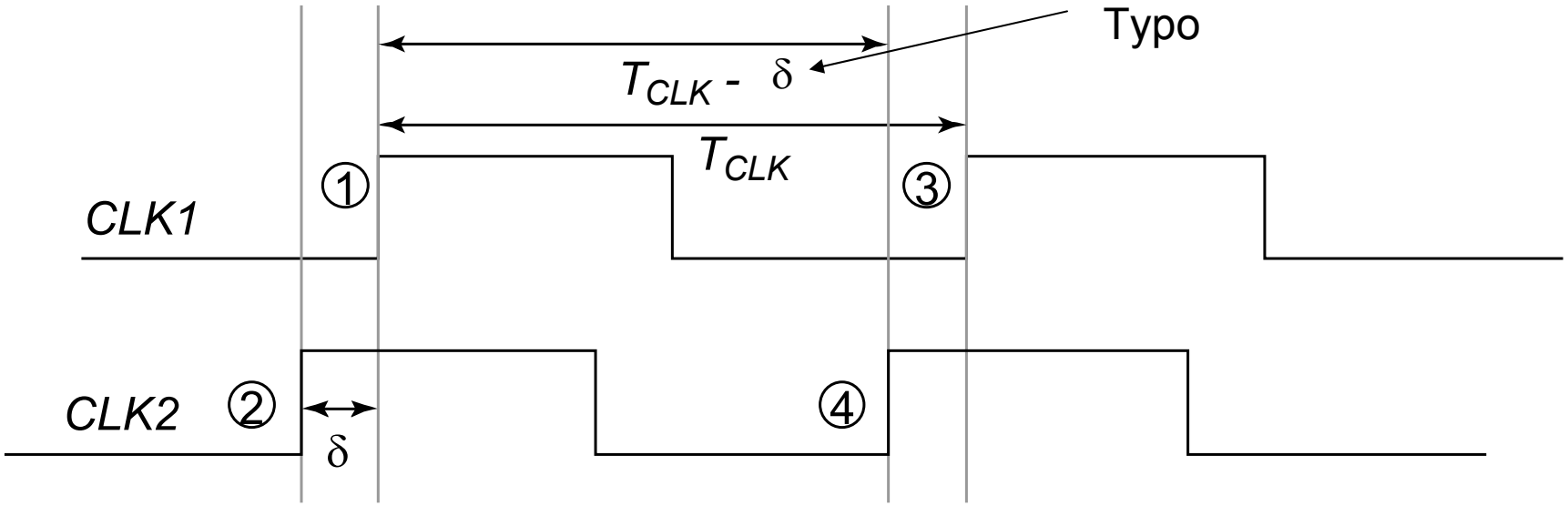
Worst case is when receiving edge arrives late
Race between data and clock

Negative Skew



Receiving edge arrives before the launching edge

Negative Skew



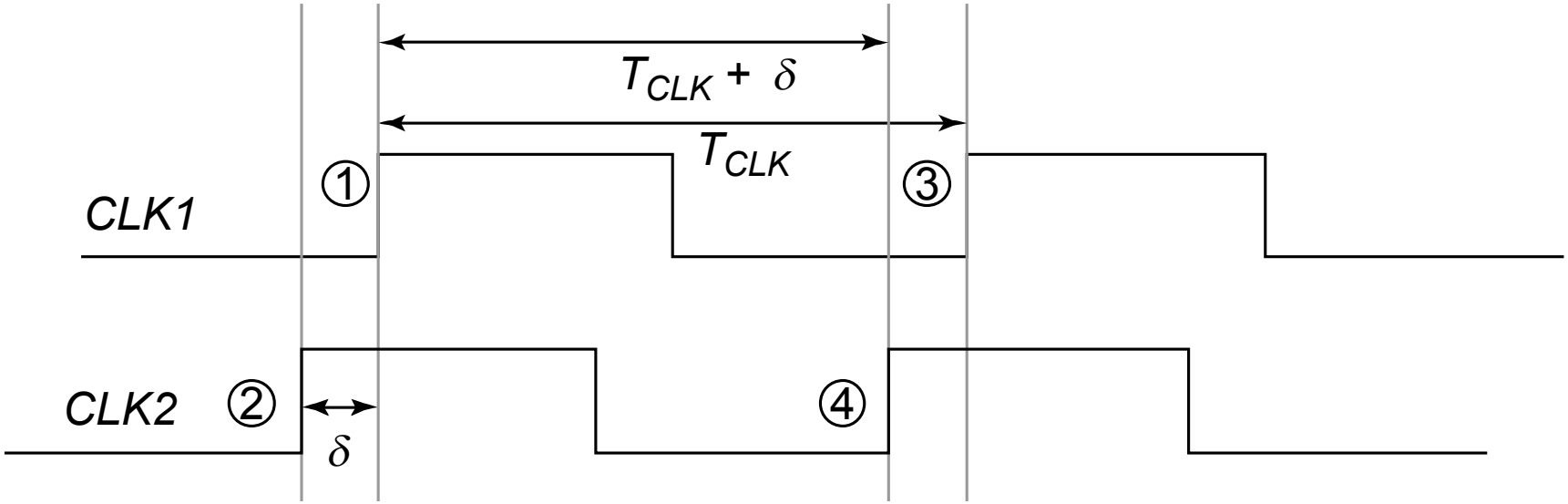
Receiving edge arrives before the launching edge
The time available for computation reduces

Minimum cycle time:

$T - \delta > t_{c\text{-}q} + t_{setup} + t_{logic\_max} \Rightarrow T > t_{c\text{-}q} + t_{setup} + t_{logic\_max} + \delta$

Slower operation

Negative Skew: No race

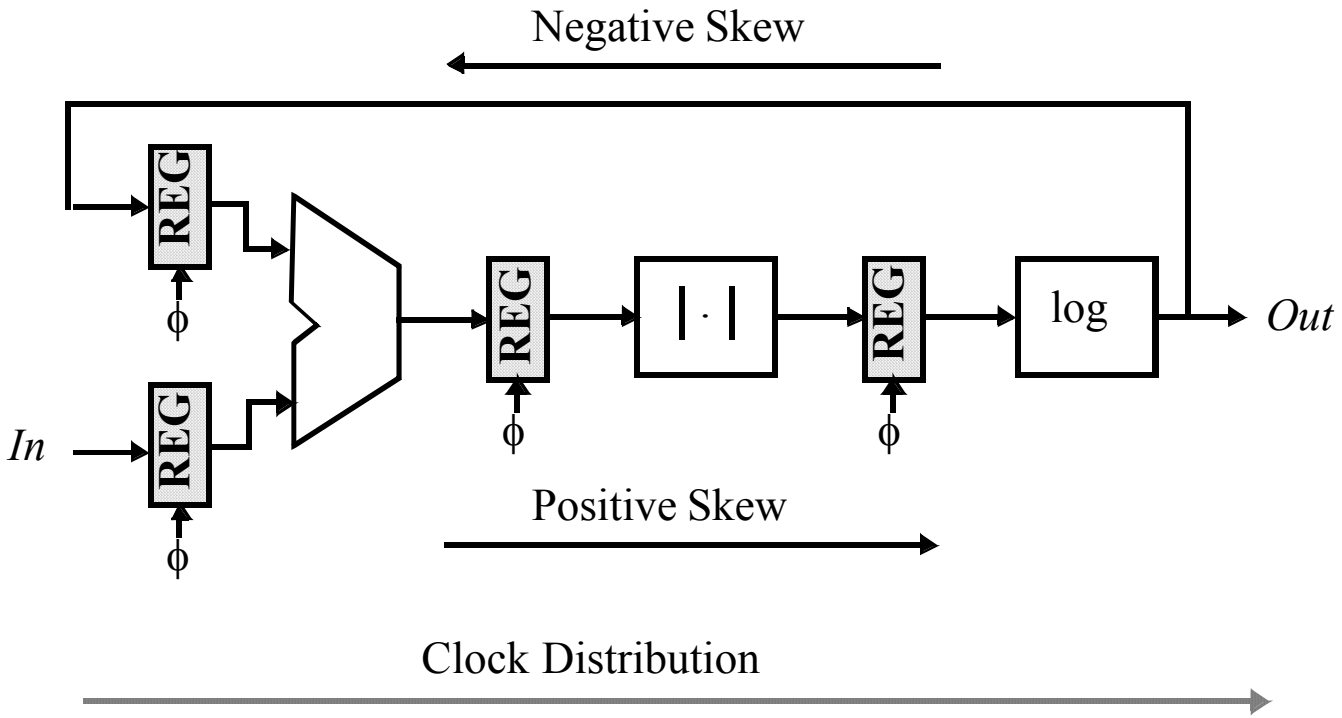


The data at R_2 is sample even before the data is sample at R1
No race condition is possible

Clock Skew

- If clock propagates in the same direction as data => positive skew
  - Can improve performance
  - Degrades reliability of the circuit
- If clock propagates in the opposite direction to data
  - Functionality is guaranteed
  - Lower operating frequency

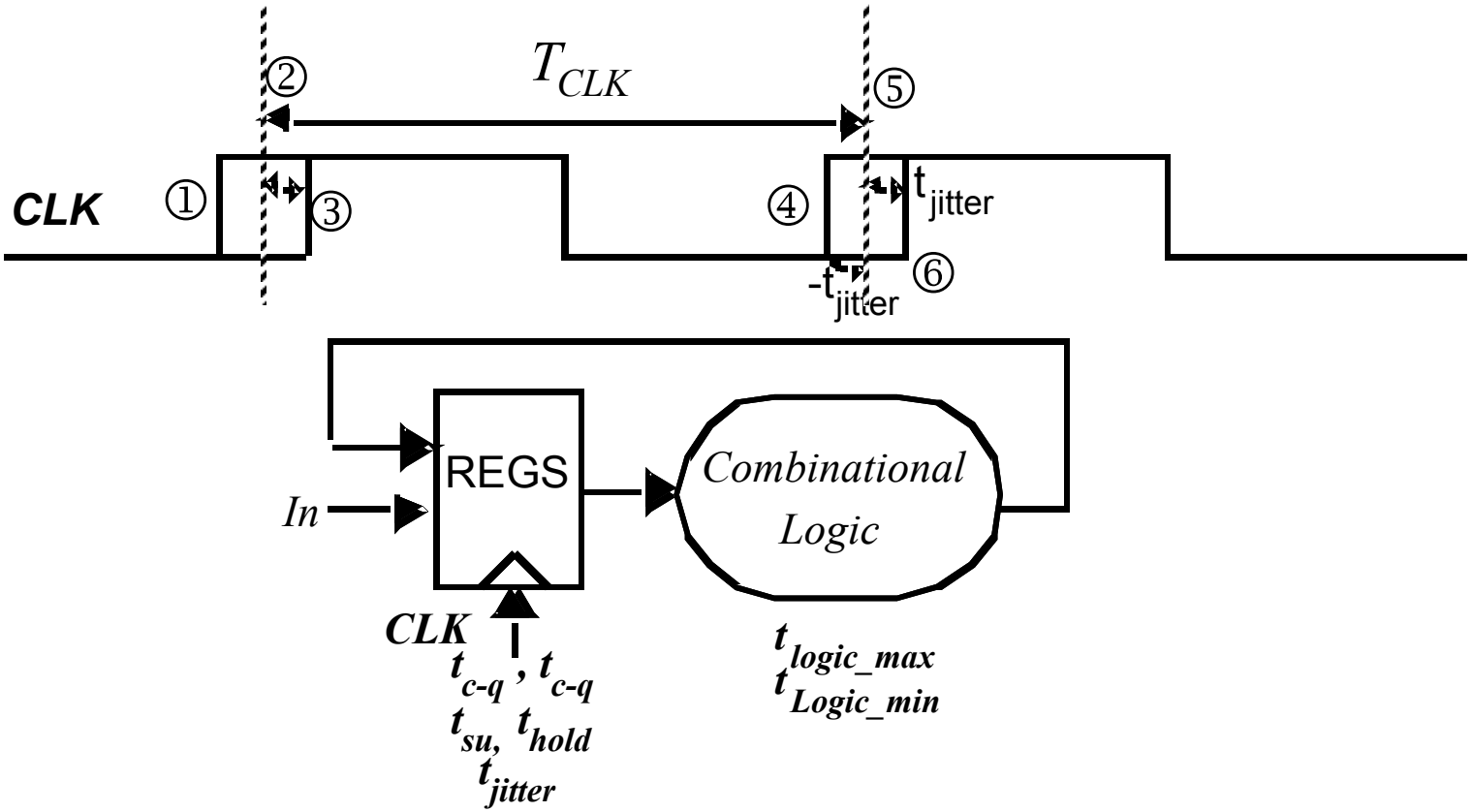
Feedback System



Data and Clock Routing

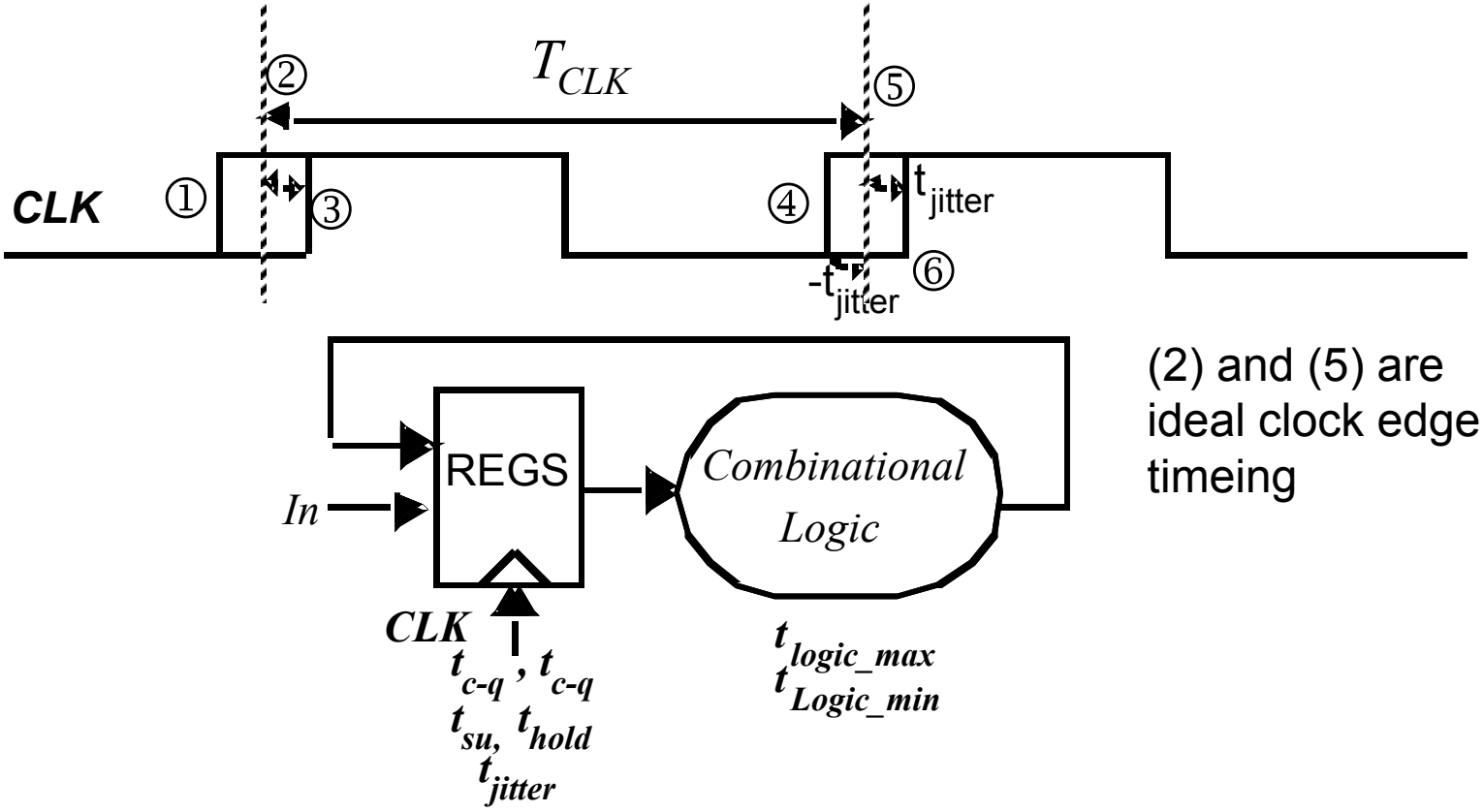
Both positive and negative skew

Impact of Jitter



- **Jitter is the temporal variation of the clock edge at a given spatial point.**
- **Abosulte jitter (t_{jitter})**
  - **Worst case shift of the clock edge at a given spatial point**
- **Cycle-to-cycle jitter (T_{jitter}):**
  - **Time-varying deviation of the single clock period relative to the ideal reference clock**

Impact of Jitter



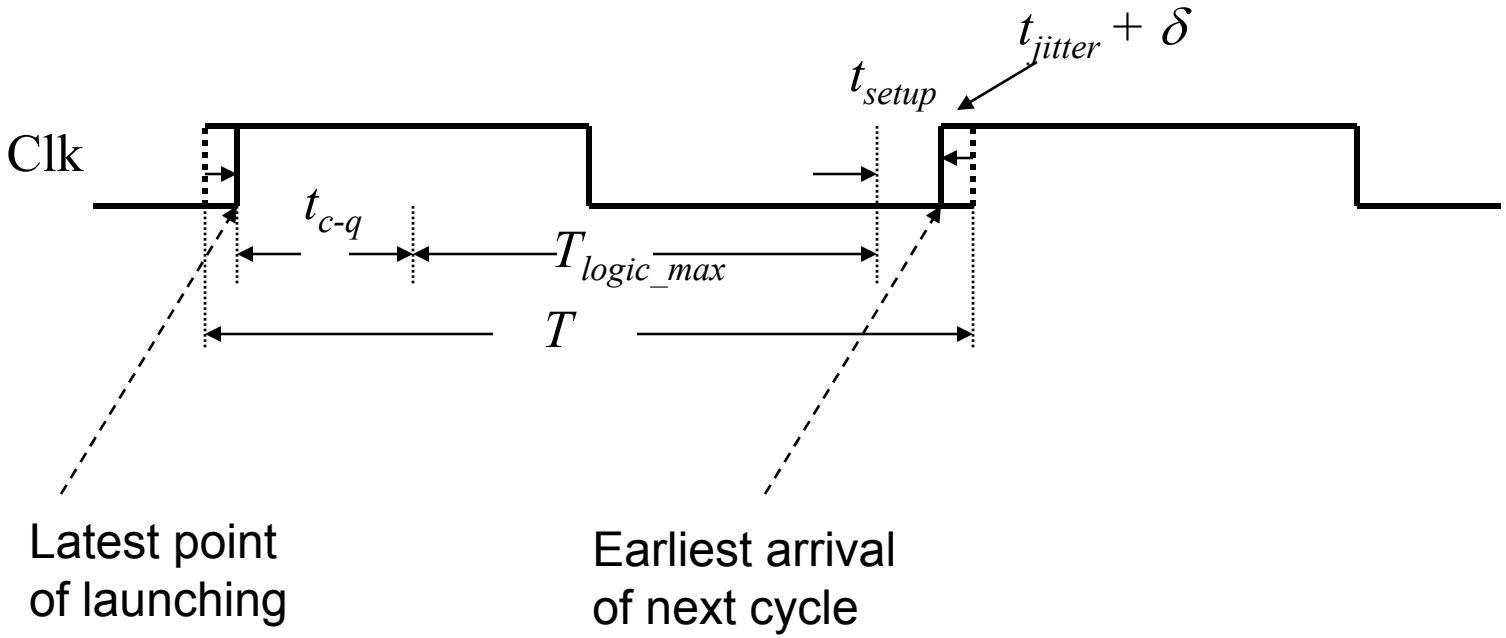
(2) and (5) are ideal clock edge timeing

Worst case: (2) arrives late (at (3)) and (5) arrives early (at (4))

$$T_{CLK} - 2t_{jitter} > t_{c-q} + t_{logic\_max} + t_{setup}$$

$$\Rightarrow T_{CLK} > t_{c-q} + t_{logic\_max} + t_{setup} + 2t_{jitter}$$

Longest Logic Path in Edge-Triggered Systems



Worst case for delay:
Launching edge arrives late : after t_{jitter}
Receiving edge arrives early: $t_{jitter} + \delta$ before the actual time (assume negative skew)

Clock Constraints in Edge-Triggered Systems

If launching edge is late and receiving edge is early, the data will not be too late if:

$$t_{c\text{-}q} + t_{logic\_max} + t_{setup} < T - t_{jitter,1} - t_{jitter,2} - \delta$$

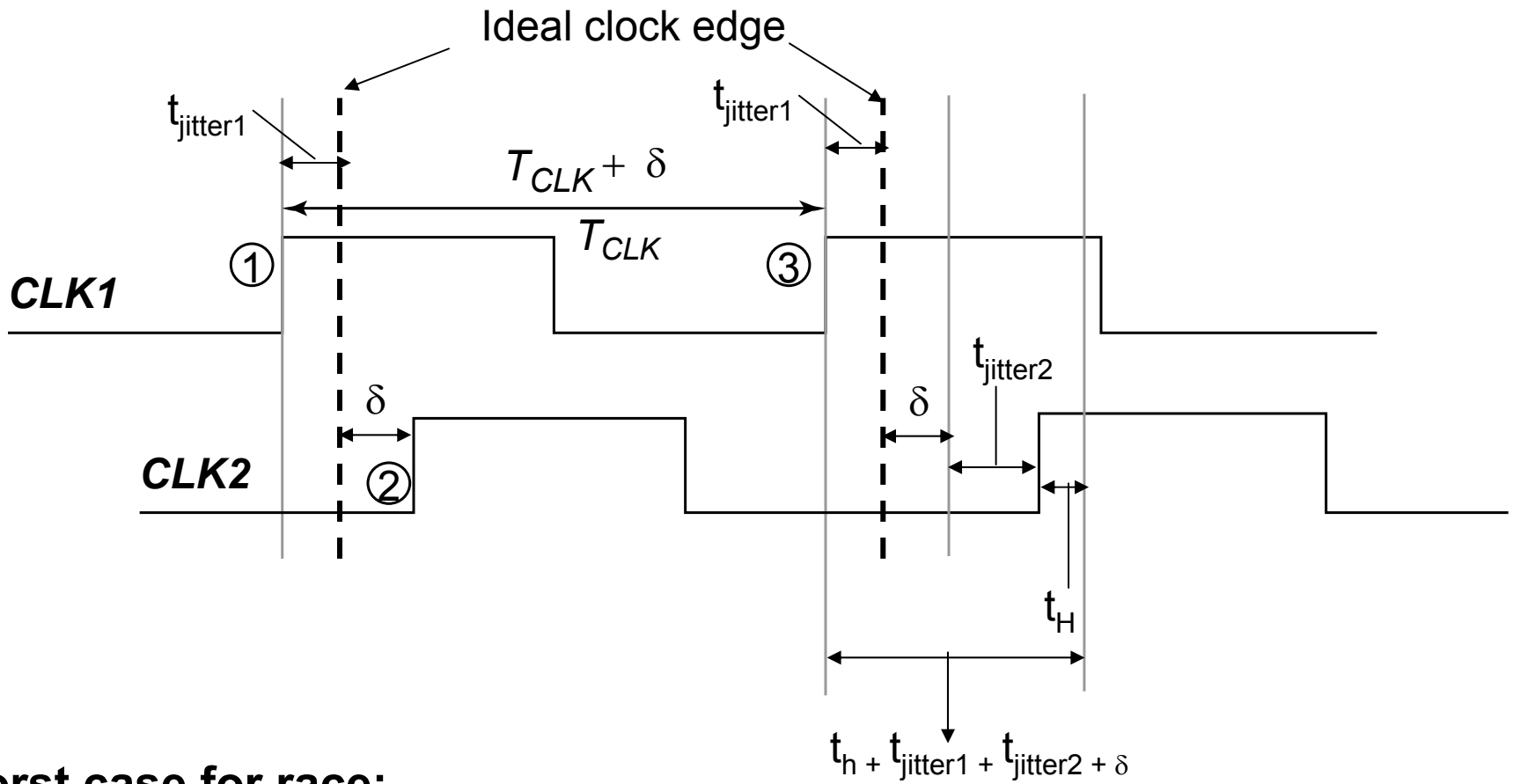
Minimum cycle time is determined by the maximum delays through the logic

$$T > t_{c\text{-}q} + t_{logic\_max} + t_{setup} + 2t_{jitter} + \delta$$

Typo

Negative skew results in worst case timing

Shortest Path Considerations



Worst case for race:

Launching edge arrives early : @t_{jitter} before the actual time

Receiving edge arrives late: @t_{jitter} + δ after the actual time

Clock Constraints in Edge-Triggered Systems

If launching edge is early and receiving edge is late:

$$t_h + t_{jitter,2} + \delta < t_{c\text{-}q} + t_{logic\_min} - t_{jitter,1}$$

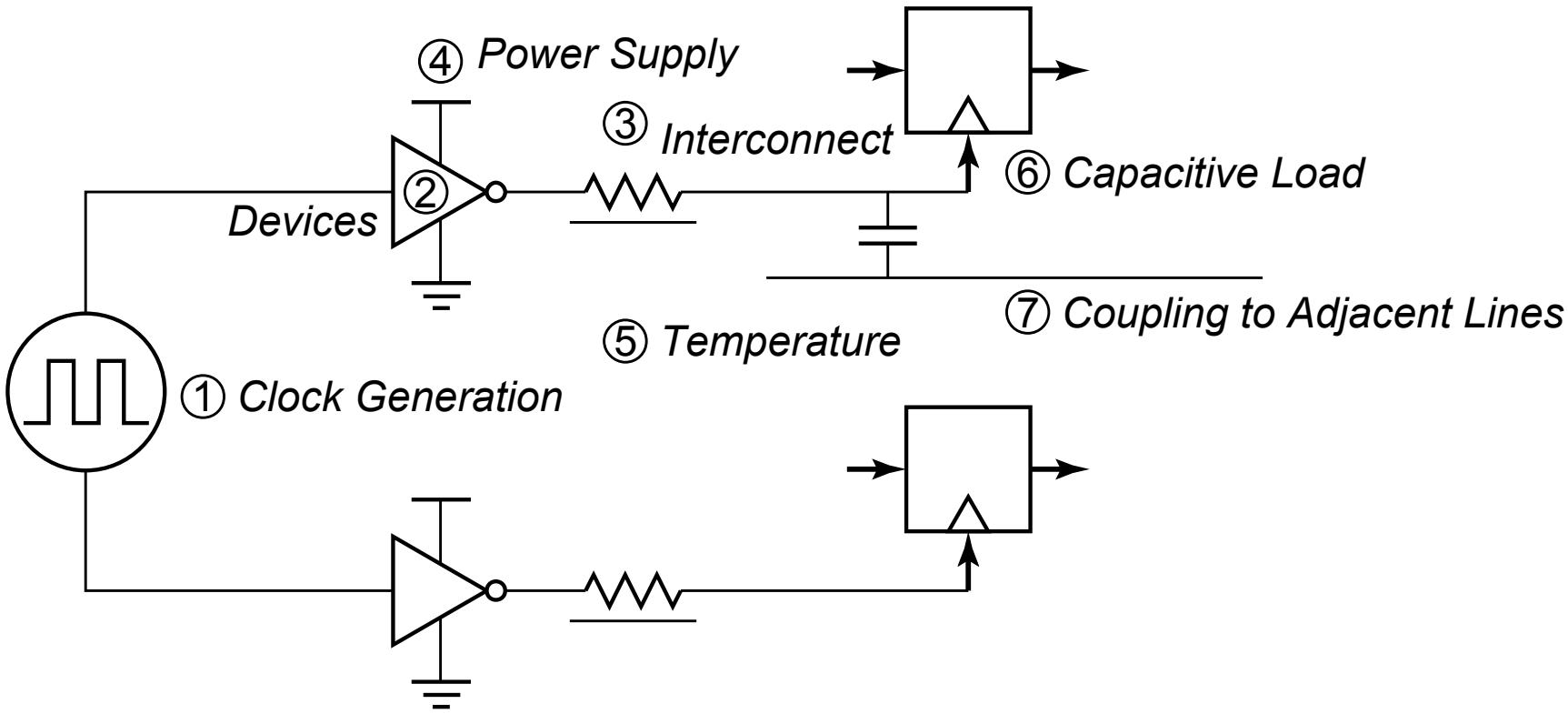
Minimum logic delay

$$t_{c\text{-}q} + t_{logic\_min} < t_h + 2t_{jitter} + \delta$$

Clock Distribution

- Key Point
  - The absolute delay of the clock arrival from the generation point to the different flip-flops is not a concern as long as it arrives at all the flip-flops at the same time

Clock Uncertainties



Sources of clock uncertainty

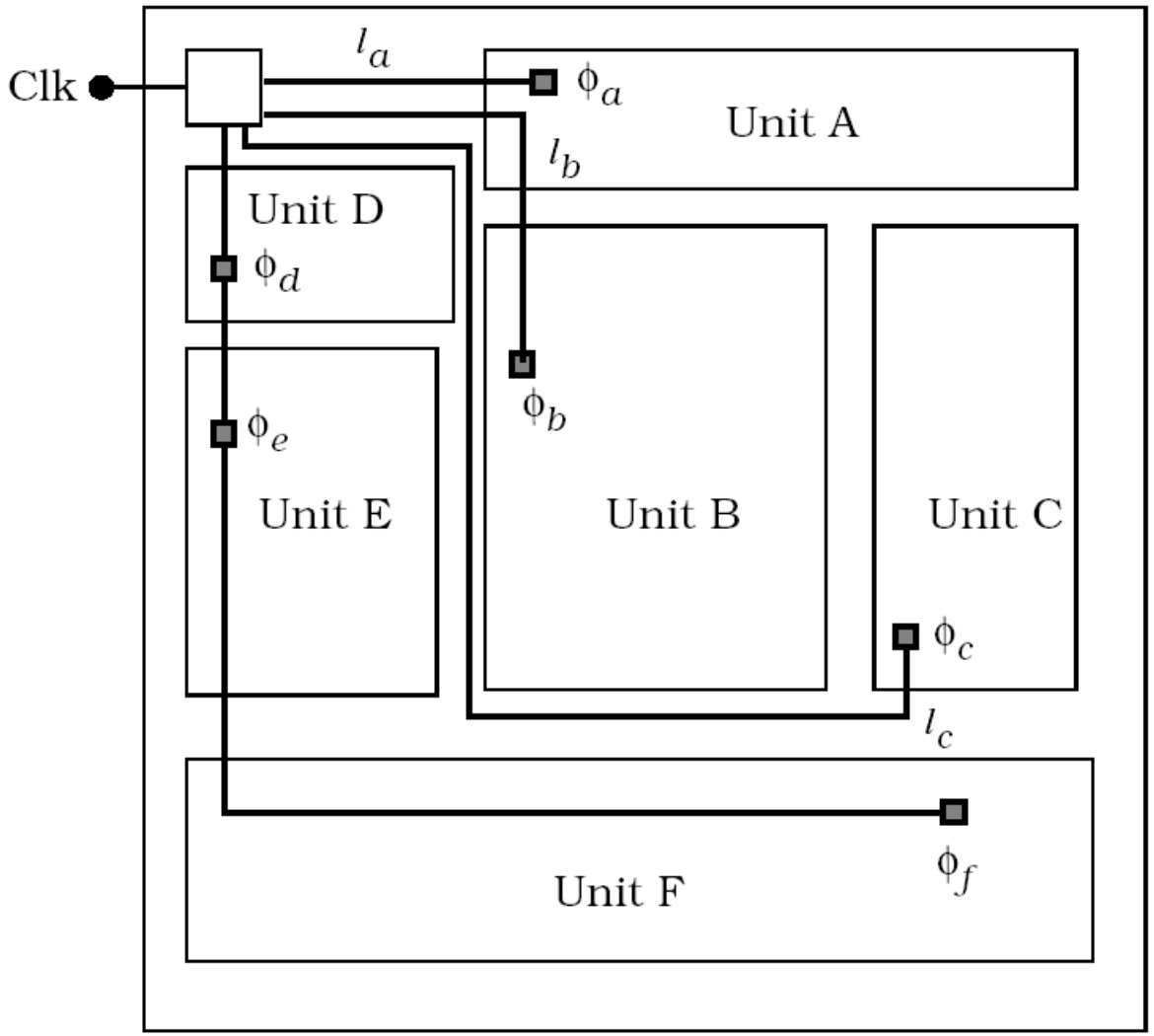
Clock Uncertainties

- Clock generation: Source of Jitter
- Device variations in clock tree: source of skew
- Interconnect variations: source of skew
- Power supply noise - major source of jitter
- Temperature: source of skew (primarily)
- Capacitive load: Skew and jitter
  - Gate capacitance of different flip-flops
  - Capacitive coupling with neigbouring signal wires

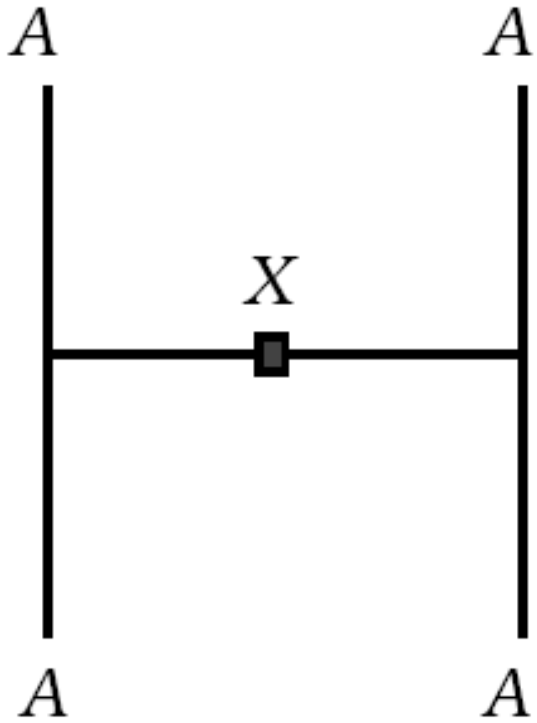
Clock distribution to on-chip modules.

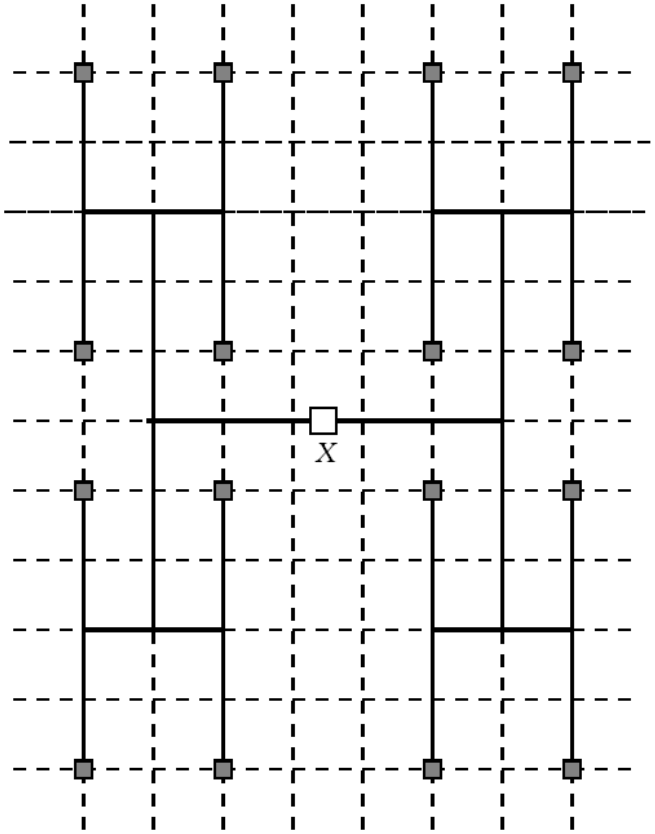
Delay α length^2

Large clock skew can be possible

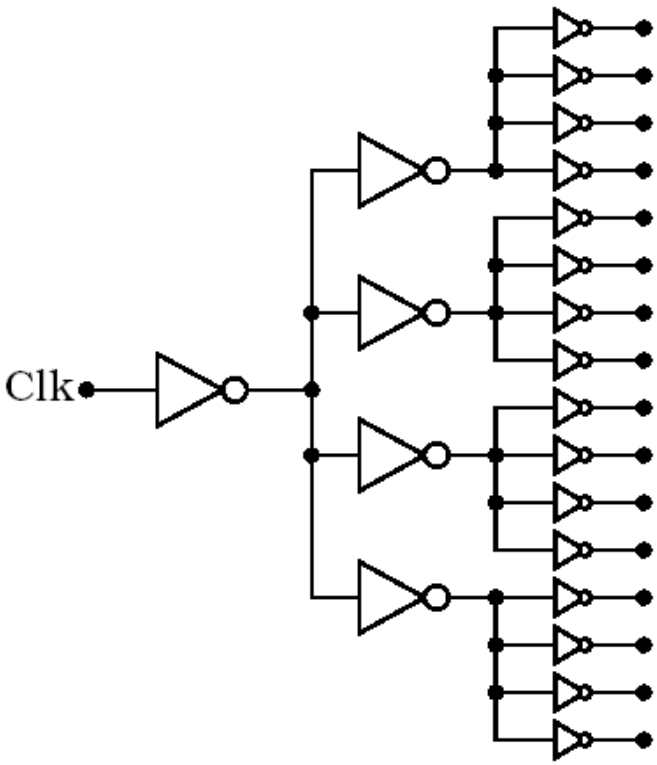


Macro-level H-type distribution tree.

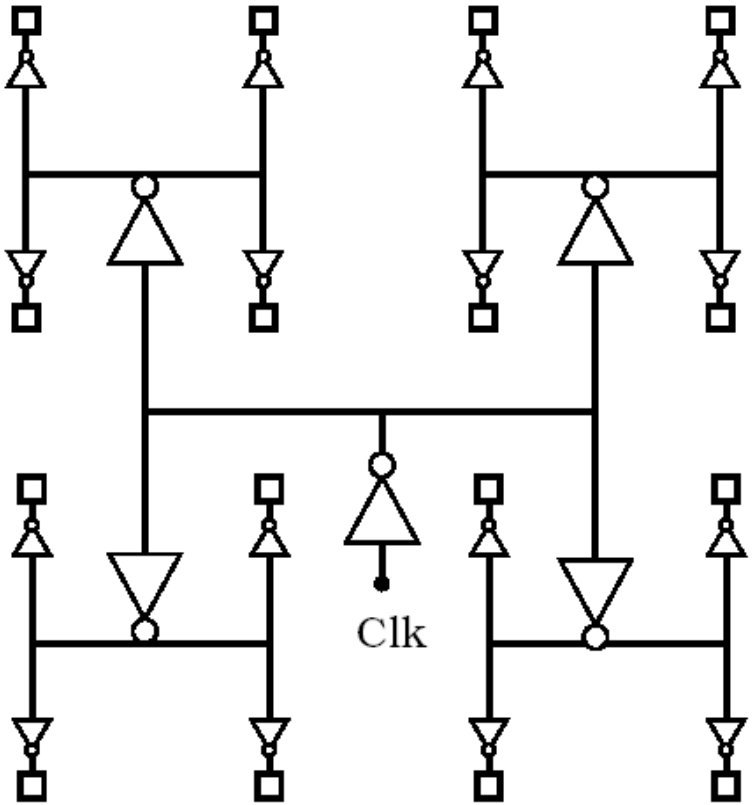




Clock Driver Network

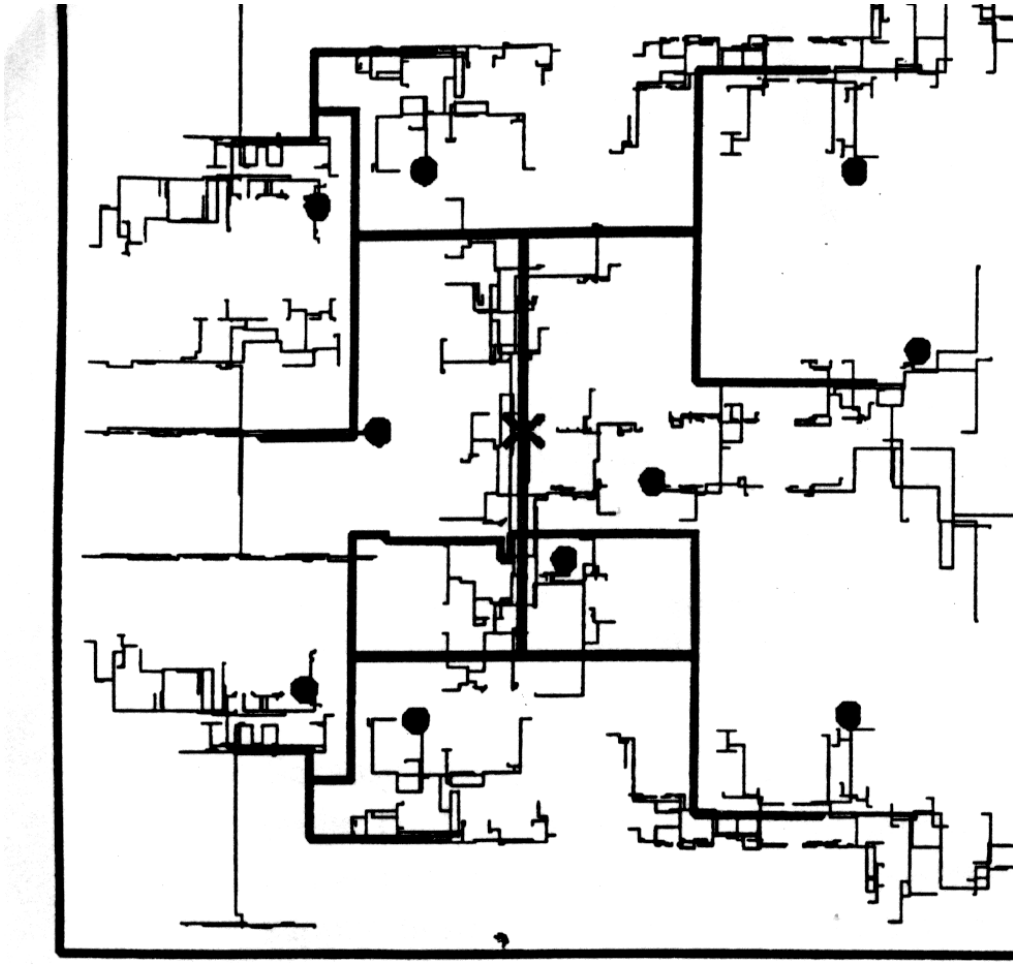


(a) Driver tree



(b) Application to H-tree

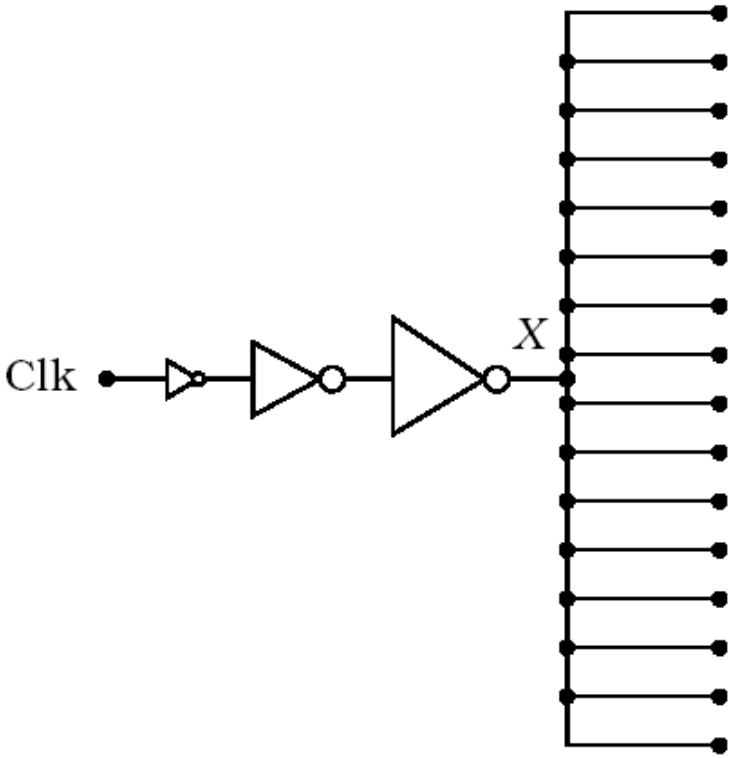
More realistic H-tree



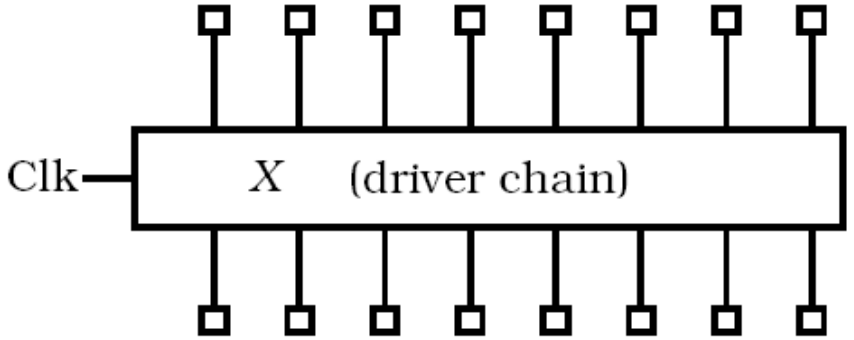
[Restle98]

Driver tree is normally routed in multiple metal layers

Single driver tree with multiple outputs.



(a) Driver tree



(b) Chip distribution

Example: DEC Alpha 21164

Clock Frequency: 300 MHz - 9.3 Million Transistors

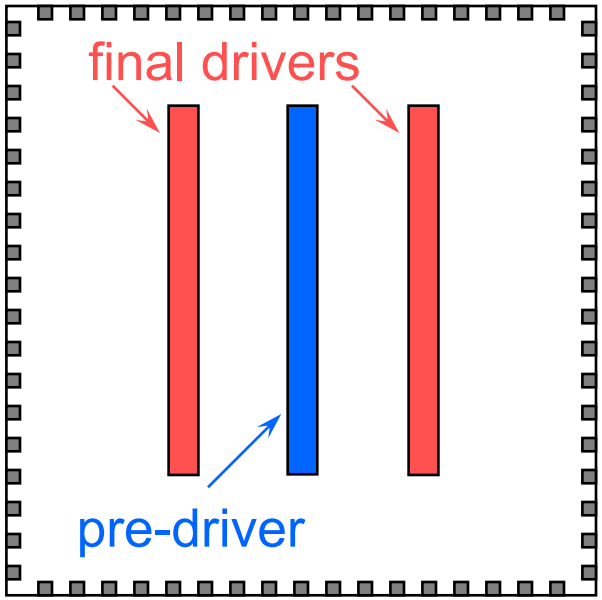
Total Clock Load: 3.75 nF

Power in Clock Distribution network : 20 W (out of 50)

Uses Two Level Clock Distribution:

- **Single 6-stage driver at center of chip**
- **Secondary buffers drive left and right side clock grid in Metal3 and Metal4**

Total driver size: 58 cm!



Location of clock driver on die

Clock Skew in Alpha Processor