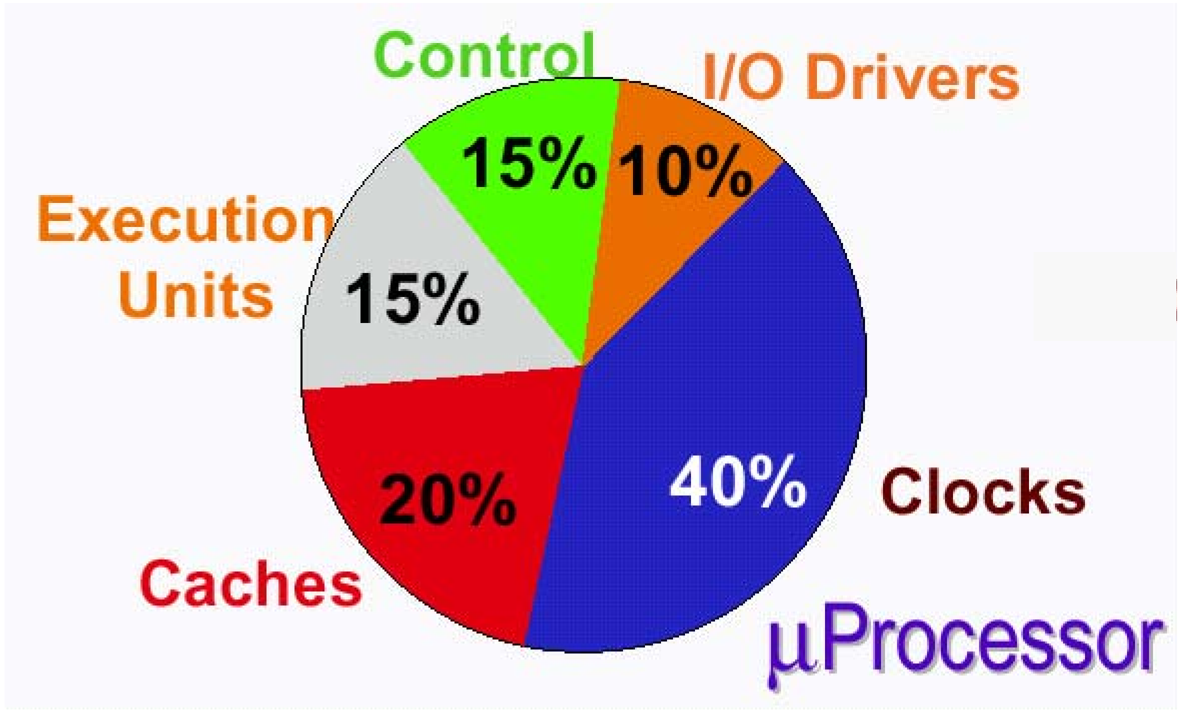
EECS 427
Lecture 18: Clocking, Timing/Latch Design

Reading: 10.3.1, 10.3.2, 7.4.1

Lecture Overview

- Last time: Timing and FlipFlop design
  - Setup-hold time, D-Q delay
- Today
  - Clock distribution
  - Impact of clock unceraínty
  - Noise sources relevant to FFs
  - Pulsed registers

Clocks: Power-Hungry



$$P = \alpha\ C\ V_{dd}^2\ f$$

Not only is the clock capacitance large, it switches every cycle!

Clocks – Objective

- Ensure that timing elements see the clock signal with zero relative phase difference.
- Clock signal seen at all timing-elements must have exactly the same frequency.(Which is why almost ALL clock distribution methods simply distribute one signal all over the chip)
- Control dissipation in the clock network, which toggles 2 times per cycle.
- Minimize area overhead of such a clock system.

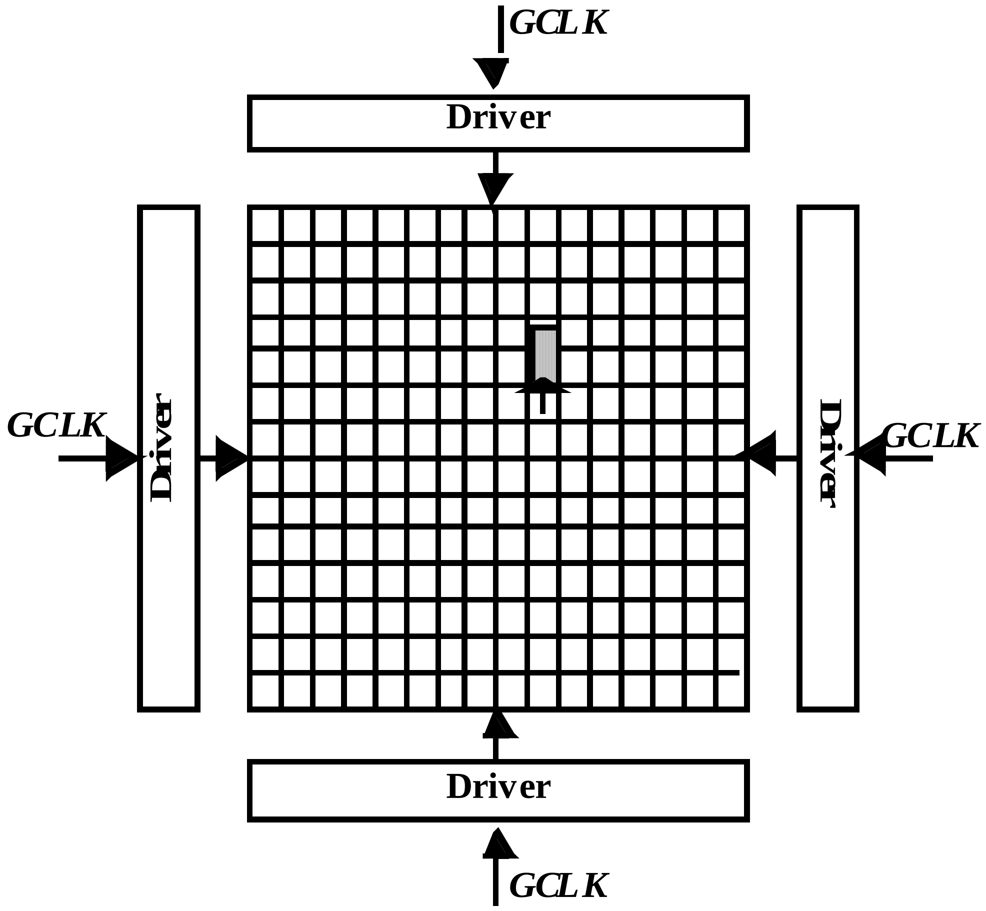
Clock Distribution Metric: Area

- Clock networks consume silicon area (clock drivers, PLL, etc.) and routing area
- Routing area is most vital
- Top-level metals are used to reduce RC delays
  - These levels are precious resources (unscaled)
  - Power routing, clock routing, key global signals
- By minimizing area used, we also reduce wiring capacitance & power
- Typical #’s: Intel Itanium – 4% of M4/5 used in clock routing

Slew Rates

- To maintain signal integrity and latch performance, minimum slew rates are required
- Too slow – clock is more susceptible to noise, process-variation, latches are slowed down, eats into timing budget
- Too fast – burning too much power, overdesigned network, enhanced ground bounce
- Rule-of-thumb: T_{rise} and T_{fall} of clock are each between 10-20% of clock period (10% - aggressive target)
  - 1 GHz clock; $T_{rise} = T_{fall} = 100\text{-}200\text{ps}$

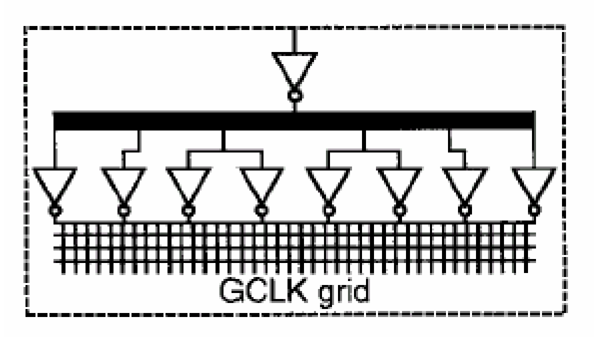
The Grid System



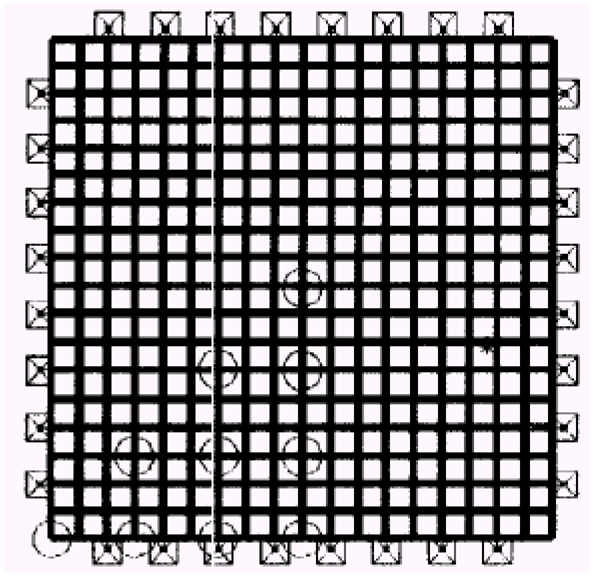
- No matching
- Large power (huge drivers)

Network Types: Grid

- Gridded clock distribution was common on earlier DEC Alpha microprocessors
- Advantages:
  - Skew determined by grid density and not overly sensitive to load position
  - Clock signals are available everywhere
  - Tolerant to process variations
  - Usually yields extremely low skew values



Pre-drivers



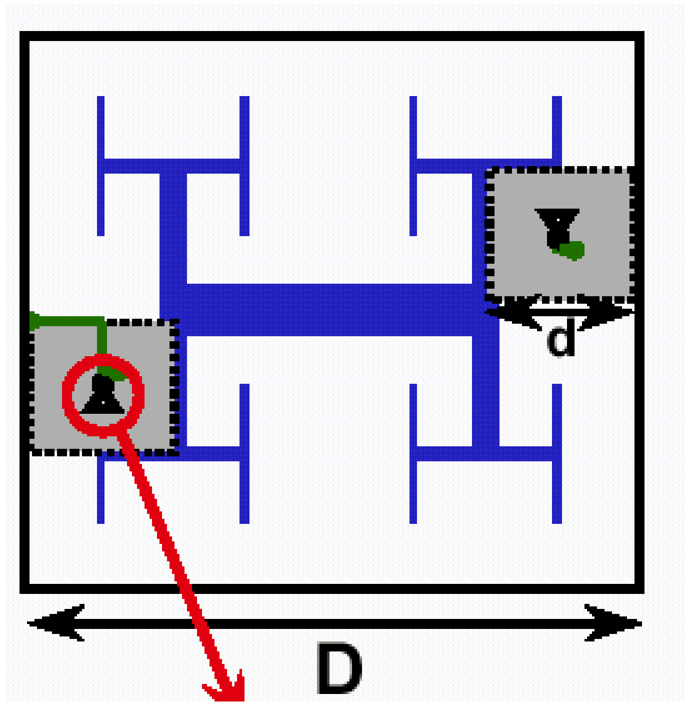
Global grid

Grid Disadvantages

- Huge amounts of wiring & power
  - Wire cap large
  - Strong drivers needed – pre-driver cap large
  - Routing area large
- To minimize all these penalties, make grid pitch coarser
  - Skew gets worse
  - Losing the main advantage
- Don’t overdesign – let the skew be as large as tolerable
- Grids aren’t feasible for most designs due to power

Network Types: Tree

- Original H-tree (Bakoglu)
  - One large central driver
  - Recursive H-style structure to match wirelengths
  - Halve wire width at branching points to reduce reflections

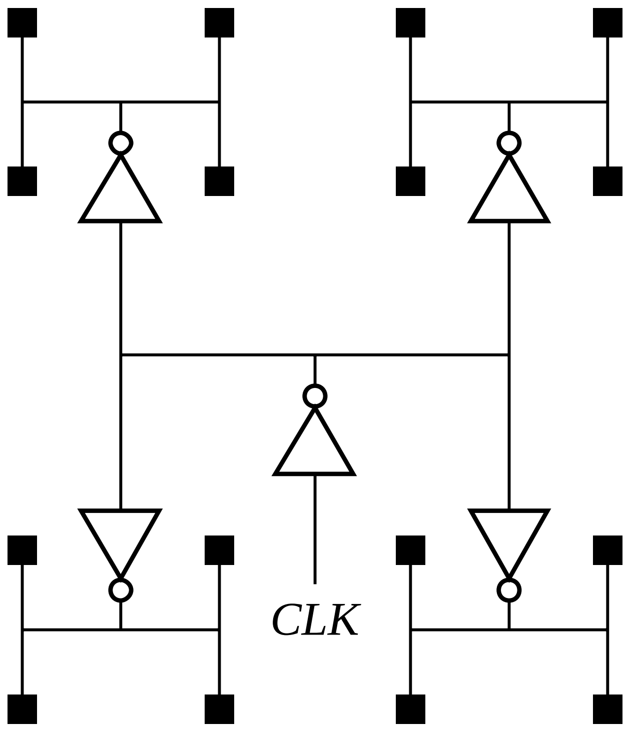


H-Tree Problems

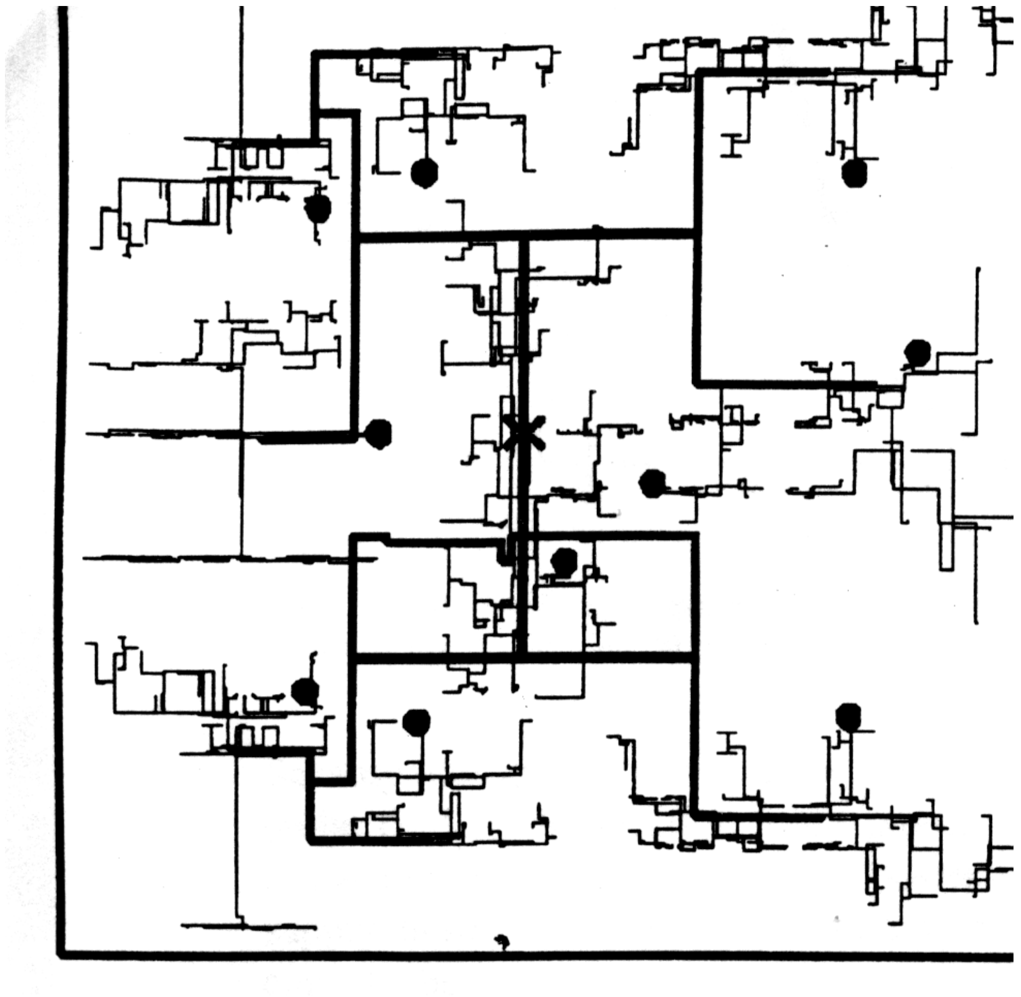
- Drawback to original tree concept
  - slew degradation along long RC paths
  - unrealistically large central driver
    - Clock drivers can create large temperature gradients (ex. Alpha 21064 ~30° C)
  - non-uniform load distribution
- Inherently non-scalable (wire resistance skyrockets)
- Solution to some problems
  - Introduce intermediate buffers along the way
  - Specifically at branching points

Buffered H-tree

- Advantages
  - Ideally zero-skew
  - Can be low power (depending on skew requirements)
  - Low area (silicon and wiring)
  - CAD tool friendly (regular)
- Disadvantages
  - Sensitive to process variations
  - Local clocking loads are inherently non-uniform

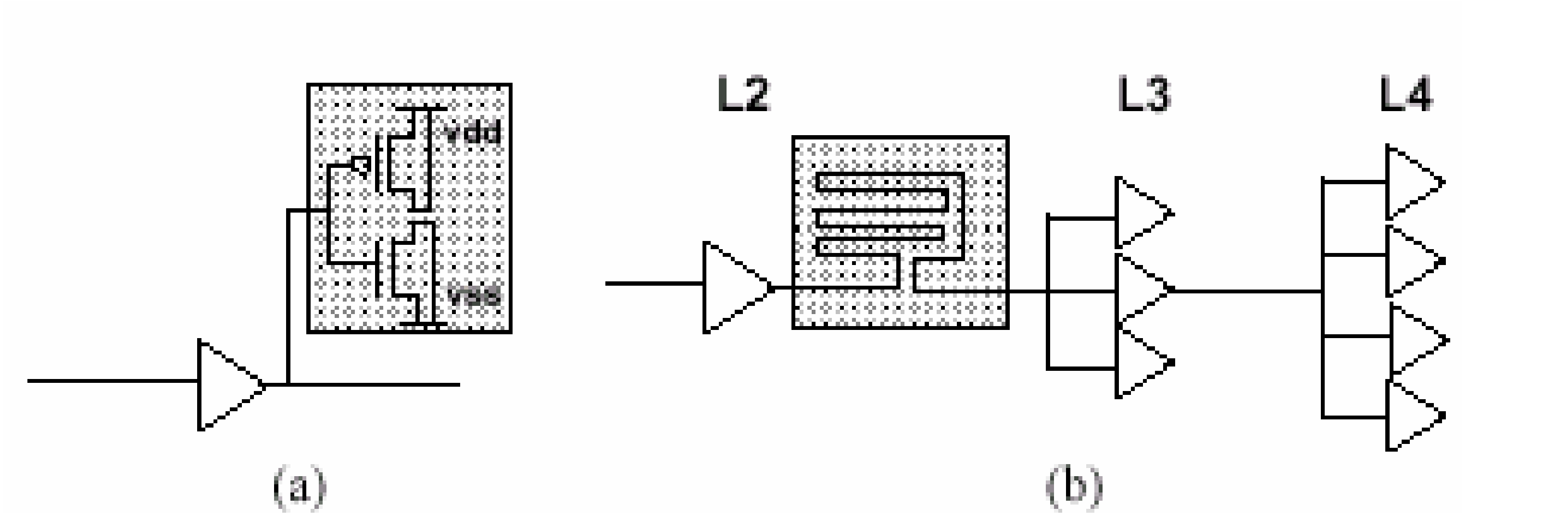


Realistic H-tree



[Restle98]

Balancing a Tree



Con: Routing area often more valuable than silicon

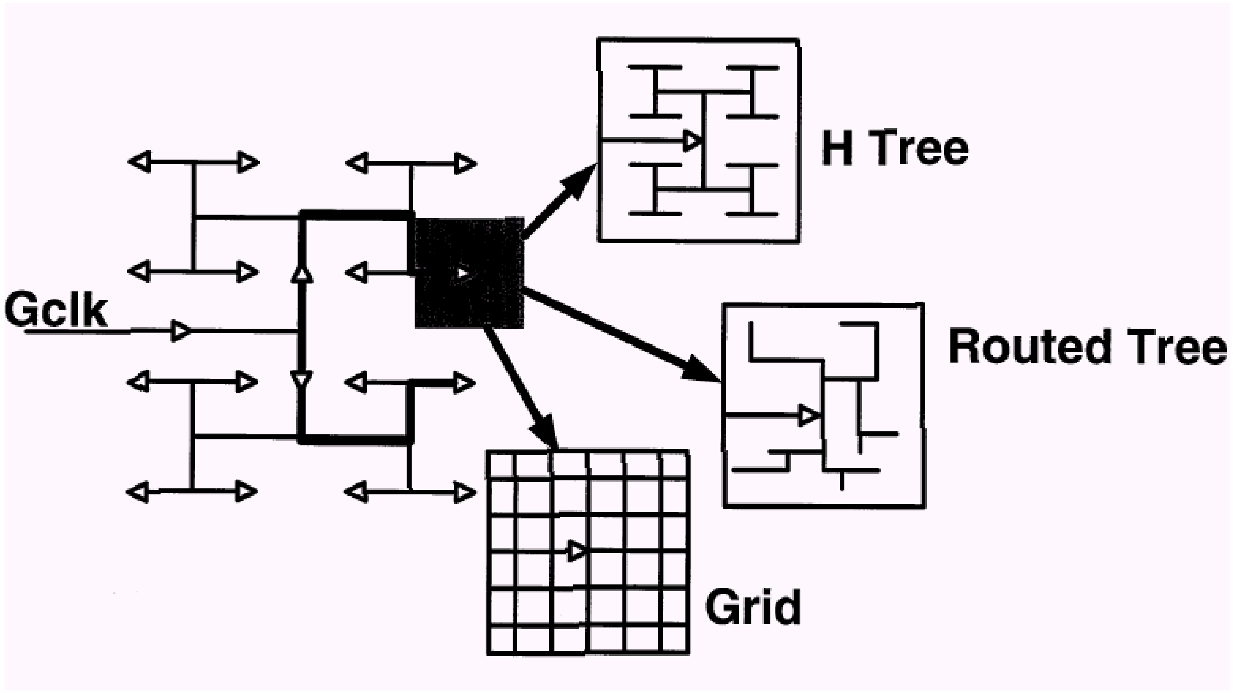
Some techniques:

a) Introduce dummy loads

b) Snaking of wirelength to match delays

Network of choice in high-performance

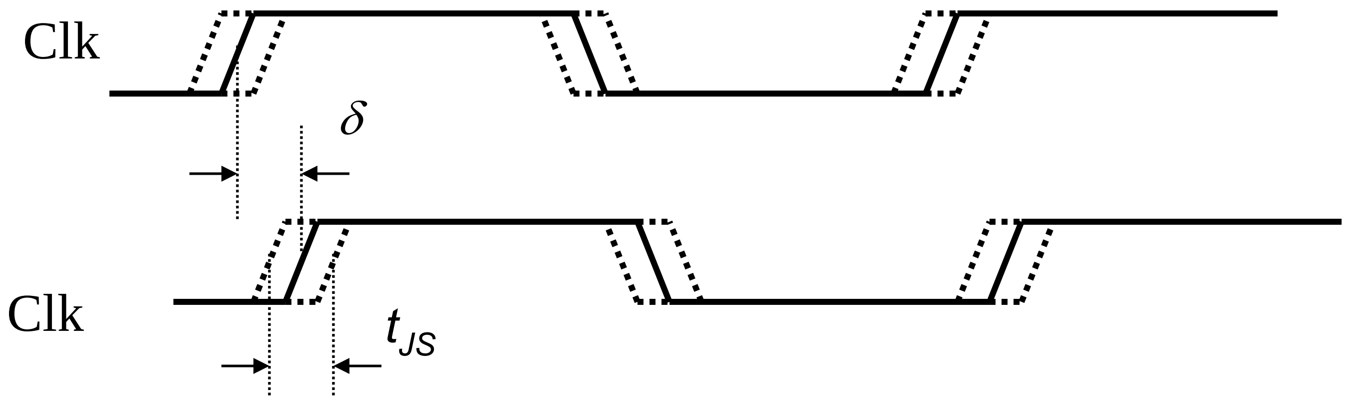
- Globally – Tree
- Why?
- Power requirements are reduced compared to global grid
  - Smaller routing requirements, frees up global tracks
- Trees are easily balanced at the *global* level
  - Keeps global skew low (with minimal process variation)



Clock Waveform Nonidealities

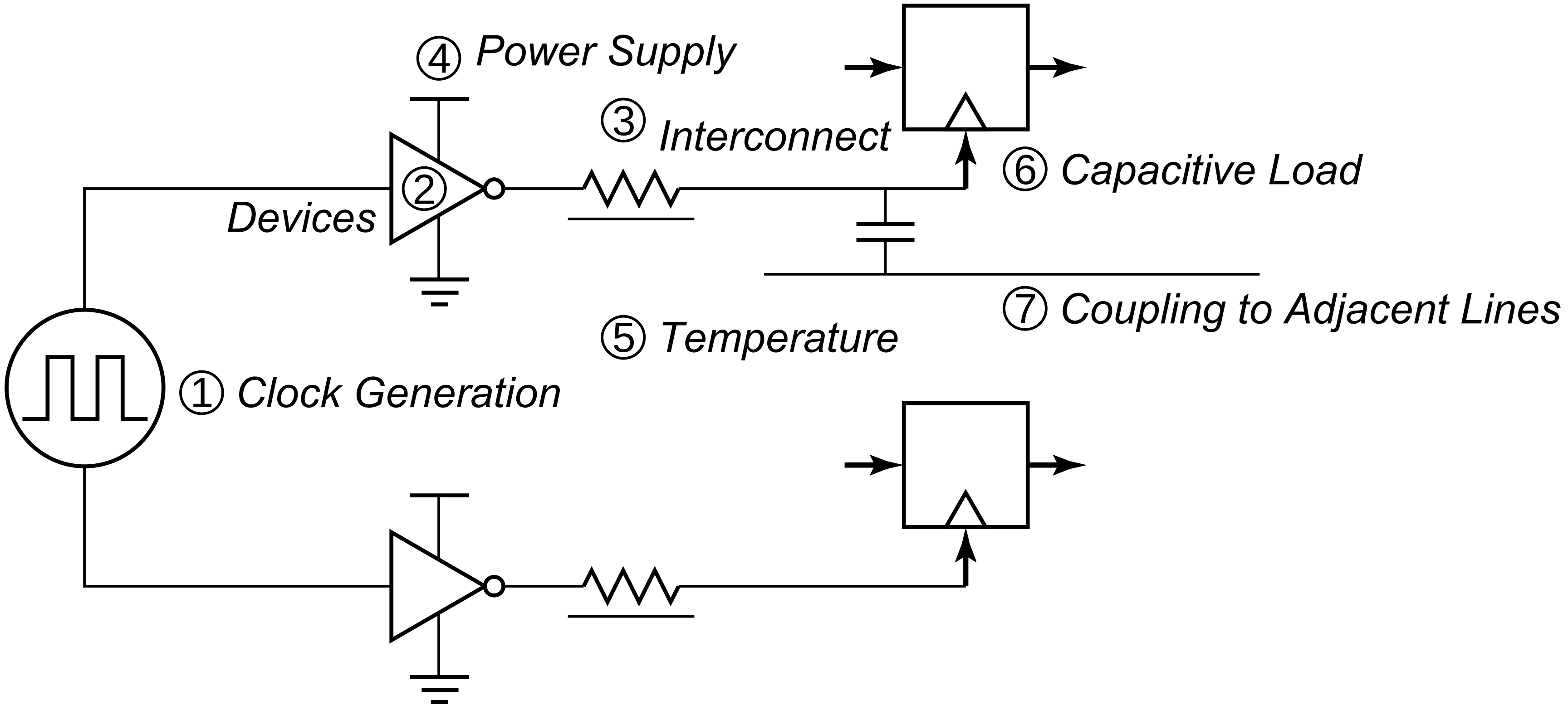
- **Clock skew**
  - Spatial variation in temporally equivalent clock edges, δ
- **Clock jitter**
  - Temporal variations in consecutive edges of the clock signal
    - Cycle-to-cycle (short-term) t_{JS}
    - Long term t_{JL}
- **Variation of the pulse width (duty cycle)**
  - Important for level sensitive (latch-based) clocking

Clock Skew and Jitter



- Both skew and jitter impact the effective cycle time
- Skew *can* be useful. Setup time – Hold time tradeoff
- Jitter always degrades performance

Clock Uncertainties

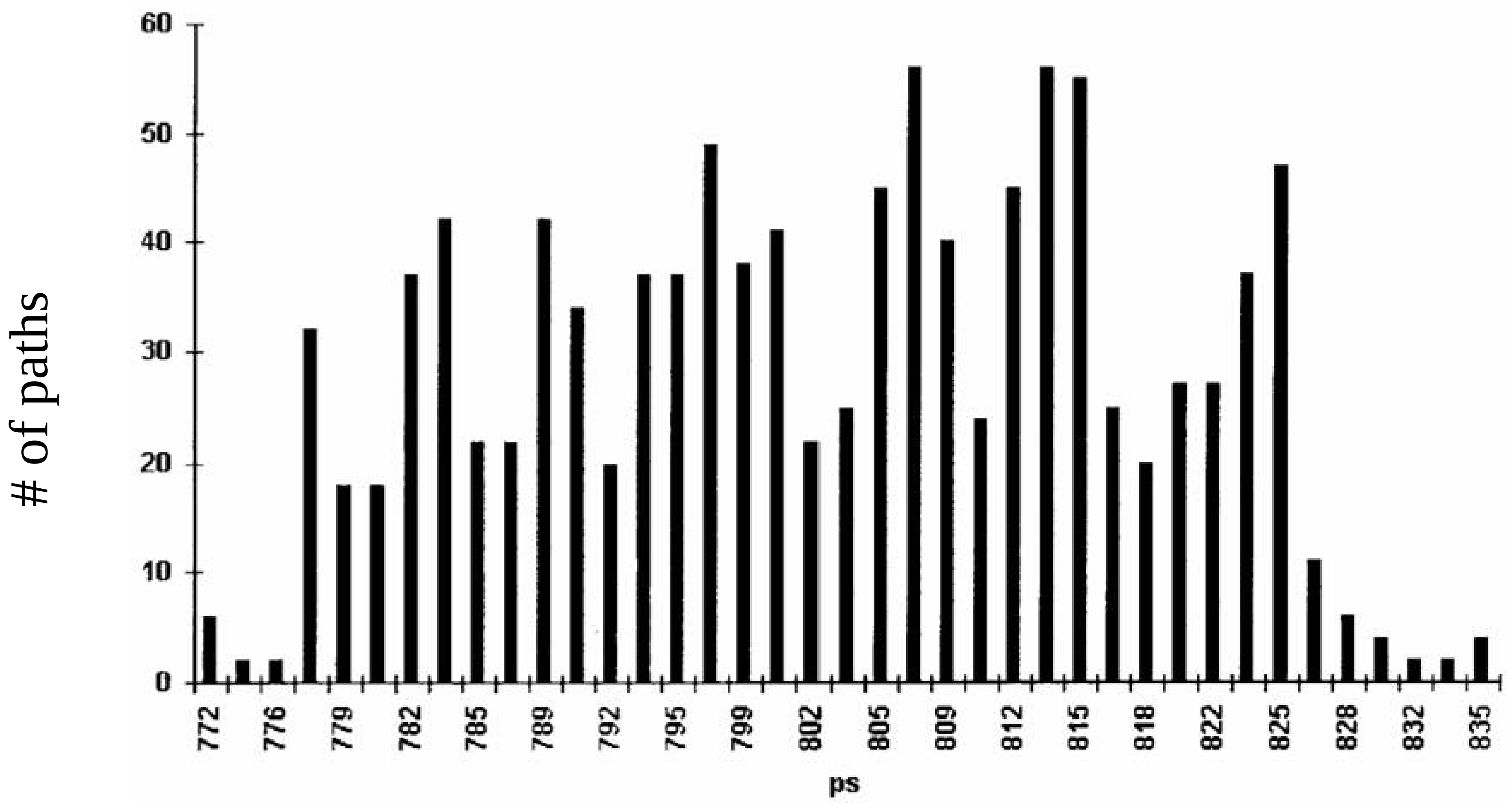


Sources of clock uncertainty

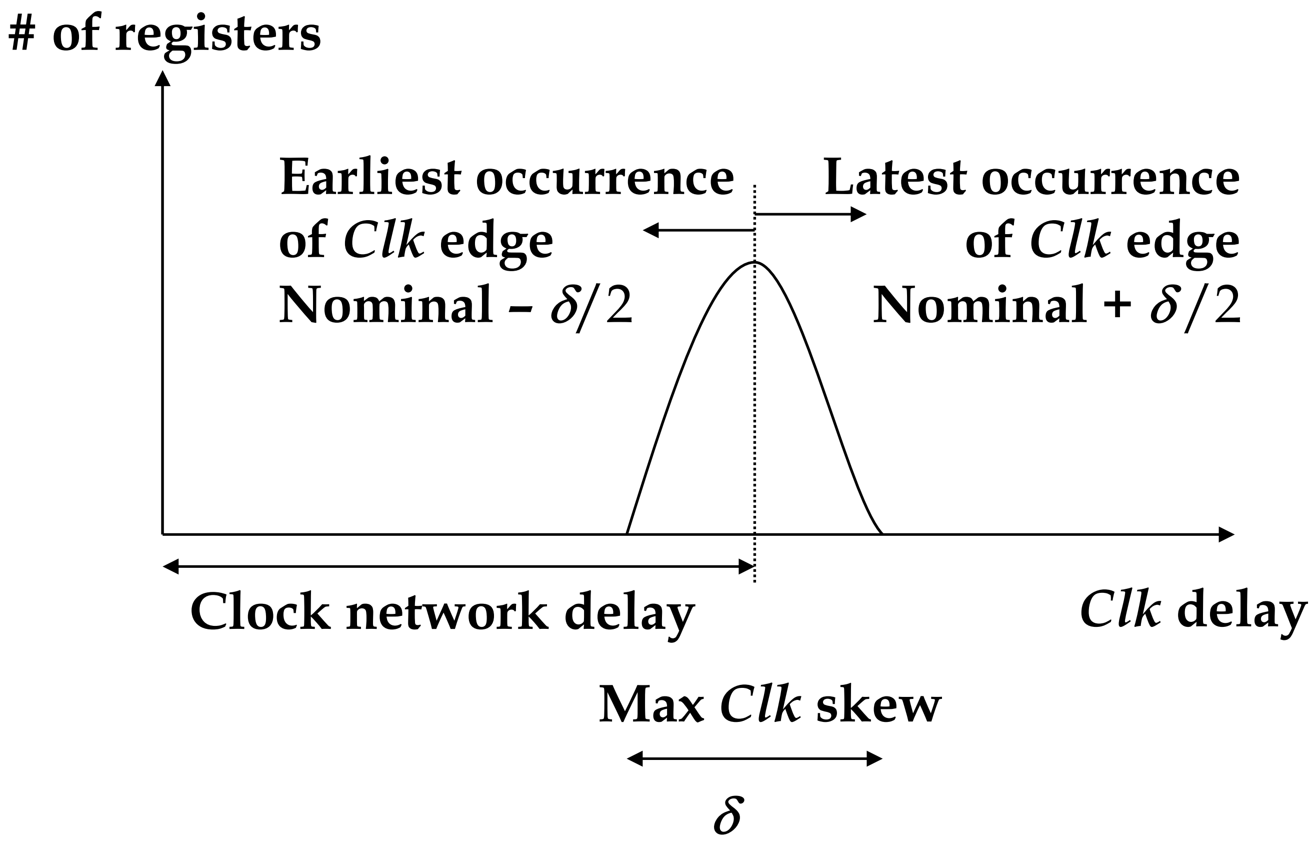
Jitter Sources

- Caused by variations in clock period that result from:
  - Ring oscillator clocks can be quite “jittery”.
  - Phased-lock loop (PLL) oscillation frequency
  - Various noise sources affecting clock generation and distribution
    - Ex. Power supply noise which dynamically alters the drive strength of intermediate buffer stages. (A “derivative effect”??)
    - What happens with a ring oscillator clock?

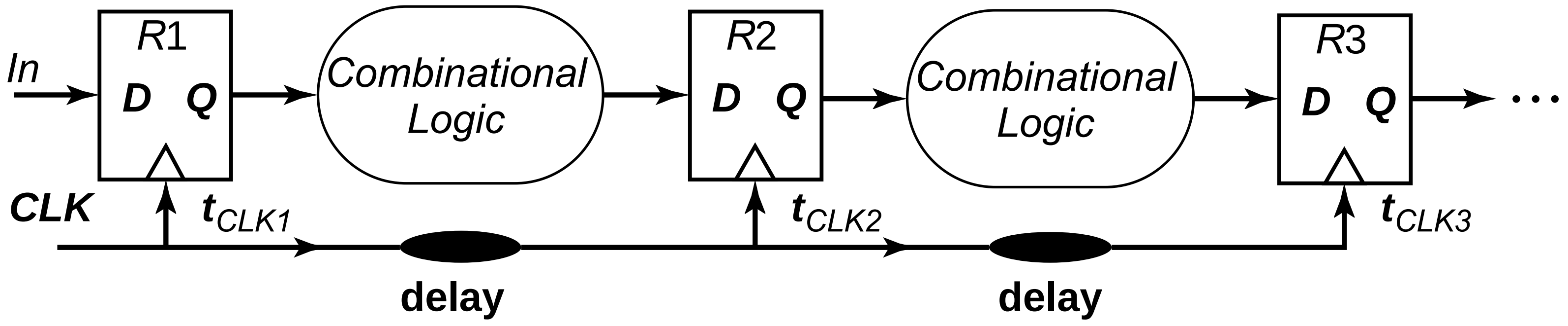
IBM microprocessor clock skew



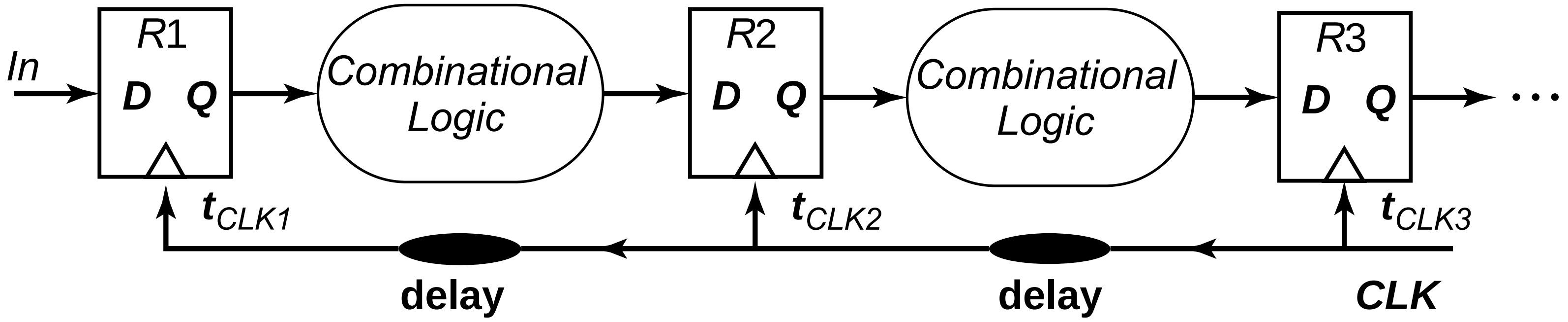
Idealized View of Clock Skew



Positive and Negative Skew

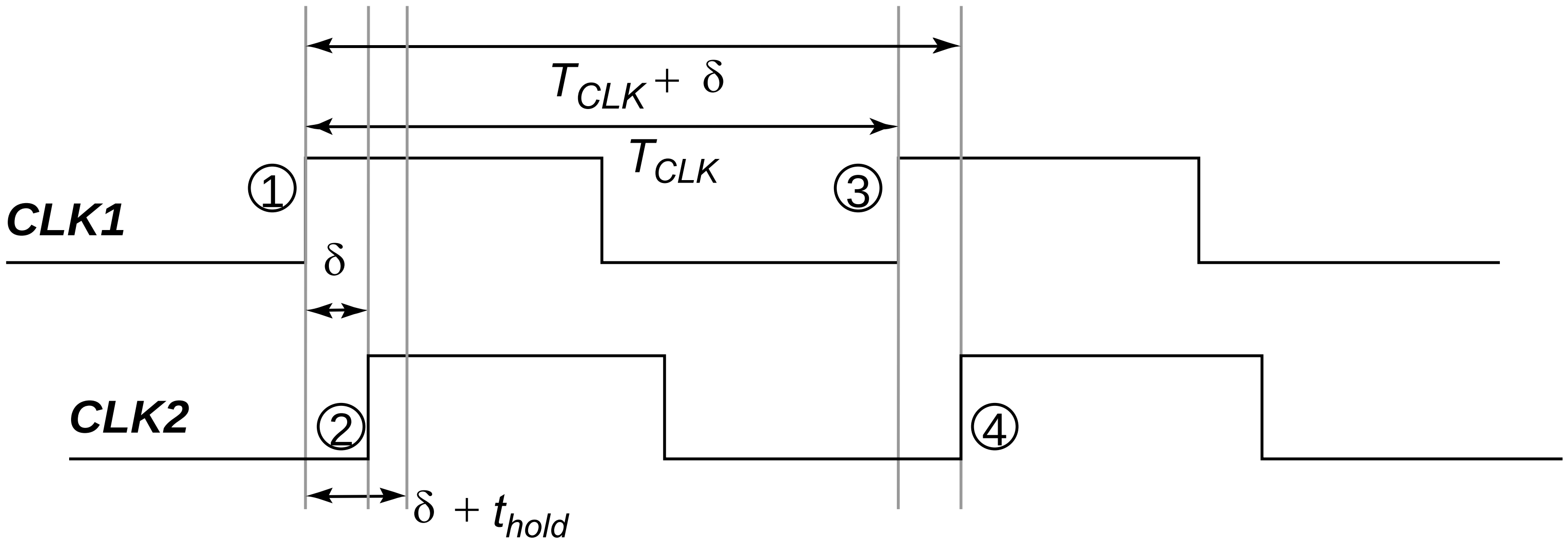


(a) Positive skew



(b) Negative skew

Positive Skew, δ > 0



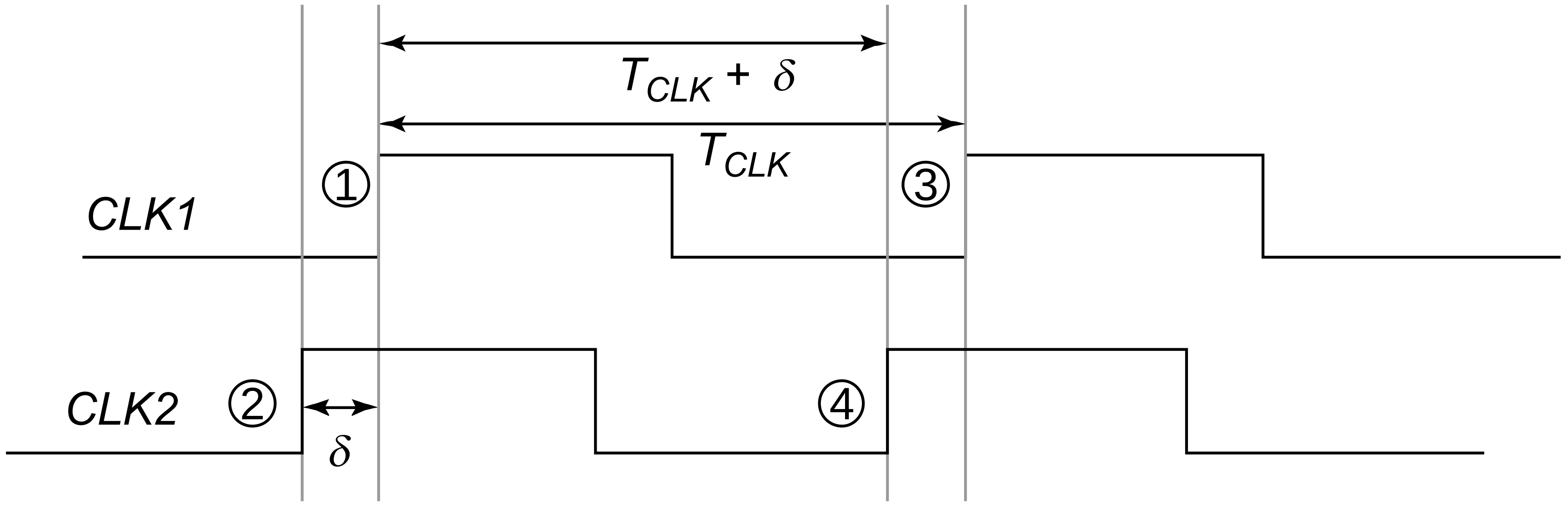
Launching edge arrives before the receiving edge

Good for performance, bad for hold time

Key: Hold time violations cannot be fixed by running the clock slower!

Question: Will it help if I scale the voltage??

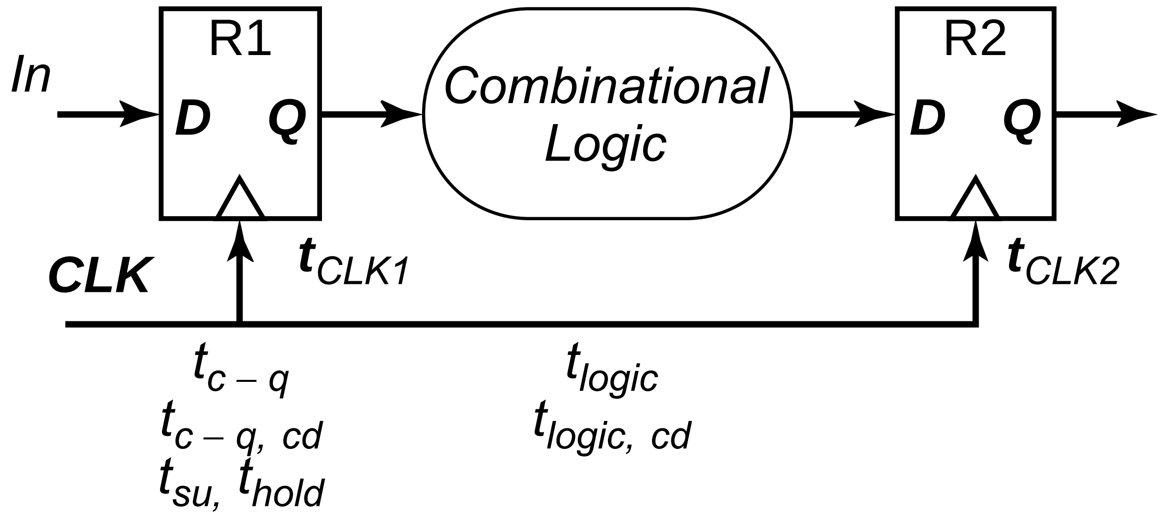
Negative Skew, $\delta < 0$



Receiving edge arrives before the launching edge

Bad for performance, good for hold time violations

Timing Constraints

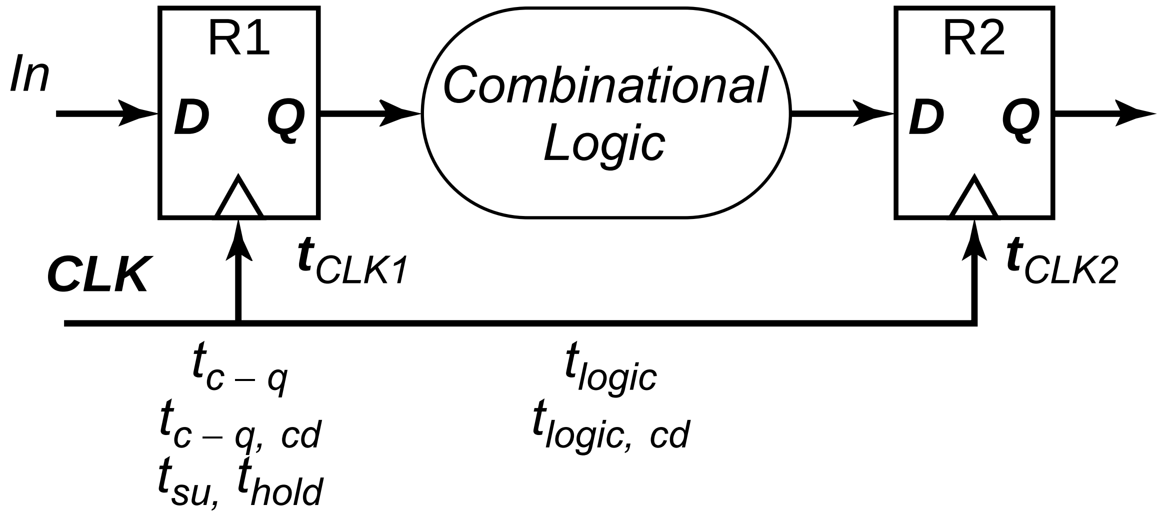


Minimum cycle time:

$T \geq t_{c\text{-}q} + t_{su} + t_{logic} - \delta$

Worst case is when receiving edge arrives early (negative δ)

Timing Constraints



Hold time constraint:

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

Worst case is when receiving edge arrives late (positive skew)
Race between data and clock
cd: contamination delay (fastest possible delay)

Requirements in Flip-Flop Design

- High speed:
  - Small Clk-Output delay
  - Small setup time
  - Small hold time$\rightarrow$Inherent race immunity
- Low power
- Small clock load (clock power is very large)
- High driving capability
- Integration of the logic into flip-flop
- Multiplexed or clock scan (testability)
- Robustness
- Crosstalk insensitivity
  - Dynamic/high impedance nodes are affected

Sources of Noise

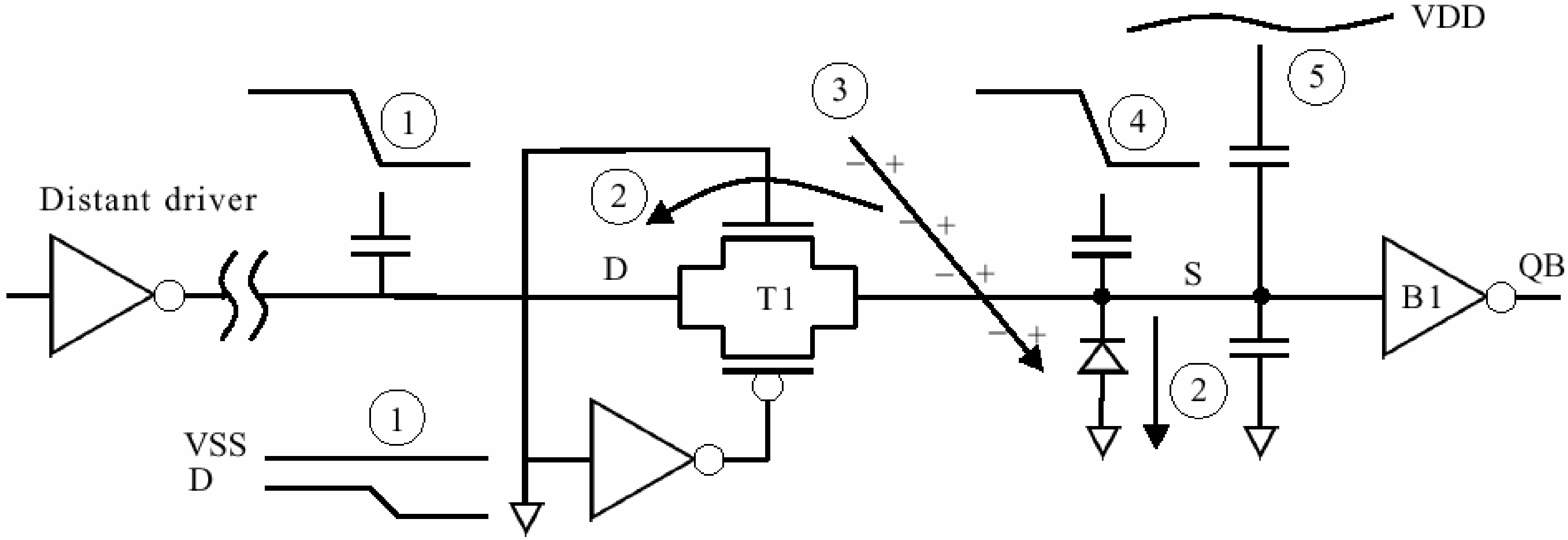
① Noise on input

② Leakage

③ α-Particle and cosmic rays

④ Unrelated signal coupling

⑤ Power supply ripple



Courtesy of IEEE Press, New York. © 2000

Flip-Flop Robustness

- Input isolation
  - Don’t use a pass-transistor directly at the input (use a buffer)

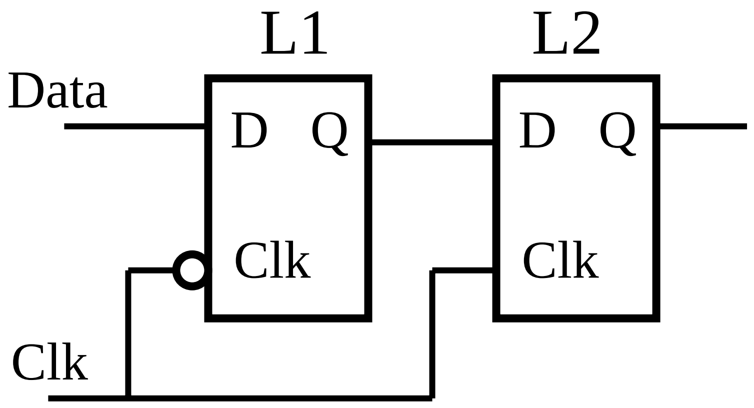
Storage node related issues:

- Robustness
  - No floating nodes, create pseudo-static storage nodes
- Min capacitance limit
  - Storage node (middle of cross-couple) should have a decent amount of capacitance for noise immunity
  - Too much will slow things down though…
- Preventing exposure
  - Wires associated with storage node should be short, suppress possible coupling to other nodes

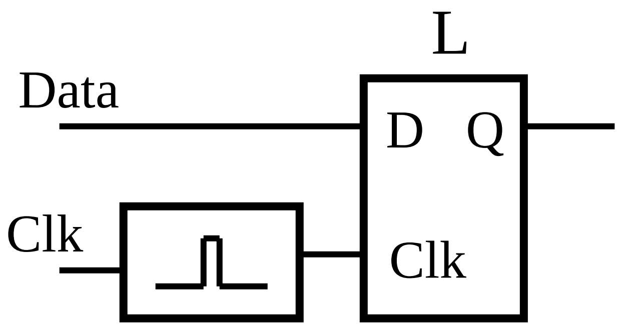
Pulse-Triggered Registers An Alternative Approach

Ways to design an edge-triggered sequential cell:

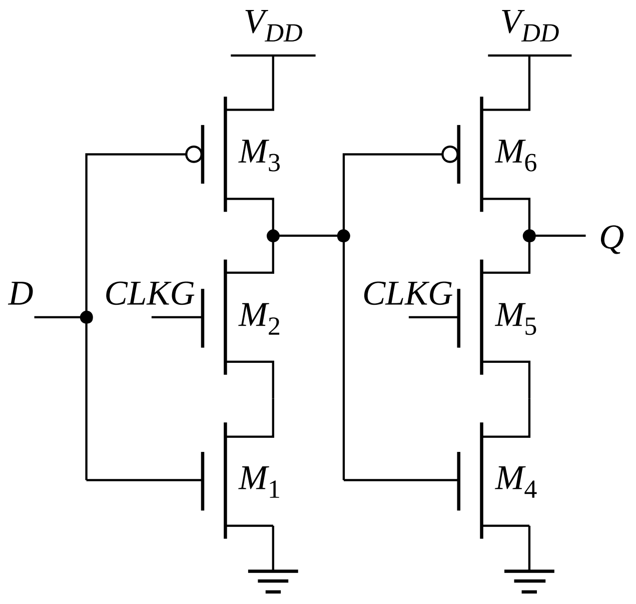
Master-Slave Latches



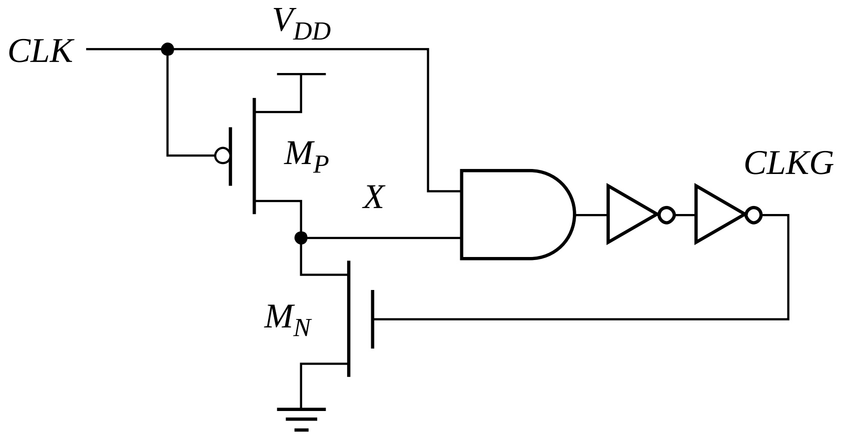
Pulse-Triggered Latch



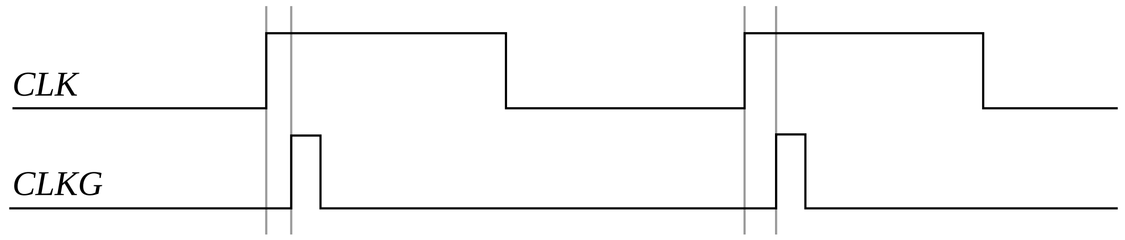
Pulsed Register



(a) register



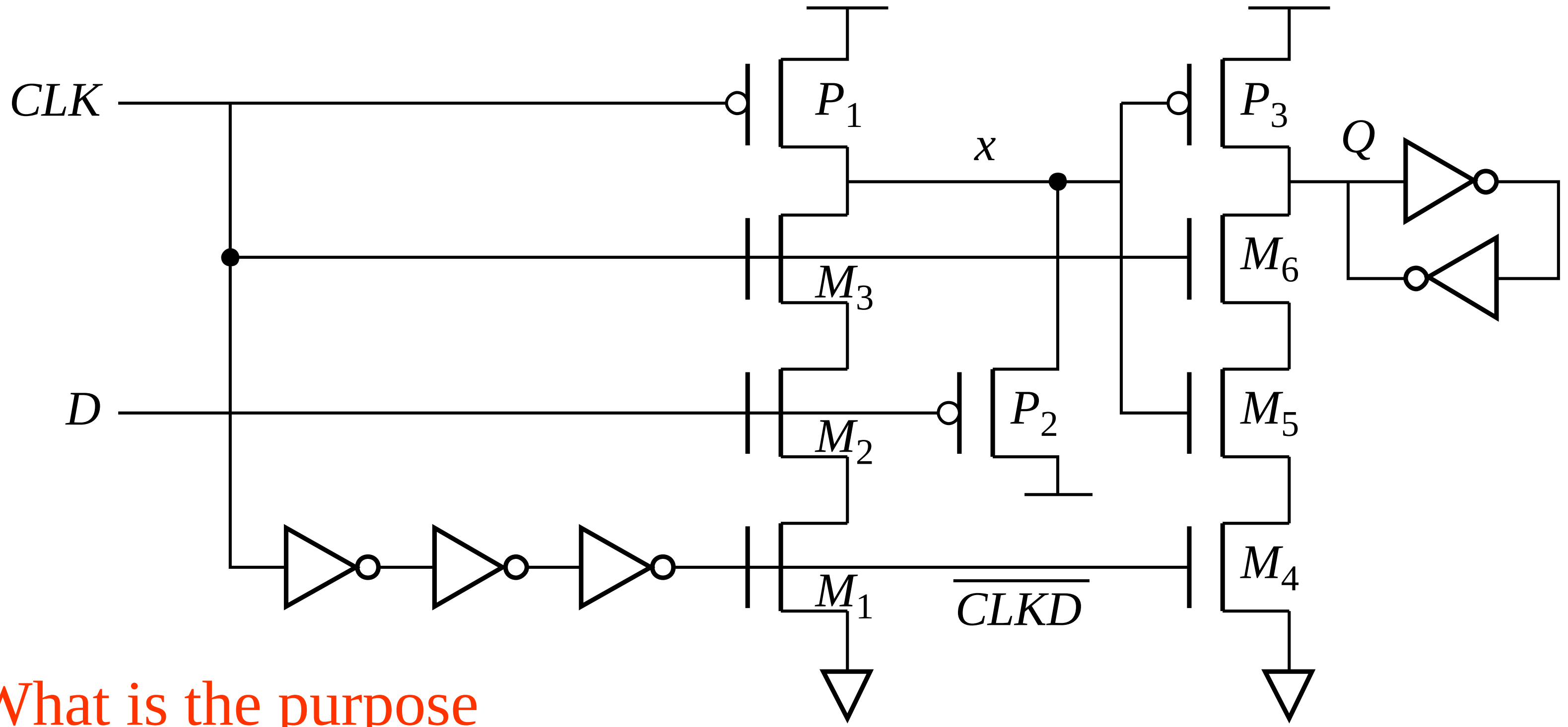
(b) glitch generation



(c) glitch clock

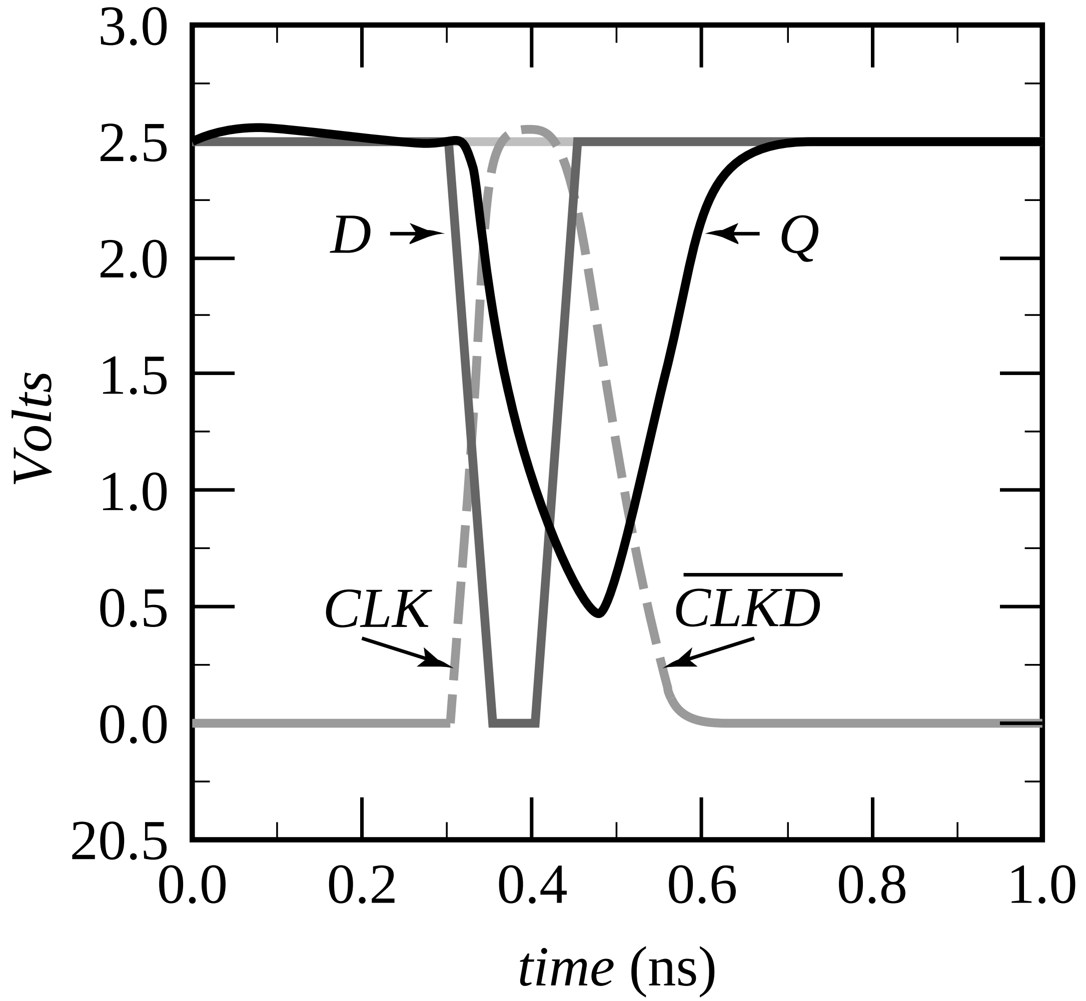
Another Pulsed Register Topology

Hybrid Latch – Flip-flop (HLFF), AMD K-6 and K-7:



What is the purpose of P_2. Is it fulfilled?

Pulsed Register Timing Diagram



Negative setup times

Fast CLK-Q delays

Limited transparency window (similar to master-slave topology)

Large hold times

Summary

- Clocks strongly impact IC performance (timing) and are not ideal
  - Skew and jitter are commonly discussed non-idealities
  - Skew is typically larger and more heavily focused on
  - More on skew later in class when we discuss clock distribution techniques
- Rough rule of thumb: skew should be kept < 10% of clock period
- Sequential elements eat up a significant amount of total timing budget + power resources
  - They are therefore extremely important to design carefully
  - Robustness is critical as well
- D-Q delay is best overall performance measure for edge-triggered registers since it effectively combines both setup and CLK-Q delays
- New designs like pulsed registers provide enhanced performance with some added design complexity