ECE 225
High Speed Integrated Circuit Design

Lecture 12

Prof. Kaustav Banerjee
Electrical and Computer Engineering
E-mail: kaustav@ece.ucsb.edu

Timing Issues

Timing Classification of Digital Systems

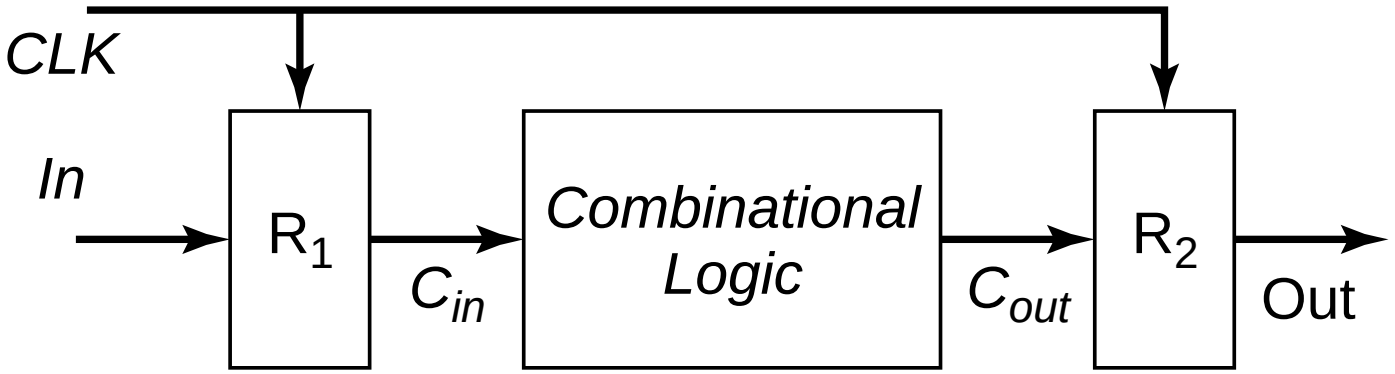
Synchronous: A signal that has exactly the same frequency as the local CLK and has a known fixed phase offset to that CLK

Mesochronous: A signal that has the same frequency as the local CLK but has an unknown phase offset w.r.t. that CLK.

Plesiochronous: A signal that has a “slightly different” frequency compared to that of the local CLK. This causes the phase difference to drift in time.

Asynchronous: A signal that has no fixed relationship w.r.t. that of the local CLK. Asynchronous signals can transition any time.

Synchronous Timing



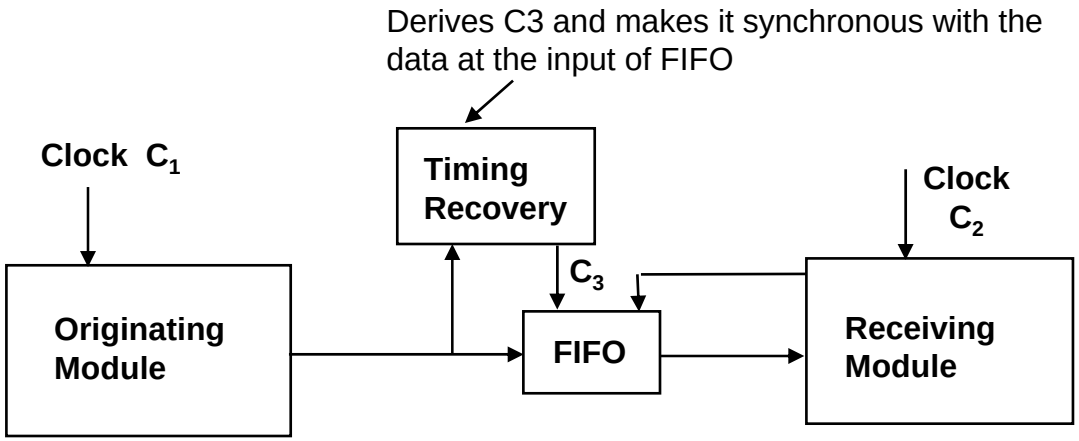
The certainty period of the signal C_{out}---the period during which data are valid is synchronized with the system CLK. Hence, R2 can sample the data with confidence.

Mesochronous Timing

- If data are being passed between two different CLK domains, the data signal transmitted from the first module can have an unknown phase relationship to the CLK of the receiving module
- Not possible to directly sample the output data at the receiving module because of uncertainty in the phase offset
- A synchronizer is needed to synchronize the data signal with the receiving CLK

Plesiochronous Timing

- Signal has slightly different frequency as compared to the local CLK---phase difference drifts over time
- Can arise due to the use of different CLK generators
- Practically, plesiochronous timing occurs in large distributed systems that involves long distance communications
- A timing recovery circuit is needed to ensure that all data are received



Plesiochronous Communication

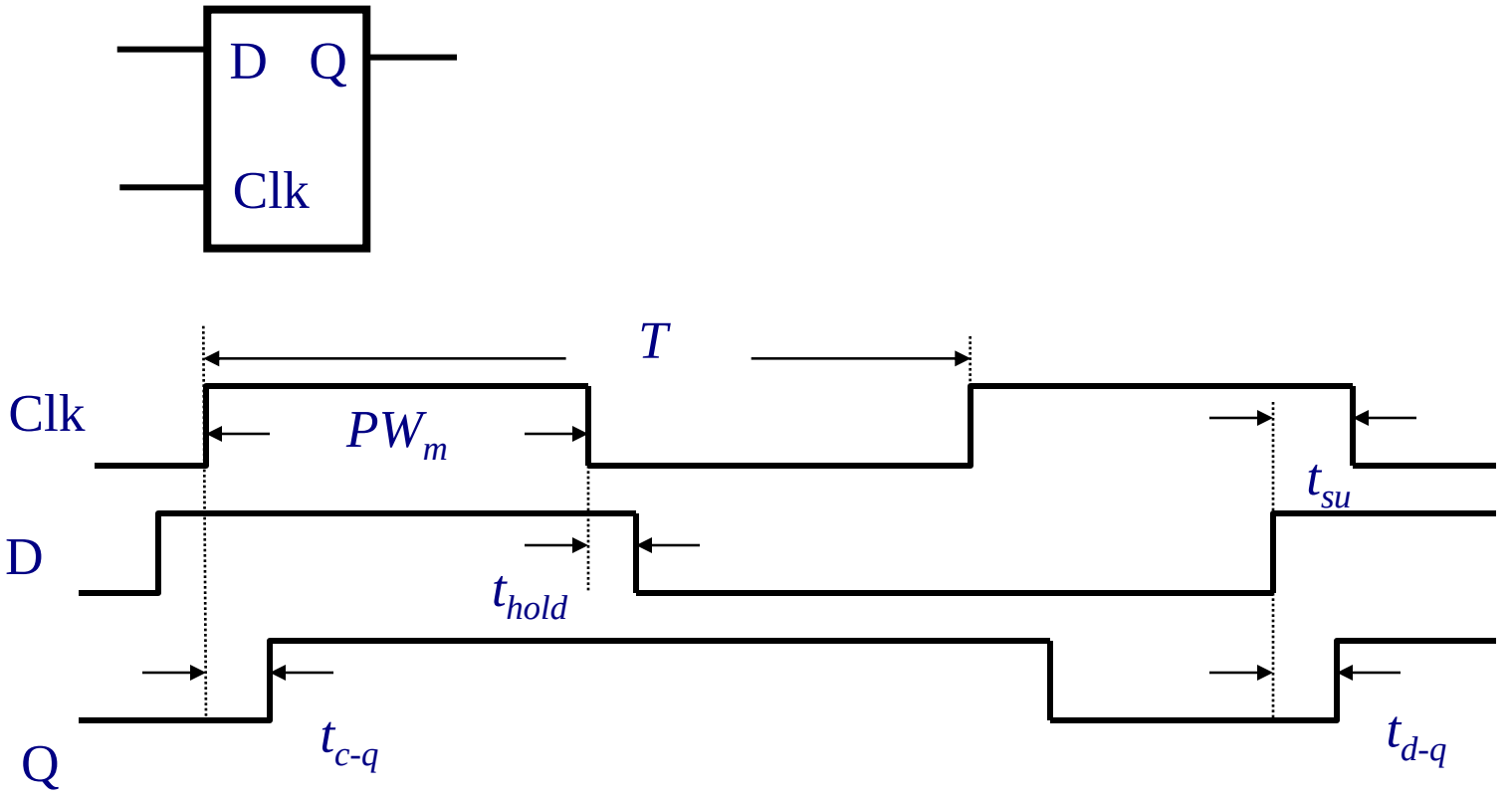
Note: FIFO is a serial memory element (eg., video memories)

Asynchronous Timing

- Signals transition at arbitrary times—not tied to a CLK
- Can synchronize asynchronous signals by detecting events and by introducing latencies into the data stream synchronized to a local CLK
- Alternatively, a self-timed asynchronous design approach may be adopted—communication between modules achieved not through a CLK but through a handshaking protocol that ensures proper ordering of operation
- Computations can be performed at speeds determined solely by the latency of the logic---no need to manage CLK skew!
- Increased complexity + overhead in communication, impacts performance. Also, CAD methodology is more complex.

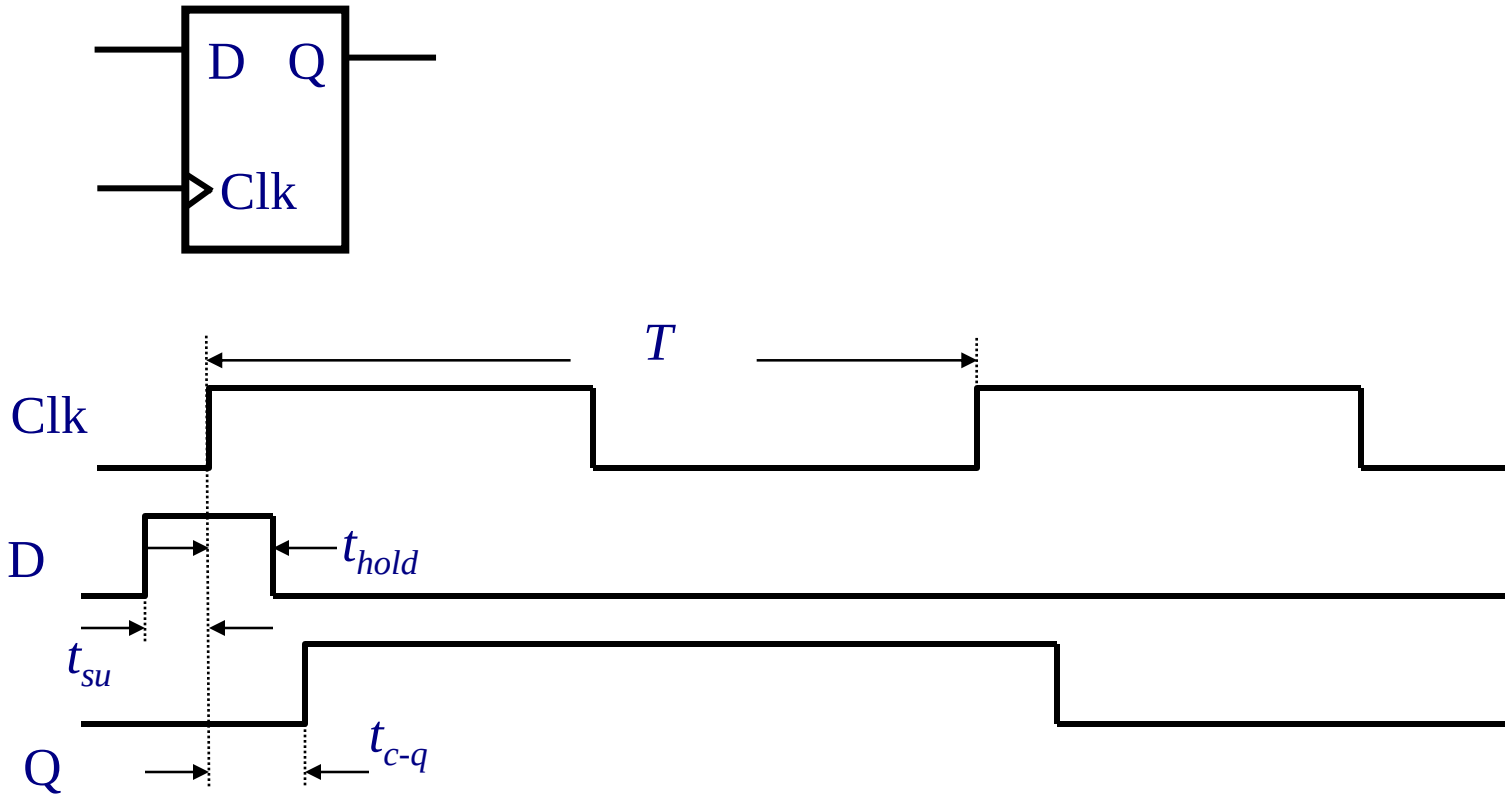
Timing Definitions

Latch Parameters



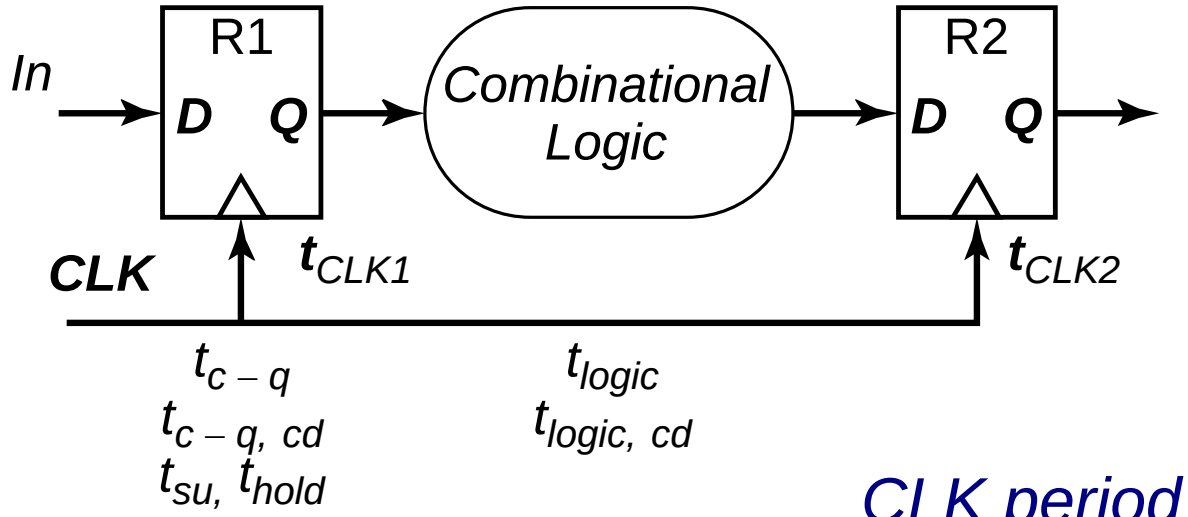
Delays can be different for rising and falling data transitions

Register Parameters



Delays can be different for rising and falling data transitions

Timing Constraints



CLK period constraint:

$$T_{CLK} > t_{c\text{-}q} + t_{logic} + t_{su}$$

Hold time constraint:

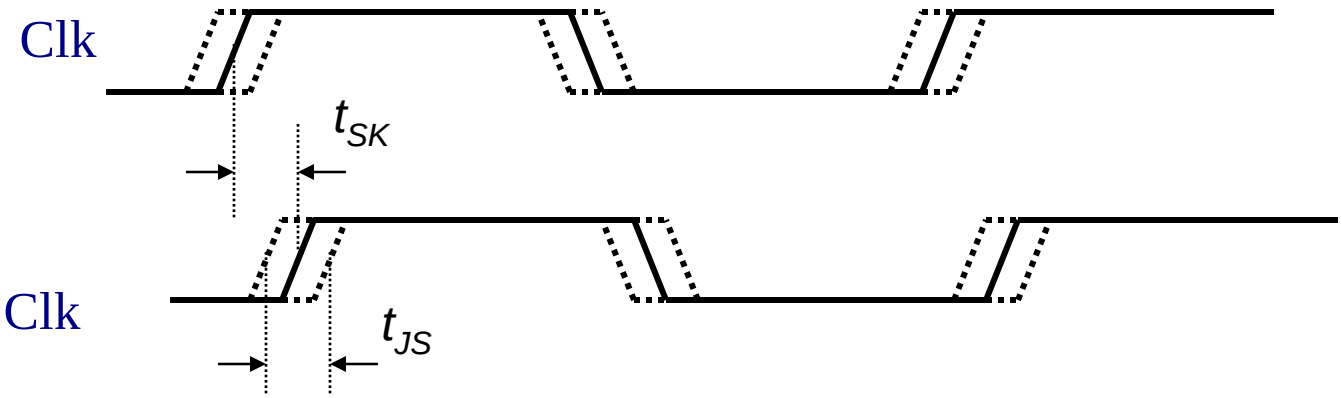
$$t_{(c\text{-}q,\ cd)} + t_{(logic,\ cd)} > t_{hold}$$

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

Clock Nonidealities

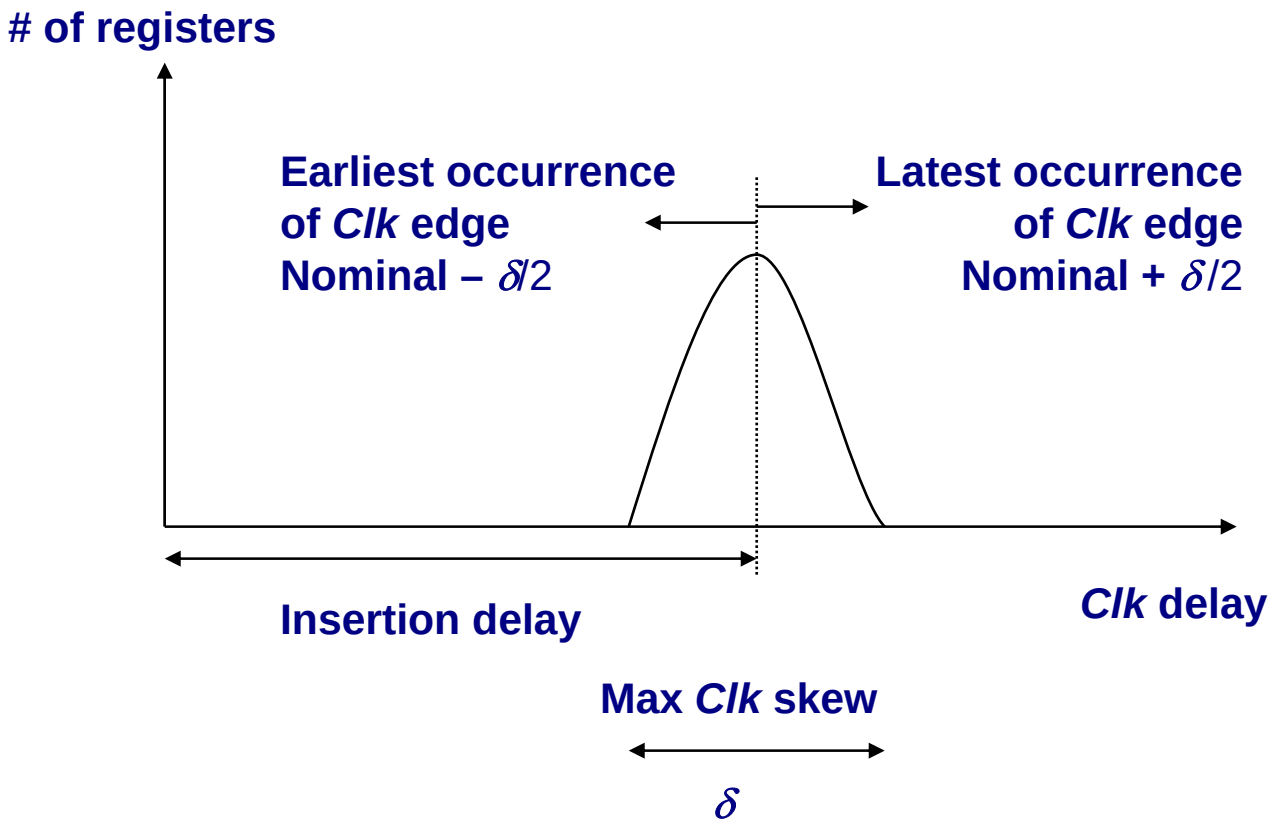
- **Clock skew**
  - Spatial variation in temporally equivalent clock edges; deterministic + random, t_{SK}
  - Constant from cycle-to-cycle
  - Results only in phase shift
- **Clock jitter**
  - Temporal variations in consecutive edges of the clock signal; modulation + random noise
  - Cycle-to-cycle (short-term) t_{JS}
  - Long term t_{JL}
- **Variation of the pulse width**
  - Important for level sensitive clocking

Clock Skew and Jitter

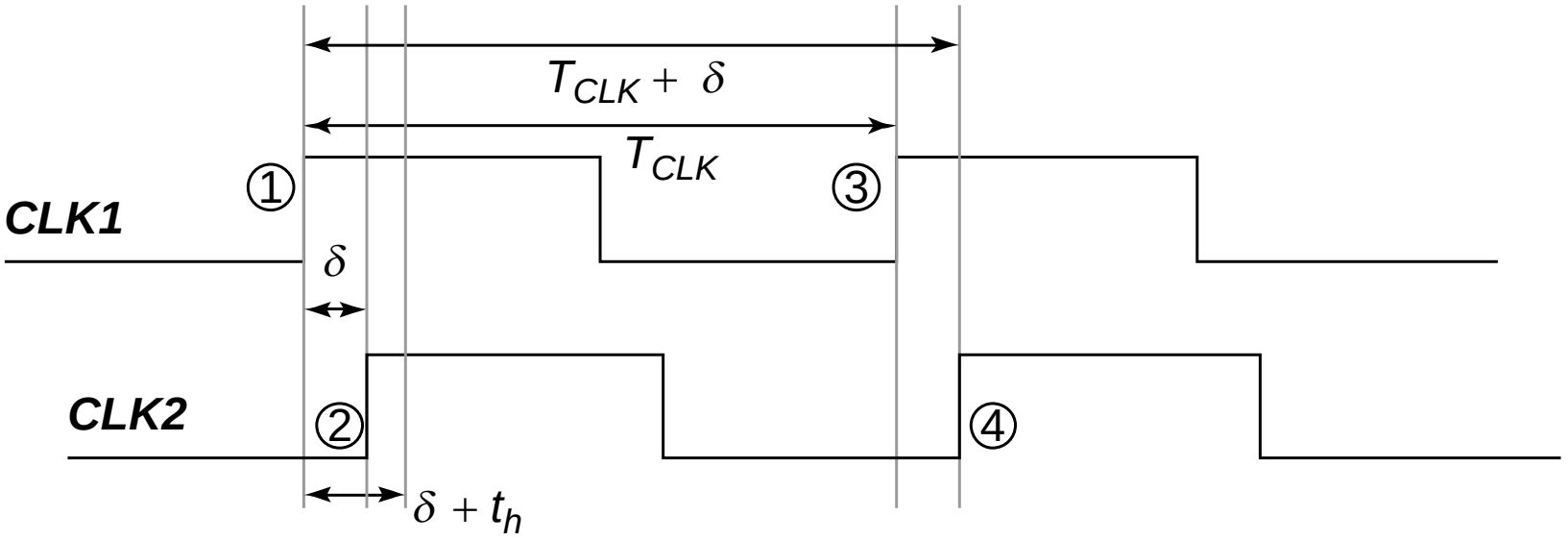


- Both skew and jitter affect the effective cycle time
- Only skew affects the race margin

Clock Skew

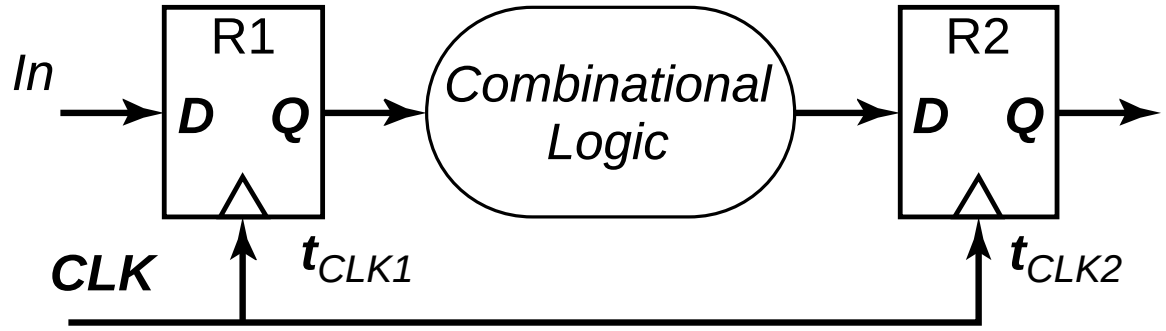

# of registers
Earliest occurrence of *Clk* edge Nominal – $\delta/2$
Latest occurrence of *Clk* edge Nominal + $\delta/2$
Insertion delay
Clk delay
Max *Clk* skew
δ

Positive Skew



- ***Launching edge arrives before the receiving edge***
- ***Time available for a signal to propagate from a register to an adjacent register is increased by the skew δ***

Timing Constraints: +ve Skew



Minimum cycle time:

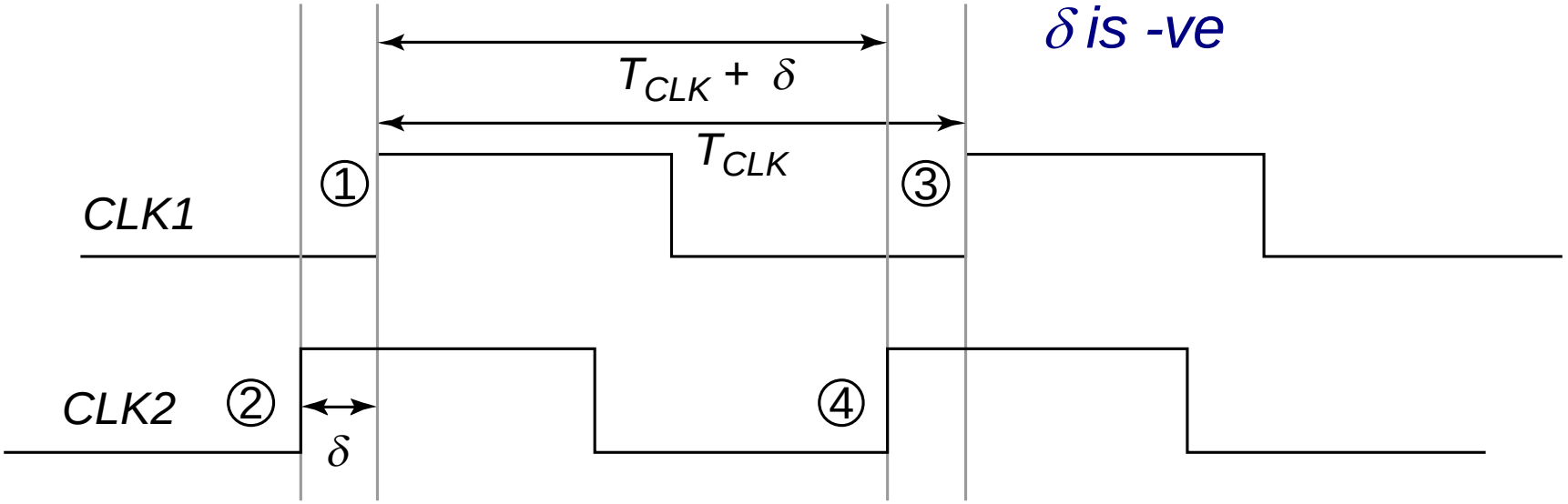
$$T_{CLK} + \delta = t_{c\text{-}q} + t_{su} + t_{logic}$$

CLK skew can improve performance?

Watch out for RACE problems….(if minimum logic delay is small, inputs to R2 can change before the CLK2 edge at 2)

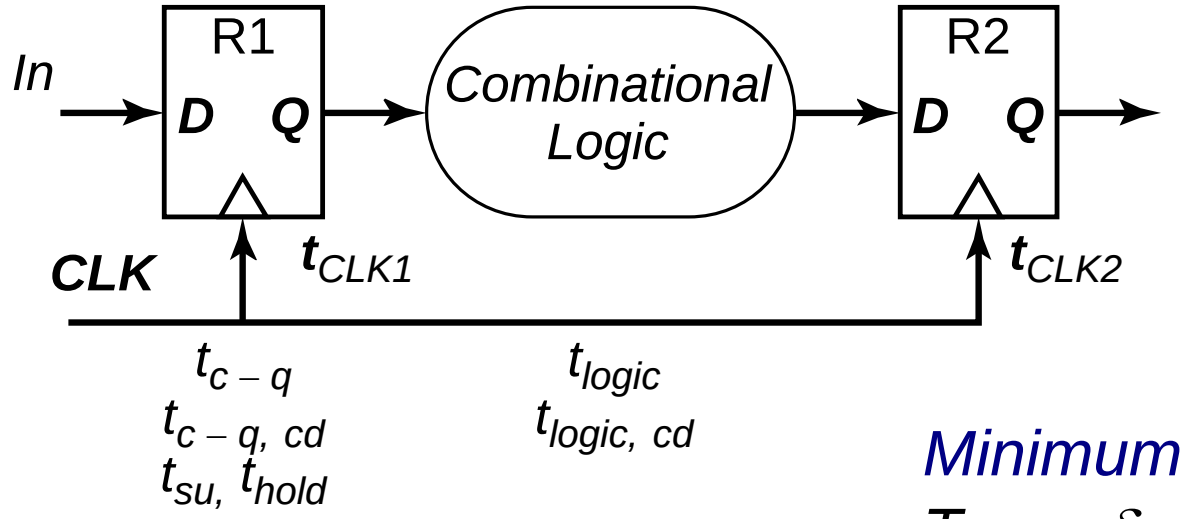
Minimum delay constraint: $\delta + t_{hold} < t_{c\text{-}q,cd} + t_{logic,cd}$

Negative Skew



Receiving edge arrives before the launching edge

Timing Constraints: -ve Skew



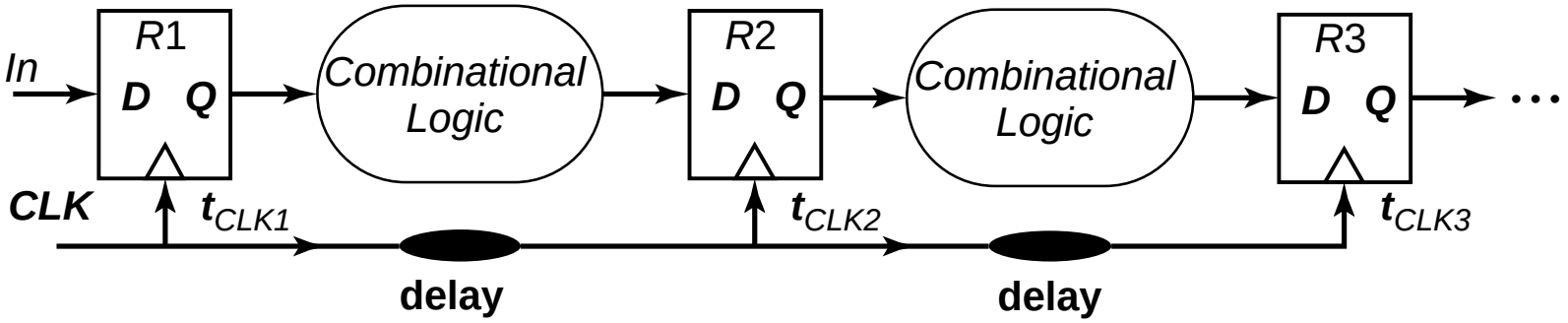
Minimum cycle time:

$$T_{CLK} - \delta = t_{c\text{-}q} + t_{su} + t_{logic}$$

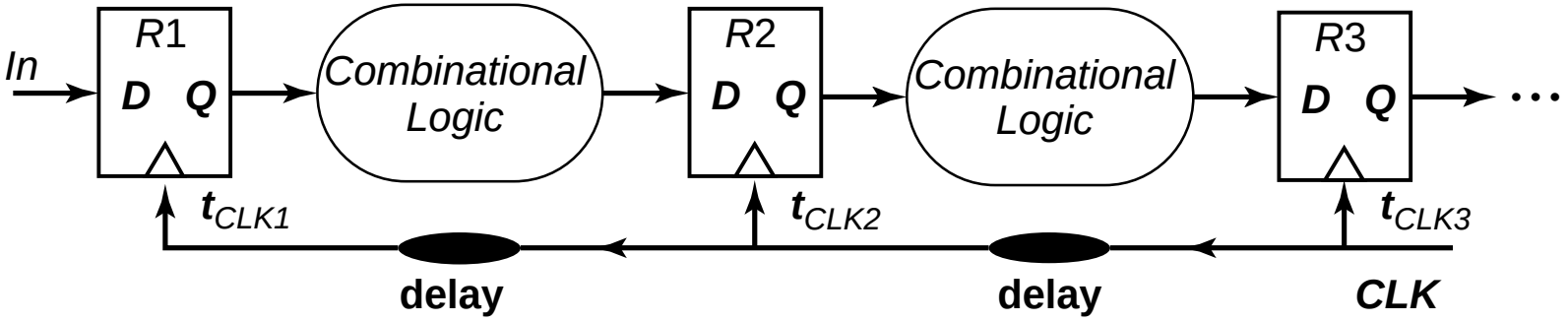
-ve skew adversely affects the performance….

Minimum delay constraint: $\delta + t_{hold} < t_{c\text{-}q,cd} + t_{logic,cd}$*: eliminates RACE*

Positive and Negative Skew



(a) Positive skew CLK and data in the same direction

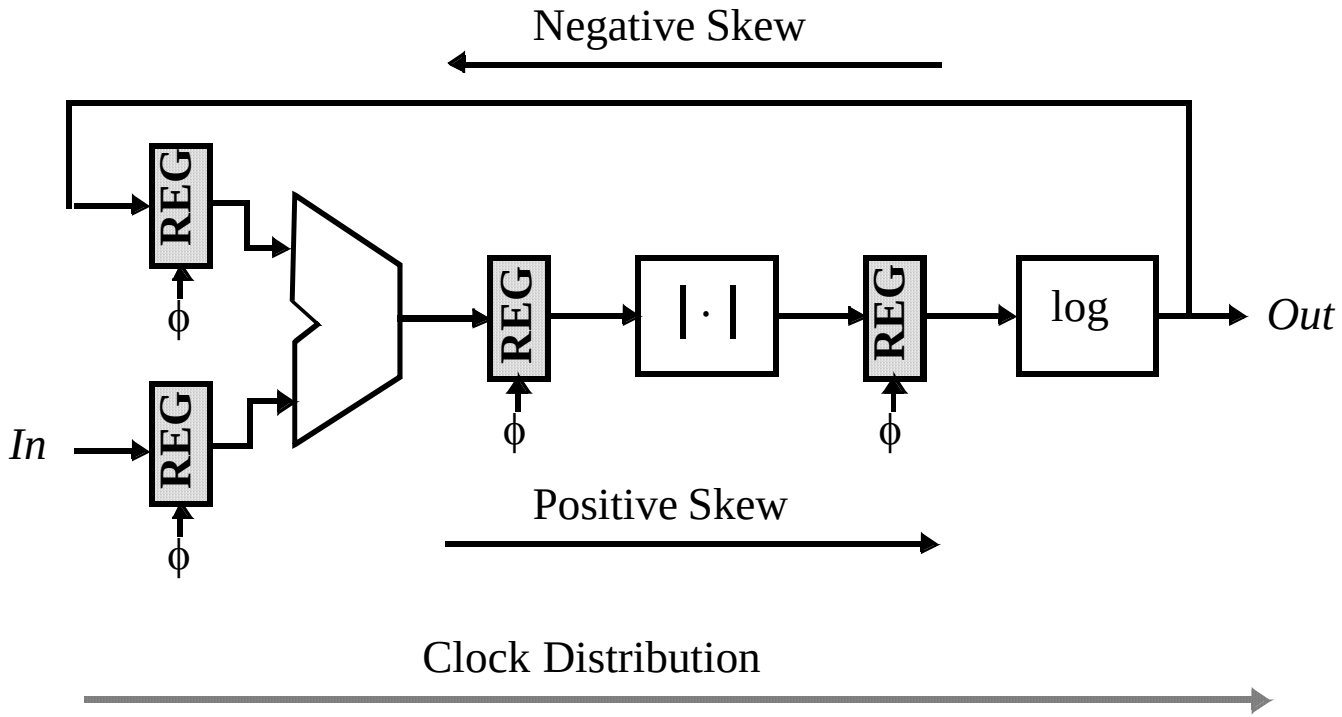


If hold time = 0 or –ve, RACE eliminated

(b) Negative skew CLK and data in the opposite direction

Skew reduces time available for computation, must increase CLK period by d. Hence performance degrades….

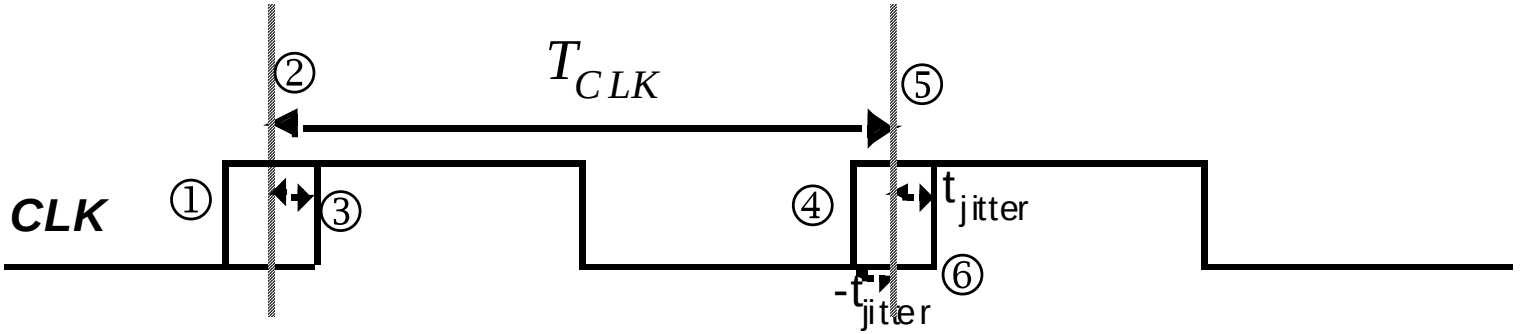
How to counter Clock Skew?



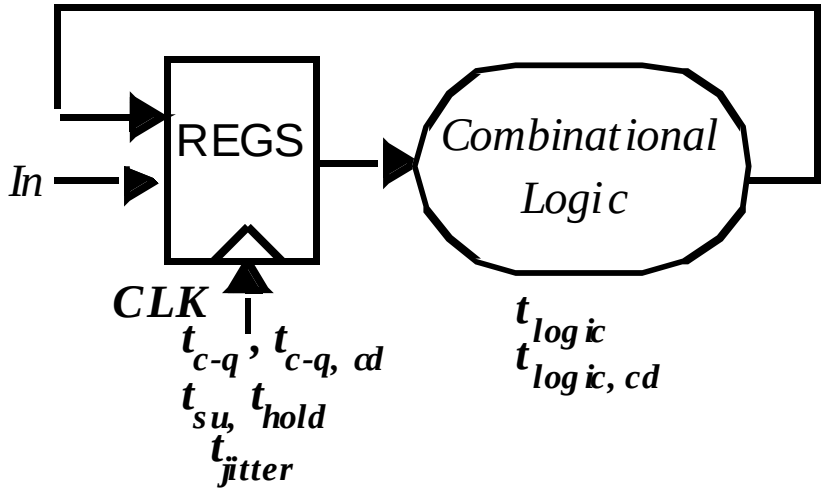
Data and Clock Routing

Must account for the worst case skew…

Impact of Jitter



Jitter is a zero-mean random variable and directly impacts performance



Worst Case:

$T_{jitter} = 2t_{jitter}$

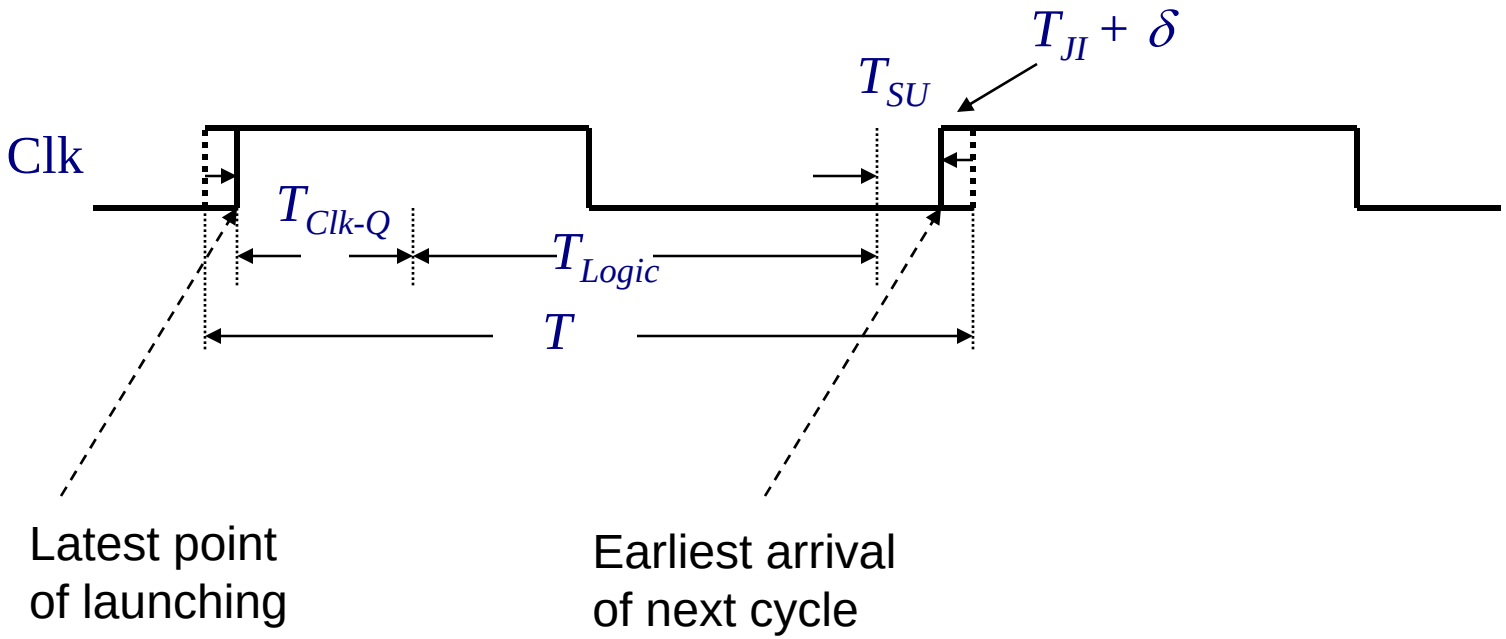
Absolute jitter = t_{jitter} = *worst case variation of a CLK edge at a given location w.r.t. an ideally periodic reference CLK edge*

Cycle-to-Cycle Jitter = T_{jitter} = *time varying deviations of a single clock period relative to an ideal reference CLK:*

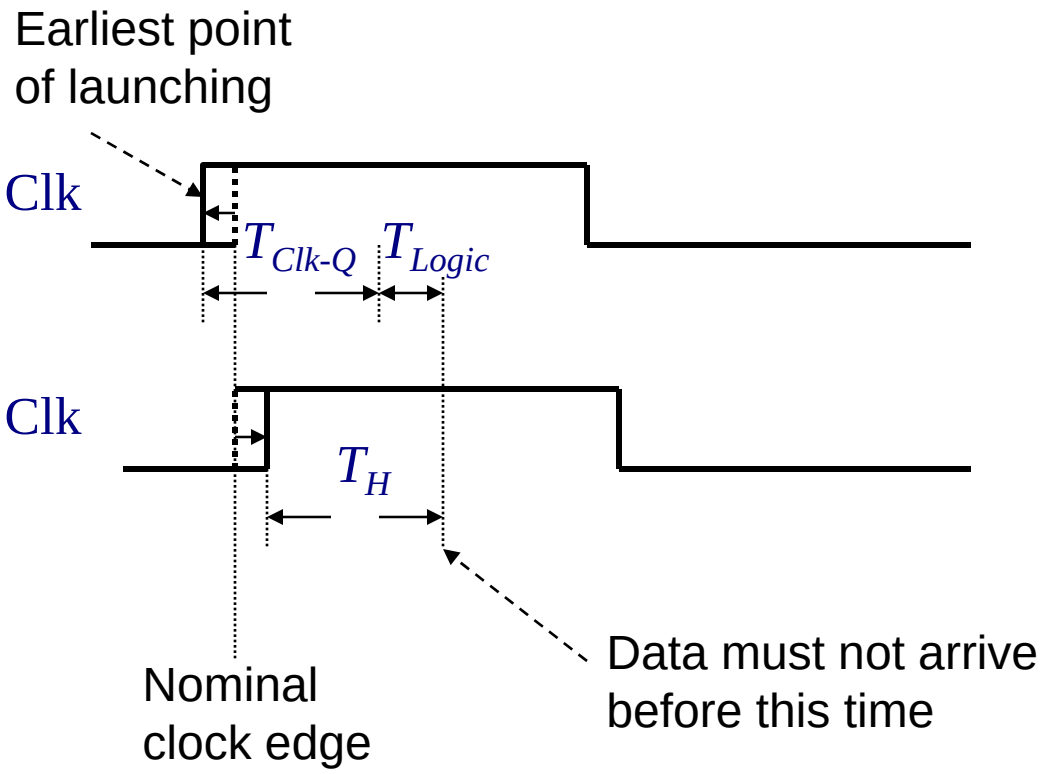
$$T^{i}_{jitter}(n) = t^{i}_{clk,n+1} - t^{i}_{clk,n} - T_{clk}$$

Arrival time of the nth CLK edge at i

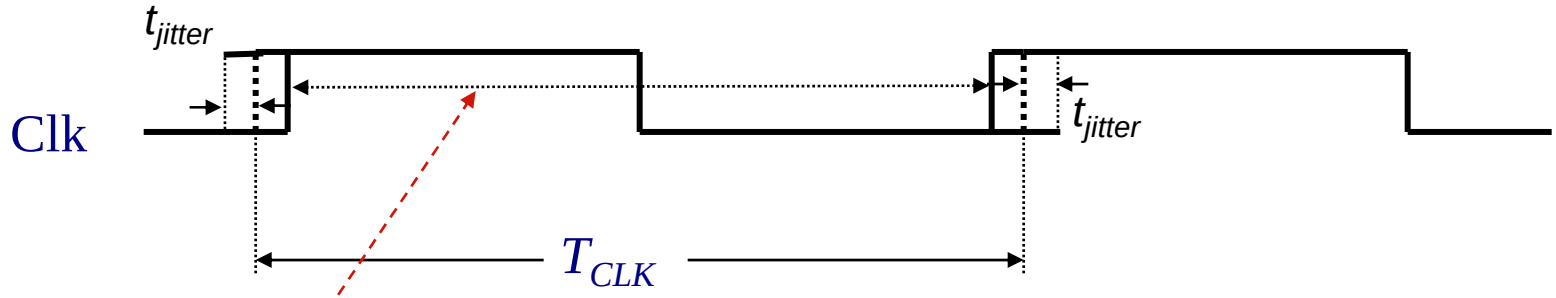
Longest Logic Path in Edge-Triggered Systems



Shortest Path



Impact of Jitter on Clock Constraints in Edge-Triggered Systems



Jitter directly impacts the performance of sequential systems:

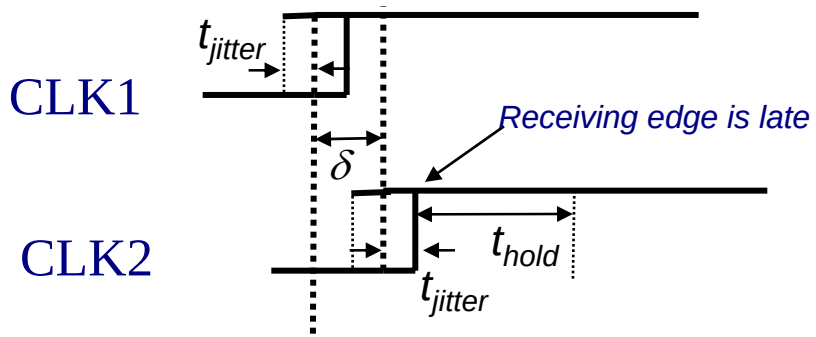
$$T_{CLK} - 2t_{jitter} > t_{c\text{-}q} + t_{logic} + t_{su}$$

Combined effects of Skew and Jitter:

$$T_{CLK} + \delta - 2\,t_{jitter} > t_{c\text{-}q} + t_{logic} + t_{su}$$

Skew can be either positive or negative. +ve skew improves performance, jitter always degrades performance

Impact of Jitter and Skew on Minimum Delay Constraints in Edge-Triggered Systems



If launching edge (CLK1) is early and receiving edge (CLK2) is late:

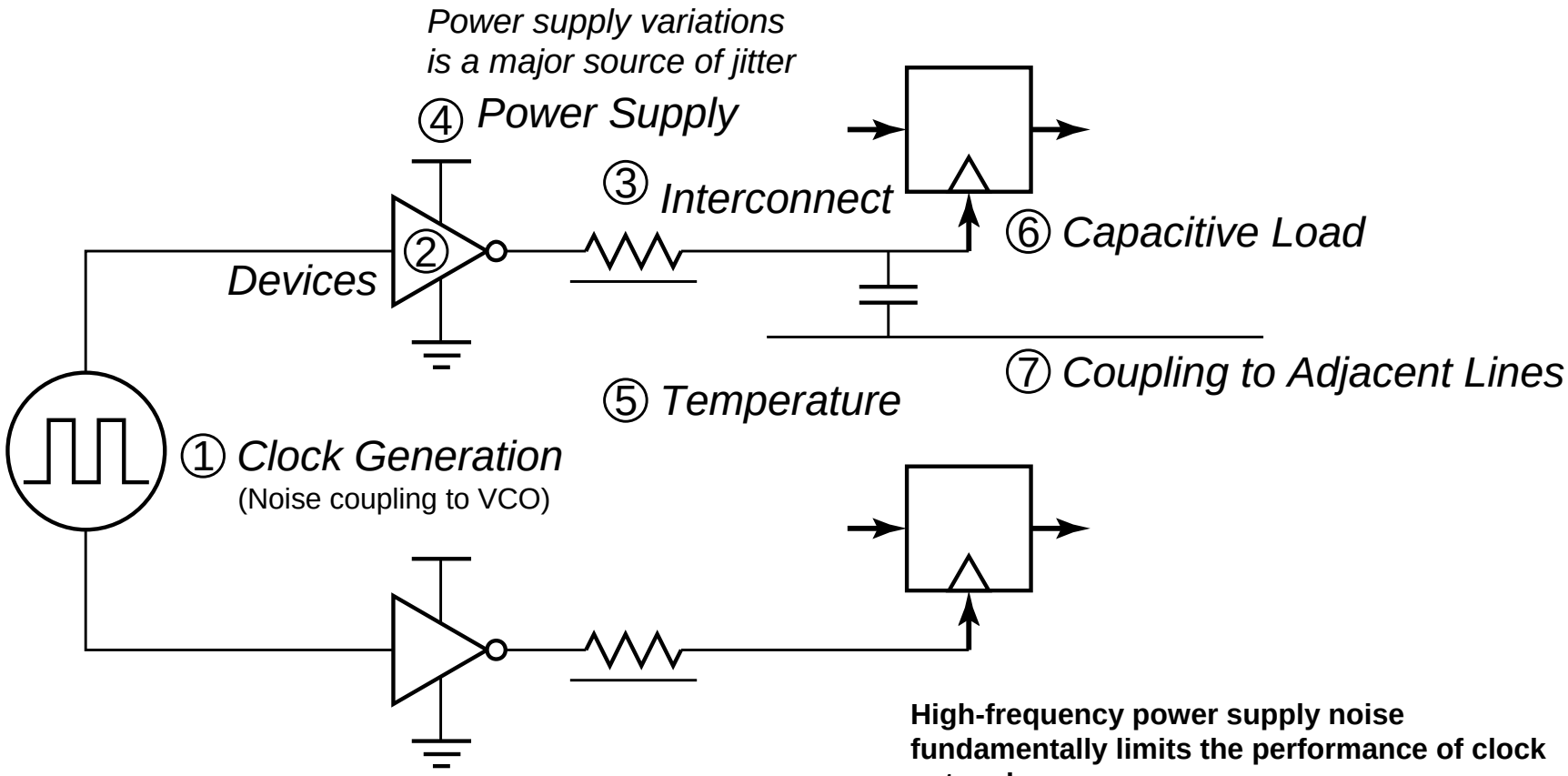
$$t_{hold} + \delta + 2t_{jitter} < t_{c\text{-}q,\,cd} + t_{logic,\,cd}$$

Minimum logic delay constraint:

$$\delta < t_{c\text{-}q,cd} + t_{logic,cd} - t_{hold} - 2t_{jitter}$$

Acceptable skew is reduced by the jitter of the two signals

Clock Uncertainties



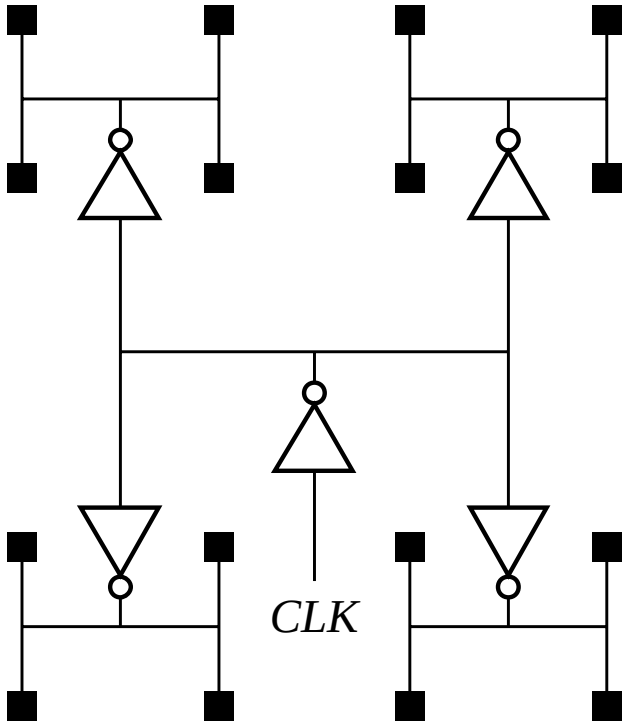
Sources of clock uncertainty:

- *Systematic Vs Random*
- *Static Vs Time-Dependent*

Clocking Issues

Clock Distribution

H-tree

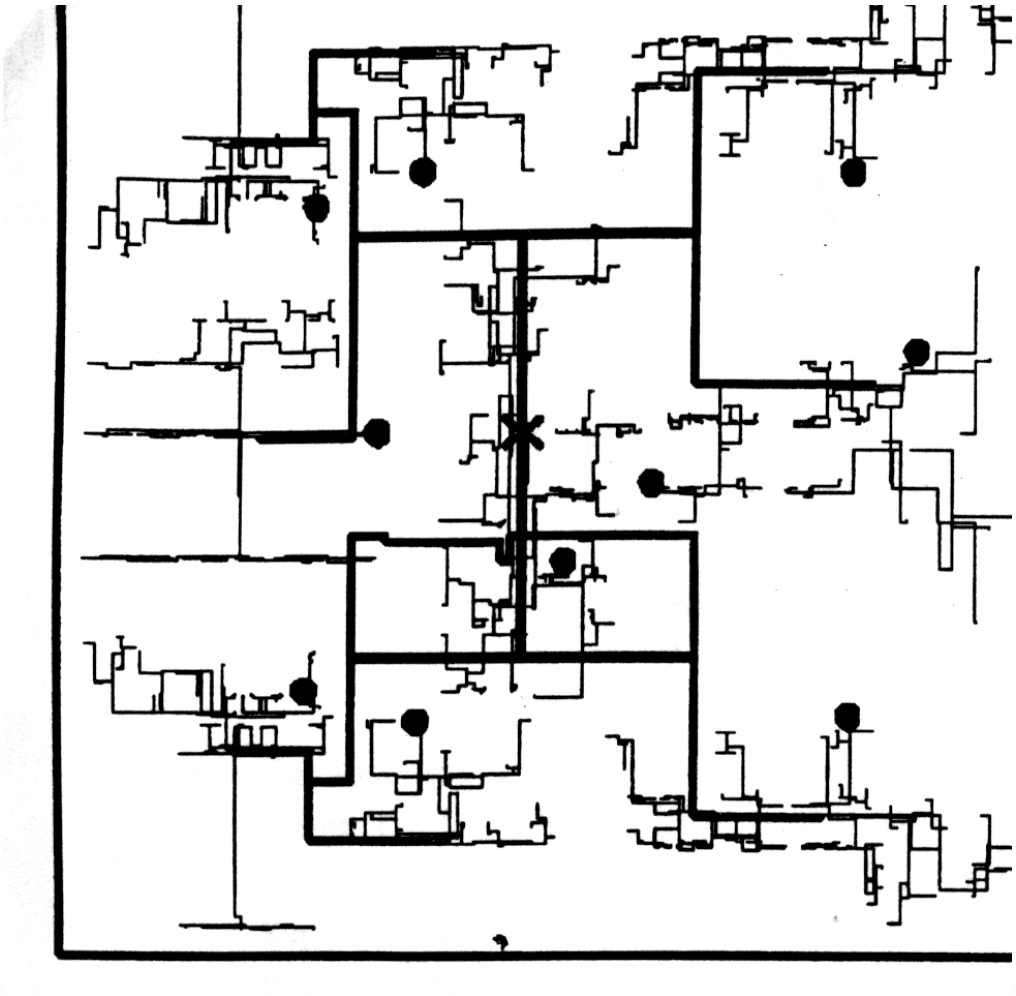


Each path must be balanced: equal interconnect and device load

Difficult to achieve due to parameter variations

Clock is distributed in a tree-like fashion, useful for regular array networks

More Realistic H-tree



Matched RC Trees

Doesn't not require a regular physical structure

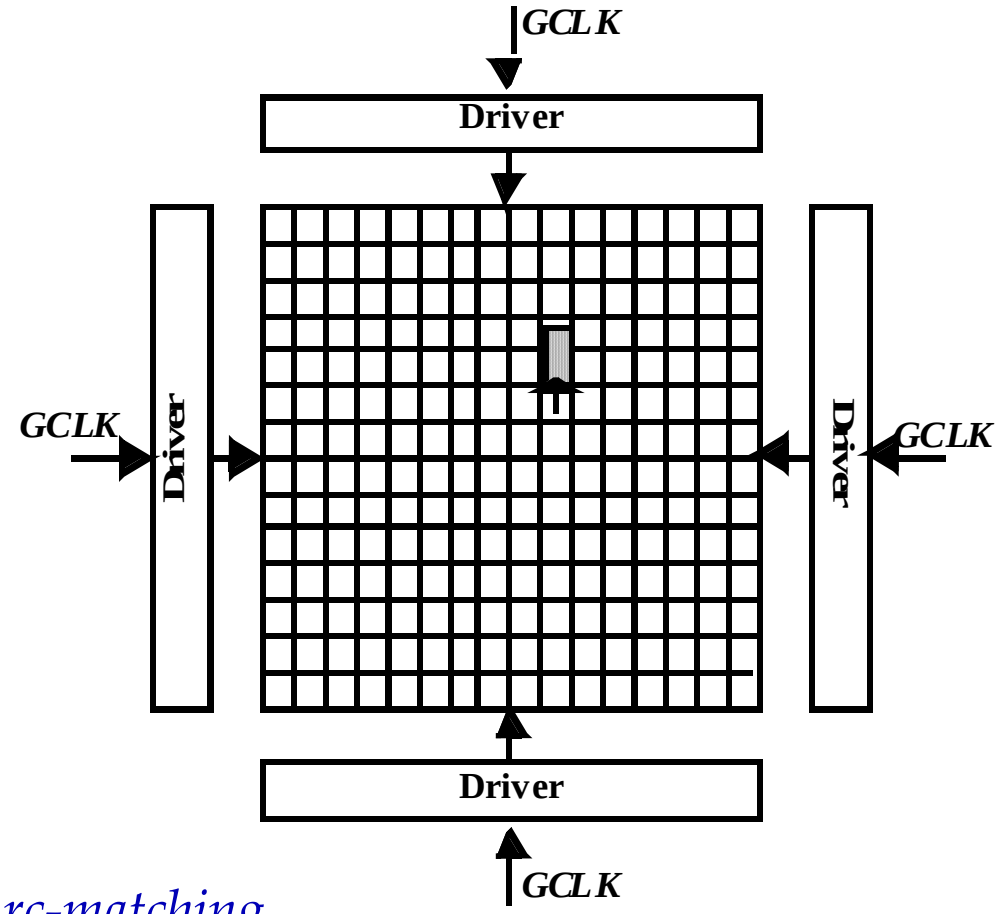
interconnections carrying CLK signals to different sub-blocks are of equal length

Chip is partitioned into balanced load segments (tiles)

Global CLK driver distributes the CLK to the tile drivers located at the dots in the figure

[Restle98]

The Grid System



- *No rc-matching*
- *Large power*

CLK is more easily accessible

Delay from the final driver to each load is not matched

Absolute delay is minimized

Allows for late design changes

Large power dissipation: due to excess interconnects

Example: DEC Alpha 21164

Clock Frequency: 300 MHz - 9.3 Million Transistors

0.5 um CMOS process

Total Clock Load: 3.75 nF

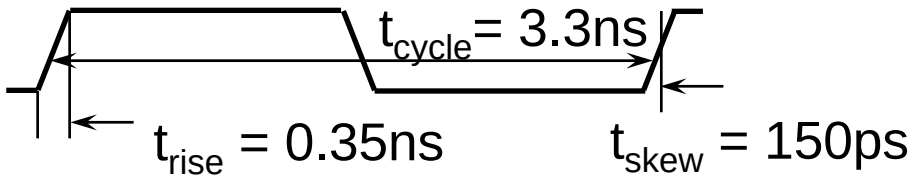
Power in Clock Distribution network : 20 W (out of 50)

Uses Two Level Clock Distribution:

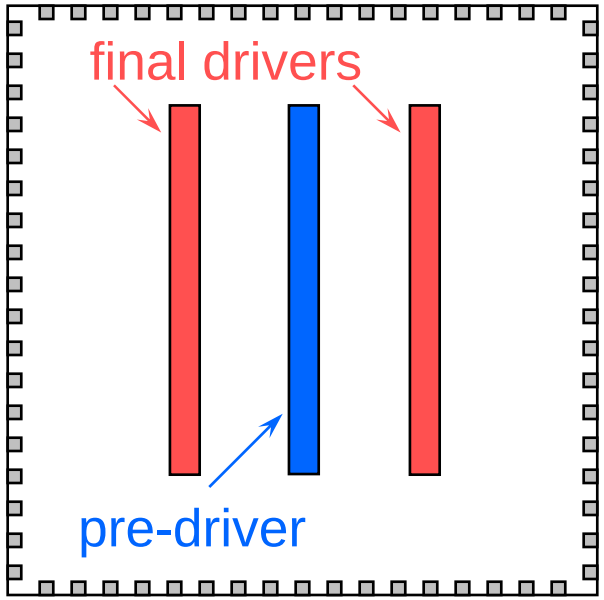
- **Single 6-stage driver at center of chip**
- **Secondary buffers drive left and right side clock grid in Metal3 and Metal4**

Total driver size: 58 cm!

21164 Clocking

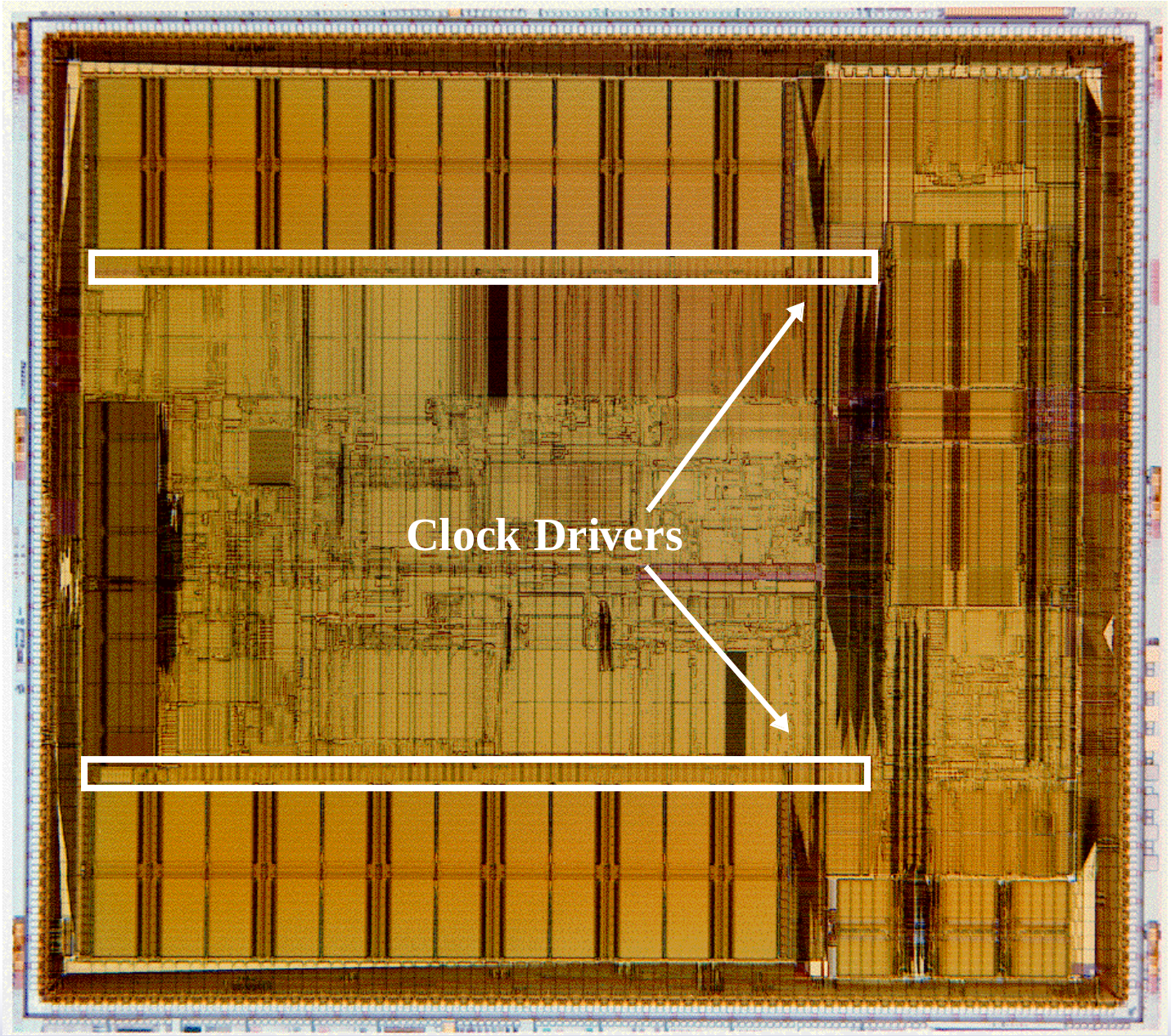


Clock waveform

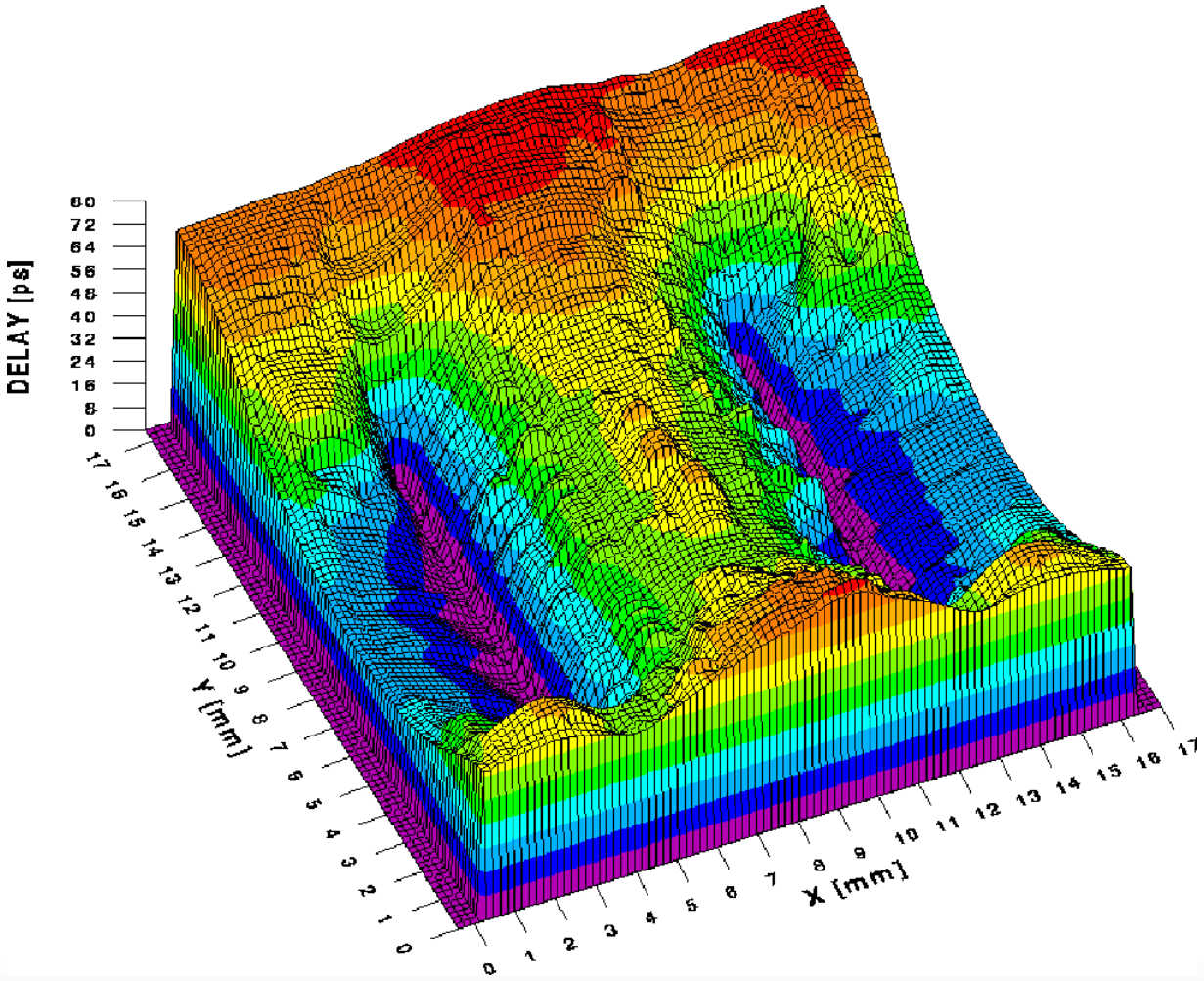


Location of clock driver on die

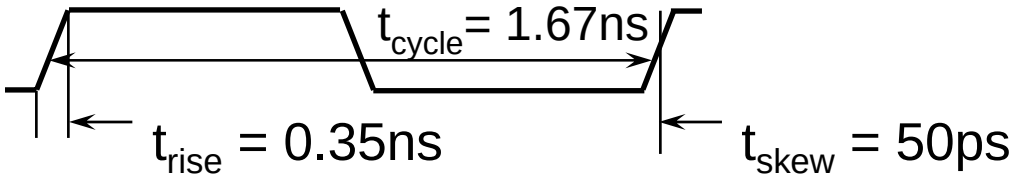
- 2 phase single wire clock, distributed globally
- 2 distributed driver channels
  - Reduced RC delay/skew
  - Improved thermal distribution
  - 3.75nF clock load
  - 58 cm final driver width
- Local inverters for latching
- Conditional clocks in caches to reduce power
- More complex race checking
- Device variation


Clock Drivers

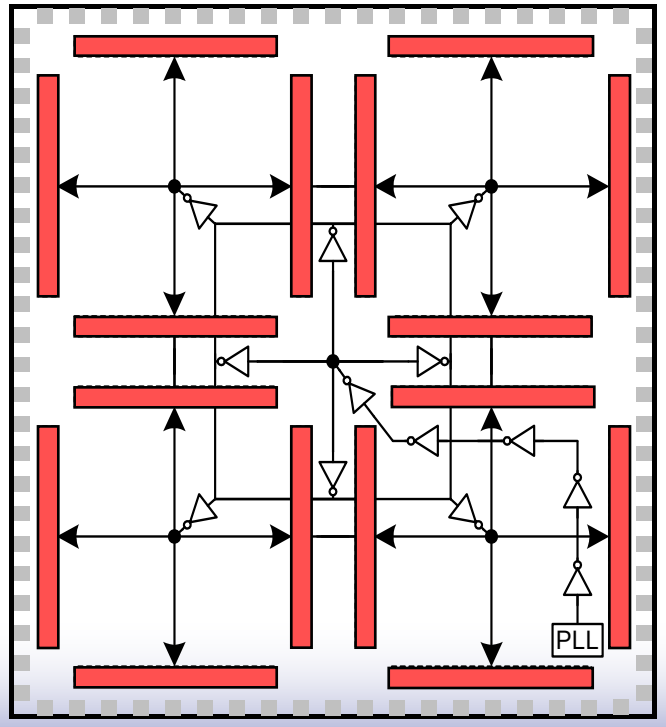
Clock Skew in Alpha Processor



EV6 (Alpha 21264) Clocking
600 MHz – 0.35 micron CMOS

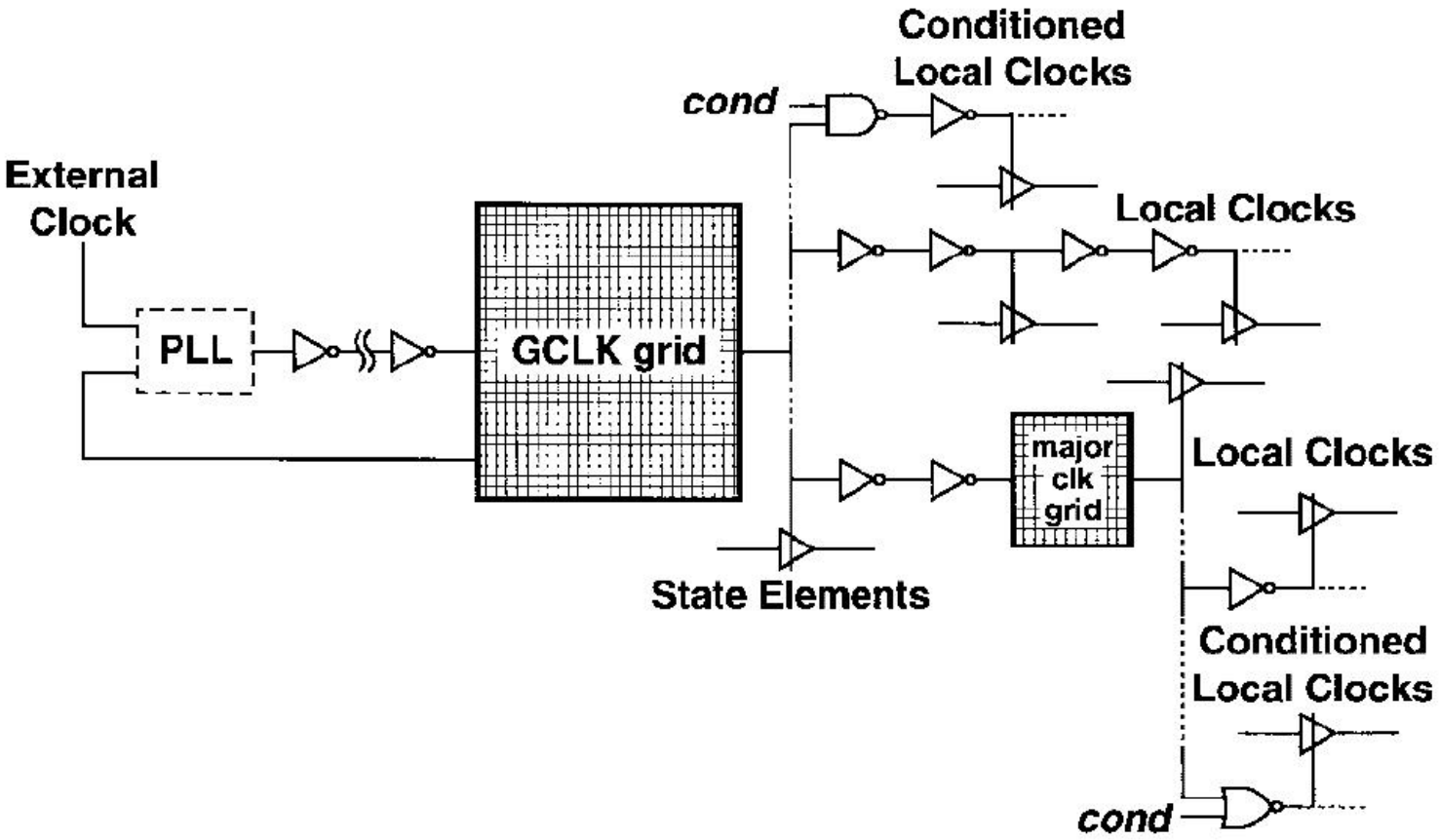


Global clock waveform

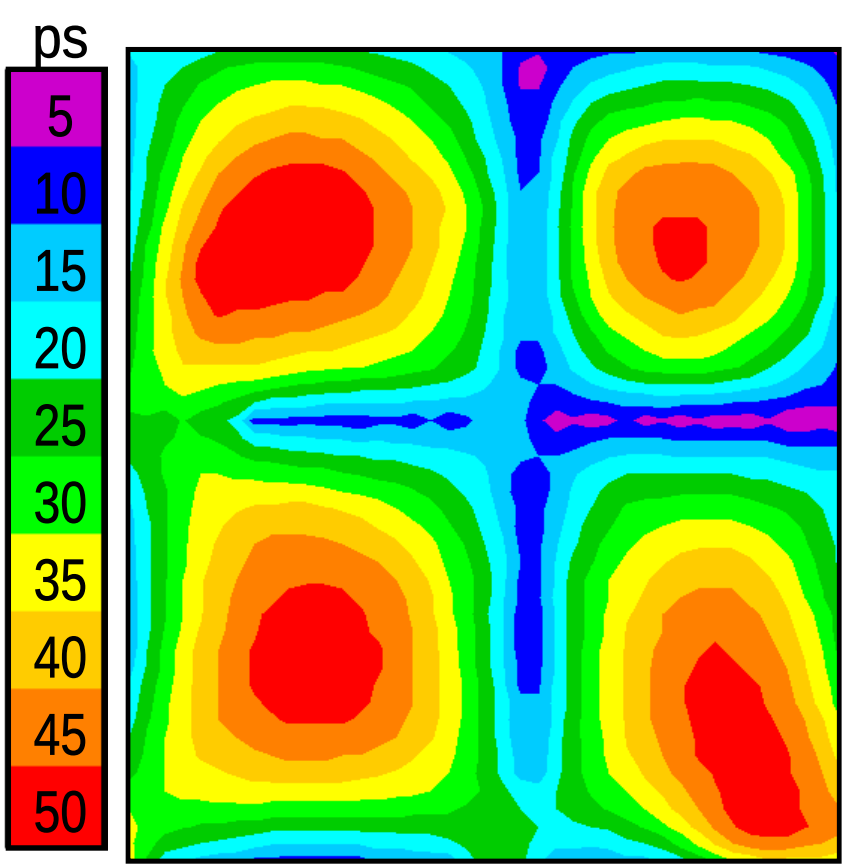


- 2 Phase, with multiple conditional buffered clocks
  - 2.8 nF clock load
  - 40 cm final driver width
- Local clocks can be gated “off” to save power
- Reduced load/skew
- Reduced thermal issues
- Multiple clocks complicate race checking

21264 Clocking

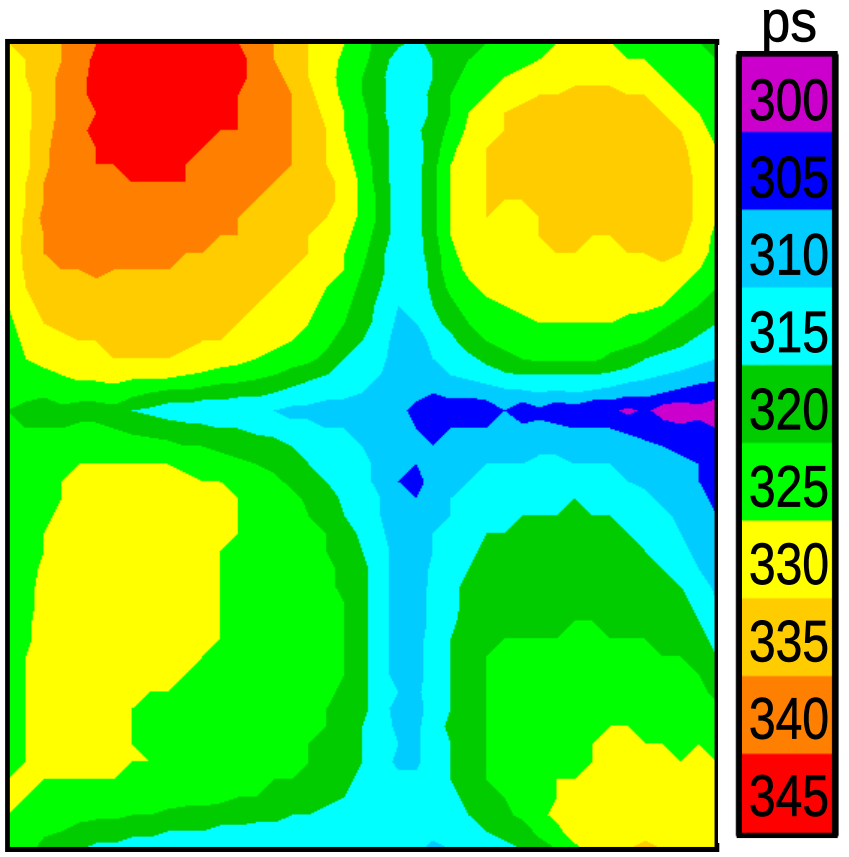


EV6 Clock Results



GCLK Skew

(at Vdd/2 Crossings)

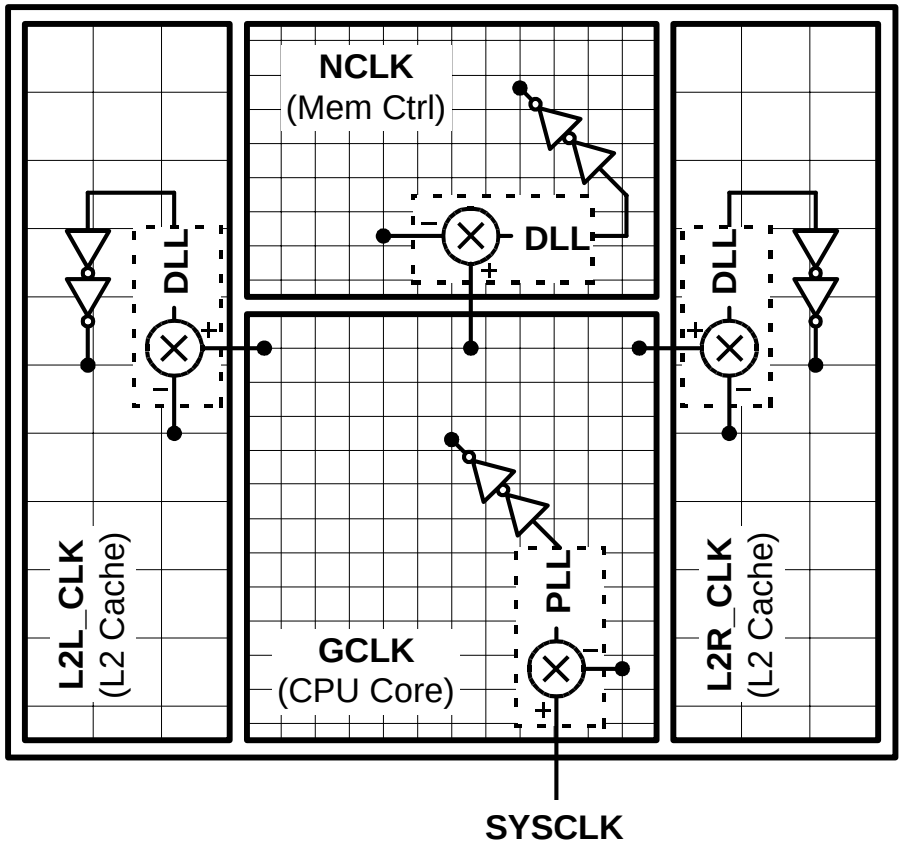


GCLK Rise Times

(20% to 80% Extrapolated to 0% to 100%)

EV7 Clock Hierarchy

Active Skew Management and Multiple Clock Domains



+ widely dispersed drivers

+ DLLs compensate static and low-frequency variation

+ divides design and verification effort

- DLL design and verification is added work

+ tailored clocks