Lecture 09: Static Timing Analysis

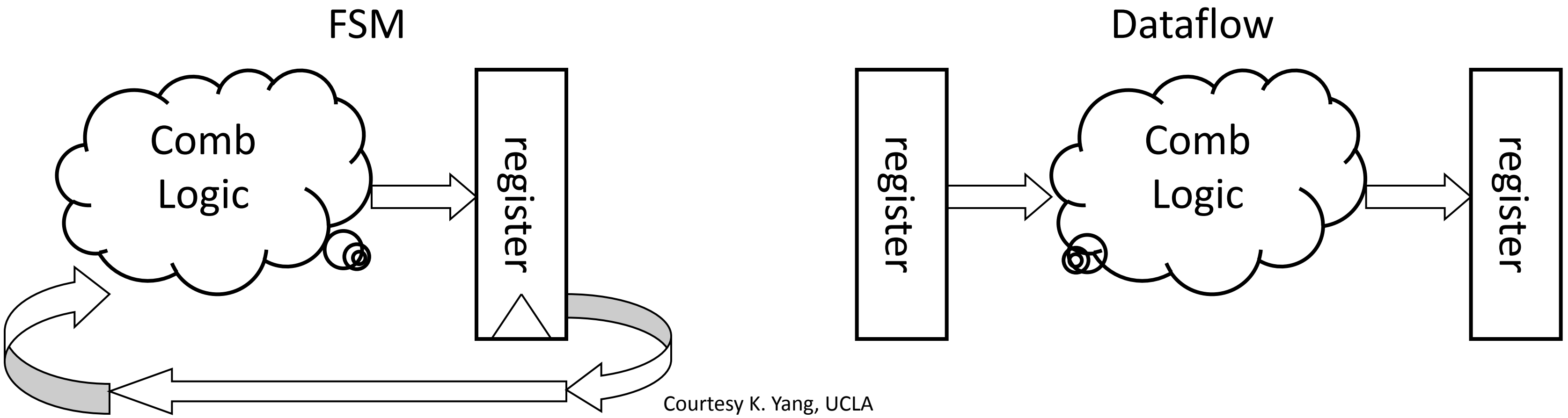
Matthew Guthaus
Professor
UCSC, Computer Science and Engineering
http://vlsida.soe.ucsc.edu
mrg@ucsc.edu

Today's Topics

- Timing Constraints
  - Long paths
  - Short paths
- Static Timing
- Timing Corners

Why Clocks?

- Clocks provide the means to synchronize
  - By allowing events to happen at known timing boundaries, we can sequence these events
- Greatly simplifies building of state machines
- No need to worry about variable delay through combinational logic (CL)
  - All signals delayed until clock edge (clock imposes the worst case delay)



Courtesy K. Yang, UCLA

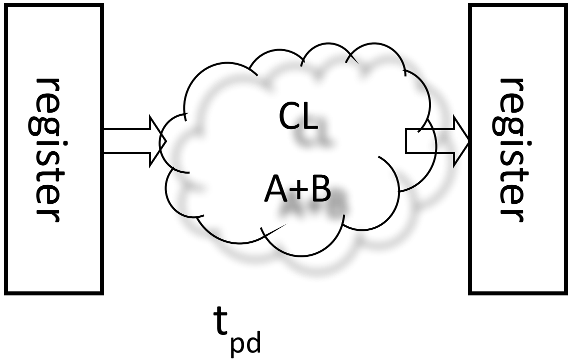
Clock Cycle Time

- Cycle time is determined by the delay through the CL
  - Signal must arrive before the latching edge
  - If too late, it waits until the next cycle
    - Synchronization and sequential order becomes incorrect
- Constraint: Tcycle > Tprop_delay_through_CL + Toverhead
  - Example: 3.0 GHz Pentium-4 □ Tcycle = 333ps
- Can change circuit architecture to obtain smaller Tcycle

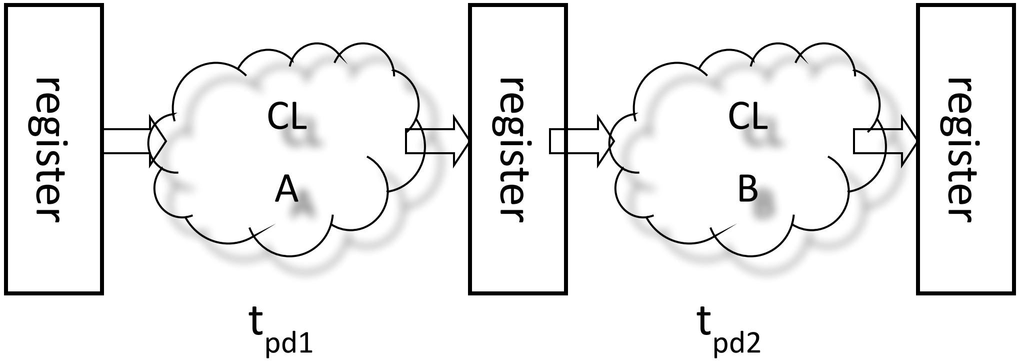
Pipelining

- For dataflow:
  - Instead of a long critical path, split the critical path into chunks
  - Insert registers to store intermediate results
  - This allows 2 waves of data to coexist within the CL
- Can we extend this ad infinitum?
  - Overhead eventually limits the pipelining
    - E.g., 1.5 to 2 gate delays for latch or FF
  - Granularity limits as well
    - Minimum time quantum: delay of a gate

- $T_{cycle} > T_{pd} + T_{overhead}$



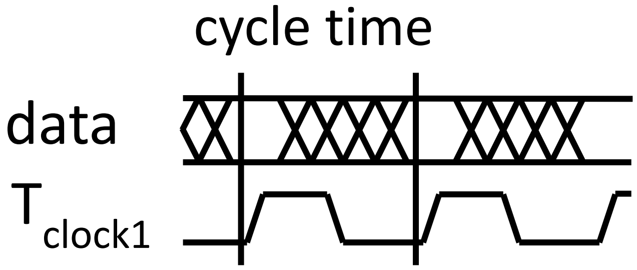
- $T_{cycle} > \max(t_{pd1,}\ t_{pd2}) + T_{overhead}$



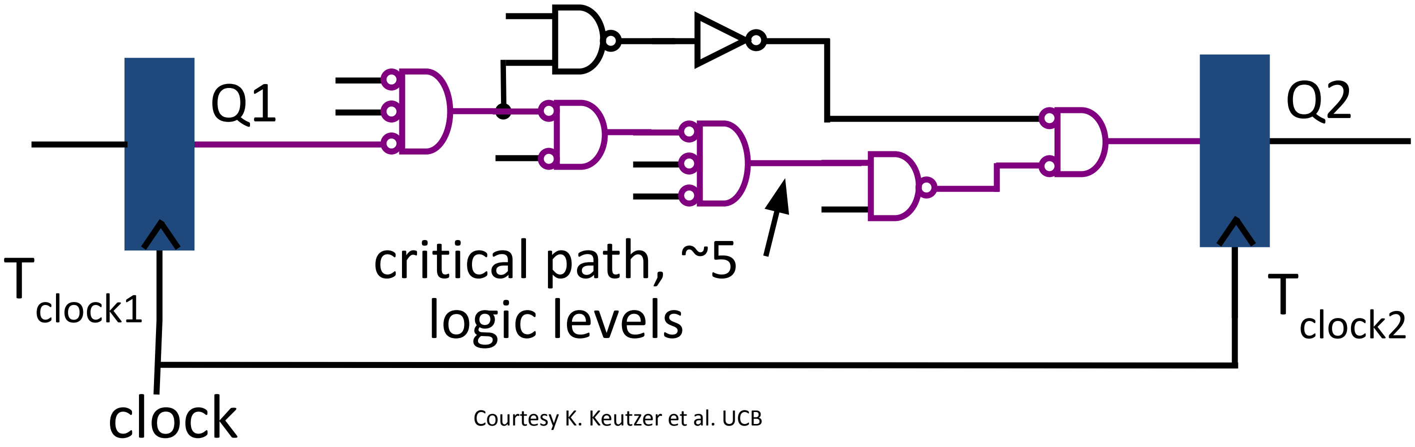
Courtesy K. Yang, UCLA

Let’s Revisit Cycle Time and Path Delay

- Cycle time (T) cannot be smaller than longest path delay (T_{max})
- Longest (critical) path delay is a function of:
  - Total gate
  - Wire delays
  - (Logic levels)



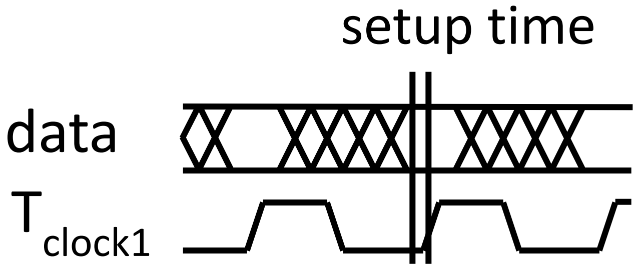
$T_{max} \leq T$



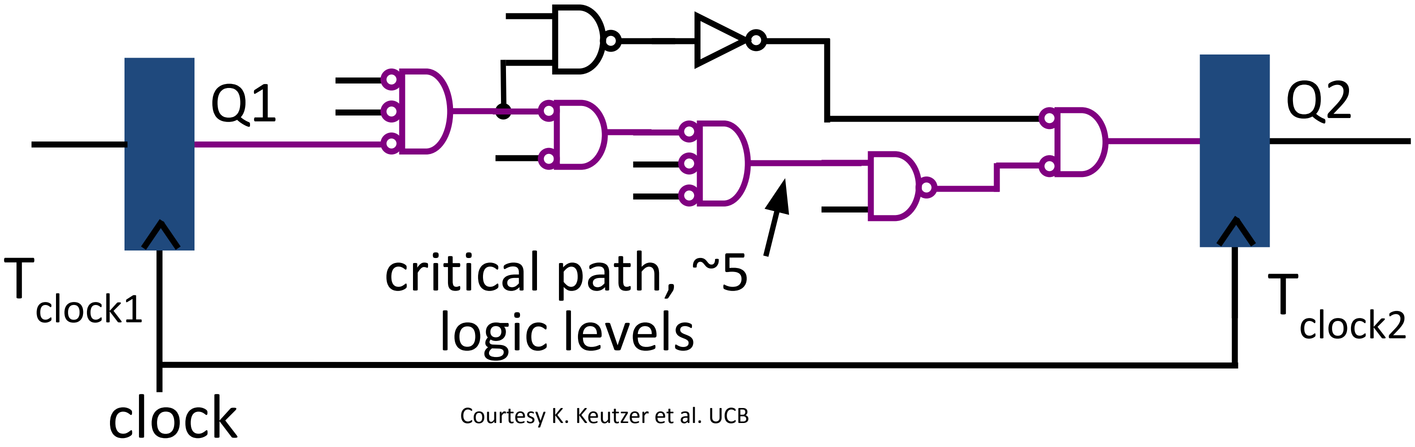
Courtesy K. Keutzer et al. UCB

Cycle Time - Setup Time

- For FFs to correctly capture data, must be stable for:
  - Setup time (T_{setup}) *before* clock arrives



$$T_{max} + T_{setup} \leq T$$

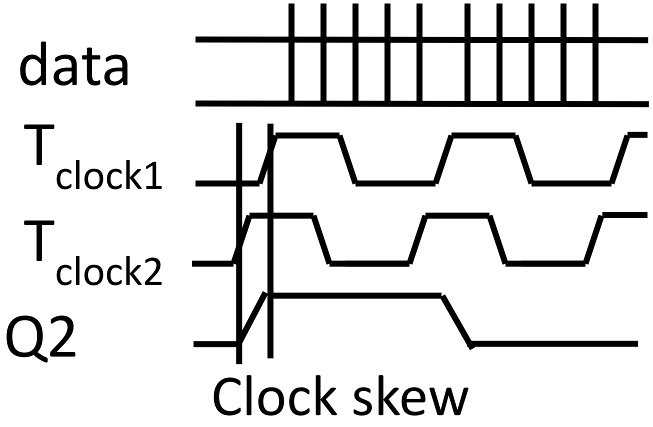


Courtesy K. Keutzer et al. UCB

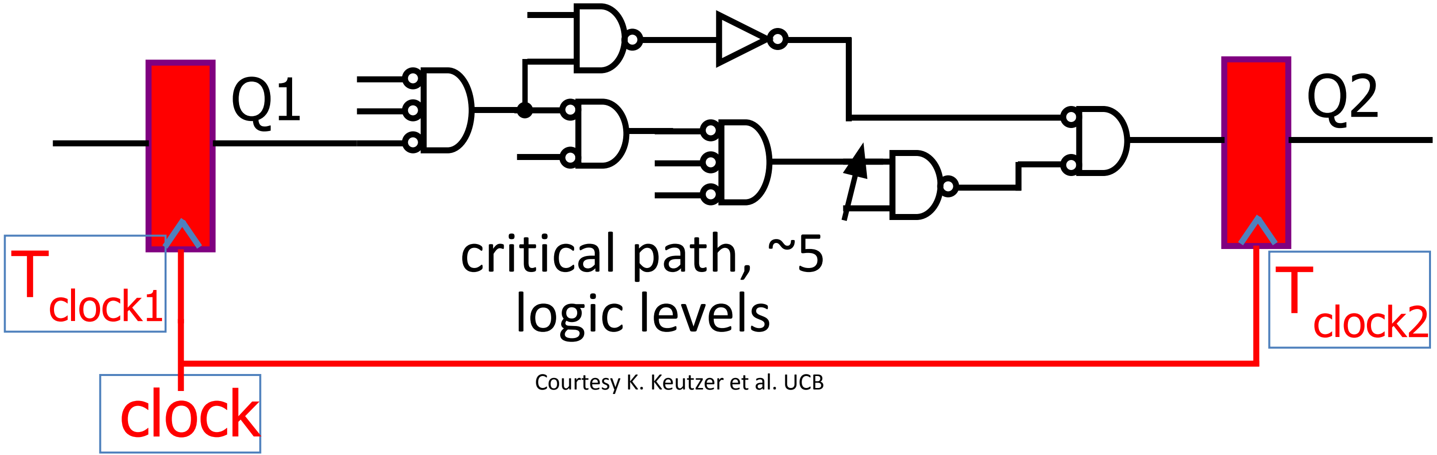
Cycle Time – Clock Skew

- If clock network has unbalanced delay – clock skew

- Cycle time is also a function of clock skew (T_{skew})



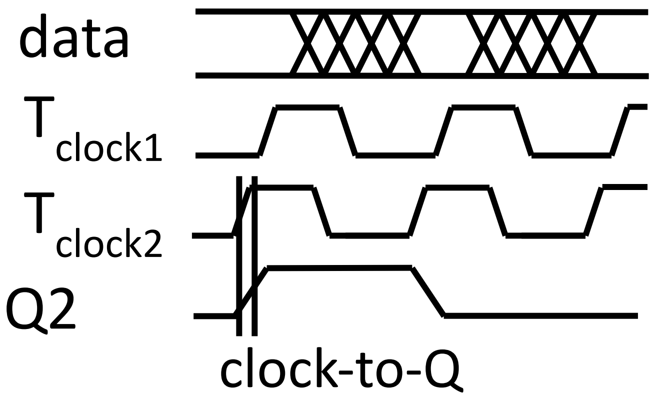
$$T_{max} + T_{setup} + T_{skew} \leq T$$



Courtesy K. Keutzer et al. UCB

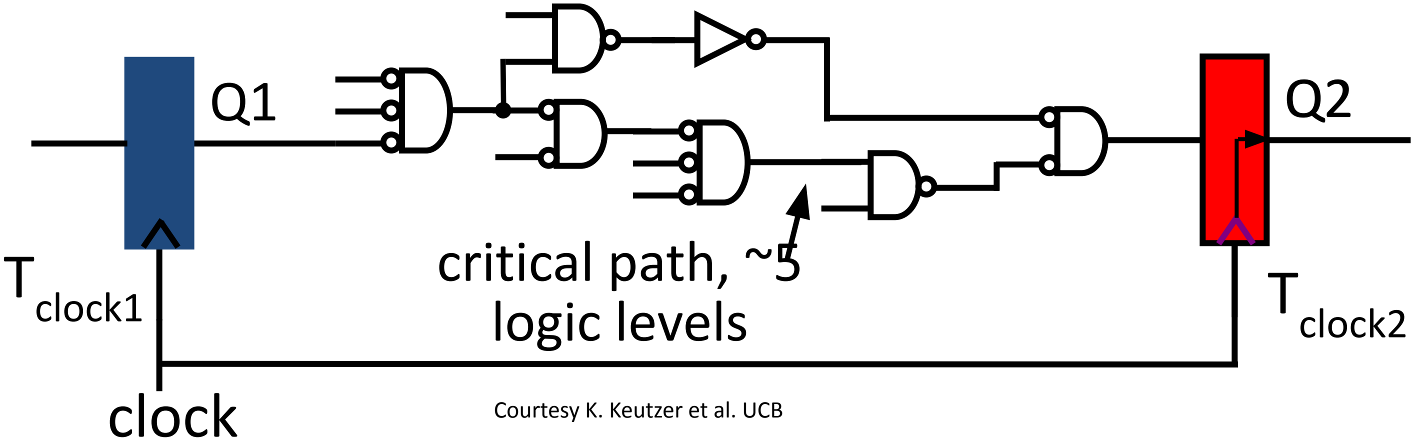
Cycle Time – Flip-Flop Delay (Clock to Q)

- Cycle time is also a function of propagation delay of FF ($T_{clk\text{-}to\text{-}Q}$ or T_{c2q})



- T_{c2q} : time from arrival of clock signal till change at FF output

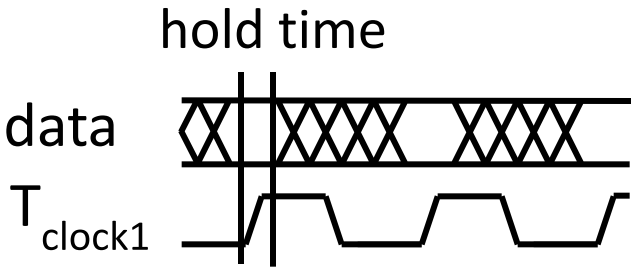
$$T_{max} + T_{setup} + T_{skew} + T_{clk-to-Q} \le T$$



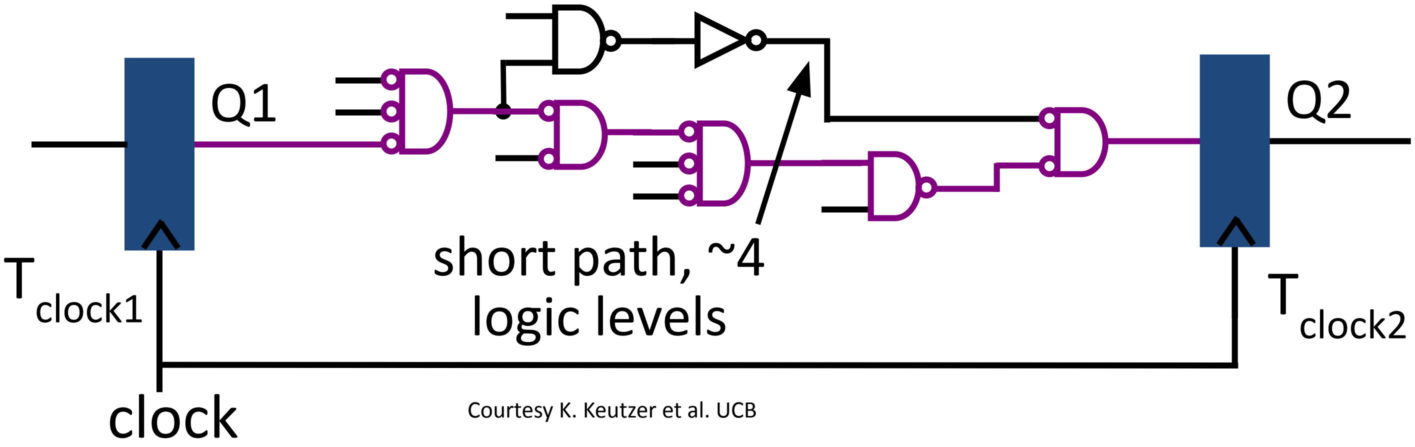
Courtesy K. Keutzer et al. UCB

Min Path Delay - Hold Time

- For FFs to correctly latch data, data must be stable during:
  - Hold time (T_{hold}) *after* clock arrives
- Determined by delay of shortest path in circuit (T_{min}) and clock skew (T_{skew})

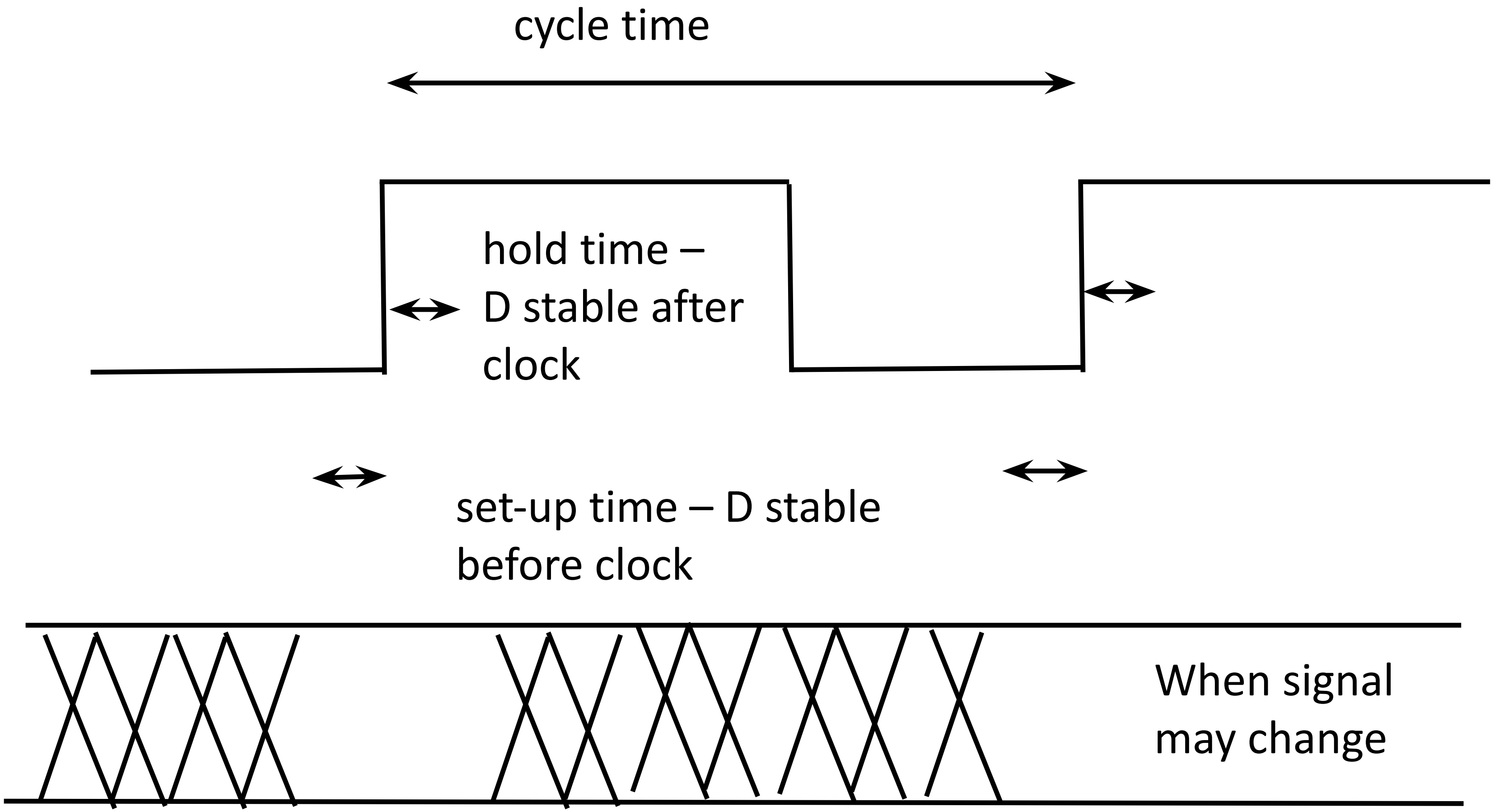


$$T_{min} \geq T_{hold} + T_{skew}$$



Courtesy K. Keutzer et al. UCB

Setup, Hold, Cycle Times



cycle time

hold time – D stable after clock

set-up time – D stable before clock

When signal may change

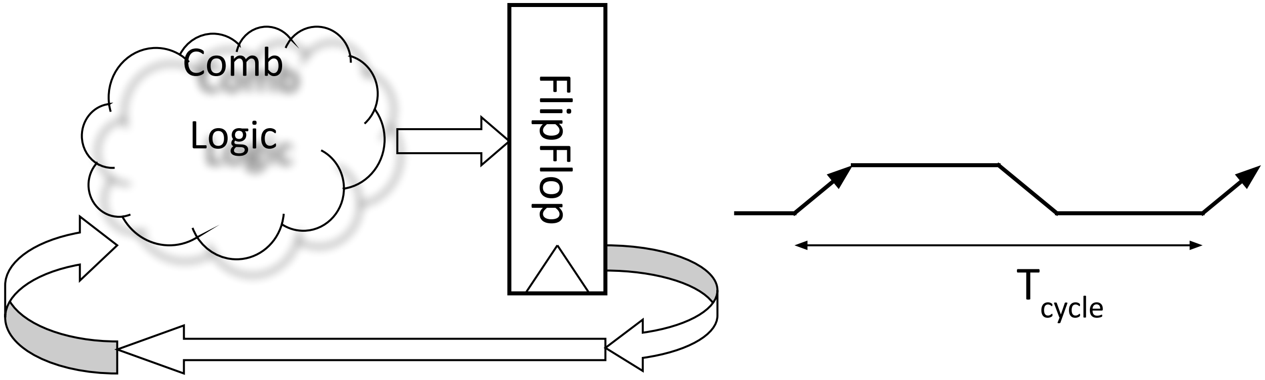
Example of a single phase clock

Courtesy K. Keutzer et al. UCB

Positive or Negative Clock Skew?

- Usually defined by receiving minus launching clock
- However, often a single value or “budget” can be assumed of Tskew
  - Used as “worst case” for all pairs of flops
  - Especially if pre-CTS (i.e. during synthesis and placement)
- Technically…
  - Positive skew helps setup paths
  - Negative skew helps hold paths

Timing Constraints for Edge-Triggered FFs

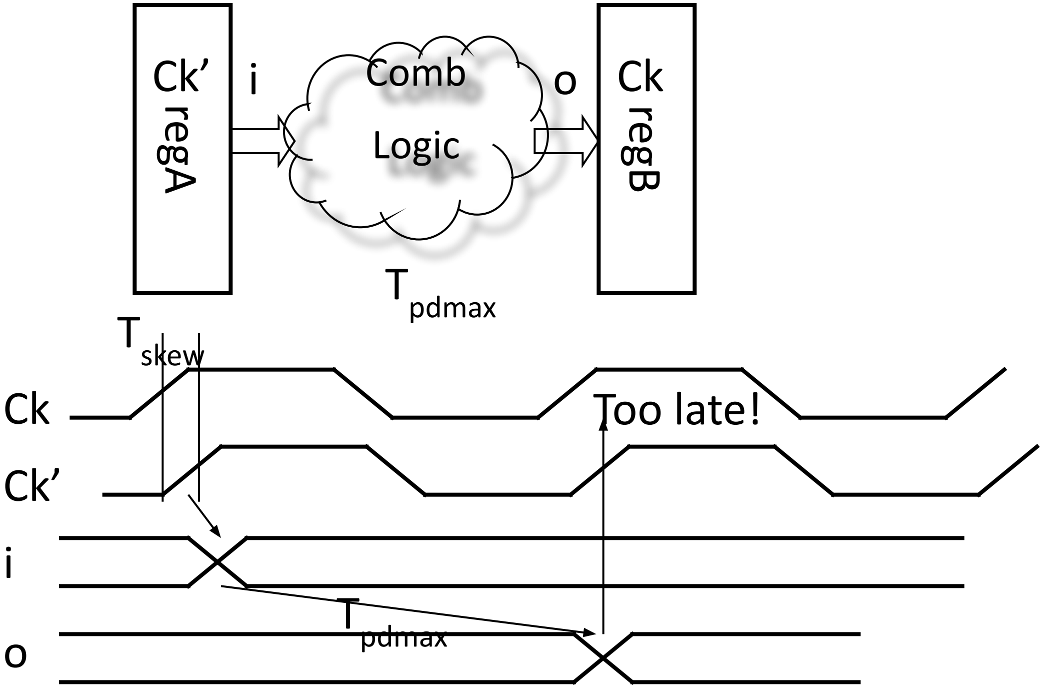


Courtesy K. Yang, UCLA

- $\mathrm{Max}(T_{pd}) < T_{cycle} - T_{setup} - T_{c2q} - T_{skew}$
  - Delay is too long for data to be captured
- $\mathrm{Min}(T_{pd}) > T_{hold} - T_{c2q} + T_{skew}$
  - Delay is too short and data can race through, skipping a state

Example of T_{pdmax} Violation

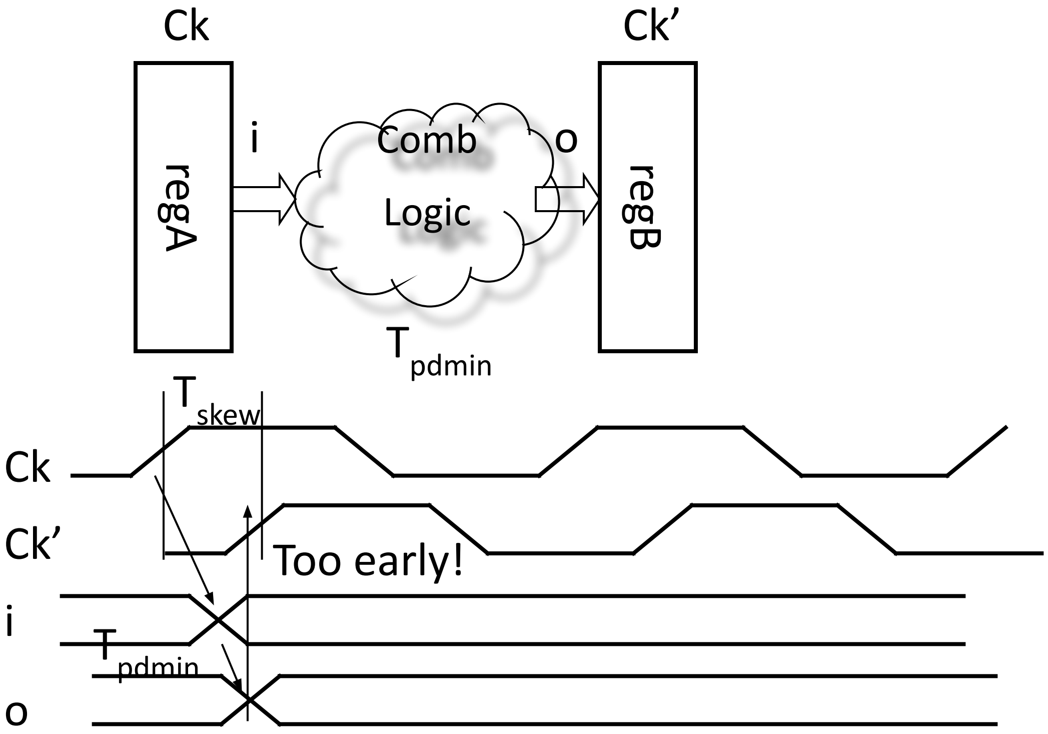
- Suppose there is skew between the registers in a dataflow (regA after regB)
- "i" gets its input values from regA at transition in Ck'
- CL output "o" arrives after Ck transition due to skew
- To correct this problem, can *increase* cycle time



Courtesy K. Yang, UCLA

Example of T_{pdmin} Violation: Race Through

- Suppose clock skew causes regA to be clocked before regB
- “i” passes through the CL with little delay (tpdmin)
- “o” arrives before the rising Ck’ causes the data to be latched
- <u>Cannot be fixed by changing frequency</u> □ have rock instead of chip



Courtesy K. Yang, UCLA

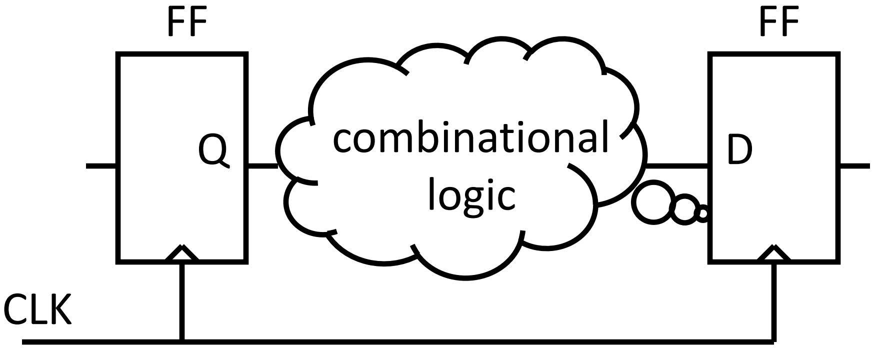
Summary: Timing Constraints

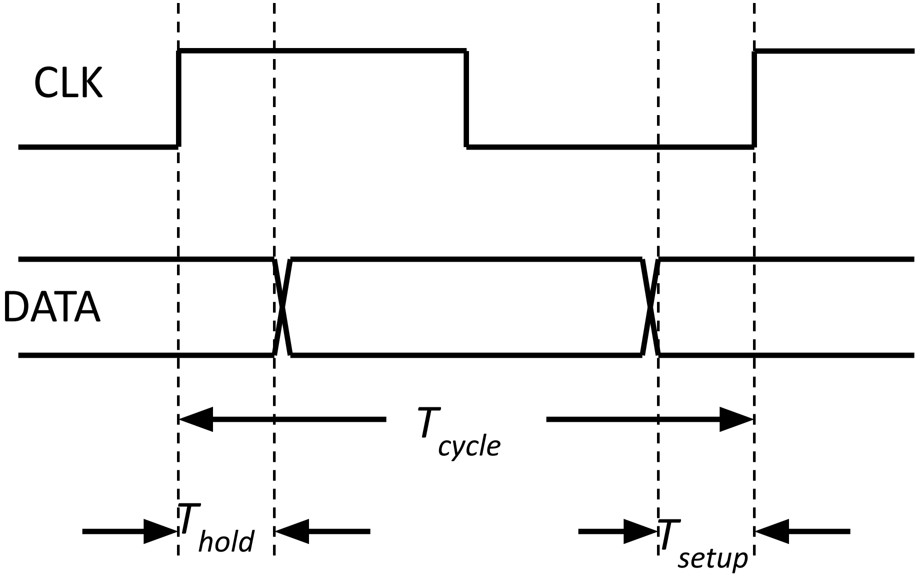
- Synchronous design
  = combinational logic
  + sequential elements

- For each flip-flop:
  - $T_{max} + T_{setup} < T_{cycle} - T_{skew}$
  - $T_{min} > T_{hold} + T_{skew}$

- T_{max}: *longest data propagation path delay*

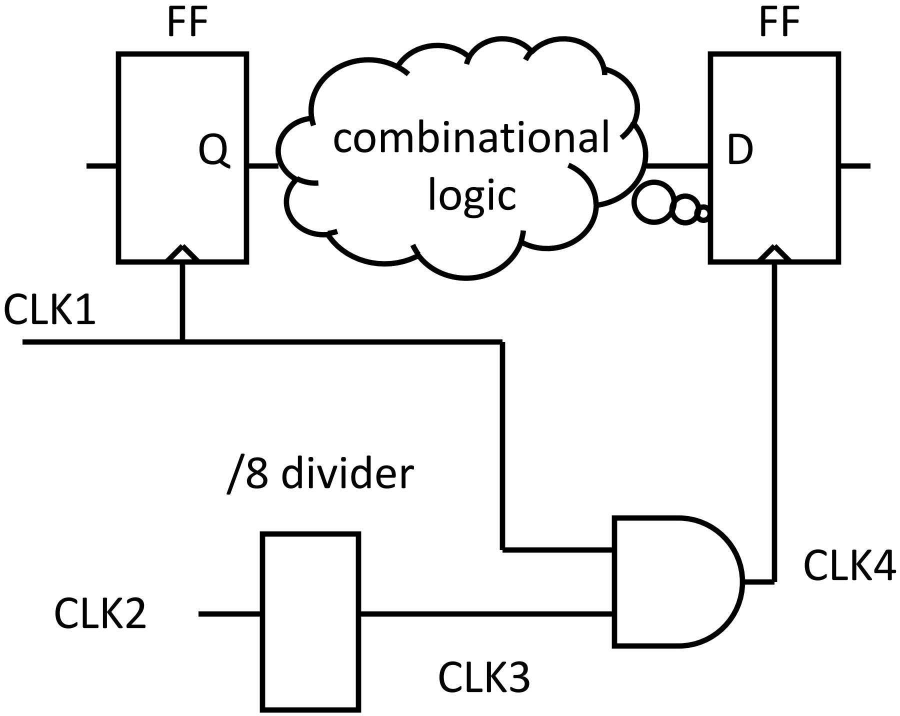
- T_{min}: *shortest data propagation path delay*





Clock Identification

- Partition the design
- Clock network
  - Clock definition
  - Derived clock
  - Clock groups
  - Clock delay (skew) calculation
  - Timing constraints exist between clocks with a common divisor frequency
- Data paths with timing constraints

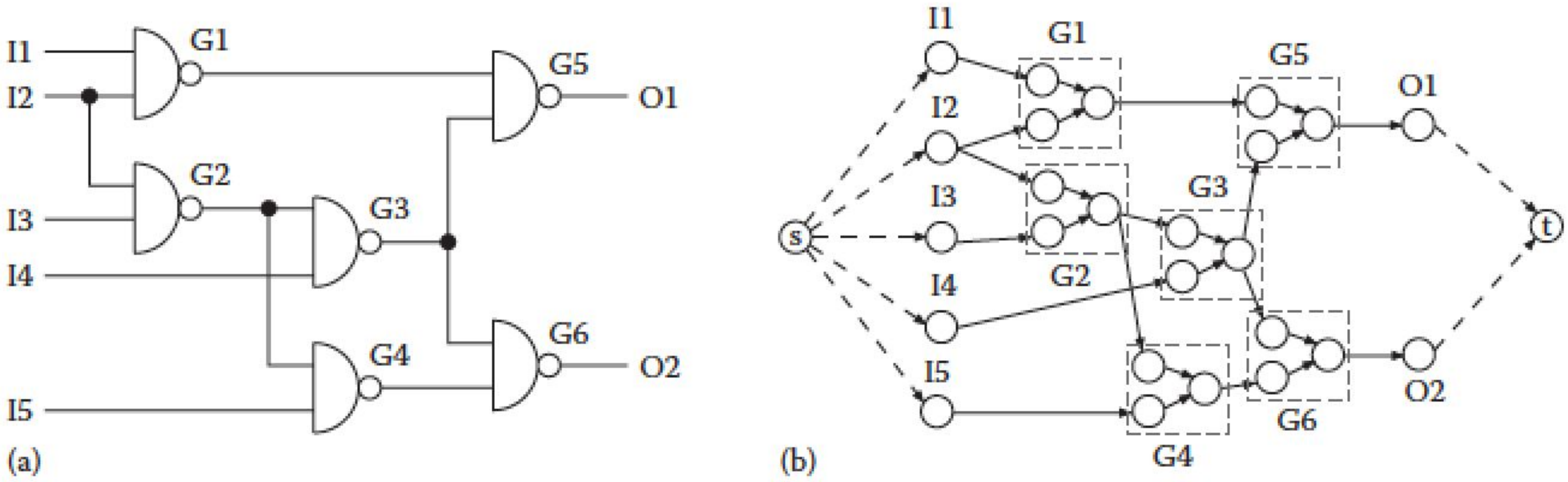


STA Overview

- Questions it can answer
  - Can my circuit run at X Ghz?
  - What is the worst path of my circuit?
  - Do I have any hold violations?
- Static vs Dynamic Timing
  - For n inputs and m FF bits, there are 2^(m+n)-1 possible patterns.
  - This cannot be exhaustively tested!
- Incremental timing for optimization
  - If I have a timing problem, how do I fix it?
- Corner analysis for temperature, process, etc.
  - Will my circuit work under all conditions?

Circuit Representation

- Directed Acyclic Graph (DAG)
  - Circuit is broken at sequential elements (e.g. FFs)
  - s can be circuit input or FF output (input delay)
  - t can be circuit output or FF input (setup time)
- Edges are gate or wire delays
- Nodes are pins on gates or inputs/outputs

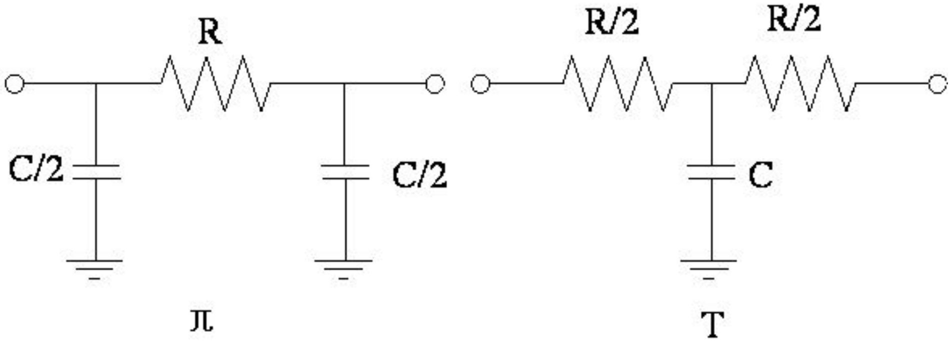


Gate Delay Models

- Discussed in Cell Library lecture
- Use as inputs:
  - output load
  - input signal slew (transition time)

Capacitance

- Lumped capacitance
  - Self loading capacitance
  - Fanout gate pin capacitances
  - Fanout wire substrate/coupling capacitances
  - More on crosstalk later
- Effective capacitance
  - Not all capacitance is seen at the output due to "resistive shielding"
  - Usually a Pi (or T) model



Wire Delay Models

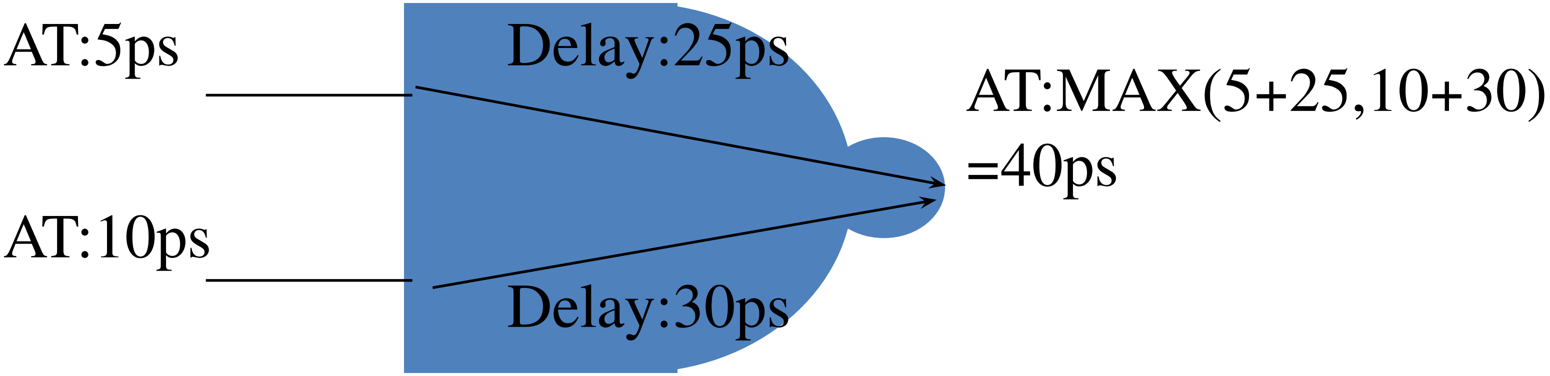
- Wire delay from driver pin to each fanout gate pin
- **More in Interconnect lecture…**

Block Based Timing Analysis

- Ignores function of gates
- Propagates the worst cumulative delay from inputs to outputs
- Based on 1966 project management technique: Program Evaluation and Review Technique (PERT)

Arrival Time

- Typically compute for (Rise,Fall) x (Early,Late)
- Also propagate slew (delay needs the slew)



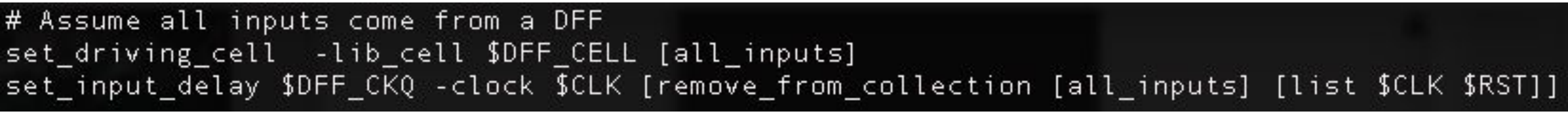
AT:5ps

AT:10ps

Delay:25ps

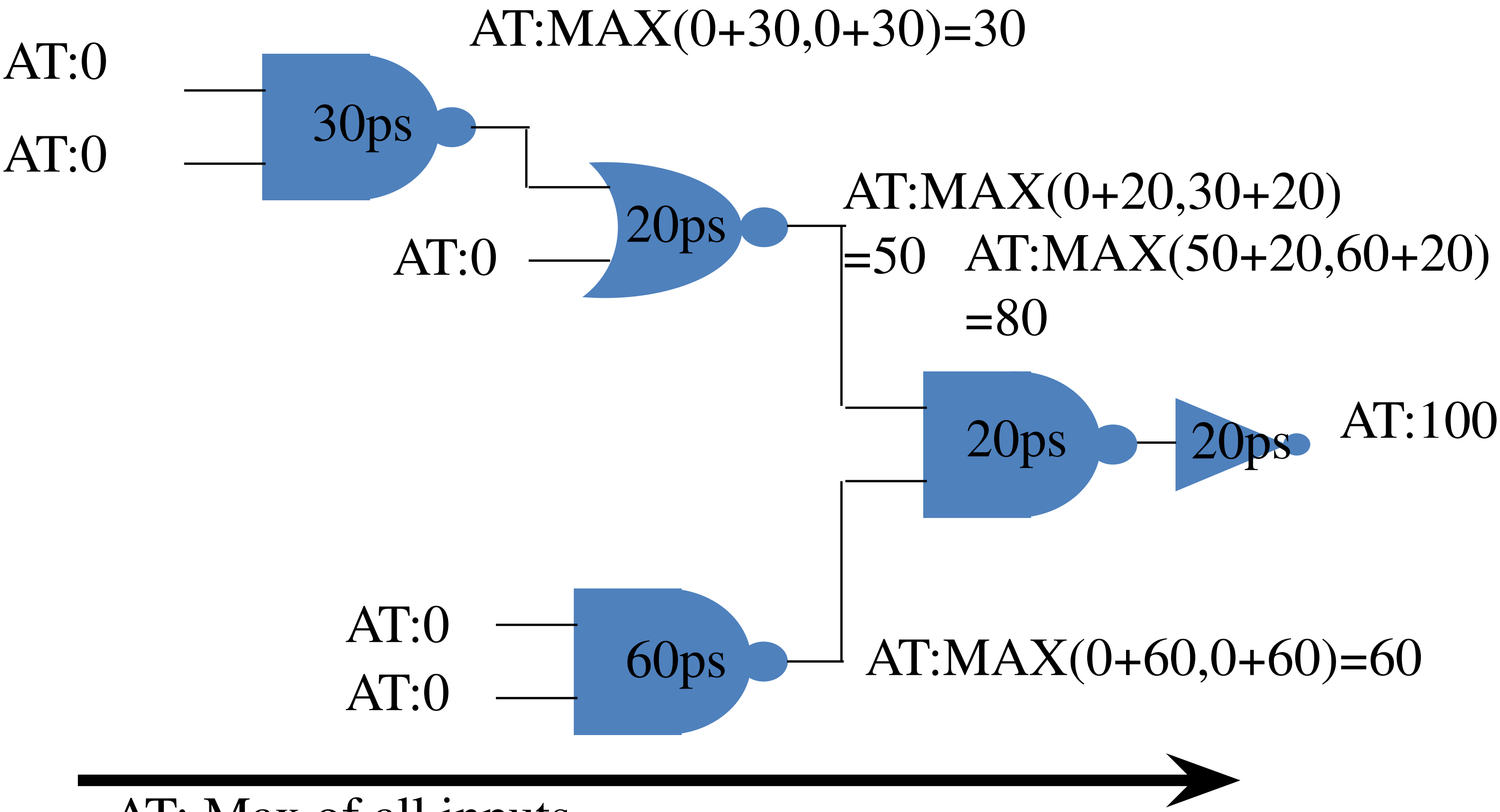
Delay:30ps

AT:MAX(5+25,10+30)
=40ps



```
# Assume all inputs come from a DFF
set_driving_cell  -lib_cell $DFF_CELL [all_inputs]
set_input_delay $DFF_CKQ -clock $CLK [remove_from_collection [all_inputs] [list $CLK $RST]]
```

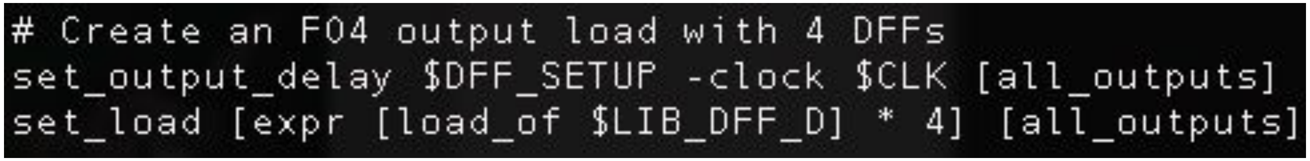
Arrival Time



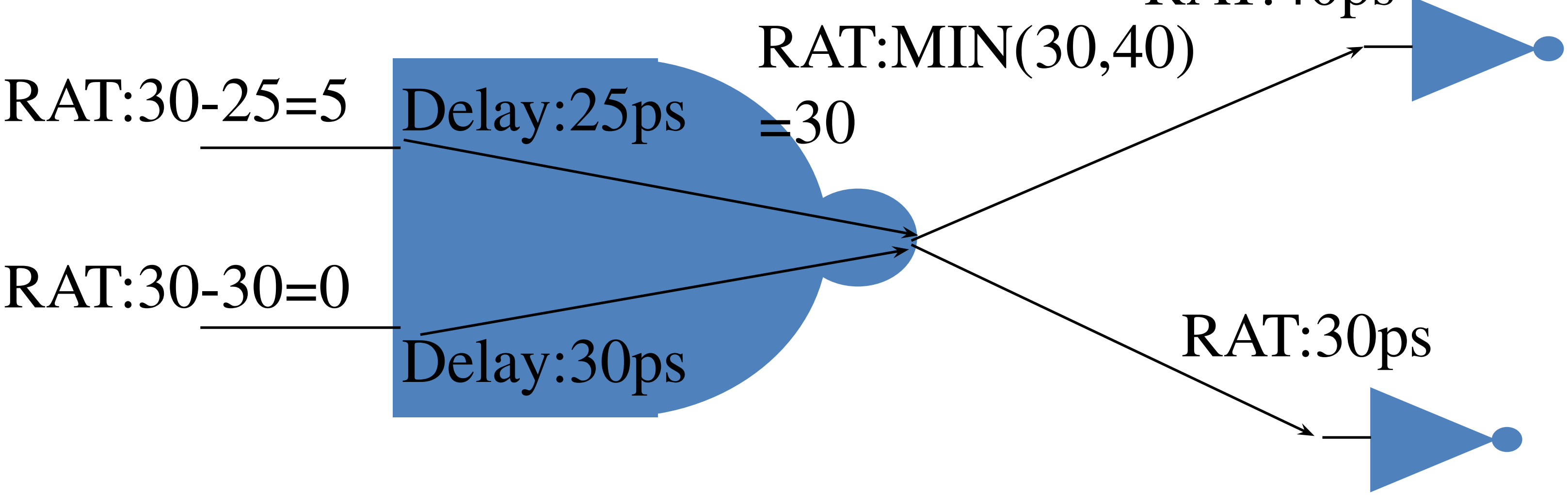
AT: Max of all inputs…

Required Arrival Time

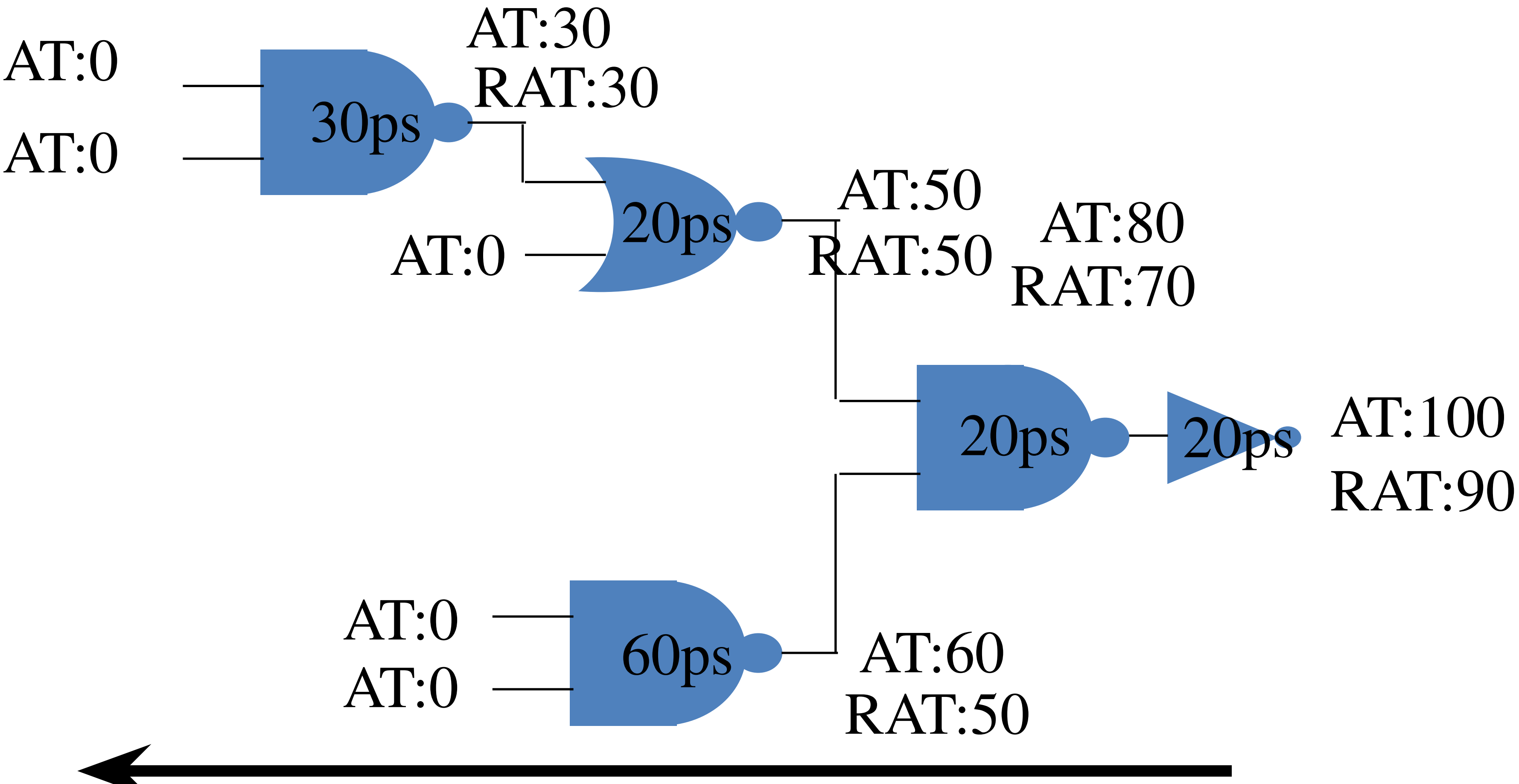
- Time signal is needed at a point to get to output by constrained time
  - Late checks set RAT to clock period at outputs/FFs
  - Early checks set RAT to hold time at output/FFs
- Typically compute for (Rise,Fall) x (Early,Late)



```
# Create an FO4 output load with 4 DFFs
set_output_delay $DFF_SETUP -clock $CLK [all_outputs]
set_load [expr [load_of $LIB_DFF_D] * 4] [all_outputs]
```

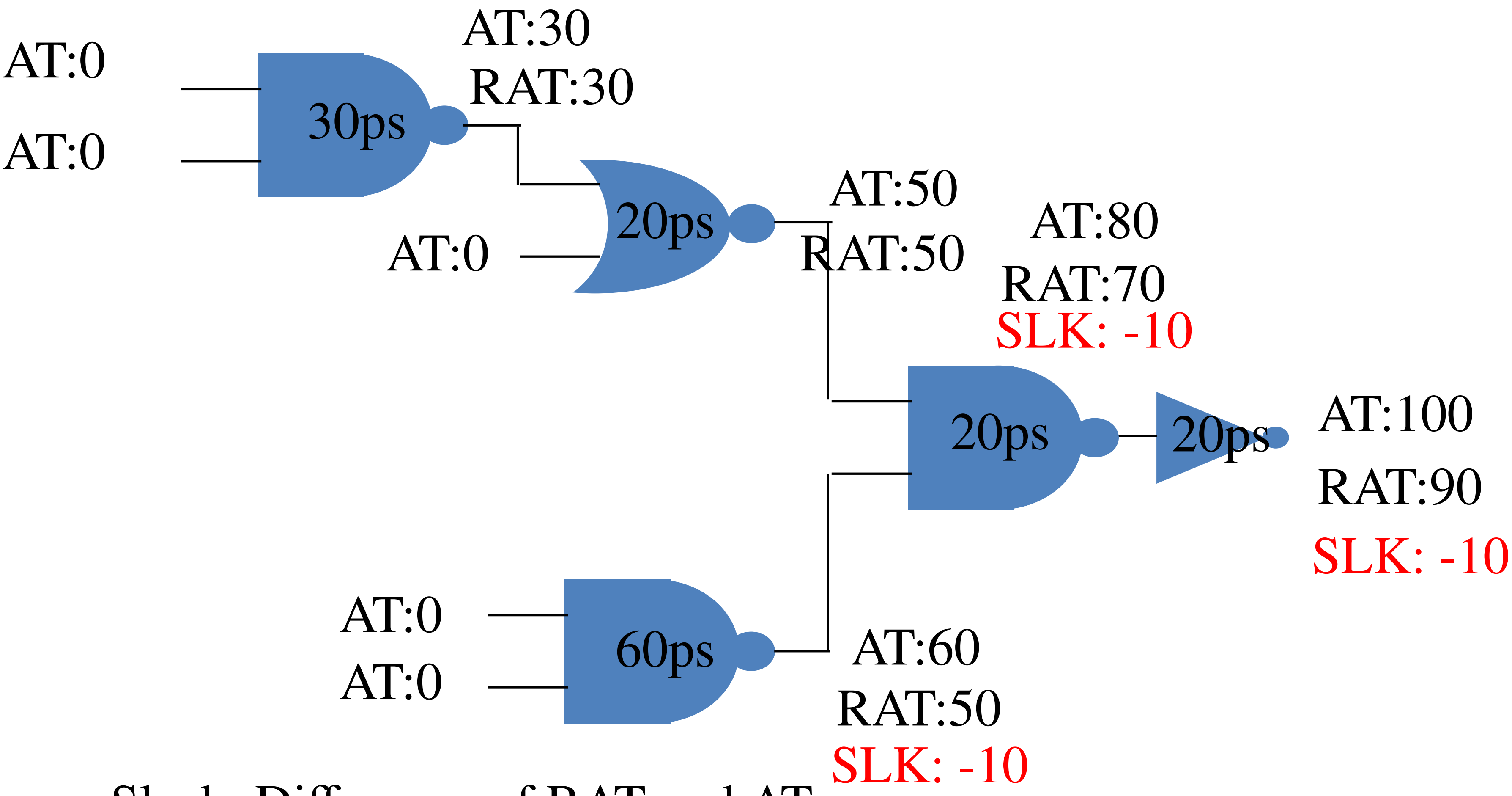


Required Arrival Time



RAT: Min of all outputs...

Slack: RAT-AT



Slack: Difference of RAT and AT

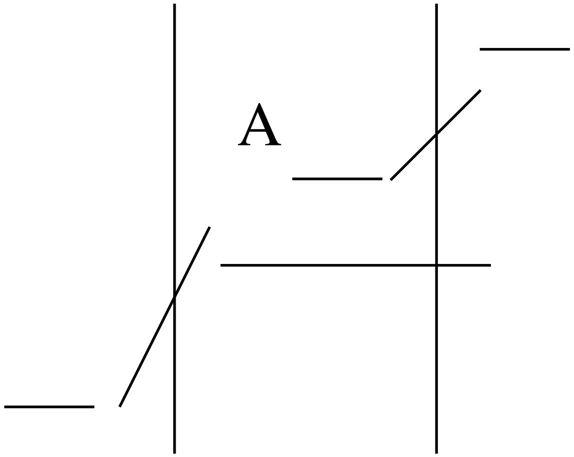
Transition Time/Slew Propagation

- The above talked about wire and gate delays, but we also must propagate slew to every wire.
- What slew do we propagate?
  - Biggest slew of all inputs? Could be conservative if it is too early.
  - Latest arriving signal? This may not determine the worst delay.
  - Selectable in STA:
    - set timing_slew_propagation_mode worst_value

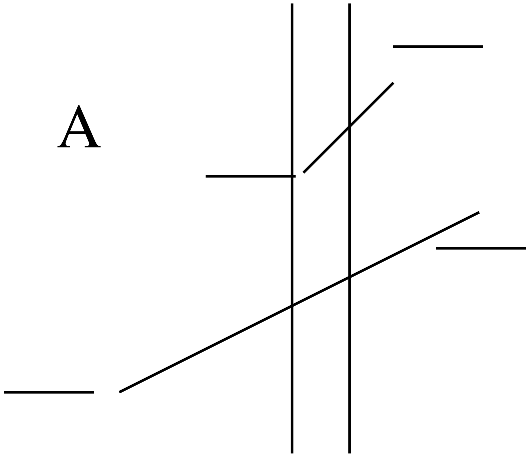
B

Slew Challenge Example

- Remember: Delay is 50%-50%, Slew is 10%-90%
- Signal A may have a bigger delay, but B has such a bad slew that it determines the output delay



A determines delay



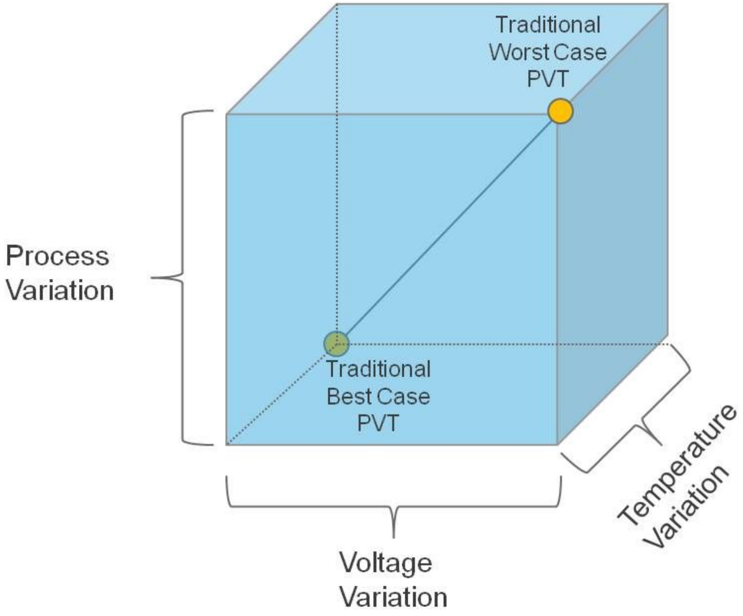
B determines delay despite arriving earlier than A

Other Delay Assumptions

- Single input switching assumption
  - Could turn on multiple paths in CMOS network which result in ramp with two different slopes
  - Switching a second input could add internal gate capacitance to output
- Slew is a single ramp
  - If second path in network turns on…
  - Inductance can have non-monotonic behavior

Timing Corners

- The previous gate and wire delays can also have different "corners"
  - Temperature
  - Process (PMOS/NMOS variation)
  - Supply Voltage
- Worst case corners
  - Early will want to use the fastest process, highest supply, lowest temperature
  - Late will want to use the slowest process, lowest supply, highest temp
- Worst case corners are becoming too pessimistic!



Demo in OpenRoad/OpenLane

Text STA reports

- Hold
- Setup
- Corners

GUI browser