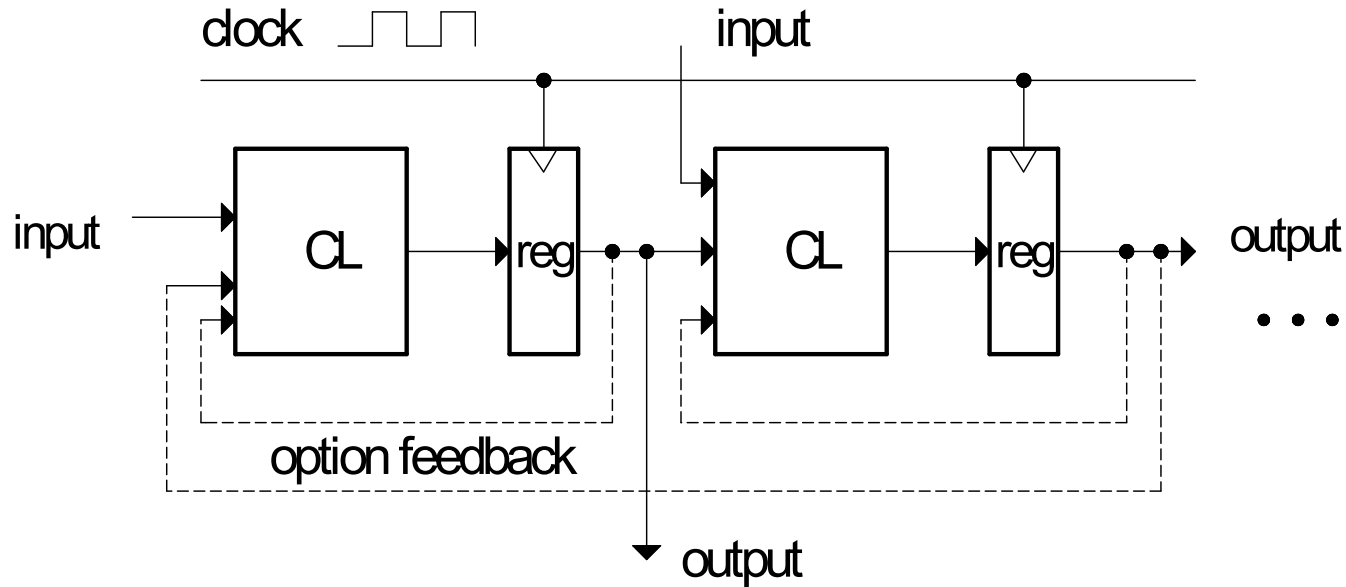


EECS150 - Digital Design
Lecture 13 - Circuit Timing

Feb 28, 2012
John Wawrzynek

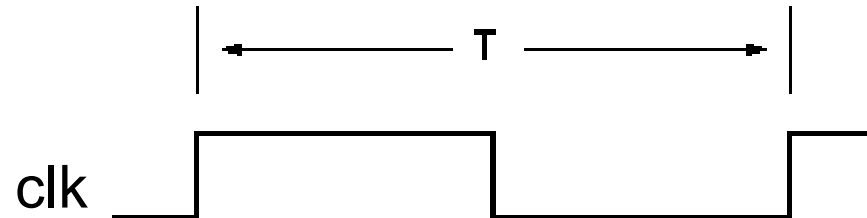
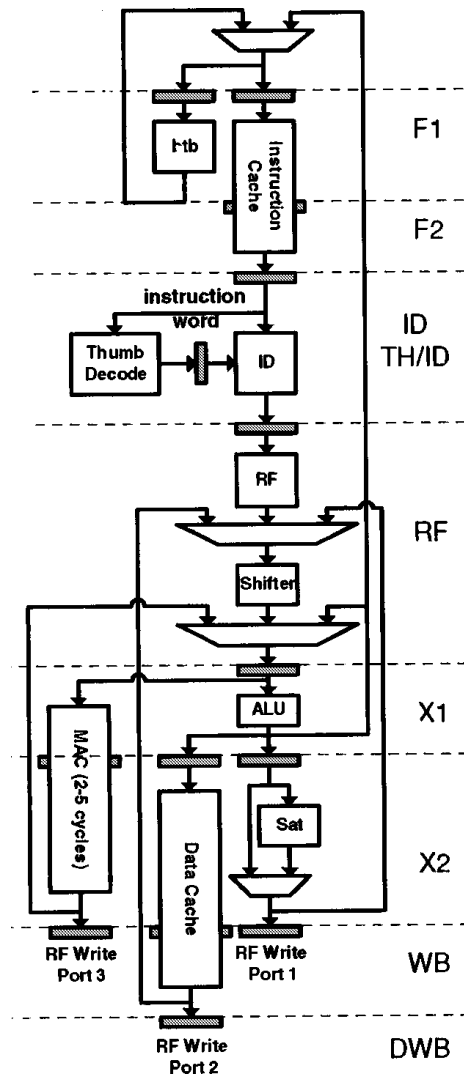
Performance, Cost, Power



- How do we measure performance?
operations/sec? cycles/sec?
- Performance is directly proportional to clock frequency.
Although it may not be the entire story:
Ex: CPU performance
= # instructions X CPI X clock period

Timing Analysis

ARM processor Microarch

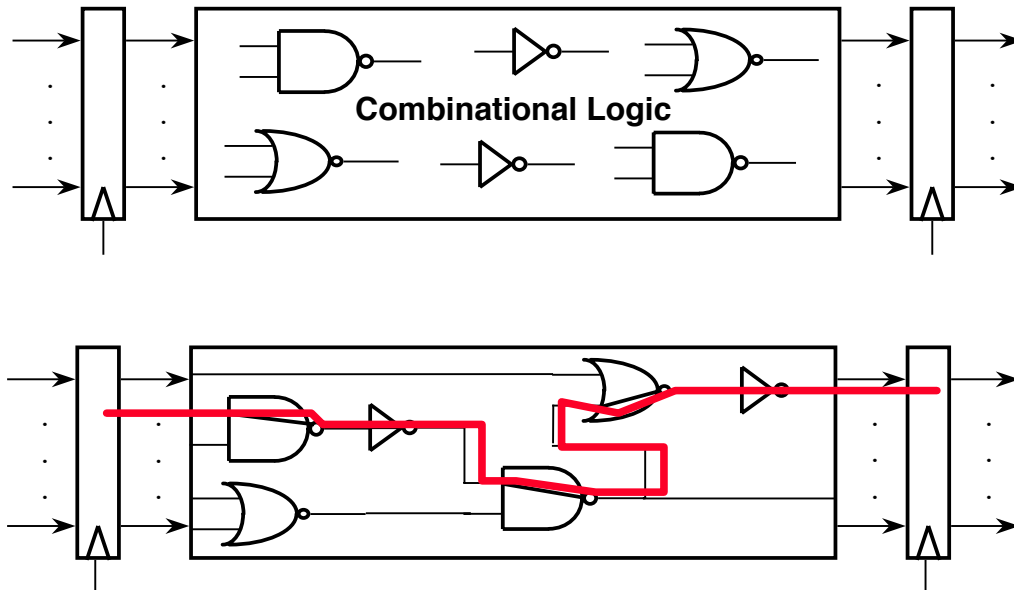
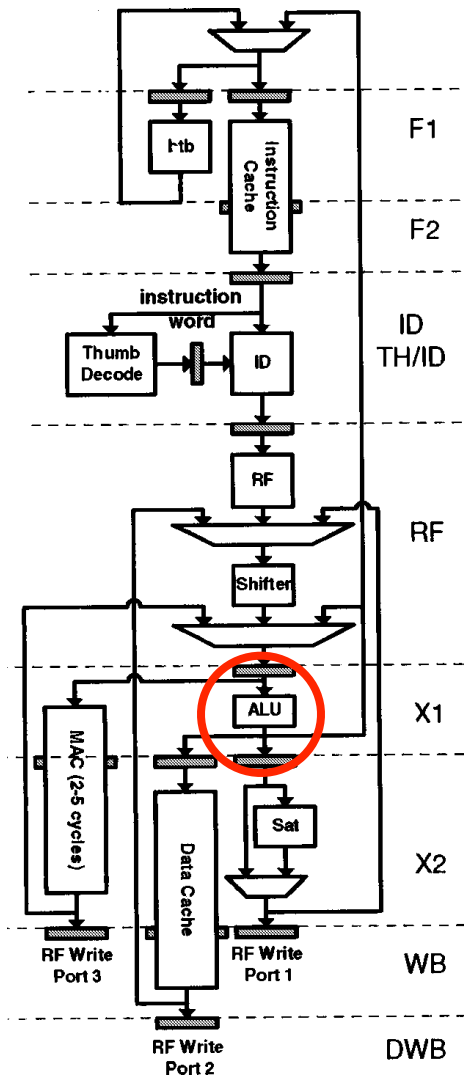


Timing Analysis

What is the smallest T that produces correct operation?

f	T
1 MHz	1 μ s
10 MHz	100 ns
100 MHz	10 ns
1 GHz	1 ns

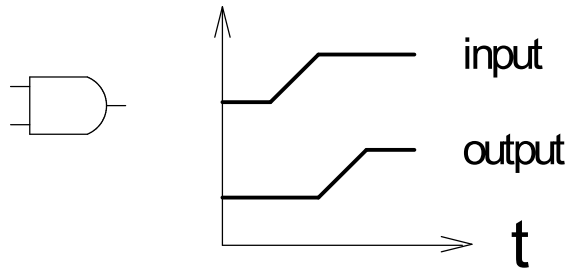
Timing Analysis and Logic Delay



If $T >$ worst-case delay through CL, does this ensure correct operation?

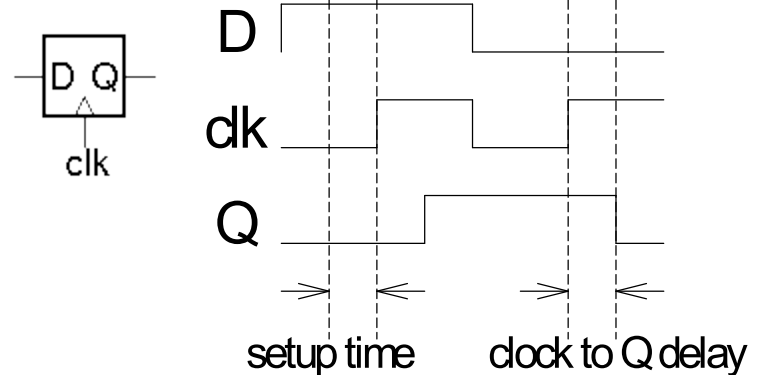
Limitations on Clock Rate

1 Logic Gate Delay



What are typical delay values?

2 Delays in flip-flops

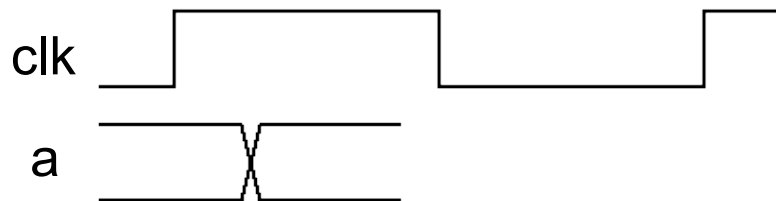
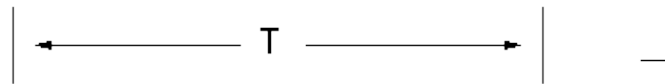
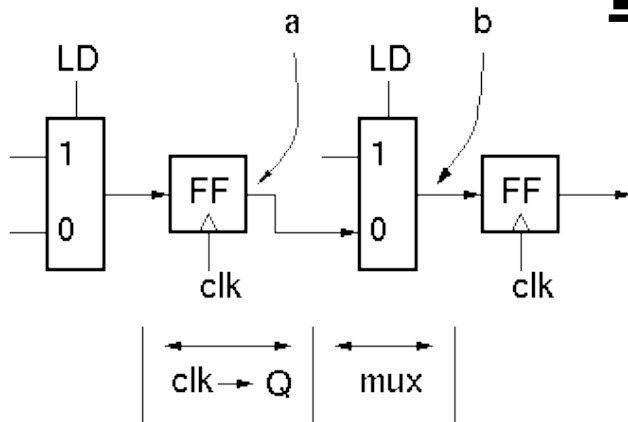


Both times contribute to limiting the clock period.

- What must happen in one clock cycle for correct operation?
 - All signals connected to FF (or memory) inputs must be ready and "setup" before rising edge of clock.
 - For now we assume perfect clock distribution (all flip-flops see the clock at the same time).

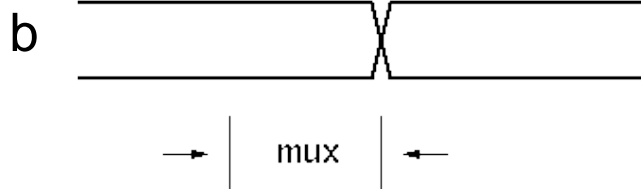
Example

Parallel to serial
converter circuit

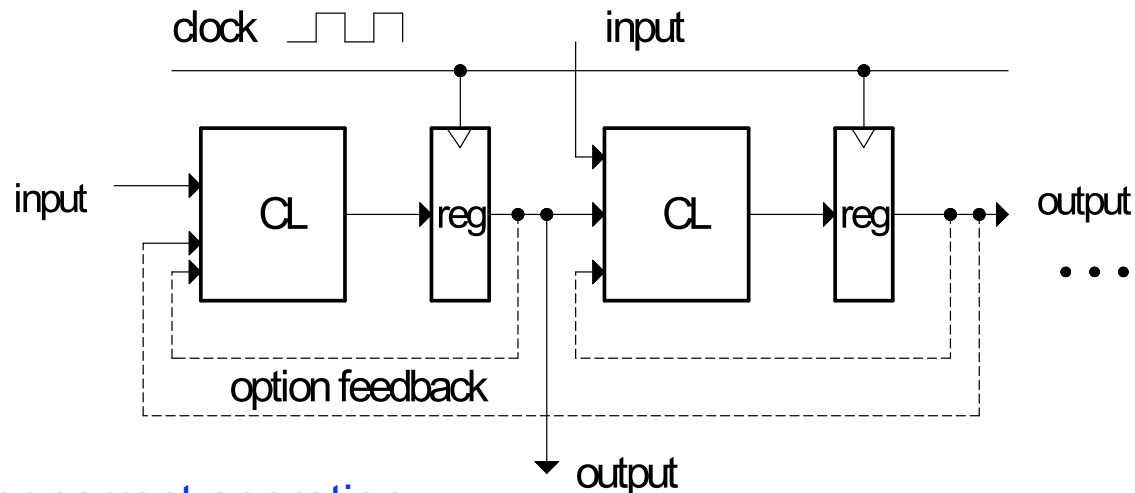


$$T \geq \text{time}(\text{clk} \rightarrow \text{Q}) + \text{time}(\text{mux}) + \text{time}(\text{setup})$$

$$T \geq \tau_{\text{clk} \rightarrow \text{Q}} + \tau_{\text{mux}} + \tau_{\text{setup}}$$



In General ...



For correct operation:

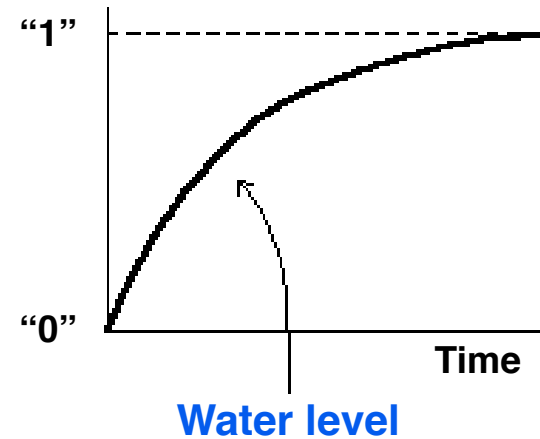
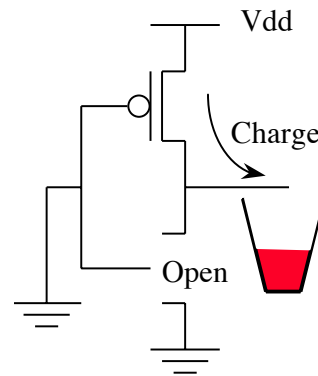
$$T \geq \tau_{\text{clk} \rightarrow \text{Q}} + \tau_{\text{CL}} + \tau_{\text{setup}} \quad \text{for all paths.}$$

- How do we enumerate **all** paths?
 - Any circuit input or register output to any register input or circuit output?
- Note:
 - "setup time" for outputs is a function of what it connects to.
 - "clk-to-q" for circuit inputs depends on where it comes from.

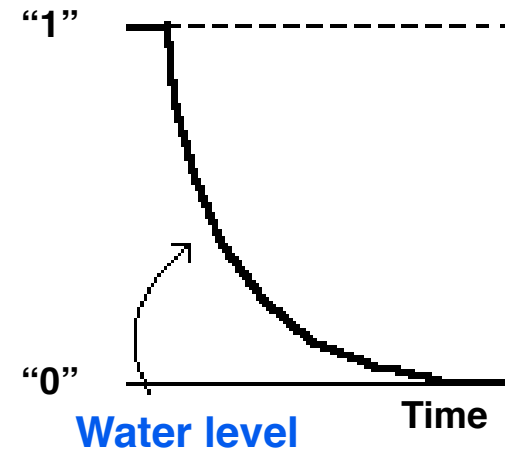
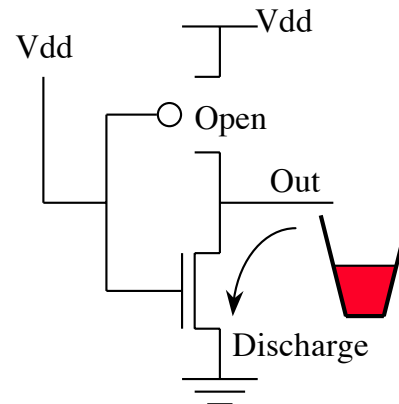
CL Delay: Transistors as water valves

If electrons are water molecules,
and a capacitor a bucket ...

A “on” p-FET fills
up the capacitor
with charge.

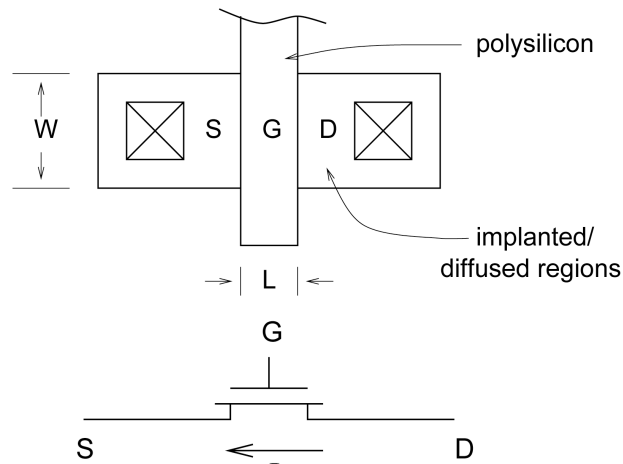


A “on” n-FET
empties the bucket.



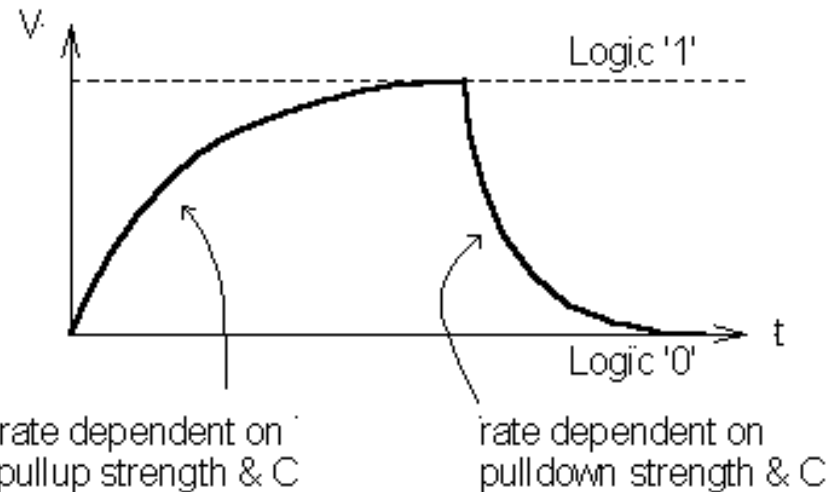
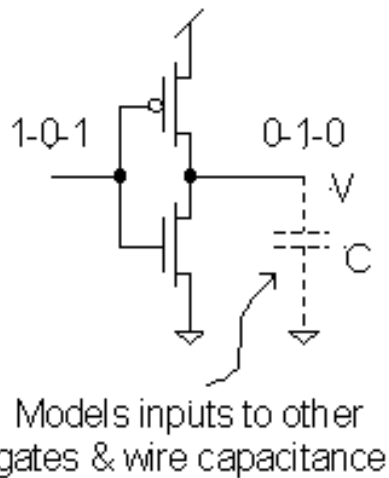
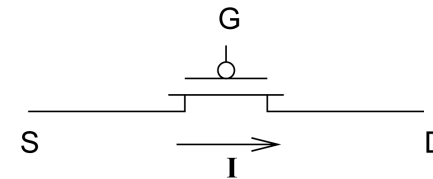
Transistors as Conductors

- Improved Transistor Model:
nFET



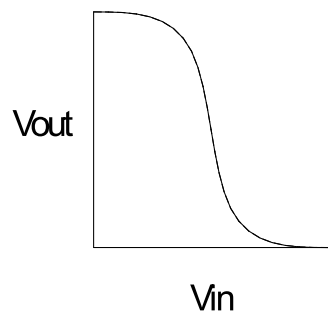
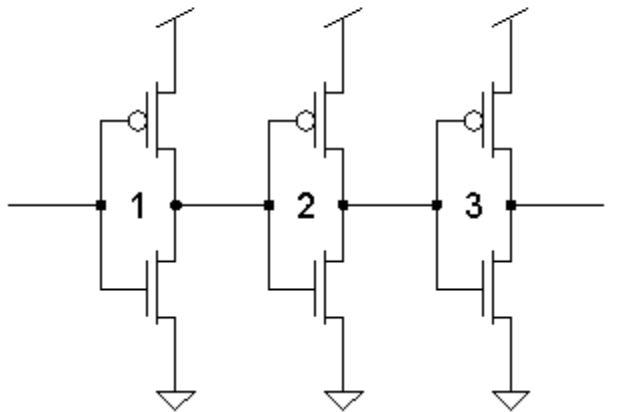
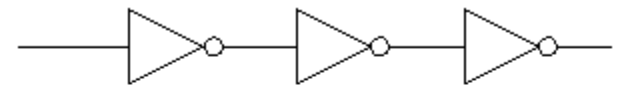
- We refer to transistor "strength" as the amount of current that flows for a given V_{ds} and V_{gs} .
- The strength is linearly proportional to the ratio of W/L .

pFET



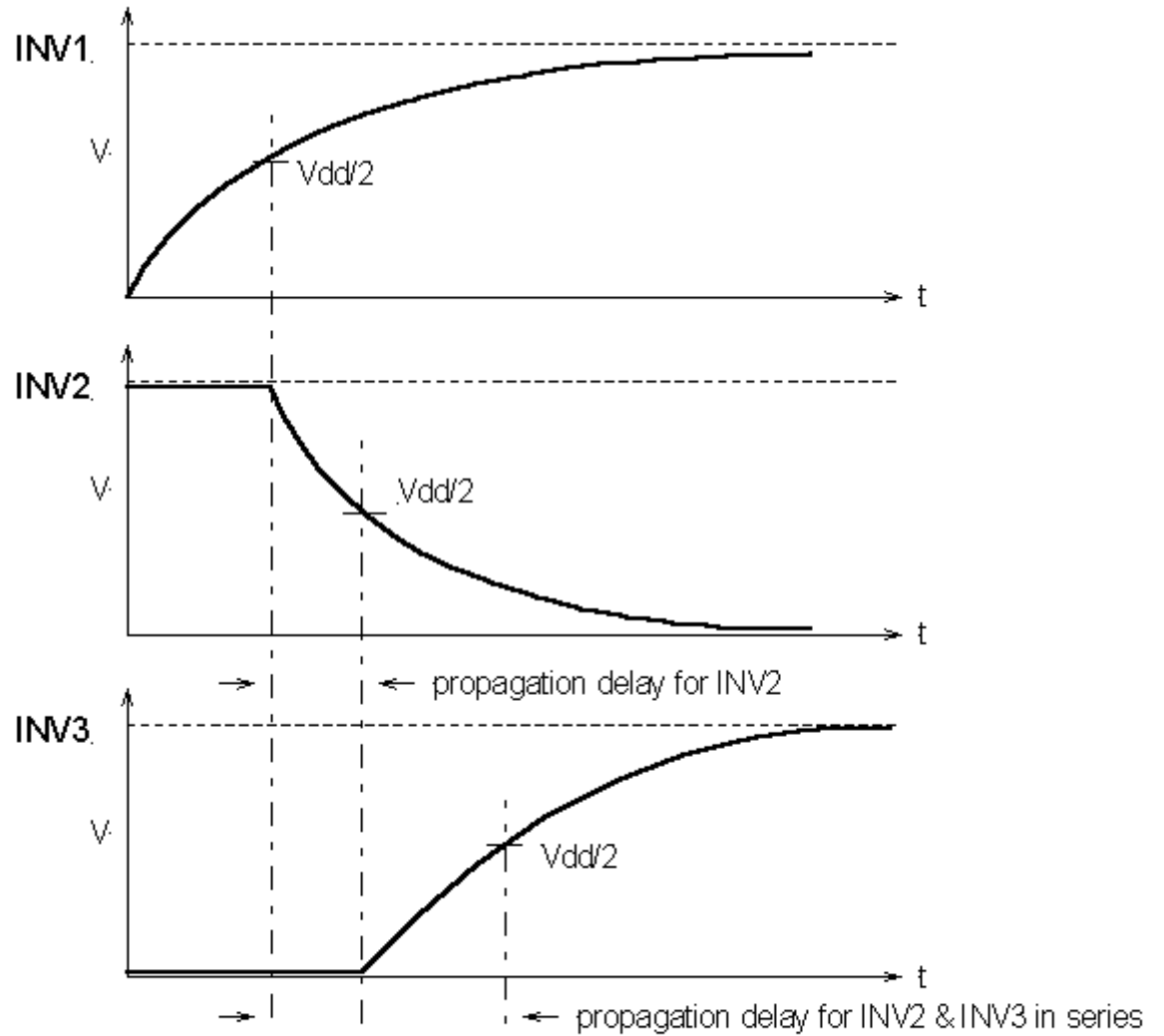
Gate Delay is the Result of Cascading

- Cascaded gates:



“transfer curve” for inverter.

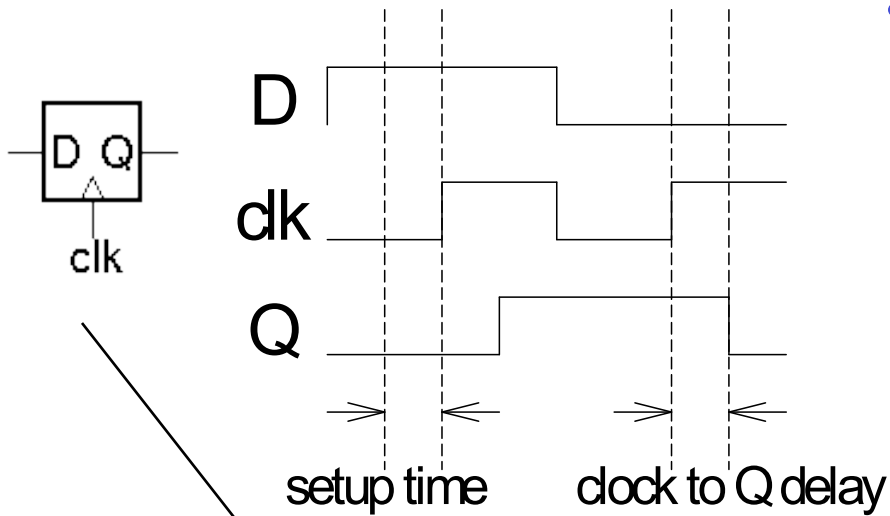
Spring 2012



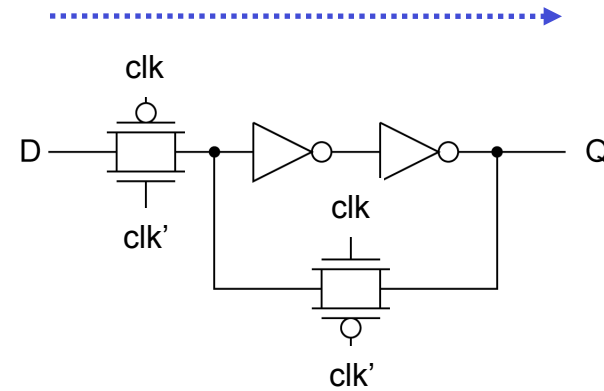
In general:

prop. delay = sum of individual prop. delays of gates in series.

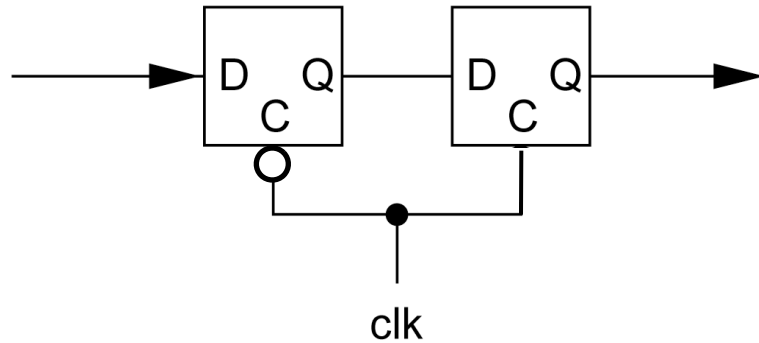
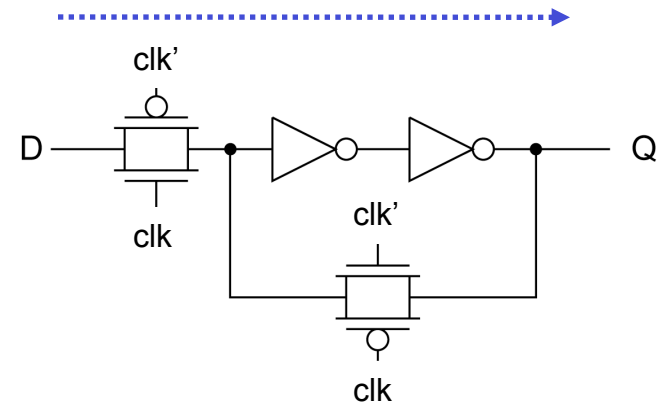
Delay in Flip-flops



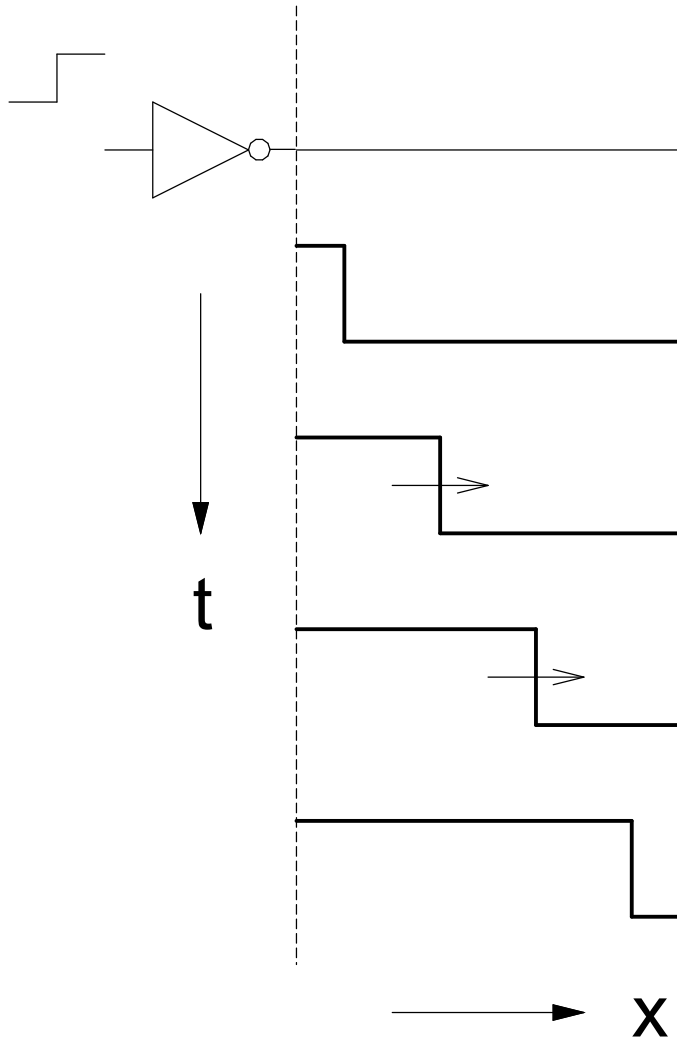
- **Setup time** results from delay through *first* latch.



- **Clock to Q delay** results from delay through *second* latch.



Wire Delay

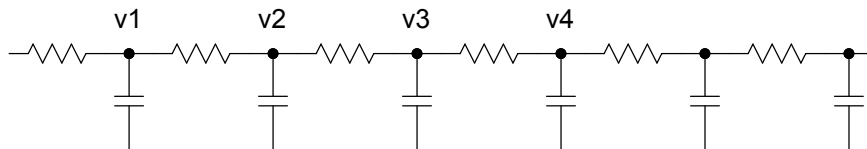


- Ideally, wires behave as “transmission lines”:
 - signal wave-front moves close to the speed of light
 - $\sim 1\text{ft/ns}$
 - Time from source to destination is called the “transit time”.
 - In ICs most wires are short, and the transit times are relatively short compared to the clock period and can be ignored.
 - Not so on PC boards.

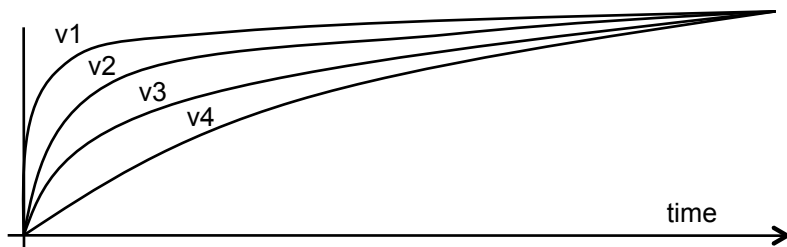
Wire Delay

- Even in those cases where the transmission line effect is negligible:

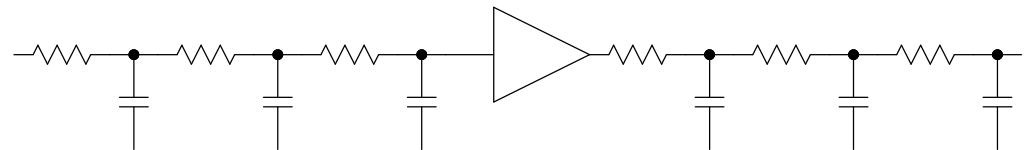
- Wires possess distributed resistance and capacitance



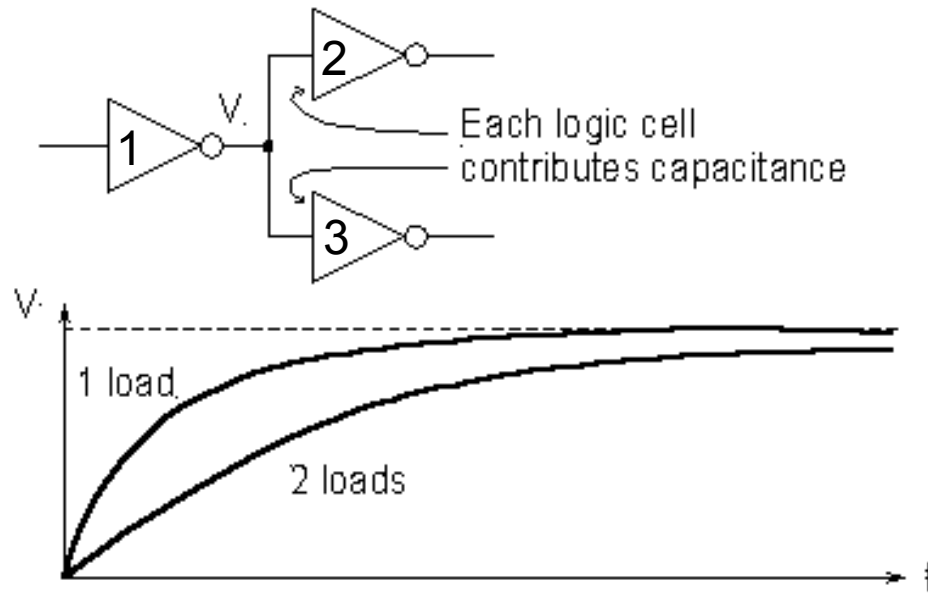
- Time constant associated with distributed RC is proportional to the *square* of the length



- For **short wires** on ICs, resistance is insignificant (relative to effective R of transistors), but C is important.
 - Typically around half of C of gate load is in the wires.
- For **long wires** on ICs:
 - busses, clock lines, global control signal, etc.
 - Resistance is significant, therefore distributed RC effect dominates.
 - signals are typically “rebuffered” to reduce delay:



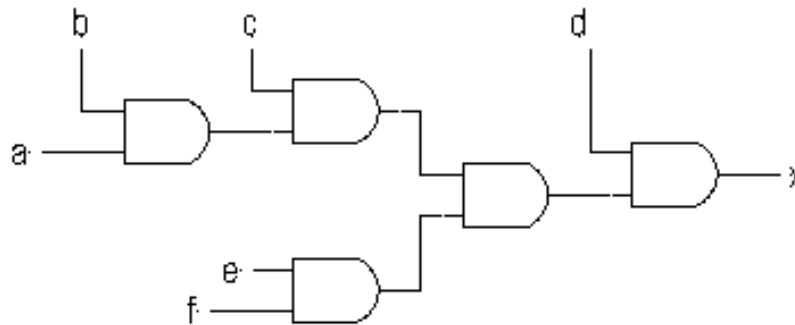
Delay and “Fan-out”



- The delay of a gate is proportional to its output capacitance. Connecting the output of gate to more than one other gate increases its output capacitance. It takes increasingly longer for the output of a gate to reach the switching threshold of the gates it drives as we add more output connections.
- Driving wires also contributes to fan-out delay.
- What can be done to remedy this problem in large fan-out situations?

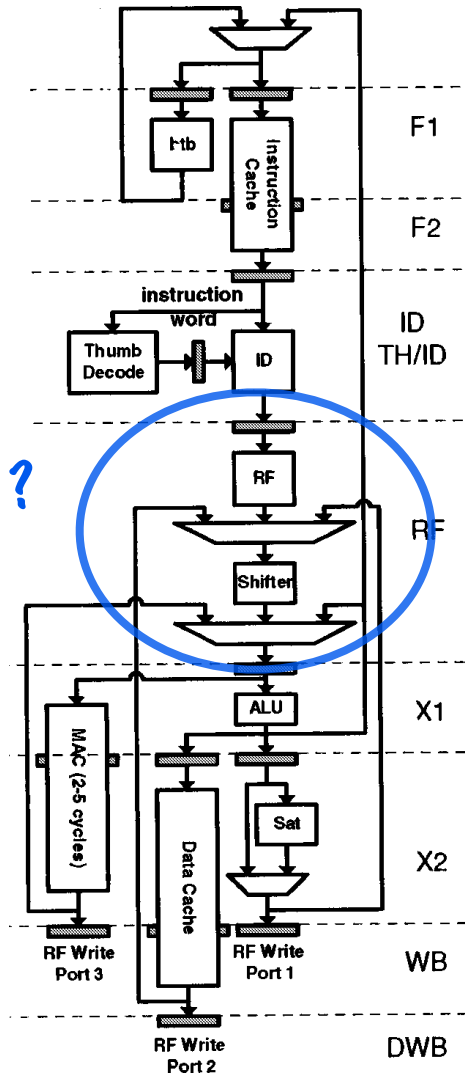
“Critical” Path

- *Critical Path*: the path in the entire design with the maximum delay.
 - This could be from state element to state element, or from input to state element, or state element to output, or from input to output (unregistered paths).
- For example, what is the critical path in this circuit?



- Why do we care about the *critical path*?

Searching for processor critical path



Must consider all connected register pairs, paths from input to register, register to output. Don't forget the controller.

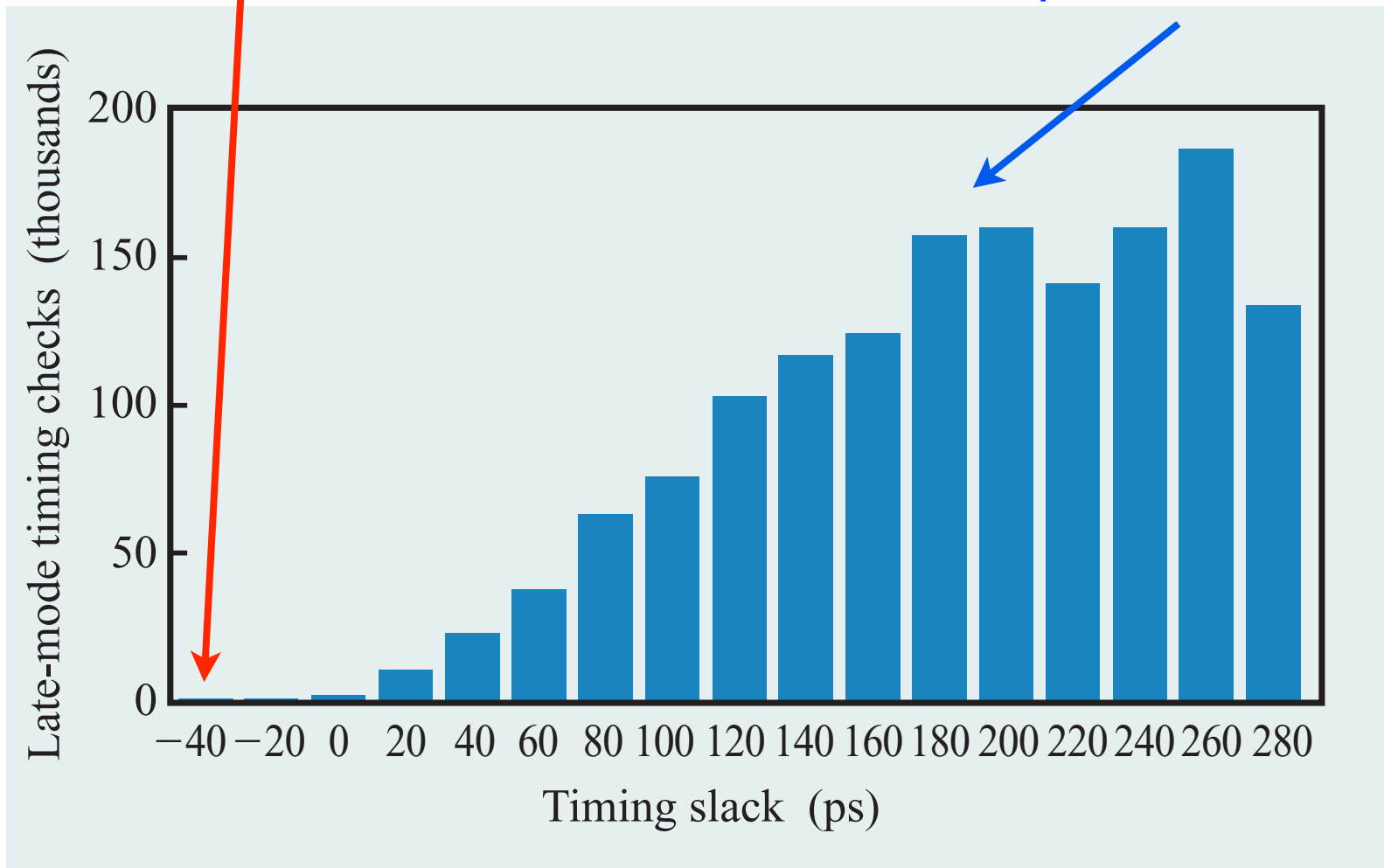
- Design tools help in the search.
- **Synthesis tools** report delays on paths,
- Special **static timing analyzers** accept a design netlist and report path delays,
- and, of course, **simulators** can be used to determine timing performance.

Tools that are expected to do something about the timing behavior (such as synthesizers), also include provisions for specifying input arrival times (relative to the clock), and output requirements (set-up times of next stage).

Real Stuff: Timing Analysis

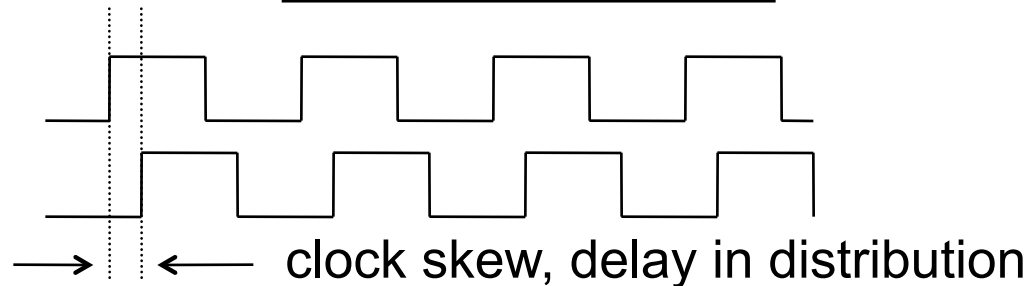
The critical path

Most paths have hundreds of picoseconds to spare.



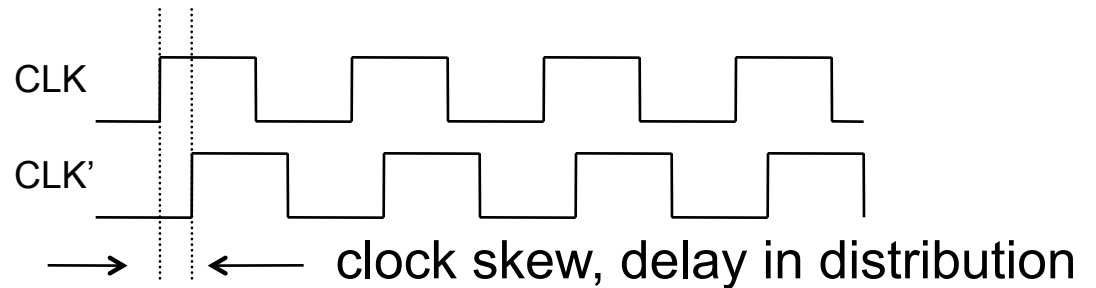
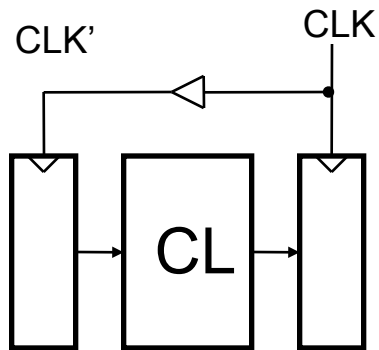
From "The circuit and physical design of the POWER4 microprocessor", IBM J Res and Dev, 46:1, Jan 2002, J.D. Warnock et al.

Clock Skew



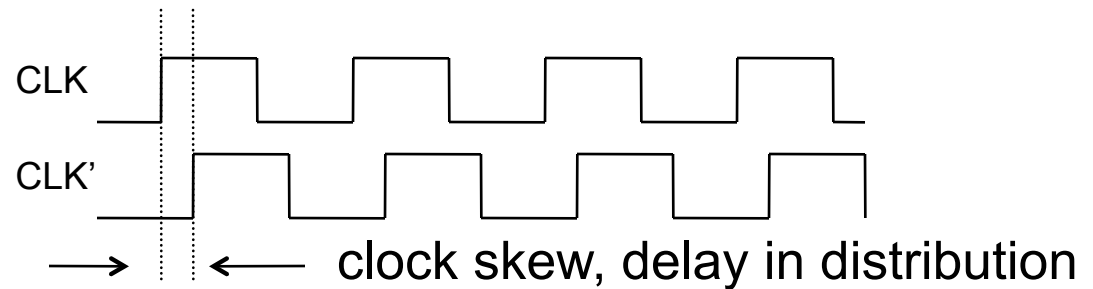
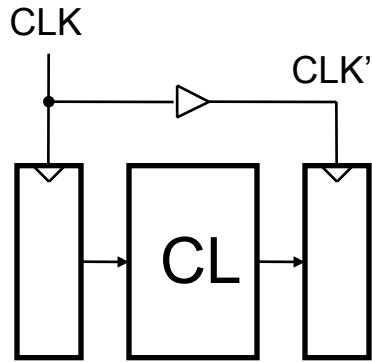
- Unequal delay in distribution of the clock signal to various parts of a circuit:
 - if not accounted for, can lead to erroneous behavior.
 - Comes about because:
 - clock wires have delay,
 - circuit is designed with a different number of clock buffers from the clock source to the various clock loads, or
 - buffers have unequal delay.
 - All synchronous circuits experience some clock skew:
 - more of an issue for high-performance designs operating with very little extra time per clock cycle.

Clock Skew (cont.)



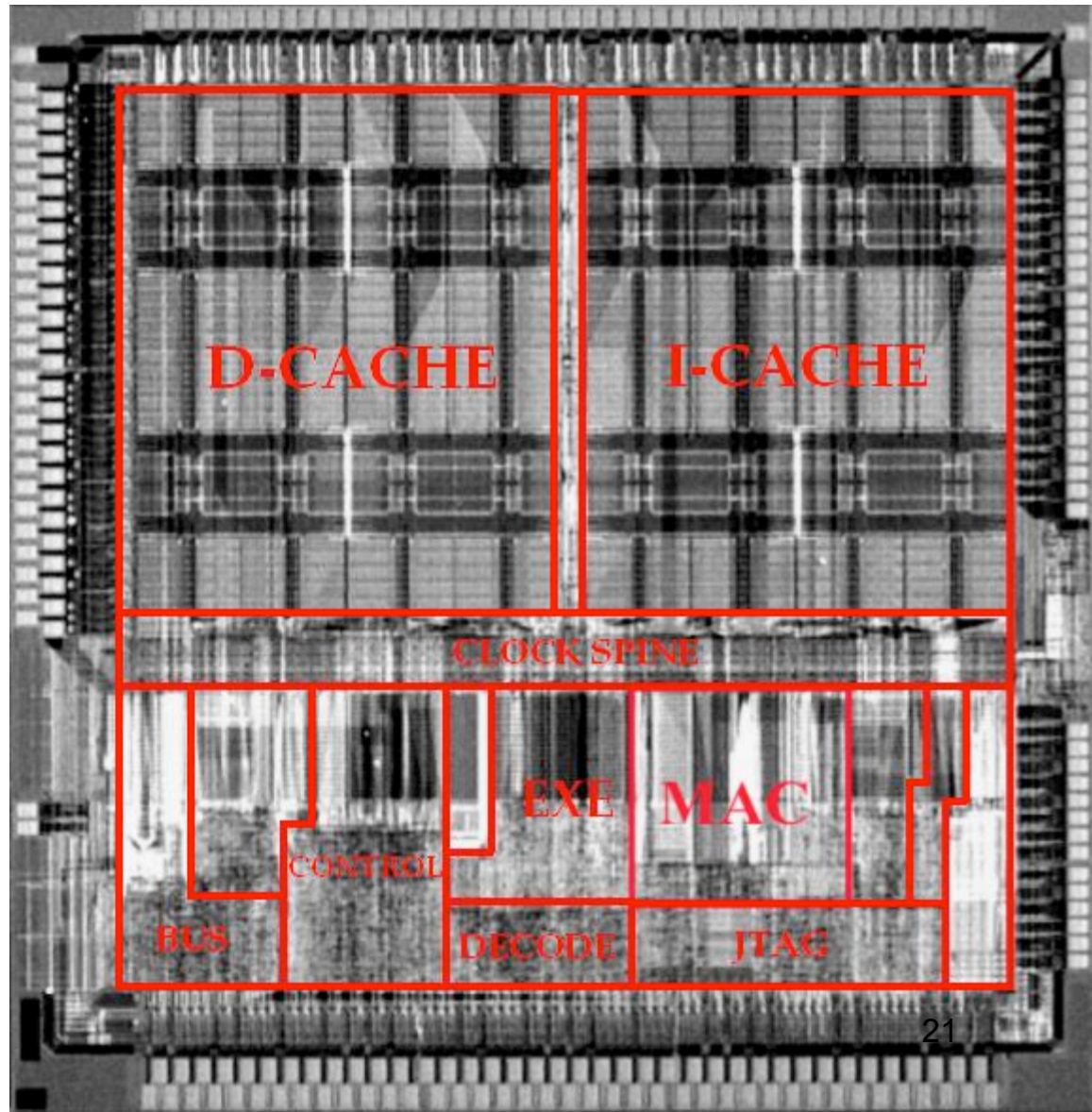
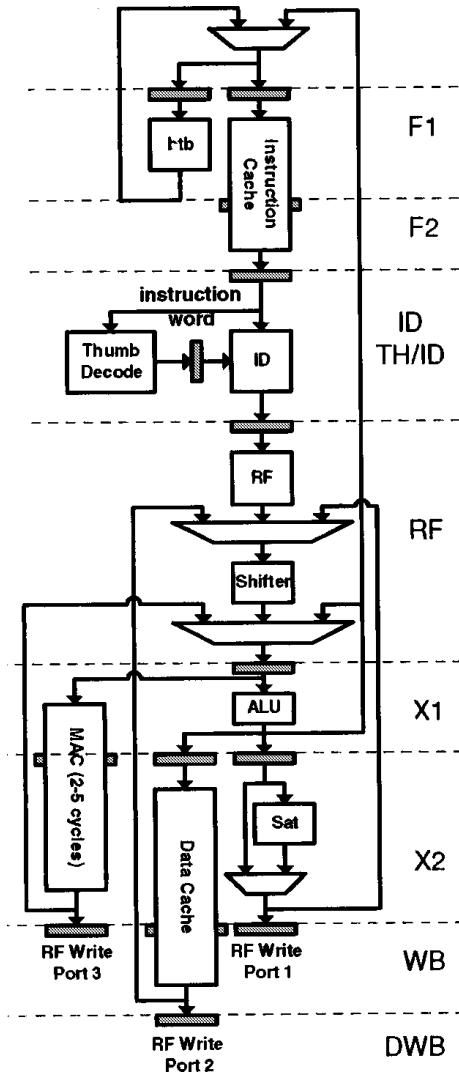
- If clock period $T = T_{CL} + T_{setup} + T_{clk \rightarrow Q}$, circuit will fail.
- Therefore:
 1. Control clock skew
 - a) Careful clock distribution. Equalize path delay from clock source to all clock loads by controlling wires delay and buffer delay.
 - b) don't "gate" clocks in a non-uniform way.
 2. $T \geq T_{CL} + T_{setup} + T_{clk \rightarrow Q} + \text{worst case skew.}$
- Most modern large high-performance chips (microprocessors) control end to end clock skew to a small fraction of the clock period.

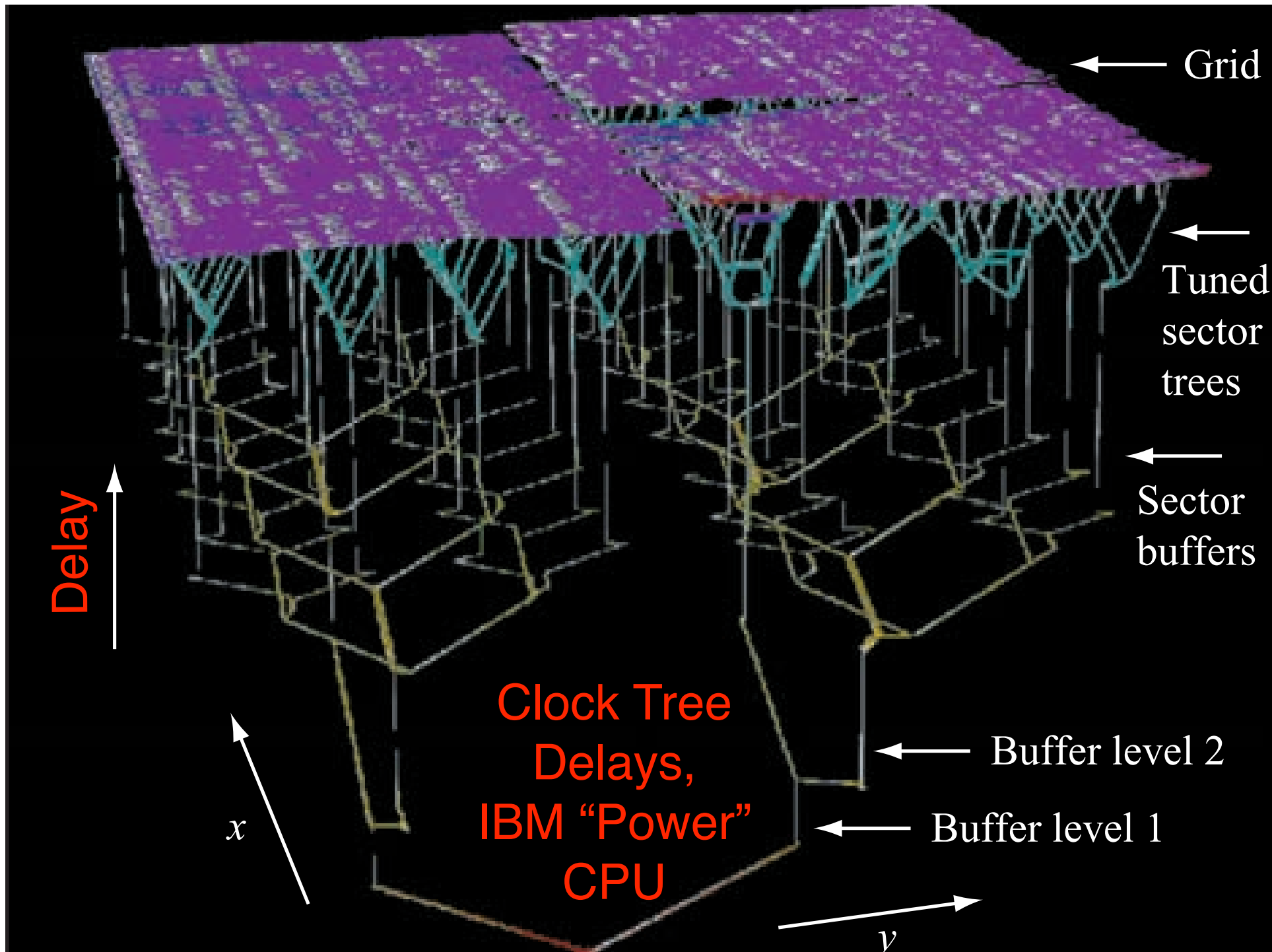
Clock Skew (cont.)

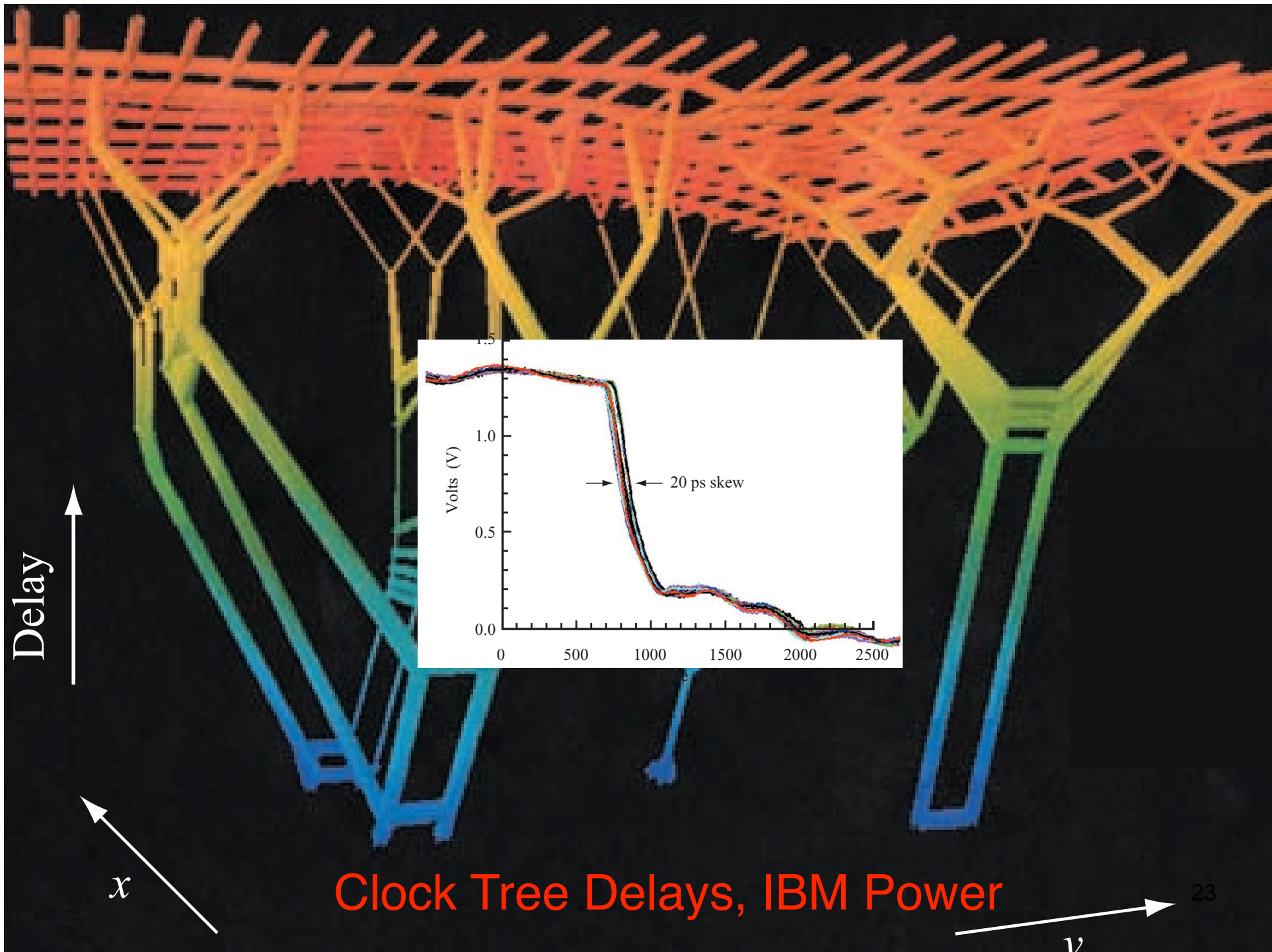


- Note reversed buffer.
- In this case, clock skew actually provides *extra time* (adds to the effective clock period).
- This effect has been used to help run circuits at higher clock rates. Risky business!

Real Stuff: Floorplanning Intel XScale 80200







Clock Tree Delays, IBM Power