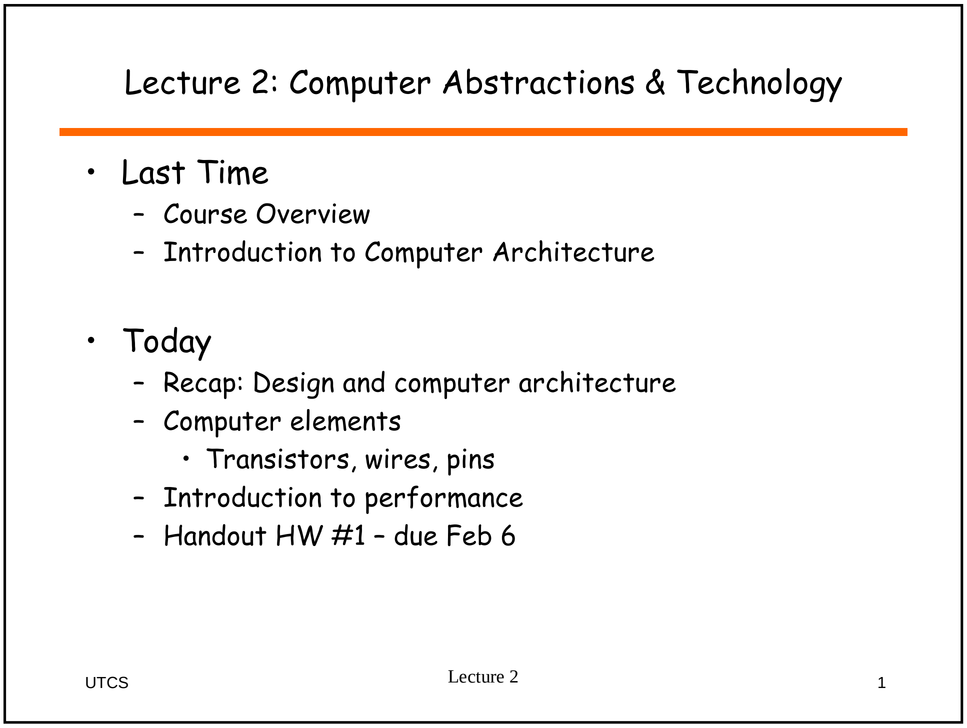
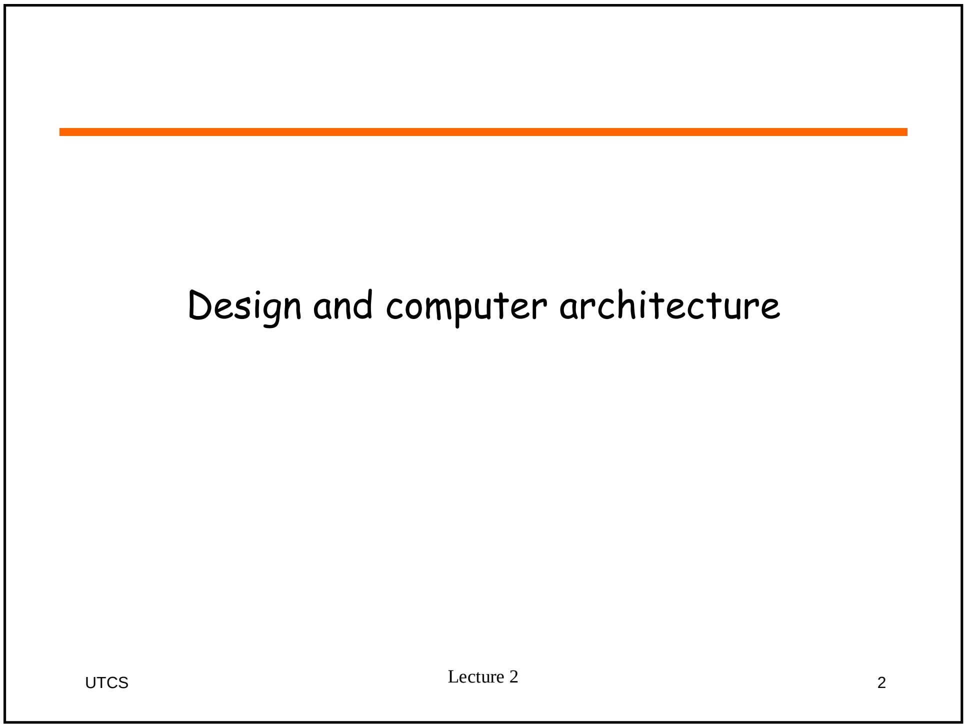


Lecture 2: Computer Abstractions & Technology

- Last Time
  - Course Overview
  - Introduction to Computer Architecture

- Today
  - Recap: Design and computer architecture
  - Computer elements
    - Transistors, wires, pins
  - Introduction to performance
  - Handout HW #1 – due Feb 6



Design and computer architecture

How to design something:

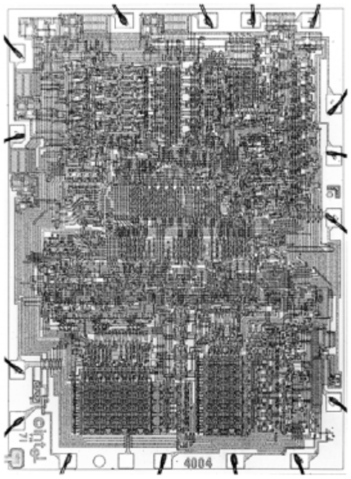
- List goals
- List constraints
- Generate ideas for possible designs
- Evaluate the different designs
- Pick the best design
- Refine it

In reality, this process is iterative.

As constraints change, best design will change too.

[Use classroom remodel as example of design process]

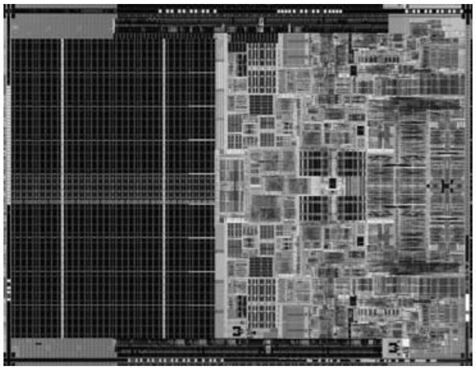
Intel 4004 - 1971



- The first microprocessor

- 2,300 transistors
- 108 KHz
- 10μm process

Intel Core 2 Duo - 2006



- "State of the art"
- 291 million transistors
- 3 GHz
- 0.065 μm (65 nm) process
- Could fit ~100,000 4004s on this chip!

Changing Technology leads to Changing Architecture

- 1970s
  - multi-chip CPUs
  - semiconductor memory very expensive
  - microcoded control
  - complex instruction sets (good code density)
- 1980s
  - single-chip CPUs, on-chip RAM feasible
  - simple, hard-wired control
  - simple instruction sets
  - small on-chip caches
- 1990s
  - lots of transistors
  - complex control to exploit instruction-level parallelism
- 2000s
  - even more transistors
  - slow wires
  - multi-core chips

Don’t forget the simple view

All a computer does is

- Store and move data
- Communicate with the external world
- Do these two things conditionally
- According to a recipe specified by a programmer

It’s complex because

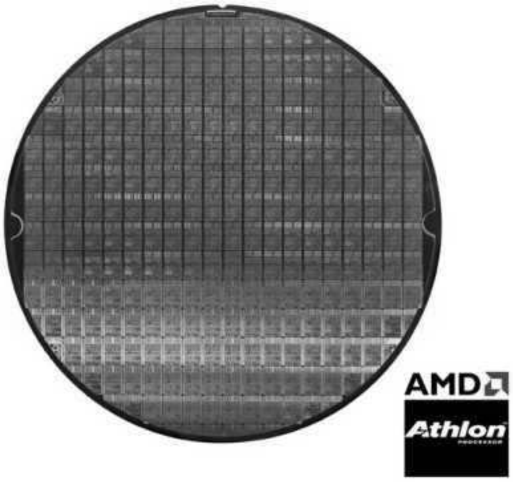
- We want it to be fast
- We want it to be reliable and secure
- We want it to be simple to use
- It must obey the laws of physics

Computer Abstractions & Technology

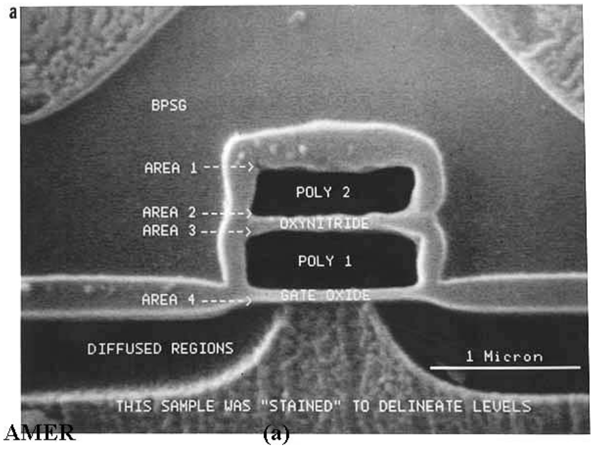
Computer Elements

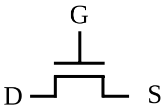
- Transistors (computing)
  - How can they be connected to do something useful?
  - How do we evaluate how fast a logic block is?
- Wires (transporting)
  - What and where are they?
  - How can they be modeled?
- Memories (storing)
  - SRAM vs. DRAM

What Comes out of the Fab?



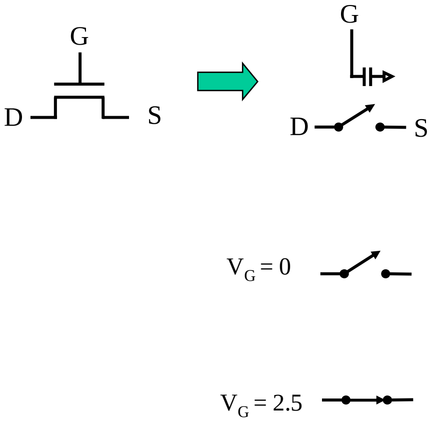
The Mighty Transistor!

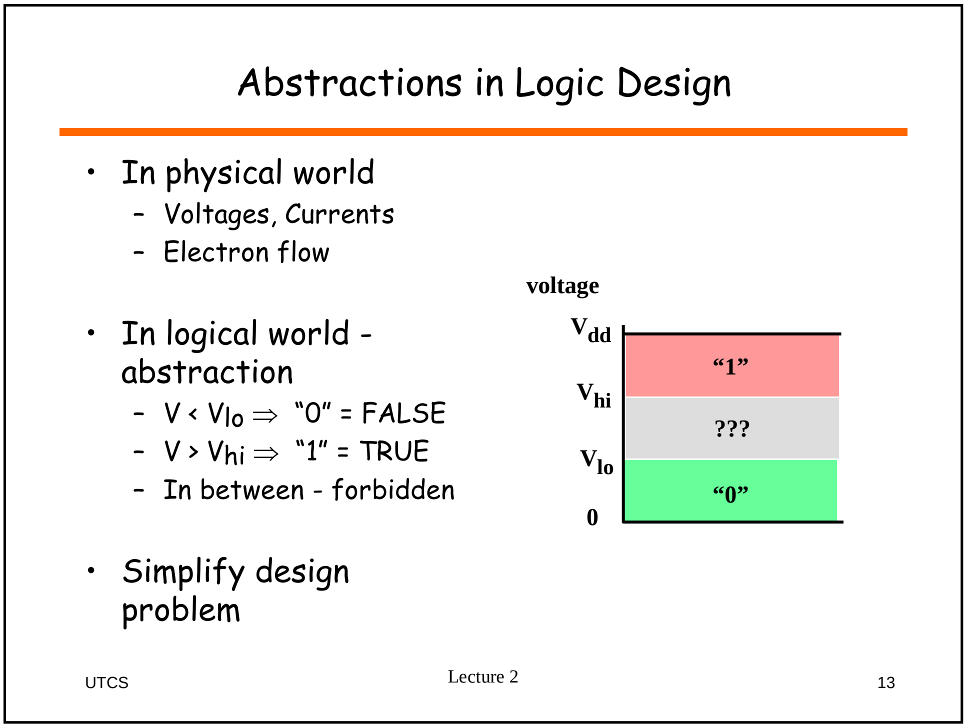




Transistor As a Switch

- Ideal Voltage Controlled Switch
- Three terminals
  - Gate
  - Drain
  - Source

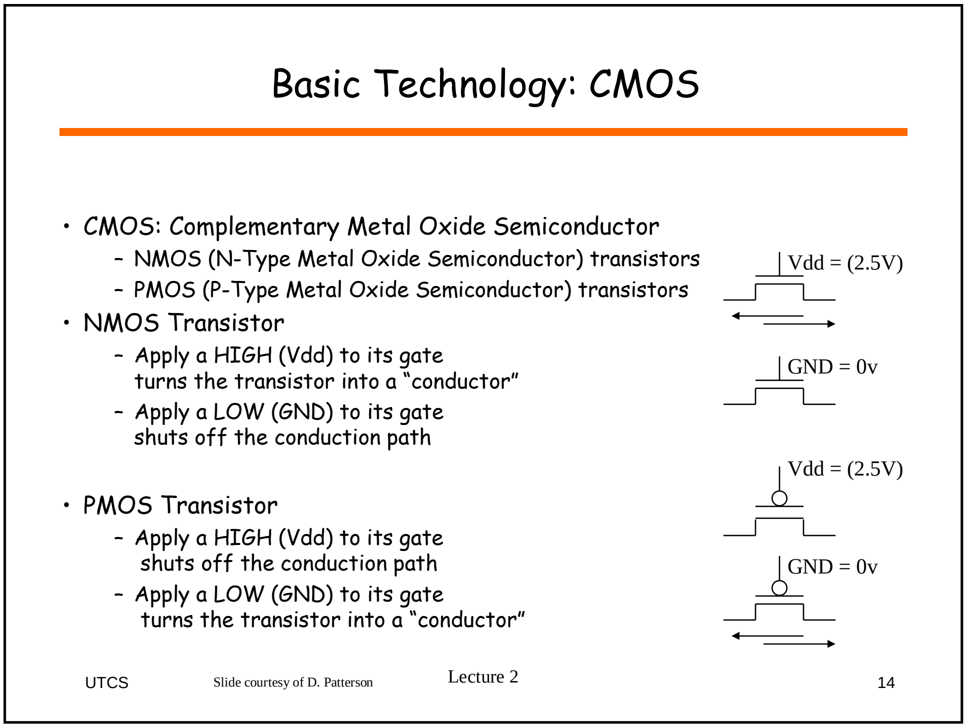




Abstractions in Logic Design

- In physical world
  - Voltages, Currents
  - Electron flow
- In logical world - abstraction
  - $V < V_{lo} \Rightarrow$ "0" = FALSE
  - $V > V_{hi} \Rightarrow$ "1" = TRUE
  - In between - forbidden
- Simplify design problem

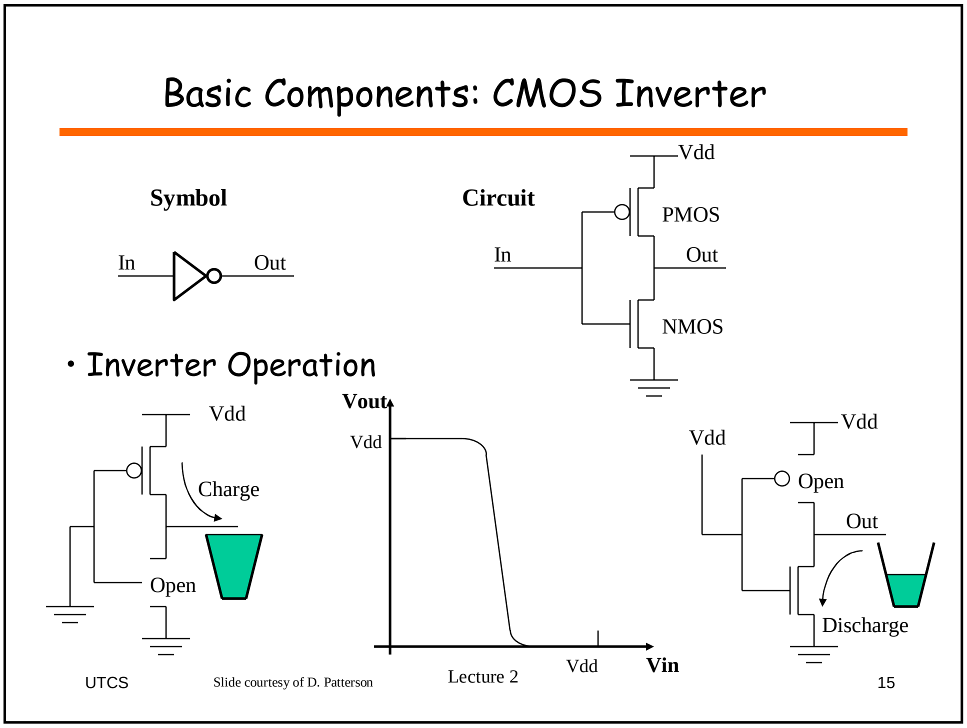
voltage
V_{dd}
V_{hi}
V_{lo}
0
"1"
???
"0"



Basic Technology: CMOS

- CMOS: Complementary Metal Oxide Semiconductor
  - NMOS (N-Type Metal Oxide Semiconductor) transistors
  - PMOS (P-Type Metal Oxide Semiconductor) transistors
- NMOS Transistor
  - Apply a HIGH (Vdd) to its gate turns the transistor into a "conductor"
  - Apply a LOW (GND) to its gate shuts off the conduction path
- PMOS Transistor
  - Apply a HIGH (Vdd) to its gate shuts off the conduction path
  - Apply a LOW (GND) to its gate turns the transistor into a "conductor"

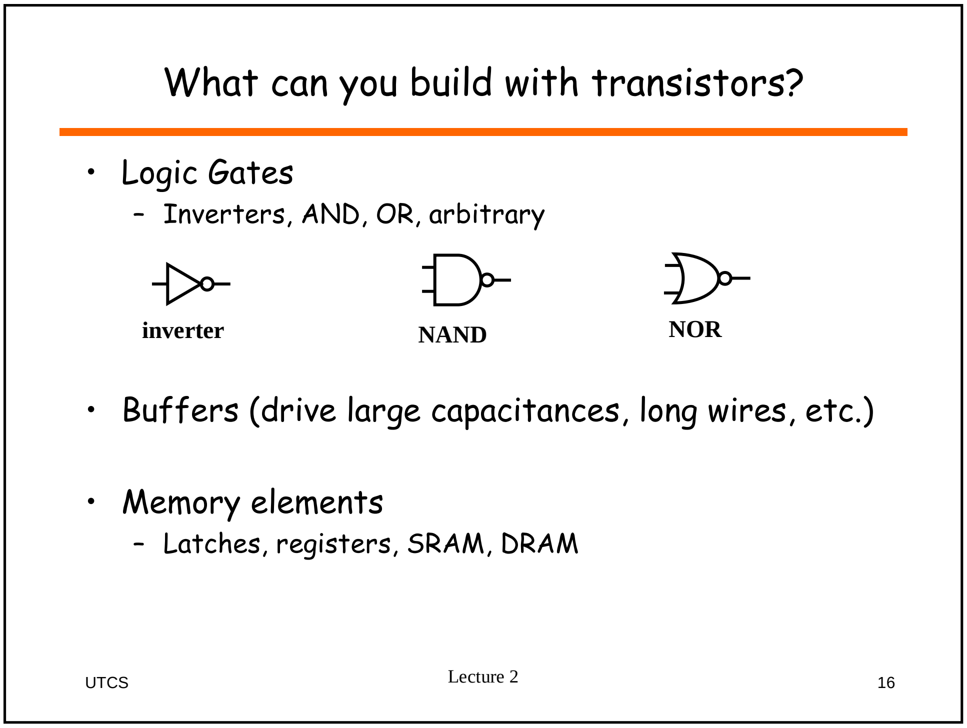
Slide courtesy of D. Patterson



Basic Components: CMOS Inverter

- Inverter Operation

UTCS Slide courtesy of D. Patterson Lecture 2 15



What can you build with transistors?

- Logic Gates
  - Inverters, AND, OR, arbitrary

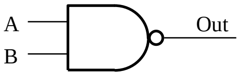
inverter
NAND
NOR

- Buffers (drive large capacitances, long wires, etc.)
- Memory elements
  - Latches, registers, SRAM, DRAM

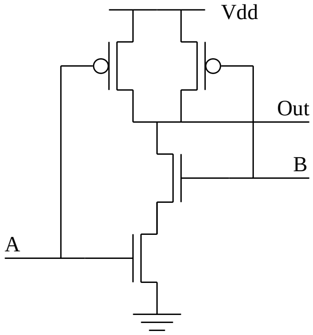
UTCS Lecture 2 16

Basic Components: CMOS Logic Gates

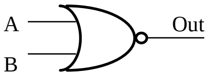
NAND Gate



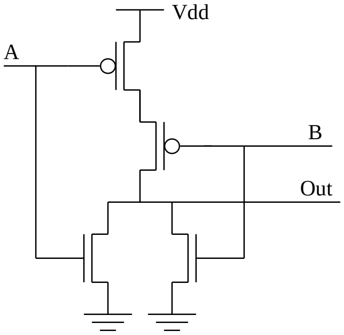
A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0



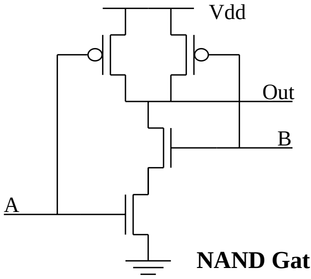
NOR Gate



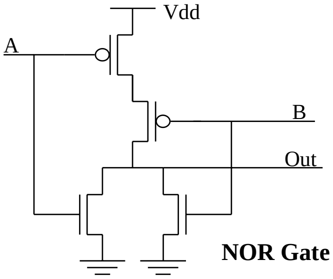
A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0



Gate Comparison



NAND Gate



NOR Gate

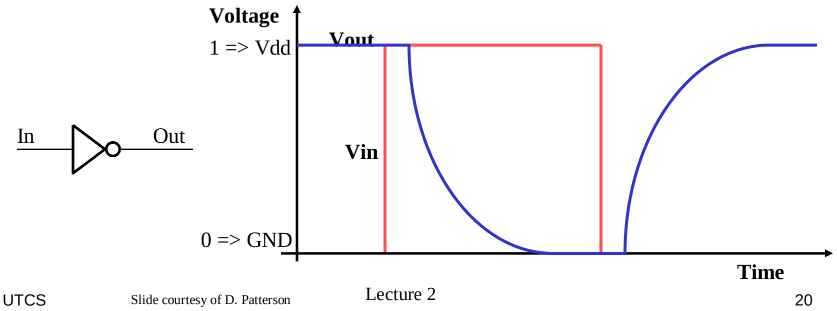
- If PMOS transistors is faster:
  - It is OK to have PMOS transistors in series
  - NOR gate is preferred
  - NOR gate is preferred also if H -> L is more critical than L -> H
- If NMOS transistors is faster:
  - It is OK to have NMOS transistors in series
  - NAND gate is preferred
  - NAND gate is preferred also if L -> H is more critical than H -> L

The Ugly Truth

- Transistors are not ideal switches!
  - Gate Capacitance (C_g)
  - Source-to-Drain resistance (R)
  - Drain capacitance
- Issues
  - Delay - actually takes real time to turn transistors on and off
  - Power/Energy
  - Noise (from transistors, power rails)
- But - we can change transistor size
  - Increase C_g, but decrease R

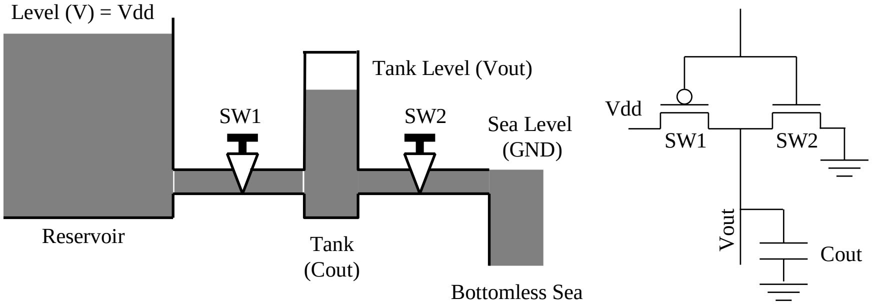
Ideal (CS) versus Reality (EE)

- When input 0 -> 1, output 1 -> 0 but NOT instantly
  - Output goes 1 -> 0: output voltage goes from Vdd (2.5v) to 0v
- When input 1 -> 0, output 0 -> 1 but NOT instantly
  - Output goes 0 -> 1: output voltage goes from 0v to Vdd (2.5v)
- Voltage does not like to change instantaneously



Slide courtesy of D. Patterson

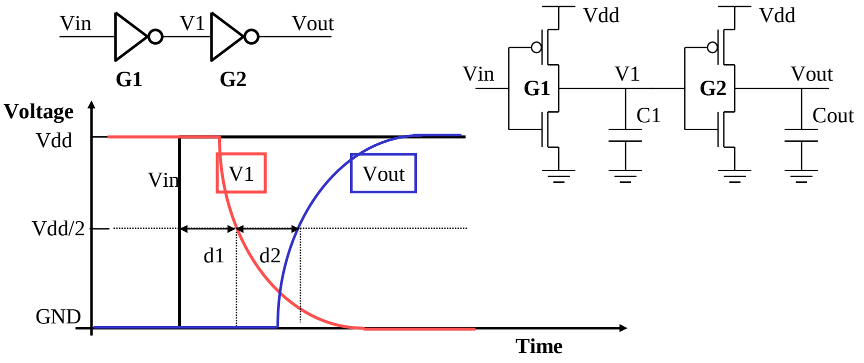
Fluid Timing Model



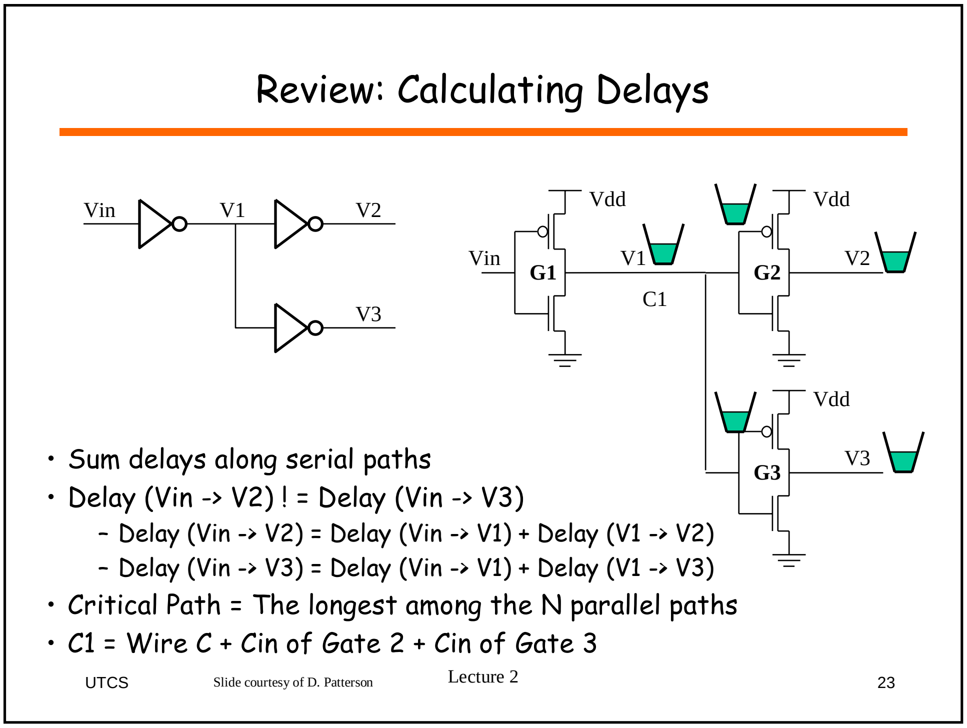
- Water <-> Electrical Charge
- Tank Capacity <-> Capacitance (C)
- Water Level <-> Voltage
- Water Flow <-> Charge Flowing (Current)
- Size of Pipes <-> Strength of Transistors (G)
- Time to fill up the tank ~ C / G

Resistance R = 1/G

Series Connection

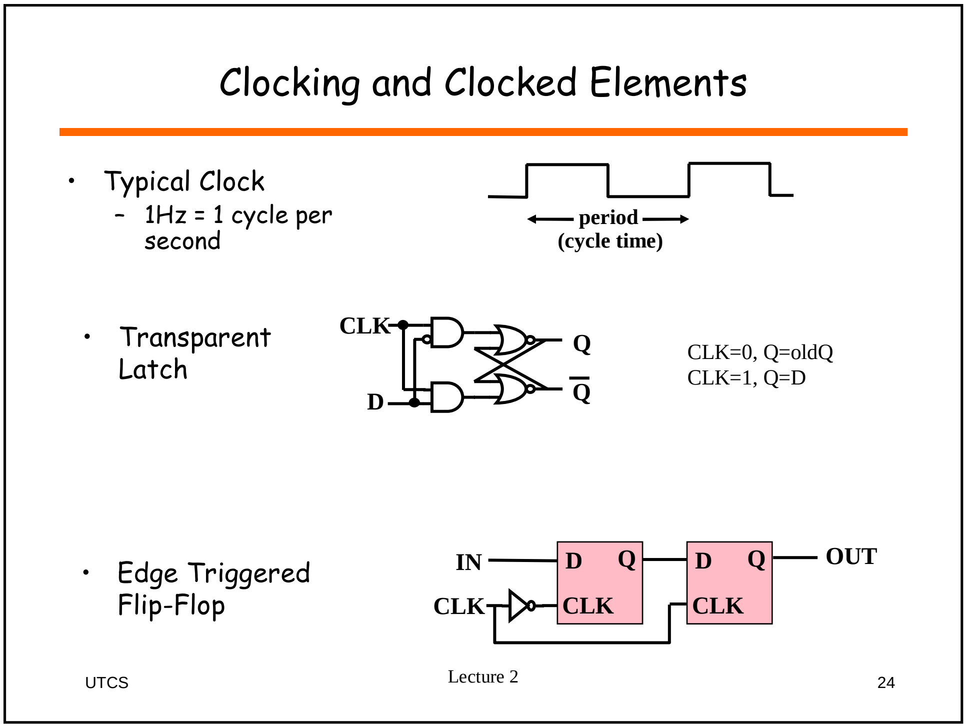


- Total Propagation Delay = Sum of individual delays = d1 + d2
- Capacitance C1 has two components:
  - Capacitance of the wire connecting the two gates
  - Input capacitance of the second inverter



Review: Calculating Delays

- Sum delays along serial paths
- Delay (Vin -> V2) ! = Delay (Vin -> V3)
  - Delay (Vin -> V2) = Delay (Vin -> V1) + Delay (V1 -> V2)
  - Delay (Vin -> V3) = Delay (Vin -> V1) + Delay (V1 -> V3)
- Critical Path = The longest among the N parallel paths
- C1 = Wire C + Cin of Gate 2 + Cin of Gate 3



Clocking and Clocked Elements

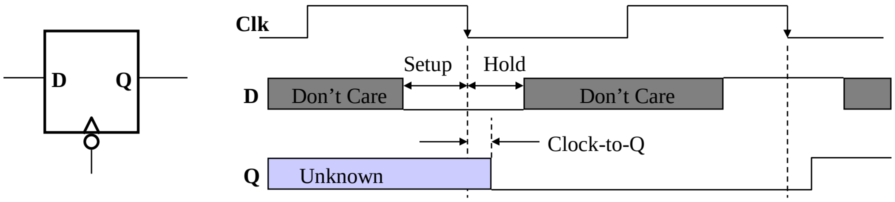
- Typical Clock
  - 1Hz = 1 cycle per second

- Transparent Latch

CLK=0, Q=oldQ
CLK=1, Q=D

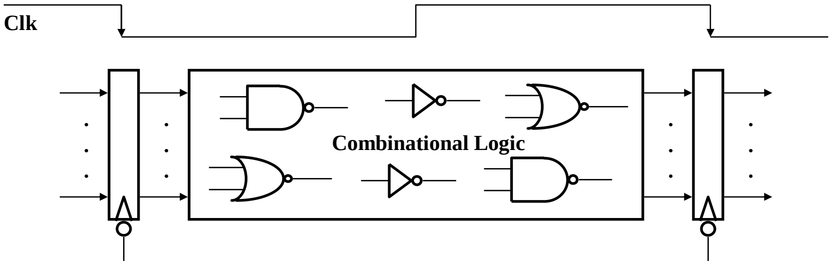
- Edge Triggered Flip-Flop

Storage Element's Timing Model



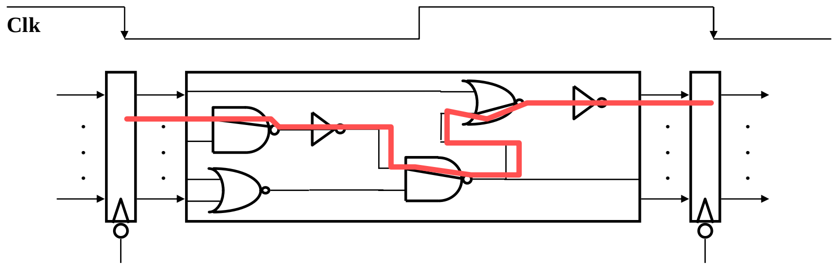
- Setup Time: Input must be stable BEFORE the trigger clock edge
- Hold Time: Input must REMAIN stable after the trigger clock edge
- Clock-to-Q time:
  - Output cannot change instantaneously at the trigger clock edge
  - Similar to delay in logic gates, two components:
    - Internal Clock-to-Q
    - Load dependent Clock-to-Q

Clocking Methodology



- All storage elements are clocked by the same clock edge
- The combination logic block's:
  - Inputs are updated at each clock tick
  - All outputs MUST be stable before the next clock tick

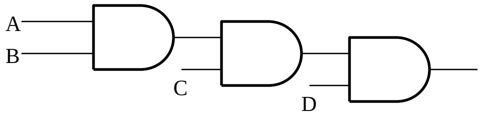
Critical Path & Cycle Time

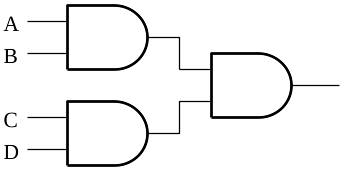


- Critical path: the slowest path between any two storage devices
- Cycle time is a function of the critical path
- must be greater than:
  - Clock-to-Q + Longest Path through the Combination Logic + Setup

Tricks to Reduce Cycle Time

- Reduce the number of gate levels



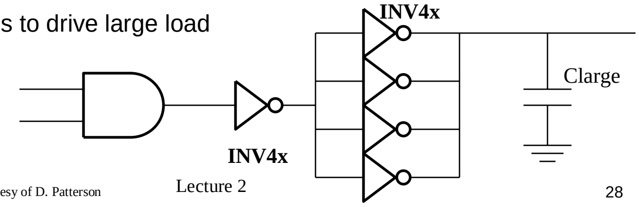


° Pay attention to loading

  ° One gate driving many gates is a bad idea

  ° Avoid using a small gate to drive a long wire

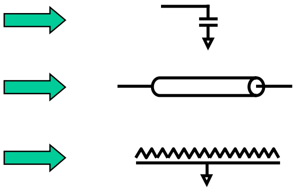
° Use multiple stages to drive large load



Wires

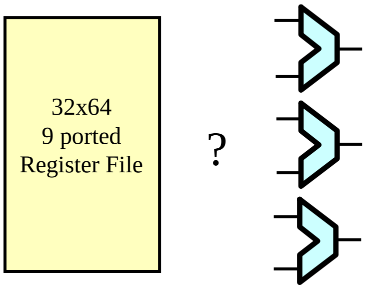
- Limiting Factor
  - Density
  - Speed
  - Power

- 3 models for wires (model to use depends on switching frequency)
  - Short
  - Lossless
  - Lossy

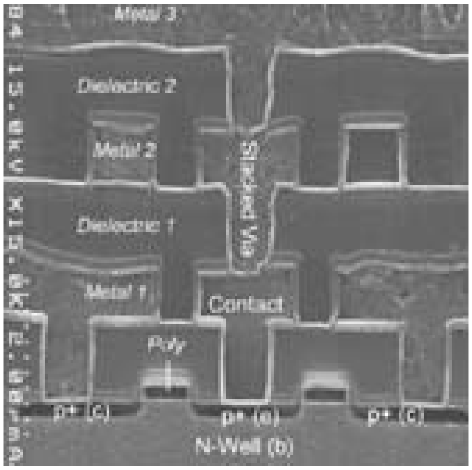


Wire Density

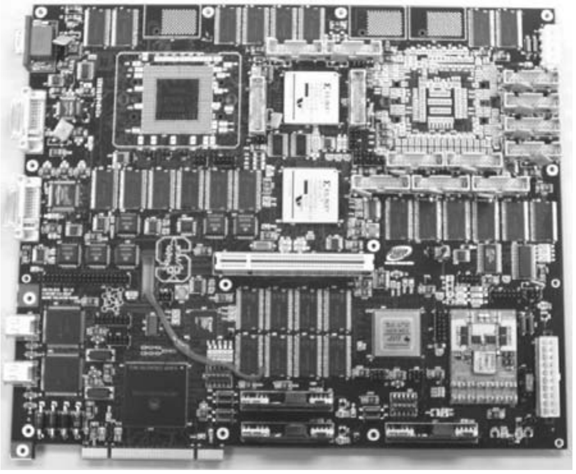
- Communication constraints
  - Must be able to move bits to/from storage and computation elements
- Example: 9 ported register file



Chip Level



Board Level



Stanford Imagine Board

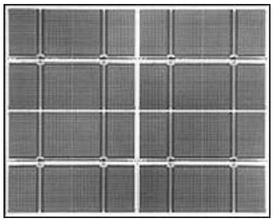
Rack Level

MIT J-Machine

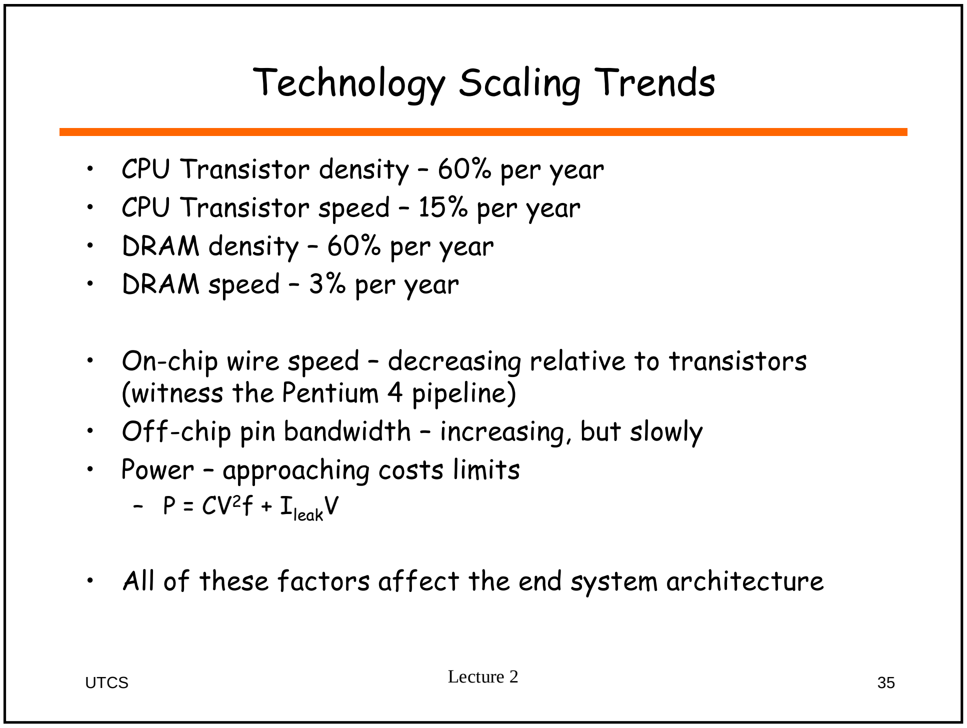
DOE ASCI White

Memory

- Moves information in time (wires move it in space)
- Provides state
- Requires energy to change state
  - Feedback circuit - SRAM
  - Capacitors – DRAM
  - Magnetic media - disk
- Required for memories
  - Storage medium
  - Write mechanism
  - Read mechanism



4Gb DRAM Die

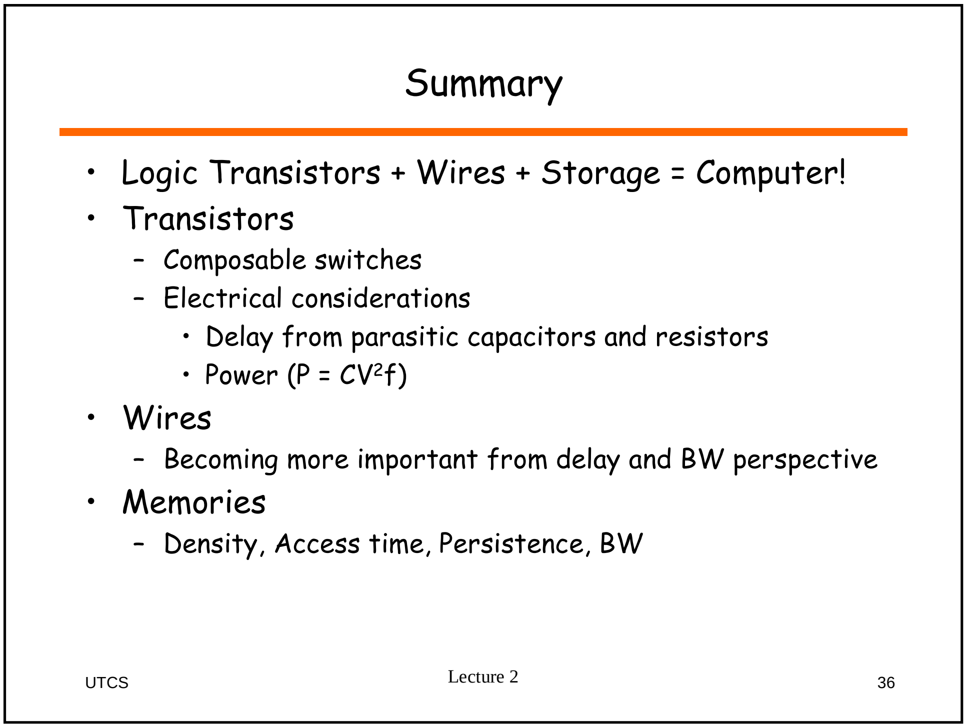


Technology Scaling Trends

- CPU Transistor density – 60% per year
- CPU Transistor speed – 15% per year
- DRAM density – 60% per year
- DRAM speed – 3% per year

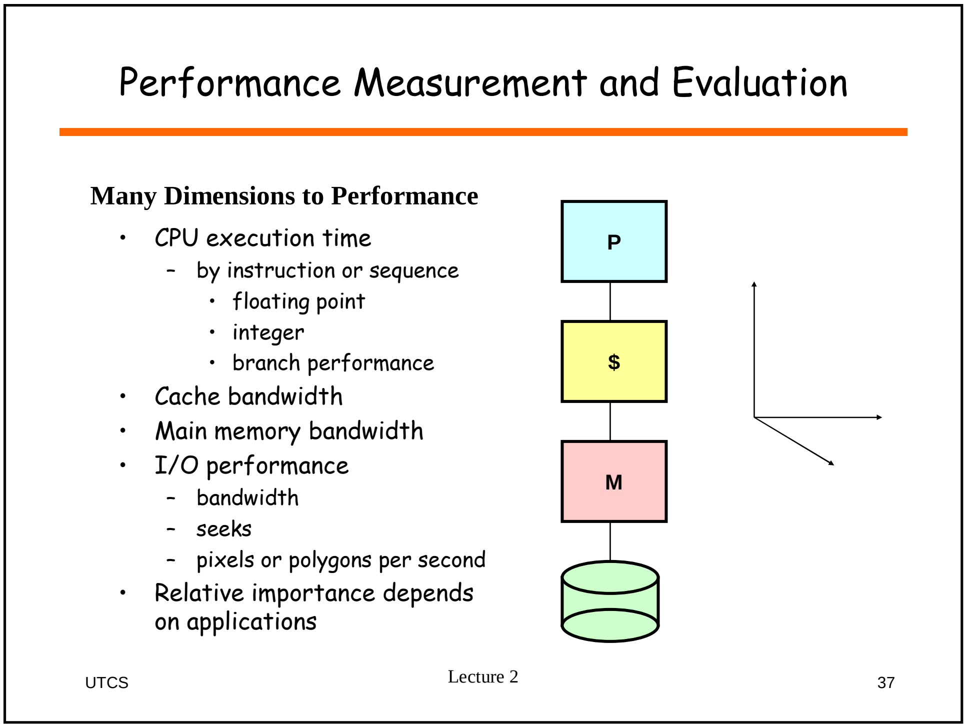
- On-chip wire speed – decreasing relative to transistors (witness the Pentium 4 pipeline)
- Off-chip pin bandwidth – increasing, but slowly
- Power – approaching costs limits
  - $P = CV^2f + I_{leak}V$

- All of these factors affect the end system architecture



Summary

- Logic Transistors + Wires + Storage = Computer!
- Transistors
  - Composable switches
  - Electrical considerations
    - Delay from parasitic capacitors and resistors
    - Power ($P = CV^2f$)
- Wires
  - Becoming more important from delay and BW perspective
- Memories
  - Density, Access time, Persistence, BW

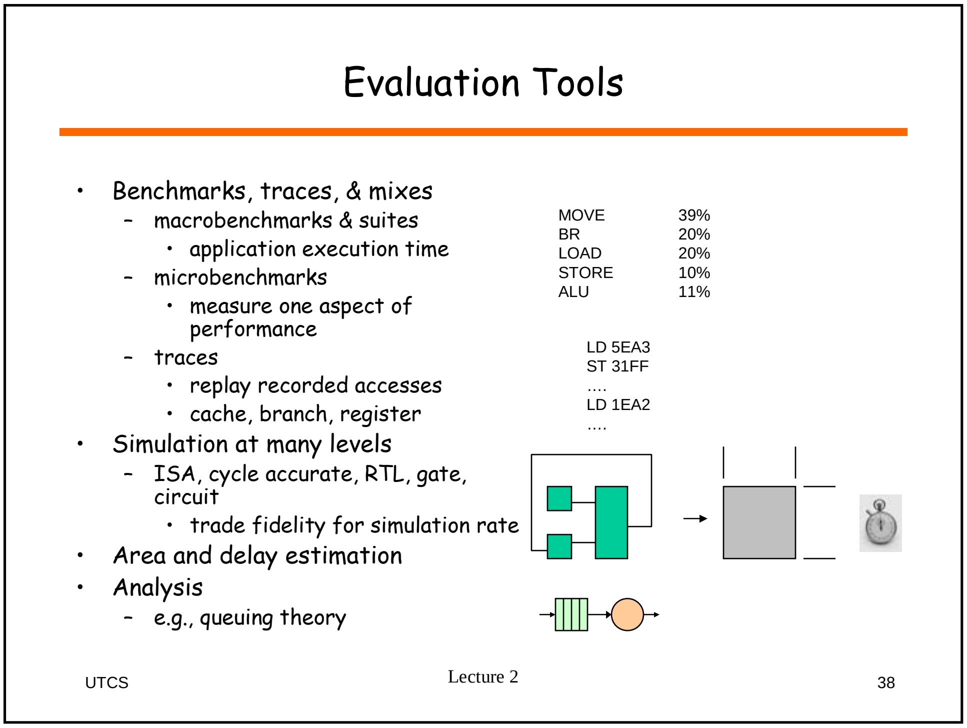


Performance Measurement and Evaluation

Many Dimensions to Performance

- CPU execution time
  - by instruction or sequence
    - floating point
    - integer
    - branch performance
- Cache bandwidth
- Main memory bandwidth
- I/O performance
  - bandwidth
  - seeks
  - pixels or polygons per second
- Relative importance depends on applications

UTCS Lecture 2 37



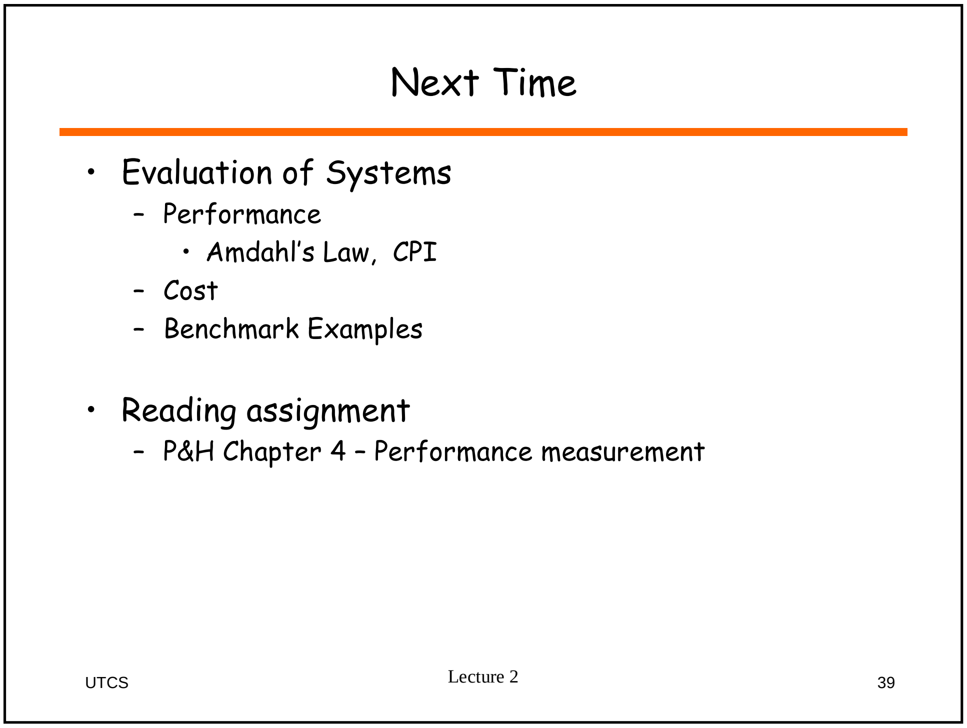
Evaluation Tools

- Benchmarks, traces, & mixes
  - macrobenchmarks & suites
    - application execution time
  - microbenchmarks
    - measure one aspect of performance
  - traces
    - replay recorded accesses
    - cache, branch, register
- Simulation at many levels
  - ISA, cycle accurate, RTL, gate, circuit
    - trade fidelity for simulation rate
- Area and delay estimation
- Analysis
  - e.g., queuing theory

MOVE	39%
BR	20%
LOAD	20%
STORE	10%
ALU	11%

```
LD 5EA3
ST 31FF
....
LD 1EA2
....
```

UTCS Lecture 2 38



Next Time

- Evaluation of Systems
  - Performance
    - Amdahl's Law, CPI
  - Cost
  - Benchmark Examples

- Reading assignment
  - P&H Chapter 4 – Performance measurement