

## Computer Architecture- 01:198:211 Fall- 2010

Prof Badri Nath  
Rutgers University  
badri@cs.rutgers.edu



## Acknowledgments



- Class notes partially based on
- 211 classes taught at Rutgers in prior years
- Material from textbook site
- Lots of material available on the web (via google search,wikipedia)

## Detail...



Me: Prof Badri Nath  
badri@cs.rutgers.edu  
Office Hours: Mondays 1:40 to 2:40 (Core 320)

TAs: 1. Binh Pham  
[bhpham@cs.rutgers.edu](mailto:bhpham@cs.rutgers.edu)  
Office Hours: Mondays 3:30 to 4:30 (tentative), Hill 412  
Recitation: Section 1: Monday ARC 107, 8:25 to 9:20 PM

2. Daniela Vianna  
[dvianna@cs.rutgers.edu](mailto:dvianna@cs.rutgers.edu)  
Office Hours: Tuesday 9 to 10, Hill 404  
Recitation: Section 3 Monday ARC 203, 8:25 to 9:20 PM

Rajat Roy  
[rajatroy@cs.rutgers.edu](mailto:rajatroy@cs.rutgers.edu)  
Office Hours: Friday 12-2 PM, Hill 257  
Recitation: Section 2 Wednesday ARC 205, 8:25 to 9:20 PM

Web: <https://sakai.rutgers.edu>  
<http://www.cs.rutgers.edu/~badri/cs211/>  
**(You're required to check this page regularly)**

## Text



- Books:
    - Computer Systems by Randall Bryant and David O'Hallaron ... Required text
    - Computer Organization and Design by David Patterson and John Hennessy (4<sup>th</sup> edition)
- Prerequisites: 198:112
- What this really means:
- You know at least one programming language.
  - You know something about how to write, run, and test programs.
  - Elementary knowledge of math and algorithms



## Ipad A4 Processor

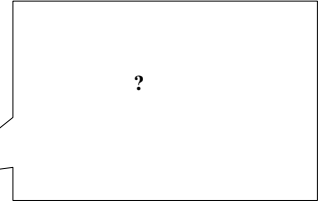


Max. CPU clock rate 1 GHz (iPad)  
 Instruction set ARM v7 32 bit RISC  
 Cores 1  
 L1 cache 64 KB  
 L2 cache 640 KB



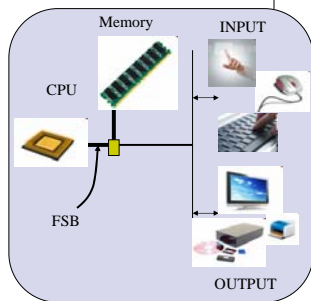
## Computer Architecture

- Architecture: art or practice of designing
- What's Inside?
- How is it put together?
- How does it work?



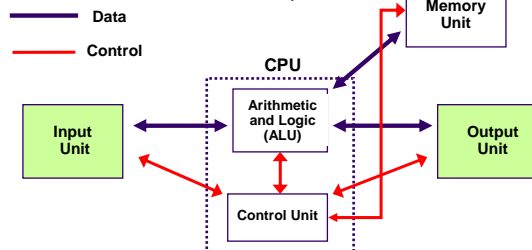
## Main Components

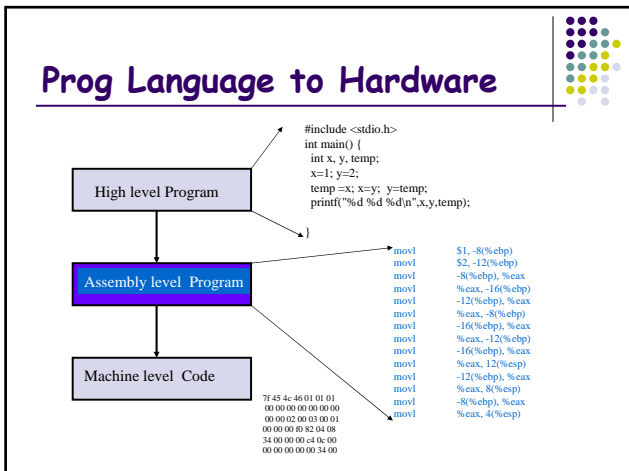
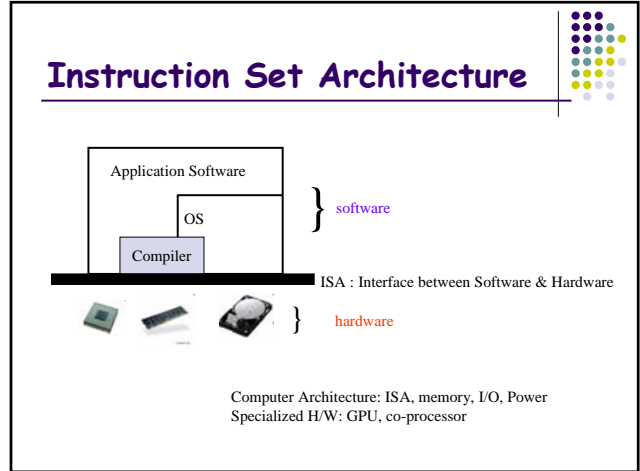
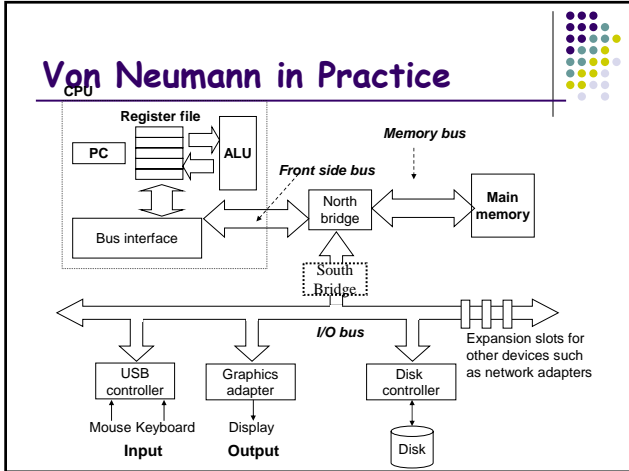
- CPU
  - Executes Instructions
- Memory
  - Stores Programs and data
- BUS
  - Transfers data
  - Front Side Bus (FSB)
- Storage
  - Permanent
- I/O devices
  - Input
    - Keypad, Mouse, Touch
  - Output
    - Printer, Screen
  - Both (input and output)
    - USB, Wifi, Touch screen,
    - DISK



## Von Neumann Architecture

- Model of a computer that used stores programs
  - Both Data and Program stored in memory
  - Allows the computer to be "Re-programmed"
- CPU is central to the computer

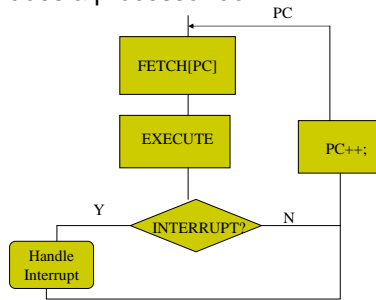




- ## What will you get out of the course?
- Basic Elements of Computer Systems
    - CPU, Memory, I/O
  - Workings of various sub systems
  - Design issues
    - Performance vs cost tradeoffs
  - Interaction of Software (programs) and the underlying hardware (on which programs execute)

## Understanding Processor/CPU performance

- What does a processor do?



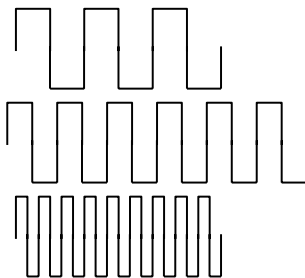
## Measuring Processor Speed

- How long does it take to execute a program
  - quicker the better
- How long does it take to go from point A to Point B
  - Speed, Distance (or)
  - Start time, End time
- Need to Know
  - Speed
  - Constant, Average, Variable
  - RPM, Circumference, distance



## CPU Clock

- Every action is driven by a clock in the CPU
- Clock time =  $1/\text{Frequency}$
- Mhz clock =  $10^{-6}$  seconds
- Ghz clock =  $10^{-9}$  seconds
- From CPU speed, you know time for 1 clock cycle



## Time for a program

- CPU executes various instructions
- A Program has several Instructions
  - How many?
    - Depends on program, compiler
- Each Instruction can take several CPU cycles
  - How many?
    - Depends on the Instruction Set Architecture (ISA)
    - ISA – Learn in this course
- Each cycle has a fixed time based on CPU , BUS speed
  - What is the clock time, memory speed etc?
    - Depends on the hardware, organization
    - Computer Architecture – Learn in this course

## CPU Performance Equation

$$\text{CPU time} = \frac{\text{Seconds}}{\text{Program}} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock Cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock Cycle}}$$

CPU time = # of Instructions × Cycles Per Instruction × Clock cycle time

How Good is the Compiler? (points to Instructions/Program)  
 Defined by the Architecture (points to Clock Cycles/Instruction)  
 How fast is the processor? (points to Seconds/Clock Cycle)

## Quantifying Computer components

- CPU Speed
  - Mhz or Ghz CPU Speed, MIPS, MFLOPS...
  - 1.33 Ghz ... Intel Atom processor
- Bus Speed
  - Front Side Bus (FSB) ... 533 Mhz Intel Atom
  - Number of channels, Number of data paths
- Memory capacity, memory speed
  - Gigabytes, Mhz x DataRate
  - 166 MHz DDR memory, Quad pump
- Disk capacity, Disk Bandwidth
  - GB, TB, MB/sec
- Power Consumption
  - Watts, mWatts,
  - Battery life time (standby vs active) Watt-Hr

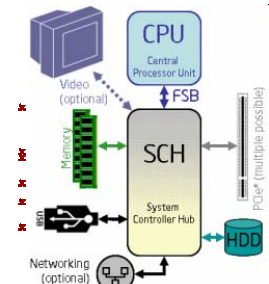
## Class of CPU

	Server	Desktop	Embedded
Cost of System	5K to 1 M	700\$ to 5K	100 to 700\$
Cost of CPU	50\$ to 1K	70 \$ to 200\$	\$ 1 to \$100
Performance metrics	Throughput Availability	Response time, price Graphics	Power, Battery life Graphics

## Intel atom

### Essentials

Status	Launched
Launch Date	Q2'08
Processor Number	Z520
# of Cores	1
Clock Speed	1.33 GHz
L2 Cache	512 KB
FSB Speed	533 MHz
FSB Parity	
Instruction Set	32-bit
Embedded	
Supplemental SKU	
Lithography	45 nm
Max TDP	2 W
VID Voltage Range	0.75-1.1V



## Other Systems

- iPhone
  - 620 Mhz ARM chip
  - SIMD, high performance integer CPU (8-stage pipeline, 675 Dhrystone, 2.1 MIPS)
  - 16 K/16 K cache
  - **0.45 mW/MHz power draw (with cache)**
- Wii
  - CPU: PowerPC-based "Broadway" processor, 729 Mhz
  - GPU: ATI "Hollywood" GPU, 243 MHz
- iPad
  - 1GHz Apple A4
  - Upto 10 hours of battery life

## Laptop Ratings

IBM ThinkPad T42 (Pentium M Processor 735 1.7GHz, 512MB RAM)

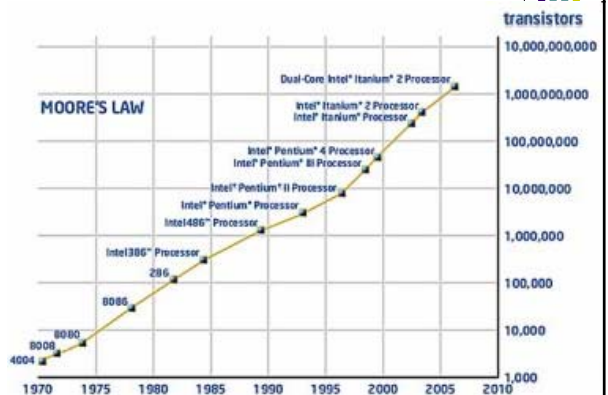


Intel® Core™ 2 Duo P8600 (2.4GHz/1066MHz FSB/3MB cache), 4G memory, 100 G disk

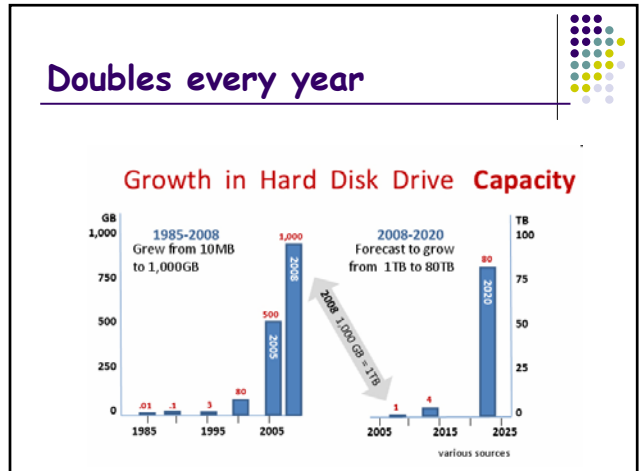
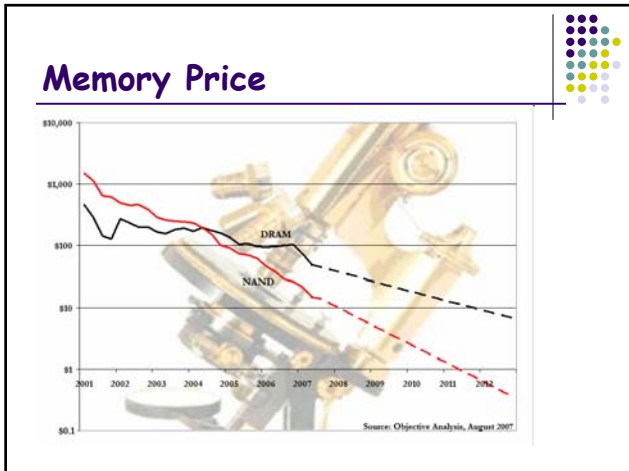
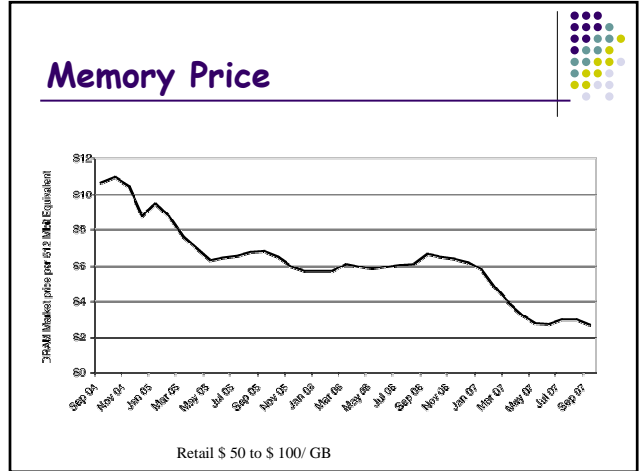
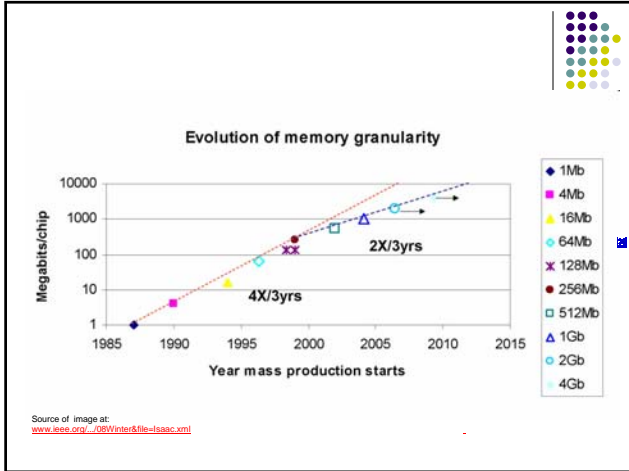
## Moore's law

- Gordon Moore was an Intel Engineer
- An observation about improvements in hardware
- No of transistors on a chip double every 18 months
- Exponential growth seen in other hardware
  - Memory capacity
    - 2x every 2 years
  - Processors Speed
    - 2x every 18 months
  - Disk capacity
    - 2x every year

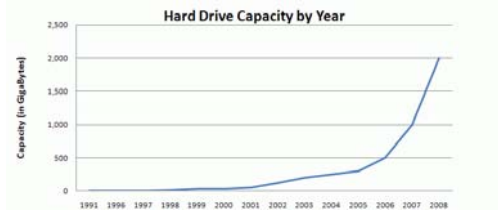
Moore's Law applied to transistors on a chip







## Disk Drive Growth



30\$ to 50 \$ per 100 G , 1 TB for 100 \$ to

## Power Ratings

- Atom Processor 1.6 Ghz , 4 W .. netbook
- AMD Athlon 1.4 Ghz, 64 W ... desktop

Model	Clock Speed	Power	Clock Speed to Power Ratio (MHz/W)
• Pentium	75 MHz	8.0 W	9.4
• Pentium	90 MHz	9.0 W	10
• Pentium	100 MHz	10.1 W	9.9
• Pentium	120 MHz	11.9 W	10.1
• Pentium	133 MHz	11.2 W	11.5
• Pentium	150 MHz	11.6 W	12.9
• Pentium	166 MHz	14.5 W	11.4
• Pentium	200 MHz	15.5 W	12.9

## Battery Lifetimes

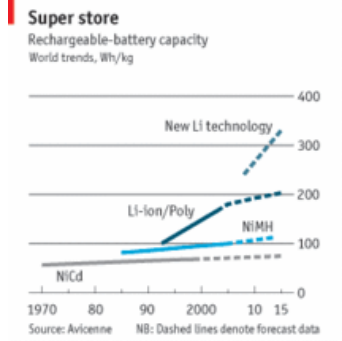
- Netbook 5 to 6 hours
- Laptop 3 to 6 hours
- iPad 8 to 10 hours
- Smart Phones
- 7 to 8 hours, Standby 24 hours

## Battery life being highlighted

Compare for yourself <sup>1</sup>		HP EliteBook 6930p Notebook PC	Dell Latitude E6400	HP EliteBook 6930p Notebook PC* <small>*Configure to order only</small>
Battery Life (up to ...)		17.25 hours	19 hours	24 hours
Starting price		\$1,199	\$879	\$1,227
Included battery		6 cell	6 cell	6 cell
Upgrade primary battery		n/a	9 cell	n/a
Add on battery		Ultra Capacity Battery	12-Cell High Capacity Slice	Ultra Capacity Battery
Display		Standard WXGA	Not Disclosed	HP Illumi-Lite LED backlit display
Storage/Hard Drive		n/a Uses standard hard drive	64GB Ultra Performance Solid State Drive	80GB Intel SSD
<b>Final Price</b>		<b>\$1,388</b>	<b>\$2,026</b>	<b>\$1,696</b>

Source: www.hp.com

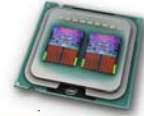
## Battery Technology



## Future

- Advanced Architectures
  - Multi-core (more than 1 CPU on a chip)
- Performance Accelerators
  - Graphic chips (Xbox, Wii, nintendo)
  - Probability Processing
- Embedded Computing
  - Processors in vending machines, washer dryers, cars
- Cloud Computing
  - Computing as a utility
- Low Energy Design
  - Green is IN

Dual Core CPU



## Course Goals

Architecture course have been traditionally taught in two way: top-down or bottom-up.

We're going in from the middle. **Programmer-Centric Approach.**

### Goals:

- Show that by knowing more about the underlying system, one can be more effective as a programmer.
  - Write programs that are more reliable and efficient.
  - Understand how programs interact with the underlying hardware.
- Learn the ins and outs of a computer's architecture.
- Learn how to do low-level programming.

## Course expectations...

### What to expect from the course:

- Will cover key issues and concepts in class.
- Recitations will provide review and teach you the tools you need.
- 4 Programming Projects (Don't freak out... yet)
- 2 mid-terms and a final
- (maybe) practice homework and quizzes

### So what do I expect of you:

- Come to class
- Read the book (Listening to me is not good enough)
- Work through the problems in the book (not really homework... but it helps)
- Do the projects
- Ask questions (**IMPORTANT**)

## Outline of Topics



Topics: Chapter Numbers from text

Intro **Chapter 1**  
C programming  
Information Representation Chapter 2  
Assembly Language Programming (x86) Chapter 3  
Digital Logic Chapter 4.2  
Processor Architecture Chapter 4.3  
Pipelining Chapters 4.4, 4.5  
Memory Hierarchy Chapter 6  
Virtual Memory Chapter 10  
I/O and System buses Chapter 11

## Project Information



### Projects Summary:

There will be 4 projects (not as bad as it sounds).  
Most projects will require some degree of programming.  
The high-level programming language of choice is C. No Java.  
Projects are very "Do-able" if you don't wait until the last minute to start them. If you do... well that's your fault.  
That being said, I will **NOT** accept any late projects. No excuses.  
Projects posted on class web-page.

### Cheating (the sad reality):

I know it happens.  
I don't like it, I won't tolerate it, and I will be looking for it .  
If you get caught, both parties will be punished.