inst.eecs.berkeley.edu/~cs61c

CS61C : Machine Structures

Lecture #1 – Number Representation

2005-08-29

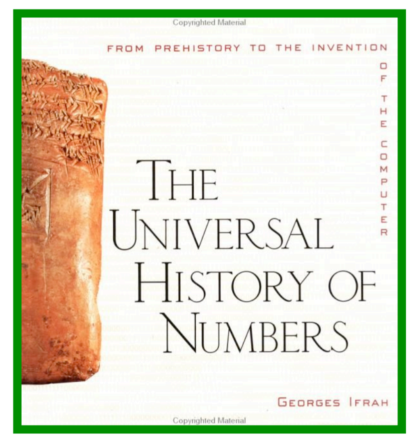
Lecturer PSOE, new dad Dan Garcia

www.cs.berkeley.edu/~ddgarcia

Great book ⇒

The Universal History of Numbers

by Georges Ifrah

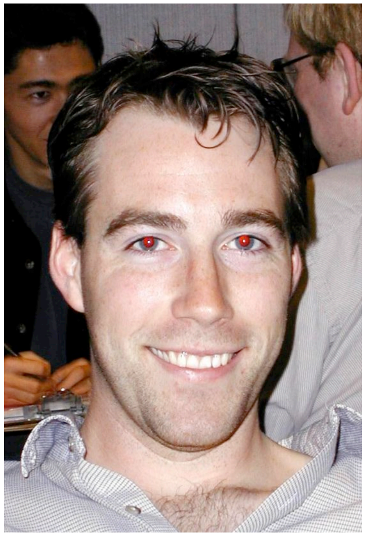


"I stand on the shoulders of giants…"

Prof
David
Patterson

Prof
John
Wawrznek

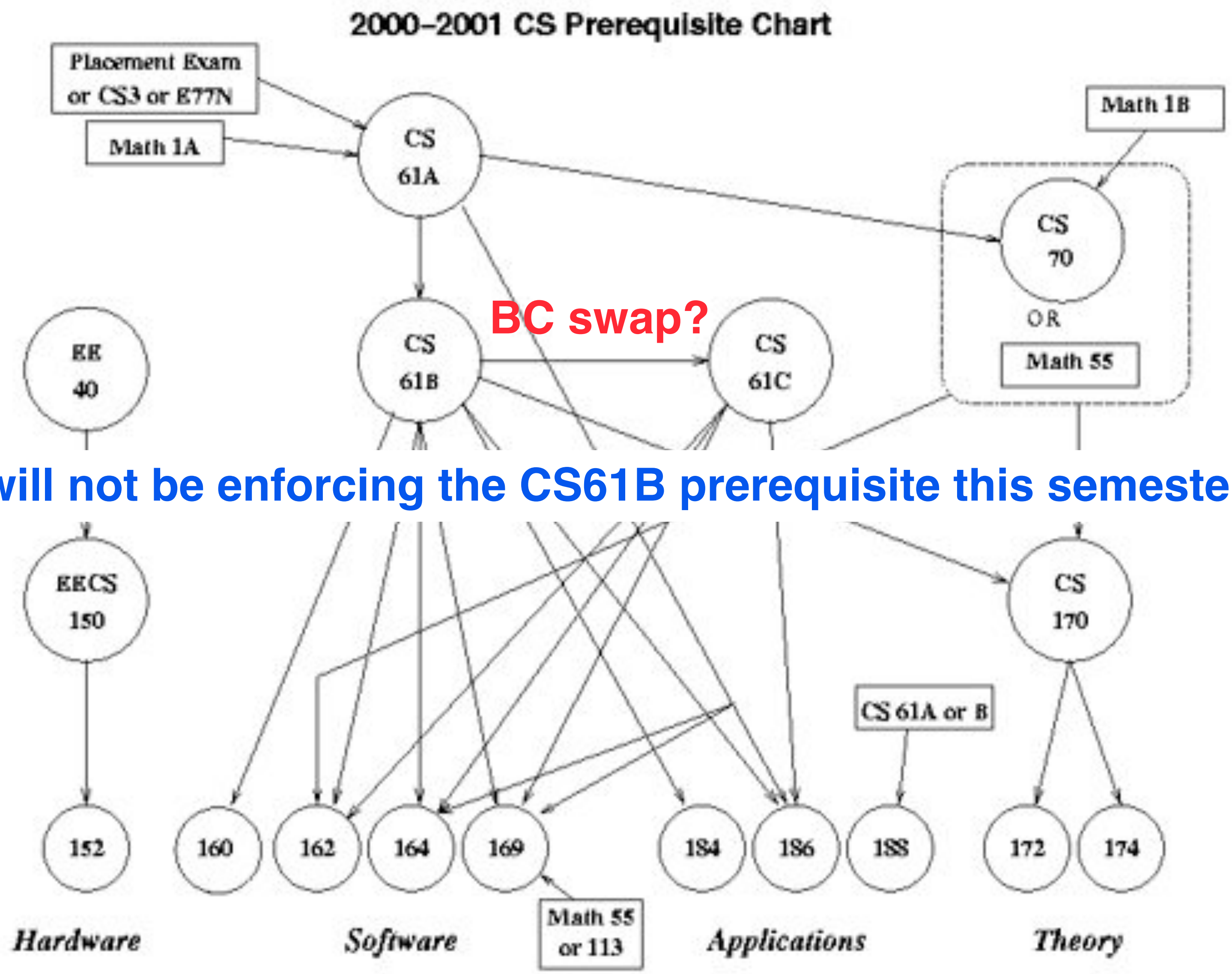
TA
Andy
Carle



TA
Kurt
Meinz

Thanks to these talented folks (& many others) whose contributions have helped make 61C a really tremendous course!

Where does CS61C fit in?



We will not be enforcing the CS61B prerequisite this semester.

http://hkn.eecs.berkeley.edu/student/cs-prereq-chart1.gif

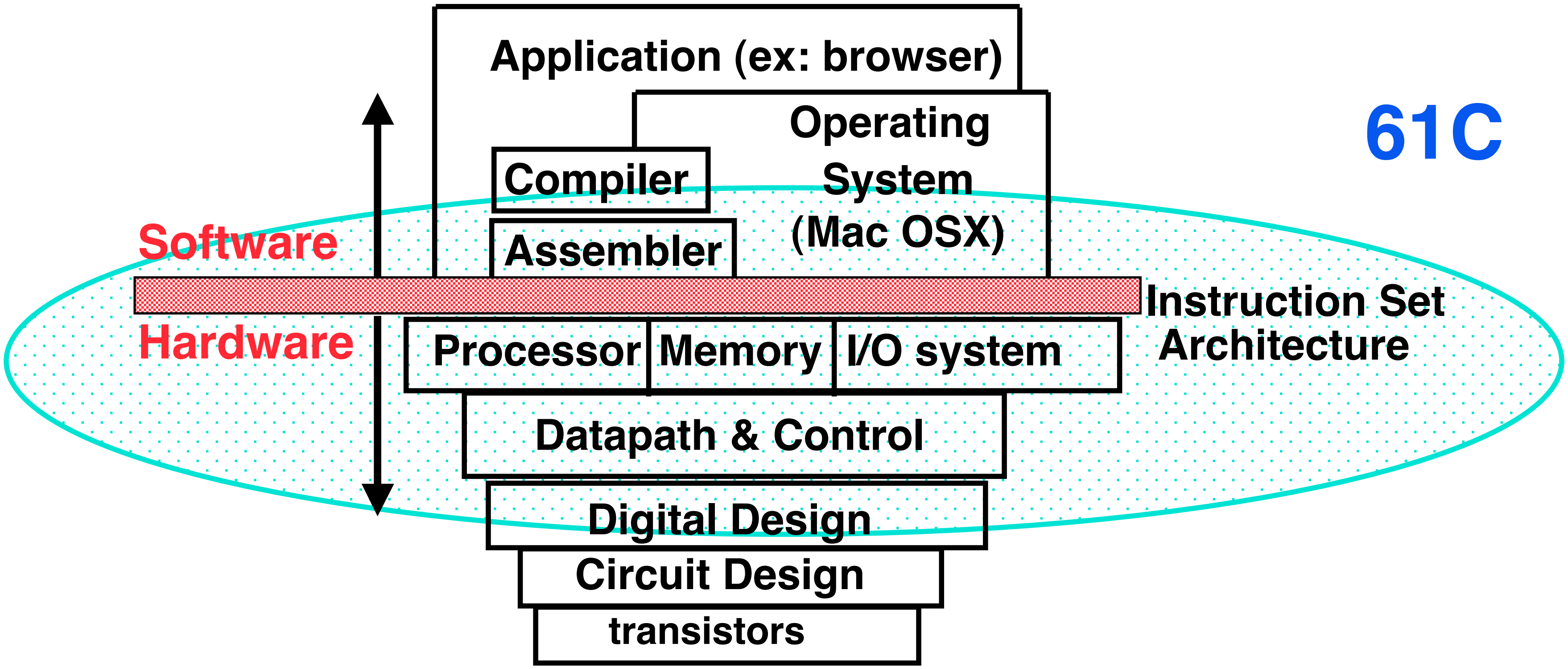
Are Computers Smart?

- To a programmer:
  - Very complex operations / functions:
    - `(map (lambda (x) (* x x)) '(1 2 3 4))`
  - Automatic memory management:
    - `List l = new List;`
  - "Basic" structures:
    - Integers, floats, characters, plus, minus, print commands

Are Computers Smart?

- In real life:
  - Only a handful of operations:
    - `{and, or, not}`
  - <u>No</u> memory management.
  - Only 2 values:
    - {0, 1} or {low, high} or {off, on}

What are “Machine Structures”?



* **Coordination of many**

levels (layers) of abstraction

61C Levels of Representation

High Level Language Program (e.g., C)

```
temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;
```

Compiler

Assembly Language Program (e.g.,MIPS)

```
lw   $t0, 0($2)
lw   $t1, 4($2)
sw   $t1, 0($2)
sw   $t0, 4($2)
```

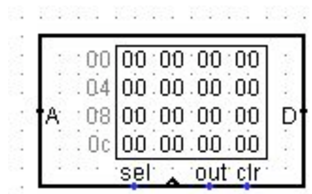
Assembler

Machine Language Program (MIPS)

```
0000 1001 1100 0110 1010 1111 0101 1000
1010 1111 0101 1000 0000 1001 1100 0110
1100 0110 1010 1111 0101 1000 0000 1001
0101 1000 0000 1001 1100 0110 1010 1111
```

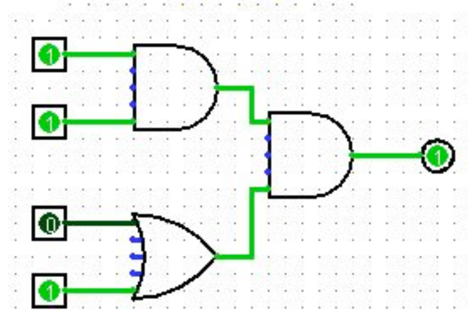
Machine Interpretation

Hardware Architecture Description (Logic, Logisim, etc.)

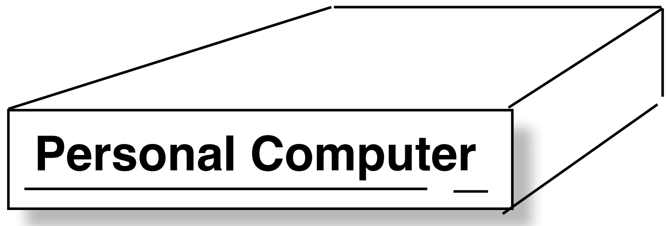


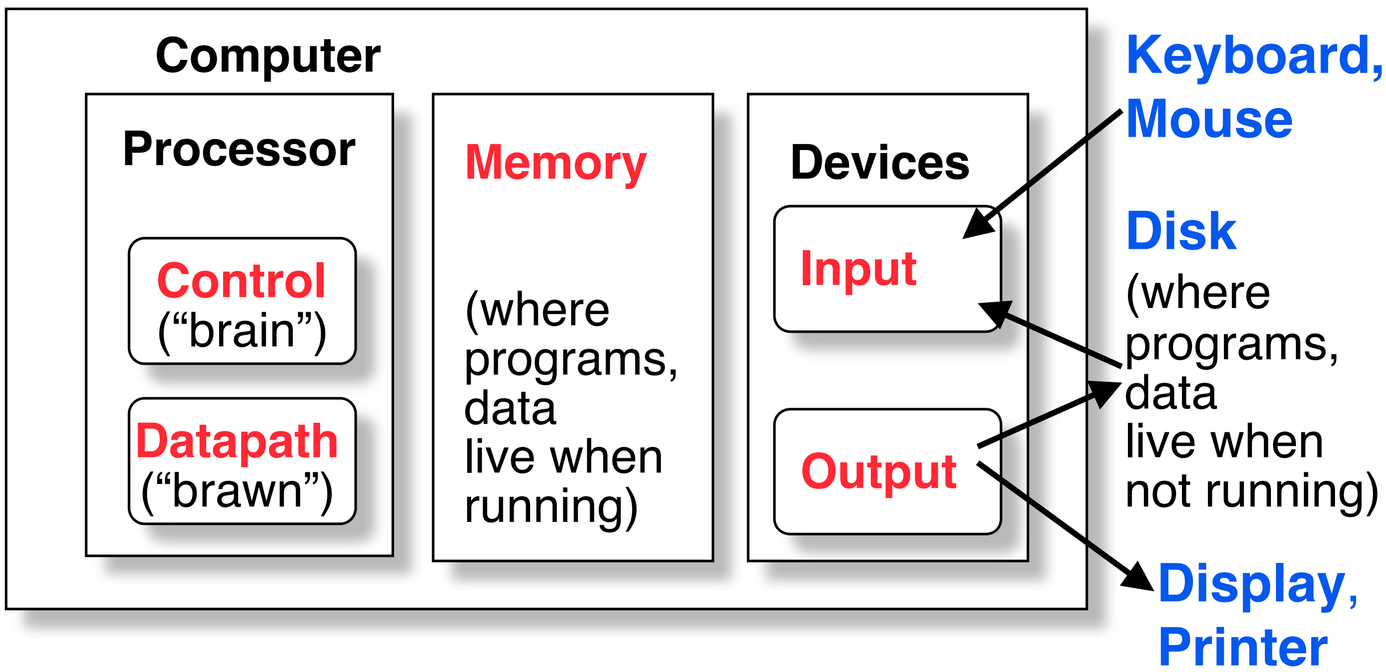
Architecture Implementation

Logic Circuit Description (Logisim, etc.)



Anatomy: 5 components of any Computer





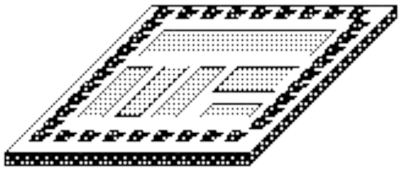
Overview of Physical Implementations

The hardware out of which we make systems.

- **Integrated Circuits (ICs)**
  - **Combinational logic circuits, memory elements, analog interfaces.**
- **Printed Circuits (PC) boards**
  - **substrate for ICs and interconnection, distribution of CLK, Vdd, and GND signals, heat dissipation.**
- **Power Supplies**
  - **Converts line AC voltage to regulated DC low voltage levels.**
- **Chassis (rack, card case, ...)**
  - **holds boards, power supply, provides physical interface to user or other systems.**
- **Connectors and Cables.**

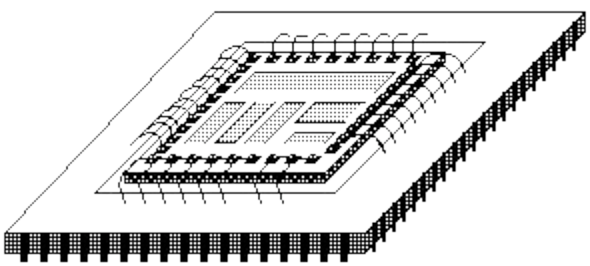
Integrated Circuits (2003 state-of-the-art)

Bare Die



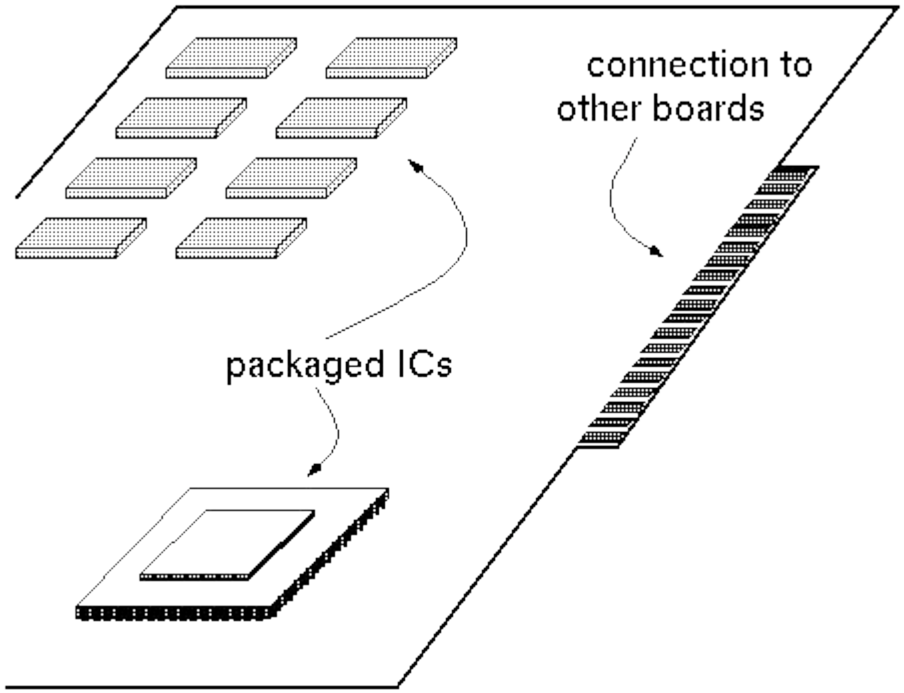
- Primarily Crystalline Silicon
- 1mm - 25mm on a side
- 2003 - feature size ~ 0.13μm = 0.13 x 10^{-6} m
- 100 - 400M transistors
- (25 - 100M “logic gates")
- 3 - 10 conductive layers
- “CMOS” (complementary metal oxide semiconductor) - most common.

Chip in Package



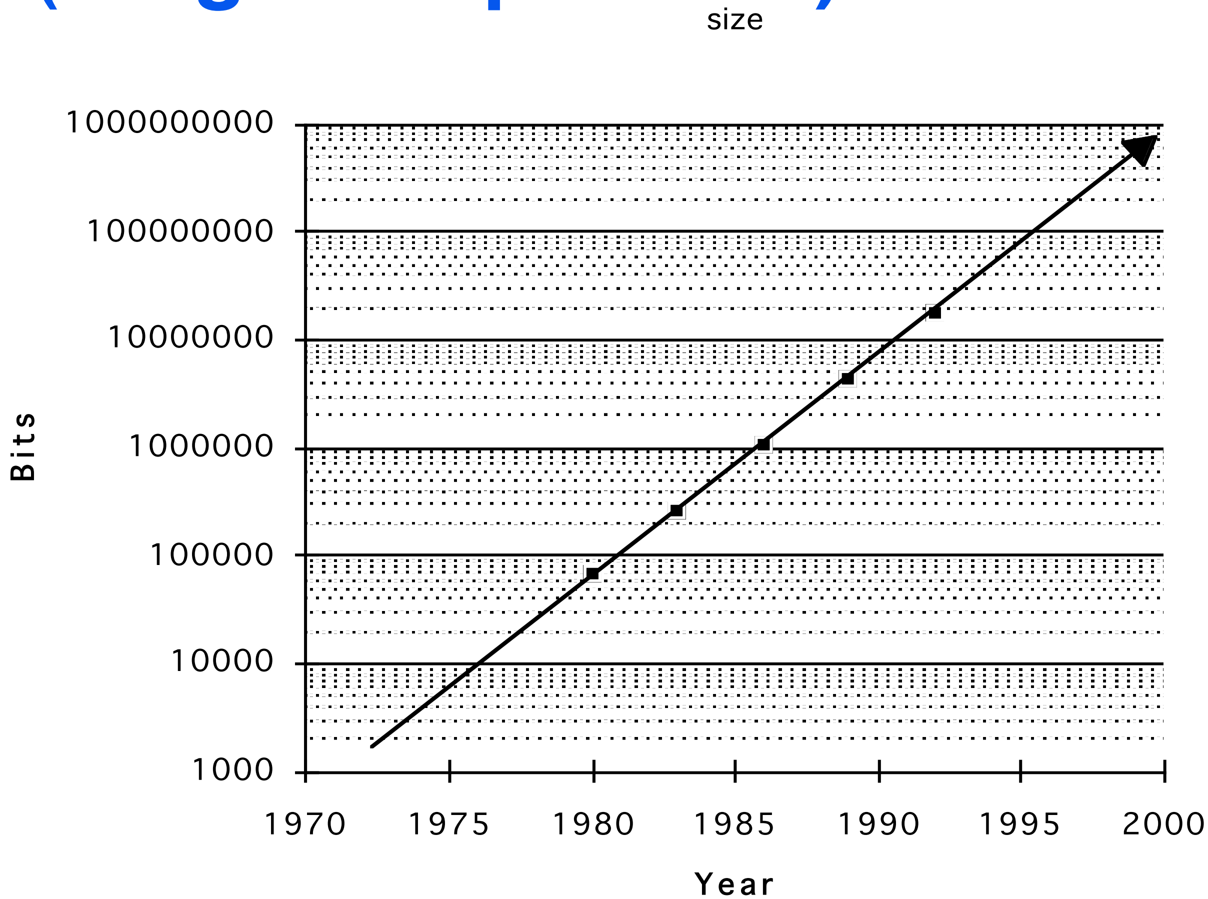
- Package provides:
  - spreading of chip-level signal paths to board-level
  - heat dissipation.
- Ceramic or plastic with gold wires.

Printed Circuit Boards



- fiberglass or ceramic
- 1-20 conductive layers
- 1-20in on a side
- IC packages are soldered down.

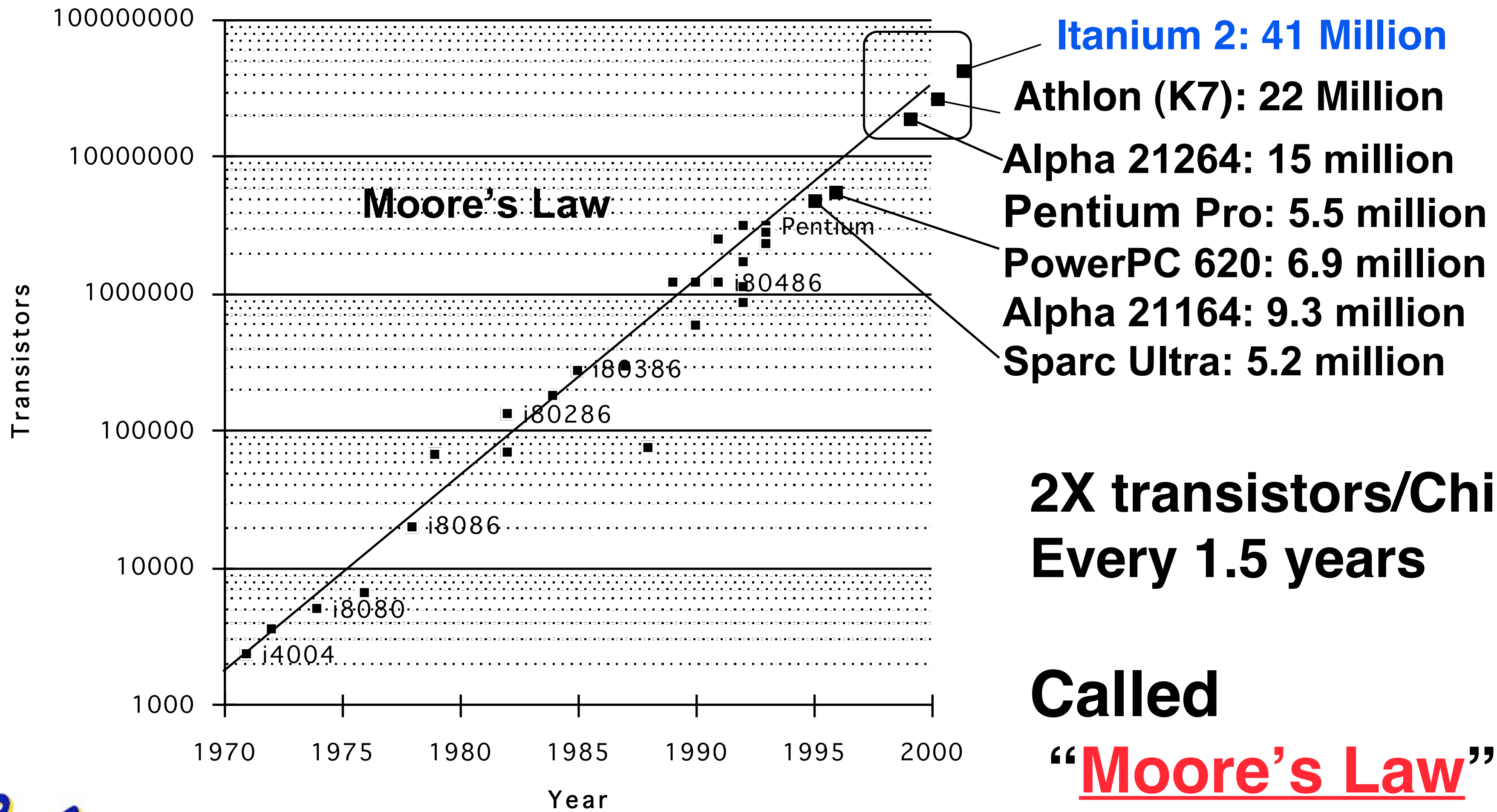
Technology Trends: Memory Capacity (Single-Chip DRAM)



year	size (Mbit)
1980	0.0625
1983	0.25
1986	1
1989	4
1992	16
1996	64
1998	128
2000	256
2002	512

- Now 1.4X/yr, or 2X every 2 years.
- 8000X since 1980!

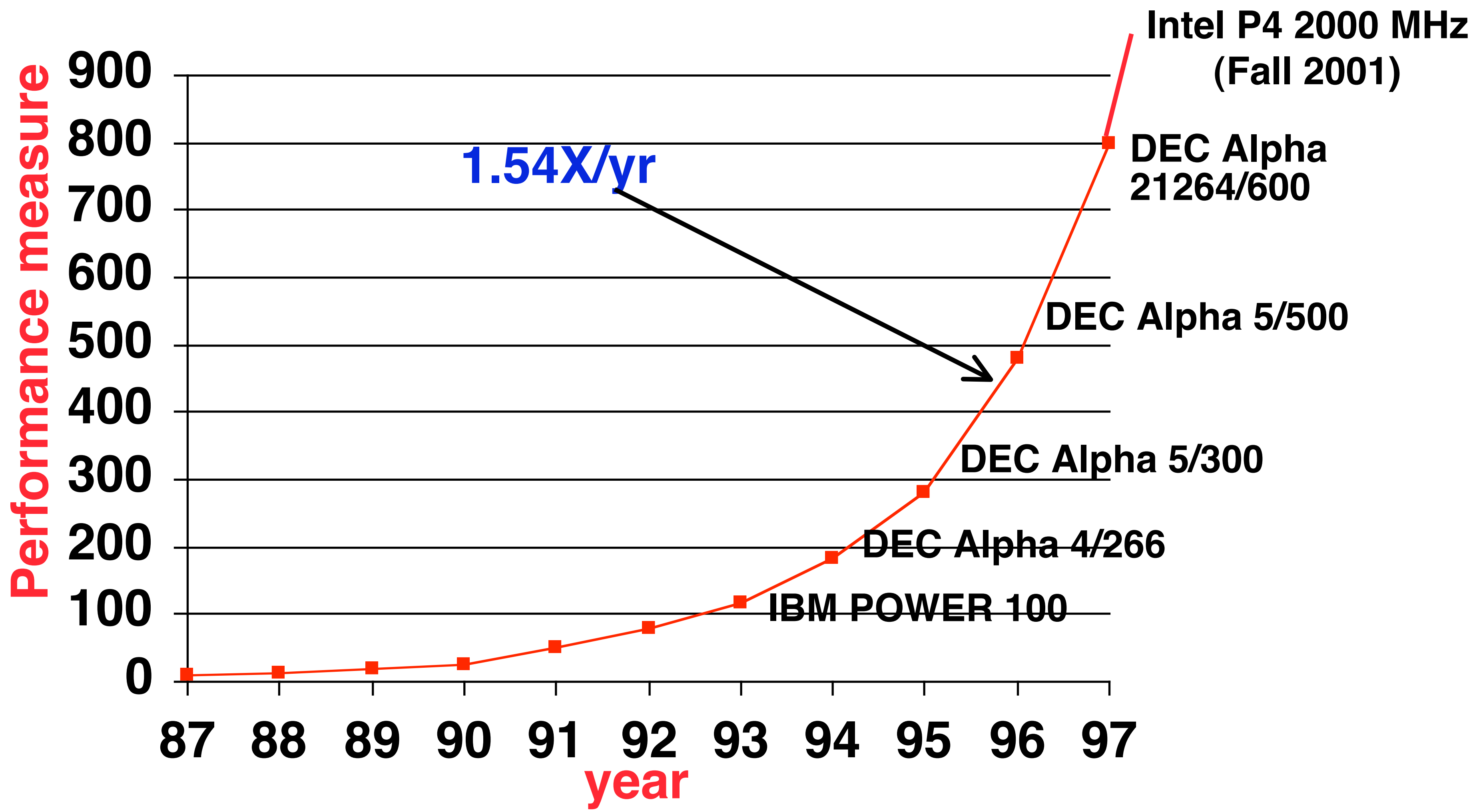
Technology Trends: Microprocessor Complexity



2X transistors/Chip Every 1.5 years

Called "Moore's Law"

Technology Trends: Processor Performance



We'll talk about processor performance later on…

Computer Technology - Dramatic Change!

- Memory
  - DRAM capacity: 2x / 2 years (since ‘96); 64x size improvement in last decade.
- Processor
  - Speed 2x / 1.5 years (since ‘85); 100X performance in last decade.
- Disk
  - Capacity: 2x / 1 year (since ‘97) 250X size in last decade.

Computer Technology - Dramatic Change!

We'll see that Kilo, Mega, etc. are incorrect later!

- State-of-the-art PC when you graduate: (at least…)
  - Processor clock speed: 5000 MegaHertz (5.0 GigaHertz)
  - Memory capacity: 8000 MegaBytes (8.0 GigaBytes)
  - Disk capacity: 2000 GigaBytes (2.0 TeraBytes)
  - New units! Mega => Giga, Giga => Tera

    (Tera => Peta, Peta => Exa, Exa => Zetta Zetta => Yotta = 10^{24})

CS61C: So what's in it for me?

- Learn some of the big ideas in CS & engineering:
  - 5 Classic components of a Computer
  - Data can be anything (integers, floating point, characters): a program determines what it is
  - Stored program concept: instructions just data
  - Principle of Locality, exploited via a memory hierarchy (cache)
  - Greater performance by exploiting parallelism
  - Principle of abstraction, used to build systems as layers
  - Compilation v. interpretation thru system layers
  - Principles/Pitfalls of Performance Measurement

Others Skills learned in 61C

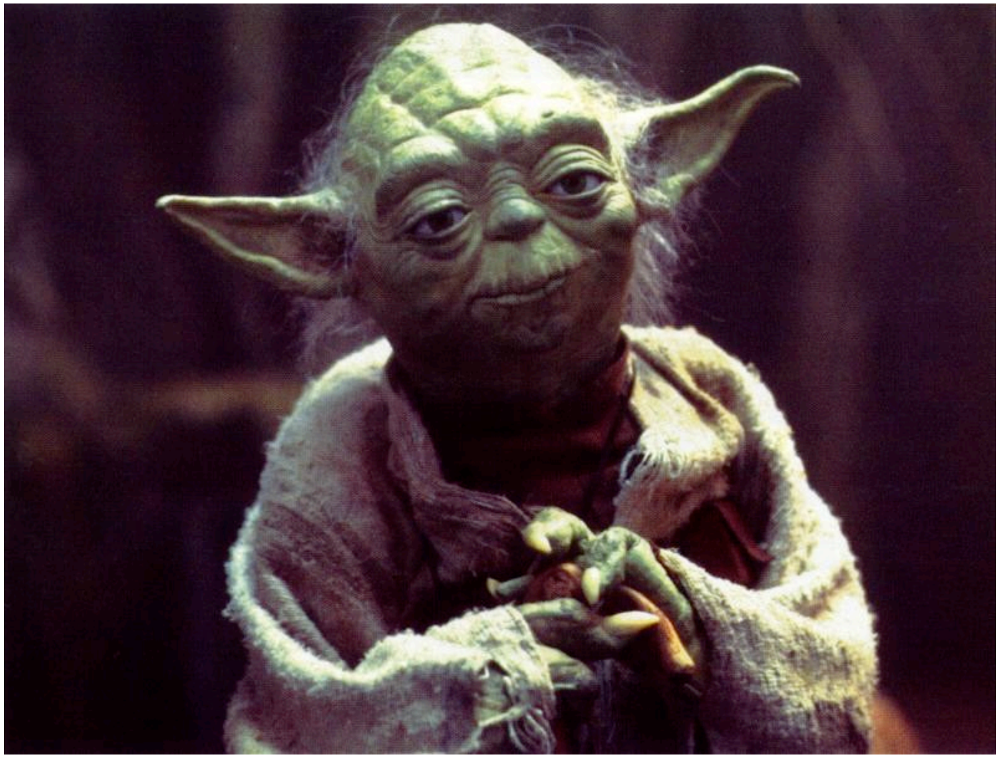
- **Learning C**
  - If you know one, you should be able to learn another programming language largely on your own
  - Given that you know C++ or Java, should be easy to pick up their ancestor, C
- **Assembly Language Programming**
  - This is a skill you will pick up, as a side effect of understanding the Big Ideas
- **Hardware design**
  - We think of hardware at the abstract level, with only a little bit of physical logic to give things perspective
  - CS 150, 152 teach this

Course Lecture Outline

- Number representations
- C-Language (basics + pointers)
- Memory management
- Assembly Programming
- Floating Point
- `make`-ing an Executable
- Logic Design
- Introduction to Logisim
- CPU organization
- Pipelining
- Caches
- Virtual Memory
- I/O
- Disks, Networks
- Performance
- Advanced Topic

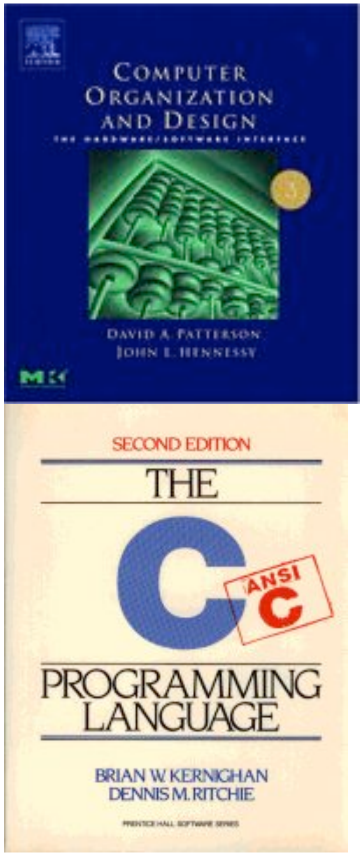
Yoda says…

“Always in motion is the future…”



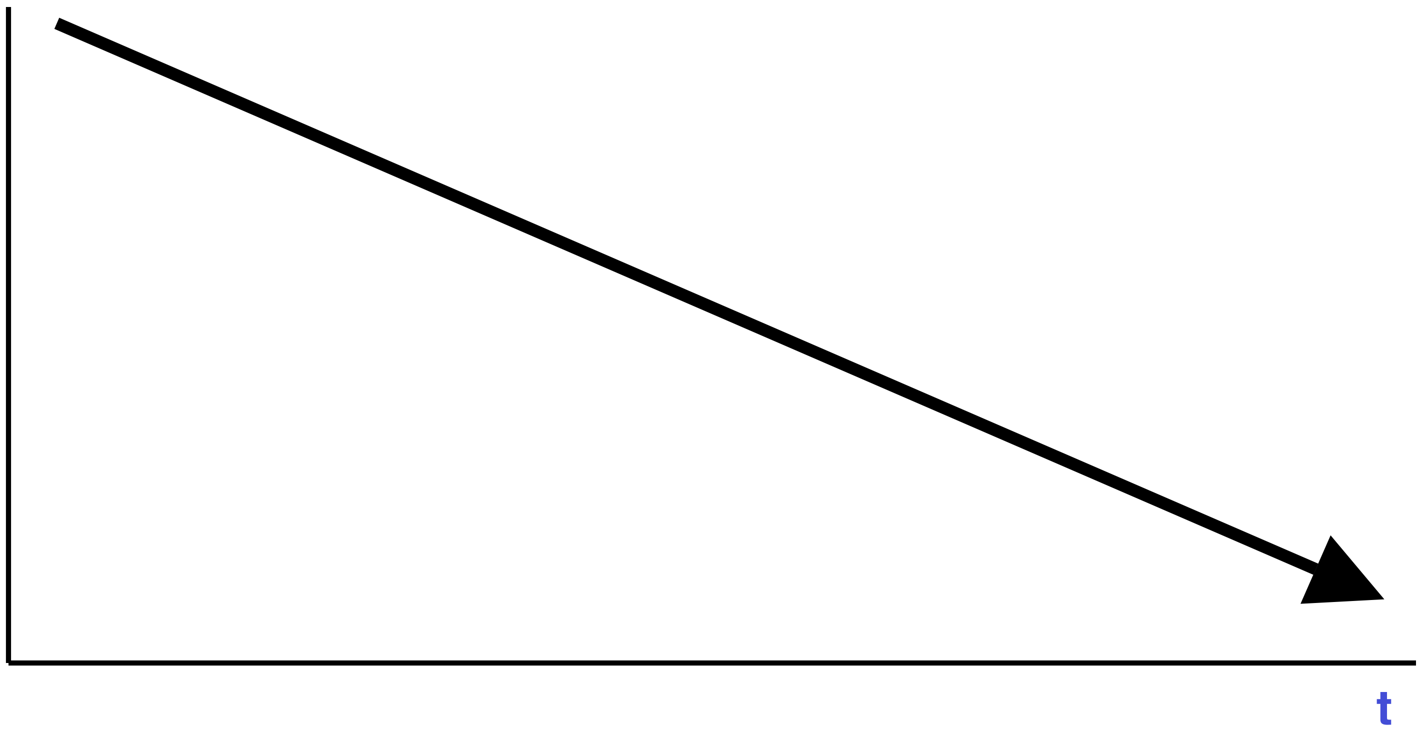
Our schedule may change slightly depending on some factors. This includes lectures, assignments & labs…

Texts



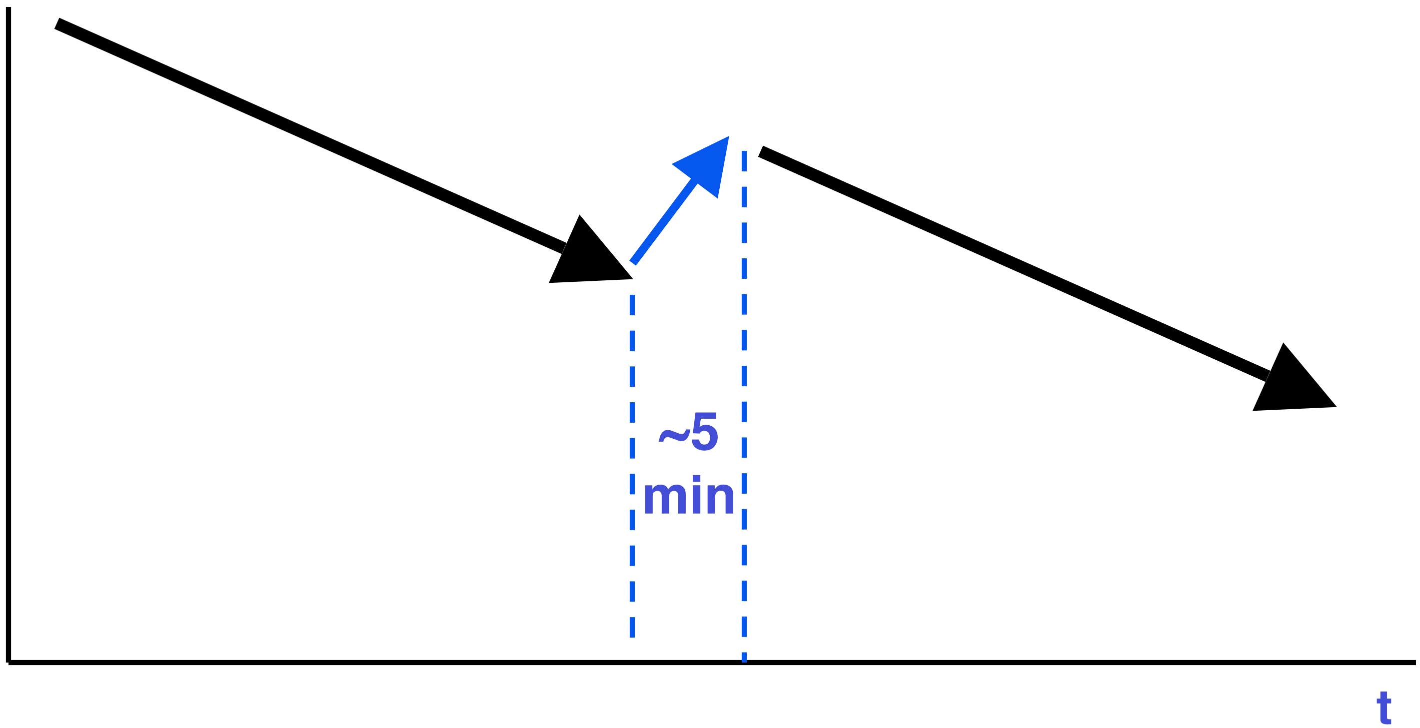
- **Required:** ***Computer Organization and Design: The Hardware/Software Interface, Third Edition*, Patterson and Hennessy (COD). *The second edition is far inferior, and is not suggested.***
- **Required: *The C Programming Language*, Kernighan and Ritchie (K&R), 2nd edition**
- **Reading assignments on web page**

What is this?



Attention over time!

What is this?!



Attention over time!

Tried-and-True Technique: Peer Instruction

- Increase real-time learning in lecture, test understanding of concepts vs. details

- As complete a "segment" ask multiple choice question
  - 1-2 minutes to decide yourself
  - 3 minutes in pairs/triples to reach consensus. Teach others!
  - 5-7 minute discussion of answers, questions, clarifications

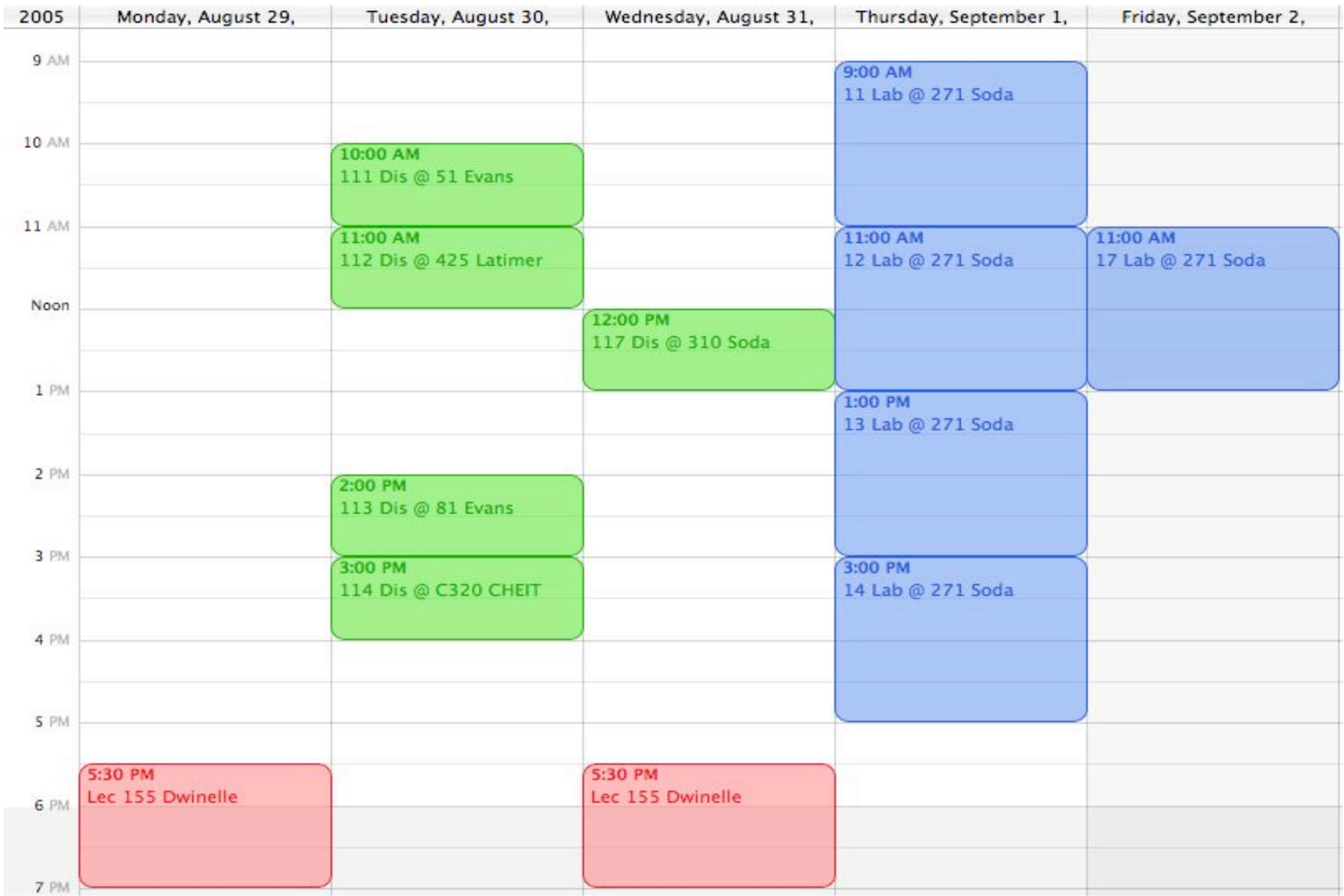
- You'll get transmitters from ASUC bookstore (or Neds) (but they're not in yet!)

Peer Instruction

- Read textbook
  - Reduces examples have to do in class
  - Get more from lecture (also good advice)
- Fill out 3-question Web Form on reading (released mondays, due every friday before lecture)
  - Graded for effort, not correctness…
  - This counts for "E"ffort in EPA score

Weekly Schedule

We are having discussion, lab and office hours this week…

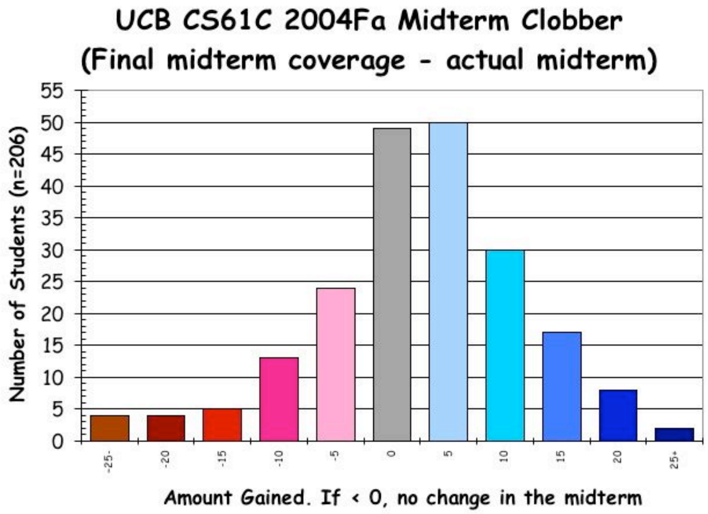


Homeworks, Labs and Projects

- **Lab exercises** (every wk; due in that lab session unless extension given by TA) – extra point if you finish in 1st hour!
- **Homework exercises** (~ every week; (HW 0) out now, due in section ***next week)***
- **Projects** (every 2 to 3 weeks)
- All exercises, reading, homeworks, projects on course web page
- We will DROP your lowest HW, Lab!
- Only one {HW, Project, Midterm} / week

2 Course Exams

- **Midterm: Monday 2005-10-17 HERE 5:30-8:30**
  - Give 3 hours for 2 hour exam (start in class)
  - One "review sheet" allowed
  - Review session Sun beforehand, time/place TBA
- **Final: Sat 2005-12-17 @ 12:30-3:30pm (grp 14)**
  - You can *clobber* your midterm grade!
  - (students last semester LOVED this…)



Your final grade

- Grading (could change before 1st midterm)
  - 15pts =   5% Labs
  - 30pts = 10% Homework
  - 60pts = 20% Projects
  - 75pts = 25% Midterm* *[can be clobbered by Final]*
  - 120pts = 40% Final
  - \+   Extra credit for EPA. What's EPA?
- Grade distributions
  - Similar to CS61B, in the absolute scale.
  - Perfect score is 300 points. 10-20-10 for A+, A, A-
  - Similar for Bs and Cs (40 pts per letter-grade)
  - … C+, C, C-, D, F (No D+ or D- distinction)
  - Differs: No F will be given if all-but-one {hw, lab}, all projects submitted and all exams taken
  - We'll "ooch" grades up but never down

Extra Credit: EPA!

- **E**ffort
  - Attending Dan's and TA's office hours, completing all assignments, turning in HW0, doing reading quizzes
- **P**articipation
  - Attending lecture and voting using the PRS system
  - Asking great questions in discussion and lecture and making it more interactive
- **A**ltruism
  - Helping others in lab or on the newsgroup
- EPA! extra credit points have the potential to bump students up to the next grade level! (but actual EPA! scores are internal)

Course Problems…Cheating

- **What is cheating?**
  - **<u>Studying</u> together in groups is <u>encouraged.</u>**
  - **Turned-in work must be <u>*completely*</u> your own.**
  - **Common examples of cheating: running out of time on a assignment and then pick up output, take homework from box and copy, person asks to borrow solution "just to take a look", copying an exam question, …**
  - **You're not allowed to work on homework/projects/exams with <u>anyone</u> (other than ask Qs walking out of lecture)**
  - **Both "giver" and "receiver" are equally culpable**
- **Cheating points: negative points for that assignment / project / exam (e.g., if it's worth 10 pts, you get -10) In most cases, F in the course.**
- **<u>Every offense</u> will be referred to the Office of Student Judicial Affairs.**

`www.eecs.berkeley.edu/Policies/acad.dis.shtml`

Student Learning Center (SLC)

- Cesar Chavez Center (on Lower Sproul)
- The SLC will offer directed study groups for students CS61C.
- They will also offer Drop-in tutoring support for about 20 hours each week.
- Most of these hours will be conducted by paid tutorial staff, but these will also be supplemented by students who are receiving academic credit for tutoring.

Decimal Numbers: Base 10

Digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Example:

3271 =

$$(3 \text{x} 10^3) + (2 \text{x} 10^2) + (7 \text{x} 10^1) + (1 \text{x} 10^0)$$

Numbers: positional notation

- **Number Base B ⇒ B symbols per digit:**
  - **Base 10 (Decimal): 0, 1, 2, 3, 4, 5, 6, 7, 8, 9**
    Base 2 (Binary): 0, 1
- **Number representation:**
  - $d_{31}d_{30} \ldots d_1d_0$ is a 32 digit number
  - value = $d_{31} \times B^{31} + d_{30} \times B^{30} + \ldots + d_1 \times B^1 + d_0 \times B^0$
- **Binary: 0,1 (In binary digits called "bits")**
  - 0b11010 $= 1\times2^4 + 1\times2^3 + 0\times2^2 + 1\times2^1 + 0\times2^0$
    $= 16 + 8 + 2$
    $= 26$
  - Here 5 digit binary # turns into a 2 digit decimal #
  - Can we find a base that converts to binary easily?

#s often written 0b…

Hexadecimal Numbers: Base 16

- **Hexadecimal:**
  0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
  - Normal digits + 6 more from the alphabet
  - In C, written as 0x… (e.g., 0xFAB5)
- **Conversion: Binary⇔Hex**
  - 1 hex digit represents 16 decimal values
  - 4 binary digits represent 16 decimal values

  ⇒1 hex digit replaces 4 binary digits
- **One hex digit is a "nibble". Two is a "byte"**
- **Example:**
  - 1010 1100 0011 (binary) = 0x_____ ?

Decimal vs. Hexadecimal vs. Binary

Examples:

1010 1100 0011 (binary)
= 0xAC3

10111 (binary)
= 0001 0111 (binary)
= 0x17

0x3F9
= 11 1111 1001 (binary)

How do we convert between hex and Decimal?

00	0	0000
01	1	0001
02	2	0010
03	3	0011
04	4	0100
05	5	0101
06	6	0110
07	7	0111
08	8	1000
09	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

MEMORIZE!

Kilo, Mega, Giga, Tera, Peta, Exa, Zetta, Yotta

physics.nist.gov/cuu/Units/binary.html

- Common use prefixes (all SI, except K [= k in SI])

Name	Abbr	Factor	SI size
Kilo	K	2^{10} = 1,024	10^{3} = 1,000
Mega	M	2^{20} = 1,048,576	10^{6} = 1,000,000
Giga	G	2^{30} = 1,073,741,824	10^{9} = 1,000,000,000
Tera	T	2^{40} = 1,099,511,627,776	10^{12} = 1,000,000,000,000
Peta	P	2^{50} = 1,125,899,906,842,624	10^{15} = 1,000,000,000,000,000
Exa	E	2^{60} = 1,152,921,504,606,846,976	10^{18} = 1,000,000,000,000,000,000
Zetta	Z	2^{70} = 1,180,591,620,717,411,303,424	10^{21} = 1,000,000,000,000,000,000,000
Yotta	Y	2^{80} = 1,208,925,819,614,629,174,706,176	10^{24} = 1,000,000,000,000,000,000,000,000

- Confusing! Common usage of "kilobyte" means 1024 bytes, but the "correct" SI value is 1000 bytes

- Hard Disk manufacturers & Telecommunications are the only computing groups that use SI factors, so what is advertised as a 30 GB drive will actually only hold about 28 x 2^{30} bytes, and a 1 Mbit/s connection transfers 10^{6} bps.

kibi, mebi, gibi, tebi, pebi, exbi, zebi, yobi

en.wikipedia.org/wiki/Binary_prefix

- New IEC Standard Prefixes [only to exbi officially]

Name	Abbr	Factor
kibi	Ki	2^{10} = 1,024
mebi	Mi	2^{20} = 1,048,576
gibi	Gi	2^{30} = 1,073,741,824
tebi	Ti	2^{40} = 1,099,511,627,776
pebi	Pi	2^{50} = 1,125,899,906,842,624
exbi	Ei	2^{60} = 1,152,921,504,606,846,976
zebi	Zi	2^{70} = 1,180,591,620,717,411,303,424
yobi	Yi	2^{80} = 1,208,925,819,614,629,174,706,176

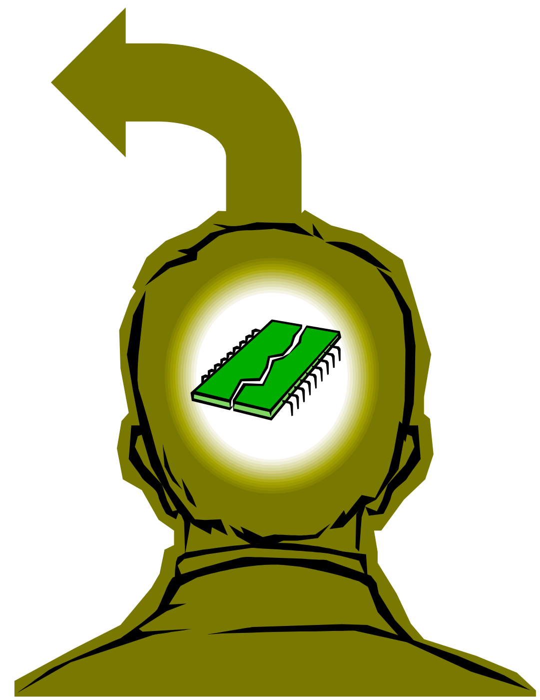
As of this writing, this proposal has yet to gain widespread use…

- International Electrotechnical Commission (IEC) in 1999 introduced these to specify binary quantities.
  - Names come from shortened versions of the original SI prefixes (same pronunciation) and *bi* is short for "binary", but pronounced "bee" :-(
  - Now SI prefixes only have their base-10 meaning and never have a base-2 meaning.

The way to remember #s

- What is 2^{34}? How many bits addresses (I.e., what's `ceil` $\texttt{log}_2$ = `lg` of) 2.5 TiB?
- Answer! 2^{XY} means…

X=0 ⇒ ---	Y=0 ⇒ 1
X=1 ⇒ kibi ~10^3	Y=1 ⇒ 2
X=2 ⇒ mebi ~10^6	Y=2 ⇒ 4
X=3 ⇒ gibi ~10^9	Y=3 ⇒ 8
X=4 ⇒ tebi ~10^{12}	Y=4 ⇒ 16
X=5 ⇒ tebi ~10^{15}	Y=5 ⇒ 32
X=6 ⇒ exbi ~10^{18}	Y=6 ⇒ 64
X=7 ⇒ zebi ~10^{21}	Y=7 ⇒ 128
X=8 ⇒ yobi ~10^{24}	Y=8 ⇒ 256
	Y=9 ⇒ 512



MEMORIZE!

Summary

- Continued rapid improvement in computing
  - 2X every 2.0 years in memory size;
    every 1.5 years in processor speed;
    every 1.0 year in disk capacity;
  - Moore's Law enables processor
    (2X transistors/chip ~1.5 yrs)
- 5 classic components of all computers

  Control Datapath Memory Input Output

  (Control + Datapath = **Processor**)

- Decimal for human calculations, binary for computers, hex to write binary more easily