

**Welcome to CS122A: Embedded System Design**

**Harry Hsieh**  
 Department of Computer Science and Engineering  
 University of California at Riverside

### CS 122A: Embedded System Design

- Lecture
  - SPR 2339
  - TuTh 9:40AM-11:00AM
- Laboratory (SURGE 173)
  - TuTh 6:10PM-9:00PM
  - Yes, there is a lab meeting tonight!!!
- Exams: midterm (10/29, in class), final (12/10, 8-11AM)
- Prerequisite: CS/EE 120B (Digital System)
- Textbooks:
  - “Embedded System Design: A Unified Hardware/Software Approach”, Vahid & Givargis
  - “Real Time Systems and Programming Languages: Ada 95, Real-Time Java and Real-Time C/POSIX”, Burn & Wellings, 3rd edition
  - (Optional) VHDL book, C book, 8051 book

CS122A: Embedded System Design, Fall02

### Instructors

- Lecture: Harry Hsieh
  - [www.cs.ucr.edu/~harry](http://www.cs.ucr.edu/~harry)
  - [harry@cs.ucr.edu](mailto:harry@cs.ucr.edu)
  - Office (787-2030)
    - SURGE 329, TuTh 11AM-noon
    - Check www for cancellation
    - Also available by appointment
  - Class Web Page
    - [www.cs.ucr.edu/~harry/classes\\_files/122A\\_Fall02.html](http://www.cs.ucr.edu/~harry/classes_files/122A_Fall02.html)
- Laboratory:
  - Jason Villarreal ([villarreal@cs.ucr.edu](mailto:villarreal@cs.ucr.edu))
  - Susan Cotterell ([susanc@cs.ucr.edu](mailto:susanc@cs.ucr.edu))
  - Office hours: In Lab

CS122A: Embedded System Design, Fall 02

### Grades

- Laboratory demo's and reports: 30%
  - Attend every lab session, the entire lab session
  - After completing one lab, move on to the next immediately
  - 20% late penalty for every late lab -day, 100% for final project
- Final: 25%, Midterm & Quizzes: 30%
  - Close book exams
  - Extra credits available
- Homeworks: 10%
  - No late homework
  - Can work as a team, but MUST do your own write-up
  - This means no copying!!!
- Class Presentations: 5%
  - Material may be used as extra credit

CS122A: Embedded System Design, Fall02

### Homework #1

- Already posted on WWW.
- Covers chapter 1 & 2 of ESD book
- Due Thursday Oct 3<sup>rd</sup>, at the beginning of the class
- Late homework will NOT be accepted
  - Solution will be handed out at the time it is due!!!

CS122A: Embedded System Design, Fall 02

### Student presentation

- 10 minutes on a computer engineering related article
  - Each student should choose 1 article from
    - List of IEEE Spectrum articles, or
    - Recent (2001/2002) articles from
      - Embedded System Programming
      - IEEE Computer (Micro, Design&Test, Intelligent Systems)
  - Some research outside the article expected
    - Some www research, read another article, a demo...etc
  - Audience questions encouraged
  - Sign up for time slot and article by Tuesday, 10/1
  - First presentation on Tuesday, 10/1

CS122A: Embedded System Design, Fall02

### Articles from IEEE Spectrum

1. "Helping Computers Help Themselves", D. Pescovitz, Sep 2002, pp 49-53.
2. "Just One Word - Plastics", S. Moore, Sep 2002, pp 55-59.
3. "Talk to the Machine", J. Kumagai, Sep 2002, pp 60-64.
4. "Weaving a Web of Ideas", S. Cherry, Sep 2002, pp 65-69.
5. "Let There Be Light", G. Zorpette, Sep 2002, pp70-74.
6. "Linking with Light", N. Savage, Aug 2002, pp 32-36.
7. "Wise Drives", G. Hughes, Aug 2002, pp 37-41.
8. "The Wizardry of Id", D. Kushner, Aug 2002, pp 42-47.
9. "Digital Hubbub", P. Wallich, Jul 2002, pp 26-31.
10. "The Sensible Superconductor", S. Das, Jul 2002, pp 34-37.
11. "Wireless Broadband In a Box", B. Schrick, Jun 2002, pp 38-43.
12. "Checking the Play in Plug-and-Play", H. Goldstein, Jun 2002, pp 50-55.
13. "The Toughest Transistor Yet", L. Eastman, May 2002, pp 28-33.
14. "A Cellphone For All Standards", B. Bing, May 2002, pp 34-39.
15. "Making Unbreakable Code", J. Mullins, May 2002, pp 40-45.
16. "Europe Cracks Down On E-Waste", A. Appelbaum, May 2002, pp 46-51.
17. "The Future of the Microprocessor Business", M. Bass, Apr 2002, pp 34-39.
18. "Can You Trust Your Car?", I. Berger, Apr 2002, pp40-45.
19. "Flywheel Batteries Come Around Again", R. Hehner, Apr 2002, pp 46-51.
20. "Keeping Watch on Glucose", J. Tamada, Apr 2002, pp 52-57.




7

CS122A: Embedded System Design, Fall02

### Articles from IEEE Spectrum

21. "Games Soldiers Play", M. Macedonia, Mar 2002, pp 32-37.
22. "Crossroads For Mixed-Signal Chips", P. Levin, Mar 2002, pp 38-43.
23. "Modular Robots", M. Yim, Feb 2002, pp 30-34.
24. "Tunable Lasers", E. Bruce, Feb 2002, pp 35-39.
25. "Sun Kings Cross the Outback", G. Zorpette, Feb 2002, pp 40-46.
26. "A Call to Disarm", J. Kumagai, Jan 2002, pp 30-37.
27. "Making Intelligence Smarter", G. Zorpette, Jan 2002, pp 38-43.
28. "Improving Security, Preserving Privacy", S. Cass, Jan 2002, pp 44-49.
29. "Extending Healthcare's Reach", S. Moore, Jan 2002, pp 66-71.
30. "Clear Skies Ahead", E. Bretz, Jan 2002, pp78-81.
31. "Building Safer Cars", W. Jones, Jan 2002, pp 82-86.
32. "Embedded Battle Royale", B. Santo, Dec 2001, pp 36-41.
33. "The Quest for the SPIN Transistor", G. Zorpette, Dec 2001, pp 30-35.
34. "Anatomy of Malice", S. Cass, Nov 2001, pp 56-60.
35. "In Search of Transparent Networks", A. Morris, Oct 2001, pp 47-51.
36. "Making Music Pay", S. Cherry, Oct 2001, pp 41-46.
37. "Biological Warfare Canaries", C. Axton, Oct 2001, pp 35-40.
38. "Mindstorms, Not Just a Kid's Toy", P. Wallie, Sep 2001, pp 52-57.
39. "Packages Go Vertical", H. Goldstein, Aug 2001, pp 46-51.
40. "The Topsy Turvy World of Quantum Computing", J. Mullins, Feb 2001, pp 42-49.

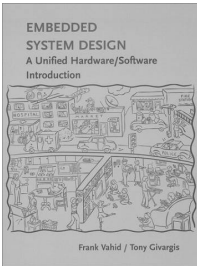


8

CS122A: Embedded System Design, Fall02

### Lecture plan

- Background: first 7 chapters from the ESD text
  - Embedded System Overview, Hardware Design, Software Design, Microcontroller Peripherals, Memory, Interface, Digital camera example
- In depth coverage:
  - High-level design
  - IC technology
  - Design technology




9

CS122A: Embedded System Design, Fall02

### Lecture plan (cont.)

- Advance Topics in Embedded Systems (tentative):
  - Concurrent Processes
  - Scheduling
  - Reliability and fault tolerance
  - Shared variable-based synchronization and communication
  - Message-based synchronization and communication
  - Real-time facility



10

CS122A: Embedded System Design, Fall02


## Embedded System Overview

### An Introduction

11

### Questions to answer

- What are Embedded Systems?
- What are the goals?
  - Design challenge
    - optimizing design metrics
- How to implement? With what?
  - Processor technologies
  - IC technologies
  - Design technologies




12

CS122A: Embedded System Design, Fall02

### Questions to answer

---

- What are Embedded Systems?
- What are the goals?
  - Design challenge
    - optimizing design metrics
- How to implement? With what?
  - Processor technologies
  - IC technologies
  - Design technologies



CS122A: Embedded System Design, Fall02
13

### Computing systems


---

- Most of us think of “desktop” computers
  - PC’s
  - Laptops
  - Mainframes
  - Servers
- But there’s another type of computing system
  - Far more common...

CS122A: Embedded System Design, Fall02
14

### 21<sup>st</sup> Century Embedded Applications

---




CS122A: Embedded System Design, Fall02
15

### Embedded computer systems

---

- Computing systems embedded within larger devices
- Repeatedly carry out a particular function
- May not be recognized by the device’s users
- Any computing system other than a desktop computer



















- Billions of units produced yearly
  - versus millions of desktop units
- About 50 per household
- About 50 per automobile



CS122A: Embedded System Design, Fall02
16

### A “short list” of embedded systems

---

Anti-lockbrakes	Modems			
Auto-focus cameras	MPEGdecoders			
Automatic teller machines	Network cards			
Automatic toll systems	Network switches/routers			
Automatic transmission	On-board navigation			
Avionic systems	Pagers			
Battery chargers	Photo copiers			
Camcorders	Point-of-sale systems			
Cell phones	Portable video games			
Cell-phone base stations	Printers			
Cellless phones	Satellite phones			
Cruise control	Scanners			
Curbside check-in systems	Smart ovens/dishwashers			
Digital cameras	Speechrecognition			
Diskdrives	Stereo systems			
Electronic card readers	Teleconferencing systems			
Electronic instruments	Television			
Electronic toys/games	Temperature controllers			
Factory control	Theft tracking systems			
Fax machines	TV set-top boxes			
Fingerprint identifiers	VCR's, DVD players			
Home security systems	Video game consoles			
Life-support systems	Video phones			
Medical testing systems	Washers and dryers			


And the list goes on and on

CS122A: Embedded System Design, Fall02
17

### Common embedded system characteristics

---

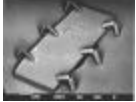
- Single-functioned
  - Executes a single program, repeatedly
- Tightly-constrained
  - Low cost, low power, small, fast, etc.
- Reactive and real-time
  - Continually reacts to changes in the system’s environment
  - Must compute certain results in real-time

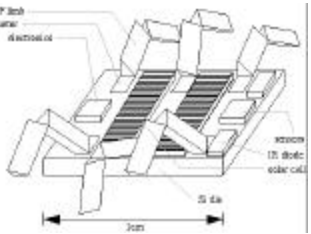


CS122A: Embedded System Design, Fall02
18

### More exotic embedded systems

**Synthetic Insects**






19  
CS122A: Embedded System Design, Fall02

### Smart pen

- Handwriting Recognitor
- Stores up to 10 Pages
- Radio Transmission
- Address Book
- Daily Planner
- Calculator
- Receive/send e-mail
  - Signature Recognition
- Contains ink, too...





Microcontroller  
RAM  
Flash memory  
LCD display  
Infrared transmitter  
Infrared receiver  
Antenna  
Battery

20  
CS122A: Embedded System Design, Fall02

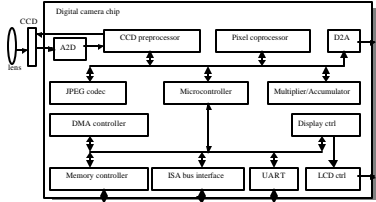
### Smart dust

- Solar Power
- Thick Film Battery
- 2-way Communication
  - Laser
- Light Sensing
- Acceleration Sensing

21  
CS122A: Embedded System Design, Fall02

### Less exotic example: digital camera




- Single-functioned – always a digital camera
- Tightly-constrained – Low cost, low power, small, light, fast
- Reactive to user input plus environment
- Real-time to users' expectation and environment constraints

22  
CS122A: Embedded System Design, Fall02

### Questions to answer

- What are Embedded Systems?
- What are the goals?
  - Design challenge
    - optimizing design metrics
- How to implement? With what?
  - Processor technologies
  - IC technologies
  - Design technologies



23  
CS122A: Embedded System Design, Fall02

### Design challenge – optimizing design metrics

- Obvious design goal:
  - Construct an implementation with desired functionality
- Key design challenge:
  - Simultaneously optimize numerous design metrics
- Design metric
  - A measurable feature of a system's implementation
  - Optimizing design metrics is a key challenge

24  
CS122A: Embedded System Design, Fall02

### Common design metrics

- Cost
  - NRE (Non-Recurring Engineering) cost
    - The one-time monetary cost of designing the system
  - Unit cost
    - cost of manufacturing each copy of the system, excluding NRE cost
- Size
- Performance
  - Execution time
  - Throughput
- Power
- Flexibility
  - the ability to change the functionality without incurring heavy NRE cost

CS122A: Embedded System Design, Fall02 25

### More design metrics

- Time-to-prototype:
  - time needed to build a working version of the system
- Time-to-market
- Maintainability
  - ability to modify the system after its initial release
- Safety
- many more...

CS122A: Embedded System Design, Fall02 26

### Design metric competition

- Improve one may worsen the other

CS122A: Embedded System Design, Fall02 27

### Time-to-market: a demanding design metric

- Time required to develop a product to the point it can be sold to customers
- Market window
  - Period during which the product would have highest sales
- Average time-to-market constraint is about 8 months
- Delays can be costly

CS122A: Embedded System Design, Fall02 28

### Simplified revenue model

- Product life = 2W, peak at W, 45% market rise rate
- Time of market entry defines a triangle
  - Triangle area equals total revenue (Area = 1/2 \* base \* height)
    - On-time total revenue = 1/2 \* 2W \* W = W<sup>2</sup>

CS122A: Embedded System Design, Fall02 29

### Losses due to delayed market entry

- Delay total revenue = 1/2 \* (2W-D)\*(W-D)
- Percentage total revenue loss = (W<sup>2</sup> - 1/2 \* (2W-D)\*(W-D))/W<sup>2</sup>
- = (2W<sup>2</sup> - 2W<sup>2</sup> - 3WD + D<sup>2</sup>)/2W<sup>2</sup>
- = D(3W-D)/2W<sup>2</sup>

CS122A: Embedded System Design, Fall02 30

### Losses due to delayed market entry

- Examples
  - Lifetime  $2W=52$  weeks, delay  $D=4$  weeks
    - means  $4(3^{*}26-4)/2^{*}26=22\%$  loss
  - $D=10$  weeks
    - means  $10(3^{*}26-10)/2^{*}26=50\%$  loss
- Delays are costly!

31  
CS122A: Embedded System Design, Fall02

### NRE and unit cost metrics

- Costs:
  - NRE cost (Non-Recurring Engineering cost)
    - The one-time monetary cost of designing the system
  - Unit cost
    - Cost of manufacturing each copy of the system, excluding NRE cost
- total cost = NRE cost + unit cost \* # of units
- per-product cost = total cost / # of units = (NRE cost / # of units) + unit cost

32  
CS122A: Embedded System Design, Fall02

### NRE cost metrics

- Compare technologies by costs -- best depends on quantity
  - Technology A: NRE=\$2,000, unit=\$100
  - Technology B: NRE=\$30,000, unit=\$30
  - Technology C: NRE=\$100,000, unit=\$2

total cost = NRE cost + unit cost \* # of units

33  
CS122A: Embedded System Design, Fall02

### Unit cost metrics

- Compare technologies by costs -- best depends on quantity
  - Technology A: NRE=\$2,000, unit=\$100
  - Technology B: NRE=\$30,000, unit=\$30
  - Technology C: NRE=\$100,000, unit=\$2

per-product cost = (NRE cost / # of units) + unit cost

34  
CS122A: Embedded System Design, Fall02

### Performance design metric

- Length of time system takes to execute "desired task"
  - Not necessarily
    - CPI, MIPS, Clock frequency, instruction/sec...
- Latency (response time)
  - start of task's execution to end
    - E.g. 0.25 second to process an image for camera A and B
- Throughput
  - number of tasks processed per unit of time
    - E.g. 4 images per second for camera A
  - Parallelism/pipelining can increase throughput
    - E.g. camera B may be able to process 6 images per second
- Speedup
  - Speedup of B over A = performance of B / performance of A
    - E.g. Throughput speedup: 1.5
    - E.g. Latency speedup: 1

35  
CS122A: Embedded System Design, Fall02

### Questions to answer

- What are Embedded Systems?
- What are the goals?
  - Design challenge
    - optimizing design metrics
- How to implement? With what?
  - Processor technologies
  - IC technologies
  - Design technologies

36  
CS122A: Embedded System Design, Fall02

### Three key embedded system technologies

- Technology
  - A manner of accomplishing a task, especially using technical processes, methods, or knowledge
- Three key technologies for embedded systems
  - Processor technology
  - IC technology
  - Design technology

37

CS122A: Embedded System Design, Fall02

### Processor technology

- The architecture of the computation engine used to implement a system's desired functionality
- Types of processors
  - General-purpose processors -- "software" (programmable)
  - Single-purpose processors -- "hardware"
  - Application-specific instruction processors (ASIP) -- in between
- Processor does not have to be programmable
  - One that processes...

38

CS122A: Embedded System Design, Fall02

### General-purpose processors

- Programmable device used in a variety of applications
  - Also known as "microprocessor"
- Features
  - Program memory
  - General datapath with large register file and general ALU
- E.g. Pentium

39

CS122A: Embedded System Design, Fall02

### General-purpose processors

- User benefits
  - Low time-to-market
  - Low NRE costs
  - High flexibility
  - Low unit cost (small quantity)
    - Compare to custom
  - High performance for some app.
    - Control and computation (possibly)
- Possible drawback
  - Need too much juice
  - May be too big
  - High unit cost (large quantity)
    - Compare make custom
  - Low performance for some app.
    - Image and sound processing (possibly)

40

CS122A: Embedded System Design, Fall02

### Customization

- Processors vary in their customization for the problem at hand

total = 0  
for i = 1 to N loop  
total += M[i]  
end loop

Desired functionality

General-purpose processor

41

CS122A: Embedded System Design, Fall02

### Single-purpose processors

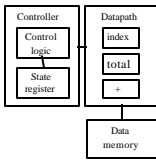
- Digital circuit designed to execute exactly one program
  - a.k.a. "coprocessor," "accelerator", "peripheral", or just a custom hardware
- Features
  - Contains only the components needed to execute a single program
  - No program memory

42

CS122A: Embedded System Design, Fall02

### Single-purpose processors

- **Benefits**
  - Low power
  - Small size
  - Low cost for large quantities
    - Compare to GP
  - High performing for some app.
- **Drawback**
  - Long time -to-market
  - High NRE costs
  - Low flexibility
  - High unit cost (small quantity)
    - Compare to GP
  - Low performance for some app.
    - Given cost constraint




43

CS122A: Embedded System Design, Fall 02


### Customization

- Processors vary in their customization for the problem at hand




Desired functionality

```
total = 0
for i = 1 to N loop
  total += M[i]
end loop
```



General-purpose processor



Single-purpose processor

44

CS122A: Embedded System Design, Fall 02

### Software vs. Hardware

#### General-purpose processors

- **User benefits**
  - Low time-to-market
  - Low NRE costs
  - High flexibility
  - Low unit cost (small quantity)
    - Compare to custom
  - High performance for some app.
    - Control and computation (possibly)
- **Possible drawback**
  - Need too much juice
  - May be too big
  - High unit cost (large quantity)
    - Compare make custom
  - Low performance for some app.
    - Image and sound processing (possibly)

#### Single-purpose processors

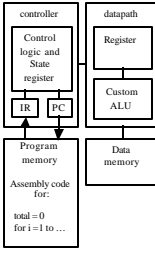
- **Benefits**
  - Low power
  - Small size
  - Low cost for large quantities
    - Compare to GP
  - High performing for some app.
- **Drawback**
  - High time -to-market
  - High NRE costs
  - Low flexibility
  - High unit cost (small quantity)
    - Compare to GP
  - Low performance for some app.
    - Given cost constraint

45

CS122A: Embedded System Design, Fall 02

### Application-specific processors

- Programmable processor optimized for a particular class of applications having common characteristics
  - Compromise between general-purpose and single-purpose processors
- **Features**
  - Program memory
  - Optimized datapath
  - Special functional units
- **Benefits**
  - Some flexibility
  - good performance, size and power
    - For certain domain of application





46

CS122A: Embedded System Design, Fall 02

### Common ASIPs

- **Microcontroller**
  - Monitor and control
  - Peripherals: Serial communication, timers, counters, PWMs, A2D, D2A...
- **Digital Signal Processor**
  - Video and audio
    - Filtering and transformation, MAC operations
  - Contain wide data path and wide memory path
- **Network Processor**
  - Massively parallel executing elements





47

CS122A: Embedded System Design, Fall 02


### Customization

- Processors vary in their customization for the problem at hand

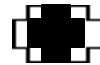


Desired functionality


```
total = 0
for i = 1 to N loop
  total += M[i]
end loop
```



General-purpose processor



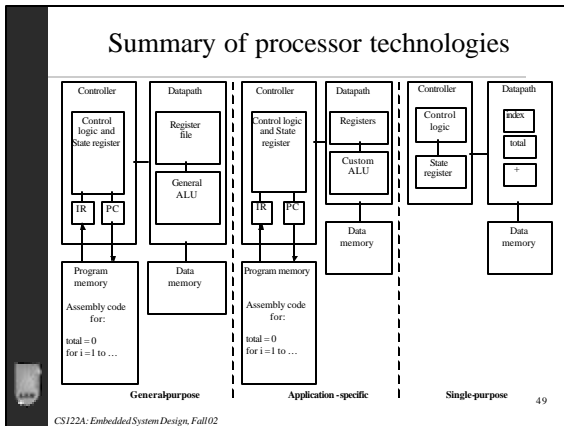
Application-specific processor



Single-purpose processor

48

CS122A: Embedded System Design, Fall 02



### Questions to answer

- What are Embedded Systems?
- What are the goals?
  - Design challenge
    - optimizing design metrics
- How to implement? With what?
  - Processor technologies
  - IC technologies
  - Design technologies

CS122A: Embedded System Design, Fall02

### IC technology

- The manner in which a digital (gate-level) implementation is mapped onto an IC
  - IC: Integrated circuit, or “chip”
  - IC technologies differ in their customization to a design
  - IC’s consist of numerous layers (10 or more)
    - IC technologies differ in when each layer is built

IC package      IC      gate      oxide      trench/channel      Silicon substrate

Intel's 100nm gate

CS122A: Embedded System Design, Fall02

### IC technology

- Three types of IC technologies
  - Full-custom/VLSI
  - Semi-custom ASIC (gate array and standard cell)
  - PLD (Programmable Logic Device)

CS122A: Embedded System Design, Fall02

### Full-custom/VLSI

- VLSI - Very Large Scale Integration
- All layers are optimized for an embedded system’s particular digital implementation
  - Placing transistors
  - Sizing transistors
  - Routing wires
- Benefits
  - Excellent performance
  - Small size
  - Low power
- Drawback
  - High NRE Cost
  - REAL BAD time -to-market
- Only use for highest performing and largest quantity

Intel Pentium IV

CS122A: Embedded System Design, Fall02


### Semi-custom ASIC

- ASIC - Application-Specific IC
- Lower layers are fully or partially built
  - Designers are left with routing of wires and maybe placing some blocks
- Benefits
  - Good performance
  - Good size
  - Lower NRE cost
    - than full-custom
  - O.k. time to market

CS122A: Embedded System Design, Fall02

### PLD

- Programmable Logic Device
  - Programmable Logic Array, Programmable Array Logic, Field Programmable Gate Array
- All layers already exist
  - Designers can purchase an IC
  - To implement desired functionality
    - Connections on the IC are either created or destroyed to implement
- Benefits
  - Very low NRE costs
  - Great time to market
- Drawback
  - High unit cost, bad for large volume
  - Power
    - Except special PLA
  - slower



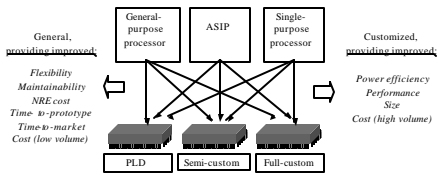
1600 usable gate, 7.5 ns  
\$7 list price

55

CS122A: Embedded System Design, Fall02

### Independence of processor and IC technologies

- Basic tradeoff
  - General vs. custom
  - With respect to processor technology or IC technology
  - The two technologies are independent




56

CS122A: Embedded System Design, Fall02

### Questions to answer


- What are Embedded Systems?
- What are the goals?
  - Design challenge
    - optimizing design metrics
- How to implement? With what?
  - Processor technologies
  - IC technologies
    - Trends
  - Design technologies

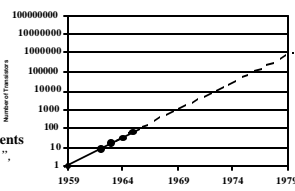


57

CS122A: Embedded System Design, Fall02

### The World According to Moore






“Cramming More Components Onto Integrated Circuits”, *Electronics*, April, 1965

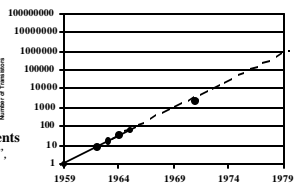
- 1959, invention of IC (Kilby/Noyce)
- Moore’s Law
  - Number of transistors shall double every year

58

CS122A: Embedded System Design, Fall02


### The World According to Moore





“Cramming More Components Onto Integrated Circuits”, *Electronics*, April, 1965

- 1959, invention of IC (Kilby/Noyce)
- Moore’s Law
  - Number of transistors shall double every year




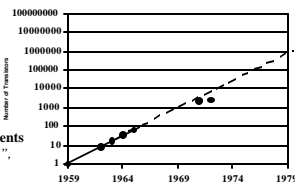
4004      8008  
2,250 trans      2,500 trans  
1971                  1972

59

CS122A: Embedded System Design, Fall02


### The World According to Moore





“Cramming More Components Onto Integrated Circuits”, *Electronics*, April, 1965

- 1959, invention of IC (Kilby/Noyce)
- Moore’s Law
  - Number of transistors shall double every year



4004      8008  
2,250 trans      2,500 trans  
1971                  1972

60

CS122A: Embedded System Design, Fall02

### The World According to Moore

**"Cramming More Components Onto Integrated Circuits"**,  
*Electronics*, April, 1965

- 1959, invention of IC (Kilby/Noyce)
- Moore's Law
  - Number of transistors shall double every year

4004 8008 8080  
2,250 trans 2,500 trans 5,000 trans<sub>1</sub>  
1971 1972 1974

CS122A: Embedded System Design, Fall02

### The World According to Moore

**"Cramming More Components Onto Integrated Circuits"**,  
*Electronics*, April, 1965

- 1959, invention of IC (Kilby/Noyce)
- Moore's Law
  - Number of transistors shall double every years

4004 8008 8080  
2,250 trans 2,500 trans 5,000 trans<sub>2</sub>  
1971 1972 1974

CS122A: Embedded System Design, Fall02

### The Rest Is History

8086 29,000 trans 1978  
286 120,000 trans 1982  
386 275,000 trans 1985  
486 1,180,000 trans 1989  
Pentium 3,100,000 trans 1993  
Pentium II 7,500,000 trans 1997  
Pentium III 24,000,000 trans 1999  
Pentium IV 42,000,000 trans<sup>63</sup> 2000

CS122A: Embedded System Design, Fall02

### What Future May Bring

SIA  
ITRS

- Take the average, let's say doubling every 18 months
- By the year 2014, each high-end IC will have
  - 5-12 billion transistors, 4000-9000 pins, 30 nm feature size, on-chip local clock of 15GHz,  $V_{dd}$  range 0.3V to 0.6 V, power dissipation at 200 W.

CS122A: Embedded System Design, Fall02

### Graphical illustration of Moore's law

- Magic of exponential growth
- A 2002 chip can have about 15,000 copies of 1981 chips inside
  - In 2002's technology

CS122A: Embedded System Design, Fall02

### Questions to answer

- What are Embedded Systems?
- What are the goals?
  - Design challenge
    - optimizing design metrics
- How to implement? With what?
  - Processor technologies
  - IC technologies
  - Design technologies

CS122A: Embedded System Design, Fall02


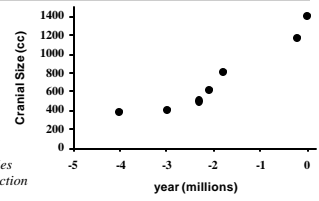
### Design Technology

- The manner in which we convert our concept of desired system functionality into an implementation
- Moore's law
  - Exponentially increasing number of available functions
- Designer must keep up with Moore's law
  - Improvement in design technology

67


CS122A: Embedded System Design, Fall 02

### The Theory of Evolution

Charles Darwin, *The Origin of Species by Means of Natural Selection* 1859

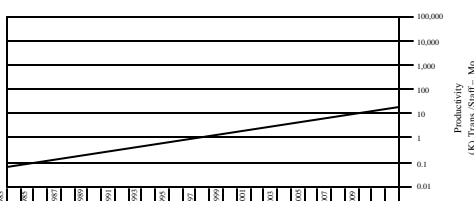
- 4 million BC, "Lucy" has cranial size of 380 cc
- Today, *Homo Sapiens* average 1400 cc
  - Double every 2.08 million years
- Assume no change in brain cell structure
  - Human productivity falls behind!!!



68

CS122A: Embedded System Design, Fall 02

### Design productivity exponential increase



- Number of transistor a designer "produce"
  - Exponential increase over the past few decades
    - Improving design technology
    - I.e. brain cells getting better...

69

CS122A: Embedded System Design, Fall 02

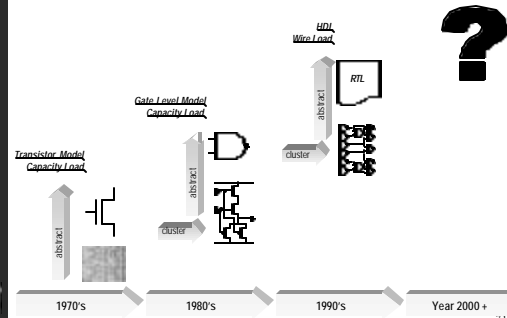
### Design Technology

- Want to design at higher level of abstraction
  - Look at the bigger picture
  - Not having to worry about little details
- Enable by:
  - Compilation/Synthesis
    - Automates exploration & insertion of implementation details for lower level
    - E.g. logic synthesis, software compiler, system synthesis
  - Library/Intellectual Property
    - Incorporates pre-designed implementation from low to higher level.
    - E.g. M-Core, ARM, Memory & Peripheral controller
  - Test/Verification
    - Ensures correct functionality at each level.
    - Reducing costly iterations between levels.
    - E.g. simulation (RTL, behavior, HWSW), model checking
- Standards, Languages, Frameworks...

70

CS122A: Embedded System Design, Fall 02

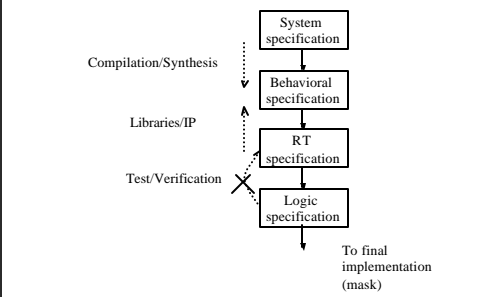
### Next Level of Abstraction?



71

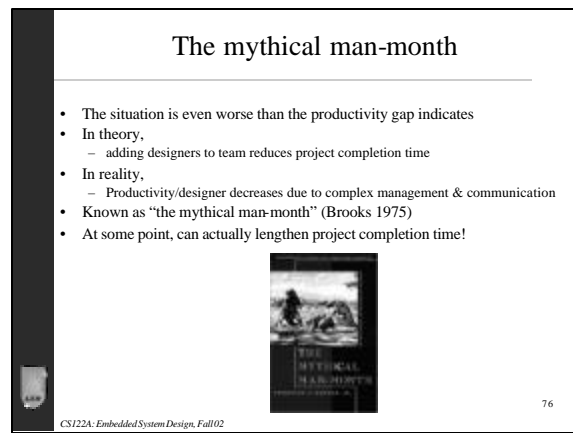
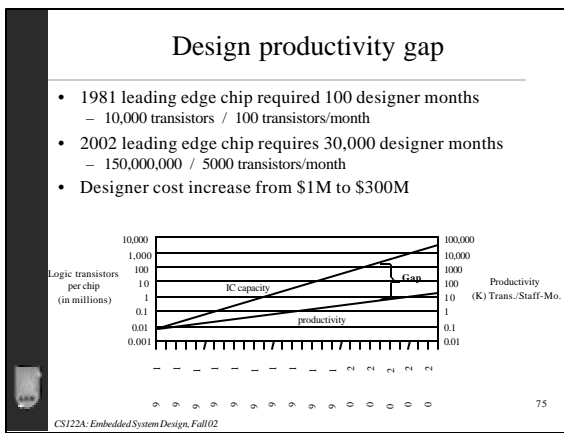
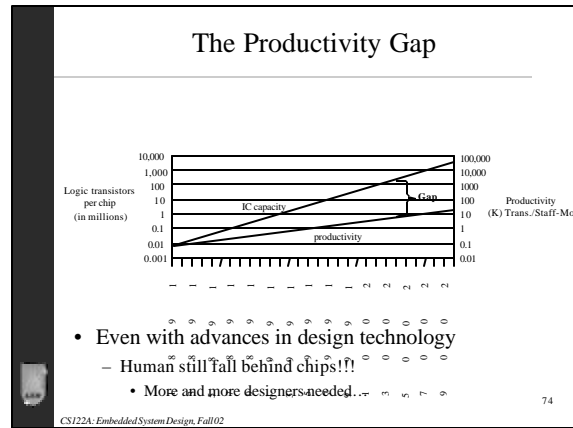
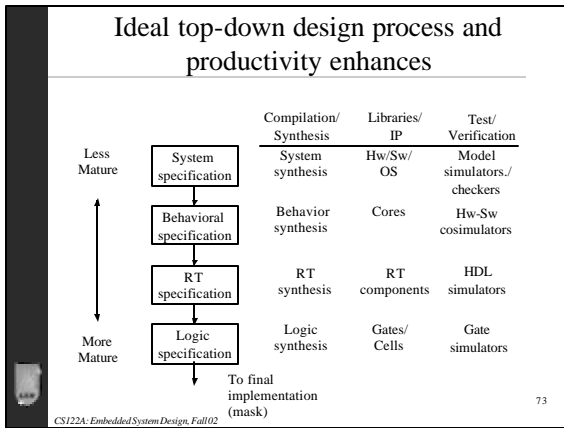
CS122A: Embedded System Design, Fall 02

### Ideal top-down design process and productivity enhances



72

CS122A: Embedded System Design, Fall 02

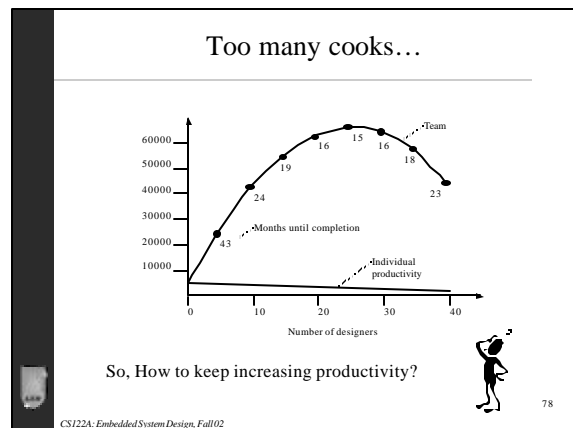


### The mythical man-month vs 486

- 1 M. Transistors; 1 designer = 5000 trans/mo; Add. Designer = -100 trans/mo

# of designers	Individual Productivity	Team Productivity
1	5000 trans/mo	5000 trans/mo
2	4900	9800
3	4800	14400
4	4700	18800
5	4600	23000
10	4000	40000
20	3000	60000
30	2000	60000
40	1000	40000

77  
CS122A: Embedded System Design, Fall02



### Next Level of Abstraction?

Transistor Model Capacity Load (1970's)

Gate Level Model Capacity Load (1980's)

RTL (1990's)

VHDL (Year 2000+)

79

CS122A: Embedded System Design, Fall 02

### The co-design ladder

- In the past:
  - Hardware and software design technologies were very different
  - Recent maturation of synthesis enables a unified view of hardware and software
- Hardware/software "codesign"

Sequential program code (e.g., C, VHDL)

Compilers (1960's, 1970's)

Behavioral synthesis (1990's)

Assembly instructions

Register transfers

Assemblers, linkers (1950's, 1960's)

RT synthesis (1980's, 1990's)

Machine instructions

Logic equations / FSM's

Logic gates

Implementation: VLSI, ASIC, or PLD

Microprocessor plus program bits: "software"

VLSI, ASIC, or PLD implementation: "hardware"

*The choice of hardware versus software for a particular function is simply a tradeoff among various design metrics, like performance, power, size, NRE cost, and especially flexibility; there is no fundamental difference between what hardware or software can implement.*

80

CS122A: Embedded System Design, Fall 02

### The solution

- Designer must have a view of the entire system
- The designer must know
  - Hardware
  - Software
  - Interface
  - Peripheral
  - Memory
  - .....
- CS122A goal
  - Let's get to it

Sequential program code (e.g., C, VHDL)

Compilers (1960's, 1970's)

Behavioral synthesis (1990's)

Assembly instructions

Register transfers

Assemblers, linkers (1950's, 1960's)

RT synthesis (1980's, 1990's)

Machine instructions

Logic equations / FSM's

Logic gates

Implementation: VLSI, ASIC, or PLD

Microprocessor plus program bits: "software"

VLSI, ASIC, or PLD implementation: "hardware"

81

CS122A: Embedded System Design, Fall 02

### Administrative Stuff

- Reminder: Hw 1 due 10/3 at beginning of class
  - No late homework accepted
- Presentation
  - Sign up for article and time slot by 10/1
  - 5% of the grade
  - Highly recommended to:
    - Send your presentation material to me no later than 8AM that day...
    - ...powerpoint files, ps, pdf...

82

CS122A: Embedded System Design, Fall 02