CSCE 4114
Embedded Systems
Class website:
http://hthreads.csce.uark.edu/wiki/4114

David Andrews

dandrews@uark.edu

Welcome !

- Today we will just cover programmatics and gee-whiz introduction

Office Hours : MWF 3:30 – 4:30
    Default Remote Via Blackboard :
    F2F per request

    -TA: TBD

Programmatics (on wiki)

Grade Breakdown

1. 2 exams (Mid/Final)   := 20% each.
   2 Exam builds on earlier material but is NOT comprehensive.
2. Homework   := 20%
3. Laboratories   := 40%

Midterm Tentative Oct 15 will confirm2 weeks before.

1. Class = Lecture + Lab

   1. Lectures: Concepts and theory
      3:1 rule applies (Three hours study for every hour lecture)
   2. Labs: Application of the Theory
      Frustrating but satisfying and fun ! You get to build stuff !

Programmatics

- Homework: Assignments given in class Usual due date is 1 week later.
    - Gives you a chance to read, try and then ask questions in next class.
    - Due at Beginning of class thru Blackboard.
        - Each late day costs 25%
        - Will waive for valid reason presented before due date

Programmatics

- Labs
    - TBD but probably ~6 Labs during semester.
    - Will use Xilinx Arty Boards. You need to pick them up this week.
    - Week #1: Prelab you do at home.
    - Lab writeups + Grading explained in more detail in lab (Week 2)

Programmatics

- Textbook:

  "Programming Embedded Systems"

  Vahid, Givargis, Miller

1. Sign in or create an account at:

   | learn.zybooks.com |

2. Enter zyBook code:

   | UARKCSCE4114AndrewsFall2021 |

3. Subscribe

   $58 Open August 10th -> Dec 23rd

Programmatics

- ## Lectures:

    1. From Textbook

    2. Lecture slides and additional materials posted on blackboard and webpage.

    3. Reading assignments will be posted the weekend before prior to coverage. You should read and familiarize yourself with materials before lecture.

Programmatics

- Academic Honesty: Very important. You are required to do your own work.
  - Labs can be done by partners. It's ok to work together, get together after to discuss. However, you need to write up your own report in your own words.
  - TA will give you formats and expectations for reports.
  - Dishonesty will be dealt with swifty !
  - See the Universities Procedure at:
    http://provost.uark.edu/245.php

Embedded Systems

- ## What you will study:
  - ### Embedded Systems interfacing and design
  - ### Where Hardware and Software co-exist
    - Hardware Organization:
      - CPU: Basic components (how to build in CSCE 2214)
      - Bus Interfacing: Signals and protocols for communication between CPU & all other components
      - Memory: Decoding and hooking up to Bus
      - Peripherals
        - I/O getting data in and out
        - Priority Interrupt Controller: How things get the CPU's attention
        - Custom Components: Accelerators and additions

Embedded Systems

- ## What you will study:
    - ### Embedded Systems interfacing and design
    - ### Where Hardware and Software co-exist
        - Software Organization:
            - Assembler := CPU's language.
                - Internal CPU Arithmetic and Boolean Instructions
                - Data Movement into and out of CPU: How to communicate with other system components
                - Protocol Stacks (How C/Java Functions & Subroutines actually get implemented)
            - Interrupt Routines:
                - Special Instructions that allows external devices to request service
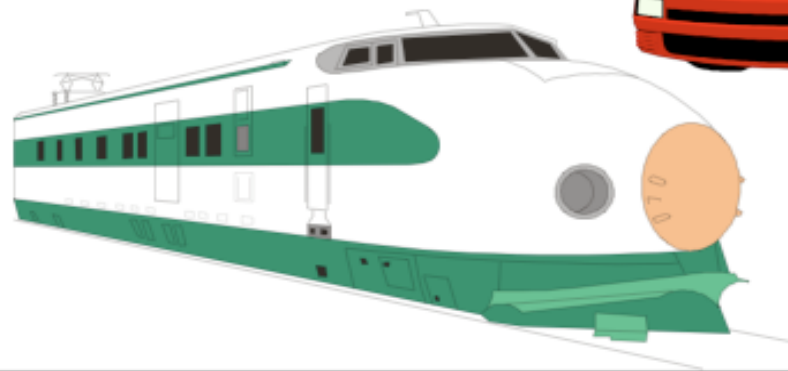
Embedded Systems

1. What is an Embedded System ?

Embedded System =
Computers Inside a Product

From Koopman http://www.ece.cmu.edu/~ece649/lectures/01_intro.pdf
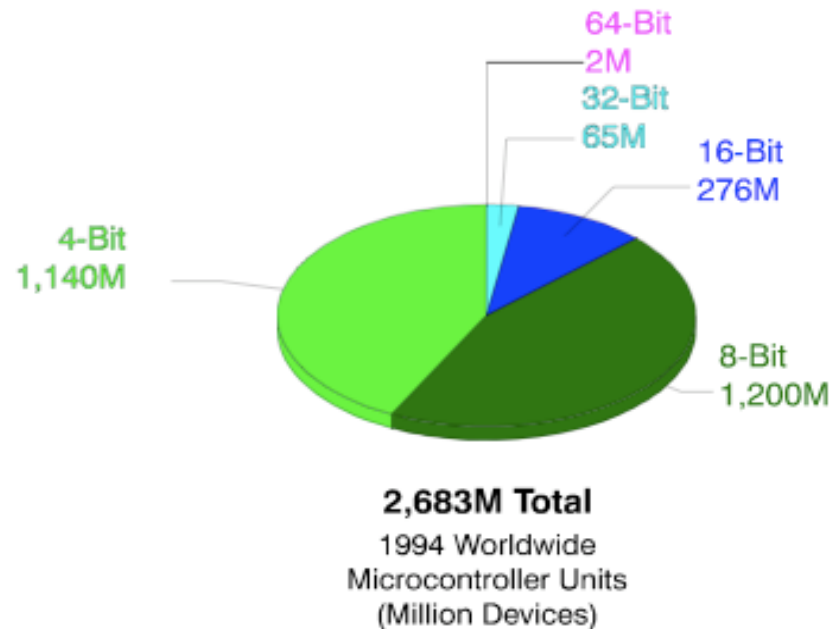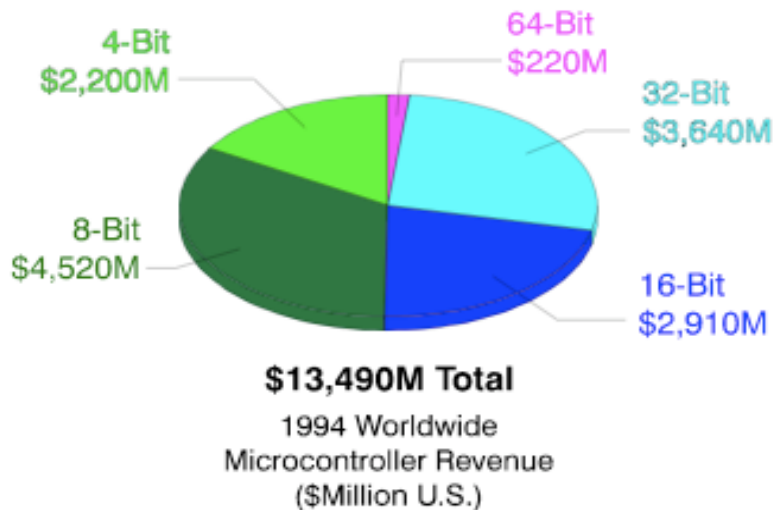
Embedded Systems

1. What is an Embedded System ?

2. Why are Embedded Systems Important ?

Small Computers Rule The Marketplace

- ◆ **~80 Million PCs vs. ~3 Billion Embedded CPUs Annually in 1995**
  - 150 Million PCs and 7.5 Billion embedded CPUs + in 2000

4-Bit $2,200M
64-Bit $220M
32-Bit $3,640M
8-Bit $4,520M
16-Bit $2,910M
$13,490M Total
1994 Worldwide
Microcontroller Revenue
($Million U.S.)

64-Bit 2M
32-Bit 65M
16-Bit 276M
4-Bit 1,140M
8-Bit 1,200M
2,683M Total
1994 Worldwide
Microcontroller Units
(Million Devices)

Approximated from EE Times, March 20, 1995
Source: The Information Architects

From Koopman http://www.ece.cmu.edu/~ece649/lectures/01_intro.pdf

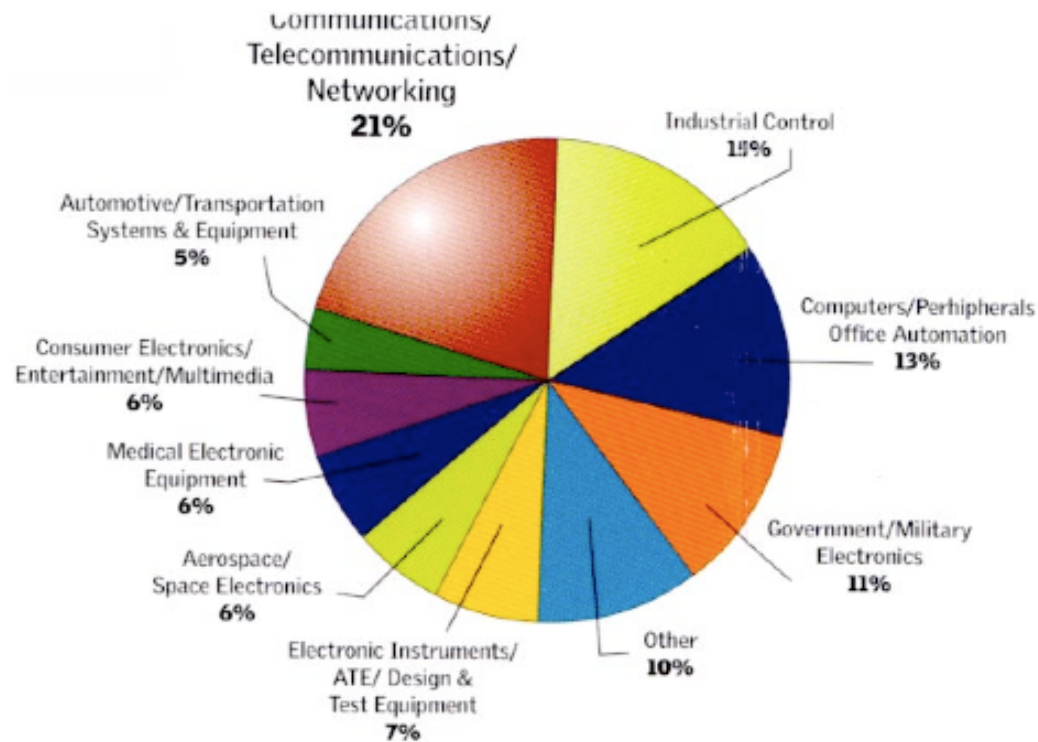> 99% of CPU's sold go into Embedded Systems

There Are Many Application Areas

Primary End Product of Embedded Subscribers
Source: *ESP* Dec. 1998 BPA Audit

- Communications/Telecommunications/Networking 21%
- Industrial Control 14%
- Computers/Perhipherals Office Automation 13%
- Government/Military Electronics 11%
- Other 10%
- Electronic Instruments/ATE/ Design & Test Equipment 7%
- Aerospace/Space Electronics 6%
- Medical Electronic Equipment 6%
- Consumer Electronics/Entertainment/Multimedia 6%
- Automotive/Transportation Systems & Equipment 5%

Which one will you be working in ?

From Koopman http://www.ece.cmu.edu/~ece649/lectures/01_intro.pdf

Embedded Systems

1. What is an Embedded System ?

2. Why are Embedded Systems Important ?

3. What do You Need to Know ?

Generic Embedded System Designer Skill Set

◆ **Appreciation for multi-disciplinary nature of design**
  - System skills; system = HW + SW + …
  - Understanding of engineering beyond digital logic
  - Ability to take a project from specification through production

◆ **Communication & teamwork skills**
  - Work with other disciplines, manufacturing, marketing
  - Work with customers to understand the real problem being solved
  - Make a good presentation; even better -- write "trade rag" articles

◆ **And, by the way, technical skills too…**
  - Low level: Microcontrollers, FPGA/ASIC, assembly language, A/D, D/A
  - High level: Object-oriented Design, C/C++, Real Time Operating Systems, Critical System design
  - Meta level: Creative solutions to highly constrained problems
  - Likely in the future: Unified Modeling Language, embedded networks
  - Uncertain future: Java, Windows CE

From Koopman http://www.ece.cmu.edu/~ece649/lectures/01_intro.pdf

Computer System Design Lab

Summary

- This course will give you appreciation for the fun and difficulty of designing and building and embedded system

    - If you are a "Software Person": you will learn how your software is being implemented. You will learn how to write embedded software

    - If you are a "Hardware Person": you will learn how your hardware is being used and controlled. You will learn how to create hardware that is usable by software.