COMPE375 - Embedded Systems Programming

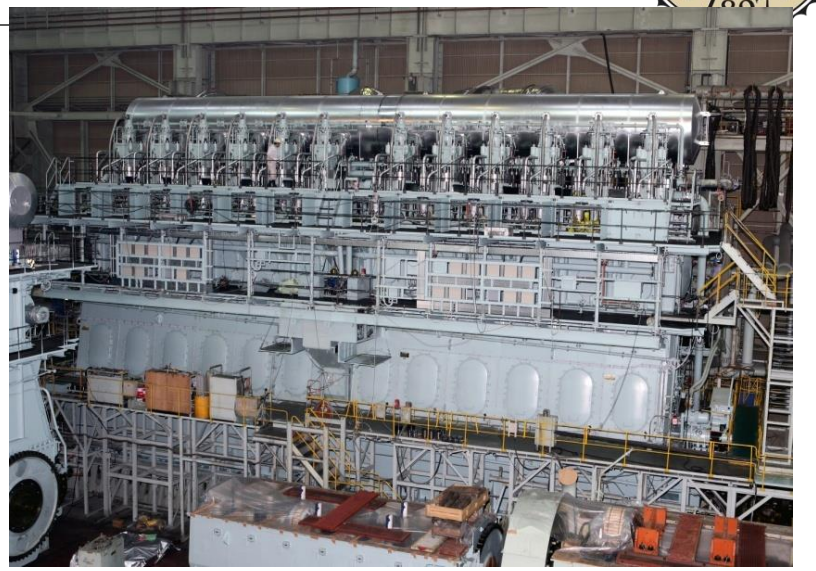
Ke Huang
Department of Electrical and Computer Engineering
San Diego State University

What are embedded systems?

- Systems which use computation to perform a *specific function*
- Embedded within a larger device and environment
- Heterogeneous & reactive to environment

Main reason for buying is **not** information processing

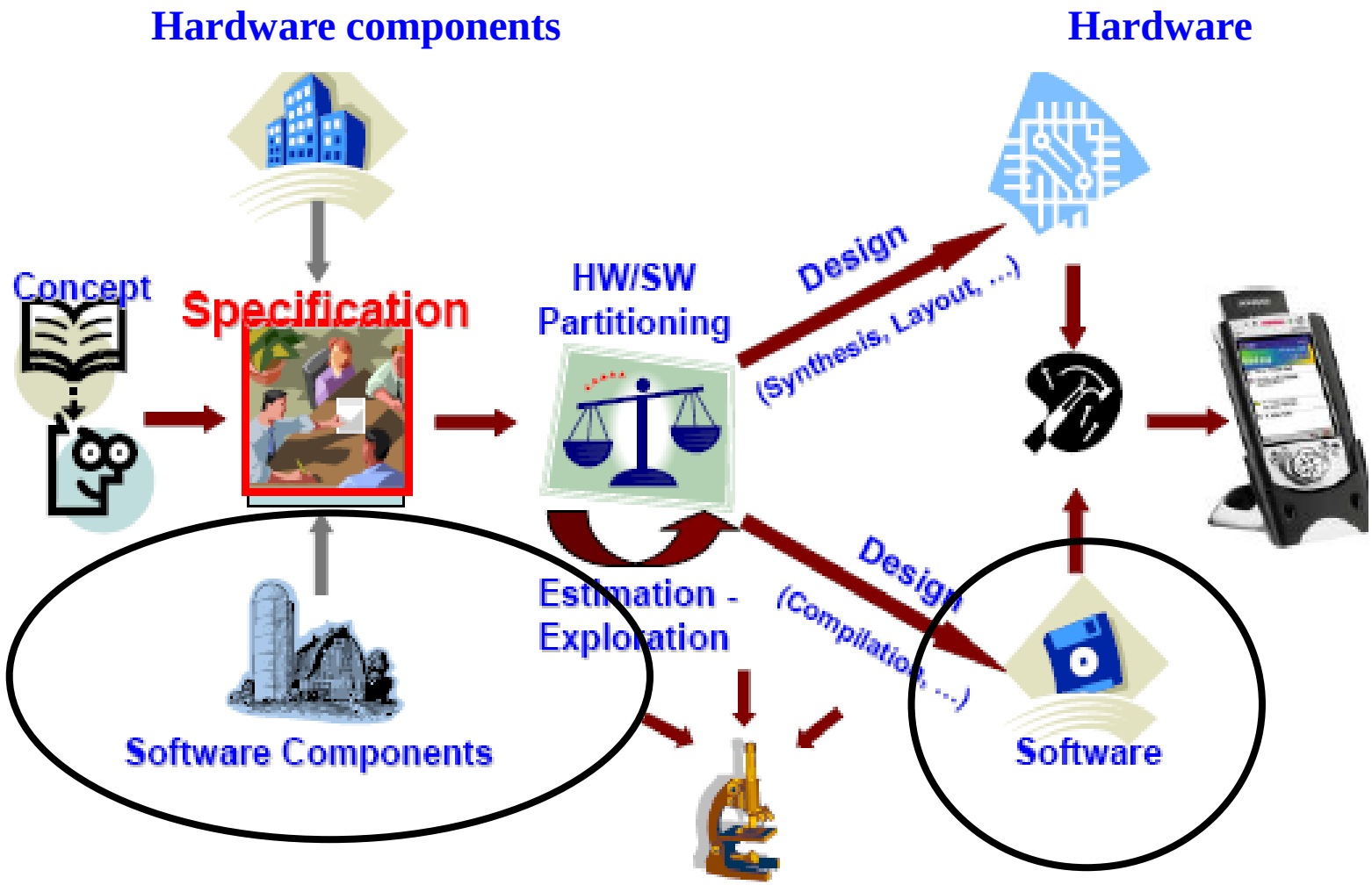
Embedded Systems



- Computers used as part of a larger system
  - That usually doesn't look like a computer
  - That usually controls physical devices
- Often reliability is critical
  - "Critical" as in "if the system fails someone might die"
- Often resources (memory, processor capacity) are limited
- Often real-time response is essential

Source: Bjarne Stroustrup

Embedded System Design



Welcome to COMPE 375!

- Computer Engineering – Embedded Systems Programming
- Instructor: Ke Huang
  - Email: **khuang@sdsu.edu**; put [COMPE375] in subject line
  - Lecture: Thursday 3:30-6:10pm @EBA-258
  - Office Hours: Monday/Wednesday 3:30pm-5:30pm@E-202B
  - Lab Sessions: Monday 2pm-4:40pm @E-207
  - Lab Assistant: Saipriyati Singh
    - Email: **ssingh5@sdsu.edu**
    - Office hours: TBD

- Class Website: We will use blackboard for announcements, updates and discussion board

Course Objectives

- Microcontroller vs. desktop PC
- Familiarity with hardware
- Familiarity with tools
- Hands-on exposure
- Low level programming, interfacing
- Microcontroller applications

Course Requirements

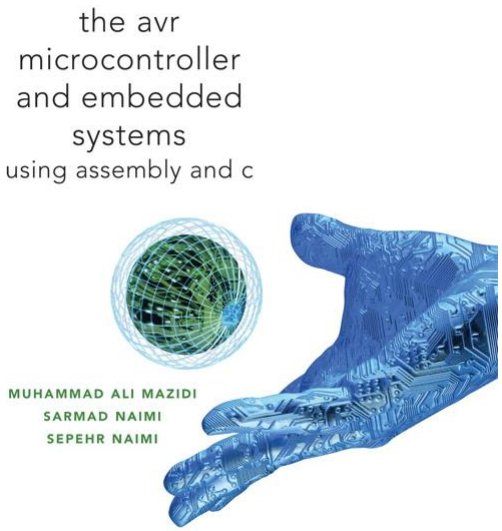
- COMPE 271– Computer Organization (grade of C- or better)
  - COMPE 160– Introduction to Computer Programming
  - COME 270 – Digital Systems
  - Are your pre-requirements in the system? All must submit a copy of unofficial transcript, degree evaluation, or transcript syllabus TO ME showing CompE-271 or equivalent

- Knowledge
  - Digital hardware, computer architecture (ISA, organization), programming, algorithms

- Skills
  - Ability to program in C – THIS IS A MUST
    - This class will not teach you C programming!!!!
  - Ability to look up hardware and coding data over the Internet

Textbook & Assigned Reading

- Avr Microcontroller and Embedded Systems The: Using Assembly and C. Muhammad Ali Mazidi, Sarmad Naimi, and Sepehr Naimi. Prentice Hall by Pearson. ISBN-10: 0138003319



- We will release additional resources for coding and embedded systems
  - Datasheets for the embedded hardware that is used in the lab sessions
  - Examples for embedded software implementation
  - Examples for refreshing coding skills

Course Grading

- Midterm exams – in class
  - First midterm (20%) – Feb 28th 2019
  - Second midterm (20%) – Apr 25th 2019
- Final exam (30%) – TBD
- Lab assignments (30%)
  - 10 laboratory assignments
- Second midterm exam will NOT be cumulative but the final exam will be

Academic Dishonesty Policy

- (from the "University Policies" section of the 2012-2013 General Catalog)
- Institutions of higher education are founded to impart knowledge, seek truth, and encourage one's development for the good of society. University students shall thus be intellectually and morally obliged to pursue their course of studies with honesty and integrity. Therefore, in preparing and submitting materials for academic courses and in taking examinations, a student shall not yield to cheating or plagiarism, which not only violate academic standards but also make the offender liable to penalties explicit in Section 41301 of Title 5,California Code of Regulations as follows:
- Expulsion, Suspension, and Probation of Students. Following procedures consonant with due process established pursuant to Section 41304, any student of a campus may be expelled, suspended, placed on probation, or given a lesser sanction for one or more of the following causes that must be campus related.

Academic Dishonesty Policy

- http://go.sdsu.edu/student_affairs/srr/cheating-plagiarism.aspx
- Cheating
- Plagiarism
- And any other form of academic dishonesty will NOT be tolerated
- I will report any incident directly to the university
- Possible consequences:
  - Expulsion, Suspension, and Probation of Students

- So, please do not engage in any form of academic dishonesty!

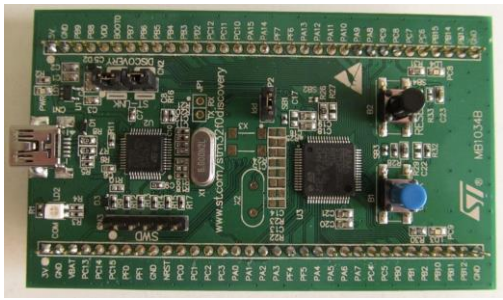
Laboratory Sessions

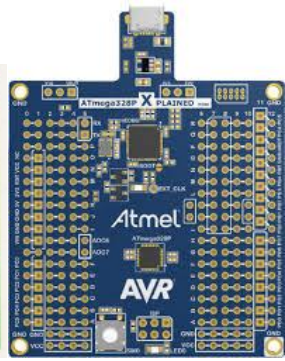
- The first day of the lab sessions: **Feb 4th, 2019 Monday**
- The last day of the lab sessions: **May 6th, 2019 Monday**

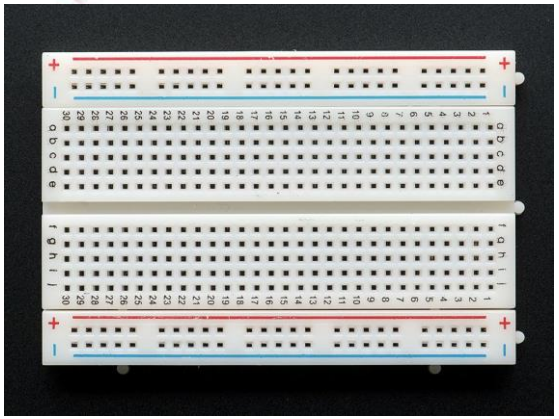
- On the first lab day, you will **receive** your lab kit
- On the last lab day, you will **return** your lab kit
  - If you do not return your lab kit, we will put a hold on your academic account
- Each week, you will receive a new assignment, which is due the following week
  - Except Labs 7, 8, 9, 10, they take two weeks each
- Please see Blackboard for detailed lab schedule

Laboratory Kits

Description	Quantity
STM32Fo-Discovery Board	1
ATmega328P Xplained Mini Platform	1
FT232R USB/Serial Board	1
Pin Strip, 1 x 3 (FOR FT232R PCA)	1
USB-A to USB Mini-B Cable	1
USB-A to USB Micro-B Cable	2
Breadboard Assy.	1
Audio Cable Assy.	1
Keyboard, Hex	1
Photocell	1
Resistor, 1.5K, 1/4W, 5%	1
Trim Pot, 5K, Vertical - Thumbwheel	1
Wire Kit, 24g	1
Female Header, Straight, 1 x 17	2
9" x 12" Zip Top Antistatic Bag	1







Laboratory Assignments

- Every lab assignment is INDIVIDUAL!!
- Lab attendance is mandatory!!
- You will use your kits to show the results for each assignment
- Each assignment will have a pre-assignment quiz to get you prepared for the lab assignment itself
- Each student should submit a report for each assignment
  - A brief paragraph describing the assignment and student's solution
  - The important code snippets from the assignment that the student implemented
  - Will be submitted through Blackboard
- We will look for how you perform in the pre-assignment, the correctness of your solution, how well you can show it to the lab instructor (demo) and your written report
- Labs 1, 2: **2%**
- Labs 3, 4, 5, 6, 8, 10: **3%**
- Labs 7, 9: **4%**
- Total: **30% of the class grade**

Lab Assignment Grading Policy

- **Each assignment will be graded as:**
- Pre-lab assignment: 10%
- Checkpoint: 20%
- Correctness: 25%
- Demo: 15%
- Report: 25%
- Attendance: 5%

- Attendance portion will be added to the corresponding lab assignment
- The first lab assignment will not have a pre-lab quiz

Lab Assignment Grading Policy

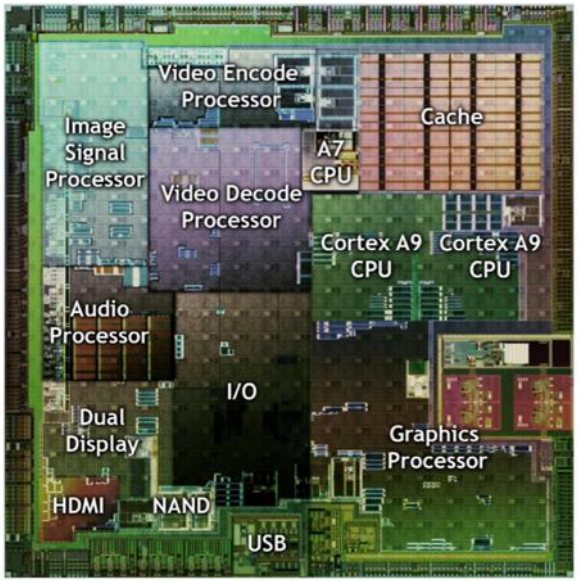
- **Pre-lab assignment:** These assignments will be released before the lab day and you need to submit the assignment in the lab session as hard copy.
- **Checkpoint:** They need to be completed on the same day that the assignment is released (no late submission for the checkpoint part). This checkpoint mechanism will help you use the lab time more effectively.
- **Correctness:** You need to show that your solution works correctly. Also, for this part, we will ask you questions about your implementation. If you cannot answer them, you might not get a full credit for this part.
- **Demo:** You need to demonstrate your implementation.
- **Report:** You need to submit a short report that should include a brief (1 paragraph) description of the assignment and the most important code snippets from the relevant assignment.

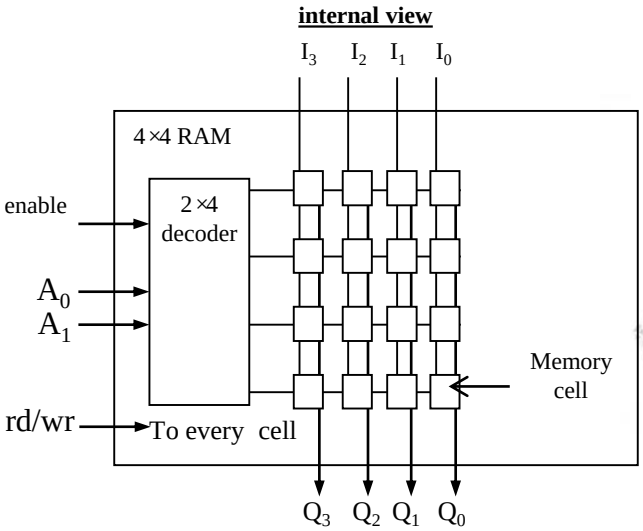
Lab Assignment Grading Policy

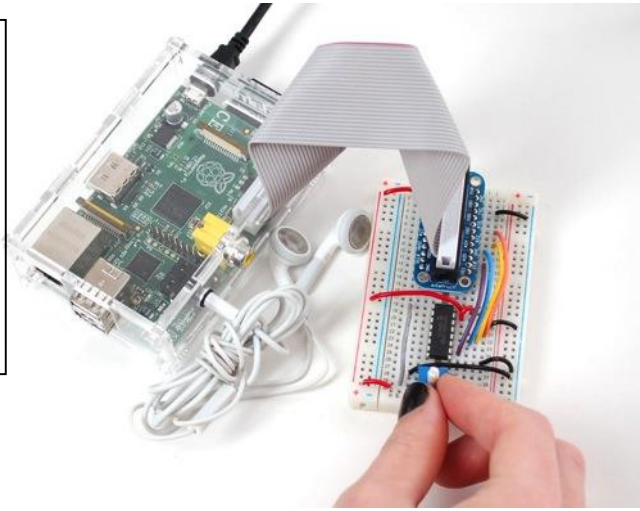
- There is an original deadline for each lab assignment, which is the next lab meeting following the release of the assignment
  - Except for Labs 7, 8, 9, 10 which have an additional week time
  - See the Lab schedule in Blackboard for details
- We accept late submissions. However, you will lose 20% for each additional week.
  - For example, the original deadline of Lab 2 assignment is Day 3 (Sep 21st)
  - If you submit your work for Lab 2 on Day 3, you are not penalized
  - If you submit your work for Lab 2 on Day 4, your maximum grade is reduced by 20%
  - If you submit your work for Lab 2 on Day 5, your maximum grade is reduced by 40%
  - And so on
- No lab assignment will be accepted after Day 13 (May 6th)

Topics – Brief Overview

- C programming language
- Instruction set architecture
- Assembly language
- Embedded microprocessor hardware

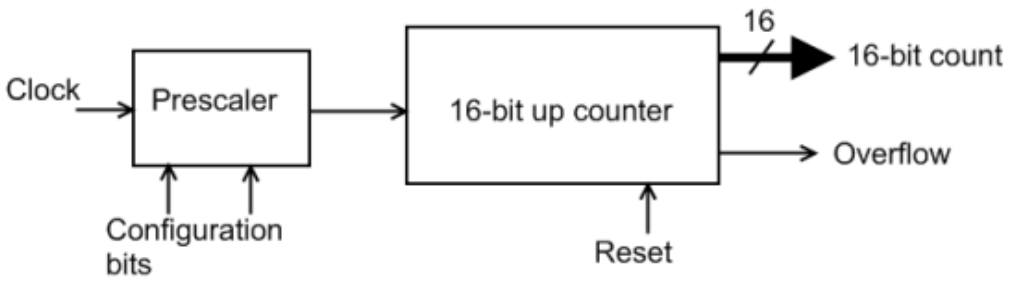


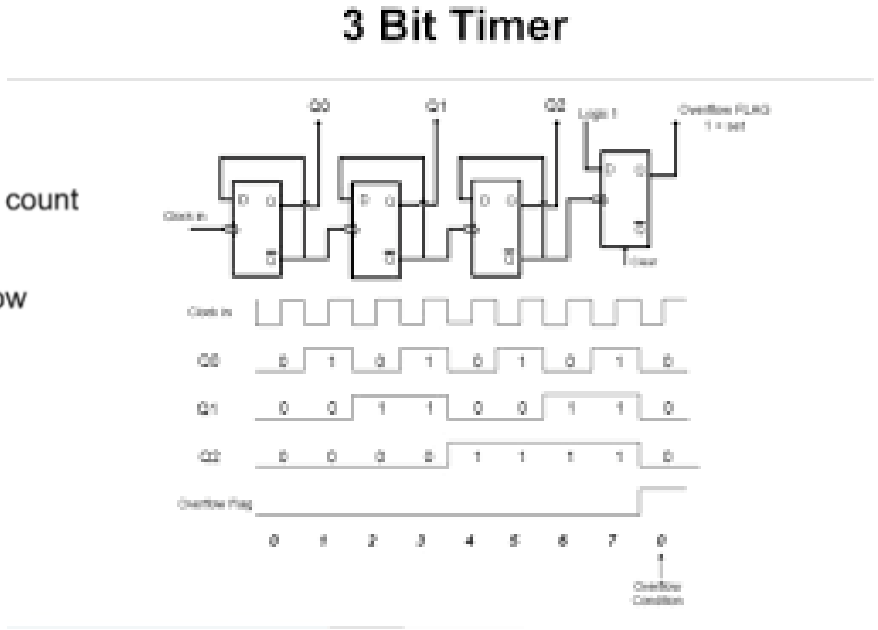




Topics – I/O on a microcontroller

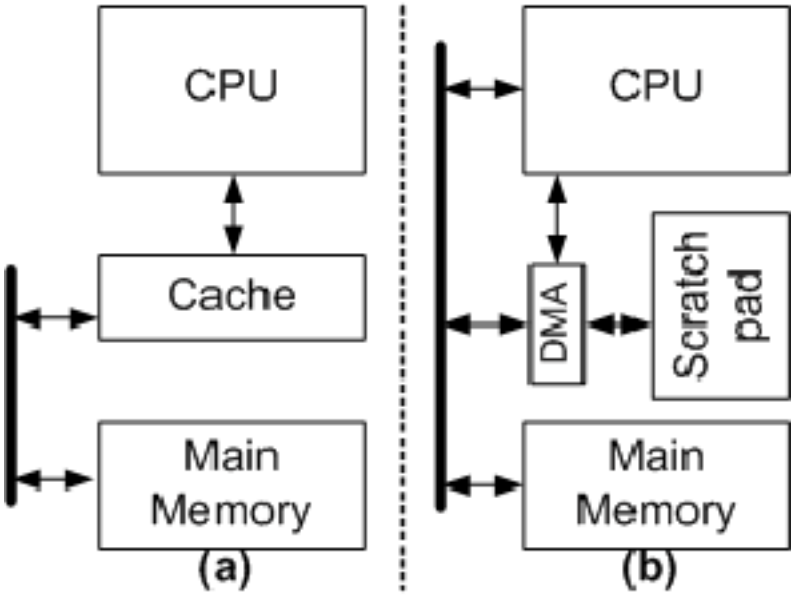
- Serial I/O
- General purpose I/O
- Counters/timers





Topics – Memory/Interrupt Programming

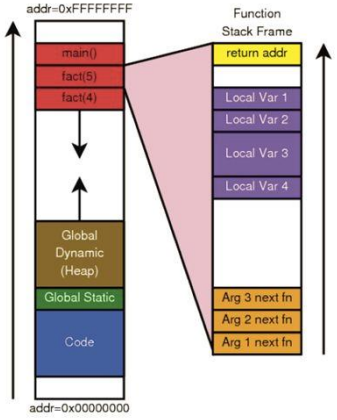
- Embedded memory architectures
- Memory organization and programming
- Real-time and interrupt programming



C Memory Model – where are variables & program code stored in the computer memory?

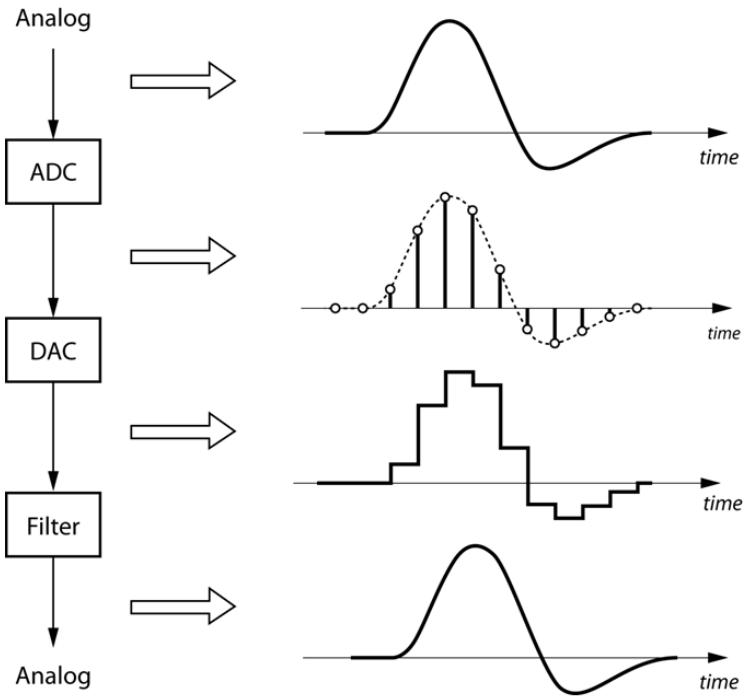
- Program code
- Function variables
  - Arguments
  - Local variables
  - Return location
- Global Variables
  - Statically allocated
  - Dynamically allocated

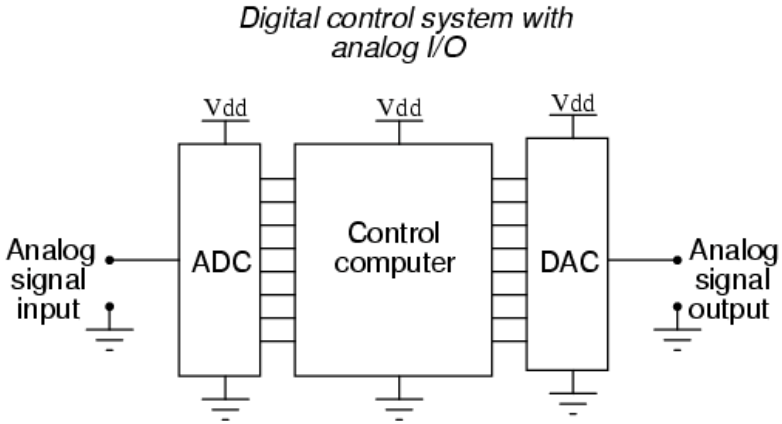
```
int fact (int n)
{
   return(n*fact(n-1));
}

void main() { … fact(5); …}
```

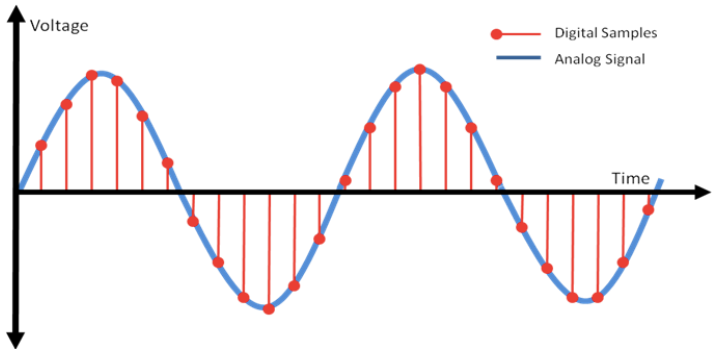


Topics – Analog I/O

- Signals, sampling
- Analog-to-digital, digital-to-analog conversions

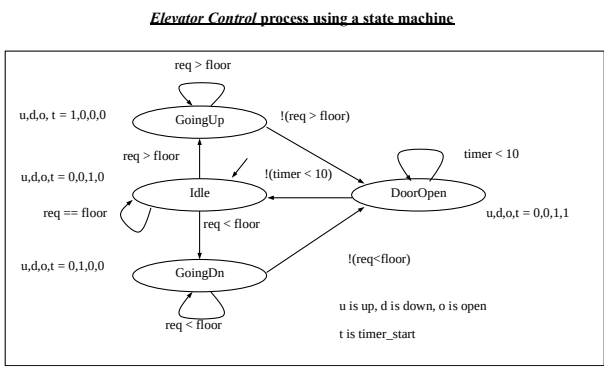


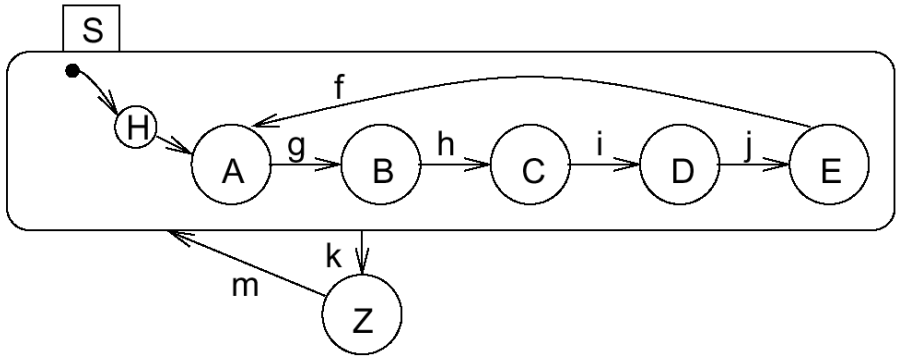


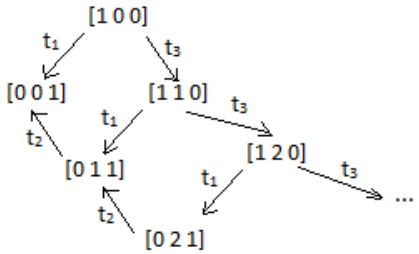


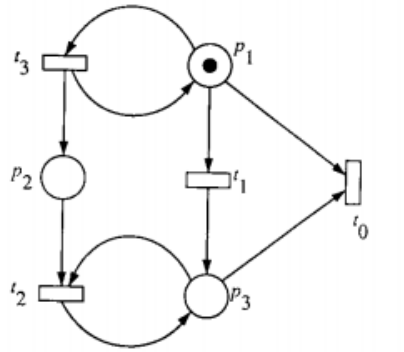
Topics – Embedded Systems Modeling

- Formal modeling for embedded systems: finite state machines, state charts, etc.



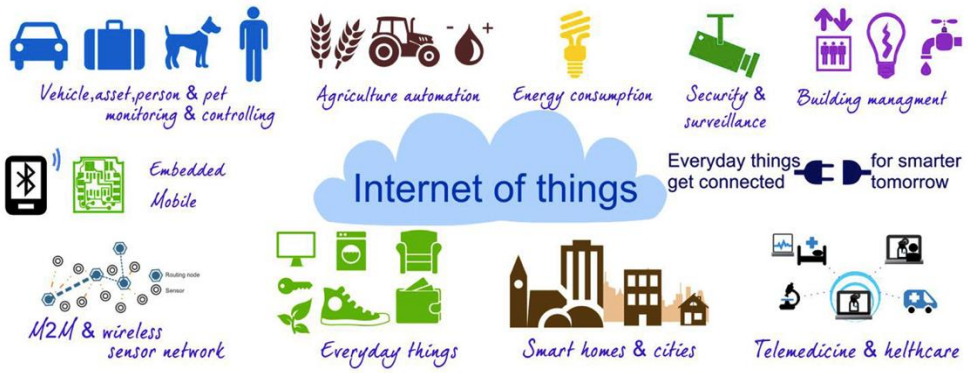


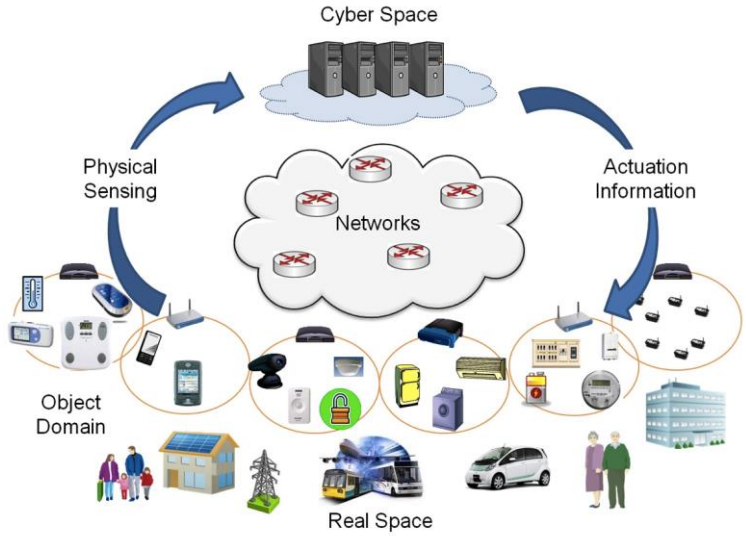




Various Other Topics

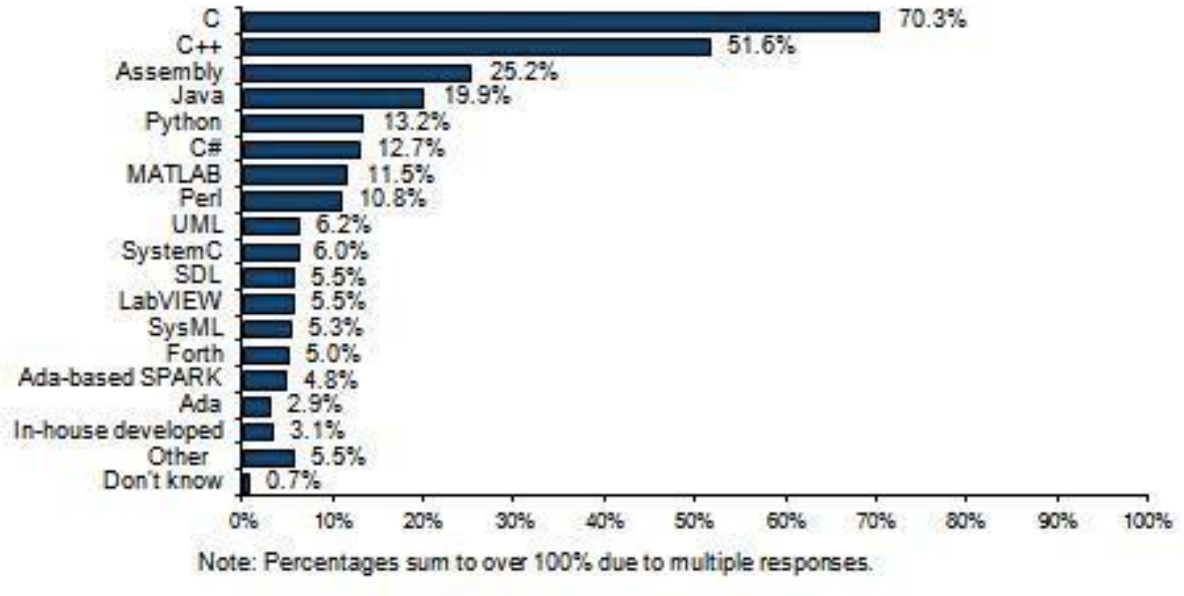
- Embedded control systems
- Sensors/actuators – how to include them in embedded system environments
- Embedded applications
  - Internet of things, cyber physical systems





Embedded Systems Programming

- Points of evaluation
  - Concurrency
  - Ability to specify thread execution times
  - Ability to control shared resources, queues etc.
  - Overhead
- Assembly
- C
- Ada
- Java



Embedded System Environment

- Sensors
  - Collect information
  - Delivers collected information to the system processor
- Real time system processor
  - Process sensor generated information
  - Generates signals for the actuator
- Actuators
  - Acts on signals provided but the system processor

Microprocessor vs. Microcontroller

- Microcontroller chip includes
  - Central processor
  - Program memory
  - Data memory
  - I/O
  - Highly integrated, low cost
- Microprocessor chip includes
  - Central processor only
  - Highest performance
  - Highest cost

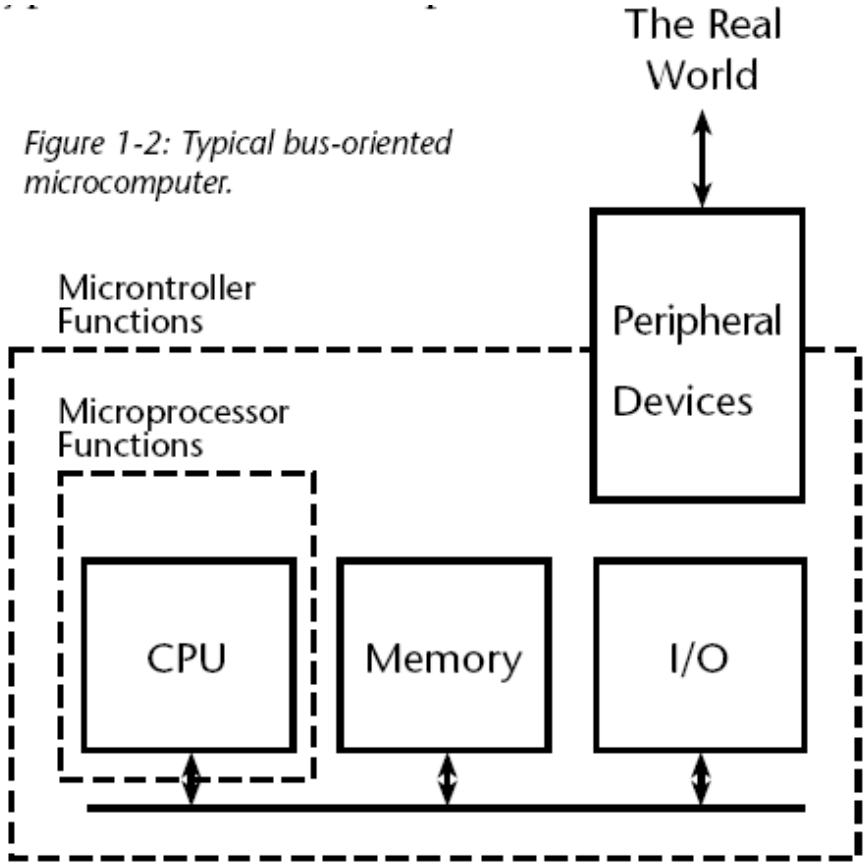


Figure 1-2: Typical bus-oriented microcomputer.

Microcontroller chips

- Advantages:
  - Fewer chips required
  - Lower cost and smaller
  - Lower power (can be power optimized)
  - Fewer connections
  - More I/O pins
  - Higher reliability
- Disadvantages:
  - Reduced flexibility
  - Expansion is limited
  - Limited performance
  - Limited I/O
  - More I/O pins
  - Design compromised to fit everything on one chip