ECE 471 – Embedded Systems
Lecture 1

Vince Weaver

http://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

30 August 2021

Introduction / Syllabus

- Go over syllabus (also found on course website)
- Class notes will be posted on the website.
- Covid concerns

Syllabus – grading

- Homeworks, 50%: 11 total, lowest dropped.
  - Most will involve the Raspberry Pi.
  - Generally will be due on Wednesday by beginning of class. Will have at least a week to do them.
  - Submission by e-mail, grades sent in response to that e-mail, if you don't like that let me know.
  - Will send e-mail when assignment posted on website.
- Midterms, two, 15% total

  Probably in October and mid-November

- Final, 10%, taken remotely
- Class participation, 5%

  Part of this is returning borrowed items at end.
- Project, 20%: Involves using what you learned to do a small embedded project, with a final writeup and demo the last week of classes. More details as we get closer.

Syllabus – Work

- No textbook.
- Late work penalty. I will consider late work, but best to turn in what you have at time.
- Will involve C coding, plus some minimal ARM assembly language and Linux knowledge. I will review everything you need to know.
- Might have some more C instruction this year based on exit-interview feedback. Believe it or not we actually listen.

Syllabus – Academic Honesty

- This has been a problem in the past!
- Do not copy code from other students, either current or from previous years.
- Asking help from the professor/TA is fine
- Asking for general help, or discussing with classmates is fine
- Even having someone look over your code to help find a problem is fine
- Just don't copy someone else's code and submit it as

your own

- Also don't copy code off the internet (again, looking for advice online is fine, but copying code directly is not)

Raspberry Pi

- We will be using a Raspberry Pi.
- See note on course website about what you need
- Brief summary: Model 3B+ is currently probably the best, but any of the models (1A, 1B, 1A+, 1B+, 2B, 3B, 400) should work with the homeworks. No compute node, no pico. Zero probably will work but a bigger pain to use (no Ethernet, no GPIO header).
- You will also need an SD card (8GB or bigger). Older

Pis take the wide ones, newer the narrow ones. Usually not a problem as they tend to come with those adapters. You will want to install Linux (I tend to use Raspbian); getting a card pre-installed with Raspbian or “NOOBS” can save an hour or so of writing the SD card.

- For power you will need a USB-micro cable. You can power from any desktop or laptop (or a 1A or higher USB charger)

- The Pi-4 is usable for this class, but it has a lot of new features which make it a bit harder to use (it used micro-HDMI for output, it needs a USB-C power supply)

Other Accessories

It can be fun to accessorize, but the stuff on the previous page is all you need. Below are some *optional* extras.

- A case can be useful, if only to avoid accidentally shorting out things. Many people get by just fine without one.
- A wall outlet adapter (a USB charger more or less)
- A dedicated GPIO connector to breadboard adapter
- HDMI cable and USB keyboard
- USB serial
- Ethernet cable (or wireless)

Other Hardware

- You will eventually need a breadboard. I know EE/CE students probably already have many already.
- I will loan out various devices/displays when necessary. I'll expect them back at the end of the year so try not to lose them.
- I'll try to hand this out soon as who knows if/when things will get shut down again.

Embedded Systems

What is an embedded system?

- Embedded.
  Inside of something.
- Fixed-purpose.
  Why? You can optimize.
  For cost, power, size, reliability, performance.
- Resource constrained.
  Small CPU, Memory, Disk, I/O, Bandwidth
- Lots of I/O
  For readig sensors (input) or controlling hardware

(output)

- Often real-time constraints.
  Want I/O to happen in guaranteed timeframe.

What are some embedded systems?

Seemingly everything has a computer in it these days. IoT.

- Cellphone (though lines blurring, general purpose)
- Vehicles (Cars/Airplanes)
- Appliances (TVs, Washers, Microwaves)
- Medical Equipment
- Industrial/Factory
- Space Probes
- Video Games?

What Size CPU/Memory?

- Anything from 8-bit/tiny RAM to 32-bit 1GHz 1GB
- Performance has greatly improved over the years. ARM Cortex A9 in an iPad2 scores same on Linpack as an early Cray supercomputer

Type			Speed	RAM	Disk	GPU
Intel	Xeon	64-bit	4GHz	16GB	1TB	Nvidia
ARM	A53	64-bit(?)	1GHz	1GB	8GB	VC4
ARM	M0	32-bit	32MHz	16kB	128kB	none
MOS	6502	8-bit	1MHz	64kB	140kB	none

Discussion

- What concerns might you have when desinging an embedded system?

  Security is a big one these days
- What language might you write your code in?

  C is still popular despite security issues.