

# **ECE 471 – Embedded Systems**

## **Lecture 34**

Vince Weaver

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

`vincent.weaver@maine.edu`

4 December 2020

# Announcements

- HW#10 will be posted
- Presentations are starting
- Feel free to return borrowed hardware.
- Please volunteer to go early if possible, Monday is fairly empty now. Bonus points!



# Final Preview

- Wednesday, December 16th, 12:15pm - 2:15pm, remote  
Like midterms, I'll send out e-mail, e-mail me back  
I'll send out a Zoom link I'll monitor for questions
- Is cumulative for whole class, but concentrates on material from latter half of class. \*No assembly language\*
- Know the definitions of an embedded system and be able to say if a certain machine meets them.



- Know hard/soft/firm realtime
- Know the benefits/downsides of an operating system
- Security/Code Quality – mostly be aware of what things can go wrong if you are not careful when coding
- Embedded busses – know the relative tradeoffs between i2c, spi, and 1-wire. Mostly speed, distance, number of devices
- Power/Performance like HW10



- Give you some C code from one of the homeworks, comment it



# HW#9 – Review



# HW#9 – Code

- C code review
- Do note, it's an LED display not LCD
- Error checking. Be sure not to segfault if dev not there!
- Following a spec?
  - Corner cases
  - should shutdown\_display() clear the display?  
Maybe, makes it harder to grade.
  - Single-digit temps, be sure to remove leading zeros
  - is Zero negative?



- Rounding
- Do you need a . after a three digit temp?
- Left/right justified for single digit
- Reporting error! Must be sure display not printing invalid info! (door on walk-in oven. If it goes from 70F to 1000F (off scale) between readings, don't want it to stay at 70F, you want ERR or HOT or some way to notify something is wrong) More realistically, probe wire broke, should it just report last reading? Or maybe go blank?





# HW#9 – Questions

- Test inputs: try to have one in each case. Also might be nice to check each digit 0..9 to make sure those are all printing well.
- List an \*example\* of poorly written embedded code.
- Why write good code?  
Cut-and-pasting, good practice, among other reasons.
- Why is touch useful? force make to rebuild  
For some reason a number of assignments somehow didn't have updated time in the tar file.



- 2038 problem

Time in Linux is seconds since 1-1-1970. Not a problem 64-bit machines, but overflows in 2038 for 32-bit. Can avoid with a 64-bit system or else a specially patched Linux system

\* discuss y2k problem \*\* worst problem year 19100 on websites

- ctime – last status (metadata) change (originally create time) things like permissions change, ownership change, rename



mtime – last modified

atime – last access

- In stat syscall. stat command. Why atime bad?  
noatime, relatime
- utime() used by touch. Cannot change ctime, set to current time
- why not believe timestamp? maybe could look at ctime.  
also set clock back if own machine.  
HW assignment at Cornell

