ECE 471 – Embedded Systems
Lecture 21

Vince Weaver

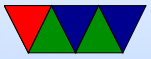
https://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

28 October 2024

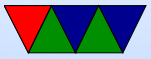
Announcements

- Hand back and go over midterm. Average was 88
- Don't forget HW#7. Stop by to pick up parts if you haven't already.



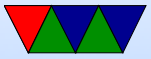
HW#7 – C Floating Point Notes

- You might not have used floating point in C much. Briefly covered in ECE271
- 32-bit integer only lets you have values from -2,147,483,648 to 2,147,483,647
- What if you want bigger? Smaller? Fractional / decimals?



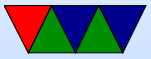
HW#7 – Floating Point

- Also gone over in ECE271
- For a 32 bit value, have one bit for sign, X bits for exponent, X bits for mantissa. How this actually works a bit beyond this class
- Can have a much wider range of values, also things like infinity, NaN, etc.
- Can do this all in hardware. Many low-end embedded systems these days will have support (all the Pis do)



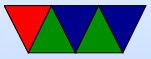
HW#7 – What do you do on systems w/o Floating Point

- Many embedded boards do not have it
- The boards in ECE271 due because Hummels uses them for DSP class
- All Raspberry Pis do (it's why they are compiled ARMHF – HF = hardware floating point)
- You can have either the operating system or the compiler "emulate" floating point in software. It's slow
- Alternatively you can use fixed point math instead



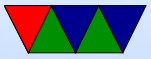
HW#7 – Fixed Point

- With integer math you can have "fixed point" where you arbitrarily put the decimal point at a fixed place in the value.
- 16.16 means 16 bits integer part, 16 bits fractional
- add/subtract work as normal, have to adjust after multiply/divide
- You might use this on an embedded system w/o hardware floating point



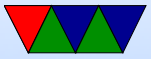
HW#7 – Arbitrary Precision Math

- You can have libraries that dynamically allocate memory for arbitrarily large numbers and fractions
- This is overkill for normal math you might want to do



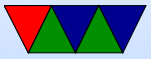
Floating Point in C

- `float` is 32-bit
- `double` is 64-bit
-
- C will auto-cast things so you can do things like this, but be careful as sometimes the rules are a bit obscure. Generally when converting float to integer it will truncate rather than round
- Converting int to floating point:



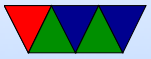
```
int value=45;
double temp;

temp=value;                    // works, but truncates
temp=(float)value;             // casts make the conver
                               // but can potentially h
```

- `float` vs `double`
  float is 32-bit, double 64-bit
- Constants 9/5 vs 9.0/5.0
  The first is an integer so just "1". Second is expected 1.8.
- Printing. First prints a double. Second prints a double with only 2 digits after decimal.



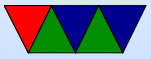
```
printf("%f\n",temp);        // print 32-bit float
printf("%lf\n",temp);       // print 64-bit double
printf("%.2lf\n",temp);     // print double with 2
```

- Explicit converting float/double to integer (rather than cast)
  - floor()
  - ceil()
  - round()
  - rint()
  - nearbyint()

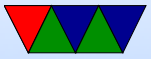


HW#7 – Floating Point Pitfalls

- Sort of like in decimal where only fractions of 1/2 and 1/5 have exact values (because base 10) on binary computers only multiples of 1/2 are exact
- This means unexpected things happen, like if you add 1.5 and 1.5 you might not get 3.0 but something close like 2.9999999999999943
- That will round normally in say printf() but if you do an implicit cast to an integer you might confusingly get 2 instead of 3

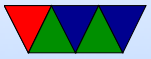


- You can use functions like `round()` and `ceil()` and `floor()` to do proper rounding
- You can force constants to be double/floats by putting a decimal point, so `9.0/5.` will be 1.8 like expected but 9/5 will be truncated to 1 (using integer math)



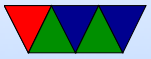
More SPI bits

- For devices needing higher bandwidth, often flash or RAM of some sort
- Dual SPI – half duplex. Uses MISO instead to be second data line, send two bits at once to double bandwidth
- Quad SPI – half duplex, adds two extra lines to send 4 bits per clock cycle
- OctoSPI – 8 bits at once?
- HexideciSPI (HSPI) and XSPI – even move, not sure if STM is just making things up

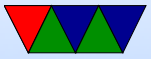


Other SPI variants

- Safe SPI –stricter standard used in automotive applications
- three wire – data one way, only 3 wires
- microwire – half duplex
- Jtag



System Booting



Boot Firmware

Provides booting, configuration/setup, sometimes provides rudimentary hardware access routines.
Kernel developers like to complain about firmware authors.
Often mysterious bugs, only tested under Windows, etc.

- BIOS – legacy 16-bit interface on x86 machines
- UEFI – Unified Extensible Firmware Interface
  ia64, x86, ARM. From Intel. Replaces BIOS
- OpenFirmware – old macs, SPARC
- LinuxBIOS