EENG 260 Lab 4 Report

Ian Kaiserman

This lab was much more comfortable being in a higher-level language than assembly, especially since a lot of my programming experience has been in Java so far, and C function calls are very similar to how it is in Java. The more difficult and tedious part is figuring out what libraries need to be included, especially with trying to navigate the TivaWare library files and documentation to make sure I don’t get dozens of errors.

I got so used to doing the various bit manipulations in assembly that it was weird having to instead navigate function documentation that can do all the work for me. For example, instead of needing to set a bit value to enable a port, actually enable the port, then do the same things for setting several ports to inputs or outputs, I can just call a function that says what it does and use | (or) with actual pin names. Or, instead of needing to worry about the 2-bit offset system for pin writing, I can just call the simple write function and throw the same types of variable names in.

Oddly enough, it almost felt like it took more lines of code in C than it would’ve taken in assembly, I’m not sure if that’s actually true in perspective of the whole project, or if it’s possibly because separate function calls have to be set for different port configurations for example, but in C it’s definitely much more readable, understandable, and easier to manipulate. I don’t have to trace through what a register is set to or pointing to, figure out if I’m safe to change it, or if it’s what I’m looking for in debugging, having to track the registers in the debug to figure that out, etc.

It’s also really nice to be able to use hex WITHIN the instructions I’m writing. Since we’ve used hex so much in the various labs, I’ve gotten used to doing the mental conversions of what it equals in decimal/binary, doing the conversions to what bits it sets, etc. This was especially useful when doing the SysTick configuration since I was used to the values we used converted to hex for things like the reload.