Lab 7 README

ECE 13

Sean Manger

Prof: Steve McGuire

F24

Author: <Sean Manger> (smanger@ucsc.edu)

Collaborators: For this lab, I again collaborated with Donny Tang. The primary thing I discussed with him was not being able to get my LEDs to turn on in the first part of the lab. This ended up being caused by me placing my flag update 1 line above where it should be. I learned this after consulting with Steve who informed me that what I was experiencing was a "Race Condition" meaning the ISR wasn't returning before my if(event) statement was evaluated leading to an error. I probably spent about 12 hours tracking down this issue as Donny wasn't sure what was happening. In addition to Donny, I also sought out Ruhai's help during tutoring briefly to see how I could make the LEDs in oscillate in Part 2 more smoothly. Ruhai's advice was to play with the LED functions to state return value until I found value until I found something that worked for me. This ended up working great and I got desirable results. Other than this all code is as always, my own original work.

Introduction:

The goal of this lab was to learn how to do input and output on our Nucleo boards using the attached hardware peripherals such as the LEDs, Analog-to-Digital converter, and 4 standard push buttons. Up until this point in the course we have simply just been interacting with our programs via the terminal. However, it is here that we finally get to use code to manipulate objects in the real world. This was an incredibly rewarding experience, and it is a lot different seeing physical objects respond in real time to your code as opposed to simply mashing buttons on the keyboard. To start off with this lab, I made sure to read and reread the Lab manual several times since Steve warned us this lab would be a step up in difficulty due to the hardware elements. Ultimately, it wasn’t too bad, and I was able to implement all parts of the lab except Part V and the Extra Credit due to running out of time.

Part I

The most difficult part of this section of the lab was dealing with the Race Condition error as I previously mentioned. Other than that, I found it quite simple. I had previously worked with LEDs before at community college using Arduino.

Part II

Part II was just an extension of part I but using switches to change the speed of the LED. I found implementing the oscillation logic simple and with Ruhai’s input I got it working great. I didn’t find the switches too difficult to implement after scouring their respective header file and the common directory for definitions and setup information.

Part III

In my opinion this was the simplest part of this lab. We were given starter code for reading ADC values and pseudocode for a rough guideline of what to do. After I figured out the logic of the window it proceeded smoothly. The lab manual suggests that we should be able to eliminate all the noise off of the ADC . However, I didn't find this to be the case since while I was able to eliminate 90+ % of the noise there was still some renaming. In the end, I was able to successfully implement this part. I was not however able to test it using my OLED screen since it turns out it is not functioning and gives an error indicating this fact: I2C Tx.

Part IV

I probably spent about an hour on this part just trying to digest each element of how the buttons worked and their respective values. What finally broke the deadlock for me was I sifted through the board.h and board.c file which proved very helpful in seeing how I could check the status of each of the buttons individually. It turned out that it was just a simple & operation with the Button positions of each of the buttons (1, 2, 4, 8). The only major change from the LEDs was that I had to account for the previous state of the button but this just ended up being checking an extra condition within the if statement and creating a variable to take in the previous state of the button. In the end, my buttons all worked as expected within the test harness. Overall, I probably spent 3 or so hours on this part of the lab. I thought the test harness we were given to implement was fairly simple and I accomplished it though some if and print statements.

This was the last part I accomplished due to running out of time. However, given that my buttons library was fully working I would imagine Part V would not have taken very long to implement.

Conclusion

I thought that this was another excellent lab that I probably spent about 20 -25 hours working on total. If I were to offer some improvements I would like to see more material related to clearing flags, race conditions, and how buttons are defined within the STM32 since these were the parts I struggled with the most. Other than that, I again thought that this was an excellent lab that further grew my skills both in C and as a programmer. I am looking forward to the toaster lab and I feel I have laid and excellent foundation for it.