Lab 3 Report | Hao Li | cs425 | 3-30-21

The code is in the Github repository that I shared to you last time.
https://github.com/hl723/cs425

Part 0

For this part, I did not use any unsafe features. Since this is the first program I made using Rust, the major difficulty was learning the semantics and making it handle concurrency. Overall, this part wasn't too bad since the most of my time was spent on learning how cargo worked and making the counter thread-safe.

Parts 1 and 2

Unlike doing these parts in C, I knew exactly what I needed to do (what bits to set and to where and in what order). However, implementing such simple tasks in C became a much more difficult program to write in Rust. I would want to say the majority of my time was spent on two tasks.The first of which was searching for the crate manuals and finding the necessary abstraction function to use to modify the register that I want. The second major timesink was simply battling the compiler to resolve errors that I never seen before (reminds me of when I was learning C). However, the assignment was much easier to finish once I got one of the peripherals to work, since if I got the user button to work, then making PC7 work is virtually following the same process.

For unsafe features, the only block of unsafe code I have is unmasking the interrupts for NVIC and setting the priorities for them. I am not sure if this is avoidable since these functions are unsafe functions. The rest of the code is written in safe Rust.

Explanation of each line of code

Line 1: Tells the compiler this does not have a main function.

Line 2: Tells the compiler we do not need the std crate.

Line 4: We use panic_halt to define the panic behavior

Line 6: We import the basic board support package for the board.

Line 8-12: We further import the stm32 crate (many of the peripheral definitions and useful functions), all the traits of the crate (prelude), and the original author's defined LedColor and Leds structs.

Line 14: We get the cortex-m peripherals crate.

Line 16: Defines the entry attribute.

Line 18: The entry attribute to signal where to start the program.