Lab 07 Toaster Oven Lab README

Jayant Dharwadkar

This lab introduced the concept of finite state machines as a tool for programming reactive systems like embedded controllers. We implemented a toaster oven simulator on the PIC32 microcontroller by following a provided state machine diagram. The key aspects were using a state machine to control the logic flow based on input events like buttons and a potentiometer, updating an OLED display with the current oven state, and driving LEDs to visualize the remaining cook time.

I first carefully read through the lab manual and tried to understand the state machine diagram before starting to code. The bounce state machine example helped cement the fundamentals. I then followed the suggested approach of implementing the OLED updater function first to test displaying the oven state. From there, I implemented the state transitions one at a time, testing each thoroughly with printf statements. A challenging part was keeping track of all the different oven data like mode, time, temperature, etc. across states. I had the most trouble putting the state machine together. It was too complicated to put it together initially so what I ended up doing was making helper functions like the lab manual suggested. Overall it was a lot cleaner and easier to understand for me, so I was able to finish it with a few errors.

After around 20 hours of work, I was able to get the core toaster oven functionality implemented. I attempted the extra credit and I believe it works fine. The combined use of events, interrupts, and the reactive state machine made this a very involved lab that really tested my embedded programming skills. I definitely have a greater appreciation for how powerful the state machine model can be for these types of systems.

Some high points were finally understanding how to properly utilize interrupt handlers and the event-driven approach. Low points were the sheer amount of bookkeeping required to update all the oven data correctly, as well as my struggles with the switch logic implementation.

If I could make any suggestions, it would be to include some more guided examples specifically around state machine implementation during the lectures. The lab manual was clear, but those code examples proved crucial for my comprehension. Overall though, this was an excellent lab for drilling these embedded programming techniques. The grading distribution seemed appropriate given how much work was required.