CSE 474 Lab 1

Rocky Zhenxiang Fang

October 22 2020

1 Procedure

1.1 Task 1

1.1.1 Task a

This part of task 1 asked us to control the on-board LEDs in a periodic pattern. The code start from initializing the correct ports and pins according to the datasheet. The important part includes finding the right pointer and understanding the meaning of things to wright in. Figure 1 shows the FSM of the work, where the number indicates the LED and t refers to certain amount of time.

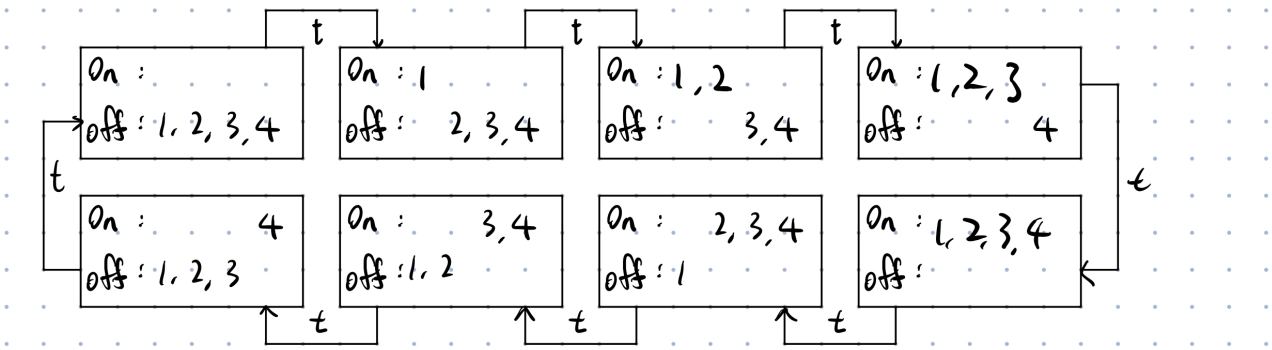


Figure 1: FSM for task1.a

Multiple functions such as SetUp(), SetLEDs(), delay(), TurnOnOneByOne() and TurnOffOneBy-One() are created. Their functions are setting up ports, setting up pins for LED, delay for a certain time using for loop since timer is not taught before, and turn on and off LEDs in periodic order.

1.1.2 Task b

This part of task 1 requires on broad buttons. Unlike LEDs, the board has safety check before accessing the button. Thus, more datasheet reading and port setup needs to be done in this part. Figure 2 shows the FSM of the implementation, while it might be hard to read, the idea is to turn on the LED as long as the corresponding button is been hold.
Besides the functions for setting up components, lightLED() is created to light the LEDs according to the pressed switches. Four conditions are coded in if-else statement to cover all four possible states.

1.2 Task 2

Comparing to previous task, this task requires an external circuit and a well-defined FSM. Figure 3, 4 shows my design before implementing the system. I selected the pin according to suggestion 1 provided by the lab assignment. The implementation of FSM is followed the example from zybook that separates transition and action. Helper functions such as greenOn, gerrnOff, redOn, etc. are created to enhance readability.

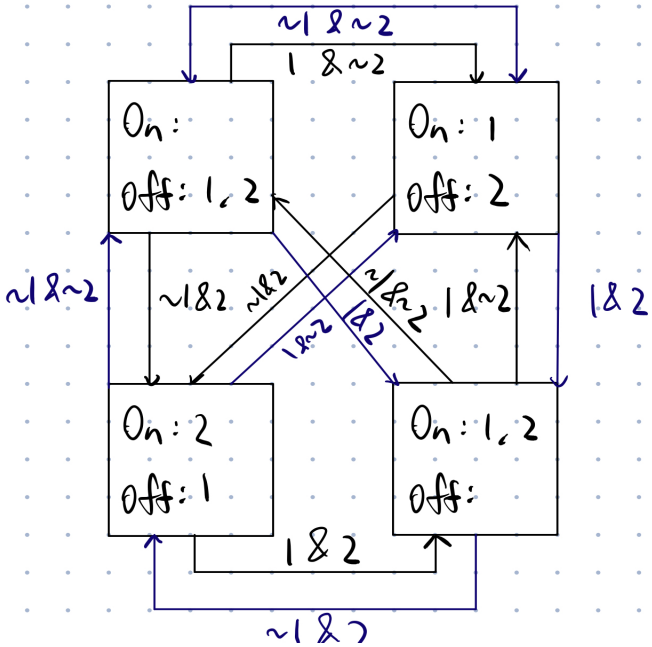


Figure 2: FSM for task1.b

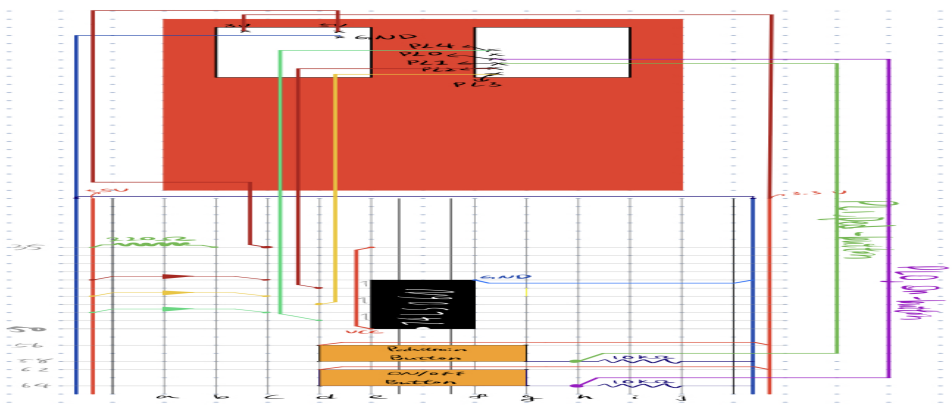


Figure 3: Circuit design for task 2

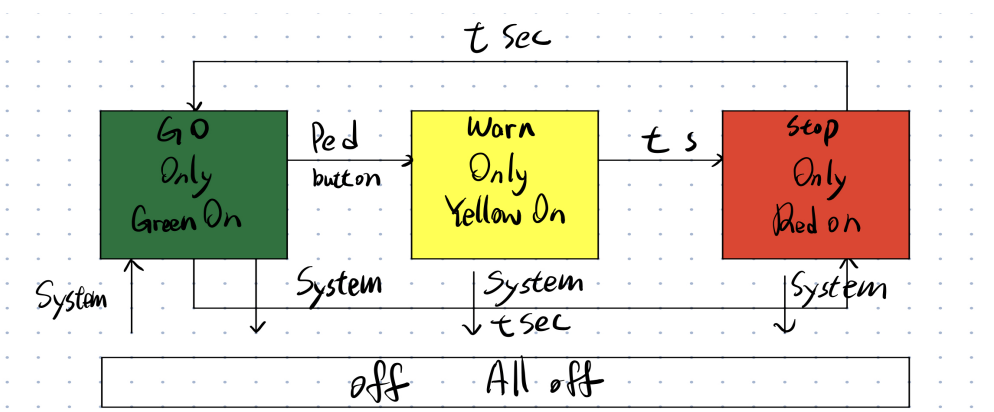
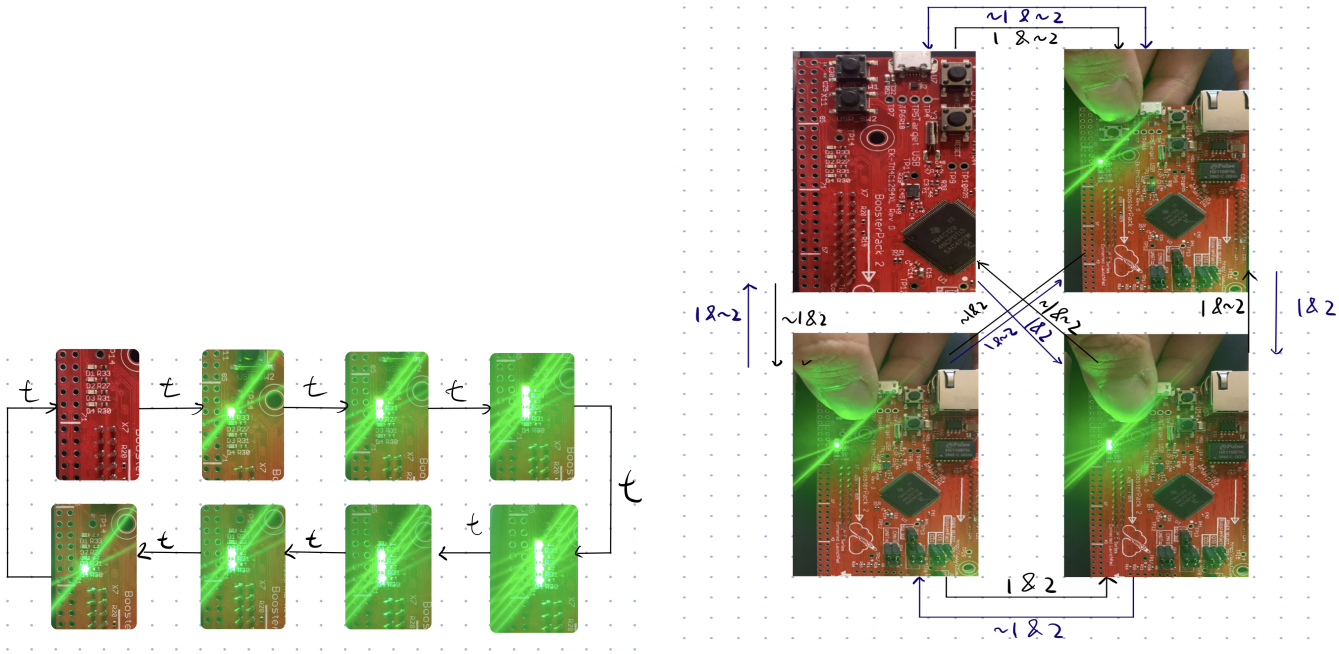


Figure 4: FSM for task 2

2 Results

2.1 Task 1

Figure 5 demonstrates the result of task1, it is displayed as different state so the reader can easily visualize by comparing this figure to the FSM graph listed above.



(a) Result of task1 a

(b) Result of task1 b

Figure 5: Result of task1

2.1.1 Task a

It can be easily seen that the result matches the requirement, which is turn on and off the LEDs in periodic pattern. When the system starts, the LEDs is turned on one by one according to its number. When all LEDs are turned on, then they will start to turn off one by one also according to its number. After all LEDs are turned off, the system loops again. Between each on or off for LEDs, the delay is produced using $for(unsignedlongj = 0; j < 1000000; j++)$ since timer is not introduced yet.

2.1.2 Task b

Like the previous task, the visualization of the system is presented by using pictures. The finger in the picture shows that which button that I was pushing, the the LED lights accordingly.

2.2 Task 2

Figure 6 visualized the result of my implementation of the FSM. As described in the lab assignment, when the system is off, all lights should be turn off, this is showed in the figure. I decided to allow the user to turn off the system in every state and the system will always be in the go state when

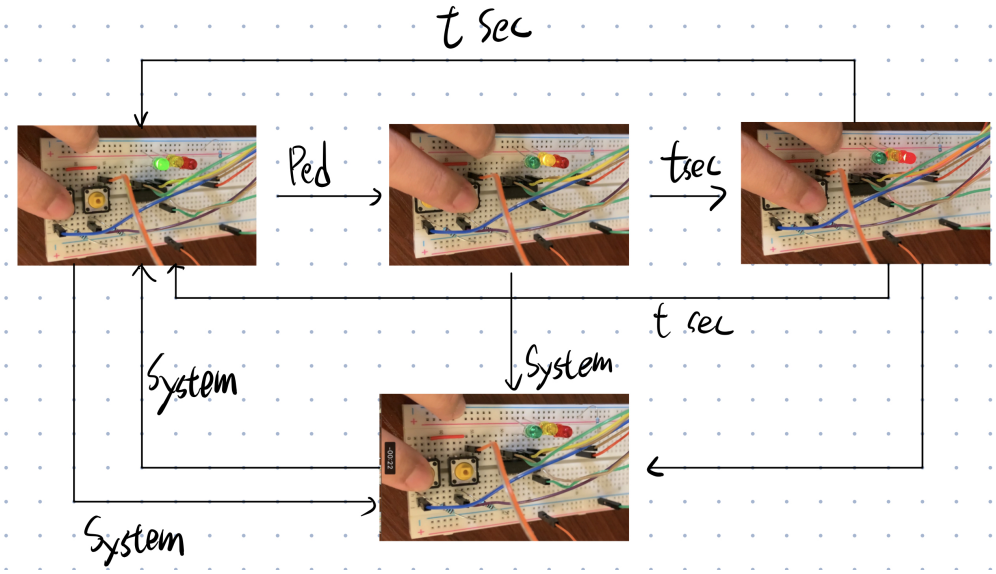


Figure 6: Result of task 2

it is turned on. The *tsec* delay made by using for loop, if a button is pressed during the for loop and the state should change, the state will change accordingly and the loop breaks. The pedestrian button will only be detected in the go state, so if the user pressed pedestrian button during either the warn state or the stop state, nothing will happen, and the for loop continues.