

## EECE 598: Embedded System Design

### Lab 3, Due Date: 10/17

#### 1. Pre-Lab

- Download, unzip, open, compile and run **C2\_TrafficLight.zip** into your computer.
- Read the procedure and prepare your state table and state diagram before you go to your lab session
- Download and install PCBArtist into your computer. The software can be downloaded from [www.4pcb.com/](http://www.4pcb.com/). When installing the application, do not accept any special offers to install third party software. After installation start the PCBArtist application and open the file EECE598\_TM4C123\_artistLAB3
- Edit the file to connect the ports with LEDs and Switches. You can drag the components you will be using closer to each other before connecting.
- Save the file and take a snapshot or save as Pdf, print and bring it with you to the lab.

#### 2. Objective

The objective of this lab is to design and implement a traffic light controller in C, using linked list data structures and the launchPad parallel ports. Also you will learn to use a circuit design tool, PCBArtist, to draw electrical circuits.

#### 3. Overview

In this lab you will be developing a Finite State Machine to control the traffic in the 3-way street shown in Figure 1. The intersection has 3 traffic light signals (S1, S2, S3) each has three lights (Red, Yellow, Green). These signals control the traffic flow on the three roads, road1, road2 and road3. There are three car sensors (SW1, SW2, SW3) placed in each road to indicate the presence of cars waiting. The controller should operate as follows:

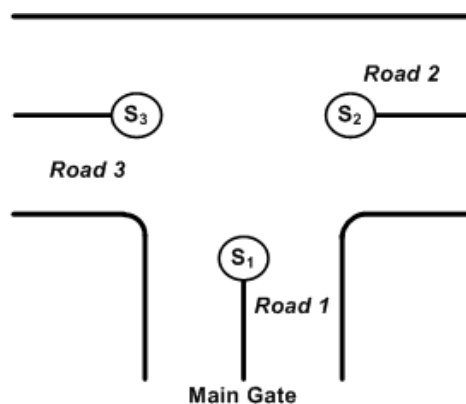


Figure 1

- Traffic coming through road1 (SW1=1, SW2=0, SW3=0), S1=G, S2=R, S3=R.
  - Traffic coming through road2 (SW1=0, SW2=1, SW3=0), S1=R, S2=G, S3=R.
  - Traffic coming through road3 (SW1=0, SW2=0, SW3=1), S1=R, S2=R, S3=G.
  - No traffic on any road (All switches off), gives priority to Main gate road.
  - To change from Red to Green implement a yellow light for 0.5 seconds (wait state for each of the three lights)
  - Green lights should last for 3 sec.
  - If cars are coming in all directions (All Switches are 1), cycle through all the states.
  - If cars are coming in two directions alternated between the corresponding states (e.g. if SW1 and SW2 are ON, alternate between S1 green to S2 green)
- You should avoid going from Green to yellow and back to green for the same traffic light. Figure 2 shows the possible car movement in the 3 directions.

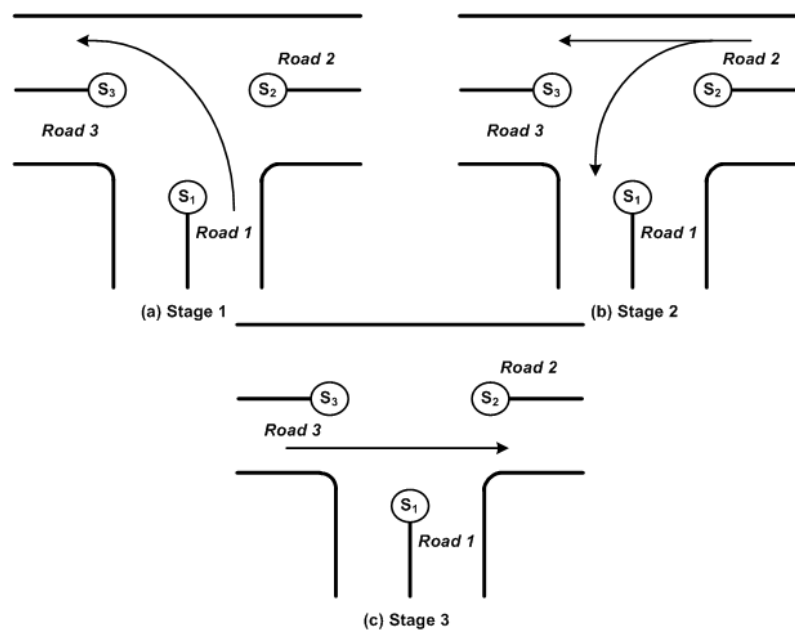


Figure 2

#### 4. Procedure

1. Read the problem overview and use PCBArtist to draw the circuit.
2. There are three inputs: SW1, SW2, SW3 where:
  - a. SW1 ON means a car is waiting on street S1, OFF means no car waiting.
  - b. SW2 ON means a car is waiting on street S2, OFF means no car waiting.
  - c. SW3 ON means a car is waiting on street S3, OFF means no car waiting.
3. Use POPRTE, PIN 2-0 for switch inputs.
4. There are 9 LEDs outputs (Red, Yellow, and Green) for each street light.
5. Use PortE PIN 5-3 (PIN5, PIN4, PIN3) for S1 traffic light.

6. Use PortB PIN 5-3 (PIN5 (Red), PIN4 (Yellow), PIN3 (Green) for S2 traffic light output.
7. Use PortB PIN 2-0 (PIN2 (Red), PIN1 (Yellow), PIN0 (Green) for S3 traffic light output.
8. Determine the number of states needed for the traffic light controller.
9. Create the state table and draw the state machine graph.
10. Create your code using linked structures.

## **5. Demonstration**

When demonstrating the program, you are expected to explain each line of code if asked. Each student in a team will be asked a different question to demonstrate his or her understanding of the project at hand.

## **6. Deliverables**

- 6.1 State machine table and state machine graph
- 6.2 C source code of your main.c program. Your program will be graded for good documentation.
- 6.3 A photo of your built project while displaying different traffic lights on.