Experiment # 12

Traffic Light Controller

Objectives

- Practice on the design of clocked sequential circuits.
- Applications of sequential circuits.

Overview

In this lab you are going to develop a Finite State Machine (FSM) for a traffic light controller that will control the operation of traffic lights at the cross road connection shown in Figure12.1.

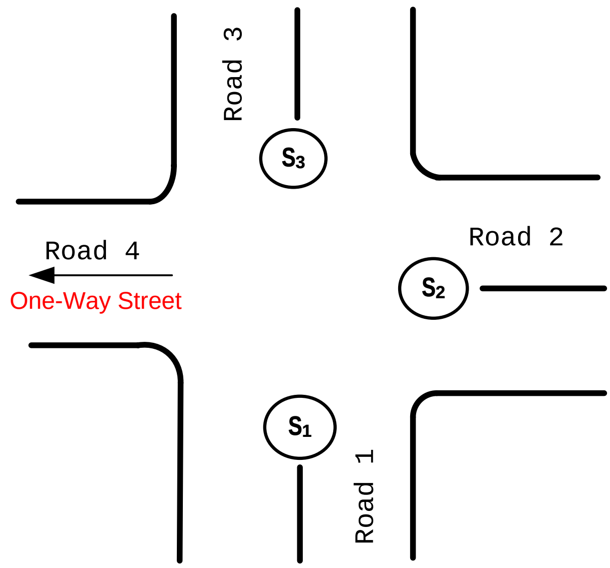

Figure 12.1: The three traffic light signals

Design Specifications

There are three traffic light signals (S_1, S_2, and S_3), each alternating between two states, *RED* and *GREEN*. These signals control the traffic flow on the three roads, *road*$_1$, *road*$_2$, *road*$_3$ in four possible states as follows.

- In *STATE 1*, traffic coming through *road*$_1$ is allowed to go forward or left. (S_1 = GREEN, S_2 = RED, S_3 = RED)
- In *STATE 2*, traffic coming through *road*$_2$ is allowed to go forward or left. (S_1 = RED, S_2 = GREEN, S_3 = RED)
- In *STATE 3*, traffic coming through *road*$_3$ is allowed to go forward or left. (S_1 = RED, S_2 = RED, S_3 = GREEN)
- In *STATE 4:* no traffic coming from any road is allowed (All signals are RED)

Road 4 is a one-way street. No traffic can possibly come out from *road*$_4$.
States 1 – 3 are enumerated in Figure 12.2

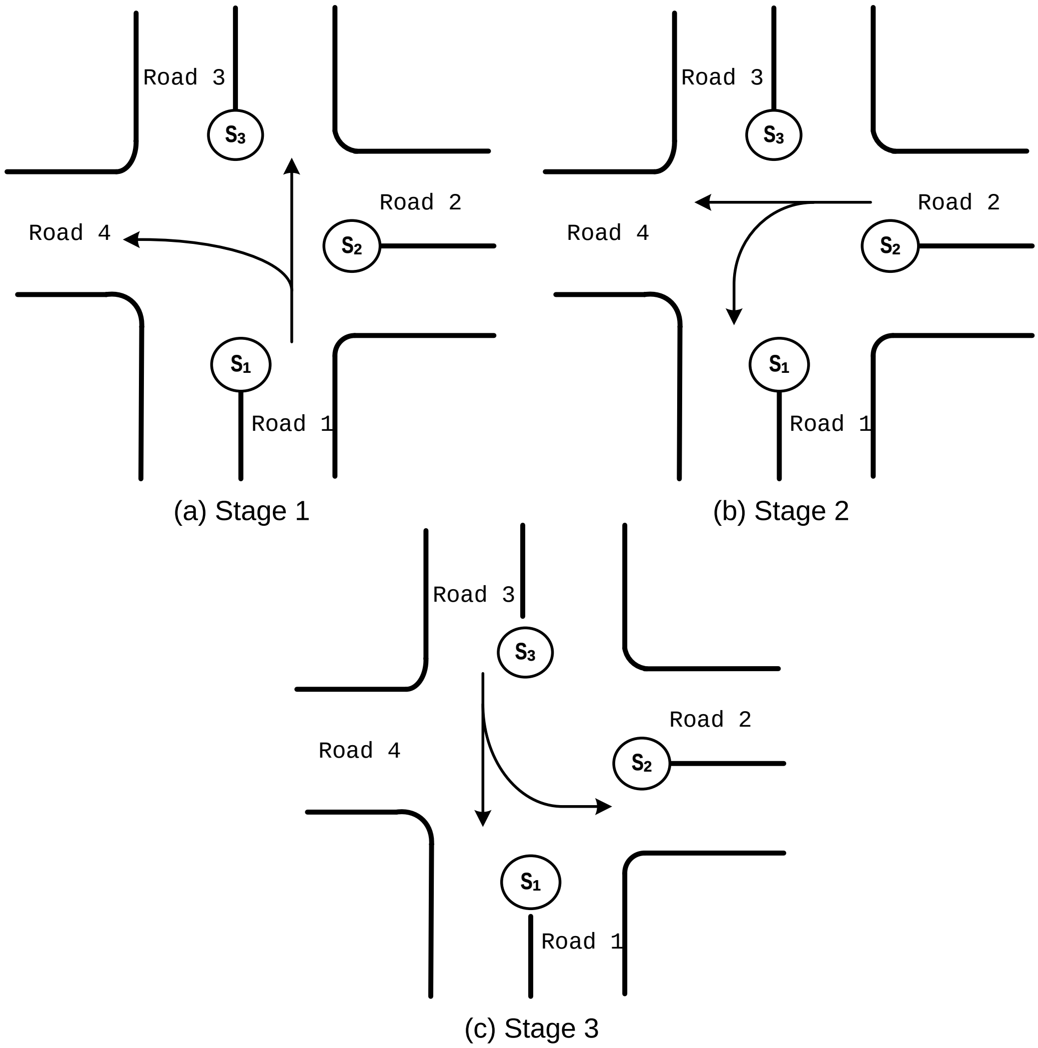

Figure 12.2: Traffic states

The operation of the three light signals (S1-S3) is controlled through an arrangement of *traffic sensors* and *traffic light controller* circuit as shown in Figure 12.3**.** There are three traffic sensors x_1, x_2 & x_3, which sense the presence of traffic on the roads 1-3 as illustrated in Table 12.1. The controller operation is determined by the output of these three sensors as enumerated in Table 12.2.

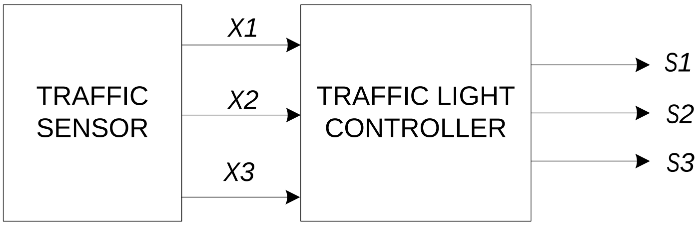

Figure 12.3: Traffic Sensor and Traffic Light Controller Circuit

Table 12.1: Traffic Sensor Signals

$x_3\,x_2\,x_1$	**Indication**
0 0 0	All roads have no traffic
0 0 1	***Road*$_2$** and ***Road*$_3$** have no traffic
0 1 0	***Road*$_1$** and ***Road*$_3$** have no traffic
0 1 1	***Road*$_1$** and ***Road*$_2$** have traffic
1 0 0	***Road*$_3$** has traffic
1 0 1	***Road*$_2$** has no traffic
1 1 0	***Road*$_1$** and has no traffic
1 1 1	All roads have traffic

Table 12.2: Traffic Sensors and Controller Operation	
$x_3\,x_2\,x_1$	**Indication**
0 0 0	Stay at ***STATE 4***
0 0 1	Stay at ***STATE 1***
0 1 0	Stay at ***STATE 2***
0 1 1	Alternate between ***STATE 1*** and ***STATE 2***
1 0 0	Stay at ***STATE 3***
1 0 1	Alternate between ***STATE 1*** and ***STATE 3***
1 1 0	Alternate between ***STATE 2*** and ***STATE 3***
1 1 1	Normal operation: ***STATE 1, STATE 2, STATE 3, STATE 1, …..***

The design of the traffic controller module requires using D-flip flops with asynchronous clear. Assume that the controller has three inputs: x_3, x_2, x_1 coming from the traffic sensor, and three outputs: S_1, S_2, and S_3, which control the operation of the three traffic light signals (logic 1 represents a GREEN signal and logic 0 represents a RED signal).

Pre-Lab

- You are expected to know the design procedure of synchronous sequential circuit.
- Obtain either a state diagram or a state table for the circuit. In this case, it would be appropriate to get the state table directly (use the table below).
- Derive the flip-flop input equations from the state table.
- Derive output equations for the traffic signals.

	x_3 x_2 x_1 A B	A^+/D_A B^+/D_B	S_1 S_2 S_3
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			
28			
29			
30			
31			

In-Lab

- In a new project, open a new schematic sheet and implement the functions that you have obtained in the pre-lab.
- Use D-Flip flops with Asynchronous Clear from the library.
- Constrain the inputs of your circuit to 3 of the level switches (SWs) and the outputs to 3 LEDs. It is also preferable to constrain flip-flop outputs to LEDs.
- Use 2 BTNs, one to provide a clock signal (*Beware of switch debouncing problems*) and the other as an asynchronous clear. Connect these inputs to the respective inputs of the flip-flops.
- Check your circuit for errors and fix them.
- Simulate your design.
- Download your design.
- Verify its functionality by applying different input combinations and compare it with your state table. Demonstrate your work to the instructor.

Post-Lab

- Document your design by providing a diagram of your circuit

- As a bonus propose a modification to your design that will take the following into account: If the circuit is alternating between different states (for example if $x_3\ x_2\ x_1 = 1\ 1\ 1$), each state should remain no more than 10 sec before moving to the next state and so on.