

# LABORATORY EXERCISE 4

## A Traffic Light FSM

The purpose of this exercise is to practice the creation of finite state machines. By the end of this lab you should have created an FSM for a traffic light controller.

### BACKGROUND

Figure 1 presents a state diagram for a traffic light system. The traffic light system has only two lights: one in the North-South direction (NS), and the other in the East-West direction (EW) (see Figure 2). In this traffic light system, only one of these lights is on at a time. Hence, we are either in the state NS in which case we turn on the lights corresponding to NS, or we are in the EW state. As in all cases, there is one more state, called the reset state.

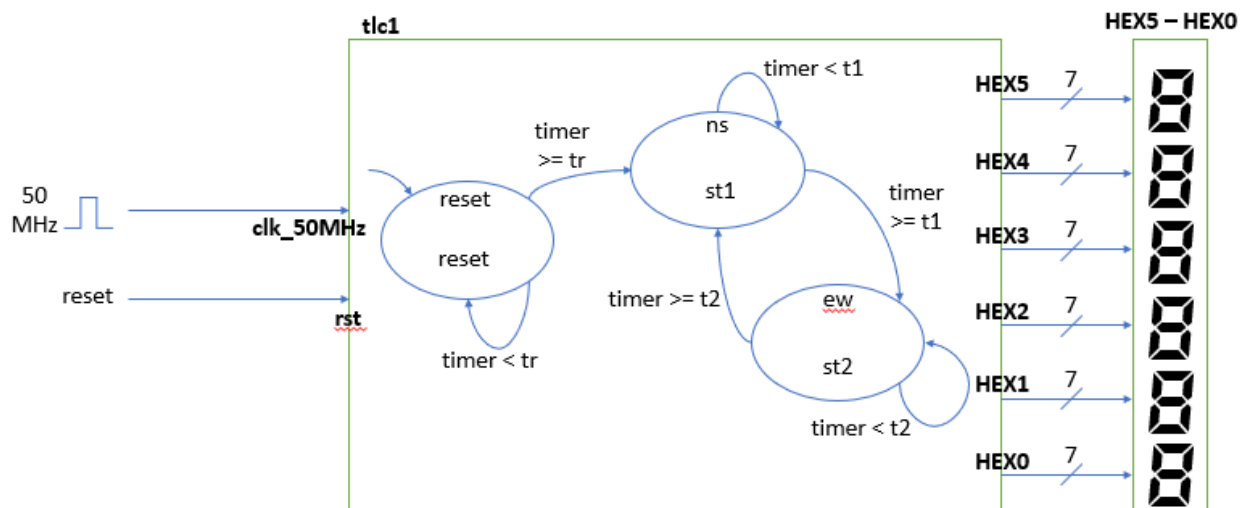


Figure 1

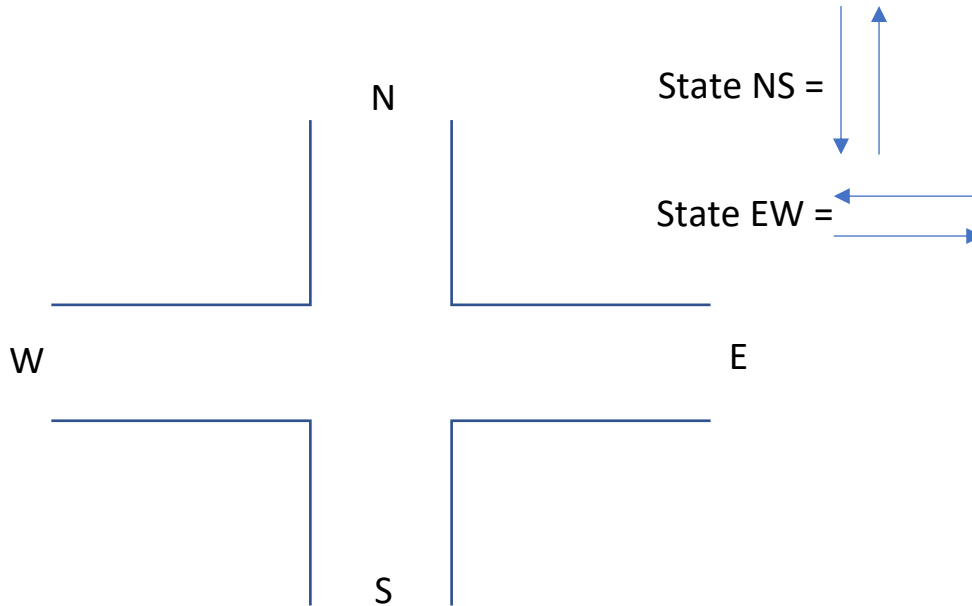


Figure 2

A reasonably finished VHDL code is provided to you in lab3.zip. In the tlc1.vhd file you will find code that implements the state machine in figure 1. Comments have been included in the file to enable proper understanding of it. A 3 process fsm is used. The reason for this is that if you decide to change the output logic, you can just modify the *output\_logic* process and not mess with the other two processes. To change the rules that determine the next state, you only need to modify the *state\_transition\_logic* process. In this three-process setup, the first process i.e. the *sync\_state\_transition* process will typically always be the exact same. All that process does is to transition from one state to another on every rising edge of the clock.

## PART 1

Modify the code you are given so that when in the NS state the display is “NSxxxx”, instead of “xst1xx” as it does at present, and when in the EW state, the display is “EWxxx”, and instead of “xst2xx”. Please note that

1. “x” here is used to represent a blank a HEX display. Hence NSxxxx means that HEX5 and HEX4 will display N and S respectively, and HEX3 to HEX0 will be off.
2. EWxxx has just 3 blank HEX displays because you will write W using two HEX displays
3. You will not be able to write uppercase N. Use lowercase n instead.

## PART 2

Modify the code further so that the total list of states will now be

- Reset
- North-south green

- North-south amber
- East-west green
- East-west amber

Instructions:

1. Create a new project called tlc2 with tlc2 as the top-level entity.
2. Draw a state diagram for this new FSM. You are to submit it with your Quartus project.
3. Determine what the state transition logics will be.
4. In each state, display whatever you feel is appropriate on the HEX displays.
5. Implement your design in VHDL as the entity called tlc2.

## PART 3

Modify the code in part 2 to include six LEDs as outputs.

Let LED0 be North-South Red light

LED1 be North-South Amber light

LED2 be North-South Green light

LED5 be East-West Red light

LED6 be East-West Amber light

LED7 be East-West Green light

Now, turn on the appropriate LEDs in the correct states.

## PART 4

Modify the tlc1 or tlc2 entity to include KEY1 on the DE10-Lite board as a pedestrian button. Whenever the button is pressed, let the FSM transition into a new state called pedestrian after 2 seconds. Let it then stay in the pedestrian state for 4 seconds and then return to whatever state you decide to return it to. Note, you are at liberty to implement this pedestrian on the tlc1 entity or the tlc2 entity.