ECE 465: Digital System design
Summer 2021

Optional Project 3: Traffic Light Controller

Due: August 06, 2021

Design a traffic light controller with the following specifications:
A busy highway is intersected by a little-used farm-road. Detectors are placed along the farm-road to raise the signal C as long as a vehicle is waiting to cross the highway.
The traffic light controller should operate as follows:

- If vehicle is detected on the farm-road, the highway light should change from green to yellow to red allowing the farm light to change to green.
- The farm-road light should stay green as long as there is a vehicle waiting or a set-time interval of time (TL) has not expired.
- Even if vehicles are waiting on the farm-road, the green light on the highway should remain on for a set interval of time (TL).
- Take time interval (TL) to be 5 clock cycles.
- The yellow lights (HWY, FRY) should be high for a single clock cycle when the signal turns from green → yellow → red.

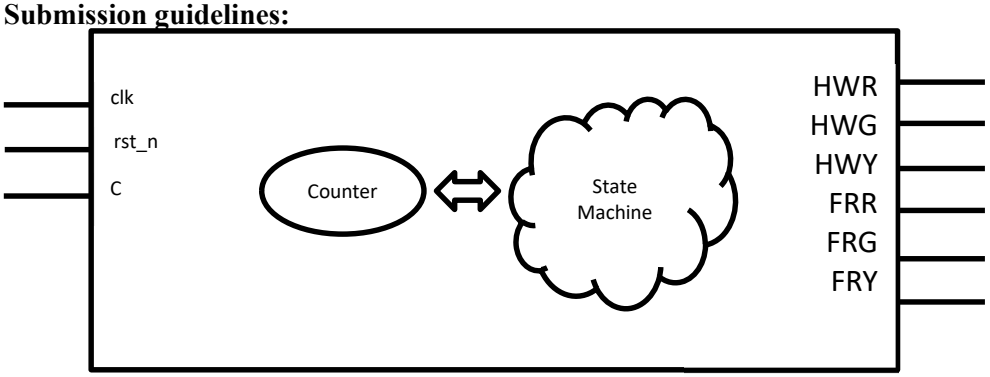
The inputs and outputs of the controller are defined as follows:

- Inputs:
  - C: output of the sensors on the farm road
  - clk: clock input
  - rst_n: **active low** reset signal

- Outputs:
  - HWR: to turn on the Highway red light
  - HWG: to turn on the Highway green light
  - HWY: to turn on the Highway yellow light
  - FRR: to turn on the Farm-road red light
  - FRG: to turn on the Farm-road green light
  - FRY: to turn on the Farm-road yellow light

HINT:
The block diagram of your traffic light controller may look something like this:

Submission guidelines:



- Please use exactly the same port names as defined above (It helps us in grading!)
- Submit the compressed version of Quartus project on blackboard.
    - The Project should also have a waveform file that demonstrates the working of your design
- A lab report (in hard form) should be submitted in class. It should have the following contents:
    - Design description using a block diagram
    - Snapshots of the RTL schematic and simulations
    - Explanation of the simulation waveforms (You may want to add internal registers to the simulation waveform. For example, you can add state of the state-machine and counter value for better explanation).
    - Conclusion
    - Verilog/VHDL code (You can use smaller font size to reduce the number of pages)