In this lab, you will be designing logic for a traffic controller installed at the intersection of a main highway and a country road as shown below.

The following specifications must be considered:

The traffic signal for the main highway gets highest priority because cars are continuously present on the main highway. Thus, the main highway signal remains green by default.

Occasionally, cars from the country road arrive at the traffic signal. The traffic signal for the country road must turn green only long enough to let the cars on the country road go.

As soon as there are no cars on the country road, the country road traffic signal turns yellow and then red and the traffic signal on the main highway turns green again.

There is a sensor to detect cars waiting on the country road. The sensor sends a signal sFarm as input to the controller. sFarm = 1 if there are cars on the country road; Otherwise, sFarm = 0.

Using a finite state machine, we realize the logic for the traffic controller. You are given a working code for designing the traffic controller.

Find out the finite state machine used in the provided code.

Simulate the design and using gtkwave, observe the inputs to the state machine and observe its state transitions.

Observe that whenever the sFarm signal goes high, the highway signal has to relinquish its control and turns yellow and then red. Similarly, as long as the sFarm is high, the farm road signal does not move from green to red. This implies that the farm road is given higher priority than the highway. Modify the code such that

The highway signal induces a wait time of N clock cycles, before turning to yellow. Use a counter to achieve this. (5 marks)

Similarly, when the farm signal turns green, use a timer to limit the number of clock cycles to M, beyond which the farm road signal relinquishes its hold on green and turns red. (5 marks)