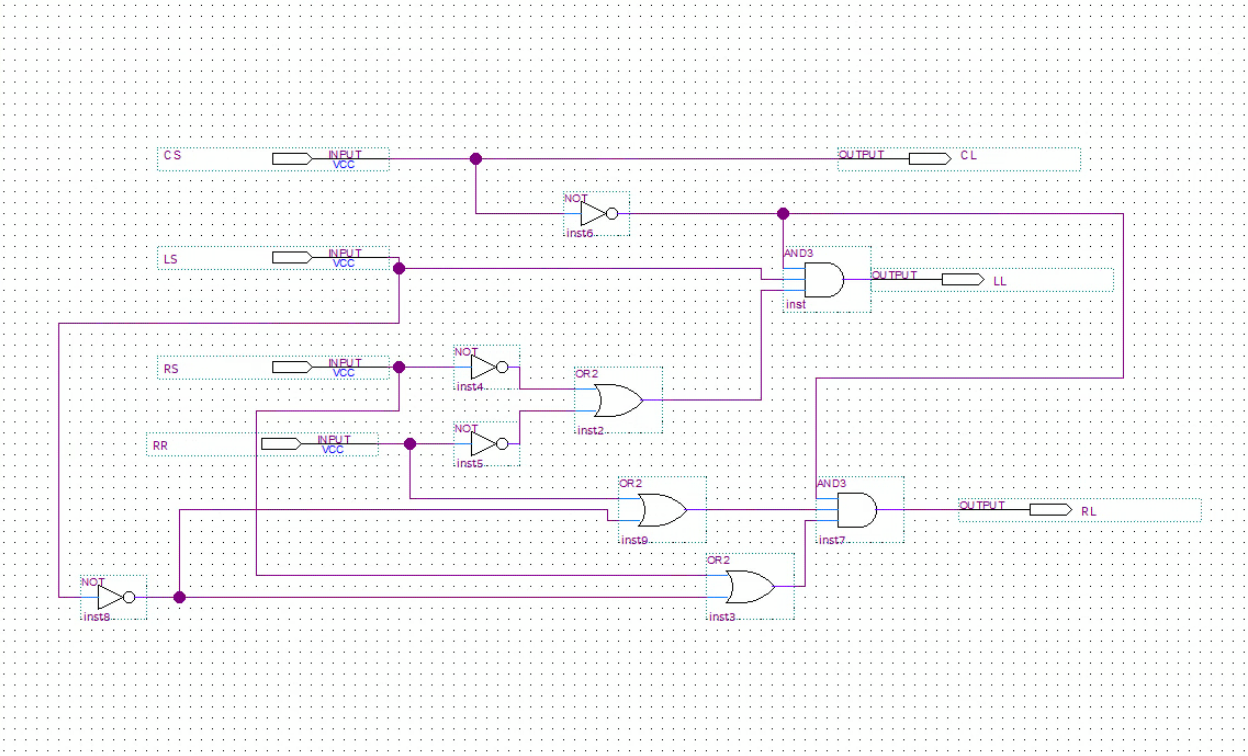
Tushar Shrivastav
Luca Chierotti
ELEN 21L
2:15-5:00
1/28/2022

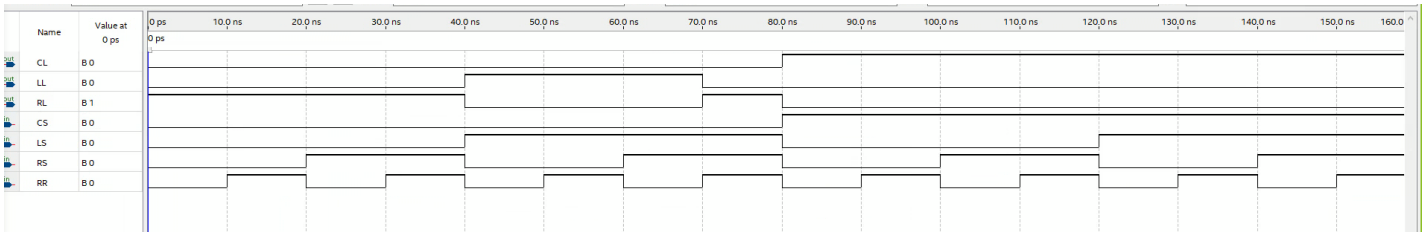
Lab 3 Report

1. Write an introduction describing how your circuit should operate and the design choices you made.

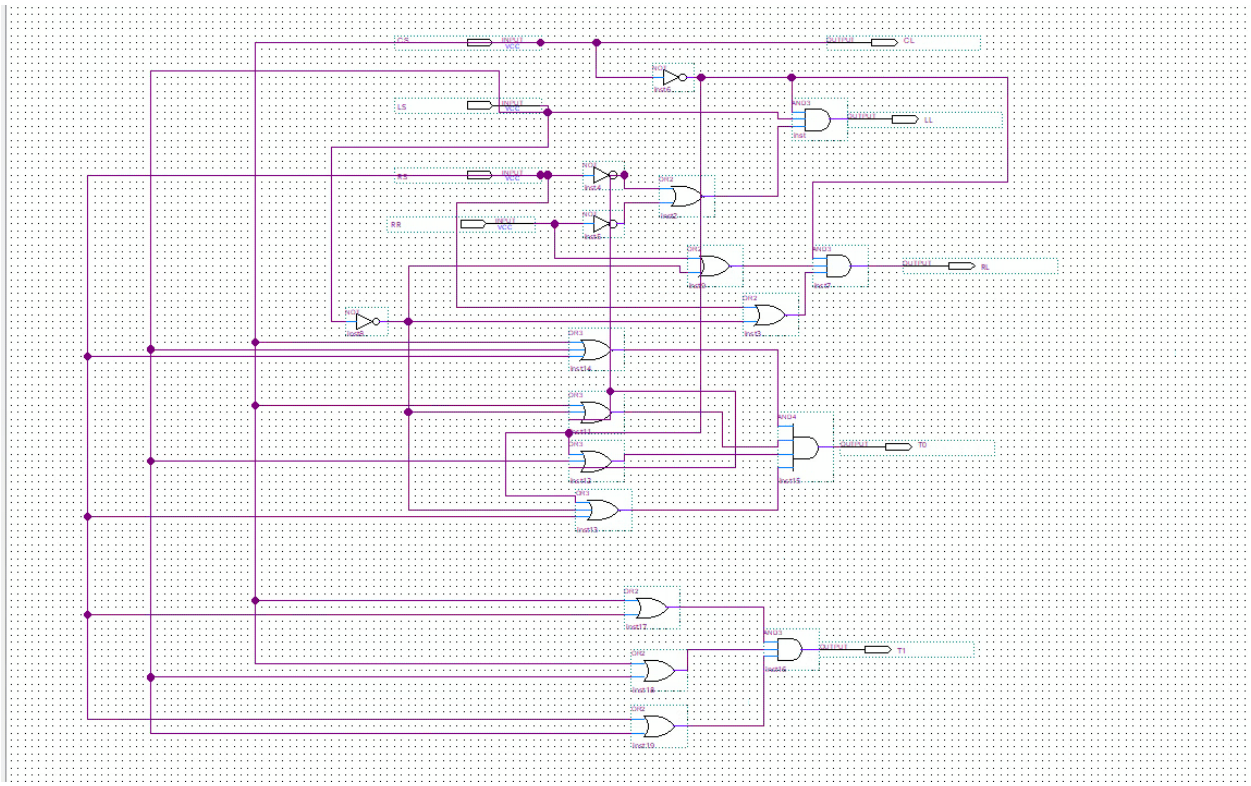
For this lab assignment, we were tasked with designing a highway entrance ramp metering controller using AND, OR, and NOT gates. In our original design, our entrance ramp had three lanes (inputs): a carpool lane, a left lane, and a right lane. Each of the three lanes has a sensor connected to a corresponding traffic light. In our simulation, only one lane can be given a green light (or a 1) at a time. The carpool lane is given priority. We also included a round-robin (input signal to make sure only one car goes at a time. Subsequently, in the lab, we were given two more outputs which acted as counters for how many cars were in the lanes. For all outputs, we first created truth tables based on the description. Then we made k-maps to find the cheapest functions for each output. These functions were then implemented into Quartus prime.

2. Include your schematic and simulation results.
POS Schematic

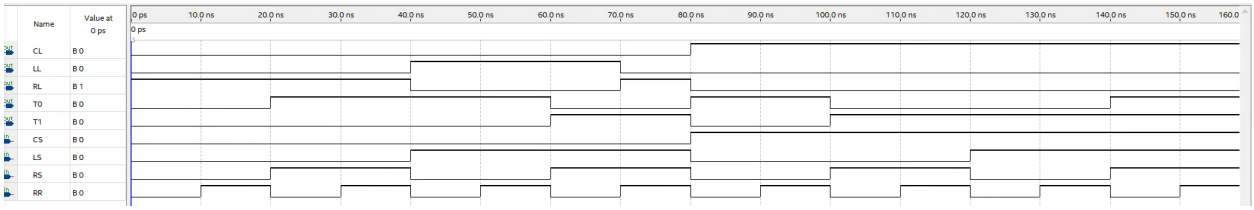




T1 + T0 Circuit



T1+T0 Timing Diagram



3. Describe the simulation strategy that you used to test your circuit design. Did it identify any errors in design or implementation? If so, what were they and how did you correct them?

Firstly, our design had no design errors. The only errors were in the implementation of the circuit into Quartus prime. We had so many connecting wires that it became confusing to understand where each wire was going. Furthermore, since there were too many wires we often had mistakes in the connection between gates. This caused us to misplace two inputs in the second half of the lab assignment, and have incorrect simulation results. We corrected this by simply redrawing the input lines to the correct destination. For our simulation strategy we first

implemented these functions in the program. Since we originally had 4 inputs, we decided to split our simulation interval into 160 ns with 10 ns intervals. This allowed us to easily match the input combinations from our 16 row table. Once we ran the simulation we found that our data matched our truth tables from the pre-lab. With the extra inputs, we also ran the simulation and found T0 to have incorrect values. This prompted us to re-check our design as we had verified our k-map functions.

4. Do you think your simulation strategy would detect all design errors for this circuit? Why or why not?

Yes, if we verify our k-map functions before implementing them into Quartus then we can be sure that our theoretical implementation will provide the correct outputs. If we then create the Quartus design and run the simulation, and receive false data, we can be sure that there is a design flaw. If we fix these flaws and the simulation matches our truth table which our k-map is based on, then we have found and resolved any design errors. However, if we are not given the ability to verify our k-map functions, we could still correctly match our quartus design to our functions. Here, unfortunately, the truth tables would match while being both incorrect. Thus, this is a flaw in our simulation strategy which could cause us to not detect all design errors.

5. What logic would you add to create a new output, ERR1 which would be 1 if two or more lights (CL, LL, and RL) were turned on at the same time. How is that logic similar to the logic that created the T1 output?

We would create three pairs of inputs (CL, LL) , (LL, RL), (CL, RL). These pairs would all be connected to 3 2-input AND gates. Then the outputs of these gates would be connected to a 3 input or gate, where if any of the AND gate results were 1 the ERR1 would turn 1. This would make sure that all combinations of possible errors are checked and if any error is detected (error meaning two or more lights are 1) then ERR1 would be 1.

This is similar to T1 because it requires the same number of gates. And if we implemented T1 as SOP we would also use 3 2-input AND gates and 1 3 input OR gate. Except for this time we are using (CL, LL, and RL) rather than (CS, LS, RS) i.e. the inputs are different.

6. Describe how you would modify your circuit to include a fifth input TM, a timer that is turned on at intervals controlled by the traffic density. When TM is “1”, the circuit operates exactly as specified in the problem statement. When TM is “0”, all output lights are red. Show a schematic for your modified circuit which includes the timer input.

We would modify our circuit to include the fifth input TM, by having 2-input and gates to go to all the outputs (CL, LL, RL). One input would be the normal circuit, and the other input would be TM. This way the problem’s requirements would be met, because if TM was 1 then the circuit would operate and if it was 0 the outputs would flash red, regardless of what the circuit is inputting.