Lab View Project

Final Project Design Document

Author:
Bishwanath Bastola

Date: May 5, 2021

I have decided to make traffic light simulation. I was inspired because of the current Nepalese Traffic System. There are not any traffic lights system in Nepal, so Nepal faces a toe of traffic problem. I wanted to learn at least design of Traffic Light in hope that I will return to my country and help in construction of traffic lights.

I simulated the design in LABVIEW in version 20.0.1 on macOS Big Sur. I will talk on the design part of the system in this report.

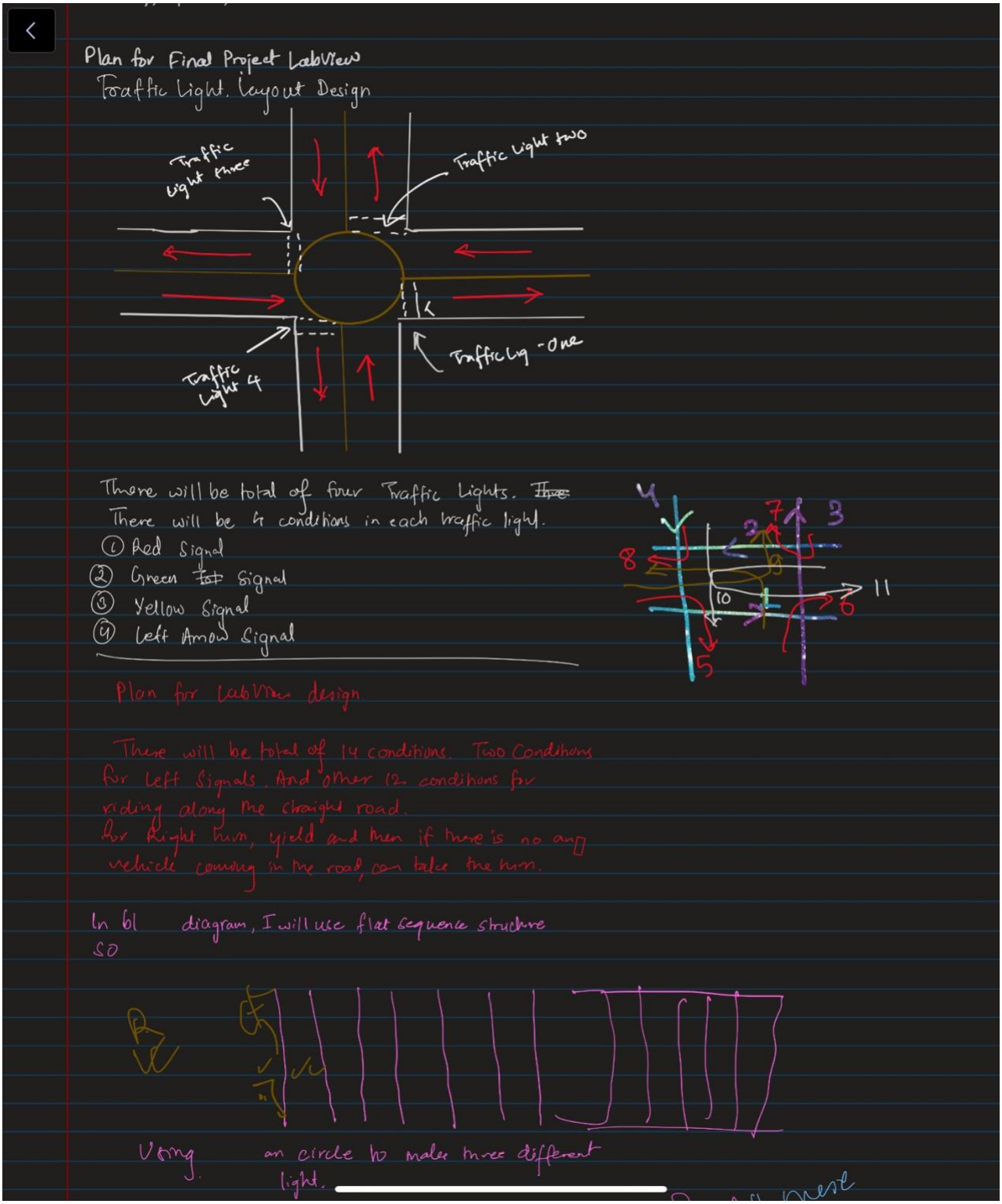
The main design in this system is simulation of traffic lights. I have designed the four-way intersection. There are altogether of 4 traffic poles. Each pole has four lights: Green, Yellow, Red, and Left Turn Signal. Along with that I made the pedestrian signal as well. To clarify more, I designed the zebra crossing as well using tools palette.

For designing the traffic signal part, I select the Boolean option from control palette and then round led. They have different colors when turning on and off at first. So, I decided to use colors like red, green and yellow when they are on and just black when they are off. For such observance, I select left on the round led and select the properties option. Then at the appearance tab, I choose the appropriate colors. Then for four poles, I did copy those lights for three more times. After that, I added the left turn signals in each of the poles. Along with it, I added the pedestrian signal as well. I used the different Boolean points or options.

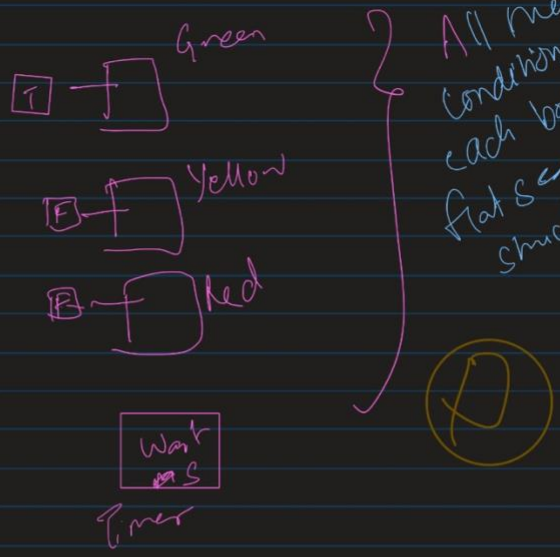
I have designed the front panel layout using decorations in Control Palette. For that, I open the front panel and left click to open the control palette. Then select the decorations option. That is how I designed the layout design for the road, and zebra crossing.

Moreover, I made one traffic light outside of the model just to show the meaning of the different colors in traffic lights. I provide the clarification using different words in the front panel. This is all about the working part of the simulation, but there is a lot of different machinery running actually a programming language in Block diagram.

Before start working with the real project, I had to work with different conditions for four-way intersection road. So, I made a plan for designing this project.



Using line an circle to make three different light signal.

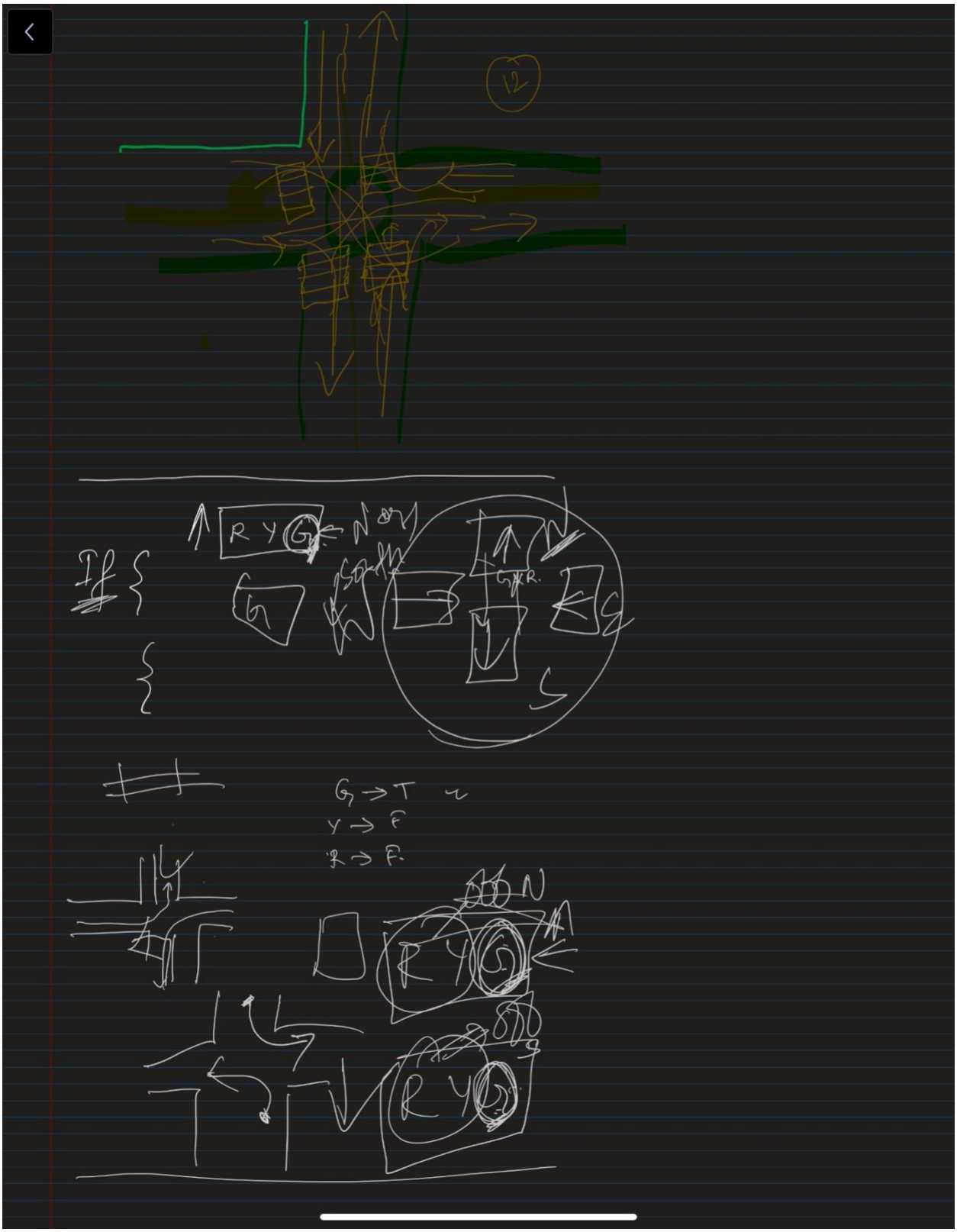


Outside flat sequence structure, we may use while loop for continuous running of traffic light.

I may include the vehicles running ~~in~~ maybe four vehicles showing each conditions.

Using the function of Image or making visualization from tools function. If I use the / design the vehicles then,

I ~~will~~ will use the case structure inside the flat sequence insider, only on the conditions having true value.


12
R Y G
If
G → T
Y → F
R → F.
R Y G
R Y G

After making my design plan, I had to execute it. So, I search for the tools that will run the codes accordingly and found out how I can use the flat sequence structure in accordance with the case statement. So, I selected the flat sequence structure from function palette. Then I enclose the flat sequence structure within the while loop. I did it because I want my simulation to run continuously until anyone stops it. Then I put the different lights on outside of the flat sequence structure

Then, inside the flat sequence structure I started using the icon of the traffic lights. At first, I had to arrange them properly, because they were a lot and in random places. Now all I had to do was to write the program for different cases.

For the first box in flat sequence structure, I used the true condition for Green 5. This is the west traffic light. Then I assigned the false value to other three conditions for the traffic lights. I had to think what happens to the other light when the Green 5 is on. The east pole green light should also be on. In this way, there will be smooth flow of traffic in East West lane. I used the time delay of 15 seconds for each transition of colors in traffic lights. I even had to make the pedestrian light green in N-S zebra crossing. For that I assigned Green 7, Green 8, Green 11, and Green 13 true value. So, those lights will turn on when the Green 5 is true. I placed all those conditions inside the case statement box.

After completing one condition, I worked for stopping the E-W traffic and flow N-S traffic. All the process is same as the previous condition. Only now the direction is different. Analogy is same as previous. I don't believe it is necessary to explain the same case. But in N-S traffic flow, E-W pedestrian signal will turn green and another signal at E-W will be red. For left turns, all the lights are assigned false value. except for the turns you are making like E-W or N-S turns.

I didn't have to make the separate 12 boxes inside the flat sequence, because I used one condition for each color and then used the case statement for cases for other light inside the same box of flat sequence structure. For adding the different conditions, all I had to do was click left on separation line of flat sequence structure and select add frame structure.

This is all about the design part of the simulation. It was not that hard thinking about the conditions, but it was hectic to see a lot of icons. It can easily create the confusion too. I could have designed it by state machine structure, but I felt easy working in

this style, so I preferred to stay in same language style. Overall speaking, I used the case statement inside the flat sequence structure. For stopping the while loop, I used the end condition within the loop.