VIET NAM NATIONAL UNIVERSITY HCM CITY
HCM UNIVERSITY OF TECHNOLOGY
FALCUTY OF COMPUTER SCIENCE AND ENGINEERING
DIVISION OF COMPUTER ENGINEERING

----------o0o----------

LOGIC DESIGN WITH HDL (CO1025)

ASSIGNMENT:

CROSSROAD TRAFFIC LIGHT

Group 1

INSTR: DR. CUONG PHAM QUOC

STD: ANH NGUYEN QUOC – 1852238

STD: THANH TRUNG PHAM TRAN – 2153929

STD: GIANG TRAN TRUONG – 2152434

STD: TRUNG TRINH MINH – 1852825

HO CHI MINH CITY, May 5th 2022

SUMMARY

The assignment is about simulate a four-way traffic light, by implementing two separate lights, each from one direction, using Verilog language and demo on the FPGA board. The simulation should be similar to it from real life, plus it must have all the functions that we carry out through working with the project.

The report explains the thoughts and processes behind building the project, with the red, yellow, green lights included. Furthermore, it proposes the potential functions that could be added but eventually left out to keep the system compact.

CONTENTS

1. INTRODUCTION

1.1 What is Traffic light

Traffic light which is one of the essentials public facilities that keeps an important role to control the traffic. Traffic light were first installed in 1868 in London, United Kingdom by J. P. Knight. First design of traffic light using gas lamps for keeps light. But soon, it exploded on January, 1869 as a result of a leak in one of the gas lines underneath the pavement, injuring and killing policeman. The concepts are banned until the electric signals appears.

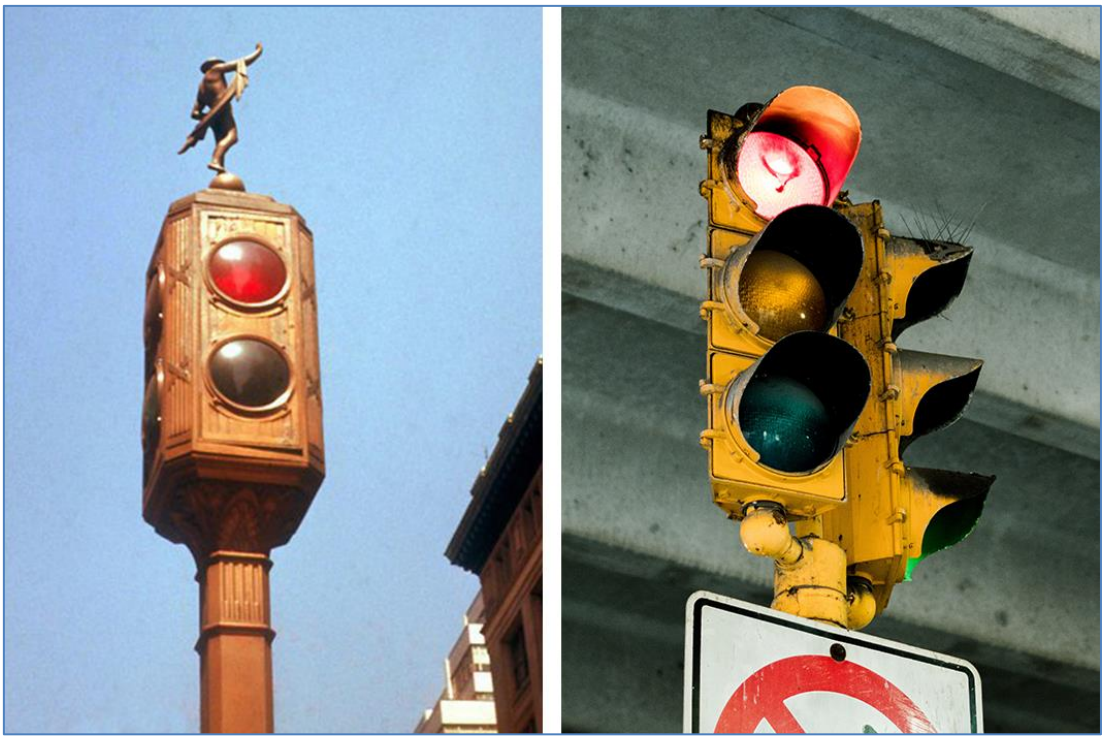

Figure 1: Two example of the traffic lights

The purpose of traffic light:

- Keep safe and efficient traffic light flow.
- Assign right of way to maximize capacity, minimize and reduce collision and conflict.

1.2 Main problems

The main problem of this project is how to implement exactly the time of the traffic light work. The signal between to flow of traffic change alternatively and match with

each other's. To do that, we need to implement the timer perfectly to receive a correct result of simulating the light.

2. DESIGN

2.1 How the traffic light normally works

Consider the concepts of traffic light show in figure.

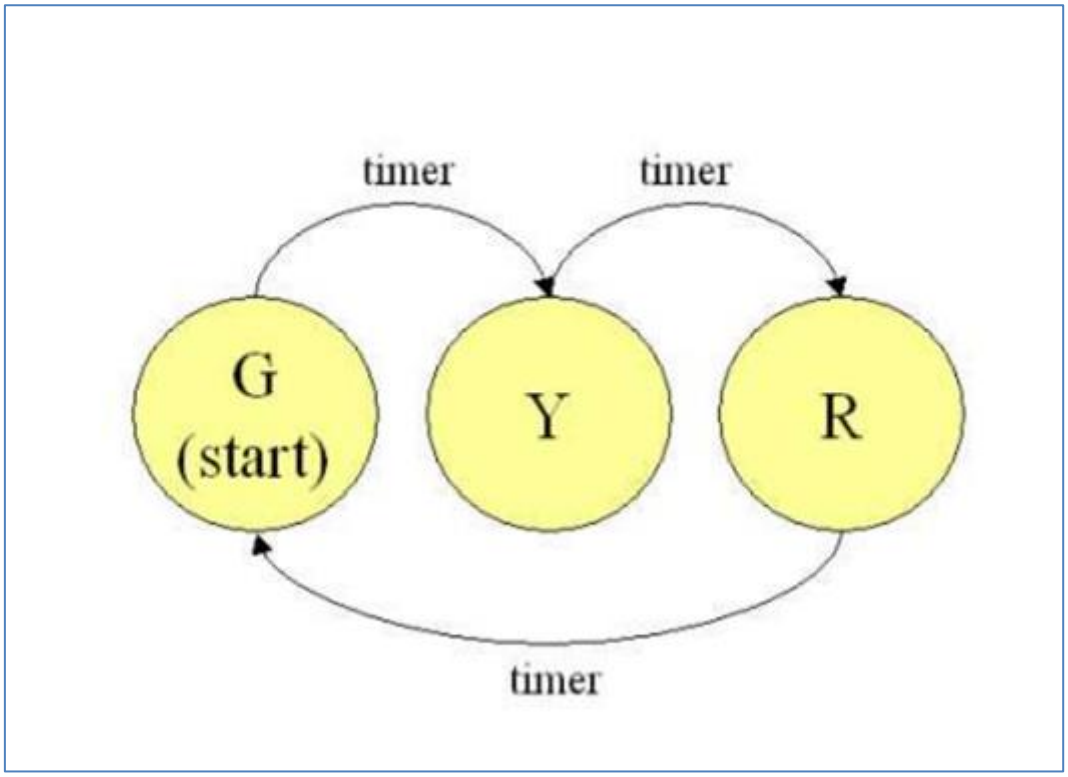

Figure 2: States of one specific light

This is the summary of how a signals change from Green, Yellow to Red. And the signals between two flow of traffic change must be properly. Note that the sequence theoretically continues forever.

Moreover, the time of the red light will equal the total time of the green and yellow light combined. People sometimes refer this as an overlapping of time (in each light's own state/sequence versus the other one). In other words, when one turns from Green to Yellow, the other light should remain as Red. Then when the first light finally turns red, until this moment the other light turns Green, hence the overlapping of time.

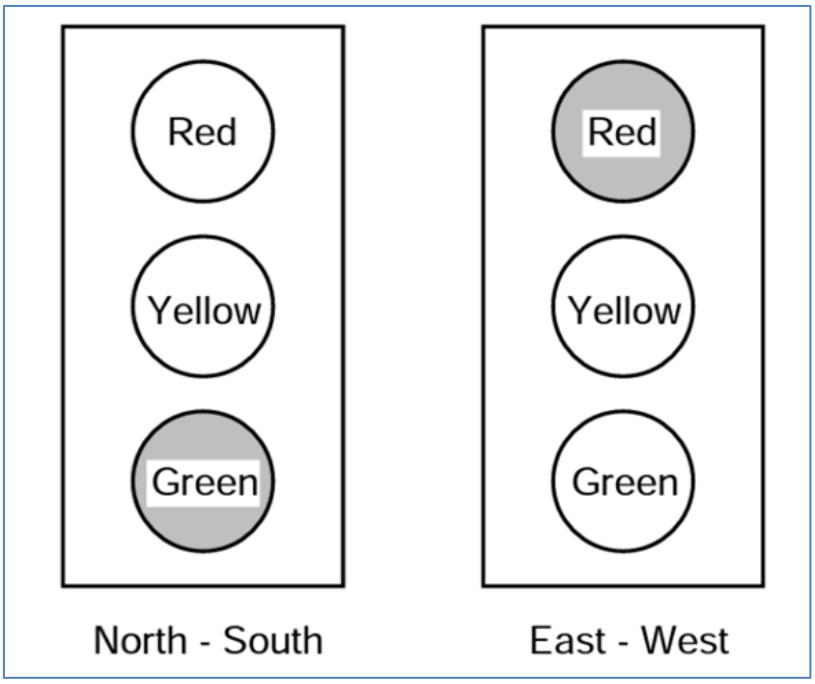

Figure 3: One instance of the two traffic lights states-relationship

This figure shows how the lights change. The timer of Red in East-West flow needs to be equal to timer of Green plus Yellow in North-South. And then vice versa.

2.2 Few-vehicle-around mode

We try to carry out a new function of the modern traffic lights on our project. The traffic lights can detect the quantities of vehicles is fewer or no vehicles around. This action might be executed by another device, like sensors or else. In our implementation, we let this signal to be manually modified (via pushing button). When the instance calls for light/very few traffic, the traffic lights will all turn Yellow and there is no timer for this mode. Furthermore, when the mode deemed unnecessary, especially when the traffic becomes heavy again, the input can be put off and the system will return to the normal mode: automatic timer traffic light.

The reason behind this mode is to improve the flow of traffic when the traffic is light. Vehicles can participate in the traffic however they want as long as no danger present (i.e heavy traffic). Regardless of the drivers/riders' skills, this state acts as a boost to traffic, where the normal timed traffic lights are abundant.

2.3 Emergency mode

Another functions that traffic lights can execute is to turn the traffic command (traffic lights' signal) into one desire need. When the instance calls for an emergency situation, like an ambulance, firetruck or high-value-convoy passing the junction, one can (manually) control the traffic to turn the Green light for such emergency. This priority, however comes with a cost: the direction which was prioritized will be set as Red lights, thus giving the other one the right to continue the traffic.

In other to set the emergency mode, user can push the predetermined buttons. There are two emergency situations, each instance gives the priority to one direction.

2.4 Reset mode

In some instance, the traffic light will return from whichever state it is to the "default" or initial state. This act can be considered as resetting the system.

One clear example is when the traffic light is in state "Few-vehicle-around mode" and the system needs to become normal, the reset function kicks in and do the job.

In our implementation, the reset mode is either built in other mode, or it can be activated via flipping back and forth a predetermined switch.

2.5 4-bit binary timer

Our traffic light system also included a 4-bit binary timer using the four basic LED of the FPGA. This is to count the waiting time of each state. The Green light intervals are 8 seconds, and the counter will count down to 0 from 7. Similarly, the Yellow light intervals are 4 seconds and the Red light intervals are 12 seconds.

This function is to help the drivers/riders to know the exact time and instance of the traffic, thus making importance decisions further down the line.

3. EVALUATIONS AND IMPROVEMENTS

After our simulation and implementation, we gave them some afterthoughts and evaluations.

The traffic light system is definitely light-weighted, but it has some useful functionalities beside the normal/basic function. Our system will handle the basic traffic flow well. Should the emergency emerges, its bonus functionalities will handle the situation beautifully. On a plus side, the system aims to keep every direction as equal as possible, and can situationally improve the flow (when the traffic is light, temporarily limit the traffic lights to yellow only).

With that being said, the truth is, there are plenty of rooms for improvements:

- Instead of a 4-bit binary timer, the system could have a 4-to-7-bit decoder to display the decimal waiting time (second) on a 7-segment LED.
- The mechanic of controlling the mode (switch between normal and emergency for example) could be implemented to edge detect logic, instead of require the input to be constant.
- Introduce more modes to deal with more situations. In other words, make the implementation for flexible.
- And maybe more...

4. IMPLEMENTATION

File name: $traffic\_ASM.v$

Testbench file: $traffic\_ASM\_tb.v$

Additional file: $clockDiv.v$

Block design:

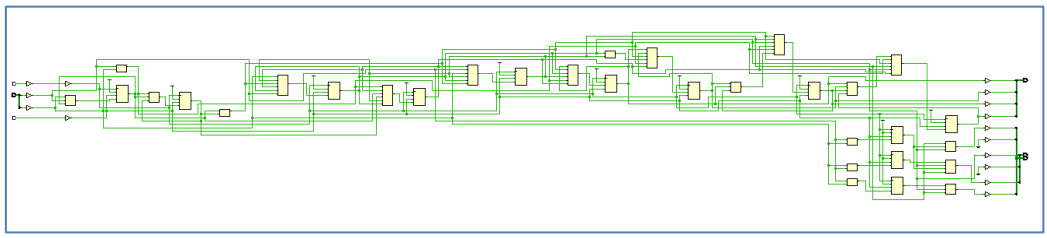

Figure 4: Block diagram

Since the image is not in high resolution, the PDF is included within the submitted files.

Schematics:

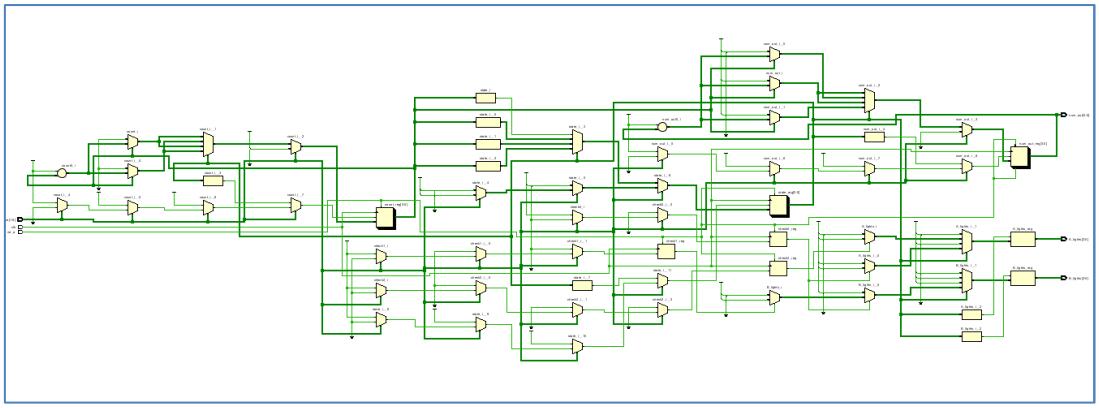

Figure 5: Schematics

Since the image is not in high resolution, the PDF is included within the submitted files.

Post-synthesis utilization:

Utilization **Post-Synthesis** | Post-Implementation

Graph | **Table**

Resource	Estimation	Available	Utilization...
LUT	16	53200	0.03
FF	12	106400	0.01
IO	14	125	11.20
BUFG	1	32	3.13

Figure 6: Post-synthesis table

Post-implementation utilization:

Utilization Post-Synthesis | **Post-Implementation**

Graph | **Table**

Resource	Utilization	Available	Utilization...
LUT	16	53200	0.03
FF	12	106400	0.01
IO	14	125	11.20
BUFG	1	32	3.13

Figure 7: Post-implementation table

Waveform:

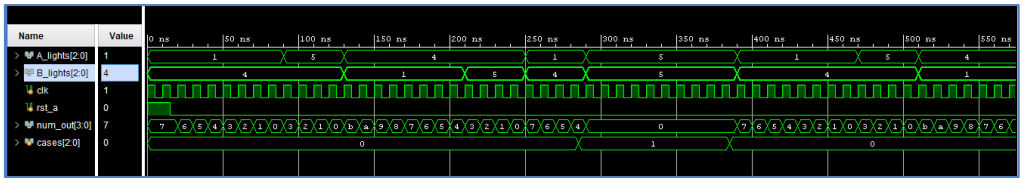

Figure 8: Waveform A

This waveform A demonstrate the normal mode of the traffic light, where it changes the output signal $Green\ (1) \rightarrow Yellow(5) \rightarrow Red(4)$ and repeat. Note that the timer (num_out) only counting for the first light (A_lights). Then mode "Few-vehicle-around" kick in and both lights turn Yellow. After that the system resets to normal again.

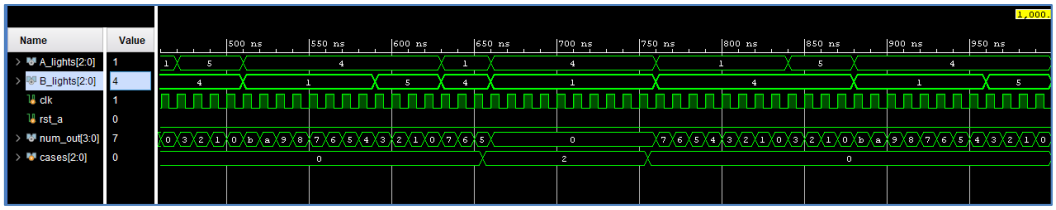

Figure 9: Waveform B

The waveform B demonstrate the system turning from normal mode to "Emergency" mode. Note the changes after "Emergency" mode is over to gain people equality.

The same result is expected for the other "Emergency" mode, but with the order of the signal reverted.