DIGITAL SYSTEM DESIGN PROJECT REPORT 2023

COURSE CODE – EC204

COURSE INSTRUCTOR – RATHNAMALA RAO

ON

Traffic Light Controller(intersection of a busy road and a street road).

Project by :-

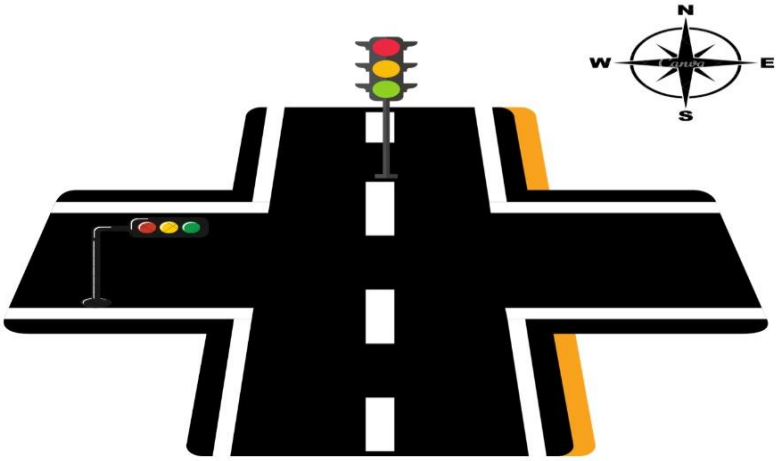
M.AKHIL TEJA

Roll no:- 221EC133

ABSTRACT :

The simple traffic light controller design project was introduced to alleviate this shortcoming and gain experience in solving implementation and interfacing problems of a modern digital system. we implement a fully functional traffic signal controller for a four-way intersection between a busy road and village road, where less traffic is present. Intersection is complete with sensors to detect the presence of vehicles waiting at or approaching the inter-section . These include HDL for modeling and finite state machines, and serial communication. Traffic lights, also known as traffic lamps, traffic signals, stoplight, stop-and-go lights semaphore or robots, are signaling devices positioned at pedestrian crossings, road intersections, and other locations to control competing flows of traffic. Traffic lights have installed in most cities around the world

to control the flow of traffic. It assign the right of way to road users by the use of lights in standard colors (Red - Yellow - Green), using a universal color code (and a precise sequence, for color blind). It requires us to develop a state machine based controller for traffic signals at a fourway intersection .



DESCRIPTION:-

As we know, Traffic lights are an essential part of road safety and help regulate the flow of traffic at intersections. We are implementing this project based on “Finite State Machine(FSM)” concepts to meet the requirements. Our project basically controls the traffic lights between intersection of a very Busy road (eg : highway road) and a normal road (eg : street road) with certain time delays for each of the color to change.

High preference is given to highway road, so green signal remains until a vehicle appears at village road. Sensors will be installed at start of street roads, If a vehicle comes in there then sensor gets high and then the

vehicle will wait for some time (Till the green signal is given) and green signal remains for some time and then get back to red.

WORKING:-

Busy road is considered as North-South road and Street road is considered as East-west road. We defined four states named S0,S1,S2,S3 in our Finite State Machine. Each State has a certain time delays from moving to another state. As per the FSM below we want to change the states. Using d-flip flops we can achieve it.

Let us say S1 and S0 represents bits of my present state and A and B represents bits of my Future state, and T be the timer for each of the states. Consider the following truth table.

FUNCTIONAL TABLES:-

INITIAL STATE			**NEXT STATE**	
S1	S0	T	A	B
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

Using K-map we can get the expression for A and B. Our expression’s are :

A = S1’ S0 T + S1 S0’ + S1 T’ B = S0’ T + S0 T’ = S0 ^ T

Each state implies to 6 outputs such as Red light in North-south road (Busy road) as Rns

Yellow light in North-South road as Yns

Green light in North-South road as Gns

Red light in East-West road (street road) as Rew

Yellow light in East-West road as Yns

Green light in East-West road as Gns.

S1	S0	Rns	Yns	Gns	Rew	Yew	Gew
0	0	0	0	1	1	0	0
0	1	0	1	0	1	0	0
1	0	1	0	0	0	0	1
1	1	1	0	0	0	1	0

Using K-map we can get the expression for Rns,Yns,Gns,Rew,Yew and Gew. Our expression’s are :

Rns = S1

Yns = S1’ S0

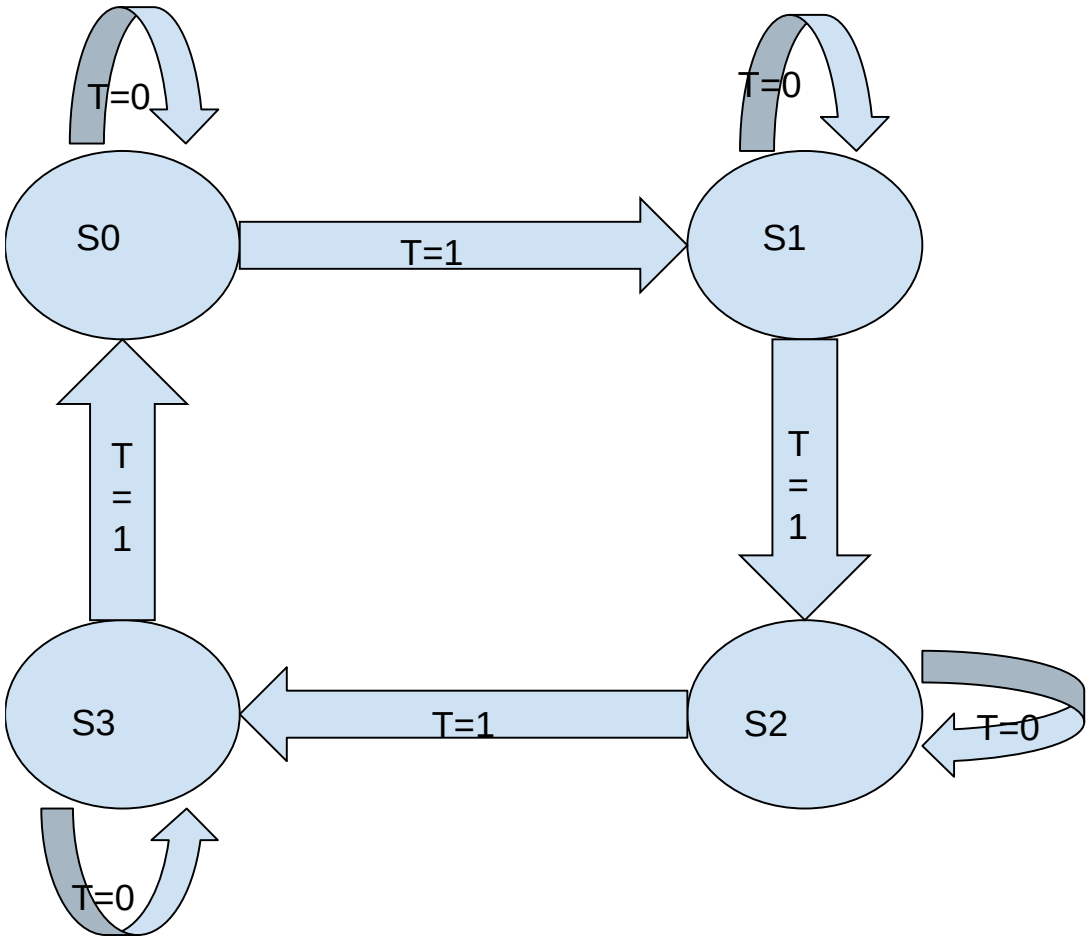
Gns = S1’ S0’

Rew = S1’

Yew = S0 S1

Gew = S1 S0’

STATE DIAGRAM:-

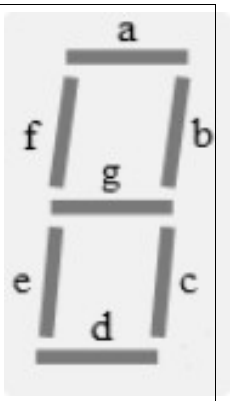


ABOUT TIMERS USED:

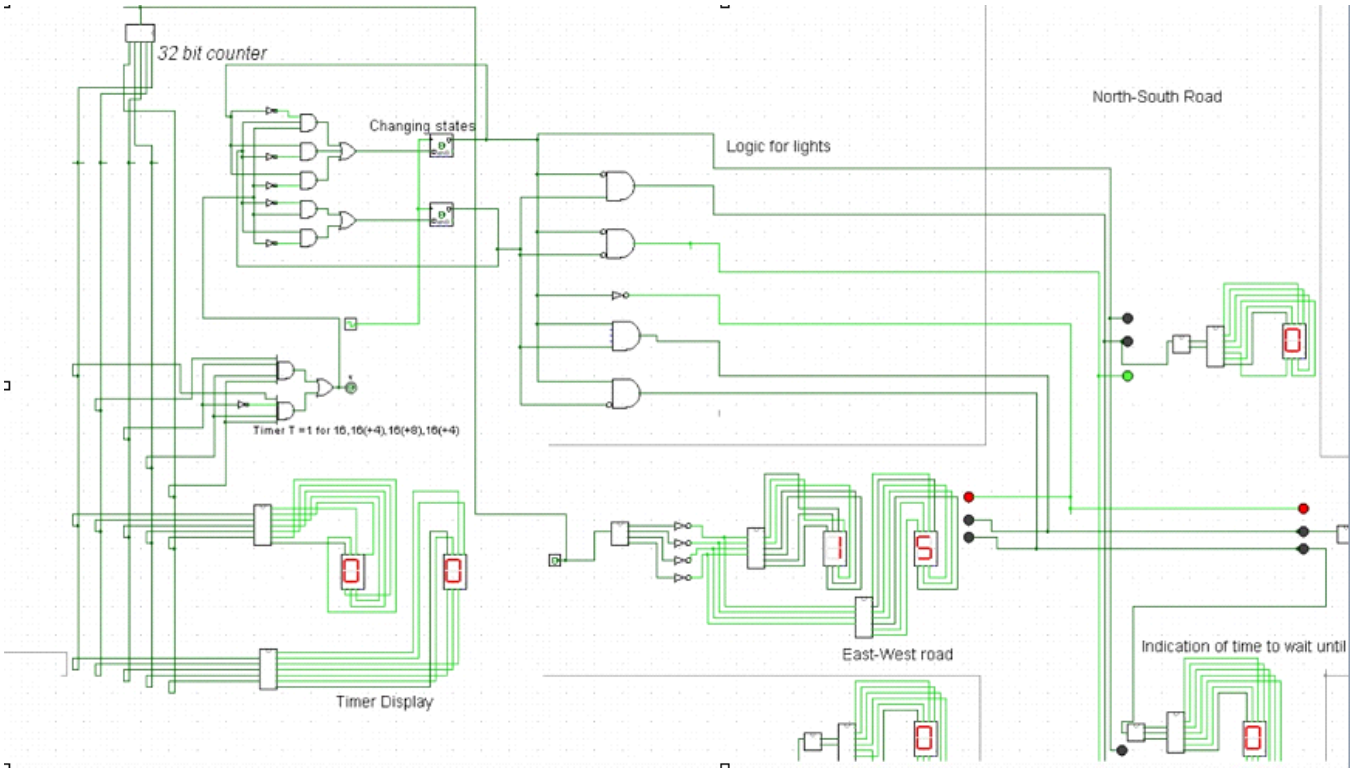
We used a timing sequence of (if a vehicle present on the street road) the red signal in east-west road or green in the north-south road remains for 16 seconds in the Hardware part (as per convinience of usage of gates) where as 32 seconds in the verilog code. Yellow remains for 4 seconds and Green light on the street road remains for 8 seconds (in te hardware part (logisim)) and 16 seconds in the verilog code. In the hardware part , we implement showing the timers using 7 segment displays using counters.

7-segment Display

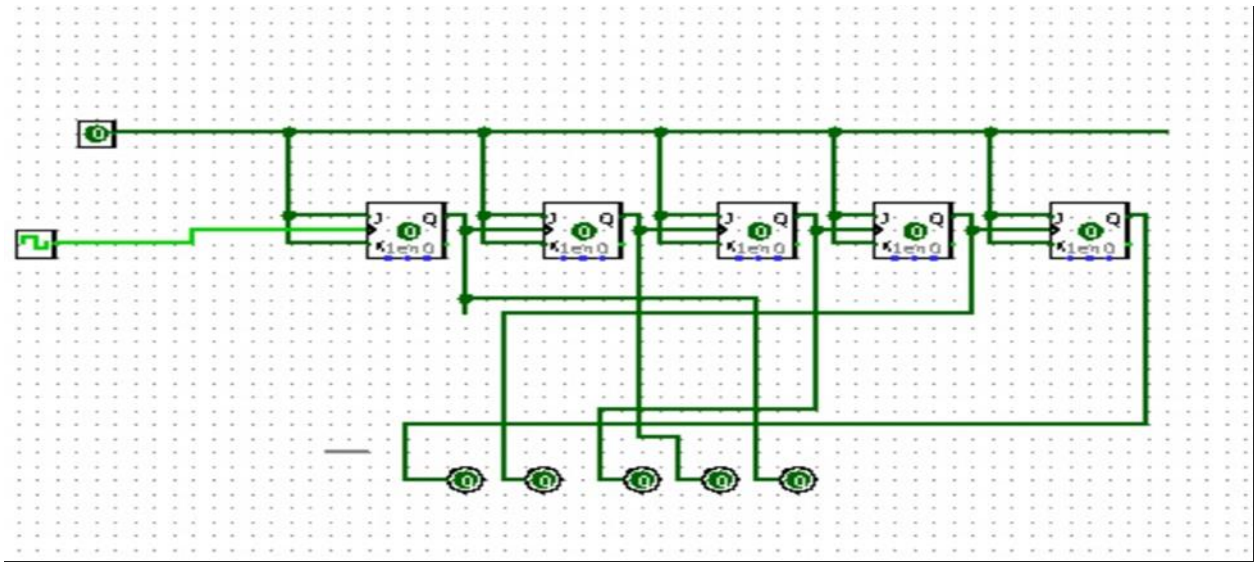
s03	s02	s01	s00	g	f	e	d	c	b
0	0	0	0	0	1	1	1	1	1
0	0	0	1	0	0	0	0	1	1
0	0	1	0	1	0	1	1	0	1
0	0	1	1	1	0	0	1	1	1
0	1	0	0	1	1	0	0	1	1
0	1	0	1	1	1	0	1	1	0
0	1	1	0	1	1	1	1	1	0
0	1	1	1	0	0	0	0	1	1
1	0	0	0	1	1	1	1	1	1
1	0	0	1	1	1	0	1	1	1
1	0	1	0	1	0	1	1	0	1
1	0	1	1	0	0	0	0	1	1
1	1	0	0	1	0	1	0	0	0
1	1	0	1	1	0	0	1	1	0
1	1	1	0	1	0	0	0	0	1
1	1	1	1	1	0	0	1	1	1



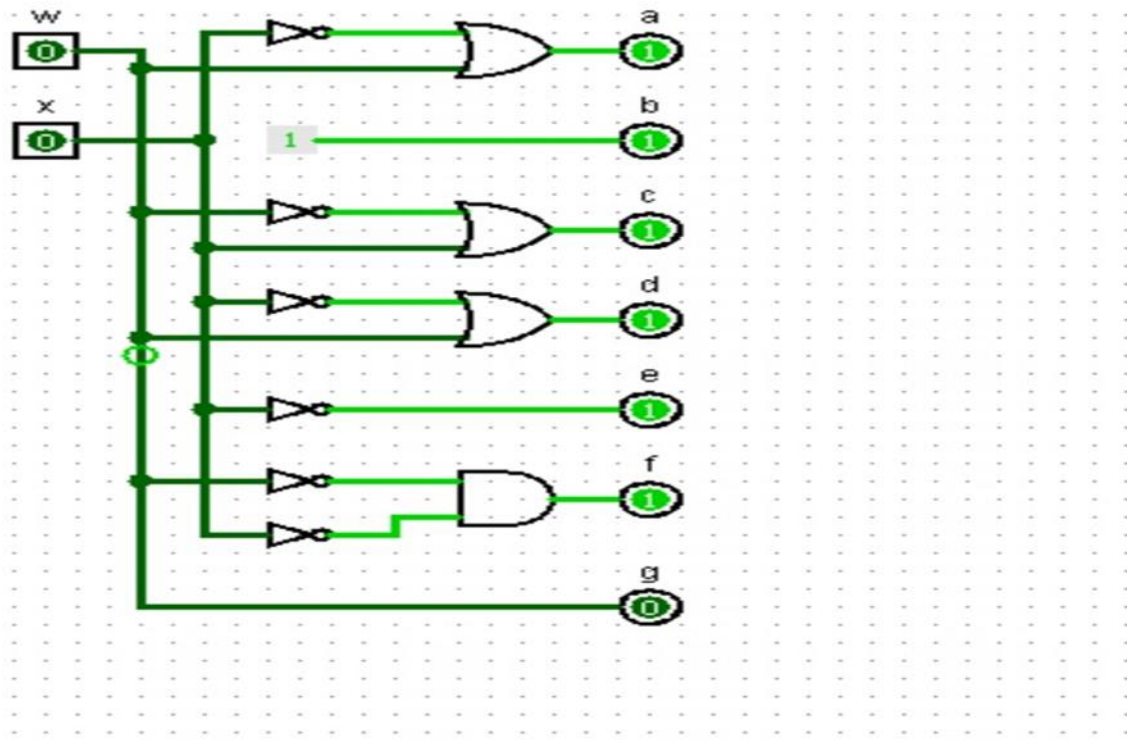
LOGISM:-



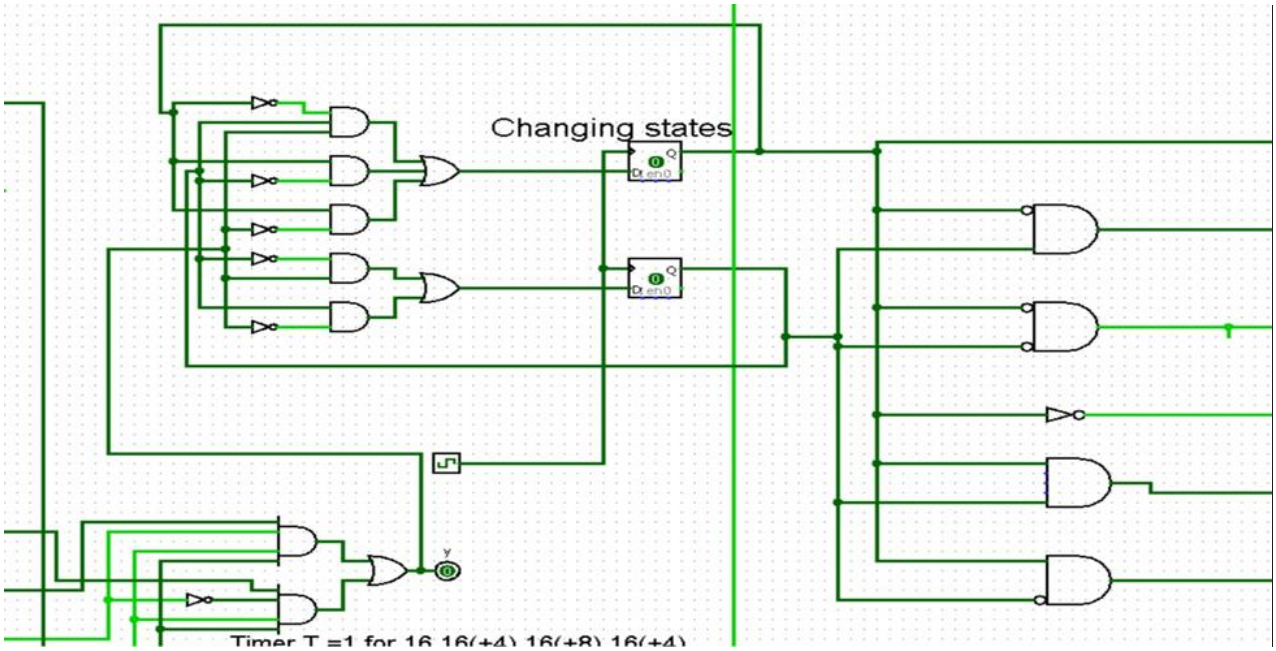
32 BIT COUNTER



2-BIT NUMBER ON 7 SEGMENT DISPLAY:-



MAIN COMPONENT:-



References:

- https://electronics.stackexchange.com/questions/50890/how-do-i-implement-a-simple-finite-state-machine-with-2-t-flip-flops
- https://www.electronics-tutorial.net/finite-state-machines/FSM-Applications/Traffic-Light-Controller/
- https://ocw.mit.edu/courses/6-111-introductory-digital-systems-laboratory-fall-2002/09bd41368db0c3f65396b13cee7cf8fb_lab2.pdf