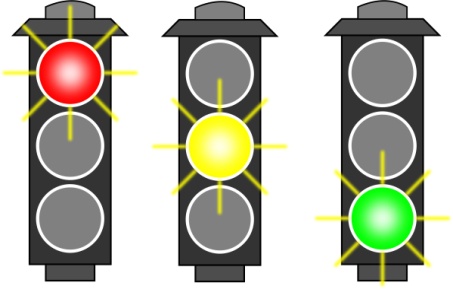
**PROJECT REPORT**



**Cross-road control system for Traffic Lights**

**Course: EEE102**

**Section 3**

**Name: Wajahat Waheed**

**Id: 21503665**

**Instructor: Ergin Atalar**

**Youtube Link:**

[**https://www.youtube.com/watch?v=GrXOH0APcuU**](https://www.youtube.com/watch?v=GrXOH0APcuU)

**Abstract:**

The main idea of my term project is to design a multi-way traffic control system which will work for not only cars but pedestrians as well. Both the cars and pedestrians will move in one straight direction on the cross-way. Added to this, I will use a 3\*5 LED matrix as a counter which counts the exact remaining time in seconds for traffic lights to change colors and guide the cars and pedestrians. Moreover, I will be using a buzzer for blind people so that they can cross the road by the help of the sound of the buzzer.

**Project’s Design Specification Plan:**

The plan is to design a system which works on the central idea of two finite state machines .One of them will be for heavy traffic and the other for light traffic. For each state in the two state machines, we will have some specific lights and when the lights of the control system changes, the state of the finite state machines should also change accordingly. Moreover, each state of the two machines should have a specific time after which it must change.

**Project’s Design Methodology:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Heavy Traffic | For Cars  North-South | For Pedestrians  North-South | For Cars  East-WEST | For Cars  East-West |
| Default value | **Green** | Red | Red | Green |
| Pushing the button  (Next 1 second) | **Yellow** | Red | Yellow | Red |
| Next five seconds | **Red** | Green | Green | Red |
| Next one second  (When the first bunch of pedestrians have crossed) | **Yellow** | Red | Yellow | Red |
| Next nine seconds | **Green** | Red | Red | Green |
| Next one second | **Yellow** | Red | Yellow | Red |
| Next five seconds | Red | Green | Green | Red |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Light Traffic | For Cars  North-South | For Pedestrians  North-South | For Cars  East-West | For Pedestrians  East-West |
| Default colour | **Green** | Red | Red | Green |
| Pushthe button  (Next 2 seconds) | **Yellow** | Red | Yellow | Red |
| Next 9 seconds | **Red** | Green | Green | Red |
| Next 2 seconds  (When the pedestrians have crossed) | **Yellow** | Red | Yellow | Red |
| Back to Default colour | Green | Red | Red | Green |

There are ten lights to operate. The Red, YELLOW and Green lights in the North-South direction will be designated as R1, Y1, and G1. Practically, we do need two signals for the north-south direction i.e. for cars coming from the other side as well but for this project since the lights on both of the signals will be the same, I am just going to represent the North-South direction as one signal. Similarly, the lights in the East-West direction will be called R2, Y2, and G2. Again the East-west direction represents two signals but will only be mentioned as one.

There are four paths in the system. One of them is for the cars in north-south direction, for cars in the east-west direction, for pedestrians crossing north-south road and for pedestrians crossing the east-west road. On the other hand, for the pedestrians there are only 2 lights namely: Red light and Green light.

To begin with, a specific colour will be assigned to the four paths as default. So, for pedestrians the colour would be red while for the cars it would be green in heavy traffic while the opposite in light traffic. For a pedestrian to cross the road, he/she has to press a single button from an option of two buttons. There will be two buttons i.e. one for less-traffic and other for heavy traffic.

I will use a 3x5 LED matrix as a counter for traffic control which counts the

exact remaining time in seconds for traffic lights to change colors and guide the cars and pedestrians. People who are blind can also cross the road easily by keeping track of a buzzer which will remind them that the green light for the pedestrians is turned on, allowing them to keep moving.

In the figure given below, the control system is displayed with arrows on the road representing cars’ direction while the black spot representing pedestrians’ path way. The blue circles in figure 2 indicate the two buttons mentioned above for heavy and light traffic.



Practical system for pedestrian button system for traffic lights

****

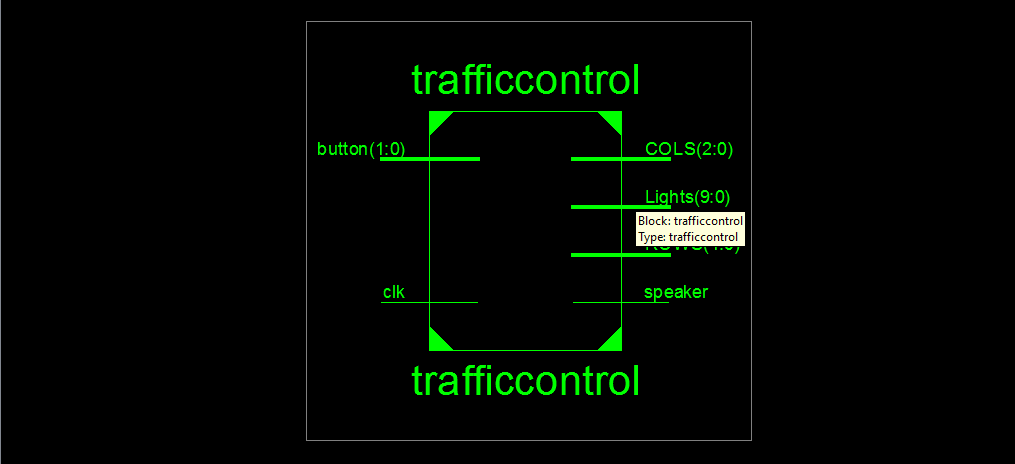
FIGURE 1

****

FIGURE 2:

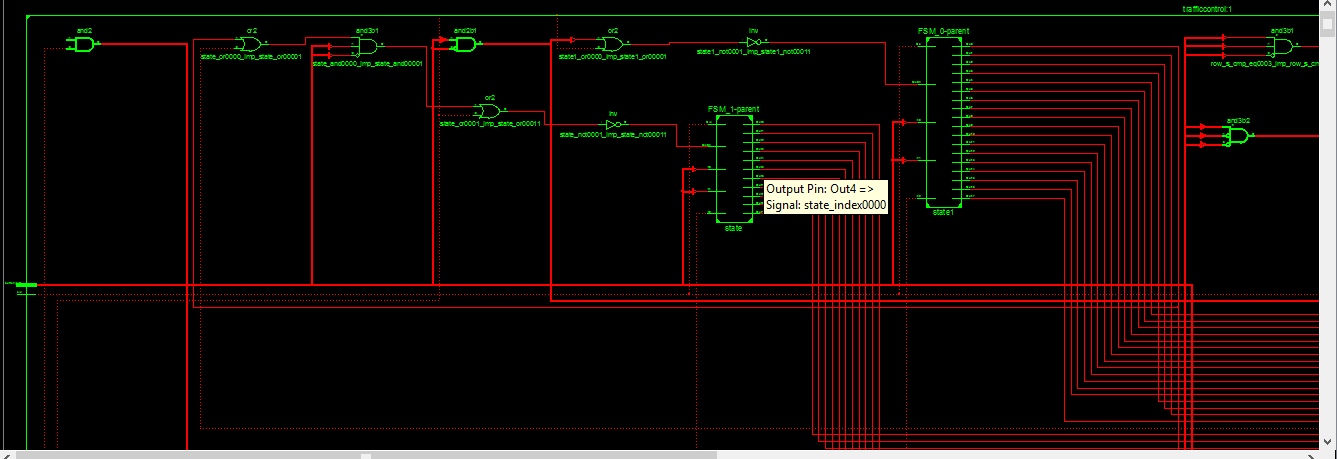
REPRESENTATION OF NORTH-SOUTH DIRECTION

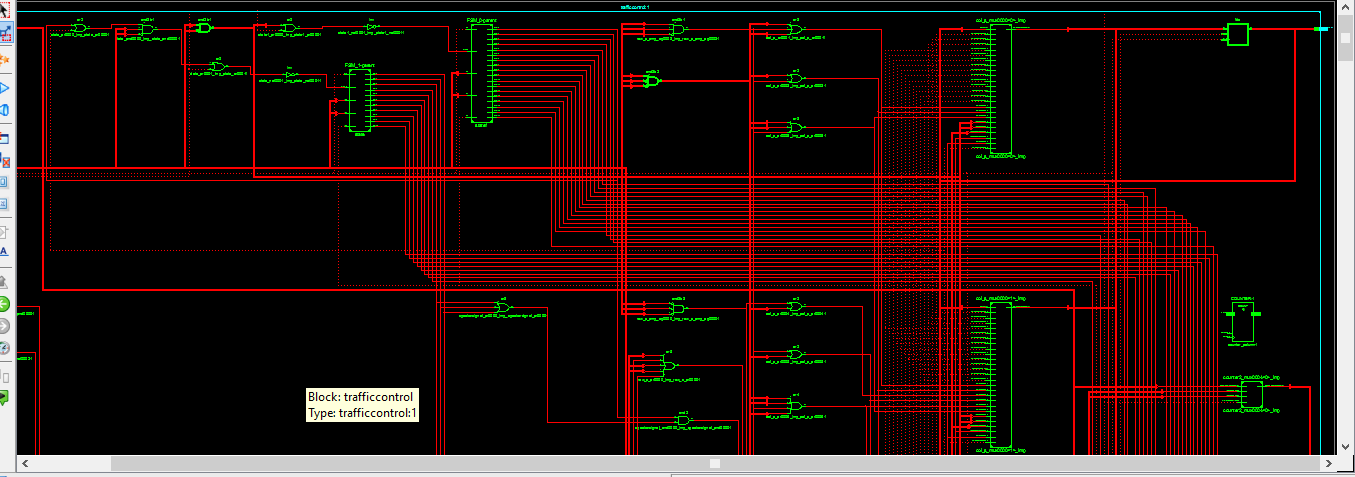
**RESULTS**

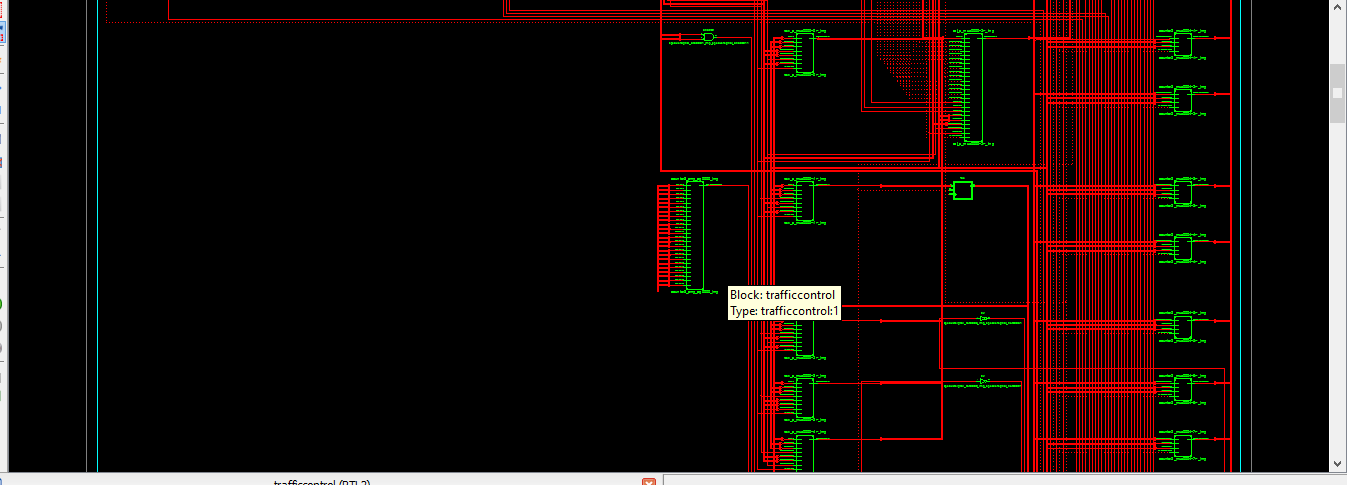
****

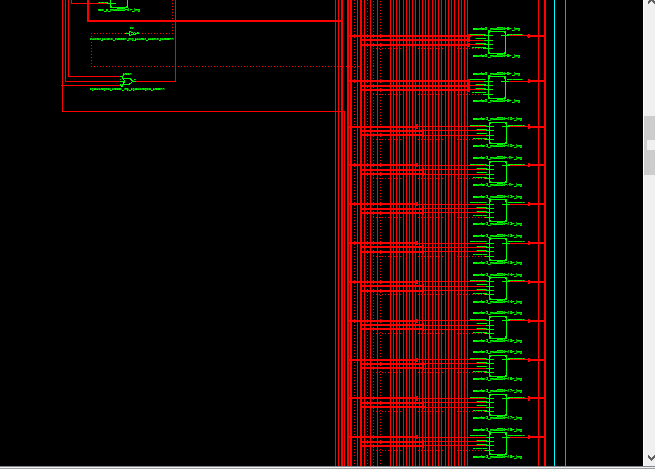
RTL SCHEMATIC (TOP-LEVEL)

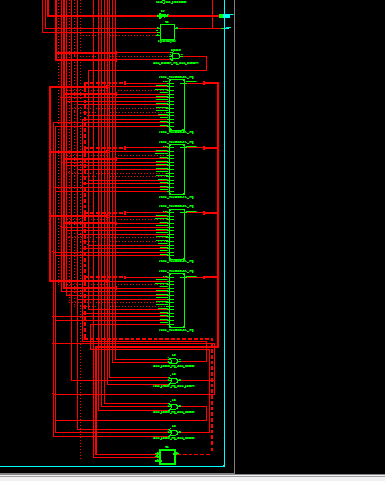
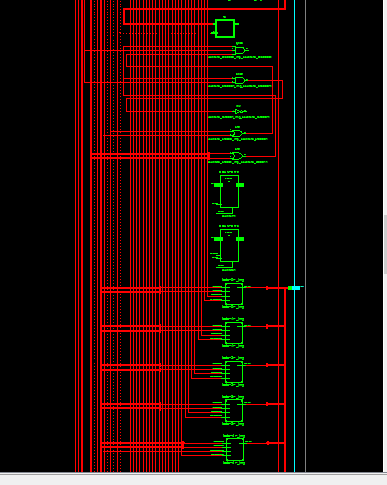
**DETAILED VIEW OF ALL RTL SCHEMATICS:**

****

****







**CONCLUSION:**

My term project of this course worked as proposed by me in the proposal. All of the objectives of this project were met. Moreover,

In this individual project, I learned a lot. First of all, I learned finite state machines much better when I used two traffic system; Heavy and Light traffic.

Secondly, I learned how to code LED matrix and learned how to understand its datasheet. Since there were only 14 pins of the led matrix as I used 5\*7 led matrix, but there were 35 LEDs. The problem was thus to find which pins represented which LEDs which I found after some strenuous trials.

Thirdly, I learned how to write a Vhdl code for a buzzer. I came to know that for us to listen a sound from the buzzer we need a specific band of frequency which was successfully implemented.

Last but not the least working individually after CS 102 group project was very fruitful. I learnt that when you are alone doing a job or a project, it becomes more convenient for one to concentrate properly. Interruptions are as well, much less, when the person works alone. On the other hand, when we are working as a group, it is pretty easy to get carried away while conversing. Although there were some disadvantages of working alone, but overall it was much better experience and I learned a lot in this project.

**VHDL CODE OF THE PROJECT:**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL ;

entity trafficcontrol is

port(

clk : in std\_logic;

ROWS : out STD\_LOGIC\_VECTOR (4 downto 0);

button : in STD\_LOGIC\_VECTOR (1 downto 0);

COLS : out STD\_LOGIC\_VECTOR (2 downto 0);

speaker : out std\_logic;

Lights : out std\_logic\_vector (9 downto 0)

);

end trafficcontrol ;

architecture Behavioral of trafficcontrol is

type state\_type1 is (s0, s1, s2, s3, s4, s5, s6); --Heavy Traffaic with 7 states

type state\_type is (s0, s1, s2, s3); --Light Traffic with 4 states

signal state: state\_type;

signal state1: state\_type1;

signal value : std\_logic\_vector (3 downto 0) := "0000"; --Number which has to decrement at every signal

signal leds: std\_logic\_vector (9 downto 0);

signal counter2 : std\_logic\_vector (25 downto 0) := (others => '0');

signal counter3 : INTEGER RANGE 0 to 10000000:= 0;

signal counter1 : std\_logic\_vector (24 downto 0) := (others => '0');

signal row\_s : std\_logic\_vector (4 downto 0) :="00000"; --5 ROWS

signal col\_s : std\_logic\_vector(2 downto 0) := "000"; -- 3 COLUMNS

signal counter\_column : std\_logic\_vector(2 downto 0) := "000";

signal speakersignal : std\_logic; --Buzzer1

begin

process (clk)

begin

if rising\_edge(clk) then

if (counter3 = 600000 and (button(0)= '1' or button(1) = '1') )then

if((state = s2) OR state1 =s2 ) then

speakersignal <= not speakersignal;

counter3 <= 0;

else

end if;

else

counter3 <= counter3 + 1;

end if;

if (counter2 < "10111110101111000010000000") then

counter2 <= counter2 + '1';

else

value <= value - '1'; --This is because of we have a countdown situation

counter2 <= (others => '0');

end if;

if (counter1 < "0000000000000000010000000") then

counter1 <= counter1 + '1';

else

--Representing number "0"

if (value <= "0000") then

if(counter\_column = "000")then

row\_s <= "01111"; --"01111" represents the row 1

col\_s <= "000"; -- 000 represents all columns are "ON" and first 0 represents column 3

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "010";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "010";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "010";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "1"

elsif (value <= "0001") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "110";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "110";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "110";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "110";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "110";

else

counter\_column <= "000";

end if;

--Representing number "2"

elsif (value <= "0010") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "011";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "110";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "3"

elsif(value <= "0011") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "011";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "011";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "4"

elsif (value <= "0100") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "010";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "010";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "011";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "011";

else

counter\_column <= "000";

end if;

--Representing number "5"

elsif (value <= "0101") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "110";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "011";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "6"

elsif (value <= "0110") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "110";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "010";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "7"

elsif (value <= "0111") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "011";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "011";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "011";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "011";

else

counter\_column <= "000";

end if;

--Representing number "8"

elsif (value <= "1000") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "010";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "010";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "9"

elsif (value <= "1001") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "010";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "011";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "A"

elsif (value <= "1010") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "010";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "010";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "010";

else

counter\_column <= "000";

end if;

--Representing number "B"

elsif (value <= "1011") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "010";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "010";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "C"

elsif (value <= "1100") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "110";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "110";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "110";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "D"

elsif (value <= "1101") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "010";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "010";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "010";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "E"

elsif (value <= "1110") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "110";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "110";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "000";

else

counter\_column <= "000";

end if;

--Representing number "F"

elsif (value <= "1111") then

if(counter\_column = "000")then

row\_s <= "01111";

col\_s <= "000";

elsif(counter\_column = "001")then

row\_s <= "10111";

col\_s <= "110";

elsif(counter\_column = "010")then

row\_s <= "11011";

col\_s <= "000";

elsif(counter\_column = "011")then

row\_s <= "11101";

col\_s <= "110";

elsif(counter\_column = "100")then

row\_s <= "11110";

col\_s <= "110";

else

counter\_column <= "000";

end if;

else

value <= "0000";

end if;

counter\_column <= counter\_column + '1';

counter1 <= (others => '0');

-- WHEN LIGHT TRAFFIC PRESSING BUTTON (0) REPRESENTED BY P11 SWITCH

if (button(0) = '1' and button(1) = '0') then

case state is

when s0 =>

if value > "0000" then

state <= s0;

else

state <= s1;

value <= "0011";

end if;

when s1 =>

if value > "0000" then

state <= s1;

else

state <= s2;

value <= "1111";

end if;

when s2 =>

if value > "0000" then

state <= s2;

else

state <= s3;

value <= "0100";

end if;

when s3 =>

if value > "0000" then

state <= s3;

else

state <= s0;

value <= "1111";

end if;

when others =>

state <= s0;

value <= "1000";

end case;

-- WHEN heavy TRAFFIC PRESSING BUTTON (1) REPRESENTED BY L3 SWITCH

elsif (button(0) = '0' and button(1) = '1') then

case state1 is

when s0 =>

if value > "0000" then

state1 <= s0;

else

state1 <= s1;

value <= "0011";

end if;

when s1 =>

if value > "0000" then

state1 <= s1;

else

state1 <= s2;

value <= "1000";

end if;

when s2 =>

if value > "0000" then

state <= s2;

else

state1 <= s3;

value <= "0011";

end if;

when s3 =>

if value > "0000" then

state1 <= s3;

else

state1<= s4;

value <= "1111";

end if;

when s4 =>

if value > "0000" then

state1 <= s4;

else

state1 <= s5;

value <= "0011";

end if;

when s5 =>

if value > "0000" then

state1 <= s5;

else

state1 <= s6;

value <= "1000";

end if;

when s6 =>

if value > "0000" then

state1 <= s6;

else

state1 <= s0;

value <= "1000";

end if;

when others =>

state1 <= s0;

value <= "1000";

end case;

else

row\_s <= "00000";

col\_s <= "000";

value <= "1111";

counter2 <= (others => '0');

state <= s0;

state1 <= s0;

end if;

end if;

end if;

end process;

process (state, state1)

begin

if (button(0) = '1' and button(1) = '0') then

case state is

when s0 => leds <= "0110000110";

when s1 => leds <= "0101001001" ;

when s2 => leds <= "1000110001";

when s3 => leds <= "0101001001";

when others => leds <= "0110000110";

end case;

elsif (button(0) = '0' and button(1) = '1') then

case state1 is

when s0 => leds <= "0110000110";

when s1 => leds <= "0101001001";

when s2 => leds <= "1000110001";

when s3 => leds <= "0101001001";

when s4 => leds <= "0110000110";

when s5 => leds <= "0101001001";

when s6 => leds <= "1000110001";

when others => leds <= "0110000110";

end case;

else

leds <= "0000000000";

end if;

end process;

ROWS <= not ROW\_S;

COLS <= COL\_S;

Lights <= leds;

speaker <= speakersignal;

end Behavioral;

**CONSTRAINT (UCF FILE) OF THE PROJECT:**

NET "clk" LOC = "b8";

NET "Button(0)" LOC = "p11"; #Light Traffic

NET "Button(1)" LOC = "l3"; # Heavy traffic

#LEDS REPRESENTATION

NET "lights(9)" LOC = "g1" ;

NET "lights(8)" LOC = "p4" ;

NET "lights(7)" LOC = "n4" ;

NET "lights(6)" LOC = "n5" ;

NET "lights(5)" LOC = "p6" ;

NET "lights(4)" LOC = "p7" ;

NET "lights(3)" LOC = "m11" ;

NET "lights(2)" LOC = "m5" ;

NET "lights(1)" LOC = "A13" ;

NET "lights(0)" LOC = "d12" ;

# ROWS

NET "ROWS(0)" LOC = "C12";

NET "ROWS(1)" LOC = "C9";

NET "ROWS(2)" LOC = "A10";

NET "ROWS(3)" LOC = "B9";

NET "ROWS(4)" LOC = "A9";

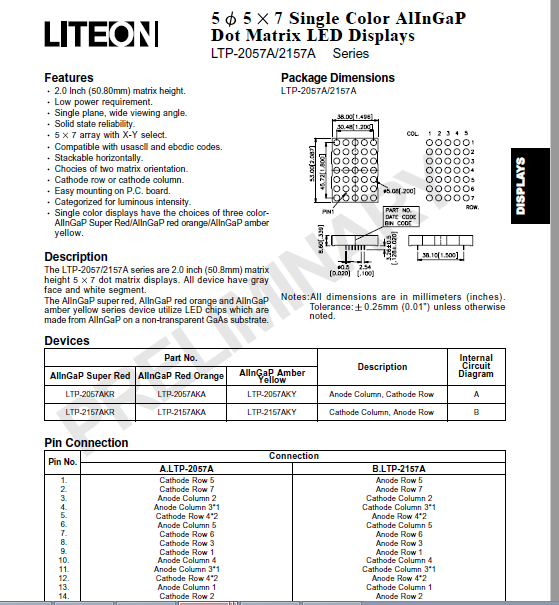
# COLUMNS

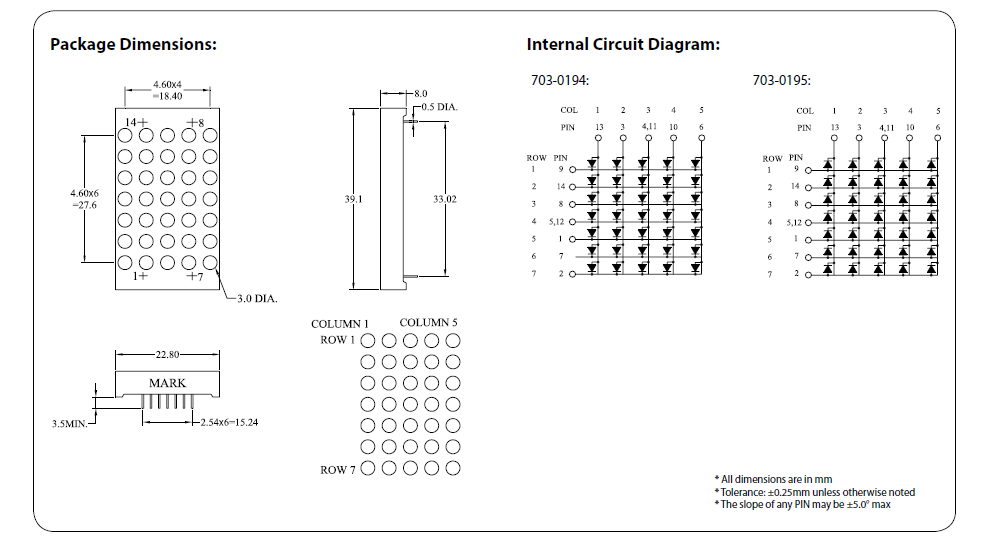
NET "COLS(0)" LOC = "B7";

NET "COLS(1)" LOC = "C5";

NET "COLS(2)" LOC = "B6";

NET "speaker" LOC = "C6";

**Datasheet of 5\*7 led matrix used:**

****

**References:**

1. <http://electronicsarea.com/electronic-two-way-traffic-light-circuit-with->

555-timer-and-cd4017-counter/

1. <http://appliedelectronicsengineering.blogspot.com.tr/2015/11/how-to-develop-traffic-light-controller.html>