Bapuji Educational Association®

BAPUJI INSTITUTE OF ENGINEERING AND TECHNOLOGY, DAVANGERE 577004

Course Project Report on

Traffic Light Control Using ARM7 LPC2148

Course: Embedded System Design Course Code: BEC601

Submitted by

Darshan N S	4BD22EC026
Darshan S K	4BD22EC028
Ganesh T M	4BD22EC032
Gowri H K	4BD22EC034

Dr. Banumathi K L B.E, MTech
Course Coordinator

Dr. G. S. Sunitha MTech. (DEAC), Ph.D., MISTE, FIETE, FIE.
Program Coordinator

Department of Electronics& Communication Engineering

2024-2025

Vision of the Department

To be in the forefront in providing quality technical education and research in Electronics & Communication Engineering to produce skilled professionals to cater to the challenges of the society.

Mission of the Department

M1: To facilitate the students with profound technical knowledge through effective teaching learning process for a successful career.

M2: To impart quality education to strengthen students to meet the industry standards and face confidently the challenges in the programme.

M3: To develop the essence of innovation and research among students and faculty by providing infrastructure and a conducive environment.

M4: To inculcate the student community with ethical values, communication skills, leadership qualities, entrepreneurial skills and lifelong learning to meet the societal needs.

Traffic Light Control Using ARM7 LPC2148

Tool Used: Keil u Vision 4, Proteus 8.

Theory:

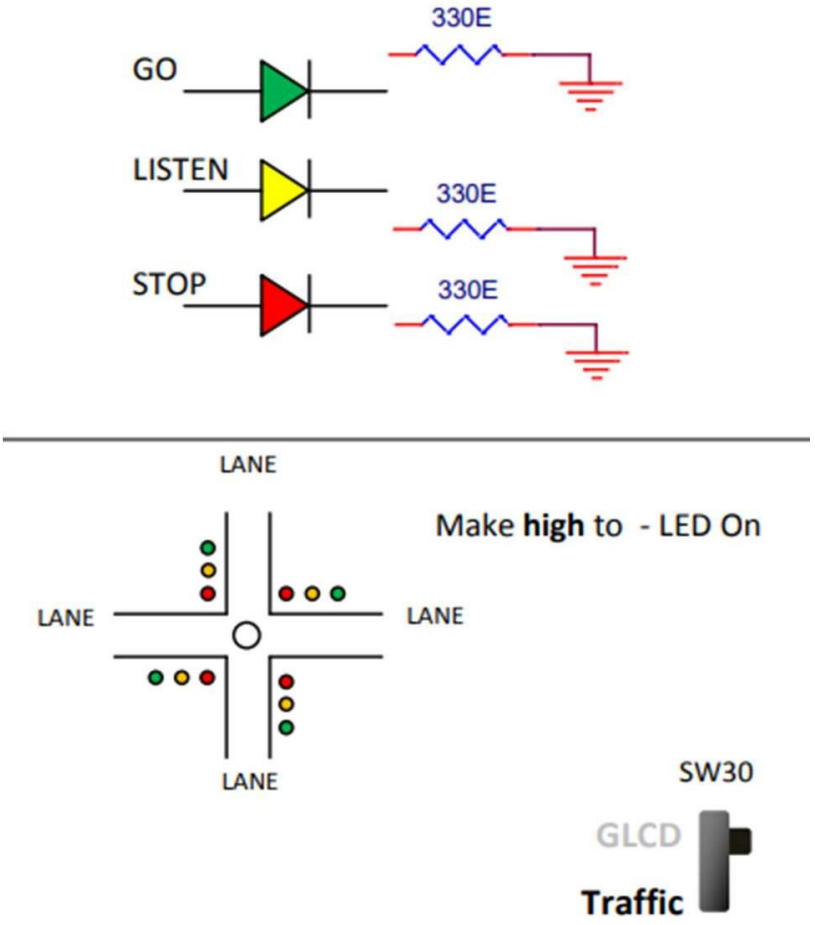
Traffic Light Control systems are used to regulate and control the flow of traffic at intersections. In this we are using an ARM7 LPC2148 microcontroller to implement the control logic for a traffic light system. The ARM7 LPC2148 microcontroller is a 32-bit microcontroller based on the ARM7TDMI-S core. It offers a wide range of peripherals and features that make it suitable for various embedded applications, including traffic light control systems. Traffic lights, which may also be known as stoplights, traffic lamps, traffic signals, signal lights, robots or semaphore, are signaling devices positioned at road intersections, pedestrian crossings and other locations to control competing flows of traffic.

In the typical sequence of colored lights:

- Illumination of the green light allows traffic to proceed in the direction denoted,
- Illumination of the yellow/aminter section. noting, if safe to do so, prepare to stop short of the intersection
- Illumination of the red signal prohibits any traffic from proceeding.

The Components used in this are:

- LPC 2148
- LED Lights(Red, Yellow, Green)
- Seven Segment Display
- LEDs are used to represent the signal and seven segment display to specify the time left for the green signal to turn red in that direction.



Pins Used:

- Port 0.16 to Port 0.30 – LEDs,
- Port 0.0 to Port 0.6 – 1st Seven Segment Display (North),
- Port 0.7 to Port 0.13 – 2nd Seven Segment Display (West),
- Port 1.16 to Port 1.22 – 3rd Seven Segment Display (South)
- Port 1.23 to Port 1.29 – 4th Seven Segment Display (East)

Pin Assignment:

For LEDs:

Direction	LPC2148 Pin	Led Action
NORTH	P0.16	D1 – Red
	P0.17	D2 – Yellow
	P0.18	D3 – Green
WEST	P0.19	D4 – Red
	P0.20	D5 – Yellow
	P0.21	D6 – Green
SOUTH	P0.22	D7 – Red
	P0.23	D8 – Yellow
	P0.25	D9 – Green
EAST	P0.28	D10 – Red
	P0.29	D11 – Yellow
	P0.30	D12 – Green

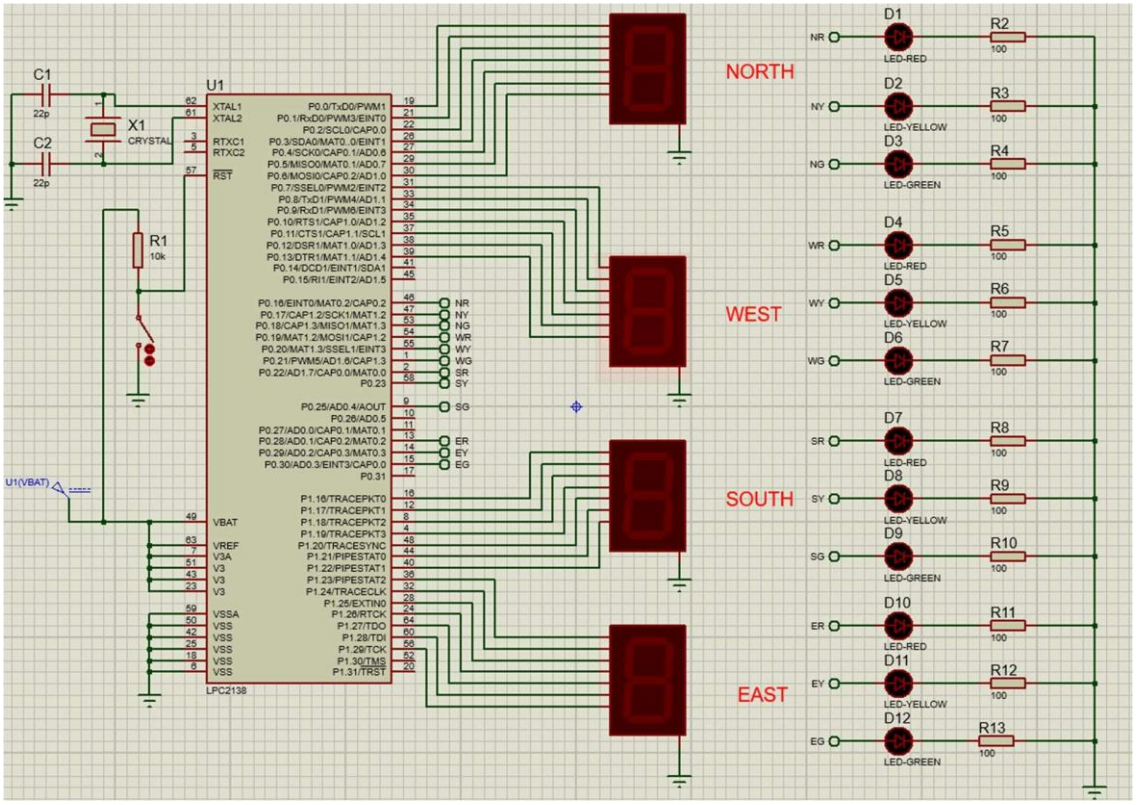
For Seven Segment Display's:

Seven Segment Display. No	a	b	c	d	e	f	g
1	P0.0	P0.1	P0.2	P0.3	P0.4	P0.5	P0.6
2	P0.7	P0.8	P0.9	P0.10	P0.11	P0.12	P0.13
3	P1.16	P1.17	P1.18	P1.19	P1.20	P1.21	P1.22
4	P1.23	P1.24	P1.25	P1.26	P1.27	P1.28	P1.29

Components Used in Proteus:

- LPC 2148
- LEDs
- Resistor
- Seven Segment Display
- Capacitor
- Crystal
- Connecting Wires

Circuit Diagram:



Program:

```
#include<lpc214x.h>
void delay();
void segment1()
  {
PINSEL0=0X00000000;
IODIR0 |=0XFF;
IOCLR0=0XFF;
```

```
    IOSET0=0X6F;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X7F;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X07;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X7D;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X6D;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X66;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X4F;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X5B;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X06;
    delay();
    IOCLR0=0XFF;
    IOSET0=0X3F;
    delay();
    IOCLR0=0XFF;
      }

  void segment2(){
    PINSEL0=0X00000000;
    IODIR0 |=0X3F80;
    IOCLR0=0XFF<<7;
    IOSET0=0X6F<<7;
    delay();
    IOCLR0=0XFF<<7;
    IOSET0=0X7F<<7;
    delay();
    IOCLR0=0XFF<<7;
    IOSET0=0X07<<7;
    delay();
```

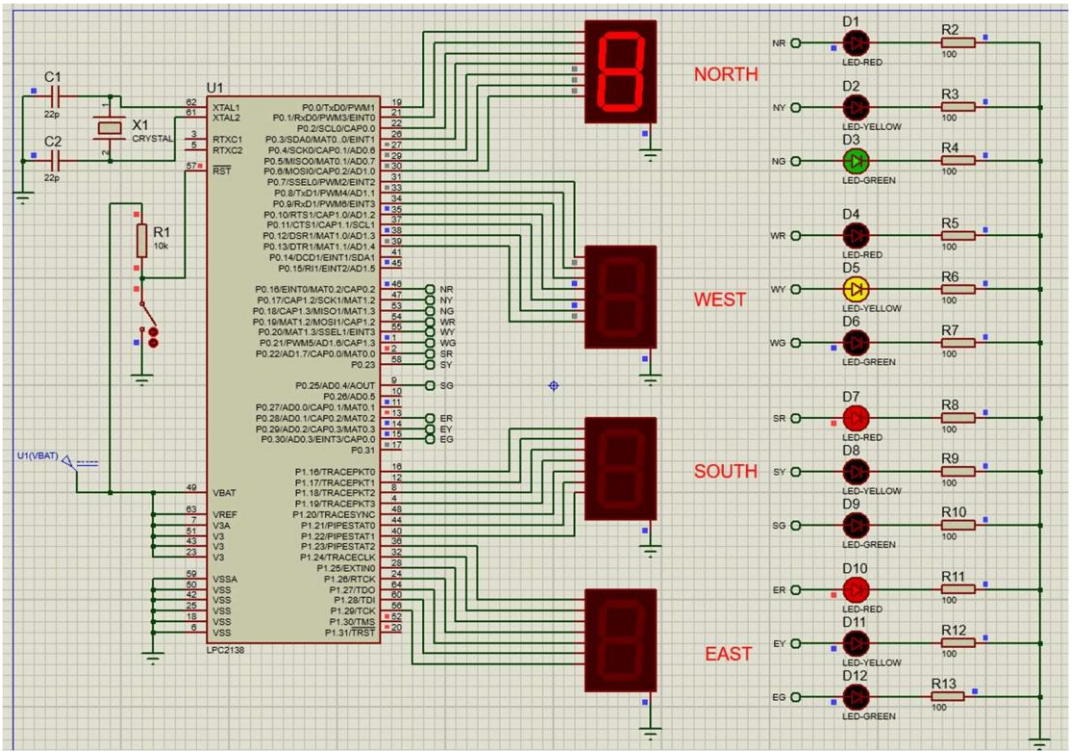
```
IOCLR0=0XFF<<7;
IOSET0=0X7D<<7;
delay();
IOCLR0=0XFF<<7;
IOSET0=0X6D<<7;
delay();
IOCLR0=0XFF<<7;
IOSET0=0X66<<7;
delay();
IOCLR0=0XFF<<7;
IOSET0=0X4F<<7;
delay();
IOCLR0=0XFF<<7;
IOSET0=0X5B<<7;
delay();
IOCLR0=0XFF<<7;
IOSET0=0X06<<7;
delay();
IOCLR0=0XFF<<7;
IOSET0=0X3F<<7;
delay();
IOCLR0=0XFF<<7;
    }
void segment3()
    {
 PINSEL2=0X00000000;
 IODIR1 |=0X7F0000;
 IOCLR1=0XFF<<16;
 IOSET1=0X6F<<16;
 delay();
 IOCLR1=0XFF<<16;
 IOSET1=0X7F<<16;
 delay();
 IOCLR1=0XFF<<16;
 IOSET1=0X07<<16;
 delay();
 IOCLR1=0XFF<<16;
 IOSET1=0X7D<<16;
 delay();
 IOCLR1=0XFF<<16;
 IOSET1=0X6D<<16;
 delay();
 IOCLR1=0XFF<<16;
 IOSET1=0X66<<16;
```

```
delay();
IOCLR1=0XFF<<16;
IOSET1=0X4F<<16;
delay();
IOCLR1=0XFF<<16;
IOSET1=0X5B<<16;
delay();
IOCLR1=0XFF<<16;
IOSET1=0X06<<16;
delay();
IOCLR1=0XFF<<16;
IOSET1=0X3F<<16;
delay();
IOCLR1=0XFF<<16;
   }
void segment4()
   {
PINSEL2=0X00000000;
IODIR1 |=0X3F800000;
IOCLR1=0XFF<<23;
IOSET1=0X6F<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X7F<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X07<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X7D<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X6D<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X66<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X4F<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X5B<<23;
delay();
IOCLR1=0XFF<<23;
```

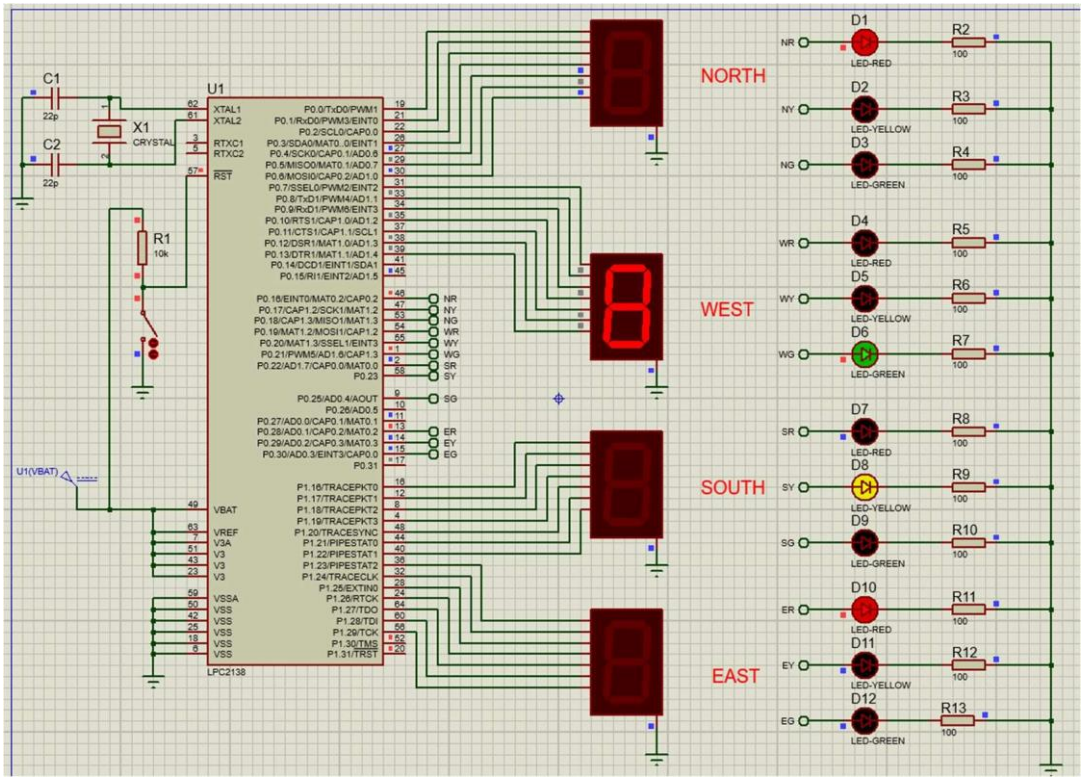
```
IOSET1=0X06<<23;
delay();
IOCLR1=0XFF<<23;
IOSET1=0X3F<<23;
delay();
IOCLR1=0XFF<<23;
    }
void delay()
    {
 int i,j;
for (i=0; i<2000; i++)
for (j = 0; j<2000; j++);
     }

int main()
    {
PINSEL1=0X00000000;
IODIR0=0X72FF0000;
while(1)
    {
IOCLR0=0X404A0000;
IOSET0=0X10540000;
segment1();
IOCLR0=0X10540000;
IOSET0=0X10A10000;
segment2();
IOCLR0=0X10A10000;
IOSET0=0X22090000;
segment3();
IOCLR0=0X22090000;
IOSET0=0X404A0000;
segment4();
     }
  }
```

Screenshots of Output:

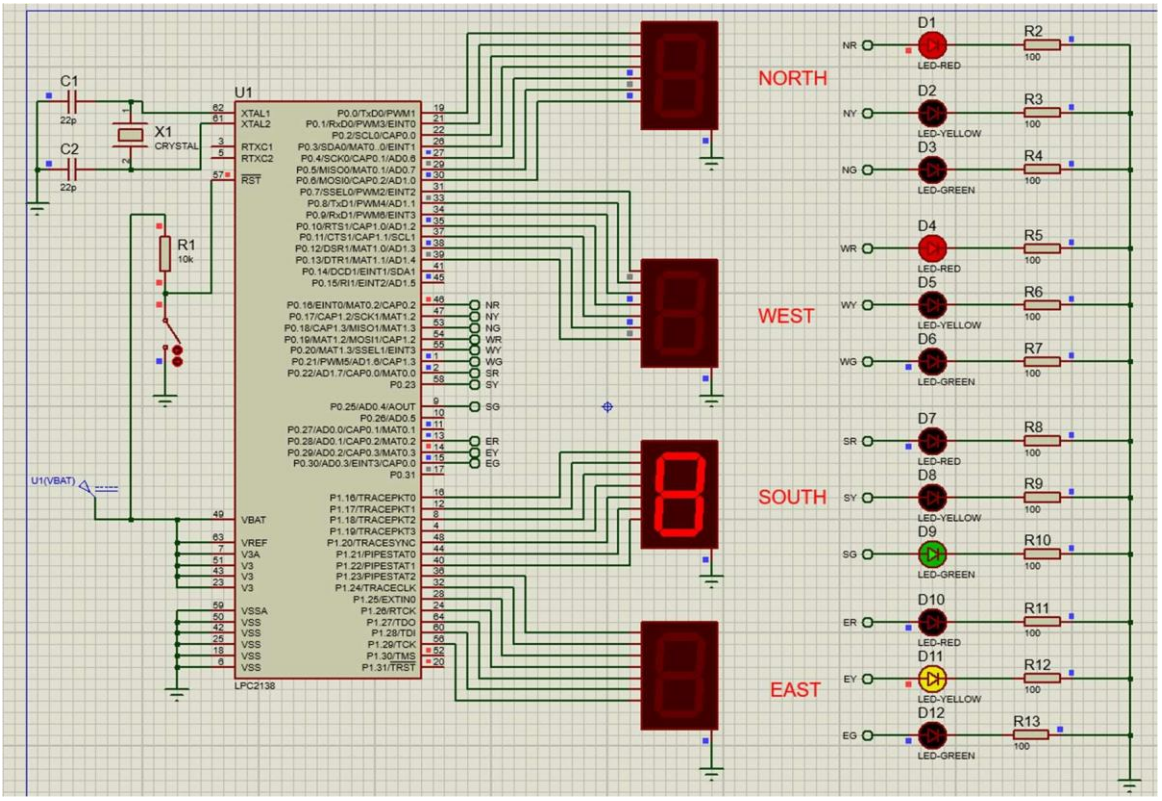
1. When North Light is Green:



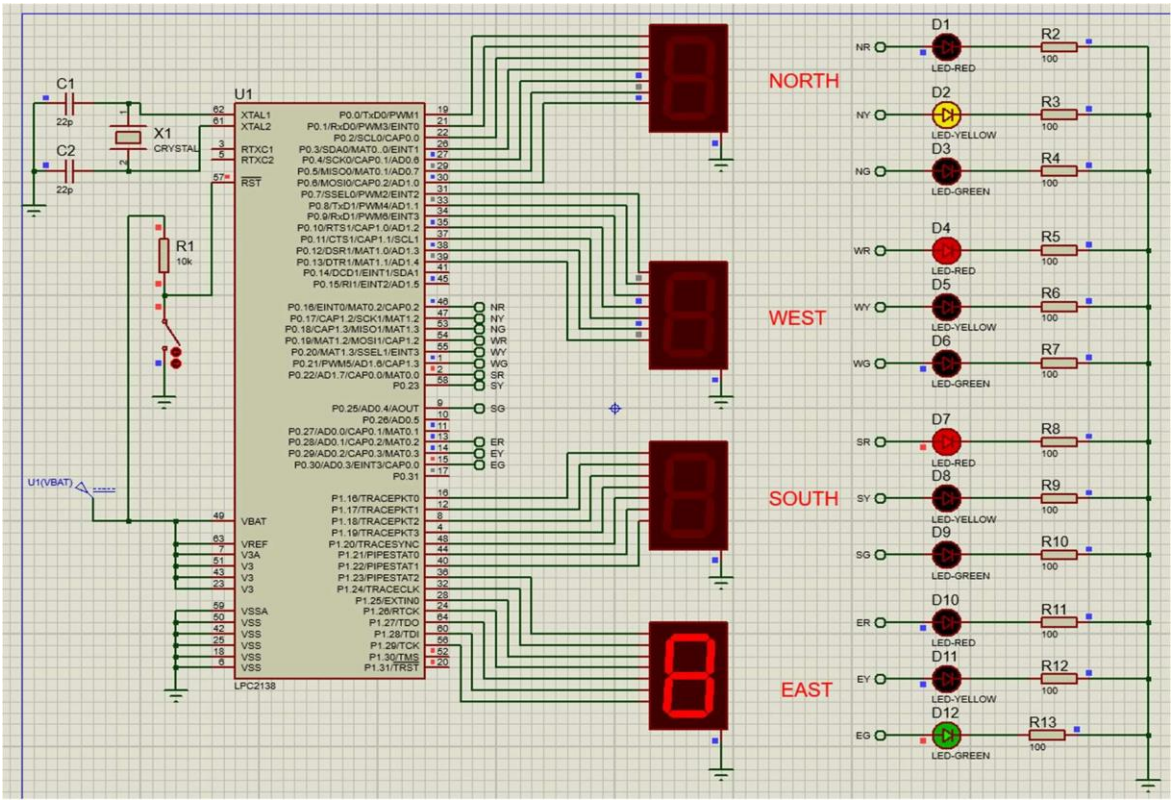
2. When West Light is Green:



3.When South Light is Green:



4. When East Light is Green:



Result:

C program to interface Traffic Light Controller with LPC 2148 using Keil micro vision 4 tool was written and the code was imported to proteus software to perform the simulation using Proteus tool .The simulation was successful, and output has been Verified.

Conclusion:

Implementing a Traffic Light Control system using the ARM7 LPC2148 microcontroller provides an efficient and flexible solution for regulating traffic flow at intersection.

****