**Timer and Port interrupt**

**Task 05**

****

Spring 2022

CSE-307 Microprocessor Based system Design

Submitted by: **Ashfaq Ahmad**

Registration No: **19PWCSE1795**

Class Section: **B**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted to:

**Dr: Bilal Habib**

**June** 6, 2022

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**Task:**

At a Motorway entry point in Peshawar, assume there are only two lanes to enter. One is for small vehicles and another one is for large ones. We have connected a sensor at each entry point. The sensor sends a signal (high-to-low edge) to our embedded system whenever a vehicle passes through the entry point and enters the Motorway. Use an 89C51 to count the number of vehicles passed through the entry point in one minute. As soon the one-minute time is over, it is indicated by turning ON a led at P3.1 pin, send the final value of small vehicles to Port-1 and larger ones to Port-2. Finally, the program goes into an infinite loop, doing nothing.

* Draw schematic along with timing diagram. The oscillator frequency is 12MHz.
* Entry of a vehicle can be simulated using a button press.
* Use two buttons: one for large vehicles and another one for smaller ones.
* Use port interrupts at P3.2 and P3.3.
* Use seven segments to display count of vehicles at Port 1 and Port 2.
* Use timers for creating a delay of 1 min.

**Hint:** Use timer interrupt and port interrupt. Use C language and Proteus ONLY

**Source Code:**

#include <reg51.h>

#include <stdio.h>

sbit test\_led=P3^1;

sbit entry\_lane1=P3^2;

sbit entry\_lane2=P3^3;

unsigned int x=0;

void timer() interrupt 3

{

x++;

if(x==5000)

{

IE=0;

test\_led=1;

}

else

{

TH1=0xD1; //we need 60 sec delay.

TL1=0x1F; //after each roll over it take 0.012 sec and 0.012x5000=60 sec.

}

}

void entry1() interrupt 0

{

P1=P1+1;

}

void entry2() interrupt 2

{

P2=P2+1;

}

void strt()

{

TR1=1;

}

void main(void)

{

entry\_lane1=1; //configure as input.

entry\_lane2=1; //configure as input.

test\_led=0;

P1=0;

P2=0;

IT0=1; //edge trigger external interrupt 0.

IT1=1; //edge trigger external interrupt 1.

IE=0x8D; //enable EX0, EX1, ET1.

TMOD=0x10; //timer1 mode 1 (16 bits)

TH1=0xD1;

TL1=0x1F;

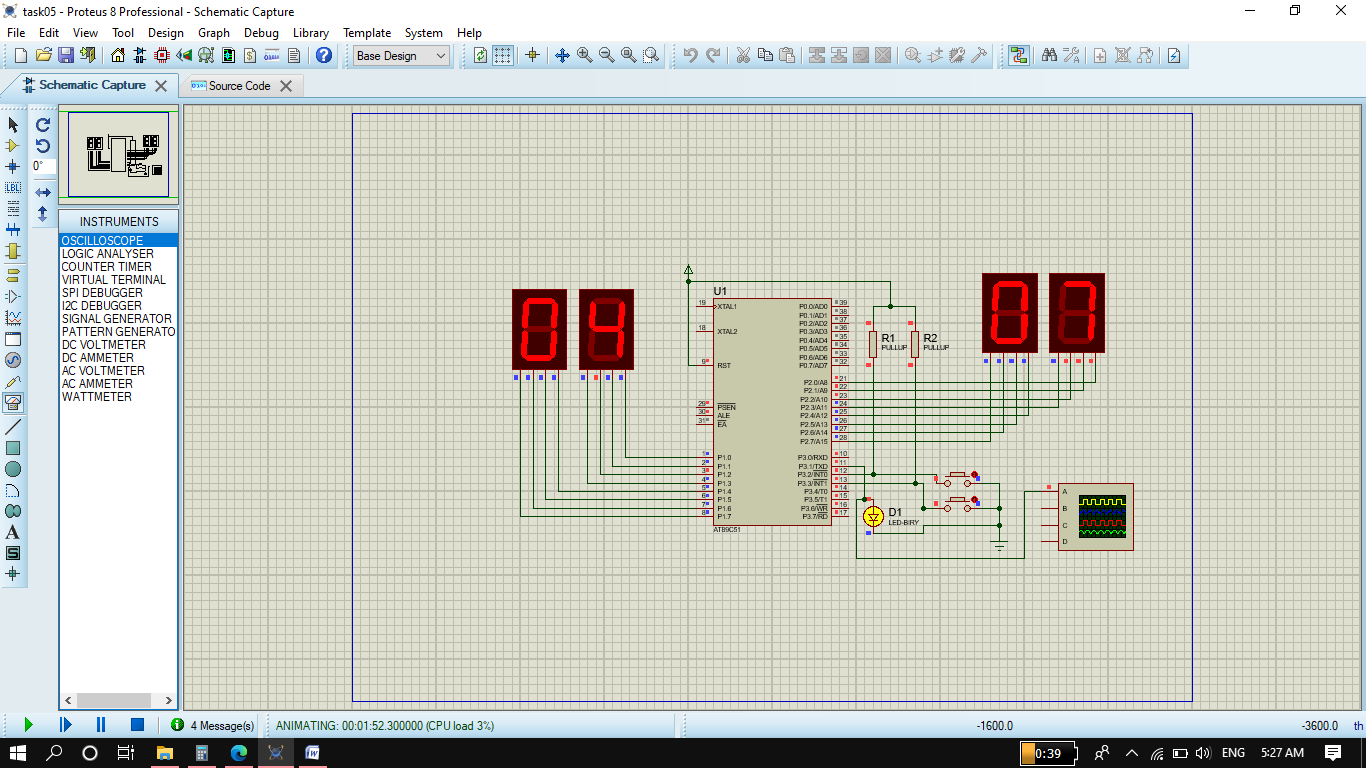
strt();

while (1);

}

**Circuit Diagram:**

**(Display is Hexadecimal)**



**Same Task but Display in Decimal:**

**Code:**

#include <reg51.h>

#include <stdio.h>

sbit test\_led=P3^1;

sbit entry\_lane1=P3^2;

sbit entry\_lane2=P3^3;

sbit unit1=P3^6;

sbit ten1=P3^7;

sbit unit2=P3^4;

sbit ten2=P3^5;

unsigned char arr[]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};

//{0,1,2,3,4,5,6,7,8,9} seven segemnt code in hex for active low (0). gfedcba

unsigned char u1,u2;

unsigned char t1,t2;

unsigned int count1=0;

unsigned int count2=0;

unsigned int x=0,y=0;

void timer() interrupt 3

{

     y++;

   if(y==5000)

   {

      IE=0;

      test\_led=1;

   }

   else

   {

      TH1=0xD1; //we need 60 sec delay.

      TL1=0x1F;  //after each roll over it take 0.012 sec and 0.012x5000=60 sec.

   }

}

void entry1() interrupt 0

{

count1++;

    u1=count1%10;

    t1=count1/10;

}

void entry2() interrupt 2

{

count2++;

  u2=count2%10;

    t2=count2/10;

}

void strt()

{

   TR1=1;

}

void main(void)

 {

    entry\_lane1=1;   //configure as input.

    entry\_lane2=1;  //configure as input.

    test\_led=0;

    P1=0;

    P2=0;

    IT0=1;        //edge trigger external interrupt 0.

    IT1=1;         //edge trigger  external interrupt 1.

    IE=0x8D;     //enable EX0, EX1, ET1.

    TMOD=0x10;   //timer1 mode 1 (16 bits)

    TH1=0xD1;

    TL1=0x1F;

    strt();

   while (1)

     {

            for(x=0; x<4000; x++);

            unit1=1; ten1=0;

            P1=arr[u1];

            for(x=0; x<4000; x++);

            unit1=0; ten1=1;

            P1=arr[t1];

            if(count1==100)

            count1=0;

            for(x=0; x<4000; x++);

            unit2=1; ten2=0;

            P2=arr[u2];

            for(x=0; x<4000; x++);

            unit2=0; ten2=1;

            P2=arr[t2];

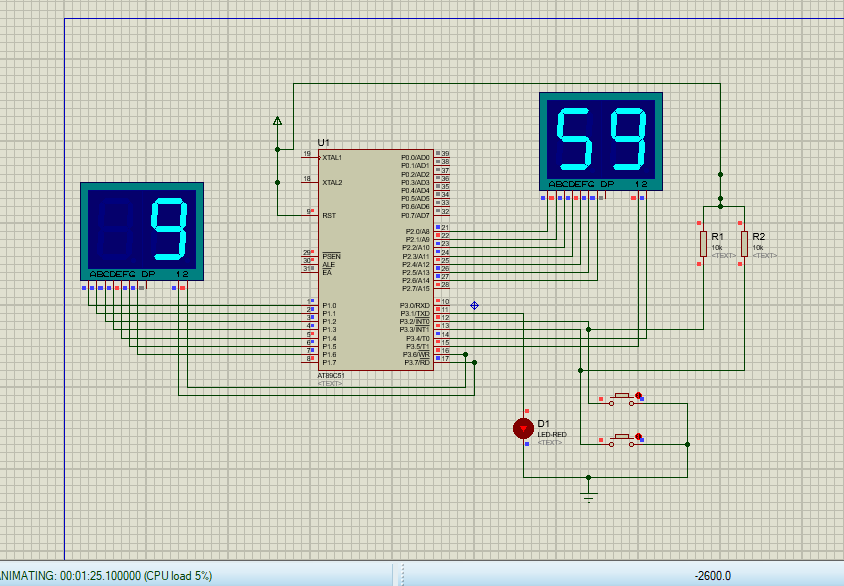
            if(count2==4000)

            count2=0;

     }

 }

**Circuit:**

****