EMBEDDED SYSTEM PRACTICAL

Name: **Arkate Samreen Gulam Haider**

Roll No: **32**

Class: **SYBSCIT**

Course Name: **B.SC. (I.T)SEM 4**

Subject: **Embedded System**

Subject code: **USIT4P2**

Academic Year: **2022-2023**

Name Of Faculty: **Heena Ansari**

Date: **04/03/2023**

DAAR-UL-REHMAT TRUST'S

A.E.KALSEKAR DEGREE COLLEGE

(PERMANENTLY AFFILIATED TO UNIVERSITY OF MUMBAI)

ADD: NEAR BHARAT GAS FACTORY, POST DAWLA,KAUSA-MUMBRA, DISTTHANE, PIN 400612, MAHARASHTRA(INDIA)

CERTIFICATE

THIS IS TO CERTIFY THAT ARKATE SAMREEN GULAM HAIDER

(EXAM SEAT NO:32) OF /SY/ B. SC. INFORMATION TECHNOLOGY CLASS HAS SATISFACTORY

COMPLETED / HER /PRACTICAL/ ON EMBEDDED SYSTEM FOR THE PARTIAL FULFILLMENT OF THE DEGREE BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY AS PRESCRIBED BY UNIVERSITY OF MUMBAI.

FOR ACADEMIC YEAR 2022 -2023

_____ _____

HOD PROFESSOR-IN-CHARG

INDEX

Practical No.	Details	Date	Sign
1.	Design and develop a reprogrammable embedded system computer using 8051 microcontroller and to show the following aspects. a. Programming b. Execution c. Debugging		
2.a	Configure timer control register of 8051 and develop a program to generate given time delay.		
b.	To demonstrate use of general purpose port i.e Input/Output port of two controller for data transfer between them.		
3.a	Port I/O: Use one of four ports of 8051 for o/p interfaced to eight LED's Simulate binary counter(8bit) on LED's.		
b	To interface 8 LEDs at Input-Output port and create different pattern.		
c	To demonstrate timer working in timer mode and blink LED without using any loop delay routine.		
4.a	Serial I/O: Configure 8051 serial port for asynchronous serial communication with serial port of PC exchange text message to PC and display on PC screen. Signify end of message by carriage return.		
b	To demonstrate interfacing of seven-segment LED display and generate counting from 0 to 99 with fixed time delay.		
c	Interface 8051 with D/A converter and generate square wave of given frequency on oscilloscope.		
5.a	Interface 8051 with D/A converter and generate triangular wave of given frequency on oscilloscope.		
b	Using D/A converter generate sine wave on oscilloscope with the help of lookup table stored in data area of 8051.		
6	Interface stepper motor with 8051 and write a program to move the motor thorough a given angle in clock wise or anti clock wise direction.		
7	Generate traffic signal.		

Practical No: 01

Aim: Design and develop a reprogrammable embedded system computer using 8051 microcontroller and to show the following aspects.

a. Programming

b. Execution

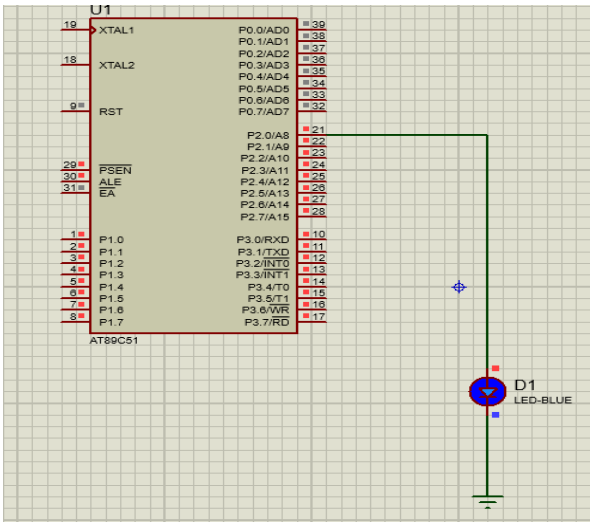
c. Debugging

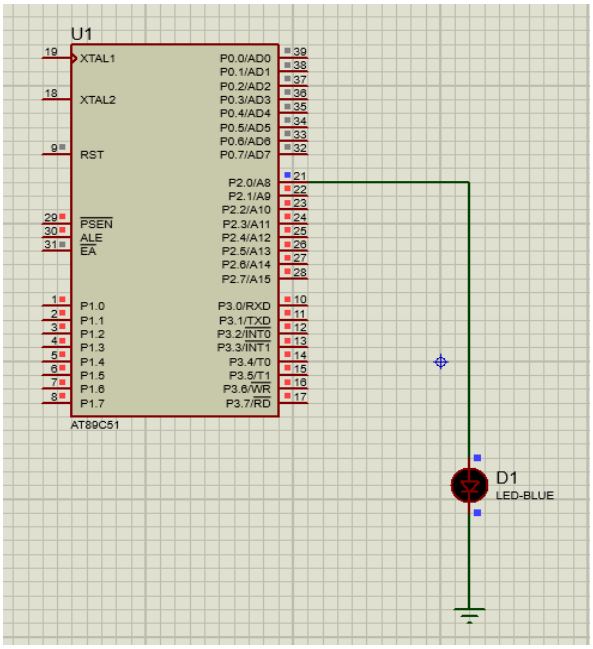
Input:

```
#include<reg52.H>

sbit LED = P2^0;

void Delay(void);

void main(void)

{

while(1)

{

LED = 0;

Delay();

LED = 1;

Delay();

}

}

void Delay(void)

{

int j;

int i;

for(i = 0; i < 10; i++)
```

```
{
for(j = 0; j < 10000; j++)
{
}
}
}
```

Output:





Practical No: 02

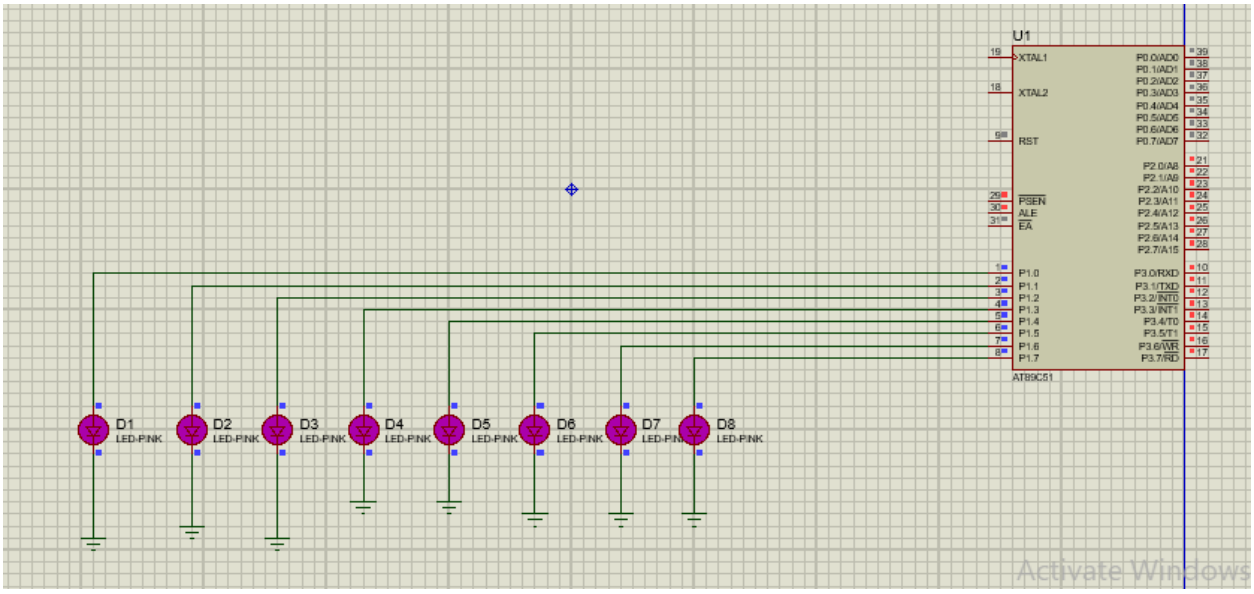
Aim: (A) Configure timer control register of 8051 and develop a program to generate given time delay.

Input:

```
#include<reg51.H>

void Delay(void);

void main(void)

{

while(1)

{

P1 = 0xFF;      // Make all bits of P1 high

Delay();

P1 = 0x00;      // Make all bits of P1 low

Delay();

}

}

void Delay(void)

{

int j;

int i;

for(i = 0; i < 1000; i++)

{

}

for(j = 0; j < 1000; j++)

{

}
```

}

Output:


U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
AT89C51
D1
LED-PINK
D2
LED-PINK
D3
LED-PINK
D4
LED-PINK
D5
LED-PINK
D6
LED-PINK
D7
LED-PINK
D8
LED-PINK
Activate Windows

(B): To demonstrate use of general purpose port i.e. Input/ output port of two controllers for data transfer between them.

Controller 1

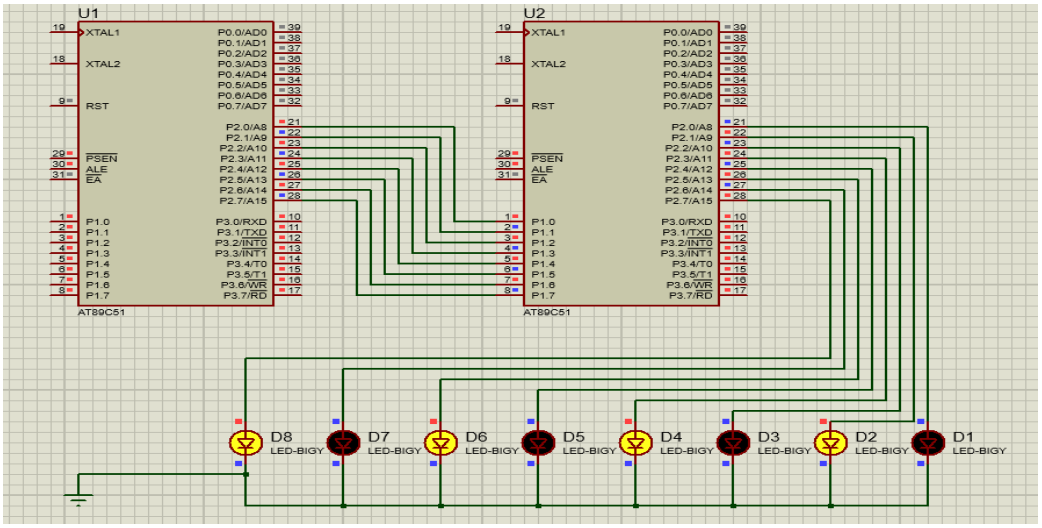
```
#include<reg51.h>

void Delay(unsigned int time);

void main(void){

P2 = 0xAA;

Delay(1000);

P2 = 0x55;

Delay(1000);

}

void Delay(unsigned int time){

unsigned int i,j;

for (i = 0; i < time; i++){

for (j = 0; j < 23; j++){}

}

}
```

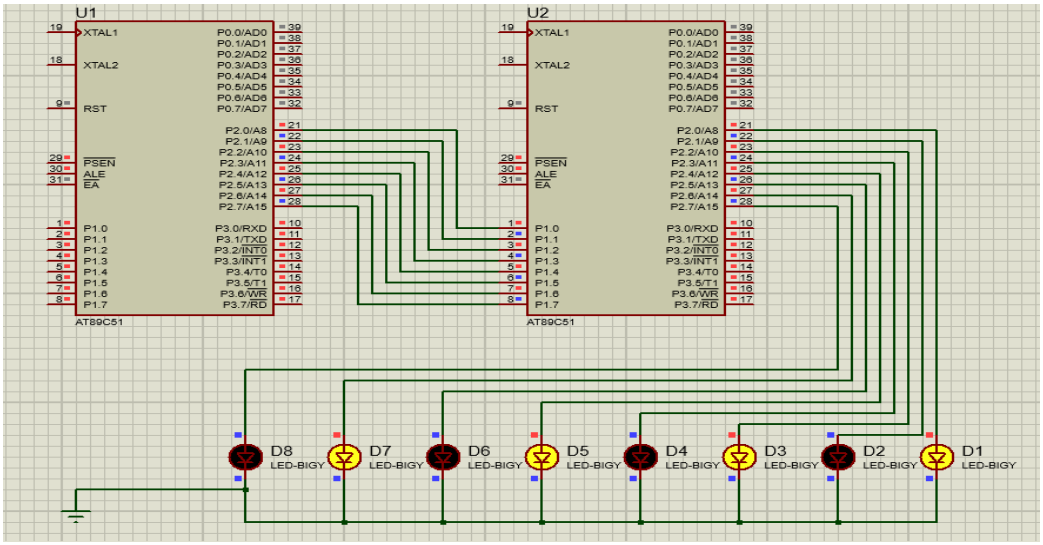
Controller 2

```
#include<reg51.h>

void Delay(unsigned int time);

void main(void){

while(1){

P2 = P1;

Delay(1000);

}
```

```
}
void Delay(unsigned int time){
unsigned int i,j;
for (i = 0; i < time; i++){
for (j = 0; j < 23; j++){
}
}
}
```

Output:





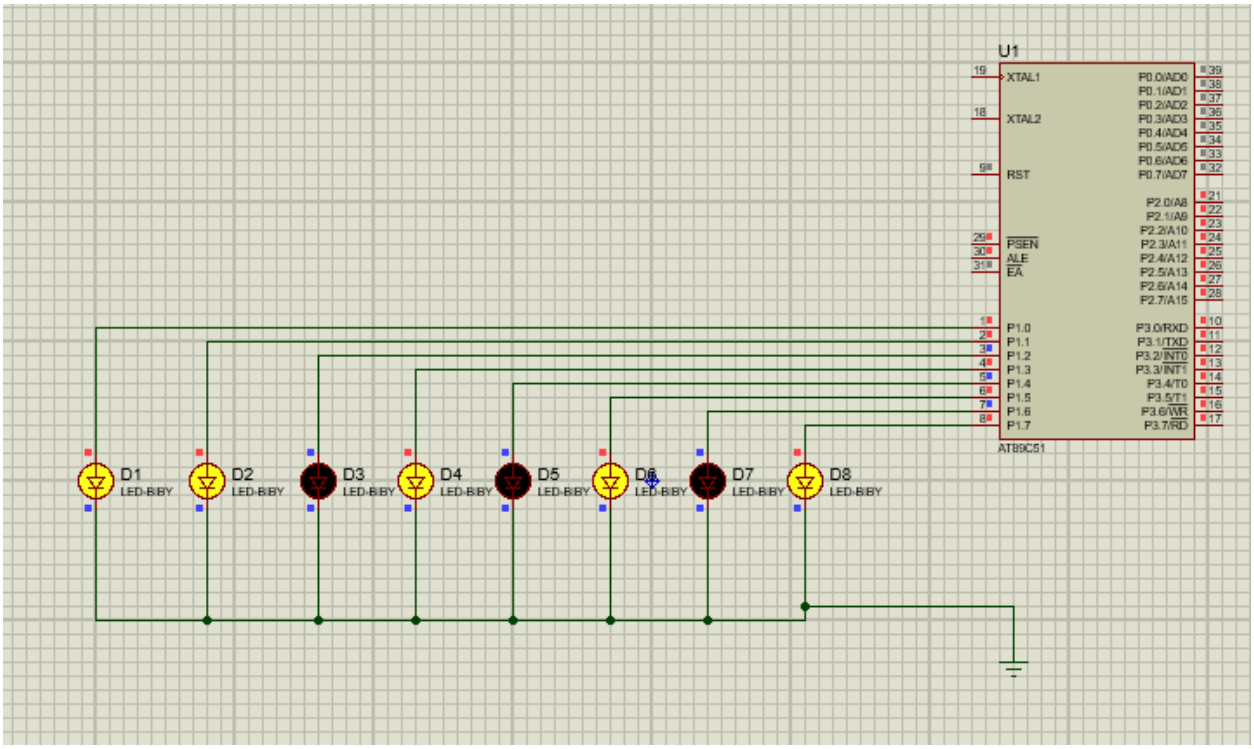
Practical No: 03

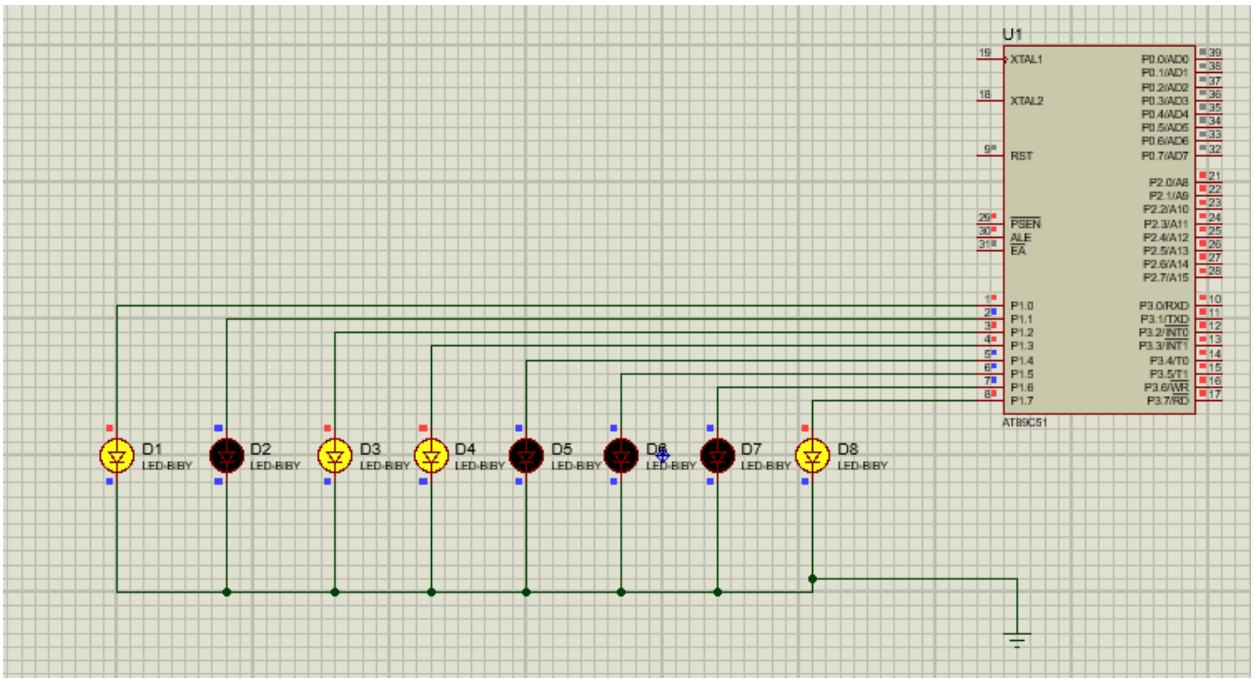
Aim:(A) Port I/O: Use one of four ports of 8051 for o/p interfaced to eight LED's Simulate binary counter(8bit) on LED's.

Input:

```
#include<reg51.h>

void delay(int time);

void main()

{

P1 = 00000000;

while(1)

{

P1++;

delay(100);

}

}

void delay(int time)

{

int i, j;

for(i = 0; i <= time; i++)

{

for(j = 0; j <= 23; j++)

{

}

}

}
```

Output:


U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
ATB9C51
D1 LED-BIBY
D2 LED-BIBY
D3 LED-BIBY
D4 LED-BIBY
D5 LED-BIBY
D6 LED-BIBY
D7 LED-BIBY
D8 LED-BIBY

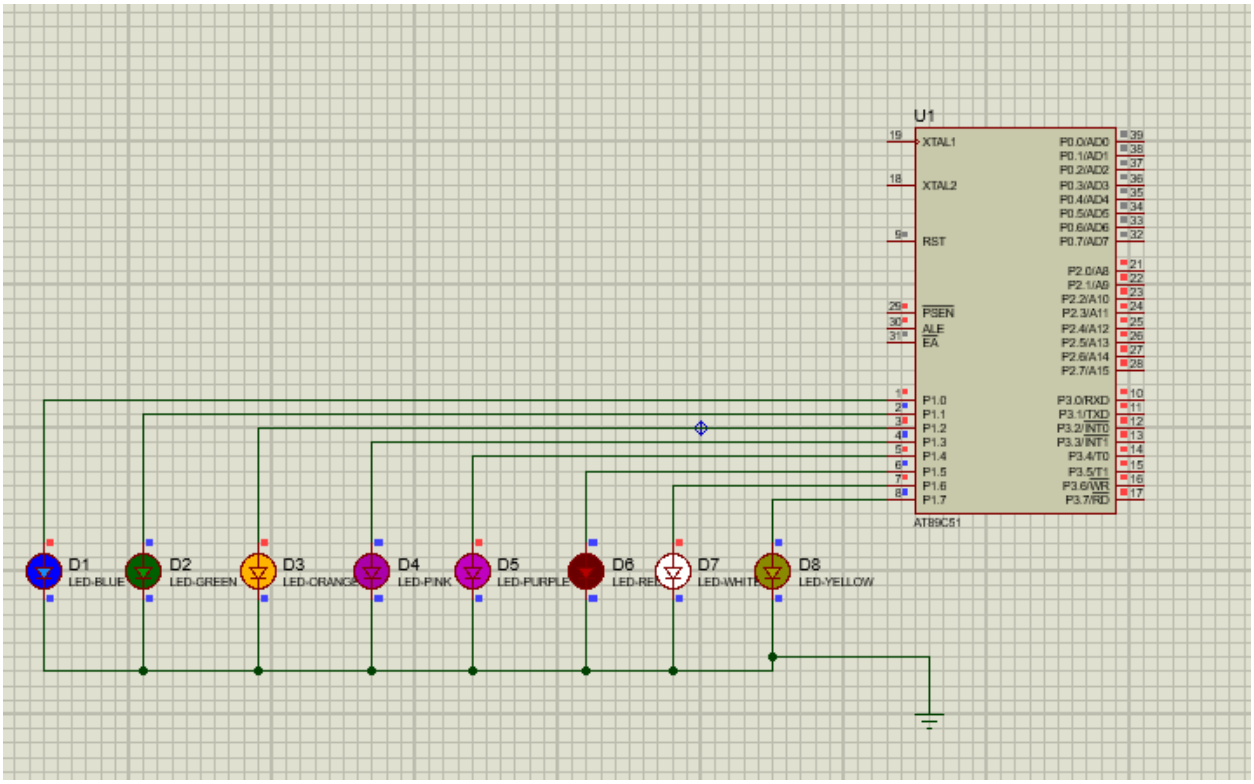

U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
ATB9C51
D1 LED-BIBY
D2 LED-BIBY
D3 LED-BIBY
D4 LED-BIBY
D5 LED-BIBY
D6 LED-BIBY
D7 LED-BIBY
D8 LED-BIBY

(B): To interface 8 LEDs at Input-Output port and create different pattern.

Input:

```
#include<reg51.h>
void delay();
void main()
{
while(1)
{
P1 = 0xAA; // Pattern to turn on alternate LEDs - 10101010
delay();
P1 = 0x55; // Reverse the pattern
delay();
}
}
void delay()
{
unsigned int i, j;
for(i = 0; i < 23; i++)
{
for(j = 0; j < 1000; j++)
{
}
}
}
```

Output:

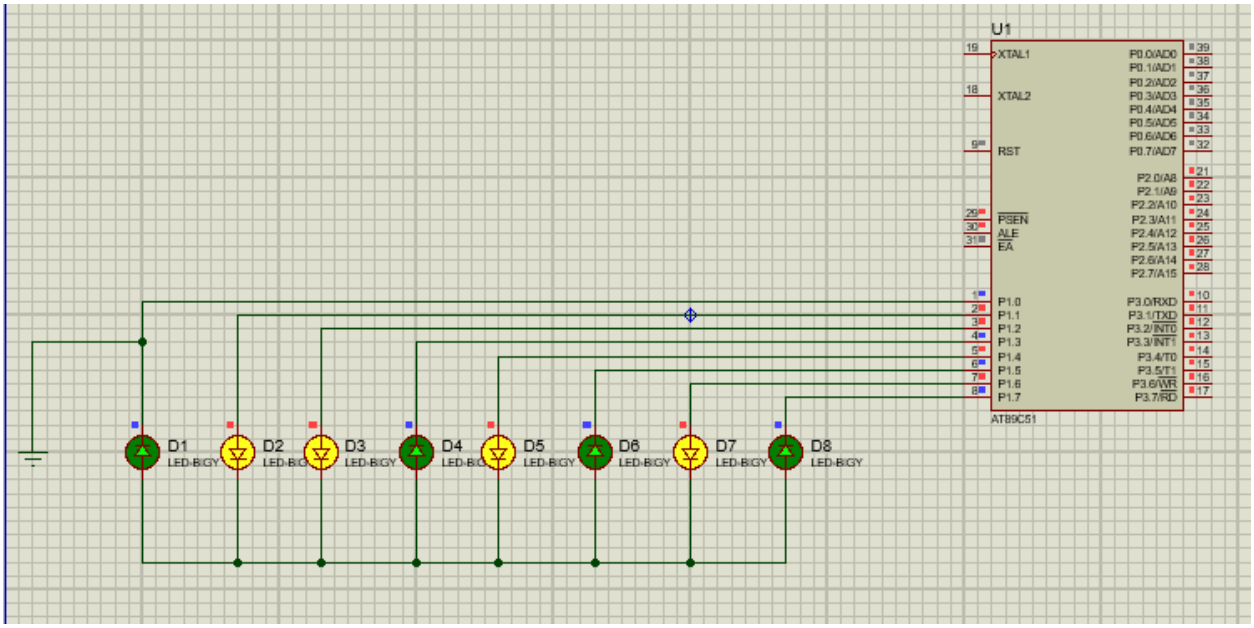

U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
AT89C51
D1
LED-BLUE
D2
LED-GREEN
D3
LED-ORANGE
D4
LED-PINK
D5
LED-PURPLE
D6
D7
D8
LED-YELLOW

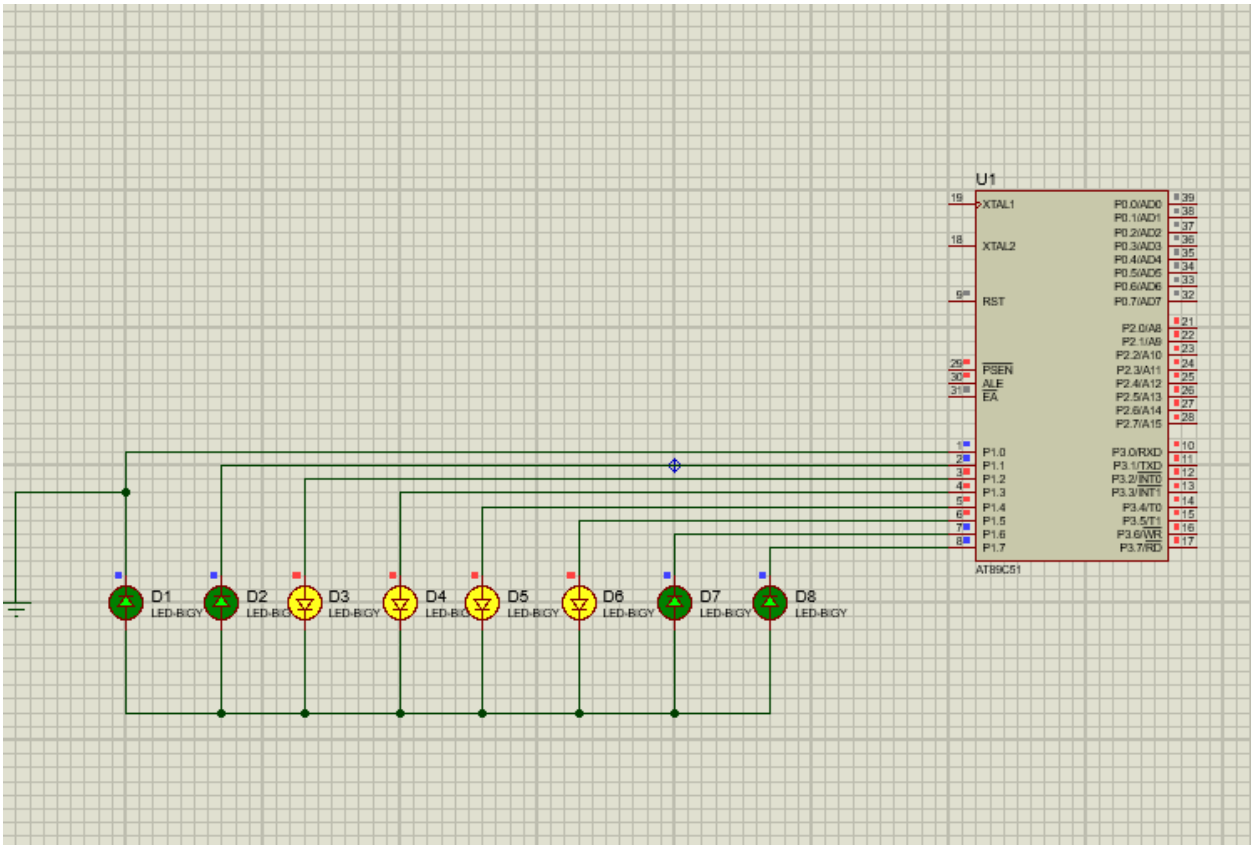
(C): To demonstrate timer working in timer mode and blink LED without using any loop delay routine.

Input:

```
#include<reg51.h>

int i = 0;

void timer_ISR(void)interrupt 1

{

i++;

if(i == 10)

{

i=0;

P1++;

}

}

void main(void)

{

TMOD = 0x01;

ET0 = 1;

TR0 = 1;

EA = 1;

while(1);

}
```

Output:


U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
ATB9C51
D1
D2
D3
D4
D5
D6
D7
D8
LED-BIGY


U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
ATB9C51
D1
D2
D3
D4
D5
D6
D7
D8
LED-BIGY

Practical No: 04

Aim:(A): Serial I/O: Configure 8051 serial port for asynchronous serial communication with serial port of PC exchange text message to PC and display on PC screen. Signify end of message by carriage return.

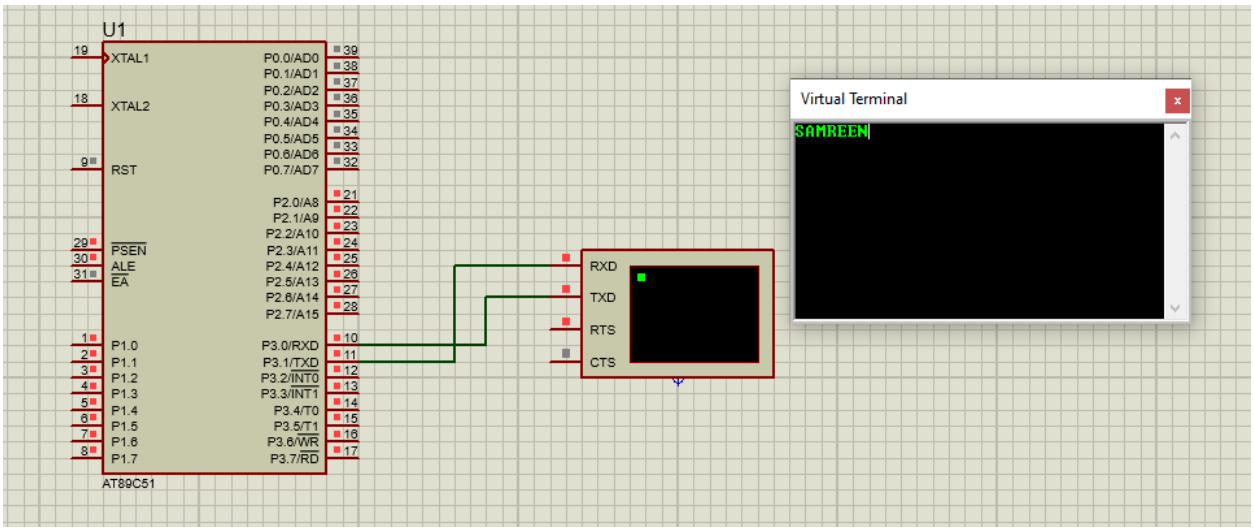
Input:

```
#include<reg51.h>

void send(char x);

void main(void)

{

TMOD = 0x20;

TH1 = 0xFD;

SCON = 0x50;

TR1 = 1;

send('S');

send('A');

send('M');

send('R');

send('E');

send('E');

send('N');

while(1);

}

void send(char x)

{

SBUF = x;

while(TI == 0);
```

TI = 0;

}

Output:


U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
P0.0/AD0
P0.1/AD1
P0.2/AD2
P0.3/AD3
P0.4/AD4
P0.5/AD5
P0.6/AD6
P0.7/AD7
P2.0/A8
P2.1/A9
P2.2/A10
P2.3/A11
P2.4/A12
P2.5/A13
P2.6/A14
P2.7/A15
P3.0/RXD
P3.1/TXD
P3.2/INT0
P3.3/INT1
P3.4/T0
P3.5/T1
P3.6/WR
P3.7/RD
AT89C51
RXD
TXD
RTS
CTS
Virtual Terminal
SAMREEN

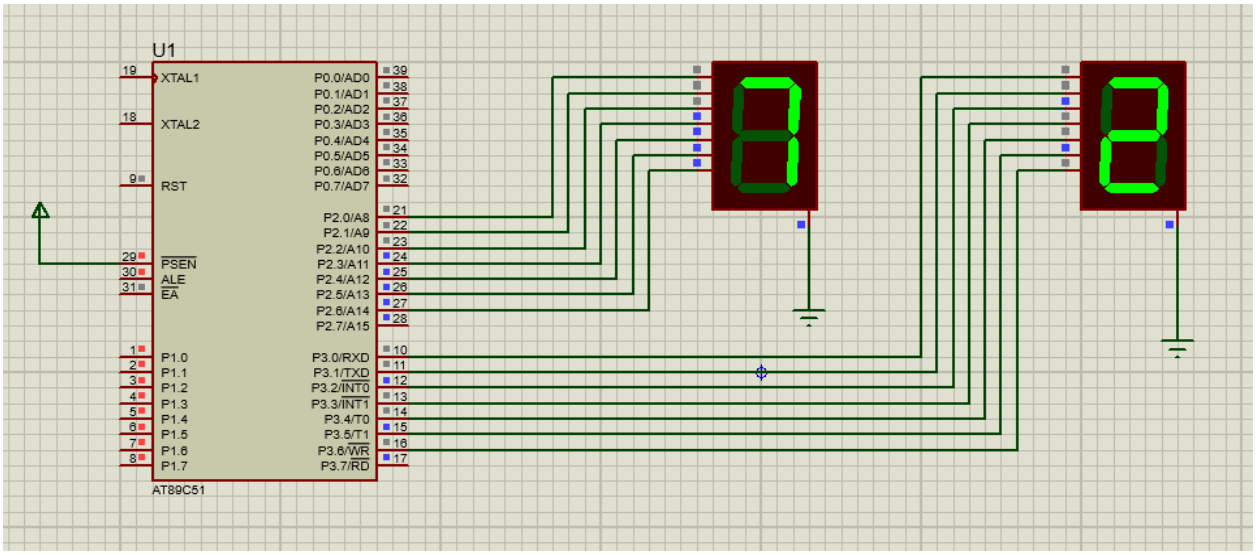
(B): To demonstrate interfacing of seven-segment LED display and generate counting from 0 to 99 with fixed time delay.

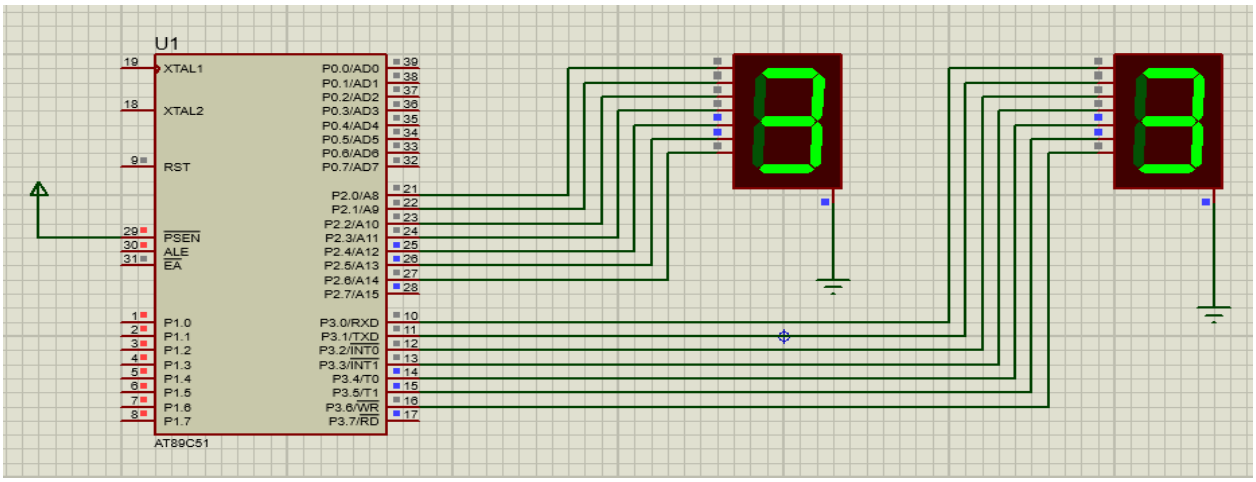
Input:

```
#include<reg51.h>
void delay(unsigned int ms)
{
unsigned int i, j;
for(i = 0; i<ms; i++)
{
for(j = 0; j<=1275; j++)
{
}
}
}
  void main(void)
{
char number[] = {0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F};
int i, j;
P2 = 0x00;
P3 = 0x00;
while(1)
{
for(i = 0; i<=9; i++)
{
P2 = number[i];
```

```
for(j = 0; j<=9; j++)

{

P3 = number[j];

delay(50);

}

}

}

}
```

Output:



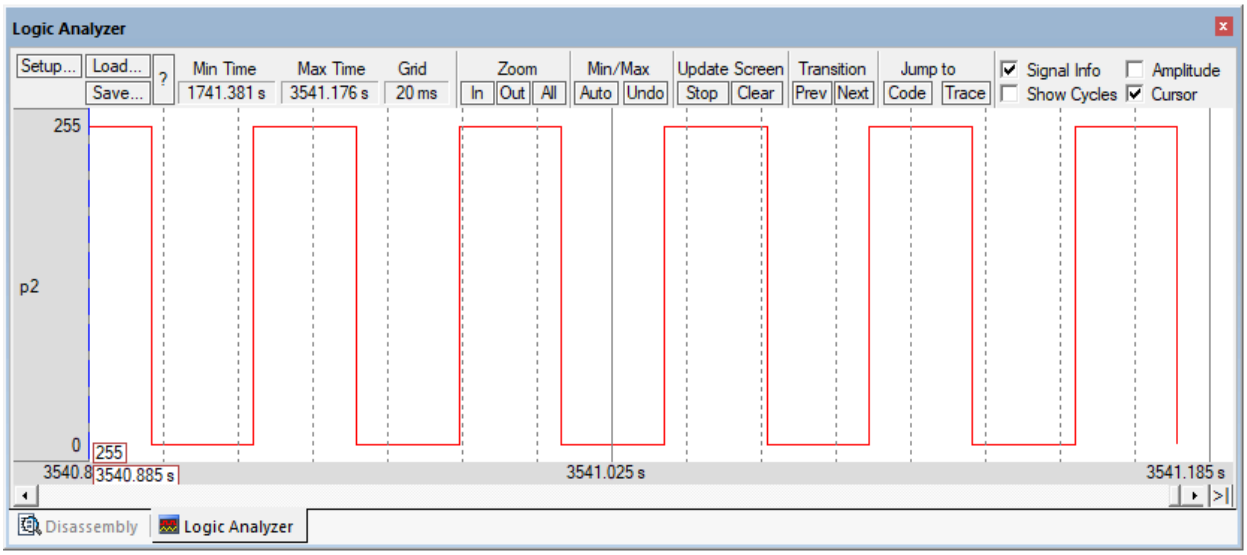


(C): Interface 8051 with D/A converter and generate square wave of given frequency on oscilloscope.

Input:

```
#include<reg51.h>

void delay();

void main()

{

P2 = 0x00;

while(1)

{

P2 = 0xFF;

delay();

P2 = 0x00;

delay();

}

}

void delay()

{

int i;

for(i = 0; i <= 5000; i++)

{

}

}
```

Output:


Logic Analyzer
Setup...
Load...
Save...
?
Min Time
1741.381 s
Max Time
3541.176 s
Grid
20 ms
Zoom
In
Out
All
Min/Max
Auto
Undo
Update Screen
Stop
Clear
Transition
Prev
Next
Jump to
Code
Trace
Signal Info
Show Cycles
Amplitude
Cursor
255
p2
0
255
3540.8
3540.885 s
3541.025 s
3541.185 s
Disassembly
Logic Analyzer

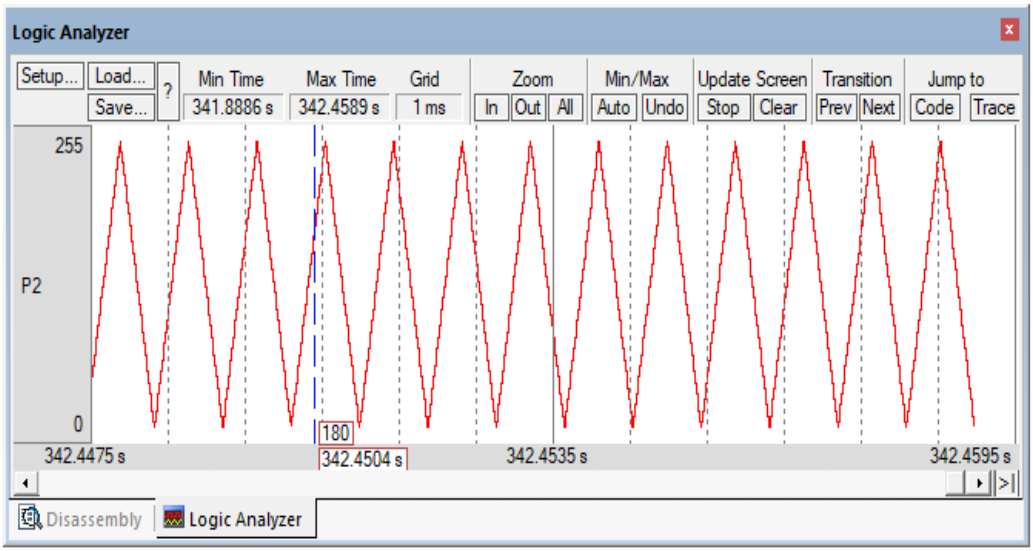
Practical No: 05

Aim:(A): Interface 8051 with D/A converter and generate triangular wave of given frequency on oscilloscope.

Input:

```
#include<reg51.h>

void main()

{

P2= 0x00;

while(1)

{

do

{

P2 += 0x05;

}

while(P2 < 0xFF);

do

{

P2 -= 0x05;

}

while(P2 > 0x00);

}

}
```

Output:

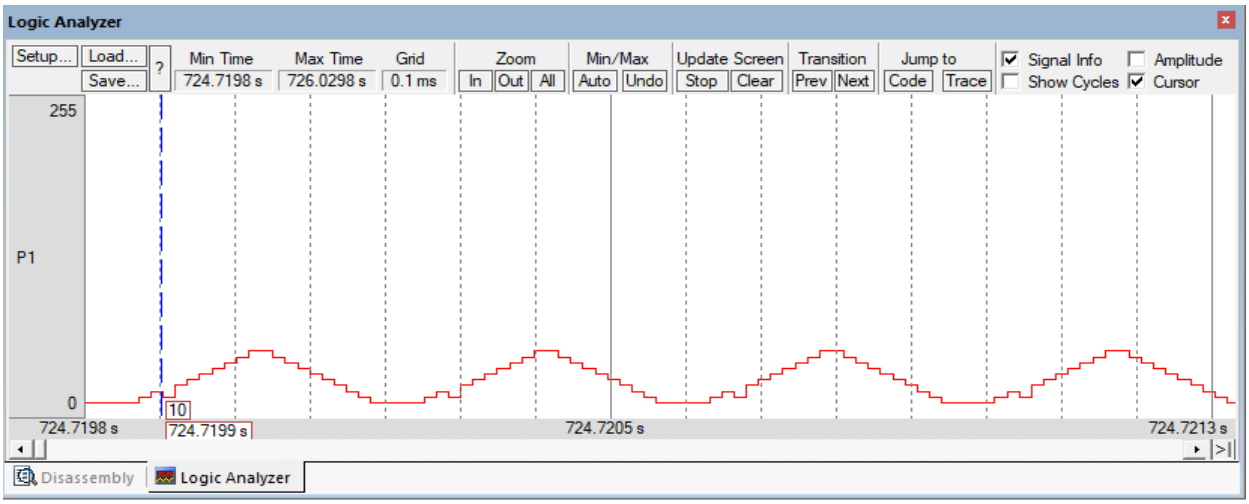

Logic Analyzer
Setup...
Load...
Save...
?
Min Time
341.8886 s
Max Time
342.4589 s
Grid
1 ms
Zoom
In
Out
All
Min/Max
Auto
Undo
Update Screen
Stop
Clear
Transition
Prev
Next
Jump to
Code
Trace
255
P2
0
180
342.4475 s
342.4504 s
342.4535 s
342.4595 s
Disassembly
Logic Analyzer

(B): Using D/A converter generate sine wave on oscilloscope with the help of lookup table stored in data area of 8051.

Input:

```
#include<reg51.h>

#include<stdio.h>

int sine[]={0,0,5,10,5,15,20,25,30,35,40,45,45,40,35,30,25,20,15,10,5,0,0};

void main()

{

int i;

P1=0x00;

while(1)

{

for(i=0;i<23;i++)

{

P1=sine[i];

}

}
```

Output:



Practical No: 06

Aim: Interface stepper motor with 8051 and write a program to move the motor thorough a given angle in clock wise or anti clock wise direction.

Input:

```
#include<reg51.h>

void delay()

{

int i, j;

for(i = 0; i <= 100; i++)

{

for(j = 0; j < 100; j++)

{

}

}

}

void main()

{

while(1)

{

P2 = 0x09;

delay();

P2 = 0x03;

delay();

P2 = 0x06;

delay();
```

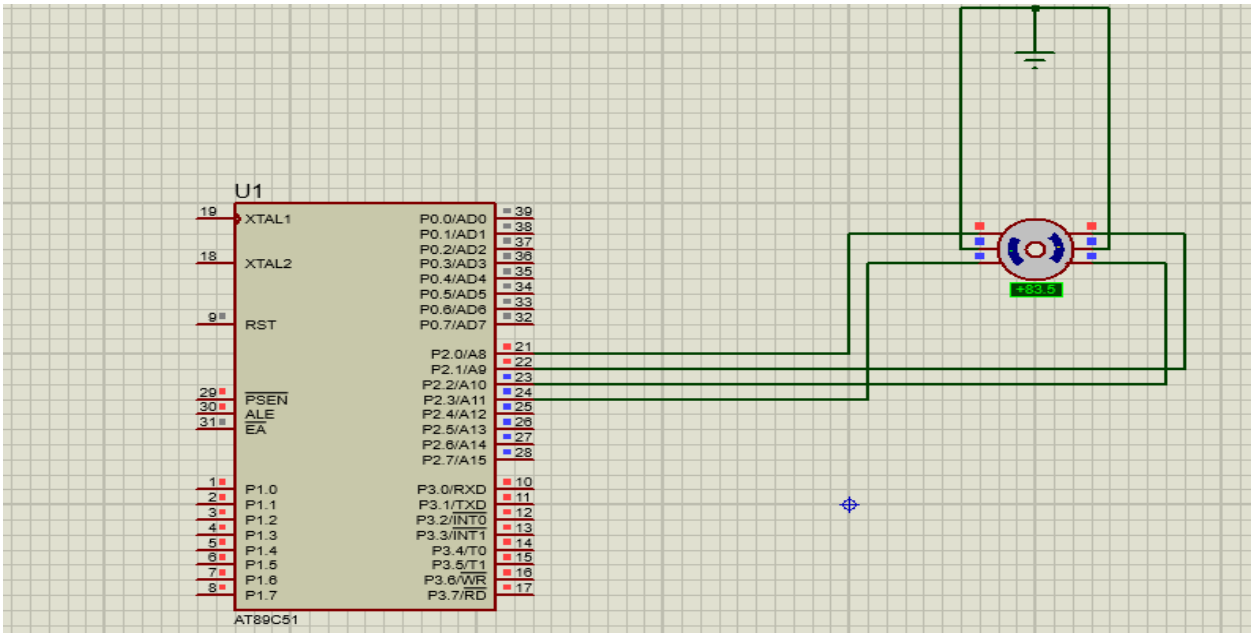
P2 = 0x0C;

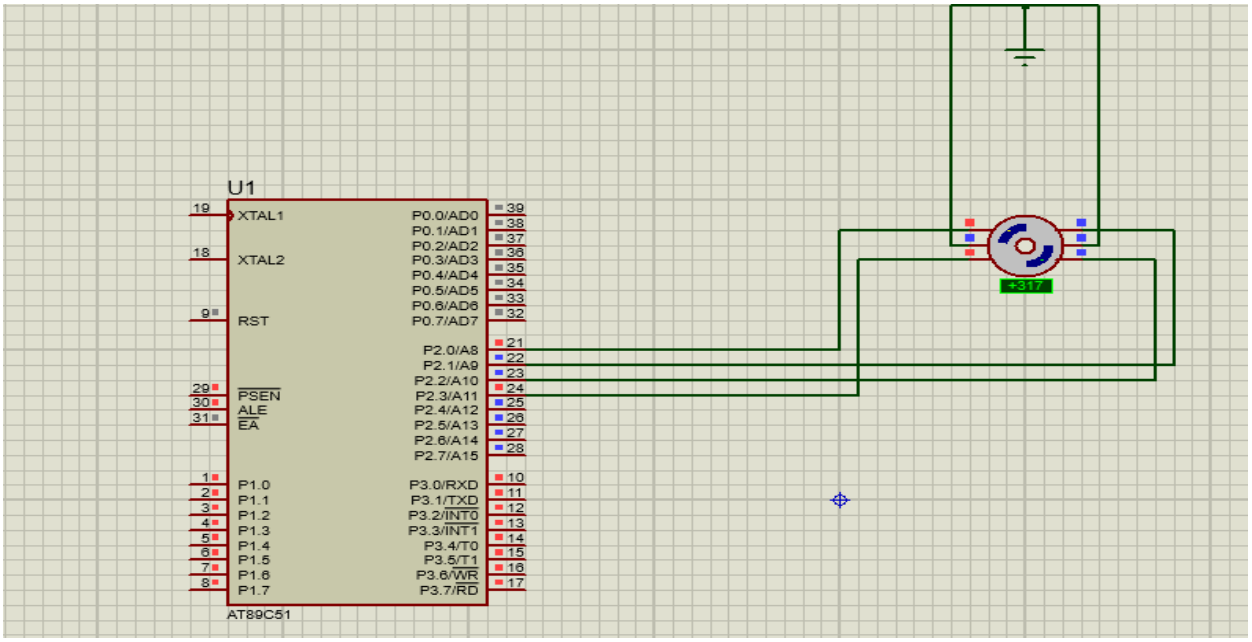
delay();

}

}

Output:





Practical No: 07

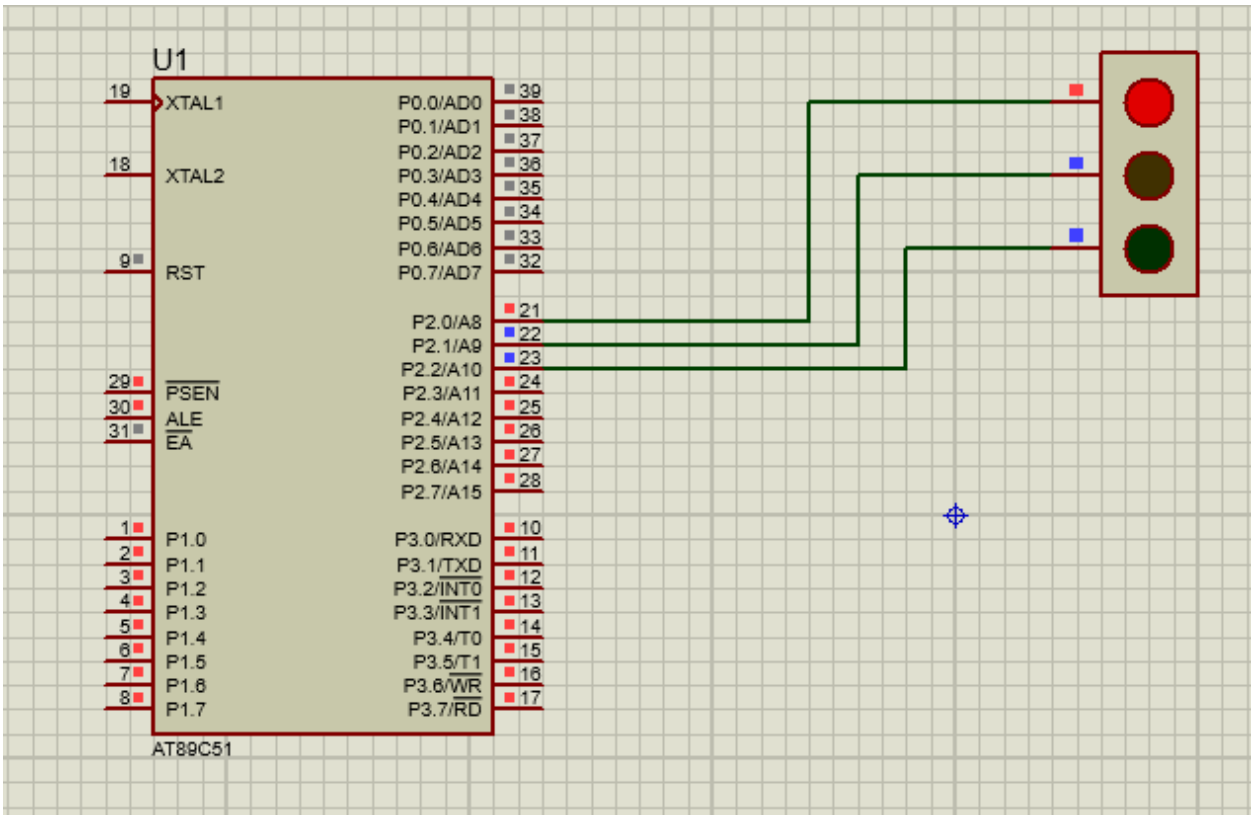
Aim: Generate traffic signal.

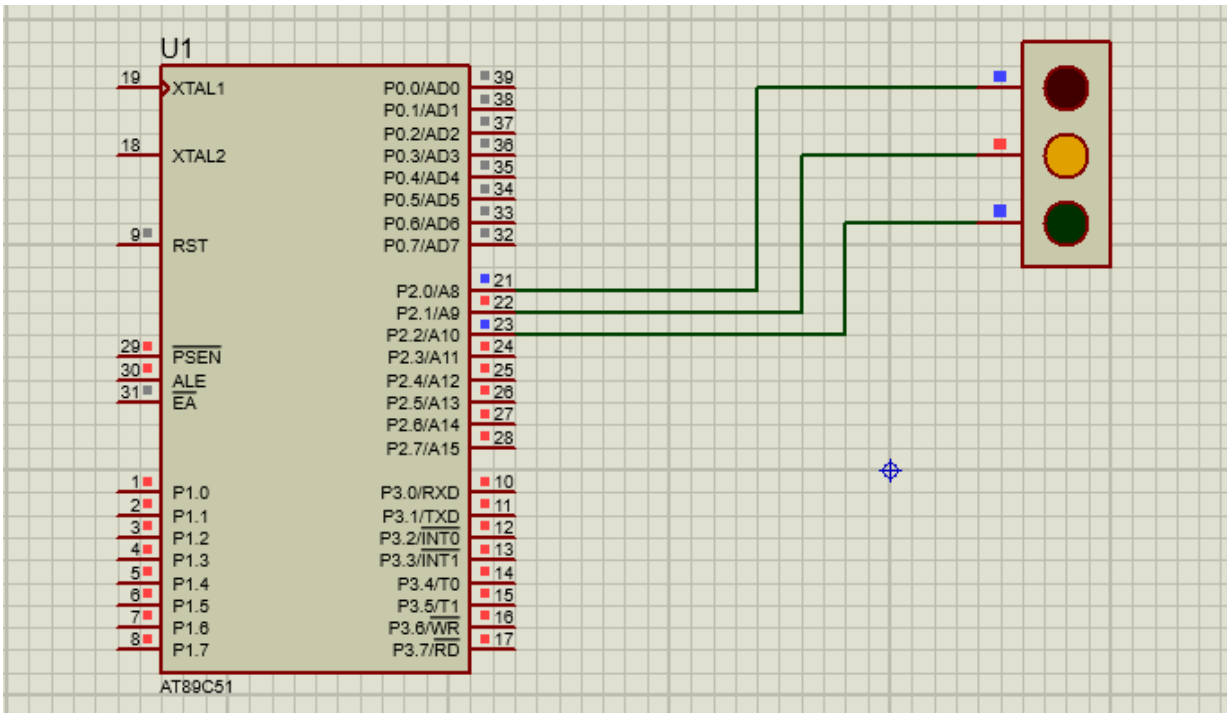
Input:

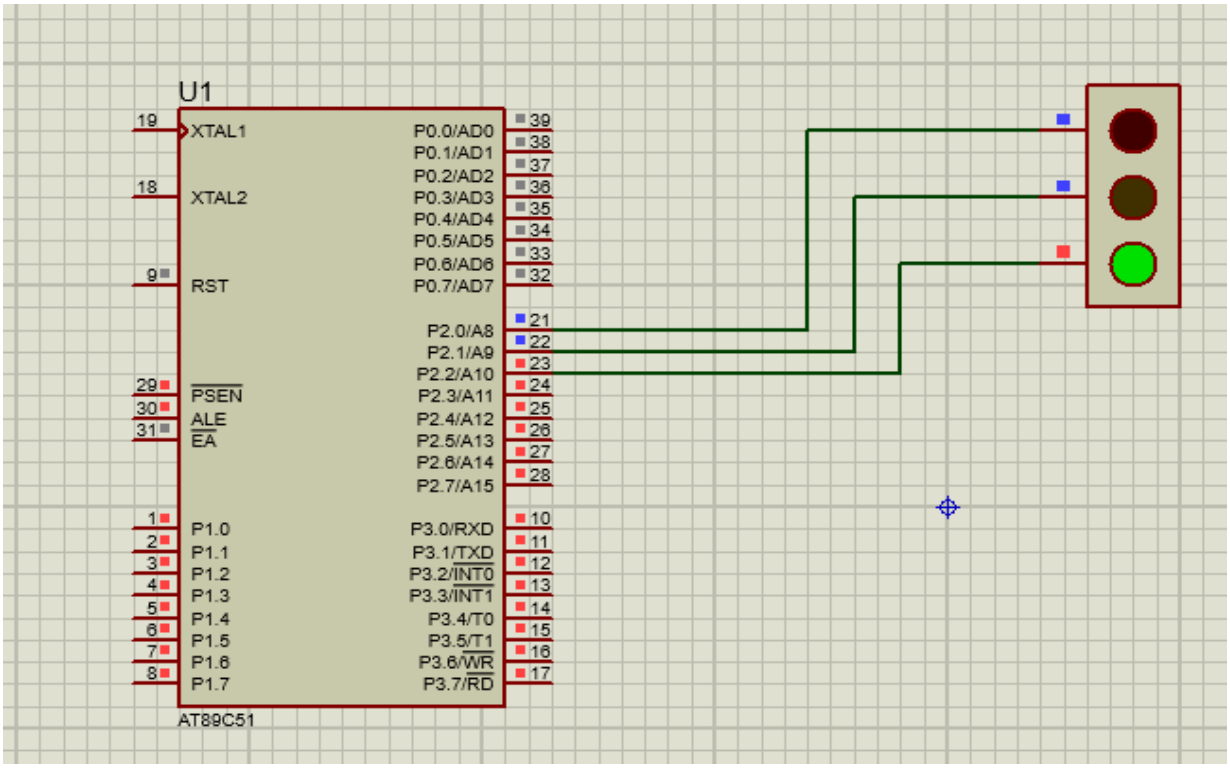
```
#include<reg51.h>
sbit red = P2^0;
sbit yellow = P2^1;
sbit green = P2^2;
void delay(int time);
void main()
{
red = yellow = green = 0;
while(1)
{
red = 1;
delay(1000);
red = 0;
yellow = 1;
delay(200);
yellow = 0;
green = 1;
delay(1000);
green = 0;
yellow = 1;
delay(200);
yellow = 0;
```

```
}

}

void delay(int time)

{

int i, j;

for(i = 0; i < time; i++)

{

for(j = 0; j < 1000; j++)

{

}

}

}
```

Output:




U1
19
XTAL1
18
XTAL2
9
RST
29
PSEN
30
ALE
31
EA
1
P1.0
2
P1.1
3
P1.2
4
P1.3
5
P1.4
6
P1.5
7
P1.6
8
P1.7
P0.0/AD0
P0.1/AD1
P0.2/AD2
P0.3/AD3
P0.4/AD4
P0.5/AD5
P0.6/AD6
P0.7/AD7
P2.0/A8
P2.1/A9
P2.2/A10
P2.3/A11
P2.4/A12
P2.5/A13
P2.6/A14
P2.7/A15
P3.0/RXD
P3.1/TXD
P3.2/INT0
P3.3/INT1
P3.4/T0
P3.5/T1
P3.6/WR
P3.7/RD
AT89C51


U1
19
XTAL1
18
XTAL2
9
RST
29
PSEN
30
ALE
31
EA
1
P1.0
2
P1.1
3
P1.2
4
P1.3
5
P1.4
6
P1.5
7
P1.6
8
P1.7
P0.0/AD0
P0.1/AD1
P0.2/AD2
P0.3/AD3
P0.4/AD4
P0.5/AD5
P0.6/AD6
P0.7/AD7
P2.0/A8
P2.1/A9
P2.2/A10
P2.3/A11
P2.4/A12
P2.5/A13
P2.6/A14
P2.7/A15
P3.0/RXD
P3.1/TXD
P3.2/INT0
P3.3/INT1
P3.4/T0
P3.5/T1
P3.6/WR
P3.7/RD
AT89C51