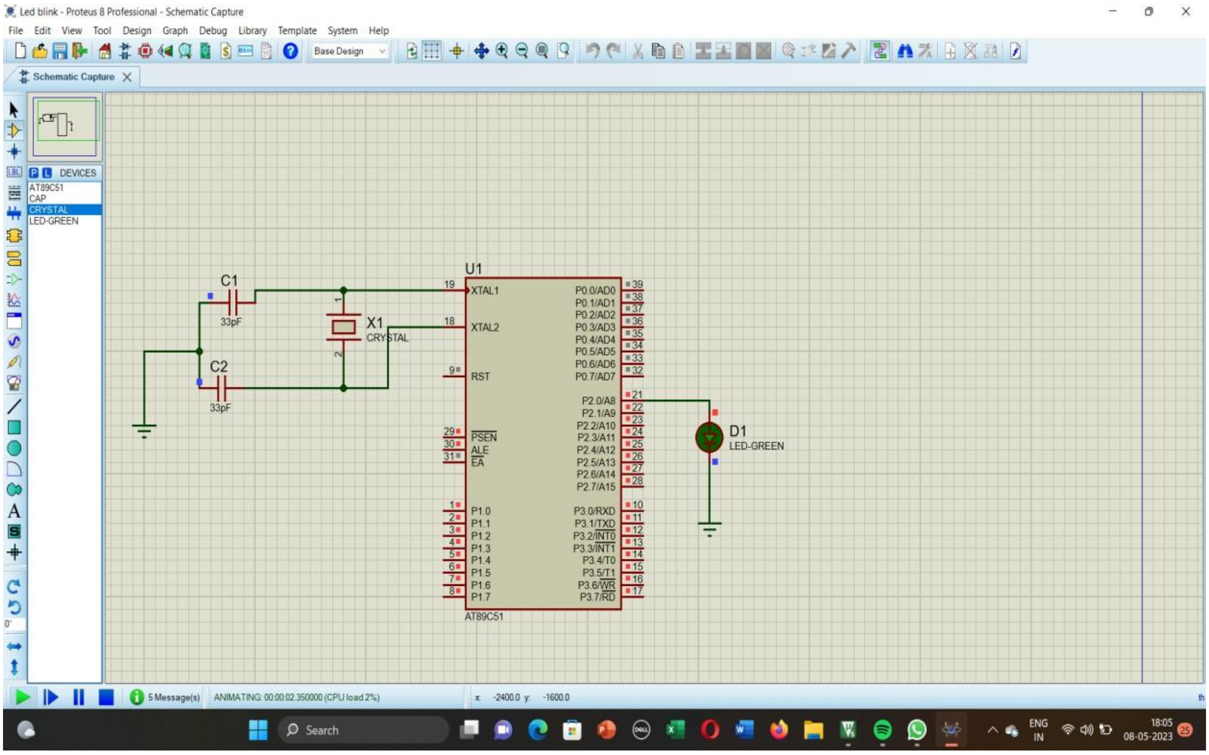
Experiment 1 [Led Blinking using 8051]

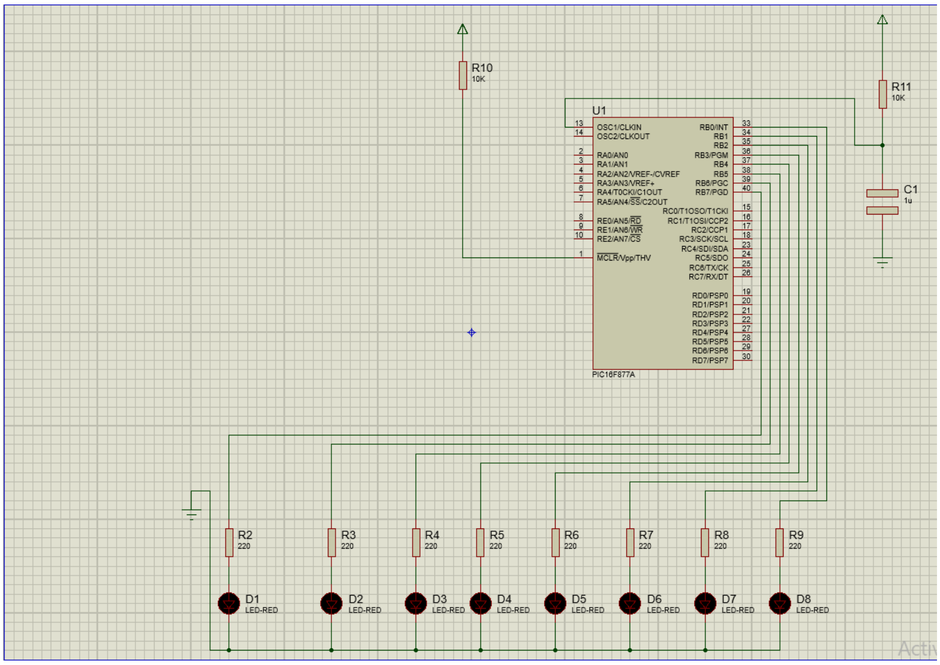
https://youtu.be/Qr1D7DK9qaY

```
ORG 0000H
UP: SETB P2.0
ACALL DELAY
CLR P2.0
ACALL DELAY
SJMP UP
DELAY: MOV R4,#35
H1:MOV R3,#255
H2:DJNZ  R3,H2
DJNZ R4,H1
RET
END
```



Experiment 2 LED FADE IN AND OFF IN 8051

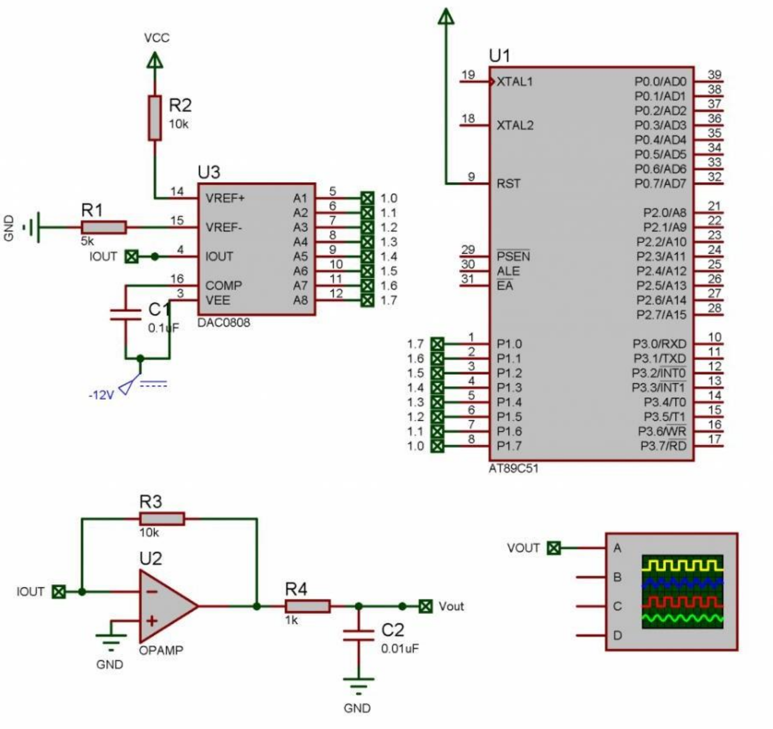
```
#include <REGX52.h>
delay(unsigned int y){
   unsigned int i,j;
for(i=0;i<y;i++){
   for(j=0;j<1275;j++){}
}
}
main(){
  while(1){
   delay(100);
     P1_0 = 0;
     delay(100);
     P1_0 = 1;
   }
}
```



Experiment 3 Genrate square wave using 8051

https://youtu.be/8ne8LAuEh9w

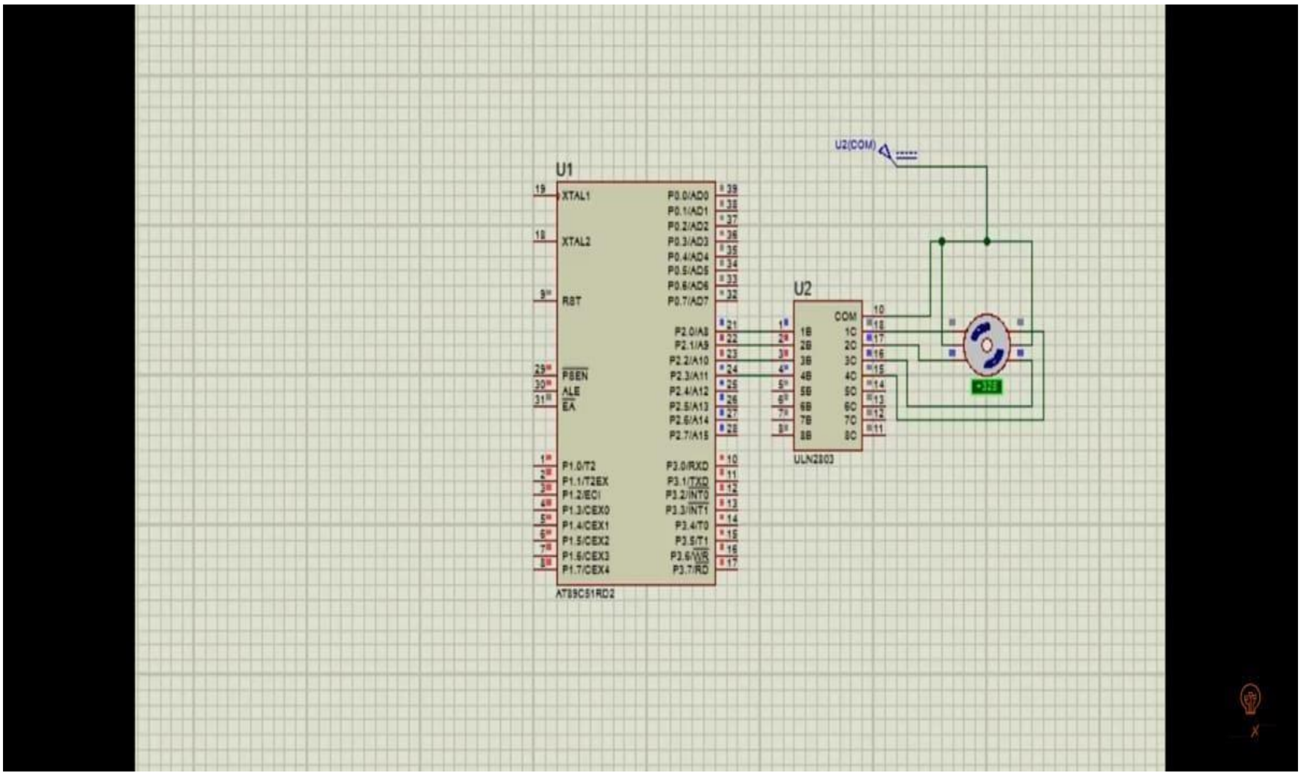
```
ORG 0000H

UP:MOV P2,#00H

ACALL DELAY

MOV P2,#OFFH

ACALL DELAY

SJMP UP

DELAY:MOV R4,#100

H1:MOV R3,#255

H2:DJNZ R3,H2

DJNZ R4,H1

RET

END
```



Experiment 4 Stepper Motor using 8051

https://youtu.be/hyuJbL39hPc

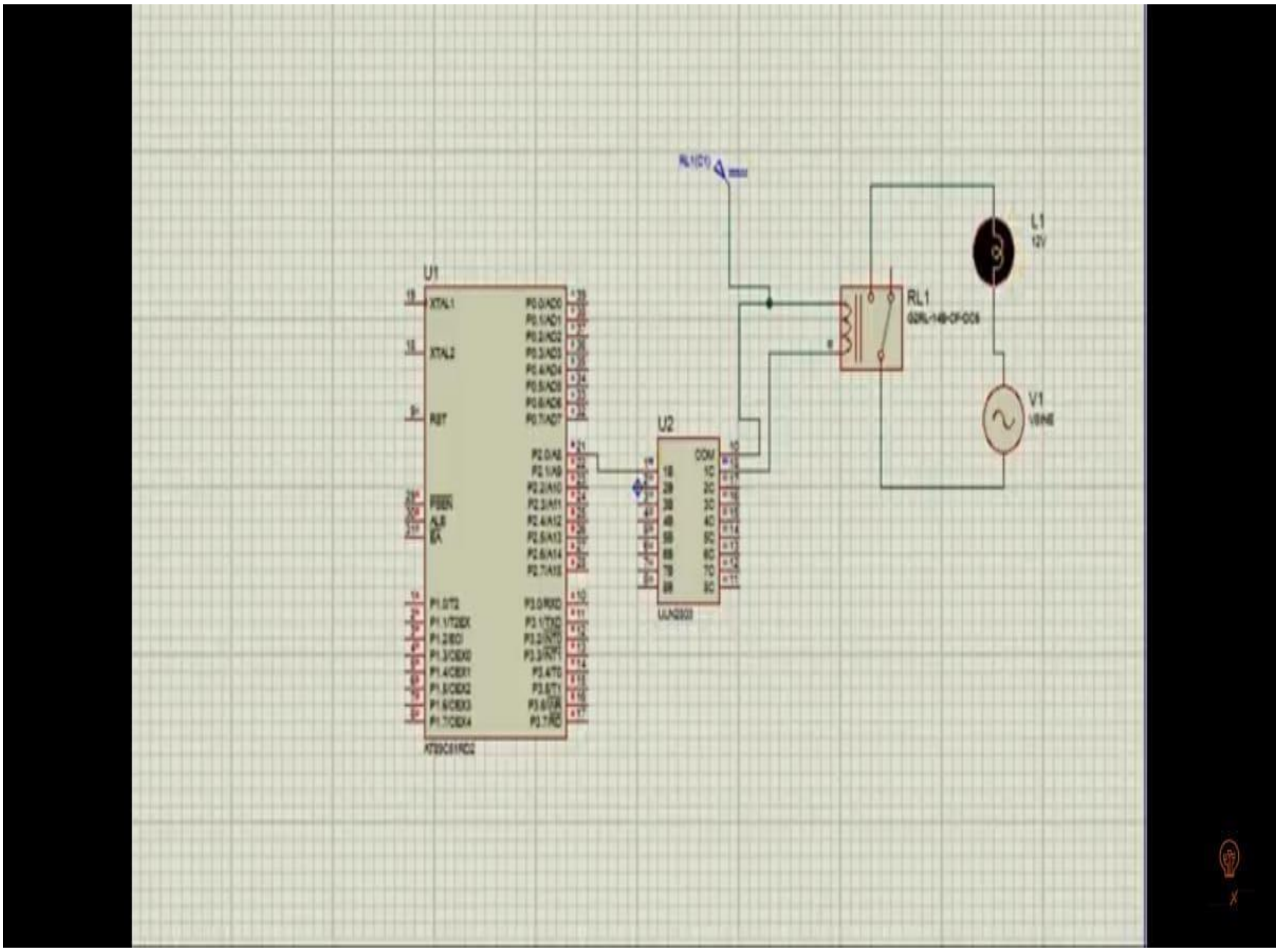
```
ORG 0000H
UP: MOV P2,#09H
ACALL DELAY
MOV P2,06H
ACALL DELAY
MOV P2,#06H
ACALL DELAY
SJMP UP

DELAY:MOV R4,#18
H1:MOV R3,#255
H2:DJNZ R3,H2
DJNZ R4,H1
RET
END
```



Experiment 5 Interfacing of Relay using 8051

https://youtu.be/f7e84HAkc58

```
ORG 0000H
  UP:SETB P2.0
        ACALL DELAY
        SJMP UP

        DELAY:MOV R4,#18
        H1:MOV R3,255
        H2:DJNZ R3,H2
        DJNZ R4,H1
        RET
        END
```



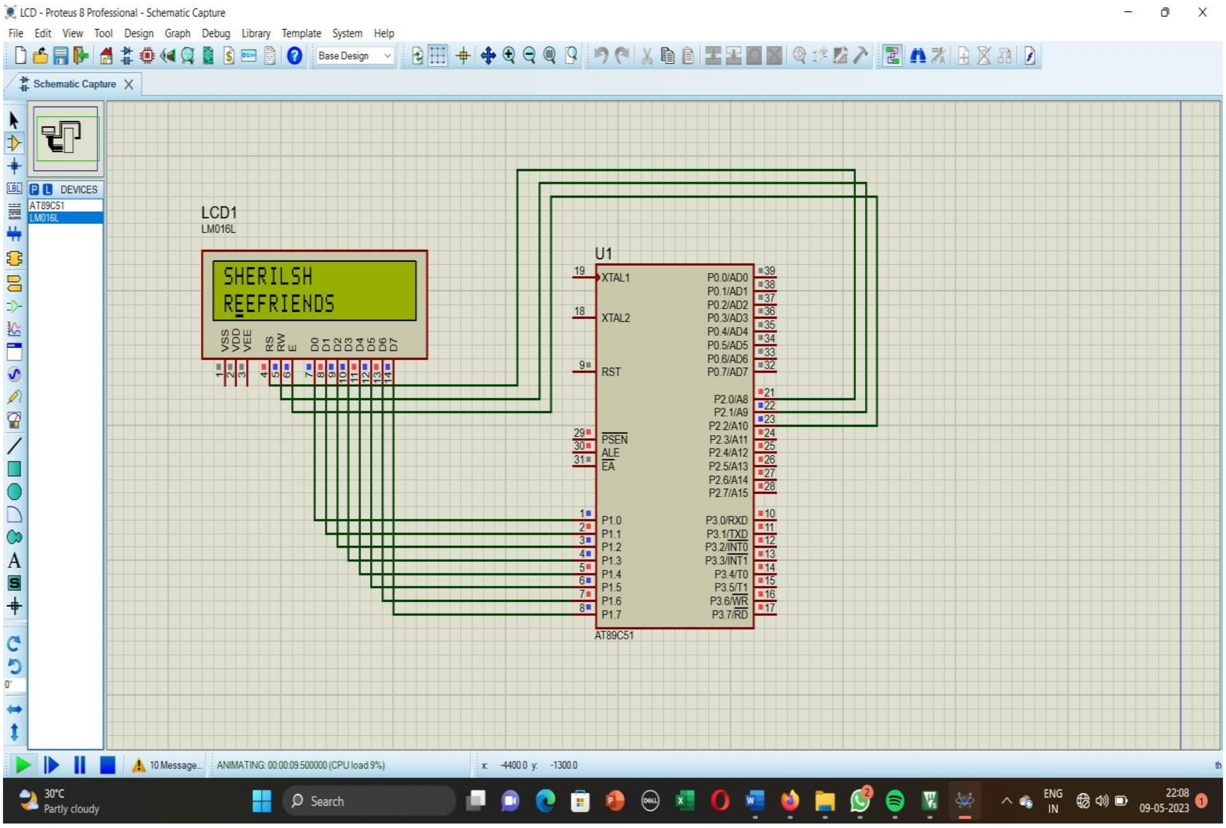
Experiment 6 LCD Display using 8051

https://youtu.be/GVUjmz0fvkk

```
ORG 0000H
    RS BIT P2.0
         RW BIT P2.1
         EN BIT P2.2
         MOV A,#38H
         ACALL CMD
         MOV A,#0EH
         ACALL CMD
         MOV A,#80H
         ACALL CMD
         MOV A,#06H
         ACALL CMD
         MOV A,#'S'
         ACALL DATA1
         MOV A,#'U'
         ACALL DATA1
         MOV A,#'B'
         ACALL DATA1
         MOV A,#'S'
         ACALL DATA1
         MOV A,#'C'
         ACALL DATA1
         MOV A,#'R'
         ACALL DATA1
         MOV A,#'I'
         ACALL DATA1
         MOV A,#'B'
         ACALL DATA1
         MOV A,#0C0H
         ACALL CMD
         MOV A,#'P'
         ACALL DATA1
```

```
        MOV A,#'R'
        ACALL DATA1
        MOV A,#'O'
        ACALL DATA1
        MOV A,#'J'
        ACALL DATA1
        MOV A,#'E'
        ACALL DATA1
        MOV A,#'X'
        ACALL DATA1
        MOV A,#'O'
        ACALL DATA1
        MOV A,#'N'
        ACALL DATA1
        MOV A,#'I'
        ACALL DATA1
        MOV A,#'C'
        ACALL DATA1
CMD:ACALL READY
    MOV P1,A
        CLR RS
        CLR RW
        SETB EN
        ACALL DELAY
        CLR EN
        RET
READY:SETB P1.7
        CLR RS
        SETB RW
        H:CLR EN
        ACALL DELAY
        SETB EN
        JB P1.7,H
        RET
```

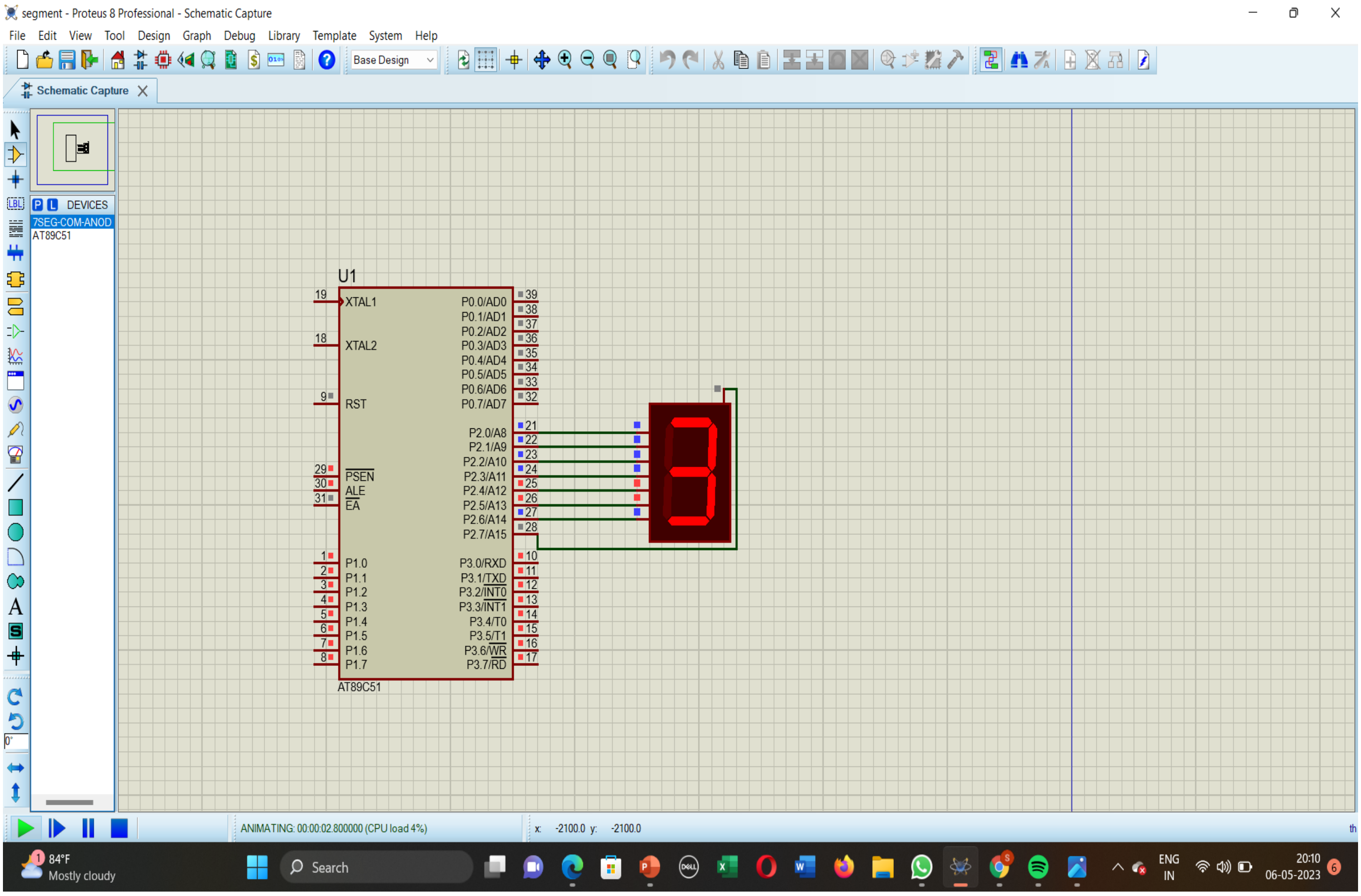
```
DATA1:ACALL READY
    MOV P1,A
        SETB  RS
        CLR  RW
        SETB EN
        ACALL DELAY
        CLR EN
DELAY:MOV R4,#180
HERE1:MOV R3,#255
HERE2:DJNZ R3,HERE2
    DJNZ R4,HERE1
        RET
        END
```



Experiment 7 [7 Segment using 8051]

https://youtu.be/faARX25GquI

```
ORG 0000H
UP: MOV P2, #0C0H
ACALL DELAY
MOV P2, #0F9H
ACALL DELAY
MOV P2, #0A4H
ACALL DELAY
MOV P2, #0B0H
ACALL DELAY
MOV P2, #99H
ACALL DELAY
MOV P2, #92H
ACALL DELAY
MOV P2, #82H
ACALL DELAY
MOV P2, #0F8H
ACALL DELAY
MOV P2, #80H
ACALL DELAY
MOV P2, #90H
ACALL DELAY

DELAY:MOV R5, #10
H1:MOV R4,#180
H2:MOV R3, #255
H3:DJNZ R3,H3
DJNZ R4,H2
DJNZ R5,H1
RET
END
```

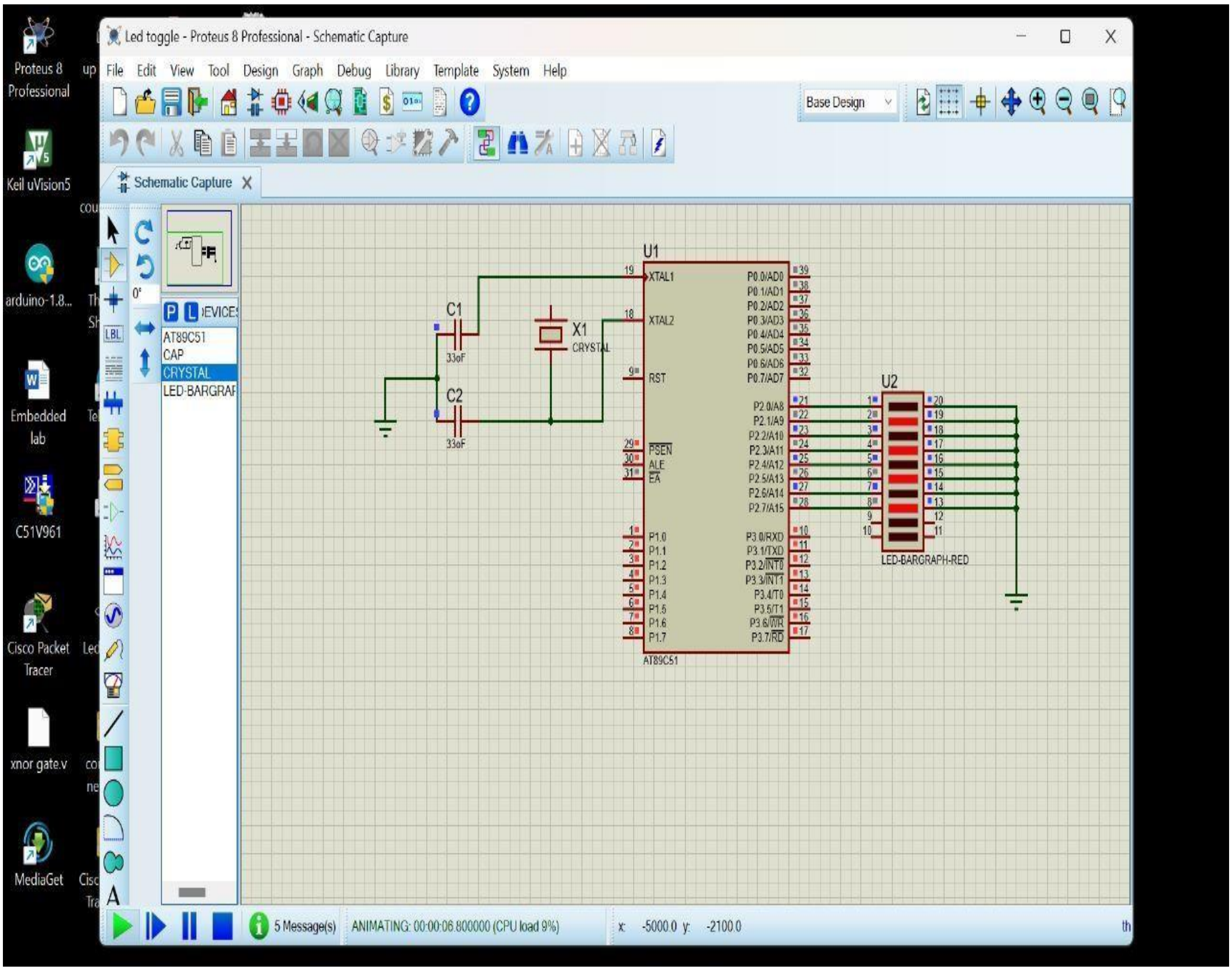


Experiment 8 Led Toggle using 8051

https://youtu.be/e6g0aHklioA

```
ORG 0000H
UP:  MOV  P2,#55H
   ACALL  DELAY
   MOV  P2,#0AAH
   ACALL  DELAY
   SJMP UP


   DELAY:MOV R4,#18
   H1:MOV R3,#255
   H2:DJNZ R3,H2
```

```
        DJNZ R4,H1
        RET
        END
```

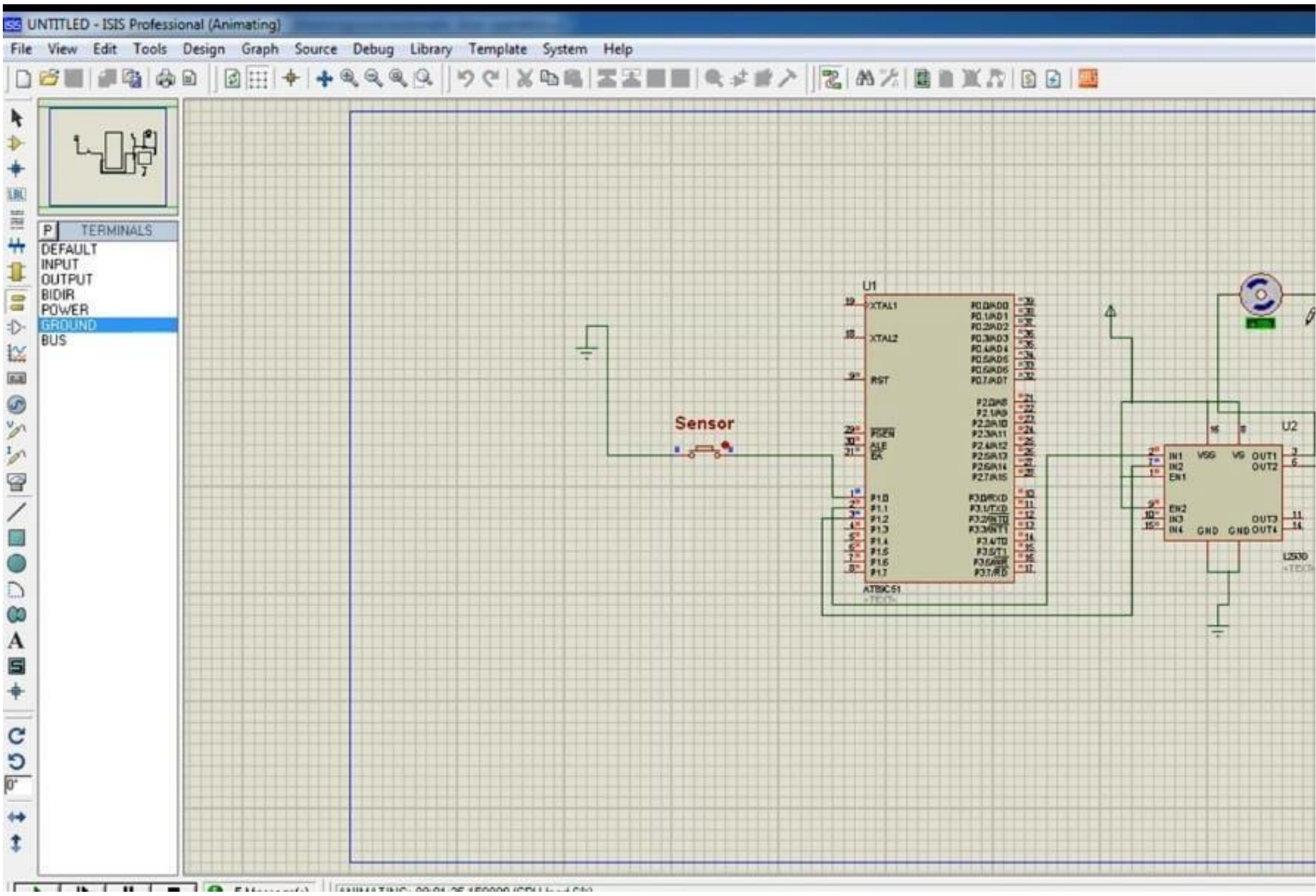


Experiment 9 Automatic door lock 8051

https://youtu.be/CEwizW2SAmE

```
              #include<reg51.h>
sbit snsor=P1^0;
sbit relay1=P1^1;
sbit relay2=p1^2;
void  delay();
void main ()
{
          relay=0;
```

```
        relay2=0;
        while(1)
        {
        if (sensor==0)// protype button , one end is connected with i/o other end with ground
        {
                relay1=1;
                relay2=0; //motor runs clockwise I-door will open
                delay();
                relay1=0;
                realy2=1;//motor starts anticlockwise- door will close
                delay();
                relay1=0;relay2=0;
}
}
}
void delay ();
{
unsigned int i,j;
        for(i=0;i<10;i++)
        {
                for(j=0;j<60000);j++);

        }
}
```

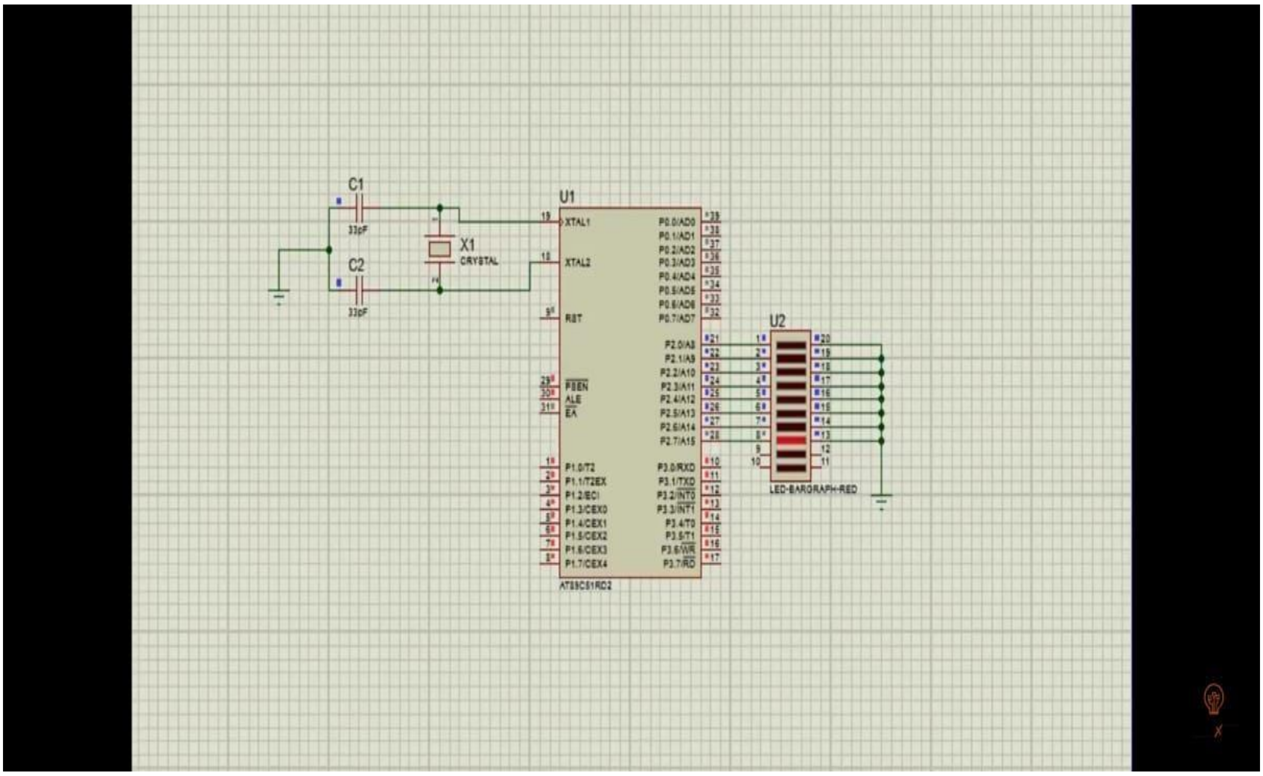

UNTITLED - ISIS Professional (Animating)
File View Edit Tools Design Graph Source Debug Library Template System Help
P TERMINALS
DEFAULT
INPUT
OUTPUT
BIDIR
POWER
GROUND
BUS
Sensor
U1
XTAL1
XTAL2
RST
AT89C51
U2
L293D

Experiment 10 Led Chaser using 8051

https://youtu.be/ZT4OA8odWHc

```
ORG 0000H
UP:MOV P2, #01H
ACALL DELAY
MOV P2, #02H
ACALL DELAY
MOV P2, #04H
ACALL DELAY
MOV P2, #08H
ACALL DELAY
MOV P2, #10H
ACALL DELAY
MOV P2, #20H
ACALL DELAY
```

```
MOV P2, #40H
ACALL DELAY
MOV P2, #80H
ACALL DELAY
SJMP UP

DELAY: MOV R4, #255
H1:DJNZ R4,H1
RET
END
```



Experiment 11 **DC MOTOR INTERFACING USING 8051 USING KEIL AND PROTEUS**

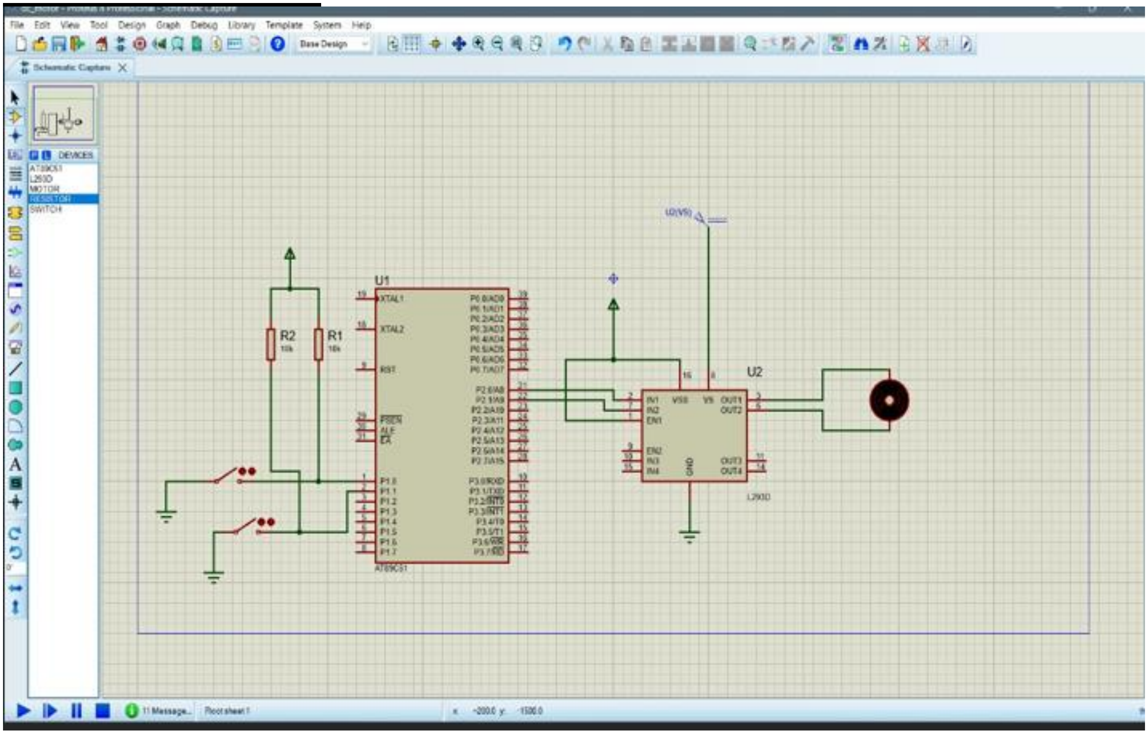
YOUTUBE LINK:

https://youtu.be/MZcjvQlX60k

PROGRAM:

```
#include<reg51.h>
sbit sw1 = P1^0;
sbit sw2 = P2^1;
sbit in1 = P2^0;
sbit in2 = P2^1;
void main(void)
{
 sw1 = sw2 = 1;
 in1 = in2 = 0;
 while(1)
 {
  if(sw1==0 & sw2 ==1)
  {
   in1 = 1;
   in2 = 0;
  }
  else if(sw1==1 & sw2==0)
  {
   in1 = 0;
   in2 = 1;
  }
  else
  {
   in1=0;
   in2=0;
    }
  }
}
```

CIRCUIT DIAGRAM:



Experiment 12 **LCD INTERFACING USING 8051 USING KEIL AND PROTEUS**

YOUTUBE LINK:

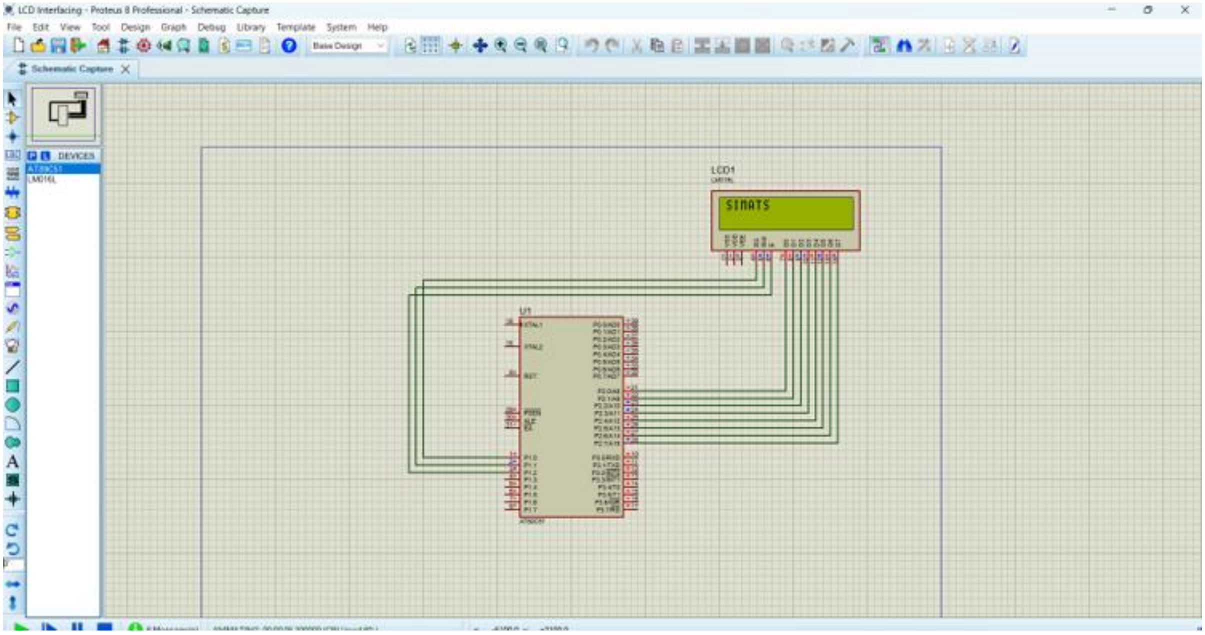
https://youtu.be/KKoXtwQDa8k

PROGRAM:

```
#include<reg51.h>
sbit rs=P1^0;
sbit rw=P1^1;
sbit en=P1^2;
void lcdcmd(unsigned char);
void lcddat(unsigned char);
void delay();
void main()
```

```
{
   P2=0*00;//output declaration,data lines d0-d7 connected
   while(1)
   {
   lcdcmd(0x38);//5*7 matrix crystal
   delay();
   lcdcmd(0x01);//clear screen
   delay();
   lcdcmd(0x10);//clear screen
   delay();
   lcdcmd(0x0c);//display on
   delay();
   lcddat('S');
   delay();
   lcddat('S');
   delay();
   lcddat('I');
   delay();
   lcddat('M');
   delay();
  lcddat('A');
   delay();
   lcddat('I');
   delay();
   lcddat('S');
   delay();

   }
}
void lcdcmd(unsigned char val)
{
P2=val;
rs=0;
rw=0;
en=1;
delay();
en=0;
}
void lcddat(unsigned char val)
{
P2=val;
rs=1;
rw=0;
en=1;
delay();
en=0;
}
void delay()
{
unsigned int i;
for(i=0;i<12000;i++);
}
```

CIRCUIT DIAGRAM:


LCD Interfacing - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Schematic Capture
DEVICES
AT89C51
LM016L
LCD1
LM016L
SIMATS
U1
AT89C51

Experiment 13 **KEYPAD INTERFACING USING 8051 USING KEIL AND PROTEUS**

YOUTUBE LINK:
https://youtu.be/nYBpdYezgVs

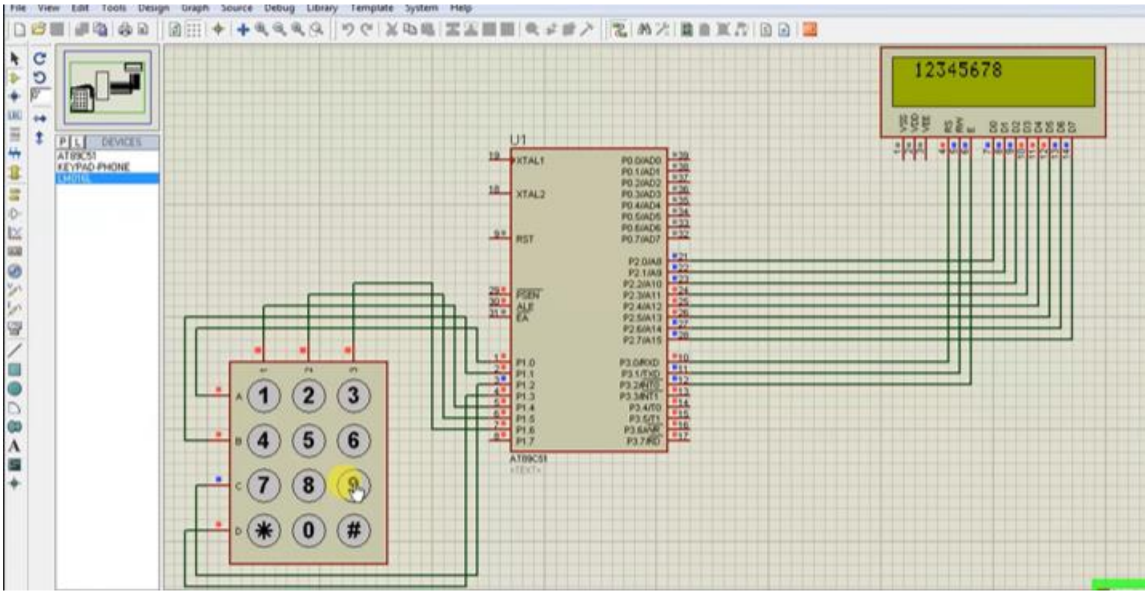
PROGRAM:

```
#include<reg51.h>
#define display_port P2      //Data pins connected to port 2 on microcontroller
sbit rs = P3^2;  //RS pin connected to pin 2 of port 3
sbit rw = P3^3;  // RW pin connected to pin 3 of port 3
sbit e =  P3^4;  //E pin connected to pin 4 of port 3
sbit C4 = P1^0;     // Connecting keypad to Port 1
sbit C3 = P1^1;
sbit C2 = P1^2;
sbit C1 = P1^3;
sbit R4 = P1^4;
sbit R3 = P1^5;
sbit R2 = P1^6;
sbit R1 = P1^7;
void msdelay(unsigned int time)  // Function for creating delay in milliseconds.
{
   unsigned i,j ;
   for(i=0;i<time;i++)
   for(j=0;j<1275;j++);
}
void lcd_cmd(unsigned char command)  //Function to send command instruction to LCD
{
   display_port = command;
   rs= 0;
   rw=0;
   e=1;
   msdelay(1);
 e=0;
```

```
}
void lcd_data(unsigned char disp_data)  //Function to send display data to LCD
{
   display_port = disp_data;
   rs= 1;
   rw=0;
   e=1;
   msdelay(1);
   e=0;
}
 void lcd_init()    //Function to prepare the LCD  and get it ready
{
   lcd_cmd(0x38);  // for using 2 lines and 5X7 matrix of LCD
   msdelay(10);
   lcd_cmd(0x0F);  // turn display ON, cursor blinking
   msdelay(10);
   lcd_cmd(0x01);  //clear screen
   msdelay(10);
   lcd_cmd(0x81);  // bring cursor to position 1 of line 1
   msdelay(10);
}
void row_finder1() //Function for finding the row for column 1
{
R1=R2=R3=R4=1;
C1=C2=C3=C4=0;
if(R1==0)
lcd_data('1');
if(R2==0)
lcd_data('4');
if(R3==0)
lcd_data('7');
if(R4==0)
lcd_data('*');
}
void row_finder2() //Function for finding the row for column 2
{
R1=R2=R3=R4=1;
C1=C2=C3=C4=0;
if(R1==0)
lcd_data('2');
if(R2==0)
lcd_data('5');
if(R3==0)
lcd_data('8');
if(R4==0)
lcd_data('0');
}
void row_finder3() //Function for finding the row for column 3
{
R1=R2=R3=R4=1;
C1=C2=C3=C4=0;
if(R1==0)
lcd_data('3');
if(R2==0)
lcd_data('6');
if(R3==0)
lcd_data('9');
if(R4==0)
lcd_data('#');
```

```
}
void row_finder4() //Function for finding the row for column 4
{
R1=R2=R3=R4=1;
C1=C2=C3=C4=0;
if(R1==0)
lcd_data('A');
if(R2==0)
lcd_data('B');
if(R3==0)
lcd_data('C');
if(R4==0)
lcd_data('D');
}
void main()
{
    lcd_init();
    while(1)
    {
        msdelay(30);
        C1=C2=C3=C4=1;
          R1=R2=R3=R4=0;
          if(C1==0)
          row_finder1();
          else if(C2==0)
           row_finder2();
          else if(C3==0)
        row_finder3();
        else if(C4==0)
        row_finder4();
    }
```

CIRCUIT DIAGRAM:



Experiment 14 **DIGITAL CLOCK INTERFACING USING 8051 USING KEIL AND PROTEUS**

YOUTUBE LINK:

https://youtu.be/YJXdJDQdiq8

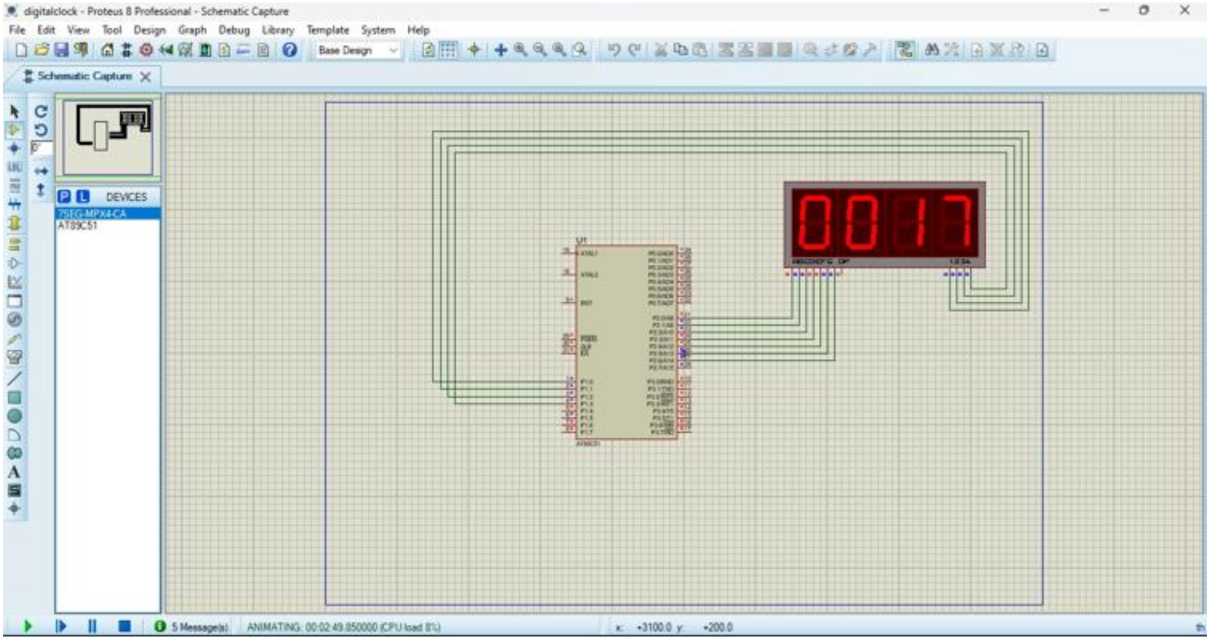
PROGRAM:

```
include <reg51.h>
#define msec 1
unsigned int arr[10]={0x40,0xF9,0x24,0x30,0x19,0x12,0x02,0xF8,0x00,0x10};
sbit d4=P1^0;
sbit d3=P1^1;
sbit d2=P1^2;
sbit d1=P1^3;
sbit d0=P1^4;
sbit d= P1^5;
unsigned int v1,v2,v3,v4,v0,v5,v6;
void delay(unsigned int count)
{
unsigned int j,k;
for (j=0;j<=count;j++)
for (k=0;k<=5;k++);
}
void main()
{
 v1=v2=v3=v4=v0=v5=v6=0;
 while(1)
{
{
    v0=v0+1;
            if(v0==130)
{
            v0=0;
            v1=v1+1;
            }
    P2=0xFF;
            d = 1;
            d3 = d2 = d4 = d0 = d1= 0;
            P2 = arr[v1];
            delay(msec);

    if(v1==10)
            {
            v1=0;
            v2=v2+1;
            }
            P2=0xFF;
            d0 = 1;
            d4 = d3 = d1 =d=d2= 0;
            P2 = arr[v2];
            delay(msec);

    if(v2==6)
            {
            v2=0;
            v3=v3+1;
            }
            P2=0xFF;
```

```
        d1 = 1;
        d2 = d4 = d3 =d=d0= 0;
        P2 = arr[v3];
        delay(msec);

if(v3==10)
        {
        v3=0;
        v4=v4+1;
        }
        P2=0xFF;
        d2 = 1;
        d3 = d4 = d1 =d=d0= 0;
        P2 = arr[v4];
        delay(msec);

        if(v4==6)
        {
        v4=0;
        v5=v5+1;
        }
        P2=0xFF;
        d3 = 1;
        d0 = d2 = d1 =d=d4= 0;
        P2 = arr[v5];
        delay(msec);

        if(v5==10)
        {
        v5=0;
        v6=v6+1;
        }
        P2=0xFF;
        d4 = 1;
        d3 = d2 = d1 =d=d0= 0;

if(v3==10)
        {
        v3=0;
        v4=v4+1;
        }
        P2=0xFF;
        d2 = 1;
        d3 = d4 = d1 =d=d0= 0;
        P2 = arr[v4];
        delay(msec);

        if(v4==6)
        {
        v4=0;
        v5=v5+1;
        }
        P2=0xFF;
        d3 = 1;
        d0 = d2 = d1 =d=d4= 0;
        P2 = arr[v5];
        delay(msec);

        if(v5==10)
```

```
            {
            v5=0;
            v6=v6+1;
            }
            P2=0xFF;
            d4 = 1;
            d3 = d2 = d1 =d=d0= 0;
            P2 = arr[v6];
            delay(msec);
            if(v6==1&&v5==2)
            {
            v1=0;
            v2=0;
            v3=0;
            v4=0;
   v5=0;
            v6=0;
   }
delay(msec);
P2=0xFF;
}
}
}
```

CIRCUIT DIAGRAM:


digitalclock - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
DEVICES
7SEG-MPX4-CA
AT89C51
U1
0017
5 Message(s)
ANIMATING: 00:02:49.850000 (CPU load 8%)
x: +3100.0 y: +200.0

Experiment 15 **AUTOMATIC DOOR LOCKING SYSTEM USING 8051 USING KEIL AND PROTEUS**

YOUTUBE LINK:

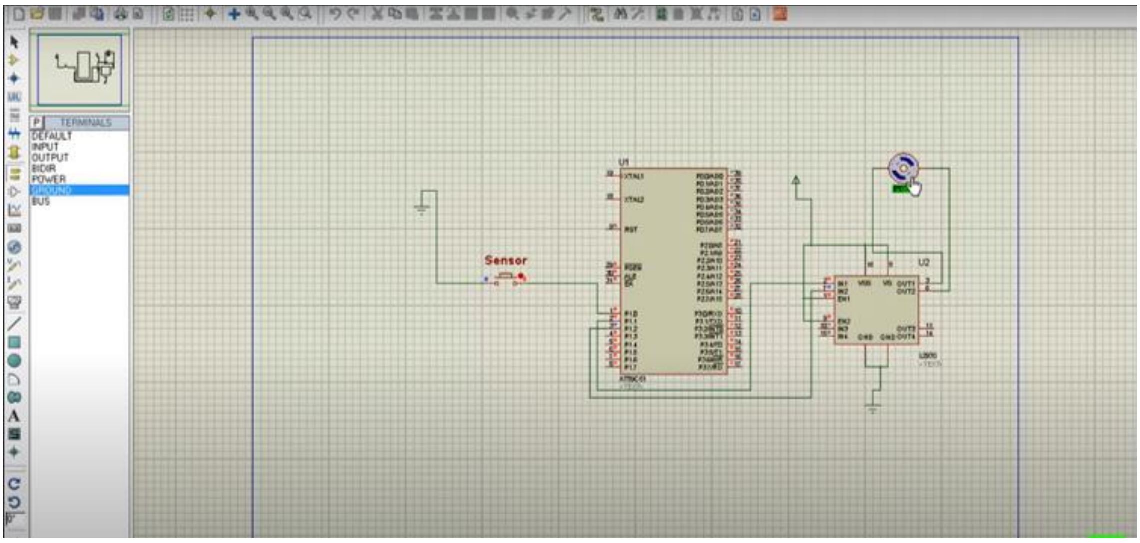
https://youtu.be/y4k4vlxtGcs

PROGRAM:

```
#include<reg52.h>
sbit r0=P2^0;    sbit r1=P2^1;    sbit r2=P2^2;    sbit r3=P2^3;    sbit c0=P2^5;    sbit c1=P2^6;
sbit c2=P2^7;    sbit en=P3^6;    sbit rs=P3^5;    sbit rw=P3^7;    sbit lock=P3^0;
char t1[]="Enter PIN:";
char t2[]="Access Granted";
char t3[]="Access Denied";
char pin[]="1234";
char pinEntered[4];
unsigned int m = 0;
unsigned int flag = 0;
void delay(unsigned int no)
{
unsigned int i,j;
for(j=0;j<=no;j++)    for(i=0;i<=10;i++);
}
void lcdcmd(unsigned int  command){
P1=command;    rw=0;    rs=0;    en=0;    delay(1000);    en=1;    delay(1000);    en=0;
}
void lcddata(char data1)
{
P1=data1;    rw=0;    rs=1;    en=0;    delay(1000);    en=1;    delay(1000);    en=0;
}
void lcdint(){
lcdcmd(0x30);    delay(1000);    lcdcmd(0x30);    delay(1000);    lcdcmd(0x30);    delay(1000);
lcdcmd(0x30);    delay(1000);    lcdcmd(0x30);    delay(1000);    lcdcmd(0x38);    delay(1000);
lcdcmd(0x01);    delay(1000);    lcdcmd(0x0F);    delay(1000);    lcdcmd(0x80);    delay(1000);
}
char keypad()
{
char c='a';
while(c!='s'){
r0=0;r1=1;r2=1;r3=1;
if(c0==0){lcddata('1');P0=0xF0;delay(10000);c='s';return '1';}
 if(c1==0){lcddata('2');P0=0xF0;delay(10000);c='s';return '2';}
 if(c2==0){lcddata('3');P0=0xF0;delay(10000);c='s';return '3';}
r0=1;r1=0;r2=1;r3=1;
if(c0==0){lcddata('4');P0=0xF0;delay(10000);c='s';return '4';}
 if(c1==0){lcddata('5');P0=0xF0;delay(10000);c='s';return '5';}
 if(c2==0){lcddata('6');P0=0xF0;delay(10000);c='s';return '6';}
r0=1;r1=1;r2=0;r3=1;
if(c0==0){lcddata('7');P0=0xF0;delay(10000);c='s';return '7';}
 if(c1==0){lcddata('8');P0=0xF0;delay(10000);c='s';return '8';}
 if(c2==0){lcddata('9');P0=0xF0;delay(10000);c='s';return '9';}
r0=1;r1=1;r2=1;r3=0;
//if(c0==0){lcddata('*');P0=0xF0;delay(10000);c='s';return '1';}
 if(c1==0){lcddata('0');P0=0xF0;delay(10000);c='s';return '0';}
// if(c2==0){lcddata('#');P0=0xF0;delay(10000);c='s';return '1';}
}
}
void main()
{
   unsigned int i=0;
   P1=0x00;    P2=0xF0;    P3=0x00;
   lcdint();
   while(1){
   i=0;
   while(t1[i]!='\0')
   {
      lcddata(t1[i]);
      i++;
   }
   lock=0;
   lcdcmd(0xC0);
```

```
  for(i=0;i<=3;i++)
   pinEntered[i] =keypad();
  i=0;
  lcdcmd(0x01);

  if(pinEntered[0]==pin[0])
  {       i++;
     if(pinEntered[1]==pin[1])
     {
        if(pinEntered[2]==pin[2])
        {
           if(pinEntered[3]==pin[3])
            {
               lock=1;     i=0;
                     while(t2[i]!='\0')
                       {
                          lcddata(t2[i]);
                          i++;
                       }i=0;
            }
        }
     }
  }else{    i=0;
      while(t3[i]!='\0')
     {
        lcddata(t3[i]);
        i++;
     } i=0; }

  if(i!=0)
  {                                  //Download Full Code For Better Understanding
i=0;
      while(t3[i]!='\0')
     {
        lcddata(t3[i]);
        i++;
     }
  }
  delay(1000000);
  lcdcmd(0x01);
  lcdcmd(0x80);
i=0;
```

CIRCUIT DIAGRAM:



Experiment 16

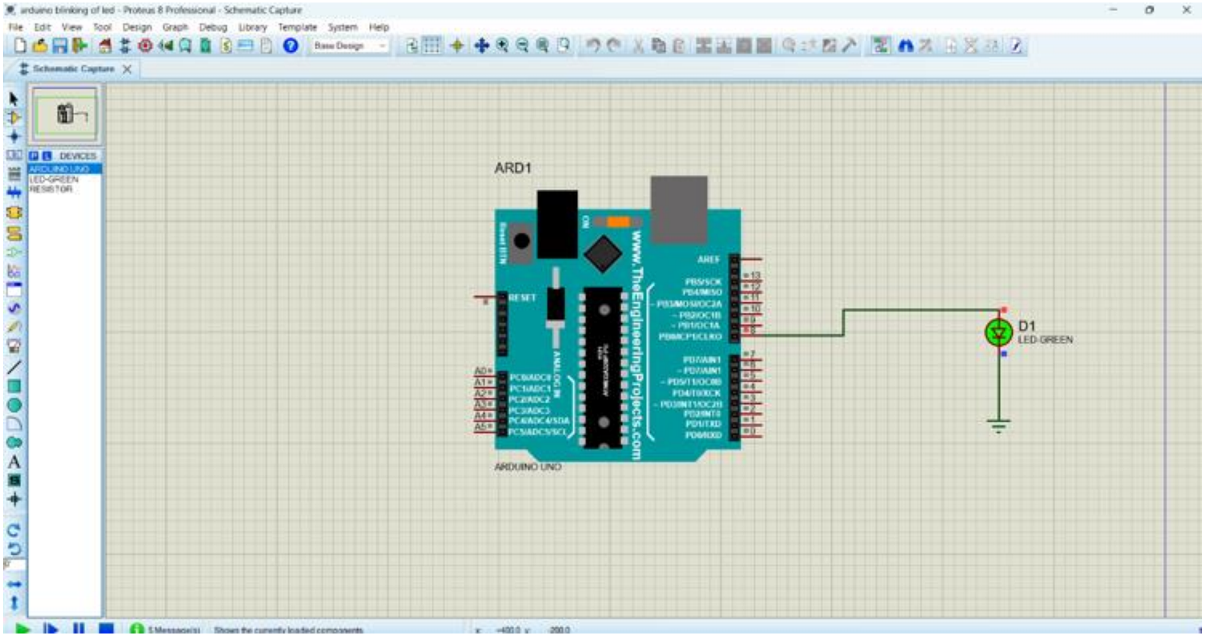
BLINKING OF LED USING ARDUINO IDE AND PROTEUS

YOUTUBE LINK:
https://youtu.be/mz1TsvJzB1c

PROGRAM:

```
int led=8;
void setup()
{
 pinMode(led, OUTPUT);
}
void loop()
{
digitalWrite(led, HIGH);
delay(1000);//Milliseconds
digitalWrite(led, LOW);
delay(1000);
}
```

CIRCUIT DIAGRAM:


arduino blinking of led - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
DEVICES
ARDUINO UNO
LED-GREEN
RESISTOR
ARD1
ARDUINO UNO
www.TheEngineeringProjects.com
RESET
AREF
D1
LED-GREEN

Experiment 17

FADE IN FADE OUT OF LED USING ARDUINO IDE AND PROTEUS

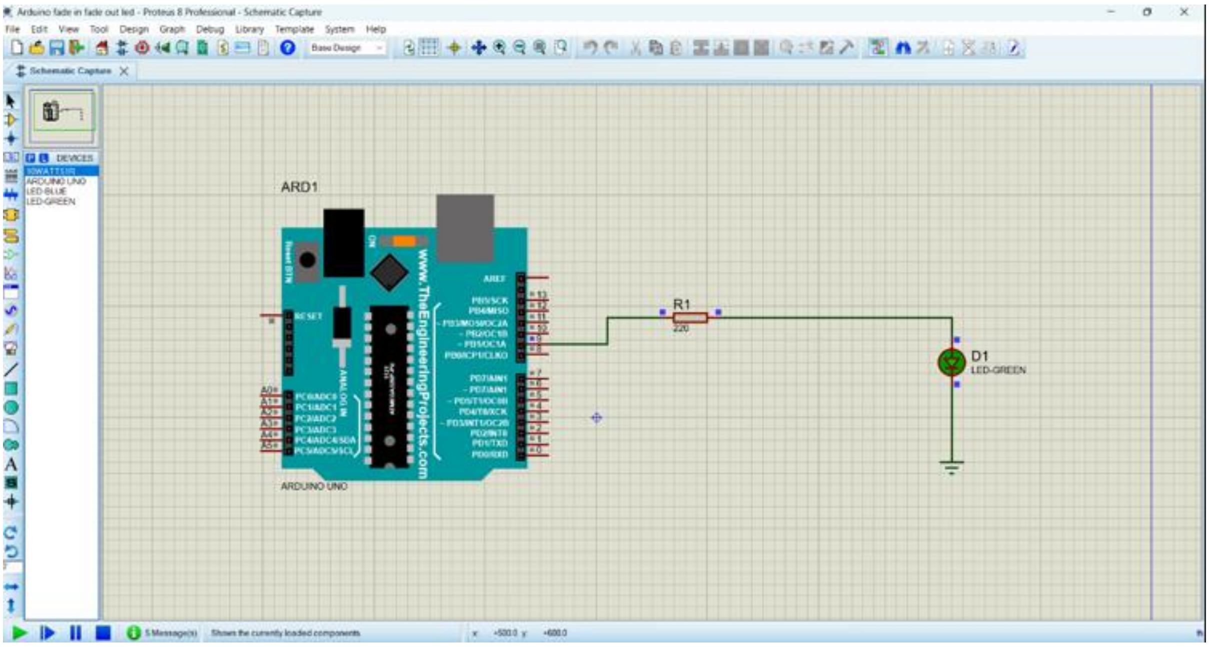
YOUTUBE LINK:
[**https://youtu.be/nPObk2D0uZ4**](https://youtu.be/nPObk2D0uZ4)

PROGRAM:

```
int led = 9;
int brightness = 0;
int fadeAmount = 5;
```

```
void setup() {
  pinMode(led, OUTPUT);
}
void loop(){
  analogWrite(led, brightness);
  brightness = brightness + fadeAmount;
  if(brightness <=0 || brightness >= 255 ) {
    fadeAmount = -fadeAmount;
  }
  delay(30);
}
```

CIRCUIT DIAGRAM:



Experiment 18

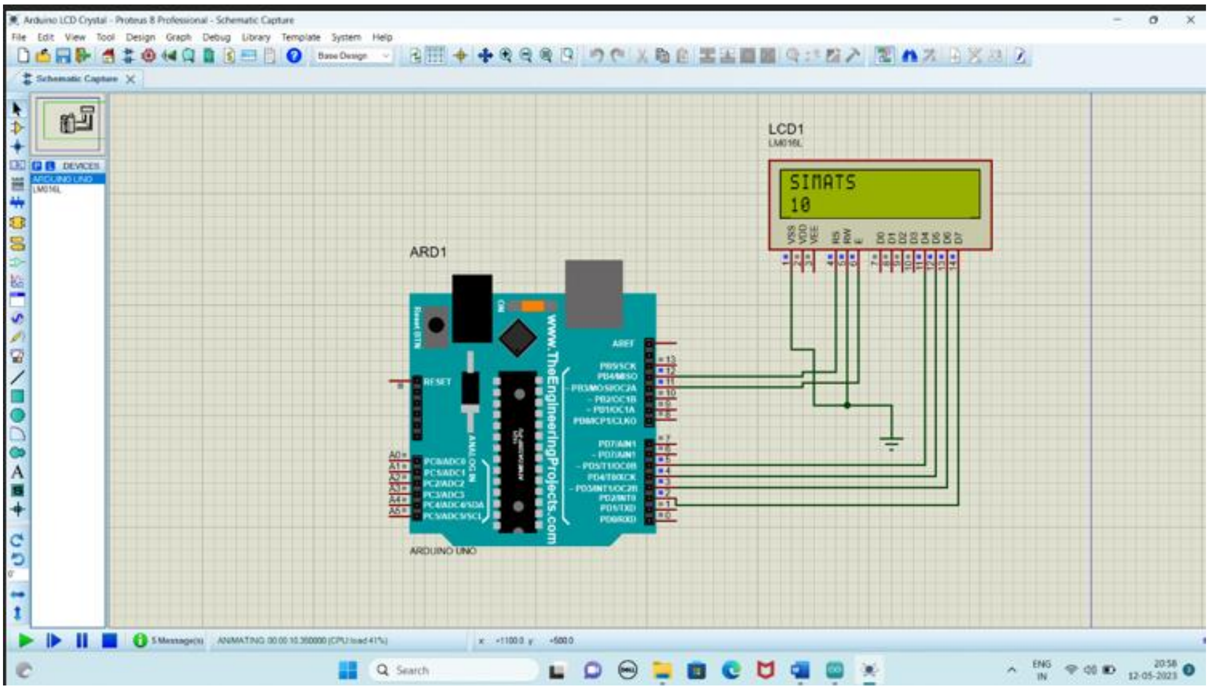
INTERFACING OF LCD USING ARDUINO IDE AND PROTEUS

YOUTUBE LINK:
https://youtu.be/P-oiNbGoLOI

PROGRAM:

```
#include<LiquidCrystal.h>
const int rs=12,en=11,d4=5,d5=4,d6=3,d7=2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {
  lcd.begin(16,2);
  lcd.print("SIMATS");
  }
void loop(){
  lcd.setCursor(0,1);
  lcd.print(millis()/1000);
}
```

CIRCUIT DIAGRAM:


Arduino LCD Crystal - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Schematic Capture
DEVICES
ARDUINO UNO
LM016L
LCD1
LM016L
SIMATS
10
ARD1
www.TheEngineeringProjects.com
RESET
AREF
ARDUINO UNO
ANIMATING

Experiment 19

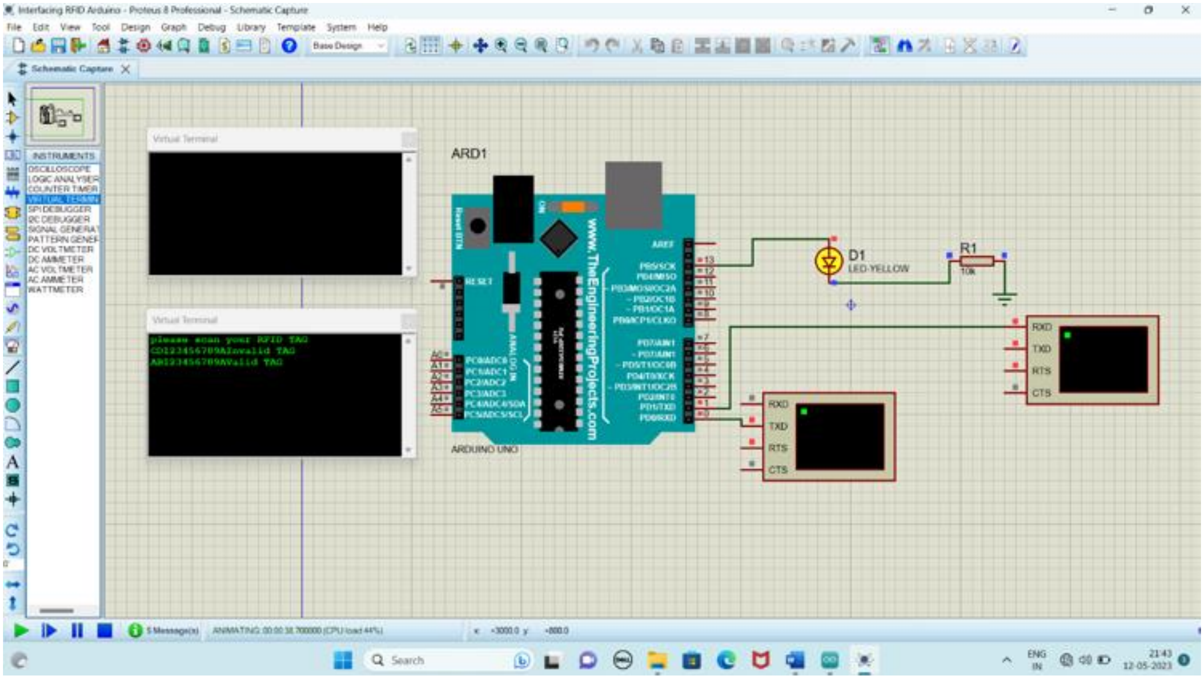
INTERFACING RFID MODULE USING ARDUINO IDE AND PROTEUS

YOUTUBE LINK:
https://youtu.be/kWaYomvnpUU

PROGRAM:

```
void setup(){
 Serial.begin(9600);
  pinMode(13,OUTPUT);
  Serial.println("please scan your RFID TAG");
}
void loop(){
  while(Serial.available()>0)
  {
    c=Serial.read();
    count++;
    id +=c;
    if(count == 12)
    {
      Serial.print(id);
      //break;

      if(id=="AB123456789A")
      {
        Serial.println("Valid TAG");
        digitalWrite(13,HIGH);
      }
      else
      {
        digitalWrite(13,LOW);
        Serial.println("Invalid TAG");
      }
    }
  }
  count = 0;
  id="";
  delay(500);
}
```

CIRCUIT DIAGRAM:



```
please scan your RFID TAG
CD123456789AInvalid TAG
AB123456789AValid TAG
```

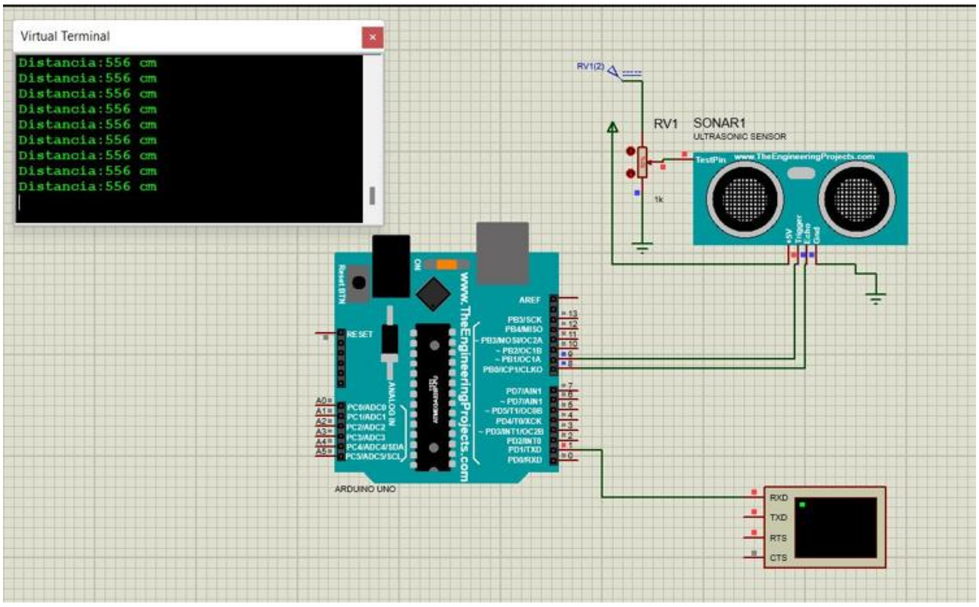
Experiment 20 **INTERFACING OF ULTRASONIC SENSOR USING ARDUINO IDE AND PROTEUS**

YOUTUBE LINK: https://youtu.be/3i1le1GbLcU

PROGRAM:

```
long duracion;
long distancia;
int echo=8;
int trig=9;
void setup()
{
  Serial.begin(9600);
  pinMode(trig,OUTPUT);
  pinMode(echo,INPUT);
}
void loop()
{
  digitalWrite(trig,LOW);
  delayMicroseconds(4);
  digitalWrite(trig,HIGH);
  delayMicroseconds(10);
  digitalWrite(trig,LOW);
  duracion=pulseIn(echo,HIGH);
  distancia=duracion/58.4;
  Serial.print("Distancia:");
  Serial.print(distancia);
  Serial.println("cm");
  delay(100);
}
```

CIRCUIT DIAGRAM:



```
Distancia:556 cm
Distancia:556 cm
Distancia:556 cm
Distancia:556 cm
Distancia:556 cm
Distancia:556 cm
Distancia:556 cm
Distancia:556 cm
Distancia:556 cm
```