

Circuit Diagram:

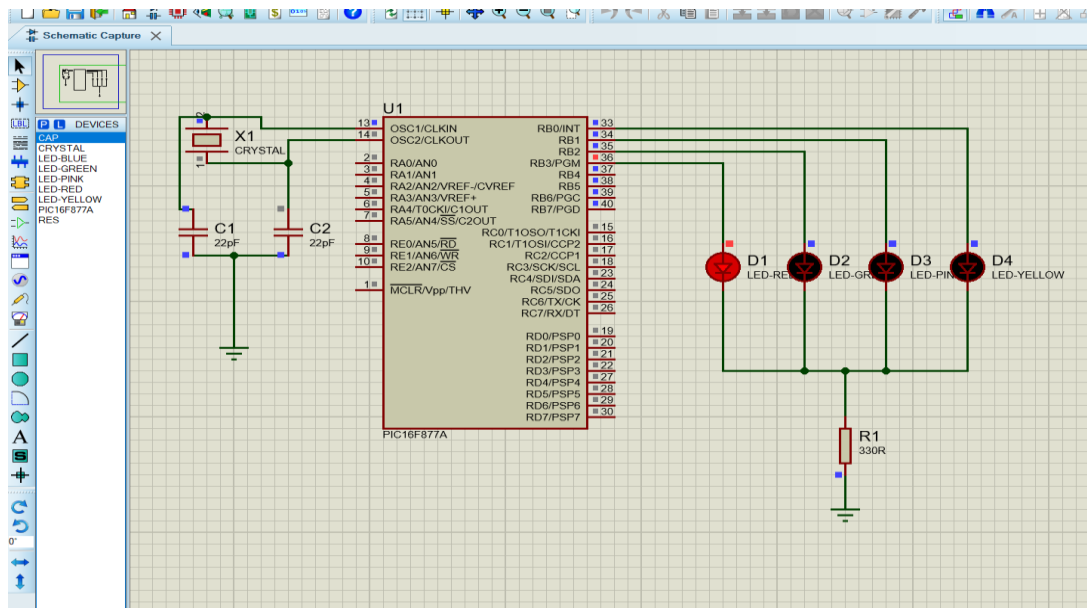


Fig: LED Blinking

Code:

```
int counter = 0;

void main() {
    TRISB = 0x00;
    PORTB = 0x00;
    while(1)
    {
        PORTB.RB3 = 1;
        Delay_ms(300);
        PORTB.RB3 = 0;
        PORTB.RB2 = 1;
        Delay_ms(300);
        PORTB.RB2 = 0;
        PORTB.RB1 = 1;
        Delay_ms(300);
        PORTB.RB1 = 0;
        PORTB.RB0 = 1;
        Delay_ms(300);
        PORTB.RB0 = 0;
        counter++;
    }
}
```

Circuit Diagram:

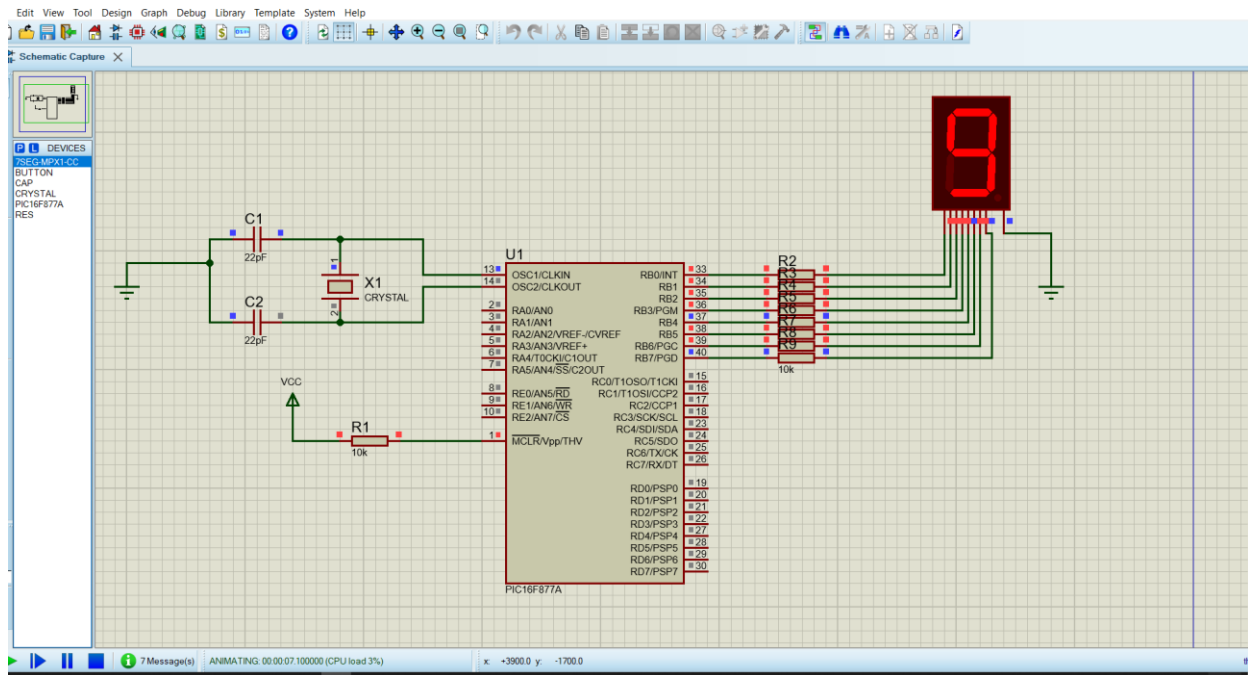


Fig: 7Segment 1Digit Display

Code:

```
char arraycc[]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
```

```
void main() {
```

```
    int i = 0;
```

```
    trisb = 0x00;
```

```
    portb = 0xff;
```

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

```
        portb = arraycc[i];
```

```
        delay_ms(100);
```

```
    }
```

```
}
```

Circuit Diagram:

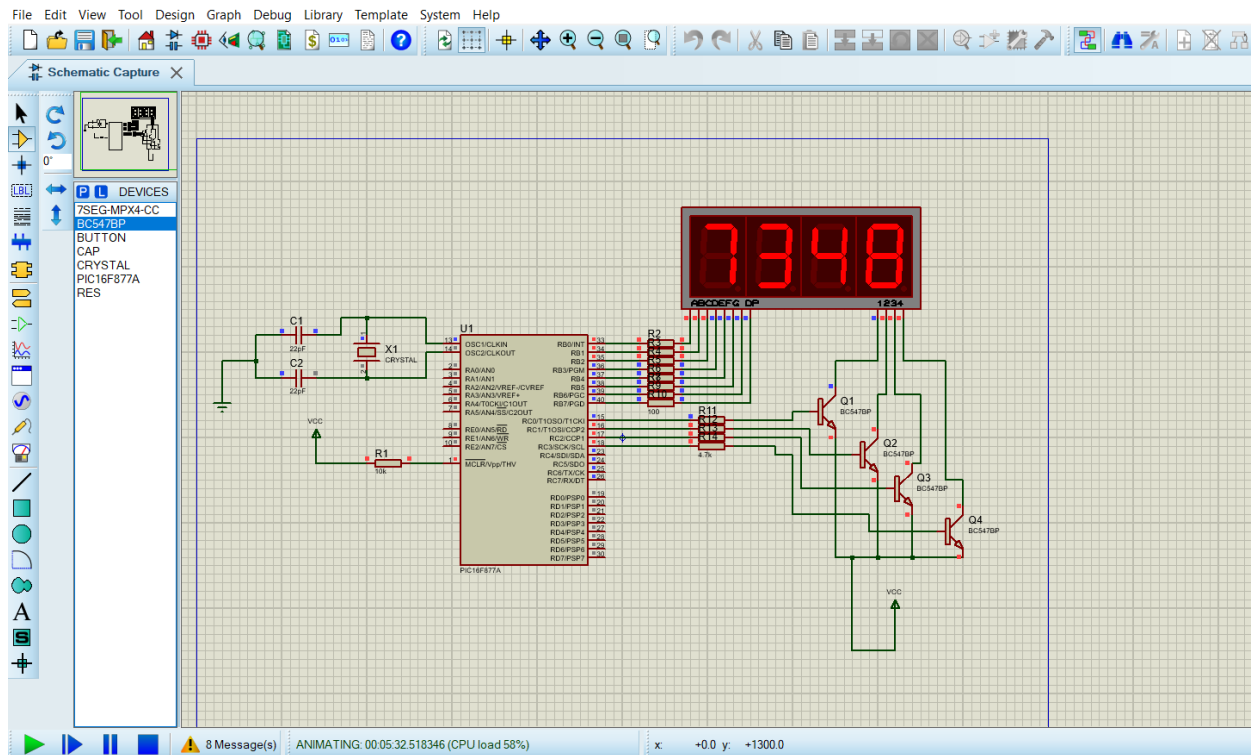


Fig: 7-Segment 4Digit Display

CODE:

```
char arraycc[]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
void main() {
    int d_zero,d_one,d_two,d_three,i,j;
    Trisb=0x00;
    Trisc=0x00;
    portb=0x00;
    portc=0x00;

    while(1)
    {
        for(i=0;i<=9999;i++) //counting 0-9999
        {
            d_zero=i/1000; // for first digit of display
            d_one=((i/100)%10); // for second digit
            d_two=((i/10)%10); // for third digit
            d_three=i%10;

            for(j=0;j<=10;j++)
            {
                portc.f0=0;
                portb=arraycc[d_zero]; //display
                delay_ms(1);
                portc.f0=1;
            }
        }
    }
}
```

```
portc.f1=0;
portb=arraycc[d_one];//display
delay_ms(1);
portc.f1=1;
```

```
portc.f2=0;
portb=arraycc[d_two];//display
delay_ms(1);
portc.f2=1;
```

```
portc.f3=0;
portb=arraycc[d_three];//display
delay_ms(1);
portc.f3=1;
```

```
}
```

```
}
```

```
}
```

```
}
```

Circuit Diagram:

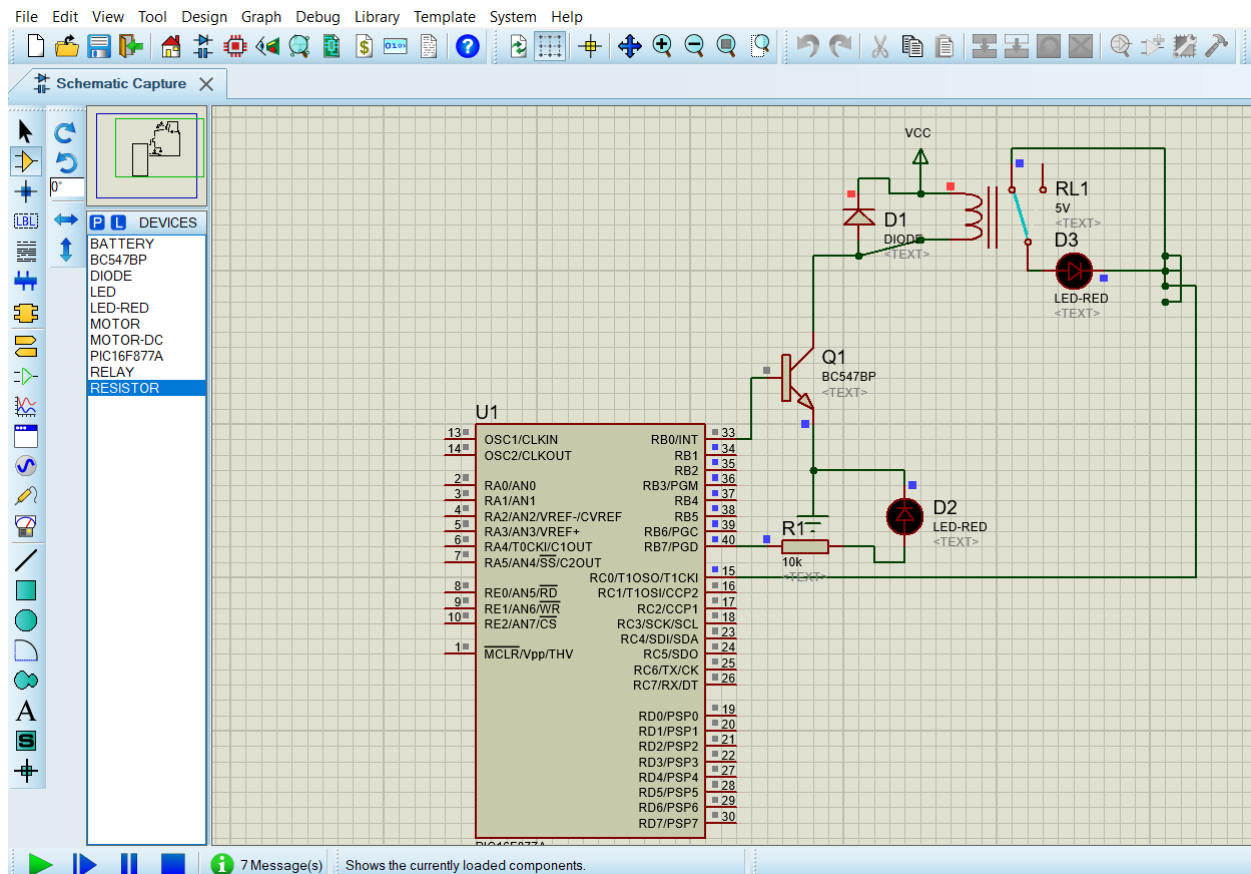


Fig: control AC current by DC current.

CODE:

```
void main() {

    trisb=0;

    portb=0;

    while(1){

        portb.f0=1;

        delay_ms(1000);

        portb.f0=0;

        delay_ms(1000);

    }

}
```

Circuit Diagram:

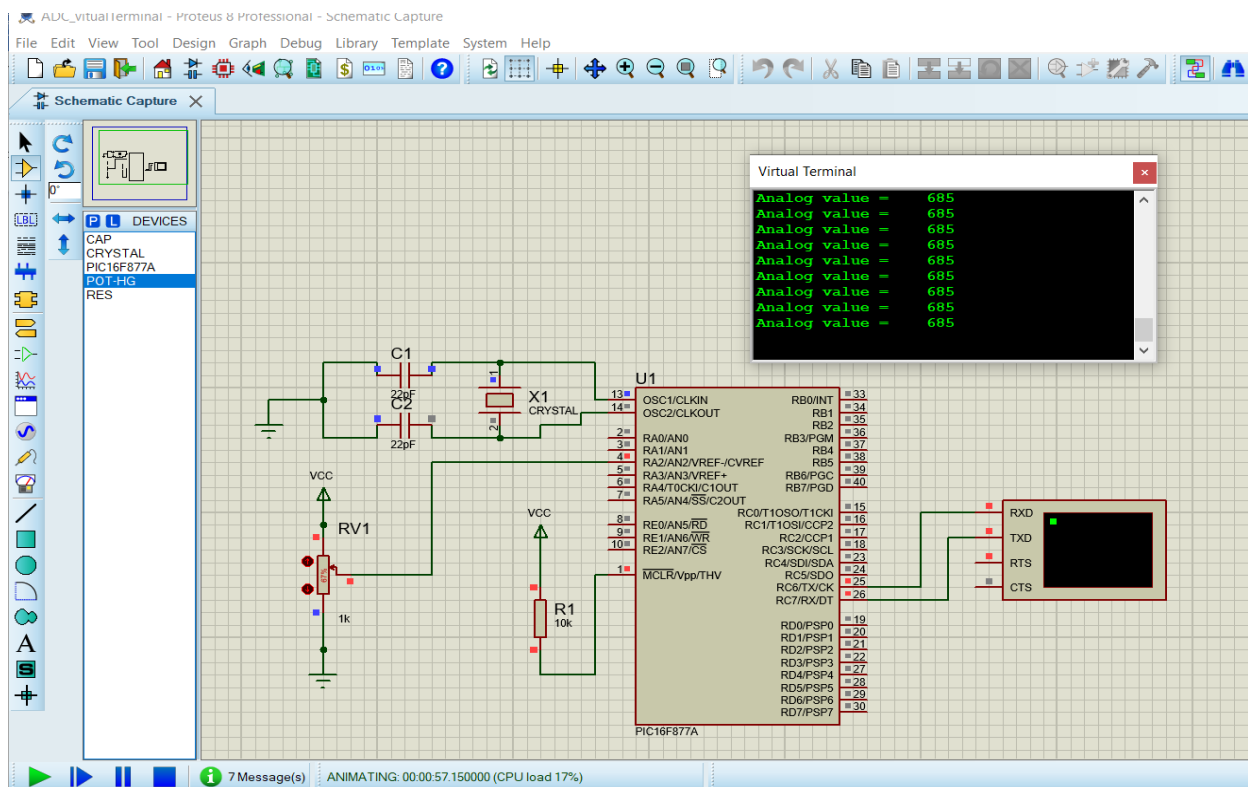


Fig: Display ADC value in the virtual terminal.

CODE:

```
int valADC;
char x[4];
void main(){
    UART1_Init(9600);//initialize ADC
    ADC_Init();
    while(1){
        valADC = ADC_Read(2);
        IntToStr(valADC,x);
        UART1_Write_Text("Analog value = ");
        UART1_Write_Text(x);
        strcpy(x,"");
        UART1_Write(13);
        Delay_ms(1000);
    }
}
```

CODE:

```

sbit LCD_RS at RB0_bit;
sbit LCD_EN at RB1_bit;
sbit LCD_D4 at RB2_bit;
sbit LCD_D5 at RB3_bit;
sbit LCD_D6 at RB4_bit;
sbit LCD_D7 at RB5_bit;

sbit LCD_RS_DIRECTION at TRISB0_bit;
sbit LCD_EN_DIRECTION at TRISB1_bit;
sbit LCD_D4_DIRECTION at TRISB2_bit;
sbit LCD_D5_DIRECTION at TRISB3_bit;
sbit LCD_D6_DIRECTION at TRISB4_bit;
sbit LCD_D7_DIRECTION at TRISB5_bit;

```

```

char display[16]="";
void main()
{
    unsigned int result;
    float volt,temp;
    trisb=0x00;
    trisa=0xff;
    adcon1=0x80;
    lcd_init();
    lcd_cmd(_lcd_clear);
    lcd_cmd(_LCD_CURSOR_OFF);
    while(1)
    {
        result = adc_read(0);
        volt=result*4.88;
        temp=volt/10;
        lcd_out(1,1,"Temp = ");
        floattostr(temp,display);
        lcd_out_cp(display);
        lcd_chr(1,16,223); //print at pos(row=1,col=13) "°" =223 =0xdf
        lcd_out_cp(" C"); //celcius
    }
}

```


Circuit Diagram:

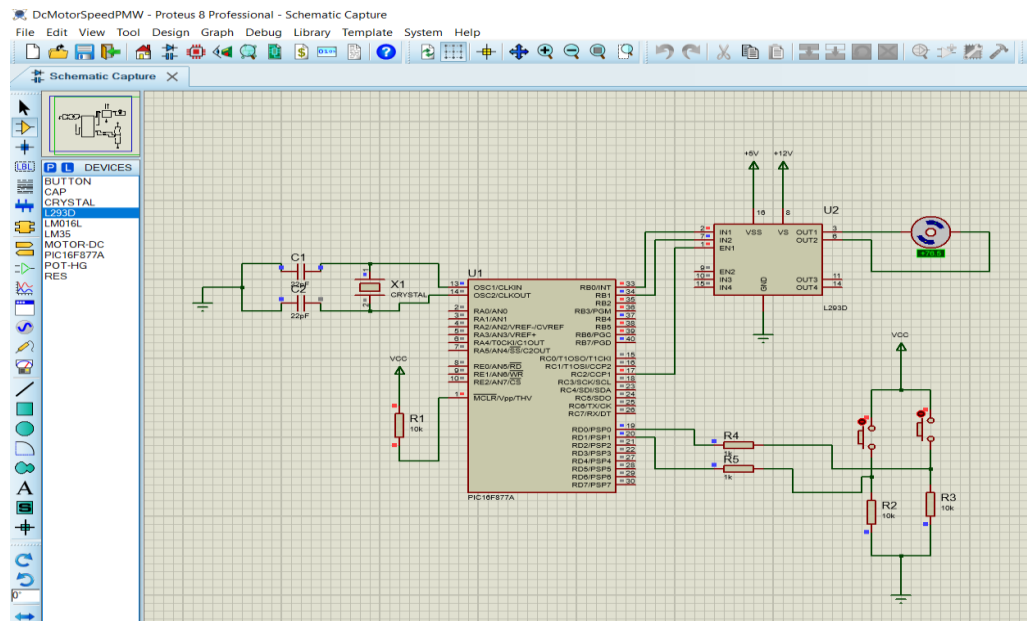


Fig: DC Motor Speed Control PWM

CODE:

```
void main() {
    short duty = 0;
    TRISD = 0xFF;
    TRISB = 0x00;

    PORTB.F0 = 0xFF;
    PORTB.F1 = 0x00;
    PWM1_Init(1000);
    PWM1_Start();
    PWM1_Set_Duty(duty);

    while (1) {
        if (RD0_bit && duty < 250) {
            Delay_ms(100);
            if (RD0_bit && duty < 250) {
                duty = duty + 10;
                PWM1_Set_Duty(duty);
            }
        }
        if (RD1_bit && duty > 0) {
            Delay_ms(100);
            if (RD1_bit && duty > 0) {
                duty = duty - 10;
                PWM1_Set_Duty(duty);
            }
        }
        Delay_ms(10);
    }
}
```

Circuit Diagram:

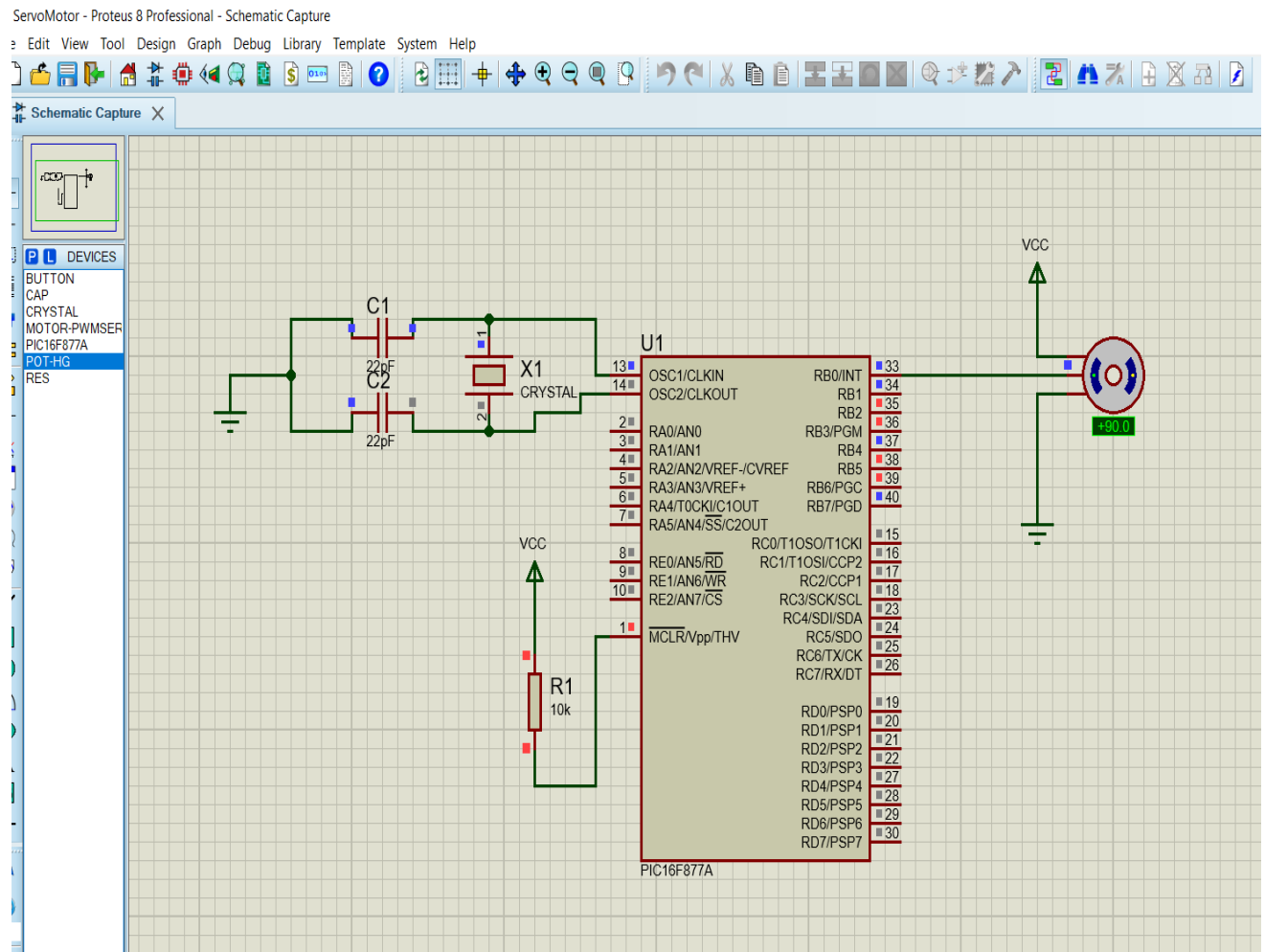


Fig: Interfacing Servo Motor with PIC Microcontroller

CODE:

```
void servoRotate0()
{
    unsigned int i;
    for(i=0;i<50;i++)
    {
        PORTB.F0 = 1;
        Delay_us(800);
        PORTB.F0 = 0;
        Delay_us(19200);
    }
}

void servoRotate90()
{
    unsigned int i;
    for(i=0;i<50;i++)
    {
        PORTB.F0 = 1;
        Delay_us(1500);
        PORTB.F0 = 0;
        Delay_us(18500);
    }
}
```

```

    }
}
void servoRotate180()
{
    unsigned int i;
    for(i=0;i<50;i++)
    {
        PORTB.F0 = 1;
        Delay_us(2200);
        PORTB.F0 = 0;
        Delay_us(17800);
    }
}
void main()
{
    TRISB = 0;
    do
    {
        servoRotate0();
        Delay_ms(2000);
        servoRotate90();
        Delay_ms(2000);
        servoRotate180();
    }
    while(1);
}

```

le Edit View Tool Design Graph Debug Library Template System Help

Schematic Capture X

TERMINALS

DEFAULT
INPUT
OUTPUT
BIDIR
POWER
GROUND
CHASSIS
DYNAMIC
BUS
NC

VCC

C1
22pF

C2
22pF

X1
CRYSTAL

U1
PIC16F877A

U2
ULN2803

U3
7-segment display

R1
10k

VCC

PIC16F877A

ULN2803

7-segment display

CODE:

```
void MSDelay(unsigned char Time)
{
    unsigned char y,z;
    for(y=0;y<Time;y++)
        for(z=0;z<20;z++);
}

void main()
{
    TRISC = 0x00;
    TRISD = 0x00;
    while(1)
    {
        PORTD = 0x80;
        PORTC = 0x00;
        MSDelay(10);

        PORTD = 0x40;
```

```
PORTC = 0xff;  
MSDelay(10);
```

```
PORTD = 0x20;  
PORTC = 0xff;  
MSDelay(10);
```

```
PORTD = 0x10;  
PORTC = 0xdb;  
MSDelay(10);
```

```
PORTD = 0x08;  
PORTC = 0xdb;  
MSDelay(10);
```

```
PORTD = 0x04;  
PORTC = 0xdb;  
MSDelay(10);
```

```
PORTD = 0x02;  
PORTC = 0xc3;  
MSDelay(10);
```

```
PORTD = 0x01;  
PORTC = 0x00;  
MSDelay(10);
```

```
}
```

```
}
```

Circuit Diagram:

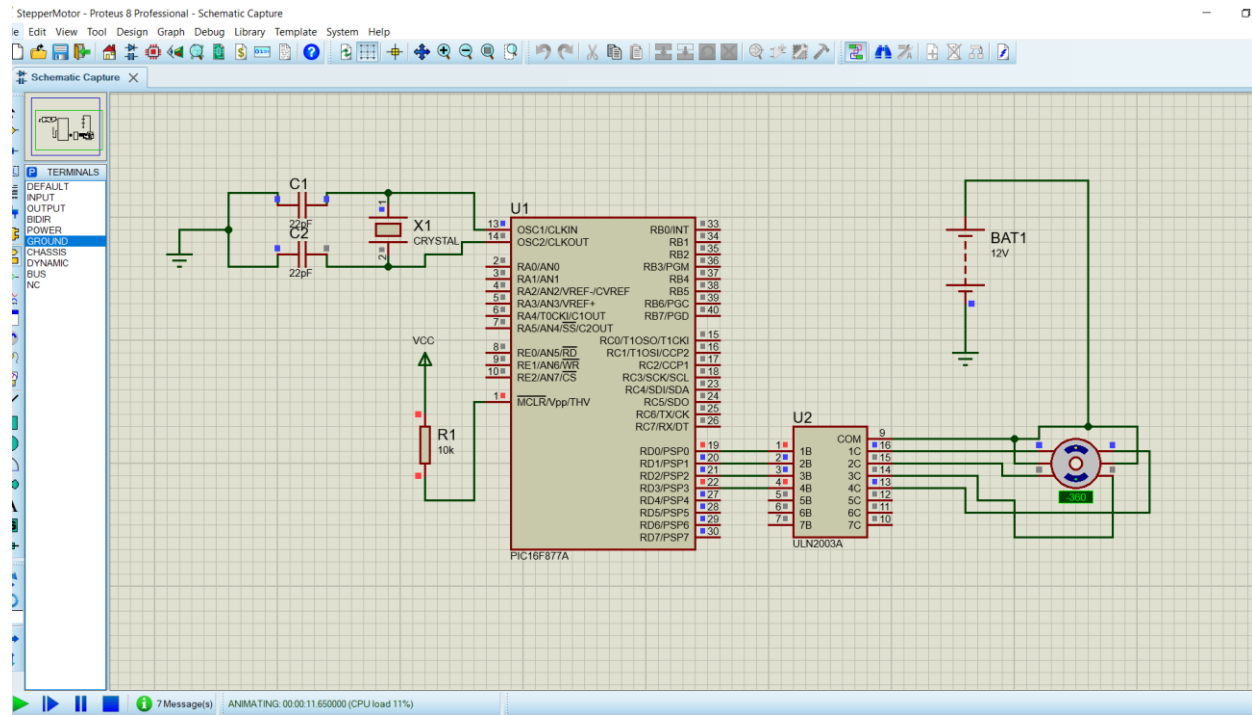


Fig: Stepper Motor Interfacing

CODE:

```
void main()
{
    TRISD = 0b00000000;
    PORTD = 0b11111111;

do
    {
        PORTD = 0b000000011;
        Delay_ms(500);
        PORTD = 0b000000110;
        Delay_ms(500);
        PORTD = 0b000001100;
        Delay_ms(500);
        PORTD = 0b000001001;
        Delay_ms(500);
    }
while(1);
}
```