KONGU ENGINEERING COLLEGE

Perundurai, Erode.

DERPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

20ECE02-MICROCONTROLLER BASED AUTOMATION

PROJECT REPORT

TEMPERATURE DEPENDENT FAN CONTROL SYSTEM USING PIC16F877A

AIM:

To simulate and design the temperature dependent fan control system using pic16f877a.

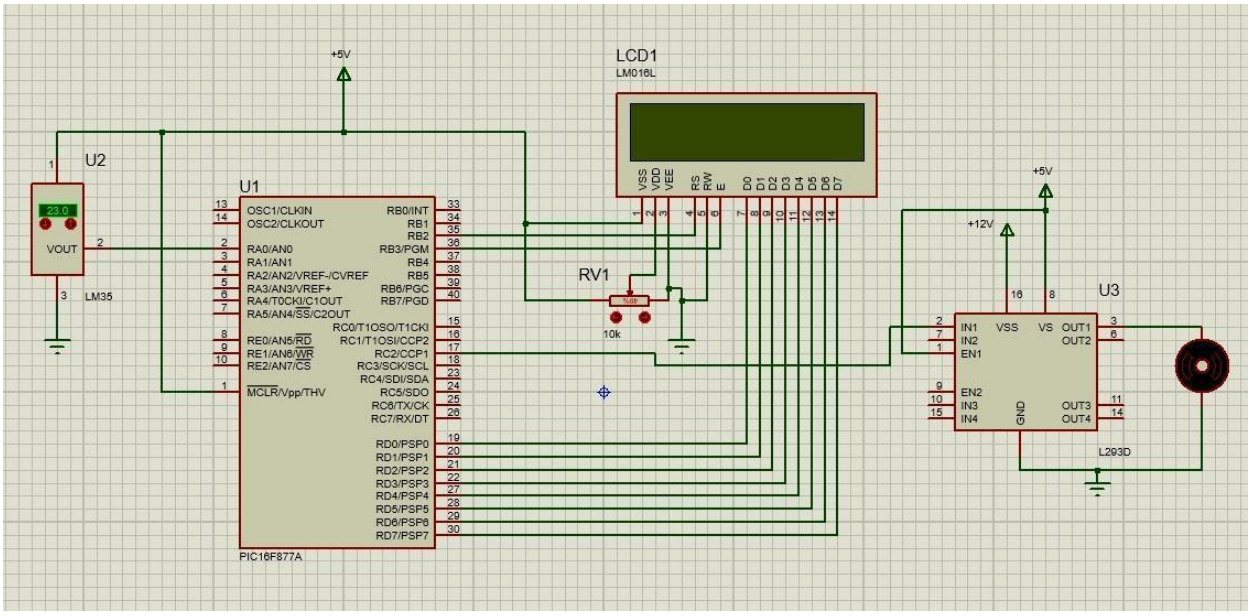
SOFTWARE REQUIRED:

Proteus 8 Professional, PIC CSS.

HARDWARE REQUIRED:

S.No	Component Name	Value	Quantity
1)	Microcontroller	PIC16F877A	1
2)	Crystal Oscillator		1
3)	Temperature sensor	LM35	1
4)	Motor driver	L293D	1
5)	DC motor	9v	1
6)	Potentiometer	10k	1
7)	LCD display		1
8)	Power supply	12v, 9v	1

CIRCUIT DIAGRAM:



WORKING:

- The temperature is measured by means of a temperature sensor LM35.
- The output voltage of the sensor is fed to the A/D channel of the Microcontroller.
- Based on the sensed temperature the speed of the motor is controlled using PWM .
- Several temperature ranges was set in the code to vary the motor speed based on the level of temperature sensed.
- The speed of the motor is controlled by using PWM.
- The motor is driven using a driver IC l293D, See a brief explanation on its working and wiring.

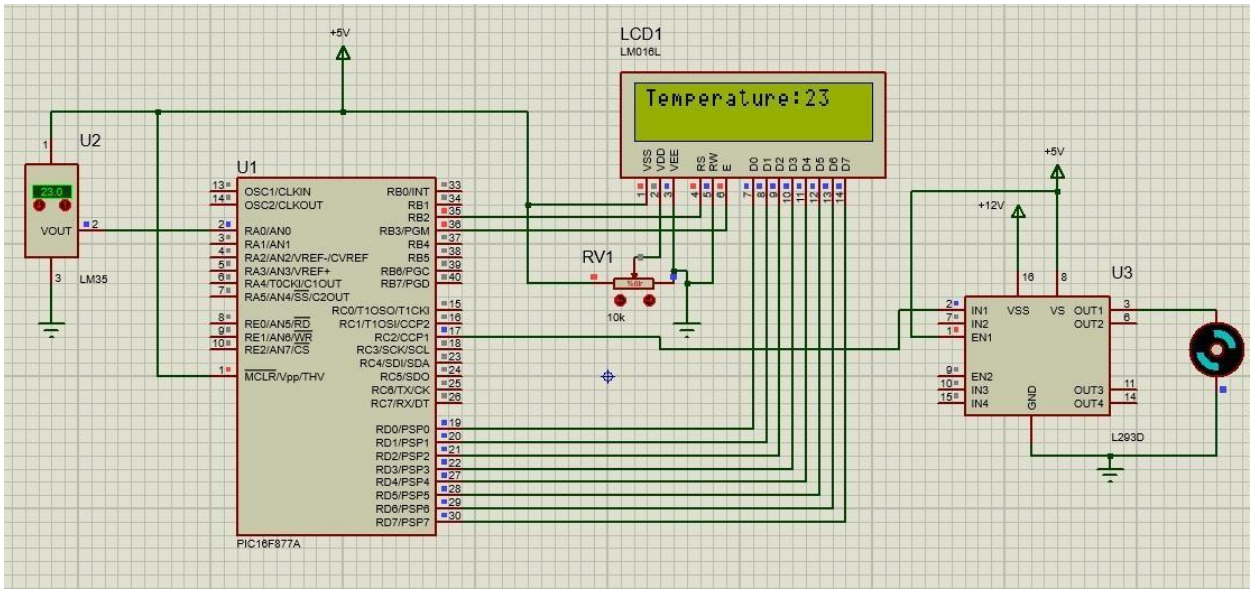
- The rise in temperature will result in increase in speed of the motor and vice versa.These type of Temperature controlled fan systems can generally be used to maintain temperature of a room or object automatically.
- The received analog value is calibrated to display the temperature in the LCD. The “real_value” int value is converted to character using “sprintf” in order to display the temperature values.
- The temperature ranges and duty cycle of the PWM is given using the subroutine “.motor”. So the microcontroller runs a check on the temperature every time and alters the speed of the motor based on it.

PROGRAM CODE:

```
#include <16F877A.h>
#device ADC=10
#FUSES NOWDT
#FUSES NOBROWNOUT
#FUSES NOLVP
#use delay(crystal=20000000)
#byte lcd=0x08
#byte TRIS_lcd=0x88
#bit rs=0x06.2
#bit en=0x06.3
#bit TRIS_rs=0x86.2
#bit TRIS_en=0x86.3

long int adc_value,real_value;
int i;
char value[4];
unsigned char text[]="Temperature:";
void display(unsigned char a, int b);
void motor();

void main()
{
 TRIS_lcd=TRIS_rs=TRIS_en=0;
 display(0x38,0);
 display(0x01,0);
 display(0x0c,0);
 for(i=0;i<=12;i++)
  {
   display(text[i],1);
  }
 setup_timer_2(T2_DIV_BY_16,255,1);
 SETUP_ADC_PORTS(AN0);
 SET_ADC_CHANNEL(0);
 SETUP_ADC(ADC_CLOCK_INTERNAL);
   while(TRUE)
    {
      delay_us(20);
      adc_value=READ_ADC();
      delay_us(10);
      real_value=adc_value/2;
      motor();
```

```
        sprintf(value,"%lu",real_value);
        display(0x8c,0);
        for(i=0;i<=3;i++)
          {
           display(value[i],1);
          }
      }
 }

void display(unsigned char a, int b)
  {
    lcd=a;
    rs=b;
    en=1;
    delay_ms(10);
    en=0;
    delay_ms(10);
  }

void motor()
{
  if(real_value<10)
   {
   setup_ccp1(CCP_OFF);
   }
  if(real_value>10)
   {
     setup_ccp1(CCP_PWM);
      if(real_value>=10&&real_value<=29)
      {
        set_pwm1_duty((int16)204);
      }
      else if(real_value>=30&&real_value<=69)
     {
       set_pwm1_duty((int16)510);
      }
      else if(real_value>=70&&real_value<=99)
      {
         set_pwm1_duty((int16)715);
       }
      else if(real_value>=100&&real_value<=150)
       {
         set_pwm1_duty((int16)919);
       }
    }
}
```

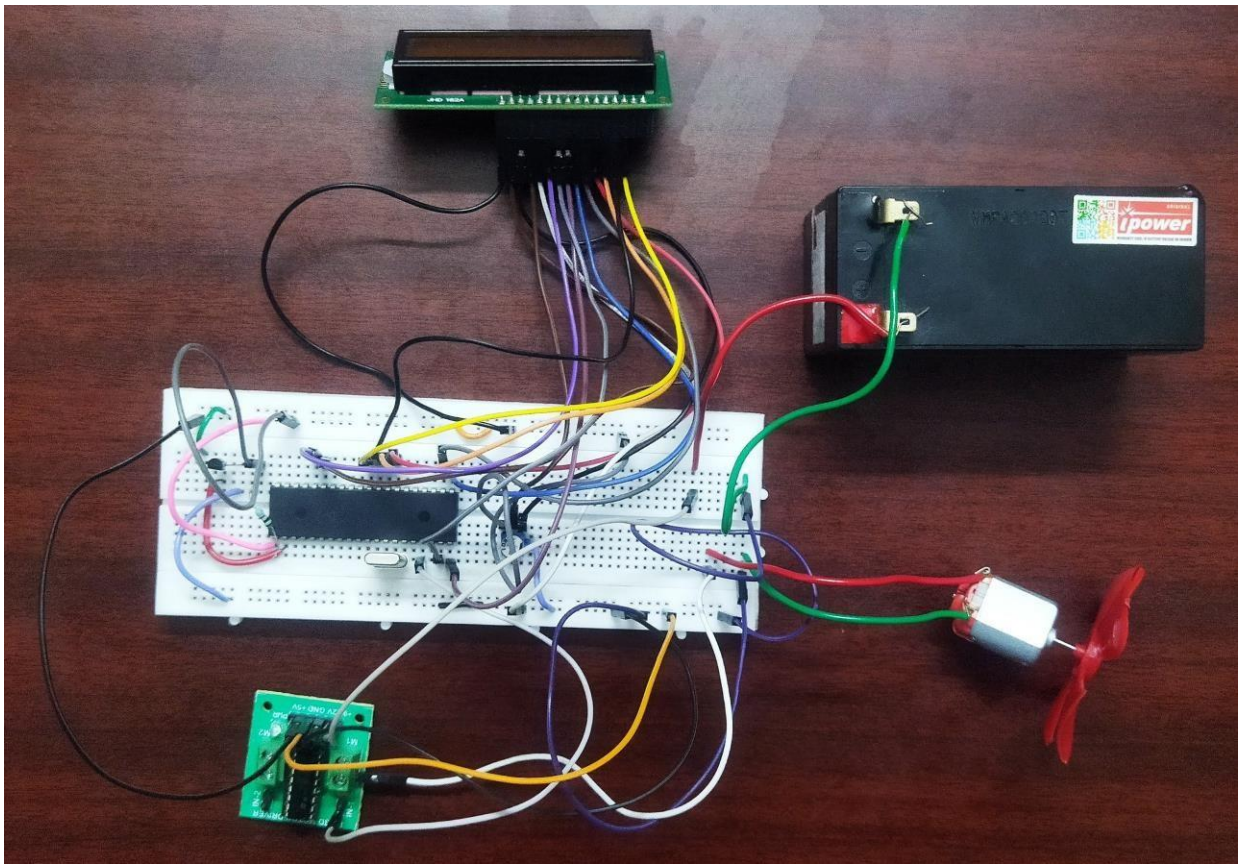
SIMULATION OUTPUT:



VIDEO LINK:

https://drive.google.com/file/d/1QG4QEs-XzDDqwN1mI3ek-xbmEeqDxJhd/view?usp=sharing

HARDWARE OUTPUT:



VIDEO LINK:

https://drive.google.com/file/d/1A84PpG_03NUvajDlF0TCT1Zk8ZHtZBSs/view?usp=sharing

RESULT:

Thus, a temperature dependent fan control system using pic16f877a microcontroller was stimulated and designed.