TEMPERATURE CONTROLLED DC FAN BY USING THERMISTOR

Abstract:

An electronic circuit employing an Arduino board can be used to create the temperature-based fan speed control system. Due to an increase in the use of electrical equipment in both home and commercial appliances, the consumption of electricity has reached a high level. Electricity conservation and management are significant since power is essential for a country's industrial and economic development. In this work, Arduino board for fan speed control because it is now one of the most innovative electronic circuits available. The proposed system is built to measure the room's temperature and transmit that data to the Arduino board. The Arduino board then runs a programme that compares the current temperature to the set temperature. With an Arduino board and an LM-35 temperature sensor, this project suggests how to create temperature-based fan speed control and monitoring. The suggested work uses a microcontroller to dynamically and more quickly adjust the speed of an electric fan. The LCD utilised in the project makes it user-friendly and displays temperature changes as well as the fan's efficiency in real time. As a result, this project helps to conserve energy and utilise less of it for its intended purpose.

Introduction:

Electric fans are among the most widely used electrical appliances because of how affordable they are and how efficiently they use power. In many applications, it is also one of the most popular and widely used. Offering a comfortable and energy-efficient space is one of the sane options.

Today, the need for precise temperature control and air freshening control has spread to many industrial sectors, including process heat, automotive, industrial settings, and office buildings where cool air is used to keep occupants comfortable. The achievement of the desired temperature and consumption optimisation are two of the key issues in the heat domain. Therefore, an automatic temperature control system is required for the purpose of regulating fan speed in response to temperature change. This project suggests using a user-friendly strategy to lower electricity usage and increase electrical component efficiency. As science and technology have advanced, electrical equipment efficiency has had to be improved. Only then can cutting-edge technology rule the forthcoming digital era. The LM 35 sensor, which senses the room's temperature and adjusts the fan speed proportionately to the environmental room temperature, would take the place of the regulator used in standard model fans in household and industrial equipment in the proposed work. By employing technology, this effort encourages consumers to use less energy in their homes and workplaces. Benefits will come from numerous studies focusing on the use of automatic temperature control systems in various industries.

Because of its inexpensive cost and advantages in low power consumption, the electric fan is one of the most well-known electrical devices. It is a common circuit that is used in many different applications. Offering a comfortable and energy-efficient space is also one of the smartest alternatives. The fan has really been used for a long time and is still available on the market. By pressing on the switch button, the fan can be physically controlled. Currently, a change in temperature does not cause the fan speed to alter. However, the fan's actual speed might alter depending on usage. Therefore, an automatic temperature control system technology is required for the purpose of adjusting the fan speed in response to temperature variations.

3.1. List of Components:

Following are the components which are used for this project-

3.1.1 Arduino:

A microcontroller is the computer control system on a signal-cheap; it has numerous electronic circuits built into it, which can decode written instructions and convert them to electrical signal; the microcontroller will then step through these instructions and execute them one by one. As an example, a microcontroller could be used to control the fan speed according to the room's temperature. There are different types of microcontrollers; this project only focused on the smallest microcontrollers [10-11-12]. A microcontroller board called Arduino is utilized to manage the entire circuit. Here, an ATMEGA328P microcontroller is built into the Arduino board. For the proposed work, an IC (integrated circuit) is an ATMEGA328P. This IC's primary function is to regulate the circuit that controls all of the parts attached to the various ports of the Arduino. For the highest possible output as well as high precision of readings displayed in the output of the LCD screen, the temperature-based fan control uses an ARDUINO UNO board. This board is programmable and uses the C programming language as its code input. The advantage of Arduino over the other boards is because many circuit embedded boards available in the market are not compatible with the code such as C, C++ etc...This board comes with a crystal oscillator with a frequency of 16MHz.This type of oscillator is employed in this board because it can deal with the time delay, time synchronization etc., Arduino has both digital as well as analog pins which is used as different ports for the input as well as output ports. The function of the voltage regulator which regulates the voltage given to the ARDUINO board circuit, since if some un-regulated voltage supply comes to the board it regulates the supply.

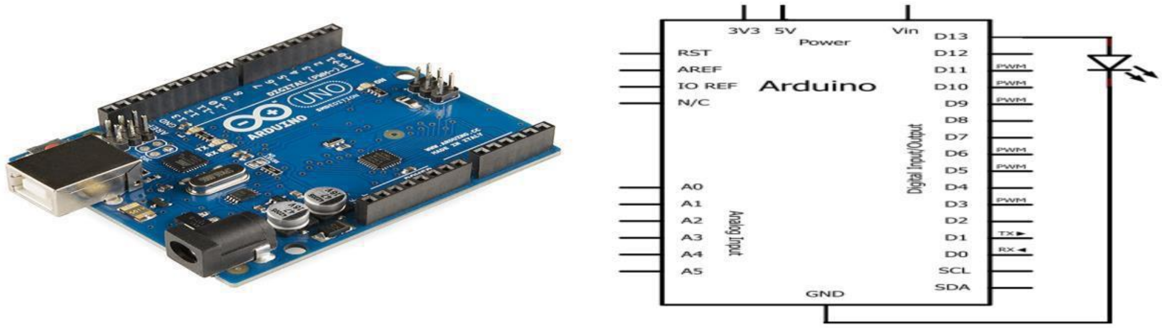


Fig.3.1.1 Pin Diagram of

3.1.1.1 Arduino IDE:

Languages like C and C++.While C++, a well-known object-oriented programming language, is one of the languages used here. The programme is written in embedded C because temperature-based fan control employs this language. The fan works. The ARDUINO IDE is a piece of software that only works with ARDUINO boards. Platform uses the two in two separate temperature ranges, larger than and less than 30˚C. When the room temperature exceeds 60 degrees Celsius, the fan functions at its peak efficiency. Windows 7 is the operating system in use here, and it offers the ARDUINO IDE programmed a system environment setup. The term "Integrated Development Environment" (IDE) is also used.

Specifications:

- Microcontroller: ATmega328
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6 DC
- Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA

3.1.2 LCD Display (liquid Crystal Display):

This component cannot be actuated by conventional IC circuits because it was made specifically to work with microcontrollers. It is utilized to show various messages on a tiny liquid crystal display. Messages can be displayed in two lines of 16 characters each. Additionally, it may show Greek letters, all of the alphabet's letters, punctuation, mathematical symbols, etc. The LCD (2 x 16 characters) and its connection are shown in Fig. 3.1.2 there are several applications for LCD all over the place, including as an electronic display component. Used mostly exclusively for industrial reasons to display the vast range of characteristics in projects, household uses, and other daily applications. In this project, user-friendly and highly clear displays are used to display metrics like the fan's efficiency, which is measurein %, and the room's temperature. In order to manage fan speed based on temperature, a 16x2 display is used, with 16 letters per line and two fixed lines. The LCD type of display is chosen over the LED type for a variety of reasons, including the ease with which it can be purchased affordably, the extent to which it can be customized, and the fact that it supports the more commonly used ASCII characters. Considering that ASCII is the global coding standard for electronic data exchange. Comparatively speaking to LED when ASCII standard is not followed, this LCD makes it simple to delete, insert, and re-enter characters. The optimum way to display the output is hence to choose an LED over an LCD display.

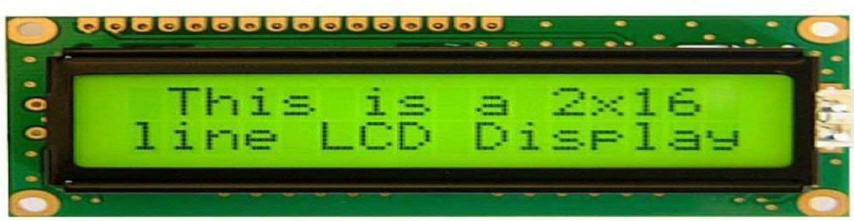


Fig. 3.1.2 LCD Display (liquid Crystal Display)

3.1.3 LM-35 Temperature Sensor:

The LM35 is an IC-based sensor that measures the temperature of a space and its surroundings, with an electrical output that is directly proportional to the temperature. Compared to a thermistor, the LM35 temperature sensor provides more accurate temperature readings. The LM35 is an alternative to a thermistor. Additionally, it generates voltage as an output in line with the ambient temperature in Celsius. The LM35 sensor doesn't need to be calibrated or trimmed externally. Additionally, it keeps a high degree of temperature precision. The LM35 is used in this research since it is widely available and is a low-cost sensor with a good degree of accuracy for monitoring environmental temperature. It is a three-pin sensor with a +5V input, an output, and GND.

Fig. 3.1.3 LM-35 Temperature Sensor

3.1.4 12V DC Fan:

When a 12 volt direct current (DC) supply is used as input, a fan will operate. By using the temperature as an input and controlling the fan's efficiency as an output, the LM35 temperature sensor regulates the fan's speed. This 12V DC fan has rotor or impeller and blades, just as other varieties of conventional model fans.

Fig. 3.1.4 12V DC Fan

3.1.5 2N2222:

This project makes use of the 2N2222 NPN bipolar junction transistor. By magnifying the lesser power, the

2N2222 BJT is utilized for general purpose applications. This application uses the 2N2222 BJT for switching purposes. It has base, emitter, and collector just like a standard transistor. It is designed to function at moderate voltages and relatively fast speeds with low to medium current ranges. To-18 metal is used in its construction. This transistor is utilized in this project because it is readily available, widely used, and reasonably priced. It is also adaptable. The popularity of this transistor makes its ongoing use obvious. Small signal transistors of this kind are extensively used in modern electronics.

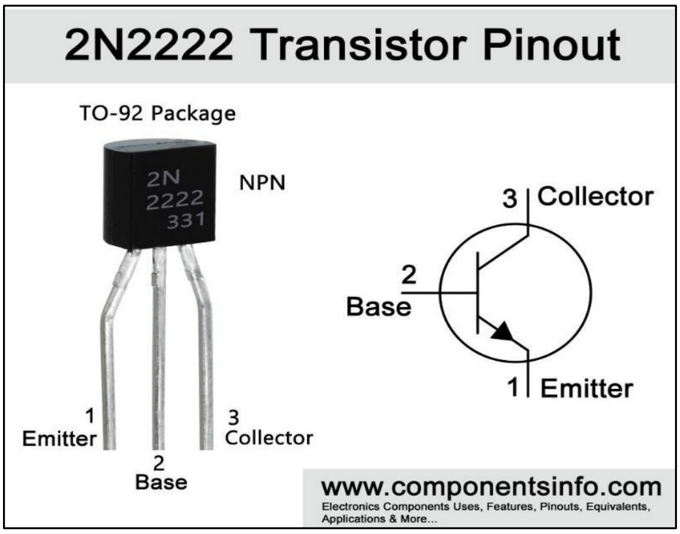


Fig. 3.1.5 2N2222

3.1.6 Resistor:

In a circuit, resistors are typically used to obstruct the flow of current. Afterward, it shields the circuit from the erratic flow of electricity. At normal temperature, the resistor also complies with Ohm's law. In our research, we are using carbon composition resistors, or CCRs, which are made of solid, resistive cylindrical elements. Using a color-coding technique sprayed on the resistor, the value of the resistance is determined. Since carbon, an extremely nonconductive material, makes up this resistor, current is being employed.

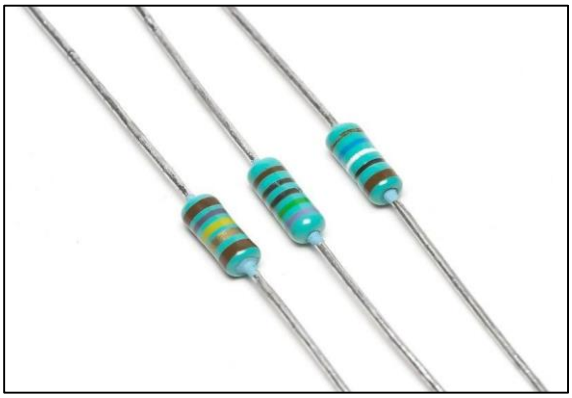


Fig. 3.1.6 Resistor

The temperature of the room and the fan's effectiveness are shown on an LCD shield. For powering the circuit, a 12V battery is used. A connection is made between the LM35's positive and negative ends and the bread board. The positive end of the capacitor, the base of the 2N2222 transistor that serves as the circuit's switch, and the resistor's 1K ohms are all connected to the Arduino board's digital pin -11. The negative of the 2N2222 is linked to the collector side of the transistor, which serves as a switch, and the collector of the transistor is connected to the collector side of the 12V DC fan via the IN4007 and the collector side of the transistor. The output pin of the LM35 is connected to the analog pin of the Arduino board. The LED pin is used to indicate whether the speed of the fan crosses minimum and maximum threshold speed of the fan.

Simulation

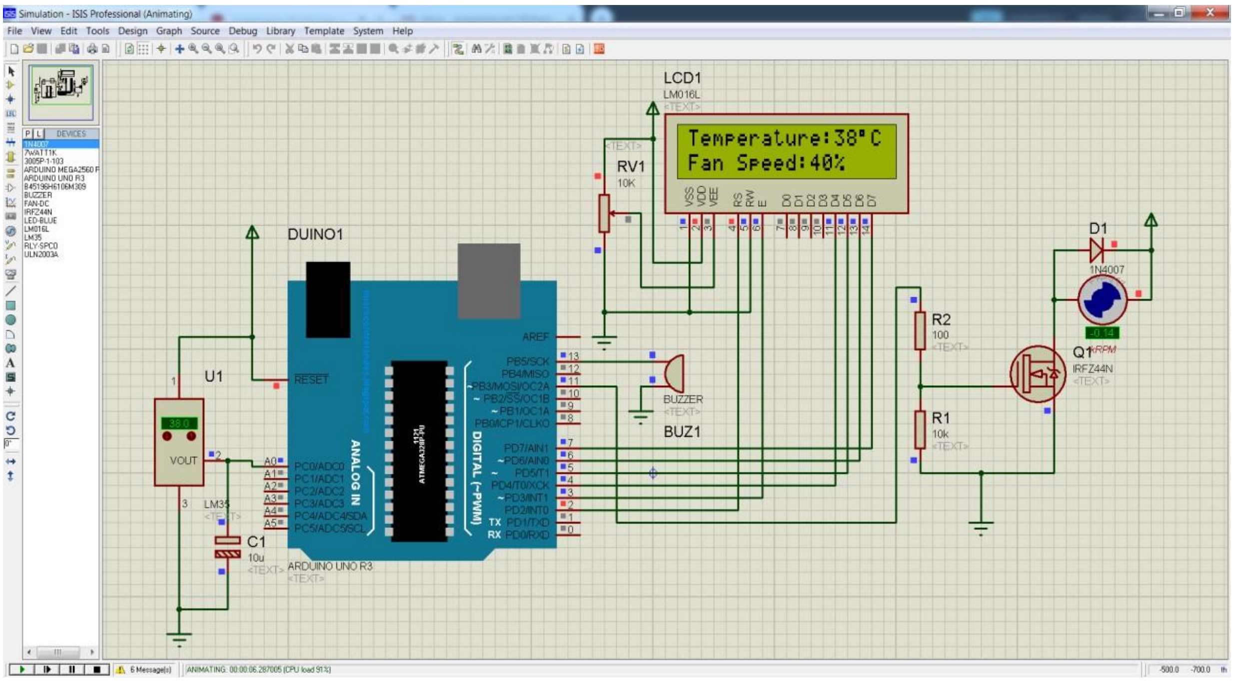


Fig 3.1.7: Circuit diagram of our project

Code

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(2,3,4,5,6,7);
int tempPin = A0; // the output pin of LM35
int fan = 11; // the pin where fan is
int led = 8; // led pin
int temp;
int tempMin = 30; // the temperature to start the fan 0%
int tempMax = 60; // the maximum temperature when fan is at 100%
int fanSpeed;
int fanLCD;
```

```
void setup() {
pinMode(fan, OUTPUT);
pinMode(led, OUTPUT);
pinMode(tempPin, INPUT);
lcd.begin(16,2);
Serial.begin(9600);
}
void loop()
{
temp = readTemp(); // get the temperature
Serial.print( temp );
if(temp < tempMin) // if temp is lower than minimum temp
{
fanSpeed = 0; // fan is not spinning
analogWrite(fan, fanSpeed);
fanLCD=0;
digitalWrite(fan, LOW);
}
if((temp >= tempMin) && (temp <= tempMax)) // if temperature is higher than minimum temp
{
fanSpeed = temp;//map(temp, tempMin, tempMax, 0, 100); // the actual speed of fan//map(temp,
tempMin, tempMax, 32, 255);
fanSpeed=1.5*fanSpeed;
fanLCD = map(temp, tempMin, tempMax, 0, 100); // speed of fan to display on LCD100
analogWrite(fan, fanSpeed); // spin the fan at the fanSpeed speed
}

if(temp > tempMax) // if temp is higher than tempMax
{
digitalWrite(led, HIGH); // turn on led
}
else // else turn of led
{
digitalWrite(led, LOW);
}

lcd.print("TEMP: ");
lcd.print(temp); // display the temperature
lcd.print("C ");
lcd.setCursor(0,1); // move cursor to next line
lcd.print("FANS: ");
lcd.print(fanLCD); // display the fan speed
lcd.print("%");
delay(200);
```

```
lcd.clear();
}

int readTemp() { // get the temperature and convert it to celsius
temp = analogRead(tempPin);
return temp * 0.48828125;
}
```

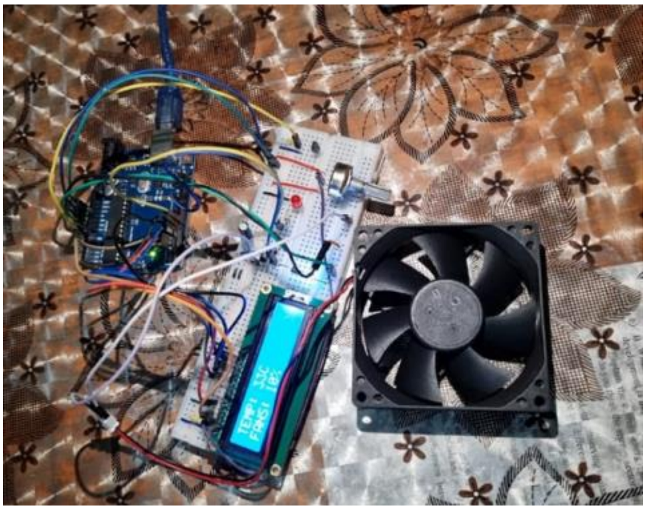
Cost:

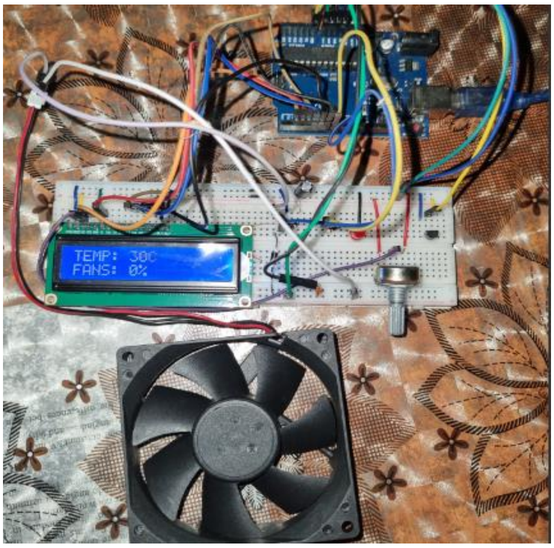
Component	Quantity	Estimated Cost (BDT)
Arduino Uno R3	**1**	**1100**
LM35 Temperature Sensor	**1**	**80**
DC Fan	**1**	**100**
LCD 16X2	**1**	**180**
NPN Transistor	**1**	**25**
Diode	**1**	**5**
Potentiometer	**1**	**20**
9V Battery	**1**	**80**
Breadboard	**1**	**100**
Jumper Wires (23 units)	**1 pack**	**50**
Total		**1850(Apx)**

3.2.1 Software

Programming in the Arduino programming language is used to create the software for the automatic temperature controller and monitor circuit. Software called Arduino IDE is used to programme the Arduino Uno. The Arduino Uno's ATmega328P has a built-in boot loader that enables users to upload new code to it without the need for an external hardware programmer. Connect the Arduino board to the computer, then in the Arduino IDE, choose the proper COM port. Put the programme (sketch) together. The programme is then uploaded to Arduino through a regular USB port after choosing the relevant board from the Tools Board menu in the Arduino IDE.

Result and discussion:





The design and building of a fan speed control system to regulate room temperature are explained in this paper. To gauge the temperature of the room, a temperature sensor was carefully selected. Additionally, the Arduino was effectively coded using C/C++ Language to compare temperature and standard temperature, set fan speed, and display their attributes on LCD. The microcontroller had been used to regulate the fan speed using the fan speed in rpm. A temperature-controlled fan is implemented using Arduino. Thus, the fan speed in this instance has been managed by an Arduino board in accordance with the temperature detected with the aid of a temperature. The project's goal is automatic temperature adjustment of the fan. Additionally, if the room temperature rises, the fan speed will follow suit. In conclusion, the technology that is now being created for programmed control works fairly well for any temperature change. Future versions of the suggested system could incorporate cloud-based predictive analytics.