International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 10 | October - 2022 Impact Factor: 7.185 ISSN: 2582-3930

Temperature Based Fan Speed Controller Using Arduino

[1,]*Raksha R Badigannavar, E&C Tontadarya college of Engineering Gadag.*

[2,]*Amaresh M Kavadamatti E&C Tontadarya college of Engineering Gadag.*

[3,]*Gururaj B Kulkarni E&C Tontadarya college of Engineering Gadag.*

Corresponding Author: Dr. Shailaja M. E&C Tontadarya college of Engineering Gadag.

----------***----------

ABSTRACT: We all know that we are moving towards the automation. So basically in this project we will be controlling the speed of the fan according to the temperature of the surrounding .The system will get the temperature from the temperature sensor and it will control the speed according to the temperature set by the user. It also uses the LCD display to display temperature from – 55 to 125c and speed of the fan in percentage. The Arduino device plays a major role as the processing part. If the surrounding temperature is greater than set temperature fan turns on else vice-versa.

By this project it is helpful to save the energy where saving of energy is one of the most important thing we are facing load shedding.

KEYWORDS: Arduino UNO, LM35, LCD display.

I. INTRODUCTION:

In this project we will be using the concept of firing angle to control fan speed. Basically the project is based on controlling fan speed with respect to the temperature.

The system will get the temperature from the temperature sensor via the ADC and it will control the speed according the set by the user. The temperature is set in the source code are used to increment and decrement the temperature value according to the user requirement.

In this project, an ardunio Uno forms the processing part, which firstly senses the temperature via the ADC the controller then compares the data with the set temperature which the user can set via the keypad. If the current temperature is greater than the set temperature the controller turns ON the fan and the set speed will be proportional to the difference between the set temperature and the current temperature. If the current temperature is less than the set temperature the fan will be turned OFF. We will be using a transistor to provide the require firing angle to the fan and the fan speed will change accordingly. We will be interfacing LCD display which will be used to display the set temperature as well as the fan speed in percentage.

.

II. BLOCK DIAGRAM:

The below fig.1 shows the block diagram of the temperature based fan speed controller,

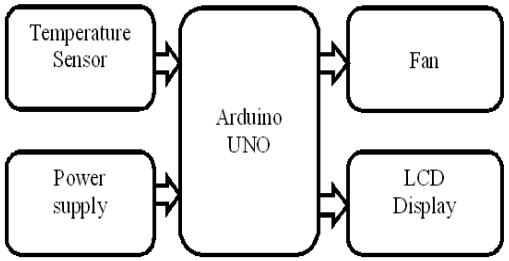

Fig.1

As shown in above block diagram ardunio uno acts as the processing devise or controlling device, temperature sensor and power supply are the input devices to the arduino, fan and LCD display are the output units.

III. CIRCUIT DIAGRAM:

The below fig.2 shows the circuit diagram of the temperature based fan speed controller,

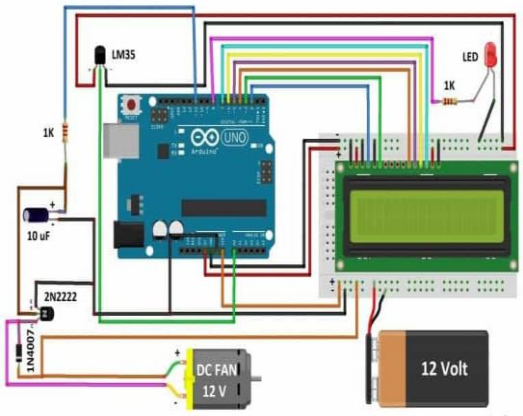

Fig.2

Working: The LM35 temperature sensor sense the souranding room temperature and analog signal is given to the ardunio. Arduino converts the analog singal into digital signal. It calculates the speed of the fan rotation in percentage and display the fan speed and temperature readings in LCD display.

The calculated fan speed is sent to the fan through low frequency pulse width modulation signal. Which adjust the speed of the fan by varying the duty cycle. A 2n2222 transistor is used because it is good efficient and which act as switch and amplify the signal.

IV. MAJOR COMPONENTS USED:

Specification of the major components used in the building of the circuit diagram is as shown in fig.3, fig.4, fig.5.

Arduino UNO:

Fig.3

Specifications:

1. Microcontroller-ATmega328
2. Operating Voltage-5V Input Voltage
3. Input Voltage–6 to 20V
4. Digital I/O Pins-14 Analog Input Pins:6
5. DC Current for the 3.3V Pin-50 mA
6. Flash Memory-32KB SRAM :2KB
7. EEPROM-1 KB
8. Clock Speed-16MHz

LM35

Fig.4

Specifications:

1. Measures directly in degree Celsius
2. Linear+10.0mV/degree Celsius
3. 0.5 degree Celsius accuracy (at+25degreeCelsius).
4. Rated between -55 to +150 degree Celsius range.
5. Low cost due to wafer-level trimming
6. Operates from4 to30volts
7. Less than 60 Microampere current drains.
8. Low self-heating, 0.08 degree Celsius and Nonlinearity only +/-1/4 degree Celsius.

LCD Display

Fig.5

Specifications:

1. Ground(0V)
2. Supply voltage; 5V(4.7V–5.3V)
3. Contrast adjustment;
4. When low command register is selected & when high data register is selected.
5. Low to write to the register; High to read from the

register
6. Sends data to data pins when a high to low pulse is given.
7. 8-bit data pins
8. Back light V_{CC}(5V)
9. Back light Ground(0V)

V. SOURCE CODE:

```
//#include <LiquidCrystal.h>

//LiquidCrystallcd(12, 11, 5, 4, 3, 2);

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd (0x3F, 16,2);

inttempPin = A0

int fan = 11;

int led = 8;

int temp;

inttempMin = 30; inttempMax = 60; intfanSpeed;

intfanLCD;

void setup() {

pinMode(fan, OUTPUT);

pinMode(led, OUTPUT);

pinMode(tempPin, INPUT);

lcd.begin(16,2);

Serial.begin(9600);

}

void loop()

{

temp = readTemp();

Serial.print( temp );

if(temp <tempMin)

{

fanSpeed = 0;

analogWrite(fan, fanSpeed);

fanLCD=0;

digitalWrite(fan, LOW);

}

if((temp >= tempMin) && (temp <= tempMax))

 {

fanSpeed = temp;//map(temp, tempMin, tempMax, 0, 100);

fanSpeed=1.5*fanSpeed;

fanLCD = map(temp, tempMin, tempMax, 0, 100); // speed of fan to display on LCD100

analogWrite(fan, fanSpeed);

}


if(temp >tempMax)

{

digitalWrite(led, HIGH);

}

else

{

digitalWrite(led, LOW);

}

lcd.print("TEMP: ");

lcd.print(temp);

lcd.print("C ");

lcd.setCursor(0,1);

lcd.print("FANSPEED: ");

lcd.print(fanLCD);

lcd.print("%");

delay(200);
```

```
lcd.clear();

}

intreadTemp()

{

 temp = analogRead(tempPin);

return temp * 0.48828125;

}
```

VI. RESULT:

The below fig.6 shows the working module of the temperature based fan speed controller,

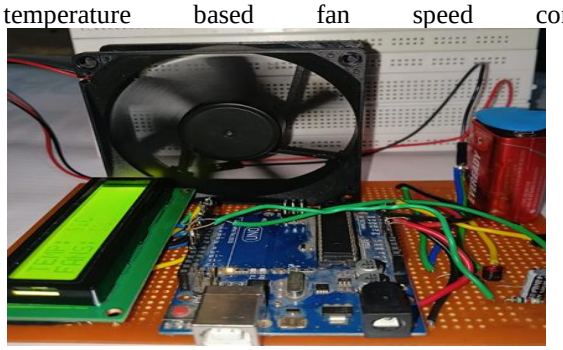

Fig.6

The following table.1 depicts the results obtained by operating the prototype model at various different temperatures. It defines the behavior of the embedded system about how it reacts to variation of temperature at real time.

Temperature	Duty Cycle	PWM Value	Fan Speed
In degree Celsius	In %		In rpm
Less than 26	0%	0	0
26	20 %	51	227
27	40%	102	428
28	60%	153	654
29	80%	204	826
Greater 29	100%	255	1000

Table.1

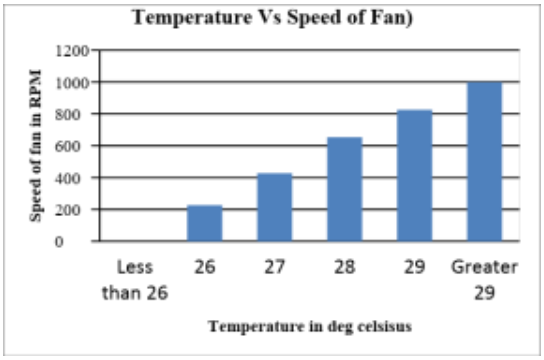

Table.2

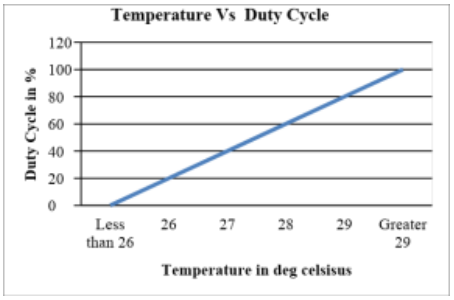

Table.3

VII. CONCLUSION & FUTURESCOPE:

Conclusion: This project elaborates the design construction of fan speed control system to control the room temperature. Efficient temperature sensor was chosen to take the room temperature. Hence the fan speed will increase automatically if the room temperature is increased. As conclusion, the system perform very well for any temperature changes in the souranding and can be classified as automatic control device.

Future scope:

- The proposed model can be improvised further to monitor humidity, light & at at a same time to control them.
- The project will concentrate on electric fan rather than other type of fan such as celling fan.
- We can also draw graphs of variation in the parameters using computer.
- It is used to send the data to the remote location using internet

REFERENCES:

1. Liu, Y., Zeng, J. and Wang, C. (2009). Temperature Monitoring in Laser Assisted Polymer Bonding for MEMS Packaging Using a thin Film Sensor Array , IEEE Sensors Applications Symposium, New Orleans, LA , USA.
2. Surabhi1, Upendra Prasad2 , Vivek Kumar Jain3,Vol. 4, Issue 7, July 2015, Design and Fabrication of Temperature based DC Fan Speed Control System utilizing Microcontroller and Pulse Width Modulation Technique.
3. MehtaV.K & Metha R. (2007). Prinnciple of electronics ,S. Chand & company Ltd. New Delhi,2015