राष्ट्रीय प्रौद्योगिकी संस्थान वारंगल

Nâtional Institute of Technology Warangal

ICA LAB

PROJECT - AUTOMATIC ROOM TEMPERATURE CONTROL SYSTEM

Presented To :

Mrs V. Rama

Department of Electronics and Communication Engineering,

National Institute of Technology, Warangal

Presented By :

Aadhya Komuravelly

202127 (IInd Year B.Tech)

Electrical and Electronics Engineering

Section - A

Acknowledgement

This project has been prepared, completed in order to fulfill the requirement of *Integrated Circuit Applications Lab (EC286)* for the successful completion of this course under *B.Tech II nd Year II nd Semester*. I must remain thankful to many people for the successful completion of this project.

I would like to express my special thanks of gratitude to my honorable Professor, Mrs. V Rama as well as ECE Department and EEE Department who gave me the golden opportunity to do this wonderful project on the topic of **AUTOMATIC ROOM TEMPERATURE CONTROL SYSTEM** which helped me in doing a lot of research regarding different component , Sensors and Circuits and I also came to know about so many new things. I am really thankful to them.

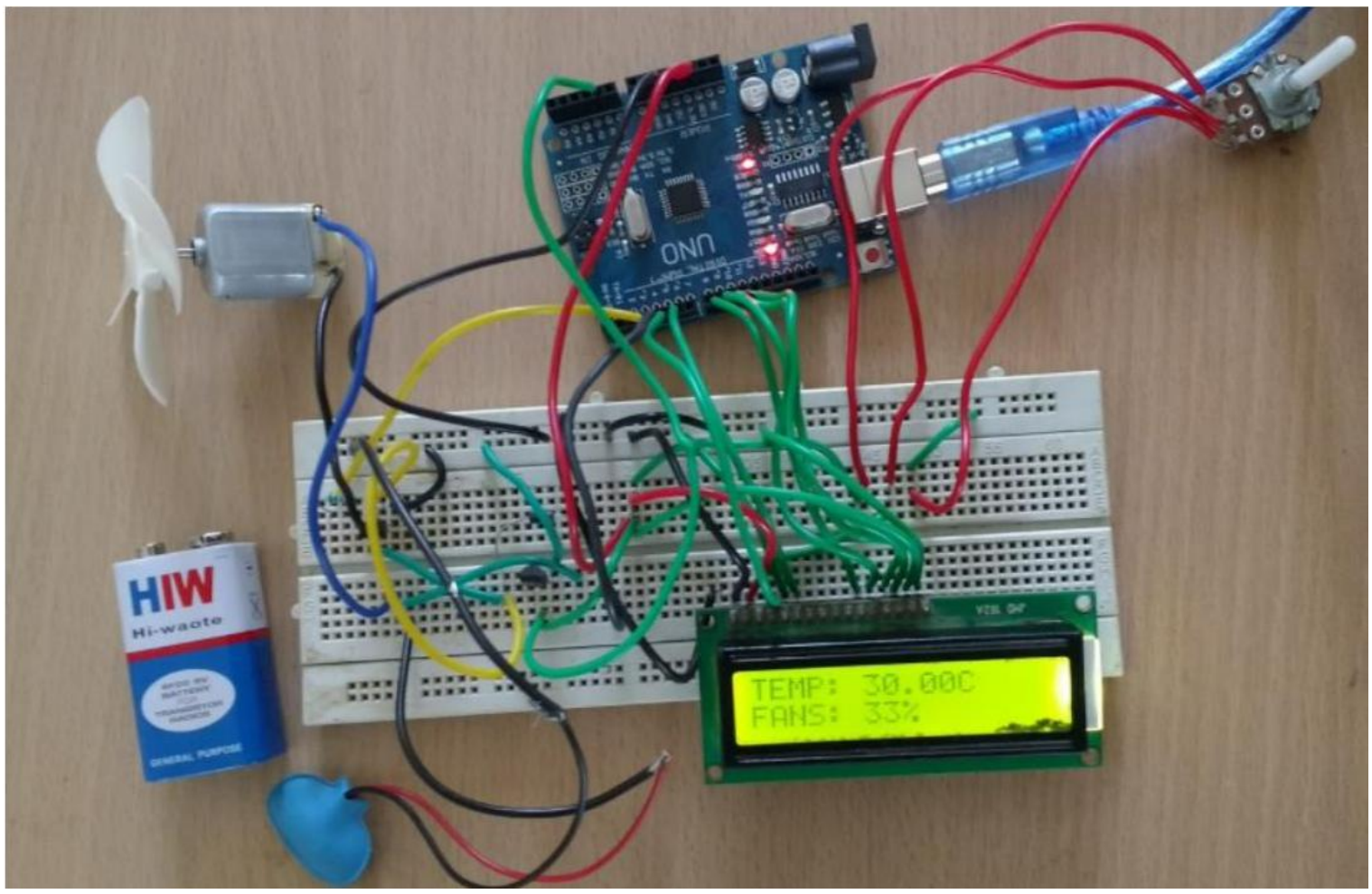
Secondly, I would like to thanks my Parents who helped me in every aspect of my life and providing all resources for completing this Project.

I am also very thankful to all respected Institute members for helping and encouraging in every aspects of my academic study during in the Institute and many thanks to National Institute of Technology for providing me the opportunity to purse my B. Tech in such a wonderful Academic environment.

-AADHYA KOMURAVELLY

TABLE OF CONTENTS

AUTOMATIC ROOM TEMPERATURE CONTROL SYSTEM



ABSTRACT:

This paper presents designing and implementing an Automatic room temperature control system using the Arduino Uno and TMP36 sensor. Here, the user sets the minimum and the maximum reference temperature range from the keypad. The TMP36 sensor senses the surrounding room temperature and gives the result in degrees Celsius. Both the reference and the measured values are displayed on the Liquid Crystal Display (LCD). The Arduino microcontroller, being the processing unit of the system, gets the sensor's measured value and compares it with the set threshold. The results are: when the measured room temperature is less than the minimum of the threshold value; then, the microcontroller turns on the heater. If the measured room temperature is greater than the maximum threshold value, then the fan triggered on.Finally, if the room's measured temperature is between the setpoint range, all the loads are turned off. That means the room temperature is maintained normal.

Automatic temperature control system is an important application used in almost all modern gadgets and smart homes. The system for controlling temperature automatically is achieved by using Arduino Uno-based microcontroller system. Arduino Uno due to its increased popularity finds its varied range of applications. Temperature sensor TMP36 and Arduino Uno are the hardware used interfaced with computer, and the temperature is controlled in the room. Temperature is displayed on LCD display employing A0 pin of hardware with the help of analog pin utilizing pulse width modulation (PWM).

INTRODUCTION:

With the gradual advancement of technology, automation has become part of human life. In this regard, modern technologies have brought several innovations that automatically implement a particular task. Among these discoveries, microcontroller plays a vital role in the smart system of the electronic world. A microcontroller is a control system on a single chip that makes possible for the automation of the designed system and control process and produces precise results. Among all places occupied by a human being, a home is the most important and needs to be maintained in the proper temperature. Nowadays, keeping living and working places at a conducive temperature is not only crucial to be healthy and productive, but also maintaining the room at average temperature helps to prevent spoiling of foods, medicine, and other goods in the room.

Commonly, people use the manually controlled system, air-condition (AC), to regulate the temperature in their living environment. However, this manually operated system has notable limitations. The drawback is that if the user forgets to switch on or adjust the AC when the temperature becomes abnormal, children, disabled persons, and perishable items could be affected. The other problem with the mechanical AC system is, sometimes even if the air condition (AC) is still working, it is difficult to maintain the room temperature. Furthermore, if not appropriately managed,

it may result in unnecessary expense and power usage. In general, its operations always require the user to turn it on and off regardless of the room temperature condition. Therefore, to address these drawbacks, the Automatic Room Temperature Control System is proposed.

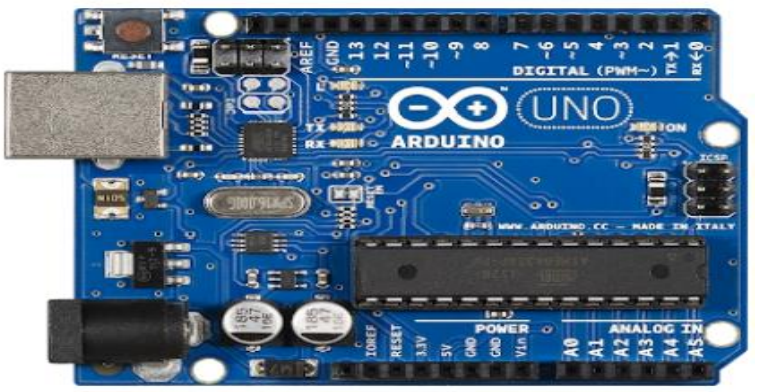
An automatic room temperature control system is a self automated temperature control system that can control the speed of the fan depending on the current room temperature. It comprised of a control unit (MCU), temperature sensor (TMP36), heater, fan, and keypad (3x4) to monitor the room temperature. According to the value of the ambient temperature, the microcontroller compares sensor temperature reading with a set value. Then the microcontroller makes a decision in accordance. This system's main advantages are easy to use, less energy usage, economical, more convenient to control temperature, and user-friendly.

COMPONENTS REQUIRED:

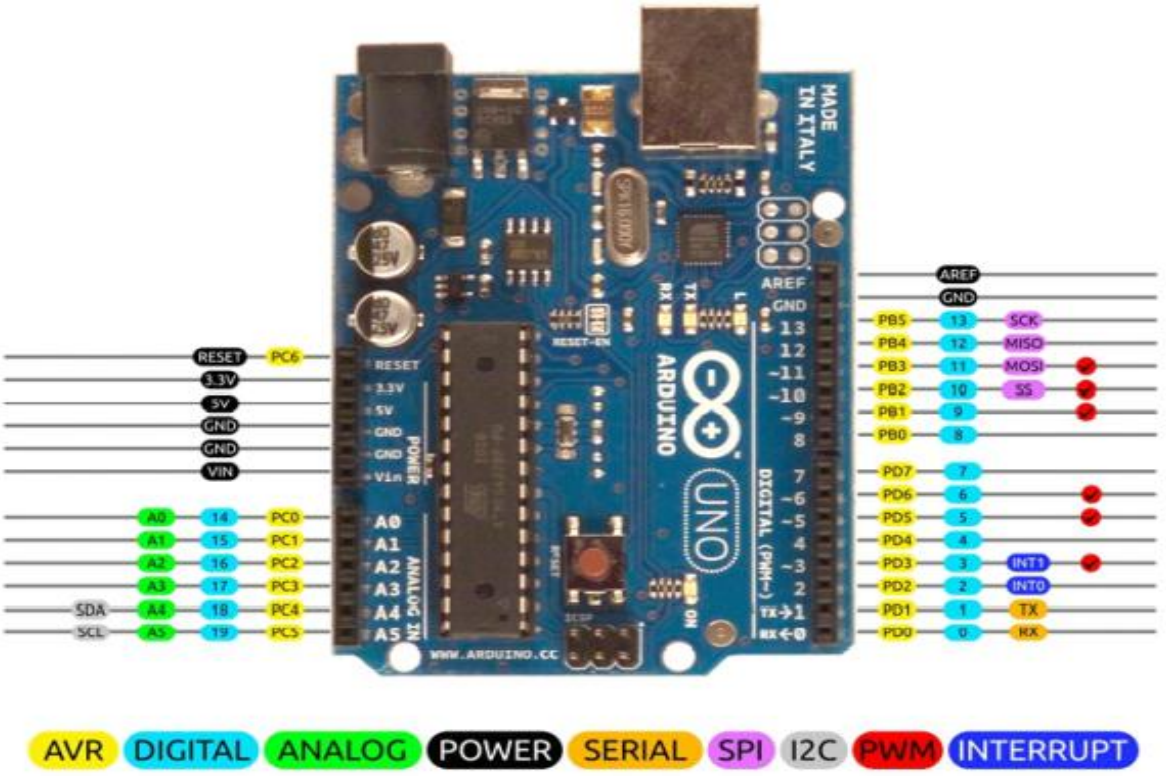
Name	Quantity	Component
U2	1	Arduino Uno R3
U3	1	LCD 16 x 2
Rpot1	1	250 kΩ Potentiometer
D1	1	Red LED
M1	1	DC Motor
R3 R4 R1	3	1 kΩ Resistor
T1	1	NPN Transistor (BJT)
D2	1	Diode
U1	1	Temperature Sensor [TMP36]

1) ARDUINO UNO R3

Arduino Uno R3 is one kind of ATmega328P based microcontroller board. It includes the whole thing required to hold up the microcontroller; just attach it to a PC with the help of a USB cable, and give the supply using AC-DC adapter or a battery to get started. The term Uno means “one” in the language of “Italian” and was selected for marking the release of Arduino’s IDE 1.0 software. The R3 Arduino Uno is the 3rd as well as most recent modification of the Arduino Uno. Arduino board and IDE software are the reference versions of Arduino and currently progressed to new releases. The Uno-board is the primary in a sequence of USB-Arduino boards, & the reference model designed for the Arduino platform.



Arduino Uno R3 (Front View)



AVR DIGITAL ANALOG POWER SERIAL SPI I2C PWM INTERRUPT

Arduino Uno Pin Diagram

Pin description-

Pin Category	Pin Name	Details
Power	Vin, 3.3V, 5V, GND	Vin: Input voltage to Arduino when using an external power source. 5V: Regulated power supply used to power microcontroller and other components on the board. 3.3V: 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50mA. GND: ground pins.
Reset	Reset	Resets the microcontroller.
Analog Pins	A0 – A5	Used to provide analog input in the range of 0-5V
Input/Output	Digital Pins 0 - 13	Can be used as input or output pins.
Pins		
Serial	0(Rx), 1(Tx)	Used to receive and transmit TTL serial data.
External Interrupts	2, 3	To trigger an interrupt.
PWM	3, 5, 6, 9, 11	Provides 8-bit PWM output.
SPI	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Used for SPI communication.
Inbuilt LED	13	To turn on the inbuilt LED.
TWI	A4 (SDA), A5 (SCA)	Used for TWI communication.
AREF	AREF	To provide reference voltage for input voltage.

Arduino Uno R3 Specifications

- The Arduino Uno R3 board includes the following specifications.
- It is an ATmega328P based Microcontroller
- The Operating Voltage of the Arduino is 5V
- The recommended input voltage ranges from 7V to 12V
- The i/p voltage (limit) is 6V to 20V
- Digital input and output pins-14
- Digital input & output pins (PWM)-6
- Analog i/p pins are 6
- DC Current for each I/O Pin is 20 mA

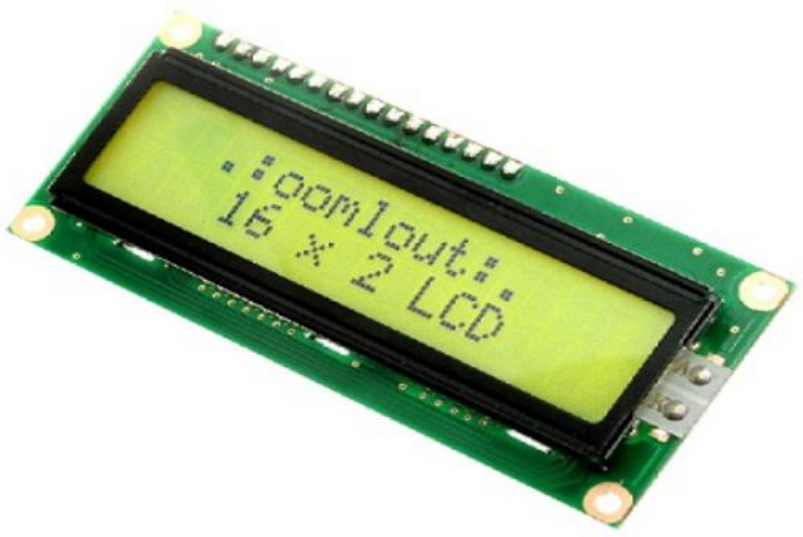
- DC Current used for 3.3V Pin is 50 mA
- Flash Memory -32 KB, and 0.5 KB memory is used by the boot loader SRAM is 2 KB
- EEPROM is 1 KB
- The speed of the CLK is 16 MHz
- In Built LED
- Length and width of the Arduino are 68.6 mm X 53.4 mm
- The weight of the Arduino board is 25

Arduino Uno R3 Programming

- The programming of an Arduino Uno R3 can be done using IDE software.
- The microcontroller on the board will come with pre-burned by a boot loader that permits to upload fresh code without using an exterior hardware programmer.
- The communication of this can be done using a protocol like STK500. We can also upload the program in the microcontroller by avoiding the boot loader using the header like the In-Circuit Serial Programming.

2) LCD 16*2

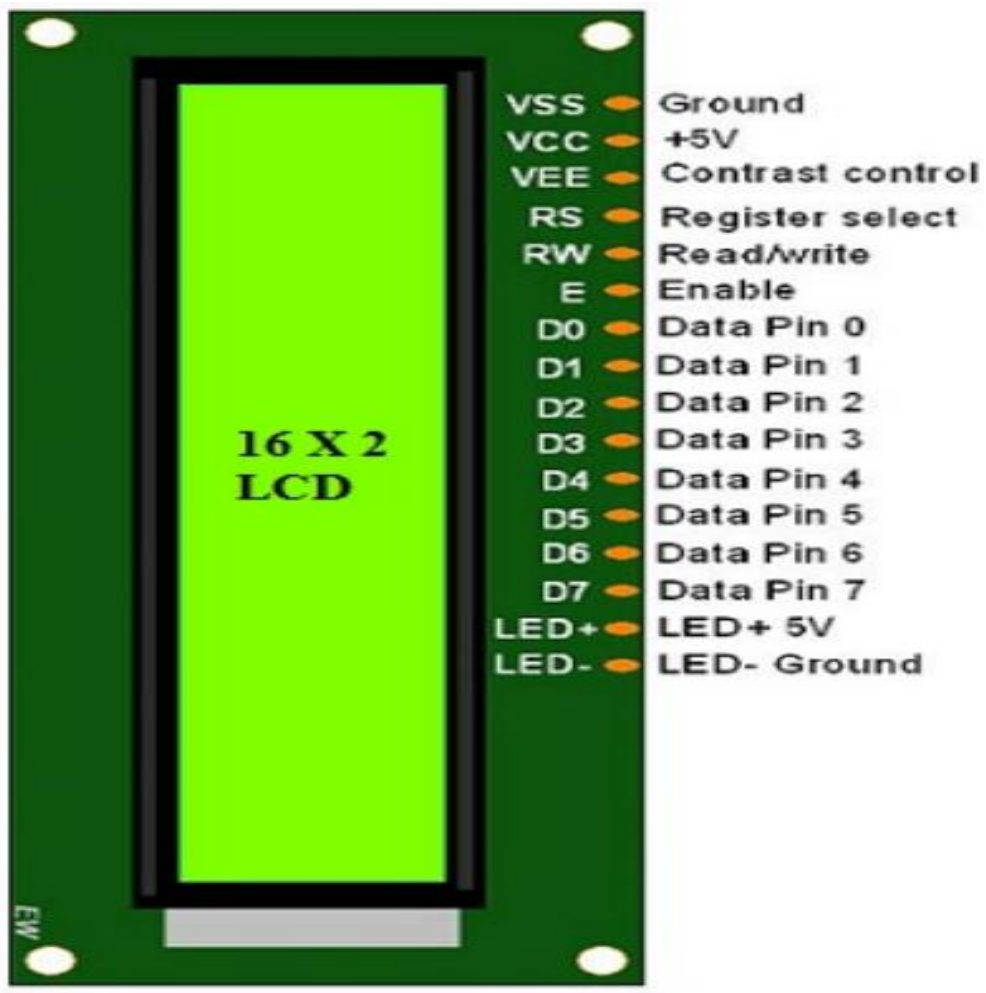
The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.



16X2 LCD

Pin description-

The 16×2 LCD pinout is shown below:



LCD-16×2-pin-diagram

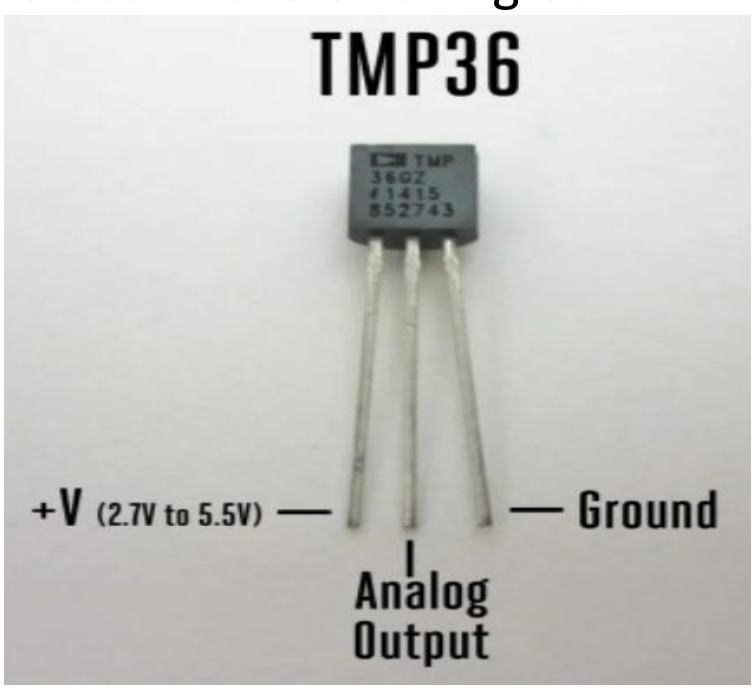
- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.

- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.

3) TEMPERATURE SENSOR-TMP36

TMP36 is a temperature sensor chip which generates an analog voltage at the output which is linearly proportional to the Celsius temperature. Then convert this voltage into temperature based on a 10 mV/°C scale factor. It has a shutdown capability which limits the output current to less than 0.5 µA. It provides a supply current of up to 50 µA.

This sensor provides a highly precise temperature in centigrade. Most importantly, it produces output in dc voltage that we can measure easily with the help of any bare metal microcontrollers such as Arduino Uno, STM32F4, PIC16F877A. On top of that, Celsius's temperature and an output voltage change linearly which makes it easy to compensate temperature/Voltage variations. Having a linear relationship is helpful. Because we will not require any external calibration circuit. Furthermore, it offers a very low output impedance. In short, it is very easy to interface this sensor with ADCs or microcontrollers having built-in ADCs.



Pin description-

- Pin1 (+V) in an input pin. Connect a positive supply at this pin.
- Pin2 (Vout) is an output in which provides an analog voltage. This analog voltage is linearly proportional to temperature (in Celsius).
- Pin 3 (Gnd) is a ground pin.

Features-

- This device operates using a single supply.
- It operates within a range of +2.7 V to +5.5 V.
- This chip is calibrated in Celsius therefore external calibration is not required. It provides an output with ±2°C accuracy over the full temperature range.
- The operating range for temperature is -40 °C to +125 °C. However, it can operate up to +150 °C temperature but accuracy reduces.
- TMP36 has an output scale factor of 10 mV/°C.
- Quiescent current is less than 50 µA.
- The device works well when the supply current is below 50 µA, which offers very low self-heating.

4) POTENTIOMETER

The potentiometer is a three-terminal device. It has a rotating contact that acts as an adjustable voltage divider. The variable resistance measured by the potentiometer can be easily read as an analog value into the Arduino board.The potentiometer structure consists of a sliding contact (called wiper), a resistive element, electrical terminals, and a housing.
The sliding contact moves along the resistive element, while the housing consists of the wiper and the element.

Working:

The fixed input voltage is applied across the two ends terminal of a potentiometer, which further produces the adjustable output voltage at the wiper or slider.

As the slider moves from one end to another, the divider can vary the output voltage from maximum to Ground.

METHODOLOGY:

• Connect Arduino Uno R3, Bread board, LCD screen, LED light, Potentiometer, NPN transistor, DC motor, Temperature sensor (TMP36) along with resistors and diodes wherever required as shown in the circuit diagram below.

• Connect the LCD screen terminals as follows:

1. Power pin to the 5V pin on the Arduino.
2. Contrast pin to wiper pin of potentiometer.
3. Enable pin is connected to any digital pin on the Arduino.
4. GND pin is connected to the GND pin on Arduino
5. Register select pin to any digital pin on Arduino.
6. Read/Write pin GND pin on Arduino.
7. DB4, DB5, DB6, DB7 pins to the digital pins on the Arduino.
8. LED anode and LED cathode are connected using a resistor on breadboard.

• Connect the DC motor via NPN transistor on bread board and connect the pins as follows:

1. Emitter of transistor is connected to GND of Arduino.
2. Base of transistor is connected to any digital pin of Arduino.
3. Collector of transistor is connected to a terminal of DC motor.
4. Another terminal of DC motor is connected to GND.

• Connect the temperature sensor as follows:

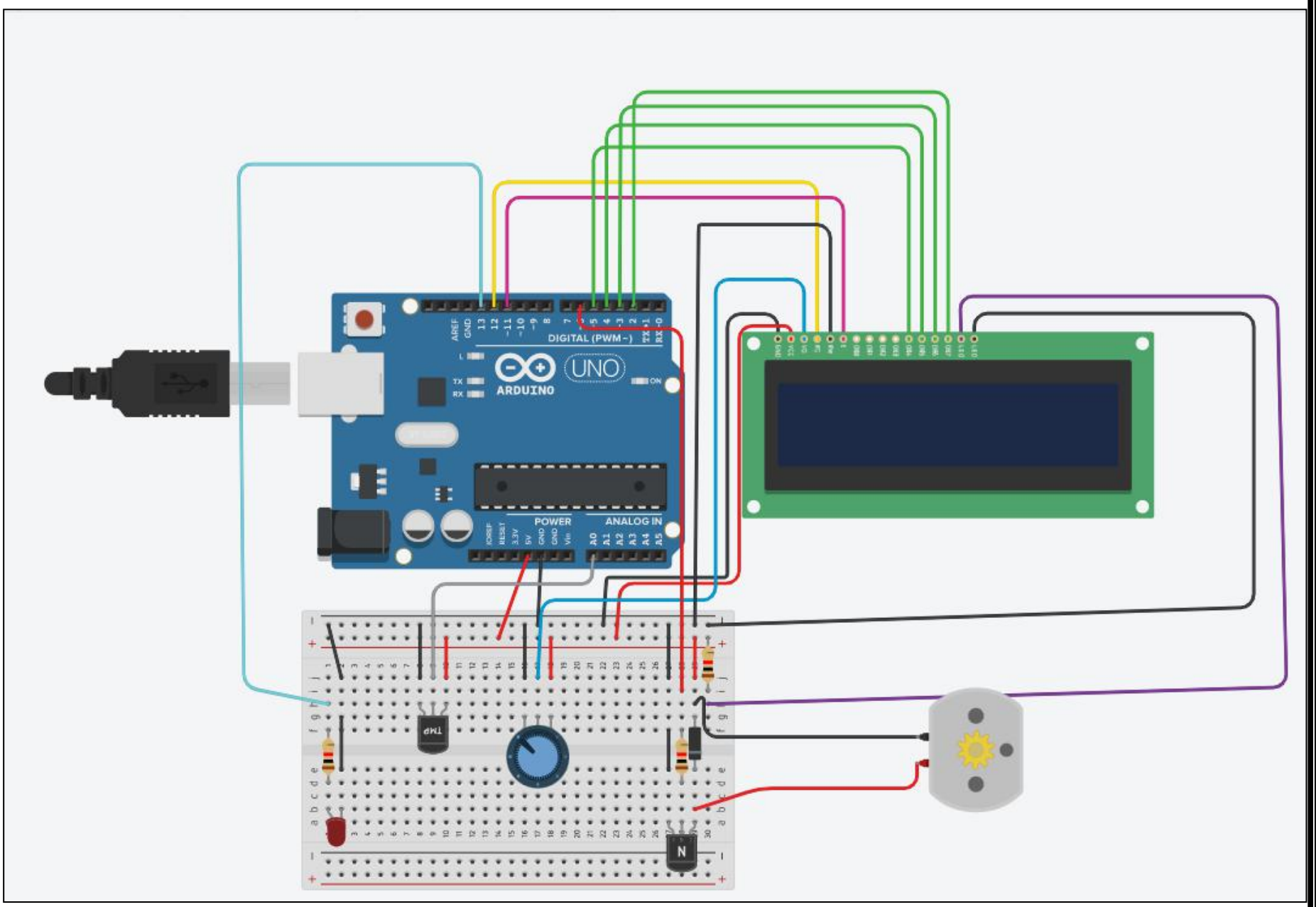
1. GND pin to GND pin of Arduino.
2. V0 pin to any analog pin of Arduino.

3. Power pin to 5V pin of Arduino.

• Connect both the extreme terminals of potentiometer to GND and 5V pin of Arduino.

• Connect the LED (anode pin to any digital pin and cathode pin is grounded).

CIRCUIT:

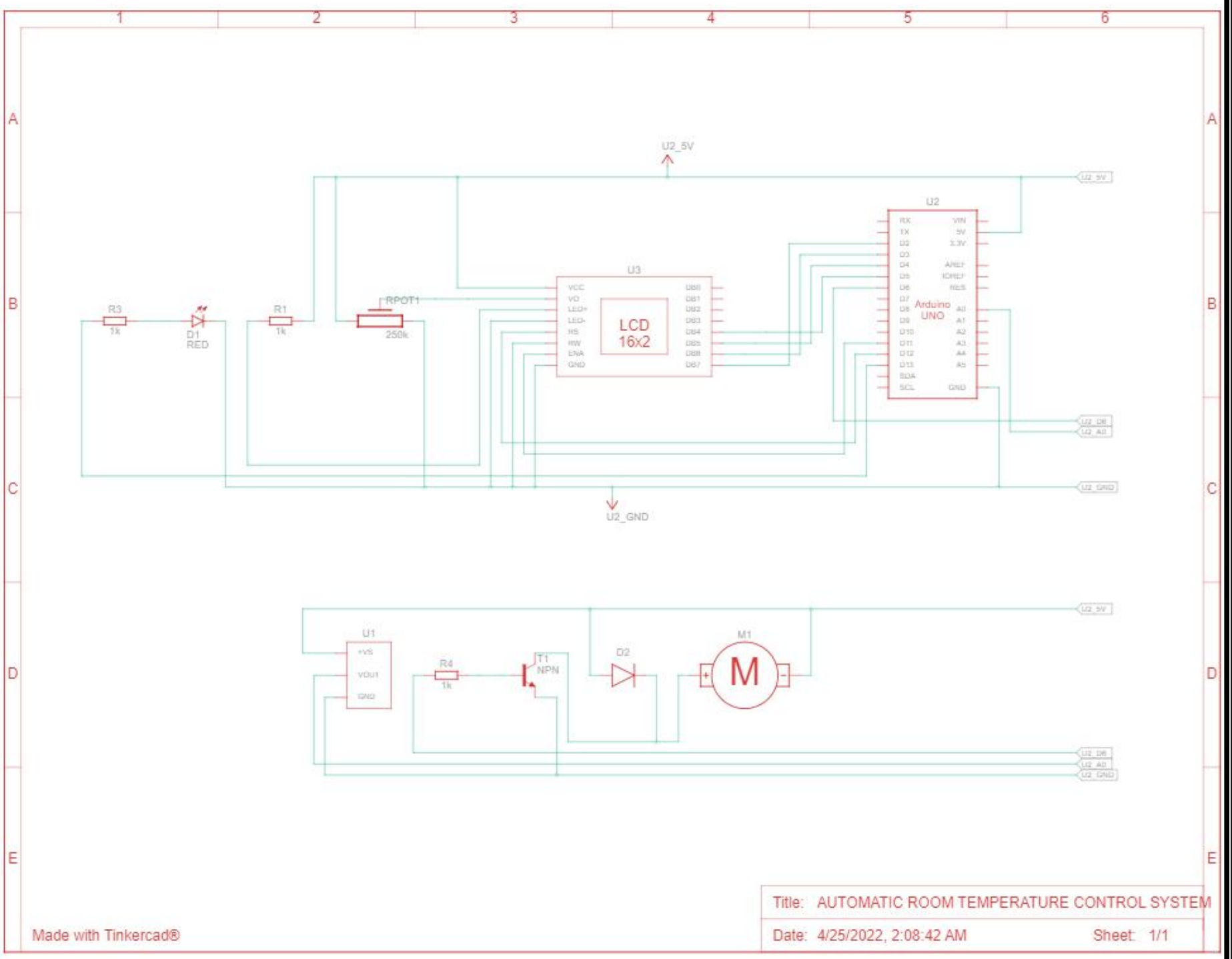


CODE:

```
const int temp_trans_pin=A0;
const int Heater_pin=13;
const int FAN_pin=6;

//Set the range of the desire temperature

float MinTemp=20, MaxTemp=25; //Room Temp [20-25]

#include<LiquidCrystal.h>
LiquidCrystal LCD(12, 11, 5, 4, 3, 2);

void setup(){
LCD.begin(16,2);
  pinMode(Heater_pin,OUTPUT);
  pinMode(FAN_pin, OUTPUT);

  LCD.print("Room Temp(C): ");
  LCD.setCursor(2,1);
  LCD.print(MinTemp);LCD.print("-");LCD.print(MaxTemp);

  delay(2000);
}

void loop(){
  float Eqv_volt, SensorTemp;
  Eqv_volt=analogRead(temp_trans_pin)*5.0/1023;
  SensorTemp=100.0*Eqv_volt-50.0;

  LCD.clear();
  LCD.print("Sensor Reading: ");
  LCD.setCursor(2,1);
  LCD.print(SensorTemp);LCD.print(" C");

  delay(2000);

  if(SensorTemp>MaxTemp){
  LCD.clear();
    LCD.print("Temp is HIGHER!");
    LCD.setCursor(0,1); LCD.print("Turn on FAN!");
    for (int i=0; i<=255; i++){
      analogWrite(FAN_pin, i);
    }
    delay (2000);
```

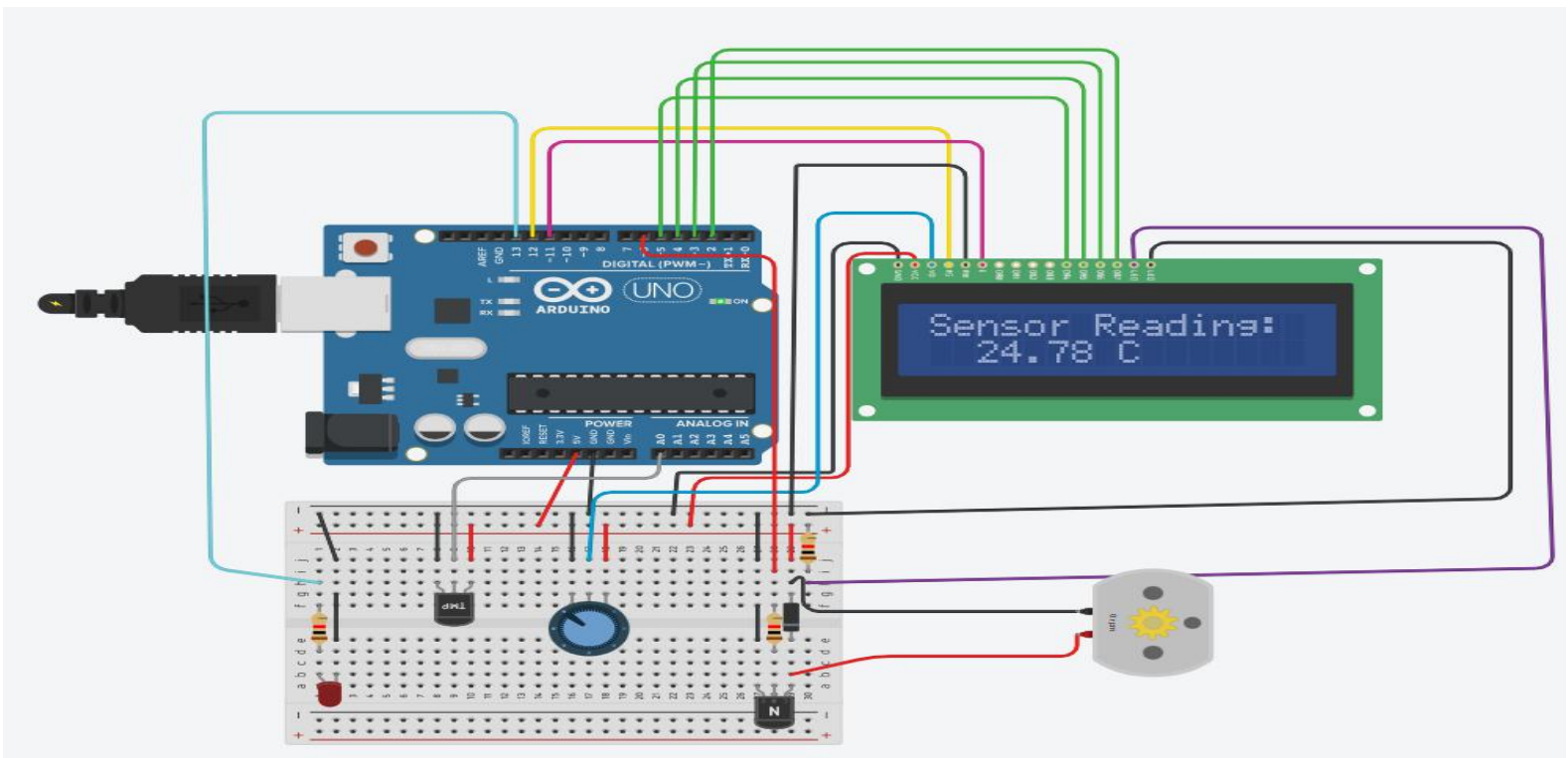
```
44
45     LCD.clear();
46      LCD.print("Now Temp is OK!");
47      LCD.setCursor(0,1);
48      LCD.print("Turn off FAN!");
49      for (int i=255; i>=0; i--){
50        analogWrite(FAN_pin, i);
51      }
52      delay (2000);
53    }
54
55     else if(SensorTemp<MinTemp){
56    LCD.clear();
57      LCD.print("Temp is LOWER!");
58      LCD.setCursor(0,1);
59      LCD.print("Turn on HEATER!");
60
61      digitalWrite(Heater_pin,HIGH);
62
63       delay(3000);
64
65       LCD.clear();
66      LCD.print("Now Temp is OK!");
67      LCD.setCursor(0,1);
68      LCD.print("Turn off HEATER!");
69      delay(1000);
70       digitalWrite(Heater_pin,LOW);
71       LCD.clear();
72  }
73
74    else if (SensorTemp>MinTemp && SensorTemp<MaxTemp){
75    LCD.clear();
76      LCD.print("Temp is normal!");
77      LCD.setCursor(2,1);
78      LCD.print("Turn off all!");
79      delay(1000);
80    LCD.clear();
81    }
82
83    else {
84       LCD.clear();
85      LCD.print("Something went wrong!");
86      LCD.setCursor(0,1);
87      LCD.print("Wrong in circuit!");
88      delay(1000);
89      LCD.clear();
90    }
91
92    delay(1000);
93  }
```

SCHEMATIC VIEW:

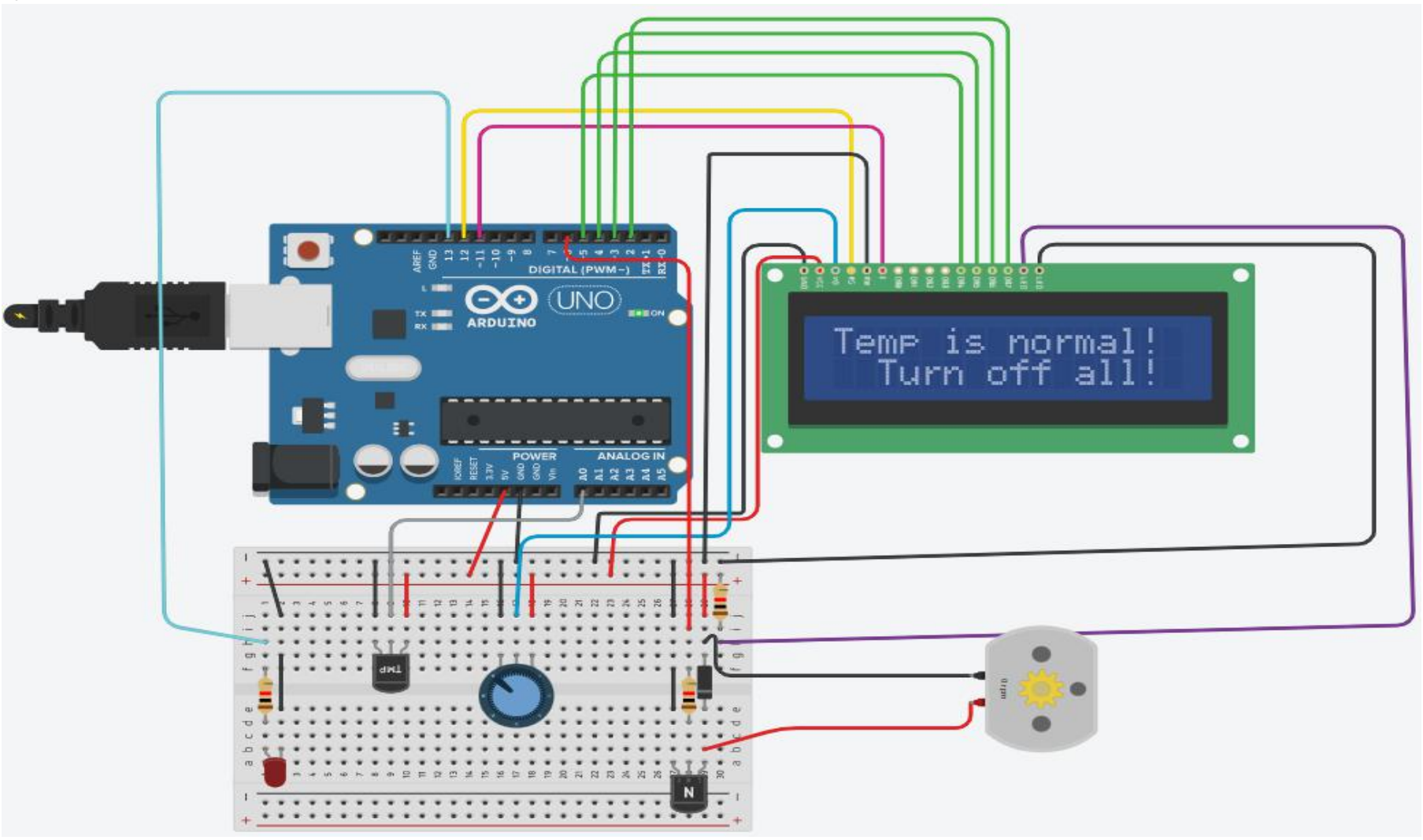


SIMULATION:

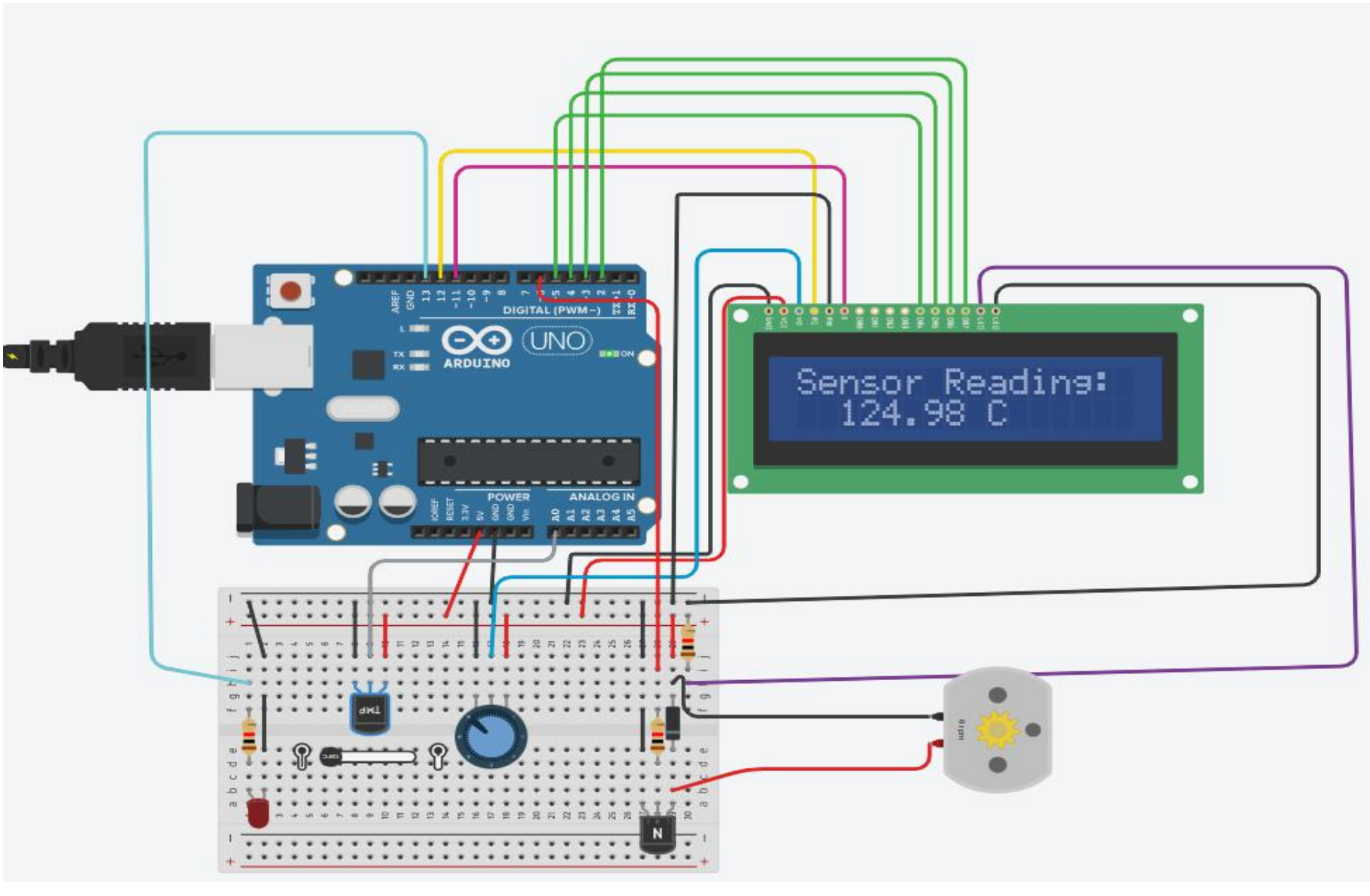
1) SENSOR READING:

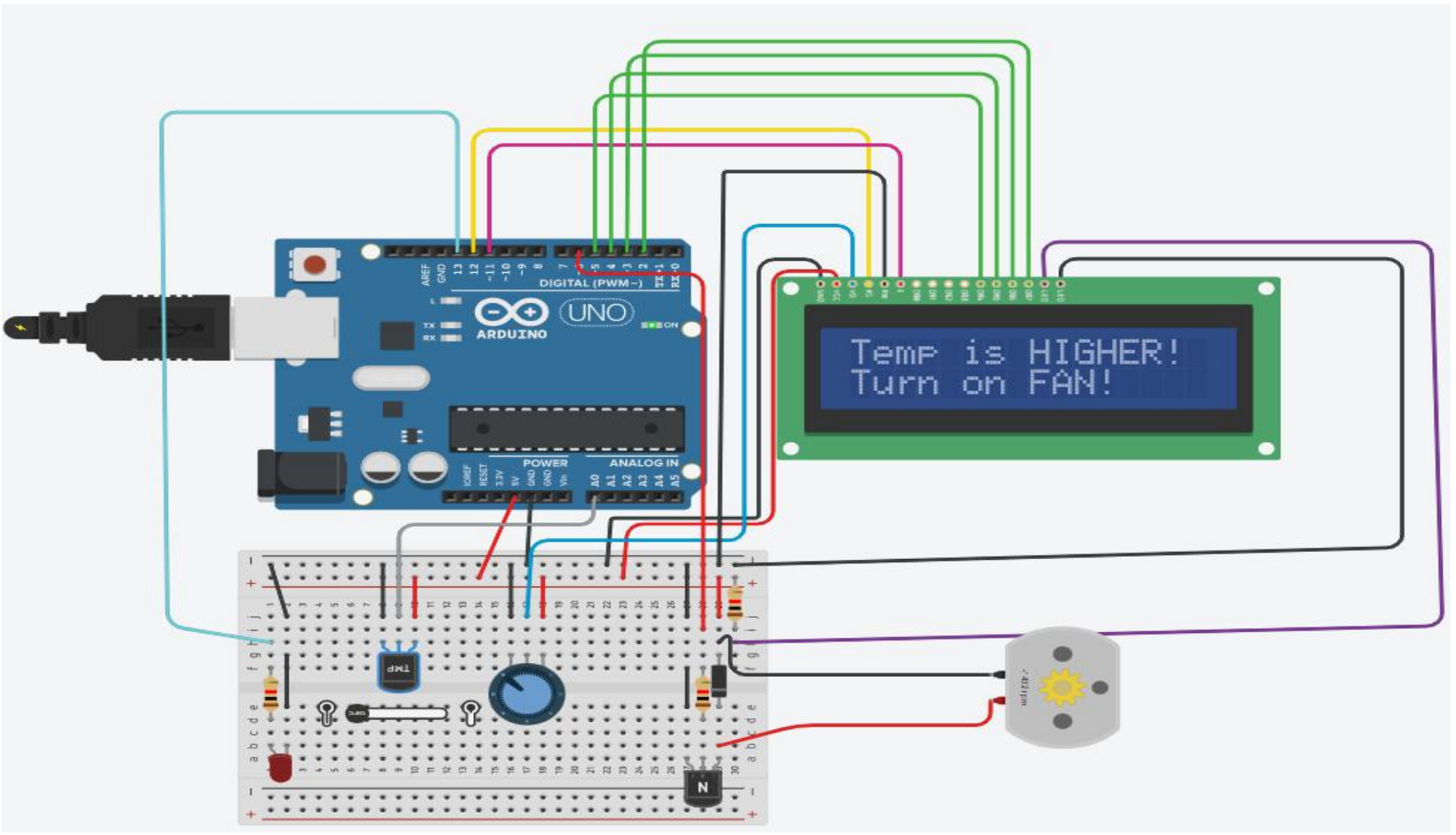


2) WHEN TEMPERATURE IS IN NORMAL RANGE:

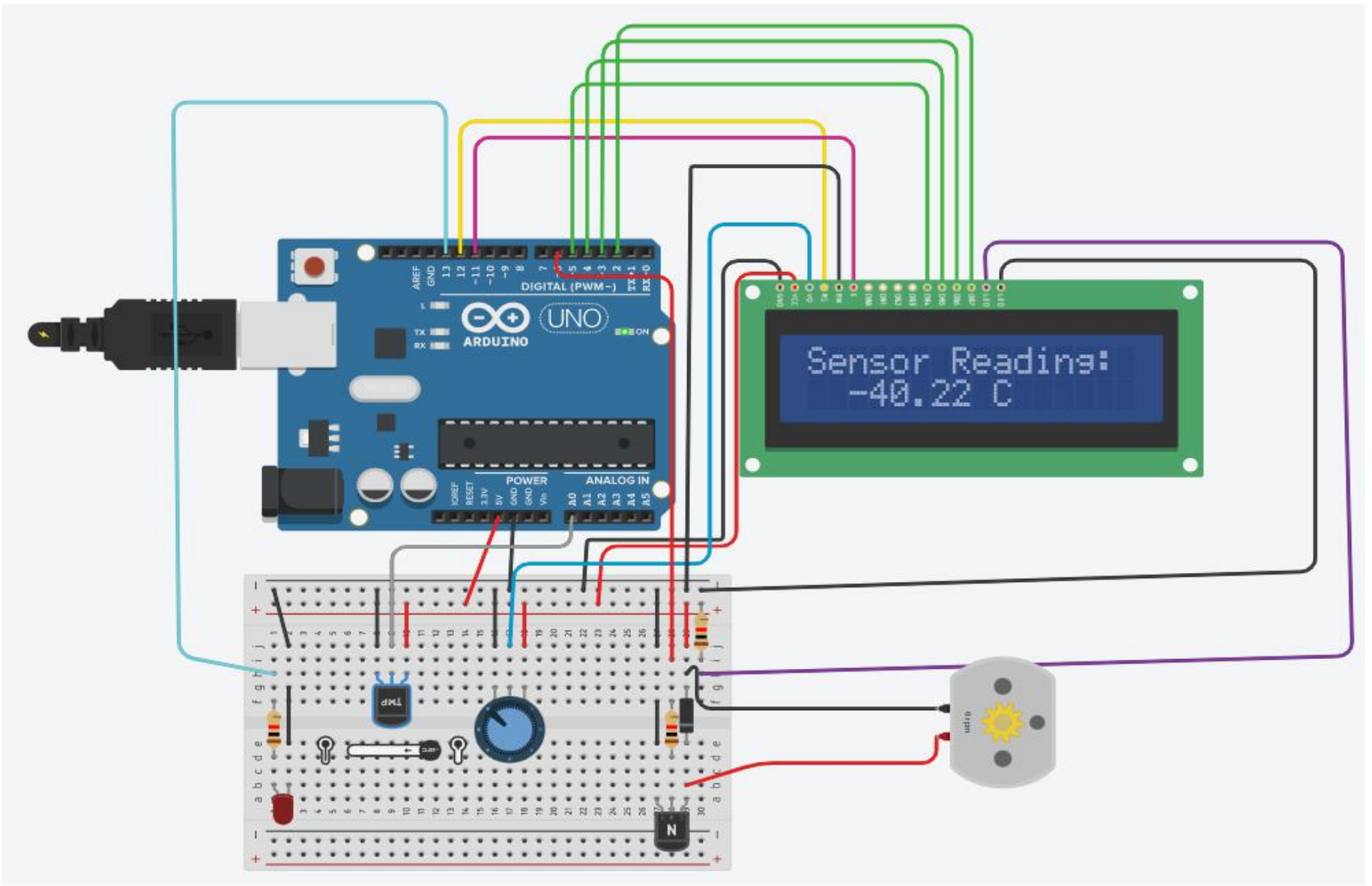


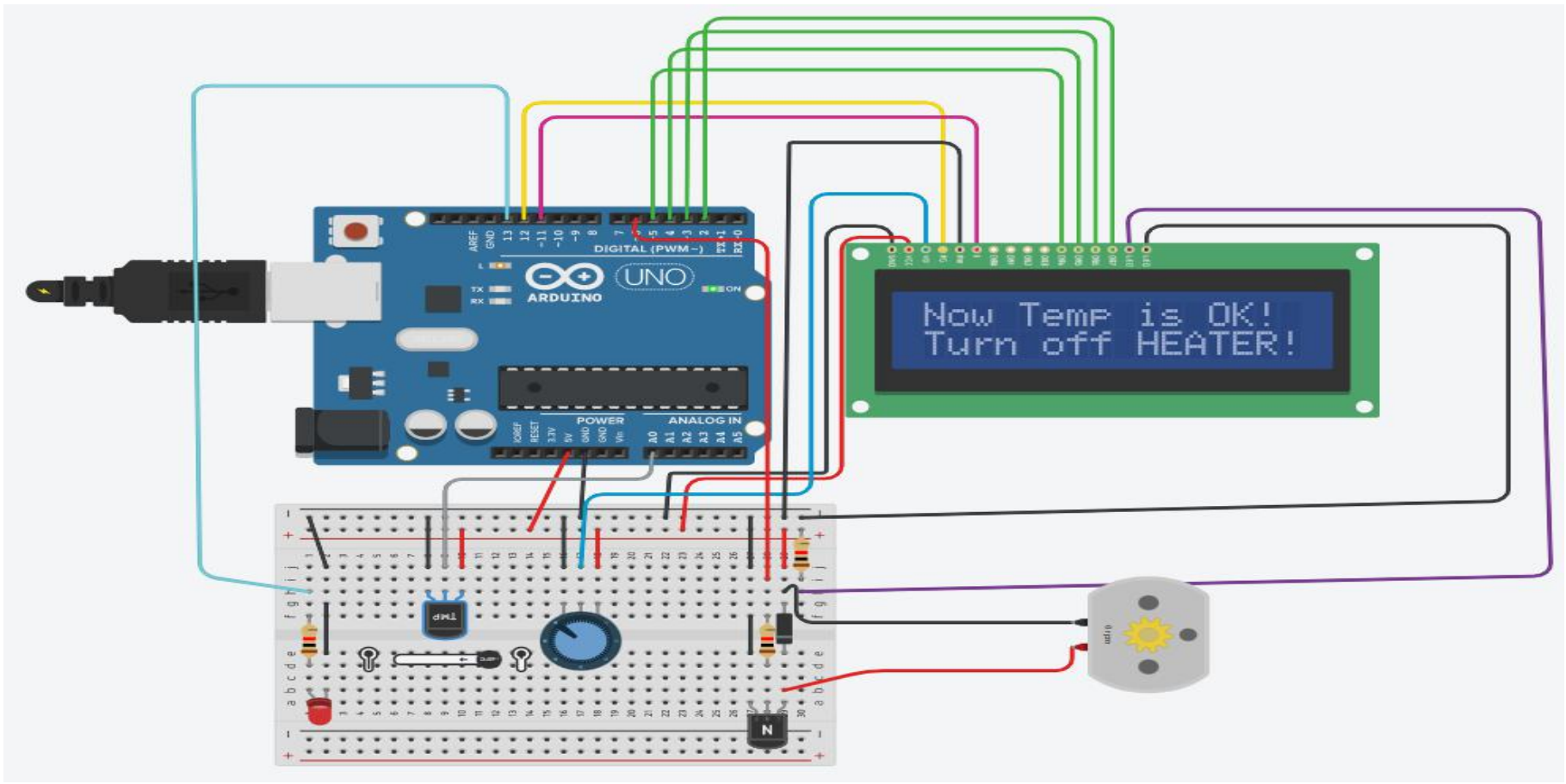
3) WHEN MEASURED TEMPERATURE > NORMAL RANGE:





4) WHEN MEASURED TEMPERATURE < NORMAL RANGE:





CIRCUIT DESIGN LINK:

[Circuit design AUTOMATIC ROOM TEMPERATURE CONTROL SYSTEM | Tinkercad]

APPLICATIONS:

I. This system can serve in many temperature fluctuations cases such as over load and high temperatures which causeS electrical (instrumental) failures
II. Following is a stat which describes accidents related to temperature fluctuations.

III. Industrial Systems- includes all industries that work under high temperatures , especially the Reactors where huge man power in invested unpredictable conditions
IV. Thermostatic controls, in applications which are triggered by temperatures or work on an activity in response to particular temperatures
V. Thermometers, but with some modifications(like 1 wire input)
VI. Agriculture, where temperatures and humidity can affect the quality of the crop. Since humidity can be relatively interpreted with temperatures , this could help in monitoring suitable temperatures for crop growth and signal at unfavourable temperatures could alert farmers so that they could go for safety measures.
VII. In consumer products storage to store products under safe environment

CONCLUSION:

In this paper, an automatic room temperature control system using Arduino and TMP36 sensor has been designed and constructed. The system used an Arduino microcontroller, LCD and TMP36 sensor to control and monitor both the heater and the fan simultaneously. The DC fan is on when the room temperature is higher than the reference temperature, and its speed is controlled based upon the room temperature. When the room temperature is lower than the minimum of the reference temperature, then the heater lamp is turned on while the DC fan triggered off. When the room temperature is within the reference range, all the loads are automatically off.

The main advantages of this system are for its low cost, ease of installation, simplicity, low power consumption, small size, and user-friendly. This project efficiently optimizes energy consumption in a room while keeping the room at a comfortable temperature.

THANK YOU!

K. AADHYA
ka942031@student.nitw.ac.in