**TEMPERATURE MONITORING SYSTEM**

**1. Introduction**

This project involves building a simple temperature monitoring system using an Arduino and an analog temperature sensor (TMP36). The system reads the current temperature and displays it either on an LCD screen or on the Serial Monitor. This task demonstrates basic sensor interfacing and real-time data display in embedded systems.

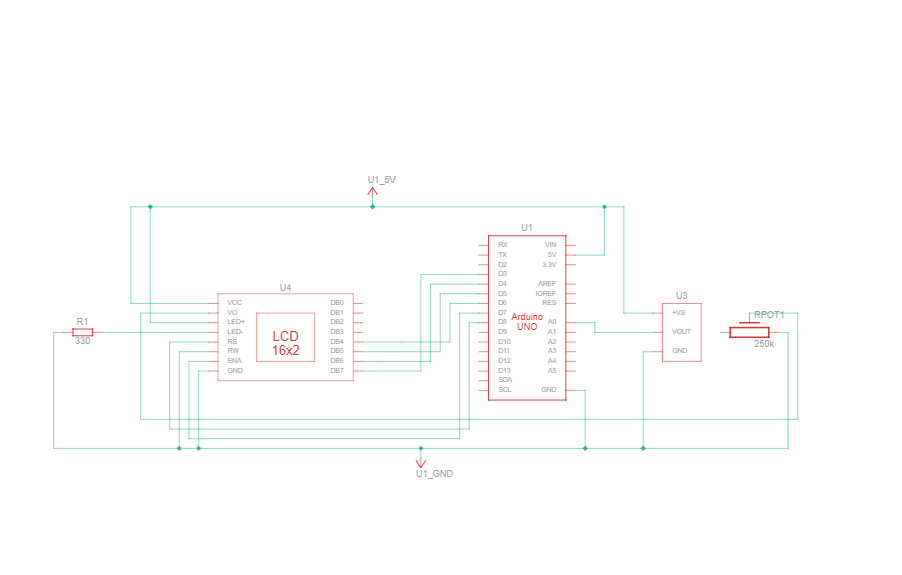
**2. Design and Methodology**

The TMP36 sensor outputs a voltage proportional to temperature. The Arduino reads the analog signal from the sensor, converts it into Celsius, and displays the result on the Serial Monitor. In a real-world setup, this data could also be logged or used for automation

**3.components Required**

1. Arduino Uno – 1
2. TMP36 Temperature Sensor – 1
3. LCD 16x2 (Parallel) – 1
4. Potentiometer (10k ohm) – 1
5. 220-ohm Resistor (optional for backlight) – 1
6. Breadboard – 1
7. Jumper Wires – as needed

**4.Circuit Design**



**5.Code**

#include "LiquidCrystal.h"

LiquidCrystal lcd(8,7,6,5,4,3);

int sensorPin = 0;

void setup()

{

Serial.begin(9600);

lcd.begin(16,2);

}

void loop()

int reading = analogRead(sensorPin);

float voltage = reading \* 4.68;

voltage /= 1024.0;

float temperatureC = (voltage - 0.5) \* 100;

Serial.print(temperatureC);

Serial.println(" degrees C");

lcd.setCursor(0,0);

lcd.print("Temperature Value ");

lcd.setCursor(0,1);

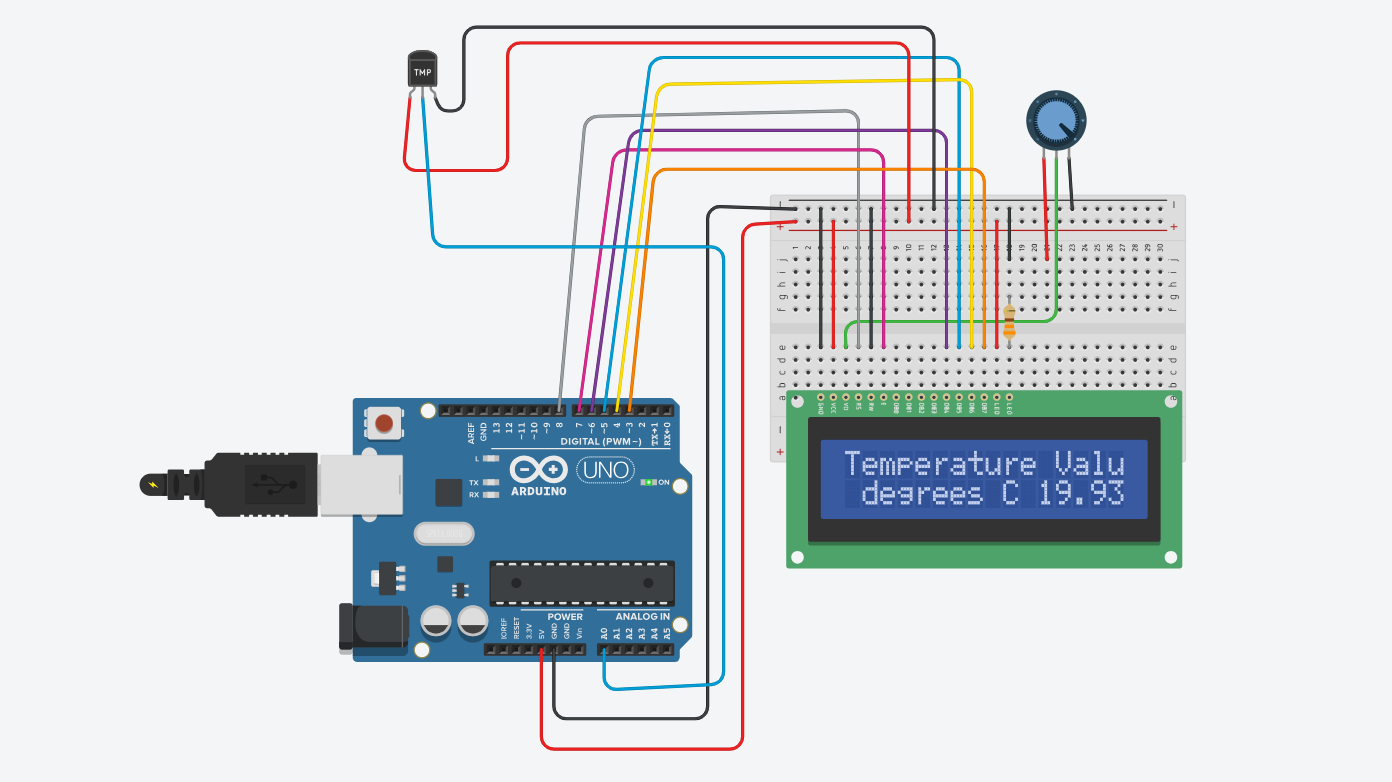
lcd.print(" degrees C");

lcd.setCursor(11,1);

lcd.print(temperatureC);

delay(100);}

**6. Output Demonstration**

****

**7.Conlusion**

The temperature monitoring system successfully reads ambient temperature using the TMP36 sensor and displays it on the Serial Monitor. This task demonstrates the basics of analog sensor interfacing, data conversion, and real-time output — forming a foundation for more advanced monitoring systems.