***PRACTICAL: 3***

**AIM:** Program using Temperature Sensors.

**Objective**:

This project is to measure temperature using the **TMP36 temperature sensor** and an **Arduino Uno**, then process and display the data for further analysis.

**Components Required:**

* **Arduino Uno** (Microcontroller Board)
* **TMP36 Temperature Sensor**
* **Jumper Wires** (Male-to-Male or Male-to-Female, as needed)

**CONNECTION:**

Step: 1: Identify TMP36 Pins:

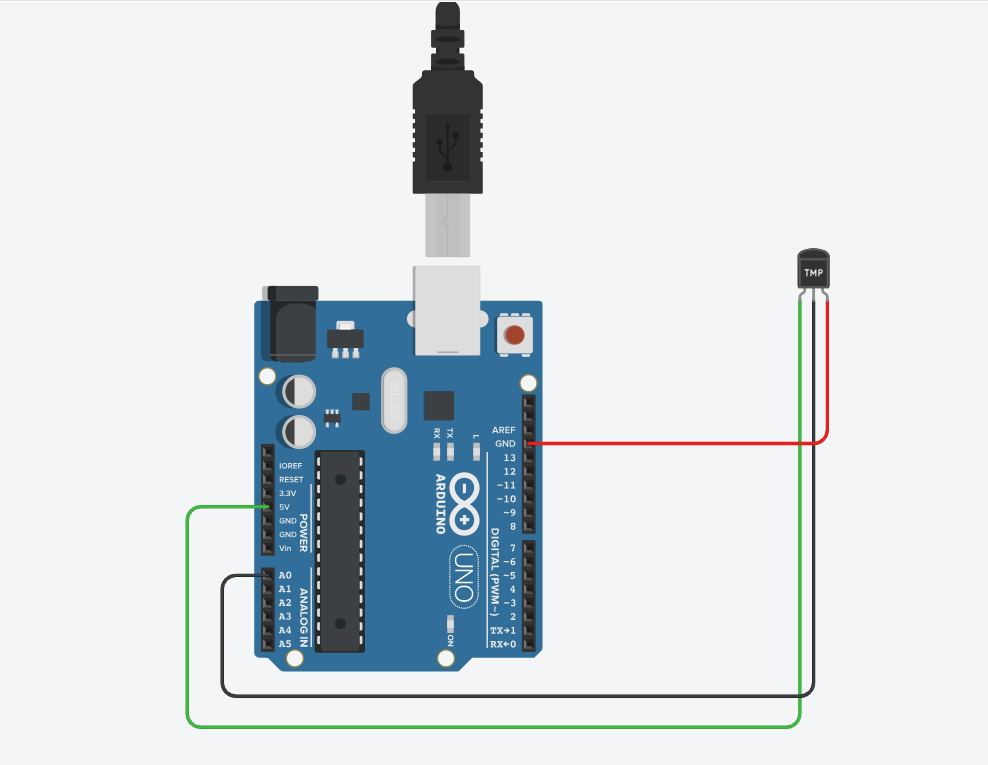
* The TMP36 sensor has three pins:

1. VCC (Left pin): Power supply (3.3V or 5V)
2. VOUT (Middle pin): Output voltage (temperature data)
3. GND (Right pin): Ground

Step: 2: Connect the TMP36 to the Arduino Uno:

* Connect the VCC (left pin) of the TMP36 to the 5V pin on the Arduino.
* Connect the GND (right pin) of the TMP36 to the GND pin on the Arduino.
* Connect the VOUT (middle pin) of the TMP36 to A0 (Analog Pin 0) on the Arduino.

***DIAGRAM***

******

***CODE***

const int sensorPin = A0;

void setup() {

Serial.begin(9600);

}

void loop() {

int sensorValue = analogRead(sensorPin);

float voltage = sensorValue \* (5.0 / 1023.0);

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

Serial.print("Temperature: ");

Serial.print(temperature);

Serial.println(" °C");

delay(1000);

}

***CODE EXPLANATION***

1. Defines the sensor pin:

* sensorPin = A0 → The TMP36 temperature sensor is connected to Analog Pin A0.

1. Setup Function (setup()):

* Initializes serial communication at 9600 baud rate to display temperature readings on the Serial Monitor.

1. Loop Function (loop()):

* Reads the analog value from the TMP36 sensor.
* Converts the analog reading into voltage using: voltage=sensorValue×(5.01023.0)\text{voltage} = \text{sensorValue} \times \left( \frac{5.0}{1023.0} \right)voltage=sensorValue×(1023.05.0​)
* Converts the voltage into temperature in Celsius using: temperature=(voltage−0.5)×100\text{temperature} = (\text{voltage} - 0.5) \times 100temperature=(voltage−0.5)×100
* Prints the temperature value to the Serial Monitor.
* Waits 1 second (1000ms) before taking the next reading.
* The Output of this Shown on Serial Monitor Like this

