

Lab Assignment - 1

Problem Statement - Write a program for sending alert messages to the user for controlling and interacting with your environment.

Component Required :

| Quantity | Component |
|----------|----------------------------|
| 1 | Arduino Uno R3 |
| 1 | Temperature Sensor (TMP36) |
| 1 | LED |
| 1 | Piezo |
| 1 | 220 Ω Resistor |

Steps To Perform :

Step 1: Power Up the Breadboard

| Component Pin | Connect To | Note |
|-----------------|------------------------------|-------------------------------------|
| Arduino 5V Pin | Breadboard Positive (+) Rail | Supplies 5 Volts to the breadboard. |
| Arduino GND Pin | Breadboard Negative (-) Rail | Creates a common ground connection. |

Step 2: Connect the Temperature Sensor

| Component Pin | Connect To | Note |
|--------------------------|------------------------------|--|
| Power Pin (Left) | Breadboard Positive (+) Rail | Powers the sensor. |
| Signal/Vout Pin (Middle) | Arduino Analog Pin A0 | Sends temperature data to the Arduino. |
| Ground Pin (Right) | Breadboard Negative (-) Rail | Completes the sensor's circuit. |

Step 3: Connect the LED and Resistor

| Component Pin | Connect To | Note |
|---------------------------|------------------------------|--|
| LED Anode (Longer Leg) | One leg of the 220Ω Resistor | This is the positive side of the LED. |
| Resistor (Other Leg) | Arduino Digital Pin 7 | The resistor protects the LED from too much current. |
| LED Cathode (Shorter Leg) | Breadboard Negative (-) Rail | This is the negative side of the LED. |

Step 4: Connect the Buzzer

| Component Pin | Connect To | Note |
|-------------------------|------------------------------|--------------------------------------|
| Buzzer Positive (+) Pin | Arduino Digital Pin 8 | Receives the signal to make a sound. |
| Buzzer Negative (-) Pin | Breadboard Negative (-) Rail | Completes the buzzer's circuit. |

Code :

```
const int sensorPin = A0; // Vout of the temperature sensor
const int ledPin = 7; // LED anode (+) directly (Tinkercad OK)
const int buzzerPin = 8; // Optional buzzer +
const float limitC = 30.0; // Alert threshold in °C

void setup() {
    Serial.begin(9600); // Open Serial Monitor at 9600 baud
    pinMode(ledPin, OUTPUT);
    pinMode(buzzerPin, OUTPUT);
}

void loop() {
    // 1. Read analog value and convert to volts
    int raw = analogRead(sensorPin);
    float voltage = (raw * 5.0) / 1023.0;

    // 2. Convert voltage to temperature (LM35 = 10 mV per °C)
    float tempC = voltage * 100.0;
```

```

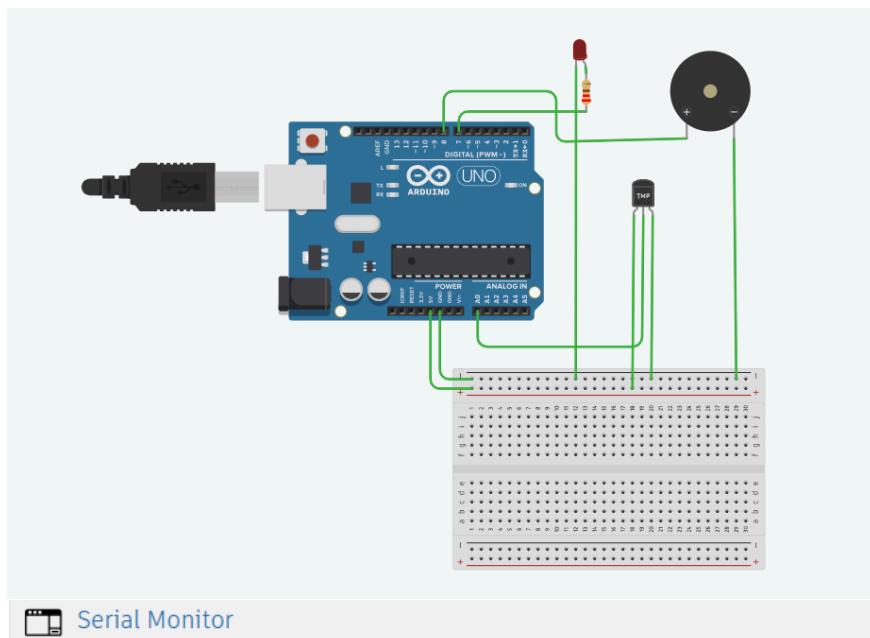
// 3. Display temperature in °C
Serial.print("Temperature: ");
Serial.print(tempC);
Serial.println(" C");

// 4. Alert if temperature is above limit
if (tempC > limitC) {
  Serial.println("ALERT: High Temperature!");
  digitalWrite(ledPin, HIGH);
  digitalWrite(buzzerPin, HIGH);
} else {
  digitalWrite(ledPin, LOW);
  digitalWrite(buzzerPin, LOW);
}

delay(1000); // Update once per second
}

```

Output :



Serial Monitor

```

Temperature: 74.78 C
ALERT: High Temperature!
Temperature: 48.88 C
ALERT: High Temperature!
Temperature: 17.11 C
Temperature: 24.93 C
Temperature: 24.93 C
Temperature: 24.93 C
Temperature: 24.93 C

```