“Lab Report-06”

Course Code:	Course Title:
CSE234	Embedded Systems and IoT Lab

Lab Report Details	
Experiment No	: 06
Experiment Name	: Experiment on Interfacing a Temperature and Humidity Sensor with Arduino and Displaying Data.
Lab Perform Date	: 04/08/2025
Report Submission Date	: 11/08/2025

Submitted To:	Submitted By:
Mr. Avizit Nandi Lecturer Department of **‘CSE’** Daffodil International University.	Name : Md. Sajjad Mojumder Anik SID : 221-15-5640 Section : ‘61_C1’ Department of ‘**CSE**’

Daffodil International University

Title: Experiment on Interfacing a Temperature and Humidity Sensor with Arduino and Displaying Data.

Objective:

1. Interface a temperature and humidity sensor (TMP36 temperature sensor used in this experiment) with an Arduino board.
2. Read real-time sensor data using Arduino.
3. Display sensor data through the Serial Monitor in the Arduino IDE.
4. Interpret sensor output values for practical applications.
5. Enhance practical skills in hardware interfacing and embedded C programming for microcontroller-based systems.

Introduction

Sensors play a key role in collecting environmental data for embedded systems and IoT applications. Among them, the TMP36 temperature sensor is widely used due to its simple operation and direct analog voltage output proportional to temperature. In this experiment, a potentiometer will be employed to simulate humidity, providing a convenient way to understand humidity sensing concepts. An Arduino microcontroller will be used to acquire analog data from both the TMP36 and the potentiometer, process it, and display the results via the Serial Monitor. This arrangement offers a hands-on approach to interfacing sensors, performing analog-to-digital conversion, and interpreting sensor readings using Arduino.

Theory

In this setup, the TMP36 serves as the temperature sensor, while the potentiometer functions as a simulated humidity source. The Arduino's Analog-to-Digital Converter (ADC) processes the incoming analog voltages.

- **TMP36 Temperature Sensor**
  The TMP36 is a three-terminal device that measures ambient temperature and outputs a voltage linearly related to the temperature in Celsius.
- **Potentiometer as a Humidity Simulator**
  The potentiometer acts as a variable resistor, producing an adjustable voltage between 0 V and 5 V. By rotating the knob, different voltage levels are generated, which the Arduino maps to a humidity range of 0% to 100%. This serves as a simplified model for real humidity measurements.
- **Arduino ADC**
  The Arduino's analog input pins use a 10-bit ADC, enabling the conversion of voltages between 0 V and 5 V into digital values ranging from 0 to 1023. These digital readings

can then be processed and scaled to represent temperature or humidity values.

Equipment Required:

1. Arduino UNO
2. Breadboard
3. TMP36 temperature sensor
4. Jumper wires
5. Potentiometer
6. Arduino IDE (installed on PC)

Circuit Diagram:

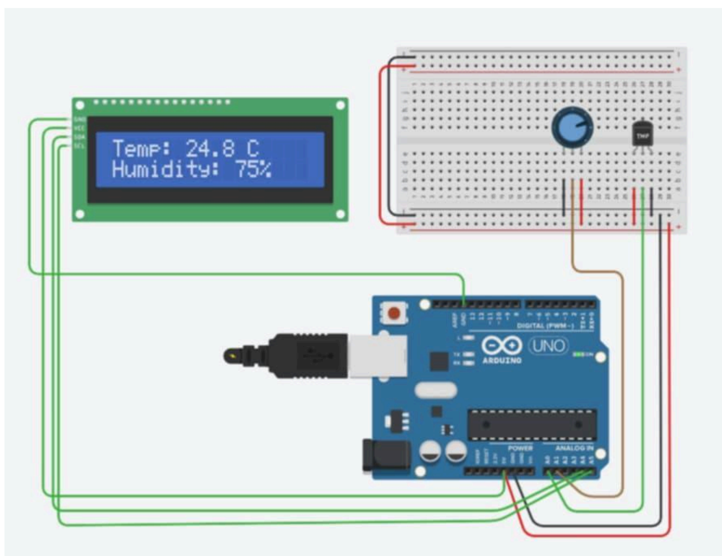

Fig. 1.1: Interfacing a Temperature and Humidity Sensor with Arduino

Arduino code: Arduino code for Temperature and Humidity Sensor

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
const int tempPin = A0;
const int humidPin = A1;
int rawTemp = 0;
```

```
double voltage = 0;
double tempC = 0;
double tempF = 0;
int humidityValue = 0;

void setup() {
Serial.begin(9600);
pinMode(tempPin, INPUT);
pinMode(humidPin, INPUT);
lcd.init();
lcd.backlight();
}
void loop() {
rawTemp = analogRead(tempPin);
voltage = (rawTemp / 1023.0) * 5000;
tempC = (voltage - 500) * 0.1;
tempF = (tempC * 1.8) + 32;
Serial.print("Raw Value (Temp) = ");
Serial.print(rawTemp);
Serial.print("\tVoltage (mV) = ");
Serial.print(voltage, 0);
Serial.print("\tTemperature (C) = ");
Serial.print(tempC, 1);
Serial.print("\tTemperature (F) = ");
Serial.println(tempF, 1);
humidityValue = analogRead(humidPin);
int humidityPercent = map(humidityValue, 0, 1023, 0, 100);
Serial.print("Raw Value (Humidity) = ");
Serial.print(humidityValue);
Serial.print("\tHumidity = ");
Serial.print(humidityPercent);
Serial.println("%");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Temp: ");
lcd.print(tempC, 1);
lcd.print(" C");
lcd.setCursor(0, 1);
lcd.print("Humidity: ");
lcd.print(humidityPercent);
lcd.print("%");
```

```
delay(1000);
}
```

Discussion:

1. The TMP36 temperature sensor was successfully interfaced with the Arduino UNO.
2. The Arduino's built-in ADC converted the sensor's analog voltage into digital values for processing.
3. Real-time temperature readings were displayed on the Serial Monitor of the Arduino IDE.
4. The sensor showed quick and accurate responses to changes in temperature, such as heating or touching.
5. The experiment demonstrated the simplicity and effectiveness of sensor microcontroller interfacing.
6. The setup can be extended for applications in weather monitoring, automation, and industrial control.

Conclusion:

1. Successfully interfaced the TMP36 temperature sensor with an Arduino UNO.
2. Wrote and executed embedded C code to read analog sensor values using analogRead().
3. Displayed live temperature readings in Celsius on the Arduino IDE Serial Monitor.
4. Learned the conversion process from raw ADC values to voltage and then to temperature.
5. Demonstrated practical sensor-based data acquisition for real-time monitoring.