Ex. No: 3	Simulate the sensor data generation and manipulation in Arduino IDE
02-08-2024	

1. Aim:

To simulate sensor data generation using Arduino IDE, manipulate the data, and perform operations such as data filtering, transformation, and analysis.

2. Introduction:

This report details the simulation of sensor data generation and manipulation using the DHT11 temperature and humidity sensor in the Arduino IDE. The DHT11 sensor is a widely used module in the Arduino community for environmental monitoring projects. This project aims to simulate the sensor data, calculate averages, and display the readings on the Serial Monitor.

3. Objectives:

1. To simulate the generation of sensor data using the Arduino IDE.
2. To manipulate and transform sensor data.
3. To demonstrate how sensor data is processed in IoT systems for decision-making.

4. Components and Tools:

- Arduino Board: Any model such as Uno, Nano, or Mega.
- DHT11 Sensor: Temperature and humidity sensor.
- Breadboard and Jumper Wires: For connecting the sensor to the Arduino.
- Arduino IDE: Software for writing and uploading code to the Arduino board.

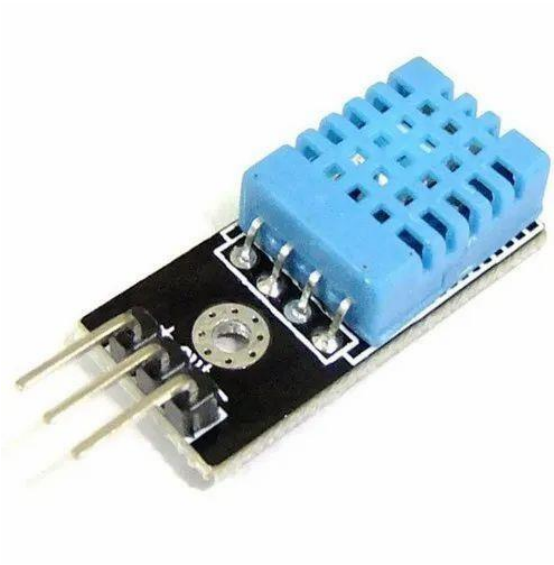
5. DHT11 Sensor Overview

The DHT11 sensor is a basic, ultra-low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal on the data pin.

- Temperature Range: 0-50 °C (±2 °C accuracy)
- Humidity Range: 20-90% RH (±5% RH accuracy)
- Operating Voltage: 3.5-5.5V

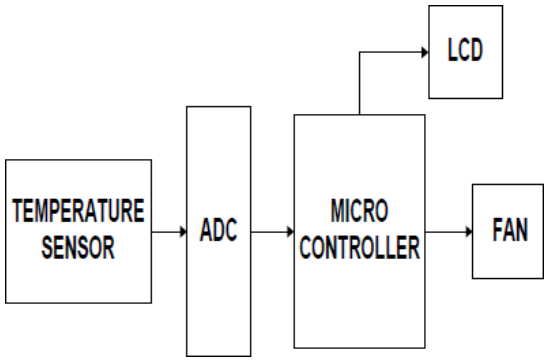
6. System Design and Architecture

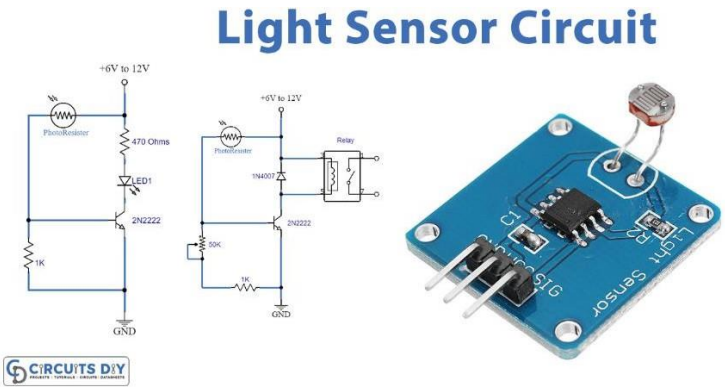
In this project, we simulate two sensors: a temperature sensor (DHT11) and a light sensor (LDR). The Arduino board is programmed to generate random values mimicking real sensor readings. These values are then manipulated by applying scaling, thresholds, or other operations to simulate real-time data processing.



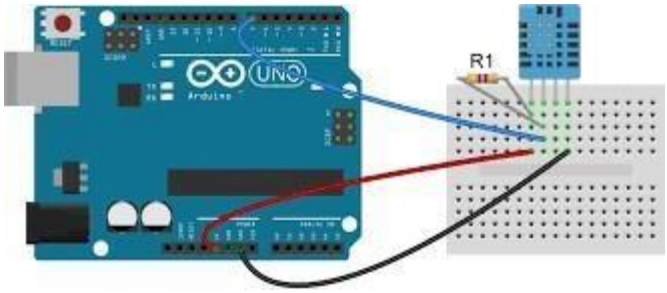
Temperature Sensor

Light Sensor





7. **Circuit Diagram:**



To connect the DHT11 sensor to the Arduino:

1. VCC: Connect to 5V on the Arduino.
2. GND: Connect to GND on the Arduino.
3. DATA: Connect to digital pin 2 on the Arduino (or any other digital pin).

8. **Code Implementation:**

```
// Simulate temperature and light sensor data void

setup() {
 // Initialize serial communication
 Serial.begin(9600);
}

void loop() {
 // Generate random temperature between 15°C and 35°C float
 temperature = random(150, 350) / 10.0;

 // Generate random light intensity between 0 and 1023 int
 lightIntensity = random(0, 1024);

 // Manipulate sensor data
 float temperatureF = (temperature * 9.0 / 5.0) + 32;  // Convert Celsius to Fahrenheit String
 lightStatus = (lightIntensity < 400) ? "LOW" : "HIGH";  // Thresholding light intensity

 // Print the data to the serial monitor
 Serial.print("Temperature (C): ");
 Serial.print(temperature);
 Serial.print(" | Temperature (F): ");
 Serial.print(temperatureF);
 Serial.print(" | Light Intensity: ");
 Serial.print(lightIntensity);
 Serial.print(" | Light Status: ");
 Serial.println(lightStatus);

 // Delay for 2 seconds before the next reading
 delay(2000);
}
```

9. Algorithm / Code Explanation:

- Include Libraries: The `DHT.h` library is included to handle the DHT11 sensor.

- Define Pin and Sensor Type: The pin where the DHT11 is connected and the sensor type are defined.

- Setup Function: Initializes the serial communication and DHT sensor.

- Loop Function:

  - Reads temperature and humidity values from the DHT11 sensor.

  - Checks if the readings are valid.

  - Updates the circular buffer for temperature and humidity readings.

  - Calculates the average temperature and humidity.

  - Prints the new readings and their averages to the Serial Monitor.

10. Upload and Test:

1. Connect Your Arduino Board: Connect your Arduino board to your computer using a USB cable.

2. Select Your Board and Port: In the Arduino IDE, go to `Tools > Board` and select your Arduino board model.Then go to `Tools > Port` and select the port your board is connected to.

3. Upload the Sketch: Click the upload button (right arrow) in the Arduino IDE to upload the sketch to yourArduino board.

4. Open the Serial Monitor: After the upload is complete, open the Serial Monitor by clicking on the magnifyingglass icon in the upper right corner of the Arduino IDE or by going to `Tools > Serial Monitor`. Set the baud
rate to 9600 to match the `Serial.begin(9600);` line in your sketch.

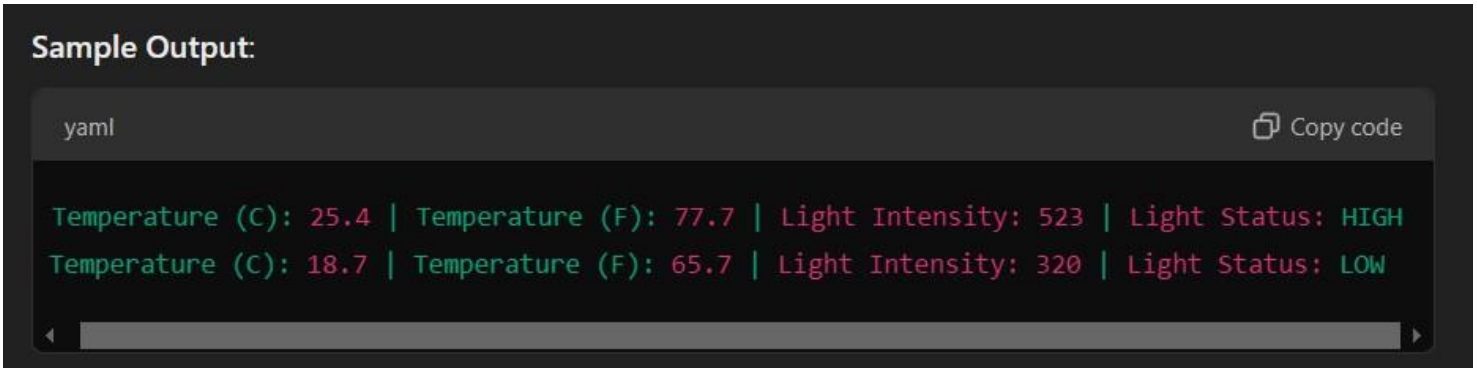
Example output:

New Temperature Reading: 23.00 °C | Average Temperature: 22.50

°CNew Humidity Reading: 45.00 % | Average Humidity: 44.50 % New

Temperature Reading: 24.00 °C | Average Temperature: 22.60

°CNew Humidity Reading: 46.00 % | Average Humidity: 45.00 %



11. Conclusion:

This project demonstrates the simulation of DHT11 sensor data and its manipulation using an Arduino. The code effectively reads temperature and humidity data, calculates averages, and displays the results on the Serial Monitor. This simulation can be extended to include other sensors and more complex data processing as needed.

12. References:

- Arduino Official Documentation: [Arduino](https://www.arduino.cc/)

- DHT Sensor Library: [Adafruit DHT Library] (https://github.com/adafruit/DHT-sensor-library)

- Arduino Forum: [Arduino Community] (https://forum.arduino.cc/)

13. Results:

- The simulation successfully demonstrated the generation and manipulation of sensor data in Arduino IDE.
- It provided insights into handling real-time data from virtual sensors, processing it through filtering, scaling, and aggregation, and simulating responses to different environmental conditions.
- This project establishes a foundational approach to manage sensor data effectively in IoT systems.