TABLE OF CONTENTS

Abstract

This project presents a system to measure and display temperature and humidity using an Arduino Uno microcontroller and a DHT11 sensor. The implementation involves interfacing the DHT11 sensor with the Arduino Uno, which collects data at specified intervals. The Arduino then processes the data and displays the temperature and humidity readings on an LCD screen. The system is programmed using the Arduino Integrated Development Environment (IDE). In conclusion, this project demonstrates the practical application of Arduino and DHT11 sensor, and a 16x2 LCD display technology for real-time temperature and humidity monitoring

Keywords: Arduino Uno, DHT11, Arduino Integrated Development Environment (IDE).

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 Background of the Study

In recent years, the importance of environmental monitoring has grown significantly due to climate change, urbanization, and the demand for real-time data in both industrial and domestic settings. Temperature and humidity are two fundamental environmental parameters that affect human comfort, agricultural productivity, and industrial processes. Traditional monitoring systems are often expensive or complex, making them inaccessible for basic educational or DIY applications.

The Arduino Uno, a widely used open-source microcontroller platform, offers an affordable and versatile solution for building small-scale environmental monitoring systems. Combined with a DHT11 sensor — a low-cost digital sensor capable of measuring temperature and humidity — it is possible to create an efficient and compact system that provides real-time environmental data.

This project explores the design and implementation of a basic temperature and humidity monitoring system using the Arduino Uno and DHT11 sensor. The system displays the data on an LCD screen and can serve as a foundation for more complex Internet of Things (IoT) applications.

1.2 Problem Statement

Accurate environmental data collection is essential in numerous applications, but existing systems may be too expensive, difficult to install, or require proprietary technology. There is a need for a simple, low-cost solution that can be used for educational purposes, small-scale monitoring, or as a base for future development.

This project aims to address the problem by creating a reliable and cost-effective temperature and humidity sensor system that is easy to build, operate, and customize.

1.3 Objectives of the Project

The main objectives of this project are:

1.) To design a simple system using Arduino Uno and DHT11 to measure temperature and humidity.

2.) To interface the sensor with an LCD module for real-time display of data.

3.) To develop a functional and efficient code for acquisition and display.

4.) To test and evaluate the accuracy and responsiveness of the system.

1.4 Scope and Limitations

This project focuses on a basic indoor temperature and humidity monitoring system. The scope includes hardware setup, software development, testing, and result analysis. The system:

- Uses Arduino Uno as the microcontroller.

Employs a DHT11 sensor for environmental data.

Displays data on a basic LCD (16x2 or similar).

- Operates in real time but does not store historical data or connect to the internet.

Limitations include:

- The DHT11 sensor has moderate accuracy and limited range.
- The system is not weatherproof or intended for harsh outdoor conditions.
- Data is not logged or transmitted wirelessly in this version.

1.5 Significance of the Study

This project demonstrates how low-cost and widely available tools can be used to develop functional environmental monitoring systems. It is especially valuable for:

- Students learning about embedded systems and sensor integration.

Hobbyists interested in home automation or environmental monitoring.

- Researchers developing scalable IoT systems for real-time data gathering.

By providing a foundation for more complex projects, this study contributes to the growing field of accessible and sustainable environmental technology.

1.6 Organization of the Thesis

This thesis is organized into five chapters:

Chapter 2: Literature Review – discusses existing research and technologies related to temperature and humidity .

Chapter 3: System Design and Methodology – explains the hardware and software setup, circuit diagrams, and logic of the project.

Chapter 4: Implementation and Results – details the

actual implementation, testing, and analysis of the outcomes

Chapter 5: Conclusion and Future Work – summarizes the findings, addresses limitation.

Chapter 2

Literature Review

2.0 Introduction

This chapter provides an overview of the background literature and related work in the field of environmental monitoring systems, particularly focusing on temperature and humidity measurement. It discusses the technology and components commonly used in similar systems, with special attention to the DHT11 sensor and Arduino Uno platform. The aim is to contextualize the current project within existing research and development.

2.1 Overview of Temperature and Humidity Monitoring

Monitoring temperature and humidity is critical in numerous applications, including meteorology, agriculture, HVAC (Heating, Ventilation, and Air Conditioning), and industrial automation. Accurate and timely data allows for better environmental control, comfort, and safety.

Traditionally, such systems required specialized equipment and high-cost sensors. However, with the advancement of microcontrollers and digital sensors, cost-effective and compact solutions are now widely accessible.

2.2 Existing Works and Related Projects

Numerous studies and projects have utilized Arduino and DHT sensors for real-time monitoring:

- A study by R. Kiran et al. (2019) developed a home environment monitoring system using Arduino Uno, DHT11, and an LCD. Their system successfully provided real-time data but lacked data logging capabilities.
- An IoT-based smart agriculture project by P. Sharma et al. (2020) used DHT22 sensors with a Wi-Fi module to transmit data to a mobile application, showcasing the scalability of Arduino-based sensor networks.
- Educational kits like the Grove Beginner Kit for
- Arduino include built-in DHT sensors, encouraging

hands-on learning in schools and universities.

These studies reinforce the relevance and effectiveness of simple Arduino-DHT configurations for temperature and humidity monitoring.

2.3 Comparisons of Temperature and Humidity Sensors (DHT11 vs Others)

Sensors used in environmental monitoring must be reliable, durable, and energy - efficient. Common sensors for temperature include thermistors, RTDs (Resistance Temperature Detectors), and digital sensors like the DS18B20. For humidity, capacitive and resistive types are prevalent.

The DHT11 sensor, used in this project, is a combined temperature and humidity sensor that provides digital output. Although it is less accurate than more expensive alternatives (such as the DHT22 or SHT31), it is widely used in educational and hobbyist applications due to its simplicity and affordability.

Sensor	Temperature Range	Humidity Range	Accuracy	Cost
DHT11	0–50 °C	20–90% RH	±2 °C / ±5%	Low

DHT22	−40–80 °C	0–100% RH	±0.5 °C / ±2%	Moderate
SHT31	−40–125 °C	0–100% RH	±0.3 °C / ±2%	High

2.4 Use of Arduino in Environment Monitoring

Arduino microcontrollers have become a foundational tool in the development of modern environmental monitoring systems. Their open-source nature, affordability, and flexibility make them ideal for both educational and professional applications. Environmental monitoring involves the measurement of physical parameters such as temperature, humidity, air quality, light intensity, and soil moisture, all of which can be effectively captured and processed using Arduino-based systems.

One of the key strengths of the Arduino platform is its ability to interface easily with a wide range of environmental sensors. Through digital and analog input/output pins, Arduino boards can collect data from sensors and transmit or display it using various communication modules (e.g., Wi-Fi, Bluetooth) or displays (e.g., LCD, OLED).

Common applications of Arduino in environmental monitoring include:

- Indoor air quality monitoring using gas sensors (e.g., MQ-135, MQ-2).
- Weather stations that measure temperature, humidity, atmospheric pressure, and rainfall.
- IoT-based greenhouse systems for automated climate control.

2.6 Summary of Literature Review

The literature confirms that low-cost microcontrollers like Arduino Uno, combined with digital sensors like the DHT11, can be effectively used in basic environmental monitoring systems. Although the DHT11 has limitations in terms of range and precision, its affordability and ease of use make it an excellent choice for learning and prototyping purposes.

This review establishes a strong foundation for the current project, which seeks to implement a similar system with local data display and potential for future enhancements.

Chapter 3
System Design and Methodology

3.1 Overview of the System Architecture

The system is designed to measure ambient temperature and humidity using the DHT11 digital sensor and display the results on a 16x2 LCD screen. At the core of the system is the Arduino Uno microcontroller, which collects data from the sensor, processes it, and updates the display in real time. The project is a standalone embedded system that demonstrates the integration of sensor technology, microcontroller programming, and user interface development.

The system architecture includes:

- Input: DHT11 Temperature and Humidity Sensor
- Processing: Arduino Uno
- Output: LCD Display (16x2 alphanumeric screen)
- Power Supply: USB

3.2 Description of Components

3.2.1 Arduino Uno

The Arduino Uno is a microcontroller board based on the ATmega328P. It features 14 digital input/output pins, 6 analog inputs, a 16 MHz quartz crystal, and a USB connection for programming. It serves as the control unit, reading data from the DHT11 sensor and outputting the information to the LCD display.

Features:

1.) **User-Friendly**: The Arduino Uno is recognized for its ease of use, and it is perfect for beginners starting with electronics and coding due to its comprehensive documentation and large community support.

2.) **Versatile I/O Capabilities**: It features 14 digital input/output pins and 6 analog input pins, allowing users to interact with various sensors, motors, and other devices easily.

3.) **Compatibility**: The Arduino Uno is compatible with a

vast array of shields and extension boards, enabling further
expansion of functionalities for complex projects.

Microcontroller: ATmega328P

Operating Voltage: 5V

Input Voltage (Recommended): 7-12V

Digital I/O Pins: 14 (of which 6 can be used as PWM outputs)

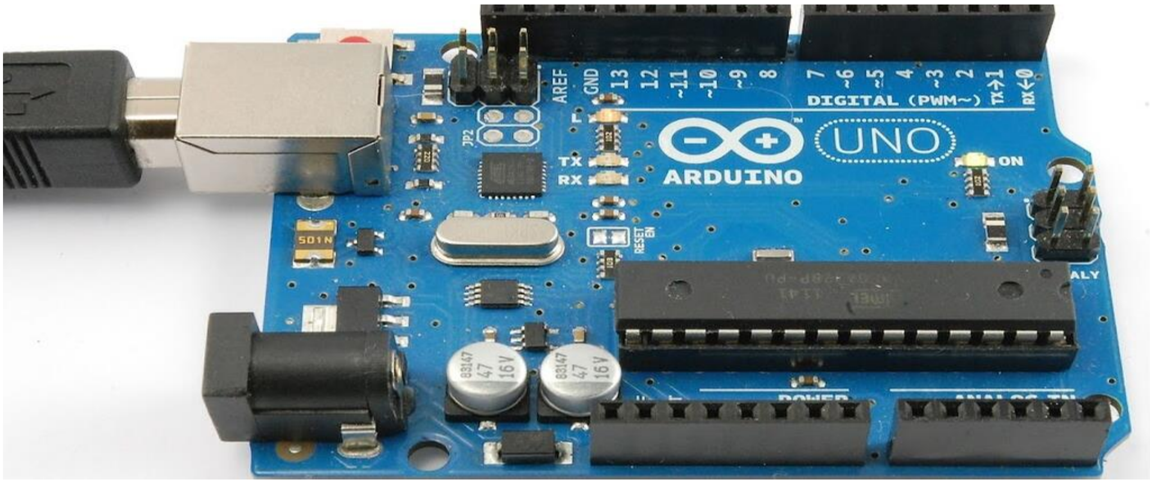
Analog Input Pins: 6

Clock Speed: 16 MHz

Flash Memory: 32 KB (0.5 KB used by the bootloader)

SRAM: 2 KB

EEPROM: 1 KB

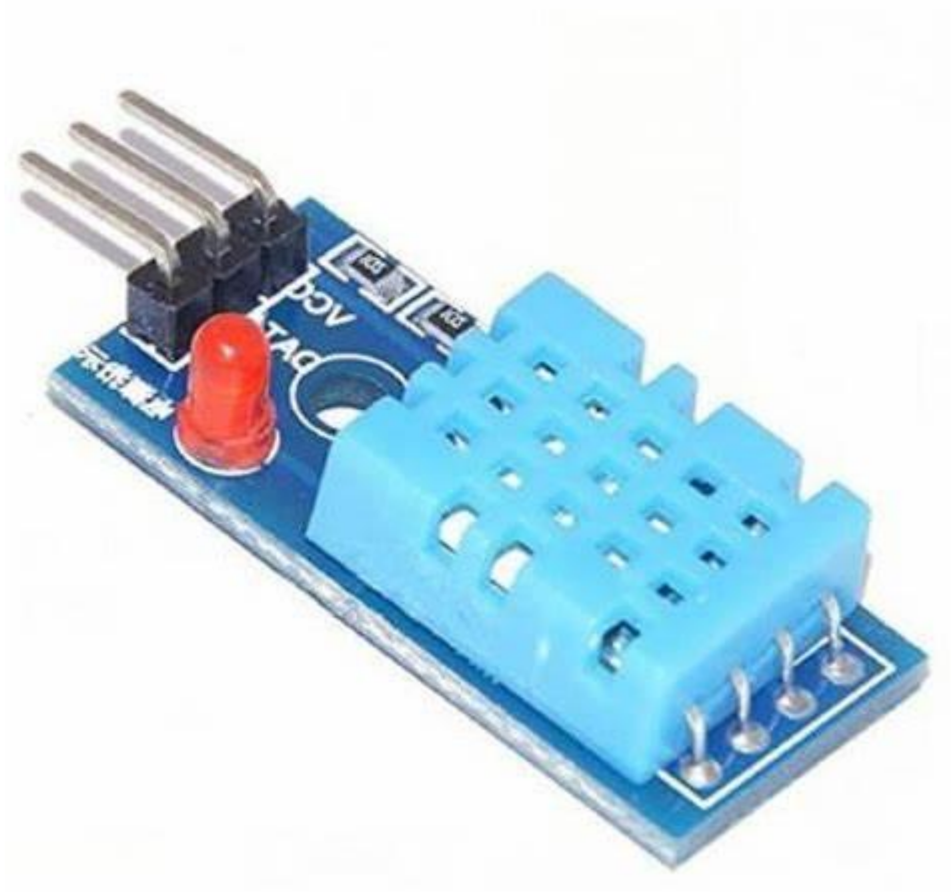


3.2.2 DHT11 Sensor

The DHT11 is a low-cost digital sensor that provides calibrated temperature and humidity readings. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and outputs a digital signal.

Specifications:

- Temperature range: 0–50°C (±2°C accuracy)
- Humidity range: 20–90% RH (±5% accuracy)
- Digital signal output
- Requires a single data pin for communication



•

Fig 3.2.2 DHT11 Sensor

3.2.3 LCD Display (16x2)

The 16x2 LCD module displays two lines of 16 characters each. It uses the Hitachi HD44780 driver and can be connected directly to the Arduino using parallel communication or via an I2C adapter to reduce wiring.

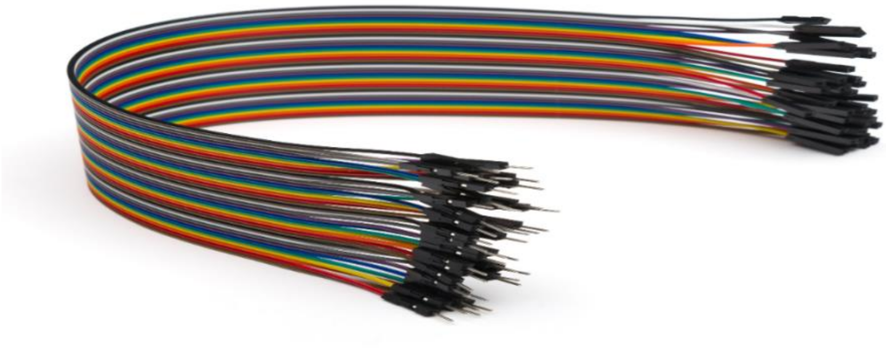
Features:

- 5V operating voltage
- Backlit display
- Can display alphanumeric and special characters

3.2.3 Lcd sensor

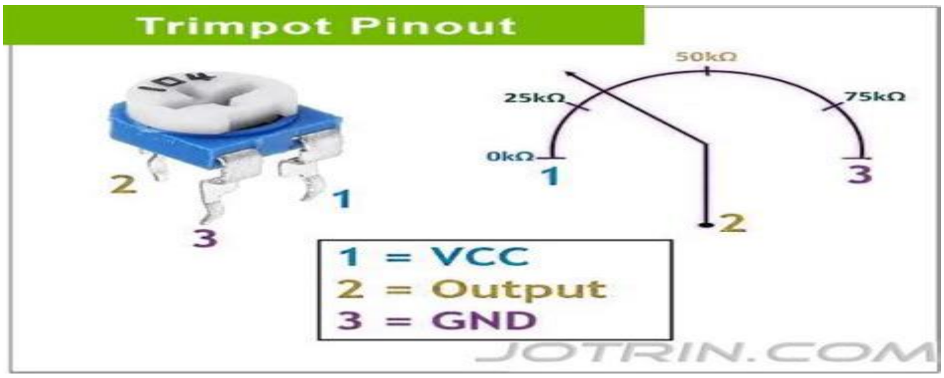
3.Supporting components

Jumper Wires: To connect components



3.2.4 Jumpers Wires

10kΩ Potentiometer: To control LCD contrast



3.2.5 10ohm potentiometer

3.3 Circuit Diagram and Explanation

The DHT11 sensor is connected to one of the Arduino's digital pins (e.g., D2), with VCC and GND connected to 5V and GND respectively. The LCD display is connected using digital pins

(e.g., D7–D12) or via I2C pins (A4 – SDA, A5 – SCL). A potentiometer may be connected to adjust the LCD brightness or contrast.

(Circuit diagram can be inserted here)

The Arduino continuously reads data from the DHT11 at set intervals (e.g., every 2 seconds) and sends formatted output to

the LCD display.

3.4 Software Development

3.4.1 Arduino IDE

The Arduino Integrated Development Environment (IDE) is used to write, compile, and upload the code to the Arduino Uno. It supports C/C++-based syntax and offers a wide range of libraries.

Required Libraries:

- DHT sensor library by Adafruit or Rob Tillaart
- Liquid Crystal library (or LiquidCrystal_I2C if using I2C LCD)

3.4.2 Code Explanation

The code performs the following functions:

- Includes necessary libraries
- Defines sensor and LCD pins
- Initializes the sensor and LCD in the setup () function
- Continuously reads data in the loop() function
- Displays temperature and humidity on the LCD

- Implements delays to prevent excessive polling

3.4.3 Code

```
#include <DHT.h>;
//I2C LCD:
#include <LiquidCrystal_I2C.h>
#include <Wire.h>



LiquidCrystal_I2C lcd(0x27,16,2); // set the LCD address to 0x27 for a
16 chars and 2 line display

//Constants
#define DHTPIN 7    // what pin we're connected to
#define DHTTYPE DHT11   // DHT 11
DHT dht(DHTPIN, DHTTYPE); //// Initialize DHT sensor for normal 16mhz
Arduino



//Variables
//int chk;
int h;  //Stores humidity value
int t; //Stores temperature value



void setup()
{
    Serial.begin(9600);
    Serial.println("Temperature and Humidity Sensor Test");
    dht.begin();
   //initialize the lcd
    lcd.backlight(); //open the backlight
}



void loop()
{
    //Read data and store it to variables h (humidity) and t
```

```
(temperature)
    // Reading temperature or humidity takes about 250 milliseconds!
    h = dht.readHumidity();
    t = dht.readTemperature();

    //Print temp and humidity values to serial monitor
    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.print(" %, Temp: ");
    Serial.print(t);
    Serial.println(" ° Celsius");

// set the cursor to (0,0):
// print from 0 to 9:



    lcd.setCursor(0, 0);
    lcd.println(" Now Temperature ");

    lcd.setCursor(0, 1);
    lcd.print("T:");
    lcd.print(t);
    lcd.print("C");



    lcd.setCursor(6, 1);
    lcd.println("2020 ");

    lcd.setCursor(11, 1);
    lcd.print("H:");
    lcd.print(h);
    lcd.print("%");

  delay(1000); //Delay 1 sec.
}
```

3.4.3.1 code

3.5 Working Principle

The system works on the principle of digital sensing and serial communication. The DHT11 measures the surrounding air's temperature and humidity and sends the data to the Arduino. The Arduino processes this data and formats it into human-readable strings. It then sends these

strings to the LCD for real-time display. This process repeats at fixed intervals, ensuring continuous monitoring

CHAPTER 4
Implementation and Results

4.1 Hardware Setup and Assembly

The hardware components were assembled on a breadboard for ease of prototyping. The Arduino Uno served as the central microcontroller, interfaced with the DHT11 temperature and humidity sensor and a 16x2 LCD for display output. The assembly process involved the following steps:

- The DHT11 sensor was connected to digital pin D2 of the Arduino Uno, with power and ground connected to 5V and GND respectively. A 10kΩ pull-up resistor was connected between the VCC and data pin of the sensor to stabilize signal communication.

- The 16x2 LCD display was connected to digital pins D12 (RS), D11 (EN), and D5–D2 (data pins D4 to D7). The contrast of the LCD was controlled using a 10kΩ potentiometer connected between VCC, GND, and the VO pin of the LCD.

- All components were powered via the USB connection from the computer or alternatively through a 9V battery connected to the Arduino power jack.

- A breadboard and jumper wires were used to organize
- connections securely and clearly.

Optional enhancements such as I2C communication for the LCD or wireless transmission were considered for future iterations.

4.2 Software Compilation and Upload

The Arduino IDE was used to write and upload the program to the Arduino Uno. The software setup included:

- Installation of the DHT sensor library (e.g., Adafruit DHT library).
- Inclusion of the Liquid Crystal library for the LCD module.
- Writing the main code to read temperature and humidity values and display them on the LCD screen.
- Compilation and verification of the code to check for syntax errors.
- Uploading the program to the Arduino board via USB.

Sample output on the LCD:

Line 1: Temp: 26.0C
Line 2: Humidity: 57%

4.3 Data Collection and Sample Output

After successful uploading and powering of the system, data collection was initiated. The DHT11 sensor read the temperature and humidity of the environment at an interval of 2 seconds. Sample data readings are shown below:

Time (HH:MM: SS)	Temperature (°C)	Humidity (%)
10:01:00	26.0	58
10:01:02	26.1	58
10:01:04	26.3	59
10:01:06	26.4	59
10:01:08	26.4	60

These values were validated against a commercial digital thermometer-hygrometer to ensure the readings were reasonably accurate for a DHT11 sensor.

4.4 Result Analysis

The DHT11 sensor provided consistent data within its specified accuracy range (±2°C for temperature and ±5% for humidity). The following observations were made:

The sensor had a response delay of approximately 1–2 seconds, which is acceptable for non-critical real-time monitoring.

- The readings were relatively stable, with minor fluctuations attributed to environmental changes or sensor resolution.
- The LCD displayed the readings clearly, and the contrast could be easily adjusted using the potentiometer.

Although the DHT11 is not suitable for high-precision applications, it performed reliably for basic monitoring tasks.

Graphical representation of the data (optional for thesis):

- Line graph showing temperature vs. time
- Line graph showing humidity vs. time

4.5 Challenges and Troubleshooting

Several issues were encountered during implementation and were addressed as follows:

Issue 1: LCD Not Displaying Text

- Resolution: Checked wiring and ensured the contrast pin was connected through a potentiometer. Also confirmed

- Liquid Crystal library pin mappings matched the circuit.

Issue 2: DHT11 Sensor Returning ‘Nan’

- Resolution: Ensured proper pull-up resistor was connected. Verified correct pin assignment and rechecked data line connection.

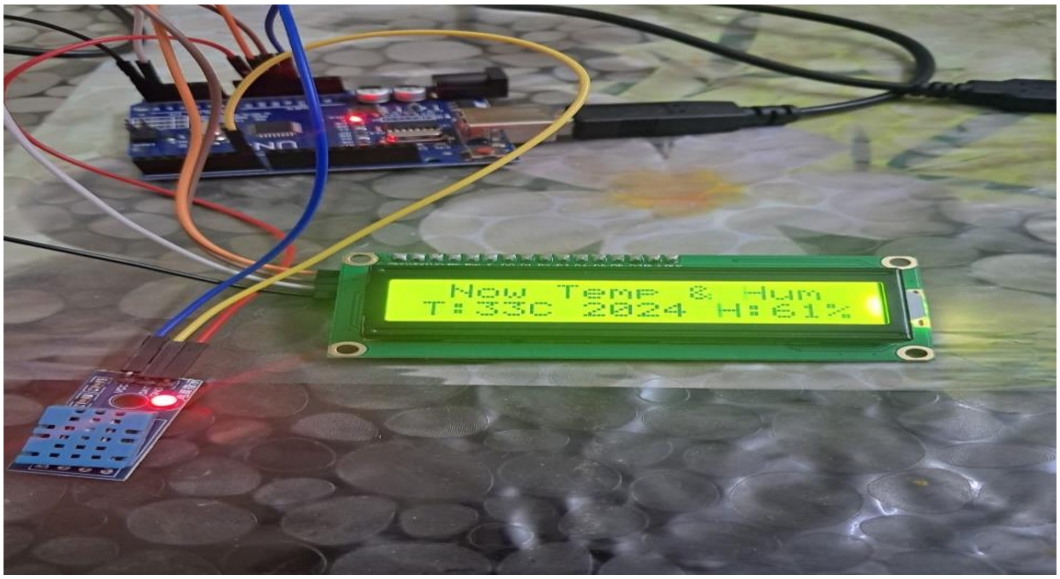
Issue 3: Arduino Not Detected by Computer

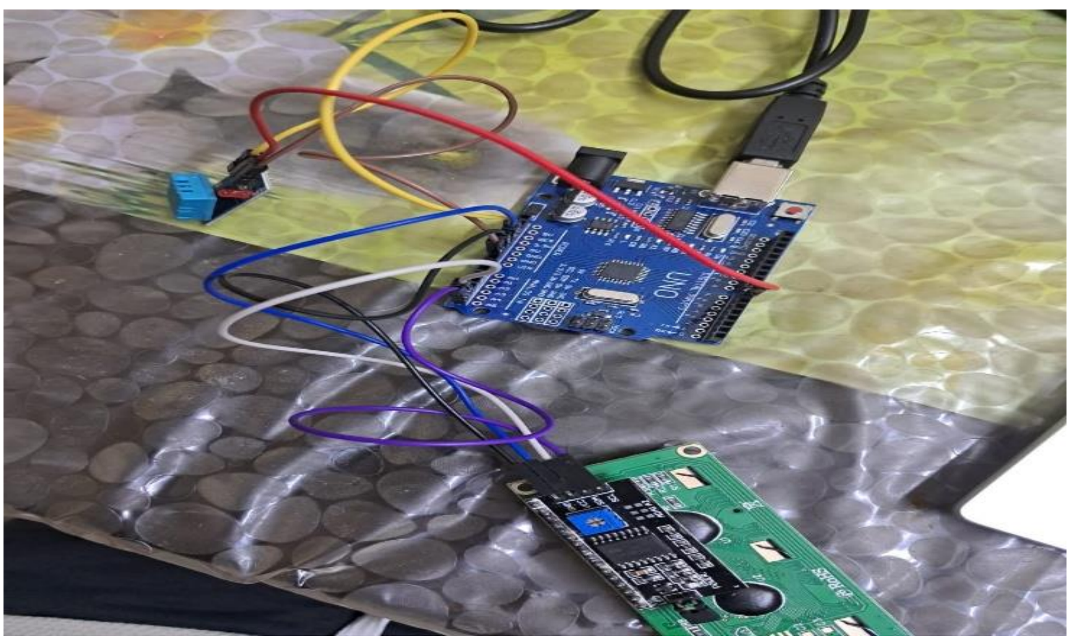
Resolution: Installed necessary drivers and selected the correct COM port and board from the Arduino IDE.

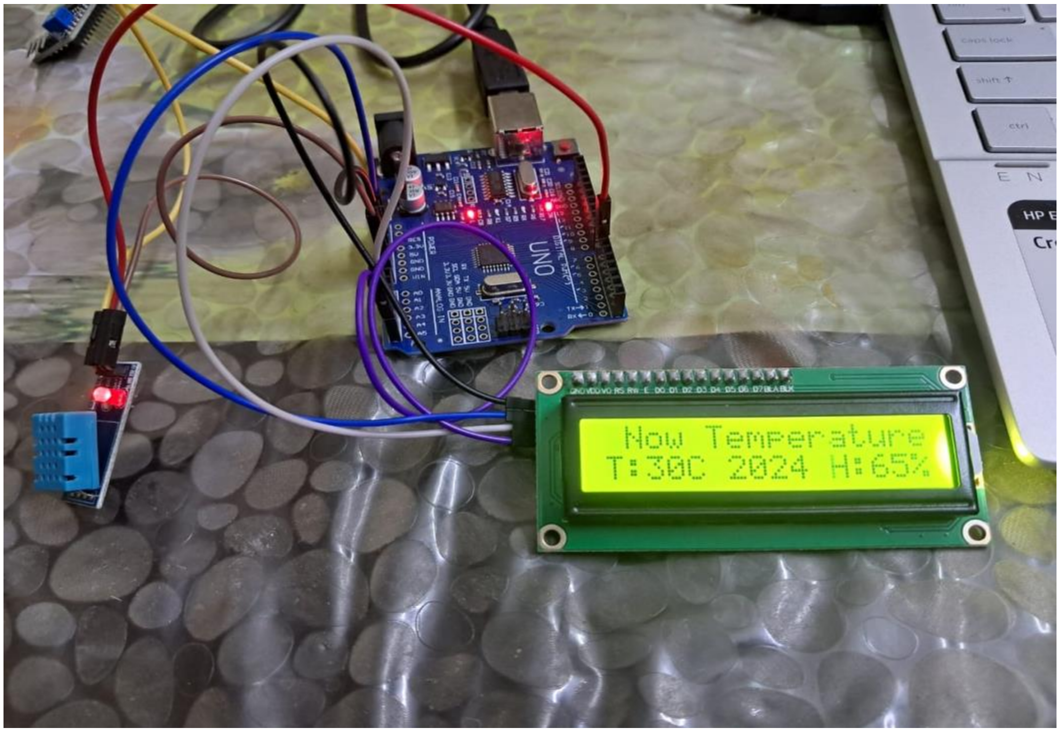
Issue 4: Inconsistent Readings

- Resolution: Added a delay of 2 seconds between reads to allow the sensor to stabilize. Avoided polling the sensor too frequently.

4.6 Output







4.6.1 output

Chapter 5
Conclusion and Future Work

5.1 Summary of the Project

This project focused on the design and implementation of a simple environmental monitoring system using an Arduino Uno microcontroller and a DHT11 temperature and humidity sensor. The goal was to create a low-cost, real-time system capable of capturing and displaying temperature and humidity data on a 16x2 LCD screen. The system was successfully assembled, programmed using the Arduino IDE, and tested in various conditions to ensure reliability and performance. Throughout the project, the integration of sensor technology with embedded systems programming was explored and demonstrated effectively.

5.2 Achievements and Findings

The key achievements and findings of the project include:

- Successful integration of the DHT11 sensor with the Arduino Uno for accurate digital measurement of ambient temperature and humidity.
- Real-time data acquisition and display of sensor reading

- on a 16x2 LCD using the Liquid Crystal library.
- Stable operation of the system with consistent and readable

  outputs updated every 2 seconds.
- Hands-on experience with embedded system design, sensor interfacing, breadboard prototyping, and Arduino-based programming.
- Validation of sensor readings against commercial weather instruments showed acceptable accuracy for non-critical applications.

This project confirmed that microcontroller-based systems can be effectively used for basic environmental monitoring applications.

5.3 Limitations

While the project achieved its primary objectives, it had several limitations:

- Sensor Accuracy: The DHT11 sensor has limited accuracy (±2°C for temperature and ±5% for humidity), making it unsuitable for precision-critical environments.
- Sampling Frequency: The DHT11 has a low refresh rate (1

Hz), which restricts high-frequency data logging.

- Lack of Data Logging: The current system displays data in real time but does not store historical data for analysis.

Power Source: The system relies on USB or a 9V battery, which limits its long-term deployment in remote areas without an external power supply or solar integration.

5.4 Recommendations for Future Improvements

To enhance the functionality and scalability of the system, the following improvements are recommended:

- Use a More Accurate Sensor: Replace the DHT11 with a more accurate sensor like the DHT22 or BME280, which offers better precision and additional parameters (e.g., barometric pressure).

- Add Data Logging: Integrate an SD card module or EEPROM to store temperature and humidity data over time for further analysis.

- Wireless Communication: Incorporate wireless modules such as ESP8266 (Wi-Fi) or HC-05 (Bluetooth) for remote

- monitoring via a smartphone or web interface.

Web/Mobile Interface: Develop a simple mobile or web app

- Solar Power: Integrate a solar power supply and battery management system for off-grid operation.

- Expand Sensor Network: Build a network of multiple sensors across different locations using I2C or wireless mesh communication for broader environmental monitoring.

These enhancements would not only make the system more versatile but also position it for practical applications in smart agriculture, indoor air quality monitoring, and IoT-based smart home systems