Innovation & Entrepreneurship Hub for Educated Rural Youth (SURE Trust – IERY)

Humidity & Temperature Monitor with DHT11 & STM32 Microcontroller

The domain of the Project

Embedded System, Internet of Things (IoT)

Under The Guidance of:

Kiran Kumar Reddy (Firmware Engineer)

By:

Mr. Utkarsh Patel, B.Tech, 3rd year pursuing

Period of the project

18th December 2024 to 16th February 2025

DECLARATION

The project titled "**Humidity & Temperature Monitor with DHT11 & STM32 Microcontroller**" has been mentored by **Kiran Kumar Reddy**, organised by SURE Trust, from 18th December 2024 to 16th February 2025, for the benefit of the educated unemployed rural youth for gaining hands-on experience in working on industry relevant projects that would take them closer to the prospective employer.

I, Mr. Utkarsh Patel, hereby declare that I have solely worked on this project under the guidance of my mentor. This project has significantly enhanced my practical knowledge and skills in the domain.

Name: **Signature**

Mr. Utkarsh Patel Utkarsh

Mentor's Name: **Signature**

Kiran Kumar Reddy P. Kiran Kumar Reddy
Firmware Engineer at Oben Electric

Seal & Signature:

Prof. Radhakumari
Executive Director & Founder
SURE Trust

TABLE OF CONTENTS

(●’◡’●)

INTRODUCTION

Background and Context:

With the growing emphasis on environmental monitoring and smart automation, temperature and humidity sensing play a crucial role in various applications, including agriculture, industrial safety, and home automation. Maintaining optimal environmental conditions is essential in greenhouses, storage units, and weather monitoring systems. Traditional monitoring methods often lack efficiency and real-time updates, necessitating automated systems that can continuously track and display environmental parameters.

Problem Statement:

Many existing temperature and humidity monitoring systems are either expensive, complex, or inefficient in real-time applications. Industries and smart home solutions require an affordable and reliable solution that integrates seamlessly into existing infrastructure. This project aims to address these issues by developing a **low-cost, real-time monitoring system** using an **STM32F103C8 microcontroller** and a **DHT11 sensor**, which offers accurate data with minimal power consumption.

Scope of the Project:

This project focuses on designing and implementing an embedded system capable of measuring **temperature and humidity** and displaying the readings on a **16x2 LCD**. The system will be simple, scalable, and suitable for applications in **weather stations, greenhouses, industrial monitoring, and home automation**. Future enhancements can integrate wireless communication for remote monitoring, making it a valuable addition to IoT-based environmental monitoring solutions.

PROJECT OBJECTIVES

The primary goal of this project is to design and implement a system that measures **humidity and temperature** using the **DHT11 sensor** interfaced with an **STM32F103C8 microcontroller**. The collected data is displayed on a **16x2 LCD screen**, providing real-time environmental monitoring.

Key Features:

1) Efficient and accurate measurement of temperature and humidity.
2) Use of a **low-cost microcontroller (STM32F103C8)** for real-time processing.
3) Displaying sensor readings on an **LCD module**.
4) Application in **weather stations, greenhouses, industrial environments, and smart homes**.

METHODOLOGY AND RESULTS

A. Methods/Technology used:

This project uses two communication protocols:

- **Single-Wire Protocol (for DHT11 Sensor):** The DHT11 sensor communicates with the STM32F103C8 microcontroller using a custom single-wire digital communication protocol. In this protocol:
  - The microcontroller sends a start signal to the sensor.
  - The sensor responds with a 40-bit data stream, which includes temperature and humidity values along with a checksum for error detection.

This single-wire communication requires only one GPIO pin (PA0) for both sending and receiving data.

- **4-bit Parallel Communication (for 16x2 LCD):**
  - The 16x2 LCD communicates with the microcontroller using a 4-bit parallel data transfer mode, which reduces the number of pins needed compared to 8-bit mode.
  - The microcontroller sends commands and data to the LCD through control lines (RS, EN) and data lines (D4 to D7).
  - The LCD interprets these signals to display the temperature and humidity readings.

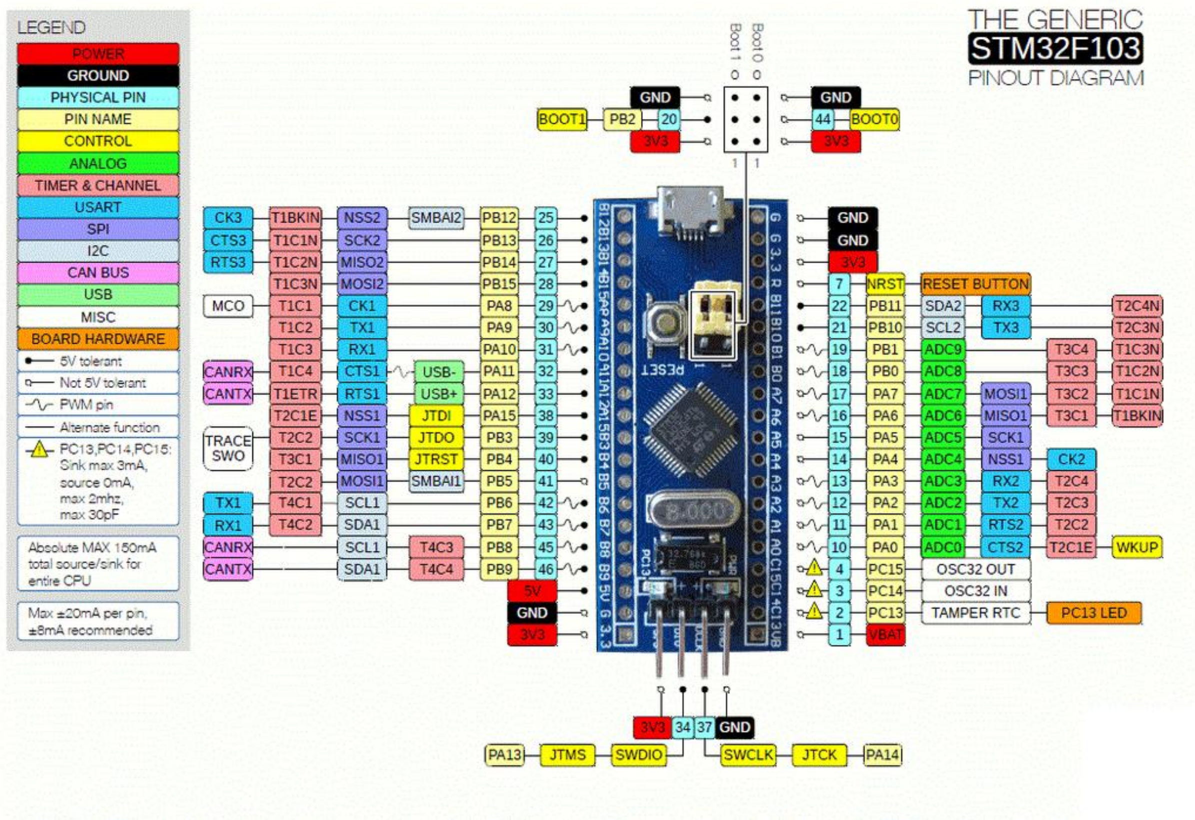
B. Tools & Software used:

1) **STM32F103C8 Microcontroller:** The STM32F103C8 is a 32-bit microcontroller from STMicroelectronics, based on the ARM Cortex-M3 core. It belongs to the STM32F1 series and is commonly used in embedded systems due to its balance of performance, power efficiency, and affordability. Some key features are: ARM Cortex-M3 (32-bit, up to 72 MHz) ,Flash Memory is of 64 KB or 128 KB (depending on version), 20 KB SRAM, 37 GPIOs, **Communication Interfaces:** 2x I2C, 3x USART, 2x SPI, 1x USB 2.0, 1x CAN

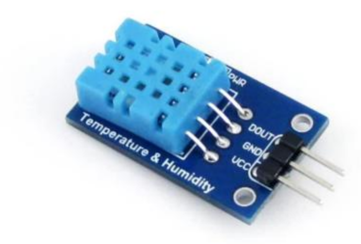
   Timers: 7 (including PWM support)

   ADC: 12-bit, 10 channels

   Operating Voltage: 2.0V – 3.6V



2) **DHT11 Sensor**: The DHT11 is a low-cost digital temperature and humidity sensor commonly used in embedded systems and IoT applications. It provides calibrated digital output, making it easy to interface with microcontrollers like Arduino, STM32, and ESP32. Some key Features are: Temperature Range is 0°C to 50°C (±2°C accuracy), Humidity Range is 20% to 90% RH (±5% accuracy), Operating Voltage is 3.3V – 5V, Communication is Single-wire digital signal



3) **LCD 16 X 2:** The 16x2 LCD (Liquid Crystal Display) is a widely used alphanumeric display module that can show 16 characters per row across 2 rows. It is based on the Hitachi HD44780 controller, making it compatible with many microcontrollers like Arduino, STM32, and ESP32. Some key Features are: Display Capacity is 16 characters × 2 rows, Operating Voltage is 4.7V – 5.3V, Interfacing method is in Parallel (4-bit or 8-bit mode) for microcontroller connection I2C module (optional) for easier interfacing, Character Size is ~2.95×5.55 mm

4) **Potentiometer (10K):** A 10K potentiometer (pot) is a variable resistor used to adjust voltage levels, control signal strength, or modify resistance in circuits. It is commonly used for contrast control in LCDs, volume adjustment in audio circuits, and sensor calibration.

5) ST Link V2 Programmer: The ST-Link V2 is an in-circuit debugger and programmer used for STM32 and STM8 microcontrollers. It allows flashing firmware and debugging applications via SWD (Serial Wire Debug) or JTAG interfaces.

6) **Arduino IDE**
7) **STM32 CubeProgrammer**

C. Circuit Connections:

The connections are made as follows:

i. DHT11 Sensor:

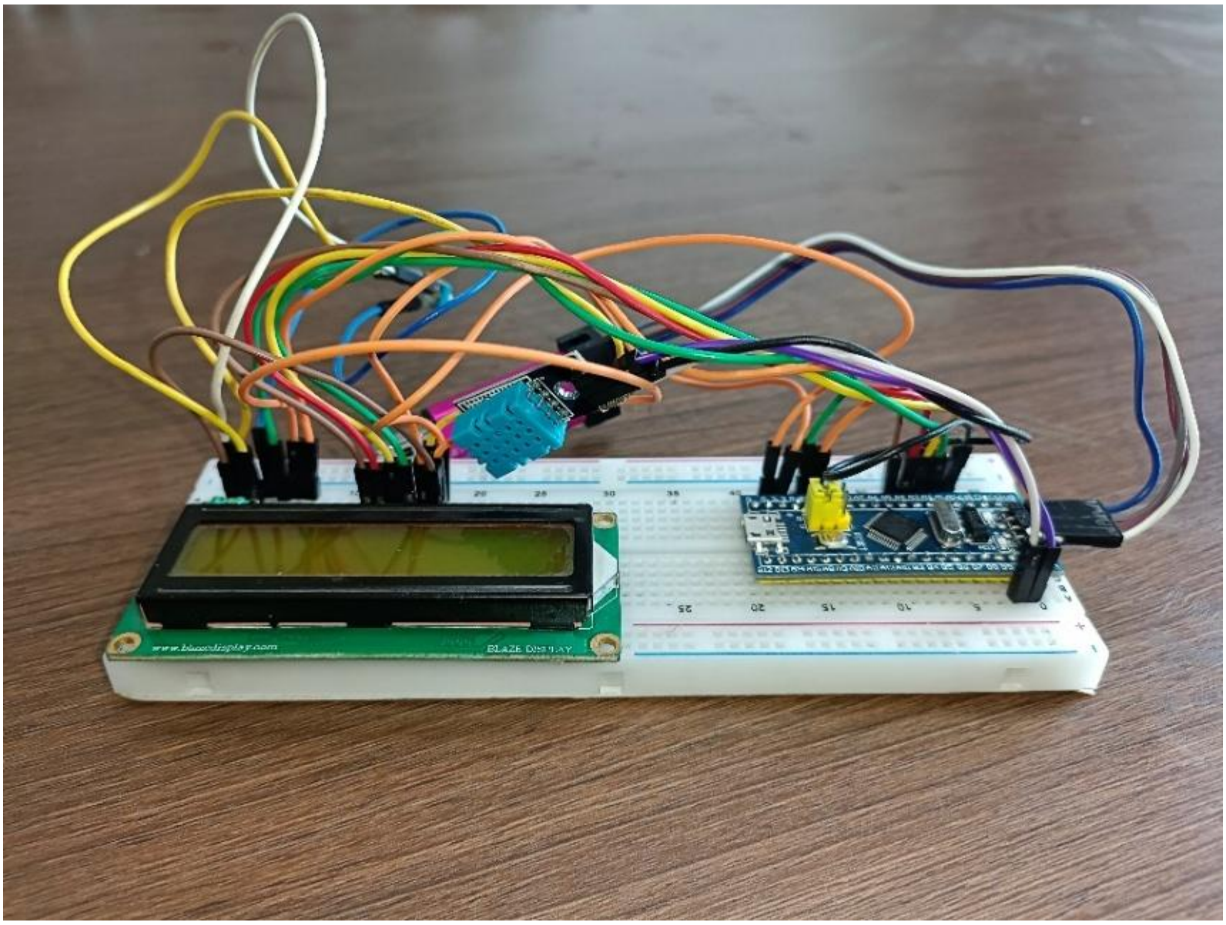
- **VCC**: Connect to 3.3V pin of the STM32F103C8.
- **GND**: Connect to GND.
- **Data**: Connect to GPIO pin PA0.

ii. 16x2 LCD:

- **RS**: Connect to PB11.
- **EN**: Connect to PB10.
- **D4-D7**: Connect to PA4, PA3, PA2, and PA1 respectively.
- **VCC**: Connect to 5V.
- **VSS**: Connect to GND.

- **V0**: Connect to a 10K potentiometer for contrast control.

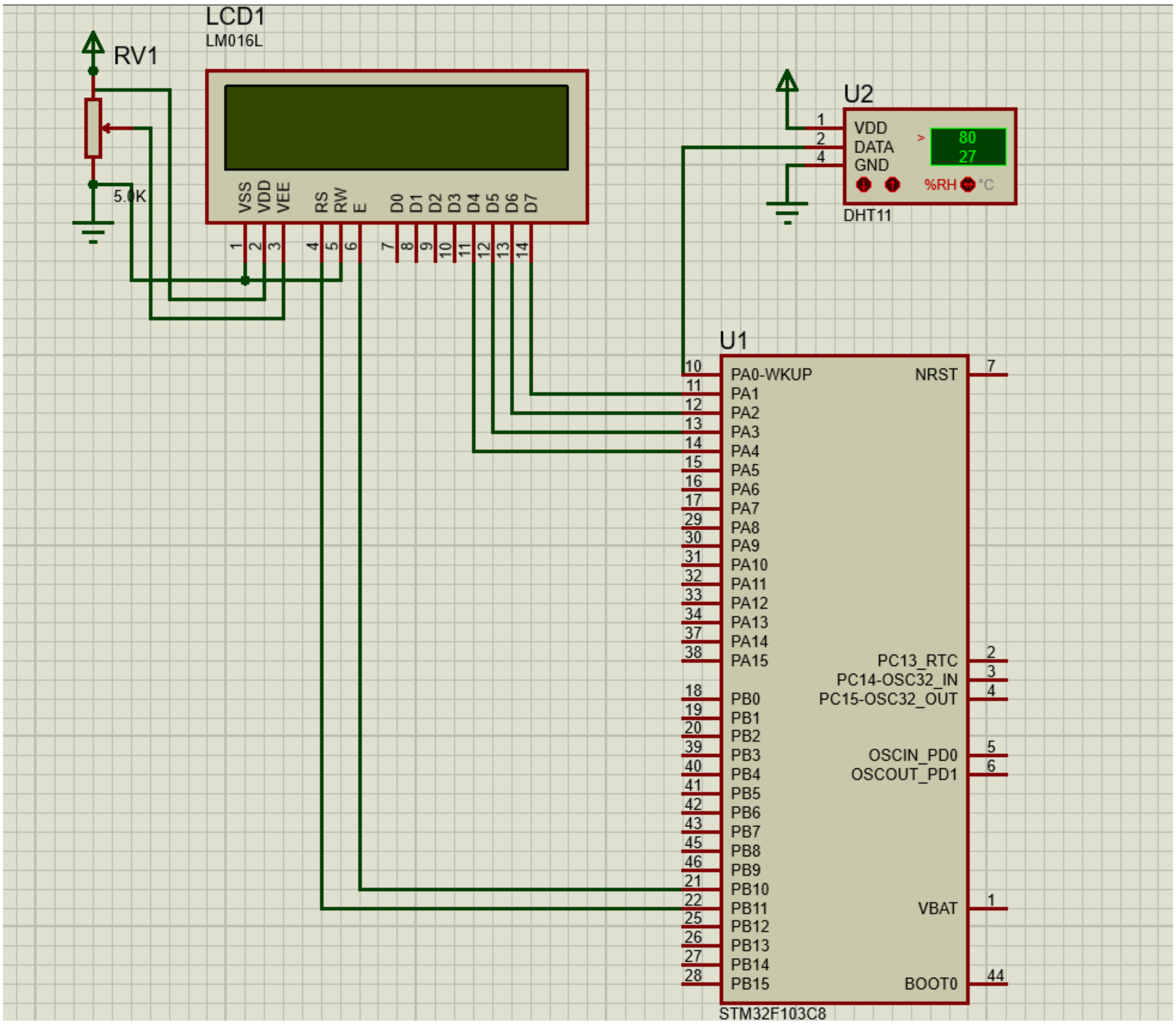
Hardware Implementation:



Implementation & Testing:

- The **STM32F103C8** microcontroller was programmed using **Arduino IDE**.
- Upon powering the circuit, the **DHT11 sensor** captures temperature and humidity data.
- The measured values are **displayed on the 16x2 LCD**.
- In case of sensor failure, the LCD displays an **error message**.

Circuit Diagram:



STM32F108C8T6 Program on Arduino IDE:

```
#include <LiquidCrystal.h>
#include <DHT.h>

// LCD pins connection
const int rs = PB11, en = PB10, d4 = PA4, d5 = PA3, d6 = PA2, d7 = PA1;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

#define DHTPIN PA0        // Pin for DHT11 data
#define DHTTYPE DHT11     // DHT11 sensor type
DHT dht(DHTPIN, DHTTYPE); // Initializing the DHT sensor

void setup()
{
  // Initializing the LCD and DHT sensor
  lcd.begin(16, 2);
  dht.begin();
  lcd.print("DHT11 with STM32");
  delay(3000); // Wait for message to display
  lcd.clear();
}
```

```

void loop()
{
  // Read sensor data
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  // Error handling for sensor readings
  if (isnan(h) || isnan(t)) {
    lcd.setCursor(0, 0);
    lcd.print("Sensor Error");
    lcd.setCursor(0, 1);
    lcd.print("Check DHT");
    delay(2000); // Wait before retrying
    return;
  }

  // Display temperature and humidity
  lcd.setCursor(0, 0);
  lcd.print("Temp: ");
  lcd.print(t, 1); // One decimal place
  lcd.print(" C");
  lcd.setCursor(0, 1);
  lcd.print("Humid: ");
  lcd.print(h, 1); // One decimal place
  lcd.print(" %");

  delay(2000); // Update every 2 seconds
}

```

SOCIAL & INDUSTRY RELEVANCE OF THE PROJECT

This project has multiple real-world applications, including:

1. **Weather Monitoring Stations** – Helps in tracking temperature and humidity for weather prediction.
2. **Greenhouse Monitoring** – Ensures optimal climate conditions for plant growth.
3. **Industrial Environment Monitoring** – Used in factories to maintain stable environmental conditions.
4. **Smart Home Automation** – Can be integrated into IoT-based home automation systems for climate control.

LEARNING AND REFLECTION

The project uses two communication protocols:

1. Single-Wire Protocol (for DHT11 Sensor):

- The **DHT11 sensor** communicates with the STM32F103C8 microcontroller using a **custom single-wire digital communication protocol**.
- In this protocol:
  - The microcontroller sends a **start signal** to the sensor.
  - The sensor responds with a **40-bit data stream**, which includes **temperature and humidity values** along with a **checksum** for error detection.
- This single-wire communication requires only **one GPIO pin (PA0)** for both sending and receiving data.

---

2. 4-bit Parallel Communication (for 16x2 LCD):

- The **16x2 LCD** communicates with the microcontroller using a **4-bit parallel data transfer mode**, which reduces the number of pins needed compared to 8-bit mode.
- The microcontroller sends commands and data to the LCD through **control lines (RS, EN)** and **data lines (D4 to D7)**.
- The LCD interprets these signals to display the temperature and humidity readings.

Summary of Protocols:

- **DHT11 to STM32:** Single-wire protocol (custom digital communication)
- **STM32 to LCD:** 4-bit parallel communication protocol.

CONCLUSION AND FUTURE SCOPE

The **Humidity & Temperature Monitor with DHT11 & STM32 Microcontroller** can be enhanced in several ways to make it more efficient, scalable, and suitable for broader applications. Some possible future developments include:

1. **Integration with IoT & Cloud Services**
    - Adding Wi-Fi (ESP8266/ESP32) or Bluetooth (HC-05/HC-06) for remote data monitoring.
    - Connecting to cloud platforms like AWS IoT, Google Firebase, or Thingspeak for real-time data storage and analysis.
2. **Mobile Application Support**
    - Developing an Android/iOS app to monitor temperature and humidity remotely.
    - Implementing real-time notifications and alerts for extreme conditions.
3. **Multiple Sensor Support**
    - Expanding the system to include additional sensors like:
        - CO_2 & Air Quality Sensors (MQ-135) for environmental monitoring.
        - Soil Moisture Sensors for smart farming applications.
        - Pressure & Light Sensors for industrial and automation use cases.

4. **Energy Efficiency Enhancements**
    - Implementing low-power modes in STM32 to reduce energy consumption.
    - Using solar power or battery backup to ensure continuous operation in remote areas.
5. **AI & Predictive Analysis**
    - Using machine learning algorithms to predict temperature and humidity trends.
    - Implementing automated climate control systems in greenhouses and industrial setups.
6. **Industrial & Smart Home Automation**
    - Integrating with Home Automation Systems using Google Home, Alexa, or Raspberry Pi.
    - Implementing automated ventilation & cooling control based on sensor readings.
7. **Wireless Sensor Networks (WSN)**
    - Deploying multiple STM32-based sensor nodes to form a distributed monitoring system.
    - Using LoRa or Zigbee for long-range data communication in agricultural and industrial applications.

GitHub Repository Link:

https://github.com/sure-trust/UTKARSH-PATEL-g11-es/tree/a179aa9dd275c97425459215ddb30b08af49b398/Final%20capstone%20project

---

(●’◡’●)