NORTH SOUTH UNIVERSITY

Project Report

Project Title
Temperature-Controlled Fan

Team members:

Anika Tabassum	2211012042
Mohosina Islam Disha	2212349042
Muhammad Tahmidur Rahman	2211295042
Mehedi Hassan Roktim	2212497042

Project Proposal: Temperature-Controlled Fan

1. Project Title and Objective

Title: Smart Temperature-Controlled Fan using STM32
Objective:
The objective of this project is to design and implement an automated cooling system where the fan speed adjusts automatically according to the surrounding temperature. The system aims to reduce manual effort, improve energy efficiency, and maintain optimal environmental conditions.

2. Problem Statement

Traditional electric fans require manual adjustment of speed depending on temperature changes. People often forget to increase or decrease fan speed based on room conditions, leading to discomfort and unnecessary power consumption. There is a need for an embedded system that automatically regulates fan speed based on real-time temperature sensing, ensuring comfort and efficient energy usage.

3. Proposed Solution Overview

The proposed system uses an STM32 microcontroller combined with a DHT11 temperature and humidity sensor to read ambient conditions and control the fan speed accordingly through an L298N motor driver.Key functionality:

- Continuously monitor temperature
- Automatically adjust fan speed in multiple levels
- Display real-time temperature, humidity, and fan status on a 16x2 I2C LCD
- Operate efficiently with minimal user intervention
  Error detection and retry mechanism for sensor failures

Control Logic:

- Below 25°C → Fan OFF (0%)
- 25-30°C → Fan SLOW (40% speed)
- 30-35°C → Fan MEDIUM (70% speed)
- Above 35°C → Fan FAST (100% speed)

4. System Components and Sensors

Microcontroller:

- STM32F103C8T6 ("Blue Pill")

Sensors:

- DHT11 Temperature Sensor

Actuators:

- DC Fan (12V)
- L298N Motor Driver Module (for bi-directional motor control with PWM)

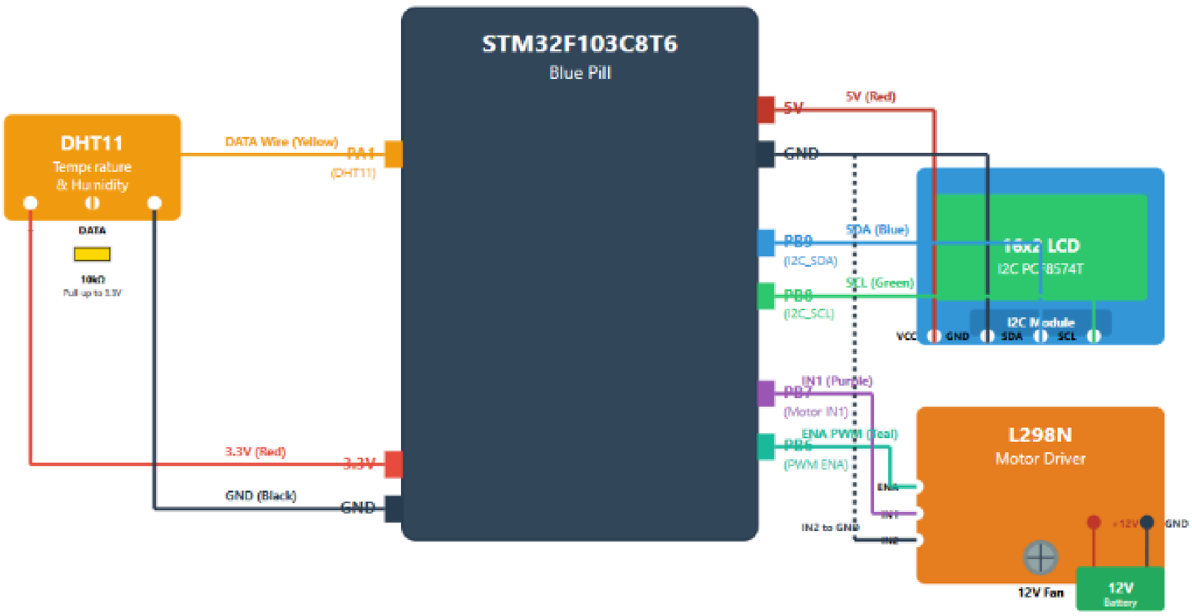
Display:

- 16x2 LCD with I2C Module (Address: 0x27)

Other Components:

- 5V Power supply for STM32 and sensors
- 12V Power supply for motor driver and fan
- Jumper wires
- Breadboard

5. Block Diagram



6. Working Principle / Algorithm

Step-by-step Logic:

1. Initialize STM32, I2C, TIM2, TIM4 (PWM), GPIO, LCD, and DHT11 sensor
2. Display welcome message on LCD
3. Perform motor diagnostic test
4. Enter continuous monitoring loop:
   - Read temperature and humidity from DHT11 (with 3 retry attempts)
   - Validate data using checksum verification
   - Apply temperature-based control logic
   - Update motor speed via PWM
   - Display current readings and fan status on LCD
   - Handle sensor errors with appropriate feedback
5. Repeat every 2.5 seconds

Pseudocode:

```
START

initialize HAL
initialize I2C, TIM2, TIM4, GPIO
start TIM2 (for microsecond delays)
start TIM4_PWM (for motor speed control)

initialize LCD
initialize Motor (direction = forward, speed = 0)

display "DHT11 Fan Ctrl"
display "Initializing..."
delay(2000ms)

run Motor_Test()  // Test 30%, 60%, 100% speeds

display "Waiting DHT11..."
delay(2000ms)

WHILE (true):

    // Try reading DHT11 sensor (up to 3 attempts)
    retry = 0
    success = false

    WHILE (retry < 3 AND success == false):
```

```
    success = readDHT11()
    IF (success == false):
        retry++
        delay(100ms)
    END IF
END WHILE

    IF (success == true):
    temp = dht11_data.temperature
    humidity = dht11_data.humidity

    IF (temp < 25):
        setFanSpeed(0)
        show("T:##C H:##%")
        show("Fan: 0%")

    ELSE IF (temp >= 25 AND temp < 30):
        setFanSpeed(40)
        show("T:##C H:##%")
        show("Fan: 40%")

    ELSE IF (temp >= 30 AND temp < 35):
        setFanSpeed(70)
        show("T:##C H:##%")
        show("Fan: 70%")

    ELSE:
        setFanSpeed(100)
        show("T:##C H:##%")
        show("Fan: 100%")
    END IF

ELSE:
    errorCount++
    show("DHT11 ERROR!")

    IF (errorCount >= 5):
        show("Check wiring!")
        setFanSpeed(0)  // Stop motor for safety
    ELSE:
        show("Retry: #/5")
    END IF
END IF
```

```
    delay(2500ms)

END WHILE

END
```

7. Expected Output / Demonstration

Normal Operation:

- LCD shows real-time temperature and humidity
- Fan speed automatically adjusts based on temperature thresholds
- Smooth transitions between OFF, SLOW, MEDIUM, and FAST speeds

Temperature Increase Test:

- Apply heat source near DHT11 sensor
- Observe fan speed increase from 0% → 40% → 70% → 100%
- LCD updates in real-time

Error Handling:

- Disconnect DHT11 sensor
- System displays "DHT11 ERROR!" and retry attempts
- After 5 failures, displays "Check wiring!" and stops motor for safety

Motor Diagnostic Test:

- On startup, motor cycles through 30%, 60%, and 100% speeds
- Confirms proper L298N driver and PWM functionality

8. Pin Mapping Table

STM32F103C8T6 Pin Configuration:

STM32 Pin	Pin Name	Function	Mode	Configuration	Connected To
PA1	GPIO	DHT11 Data	Input (dynamic)	Open-Drain + Pull-Up	DHT11 DATA pin
PB6	TIM4_CH1	Motor Speed (PWM)	Alternate Function	PWM Output	L298N ENA pin
PB7	GPIO	Motor Direction	Output	Push-Pull	L298N IN1 pin
PB8	I2C1_SCL	LCD Clock	Alternate Function	Open-Drain + Pull-Up	LCD I2C SCL
PB9	I2C1_SDA	LCD Data	Alternate Function	Open-Drain + Pull-Up	LCD I2C SDA
5V	Power	5V Supply	Power	-	DHT11 VCC, LCD VCC
GND	Ground	Common Ground	Power	-	All GND pins

DHT11 Temperature & Humidity Sensor:

DHT11 Pin	Wire Color (Typical)	STM32 Pin	Description
VCC	Red	5V	Power supply (+5V)
DATA	Yellow/White	PA1	Digital data signal (bidirectional)
GND	Black	GND	Ground (0V)

LCD Display (16x2 with I2C Module):

LCD I2C Pin	Wire Color (Typical)	STM32 Pin	Description
GND	Black	GND	Ground (0V)
VCC	Red	5V	Power supply (+5V)
SDA	Green/Blue	PB9	I2C Data line
SCL	Yellow/White	PB8	I2C Clock line

L298N Motor Driver Module:

L298N Pin	Wire Color	Connection	Description
IN1	Orange	PB7	Direction control (GPIO)
ENA	Yellow	PB6 (TIM4_CH1)	Enable/Speed control (PWM)
+12V	Red (thick)	12V Power Supply (+)	Motor power supply
GND	Black (thick)	Power Supply GND	Motor ground (power)
+5V (Logic)	Red (thin)	5V or jumper ON	Logic power (can use onboard regulator)
GND (Logic)	Black (thin)	GND (Common)	Logic ground
OUT1	Red	DC Fan (+)	Motor output positive
OUT2	Black	DC Fan (-)	Motor output negative

DC Fan (12V):

Fan Pin	Wire Color	Connection	Description
Positive (+)	Red	L298N OUT1	Positive terminal
Negative (-)	Black	L298N OUT2	Negative terminal

9. Technology and Tools

IDE: STM32CubeIDE
Firmware Libraries: HAL Drivers
Programming Language: C
Hardware Tools: Soldering iron, breadboard
Debug Tools: ST-Link V2 programmer

10. Tentative Timeline

Week	Task

Week 1	Finalize idea, collect components, create block diagram
Week 2	Configure STM32 ADC, PWM, and read sensor values
Week 3	Implement fan speed control logic using PWM
Week 4	Integrate LCD display and test full system
Week 5	Debug, optimize, and finalize hardware setup
Week 6	Prepare report, presentation, and final demonstration

11. Cost Table(Tentative)

Item	Qty	Unit Price (BDT)	Total (BDT)
STM32F103C8T6 MCU	1	450	**450**
DHT11 Sensor	1	150	**150**
L298N Motor Driver Module	1	250	**250**
DC Fan (12V)	1	200	**200**
16x2 LCD with I2C Module	1	350	**350**
12V Power Adapter	1	250	**250**
5V Power Supply/USB Cable	1	100	**100**
Breadboard + Jumper Wires	1	150	**150**
Misc. Components (Resistors, etc.)	-	100	**100**
Total Estimated Cost			**2000 BDT**

12. Business Model

Temperature-controlled fans with real-time monitoring are increasingly demanded in home automation and industrial systems. Existing smart cooling devices are often expensive and lack customization.

Market Gap:

- High-end smart cooling systems with temperature sensors cost 4000–6000 BDT
- Most systems don't display humidity or provide detailed monitoring
- Limited customization and repair options

Our Proposed System Offers:

- Advanced features (temperature + humidity monitoring) at ≈2000 BDT
- Real-time LCD display showing both environmental data and fan status
- Fully customizable, open-source, and easily repairable
- Energy-efficient with intelligent 4-level speed control
- Reduces electricity usage by up to 35%
- Suitable for homes, classrooms, server rooms, small offices, and embedded system demonstrations

Profitability: By producing in bulk with optimized component sourcing, total cost can drop to around 1600 BDT, allowing a resale price of 2800–3200 BDT with a healthy profit margin (40-50%) while remaining significantly cheaper than market alternatives.

Target Market:

- Students and educational institutions
- Small offices and home users
- Electronics enthusiasts and makers
- Server rooms requiring automated cooling
- Prototype for larger industrial applications