NATIONAL INSTITUTE OF TECHNOLOGY SIKKIM

Embedded Systems
(EC16101)
Project Report
Academic Year 2023-2024

Submitted by:

ARTI GUPTA

Semester VI

Submitted to:
Dr. Neha K. Nawandar
(Course Instructor)
Department of EEE, NIT Sikkim

Contents

1. Problem Statement:

Design an automated fan that turns On/Off based on the surrounding temperature. As Sikkim is a cold place, keep the temperature thresholds so the fan turns on in certain scenarios

2. Hardware and Software Components:

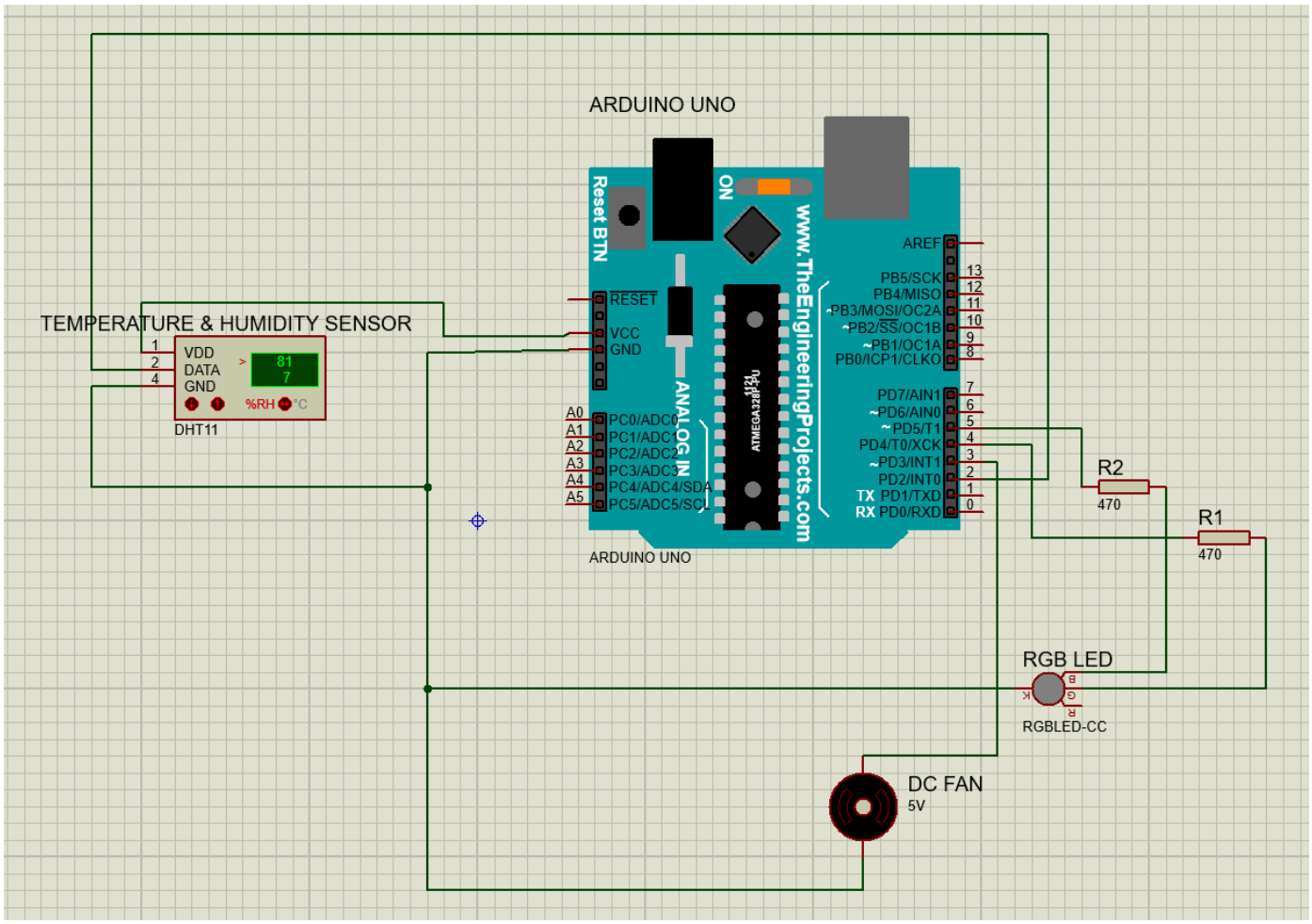
Hardware:

- Arduino Uno microcontroller
- Temperature and Humidity sensor (DHT11)
- 5V DC Cooling Fan
- Breadboard
- Jumper wires
- Power Supply

Software:

- Arduino IDE for programming the Arduino Uno

3. Circuit Diagram/Design Schematic:



Schematic in Proteus 8 Professional

4. Theory:

Temperature and humidity are critical parameters in various applications, from climate control in homes to monitoring environmental conditions in industrial settings. The DHT11 sensor is a popular choice for measuring both temperature and humidity due to its low cost, simplicity, and reasonable accuracy. It utilizes a capacitive humidity sensor and a thermistor to measure relative humidity and temperature, respectively. The DHT11 sensor outputs digital data that can be easily read and processed by microcontrollers like the Arduino Uno.

The Arduino Uno serves as the central control unit in this project, interfacing with the DHT11 sensor, RGB LED, and cooling fan. It provides the nEEEssary computational power and input/output capabilities to implement the desired logic for temperature and humidity-based control of the fan and RGB LED. The Arduino Uno's versatility and ease of programming make it an ideal choice for such applications.

The RGB LED (Red, Green, Blue Light Emitting Diode) is a special type of LED capable of emitting light in multiple colors by adjusting the intensity of its red, green, and blue components. By controlling the intensity of each color component independently, a wide range of colors can be achieved. In this project, the RGB LED is used to visually indicate the state of the system, with blue indicating that the fan is off and green indicating that the fan is on.

The cooling fan is a DC fan used to regulate the temperature by circulating air in the surrounding environment. When the fan is turned on, it draws in ambient air and expels it, helping to dissipate heat and lower the temperature. The fan is connected to a digital pin of the Arduino Uno, allowing it to be controlled programmatically based on temperature and humidity readings from the DHT11 sensor.

The resistors (two 470 Ohm resistors) are used to limit the current flowing through the green and blue terminals of the RGB LED when connected to the Arduino Uno's digital pins. By connecting a resistor in series with each LED terminal, the current is restricted to a safe level, preventing damage to the LED and the Arduino Uno.

The connection of the DHT11 sensor to the Arduino Uno involves connecting its Vcc pin to the 3.3V output of the Arduino Uno, its GND pin to the ground (GND) of the Arduino Uno, and its data pin to a digital pin (e.g., pin 2) of the Arduino Uno. This allows the Arduino Uno to read digital data from the DHT11 sensor, including temperature and humidity readings.

The fan is connected to a digital pin (e.g., pin 3) of the Arduino Uno to control its operation. When the digital pin is set to high, the fan is turned on, and when it is set to low, the fan is turned off. This enables the Arduino Uno to regulate the fan based on the temperature and humidity readings from the DHT11 sensor.

The RGB LED is connected to digital pins of the Arduino Uno to control its color based on the system's state. One terminal of the LED is connected to the ground (GND) of the Arduino Uno, while the red, green, and blue terminals are connected to separate digital pins (e.g., pins 4 and 5). By adjusting the voltage levels on these pins, the Arduino Uno can control the intensity of each color component independently, enabling the RGB LED to display different colors.

The logic implemented in the Arduino Uno's code involves continuously reading temperature and humidity data from the DHT11 sensor. If the temperature is below 12°C, the Arduino Uno turns off the fan and sets the RGB LED to display blue color. Conversely, if the temperature exceeds 12°C, the Arduino Uno turns on the fan and sets the RGB LED to display green color. This logic ensures that the fan is activated when needed to maintain a comfortable temperature while providing visual feedback through the RGB LED.

The DHT11 sensor communicates with the Arduino Uno using a single-wire serial protocol. The Arduino Uno sends a request signal to the sensor, prompting it to send temperature and humidity data. The sensor responds by transmitting a 40-bit data packet containing the measured values. The Arduino Uno then interprets this data packet and extracts the temperature and humidity readings for further processing.

The fan's operation is controlled using pulse-width modulation (PWM) signals generated by the Arduino Uno. PWM allows the Arduino Uno to simulate analog output by rapidly switching the digital output pin on and

off at varying duty cycles. By adjusting the duty cycle of the PWM signal, the Arduino Uno can control the speed of the fan, thereby regulating the airflow and temperature in the surrounding environment.

The resistors connected between the digital pins of the Arduino Uno and the green and blue terminals of the RGB LED serve to limit the current flowing through the LED. Without these resistors, the LED may draw too much current from the Arduino Uno's digital pins, potentially damaging them. By adding resistors in series with the LED terminals, the current is restricted to a safe level, ensuring the proper operation of the LED and the Arduino Uno.

The 3.3V output of the Arduino Uno is used to power the DHT11 sensor. This voltage level is sufficient to operate the sensor and provide stable readings of temperature and humidity. By connecting the sensor's Vcc pin to the 3.3V output, the Arduino Uno ensures that the sensor rEEEives the nEEEssary power supply for its operation.

The ground (GND) connections between the Arduino Uno, DHT11 sensor, fan, and RGB LED provide a common reference point for the flow of electrical current. By connecting all the ground terminals together, the Arduino Uno ensures that there is a complete circuit for the electrical signals to flow properly between the various components. This ground connection is essential for the reliable operation of the entire system.

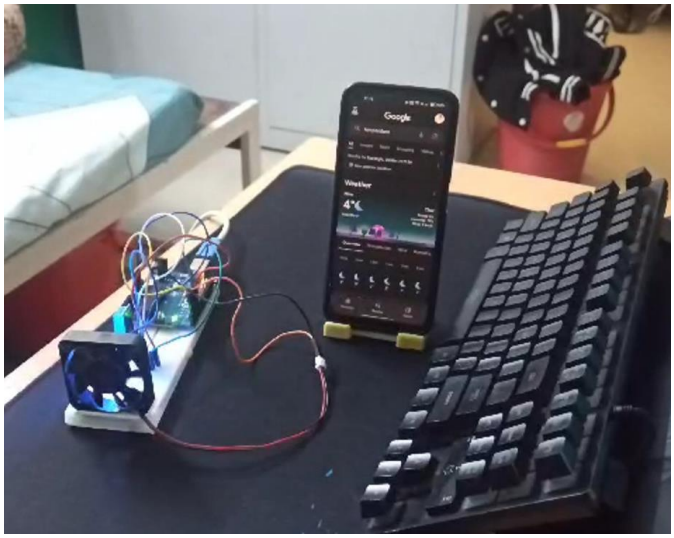
In conclusion, the theory behind this project involves the integration of various hardware components, including the DHT11 sensor, Arduino Uno, cooling fan, RGB LED, and resistors, along with software programming to implement the desired logic for temperature and humidity-based control of the fan and RGB LED. By understanding the operation principles of each component and their interactions within the system, a reliable and effective temperature and humidity-controlled system can be developed for various applications.

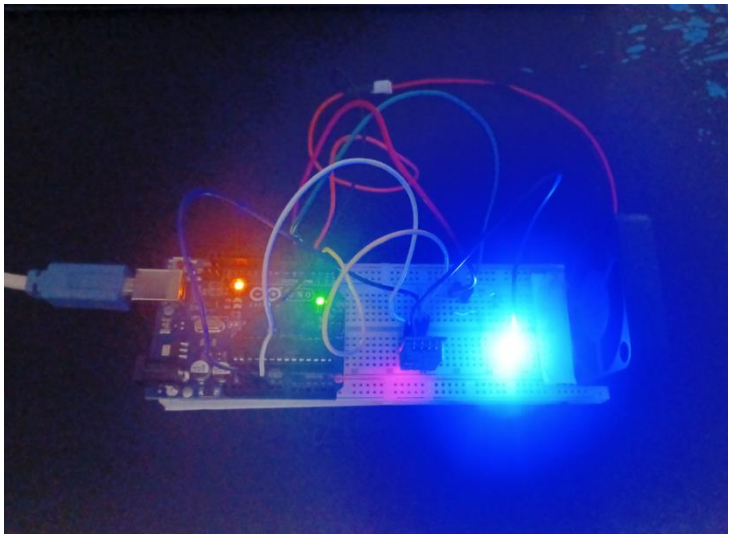
5. Codes:

```
#include <DHT.h>

#define DHTPIN 2
#define DHTTYPE DHT11

#define FAN_PIN 3
#define GREEN_LED_PIN 4
#define BLUE_LED_PIN 5

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  pinMode(FAN_PIN, OUTPUT);
  pinMode(GREEN_LED_PIN, OUTPUT);
  pinMode(BLUE_LED_PIN, OUTPUT);
}

void loop() {
  float humidity = dht.readHumidity();
  float temperature = dht.readTemperature();
  float temp;
  if(temperature < 0){
    temp = temperature +22.7;
    Serial.print("%, Temp: ");
    Serial.print(temp);
     Serial.print("%, temperature: ");
    Serial.print(temperature);
  }else if(temperature > 0 ){
    temp = temperature+0;
    Serial.print("%, Temp: ");
    Serial.print(temp);
     Serial.print("%, temperature: ");
    Serial.print(temperature);
  }

  Serial.print("Humidity: ");
  Serial.print(humidity);

  Serial.println("°C");
```

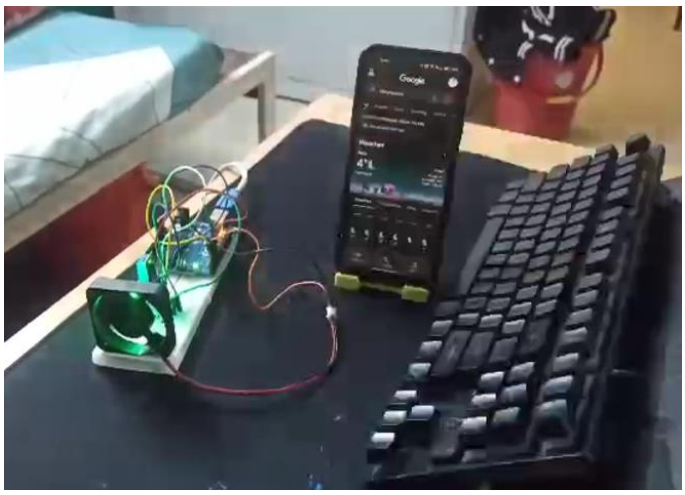
```
  if (temp > 12.0) {
    digitalWrite(FAN_PIN, HIGH);       // Turn on the fan
    digitalWrite(GREEN_LED_PIN, HIGH); // Turn on the green LED
    digitalWrite(BLUE_LED_PIN, LOW);   // Turn off the blue LED
    Serial.println("green");
  } else if(temp < 10){
    digitalWrite(FAN_PIN, LOW );        // Turn off the fan
    digitalWrite(GREEN_LED_PIN, LOW);  // Turn off the green
LED
    digitalWrite(BLUE_LED_PIN, HIGH);  // Turn on the blue LED
    Serial.println("blue");
  }

  delay(1000); // Delay for better readability in Serial
Monitor
}
```

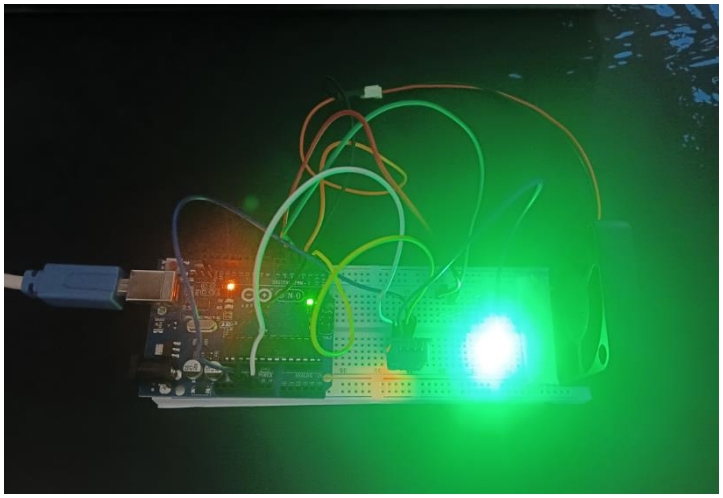
6. Results:





Fan OFF when the Temperature is below 12^0 C





Fan ON when the Temperature is Above 12^0 C

Video link : [Temperature Based Fan Control System]

7. Discussion and Conclusion:

The integration of the DHT11 sensor, Arduino Uno, cooling fan, and RGB LED in this project offers a versatile solution for temperature and humidity control. The system's ability to regulate the fan based on temperature readings ensures efficient cooling when needed, while the RGB LED provides clear visual feedback on the system's status. By employing PWM signals and resistors, the Arduino Uno effectively manages power distribution and LED operation. Furthermore, the project demonstrates the flexibility and adaptability of Arduino-based systems in creating custom environmental control solutions. In conclusion, this project successfully showcases a practical implementation of temperature and humidity-based control using Arduino Uno and sensor technologies. With its simple yet effective design, the system offers potential applications in home automation, climate control, and environmental monitoring. Overall, it underscores the value of leveraging microcontroller platforms like Arduino Uno for creating customizable and efficient control systems.