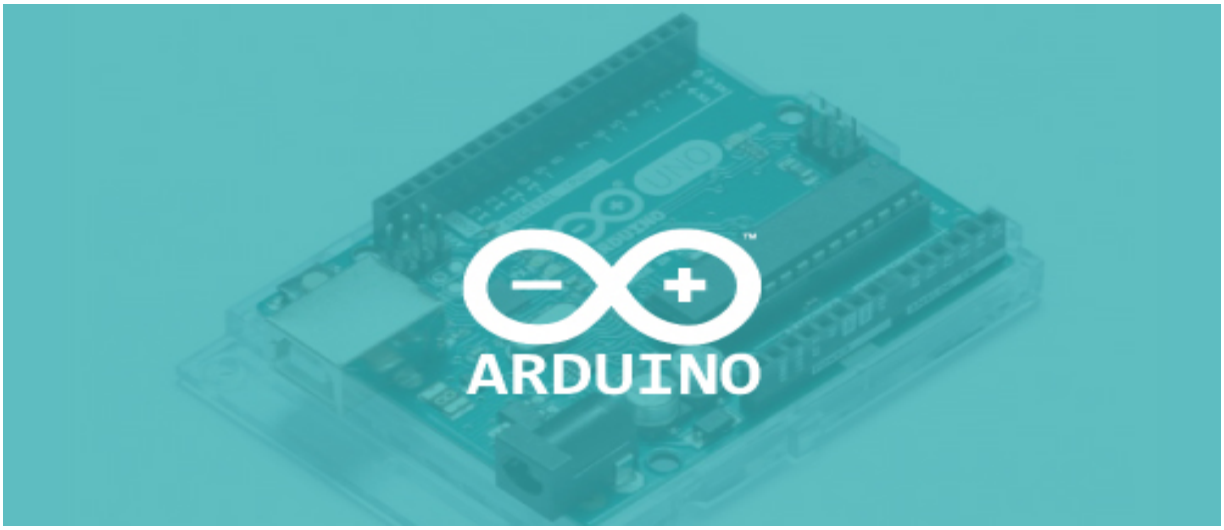


Tinkering Lab (GE107)

Project

"Indoor weather monitoring using Temperature and humidity sensors"

A.Y. 2022-2023 2nd Semester

(Under the Supervision of Dr. Basant Subba, Dr. Sudeepta Mishra, and Mukesh Saini)

Group No. : 06

Team Members :

Nikita Kumari (2020meb1296)
Nehal Sonkar (2021meb1304)
Saket Samota (2021meb1316)
Sakshi Prakash Burud (2021meb1318)
Soumyadip Mondal (2021meb1327)
Altaf Khan (2021meb1266)

Objective :

The objective of the project "Indoor weather monitoring using Temperature and humidity sensors" is to design a system that can accurately measure and report temperature and humidity levels inside the room. The system should be able to monitor and display the data and give signals when the readings fall outside of the specified range.

By ensuring that the temperature and humidity levels stay within acceptable norms, our project aim is to offer comfort. Users' HVAC (heating, ventilation, and air conditioning) maintenance will be made easier.

Components Required :

In this project "Indoor weather monitoring using temperature and humidity sensors", we need the following components :

- **Liquid crystal display (LCD):** With the help of this we can display temperature and humidity values.

- **Temperature and humidity sensors (DHT11 Module):** This is used to collect indoor temperature and humidity levels data.

- **Microcontroller (ESP32):** This microcontroller unit is needed to process the sensor data and provide an interface to transmit the data.

- **Wiring and connections:** To connect the sensor, display, power supply, and microcontroller.

- **Power supply:** We need a battery for the power supply that will help to operate or an AC adapter to provide power to the system.

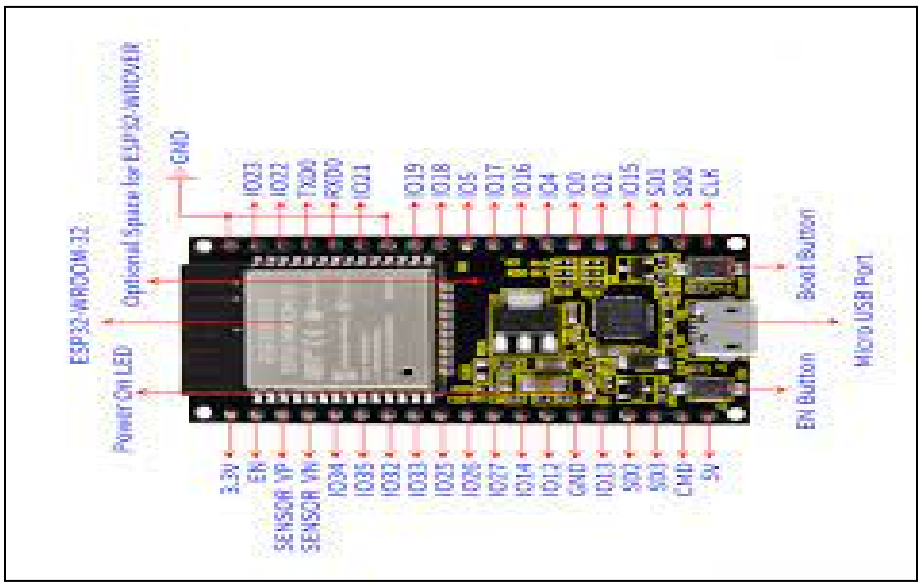
- **Housing and mounting:** In this whole system we may require a housing or enclosure that will protect the components and mounting hardware.

- **Software development tools:** For all this to happen we will need tools like programming languages like C/C++, integrated development environment (IDE), and libraries that will program the microcontroller and develop the system software.

Description of components :

1.ESP32 Microcontroller :

We can define ESP32 as a microcontroller chip that can be programmed using various languages and the Arduino programming language is one of them. By programming, we can perform various work, like reading sensor data and controlling other devices. The integrated development environment which is Arduino IDE gives us a user-friendly interface for the ESP32 microcontroller programming, making it very basic or easier for developers to get into the platform. Here we have attached the image of the ESP32 microcontroller describing its components :

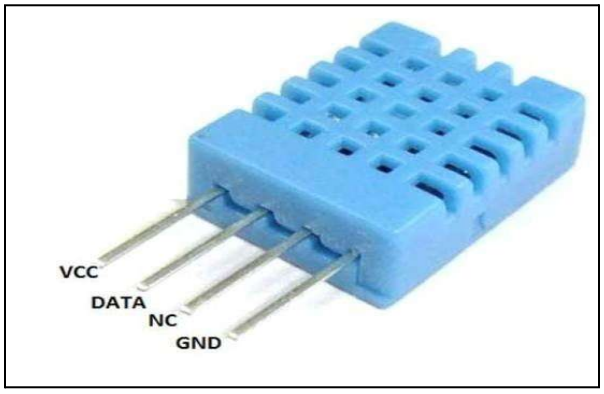


The ESP32 microcontroller can be used for temperature and humidity data collection from sensors. To use the ESP32 microcontroller we need to connect the DHT11 module to the board. The sensor data must then be sent to a local display or a distant server using the ESP32's integrated Wi-Fi or Bluetooth module. This device is very suitable because of its energy efficiency, and affordability.

2.DHT11 Module :

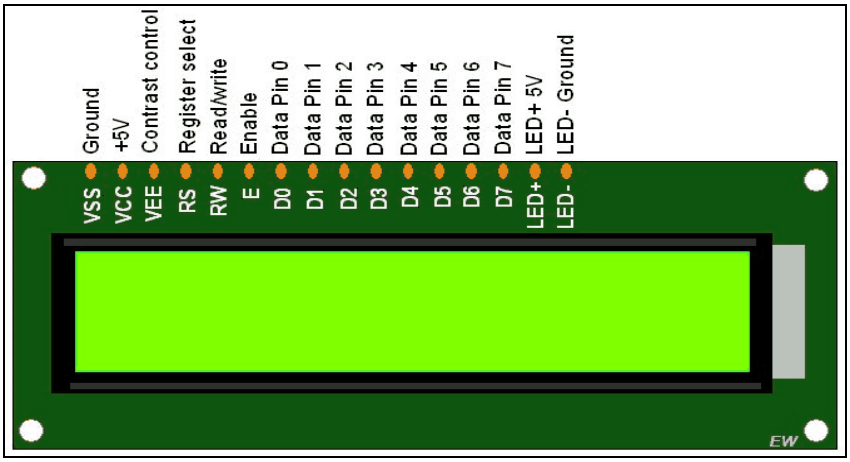
DHT11 module is a Temperature and Humidity sensing module, which uses Digital Signal Acquisition, which converts the Temperature and Humidity to a digital Reading, which can be easily read by a Microcontroller. The temperature range of DHT11 is from 0 to 50 degrees Celsius with a 2-degree accuracy. The humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz.

Each DHT11 element is strictly calibrated in the laboratory which is extremely accurate in humidity calibration. The calibration coefficients are stored as programs in the OTP memory, which are used by the sensor's internal signal-detecting process. It makes the system very easy and quick because of the single-wire serial interface. This device has up to 20-meter signal transmission strength with low power consumption. It has a 4-pin package in a single row.



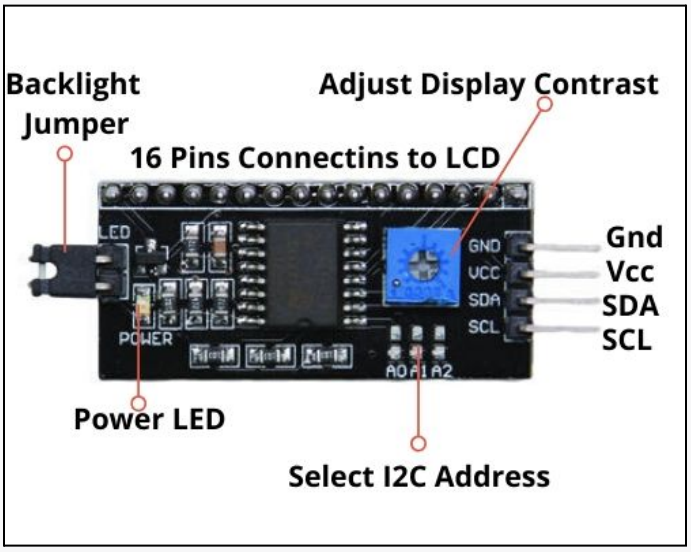
3.LCD (Liquid Crystal Display) :

LCD is a 'liquid crystal display', which is used to display data. LCD screens are made of a thin layer of liquid crystal material between two sheets of polarizing material. When an electric current is applied, the liquid crystal molecules align themselves in such a way that either allows or blocks the passing of light. This created the images that we see on display.
To connect an LCD with an ESP32 microcontroller, we need to use the I2C Protocol. It requires two wires (SDA and SCL) to communicate with the LCD and to transfer the data.
The function which is used to display the Data is lcd.print() and to set the position on the screen at a specific location we can use the function lcd.setCursor().



4.I2C Module :

It's a piece of hardware that ensures communication between different devices and it uses the I2C protocol. It has a method of communication by two wires to send and accept data between these devices. It works as a mediator that allows devices to communicate with each other over a single bus. It has a very significant use in electronic devices like microcontrollers, sensors,s and displays. It is frequently used for short-range, intra-board communication between processors and microcontrollers and lower-speed peripheral ICs. Its ability to let a microcontroller control a network of device chips using only two general-purpose I/O pins and software is one of its main advantages.



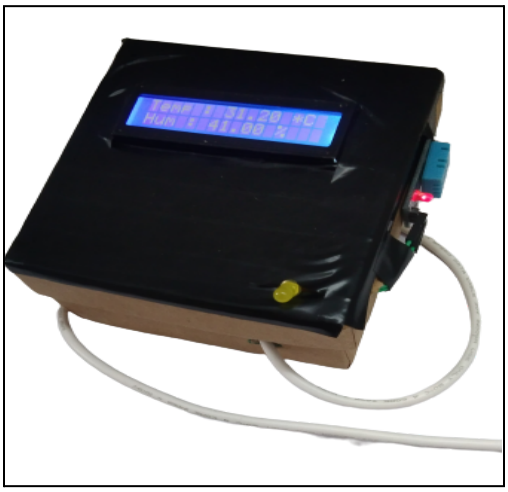
Code Description :

```
/* The objective of the project "Indoor weather monitoring using
temperature and
humidity sensors" is to design a system that can accurately measure and
report
temperature and humidity levels inside the room on the display(LCD).*/

#include <LiquidCrystal_I2C.h> //this library is used to control the LCD
Display.
#include <DHT.h> //this library is used for getting the reading of
temperature and humidity from DHT11 module.
#include <Wire.h> //this library is used for communication and
transferring the Data.

#define DHTTYPE DHT11  //It defines a constant DHTTYPE and assigns it the
value DHT11. This constant is used later to specify the type of DHT sensor
used in the DHT object.
int dhtPin = 32; //This line declares an integer variable dhtPin and
assigns it the value 32.


DHT dht(dhtPin, DHT11); //It creates an instance of the DHT class with the
name dht and initializes it with the pin number dhtPin and the type of DHT
sensor DHT11.

LiquidCrystal_I2C lcd(0x27,16,2); //BY this we create an instance of the
LiquidCrystal_I2C class with the name lcd and initialize it with the I2C
address 0x27, 16 columns and 2 rows.

void setup() {

  dht.begin();
  Serial.begin(9600);
  pinMode(18, OUTPUT);

  lcd.init();
  lcd.backlight();   //These lines initialize the DHT11 sensor, serial
communication, the output pin 18, and the LCD screen backlight.
```

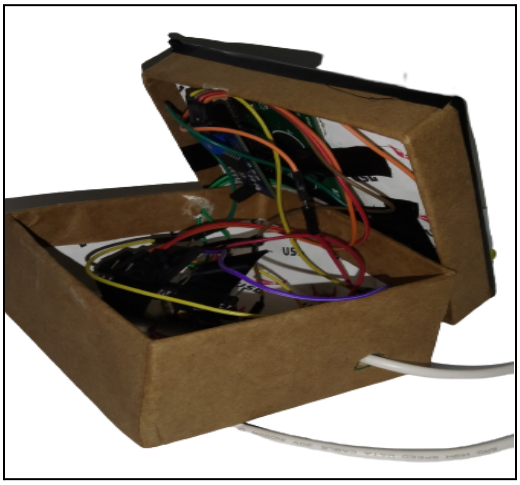
```
}

void loop() {

  float tempreading = dht.readTemperature();
  float humadityreading = dht.readHumidity();
  //These lines read the temperature and humidity data from the DHT11 sensor and store the values in the tempreading and humadityreading variables, respectively.

  if (tempreading >40){
    lcd.setCursor(0,0);// this line of code is used for cursor position that needs to be fixed on screen.
    lcd.print("Temperature is ");
    lcd.setCursor(0,1);
    lcd.print("too high.        ");
    digitalWrite(18, HIGH);
    delay(500);
    digitalWrite(18, LOW);
    delay(500);
      //This if statement checks if the temperature reading is greater than 40 degrees Celsius.
      //If it is, the LCD displays a warning message and an LED connected to pin 18 is turned on and off at 500ms intervals.
  }


  else{

    lcd.setCursor(0,0);
    lcd.print("Temp :");

    lcd.setCursor(7,0);
    lcd.print(tempreading);

    lcd.setCursor(13,0);
    lcd.print("*C");

    lcd.setCursor(0,1);
    lcd.print("Hum :");
```

```
    lcd.setCursor(6,1);
    lcd.print(humadityreading);

    lcd.setCursor(12,1);
    lcd.print("%");

    delay(2000);
  }  //If the temperature reading is less than or equal to 40 degrees
Celsius, this code block is executed.
  //It displays the temperature and humidity readings on the LCD screen
and then waits for 2 seconds before repeating the loop.

}
```

Project images :



Appearance and design



Internal workings, circuit boards, etc.

Conclusion and Discussion :

In this project we have achieved the objective, that we can detect the temperature and humidity of a closed environment using the DHT11 module, then by using the I2C module and ESP microcontroller we can show the values on LCD display which is real-time data. By doing further modifications the sensor can be used in various fields like greenhouse monitoring, and smart home systems like air conditioning, control heating, and ventilation. By doing all this we can help to improve the efficiency of energy and also prevent damage to sensitive materials.

Thank You!!