Weather Station using Arduino UNO and LCD Display

---

1. Introduction

This project is a simple yet effective implementation of a **weather station** using an **Arduino UNO**, **DHT11 temperature and humidity sensor**, and an **LCD display**. It is designed to read real-time temperature and humidity from the environment and display it clearly on an LCD screen using the U8glib library. The project offers a solid foundation for further enhancements in environmental monitoring or home automation systems.

---

2. Components Used

- **Arduino UNO**
- **DHT11 Sensor** (for temperature and humidity)
- **LCD Display 128x64 (I2C)**
- **Jumper Wires**
- **Breadboard or PCB (optional)**
- **Power Supply (USB or 9V battery)**

---

3. Working

The system continuously reads temperature and humidity data from the DHT11 sensor. The Arduino processes this data and displays it on an LCD display using the U8glib library. The readings update every 2 seconds. If the sensor fails to read, the code handles the error gracefully and displays a fallback value.

---

4. Circuit Diagram (Description)

- **DHT11 VCC** → 5V on Arduino
- **DHT11 GND** → GND on Arduino
- **DHT11 Signal** → Digital Pin 2
- **LCD VCC** → 5V
- **LCD GND** → GND
- **LCD SDA** → A4
- **LCD SCL** → A5

---

5. Code

The complete Arduino sketch used for this project is:

```
cpp
CopyEdit
#include <Wire.h>
#include <DHT.h>
#include "U8glib.h"

#define DHTPIN 2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE|U8G_I2C_OPT_DEV_0); // I2C,
Address 0x3D

void setup() {
  Serial.begin(9600);
  dht.begin();
}

void draw(float temperature, float humidity) {
  u8g.setFont(u8g_font_8x13);
  char temp_str[16], humidity_str[16];
  dtostrf(temperature, 1, 2, temp_str);
  dtostrf(humidity, 1, 2, humidity_str);

  int16_t tempWidth = u8g.getStrWidth(temp_str);
  int16_t humidityWidth = u8g.getStrWidth(humidity_str);

  u8g.drawStr(5, 10, "Temp:");
  u8g.drawStr(45, 10, temp_str);
  u8g.drawStr(45 + tempWidth + 5, 10, "C");

  u8g.drawStr(5, 30, "Humi:");
  u8g.drawStr(45, 30, humidity_str);
  u8g.drawStr(45 + humidityWidth + 5, 30, "%");
}

void loop() {
  float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();

  if (isnan(temperature) || isnan(humidity)) {
    Serial.println("Failed to read from DHT sensor!");
    temperature = -99.0;
    humidity = -99.0;
  }

  Serial.print("Temperature: ");
  Serial.print(temperature);
  Serial.print("°C  Humidity: ");
  Serial.print(humidity);
  Serial.println("%");

  u8g.firstPage();
  do {
    draw(temperature, humidity);
  } while (u8g.nextPage());
```

```
  delay(2000);
}
```

---

6. Output

- Real-time temperature and humidity displayed on the LCD screen
- Serial Monitor logs environmental data every 2 seconds
- Graceful handling of DHT sensor read failures

---

7. Applications

- Smart Home Environment Monitoring
- Educational Demonstrations
- Agriculture and Greenhouse Climate Monitoring
- Mini Weather Kiosks

---

8. Advantages

- Compact and low-cost solution
- Easily expandable to include more sensors (e.g., BMP280, MQ135)
- Beginner-friendly and ideal for learning microcontroller integration

---

9. Limitations

- DHT11 has limited accuracy and range
- LCD visibility depends on ambient lighting
- No data logging or long-term storage

---

10. Conclusion

This project demonstrates a minimal but functional weather station using Arduino and common components. While simple, it introduces key embedded concepts like sensor interfacing, I2C communication, and real-time data visualization. The project is ideal for beginners and serves as a base for further IoT or environmental monitoring enhancements.