Cloud Weather Station

Radulescu Kyra-Alexandra

Introduction

Cloud Weather Station is an Arduino project that will read weather data such as temperature and humidity and it will be synchronized with the Arduino IoT Cloud, a service that let's us remotely monitor data.

This means that whenever we read data on the board, it will also be visible in the cloud dashboard, where we can create different ways of visualizing the data.

General Description

We use the Amica NODEMCU cp2102 wifi module as our base arduino board, in which we will plug in our temperature and humidity sensors, we connect it to WIFI, then to Arduino IoT Cloud and put together the dashboard where we will see the data.
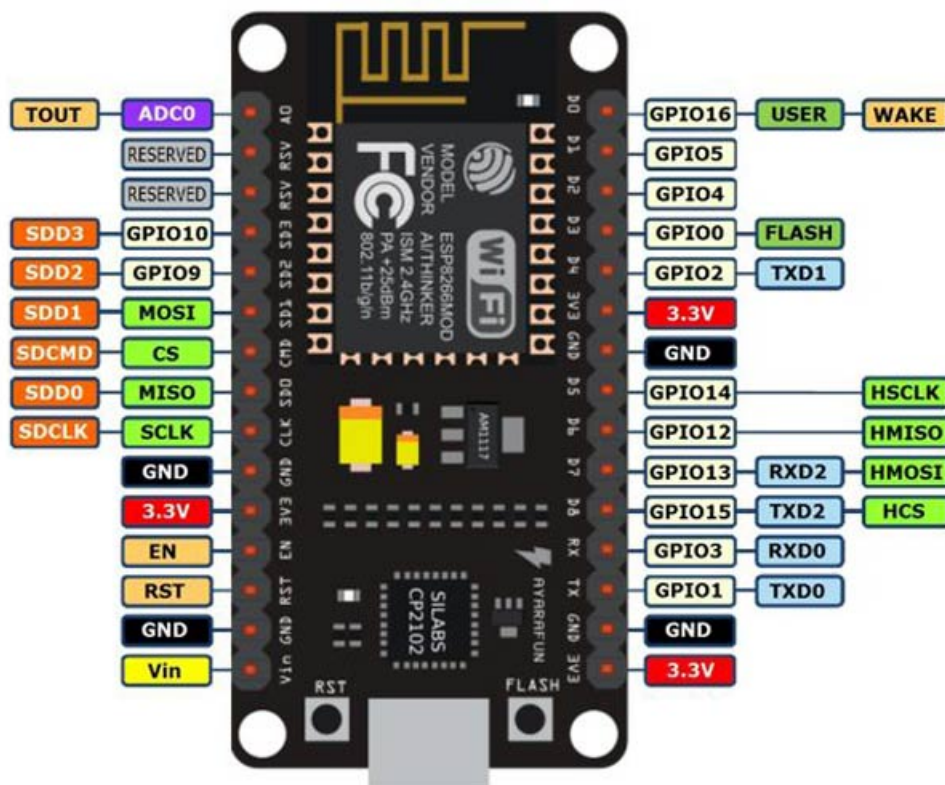
How the **dashboard** should look like:
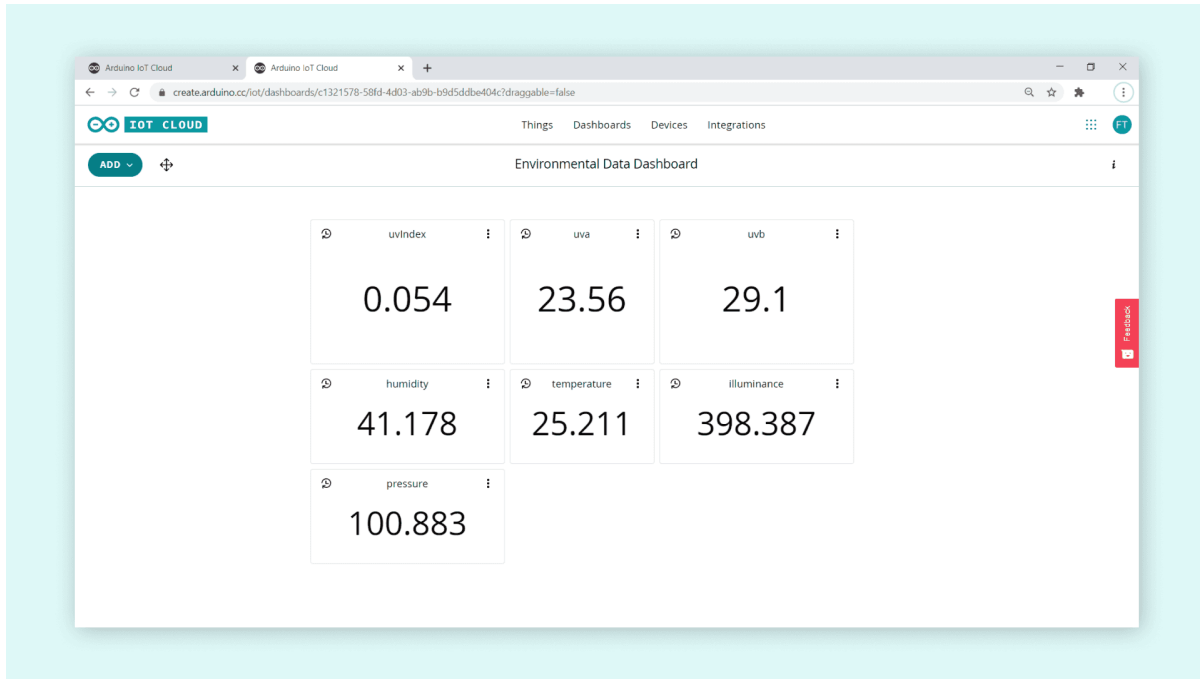
After installing all drivers and libraries for NODEMCU ESP8266 wifi module, it doesn't work. I have consulted the arduino.cc forum and many people have the same problem, the drivers don't work, so I had to change the project a little bit.

Hardware Design

The components used are as follows:

1. Amica NODEMCU cp2102 wifi module 2. CONNFLY temperature sensor 3. CONNFLY humidity sensor

After several tries, the NODEMCU ESP8266 drivers do not work. There is only 1 driver available and it is the same on arduino.cc, nodemcu support page and multiple forums.

```
1    #include "Arduino.h"
2
3    #include "Netdump.h"
4    #include <ESP8266WiFi.h>
5    #include <ESP8266WebServer.h>
6    #include <ESP8266mDNS.h>
7    //#include <FS.h>
8    #include <LittleFS.h>
9    #include <map>
10
11   using namespace NetCapture;
12
13   #ifndef STASSID
14   #define STASSID "VALYPUFOSUL5"
15   #define STAPSK "Megane2008"
16   #endif
17
18   const char* ssid = STASSID;
```

```
Output
         SEGMENT    BYTES       DESCRIPTION
    ICACHE   32768      reserved space for flash instruction cache
    IRAM     27539      code in IRAM
. Code in flash (default, ICACHE_FLASH_ATTR), used 331332 / 1048576 bytes (31%)
    SEGMENT  BYTES      DESCRIPTION
    IROM     331332     code in flash
esptool.py v3.0
Serial port COM1
Connecting........_.....,......,......,.......,......,.......,......
A fatal esptool.py error occurred: Failed to connect to ESP8266: Timed out waiting for packet header
```
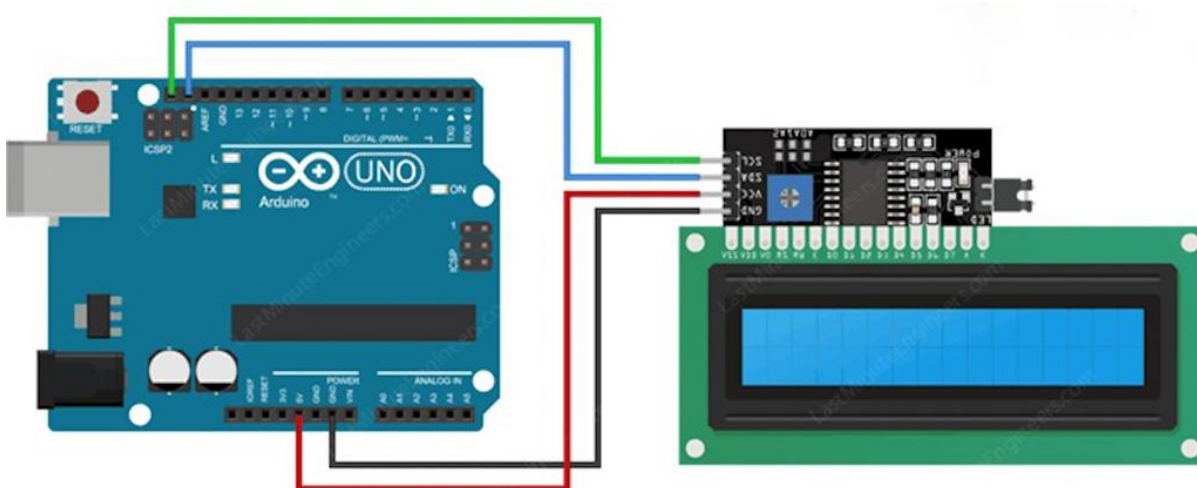
So, I've decided to **change** the project a little bit so I can make it work.

Instead of it transmitting the data from sensors to arduino cloud, it will show it on an 16×2 I2C LCD.
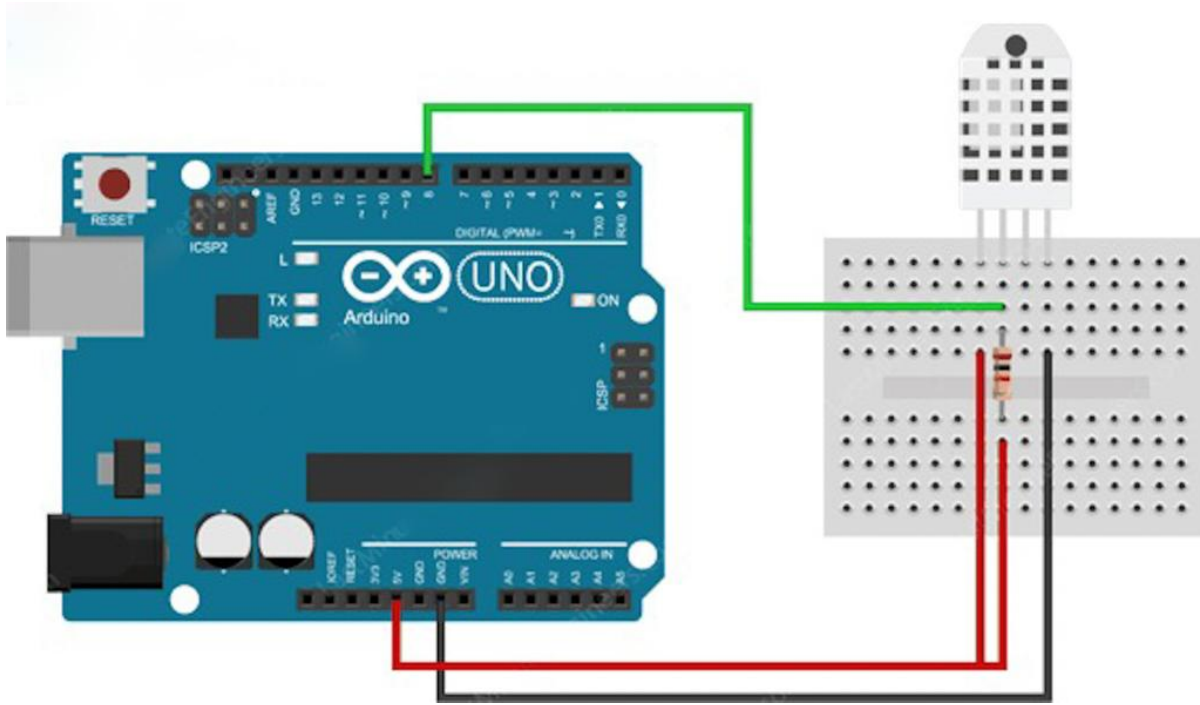
THE NEW SETUP LOOKS LIKE THIS:

1. Arduino UNO 2. 16×2 I2C LCD 3. DHT22 temperature and humidity sensor 4. wires 5. breadboard
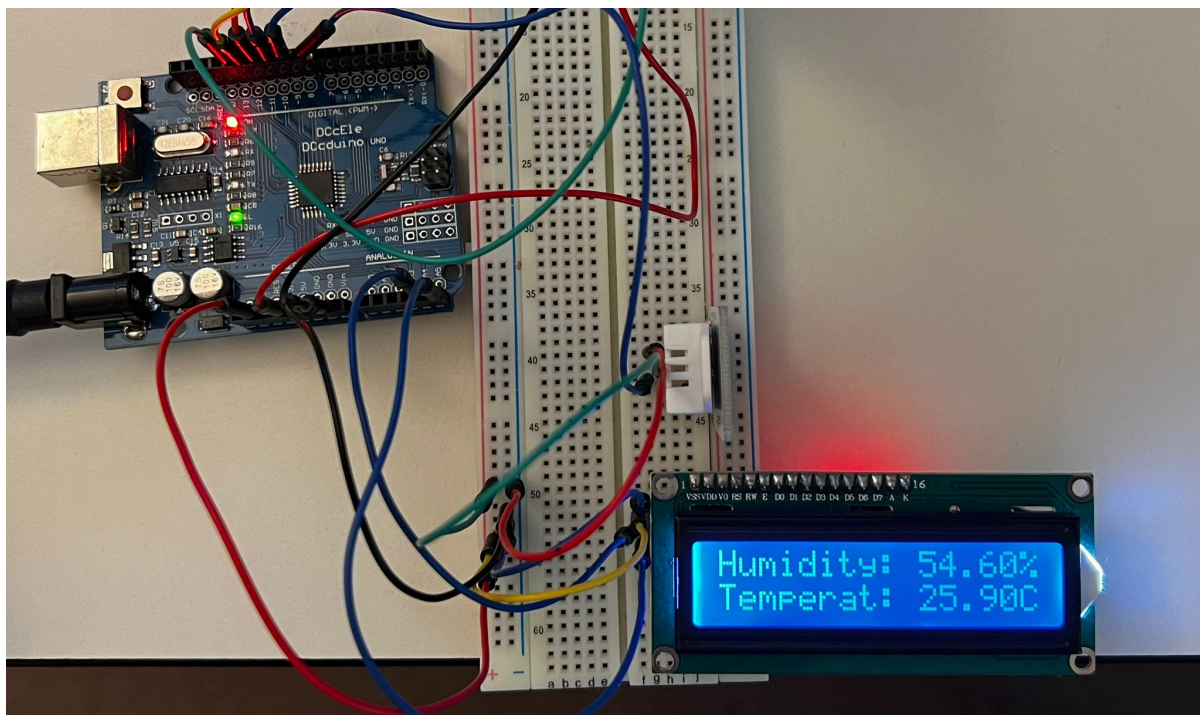
How to connect the I2C LCD to Arduino UNO:

After connecting the I2C LCD to Arduino UNO, we will **connect the DHT22 temperature and humidity sensor** too.

After connecting both the **I2C LCD** and **DHT22** temperature and humidity sensor, **the setup looks like this:**

Software Design

Libraries #include <dht.h> #include <LiquidCrystal_I2C.h> #include <Wire.h> dht DHT;

Constants #define DHT22_PIN 8 pin used for DHT22 LiquidCrystal_I2C lcd(0x27,16,2); set the LCD address to 0x27 after finding it from serial monitor for a 16 chars and 2 line display

Variables float hum; Stores humidity value float temp; *Stores temperature value*

void setup() {

```
  Serial.begin(9600);
  lcd.init();                          // initialize the  lcd
// Print a message to the LCD.
lcd.backlight();
lcd.setBacklight(HIGH);
```

}

void loop() {

```
  int chk = DHT.read22(DHT22_PIN);
  //Read data and store  it to variables hum and temp
  hum = DHT.humidity;
  temp= DHT.temperature;
  //Print temp and humidity values to LCD
  lcd.setCursor(0,0);
  lcd.print("Humidity: ");
  lcd.print(hum);
  lcd.print("%");
  lcd.setCursor(0,1);
  lcd.print("Temperat: ");
  lcd.print(temp);
  lcd.println("C");
  delay(2000); //Delay 2 sec between temperature/humidity check.
```

}

Results Obtained

I obtained a great thermometer that I can use in my house or anywhere else. You just need to plug it to a power source and it will tell you the humidity and temperature wherever you are.

Conclusions

Try using genuine arduino wifi boards, some of the other ones are not well maintained in matters of drivers. I've had problems for days with the NODEMCU ESP8266 wifi because the micro sub to usb drivers do not work and the board is simply unusable. I tried 7 different micro usb cables, tried programming it with 3 different computers with the drivers installed on each one. **NOTHING WORKED**. Checked arduino.cc forum and most people have the same issue, it's unusable. I chose to change my project a little bit so I can work it out. Instead of making it read data from sensors and transmit them to arduino IoT cloud, I made it read the data from the sensors and show it on an LCD

screen. Now it works perfectly fine, the only thing i need to find now is a power source.

I recommend informing yourself about the things you want to buy if they're not genuine since some of them have issues and **will not work**, making it a hard time for you.

Download

Jurnal

Bibliography/Resources

https://lastminuteengineers.com/dht11-dht22-arduino-tutorial/
https://lastminuteengineers.com/i2c-lcd-arduino-tutorial/ https://forum.arduino.cc/
https://www.ardumotive.com/how-to-use-dht-22-sensor-en.html
https://www.instructables.com/How-to-Connect-I2C-Lcd-Display-to-Arduino-Uno/
https://www.circuitgeeks.com/arduino-i2c-lcd-tutorial/

Export to PDF

From:
http://ocw.cs.pub.ro/courses/ - **CS Open CourseWare**

Permanent link:
http://ocw.cs.pub.ro/courses/pm/prj2023/avaduva/cloud_weather_station

Last update: **2023/05/28 13:00**