

(<https://www.dfrobot.com/product-1308.html>)

Introduction

Ever want to build your own weather station? DFRobot brings you this new version of the weather station kit that integrates anemometer, wind vane and rain gauge. All data can be read directly through the serial port. Also, it is compatible with Arduino devices, which is convenient for users to use. The weather station kit can effectively detect regional weather parameters like, wind speed, wind direction, and rainfall, or be used in agriculture, industry, and climate research fields in combination with DFRobot's other sensors.

Note:

- Update on 2016/3/28: upgrade the temperature and humidity sensor, improve the measuring range
- Update on 2020/6/18: modify the wind speed parameter description



Specification

- Operating Voltage: 3.3V
- Temperature Range: -40 - 125°C
- Humidity Range: 0~100%
- Dimension: 24 X 69 X 100mm / 0.952.723.94 inches

Application

- Weather station
- Weather monitoring

Board Overview



No.	Description
B	Rainfall
C	Wind speed and direction
D	Data output

E	Indicator light
I	I2C temperature, humidity and air pressure port

No.	Description
J1	Metric & Imperialb Jumpers
J2	Data interface 2400/9600 baud rate Jumpers

Data Output Format

```
c000s000g000t082r000p000h48b10022*3C
```

Output 37 bytes per second, including CR/LF at the end of the data.

Data Analysis:

- c000: Wind direction angle, unit: degree.
- s000: wind speed in 1 minute, unit: miles per hour
- g000: the highest wind speed in 5 minutes, unit: miles per hour
- t086: Temperature (Fahrenheit)
- r000: Rainfall in 1 hour (0.01 inches)
- p000: Rainfall in 24 hours (0.01 inches)
- h53: Humidity (00% = 99%)
- b10020: Air pressure (0.1 hpa)

Note: The board will make a hardware self-check before it works, it will output “...” when it doesn’t detect the related devices. For example, If the temperature & humidity sensor and barometer are not installed or broken, it will output: ""c000s000g000t...r000p000h..b..... ""

Indicator Light

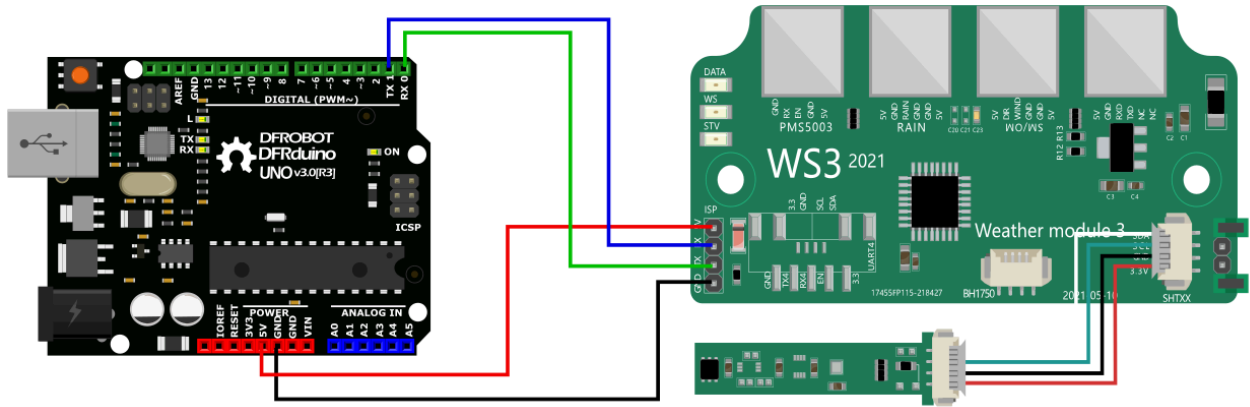
- DAT flashes synchronously when sending data
- TX flashes synchronously when sampling wind speed
- RX flashes synchronously when rain sensor is working

Tutorial

Requirements

- **Hardware**
 - DFRduino UNO R3 (<https://www.dfrobot.com/product-838.html>) (or similar) x 1
 - APRS Weather Station Kit x 1
 - M-M/F-M/F-F Jumper wires
- **Software**
 - Arduino IDE (<https://www.arduino.cc/en/Main/Software>)

Connection Diagram



Please unplug the cable on the TX&RX interface, or it will interfere with the sketch uploading.

Sample Code

```

/*!
 * @file SEN0186.ino
 * @brief DFRobot brings you this new version of the weather station kit that integrates anemometer, wind vane and rain gauge.
 * @copyright Copyright (c) 2010 DFRobot Co.Ltd (http://www.dfrobot.com)
 * @license The MIT License (MIT)
 * @author DFRobot
 * @version V1.0
 * @date 2023-08-03
 */

char          databuffer[35];
double        temp;

void getBuffer()                                //Get weather status data
{
    int index;
    for (index = 0; index < 35; index++) {
        if (Serial.available()) {
            databuffer[index] = Serial.read();
            if (databuffer[0] != 'c') {
                index = -1;
            }
        } else {
            index--;
        }
    }
}

int transCharToInt(char* _buffer, int _start, int _stop)    //char to int
{
    int _index;
    int result = 0;
    int num = _stop - _start + 1;
    int _temp[num];
    for (_index = _start; _index <= _stop; _index++) {
        _temp[_index - _start] = _buffer[_index] - '0';
        result = 10 * result + _temp[_index - _start];
    }
    return result;
}

int transCharToInt_T(char* _buffer)
{
    int result = 0;
    if (_buffer[13] == '-') {
        result = 0 - (((_buffer[14] - '0') * 10) + (_buffer[15] - '0'));
    } else {
        result = ((_buffer[13] - '0') * 100) + ((_buffer[14] - '0') * 10) + (_buffer[15] - '0');
    }
    return result;
}

int WindDirection()                                //Wind Direction
{
    return transCharToInt(databuffer, 1, 3);
}

float WindSpeedAverage()                            //air Speed (1 minute)
{
    temp = 0.44704 * transCharToInt(databuffer, 5, 7);
    return temp;
}

float WindSpeedMax()                                //Max air speed (5 minutes)
{
    temp = 0.44704 * transCharToInt(databuffer, 9, 11);
    return temp;
}

float Temperature()                                //Temperature ("C")
{
    temp = (transCharToInt_T(databuffer) - 32.00) * 5.00 / 9.00;
    return temp;
}

float RainfallOneHour()                            //Rainfall (1 hour)
{
    temp = transCharToInt(databuffer, 17, 19) * 25.40 * 0.01;
}

```

```

    temp = transCharToInt(databuffer, 17, 19) * 25.40 * 0.01;
    return temp;
}

float RainfallOneDay()                                     //Rainfall (24 hours)
{
    temp = transCharToInt(databuffer, 21, 23) * 25.40 * 0.01;
    return temp;
}

int Humidity()                                           //Humidity
{
    return transCharToInt(databuffer, 25, 26);
}

float BarPressure()                                       //Barometric Pressure
{
    temp = transCharToInt(databuffer, 28, 32);
    return temp / 10.00;
}

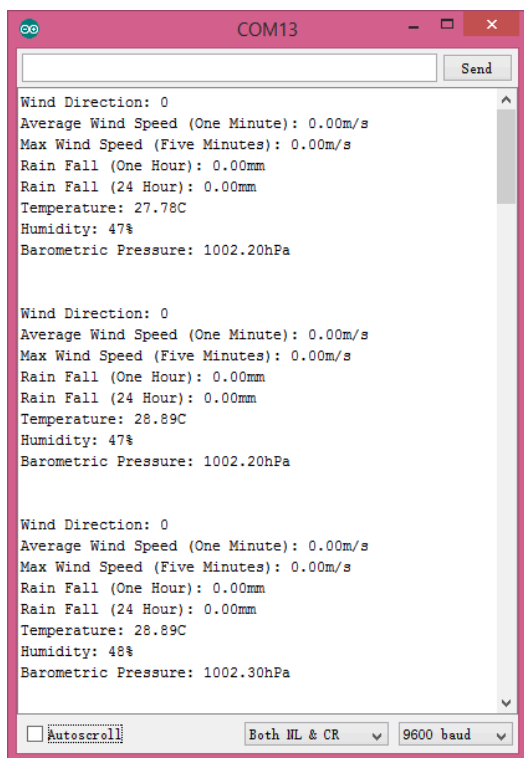
void setup()
{
    Serial.begin(9600);
}

void loop()
{
    getBuffer();                                           //Begin!
    Serial.print("Wind Direction: ");
    Serial.print(WindDirection());
    Serial.println(" ");
    Serial.print("Average Wind Speed (One Minute): ");
    Serial.print(WindSpeedAverage());
    Serial.println("m/s ");
    Serial.print("Max Wind Speed (Five Minutes): ");
    Serial.print(WindSpeedMax());
    Serial.println("m/s");
    Serial.print("Rain Fall (One Hour): ");
    Serial.print(RainfallOneHour());
    Serial.println("mm ");
    Serial.print("Rain Fall (24 Hour): ");
    Serial.print(RainfallOneDay());
    Serial.println("mm");
    Serial.print("Temperature: ");
    Serial.print(Temperature());
    Serial.println("C ");
    Serial.print("Humidity: ");
    Serial.print(Humidity());
    Serial.println("% ");
    Serial.print("Barometric Pressure: ");
    Serial.print(BarPressure());
    Serial.println("hPa");
    Serial.println("");
    Serial.println("");
}

```

Expected Results

Note: Due to the fact that there is no wind in the test environment, all parameters related to wind speed and wind direction are 0.



FAQ

Q1. About the module assembly: I only found there are **only two** installation blocks for the different modules on the Converter Board, but there are **three** modules: Anemometer, Wind vane and Rain bucket to be installed on the Converter Board. How come?


A: There are two ports under the anemometer, you need to connect the anemometer to the wind direction and then connect to the adapter board, as shown in the figure:



For any questions, advice or cool ideas to share, please visit the **DFRobot Forum** (<https://www.dfrobot.com/forum/>).

More Documents

Old Version of SEN0186 Wiki (https://wiki.dfrobot.com/Weather_Station_with_Anemometer_Wind_vane_Rain_bucket_SKU_SEN0186)

 Get APRS Weather Station Kit (<https://www.dfrobot.com/product-1308.html>) from DFRobot Store or **DFRobot Distributor**. (<https://www.dfrobot.com/distributor>)

[Turn to the Top](#)