



## Instruction Manual

# DIY Weather Station

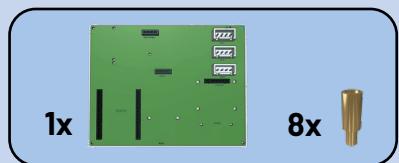
## Model v1.2



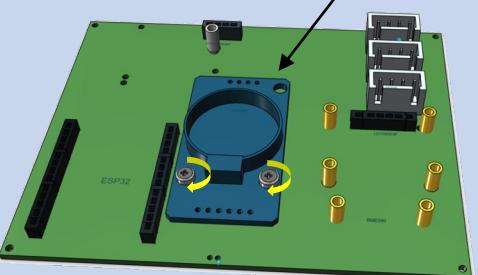
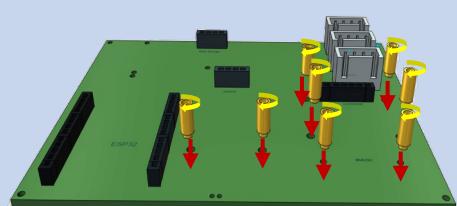
# Circuit Board Assembly



ix



1x  
2



1

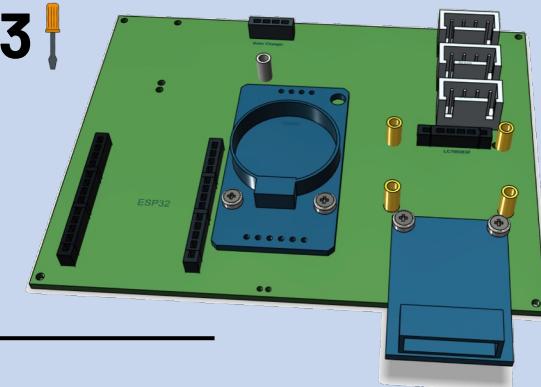


1x

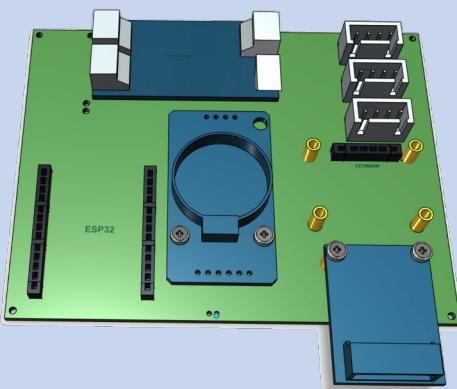


x

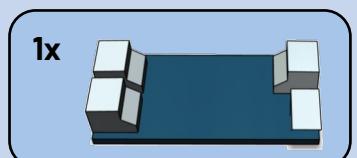
M2 (5mm)



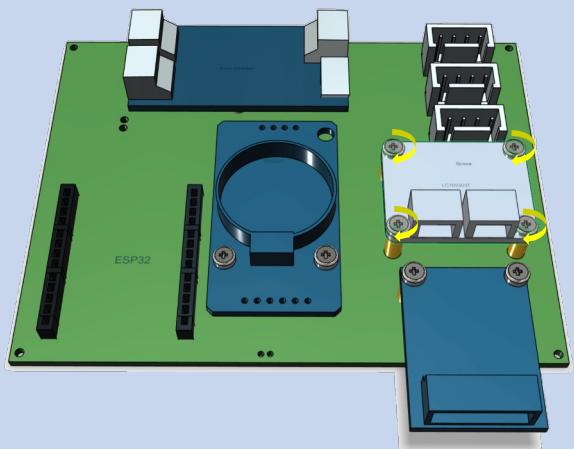
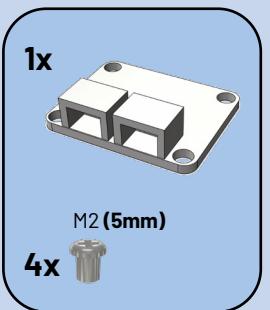
4



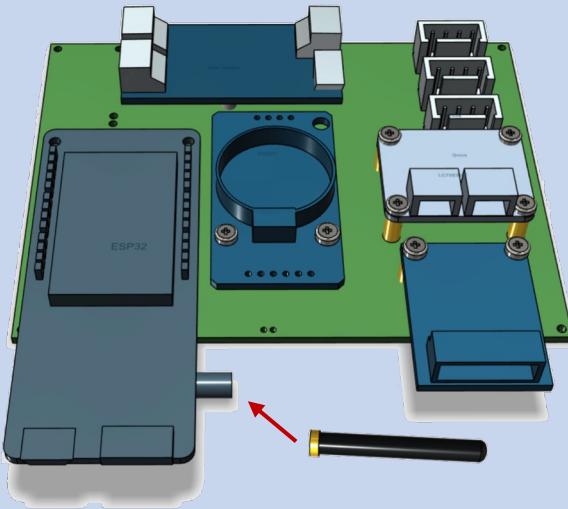
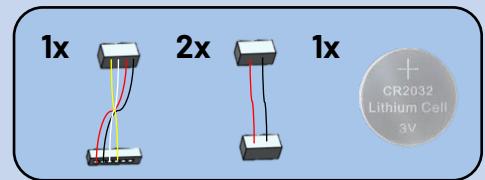
1x



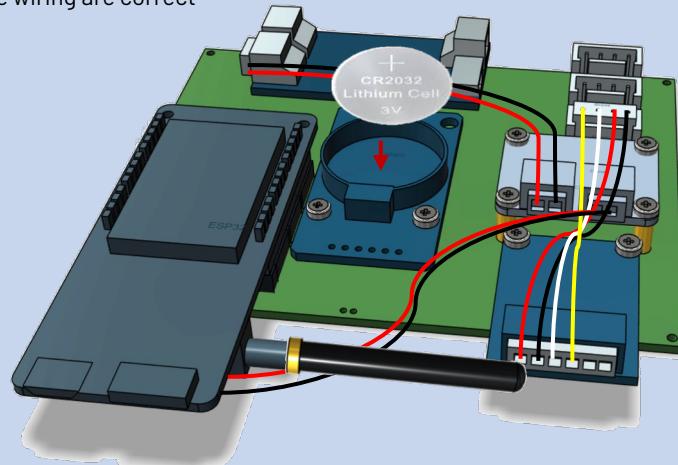
5



2

**6****7**

Please check if the wiring are correct

**3**

## Sensors Testing & Programming

### Arduino IDE and Libraries installation



#### Arduino IDE 2.0.4

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

**SOURCE CODE**  
The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

**DOWNLOAD OPTIONS**

Windows	Win 10 and newer, 64 bits
Windows	MSI installer
Windows	ZIP file
Linux	Appimage 64 bits (X86-64)
Linux	ZIP file 64 bits (X86-64)
macOS	Intel, 10.14: "Mojave" or newer, 64 bits
macOS	Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)



QR code to the software  
downloading page

Please download with link: <https://www.arduino.cc/en/software>

### 1 Install Microcontroller Chip Driver: CH9102 [https://github.com/Xinyuan-LilyGO/CH9102\\_Driver](https://github.com/Xinyuan-LilyGO/CH9102_Driver)

[Xinyuan-LilyGO / CH9102\\_Driver](#) Public

Code Issues Pull requests Actions Projects Security Insights

main 1 branch 0 tags Go to file Add file Code

LilyGO Driver upload 363899b on Apr 27, 2022 8 commits

CH9102\_WIN.EXE Driver upload last year

README.md Update README.md 2 years ago

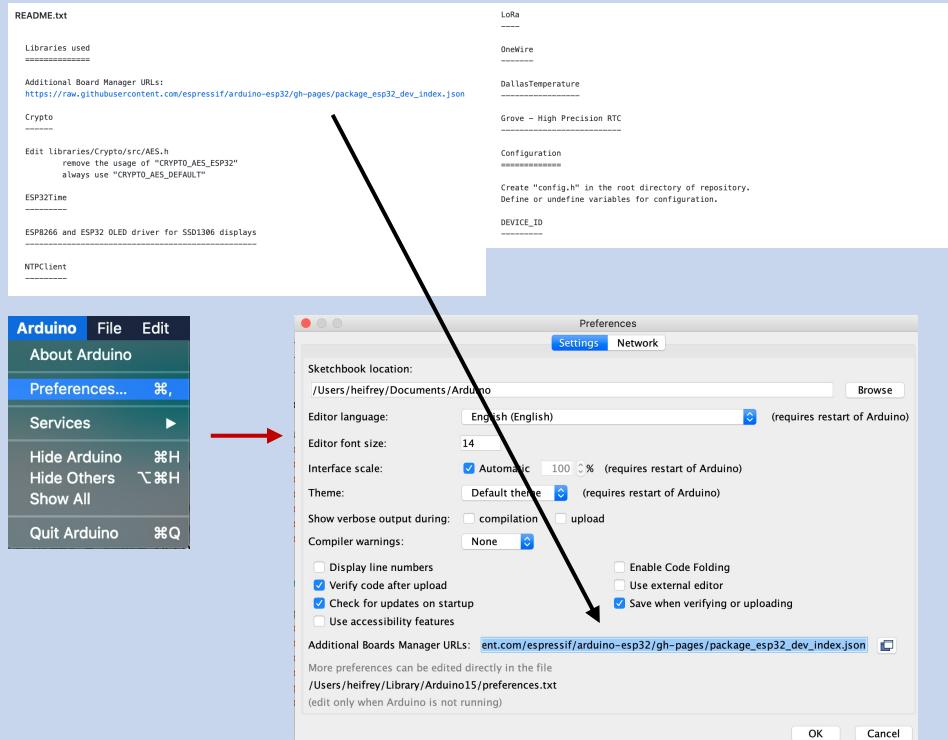
**CH9102\_Driver**

This is CH9102 windows driver file

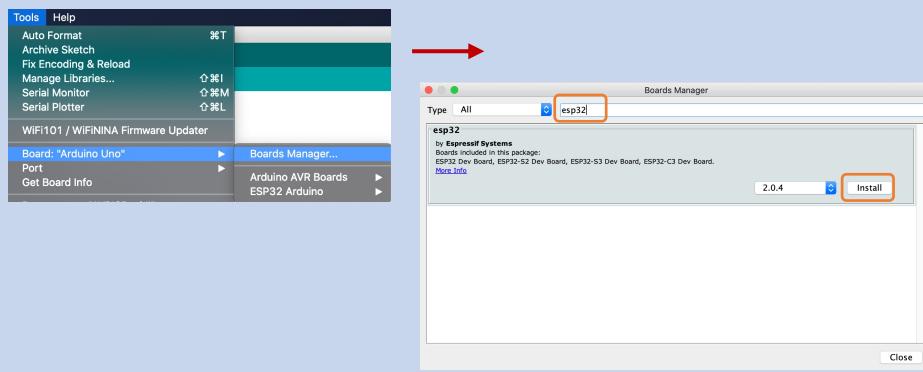
Download  
CH9102 driver  
by clicking here

**4**

## 2 Install Board Libraries with refer to Co-WIN GitHub site: <https://github.com/Cowin2020/LoRa3>

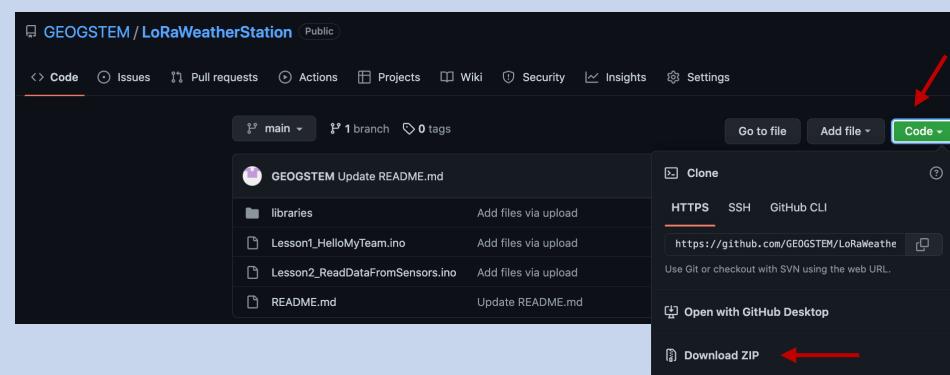


## 3 Download "ESP32" from "Boards Manager"

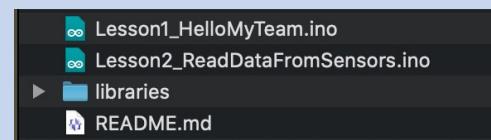


5

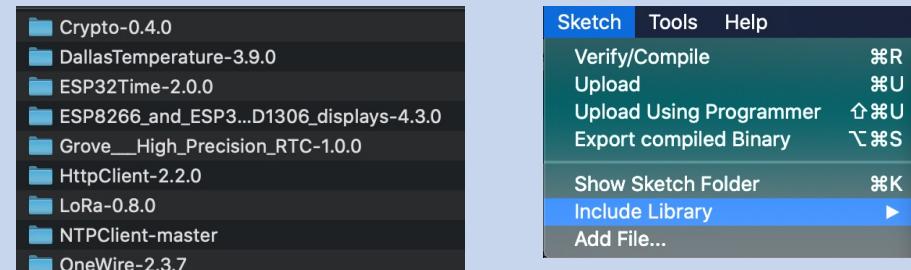
## 4 Download the material package at GeogSTEM GitHub site: <https://github.com/GEOGSTEM/LoRaWeatherStation>



## 5 Unzip the downloaded "LoRaWeatherStation-main.zip"



## 6 Unzip all the zip files in libraries and copy those folders to ~/Arduino/libraries



! You can see all the imported sensor libraries under "Include Library" after the completion

6

# Exercise 1 Open file: Lesson1\_HelloMyTeam.ino

```
Lesson1_HelloMyTeam | Arduino 1.8.13
Lesson1_HelloMyTeam
// Lesson 1: Print team name on the OLED display

#include <Arduino.h>
#include <Adafruit_SSD1306.h>

static uint8_t const OLED_WIDTH = 128;
static uint8_t const OLED_HEIGHT = 64;
static uint8_t const OLED_I2C_ADDR = 0x3C;

static Adafruit_SSD1306 display(OLED_WIDTH, OLED_HEIGHT);

void setup() {
    display.begin(SSD1306_SWITCHCAPVCC, OLED_I2C_ADDR);
    display.invertDisplay(false);
    display.setRotation(3);
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE, SSD1306_BLACK);
    display.clearDisplay();
    display.display();
}

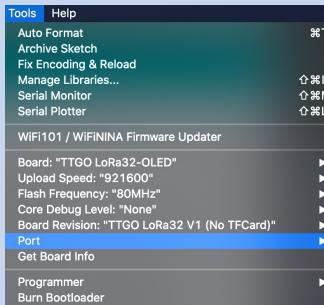
static int16_t position = 0;

void loop() {
    display.clearDisplay();
    display.setCursor(0, position);
    display.print("My Team is XXX");
    display.display();
    position = (position + 1) % OLED_WIDTH;
    delay(100); // milliseconds
}
```

This exercise is to test if the microcontroller is running well and serve as an exercise to practice how the edit script and upload to the boards.

Try to edit your team name here

Select your board "TTGO LoRa32-OLED" under Tools



Select the right USB port and connect the wire between Arduino and computer

Click to compile and upload script to Arduino board

And your team name will be shown on OLED screen after completion

# Exercise 2 Open file: Lesson2\_ReadDataFromSensors.ino

```
Lesson2_ReadDataFromSensors | Arduino 1.8.13
Lesson2_ReadDataFromSensors
// Lesson 2: get data from the sensor

#include <Arduino.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_BME280.h>

static uint8_t const OLED_WIDTH = 128;
static uint8_t const OLED_HEIGHT = 64;
static uint8_t const OLED_I2C_ADDR = 0x3C;

static Adafruit_SSD1306 display(OLED_WIDTH, OLED_HEIGHT);
static Adafruit_BME280 sensor;

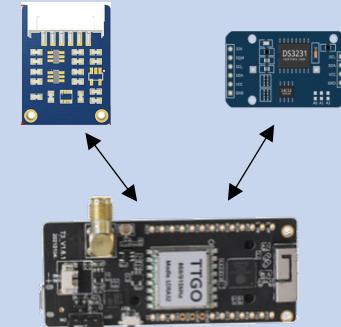
static bool ok = false;

void setup() {
    display.begin(SSD1306_SWITCHCAPVCC, OLED_I2C_ADDR);
    display.invertDisplay(false);
    display.setRotation(2);
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE, SSD1306_BLACK);
    display.clearDisplay();

    ok = sensor.begin();
    if (!ok) {
        display.println("sensor not found");
    }
}

display.display();
```

This exercise is intended to test the connection between microcontroller and different sensors.



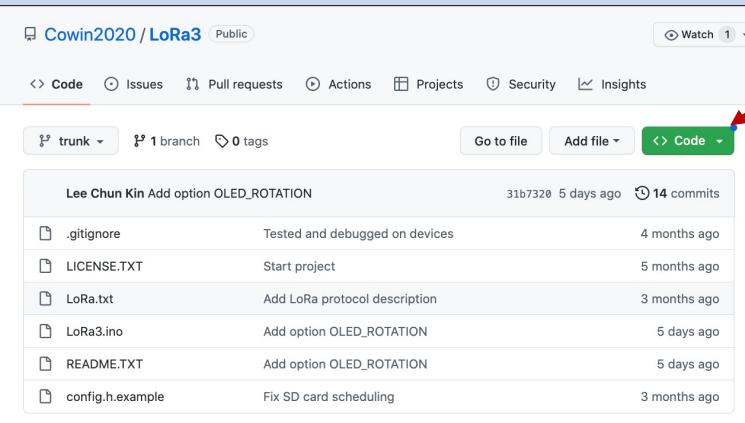
Click to compile and upload script to Arduino board

Your sensor data will be shown on OLED screen after completion

# Exercise 3

## Activate LoRa communication network

Access Co-WIN GitHub and download files:  
<https://github.com/Cowin2020/LoRa3>



A screenshot of a GitHub repository page for 'Cowin2020 / LoRa3'. The page shows a list of files and their commit history. A red arrow points from the top right towards the 'Code' dropdown menu.

**Copy and rename "config.h.example" to "config.h"**

```
1 #undef DEVICE_ID
2 #define DEVICE_ID 1 ← Edit DEVICE_ID
3 #undef NUMBER_OF_SENDERS
4 #define NUMBER_OF_SENDERS 6 ← Edit total number of devices
5
6 #define ENABLE_LED
7 #define ENABLE_COM_OUTPUT
8 #define ENABLE_OLED_OUTPUT
9 #define ENABLE_CLOCK CLOCK_DS3231
10 #define ENABLE_SD_CARD
11 #define ENABLE_BME280
12
13 #if DEVICE_ID == 0
14     #undef ENABLE_CLOCK
15 #endif
16
17 #undef WIFI_SSID
18 #undef WIFI_PASS
19 #define WIFI_SSID "My WiFi ID" ← Edit WiFi name and password
20 #define WIFI_PASS "WiFi password"
21
22 #undef HTTP_UPLOAD_FORMAT
23 #define HTTP_UPLOAD_FORMAT \
24     "http://www.example.com/REST/upload?device=%1$s&time=%3$s" \
25     "&temperature=%4$.1F&pressure=%5$.1F&humidity=%6$.1F" ← Edit server http link
26 #undef HTTP_AUTHORIZATION_TYPE
27 #define HTTP_AUTHORIZATION_TYPE "Basic"
28 #undef HTTP_AUTHORIZATION_CODE
29 #define HTTP_AUTHORIZATION_CODE "API token" ← Edit password of your LoRa network
```

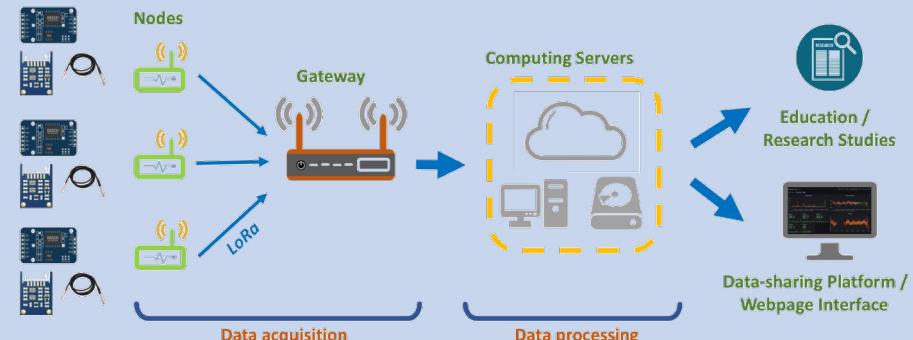
Click  to compile and upload script to Arduino board

## Edit DEVICE\_ID

- Each of the device will have their unique name i.e. Device 1, Device 2 ...

## Edit total number of devices

- How many sensors you have in the same area



## Edit WiFi name and password / Server http link

- Sensor data will be sent to internet by LoRa Gateway Therefore, we need to connect the Gateway device with WiFi

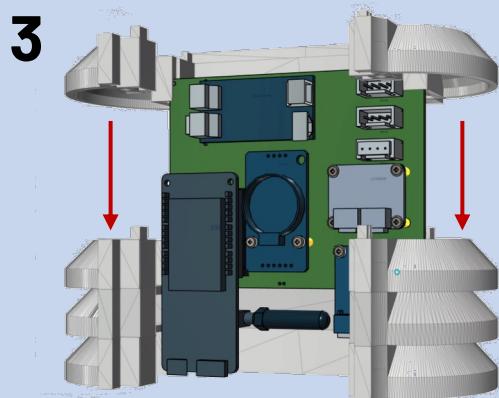
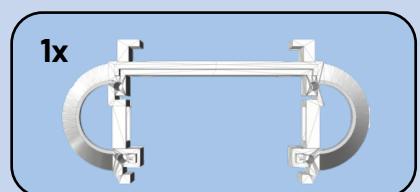
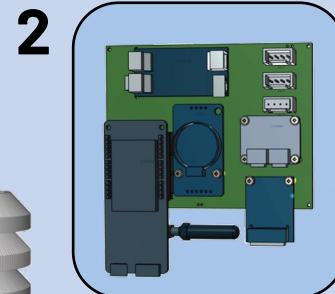
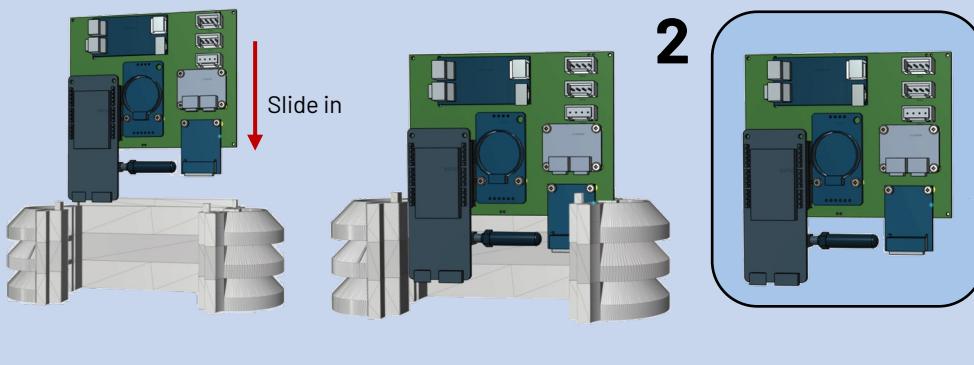
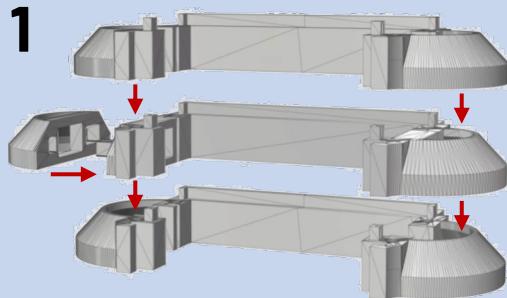
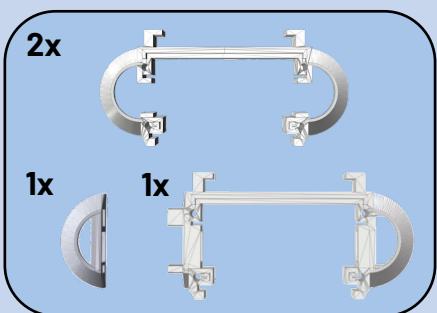
```
#define HTTP_UPLOAD_FORMAT \
    "http://www.example.com/REST/upload?device=%1$s&time=%3$s" \
    "&temperature=%4$.1F&pressure=%5$.1F&humidity=%6$.1F"
```

## Edit password of your LoRa netwrk

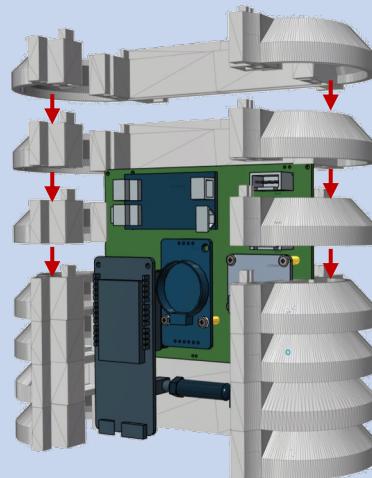
- To secure our measured data, we can setup our own network's password to prevent other people accessing our data



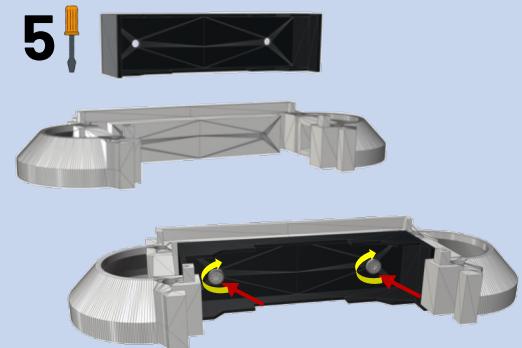
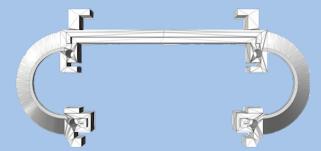
## Enclosure Building



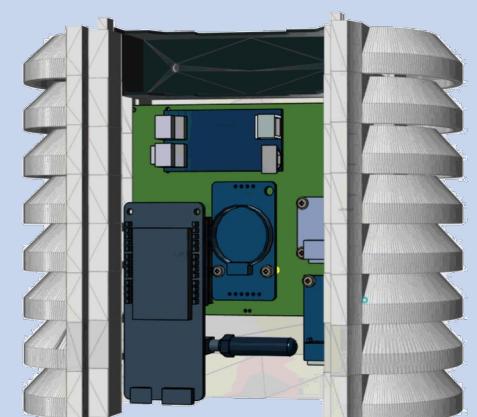
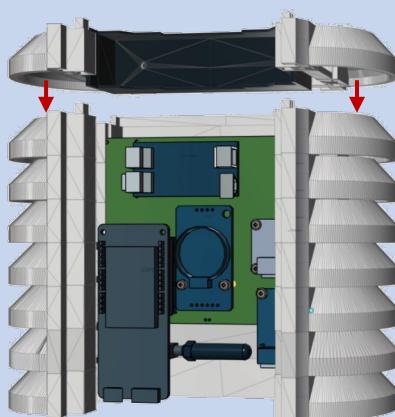
4



3x

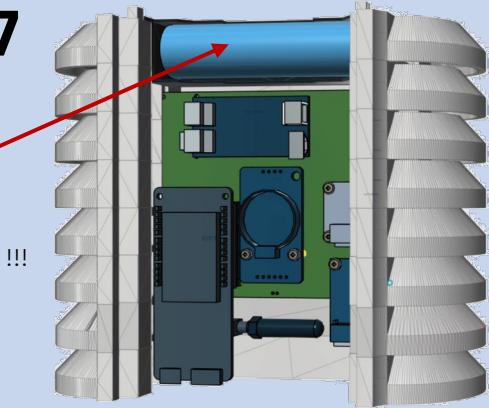


6

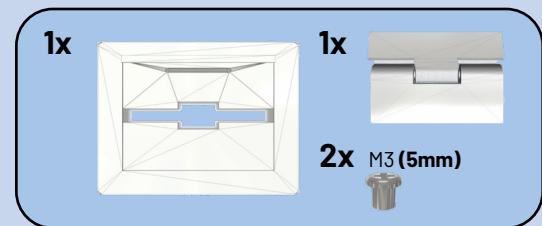




 PAY ATTENTION to the "+" and "-" pole of the battery, placing in wrong direction may lead to fire or explosion !!!



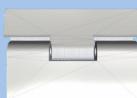
7



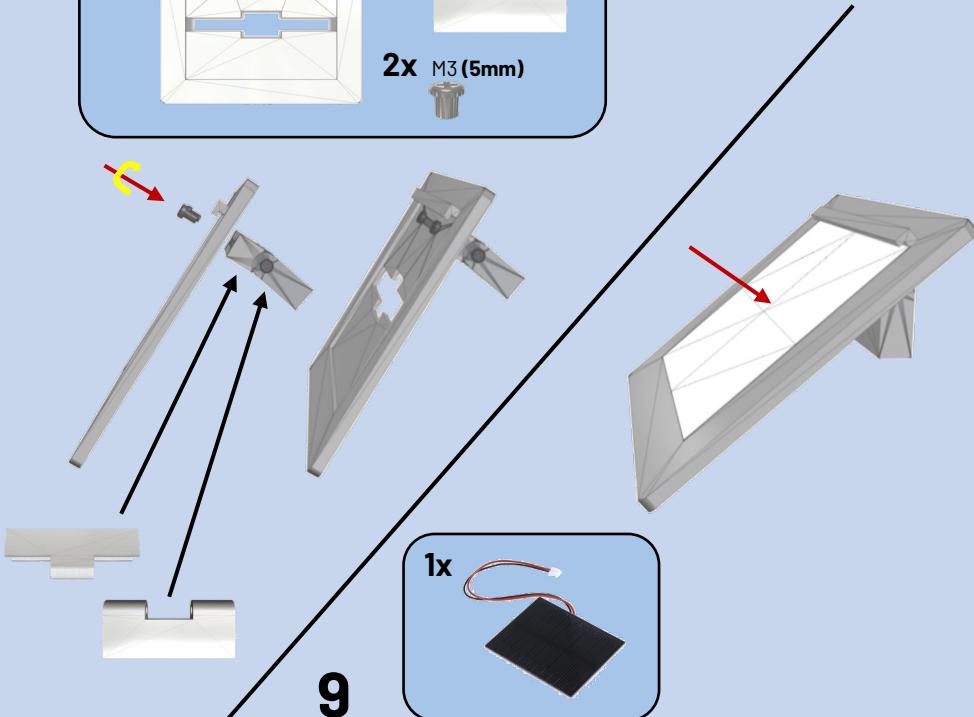
1y

1y

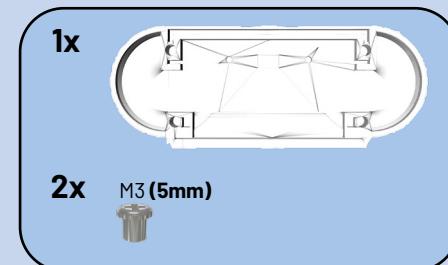
1v



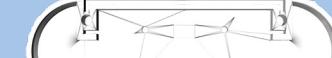
**2x M3 (5mm)**



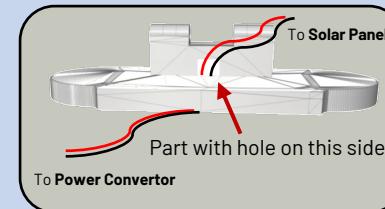
9



1

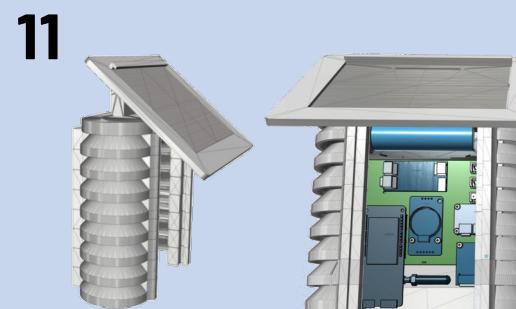


**2x M3 (5mm)**



The diagram shows a red line labeled "To Solar Panel" connecting the solar panel to the central unit. A red arrow points to the connection point on the central unit.

## Power Converters



11

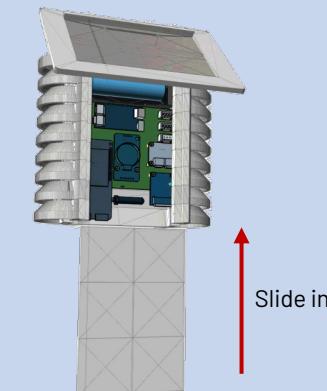
## To **Batt**

 Please check if the wiring are correct

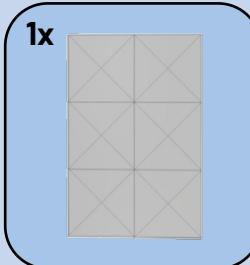
1

To Microcontrol

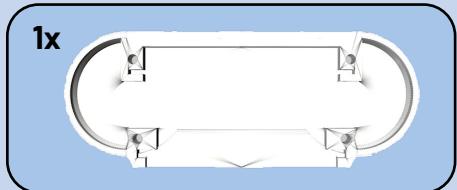
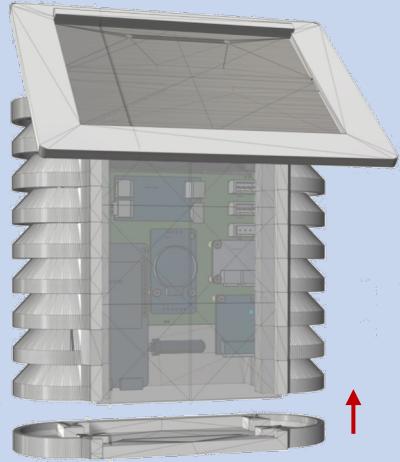
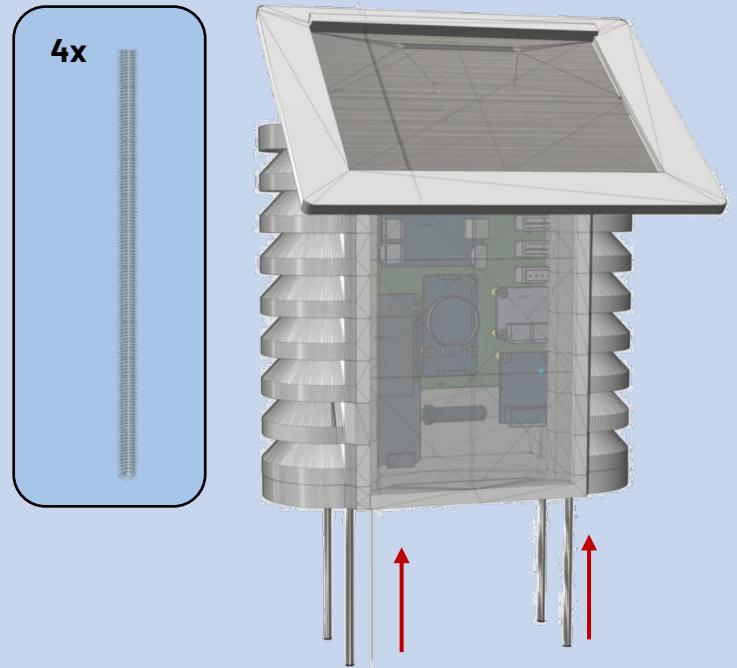
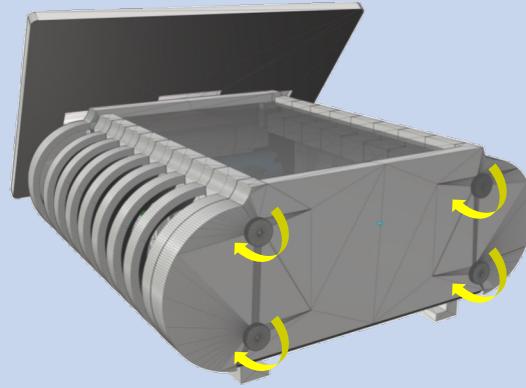
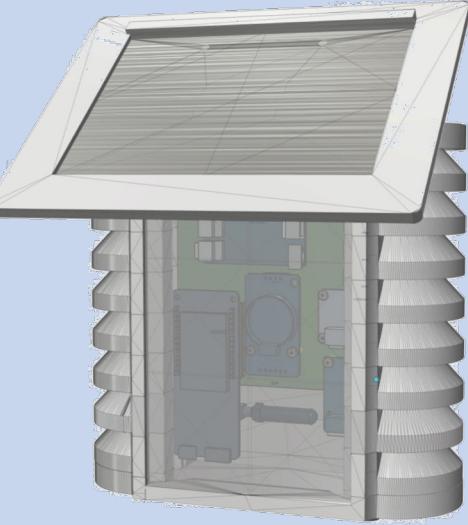
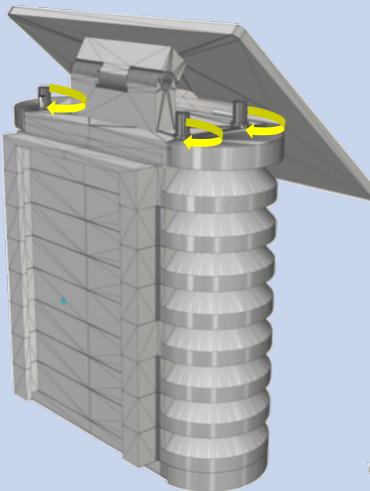
To Solar Panel



Slide in

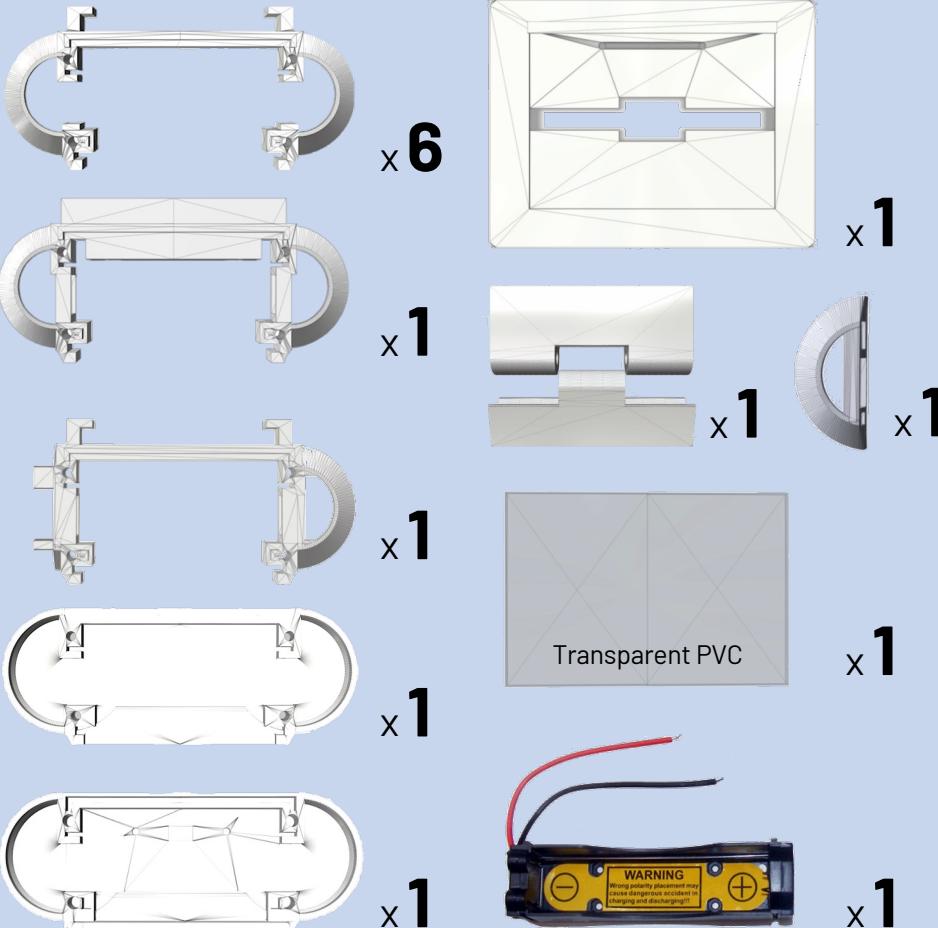


13

**13****14****14****15****16****16**



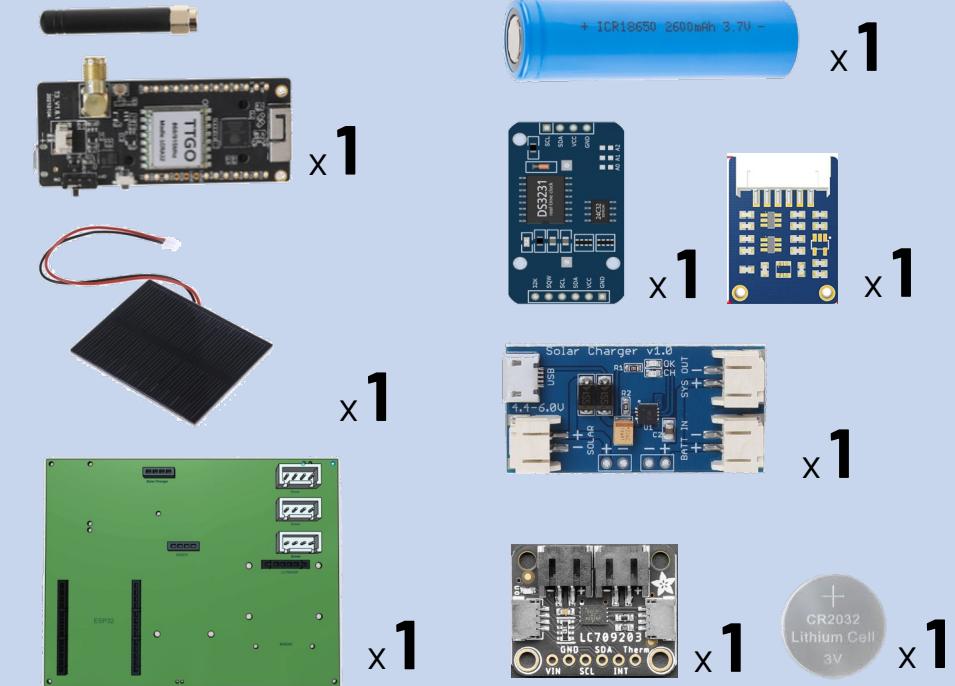
### Plastic Components:



### Wire Components :



### Electronic Components :



### Screw Components :

