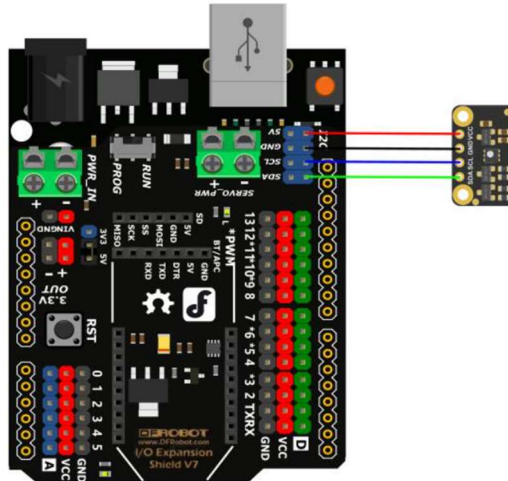


## 1. Test Sensor\_SGP40.ino

Connect the SGP40 sensor to the Arduino by referring to the following wiring diagram and upload the Sensor\_SGP40.ino from the current folder to the Arduino.



When the VOC index appears normally in the serial monitor, the sensor is normal. The sensor can be connected to a Beacon for testing.

```
Arduino Uno
Sensor_SGP40.ino  sensirion_arch_config.h  sensirion_voc_algorithm.c  sensirion_voc_algorithm.h

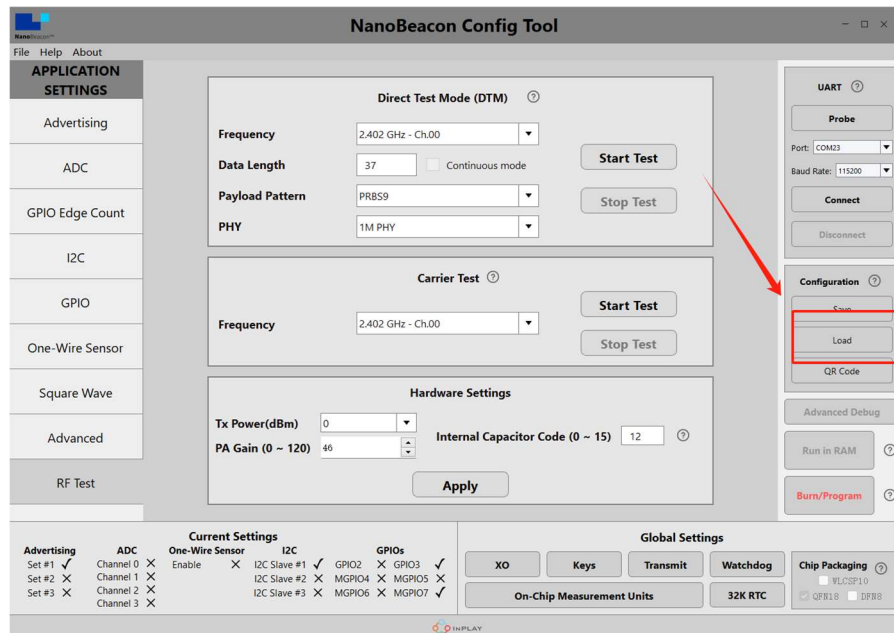
1 #include <Wire.h>
2
3 extern "C" {
4 #include "sensirion_arch_config.h"
5 #include "sensirion_voc_algorithm.h"
6 };
7
8 #define MODULE_I2C_ADDRESS ((uint8_t)0x59) // 传感器的 I2C 地址, 此处的 5x
9
10 #define TEST_OK 0x0400
11
12 #define CMD_HEATER_OFF_H 0x36
13 #define CMD_HEATER_OFF_L 0x15
14 #define CMD_HEATER_OFF_SIZE 2
15
16 #define CMD_MEASURE_TEST_H 0x28
17 #define CMD_MEASURE_TEST_L 0x0E
18 #define CMD_MEASURE_TEST_SIZE 2
19
20 #define CMD_SOFT_RESET_H 0x00
21 #define CMD_SOFT_RESET_L 0x06
22 #define CMD_SOFT_RESET_SIZE 2
23
24 #define CMD_MEASURE_RAW_H 0x26
25 #define CMD_MEASURE_RAW_L 0x0F

Output Serial Monitor x
Message (Enter to send message to 'Arduino Uno' on 'COM4')
vociIndex = 119
74 F0 B8
vociIndex = 119
75 25 B8
vociIndex = 118
75 50 B3
vociIndex = 115
75 75 C0
vociIndex = 113
75 A9 B9
vociIndex = 110
75 DC B4
vociIndex = 107
```

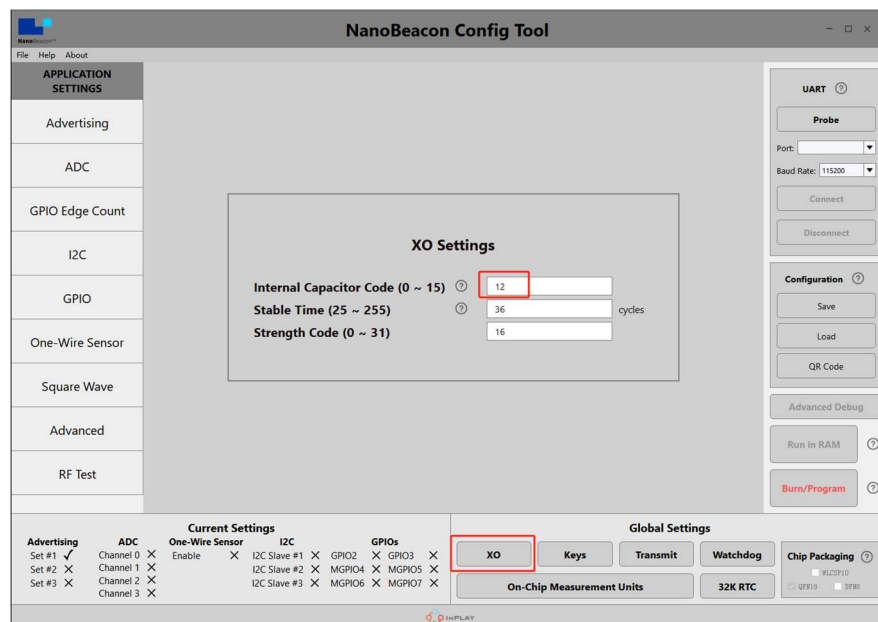
## 2. Burning Beacon and Connecting Sensors

Please use a USB-TTL converter to burn the .cfg file into the Beacon.

NanoBeacon Config Tool can Load the SGP40.cfg file in this folder.



Check that the XO capacitor configuration is 12

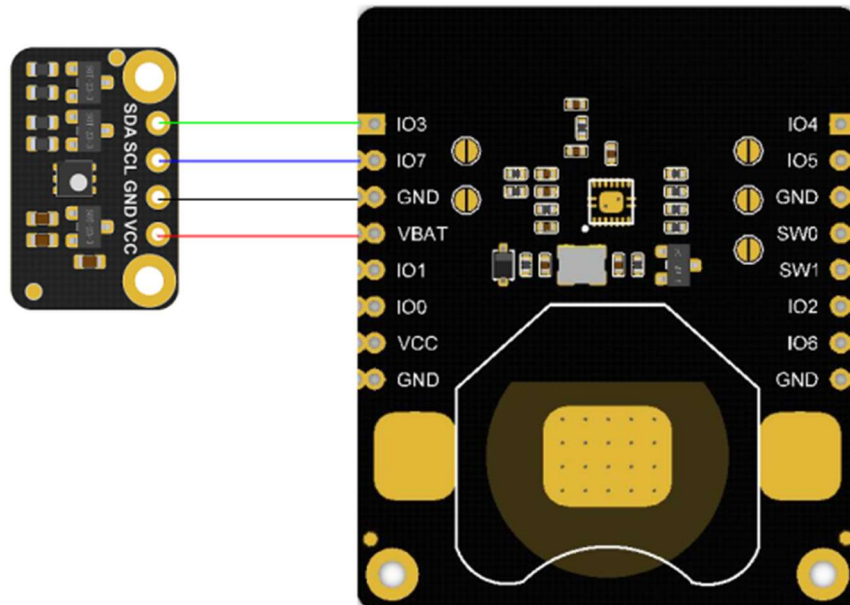


Please refer to Beacon's wiki for the burn-in process.:

[https://wiki.dfrobot.com.cn/\\_SKU\\_TEL0168\\_Fermion\\_BLE\\_%E4%BC%A0%E6%84%9F%E5%99%A8%E4%BF%A1%E6%A0%87#target\\_4](https://wiki.dfrobot.com.cn/_SKU_TEL0168_Fermion_BLE_%E4%BC%A0%E6%84%9F%E5%99%A8%E4%BF%A1%E6%A0%87#target_4)

After the burn-in is complete, refer to the diagram below to connect the Beacon and the sensor.

**Note:** Our .cfg example file defaults to SCL->GPIO7, SDA->GPIO3.



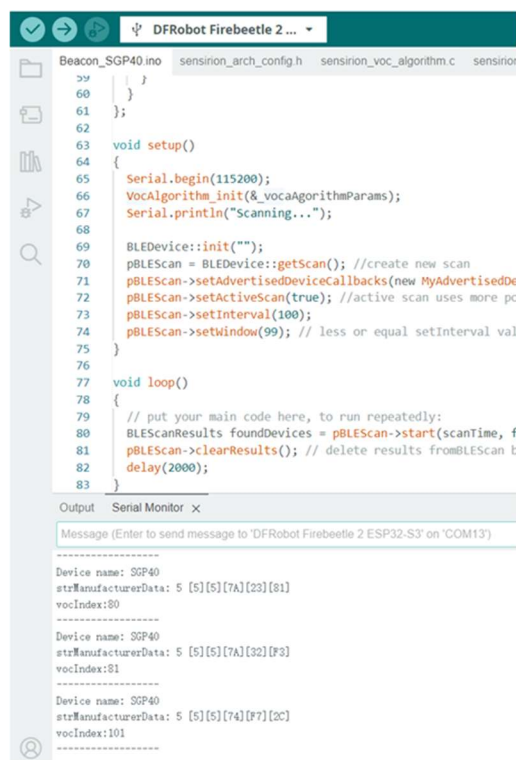
### 3. Upload ESP32 code and get readings

Upload the Beacon\_SGP40.ino in the same directory to the ESP32 motherboard.

And power up the Beacon and sensors with optional CR2032 coin cell battery, or VCC and GND input 3.3V.

You will see the relevant data printed in the serial monitor.

**Note:** Since the VOC algorithm of SGP40 needs to collect a lot of base data for calculation and Beacon\_SGP40.ino is set to fetch once in 5 seconds. So after powering up the Beacon and SGP40. You will need to wait 5-10 minutes before you can see the VOC data.



The screenshot shows the Arduino IDE interface. The top toolbar includes icons for checking, running, and uploading code, along with a dropdown menu for the board, currently set to 'DFRobot Firebeetle 2...'. The file explorer on the left shows the project files: 'Beacon\_SGP40.ino', 'sensirion\_arch\_config.h', 'sensirion\_voc\_algorithm.c', and 'sensirion\_voc\_algorithm.h'. The main editor displays the 'Beacon\_SGP40.ino' code, which includes headers, a setup function for serial communication and BLE scanning, and a loop function that starts the scanning process. The Serial Monitor at the bottom shows the output of the code, displaying three lines of data for the SGP40 sensor, each preceded by 'Device name: SGP40'. The data includes 'strManufacturerData' (a hex string), 'vocIndex', and 'vocIndex:80', '81', and '101' respectively.

```
Beacon_SGP40.ino
59 {
60 }
61 };
62
63 void setup()
64 {
65   Serial.begin(115200);
66   VocAlgorithm_init(&vocAlgorithmParams);
67   Serial.println("Scanning...");
68
69   BLEDevice::init("");
70   pBLEScan = BLEDevice::getScan(); //create new scan
71   pBLEScan->setAdvertisedDeviceCallbacks(new MyAdvertisedDe
72   pBLEScan->setActiveScan(true); //active scan uses more pc
73   pBLEScan->setInterval(100);
74   pBLEScan->setWindow(99); // less or equal setInterval val
75 }
76
77 void loop()
78 {
79   // put your main code here, to run repeatedly:
80   BLEScanResults foundDevices = pBLEScan->start(scanTime, f
81   pBLEScan->clearResults(); // delete results fromBLEScan t
82   delay(2000);
83 }
```

Output Serial Monitor x

Message (Enter to send message to 'DFRobot Firebeetle 2 ESP32-S3' on 'COM13')

Device name: SGP40  
strManufacturerData: 5 [5][5][7A][23][81]  
vocIndex:80  
-----

Device name: SGP40  
strManufacturerData: 5 [5][5][7A][32][F3]  
vocIndex:81  
-----

Device name: SGP40  
strManufacturerData: 5 [5][5][74][F7][2C]  
vocIndex:101  
-----