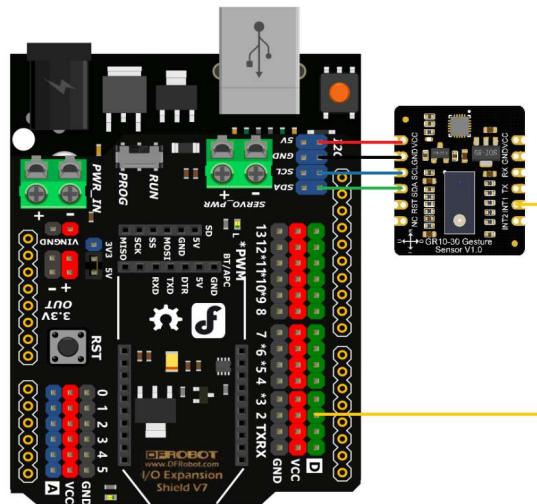


1. 测试 Sensor_GR10_30.ino

将 GR10_30 传感器参考如下连线图连接至 Arduino，并且上传当前文件夹中的 Sensor_GR10_30.ino 至 Arduino



当串口监视器中正常读数，说明传感器正常。传感器可以被连接至 Beacon 测试。

```
Arduino Uno
Sensor_GR10_30.ino
53 uint16_t state = ((uint16_t)0);
54 uint16_t gestures = ((uint16_t)0);
55 if (state) {
56     if (gestures & GESTURE_UP) {
57         Serial.println("Up");
58     }
59     if (gestures & GESTURE_DOWN) {
60         Serial.println("Down");
61     }
62     if (gestures & GESTURE_LEFT) {
63         Serial.println("Left");
64     }
65     if (gestures & GESTURE_RIGHT) {
66         Serial.println("Right");
67     }
68     if (gestures & GESTURE_FORWARD) {
69         Serial.println("Forward");
70     }
71 }
```

Output Serial Monitor x

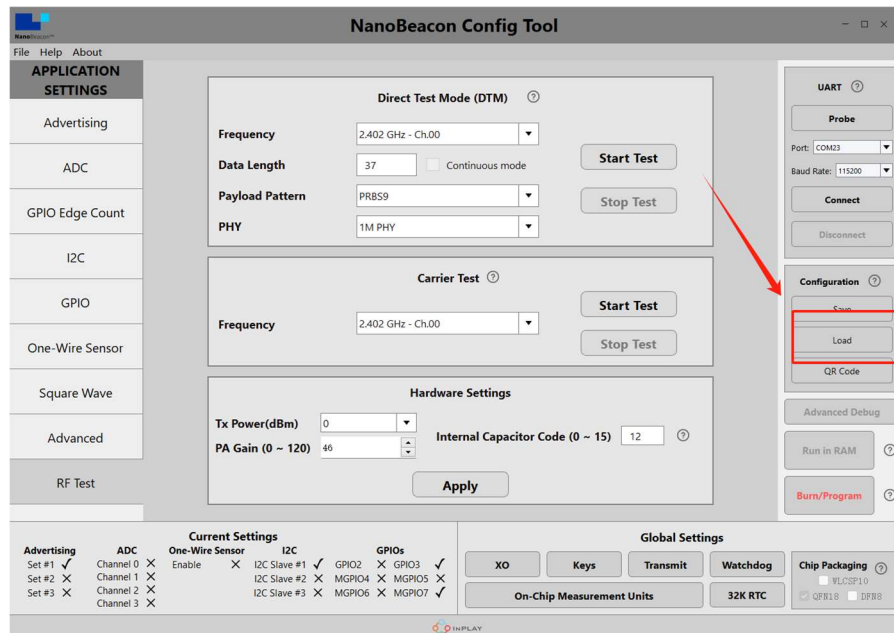
Message (Enter to send message to 'Arduino Uno' on 'CC

Right
Right
Clockwise
Continuous clockwise
Continuous clockwise
Continuous clockwise
Continuous clockwise
Continuous clockwise
Continuous clockwise
Continuous clockwise
Continuous counter-clockwise
Continuous clockwise
Continuous counter-clockwise
Continuous counter-clockwise
Continuous counter-clockwise
Continuous counter-clockwise
Down
Up
Right
Left
Down

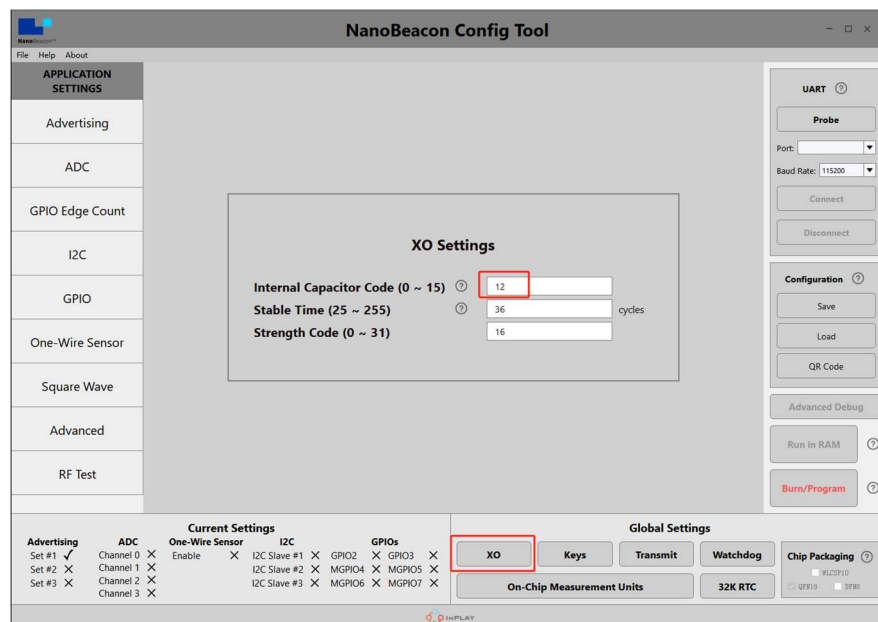
2. 烧录 Beacon 并且连接传感器

请您使用 USB-TTL 转换器将.cfg 文件烧录进 Beacon。

NanoBeacon Config Tool 中可以 Load 本文件夹中的 GR10_30.cfg 文件。



检查 XO 电容配置是否为 12

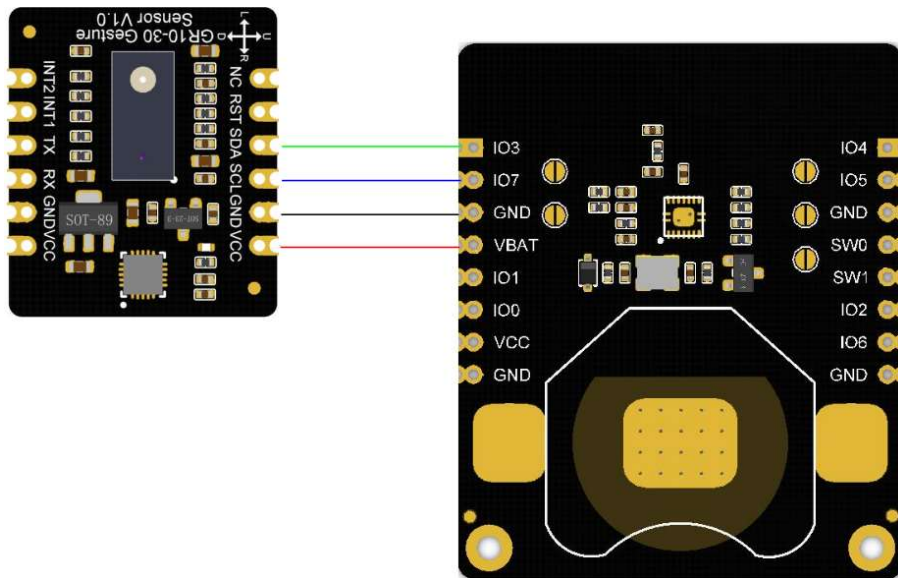


烧录流程请参考 Beacon 的 wiki:

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

在烧录完成后，参考下图连接 Beacon 和传感器。

注：我们的.cfg 示例文件默认 SCL->GPIO7, SDA->GPIO3。



3. 上传 ESP32 代码并获取读数

将同目录下的 Beacon_GR10_30.ino 上传至 ESP32 主板。

并且将 Beacon 和传感器供电，供电方式可选 CR2032 纽扣电池，或者 VCC 和 GND 输入 3.3V。

您将会看到串口监视器中打印相关数据。

The screenshot displays the Arduino IDE 2.2.1 interface. The top menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. The 'Boards Manager' tab is active, showing a list of boards. The 'FireBeetle ESP32' board is selected, with version 0.2.1 installed. The 'Code' tab shows a sketch named 'Beacon_GR10_30.ino' with the following code:

```
108 pBLEScan->setAdvertisedDeviceCallbacks(new pBLEScan->getDeviceCallback()); // active scan
109 pBLEScan->setActiveScan(true); // active scan
110 pBLEScan->setInterval(100);
111 pBLEScan->setWindow(99); // less or equal to interval
112 }
113 void loop()
114 {
115     // put your main code here, to run repeatedly
116     BLEScanResults foundDevices = pBLEScan->scan();
117     pBLEScan->clearResults(); // delete results
118     delay(2000);
119 }
```

The 'Serial Monitor' tab is open, showing the output of the sketch. The output is as follows:

```
strManufacturerData: 6 [5] [5] [0] [0] [0] [0]
Device name: GR10_30
strManufacturerData: 6 [5] [5] [0] [0] [0] [0]
Device name: GR10_30
strManufacturerData: 6 [5] [5] [0] [0] [0] [0]
Device name: GR10_30
strManufacturerData: 6 [5] [5] [0] [1] [0] [1]
Up
Device name: GR10_30
strManufacturerData: 6 [5] [5] [0] [1] [0] [8]
Right
Device name: GR10_30
strManufacturerData: 6 [5] [5] [0] [1] [0] [4]
Left
Device name: GR10_30
strManufacturerData: 6 [5] [5] [0] [0] [0] [4]
Device name: GR10_30
strManufacturerData: 6 [5] [5] [0] [1] [0] [4]
Left
```