



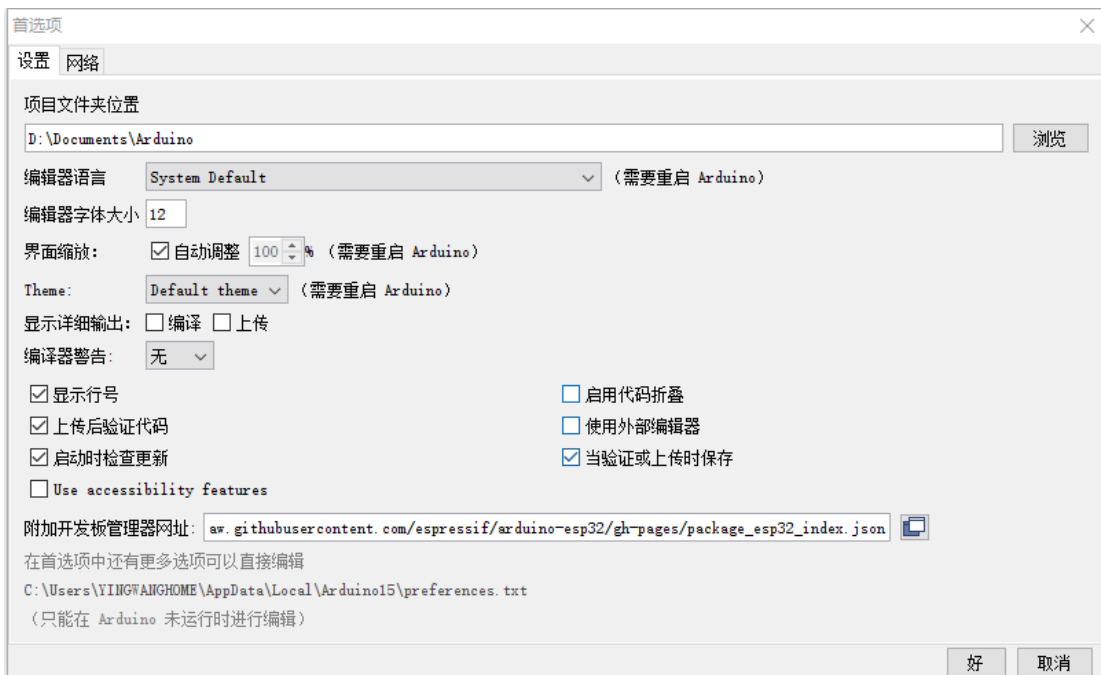
迎加入开源智能四足机器人交流群（QQ 群号: 932148）

ESP32 开发板 ARDUINO 环境配置

要使用 ARDUINO 环境给 ESP32 编程：

- 1、首先需要给 ARDUINO 配置 ESP32 环境，在 ARDUINO 的文件菜单栏下，找到首选项，打开后出现如下界面，在附加开发板管理器网址加上如下网址：

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json,



点击 OK(好)，然后退出 ARDUINO 重新打开程序，这一步很重要，很多人说找到 ESP32 就是因为没有重启程序。

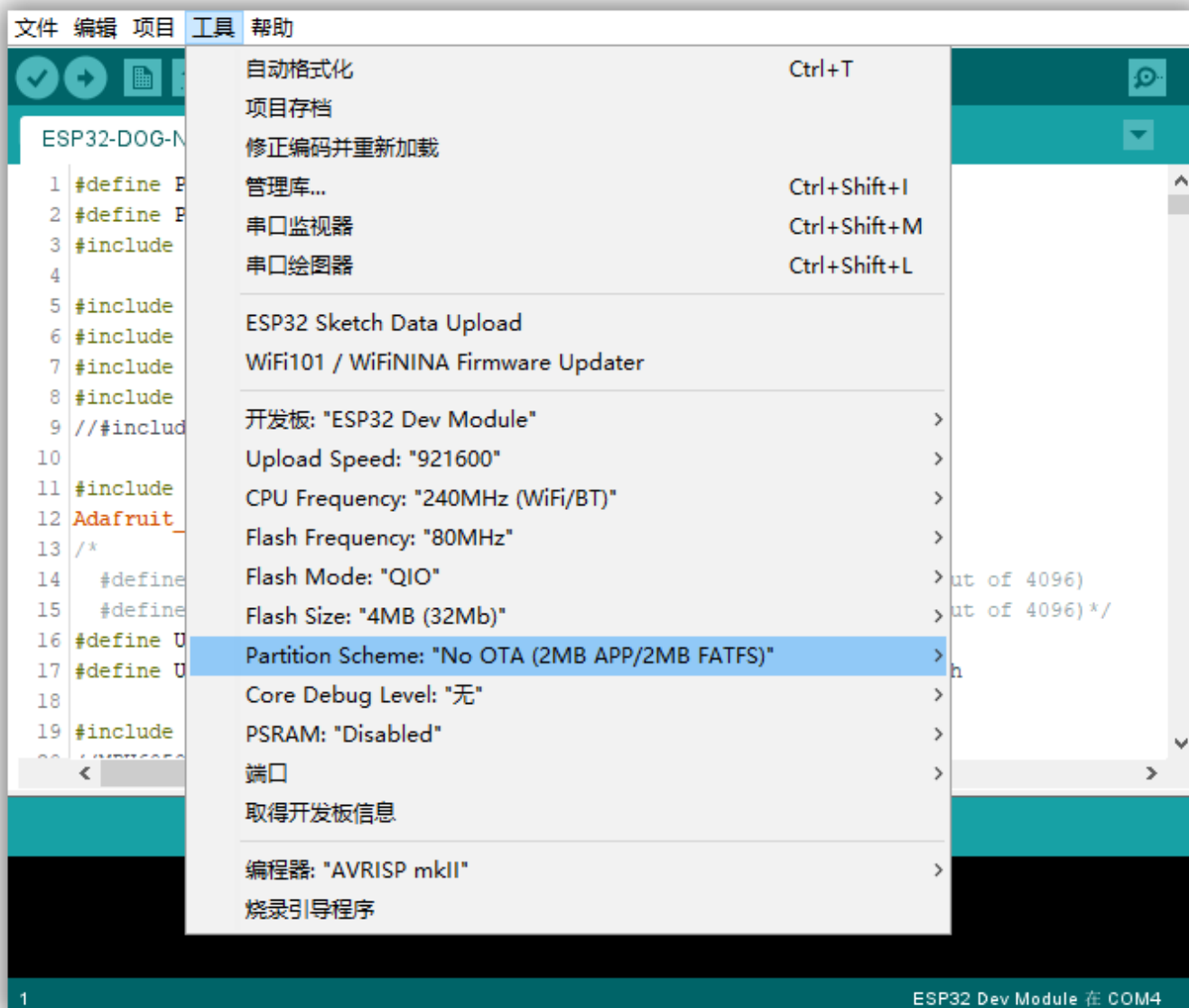
- 2、在工具菜单栏下，打开开发板——>开发板管理器，在搜索框中输入 ESP32,搜索到后安装。

欢迎加入开源智能四足机器人交流群 (QQ 群号: 932148)



- 3、至此，我们就可以利用熟悉的 ARDUINO 给 ESP32 编程了。在上传我程序到开发板时，只需要在工具->开发板下选中 ESP Dev Module,串口选择对应的端口，其他默认就可以上传了。
- 4、如果程序过大，使用默认分区烧录不进，我们可以改变分区，使用 HUG APP 分区，此时有 3M 的空间可供程序使用。其他分区模式可以自行 Google 说明。

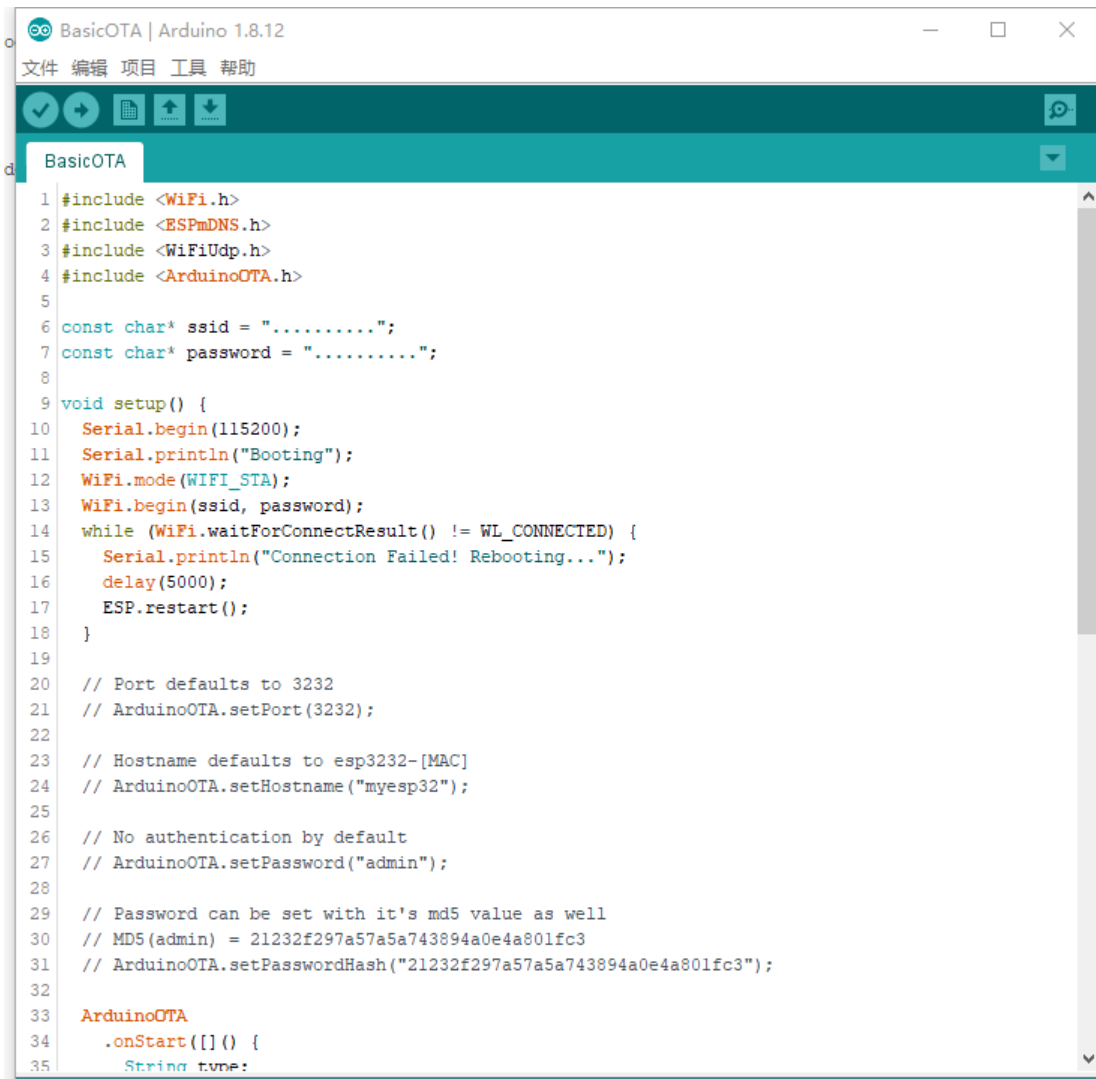
欢迎加入开源智能四足机器人交流群（QQ 群号: 932148）



- 5、在上传程序过程中，当出现 ...时，按住 BOOT 按钮，当出现上传进度时就可以松开。当出现上传完成里，我们的程序就被上传到 ESP32 开发板。
- 6、使用 ESP32 我们一定要试一下 OTA 功能，使我们的 ESP 连接到局域网，我们在上传程序或是更新时，只要端口选择带 ip 地址的 ESP32 开发板，我们就可以上传程序。具体可以参考使用

欢迎加入开源智能四足机器人交流群（QQ 群号: 932148）

ARDUINO 自带的例程，是不是很方便呢？

A screenshot of the Arduino IDE interface. The title bar reads 'BasicOTA | Arduino 1.8.12'. The menu bar includes '文件', '编辑', '项目', '工具', and '帮助'. The toolbar shows icons for opening files, saving, and uploading. The main editor window displays the 'BasicOTA' sketch. The code includes headers for WiFi, ESPmDNS, WiFiUdp, and ArduinoOTA. It defines SSID and password constants, a setup function for serial and WiFi, and an ArduinoOTA onStart callback. The code is as follows:

```
1 #include <WiFi.h>
2 #include <ESPmDNS.h>
3 #include <WiFiUdp.h>
4 #include <ArduinoOTA.h>
5
6 const char* ssid = ".....";
7 const char* password = ".....";
8
9 void setup() {
10   Serial.begin(115200);
11   Serial.println("Booting");
12   WiFi.mode(WIFI_STA);
13   WiFi.begin(ssid, password);
14   while (WiFi.waitForConnectResult() != WL_CONNECTED) {
15     Serial.println("Connection Failed! Rebooting...");
16     delay(5000);
17     ESP.restart();
18   }
19
20   // Port defaults to 3232
21   // ArduinoOTA.setPort(3232);
22
23   // Hostname defaults to esp3232-[MAC]
24   // ArduinoOTA.setHostname("myesp32");
25
26   // No authentication by default
27   // ArduinoOTA.setPassword("admin");
28
29   // Password can be set with it's md5 value as well
30   // MD5(admin) = 21232f297a57a5a743894a0e4a801fc3
31   // ArduinoOTA.setPasswordHash("21232f297a57a5a743894a0e4a801fc3");
32
33   ArduinoOTA
34     .onStart([]() {
35       String type;
```