

Home Assistant ESP8266 自制

WiFi 控制 GPIO

Phodal Huang

September 8, 2017

目录

步骤 1: 环境准备 3

步骤 2: 现在为 ESP8622 安装 Arduino 软件 3

步骤 3: ESP8266 WiFi 网络的代码 4

步骤 4: ESP8266 接线 6

步骤 5: 最后的想法和视频 Demo 8

玩点什么: <https://www.wandianshenme.com>

原文链接:<https://www.wandianshenme.com/play/home-assistant-control-custom-wifi-gpio>

你使用 HomeAssistant?

你要创建自己的 WiFi 可控制的设备吗?

如果您对这两个问题回答“是”，那么您需要好好阅读这篇文章了。在这里，您将了解如何使用：

- 启用 ESP8622 WiFi 功能的设备来加入本地 WiFi 网络
- 通过在 Raspberry Pi 3 上运行的 Home Assistant 来交换对 ESP8266 板载 GPIO 的控制

步骤 1: 环境准备

首先，您将需要设置您的 ESP8622 WiFi 模块。你需要的东西有：

- ESP8622 WiFi 模块
- USB 转串行 FTDI 连接器

好消息是，这里的内容是 Arduino 兼容的。从这个链接可以下载 Arduino 开发环境：<https://www.arduino.cc/en/Main/Software>，并安装 Arduino 软件。

步骤 2: 现在为 ESP8622 安装 Arduino 软件

现在，安装完毕后，您将需要启动您的 Arduino 软件。

然后打开：Tools -> Board -> Boards Manager

您将需要在 Boards Manager (开发板管理器) 的搜索字段中键入 ESP，然后按 enter 键，来下载 ESP8622 的开发板定义。选择一个标签为 ESP8266 的选项。一旦选择，选择右下角的“install (安装)”，这将安装自定义的开发板。

完成安装后，您可以返回到 Tools -> Boards，然后选择 Generic ESP8266 开发板。

现在你需要安装一个最后的东西，那就是这个库，所以你可以编程。要做到这一点，你需要去 Sketch -> Include Library -> Manage Libraries。

一旦在那里，您将通过输入 ESP 进行搜索，然后选择与 ESP8266WiFi.h 相关联的一个来执行库。

一旦安装完毕，您现在可以开始编程了。

步骤 3: **ESP8266 WiFi** 网络的代码

现在您将需要构建代码。

代码如下所示。您唯一需要更改的是网络的 **SSID** 和密码的变量。不要担心此步骤中的 **MQTT** 变量，它将在后面介绍。

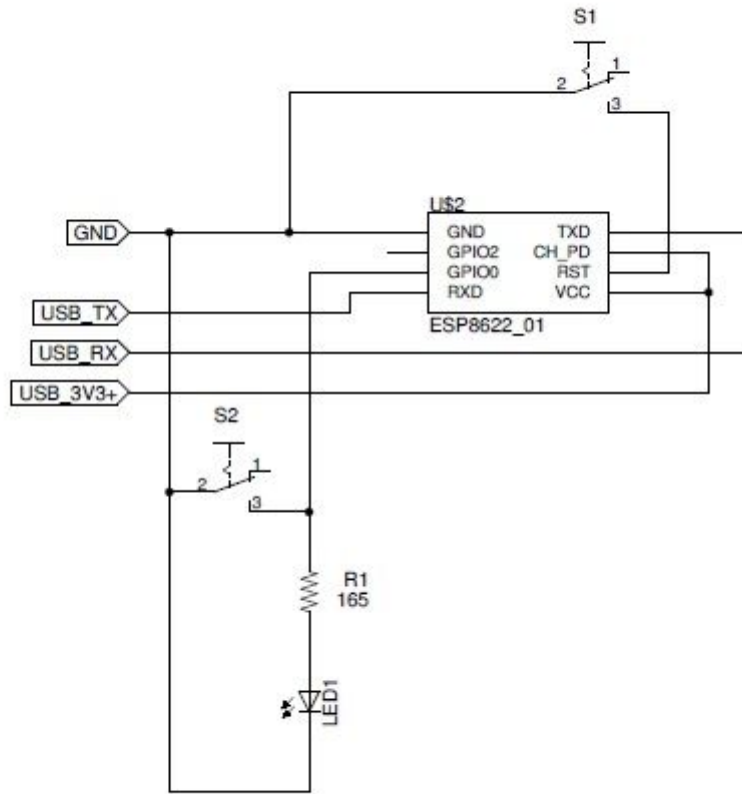
```
1 #include <ESP8266WiFi.h>
2 #include <PubSubClient.h>
3
4 // Update these with values suitable for your network.
5 const char* ssid = "";
6 const char* password = "";
7 const char* mqtt_server = "";
8
9 WiFiClient espClient;
10 PubSubClient client(espClient);
11 int SwitchedPin = 0;
12 String switch1;
13 String strTopic;
14 String strPayload;
15
16 void setup_wifi() {
17   Serial.begin(115200);
18   delay(100);
19
20   // We start by connecting to a WiFi network
21
22   Serial.println();
23   Serial.println();
24   Serial.print("Connecting to ");
25   Serial.println(ssid);
26
27   WiFi.begin(ssid, password);
28
29   while (WiFi.status() != WL_CONNECTED) {
30     delay(500);
31     Serial.print(".");
```

```
32 }
33 Serial.println(""); Serial.println("WiFi connected");
34 Serial.println("IP address: ");
35 Serial.println(WiFi.localIP());
36 }
37
38 void callback(char* topic, byte* payload, unsigned int length) {
39   payload[length] = '\0';
40   strTopic = String((char*)topic);
41   if(strTopic == "ha/switch1")
42     {
43     switch1 = String((char*)payload);
44     if(switch1 == "ON")
45       {
46       Serial.println("ON");
47       digitalWrite(SwitchedPin, HIGH);
48       }
49     else
50       {
51       Serial.println("OFF");
52       digitalWrite(SwitchedPin, LOW);
53       }
54     }
55 }
56
57
58 void reconnect() {
59   // Loop until we're reconnected
60   while (!client.connected()) {
61     Serial.print("Attempting MQTT connection...");
62     // Attempt to connect
63     if (client.connect("arduinoClient")) {
64       Serial.println("connected");
65       // Once connected, publish an announcement...
66       client.subscribe("ha/#");
67     } else {
```

```
68     Serial.print("failed, rc=");
69 Serial.print(client.state());      Serial.println(" try again in 5
        seconds");
70     // Wait 5 seconds before retrying
71     delay(5000);
72 }
73 }
74 }
75
76 void setup()
77 {
78     setup_wifi();
79     client.setServer(mqtt_server, 1883);
80     client.setCallback(callback);
81
82     pinMode(SwitchedPin, OUTPUT);
83     digitalWrite(SwitchedPin, LOW);
84 }
85
86 void loop()
87 {
88     if (!client.connected()) {
89         reconnect();
90     }
91     client.loop();
92 }
```

步骤 4: ESP8266 接线

现在您需要连接设备。



我已经提供了上面的图表，你可以看看如何连接它，并做一些适合的修改。

当你连接了 RS232 到 USB 转换器，你需要将 GPIO #0 引脚接地。这样当你重置设备时，可以将器件上的 RST 引脚接地，使其保持复位状态，然后将其引导到引导程序模式。

在这里，你需要选择对应的开发板，并进行编程。这个时候就需要从菜单上选择：
Tools -> Board -> Generic ESP8266 Module

再往下，您将需要查看您的 USB 转串口转换器的 COM 端口，并通过以下方式选择：
Tools -> Port

如果您无法找到 USB 转 FTDI 设备的 COM 端口，只需在 Windows 机器上查看设备管理器，或者在您的 Linux 设备上执行 `ls usb`。

一旦你完成所有设置，只需要在你的 Arduino IDE 软件中按下载（upload）。

步骤 5: 最后的想法和视频 **Demo**

Demo 地址: <https://www.youtube.com/watch?v=aWWtbCAXGPI>

我希望您喜欢使用本教程让您的 **ESP8266** 设备启动并运行。我已经收录在我的 **YouTube** 频道中, 以帮助您直观地展示: 如何进行全部设置和运行。

其中还包括如何设置 **MQTT**, 以便在 **ESP8266** 和 **Home Assistant** 之间发送 **IoT** 命令。

Enjoy play it!。

原文链接: <https://www.wandianshenme.com/play/home-assistant-control-custom-wifi-gpio>