**

**项目报告**

**课程名称**  IT技术创新项目实践

**项目名称** 改造普通灯变成WiFi控制灯

**班级与班级代码** 19级计算机科学与技术（实验班）

**实验室名称（或课室）** 702

**专 业** 计算机科学与技术

**任课教师 肖银皓**

**学 号：** 19251106139

**姓 名：** 李俊彬

**实验日期：** 2022 年 4 月 1 日

广东财经大学教务处 制

**姓名 李俊彬 项目报告成绩**

**评语：**

**指导教师（签名）**

**年 月 日**

**改造普通灯变成WiFi控制灯**--------------------------------------------------------------------------------------------------------------------------------

1. **项目目的**

能够改造普通灯变成WiFi控制灯

1. **项目环境**

1、 NodeMCU（ESP8266）开发板一块

2、 高低电平继电器模块一个

3、 普通USB灯一个

4、 220V-5V变压器（根据具体情况选型）

5、 手机一台

6、 开发环境Arduino1.8.19

7、 相应的驱动安装（CH341SerSetup）  
8、 在arduino中安装esp8266的库

1. **项目原理**

Esp8266相当于作为一个web服务器，当我连接wifi后通过外部设备输入相应的IP，esp8266进行解析，将存储在esp8266，falsh中的网页读取并显示出来，当我点击网页上的按钮后，esp8266进行解析，控制灯亮。网页与服务器之间使用http协议。

**ESP8266原理：**

ESP8266的特点便是wifi芯片和它极小的体积，谁不想有一块能连接网络查看天气的开发板做成的手表呢。其本质是一个带有WLAN收发器的单片机，由于其内核为较为冷门的tensilica框架，所以主流的开发工具几乎都不支持，目前可用的开发工具有：

Arduino IDE，适用于C++语言编程。

刷Node MCU固件，使用Lua脚本编程，有第三方类IDE工具。

刷AT固件，使用额外的控制器通过AT指令来控制。

使用官方SDK从底层控制。

目前我们不会使用裸ESP8266芯片，而将它和开发板一起学习，我所使用的是乐鑫的nodemcu开发板。

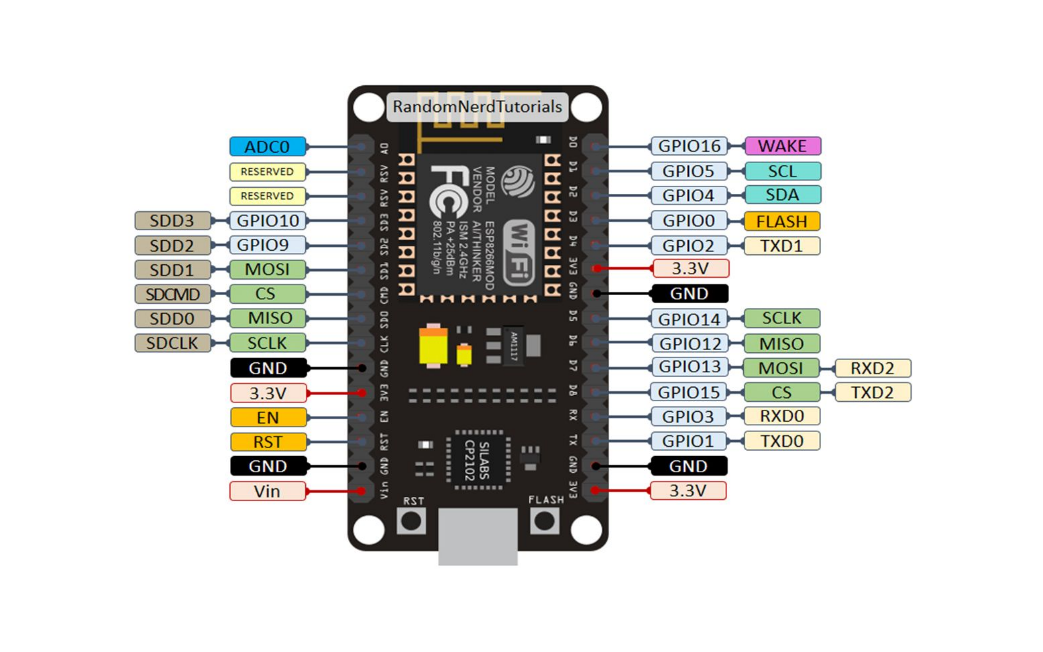
NodeMCU是一个开源互联网平台，基于esp8266开发的固件。主要有三种开发模式：

* + 1. AP模式：提供无线接入服务，允许其他无线设备接入，提供数据访问。比如路由器。
    2. STA模式：类似于无线终端，station本身并不接受无线的接入，可以连接到AP。就像无线网卡、手机一样，搜索到无线网后连接。
    3. AP+STA模式：二者合一

其硬件详细配置如下：

* + - 1. 核心模组为 ESP8266
      2. MCU 为 Xtensa L106
      3. RAM 50K
      4. Flash 512K
      5. D1~D10：10 GPIO, 每个都能配置为 PWM, I2C, 1-wire

下图为开发板引脚(GPIO)：



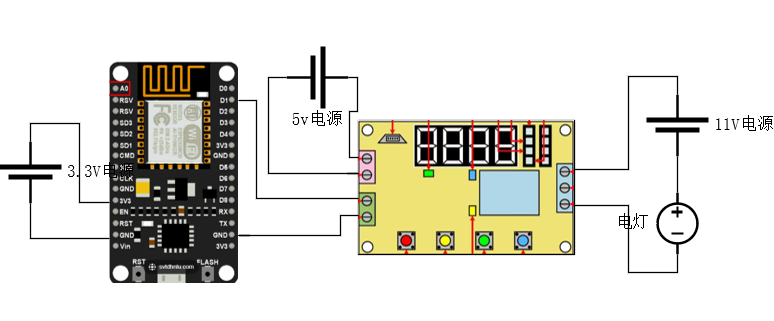
**HTTP工作原理:**

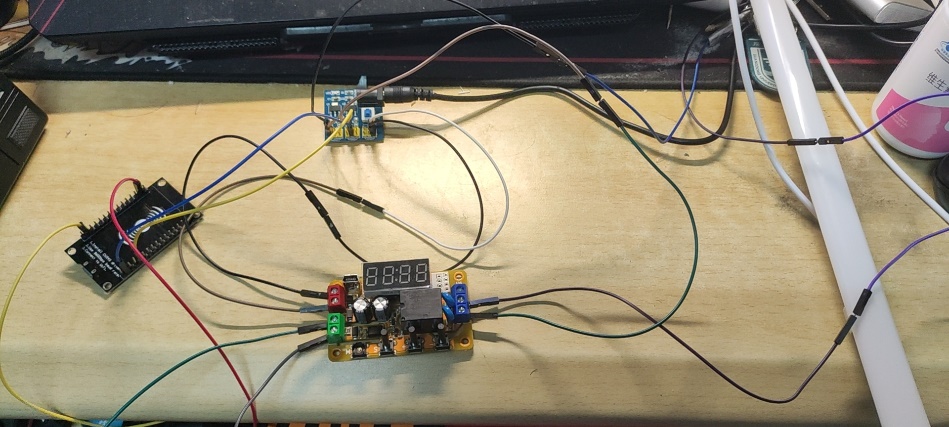
HTTP协议工作于客户端-服务端架构上。浏览器作为HTTP客户端通过URL向HTTP服务端即WEB服务器发送所有请求。

Web服务器有：Apache服务器，IIS服务器（Internet Information Services）等。

Web服务器根据接收到的请求后，向客户端发送响应信息。

HTTP默认端口号为80，但是你也可以改为8080或者其他端口。

1. **项目步骤与结果**
   * 1. **连接线路**

****

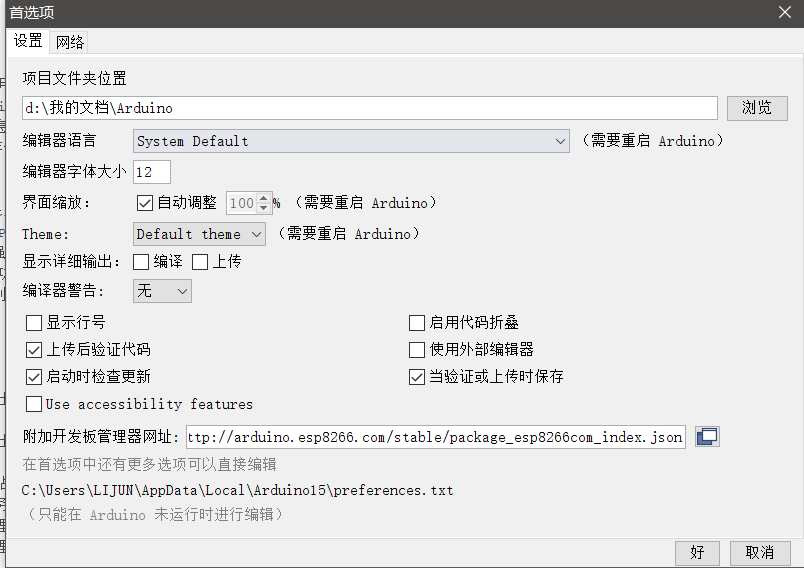
* + 1. **安装好NodeMCU的串口驱动模块，驱动安装程序在Github中**

****

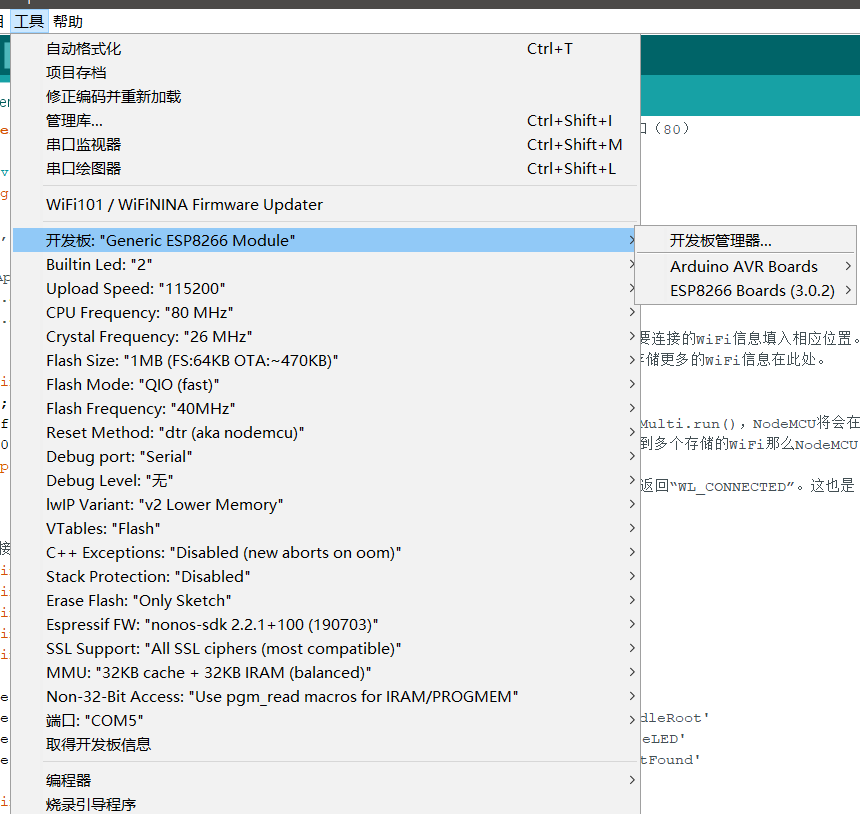
* + 1. **ESP8266的Arduino开发环境搭建**

先在Arduino官网下载Arduino，然后添加ESP8266板级支持

进入首选项（Preferences），找到附加开发板管理器地址（Additional Board Manager URLs），并在其后添加如下信息：<http://arduino.esp8266.com/stable/package_esp8266com_index.json>



之后点击工具 - 开发板 - 开发板管理器，进入开发板管理器界面：



在搜索栏上面输入ESP8266，选择最新版本点击“安装”，

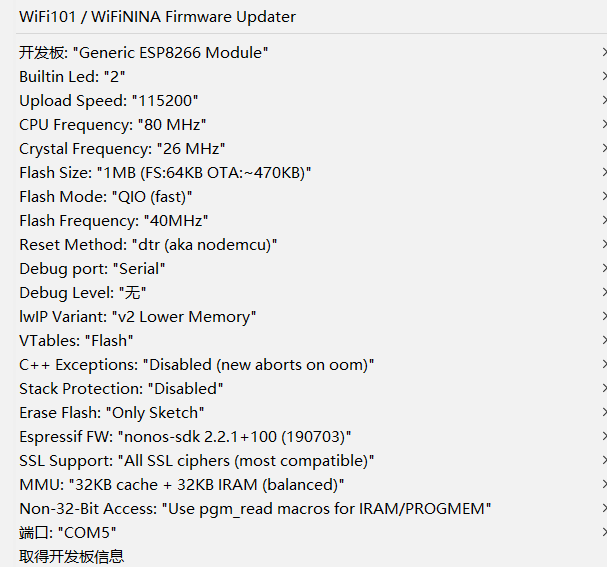


即可得到esp8266的各种类型的板级支持包。

这种方法是最长用的，但是这个源在国外，下载起来非常的慢，并且有时会出现无法下载，需要反复多次下载。

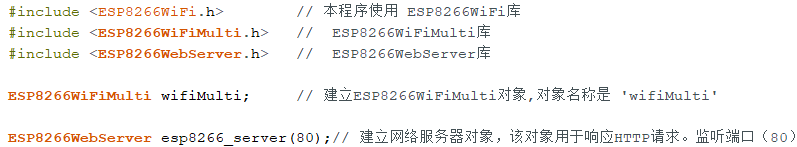
* + 1. **Wifi灯项目开发**
       1. 开发环境搭建好后，就可以进行esp8266的Arduino开发了，使用市面上的自带CH340 USB转串口的模块，通过USB线连接到COM口

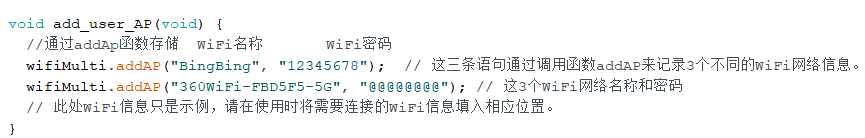
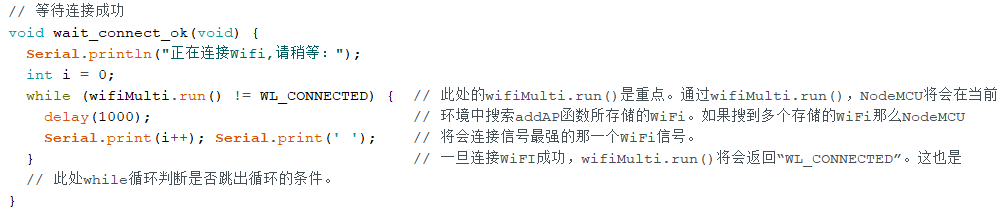
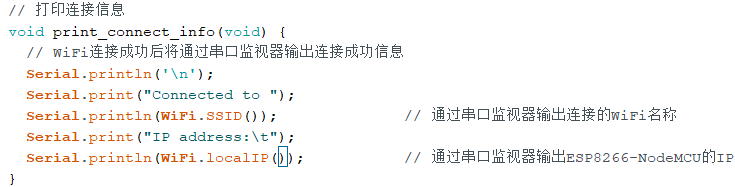


* + - 1. 然后打开IDE中的工具，即可以看到当前开发板连接的信息。配置如下：

接下来进行Arduino代码的编写，实现在Web端通过WiFi点亮ESP8266模组上的蓝色小灯的案例。

* + - 1. 引入ESP8266WiFi库、ESP8266WiFiMulti库、ESP8266WebServer库 ，并启动端口。



* + - 1. 然后将定义好的“用户名”和“密码”分别改为你的PC端正连接的WiFi热点和密码，即将esp8266与PC连接在同一网段下。
      2. 等待连接成功
      3. 打印连接信息
      4. 启动网络服务功能



* + - 1. 在setup()中配置服务端。将波特率设置为115200，并对模组上的蓝色小灯进行初始化，输出为低电压，让它先处于打开状态，4号脚位定为输出。我这里是4号引脚接LED。

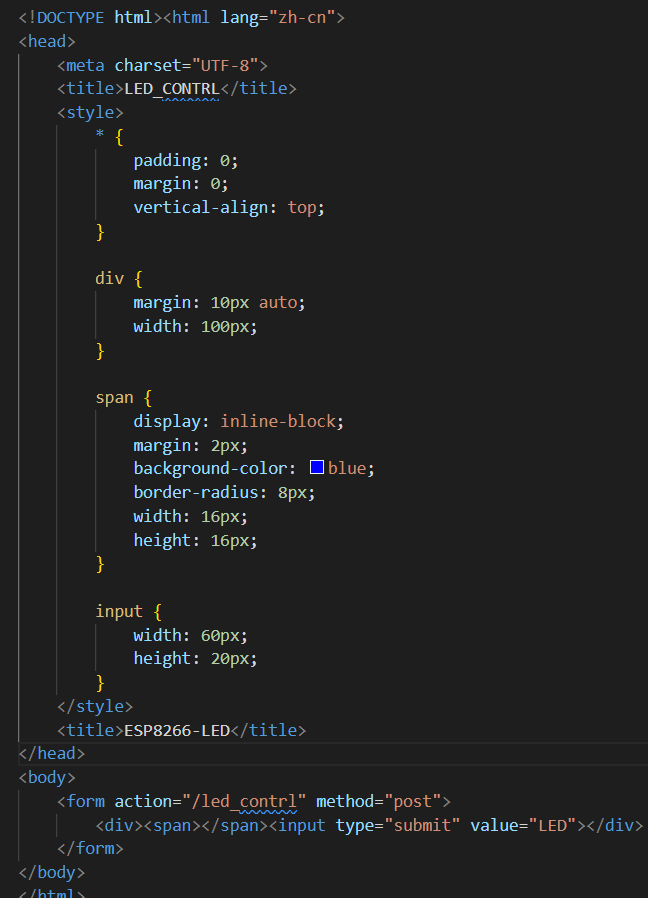


* + - 1. 定义HTML代码，生成按钮



代码解释

show\_html()就是将下面的html网页代码文本，逐行转换为字符串。

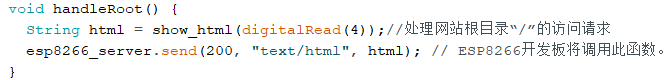


* + - 1. 设置服务器根目录即'/'的函数handleRoot()

该函数的作用是每当有客户端访问NodeMCU服务器根目录时，NodeMCU都会向访问设备发送 HTTP 状态 200 (Ok) 这是send函数的第一个参数。

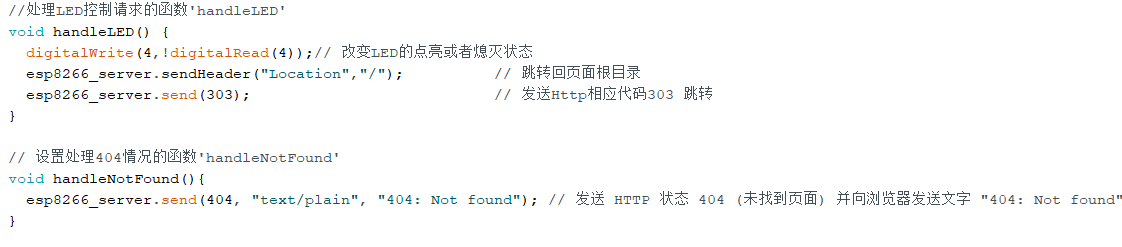
同时NodeMCU还会向浏览器发送HTML代码，以下示例中send函数中第三个参数，

也就是html=show\_html()就是NodeMCU发送的HTML代码。该代码可在网页中产生LED控制按钮。



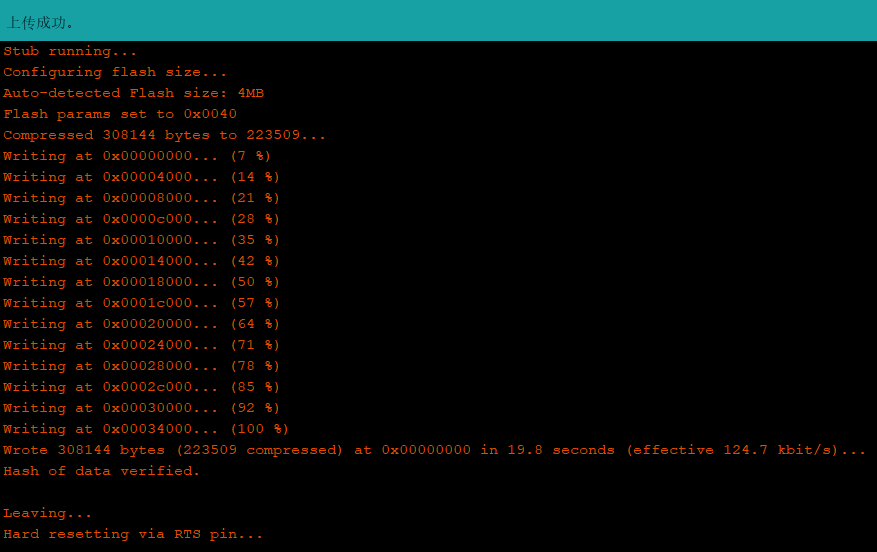
当用户按下按钮时，浏览器将会向NodeMCU的/LED页面发送HTTP请求，请求方式为POST。

NodeMCU接收到此请求后将会执行handleLED函数内容

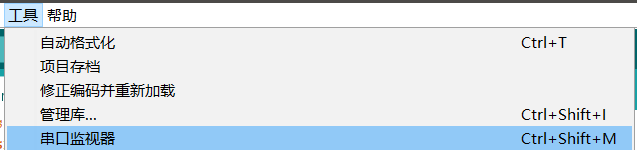


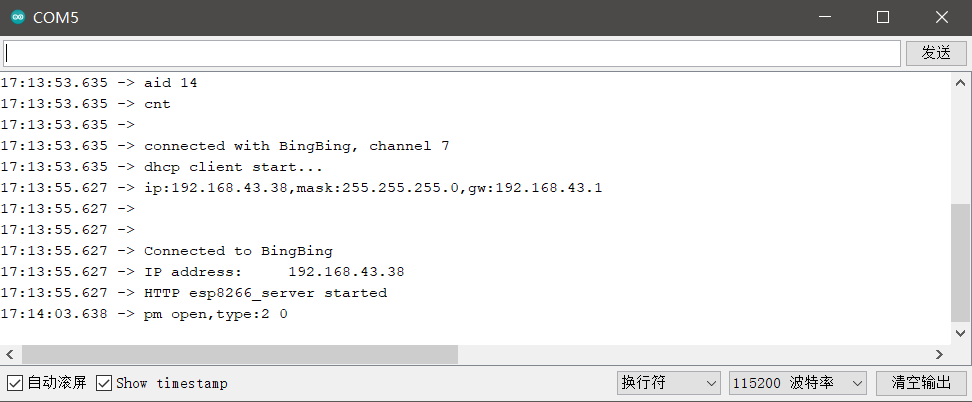
* + - 1. 进行程序烧录

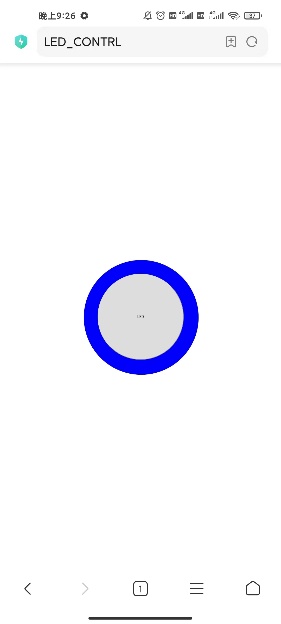
点击上传



* + 1. **效果展现**
       1. 打开串口监视器

****

* + - 1. 可以得到esp8266派发的ip地址
      2. 打开网页



* + 1. 控制电灯开关



视频链接：

1. **项目总结**

本次实验通过软件硬件相结合。完成了搭建服务器，响应网页请求，实现网页交互效果，根据请求操作设置高低电平修改继电器，形成回路点亮电灯，完成了对该项目的实验要求。实验中遇到的问题的分析与处理：

Wifi模块接入电路后，在开关灯时引发电灯频闪：

依旧有频闪现象。尝试接入其他电源为ESP8266供电，频闪现象得以改善。分析得电源负载过高导致电流不稳定。依旧有频闪现象。还通过进行线路的固定，频闪现象得以解决。