# ESP8266使用入门教程

**本文目标：了解esp8266以及其开发流程**

**芯片介绍：8266片上集成wifi+MCU，使用的是一个M0的内核，而且成本很低，因为片上有wifi和MCU，所以作为网络终端非常的方便，当然，因为是wifi，所以低功耗方面就别想了，低功耗+联网，NB-IOT更加合适。**

**固件：下面先介绍一下芯片固件的概念，说白了，esp8266也是一个单片机，上电还是得从0地址开始跑，平时我们使用单片机，一般都是使用keil等软件编程，然后下载，软件很多事情已经帮我们做好了，我们的重心放在main函数之后就行了。所谓的固件，我们可以把它看做一个很大的程序，只不过人家帮我们写好了，上电就开始运行，然后一直等待我们给单片机发送指令，我们发送指令后就执行相应的操作。**

## esp8266的固件有两种

- AT固件，芯片出厂的时候里边刷的就是AT固件，AT固件，用户主要通过串口使用AT指令跟8266交互，要控制8266。所以使用这种固件的时候还需要一个主机通过串口跟8266连接，这种使用方法，就单纯将8266当做一个网络传输芯片，串口转wifi，本文不讨论AT固件。
- Node-mcu固件，重点来了，因为这个固件才能完全发挥8266的魅力，先说一下这个固件的魅力，官方介绍是，这套固件，可以让8266像Arduino一样操作硬件IO，而且让你能完全使用API接口进行开发，更要命的是，固件里边可操作的模块还很多，像gpio操作、json处理、file文件创建管理、网络连接等等。举个例子说明一个这个固件：这个固件就像是安卓手机的刷机包，刷机之后我们就可以通过图形界面进行各种操作，在安卓手机上运行各种应用程序，esp8266刷入nodemcu固件之后，也能在上边运行我们编写的应用程序。

下面放几段操作8266的代码

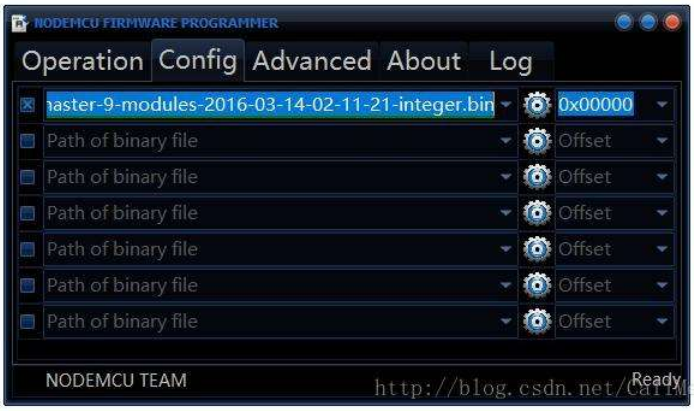
```
--操作GPIO
pin = 1
gpio.mode(pin,gpio.OUTPUT)
gpio.write(pin,gpio.HIGH)
gpio.mode(pin,gpio.INPUT)
print(gpio.read(pin))

--连接wifi
wifi.setmode(wifi.STATION)
wifi.sta.config("SSID","password")
print(wifi.sta.getip())
```

代码基本上不用注释，一看名称知道是做啥的

## 下面开始讲如何搭建开发环境

就像上文说的，芯片出厂的时候是AT固件，要刷如nodemcu固件才能使用这种开发方式，说以先要刷固件，当初我开始看的时候，网上一大堆各种各样的各种版本的固件，不是说只有两种固件吗？现在先不用管这个，按照步骤来，后边慢慢说。
先连接8266，建议大家开始研究的时候使用开发板，这样能省下很多时候时间，后期再上核心板

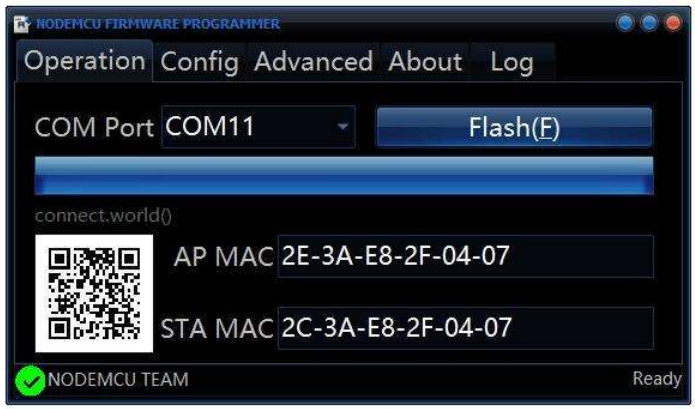
1.首先打开刷固件工具ESP8266Flasher.exe，选择要刷入的固件




2.点击Flash开始烧写

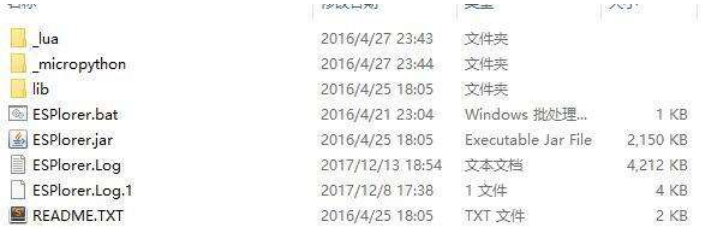



3.等待一会烧写成功，如果不成功多试几次就行了





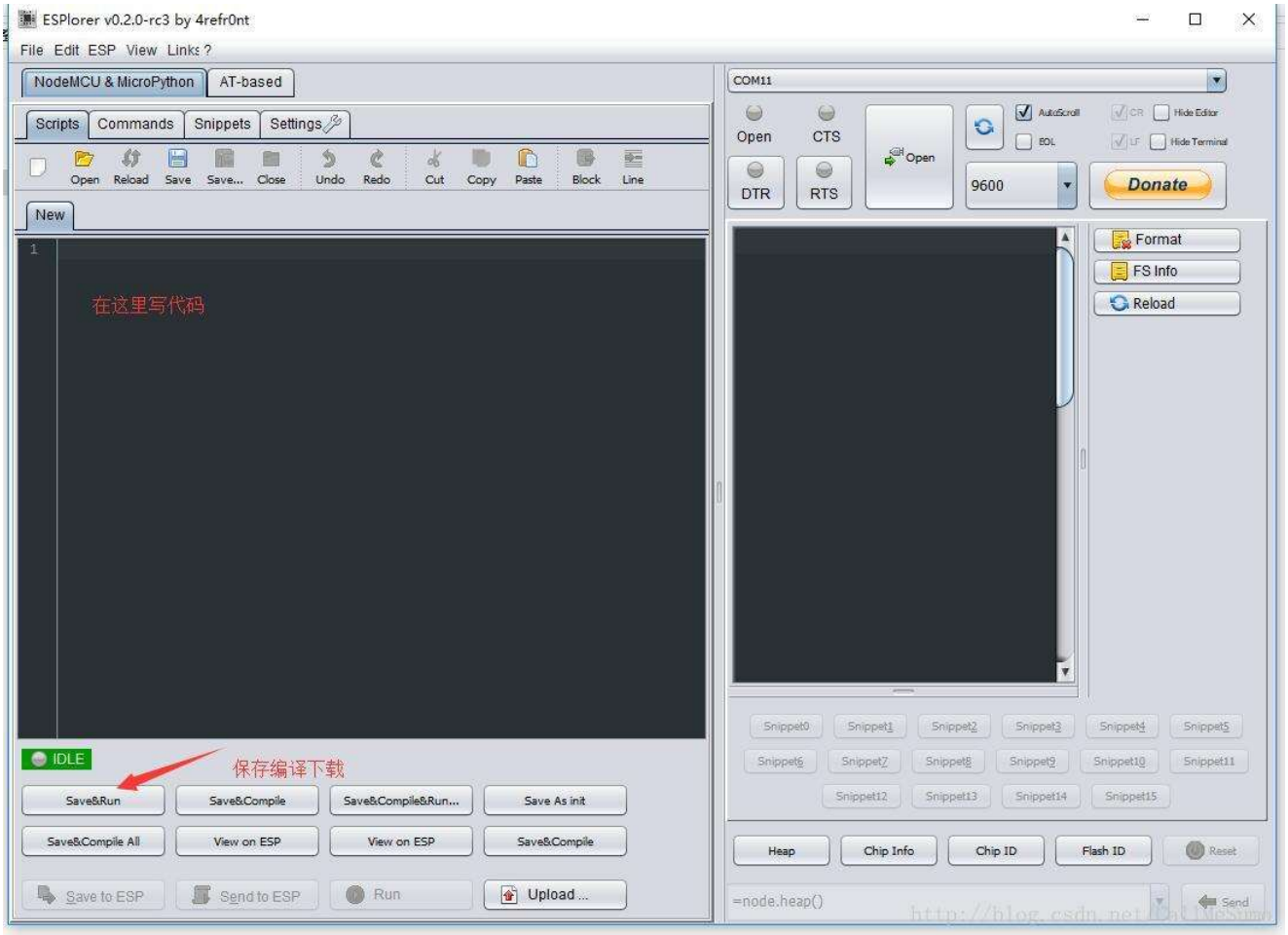

接下来就可以开始写程序了，程序使用Lua语言编写的，至于为啥是Lua语言，因为这个固件里边包含一个Lua语言解释器，就好比安卓上使用java语言开发应用程序。
开始写第一个程序，最简单的就是串口输出了
程序编辑以及烧写，使用另外一个软件ESPlorer
1.解压ESPlorer.zip文件，得到以下东西



| | | | |
|---|---|---|---|
| _lua | 2016/4/27 23:43 | 文件夹 | |
| _micropython | 2016/4/27 23:44 | 文件夹 | |
| lib | 2016/4/25 18:05 | 文件夹 | |
| ESPlorer.bat | 2016/4/21 23:04 | Windows 批处理... | 1 KB |
| ESPlorer.jar | 2016/4/25 18:05 | Executable Jar File | 2,150 KB |
| ESPlorer.Log | 2017/12/13 18:54 | 文本文档 | 4,212 KB |
| ESPlorer.Log.1 | 2017/12/8 17:38 | 1 文件 | 4 KB |
| README.TXT | 2016/4/25 18:05 | TXT 文件 | 2 KB |

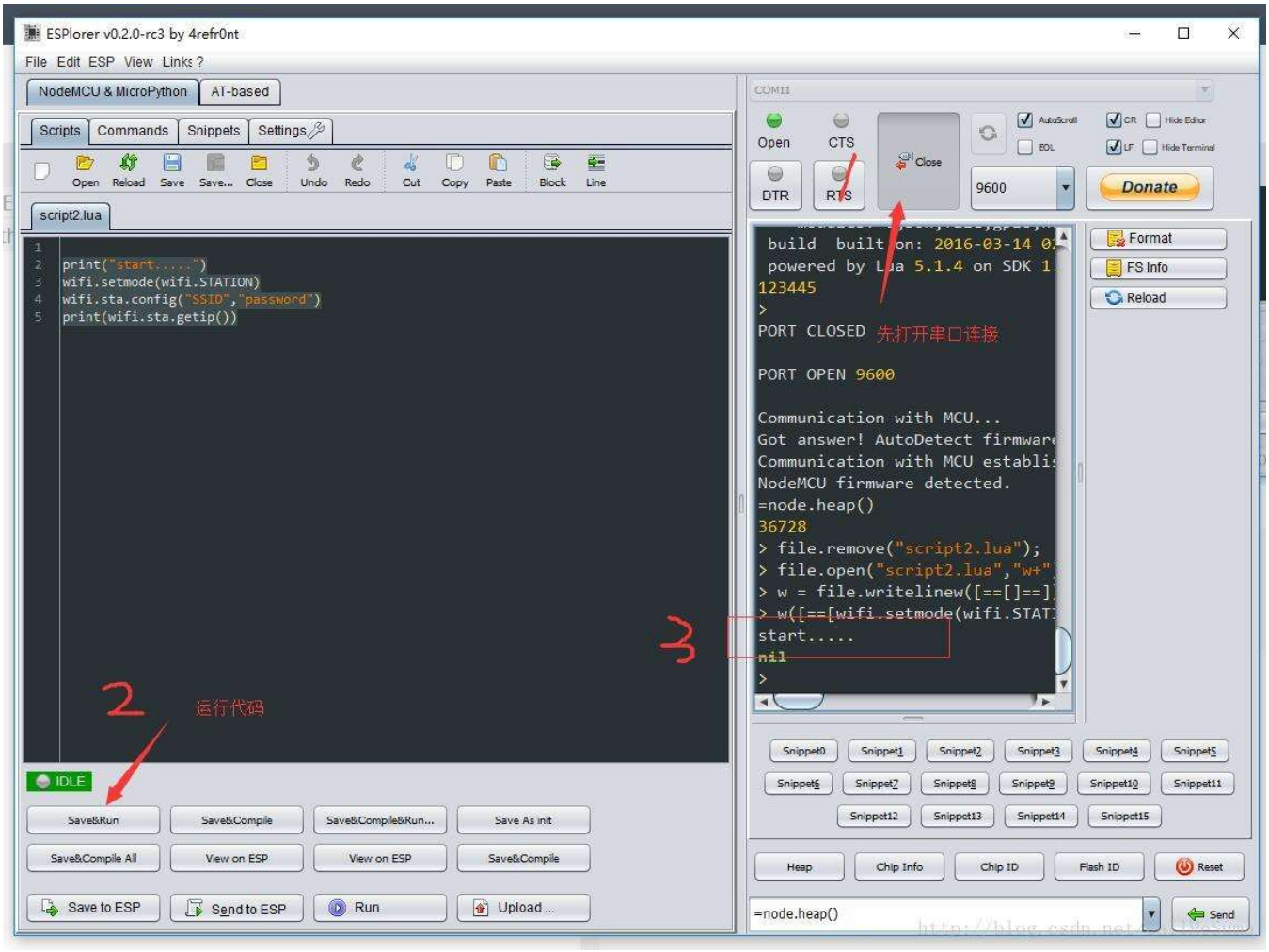


2.打开 ESPlorer.bat




3.开始写代码，我们让8266连接手机的wifi热点，当手机提示有新的终端接入的时候，就证明代码正确执行了

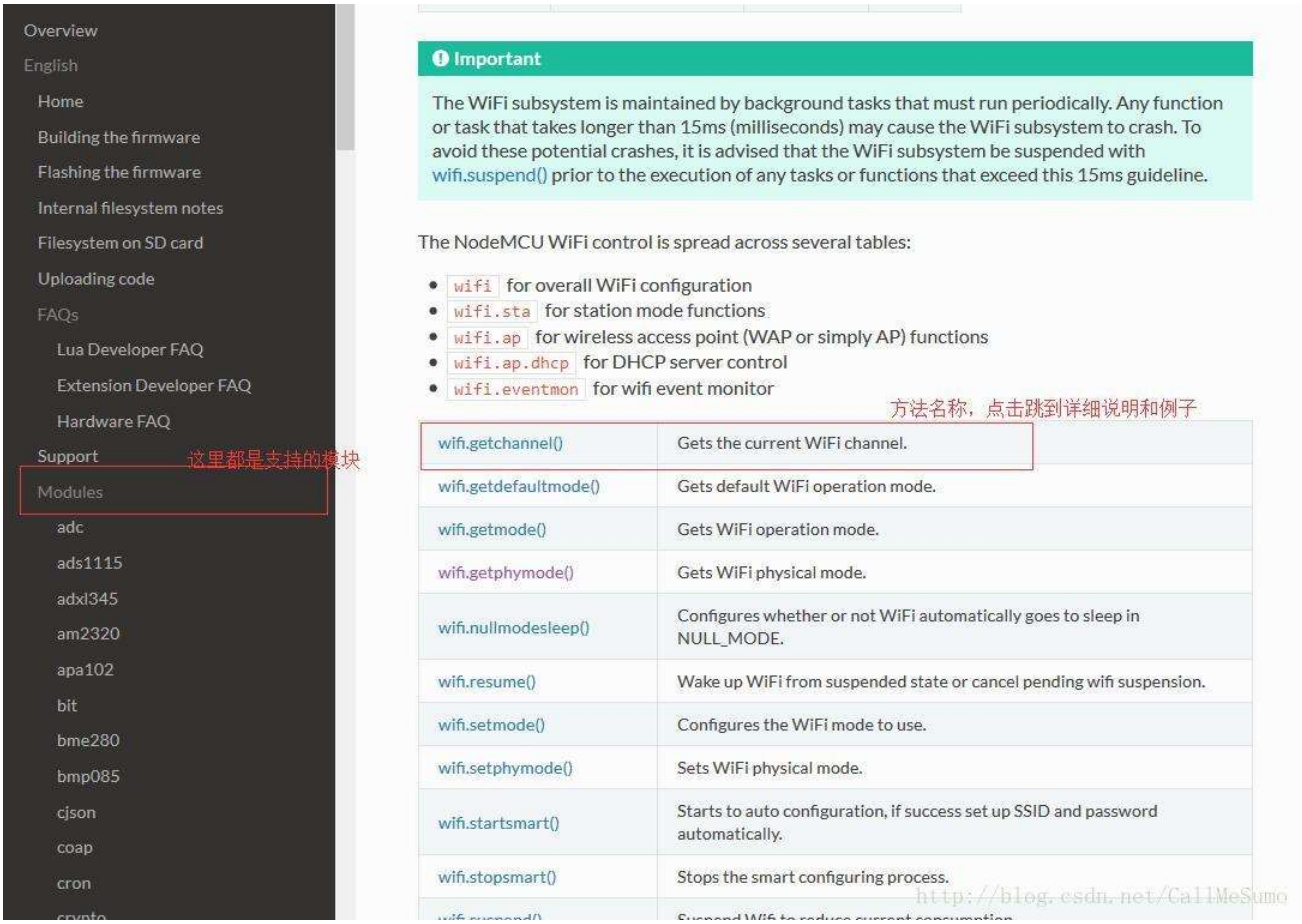
```
print("start.....")
wifi.setmode(wifi.STATION)
wifi.sta.config("SSID","password")
print(wifi.sta.getip())
```




看到串口这边有输出 “start.....”,证明代码已经执行，等一会手机的热点应该会提示有新的设备接入了。
这里说明一下，8266复位的时候，默认是执行init.lua这个程序，所以我们要让程序一上电就开始运行，在保存文件的时候，就要就将文件的名称的改为init.lua，这样才能实现上电就运行

### 程序怎么写?

现在知道程序怎么写之后，就可以开始看一看这个固件的API文档了，里边有所有模块的API用法以及例子
网址：https://nodemcu.readthedocs.io/en/master/en/modules/wifi/




## 固件的编译

可以看到nodemcu里边包含的模块很多，但是8266的资源是有限制的，如果固件里边全部包含了这些模块，就很占用空间，这样我们可以写代码的地方就少了，而且有些模块并不是我们需要的，所以我们要能选择自己需要的模块，然后编译成自己定制的固件，然后再烧到芯片里边。
这个nodemcu是开源的，下载源码，设置好交叉编译链，选择需要的模块，在linux下可以编译出自己的固件，但是这样太麻烦，需要linux环境。官方还提供了一种方法，就是在线编译，选择自己需要的模块，填写自己的电子邮箱，一会之后就会将编译好的固件发送到你填写的邮箱

网址：https://nodemcu-build.com/


按照步骤来就行了，这就不细说了

工具下载：https://pan.baidu.com/s/1nuDcgAX
密码：i94p