**ESP8288模块基本操作**

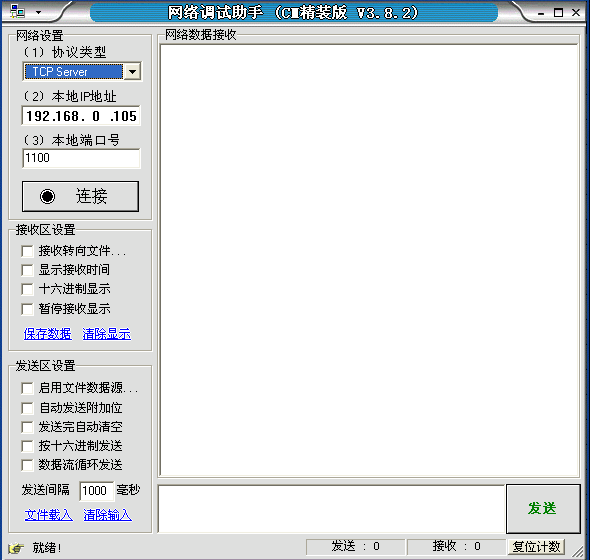
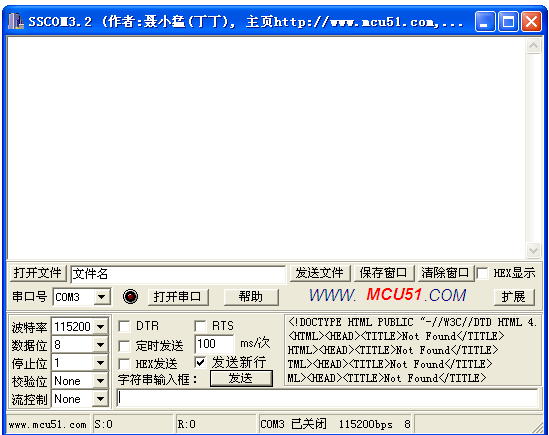
本人对于ESP8266S模块是菜鸟，这里只是把我的使用心得写给别的菜鸟，菜鸟写的东西，极可能会有错误或不妥的地方，所以本文有风险，阅读须小心。

ESP8266模块有多种型号，我手里有三只ESP8266-01型模块，我也只试了01型，所以本文都是关于01型的，我这三只01型模块分购于三家，后来确认有两只是山寨产品，但基本功可以用。拿到块子后，首先是读资料，三家的资料大同小异，估计也是复制过去复制过来，所以我读任何一家的资料都读得很累，找不到头绪，于是我判断是资料写得不好，这只是我个人的判断而已，进而产生了给像我一样的菜鸟们写一个容易入门的资料的想法，所以本文很浅显，非常适合菜鸟。如果你玩过单片机，读本文就没有难度了，要不，你至少从网上购了模块，还下载了一些资料，电脑的那点常规动作还是会的，当然你还得会玩手机，会在手机上安装个程序什么的，那就可以开始了。

先准备好必要的工具，玩这款模块，可以使用的软、硬件工具很多，但菜鸟只需以下几件就行了，入门后再玩别的工具或自己开发工具。软件工具如下：

1. 串口助手sscom32.exe（用于电脑）
2. 网络助手NetAssist.exe（用于电脑）
3. 有人网络助手USR-TCP-Test.apk（用于手机）

下面是这些软件打开前、后的样子。



有人网络助手USR-TCP-Test.apk需要在手机上安装，安装前、安装后、打开后分别是这个样子。



有人网络助手打开后，点tcp server和tcp client，再点配置分别是这个样子，



其他自己试。

以上这些工具软件有不同版本，界面差别不大，使用方法也差不多。

硬件如下：

1，ESP8266（这里用的是01型）

2，电脑，可能不止1台。

3，USB转TTL模块，比如CH340型、PL2303型都可以

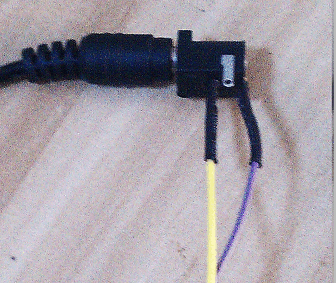
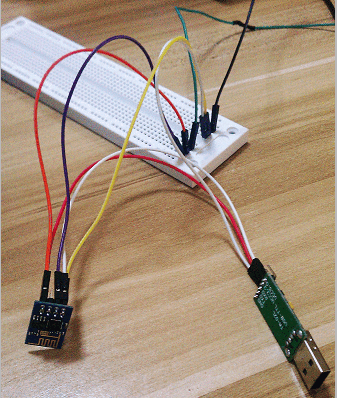
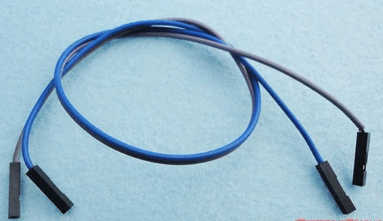
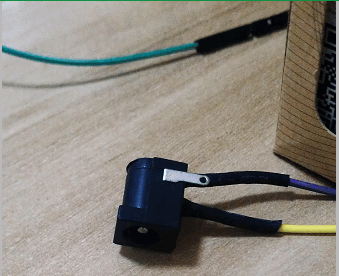
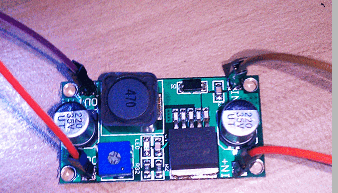
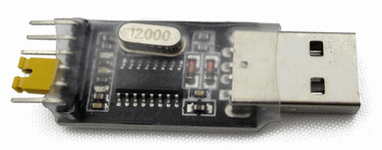
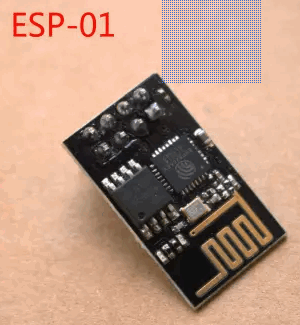
4，固定型或可调型电源模块，输出3.3V，这里用的是可调型。

5，9V或者12V电源，就是家里路由器用的那种电源，用于给可调电源模块供电，需配一个 5.5\*2.1的DC座子，就是直流座子的意思。

6，杜邦线（两头都可以插）若干，拨插方便，同时还可以配点别的硬件，比如面包板，也是 为了方便。

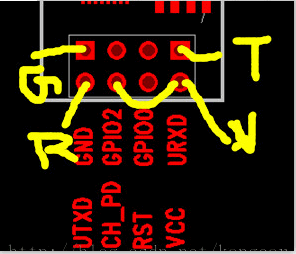
第4和5条可以省掉，改用两节干电池，别用充电型的，电压不够。我在试验中，先使用电池，出了些怪现象，可能是接触不好，也可能是电池电量下降，后来改为电源模块，就没有那些怪现象了，所以做试验时，我个人推荐使用电源模块。如果手里暂时没有，就先用电池吧。

下面是这些硬件的样子，电脑图片就不贴了。



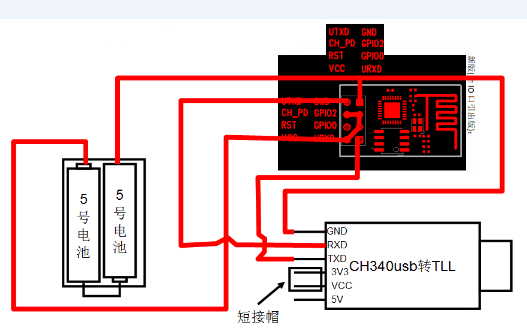
最好是用电烙铁把电池盒的正负极都焊上杜邦线头，这样方便使用，如果用电源模块，焊接工作就多一点点，那个可调的电源模块，IN是9V或12V进端，印有正负符号来区分，OUT是输出端，自己调那个蓝色的可变电阻，同时用万用表看看是不是3.3V。DC座的后面那条蓝线是9V或12V的正极。

接线图，这里有两种基本接线方式，



图中G、R、T分别与USB转TTL模块的GND、RXD、TXD连接，同时GND与电池负或3.3V负连接，V和CH\_PD连接，同时V还与电池正或3.3V的正极连接，接线完成。

另一种接线图，我没有试过，如下。



下面就可以上电调试了，上电前请确认接线是否正确，供ESP8266板子的电源是否为3.3V，这两条没问题，就出不了大事。把9V或12V电源插在家里的220V插座上，出来就是9V或12V了，经过那个可调电源模块，出来就3.3V了，就直接对板子供电，此时ESP8266上的红色指标灯亮，同时蓝色灯闪几下后灭，再把USB转TTL模块插进电脑的USB口，就是插U盘的口，随便哪个都行，注意，电脑需安装相关驱动程序，安装使用很简单，购时店主会告诉你这些或有相关资料，插入后，可以查看一下是几号口，右击我的电脑/管理/设备管理器/双击端口，可以看到如这样的情况，我这里是COM4口，XP系统和WIN7系统的操作几乎一样。



接下来，打开串口助手sscom32.exe，看看串口号是不是刚才的那个口，如果不是，就下拉后选COM4，这个数字每次可能会变，所以需要先看一下，然后勾上“发送新行”，波特率选115200，点“打开串口”，打开后显示“关闭串口”，旁边有个红灯就是已经打开了，别的不用动，在输入框里试着输，比如AT+RST，点“发送”，如果发现有什么不对，可以关掉串口助手，重新打开，还不行的话，拨一下电池或3.3V的电源线，拨一根，比如正极就行，再插上，应该行了，在输入的时候，要在英文状态下以大写输入，AT+RST是一条AT指令，所有的AT指令前后和中间都不要有空格，试验完后，要先关闭串口助手再拨出USB转TTL模块，养成这个习惯，如果用别的串口助手软件，在输入完后要回车一次，这个串口助手因为勾选了“发送新行”就不用回车了，如果先显示一堆乱码，再显示ready，恭喜你，板子正常，乱码对厂家有用，菜鸟阶段不用关心，如果最后出现invalid，也恭喜你，板子接线没错，可能是出生不好，是山寨厂出的，有基本功能，没有云服务功能，具说信号也要差一些。

先说明一下我们接下来要做哪些试验以及与这试验有关的一些术语，先说术语，AP、STA、AP+STA、透传、固件、AT指令、TCP、UDP、IP。AP在这里可以理解为一个无形的wifi插座，就像家里的电源插座一样，浑身有多个插孔供外界使用，只不过wifi插座是无形的，模块、手机、电脑的wifi信号都可以“插”在AP上。STA是只有一个插头，没有插孔的终端，所以STA只能插别人，别人不能插它。AP是一个信息转发站，自己并不查看这些信息，就像传达室，只负责转发信件，从不打开信封看里面的内容。STA是可以从传达室收信，并打开看里面的内容，也可以写信交传达室转发出去。如果有一封信是寄给传达室的工作人员的，这个人收信后就可以打开看内容，这时的模式就是AP+STA。当然，这个人也可以写一封信通过传达室转发出去。透传，透明传输的意思，是指使用者在一个网络的一端输入一个数据，另一端输出就原封不动得到这个数据，中间过程不需要使用者关心，其实一点都不透明，也许改为“黑箱传输”更适合，可能是为了特别强调数据传输时不需要加密，是什么数据就传什么数据，所以是透明的，不过这里的透明，只是对使用者而言，事实上无线模块在传输时都是不透明的，因为模块本身在发送时都是要加密的，在这个基础上，使用者对数据加密或不加密，一般来说民用时都是安全的了，真要像军队、公安、银行、税务等要害部门，估计也不会选用ESP8266，他们不缺钱，都是选我们一般不怎么关注的产品。固件其实也是软件，它存储在某芯片上，使用者改不了，对使用者是固定的，但ESP8266固件可以刷新，一刷就全部重来，而不能改某个语句，固件首先是硬件的驱动程序，即固件是指挥硬件干事的，同时再增加一些应用程序的功能，这样使用者只需要下达指令，而不用关心块子处理这个指令的具体过程，比如你按遥控器的某个键，电视机就会转到CCTV1，这个过程很复杂，但你不用关心，你按键的这个动作就相当于ESP8266的一个AT指令，比如AT+RST、AT+GMR都是AT指令，AT指令输入后都有返回，见到如OK、CONNECT、WIFI CONNECTED等那就是对的，如果见到如ERROR、WIFI DISCONNECT、FAIL等那就是不对，出现invalid，要么是山寨产品，要么是没有激活，基本功能可用，大多数情况下，瞬间反回，连网的时候可能会多等一阵，几秒到十几秒甚至更久，有时可能要多连几次，大多数时候，连接一两次不行，估计设置就有问题，如果连上后一段时间内没有进行收发，可能会自动断开连接，这是正常现象。UDP、TCP、IP都是网络用语，挺复杂，简单的说，UDP和TCP其实是一家人，UDP是简化版的TCP，都是用于网络传输的一堆协议，这里的协议，是指发送方和接收方共同遵守的一些约定，就如同接头暗号，它们俩最大区别在于TCP传数据前要查接收方在不、忙不，UDP不管这些，有数据就发，有没有人接收它就不管了。一般来说UDP用于数据量少，不太重要的地方，比如亮灭一盏灯什么的，就算没收到，也不会有严重后果，IP可以理解为网络上的地址，用来区分网络上的不同用户，和家庭地址的作用类似，在同一个局域内IP必须唯一，格式如 192.168.4.1。

我们菜鸟只做以下这几个试验：（别的试验有空也要做哦）

1. 模块与手机直接通信。
2. 模块与电脑直接通信（电脑是自带无线功能的，如手提电脑）。
3. 模块与模块直接通信。
4. 模块与手机通过路由器通信。
5. 模块与电脑通过路由器通信（电脑是自带无线功能的，如手提电脑）。
6. 模块与模块通过路由器通信。
7. 网外远程通信。

下面开始做试验，做单个模块试验时，如果连不上，除按前面说的检查外，看旁边还有没有别的模块正处于连接状态，断开后再试。

1，模块与手机直接通信（选手机作服务器）

//按顺序在串口助手的输入框中输入下列AT指令，点“发送”，有返回后再输下一条。

AT+CWMODE=3 //设置为AP+STA

AT+RST //重启

AT+CWSAP="ESP8266","0123456789",1,0 //模块发起连接，名称 密码 通道（1-14），不加密

//这时就可以打开手机连接到ESP8266上，再打开手机上的有人网络助手，TCP server→//配置→激活→能看到此时手机的IP和端口号，要记下，下面要用，我前面已经设置多//次，现在端口号是1100，有时可能由于前面对手机进行了不同设置，看到的手机IP可//能有点问题，可以退出有人网络助手再打开，一般没有问题。继续输入以下指令。

AT+CIPMODE=1 //透传

AT+CIPMUX=0 //单路模式

AT+CIPSTART="TCP","192.168.4.2",1100 //填刚才记下的手机IP和端口号

//可能要等一阵，如果不行，再试一次，有可能要从头来，原因很多，这时手机已经能向//模块发信息了，但模块不能发。

AT+CIPSEND //模块进入透传，要退出透传，取消“发送新行”的勾，发“+++”后重新勾上

现在模块和手机就能通信了。在试验过程中有一些有用的AT指令会常用到，应熟悉一下。

AT+CIFSR //获取本模块IP

AT+CWLIF //查看接入的客户端IP

AT+CWLAP //列出当前可用AP

AT+CIPMODE? //查模块的当前状态

AT+CIPMUX? //查询连接模式

AT+CIPMUX？ //查传输模式

AT+CIPSTATUS //查模块的连接状态和连接参数

AT+RESTORE //恢复出厂设置，设置乱了就恢复重来

2，模块与电脑直接通信（自带无线功能的，如手提电脑，选电脑作服务器）。

AT+CWMODE=3 //设置为AP+STA

AT+RST //重启

AT+CWSAP="ESP8266","0123456789",1,0 //模块发起连接，名称 密码 通道（1-14），不加密

//记得关掉手机和ESP8266的连接，不然这里可能连不上。这时电脑可以连接到ESP8266 //上，再打开电脑网络助手，TCP server→端口号1100，记住本地IP，下面要用。

//继续输入以下指令。

AT+CIPMODE=1 //透传

AT+CIPMUX=0 //单路模式

AT+CIPSTART="TCP","192.168.0.105",1100 //填电脑IP和端口号

//可能要等一阵，如果不行，再试一次，有可能要从头来，原因很多，这时电脑已经能向//模块发信息了，但模块不能发。

AT+CIPSEND //模块进入透传

3，模块与模块直接通信。

这个试验需要两台电脑，因为两只模块都需要通过串口在电脑上操作，走过菜鸟阶段，就可以用单片机连接模块而不需要电脑。现在我们用电脑操作，按前面的操作，将两只模块接好线，并分别插在两台电脑上。

///// 分A机和B机，A机作服务器，先设置A机/////////////////////////////////////////

AT+CWMODE=3 // A机设置为AP

AT+RST //重启

AT+CWSAP="ESP8266","0123456789",1,0 //模块发起建立连接，名称 密码 通道（1-14），不加密

AT+CIPMODE=0 //非透传

AT+CIPMUX=1 //多路模式

AT+CIPSERVER=1,1100 // A机作服务器，端口1100（可自己设定）

AT+CIFSR //查看A机的IP并记住

//去设置B机

AT+CIPSEND=0,4 // ID号0，长度4，不能连续发，因为如果再挂一只模块，

// ID号就变1，不同的ID号就发给不同的模块，每次都要选ID号。

/////设置B机////////////////////////////////////////////////////////////////////

AT+CWMODE=1 //B机设置为STA,

AT+RST //重启

AT+CWJAP="ESP8266","0123456789" //搜索并加入A机的连接

AT+CIPMODE=1 //透传

AT+CIPMUX=0 //单路模式

AT+CIPSTART="TCP","192.168.4.1",1100 //填A机的IP和端口

AT+CIPSEND //进入透传

4，模块与手机通过路由器通信（手机作服务器）

AT+CWMODE=1 //设置为STA

AT+RST //重启

AT+CWJAP="miwalk\_office","cqmyg14dss.miwalk" //搜索并连接路由器，x为路由器名，\*为密码

//再打开手机有人网络助手，TCP server→端口号1100，记住本地IP，下面要用。

AT+CIPMODE=1 //透传

AT+CIPMUX=0 //单路模式

AT+CIPSTART="TCP","192.168.0.100",1234 //填刚才记下的IP和端口号

AT+CIPSEND //进入透传

取消“发送新行”的勾，发送“+++”，退出透传模式并重新勾上。

5，模块与电脑通过路由器通信（电脑作服务器，这里的电脑是自带无线功能的）

本试验操作与试验4步骤相同，这个试验需打开电脑上的网络助手而非前面试验中的有人网络助手。

6，模块与模块通过路由器通信。

同试验3一样需要两台电脑，电脑是否有无线功能无所谓，我们只是用电脑来操作和查看，以后可以不用电脑。

///// 分A机和B机，A机作服务器，先设置A机/////////////////////////////////////////

AT+CWMODE=1 //设置为AP

AT+RST //重启

AT+CWJAP="xxxxxx","\*\*\*\*\*\*" //搜索并连接路由器，x为路由器名，\*为密码

AT+CIPMODE=0 //非透传

AT+CIPMUX=1 //多路模式

AT+CIPSERVER=1,1100

AT+CIFSR //查看A机的IP并记住

//去设置B机

AT+CIPSEND=0,4 // ID号0，长度4，不能连续发，因为如果再挂一只模块，

// ID号就变1，不同的ID号就发给不同的模块，每次都要选ID号。

////设置B机////////////////////////////////////

AT+CWMODE=1 //设置为STA

AT+RST //重启

AT+CWJAP="xxxxxx","\*\*\*\*\*\*" //搜索并连接路由器，x为路由器名，\*为密码

AT+CIPMODE=1 //透传

AT+CIPMUX=0 //单路模式

AT+CIPSTART="TCP","192.168.1.101",1100 //填刚才记下的IP和端口号

AT+CIPSEND //进入透传

前面的试验熟练后，以下综合试验可以自己试着玩了。

手机作服务器，通过路由器搭载多只模块、手机、电脑。

电脑作服务器，通过路由器搭载多只模块、手机、电脑。

模块作服务器，通过路由器搭载多只模块、手机、电脑。

手机作服务器的试验和试验4一样，电脑作服务器的操作也同理。

按试验4可以设置多只模块、手机、电脑，这样作为服务器的手机上就有多只模块的IP，手机要和哪只模块通信，就点这只模块的IP。模块都设置成STA，不管是手机还是电脑作服务器时，设置其他要搭载的手机和电脑，设置应选TCP client。

模块作服务器，

就是试验6，按B机模式可以设置多个模块、手机、电脑，服务器发送时选不同的ID号就能向不同模块、手机、电脑发送。

以上这些试验，实现的方式不止一种，这些试验所采用的方式不一定是最佳的，同时每一个试验，其功能并不完善，这里强调的是“入门”，相关优化和完善工作请自行处理。模块与模块直接通信，在这里虽然做了试验，但这个试验本身意义并不大，因为如果只是模块与模块之间进行无线通信的话，有更便宜的块子，ESP8266的价值在于“通过路由器”，在实际运用中，通过路由器搭载多只模块、手机、电脑组成的网络比较常见，这个局域网一旦建成，一个基本的物联网的最后十米的工程就完成了，因为通过路由器可以搭多个终端通信，而路由器和互联网是连在一起的，这才是关键，外网远程控制时，不论是通过云服务器还是别的手段，最终信号都要通过内网的路由器实现远程控制，所以玩ESP8266模块，主要就是玩它和路由器的配合，我敢肯定，开发ESP8266的最原始想法就是，以小的体量和低的价格解决当今正越来越流行的物联网最底层的无线连接问题，其最重要的任务，也是最基本的任务就是把小数据量的无线wifi信号从用电器传至路由器或电脑或手机，总体来说，ESP8266性价比真是挺高的。

7，网外远程通信。

该试验用模块和手机进行通信，模块挂在自家的路由器上，手机使用外网用流量，或上另一个地方的路由器，总之模块和手机不在同一个局域网。

AT+CWMODE=1 // 设置为STA

AT+RST //重启

AT+CIPMODE=1 //透传

AT+CIPMUX=0 //单路模式

AT+CWJAP="miwalk\_office","cqmyg14dss.miwalk" //搜索并连接自家路由器，x为路由器名，\*为密码

AT+CIPSTART="TCP","117.136.31.17",1234 //上外网的服务器

AT+CIPSEND //进入透传

login:abc,123 //模块方名称abc，密码123，自己改

手机：选TCP client，IP：118.193.213.167，端口1000

login:def,456 //在手机上输入手机方名称def，密码456，自己改

peer:def,456 //在模块的串口助手上输入刚才在手机上输入的名称和密码

在吗 //现在可以在串口助手上直接输入中文了，并会传到手机上

在的 //在手机上输入中文，在串助手上就能看见

这就相当于在QQ聊天，也就相当于实现了外网的远程控制，然后自己可以写手机端程序使界面更好看，遥控功能更多，更方便，写模块方单片机程序使模块收到你特定的指令去开一盏灯、关电扇、启动家里的防益装置等，进一步还可以调灯的亮度，音量大小，这些就是自己闹着玩的了，与ESP8266没多大关系了，因为ESP8266只负责传信号，不负责调信号。

有关ESP8266模块的东西远不止这些，本文任务到此为止，好了，往深的玩吧，希望菜鸟早日成大鸟。