ESP8266 系列入门教程

版本 1

Ai-Thinker Inc

目录

入门教程

一、概述

ESP8266 系列无线模块是安信可科技自主研发设计的一系列高性价比 WiFi SOC 模组。该系列模块支持标准的 IEEE802.11 b/g/n 协议，内置完整的 TCP/IP 协议栈。用户可以使用该系列模块为现有的设备添加联网功能，也可以构建独立的网络控制器。

安信可科技为客户提供完整的硬件、软件参考方案，以便缩短您的产品研发周期，为您节省成本投入。

1.1 产品特性

- 体积超小的 802.11b/g/n Wi-Fi SOC 模块
- 采用低功率 32 位 CPU，可兼作应用处理器
- 主频最高可达 160MHz
- 内置 10 bit 高精度 ADC
- 支持 UART/GPIO/IIC/PWM/ADC/HSPI 等接口
- 集成 Wi-Fi MAC/ BB/RF/PA/LNA
- 支持多种休眠模式，深度睡眠电流低至 20uA
- 内嵌 Lwip 协议栈
- 支持 STA/AP/STA+AP 工作模式
- 支持 Smart Config/AirKiss 一键配网
- 串口速率最高可达 4Mbps
- 通用 AT 指令可快速上手
- 支持 SDK 二次开发
- 支持串口本地升级和远程固件升级（FOTA）

1.2 选型表

型号	ESP-01S	ESP-01M New	ESP-01F New	ESP-07S	ESP-12F	ESP-12S New
封装	DIP-8	SMD-18	SMD-18	SMD-16	SMD-22	SMD-16
尺寸(mm)	24.7*14.4*11.0	18*18*2.8	11*10*2.8	17.0*16.0*3.0	24.0*16.0*3.0	24.0*16.0*3.0
板层	2	4	4	4	4	4
Flash	8Mbit	8Mbit	8Mbit	32Mbit	32Mbit	32Mbit
认证	-	FCC/CE	-	FCC/CE	FCC/CE/IC	FCC/CE/SRRC
天线	PCB天线	PCB天线	无内置天线	IPEX	PCB天线	PCB天线
指示灯	GPIO2	-	-	-	GPIO2	GPIO2
可用IO	2	11	9	9	9	9

表 1：ESP 模块选型表

注意：选型表中的模组为安信可热卖款。选型表中的信息会变更，例如模组的型号会不断的更新，模组的相关认证，我司也会持续的做模组的相关认证，请以官网信息为准。

电话：18923720150 邮箱：2355526548@qq.com

二、接线

安信可系列模组型号较多，在此提供基础的典型接线图，用户可基于如下典型接线图接线调试。更多电路设计请参考用户手册：http://wiki.ai-thinker.com/esp8266/docs

注意：尽量不要使用 USB 转 TTL 的 3.3V 或 5V 供电。建议使用 2 节干电池或经过 LDO 转换后的 3.3V 供电。

2.1 启动模式

模式	CH_PD (EN)	RST	GPIO15	GPIO0	GPIO2	TXD0
下载模式	高	高	低	低	高	高
运行模式	高	高	低	高	高	高
测试模式	高	高	-	-	-	低

表 2：启动模式设置

注意：如果要进入下载模式，参考章节 2.2 的典型接线图，在该接线的基础上将 GPIO0 接地即可。参考章节三进行烧录。

2.2 典型接线图

各型号 ESP 典型接线图如下所示，图中电阻均为 10K.

如需进入下载模式需要在对应模块图示基础上将 GPIO0 接地。

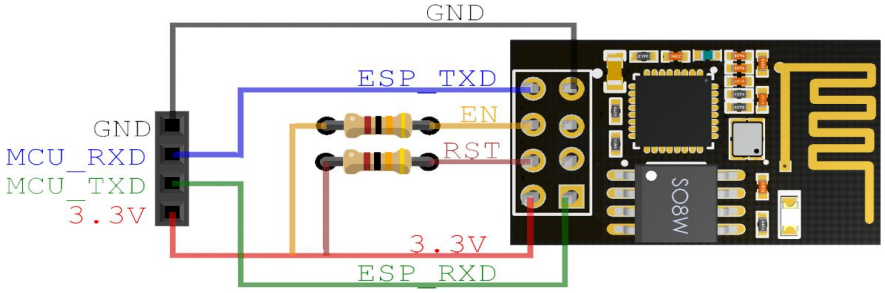


图 1：ESP-01 接线示意图

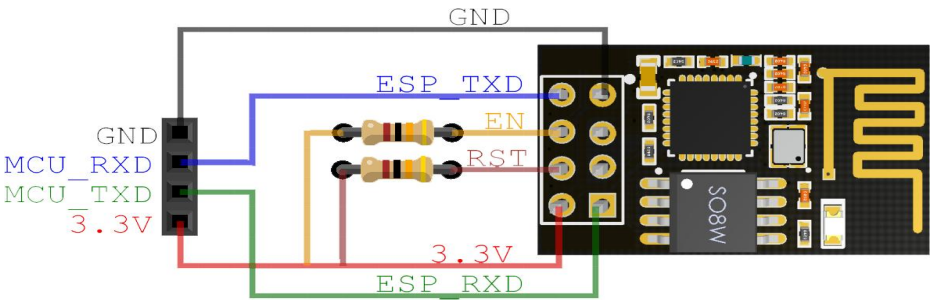


图 2：ESP-01S 接线示意图

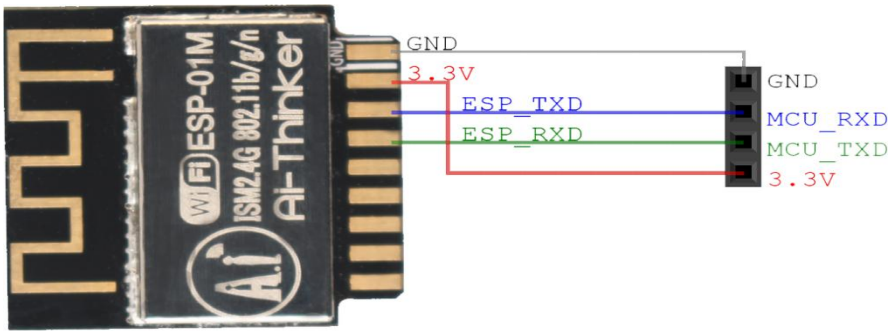


图 3：ESP-01M 接线示意图

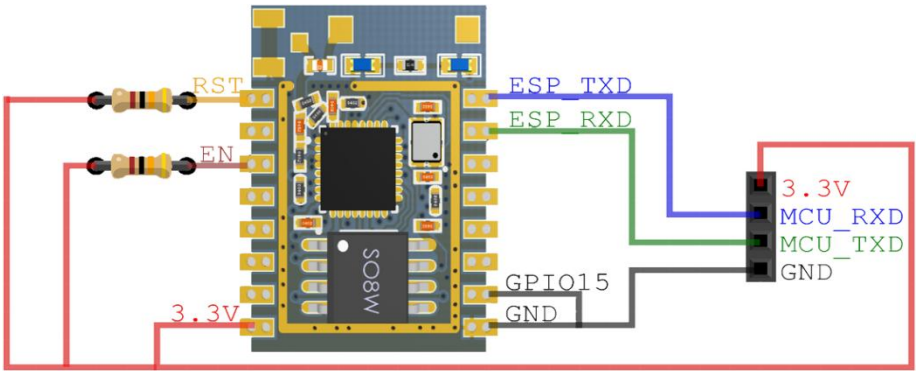


图 4：ESP-07 接线示意图

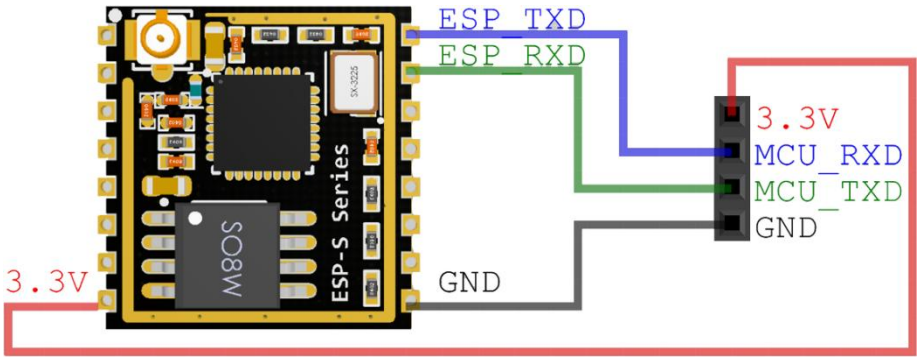


图 5：ESP-07S 接线示意图

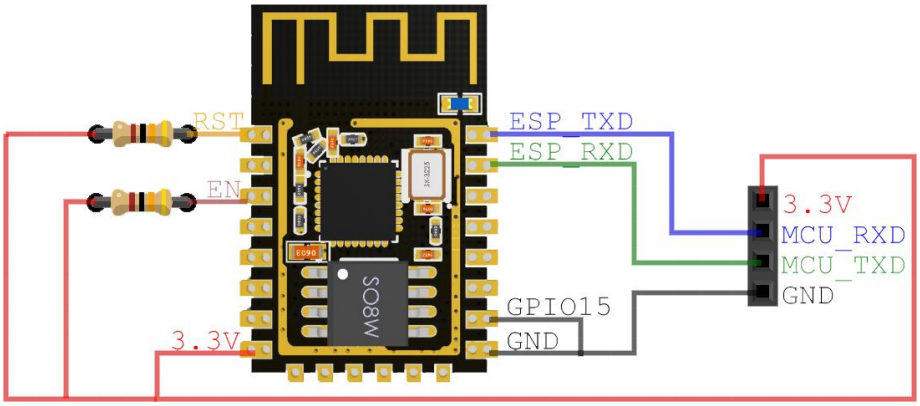


图 6：ESP-12F 接线示意图

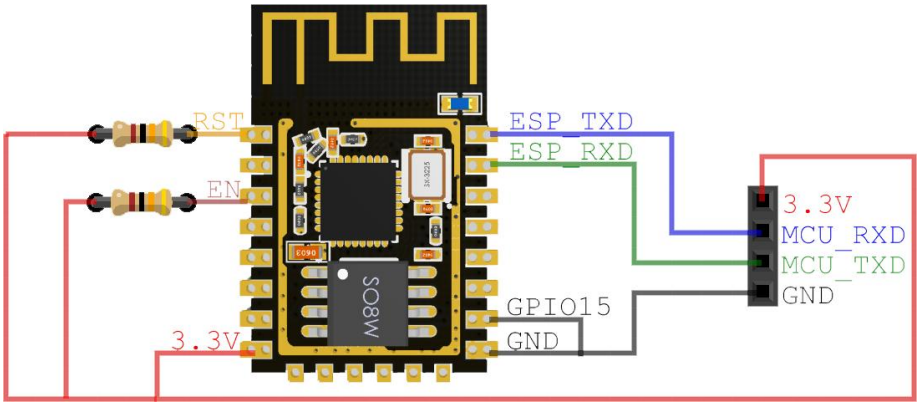


图 7：ESP-12S 接线示意图

电话：18923720150

邮箱：2355526548@qq.com

三、烧录

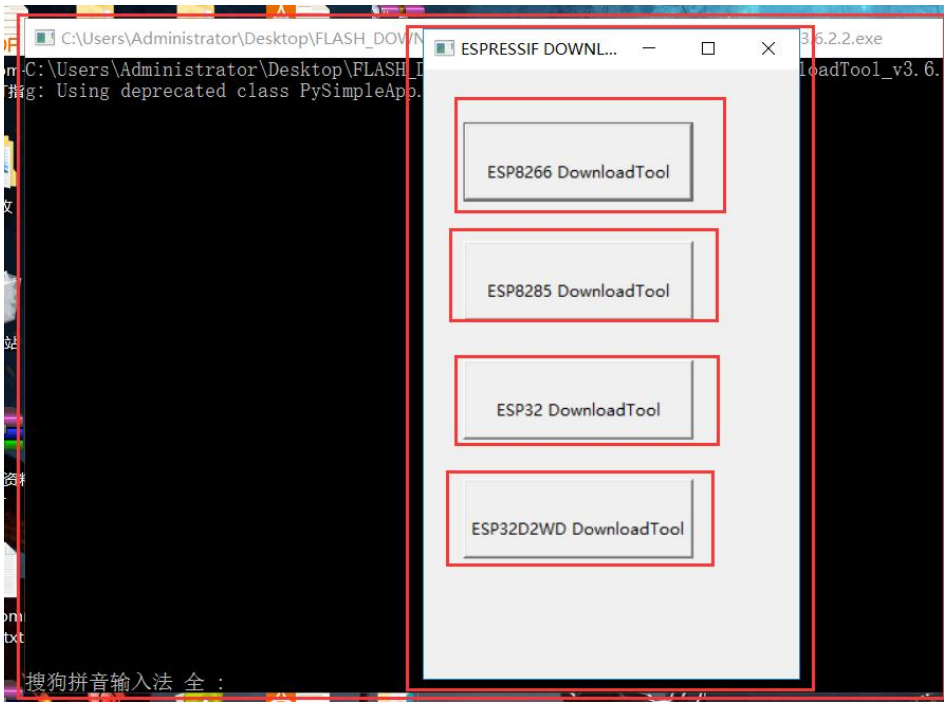
本章节主要讲解如何使用烧录工具为模组下载程序，其中包括：

- 下载软件的配置选项说明
- 完整的固件所包含的文件说明
- 如何将下载所需的文件打包合成一个完整的固件
- 烧录失败的原因

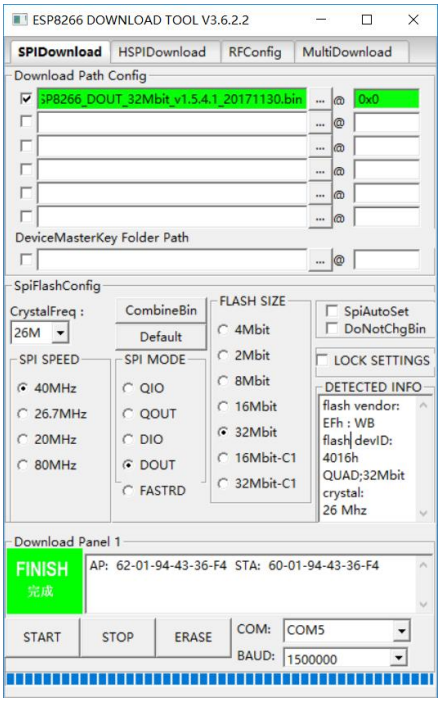
3.1 软件说明

该软件可以为 esp8266，esp8255（ESP-01M,ESP-01F 为 8285 的芯片，下载时请选择 esp8285），esp32 刷写固件，用户可以将编译生成的文件下载到模块中指定的位置。以刷写 ESP8266 固件示例，具体操作步骤如下。

1. 下载、解压并执行 ESPFlashDownloadTool_vx.xx.xx.exe，将弹出如下界面。黑窗可以查看烧录过程的一些信息，另一个为登录界面，用来选择相应芯片的下载界面。



2.点击 Esp8266 Download Tool，弹出如下界面



固件烧写软件配置参数说明如下：

配置选项	配置说明
Download_Path_Config	选择要下载的文件以及下载地址，点击 start 下载勾选后的文件
CrystalFreq	设置晶振频率。8266 的晶振频率为 26M，此处禁止修改
SPI_SPEED	设置 SPI 速率，默认 40M，此处禁止修改
SPI_MODE	设置 SPI 下载模式，默认为通用下载模式 DOUT
Flash_Size	设置 flash 容量(参考下一小节 3.2 章节) 根据实际编译的配置对应选择的 Flash 大小 16Mbit-C1 为 1204+1024 的布局，32Mbit-C1 为 1204+1024 的布局
CombineBin	打包合并固件,下载地址为 0x0000。参考章节 3.3
DoNotChgBin	● 选择该项，Flash 的运行频率、方式、布局会以用户编译时的配置为准 ● 未选择该项，Flash 的运行频率、方式、布局会以下载工具最终的配置为准

电话：18923720150 邮箱：2355526548@qq.com

	● 下载安信可官方 AT 固件时建议勾选，其他不建议勾选
Lock settings	选择该项，将锁住配置页面 该选项一般在工厂生产中使用，避免操作过程中改动了软件上的配置，造成生产问题
Default	选择该项，将恢复默认的软件配置。
Detected info	该窗口将会显示 flash 的大小和晶振频率
MAC address	● 该窗口将会显示 ESP8266 芯片的 MAC 地址，包括 STA MAC ADDRESS 和 AP MAC ADDRESS。 ●
COM	设置 COM 口
BAUD	设置下载波特率 下载时可以适当降低下载波特率，保证稳定下载（有些串口工具不支持 1500000 的波特率下载）
START	点击该按钮，开始烧录程序
STOP	点击该按钮，停止烧录程序
ERASE	点击该按钮，擦除整个 flash

表 3：下载配置参数说明

3.2 固件烧写

按照烧录文件的不同，分为两种情况：支持云端升级、不支持云端升级。另外，根据 flash 容量的不同，还需要调整 bin 文件的烧录地址。在安信可官网下载的 AT 固件都是打包合并过的固件，参考章节 3.3

不支持云端升级 （NoBoot 模式）

文件名称	8Mbit 地址分配	16Mbit 地址分配	32Mbit 地址分配	备注
eagle.flash.bin	0x00000	0x00000	0x00000	主程序，由代码编译生成
eagle.irom0text.bin	0x40000	0x40000	0x40000	主程序，由代码编译生成
esp_init_data_default.bin	0xFC000	0x1FC000	0x3FC000	由乐鑫在 SDK 中提供
blank.bin	0xFE000	0x1FE000	0x3FE000	由乐鑫在 SDK 中提供

表 4：NoBoot 模式下载地址表

注意：乐鑫不同版本的 SDK 中可能会改变 eagle.irom0text.bin 文件的烧录地址，以控制台输出的地址为准。

支持云端升级 （Boot 模式）

文件名称	8Mbit 地址分配	16Mbit 地址分配	32Mbit 地址分配	备注
boot.bin	0x00000	0x00000	0x00000	由乐鑫在 SDK 中提供，建议一直使用最新版本
user1.bin	0x01000	0x01000	0x01000	主程序，由代码编译生成
user2.bin	0x81000	0x81000	0x81000	主程序，由代码编译生成
esp_init_data_default.bin	0xFC000	0x1FC000	0x3FC000	由乐鑫在 SDK 中提供
blank.bin	0xFE000	0x1FE000	0x3FE000	由乐鑫在 SDK 中提供

表 5：Boot 模式下载地址表

注意：支持云端升级的固件，在 flash 布局会分为两个区，一个用来执行程序，另一个用来保存要升级的固件。当程序运行在 user1 时开始升级，程序会下载到 user2 区域，下载完毕后，下次启动运行 user2 的程序，依次替换，实现云端升级。

- User1.Bin 文件和 User2bin 文件烧录时只烧录其中一个
- Boot.bin 文件使用最新版本

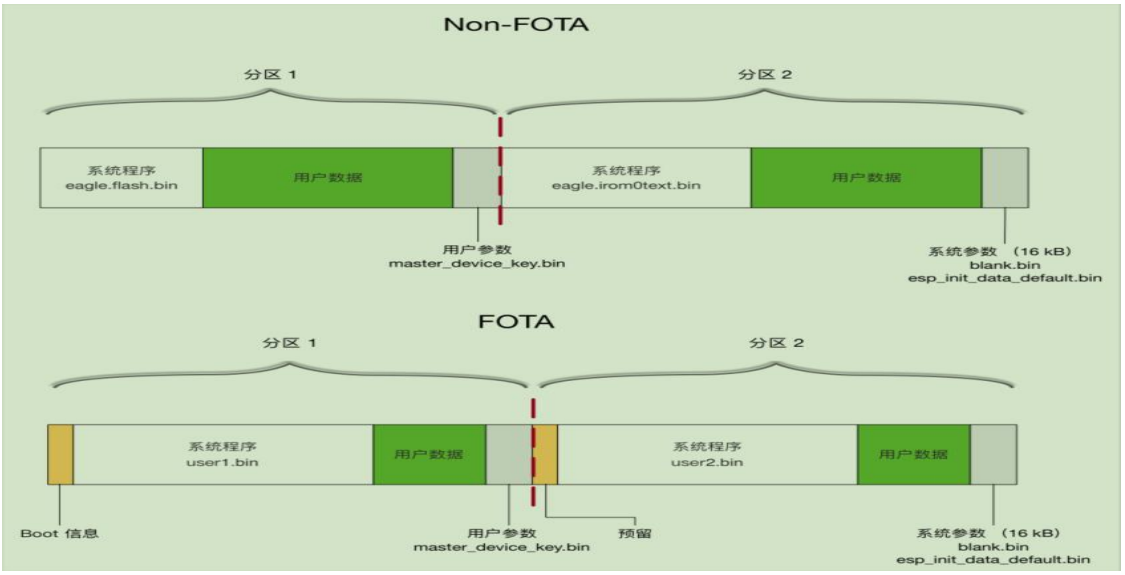


图 8：flash 布局说明

电话：18923720150

邮箱：2355526548@qq.com

分区说明：

- **系统程序：**用于存放运行系统必要的固件
- **用户数据：**当系统数据未占满整个 flash 空间时，空闲区域可用于存放用户数据。
- **用户参数：**地址由用户自定义，IOT_Demo 中设置为 0x3C000 开始的 4 个扇区，用户可以设置为任意未占用的地址
- **系统阐述：**固件 flash 的最后 4 个扇区
  - Blank.bin 下载地址为 Flash 的倒数第 2 个扇区
  - Esp_init_data_default.bin 下载地址为 flash 的倒数第四个扇区
- Boot 信息：位于 FOTA 固件的分区 1，存放 FOTA 升级相关信息
- **预留：**位于 FOTA 固件的分区 2，与分区 1 Boot 信息区对应的预留区域
- User.bin 说明 user1.bin 和 user2.bin 是同一个应用程序，选择不同的编译步骤分别生成的两个固件，存放在 SPI Flash 不同位置，启动时先运行行 Boot，Boot 读取系统参数区中的标志位，判断运行行 user1.bin 还是 user2.bin，然后到 SPI Flash 的对应位置读取运行行

3.3 烧写示例：

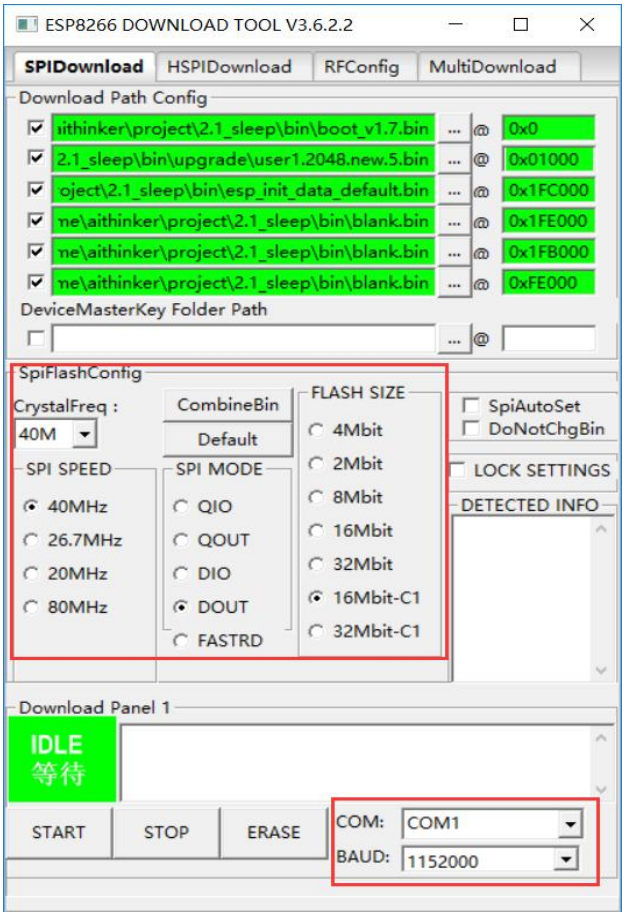


图 9：16M flash 烧写配置

以 16Mbit –C1 Flash , 1024-1024 map 为说明，烧录如下：

BIN	烧录地址	说明
boot.bin	0x0000	主程序
esp_init_data_defauit.bin	0x1FC000	初始化其他射频参数区，至少烧录一次 当 RF_CAL 参数区初始化烧录时，本区域也会烧录
user1.2048.new.5.bin	0x01000	主程序
blink.bin	0xFE000	初始化用户参数区
blink.bin	0x1FE000	初始化系统参数区
blink.bin	0x1FB000	初始化 RF_CAL 参数区

表 6：16M flash 烧写配置表

电话：18923720150 邮箱：2355526548@qq.com

3.4 如何确认是否进入下载模式

在第二章节有讲到模组启动模式，在我们烧录固件之前，最好先确认一下，模组是否进入下载模式。

1. 首先确保模组可以正常运行，发送 AT 指令（AT 固件）可以有回复 OK（即确保电源和串口都是正常的）；
2. 参考章节 2.2 接线，在 74880 波特率下观察模组启动或复位后的打印信息；
3. 若出现以下信息则认为模组已经进入了下载模式，可以进行下载；（参考章节 4.1 ， 74880 波特率下输出的系统日志信息说明）

图 10： 下载模式

3.5 烧录失败的原因

烧录导致失败的原因一般分为几种。

- COM 口选择错误或者 COM 被占用；
- 电源电压不稳定；
- 如果卡在了等待上电过程，在确认接线无误的情况下，将 RST 引脚接地复位一下即可；
- 串口芯片选型不对，该模组的串口电平为 TTL 电平，串口芯片建议使用类似 CH340、CP210X 等芯片。不要使用 232、485 甚至 PC 的九针孔接口来烧录；
- 串口不稳定，接入串口时，一定要把地线接上；
- 下载软件的 flash_size 选项超过了模组实际的 flash 大小，即 8M 的 flash

按照 32M 烧录肯定是不可以的；

- 下载波特率过大。有部分串口芯片的下载波特率并不支持 1500000，甚至由于接线方式的原因、使用的烧录线品质较差、线太长等原因导致太高的波特率下载容易失败，需要适当降低下载波特率；
- efuse 损坏。由于静电的原因导致芯片损坏，下载软件的 efuse 校验无法通过。可以参考一下黑窗输出的信息；
- 当 wifi 模组的串口已经连接了用户产品的 MCU 串口，此时为 wifi 模组烧录固件建议切断 MCU 与 wifi 模组的串口连接或者将 MCU 的复位引脚拉低，让 MCU 处于复位模式；

3.6 固件合并说明

使用软件上的 CombineBin 按键可以将文件打包合并成一个完整的固件。

Esp 系列模组在烧录固件时是按照要烧录的文件地址烧录对应文件的大小到 flash，其他部分的 flash 未改动，例如 user1.bin 文件为 320K，从 flash 地址 0x01000 地址开始烧录，烧录 320K 字节，如果第二次烧录的时候，编译生成的 user1.bin 只有 300k，那么比对上一次烧录的 user1.bin 文件在 flash 中的存储，后面的 20K falsh 是不会擦除的。

这样的烧录方式在大批量生产中是不安全的，尤其是有部分客户会在 flash 中添加自己的一些数据直接烧录进去。

将所有的固件打包合并成一个完整的固件，烧录时会填充整个 flash，对应地址没有程序部分的 flash 会被 0xFF 填充。

3.6.1 配置合并

固件主要分为不支持线升级和支持在线升级两种，具体地址分配表如下。

（1）不支持云端升级（Noboot 模式）

文件名称	8Mbit 地址分配	16Mbit 地址分配	32Mbit 地址分配
eagle.flash.bin	0x00000	0x00000	0x000
eagle.irom0text.bin	0x40000	0x40000	0x40000
esp_init_data_default.bin	0xFC000	0x1FC000	0x3FC000
blank.bin	0xFE000	0x1FE000	0x3FE000
blank.bin（见备注）	0xFF000	0x1FF000	0x3FF000

表 7：不支持云端升级

注意：乐鑫不同版本的 SDK 中可能会改变 eagle.irom0text.bin 文件的烧录地址，以控制台输出的地址为准。

（2）支持云端升级（Boot 模式）

文件名称	8Mbit 地址分配	16Mbit 地址分配	32Mbit 地址分配
Boot.bin	0x00000	0x00000	0x000
user1.bin	0x01000	0x01000	0x01000
esp_init_data_default.bin	0xFC000	0x1FC000	0x3FC000
blank.bin	0xFE000	0x1FE000	0x3FE000
blank.bin（见备注）	0xFF000	0x1FF000	0x3FF000

表 8：支持云端升级

电话：18923720150 邮箱：2355526548@qq.com

3.6.2 配置示例

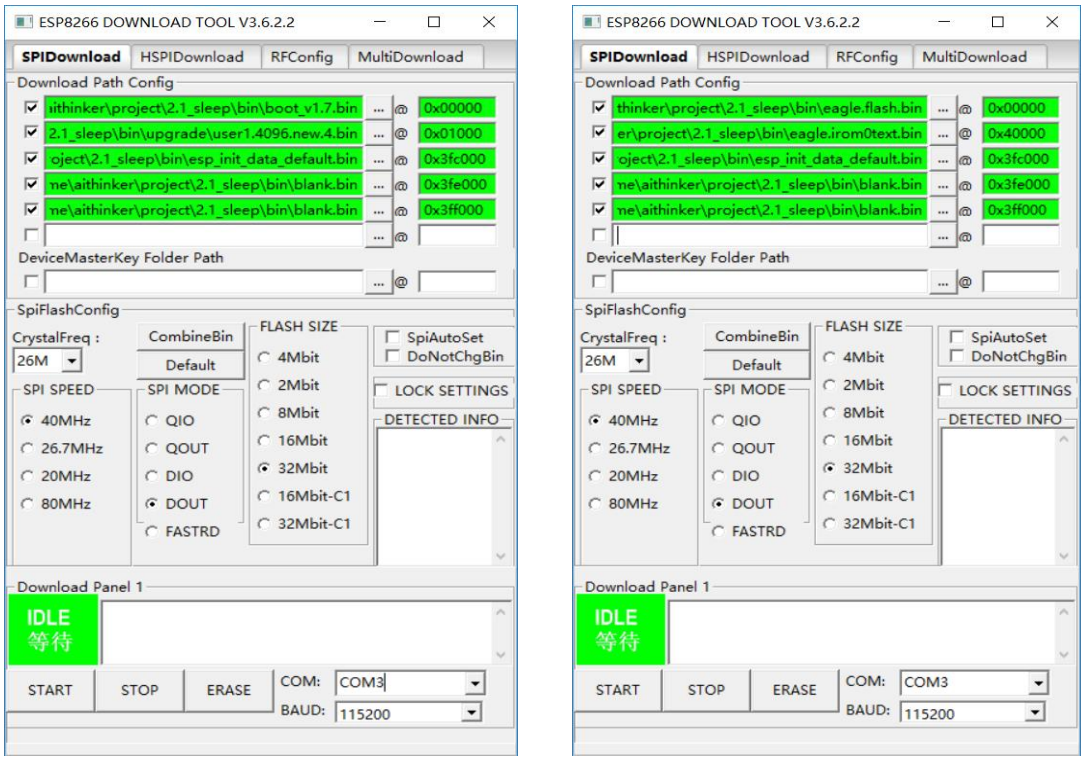


图 11：下载配置

备注：

- 需要将 blank.bin 放置到所需要使用的 Flash 的最后一个扇区，以确保 Flash 量产烧录时可以被完全覆盖，避免生产时可能出现的一些问题。
- 为方便管理和生产，在合并固件时，必须将“SPI MODE”选择“DOUT”模式，切勿选择其他模式，如选择其他模式，届时可能会造成固件无法启动和运行。
- 乐鑫在不同版本的 SDK 中有可能会改变这些烧录位置，本说明仅为参考，需以开发时的 Console 输出信息为准。

3.6.3 合并固件

点击【CombineBin】按钮， 软件将自动按照上一节的配置生成一个 target.bin 文件。

电话：18923720150 邮箱：2355526548@qq.com

图 12：生成文件

在 windows 资源管理器下查看生成文件详细大小如下：

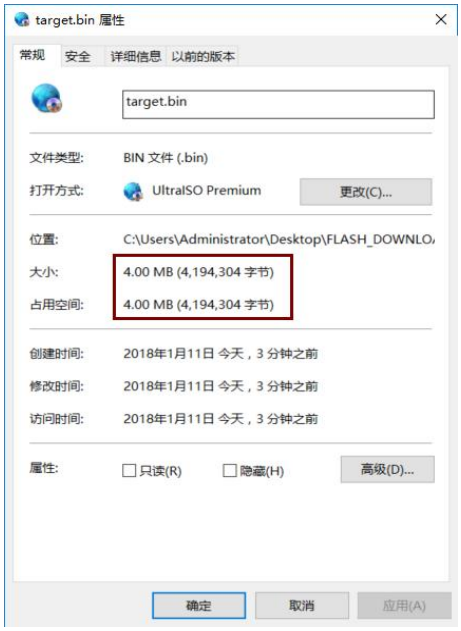


图 13：生成文件详细信息

使用 winhex 或者 ultraedit 类似软件查看得到如下数据，说明该文件大小是正确的（32Mbit Flash Size）

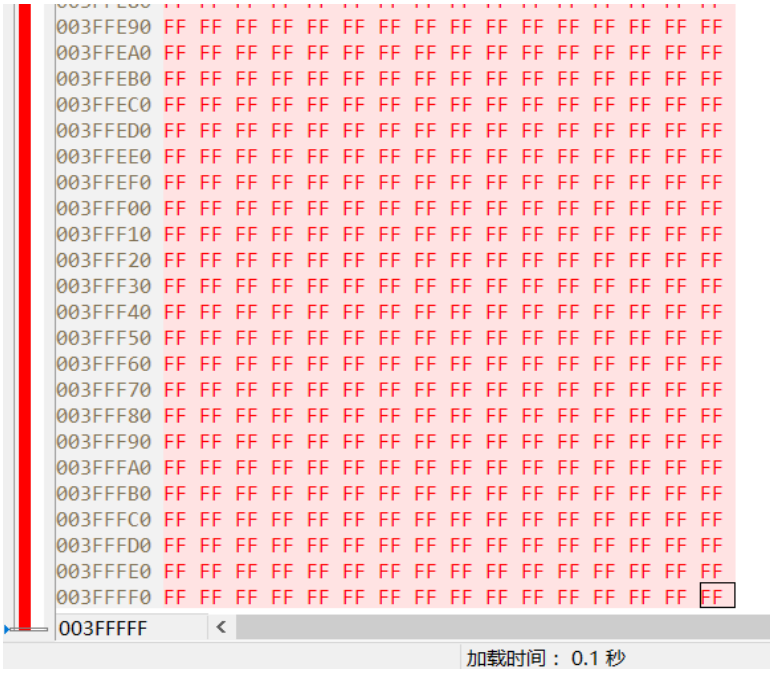


图 14：查看十六进制文件

3.6.4 校验固件

合并后，固件内已经包含了面板上的配置信息以及地址范围。烧录地址为0x00000。

注意：校验固件时必须选中 DoNotChgBin 选项，此时下载不受下载配置影响。

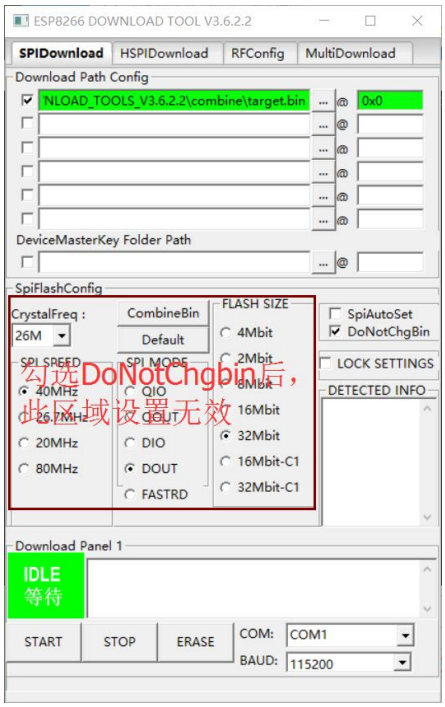


图 15：下载不受配置影响

电话：18923720150

邮箱：2355526548@qq.com

四、基础调试

本节主要涉及 esp8266 系列上电输出信息的一些基础讲解。

4.1 上电信息说明

Esp8266 系列模组出厂使用的 AT **固件，默认波特率为** 115200。

实际上，模组在上电过程中首先是在 74880 波特率下打印输出了系统日志信息，随后切换到 115200 波特率下完成初始化，当输出 ready 字样的字符串后，则表明初始化完成，此时可以发送 AT 指令去调试模组。

如下图，串口在 115200 波特率下首先输出一段乱码，随后输出了 Ai-Thinker Technology Co. Ltd. Ready。此时固件启动完成。这一串乱码可以在 74880 波特率下查看系统日志信息。

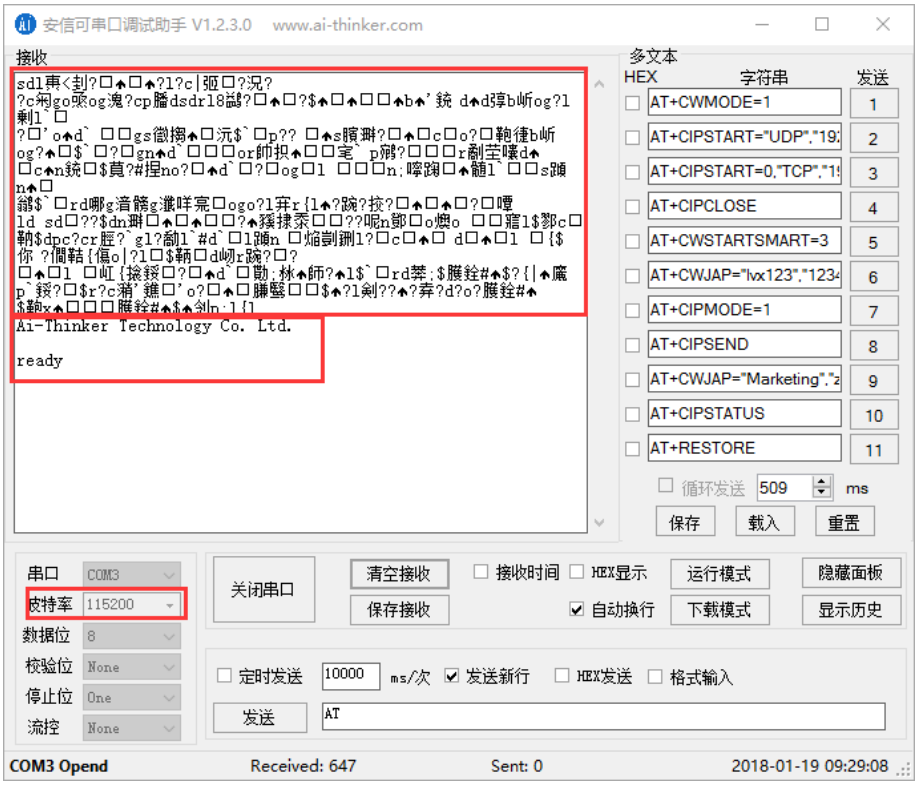


图 16：115200bps 下开机信息

电话：18923720150 邮箱：2355526548@qq.com

74880 波特率输出系统日志信息

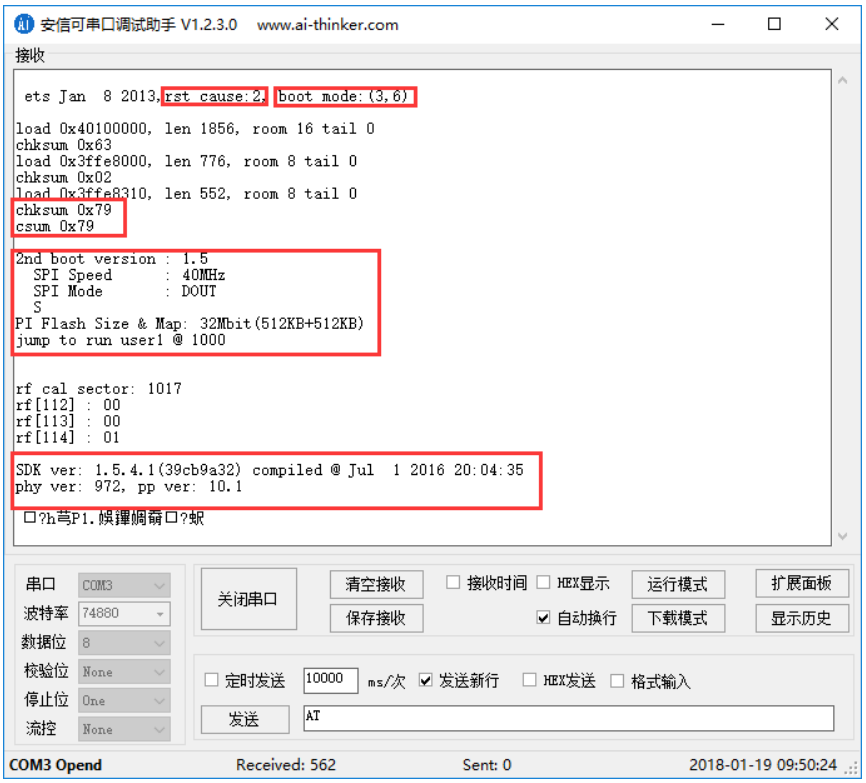


图 17：74880 下启动信息

- rst cause : 1 上电
  2 外部复位
  4 硬件看门狗复位
- Boot mode : 启动模式后面有两个参数，只看第一个参数即可
  1 下载模式
  3 运行模式
- chksum ： chksum 与 csum 值相等，表明启动过程中 Flash 读值正确

4.2 各种状态的启动信息说明

8266 在实际使用过程中由于用户的接线方式、烧录方式以及固件编写的方式不同，会有不同的输出信息，通常我们在 74880 波特率下输出的系统日志信息来分析。下面列举集中常见的输出信息。

4.2.1 运行模式

- **波特率：** 74880
- **固件：** 任意固件
- **描述：** boot mode : 3 表明该模式为模组的正常运行状态。

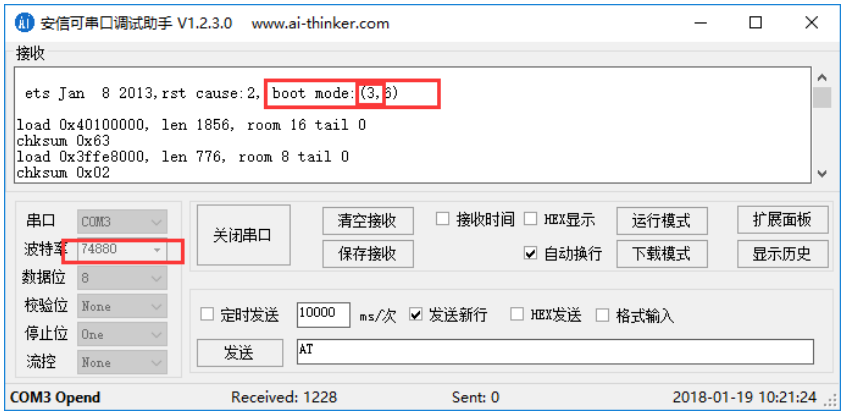


图 18：正常模式

4.2.2 下载模式

- **波特率：** 74880
- **固件：** 任意固件
- **描述：** boot mode : 1 表明该模式为模组的下载模式，当出现该字样时，表明模组进入了下载模式

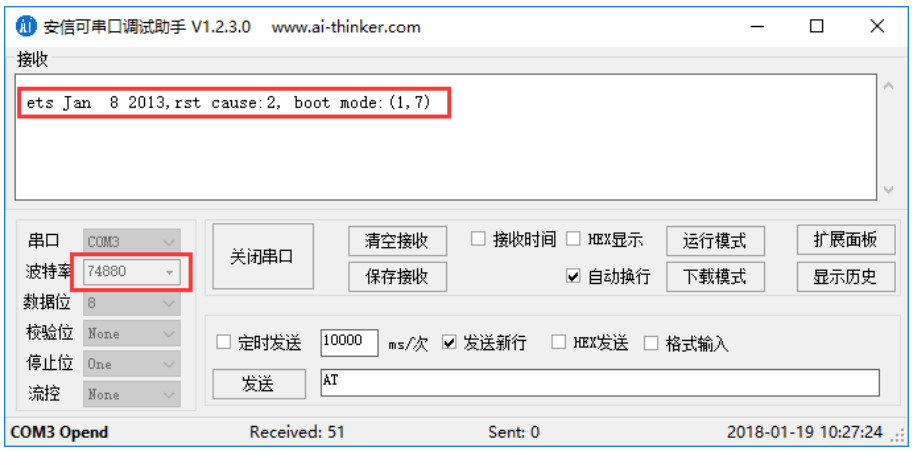


图 19：下载模式

电话：18923720150 邮箱：2355526548@qq.com

4.2.3 Waiting for host

- **波特率：**74880
- **固件：** 任意固件
- **描述：** waiting for host 意味着启动引脚电平不对，请参考 2.2 章节的启动模式的引脚电平说明来接线。

图 20：引脚错误

4.2.4 ets_main.c

- **波特率：**74880
- **固件：** 任意固件
- **描述：** est_main.c 意味着固件出现异常，一般为静电导致的模组固件损坏，或者烧录的时候 0x0 地址的 boot 文件烧录错误。

图 21：异常模式

4.2.5 Fatal exception （x）

- **波特率：**工作波特率
- **固件：** 任意固件
- **描述：** Fatal exception (x)出现的原因较多，一般为自己开发的 SDK 固件程序崩溃或者烧录错误，出现类似的错误首先检查一下是不是烧录固件过程中出现了错误，其次参考一下该文档

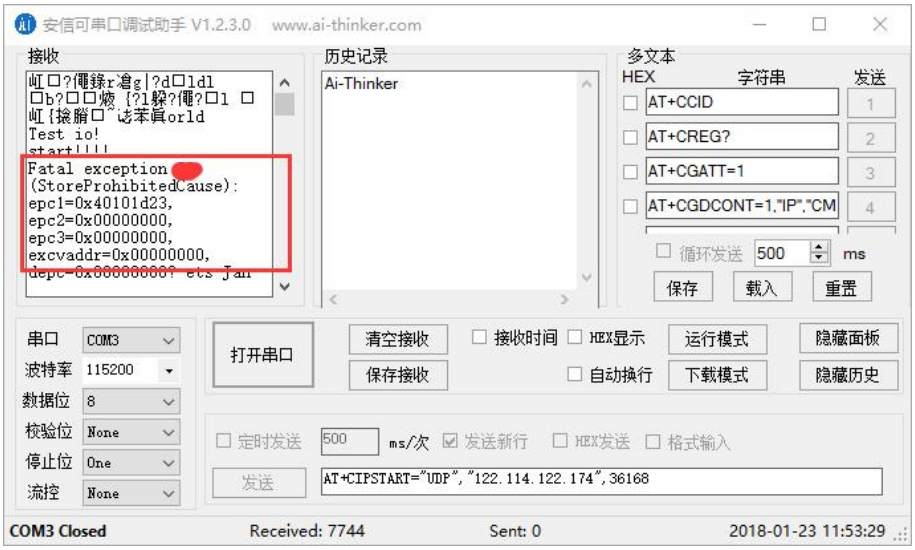


图 22：异常模式

五、演示 DEMO

该章节主要讲述几种演示 DEMO、AT 的调用方式和可实现的功能。

5.1 演示 DEMO

DEMO1：基于 Ai Cloude 2.0 智能配网实现远程控制（支持 web 配网）

测试 APP ：Ai smart

测试固件 ：Ai Cloud 2.0 AT 固件（默认的出厂固件不能连 AiCloude2.0 云服务器）

测试服务器： Ai Cloud 2.0(注意，该服务不支持商用，仅作为演示)

注意事项

示例链接

DEMO2：基于透传云实现远程控制

测试 APP ：Netassist(android ios)

测试固件 ：任意 AT 固件

测试服务器 透传云 ：

注意事项

示例链接

DEMO3：基于机智云平台调节灯光

测试 APP ：机智云 APP（第一次点击创建账号，第二次点击可以下载 APP 页面）

测试固件 ：机智云固件（非 AT 固件）

测试硬件 ：机智云开发板（任意 8266 系列都可以，涉及到 PWM 调节灯光，建议使用配套的机智云开发板）

测试服务器 ：机智云服务器

5.2 简单示例

测试 APP ：Netassist(android ios)

测试软件 ：安信可串口调试助手,PC 端网络调试助手

测试固件 ：AiCloud 2.0 AT 固件

测试硬件 ： Nodemcu 开发板（任意 8266 系列都可以）

5.2.1 TCP 相关的 demo

NO.01 STA 模式

ESP8266 Client:

->	AT+CWMODE=1	//设置为 STA 模式
->	AT+CWJAP_DEF="FAE","123456789."	//连接无线网络 wifi
->	AT+CIPSTART="TCP","192.168.0.116",8888	//连接服务端
->	AT+CIPSEND=5	//启动发送

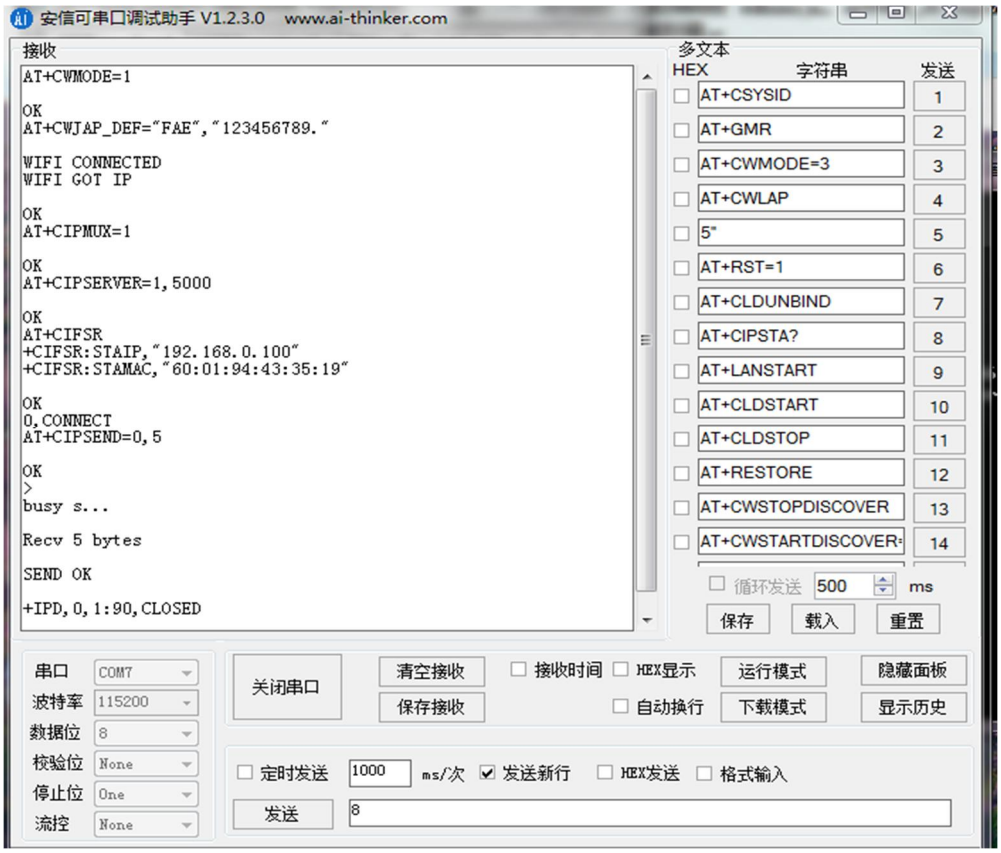
-> 手机需连接相同的 wif 如下图，并开启服务

NetAssist Server:

NO.02 ESP8266 Server:

->	AT+CWMODE=1	//设置为 STA 模式
->	AT+CWJAP_DEF="FAE","123456789."	//连接无线网络 wifi
->	AT+CIPMUX=1	//开启多连接
->	AT+CIPSERVER=1,5000	//开启服务端
->	AT+CIFSR	//查看 mac 地址和 IP
->	AT+CIPSEND=0,5	//启动发送

-> 手机需连接相同的 wif 如下图，并开启服务



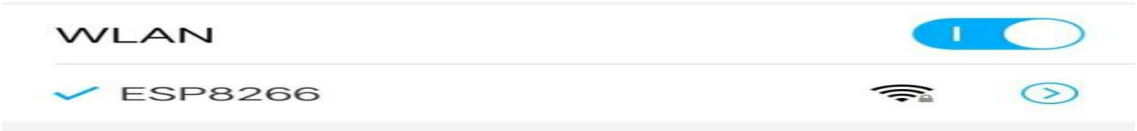
NetAssist Client:

NO.03 AP 模式

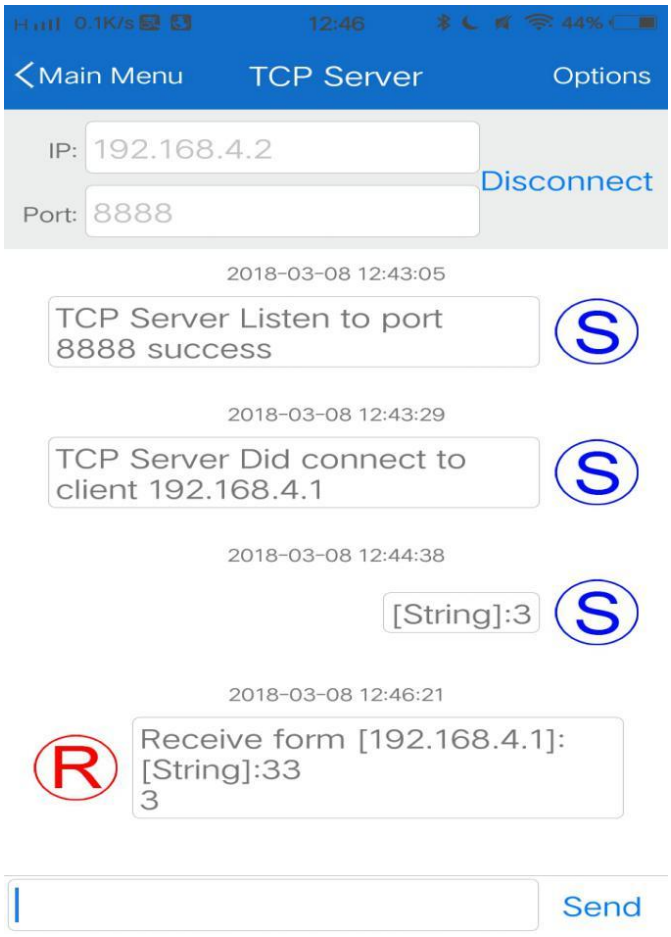
ESP8266 Client:

->	AT+CWMODE=2	//设置为 AP 模式
->	AT+CWSAP="ESP8266","12345678",11,3	//开启 wifi 热点
->	AT+CIPSTART="TCP","192.168.4.2",8888	//连接服务端
->	AT+CIPSEND=5	//启动发送

-> 在手机端连接 ESP8266 的热点



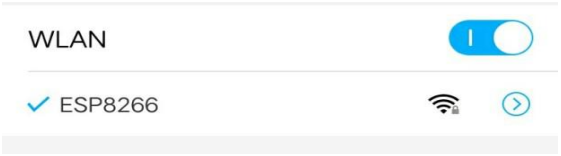
NetAssist Client:



NO.04 ESP8266 Server:

->	AT+CWMODE=2	//设置为 AP 模式
->	AT+CWSAP="ESP8266","12345678",11,3	//开启 wifi 热点
->	AT+CIPMUX=1	//开启多连接
->	AT+CIPSERVER=1,5000	//开启服务端
->	AT+CIFSR	//查看 mac 地址和 IP
->	AT+CIPSEND=0,5	//启动发送

-> 在手机端连接 ESP8266 的热点



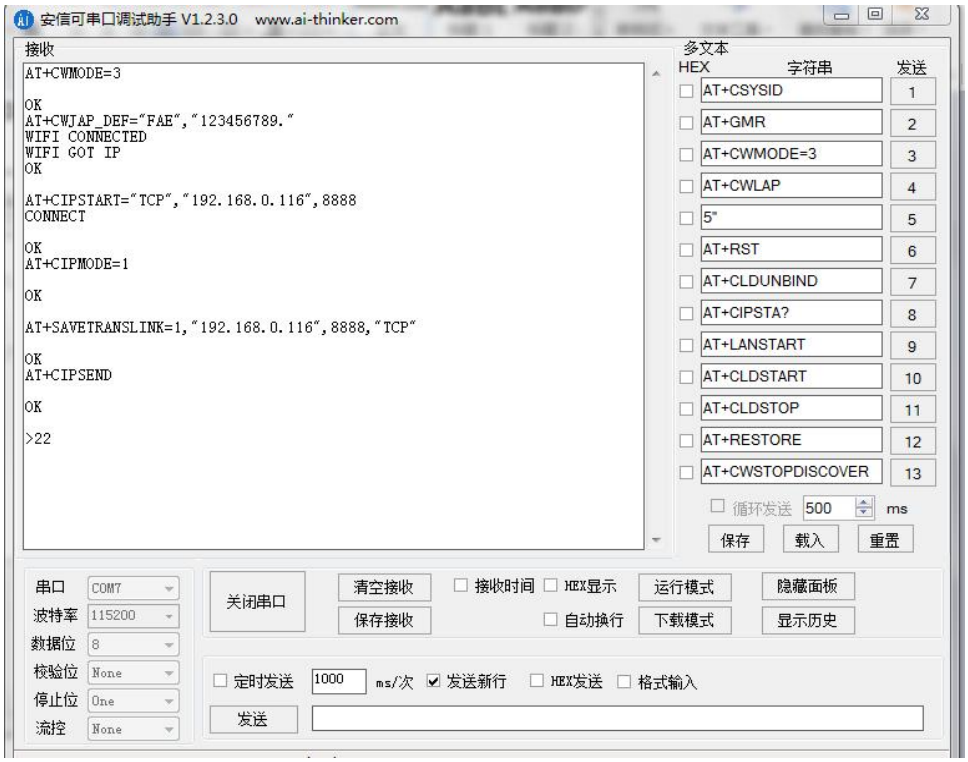
NetAssist Client:

NO.05　STA+AP 模式

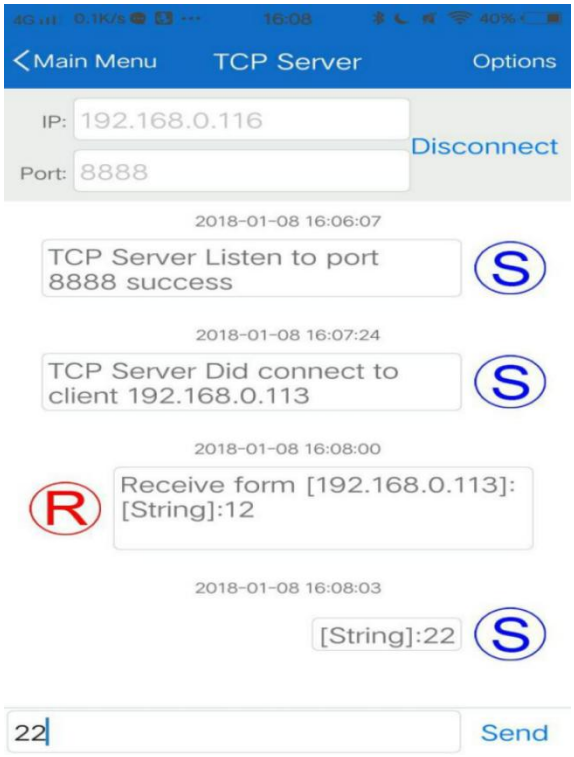
ESP8266 Client:(注意：本 demo 加入了保存透传)

->	AT+CWMODE=3	//设置为 AP+STA 模式
->	AT+CWJAP_DEF="FAE","123456789."	//连接无线网络 wifi
->	AT+CIPSTART="TCP","192.168.0.116",8888	//连接服务端
->	AT+CIPMODE=1	//进入透传模式
->	AT+SAVETRANSLINK=1,"192.168.0.116",8888,"TCP"	//保存透传设置
->	AT+CIPSEND	//启动发送

*手机需连接相同的 wif 如下图，并开启服务



NetAssist Client:



电话：18923720150 邮箱：2355526548@qq.com

NO.06 ESP8266 Server:

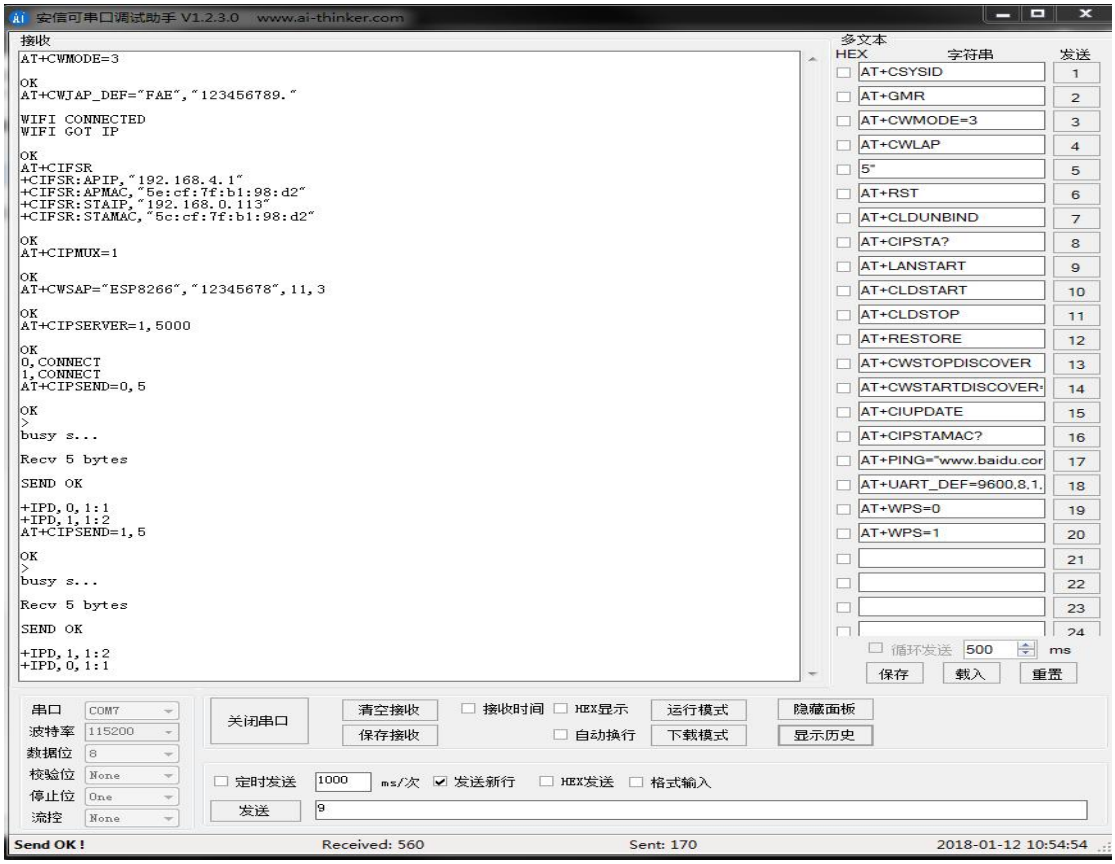
->	AT+CWMODE=3	//设置为 AP+STA 模式
->	AT+CWJAP_DEF="FAE","123456789."	//连接无线网络 wifi
->	AT+CWSAP="ESP8266","12345678",11,3	//开启 wifi 热点
->	AT+CIPMUX=1	//开启多连接
->	AT+CIPSERVER=1,5000	//开启服务端
->	AT+CIFSR	//查看 mac 地址和 IP
->	AT+CIPSEND=0,5	//向客户端 0 发送 5 个数据
->	AT+CIPSEND=1,5	//向客户端 1 发送 5 个数据

(注意：本 demo 加入了多连接)

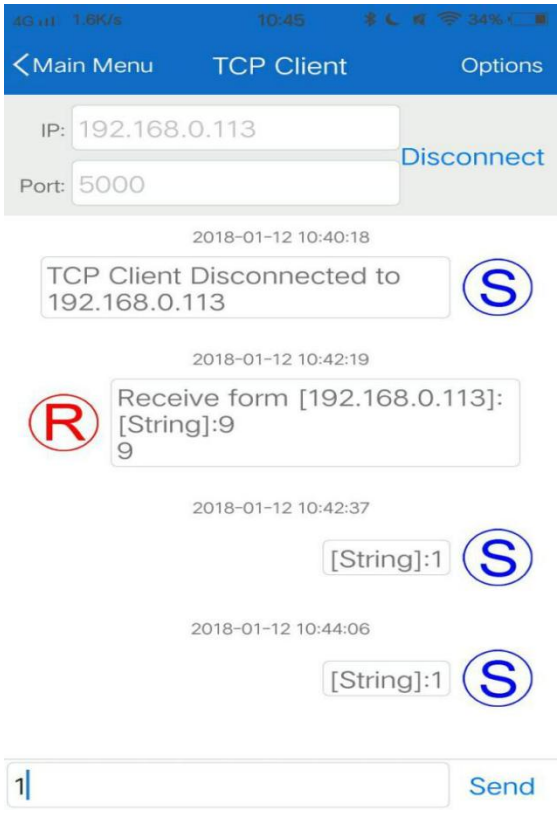
-> 手机需连接相同的 wifi，如下图

-> 在手机端连接 ESP8266 的服务端

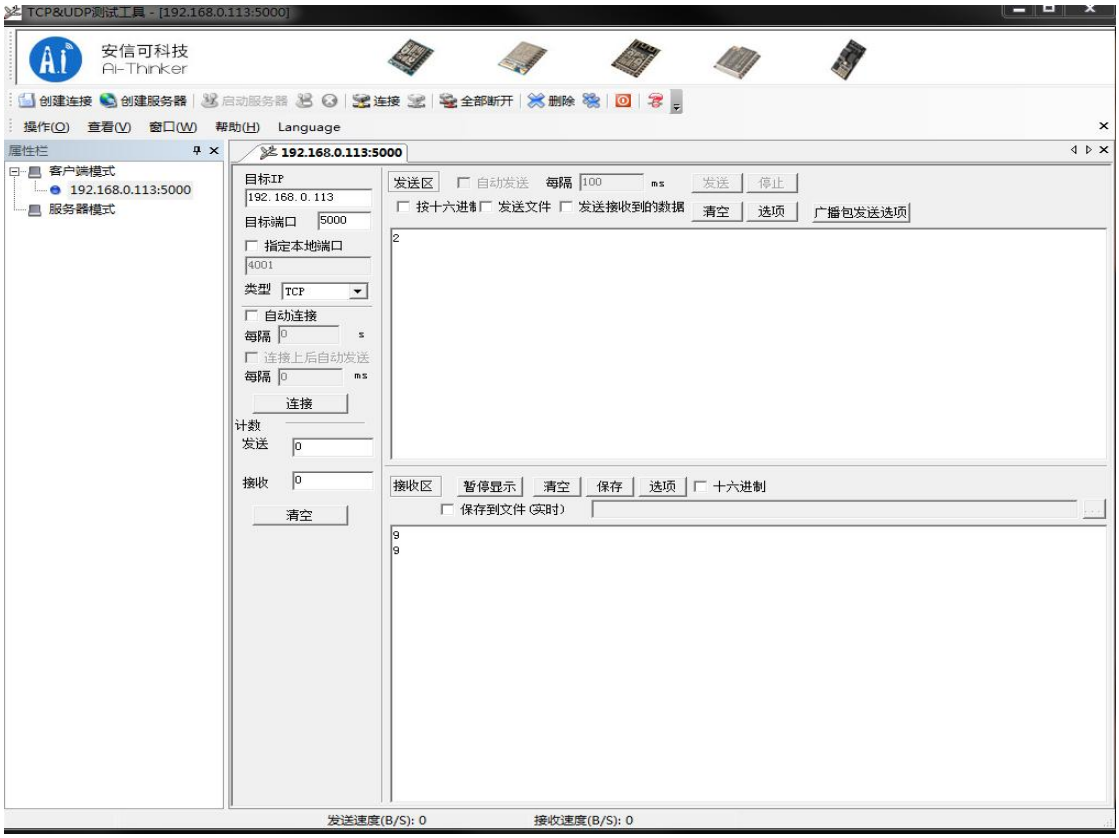
-> 在 PC 端连接 ESP8266 的服务端



NetAssist Client0:



PC Client1:



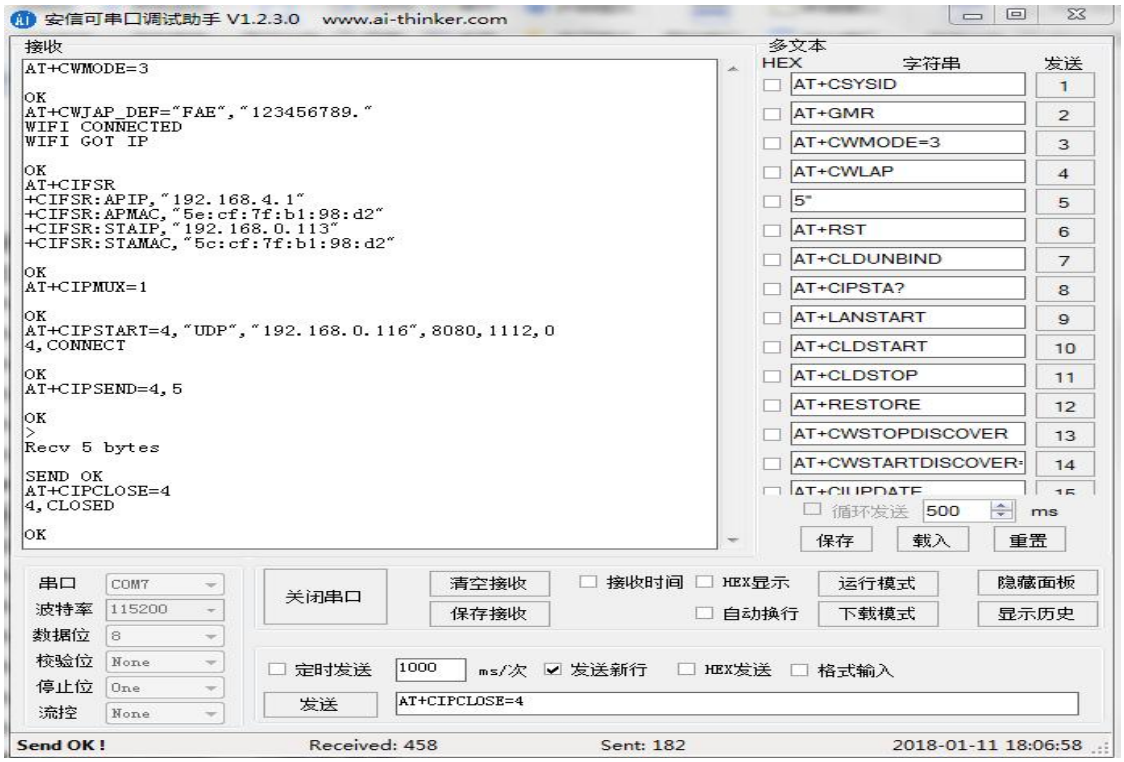
5.2.2 UDP 相关的 demo

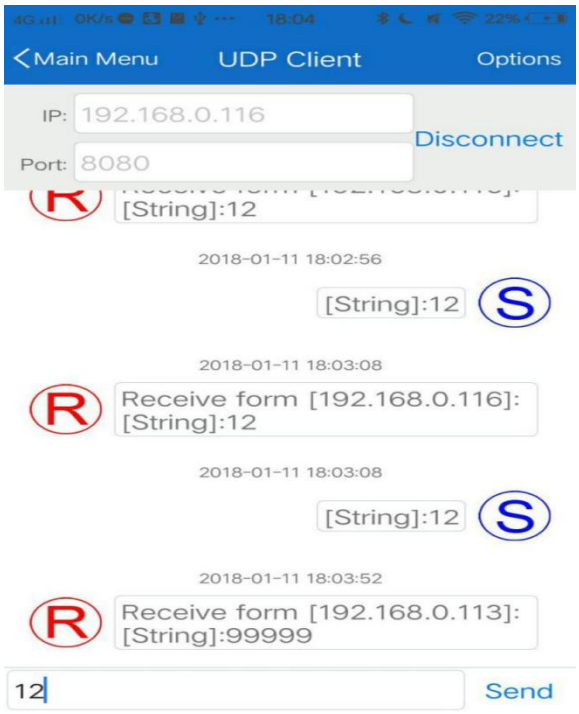
(1) 固定远端

-> AT+CWMODE=3	//设置 AP+STA 模式
-> AT+CWJAP_DEF="FAE","123456789."	//连接网络
-> AT+CIFSR	//查询设备 IP 地址
-> AT+CIPMUX=1	//使能多连接
-> AT+CIPSTART=4,"UDP","192.168.0.116",8080,1112,0	//创建 UDP,最后的参数 0 表示固定远端
-> AT+CIPSEND=4,5	//发送数据
-> AT+CIPCLOSE=4	//断开 UDP 传输

-> 手机需连接相同的 wifi，如下图


WLAN
FAE


安信可串口调试助手 V1.2.3.0 www.ai-thinker.com
接收
AT+CWMODE=3
OK
AT+CWJAP_DEF="FAE","123456789."
WIFI CONNECTED
WIFI GOT IP
OK
AT+CIFSR
+CIFSR:APIP,"192.168.4.1"
+CIFSR:APMAC,"5e:cf:7f:b1:98:d2"
+CIFSR:STAIP,"192.168.0.113"
+CIFSR:STAMAC,"5c:cf:7f:b1:98:d2"
OK
AT+CIPMUX=1
OK
AT+CIPSTART=4,"UDP","192.168.0.116",8080,1112,0
4,CONNECT
OK
AT+CIPSEND=4,5
OK
>
Recv 5 bytes
SEND OK
AT+CIPCLOSE=4
4,CLOSED
OK
多文本
HEX
字符串
发送
AT+CSYSID
AT+GMR
AT+CWMODE=3
AT+CWLAP
5"
AT+RST
AT+CLDUNBIND
AT+CIPSTA?
AT+LANSTART
AT+CLDSTART
AT+CLDSTOP
AT+RESTORE
AT+CWSTOPDISCOVER
AT+CWSTARTDISCOVER
循环发送
500
ms
保存
载入
重置
串口
COM7
波特率
115200
数据位
8
校验位
None
停止位
One
流控
None
关闭串口
清空接收
保存接收
接收时间
HEX显示
自动换行
运行模式
下载模式
隐藏面板
显示历史
定时发送
1000
ms/次
发送新行
HEX发送
格式输入
发送
AT+CIPCLOSE=4
Send OK !
Received: 458
Sent: 182
2018-01-11 18:06:58


Main Menu
UDP Client
Options
IP: 192.168.0.116
Port: 8080
Disconnect
[String]:12
2018-01-11 18:02:56
[String]:12
2018-01-11 18:03:08
Receive form [192.168.0.116]:
[String]:12
2018-01-11 18:03:08
[String]:12
2018-01-11 18:03:52
Receive form [192.168.0.113]:
[String]:99999
12
Send

(2) 远端可变

```
->  AT+CWMODE=3                                                    //设置 AP+STA 模式
OK
->  AT+CWJAP_DEF="FAE","123456789."                               //连接网络
WIFI CONNECTED
WIFI GOT IP
OK
->  AT+CIFSR                                                      //查询设备 IP 地址
+CIFSR:APIP,"192.168.4.1"
+CIFSR:APMAC,"5e:cf:7f:b1:98:d2"
+CIFSR:STAIP,"192.168.0.107"
+CIFSR:STAMAC,"5c:cf:7f:b1:98:d2"
OK
->  AT+CIPSTART=4,"UDP","192.168.0.116",8080,1112,2   //创建 UDP，最后的参数 2 表示远端
可变
4,CONNECT
OK
->  AT+CIPMODE=1                                                  //进入透传模式
      OK
->  AT+SAVETRANSLINK=1,"192.168.0.116","UDP",8080                 //保存透传设置
        OK
->  AT+CIPSEND=5                                                  //发送数据
        OK
->  AT+CIPCLOSE=4                                                 //断开 UDP 传输
4,CLOSED
        OK
```

5.2.3 HTTP 相关的 demo

-> AT+GMR //启动查询版本信息

AT version:1.4.0.0(May 5 2017 16:10:59)

SDK version:2.1.0(116b762)

Ai-Thinker Technology Co., Ltd.

Integrated AiCloud 2.0 v0.0.0.7s

Aug 4 2017 19:55:15

OK

-> AT+CWMODE_DEF=1 //配置 WIFI 模组为单 STA 模式并保存

OK

-> AT+CWJAP_DEF="FAE","123456789." //连接无线路由 wifi

WIFI CONNECTED

WIFI GOT IP

OK

-> AT+CWAUTOCONN=1 //使能上电自动连接 AP

OK

-> AT+CIPSTART="TCP","www.baidu.com",80 //连接网络

CONNECT

OK

-> AT+CIPMODE=1 //设置透传

OK

-> AT+CIPSEND //启动发送

OK

-> > GET 请求

GET /devices/5835707 HTTP/1.1

api-key: xUrvOCDB=iRuS5noq9FsKrvoW=s=

Host:api.heclouds.com

\r\n\r\n(结束)

回应：

HTTP/1.1 200 OK

Date: Sun, 14 Dec 2018 13:13:25 GMT

Content-Type: application/json

Content-Length: 213

Connection: keep-alive

Server: Apache-Coyote/1.1

Pragma: no-cache

{ “ errno ” :0, “ data ” :{ “ private ” :false, “ protocol ” : “ EDP ” , “ create_time ” : “ 2018-01-14 13:13:25 ” , “ online ” :false, “ location ” :{ “ lon ” :0, “ lat ” :0}, “ id ” : “ 5835707 ” , “ auth_info ” : “ Light001 ” , “ title ” : “ SLight ” , “ tags ” :[]}, “ error ” : “ succ ” }

-> >POST 请求

POST /devices/5835707/datapoints HTTP/1.1

api-key: xUrvOCDB=iRuS5noq9FsKrvoW=s=

Host:api.heclouds.com

Content-Length:60

\r\n

{ “ datastreams ” :[{ “ id ” : “ switch ” , “ datapoints ” :[{ “ value ” :1}]}]}|(结束)

回应：

HTTP/1.1 200 OK

Date: Sun, 14 Dec 2018 14:05:48 GMT

Content-Type: application/json

Content-Length: 26

Connection: keep-alive

Server: Apache-Coyote/1.1

Pragma: no-cache

{ “ errno ” :0, “ error ” : “ succ ” }

-> +++ //退出透传，不要勾选发送新行 如下图：

5.2.4 Smartconfig 配网相关的 demo

说明：使用两种方式进行配网，乐鑫 ESP-Touch 和微信 Airkiss。

准备：一块 NodeMCU 或 ESP8266 wifi 模组，ESP-Touch APP

*手机 APP 源码：https://github.com/espressifapp

注意：手机与电脑需连接同一个 wifi。

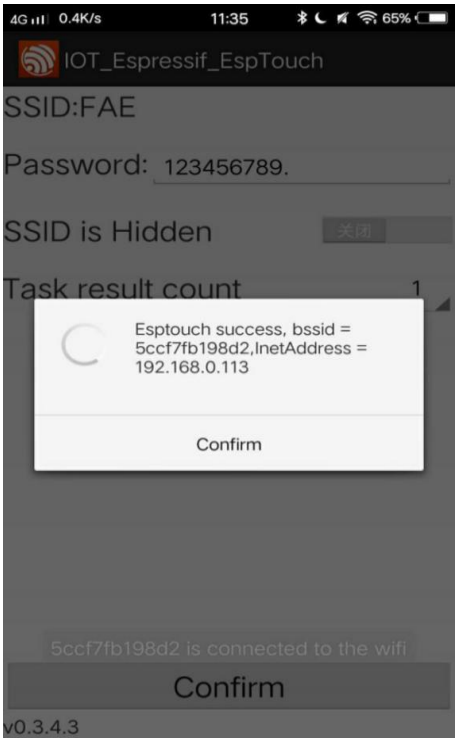
方式一：ESP-Touch

-> AT+CWMODE_DEF=1 //配置 WIFI 模组为单 STA 模式并保存

-> AT+CWAUTOCONN=1 //使能上电自动连接 AP

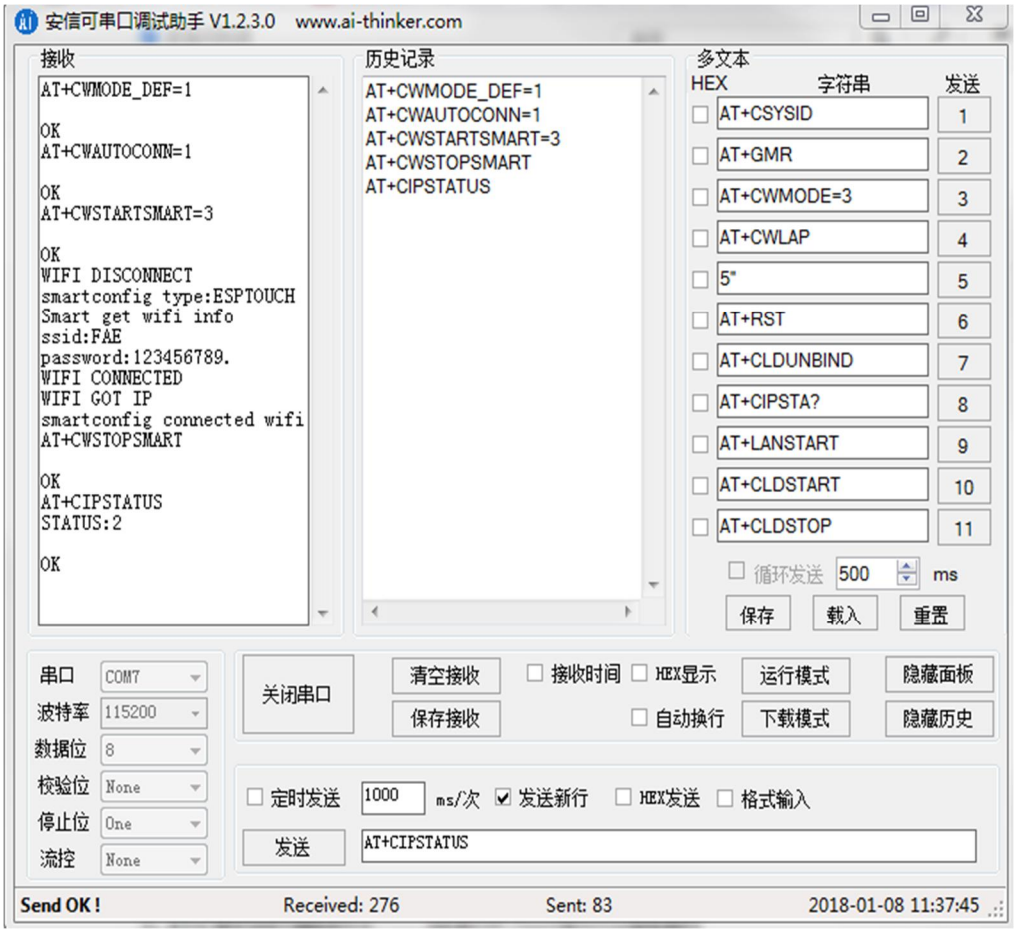
-> AT+CWSTARTSMART=3 //支持 ESP-Touch 和 Airkiss 智能配网

-> 手机连上需要的 AP，打开手机 APP ESP-Touch 输入密码，点击确定，等待配网成功，如下图：



-> AT+CWSTOPSMART //无论配网是否成功，都需要释放内存

-> AT+CIPSTATUS //查询网络连接状态，如下图：



5.2.5 Airkiss

-> AT+CWMODE_DEF=1 //配置 WIFI 模组为单 STA 模式并保存

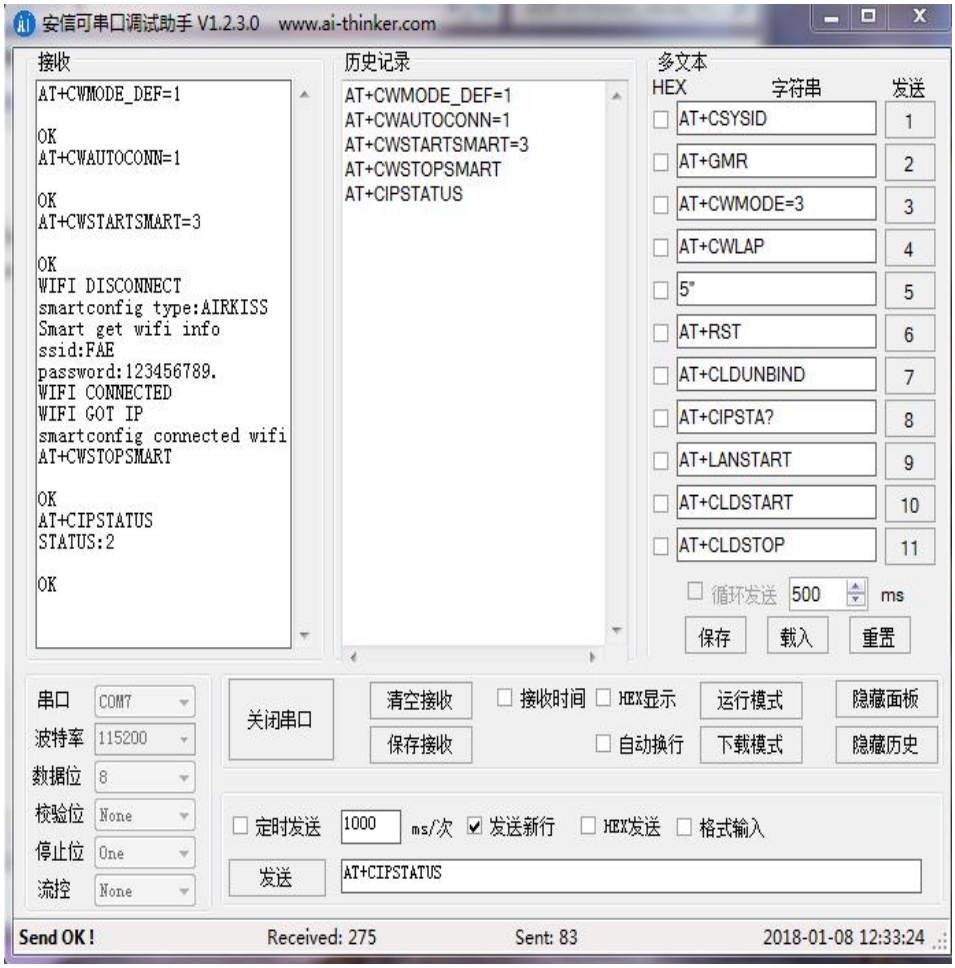
-> AT+CWAUTOCONN=1 //使能上电自动连接 AP

-> AT+CWSTARTSMART=3 //支持 ESP-Touch 和 Airkiss 智能配网

-> 打开微信，关注微信公众号“安信可科技”，点击 WIFI 设置，点击开始配置，输入密码，点击连接，如下图：

-> AT+CWSTOPSMART //无论配网是否成功，都需要释放内存

-> AT+CIPSTATUS //查询网络连接状态，如下图：



电话：18923720150 邮箱：2355526548@qq.com

5.2.6 两个 ESP8266 之间通信的 demo

注意：准备 2 个 ESP8266 的模组，分别将其设为 Server 和 Client

Server 端

-> AT+CWMODE=2 //设置为 AP 模式

-> AT+CIPMUX=1 //开启多连接

-> AT+CWSAP="ESP_8266","12345678",5,3 //开启 wifi 热点

-> AT+CIPSERVER=1,5000 //开启服务

-> AT+CIFSR //查看模组 MAC 地址和 IP 地址

-> AT+CIPSEND=0,5 //发送数据给客户端 0

Client 端

-> AT+CWMODE=1 //设置为 STA 模式

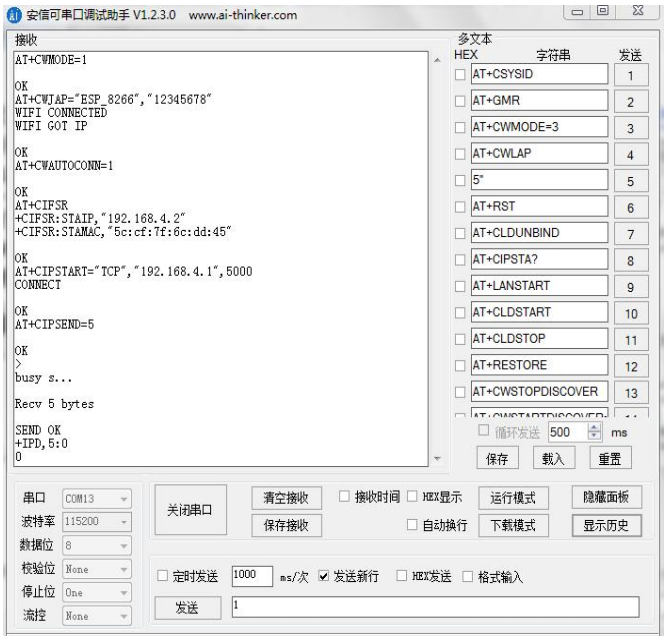
-> AT+CWJAP="ESP_8266","12345678" //链接 Server 端 ESP8266 模块的 wifi 热点

-> AT+CWAUTOCONN=1 //上电自动连接 AP

-> AT+CIFSR //查看模组 MAC 地址和 IP 地址

-> AT+CIPSTART="TCP","192.168.4.1",5000 // 连接 server 端

-> AT+CIPSEND=5 //发送数据



5.2.7 如何设置静态 IP

-> AT+CWDHCP_DEF=0,0 //保存设置为 AP 模式,后面的参数 0 为关闭 DHCP，可设置静态 IP。（若开启 DHCP，则静态 IP 无效。）

-> AT+CIFSR //查看模组 MAC 地址和 IP 地址

-> AT+CIPAP_DEF="192.168.5.109","192.168.5.1","255.255.255.0" //设置静态 IP 地址并保存

-> AT+CIFSR //查看模组 MAC 地址和 IP 地址

5.2.8 修改波特率及其恢复原始状态

-> AT+UART?

+UART:115273,8,1,0,1 (实际的波特率为 115200，有误差）

OK

-> AT+CIOBAUD=115200

OK

-> AT+UART_CUR=115200,8,1,0,3 //临时设置波特率，最后参数 3 表示 RTS&CTS 流控

OK

-> AT+UART_DEF=115200,8,1,0,3 //保存设置波特率，最后参数 3 表示 RTS&CTS 流控 ,使能流控需要硬件支持。

OK

注意：

设置波特率使用流控时，需要硬件支持流控。

- MTCK 为 UART0 CTS
- MTDO 为 UART0 RTS

否则建议使用：

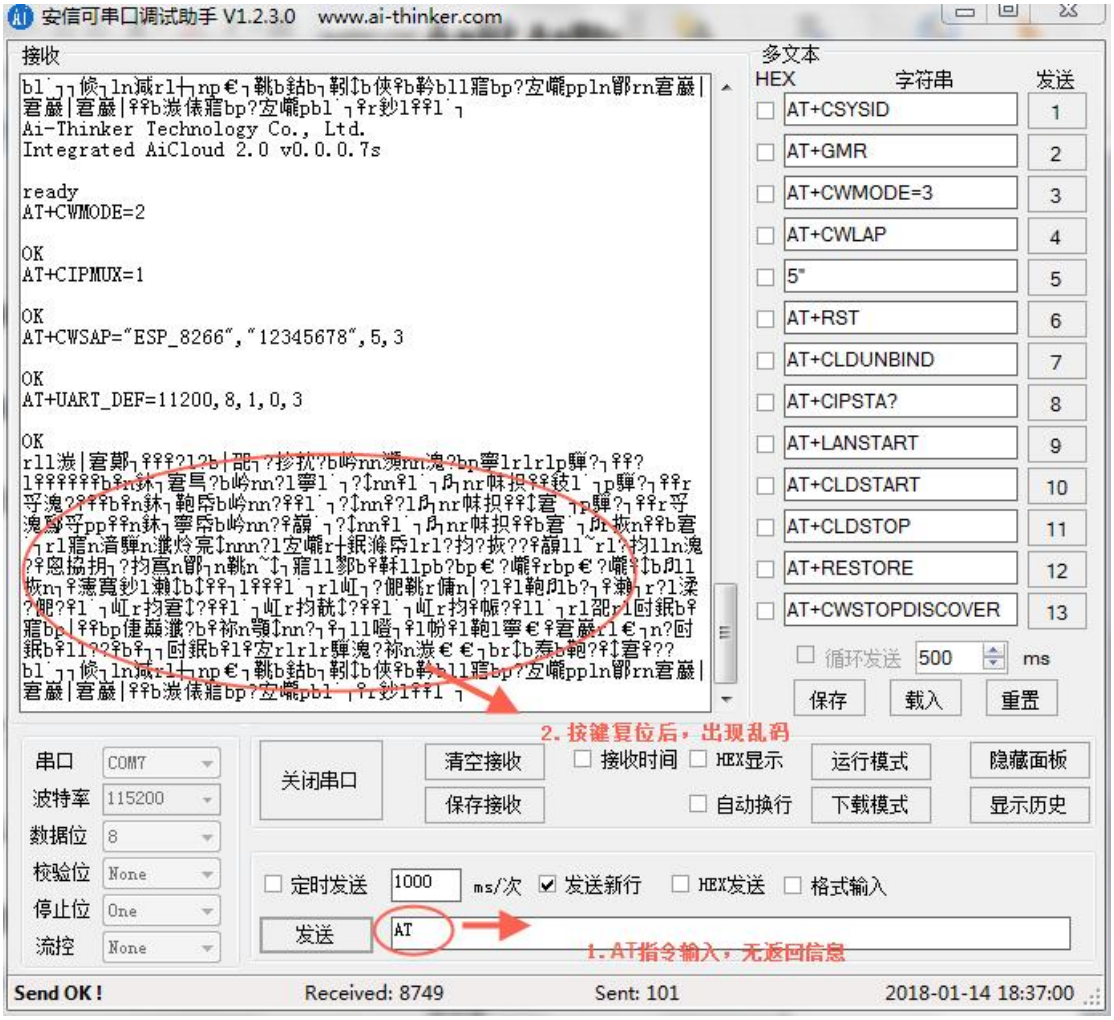
AT+UART_DEF=115200,8,1,0,0　　//非流控的波特率设置

波特率支持范围：110~115200*40

c. 在 AP 模式下设置波特率出现错误，如何恢复原来的波特率设置？

```
->  AT+CWMODE=2

OK

->  AT+CIPMUX=1

OK

->  AT+CWSAP="ESP_8266","12345678",5,3

OK

->  AT+UART_DEF=11200,8,1,0,3

OK
```

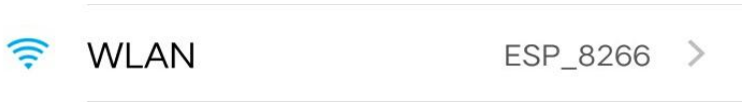
这样设置后，输入 AT 指令没有回应，然后重启如下图：



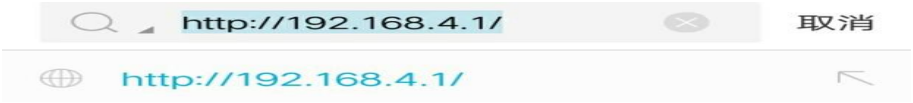
方法一、通过重新烧录原来的固件恢复正常。

方法二、通过访问浏览器恢复原来的设置。（注意：AiCloud 2.0 固件才支持）

1.用手机连接 ESP8266 开启的 wifi。



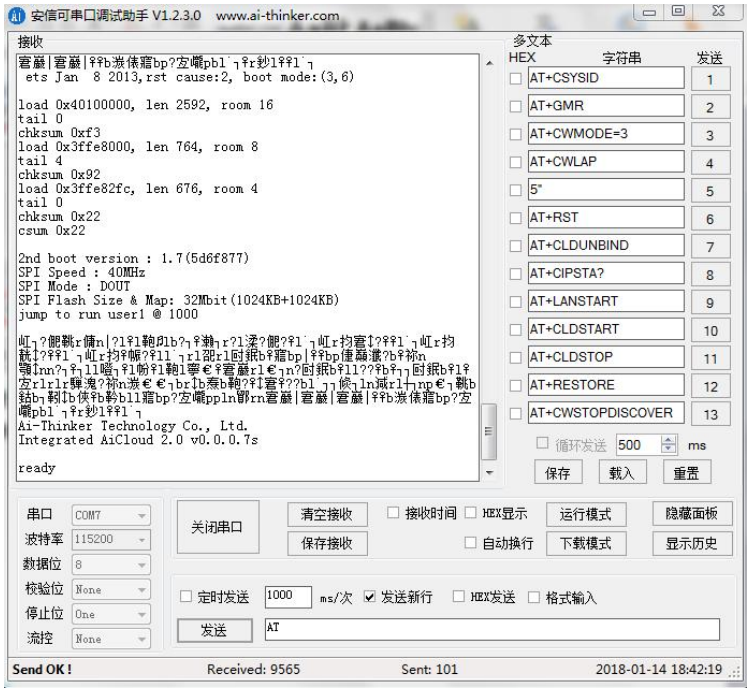
2.打开浏览器，输入 192.168.4.1，进入 ESP8266 WebConfig 网页。



3.翻页后，点击 Restore 按钮后保存。


ESP8266 Webconfig
Restore
Reboot
Station
' http://192.168.4.1/ '上的网页显示：
Save OK!
确定
Gateway:
0.0.0.0
Mac:
5c:cf:7f:b1:98:d2
Save

4.恢复正常，如下图:


安信可串口调试助手 V1.2.3.0 www.ai-thinker.com
接收
ets Jan 8 2013,rst cause:2, boot mode:(3,6)
load 0x40100000, len 2592, room 16
tail 0
chksum 0xf3
load 0x3ffe8000, len 764, room 8
tail 4
chksum 0x92
load 0x3ffe82fc, len 676, room 4
tail 0
chksum 0x22
csum 0x22
2nd boot version : 1.7(5d6f877)
SPI Speed : 40MHz
SPI Mode : DOUT
SPI Flash Size & Map: 32Mbit(1024KB+1024KB)
jump to run user1 @ 1000
Ai-Thinker Technology Co., Ltd.
Integrated AiCloud 2.0 v0.0.0.7s
ready
多文本
HEX
字符串
发送
AT+CSYSID
AT+GMR
AT+CWMODE=3
AT+CWLAP
5"
AT+RST
AT+CLDUNBIND
AT+CIPSTA?
AT+LANSTART
AT+CLDSTART
AT+CLDSTOP
AT+RESTORE
AT+CWSTOPDISCOVER
循环发送
500
ms
保存
载入
重置
串口
COM7
波特率
115200
数据位
8
校验位
None
停止位
One
流控
None
关闭串口
清空接收
保存接收
接收时间
HEX显示
自动换行
运行模式
下载模式
隐藏面板
显示历史
定时发送
1000
ms/次
发送新行
HEX发送
格式输入
发送
AT
Send OK!
Received: 9565
Sent: 101
2018-01-14 18:42:19

六、SDK 开发

安信可 ESP 系列一体化开发环境是安信可科技为方便广大用户而推出的基于 Windows + Cygwin + Eclipse + GCC 的综合 IDE 环境，安信可一体化开发环境有以下特点：

- 支持 ESP8266 NONOS 和 FreeRTOS 环境开发
- 支持 ESP31B/ESP32 FreeRTOS 环境开发
- 下载即用，无需另外配置环境
- 可直接编译所有乐鑫官方推出的 SDK 开发包

在官方提供的 SKD 中，在 example 文件夹下有官方提供的相关 DEMO，用户可以在相关 DEMO 的基础上进行二次开发，可以缩短开发周期，经 example 下的相关 demo 复制到根目录下编译即可生成.bin 文件，同时给出相关的下载地址，可以直接使用下载工具下载到相应的地址即可。

用户若编写自己的程序代码时需要先找到相应 demo 下的 user 文件夹下的 user_main.c 文件，程序的入口函数为此文件夹中的 user_init()函数，此函数相当于平时 C 语言中的 main 函数，用户可以在此函数中修改并添加用户代码。

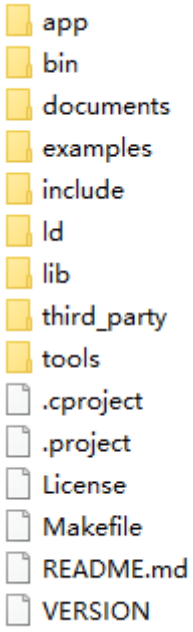
由于 ESP8266 默认开启看门狗，所以程序中不能存在死循环或延时较长的等待函数，如果需要较长时间的等待可以使用定时器来实现。

6.1 环境搭建

.8244740.0.0

6.2 代码编写

1. 目录解释

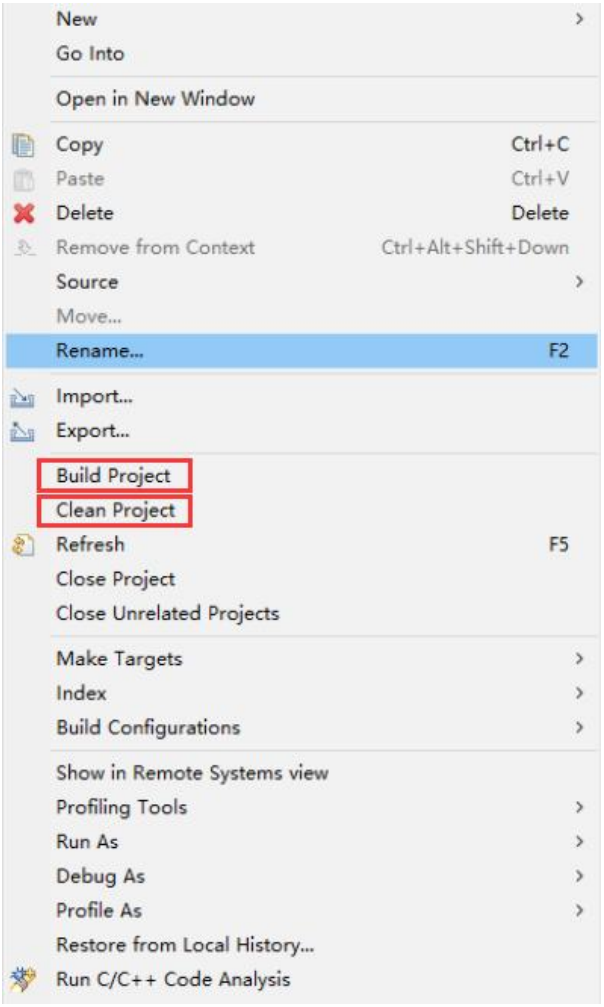


Bin	编译工程后生成的 bin 文件，可直接烧录在 flash 中

App	应用层目录，用户函数入口就在 app/user/user_main.c
Document	Sdk 相关的介绍以及链接
Examples	一些例程，例如：smartconfig
Include	在 SDK 中预先安装头文件。这些文件包含相关的 API 函数和其他宏定义。用户不需要修改它们。
Ld	链接器脚本。我们建议用户不修改它们
Lib	SDK 中提供的库文件。
Third_party	第三方库的 ESPRESIF 的开源代码，目前包括免费 RTOS，JSON，LWIP，MBOTTLS，NPOLL，OpenSSL，SIFFF，和 SSL。
Tools	编译二进制文件所需的工具。用户不需要修改它们。

2.工程的编译

Build project 前必须 clean project 若编译前修改过文件一定要保存后在编译（CTRL+S），点击工程顶部然后点击鼠标右键会出现下图菜单。



3.如何添加.c .h 的文件

在 sdk 中编写自己的文件就需要在 app 目录下添加相应的.c 或者.h 文件，c 文件添加到 app/user 下面.h 文件添加到 app/include 下面。

电话：18923720150 邮箱：2355526548@qq.com

FAQ

（1） 所有的 AT 指令后面都要添加指令结束标志“ \r\n ”,或者在串口调试工具中勾选发送新行选项,透传退出时发送“ +++”除外。

（2） 要连接透传云的 IP 地址和端口号在透传云接收框的下部，并且若 3 分钟内未连接需要刷新并使用新的 IP 和端口号

（3） AT+SAVETRANSLINK – 保存透传配置到 Flash（永久保存），下次开机时会自动进入透传模式，若取消上面配置需要先发送+++退出透传模式，然后发送 AT+RESTORE 回复出厂设置。

（4） ESP8266 作为服务器，手机端作为 TCP 客户端时，若连接的是 ESP8266 的 AP 则 IP 地址用 192.168.4.1，否则用 ESP8266 申请到的 IP。

（5） 一体化开发环境下载后的文件夹中是一个.exe 结尾的应用程序，直接点击运行无法正常运行，需要对该文件进行加压。

问题：

（1） Ai Smart 注册过程中出现内部错误。（从官网 wiki 下载 Android 工具 NetAssist

（2） UDP 相关 DEMO（远端可变）测试中使用

AT+CIPSTART=4,"UDP","192.168.40.108",8080,1112,2 连接会返回 Link type ERROR

把 linkID 去掉后连接正常

->AT+CIPSTART="UDP","192.168.40.108",8080,1112,2

->CONNECT

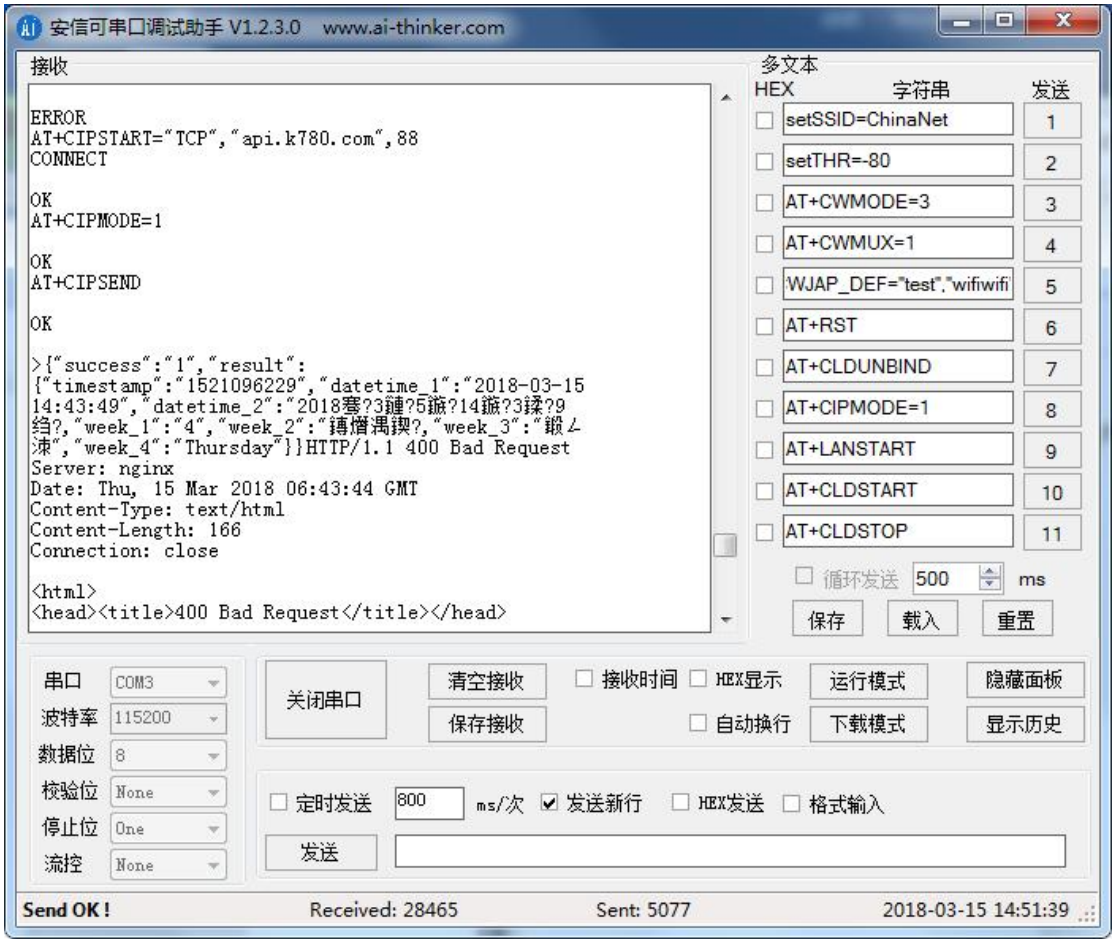
Demo:网上找的其他 HTTP 的使用（不同点：直接 get 链接）

HTTP 相关的 demo

-> AT+GMR //启动查询版本信息

AT version:1.4.0.0(May 5 2017 16:10:59)

```
SDK version:2.1.0(116b762)
Ai-Thinker Technology Co., Ltd.
Integrated AiCloud 2.0 v0.0.0.7s
Aug  4 2017 19:55:15
OK
-> AT+CWMODE_DEF=1   //配置 WIFI 模组为单 STA 模式并保存
OK
-> AT+CWJAP_DEF="test","wifiwifi"   //连接无线路由 wifi
WIFI CONNECTED
WIFI GOT IP
OK
-> AT+CWAUTOCONN=1   //使能上电自动连接 AP
OK
-> AT+CIPSTART="TCP","api.k780.com",88   //连接网络
CONNECT
OK
-> AT+CIPMODE=1   //设置透传
OK
-> AT+CIPSEND      //启动发送
OK
```



直接浏览器访问得到的结果如下：


```
success:        "1"
result:
   timestamp:   "1521095907"
   datetime_1:  "2018-03-15 14:38:27"
   datetime_2:  "2018年03月15日 14时38分27秒"
   week_1:      "4"
   week_2:      "星期四"
   week_3:      "周四"
   week_4:      "Thursday"
```


提示：通过串口发送正确的 Get 请求时会得到返回的数据，通过通过串口发送其他内容(无效的链接)会返回如下代码提示请求无效。

HTTP/1.1 400 Bad Request

Server: nginx

Date: Thu, 15 Mar 2018 06:54:21 GMT

Content-Type: text/html

```
Content-Length: 166
Connection: close

<html>
<head><title>400 Bad Request</title></head>
<body bgcolor="white">
<center><h1>400 Bad Request</h1></center>
<hr><center>nginx</center>
</body>
</html>
```