

刚拿到 ESP8266 模块串口没有输出或输出乱码怎么办

暨启动信息解释

问 1: 刚买了一个 ESP8266 的模块, 接好串口上电却没有看到串口输出, 怎么回事?

问 2: 刚买了一个 ESP8266 的模块, 接好串口上电却看到串口输出乱码, 怎么回事?

问 3: 刚买了一个 ESP8266 的模块, 发送 AT 指令却没有反应, 怎么回事?

答: 可能需要检查接法是否正确, 串口通信设置是否正确电平是否匹配, 以及模块硬件是否正常。

背景

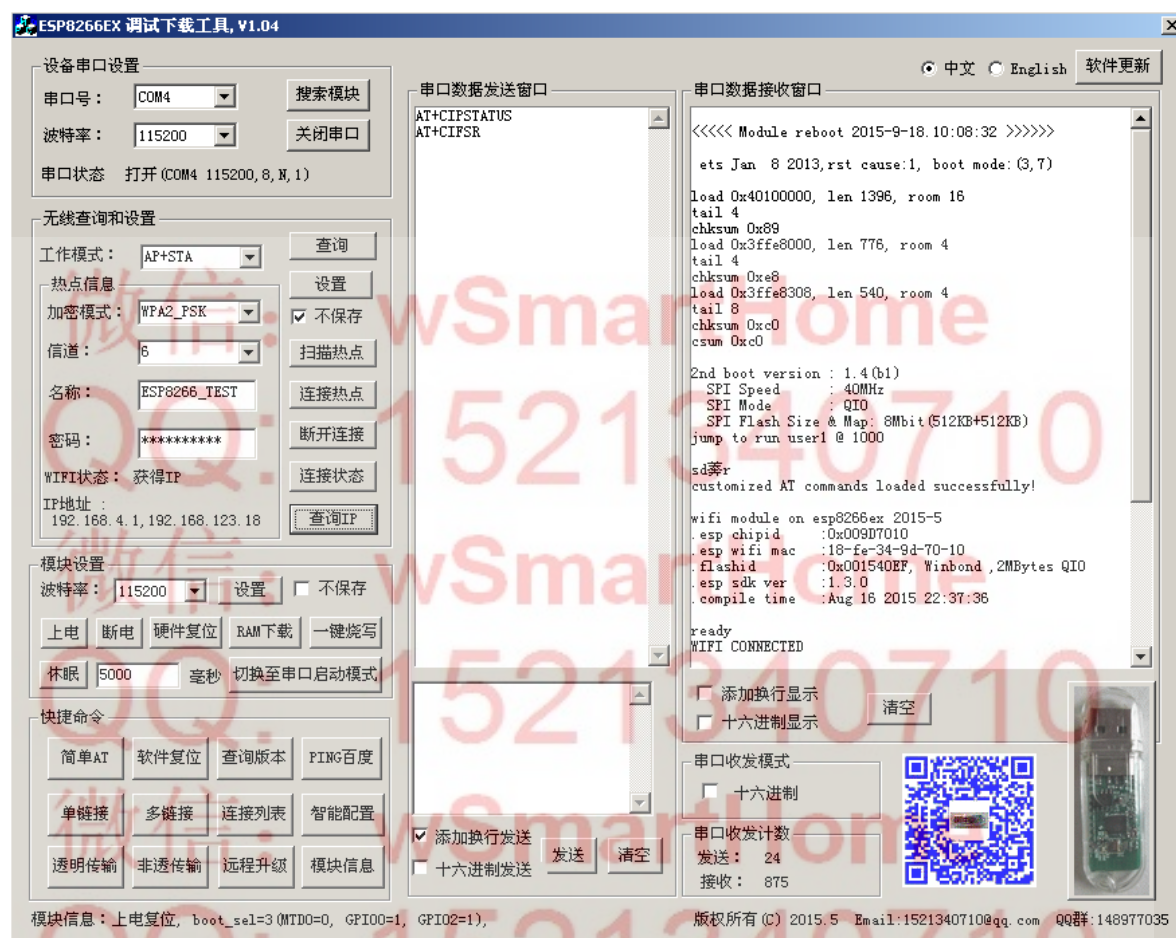
一些朋友在刚拿到模块时, 因为对模块本身不熟悉、或者因为需要自己手工外接串口没接好、或者所购买模块对外接电源要求过于严格等原因, 导致很难开始成功地使用模块进行试用, 在开始使用阶段, 往往浪费大量的时间。

根据某些群里的一些朋友刚拿到各种基于 ESP8266 芯片的模块后的一些问题汇总, 这里总结了利用本软件进行初步调试的一些小技巧, 大家可以按照下面的步骤一步步实验来缩小定位。

下面的方法, 适用于市面上大多数的模块。

辅助工具

ESP8266EX 调试与下载工具, 下载地址: <http://pan.baidu.com/s/1pJ2xTJd>



第一步 先仔细检查接线是否正确，在启动模块时至少保证以下管脚的连接：

管脚	上下拉	备注
GPIO15	必须下拉	这三个管脚在启动时会被采集以选择启动模式。正常从外部 FLASH 启动，这个管脚在启动的输入电平必须为 b011。又因为这三个管脚在复位时都有内部上拉，所以只需要 GPIO15 下拉即可。
GPIO0	上拉，可悬空	
GPIO2	上拉，可悬空	
nRESET	上拉，可悬空	因为 nRESET 管脚有内部弱上拉，所以也可以悬空。
CH_PD/CH_EN	必须上拉	这个管脚是芯片的使能管脚，这个管脚没有内部上拉，所以必须外部上拉。
SD3, SD2, SD0		这三个管脚组成的位组合不能等于 b010

与接线有关的问题，在大多数情况下，都是因为 CH_PD/CH_EN 管脚没有上拉。如果这个管脚没有上拉，芯片是不工作的。所以也就没有输出。而大多数模块，如果 GPIO15 管脚没有引出时，多半都已经在板上接了下拉。

详细说明请参看文档《ESP8266EX 启动的提示信息与上下拉跳线的解释和问题分析》。

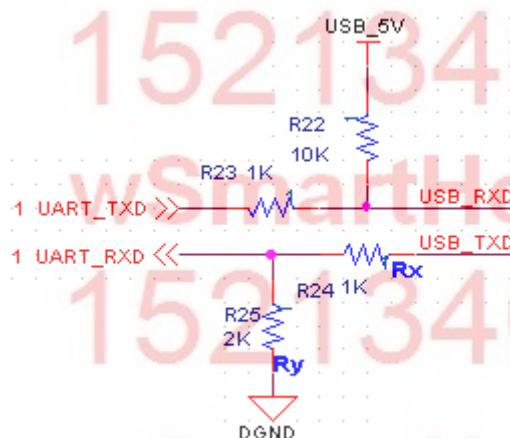
第二步 检查 TXD 和 RXD 没有接反, 且收发两端的电平是兼容的

发送和接收必须对接, 这一点不用赘述。只是当出现问题时, 请先确保没有接反。

ESP8266EX 芯片的 TXD 和 RXD 管脚是 LVTTTL 电平, 兼容 3.3V 逻辑。市面上有一些 USB 转串口 TTL 的模块, 则可能是 5V 逻辑, 或者自己的单片底板电路是 5V 逻辑的, 则可能出现问题。

而因为在大多数情况下, 一个 3.3V 逻辑的输出, 可能可以被一个 5V 逻辑的输入所识别; 但是一个 5V 逻辑的输出, 则往往因为电平超过了 3.3V 的而不能被 3.3V 的输入所识别。所以在许多情况下, 如果将 3.3V 逻辑的 ESP8266 模块, 接上了一个 USB 转串口 TTL 的模块, 也可能看到模块的输出, 但是模块对于输入却没有反应。这个时候, 不代表两者的电平是兼容的。

如果手上只有兼容 5V 的 USB 转串口, 一种简单地小技巧是, 可以通过分压电阻来解决。如下图所示:



图示中的 UART_TXD 和 UART_RXD 代表来自 ESP8266EX 芯片管脚的信号, USB_RXD 和 USB_TXD 代表来自 USB 转串口 5V 逻辑的模块的信号。在大多数情况下, 可以只连接 RX 和 RY, 而让 UART_TXD 和 USB_RXD 直连, 省略 R22 和 R23。

第三步 检查串口波特率是否设置正确

因为市面上的模块很多种, 模块缺省的波特率也不尽相同。如果你没有设置对波特率, 就不能通过串口和模块进行正常通信。所以, 大家遇到的问题, 往往就是波特率没有设置正确。

此时, 你可以借助前面提到辅助工具“ESP8266EX 调试与下载工具”, 来进行自动搜索进行波特率的匹配。

如果搜索成功, 恭喜你, 你遇到的问题很简单, 接着你就可以发送 AT 指令开始你的体验了。不必进行下面的步骤了。

如果按照这一步的方法搜索不到模块, 那么很不幸, 你遇到的问题, 可能比上面这个问题要复杂。这个时候, 出现问题的可能性包括但不限于:

- (1) 串口连线通信异常

(2) 供电问题, 供电电压需要是 3.3V 的, 不要超出太多也不要低出太多。

(3) 模块硬件问题

(4) 模块固件问题。

其中供电问题和模块的硬件问题, 需要大家自己用其他的方法去检查和排除, 这里谈谈如何根据用[软件工具的方法如何执导问题定位](#)的方法。

第四步 利用 ESP8266 调试和下载工具将波特率设置到 74880bps,

通过上电复位或管脚复位查看初始启动信息, 来判断芯片是否正常工作 and 模块的串口输出一路是否正常

在模块启动的前期, 会通过 ESP8266EX 会输出一段初始启动提示信息, 即类似与下面的
ets Jan 8 2013 rst cause:2 boot mode : (3,6)

在模块上电启动或通过复位管脚复位启动的前期, **只要 (1) 芯片焊接正常和供电没有问题、(2) 外接晶体正常工作 (3) 模块的串口输出一路连接正常, 初始启动提示信息会通过 ESP8266EX 芯片的串口 0 打印出来。而不需要任何其他前提, 也不需要外部的加载外设 (例如 SPI FLASH) 是否存在或者里面程序是否正常。**

详细解释, 可参看文档《ESP8266EX 启动的提示信息与上下拉跳线的解释和问题分析》。

具体操作如下:

1. 接好电源线和串口线, 上拉 CH_PD / CH_EN 管脚,

为了避免其他的连接对定位分析造成干扰, 可以只接三根线, 即电源线、地线、模块的 TXD 线, 同时确保 CH_PD 上拉, 其他的管脚可以都悬空。

看初始启动信息, 也不必关注 GPIO0、GPIO2、GPIO15 等选择启动模式的管脚, 直接悬空和上下拉都可以, 对输出初始启动信息没有影响。

2. 手工配置串口工具, 选择对串口的端口好, 将串口的波特率设置到 74880bps, 并打开串口准备开始接收串口数据

大多数模块使用晶体的频率为 26MHz, 在这个频率下, 初始启动信息输出时对应的串口波特率为 74880bps。

如果模块的晶体频率是 40MHz, 那么初始启动信息输出时对应的串口波特率为 115200bps。

3. 上电复位或通过复位管脚对模块进行复位

上电复位, 可以通过断电再通电的方式实现, 一个简单的方式是, 在 GND 和 TXD 都飞线

连接好的前提下, 将 VCC 两端的导线头轻轻接触, 就实现了上电又断电的过程, 其中上电操作就是一个上电重启的过程。

通过复位管脚复位, 可以通过在复位管脚上给出一个由低到高的电平来实现。一个简单的方式是, 用裸手拿着一个导电性能好的金属物, 比如镊子, 去轻轻接触 nRESET 管脚再拿开。因为人本身也是一个导体, 所以轻轻接触的再拿开的过程, 也可能造成模块芯片的复位。

4. 查看串口工具的接收数据

如果看到了初始启动信息, 那么恭喜你, 可以初步判断芯片正常工作了, 也不用去怀疑外接晶体的问题, 模块的串口输出这一路的连接是可用的。接下来要进入下一步继续检查。

而如果没有看到了初始启动信息, 或者看到的是乱码, 那么你要检查芯片供电是否正常、CH_PD 管脚是否上拉、外接晶体是否起振、模块的串口输出这一路的连接是否正常。

第五步 利用 ESP8266 调试和下载工具将模块设置到串口启动模式, 通过同步通信来检查模块的串口接收是否正常

要想成功地通过串口和模块进行通信, 除了按照前面的方法判断模块的串口发送一路是否正常之外, 还需要检查模块的串口接收一路也是正常。无论是烧写代码, 还是 AT 指令测试, 都需要确保串口双向通信都是正常的。

可以用 ESP8266 调试和下载工具的同步通信功能, 来检查模块串口的接收是否正常。这个同步通信功能, 只与芯片和串口通信有关, 与外部是否接上了 FLASH, 以及 FLASH 里是否有代码或者代码是否正确无关, 所以, 也可以用来排除其他一些实验干扰。

具体操作如下:

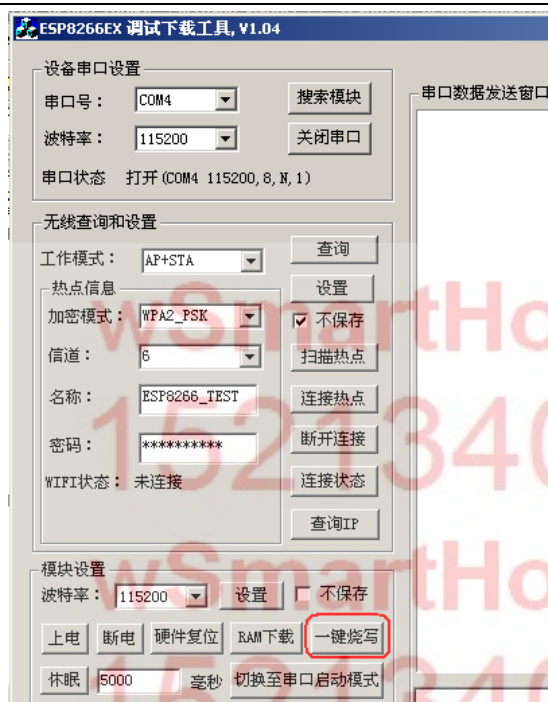
1. 接好电源线、串口线的 TXD 和 RXD, 上拉 CH_PD / CH_EN 管脚, 下拉 GPIO15 和 GPIO0, 悬空或上拉 GPIO2。

这个时候因为需要用到从串口 0 启动, 所以需要下拉 GPIO15 和 GPIO0。

2. 上电复位或通过复位管脚对模块进行复位

方法同第四步。复位的目的是为了让芯片进入串口启动的状态。

3. 点击 ESP8266 调试和下载工具主界面的“一键烧写”按钮, 进入串口通信模式

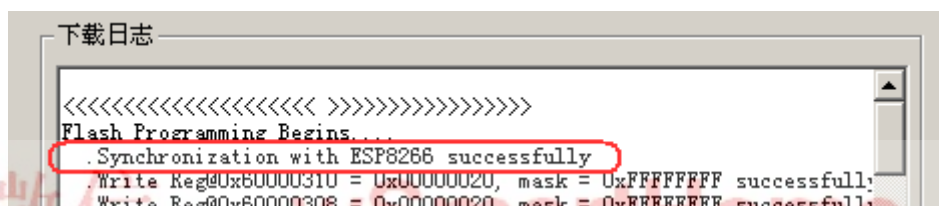


3. 弹出的如下对话框里，不要选择任何文件，如下所示，直接点击“下载”按钮



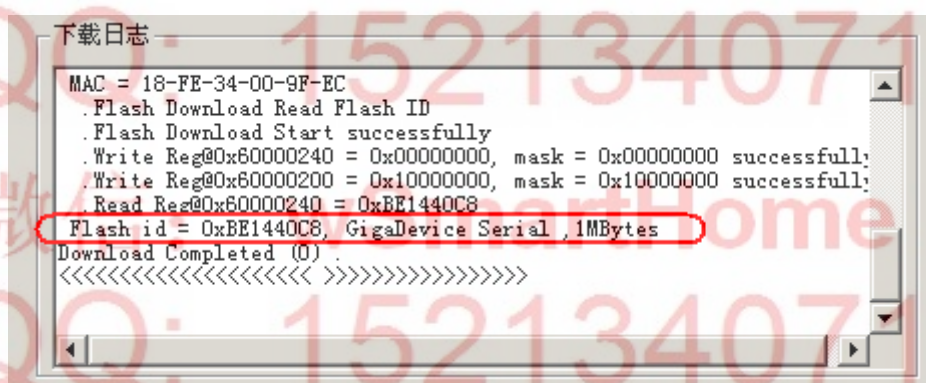
这个操作，将引导串口工具和模块之间进行双向同步通信。所以可以用来检查模块的串口接收是否正常。

串口日志的第一句话会提示同步是否成功，如下图所示。



如果在串口日志看到的是“**Synchronization with ESP8266 failed**”类似的提示，那么你的模块的串口接收一路存在问题(因为前面的步骤已经确认了模块的串口发送一路是没有问题)，**你需要重点检查串口接收一路的电路问题。**

在上一步的末尾，如果在日志的末尾你看到了如下类似的信息，则表明你的 **FLASH** 也别识别出来了。接着进行下一步 **FLASH** 代码的检查。



必须确保读出了 `flash_id`。如果没有读出来，则表明 FLASH 型号不被支持，**必须换成支持 QIO 模式的 SPI FLASH**，或者仔细核对 FLASH 的硬件连接是否存在问题。

如果通过前面的方法排除了前面的一些可能的原因，即芯片基本没有问题，串口双向通信也没有问题，**FLASH** 的硬件基本也没有问题。那么接下来检查，**FLASH** 内的数据和读写 **FLASH** 数据的通信是否正常。

1. 接好电源线、串口线的 TXD 和 RXD，上拉 CH_PD / CH_EN 管脚，下拉 GPIO15，悬空或上拉 GPIO0 和 GPIO2。

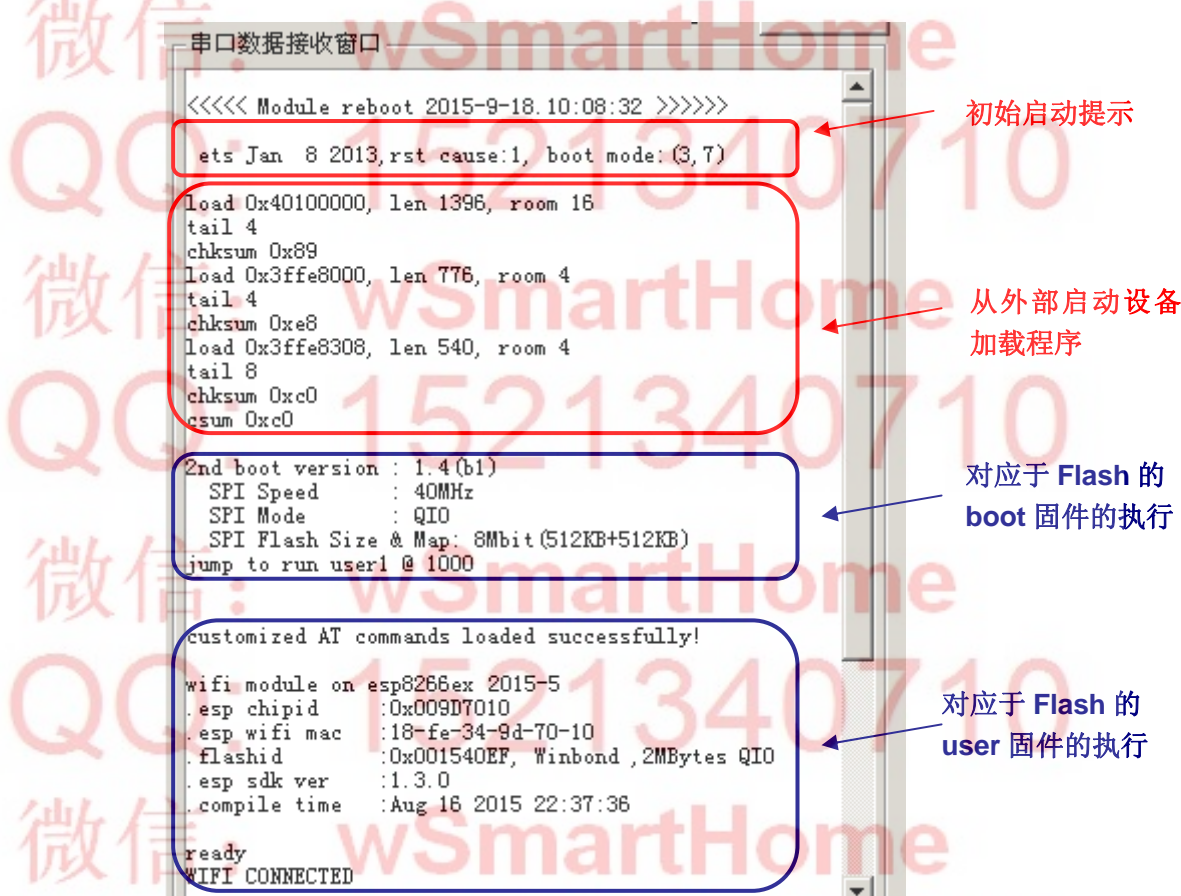
这个时候因为需要用到从外部 FLASH 启动, 所以需要下拉 GPIO15, 并上拉或悬空 GPIO0 和 GPIO2。

2. 上电复位或通过复位管脚对模块进行复位

方法同第四步。复位的目的是为了芯片进入外部 FLASH 启动的状态, 并查看相关的启动信息。

3. FLASH 启动信息分析

如下图所示, 一个完整的启动信息提示, 可能包括以下几个部分。其中, 红色的部分, 表示对对应与芯片内部固有代码的执行输出, 蓝色部分对应从 FLASH 加载进来的程序。



“初始启动提示”是芯片在最初确定从哪里加载程序时的一个提示, 所以, 不论外部是否存在启动设备, 或者启动设备里是否存在代码, 都会给出这个提示。这也是前面用于判断芯片是否工作正常串口发送是否正等等的一个依据。

“从外部启动设备加载程序”也是芯片内固有的一小段程序在继续执行。当芯片根据在启动时外部管脚的电平, 判断出从哪个外部设备开始启动之后, 就开始从这个启动设备开始加载程序 (将程序从外部设备复制到内部的 RAM 中)。

如果此时外部设备没有代码, 就不会给出图示中的加载提示; 如果程序的代码存在问题, 这

一段提示中的尾部有一个很关键的给出了程序里原来的校验和, 以及读取得到的代码计算出来的校验和。如果得不到这个加载提示, 或者校验和不对, 则表明固件存在问题, 或者 **FLASH** 的数据通信存在问题。你可以尝试重新烧写程序, 或检查 **FLASH** 访问部分的管脚焊接是否牢固, 附近是否存在虚短虚断等。

一般说来, 因为 **ESP8266** 模块的尺寸比较小 (所以走线都比较短), **ESP8266** 访问 **FLASH** 的时钟频率都不高才几十 **MHz**, 所以线路上只要联通没有短路断路, 一般不存在信号完整性等问题, 不用去担心是不是布线不好等问题。在和群友网友的探讨和支持过程中, 我们曾经遇到一个“布线非常糟糕”的设计, 在改正了串口收发的电平不兼容的问题之后, 一样可以稳定和正常地从 **FLASH** 启动。这也是 **ESP8266EX** 芯片本身的魅力所在。

“对应于 **Flash** 的 **boot** 固件的执行”和“对应于 **Flash** 的 **user** 固件的执行”分别对应于之前烧写进入的 **FLASH** 而后在启动的初始阶段被加载进入 **RAM** 来执行的 **BOOT** 代码和用户 **USER** 代码。根据提示, 可以判断这部分代码是否正确, 确定是否需要重新烧写代码。

第八步 如果经过了上述步骤之后, 依然还存在问题, 可以和技术 **QQ** 群联系, 获得直接支持。

技术 **QQ** 群: 148977035

ESP8266

ESP8266EX 新手问题系列, 2015-

电邮: 1521340710@qq.com QQ 群: 148977035

欢迎大家试用我们的一键烧写的 **USBWIFI ESP8266 模块**、**排针版 ESP8266 模块**、调试下载软件工具，特别方便于二次开发和程序烧写，有空去踩踩吧。
在下面的链接里有相关的产品说明书简介（详细说明书可以和我们联系索取）。

ESP8266EX 模块, USB 版

<http://item.taobao.com/item.htm?id=522160552246>



ESP8266EX 模块, 插针版

<http://item.taobao.com/item.htm?id=522158628730>

