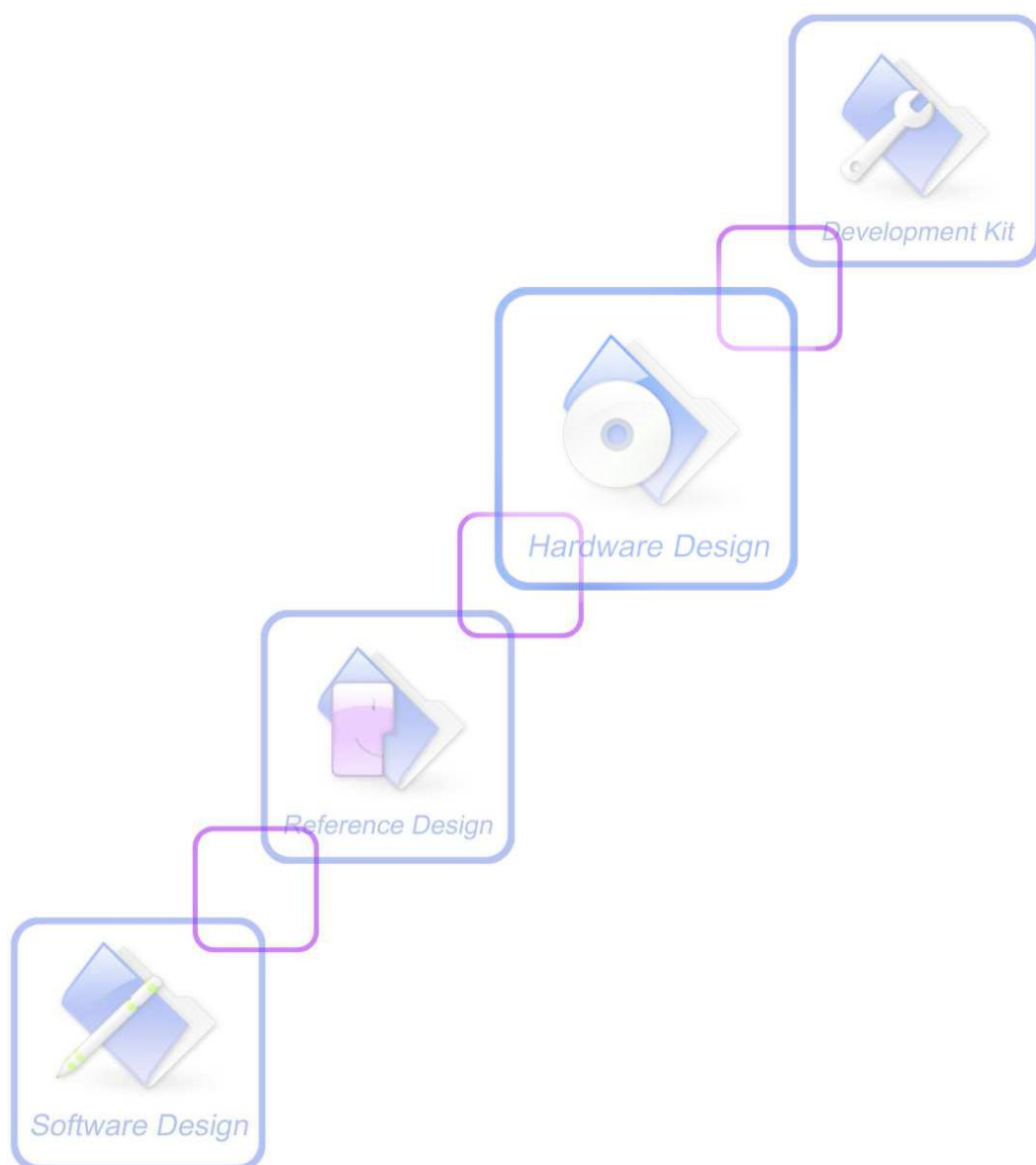


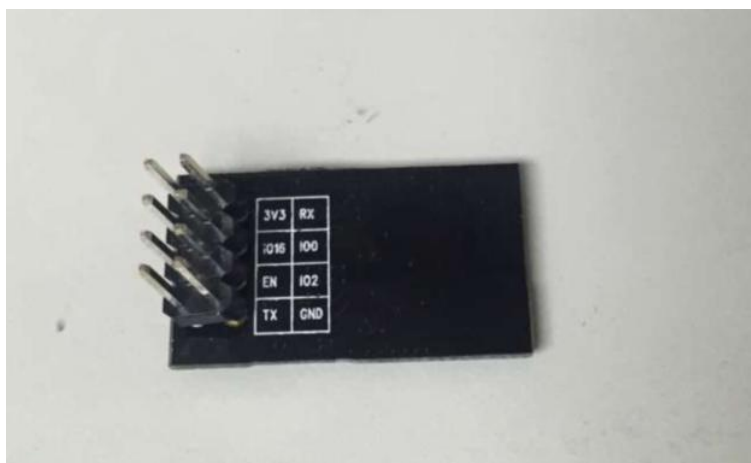


EPS8266-01 使用入门手册 V1.0.3

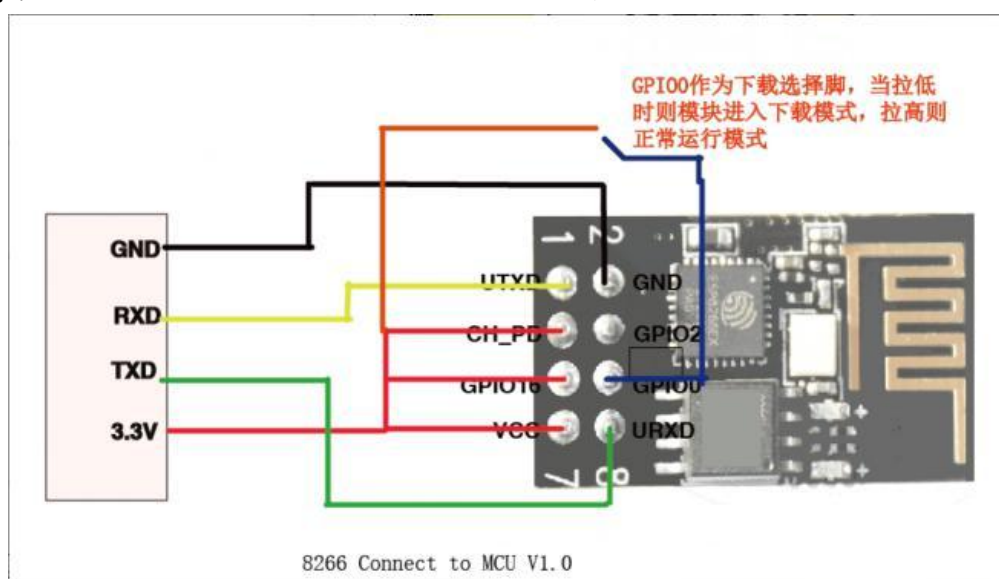


一、准备工作

1、本文档针对如下图所示板子的版本



2、确保模块本身工作正常，下一步按照正常运行模式（AT 模式）方式连线，模块使用的是 3.3v 的供电，接线图如下图所示：



上图可知我们将 3.3v--VCC GND--GND RXD--UTXD TXD--URXD。 （完成了这步我们会发现红灯亮了起来，如果红灯没亮或者光比较微弱，建议用万用表测试 VCC 和 GND 之间的电压是否达到 3.3V）

然后将 **CH_PD(EN)**, **GPIO16** (可做 **rest** 脚使用) 和 **GPIO0** 这三个脚拉高, 这个时候我们就进入了正常运行模式, 可以开始发 AT 测试了, 如果使用独立电源供电, 一定要记得将串口地与电源地共地。

二、正常工作验证

本模块可以工作在三种模式：1. STA 2. AP 3. AP+STA，出厂设置为第三种

上电后，蓝色灯微弱闪烁后熄灭，红灯长亮

1. 用手机搜索无线网络，可见 ESP_XXXXXX 已经处于列表中(后面的数字是 MAC 地址后几位)



连接该网络以后，查看连接状态：



手机搜索该网络，也可连接上：

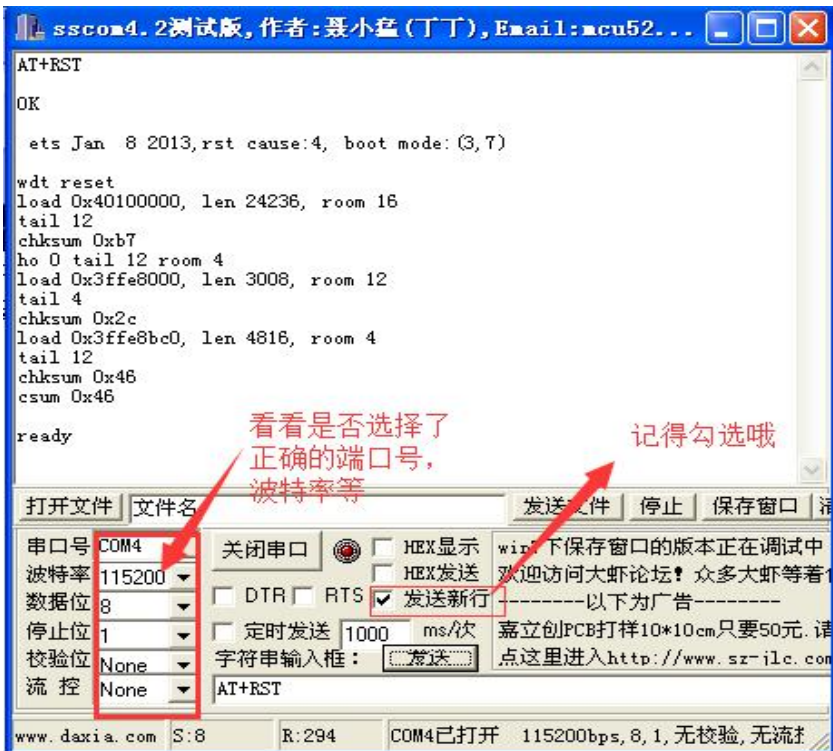


2. 使用 sscom42.exe 进行测试

测试软件在《.03_软件调试\调试工具\串口调试助手》

注意：勾选上“发送新行”

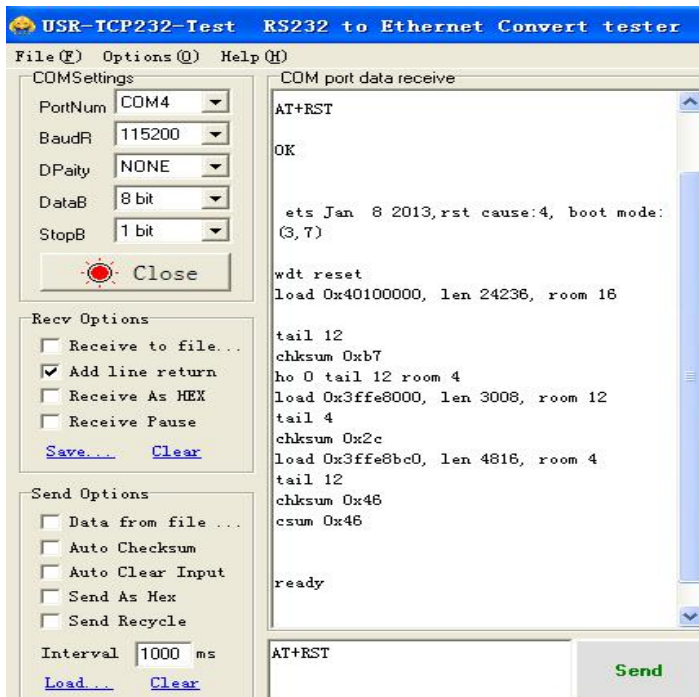
发送命令 **AT+RST**（重启模块），返回如下：



3. 使用 USR-TCP232-Test.exe 进行测试:

软件配置如下图:

注意: (1)波特率一般在出厂情况下默认的是 115200。如果在 115200 情况下收到的是乱码可以试试其他波特率
(2)在输入命令后必须再按一下回车键, 然后再按发送!



以上三种方法均可以判断，我们手上的模块是接线正确，运行正常的。然后我们就可以进入下一个环节，如果您需要烧录自己编写的固件或者其他的固件，请至《[烧写入门文档](#)》；

如果您需要自己编译固件，可以转到《[编译入门文档](#)》；

如果是要使用我们出厂就烧写好的标准固件，那么往下拉，下一章，我们进入调试环节。

三 测试例程

原来推荐的测试例程比较粗略，这里放上我自己的测试过程，理解可能有误，仅供参考。

请勿直接复制指令，部分格式经过 WORD 编辑后有误！可能导致出错！

（一）AP 模式

1. 建立 AP

(1)重启模块

发送命令：AT+RST(执行指令)

指令：AT+RST

响应：OK

(2)设置模块

发送命令：AT+CWMODE=3 或 AT+CWMODE=2(设置指令)

指令：AT+CWMODE=<mode>

说明：<mode>:1-Station模式，2-AP模式，3-AP兼Station模式

响应：OK

说明：需重启后生效(AT+RST)

```
AT+CWMODE=3
no change

AT+CWMODE=2

OK

AT+RST

OK
```

(3)配置 AP 参数

发送命令：AT+CWSAP="TEST","123456123456",1,3(设置指令)

指令：AT+CWSAP=<ssid>,<pwd>,<chl>,<ecn>

说明：指令只有在AP模式开启后有效

<ssid>:字符串参数，接入点名称

<pwd>:字符串参数，密码最长64字节，ASCII

<chl>:通道号

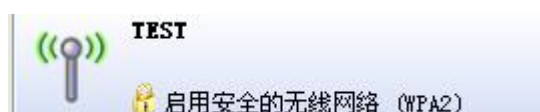
<ecn>:0-OPEN， 1-WEP， 2-WPA_PSK， 3-WPA2_PSK， 4-WPA_WPA2_PSK

响应：OK

```
AT+CWSAP="TEST","123456123456",1,3

OK
```

刷新无线网络列表，可见到 SSID 为 TEST 的无线网络列于其中：



注意：此时连接网络会可能出现连接不上的情况，请发送 AT+RST 命令并等待几分钟之后再连接

(4)查看已接入设备的 IP

连接上 TEST 后，发送命令：AT+CWLIF(执行指令)

指令：AT+CWLIF

说明：查看已接入设备的 IP

响应：<ip addr>

OK

说明：<ip addr>:已接入设备的 IP 地址

```
AT+CWLIF
192.168.4.100
OK
```

如果返回命令如下：

```
AT+CWLIF
OK
```

表示网络成功建立，目前无设备连入。

如果无线网络实际已连接上，请等待几分钟后再次发送 AT+CWLIF 命令进行查询。

(4)查询本机 IP 地址

发送命令：AT+CIFSR(执行指令)

指令：AT+CIFSR

说明：查看本模块的 IP 地址

注意：AP 模式下无效！会造成死机现象！

响应：<ip addr>

说明：<ip addr>:本模块 IP 地址

```
AT+CIFSR
192.168.4.1
```

但实际测试中并未发生死机现象。

查看本机配置模式：CWMODE=2，为 AP 模式。

```
AT+CWMODE?
+CWMODE:2
OK
```

发送命令：AT+CIFSR=? (测试指令)，返回响应如下：

```
AT+CIFSR=?
OK
```

同样未出现死机现象。

2. Server 方法收发

(0)查询此时模块状态(该步骤可省略)

发送命令 AT+CWMODE?(查询指令)

指令：AT+CWMODE?

说明：查看本模块的 WIFI 应用模式

响应：+CWMODE:<mode>

OK

说明：<mode>:1-Station 模式，2-AP 模式，3-AP 兼 Station 模式

```
AT+CWMODE?  
+CWMODE:1  
  
OK
```

发送命令 AT+CIPMUX? (查询指令)

指令: AT+CIPMUX?

说明: 查询本模块是否建立多连接

响应: + CIPMUX:<mode>

OK

说明: <mode>:0-单路连接模式, 1-多路连接模式

```
AT+CIPMUX?  
+CIPMUX:0  
  
OK
```

发送命令 AT+CIPMODE? (查询指令)

指令: AT+CIPMODE?

说明: 查询本模块的传输模式

响应: + CIPMODE:<mode>

OK

说明: <mode>:0-非透传模式, 1-透传模式

```
AT+CIPMODE?  
+CIPMODE:0  
  
OK
```

发送命令 AT+CIPSTO? (查询指令)

指令: AT+CIPSTO?

说明: 查询本模块的服务器超时时间

响应: + CIPSTO:<time>

OK

说明: <time>:服务器超时时间, 0~2880, 单位为 s

```
AT+CIPSTO?  
+CIPSTO:180  
  
OK
```

(1)开启多连接模式

发送命令: AT+CIPMUX=1(设置指令)

指令: AT+CIPMUX=<mode>

说明: <mode>:0-单路连接模式, 1-多路连接模式

响应: OK

```
AT+CIPMUX=1  
  
OK
```

查询可知, 设置成功


```
AT+CIPMUX?
+CIPMUX:1
OK
```

(2)创建服务器

发送命令: **AT+CIPSERVER=1,8080**(设置指令)

指令: **AT+CIPSERVER=<mode>[,<port>]**

说明: <mode>:0-关闭 server 模式, 1-开启 server 模式

<port>:端口号, 缺省值为 333

响应: OK

说明: (1) **AT+ CIPMUX=1** 时才能开启服务器; 关闭 server 模式需要重启

(2)开启 server 后自动建立 server 监听,当有 client 接入会自动按顺序占用一个连接。

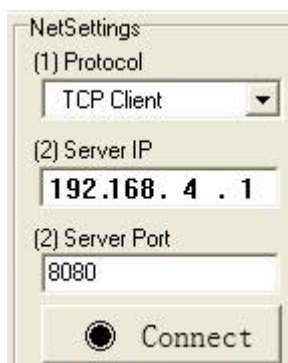
开启 server 服务如下图所示:

```
AT+CIPSERVER=1,8080
OK
```

关闭 server 服务如下图所示:

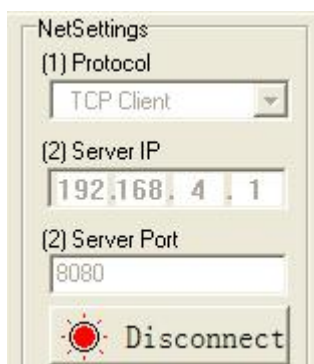
```
AT+CIPSERVER=0
we must restart
AT+RST
OK
```

打开 USR-TCP232-Test.exe, 点击 Connect 按钮连接不上, 可知 server 服务未开启



重新开启 server 服务(注意: 之前需要再发送一遍 **AT+CIPMUX=1** 以重新开启多连接模式)。

点击 Connect 按钮



连接成功后, 串口收到模块返回的数据串: Link

180S(默认值)后, 连接自动断开, 返回 Unlink,客户端的相应按钮变成灰色。

全过程如下图:

```
AT+CIPMUX=1
OK
AT+CIPSERVER=1,8080
OK
Link
Unlink
```

(3) 设置服务器超时时间

发送命令 **AT+CIPSTO=2880** (设置指令)

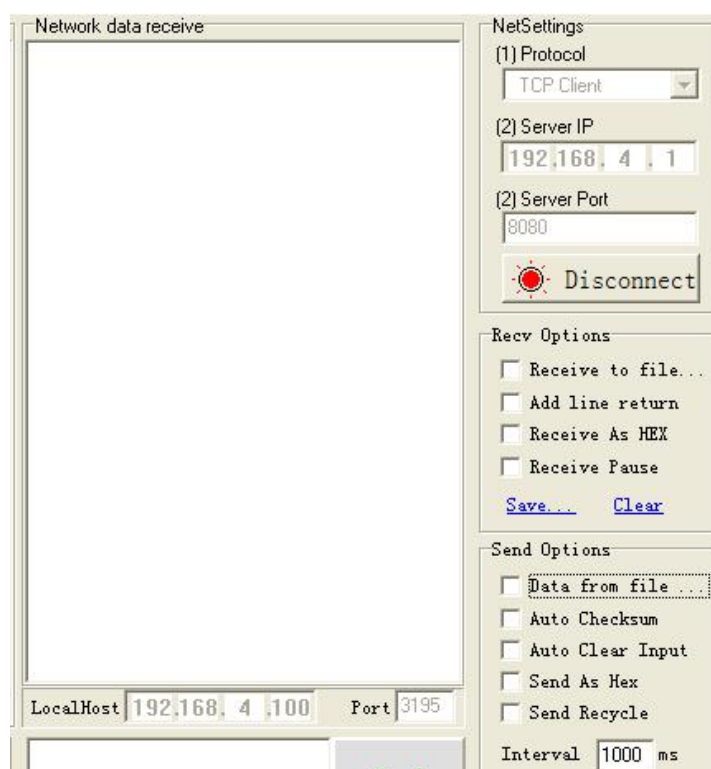
指令: **AT+CIPSTO=<time>**

说明: <time>:服务器超时时间, 0~2880, 单位为 s

响应: OK

```
AT+CIPSTO=2880
OK
```

(4) 建立客户端



(5) 查看当前连接

发送命令 **AT+CIPSTATUS** (执行指令)

指令: **AT+CIPSTATUS**

响应: **STATUS:<stat>**

+ CIPSTATUS:<id>,<type>,<addr>,<port>,<tetype>

OK

说明: <id>:连接的 id 号 0-4

<type>:字符串参数, 类型 TCP 或 UDP

<addr>:字符串参数, IP 地址

<port>:端口号

<tetype>: 0-本模块做 client 的连接, 1-本模块做 server 的连接

```
AT+CIPSTATUS
STATUS:3
+CIPSTATUS:0,"TCP","192.168.4.100",3195,1
+CIPSTATUS:1,"TCP","192.168.4.100",3792,1
OK
```

(之前电脑的 USB 无线网卡发烫厉害, 拔出后重新插上, 再次连接后自动分配的端口产生了变化, 所以刷出了两个客户端, 实际中存在的仅有 ID=1 的客户端)

(6) 向某个连接发送数据

发送命令 AT+CIPSEND=1,6(设置指令) (通过上一条指令 AT+CIPSTATUS 得知 ID=1)

指令: 1)单路连接时(+CIPMUX=0), 指令为: AT+CIPSEND=<length>

2)多路连接时(+CIPMUX=1), 指令为: AT+CIPSEND= <id>,<length>

响应: 收到此命令后先换行返回">", 然后开始接收串口数据

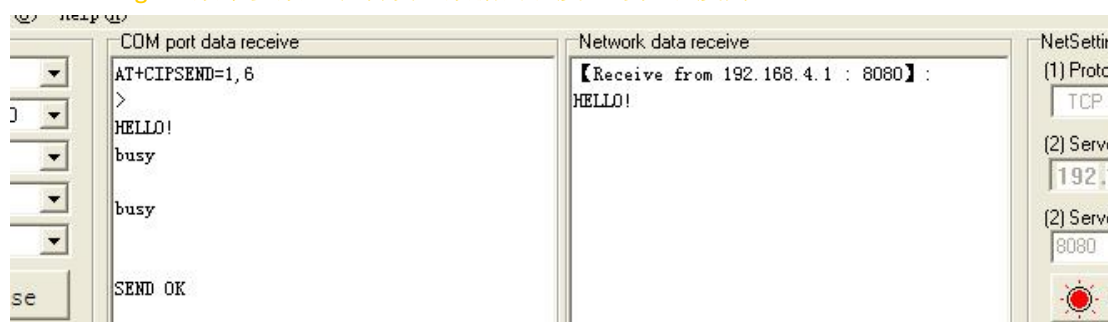
当数据长度满 length 时发送数据。

如果未建立连接或连接被断开, 返回 ERROR

如果数据发送成功, 返回 SEND OK

说明: <id>:需要用于传输连接的 id 号

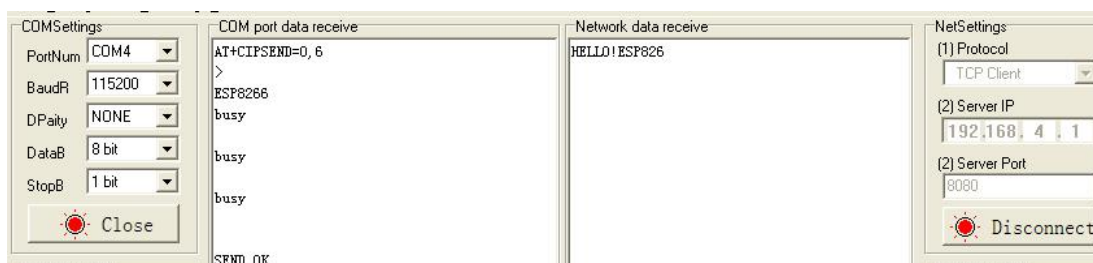
<length>:数字参数, 表明发送数据的长度, 最大长度为 2048



断开客户端后再发送一次, 得到以下响应:

```
HELLO!
Error
```

发送数据长度大于 LENGTH 时响应如下(HELLO!为之前一次发送的数据):



此时连接已建

立, 可以进行数据的双向收发。

用客户端向 ESP8266 发送数据, 正常。

```
+IPD,0,5:happy
OK

+IPD,0,4:测试
OK

+IPD,0,22:多次测试数据发送和接收
OK
```

3. Client 方法收发

(0) 关闭 server 服务 (如果没有开启 server 服务, 可免除此步骤)

发送命令: **AT+CIPSERVER=0**(设置指令)

指令: **AT+CIPSERVER=<mode>[,<port>]**

说明: <mode>:0-关闭 server 模式, 1-开启 server 模式

<port>:端口号, 缺省值为 333

响应: OK

说明: (1) AT+ CIPMUX=1 时才能开启服务器; 关闭 server 模式需要重启

(2) 开启 server 后自动建立 server 监听, 当有 client 接入会自动按顺序占用一个连接。

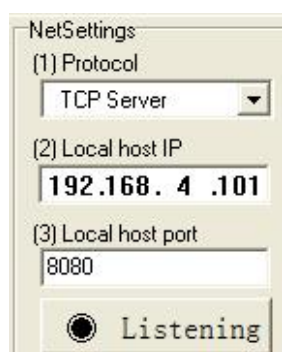
关闭 server 服务如下图所示:

```
AT+CIPSERVER=0
we must restart

AT+RST

OK
```

(1) 创建服务器



点击 Listening, 创建成功后, 该按钮变为:



(2) 开启多连接模式

发送命令：AT+CIPMUX=1(设置指令)

指令：AT+CIPMUX=<mode>

说明：<mode>:0-单路连接模式，1-多路连接模式

响应：OK

```
AT+CIPMUX=1
OK
```

(3) 建立 TCP 连接

发送命令 AT+CIPSTART=2,"TCP","192.168.4.101",8080 (设置指令)

指令：1)单路连接时(+CIPMUX=0)，指令为：AT+CIPSTART= <type>,<addr>,<port>

2)多路连接时(+CIPMUX=1)，指令为：AT+CIPSTART=<id>,<type>,<addr>,<port>

响应：如果格式正确且连接成功，返回 OK，否则返回 ERROR

如果连接已经存在，返回 ALREADY CONNECT

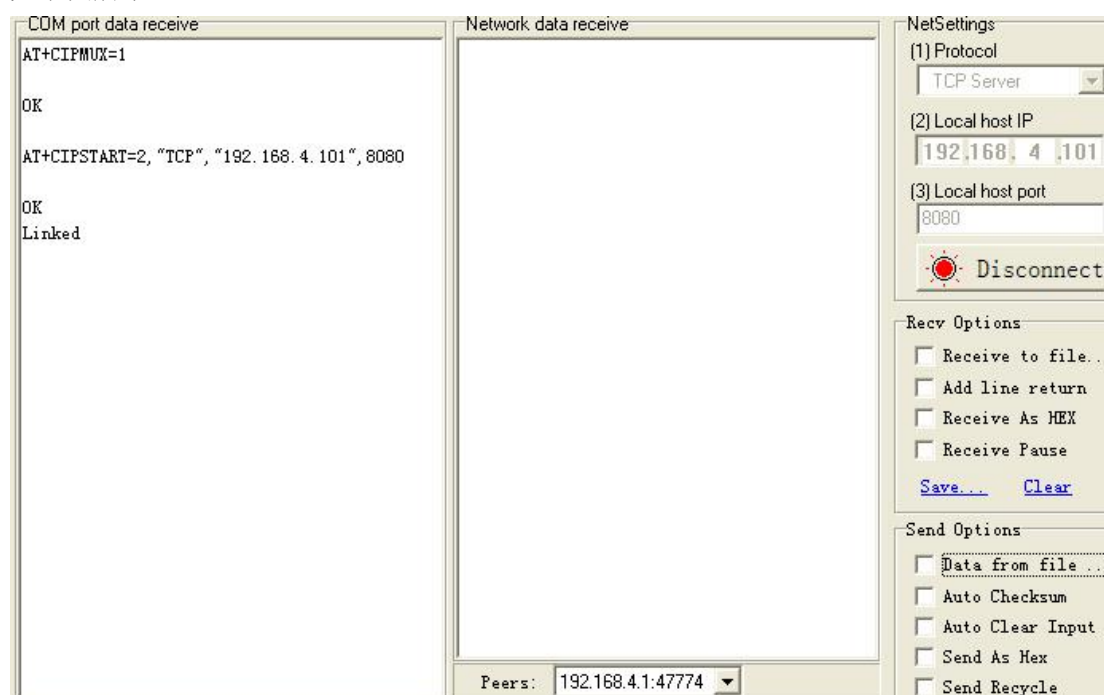
说明：<id>:0-4，连接的 id 号

<type>:字符串参数，表明连接类型，“TCP”-建立 tcp 连接，“UDP”-建立 UDP 连接

<addr>:字符串参数，远程服务器 IP 地址

<port>:远程服务器端口号

如下图所示：



(4) 向服务器发送数据

发送命令 AT+CIPSEND=2,8(设置指令) （通过上一条指令 AT+CIPSTART 设置为 ID=2）

指令：1)单路连接时(+CIPMUX=0)，指令为：AT+CIPSEND=<length>

2)多路连接时(+CIPMUX=1)，指令为：AT+CIPSEND=<id>,<length>

响应：收到此命令后先换行返回">", 然后开始接收串口数据

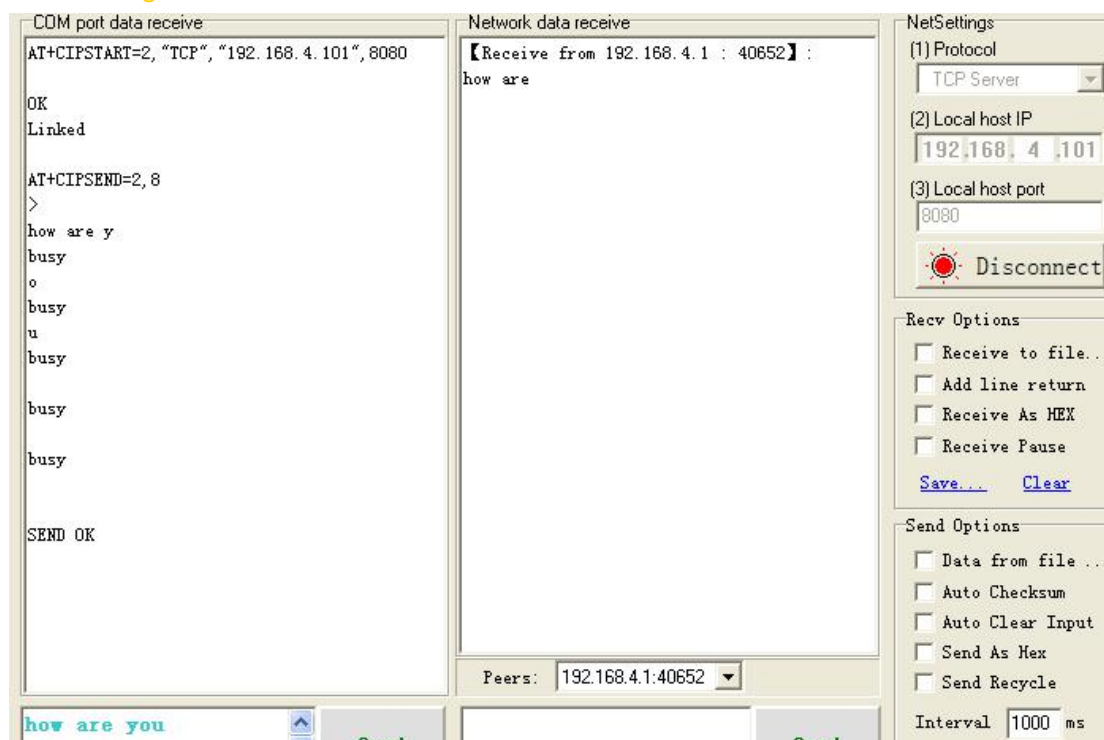
当数据长度满 length 时发送数据。

如果未建立连接或连接被断开，返回 ERROR

如果数据发送成功，返回 SEND OK

说明：<id>:需要用于传输连接的 id 号

<length>:数字参数，表明发送数据的长度，最大长度为 2048



接收正确，超过 8 个字符的部分不接收。

此时连接已建立，可以进行数据的双向收发。用服务器向 ESP8266 发送数据，正常。

```
+IPD,2,9:try again
OK

+IPD,2,10:happy hour
OK
```

(二) STA 模式

1. 建立 STA

(1) 重启模块

发送命令：AT+RST(执行指令)

指令：AT+RST

响应: OK

```
AT+RST
OK

ets Jan  8 2013,rst cause:4, boot mode:
(3,7)

wdt reset

load 0x40100000, len 24236, room 16
tail 12
chksum 0xb7
ho 0 tail 12 room 4
load 0x3ffe8000, len 3008, room 12
tail 4
chksum 0x2c
load 0x3ffe8bc0, len 4816, room 4
tail 12
chksum 0x46
csum 0x46

ready
```

(2) 设置模块

发送命令: **AT+CWMODE=3 或 AT+CWMODE=1(设置指令)**

指令: **AT+CWMODE=<mode>**

说明: <mode>:1-Station模式, 2-AP模式, 3-AP兼Station模式

响应: OK

说明: 需重启后生效(AT+RST)

```
AT+CWMODE=1
OK

AT+RST
OK
```

将模块设置为模式 1,此时刷新网络列表可见由 ESP8266 建立的网络从列表中消失了

(3) 查看当前无线路由器列表

发送命令: **AT+CWLAP(执行指令)**

指令: **AT+CWLAP**

响应: 正确: (终端返回AP列表)

+ CWLAP: <ecn>,<ssid>,<rssi>

OK

错误: ERROR

说明: <ecn>:0-OPEN, 1-WEP, 2-WPA_PSK, 3-WPA2_PSK, 4-WPA_WPA2_PSK

<ssid>:字符串参数, 接入点名称

<rssi>:信号强度

```
AT+CWLAP
+CWLAP: (0, "", 0)
+CWLAP: (0, "CMCC-FREEGAME", -82)
+CWLAP: (4, "shiningwuxi", -91)
+CWLAP: (4, "TP-LINK_shz", -72)
+CWLAP: (0, "CMCC", -85)
+CWLAP: (1, "TP-LINK_lq", -79)

+CWLAP: (2, "ChinaNet-emrG", -53)
+CWLAP: (2, "iTV-emrG", -55)
+CWLAP: (4, "908", -89)
+CWLAP: (4, "AFD", -65)
+CWLAP: (4, "MERSAIN", -55)
+CWLAP: (4, "FAST_DACD2C", -94)
+CWLAP: (0, "CMCC-AUTO", -85)
+CWLAP: (2, "Tenda_330170", -83)
+CWLAP: (0, "CMCC-FREEGAME", -85)
+CWLAP: (2, "ChinaNet-P9Gt", -76)
+CWLAP: (4, "LINKSYS", -91)
+CWLAP: (4, "maxhome", -89)
+CWLAP: (4, "TP-LINK_FC566A", -75)
+CWLAP: (4, "tfdy", -88)
+CWLAP: (4, "liu", -91)

OK
```

如搜索不到信号，响应：

```
AT+CWLAP
ERROR
```

请重新上电并严格按照 AT 命令再发送一遍。

没有列表返回的时候请耐心等待，否则会出现“busy”

(4) 加入当前无线网络

发送命令： AT+CWLAP="MERSAIN","XXXXXXXX"(设置指令)

指令： AT+CWLAP=<ssid>,<pwd>

说明： <ssid>:字符串参数，接入点名称

<pwd>:字符串参数，密码，最长64字节ASCII

响应： 正确： OK

错误： ERROR

```
AT+CWLAP="MERSAIN", "██████████"

OK
```

检测是否真的连上该路线网络

发送命令： AT+CWLAP?(查询指令)

指令： AT+CWLAP?

响应： 返回当前选择的AP

+ CWJAP:<ssid>

OK

说明: <ssid>:字符串参数, 接入点名称

```
AT+CWJAP?
+CWJAP:"MERSAIN"
OK
```

断电。上电后发送 **AT+CWJAP?**, 返回同上, 系统保持上次状态。

查看模块 IP 地址

发送命令: **AT+CIFSR**(执行指令)

指令: **AT+CIFSR**

响应: 正确: **+CIFSR:<IP address>**

OK

错误: **ERROR**

说明: <ssid>:字符串参数, 接入点名称

```
AT+CIFSR
192.168.1.102
```

群里 (@云海之梦) 反映: 错误的 SSID 也可以连接上, 测试后发现果然如此, 输入实际不存在的账号密码返回也是 OK

```
ready
AT+CWJAP="JOKER", "111111"
OK
AT+CWJAP?
+CWJAP:"JOKER"
OK
AT+CIFSR
ERROR
```

但发送 **AT+CIFSR**(查看模块地址), 返回 **ERROR**, 可通过此命令来判断有没有真正连接上。

2. Server 方法收发

(0) 查询此时模块状态(该步骤可省略)

发送命令 **AT+CWMODE?**(查询指令)

指令: **AT+CWMODE?**

说明: 查看本模块的 **WIFI** 应用模式

响应: **+CWMODE:<mode>**

OK

说明: <mode>:1-Station 模式, 2-AP 模式, 3-AP 兼 Station 模式

```
AT+CWMODE?
+CWMODE:2
OK
```

发送命令 **AT+CIPMUX?** (查询指令)

指令: **AT+CIPMUX?**

说明: 查询本模块是否建立多连接

响应: **+CIPMUX:<mode>**

OK

说明: <mode>:0-单路连接模式, 1-多路连接模式

```
AT+CIPMUX?
+CIPMUX:0
OK
```

发送命令 AT+CIPMODE? (查询指令)

指令: AT+CIPMODE?

说明: 查询本模块的传输模式

响应: + CIPMODE:<mode>

OK

说明: <mode>:0-非透传模式, 1-透传模式

```
AT+CIPMODE?
+CIPMODE:0
OK
```

发送命令 AT+CIPSTO? (查询指令)

指令: AT+CIPSTO?

说明: 查询本模块的服务器超时时间

响应: + CIPSTO:<time>

OK

说明: <time>:服务器超时时间, 0~2880, 单位为 s

```
AT+CIPSTO?
+CIPSTO:180
OK
```

(1)开启多连接模式

发送命令: AT+CIPMUX=1(设置指令)

指令: AT+CIPMUX=<mode>

说明: <mode>:0-单路连接模式, 1-多路连接模式

响应: OK

```
AT+CIPMUX=1
OK
```

查询可知, 设置成功

```
AT+CIPMUX?
+CIPMUX:1
OK
```

(2)创建服务器

发送命令: AT+CIPSERVER=1,8080(设置指令)

指令: AT+CIPSERVER=<mode>[,<port>]

说明: <mode>:0-关闭 server 模式, 1-开启 server 模式

<port>:端口号, 缺省值为 333

响应: OK

说明: (1) AT+ CIPMUX=1 时才能开启服务器; 关闭 server 模式需要重启

(2)开启 server 后自动建立 server 监听,当有 client 接入会自动按顺序占用一个连接。

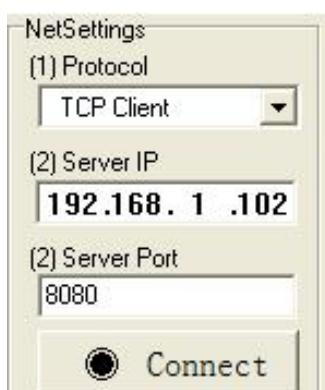
开启 server 服务如下图所示:

```
AT+CIPSERVER=1,8080
OK
```

关闭 server 服务如下图所示:

```
AT+CIPSERVER=0
we must restart
AT+RST
OK
```

打开 USR-TCP232-Test.exe, 点击 Connect 按钮连接不上, 可知 server 服务未开启



重新开启 server 服务(注意: 之前需要再发送一遍 AT+CIPMUX=1 以重新开启多连接模式)。

点击 Connect 按钮



连接成功后, 串口收到模块返回的数据串: Link

180S(默认值)后, 连接自动断开, 返回 Unlink,客户端的相应按钮变成灰色。

全过程如下图:

```

AT+CIPMUX=1

OK

AT+CIPSERVER=1,8080

OK

Link

Unlink

```

(3) 设置服务器超时时间

发送命令 **AT+CIPSTO=2880** (设置指令)

指令: **AT+CIPSTO=<time>**

说明: <time>:服务器超时时间, 0~2880, 单位为 s

响应: OK

```

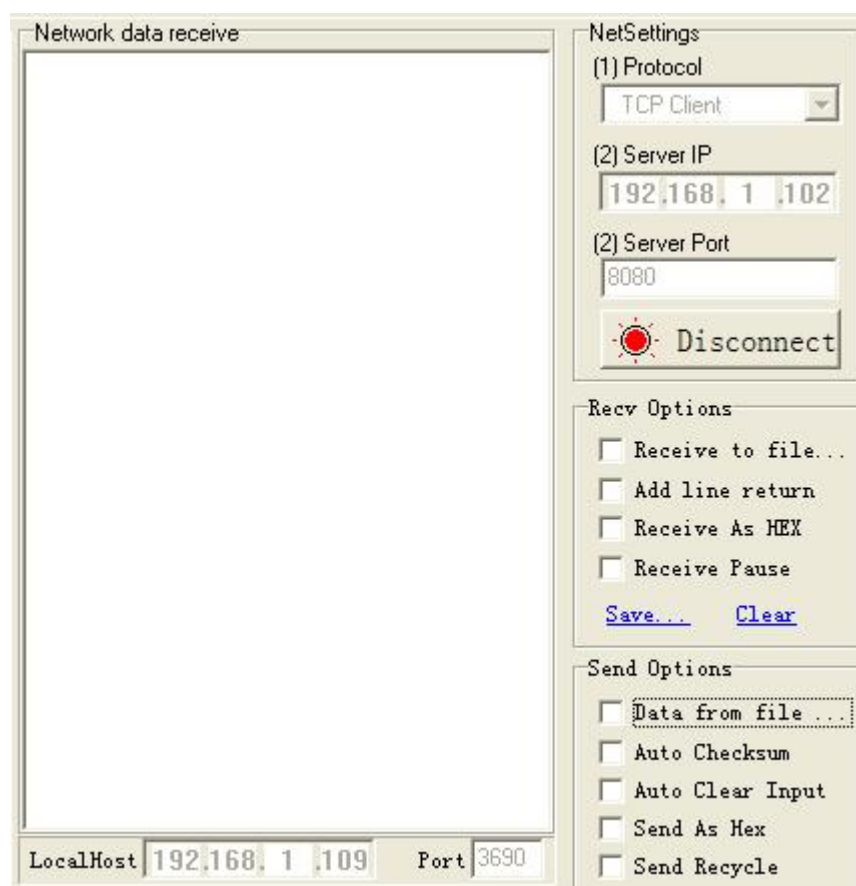
AT+CIPSTO=2880

OK

```

(4) 建立客户端

界面设置如下图所示:



(5) 查看当前连接

发送命令 **AT+CIPSTATUS** (执行指令)

指令: **AT+CIPSTATUS**

响应: **STATUS:<stat>**

+ CIPSTATUS:<id>,<type>,<addr>,<port>,<tetype>

OK

说明: <id>:连接的 id 号 0-4

<type>:字符串参数, 类型 TCP 或 UDP

<addr>:字符串参数, IP 地址

<port>:端口号

<tetype>: 0-本模块做 client 的连接, 1-本模块做 server 的连接

```
AT+CIPSTATUS
STATUS:3
+CIPSTATUS:0,"TCP","192.168.1.109",3690,
1
OK
```

(6) 向某个连接发送数据

发送命令 **AT+CIPSEND=0,10(设置指令)** (通过上一条指令 AT+CIPSTATUS 得知 ID=0)

指令: 1)单路连接时(+CIPMUX=0), 指令为: AT+CIPSEND=<length>

2)多路连接时(+CIPMUX=1) , 指令为: AT+CIPSEND= <id>,<length>

响应: 收到此命令后先换行返回">", 然后开始接收串口数据

当数据长度满 length 时发送数据。

如果未建立连接或连接被断开, 返回 ERROR

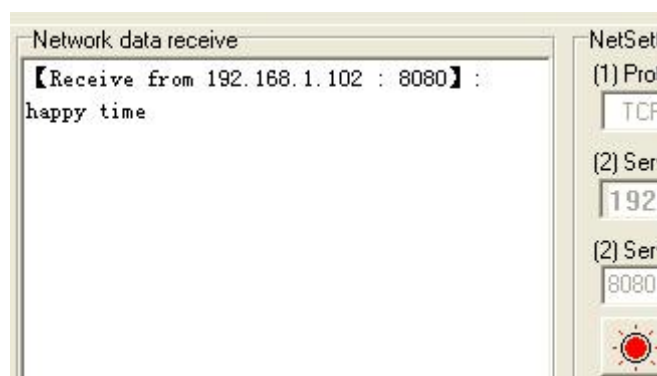
如果数据发送成功, 返回 SEND OK

说明: <id>:需要用于传输连接的 id 号

<length>:数字参数, 表明发送数据的长度, 最大长度为 2048

```
AT+CIPSEND=0,10
>
happy time
busy
busy
happy time
```

发:



收:

断开客户端后再发送一次, 得到以下响应:

```
HELLO!
Error
```

此时连接已建立，可以进行数据的双向收发。

3. Client 方法收发

(0) 关闭 server 服务 (如果没有开启 server 服务，可免除此步骤)

发送命令：AT+CIPSERVER=0(设置指令)

指令：AT+CIPSERVER=<mode>[,<port>]

说明：<mode>:0-关闭 server 模式，1-开启 server 模式

<port>:端口号，缺省值为 333

响应：OK

说明：(1) AT+ CIPMUX=1 时才能开启服务器；关闭 server 模式需要重启

(2)开启 server 后自动建立 server 监听,当有 client 接入会自动按顺序占用一个连接。

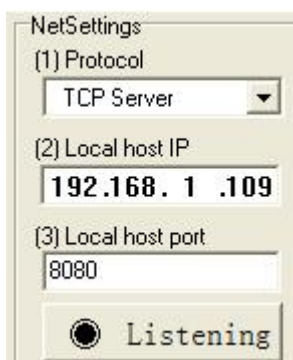
关闭 server 服务如下图所示：

```
AT+CIPSERVER=0
we must restart

AT+RST

OK
```

(1)创建服务器



点击 Listening，创建成功后，该按钮变为：



(2) 开启多连接模式

发送命令：AT+CIPMUX=1(设置指令)

指令：AT+CIPMUX=<mode>

说明：<mode>:0-单路连接模式，1-多路连接模式

响应：OK

```
AT+CIPMUX=1

OK
```

(3) 建立 TCP 连接

发送命令 AT+CIPSTART=2,"TCP","192.168.1.109",8080 (设置指令)

指令：1)单路连接时(+CIPMUX=0)，指令为：AT+CIPSTART=<type>,<addr>,<port>
2)多路连接时(+CIPMUX=1)，指令为：AT+CIPSTART=<id>,<type>,<addr>,<port>

响应：如果格式正确且连接成功，返回 OK，否则返回 ERROR

如果连接已经存在，返回 ALREADY CONNECT

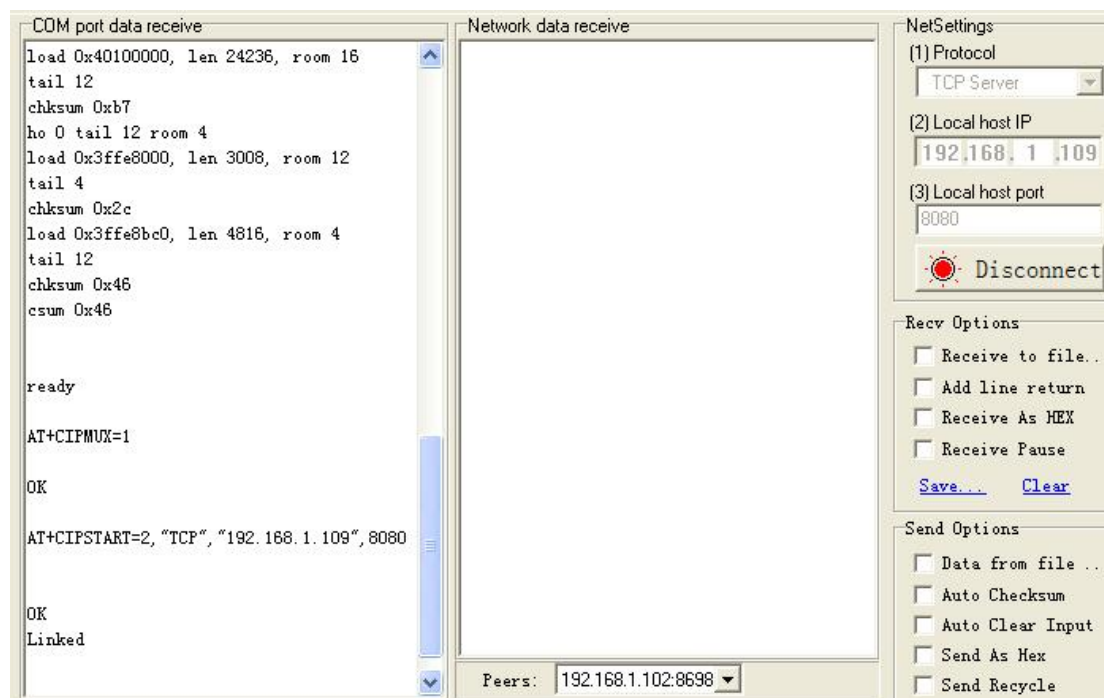
说明：<id>:0-4，连接的 id 号

<type>:字符串参数，表明连接类型，“TCP”-建立 tcp 连接，“UDP”-建立 UDP 连接

<addr>:字符串参数，远程服务器 IP 地址

<port>:远程服务器端口号

如下图所示：



(4) 向服务器发送数据

发送命令 AT+CIPSEND=2,10(设置指令) （通过上一条指令 AT+CIPSTART 设置为 ID=2）

指令：1)单路连接时(+CIPMUX=0)，指令为：AT+CIPSEND=<length>
2)多路连接时(+CIPMUX=1)，指令为：AT+CIPSEND=<id>,<length>

响应：收到此命令后先换行返回">", 然后开始接收串口数据

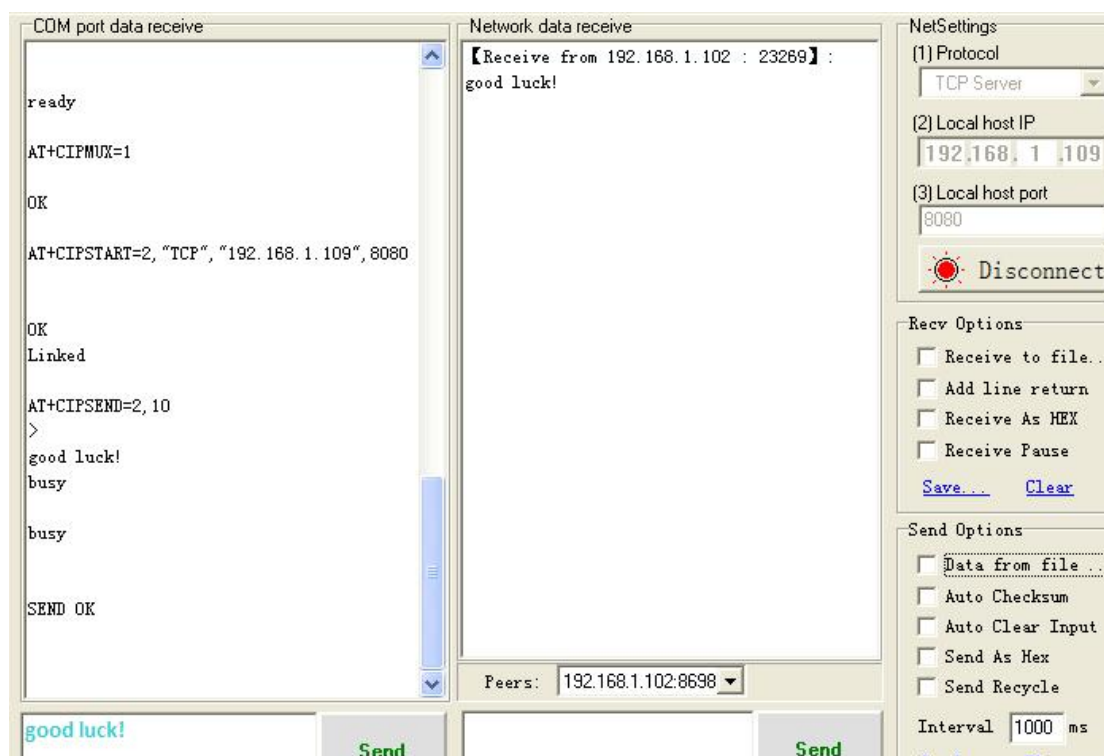
当数据长度满 length 时发送数据。

如果未建立连接或连接被断开，返回 ERROR

如果数据发送成功，返回 SEND OK

说明：<id>:需要用于传输连接的 id 号

<length>:数字参数，表明发送数据的长度，最大长度为 2048



此时连接已建立，可以进行数据的双向收发。