



华为 GTM900 无线模块 TCPIP AT 命令
V100R001

使用指导书

文档版本 03

发布日期 2007-05-22

部件编码

华为技术有限公司为客户提供全方位的技术支持，用户可与就近的华为办事处联系，也可直接与公司总部联系。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编： 518129

网址： <http://www.huawei.com>

客户服务电话： 0755-28560000 8008302118

客户服务传真： 0755-28560111

客户服务邮箱： Support@huawei.com

版权所有 © 华为技术有限公司 2007。 保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目 录

1 命令列表	1-1
2 打开一条TCP链接	2-1
2.1 配置apn.....	2-1
2.2 进入tcpip功能.....	2-1
2.3 打开一条tcp链接.....	2-1
2.4 发送数据到tcp对端(%IOMODE=1).....	2-1
2.5 发送数据到tcp对端(%IOMODE=0).....	2-2
2.6 新数据直接提交到串口，默认缓存没有打开(%IOMODE=1,1,1).....	2-2
2.7 缓存打开，从接收缓存中数据读取(%IOMODE=1,,0).....	2-2
2.8 关闭链接(单链接).....	2-2
3 打开一条UDP链接	3-1
3.1 配置apn.....	3-1
3.2 进入tcpip功能.....	3-1
3.3 打开一条udp链接.....	3-1
3.4 发送数据到udp对端.....	3-1
3.5 新数据直接提交到串口，默认缓存没有打开.....	3-2
3.6 缓存打开，接收缓存中数据读取(%IOMODE=1,,0).....	3-2
3.7 关闭链接(单链接).....	3-2
4 多链接功能的使用	4-1
4.1 配置apn.....	4-1
4.2 进入tcpip功能.....	4-1
4.3 设置多链接模式.....	4-1
4.4 分别打开一条tcp和udp链接.....	4-2
4.5 发送数据到链接 1 或 2 (%IOMODE=1).....	4-2
4.6 新数据直接提交到串口，默认缓存没有打开.....	4-2
4.7 缓存功能打开，但要区分link_num来判断数据来源.....	4-2
4.8 关闭链接.....	4-3
5 TCP服务器AT命令的使用	5-1
5.1 配置APN.....	5-1

5.2 进入tcpip功能.....	5-1
5.3 开启服务器功能.....	5-1
5.4 查询服务器状态(可选).....	5-1
5.5 成功接入客户端.....	5-2
5.6 查询链路状态（可选）.....	5-2
5.7 关闭服务器链路.....	5-2
5.8 查询服务器状态(可选).....	5-2
5.9 查询链路状态(可选).....	5-2
6 UDP服务器AT命令的使用.....	6-1
6.1 配置APN.....	6-1
6.2 进入tcpip功能.....	6-1
6.3 开启服务器功能.....	6-1
6.4 查询服务器状态(可选).....	6-1
6.5 收到客户端数据后自动建立连接.....	6-2
6.6 查询链路状态（可选）.....	6-2
7 延时发送AT命令的使用.....	7-1
7.1 设置延时.....	7-1
7.2 发送数据.....	7-1
7.3 查询待发内存状态.....	7-1
7.4 数据发送.....	7-1
8 其它AT命令的使用.....	8-1
8.1 TCP ACK查询.....	8-1
8.2 接收缓存中数据查询.....	8-1
8.3 删除接收缓存中的数据.....	8-1
8.4 限定UDP数据对方发送端口.....	8-2
8.5 退出tcpip功能.....	8-2
8.6 域名解析.....	8-2
9 附录.....	9-1
9.1 注意事项.....	9-1
9.2 各种命令返回时间.....	9-1
9.3 转换函数.....	9-2

1 命令列表

1	AT%ETCPIP	PDP 激活, 初始化 TCPIP
2	AT%IPOPEN	打开一条 TCP/UDP 链接
3	AT%IPOPENX	打开多条 TCP/UDP 链接
4	AT%IOMODE	设置数据模式, 单链接或多链接模式选择, 缓存设置
5	AT%IPSEND	单链接模式下发送数据
6	AT%IPSENDX	多链接模式下发送数据
7	AT%IPCLOSE	关闭链接
8	AT%TXSTATE	TCP 的 ACK 查询
9	AT%IPDQ	数据查询
10	AT%IPDR	数据读取
11	AT%IPDD	数据删除
12	AT%IPDDMODE	数据删除模式
13	AT%DNSR	域名解析
14	AT%IPLISTEN	服务器功能
15	AT%IPSP	延时发送功能

2 打开一条 TCP 链接

2.1 配置 apn

```
AT+CGDCONT=1,"IP","CMNET"  
OK
```

2.2 进入 tcpip 功能

```
AT%ETCPIP  
OK
```

如果需要输入用户名和密码，假如用户名：user，密码：gprs

```
AT%ETCPIP="user","gprs"  
OK
```

2.3 打开一条 tcp 链接

```
AT%IPOPEN="TCP","202.206.1.26",23,,4098  
CONNECT
```



说明

4098 为绑定本地端口

2.4 发送数据到 tcp 对端(%IOMODE=1)

假如发送数据 0x01 0x02 0x03 0x04 0xEF

在发送中需要把 0x01 转换成 0x30 0x31，也就是可显字符 ‘0’，‘1’，依次类推，可以参考附录转换函数

```
AT%IPSEND="01020304EF"
```

```
%IPSEND: 15  
OK
```

2.5 发送数据到 tcp 对端(%IOMODE=0)

假如发送的数据为可显，且不包含非法字符‘;’和’”，比如需要发送字符串“0123456789ABCD%^&EFGH”到 TCP 对端：

```
AT%IPSEND="0123456789ABCD%^&EFGH"  
%IPSEND: 15  
OK
```

2.6 新数据直接提交到串口，默认缓存没有打开(%IOMODE=1,1,1)

```
%IPDATA:<len>,<data>  
eg:  
%IPDATA: 6, "78125578AABB"
```

表明：实际数据长度为 6

数据为：0x78 0x12 0x55 0x78 0xAA 0xBB

2.7 缓存打开，从接收缓存中数据读取(%IOMODE=1,,0)

```
AT%IPDR  
%IPDR: 1, 2, 4, "3038EF55"
```

- 1: 连接号
- 2: 包序号
- 4: 数据包长度

0x30,0x38,0xEF,0x55: 数据内容

2.8 关闭链接(单链接)

```
AT%IPCLOSE  
OK
```

这个 OK 最长需要 15 秒

3 打开一条 UDP 链接

3.1 配置 apn

```
AT+CGDCONT=1,"IP","CMNET"  
OK
```

3.2 进入 tcpip 功能

```
AT%ETCPIP  
OK
```

如果需要输入用户名和密码，假如用户名：user，密码：gprs

```
AT%ETCPIP="user","gprs"  
OK
```

3.3 打开一条 udp 链接

```
AT%IPOPEN="UDP","61.171.1.26",23,,4098  
CONNECT
```



说明

4098 为绑定本地端口

3.4 发送数据到 udp 对端

假如发送数据 0x01 0x02 0x03 0x04 0xEF

在发送中需要把 0x01 转换成 0x30 0x31，也就是可显字符 ‘0’，’1’，依次类推，可以参考附录转换函数

```
AT%IPSEND="01020304EF"
```

OK

3.5 新数据直接提交到串口，默认缓存没有打开

```
%IPDATA:<len>,<data>
```

eg:

```
%IPDATA:6,"78125578AABB"
```

表明：实际数据长度为 6

数据为：0x78 0x12 0x55 0x78 0xAA 0xBB

3.6 缓存打开，接收缓存中数据读取(%IOMODE=1,,0)

```
AT%IPDR
```

```
%IPDR: 1, 2, 4, "3038EF55"
```

1: 连接号

2: 包序号

4: 数据包长度

0x30,0x38,0xEF,0x55: 数据内容

3.7 关闭链接(单链接)

```
AT%IPCLOSE
```

OK

这个 OK 很快返回，因为是 UDP

4 多链接功能的使用

4.1 配置 apn

```
AT+CGDCONT=1,"IP","CMNET"
```

```
OK
```

4.2 进入 tcpip 功能

```
AT%ETCPIP
```

```
OK
```

如果需要输入用户名和密码，假如用户名：user，密码：gprs

```
AT%ETCPIP="user","gprs"
```

```
OK
```

4.3 设置多链接模式

```
AT%IOMODE=,2
```

```
OK
```



注意

当已经有链接存在的情况下，也就是使用了 AT%IOPEN 或者 AT%IOPENX，就无法对 AT%IOMODE 第二个参数进行设置。

4.4 分别打开一条 tcp 和 udp 链接

```
AT%IPOPENX=1,"TCP","202.206.1.26",23
```

```
CONNECT
```

```
AT%IPOPENX=2,"UDP","61.175.8.90",8888
```

```
CONNECT
```

4.5 发送数据到链接 1 或 2 (%IOMODE=1)

```
AT%IPSENDX=1,"032078291F"
```

```
%IPSEND: 15
```

```
OK
```

```
AT%IPSENDX=2,"AA0F3899"
```

```
OK
```

4.6 新数据直接提交到串口，默认缓存没有打开

```
%IPDATA:<link_num>,<len>,<data>
```

```
eg:
```

```
%IPDATA:1,6,"78125578AABB"
```

表明： 数据来源于链接 1

实际数据长度为 6

数据为： 0x78 0x12 0x55 0x78 0xAA 0xBB

4.7 缓存功能打开，但要区分 link_num 来判断数据来源

```
AT%IPDR
```

```
%IPDR: 1, 2, 4, "3038EF55"
```

1: 连接号

2: 包序号

4: 数据包长度

0x30,0x38,0xEF,0x55: 数据内容

AT%IPDR

%IPDR: 2, 10, 4, "3038EF55"

2: 连接号

10: 包序号

4: 数据包长度

0x30,0x38,0xEF,0x55: 数据内容

4.8 关闭链接

AT%IPCLOSE=1

OK

AT%IPCLOSE=2

OK

5 TCP 服务器 AT 命令的使用

5.1 配置 APN

```
AT+CGDCONT=1,"IP","CMNET"
```

```
OK
```

5.2 进入 tcpip 功能

```
AT%ETCPIP
```

```
OK
```

如果需要输入用户名和密码，假如用户名：user，密码：gprs

```
AT%ETCPIP="user","gprs"
```

```
OK
```

5.3 开启服务器功能

```
AT%IPLISTEN="TCP",5000
```

```
OK
```



说明

开启服务器,类型 TCP,端口 5000

5.4 查询服务器状态(可选)

```
AT%IPLISTEN?
```

```
%IPLISTEN: "TCP",5000,1
```



说明

1 表示目前还可连接 1 个客户端(当前为单链接模式)

5.5 成功接入客户端

```
CONNECT: "10.10.100.115",4097,1
```



说明

该步骤由客户端发起，上面为服务器端显示的非请求结果码

5.6 查询链路状态（可选）

```
AT%IPOPEN?
```

```
%IPOPEN: "TCP", "10.10.100.115", 4097, 5000
```



说明

"10.10.100.115"为客户端 IP，4097 为客户端端口，5000 为本地端口。

5.7 关闭服务器链路

```
AT%IPCLOSE=6
```

```
%IPCLOSE: 6
```

```
OK
```



说明

6 代表服务器正常关闭

5.8 查询服务器状态(可选)

```
AT%IPLISTEN?
```

```
%IPLISTEN: 0
```



说明

0 表示服务器已关闭

5.9 查询链路状态(可选)

```
AT%IPOPEN?
```

```
%IPOPEN: "TCP", "10.10.100.115", 4097, 5000
```


6 UDP 服务器 AT 命令的使用

6.1 配置 APN

```
AT+CGDCONT=1,"IP","CMNET"
```

```
OK
```

6.2 进入 tcpip 功能

```
AT%ETCPIP
```

```
OK
```

如果需要输入用户名和密码，假如用户名：user，密码：gprs

```
AT%ETCPIP="user","gprs"
```

```
OK
```

6.3 开启服务器功能

```
AT%IPLISTEN="UDP",5000
```

```
OK
```



说明

开启服务器,类型 UDP,端口 5000

6.4 查询服务器状态(可选)

```
AT%IPLISTEN?
```

```
%IPLISTEN: "UDP",5000,1
```

6.5 收到客户端数据后自动建立连接

```
%IPDATA:1,10,"0123456789"
```



说明

该步骤由客户端发起。当客户端向服务器端的监听端口成功发送一包数据后，服务器会自动建立连接。以上为服务器端收到数据后的非请求结果码。

6.6 查询链路状态（可选）

```
AT%IPOPEN?
```

```
%IPOPEN:"UDP","10.10.100.120",4097,5000,4097
```



说明

第一个 4097 为客户端的接收端口；5000 为本地端口，第二个 4097 表示客户端的发送端口。

7 延时发送 AT 命令的使用

7.1 设置延时

```
AT%IPSP=5000 //设置 5 秒延时
OK
AT%IPSP?
%IPSP:5000,1400 //5000 为延时时间, 1400 为目前剩余内存
```

7.2 发送数据

```
AT%IPSEND="01234" //发送第一包数据, 最大长度 512 字节
OK
AT%IPSEND="56789" //发送第二包数据
OK
```

7.3 查询待发内存状态

```
AT%IPSP?
%IPSP:5000,1390
```



说明

5000 为延时时间, 1390 为目前剩余内存(最大量为 1400,之前已放入 10 个字节)

7.4 数据发送

```
%IPSD: 10
%IPSEND: 15
```

OK

 **说明**

延时结束发送数据，10 表示发送 10 个数据,该步骤由设备自动执行，以上为非请求结果码。此时对端只收到 1 个数据包，其内容为"0123456789"

8 其它 AT 命令的使用

8.1 TCP ACK 查询

AT%TXSTATE

% TXSTATE:

1,6,6

2,0,0

3,150,149

OK

表示链接 1，发了 6 包，并且对方收到了 6 包

链接 2 没有建立或者是 UDP

链接 3 发送了 150 包，对方确认了 149 包

8.2 接收缓存中数据查询

AT%IPDQ

%IPDQ: 4, 5

OK

4: 表示接收缓存中有 4 包未读数据包

5: 表示接收缓存中共有 5 包数据（接收缓存总共能储存 10K 字节数据）

8.3 删除接收缓存中的数据包

AT%IPDD

%IPDD:1

OK

1:删除数据包的个数

如果不进行 AT%IPDDMODE=1 的设置，使用 AT%IPDR 读取后，会自动删除。缺省情况下，自动删除。

8.4 限定 UDP 数据对方发送端口

```
AT%IPOPENX=2,"UDP","61.175.8.90",1000,2000
```

```
CONNECT
```

这个时候，模块只会把远端服务器 61.175.8.90 2000 端口的数据提交给用户，其他端口数据直接丢弃。

8.5 退出 tcpip 功能

```
AT%IPCLOSE=5
```

```
OK
```

模块从 GPRS 网络注销，仅进行 PDP 去激活，没有进行 GPRS DETATCH，也就是说 GPRS 仍然处于附着状态，如果您想再次打开链接，请首先进行 AT%ETCPIP 操作。

8.6 域名解析

```
AT%DNSR="www.huawei.com"
```

```
%DNSR: 219.133.0.61
```

```
OK
```

9 附录

9.1 注意事项

1. 模块上电 20-30 秒后再进行 TCPIP 相关命令的使用，因为模块需要时间搜索网络，以及相关信息初始化等。
2. 使用 AT%ETCPIP 命令，当命令还没有返回的时候，串口有任何输入都会导致该命令强制结束，有%IPCLOSE: 5 的返回，表明模块退出 TCPIP 功能，这个时候必须重新使用 AT%ETCPIP 进行 TCPIP 功能的打开操作。
3. 使用%IPSEND 发送数据，在%IOMODE=1 的时候一包请不要超过 1024 字节，如果要发送大于 1024 字节的数据包，可以采用延时发送。%IOMODE=0 的时候一包请不要超过 512 字节。推荐使用 MODE 1，因为 MODE 0 输入的必须为可见字符，并且不能为引号 (") 和分号 (;)。
4. UDP 方式下，服务器下发数据一包请不要超过 512 字节，否则多余的部分会丢失。如果要发送大于 1024 字节的数据包，可以采用延时发送。
5. 打电话的时候请不要使用 AT%IPSEND 或者 AT%IPSENDX 进行数据发送。
6. 打电话的时候请不要使用 AT%IPCLOSE=5 进行 GPRS 网络注销。
7. UDP 远端发送端口的校验。在打开 UDP 连接时，如果设置了远端发送端口，则模块对于该链路只接收该端口的发送过来的数据；如果没有设置远端发送端口，模块将接收到的第一包 UDP 数据中的端口设为该链路的远端发送端口，以后对该端口进行校验。

9.2 各种命令返回时间

1. 使用 AT%ETCPIP 返回时间与网络状况有关，3-60 秒不等，应用最长等待时间可设置为 30 秒。为了保证通信质量，要确保 AT+CSQ 返回值在 15 以上，如果长时间没有结果返回，可以通过输入回车符结束这条命令，然后重新使用。
2. 使用 AT%IPOPEN 和 AT%IPOPENX 最长需要等待 20-30 秒钟左右。
3. 使用 AT%IPCLOSE 关闭一条 TCP 链接最长需要等待 15 秒钟左右有 OK 返回。
4. 使用 AT%DNSR 最长需要等待 13 秒左右，有结果返回。
5. 其他命令一般都能在 500ms 左右有返回，应用超时等待可以设置成 3 秒。

9.3 转换函数

当您设置 AT%IOMODE=1 后，数据收发需要进行转换，可参考下面 2 个函数：

```
#define MAX_USER_DATA 1024
typedef unsigned char UBYTE
/*****
ASCII 到 HEX 的转换函数
入口参数： O_data: 转换数据的入口指针，
           N_data: 转换后新数据的入口指针
           len : 需要转换的长度
返回参数： -1: 转换失败
           其它: 转换后数据长度
注意： O_data[] 数组中的数据在转换过程中会被修改。
*****/
int ascii_2_hex(UBYTE *O_data, UBYTE *N_data, int len)
{
    int i,j,tmp_len;
    UBYTE tmpData;
    UBYTE *O_buf = O_data;
    UBYTE *N_buf = N_data;

    for(i = 0; i < len; i++)
    {
        if ((O_buf[i] >= '0') && (O_buf[i] <= '9'))
        {
            tmpData = O_buf[i] - '0';
        }
        else if ((O_buf[i] >= 'A') && (O_buf[i] <= 'F')) //A...F
        {
            tmpData = O_buf[i] - 0x37;
        }
        else if((O_buf[i] >= 'a') && (O_buf[i] <= 'f')) //a...f
        {
            tmpData = O_buf[i] - 0x57;
        }
        else

```



```
{
    return -1;
}

O_buf[i] = tmpData;
}

for(tmp_len = 0, j = 0; j < i; j+=2)
{
    N_buf[tmp_len++] = (O_buf[j]<<4) | O_buf[j+1];
}

return tmp_len;
}
```

/******

HEX 到 ASCII 的转换函数

入口参数: data: 转换数据的入口指针
 buffer: 转换后数据入口指针
 len : 需要转换的长度

返回参数: 转换后数据长度

```
*****/
int hex_2_ascii(UBYTE *data, CHAR *buffer, int len)
{
    const CHAR ascTable[17] = {"0123456789ABCDEF"};
    CHAR *tmp_p = buffer;

    int i, pos;
    pos = 0;

    for(i = 0; i < len; i++)
    {
        tmp_p[pos++] = ascTable[data[i] >> 4];
        tmp_p[pos++] = ascTable[data[i] & 0x0f];
    }
}
```

```
    }  
  
    tmp_p[pos] = '\\0';  
  
    return pos;  
}
```