

## 文档历史

### 修订记录

版本	日期	作者	变更描述
V1.0	2021-10	王正	初始版本
V2.0	2022-12-15	王和洋	增加短消息相关命令及矫正其它错误描述
V3.0	2023-07-08	王和洋	增加呼叫控制命令以及 USIM 相关命令
V3.1	2023-07-22	王和洋	增加电话簿相关命令、修改例程格式
V3.2	2023-09-07	王和洋	增加通用命令
	2023-10-24	王和洋	增加+CGREG、+CREG，修改例程格式

**备注：**该指令集会不定期更新，如若您手中的模组不支持部分指令，请尝试将模组升级到最新的通用版本。

## 目录

文档历史	1
目录	2
1. 文档介绍	7
1.1. 文档范围	7
1.2. 惯例和术语缩写	7
1.3. 命令语法	7
1.4. 注意事项	8
2. 模组信息识别命令	9
2.1. AT+CPSV 读取 GPRS 协议栈软件版本信息	9
2.2. AT+CGMI 读取厂商名	9
2.3. AT+CGMM 读取设备名	9
2.4. AT+CGMR 读取设备版本信息	10
2.5. AT+CGSN 读取 IMEI	10
3. 通用命令	10
3.1. AT+UCRESET 模块重启命令	10
3.2. AT+UCMODE 模式切换命令	10
3.3. AT+UCGNSSSTATE GNSS 定位状态查询	11
3.4. AT+UCGNSSDATA GNSS 定位数据	11
4. (U)SIM 相关命令	12
4.1. AT+CIMI 读取 IMSI	12
4.2. AT+CPIN 输入 PIN 码	12
4.3. AT+CPUN 查询 PIN 和 PUK 重试次数	13
4.4. AT+CCID 查询 CCID	13
4.5. AT+CNUM 查询用户号码	13
4.6. AT+CRSM 有限的 SIM 卡访问	14
5. 网络服务相关命令	15
5.1. AT+CELLLOCK 锁频	15

5.2. AT+CGCLASS 能力模式设置并附着网络 .....	16
5.3. AT+CGREG PS 域网络注册状态 .....	16
5.4. AT+CREG CS 域网络注册状态 .....	18
5.5. AT+CSQ 检测信号强度 .....	19
5.6. AT+COPS 运营商选择 .....	19
<b>6. GPRS 相关命令 .....</b>	<b>22</b>
6.1. AT+CFUN 设定模块功能 .....	22
6.2. AT+CGATT GPRS 注册注销 .....	22
6.3. AT+CGACT PDP 激活去激活 .....	23
6.4. AT+CGQREQ 设置 QOS .....	24
6.5. AT+CGQMIN 设置 PDP 最小 QOS .....	26
6.6. AT+CGPADDR 显示 PDP 地址 .....	28
6.7. AT+CGDCONT 定义 PDP 上下文 .....	29
6.8. AT+CGDATA 进入数据态 .....	31
示例-完整 PDP 激活流程 .....	32
示例-完整数据传输流程 .....	32
<b>7. 呼叫控制命令 .....</b>	<b>33</b>
7.1. ATD 呼叫指令 .....	33
7.2. ATA 来电接听 .....	33
7.3. ATH 挂断通话 .....	34
示例-主被叫通话 AT 应用 .....	34
7.4. AT+CLIP 设置来电显示 .....	35
7.5. AT+CLCC 查询当前呼叫列表 .....	35
7.6. AT+CSVC 扬声器音量控制 .....	37
7.7. AT+VTS DTMF 音调发送 .....	38
示例-完整正常执行拨打接听电话 .....	38
<b>8. 电话簿相关命令 .....</b>	<b>38</b>
8.1. AT+CPBS 选择电话簿存储单元 .....	38
8.2. AT+CPBW 写删电话簿记录 .....	39

8.3. AT+CPBR 读取电话簿记录 .....	40
8.4. AT+CPBF 查找电话簿记录 .....	41
<b>9. TCP/IP 相关命令 .....</b>	<b>41</b>
9.1. AT+IPCREATE 创建 Socket .....	41
9.2. AT+IPSEND Socket 发送数据 .....	42
9.3. AT+IPSENDDATA Socket 数据透传 .....	43
9.4. AT+IPCLOSE 关闭 Socket .....	43
9.5. TCP 使用示例 .....	44
9.6. UDP 使用示例 .....	45
<b>10. FTP 相关命令 .....</b>	<b>45</b>
10.1. AT+FTPMODE 设置 FTP 模式 .....	45
10.2. AT+FTPYPE 设置 FTP 数据传输类型 .....	46
10.3. AT+FTPRESET FTP 断点续传 .....	46
10.4. AT+FTPSERV 设置 FTP 服务地址 .....	47
10.5. AT+FTPUN 设置 FTP 用户名 .....	47
10.6. AT+FTPPW 设置 FTP 密码 .....	48
10.7. AT+FTPGETNAME 设置 FTP 客户端 get 文件名字 .....	49
10.8. AT+FTPPATH FTP 切换路径 .....	49
10.9. AT+FTPPUTNAME 设置 FTP 客户端 put 的文件名字 .....	50
10.10. AT+FTPPUT FTP PUT 文件 .....	51
10.11. AT+FTPMKD FTP 创建目录 .....	51
10.12. AT+FTPMD FTP 删除目录 .....	51
10.13. AT+FTPGET FTP GET 文件 .....	52
10.14. AT+FTPDELE FTP 删除文件 .....	52
10.15. AT+FTPLIST FTP 文件目录列表 .....	53
10.16. AT+FTPSCON 查询 FTP 配置 .....	53
10.17. AT+FTPQUIT 退出 FTP .....	54
10.18. FTP 实例 .....	54
<b>11. HTTP 相关命令 .....</b>	<b>56</b>

---

11.1. AT+HTTPTLS 设置 HTTPS 的 TLS .....	56
11.2. AT+HTTPPARA 设 HTTP 参数 .....	57
11.3. AT+HTTPDATA HTTP 数据 .....	58
11.4. AT+HTTPACTION HTTP 激活模式 .....	58
11.5. AT+HTTPREAD 读取 HTTP 响应数据 .....	59
11.6. AT+HTTPTERM 关闭 HTTP .....	59
11.7. HTTP get 流程测试 .....	60
11.8. HTTP post 流程测试 .....	60
11.9. HTTP onenet 平台流程测试 .....	61
11.10. HTTPS 钉钉机器人平台流程测试 .....	62
<b>12. MQTT 相关命令 .....</b>	<b>63</b>
12.1. AT+MQTTLS 设置 MQTT 的 TLS 参数 .....	63
12.2. AT+MQTTCONFIG 设置 MQTT 相关参数 .....	63
12.3. 客户端连接服务器, AT+MQTTCONNECT .....	64
12.4. 发布消息, AT+MQTTPUB .....	65
12.5. 订阅主题, AT+MQTTSUB .....	65
12.6. 取消订阅主题, AT+MQTTUNSUB .....	66
12.7. 查询连接状态, AT+MQTTSTATE .....	66
12.8. 客户端断开连接, AT+MQTTDISCONNECT .....	66
12.9. 数据上报, MQTTSUB .....	66
12.10. MQTT 测试 .....	67
<b>13. 短消息相关命令 .....</b>	<b>70</b>
13.1. AT+CMGF 配置短消息模式 .....	70
13.2. AT+CSCA 配置短消息服务中心地址 .....	70
13.3. AT+CMGW 写短消息 .....	71
13.4. AT+CMSS 从存储器发送短消息 .....	73
13.5. AT+CMGD 删除短消息 .....	73
13.6. AT+CMGL 列出存储器中的短消息 .....	74
13.7. AT+CMGR 读取短消息 .....	76

13. 8.	AT+CMGS 发送短消息	79
13. 9.	AT+CAMS 配置下行短信确认方式	80
13. 10.	AT+CNMI 新短消息指示	80
13. 11.	AT+CNMA 新短消息确认	82
13. 12.	AT+CSDH 显示文本格式参数	82
13. 13.	AT+CSMP 设置文本格式参数	83
13. 14.	+CMTI/+CMT 指示新的短消息	85
13. 15.	短信测试示例	86
14.	硬件相关命令	91
14. 1.	AT+CHIPTEMP 读取芯片内部温度	91
14. 2.	AT+BATVOL 读取电源供电电压	92
15.	低功耗控制命令	92
15. 1.	AT+LPM 低功耗控制	92
16.	工厂测试命令	92
16. 1.	AT+FACTORYTEST 工厂测试控制命令	92
	测试-工厂测试流程	93
17.	其它命令	96
17. 1.	AT+CMEE 设置日志输出明细	96
17. 2.	示例	97
18.	附表	98
18. 1.	CMEE 错误码	98
18. 2.	EXT 错误码	99
18. 3.	HTTP 错误码	100
18. 4.	MQTT 错误码	102
18. 5.	FTP 错误码	102

## 1. 文档介绍

### 1.1. 文档范围

本手册详细介绍了 UC8088 GPRS 模块提供的 AT 指令集。

### 1.2. 惯例和术语缩写

在本手册中可能会用到的术语解释如下：

- 1) MT 移动终端
- 2) TA 终端适配器
- 3) TE 终端设备
- 4) SIM 用户识别模块
- 5) ME 移动设备，包括 MT，TA 和 TE 等功能部件
- 6) MS 移动台，包括 ME 和 SIM
- 7) DCE 数据通信设备，或者传真 DCE（传真调制解调器，传真板）
- 8) DTE 数据终端设备，即一个嵌入式应用

在实际应用中，GPRS 模块可能被称为 ME，MS，TA 或 DCE，而通过串口发 AT 指令来控制 GPRS 模块的控制器则可能被称为 TE 或 DTE。

### 1.3. 命令语法

#### 1.3.1. 命令格式

本手册中所有命令行必须以” AT” 或” at” 作为开头，以回车换行（<CR><LF>）作为结尾。响应通常紧随命令之后，且通常以” <回车><换行><响应内容><回车><换行>”（<CR><LF><响应内容><CR><LF>）的形式出现。在命令介绍时，“<回车><换行>”（<CR><LF>）通常被省略了。

#### 1.3.2. 命令类型

通常命令可以有如下表所示的四种类型中的一种或多种形式。

类型	格式	说明
测试命令	AT+<cmd>=?	用于查询设置命令或内部程序设置的参数及其取值范围
查询命令	AT+<cmd>?	用于返回参数的当前值
设置命令	AT+<cmd>=<...>	用于设置用户自定义的参数值
执行命令	AT+<cmd>	用于读取只读参数或不需要额外参数的情况

表 1 命令类型与响应

### 1.3.3. 参数类型

命令参数虽然多种多样，但是都可以简单地归结为整数类型和字符串类型（包括不带双引号的字符串和带双引号的字符串）这两种基本的类型，如下表所示。

类型	示例
整数类型	123
字符串类型	abc
	“Hello, World!”

表 2 参数类型

### 1.4. 注意事项

- ❖ AT 串口输入时不支持回删键 (backspace) 功能
- ❖ 本文档+ERROR 指+CME ERROR 或者+EXT ERROR

## 2. 模组信息识别命令

### 2.1. AT+CPSV 读取 GPRS 协议栈软件版本信息

Command	Possible response(s)
+CPSV	+UPDATE_TIME:<update_time> +COMPILE_TIME:<compile_time> +TAG:<version_num> OK

#### 参数

<update\_time>: 程序最后更新时间

<compile\_time>: 版本编译时间

<version\_num>: 协议栈版本号

举例:

发 → ◇ AT+CPSV

收 ← ◆ +UPDATE\_TIME:Thu Jul 20 09:41:57 2023

收 ← ◆ +COMPILE\_TIME:Thu Jul 20 09:53:01 CST 2023

收 ← ◆ +TAG:v2.6

收 ← ◆ OK

### 2.2. AT+CGMI 读取厂商名

Command	Possible response(s)
+CGMI	<manufacturer> OK +ERROR: <err>
+CGMI=?	

#### 参数

<manufacturer>: 厂商名

举例:

发 → ◇ AT+CGMI

收 ← ◆ ucchip

收 ← ◆ OK

### 2.3. AT+CGMM 读取设备名

Command	Possible response(s)
+CGMM	<model> OK +ERROR: <err>
+CGMM=?	

#### 参数

<model>: 设备 ID

举例:

发 → ◇ AT+CGMM

收 ← ◆ 8088

收 ← ◆ OK

## 2.4. AT+CGMR 读取设备版本信息

Command	Possible response(s)
+CGMR	<revision> +ERROR: <err>
+CGMR=?	

### 参数

<revision>: 设备版本信息

举例:

发 → ◇ AT+CGMR

收 ← ◆ +SOFTV:012345678

收 ← ◆ +HARDV:00123456

收 ← ◆ +MANUF:ucchip

收 ← ◆ OK

## 2.5. AT+CGSN 读取 IMEI

Command	Possible response(s)
+CGSN	<sn> +ERROR: <err>
+CGSN=?	

### 参数

<sn>: IMEI 号

举例:

发 → ◇ AT+CGSN

收 ← ◆ 863711020146740

收 ← ◆ OK

## 3. 通用命令

### 3.1. AT+UCRESET 模块重启命令

Command	Possible response(s)
+UCRESET	OK +SYSTEM: START

举例

发 → ◇ AT+UCRESET

收 ← ◆ OK

收 ← ◆ +SYSTEM: START

### 3.2. AT+UCMODE 模式切换命令

Command	Possible response(s)
+UCMODE=<mode>	OK
+UCMODE?	+UCMODE: <mode> OK

+UCMODE=?	+UCMODE: ("GPRS","GNSS") OK
-----------	--------------------------------

### 参数

<mode>: 切换模式(GPRS、GNSS 两种)

### 举例

发→◇AT+UCMODE="GNSS" //切换到 GNSS 模式

收←◆OK

发→◇AT+UCMODE="GPRS" //切换到 GPRS 模式

收←◆OK

发→◇AT+UCMODE?

收←◆+UCMODE: <mode>

收←◆OK

发→◇AT+UCMODE=?

收←◆+UCMODE: ("GPRS","GNSS")

收←◆OK

## 3.3. AT+UCGNSSSTATE GNSS 定位状态查询

Command	Possible response(s)
+UCGNSSSTATE	+UCGNSSSTATE: <state> OK

### 参数

<state>: GNSS 定位状态

0 未定位

1 已定位

### 举例

发→◇AT+UCGNSSSTATE

收←◆+UCGNSSSTATE: 1

收←◆OK

## 3.4. AT+UCGNSSDATA GNSS 定位数据

Command	Possible response(s)
+UCGNSSDATA	+UCGNSSDATA: <lon>,<lat>,<speed>,<tn>,<tvn>,<time> OK

### 参数

<lon>: 经度

<lat>: 纬度

<speed>: 平面速度

<tn>: 参与定位卫星总数

<tvn>: 可见卫星总数

<time>: 日期+时间 (年/月/日-时:分:秒)

举例

发→◇AT+UCGNSSDATA

收←◆+UCGNSSDATA: 106.3007, 29.6025, 0.615, 5, 8, 2023/09/11-19:33:45

收←◆OK

## 4. (U) SIM 相关命令

### 4.1. AT+CIMI 读取 IMSI

Command	Possible response(s)
+CIMI	<IMSI> OK ERROR

#### 参数

<IMSI>: IMSI 码

举例:

发→◇AT+CIMI //读取 IMSI

收←◆460040812002376

收←◆OK

### 4.2. AT+CPIN 输入 PIN 码

Command	Possible Response(s)
+CPIN=<pin>[, <newpin>]	OK ERROR
+CPIN?	+CPIN: <code> OK

使用该配置命令, 可向 ME 发送操作需要的密码 (SIM PIN, SIM PUK, PH-SIM PIN 等)。

#### 参数

<pin>: 原密码 (字符串型, 需带引号) (SIM 卡的 PIN 码或 PUK 密码)

<new pin>: 新密码 (字符串型, 需带引号)

<code>:

READY 不需要输入任何密码

SIM PIN 需要输入 PIN 码

SIM PUK PIN 码解锁密码

#### 备注:

如果<code>是如下状态 相应的输入命令

SIM PIN AT+CPIN=<pin>

SIM PUK AT+CPIN=<puk>, <newpin>

举例:

发→◇AT+CPIN="1234"

收←◆OK

说明：输入 PIN 码” 1234”。

发→◇AT+CPIN?

收←◆+CPIN: SIM PIN

收←◆OK

说明：查看当前 SIM 卡的状态，SIM PIN 即说明需要解 PIN。

**注意：**使用+CPIN 前需先激活 SIM 卡，即先执行 AT+CFUN=1

### 4.3. AT+CPUN 查询 PIN 和 PUK 重试次数

Command	Possible Response(s)
+CPUN	+CPUN: <pinnumber>, <puknumber> OK

查询 PIN 和 PUK 剩余重试次数

#### 参数

<pinnumber>: PIN 剩余重试次数

<puknumber>: PUK 剩余重试次数

举例：

发→◇AT+CPUN

收←◆+CPUN: 3, 10

收←◆OK

说明：PIN 剩余重试次数为 3，PUK 剩余重试次数为 10。

### 4.4. AT+CCID 查询 CCID

Command	Possible response(s)
+CCID	+CCID: <CCID> OK

查询 UICC 中存储的 CCID

#### 参数

<CCID>: 集成电路卡识别码。

举例：

发→◇AT+CCID //读取 CCID

收←◆+CCID: 898602a231bc83890991

收←◆OK

### 4.5. AT+CNUM 查询用户号码

Command	Possible response(s)
+CNUM	+CNUM: <number>, <type> OK

查询 UICC 中存储的 MSISDN

#### 参数

<number>: 电话号码

<type>: 整数型的八位字节地址类型

119 未知类型

145 国际类型

161 国内型

举例:

发→◇AT+CNUM //读取 MSISDN

收←◆+CNUM: "hywang", 8615723244049, 145

收←◆OK

## 4.6. AT+CRSM 有限制的 SIM 卡访问

Command	Possible response(s)
+CRSM=<command>[, <fileid>[, <p1>[, <p2>[, <p3>[, <data>]]]]	+CRSM:<sw1>,<sw2>[, <response>] OK

### 参数

<command>: 命令类型

176 读取二进制文件

178 读取记录文件

192 获取响应

214 更新二进制文件

220 更新记录文件

242 状态

<fileid>: 文件 FID, 整型, 除了 STATUS 外, 其他命令都需要该参数。

<P1>, <P2>, <P3>: 命令参数, 整型, 除了 GET RESPONSE 和 STATUS 外, 其他命令都需要该参数。根据 27007 的定义, 三个参数要么同时存在, 要么同时不存在, 其他情况都是非法。

<data>: 写到卡的数据, 十六进制字符的形式, 例如 7F205F70

注: 目前设计 P3 固定为 0, 读取整个文件有效数据。

<sw1>, <sw2>: 整型; 来自 SIM 卡实际命令的执行信息

<response>: 前一个成功执行的命令的返回值

举例:

1) READ BINARY

发→◇AT+CRSM=176, 12258, 0, 0, 0

收←◆+CRSM: 144, 0, "9868202A13CB38989019"

收←◆OK

说明: 二进制文件, 读取 ICCID(0x2FE2)

发→◇AT+CRSM=176, 28423, 0, 0, 0

收←◆+CRSM: 144, 0, "084906202583799019"

收←◆OK

说明：二进制文件，读取 IMSI (0x6F07)

2) READ RECORD

发→◇AT+CRSM=178,28482,1,4,0

收←◆+CRSM: 144,0,"FF0891683108200305F0FFFFFFFFFFFF"

收←◆OK

说明：线性固定文件，读取 SMS (0x6F42)

3) UPDATE RECORD

发→◇AT+CRSM=220,28480,1,4,28,FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF0891688123436604F1FFFFFFFFFFFF

收←◆+CRSM: 144,0

收←◆OK

说明：更新记录文件，写电话号码到 MSISDN (0x6F40)

4) STATUS

发→◇AT+CRSM=176,12258,0,0,0

收←◆+CRSM: 144,0,"9868202A13CB38989019"

收←◆OK

发→◇AT+CRSM=242

收←◆+CRSM: 144,0,"62208202782183023F00A5038001718A01058B032F060CC60990014083010183010A"

收←◆OK

说明：读取 ICCID 的数据，再查询对应目录的信息。

## 5. 网络服务相关命令

### 5.1. AT+CELLLOCK 锁频

Command	Possible Response(s)
+CELLLOCK= [<state>],[<freq>]	OK ERROR
+CELLLOCK?	+CELLLOCK: <state>,<freq>
+CELLLOCK=?	+CELLLOCK: (list of supported <state>s),freq

锁定 MS 到固定频率，注意只能在未注册状态才能锁频成功，支持的锁频频率范围：

E-GSM900 DCS1800 PCS1900

#### 参数

<state>: 锁屏操作类型

0:取消锁屏

## 1:锁屏

<freq>: 频率值 HZ

举例:

发→◇AT+CELLLOCK=? //获取可配置枚举值

收←◆+CELLLOCK: (0,1),freq

收←◆OK

发→◇AT+CELLLOCK=1,935400000 //锁定频率 935.4MHZ

收←◆OK

发→◇AT+CELLLOCK? //查询当前锁频状态

收←◆+CELLLOCK: 1,935400000 //锁定频率 935400000HZ

收←◆OK

发→◇AT+CELLLOCK? //查询当前锁频状态

收←◆+CELLLOCK: 0,0 //未锁频

收←◆OK

## 5.2. AT+CGCLASS 能力模式设置并附着网络

Command	Possible Response(s)
+CGCLASS= [<class>]	OK ERROR

## 参数

<class>

CC MT 只运行 CS 服务

CG MT 只运行 PS 服务

举例:

发→◇AT+CGCLASS=CC //附着到 CS 域网络

收←◆OK

发→◇AT+CGCLASS=CG //附着到 PS 域网络

收←◆OK

## 5.3. AT+CGREG PS 域网络注册状态

Command	Possible response(s)
+CGREG=[<n>]	OK
+CGREG?	+CGREG: <n>,<stat>[,<lac>,<ci>] OK ERROR

+CGREG=?	+CGREG: (list of supported <n>s) OK
----------	--

当 MT 的 GPRS 网络注册状态变化时，是否主动上报+CGREG=<stat>注册状态值。

### 参数

<n>

- 0: 禁止主动上报
- 1: 使能主动上报

<stat>

- 0: 未注册，MT 当前不在搜索注册一个新的运营商
- 1: 已注册，在本地网络
- 2: 未注册，MT 当前正在搜索注册一个新的运营商
- 3: 拒绝注册
- 4: 位置
- 5: 已注册，漫游中

<lac>

基站的位置区域码

<ci>

小区 ID

举例：

发→◇AT+CGREG=?

收←◆+CGREG: (0-1)

收←◆OK

发→◇AT+CGREG=1

收←◆OK

发→◇AT+CGREG?

收←◆+CGREG: 0, 0, 0, 0

收←◆OK

示例：

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CGREG=1 //设置命令 使能主动上报网络注册情况

收←◆OK

发→◇AT+CGATT=1 //GPRS 注册

收←◆+CGREG: 2 //主动上报 2:stat 未注册，MT 当前正在搜索注册一个新的运营商

收←◆+CGREG: 1 //主动上报 1:stat 已注册, 在归属网络

收←◆OK //AT+CGATT 注册成功返回 OK

发→◇AT+CGREG?

收←◆+CGREG: 1, 1, 13122, 32451 //1:开启主动上报 1:已注册 13122:lac 32451:ci

收←◆OK

#### 5.4. AT+CGREG CS 域网络注册状态

Command	Possible response(s)
+CGREG=[<n>]	OK
+CGREG?	+CGREG: <n>,<stat>[,<lac>,<ci>] OK ERROR
+CGREG=?	+CGREG: (list of supported <n>s) OK

当 MT 的 GPRS 网络注册状态变化时, 是否主动上报+CGREG=<stat>注册状态值。

#### 参数

<n>

0: 禁止主动上报

1: 使能主动上报

<stat>

0: 未注册, MT 当前不在搜索注册一个新的运营商

1: 已注册, 在本地网络

2: 未注册, MT 当前正在搜索注册一个新的运营商

3: 拒绝注册

4: 位置

5: 已注册, 漫游中

<lac>

基站的位置区域码

<ci>

小区 ID

举例:

发→◇AT+CGREG=?

收←◆+CGREG: (0-1)

收←◆OK

发→◇AT+CGREG=1

收←◆OK

发→◇AT+CGREG?

收←◆+CGREG: 0, 0, 0, 0

收←◆OK

## 5.5. AT+CSQ 检测信号强度

Command	Possible response(s)
+CSQ	+CSQ: <rss>, <ber> +ERROR: <err>
+CSQ=?	+CSQ: (list of supported <rss>s), (list of supported <ber>s)

### 参数

<rss> 接收信号强度

0	<=-113dBm
1	-111dBm
2-30	-109 到-53dBm
31	-51dBm or greater
99	未知或不可检测

<ber> 信道误码率, 0~7 表示误码率由低到高, 99 表示未知.

0	BER < 0.2%
1	0.2% < BER < 0.4%
2	0.4% < BER < 0.8%
3	0.8% < BER < 1.6%
4	1.6% < BER < 3.2%
5	3.2% < BER < 6.4%
6	6.4% < BER < 12.8%
7	12.8% < BER
99	未知或不可测

举例:

发→◇AT+CSQ=?

收←◆+CSQ: (2-31, 99), (0-7, 99)

收←◆OK

发→◇AT+CSQ

收←◆+CSQ: 21, 99 //21 rssi 为-71dBm, 99 ber 未知

收←◆OK

## 5.6. AT+COPS 运营商选择

Command	Possible response(s)
+COPS=[<mode>[<format>, [<stat>, <oper>]]]	+ERROR: <err> 或者 正确可用的网络
+COPS?	+COPS: <mode>[<format>, <stat>, <oper>] +ERROR: <err>

+COPS=?	+COPS: [list of supported (<stat>,long alphanumeric <oper>,short alphanumeric <oper>,numeric <oper>)s] [,,(list of supported <mode>s),(list of supported <format>s)] +ERROR: <err>
---------	--

搜索网络，设置注册网络，查询注册网络

### 参数

<mode>运营商选择模式

- 0-自动 忽略 format, oper
- 1-手动 format, oper 必须有
- 2-强制注销网络
- 3-设置查询格式(用于+cops?返回格式)
- 4-手动/自动 因为有手 oper 自动必须有，如果手动强制失败，则进入自动模式

<format>运营商名字表示方式

- 0:长字符串，如 chinaMobile
- 1:短字符串，如 cmee
- 2:数组，如 46000

<stat>运营商可用状态

- 0 unknown
- 1 available 可用的运营的
- 2 current 当前 SIM 卡正在使用的 PLMN
- 3 forbidden 禁止 PLMN

举例：

搜索 PLMN，列出当前搜索到的 PLMN，并告诉哪些运营商可用；只搜网不注册。

返回：[stat,"长串","短串",Num 编号],,支持的 mode,支持的 format。

发→◇AT+COPS=?

收←◆+COPS: (1,"","","26242"),(0,"","","46242")

收←◆OK

说明：

(1,"","","26242")

1: 搜索到的运营商可用

"": 运营商长字符串为空

"": 运营商短字符串为空

"26242": 运营商数字编号

发→◇AT+COPS=1,2,"46000" //空闲态下强制设置运营商并注册到运营商网络。

收←◆+COPS: (1,"","","26242"),(0,"","","46242")

收←◆OK

发→◇AT+COPS?

收←◆+COPS: 0, 2, "26242"

收←◆OK

说明:

+COPS: 0, 2, "26242"

0: 自动选择运营商模式

2: 选择的运营商用数字表示

"26242": 选择的运营商

示例

发→◇AT+CFUN=1 //激活 SIM 卡

收←◆OK

发→◇AT+COPS=? //搜索当前 PLMN

收←◆+COPS: (1, "", "", "26242") //1: stat 可用的

收←◆OK

发→◇AT+COPS=1, 2, "46000" //手动设置 PLMN

收←◆OK

发→◇AT+CGATT? //查询注册状态

收←◆+CGATT: 1 //已注册

收←◆OK

发→◇AT+CGATT=0 //注销

收←◆OK //注销成功

发→◇AT+COPS=1, 2, "26242" //手动设置不存在 PLMN

收←◆+COPS: (1, "", "", "46000") //手动设置 PLMN 失败, 返回可用的 PLMN 列表

收←◆OK

发→◇AT+CGATT?

收←◆+CGATT: 0 //未注册

收←◆OK

## 6. GPRS 相关命令

### 6.1. AT+CFUN 设定模块功能

Command	Possible response(s)
+CFUN=[<fun>[, <rst>]]	OK +ERROR: <err>
+CFUN?	+CFUN: <fun> +ERROR: <err>
+CFUN=?	+CFUN: (list of supported <fun>s), (list of supported <rst>s) +ERROR: <err>

该命令用于选择 MT 功能，当选择全功能时消耗最大功率，当选择最小功能时消耗最小功率。

#### 参数

<fun>功能选择

0:最小功能

1:全功能

<rst>是否需要复位

0:不复位

1:复位

举例:

发→◇AT+CFUN=?

收←◆+CFUN: (0, 1), (0)

收←◆OK

发→◇AT+CFUN=1 //设备上电

收←◆OK

发→◇AT+CFUN?

收←◆+CFUN: (0, 1), (0)

收←◆OK

### 6.2. AT+CGATT GPRS 注册注销

Command	Possible Response(s)
+CGATT= [<state>]	OK ERROR
+CGATT?	+CGATT: <state>
+CGATT=?	+CGATT: (list of supported <state>s)

#### 参数

<state>: GPRS 注册状态

0:注销

1:注册

举例:

发→◇AT+CGATT=?

收←◆+CGATT: (0, 1)

收 ← ◆ OK

发 → ◇ AT+CGATT=1

收 ← ◆ OK

发 → ◇ AT+CGATT? // 查询注册状态

收 ← ◆ +CGATT: 1

收 ← ◆ OK

### 6.3. AT+CGACT PDP 激活去激活

Command	Possible Response(s)
+CGACT=[<state> [, <cid>[, <cid>[, ...]]]]	OK ERROR
+CGACT?	+CGACT: <cid>, <state> [<CR><LF>+CGACT: <cid>, <state> [...]]
+CGACT=?	+CGACT: (list of supported <state>s)

#### 参数

<state>指示 PDP 上行文激活还是去激活

0:去激活

1:激活

<cid>PDP 上下文标识 ID, 可通过+CGDCONT 设置 ID 的 PDP 激活参数, 未设置时使用默认 PDP 激活参数

注: state 必须赋值, cid 可选参数

state:激活 cid:无, 则激活所有已定义的 PDP;

state:去激活 cid:无, 则去激活所有已激活的 PDP 激活时, 如果 MT 未注册, 则先注册在激活。

注意: 目前我们系统最大只支持 2 个 cid 编号为 1, 2; 其它值为异常值

举例:

发 → ◇ AT+CGACT=?

收 ← ◆ +CGACT: (1, 0)

收 ← ◆ OK

发 → ◇ AT+CGACT=1, 1 // 激活 cid 为 1 的 PDP

收 ← ◆ OK

发 → ◇ AT+CGACT?

收 ← ◆ +CGACT:1, 1 // cid 为 1 的 PDP 处于激活状态

收 ← ◆ OK

## 示例

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CGACT=? //测试命令 返回支持的 stat 枚举值

收←◆+CGACT:(0,1) //支持 0:deactivated 1:activated

收←◆OK

发→◇AT+CGACT=1,1,2 //设置命令 1:激活 PDP 1,2:cid 为 1 和 2 两个 PDP

收←◆OK

发→◇AT+CGACT? //查询当前 PDP 激活状态

收←◆+CGACT:1,1 //1:cid 为 1 1:激活

收←◆+CGACT:2,1 //2:cid 为 2 1:激活

收←◆OK

## 6.4. AT+CGQREQ 设置 QOS

Command	Possible Response(s)
+CGQREQ=[<cid> [,<precedence> <delay> [,<reliability.> [,<peak> [,<mean>]]]]]	OK ERROR
+CGQREQ?	+CGQREQ: <cid>, <precedence>, <delay>, <reliability>, <peak>, <mean> [<CR><LF>+CGQREQ: <cid>, <precedence>, <delay>, <reliability.>, <peak>, <mean> [...]]
+CGQREQ=?	+CGQREQ: <PDP_type>, (list of supported <precedence>s), (list of supported <delay>s), (list of supported <reliability>s), (list of supported <peak>s), (list of supported <mean>s) [<CR><LF>+CGQREQ: <PDP_type>, (list of supported <precedence>s), (list of supported <delay>s), (list of supported <reliability>s), (list of supported <peak>s), (list of supported <mean>s) [...]]

在 PDP 激活前，设置 PDP 激活请求 QOS 参数值，是对 AT+CGDCONT 命令 (PDP 上下文定义) 补充

## 参数

<cid>PDP 上下文标识 ID

<PDP\_type>PDP 类型, X.25 IP IPV6 OSPIH PPP

<precedence>优先级

0:预定的

1:高优先级

2:正常优先级

3:低优先级

<delay>时延

0:默认值

1~3:QOS 延时级别

<reliability>可靠性

0:默认值

1:LLC ack, LLC data prot, RLC ack, GTP ack

2:LLC ack, LLC data prot, RLC ack, GTP unack

3:LLC unack, LLC data prot, RLC ack GTP unack

4:LLC unack, LLC data prot, RLC unack GTPunack

5:LLC unack, no LLC data prot, RLC unack

<peak>峰值吞吐量

0:默认值

1~9:峰值吞吐量级别

<mean>平均吞吐量

0:默认值

1~18:平均吞吐量级别

31:尽力而为的平均吞吐量级别

注：设置 PDP 上下文 QOS 参数，只设置 cid 时，则对于 cid PDP 上行文用默认 QOS。

举例：

发→◇AT+CGQREQ=?

收←◆+CGQREQ: "IP", (1-3), (1-4), (1-5), (1-9), (1-18, 31)

收←◆OK

说明：

+CGQREQ: "IP", (1-3), (1-4), (1-5), (1-9), (1-18, 31)

“IP”：只支持 IP 的 PDP 类型

(1-3)：QOS 优先级，详见 GSM03.60

(1-4)：QOS 时延，详见 GSM03.60

(1-5)：QOS 可靠性，详见 GSM03.60

(1-9)：QOS 峰值吞吐量，详见 GSM03.60

(1-18, 31)：QOS 平均吞吐量，详见 GSM03.60

发→◇AT+CGQREQ=1, 1, 4, 5, 2, 14

收←◆OK

发→◇AT+CGQREQ? //查询当前 PDP QOS 设置值；注意只能查询到未激活的 PDP QOS 值

收←◆+CGQREQ: 1, 0, 0, 0, 0, 0

收←◆OK

示例

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CGDCONT=1,"IP" //设置 PDP cid=1 ip 类型

收←◆OK

发→◇AT+CGQRE=? //测试命令

收←◆+CGQREQ: "IP", (1-3), (1-4), (1-5), (1-9), (1-18,31)//只支 IP 类型 QOS 设置

收←◆OK

发→◇AT+CGQRE? //查询命令 当前 PDP 设置情况

收←◆+CGQREQ: 1, 0, 0, 0, 0, 0

收←◆OK

发→◇AT+CGQRE=1, 1, 4, 5, 2, 14 //设置命令 设置 cid 为 1 的 QOS

收←◆OK

发→◇AT+CGQRE? //查询命令 当前 PDP 设置情况

收←◆+CGQREQ: 1, 1, 4, 5, 2, 14

收←◆OK

发→◇AT+CGACT=1,1 //激活 cid 为 1 的 PDP

收←◆OK

## 6.5. AT+CGQMIN 设置 PDP 最小 QOS

Command	Possible Response(s)
+CGQMIN=[[<cid> [,<precedence > [,<delay> [,<reliability.> [,<peak> [,<mean>]]]]]]	OK ERROR
+CGQMIN?	+CGQMIN: <cid>, <precedence >, <delay>, <reliability>, <peak>, <mean> [<CR><LF>+CGQMIN: <cid>, <precedence >, <delay>, <reliability.>, <peak>, <mean> [...]]
+CGQMIN=?	+CGQMIN: <PDP_type>, (list of supported <precedence>s), (list of supported <delay>s), (list of supported <reliability>s) , (list of supported <peak>s), (list of supported <mean>s) [<CR><LF>+CGQMIN: <PDP_type>, (list of supported <precedence>s), (list of supported <delay>s), (list of supported <reliability>s) , (list of supported <peak>s), (list of supported <mean>s) [...]]

设置 PDP 激活请求最小 QOS 参数值。

举例：

发→◇AT+CGQMIN=?

收←◆+CGQMIN: "IP", (1-3), (1-4), (1-5), (1-9), (1-18, 31)

收←◆OK

说明：

+CGQMIN: "IP", (1-3), (1-4), (1-5), (1-9), (1-18, 31)

“IP”：只支持 IP 的 PDP 类型

(1-3)：QOS 优先级，详见 GSM03.60

(1-4)：QOS 时延，详见 GSM03.60

(1-5)：QOS 可靠性，详见 GSM03.60

(1-9)：QOS 峰值吞吐量，详见 GSM03.60

(1-18, 31)：QOS 平均吞吐量，详见 GSM03.60

GSM 03.60

precedence: 优先级

delay: 时延

reliability: 可靠性

peak: 峰值吞吐量

mean: 平均吞吐量

发→◇AT+CGQMIN=1, 1, 4, 5, 2, 14 //设置 PDP 上下文最小 QOS 参数

收←◆OK

发→◇AT+CGQMIN?

收←◆+CGQMIN: 1, 0, 0, 0, 0, 0

收←◆OK

示例

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CGDCONT=1, "IP" //设置 PDP

收←◆OK

发→◇AT+CGQMIN=? //测试命令 命令参数枚举值

收←◆+CGQMIN: "IP", (1-3), (1-4), (1-5), (1-9), (1-18, 31)

收←◆OK

发→◇AT+CGQMIN? //查询命令 当前定义 PDP Min\_QOS 值

```

收←◆+CGQMIN: 1,0,0,0,0,0
收←◆OK

发→◇AT+CGQMIN=1,1,4,5,2,14 //设置命令 设置 cid 为 1 的 Min_QOS
收←◆OK

发→◇AT+CGQMIN? //查询命令 当前定义 PDP Min_QOS 值, 结果应该与设置相同
收←◆+CGQMIN: 1,1,4,5,2,14
收←◆OK

发→◇AT+CGACT=1,1 //激活 PDP
收←◆ERROR //错误, 应用 PDP accept 中的 QOS 与设置的 Min_QOS 校验未通过

```

## 6.6. AT+CGPADDR 显示 PDP 地址

Command	Possible response(s)
+CGPADDR=[<cid> [,<cid> [,...]]]	+CGPADDR: <cid>,<PDP_addr> [<CR><LF>+CGPADDR: <cid>,<PDP_addr> [...]]
+CGPADDR=?	+CGPADDR: (list of defined <cid>s)

举例:

```

发→◇AT+CGPADDR=?
收←◆+CGPADDR: (1,2) //当前已定义 cid 1 和 2 的 PDP
收←◆OK

发→◇AT+CGPADDR=1,2
收←◆+CGPADDR: 1,"192.168.40.13"
收←◆+CGPADDR: 1,"192.168.40.13"
收←◆OK

```

示例

```

发→◇AT+CFUN=1
收←◆OK

发→◇AT+CGACT=1,1,2 //激活 PDP cid 为 1 和 2
收←◆OK

发→◇AT+CGACT? //查询 PDP 激活状态
收←◆+CGACT: 1,1 //cid 为 1 激活

```

```

收←◆+CGACT: 2,1 //cid为2 激活
收←◆OK

发→◇AT+CGPADDR=? //读取当前定义 PDP cid 值
收←◆+CGPADDR: (1,2) //当前 cid 有 1 和 2
收←◆OK

发→◇AT+CGPADDR=1,2 //读取 cid 1,2 的 PDP 地址
收←◆+CGPADDR: 1,"192.168.40.13"
收←◆+CGPADDR: 2,"192.168.40.14"
收←◆OK

发→◇AT+CGPADDR= //读取所有 cid PDP 地址
收←◆+CGPADDR: 1,"192.168.40.13"
收←◆+CGPADDR: 2,"192.168.40.14"
收←◆OK
    
```

## 6.7. AT+CGDCONT 定义 PDP 上下文

Command	Possible response(s)
+CGDCONT=[<cid> [, <PDP_type> [, <APN> [, <PDP_addr> [, <d_comp> [, <h_comp>]]]]]	OK ERROR
+CGDCONT?	+CGDCONT: <cid>, <PDP_type>, <APN>, <PDP_addr>, <d_comp>, <h_comp>[, <pd1>[, ...[, <pdN>]]] [<CR><LF>+CGDCONT: <cid>, <PDP_type>, <APN>, <PDP_addr>, <d_comp>, <h_comp> [...]]
+CGDCONT=?	+CGDCONT: (range of supported <cid>s), <PDP_type>,, (list of supported <d_comp>s), (list of supported <h_comp>s) [<CR><LF>+CGDCONT: (range of supported <cid>s), <PDP_type>,, (list of supported <d_comp>s), (list of supported <h_comp>s) [...]]

PDP 上行文定义，用于后续 PDP 激活使用；

### 参数

<cid>PDP 上下文标识 ID

<PDP\_type>PDP 类型，X.25 IP IPV6 OSPiH PPP

<APN>业务接入点，常见如 cmnet cmwap ims

<PDP\_addr>指定 PDP IP 地址

<d\_comp>PDP 数据压缩指示

0:off

1:on

2:V.42bit

<h\_comp>PDP 数据头压缩

0:off  
1:on  
2:RFC1144  
3:RFC2507

举例:

发→◇AT+CGDCONT=? //查看系统支持的 PDP 参数范围

收←◆+CGDCONT: (1-2), "IP", ,, (0, 1), (0, 1)

收←◆OK

说明:

+CGDCONT: cid 范围, pdp\_type, ,, pdp 数据压缩枚举值, pdp 头压缩枚举值

+CGDCONT: (1-2), "IP", ,, (0, 1), (0, 1)

(1-2): 支持两个 CID 1 和 2

"IP": 支持 PDP 类型为 IP

(0, 1): 支持 PDP 数据压缩 0 关闭 1 打开

(0, 1): 支持 PDP 头压缩 0 关闭 1 打开

//定义 PDP context, 如 PDP 类型/CID 等, 并不发起 PDP 激活, 如需激活需配合 AT+CGACT

发→◇AT+CGDCONT=1, IP //1: 设置 CID 为 1 的 PDP 上下文 "IP": PDP 类型为 IP

收←◆OK

发→◇AT+CGDCONT?

收←◆+CGDCONT: 1, "IP", "", "", 0, 0

收←◆OK

说明:

+CGDCONT: 1, "IP", "", "", 0, 0

1: 已设置 CID 值为 1 的 PDP

"IP": PDP 类型为 IP

0: 关闭 PDP 数据压缩

0: 关闭 PDP 头压缩

示例

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CGDCONT=? //测试命令 返回命令设置说明

收←◆+CGDCONT: (1-2), "IP", ,, (0, 1), (0, 1)

收←◆OK

发→◇AT+CGDCONT=1, IP//定义 cid 为 1, PDP 类型为 IP, 注意 IP 必须大写

收←◆OK

发→◇AT+CGDCONT? //查询命令 pdp 设置情况

收←◆+CGDCONT: 1,"IP", "", "", 0, 0

收←◆OK

发→◇AT+CGACT=1,1 //PDP 激活操作，激活 CID 为 1 的 PDP

收←◆OK

发→◇AT+CGACT? //查询命令 PDP 激活情况

收←◆+CGACT: 1,1 //cid 为 1 激活状态

收←◆OK

发→◇AT+CGDCONT? //查询命令 pdp 设置情况

收←◆+CGDCONT: 1,"IP", "", "", 0, 0 //1:cid 值 IP:pdp\_type 为 IP

收←◆OK

## 6.8. AT+CGDATA 进入数据态

Command	Possible Response(s)
+CGDATA=[<L2P> , [<cid> [, <cid> [, ...]]]]	CONNECT ERROR
+CGDATA=?	+CGDATA: (list of supported <L2P>s)

AT 命令态切换到数据态，开始传输数据，在此期间其他 AT 指令无效，要等数据传送完且由数据态切换到命令态成后才能继续 AT 指令。执行该命令相当于完成 attach 和 activePDP。

<L2P>TE 和 MT 间层 2 协议，常见如 PPP PAD X25

举例：

发→◇AT+CGDATA=? //列出数据态支持的 TE 和 MT 间 L2 协议

收←◆+CGDATA: "PPP" //"PPP": 在 TE 和 MT 之间使用 PPP 协议

收←◆OK

发→◇AT+CGDATA="PPP", 1

收←◆CONNECT

说明：

设置对应 CID 的 PDP 从命令态进入数据态。

所有参数都是可选参数，不配置时选择默认 PDP

L2P 配置时，只能配置 PPP

cid 配置时，指定对应的 PDP

示例

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CGDATA="PPP",1 // cid 为 1 的 PDP 进入数据态

收←◆CONNECT

### 示例-完整 PDP 激活流程

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CGDCONT=? //测试命令 返回命令设置说明

收←◆+CGDCONT: (1-2), "IP",,, (0, 1), (0, 1)

收←◆+CGDCONT: (1-2), "PPP",,, (0, 1), (0, 1)

收←◆OK

发→◇AT+CGDCONT=1,"IP"//定义 cid 为 1, PDP 类型为 IP, 注意 IP 必须大写

收←◆OK

发→◇AT+CGDCONT? //查询命令 pdp 设置情况

//1:cid 值 IP:pdp\_type 为 IP 不支持头压缩数据压缩

收←◆+CGDCONT: 1, "IP", "", "", 0, 0

收←◆OK

发→◇AT+CGACT=1,1 //PDP 激活操作, 激活 CID 为 1 的 PDP

收←◆OK //返回 OK 表示 PDP 激活成功

发→◇AT+CGACT? //查询命令 PDP 激活情况

收←◆+CGACT: 1,1 //cid 为 1 激活状态

收←◆OK

### 示例-完整数据传输流程

发→◇AT+CFUN=1 //设备上电

收←◆OK

发→◇AT+CGDATA=? //获取 TE 和 MT 间 L2 协议

收←◆+CGDATA: "PPP"

收←◆OK

```
//cid 为 1 PDP 进入数据态, 完成 attach pdp 激活命令态到数据态切换
发→◇AT+CGDATA="PPP", 1
收←◆CONNECT //表示切换到数据态成功, 后续可进行数据发送
```

## 7. 呼叫控制命令

### 7.1. ATD 呼叫指令

Command	Possible Response(s)
ATD<dial string>; ATD><n>; ATD>"name";	OK ERROR  result

该命令用于拨打电话。

#### 参数

<dial string> 电话号码

<result> OK 命令执行成功

ERROR 命令执行失败

CONNECT 对端接听电话

BUSY 对端未接电话直接挂断 **(注: 基站端下发的挂断码可能不一样, 对方**

**忙/呼叫方不可达有时未界定)**

NO CARRIER 呼叫方不可达/被挂断电话

NO ANSWER 振铃超时未响应

举例:

发→◇ATD15723244049; //拨打电话

收←◆OK

收←◆CONNECT //对方应答

发→◇ATD>2; //使用当前电话本中的记录号拨打电话

收←◆OK

收←◆CONNECT

发→◇ATD>"lisi"; //使用当前电话本中的名称来拨打电话, 对方接通后主动挂断

收←◆OK

收←◆CONNECT

收←◆NO CARRIER

### 7.2. ATA 来电接听

Command	Possible Response(s)
ATA	OK ERROR

应答来电, 建立通话链接。

当回码显示 RING 时表示有来电。

### 参数

无

举例:

发→◇ATA //接听电话

收←◆OK

## 7.3. ATH 挂断通话

Command	Possible Response(s)
ATH	OK ERROR

### 参数

无

举例:

发→◇ATH //挂断通话连接 或者 拒绝接听来电，直接挂断

收←◆OK

## 示例-主被叫通话 AT 应用

### 主叫 (UC8088——>手机)

(1) 主叫-呼叫方不可达 (如: 己方或对方网络不好)

发→◇ATD456789;

收←◆OK

收←◆NO CARRIER

(2) 主叫-对方未接电话

发→◇ATD15723244094;

收←◆OK

收←◆NO ANSWER //一段时间后返回

(3) 主叫-对方接听电话后挂断

发→◇ATD15723244094;

收←◆OK

收←◆CONNECT

收←◆NO CARRIER

(4) 主叫-对方接听电话后己方挂断

发→◇ATD15723244094;

收←◆OK

收←◆CONNECT

发→◇ATH

收←◆OK

(5) 主叫-对方处于通话状态

发→◇ATD15723244094;

收←◆OK

收←◆BUSY

**被叫 (UC8088<——手机)**

(1) 被叫-未接电话

收←◆RING //振铃

收←◆NO CARRIER

(2) 被叫-未接电话对方挂断

收←◆RING //振铃

收←◆NO CARRIER

(3) 被叫-接听电话后对方挂断

收←◆RING

发→◇ATA

收←◆OK

收←◆NO CARRIER

(4) 被叫-接听电话后自己挂断

收←◆RING

发→◇ATA

收←◆OK

发→◇ATH

收←◆OK

(5) 被叫-己方处于通话状态

手机拨打板子, 板子无响应, 且手机端显示“对方忙”

**7.4. AT+CLIP 设置来电显示**

Command	Possible Response(s)
+CLIP=<n>	OK ERROR

**参数**

<n>

0: 禁用来电号码的显示

1: 启用来电号码的显示

举例:

发→◇AT+CLIP=1 //启用来电号码的显示

收←◆OK

**7.5. AT+CLCC 查询当前呼叫列表**

Command	Possible Response(s)
+CLCC	+CLCC: <id1>,<dir>,<stat>,<mode>,<empty>[,<number>,<type>] OK

## 参数

<id1> 呼叫标识符。如果呼叫标识符为 0，则表示无呼叫。

<dir> 呼叫方向

0: 表示呼出

1: 表示呼入

<stat> 呼叫状态

0: 表示通话中

~~1: 表示呼叫保持~~

2: 表示拨号中 (MO)

3: 表示呼叫振铃 (MO)

4: 表示正在呼入 (MT)

~~5: 表示呼叫等待 (MT)~~

<mode> 呼叫模式 ? ? ? ?

0: 表示语音呼叫

~~1: 表示数据呼叫~~

~~2: 表示传真呼叫~~

~~3: 表示语音/传真呼叫~~

~~4: 表示语音/数据呼叫~~

<mpty> 呼叫组成员的数量

0: 表示单一呼叫

~~1: 表示呼叫组成员之一~~

<number> 呼叫电话号码

<type> 电话号码类型

### 举例:

#### (1) 未进行主被叫操作

发 → ◇ AT+CLCC

收 ← ◆ OK // 直接返回 OK，不返回 +CLCC: 数据

#### (2) 主叫 (UC8088 → 手机)

// 拨号中

发 → ◇ AT+CLCC

收 ← ◆ +CLCC: 1, 0, 2, 0, 0, "456789", 128

收 ← ◆ OK

// 振铃中

发 → ◇ AT+CLCC

收 ← ◆ +CLCC: 1, 0, 3, 0, 0, "456789", 128

收 ← ◆ OK

// 通话中

发→◇AT+CLCC  
 收←◆+CLCC: 1, 0, 0, 0, 0, "456789", 128  
 收←◆OK

### (3) 被叫 (UC8088←——手机)

//振铃中  
 发→◇AT+CLCC  
 收←◆+CLCC: 1, 1, 4, 0, 0, "456789", 161  
 收←◆OK

//通话中  
 发→◇AT+CLCC  
 收←◆+CLCC: 1, 1, 0, 0, 0, "456789", 161  
 收←◆OK

## 7.6. AT+CSVC 扬声器音量控制

Command	Possible Response(s)
+CSVC=<level>	OK ERROR
+CSVC?	+CSVC: <level> OK
+CSVC=?	+CSVC: (<level>取值列表) OK

设置扬声器的音量大小。

### 参数

<level> 扬声器音量等级

0: 静音

1-4: 逐级增大

5: 最高音量等级

举例:

发→◇AT+CSVC=?  
 收←◆+CSVC: (1-5)  
 收←◆OK

发→◇AT+CSVC=5 //设置最高音量等级  
 收←◆OK

发→◇AT+CSVC?  
 收←◆+CSVC: 5  
 收←◆OK

## 7.7. AT+VTS DTMF 音调发送

Command	Possible Response(s)
+VTS=?	+VTS: (<dtmf-string>取值列表) OK
+VTS=<dtmf-string>	OK ERROR

该命令可发送一个或多个 ASCII 字符，这些字符的作用是使 MSC (Mobile Switching Center) 为远程用户发射双音多频 DTMF (Dual Tone Multi Frequency) 音调。

### 参数

<dtmf-string> 最大值为 20 个字符。必须放在双引号间。它由下面的字符组成，间隔以“,” 隔开，但是单个字符不需要引号。<dtmf> 单个 ASCII 字符，范围：0-9, #, \*, A-D。

举例：

发 → ◊ AT+VTS=1 // 发送 DTMF 音调 1

收 ← ◆ OK

### 示例-完整正常执行拨打接听电话

发 → ◊ AT+CFUN=1 // 开机读卡

收 ← ◆ OK

发 → ◊ AT+CGCLASS=CC // 附着到网络

收 ← ◆ OK

发 → ◊ AT+CLIP=1 // 设置来电显示(无使用顺序限制)

收 ← ◆ OK

发 → ◊ ATD15723244094; // 拨打电话到“15723244094”

收 ← ◆ OK

发 → ◊ ATH // 主动挂断电话

收 ← ◆ OK

**// 来电**

收 ← ◆ RING // 来电振铃

收 ← ◆ +CLIP: "15723244094", 161 // 显示对方的电话号码

发 → ◊ ATA // 接听电话

收 ← ◆ OK

收 ← ◆ NO CARRIER // 对方挂断

## 8. 电话簿相关命令

### 8.1. AT+CPBS 选择电话簿存储单元

该命令可选择电话簿存储单元<storage>，<storage>可用于其它电话簿命令。查询命令用于返回当前选择的存储单元。

Command	Possible Response(s)
+CPBS=<storage>	OK ERROR
+CPBS?	+CPBS:<storage>[, <used>, <total>] OK
+CPBS=?	+CPBS: (<storage>取值列表) OK

### 参数

<storage>

“UD” SIM 卡（或 ME）中的号码（MSISDN）清单（也可以使用 AT+CNUM 命令读出该存储器中的信息）

“AD” SIM 电话簿

<used> 整数型；表示所选存储器中使用过的位置号码个数

<total> 整数型；表示所选存储器中位置号码总数

举例：

发→◇AT+CPBS="AD" //设置当前的电话簿为"AD"

收←◆OK

## 8.2. AT+CPBW 写删电话簿记录

该命令可在当前电话簿存储器的位置号码<index>中写入电话簿记录。通过 AT+CPBS，可选择当前电话簿存储器。写入的记录字段是与号码有关的电话号码<number>（使用<type>格式）和文本<text>。若省略这些字段，则将删除电话簿记录。若<index>为空，但给定<number>，则记录将被写入电话簿中第一个空闲位置。

Command	Possible Response(s)
+CPBW=<index1>[, <number>, [<type>, [<text>]]]	OK ERROR
+CPBW=?	+CPBW: (<index>取值列表), <nlength>, (<type>取值列表), <tlength> OK

### 参数

<index> 电话簿存储器中位置号码范围内的整数型取值

<number> 字符型；<type>格式的电话号码

<type> 整数型的八位字节地址类型（请参考 GSM 04.08[8]第 10.5.4.7 小节）；当拨号字符串包括国际接入代码字符“+”时，缺省值为 145；其他情况下，缺省值为 129

129 未知类型（ISDN 格式）

145 国际号码类型（ISDN 格式）

<text> 字符型；<tlength>最大长度的字符型字段；

<nlength> 表示<number>字段最大长度的整数型取值

<tlength> 表示<text>字段最大长度的整数型取值

举例：

发→◇AT+CPBW=1, "8618323466401", 145, "hywang" //写入记录

收←◆OK

发→◇AT+CPBW=1 //删除位置号为 1 的记录

收←◆OK

发→◇AT+CPBW=?

收←◆+CPBW: (1-4), 20, (128-201), 14

收←◆OK

### 8.3. AT+CPBR 读取电话簿记录

该命令操作可返回位置号码范围为<index1> - <index2>的电话簿记录，该位置号码范围通过使用 AT+CPBS 从当前电话簿存储中选择。若<index2>为空，则只返回<index1>指定的电话簿。

Command	Possible Response(s)
+CPBR=<index1>[, <index2>]	+CPBR:<index1>, <number>, <type>, <text>[<CR><LF>+CPBR:.. ... +CPBR:<index2>, <number>, <type>, <text>] OK ERROR
+CPBR=?	+CPBW: (<index>取值列表), <nlength>, <tlength> OK

#### 参数

<index1> 电话簿存储器中位置号码范围内的整数型取值

<index2> 电话簿存储器中位置号码范围内的整数型取值

<number> 字符型；<type>格式的电话号码

<type> 整数型的八位字节地址类型（请参考 GSM 04.08[8]第 10.5.4.7 小节）

<text> 字符型；<tlength>最大长度的字符型字段；

<nlength> 表示<number>字段最大长度的整数型取值

<tlength> 表示<text>字段最大长度的整数型取值

#### 备注

<index1>应小于<index2>数值

举例：

发→◇AT+CPBR=1

收←◆+CPBR: 1, "8618323466410", 145, "hywang"

收←◆OK

发→◇AT+CPBR=1, 4

收←◆+CPBR: 1, "8618323466410", 145, "hywang"

收←◆+CPBR: 4, "8613971892961", 145, "xixi"

收←◆OK

发→◇AT+CPBR=?

收←◆+CPBR: (1-4), 20, 14

收←◆OK

## 8.4. AT+CPBF 查找电话簿记录

该命令可从当前电话簿存储器中查找电话簿记录。通过 AT+CPBS，可选择当前电话簿存储器。该字母数字混编方式的字段以<findtext>字符串开始。

Command	Possible Response(s)
+CPBF=[<findtext>]	[+CPBF:<index1>, <number>, <type>, <text>[... ]<CR><LF> +CBPF:<index2>, <number>, <type>, <text>] OK ERROR
+CPBF=?	+CPBF:<nlength>, <tlength> OK

### 参数

<findtext> <tlength>最大长度的字符型字段；与“选择 TE 字符集”命令 AT+CSCS 规定的字符集相同

<index1> 电话簿存储器中位置号码范围内的整数型取值

<index2> 电话簿存储器中位置号码范围内的整数型取值

<number> 字符型；<type>格式的电话号码

<type> 整数型的八位字节地址类型（请参考 GSM 04.08[8]第 10.5.4.7 小节）

129 未知类型（ISDN 格式）

145 国际号码类型（ISDN 格式）

<text> 字符型；<tlength>最大长度的字符型字段；

<nlength> 表示<number>字段最大长度的整数型取值

<tlength> 表示<text>字段最大长度的整数型取值

举例：

发→◇AT+CPBF="hywang"

收←◆+CPBF: 1, "8618323466410", 145, "hywang"

收←◆OK

发→◇AT+CPBF=?

收←◆+CPBF: 20, 14

收←◆OK

## 9. TCP/IP 相关命令

### 9.1. AT+IPCREATE 创建 Socket

AT+IPCREATE 支持创建 TCP 和 UDP 客户端。当前只能支持 4 个客户端，不支持服务端。

Command	Possible Response(s)
+IPCREATE=[<index>], [<mode>], [<adress>], [<Port>]	OK <index>, CONNECT OK CONNECT FAIL ERROR

+IPCREATE?	+IPCREATE: <index>, <state>
+IPCREATE=?	+IPCREATE: index, mode (TCP/UDP), ip/address, port

创建 TCP 或者 UDP 链接。如果创建网络链接前 PDP 没有激活，将自动完成 PDP 激活后创建。如果创建 TCP 连接连续 3 次都失败了，将返回“CONNECT FAIL”；TCP 创建成功返回“CONNECT OK”

### 参数

<index>: 创建连接的通道，目前可选 0-3 四个通道

<mode>: 创建 TCP 连接或者 UDP 连接

TCP: 创建 TCP 连接

UDP: 创建 UDP 连接

<adress>: 创建客户端要连接服务端的地址，可以用 IP 地址或者域名，具有将域名转换成 IP 的功能。

<Port>: 服务器的端口号

<state>: 通道状态

举例:

发 → ◇ AT+IPCREATE=?

收 ← ◆ +IPCREATE: index, mode (TCP/UDP), ip/address, port

收 ← ◆ OK

发 → ◇ AT+IPCREATE=0, "TCP", "192.168.1.1", 50

收 ← ◆ CONNECT OK

收 ← ◆ OK

//说明: 返回 CONNECT OK 说明创建 TCP 连接成功。

发 → ◇ AT+IPCREATE?

收 ← ◆ +IPCREATE:

收 ← ◆ 0, TCP

收 ← ◆ 1, NUL

收 ← ◆ 2, NUL

收 ← ◆ 3, NUL

收 ← ◆ OK

## 9.2. AT+IPSEND Socket 发送数据

Command	Possible Response(s)
+IPSEND = [<index>], [<lent>], [<data>]	+IPSEND: lent OK ERROR
+IPSEND =?	+IPSEND :index, lent, data

通过已经建立的 SOCKET 发送数据。

### 参数

<index>: 已创建连接的通道

<lent>: 发送的数据的长度 (若数据中无 '\0', 则长度可写 0, 让其自动计算)

<data>: 发送的数据

<len>: 成功发送数据的长度

举例:

发 → ◇ AT+IPSEND =?

收 ← ◆ +IPCREATE: index, lent, data

收 ← ◆ OK

发 → ◇ AT+IPSEND=0, 0, 123

收 ← ◆ +IPSEND:3

收 ← ◆ OK

## 9.3. AT+IPSENDATA Socket 数据透传

Command	Possible Response(s)
+IPSENDATA = [<index>], [<lent>]	> +IPSENDATA: lent OK ERROR

在已创建的连接上开启数据透传模式。在完成指定的透传长度后或超时后退出。

### 参数

<index>: 透传通道号

<lent>: 需要透传的数据长度

举例:

发 → ◇ AT+IPSENDATA=0, 20

收 ← ◆ > //返回>表示透传开启成功

//输完透传数据后, 按组合键 CTRL+Z>发送 或者 发送十六进制 0x1A 发送, 按键 ESC>退出 或者 发送十六进制 0X1B 退出

发 → ◇ \\r\\n\\0\\\naaaa

收 ← ◆ +IPSENDATA:20 //返回透传完成的数据长度

收 ← ◆ OK

## 9.4. AT+IPCLOSE 关闭 Socket

Command	Possible Response(s)
+IPCLOSE = [<index>]	OK

	ERROR
+IPCLOSE =?	+IPCLOSE :index

关闭已经创建的 Socket. 重启 Socket, 必须已经存在的 Socket 才能重新创建。

### 参数

<index>: 关闭通道号

举例:

发→◇AT+IPCLOSE =?

收←◆+IPCLOSE:index

收←◆OK

发→◇AT+IPCLOSE =0

收←◆OK

## 9.5. TCP 使用示例

举例:

发→◇AT+CFUN=1 //设备上电

收←◆OK

发→◇AT+CGACT=1,1 //PDP 激活

收←◆+CGACT=(0,1)

收←◆OK

发→◇AT+IPCREATE? //查询是否已经存在 SOCK

收←◆+IPCREATE:

0, NUL

1, NUL

2, NUL

3, NUL

收←◆OK

发→◇AT+IPCREATE=0,"TCP","192.168.40.2",7 //创建 TCP

收←◆CONNECT OK

收←◆OK

发→◇AT+IPSEND=0,3,123 //TCP 发送“123”给服务端

收←◆+IPSEND=3

收←◆OK

收←◆+IPRECV=0,TCP,3,123 //TCP 接收上报 TCP 数据

发→◇AT+IPCLOSE=0 //关闭 TCP 连接  
收←◆OK

## 9.6. UDP 使用示例

举例:

发→◇AT+CFUN=1 //设备上电  
收←◆OK

发→◇AT+CGACT=1,1 //PDP 激活  
收←◆+CGACT=(0,1)  
收←◆OK

发→◇AT+IPCREATE=1,"UDP","192.168.40.2",7 //创建 UDP  
收←◆OK

发→◇AT+IPSEND=1,3,123 // 发送“123”给服务端  
收←◆+IPSEND=3  
收←◆OK

收←◆+IPRECV=1,"UDP",3,123 //UDP 接收上报 TCP 数据

发→◇AT+IPCLOSE=1 //关闭 UDP 连接  
收←◆OK

## 10. FTP 相关命令

### 10.1. AT+FTPMODE 设置 FTP 模式

AT+FTPMODE 支持主动模式和被动模式。

Command	Possible Response(s)
+FTPMODE= <value>	OK ERROR
+FTPMODE?	+FTPMODE: <value> OK
+FTPMODE=?	+ FTPMODE: (0-1)

设置 FTP 工作的模式。

#### 参数

< value >: 设置 FTP 模式值  
0:主动模式

## 1: 被动模式

举例:

发→◇AT+FTPMODE=?

收←◆+ FTPMODE: (0-1)

收←◆OK

发→◇AT+ FTPMODE =0

收←◆OK

发→◇AT+ FTPMODE?

收←◆+ FTPMODE: 0

收←◆OK

## 10.2. AT+FTPTYPE 设置 FTP 数据传输类型

Command	Possible Response(s)
+FTPTYPE= <value>	OK ERROR
+FTPTYPE?	+FTPTYPE: <value> OK
+FTPTYPE=?	+ FTPTYPE: (I/A)

FTPTYPE: (I/A)

I:FTP Binary 字符集

A:FTP ASCII 字符集

## 10.3. AT+FTPRESET FTP 断点续传

AT+ FTPRESET 标识出文件内的数据点，将从这个点开始继续传送文件

Command	Possible Response(s)
+FTPRESET=<value>	OK ERROR
+ FTPRESET?	+ FTPRESET OK
+ FTPRESET =?	+ FTPMODE

FTP 复位。

注: RESET 命令只在 TYPE i 模式下有效

举例:

发→◇AT+ FTPRESET =?

收←◆+ FTPRESET:<value>

收←◆OK

发→◇AT+ FTPRESET=5 //创建 FTP 服务

收 ← ◆ OK

发 → ◇ AT+ FTPRESET?

收 ← ◆ + FTPRESET:5

收 ← ◆ OK

## 10.4. AT+FTPSERV 设置 FTP 服务地址

AT+ FTPSERV 设置 FTP server 地址。

Command	Possible Response(s)
+FTPSERV= <ip>,<port>	OK ERROR
+ FTPSERV?	+ FTPSERV: <ip>,<port> OK
+ FTPSERV =?	+ FTPSERV: <ip>,<port>

结果返回码列表

返回码	含义
0	成功
1	pdp 未激活
2	ip 地址有误
3	网络 ip 连接不上
4	ip 连接超时
5	服务器 ftp 拒绝

### 参数

<ip>: 设置 FTP 服务器的 IP 地址

<port>: 端口号

举例:

发 → ◇ AT+ FTPSERV =?

收 ← ◆ + FTPMODE: <IP>,<PORT>

收 ← ◆ OK

发 → ◇ AT+ FTPSERV =129.168.0.1,20

收 ← ◆ + FTPSERV:0

收 ← ◆ OK

发 → ◇ AT+ FTPSERV?

收 ← ◆ + FTPSERV: 129.168.0.1,20

收 ← ◆ OK

## 10.5. AT+FTPUN 设置 FTP 用户名

AT+ FTPUN 设置 FTP 客户端用户名。

Command	Possible Response(s)
+FTPUN= <username>	OK ERROR
+ FTPUN?	+ FTPUN: <username> OK
+ FTPUN=?	+ FTPUN: <username>

设置 FTP 客户端用户名。

### 参数

<username>: 设置登录 FTP 客户端的用户名。执行 FTPUN 时, 登录 FTP 客户端。用户名最大长度为 64 字节。

举例:

发→◇AT+ FTPUN =?

收←◆+ FTPUN: <username>

收←◆OK

发→◇AT+ FTPUN =xxx

收←◆OK

发→◇AT+ FTPUN?

收←◆+ FTPUN: XX

收←◆OK

## 10.6. AT+FTPPW 设置 FTP 密码

AT+ FTTPW 设置 FTP 客户端密码。

Command	Possible Response(s)
+FTPPW= <password>	OK ERROR
+ FTTPW?	+ FTTPW: <username> OK
+ FTTPW=?	+ FTTPW: <username>

设置 FTP 客户端密码。

### 参数

<password>: 设置登录 FTP 客户端的密码, 最长为 64 字节。执行 FTTPW 时, 登录 FTP 客户端用户名已经设置成功了。

举例:

发→◇AT+ FTTPW =?

收←◆+ FTTPW: <password>

收←◆OK

发→◇AT+ FTTPW =xxx

收←◆OK

```
发→◇AT+ FTPPW?
收←◆+ FTPPW: XX
收←◆OK
```

## 10.7. AT+FTPGETNAME 设置 FTP 客户端 get 文件名字

AT+ FTPGETNAME 设置 FTP get 文件名字。

Command	Possible Response(s)
+ FTPGETNAME = <file name>, <max receive time>	OK ERROR
+ FTPGETNAME?	+ FTPGETNAME: < file name > OK
+ FTPGETNAME =?	+ FTPGETNAME: < file name >

设置 FTP 客户端 get 方式获取的文件名字

### 参数

<file name>: 设置 FTP get 方式获取文件的名字。文件名字最长长度为 64 字节。

<max receive time>: 设置最大接收时间, 超时将结束接收, 默认 20s, 0 为永不超时

举例:

```
发→◇AT+ FTPGETNAME =?
收←◆+ FTPGETNAME:XXX
收←◆OK
```

```
发→◇AT+ FTPGETNAME =xxx
收←◆OK
```

```
发→◇AT+ FTPGETNAME?
收←◆+ FTPGETNAME: XX
收←◆OK
```

## 10.8. AT+FTPPATH FTP 切换路径

AT+ FTTPATH 设置 FTP 获取文件路径。

Command	Possible Response(s)
+ FTTPATH= <file path>	OK ERROR
+ FTTPATH?	+ FTTPATH: < file path > OK
+ FTTPATH=?	+ FTTPATH: < file path >

设置 FTP 客户端获取的文件路径

### 参数

<file path>: 设置 FTP 获取文件的路径。文件路径最长长度为 64 字节。

举例：

发→◇AT+ FTTPATH=?

收←◆+ FTTPATH: <file path>

收←◆OK

发→◇AT+ FTTPATH=XXX

收←◆OK

发→◇AT+ FTTPATH? //查询已经设置的 FTP 获取的文件路径

收←◆+ FTTPATH: XX

收←◆OK

## 10.9. AT+FTPPUTNAME 设置 FTP 客户端 put 的文件名字

AT+ FTTPUTNAME 设置 FTP put 文件名字。

Command	Possible Response(s)
+ FTTPUTNAME = <file name>	OK ERROR
+ FTTPUTNAME?	+ FTTPUTNAME: < file name > OK
+ FTTPUTNAME =?	+ FTTPUTNAME: < file name >

设置 FTP 客户端 put 方式获取的文件名字

### 参数

<file name >: 设置 FTP put 方式获取文件的名字。文件名字最长长度为 64 字节。如果在路径名中指定的文件已经在服务站点上存在，则此文件的内容将被传输过来的数据所替代。如果在路径名中指定的文件尚未存在，将创建一个新文件。

举例：

发→◇AT+ FTTPUTNAME =?

收←◆+ FTTPUTNAME:XXX

收←◆OK

发→◇AT+ FTTPUTNAME =XXX

收←◆OK

发→◇AT+ FTTPUTNAME? //查询已经设置的 FTP put 方式获取的文件名字

收←◆+ FTTPUTNAME: XXX

收←◆OK

## 10.10. AT+FTPPUT FTP PUT 文件

AT+FTPPUT put 文件

Command	Possible Response(s)
+FTPPUT=lent,<data>	OK ERROR
+FTPPUT=?	+FTPPUT

FTP PUT 文件内容。PUT 文件完成后，以”AT+FTPPUT=0,”结尾。否则 FTP 一直处于上传状态。

举例：

发→◇AT+FTPPUT=?

收←◆+FTPPUT

收←◆OK

发→◇AT+FTPPUT=2,12 //发送内容为”12”，长度为2

收←◆OK

## 10.11. AT+FTPMKD FTP 创建目录

AT+FTPMKD 创建目录

Command	Possible Response(s)
+FTPMKD = <dir>	OK ERROR
+FTPMKD =?	+FTPMKD: <dir>

FTP 创建目录

**参数**

<dir >: FTP 创建目录, 目录最长长度为 64 字节。

举例：

发→◇AT+FTPMKD =?

收←◆+FTPMKD: < dir >

收←◆OK

发→◇AT+FTPMKD = XXX

收←◆OK

## 10.12. AT+FTPRMD FTP 删除目录

AT+FTPRMD 删除目录

Command	Possible Response(s)
+FTPRMD = <dir>	OK

	ERROR
+ FTPRMD =?	+ FTPRMD: <dir>

FTP 删除目录

### 参数

<dir >: FTP 删除目录, 目录最长长度为 64 字节。

举例:

发→◇AT+ FTPRMD =?

收←◆+ FTPRMD:XXX

收←◆OK

发→◇AT+ FTPRMD =xxx

收←◆OK

## 10.13. AT+FTPGET FTP GET 文件

AT+ FTPGET get 文件

Command	Possible Response(s)
+ FTPGET=<max receive time>	+ FTPGET: lent XXX OK ERROR
+ FTPGET =?	+ FTPGET:<max receive time>

FTP GET 文件。

### 参数

<max receive time>:设置最大接收时间, 超时将结束接收, 0 为永不超时, 该参数可省略, 默认 20s 超时。

举例:

发→◇AT+ FTPGET?

收←◆+ FTPGET:<max receive time>

收←◆OK

发→◇AT+ FTPGET

收←◆OK

## 10.14. AT+FTPDELE FTP 删除文件

AT+ FTPDELE FTP 删除文件

Command	Possible Response(s)
+ FTPDELE =<FileName>	OK ERROR
+ FTPPUTNAME =?	+ FTPPUTNAME =<FileName>

设置 FTP 客户端 put 方式获取的文件名字

### 参数

<file name >: 设置 FTP put 方式获取文件的名字。文件名字最长长度为 64 字节。

如果在路径名中指定的文件已经在服务站点上存在，则此文件的内容将被传输过来的数据所替代。如果在路径名中指定的文件尚未存在，将创建一个新文件。

举例：

发→◇AT+ FTPPUTNAME =?

收←◆+ FTPPUTNAME =<FileName>

收←◆OK

发→◇AT+ FTPPUTNAME =FileName

收←◆OK

### 10.15. AT+FTPLIST FTP 文件目录列表

AT+ FTPLIST 设置 FTP 服务器文件目录列表

Command	Possible Response(s)
+ FTPLIST	+ FTPLIST: mode, lent XXX OK ERROR
+ FTPLIST?	+ FTPLIST

获取 FTP 文件目录列表

举例：

发→◇AT+ FTPLIST

收←◆+FTPLIST:mode, lent

XXX

收←◆OK

发→◇AT+ FTPLIST?

收←◆+ FTPLIST

收←◆OK

### 10.16. AT+FTPSCON 查询 FTP 配置

AT+ FTPSCON 查询 FTP 配置

Command	Possible Response(s)
+ FTPSCON	+ FTPSCON XXXX OK ERROR
+ FTPSCON =?	+ FTPSCON

FTP 查询 FTP 配置

举例：

发→◇AT+ FTPSCON

收←◆+ FTPSCON :

```

XXX
收←◆OK

发→◇AT+ FTPSCON =?
收←◆+ FTPSCON
收←◆OK

```

### 10.17. AT+FTPQUIT 退出 FTP

Command	Possible response(s)
+FTPQUIT	OK ERROR
+FTPQUIT=?	FTPQUIT

```

举例：
发→◇AT+ FTPQUIT
收←◆OK

发→◇AT+ FTPQUIT =?
收←◆+ FTPQUIT
收←◆OK

```

### 10.18. FTP 实例

```

举例：
发→◇AT+ FTPQUIT
收←◆OK

发→◇at+cfun=1
收←◆OK

发→◇AT+CGACT=1, 1
收←◆OK

发→◇AT+FTPSERV=192.168.2.222, 20
收←◆+FTPSERV:0
收←◆OK

发→◇AT+FTPUN=ftp
收←◆OK

```

发→◇AT+FTPPW=ucchip123

收←◆OK

发→◇AT+FTPPATH=/home/uchip/ftp/bin

收←◆+FTPPATH:250

收←◆OK

发→◇AT+FTPTYPE=i

收←◆+FTPTYPE:0

收←◆OK

发→◇AT+FTPLIST

收←◆OK

收←◆+FTPLIST:0,370

-rwxrwxrwx	1	1000	1000	8 May 14 12:04	1.bin
-rw-----	1	128	137	12 Jun 18 19:50	12
-rw-----	1	128	137	24 Jun 18 20:09	123
-rw-----	1	128	137	8 Jun 18 14:53	88
-rw-----	1	128	137	8 May 14 12:19	99
-rwxrwxrwx	1	0	0	5 May 17 17:11	test.txt

发→◇AT+FTPGETNAME=123

收←◆OK

发→◇AT+FTPGET

收←◆OK

收←◆+FTPGET:24,123456781234567812345678

发→◇AT+FTPSIZE=test.txt

收←◆+FTPSIZE:5

收←◆OK

发→◇AT+FTPPUTNAME=1234

收←◆OK

发→◇AT+FTPPUT=5,12345

收←◆OK

发→◇AT+FTPPUT=5,12345

收←◆OK

发→◇AT+FTPPUT=5,12345

收←◆OK

发→◇AT+FTPPUT=5,12345

收←◆OK

发→◇AT+FTPPUT=0,

收←◆OK

发→◇AT+FTPGETNAME=1234

收←◆OK

发→◇AT+FTPGET

收←◆+FTPGET:20,12345123451234512345

收←◆OK

发→◇AT+FTPRESET=3

收←◆OK

发→◇AT+FTPGET

收←◆+FTPGET:17,45123451234512345

收←◆OK

发→◇AT+FTPQUIT

收←◆OK

## 11. HTTP 相关命令

### 11.1. AT+HTTPTLS 设置 HTTPS 的 TLS

Command	Possible response(s)
+HTTPTLS=<t1s>	OK
	ERROR
+HTTPTLS?	+HTTPTLS:0 OK

+HTTPTLS=?	+HTTPTLS: (0-1) OK
------------	-----------------------

HTTPTLS 设置 HTTPS 的 TLS 参数。

### 参数

<tls> tls 使能

0: 非 tls

1: tls

注:在执行 HTTPS 后,硬件若未重新上电或者主动更改 tls 的值来执行 HTTP, tls 保持为 1。

举例:

发→◇AT+HTTPTLS=?

收←◆+HTTPTLS: (0-1)

收←◆OK

发→◇AT+HTTPTLS=1 //使能 HTTPS

收←◆OK

发→◇AT+HTTPTLS? //查询 HTTPS 的 TLS 参数

收←◆+HTTPTLS:1

收←◆OK

## 11.2. AT+HTTTPARA 设 HTTP 参数

Command	Possible response(s)
+HTTTPARA=[<httpparatag>:<httppara value>]	OK ERROR
+ HTTTPARA?	
+ HTTTPARA=?	

该命令用于 HTTP 参数配置<httpparavalue> 参数说明: 根据<httpparatag>参数类型给数据。

### 参数

<httpparatag>参数说明

<URL>: 必选参数, 客户端 URL ( http://xxxx 或者 192.168.1.1 或者 192.168.1.1:80 或者 http://xxxx:80), 最长参数 1024 字节;

<PORT>: HTTP 代理服务器的端口;

<TIMEOUT>:HTTP 超时时间, 默认时间 50S;

<USRE>:HTTP user 数据, 最大长度 512 字节, (数据中“\r\n”需要转义) 同 AT+HTTTPDATA;

<BODY>:HTTP body 参数, 最大长度 512 字节;

<MODE>:HTTP 数据主动读模式 (0) 和数据上报模式 (>0)。

举例：

发→◇AT+HTTTPARA =?

收←◆+HTTTPARA: [<httpparatag>:<httpparavalue>]

收←◆OK

发→◇AT+HTTTPARA =xx, xxx //设置参数

收←◆OK

### 11.3. AT+HTTPDATA HTTP 数据

Command	Possible response(s)
+HTTPDATA=[<lent>,<data>]	OK ERROR
+HTTPDATA=?	+HTTPDATA: <lent>,<data>

HTTP 的数据长度和数据

#### 参数

<lent>: http 数据的长度

<data>:http 数据

举例：

发→◇AT+HTTPDATA=?

收←◆+HTTPDATA: <lent>,<data>

收←◆OK

发→◇AT+HTTPDATA=1, X

收←◆OK

### 11.4. AT+HTTPACTION HTTP 激活模式

Command	Possible response(s)
+HTTPACTION=<mode>	+HTTPACTION=<STATE>,<LENT> OK ERROR
+HTTPACTION=?	+HTTPACTION: (0-3)

HTTP 激活。

#### 参数

<mode> 激活模式；

0: get 方式

1: head 方式

2: put 方式

3: post 方式

<STATE> 返回 HTTP 请求类型;

0: HTTP 响应成功

非 0: 失败状态码

<LENT> 返回 HTTP 后有效数据的长度

举例:

发 → ◇ AT+HTTPACTION=?

收 ← ◆ +HTTPACTION: (0-3)

收 ← ◆ OK

发 → ◇ AT+HTTPACTION=0

收 ← ◆ +HTTPACTION=0, 192

收 ← ◆ OK

### 11.5. AT+HTTPREAD 读取 HTTP 响应数据

Command	Possible response(s)
+HTTPREAD	+HTTPREAD=<lent>, <data> OK ERROR
+HTTPREAD=?	+HTTPREAD

HTTPREAD 读取 http 响应参数。

举例:

发 → ◇ AT+HTTPREAD=?

收 ← ◆ +HTTPREAD

收 ← ◆ OK

发 → ◇ AT+HTTPREAD //HTTP 读取响应数据

收 ← ◆ +HTTPREAD: XXX, XXX

收 ← ◆ OK

### 11.6. AT+HTTPTERM 关闭 HTTP

Command	Possible response(s)
+HTTPTERM	OK ERROR

举例:

发 → ◇ AT+ HTTPTERM

收 ← ◆ OK

## 11.7. HTTP get 流程测试

举例：

发→◇at+httppara=path, /

收←◆OK

发→◇at+httppara=url, http:\\www.baidu.com

收←◆OK

发→◇at+httpaction=0

收←◆+httpaction:0,192

收←◆OK

发→◇at+httpread

收←◆+httpread: xxxx

收←◆OK

## 11.8. HTTP post 流程测试

举例：

发→◇at+cfun=1

收←◆OK

发→◇at+cgact=1,1

收←◆OK

发→◇at+httppara=url, http://httpbin.org/post

收←◆OK

发→◇at+httpinit

收←◆OK

发→◇at+httpaction=3

收←◆+HTTPACTION:0,292

收←◆OK

发→◇at+httpread

收←◆+httpread: xxxx

收←◆OK

## 11.9. HTTP onenet 平台流程测试

举例:

发→◇at+cfun=1

收←◆OK

发→◇at+cgact=1,1

收←◆OK

发→◇at+httppara=url,http://api.heclouds.com/devices/632952758

//设备 ID:632952758

收←◆OK

发→◇at+httpinit

收←◆OK

发→◇at+httpaction=0

收←◆+HTTPACTION:0,292

收←◆OK

发→◇at+httpread

收←◆+httpread: xxxx

收←◆OK

发→◇at+httppara=url,http://api.heclouds.com/devices/632952758/datastreams?

//获取设备数据

收←◆OK

发→◇at+httpaction=0

收←◆+HTTPACTION:0,292

收←◆OK

发→◇at+httpread

收←◆+httpread: xxxx

收←◆OK

发→◇at+httppara=url,http://api.heclouds.com/devices/632952758/

datastreams?datastream\_ids=humi

收←◆OK

```
//获取设备某种类型的数据
发→◇at+httpaction=0
收←◆+HTTPACTION:0,292
收←◆OK

发→◇at+httpread
收←◆+httpread: xxxx
收←◆OK

//ONENET 上传数据
发→◇at+httppara=url,http://api.heclouds.com/devices/632952758/datapoints
收←◆OK

发→◇at+httppara=user,api-key:z40o57s9ctWnrAySMfpIfyq1A5M=
//z40o57s9ctWnrAySMfpIfyq1A5M= 密钥
收←◆OK

发→◇at+httppara=body,{"datastreams":[{"id":"humi","datapoints":
[{"value":20}]}]}
收←◆OK

发→◇at+httppara=user,Content-Type:application/json
收←◆OK

发→◇at+httpinit
收←◆OK

发→◇at+httpaction=3
收←◆+HTTPACTION:0,292
收←◆OK
```

## 11.10. HTTPS 钉钉机器人平台流程测试

```
举例：
发→◇at+httpstls=1
收←◆OK

发→◇at+httppara=url,https://oapi.dingtalk.com/robot/send/?access_token
```

```

=e9fe2495726f5763f5f22306ca6785b7ef76c7b5ae991646c18230278872581b
//PS:?前面需要加/
收←◆OK

发→◇at+httppara=user, Content-Type:application/json
收←◆OK

发→◇at+httpinit
收←◆OK

发→◇at+httppara=body, {"msgtype":"text", "text":{"content":"Happening:
sys123"}}
收←◆OK

收←◆+HTTPACTION:0, 23
收←◆OK
    
```

## 12. MQTT 相关命令

### 12.1. AT+MQTTLS 设置 MQTT 的 TLS 参数

Command	Possible response(s)
+MQTTLS=<tls>,<type>,<cert file content>,<cert file size>	OK ERROR
+MQTTLS?	+MQTTLS:(0-1),<type>,<cert file content>,<cert file size>
+MQTTLS=?	+MQTTLS=<tls>,<type>,<cert file content>,<cert file size>

#### 参数定义

参数	定义	取值	说明
<tls>	tls 加密使能标志	0-1(默认值: 0)	0 tcp 1 tls (MQTT 断开连接后需重新配置)
<type>	证书类型	0-2	0 根证书名 1 客户端证书名 2 客户端私钥名
<cert file content>	(预存的) 证书内容	字符串	最多 1536 字节 需要用" "括住
<cert file size>	(预存的) 指示证书数据长度	length	证书数据长度, 不超过 1536

### 12.2. AT+MQTTCONFIG 设置 MQTT 相关参数

Command	Possible response(s)
+MQTTCONFIG=<url>,<port>,<clientid>,<cleansession>,<keepalive>,<username>,<password>	OK ERROR

+ MQTTCONFIG?	+MQTTCONFIG: <url>, <port>, <clientid>, (0-1), <keepalive>, <username>, <password>
+ MQTTCONFIG=?	+MQTTCONFIG: <url>, <port>, <clientid>, (0-1), <keepalive>, <username>, <password>

### 参数定义

参数	定义	取值	说明
<url>	服务器 IP 地址或者域名	Domain name 或者 XXX.XXX.XXX.XXX	最多 128 字节 XXX 取值: 0-255 需要用 " " 括住
<port>	服务器端口号	1-65535	
<clientid>	客户端标识	字符串	最多 64 字节 需要用 " " 括住
<cleansession>	Clean session	0-1	0 保持原会话 1 新建会话
<keepalive>	发送数据最大间隔时间	0-65535	单位秒, 默认 5S。
<username>	用户名	字符串	最多 64 字节 需要用 " " 括住
<password>	密码	字符串	最多 128 字节 需要用 " " 括住

### 12.3. 客户端连接服务器, AT+MQTTCONNECT

Command	Possible response(s)
+MQTTCONNECT	+MQTTCONNECT: 0 OK ERROR
+ MQTTCONNECT=?	+MQTTCONFIG

MQTT CONNECT 连接超时默认是 5S。

## 12.4. 发布消息, AT+MQTTPUB

Command	Possible response(s)
+MQTTPUB=<topic>,<qos>,<retain><message>	OK ERROR

### 参数定义

参数	定义	取值	说明
<topic>	发布主题	字符串	最多 1024 字节 需要用” ” 括住
<qos>	消息 qos	0-2	0: 最多一次 1: 最少一次 2: 做多一次
<message>	用户消息	字符串	最多 1024 字节 需要用” ” 括住

## 12.5. 订阅主题, AT+MQTTSUB

订阅上报的数据最长 200 字节。数据大于 200 字节，则分段上报。

Command	Possible response(s)
+MQTTSUB=<topic>,<qos>	OK ERROR
+MQTTLS?	+MQTTSUB: <topic>,(0-2) OK

### 参数定义

参数	定义	取值	说明
<topic>	订阅主题	字符串	最多 1024 字节 需要用” ” 括住
<qos>	消息 qos	0-2	0: 最多一次 1: 最少一次 2: 做多一次

## 12.6. 取消订阅主题, AT+MQTTUNSUB

Command	Possible response(s)
++MQTTUNSUB=<topic>	OK ERROR
+MQTTCONFIG?	+MQTTUNSUB: <topic> OK

### 参数定义

参数	定义	取值	说明
<topic>	订阅主题	字符串	最多 1024 字节 需要用” ”括住

## 12.7. 查询连接状态, AT+MQTTSTATE

Command	Possible response(s)
+MQTTSTATE	+MQTTSTATE: <status> OK ERROR

### 参数定义

参数	定义	取值	说明
<status>	状态	0-1	0: 连接服务器失败 1: 连接服务器成功

## 12.8. 客户端断开连接, AT+MQTTDISCONNECT

Command	Possible response(s)
+MQTTDISCONNECT	OK ERROR

## 12.9. 数据上报, MQTTSUB

Command	Possible response(s)
+MQTTSUB	+MQTTSUB:<topic>,<msglen>,<message>

### 参数定义

参数	定义	取值	说明
<topic>	发布主题	字符串	最多 1024 字节 需要用” ”括住
<msglen>	消息长度	1-1024	消息长度
<message>	用户消息	字符串	最多 1024 字节 需要用” ”括住

## 12.10. MQTT 测试

举例：

发→◇AT+MQTTTLS=1,0,"#1",1181

收←◆OK

发→◇AT+MQTTCONFIG="192.168.1.100",18830,"example",1,60,"",""

收←◆OK

发→◇AT+MQTTCONNECT

收←◆OK

发→◇AT+MQTTSTATE

收←◆+MQTTSTATE:1

收←◆OK

发→◇AT+MQTTSUB="topic1",1

收←◆OK

发→◇AT+MQTTPUB="topic1",1,0,"hello"

收←◆OK

收←◆+MQTTSUB:"topic1",5,"hello"

发→◇AT+MQTTDISCONNECT

收←◆OK

### 12.10.1. onenet MQTT 无加密方式

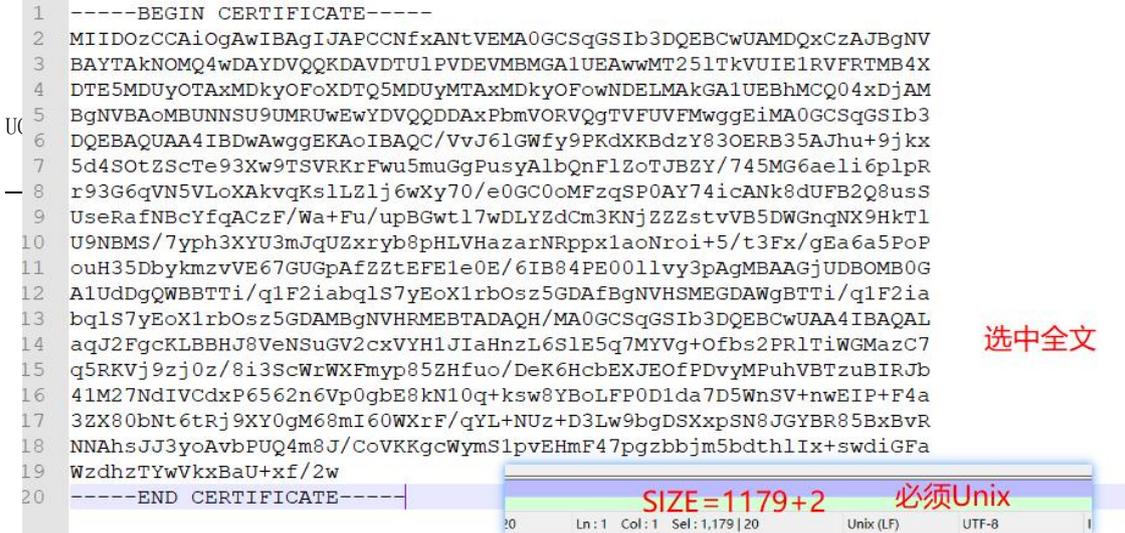
```
AT+MQTTCONFIG="183.230.40.96",1883,"first",1,60,"394157","version=2018-10-31&res=products%2F394157%2Fdevices%2Ffirst&et=1640064378&method=md5&sign=salJMK20Xwj%2BCJGrRkIdtQ%3D%3D"
```

```
AT+MQTTCONNECT
```

```
AT+MQTTSUB="$sys/394157/first/dp/post/json/",1
```

```
AT+MQTTPUB="$sys/394157/first/dp/post/json",1,0,"{\"id\":123,\"dp\":{\"temperature\":[{\"v\":99}],\"power\":[{\"v\":55}]}"
```

```
AT+MQTTDISCONNECT
```



## 12.10.2. onenet MQTTS 加密方式

添加证书:

证书支持: PEM 导入 (需要转换)

步骤 1: 获取数据长度.

用 notepad 打开, 注意选项: 一定为 Unix 模式

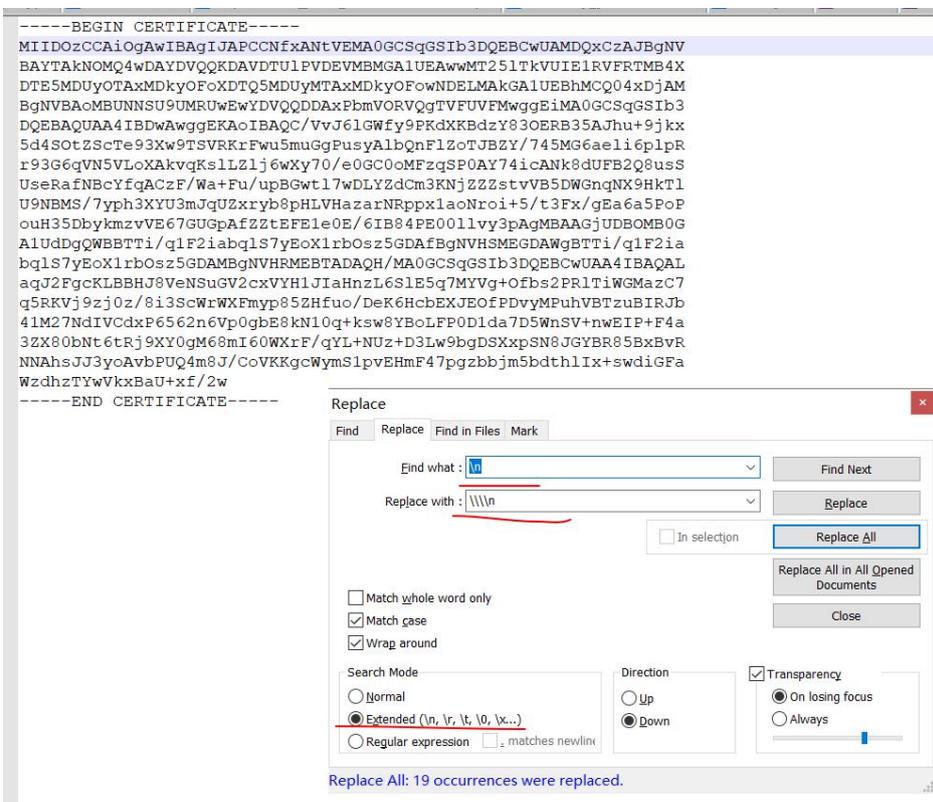
size 大小为 全文内容+2

步骤 2: 获取数据内容.

进行如图方式进行 replace All.

然后替换后内容进行复制到#1.

AT+MQTTLS=1,0,"#1",size



举例:

发 → AT+MQTTLS=1,0,"#1",1181

收 ← ◆ OK

发 → ◇ AT+MQTTCONFIG="183.230.40.16",8883,"first",1,60,"394157","version=2018-10-31&res=products%2F394157%2Fdevices%2Ffirst&et=1640064378&method=md5&sign=salJMK20Xwj%2BCJGrRkIdtQ%3D%3D"

收 ← ◆ OK

发 → ◇ AT+MQTTCONNECT

收 ← ◆ OK

发 → ◇ AT+MQTTSUB="\$sys/394157/first/dp/post/json/",1

收 ← ◆ OK

发 → ◇ AT+MQTTPUB="\$sys/394157/first/dp/post/json",1,0,"{\id\":123,\dp\":{\temperatrue\":[{\v\":99,}],\power\":[{\v\":55,}]}}"

收 ← ◆ OK

发 → ◇ AT+MQTTDISCONNECT

收 ← ◆ OK

### 12.10.3. 阿里云 MQTT

举例：

发 → ◇ AT+MQTTCONFIG="alycvwF89HH.iot-as-mqtt.cn-shanghai.aliyuncs.com",1883,"12345|securemode=3,signmethod=hmacsha1|",1,60,"test&alycvwF89HH","77B62175AB7B5AA8F4128026E1CC36DAFBDFDDCO"

收 ← ◆ OK

发 → ◇ AT+MQTTCONNECT

收 ← ◆ OK

发 → ◇ AT+MQTTSUB="/sys/alycvwF89HH/test/thing/event/property/post\_reply",1

收 ← ◆ OK

发 → ◇ AT+MQTTPUB="/sys/alycvwF89HH/test/thing/event/property/post",1,0,"{\params\":{\TargetTemperature\":72}}"

收 ← ◆ OK

## 13. 短消息相关命令

### 13.1. AT+CMGF 配置短消息模式

Command	Possible Response(s)
+CMGF=[<mode>]	OK ERROR +CME ERROR:<err>
+CMGF?	+CMGF:<mode> OK
+CMGF=?	+CMGF:(list of supported<mode>) OK

该命令用于指定短消息的发送和接收的模式。

#### 参数

<mode> 0 PDU 模式  
1 文本模式

举例：

发 → ◇ AT+CMGF=?

收 ← ◆ +CMGF: (0, 1)

收 ← ◆ OK

发 → ◇ AT+CMGF=1

收 ← ◆ OK

发 → ◇ AT+CMGF?

收 ← ◆ +CMGF: 1

收 ← ◆ OK

### 13.2. AT+CSCA 配置短消息服务中心地址

Command	Possible Response(s)
+CSCA=<sca>[, <tosca>]	OK ERROR +CME ERROR:<err>
+CSCA?	+CSCA:<sca>, <tosca> OK
+CSCA=?	+CSCA:(list of supported<mode>) OK

该命令用于设置 SMSC (Short Message Service Center, 短消息服务中心) 地址。通过该地址, 可以发送 SMS。文本模式下, 发送和设置命令都可以使用该项设置。PDU 模式下, 发送和设置命令也可以使用该项设置, 但条件是编码后的 SMSC 地址长度等于 0, 编码后的 SMSC 地址用参数 <pdu> 表示。

#### 参数

<sca> GSM 04.11 RP SC 使用字符型的地址取值字段; BCD 数字 (或 GSM 缺省字母字符) 需转换为字符; <tosca> 指定的地址类型

<tosca> 服务中心地址格式； GSM 04.11 RP SC 使用整数型的 8 位地址类型（缺省值请参考<toda>）

### 备注

- 输入 SMS 服务中心地址时应该使用服务供应商规定的格式
- 强烈建议使用过程中不要改写 SIM 卡的 SMS 服务中心地址，在发送短消息之前首先必须搞清楚 SIM 卡所属的短消息服务中心地址，否则短消息发送不出去。

举例：

发→◇AT+CSCA=?

收←◆+CSCA: <sca>,<tosca>

收←◆OK

发→◇AT+CSCA="+8613800230500",145 //配置短消息服务中心号码（重庆移动）

收←◆OK

发→◇AT+CSCA? //配置后查询短消息服务中心号码

收←◆+CSCA: "+8613800230500",145

收←◆OK

## 13.3. AT+CMGW 写短消息

Command	Possible Response(s)
文本模式 (+CMGF=1) : +CMGW=<oa/da>[,<tooa/toda>[,<stat>]] <CR> 文本输入<ctrl-Z/ESC> <ESC>退出发送	1) 写短消息成功, 返回: +CMGW: <index> OK
PDU 模式 (+CMGF=0) : +CMGW=<length>[,<stat>]<CR> PDU is given <ctrl-Z/ESC>	2) 功能相关报错: +CMS ERROR: <err>

该命令可将 SMS (SMS-DELIVER 或 SMS-SUBMIT) 从 TE 发送到存储器<mem2>, 并返回已存储短消息的存储位置<index>参数。除非<stat>指定其他参数, 否则该短消息的状态将被设置为一存储未发送。

### 参数

<oa> 字符型的 GSM 03.40 中的 TP-Destination-Address 地址——取值字段；将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符；<toda>给定的地址类型

<da> 字符型的 GSM 03.04 中的 TP-Destination-Address 地址——取值字段；将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符；<toda>给定的地址类型

<tooa> 整数型的 GSM 04.11 TP-Originating-Address 中的 8 位—类型“地址”字段（参考<toda>）

<toda> 整数型的 GSM 04.11 TP-Destination-Address 中的 8 位：“类型—地址”字段（当<da>的首字符为+（IRA 43）时，缺省值为 145，否则缺省值为 129）

129 未知类型 (IDSN 格式号码)

145 国际号码类型 (IDSN 格式号码)

<length> 整数型；文本模式 (AT+CMGF=1) 下，用字符表示<data> (或<cdata>) 消息正文的长度；PDU 模式 (AT+CMGF=0) 下，8 位真实 TP 数据单位的长度 (即 RP 层的 SMSC 地址中的 8 位字符将不计算在该长度内)

<pdu> ISDMS 情况下：GSM 03.40 TPDU。十六进制，遵循 GSM 04.11SC 地址；ME/TA 把 TP 数据单元中每个 8 位字符转换为包含 2 个 IRA 字符的十六进制数字。如：整数取值为 42 的 8 位字符作为 2 位数字 (2A, 即 IRA 50 和 65) 发给 TE) 广播消息情况下：使用十六进制的 GSM 03.41 TPDU

<index> 整数类型；关联存储器支持的地址编号范围内取值

<stat>	PDU 模式/文本模式	说明
0	"REC UNREAD"	已收到但未读的短消息
1	"REC READ"	已收到且已读的短消息
2	"STO UNSENT"	已存储但未发送的短消息
3	"STO SENT"	已存储且已发送的短消息
4	"ALL"	所有短消息

### 备注

●写短消息详细操作请参考 GSM 模块短信应用指导文档

**注 1:** 使用+CMGW 存储前要先执行+CFUN (读 SIM) 开机、+CMGF 指定模式。若为 TEXT 模式，则使用+CSMP 设置模式参数 (不设，则为缺省参数)。

**注 2:** 长短信暂未支持，即 7bit 下可最多存储 160 字符，8bit 下最多存储 140 字符，16bit 下最多存储 70 字符。

举例：

1) 文本模式 (+CMGF=1)

发→◇AT+CMGW="15723244049"

收←◆◇

发→◇This is a test from ucchip // 输入短消息内容，<CTRL+Z>写入，<ESC>退出

收←◆+CMGW: 0

收←◆OK

2) PDU 模式 (+CMGF=0)

发→◇AT+CMGW=20

收←◆◇

//给号码 15723244049 发送内容为 "123456" 的 PDU 数据

发→◇049168000001000D91685127234440F900000631D98C56B301

收←◆+CMGW: 1

收←◆OK

测试使用 PDU 编码生成工具：<http://www.sendsms.cn/pdu/>

### 13.4. AT+CMSS 从存储器发送短消息

Command	Possible Response(s)
+CMSS=<index>[,<da>[,<todo>]]	1) 文本模式 (+CMGF=1) 且执行成功: +CMSS: <mr> OK 2) PDU 模式 (+CMGF=0) 且执行成功: +CMSS: <mr> OK 功能相关报错 +CMS ERROR: <err>

该命令可将短消息存储器 <mem2> 中，位置取值参数为 <index> 的消息发送到网络侧 (SMS-SUBMIT 或 SMS-COMMAND)。若给定 SMS-SUBMIT 短消息的新接收地址参数 <da>，应使用该参数，而不能使用已存储短消息的参数。发送成功后，参考值 <mr> 将返回给 TE。在接收到非请求发送状态报告结果码时，可使用该命令的取值进行短消息识别。

#### 参数

<index> 整数类型；关联存储器支持的地址编号范围内取值

<da> 字符型的 GSM 03.04 中的 TP-Destination-Address 地址——取值字段；将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符；<todo> 给定的地址类型

<todo> 整数型的 GSM 04.11 TP-Destination-Address 中的 8 位：“类型—地址”字段（当 <da> 的首字符为+（IRA 43）时，缺省值为 145，否则缺省值为 129）

129 未知类型（IDSN 格式号码）

145 国际号码类型（IDSN 格式号码）

<mr> 整数型的 GSM 03.40 TP-Message-Reference

举例：

发 → ◇ AT+CMSS=3

收 ← ◆ +CMSS: 2

收 ← ◆ OK

### 13.5. AT+CMGD 删除短消息

Command	Possible Response(s)
+CMGD=<index>[,<delflag>]	OK ERROR +CME ERROR:<err>
+CMGD=?	+CMGD:(list of supported<index>) OK

该命令可删除优选短消息存储器 <mem1> 中，位置号码参数为 <index> 的消息。

#### 参数

<index> 整数型；关联存储器支持的地址编号范围内的取值

<delflag> 0 删除 <index> 指定的短信

1 全部删除存储器中的已读短信

2 全部删除存储器中的已读和已发送短信

3 全部删除存储器中的已读、已发送和未发送短信

4 全部删除存储器中的已读、未读、已发送和未发送短信

## 备注

- 如果当前存储器中没有短信，操作 AT+CMGD 删除动作，仍返回 OK

举例：

发→◇AT+CMGD=?

收←◆+CMGD: <index>

收←◆OK

发→◇AT+CMGD=3 //删除下标为 3 存储区的短信

收←◆OK

## 13.6. AT+CMGL 列出存储器中的短消息

Command	Possible Response(s)
+CMGL=<stat>	1) 文本模式 (AT+CMGF=1) 且该命令执行成功对于 SMS-SUBMIT 和 /或 SMS-DELIVER: +CMGL: <index>, <stat>, <oa/da>, [<scts>], [<tooa/toda>], <length><CR> <LF><data><CR><LF> +CMGL: <index>, <stat>, <da/oa>[, <scts>][, <tooa/toda>], <length><CR> <LF><data>[... ]]  2) PDU 模式 (AT+CMGF=0) 且该命令执行成功 +CMGL:<index>, <stat>, <length><CR><LF><pdu><CR><LF> +CMGL: <index>, <stat>, <length><CR><LF><pdu> [... ]] OK 功能相关错误, 返回 +CMS ERROR: <err>
+CMGL=?	+CMGL:(list of supported<stat>) OK

该命令可查询优选短消息存储器<mem1>中，状态值为<stat>的消息显示在 TE 中。

### 参数

<stat>

为了使用方便，PDU/TEXT 表示共用

- |   |             |
|---|-------------|
| 0 | 已接收但未读的短消息  |
| 1 | 已接收且已读的短消息  |
| 2 | 已存储但未发送的短消息 |
| 3 | 已存储且已发送的短消息 |
| 4 | 所有短消息       |

<da> 字符型的 GSM 03.04 中的 TP-Destination-Address 地址——取值字段；将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符；<toda>给定的地址类型。

<data> 短消息情况下：GSM 03.40 TP-User-Data 在文本模式下返回，格式定义如下：

- 若<dc>指定用 GSM 03.38 缺省字母格式字符，且<fo>指定 GSM 03.40，则没有设置 TPUser-Data-Header-Indication。

- 若 TE 字符集不为十六进制数据（参考 TS 07.07 中的 AT+CSCS 命令选择的 TE 字符集）；则根据 Annex A 规则，ME/TA 将 GSM 字母格式字符转换为当前 TE 的字符集。

- 若 TE 字符集为十六进制数据，则 ME/TA 将 GSM 字母格式字符的每个 7 位字符转换为两个 IRA 字符表示的十六进制数（如字符 P（GSM 32）作为 17（IRA 49 和 55））

- 若 <dc> 指定用 8 位或 UCS2 编码方案，或 <fo> 指定 GSM03.40；则设置 TPUser-Data-Header-Indication: ME/TA 将 GSM 字母格式字符的每个 8 位字符转换为用两个 IRA 表示的十六进制数（如：8 位字符代表的整数 42 传给 TE 时用两个字符来表示 2A（IRA 50 和 65））小区广播情况下：GSM 03.41CBM 消息内容，在文本模式下返回，格式定义如下：

- 若 <dc> 指定用 GSM 03.38 缺省字母格式字符，则 ME/TA 将 GSM 转换成当前 TE 支持的字符。

- 若 TE 字符集不为十六进制数（参考 TS 07.07 中的 AT+CSCS 命令选择的字符集）；则根据 Annex A 规则，ME/TA 将 GSM 字母格式字符转换为当前 TE 的字符集。

- 若 TE 字符集为十六进制数，则 ME/TA 将 GSM 字母格式字符的每个 7 位总字符自己转换为两个 IRA 字符表示的十六进制数。

- 若 <dc> 指定用 8 位或 UCS2 编码方案，或 <fo> 指定 GSM 03.40；则设置 TPUser-Data-Header-Indication: ME/TA 将 GSM 字母格式字符的每个 8 位字符转换为用两个 IRA 表示的 16 进制数。

<length> 整数类型；文本模式（AT+CMGF=1）下，用字符表示 <data>（或 <cdata>）消息正文的长度；PDU 模式（AT+CMGF=0）下，8 位真实 TP 数据单位的长度（即 RP 层的 SMSC 地址中的 8 位字符将不计算在该长度内）

<index> 整数类型；在相应存储器支持的短消息容量内的值

<oa> 字符型的 GSM 03.40 中的 TP-Destination-Address 地址——取值字段；将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符；<toa> 给定的地址类型

<pdu> ISDMS 情况下：GSM 03.40 TPDU。十六进制，遵循 GSM 04.11SC 地址；ME/TA 把 TP 数据单元中每个 8 位字符转换为包含 2 个 IRA 字符的十六进制数字（如：整数取值为 42 的 8 位字符作为 2 位数字（2A，即 IRA 50 和 65）发送给 TE）广播消息情况下：使用十六进制的 GSM 03.41 TPDU

<scts> 使用“时间—字符串”格式的 GSM 03.40 TP-Service-Center-Time-Stamp（参考 <dt>）

<toa> 整数型的 GSM 04.11 TP-Destination-Address 中的 8 位：“类型—地址”字段当 <da> 的首字符为+（IRA 43）时，缺省值为 145，否则缺省值为 129）

<tooa> 整数型的 GSM 04.11 TP-Originating-Address 中的 8 位“类型—地址”字段（参考 <toa>）

举例：

```

发→◇AT+CMGL=?
收←◆+CMGL: (0-4)
收←◆OK

//读命令
1) 文本模式 (+CMGF=1)
发→◇AT+CMGL=4
收←◆+CMGL: 1, 2, "862856044778640", 52
收←◆049168000011000F9168826540748746F00000AA0531D98C5603
收←◆+CMGL: 2, 2, "862856044778640", 5
收←◆12345
收←◆OK

2) PDU 模式 (+CMGF=0)
发→◇AT+CMGL=4
收←◆+CMGL: 2, 2, 22, 0491680000110C0F9168826540748746F00000AA0531D98C5603
收←◆+CMGL: 4, 2, 22, 0491680000110E0F9168826540748746F00000AA0531D98C5603
收←◆+CMGL: 3, 2, 63, 0491680000110F0F9168826540748746F000000034305A2E66C3C1
6030582C0683C18CB9980D8793D96A34D88D86BBD16C46180C06830583B0DA2C46CCE1863
51B6C06
收←◆OK
    
```

### 13.7. AT+CMGR 读取短消息

Command	Possible Response(s)
+CMGR=<index>	1) 文本模式 (AT+CMGF=1) 且该命令执行成功: 对于 SMS-DELIVER: +CMGR: <stat>, <oa>, <scts>[, <tooa>, <fo>, <pid>, <dcsc>, <sca>, <tosca>] , <length><CR><LF><data> 对于 SMS-SUBMIT: +CMGR: <stat>, <da>[, <toda>, <fo>, <pid>, <dcsc>, [ <vp> ], <sca>, <tosca>] , <length><CR><LF><data>  2) PDU 模式 (AT+CMGF=0) 且该命令执行成功: +CMGR: <stat>, <length><CR><LF><pdu> OK 功能相关报错 +CMS ERROR: <err>
+CMGR=?	+CMGR: (list of supported<index>) OK

该命令可将短消息存储器<mem1>中，索引号为<index>的短消息返回到 TE。

#### 参数

<index> 整数类型；关联存储器支持的地址编号范围内取值

<da> 字符型的 GSM 03.04 中的 TP-Destination-Address 地址——取值字段；  
将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符；

<tda>给定的地址类型

<data> 短消息情况下：GSM 03.40 TP-User-Data 在文本模式下返回，格式定义如下：

- 若<dcs>指定用 GSM 03.38 缺省字母格式字符，且<fo>指定 GSM 03.40，则没有设置 TPUser-Data-Header-Indication

- 若 TE 字符集不为十六进制数据（参考 TS 07.07 中的 AT+CSCS 命令选择的 TE 字符集）；则根据 Annex A 规则，ME/TA 将 GSM 字母格式字符转换为当前 TE 的字符集

- 若 TE 字符集为十六进制数据，则 ME/TA 将 GSM 字母格式字符的每个 7 位字符转换为两个 IRA 字符表示的十六进制书（如字符 P（GSM 32）作为 17（IRA 49 和 55））

- 若<dcs>指定用 8 位或 UCS2 编码方案，或<fo>指定 GSM03.40；则设置 TPUser-Data-Header-Indication：ME/TA 将 GSM 字母格式字符的每个 8 位字符转换为用两个 IRA 表示的十六进制数（如：8 位字符代表的整数 42 传给 TE 时用两个字符来表示 2A（IRA 50 和 65））  
小区广播情况下：GSM 03.41 CBM 消息内容，在文本模式下返回，格式定义如下：

- 若<dcs>指定用 GSM 03.38 缺省字母格式字符，则 ME/TA 将 GSM 转换成当前 TE 支持的字符

- 若 TE 字符集不为十六进制数（参考 TS 07.07 中的 AT+CSCS 命令选择的 TE 字符集）；则根据 Annex A 规则，ME/TA 将 GSM 字母格式字符转换为当前 TE 的字符集

- 若 TE 字符集为十六进制数，则 ME/TA 将 GSM 字母格式字符的每个 7 位总字符自己转换为两个 IRA 字符表示的十六进制数

- 若<dcs>指定用 8 位或 UCS2 编码方案，或<fo>指定 GSM 03.40；则设置 TPUser-Data-Header-Indication：ME/TA 将 GSM 字母格式字符的每个 8 位字符转换为用两个 IRA 表示的 16 进制数

<dcs> 取决于该命令或该命令结果码 GSM 03.38 中的 SMS 数据编码方案。（缺省值 0）或者小区广播数据编码方案

<fo> 取决于该命令结果码 GSM 03.40 SMS-DELIVER，SMS-SUBMIT（缺省值 17），SMS-STATUS-REPORT 或采用整数型 SMS-COMMAND（缺省值 2）

<length> 整数型；文本模式（AT+CMGF=1）下，用字符表示<data>（或<ldata>）消息正文的长度；PDU 模式（AT+CMGF=0）下，8 位真实 TP 数据单位的长度（即 RP 层的 SMSC 地址中的 8 位字符将不计算在该长度内）

<mid> 整数型的 GSM 03.41 CBM 消息标识

<oa> 字符型的 GSM 03.40 中的 TP-Destination-Address 地址——取值字段；  
将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符（参

考 TA 07.07 中的 AT+CSCS 命令)；<toda>给定的地址类型

<pdu> ISDMS 情况下：GSM 03.40 TPDU。十六进制，遵循 GSM 04.11SC 地址；ME/TA 把 TP 数据单元中每个 8 位字符转换为包含 2 个 IRA 字符的十六进制数字(如：整数取值为 42 的 8 位字符作为 2 位数字 (2A, 即 IRA 50 和 65) 发送给 TE) 广播消息情况下：使用十六进制的 GSM 03.41 TPDU

<pid> 参考 GSM03.40；TP-协议-标识 (缺省值 0)

<sca> GSM 04.11 中的 RP SC 地址是字符串型参数；将 BCD 数值 (或缺省 GSM 默认字母的字符) 转换成 TE 当前选择字符集中的字符；<tosca>用来指定地址类型。

<scts> 使用“时间-字符串”格式的 GSM03.40 TP-Service-Center-Time-Stamp (参考<dt>)

<stat> PDU 模式/TEXT 模式

- |   |             |
|---|-------------|
| 0 | 已收到但未读的短消息  |
| 1 | 已收到且已读的短消息  |
| 2 | 已存储但未发送的短消息 |
| 3 | 已存储且已发送的短消息 |
| 4 | 所有短消息       |

<toda> 整数型的 GSM 04.11 TP-Destination-Address 中的 8 位：“类型-地址”字段 (当<da>的首字符为+ (IRA 43) 时，缺省值为 145，否则缺省值为 129)

<toa> 整数型的 GSM 04.11 TP-Originating-Address 中的 8 位“类型-地址”字段 (参考 <toda>)

<tosca> 服务中心地址格式；GSM 04.11 RP SC 使用整数型的 8 位地址类型 (缺省值请参考<toda>)

<vp> 取决于 SMS-SUBMIT 的<fo>的设置；采用整数型 (缺省值 167)，或时间-字符串型 (参考<dt>) 的 GSM 03.40 TP-有效期

### 备注

- 如果当前存储器中没有短信，操作 AT+CMGR 读取短消息动作，仍返回 OK

举例：

发→◇AT+CMGR=?

收←◆+CMGR: <index>

收←◆OK

//读命令

1) 文本模式 (+CMGF=1)

发→◇AT+CMGR=4

收←◆+CMGR: 2, "862856044778640", 145, 17, 0, 0, 170, "860000", 145, 5

收←◆12345

收←◆OK

```

2) PDU 模式 (+CMGF=0)
发→◇AT+CMGR=4
收←◆+CMGR: 2, 22, 0491680000110E0F9168826540748746F00000AA0531D98C5603
收←◆OK
    
```

### 13.8. AT+CMGS 发送短消息

Command	Possible Response(s)
1) 文本模式 (+CMGF=1) : +CMGS=<da>[,<toda>]<CR>text is entered <ctrl-Z/ESC> ESC 退出发送 2) PDU 模式 (+CMGF=0) : +CMGS=<length><CR> PDU is given <ctrl-Z/ESC>	1) 文本模式 (+CMGF=1) 且发送成功 +CMGS:<mr> OK 2) PDU 模式 (+CMGF=0) 且发送成功 +CMGS:<mr> OK 3) 功能相关报错 +CMS ERROR:<err>

该命令可将 SMS (SMS-SUBMIT) 从 TE 发送到网络层。发送成功后，短消息参考值<mr>将返回给 TE。在接收到非请求发送状态报告结果码时，使用该取值可进行短消息识别。

#### 参数

<da> 字符型的 GSM 03.04 中的 TP-Destination-Address 地址——取值字段；将 BCD 数值（或缺省 GSM 字母格式的字符）转换为当前选择的 TE 字符集中的字符；<toda> 给定的地址类型

<toda> 整数型的 GSM 04.11 TP-Destination-Address 中的 8 位：“类型—地址”字段（当<da>的首字符为+（IRA 43）时，缺省值为 145，否则缺省值为 129）

<length> 整数型；文本模式（AT+CMGF=1）下，用字符表示<data>（或<cdata>）消息正文的长度，文本模式最大长度为 612 个字符；PDU 模式（AT+CMGF=0）下，8 位真实 TP 数据单位的长度（即 RP 层的 SMSC 地址中的 8 位字符将不计算在该长度内）

<mr> 整数型的 GSM 03.40 TP-Message-Reference

#### 备注

- 发送短消息详细操作请参考 GSM 模块短信应用指导文档
- 支持 TEXT 模式下长短信发送，PDU 模式不支持。TEXT 模式下，最长可发送 1280 个字符（大于此长度时，将发送失败）。

举例：

```

1) 文本模式 (+CMGF=1)
发→◇AT+CMGS="15723244049"
收←◆>
    
```

//输完短信内容后,按组合键 CTRL+Z>发送 或者 发送十六进制 0x1A 发送,按键 ESC>退出 或者 发送十六进制 0x1B 退出

```

发→◇hello world
收←◆+CMGS: 10
收←◆OK
    
```

```

1) PDU 模式 (+CMGF=0)
    
```

```

发→◇AT+CMGS=20
收←◆>
//输完短信内容后,按组合键 CTRL+Z>发送 或者 发送十六进制 0x1A 发送,按键 ESC>
退出 或者 发送十六进制 0x1B 退出
收←◆049168000001000D91685127234440F900000631D98C56B301
收←◆+CMGS: 11
收←◆OK

```

### 13.9. AT+CAMS 配置下行短信确认方式

Command	Possible Response(s)
+CAMS=[<mode>]	OK ERROR +CME ERROR:<err>
+CAMS?	+CAMS:<mode> OK
+CAMS=?	+CAMS:(list of supported<mode>) OK

该命令用于指定接收短信后的确认方式。

#### 参数

- <mode> 0 自动确认模式（默认）  
1 手动确认模式

举例：

```

发→◇AT+CAMS=?
收←◆+CAMS: (0, 1)
收←◆OK

```

```

发→◇AT+CAMS=1
收←◆OK

```

```

发→◇AT+CAMS?
收←◆+CAMS: 1
收←◆OK

```

### 13.10. AT+CNMI 新短消息指示

Command	Possible Response(s)
+CNMI=[<mt>]	OK ERROR +CME ERROR:<err>
+CNMI?	+CNMI:<mt> OK
+CNMI=?	+CNMI:(list of supported<mt>) OK
参考 GSM 07.05	相关 URC 上报: 1) 新短消息到达, 上报提示: +CMTI: <mem>, <index> 2) 新短消息到达, 直接显示短消息内容: +CMT: <length><CR><LF><pdu>

## 参数

<mt> 存储接收短消息的规则取决于数据编码方案（请参考 GSM 03.38[2]）；优选短消息存储器命令（AT+CPMS）的设置和本取值

- 0 没有 SMS-DELIVER 的指示发给 TE。
- 1 若 SMS-DELIVER 存储在 ME/TA，存储位置靠非请求结果码+CMTI：  
     <mem>,<index>来提示给 TE。
- 2 SMS-DELIVERs（除 Class2 短消息）直接发送给 TE,使用如下非请求结果码：  
     +CMT: <length><CR><LF><pdu> （启用 PDU 模式）或者+CMT:  
     <oa>,<scts> [,<toa>,<fo>,<pid>,<dcs>,<sca>,<tosca>,<length>]  
     <CR><LF><data> （启用文本模式）；以上斜体部分参数定义参见命令 AT+CSDH.Class2 短消息指示方式同<mt>=1

举例：

发→◇AT+CNMI=?

收←◆+CNMI: (0-2)

收←◆OK

发→◇AT+CNMI=2

收←◆OK

发→◇AT+CNMI?

收←◆+CNMI: 2

收←◆OK

## 示例

AT+CMGF=1 //设置文本短信模式

OK

AT+CNMI=1 //设置将短信存储到 ME 或 SIM 卡后，再给出新短信指示

OK

+CMTI: "ME",5 //新短信指示

AT+CNMI=2 //设置收到新短信，直接打印短信内容

OK

+CMT: "8615723244049","22/10/19,11:18:19+032",161,4,0,0,"0000",161,11

"hello world" //收到新短信，直接打印短信内容

### 13.11. AT+CNMA 新短消息确认

Command	Possible Response(s)
+CNMA[=<n>	OK ERROR +CME ERROR:<err>

使用执行命令，可确认是否正确接收新消息(SMS-DELIVER)，该新消息是由 MT 直接发送到 TE 而不缓存。在同时满足如下条件的情况下，需要通过 AT+CNMA 给出短信确认：

→通过 AT+CAMS=1，将<mode>设为 1；

#### 参数

<n> 0 该指令与文本模式定义的指令执行类似

1 发送 RP-ACK(或正确接收到的缓冲结果码)

2 发送 RP-ERROR

举例：

发→◇AT+CNMA=1 //下行短信进行确认

收←◆OK

### 13.12. AT+CSDH 显示文本格式参数

Command	Possible Response(s)
+CSDH=[<show>]	OK ERROR +CME ERROR:<err>
+CSDH?	+CSDH:<show> OK
+CSDH=?	+CSDH:(list of supported<show>) OK

该命令可用于控制是否在文本模式下的结果码中显示详细的头信息。

#### 参数

<show> 0 不在+CMT， +CMGL， +CMGR 命令中显示<sca>、 <tosca>、 <fo>、 <vp>、 <pid>和<dcs>（+CSCA 和+CSMP 命令中设置的取值），也不显示用于 SMS-DELIVER 和 SMS-SUBMIT 消息的结果码中的<length>、 <toda>或<tooa>；

1 在结果码中显示这些取值

举例：

发→◇AT+CSDH=?

收←◆+CSDH: (0, 1)

收←◆OK

发→◇AT+CSDH=1

收←◆OK

发→◇AT+CSDH?

收←◆+CSDH: 1

收←◆OK

### 示例

发→◇AT+CSDH=0

收←◆OK

发→◇AT+CMGR=3

收←◆+CMGR: 1, "862856044778640", 11

收←◆hello world

收←◆OK

发→◇AT+CSDH=1

收←◆OK

发→◇AT+CMGR=3

收←◆+CMGR: 1, "862856044778640", 161, 4, 0, 0, "0000", 161, 11

收←◆hello world

收←◆OK

### 13.13. AT+CSMP 设置文本格式参数

Command	Possible Response(s)
+CSMP=[<fo>[<vp>[, <pid>[, <dc>]]]]	OK ERROR +CME ERROR:<err>
+CSMP?	+CSMP:<fo>, <vp>, <pid>, <dc> OK
+CSMP=?	+CSMP:(list of supported<fo>), (list of supported<vp>), (list of supported<pid>), (list of supported<dc>) OK

文本模式下，当向网络侧发送短消息或将短消息存放在存储器中时，使用设置命令可选择需要附加参数取值。除此之外，还可以用于设置从 SMSC 接收到该短消息时算起的有效期（<vp>的取值范围：0~255）或定义有效期终止的绝对时间（<vp>为字符串时）。<vp>的格式由<fo>指定。若 TA 支持增强型有效期格式（\$(EVPF)\$，请参考 GSM 03.40），应该把 16 进制的编码字符串放于双引号中（请参考<pdu>）。

#### 参数

<fo> 取决于该命令或结果码； GSM 03.40 SMS-DELIVER， SMS-SUBMIT（缺省值：17）， SMS-STATUS-REPORT 的前 8 位；或采用整数型 SMS-COMMAND（缺省值：2）；在文本模式下若<fo>值设为 49，则支持消息状态报告。

b7	b6	b5	b4	b3	b2	b1	b0
RP	UDHI	SRR	VPF		RD	MTI	
0	0	0	1	0	0	0	1

(1) 应答路径—TP-RP (TP-Reply-Path) :

0—不设置

1—设置

(2) 用户数据头标识—TP-UDHL (TP-User-Data-Header-Indicator) :

0—不含任何头信息

1—含头信息

(3) 状态报告要求—TP-SPR (TP-Status-Report-Request) :

0—需要报告

1—不需要报告

(4) 有效期格式—TP-VPF (TP-Validity-Period-Format) :

00—不提供 (Not present) ;

10—整型 (标准) ;

01—预留;

11—提供 8 位字节的一半 (Semi-Octet Represented)

(5) 拒绝复制—TP-RD (TP-Reject-Duplicates) :

0—接受复制

1—拒绝复制

(6) 信息类型提示—TP-MTI (TP-Message-Type-Indicator) :

00—读出 (Deliver)

01—提交 (Submit)

<vp> 取决于 SMS-SUBMIT 的<fo>的设置; 采用整数型 (缺省值: 167), 或时间—字符型 (参考<dt>) 的 GSM 03.40 TP-有效期

<pid> 整数型; 参考 GSM 03.40 TP-协议-标识 (缺省值: 0)

<dcs> 整数型; 取决于该命令或该命令结果码 GSM 03.38 中的 SMS 数据编码方案。

0—默认的字母表 (7bit 编码)

4—8bit

8—USC2 (16bit) 编码

举例:

发→◇AT+CSMP=?

收←◆+CSMP: <fo>, <vp>, <pid>, <dcs>

收←◆OK

发→◇AT+CSMP=17, 0, 0, 0

收←◆OK

```
发-><◇AT+CSMP?
收<-◆+CSMP: 17,0,0,0
收<-◆OK
```

### 13.14. +CMTI/+CMT 指示新的短消息

Command	Possible Response(s)
	+CMTI:<mem>,<index> OK
	+CMT: <length><CR><LF><pdu> (启用 PDU 模式) +CMT:<oa>,<scts>[,<toa>,<fo>,<pid>,<dcs>,<sca>,<tosca>,<length>]<CR><LF><data> (启用文本模式) OK

当收到新短信，发送+ CMTI 或+ CMT [ +CD 的消息报告]

#### 参数

<mem>字符串类型存储器存储新消息

<index>整数类型价值定位数范围由关联的内存支持

<length>整数类型；值指示在文本模式 (+ CMGF = 1) 的消息体<数据>长度 (或 <cdata>) 字符；或在 PDU 模式 (+ CMGF = 0)，在字节实际 TP 数据单元的长度 (即 RP 层 SMSC 地址字节不计入长度)

<fo> 根据命令或结果代码：3G TS 23.040 [ 3 ] SMS-DELIVER 第一个八位组，SMS-SUBMIT(默认为17)， sms-status-report， 或 sms-command (默认为2) 在整数格式

<VP>取决于 sms-submit 支持，增强的格式 (十六进制编码的字符串用双引号)

< pid > 3G TS 23.040 [ 3 ] TP 协议整型格式标识符 (默认为0)

< dcs >根据命令或结果代码：3G TS 23.038 [ 2 ]短信数据编码方案 (default0)，或小区广播数据在整数格式的编码方案

< sca > 3G TS 24.011 [ 6 ] RP SC 地址字符串格式的的地址值字段；

<tosca> 3G TS 24.011 [ 6 ] RP SC 地址格式地址字节整数类型

< scts> 3G TS 23.040 [ 3 ] TP 服务中心的时间戳字符串格式的时间 (参阅<dt>) 对应的条目机器翻译电话簿找到；实此功能在具体制造商的；使用的字符集应该是一个选择命令选择的技术特点。

#### 举例

```
发-><◇AT+CNMI=1
收<-◆OK
```

```
发-><◇AT+CMGF=0 //设置短信为PDU模式
收<-◆OK
```

```
发-><◇AT+CNMI=0,2,0,0,0
```

```

收←◆OK

收←◆+CMT: 31,03A10000040FA168826540748746F00000220191113261020BE8329BFD
06DDDF723619

发→◇AT+CMGF=1 //设置短信为文本模式
收←◆OK

发→◇AT+CSDH=1
收←◆OK

发→◇AT+CNMI=2
收←◆OK
收←◆+CMT: "862856044778640", "22/10/19, 11:18:19+032", 161, 4, 0, 0, "0000"
, 161, 11
收←◆"hello world"

```

## 13.15. 短信测试示例

### 13.15.1. 短信直发 (+CMGS)

#### 13.15.1.1. TEXT 模式 收发短信

```

发→◇at+cfun=1 //上电
收←◆OK

发→◇at+cgclass=CC //CS域附着
收←◆OK

发→◇at+cmgf=1 //设置为TEXT模式
收←◆OK

发→◇at+csmf= 17,0,0,0 //设置TEXT模式参数
收←◆OK

发→◇at+cmgs="+8615723244049" //发送短信
收←◆>
发→◇hello world
收←◆+CMGS: 10
收←◆OK

```

```
发→◇at+cnmi=2 //设置TE显示
```

```
收←◆OK
```

```
收←◆+CMT: "8615723244049",11 //接收到短信内容
```

```
"hello world"
```

### 13.15.1.2. PDU 模式 收发短信

```
发→◇at+cfun=1 //上电
```

```
收←◆OK
```

```
发→◇at+cgclass=CC //CS域附着
```

```
收←◆OK
```

```
发→◇at+cmgf=0//设置为PDU模式
```

```
收←◆OK
```

```
发→◇at+cmgs=24 //PDU模式下发送
```

```
收←◆>
```

```
发→◇04916800001000D91685127234440F900000BE8329BFD06DDDF723619
```

```
收←◆+CMGS: 13
```

```
收←◆OK
```

```
发→◇at+cnmi=2 //设置TE显示
```

```
收←◆OK
```

```
//收到短信内容
```

```
收←◆+CMT: 26, 03A10000040FA168826540748746F000002201911184850205F072785C06
```

### 13.15.2. 短信存储发送 (+CMGW +CMSS)

#### 13.15.2.1. TEXT 模式 存储+发送

```
发→◇at+cfun=1 //上电
```

```
收←◆OK
```

```
发→◇at+cgclass=CC //CS域附着
```

```
收←◆OK
```

```
发→◇at+cmgf=1 //设置为TEXT模式
收←◆OK

发→◇at+csmf= 17,0,0,0 //设置TEXT模式参数
收←◆OK

发→◇at+cmgw="+862856044778640" //存储短信
收←◆>
发→◇hello world
收←◆+CMGW: 4
收←◆OK

发→◇at+cmss=4 //从存储区发送
收←◆+CMSS: 14
收←◆OK
```

### 13.15.2.2. PDU 模式 存储+发送

```
发→◇at+cfun=1 //上电
收←◆OK

发→◇at+cgclass=CC //CS域附着
收←◆OK

发→◇at+cmgf=0 //设置为PDU模式
收←◆OK

发→◇at+cmgw=24 //存储短信
收←◆>
发→◇049168000001000D91685127234440F900000BE8329BFD06DDDF723619
收←◆+CMGW: 5
收←◆OK

发→◇at+cmss=5 //从存储区发送
收←◆+CMSS: 15
收←◆OK
```

## 13.15.3. 短信的查看删除 (+CMGR +CMGL +CMGD)

**注：**TEXT 模式下，使用+CMGW 进行短信存储，一定要先执行2个 AT

AT+CFUN=1（从 SIM 卡读取存储所需要的短消息中心地址，或者+CSCA 指定）

AT+CMGF=1（指定为 TEXT 模式）（模式参数，在设置 TEXT 模式后得到缺省值）

```
发→◇at+cfun=1 //上电
```

```
收←◆OK
```

```
发→◇at+cgclass=CC //CS域附着
```

```
收←◆OK
```

```
发→◇at+cmgf=0 //设置为PDU模式
```

```
收←◆OK
```

```
发→◇at+cmgw=24 //存储短信
```

```
收←◆>
```

```
发→◇04916800001000D91685127234440F900000BE8329BFD06DDDF723619
```

```
收←◆+CMGW: 4
```

```
收←◆OK
```

```
发→◇at+cmgl=4 //PDU模式下，查看存储的所有短信
```

```
收←◆+CMGL: 0, 2, 22, 0491680000110A0F9168826540748746F000000005E8329BFD06
```

```
收←◆+CMGL: 1, 2, 23, 0491680000110B0F9168826540748746F00000000631D98C56B301
```

```
收←◆+CMGL: 2, 2, 21, 0491680000110C0F9168826540748746F000000004375C0E06
```

```
收←◆+CMGL: 3, 2, 27, 0491680000110D0F9168826540748746F00000000BE8329BFD06
```

```
DDDF723619
```

```
收←◆OK
```

```
发→◇at+cmgr=1 //查看index为1短信内容
```

```
收←◆+CMGR: 2, 23, 0491680000110B0F9168826540748746F00000000631D98C56B301
```

```
收←◆OK
```

```
发→◇at+cmgf=1 //设置为TEXT模式
```

```
收←◆OK
```

```
发→◇at+csmf=17,0,0,0 //设置TEXT模式参数
```

```
收←◆OK
```

发→◇at+cmgr=3 //查看index为4短信内容

收←◆+CMGR: 2, "862856044778640", 11

hello world

收←◆OK

发→◇at+cmgw="862856044778640" //存储短信

收←◆>

发→◇hello hywang

收←◆+CMGW: 4

收←◆OK

发→◇at+cmgl=4 //TEXT模式下, 查看存储的所有短信

收←◆+CMGL: 0, 2, "862856044778640", 5

hello

收←◆+CMGL: 1, 2, "862856044778640", 6

123456

收←◆+CMGL: 2, 2, "862856044778640", 4

7890

收←◆+CMGL: 3, 2, "862856044778640", 11

hello world

收←◆OK

发→◇at+cmgd=2 //删除index为2的短信

收←◆OK

发→◇at+cmgl=4 //TEXT模式下, 查看存储的所有短信

收←◆+CMGL: 0, 2, "862856044778640", 5

hello

收←◆+CMGL: 1, 2, "862856044778640", 6

123456

收←◆+CMGL: 3, 2, "862856044778640", 11

hello world

收←◆+CMGL: 4, 2, "862856044778640", 12

hello hywang

收←◆OK

### 13.15.4. 长短信发送

**注：**当前支持 TEXT 模式下的长短信发送，不支持 PDU 模式下长短信发送。

```

发→◇at+cfun=1      //上电
收←◆OK

发→◇at+cgclass=CC //CS域附着
收←◆OK

发→◇at+cmgf=1 //设置为TEXT模式
收←◆OK

发→◇at+csmf= 81,0,0,0 //设置TEXT模式参数，81表示长短信 7bit编码 可发送数
据大于160字符
收←◆OK

发→◇at+cmgs="15723244049" //存储短信
收←◆>
发→◇
012345678901234567890123456789012345678901234567890123456789012345678901234
5678901234567890123456789012345678901234567890123456789012345678901234
45678901234567890123456789
收←◆+CMGS: 14
收←◆OK

```

## 14. 硬件相关命令

### 14.1. AT+CHIPTEMP 读取芯片内部温度

Command	Possible Response(s)
+CHIPTEMP	+CHIPTEMP:<value> OK

#### 参数

<value> 芯片内部温度值

举例：

```

发→◇AT+CHIPTEMP
收←◆+CHIPTEMP: 30.20
收←◆OK

```

## 14.2. AT+BATVOL 读取电源供电电压

Command	Possible Response(s)
+BATVOL	+BATVOL:<value> OK

### 参数

<value> 电源供电电压值

举例:

发 → ◇ AT+BATVOL

收 ← ◆ +BATVOL: 3.96

收 ← ◆ OK

## 15. 低功耗控制命令

### 15.1. AT+LPM 低功耗控制

Command	Possible Response(s)
+LPM= [<mode>], [<control>]	OK ERROR
+LPM?	+LPM: <mode>, <control>
+LPM=?	+LPM: <mode>, <control state>

用于低功耗控制，主要有 normal、sleep、retention 2 种模式，默认状态为 normal，不进入低功耗：

### 参数

<mode>: GPRS 工作模式，主要用于功耗控制

0:sleep

sleep 模式，芯片进入睡眠，外部中断唤醒后重新启动；

1:retention

retention 模式，GPRS 根据基站的通信选择空闲状态进入睡眠，有任务后继续工作，外部中断可以正常唤醒。串口中断唤醒后在 1S 内不会进入 retention 模式（串口数据触发中断唤醒会造成数据丢失，如果是通过 AT 唤醒，则需要连续发送 2 个 AT 命令，第一个 AT 由于部分数据丢失会返回 ERROR，在 1S 内第二个 AT 能正常执行）。

<control>: 控制功耗开关

0: 打开。允许进入低功耗。

1: 关闭。不允许进入低功耗。

## 16. 工厂测试命令

### 16.1. AT+FACTORYTEST 工厂测试控制命令

工厂测试指令只支持工厂 FACTORY 版本，AT 版本和 OPENCPU 版本都不支持此指令。

Command	Possible Response(s)
+FACTORYTEST=<type> ,<control > ,<control1> ,<band>,<power>,<rssi>	OK ERROR
+FACTORYTEST=?	+FACTORYTEST:<type>,<control>,<control1>

用于工厂测试各项功能. 没有配置相关参数, 默认值为 0. (工厂测试 AT 只有在工厂测试版本上可以用)

<type> 测试的类型:

- 0: RF 测试
- 1: SIM 测试
- 2: LED 测试
- 3: AD A 通道测试
- 4: AD B 通道测试
- 5: AD C 通道测试
- 6: DA 测试, 输出固定 1.2V
- 7: GPS 输出方波测试
- 8: DCXO 校准
- 9: TEM 温度测试
- 10: I2C 控制
- 11: SPI MASTER 控制
- 12: GPIO 口控制
- 13: PRINTF 控制

<control> 控制

0/1:开/关或者拉低/拉高/GPIO

<control1> 控制

- 0: TX
- 1: RX

<band> 频点

- 0: 低频点频率是 89.04M
- 1: 高频点频率是 17840.2M

<power> 功率

<rssi> RSSI 值

## 测试-工厂测试流程

1、RF 控制返回值格式+FACTORYTEST:0, x

- X 含义
- 0 关闭测试模式
- 1 通过测试
- 2 测试失败
- 3 发送正常

- 4 同步失败
- 5 GPS 发送信号开始
- 6 DCXO  $\Delta f$  正常
- 7 数据接收正常, RSSI 接收功率异常
- 8 其他错误

#### 1.1、RF 开始接收

```
at+factorytest=0,1,1,0,800,-65
+FACTORYTEST:0,1
OK
```

#### 1.2、RF 结束接收

```
at+factorytest=0,0,1,0,800,-65
+FACTORYTEST:0,0
OK
```

#### 1.3、RF 开始发送

```
at+factorytest=0,1,0,0,800,-65
+FACTORYTEST:0,3
OK
```

#### 1.4、RF 结束发送

```
at+factorytest=0,0,0,0,800,-65
+FACTORYTEST:0,0
OK
```

### 2、SIM 测试

```
at+factorytest=1
+FACTORYTEST:1,0
OK
```

### 3、LED 测试

#### 3.1、LED 的 GPIO 拉高

```
at+factorytest=2,1
OK
```

#### 3.2、LED 的 GPIO 拉低

```
at+factorytest=2,0
OK
```

### 4、AD A 通道测试

返回结果是 AD 采样值, 换算公式:  $AD \text{ 值} / 4096 * 1.6$  (单位是 V)

```
at+factorytest=3
+FACTORYTEST:3,1135
OK
```

## 5、AD B 通道测试

返回结果是 AD 采样值，换算公式： $AD \text{ 值} / 4096 * 1.6$ （单位是 V）

```
at+factorytest=4
+FACTORYTEST:4,573
OK
```

## 6、AD C 通道测试

返回结果是 AD 采样值，换算公式： $AD \text{ 值} / 4096 * 1.6$ （单位是 V）

```
at+factorytest=5
+FACTORYTEST:4,573
OK
```

## 7、DA 通道测试

```
at+factorytest=6,1
OK
```

## 8、GPS 输出方波测试

## 8.1、GPS 可变功率

```
at+factorytest=7,4,0,0,800,-65
+FACTORYTEST:7,5
OK
```

## 8.2、GPS 恒定功率

```
at+factorytest=7,3,0,0,800,-65
+FACTORYTEST:7,5
OK
```

9、DCXO 校准的  $\Delta \text{ppm}$ ，返回  $\Delta \text{ppm}$  数据必须乘 100 后写入 flash 指定地址  
(0x1FE2F8)

```
at+factorytest=8,1
+FACTORYTEST:8,2.2
OK
```

## 10、温度测试

```
at+factorytest=9
+FACTORYTEST:9,22
OK
```

## 11、SPI MASTER 通过读 flash id 测试

```
at+factorytest=10,0
+FACTORYTEST:10,0xb14
OK
```

## 12、I2C 通过读写 flash 测试

### 12.1、I2C 读写 flash 成功

```
at+factorytest=11,0
+FACTORYTEST:11,0
OK
```

### 12.2、I2C 读写 flash 失败

```
at+factorytest=11,0
+FACTORYTEST:11,1
OK
```

## 13、GPIO 控制

默认 gpio 输出功能，可以控制输出高/低。

格式： +FACTORYTEST:type, gpiox, 0/1

-type: 12

-gpiox: [0 - 32]

-0/1: 输出 低电平/高电平

### 13.1、拉低 GPIO8

```
at+factorytest=12,8,0
OK
```

### 13.2、拉高 GPIO 8

```
at+factorytest=12,8,1
OK
```

## 14、PRINTF 控制

### 14.1、关闭 PRINTF

```
at+factorytest=13,0
OK
```

### 14.2、打开 PRINTF

```
at+factorytest=13,1
OK
```

## 17. 其它命令

### 17.1. AT+CMEE 设置日志输出明细

Command	Possible response(s)
+CMEE=[<n>]	
+CMEE?	+CMEE: <n>
+CMEE=?	+CMEE: (list of supported <n>s)

#### 参数

<n>: 报告模块开启状态

0 表示关闭模块的出错报告, 当模块的 AT 执行错误时, 返回值仅为"ERROR"

1 表示开启模块的出错报告,当模块的 AT 执行错误时,返回值为"+CME ERROR: NUM " 或者"+EXT ERROR: NUM ", NUM 代表错误代码,依照此代码可查询错误类型,可附表 3.1CME 错误码和 3.2EXT 错误码

2 表示开启模块的出错报告,当模块的 AT 执行错误时,返回值为"+CME ERROR: verbose "或者"+EXT ERROR: NUM ",verbose 代表错误细节

注意: 本文档+ERROR 指+CME ERROR 或者+EXT ERROR

举例:

发→◇AT+CMEE=?

收←◆+CMEE: (0-2)

收←◆OK

发→◇AT+CMEE=2

收←◆OK

发→◇AT+CMEE? //查询当前日志输出类型配置

收←◆+CMEE: 2

收←◆OK

## 17.2. 示例

举例:

发→◇AT+CFUN=1

收←◆OK

发→◇AT+CMEE=? //测试命令 列出 CMEE 支持枚举值范围

收←◆+CMEE: (0-2)

收←◆OK

发→◇AT+CMEE? //查询命令 查看当前配置值

收←◆+CMEE: 0

收←◆OK

发→◇AT+CMEE=2 //设置命令 设置日志级别 2 显示错误明细

收←◆OK

发→◇AT+CMEE? //查询命令 判断设置是否正确

收←◆+CMEE: 2

收←◆OK

发→◇AT+CGACT=1,5 //激活 PDP cid 为异常值 5 测试是否正确输出日志明细

收←◆+CME ERROR: operation not allowed //返回错误明细

## 18. 附表

### 18.1. CMEE 错误码

code	mean
-1	other error
0	phone failure
1	no connection to phone
2	phone-adaptor link reserved
3	operation not allowed
4	operation not supported
5	PH-SIM PIN required
10	SIM not inserted
11	SIM PIN required
12	SIM PUK required
13	SIM failure
14	SIM busy
15	SIM wrong
16	incorrect password
17	SIM PIN2 required
18	SIM PUK2 required
20	memory full
21	invalid index
22	not found
23	memory failure
24	text string too long
25	invalid characters in text string
26	dial string too long
27	invalid characters in dial string
30	no network service
31	network timeout
32	network not allowed - emergency calls only
33	last AT cmd not completed
100	unknown
103	illegal MS

---

106	illegal ME
107	GPRS service not allowed
111	PLMN not allowed
112	Location not allowed
113	Roaming not allowed in Location Area
132	GPRS service option not supported
133	requested service option not subscribed
134	service option temporarily out of order
148	unspecified GPRS error
149	PDP authorisation error
150	invalid module class
512	failed to abort
513	ACM reset needed
514	IP CLOSE FAIL

## 18.2. EXT 错误码

code	mean
-1	other error
0	parameter not allowed
1	data corrupted
2	internal error
3	call table full
4	service table full
5	call not found
6	no data-call supported
7	one call on hold
8	hold call not supported for this type
9	number not allowed by FDN
10	number not allowed by BDN
11	parallel USSD not supported
12	fax minimum speed condition
13	conflict with command details
14	no pdp context defined
15	no set device function(at+cfun=1)
16	error unknown

### 18.3. HTTP 错误码

code	mean
0	a successful operation
101	An invalid parameter was passed into an API function
102	Invalid payload
103	HTTPS message was too large to fit into a configured synchronous body buffer
104	Overflow occurred somewhere
105	A buffer provided could not hold data required by the library
106	Queue full
107	Operation retry. Could not find an item specified by an API.
108	Returned for not being able to find the address in a URL, the path in a URL, or a header field from the response headers. The HTTP request message was finished being written and we
109	cannot write more with @ref https_client_function_writerequestbody.
201	An error occurred internally to the library.
202	A network error occurred
203	A network connection error occurred.
204	A stream error occurred.
205	An authentication error occurred.
206	A TLS error occurred
207	An error occurred during the user callback.
208	The synchronous response could not be received in the specified timeout in @ref https_client_function_sendsync.
209	An error in the HTTP protocol.
210	The HTTPS request send was cancelled.
211	The HTTPS response receiving was cancelled.
212	The asynchronous request had an error being scheduled.
213	There was an error parsing the HTTP response
901	Fatal HTTP library error.
902	The connection is busy and cannot be cleaned up. The connection was closed, but @ref

---

`https_client_function_disconnect`  
must be called again to cleanup connection resources.

903 Try again

904 Data exists.

905 The operation on the public API is not supported.

## 18.4. MQTT 错误码

code	mean
0	MQTT operation completed successfully.
1	MQTT operation queued, awaiting result.
2	Initialization failed.
3	At least one parameter is invalid.
4	MQTT operation failed because of memory allocation failure.
5	MQTT operation failed because the network was unusable.
6	MQTT operation could not be scheduled, i.e. enqueued for sending.
7	MQTT response packet received from the network is malformed.
8	A blocking MQTT operation timed out.
9	A CONNECT or at least one subscription was refused by the server.
10	A QoS 1 PUBLISH received no response and [the retry limit]
11	An API function was called before @ref mqtt_function_init

## 18.5. FTP 错误码

code	mean
0	FTP_NO_ERROR
1	FTP_OTHER_ERROR
2	FTP_PLEASE_LOGIN_USERNAME_ERROR
3	FTP_EXISTING
4	FTP_CONNECT_ERROR.
5	FTP_SEND_ERROR,
6	FTP_RECV_ERROR
7	FTP_STATE_ERROR,
8	FTP_WAIT_RECV_TIMEOUT
9	FTP_PASV_OK
10	FTP_SIZE_OK
11	FTP_DOWN_OK
12	FTP_PASV_TASK_BUSY
13	FTP_PASV_PORT_ERR
14	FTP_PASV_OPEN_ERR
110	FTP_RESTART_TAG_REPLY
120	FTP_SERVICE_READY_IN_X_MINUTES

---

125	FTP_DATA_CONNECT_OPENED
150	FTP_READY_OPEN_DATA_CONNECT
200	FTP_COMMAND_SUCCEED
202	FTP_NO_COMMAND
211	FTP_SYSTEM_STATE
212	FTP_DIR_STATE
213	FTP_FILE_STATE
215	FTP_NAME
220	FTP_SERVICE_READY_FOR_NEW_USERS
221	FTP_SERVICE_CLOSING_CONTROL_CONNECT
226	FTP_CLOSE_DATA_CONNECT
227	FTP_INPUT_PASSIVE_MODE
230	FTP_USER_LOGGED_IN
250	FTP_CWD_PATH_OK
257	FTP_PATHNAME_HAVE_CREATE
331	FTP_USER_NEED_PASS
332	FTP_NEED_USER
350	FTP_REQUEST_FILE_OPERATION_WAIT
400	FTP_ERROR
421	FTP_SERVER_INVALID
425	FTP_UNABLE_OPEN_DATA_CONNECT
426	FTP_CLOSE_CONNECT
450	FTP_FILE_OPERATION_COULD_NOT_PERFORMED
451	FTP_FILE_OPERATION_ABORTED
500	FTP_SYNTAX_ERROR
501	FTP_PARAMENT_ERROR
502	FTP_COMMAND_NOT_IMPLEMENTED
504	FTP_NO_PARAMENT
530	FTP_UNABLE_LOGIN
532	FTP_ACCOUNT_NUM_REQUEST_FILE_STORAGE
550	FTP_REQUEST_NOT_EXECUTED
551	FTP_OPRATE_ABOART
552	FTP_FILE_OPRATE_ABOART
553	FTP_REQUEST_ACTION_COULD_NOT_PERFORMED