

# ESP8266

## AT 指令集



版本 3.0  
乐鑫信息科技  
版权所有 © 2018

# 关于本手册

本文档提供 ESP8266\_NONOS\_SDK 的 AT 指令说明。

## 发布说明

日期	版本	发布说明
2016.04	V1.5.3	首次发布。
2016.05	V1.5.4	更新章节 5.2.16 和 5.2.19。
2016.07	V2.0.0	增加章节 3.2.11, 更新章节 1.2。
2017.05	V2.1.0	更新章节 3.2, 4.1 和 5.2。
2017.08	V2.1.1	增加附录 B
2018.02	V2.2	增加章节 3.2.22, 3.2.23, 4.2.39, 4.2.40 和 5.2.15。 更新章节 4.2.7, 4.2.8 和附录 B。
2018.05	V2.2.1	更新章节 4.2.10, 4.2.11 和 4.2.12。
2018.05	V2.2.2	增加章节 5.2.11, 5.2.23, 5.2.24。 更新章节 3.2.10。
2018.08	V3.0	增加章节 5.2.5, 5.2.25, 5.2.26, 5.2.27。 更新第 1 章, 章节 4.2.3, 4.2.4, 4.2.11, 4.2.12, 附录 A。 删除 AT+RFAUTOTRACE 指令。

## 文档变更通知

用户可通过乐鑫官网订阅页面 <https://www.espressif.com/zh-hans/subscribe> 订阅技术文档变更的电子邮件通知。

## 证书下载

用户可通过乐鑫官网证书下载页面 <https://www.espressif.com/zh-hans/certificates> 下载产品证书。

# 目录

1. 前言 .....	1
1.1. 客制化 AT 固件.....	1
1.1.1. 编译 AT 工程.....	1
1.1.2. 客制化功能.....	1
1.1.3. 新增自定义 AT 指令 .....	1
1.2. 烧录 AT 固件.....	2
1.2.1. 16 Mbit Flash, Map: 1024 KB + 1024 KB.....	3
1.2.2. 32 Mbit Flash, Map: 1024 KB + 1024 KB.....	3
1.2.3. 4 Mbit Flash.....	3
1.2.4. 8 Mbit Flash.....	4
1.2.5. 16 Mbit Flash, Map: 512 KB + 512 KB.....	4
1.2.6. 32 Mbit Flash, Map: 512 KB + 512 KB.....	4
2. 指令说明.....	6
3. 基础 AT 指令.....	7
3.1. 基础 AT 指令一览表.....	7
3.2. 基础 AT 指令描述 .....	8
3.2.1. AT—测试 AT 启动 .....	8
3.2.2. AT+RST—重启模块.....	8
3.2.3. AT+GMR—查询版本信息.....	8
3.2.4. AT+GSLP—进入 Deep-sleep 模式 .....	8
3.2.5. ATE—开关回显功能 .....	9
3.2.6. AT+RESTORE—恢复出厂设置.....	9
3.2.7. AT+UART_CUR—设置 UART 当前临时设置，不保存到 Flash.....	10
3.2.8. AT+UART_DEF—设置 UART 配置，保存到 Flash.....	11
3.2.9. AT+SLEEP—设置 sleep 模式.....	12

3.2.10. AT+WAKEUPGPIO—设置 GPIO 唤醒 Light-sleep 模式.....	12
3.2.11. AT+RFPOWER—设置 RF TX Power 上限.....	13
3.2.12. AT+RFVDD—根据 VDD33 设置 RF TX Power .....	13
3.2.13. AT+SYSRAM—查询系统当前剩余内存.....	14
3.2.14. AT+SYSADC—查询 ADC 值.....	14
3.2.15. AT+SYSIOSETCFG—设置 IO 工作模式.....	14
3.2.16. AT+SYSIOGETCFG—查询 IO 工作模式 .....	14
3.2.17. AT+SYSGPIODIR—设置 GPIO 工作为输入或输出 .....	15
3.2.18. AT+SYSGPIOWRITE—设置 GPIO 的输出电平 .....	15
3.2.19. AT+SYSGPIOREAD—读取 GPIO 的电平状态.....	16
3.2.20. AT+SYSMSG_CUR—设置当前系统消息，不保存到 flash.....	17
3.2.21. AT+SYSMSG_DEF—设置默认系统消息，保存到 flash .....	17

#### 4. 基础 AT 指令.....18

4.1. 基础 Wi-Fi 功能 AT 指令一览表 .....	18
4.2. 基础 Wi-Fi 功能 AT 指令描述.....	20
4.2.1. AT+CWMODE_CUR—设置当前 Wi-Fi 模式，不保存到 Flash .....	20
4.2.2. AT+CWMODE_DEF—设置当前 Wi-Fi 模式并保存到 Flash.....	20
4.2.3. AT+CWJAP_CUR—临时连接 AP .....	21
4.2.4. AT+CWJAP_DEF—连接 AP，保存到 Flash.....	22
4.2.5. AT+CWLAPOPT—设置 CWLAP 指令的属性 .....	23
4.2.6. AT+CWLAP—扫描当前可用的 AP .....	24
4.2.7. AT+CWQAP—断开与 AP 的连接 .....	25
4.2.8. AT+CWSAP_CUR—配置 ESP8266 SoftAP 当前参数.....	26
4.2.9. AT+CWSAP_DEF—配置 ESP8266 SoftAP 当前参数，保存到 Flash.....	26
4.2.10. AT+CWLIF—查询连接到 ESP8266 SoftAP 的 Station 信息.....	28
4.2.11. AT+CWDHCP_CUR—设置 DHCP，不保存到 Flash .....	28
4.2.12. AT+CWDHCP_DEF—设置 DHCP，保存到 Flash.....	29

4.2.13. AT+CWDHCPS_CUR—设置 ESP8266 SoftAP DHCP 分配的 IP 范围，不保存到 Flash.....	30
4.2.14. AT+CWDHCPS_DEF—设置 ESP8266 SoftAP DHCP 分配的 IP 范围，保存到 Flash.....	30
4.2.15. AT+CWAUTOCONN—上电是否自动连接 AP .....	31
4.2.16. AT+CIPSTAMAC_CUR—设置 ESP8266 Station 当前 MAC 地址，不保存到 Flash.....	31
4.2.17. AT+CIPSTAMAC_DEF—设置 ESP8266 Station 当前 MAC 地址，保存到 Flash.....	32
4.2.18. AT+CIPAPMAC_CUR—设置 ESP8266 SoftAP 当前 MAC 地址，不保存到 Flash.....	32
4.2.19. AT+CIPAPMAC_DEF—设置 ESP8266 SoftAP 默认 MAC 地址，保存到 Flash .....	32
4.2.20. AT+CIPSTA_CUR—设置 ESP8266 Station 的 IP 地址，不保存到 Flash.....	33
4.2.21. AT+CIPSTA_DEF—设置 ESP8266 Station 的 IP 地址，保存到 Flash .....	33
4.2.22. AT+CIPAP_CUR—设置 ESP8266 SoftAP 的 IP 地址，不保存到 Flash.....	34
4.2.23. AT+CIPAP_DEF—设置 ESP8266 SoftAP 的 IP 地址，保存到 Flash.....	34
4.2.24. AT+CWSTARTSMART—开启 SmartConfig .....	35
4.2.25. AT+CWSTOPSMART—停止 SmartConfig.....	36
4.2.26. AT+CWSTARTDISCOVER—开启可被局域网内微信探测模式 .....	36
4.2.27. AT+CWSTOPDISCOVER—关闭可被局域网内微信探测模式.....	37
4.2.28. AT+WPS—设置 WPS 功能 .....	37
4.2.29. AT+MDNS—设置 MDNS 功能 .....	37
4.2.30. AT+CWHOSTNAME—设置 ESP8266 Station 的主机名称 .....	38
4.2.31. AT+CWCOUNTRY_CUR—设置 ESP8266 WiFi 国家码 .....	38
4.2.32. AT+CWCOUNTRY_DEF—设置默认的 ESP8266 WiFi 国家码，并保存到 flash.....	39

<b>5. TCP/IP 功能 AT 指令 .....</b>	<b>40</b>
5.1. TCP/IP 指令一览表.....	40
5.2. TCP/IP 指令描述 .....	41
5.2.1. AT+CIPSTATUS—查询网络连接信息 .....	41
5.2.2. AT+CIPDOMAIN—域名解析功能 .....	41
5.2.3. AT+CIPSTART—建立 TCP 连接，UDP 传输或 SSL 连接.....	42
5.2.4. AT+CIPSSLSIZE—设置 SSL Buffer 容量.....	44

5.2.5. AT+CIPSSLCONF—配置 SSL Client .....	44
5.2.6. AT+CIPSEND—发送数据 .....	45
5.2.7. AT+CIPSENDEX—发送数据 .....	46
5.2.8. AT+CIPSENDERBUF—数据写入 TCP 发包缓存 .....	46
5.2.9. AT+CIPBUFRESET—重新计数 .....	47
5.2.10. AT+CIPBUFSTATUS—查询 TCP 发包缓存的状态 .....	48
5.2.11. AT+CIPCHECKSEQ—查询写入 TCP 发包缓存的某包是否发送成功 .....	48
5.2.12. AT+CIPCLOSEMODE—设置 TCP 连接的断开方式 .....	49
5.2.13. AT+CIPCLOSE—关闭 TCP/UDP/SSL 传输 .....	49
5.2.14. AT+CIFSR—查询本地 IP 地址 .....	50
5.2.15. AT+CIPMUX—设置多连接 .....	50
5.2.16. AT+CIPSERVER—建立 TCP 服务器 .....	50
5.2.17. AT+CIPSERVERMAXCONN—设置服务器允许建立的最大连接数 * .....	51
5.2.18. AT+CIPMODE—设置传输模式 .....	51
5.2.19. AT+SAVETRANSLINK—保存透传到 Flash .....	52
5.2.20. AT+CIPSTO—设置 TCP 服务器超时时间 .....	53
5.2.21. AT+PING—Ping 功能 .....	53
5.2.22. AT+CIUPDATE—通过 Wi-Fi 升级软件 .....	53
5.2.23. AT+CIPDINFO—接收网络数据时是否提示对端 IP 和端口 .....	54
5.2.24. +IPD—接收网络数据 .....	54
5.2.25. AT+CIPRCVMODE—设置 TCP 连接的数据接收方式 .....	55
5.2.26. AT+CIPRCVDATA—被动接收模式时, 读取缓存的 TCP 数据 .....	55
5.2.27. AT+CIPRCVLEN—被动接收模式时, 查询缓存 TCP 数据的长度 .....	56
5.2.28. AT+CIPSNTPCFG—设置时域和 SNTP 服务器 .....	56
5.2.29. AT+CIPSNTPTIME—查询 SNTP 时间 .....	56
5.2.30. AT+CIPDNS_CUR—自定义 DNS 服务器, 不保存到 Flash .....	57
5.2.31. AT+CIPDNS_DEF—自定义 DNS 服务器, 保存到 Flash .....	57

A. 附录 A.....59

B. 附录 B .....60

C. Q&A.....61



# 1.

# 前言

乐鑫官方发布 ESP8266 [AT 固件](#)，供用户直接下载使用。同时，乐鑫也提供 [AT 工程](#) 供用户在乐鑫 AT 的基础上，自行开发客制化 AT 固件。

本文包含客制化 AT 固件的方法，AT 固件的烧录，以及乐鑫已有的 AT 指令集说明。

## ⚠ 注意：

- 请确保正确的 BIN (*/ESP8266\_NONOS\_SDK/bin/at*) 已经参考 *ESP8266\_NONOS\_SDK/bin/at* 中 *readme.txt* 烧录到 ESP8266 模块，再执行文档中的 AT 指令。
- AT 底层已占用 `system_os_task` 优先级 0 和 1，如果用户基于 AT 开发，仅支持建立一个优先级为 2 的任务。

## 1.1. 客制化 AT 固件

### 1.1.1. 编译 AT 工程

用户如需编译客制化 AT 源代码，新增自定义的 AT 指令，请将 example 中的文件夹 [at](#) 拷贝到对应版本的 [ESP8266\\_NONOS\\_SDK](#) 根目录下，再进入 at 文件夹进行开发及编译。详细编译说明，可参考 [ESP8266 入门指南](#)。

### 1.1.2. 客制化功能

#### • OTA 功能：

- 乐鑫发布的官方 AT 固件，默认支持使用 AT+CIUPDATE 指令，从乐鑫云下载升级到新版本的 AT 固件。
- 如果用户自行客制化 AT 固件，则需要自行实现 OTA 升级功能，从用户自己的云端下载升级自定义的新版本固件。乐鑫在 [at\\_upgrade.c](#) 中提供了 OTA 示例，以供参考。

#### • SmartConfig 功能：

- 乐鑫发布的官方 AT 固件，默认支持 AT+CWSTARTSMART 和 AT+CWSTOPSMART 指令。
- 如果用户无需 SmartConfig 功能，可以自行重新编译 at，在 [user\\_config.h](#) 中关闭 CONFIG\_AT\_SMARTCONFIG\_COMMAND\_ENABLE，从而减小 bin size，节省内存。

### 1.1.3. 新增自定义 AT 指令

自定义 AT 指令命名时，使用英文字符，请勿使用其他特殊字符或数字。





AT 基于 ESP8266\_NONOS\_SDK 编译, *ESP8266\_NONOS\_SDK/example/at* 中提供了开发者自定义 AT 指令的示例。乐鑫原本提供的 AT 指令以库文件 *libat.a* 的形式提供, 将包含在编译生成的 AT BIN 固件中。

*!ESP8266\_NONOS\_SDK/examples/at/user/user\_main.c* 中提供了实现自定义 AT 指令的示例, 可参考实现。结构体 *at\_funcationType* 用于定义一条指令的四种类型, 类型定义如下表所示。

类型定义	类型描述	说明	
at_testCmd	测试指令	对应指令	AT+TEST=?
		AT 示例中注册的实现回调	at_testCmdTest
		建议指令功能	返回参数的取值范围
		若此参数注册为 NULL, 则无测试指令	
at_queryCmd	查询指令	对应指令	AT+TEST?
		AT 示例中注册的实现回调	at_queryCmdTest
		建议指令功能	返回当前值
		若此参数注册为 NULL, 则无查询指令	
at_setupCmd	设置指令	对应指令格式	AT+TEST=parameter1,parameter2,...
		AT 示例中注册的实现回调	at_setupCmdTest
		建议指令功能	设置参数值
		若此参数注册为 NULL, 则无设置指令	
at_exeCmd	执行指令	对应指令	AT+TEST
		AT 示例中注册的实现回调	at_exeCmdTest
		建议指令功能	执行某项操作
		若此参数注册为 NULL, 则无执行指令	

## 1.2. 烧录 AT 固件

参考 *ESP8266\_NONOS\_SDK/bin/at/readme.txt* 进行烧录, 请使用乐鑫官方烧录工具, 烧录时注意选择对应的 Flash 大小。

乐鑫官方烧录工具链接:

[http://espressif.com/zh-hans/support/download/other-tools?keys=&field\\_type\\_tid%5B%5D=14](http://espressif.com/zh-hans/support/download/other-tools?keys=&field_type_tid%5B%5D=14)。

从 ESP8266\_NONOS\_SDK\_V3.0.0, AT\_V1.7 起, 由于 AT bin size 增大, AT 默认只支持 1024 KB + 1024 KB 的 flash map。



### 1.2.1. 16 Mbit Flash, Map: 1024 KB + 1024 KB

使用 Espressif 官方烧录工具，烧录时选择 Flash size : 16 Mbit-C1。

BIN	烧录地址	说明
blank.bin	0x1FB000	初始化 RF_CAL 参数区
esp_init_data_default.bin	0x1FC000	初始化其他射频参数区，至少烧录一次。 当 RF_CAL 参数区初始化烧录时，本区域也需烧录。
blank.bin	0xFE000	初始化用户参数区，详见附录
blank.bin	0x1FE000	初始化系统参数区，详见附录
boot.bin	0x00000	主程序，位于 <i>/bin/at</i>
user1.2048.new.5.bin	0x01000	主程序，位于 <i>/bin/at/1024+1024</i>

### 1.2.2. 32 Mbit Flash, Map: 1024 KB + 1024 KB

使用 Espressif 官方烧录工具，烧录时选择 Flash size: 32 Mbit-C1。

BIN	烧录地址	说明
blank.bin	0x3FB000	初始化 RF_CAL 参数区
esp_init_data_default.bin	0x3FC000	初始化其他射频参数区，至少烧录一次。 当 RF_CAL 参数区初始化烧录时，本区域也需烧录。
blank.bin	0xFE000	初始化用户参数区，详见附录
blank.bin	0x3FE000	初始化系统参数区，详见附录
boot.bin	0x00000	主程序，位于 <i>/bin/at</i>
user1.2048.new.5.bin	0x01000	主程序，位于 <i>/bin/at/1024+1024</i>

### 1.2.3. 4 Mbit Flash

从 ESP8266\_NONOS\_SDK\_V2.0.0, AT\_V1.3 起，AT 固件可以使用 4 Mbit Flash，但不支持升级功能。

BIN	烧录地址	说明
blank.bin	0x78000	初始化 RF_CAL 参数区。
esp_init_data_default.bin	0x7C000	初始化其他射频参数区，至少烧录一次。 当 RF_CAL 参数区初始化烧录时，本区域也需烧录。
blank.bin	0x7A000	初始化用户参数区，详见附录。



BIN	烧录地址	说明
blank.bin	0x7E000	初始化系统参数区，详见附录。
eagle.flash.bin	0x00000	主程序，位于 <i>/bin/at/noboot</i> 。
eagle.irom0text.bin	0x10000	主程序，位于 <i>/bin/at/noboot</i> 。

#### 1.2.4. 8 Mbit Flash

固件升级功能（对应指令 AT+CIUPDATE）要求 Flash 容量为 8 Mbit 或以上，采用 boot mode 的烧录方式。使用 Espressif 官方烧录工具，烧录时选择 Flash size：8 Mbit。

BIN	烧录地址	说明
blank.bin	0xFB000	初始化 RF_CAL 参数区
esp_init_data_default.bin	0xFC000	初始化其他射频参数区，至少烧录一次。 当 RF_CAL 参数区初始化烧录时，本区域也需烧录。
blank.bin	0x7E000	初始化用户参数区，详见附录
blank.bin	0xFE000	初始化系统参数区，详见附录
boot.bin	0x00000	主程序，位于 <i>/bin/at</i>
user1.1024.new.2.bin	0x01000	主程序，位于 <i>/bin/at/512+512</i>

#### 1.2.5. 16 Mbit Flash, Map: 512 KB + 512 KB

使用 Espressif 官方烧录工具，烧录时选择 Flash size：16 Mbit。

BIN	烧录地址	说明
blank.bin	0x1FB000	初始化 RF_CAL 参数区
esp_init_data_default.bin	0x1FC000	初始化其他射频参数区，至少烧录一次。 当 RF_CAL 参数区初始化烧录时，本区域也需烧录。
blank.bin	0x7E000	初始化用户参数区，详见附录
blank.bin	0x1FE000	初始化系统参数区，详见附录
boot.bin	0x00000	主程序，位于 <i>/bin/at</i>
user1.1024.new.2.bin	0x01000	主程序，位于 <i>/bin/at/512+512</i>

#### 1.2.6. 32 Mbit Flash, Map: 512 KB + 512 KB

使用 Espressif 官方烧录工具，烧录时选择 Flash size：32 Mbit。



BIN	烧录地址	说明
blank.bin	0x3FB000	初始化 RF_CAL 参数区
esp_init_data_default.bin	0x3FC000	初始化其他射频参数区，至少烧录一次。 当 RF_CAL 参数区初始化烧录时，本区域也需烧录。
blank.bin	0x7E000	初始化用户参数区，详见附录
blank.bin	0x3FE000	初始化系统参数区，详见附录
boot.bin	0x00000	主程序，位于 <i>/bin/at</i>
user1.1024.new.2.bin	0x01000	主程序，位于 <i>/bin/at/512+512</i>



## 2.

# 指令说明

AT 指令可以细分为四种类型：

类型	指令格式	描述
测试指令	AT+<x>=?	该命令用于该命令用于查询设置指令的参数以及取值范围。
查询指令	AT+<x>?	该命令用于返回参数的当前值。
设置指令	AT+<x>=<...>	该命令用于设置用户自定义的参数值。
执行指令	AT+<x>	该命令用于执行受模块内部程序控制的变参数不可变的功能。

### ⚠ 注意：

- 不是每条 AT 指令都具备上述 4 种类型的命令。
- [] 括号内为缺省值，不必填写或者可能不显示。
- 使用双引号表示字符串数据 "string"，例如：AT+CWSAP="ESP756290","21030826",1,4
- 默认波特率为 115200。
- AT 指令必须大写，并且以回车换行符结尾 (CR LF)。



# 3. 基础 AT 指令

## 3.1. 基础 AT 指令一览表

指令	描述
AT	测试 AT 启动
AT+RST	重启模块
AT+GMR	查看版本信息
AT+GSLP	进入 Deep-sleep 模式
ATE	开关回显功能
AT+RESTORE	恢复出厂设置
AT+UART_CUR	UART 当前临时配置
AT+UART_DEF	UART 默认配置，保存到 Flash
AT+SLEEP	设置 sleep 模式
AT+WAKEUPGPIO	设置 GPIO 唤醒 Light-sleep 模式
AT+RFPOWER	设置 RF TX Power 上限
AT+RFVDD	根据 VDD33 设置 RF TX Power
AT+SYSRAM	查询系统当前剩余内存
AT+SYSADC	查询 ADC 值
AT+SYSIOSETCFG	设置 IO 工作模式
AT+SYSIOGETCFG	查询 IO 工作模式
AT+SYSGPIODIR	设置 GPIO 工作为输入或输出
AT+SYSGPIOWRITE	设置 GPIO 的输出电平
AT+SYSGPIOREAD	读取 GPIO 的电平状态
AT+SYMSG_CUR	设置当前系统消息
AT+SYMSG_DEF	设置默认系统消息



## 3.2. 基础 AT 指令描述

### 3.2.1. AT—测试 AT 启动

执行指令	AT
响应	OK
参数说明	-

### 3.2.2. AT+RST—重启模块

执行指令	AT+RST
响应	OK
参数说明	-

### 3.2.3. AT+GMR—查询版本信息

执行指令	AT+GMR
响应	<AT version info> <SDK version info> <compile time>  OK
参数说明	<ul style="list-style-type: none"><li>• &lt;AT version info&gt;: AT 版本信息</li><li>• &lt;SDK version info&gt;: SDK 版本信息</li><li>• &lt;compile time&gt;: 编译生成时间</li></ul>

### 3.2.4. AT+GSLP—进入 Deep-sleep 模式

设置指令	AT+GSLP=<time>
响应	<time> OK
参数说明	<time>: 设置 ESP8266 的睡眠时长, 单位: 毫秒。ESP8266 会在休眠设定时长后自动唤醒。
注意	Deep-sleep 功能需要硬件支持, 将 XPD_DCDC 通过 0 欧姆电阻连接到 EXT_RSTB, 用作 Deep-sleep 唤醒。



### 3.2.5. ATE—开关回显功能

执行指令	ATE
响应	OK
参数说明	<ul style="list-style-type: none"><li>• ATE0: 关闭回显</li><li>• ATE1: 开启回显</li></ul>

### 3.2.6. AT+RESTORE—恢复出厂设置

执行指令	AT+RESTORE
响应	OK
说明	恢复出厂设置，将擦除所有保存到 Flash 的参数，恢复为默认参数。 恢复出厂设置会导致机器重启。





## 3.2.7. AT+UART\_CUR—设置 UART 当前临时设置，不保存到 Flash

指令	查询指令： AT+UART_CUR?	设置指令： AT+UART_CUR=<baudrate>,<databits>,<stopbits>,<parity>,<flow control>
响应	+UART_CUR:<baudrate>,<databits>,<stopbits>,<parity>,<flow control> OK	OK
注意	查询返回的响应是 UART 实际参数值，由于时钟分频的原因，UART 实际参数值与设置值有一定误差，是正常情况。  例如，设置 UART 波特率为 115200 时，查询实际波特率值为 115273。	
参数说明	<ul style="list-style-type: none"> <li>• &lt;baudrate&gt;: UART 波特率</li> <li>• &lt;databits&gt;: 数据位 <ul style="list-style-type: none"> <li>▶ 5: 5 bit 数据位</li> <li>▶ 6: 6 bit 数据位</li> <li>▶ 7: 7 bit 数据位</li> <li>▶ 8: 8 bit 数据位</li> </ul> </li> <li>• &lt;stopbits&gt;: 停止位 <ul style="list-style-type: none"> <li>▶ 1: 1 bit 停止位</li> <li>▶ 2: 1.5 bit 停止位</li> <li>▶ 3: 2 bit 停止位</li> </ul> </li> <li>• &lt;parity&gt;: 校验位 <ul style="list-style-type: none"> <li>▶ 0: None</li> <li>▶ 1: Odd</li> <li>▶ 2: Even</li> </ul> </li> <li>• &lt;flow control&gt;: 流控 <ul style="list-style-type: none"> <li>▶ 0: 不使能流控</li> <li>▶ 1: 使能 RTS</li> <li>▶ 2: 使能 CTS</li> <li>▶ 3: 同时使能 RTS 和 CTS</li> </ul> </li> </ul>	
注意	<ol style="list-style-type: none"> <li>1. 本设置<b>不保存</b>在 Flash。</li> <li>2. 使用流控需要硬件支持流控： <ul style="list-style-type: none"> <li>▶ MTCK 为 UART0 CTS</li> <li>▶ MTDO 为 UART0 RTS</li> </ul> </li> <li>3. 波特率支持范围：110~115200*40。</li> </ol>	
示例	AT+UART_CUR=115200,8,1,0,3	



## 3.2.8. AT+UART\_DEF—设置 UART 配置，保存到 Flash

指令	查询指令： AT+UART_DEF?	设置指令： AT+UART_DEF=<baudrate>,<databits>,<stopbits>,<parity>,<flow control>
响应	+UART_DEF:<baudrate>,<databits>,<stopbits>,<parity>,<flow control> OK	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;baudrate&gt;: UART 波特率</li> <li>• &lt;databits&gt;: 数据位 <ul style="list-style-type: none"> <li>▶ 5: 5 bit 数据位</li> <li>▶ 6: 6 bit 数据位</li> <li>▶ 7: 7 bit 数据位</li> <li>▶ 8: 8 bit 数据位</li> </ul> </li> <li>• &lt;stopbits&gt;: 停止位 <ul style="list-style-type: none"> <li>▶ 1: 1 bit 停止位</li> <li>▶ 2: 1.5 bit 停止位</li> <li>▶ 3: 2 bit 停止位</li> </ul> </li> <li>• &lt;parity&gt;: 校验位 <ul style="list-style-type: none"> <li>▶ 0: None</li> <li>▶ 1: Odd</li> <li>▶ 2: Even</li> </ul> </li> <li>• &lt;flow control&gt;: 流控 <ul style="list-style-type: none"> <li>▶ 0: 不使能流控</li> <li>▶ 1: 使能 RTS</li> <li>▶ 2: 使能 CTS</li> <li>▶ 3: 同时使能 RTS 和 CTS</li> </ul> </li> </ul>	
注意	<ol style="list-style-type: none"> <li>1. 本设置将保存在 Flash user parameter 区，重新上电后仍生效。</li> <li>2. 使用流控需要硬件支持流控： <ul style="list-style-type: none"> <li>▶ MTCK 为 UART0 CTS</li> <li>▶ MTDO 为 UART0 RTS</li> </ul> </li> <li>3. 波特率支持范围：110~115200*40。</li> </ol>	
示例	AT+UART_DEF=115200,8,1,0,3	



## 3.2.9. AT+SLEEP—设置 sleep 模式

指令	查询指令： AT+SLEEP?	设置指令： AT+SLEEP=<sleep mode>
响应	当前 sleep 模式 OK	OK 或 ERROR
参数说明	<sleep mode>: <ul style="list-style-type: none"> <li>▶ 0: 禁用休眠模式</li> <li>▶ 1: Light-sleep 模式</li> <li>▶ 2: Modem-sleep 模式</li> </ul>	
注意	Sleep 模式仅在单 Station 模式下生效。默认为 Modem-sleep 模式。	
示例	AT+SLEEP=0	

## 3.2.10. AT+WAKEUPGPIO—设置 GPIO 唤醒 Light-sleep 模式

设置指令	AT+WAKEUPGPIO=<enable>,<trigger_GPIO>,<trigger_level>[,<awake_GPIO>,<awake_level>] ]
响应	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;enable&gt; <ul style="list-style-type: none"> <li>▶ 0: 禁用 GPIO 唤醒 Light-sleep 功能</li> <li>▶ 1: 使能 GPIO 唤醒 Light-sleep 功能</li> </ul> </li> <li>• &lt;trigger_GPIO&gt; <ul style="list-style-type: none"> <li>▶ 设置用于唤醒 Light-sleep 的 GPIO, 有效范围: [0, 15]</li> </ul> </li> <li>• &lt;trigger_level&gt; <ul style="list-style-type: none"> <li>▶ 0: 低电平唤醒</li> <li>▶ 1: 高电平唤醒</li> </ul> </li> <li>• [&lt;awake_GPIO&gt;] <ul style="list-style-type: none"> <li>▶ 选填参数, 设置 Light-sleep 唤醒后的标志 GPIO, 有效范围: [0, 15]</li> </ul> </li> <li>• [&lt;awake_level&gt;] <ul style="list-style-type: none"> <li>▶ 选填参数</li> <li>▶ 0: Light-sleep 唤醒后置为低电平</li> <li>▶ 1: Light-sleep 唤醒后置为高电平</li> </ul> </li> </ul>



注意	<ul style="list-style-type: none"> <li>Light sleep 唤醒过程约 5 ms，请等待至少 5ms 后再发送 AT 指令。</li> <li>&lt;trigger_GPIO&gt; 与 &lt;awake_GPIO&gt; 不能相同</li> <li>由 &lt;trigger_GPIO&gt; 触发 ESP8266 从 Light-sleep 唤醒之后，如需再次进入休眠时，ESP8266 将判断 &lt;trigger_GPIO&gt; 的状态： <ul style="list-style-type: none"> <li>如果 &lt;trigger_GPIO&gt; 仍然处于唤醒状态，则进入 Modem-sleep 休眠；</li> <li>如果 &lt;trigger_GPIO&gt; 不处于唤醒状态，则进入 Light-sleep 休眠。</li> </ul> </li> </ul>
示例	<ul style="list-style-type: none"> <li>设置 GPIO0 低电平唤醒 Light-sleep 模式： AT+WAKEUPGPIO=1,0,0</li> <li>设置 GPIO0 高电平唤醒 Light-sleep 模式，唤醒后，将 GPIO13 设置为高电平： AT+WAKEUPGPIO=1,0,1,13,1</li> <li>取消 GPIO 唤醒 Light-sleep 模式的功能： AT+WAKEUPGPIO=0</li> </ul>

### 3.2.11. AT+RFPOWER—设置 RF TX Power 上限

设置指令	AT+RFPOWER=<TX Power>
响应	OK
参数说明	<TX Power>：RF TX Power 值，参数范围：[0, 82]，单位：0.25 dBm
注意	RF TX Power 的设置并不精准，此时设置的是 RF TX Power 的最大值，实际值可能小于设置值。
示例	AT+RFPOWER=50

### 3.2.12. AT+RFVDD—根据 VDD33 设置 RF TX Power

指令	查询指令： AT+RFVDD? 功能：查询 ESP8266 VDD33 的值。	设置指令： AT+RFVDD=<VDD33> 功能：ESP8266 根据传入的 <VDD33> 调整 RF TX Power。	执行指令： AT+RFVDD 功能：ESP8266 自动根据实际的 VDD33 调整 RF TX Power。
响应	+RFVDD: <VDD33> OK	OK	OK
参数说明	<VDD33>：VDD33 电压值，单位：1/1024 V	<VDD33>：VDD33 电压值，取值范围：[1900, 3300]	-
注意	本查询指令必须在 TOUT 管脚悬空的情况下使用，否则，查询返回无效值。	-	本查询指令必须在 TOUT 管脚悬空的情况下使用。



示例	AT+RFVDD=2800
----	---------------

### 3.2.13. AT+SYSRAM—查询系统当前剩余内存

查询指令	AT+SYSRAM?
响应	+SYSRAM:<RAM> OK
参数说明	<RAM>: 系统当前剩余内存, 单位: byte

### 3.2.14. AT+SYSADC—查询 ADC 值

查询指令	AT+SYSADC?
响应	+SYSADC:<ADC> OK
参数说明	<ADC>: 查询到的 ADC 值, 单位: 1/1024V

### 3.2.15. AT+SYSIOSETCFG—设置 IO 工作模式

设置指令	AT+SYSIOSETCFG=<pin>,<mode>,<pull-up>
响应	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;pin&gt;: IO 管脚号</li> <li>• &lt;mode&gt;: IO 工作模式</li> <li>• &lt;pull-up&gt; <ul style="list-style-type: none"> <li>▸ 0: 不使能上拉</li> <li>▸ 1: 使能上拉</li> </ul> </li> </ul>
注意	AT+SYSIO 系列指令的使用, 请参考 <a href="#">ESP8266 管脚清单</a> 。
示例	AT+SYSIOSETCFG=12,3,1 //设置 GPIO12 工作为 GPIO 模式

### 3.2.16. AT+SYSIOGETCFG—查询 IO 工作模式

设置指令	AT+SYSIOGETCFG=<pin>
响应	+SYSIOGETCFG:<pin>,<mode>,<pull-up> OK



参数说明	<ul style="list-style-type: none"> <li>• &lt;pin&gt;: IO 管脚号</li> <li>• &lt;mode&gt;: IO 工作模式</li> <li>• &lt;pull-up&gt; <ul style="list-style-type: none"> <li>▸ 0: 不使能上拉</li> <li>▸ 1: 使能上拉</li> </ul> </li> </ul>
注意	AT+SYSIO 系列指令的使用, 请参考 <a href="#">ESP8266 管脚清单</a> 。

### 3.2.17. AT+SYSGPIODIR—设置 GPIO 工作为输入或输出

设置指令	AT+SYSGPIODIR=<pin>,<dir>
响应	<ul style="list-style-type: none"> <li>• 如果成功, 提示 OK</li> <li>• 如果 IO 管脚不处于 GPIO 模式, 则提示 NOT GPIO MODE ! ERROR</li> </ul>
参数说明	<ul style="list-style-type: none"> <li>• &lt;pin&gt;: GPIO 号</li> <li>• &lt;dir&gt;: <ul style="list-style-type: none"> <li>▸ 0: 设置 GPIO 为输入</li> <li>▸ 1: 设置 GPIO 为输出</li> </ul> </li> </ul>
注意	AT+SYSGPIO 系列指令的使用, 请参考 <a href="#">ESP8266 管脚清单</a> 。
示例	<pre>AT+SYSIOSETCFG=12,3,1 //设置 GPIO12 工作为 GPIO 模式 AT+SYSGPIODIR=12,0 //设置 GPIO12 为输入</pre>

### 3.2.18. AT+SYSGPIOWRITE—设置 GPIO 的输出电平

设置指令	AT+SYSGPIOWRITE=<pin>,<level>
响应	<ul style="list-style-type: none"> <li>• 如果成功, 提示 OK</li> <li>• 如果 IO 管脚不处于输出模式, 则提示 NOT OUTPUT ! ERROR</li> </ul>



参数说明	<ul style="list-style-type: none"> <li>• &lt;pin&gt;: GPIO 号</li> <li>• &lt;level&gt;: <ul style="list-style-type: none"> <li>▶ 0: 低电平</li> <li>▶ 1: 高电平</li> </ul> </li> </ul>
注意	AT+SYSGPIO 系列指令的使用, 请参考 <a href="#">ESP8266 管脚清单</a> 。
示例	<pre>AT+SYSIOSETCFG=12,3,1 //设置 GPIO12 工作为 GPIO 模式 AT+SYSGPIODIR=12,1 //设置 GPIO12 为输出 AT+SYSGPIOWRITE=12,1 //设置 GPIO12 输出高电平</pre>

### 3.2.19. AT+SYSGPIOREAD—读取 GPIO 的电平状态

设置指令	AT+SYSGPIOREAD=<pin>
响应	<ul style="list-style-type: none"> <li>• 如果成功, 返回 +SYSGPIOREAD:&lt;pin&gt;,&lt;dir&gt;,&lt;level&gt; OK</li> <li>• 如果 IO 管脚不处于输出模式, 则提示 NOT GPIO MODE ! ERROR</li> </ul>
参数说明	<ul style="list-style-type: none"> <li>• &lt;pin&gt;: GPIO 号</li> <li>• &lt;dir&gt;: <ul style="list-style-type: none"> <li>▶ 0: 设置 GPIO 为输入</li> <li>▶ 1: 设置 GPIO 为输出</li> </ul> </li> <li>• &lt;level&gt;: <ul style="list-style-type: none"> <li>▶ 0: 低电平</li> <li>▶ 1: 高电平</li> </ul> </li> </ul>
注意	AT+SYSGPIO 系列指令的使用, 请参考 <a href="#">ESP8266 管脚清单</a> 。
示例	<pre>AT+SYSIOSETCFG=12,3,1 //设置 GPIO12 工作为 GPIO 模式 AT+SYSGPIODIR=12,0 //设置 GPIO12 为输入 AT+SYSGPIOREAD=12</pre>



## 3.2.20. AT+SYMSMSG\_CUR—设置当前系统消息，不保存到 flash

设置指令	AT+SYMSMSG_CUR=<n>
响应	OK
参数说明	<p>&lt;n&gt;:</p> <ul style="list-style-type: none"> <li>bit0: 控制退出透传的消息 <ul style="list-style-type: none"> <li>若 bit0 为 0, 退出透传时, 无提示消息; 默认为 0</li> <li>若 bit0 为 1, 退出透传时, 提示消息 +QUIT // Quit transparent transmission</li> </ul> </li> <li>bit1: 控制建立网络连接的消息 <ul style="list-style-type: none"> <li>若 bit1 为 0, 网络连接建立时, 提示消息 &lt;Link_ID&gt;,CONNECT; 默认为 0</li> <li>若 bit1 为 1, 网络连接建立时, 提示消息 +LINK_CONN:&lt;status_type&gt;,&lt;link_id&gt;,"UDP/TCP/SSL",&lt;c/s&gt;,&lt;remote_ip&gt;,&lt;remote_port&gt;,&lt;local_port&gt;; <ul style="list-style-type: none"> <li>&lt;status_type&gt; 为 0 表示连接成功, 为 1 表示连接失败;</li> <li>&lt;c/s&gt; 为 0 表示 ESP 作为 client, 为 1 表示 ESP 作为 server。</li> </ul> </li> </ul> </li> </ul>
注意	本设置不保存到 flash 中, 重新上电后失效。
示例	AT+SYMSMSG_CUR=3

## 3.2.21. AT+SYMSMSG\_DEF—设置默认系统消息，保存到 flash

设置指令	AT+SYMSMSG_DEF=<n>
响应	OK
参数说明	<p>&lt;n&gt;:</p> <ul style="list-style-type: none"> <li>bit0: 控制退出透传的消息 <ul style="list-style-type: none"> <li>若 bit0 为 0, 退出透传时, 无提示消息; 默认为 0</li> <li>若 bit0 为 1, 退出透传时, 提示消息 +QUIT // Quit transparent transmission</li> </ul> </li> <li>bit1: 控制建立网络连接的消息 <ul style="list-style-type: none"> <li>若 bit1 为 0, 网络连接建立时, 提示消息 &lt;Link_ID&gt;, CONNECT; 默认为 0</li> <li>若 bit1 为 1, 网络连接建立时, 提示消息 +LINK_CONN:&lt;status_type&gt;,&lt;link_id&gt;,"UDP/TCP/SSL",&lt;c/s&gt;,&lt;remote_ip&gt;,&lt;remote_port&gt;,&lt;local_port&gt;; <ul style="list-style-type: none"> <li>&lt;status_type&gt; 为 0 表示连接成功, 为 1 表示连接失败;</li> <li>&lt;c/s&gt; 为 0 表示 ESP 作为 client, 为 1 表示 ESP 作为 server。</li> </ul> </li> </ul> </li> </ul>
注意	本设置保存到 flash 用户参数区, 重新上电后仍然有效。
示例	AT+SYMSMSG_DEF=3





# 4. 基础 AT 指令

## 4.1. 基础 Wi-Fi 功能 AT 指令一览表

指令	说明
AT+CWMODE_CUR	设置 Wi-Fi 模式 (STA/AP/STA+AP), 不保存到 Flash
AT+CWMODE_DEF	设置 Wi-Fi 模式 (STA/AP/STA+AP), 保存到 Flash
AT+CWJAP_CUR	连接 AP, 不保存到 Flash
AT+CWJAP_DEF	连接 AP, 保存到 Flash
AT+CWLAPOPT	设置 AT+CWLAP 指令扫描结果的属性
AT+CWLAP	扫描附近的 AP 信息
AT+CWQAP	与 AP 断开连接
AT+CWSAP_CUR	设置 ESP8266 SoftAP 配置, 不保存到 Flash
AT+CWSAP_DEF	设置 ESP8266 SoftAP 配置, 保存到 Flash
AT+CWLIF	获取连接到 ESP8266 SoftAP 的 station 的信息
AT+CWDHCP_CUR	设置 DHCP, 不保存到 Flash
AT+CWDHCP_DEF	设置 DHCP, 保存到 Flash
AT+CWDHCPS_CUR	设置 ESP8266 SoftAP DHCP 分配的 IP 范围, 不保存到 Flash
AT+CWDHCPS_DEF	设置 ESP8266 SoftAP DHCP 分配的 IP 范围, 保存到 Flash
AT+CWAUTOCONN	设置上电时是否自动连接 AP
AT+CIPSTAMAC_CUR	设置 ESP8266 Station 的 MAC 地址, 不保存到 Flash
AT+CIPSTAMAC_DEF	设置 ESP8266 Station 的 MAC 地址, 保存到 Flash
AT+CIPAPMAC_CUR	设置 ESP8266 SoftAP 的 MAC 地址, 不保存到 Flash
AT+CIPAPMAC_DEF	设置 ESP8266 SoftAP 的 MAC 地址, 保存到 Flash
AT+CIPSTA_CUR	设置 ESP8266 Station 的 IP 地址, 不保存到 Flash
AT+CIPSTA_DEF	设置 ESP8266 Station 的 IP 地址, 保存到 Flash
AT+CIPAP_CUR	设置 ESP8266 SoftAP 的 IP 地址, 不保存到 Flash



AT+CIPAP_DEF	设置 ESP8266 SoftAP 的 IP 地址，保存到 Flash
AT+CWSTARTSMART	开始 SmartConfig
AT+CWSTOPSMART	停止 SmartConfig
AT+CWSTARTDISCOVER	开启可被局域网内的微信探测的模式
AT+CWSTOPDISCOVER	关闭可被局域网内的微信探测的模式
AT+WPS	设置 WPS 功能
AT+MDNS	设置 MDNS 功能
AT+CWHOSTNAME	设置 ESP8266 Station 的主机名称
AT+CWCOUNTRY_CUR	设置 ESP8266 当前 WiFi 国家码
AT+CWCOUNTRY_DEF	设置 ESP8266 默认 WiFi 国家码



## 4.2. 基础 Wi-Fi 功能 AT 指令描述

### 4.2.1. AT+CWMODE\_CUR—设置当前 Wi-Fi 模式，不保存到 Flash

指令	测试指令： AT+CWMODE_CUR=?	查询指令： AT+CWMODE_CUR? 功能：查询 ESP8266 当前 Wi-Fi 模式。	设置指令： AT+CWMODE_CUR=<mode> 功能：设置 ESP8266 当前 Wi-Fi 模式。
响应	+CWMODE_CUR:<mode> 取值列表 OK	+CWMODE_CUR:<mode> OK	OK
参数说明	<mode>： <ul style="list-style-type: none"> <li>▶ 1: Station 模式</li> <li>▶ 2: SoftAP 模式</li> <li>▶ 3: SoftAP+Station 模式</li> </ul>		
注意	本设置不保存到 Flash。		
示例	AT+CWMODE_CUR=3		

### 4.2.2. AT+CWMODE\_DEF—设置当前 Wi-Fi 模式并保存到 Flash

指令	测试指令： AT+CWMODE_DEF=?	查询指令： AT+CWMODE_DEF? 功能：查询 ESP8266 当前 Wi-Fi 模式。	设置指令： AT+CWMODE_DEF=<mode> 功能：设置 ESP8266 当前 Wi-Fi 模式。
响应	+CWMODE_DEF:<mode> 取值列表 OK	+CWMODE_DEF:<mode> OK	OK
参数说明	<mode>： <ul style="list-style-type: none"> <li>▶ 1: Station 模式</li> <li>▶ 2: SoftAP 模式</li> <li>▶ 3: SoftAP+Station 模式</li> </ul>		
注意	本设置保存到 Flash system parameter 区域。		
示例	AT+CWMODE_DEF=3		



## 4.2.3. AT+CWJAP\_CUR—临时连接 AP

指令	查询指令： AT+CWJAP_CUR? 功能：查询 ESP8266 Station 已连接的 AP 信息。	设置指令： AT+CWJAP_CUR=<ssid>,<pwd>,[<bssid>][,<pci_en>] 功能：设置 ESP8266 Station 需连接的 AP。
响应	+CWJAP_CUR:<ssid>,<bssid>,<channel>,<rssi> > OK	OK 或者 +CWJAP_CUR:<error code>  FAIL
参数说明	<ul style="list-style-type: none"> <li>• &lt;ssid&gt;: 字符串参数, AP 的 SSID</li> <li>• &lt;bssid&gt;: AP 的 MAC 地址</li> <li>• &lt;channel&gt;: 信道号</li> <li>• &lt;rssi&gt;: 信号强度</li> </ul>	<ul style="list-style-type: none"> <li>• &lt;ssid&gt;: 目标 AP 的 SSID</li> <li>• &lt;pwd&gt;: 密码最长 64 字节 ASCII</li> <li>• [&lt;bssid&gt;]: 选填参数, 目标 AP 的 MAC 地址, 一般用于有多个 SSID 相同的 AP 的情况</li> <li>• [&lt;pci_en&gt;]: 选填参数, 不允许连接 WEP 和 open 的路由器, 可用于 PCI 认证</li> <li>• &lt;error code&gt;: (仅供参考, 并不可靠)             <ul style="list-style-type: none"> <li>▶ 1: 连接超时</li> <li>▶ 2: 密码错误</li> <li>▶ 3: 找不到目标 AP</li> <li>▶ 4: 连接失败</li> </ul> </li> </ul> <p>参数设置需要开启 Station 模式, 若 SSID 或者 password 中含有特殊符号, 例如, 或者 “ 或者 \ 时, 需要进行转义, 其它字符转义无效。</p>
注意	本设置不保存到 Flash。	
示例	AT+CWJAP_CUR="abc","0123456789" 例如, 目标 AP 的 SSID 为 "ab\,c", password 为 "0123456789\", 则指令如下: AT+CWJAP_CUR="ab\\,c","0123456789\\"" 如果有多个 AP 的 SSID 均为 "abc", 可通过 BSSID 确定目标 AP: AT+CWJAP_CUR="abc","0123456789","ca:d7:19:d8:a6:44"	



## 4.2.4. AT+CWJAP\_DEF—连接 AP，保存到 Flash

指令	查询指令： AT+CWJAP_DEF? 功能：查询 ESP8266 Station 已连接的 AP 信息。	设置指令： AT+CWJAP_DEF=<ssid>,<pwd>[,<bssid>][,<pci_en>] 功能：设置 ESP8266 Station 需连接的 AP。
响应	+CWJAP_DEF:<ssid>,<bssid>,<channel>,<rs si> OK	OK 或者 +CWJAP_DEF:<error code> FAIL
参数说明	<ul style="list-style-type: none"> <li>• &lt;ssid&gt;: 字符串参数，AP 的 SSID</li> <li>• &lt;bssid&gt;: AP 的 MAC 地址</li> <li>• &lt;channel&gt;: 信道号</li> <li>• &lt;rsi&gt;: 信号强度</li> </ul>	<ul style="list-style-type: none"> <li>• &lt;ssid&gt;: 目标 AP 的 SSID</li> <li>• &lt;pwd&gt;: 密码最长 64 字节 ASCII</li> <li>• [&lt;bssid&gt;]: 目标 AP 的 MAC 地址，一般用于有多个 SSID 相同的 AP 的情况</li> <li>• [&lt;pci_en&gt;]: 选填参数，不允许连接 WEP 和 open 的路由器，可用于 PCI 认证</li> <li>• &lt;error code&gt;: (仅供参考，并不可靠)             <ul style="list-style-type: none"> <li>▶ 1: 连接超时</li> <li>▶ 2: 密码错误</li> <li>▶ 3: 找不到目标 AP</li> <li>▶ 4: 连接失败</li> </ul> </li> </ul> <p>参数设置需要开启 Station 模式，若 SSID 或者 password 中含有特殊符号，例如，或者“或者\时，需要进行转义，其它字符转义无效。</p>
注意	本设置保存到 Flash 系统参数区。	
示例	AT+CWJAP_DEF="abc","0123456789" 例如，目标 AP 的 SSID 为 "ab\,c"，password 为 "0123456789\"，则指令如下： AT+CWJAP_DEF="ab\\,c","0123456789\"" 如果有多个 AP 的 SSID 均为 "abc"，可通过 BSSID 确定目标 AP： AT+CWJAP_DEF="abc","0123456789","ca:d7:19:d8:a6:44"	



## 4.2.5. AT+CWLAPOPT—设置 CWLAP 指令的属性

设置指令	AT+CWLAPOPT=<sort_enable>,<mask>
响应	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;sort_enable&gt;: 指令 AT+CWLAP 的扫描结果是否按照信号强度 RSSI 值排序: <ul style="list-style-type: none"> <li>▶ 0: 不排序</li> <li>▶ 1: 为根据 RSSI 排序</li> </ul> </li> <li>• &lt;mask&gt;: 对应 bit 若为 1, 则指令 AT+CWLAP 的扫描结果显示相关属性, 对应 bit 若为 0, 则不显示。具体如下: <ul style="list-style-type: none"> <li>▶ bit 0: 设置 AT+CWLAP 的扫描结果是否显示 &lt;ecn&gt;</li> <li>▶ bit 1: 设置 AT+CWLAP 的扫描结果是否显示 &lt;ssid&gt;</li> <li>▶ bit 2: 设置 AT+CWLAP 的扫描结果是否显示 &lt;rssi&gt;</li> <li>▶ bit 3: 设置 AT+CWLAP 的扫描结果是否显示 &lt;mac&gt;</li> <li>▶ bit 4: 设置 AT+CWLAP 的扫描结果是否显示 &lt;channel&gt;</li> <li>▶ bit 5: 设置 AT+CWLAP 的扫描结果是否显示 &lt;freq offset&gt;</li> <li>▶ bit 6: 设置 AT+CWLAP 的扫描结果是否显示 &lt;freq calibration&gt;</li> <li>▶ bit 7: 设置 AT+CWLAP 的扫描结果是否显示 &lt;pairwise_cipher&gt;</li> <li>▶ bit 8: 设置 AT+CWLAP 的扫描结果是否显示 &lt;group_cipher&gt;</li> <li>▶ bit 9: 设置 AT+CWLAP 的扫描结果是否显示 &lt;bgn&gt;</li> <li>▶ bit 10: 设置 AT+CWLAP 的扫描结果是否显示 &lt;wps&gt;</li> </ul> </li> </ul>
示例	<p>AT+CWLAPOPT=1,2047</p> <p>第一个参数为 1, 表示后续如果使用 AT+CWLAP 指令, 扫描结果将按照信号强度 RSSI 值排序;</p> <p>第二个参数为 2047, 即 0x7FF, 表示 &lt;mask&gt; 的相关 bit 全部置为 1, 后续如果使用 AT+CWLAP 指令, 扫描结果将显示所有参数。</p>



## 4.2.6. AT+CWLAP—扫描当前可用的 AP

指令	设置指令： AT+CWLAP=<ssid>[,<mac>,<channel>,<scan_type>,<scan_time_min>,<scan_time_max>] 功能：列出符合特定条件的 AP。	执行指令： AT+CWLAP 功能：列出当前可用的 AP。
响应	+CWLAP:<ecn>,<ssid>,<rssi>,<mac>,<channel>,<freq offset>,<freq cali>,<pairwise_cipher>,<group_cipher>,<bgn>,<wps> OK	+CWLAP:<ecn>,<ssid>,<rssi>,<mac>,<channel>,<freq offset>,<freq cali>,<pairwise_cipher>,<group_cipher>,<bgn>,<wps> OK
参数说明	<ul style="list-style-type: none"> <li>• [&lt;scan_type&gt;]: 选填参数           <ul style="list-style-type: none"> <li>▶ 0: active scan</li> <li>▶ 1: passive scan</li> </ul> </li> <li>• [&lt;scan_time_min&gt;]: 选填参数, 单位 ms, 范围 [0,1500]           <ul style="list-style-type: none"> <li>▶ 对于 active scan, 此参数为每个信道的最小扫描时间; 默认值为 0;</li> <li>▶ 对于 passive scan, 此参数无效</li> </ul> </li> <li>• [&lt;scan_time_max&gt;]: 选填参数, 单位 ms, 范围 [0,1500]           <ul style="list-style-type: none"> <li>▶ 对于 active scan, 此参数为每个信道的最大扫描时间; 当设置为 0 时, 采用默认值 120ms;</li> <li>▶ 对于 passive scan, 此参数为每个信道扫描的时间; 默认值为 360ms</li> </ul> </li> <li>• &lt;ecn&gt;: 加密方式           <ul style="list-style-type: none"> <li>▶ 0: OPEN</li> <li>▶ 1: WEP</li> <li>▶ 2: WPA_PSK</li> <li>▶ 3: WPA2_PSK</li> <li>▶ 4: WPA_WPA2_PSK</li> <li>▶ 5: WPA2_Enterprise (目前 AT 不支持连接这种加密 AP)</li> </ul> </li> <li>• &lt;ssid&gt;: 字符串参数, AP 的 SSID</li> <li>• &lt;rssi&gt;: 信号强度</li> <li>• &lt;mac&gt;: 字符串参数, AP 的 MAC 地址</li> <li>• &lt;channel&gt;: 信道号</li> <li>• &lt;freq offset&gt;: AP 频偏, 单位: kHz。此数值除以 2.4, 可得到 ppm 值</li> <li>• &lt;freq cali&gt;: 频偏校准值</li> <li>• &lt;pairwise_cipher&gt;:           <ul style="list-style-type: none"> <li>▶ 0: CIPHER_NONE</li> <li>▶ 1: CIPHER_WEP40</li> <li>▶ 2: CIPHER_WEP104</li> <li>▶ 3: CIPHER_TKIP</li> <li>▶ 4: CIPHER_CCMP</li> <li>▶ 5: CIPHER_TKIP_CCMP</li> <li>▶ 6: CIPHER_UNKNOWN</li> </ul> </li> <li>• &lt;group_cipher&gt;: 定义与 &lt;pairwise_cipher&gt; 相同</li> <li>• &lt;bgn&gt;:           <ul style="list-style-type: none"> <li>▶ bit0 代表 b 模式; bit1 代表 g 模式; bit2 代表 n 模式</li> <li>▶ 若对应 bit 为 1, 表示该模式使能; 若对应 bit 为 0, 则该模式未使能。</li> </ul> </li> <li>• &lt;wps&gt;: 0, WPS 未使能; 1, WPS 使能</li> </ul>	



注意	<ul style="list-style-type: none"><li>• scan_time_min 不能大于 scan_time_max, 即要求 scan_time_max &gt;= scan_time_min。</li><li>• 若 scan_time_min 和 scan_time_max 设置均为 0, 则采用默认值, 默认值具体如下:<ul style="list-style-type: none"><li>▸ 对于 active scan, 每个信道的最小扫描时间是 0, 最大扫描时间是 120ms</li><li>▸ 对于 passive scan, 每个信道扫描的时间是 360ms</li></ul></li></ul>
示例	AT+CWLAP="Wi-Fi", "ca:d7:19:d8:a6:44", 6, 0, 50, 120, 或者查找指定 SSID 的 AP: AT+CWLAP="Wi-Fi" 或者使能 passive scan 模式: AT+CWLAP=,,,1,,

#### 4.2.7. AT+CWQAP—断开与 AP 的连接

执行指令	AT+CWQAP
响应	OK
参数说明	-





## 4.2.8. AT+CWSAP\_CUR—配置 ESP8266 SoftAP 当前参数

指令	查询指令： AT+CWSAP_CUR? 功能：查询 ESP8266 SoftAP 的配置参数。	设置指令： AT+CWSAP_CUR=<ssid>,<pwd>,<chl>,<ecn>[,<max conn>][,<ssid hidden>] 功能：设置 ESP8266 SoftAP 的配置参数。
响应	+CWSAP_CUR:<ssid>,<pwd>,<chl>,<ecn>,<max conn>,<ssid hidden>	OK 或 ERROR
参数说明	<ul style="list-style-type: none"> <li>• &lt;ssid&gt;: 字符串参数, 接入点名称</li> <li>• &lt;pwd&gt;: 字符串参数, 密码长度范围: 8 ~ 64 字节 ASCII</li> <li>• &lt;chl&gt;: 通道号</li> <li>• &lt;ecn&gt;: 加密方式, 不支持 WEP             <ul style="list-style-type: none"> <li>▸ 0: OPEN</li> <li>▸ 2: WPA_PSK</li> <li>▸ 3: WPA2_PSK</li> <li>▸ 4: WPA_WPA2_PSK</li> </ul> </li> <li>• [&lt;max conn&gt;] (选填参数): 允许连入 ESP8266 SoftAP 的最多 Station 数目, 取值范围 [1, 8]。</li> <li>• [&lt;ssid hidden&gt;] (选填参数): 默认为 0, 开启广播 ESP8266 SoftAP SSID。             <ul style="list-style-type: none"> <li>▸ 0: 广播 SSID</li> <li>▸ 1: 不广播 SSID</li> </ul> </li> </ul>	同左。 <b>⚠ 注意:</b> 指令只有在 SoftAP 模式开启后有效。
注意	本设置不保存到 Flash。	
示例	AT+CWSAP_CUR="ESP8266","1234567890",5,3	

## 4.2.9. AT+CWSAP\_DEF—配置 ESP8266 SoftAP 当前参数, 保存到 Flash

指令	查询指令： AT+CWSAP_DEF? 功能：查询 ESP8266 SoftAP 的配置参数。	设置指令： AT+CWSAP_DEF=<ssid>,<pwd>,<chl>,<ecn>[,<max conn>][,<ssid hidden>] 功能：设置 ESP8266 SoftAP 的配置参数。
响应	+CWSAP_DEF:<ssid>,<pwd>,<chl>,<ecn>,<max conn>,<ssid hidden>	OK 或 ERROR



参数说明	<ul style="list-style-type: none"><li>• &lt;ssid&gt;: 字符串参数, 接入点名称</li><li>• &lt;pwd&gt;: 字符串参数, 密码长度范围: 8 ~ 64 字节 ASCII</li><li>• &lt;chl&gt;: 通道号</li><li>• &lt;ecn&gt;: 加密方式, 不支持 WEP<ul style="list-style-type: none"><li>▶ 0: OPEN</li><li>▶ 2: WPA_PSK</li><li>▶ 3: WPA2_PSK</li><li>▶ 4: WPA_WPA2_PSK</li></ul></li><li>• [&lt;max conn&gt;] (选填参数): 允许连入 ESP8266 SoftAP 的最多 Station 数目, 取值范围 [1, 8]。</li><li>• [&lt;ssid hidden&gt;] (选填参数): 默认为 0, 开启广播 ESP8266 SoftAP SSID。<ul style="list-style-type: none"><li>▶ 0: 广播 SSID</li><li>▶ 1: 不广播 SSID</li></ul></li></ul>	同左。 <b>⚠ 注意:</b> 指令只有在 SoftAP 模式开启后有效。
注意	本设置保存到 Flash system parameter 区域。	
示例	AT+CWSAP_DEF="ESP8266", "1234567890", 5, 3	



## 4.2.10. AT+CWLIF—查询连接到 ESP8266 SoftAP 的 Station 信息

执行指令	AT+CWLIF
响应	+CWLIF:<ip addr>,<mac> OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;ip addr&gt;: 连接到 ESP8266 SoftAP 的 Station IP 地址</li> <li>• &lt;mac&gt;: 连接到 ESP8266 SoftAP 的 Station MAC 地址</li> </ul>
注意	本指令无法查询静态 IP, 仅支持在 ESP8266 SoftAP 和连入的 Station DHCP 均使能的情况下有效。

## 4.2.11. AT+CWDHCP\_CUR—设置 DHCP, 不保存到 Flash

指令	查询指令: AT+CWDHCP_CUR?	设置指令: AT+CWDHCP_CUR=<mode>,<en> 功能: 设置 DHCP。
响应	DHCP 是否使能	OK
参数说明	<ul style="list-style-type: none"> <li>• Bit0:             <ul style="list-style-type: none"> <li>▸ 0: SoftAP DHCP 关闭</li> <li>▸ 1: SoftAP DHCP 开启</li> </ul> </li> <li>• Bit1:             <ul style="list-style-type: none"> <li>▸ 0: Station DHCP 关闭</li> <li>▸ 1: Station DHCP 开启</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• &lt;mode&gt;:             <ul style="list-style-type: none"> <li>▸ 0: 设置 ESP8266 SoftAP</li> <li>▸ 1: 设置 ESP8266 Station</li> <li>▸ 2: 设置 ESP8266 SoftAP 和 Station</li> </ul> </li> <li>• &lt;en&gt;:             <ul style="list-style-type: none"> <li>▸ 0: 关闭 DHCP</li> <li>▸ 1: 开启 DHCP</li> </ul> </li> </ul>
注意	<ul style="list-style-type: none"> <li>• 本设置不保存到 Flash。</li> <li>• 本设置指令与设置静态 IP 的指令 (AT+CIPSTA 系列和 AT+CIPAP 系列) 互相影响:             <ul style="list-style-type: none"> <li>▸ 设置使能 DHCP, 则静态 IP 无效;</li> <li>▸ 设置静态 IP, 则 DHCP 关闭;</li> <li>▸ 以最后的设置为准。</li> </ul> </li> </ul>	
示例	AT+CWDHCP_CUR=0,1	



## 4.2.12. AT+CWDHCP\_DEF—设置 DHCP，保存到 Flash

指令	查询指令： AT+CWDHCP_DEF?	设置指令： AT+CWDHCP_DEF=<mode>,<en> 功能：设置 DHCP。
响应	DHCP 是否使能	OK
参数说明	<ul style="list-style-type: none"><li>• Bit0:<ul style="list-style-type: none"><li>▶ 0: SoftAP DHCP 关闭</li><li>▶ 1: SoftAP DHCP 开启</li></ul></li><li>• Bit1:<ul style="list-style-type: none"><li>▶ 0: Station DHCP 关闭</li><li>▶ 1: Station DHCP 开启</li></ul></li></ul>	<ul style="list-style-type: none"><li>• &lt;mode&gt;:<ul style="list-style-type: none"><li>▶ 0: 设置 ESP8266 SoftAP</li><li>▶ 1: 设置 ESP8266 Station</li><li>▶ 2: 设置 ESP8266 SoftAP 和 Station</li></ul></li><li>• &lt;en&gt;:<ul style="list-style-type: none"><li>▶ 0: 关闭 DHCP</li><li>▶ 1: 开启 DHCP</li></ul></li></ul>
注意	<ul style="list-style-type: none"><li>• 本设置保存到 Flash 用户参数区。</li><li>• 本设置指令与设置静态 IP 的指令（AT+CIPSTA 系列和 AT+CIPAP 系列）互相影响：<ul style="list-style-type: none"><li>▶ 设置使能 DHCP，则静态 IP 无效；</li><li>▶ 设置静态 IP，则 DHCP 关闭；</li><li>▶ 以最后的设置为准。</li></ul></li></ul>	
示例	AT+CWDHCP_DEF=0,1	



## 4.2.13. AT+CWDHCPS\_CUR—设置 ESP8266 SoftAP DHCP 分配的 IP 范围，不保存到 Flash

指令	查询指令： AT+CWDHCPS_CUR?	设置指令： AT+CWDHCPS_CUR=<enable>,<lease time>,<start IP>,<end IP>  功能：设置 ESP8266 SoftAP DHCP 服务器分配的 IP 范围。
响应	+CWDHCPS_CUR=<lease time>,<start IP>,<end IP>	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;enable&gt;: <ul style="list-style-type: none"> <li>▸ 0: 清除设置 IP 范围，恢复默认值，后续参数无需填写</li> <li>▸ 1: 使能设置 IP 范围，后续参数必须填写</li> </ul> </li> <li>• &lt;lease time&gt;: 租约时间，单位：分钟，取值范围 [1, 2880]</li> <li>• &lt;start IP&gt;: DHCP 服务器 IP 池的起始 IP</li> <li>• &lt;end IP&gt;: DHCP 服务器 IP 池的结束 IP</li> </ul>	
注意	<ul style="list-style-type: none"> <li>• 本设置不保存到 Flash。</li> <li>• 本指令必须在 ESP8266 SoftAP 模式使能，且开启 DHCP 的情况下使用，设置的 IP 范围必须与 ESP8266 SoftAP 在同一网段。</li> </ul>	
示例	AT+CWDHCPS_CUR=1,3,"192.168.4.10","192.168.4.15" 或者 AT+CWDHCPS_CUR=0 //清除设置，恢复默认值	

## 4.2.14. AT+CWDHCPS\_DEF—设置 ESP8266 SoftAP DHCP 分配的 IP 范围，保存到 Flash

指令	查询指令： AT+CWDHCPS_DEF?	设置指令： AT+CWDHCPS_DEF=<enable>,<lease time>,<start IP>,<end IP>  功能：设置 ESP8266 SoftAP DHCP 服务器分配的 IP 范围。
响应	+CWDHCPS_DEF=<lease time>,<start IP>,<end IP>	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;enable&gt;: <ul style="list-style-type: none"> <li>▸ 0: 清除设置 IP 范围，恢复默认值，后续参数无需填写</li> <li>▸ 1: 使能设置 IP 范围，后续参数必须填写</li> </ul> </li> <li>• &lt;lease time&gt;: 租约时间，单位：分钟，取值范围 [1, 2880]</li> <li>• &lt;start IP&gt;: DHCP 服务器 IP 池的起始 IP</li> <li>• &lt;end IP&gt;: DHCP 服务器 IP 池的结束 IP</li> </ul>	



注意	<ul style="list-style-type: none"> <li>本设置保存到 Flash 用户参数区。</li> <li>本指令必须在 ESP8266 SoftAP 模式使能，且开启 DHCP 的情况下使用，设置的 IP 范围必须与 ESP8266 SoftAP 在同一网段。</li> </ul>
示例	<pre>AT+CWDHCPS_DEF=1,3,"192.168.4.10","192.168.4.15"</pre> <p>或者</p> <pre>AT+CWDHCPS_DEF=0 //清除设置，恢复默认值</pre>

#### 4.2.15. AT+CWAUTOCONN—上电是否自动连接 AP

设置指令	AT+CWAUTOCONN=<enable>
响应	OK
参数说明	<p>&lt;enable&gt;:</p> <ul style="list-style-type: none"> <li>0: 上电不自动连接 AP</li> <li>1: 上电自动连接 AP</li> </ul> <p>ESP8266 Station 默认上电自动连接 AP。</p>
注意	本设置保存到 Flash 系统参数区。
示例	AT+CWAUTOCONN=1

#### 4.2.16. AT+CIPSTAMAC\_CUR—设置 ESP8266 Station 当前 MAC 地址，不保存到 Flash

指令	<p>查询指令:</p> <pre>AT+CIPSTAMAC_CUR?</pre>	<p>设置指令:</p> <pre>AT+CIPSTAMAC_CUR=&lt;mac&gt;</pre> <p>功能: 设置 ESP8266 Station 的 MAC 地址。</p>
响应	<pre>+CIPSTAMAC_CUR:&lt;mac&gt;</pre> <p>OK</p>	OK
参数说明	<mac>: 字符串参数，ESP8266 Station 的 MAC 地址	
注意	<ul style="list-style-type: none"> <li>本设置不保存到 Flash。</li> <li>ESP8266 SoftAP 和 Station 的 MAC 地址并不相同，请勿将其设置为同一 MAC 地址。</li> <li>ESP8266 MAC 地址第一个字节的 bit 0 不能为 1，例如，MAC 地址可以为 "18:..." 但不能为 "15:..."。</li> </ul>	
示例	AT+CIPSTAMAC_CUR="18:fe:35:98:d3:7b"	



## 4.2.17. AT+CIPSTAMAC\_DEF—设置 ESP8266 Station 当前 MAC 地址，保存到 Flash

指令	查询指令： AT+CIPSTAMAC_DEF?	设置指令： AT+CIPSTAMAC_DEF=<mac> 功能：设置 ESP8266 Station 的 MAC 地址。
响应	+CIPSTAMAC_DEF:<mac> OK	OK
参数说明	<mac>：字符串参数，ESP8266 Station 的 MAC 地址	
注意	<ul style="list-style-type: none"> <li>本设置保存到 Flash 用户参数区。</li> <li>ESP8266 SoftAP 和 Station 的 MAC 地址并不相同，请勿将其设置为同一 MAC 地址。</li> <li>ESP8266 MAC 地址第一个字节的 bit 0 不能为 1，例如，MAC 地址可以为 "18:..." 但不能为 "15:..."。</li> </ul>	
示例	AT+CIPSTAMAC_DEF="18:fe:35:98:d3:7b"	

## 4.2.18. AT+CIPAPMAC\_CUR—设置 ESP8266 SoftAP 当前 MAC 地址，不保存到 Flash

指令	查询指令： AT+CIPAPMAC_CUR? 功能：查询 ESP8266 SoftAP 的 MAC 地址。	设置指令： AT+CIPAPMAC_CUR=<mac> 功能：设置 ESP8266 SoftAP 的 MAC 地址。
响应	+CIPAPMAC_CUR:<mac> OK	OK
参数说明	<mac>：字符串参数，ESP8266 SoftAP 的 MAC 地址	
注意	<ul style="list-style-type: none"> <li>本设置不保存到 Flash。</li> <li>ESP8266 SoftAP 和 Station 的 MAC 地址并不相同，请勿将其设置为同一 MAC 地址。</li> <li>ESP8266 MAC 地址第一个字节的 bit 0 不能为 1，例如，MAC 地址可以为 "18:..." 但不能为 "15:..."。</li> </ul>	
示例	AT+CIPAPMAC_CUR="1a:fe:36:97:d5:7b"	

## 4.2.19. AT+CIPAPMAC\_DEF—设置 ESP8266 SoftAP 默认 MAC 地址，保存到 Flash

指令	查询指令： AT+CIPAPMAC_DEF? 功能：查询 ESP8266 SoftAP 的 MAC 地址。	设置指令： AT+CIPAPMAC_DEF=<mac> 功能：设置 ESP8266 SoftAP 的 MAC 地址。
响应	+CIPAPMAC_DEF:<mac> OK	OK



参数说明	<mac>: 字符串参数, ESP8266 SoftAP 的 MAC 地址
注意	<ul style="list-style-type: none"> <li>• 本设置保存到 Flash 用户参数区。</li> <li>• ESP8266 SoftAP 和 Station 的 MAC 地址并不相同, 请勿将其设置为同一 MAC 地址。</li> <li>• ESP8266 MAC 地址第一个字节的 bit 0 不能为 1, 例如, MAC 地址可以为 "18:..." 但不能为 "15:..."。</li> </ul>
示例	AT+CIPAPMAC_DEF="1a:fe:36:97:d5:7b"

#### 4.2.20. AT+CIPSTA\_CUR—设置 ESP8266 Station 的 IP 地址, 不保存到 Flash

指令	查询指令: AT+CIPSTA_CUR? 功能: 查询 ESP8266 Station 的 IP 地址。	设置指令: AT+CIPSTA_CUR=<ip>[, <gateway>, <netmask>] 功能: 设置 ESP8266 Station 的 IP 地址。
响应	+CIPSTA_CUR:<ip> +CIPSTA_CUR:<gateway> +CIPSTA_CUR:<netmask> OK	OK
参数说明	<b>⚠ 注意:</b> ESP8266 Station IP 需连上 AP 后, 才可以查询。	<ul style="list-style-type: none"> <li>• &lt;ip&gt;: 字符串, ESP8266 Station 的 IP 地址</li> <li>• [&lt;gateway&gt;]: 网关</li> <li>• [&lt;netmask&gt;]: 子网掩码</li> </ul>
注意	<ul style="list-style-type: none"> <li>• 本设置不保存到 Flash。</li> <li>• 本设置指令与设置 DHCP 的指令 (AT+CWDHCP 系列) 互相影响:             <ul style="list-style-type: none"> <li>▸ 设置静态 IP, 则 DHCP 关闭;</li> <li>▸ 设置使能 DHCP, 则静态 IP 无效;</li> <li>▸ 以最后的设置为准。</li> </ul> </li> </ul>	
示例	AT+CIPSTA_CUR="192.168.6.100", "192.168.6.1", "255.255.255.0"	

#### 4.2.21. AT+CIPSTA\_DEF—设置 ESP8266 Station 的 IP 地址, 保存到 Flash

指令	查询指令: AT+CIPSTA_DEF? 功能: 查询 ESP8266 Station 的 IP 地址。	设置指令: AT+CIPSTA_DEF=<ip>[, <gateway>, <netmask>] 功能: 设置 ESP8266 Station 的 IP 地址。
响应	+CIPSTA_DEF:<ip> +CIPSTA_DEF:<gateway> +CIPSTA_DEF:<netmask> OK	OK





参数说明	<p><b>⚠ 注意：</b></p> <p>ESP8266 Station IP 需连上 AP 后，才可以查询。</p>	<ul style="list-style-type: none"> <li>• &lt;ip&gt;: 字符串, ESP8266 Station 的 IP 地址</li> <li>• [&lt;gateway&gt;]: 网关</li> <li>• [&lt;netmask&gt;]: 子网掩码</li> </ul>
注意	<ul style="list-style-type: none"> <li>• 本设置保存到 Flash 用户参数区。</li> <li>• 本设置指令与设置 DHCP 的指令 (AT+CWDHCP 系列) 互相影响: <ul style="list-style-type: none"> <li>▶ 设置静态 IP, 则 DHCP 关闭;</li> <li>▶ 设置使能 DHCP, 则静态 IP 无效;</li> <li>▶ 以最后的设置为准。</li> </ul> </li> </ul>	
示例	AT+CIPSTA_DEF="192.168.6.100", "192.168.6.1", "255.255.255.0"	

#### 4.2.22. AT+CIPAP\_CUR—设置 ESP8266 SoftAP 的 IP 地址，不保存到 Flash

指令	<p>查询指令:</p> <p>AT+CIPAP_CUR?</p> <p>功能: 查询 ESP8266 SoftAP 的 IP 地址。</p>	<p>设置指令:</p> <p>AT+CIPAP_CUR=&lt;ip&gt;[,&lt;gateway&gt;,&lt;netmask&gt;]</p> <p>功能: 设置 ESP8266 SoftAP 的 IP 地址。</p>
响应	<p>+CIPAP_CUR:&lt;ip&gt;</p> <p>+CIPAP_CUR:&lt;gateway&gt;</p> <p>+CIPAP_CUR:&lt;netmask&gt;</p> <p>OK</p>	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;ip&gt;: 字符串, ESP8266 SoftAP 的 IP 地址</li> <li>• [&lt;gateway&gt;]: 网关</li> <li>• [&lt;netmask&gt;]: 子网掩码</li> </ul>	
注意	<ul style="list-style-type: none"> <li>• 本设置不保存到 Flash。</li> <li>• 目前仅支持 C 类 IP 地址。</li> <li>• 本设置指令与设置 DHCP 的指令 (AT+CWDHCP 系列) 互相影响: <ul style="list-style-type: none"> <li>▶ 设置静态 IP, 则 DHCP 关闭;</li> <li>▶ 设置使能 DHCP, 则静态 IP 无效;</li> <li>▶ 以最后的设置为准。</li> </ul> </li> </ul>	
示例	AT+CIPAP_CUR="192.168.5.1", "192.168.5.1", "255.255.255.0"	

#### 4.2.23. AT+CIPAP\_DEF—设置 ESP8266 SoftAP 的 IP 地址，保存到 Flash

指令	<p>查询指令:</p> <p>AT+CIPAP_DEF?</p> <p>功能: 查询 ESP8266 SoftAP 的 IP 地址。</p>	<p>设置指令:</p> <p>AT+CIPAP_DEF=&lt;ip&gt;[,&lt;gateway&gt;,&lt;netmask&gt;]</p> <p>功能: 设置 ESP8266 SoftAP 的 IP 地址。</p>
----	---	---



响应	+CIPAP_DEF:<ip> +CIPAP_DEF:<gateway> +CIPAP_DEF:<netmask> OK	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;ip&gt;: 字符串, ESP8266 SoftAP 的 IP 地址</li> <li>• [&lt;gateway&gt;]: 网关</li> <li>• [&lt;netmask&gt;]: 子网掩码</li> </ul>	
注意	<ul style="list-style-type: none"> <li>• 本设置保存到 Flash 用户参数区。</li> <li>• 目前仅支持 C 类 IP 地址。</li> <li>• 本设置指令与设置 DHCP 的指令 (AT+CWDHCP 系列) 互相影响: <ul style="list-style-type: none"> <li>▶ 设置静态 IP, 则 DHCP 关闭;</li> <li>▶ 设置使能 DHCP, 则静态 IP 无效;</li> <li>▶ 以最后的设置为准。</li> </ul> </li> </ul>	
示例	AT+CIPAP_DEF="192.168.5.1", "192.168.5.1", "255.255.255.0"	

#### 4.2.24. AT+CWSTARTSMART—开启 SmartConfig

指令	执行指令: AT+CWSTARTSMART 功能: 开启 SmartConfig。 (SmartConfig 类型为 ESP-TOUCH+AirKiss)	设置指令: AT+CWSTARTSMART=<type> 功能: 开启某指定类型的 SmartConfig。
响应	OK	
参数说明	<type> : <ul style="list-style-type: none"> <li>▶ 1: ESP-TOUCH</li> <li>▶ 2: AirKiss</li> <li>▶ 3: ESP-TOUCH+AirKiss</li> </ul>	
说明	SmartConfig 连接过程中的提示信息如下: <pre> smartconfig type: &lt;type&gt; // AIRKISS, ESPTOUCH or UNKNOWN smart get wifi info // got SSID and password ssid:&lt;AP's SSID&gt; password:&lt;AP's password&gt; // ESP8266 will try to connect to the AP WIFI CONNECTED WIFI GOT IP smartconfig connected wifi // if the connection failed, it will prompt "smartconfig connect fail" </pre>	



注意	<ul style="list-style-type: none"> <li>• 用户可以参考 <a href="#">ESP-TOUCH 用户指南</a> 来了解 SmartConfig 的详细介绍。</li> <li>• 仅支持在 ESP8266 单 Station 模式下调用。</li> <li>• 消息 <code>smart get wifi info</code> 表示 SmartConfig 成功获取到 AP 信息，之后 ESP8266 尝试连接 AP，打印连接过程。</li> <li>• 消息 <code>smartconfig connected wifi</code> 表示成功连接到 AP，此时可以调用 <code>AT+CWSTOPSMART</code> 停止 SmartConfig 再执行其他指令。注意，在 SmartConfig 过程中请勿执行其他指令。</li> <li>• 从 AT_v1.0 开始，SmartConfig 可以自动获取协议类型，AirKiss 或者 ESP-TOUCH。</li> <li>• 用户如果无需 SmartConfig 功能，可以参考 <a href="#">章节 1.1</a> 自行重新编译 at，在 <a href="#">user_config.h</a> 中关闭 <code>CONFIG_AT_SMARTCONFIG_COMMAND_ENABLE</code>，从而减小 bin size，节省内存。</li> </ul>
示例	<pre>AT+CWMODE=1 AT+CWSTARTSMART</pre>

#### 4.2.25. AT+CWSTOPSMART—停止 SmartConfig

执行指令	AT+CWSTOPSMART
响应	OK
参数说明	-
注意	无论 SmartConfig 成功与否，都请调用 <code>AT+CWSTOPSMART</code> 释放快连占用的内存。
示例	AT+CWSTOPSMART

#### 4.2.26. AT+CWSTARTDISCOVER—开启可被局域网内微信探测模式

设置指令	<pre>AT+CWSTARTDISCOVER= &lt;WeChat number&gt;,&lt;dev_type&gt;,&lt;time&gt;</pre>
响应	OK
参数说明	<ul style="list-style-type: none"> <li>• <code>&lt;WeChat number&gt;</code>: 微信公众号，必须从微信获取。</li> <li>• <code>&lt;dev_type&gt;</code>: 设备类型，必须从微信获取。</li> <li>• <code>&lt;time&gt;</code>: 主动发包时间间隔，取值范围：0 ~ 24x3600，单位：秒。 <ul style="list-style-type: none"> <li>▶ 0: ESP8266 不主动向外发包，需要手机微信查询时才回复。</li> <li>▶ 其他值: ESP8266 主动发包的时间间隔，以便于局域网中的手机微信发现本设备。</li> </ul> </li> </ul>
注意	<ul style="list-style-type: none"> <li>• 可参考微信官网内网发现功能的介绍 <a href="http://iot.weixin.qq.com">http://iot.weixin.qq.com</a>。</li> <li>• 本指令需在 ESP8266 Station 连入局域网，获得 IP 地址后生效。</li> </ul>
示例	<pre>AT+CWSTARTDISCOVER="gh_9e2cff3dfa51","122475",10</pre>



## 4.2.27. AT+CWSTOPDISCOVER—关闭可被局域网内微信探测模式

执行指令	AT+CWSTOPDISCOVER
响应	OK
示例	AT+CWSTOPDISCOVER

## 4.2.28. AT+WPS—设置 WPS 功能

设置指令	AT+WPS=<enable>
响应	OK
参数说明	<p>&lt;enable&gt;:</p> <ul style="list-style-type: none"> <li>▶ 1: 开启 WPS</li> <li>▶ 0: 关闭 WPS</li> </ul>
注意	<ul style="list-style-type: none"> <li>• WPS 功能必须在 ESP8266 Station 使能的情况下调用。</li> <li>• WPS 不支持 WEP 加密方式。</li> </ul>
示例	<pre>AT+CWMODE=1 AT+WPS=1</pre>

## 4.2.29. AT+MDNS—设置 MDNS 功能

设置指令	AT+MDNS=<enable>,<hostname>,<server_name>,<server_port>
响应	<p>OK</p> <p>或</p> <p>opmode mismatch when mdns</p> <p>ERROR</p>
参数说明	<ul style="list-style-type: none"> <li>• &lt;enable&gt;: <ul style="list-style-type: none"> <li>▶ 1: 开启 MDNS 功能, 后续参数需要填写</li> <li>▶ 0: 关闭 MDNS 功能, 后续参数无需填写</li> </ul> </li> <li>• &lt;hostname&gt;: MDNS 主机名称</li> <li>• &lt;server_name&gt;: MDNS 服务器名称</li> <li>• &lt;server_port&gt;: MDNS 服务器端口</li> </ul>
注意	<ul style="list-style-type: none"> <li>• &lt;hostname&gt; 和 &lt;server_name&gt; 不能包含特殊字符 (例如 . 符号), 或者设置为协议名称 (例如不能定义为 http)。</li> <li>• ESP8266 SoftAP 模式暂时不支持 MDNS 功能。</li> </ul>



示例	AT+MDNS=1,"espressif","iot",8080
----	----------------------------------

#### 4.2.30. AT+CWHOSTNAME—设置 ESP8266 Station 的主机名称

指令	查询指令： AT+CWHOSTNAME? 功能：查询 ESP8266 Station 的主机名称。	设置指令： AT+CWHOSTNAME=<hostname> 功能：设置 ESP8266 Station 的主机名称。
响应	+CWHOSTNAME:<host name> OK 如果未使能 ESP8266 Station 模式，则返回 +CWHOSTNAME:<null> OK	如果成功，返回 OK 如果未使能 ESP8266 station 模式，则提示 ERROR
参数说明	<hostname>：主机名称，最长 32 字节	
注意	<ul style="list-style-type: none"> <li>• 本设置不保存到 Flash，重启后将恢复默认值。</li> <li>• ESP8266 Station 默认的主机名称为“ESP_MAC 地址低 3 个字节”。例如，+CWHOSTNAME:&lt;ESP_A378DA&gt;。</li> </ul>	
示例	AT+CWMODE=3 AT+CWHOSTNAME="my_test"	

#### 4.2.31. AT+CWCOUNTRY\_CUR—设置 ESP8266 WiFi 国家码

指令	查询指令： AT+CWCOUNTRY_CUR? 功能：查询 ESP8266 WiFi 国家码。	设置指令： AT+CWCOUNTRY_CUR=<country_policy>,<country_code>,<start_channel>,<total_channel_number> 功能：设置 ESP8266 WiFi 国家码。
响应	+CWCOUNTRY_CUR:<country_policy>,<country_code>,<start_channel>,<total_channel_number> OK 注意： 查询返回值为实际的国家码信息。 如果设置为跟随 AP 切换，则查询值可能与之前的设置值不同。	OK



参数说明	<p>&lt;country_policy&gt;:</p> <ul style="list-style-type: none"> <li>• 0: 根据 AP 自动切换国家信息;</li> <li>• 1: 使用设置的国家信息</li> </ul> <p>&lt;country_codes&gt;: 国家码字符串, 长度最多支持三个字符; 第三个为特殊字符, 在查询时不显示。</p> <p>&lt;start_channel&gt; : 起始信道号</p> <p>&lt;total_channel_number&gt; : 总信道个数</p>
注意	<ul style="list-style-type: none"> <li>• 本设置不保存到 Flash, 重启后将恢复默认值。</li> </ul>
示例	<pre>AT+CWMODE=3 AT+CWCOUNTRY_CUR=1,"CN",1,5</pre>

#### 4.2.32. AT+CWCOUNTRY\_DEF—设置默认的 ESP8266 WiFi 国家码, 并保存到 flash

指令	<p>查询指令:</p> <pre>AT+CWCOUNTRY_DEF?</pre> <p>功能: 查询 ESP8266 WiFi 国家码。</p>	<p>设置指令:</p> <pre>AT+CWCOUNTRY_DEF=&lt;country_policy&gt;,&lt;country_code&gt;,&lt;start_channel&gt;,&lt;total_channel_number&gt;</pre> <p>功能: 设置 ESP8266 WiFi 国家码。</p>
响应	<pre>+CWCOUNTRY_DEF:&lt;country_policy&gt;,&lt;country_code&gt;,&lt;start_channel&gt;,&lt;total_channel_number&gt;</pre> <p>OK</p> <p>注意:</p> <p>查询返回值为保存在 flash 的默认国家码信息。</p> <p>即使设置为跟随 AP 切换, 查询值仍然与之前的设置值一致。</p>	OK
参数说明	<p>&lt;country_policy&gt;:</p> <ul style="list-style-type: none"> <li>• 0: 根据 AP 自动切换国家信息;</li> <li>• 1: 使用设置的国家信息</li> </ul> <p>&lt;country_code&gt;: 国家码字符串, 长度最多支持三个字符; 第三个为特殊字符, 在查询时不显示。</p> <p>&lt;start_channel&gt; : 起始信道号</p> <p>&lt;total_channel_number&gt; : 总信道个数</p>	
注意	<ul style="list-style-type: none"> <li>• 本设置将保存到 Flash 用户参数区。</li> </ul>	
示例	<pre>AT+CWMODE=3 AT+CWCOUNTRY_DEF=1,"CN",1,13</pre>	



# 5. TCP/IP 功能 AT 指令

## 5.1. TCP/IP 指令一览表

指令	描述
AT+CIPSTATUS	查询网络连接信息
AT+CIPDOMAIN	域名解析功能
AT+CIPSTART	建立 TCP 连接, UDP 传输或者 SSL 连接
AT+CIPSSLSIZE	设置 SSL buffer 大小
AT+CIPSSLCONF	配置 ESP SSL client
AT+CIPSEND	发送数据
AT+CIPSENDEX	发送数据, 达到设置长度, 或者遇到字符 \0, 则发送数据
AT+CIPSENDERBUF	数据写入 TCP 发包缓存
AT+CIPBUFRESET	重置计数 (TCP 发包缓存)
AT+CIPBUFSTATUS	查询 TCP 发包缓存的状态
AT+CIPCHECKSEQ	查询写入 TCP 发包缓存的某包是否成功发送
AT+CIPCLOSE	关闭 TCP/UDP/SSL 传输
AT+CIFSR	查询本地 IP 地址
AT+CIPMUX	设置多连接模式
AT+CIPSERVER	设置 TCP 服务器
AT+CIPSERVERMAXCONN	设置服务器允许建立的最大连接数。
AT+CIPMODE	设置透传模式
AT+SAVETRANSLINK	保存透传连接到 Flash
AT+CIPSTO	设置 ESP8266 作为 TCP 服务器时的超时时间
AT+PING	Ping 功能
AT+CIUPDATE	通过 Wi-Fi 升级软件
AT+CIPDINFO	接收网络数据时, +IPD 是否提示对端 IP 和端口
AT+CIPRECVMODE	设置 TCP 接收模式
AT+CIPRECVDATA	TCP 被动接收模式下, 读取缓存的 TCP 数据
AT+CIPRECVLEN	TCP 被动接收模式下, 查询缓存 TCP 数据的长度



AT+CIPSNTPCFG	设置时域和 SNTP 服务器
AT+CIPSNTPTIME	查询 SNTP 时间
AT+CIPDNS_CUR	自定义 DNS 服务器，设置不保存到 flash
AT+CIPDNS_DEF	自定义 DNS 服务器，设置保存到 flash

## 5.2. TCP/IP 指令描述

### 5.2.1. AT+CIPSTATUS—查询网络连接信息

执行指令	AT+CIPSTATUS
响应	STATUS:<stat> +CIPSTATUS:<link ID>,<type>,<remote IP>,<remote port>,<local port>,<tetype>
参数说明	<ul style="list-style-type: none"> <li>• &lt;stat&gt;: ESP8266 Station 接口的状态 <ul style="list-style-type: none"> <li>▶ 2: ESP8266 Station 已连接 AP, 获得 IP 地址</li> <li>▶ 3: ESP8266 Station 已建立 TCP 或 UDP 传输</li> <li>▶ 4: ESP8266 Station 断开网络连接</li> <li>▶ 5: ESP8266 Station 未连接 AP</li> </ul> </li> <li>• &lt;link ID&gt;: 网络连接 ID (0 ~ 4), 用于多连接的情况</li> <li>• &lt;type&gt;: 字符串参数, "TCP" 或者 "UDP"</li> <li>• &lt;remote IP&gt;: 字符串, 远端 IP 地址</li> <li>• &lt;remote port&gt;: 远端端口值</li> <li>• &lt;local port&gt;: ESP8266 本地端口值</li> <li>• &lt;tetype&gt;: <ul style="list-style-type: none"> <li>▶ 0: ESP8266 作为客户端</li> <li>▶ 1: ESP8266 作为服务器</li> </ul> </li> </ul>

### 5.2.2. AT+CIPDOMAIN—域名解析功能

执行指令	AT+CIPDOMAIN=<domain name>
响应	+CIPDOMAIN:<IP address> OK 或者 DNS Fail ERROR
参数说明	<domain name>: 待解析的域名, 可支持长度小于 64 的域名





示例	<pre>AT+CWMODE=1 // set Station mode AT+CWJAP="SSID","password" // access to the internet AT+CIPDOMAIN="iot.espressif.cn" // DNS function</pre>
----	---

### 5.2.3. AT+CIPSTART—建立 TCP 连接, UDP 传输或 SSL 连接

#### 建立 TCP 连接

设置指令	TCP 单连接 (AT+CIPMUX=0) 时: AT+CIPSTART=<type>,<remote IP>,<remote port>[,<TCP keep alive>]	TCP 多连接 (AT+CIPMUX=1) 时: AT+CIPSTART=<link ID>,<type>,<remote IP>,<remote port>[,<TCP keep alive>]
响应	OK 或 ERROR 如果连接已经存在, 则返回 ALREADY CONNECTED	
参数说明	<ul style="list-style-type: none"> <li>• &lt;link ID&gt;: 网络连接 ID (0 ~ 4), 用于多连接的情况</li> <li>• &lt;type&gt;: 字符串参数, 连接类型, "TCP", "UDP"或"SSL"</li> <li>• &lt;remote IP&gt;: 字符串参数, 远端 IP 地址</li> <li>• &lt;remote port&gt;: 远端端口号</li> <li>• [&lt;TCP keep alive&gt;]: TCP keep-alive 侦测时间, 默认关闭此功能               <ul style="list-style-type: none"> <li>▸ 0: 关闭 TCP keep-alive 功能</li> <li>▸ 1 ~ 7200: 侦测时间, 单位为 1s</li> </ul> </li> </ul>	
示例	<pre>AT+CIPSTART="TCP","iot.espressif.cn",8000 AT+CIPSTART="TCP","192.168.101.110",1000</pre> 详细请参考 <a href="#">ESP8266 AT 指令使用示例</a> 。	

#### 建立 UDP 传输

设置指令	单连接模式 (AT+CIPMUX=0) 时: AT+CIPSTART=<type>,<remote IP>,<remote port>[,(<UDP local port>),(<UDP mode>)]	多连接模式 (AT+CIPMUX=1) 时: AT+CIPSTART=<link ID>,<type>,<remote IP>,<remote port>[,(<UDP local port>),(<UDP mode>)]
响应	OK 或 ERROR 如果连接已经存在, 则返回: ALREADY CONNECTED	



参数说明	<ul style="list-style-type: none"> <li>• &lt;link ID&gt;: 网络连接 ID (0 ~ 4), 用于多连接的情况</li> <li>• &lt;type&gt;: 字符串参数, 连接类型, "TCP", "UDP"或"SSL"</li> <li>• &lt;remote IP&gt;: 字符串参数, 远端 IP 地址</li> <li>• &lt;remote port&gt;: 远端端口号</li> <li>• [&lt;UDP local port&gt;]: UDP 传输时, 设置本地端口</li> <li>• [&lt;UDP mode&gt;]: UDP 传输的属性, 若透传, 则必须为 0             <ul style="list-style-type: none"> <li>▸ 0: 收到数据后, 不更改远端目标, 默认值为 0</li> <li>▸ 1: 收到数据后, 改变一次远端目标</li> <li>▸ 2: 收到数据后, 改变远端目标</li> </ul> </li> </ul> <p><b>⚠ 注意:</b> 使用 &lt;UDP mode&gt; 必须先填写 &lt;UDP local port&gt;。</p>
示例	<pre>AT+CIPSTART="UDP", "192.168.101.110", 1000, 1002, 2</pre> <p>详细请参考 <a href="#">ESP8266 AT 指令使用示例</a>。</p>

#### 建立 SSL 连接

设置指令	<pre>AT+CIPSTART=[&lt;link ID&gt;,&lt;type&gt;,&lt;remote IP&gt;,&lt;remote port&gt;[,&lt;TCP keep alive&gt;]</pre>
响应	<pre>OK</pre> <p>或</p> <pre>ERROR</pre> <p>如果连接已经存在, 则返回:</p> <pre>ALREADY CONNECTED</pre>
参数说明	<ul style="list-style-type: none"> <li>• &lt;link ID&gt;: 网络连接 ID (0 ~ 4), 用于多连接的情况</li> <li>• &lt;type&gt;: 字符串参数, 连接类型, "TCP", "UDP"或"SSL"</li> <li>• &lt;remote IP&gt;: 字符串参数, 远端 IP 地址</li> <li>• &lt;remote port&gt;: 远端端口号</li> <li>• [&lt;TCP keep alive&gt;]: TCP keep-alive 侦测时间, 默认关闭此功能             <ul style="list-style-type: none"> <li>▸ 0: 关闭 TCP keep-alive 功能</li> <li>▸ 1 ~ 7200: 侦测时间, 单位为 1s</li> </ul> </li> </ul>
注意	<ul style="list-style-type: none"> <li>• ESP8266 最多仅支持建立 1 个 SSL 连接。</li> <li>• SSL 连接不支持透传。</li> <li>• SSL 需要占用较多空间, 如果空间不足, 会导致系统重启。用户可以使用指令 <code>AT+CIPSSLSIZE=&lt;size&gt;</code> 增大 SSL 缓存。</li> </ul>
示例	<pre>AT+CIPSSLSIZE=4096</pre> <pre>AT+CIPSTART="SSL", "iot.espressif.cn", 8443</pre>



## 5.2.4. AT+CIPSSLSIZE—设置 SSL Buffer 容量

设置指令	AT+CIPSSLSIZE=<size>
响应	OK
参数说明	<size>: SSL buffer 大小, 取值范围: [2048, 4096]。
示例	AT+CIPSSLSIZE=4096

## 5.2.5. AT+CIPSSLCONF—配置 SSL Client

指令	查询指令: AT+CIPSSLCONF? 功能: 查询 ESP32 作为 SSL client 时的认证方式。	设置指令: AT+CIPSSLCONF=<SSL mode> 功能: 配置 ESP32 作为 SSL client 时的认证方式。
响应	+CIPSSLCONF:<SSL mode> OK	OK
参数说明	<ul style="list-style-type: none"> <li>&lt;SSL mode&gt;: SSL 认证方式             <ul style="list-style-type: none"> <li>▶ bit0: 是否加载 cert 和 private key, 以供 server 认证</li> <li>▶ bit1: 是否加载 CA, 认证 server 的 cert 和 private key</li> </ul> </li> </ul>	
注意	<ul style="list-style-type: none"> <li>如需设置本指令, 请在建立 SSL 连接之前调用。本设置对所有 SSL 连接均生效。</li> <li>如需证书认证, 请参考文档 <a href="#">ESP8266 SSL 指南</a> 第 4 章 ESP8266 作为 SSL Client, 生成 SSL 所需证书。             <ul style="list-style-type: none"> <li>- esp_cert_private_key.bin 默认烧录到地址 0xFC000</li> <li>- esp_ca_cert.bin 默认烧录到地址 0xFB000</li> <li>- 用户可以自行修改 <a href="#">user_main.c</a> 中的 SYSTEM_PARTITION_SSL_CLIENT_CERT_PRIVKEY_ADDR 和 SYSTEM_PARTITION_SSL_CLIENT_CA_ADDR 地址, 重新编译指定位置。</li> </ul> </li> <li>本设置将保存到 flash 用户参数区。如果 AT+SAVETRANSLINK 指令设置为 SSL 透传, 那么开机透传时, SSL 连接将根据本配置进行连接。</li> </ul>	
示例	AT+CWMODE=1 // 设置 sta 模式 AT+CWJAP="SSID","PASSWORD" // 连接路由器, 连接网络 AT+CIPSNTPCFG=1,8 // 设置时区 AT+CIPSNTPTIME? // 查询当前时间 AT+CIPSSLCONF=2 AT+CIPSTART="SSL","192.168.3.38",8443	



## 5.2.6. AT+CIPSEND—发送数据

<p>指令</p>	<p>设置指令：</p> <ol style="list-style-type: none"> <li>单连接时：(+CIPMUX=0) AT+CIPSEND=&lt;length&gt;</li> <li>多连接时：(+CIPMUX=1) AT+CIPSEND=&lt;link ID&gt;,&lt;length&gt;</li> <li>如果是 UDP 传输，可以设置远端 IP 和端口： AT+CIPSEND=[&lt;link ID&gt;,&lt;length&gt; &lt;remote IP&gt;,&lt;remote port&gt;]</li> </ol> <p>功能：在普通传输模式时，设置发送数据的长度。</p>	<p>执行指令： AT+CIPSEND</p> <p>功能：在透传模式时，开始发送数据。</p>
<p>响应</p>	<p>发送指定长度的数据。</p> <p>收到此命令后先换行返回 &gt;，然后开始接收串口数据，当数据长度满 &lt;length&gt; 时发送数据，回到普通指令模式，等待下一条 AT 指令。</p> <p>如果未建立连接或连接被断开，返回： ERROR</p> <p>如果数据发送成功，返回： SEND OK</p> <p>如果数据发送失败，返回： SEND FAIL</p>	<p>收到此命令后先换行返回 &gt;。</p> <p>进入透传模式发送数据，每包最大 2048 字节，或者每包数据以 20 ms 间隔区分。</p> <p>当输入单独一包 +++ 时，返回普通 AT 指令模式。发送 +++ 退出透传时，请至少间隔 1 秒再发下一条 AT 指令。</p> <p>本指令必须在开启透传模式以及单连接下使用。</p> <p>若为 UDP 透传，指令 AT+CIPSTART 参数 &lt;UDP mode&gt; 必须为 0。</p>
<p>参数说明</p>	<ul style="list-style-type: none"> <li>&lt;link ID&gt;：网络连接 ID 号 (0 ~ 4)，用于多连接的情况</li> <li>&lt;length&gt;：数字参数，表明发送数据的长度，最大长度为 2048</li> <li>[&lt;remote IP&gt;]：UDP 传输可以设置对端 IP</li> <li>[&lt;remote port&gt;]：UDP 传输可以设置对端口</li> </ul>	<p>-</p>
<p>示例</p>	<p>详细请参考 <a href="#">ESP8266 AT 指令使用示例</a>。</p>	



## 5.2.7. AT+CIPSENDEX—发送数据

设置指令	<p>1. 单连接时: (+CIPMUX=0) AT+CIPSENDEX=&lt;length&gt;</p> <p>2. 多连接时: (+CIPMUX=1) AT+CIPSENDEX=&lt;link ID&gt;,&lt;length&gt;</p> <p>3. 如果是 UDP 传输, 可以设置远端 IP 和端口: AT+CIPSENDEX=[&lt;link ID&gt;,&lt;length&gt;[,&lt;remote IP&gt;,&lt;remote port&gt;]</p> <p>指令功能: 在普通传输模式时, 设置发送数据的长度。</p>
响应	<p>发送指定长度的数据。</p> <p>收到此命令后先换行返回 &gt;, 然后开始接收串口数据, 当数据长度满 length 或者遇到字符 \0 时, 发送数据。</p> <p>如果未建立连接或连接被断开, 返回: ERROR</p> <p>如果数据发送成功, 返回: SEND OK</p> <p>如果数据发送失败, 返回: SEND FAIL</p>
参数说明	<ul style="list-style-type: none"> <li>• &lt;link ID&gt;: 网络连接 ID 号 (0 ~ 4), 用于多连接的情况</li> <li>• &lt;length&gt;: 数字参数, 表明发送数据的长度, 最大长度为 2048</li> <li>• 当接收数据长度满 length 或者遇到字符 \0 时, 发送数据, 回到普通指令模式, 等待下一条 AT 指令。</li> <li>• 用户如需发送 \0, 请转义为 \\0。</li> </ul>

## 5.2.8. AT+CIPSENDERBUF—数据写入 TCP 发包缓存

设置指令	<p>1. 单连接时: (+CIPMUX=0) AT+CIPSENDERBUF=&lt;length&gt;</p> <p>2. 多连接时: (+CIPMUX=1) AT+CIPSENDERBUF=&lt;link ID&gt;,&lt;length&gt;</p>
------	---



响应	<p>&lt;本次 segment ID&gt;,&lt;已成功发送的 segment ID&gt;</p> <p>OK</p> <p>&gt;</p> <ul style="list-style-type: none"> <li>收到此命令后先返回 packet ID, 再换行返回 &gt;, 然后开始接收串口数据, 当数据长度满 length 或者遇到字符 \0 时, 发送数据; 超过 length 的数据丢弃, 并提示 busy。</li> <li>如果未建立连接或并非 TCP 连接或 buffer 满等出错, 返回:</li> </ul> <p>ERROR</p> <ul style="list-style-type: none"> <li>如果某包数据发送成功 <ul style="list-style-type: none"> <li>单连接时, 返回: <p>&lt;segment ID&gt;,SEND OK</p> </li> <li>多连接时, 返回: <p>&lt;link ID&gt;,&lt;segment ID&gt;,SEND OK</p> </li> </ul> </li> <li>如果数据发送失败, 返回: <p>SEND FAIL</p> </li> </ul>
参数说明	<ul style="list-style-type: none"> <li>&lt;link ID&gt;: 网络连接 ID 号 (0 ~ 4), 用于多连接的情况;</li> <li>&lt;segment ID&gt;: uint32, 给每包写入数据分配的 ID, 从 1 开始计数, 每写入一包则自加一, 计数满则重新从 1 计数;</li> <li>&lt;length&gt;: 数据长度, 超过长度的数据则丢弃。</li> </ul>
说明	<p>本指令将数据写入 TCP 发包缓存, 无需等待 SEND OK, 可连续调用; 发送成功后, 会返回数据包 ID 及 SEND OK。</p> <p>在数据没有传入完成时, 传入 +++ 可退出发送, 之前传入的数据将直接丢弃。</p> <p>SSL 连接不支持使用本指令。</p>

### 5.2.9. AT+CIPBUFRESET—重新计数

设置指令	<ol style="list-style-type: none"> <li>单连接时: (+CIPMUX=0) AT+CIPBUFRESET</li> <li>多连接时: (+CIPMUX=1) AT+CIPBUFRESET=&lt;link ID&gt;</li> </ol>
响应	<p>OK</p> <p>如果有数据包未发送完毕, 或者连接不存在, 则返回:</p> <p>ERROR</p>
参数说明	<p>&lt;link ID&gt;: 网络连接 ID 号 (0 ~ 4), 用于多连接的情况。</p>
注意	<p>本指令基于 AT+CIPSENBUFFER 实现功能。</p>



## 5.2.10. AT+CIPBUFSTATUS—查询 TCP 发包缓存的状态

设置指令	<p>1. 单连接时: (+CIPMUX=0)</p> <p>AT+CIPBUFSTATUS</p> <p>2. 多连接时: (+CIPMUX=1)</p> <p>AT+CIPBUFSTATUS=&lt;link ID&gt;</p>
响应	<p>&lt;下次的 segment ID&gt;,&lt;已发送的 segment ID&gt;,&lt;成功发送的segment ID&gt;,&lt;remain buffer size&gt;,&lt;queue number&gt;</p> <p>OK</p>
参数说明	<ul style="list-style-type: none"> <li>• &lt;下次的 segment ID&gt;: 下次调用 AT+CIPSENDERBUF 将分配的 ID;</li> <li>• &lt;已发送的 segment ID&gt;: 已发送的 TCP 数据包 ID;</li> <li>• 仅当 &lt;下次的 segment ID&gt; - &lt;已发送的 segment ID&gt; = 1 的情况下, 可调用 AT+CIPBUFRESET 重置计数。</li> <li>• &lt;成功发送的segment ID&gt;: 成功发送的 TCP 数据包 ID;</li> <li>• &lt;remain buffer size&gt;: TCP 发包缓存剩余的空间;</li> <li>• &lt;queue number&gt;: 底层可用的 queue 数目, 并不可靠, 仅供参考。</li> </ul>
注意	本指令不支持对 SSL 连接使用。
示例	<p>例如, 单连接时 AT+CIPBUFSTATUS 的返回值为:</p> <p>20,15,10,200,7</p> <p>说明:</p> <ul style="list-style-type: none"> <li>• 20: 表示当前数据包序号已经分配到了 19, 下次调用 AT+CIPSENDERBUF 将为数据包分配序号 20;</li> <li>• 15: 表示当前已发送了序号为 15 的数据包, 但并不一定发送成功了;</li> <li>• 10: 表示成功发送到了序号为 10 的数据包;</li> <li>• 200: 表示网络层 TCP 发包缓存剩余的空间为 200 bytes;</li> <li>• 7: 表示当前网络层还剩余 7 个 queue 供数据传输, 仅供参考, 并不可靠; 当 queue 为 0 时, 不允许数据发送。</li> </ul>

## 5.2.11. AT+CIPCHECKSEQ—查询写入 TCP 发包缓存的某包是否发送成功

设置指令	<p>1. 单连接时: (+CIPMUX=0)</p> <p>AT+CIPCHECKSEQ=&lt;segment ID&gt;</p> <p>2. 多连接时: (+CIPMUX=1)</p> <p>AT+CIPCHECKSEQ=&lt;link ID&gt;,&lt;segment ID&gt;</p>
响应	<p>[&lt;link ID&gt;,&lt;segment ID&gt;,&lt;status&gt;</p> <p>OK</p>



参数说明	<ul style="list-style-type: none"> <li>• 最多记录最后的 32 个 segment ID 数据包的状态。</li> <li>• [&lt;link ID&gt;]: 网络连接 ID (0~4), 用于多连接的情况;</li> <li>• &lt;segment ID&gt;: 调用 AT+CIPSEND 写入数据时分配的 ID;</li> <li>• &lt;status&gt;: <ul style="list-style-type: none"> <li>▶ FALSE: 发送失败;</li> <li>▶ TRUE: 发送成功。</li> </ul> </li> </ul>
注意	本指令基于 AT+CIPSEND 实现功能。

### 5.2.12. AT+CIPCLOSEMODE—设置 TCP 连接的断开方式

指令	<ol style="list-style-type: none"> <li>1. 单连接时: (+CIPMUX=0) AT+CIPCLOSEMODE=&lt;enable_abort&gt;</li> <li>2. 多连接时: (+CIPMUX=1) AT+CIPCLOSEMODE=&lt;link ID&gt;,&lt;enable_abort&gt;</li> </ol>
响应	OK
参数说明	<link ID>: 连接 ID 号 <enable_abort>: 断开方式; 0, 正常断开; 1, abort 强制断开。
注意	<ul style="list-style-type: none"> <li>• 默认 TCP 连接为正常断开模式。一般不建议使用 abort 强制断开模式。</li> <li>• 如需设置为 abort 强制断开模式: <ul style="list-style-type: none"> <li>- 需在 TCP 连接建立之后设置;</li> <li>- 设置仅对当次连接有效; 如果当前 TCP 连接断开, 重新建立连接后, 需要重新设置;</li> </ul> </li> <li>• 设置仅对普通 TCP 连接有效, 对 SSL 连接无效。</li> </ul>
示例	AT+CIPSTART=0,"TCP","192.168.3.60",3400 AT+CIPCLOSEMODE=0,1 AT+CIPCLOSE=0

### 5.2.13. AT+CIPCLOSE—关闭 TCP/UDP/SSL 传输

指令	设置指令 (用于多连接的情况) : AT+CIPCLOSE=<link ID> 功能: 关闭 TCP/UDP 传输。	执行指令 (用于单连接的情况) : AT+CIPCLOSE
响应	OK	
参数说明	<link ID>: 需要关闭的连接 ID 号。当 ID 为 5 时, 关闭所有连接。(开启 server 后 ID 为 5 无效)	-





## 5.2.14. AT+CIFSR—查询本地 IP 地址

执行指令	AT+CIFSR
响应	+CIFSR:APIP,<SoftAP IP address> +CIFSR:APMAC,<SoftAP MAC address> +CIFSR:STAIP,<Station IP address> +CIFSR:STAMAC,<Station MAC address> OK
参数说明	<IP address>: ESP8266 SoftAP 的 IP 地址 ESP8266 Station 的 IP 地址 <MAC address>: ESP8266 SoftAP 的 MAC 地址 ESP8266 Station 的 MAC 地址
注意	ESP8266 Station IP 需连上 AP 后, 才可以查询。

## 5.2.15. AT+CIPMUX—设置多连接

指令	查询指令: AT+CIPMUX?	设置指令: AT+CIPMUX=<mode> 功能: 设置连接类型。
响应	+CIPMUX:<mode> OK	OK
参数说明	<mode>: ‣ 0: 单连接模式 ‣ 1: 多连接模式	
注意	<ul style="list-style-type: none"> <li>• 默认为单连接;</li> <li>• 只有非透传模式 (AT+CIPMODE=0), 才能设置为多连接;</li> <li>• 必须在没有连接建立的情况下, 设置连接模式;</li> <li>• 如果建立了 TCP 服务器, 想切换为单连接, 必须关闭服务器 (AT+CIPSERVER=0), 服务器仅支持多连接。</li> </ul>	
示例	AT+CIPMUX=1	

## 5.2.16. AT+CIPSERVER—建立 TCP 服务器

设置指令	AT+CIPSERVER=<mode>[, <port>]
响应	OK



参数说明	<ul style="list-style-type: none"> <li>• &lt;mode&gt;: <ul style="list-style-type: none"> <li>▶ 0: 关闭服务器</li> <li>▶ 1: 建立服务器</li> </ul> </li> <li>• &lt;port&gt;: 端口号, 默认为 333。</li> </ul>
注意	<ul style="list-style-type: none"> <li>• 多连接情况下 (AT+CIPMUX=1), 才能开启 TCP 服务器。</li> <li>• 创建 TCP 服务器后, 自动建立 TCP 服务器监听。</li> <li>• 当有 TCP 客户端接入, 会自动占用一个连接 ID。</li> </ul>
示例	<pre>AT+CIPMUX=1 AT+CIPSERVER=1,1001</pre>

### 5.2.17. AT+CIPSERVERMAXCONN—设置服务器允许建立的最大连接数 \*

指令	查询指令: <pre>AT+CIPSERVERMAXCONN?</pre> 功能: 查询服务器允许建立的最大连接数。	设置指令: <pre>AT+CIPSERVERMAXCONN=&lt;num&gt;</pre> 功能: 设置服务器允许建立的最大连接数。
响应	<pre>+CIPSERVERMAXCONN:&lt;num&gt; OK</pre>	<pre>OK</pre>
参数说明	<num>: 服务器允许建立的最大连接数, 取值范围: [1, 5]。	
注意	如需设置最大连接数, 请在创建服务器之前设置。	
示例	<pre>AT+CIPMUX=1 AT+CIPSERVERMAXCONN=2 AT+CIPSERVER=1,80</pre>	

### 5.2.18. AT+CIPMODE—设置传输模式

指令	查询指令: <pre>AT+CIPMODE?</pre> 功能: 查询传输模式。	设置指令: <pre>AT+CIPMODE=&lt;mode&gt;</pre> 功能: 设置传输模式。
响应	<pre>+CIPMODE:&lt;mode&gt; OK</pre>	<pre>OK</pre>
参数说明	<mode>: <ul style="list-style-type: none"> <li>▶ 0: 普通传输模式</li> <li>▶ 1: 透传模式, 仅支持 TCP 单连接和 UDP 固定通信对端的情况</li> </ul>	
注意	<ul style="list-style-type: none"> <li>• 本设置不保存到 Flash。</li> <li>• 透传模式传输时, 如果连接断开, ESP8266 会不停尝试重连, 此时单独输入 +++ 退出透传, 则停止重连; 普通传输模式则不会重连, 提示连接断开。</li> </ul>	
示例	<pre>AT+CIPMODE=1</pre>	



## 5.2.19. AT+SAVETRANSLINK—保存透传到 Flash

## 保存透传（TCP 单连接）到 Flash

设置指令	AT+SAVETRANSLINK=<mode>,<remote IP or domain name>,<remote port>[,<type>,<TCP keep alive>]
响应	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;mode&gt;: <ul style="list-style-type: none"> <li>▸ 0: 取消开机透传</li> <li>▸ 1: 保存开机进入透传模式</li> </ul> </li> <li>• &lt;remote IP&gt;: 远端 IP 或者域名</li> <li>• &lt;remote port&gt;: 远端端口</li> <li>• [&lt;type&gt;] (选填参数): TCP 或者 UDP, 缺省默认为 TCP</li> <li>• [&lt;TCP keep alive&gt;] (选填参数): TCP keep-alive 侦测, 缺省默认关闭此功能 <ul style="list-style-type: none"> <li>▸ 0: 关闭 TCP keep-alive 功能</li> <li>▸ 1 ~ 7200: 侦测时间, 单位为秒</li> </ul> </li> </ul>
注意	<ul style="list-style-type: none"> <li>• 本设置将透传模式及建立的 TCP 连接均保存在 Flash system parameter 区域, 下次上电自动建立 TCP 连接并进入透传。</li> <li>• 只要远端 IP, 端口的值符合规范, 本设置就会被保存到 Flash。</li> </ul>
示例	AT+SAVETRANSLINK=1,"192.168.6.110",1002,"TCP"

## 保存透传（UDP 传输）到 Flash

设置指令	AT+SAVETRANSLINK=<mode>,<remote IP>,<remote port>,<type>[,<UDP local port>]
响应	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;mode&gt;: <ul style="list-style-type: none"> <li>▸ 0: 取消开机透传</li> <li>▸ 1: 保存开机进入透传模式</li> </ul> </li> <li>• &lt;remote IP&gt;: 远端 IP</li> <li>• &lt;remote port&gt;: 远端端口</li> <li>• &lt;type&gt;: UDP, 缺省默认为 TCP</li> <li>• [&lt;UDP local port&gt;] (选填参数): 开机进入 UDP 传输时, 使用的本地端口</li> </ul>
注意	<ul style="list-style-type: none"> <li>• 本设置将透传模式及建立的 UDP 传输均保存在 Flash 用户参数区, 下次上电自动建立 UDP 传输并进入透传。</li> <li>• 只要远端 IP, 端口的数值符合规范, 本设置就会被保存到 Flash。</li> </ul>
示例	AT+SAVETRANSLINK=1,"192.168.6.110",1002,"UDP",1005



## 5.2.20. AT+CIPSTO—设置 TCP 服务器超时时间

指令	查询指令： AT+CIPSTO? 功能：查询 TCP 服务器超时时间。	设置指令： AT+CIPSTO=<time> 功能：设置 TCP 服务器超时时间。
响应	+CIPSTO:<time> OK	OK
参数说明	<time>: TCP 服务器超时时间, 取值范围 0 ~ 7200s。	
注意	<ul style="list-style-type: none"> <li>ESP8266 作为 TCP 服务器, 会断开一直不通信直至超时的 TCP 客户端连接。</li> <li>如果设置 AT+CIPSTO=0, 则永远不会超时, 不建议这样设置。</li> </ul>	
示例	AT+CIPMUX=1 AT+CIPSERVER=1,1001 AT+CIPSTO=10	

## 5.2.21. AT+PING—Ping 功能

设置指令	AT+PING=<IP> 功能：ping 功能。	
响应	+<time> OK 或 +timeout ERROR	
参数说明	<ul style="list-style-type: none"> <li>&lt;IP&gt;: 字符串参数, IP 地址</li> <li>&lt;time&gt;: ping 响应时间</li> </ul>	
示例	AT+PING="192.168.1.1" AT+PING=" <a href="http://www.baidu.com">www.baidu.com</a> "	

## 5.2.22. AT+CIUPDATE—通过 Wi-Fi 升级软件

执行指令	AT+ CIUPDATE 功能：软件升级。	
响应	+CIPUPDATE:<n> OK	



参数说明	<p>&lt;n&gt;:</p> <ul style="list-style-type: none"> <li>• 1: 找到服务器</li> <li>• 2: 连接到服务器</li> <li>• 3: 获得软件版本</li> <li>• 4: 开始升级</li> </ul>
说明	<ul style="list-style-type: none"> <li>• 升级过程由于网络条件的好坏, 有快慢差异;</li> <li>• 升级失败会提示 ERROR, 请耐心等待。</li> </ul>
注意	<ul style="list-style-type: none"> <li>• 若直接使用乐鑫提供的 AT BIN (<i>/ESP8266_NONOS_SDK/bin/at</i>), 本指令将从 Espressif Cloud 下载 AT 固件升级。</li> <li>• 若用户自行编译 AT 源代码, 则请自行实现 AT+CIUPDATE 指令的升级功能, 乐鑫提供本地升级的 Demo 作为参考 (<i>/ESP8266_NONOS_SDK/example/at</i>) 。</li> <li>• 建议升级 AT 固件后, 调用 AT+RESTORE 恢复出厂设置, 重新初始化。</li> </ul>

### 5.2.23. AT+CIPDINFO—接收网络数据时是否提示对端 IP 和端口

设置指令	<p>AT+CIPDINFO=&lt;mode&gt;</p> <p>功能: 接收网络数据时, +IPD 是否提示对端 IP 和端口。</p>
响应	OK
参数说明	<p>&lt;mode&gt;: 字符串参数, IP 地址</p> <ul style="list-style-type: none"> <li>• 0: 不显示对端 IP 和端口</li> <li>• 1: 显示对端 IP 和端口</li> </ul>
示例	AT+CIPDINFO=1

### 5.2.24. +IPD—接收网络数据

指令	<p>单连接时:</p> <p>(+CIPMUX=0)+IPD,&lt;len&gt;[,&lt;remote IP&gt;,&lt;remote port&gt;]:&lt;data&gt;</p>	<p>多连接时:</p> <p>(+CIPMUX=1)+IPD,&lt;link ID&gt;,&lt;len&gt;[,&lt;remote IP&gt;,&lt;remote port&gt;]:&lt;data&gt;</p>
参数说明	<p>此指令在普通指令模式下有效, ESP8266 接收到网络数据时向串口发送 +IPD 和数据。</p> <ul style="list-style-type: none"> <li>• [&lt;remote IP&gt;]: 网络通信对端 IP, 由指令 AT+CIPDINFO=1 使能显示</li> <li>• [&lt;remote port&gt;]: 网络通信对端端口, 由指令 AT+CIPDINFO=1 使能</li> <li>• &lt;link ID&gt;: 收到网络连接的 ID 号</li> <li>• &lt;len&gt;: 数据长度</li> <li>• &lt;data&gt;: 收到的数据</li> </ul>	



## 5.2.25. AT+CIPRECVMODE—设置 TCP 连接的数据接收方式

指令	设置指令： AT+CIPRECVMODE=<mode>	查询指令： AT+CIPRECVMODE?
响应	+CIPRECVMODE:<mode> OK	OK
参数说明	<p>&lt;mode&gt;: TCP 连接的数据接收方式, 默认为主动模式</p> <ul style="list-style-type: none"> <li>• 0: 主动模式; 收到 TCP 数据时, ESP8266 通过 +IPD 主动将数据通过串口发给 MCU;</li> <li>• 1: 被动模式; ESP8266 默认使用 2920 bytes 的 buffer 用于缓存接收到的 TCP 数据, 被动等待 MCU 读取; 接收 buffer 满后, 将阻塞对端 TCP 发送数据。</li> </ul>	
注意	<ul style="list-style-type: none"> <li>• 本设置仅针对 TCP 连接的普通传输模式有效, 对 UDP、SSL 连接和透传模式无效。</li> <li>• 被动模式开启后, 收到 TCP 数据时: <ul style="list-style-type: none"> <li>- 多连接 (AT+CIPMUX=1) 将提示: +IPD,&lt;link ID&gt;,&lt;len&gt;</li> <li>- 单连接 (AT+CIPMUX=0) 将提示: +IPD,&lt;len&gt;</li> <li>- 其中 &lt;len&gt; 表示 buffer 中缓存的总数据长度</li> </ul> </li> </ul>	
示例	AT+CIPRECVMODE=1	

## 5.2.26. AT+CIPRECVDATA—被动接收模式时, 读取缓存的 TCP 数据

设置指令	<ul style="list-style-type: none"> <li>• 对于单连接模式 (AT+CIPMUX=0) : AT+CIPRECVDATA=&lt;len&gt;</li> <li>• 对于多连接模式 (AT+CIPMUX=1) : AT+CIPRECVDATA=&lt;link_id&gt;,&lt;len&gt;</li> </ul>
响应	+CIPRECVDATA,<actual_len>:<data> OK
参数说明	<p>&lt;link_id&gt;: 多连接的连接 ID 号</p> <p>&lt;len&gt;: 读取的数据长度, 每次最多可读取 2048 字节</p> <p>&lt;actual_len&gt;: 实际读到的数据长度</p> <p>&lt;data&gt;: 读到的 TCP 数据</p>
注意	<ul style="list-style-type: none"> <li>• 如果在缓存了 TCP 数据的情况下, 当前连接断开, 缓存的数据仍然保留, 允许 MCU 读取, 直至创建新的连接; 如果新的同 ID 连接创建, 则前一次连接缓存的数据将被丢弃。</li> </ul>
示例	AT+CIPRECVMODE=1 //例如, 收到提示 +IPD,0,100 第 0 号连接接收到 100 字节数据 AT+CIPRECVDATA=0,100 // 从 0 号连接读取 100 字节数据



## 5.2.27. AT+CIPRECLEN—被动接收模式时，查询缓存 TCP 数据的长度

查询指令	AT+CIPRECLEN?
响应	+CIPRECLEN:<data length of link0>,<data length of link1>,<data length of link2>,<data length of link3>,<data length of link4> OK
参数说明	<data length of link>: 被动接收模式时，该连接缓存的 TCP 数据长度
示例	AT+CIPRECLEN? +CIPRECLEN:100,,,,, OK

## 5.2.28. AT+CIPSNTPCFG—设置时域和 SNTP 服务器

指令	查询指令： AT+CIPSNTPCFG?	设置指令： AT+CIPSNTPCFG=<enable>[,<timezone>][,<SNTP server0>,<SNTP server1>,<SNTP server2>]
响应	+CIPSNTPCFG:<enable>,<timezone>,<SNTP server0>[,<SNTP server1>,<SNTP server2>] OK	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;enable&gt; : <ul style="list-style-type: none"> <li>▸ 0: SNTP 未使能;</li> <li>▸ 1: SNTP 使能。</li> </ul> </li> <li>• &lt;timezone&gt;: 时域，范围：[-11,13]; 若 SNTP 使能，此参数必填；否则，无需填写；</li> <li>• &lt;SNTP server0&gt;: 第一个 SNTP 服务器，可不填；</li> <li>• &lt;SNTP server1&gt;: 第二个 SNTP 服务器，可不填；</li> <li>• &lt;SNTP server2&gt;: 第三个 SNTP 服务器，可不填。</li> </ul>	
说明	设置指令若未填写 SNTP server，则默认使用 "cn.ntp.org.cn","ntp.sjtu.edu.cn","us.pool.ntp.org"	
示例	AT+CIPSNTPCFG=1,8,"cn.ntp.org.cn","ntp.sjtu.edu.cn","us.pool.ntp.org"	

## 5.2.29. AT+CIPSNTPTIME—查询 SNTP 时间

查询指令	AT+CIPSNTPTIME?
响应	+CIPSNTPTIME:<time> OK



参数说明	<p>&lt;time&gt;: 通过 SNTP 查询到的时间。</p> <p>例如, 查询返回时间格式如下:</p> <p>+CIPSNTPTIME:Thu Aug 04 14:48:05 2016</p>
示例	<p>AT+CWMODE=1 //设置为 station 模式</p> <p>AT+CWJAP="DemoAP","password" //连接到路由器, 连接网络</p> <p>AT+CIPSNTPCFG=1,8 //设置时区</p> <p>AT+CIPSNTPTIME? //查询时间</p>

### 5.2.30. AT+CIPDNS\_CUR—自定义 DNS 服务器, 不保存到 Flash

指令	<p>功能: 查询当前使用的 DNS 服务器</p> <p>查询指令:</p> <p>AT+CIPDNS_CUR?</p>	<p>功能: 设置自定义 DNS 服务器</p> <p>设置指令:</p> <p>AT+CIPDNS_CUR=&lt;enable&gt;[,&lt;DNS server0&gt;,&lt;DNS server1&gt;]</p>
响应	<p>[+CIPDNS_CUR:&lt;DNS server0&gt;]</p> <p>[+CIPDNS_CUR:&lt;DNS server1&gt;]</p> <p>OK</p>	OK
参数说明	<ul style="list-style-type: none"> <li>• &lt;enable&gt; : <ul style="list-style-type: none"> <li>▸ 0: 不使能自定义 DNS 服务器;</li> <li>▸ 1: 使能自定义 DNS 服务器。</li> </ul> </li> <li>• &lt;DNS server0&gt;: 第一个 DNS 服务器, 可不填;</li> <li>• &lt;DNS server1&gt;: 第二个 DNS 服务器, 可不填。</li> </ul>	
说明	<ul style="list-style-type: none"> <li>• 对于指令 AT+CIPDNS_CUR=1 (即设置使能自定义 DNS 服务器, 但未填写 &lt;DNS server&gt; 参数), 则默认使用 "208.67.222.222" 作为 DNS 服务器。</li> <li>• 对于指令 AT+CIPDNS_CUR=0 (即不使能自定义 DNS 服务器), 则默认使用 "208.67.222.222" 作为 DNS 服务器。并且, 在与路由器交互的过程中, DNS 服务器可能随着路由器的配置更改。</li> </ul>	
示例	AT+CIPDNS_CUR=1,"208.67.220.220"	

### 5.2.31. AT+CIPDNS\_DEF—自定义 DNS 服务器, 保存到 Flash

指令	<p>功能: 查询保存到 flash 的 DNS 服务器</p> <p>查询指令:</p> <p>AT+CIPDNS_DEF?</p>	<p>功能: 设置自定义 DNS 服务器</p> <p>设置指令:</p> <p>AT+CIPDNS_DEF=&lt;enable&gt;[,&lt;DNS server0&gt;,&lt;DNS server1&gt;]</p>
----	---	---





响应	[+CIPDNS_DEF:<DNS server0>] [+CIPDNS_DEF:<DNS server1>] OK	OK
参数说明	<ul style="list-style-type: none"><li>• &lt;enable&gt; :<ul style="list-style-type: none"><li>▸ 0: 不使能自定义 DNS 服务器;</li><li>▸ 1: 使能自定义 DNS 服务器。</li></ul></li><li>• &lt;DNS server0&gt;: 第一个 DNS 服务器, 可不填;</li><li>• &lt;DNS server1&gt;: 第二个 DNS 服务器, 可不填。</li></ul>	
说明	<ul style="list-style-type: none"><li>• 本设置将保存到 Flash 用户参数区。</li><li>• 对于指令 AT+CIPDNS_DEF=1 (即设置使能自定义 DNS 服务器, 但未填写 &lt;DNS server&gt; 参数), 则默认使用 "208.67.222.222" 作为 DNS 服务器。</li><li>• 对于指令 AT+CIPDNS_DEF=0 (即不使能自定义 DNS 服务器), 则默认使用 "208.67.222.222" 作为 DNS 服务器。并且, 在与路由器交互的过程中, DNS 服务器可能随着路由器的配置更改。</li></ul>	
示例	AT+CIPDNS_DEF=1,"208.67.220.220"	



## A.

## 附录 A

以下 ESP8266 AT 指令会保存设置到 Flash:

指令	示例
保存在 Flash 用户参数区	
AT+UART_DEF	AT+UART_DEF=115200,8,1,0,3
AT+CWDHCP_DEF	AT+CWDHCP_DEF=1,1
AT+CIPSTAMAC_DEF	AT+CIPSTAMAC_DEF="18:fe:35:98:d3:7b"
AT+CIPAPMAC_DEF	AT+CIPAPMAC_DEF="1a:fe:36:97:d5:7b"
AT+CIPSTA_DEF	AT+CIPSTA_DEF="192.168.6.100"
AT+CIPAP_DEF	AT+CIPAP_DEF="192.168.5.1"
AT+CWDHCPS_DEF	AT+CWDHCPS_DEF=1,3,"192.168.4.10","192.168.4.15"
AT+SAVETRANSLINK	AT+SAVETRANSLINK_DEF=1,"192.168.6.10",1001
AT+CIPDNS_DEF	AT+CIPDNS_DEF=1,"208.67.220.220"
AT+SYMSMSG_DEF	AT+SYMSMSG_DEF=3
AT+CWCOUNTRY_DEF	AT+CWCOUNTRY_DEF=1,"CN",1,5
AT+CIPSSLCCONF	AT+CIPSSLCCONF=2
保存在 Flash 系统参数区	
AT+CWMODE_DEF	AT+CWMODE_DEF=3
AT+CWJAP_DEF	AT+CWJAP_DEF="abc","0123456789"
AT+CWSAP_DEF	AT+CWSAP_DEF="ESP8266","12345678",5,3
AT+CWAUTOCONN	AT+CWAUTOCONN=1

**⚠ 注意:**

- 以上指令设置时, 会先读取 Flash 中的原配置, 仅新配置与原配置不同时, 才写 Flash 保存新配置。
- 对于 512 KB + 512 KB Flash Map, 用户参数区为 0x7C000 ~ 0x80000, 16 KB;
- 对于 1024 KB + 1024 KB Flash Map: 用户参数区为 0xFC000 ~ 0x100000, 16 KB;
- 系统参数区始终为 Flash 的最后 16 KB。



# B.

# 附录 B

ESP8266 AT 指令中的提示信息说明如下：

提示信息	说明
ready	AT 固件成功启动
ERROR	指令输入错误，或者指令执行出错
WIFI CONNECTED	ESP8266 station 连接到 AP
WIFI GOT IP	ESP8266 station 获取到 IP 地址
WIFI DISCONNECT	ESP8266 station 的 WiFi 连接断开
busy s...	busy sending，表示系统正在发送数据的过程中，无法响应当前输入
busy p...	busy processing，表示系统正在处理前一条指令，无法响应当前输入
<conn_id>,CONNECT	建立了 <conn_id> 号网络连接
<conn_id>,CLOSED	<conn_id> 号网络连接断开
+IPD	接收到网络数据
+STA_CONNECTED:<sta_mac>	有 station 连入 ESP8266 softAP
+DIST_STA_IP:<sta_mac>,<sta_ip>	ESP8266 softAP 给连入的 station 分配 IP 地址
+STA_DISCONNECTED:<sta_mac>	station 从 ESP8266 softAP 断开连接



# C.

# Q&A

如遇到 AT 使用异常，请发邮件至[乐鑫技术支持](#)，附上如下信息：

- AT 软件的版本号，指令 AT+GMR 可获取版本信息；
- 硬件模块的信息，例如：ESP-WROOM-02；
- 详细的测试步骤说明，例如：

```
AT+CWMODE_CUR=1
OK
AT+GMR
AT version:0.23.0.0(Apr 24 2015 21:11:01)
SDK version:1.0.1
compile time:Apr 24 2015 21:19:31
OK
AT+CIPSTAMAC_DEF="14:CF:11:22:33:05"
OK
```

- 如能提供 log 打印信息，请附上异常 log 信息，例如以下截屏：

```
ets Jan 8 2013,rst cause: 1, boot mode: (3,3)
load 0x40100000, len 26336, room 16
tail 0
chksum 0xde
load 0x3ffe8000, len 5672, room 8
tail 0
chksum 0x69
load 0x3ffe9630, len 8348, room 8
tail 4
chksum 0xcb
csum 0xcb
SDK version: 0.9.1
addr not ack when tx write cmd
mode : sta(18: fe: 34: 97: d5: 7b) + softAP(1a: fe: 34: 97: d5: 7b)
```



#### 免责声明和版权公告

本文中的信息，包括供参考的 URL 地址，如有变更，恕不另行通知。

文档“按现状”提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档不负任何责任，包括使用本文档内信息产生的侵犯任何专利权行为的责任。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。

Wi-Fi 联盟成员标志归 Wi-Fi 联盟所有。蓝牙标志是 Bluetooth SIG 的注册商标。文中提到的所有商标名称、商标和注册商标均属其各自所有者的财产，特此声明。

版权归© 2018 乐鑫所有。保留所有权利。